Searches for nonresonant Higgs boson pair production and long-lived particles at the LHC & machine-learning solutions for the High-Luminosity LHC era

Thesis by
Thong Quang Nguyen

In Partial Fulfillment of the Requirements for the
Degree of
Doctor of Philosophy

CALIFORNIA INSTITUTE OF TECHNOLOGY
Pasadena, California

2022
Defended 23 September 2021

© 2021

Thong Quang Nguyen
ORCID: 0000-0003-3954-5131

All rights reserved except where otherwise noted

ACKNOWLEDGEMENTS

Let me start by thanking my advisor, Maria Spiropulu, for her tremendous support throughout my journey. Her relentless spirit and dedication to scientific innovation have been my source of inspiration for the past six years, and will be for many years to come.

I thank Maurizio Pierini, the guardian angel who took me under his wings during my three and a half years at CERN. I was truly fortunate to be in part of his grand vision. His support and wisdom made me a better scientist, while his direct and constructive feedback made me a better human being.

My research career would not have existed without my undergraduate advisor, Joseph Izen, who spent four hours in his office on a summer day in 2012 encouraging me to become a physicist—a decision I never regret. I am forever in debt to his patience and thorough guidance throughout my undergraduate years and beyond.

Dustin[1]

My days at Caltech became much brighter thanks to these friends: Zhicai Zhang, for taking me on wild journeys and encouraging me to try out extreme sports, Si Xie, for his wisdom on how to navigate this world, Justas Balcas, for the daily afternoon coffee, Christina Wang, Irene Dutta, Jiajing Mao, Olmo Cerri, for the dinner parties and office banter, and Pei-Yu Tsai. You all made my twenties memorable and worthwhile.

I thank Dmytro Kovalskyi for showing me true leadership in CMS Computing.

To my second home on Rue de Berne—Benoit Perrin, Christian Tasso, Federica Settimi, and Eleonora Guzzi: you warmed my heart during the chilly days in Switzerland, and will always be in my mind when I make carbonara while singing *Shallow*.

My Vietnamese friends in Geneva: Nguyen Kim Cong and Nguyen Thanh Thuy, thank you for making me Vietnamese foods when I was homesick, and fondue when I craved it; Tran Thu Tra, for helping me with accommodation and taking me on beautiful hiking trips in the Swiss Alps.

[1] Anderson, who mentored me during my first year at CERN when I got lost. Thank you for the daily mental support until this day. You know why you are buried in the footnote.

A part of my PhD was spent in industry, where I am grateful to many people: Dave DeBarr, for his dedicated guidance at Microsoft and continuous encouragement during my writing of this thesis; Bradley Zamft, for sharing with me his ambitious vision at Google X. Daily meetings with Joey Havelick, Mathias Voges, Mina Aiken, and Brad had been so joyful that made my residency a truly happy period of my life.

There are people whom I have never met in real life, but made such a big impact on my PhD career with their online educational resources. I would like to thank in particular Jeremy Howard and Joshua Starmer, from whom I learned a tremendous amount of machine learning and statistics. Their dedication and efforts inspire me to contribute towards a future of free education for the next generations.

Lastly, I thank my parents for trusting me to make my own life decisions since I was little. Their unconditional love made me who I am today, for which I am forever grateful.

ABSTRACT

This thesis presents two physics analyses using 137 fb^{-1} proton-proton collision data collected by the CMS experiment at $\sqrt{s}$ = 13 TeV, along with a series of machine-learning solutions to extend the physics program at the LHC and to address the computational challenges in the High-Luminosity LHC era. The first analysis searches for nonresonant Higgs boson pair production in final states with two photons and two bottom quarks, with no significant deviation from the background-only hypothesis observed. The observed (expected) upper limit on the product of the Higgs boson pair production cross section and branching fraction into $b\bar{b}\gamma\gamma$ is 0.67 (0.45) fb, corresponding to 7.7 (5.2) times the Standard Model prediction. The modifier of the Higgs trilinear self-coupling is constrained within the range $-3.3 < \kappa_\lambda < 8.5$. The modifier for coupling between a pair of Higgs bosons and a pair of vector bosons, along with the 2-dimensional constraint of the modifiers of Higgs self-coupling and Yukawa coupling, are also reported. A graph-based algorithm to identify boosted $H \rightarrow b\bar{b}$ jets to improve future Higgs search is presented. The second analysis searches for long-lived supersymmetry particles decaying to photons and gravitinos in the context of gauge-mediated supersymmetry breaking model. Results are presented in terms of 95% confidence level expected exclusion limits on the masses and proper decay lengths of the neutralino, which exceed the limits from the previous searches by up to 100 GeV for the neutralino mass and by five times for the neutralino proper decay length. A strategy for model-independent new physics searches is presented with an anomaly trigger based on unsupervised learning algorithms that can be deployed in both the high-level trigger and the Level-1 trigger in CMS. Three other machine-learning solutions are presented to address the computational challenges in the HL-LHC era: a layer based on multi-modal deep neural networks that can reduce the false-positive events selected by the trigger by over one order of magnitude while retaining 99% of signal events, a full-event simulation algorithm based on recurrent generative adversarial networks that has potential to replace traditional simulation method while being five orders of magnitude faster, and a fast simulation algorithm for specific analyses based on encoder-decoder architecture that would result in about an order-of-magnitude reduction in computing and storage requirements for the collision simulation workflow.

PUBLISHED CONTENT AND CONTRIBUTIONS

[1] Cheng Chen, Olmo Cerri, Thong Q. Nguyen, Jean-Roch Vlimant, and Maurizio Pierini. "Data augmentation at the LHC through analysis-specific fast simulation with deep learning." In: *Computing and Software for Big Science* 5.15 (2021). DOI: 10.1007/s41781-021-00060-4.
T.Q.N participated in the algorithm design and implementation in this paper.

[2] CMS Collaboration. "Search for nonresonant Higgs boson pair production in final states with two bottom quarks and two photons in proton-proton collisions at $\sqrt{s}$ = 13 TeV." In: *Journal of High Energy Physics* 2021.257 (2021). DOI: 10.1007/JHEP03(2021)257.
T.Q.N was a major contributor of this analysis. He designed, implemented, and validated the algorithm used in the reduction of the resonant background of single Higgs production associated with a top quark pair.

[3] Ekaterina Govorkova et al. *Autoencoders on FPGAs for real-time, unsupervised new physics detection at 40 MHz at the Large Hadron Collider.* 2021. arXiv: 2108.03986 [physics.ins-det].
T.Q.N initiated this project and participated in the design of the models used in this paper.

[4] Oliver Knapp, Olmo Cerri, Günther Dissertori, Thong Q. Nguyen, Maurizio Pierini, and Jean-Roch Vlimant. "Adversarially learned anomaly detection on CMS open data: Re-discovering the top quark." In: *The European Physical Journal Plus* 136.236 (2021). DOI: 10.1140/epjp/s13360-021-01109-4.
T.Q.N participated in data processing, uncertainty quantification study, and contributed to writing the manuscript for publication.

[5] CMS Collaboration. "The Phase-2 upgrade of the CMS Level-1 trigger." In: CERN-LHCC-2020-004. CMS-TDR-021 (2020). Technical Design Report. URL: https://cds.cern.ch/record/2714892.
T.Q.N was a major contributor of Section 3.7.2. "Machine learning based trigger algorithms." He designed and implemented the autoencoder architecture, and participated in the validation study of the algorithm.

[6] Eric A. Moreno et al. "JEDI-net: a jet identification algorithm based on interaction networks." In: *European Physical Journal C* 80.58 (2020). DOI: 10.1140/epjc/s10052-020-7608-4.
T.Q.N participated in the algorithm design, validation study, and contributed to writing the manuscript for publication.

[7] Eric A. Moreno, Thong Q. Nguyen, Jean-Roch Vlimant, Olmo Cerri, Harvey B. Newman, Avikar Periwal, Maria Spiropulu, Javier M. Duarte, and Maurizio Pierini. "Interaction networks for the identification of boosted $H \to b\bar{b}$ decays." In: *Physical Review D* 102.012010 (2020). DOI: 10.1103/PhysRevD.102.012010.

T.Q.N participated in the algorithm design, validation study, and contributed to writing the manuscript for publication.

[8] Wozniak, Kinga Anna et al. "New physics agnostic selections for new physics searches." In: *EPJ Web Conf.* 245 (2020), p. 06039. DOI: 10.1051/epjconf/202024506039.
T.Q.N designed the model architectures and developed the statistical methods for this project.

[9] Jesús Arjona Martínez, Thong Q. Nguyen, Maurizio Pierini, Maria Spiropulu, and Jean-Roch Vlimant. "Particle generative adversarial networks for full-event simulation at the LHC and their application to pileup description." In: *Journal of Physics: Conference Series* 1525.012081 (2019). DOI: 10.1088/1742-6596/1525/1/012081.
T.Q.N was one of the leading contributors of the project. He designed, implemented, and optimized the model, ran the experiment, and contributed to writing the manuscript for publication.

[10] Olmo Cerri, Thong Q. Nguyen, Maurizio Pierini, Maria Spiropulu, and Jean-Roch Vlimant. "Variational autoencoders for new physics mining at the Large Hadron Collider." In: *Journal of High Energy Physics* 2019.36 (2019). DOI: 10.1007/JHEP05(2019)036.
T.Q.N was one of the leading contributors of the project. He designed and implemented different flavors of variational autoencoder algorithm, and contributed to writing the manuscript for publication.

[11] Thong Q. Nguyen et al. "Topology classification with deep learning to improve real-time event selection at the LHC." In: *Computing and Software for Big Science* 3.12 (2019). DOI: 10.1007/s41781-019-0028-1.
T.Q.N was the leading contributor of the project. He processed the data, designed, implemented, and optimized the models, performed the validation study, and contributed to writing the manuscript for publication. Permission has been secured to use the material.

[12] Jean-Roch Vlimant et al. "Large-scale distributed training applied to generative adversarial networks for calorimeter simulation." In: *EPJ Web of Conferences* 214.06025 (2019). DOI: 10.1051/epjconf/201921406025.
T.Q.N was a major contributor to the development of the large-scale distributed training system used in this project.

TABLE OF CONTENTS

Acknowledgements . . . iii
Abstract . . . v
Published Content and Contributions . . . vi
Table of Contents . . . vii
List of Illustrations . . . xi
List of Tables . . . xxxi

Introduction **1**

I Theoretical foundation **4**
Chapter I: The Standard Model of particle physics . . . 5
Chapter II: Physics beyond the Standard Model . . . 11

II Experimental apparatus **16**
Chapter III: The Large Hadron Collider . . . 17
Chapter IV: The Compact Muon Solenoid experiment . . . 21
4.1 The superconducting solenoid . . . 23
4.2 The tracker . . . 23
4.3 The electromagnetic calorimeter . . . 25
4.4 The hadron calorimeter . . . 27
4.5 The muon system . . . 29
4.6 The trigger system . . . 31
4.7 The globally distributed data processing system . . . 36

III Physics of the Higgs at the LHC **39**
Chapter V: Search for nonresonant Higgs boson pair production in final states with two bottom quarks and two photons . . . 40
5.1 Event samples . . . 43
5.2 Physics object reconstruction . . . 44
5.3 Analysis strategy . . . 45
5.4 Resonant background reduction . . . 48
5.5 Nonresonant background reduction . . . 53
5.6 Event categorization . . . 58
5.7 Combination of the HH and $t\bar{t}H$ signals to constrain κ_λ and κ_t . . . 61
5.8 Signal modeling . . . 63

5.9 Background modeling 65
5.10 Systematic uncertainties 67
5.11 Results 69
5.12 Summary 76
Chapter VI: Boosting future Higgs searches with boosted H $\rightarrow$ $b\bar{b}$ jet identification based on graph interaction networks 79
6.1 Introduction 79
6.2 Related work 83
6.3 Dataset description 83
6.4 JEDI-net: Jet identification algorithm based on interaction networks . 94
6.5 Modified JEDI-net for the identification of boosted H $\rightarrow$ $b\bar{b}$ decays . 104
6.6 Decorrelation with the jet mass 109
6.7 Deep double-b tagger models 111
6.8 Results 112
6.9 Summary 115

IV New physics searches at the LHC 118
Chapter VII: Search for long-lived particles with delayed photon signature on Run 2 data in CMS 119
7.1 Event samples 121
7.2 Physics object reconstruction 122
7.3 Delayed photon identification 124
7.4 Photon time reconstruction 133
7.5 Event selection 137
7.6 Data-driven background estimation 141
7.7 Systematic uncertainties 146
7.8 Results 147
7.9 Summary 160
Chapter VIII: Toward model-independent new physics searches with unsupervised learning 165
8.1 Introduction 165
8.2 Related work 167
8.3 Data samples 168
8.4 Model description 172
8.5 Results with VAE 183
8.6 Deployment in high-level triggers 188
8.7 Deployment in Level-1 triggers 190
8.8 An alternative model 192
8.9 Summary 198

V Machine-learning solutions for the High-Luminosity LHC era 201
Chapter IX: The Phase-II upgrade 202

9.1 The challenges of CMS computing in the HL-LHC era 203
9.2 Heterogeneous platform for the future of CMS computing 203
9.3 Machine-learning solutions for the HL-LHC era 205
Chapter X: Trigger improvements with topology classifier 207
10.1 Introduction 207
10.2 Dataset 210
10.3 Model description 215
10.4 Results 217
10.5 Impact on other topologies 221
10.6 Robustness study 223
10.7 An alternative use case 224
10.8 Related works 225
10.9 Summary 227
Chapter XI: Generative adversarial networks for full-event simulation 229
11.1 Introduction 229
11.2 Pileup simulation 230
11.3 Dataset 230
11.4 Network architectures 231
11.5 Experimental evaluation 235
11.6 Distributed training 239
11.7 Summary 241
Chapter XII: Generative models for analysis-specific fast simulation 242
12.1 Introduction 242
12.2 Benchmark dataset 247
12.3 Model description and training 249
12.4 Results 250
12.5 Computing resources 255
12.6 Resource utilization for a standard GEANT4-based generation workflow 259
12.7 Further validation of the deep learning generation workflow 259
12.8 Scaling with dataset size 260
12.9 Summary 261

Conclusion **265**
Bibliography 267

LIST OF ILLUSTRATIONS

Number *Page*

1.1 The catalog of elementary particles in the Standard Model [1]. 5
1.2 The "Mexican hat" shape of the Brout-Englert-Higgs potential $V(\phi)$ as a function of the complex scalar field ϕ. 8
1.3 The RG evolution of the SM Higgs self-coupling as a function of the energy scale [14]. 10
2.1 Example Feynman diagrams for the loop corrections of the Higgs boson mass in SUSY theory. The left diagrams shows the SM fermion loop, while the addition of the scalar superpartner correction is shown on the right diagram. 11
2.2 Illustration of the particle content in the MSSM as an extension of the Standard Model [21]. 12
3.1 The schematic layout of the CERN accelerator complex [34]. 18
3.2 Cumulative luminosity versus day delivered to CMS during stable beams for pp collisions at nominal center-of-mass energy during the Run 2 data-taking period [35]. 19
3.3 Distributions of average number of interactions per crossing (pileup) for pp collisions in each year during the Run 2 data-taking period [35]. 20
4.1 Cutway diagrams of the CMS detector [39]. 21
4.2 A cross-sectional-slice view of the CMS detector illustrating the interactions of various particle types with different detector components [40]. 22
4.3 The magnetic field values (left) and the magnetic field lines (right) produced by the superconducting solenoid magnet of the CMS detector [41]. 23
4.4 Diagram of the CMS inner tracker in one r-z quadrant, with the collision point at the origin. Green lines depict the pixel detector, while the single-sided and double-sided strip trackers are shown in red and blue, respectively [45]. 24
4.5 Spatial resolution of the pixel along the $r-\phi$ direction as a function of the angle between the track direction and the normal to the sensor plane [38]. 24

4.6 Transverse momentum resolution as a function of p_T (left) and η (right) for single, isolated muons in the barrel, transition, and endcap regions of the tracker [47]. 25
4.7 Layout of the CMS electromagnetic calorimeter in one r-z quadrant [38]. 26
4.8 ECAL energy resolution as a function of electron energy measured in a 3×3 crystal cluster [38]. 27
4.9 Layout of the CMS hadron calorimeter in one r-z quadrant before (top) and after (bottom) the SiPM upgrade. "FEE" indicates the Front End Electronics' locations. Light from layers depicted with the same color are optically added together before reaching the photosensors. The superior photon detection efficiency and response of SiPMs allow for an increased longitudinal granularity with up to 7 depths in HE and 4 depths in HB after the upgrade [49]. 28
4.10 The transverse energy resolution as a function of the simulated jet transverse energy for barrel jets ($|\eta| < 1.4$), endcap jets ($1.4 < |\eta| < 3.0$), and forward jets ($3.0 < |\eta| < 5.0$) [38]. 29
4.11 Layout of the CMS muon system in one r-z quadrant [38]. 30
4.12 The spatial resolution for DT hits in ϕ superlayers (squares) and θ superlayers (diamonds) [52]. 31
4.13 Diagram of the L1 trigger system during Run 2 [53]. 33
4.14 The time-multiplexed architecture of the upgraded calorimeter trigger in L1 trigger [53]. 34
4.15 Global HTCondor pool size in number of CPU cores averaged daily during Run 2 in CMS [62]. 37
5.1 Feynman diagrams for ggHH processes. Top: Contributions from Standard Model processes at leading order, referred to as box and triangle diagrams, respectively. Bottom: BSM processes that describe contact interactions of two Higgs bosons with two top quark (left), between the Higgs boson and gluons (middle and right). 41
5.2 Feynman diagrams that contribute to the production of SM Higgs boson pairs via VBF at LO. The left diagram involves a HHH vertex (λ_{HHH}) and a HVV vertex (c_V), the middle diagram involves two HVV vertices (c_V), and the right diagram involves a HHVV vertex (c_{2V}). 41

5.3 The invariant mass distributions of the reconstructed Higgs boson candidates $m_{\gamma\gamma}$ (left) and m_{jj} (right) in data and simulated events. Data, dominated by the $\gamma\gamma$ and γ+jets backgrounds, are compared to the SM ggF HH signal samples and single H samples ($t\bar{t}H$, ggH, VBF H , VH) after imposing the selection criteria described in Sec. 5.2. The error bars on the data points indicate statistical uncertainties. The HH signal has been scaled by a factor of 10^3 for display purposes. 47

5.4 Distributions of $\widetilde{M}_X$. The SM ggF HH signal is compared with several BSM hypotheses listed in Table 5.1 (left), and the SM VBF HH signal is compared with two different anomalous values of c_{2V} (right). All distributions are normalized to unity. 47

5.5 Angular variables used in the training, from left to right: $\Delta R(\gamma, \text{jet})$, $\cos(\theta_{CS})$, and $\cos(\theta_{b\bar{b}})$. These variables are used for the training of the $t\bar{t}H$ discriminant. 49

5.6 Major variables used in the training to reject events with a leptonic-decay W boson, from left to right, top to bottom: p_T^{miss}, $\Delta\phi(p_T^{miss}, \text{jet}_1)$, $\Delta\phi(p_T^{miss}, \text{jet}_2)$, and the transverse momentum of the leading and subleading electrons and muons. In signal events, there is no subleading electron, which explains its absence in the subleading electron's p_T distribution plot. 49

5.7 χ^2_t variables in training to reject events with a hadronic-decay W boson, for events with at least 2 additional jets (left) and 4 additional jets (right) besides the two b jets. 50

5.8 The multimodal network architecture for the $t\bar{t}H$ tagger. $\text{Object}_{1..N}$ contain the kinematic and identity information of the reconstructed PF objects, ordered by their transverse momenta. The masking layer filters out the zero-padded object, *i.e.*, if an object does not exist in a given event, that object's information does not enter the network. The output of the recurrent neural network is merged with the high-level information, such as p_T^{miss}, $\Delta\phi(p_T^{miss}, \text{jet}_1)$, *etc.*, and then goes through a fully connected block to compute the output prediction. 51

5.9 The performance of the $t\bar{t}H$ tagger. Left: the output score on $HH \to b\bar{b}\gamma\gamma$ signal and $t\bar{t}H$ background. Right: Performance in terms of signal efficiency and background contamination in a receiver operating characteristic (ROC) curve. 52

5.10 Comparison for $t\bar{t}H$ tagger score distributions between data and MC simulation normalized to cross section times total luminosity over 2016 and 2017. 53
5.11 $t\bar{t}H$ score cut optimization with Monte Carlo simulation. The vertical axis indicates the percentage improvement on the 95% CL upper limit on the product of the HH cross section and its branching ratio to $b\bar{b}\gamma\gamma$. The horizontal axis corresponds to different thresholds for the $t\bar{t}H$ tagger score. 54
5.12 The distributions of the $t\bar{t}H$ tagger score (left) and MVA output for nonresonant background in the ggF HH signal region (right) in data and simulated events. Data, dominated by $\gamma\gamma$ + jets and γ + jets backgrounds, are compared to the SM ggF HH signal samples and single H samples ($t\bar{t}H$, ggH, VBF H, VH) after imposing the selection criteria defined in Sec. 5.2. The error bars on the data points indicate statistical uncertainties. The HH signal has been scaled by a factor of 10^3 for display purposes. 55
5.13 The distribution of the two MVA outputs is shown in data and simulated events in the two VBF $\widetilde{M}_X$ regions: $\widetilde{M}_X > 500$ GeV (left) and $\widetilde{M}_X < 500$ GeV (right). Data, dominated by the $\gamma\gamma$ + jets and γ + jets backgrounds, are compared to the VBF HH signal samples with SM couplings and $c_{2V} = 0$, SM ggF HH and single H samples ($t\bar{t}H$, ggH, VBF H, VH) after imposing the VBF selection criteria described in Sec. 5.2. The error bars on the data points indicate statistical uncertainties. The HH signal has been scaled by a factor of 10^3 for display purposes. 57
5.14 Distributions of input variables for non-resonant background in simulated SM HH sample and data for full Run II. 59
5.15 Distributions of input variables for non-resonant background in simulated SM HH sample and data for full Run II. 60
5.16 Distributions of input variables for non-resonant background in simulated SM HH sample and data for full Run II. 61
5.17 Distributions of outputs of MVA and $t\bar{t}H$ tagger for non-resonant background in simulated SM HH sample and data for full Run II. . . 61

5.18 Parametrized signal shape for $m_{\gamma\gamma}$ (left) and m_{jj} (right) in the best resolution ggF (upper) and VBF (lower) categories. The open squares represent simulated events and the blue lines are the corresponding models. Also shown are the σ_{eff} value (half the width of the narrowest interval containing 68.3% of the invariant mass distribution) and the corresponding interval as a gray band, and the full width at half the maximum (FWHM) and the corresponding interval as a double arrow. 64

5.19 Parametrized background shape for m_{jj} distributions for ggH (top left), VBF H (top right), VH (bottom left), and $t\bar{t}H$ (bottom right) in one of the MVA categories. The open squares represent simulated events and the blue lines are the corresponding models. Also shown are the σ_{eff} value (half the width of the narrowest interval containing 68.3% of the invariant mass distribution) and the corresponding interval as a gray band, and the full width at half the maximum (FWHM) and the corresponding interval as a double arrow. 66

5.20 Invariant mass distribution $m_{\gamma\gamma}$ (upper) and m_{jj} (lower) for the selected events in data (black points) in the best resolution ggF (CAT 0) and VBF (CAT 0) categories. The solid red line shows the sum of the fitted signal and background (HH+H+B), the solid blue line shows the background component from the single Higgs boson and the nonresonant process (H+B), and the dashed black line shows the nonresonant background component (B). The normalization of each component (HH, H, B) is extracted from the combined fit to the data in all analysis categories. The one (green) and two (yellow) standard deviation bands include the uncertainties in the background component of the fit. The lower panel in each plot shows the residual signal yield after the background (H+B) subtraction. 71

5.21 Invariant mass distribution $m_{\gamma\gamma}$ (left) and m_{jj} (right) for the selected events in data (black points) weighted by S/(S+B), where S (B) is the number of signal (background) events extracted from the signal-plus-background fit. The solid red line shows the sum of the fitted signal and background (HH+H+B), the solid blue line shows the background component from the single Higgs boson and the nonresonant process (H+B), and the dashed black line shows the nonresonant background component (B). The normalization of each component (HH, H, B) is extracted from the combined fit to the data in all analysis categories. The one (green) and two (yellow) standard deviation bands include the uncertainties in the background component of the fit. The lower panel in each plot shows the residual signal yield after the background (H+B) subtraction. 72

5.22 Expected and observed 95% CL upper limits on the product of the HH production cross section and $\mathcal{B}(HH \rightarrow b\bar{b}\gamma\gamma)$ obtained for different values of κ_λ assuming $\kappa_t = 1$. The green and yellow bands represent, respectively, the one and two standard deviation extensions beyond the expected limit. The long-dashed red line shows the theoretical prediction. 73

5.23 Negative log-likelihood, as a function of κ_λ, evaluated with an Asimov data set assuming the SM hypothesis (left) and the observed data (right). The 68 and 95% CL intervals are shown with the dashed gray lines. The two curves are shown for the HH (blue) and HH + $t\bar{t}H$ (orange) analysis categories. All other couplings are set to their SM values. 73

5.24 Negative log-likelihood contours at 68 and 95% CL in the (κ_λ, κ_t) plane evaluated with an Asimov data set assuming the SM hypothesis (left) and the observed data (right). The contours obtained using the HH analysis categories only are shown in blue, and in orange when combined with the $t\bar{t}H$ categories. The best fit value for the HH categories only ($\kappa_\lambda = 0.6$, $\kappa_t = 1.2$) is indicated by a blue circle, for the HH + $t\bar{t}H$ categories ($\kappa_\lambda = 1.4$, $\kappa_t = 1.3$) by an orange diamond, and the SM prediction ($\kappa_\lambda = 1.0$, $\kappa_t = 1.0$) by a black star. The regions of the 2D scan whether the κ_t parametrization for anomalous values of κ_λ at LO is not reliable are shown with a gray band. 74

5.25 Negative log-likelihood scan, as a function of κ_t, evaluated with an Asimov data set assuming the SM hypothesis (left) and the observed data (right). The 68 and 95% CL intervals are shown with the dashed gray lines. The two curves are shown for the HH (blue) and the HH + $t\bar{t}H$ (orange) analysis categories. All other couplings are fixed to their SM values. 74

5.26 Expected and observed 95% CL upper limits on the product of the VBF HH production cross section and $\mathcal{B}(HH \rightarrow b\bar{b}\gamma\gamma)$ obtained for different values of c_{2V}. The green and yellow bands represent, respectively, the one and two standard deviation extensions beyond the expected limit. The long-dashed red line shows the theoretical prediction. 75

5.27 Negative log-likelihood contours at 68 and 95% CL in the (κ_λ, c_{2V}) plane evaluated with an Asimov data set assuming the SM hypothesis (left) and with the observed data (right). The contours are obtained using the HH analysis categories only. The best fit value (κ_λ = 0.0, c_{2V} = 0.3) is indicated by a blue circle, and the SM prediction (κ_λ = 1.0, c_{2V} = 1.0) by a black star. 75

5.28 Expected and observed 95% CL upper limits on the product of the ggF HH production cross section and $\mathcal{B}(HH \rightarrow b\bar{b}\gamma\gamma)$ obtained for different nonresonant benchmark models (defined in Table 5.1) (upper) and BSM coupling c_2 (lower). In this fit, the yield of the VBF HH signal is constrained within uncertainties to the one predicted in the SM. The green and yellow bands represent, respectively, the one and two standard deviation extensions beyond the expected limit. On the lower plot, the long-dashed red line shows the theoretical prediction. 78

6.1 Pictorial representation of ordinary quark and gluon jets (top left), b jets (top center), and boosted-jet topologies, emerging from high-p_T W and Z bosons (top right), Higgs bosons (bottom left), and top quarks (bottom right) decaying to all-quark final states. 80

6.2 Distributions of the 16 high-level features used in this study, described in Ref. [181]. 84

6.3 Average 100×100 images for the five jet classes considered in this study: q (top left), g (top center), W (top right), Z (bottom left), and top jets (bottom right). The temperature map represents the amount of p_{T} collected in each cell of the image, measured in GeV and computed from the scalar sum of the p_{T} of the particles pointing to each cell. 85
6.4 Example of 100×100 images for the five jet classes considered in this study: q (top-left), g (top-right), W (center-left), Z (center-right), and top jets (bottom). The temperature map represents the amount of p_{T} collected in each cell of the image, measured in GeV and computed from the scalar sum of the p_{T} of the particles pointing to each cell. . . 85
6.5 Distributions of kinematic features described in the text for the 150 highest-p_{T} particles in each jet. 87
6.6 An example graph with three fully connected vertices and the corresponding six edges. 95
6.7 A flowchart illustrating the interaction network scheme. 96
6.8 ROC curves for JEDI-net and the three alternative models, computed for gluons (top-left), light quarks (top-right), W (center-left) and Z (center-right) bosons, and top quarks (bottom). The solid lines represent the average ROC curves derived from 10 k-fold trainings of each model. The shaded bands around the average lines are represent one standard deviation, computed with the same 10 k-fold trainings. . 100
6.9 Two-dimensional distributions between (top to bottom) $\overline{O}_1$ and constituents multiplicity, $\overline{O}_4$ and $\tau_1^{(\beta=2)}$, $\overline{O}_2$ and $\tau_3^{(\beta=1)}$, $\overline{O}_9$ and $\tau_3^{(\beta=2)}$, for jets originating from (right to left) gluons, light flavor quarks, W bosons, Z bosons, and top quarks. For each distribution, the linear correlation coefficient ρ is reported. 102
6.10 Two example graphs with 3 particles and 2 vertices and the corresponding edges. 105

6.11 Illustration of the modified JEDI-net for the boosted $H \to b\bar{b}$ classifier. The particle feature matrix X is multiplied by the receiving and sending matrices R_R and R_S to build the particle-particle interaction feature matrix B_{pp}. Similarly, the particle feature matrix X and the vertex feature matrix Y are multiplied by the adjacency matrices R_K and R_V, respectively, to build the particle-vertex interaction feature matrix B_{vp}. These pairs are then processed by the interaction functions f_R^{pp} and f_R^{vp}, and the post-interaction function f_O, which are expressed as neural networks and learned in the training process. This procedure creates a learned representation of each particle's post-interaction features, given by N_p vectors of size D_O. The N_p vectors are summed, giving D_O features for the entire jet, which is given as input to a classifier ϕ_C, also represented by a neural network. More details on the various steps are given in the text. 107

6.12 Performance of the IN, all-particle IN, DDB, and DDB+ algorithms quantified with a ROC curve of FPR (QCD mistagging rate) versus TPR ($H \to b\bar{b}$ tagging efficiency). The performance of each baseline algorithm is compared to that of the algorithms after applying the DDT procedure to decorrelate the tagger score from the jet mass. This decorrelation results in a smaller TPR for a given FPR. 112

6.13 An illustration of the "sculpting" of the background jet mass distribution (left) and the signal jet mass distribution (right) after applying a threshold on the tagger score corresponding to a 1% FPR for several different algorithms. The unmodified interaction network is highly correlated with the jet mass, but after applying the methods described in the text, the correlation is reduced for the background while the peak of the signal distribution is still retained. 113

6.14 The mass decorrelation metric $1/D_{\text{JS}}$ as a function of background rejection for the baseline and decorrelated IN, DDB, and DDB+ taggers. The decorrelation is quantified as the inverse of the JS divergence between the background mass distribution passing and failing a given threshold cut on the classifier score. Greater values of this metric correspond to better mass decorrelation. The background rejection is quantified as the inverse of the FPR, while the signal efficiency is equal to the TPR. 115

6.15 TPR of the baseline and decorrelated IN, DDB, and DDB+ taggers as a function of the number of reconstructed PVs for a 1% FPR. . . . 116
7.1 Example Feynman diagrams for SUSY processes with diphoton (left) and single photon (middle and right) final states via pair production of squark (upper) and gluino (lower) from pp collisions at the LHC. . 119
7.2 The 95% CL exclusion contours for the GMSB SPS8 neutralino production cross section set by the previous CMS search on 2016 and 2017 data, along with the ATLAS and CMS results in Run 1 [235]. . 120
7.3 The displacements of GMSB signal photons in the longitudinal (left) and transverse (right) directions from a representative point Λ = 400 TeV and $c\tau$ = 200 cm. In a typical signal region (γ_1 time > 1.5 ns and $p_{\mathrm{T}}^{\mathrm{miss}}$ > 150 GeV), the GMSB signal photons are enriched in the phase space where the displacements are greater than 20 cm in both longitudinal and transverse directions. 126
7.4 The signal and background MC distributions of the 7 input variables to the deep neural networks for the leading photon for 2016. GMSB signal histograms are scaled up to have the same bin integrals as the sum of background histograms. 127
7.5 The signal and background MC distributions of the 7 input variables to the deep neural networks for the leading photon for 2017. GMSB signal histograms are scaled up to have the same bin integrals as the sum of background histograms. 128
7.6 The signal and background MC distributions of the 7 input variables to the deep neural networks for the second photon in the ECAL barrel region for 2017. GMSB signal histograms are scaled up to have the same bin integrals as the sum of background histograms. 129
7.7 The signal and background MC distributions of the 7 input variables to the deep neural networks for the second photon in the ECAL endcap region for 2017. GMSB signal histograms are scaled up to have the same bin integrals as the sum of background histograms. 130
7.8 The receiver operating characteristic (ROC) curve of the leading photon's DNN identifiers for 2016 (left) and 2017+2018 (right). Signal consists of all GMSB signal samples with some removals based on sample points and displacements as described in Sec. 7.3. The performances of the cut-based identifiers used in the previous CMS search are denoted with the blue dots. 131

7.9 The distributions of the leading photon's DNN scores on MC signal and background samples for 2016 (left) and 2017+2018 (right). Signal consists of all GMSB signal samples with some removals based on sample points and displacements as described in Sec. 7.3. 131
7.10 The receiver operating characteristic (ROC) curve of the subleading photon's DNN identifiers for 2017 and 2018 in the ECAL barrel (left) and endcap (right). Signal consists of all GMSB signal samples with some removals based on sample points and displacements as described in Sec. 7.3. The performances of the cut-based identifiers used in the previous CMS search are denoted with the green dots. . . 132
7.11 The distributions of DNN scores for subleading photons in the ECAL barrel (left) and endcap (right) on MC signal and background samples for 2017 and 2018. Signal consists of all GMSB signal samples with some removals based on sample points and displacements as described in Sec. 7.3. 132
7.12 Importance ranking of input variables to the DNN for the leading photon. Color indicates the feature values with respect to its mean. SHAP values on the left of the vertical bar indicate events likely to be from background processes, while SHAP values on the right of the vertical bar indicate events likley to be from signal processes. . . 133
7.13 Example representation of an ECAL pulse shape. (a) ECAL pulse shape as a function of the difference between the time T of each read-out sample along the pulse and the time T_{max} when the pulse reaches its maximum amplitude A_{max}. The red dots represent the amplitudes of ten discrete readout samples from a single pulse, normalized to the maximum amplitude. The solid line represents the average pulse shape, which is universal across all crystals to the first order. (b) An alternative pulse shape representation using the time difference from T_{max} as a function of the ratio of two consecutive readout samples' amplitudes [41]. 134
7.14 Local time resolution of ECAL versus the effective amplitude of the neighboring crystals from the same read-out electronics measured in data and simulation for 2016 (upper left), 2017 (upper right), 2018 eras ABC (lower left), and 2018 era D (lower right). The data in 2018 era D is centrally processed differently by CMS, therefore it requires a separate analysis from 2018 data in previous eras. 136

7.15 Timing correction for simulation using $Z \rightarrow ee$ events and data for the 2017 data-taking period. On the left is the correction of the mean electron cluster's arrival times in different electron's energy bins. On the right is the correction of the standard deviations of distributions of electron cluster's arrival time in different electron energy bins. The purple lines overlapping with the blue lines indicate that after correction, the mean and standard deviation of electron arrival time distributions from simulation match those from data. 138
7.16 Examples of electron arrival times before (top) and after (bottom) the correction procedure in different energy bins. 138
7.17 Illustration of the four bins A, B, C, and D in the two-dimensional distribution of $p_{\mathrm{T}}^{\mathrm{miss}}$ and t_{γ} in the ABCD method. 144
7.18 Example shape templates to extract $r_{\mathrm{B/A}}$ and $r_{\mathrm{D/A}}$ to compute the expected background yields in bin B, C, and D from bin A in the modified ABCD method to obtain the expected upper limit on the signal strength. The vertical lines represent the bin boundaries. $r_{\mathrm{B/A}}$ is computed as the ratio between number of events on the right of the bin boundary and number of events on the left of the bin boundary from the $p_{\mathrm{T}}^{\mathrm{miss}}$ plot; $r_{\mathrm{D/A}}$ is computed as the ratio between number of events on the right of the bin boundary and number of events on the left of the bin boundary on the t_{γ} plot. 145
7.19 Distributions of t_{γ} in different $p_{\mathrm{T}}^{\mathrm{miss}}$ bins from 2016 data in the control region where the DNN photon ID requirement on the leading photon is inverted. All distributions are normalized to unity. 147
7.20 The expected 95% CL upper limit on the product of GMSB SPS8 neutralino production cross section and its branching ratio as a function of the neutralino mass for $c\tau$ between 10 cm and 200 cm, obtained from the full Run 2 data. 162
7.21 The expected 95% CL exclusion limit on the product of GMSB SPS8 neutralino production cross section and its branching ratio as a function of the neutralino mass for $c\tau$ above 200 cm, obtained from the full Run 2 data. 163

7.22 The expected 95% CL exclusion contours for the GMSB SPS8 model as functions of the SUSY breaking scale Λ and the neutralino's lifetime $c\tau$ from different years along with the combined Run 2 result. Top: The exclusion boundaries from this search compared to previously published analyses from ATLAS and CMS. Bottom: The exclusion boundaries along with a color map of the expected 95% CL upper limit on the signal cross section obtained from full Run 2 data. 164
8.1 Distribution of the HLF quantities for the four considered SM processes. 173
8.2 Distribution of the HLF quantities for the four considered BSM benchmark models. 174
8.3 Schematic representation of the VAE architecture presented in the text. The size of each layer is indicated by the value within brackets. The blue rectangle X represents the input layer, which is connected to a stack of two consecutive fully connected layers (black boxes). The last of the two black box is connected to two layers with four nodes each (red boxes), representing the μ_z and σ_z parameters of the encoder pdf $p(z|x)$. The green oval represents the sampling operator, which returns a set of values for the 4-dimensional latent variables z. These values are fed into the decoder, consisting of two consecutive hidden layers of 50 nodes each (black boxes). The last of the decoder hidden layer is connected to the three output layers, whose nodes correspond to the parameters of the predicted distribution in the initial 21-dimension space. The pink ovals represent the computation of the two parts of the loss function: the KL loss and the reconstruction loss (see text). The computation of the KL requires 8 additional learnable parameters (μ_p and σ_p, represented by the orange boxes on the top-left part of the figure), corresponding to the means and RMS of the four-dimensional Gaussian prior $p(z)$. The total loss in computed as described by the formula in the bottom-left black box (see Eq. (8.6)). 176
8.4 Training history for VAE. Total loss, reconstruction negative-log-likelihood ($\text{Loss}_{\text{reoc}}$) and KL divergence ($D_{KL}$) are shown separately for training and validation set though all the training epochs. 179

8.5 Comparison of input (blue) and output (red) probability distributions for the HLF quantities in the validation sample. The input distributions are normalized to unity. The output distributions are obtained summing over the predicted pdf of each event, normalized to the inverse of the total number of events (so that the total sum is normalized to unity). 181
8.6 ROC curves for the fully-supervised BDT classifiers, optimized to separate each of the four BSM benchmark models from the SM cocktail dataset. 182
8.7 Distribution of the VAE's loss components, $\text{Loss}_{\text{reco}}$ (left) and D_{KL} (right), for the validation dataset. For comparison, the corresponding distribution for the four benchmark BSM models are shown. The vertical line represents a lower threshold such that $5.4 \cdot 10^{-6}$ of the SM events would be retained, equivalent to $\sim$ 1000 expected SM events per month. 183
8.8 Comparison between the input distribution for the 21 HLF of the validation dataset (blue histograms) and the distribution of the SM outlier events selected from the same sample by applying the $\text{Loss}_{\text{reco}}$ threshold (red dots). The outlier events cover a large portion of the HLF definition range and do not cluster on the tails. 185
8.9 Comparison between the distribution of the 21 HLF distribution for $A \rightarrow 4\ell$ full dataset (blue) and $A \rightarrow 4\ell$ events selected by applying the $\text{Loss}_{\text{reco}}$ threshold (red). The selected events are not trivially sampled from the tail. 186
8.10 Performance of the VAE for different BSM scenarios. Left: ROC curves for the VAE trained only on SM events (solid), compared to the corresponding curves for the four supervised BDT models (dashed) described in Section 8.4. Right: Normalized p-value distribution distribution for the SM cocktail events and the four BSM benchmark processes. 187
8.11 ROC curves for the VAE trained on SM contaminated with and without $A \rightarrow 4\mu$ contamination. Different levels of contamination are reported corresponding to 0.02% (σ = 7.15 pb - equal to the estimated one to have 100 events per month), 0.19% (σ = 71.5 pb) and 1.89% (σ = 715 pb) of the training sample. 191

8.12 Comparison of Deep Network architectures: (a) In a GAN, a generator G returns samples $G(z)$ from latent-space points z, while a discriminator D_x tries to distinguish the generated samples $G(z)$ from the real samples x. (b) In an autoencoder, the encoder E compresses the input x to a latent-space point z, while the decoder D provides an estimate $D(z) = D(E(x))$ of x. (c) A BiGAN is built by adding to a GAN an encoder to learn the z representation of the true x, and using the information both in the real space $\mathcal{X}$ and the latent space $\mathcal{Z}$ as input to the discriminator. (d) The ALAD model is a BiGAN in which two additional discriminators help converging to a solution which fulfils the cycle-consistency conditions $G(E(x)) \approx x$ and $E(G(z)) \approx z$. The $\oplus$ symbol in the figure represents a vector concatenation. 192

8.13 ROC curves for the ALAD trained on the SM cocktail training set and applied to SM+BSM validation samples. The VAE curve corresponds to the best result of the VAE, which is shown here for comparison. The other four lines correspond to the different anomaly score models of the ALAD. 197

8.14 Distribution for the A_{L_1} anomaly score. The "SM cocktail" histogram corresponds to the anomaly score for the validation sample. The other four distributions refer to the scores of the four BSM benchmark models. 198

8.15 Left: ROC curves for each BSM process obtained with the ALAD L_1-score model. Right: LR_+ curves corresponding to the ROC curves on the left. 198

9.1 The baseline plan of the LHC for the next decade and beyond. After LS3 with the installation of the HL-LHC and the high luminosity upgrade for CMS, the machine will start collect data near the end of 2027 with the target peak luminosity of 5×10^{34} $\mathrm{cm}^{-2}\mathrm{s}^{-1}$ and an integrated luminosity of 250 fb^{-1} per year, with the goal of 3000 fb^{-1} by 2040 [293]. 202

9.2 CMS experiment's projection for the CPU time (top) and disk space (bottom) requirements needed annually for CMS processing and analyses versus resource availability under 2 scenarios: (1) a running scenario of 275 fb^{-1} per year during Run 4, with a trigger rate of 7.5 kHz, as shown by the solid blue line; (2) a running scenario of 500 fb^{-1} per year during Run 4, with a trigger rate of 10 kHz, as shown by dashed blue line. The projected resources needed are summed across Tier-0, Tier-1, and Tier-2 resources. The black curves show the projected resources availability assuming an annual increase between 10% and 20% [296]. 204

10.1 Relative composition of the isolated-lepton sample after the acceptance requirement (left) and the trigger selection (right), as described in the text. 209

10.2 An example of a $t\bar{t}$ event as the input of the raw-image classifier. Vertical and horizontal axess are the ϕ and η coordinates, respectively, of the sub-detectors. 214

10.3 Example of a $t\bar{t}$ event, represented as a 5-channel abstract images of photons (top-left), charged hadrons (top-center), neutral hadrons (top-right), the isolated lepton (bottom-kleft), and the event E_T^{miss} (botton-right). 214

10.4 Network architecture of the inclusive classifier. 217

10.5 ROC curves for the $t\bar{t}$ (left) and W (right) selectors described in the chapter. 218

10.6 Pearson correlation coefficients between the $y_{t\bar{t}}$ (left) and y_W (right) scores of the Particle-sequence classifier and the 14 quantities of the HLF dataset. 219

10.7 Selection efficiency using 99% TPR working point as functions of lepton p_T, M_T^2, and E_T^{miss} for the $t\bar{t}$ selector on $t\bar{t}$ events (top) and the W selector on W events (bottom). 220

10.8 Dependence of TPR and FPR on the amount of pileup in the event (estimated through the number of vertices) for the inclusive $t\bar{t}$ selector when applying the 99% TPR working-point threshold. The gray histogram shows the distribution of the number of vertices in the training dataset, covering a wide range from $\sim$ 10 to $\sim$ 40 following a Poisson distribution with mean value of 20. 221

10.9 Selection efficiencies of different BSM models using 99% TPR working point as functions of lepton p_T, M_T^2, and E_T^{miss}. From top to bottom, $A \to H^+W^-$, High-mass $A \to H^+W^-$, $A \to 4\ell$, W', and Z'. . 222
10.10 Distributions of the validation sample and pseudo-data. The pseudo-data is created by adding a Gaussian noise of mean zero and standard deviation of 10% to the validation sample's particle momenta. The high-level features are then recomputed with the new list of particles. 223
10.11 ROC curves for the $t\bar{t}$ (left) and W (right) selectors described in the chapter, trained on a dataset defined by a tighter baseline selection. . 225
11.1 The architecture of conditional pGAN: generator $\mathcal{G}^{\text{cond}}$ (top) and discriminator $\mathcal{D}^{\text{cond}}$ (bottom). Arrows signify concatenation. Details are described in the text. 232
11.2 The architecture of unconditional pGAN: generator $\mathcal{G}^{\text{cond}}$ (top) and discriminator $\mathcal{D}^{\text{cond}}$ (bottom). Arrows signify concatenation. Details are described in the text. 233
11.3 Comparison of the transverse momentum p_T (left), azimuth angle ϕ (center) and pseudorapidity η (right) for charged particles between the test data and the events generated by unconditional pGAN (top) and conditional pGAN (bottom). For the conditional pGAN, ϕ is transformed to be the azimuth angle between the particles' momenta and $\vec{p}_T^{\text{miss}}$. 236
11.4 Comparison of the transverse thrust and $p_{\text{T}}^{\text{miss}}$ distributions between the test data and the generated events by pGANs. 237
11.5 Evolution of our performance metric (solid black) as a function of training. EM distances for some of the individual quantities are superposed. 237
11.6 Comparison between leading jet p_T distributions for events with no pileup (solid black) and pileup generated by `Pythia8` (green) and by the network (magenta). Distributions are shown both before (left) and after (right) running the *SoftKiller* pileup mitigation algorithm. The bottom plot shows the Pythia/GAN ratio. 239

12.1 The event generation workflow of the CMS experiment. The pp collision process is simulated up to the production of stable (hence observable) particles (GEN). The simulation of the detector response is modelled by the `GEANT4` library (SIM). The resulting energy deposits are turned into digital signals (DIGI) that are then reconstructed by the same software used to process real collision events (RECO). At this stage, high-level objects such as jets are reconstructed. Starting from the RECO data format, a reduced analysis data format (MINIAOD) is derived. 243

12.2 Computing resource breakdown for the generation workflow of the CMS experiment, in terms of CPU (left) and storage disk (right). See Sec. 12.6 for details. 243

12.3 Model architecture: a feature vector at generator level $\vec{x}_G$ is given as input to two regression models, returning vectors of central values ($\vec{\mu}_{DL}$) and RMS ($\vec{\sigma}_{DL}$), from which the reconstructed feature vector predicted by the DL model $\vec{x}_{DL}$ is generated. 249

12.4 Distribution of reconstructed and model-predicted quantities for the feature-vector quantities, compared to the corresponding quantities from generator-level quantities provided as input to the model. The bottom panel below each plot shows the bin-by-bin ratio of the model-predicted over reconstructed distribution for each quantity, labelled DL/Reco. The error bars on the model-predicted quantities is composed of the statistical uncertainty and systematic uncertainty associated with model training, represented by the different colors. 251

12.5 Distribution of reconstructed and model-predicted auxiliary quantities, compared to the corresponding generator-level quantities. The bottom panel below each plot shows the bin-by-bin ratio of the model-predicted over reconstructed distribution for each quantity, labelled DL/Reco. The error bars on the model-predicted quantities is composed of the statistical uncertainty and systematic uncertainty associated with model training, represented by the different colors. 252

12.6 Relative residual distribution for reconstructed and model-predicted quantities in the feature vector, computing with respect to the reference input. The bottom panel of each plot shows the ratio between the two relative residuals, expected to be consistent with 1 for a DL model which correctly models the detector response of the traditional workflow. The error bars on the model-predicted quantities is composed of the statistical uncertainty and systematic uncertainty associated with model training, represented by the different colors. . 254
12.7 Relative residual distribution for reconstructed and model-predicted auxiliary quantities, computing with respect to the reference input. The bottom panel of each plot shows the ratio between the two relative residuals, expected to be consistent with 1 for a DL model which correctly models the detector response of the traditional workflow. The error bars on the model-predicted quantities is composed of the statistical uncertainty and systematic uncertainty associated with model training, represented by the different colors. 255
12.8 Distribution of reconstructed and model-predicted quantities for the feature-vector quantities, compared to the corresponding quantities from generator-level input. In this case, the model is applied to a dataset five times larger than the training dataset. The error bars on the model-predicted quantities is composed of the statistical uncertainty and systematic uncertainty associated with model training, represented by the different colors. 256
12.9 Distribution of reconstructed and model-predicted quantities in the auxiliary quantities, compared to the corresponding quantities from generator-level input. In this case, the model is applied to a dataset five times larger than the training dataset. The error bars on the model-predicted quantities is composed of the statistical uncertainty and systematic uncertainty associated with model training, represented by the different colors. 257
12.10 Differential double ratio distribution (high-statistics over low-statistics) for the reco-to-DL ratios shown in Figs. 12.4 and 12.8 and in Fig. 12.5 and Fig. 12.9. 258
12.11 Distribution of input features predicted by the model as a function of the corresponding quantities from detector simulation. 260

12.12 Distribution of auxiliary features predicted by the model as a function of the corresponding quantities from detector simulation. 261
12.13 Predict on an inference dataset five times larger than the training dataset. Relative residual distribution for reconstructed and model-predicted quantities in the feature vector, computing with respect to the reference input. 262
12.14 Predict on an inference dataset five times larger than the training dataset. Relative residual distribution for reconstructed and model-predicted auxiliary quantities, computing with respect to the reference input. 263

LIST OF TABLES

Number *Page*

4.1 The transverse spatial resolution per CSC station [52]. 32
4.2 CMS data formats for physics analyses [73]. 38
5.1 Parameter values of the 12 BSM benchmarks along with the Standard Model point. 43
5.2 Photon selection criteria on photon candidates. 45
5.3 Summary of the baseline selection criteria for HH $\rightarrow b\bar{b}\gamma\gamma$ events. . 46
5.4 Expected number of SM HH signal events after the baseline preselection for each data-taking year (scaled according to luminosity) and full Run II. 46
5.5 Performance of the $t\bar{t}H$ tagger. The uncertainty in background efficiency is obtained from k-fold cross-validation. 52
5.6 Summary of the analysis categories. Two VBF- and twelve ggF-enriched categories are defined based on the output of the MVA classifiers and the mass of the Higgs boson pair system $\widetilde{M}_X$. The VBF and ggF categories are mutually exclusive. 62
6.1 Charged particle features. The IN and DDB+ models use all of the features, while DDB algorithm uses the subset of features indicated in bold. 89
6.2 Secondary vertex features. The IN and DDB+ models use all of the features, while the DDB algorithm uses the subset of features indicated in bold. 91
6.3 High-level features used by the DDB algorithm. 91
6.4 Additional features for charged or neutral particles. The all-particle IN model uses these features. 94
6.5 Optimal JEDI-net hyperparameter setting for different input data sets, when the summed $\overline{O}_i$ quantities are given as input to the ϕ_C network. The best result, obtained when considering up to 150 particles per jet, is highlighted in bold. 98
6.6 Optimal JEDI-net hyperparameter setting for different input data sets, when all the O_{ij} elements are given as input to the ϕ_C network. The best result, obtained when considering up to 100 particles per jet, is highlighted in bold. 99

6.7 True positive rates (TPR) for the optimized JEDI-net taggers and the three alternative models (DNN, CNN, and GRU), corresponding to a false positive rate (FPR) of 10% (top) and 1% (bottom). The largest TPR value for each case is highlighted in bold. 101
6.8 Resource comparison across models. The quoted number of parameters refers only to the trainable parameters for each model. The inference time is measured by applying the model to batches of 1000 events 100 times: the 50% median quantile is quoted as central value and the 10%-90% semi-distance is quoted as the uncertainty. The GPU used is an NVIDIA GTX 1080 with 8 GB memory, mounted on a commercial desktop with an Intel Xeon CPU, operating at a frequency of 2.60GHz. The tests were executed in PYTHON 3.7 with no other concurrent process running on the machine. 103
6.9 Performance metrics of the different baseline and decorrelated models, including accuracy, area under the ROC curve, background rejection at a true positive rate of 30% an 50%, and true positive rate and mass decorrelation metric $1/D_{\mathrm{JS}}$ at a false positive rate of 1%. For the DDT models, the corresponding accuracy is listed for the tagger after the decorrelation is performed for a FPR of 50%. 114
7.1 List of primary data sets used in this analysis and their respective sizes. `RAW` data are recorded directly from the detector to disk, while `MiniAOD` are reduced data format after physics object reconstruction to be used for most analyses. 121
7.2 List of all generated GMSB SPS8 signal models, parametrized by the SUSY breaking scale Λ, and their corresponding production cross section and masses of the gluinos $\tilde{g}$, neutralinos $\widetilde{\chi}^0$, and gravitino $\tilde{G}$. For each Λ value, 10 different neutralino's lifetime values are generated, corresponding to 10 gravitino's masses $\mathrm{M}_{\tilde{G}}$. 122
7.3 Jet identification criteria recommended by CMS for different years. . 122
7.4 Trigger selection criteria for photons. In 2016, the trigger requires the presence of two photons. In 2017 and 2018, only one photon with p_{T} greater than 60 GeV is required. 123
7.5 Fit results for the ECAL timing resolution parameters from data in different year periods. 137
7.6 Summary of event selection criteria in this search. 139

7.7 Event selection efficiency for GMSB SPS8 $c\tau$ = 10 cm and varying Λ (unit of efficiency: %; unit of Λ : TeV) using the 2016 event selection flow summarized in Table 7.6. 140
7.8 Event selection efficiency for GMSB SPS8 $c\tau$ = 100 cm and varying Λ (unit of efficiency: %; unit of Λ : TeV) using the 2016 event selection flow summarized in Table 7.6. 140
7.9 Event selection efficiency for GMSB SPS8 $c\tau$ = 1000 cm and varying Λ (unit of efficiency: %; unit of Λ : TeV) using the 2016 event selection flow summarized in Table 7.6. 141
7.10 Event selection efficiency for GMSB SPS8 $c\tau$ = 10 000 cm and varying Λ (unit of efficiency: %; unit of Λ : TeV) using the 2016 event selection flow summarized in Table 7.6. 141
7.11 Event selection efficiency for GMSB SPS8 $c\tau$ = 10 cm and varying Λ (unit of efficiency: %; unit of Λ : TeV) using the 2017 event selection flow summarized in Table 7.6. 142
7.12 Event selection efficiency for GMSB SPS8 $c\tau$ = 100 cm and varying Λ (unit of efficiency: %; unit of Λ : TeV) using the 2017 event selection flow summarized in Table 7.6. 142
7.13 Event selection efficiency for GMSB SPS8 $c\tau$ = 1000 cm and varying Λ (unit of efficiency: %; unit of Λ : TeV) using the 2017 event selection flow summarized in Table 7.6. 143
7.14 Event selection efficiency for GMSB SPS8 $c\tau$ = 10 000 cm and varying Λ (unit of efficiency: %; unit of Λ : TeV) using the 2017 event selection flow summarized in Table 7.6. 143
7.15 Pearson's correlation coefficient r between $p_{\mathrm{T}}^{\mathrm{miss}}$ and t_γ in different t_γ regions, computed on the control region from 2016 data, where the DNN photon ID requirement on the leading photon is inverted. . . . 146
7.16 Summary of systematics and their assigned values in this analysis. . . 148
7.17 Predicted signal yields in bins A, B, C, and D using the event selection described in Sec. 7.5 for the year 2016 in different Λ and $c\tau$ values. . 148
7.18 Predicted signal yields in bins A, B, C, and D using the event selection for the single-photon category in 2017. 151
7.19 Predicted signal yields in bins A, B, C, and D using the event selection for the diphoton category in 2017. 153
7.20 Predicted signal yields in bins A, B, C, and D using the event selection for the single-photon category in 2018. 156

7.21 Predicted signal yields in bins A, B, C, and D using the event selection for the diphoton category in 2018. 158
7.22 Predicted background yields in bins B, C, and D from observed data in bin A using the shape templates described in Sec. 7.6 for 2016. . . 160
7.23 Predicted background yields in bins B, C, and D from observed data in bin A using the shape templates described in Sec. 7.6 for 2017. . . 161
7.24 Predicted background yields in bins B, C, and D from observed data in bin A using the shape templates described in Sec. 7.6 for 2018. . . 161
8.1 Acceptance and L1 trigger (i.e. p_T^ℓ and Iso requirement) efficiency for the four studied SM processes and corresponding values for the BSM benchmark models. For SM processes, we quote the total cross section before the trigger, the expected number of events per month and the fraction in the SM cocktail. For BSM models, we compute the production cross section corresponding to an average of 100 BSM events per month passing the acceptance and L1 trigger requirements. The monthly event yield is computed assuming an average luminosity per month of 5 fb^{-1}, corresponding to the running conditions discussed in Section 8.1. 170
8.2 Classification performance of the four BDT classifiers described in the text, each trained on one of the four BSM benchmark models. The two set of values correspond to the area under ROC curve (AUC), and to the true positive rate (TPR) for a SM false positive rate $\epsilon_{SM} = 5.4 \cdot 10^{-6}$, i.e., to $\sim$ 1000 SM events accepted every month. 180
8.3 By-process acceptance rate for the anomaly detection algorithm described in the text, computed applying the threshold on $\text{Loss}_{\text{reco}}$ shown in Fig. 8.7. The threshold is tuned such that a fraction of about $\epsilon_{SM} = 5.4 \cdot 10^{-6}$ of SM events would be accepted, corresponding to $\sim$ 1000 SM events/month, assuming the LHC running conditions listed in Section 8.1. The sample composition refers to the subset of SM events accepted by the anomaly detection algorithm. All quoted uncertainties refer to 95% CL regions. 187

8.4 Breakdown of BSM processes efficiency, and cross section values corresponding to 100 selected events in a month and to a signal-over-background ratio of 1/3 (i.e., an absolute yield of ∼ 400 events/month). The monthly event yield is computed assuming an average luminosity per month of 5 fb^{-1}, computing by taking the LHC 2016 data delivery (∼ 40 fb^{-1} collected in 8 months). All quoted efficiencies are computed fixing the VAE loss threshold $\epsilon_{SM} = 5.4 \cdot 10^{-6}$. 188

8.5 Selection efficiencies for a typical single lepton trigger (SLT) and the proposed VAE selection, shown for the four benchmark BSM models and for the SM cocktail. The last row quotes the corresponding BSM-to-SM ratio of signal-over-background ratios (SBRs), quantifying the purity of the selected sample. 189

8.6 Hyperparameters for the ALAD algorithm. Parameters in bold have been optimized for. No Dropout layer is applied wherever a dropout rate is not specified. 196

10.1 False positive rate (FPR) and trigger rate (TR) at different values of the true positive rate (TPR), for a $t\bar{t}$ (top) and W selector. Rate values are estimated scaling the TPR and process-dependent FPR values by the acceptance and efficiency, assuming a leading-order (LO) production cross section and luminosity of $2{\times}10^{34}$ cm^{-2} s^{-1}. TR values should be taken only as suggestions of the actual rates, since the accuracy is limited by the use of LO cross sections and a parametric detector simulation. 219

10.2 Signal efficiency (TPR) at different values of the false positive rate (FPR) for the *inclusive classifier* selecting $t\bar{t}$ evaluated on the validation sample and the pseudo-data. 224

10.3 False positive rate (FPR) and trigger rate (TR) corresponding to different values of the true positive rate (TPR), for a $t\bar{t}$ (top) and W selector. Rate values are estimated scaling the TPR and process-dependent FPR values by the acceptance and efficiency, assuming a leading-order (LO) production cross section and luminosity of $2{\times}10^{34}$ cm^{-2} s^{-1}. TR values should be taken only as a loose indication of the actual rates, since the accuracy is limited by the use of LO cross sections and a parametric detector simulation. 226

11.1 Mean leading-jet p_T for events with no pileup and pileup generated by `Pythia8` and by the network (pGAN), before and after running *SoftKiller*. 238

INTRODUCTION

The past century has witnessed an unprecedented progress of humanity in studying the fundamentals of the universe. The creation of the Standard Model of particle physics, along with the development of experimental high energy physics, set a milestone in understanding the universe at the submicroscopic level via a remarkable interplay between theory and experiments on a global scale. The discovery of the Higgs boson in 2011 by the ATLAS and CMS collaborations at the Large Hadron Collider (LHC) completed the 47-year-old prediction of the last piece of the Standard Model, confirming its success in describing the fundamental matters and their interactions.

After the Higgs was observed, major efforts in the particle physics community have been dedicated to studying its properties via precision measurements and comparing the results with the Standard Model predictions to test the limits of the theory. While most results have been so far consistent with theory's predictions, the Higgs self-coupling parameter, which encapsulates key information to describe the shape of the Higgs potential that has strong implication on the stability of the universe, has never been measured. Setting limit on this parameter is one major goal of this thesis.

Despite its tremendous success, the Standard Model does not explain all the physical phenomena, such as the nature of dark energy and dark matter, the dominance of matters over antimatters, or the unnatural fine-tuning in the formulation of the Higgs mass. Supersymmetry (SUSY) theory emerges as a beautiful solution to the fine-tuning problem, with implication on the nature of dark matter as the lightest SUSY particle. This is also known as the "WIMP miracle." The last decade of experimental results has ruled out the existence of SUSY in most phase space, yet still leaves out the possibility of SUSY particles with long lifetimes. In the context of collider physics, these long-lived particles, created from proton-proton collisions, travel within the detector before decaying, inducing unique signatures in the detector's responses. A major part of this thesis is dedicated to searching for one type of these long-lived particles with the signature of "delayed photons," *i.e.,* photons that decay from these long-live particles and do not point to the collision vertex.

Given the null results so far in the quest of searching for physics beyond the Standard Model, one might take a step back and rethink the whole pipeline of new physics

searches. From one billion proton-proton collisions per second at the LHC, the current technology only allows for saving up to one thousand collision events per second for offline analysis. This drastic reduction requires us to carefully select only physics events whose signatures are most likely associated with new physics based on our own assumptions of how new physics should look like. But what if our assumptions are wrong? Have we been chasing the unicorn while ignoring the dragon? This question motivated us to explore a model-independent search strategy: let's catch anything that looks strange enough and examine it later. This idea materializes into a proposal for an anomaly trigger, which uses unsupervised learning algorithms trained on Standard Model processes to catch physics events that do not resemble Standard Model. This thesis will discuss the detail of the proposal and practical considerations in terms of implementations.

The LHC era coincides with the breakthrough of the deep learning era, largely due to the realization that general-purpose graphics processing units can speed up deep learning algorithms by several orders of magnitude. Since then, numerous deep learning algorithms have been developed with significant impacts across many scientific domains. This thesis will introduce several algorithms developed for high energy physics, where domain knowledge is particularly crucial in creating efficient algorithms for our highly complex data. More importantly, the next decade of the LHC will observe its biggest upgrade into the so-called High-Luminosity LHC, with ten times higher integrated luminosity collected over twelve years. This significant upgrade imposes unprecedented challenges to the computing system, where the required computing resources are projected to increase by more than fifty folds while the budget stays relatively flat. As a powerful universal approximator, a deep learning algorithm can replace some classical algorithms by providing an approximate solution at a fraction of the computing cost. Furthermore, deep learning can outperform many state-of-the-art traditional solutions, especially in the regime of supervised learning, due to its flexibility in the retrieval of non-tabular input data, which is ubiquitous in high energy physics, in combination with its powerful learning capability for very high dimension manifolds via backpropagation and stochastic gradient descent.

This thesis is organized into five parts. Part I lays the theoretical foundation for this thesis, including a summary of the Standard Model and the motivation to measure the Higgs self-coupling via double Higgs production in Chapter 1. Chapter 2 describes

SUSY as the solution for the hierarchy problem introduced by the Standard Model, motivating the searches for new physics in Part IV.

Part II describes the machines used in our experiments. In particular, Chapter 3 gives a brief description of the Large Hadron Collider in the CERN accelerator complex. Chapter 4 describes the CMS detector and its components, along with the trigger system and global data processing infrastructure.

In Part III, we focus on the studies of the Higgs bosons. Chapter 5 presents the results of the searches for nonresonant double Higgs production in the $\mathrm{HH} \to \mathrm{b\bar{b}}\gamma\gamma$ final state using CMS data collected in Run 2. Chapter 6 proposes a novel boosted $\mathrm{H} \to \mathrm{b\bar{b}}$ jet identification technique based on graph interaction networks that takes as input the jet constituents along with secondary vertex parameters. This technique can be used for future Higgs searches that benefit from final states including energetic Higgs bosons decaying into pairs of bottom quark-antiquark.

Part IV is dedicated to new physics searches. Chapter 7 describes the search for long-lived particles in the context of gauge-mediated supersymmetry breaking framework with delayed photon signatures using CMS Run 2 data. Chapter 8 describes the aforementioned proposal for an anomaly trigger in pursuit of model-independent new physics searches based on unsupervised learning algorithms and their implementations in the high-level trigger and Level-1 trigger systems in CMS.

Finally, Part V proposes a series of machine-learning solutions to address the computational challenges in the High-Luminosity LHC era. These challenges are explained in Chapter 9, where the opportunities for deep-learning solutions arise thanks to a novel heterogeneous computing platform being installed in the next run of the LHC. Chapter 10 addresses the noisy selection in the current trigger system and proposes an additional cleanup layer that can retain 99% signal while reducing the background by over one order of magnitude. Chapter 11 describes our efforts in replacing full-event simulation with recurrent generative adversarial networks, starting with the generation of pileup events. Chapter 12 introduces a more practical approach for fast simulation, where we use the encoder-decoder architecture to regress reconstructed event information from generator level, skipping the intermediate classical simulation step, which would result in about an order-of-magnitude reduction in computing and storage requirements for the collision simulation workflow.

Part I

Theoretical foundation

Chapter 1

THE STANDARD MODEL OF PARTICLE PHYSICS

The Standard Model (SM) of particle physics is a renormalizable quantum field theory describing the interactions of all the known elementary particles, governed by the gauge symmetry group $\mathrm{SU}(3)_c \times \mathrm{SU}(2)_L \times \mathrm{U}(1)_Y$, where the subscript c refers to the conserved quantum numbers for color, L for left-handedness, and Y for the hypercharge. There are three classes of particles in the SM: (1) the force-carrying particles, which are bosons of spin one, transmitting the electromagnetic, weak, and strong forces; (2) the matter particles, including quarks and leptons, which are fermions of spin one half; and (3) the Higgs particle, which is a boson of spin zero. Fermions are grouped into three *generations*: (ν_e, e, u, d), (ν_μ, μ, c, s), and (ν_τ, τ, t, b). Fig. 1.1 summarizes the particle content of the SM.

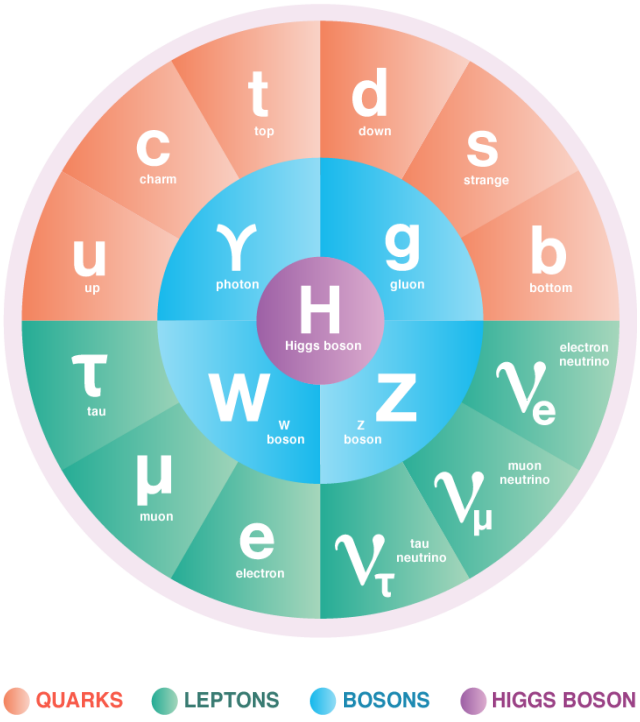


Figure 1.1: The catalog of elementary particles in the Standard Model [1].

The $\mathrm{SU}(3)_c$ group corresponds to the local symmetry where "gauge-ing," *i.e.,* requiring the Lagrangian is invariant under continuous space-time-dependent phase changes (local gauge transformation), gives rise to quantum chronodynamics (QCD),

where the strong interaction is invariant under rotations in color space. The gluons, which mediate the strong force, arise from the eight generators of the SU(3) symmetry group, which are also known as the Gell-Mann matrices [2].

Formally, starting with the Dirac equation for a free quark:

$$(i\gamma^{\mu}\partial_{\mu} - m)\psi(x) = 0, \tag{1.1}$$

where γ^{μ} are the Dirac matrices and ψ(x) is the $\mathrm{SU(3)}_c$ triplet for the quark field, one imposes the local $\mathrm{SU(3)}_c$ transformation:

$$\psi(x) \rightarrow e^{ig_s \frac{\lambda_j}{2}\theta_j(x)}\psi(x), \tag{1.2}$$

where g_s is the strong coupling constant, λ_j are the Gell-Mann matrices, and θ_j(x) are the rotation angles.

The invariance of the Dirac equation under such a local phase transformation is only possible by introducing the covariance derivative:

$$D_{\mu} = \partial_{\mu} - i\frac{g_s}{2}\lambda_j G_j^{\mu}(x), \tag{1.3}$$

where $G_j^{\mu}(x)$ are the gauge vector fields corresponding to the eight gluons. Consequently, the QCD Langrangian density becomes:

$$\mathcal{L}_{\mathrm{QCD}} = \bar{\psi}(x)(i\gamma_{\mu}D^{\mu} - m)\psi(x) - \frac{1}{4}F_{j,\mu\nu}F_j^{\mu\nu}, \tag{1.4}$$

where $F_j^{\mu\nu}$ is the field strength tensor:

$$F_j^{\mu\nu} = \partial^{\mu}G_j^{\nu} - \partial^{\nu}G_j^{\mu} - g_s f_{jkl}G_k^{\mu}G_l^{\nu}. \tag{1.5}$$

The non-abelian nature of the $\mathrm{SU(3)}_c$ group has profound consequences on the phenomenology of QCD. Quarks and gluons exhibit a phenomena called *confinement*, where they can neither be isolated nor observed directly [3]. Instead, they must group together to form *hadrons*, which can be either *mesons* (comprising one quark and one antiquark) or *baryons* (comprising three quarks). Since the Pauli exclusion principle forbids identical fermions to occupy the same quantum state, all hadrons are color singlets. Finally, as the strong coupling constant changes its value as a function of energy, the strong force becomes weakened at high energy, making quarks appear to be "free" when probed by high energy photons. This property is called *asymptotic freedom* [4].

Similarly, gauge-ing the $SU(2)_L \times U(1)_Y$ group, associated with the electroweak interaction, gives rise to a collection of massless spin 1 gauge bosons. However, the experimentally verified mediators of the weak forces, the $W^{\pm}$ and Z bosons, do have mass. Additionally, the fermion's mass term $m\bar{\psi}(x)\psi(x)$ in Eq. 1.4 is not gauge invariant. To solve this problem, the electroweak theory introduces an additional complex $SU(2)_L$ doublet of spin zero fields that will keep the full Langrangian invariant under $SU(2)_L \times U(1)_Y$ via the Brout-Englert-Higgs (BEH) mechanism [5–13]. The field is defined as:

$$\phi(x) = \frac{1}{\sqrt{2}} \begin{pmatrix} \phi_1 + i\phi_2 \\ \phi_3 + i\phi_4 \end{pmatrix}, \tag{1.6}$$

where ϕ_i (i = [1..4]) are normalized scalar fields. This field is introduced in the BEH mechanism via the BEH scalar part of the SM Lagrangian density:

$$\mathcal{L}_{\text{BEH}} = (D^{\mu}\phi)^{\dagger}(D_{\mu}\phi) - V(\phi). \tag{1.7}$$

The renomalizability and invariance of $SU(2)_L \times U(1)_Y$ require the BEH potential $V(\phi)$ to be of the form:

$$V(\phi) = -\mu^2\phi^{\dagger}\phi + \lambda(\phi^{\dagger}\phi)^2, \tag{1.8}$$

where λ and μ^2 are positive. At lower values of ϕ, the potential is dominated by the first term, exhibiting a concave downwards behaviors. As ϕ goes higher, the second term in Eq. 1.8 dominates the potential value, trending the function $V(\phi)$ upwards. This results in the characteristic "Mexican hat" shape of the BEH potential, as illustrated in Fig. 1.2.

The BEH potential has a set of degenerate minima lying on a ring in the complex plane with the radius:

$$|\phi_0| = \sqrt{\phi_0^{\dagger}\phi_0} = \sqrt{\frac{\mu^2}{2\lambda}} \equiv \frac{v}{\sqrt{2}}, \tag{1.9}$$

where v is known as the vacuum expectation value (vev) of the scalar potential $V(\phi)$. Each ground state on the ring is asymmetric under the U(1) symmetry of the Lagrangian. Therefore, the global U(1) symmetry is *spontaneously broken*: all ground states result in the same physical observables, but differ in their mathematical descriptions.

The non-zero vev allows for perturbative expansion of the field around the minimum by shifting the ϕ field by an amount of v, so that the new field $h = \phi - v$ is centered

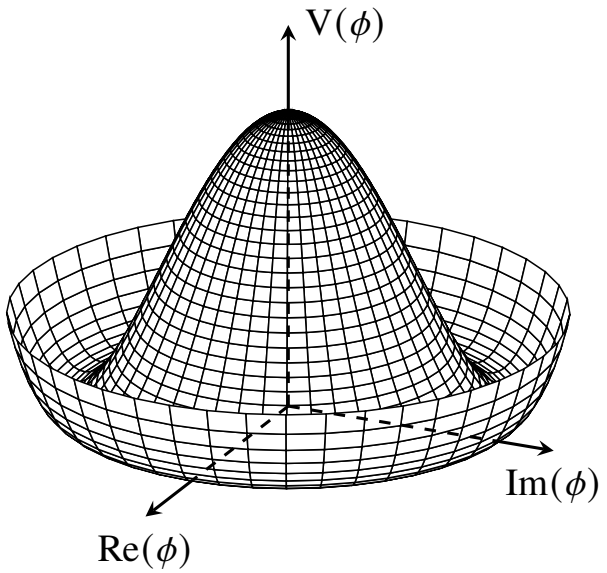


Figure 1.2: The "Mexican hat" shape of the Brout-Englert-Higgs potential $V(\phi)$ as a function of the complex scalar field ϕ.

at the vacuum, i.e., $\langle h \rangle = 0$. Additionally, we can choose the coordinates for $\phi(x)$ in Eq. 1.6 such that $\langle \phi_1 \rangle = \langle \phi_2 \rangle = \langle \phi_4 \rangle = 0, \langle \phi_3 \rangle = v + \langle h \rangle$, so that the field becomes:

$$\phi(h) = \frac{1}{\sqrt{2}} \exp\left(\frac{i\xi_a \sigma^a}{v}\right) \begin{pmatrix} 0 \\ v + h \end{pmatrix}, \tag{1.10}$$

where ξ_a are the fields and σ^a are the Pauli matrices with a summed over 1 to 3. This expression is equivalent to Eq. 1.6 in the infinitesimal fluctuations around the vacuum up to linear order. Under the $\mathrm{SU}(2)_L$ gauge transformation:

$$\phi \rightarrow \exp\left(i\lambda_L^a(x)\frac{\sigma^a}{2}\right)\phi, \tag{1.11}$$

by choosing the rotation angles $\lambda_L^a(x) = -\frac{2\xi_a}{v}$, the field becomes:

$$\phi(h) = \frac{1}{\sqrt{2}} \begin{pmatrix} 0 \\ v + h \end{pmatrix}. \tag{1.12}$$

The gauge transformation has entirely removed the dependence on ξ_a, or equivalently ϕ_1, ϕ_2, ϕ_4. The only physical degree of freedom left is the real scalar field $h(x)$ that corresponds to the a massive physical field, called the Higgs field. This gauge choice is called *unitary gauge*.

Expanding the unitary gauge field into the kinetic term in Eq. 1.7, we obtain:

$$(D_\mu \phi)^\dagger (D^\mu \phi) \supset \frac{g^2 v^2}{4} W_\mu^+ W^{-\mu} + \frac{(g^2 + g'^2)v^2}{8} Z_\mu Z^\mu, \tag{1.13}$$

where the covariant derivative takes the form:

$$D_\mu = \partial_\mu - i\frac{g'}{2}B_\mu - i\frac{g}{2}W_\mu^a\sigma^a, \tag{1.14}$$

where $W^\pm$ and Z fields are linear combinations of the $\mathrm{SU}(2)_L$ and $\mathrm{U}(1)_Y$ gauge bosons:

$$W_\mu^\pm = \frac{W_\mu^1 \mp iW_\mu^2}{\sqrt{2}}, \qquad Z_\mu = \frac{gW_\mu^3 - g'B_\mu}{\sqrt{g^2+g'^2}}, \tag{1.15}$$

and g and g' are the gauge couplings for the gauge groups $\mathrm{U}(1)_Y$ and $\mathrm{SU}(2)_L$, respectively.

For a real vector field A with the mass term m, $\mathcal{L} \supset \frac{1}{2}m^2A_\mu A^\mu$. Therefore, Eq. 1.13 gives rise to the mass of $W^\pm$ and Z bosons:

$$m_W^2 = \frac{g^2v^2}{4}, \qquad m_Z^2 = \frac{(g^2+g'^2)v^2}{4}. \tag{1.16}$$

Now expanding the unitary gauge field into the BEH potential in Eq. 1.8, we obtain:

$$V = \lambda v^2h^2 + \lambda_3 vh^3 + \frac{1}{4}\lambda_4 h^4 + \text{const.} \tag{1.17}$$

For a real scalar field ϕ with mass m, $V \supset \frac{1}{2}m^2\phi^2$. Therefore, the first term in Eq. 1.17 gives rise to the Higgs mass:

$$m_h = \sqrt{2\lambda v^2}. \tag{1.18}$$

The second and third terms in Eq. 1.17 describe the Higgs trilinear and quartic self-couplings. In the SM, $\lambda_3 = \lambda_4 = \lambda_{\mathrm{SM}} = m_h^2/(2v^2)$. Any deviation from the SM will start from the cubic and higher terms of the self-couplings, since any change in $O(h^2)$ will be absorbed into a redefinition of m_h.

Moreover, in view of the renormalization group (RG) method [15, 16], where the coupling parameters λ are "running" parameters depending on the energy scale μ via the beta function:

$$\beta(\lambda) = \frac{d\lambda}{d\log\mu}, \tag{1.19}$$

one can compute the one-loop beta functions for the Higgs quartic coupling:

$$16\pi^2\frac{d\lambda}{d\log\mu} = 12(\lambda^2 + \lambda y_t^2 - y_t^4) + O(g^4) + O(g^2\lambda), \tag{1.20}$$

where y_t is the Yukawa coupling between the Higgs field and the fermion fields that also runs on the energy scale μ. The negative quartic term in Eq. 1.20 implies that

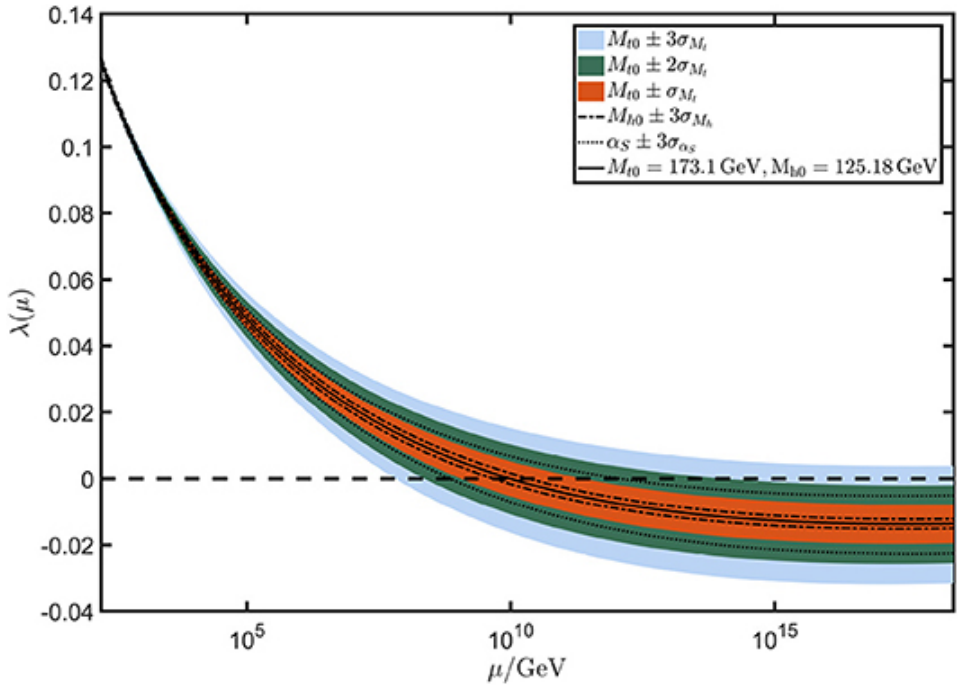


Figure 1.3: The RG evolution of the SM Higgs self-coupling as a function of the energy scale [14].

the Yukawa coupling y_t can drive the Higgs self-coupling λ to negative values at some high energy scale $\mu \sim 10^{10}$ GeV, as shown in Fig. 1.3. The negative Higgs self-coupling then creates a region with potential energy lower than the current electroweak vacuum found in Eq. 1.9, rendering it a metastable "false vacuum." Consequently, at some point in the universe, the false vacuum can decay to the true minima via bubble nucleation to favor the lower energy state [14, 17, 18].

The Higgs boson self-coupling therefore plays a crucial role in understanding the fate of the universe as well as in the search for any deviation from the SM, yet has never been directly measured. Measuring the Higgs trilinear self-coupling is the focus of Part. III of this thesis.

Chapter 2

PHYSICS BEYOND THE STANDARD MODEL

In the effective field theory approach of the Standard Model, the Yukawa interaction between the Higgs and fermion fields induces quadratic corrections to the Higgs mass that go up to the ultraviolet cutoff scale Λ_{UV} [19]:

$$\begin{aligned} m_h^2 &= m_0^2 + \Delta m_h^2, \\ \Delta m_h^2 &\supset -\frac{|y_{\mathrm{f}}|^2}{8\pi^2}\Lambda_{\mathrm{UV}}^2, \end{aligned} \tag{2.1}$$

where m_0 is the bare Higgs boson mass and Δm^2 are the loop corrections, which are dominated by the top quark loop. The Λ_{UV} cutoff scale is on the order of the Planck scale, $\mathrm{M}_P \sim O(10^{19})$ GeV, while the mass of the Higgs is experimentally measured to be $\sim$ 125 GeV. This turns Eq. 2.1 into a subtraction of two extremely large numbers that precisely differ by the square of the Higgs mass, which is 34 orders of magnitude smaller. This enormous fine-tuning inherently conflicts with the idea of *naturalness* [20], which suggests that physical parameters of a theory should be of the same order. In this case, physics at a lower energy scale (electroweak) should not be sensitive to the physics at a much higher energy scale (Planck scale). This conflict is known as the *hierarchy problem*, a key motivation for a new physics theory called *supersymmetry* (SUSY) [19].

SUSY theory posits a new spacetime symmetry between fermions and bosons, where each particle in the SM has a *superpartner*. In this scenario, every fermion loop in the Higgs mass correction in Eq. 2.1 is accompanied with a new scalar loop induced by the superpartner, as shown in Fig. 2.1.

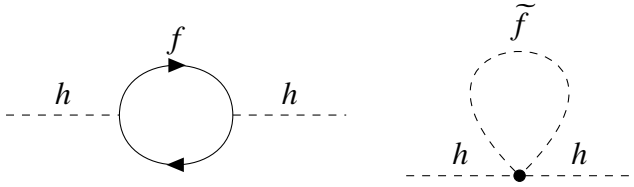


Figure 2.1: Example Feynman diagrams for the loop corrections of the Higgs boson mass in SUSY theory. The left diagrams shows the SM fermion loop, while the addition of the scalar superpartner correction is shown on the right diagram.

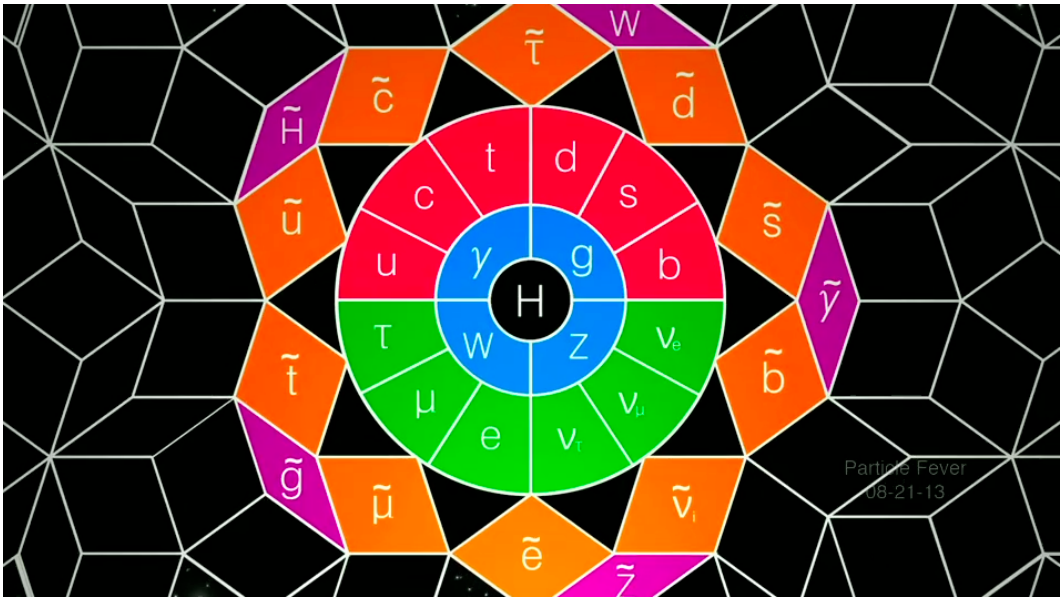


Figure 2.2: Illustration of the particle content in the MSSM as an extension of the Standard Model [21].

The new scalar superpartner, along with the SM fermion loops, contribute to the correction of the SM Higgs mass as:

$$\Delta m_h^2 \approx \frac{1}{16\pi^2}\left(-2|y_f|^2 + \lambda_S\right)\Lambda_{UV}^2, \tag{2.2}$$

where λ_S is the coupling of the Higgs field with the scalar superpartner. If $|y_f|^2 = \lambda_S/2$, the quadratic contribution of the Λ_{UV} cutoff would be exactly cancelled out in the Higgs mass correction, elegantly solving the hierarchy problem. There are many extended models of the SM that realizes supersymmetry. This thesis focuses on the Minimal Supersymmetric Standard Model (MSSM), the simplest extension of the SM where a minimum amount of new particle states and interactions are added to remain consistent with experimental results in the SM.

The naming conventions of SUSY particles are follows: superpartners of SM fermions are bosons with "s-" prefixing to the SM fermion counterpart, *e.g.,* the superpartner of tau lepton is "stau slepton"; superpartners of SM bosons are fermions with "-ino" appending to the SM boson counterpart, *e.g.,* the superpartner of Higgs is called "Higgsino" and the "-on" is dropped if the boson's name ends with it, *e.g.,* gluon becomes "gluino." To denote the superpartner fields or particle states, a tilde is added on their SM counterparts, *e.g.,* the top quark "t" has the superpartner called stop squark "$\widetilde{\text{t}}$." An illustration of particle content in the MSSM with the superpartners of the SM is shown in Fig. 2.2.

In constructing the MSSM, to allow the symmetry algebra to close off-shell, preserving SUSY as a quantum theory, additional fields called "auxiliary" fields must be added. These are bookkeeping fields to ensure that the number of degrees of freedom for boson and fermion fields are the same at higher orders in perturbation theory [19].

If the symmetry in SUSY theory is unbroken, each superpartner would have the exact same mass and charge as its SM counterpart. Since no such particles have been discovered, supersymmetry needs to be a *broken* symmetry in the vacuum state to drive the superpartners' masses to the unexplored TeV scale. The Lagrangian term that "breaks" SUSY, *i.e.,* introduces the difference between particles and their superpartners, must not contain dimensionless coupling parameters to preserve the exact cancellation of the Λ_{UV}^2 term in Eq. 2.1 from the unbroken SUSY. The effective Lagrangian of the MSSM therefore takes this form:

$$\mathcal{L} = \mathcal{L}_{\mathrm{SUSY}} + \mathcal{L}_{\mathrm{soft}}, \tag{2.3}$$

where $\mathcal{L}_{\mathrm{SUSY}}$ contains all the gauge and Yukawa interactions that do not break supersymmetry and $\mathcal{L}_{\mathrm{soft}}$ contains only mass terms and coupling parameters with positive mass dimension that violate supersymmetry. The contribution of $\mathcal{L}_{\mathrm{soft}}$ to the Higgs mass correction is of the form [19]:

$$\Delta m_h^2 \supset m_{\mathrm{soft}}^2 \left(\frac{\lambda}{16\pi^2} \log \frac{\Lambda_{\mathrm{UV}}}{m_{\mathrm{soft}}} \right), \tag{2.4}$$

where m_{soft} is the largest mass scale associated with $\mathcal{L}_{\mathrm{soft}}$. As this finite contribution depends only logarithmically on Λ_{UV}, any modification of physics at higher energy scale does not significantly affect physics at the electroweak scale, preserving naturalness, assuming m_{soft} is on the TeV scale. This process is therefore called *soft* supersymmetry breaking (where the "softness" comes from the fact that only physics processes with low energies are being changed from supersymmetry breaking).

Different frameworks have been proposed to generate terms in $\mathcal{L}_{\mathrm{soft}}$ that sufficiently break supersymmetry. As there is no appropriate field in the MSSM that can acquire the non-zero vev to be the primary source of SUSY breaking, additional fields must be added to the MSSM in which breaking can occur. These fields can only come from a *hidden sector*, where the spontaneous symmetry breaking effects are then mediated to the visible sector, which is the MSSM, via a *messenger sector*. A framework of interest in this thesis to describe the mediating interactions is called gauge-mediated supersymmetry breaking (GMSB) [22–31].

In GMSB, assuming the vev acquired in the hidden sector is $\langle F \rangle$, the scalar and the auxiliary fields in MSSM will acquire vev's $\langle S \rangle$ and $\langle F_S \rangle$, respectively, such that:

$$\langle F_S \rangle \lesssim \langle F \rangle \ll M_P. \tag{2.5}$$

The masses of the gauginos will take the form:

$$m_a \sim \frac{g_a^2}{16\pi^2}\Lambda, \tag{2.6}$$

where g_a are parametrization of the SM gauge couplings: g_s, g, and g'; $\Lambda = \langle F_S \rangle / \langle S \rangle$ is the effective SUSY breaking scale. The scalar masses for the MSSM fields are given by:

$$\widetilde{m}^2 \sim 2\Lambda^2 \sum_{a=1}^{3} C_a, \tag{2.7}$$

where C_a is the quadratic Casimir invariant of the relevant MSSM scalar [32]. The lightest supersymmetric particle (LSP) is the gravitino, whose mass takes the form:

$$m_{\widetilde{G}} = \frac{\langle F \rangle}{\sqrt{3} M_P}. \tag{2.8}$$

An important property of the MSSM is the conservation of the R-parity:

$$P_R = (-1)^{3(B-L)+2s}, \tag{2.9}$$

where B is the baryon number, L the lepton number, and s the spin of the particle. All SM particles have even R-parity ($P_R = +1$) while all the supersymmetric particles (sparticles) have odd R-parity ($P_R = -1$). Conservation of R-parity requires that every interaction vertex contains an even number of odd R-parity $P_R = -1$. Consequently:

- The LSP must be absolutely stable.
- Decay products of sparticles other than the LSP must contain an odd number of sparticles.
- In a pp collision, the sparticles are always pair produced.

In GMSB, sparticles decay into a gravitino plus other SM particles with the decay rate:

$$\Gamma(\widetilde{X} \to \widetilde{G}X) = \frac{m_{\widetilde{X}}^5}{16\pi\langle F\rangle^2}\left[1 - \left(\frac{m_X}{m_{\widetilde{X}}}\right)^2\right]^4, \tag{2.10}$$

where $\widetilde{X}$ is a sparticle and X is its SM counterpart. For large vev $\langle F\rangle$ in the hidden sector, the decay rate of the sparticle $\widetilde{X}$ could be very small, resulting in a long lifetime. Searching for such long-lived sparticles in the GMSB model is the focus of Ch. 7 in Part IV of this thesis.

Part II

Experimental apparatus

Chapter 3

THE LARGE HADRON COLLIDER

The Large Hadron Collider (LHC) [33] is the world's most powerful particle accelerator. Constructed by the European Organization for Nuclear Research (CERN), it consists of a 26.7-kilometer ring of superconducting magnet located on the border between Switzerland and France that circulates two counter-rotating hadron beams and delivers proton-proton (pp) collisions at center-of-mass energies up to 14 TeV with a peak crossing rate of 40 MHz in its most common configuration. Its goal is to provide the data needed for physicists to test different theories of particle physics, to understand the mechanism of electroweak symmetry breaking through measurements of properties of the Higgs boson, and to search for evidence of new particles beyond the Standard Model, such as supersymmetry and dark matter.

The LHC is located within the CERN accelerator complex, which is a series of machines that accelerate particles to increasingly higher energy. The protons start from a bottle of hydrogen gas, where an electric field strips the electrons out of hydrogen atoms to produce the protons. These protons go through the Linac 2 accelerator to achieve an energy of 50 MeV, and then pass through the Proton Synchrotron Booster (PSB), the Proton Synchrotron (PS), and the Super Proton Synchrotron (SPS), which boost the proton beams to 1.4 GeV, 25 GeV, and 450 GeV, respectively, before transferring to the two beamlines of the LHC. The layout of the CERN accelerator complex is shown in Fig. 3.1.

Inside the LHC, the two counter-rotating beams are captured, accelerated, and stored with the 400 MHz superconducting radio-frequency cavity system. The beams take approximately 20 minutes to reach their maximum energy of 6.5 TeV before they are brought to collision at four collision points that house the four main experiments at the LHC: ATLAS, ALICE, CMS, and LHCb. The two beams collide at a half crossing angle of 150 microradian (μrad) at the start of the collisions, and then reduce by 10 μrad every few hours down to a minimum of 120 μrad as the beams lose their intensity. During the Run 2 data-taking period (from 2016 to 2018), the interval between two consecutive collisions is 25 ns.

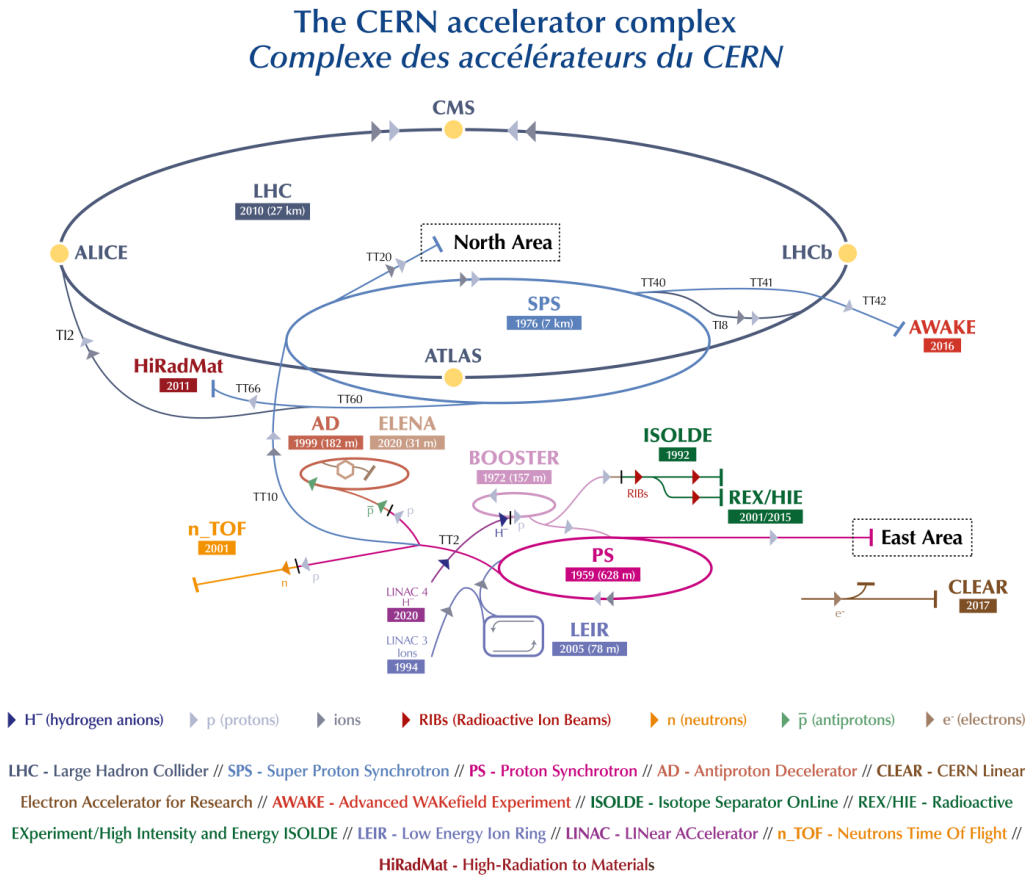


Figure 3.1: The schematic layout of the CERN accelerator complex [34].

For any physics process occurring at the LHC, the number of events produced N_{exp} is the product of the experimental cross section σ_{exp} and the integrated luminosity:

$$N_{\text{exp}} = \sigma_{\text{exp}} \int \mathcal{L}(t)dt, \tag{3.1}$$

where $\mathcal{L}(t)$ is the instantaneous luminosity, defined as [33, 36]:

$$\mathcal{L} = \frac{N_b^2 n_b f_{\text{rev}} \gamma_r}{4\pi\epsilon_n \beta^*} R, \tag{3.2}$$

where N_b^2 is the number of protons per bunch, n_b is the number of bunches per beam, f_{rev} is the revolution frequency, γ_r is the relativistic factor, ϵ_n is the normalized transverse beam emittance (measuring the average spread of the beam), β^* is the beta function at the collision point (measuring the transverse size of the beam), and R is the geometric luminosity reduction factor due to non-zero crossing angle at the collision point, defined as:

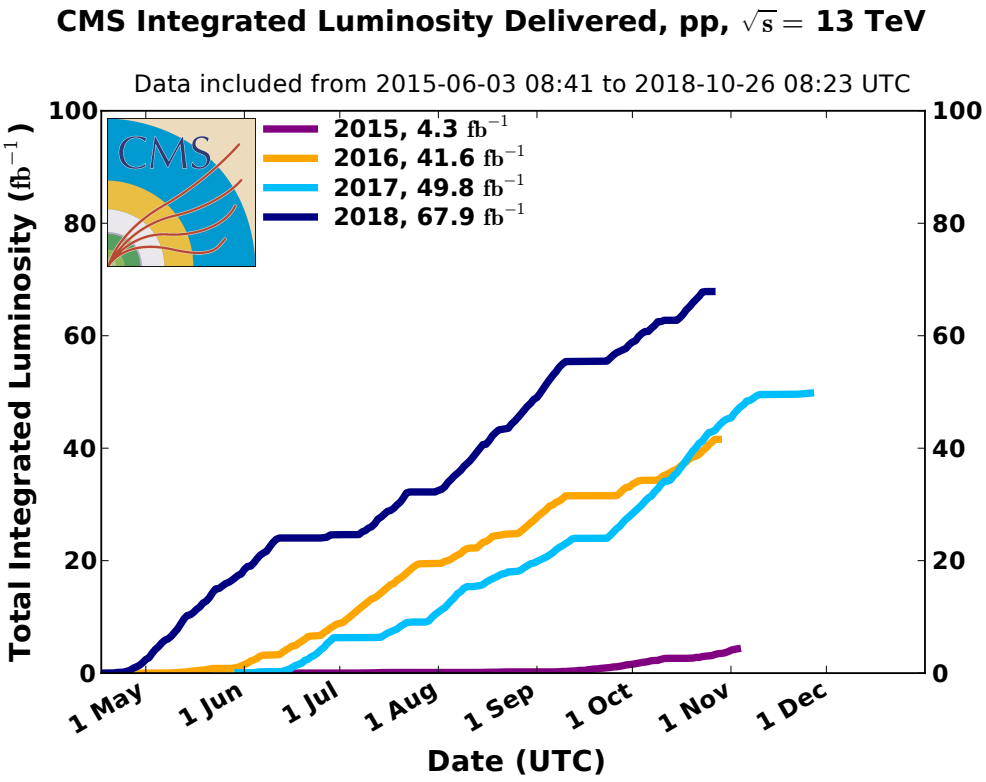


Figure 3.2: Cumulative luminosity versus day delivered to CMS during stable beams for pp collisions at nominal center-of-mass energy during the Run 2 data-taking period [35].

$$R = \left[1 + \left(\frac{\theta_c \sigma_z}{2\sigma^*}\right)^2\right]^{-\frac{1}{2}}, \tag{3.3}$$

where θ_c is the full crossing angle at the collision point, σ_z is the RMS bunch length and σ^* is the transverse RMS beam size at the collision point.

The designed peak luminosity of the LHC is $\mathcal{L}(t) = 10^{34}$ cm^{-2}s^{-1} [36]. Fig. 3.2 shows the integrated luminosity delivered to the CMS experiment during the Run 2 data-taking period, which is about 41.6 fb^{-1}, 49.8 fb^{-1}, and 67.9 fb^{-1} during 2016, 2017, and 2018, respectively. Note that the LHC also delivered around 4.3 fb^{-1} to CMS during 2015, but this small amount of data is not considered in most physics analyses since adding them would involve substantial computational resources regarding data processing, simulation, and calibration with insignificant gain.

An LHC collision occurs when two proton bunches from the two beams pass through each other. As each bunch contains up to 1.15×10^{11} protons, there are multiple

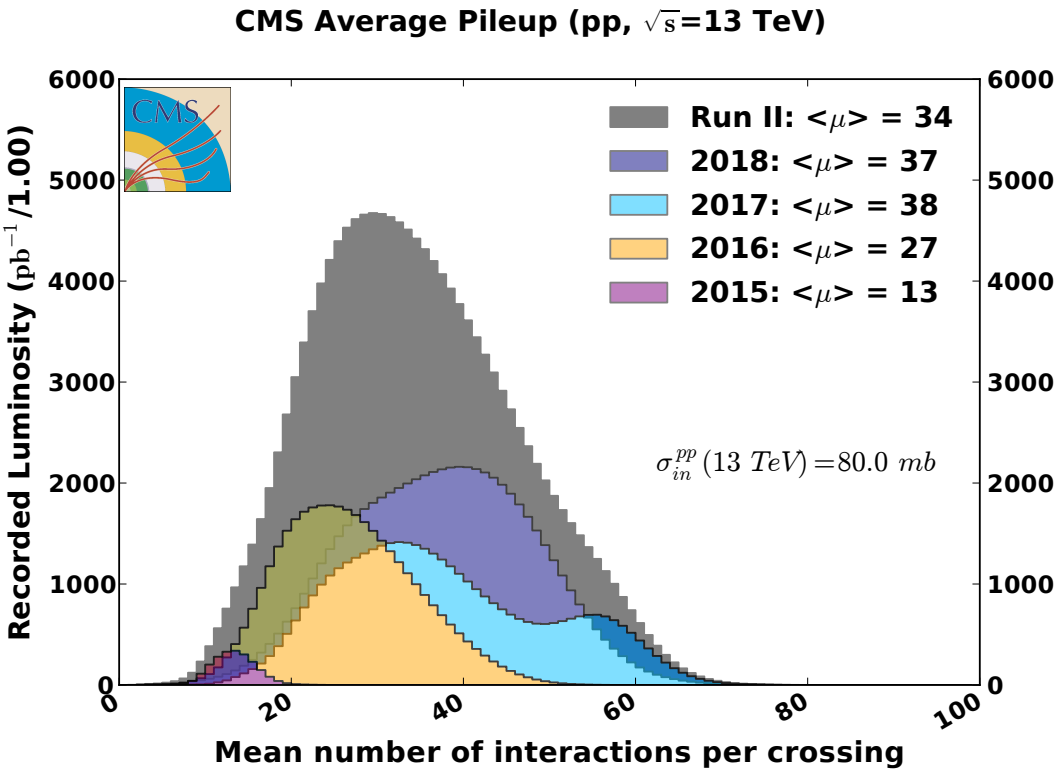


Figure 3.3: Distributions of average number of interactions per crossing (pileup) for pp collisions in each year during the Run 2 data-taking period [35].

collisions happen within one bunch crossing. Each collision event therefore contains many superposed particle interactions, in which typically at most one interaction of interest triggers the system to be saved for downstream processing. The other interactions, called *pileup* interactions, are parasitic background processes that mainly consist of low transverse momentum particles in the events. These pileup interactions require further efforts to simulate, characterize, and reduce in analyses. The average number of pileup per bunch crossing can be computed as follows:

$$N_{\text{pileup}} = \frac{\sigma_{\text{inel}}\mathcal{L}}{f}, \tag{3.4}$$

where σ_{inel} is the inelastic pp scattering cross section (measured to be 71.3 mb at 13 TeV [37]), $\mathcal{L}$ is the usual instantaneous luminosity, and f is the frequency of bunch crossings (with nominal value of 28.7 MHz for 2018). The pileup distributions for each year during the Run 2 data-taking period are shown in Fig. 3.3.

Chapter 4

THE COMPACT MUON SOLENOID EXPERIMENT

The Compact Muon Solenoid (CMS) is one of the two general-purpose detectors operating at the LHC. Its most outstanding features include a superconducting solenoid magnet coil that produces a magnetic field of 3.8 Tesla, wrapping a system of hadron calorimeter, electromagnetic calorimeter, and a silicon-based pixel and strip tracker. Outside the solenoid magnet coil is a muon chamber system interspersed with iron return yoke. The total system has a cylindrical structure that spans 28.7 m long, with a radius of 7.5 m, and weighs 14 000 tonnes. A cutway diagram of the CMS detector is shown in Fig. 4.1, and Fig. 4.2 shows a cross-sectional-slice view illustrating the interactions of various particle types with different detector components. More detailed description of the CMS detector can be found in [38].

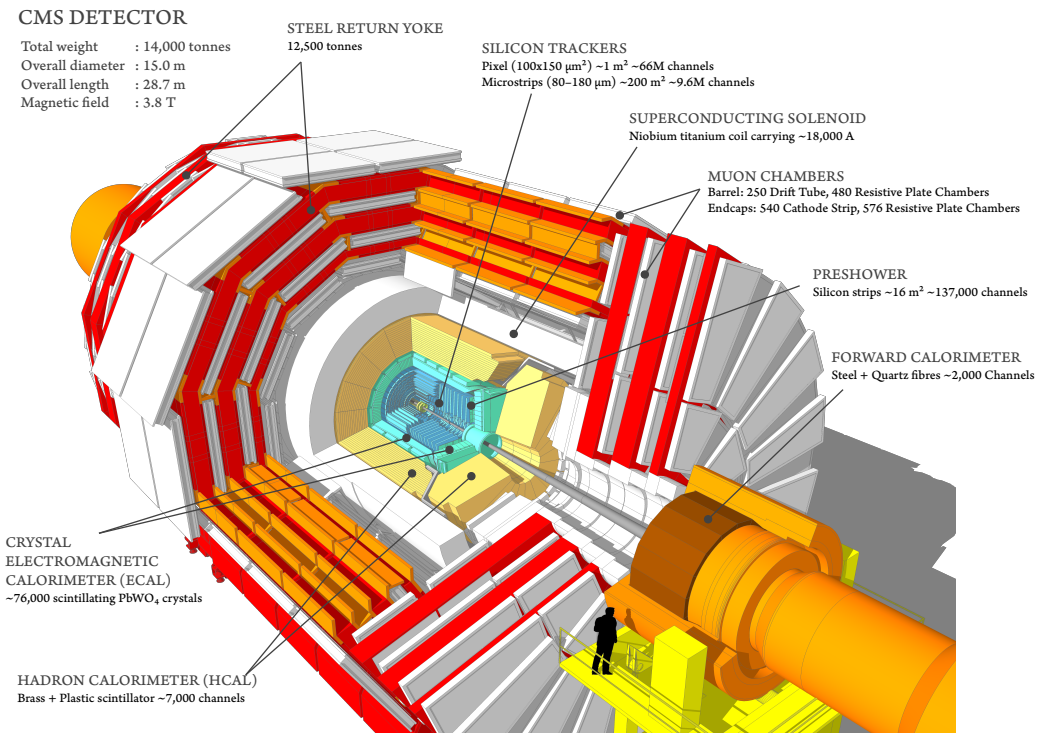


Figure 4.1: Cutway diagrams of the CMS detector [39].

CMS adopts a Cartesian coordinate system with the origin located at the nominal interaction point. The x- and z-axes lie on the horizontal plane, with the positive x-axis points toward the center of the LHC ring, the positive z-axis points along the

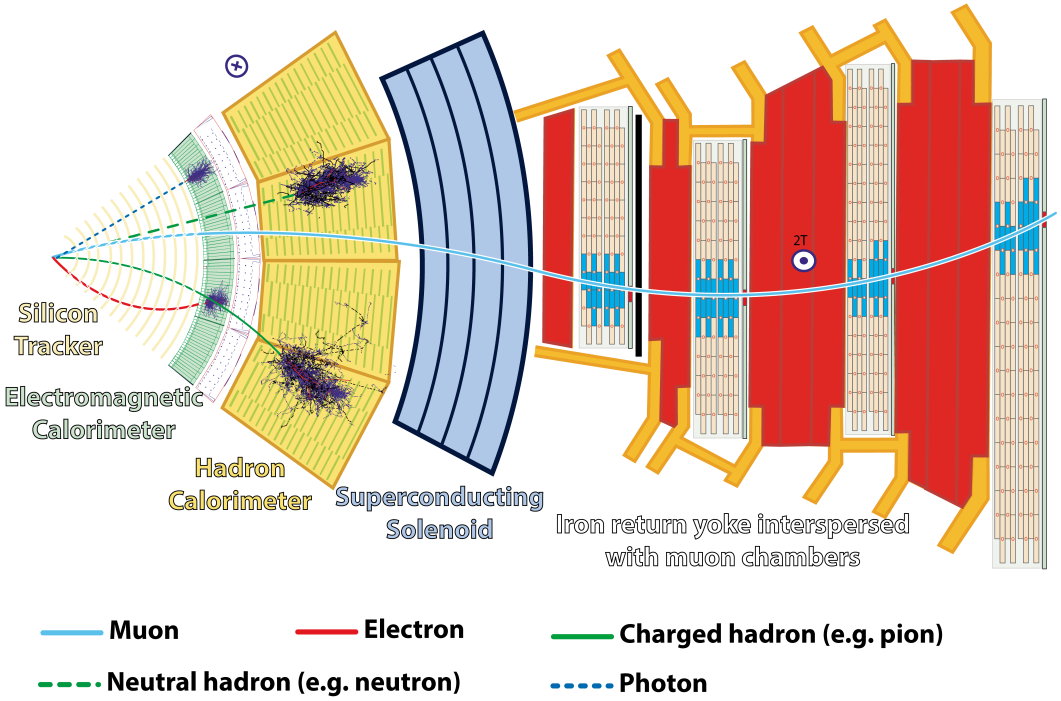


Figure 4.2: A cross-sectional-slice view of the CMS detector illustrating the interactions of various particle types with different detector components [40].

counter-clockwise beam at the intersection point westward of CMS. The positive y-axis points vertically upward. Alternatively, CMS also commonly uses a modified cylindrical coordinate system, with the radial distance $\rho = \sqrt{x^2 + y^2}$, the azimuthal angle ϕ defined in the transverse x-y plane measured from the +x axis, and the polar angle θ measured from the +z axis. In hadron collider physics, the *pseudorapidity* $\eta \equiv -\ln\left[\tan\frac{\theta}{2}\right]$ is used instead of θ. The pseudorapidity is equivalent to the *rapidity* of a particle in the limit $p \gg m$, where p and m are the momentum and mass, respectively, of the particle. The rapidity y is defined as:

$$y \equiv \frac{1}{2}\log\frac{E + p_z}{E - p_z}. \tag{4.1}$$

Expanding y in terms of η and $\alpha = m/p_{\mathrm{T}}$, where $p_{\mathrm{T}} = p\sin\theta$, we obtain:

$$\begin{aligned} y &= \frac{1}{2}\log\frac{\sqrt{\cosh^2\eta + \alpha^2} + \sinh\eta}{\sqrt{\cosh^2\eta + \alpha^2} - \sinh\eta} \\ &\approx \eta - \frac{1}{2}\alpha^2\tanh\eta + O(\alpha^3) \xrightarrow{\alpha\to 0} \eta. \end{aligned} \tag{4.2}$$

The rapidity y is used to define the angular separation between particles, $\Delta R \equiv \sqrt{(\Delta y)^2 + (\Delta\phi)^2}$, which is Lorentz invariant under a boost along the longitudinal

axis: $dy \xrightarrow{z\text{−boost}} dy$. While the rapidity depends on both the particle mass and polar angle, the pseudorapidity only depends on the polar angle and is approximately Lorentz invariant under z boosts for energetic particles ($p \gg m$), which are of interest to CMS.

4.1 The superconducting solenoid

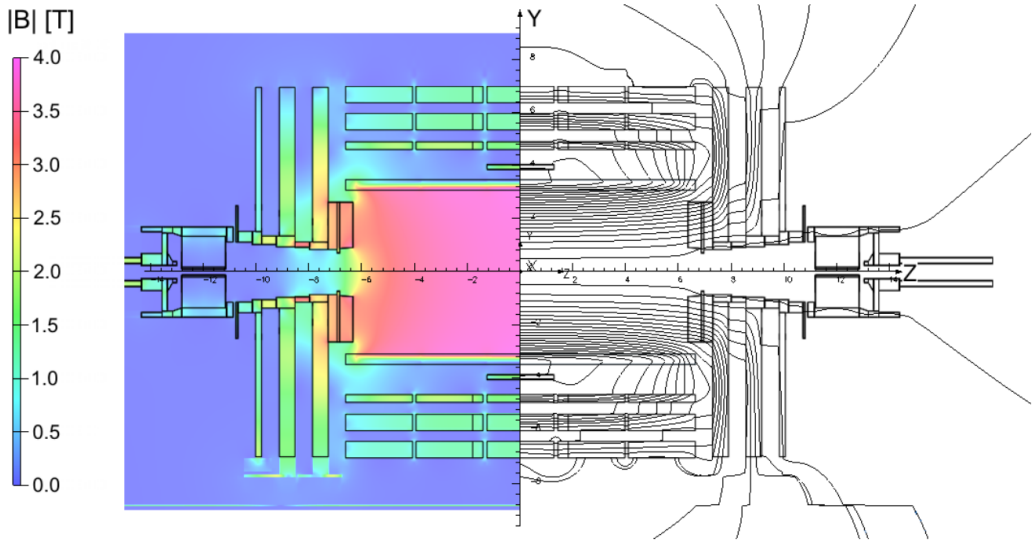


Figure 4.3: The magnetic field values (left) and the magnetic field lines (right) produced by the superconducting solenoid magnet of the CMS detector [41].

The superconducting solenoid magnet of CMS is the central component of the detector. It creates a 3-m radius and 12.5-m length free bore, composed of 2179 turns of superconducting wire braid made of NbTi wire wound in 4 layers, producing a central homogeneous magnetic field of 3.8 T along the beam line inside the solenoid [38, 42–44]. It is embraced by a 10 000-ton return yoke made of construction steel. The return yoke includes 5 dodecagonal wheels in the barrel and six disks at the endcaps. Each barrel wheel except the central one has three layers of steel, divided in 12 sectors in the transverse plane. The central barrel wheel has an extra layer of steel. The map of the magnetic field in the $y - z$ plane is shown in Fig. 4.3.

4.2 The tracker

The silicon tracking system is designed to record the trajectories of charged particles coming from interaction points, thus measuring their momenta as well as primary and secondary vertices. This subdetector system, located closest to the collision points, consists of an inner pixel detector and an outer strip tracker, with an outer

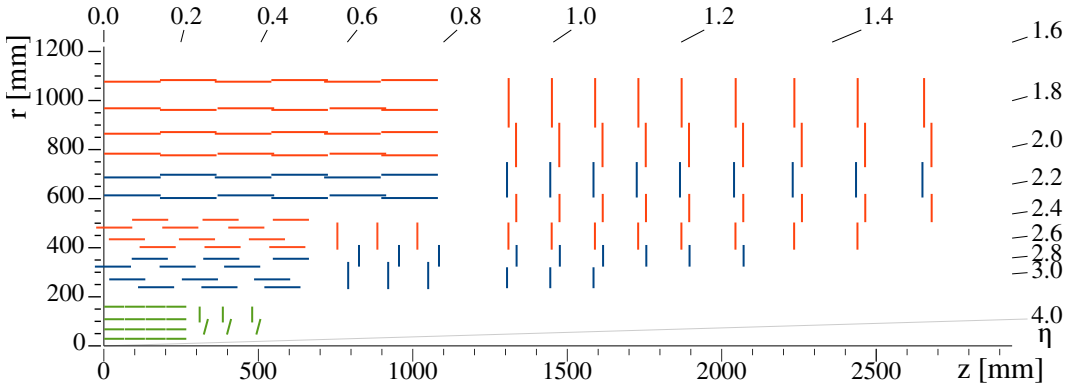


Figure 4.4: Diagram of the CMS inner tracker in one r-z quadrant, with the collision point at the origin. Green lines depict the pixel detector, while the single-sided and double-sided strip trackers are shown in red and blue, respectively [45].

radius of 110 cm and total length of 540 cm, covering the range $|\eta| < 2.5$. Fig. 4.4 shows the layout of the CMS inner tracker.

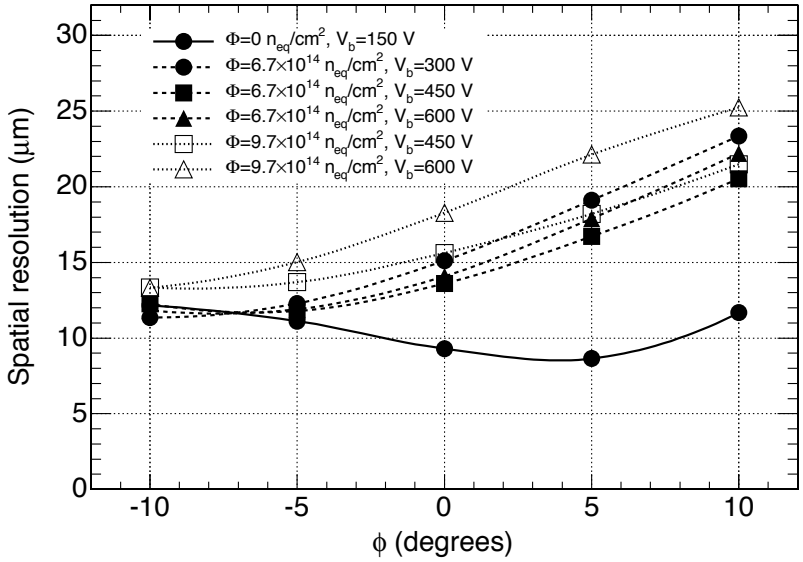


Figure 4.5: Spatial resolution of the pixel along the $r-\phi$ direction as a function of the angle between the track direction and the normal to the sensor plane [38].

The inner pixel detectors compose of a total of 66 million silicon sensors, each with a pixel size of $100\times150\mu m^2$ in $r\phi\times rz$. The barrel region contains three layers of pixels at radii of 4.3, 7.3, and 10.4 cm, consisting of 48 million pixels. The remaining 18 million pixels are in the four disks of the endcap. As the peak luminosity approaches two times the nominal design value during Run 2, the inefficiencies of the pixel detectors increase by 16% due to a limited readout bandwidth [46]. To

maintain the high tracking efficiency, the CMS collaboration installed a new pixel detector in March 2017 as part of the Phase I upgrade, with four barrel layers and six endcap disks, containing 79 million pixels in the barrel and 45 million pixels in the endcap. The spatial resolution for the inner pixel detectors is shown in Fig. 4.5.

The outer strip tracker consists of 9.3 million strip sensors. Its barrel region includes 2 parts: a Tracker Inner Barrel (TIB) and a Tracker Outer Barrel (TOB), while its endcap region is divided into the Tracker End Cap (TEC) and Tracker Inner Disks (TID). The TIB consists of 4 layers of 320-μm-thick silicon sensors and a strip pitch of 80 to 120 μm. The TOB includes 6 layers of 500-μm-thick sensors and a strip pitch of 120 to 180 μm. As the TOB is farther away from the collision points, the particle flux and radiation levels are smaller compared to TIB, thicker sensors can be used to maintain a good signal over noise ratio for longer strip length and wider pitch. The TEC comprises 9 disks of 320 μm silicon sensors and the TID comprises 3 small disks of silicon sensors of the same thickness that fill the gap between the TIB and TEC.

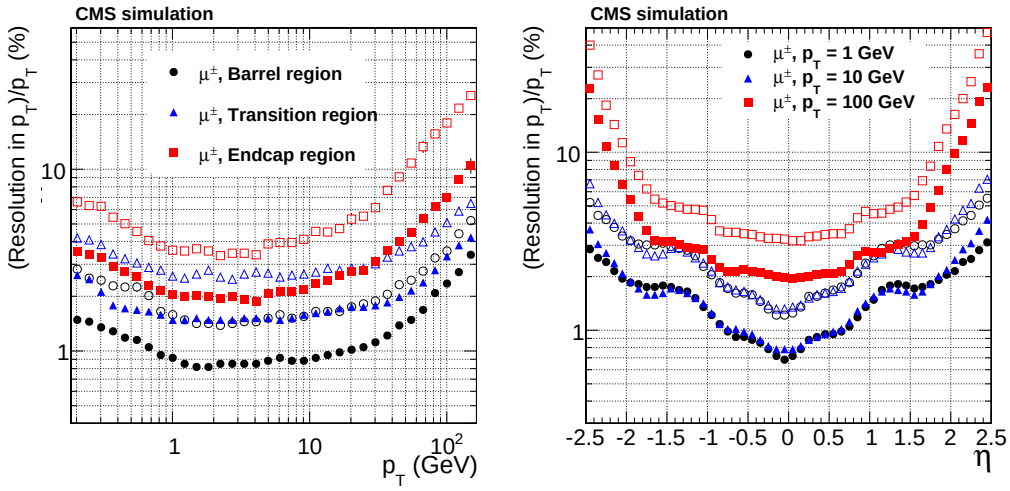


Figure 4.6: Transverse momentum resolution as a function of p_{T} (left) and η (right) for single, isolated muons in the barrel, transition, and endcap regions of the tracker [47].

Fig. 4.6 shows the transverse momentum resolution as a function of p_{T} and η of the tracker. The best resolution (less than 1%) is achieved with particles having transverse momenta close to 1 GeV and trajectories on the transverse plane.

4.3 The electromagnetic calorimeter

The electromagnetic calorimeter (ECAL) is a hermetic, homogeneous calorimeter surrounding the tracker. It is made of lead tungstate ($PbWO_4$) crystals, which have

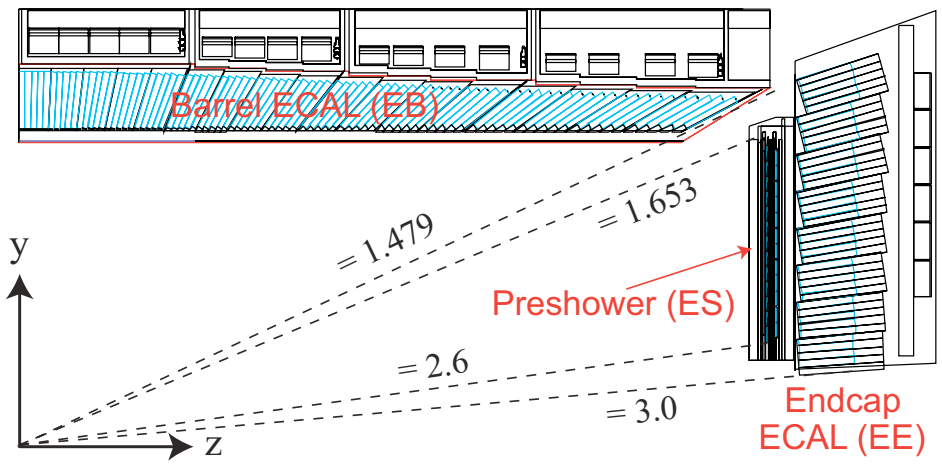


Figure 4.7: Layout of the CMS electromagnetic calorimeter in one r-z quadrant [38].

short radiation length and Moliére radius ($X_0 = 0.85\,\text{cm}$, $R_M = 2.19\,\text{cm}$) [38, 48], allowing for better shower position resolution and a compact calorimeter. The ECAL central barrel region (EB) contains 61 200 crystals with pseudorapidity coverage up to $|\eta| < 1.48$, closed by the two endcaps (EE) with 16 468 crystals, extending the coverage up to $|\eta| < 3.0$. Crystals are positioned slightly off-pointing with an angle of $\sim 3^\circ$ relative to the interaction point to avoid cracks aligned with particle trajectories. Additionally, a preshower detector (ES) made of 4288 sensors and 137 216 silicon strips is placed in front of the endcaps to identify neutral pions (π^0) and improve granularity. Fig. 4.7 shows a geometrical layout of the ECAL.

In the barrel region, two avalanche photodiodes (APDs), each with an active area of $5 \times 5\ \text{mm}^2$, are glued to the back of each crystal. The APD has rise time less than 2 ns, with an operating voltage between 340 – 430 V and a typical dark current of 3 nA. Photodetectors in the endcaps are vacuum phototriodes (VPTs). Each VPT, measured 25 mm in diameter, is glued to the back of each endcap crystal. When an electron or photon hits the ECAL crystals, a light signal is produced via the scintillation process. This scintillation light produces ADP/VPT pulses that are recorded and further processed by the readout electronics system.

The energy resolution of the ECAL is parameterized as a function of energy [38]:

$$\left(\frac{\sigma}{E}\right)^2 = \left(\frac{S}{\sqrt{E}}\right)^2 + \left(\frac{N}{\sqrt{E}}\right)^2 + C^2, \tag{4.3}$$

where S is the stochastic term, related to statistic fluctuations in the signal, N the noise, related to electronics noise and pileup, and C the constant term, related

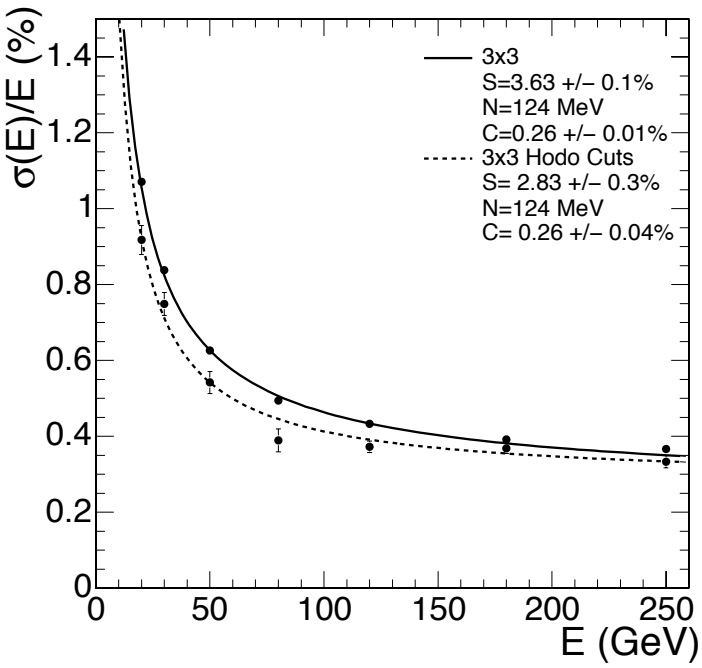


Figure 4.8: ECAL energy resolution as a function of electron energy measured in a 3×3 crystal cluster [38].

to the uncertainties associated with channel-by-channel calibration, leakage, dead material, etc. These terms are measured with a parametric fit to the resolutions obtained from Gaussian fits to the reconstructed energy distributions in different energy bins, with the values shown in Fig. 4.8. Note that while the momentum resolution of the tracker increases linearly with p_{T}, as shown in Fig. 4.6 (since it depends on curvatures of the particle trajectories in the magnetic field), the energy resolution in the ECAL improves with higher energy and is below 1% for electrons above 30 GeV. Additionally, the ECAL is the only source of information for charged particles in the forward region ($|\eta| > 2.5$), where the tracker does not cover.

4.4 The hadron calorimeter

In between the ECAL and the superconducting solenoid lies the hadron calorimeter (HCAL), which is made to measure the energy of the hadrons produced from particle showers. Unlike the homogeneous ECAL, the HCAL is a *sampling* calorimeter, meaning it is made of alternating layers of dense absorber and tiles of plastic scintillator. When a hadronic particle hits the brass absorber, cascades of secondary particles are produced and interact with the alternating layers of active scintillation material, causing them to emit blue-violet light. A tiny optical wavelength-shifting

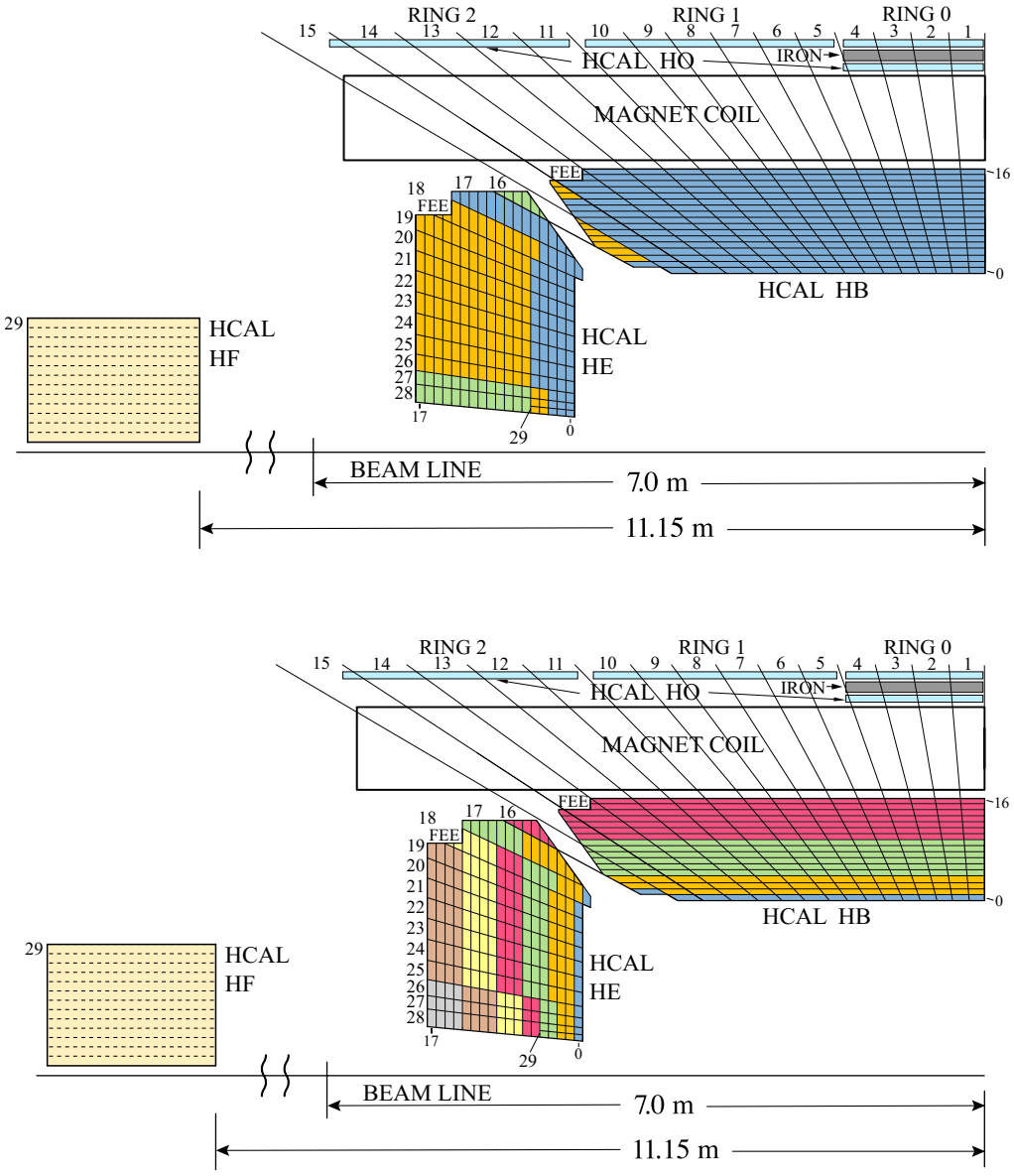


Figure 4.9: Layout of the CMS hadron calorimeter in one r-z quadrant before (top) and after (bottom) the SiPM upgrade. "FEE" indicates the Front End Electronics' locations. Light from layers depicted with the same color are optically added together before reaching the photosensors. The superior photon detection efficiency and response of SiPMs allow for an increased longitudinal granularity with up to 7 depths in HE and 4 depths in HB after the upgrade [49].

fiber in each tile absorbs this light and carries it to the readout system. The photodetectors in the HCAL are hybrid photodiodes (HPDs), which can amplify the calorimetry signals by approximately 2000 times. As part of the Phase I upgrade,

HPDs were replaced by silicon photomultipliers (SiPMs) in HE (2017) and HB (2019) as they offer 2.5 times higher photon detection efficiency and 400 times higher response while being insensitive to magnetic fields [49–51].

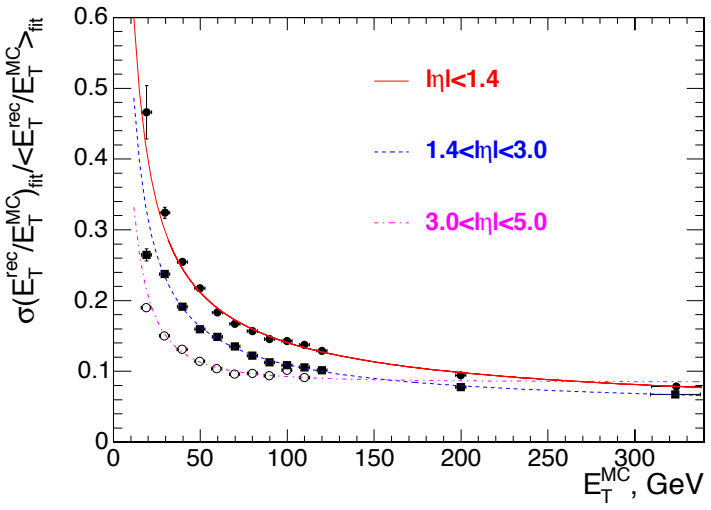


Figure 4.10: The transverse energy resolution as a function of the simulated jet transverse energy for barrel jets ($|\eta| < 1.4$), endcap jets ($1.4 < |\eta| < 3.0$), and forward jets ($3.0 < |\eta| < 5.0$) [38].

The HCAL consists of 4 components: the hadron barrel (HB) with 32 η towers covering the pseudorapidity region $|\eta| < 1.4$, the hadron outer (HO) containing 10-mm-thick scintillators outside the outer vacuum tank of the coil covering the region $|\eta| < 1.26$, the hadron endcap (HE) with 14 η towers covering the region $1.3 < |\eta| < 3.0$, and the hadron forward (HF) covering the region $3.0 < |\eta| < 5.0$ with steel/quartz fiber. The structure of the HCAL is sketched in Fig. 4.9. Fig. 4.10 shows the jet energy resolution of different HCAL components as functions of transverse energy (E_T).

4.5 The muon system

As muons can penetrate several meters of iron without interacting, they are not stopped by CMS calorimeters. A dedicated muon detector system is therefore placed at the very edge of the experiment, outside the solenoid, to identify muons. Based in gas ionization chambers, the muon system consists of 3 different technologies: drift tube chambers (DTs), cathode strip chambers (CSCs), and resistive plate chambers (RPCs). The DTs cover the barrel region with the pseudorapidity range up to $|\eta| < 1.2$, where the neutron induced background is small, the muon rate is low and the residual magnetic field in the chamber is low; the CSCs cover the endcap

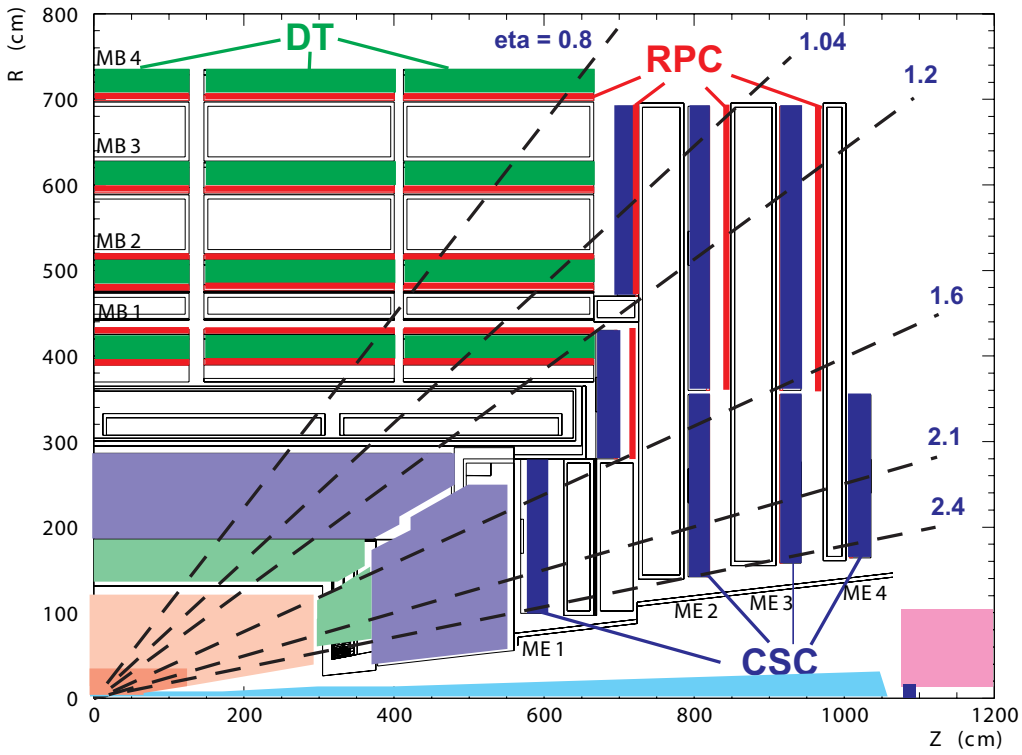


Figure 4.11: Layout of the CMS muon system in one r-z quadrant [38].

region $0.9 < |\eta| < 2.4$, where the muon rate, the neutron induced background rate, and the magnetic field are all high; the RPCs are deployed in both the barrel and endcap, operating in avalanche mode to ensure good operation at high rates since they have coarser position resolution, but finer time resolution than DTs and CSCs. In the barrel region, four DT stations are interleaved with the iron return yoke of the magnet, each DT is also coupled with an RPC. In each of the endcaps, the CSCs and RPCs are arranged in 4 disks perpendicular to the beam. The schematic layout of the muon chamber system is shown in Fig. 4.11.

Fig. 4.12 shows the performance of the DT in terms of spatial resolution for reconstructed hits for DT ϕ and θ superlayers. Each superlayer comprises four staggered layers of parallel drift cells. The spatial resolution is defined as the width of the distribution of residuals between the reconstructed and expected hit positions obtained from segment fit. The mean spatial resolution in each CSC station and ring is shown in Table 4.1.

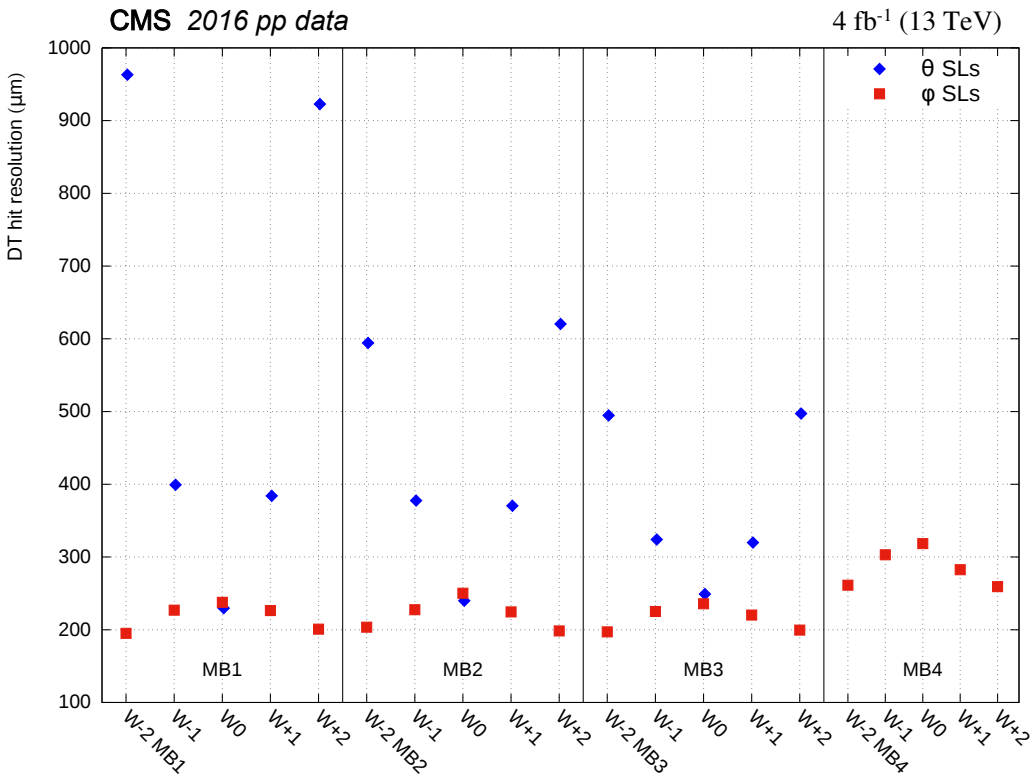


Figure 4.12: The spatial resolution for DT hits in ϕ superlayers (squares) and θ superlayers (diamonds) [52].

4.6 The trigger system

The LHC collides 40 million bunch crossings every second at the LHC. Processing and keeping all of those collision events in CMS is impractical due to limitations in bandwidth and processing capability. The trigger system is used to make real-time decisions to keep or discard events under strict latency constraints. The CMS trigger system comprises 2 stages: the Level-1 (L1) trigger and the high-level trigger (HLT). Out of 1 billion pp collisions per second at the LHC, the L1 trigger selects approximately 100 thousand events. These events are then passed to the HLT, where it does further real-time processing on CPU and only keeps $\sim$ 1000 events per second, which are saved to disk for offline processing.

The L1 trigger

The L1 trigger operates on custom ASIC chips and FPGA cards. With signal information received from the detector for an event, the L1 trigger performs basic reconstruction of physics objects and makes a trigger decision with a typical latency under 4 μs. Constrained by the CMS readout electronics limits, the output rate of

Table 4.1: The transverse spatial resolution per CSC station [52].

Station/ring	Spatial resolution (μm)		
	Run 1	Run 2	
	2012	2015	2016
ME1/1a	66	48	45
ME1/1b	57	54	52
ME1/2	93	93	90
ME1/3	108	110	105
ME2/1	132	130	125
ME2/2	140	142	134
ME3/1	125	125	120
ME3/2	142	143	135
ME4/1	127	128	123
ME4/2	147	143	134

the L1 trigger must not exceed 100 kHz. Fig. 4.13 shows a diagram of the decision workflow of the L1 trigger with signal inputs from the detector. There are two main components of the L1 trigger system: calorimeter trigger and muon trigger.

The calorimeter trigger consists of two layers, namely Layer-1 and Layer-2. Layer-1 receives, calibrates, and sorts the local energy deposits, which are called "trigger primitives" (TPG), in the HCAL and ECAL. Layer-2 uses these calibrated trigger primitives to reconstruct and calibrate physics objects, such as electrons, tau leptons, jets, etc. During Run 2, the L1 trigger has been upgraded from conventional trigger to time-multiplexed trigger [53–55]. In a conventional trigger, regional segmentation information is kept separated in the processing stage in synchronization, requiring sharing links to make decision based on multiple segments, such as a reconstructed jet spanning multiple calorimeter towers. The limitations on the number of sharing links and link speed impose boundary constraints on reconstructed physics objects. The time-multiplexed trigger is designed to overcome this limitation. In this new design, the TPG data are time-multiplexed, *i.e.*, multiple TPG data sources are sent to a single destination (node) for assembly and event processing, so that each node has access to the whole event information. The node, run by an FPGA card, takes time equivalent to 10 bunch crossings to process the data. Therefore there are 10 FPGA nodes running in a round robin fashion to handle every bunch crossing. This design also allows for the flexibility to add nodes as required by more complex trigger algorithms. The processed event information is then re-formatted

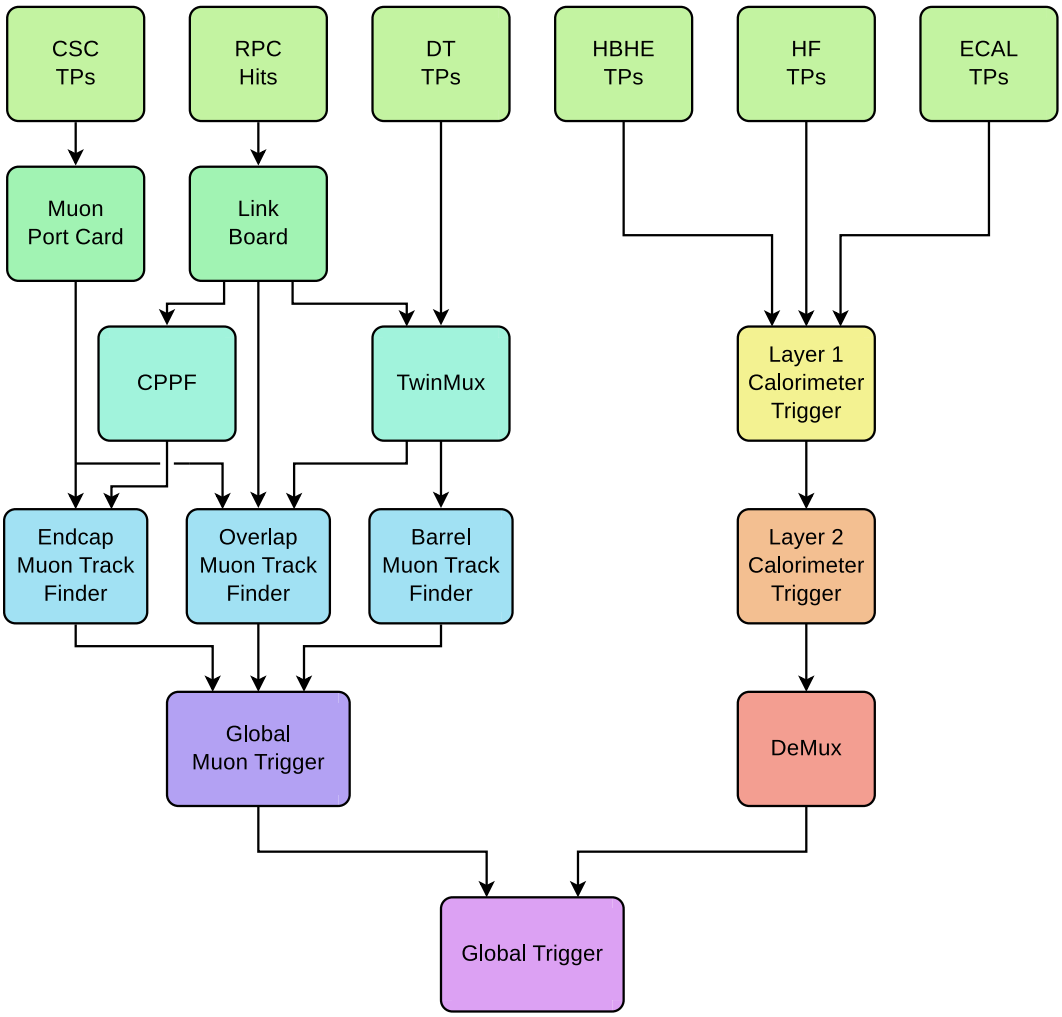


Figure 4.13: Diagram of the L1 trigger system during Run 2 [53].

by a demultiplexer (DeMux) board for the global trigger (μGT) processing. The architecture of the time-multiplexed calorimeter trigger system is shown in Fig. 4.14.

The muon trigger system consists of three muon track finders (MTF), which reconstruct muons in the barrel (BMTF), overlap (OMTF), and endcap (EMTF) regions of the muon system. A global muon trigger (μGMT) is included for the final muon selection. The BMTF uses information in the barrel region ($|\eta| < 0.83$) from DT and RPC chambers, the OMTF uses information from all three muon subsystems in the overlap region between barrel and endcap ($0.83 < |\eta| < 1.24$), and the EMTF uses information in the endcap ($1.24 < |\eta| < 2.4$) from CSC and RPC chambers. The BMTF uses look-up tables (LUTs) for track finding to assign p_{T}, ϕ, and η of a track from the bending angle and the quality of an inner station's superprimitives (*i.e.,* the combination of trigger primitives from DT and RPC). The OMTF collects

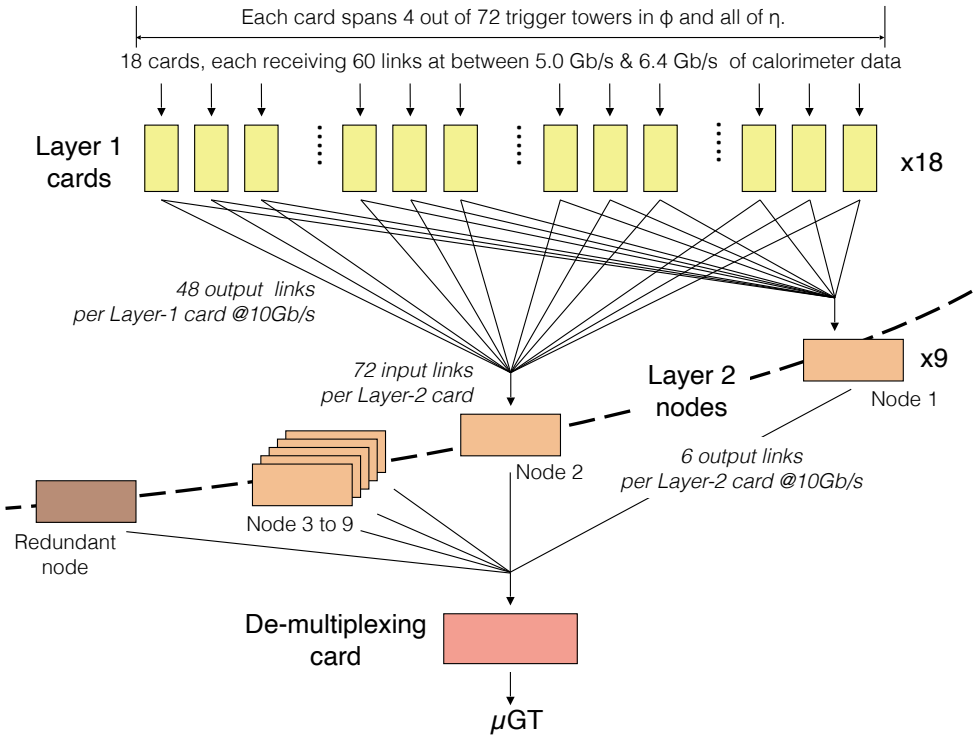


Figure 4.14: The time-multiplexed architecture of the upgraded calorimeter trigger in L1 trigger [53].

up to four reference hits from all three muon subsystems, favoring hits from inner layers with better resolutions, and associates these hits with patterns generated from simulated events. There are 26 patterns for each muon change, corresponding to different p_{T} bins ranging from 2 to 140 GeV, with probability density functions of hits spreading in ϕ in each layer. The best matched patterns, along with the reference hits, are sent to the internal muon sorter to remove possible duplicates. Three best muon candidates per board are then sent to the μGMT, resulting in a maximum of 36 muon candidates. The EMTF builds tracks using a similar pattern recognition algorithm. Additionally, a boosted decision tree (BDT) is also used to calculate the track p_{T} based on the bending angles in ϕ and θ of the muon track. The BDT is trained with Monte Carlo simulation of single-muon events, and its output values are stored in a LUT loaded in a memory module for fast look-up. Up to 108 muon candidates are sent from the three muon track finders to the μGMT. The μGMT then sorts the muons and removes duplicates, sending up to 8 muons to the μGT. More detailed information of the muon track finders can be found in Ref. [53].

The HLT

After events are selected by the L1 trigger, they are sent to the HLT on a single processor farm at LHC Point 5 for further selection with finer grained reconstruction and reducing the event rate to $\sim$ 1000 Hz. There are a few hundred trigger paths, each selecting for a particle physics signature, running in parallel for each physics event. An event is accepted if it is selected by any path. Constrained by the total CPU time needed to process an event, the HLT reconstruction and event selection are optimized by rejecting events as early as possible along the paths. For example, a path searching for an electron requires the reconstruction of an ECAL cluster, the matching of pixel hits, and the subsequent reconstruction of a full charged particle track in the tracker. Events without ECAL clusters are rejected immediately in this path without considering further information from pixel and tracker. Following this partial event reconstruction strategy, the majority of CPU budget can be spent on the most expensive reconstruction tasks, mainly track and vertex reconstruction, at the end of the sequences.

For some measurements and calibration studies that do not require the full statistics of data, the corresponding HLT paths are *prescaled*. A prescaled trigger with a prescale factor N only runs on one in every N events entering the HLT. For some trigger paths where the selection would result in prohibitively high event rate, such as dijet trigger, they are also prescaled to ensure proper resource allocation.

For electron and photon, the HLT selection is done in 3 steps. The first step uses the calorimeter information alone. The second step requires hits in the pixel detectors. If the hits in the pixel detector match the energy in the ECAL, an electron candidate is found, otherwise it is considered a photon candidate given the energy is above a certain threshold. The third step, which is required for an electron candidate, reconstructs the full track using information from the tracker with seeds from pixel hits.

The HLT selects muon in 2 steps. The first step involves muon reconstruction from the muon chamber information, which confirms the L1 decision and refines the p_{T} measurement with more precise information. The second step extends the muon trajectory to include hits in the silicon tracker system, which further improves the p_{T} measurement. After each step, isolation requirement is applied to the muon candidates, since the integrated rate of muons at LHC is dominated by muons from b, c, K, and π decays, which are generally accompanied by other nearby particles [56].

To reconstruct and identify jet objects, the HLT uses an iterative clustering algorithm named anti-k_T [57] with cone size parameter $R = 0.5$ [58]. The inputs for the jet clustering algorithm can be either the calorimeter towers (CaloJet), allowing for fast reconstruction, or the reconstructed Particle Flow objects (PFJet), which require significantly more CPU consumption. Generally, single PFJet paths would require a matching between CaloJet and PFJet and have preselection based on CaloJet objects.

Many important physics processes require the identification of b jets, such as the identification of the Higgs decaying into 2 b jets, where the long lifetime of b quark results in the characteristic secondary vertex of the jets. At the HLT, b jets are identified with a b jet tagging algorithm, which is Combined Secondary Vertex (CSVv2) before 2018 and DeepCSV in 2018 [59].

HLT paths sharing a similar purpose and having the same output event content are grouped into *data streams*. For example, streams for hardware calibration only save the relevant parts of the raw data for further processing, such as ECAL data for ECAL calibration, while physics streams save the full raw event information for offline analysis. Streams are further divided into different *primary datasets*, which have the same event format and are processed the same way in offline processing. The primary datasets are mainly defined based on the particle candidate reconstructed in the event final state by the HLT, such as `SingleMu`, `DoubleElectron`, `SinglePhoton`, *etc.*

4.7 The globally distributed data processing system

Data selected by the HLT farm are sent from LHC Point 5 directly to CERN computing center, also known as Tier-0 (T0). T0 performs the first pass reconstruction on RAW data, namely *prompt* reconstruction, which produces reconstructed data (RECO), Analysis Object Data (AOD), and reduced AOD data (MiniAOD). From T0, data are distributed to the next stage—Tier-1 (T1) resources. There are seven T1 sites globally, which are large regional computing centers in CMS collaborating countries: Fermilab (United States), IN2P3 (France), PIC (Spain), ASGC (Taiwan), CLRC (United Kingdom), GridKa (Germany), and INFN (Italy). Two copies of RAW data are saved, one at CERN T0, another at a T1. The T1 sites are generally used for large-scale, centrally organized computing activities, providing data to and receiving data from Tier-2 (T2) sites in local regions. T2 sites are local computing centers, such as universities, but with substantial CPU resources, mainly serving local communities. This distributed hierarchical model follows MONARC

recommendations [60], running on GRID distribute infrastructures, also known as the Worldwide LHC Computing GRID [61].

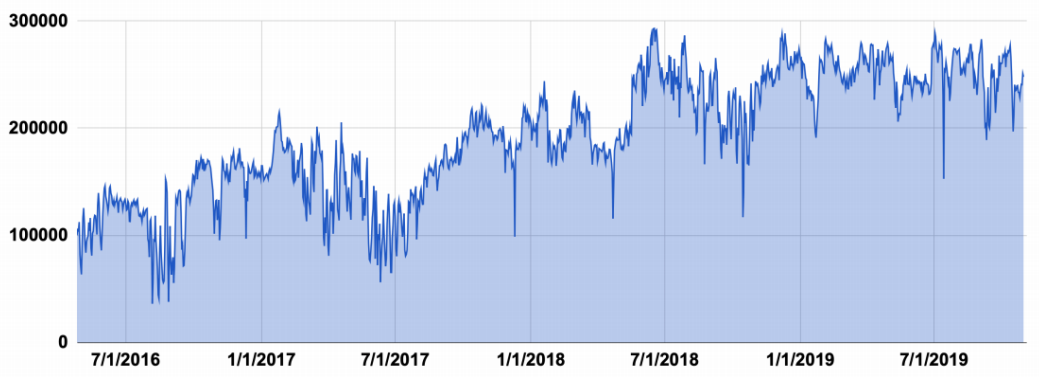


Figure 4.15: Global HTCondor pool size in number of CPU cores averaged daily during Run 2 in CMS [62].

Centralized workflows such as data reconstruction and the generation of simulated events over the computing grid are managed by the CMS workload management (WM) system. The WM system, based on the concept of WMAgent framework [63], handles tasks such as workload splitting into jobs, job assignment to appropriate computing sites, job priority adjustment, retrial of unsuccessful jobs, merging output files, and log collection [64]. The CMS Submission Infrastructure [62] performs resource allocation by employing GlideinWMS [65], a tool built on top of the HTCondor batch system [66–68] that matches jobs to resources. Over the Run 2 data-taking period, more resources have been added to the global HTCondor pool, including the HLT farm for offline processing when not in data-taking mode, as well as cloud and HPC resources [69], resulting in a total of 250 000 cores running routinely, with a peak of 300 000 cores, as shown in Fig. 4.15.

The CMS infrastructure model requires input data to be locally present at the processing site with the assumption of poor connectivity between sites. As a result, each dataset may have multiple copies scattered around different sites. While this ensures reliable local read for processing sites, the dataset multiplicity imposes significant stress on storage. Some Monte Carlo simulation workflows require reading pre-made pileup datasets remotely, as these pileup datasets are too gigantic to fit in any local storage. The remote data access service, namely AAA (Any Data, Anytime, Anywhere) [70], is a CMS-customized implementation based on the XRootD framework [71] that allows for analyzing CMS data remotely without downloading to local storages. Recently, there are R&D programs experimenting

with new storage models, such as DataLake [72], which allows for centralization of storage into fewer and bigger sites and enables remote access via streaming and caching, due to general improvements of connectivity between sites, reducing the stress on storage capacity.

Table 4.2: CMS data formats for physics analyses [73].

Data format	Mainly used for the period	Size (kB/event)
RECO	2010–2011	3000
AOD	2011–2015	400
MiniAOD	2016–2019	50
NanoAOD	2019-	1

Table 4.2 summarizes the evolution of main input for physics analyses over the years. The drastic reduction in data size comes from the increased understanding of the accelerator conditions, detector calibrations, and analysis patterns. For Monte Carlo simulation events, instead of RAW data tier, there are GEN, SIM, and DIGI data tiers: GEN contains information about generated Monte Carlo event, SIM contains energy depositions of the generated particles in detector, also known as *sim hits*, DIGI contains the detector responses converted from sim hits. The content of DIGI data tier is basically the same as the RAW output of the detector.

Part III

Physics of the Higgs at the LHC

Chapter 5

SEARCH FOR NONRESONANT HIGGS BOSON PAIR PRODUCTION IN FINAL STATES WITH TWO BOTTOM QUARKS AND TWO PHOTONS

The discovery of the Higgs boson by the CMS and ATLAS experiments in 2012 opened up a new portal to deepen our understanding of the electroweak symmetry breaking mechanism. While this final missing piece of the Standard Model has been found, many questions still follow with numerous implications on physics beyond the Standard Model (BSM), the origin of the matter-antimatter imbalance in early universe, and the meta-stability of the universe. Understanding of the structure of the Higgs potential could provide the answers to some of these questions; and the Higgs trilinear self-coupling, which could be measured through the production of a pair of Higgs boson, is a direct probe to the Higgs potential structure.

The production of di-Higgs is an extremely rare process, with an expected total cross section at 13 TeV of less than 35 fb [74–76], roughly 3 orders of magnitude less than that of single Higgs production. Even though we do not expect to see the Standard Model di-Higgs production during the LHC Run 2, many BSM theories suggest the presence of heavy additional particles that couple to the Higgs boson pair, modifying the Higgs boson's couplings and could end up significantly amplifying the cross section.

Using an effective field theory (EFT) framework, these BSM effects can be described to leading approximation with the following Lagrangian [77]:

$$\begin{aligned}\mathcal{L}_{\mathrm{HH}} = \kappa_\lambda \lambda^{\mathrm{SM}}_{\mathrm{HHH}} v \mathrm{H}^3 - \frac{m_\mathrm{t}}{v}\left(\kappa_t \mathrm{H} + \frac{c_2}{v}\mathrm{H}^2\right)\left(\bar{\mathrm{t}}_\mathrm{L}\mathrm{t}_\mathrm{R} + \mathrm{h.c.}\right) \\ + \frac{1}{4}\frac{\alpha_s}{3\pi v}\left(c_g \mathrm{H} - \frac{c_{2g}}{2v}\mathrm{H}^2\right)\mathrm{G}^{\mu\nu}\mathrm{G}_{\mu\nu},\end{aligned} \tag{5.1}$$

where $v \approx 246$ GeV is the Higgs vacuum expectation value and $m_\mathrm{t} \approx 173$ GeV is the top quark mass. The BSM effects are parametrized with 5 parameters: $\kappa_\lambda = \lambda_{\mathrm{HHH}}/\lambda^{\mathrm{SM}}_{\mathrm{HHH}}$ and $\kappa_t = y_t/y_t^{\mathrm{SM}}$ are the modifiers of SM values of the Higgs trilinear self-coupling, $\lambda^{\mathrm{SM}}_{\mathrm{HHH}} \equiv m_\mathrm{H}^2/(2v^2) \approx 0.129$, and the Yukawa coupling, $y_t^{\mathrm{SM}} \equiv m_\mathrm{t}/v \approx 0.7$, respectively, and three additional coupling parameters not present in the SM. These include the contact interactions between two Higgs bosons

and two gluons (c_g), between one Higgs boson and two gluons (c_{2g}), and between two Higgs bosons and two top quarks (c_2), which are illustrated in the bottom diagrams in Fig. 5.1. t_L and t_R are the top quark fields with left and right chiralities, respectively. H denotes the Higgs boson field, $G^{\mu\nu}$ is the gluon field strength tensor, and h.c. denotes the Hermitian conjugate.

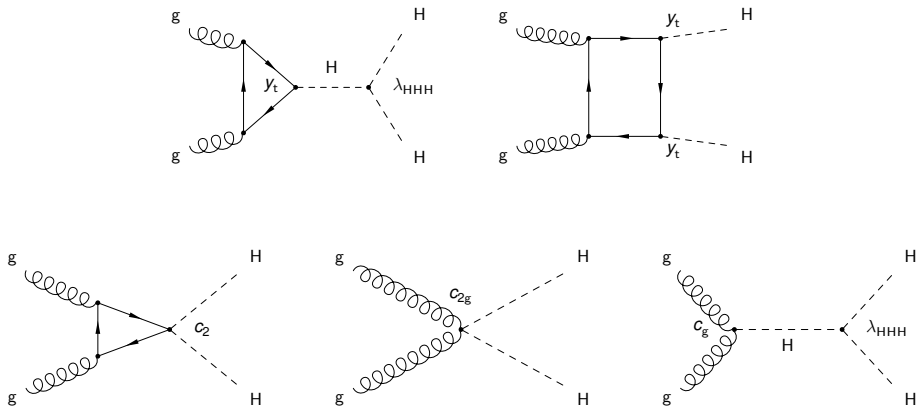


Figure 5.1: Feynman diagrams for ggHH processes. Top: Contributions from Standard Model processes at leading order, referred to as box and triangle diagrams, respectively. Bottom: BSM processes that describe contact interactions of two Higgs bosons with two top quark (left), between the Higgs boson and gluons (middle and right).

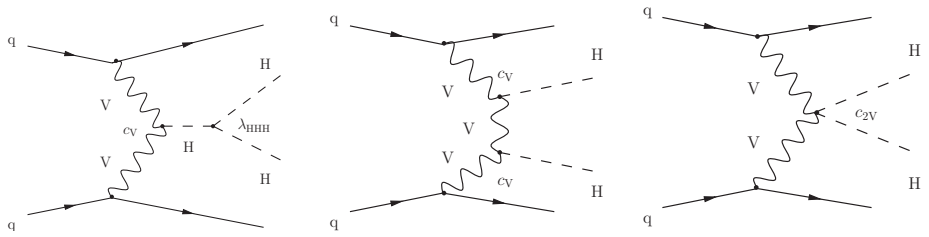


Figure 5.2: Feynman diagrams that contribute to the production of SM Higgs boson pairs via VBF at LO. The left diagram involves a HHH vertex (λ_{HHH}) and a HVV vertex (c_V), the middle diagram involves two HVV vertices (c_V), and the right diagram involves a HHVV vertex (c_{2V}).

To avoid a huge number of BSM samples to be generated in this analysis, we use twelve possible combinations of the five parameters above in such a way that they are representative of the full phase space, following the recommendation from [77]. Further details on these parameters are described in Sec. 5.1.

We also investigate vector-boson fusion (VBF) HH production mode, which gives access to λ_{HHH}, as well as to the coupling between two vector bosons and the Higgs boson (HVV) and the coupling between a pair of Higgs bosons and a pair of vector bosons (HHVV), as illustrated by the Feynman diagrams in Fig. 5.2. While λ_{HHH} is mainly constrained by measurements of HH production via ggF, and the HVV coupling modifier (c_{V}) is constrained by the measurements of vector boson associated production of a single Higgs boson and the decay of the Higgs boson to a vector boson pair [78], the HHVV coupling modifier ($c_{2\mathrm{V}}$) is only directly accessible via VBF HH production.

Previous searches for nonresonant production of a Higgs boson pair via ggF were performed by both the ATLAS and CMS Collaborations using the LHC data collected at $\sqrt{s}$ = 8 and 13 TeV [79–89]. Searches in the $\mathrm{b\bar{b}}\gamma\gamma$ channel performed by the ATLAS [79] and CMS [89] Collaborations using up to 36.1 fb^{-1} of pp collision data at $\sqrt{s}$ = 13 TeV set upper limits at 95% confidence level (CL) on the product of the HH cross section and the branching fraction into $\mathrm{b\bar{b}}\gamma\gamma$. The observed upper limits are found to be 24 (30 expected) and 26 (20 expected) times the SM expectation for the ATLAS and CMS searches, respectively. Statistical combinations of search results in various decay channels were also performed by the two experiments [80, 90]. Recently, the first search for HH production via VBF was carried out by the ATLAS Collaboration in the $\mathrm{b\bar{b}b\bar{b}}$ channel [91].

This chapter describes the search for double Higgs production in the $\mathrm{HH} \rightarrow \mathrm{b\bar{b}}\gamma\gamma$ decay channel, where one Higgs bosons decays into a pair of photons and the other Higgs boson decays into a bottom quark-antiquark pair, using a data sample of 137 fb^{-1} collected by the CMS experiment from 2016 to 2018. The high branching ratio of the $\mathrm{H} \rightarrow \mathrm{b\bar{b}}$ decay, along with the high signal over background ratio of the $\mathrm{H} \rightarrow \gamma\gamma$ decay, provides this channel with an expected high sensitivity. The $\mathrm{b\bar{b}}\gamma\gamma$ final state has a combined branching fraction of $2.63 \pm 0.06 \times 10^{-3}$ for a Higgs boson mass of 125 GeV [92].

The chapter is organized as follows: Sec. 5.1 describes the data and simulated samples used in this search. In Sec. 5.2, we describe the algorithms used for reconstruction of physics objects and the preselection criteria of the analysis. Sec. 5.3 presents the analysis strategy. Sec. 5.4 describes the resonant background reduction technique, namely the $\mathrm{t\bar{t}H}$ tagger. Nonresonant background reduction procedures are shown in Sec. 5.5. Sec. 5.6 describes the event categorization. Signal and background modeling for data-driven background estimation are described in Sec. 5.18

Table 5.1: Parameter values of the 12 BSM benchmarks along with the Standard Model point.

Benchmark	κ_λ	κ_t	c_2	c_g	c_{2g}
1	7.5	1.0	-1.0	0.0	0.0
2	1.0	1.0	0.5	-0.8	0.6
3	1.0	1.0	-1.5	0.0	-0.8
4	-3.5	1.5	-3.0	0.0	0.0
5	1.0	1.0	0.0	0.8	-1.0
6	2.4	1.0	0.0	0.2	-0.2
7	5.0	1.0	0.0	0.2	-0.2
8	15.0	1.0	0.0	-1.0	1.0
9	1.0	1.0	1.0	-0.6	0.6
10	10.0	1.5	-1.0	0.0	0.0
11	2.4	1.0	0.0	1.0	-1.0
12	15.0	1.0	1.0	0.0	0.0
SM	1.0	1.0	0.0	0.0	0.0

and 5.9, respectively. Sec. 5.10 describes the systematic uncertainties in this analysis. Results are shown in Sec. 5.11. Finally, Sec. 5.12 provides a summary of this search.

5.1 Event samples

This search is performed on the full Run 2 data collected by the CMS experiment over the data-taking period of 3 years from 2016 to 2018, corresponding to an integrated luminosity of 137 fb^{-1}. Events are selected with double-photon triggers that require the photons' transverse momenta to be above 30 GeV for the leading photon and 18 (22) GeV for the second photon for the data collected during 2016 (2017 and 2018).

To produce signal samples for the gluon-fusion (ggF) HH process at NLO [93–97], POWHEG 2.0 is used to generate samples with different values of κ_λ that include the full top quark mass dependence [98]. MADGRAPH5_aMC@NLO [99–101] is used to generate samples for BSM benchmark hypotheses and the vector-boson fusion (VBF) HH processes at LO. The parameter values of the 12 BSM benchmarks are shown in Table 5.1.

Dominant background processes in this analysis are irreducible prompt diphoton production ($\gamma\gamma$+jets), which is modeled with SHERPA v.2.2.1 at LO [102], and the reducible background from γ+jets events, where the jets are mistagged as isolated

photons and b jets. This reducible background process is modeled with PYTHIA 8.212 at LO [103].

Resonant background processes, which contain single Higgs boson production, where the Higgs decays into two photons, are simulated at NLO with POWHEG 2.0 [93, 104–106] for ggF H and VBF H, and with MADGRAPH5_aMC@NLO for Higgs production associated with a top quark-antiquark pair ($t\bar{t}H$), vector boson associated production (VH), and production associated with a single top quark. The cross sections and decay branching fractions are taken from Ref. [92]. The contribution from the other single H decay modes is negligible.

Parton showering and fragmentation are simulated via PYTHIA interface with the standard p_T-ordered parton shower (PS) scheme for all simulated samples. Underlying events are modeled with CUETP8M1 tune for 2016 and the CP5 tune for 2017-2018 [107, 108]. Parton distribution function sets are generated from NNPDF3.0 [109] NLO for 2016 and NNPDF3.1 [110] NNLO for 2017 and 2018. The CMS detector responses are simulated with GEANT4 toolkit [111].

The simulated VBF HH samples are further simulated with initial-state radiation and final-state radiation with the PYTHIA dipole shower scheme to take into account the structure of the color flow between in-coming and outgoing quark lines. The predictions of the simulation are in good agreement with the NNLO QCD calculation, as reported in [112].

5.2 Physics object reconstruction

All physics objects are reconstructed with the particle-flow (PF) algorithm [113]. This search uses photons, b jets, and VBF jets.

Photon candidates are identified from clusters of reconstructed hits in the ECAL crystals. A photon identification technique (photon ID) based on multivariate analysis with boosted decision tree is developed to separate photons from jets. Further details of this photon ID can be found in [114]. Photons falling in the gap between the barrel and ecap region of the ECAL ($1.442 < |\eta| < 1.566$) are excluded in this search because the performance of the photon reconstruction in this region is not optimal. Additional requirements on the photon shower shapes, energy, and isolation are imposed to improve the selection efficiency. Table 5.2 summarizes the selection criteria on the photon candidates in this search.

Once two photon candidates are found, we compute their invariant mass and require $100 < m_{\gamma\gamma} < 180\,\text{GeV}$, $p_T^{\gamma_1}/m_{\gamma\gamma} > 1/3$, and $p_T^{\gamma_2}/m_{\gamma\gamma} > 1/4$. When there are more

Table 5.2: Photon selection criteria on photon candidates.

Requirements	Leading Photon	Subleading Photon
p_{T}	> 30 GeV	> 20 GeV
$\lvert\eta\rvert$	$0 < \lvert\eta\rvert < 1.442$ or $1.566 < \lvert\eta\rvert < 2.5$	
R_9	> 0.8	
Charge isolation	< 20	
H/E	< 0.08	
BDT score	> -0.9	

than two photon candidates, the photon pair with the highest transverse momentum $p_{\mathrm{T}}^{\gamma\gamma}$ is selected to construct the Higgs boson candidate.

Jet candidates are selected from PF candidate clusters undergoing the anti-k_T algorithm with a distance parameter of 0.4. We require the jet candidates to have $p_{\mathrm{T}}>$ 25 GeV and $|\eta| < 2.4(2.5)$ for 2016 (2017-2018) (the new CMS pixel detector installed during the Phase-1 upgrade allows for the extended η range for jet candidates in 2017 and 2018). The candidates are also required to be outside of the cone radii 0.4 centered on the identified photons.

Once jet candidates are identified, a deep neural network (DNN)-based algorithm called DeepJet [115, 116] is deployed to identify jets from the hadronization of b quarks. Furthermore, a b jet energy regression algorithm [117] is used to correct the energy and improve the energy resolution of the b jets.

If an event has more than 2 jets, we select the jet pair with the highest b tagging scores to construct the Higgs boson. The invariant mass of the dijet system is required to be $70 < m_{\mathrm{jj}} < 190$ GeV. Another multivariate regressor is developed to improve the mass resolution of the reconstructed Higgs boson that decays into 2 b jets.

Once the jet pair and photon pair are identified, an event is selected. Table 5.3 summarizes the baseline selection criteria for photons and jets in each event in this analysis. The expected number of SM HH signal events after these selections for each data-taking year is listed in Table 5.4.

5.3 Analysis strategy

This search relies on the peaks in the invariant mass distributions of the dijet and diphoton system, m_{jj} and $m_{\gamma\gamma}$, respectively, around the value of the Higgs boson

Table 5.3: Summary of the baseline selection criteria for $HH \to b\bar{b}\gamma\gamma$ events.

Photons		Jets	
$p_T^{\gamma 1}$	$> m_{\gamma\gamma}/3$	p_T [GeV]	$> 25.$
$p_T^{\gamma 2}$	$> m_{\gamma\gamma}/4$	$\Delta R_{\gamma j}$	> 0.4
$\lvert\eta\rvert$	< 2.5	$\lvert\eta\rvert$	< 2.4
$m_{\gamma\gamma}$ [GeV]	$\in [100, 180]$	m_{jj} [GeV]	$\in [70, 190]$
Select pair of highest $p_T^{\gamma\gamma}$		Select pair of highest DeepJet score	

Table 5.4: Expected number of SM HH signal events after the baseline preselection for each data-taking year (scaled according to luminosity) and full Run II.

Year	Expected SM HH events
2016	1.14
2017	1.26
2018	1.84
Run 2	4.24

mass (125 GeV). Therefore, we extract the number of signal and background events using a simultaneous parametric fit on $m_{\gamma\gamma}$ and m_{jj}.

There are two types of backgrounds in this analysis: non-resonant background and resonant background. For the non-resonant background, mainly $\gamma\gamma$+jets and γ+jets, both $m_{\gamma\gamma}$ and m_{jj} exhibit a falling spectrum. For the resonant background, where a single SM Higgs boson is produced, $m_{\gamma\gamma}$ is peaking at the Higgs boson mass. After the preselection criteria described in Sec. 5.2, the two most relevant processes are gluon-gluon fusion (ggH) and associated production with top quarks ($t\bar{t}H$), where the Higgs bosons decay into a pair of photons.

To improve the sensitivity of this search, we first reduce the resonant background with a deep-learning based classifier. Afterwards, MVA techniques are used to distinguish the ggF and VBF HH signal from the dominant nonresonant background. The outputs of the MVA classifiers are then used to define mutually exclusive analysis categories targeting VBF and ggF HH production.

The distribution of $\widetilde{M}_X$, defined as:

$$\widetilde{M}_X = m_{jj\gamma\gamma} - (m_{jj} - m_H) - (m_{\gamma\gamma} - m_H) \tag{5.2}$$

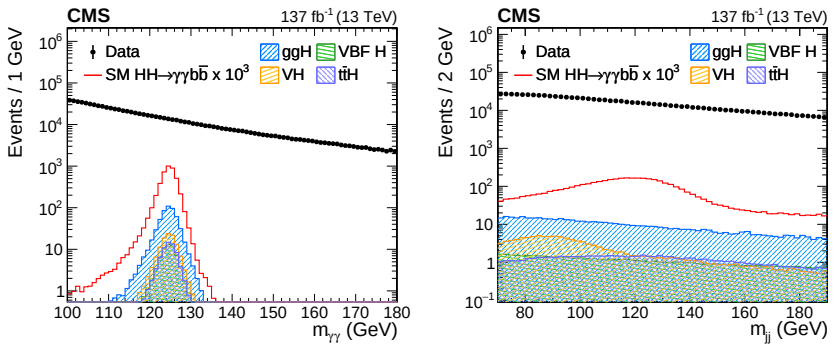


Figure 5.3: The invariant mass distributions of the reconstructed Higgs boson candidates $m_{\gamma\gamma}$ (left) and m_{jj} (right) in data and simulated events. Data, dominated by the $\gamma\gamma$ and γ+jets backgrounds, are compared to the SM ggF HH signal samples and single H samples ($\mathrm{t\bar{t}H}$, ggH, VBF H , VH) after imposing the selection criteria described in Sec. 5.2. The error bars on the data points indicate statistical uncertainties. The HH signal has been scaled by a factor of 10^3 for display purposes.

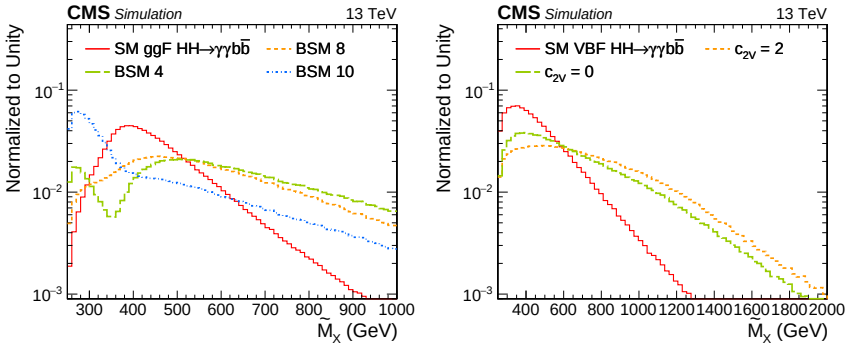


Figure 5.4: Distributions of $\widetilde{M}_{\mathrm{X}}$. The SM ggF HH signal is compared with several BSM hypotheses listed in Table 5.1 (left), and the SM VBF HH signal is compared with two different anomalous values of $c_{2\mathrm{V}}$ (right). All distributions are normalized to unity.

is particularly sensitive to different BSM parameters defined in the introduction of this chapter. The $\widetilde{M}_{\mathrm{X}}$ distribution is less dependent on the dijet and diphoton energy resolutions than $m_{\mathrm{jj}\gamma\gamma}$ if the dijet and diphoton pairs originate from a Higgs boson decay [118]. In Fig. 5.4, the distribution of $\widetilde{M}_{\mathrm{X}}$ is shown for several BSM benchmark hypotheses affecting ggF HH production (described in Table 5.1) and for different values of $c_{2\mathrm{V}}$ affecting the VBF HH production mode. The SM HH process exhibits a broad structure in $\widetilde{M}_{\mathrm{X}}$, induced by the interference between different processes contributing to HH production and shaped by the analysis selection. The signals with $c_{2\mathrm{V}} = 0$ and $c_{2\mathrm{V}} = 2$ have a much harder spectrum than the SM VBF HH signal, as shown on the right of Fig. 5.4.

5.4 Resonant background reduction

Single Higgs boson production is an important resonant background in the $b\bar{b}\gamma\gamma$ final state, with $t\bar{t}H$ production being dominant in high purity signal regions. To reduce $t\bar{t}H$ background contamination, a dedicated classifier is developed. The classifier is trained on a mixture of SM HH events and events generated for the 12 BSM benchmark hypotheses (described in Table 5.1) as signal, and $t\bar{t}H$ events as background.

Event classification using a combination of high-level information from event kinematics and low-level information from individual particles has been demonstrated as the most performant among popular neural-network-based classification techniques [119]. Thus, for the $t\bar{t}H$ discriminator, we use physics objects reconstructed and calibrated, including electrons and muons, as the low-level information, and kinematic variables as the high-level information to train the networks.

The kinematic variables used in training can be described in three groups: angular variables, variables to discriminate semi-leptonic W bosons produced in the top quark decay, and variables to discriminate hadronic W boson decays.

Angular variables

The angular separation ($\Delta R(\gamma, \text{jet})$) between the photon and b-tagged jet in the event is used since the photon and b jets are expected to be well separated for signal events. The angle $\cos(\theta_{CS})$ in the Collins-Sopper frame between the reconstructed dijet and diphoton candidates and angle $\cos(\theta_{b\bar{b}})$ between the two b jets are shown as powerful discriminants between signals and backgrounds, as shown in Fig. 5.5, thus are used as training variables.

Variables to reject events with a leptonic-decay W boson

Events with a leptonic-decay W boson are expected to have significant p_T^{miss} due to the presence of neutrinos. To maximize the probability of having the neutrino coming form W boson decay, the azimuthal angle separations between the p_T^{miss} and the two b jets ($\Delta\phi(p_T^{miss}, \text{jet}_1)$, $\Delta\phi(p_T^{miss}, \text{jet}_2)$) are used as training variables. Leptonic-decay W boson event also could have leptons reconstructed in the final state. Thus the four momenta of reconstructed leptons with $p_T > 10$ GeV are also included in the training. The distributions of most of these variables are shown in Fig. 5.6.

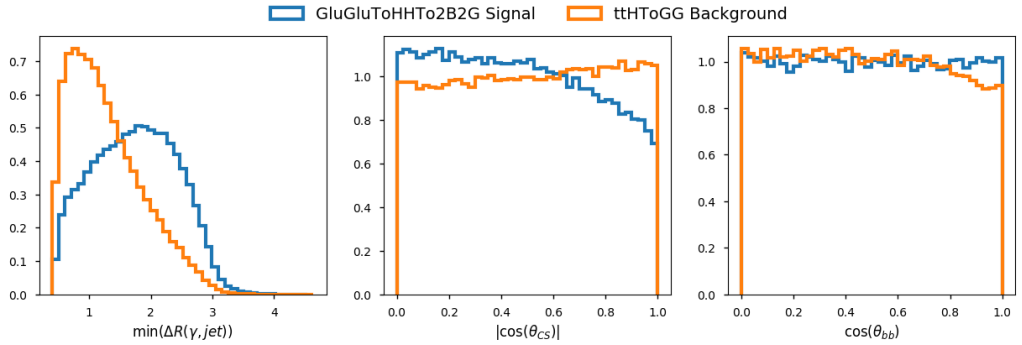


Figure 5.5: Angular variables used in the training, from left to right: $\Delta R(\gamma, \text{jet})$, $\cos(\theta_{CS})$, and $\cos(\theta_{b\bar{b}})$. These variables are used for the training of the $t\bar{t}H$ discriminant.

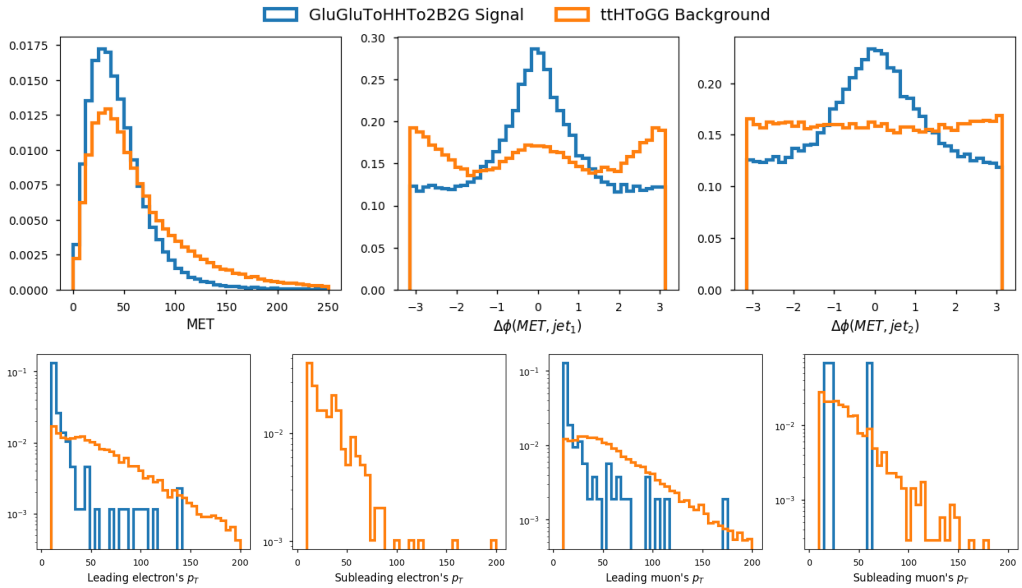


Figure 5.6: Major variables used in the training to reject events with a leptonic-decay W boson, from left to right, top to bottom: p_T^{miss}, $\Delta\phi(p_T^{miss}, \text{jet}_1)$, $\Delta\phi(p_T^{miss}, \text{jet}_2)$, and the transverse momentum of the leading and subleading electrons and muons. In signal events, there is no subleading electron, which explains its absence in the subleading electron's p_T distribution plot.

Variables to reject events with a hadronic-decay W boson

To reduce $t\bar{t}H$ events with hadronic-decay W bosons, we use χ^2_t, defined as:

$$\chi^2_t = \left(\frac{m_W - m_{jj}}{0.1 \times m_W}\right)^2 + \left(\frac{m_t - m_{bjj}}{0.1 \times m_t}\right)^2, \tag{5.3}$$

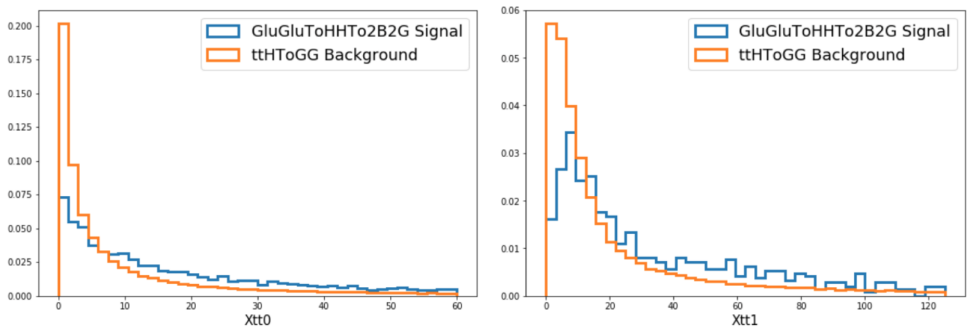


Figure 5.7: χ^2_t variables in training to reject events with a hadronic-decay W boson, for events with at least 2 additional jets (left) and 4 additional jets (right) besides the two b jets.

where m_W and m_t are the true mass values of the W boson and the top quark, taken to be 80.3 GeV and 173.5 GeV, respectively. The value of χ^2_t distributes towards zero if there is a hadronic-decay W boson from a top quark decay in the event. This variable is calculated for events with at least 2 additional jets and 4 additional jets besides the two b jets, and the distributions in these cases are shown in Fig. 5.7.

Network architecture

The $t\bar{t}H$ discriminant is implemented with a multimodal neural networks (NN) combining a feed-forward and a recurrent NNs, based on the topology-classifier architecture introduced in [119]. The momenta (p_T, η, ϕ) and identities of the physics objects (2 leading electrons, 2 leading muons, 2 b jets, diphoton) and the p_T^{miss} are fed into the recurrent layers, ordered by their transverse momentum amplitudes. The objects' momenta are normalized such that they have means of zero and standard deviation of one. Objects that do not appear in the events (such as subleading leptons) are padded with matrices of zeros, which are filtered out before the recurrent layers. The output of the recurrent layers is combined with the high-level information from event kinematics through fully-connected layers, activated with rectified linear unit functions. The final output layer, which indicates the probability of the event being or signal or background, is activated with a sigmoid function. Figure 5.8 illustrates the overall architecture of this multimodal neural network.

The network hyperparameters are optimized with the Bayesian optimization technique, where the average over 3-fold cross-validation accuracy is used as the figure of merit. The network is implemented in KERAS [120] on the TENSORFLOW platform [121].

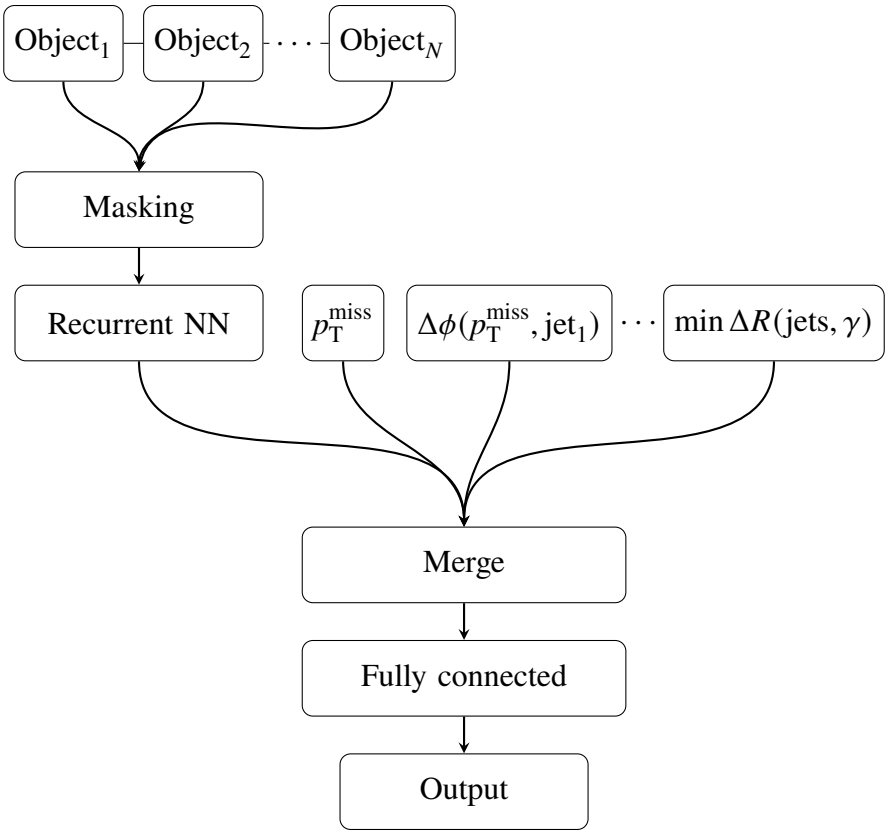


Figure 5.8: The multimodal network architecture for the $\mathrm{t\bar{t}H}$ tagger. Object$_{1..N}$ contain the kinematic and identity information of the reconstructed PF objects, ordered by their transverse momenta. The masking layer filters out the zero-padded object, *i.e.*, if an object does not exist in a given event, that object's information does not enter the network. The output of the recurrent neural network is merged with the high-level information, such as $p_{\mathrm{T}}^{\mathrm{miss}}$, $\Delta\phi(p_{\mathrm{T}}^{\mathrm{miss}}, \mathrm{jet}_1)$, *etc.*, and then goes through a fully connected block to compute the output prediction.

Performance

The output scores of the $\mathrm{t\bar{t}H}$ tagger for both $\mathrm{HH} \rightarrow \mathrm{b\bar{b}}\gamma\gamma$ signal and the $\mathrm{t\bar{t}H}$ background are shown on the left plot of Fig. 5.9. Its performance in terms of signal efficiency (true positive rate) and background contamination (false positive rate) is shown on the right plot of Fig. 5.9 and on Table 5.5.

Agreement between data and simulation

In order to get the best upper limits on the signal cross section, a cut on the $\mathrm{t\bar{t}H}$ tagger score is optimized simultaneously with the MVA categorization that will be described later in this chapter.

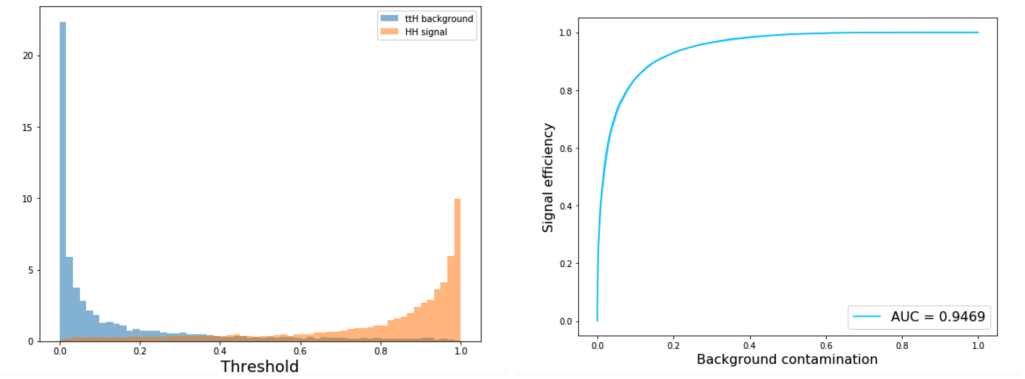


Figure 5.9: The performance of the $t\bar{t}H$ tagger. Left: the output score on $HH \to b\bar{b}\gamma\gamma$ signal and $t\bar{t}H$ background. Right: Performance in terms of signal efficiency and background contamination in a receiver operating characteristic (ROC) curve.

Table 5.5: Performance of the $t\bar{t}H$ tagger. The uncertainty in background efficiency is obtained from k-fold cross-validation.

$t\bar{t}H$ score	$HH \to b\bar{b}\gamma\gamma$ signal efficiency (%)	$t\bar{t}H$ background contamination (%)
0.1618	97.04	28.31 ± 0.77
0.2528	94.98	21.14 ± 0.36
0.3627	91.96	15.39 ± 0.11
0.7560	75.36	4.42 ± 0.32
0.8938	57.77	1.58 ± 0.06
0.9587	38.37	0.41 ± 0.01

The comparison of the $t\bar{t}H$ tagger score distributions for data and MC simulation is studied. For MC simulation, we combine all background processes, normalized to cross section times total luminosity over 2016 and 2017 (77.4 fb^{-1}). As shown in Fig. 5.10, the agreement between data and MC simulation for the $t\bar{t}H$ tagger is reasonable.

Expected improvement with the $t\bar{t}H$ tagger

We study the expected improvement on the final limit as a function of the selection on the $t\bar{t}H$ tagger score cut. As shown in Fig. 5.11, we optimize the cut on the $t\bar{t}H$ tagger score to minimize the 95% CL upper limit on the product of the HH cross section and its branching ratio to $b\bar{b}\gamma\gamma$. The best $t\bar{t}H$ score threshold is determined to improve the upper limit by 11% compared to not using the $t\bar{t}H$ tagger, *i.e.*, setting the threshold to zero.

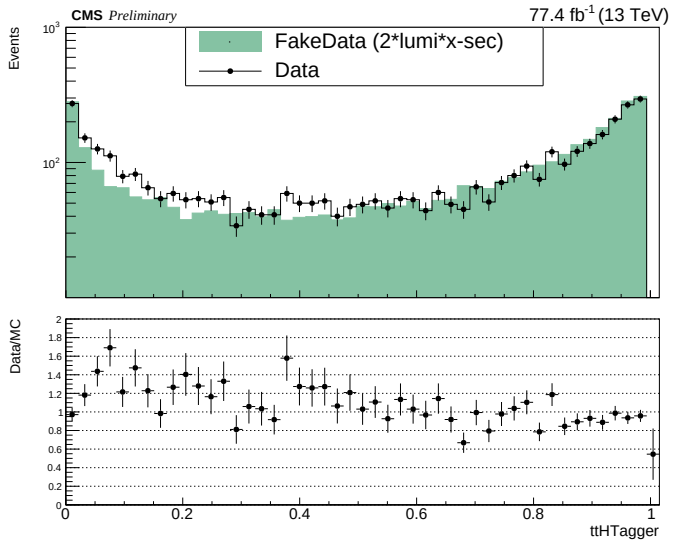


Figure 5.10: Comparison for $t\bar{t}H$ tagger score distributions between data and MC simulation normalized to cross section times total luminosity over 2016 and 2017.

5.5 Nonresonant background reduction

Background reduction in the ggF HH signal region

An MVA discriminant implemented with a boosted decision tree (BDT) is used to separate the ggF HH signal and the dominant nonresonant $\gamma\gamma$ + jets and γ + jets backgrounds. We select several discriminating observables to be used in the training. They can be classified in three groups: kinematic variables, object identification variables, and object resolution variables. The first group exploits the kinematic properties of the HH system, the second helps to separate the signal from the reducible γ + jets background, and the third takes into account the resonant nature of the $\gamma\gamma$ and $b\bar{b}$ final states for signal. The following discriminating variables were chosen:

- The H candidate kinematic variables: $p_T^{\gamma}/m_{\gamma\gamma}$, p_T^{j}/m_{jj} for leading and subleading photons and jets, where p_T^{γ} and p_T^{j} are the transverse momenta of the selected photon and jet candidates
- The HH transverse balance: $p_T^{\gamma\gamma}/m_{jj\gamma\gamma}$ and $p_T^{jj}/m_{jj\gamma\gamma}$, where $p_T^{\gamma\gamma}$ and p_T^{jj} are the transverse momenta of the diphoton and dijet candidates
- Helicity angles: $|\cos\theta_{HH}^{CS}|$, $|\cos\theta_{jj}|$, and $|\cos\theta_{\gamma\gamma}|$ where $|\cos\theta_{HH}^{CS}|$ is the Collins-Soper angle [85] between the direction of the $H \to \gamma\gamma$ candidate and

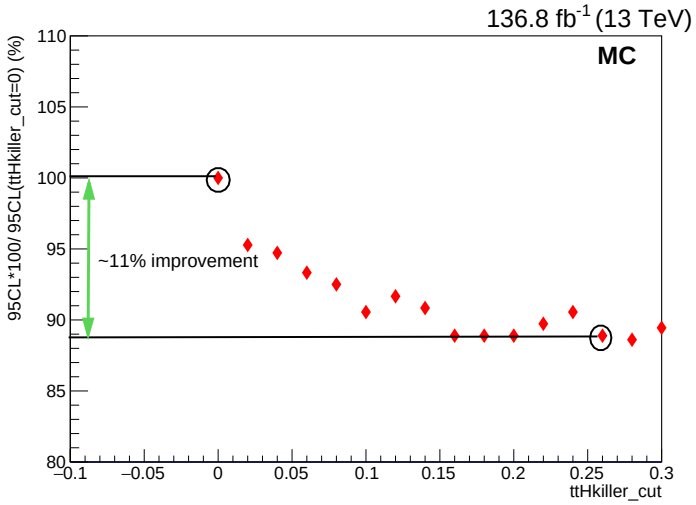


Figure 5.11: $t\bar{t}H$ score cut optimization with Monte Carlo simulation. The vertical axis indicates the percentage improvement on the 95% CL upper limit on the product of the HH cross section and its branching ratio to $b\bar{b}\gamma\gamma$. The horizontal axis corresponds to different thresholds for the $t\bar{t}H$ tagger score.

the average beam direction in the HH center-of-mass frame, while $|\cos\theta_{jj}|$, and $|\cos\theta_{\gamma\gamma}|$ are the angles between one of the Higgs boson decay products and the direction defined by the Higgs boson candidate

- Angular distance: minimum $\Delta R_{\gamma j}$ between a photon and a jet, $\Delta R^{min}_{\gamma j}$, considering all combinations between objects passing the selection criteria, and $\Delta R_{\gamma j}$ between the other photon-jet pair not used in the $\Delta R^{min}_{\gamma j}$ calculation
- b tagging: the b tagging score for each jet in the dijet candidate
- photon ID: photon identification variables for leading and subleading photons
- Object resolution: energy resolution for the leading and subleading photons and jets obtained from the photon [114] and b jet [117] energy regression, the mass resolution estimators of the diphoton and dijet candidates.

The BDT is trained using the XGBOOST [122] software package using a gradient boosting algorithm. The $\gamma\gamma$ + jets and γ + jets MC samples are used as background, while an ensemble of SM HH and the 12 BSM HH benchmark hypotheses listed in Table 5.1 is used as signal. Training on an ensemble of BSM and SM HH signals makes the BDT sensitive to a broad spectrum of theoretical scenarios. During the training, signal events are weighted with the product of the inverse mass resolution of

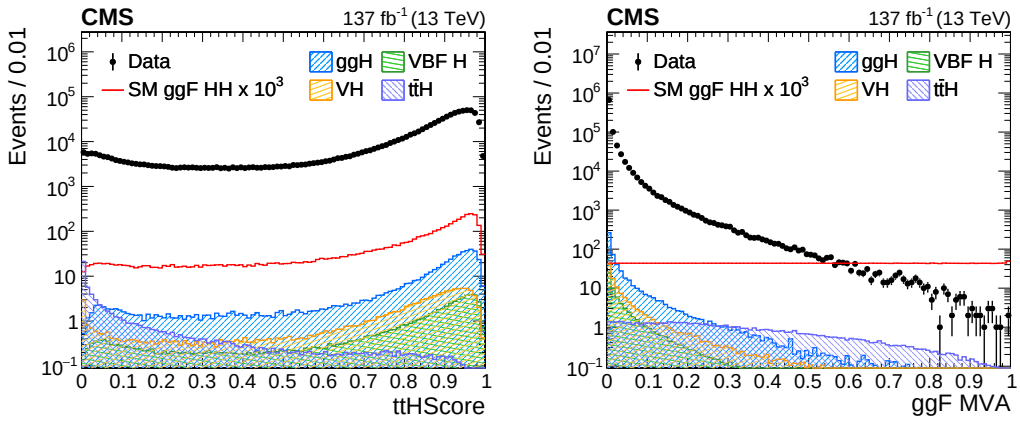


Figure 5.12: The distributions of the $t\bar{t}H$ tagger score (left) and MVA output for nonresonant background in the ggF HH signal region (right) in data and simulated events. Data, dominated by $\gamma\gamma$ + jets and γ + jets backgrounds, are compared to the SM ggF HH signal samples and single H samples ($t\bar{t}H$, ggH, VBF H, VH) after imposing the selection criteria defined in Sec. 5.2. The error bars on the data points indicate statistical uncertainties. The HH signal has been scaled by a factor of 10^3 for display purposes.

the diphoton and dijet systems. These resolutions are obtained using the per-object resolution estimators provided by the energy regressions developed for photons and b jets. In the training, the mass dependence of the classifier is removed by using only dimensionless kinematic variables. The inverse resolution weighting at training time improves the performance by bringing back the information about the resonant nature of the signal. Independent training and testing samples are created by splitting the signal and background samples. The classifier hyperparameters are optimized using randomized grid search and a 5-fold cross-validation technique. The BDT is trained separately for the 2016, 2017, and 2018 data-taking years. The BDT output distribution is very similar among the three years, leading to the same definitions of optimal signal regions based on the BDT output. The BDT output distribution is very similar among the three years, leading to the same definitions of optimal signal regions based on the BDT output. Therefore, during the event categorization, a single set of analysis categories is defined using data from 2016-2018. The distributions of the BDT output for signal and background are very well separated. In order to avoid problems of numerical precision when defining optimal signal-enriched regions, the BDT output is transformed such that the signal distribution is uniform. This transformation is applied to all events, both in simulation and data. The distribution of the MVA output for data and simulated events is shown in Fig. 5.12 (right).

Background reduction in the VBF HH signal region

Similarly to the ggF HH analysis strategy, an MVA discriminant is employed to separate the VBF HH signal from the background. As for the ggF case, the $\gamma\gamma$ + jets and γ + jets processes are the dominant sources of background. For the VBF production mode, the ggF HH events are considered as background. About a third of the ggF HH events passing the selection requirements described in Sec. 5.2 also pass the dedicated VBF selection criteria. The distinctive topology of the VBF HH process is used to separate the VBF HH signal from the various sources of background. In additional to the discriminating features of the HH signal described in Sec. 5.3 and in ggF HH part above, the following set of VBF-discriminating features were identified:

- VBF-tagged jet kinematic variables: $p_{\mathrm{T}}^{\mathrm{VBF}}/m_{\mathrm{jj}}^{\mathrm{VBF}}$, η^{VBF} for VBF-tagged jets
- VBF-tagged jet invariant mass: $m_{\mathrm{jj}}^{\mathrm{VBF}}$
- Rapidity gap: product of the difference in the pseudorapidity of the two VBF-tagged jets
- Quark-gluon likelihood [123, 124] of the two VBF-tagged jets. A likelihood discriminator used to distinguish between jets originating from quarks and from gluons
- Kinematic variables related to the HH system: $\widetilde{M}_{\mathrm{X}}$ and the transverse momentum of the pair of reconstructed Higgs bosons
- Angular distance: minimum ΔR between a photon and a VBF-tagged jet, and between a b jet and a VBF-tagged jet
- Centrality variables for the reconstructed Higgs boson candidates:

$$C_{\mathrm{H}} = \exp\left[-\frac{4}{(\eta_1^{\mathrm{VBF}} - \eta_2^{\mathrm{VBF}})^2}\left(\eta^{\mathrm{H}} - \frac{\eta_1^{\mathrm{VBF}} + \eta_2^{\mathrm{VBF}}}{2}\right)^2\right], \tag{5.4}$$

  where H is the Higgs boson candidate reconstructed either from diphoton or dijet pairs, and η_1^{VBF} and η_2^{VBF} are the pseudorapidities of the two VBF-tagged jets

We split events into two regions: $\widetilde{M}_{\mathrm{X}} < 500$ GeV and $\widetilde{M}_{\mathrm{X}} \geq 500$ GeV. While the region of $\widetilde{M}_{\mathrm{X}} \geq 500$ GeV is sensitive to anomalous values of $c_{2\mathrm{V}}$, the $\widetilde{M}_{\mathrm{X}} < 500$ GeV region retains the sensitivity to SM VBF HH production.

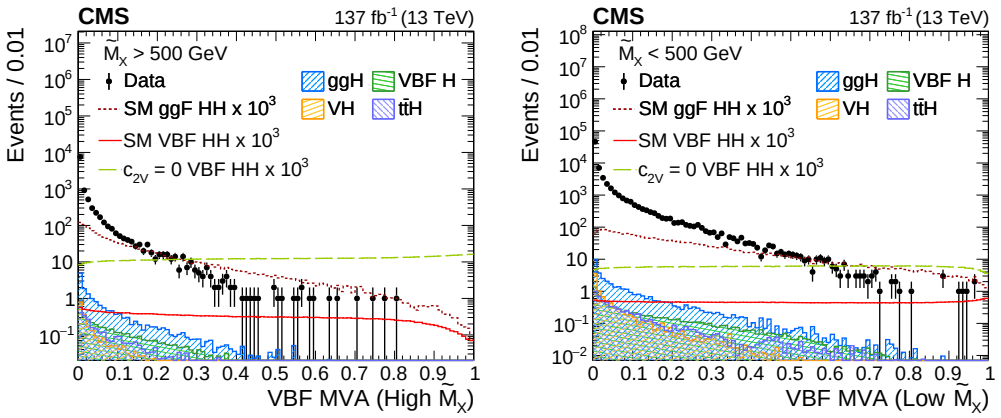


Figure 5.13: The distribution of the two MVA outputs is shown in data and simulated events in the two VBF $\widetilde{M}_X$ regions: $\widetilde{M}_X > 500$ GeV (left) and $\widetilde{M}_X < 500$ GeV (right). Data, dominated by the $\gamma\gamma$ + jets and γ + jets backgrounds, are compared to the VBF HH signal samples with SM couplings and $c_{2V} = 0$, SM ggF HH and single H samples ($t\bar{t}H$, ggH, VBF H, VH) after imposing the VBF selection criteria described in Sec. 5.2. The error bars on the data points indicate statistical uncertainties. The HH signal has been scaled by a factor of 10^3 for display purposes.

A multi-class BDT, using a gradient boosting algorithm and implemented in the XGBOOST framework, is trained to separate the VBF HH signal from the $\gamma\gamma$ + jets, γ + jets, and SM ggF HH background. A mix of VBF HH samples with the SM couplings and quartic coupling $c_{2V} = 0$ is used as signal. Training on the mix of samples makes the BDT sensitive to both SM and BSM scenarios. Although the kinematic properties of different BSM signals with anomalous values of c_{2V} are similar, the cross section of the signal with $c_{2V} = 0$ is significantly enhanced with respect to that predicted by the SM. Therefore, the signal samples used for the training were chosen to maximize sensitivity of the analysis to a range of potential signals. Signal events are weighted with the inverse of the mass resolution of the diphoton and dijet systems during the training, as it is done for the ggF MVA. The BDT is trained separately for each of the three data-taking years in the two $\widetilde{M}_X$ regions. As it is done for the ggF MVA output, data from 2016-2018 are merged to create a single set of analysis categories based on the BDT output. The BDT output is transformed such that the distribution of the mix of the VBF HH signals with SM couplings and quartic coupling $c_{2V} = 0$ is uniform. The transformation is applied to all events in the two $\widetilde{M}_X$ regions. The distributions of the MVA outputs for data and simulated events are shown in Fig. 5.13.

Agreement between data and simulation

For the final signal extraction, the background is estimated in a data-driven way. The Monte Carlo simulation is only used to train the MVA. Nevertheless, $\gamma\gamma$ + jets and γ + jets simulation is known not to describe data very well because the MC simulation is done at LO while high order corrections are not negligible. We perform a check on the agreement between data and simulation for the input variables of the MVA classifier, as well as for the final MVA output. Figs. 5.14-5.16 show the comparison between data and simulation for MVA inputs, and Fig. 5.17 shows the distribution of the discriminator outputs. The only variable where the agreement is not good is the photon ID; however, this is only due to the fact that QCD MC is not included in these plots. In QCD processes the fake photon contributions would populate the region where photon ID < 0. The photon identification was thoroughly validated within the CMS working group; therefore, the disagreement between data and simulation for photon ID shown in Fig. 5.14 is well understood and does not affect the analysis.

5.6 Event categorization

In order to maximize the sensitivity of the search, events are split into different categoreis according to the ouptut of the MVA classifier and the mass of the Higgs boson pair system $\widetilde{M}_{\mathrm{X}}$. The $\widetilde{M}_{\mathrm{X}}$ distribution changes significantly for different BSM hypotheses, as shown in Fig. 5.4. Therefore, a categorization of HH events in $\widetilde{M}_{\mathrm{X}}$ creates signal regions sensitive to multiple theoretical scenarios. In the search for VBF HH production, the categories in $\widetilde{M}_{\mathrm{X}}$ are defined before the MVA is trained, as described in Sec. 5.5. For the categories that target ggF HH production, categories in $\widetilde{M}_{\mathrm{X}}$ are defined after the MVA is trained.

The categorization is optimized by maximizing the expected significance estimated as the sum in quadrature of $\mathrm{S}/\sqrt{\mathrm{B}}$ over all categories in a window centered on m_{H}: $115 < m_{\gamma\gamma} < 135$ GeV. Here, S and B are the numbers of expected signal and background events, respectively. Simulated events are used for this optimization. The SM HH process is considered as signal, while the background consists of the $\gamma\gamma$ + jets, γ + jets, and the $\mathrm{t\bar{t}H}$ processes. The MVA categories are optimized simultaneously with a threshold on the value of the $\mathrm{t\bar{t}H}$ tagger score. Two VBF and three ggF categories are optimized based on the MVA output. For ggF HH in each MVA category, a set of $\widetilde{M}_{\mathrm{X}}$ categories is then optimized. The optimization procedure leads to 12 ggF analysis categories: four categories in $\widetilde{M}_{\mathrm{X}}$ in each of the three categories in the MVA score. The optimized selection on $\mathrm{t\bar{t}H}$ tagger score > 0.26 corresponds to 80 (85)% $\mathrm{t\bar{t}H}$ background rejection at 95 (90)% signal efficiency

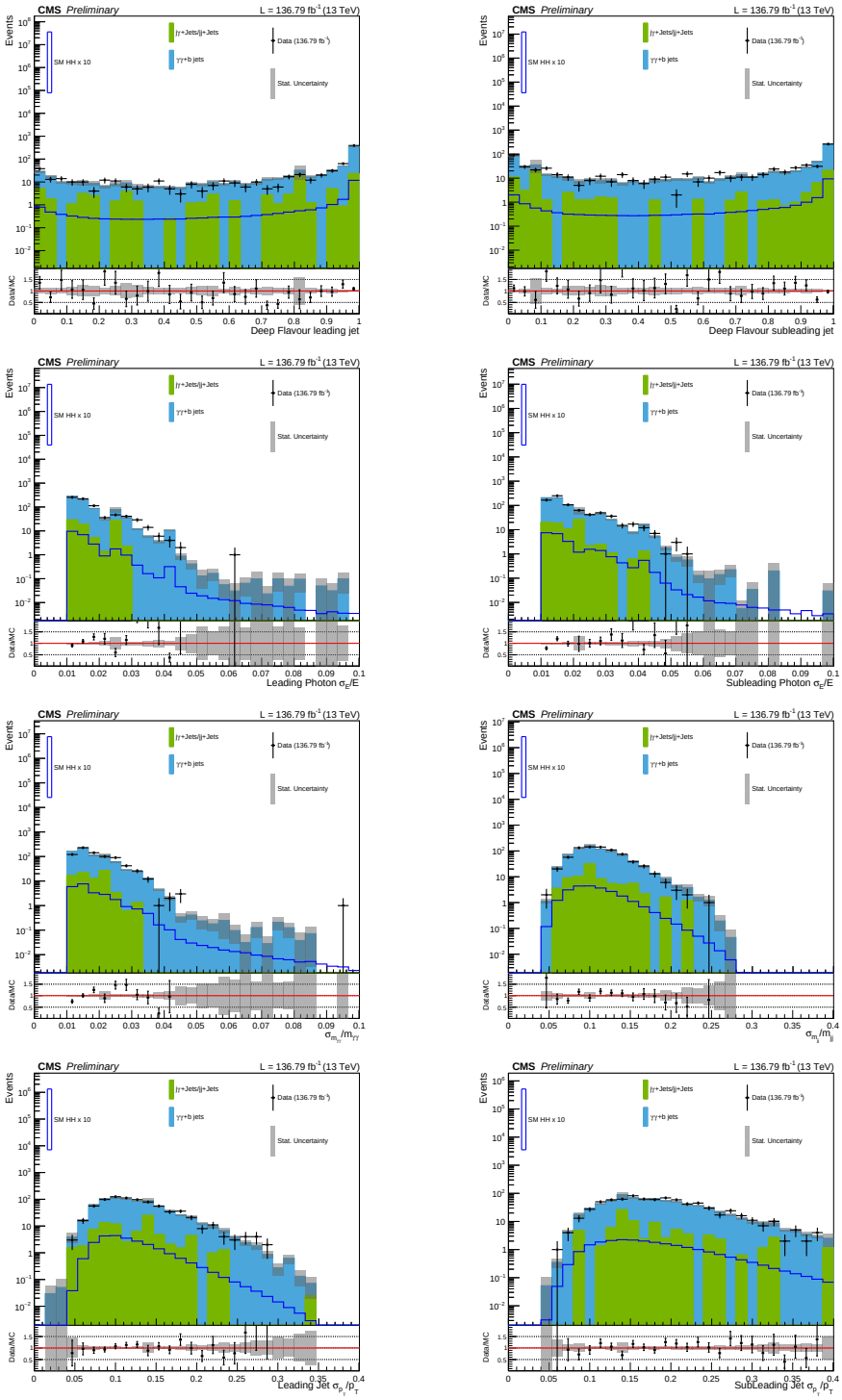


Figure 5.14: Distributions of input variables for non-resonant background in simulated SM HH sample and data for full Run II.

for the 12 ggF (2 VBF) categories. The categorization is summarized in Table 5.6. The VBF and ggF categories are mutually exclusive, as we only consider events that

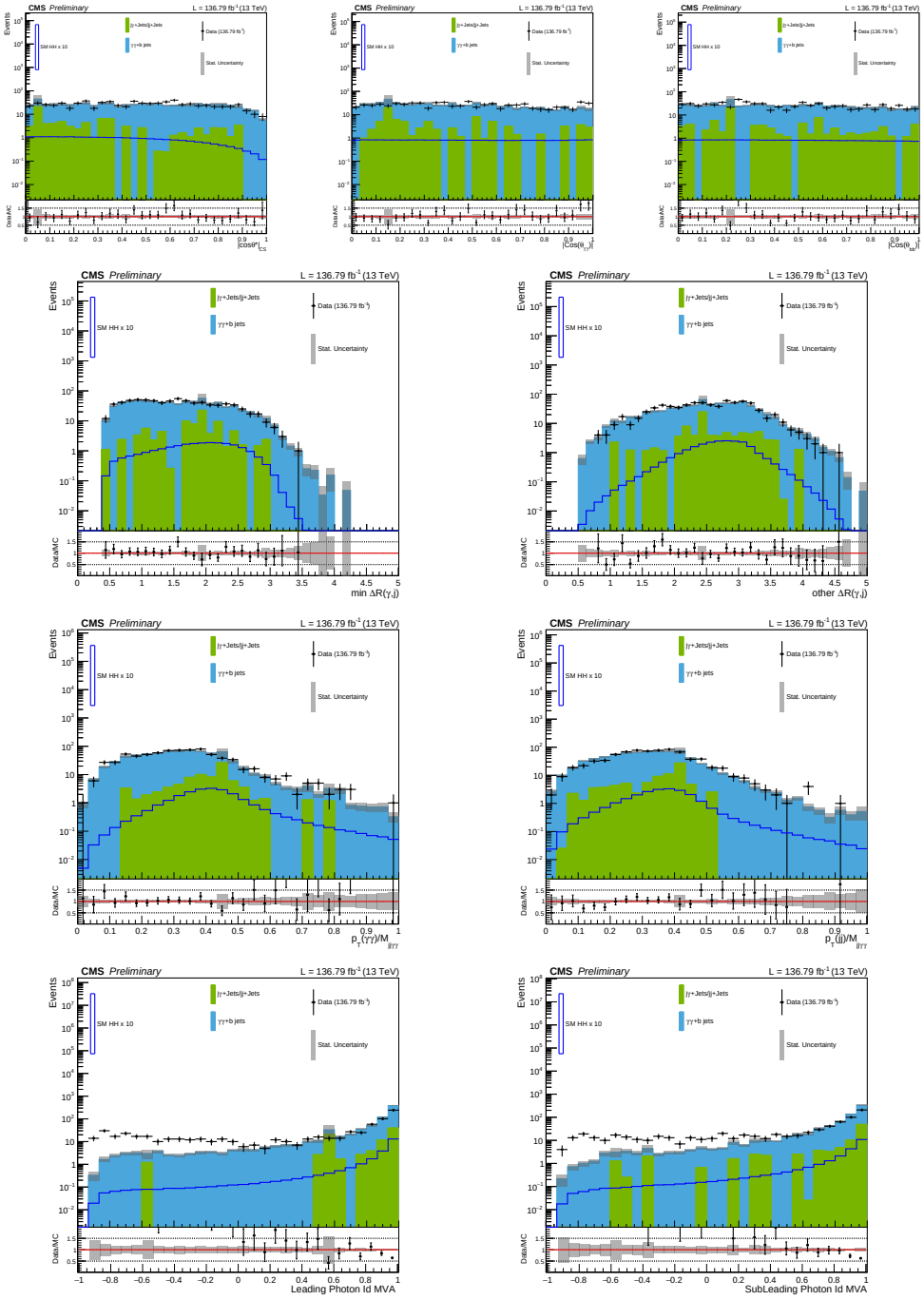


Figure 5.15: Distributions of input variables for non-resonant background in simulated SM HH sample and data for full Run II.

do not enter the VBF categories for the ggF categories. Events with the VBF MVA scores below 0.52 (0.86) for $\widetilde{M}_X > 500$ ($\widetilde{M}_X < 500$) GeV are not considered in the VBF signal region. Because of the overwhelming background contamination, such

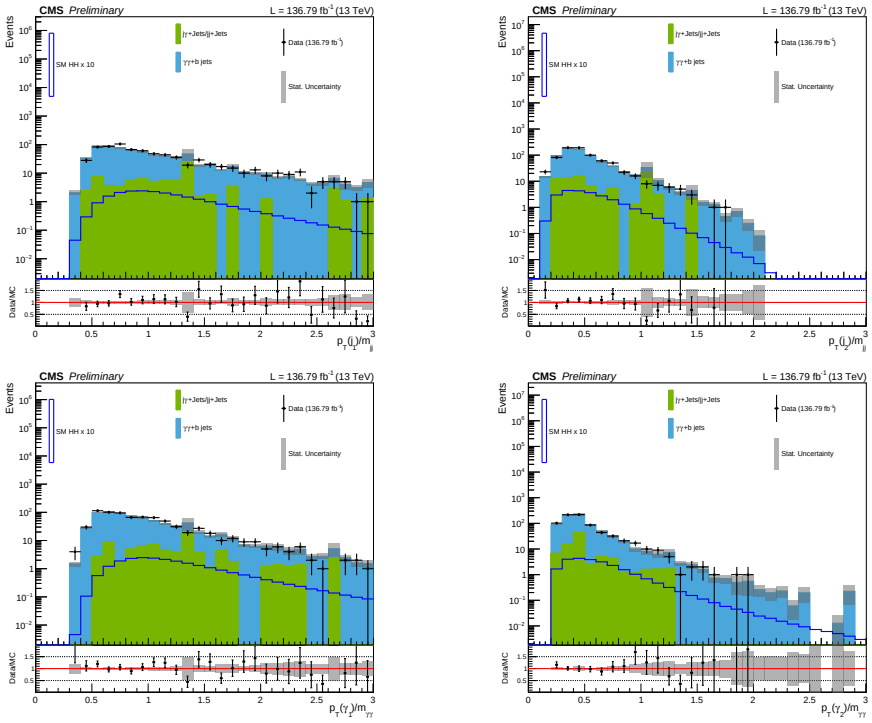


Figure 5.16: Distributions of input variables for non-resonant background in simulated SM HH sample and data for full Run II.

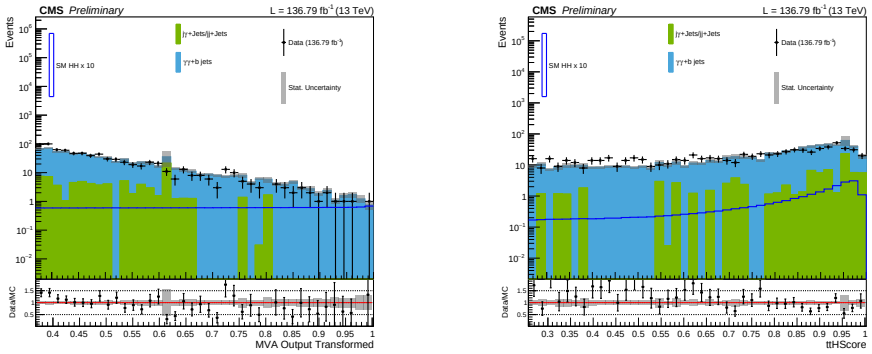


Figure 5.17: Distributions of outputs of MVA and $t\bar{t}H$ tagger for non-resonant background in simulated SM HH sample and data for full Run II.

events do not improve the expected sensitivity of the analysis. Similarly, events with ggF MVA scores below 0.37 are not considered in the ggF signal region.

5.7 Combination of the HH and $t\bar{t}H$ signals to constrain κ_λ and κ_t

As discussed in the introduction of this chapter, the HH production cross section depends on κ_λ and κ_t. The production cross section of the single H processes also depends on κ_λ, as a result of the NLO electroweak corrections [125]. The

Table 5.6: Summary of the analysis categories. Two VBF- and twelve ggF-enriched categories are defined based on the output of the MVA classifiers and the mass of the Higgs boson pair system $\widetilde{M}_{\mathrm{X}}$. The VBF and ggF categories are mutually exclusive.

Category	MVA	$\widetilde{M}_{\mathrm{X}}$ (GeV)
VBF CAT 0	0.52–1.00	> 500
VBF CAT 1	0.86–1.00	250–500
ggF CAT 0	0.78–1.00	> 600
ggF CAT 1		510–600
ggF CAT 2		385–510
ggF CAT 3		250–385
ggF CAT 4	0.62–0.78	> 540
ggF CAT 5		360–540
ggF CAT 6		330–360
ggF CAT 7		250–330
ggF CAT 8	0.37–0.62	> 585
ggF CAT 9		375–585
ggF CAT 10		330–375
ggF CAT 11		250–330

ggH and $\mathrm{t\bar{t}H}$ production cross sections additionally depend on κ_t. Therefore, the $\mathrm{HH} \rightarrow \mathrm{b\bar{b}}\gamma\gamma$ signal can be combined with the single H production modes to provide an improved constraint on the κ_λ and κ_t parameters. In the case of anomalous values of κ_λ, the single H process with the largest modification of the cross section is $\mathrm{t\bar{t}H}$. For this reason, additional orthogonal categories targeting the $\mathrm{t\bar{t}H}$ process are included in the analysis: the "$\mathrm{t\bar{t}H}$ leptonic" and "$\mathrm{t\bar{t}H}$ hadronic" categories, developed and optimized for the measurement of the $\mathrm{t\bar{t}H}$ production cross section in the diphoton decay channel [126]. The events that do not pass the selections for the HH categories defined in Table 5.6 are tested for the $\mathrm{t\bar{t}H}$ categories. This ensures the orthogonality between the events selected by the HH and $\mathrm{t\bar{t}H}$ categories.

The $\mathrm{H} \rightarrow \gamma\gamma$ candidate selection is the same as described in Sec. 5.2. The $\mathrm{t\bar{t}H}$ leptonic categories target $\mathrm{t\bar{t}H}$ events where at least one W boson, originating from the top or antitop quark, decays leptonically. At least one isolated electron (muon) with $|\eta| < 2.4$ and $p_{\mathrm{T}} > 10$ (5) GeV, and at least one jet with $p_{\mathrm{T}} > 25$ GeV are required. The $\mathrm{t\bar{t}H}$ hadronic categories target hadronic decays of W bosons. In these categories at least 3 jets are required, one of which must be b tagged, and a lepton veto is imposed. In order to maximize the sensitivity, an MVA approach is used to separate the $\mathrm{t\bar{t}H}$ events from the background, dominated by $\gamma\gamma$ + jets, γ + jets, $\mathrm{t\bar{t}}$ + jets, $\mathrm{t\bar{t}} + \gamma$, and $\mathrm{t\bar{t}} + \gamma\gamma$ events. A BDT classifier is trained for each of the

two channels using simulated events. The variables used for the training include kinematic properties of the reconstructed objects, object identification variables, and global event properties such as jet and lepton multiplicities. The BDT input variables also include the outputs of other machine learning algorithms trained specifically to target different backgrounds. These include DNN classifiers trained to reduce $t\bar{t}+\gamma\gamma$ and $\gamma\gamma$ + jets background, and a top quark tagger based on a BDT [127]. The output scores of the BDTs are used to reject background-like events and to classify the remaining events in four subcategories for each of the two channels. The boundaries of the categories are optimized by maximizing the expected significance of the $t\bar{t}H$ signal.

5.8 Signal modeling

In each of the HH categories, a parametric fit in the ($m_{\gamma\gamma}$, m_{jj}) plane is performed. In the $t\bar{t}H$ categories, the m_{jj} distribution is fitted to extract the signal. When the HH and $t\bar{t}H$ categories are combined, both the HH and $t\bar{t}H$ production modes are considered as signals.

The shape templates of the diphoton and dijet invariant mass distributions are constructed from simulation. In each HH and $t\bar{t}H$ analysis category, the $m_{\gamma\gamma}$ distribution is fitted using a sum of, at most, five Gaussian functions. Figure 5.18 (left) shows the signal model for $m_{\gamma\gamma}$ in the VBF and ggF CAT 0 categories, which are categories with the best resolution.

For the HH categories, the m_{jj} distributions are modeled with a double-sided Crystal Ball (CB) function, a modified version of the standard CB function [128] with two independent exponential tails. Figure 5.18 (right) shows the signal model for the m_{jj} in the VBF and ggF categories with the best resolution.

For the HH signal, the final two-dimensional (2D) signal probability distribution is a product of the independent $m_{\gamma\gamma}$ and m_{jj} models. The possible correlations are investigated by comparing the 2D $m_{\gamma\gamma}$–m_{jj} distributions in the simulated signal samples with the 2D probability distributions built as a product of the one-dimensional (1D) ones. With the statistical precision available in this analysis, the correlations have been found to be negligible.

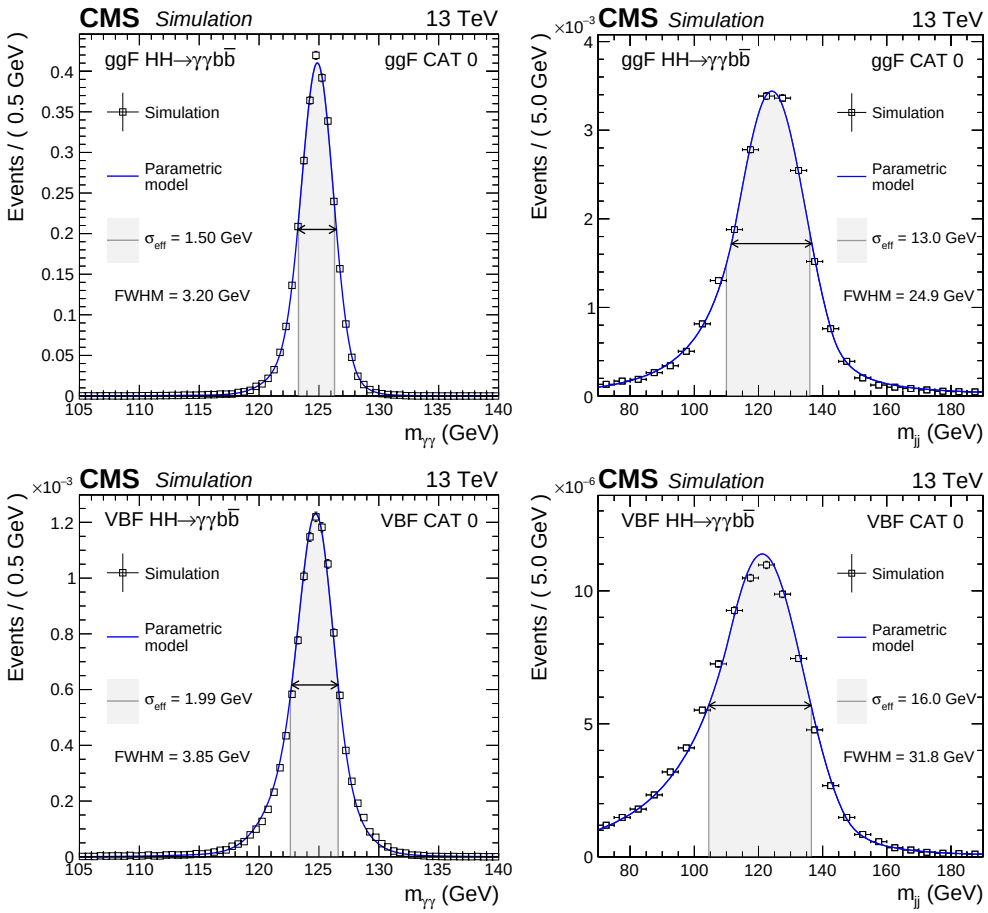


Figure 5.18: Parametrized signal shape for $m_{\gamma\gamma}$ (left) and m_{jj} (right) in the best resolution ggF (upper) and VBF (lower) categories. The open squares represent simulated events and the blue lines are the corresponding models. Also shown are the σ_{eff} value (half the width of the narrowest interval containing 68.3% of the invariant mass distribution) and the corresponding interval as a gray band, and the full width at half the maximum (FWHM) and the corresponding interval as a double arrow.

5.9 Background modeling

Single Higgs background modeling

The SM single H background shape is constructed from the simulation following the same methodology as used for the signal model described in Sec. 5.8. For each analysis category and single H production mode, the m_{jj} distributions are fitted using a sum of, at most, five Gaussian functions. The m_{jj} modeling in the HH categories depends on the production mechanism, and a parametrization is obtained from the simulated distributions: for the ggH and VBF H processes, the m_{jj} distribution is modeled with a Bernstein polynomial. This is motivated by the fact that there is no intrinsic mass scale for the jet production in these processes; we expect a falling spectrum. For VH production, a CB function is used to model the distribution of the hadronic decays of vector bosons. For $t\bar{t}H$, where the b jets are produced from top quark decay, a Gaussian function with a mean around 120 GeV is used. The minimal mass of a system of 2 top quarks is 350 GeV, and the $b\bar{b}$ system typically takes 1/3 of this energy. Like the signal modeling, the final 2D SM single-Higgs boson model is an independent product of models of the $m_{\gamma\gamma}$ and m_{jj} distributions.

Due to the limited statistics left in the single Higgs simulation after the categorization, it is impractical to construct single Higgs shapes in each individual category per year. In case of $m_{\gamma\gamma}$ distribution for all single Higgs production modes, a narrow Higgs peak is observed. Therefore, if there is not enough MC statistics in a given category, the $m_{\gamma\gamma}$ shape constructed for $t\bar{t}H$ process is used. $t\bar{t}H$ is chosen because of simulation statistics in all categories.

For m_{jj} modeling, where m_{jj} model depends on the production mechanism, we merge 3 data-taking years and in addition merge 12 (3 MVA $\times$ 4 $\widetilde{M}_X$) categories into 3 MVA categories and fit the shape in each of the 3 MVA categories. Then the fitted shape is propagated to the 12 categories used in the analysis with the correct normalization in each category. Fig. 5.19 shows the models for ggH, VBF H, VH, and $t\bar{t}H$ processes in one of the MVA categories.

Nonresonant background modeling

The model used to describe the nonresonant background is extracted from data using the discrete profiling method [129] as described in Ref. [130]. This technique was designed as a way to estimate the systematic uncertainty associated with choosing a particular analytic function to fit the background m_{jj} and $m_{\gamma\gamma}$ distributions. The method treats the choice of the background function as a discrete nuisance parameter

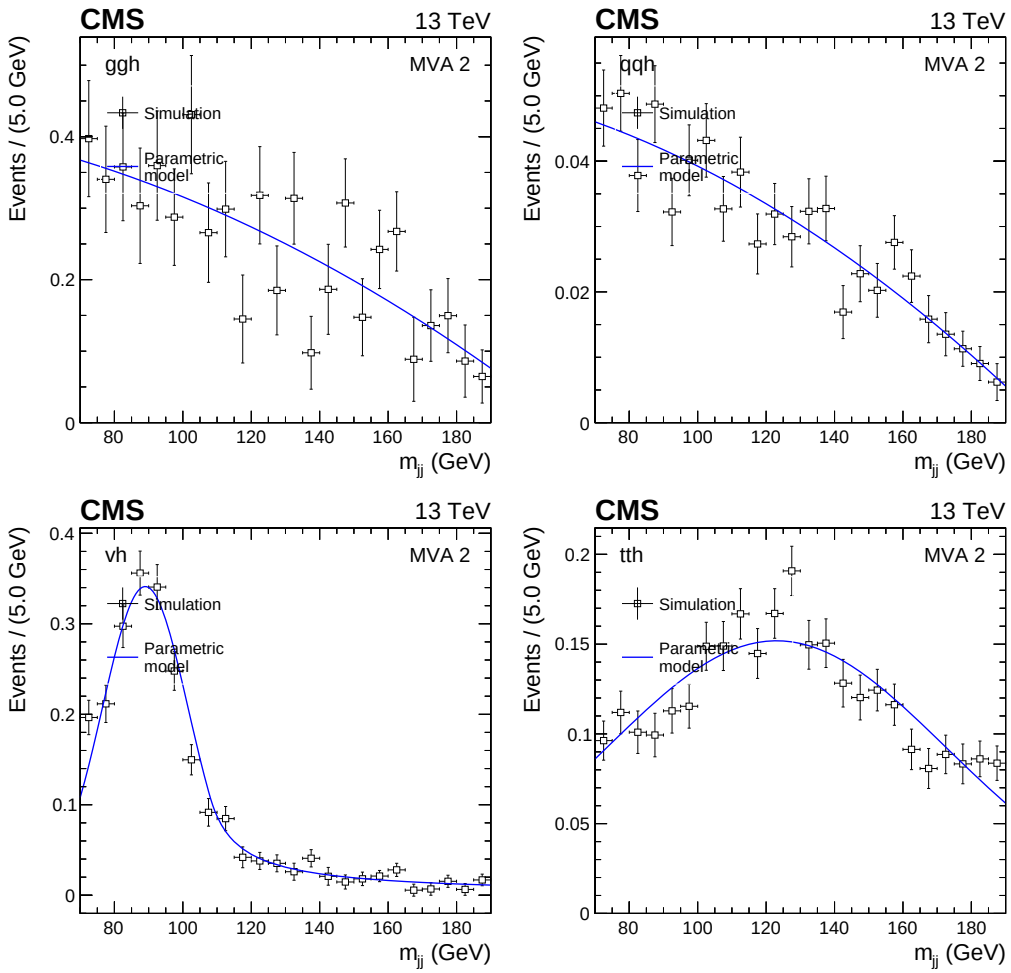


Figure 5.19: Parametrized background shape for m_{jj} distributions for ggH (top left), VBF H (top right), VH (bottom left), and $t\bar{t}H$ (bottom right) in one of the MVA categories. The open squares represent simulated events and the blue lines are the corresponding models. Also shown are the σ_{eff} value (half the width of the narrowest interval containing 68.3% of the invariant mass distribution) and the corresponding interval as a gray band, and the full width at half the maximum (FWHM) and the corresponding interval as a double arrow.

in the likelihood to fit the data. We consider three families of analytic functions: polynomials in the Bernstein basis, sums of exponentials, and sums of power-law functions. An F-test is used to select the representative functions from each of these families to proceed with discrete profiling. To choose maximum orders of functions in each family, we fit the data sequentially, increasing the function order and the difference of negative log-likehood (2ΔNLL) between two consecutive fits. This 2ΔNLL is distributed as a $\chi^2(n)$ distribution, where the degrees of freedom n equal to the difference in the number of free parameters between two consecutive orders.

We continue to increase the function order until the p-value of having a 2ΔNLL higher than the one calculated before it becomes larger than 0.05. This means that the function with the next order gives no significant improvement to the fit to the data. We perform this procedure for $m_{\gamma\gamma}$ and m_{jj} variables separately, and then use all combinations of functions chosen for $m_{\gamma\gamma}$ and m_{jj} projections for discrete profiling method.

5.10 Systematic uncertainties

The systematic uncertainties only affect the signal model and the resonant single H background, since the nonresonant background model is constructed in a data-driven way with the uncertainties associated with the choice of a background fit function taken into account by the discrete profiling method described in Sec. 5.9. The systematic uncertainties can affect the overall normalization, or a variation in category yields, representing event migration between the categories. Theoretical uncertainties have been applied to the HH and single H normalizations. The following sources of theoretical uncertainty are considered: the uncertainty in the signal cross section arising from scale variations, uncertainties on α_{S}, PDFs and in the prediction of the branching fraction $\mathcal{B}(\mathrm{HH} \to \mathrm{b\bar{b}}\gamma\gamma)$. The dominant theoretical uncertainties arise from the prediction of the SM HH and $\mathrm{t\bar{t}H}$ production cross sections. In addition, a conservative parton shower (PS) uncertainty is assigned to the VBF HH signal, defined as the full symmetrized difference in yields in each category obtained with simulated samples of VBF HH events interfaced with the standard p_{T}-ordered and dipole shower PS schemes.

The dominant experimental uncertainties are:

- *Photon identification BDT score*: the uncertainty arising from the imperfect MC simulation of the input variables to the photon ID is estimated by rederiving the corrections with equally sized subsets of the $\mathrm{Z} \to \mathrm{ee}$ events used to train the quantile regression BDTs. Its magnitude corresponds to the standard deviation of the event-by-event differences in the photon ID evaluated on the two different sets of corrected input variables. This uncertainty reflects the limited capacity of the BDTs arising from the finite size of the training set. It is seen to cover the residual discrepancies between data and simulation. The uncertainty in signal yields is estimated by propagating this uncertainty through the full category selection procedure.

- *Photon energy scale and resolution*: the uncertainties associated with the corrections applied to the photon energy scale in data and the resolution in simulation are evaluated using $\mathrm{Z} \to \mathrm{ee}$ events [131].

- *Per-photon energy resolution estimate*: the uncertainty in the per-photon resolution is parametrized as rescaling of the resolution by ±5% around its nominal value. This is designed to cover all differences between data and simulation in the distribution, which is an output of the energy regression.

- *Jet energy scale and resolution corrections*: the energy scale of jets is measured using the p_{T}balance of jets with Z bosons and photons in $\mathrm{Z} \to \mathrm{ee}, \mathrm{Z} \to \mu\mu$, and γ + jets events, as well as using the p_{T}balance between jets in dijet and multijet events [124, 132]. The uncertainty in the jet energy scale and resolution is a few percent and depends on p_{T} and η. The impact of uncertainties on the event yields is evaluated by varying the jet energy corrections within their uncertainties and propagating the effect to the final result. Some source of the jet energy scale uncertainty are fully (anti-)correlated, while others are considered uncorrelated.

- *Jet b tagging*: uncertainties in the b tagging efficiency are evaluated by comparing data and simulated distributions for the b tagging discriminator [59]. These include the statistical uncertainty in the estimate of the fraction of heavy- and light-flavor jets in data and simulation.

- *Trigger efficiency*: the efficiency of the trigger selection is measured with $\mathrm{Z} \to \mathrm{ee}$ events using a tag-and-probe technique [133]. An additional uncertainty is introduced to account for a gradual shift in the timing of the inputs of the ECAL L1 trigger in the region $|\eta| > 2.0$, which caused a specific trigger inefficiency during 2016 and 2017 data taking. Both photons, and to a greater extent, jets can be affected by this inefficiency, which has a small impact.

- *Photon preselection*: the uncertainty in the preselection efficiency is computed as the ratio between the efficiency measured in data and in simulation. The preselection efficiency in data is measured with the tag-and-probe technique in $\mathrm{Z} \to \mathrm{ee}$ events [133].

- *Integrated luminosity*: uncertainties are determined by the CMS luminosity monitoring for the 2016–2018 data-taking years [134–136] and are in the range of 2.3–2.5%. To account for common sources of uncertainty in the

luminosity measurement schemes, some sources are fully (anti-)correlated across different data-taking years, while others are considered uncorrelated. The total 2016–2018 integrated luminosity has an uncertainty of 1.8%.

- *Pileup jet identification*: the uncertainty in the pileup jet classification output score is estimated by comparing the score of jets in events with a Z boson and one balanced jet in data and simulation. The assigned uncertainty depends on $p_{\mathrm{T}}$and η, and is designed to cover all differences between data and simulation in the distribution.

Most of the experimental uncertainties are uncorrelated among the three data-taking years. Some sources of uncertainty in the measured luminosity and jet energy corrections are fully (anti-)correlated, while others considered uncorrelated. This search is statistically limited, and the total impact of systematic uncertainties on the result is about 2%.

5.11 Results

An unbinned maximum likelihood fit to the $m_{\gamma\gamma}$ and m_{jj} distributions is performed simultaneously in the 14 HH categories to extract the HH signal. A likelihood function is defined for each analysis category using analytic models to describe the $m_{\gamma\gamma}$ and m_{jj} distributions of signal and background events, with nuisance parameters to account for the experimental and theoretical systematic uncertainties described in Sec. 5.10. The fit is performed in the mass ranges $100 < m_{\gamma\gamma} < 180$ GeV and $70 < m_{\mathrm{jj}} < 190$ GeV for all categories apart from ggF CAT 10 and CAT 11. In those two categories, a small but non-negligible shoulder was observed in the m_{jj} distribution. Therefore, the m_{jj} fit range is reduced to $90 < m_{\mathrm{jj}} < 190$ GeV to avoid a possible bias with minimal impact on the analysis sensitivity.

In order to determine κ_t and κ_λ, the HH and $\mathrm{t\bar{t}H}$ categories are used together in a simultaneous maximum likelihood fit. In the $\mathrm{t\bar{t}H}$ categories, a binned maximum likelihood fit is performed to $m_{\gamma\gamma}$ in the mass range $100 < m_{\gamma\gamma} < 180$ GeV.

The data and the signal-plus-background model fit to $m_{\gamma\gamma}$ and m_{jj} are shown in Fig. 5.20 for the best resolution ggF and VBF categories. The distribution of events weighted by S/(S+B) from all HH categories is shown in Fig. 5.21 for $m_{\gamma\gamma}$ and m_{jj}. In this expression, S (B) is the number of signal (background) events extracted from the signal-plus-background fit.

No significant deviation from the background-only hypothesis is observed. We set upper limits at 95% CL on the product of the production cross section of a pair of Higgs bosons and the branching fraction into $b\bar{b}\gamma\gamma$, $\sigma_{HH}\mathcal{B}(HH \rightarrow b\bar{b}\gamma\gamma)$, using the modified frequentist approach for confidence levels (CL_s), taking the LHC profile likelihood ratio as a test statistic [137–140] in the asymptotic approximation. The observed (expected) 95% CL upper limit on $\sigma_{HH}\mathcal{B}(HH \rightarrow b\bar{b}\gamma\gamma)$ amounts to 0.67 (0.45) fb. The observed (expected) limit corresponds to 7.7 (5.2) times the SM prediction. All results were extracted assuming $m_H = 125$ GeV. We observe a variation smaller than 1% in both the expected and observed upper limits when using $m_H = 125.38 \pm 0.14$ GeV, corresponding to the most precise measurement of the Higgs boson mass to date [141].

Limits are also derived as a function κ_λ, assuming that the top quark Yukawa coupling is SM-like ($\kappa_t = 1$). The result is shown in Fig. 5.22. The variation in the excluded cross section as a function κ_λ is directly related to changes in the kinematic properties of HH production. At 95% CL, κ_λ is constrained to values in the interval [-3.3, 8.5], while the expected constraint on the κ_λ is in the interval [-2.5, 8.2]. This is the most sensitive search to date.

Assuming instead that an HH signal exists with the properties predicted by the SM, constraints on λ_{HHH} can be set. The results are obtained both with the HH categories only, and with the HH categories combined with the $t\bar{t}H$ categories in a simultaneous maximum likelihood fit. The HH signal is considered with the single H processes ($t\bar{t}H$, ggH, VBF H, VH, and Higgs boson production in association with a single top quark). The cross sections and branching fractions of the HH and single H processes are scaled as a function of κ_λ, while the top quark Yukawa coupling is assumed to be SM-like, $\kappa_t = 1$. One-dimensional negative log-likelihood scans for κ_λ are shown in Fig. 5.23 for an Asimov dataset [139] generated with the SM signal-plus-background hypothesis, $\kappa_\lambda = 1$, and for the observed data. When combining with the HH analysis categories with the $t\bar{t}H$ categories, we obtain $\kappa_\lambda = 0.6^{+6.3}_{-1.8}$ ($1.0^{+5.7}_{-2.5}$ expected). Values of κ_λ outside the interval [-2.7, 8.6] are excluded at 95% CL. The expected exclusion at 95% CL corresponds to the region outside the interval [-3.3, 8.6]. The shape of the likelihood as function of κ_λ in Fig. 5.23 is characterized by 2 minima. This is related to an interplay between the cross section dependence on κ_λ and differences in acceptance between the analysis categories.

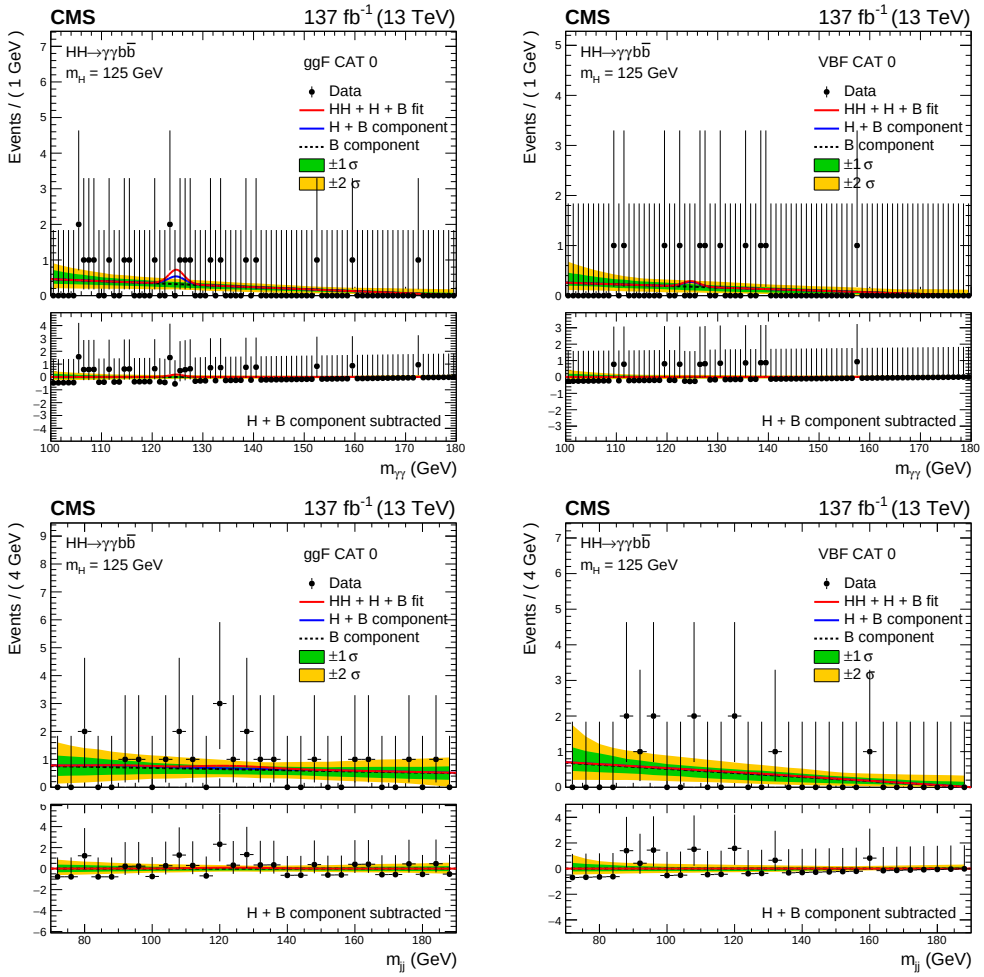


Figure 5.20: Invariant mass distribution $m_{\gamma\gamma}$ (upper) and m_{jj} (lower) for the selected events in data (black points) in the best resolution ggF (CAT 0) and VBF (CAT 0) categories. The solid red line shows the sum of the fitted signal and background (HH+H+B), the solid blue line shows the background component from the single Higgs boson and the nonresonant process (H+B), and the dashed black line shows the nonresonant background component (B). The normalization of each component (HH, H, B) is extracted from the combined fit to the data in all analysis categories. The one (green) and two (yellow) standard deviation bands include the uncertainties in the background component of the fit. The lower panel in each plot shows the residual signal yield after the background (H+B) subtraction.

The HH and single Higgs boson production cross sections depend not only on κ_λ, but also on κ_t. To better constrain the κ_λ and κ_t coupling modifiers, a 2D negative log-likelihood scan in the $(\kappa_\lambda, \kappa_t)$ plane is performed, taking into account the modification of the production cross sections and $\mathcal{B}(\mathrm{H} \to \mathrm{b\bar{b}})$, $\mathcal{B}(\mathrm{H} \to \gamma\gamma)$ for anomalous $(\kappa_\lambda, \kappa_t)$ values. The modification of the single H production cross section for anomalous κ_λ is modeled at NLO, while the dependence on κ_t is parametrized

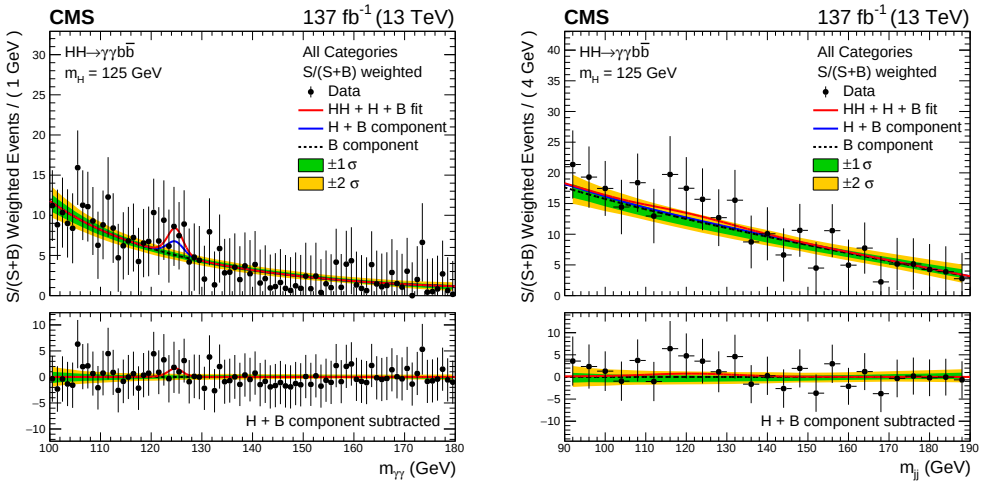


Figure 5.21: Invariant mass distribution $m_{\gamma\gamma}$ (left) and m_{jj} (right) for the selected events in data (black points) weighted by S/(S+B), where S (B) is the number of signal (background) events extracted from the signal-plus-background fit. The solid red line shows the sum of the fitted signal and background (HH+H+B), the solid blue line shows the background component from the single Higgs boson and the nonresonant process (H+B), and the dashed black line shows the nonresonant background component (B). The normalization of each component (HH, H, B) is extracted from the combined fit to the data in all analysis categories. The one (green) and two (yellow) standard deviation bands include the uncertainties in the background component of the fit. The lower panel in each plot shows the residual signal yield after the background (H+B) subtraction.

at LO only, neglecting NLO effects [125]. This approximation holds as long as the value of $|\kappa_t|$ is close to unity, roughly in the range $0.7 < \kappa_t < 1.3$. The parametric model is not reliable outside of this range. Fig. 5.24 shows the 2D likelihood scan of κ_λ versus κ_t for an Asimov data set assuming the SM hypothesis and for the observed data. The regions of the 2D scan where the κ_t parametrization for anomalous values of κ_λ at LO is not reliable are shown wiht a gray band.

The inclusion of the $\mathrm{t\bar{t}H}$ categories significantly improves the constraint on κ_t. The 1D negative log-likelihood scan, as a function κ_t with κ_λ fixed at $\kappa_\lambda = 1$, is shown in Fig. 5.25 for an Asimov data set generated assuming the SM hypothesis, $\kappa_t = 1$, as well as for the observed data. The measured values of κ_t is $\kappa_t = 1.3^{+0.2}_{-0.2}$ ($1.0^{+0.2}_{-0.2}$ expected). Values of κ_t outside the interval [0.9, 1.9] are excluded at 95% CL. The constraint on κ_t is comparable to the one recently set in Ref. [142], where anomalous values of c_{V} were also considered.

Upper limits at 95% CL are also set on the product of the HH VBF production cross section and branching fraction, $\sigma_{\mathrm{VBF\,HH}}\mathcal{B}(\mathrm{HH} \to \mathrm{b\bar{b}}\gamma\gamma)$, with the yield of

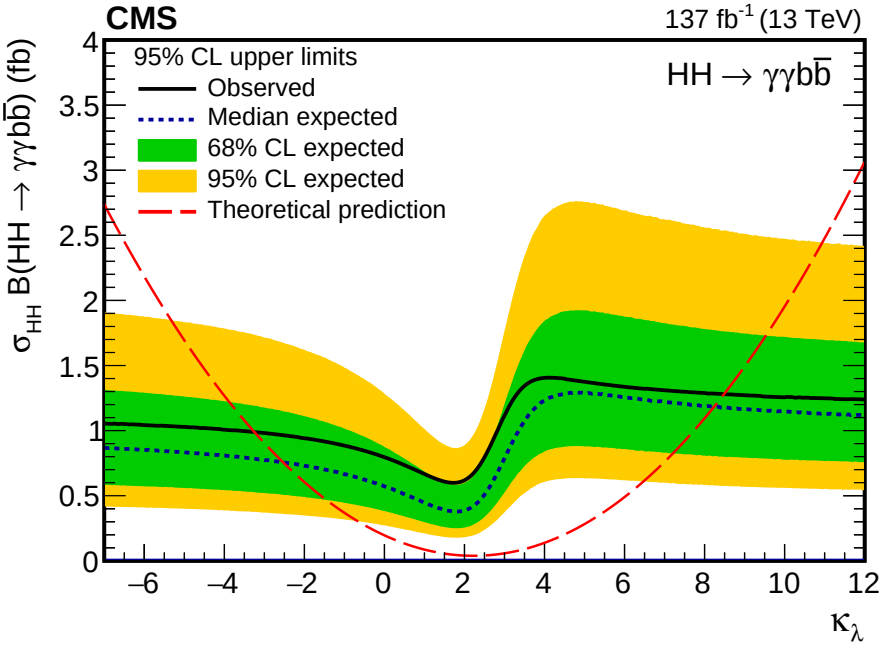


Figure 5.22: Expected and observed 95% CL upper limits on the product of the HH production cross section and $\mathcal{B}(\mathrm{HH} \rightarrow \mathrm{b\bar{b}}\gamma\gamma)$ obtained for different values of κ_λ assuming $\kappa_t = 1$. The green and yellow bands represent, respectively, the one and two standard deviation extensions beyond the expected limit. The long-dashed red line shows the theoretical prediction.

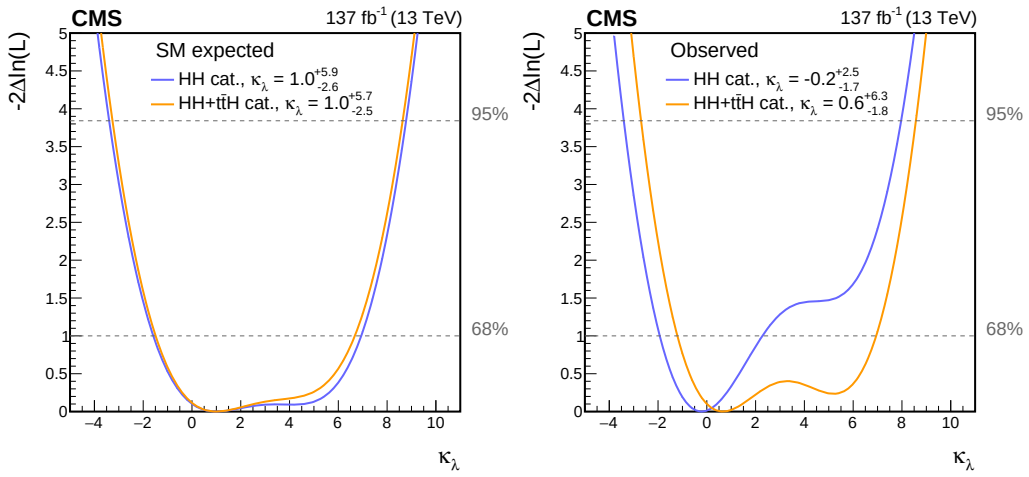


Figure 5.23: Negative log-likelihood, as a function of κ_λ, evaluated with an Asimov data set assuming the SM hypothesis (left) and the observed data (right). The 68 and 95% CL intervals are shown with the dashed gray lines. The two curves are shown for the HH (blue) and HH + $\mathrm{t\bar{t}H}$ (orange) analysis categories. All other couplings are set to their SM values.

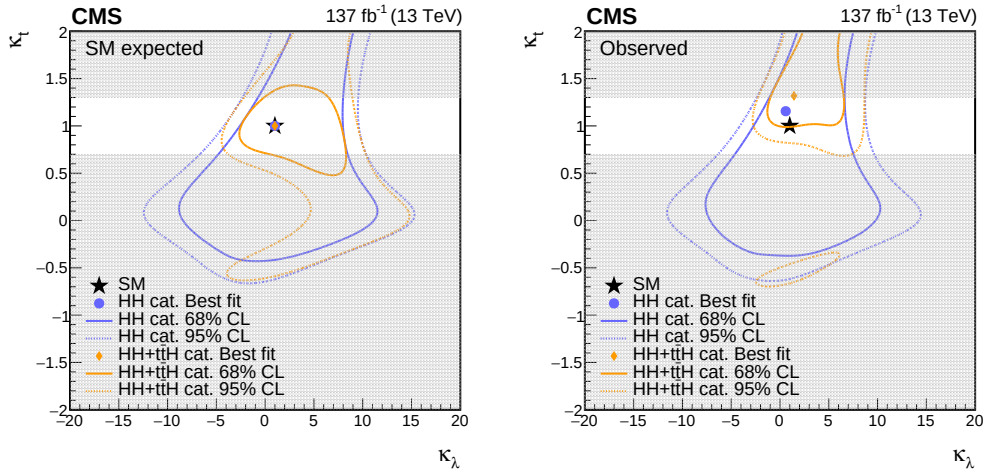


Figure 5.24: Negative log-likelihood contours at 68 and 95% CL in the (κ_λ, κ_t) plane evaluated with an Asimov data set assuming the SM hypothesis (left) and the observed data (right). The contours obtained using the HH analysis categories only are shown in blue, and in orange when combined with the $t\bar{t}H$ categories. The best fit value for the HH categories only ($\kappa_\lambda = 0.6$, $\kappa_t = 1.2$) is indicated by a blue circle, for the HH + $t\bar{t}H$ categories ($\kappa_\lambda = 1.4$, $\kappa_t = 1.3$) by an orange diamond, and the SM prediction ($\kappa_\lambda = 1.0$, $\kappa_t = 1.0$) by a black star. The regions of the 2D scan whether the κ_t parametrization for anomalous values of κ_λ at LO is not reliable are shown with a gray band.

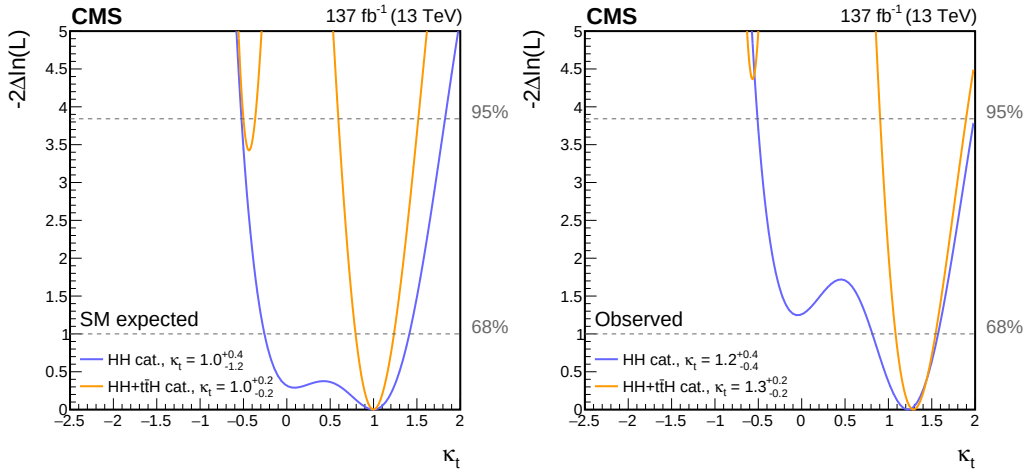


Figure 5.25: Negative log-likelihood scan, as a function of κ_t, evaluated with an Asimov data set assuming the SM hypothesis (left) and the observed data (right). The 68 and 95% CL intervals are shown with the dashed gray lines. The two curves are shown for the HH (blue) and the HH + $t\bar{t}H$ (orange) analysis categories. All other couplings are fixed to their SM values.

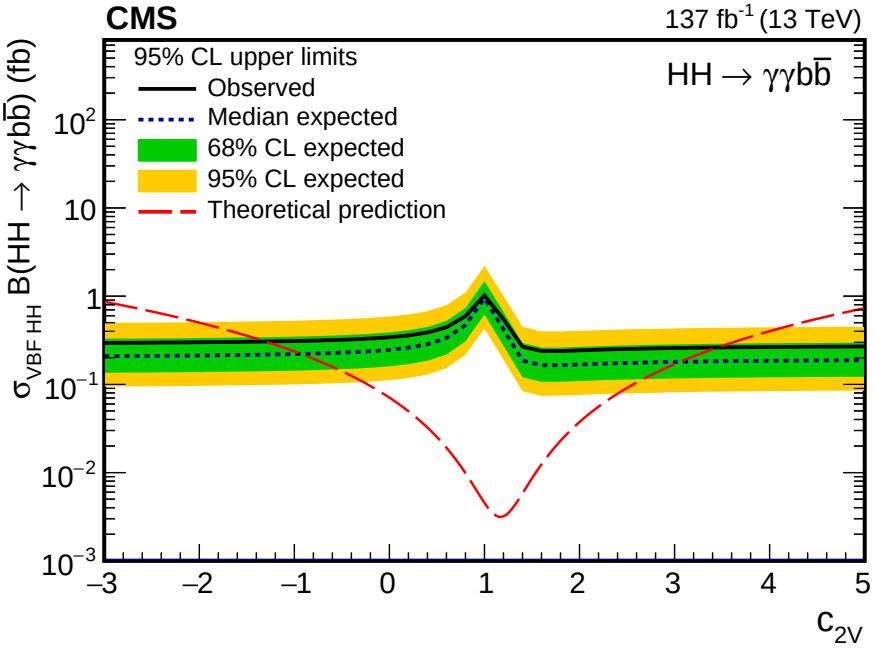


Figure 5.26: Expected and observed 95% CL upper limits on the product of the VBF HH production cross section and $\mathcal{B}(\mathrm{HH} \to \mathrm{b\bar{b}}\gamma\gamma)$ obtained for different values of $c_{2\mathrm{V}}$. The green and yellow bands represent, respectively, the one and two standard deviation extensions beyond the expected limit. The long-dashed red line shows the theoretical prediction.

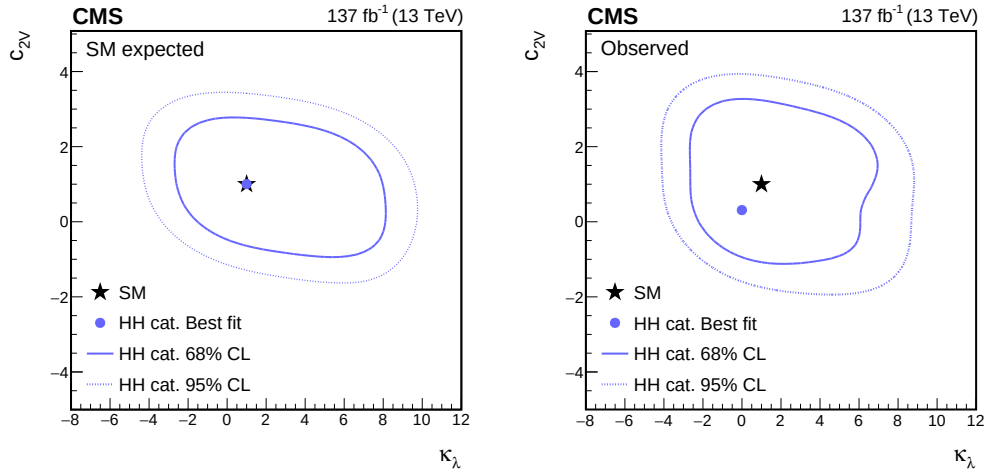


Figure 5.27: Negative log-likelihood contours at 68 and 95% CL in the (κ_λ, $c_{2\mathrm{V}}$) plane evaluated with an Asimov data set assuming the SM hypothesis (left) and with the observed data (right). The contours are obtained using the HH analysis categories only. The best fit value ($\kappa_\lambda = 0.0$, $c_{2\mathrm{V}} = 0.3$) is indicated by a blue circle, and the SM prediction ($\kappa_\lambda = 1.0$, $c_{2\mathrm{V}} = 1.0$) by a black star.

the ggF HH signal constrained within uncertainties to the one predicted in the SM. The observed (expected) 95% CL upper limit on $\sigma_{\mathrm{VBFHH}}\mathcal{B}(\mathrm{HH} \to \mathrm{b\bar{b}}\gamma\gamma)$ amounts to 1.02(0.94) fb. The limit corresponds to 225 (208) times the SM prediction. This is the most stringent constraint on $\sigma_{\mathrm{VBF\ HH}}\mathcal{B}(\mathrm{HH} \to \mathrm{b\bar{b}}\gamma\gamma)$ to date.

Limits are also set, as a function of $c_{2\mathrm{V}}$, as presented in Fig. 5.26. The observed excluded region corresponds to $c_{2\mathrm{V}} < -1.3$ and $c_{2\mathrm{V}} > 3.5$, while the expected exclusion is $c_{2\mathrm{V}} < -0.9$ and $c_{2\mathrm{V}} > 3.1$. It can be seen in Fig. 5.26 that this analysis is more sensitive to anomalous values of $c_{2\mathrm{V}}$ than to the region around the SM prediction. This is related to the fact that, for anomalous values of $c_{2\mathrm{V}}$, the total cross section is enhanced and the $\widetilde{M}_{\mathrm{X}}$ spectrum is harder as shown in Fig. 5.4 (right). This leads to an increase in the product of signal acceptance and efficiency as well as a more distinct signal topology.

Assuming HH production occurs via the VBF and ggF modes, we set constraints on the κ_λ and $c_{2\mathrm{V}}$ coupling modifiers simultaneously. A 2D negative log-likelihood scan in the (κ_λ, c_2) plane is performed using the 14 HH analysis categories. Fig. 5.27 shows 2D likelihood scans for the observed data and for an Asimov data set assuming all couplings are at their SM values.

We also set upper limits at 95% CL for the twlve BSM benchmark hypotheses defined in Table 5.1. In this fit, the yield of the VBF HH signal is constrained within uncertainties to the ones predicted in the SM. The limits for different BSM hypotheses are shown in Fig. 5.28 (upper). In addition, limits are also calculated as a function of the BSM coupling between two Higgs bosons and two top quarks, c_2, as presented in Fig. 5.28 (lower). The observed excluded region corresponds to $c_2 < -0.6$ and $c_2 > 1.1$, while the expected exclusion is $c_2 < -0.4$ and $c_2 > 0.9$.

5.12 Summary

A search for nonresonant Higgs boson pair production has been presented, where one of the Higgs bosons decays to a pair of bottom quarks and the other to a pair of photons. This search uses proton-proton collision data collected at $\sqrt{s} = 13$ TeV by the CMS experiment at the LHC, corresponding to a total integrated luminosity of 137 fb^{-1}. No significant deviation from the background-only hypothesis is observed. Upper limit at 95% confidence level (CL) on the product of the HH production cross section and the branching ratio fraction into $\mathrm{b\bar{b}}\gamma\gamma$ are extracted for production in the Standard Model (SM) and in several scenarios beyond the SM. The expected upper limit at 95% CL on $\sigma_{\mathrm{HH}}\mathcal{B}(\mathrm{HH} \to \mathrm{b\bar{b}}\gamma\gamma)$ is 0.45 fb, corresponding to about 5.2

times the SM prediction, while the observed upper limit is 0.67 fb, corresponding to 7.7 times the expected value for the SM process. The presented search has the highest sensitivity to the SM HH production to date. Upper limits at 95% CL on the SM HH production cross section are also derived as a function of the Higgs boson self-coupling modifier $\kappa_\lambda \equiv \lambda_{\mathrm{HHH}}/\lambda^{\mathrm{SM}}_{\mathrm{HHH}}$ assuming that the top quark Yukawa coupling is SM-like. The coupling modifier κ_λ is constrained within a range $-3.3 < \kappa_\lambda < 8.5$, while the expected constraint is within a range $-2.5 < \kappa_\lambda < 8.2$ at 95% CL.

This search is combined with an analysis that targets top quark-antiquark associated production of a single Higgs boson decays to a diphoton pair. In the scenario in which the HH signal has the properties predicted by the SM, the coupling modifier κ_λ has been constrained. In addition, a simultaneous constraint on the κ_λ and the modifier of the coupling beteween the Higgs boson and the top quark κ_t is presented when both the HH and single Higgs boson processes are considered as signals.

Limits are also set on the cross section of the nonresonant HH production via vector boson fusion (VBF). The most stringent limit to date is set on the product of the HH VBF production cross section and the branching ratio into $\mathrm{b\bar{b}}\gamma\gamma$. The observed (expected) upper limit at 95% CL amounts to 1.02 (0.94) fb, corresponding to 225 (208) times the SM prediction. Limits are also set as a function of the modifier of the coupling between two vector bosons and two Higgs bosons, $c_{2\mathrm{V}}$. The observed excluded region corresponds to $c_{2\mathrm{V}} < -1.3$ and $c_{2\mathrm{V}} > 3.5$, while the expected exclusion is $c_{2\mathrm{V}} < -0.9$ and $c_{2\mathrm{V}} > 3.1$.

Numerous hypotheses on coupling modifiers beyond the SM have been explored, both in the context of inclusive Higgs boson pair production and for HH production via gluon-gluon fusion and VBF. The production of Higgs boson pairs was also combined with the top quark-antiquark pair associated production of a single Higgs boson. Overall, all the results are consistent with the SM predictions.

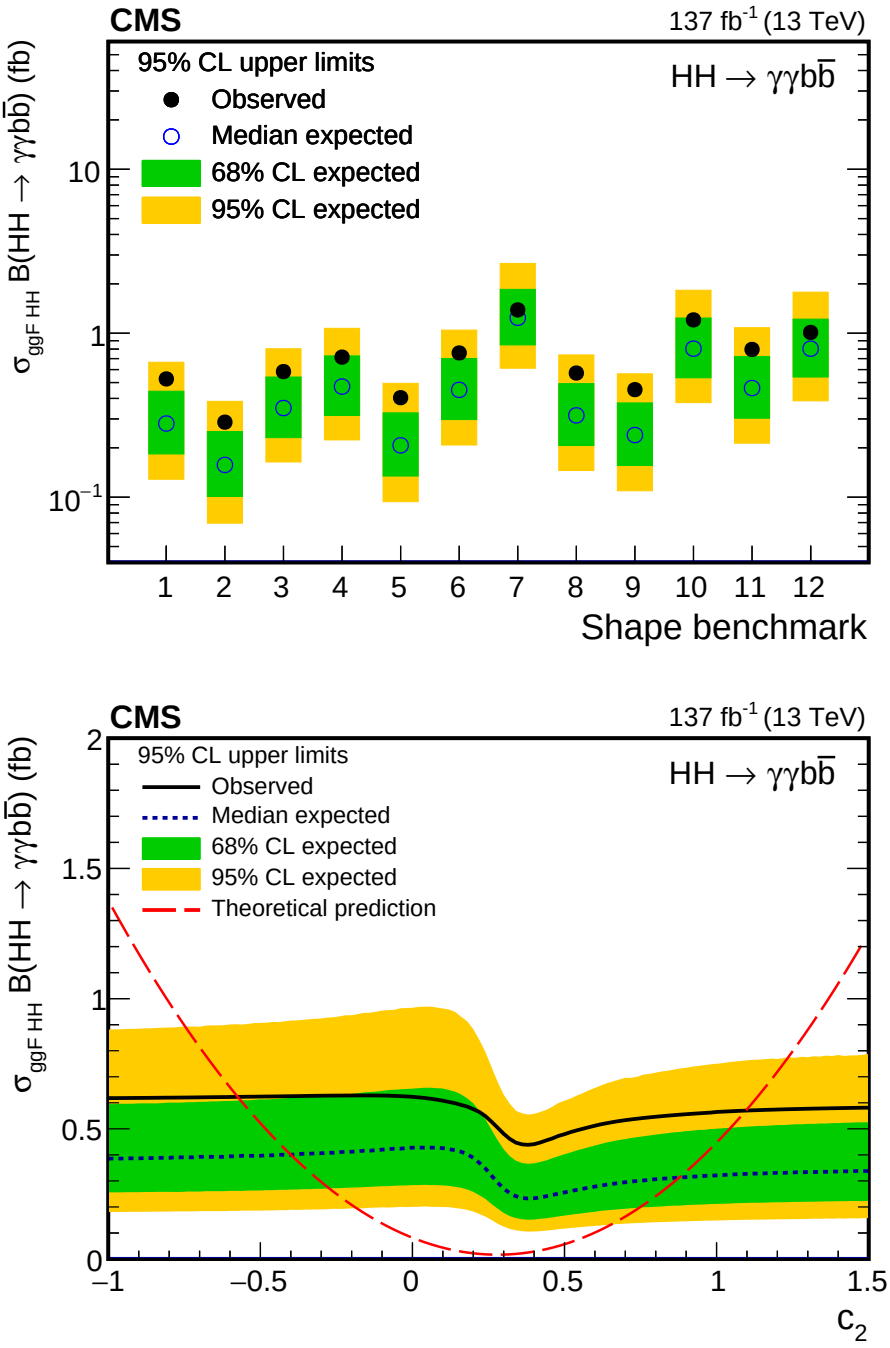


Figure 5.28: Expected and observed 95% CL upper limits on the product of the ggF HH production cross section and $\mathcal{B}(\mathrm{HH} \rightarrow \mathrm{b\bar{b}}\gamma\gamma)$ obtained for different nonresonant benchmark models (defined in Table 5.1) (upper) and BSM coupling c_2 (lower). In this fit, the yield of the VBF HH signal is constrained within uncertainties to the one predicted in the SM. The green and yellow bands represent, respectively, the one and two standard deviation extensions beyond the expected limit. On the lower plot, the long-dashed red line shows the theoretical prediction.

Chapter 6

BOOSTING FUTURE HIGGS SEARCHES WITH BOOSTED $H \to b\bar{b}$ JET IDENTIFICATION BASED ON GRAPH INTERACTION NETWORKS

In this chapter, we introduce a novel jet identification algorithm based on interaction networks to identify high-transverse-momentum Higgs bosons decaying to bottom quark-antiquark pairs and distinguish them from ordinary jets originating from the hadronization of quarks and gluons. The algorithms inputs are features of the reconstructed charged particles in a jet and the secondary vertices associated with them. Describing the jet shower as a combination of particle-to-particle and particle-to-vertex interactions, the model is trained to learn a jet representation on which the classification problem is optimized. The algorithm is trained on simulated samples of realistic LHC collisions, released by the CMS Collaboration on the CERN Open Data Portal. The interaction network achieves a drastic improvement in the identification performance with respect to state-of-the-art algorithms, and can be used to improve the sensitivity of future searches involving Higgs boson decaying into a pair of bottom quarks.

6.1 Introduction

Jets are collimated cascades of particles produced at particle accelerators. Quarks and gluons originating from hadron collisions, such as the proton-proton collisions at the CERN Large Hadron Collider (LHC), generate a cascade of other particles (mainly other quarks or gluons) that then arrange themselves into hadrons. The stable and unstable hadrons' decay products are observed by large particle detectors, reconstructed by algorithms that combine the information from different detector components, and then clustered into jets, using physics-motivated sequential recombination algorithms such as those described in Ref. [57, 143, 144]. Jet identification, or *tagging*, algorithms are designed to identify the nature of the particle that initiated a given cascade, inferring it from the collective features of the particles generated in the cascade.

Traditionally, jet tagging was meant to distinguish three classes of jets: light flavor quarks $q = u, d, s, c$, gluons g, or bottom quarks (b). At the LHC, due to the large collision energy, new jet topologies emerge. When heavy particles, e.g. W, Z, or

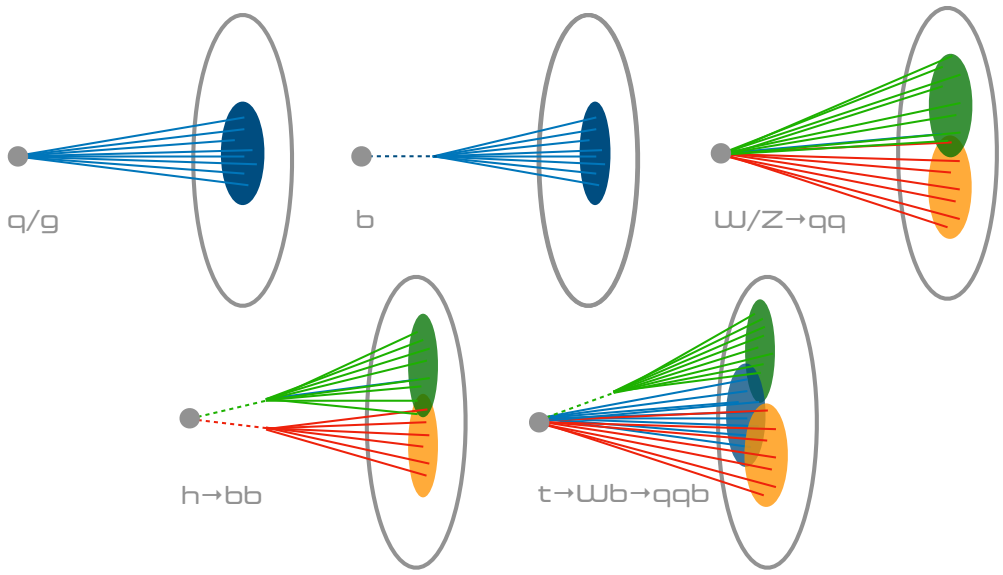


Figure 6.1: Pictorial representation of ordinary quark and gluon jets (top left), b jets (top center), and boosted-jet topologies, emerging from high-p_{T} W and Z bosons (top right), Higgs bosons (bottom left), and top quarks (bottom right) decaying to all-quark final states.

Higgs (H) bosons or the top quark, are produced with large momentum and decay to all-quark final states, the resulting jets are contained in a small solid angle. A single jet emerges from the overlap of two (for bosons) or three (for the top quark) jets, as illustrated in Fig. 6.1. These jets are characterized by a large invariant mass (computed from the sum of the four-momenta of their constituents) and they differ from ordinary quark and gluon jets, due to their peculiar momentum flow around the jet axis.

Several techniques have been proposed to identify these jets by using physics-motivated quantities, collectively referred to as "jet substructure" variables. A review of the different techniques can be found in Ref. [145]. As discussed in the review, approaches based on deep learning (DL) have been extensively investigated (see also Sec. 6.2), processing sets of physics-motivated quantities with dense layers or raw data representations (e.g. jet images or particle feature lists) with more complex architectures (e.g. convolutional or recurrent networks).

While existing DL approaches have been successfully applied to jet tagging, particle jets involve multiple entities with complex interactions that are not easily encoded as images or lists. Graphs provide a natural representation for such relational information. Traditional machine learning methods use feature engineering

and preprocessing to learn from these graphs, which can be time consuming and costly, and may miss important features present in the data. Graph representation learning, including graph convolution networks [146–149] and graph generative models [150, 151], leverages DL to learn directly from graph-structured data. In contrast to other DL methods, graph representation learning can (1) handle irregular grids with non-Euclidean geometry [152], (2) encode physics knowledge via graph construction [153], and (3) introduce relational inductive bias into data-driven learning systems [154]. For example, while convolutional neural networks (CNNs) are powerful classifiers that work extremely well for data represented on a grid [155, 156], geometric DL algorithms, such as graph neural networks (GNNs) [157, 158], are applicable even without an underlying grid structure. Because the data in many scientific domains are not Euclidean, GNNs emerge as a more natural choice.

In this work, we compare the typical performance of some of these approaches to what is achievable with a novel jet identification algorithm based on an interaction network known as JEDI-net. Interaction networks [159] (INs) were designed to decompose complex systems into distinct objects and relations, and reason about their interactions and dynamics. One of the first uses of INs was to predict the evolution of physical systems under the influence of internal and external forces, for example, to emulate the effect of gravitational interactions in n-body systems. The n-body system is represented as a set of objects subject to one-on-one interactions. The n bodies are embedded in a graph and these one-on-one interaction functions, expressed as trainable neural networks, are used to predict the post-interaction status of the n-body system. We study whether this type of network generalizes to a novel context in high energy physics. In particular, we represent a jet as a set of particles, each of which is represented by its momentum and embedded as a vertex in a fully-connected graph. We use neural networks to learn a representation of each one-on-one particle *interaction* [1] in the jet, which we then use to define jet-related high-level features (HLFs). Based on these features, a classifier associates each jet to one of the five categories shown in Fig. 6.1.

For comparison, we consider other classifiers based on different architectures: a dense neural network (DNN) [160] receiving a set of jet-substructure quantities, a convolutional neural network (CNN) [161–163] receiving an image representation

[1] Here, we refer to the abstract message-passing interaction represented by the edges of the graph and not the physical interactions due to quantum chromodynamics, which occur before the jet constituents emerge from the hadronization process.

of the transverse momentum (p_{T}) flow in the jet [2], and a recurrent neural network (RNN) [164–166] with gated recurrent units [167] (GRUs), which process a list of particle features. These models can achieve state-of-the-art performance although they require additional ingredients: the DNN model requires processing the constituent particles to pre-compute HLFs, the GRU model assumes an ordering criterion for the input particle feature list, and the CNN model requires representing the jet as a rectangular, regular, pixelated image. Any of these aspects can be handled in a reasonable way (e.g. one can use a jet clustering metric to order the particles), sometimes sacrificing some detector performance (e.g., with coarser image pixels than realistic tracking angular resolution, in the case of many models based on CNN). It is then worth exploring alternative solutions that could reach state-of-the-art performance without making these assumptions. In particular, it is interesting to consider architectures that directly takes as input jet constituents and are invariant for their permutation. This motivated the study of jet taggers based on recursive [168], graph networks [169, 170], and energy flow networks [171]. In this context, we aim to investigate the potential of INs in jet identification.

This chapter is organized as follows: Sec. 6.2 provides a list of relevant works in the context of jet tagging algorithms. In Sec. 6.3, we describe the datasets used for the two studies presented in this chapter: (1) the JEDI-net, which is the base model of the interaction network used for jet tagging, and (2) the modified JEDI-net for boosted $\mathrm{H} \to \mathrm{b\bar{b}}$ jet classification, where we extend the base JEDI-net model by incorporating the secondary vertex interaction. Details of the base JEDI-net model are described in Sec. 6.4, including the model architecture, its performance in discriminating between jets originating from gluons, light quarks, W and Z bosons, and top quarks. This section also characterizes the information learned by JEDI-net and compares resource consumption between JEDI-net and other popular deep-learning based models. Sec. 6.5 described the model for boosted $\mathrm{H} \to \mathrm{b\bar{b}}$ jet tagging based on JEDI-net, with additional information from the secondary vertices of the b quarks. As many Higgs searches rely on the jet mass spectrum, it's important to keep the jet tagging score uncorrelated from the jet mass, which is the focus of Sec. 6.6. Sec. 6.7 describes our reproduction of the `DeepDoubleB` model, which is

[2] We use a Cartesian coordinate system with the z axis oriented along the beam axis, the x axis on the horizontal plane, and the y axis oriented upward. The x and y axes define the transverse plane, while the z axis identifies the longitudinal direction. The azimuthal angle ϕ is computed from the x axis. The polar angle θ is used to compute the pseudorapidity $\eta = -\log(\tan(\theta/2))$. We use natural units such that $c = \hbar = 1$ and we express energy in units of electronvolt (eV) and its prefix multipliers.

the state-of-the-art algorithm used in CMS, for benchmarking purpose. Results are presented in Sec. 6.8. Sec. 6.9 summarizes the chapter.

6.2 Related work

Jet tagging is one of the most popular LHC-related tasks to which DL solutions have been applied. Several classification algorithms have been studied in the context of jet tagging at the LHC [172–179] using DNNs, CNNs, or physics-inspired architectures. Recurrent and recursive layers have been used to construct jet classifiers starting from a list of reconstructed particle momenta [168–170]. Recently, these different approaches, applied to the specific case of top quark jet identification, have been compared in Ref. [180]. While many of these studies focus on data analysis, work is underway to apply these algorithms in the early stages of LHC real-time event processing, i.e. the trigger system. For example, Ref. [181] focuses on converting these models into firmware for field programmable gate arrays (FPGAs) optimized for low latency (less than 1 μs). If successful, such a program could allow for a more resource-efficient and effective event selection for future LHC runs.

Graph neural networks have also been considered as jet tagging algorithms [182, 183] as a way to circumvent the sparsity of image-based representations of jets. These approaches demonstrate remarkable categorization performance. Motivated by the early results of Ref. [182], graph networks have been also applied to other high energy physics tasks, such as event topology classification [184, 185], particle tracking in a collider detector [186], pileup subtraction at the LHC [187], and particle reconstruction in irregular calorimeters [188].

6.3 Dataset description

Dataset for JEDI-net study

This study is based on a data set consisting of simulated jets with an energy of $p_{\mathrm{T}} \approx 1$ TeV, originating from light quarks q, gluons g, W and Z bosons, and top quarks produced in $\sqrt{s} = 13$ TeV proton-proton collisions. The data set was created using the configuration and parametric description of an LHC detector described in Ref. [181, 189], and is available on the Zenodo platform [190–193].

Jets are clustered from individual reconstructed particles, using the anti-k_{T} algorithm [57, 194] with jet-size parameter $R = 0.8$. Three different jet representations are considered:

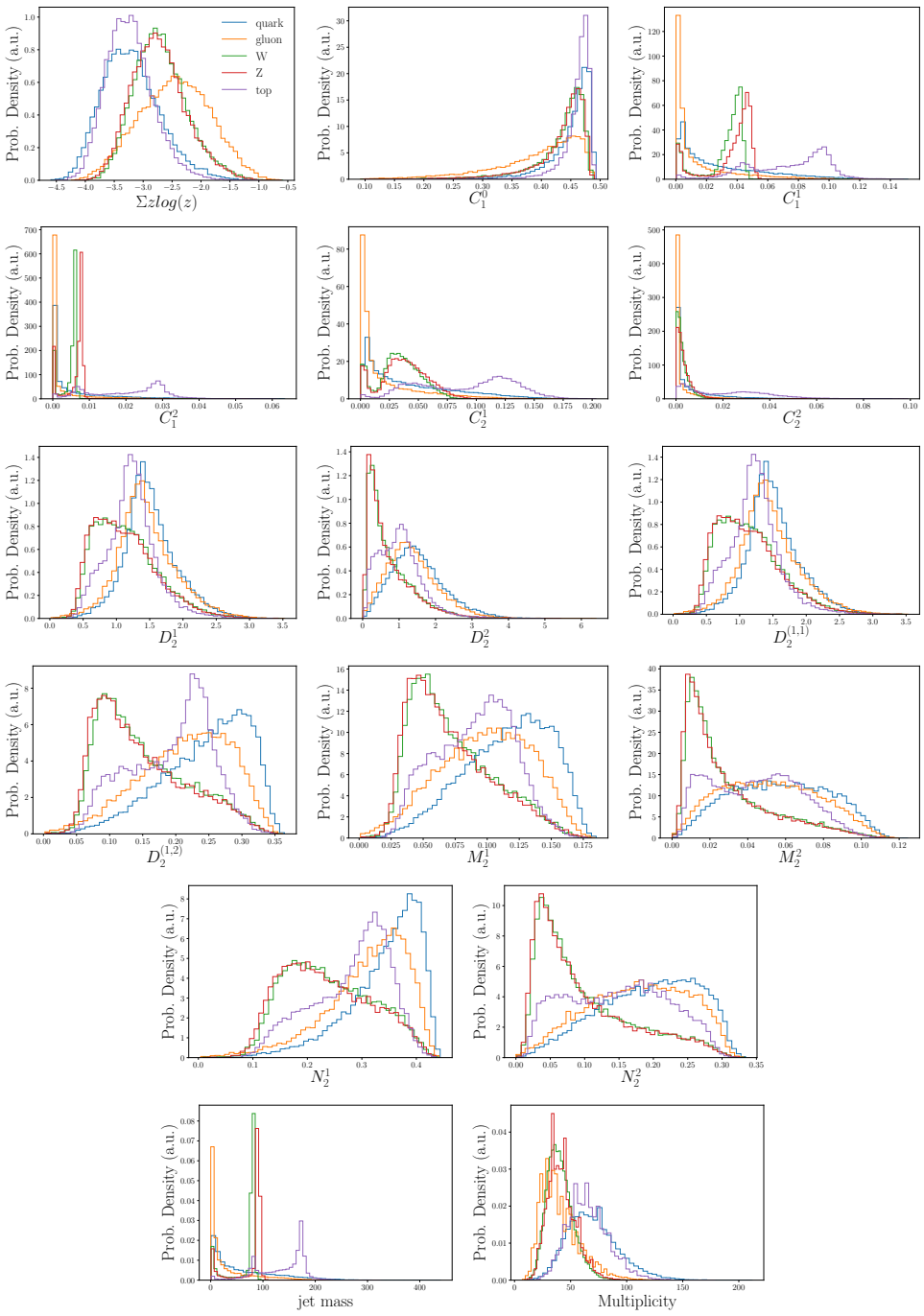


Figure 6.2: Distributions of the 16 high-level features used in this study, described in Ref. [181].

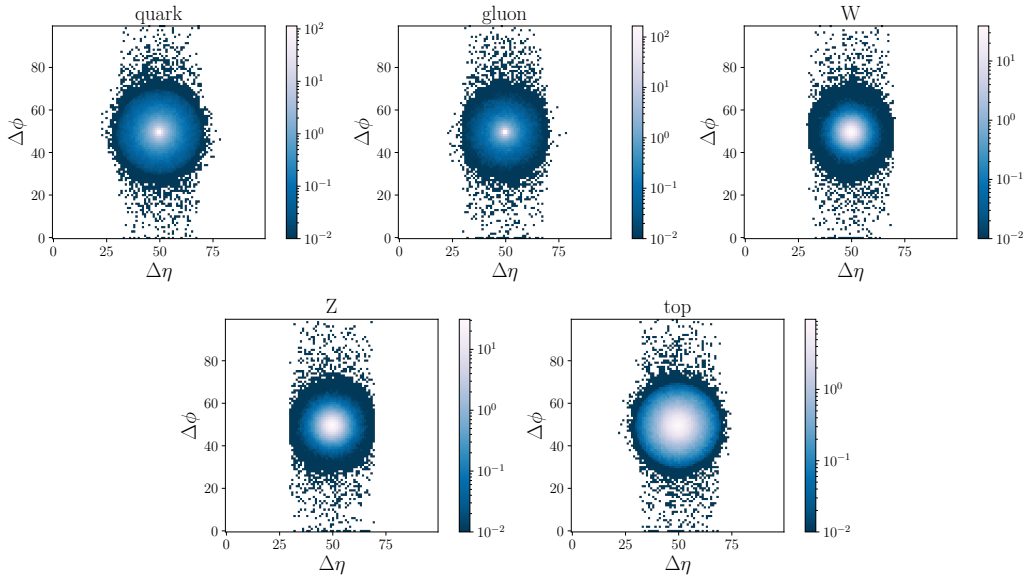


Figure 6.3: Average 100×100 images for the five jet classes considered in this study: q (top left), g (top center), W (top right), Z (bottom left), and top jets (bottom right). The temperature map represents the amount of p_{T} collected in each cell of the image, measured in GeV and computed from the scalar sum of the p_{T} of the particles pointing to each cell.

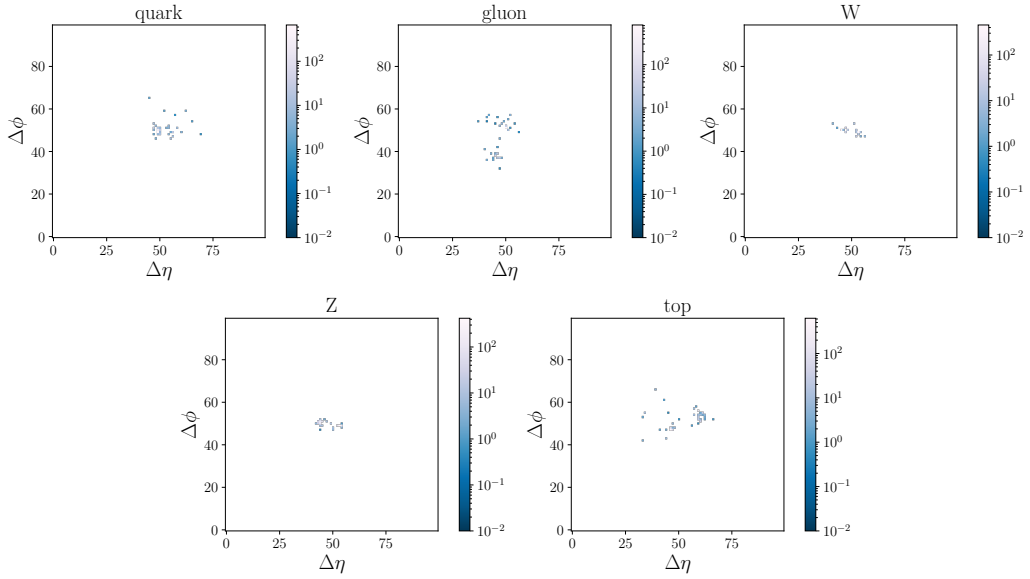


Figure 6.4: Example of 100×100 images for the five jet classes considered in this study: q (top-left), g (top-right), W (center-left), Z (center-right), and top jets (bottom). The temperature map represents the amount of p_{T} collected in each cell of the image, measured in GeV and computed from the scalar sum of the p_{T} of the particles pointing to each cell.

- A list of 16 HLFs, described in Ref. [181], given as input to a DNN. The 16 distributions are shown in Fig. 6.2 for the five jet classes.

- An image representation of the jet, derived by considering a square with pseudorapidity and azimuthal distances $\Delta\eta = \Delta\phi = 2R$, centered along the jet axis. The image is binned into 100×100 pixels. Such a pixel size is comparable to the cell of a typical LHC electromagnetic calorimeter, but much coarser than the typical angular resolution of a tracking device for the p_{T} values relevant to this task. Each pixel is filled with the scalar sum of the p_{T} of the particles in that region. These images are obtained by considering the 150 highest-p_{T} constituents for each jet. This jet representation is used to train a CNN classifier. The average jet images for the five jet classes are shown in Fig. 6.3. For comparison, a randomly chosen set of images is shown in Fig. 6.4.

- A constituent list for up to 150 particles, in which each particle is represented by 16 features, computed from the particle four-momenta: the three Cartesian coordinates of the momentum (p_x, p_y, and p_z), the absolute energy E, p_{T}, the pseudorapidity η, the azimuthal angle ϕ, the distance $\Delta R = \sqrt{\Delta\eta^2 + \Delta\phi^2}$ from the jet center, the relative energy $E^{\mathrm{rel}} = E^{\mathrm{particle}}/E^{\mathrm{jet}}$ and relative transverse momentum $p_{\mathrm{T}}^{\mathrm{rel}} = p_{\mathrm{T}}^{\mathrm{particle}}/p_{\mathrm{T}}^{\mathrm{jet}}$ defined as the ratio of the particle quantity and the jet quantity, the relative coordinates $\eta^{\mathrm{rel}} = \eta^{\mathrm{particle}} - \eta^{\mathrm{jet}}$ and $\phi^{\mathrm{rel}} = \phi^{\mathrm{particle}} - \phi^{\mathrm{jet}}$ defined with respect to the jet axis, $\cos\theta$ and $\cos\theta^{\mathrm{rel}}$ where $\theta^{\mathrm{rel}} = \theta^{\mathrm{particle}} - \theta^{\mathrm{jet}}$ is defined with respect to the jet axis, and the relative η and ϕ coordinates of the particle after applying a proper Lorentz transformation (rotation) as described in Ref. [195]. Whenever less than 150 particles are reconstructed, the list is filled with zeros. The distributions of these features considering the 150 highest-p_{T} particles in the jet are shown in Fig. 6.5 for the five jet categories. This jet representation is used for a RNN with a GRU layer and for JEDI-net.

Dataset for the study of modified JEDI-net for boosted $\mathrm{H} \to \mathrm{b\bar{b}}$ jet identification

This study uses the CMS open data and simulation, which are available from the CERN Open Data Portal [196], including releases of 2010, 2011, and 2012 CMS collision data as well as 2011, 2012, and 2016 CMS simulated data.

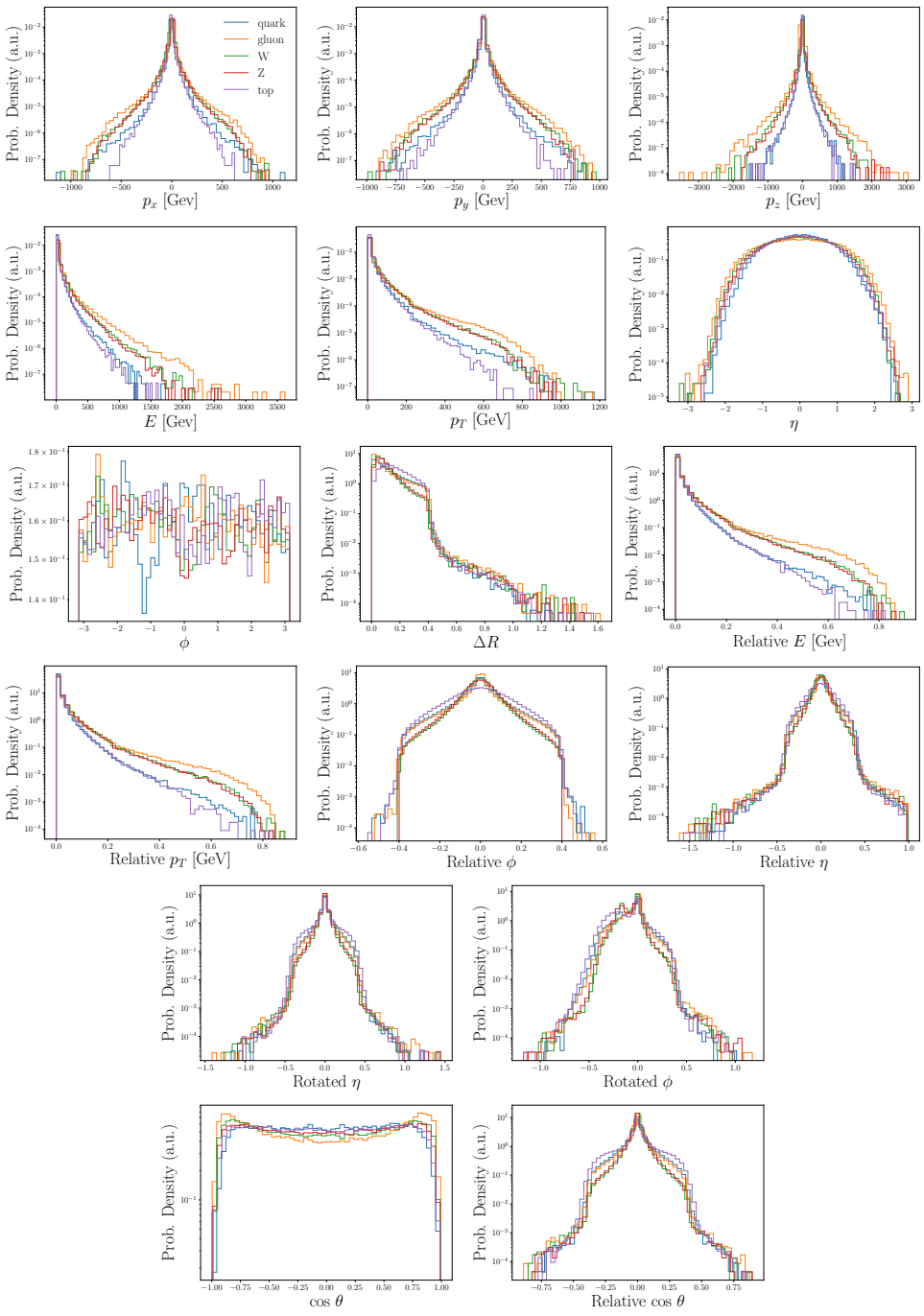


Figure 6.5: Distributions of kinematic features described in the text for the 150 highest-p_{T} particles in each jet.

Samples of $H \to b\bar{b}$ jets are available from simulated events containing Randall-Sundrum gravitons [197] decaying to two Higgs bosons, which subsequently decay to $b\bar{b}$ pairs. The event generation was done by the CMS Collaboration with MADGRAPH5_aMCATNLO 2.2.2 at leading order, with graviton masses ranging between 0.6 and 4.5 TeV. Generation of this process enables better sampling of events with large Higgs boson p_T. The main source of background originates from multijet events. The background dataset was generated with PYTHIA 8.205 [103] in different bins of the average p_T of the final-state partons ($\hat{p}_T$). The parton showering and hadronization was performed with PYTHIA 8.205 [103], using the CMS underlying event tune CUETP8M1 [107] and the NNPDF 2.3 [198] parton distribution functions. Pileup interactions are modeled by overlaying each simulated event with additional minimum bias collisions, also generated with PYTHIA 8.205. The CMS detector response is modeled by GEANT4 [111].

The outcome of the default CMS reconstruction workflow is provided in the open simulation [199]. In particular, particle candidates are reconstructed using the particle-flow (PF) algorithm [200]. Charged particles from pileup interactions are removed using the CHS algorithm. Jets are clustered from the remaining reconstructed particles using the anti-k_T algorithm [57, 194] with a jet-size parameter $R = 0.8$. The standard CMS jet energy corrections are applied to the jets. In order to remove soft, wide-angle radiation from the jet, the soft-drop (SD) algorithm [201, 202] is applied, with angular exponent $\beta = 0$, soft cutoff threshold $z_{cut} < 0.1$, and characteristic radius $R_0 = 0.8$ [203]. The SD mass (m_{SD}) is then computed from the four-momenta of the remaining constituents.

A signal $H \to b\bar{b}$ jet is defined as a jet geometrically matched to the generator-level Higgs boson and both b quark daughters. Jets from QCD multijet events are used to define a sample of fake $H \to b\bar{b}$ candidates.

The dataset is reduced by requiring the AK8 jets to have $300 < p_T < 2400$ GeV, $|\eta| < 2.4$, and $40 < m_{SD} < 200$ GeV. After this reduction, the dataset consists of 3.9 million $H \to b\bar{b}$ jets and 1.9 million inclusive QCD jets. Charged particles are required to have $p_T > 0.95$ GeV and reconstructed secondary vertices (SVs) are associated with the AK8 jet using $\Delta R = \sqrt{\Delta\phi^2 + \Delta\eta^2} < 0.8$. The dataset is divided into blocks of features, referring to different objects. Different blocks are used as input by the models described in the rest of the paper.

The IN uses 30 features related to charged particles (see Table 6.1). The IN also uses 14 SV features listed in Table 6.2. The DDB tagger [204] uses a subset of the above

features (8 features for each particle and 2 features for each SV), chosen to minimize the correlation with the jet mass. In addition, the DDB tagger uses 27 high-level features (HLF) listed in Table 6.3 and first used in a previous version of the algorithm, described in Ref. [59]. To isolate the effects of the different architecture, the DDB+ tagger uses the same inputs as the IN tagger, while retaining the architecture of the DDB tagger. The charged particles (SVs) are sorted in descending order of the 2D impact parameter significance (2D flight distance significance) and only the first 60 (5) are considered.

Table 6.1: Charged particle features. The IN and DDB+ models use all of the features, while DDB algorithm uses the subset of features indicated in bold.

Variable	Description
`track_ptrel`	p_{T} of the charged particle divided by the p_{T} of the AK8 jet
`track_erel`	Energy of the charged particle divided by the energy of the AK8 jet
`track_phirel`	$\Delta\phi$ between the charged particle and the AK8 jet axis
`track_etarel`	$\Delta\eta$ between the charged particle and the AK8 jet axis
`track_deltaR`	ΔR between the charged particle and the AK8 jet axis
`track_drminsv`	ΔR between the associated SVs and the charged particle
`track_drsubjet1`	ΔR between the charged particle and the first soft drop subjet
`track_drsubjet2`	ΔR between the charged particle and the second soft drop subjet
`track_dz`	Longitudinal impact parameter of the track, defined as the distance of closest approach of the track trajectory to the PV projected on to the z direction
`track_dzsig`	Longitudinal impact parameter significance of the track

track_dxy	Transverse (2D) impact parameter of the track, defined as the distance of closest approach of the track trajectory to the beam line in the transverse plane to the beam
track_dxysig	Transverse (2D) impact parameter of the track
track_normchi2	Normalized χ^2 of the track fit
track_quality	Track quality: undefQuality=-1, loose=0, tight=1, highPurity=2, confirmed=3, looseSetWithPV=5, highPuritySetWithPV=6, discarded=7, qualitySize=8
track_dptdpt	Track covariance matrix entry (p_T, p_T)
track_detadeta	Track covariance matrix entry (η, η)
track_dphidphi	Track covariance matrix entry (ϕ, ϕ)
track_dxydxy	Track covariance matrix entry (d_{xy}, d_{xy})
track_dzdz	Track covariance matrix entry (d_z, d_z)
track_dxydz	Track covariance matrix entry (d_{xy}, d_z)
track_dphidz	Track covariance matrix entry (d_ϕ, d_z)
track_dlambdadz	Track covariance matrix entry (λ, d_z)
trackBTag_EtaRel	$\Delta\eta$ between the track and the AK8 jet axis
trackBTag_PtRatio	Component of track momentum perpendicular to the AK8 jet axis, normalized to the track momentum
trackBTag_PParRatio	Component of track momentum parallel to the AK8 jet axis, normalized to the track momentum
trackBTag_Sip2dVal	Transverse (2D) signed impact parameter of the track
trackBTag_Sip2dSig	Transverse (2D) signed impact parameter significance of the track
trackBTag_Sip3dVal	3D signed impact parameter of the track
trackBTag_Sip3dSig	3D signed impact parameter significance of the track
trackBTag_JetDistVal	Minimum track approach distance to the AK8 jet axis

Table 6.2: Secondary vertex features. The IN and DDB+ models use all of the features, while the DDB algorithm uses the subset of features indicated in bold.

Variable	Description
sv_ptrel	p_T of the SV divided by the p_T of the AK8 jet
sv_erel	Energy of the SV divided by the energy of the AK8 jet
sv_phirel	$\Delta\phi$ between the SV and the AK8 jet axis
sv_etarel	$\Delta\eta$ between the SV and the AK8 jet axis
sv_deltaR	ΔR between the SV and the AK8 jet axis
sv_pt	p_T of the SV
sv_mass	Mass of the SV
sv_ntracks	Number of tracks associated with the SV
sv_normchi2	Normalized χ^2 of the SV fit
sv_costhetasvpv	$\cos\theta$ between the SV and the PV
sv_dxy	Transverse (2D) flight distance of the SV
sv_dxysig	Transverse (2D) flight distance significance of the SV
sv_d3d	3D flight distance of the SV
sv_d3dsig	3D flight distance significance of the SV

Table 6.3: High-level features used by the DDB algorithm.

Variable	Description
fj_jetNTracks	Number of tracks associated with the AK8 jet
fj_nSV	Number of SVs associated with the AK8 jet ($\Delta R < 0.7$)
fj_tau0_trackEtaRel_0	Smallest track $\Delta\eta$ relative to the jet axis, associated to the first N-subjettiness axis
fj_tau0_trackEtaRel_1	Second smallest track $\Delta\eta$ relative to the jet axis, associated to the first N-subjettiness axis
fj_tau0_trackEtaRel_2	Third smallest track $\Delta\eta$ relative to the jet axis, associated to the first N-subjettiness axis

fj_tau1_trackEtaRel_0	Smallest track $\Delta\eta$ relative to the jet axis, associated to the second N-subjettiness axis
fj_tau1_trackEtaRel_1	Second smallest track $\Delta\eta$ relative to the jet axis, associated to the second N-subjettiness axis
fj_tau1_trackEtaRel_2	Third smallest track $\Delta\eta$ relative to the jet axis, associated to the second N-subjettiness axis
fj_tau_flightDistance2dSig_0	Transverse (2D) flight distance significance between the PV and the SV with the smallest uncertainty on the 3D flight distance associated to the first N-subjettiness axis
fj_tau_flightDistance2dSig_1	Transverse (2D) flight distance significance between the PV and the SV with the smallest uncertainty on the 3D flight distance associated to the second N-subjettiness axis
fj_tau_vertexDeltaR_0	ΔR between the first N-subjettiness axis and SV direction
fj_tau_vertexEnergyRatio_0	SV energy ratio for the first N-subjettiness axis, defined as the total energy of all SVs associated with the first N-subjettiness axis divided by the total energy of all the tracks associated with the AK8 jet that are consistent with the PV
fj_tau_vertexEnergyRatio_1	SV energy ratio for the second N-subjettiness axis
fj_tau_vertexMass_0	SV mass for the first N-subjettiness axis, defined as the invariant mass of all tracks from SVs associated with the first N-subjettiness axis

fj_tau_vertexMass_1	SV mass for the second N-subjettiness axis
fj_trackSip2dSigAboveBottom_0	Track 2D signed impact parameter significance of the first track lifting the combined invariant mass of the tracks above the b hadron threshold mass (5.2 GeV)
fj_trackSip2dSigAboveBottom_1	Track 2D signed impact parameter significance of the second track lifting the combined invariant mass of the tracks above the b hadron threshold mass (5.2 GeV)
fj_trackSip2dSigAboveCharm_0	Track 2D signed impact parameter significance of the first track lifting the combined invariant mass of the tracks above the c hadron threshold mass (1.5 GeV)
fj_trackSipdSig_0	Largest track 3D signed impact parameter significance
fj_trackSipdSig_1	Second largest track 3D signed impact parameter significance
fj_trackSipdSig_2	Third largest track 3D signed impact parameter significance
fj_trackSipdSig_3	Fourth largest track 3D signed impact parameter significance
fj_trackSipdSig_0_0	Largest track 3D signed impact parameter significance associated to the first N-subjettiness axis
fj_trackSipdSig_0_1	Second largest track 3D signed impact parameter significance associated to the first N-subjettiness axis
fj_trackSipdSig_1_0	Largest track 3D signed impact parameter significance associated to the second N-subjettiness axis

fj_trackSipdSig_1_1	Second largest track 3D signed impact parameter significance associated to the second N-subjettiness axis
fj_z_ratio	z ratio variable as defined in Ref. [59]

Table 6.4: Additional features for charged or neutral particles. The all-particle IN model uses these features.

Variable	Description
pfcand_ptrel	p_{T} of the charged or neutral particle divided by the p_{T} of the AK8 jet
pfcand_erel	Energy of the charged or neutral particle divided by the energy of the AK8 jet
pfcand_phirel	$\Delta\phi$ between the charged or neutral particle and the AK8 jet axis
pfcand_etarel	$\Delta\eta$ between the charged or neutral particle and the AK8 jet axis
pfcand_deltaR	ΔR between the charged or neutral particle and the AK8 jet axis
pfcand_puppiw	Pileup per particle identification (PUPPI) weight [205] for the charged or neutral particle
pfcand_drminsv	ΔR between the associated SVs and the charged or netural particle
pfcand_drsubjet1	ΔR between the charged or neutral particle and the first soft drop subjet
pfcand_drsubjet2	ΔR between the charged or neutral particle and the second soft drop subjet
pfcand_hcalFrac	Fraction of energy of the charged or neutral particle deposited in the hadron calorimeter

6.4 JEDI-net: Jet identification algorithm based on interaction networks

We apply an interaction network (IN) [159] architecture to learn a representation of a given input graph (the set of constituents in a jet) and use it to accomplish a classification task (tagging the jet), where we name it JEDI-net. One can see the

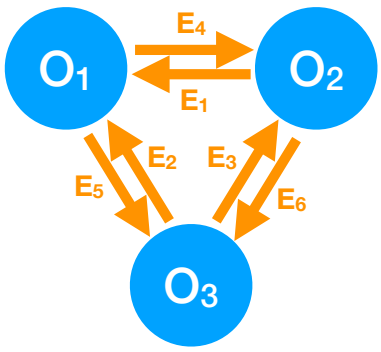


Figure 6.6: An example graph with three fully connected vertices and the corresponding six edges.

IN architecture as a processing algorithm to learn a new representation of the initial input. This is done replacing a set of input features, describing each individual vertex of the graph, with a set of engineered features, specific of each vertex, but whose values depend on the connection between the vertices in the graph.

The starting point consists of building a graph for each input jet. The N_O particles in the jet are represented by the vertices of the graph, fully interconnected through directional edges, for a total of $N_E = N_O \times (N_O - 1)$ edges. An example is shown in Fig. 6.6 for the case of a three-vertex graph. The vertices and edges are labeled for practical reasons, but the network architecture ensures that the labeling convention plays no role in creating the new representation.

Once the graph is built, a receiving matrix (R_R) and a sending matrix (R_S) are defined. Both matrices have dimensions $N_O \times N_E$. The element $(R_R)_{ij}$ is set to 1 when the i^{th} vertex receives the j^{th} edge and is 0 otherwise. Similarly, the element $(R_S)_{ij}$ is set to 1 when the i^{th} vertex sends the j^{th} edge and is 0 otherwise. In the case of the graph of Fig. 6.6, the two matrices take the form:

$$R_S = \begin{array}{c} \\ O_1 \\ O_2 \\ O_3 \end{array} \begin{array}{c} \begin{array}{cccccc} E_1 & E_2 & E_3 & E_4 & E_5 & E_6 \end{array} \\ \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 \end{pmatrix} \end{array} \tag{6.1}$$

$$R_R = \begin{array}{c} \\ O_1 \\ O_2 \\ O_3 \end{array} \begin{array}{c} \begin{array}{cccccc} E_1 & E_2 & E_3 & E_4 & E_5 & E_6 \end{array} \\ \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} \end{array} . \tag{6.2}$$

The input particle features are represented by an input matrix I. Each column of the matrix corresponds to one of the graph vertices, while the rows correspond to the P features used to represent each vertex. In our case, the vertices are the particles inside the jet, each represented by its array of features (i.e., the 16 features shown in Fig. 6.5). Therefore, the I matrix has dimensions $P \times N_O$.

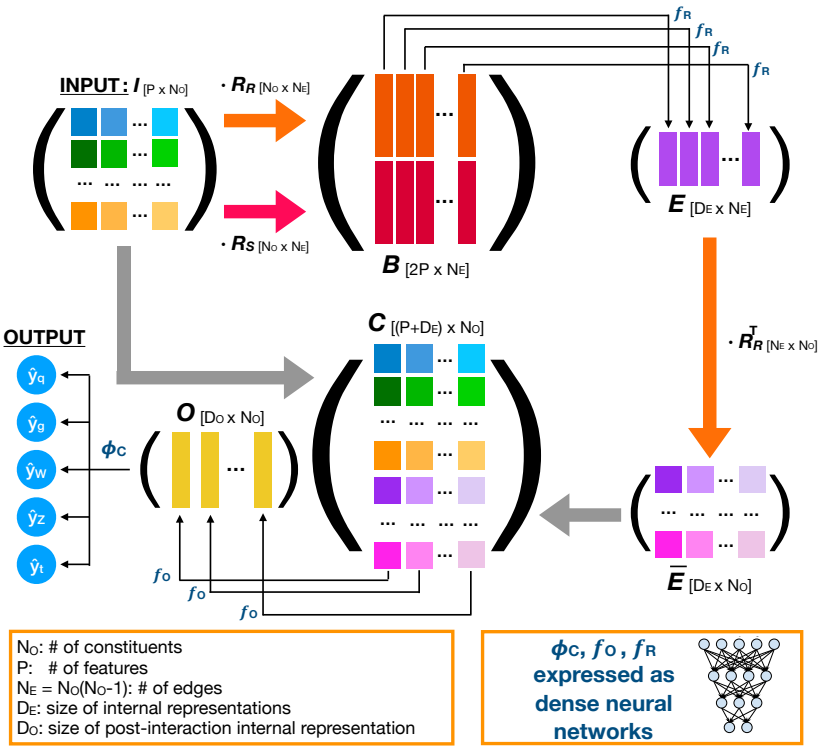


Figure 6.7: A flowchart illustrating the interaction network scheme.

The I matrix is processed by the IN in a series of steps, represented in Fig. 6.7. The I matrix is multiplied by the R_R and R_S matrices and the two resulting matrices are then concatenated to form the B matrix, having dimension $2P \times N_E$:

$$B = \begin{pmatrix} I \times R_R \\ I \times R_S \end{pmatrix}. \tag{6.3}$$

Each column of the B matrix represents an edge, i.e. a particle-to-particle interaction. The $2P$ elements of each column are the features of the sending and receiving vertices for that edge. Using this information, a D_E-dimensional hidden representation of the interaction edge is created through a trainable function $f_R : \mathbb{R}^{2P} \mapsto \mathbb{R}^{D_E}$. This gives a matrix E with dimensions $D_E \times N_E$. The cumulative effects of the interactions received by a given vertex are gathered by summing the D_E hidden

features over the edges arriving to it. This is done by computing $\overline{E} = ER_R^\top$ with dimensions $D_E \times N_O$, which is then appended to the initial input matrix I:

$$C = \begin{pmatrix} I \\ \overline{E} \end{pmatrix}. \tag{6.4}$$

At this stage, each column of the C matrix represents a constituent in the jet, expressed as a $(P + D_E)$-dimensional feature vector, containing the P input features and the D_E hidden features representing the combined effect of the interactions with all the connected particles. A trainable function $f_O : \mathbb{R}^{P+D_E} \mapsto \mathbb{R}^{D_O}$ is used to build a post-interaction representation of each jet constituent. The function f_O is applied to each column of C to build the post-interaction matrix O with dimensions $D_O \times N_O$.

A final classifier ϕ_C takes as input the elements of the O matrix and returns the probability for that jet to belong to each of the five categories. This is done in two ways: (i) in one case, we define the quantities $\overline{O}_i = \sum_j O_{ij}$, where j is the index of the vertex in the graph (the particle, in our case), and the $i \in [0, D_E]$ index runs across the D_E outputs of the f_O function. The $\overline{O}$ quantities are used as input to $\phi_C : \mathbb{R}^{D_O} \mapsto \mathbb{R}^N$. This choice allows to preserve the independence of the architecture on the labeling convention adopted to build the I, R_R, and R_S matrices, at the cost of losing some discriminating information in the summation. (ii) Alternatively, the ϕ_C matrix is defined directly from the $D_O \times N_O$ elements of the O matrix, flattened into a one-dimensional array. The full information from O is preserved, but ϕ_C assumes an ordering of the N_O input objects. In our case, we rank the input particles in descending order by p_T.

The trainable functions f_O, f_R, and ϕ_C consist of three DNNs. Each of them has two hidden layers, the first (second) having N_n^1 ($N_n^2 = \lfloor N_n^1/2 \rfloor$) neurons. The model is implemented in PyTorch [206] and trained using an NVIDIA GTX1080 GPU. The training (validation) data set consists of 630,000 (240,000) examples, while 10,000 events are used for testing purposes.

The architecture of the three trainable functions is determined by minimizing the loss function through a Bayesian optimization, using the GpyOpt library [207], based on Gpy [208]. We consider the following hyperparameters:

- The number of output neurons of the f_R network, D_E (between 4 and 14).
- The number of output neurons of the f_O network, D_O (between 4 and 14).

- The number of neurons N_n^1 in the first hidden layer of the f_O, f_R, and ϕ_C network (between 5 and 50).
- The activation function for the hidden and output layers of the f_R network: ReLU [209], ELU [210], or SELU [211] functions.
- The activation function for the hidden and output layers of the f_O network: ReLU, ELU, or SELU.
- The activation function for the hidden layers of the ϕ_C network: ReLU, ELU, or SELU.
- The optimizer algorithm: Adam [212] or AdaDelta [213].

In addition, the output neurons of the ϕ_C network are activated by a softmax function. A learning rate of 10^{-4} is used. For a given network architecture, the network parameters are optimized by minimizing the categorical cross entropy. The Bayesian optimization is repeated four times. In each case, the input particles are ordered by descending p_{T} value and the first 30, 50, 100, or 150 particles are considered. The parameter optimization is performed on the training data set, while the loss for the Bayesian optimization is estimated on the validation data set.

Tables 6.6 and 6.5 summarize the result of the Bayesian optimization for the JEDI-net architecture with and without the sum over the columns of the O matrix, respectively. The best result of each case, highlighted in bold, is used as a reference for the rest of the section.

Table 6.5: Optimal JEDI-net hyperparameter setting for different input data sets, when the summed $\overline{O}_i$ quantities are given as input to the ϕ_C network. The best result, obtained when considering up to 150 particles per jet, is highlighted in bold.

Hyperparameter	Number of jet constituents			
	30	50	100	**150**
N_n^1	6	50	30	**50**
D_E	8	12	4	**14**
D_O	6	14	4	**10**
f_R activation	ReLU	ReLU	SELU	**SELU**
f_O activation	ELU	ReLU	ReLU	**SELU**
ϕ_C activation	ELU	SELU	SELU	**SELU**
Optimizer	Adam	Adam	Adam	**Adam**
Optimized loss	0.84	0.58	0.62	**0.55**

Table 6.6: Optimal JEDI-net hyperparameter setting for different input data sets, when all the O_{ij} elements are given as input to the ϕ_C network. The best result, obtained when considering up to 100 particles per jet, is highlighted in bold.

Hyperparameter	Number of jet constituents			
	30	50	**100**	150
N_n^1	50	50	**30**	10
D_E	12	12	**10**	4
D_O	6	14	**10**	14
f_R activation	ReLU	ELU	**ELU**	SELU
f_O activation	SELU	SELU	**ELU**	SELU
ϕ_C activation	SELU	ELU	**ELU**	SELU
Optimizer	Adam	Adam	**Adam**	Adam
Optimized loss	0.63	0.57	**0.56**	0.62

For comparison, three alternative models are trained on the three different representations of the same data set described in Sec. 6.3: a DNN model taking as input a list of HLFs, a CNN model processing jet images, and a recurrent model applying GRUs on the same input list used for JEDI-net. The three benchmark models are optimized through a Bayesian optimization procedure, as done for the INs.

Results

Figure 6.8 shows the receiver operating characteristic (ROC) curves obtained for the optimized JEDI-net tagger in each of the five jet categories, compared to the corresponding curves for the DNN, CNN, and GRU alternative models. The curves are derived by fixing the network architectures to the optimal values based on Table 6.6 and performing a k-fold cross-validation training, with $k = 10$. The solid lines represent the average ROC curve, while the shaded bands quantify the ± 1 RMS dispersion. The area under the curve (AUC) values, reported in the figure, allow for a comparison of the performance of the different taggers.

The algorithm's tagging performance is quantified computing the true positive rate (TPR) values for two given reference false positive rate (FPR) values (10% and 1%). The comparison of the TPR values gives an assessment of the tagging performance in a realistic use case, typical of an LHC analysis. Tables 6.7 shows the corresponding FPR values for the optimized JEDI-net taggers, compared to the corresponding values for the benchmark models. The largest TPR value for each class is highlighted in bold. As shown in Fig. 6.8 and Table 6.7, the two JEDI-net models outperform the other architectures in almost all cases. The only notable exception is the tight

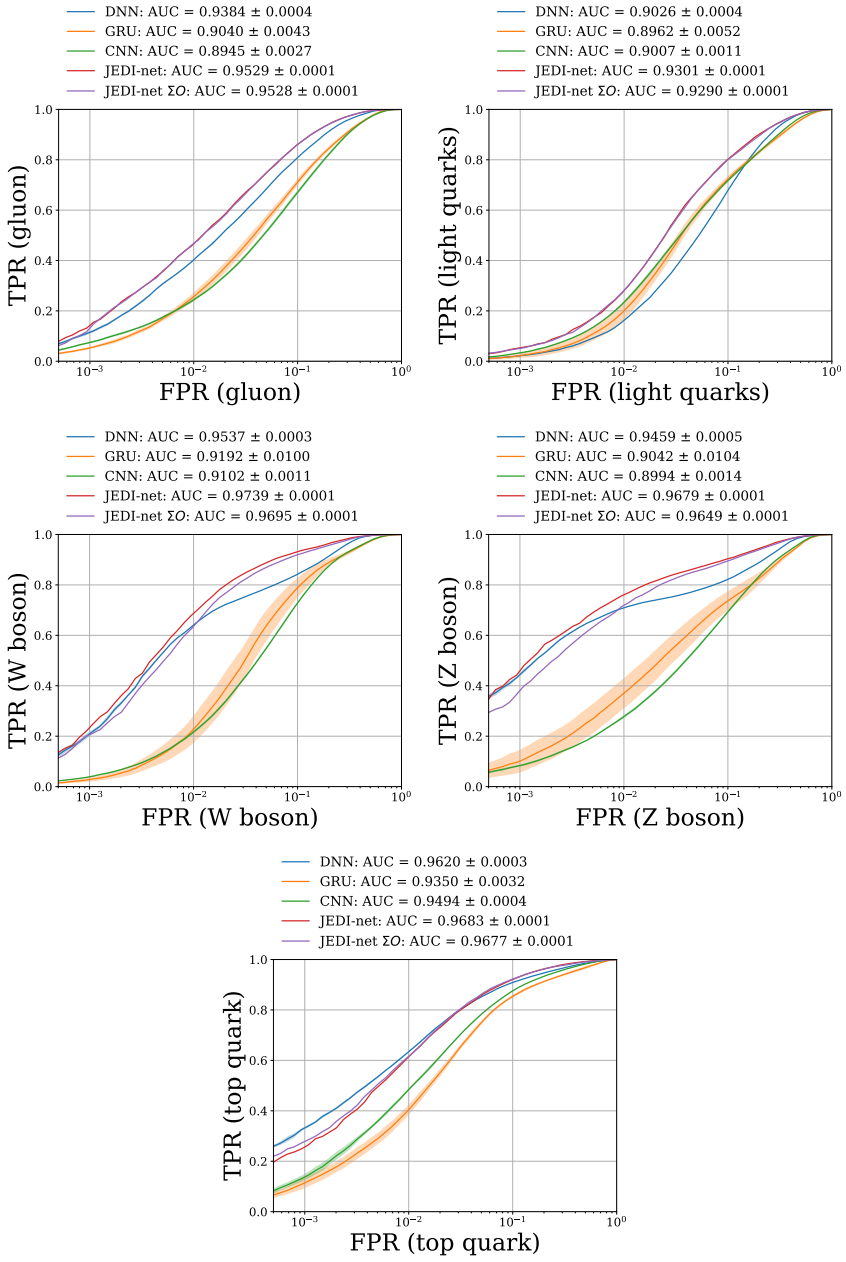


Figure 6.8: ROC curves for JEDI-net and the three alternative models, computed for gluons (top-left), light quarks (top-right), W (center-left) and Z (center-right) bosons, and top quarks (bottom). The solid lines represent the average ROC curves derived from 10 k-fold trainings of each model. The shaded bands around the average lines are represent one standard deviation, computed with the same 10 k-fold trainings.

Table 6.7: True positive rates (TPR) for the optimized JEDI-net taggers and the three alternative models (DNN, CNN, and GRU), corresponding to a false positive rate (FPR) of 10% (top) and 1% (bottom). The largest TPR value for each case is highlighted in bold.

Jet category	DNN	GRU	CNN	JEDI-net	JEDI-net with $\sum O$
TPR for FPR=10%					
gluon	0.830 ± 0.002	0.740 ± 0.014	0.700 ± 0.008	0.878 ± 0.001	**0.879 ± 0.001**
light quarks	0.715 ± 0.002	0.746 ± 0.011	0.740 ± 0.003	**0.822 ± 0.001**	0.818 ± 0.001
W boson	0.855 ± 0.001	0.812 ± 0.035	0.760 ± 0.005	**0.938 ± 0.001**	0.927 ± 0.001
Z boson	0.833 ± 0.002	0.753 ± 0.036	0.721 ± 0.006	**0.910 ± 0.001**	0.903 ± 0.001
top quark	0.917 ± 0.001	0.867 ± 0.006	0.889 ± 0.001	0.930 ± 0.001	**0.931 ± 0.001**
TPR for FPR=1%					
gluon	0.420 ± 0.002	0.273 ± 0.018	0.257 ± 0.005	**0.485 ± 0.001**	0.482 ± 0.001
light quarks	0.178 ± 0.002	0.220 ± 0.037	0.254 ± 0.007	**0.302 ± 0.001**	0.301 ± 0.001
W boson	0.656 ± 0.002	0.249 ± 0.057	0.232 ± 0.006	**0.704 ± 0.001**	0.658 ± 0.001
Z boson	0.715 ± 0.001	0.386 ± 0.060	0.291 ± 0.005	**0.769 ± 0.001**	0.729 ± 0.001
top quark	**0.651 ± 0.003**	0.426 ± 0.020	0.504 ± 0.005	0.633 ± 0.001	0.632 ± 0.001

working point of the top-jet tagger, for which the DNN model gives a TPR higher by about 2%, while the CNN and GRU models give much worse performance.

The TPR values for the two JEDI-net models are within 1%. The only exception is observed for the tight working points of the W and Z taggers, for which the model using the $\overline{O}$ sums shows a drop in TPR of $\sim 4\%$. In this respect, the model using summed $\overline{O}$ features is preferable (despite this small TPR loss), given the reduced model complexity (see Section 6.4) and its independence on the labeling convention for the particles embedded in the graph and for the edges connecting them.

What did JEDI-net learn?

In order to characterize the information learned by JEDI-net, we consider the $\overline{O}$ sums across the N_O vertices of the graph and we study their correlations to physics motivated quantities, typically used when exploiting jet substructure in a search. We consider the HLF quantities used for the DNN model and the N-subjettiness variables $\tau_N^{(\beta)}$ [214], computed with angular exponent $\beta = 1, 2$.

Not all the $\overline{O}$ sums exhibit an obvious correlation with the considered quantities, i.e., the network engineers high-level features that encode other information than what is used, for instance, in the DNN model.

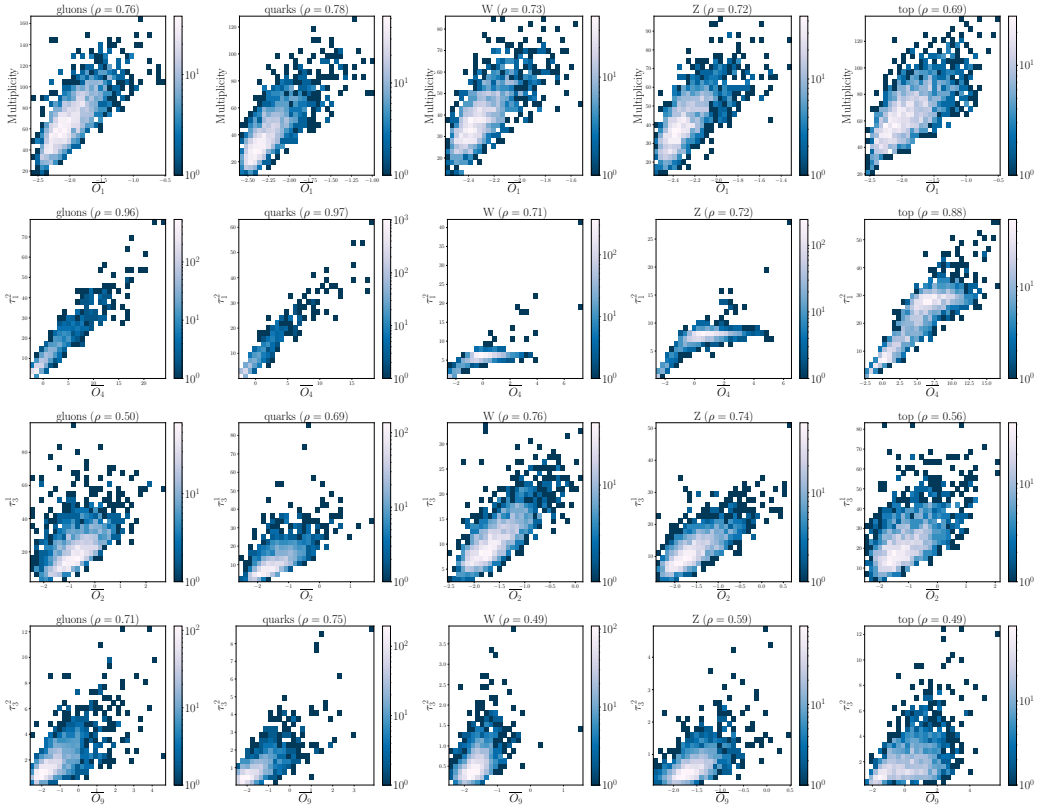


Figure 6.9: Two-dimensional distributions between (top to bottom) $\overline{O}_1$ and constituents multiplicity, $\overline{O}_4$ and $\tau_1^{(\beta=2)}$, $\overline{O}_2$ and $\tau_3^{(\beta=1)}$, $\overline{O}_9$ and $\tau_3^{(\beta=2)}$, for jets originating from (right to left) gluons, light flavor quarks, W bosons, Z bosons, and top quarks. For each distribution, the linear correlation coefficient ρ is reported.

Nevertheless, some interesting correlation pattern between the physics motivated quantities and the $\overline{O}_i$ sums is observed. The most relevant examples are given in Fig. 6.9, where the 2D histograms and the corresponding linear correlation coefficient (ρ) are shown. The correlation between $\overline{O}_1$ and the particle multiplicity in the jet is not completely unexpected. As long as the O quantities aggregated across the graph have the same order of magnitude, the corresponding sum $\overline{O}$ would be proportional to jet-constituent multiplicity.

The strong correlation between the $\overline{O}_4$ and $\tau_1^{(\beta=2)}$ (with ρ values between 0.69 and 0.97, depending on the jet class) is much less expected. The τ_1^{β} quantities assume small values when the jet constituents can be arranged into a single *sub-jet* inside the jet. Aggregating information from the constituent momenta across the jet, the JEDI-net model based on the $\overline{O}$ quantities learns to build a quantity very close to $\tau_1^{(\beta=2)}$. The last two rows of Fig. 6.9 show two intermediate cases: the correlation between $\overline{O}_2$ and $\tau_3^{(\beta=1)}$ and between $\overline{O}_9$ and $\tau_3^{(\beta=2)}$. The two $\overline{O}$ sums considered

are correlated to the corresponding substructure quantities, but with smaller (within 0.48 and 0.77) correlation coefficients.

Resource comparison

Table 6.8 shows a comparison of the computational resources needed by the different models discussed in this section. The best-performing JEDI-net model has more than twice the number of trainable parameters than the DNN and GRU model, but approximately a factor of 6 less parameters than the CNN model. The JEDI-net model based on the summed $\overline{O}$ features achieves comparable performance with about a factor of 4 less parameters, less than the DNN and GRU models. While being far from expensive in terms of number of parameters, the JEDI-net models are expensive in terms of the number of floating point operations (FLOP). The simple model based on $\overline{O}$ sums, using as input a sequence of 150 particles, uses 458 MFLOP. The increase is mainly due to the scaling with the number of vertices in the graph. Many of these operations are the $\times 0$ and $\times 1$ products involving the elements of the R_R and R_S matrices. The cost of these operations could be reduced with an IN implementation optimized for inference, e.g., through an efficient sparse-matrix representation.

Table 6.8: Resource comparison across models. The quoted number of parameters refers only to the trainable parameters for each model. The inference time is measured by applying the model to batches of 1000 events 100 times: the 50% median quantile is quoted as central value and the 10%-90% semi-distance is quoted as the uncertainty. The GPU used is an NVIDIA GTX 1080 with 8 GB memory, mounted on a commercial desktop with an Intel Xeon CPU, operating at a frequency of 2.60GHz. The tests were executed in PYTHON 3.7 with no other concurrent process running on the machine.

Model	Number of parameters	Number of FLOP	Inference time/batch [ms]
DNN	14725	27 k	1.0 ± 0.2
CNN	205525	400 k	57.1 ± 0.5
GRU	15575	46 k	23.2 ± 0.6
JEDI-net	33625	116 M	121.2 ± 0.4
JEDI-net with $\sum O$	8767	458 M	402 ± 1

In addition, we quote in Table 6.8 the average inference time on a GPU. The inference time is measured applying the model to 1000 events, as part of a PYTHON application based on TENSORFLOW [121]. To this end, the JEDI-net models, implemented and trained in PYTORCH, are exported to ONNX [215] and then loaded as TENSORFLOW

graph. The quoted time includes loading the data, which occurs for the first inference and is different for different event representations, that is smaller for the JEDI-net models than for the CNN models. The GPU used is an NVIDIA GTX 1080 with 8 GB memory, mounted on a commercial desktop with an Intel Xeon CPU, operating at a frequency of 2.60 GHz. The tests were executed in PYTHON 3.7, with no other concurrent process running on the machine. Given the larger number of operations, the GPU inference time for the two IN models is much larger than for the other models.

The current IN algorithm is costly to deploy in the online selection environment of a typical LHC experiment. A dedicated R&D effort is needed to reduce the resource consumption in a realistic environment in order to benefit from the improved accuracy that INs can achieve. For example, one could trade model accuracy for reduced resource needs by applying neural network pruning [216, 217], reducing the numerical precision [218, 219], and limiting the maximum number of particles in each jet representation.

6.5 Modified JEDI-net for the identification of boosted $\mathrm{H} \to \mathrm{b\bar{b}}$ decays

Based on JEDI-net, we develop an algorithm to identify high-transverse-momentum Higgs bosons decaying to bottom quark-antiquark pairs and distinguish them from ordinary jets that reflect the configurations of quarks and gluons at short distances. The algorithm's inputs are features of the reconstructed charged particles in a jet and the secondary vertices associated with them. Describing the jet shower as a combination of particle-to-particle and particle-to-vertex interactions, the model is trained to learn a jet representation on which the classification problem is optimized. The algorithm is trained on simulated samples of realistic LHC collisions, released by the CMS Collaboration on the CERN Open Data Portal. The interaction network achieves a drastic improvement in the identification performance with respect to state-of-the-art algorithms.

The IN is based on two input collections comprising N_p particles, each represented by a feature vector of length P, and N_v vertices, each represented by a feature vector of length S. Although kinematic features of neutral particles could also be taken into account with an additional input graph, we verified that doing so does not significantly improve the performance for this task as shown in Sec. 6.8. Further, excluding neutral particles has the benefit of improved robustness to pileup. For a single jet, the input consists of an X and a Y matrix, with sizes $P \times N_p$ and $S \times N_v$,

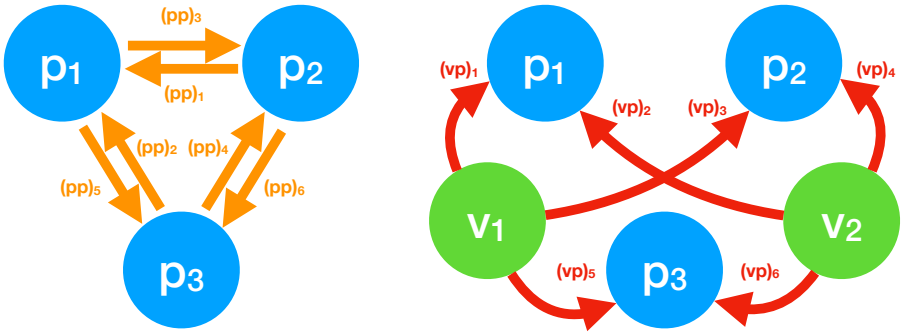


Figure 6.10: Two example graphs with 3 particles and 2 vertices and the corresponding edges.

respectively. The X matrix contains the input features (columns) of the charged particles (rows), while the Y matrix contains the input features of the SVs.

A particle graph $\mathcal{G}_p$ is constructed by connecting each particle to every other particle through $N_{pp} = N_p(N_p - 1)$ directed edges. Similarly, a particle-vertex graph $\mathcal{G}_{pv}$ is constructed by connecting each vertex to each particle through $N_{pv} = N_p N_v$ directed edges. As described below, we only consider those edges that are received by particles because the final aggregation is performed over the particles. These graphs are pictorially represented in Fig. 6.10 for the case of three particles and two vertices. As shown in the figure, the graph nodes and edges are arbitrarily enumerated. The result of the graph processing is independent of the labeling order, as described below.

For the graph $\mathcal{G}_p$, a receiving matrix (R_R) and a sending matrix (R_S) are defined, both of size $N_p \times N_{pp}$. The element $(R_R)_{ij}$ is set to 1 when the ith particle receives the jth edge and is 0 otherwise. Similarly, the element $(R_S)_{ij}$ is set to 1 when the ith particle sends the jth edge and is 0 otherwise. For the second graph, the corresponding adjacency matrices R_K (of size $N_p \times N_{vp}$) and R_V (of size $N_v \times N_{vp}$) are defined. In the example of Fig. 6.6, the R_R, R_S, R_K, and R_V matrices would be written as:

$$R_R = \begin{array}{c} \\ p_1 \\ p_2 \\ p_3 \end{array} \begin{array}{c} \begin{array}{cccccc} (pp)_1 & (pp)_2 & (pp)_3 & (pp)_4 & (pp)_5 & (pp)_6 \end{array} \\ \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} \end{array}, \tag{6.5}$$

$$R_S = \begin{array}{c} \\ p_1 \\ p_2 \\ p_3 \end{array} \begin{array}{c} \begin{array}{cccccc} (pp)_1 & (pp)_2 & (pp)_3 & (pp)_4 & (pp)_5 & (pp)_6 \end{array} \\ \begin{pmatrix} 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 \end{pmatrix} \end{array}, \tag{6.6}$$

$$R_K = \begin{array}{c} \\ p_1 \\ p_2 \\ p_3 \end{array} \begin{array}{c} \begin{array}{cccccc} (vp)_1 & (vp)_2 & (vp)_3 & (vp)_4 & (vp)_5 & (vp)_6 \end{array} \\ \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} \end{array}, \tag{6.7}$$

$$R_V = \begin{array}{c} \\ v_1 \\ v_2 \end{array} \begin{array}{c} \begin{array}{cccccc} (vp)_1 & (vp)_2 & (vp)_3 & (vp)_4 & (vp)_5 & (vp)_6 \end{array} \\ \begin{pmatrix} 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix} \end{array}. \tag{6.8}$$

Each column of an adjacency matrix corresponds to a directional connection from one particle to another, $(pp)_i$, or from a vertex and to a particle, $(vp)_j$. Column entries that are 1 in a given row in the receiving matrix R_R indicate that the corresponding particle receives that connection. Likewise, if a column entry is 1 in a given row in the sending matrix R_S, the corresponding particle is the sender for that connection. Because the fully connected particle graph we consider has no self-connections, i.e. no particle sends and receives the same connection, the rows of R_R and R_S do not share any of the same nonzero column entries. For the R_R and R_V adjacency matrices, we only consider those connections that are sent to particles because the final aggregation is performed over the particles. We tested a version of the IN architecture in which we considered connections that are sent to vertices as well and aggregated separately before being processed by the final network, but found no significant improvement.

The data flow of the IN model is pictorially represented in Fig. 6.11. The input processing starts by creating the $2P \times N_{pp}$ particle-particle interaction matrix B_{pp} and the $(P + S) \times N_{vp}$ particle-vertex interaction matrix B_{vp} defined as:

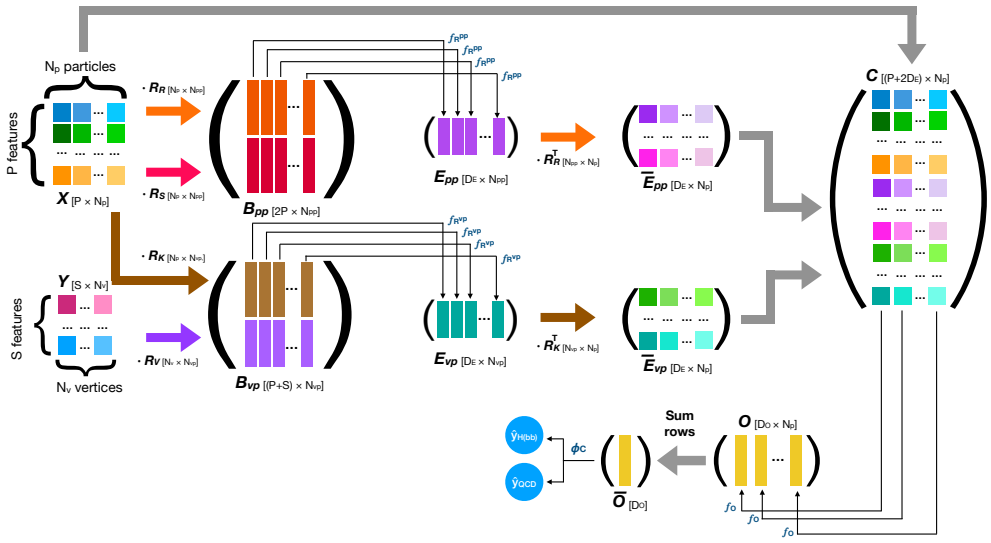


Figure 6.11: Illustration of the modified JEDI-net for the boosted H $\longrightarrow$ b$\bar{\text{b}}$ classifier. The particle feature matrix X is multiplied by the receiving and sending matrices R_R and R_S to build the particle-particle interaction feature matrix B_{pp}. Similarly, the particle feature matrix X and the vertex feature matrix Y are multiplied by the adjacency matrices R_K and R_V, respectively, to build the particle-vertex interaction feature matrix B_{vp}. These pairs are then processed by the interaction functions f_R^{pp} and f_R^{vp}, and the post-interaction function f_O, which are expressed as neural networks and learned in the training process. This procedure creates a learned representation of each particle's post-interaction features, given by N_p vectors of size D_O. The N_p vectors are summed, giving D_O features for the entire jet, which is given as input to a classifier ϕ_C, also represented by a neural network. More details on the various steps are given in the text.

$$B_{pp} = \begin{pmatrix} X \cdot R_R \\ X \cdot R_S \end{pmatrix}, \tag{6.9}$$

$$B_{vp} = \begin{pmatrix} X \cdot R_K \\ Y \cdot R_V \end{pmatrix}, \tag{6.10}$$

where $\cdot$ indicates the ordinary matrix product. Each column of B_{pp} consists of the $2P$ features of the sending and receiving nodes of each particle-particle interaction, while each column of B_{vp} consists of the $P + S$ features of each particle-vertex one.

Processing each column of B_{pp} by the function f_R^{pp}, one builds an internal representation of the particle-particle interaction with a function $f_R^{pp} : \mathbb{R}^{2P} \mapsto \mathbb{R}^{D_E}$, where D_E is the size of the internal representation. This results in an *effect matrix* E_{pp}

with dimensions $D_E \times N_{pp}$. We similarly build the E_{vp} matrix, with dimensions $D_E \times N_{vp}$, using a function $f_R^{vp} : \mathbb{R}^{P+S} \mapsto \mathbb{R}^{D_E}$.

We then propagate the particle-particle interactions back to the particles receiving them, by building $\overline{E}_{pp} = E_{pp} R_R^\top$ with dimension $D_E \times N_p$. We also build $\overline{E}_{vp} = E_{vp} R_V^\top$ with dimension $D_E \times N_p$, which collects the information of the particle-vertex interactions for each particle and across all of the vertices.

The next step consists of building the C matrix, with dimensions $(P+2D_E) \times N_p$, by combining the input information for each particle (X) with the learned representation of the particle-particle ($\overline{E}_{pp}$) and particle-vertex ($\overline{E}_{vp}$) interactions:

$$C = \begin{pmatrix} X \\ \overline{E}_{pp} \\ \overline{E}_{vp} \end{pmatrix}. \tag{6.11}$$

The final aggregator combines the input and interaction information to build the postinteraction representation of the graph, summarized by the matrix O, with dimensions $D_O \times N_p$. The aggregator consists of a function $f_O : \mathbb{R}^{P+2D_E} \mapsto \mathbb{R}^{D_O}$, which computes the elements of the O matrix The elements of the O matrix are computed by a function $f_O : \mathbb{R}^{P+2D_E} \mapsto \mathbb{R}^{D_O}$, which returns the postinteraction representation for each of the input nodes. As is done for f_R^{pp} and f_R^{vp}, f_O is applied to each column of C.

We stress the fact that the by-column processing applied by the f_R^{pp}, f_R^{vp}, and f_O functions and the sum across interactions by defining the $\overline{E}_{pp}$ and $\overline{E}_{vp}$ matrices are essential ingredients to make the outcome of the IN tagger independent of the order used to label the N_p input particles and N_v input vertices. In other words, while the representations of the R_R, R_S, R_K, and R_V matrices depend on the adopted labeling convention, the final representation of each particle does not.

The learned representation of the post-interaction graph, given by the elements of the O matrix, can be used to solve the specific task at hand. Depending on the task, the final function that computes the classifier output may be chosen to preserve the permutation invariance of the input particles and vertices. In this case, we first sum along each row (corresponding to a sum over particles) of O to produce a feature vector $\overline{O}$ with length D_O for the jet as a whole. This is passed to a function $\phi_C : \mathbb{R}^{D_O} \mapsto \mathbb{R}^N$, which produces the output of the classifier.

The training of the IN is performed with the CMS open simulation with 2016 conditions. The input dataset is split into training, validation, and test samples with percentages of 80%, 10%, and 10%, respectively.

We use PyTorch [206] to implement and train the classifier on one NVIDIA GeForce GTX 1080 GPU. We also convert the interaction network into a TensorFlow model.The model is implemented with each of f_R^{pp} and f_R^{vp} expressed as a sequence of 3 dense layers of sizes $(60, 30, 20)$ with a rectified linear unit (ReLU) activation function after each layer. The function f_O is a similar sequence of dense layers of sizes $(60, 30, 24)$ with ReLU activations. We use up to $N_p = 60$ charged particles and $N_v = 5$ secondary vertices as inputs to the IN tagger. Given the size of these layers, the total number of trainable parameters is 18,144. We train the model using the Adam optimizer [212] with an initial learning rate of 10^{-4} and a batch size of 128 for up to 200 epochs, enforcing early stopping [220] on the validation loss with a patience of 5 epochs. The size of the batch is constrained by the required memory utilization of the GPU. The training takes approximately 25 minutes per epoch on the GPU and stopped after 110 epochs.

For the baseline algorithm, we minimize the categorical cross-entropy loss function for this classification task L_{C} and let the network exploit all of the discriminating information in the dataset.

To determine the impact of neutral particles, we also train an augmented all-particle IN model, which consumes an additional input set with 10 kinematic features for up to 100 charged or neutral particles, listed in Table 6.4. This additional input set is processed by the model in a similar way to the SV input set: the set of all particles is fully connected to the set of charged particles. The effect matrix for these interactions is computed by an independent neural network and then appended to an enlarged C matrix, now of size $(P + 3D_E) \times N_p$, before being processed by the network f_O. The remaining steps of the model proceed as described above. The total number of trainable parameters for this model is 24,254.

6.6 Decorrelation with the jet mass

Many possible applications of a jet tagging algorithm would require the final score to be uncorrelated from the jet mass, so that a selection based on the tagger score does not change the jet mass distribution. This is particularly relevant for the background distribution, but is required to some extent also for the signal one. Several techniques exist to deliver a tagger with minimal effects on the jet mass

distribution. For taggers based on high-level features, one could remove those features more correlated to the jet mass or divide those correlated features by the jet mass. For taggers based on a more *raw* representation of the jet (as in this case), one could perform an adversarial training [221–225]. One could also reweight or remove background events such that the background m_{SD} distribution is indistinguishable from the signal m_{SD} distribution [226]. Finally, one could also define a mass-dependent threshold based on simulation as in the "designing decorrelated taggers" (DDT) procedure proposed in Ref. [227]. We found the DDT method to be the most robust and performant deocorrelation procedure. As such, we use it as the nominal decorrelation method in the following results.

Designing decorrelated taggers

Following the DDT procedure [227], the tagger threshold for a given false positive rate (FPR) or "working point" is determined as a function of m_{SD}. By creating a m_{SD}-dependent tagger threshold, the background jet m_{SD} distribution for events passing and failing this threshold can be made identical. In practice, this is done by considering the distribution of the network score versus the jet m_{SD} for the training dataset. A quantile regression was used to find the threshold on the network score as a function of m_{SD} distribution that would correspond to a fixed quantile (the chosen 1−FPR value). By construction, this procedure results in near-perfect mass decorrelation.

In this case, a gradient boosted regressor [228, 229] with the following parameters was used:

- α-quantile of 1 − FPR,
- number of estimators of 500,
- minimum number of samples at a leaf node of 50,
- minimum number of samples to split an internal node of 2500,
- maximum depth of 5,
- validation set of 20%,
- early stopping with tolerance of 10.

6.7 Deep double-b tagger models

The DDB tagger is a convolutional and recurrent neural network model developed by CMS [204] to identify boosted $H \to b\bar{b}$ jets. We reconstruct this model based on publicly available information from the CMS Collaboration as follows. The model takes as input 27 HLFs used in Ref. [59], as well as 8 particle-specific features of up to 60 charged particles, and 2 properties of up to 5 SVs associated with the jet. Each block of inputs is treated as a one-dimensional list, with batch normalization [230] applied directly to the input layers. For each collection of charged particles and SVs, separate 1D convolutional layers [231], with a kernel size of 1, are applied: 2 hidden layers with 32 filters each and ReLU [209] activation. The outputs are then separately fed into two gated recurrent units (GRUs) with 50 output nodes each and ReLU activations. Finally, the GRU outputs are concatenated with the HLFs and processed by a dense layer with 100 nodes and ReLU activation, and another final dense layer with 2 output nodes with softmax activation. Dropout [232] (with a rate of 10%) is used in each layer to prevent overfitting. The nominal DDB tagger model has 40,344 trainable parameters, 32% of which are found in the fully connected layers.

We define a variant of this model, the DDB+ model, which takes as input all 30 features of charged particles and all 14 features of the SVs. In this variant, we do not consider the HLFs. Thus, the final dense layer only receives the GRU outputs from processing the low-level charged particle and SV information. This extended DDB+ tagger algorithm has 38,746 trainable parameters. The number of parameters is less overall because the increase in the size of the convolutional and recurrent layers is compensated by the decrease in the size of the fully connected layers.

We train the DDB and DDB+ models using the CMS open simulation dataset with KERAS [120] over up to 200 epochs with an early stopping patience of 5 epochs and a batch size of 4096 using the Adam optimizer with an initial learning rate of 10^{-3}. For both models, one training epoch takes about 3 minutes and training stops after approximately 50 epochs. In this case, the larger batch size is possible due to the smaller GPU memory utilization of the model during training. We find consistent performance for different batch size choices with no evidence of overfitting with larger batch sizes.

In order to decorrelate the tagger output from the jet mass, we use the same DDT procedure described in Sec. 6.6 applied to both the DDB and DDB+ taggers.

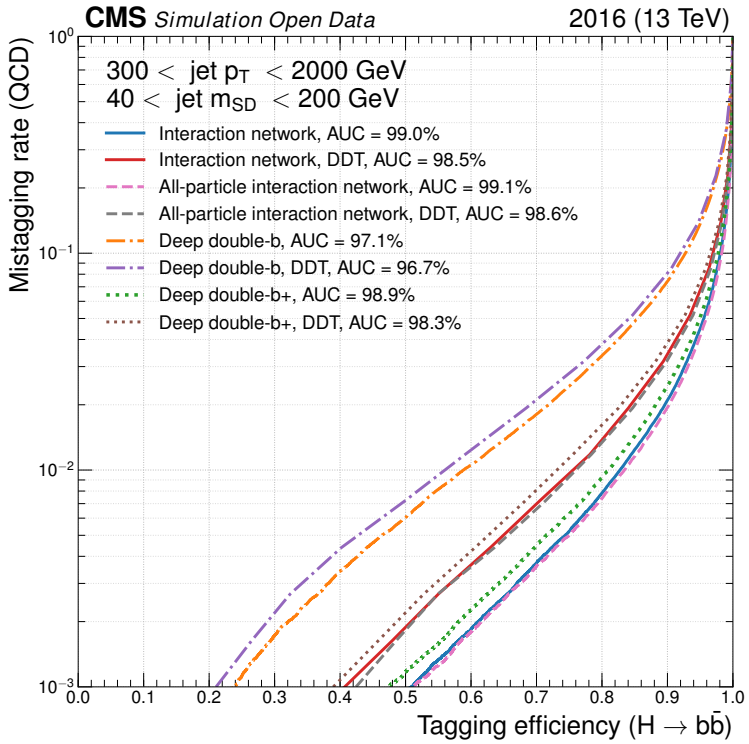


Figure 6.12: Performance of the IN, all-particle IN, DDB, and DDB+ algorithms quantified with a ROC curve of FPR (QCD mistagging rate) versus TPR (H $\rightarrow$ b$\bar{\text{b}}$ tagging efficiency). The performance of each baseline algorithm is compared to that of the algorithms after applying the DDT procedure to decorrelate the tagger score from the jet mass. This decorrelation results in a smaller TPR for a given FPR.

6.8 Results

In Fig. 6.12 the performance of the IN, all-particle IN, DDB, and DDB+ algorithms are quantified in a ROC curve. The axes are the TPR, or H $\rightarrow$ b$\bar{\text{b}}$ tagging efficiency and the false positive rate, or QCD mistagging rate. As shown in Fig. 6.12, the IN provides an improved performance with respect to the DDB and DDB+ taggers. At a 1% FPR, the IN tagger outperforms the DDB and DDB+ taggers by 37% and 2% in TPR, respectively. Likewise, at a 50% TPR, the IN tagger yields a factor of 6 or 1.2 better background rejection (1/FPR) than the DDB or DDB+ tagger, respectively. Thus, while the additional inputs provide a significant improvement for the DDB+ model, the IN architecture is also important to achieve a better performance with significantly less parameters than the DDB+ model.

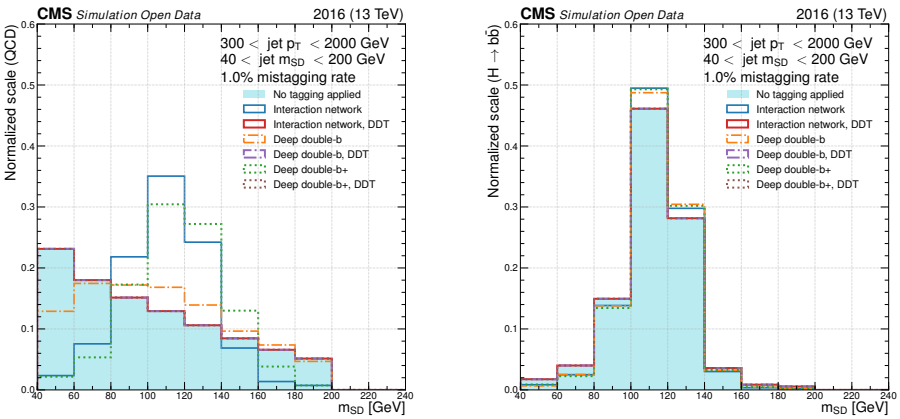


Figure 6.13: An illustration of the "sculpting" of the background jet mass distribution (left) and the signal jet mass distribution (right) after applying a threshold on the tagger score corresponding to a 1% FPR for several different algorithms. The unmodified interaction network is highly correlated with the jet mass, but after applying the methods described in the text, the correlation is reduced for the background while the peak of the signal distribution is still retained.

We verified that one could match the performance obtained by the IN with a DDB-inspired architecture and expanding the model size. With 150,786 trainable parameters, a DDB architecture achieves the same performance as the IN at the cost of 8 times more parameters.

Because of this the IN model holds an advantage in terms of memory usage during inference over this alternative model.

Figure 6.12 also shows that there is only a modest improvement in the AUC and accuracy by including information in the IN model from neutral particles. For this reason and to preserve robustness to increased pileup, in the following results, we consider the original IN model that excludes neutral particles.

Figure 6.13 shows an illustration of how the signal and background jet mass distributions change after applying a threshold on the different baseline and DDT-decorrelated tagger scores. Following Ref. [225], we quantify the impact of these algorithms on the mass decorrelation by computing the Jensen-Shannon (JS) divergence:

$$D_{\mathrm{JS}}(P \parallel Q) = \frac{1}{2} D_{\mathrm{KL}}(P \parallel M) + \frac{1}{2} D_{\mathrm{KL}}(Q \parallel M), \tag{6.12}$$

where $M = \frac{1}{2}(P + Q)$ is the average of the normalized m_{SD} distributions of the background jets passing (P) and failing (Q) a given tagger score and $D_{\mathrm{KL}}(P \parallel Q) =$

Table 6.9: Performance metrics of the different baseline and decorrelated models, including accuracy, area under the ROC curve, background rejection at a true positive rate of 30% an 50%, and true positive rate and mass decorrelation metric $1/D_{\mathrm{JS}}$ at a false positive rate of 1%. For the DDT models, the corresponding accuracy is listed for the tagger after the decorrelation is performed for a FPR of 50%.

Baseline models	Interaction network	Deep double-b	Deep double-b+
Parameters	**18,144**	40,344	38,746
Accuracy	**95.5%**	91.7%	95.3%
AUC	**99.0%**	97.2%	98.8%
$1/\epsilon_b$ @ $\epsilon_s = 30\%$	**4616.9**	578.0	3863.1
$1/\epsilon_b$ @ $\epsilon_s = 50\%$	**1028.8**	165.3	852.7
$1/\epsilon_s$ @ $\epsilon_b = 1\%$	**82.8%**	60.6%	81.5%
$1/D_{\mathrm{JS}}$ @ $\epsilon_b = 1\%$	4.5	**75.3**	4.4
Decorrelated models	Interaction network, DDT	Deep double-b, DDT	Deep double-b+, DDT
Parameters	**18,144**	40,344	38,746
Accuracy	**93.2%**	86.8%	92.9%
AUC	**98.5%**	96.7%	98.3%
$1/\epsilon_b$ @ $\epsilon_s = 30\%$	**2258.7**	456.6	1973.8
$1/\epsilon_b$ @ $\epsilon_s = 50\%$	**540.0**	136.8	466.6
$1/\epsilon_s$ @ $\epsilon_b = 1\%$	**75.6%**	55.9%	72.9%
$1/D_{\mathrm{JS}}$ @ $\epsilon_b = 1\%$	29,265.3	**48,099.0**	15,171.2

$\sum_i P_i \log(P_i/Q_i)$ is the Kullback-Leibler (KL) divergence. Larger values of the metric $1/D_{\mathrm{JS}}$ correspond to a better decorrelation.

After applying the mass decorrelation techniques, the performance of each of the taggers worsens slightly but the IN algorithm still significantly outperforms the DDB and DDB+ taggers. Figure 6.14 displays the trade-off between the background rejection and $1/D_{\mathrm{JS}}$ at different TPRs for the baseline and DDT-decorrelated algorithms. At a 50% TPR, the decorrelated IN algorithm achieves a significantly better $1/D_{\mathrm{JS}}$ by a factor of about 2,200 while the background rejection decreases by a factor of about 3.3 compared to the baseline IN algorithm. At a 1% FPR, the DDT-decorrelated IN tagger has a TPR of 75.6% compared to the DDT-decorrelated DDB (DDB+) tagger with a 55.9% (72.9%) TPR, corresponding to an improvement of 35% (4%). Table 6.9 summarizes different performance metrics for the three considered models and their decorrelated versions. For the DDT models, the corresponding accuracy is listed for the tagger after the decorrelation is performed for a FPR of 50%.

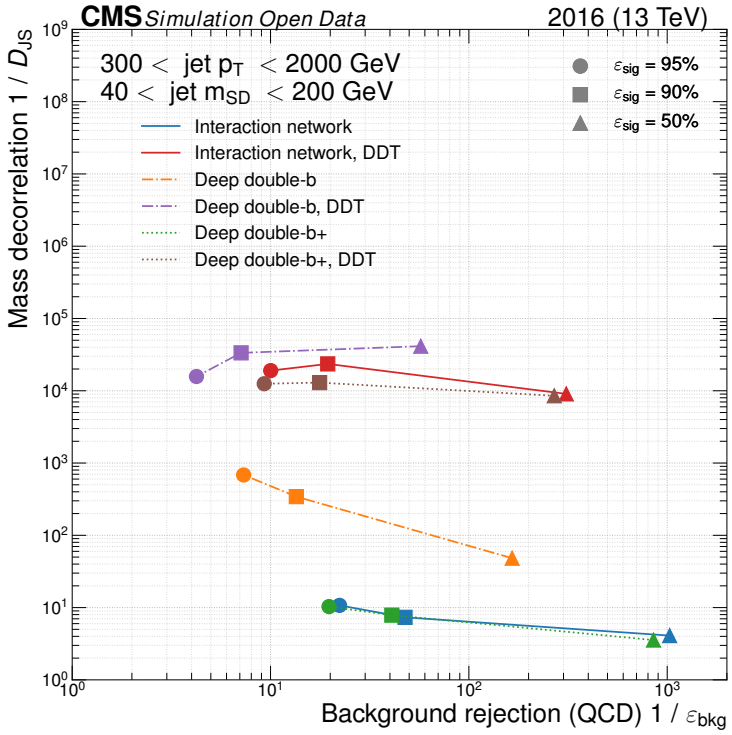


Figure 6.14: The mass decorrelation metric $1/D_{\mathrm{JS}}$ as a function of background rejection for the baseline and decorrelated IN, DDB, and DDB+ taggers. The decorrelation is quantified as the inverse of the JS divergence between the background mass distribution passing and failing a given threshold cut on the classifier score. Greater values of this metric correspond to better mass decorrelation. The background rejection is quantified as the inverse of the FPR, while the signal efficiency is equal to the TPR.

To quantify the dependence on the number of pileup interactions, Fig. 6.15 shows the performance of the different algorithms as a function of the number of primary vertices in the event, which scales linearly with the number of pileup collisions. Using only charged particles and secondary vertices as input, the IN tagger is robust against an increasing number of pileup interactions, exhibiting behavior similar to the DDB and DDB+ taggers.

6.9 Summary

We presented a novel technique using a graph representation of the jet's constituents and secondary vertices based on an interaction network to identify Higgs bosons decaying to bottom quark-antiquark pairs ($\mathrm{H} \to \mathrm{b\bar{b}}$) in LHC collisions. This model

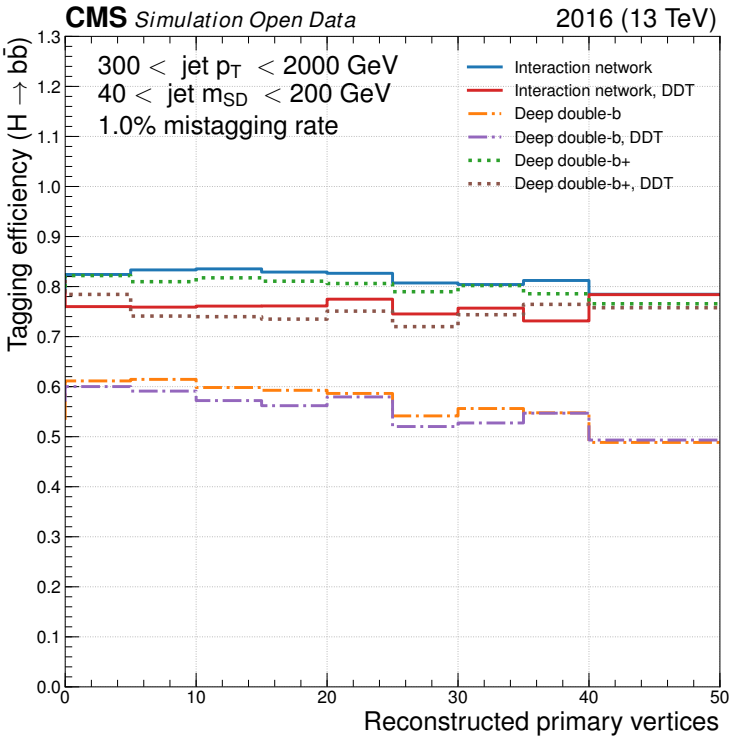


Figure 6.15: TPR of the baseline and decorrelated IN, DDB, and DDB+ taggers as a function of the number of reconstructed PVs for a 1% FPR.

can operate on a variable number of jet constituents and secondary vertices and does not depend on the ordering schemes of these objects. The interaction network was trained on an open simulation dataset released by the CMS Collaboration in the CERN Open Data Portal. A significant improvement in performance is observed with respect to two alternative taggers based on the deep double-b tagger created by the CMS Collaboration. By design, the interaction network uses extended low-level input features for particles and vertices, offers a more flexible representation of jet data, and is robust against the noise generated by pileup collisions. Even when trained with the same set of input features, the interaction network architecture outperforms the deep double-b architecture. Thus, while part of the improvement is due to the extended input representation, additional improvement comes from the interaction network architecture, despite using on half as many parameters.

Together with the best-performing models, we presented additional models, obtained by applying different decorrelation techniques between the network score and the jet-mass distribution. This was done to minimize the selection bias of the classifier

output towards any values of the jet mass, which would make the algorithms suitable for physics analyses relying on the jet mass as a discriminating variable. As expected, the decorrelation procedure results in a reduction of the $H \rightarrow b\bar{b}$ identification performance. Nevertheless, the decorrelated interaction network model outperforms the decorrelated deep double-b models.

Once applied to a full data analysis, this graph-based tagging algorithm could contribute a substantial improvement to the experimental precision of $H \rightarrow b\bar{b}$ measurements, including those sensitive to beyond the standard model physics and the Higgs boson self-coupling. These results motivate further exploration of applications based on interaction networks (and graph neural networks in general) for object tagging and other similar tasks in experimental high energy physics.

Part IV

New physics searches at the LHC

Chapter 7

SEARCH FOR LONG-LIVED PARTICLES WITH DELAYED PHOTON SIGNATURE ON RUN 2 DATA IN CMS

After decades of searching for new physics in the prompt regime, where particles beyond the SM decay promptly near the beam spot, we almost exhausted the search phase space available within the reach of the collider's energy. However, at the weak scale, particles beyond the SM can generally have long lifetimes with decay lengths of up to a few meters. Such long-lived particles (LLPs) open up a promising avenue of new physics to explore at the LHC.

In this chapter, we present a search for LLPs under the gauge-mediated SUSY breaking (GMSB) [22–28, 30, 233] scenario, commonly referred to as the "Snowmass Points and Slopes 8" (SPS8) benchmark model [234]. In this scenario, the LLPs are the neutralino $\widetilde{\chi}^0$, which is the next-to-lightest SUSY particle (NLSP), decaying into the lightest SUSY particle (LSP) – the gravitino $\widetilde{\mathrm{G}}$ – as shown in Fig. 7.1. In the SPS8 model, the SUSY breaking scale Λ is a free parameter that determines the decay rate of SUSY particles, the primary production mode, and is linearly proportional to the mass of the neutralino $\widetilde{\chi}^0$. The NLSP dominantly decays into a photon and a gravitino, resulting in a final state with one or two photons and a missing transverse momentum ($p_{\mathrm{T}}^{\mathrm{miss}}$) induced by the gravitinos escaping the detector.

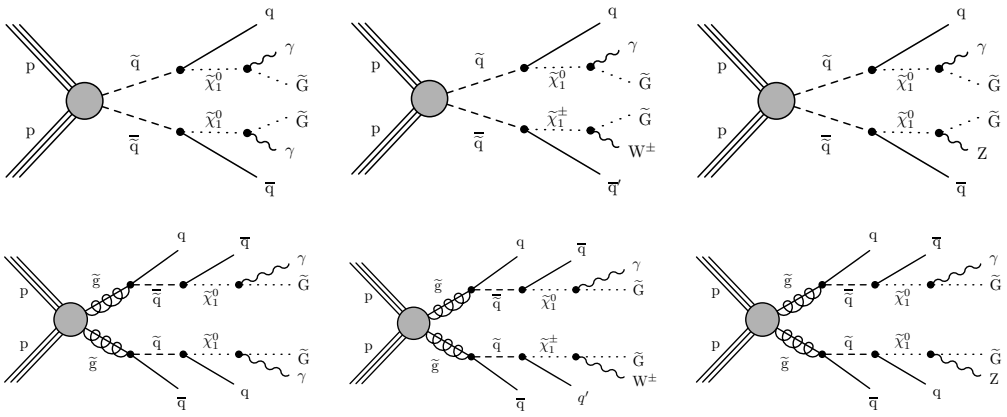


Figure 7.1: Example Feynman diagrams for SUSY processes with diphoton (left) and single photon (middle and right) final states via pair production of squark (upper) and gluino (lower) from pp collisions at the LHC.

Previously, CMS performed a similar search on 2016 and 2017 data with 77.4 fb^{-1} at $\sqrt{s}$ = 13 TeV [235], which excludes such GMSB models for a neutralino mass under 220 GeV at $c\tau$ of 1 m. ATLAS reported a similar search on Run 1 data with 20.3 fb^{-1} at $\sqrt{s}$ = 8 TeV [236], which excludes such GMSB models for a neutralino mass under 100 GeV at $c\tau$ of 30 cm. These results are summarized in Figure 7.2.

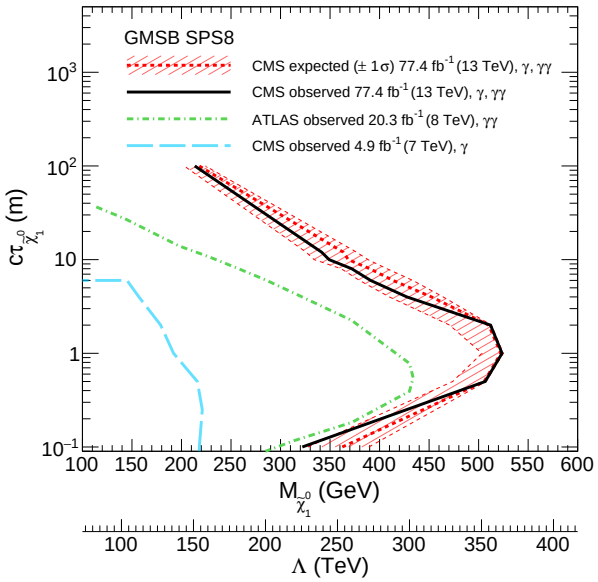


Figure 7.2: The 95% CL exclusion contours for the GMSB SPS8 neutralino production cross section set by the previous CMS search on 2016 and 2017 data, along with the ATLAS and CMS results in Run 1 [235].

The analysis introduces a two-fold improvement over the previous CMS search. First, the analysis is performed on the whole Run 2 data with 136.33 fb^{-1} at $\sqrt{s}$ = 13 TeV. Second, we develop a new delayed photon identification with deep neural networks to improve upon the cut-based identification method in the previous CMS analysis. The photon arrival time in the ECAL is then reconstructed and calibrated as one of the two shape variables, along with the missing transverse energy $p_{\mathrm{T}}^{\mathrm{miss}}$ to extract the signal yield.

This chapter is organized as follows: Section 7.1 discusses the data used in this analysis and the simulated MC samples used for training the delayed photon identification and for the photon time reconstruction studies. Section 7.2 lays out details on reconstructing physics objects, such as photons, missing transverse energy $p_{\mathrm{T}}^{\mathrm{miss}}$, and jets. The development, validation, and deployment of the neural-network-based delayed photon identification are described in Section 7.3. Section 7.4 is dedicated to explaining the photon arrival time reconstruction process in the ECAL, as well

as the calibration of the photon time in MC. The event selection and categorization are given in Section 7.5. Section 7.6 describes the data-driven method to estimate the background and the process to extract the upper limit for the signal strength. Section 7.7 lists the systematic uncertainties in this analysis. Finally, the results are presented in Section 7.8.

7.1 Event samples

The search uses the whole Run 2 data of proton-proton collisions with a total integrated luminosity of 136.33 fb^{-1}, spanning across 3 years from 2016 to 2018 at $\sqrt{s}$ = 13 TeV. Simulated samples includes the GMSB signals, the SM background processes, and the $Z \to e^+e^-$ sample for the timing studies.

As data recorded in CMS are grouped into different primary data sets based on their physics contents, this analysis uses primary data sets that contain at least 1 photon in the event final state. In particular, the primary datasets used in this analysis are `DoubleEG`, `SinglePhoton`, and `EGamma` from the data collected in 2016, 2017, and 2018, respectively. The sizes of these primary data sets are listed in Table 7.1.

Table 7.1: List of primary data sets used in this analysis and their respective sizes. `RAW` data are recorded directly from the detector to disk, while `MiniAOD` are reduced data format after physics object reconstruction to be used for most analyses.

Year	Primary data set	`RAW` data size	`MiniAOD` data size
2016	`DoubleEG`	251 TB	12 TB
2017	`SinglePhoton`	77 TB	3 TB
2018	`EGamma`	972 TB	47 TB

The production of GMSB SPS8 signal models is generated with PYTHIA 8 [103]. The model specifications, input parameters, coupling spectra, and decay tables are tabulated in a set of SUSY Les Houches According (SLHA) files produced via ISASUGRA toolkit as part of ISAJET generator package [237]. The list of all generated GMSB signal samples and the corresponding masses of the gluino, neutralino, and gravitino, as well and their production cross section, is shown in Table 7.2.

The quantum chromodynamics (QCD) background events are generated with PYTHIA v8.3 [103] at leading-order (LO) precision, enriched with photons. The MADGRAPH5_aMC@NLO v2.2.2 generator [99–101] is used at next-to-leading-order in QCD to simulate events originating from γ+jets, W+jets, and Z+jets production. Diphoton events, including Born processes with up to 3 additional jets and box-diagram processes at leading-order precision, are generated with SHERPA v2.2.4

Table 7.2: List of all generated GMSB SPS8 signal models, parametrized by the SUSY breaking scale Λ, and their corresponding production cross section and masses of the gluinos $\tilde{g}$, neutralinos $\widetilde{\chi}^0$, and gravitino $\tilde{G}$. For each Λ value, 10 different neutralino's lifetime values are generated, corresponding to 10 gravitino's masses $M_{\tilde{G}}$.

Λ	$M_{\tilde{g}}$	$M_{\tilde{\chi}_1^0}$	$M_{\tilde{G}}$ (eV) for different $\widetilde{\chi}^0$ lifetime values (m)										cross section	
(TeV)	(GeV)	(GeV)	0.1	0.5	1	2	4	6	8	10	12	100	(fb)	
100	838	139	0.2	0.3	0.5	0.7	1.0	1.2	1.4	1.5	1.7	4.9	2175 ±	14
150	1207	212	0.5	1.0	1.5	2.1	3.0	3.6	4.2	4.7	5.1	15	228.1 ±	1.5
200	1565	285	1.0	2.2	3.2	4.5	6.4	7.8	9.0	10	11	32	43.7 ±	0.3
250	1915	358	1.8	4.0	5.7	8.1	11	14	16	18	20	57	12.6 ±	0.1
300	2260	430	2.9	6.5	9.2	13	18	23	26	29	32	92	4.45 ±	0.03
350	2599	503	4.3	9.6	14	19	27	33	39	43	47	136	1.78 ±	0.01
400	2935	576	6.1	14	19	27	38	47	54	61	67	192	0.778 ±	0.005
450	3267	650	8.2	18	26	37	52	64	73	82	90	259	0.344 ±	0.001
500	3595	723	11	24	34	48	68	83	96	107	118	340	0.165 ±	0.000

[102]. The fragmentation and parton showering are modeled with the CMS PYTHIA8 (CP5) [107, 108] underlying event tune, and the parton distribution function sets are generated from NNPDF3 [109] NLO for 2016 and NNPDF3.1 [110] for 2017 and 2018.

7.2 Physics object reconstruction

All physics objects are reconstructed with the particle-flow (PF) algorithm [113]. This physics analysis uses jet, photon, and $p_{\mathrm{T}}^{\mathrm{miss}}$ objects.

PF candidates are clustered with the anti-k_T algorithm with a cone size of 0.4, and then undergo a set of identification criteria listed in Table 7.3 to form a jet. This search further requires the jets to have transverse momentum $p_{\mathrm{T}} > 30$ GeV and $|\eta| < 3.0$, as well as to be outside the cone radii of 0.3 of the two leading photon objects.

Table 7.3: Jet identification criteria recommended by CMS for different years.

	2016	2017	2018
Neutral hadronic energy fraction	< 0.99	< 0.90	< 0.90
Neutral electromagnetic energy fraction	< 0.99	< 0.90	< 0.090
Number of constituents	> 1	> 1	> 1
Charged hadron fraction	> 0	> 0	> 0
Charged multiplicity	> 0	> 0	> 0
Charged electromagnetic energy fraction	< 0.99	-	-

Photons are identified from clusters of reconstructed hits in ECAL crystals. Photons falling in the gap between the ECAL barrel (EB) and endcap (EE) regions ($1.4442 < |\eta| < 1.566$) are excluded from this analysis because the performance of the photon reconstruction algorithm there is not optimal. Two types of photon collections are recorded in MINIAOD events: general-event-description (GED) photons and out-of-time (OOT) photons. The GED photon collection only records photons with arrival time in the ECAL up to 3 ns with respect to a prompt photon from the primary vertex. The OOT photon collection saves photons that are not recorded in the GED collection. Photons are required to pass a set of online selection criteria, defined by the triggers used in this analysis, which are summarized in Table 7.4. The final photons are chosen from the neural-network-based delayed photon identifier, which will be described in Section 7.3.

Table 7.4: Trigger selection criteria for photons. In 2016, the trigger requires the presence of two photons. In 2017 and 2018, only one photon with p_{T} greater than 60 GeV is required.

	2016	2017 and 2018
R_9	≥ 0.85	≥ 0.9
H/E	≤ 0.1	≤ 0.15
$\sigma_{i\eta i\eta}$	≤ 0.024	≤ 0.014
ECAL cluster isolation	$\leq 8.0 + 0.012\ p_{\mathrm{T}}$	$\leq 5.0 + 0.01\ p_{\mathrm{T}}$
HCAL cluster isolation	$5.0 + 0.005\ p_{\mathrm{T}}$	$12.5 + 0.03\ p_{\mathrm{T}} + 3.0 \times 10^{-5}\ {p_{\mathrm{T}}}^2$
Tracker isolation	$\leq 8.0 + 0.002\ p_{\mathrm{T}}$	$\leq 6.0 + 0.002\ p_{\mathrm{T}}$

Both OOT and GED photon collections use the same reconstruction algorithm starting from a seed crystal. If the seed time is less than 3 ns, the reconstructed photon goes to the GED collection, and if the seed time is greater than 3 ns, the photon goes to the OOT collection. In some rare cases, there can be a partial overlap between reconstructed GED and OOT photon clusters such that the two clusters share some common crystals. Consequently, the reconstructed photons from the OOT and GED collections can overlap with each other. To avoid double counting, whenever a GED photon is within the radius $\Delta R < 0.3$ of an OOT photon, the photon with with smaller transverse momentum p_{T} is removed from the combined list of photon objects.

The missing transverse momentum $p_{\mathrm{T}}^{\mathrm{miss}}$ is the negative vectorial sum of all reconstructed PF candidates' transverse momenta in an event. Miscalibration of the detector, noise, or beam-induced backgrounds can occasionally produce "fake"

high-$p_{\mathrm{T}}^{\mathrm{miss}}$ events, therefore we apply a series of filters to remove these anomalous events. The $p_{\mathrm{T}}^{\mathrm{miss}}$ algorithm originally does not include OOT photons when calculating the vectorial sum of all PF candidates' transverse momenta since the full particle flow reconstruction algorithm does not run on OOT photons. Therefore, the transverse momenta of OOT photons are added back to the list of PF candidates, and GED photons are removed from that list if it is removed from the combined photon list from the overlapping photon removal procedure described earlier.

7.3 Delayed photon identification

One major improvement of this search over the previous CMS search on 2016 and 2017 data is a novel delayed photon identifier based on deep neural networks (DNN). Separated identifiers are developed for leading photon, subleading photon in the ECAL barrel, and subleading photon in ECAL endcap. Identifiers are also developed independently for the year 2016 versus 2017 and 2018 because of the different triggers. Seven variables are used as the input to the DNN, four of which are based on the shower shape of the photon in the ECAL cluster, and the remaining three are the isolation variables from the detector components.

The four shower shape variables are:

- R_9: The ratio between the energy sum over the 3×3 crystal matrix centered on the most energetic crystal and the total energy of the supercluster. Isolated, unconverted photons deposit most of their energy within this 3×3 matrix grid, while converted photons have a much wider range of energy deposit.

- $\sigma_{i\eta i\eta}$: The width of the energy-weighted shower distribution within the 5×5 crystal matrix centered on the most energetic crystal along the η coordination, defined in Eq. 7.1:

$$\sigma_{i\eta i\eta} = \left(\frac{\sum (\eta_i - \bar{\eta})^2 \omega_i}{\sum \omega_i} \right)^{\frac{1}{2}}, \text{where} \tag{7.1}$$

$$\bar{\eta} = \frac{\sum \omega_i \eta_i}{\sum \omega_i}, \text{and} \tag{7.2}$$

$$\omega_i = \max \left(0, 4.7 + \log \frac{E_i}{E_{5\times5}} \right). \tag{7.3}$$

  The threshold value 4.7 in Eq. 7.3 is to exclude crystals with pure noise in the 5×5 crystal matrix from the shower shape computation, *i.e.*, $\log \frac{E_i}{E_{5\times5}} > -4.7$, or $E_i > 0.009 E_{5\times5}$.

- S_{major}: The semi-major axis of the elliptical photon shower.
- S_{minor}: The semi-minor axis of the elliptical photon shower.

  Formally, S_{major} and S_{minor} are the diagonal elements of the shower shape covariance matrix, defined in Eq. 7.4:

$$\text{Covariance}_{\eta\phi} = \begin{pmatrix} S_{\eta\eta} & S_{\eta\phi} \\ S_{\phi\eta} & S_{\phi\phi} \end{pmatrix}, \tag{7.4}$$

  where $S_{\eta\eta}$ and $S_{\phi\phi}$ are the energy-weighted variances of the shower shape along the η and ϕ directions, respectively, while $S_{\eta\phi}$ and $S_{\phi\eta}$ are the corresponding covariances along the 2 directions:

$$S_{\eta\eta} = \frac{\sum \omega_i (\eta_i - \bar{\eta})^2}{\sum \omega_i}, \tag{7.5}$$

$$S_{\phi\phi} = \frac{\sum \omega_i (\phi_i - \bar{\phi})^2}{\sum \omega_i}, \tag{7.6}$$

$$S_{\eta\phi} \equiv S_{\phi\eta} = \frac{\sum \omega_i (\eta_i - \bar{\eta})(\phi_i - \bar{\phi})}{\sum \omega_i}, \tag{7.7}$$

  where $\bar{\eta}$ and ω_i are defined in Eq. 7.2 and 7.3, respectively, and $\bar{\phi}$ is the energy-weighted mean in ϕ direction:

$$\bar{\phi} = \frac{\sum \omega_i \phi_i}{\sum \omega_i}. \tag{7.8}$$

  Once the covariance matrix in Eq. 7.4 is diagonized, S_{major} and S_{minor} can be computed directly as in Eq. 7.9:

$$S_{\frac{\text{major}}{\text{minor}}} = \frac{S_{\phi\phi} + S_{\eta\eta} \pm \sqrt{(S_{\phi\phi} - S_{\eta\eta})^2 + 4S_{\eta\phi}^2}}{2}. \tag{7.9}$$

The three isolation variables are:

- `EcalPFClusterIso/p`$_T$: The scalar sum of p_T of all PF candidates in a $\Delta R = 0.3$ cone around the photon direction in the ECAL divided by the photon's p_T.
- `HcalPFClusterIso/p`$_T$: The scalar sum of p_T of all PF candidates in a $\Delta R = 0.3$ cone around the photon direction in the HCAL divided by the photon's p_T. In 2016, this quantity was not recorded in the OOT photon collection in MiniAOD, therefore, it was replaced by `NeutralHadPFIso/p`$_T$, which is the scalar sum of p_T of all neutral hadron PF candidates in a $\Delta R = 0.3$ cone around the photon direction divided by the photon's p_T.

- `TrkSumPtHollowConeDR03/p`$_{\mathrm{T}}$: the scalar sum of p_{T} of all tracks in a $\Delta R = 0.3$ cone around the photon direction divided by the photon's p_{T}.

Samples of all available points in the $(\Lambda, c\tau)$ space are mixed together as signal for training. However, photons with distances from the production vertices to the primary vertex greater than 20 cm in the longitudinal and transverse directions are removed from the mixture so that the signal mixture is enriched in displaced photons, as shown in Fig. 7.3. Additionally, photons coming from GMSB sample points with $\Lambda < 200$ TeV and $c\tau < 200$ cm are also removed because these points are already excluded in the previous CMS search [235].

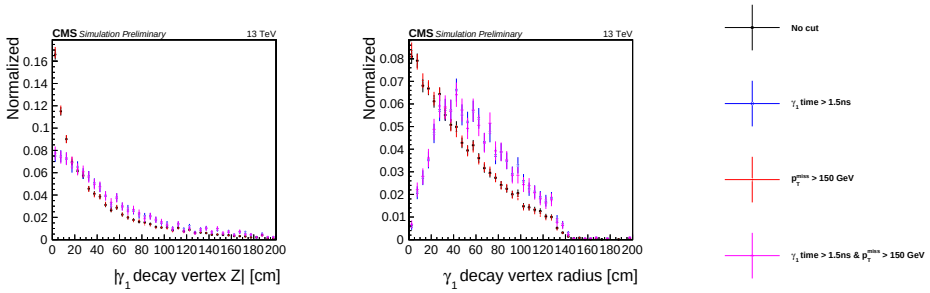


Figure 7.3: The displacements of GMSB signal photons in the longitudinal (left) and transverse (right) directions from a representative point $\Lambda = 400$ TeV and $c\tau = 200$ cm. In a typical signal region (γ_1 time > 1.5 ns and $p_{\mathrm{T}}^{\mathrm{miss}} > 150$ GeV), the GMSB signal photons are enriched in the phase space where the displacements are greater than 20 cm in both longitudinal and transverse directions.

The distributions of these DNN input variables for the leading photon for the year 2016 and 2017 are shown in Fig. 7.4 and 7.5, respectively. The same variables for subleading photons in EB and EE for 2017 are, respectively, shown in Fig. 7.6 and 7.7.

Background samples are mixed together, weighted by the cross section of each process. Before training, signal and background classes are re-weighted so that the total weights are equal between the 2 classes.

The samples are randomly split between training/validation/test with a 80/10/10 ratio. The training input variables are standardized to have means of 0 and standard deviations of 1 before getting fed into the neural networks. The DNN are implemented in PyTorch with 3 hidden layers of sizes 300, 200, and 20, respectively. Dropout is used after each hidden layer with a rate between 0 and 0.2. The output of each hidden layer is activated with a ReLU function. The model is trained for a

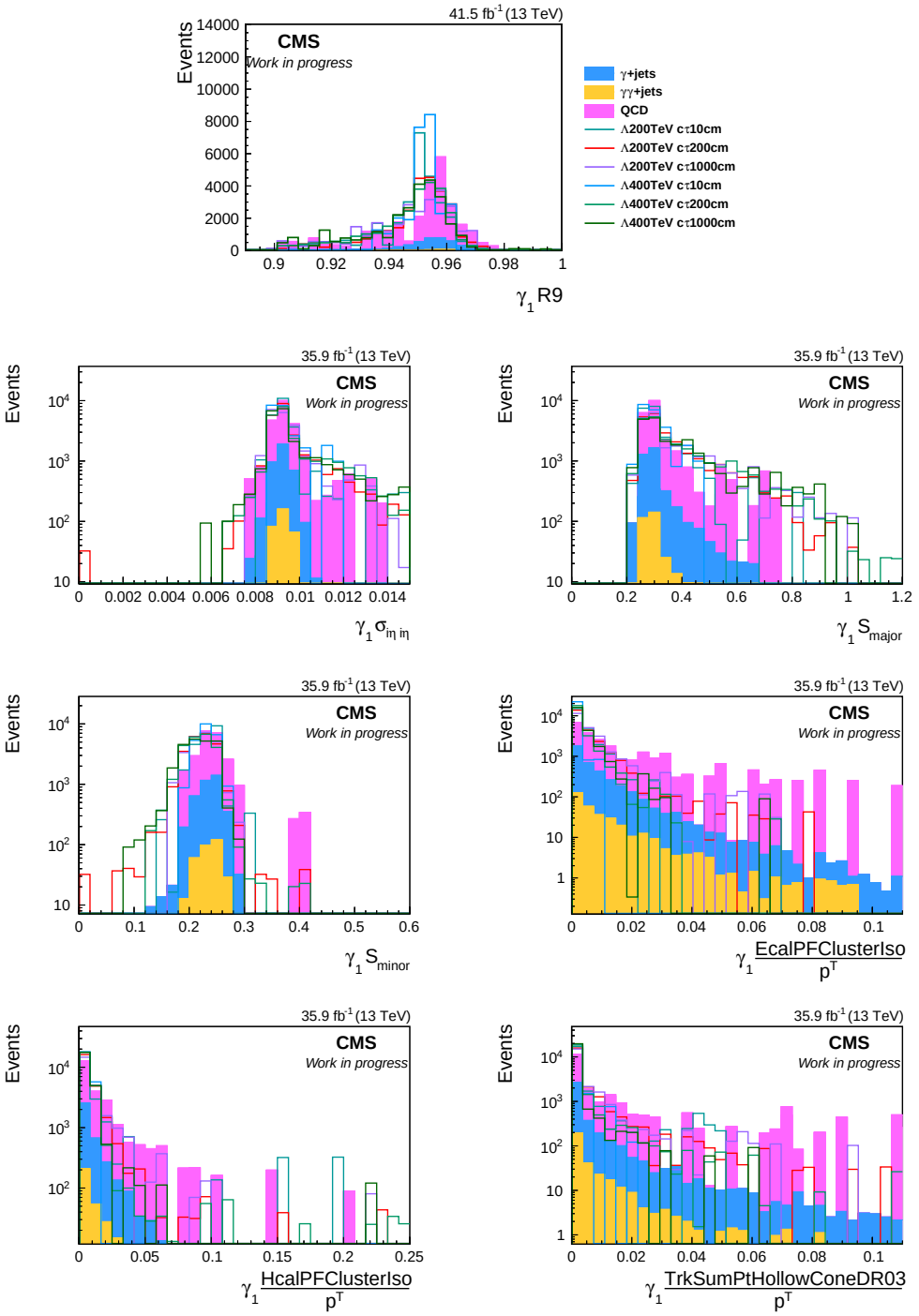


Figure 7.4: The signal and background MC distributions of the 7 input variables to the deep neural networks for the leading photon for 2016. GMSB signal histograms are scaled up to have the same bin integrals as the sum of background histograms.

maximum of 500 epochs with `Adam` optimizer, minimizing the binary cross-entropy losses between the predictions and target labels. The initial learning rate is set to

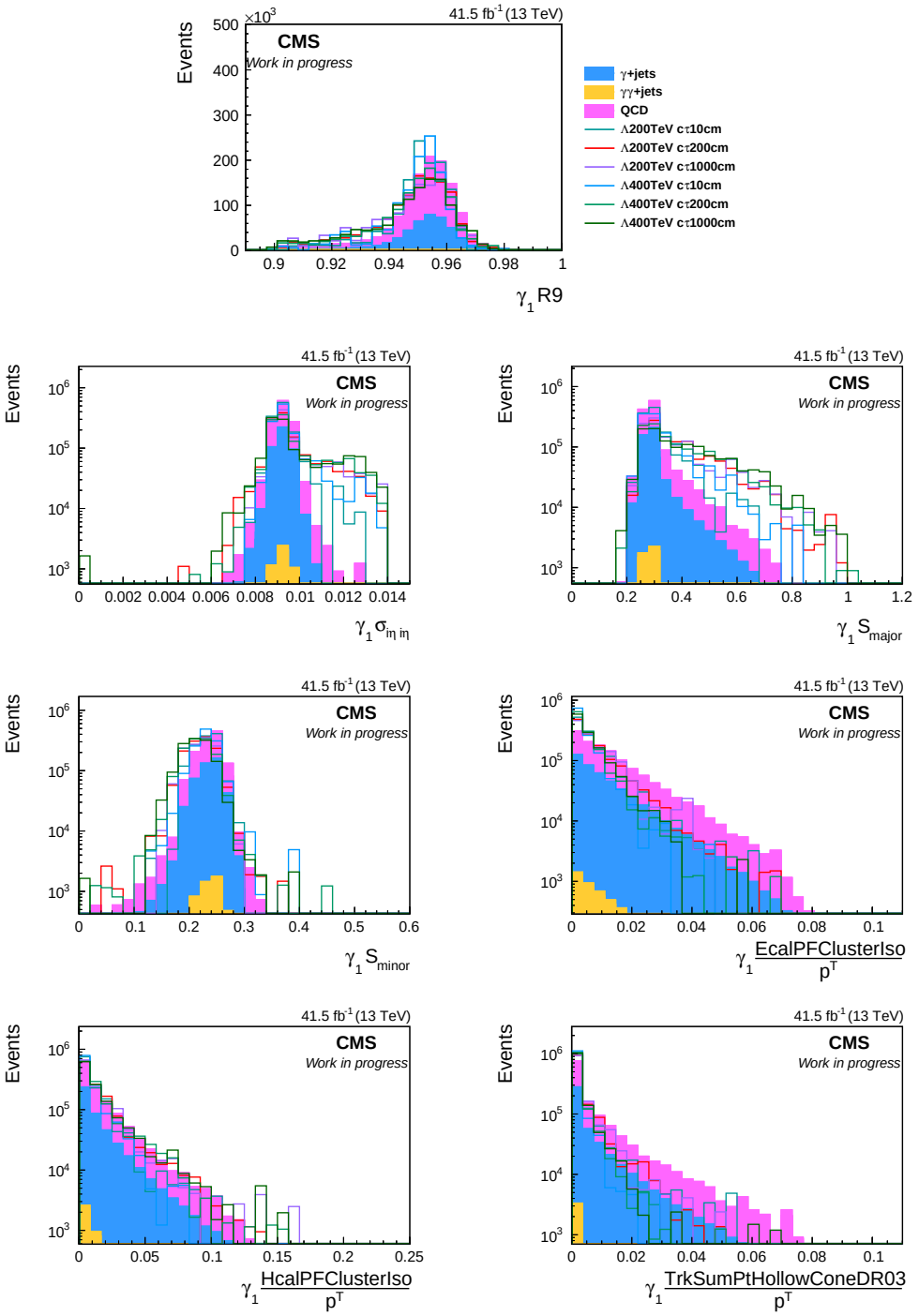


Figure 7.5: The signal and background MC distributions of the 7 input variables to the deep neural networks for the leading photon for 2017. GMSB signal histograms are scaled up to have the same bin integrals as the sum of background histograms.

1.0×10^{-3}, reduced by a factor of 0.3 whenever the validation loss does not decrease

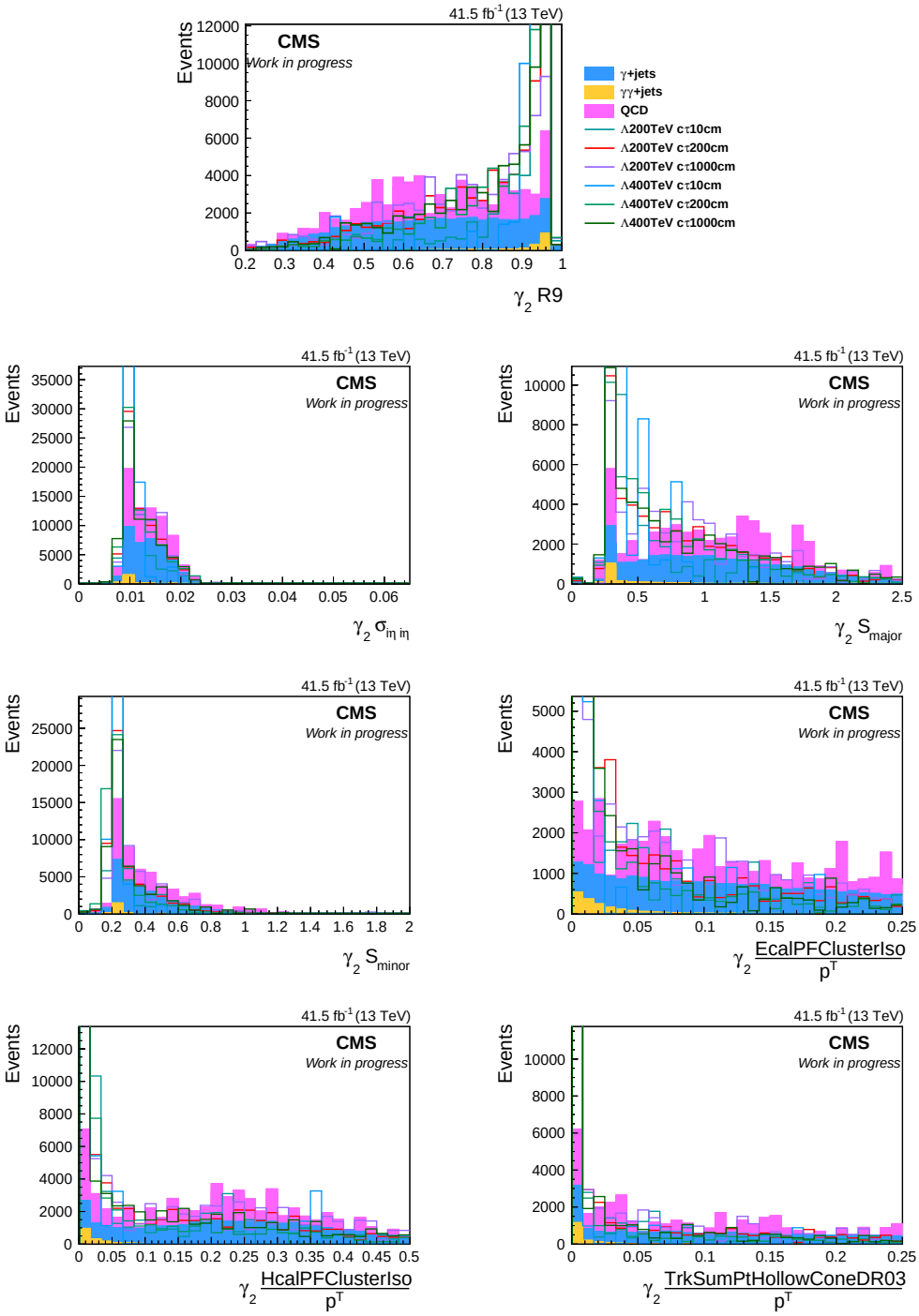


Figure 7.6: The signal and background MC distributions of the 7 input variables to the deep neural networks for the second photon in the ECAL barrel region for 2017. GMSB signal histograms are scaled up to have the same bin integrals as the sum of background histograms.

after the last 8 epochs. The training is early terminated when the validation loss stops decreasing after the last 30 epochs.

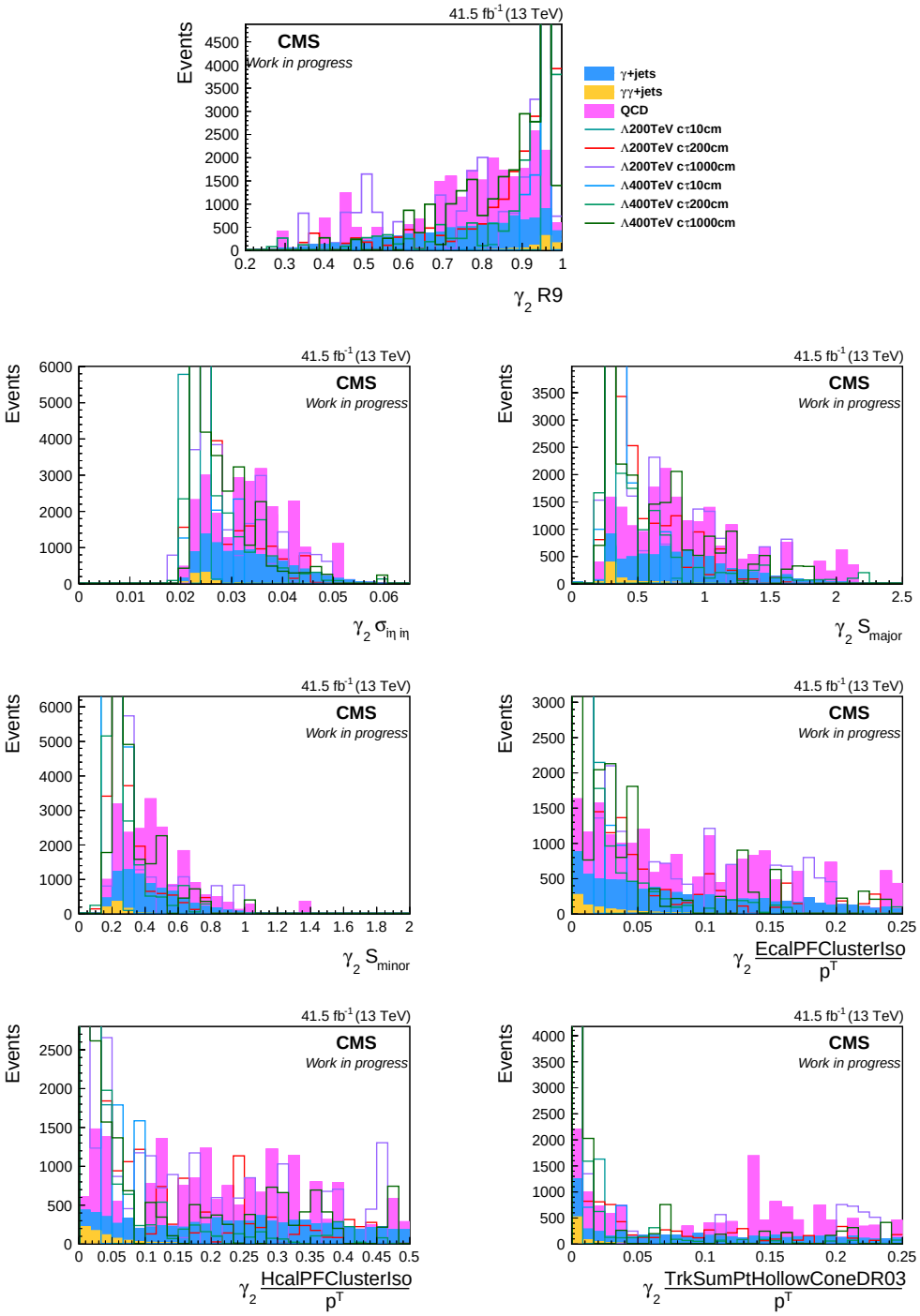


Figure 7.7: The signal and background MC distributions of the 7 input variables to the deep neural networks for the second photon in the ECAL endcap region for 2017. GMSB signal histograms are scaled up to have the same bin integrals as the sum of background histograms.

The performances of the DNN for the leading photon are shown in Fig. 7.8, with the corresponding DNN score distributions shown in Fig. 7.9. For subleading photons,

the receiver operating characteristic (ROC) curves for 2017+2018 identifiers are shown in Fig. 7.10, with the corresponding DNN score distributions shown in Fig. 7.11. There is no identifier for subleading photons for 2016, as in the previous CMS search, since the trigger used in 2016 already puts stringent requirements on the second photon.

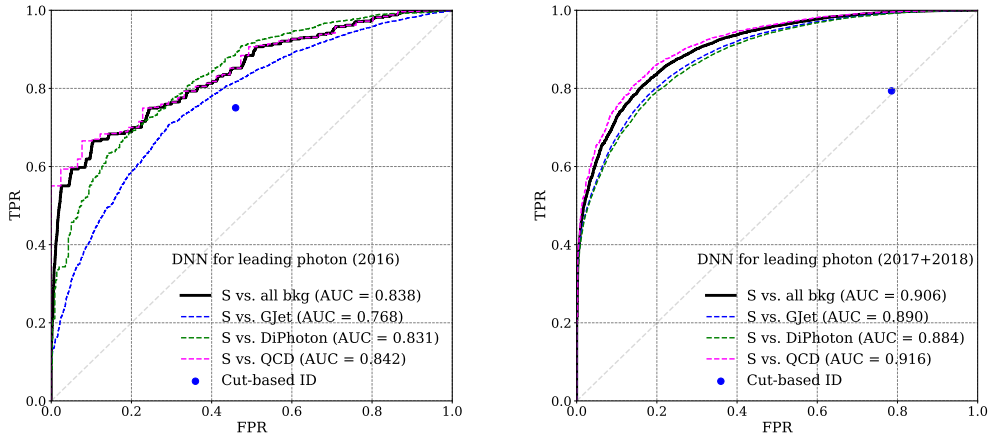


Figure 7.8: The receiver operating characteristic (ROC) curve of the leading photon's DNN identifiers for 2016 (left) and 2017+2018 (right). Signal consists of all GMSB signal samples with some removals based on sample points and displacements as described in Sec. 7.3. The performances of the cut-based identifiers used in the previous CMS search are denoted with the blue dots.

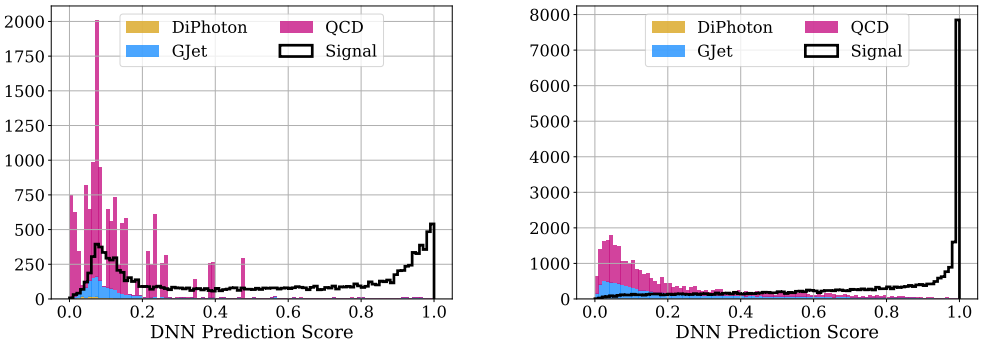


Figure 7.9: The distributions of the leading photon's DNN scores on MC signal and background samples for 2016 (left) and 2017+2018 (right). Signal consists of all GMSB signal samples with some removals based on sample points and displacements as described in Sec. 7.3.

To understand how the DNN photon identifier makes decisions, we use the SHAP framework [238], which uses Shapley values, a concept in cooperative game theory

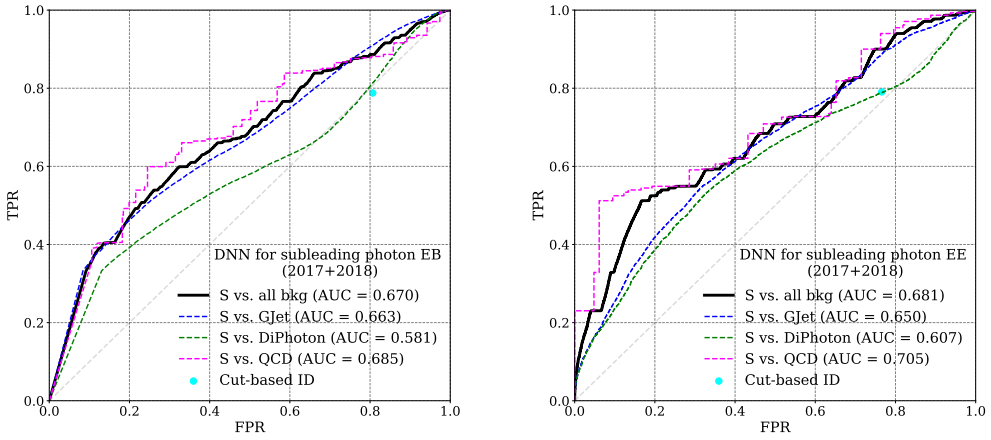


Figure 7.10: The receiver operating characteristic (ROC) curve of the subleading photon's DNN identifiers for 2017 and 2018 in the ECAL barrel (left) and endcap (right). Signal consists of all GMSB signal samples with some removals based on sample points and displacements as described in Sec. 7.3. The performances of the cut-based identifiers used in the previous CMS search are denoted with the green dots.

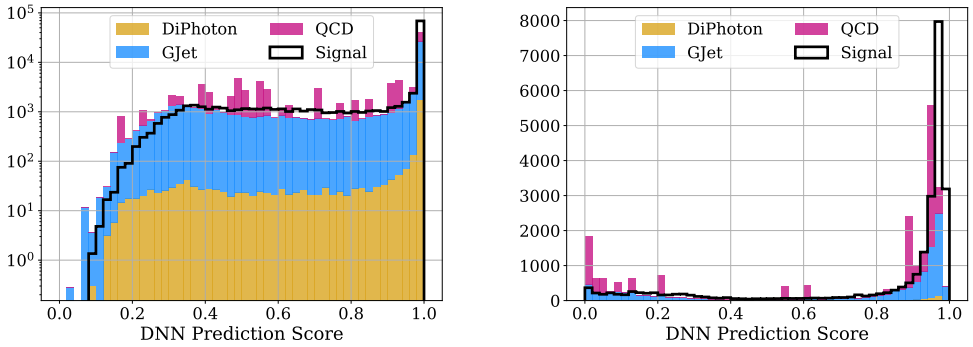


Figure 7.11: The distributions of DNN scores for subleading photons in the ECAL barrel (left) and endcap (right) on MC signal and background samples for 2017 and 2018. Signal consists of all GMSB signal samples with some removals based on sample points and displacements as described in Sec. 7.3.

to assign credits to players in a coalition [239], to approximate the importance of each input variable to the final decision of the DNN. As shown in Fig. 7.12, the most important input variables to the DNN are the isolation variables, followed by the $S_{major}$and S_{minor}variables. Low values of isolation will positively impact the model output, which means photons with low isolation are more likely to be from signal processes, as opposed to fake photons. Similarly, photons with high S_{major} values

are more likely to be from signal processes, as indicated by the red dots on the right side of the SHAP value spectrum in the S_{major} row.

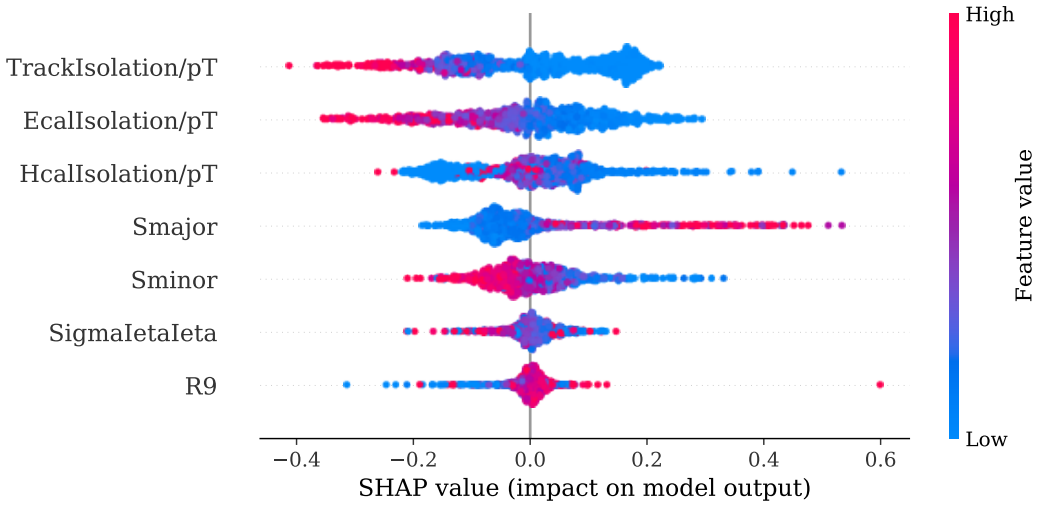


Figure 7.12: Importance ranking of input variables to the DNN for the leading photon. Color indicates the feature values with respect to its mean. SHAP values on the left of the vertical bar indicate events likely to be from background processes, while SHAP values on the right of the vertical bar indicate events likley to be from signal processes.

7.4 Photon time reconstruction

This search relies on the high precision of the CMS ECAL to measure the arrival time of photons, which is a key discriminant of the GMSB signal processes over the Standard Model background. This section first introduces the fundamentals of ECAL time reconstruction in each crystal and then in the photon supercluster. Afterwards, a dedicated measurement of the ECAL time resolution is discussed. Finally, the process of photon time calibration for MC is described.

When particles interact with the ECAL crystals, scintillation light is produced and recorded by a photodetector at the end of each crystal. Front-end electronics amplify and shape this signal into a pulse, which is then digitized into 10 consecutive samples spaced 25 ns apart, as shown in Fig. 7.13. The time of the pulse, T_{max}, is defined as when the pulse reaches its peak, which can be extracted by taking the weighted average of the estimated $t_{max,i}$ from each readout sample [240].

In particular, since the pulse shape is universal and independent of the maximum amplitude of the pulse, the time difference between any readout sample T and T_{max} can be represented as a function of the ratios of consecutive amplitude measurements,

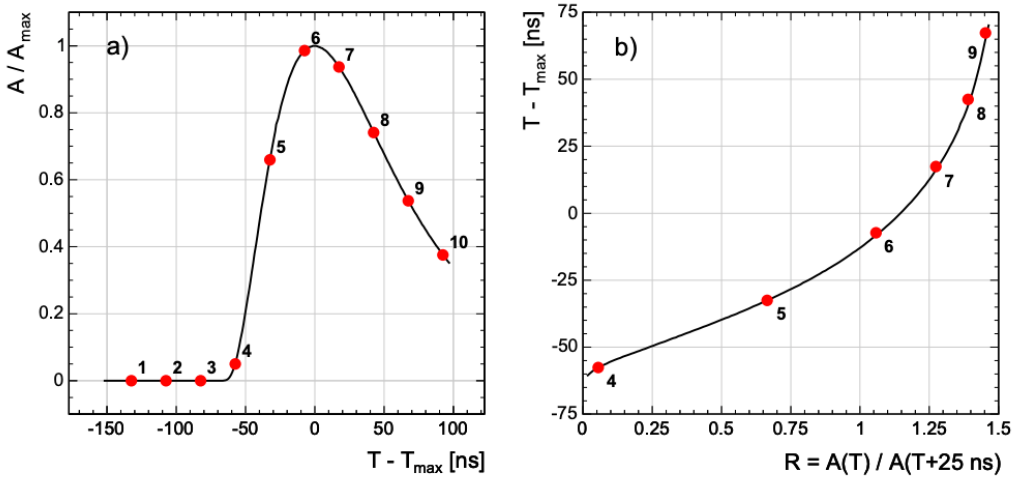


Figure 7.13: Example representation of an ECAL pulse shape. (a) ECAL pulse shape as a function of the difference between the time T of each readout sample along the pulse and the time T_{max} when the pulse reaches its maximum amplitude A_{max}. The red dots represent the amplitudes of ten discrete readout samples from a single pulse, normalized to the maximum amplitude. The solid line represents the average pulse shape, which is universal across all crystals to the first order. (b) An alternative pulse shape representation using the time difference from T_{max} as a function of the ratio of two consecutive readout samples' amplitudes [41].

$R = A(T)/A(T + 25\text{ns})$, as shown on the right of Fig. 7.13. A polynomial function is fit to these points. The estimated $t_{\text{max},i}$ can be obtained from each point by taking $t_{\text{max},i} = t_i - t(R_i)$, where t_i is the time of sample i and $t(R_i)$ is obtained from the fit where $R_i = A_i/A_{i+1}$. The time of a single crystal hit is determined to be the weighted average of $t_{\text{max},i}$ from each point, as shown in Eq. 7.10:

$$t_{\text{crystal}} = \frac{\sum \frac{t_{\text{max},i}}{\sigma^2_{\text{max},i}}}{\sum \frac{1}{\sigma^2_{\text{max},i}}}, \tag{7.10}$$

where $\sigma^2_{\text{max},i}$ is the uncertainty squared associated with each $t_{\text{max},i}$. This uncertainty term includes three independent contributions, which are added in quadrature: the noise fluctuations, the pedestal value's uncertainty, and the uncertainty due to truncation during digitization [41].

Once the time of each crystal hit is computed, the photon cluster time can be reconstructed by taking the weighted average of the times of all crystal hits in the cluster:

$$t_{\text{cluster}} = \frac{\sum \frac{t_{\text{crystal},i}}{\sigma^2_{\text{crystal},i}}}{\sum \frac{1}{\sigma^2_{\text{crystal},i}}}, \tag{7.11}$$

where $\sigma^2_{\text{crystal},i}$ is the time resolution of the crystal i, which is modeled by Eq. 7.12:

$$\sigma^2_{\text{crystal}} = \left(\frac{N}{A/\sigma_{\text{n}}}\right)^2 + C^2, \tag{7.12}$$

where A and σ_{n} are the A_{max} and standard deviation of the pedestal mean distribution for a given crystal, also known as the pedestal noise. The noise term, N, and the constant term, C, are 2 constants represent the statistical and systematic uncertainties, respectively, on measuring the time resolution of the crystal. These 2 constants can be extracted through an extensive study on the ECAL time performance described below.

To extract N and C, we first measure the time difference between 2 neighboring crystals, *i.e.*, $\Delta t = t_{\text{crystal},1} - t_{\text{crystal},2}$. These two crystals are required to have energy deposits within 20% of each other, *i.e.*, $0.8 < E_1/E_2 < 1.2$, belong to the same readout electronics, and are the highest energy pair within the photon cluster that share an edge. The pair is also required to have energies between 1 and 120 GeV to avoid noisy channels and gain switch effects in the readout.

We plot the Δt distribution in different bins of the effective crystal amplitude normalized to the pedestal noise of the crystal pair, given by Eq. 7.13:

$$A_{\text{eff}}/\sigma_{\text{n}} = \frac{(A_1/\sigma_{\text{n}_1})(A_2/\sigma_{\text{n}_2})}{\sqrt{(A_1/\sigma_{\text{n}_1})^2 + (A_2/\sigma_{\text{n}_2})^2}}, \tag{7.13}$$

where A_1/σ_{n_1}, A_2/σ_{n_2} are the amplitudes normalized by the pedestal noises for the first and second crystals, respectively.

For each bin of $A_{\text{eff}}/\sigma_{\text{n}}$, we fit the measured Δt distribution to a Gaussian function. We then extract the standard deviation σ from the fit results and trend the σ obtained from each $A_{\text{eff}}/\sigma_{\text{n}}$ bin versus $A_{\text{eff}}/\sigma_{\text{n}}$ itself. The trend is fit to the model of $\sigma(\Delta t)$, which is derived in Eq. 7.14 as the sum in quadrature of uncertainties of the 2 neighboring crystals:

$$
\begin{aligned}
\sigma^2(\Delta t) &= \sigma^2_{\text{crystal, 1}} + \sigma^2_{\text{crystal, 2}} \\
&= \left(\frac{N}{A_1/\sigma_{\text{n}_1}}\right)^2 + C^2 + \left(\frac{N}{A_2/\sigma_{\text{n}_2}}\right)^2 + C^2 \\
&= N^2 \left(\frac{(A_1/\sigma_{\text{n}_1})^2 + (A_2/\sigma_{\text{n}_2})^2}{(A_1/\sigma_{\text{n}_1})^2 (A_2/\sigma_{\text{n}_2})^2}\right) + 2C^2 \\
&= \left(\frac{N}{A_{\text{eff}}/\sigma_{\text{n}}}\right)^2 + 2C^2.
\end{aligned}
\tag{7.14}
$$

N and C are finally extracted from the fit results of Eq. 7.14, which describes the local time resolution as a function of $A_{\text{eff}}/\sigma_{\text{n}}$.

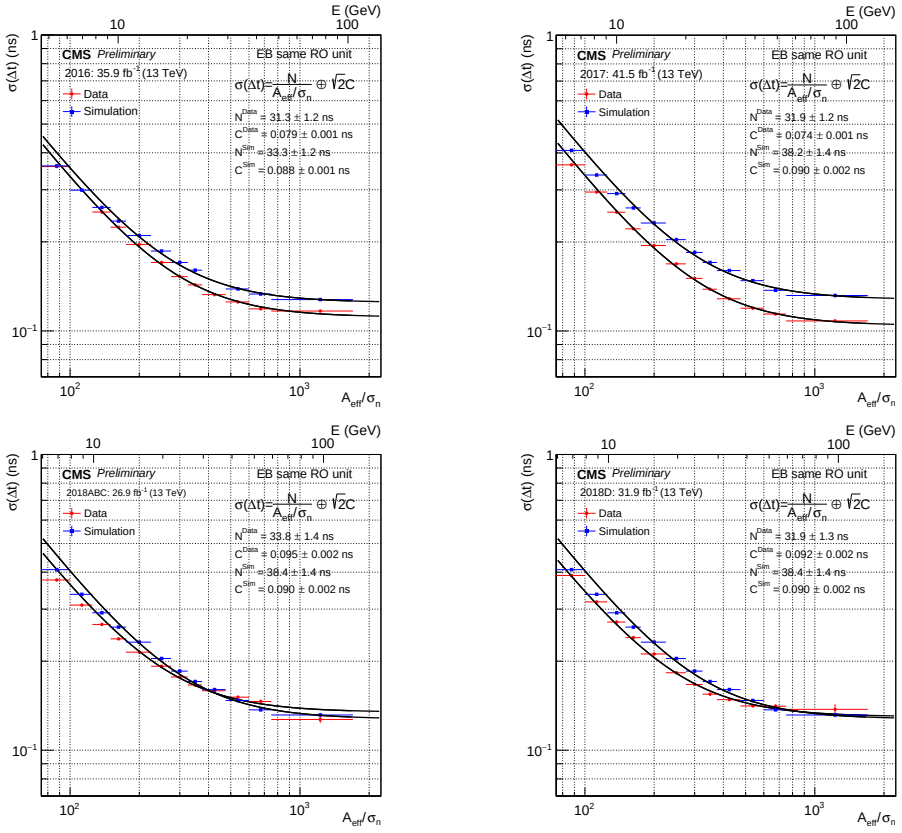


Figure 7.14: Local time resolution of ECAL versus the effective amplitude of the neighboring crystals from the same read-out electronics measured in data and simulation for 2016 (upper left), 2017 (upper right), 2018 eras ABC (lower left), and 2018 era D (lower right). The data in 2018 era D is centrally processed differently by CMS, therefore it requires a separate analysis from 2018 data in previous eras.

The fit results for data and simulated events of different years are shown in Fig. 7.14. As summary in Table 7.5, the noise term is fairly stable over the years while the constant term C degrades significantly in 2018 data.

Table 7.5: Fit results for the ECAL timing resolution parameters from data in different year periods.

	Parameters	
Era	N	C
2016 Data	31.3 ± 1.2	0.079 ± 0.001
2017 Data	31.9 ± 1.2	0.074 ± 0.001
2018 Data (eras ABC)	33.8 ± 1.4	0.095 ± 0.002
2018 Data (era D)	31.9 ± 1.3	0.092 ± 0.002

After N and C are extracted to properly reconstruct the photon cluster time, the last step is to calibrate the timing in simulated events to match the data. As shown in [FIGURE], the photon cluster time in MC are not correctly simulated, as its distribution has shifted mean and higher standard deviation compared to those of data's distribution. To correct the photon arrival time in simulated events, we derive correction with $Z \rightarrow e^+e^-$ simulated events and measure t_{cluster} of the leading electrons and positrons in bins of $e^{\pm}$ energy. For each bin, we fit the distribution to a single Gaussian, then extract the mean and sigma of the fit results and trend them as a function of the $e^{\pm}$ energy. We compute the difference in mean $e^{\pm}$ time between data and simulated events in each bin, then shift the time in simulated events by this difference to match the data. Similarly, we compute the difference in quadrature between the standard deviations of distributions in data and simulation, then use this difference to smear the time distribution in simulated events. Illustration of the correction for 2017 is shown in Fig. 7.15, and examples of the electron arrival time before and after correction are shown in Fig. 7.16.

7.5 Event selection

In 2016 data, we select events with at least 2 photons. The leading photon is required to have p_{T} above 70 GeV and in the ECAL barrel region, while the subleading photon is required to have p_{T} above 40 GeV. Each photon is then required to pass the trigger selection criteria, summarized in Table 7.4. We also apply a conversion-safe electron veto cut [241] on the leading photon to remove electrons that are misidentified as photons. Finally, we apply the DNN selection described in Sec. 7.3 on the leading photon such that the false-positive rate is 0.5 and the true-positive rate is 0.88. As

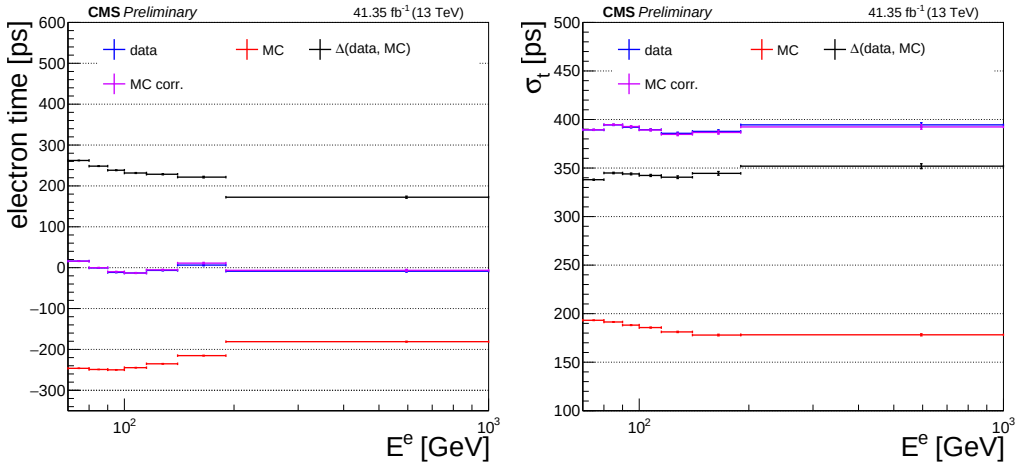


Figure 7.15: Timing correction for simulation using $Z \rightarrow ee$ events and data for the 2017 data-taking period. On the left is the correction of the mean electron cluster's arrival times in different electron's energy bins. On the right is the correction of the standard deviations of distributions of electron cluster's arrival time in different electron energy bins. The purple lines overlapping with the blue lines indicate that after correction, the mean and standard deviation of electron arrival time distributions from simulation match those from data.

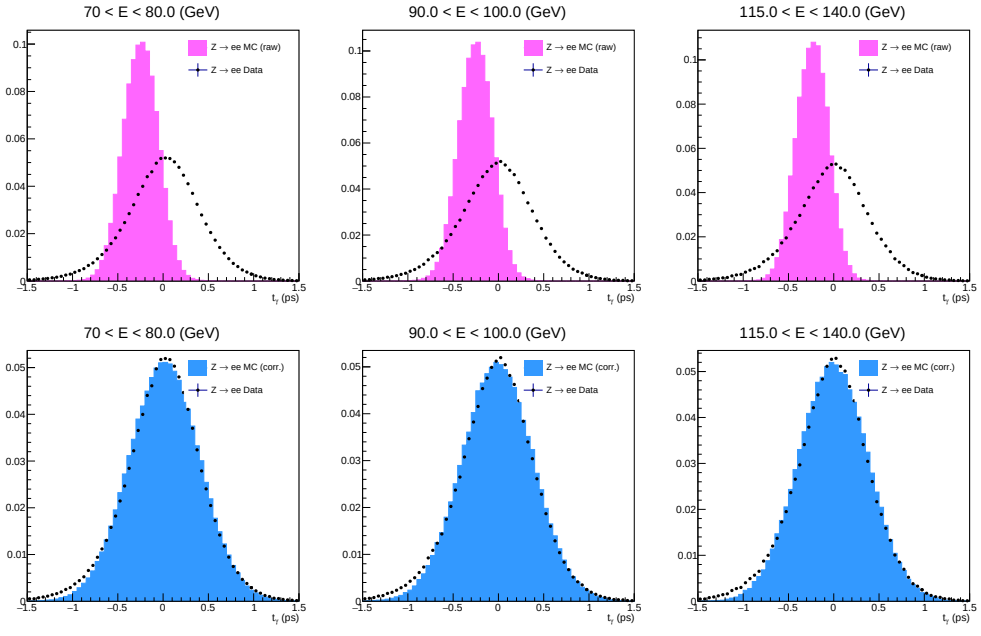


Figure 7.16: Examples of electron arrival times before (top) and after (bottom) the correction procedure in different energy bins.

mentioned in Sec. 7.3, there is no further selection on the subleading photon because the trigger requirement on the second photon is already too tight.

Table 7.6: Summary of event selection criteria in this search.

Objects	Selection criteria	
	2016 data	2017 and 2018 data
Trigger	Dedicated signal trigger fired	
$p_{\mathrm{T}}^{\mathrm{miss}}$ filters	Pass all $p_{\mathrm{T}}^{\mathrm{miss}}$ filters	
Number of photons	≥ 2 photons	≥ 1 photon
Leading photon	$p_{\mathrm{T}} > 70$ GeV $\lvert\eta\rvert < 1.4442$ Pass all trigger requirements (Table 7.4)	
	DNN score > 0.089 Conversion-safe electron veto	DNN score > 0.127 ΔR(track, γ) > 0.2 $\mathrm{p}_T^{\mathrm{track}} > 5$ GeV
Subleading photon	$p_{\mathrm{T}} > 40$ GeV $\lvert\eta\rvert < 1.4442$ or $1.566 < \lvert\eta\rvert < 2.5$ Pass all trigger requirements (Table 7.4)	**Single-photon category**: explicitly veto events passing subleading photon selections defined in 2016 data. **Diphoton category**: require events passing subleading photon selections defined in 2016 data, plus: EB γ: DNN score > 0.473 EE γ: DNN score > 0.130
Jets	nJets ≥ 3 $p_{\mathrm{T}} > 30$ GeV $\lvert\eta\rvert < 3.0$ Passing jet cuts (Table 7.3)	
	No H_T requirement	$\mathrm{H}_T > 400$ GeV

In 2017 and 2018 data, we select events with at least 1 photon due to a new trigger menu. The selections on the leading photon is similar to those in 2016, except that the conversion-safe electron veto is replaced with a track veto as required by the trigger: within the radius $\Delta R = 0.2$ around the photon, there must be no track with p_{T} above 5 GeV. For the subleading photon, we apply the trigger selections described in Table 7.4, and then apply the DNN selection such that the false-positive rate is 0.5. Using this working point, the true-positive rate for the subleading photon is 0.94 for photons in both barrel and endcap regions of the ECAL.

For all years, we further require events to have at least three jets with p_T above 30 GeV, $|\eta| < 3.0$ and pass the jet identification criteria listed in Table 7.3 to suppress non-collisional background events.

In 2017 and 2018 data, we divide the signal region into two categories: a single photon category, which requires exclusively 1 photon in the event, and a diphoton category, which requires at least 2 photons in the event. For 2016 data, since the trigger requires at least 2 photons, only the diphoton category is used. Table 7.6 summarizes the selections used in this search for each category. The signal efficiency along with the selection flow are listed in Tables 7.7-7.10 for 2016, and in Tables 7.11-7.14 for 2017 and 2018.

Table 7.7: Event selection efficiency for GMSB SPS8 $c\tau = 10$ cm and varying Λ (unit of efficiency: %; unit of Λ : TeV) using the 2016 event selection flow summarized in Table 7.6.

	$\Lambda = 100$	$\Lambda = 200$	$\Lambda = 300$	$\Lambda = 400$
-	100.00 ± 0.00	100.00 ± 0.00	100.00 ± 0.00	100.00 ± 0.00
+ Signal trigger	62.38 ± 0.32	61.79 ± 0.32	62.42 ± 0.32	65.12 ± 0.33
+ $\gamma_1 p_T > 70$ GeV	59.25 ± 0.31	61.13 ± 0.32	62.23 ± 0.32	65.04 ± 0.33
+ $\gamma_1 \|\eta\| < 1.4442$	59.25 ± 0.31	61.13 ± 0.32	62.23 ± 0.32	65.04 ± 0.33
+ γ_1 DNN ID	35.88 ± 0.22	47.57 ± 0.27	54.45 ± 0.29	58.32 ± 0.30
+ γ_1 electron veto	32.90 ± 0.21	44.21 ± 0.25	51.09 ± 0.28	54.66 ± 0.29
+ nJets ≥ 3	29.01 ± 0.19	34.66 ± 0.22	38.47 ± 0.23	40.78 ± 0.24
+ p_T^{miss} filters	28.73 ± 0.19	34.25 ± 0.22	38.06 ± 0.23	40.33 ± 0.24
+ $\gamma_2 p_T > 40$ GeV	26.52 ± 0.18	32.44 ± 0.21	36.26 ± 0.22	38.67 ± 0.23

Table 7.8: Event selection efficiency for GMSB SPS8 $c\tau = 100$ cm and varying Λ (unit of efficiency: %; unit of Λ : TeV) using the 2016 event selection flow summarized in Table 7.6.

	$\Lambda = 100$	$\Lambda = 200$	$\Lambda = 300$	$\Lambda = 400$
-	100.00 ± 0.00	100.00 ± 0.00	100.00 ± 0.00	100.00 ± 0.00
+ Signal trigger	35.62 ± 0.16	40.38 ± 0.19	45.26 ± 0.21	48.77 ± 0.23
+ $\gamma_1 p_T > 70$ GeV	33.16 ± 0.15	39.70 ± 0.19	44.96 ± 0.21	48.61 ± 0.23
+ $\gamma_1 \|\eta\| < 1.4442$	33.16 ± 0.15	39.70 ± 0.19	44.96 ± 0.21	48.61 ± 0.23
+ γ_1 DNN ID	19.07 ± 0.11	30.04 ± 0.16	38.69 ± 0.19	42.98 ± 0.21
+ γ_1 electron veto	16.27 ± 0.10	26.92 ± 0.15	35.43 ± 0.18	39.70 ± 0.20
+ nJets ≥ 3	14.06 ± 0.09	20.40 ± 0.12	25.93 ± 0.15	28.84 ± 0.17
+ p_T^{miss} filters	13.23 ± 0.09	19.31 ± 0.12	24.69 ± 0.14	27.33 ± 0.16
+ $\gamma_2 p_T > 40$ GeV	11.93 ± 0.08	17.99 ± 0.12	23.26 ± 0.14	26.09 ± 0.16

Table 7.9: Event selection efficiency for GMSB SPS8 $c\tau$ = 1000 cm and varying Λ (unit of efficiency: %; unit of Λ : TeV) using the 2016 event selection flow summarized in Table 7.6.

	Λ = 100	Λ = 200	Λ = 300	Λ = 400
-	100.00 ± 0.00	100.00 ± 0.00	100.00 ± 0.00	100.00 ± 0.00
+ Signal trigger	10.02 ± 0.11	10.17 ± 0.11	11.19 ± 0.11	12.44 ± 0.12
+ $\gamma_1 p_{\mathrm{T}} > 70$ GeV	8.97 ± 0.10	9.65 ± 0.10	10.79 ± 0.11	12.14 ± 0.12
+ $\gamma_1 \lvert\eta\rvert < 1.4442$	8.97 ± 0.10	9.65 ± 0.10	10.79 ± 0.11	12.14 ± 0.12
+ γ_1 DNN ID	4.41 ± 0.07	6.30 ± 0.08	8.41 ± 0.10	9.74 ± 0.10
+ γ_1 electron veto	2.07 ± 0.05	3.76 ± 0.06	5.43 ± 0.08	6.30 ± 0.08
+ nJets ≥ 3	1.78 ± 0.04	2.68 ± 0.05	3.48 ± 0.06	3.89 ± 0.06
+ $p_{\mathrm{T}}^{\mathrm{miss}}$ filters	1.69 ± 0.04	2.54 ± 0.05	3.21 ± 0.06	3.58 ± 0.06
+ $\gamma_2 p_{\mathrm{T}} > 40$ GeV	1.47 ± 0.04	2.21 ± 0.05	2.81 ± 0.05	3.24 ± 0.06

Table 7.10: Event selection efficiency for GMSB SPS8 $c\tau$ = 10 000 cm and varying Λ (unit of efficiency: %; unit of Λ : TeV) using the 2016 event selection flow summarized in Table 7.6.

	Λ = 100	Λ = 200	Λ = 300	Λ = 400
-	100.00 ± 0.00	100.00 ± 0.00	100.00 ± 0.00	100.00 ± 0.00
+ Signal trigger	6.36 ± 0.05	5.65 ± 0.05	5.36 ± 0.05	6.01 ± 0.05
+ $\gamma_1 p_{\mathrm{T}} > 70$ GeV	5.63 ± 0.05	5.21 ± 0.05	5.01 ± 0.05	5.72 ± 0.05
+ $\gamma_1 \lvert\eta\rvert < 1.4442$	5.63 ± 0.05	5.21 ± 0.05	5.01 ± 0.05	5.72 ± 0.05
+ γ_1 DNN ID	2.65 ± 0.03	3.01 ± 0.04	3.60 ± 0.04	4.33 ± 0.05
+ γ_1 electron veto	0.56 ± 0.01	0.81 ± 0.02	0.96 ± 0.02	1.17 ± 0.02
+ nJets ≥ 3	0.50 ± 0.01	0.58 ± 0.02	0.53 ± 0.01	0.61 ± 0.02
+ $p_{\mathrm{T}}^{\mathrm{miss}}$ filters	0.50 ± 0.01	0.56 ± 0.02	0.52 ± 0.01	0.60 ± 0.02
+ $\gamma_2 p_{\mathrm{T}} > 40$ GeV	0.42 ± 0.01	0.48 ± 0.01	0.43 ± 0.01	0.49 ± 0.02

7.6 Data-driven background estimation

This search relies on two main discriminating variables: the photon cluster time, t_γ and the missing transverse momentum, $p_{\mathrm{T}}^{\mathrm{miss}}$. The photon cluster time is sensitive to non-collisional background processes, such as from beam halo or cosmic ray muons, which can create late-arrival photons that are indistinguishable from GMSB photons. These non-collisional background processes are not accounted for in simulated events. Furthermore, imperfect simulation of the detector response can lead to inaccurate simulation of $p_{\mathrm{T}}^{\mathrm{miss}}$. For these reasons, we use a data-driven approach called the ABCD method to estimate background contributions and extract signal strengths.

Table 7.11: Event selection efficiency for GMSB SPS8 $c\tau$ = 10 cm and varying Λ (unit of efficiency: %; unit of Λ : TeV) using the 2017 event selection flow summarized in Table 7.6.

	Λ = 100	Λ = 200	Λ = 300	Λ = 400
-	100.00 ± 0.00	100.00 ± 0.00	100.00 ± 0.00	100.00 ± 0.00
+ Signal trigger	54.37 ± 0.29	71.92 ± 0.37	79.60 ± 0.38	82.88 ± 0.39
+ $\gamma_1 p_{\mathrm{T}} > 70$ GeV	54.37 ± 0.29	71.92 ± 0.37	79.60 ± 0.38	82.88 ± 0.39
+ $\gamma_1 \lvert\eta\rvert < 1.4442$	54.37 ± 0.29	71.92 ± 0.37	79.60 ± 0.38	82.88 ± 0.39
+ γ_1 DNN ID	26.81 ± 0.18	49.56 ± 0.29	63.56 ± 0.32	69.71 ± 0.34
+ γ_1 track veto	21.96 ± 0.16	40.31 ± 0.25	51.44 ± 0.28	56.29 ± 0.30
+ nJets ≥ 3	19.77 ± 0.15	31.37 ± 0.21	39.69 ± 0.24	43.36 ± 0.25
+ $p_{\mathrm{T}}^{\mathrm{miss}}$ filters	19.51 ± 0.15	30.97 ± 0.21	39.17 ± 0.23	42.74 ± 0.25
+ $\gamma_2 p_{\mathrm{T}} > 40$ GeV	7.39 ± 0.09	12.47 ± 0.12	16.21 ± 0.14	16.98 ± 0.14
+ γ_2 DNN ID	6.94 ± 0.09	11.94 ± 0.12	15.59 ± 0.13	16.25 ± 0.14
+ $\mathrm{H}_T \geq 400$ TeV	5.87 ± 0.08	6.33 ± 0.09	10.67 ± 0.11	13.18 ± 0.12

Table 7.12: Event selection efficiency for GMSB SPS8 $c\tau$ = 100 cm and varying Λ (unit of efficiency: %; unit of Λ : TeV) using the 2017 event selection flow summarized in Table 7.6.

	Λ = 100	Λ = 200	Λ = 300	Λ = 400
-	100.00 ± 0.00	100.00 ± 0.00	100.00 ± 0.00	100.00 ± 0.00
+ Signal trigger	34.19 ± 0.16	51.40 ± 0.21	60.55 ± 0.23	64.64 ± 0.36
+ $\gamma_1 p_{\mathrm{T}} > 70$ GeV	34.19 ± 0.16	51.40 ± 0.21	60.55 ± 0.23	64.64 ± 0.36
+ $\gamma_1 \lvert\eta\rvert < 1.4442$	34.19 ± 0.16	51.40 ± 0.21	60.55 ± 0.23	64.64 ± 0.36
+ γ_1 DNN ID	17.80 ± 0.11	38.04 ± 0.17	50.49 ± 0.20	55.75 ± 0.32
+ γ_1 track veto	15.57 ± 0.10	33.87 ± 0.16	45.04 ± 0.19	49.63 ± 0.30
+ nJets ≥ 3	13.97 ± 0.09	25.44 ± 0.13	33.31 ± 0.15	36.70 ± 0.24
+ $p_{\mathrm{T}}^{\mathrm{miss}}$ filters	12.55 ± 0.09	23.00 ± 0.12	30.24 ± 0.14	33.30 ± 0.23
+ $\gamma_2 p_{\mathrm{T}} > 40$ GeV	4.68 ± 0.05	8.91 ± 0.07	12.01 ± 0.08	13.09 ± 0.13
+ γ_2 DNN ID	4.23 ± 0.05	8.23 ± 0.07	11.17 ± 0.08	12.17 ± 0.13
+ $\mathrm{H}_T \geq 400$ TeV	3.50 ± 0.04	4.30 ± 0.05	7.55 ± 0.07	9.71 ± 0.11

We define in a two-dimensional distribution of $p_{\mathrm{T}}^{\mathrm{miss}}$ and t_γ four rectangular bins A, B, C, and D, where bin A contains events with low $p_{\mathrm{T}}^{\mathrm{miss}}$ and low t_γ, bin B high $p_{\mathrm{T}}^{\mathrm{miss}}$ and low t_γ, bin C high $p_{\mathrm{T}}^{\mathrm{miss}}$ and high t_γ, and bin D low $p_{\mathrm{T}}^{\mathrm{miss}}$ and high t_γ, as shown in Fig. 7.17. The signal is most enriched in bin C, while bin A contains most background.

The ABCD method assume two variables $p_{\mathrm{T}}^{\mathrm{miss}}$ and t_γ are independent, which will be verified later on. Therefore, the prediction for the background yield in bin C can be defined as:

Table 7.13: Event selection efficiency for GMSB SPS8 $c\tau$ = 1000 cm and varying Λ (unit of efficiency: %; unit of Λ : TeV) using the 2017 event selection flow summarized in Table 7.6.

	$\Lambda = 100$	$\Lambda = 200$	$\Lambda = 300$	$\Lambda = 400$
-	100.00 ± 0.00	100.00 ± 0.00	100.00 ± 0.00	100.00 ± 0.00
+ Signal trigger	6.78 ± 0.09	11.35 ± 0.12	14.62 ± 0.13	16.32 ± 0.14
+ $\gamma_1 p_T > 70$ GeV	6.78 ± 0.09	11.35 ± 0.12	14.62 ± 0.13	16.32 ± 0.14
+ $\gamma_1 \lvert\eta\rvert < 1.4442$	6.78 ± 0.09	11.35 ± 0.12	14.62 ± 0.13	16.32 ± 0.14
+ γ_1 DNN ID	3.30 ± 0.06	8.08 ± 0.10	11.66 ± 0.11	13.49 ± 0.13
+ γ_1 track veto	2.71 ± 0.05	6.94 ± 0.09	9.96 ± 0.10	11.41 ± 0.12
+ nJets ≥ 3	2.49 ± 0.05	5.08 ± 0.08	6.93 ± 0.09	7.58 ± 0.09
+ p_T^{miss} filters	2.35 ± 0.05	4.74 ± 0.07	6.36 ± 0.08	6.88 ± 0.09
+ $\gamma_2 p_T > 40$ GeV	0.84 ± 0.03	1.45 ± 0.04	1.96 ± 0.04	2.21 ± 0.05
+ γ_2 DNN ID	0.71 ± 0.03	1.26 ± 0.04	1.70 ± 0.04	1.96 ± 0.05
+ $H_T \geq 400$ TeV	0.64 ± 0.03	0.73 ± 0.03	1.07 ± 0.03	1.45 ± 0.04

Table 7.14: Event selection efficiency for GMSB SPS8 $c\tau$ = 10 000 cm and varying Λ (unit of efficiency: %; unit of Λ : TeV) using the 2017 event selection flow summarized in Table 7.6.

	$\Lambda = 100$	$\Lambda = 200$	$\Lambda = 300$	$\Lambda = 400$
-	100.00 ± 0.00	100.00 ± 0.00	100.00 ± 0.00	100.00 ± 0.00
+ Signal trigger	1.99 ± 0.05	2.72 ± 0.05	3.46 ± 0.06	4.08 ± 0.07
+ $\gamma_1 p_T > 70$ GeV	1.99 ± 0.05	2.72 ± 0.05	3.46 ± 0.06	4.08 ± 0.07
+ $\gamma_1 \lvert\eta\rvert < 1.4442$	1.99 ± 0.05	2.72 ± 0.05	3.46 ± 0.06	4.08 ± 0.07
+ γ_1 DNN ID	0.81 ± 0.03	1.59 ± 0.04	2.36 ± 0.05	2.91 ± 0.05
+ γ_1 track veto	0.49 ± 0.02	1.05 ± 0.03	1.53 ± 0.04	1.85 ± 0.04
+ nJets ≥ 3	0.47 ± 0.02	0.78 ± 0.03	1.00 ± 0.03	1.18 ± 0.03
+ p_T^{miss} filters	0.47 ± 0.02	0.77 ± 0.03	0.98 ± 0.03	1.16 ± 0.03
+ $\gamma_2 p_T > 40$ GeV	0.14 ± 0.01	0.21 ± 0.01	0.26 ± 0.02	0.30 ± 0.02
+ γ_2 DNN ID	0.14 ± 0.01	0.21 ± 0.01	0.26 ± 0.02	0.30 ± 0.02
+ $H_T \geq 400$ TeV	0.12 ± 0.01	0.14 ± 0.01	0.16 ± 0.01	0.22 ± 0.01

$$N_C^{pred} = \frac{N_B^{obs} \times N_D^{obs}}{N_A^{obs}}, \tag{7.15}$$

where N_X^{obs} is the observed data in bin X. This is also known as the "classical ABCD" method. However, the signals we search for in this analysis include a wide range of mass values and lifetimes, resulting in signal events with different values in p_T^{miss} and t_γ that scatter over all 4 bins A, B, C, and D. Therefore, we use a "modified ABCD" method, which is described as follows:

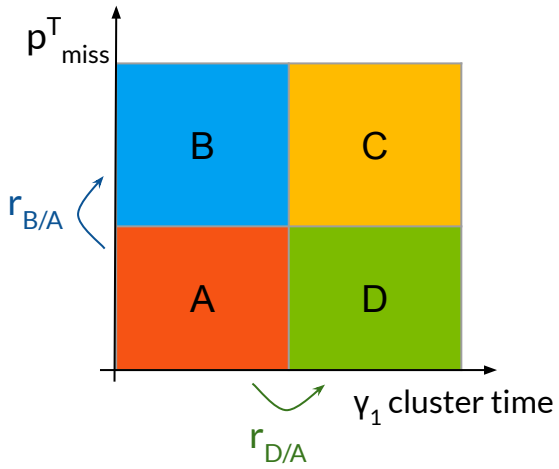


Figure 7.17: Illustration of the four bins A, B, C, and D in the two-dimensional distribution of $p_{\mathrm{T}}^{\mathrm{miss}}$ and t_γ in the ABCD method.

- To extract the observed upper limit on the signal strength, we use a system of four equations with four unknowns as shown in Eq. 7.16:

$$\begin{aligned} N_{\mathrm{A}}^{\mathrm{obs}} &= \mathrm{Bkg}_{\mathrm{A}} + \mu \times \mathrm{Sig}_{\mathrm{A}} \\ N_{\mathrm{B}}^{\mathrm{obs}} &= r_{\mathrm{B/A}} \times \mathrm{Bkg}_{\mathrm{A}} + \mu \times \mathrm{Sig}_{\mathrm{B}} \\ N_{\mathrm{C}}^{\mathrm{obs}} &= r_{\mathrm{B/A}} \times r_{\mathrm{D/A}} \times \mathrm{Bkg}_{\mathrm{A}} + \mu \times \mathrm{Sig}_{\mathrm{C}} \\ N_{\mathrm{D}}^{\mathrm{obs}} &= r_{\mathrm{D/A}} \times \mathrm{Bkg}_{\mathrm{A}} + \mu \times \mathrm{Sig}_{\mathrm{D}}, \end{aligned} \tag{7.16}$$

  where:

  - $\mathrm{Bkg}_{\mathrm{A}}$ is the background yield in bin A (unknown),
  - $r_{\mathrm{B/A}}$ is the ratio between numbers of background events in bin B and A (unknown),
  - $r_{\mathrm{D/A}}$ is the ratio between numbers of background events in bin D and A (unknown),
  - μ is the overall signal strength (unknown),
  - $N_{\mathrm{X}}^{\mathrm{obs}}$ is the observed yields from data in bin X (known),
  - $\mathrm{Sig}_{\mathrm{X}}$ is the predicted signal yields in bin X, taken directly from signal simulation (known).

  We then perform a simultaneous maximum-likelihood fit to the observed data in all four bins, extracting the four unknown variables.

- To extract the expected upper limit on the signal strength, as the data is "blinded" to avoid potential biases, we use shape templates to compute the expected background yields in bin B, C, and D from the number of events in bin A, which is the control region. In particular, we obtain the shape template for $p_{\mathrm{T}}^{\mathrm{miss}}$ from data, requiring $t_\gamma < 1$ ns. Similarly, the shape template for t_γ is obtained from data with $p_{\mathrm{T}}^{\mathrm{miss}} < 100$ GeV, as shown in Fig. 7.18. We compute $r_{\mathrm{B/A}}$ and $r_{\mathrm{D/A}}$ from the $p_{\mathrm{T}}^{\mathrm{miss}}$ and t_γ shape templates, respectively, by dividing number of events on the right of the vertical line by the number of events on the left of the vertical line. We proceed to compute the expected background yields in bin B, C, and D following Eq. 7.16, and set the observed event yields to be the same as the expected event yields. The expected upper limit on the signal strength is computed from the maximum likelihood fit to all four bins.

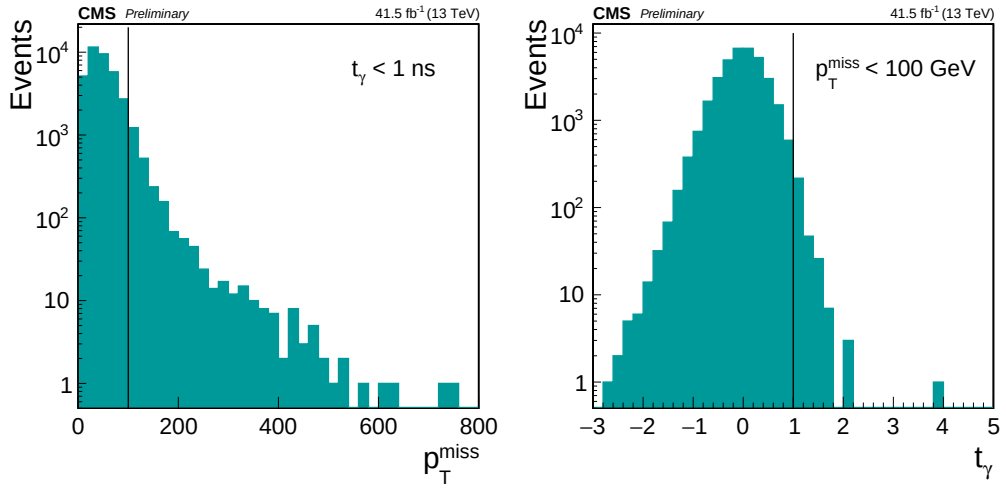


Figure 7.18: Example shape templates to extract $r_{\mathrm{B/A}}$ and $r_{\mathrm{D/A}}$ to compute the expected background yields in bin B, C, and D from bin A in the modified ABCD method to obtain the expected upper limit on the signal strength. The vertical lines represent the bin boundaries. $r_{\mathrm{B/A}}$ is computed as the ratio between number of events on the right of the bin boundary and number of events on the left of the bin boundary from the $p_{\mathrm{T}}^{\mathrm{miss}}$ plot; $r_{\mathrm{D/A}}$ is computed as the ratio between number of events on the right of the bin boundary and number of events on the left of the bin boundary on the t_γ plot.

We conduct a correlation study between $p_{\mathrm{T}}^{\mathrm{miss}}$ and t_γ to verify that they are independent by defining a control region from data, which has the same requirements as the signal region, but the DNN photon ID on the leading photon is inverted. We measure the Pearson's correlation coefficient r between $p_{\mathrm{T}}^{\mathrm{miss}}$ and t_γ, which is defined as:

$$r = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2 \sum(y_i - \bar{y})^2}}, \tag{7.17}$$

where x_i, y_i are the values of $p_{\mathrm{T}}^{\mathrm{miss}}$ and t_y, respectively, and $\bar{x}$, $\bar{y}$ are their corresponding mean values. We measure the correlation coefficients for different time regions, which are listed in Table. 7.15. The largest correlation observed is around -2.4% on the whole t_γ region. In the high t_γ region, this correlation reduces to -1.1%. These small correlations are taken into account as systematic uncertainties. We also observe that the shapes of t_γ distributions in different $p_{\mathrm{T}}^{\mathrm{miss}}$ bins are similar, as shown in Fig. 7.19.

Table 7.15: Pearson's correlation coefficient r between $p_{\mathrm{T}}^{\mathrm{miss}}$ and t_γ in different t_γ regions, computed on the control region from 2016 data, where the DNN photon ID requirement on the leading photon is inverted.

t_y region [ns]	Number of events	Pearson's r
(-10, 25)	23802	-0.024 ± 0.006
(-2, 25)	23795	-0.018 ± 0.006
(1, 25)	255	-0.011 ± 0.063

7.7 Systematic uncertainties

The dominant uncertainties of this search come from the uncertainties in the background and signal fit parameters. In many scenarios, the predicted background yields in bin C are less than 1, resulting in a large statistical uncertainty, which is taken automatically into account when computing cross section's upper limit.

Subdominant uncertainties in this analysis mainly come from instrumental effects, which are included as nuissance parameters of the likelihood. These include the integrated luminosity uncertainty, which affects the overall signal normalization in each of the four bins. For data collected in 2016, 2017, and 2018, the systematic uncertainties from integrated luminosity are 2.5%, 2.3%, and 2.5%, respectively [134–136]. The uncertainty from trigger in 2016 is measured to be 2%. No uncertainty is assigned to the trigger efficiency in 2017 and 2018, as the difference between the trigger response in data and simulation is less than one percent. The uncertainty on the DNN-based photon identification is estimated to be 5%. Another type of systematic uncertainties arises from imperfect simulation of the detector response, including uncertainties on the photon energy scale and resolution, jet energy scale and resolution, and the photon cluster time bias and resolution. Finally,

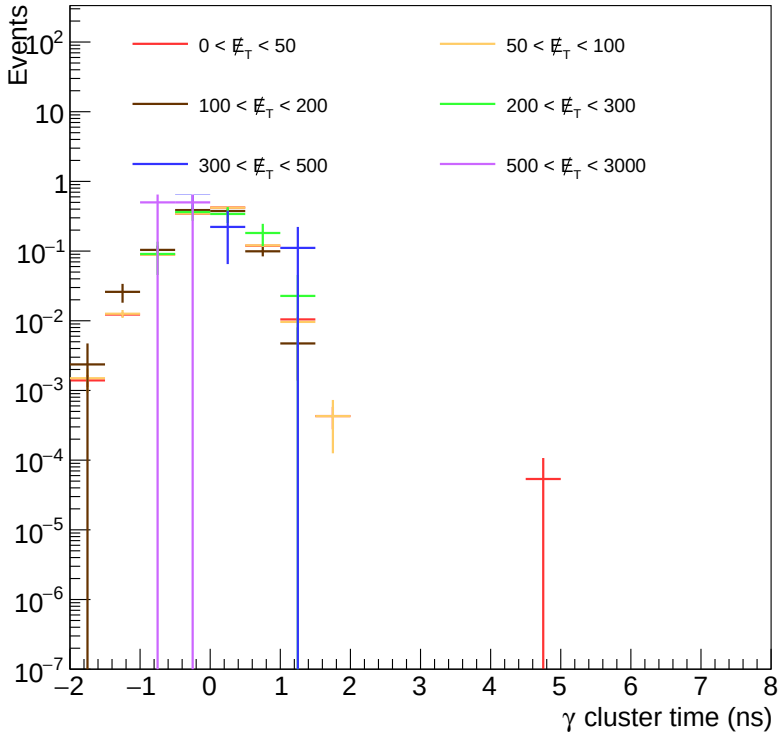


Figure 7.19: Distributions of t_γ in different $p_{\mathrm{T}}^{\mathrm{miss}}$ bins from 2016 data in the control region where the DNN photon ID requirement on the leading photon is inverted. All distributions are normalized to unity.

the systematic uncertainties accounting for the dependence between $p_{\mathrm{T}}^{\mathrm{miss}}$ and t_γ is measured from a study with γ+jets control regions, where we compare the predicted background events in bin C ($N_{\mathrm{C}}^{\mathrm{pred}} = N_{\mathrm{B}} \cdot N_{\mathrm{D}}/N_{\mathrm{A}}$), with the actual events in bin C (N_{C}).

Table 7.16 summarizes the systematic uncertainties in this analysis.

7.8 Results

After selecting events as described in Sec. 7.5, we estimate the predicted signal yields in bin A, B, C, and D, as listed in Tables 7.17-7.21 for each year during Run 2. As the $c\tau$ of the signal model increases, the detector's acceptance decreases, resulting in the decrease of the expected signal yields. Similarly, the Λ of the signal model is inversely proportional to its production cross section, therefore the expected signal yields also decrease as Λ increases.

Table 7.16: Summary of systematics and their assigned values in this analysis.

Systematic	Sig/Bkg	Bins	2016	2017	2018	Correlation
Luminosity	Sig	A, B, C, D	2.5%	2.3%	2.5%	Uncorrelated
Photon energy scale	Sig	A, B, C, D	1%	2%	2%	100% correlated
Photon energy resolution	Sig	A, B, C, D	1%	1%	1%	100% correlated
Jet energy scale	Sig	A, B, C, D	1.5%	2%	2%	100% correlated
Jet energy resolution	Sig	A, B, C, D	1.5%	1.5%	1.5%	Uncorrelated
Photon time bias	Sig	A, B, C, D	1.5%	1%	1%	100% correlated
Photon time resolution	Sig	A, B, C, D	0.5%	0.5%	0.5%	100% correlated
Trigger efficiency	Sig	A, B, C, D	2%	-	-	N/A
Photon identification	Sig	A, B, C, D	2%	3%	5%	100% correlated
Closure in bin C ($c\tau < 10\,\text{cm}$)	Bkg	C	2%	3.5%	3.5%	100% correlated
Closure in bin C ($c\tau > 10\,\text{cm}$)	Bkg	C	90%	90%	90%	100% correlated

To avoid potential bias, observed data in bins B, C, and D remain blinded throughout the whole analysis and will only be uncovered after the analysis is finalized, which are not included in this thesis. Observed data in the control region (bin A), along with the predicted background yields in bin B, C, and D are showed in Tables 7.22, 7.23, and 7.24 for the year 2016, 2017, and 2018, respectively.

The predicted signal and background yields, along with the uncertainties described in Sec. 7.7, are used to compute the expected upper limit on the signal strength at 95% confidence level (CL) using the asymptotic formulae for likelihood-based tests [139]. If the upper limit on the signal strength of a signal point is less than 1, the signal point is excluded. The upper limit on the signal strength is multiplied with the theoretical production cross section to established the 95% CL upper limits on the signals' cross sections, which are shown in Figs. 7.20-7.21 as functions of the neutralino's masses for different $c\tau$.

At large $c\tau$, the sensitivity is improved significantly with the 2017 and 2018 data compared with 2016, as shown in Fig. 7.22, because of the dedicated displaced single-photon trigger. The 2018 analysis also shows some improvements at very low $c\tau$due to the better time resolution in the 2018 A, B, and C data-taking eras.

Table 7.17: Predicted signal yields in bins A, B, C, and D using the event selection described in Sec. 7.5 for the year 2016 in different Λ and $c\tau$ values.

$c\tau$ (cm)	Yield in bin A	Yield in bin B	Yield in bin C	Yield in bin D
$\Lambda = 100\,\text{TeV}$				

10	3126 ± 50	1254 ± 31	2557 ± 45	6473 ± 73
50	2508 ± 32	7110 ± 56	236.00 ± 9.97	128.06 ± 7.34
100	1575 ± 26	4236 ± 43	283.99 ± 11.02	167.32 ± 8.46
200	691.69 ± 23.34	2124 ± 41	199.13 ± 12.49	100.57 ± 8.87
400	314.84 ± 15.75	1055 ± 28	139.28 ± 10.47	62.84 ± 7.03
600	174.08 ± 11.71	715.294 ± 23.824	94.935 ± 8.645	40.721 ± 5.660
800	142.870 ± 6.063	644.432 ± 12.918	93.255 ± 4.897	26.254 ± 2.597
1000	98.039 ± 8.778	447.894 ± 18.804	38.815 ± 5.521	19.948 ± 3.957
1200	89.072 ± 8.332	388.139 ± 17.427	38.479 ± 5.475	16.011 ± 3.531
10000	98.572 ± 5.378	65.591 ± 4.386	2.756 ± 0.899	8.249 ± 1.555
		$\Lambda = 150$ TeV		
10	315.41 ± 5.18	198.44 ± 4.08	435.28 ± 6.13	696.30 ± 7.87
50	232.39 ± 3.42	894.60 ± 6.96	59.010 ± 1.706	19.626 ± 0.981
100	139.04 ± 2.57	542.38 ± 5.21	74.017 ± 1.874	27.338 ± 1.136
200	72.979 ± 2.457	277.47 ± 4.85	49.989 ± 2.031	20.812 ± 1.308
400	30.421 ± 1.642	131.538 ± 3.435	25.245 ± 1.495	9.192 ± 0.901
600	18.656 ± 1.231	90.040 ± 2.717	20.573 ± 1.293	7.369 ± 0.773
800	11.087 ± 0.954	66.393 ± 2.342	15.867 ± 1.141	3.964 ± 0.570
1000	7.928 ± 0.809	54.130 ± 2.119	13.353 ± 1.050	4.053 ± 0.578
1200	8.193 ± 0.926	48.813 ± 2.266	9.812 ± 1.013	3.069 ± 0.567
10000	2.376 ± 0.299	15.924 ± 0.774	1.362 ± 0.226	0.151 ± 0.075
		$\Lambda = 200$ TeV		
10	59.237 ± 0.997	50.525 ± 0.918	115.288 ± 1.414	143.650 ± 1.591
50	35.812 ± 0.527	197.352 ± 1.302	19.948 ± 0.391	4.151 ± 0.178
100	24.697 ± 0.484	120.381 ± 1.101	22.254 ± 0.459	5.650 ± 0.230
200	13.757 ± 0.471	66.350 ± 1.050	17.909 ± 0.538	4.837 ± 0.278
400	5.374 ± 0.294	32.596 ± 0.729	9.627 ± 0.393	2.365 ± 0.195
600	3.313 ± 0.230	21.156 ± 0.585	6.893 ± 0.333	1.488 ± 0.154
800	2.243 ± 0.192	15.860 ± 0.511	4.927 ± 0.284	1.069 ± 0.132
1000	1.739 ± 0.165	12.665 ± 0.448	4.151 ± 0.256	0.924 ± 0.120
1200	1.611 ± 0.161	10.815 ± 0.418	3.483 ± 0.237	0.650 ± 0.102
10000	0.254 ± 0.042	3.041 ± 0.146	0.398 ± 0.053	0.032 ± 0.015
		$\Lambda = 250$ TeV		
10	14.830 ± 0.263	18.044 ± 0.292	43.653 ± 0.466	39.872 ± 0.443
50	8.682 ± 0.153	63.176 ± 0.440	7.834 ± 0.145	1.275 ± 0.058

100	5.802 ± 0.110	37.914 ± 0.292	9.364 ± 0.140	1.570 ± 0.057
200	3.268 ± 0.086	21.995 ± 0.229	7.729 ± 0.133	1.398 ± 0.056
400	1.425 ± 0.080	9.887 ± 0.214	4.311 ± 0.140	0.890 ± 0.064
600	0.692 ± 0.056	6.525 ± 0.174	2.965 ± 0.117	0.496 ± 0.048
800	0.615 ± 0.053	5.152 ± 0.154	2.270 ± 0.102	0.302 ± 0.037
1000	0.386 ± 0.042	3.894 ± 0.133	1.656 ± 0.087	0.332 ± 0.039
1200	0.370 ± 0.041	3.338 ± 0.124	1.601 ± 0.086	0.218 ± 0.032
10000	0.070 ± 0.011	0.856 ± 0.037	0.191 ± 0.017	0.020 ± 0.006
		Λ = 300 TeV		
10	4.760 ± 0.089	7.779 ± 0.114	19.631 ± 0.188	12.442 ± 0.146
50	2.551 ± 0.044	25.046 ± 0.148	3.572 ± 0.052	0.406 ± 0.017
100	1.733 ± 0.040	15.366 ± 0.126	4.369 ± 0.065	0.477 ± 0.021
200	1.032 ± 0.041	8.478 ± 0.121	3.434 ± 0.076	0.457 ± 0.027
400	0.445 ± 0.028	3.959 ± 0.084	2.003 ± 0.059	0.212 ± 0.019
600	0.266 ± 0.021	2.642 ± 0.067	1.324 ± 0.047	0.169 ± 0.017
800	0.227 ± 0.019	1.869 ± 0.055	0.968 ± 0.040	0.142 ± 0.015
1000	0.170 ± 0.017	1.589 ± 0.051	0.778 ± 0.035	0.095 ± 0.012
1200	0.075 ± 0.011	1.225 ± 0.045	0.698 ± 0.034	0.068 ± 0.010
10000	0.021 ± 0.004	0.311 ± 0.014	0.075 ± 0.007	0.004 ± 0.002
		Λ = 350 TeV		
10	1.636 ± 0.033	3.482 ± 0.049	9.389 ± 0.083	4.460 ± 0.055
50	1.828 ± 0.024	10.124 ± 0.060	1.603 ± 0.022	0.278 ± 0.009
100	1.226 ± 0.020	6.248 ± 0.048	1.821 ± 0.025	0.395 ± 0.012
200	0.689 ± 0.021	3.407 ± 0.048	1.472 ± 0.031	0.318 ± 0.014
400	0.296 ± 0.014	1.616 ± 0.033	0.808 ± 0.023	0.182 ± 0.011
600	0.165 ± 0.010	1.006 ± 0.026	0.585 ± 0.020	0.109 ± 0.008
800	0.129 ± 0.009	0.764 ± 0.022	0.458 ± 0.017	0.107 ± 0.008
1000	0.100 ± 0.008	0.595 ± 0.020	0.345 ± 0.015	0.074 ± 0.007
1200	0.082 ± 0.007	0.481 ± 0.018	0.299 ± 0.014	0.037 ± 0.005
10000	0.018 ± 0.002	0.133 ± 0.006	0.038 ± 0.003	0.006 ± 0.001
		Λ = 400 TeV		
10	0.585 ± 0.013	1.664 ± 0.022	4.522 ± 0.038	1.562 ± 0.022
50	0.656 ± 0.010	4.805 ± 0.028	0.798 ± 0.011	0.118 ± 0.004
100	0.451 ± 0.009	2.894 ± 0.025	0.935 ± 0.014	0.146 ± 0.005
200	0.265 ± 0.008	1.597 ± 0.021	0.704 ± 0.014	0.123 ± 0.006
400	0.102 ± 0.005	0.736 ± 0.015	0.426 ± 0.011	0.071 ± 0.004

600	0.059 ± 0.004	0.477 ± 0.012	0.280 ± 0.009	0.046 ± 0.004
800	0.041 ± 0.003	0.342 ± 0.010	0.219 ± 0.008	0.035 ± 0.003
1000	0.038 ± 0.003	0.280 ± 0.009	0.170 ± 0.007	0.026 ± 0.003
1200	0.031 ± 0.003	0.238 ± 0.008	0.136 ± 0.006	0.016 ± 0.002
10000	0.006 ± 0.001	0.053 ± 0.003	0.018 ± 0.002	0.002 ± 0.001

Table 7.18: Predicted signal yields in bins A, B, C, and D using the event selection for the single-photon category in 2017.

$c\tau$ (cm)	Yield in bin A	Yield in bin B	Yield in bin C	Yield in bin D
		Λ = 100 TeV		
10	1119 ± 31	471.20 ± 20.53	132.13 ± 10.85	389.50 ± 18.66
50	1117 ± 24	1342 ± 26	68.814 ± 6.006	54.588 ± 5.349
100	1056 ± 23	1303 ± 25	122.452 ± 7.858	102.81 ± 7.20
200	659.723 ± 24.442	1055 ± 30	173.79 ± 12.51	110.28 ± 9.96
400	452.414 ± 21.894	751.519 ± 28.264	132.271 ± 11.817	73.754 ± 8.821
600	359.983 ± 18.087	600.157 ± 23.385	98.960 ± 9.470	57.558 ± 7.220
800	293.572 ± 16.264	502.168 ± 21.296	94.244 ± 9.205	55.796 ± 7.081
1000	247.169 ± 14.920	384.657 ± 18.627	71.856 ± 8.037	44.640 ± 6.333
1200	201.205 ± 13.390	338.849 ± 17.389	52.451 ± 6.831	44.271 ± 6.275
10000	98.488 ± 9.616	71.915 ± 8.216	4.114 ± 1.964	10.588 ± 3.151
		Λ = 150 TeV		
10	108.79 ± 3.22	56.139 ± 2.310	20.338 ± 1.388	37.294 ± 1.881
50	90.825 ± 2.136	134.147 ± 2.602	11.053 ± 0.742	11.604 ± 0.760
100	78.258 ± 1.982	135.707 ± 2.618	20.799 ± 1.019	15.171 ± 0.870
200	60.042 ± 2.388	114.544 ± 3.308	28.322 ± 1.638	16.526 ± 1.250
400	35.612 ± 1.992	88.774 ± 3.153	24.121 ± 1.638	15.040 ± 1.293
600	30.139 ± 1.761	62.862 ± 2.548	20.306 ± 1.445	10.575 ± 1.042
800	25.371 ± 1.711	56.032 ± 2.546	16.844 ± 1.393	9.120 ± 1.025
1000	21.120 ± 1.612	45.684 ± 2.374	14.459 ± 1.334	7.306 ± 0.948
1200	14.894 ± 1.186	40.798 ± 1.966	10.881 ± 1.014	5.766 ± 0.738
10000	4.253 ± 0.642	10.670 ± 1.018	1.099 ± 0.326	1.768 ± 0.414
		Λ = 200 TeV		
10	30.296 ± 0.789	15.441 ± 0.561	6.363 ± 0.359	14.289 ± 0.540

50	22.025 ± 0.619	29.968 ± 0.723	3.898 ± 0.259	3.819 ± 0.256
100	17.312 ± 0.404	26.793 ± 0.504	6.088 ± 0.239	5.622 ± 0.229
200	14.186 ± 0.513	25.118 ± 0.684	8.275 ± 0.391	6.284 ± 0.340
400	8.305 ± 0.392	17.990 ± 0.578	7.721 ± 0.378	5.410 ± 0.316
600	7.427 ± 0.370	14.942 ± 0.526	6.112 ± 0.336	3.790 ± 0.264
800	5.804 ± 0.328	10.922 ± 0.450	5.024 ± 0.305	3.292 ± 0.247
1000	4.951 ± 0.314	9.871 ± 0.444	4.852 ± 0.311	2.589 ± 0.227
1200	4.049 ± 0.273	8.655 ± 0.399	4.157 ± 0.276	2.134 ± 0.198
10000	0.930 ± 0.134	2.333 ± 0.212	0.685 ± 0.115	0.420 ± 0.090
		$\Lambda = 250$ TeV		
10	9.658 ± 0.255	6.601 ± 0.210	3.269 ± 0.147	5.325 ± 0.188
50	6.614 ± 0.150	11.535 ± 0.199	2.161 ± 0.085	1.362 ± 0.068
100	5.162 ± 0.112	9.306 ± 0.151	3.057 ± 0.086	1.889 ± 0.068
200	4.466 ± 0.161	9.163 ± 0.231	4.109 ± 0.154	2.231 ± 0.113
400	2.809 ± 0.123	6.764 ± 0.192	3.784 ± 0.143	1.702 ± 0.096
600	2.083 ± 0.104	5.443 ± 0.169	3.137 ± 0.128	1.480 ± 0.088
800	1.686 ± 0.094	4.330 ± 0.151	2.693 ± 0.119	1.251 ± 0.081
1000	1.592 ± 0.091	3.824 ± 0.142	2.390 ± 0.112	1.092 ± 0.075
1200	1.120 ± 0.076	3.302 ± 0.131	2.172 ± 0.107	1.047 ± 0.074
10000	0.315 ± 0.042	0.886 ± 0.071	0.296 ± 0.041	0.168 ± 0.031
		$\Lambda = 300$ TeV		
10	3.469 ± 0.081	3.407 ± 0.080	1.854 ± 0.059	1.842 ± 0.059
50	2.221 ± 0.044	5.113 ± 0.067	1.044 ± 0.030	0.474 ± 0.020
100	1.685 ± 0.038	4.169 ± 0.060	1.576 ± 0.037	0.676 ± 0.024
200	1.431 ± 0.051	3.795 ± 0.084	2.085 ± 0.062	0.843 ± 0.039
400	0.892 ± 0.041	2.920 ± 0.074	1.853 ± 0.059	0.700 ± 0.036
600	0.714 ± 0.036	2.353 ± 0.066	1.588 ± 0.054	0.624 ± 0.034
800	0.628 ± 0.034	1.944 ± 0.060	1.479 ± 0.052	0.463 ± 0.029
1000	0.549 ± 0.032	1.675 ± 0.056	1.224 ± 0.048	0.432 ± 0.028
1200	0.402 ± 0.027	1.636 ± 0.055	1.097 ± 0.045	0.373 ± 0.026
10000	0.116 ± 0.015	0.405 ± 0.029	0.148 ± 0.017	0.062 ± 0.011
		$\Lambda = 350$ TeV		
10	1.279 ± 0.034	1.800 ± 0.040	0.958 ± 0.029	0.711 ± 0.025
50	1.471 ± 0.023	1.749 ± 0.025	0.437 ± 0.012	0.337 ± 0.011
100	1.143 ± 0.021	1.470 ± 0.023	0.571 ± 0.015	0.488 ± 0.013
200	0.940 ± 0.027	1.306 ± 0.031	0.793 ± 0.024	0.582 ± 0.021

400	0.634 ± 0.025	1.022 ± 0.031	0.798 ± 0.028	0.470 ± 0.021
600	0.517 ± 0.020	0.917 ± 0.026	0.659 ± 0.022	0.443 ± 0.018
800	0.458 ± 0.019	0.748 ± 0.025	0.513 ± 0.020	0.333 ± 0.016
1000	0.353 ± 0.016	0.631 ± 0.022	0.493 ± 0.019	0.281 ± 0.014
1200	0.348 ± 0.016	0.546 ± 0.020	0.426 ± 0.018	0.277 ± 0.014
10000	0.087 ± 0.008	0.148 ± 0.011	0.075 ± 0.007	0.044 ± 0.006
Λ = 400 TeV				
10	0.486 ± 0.013	0.881 ± 0.017	0.522 ± 0.013	0.282 ± 0.010
50	0.558 ± 0.010	0.918 ± 0.013	0.231 ± 0.006	0.132 ± 0.005
100	0.453 ± 0.013	0.735 ± 0.016	0.303 ± 0.011	0.204 ± 0.009
200	0.356 ± 0.012	0.657 ± 0.017	0.414 ± 0.013	0.228 ± 0.010
400	0.238 ± 0.009	0.524 ± 0.013	0.409 ± 0.012	0.185 ± 0.008
600	0.204 ± 0.009	0.460 ± 0.013	0.353 ± 0.011	0.156 ± 0.007
800	0.159 ± 0.008	0.338 ± 0.011	0.320 ± 0.011	0.140 ± 0.007
1000	0.130 ± 0.007	0.305 ± 0.010	0.274 ± 0.010	0.125 ± 0.007
1200	0.136 ± 0.007	0.291 ± 0.010	0.216 ± 0.008	0.103 ± 0.006
10000	0.031 ± 0.003	0.083 ± 0.005	0.045 ± 0.004	0.020 ± 0.003

Table 7.19: Predicted signal yields in bins A, B, C, and D using the event selection for the diphoton category in 2017.

$c\tau$ (cm)	Yield in bin A	Yield in bin B	Yield in bin C	Yield in bin D
Λ = 100 TeV				
10	7178 ± 83	1987 ± 42	554.10 ± 22.27	1969 ± 42
50	5373 ± 54	3944 ± 46	103.08 ± 7.35	175.25 ± 9.59
100	3402 ± 42	2671 ± 37	144.19 ± 8.52	243.26 ± 11.08
200	1662 ± 39	1617 ± 38	144.52 ± 11.40	168.35 ± 12.31
400	834.48 ± 29.79	908.10 ± 31.09	109.00 ± 10.72	107.538 ± 10.654
600	571.30 ± 22.81	630.80 ± 23.97	72.966 ± 8.130	77.204 ± 8.363
800	468.60 ± 20.56	572.07 ± 22.73	49.287 ± 6.655	75.059 ± 8.214
1000	372.38 ± 18.32	463.22 ± 20.44	51.012 ± 6.771	46.163 ± 6.441
1200	283.32 ± 15.89	388.63 ± 18.62	44.477 ± 6.290	43.529 ± 6.222
10000	113.48 ± 10.32	78.147 ± 8.564	4.749 ± 2.110	8.626 ± 2.844
Λ = 150 TeV				

10	620.73 ± 7.90	357.07 ± 5.91	105.92 ± 3.18	200.99 ± 4.40
50	427.01 ± 4.71	541.73 ± 5.33	25.266 ± 1.123	24.358 ± 1.102
100	284.09 ± 3.81	356.39 ± 4.29	31.010 ± 1.245	34.213 ± 1.307
200	149.703 ± 3.789	210.63 ± 4.50	29.422 ± 1.669	26.571 ± 1.586
400	72.418 ± 2.846	133.78 ± 3.88	21.587 ± 1.549	17.110 ± 1.379
600	44.875 ± 2.150	87.972 ± 3.018	14.885 ± 1.237	11.970 ± 1.109
800	30.047 ± 1.862	80.015 ± 3.047	13.484 ± 1.246	8.454 ± 0.987
1000	30.149 ± 1.927	61.655 ± 2.761	10.359 ± 1.129	6.271 ± 0.878
1200	27.322 ± 1.608	53.646 ± 2.256	10.773 ± 1.009	5.616 ± 0.728
10000	7.725 ± 0.866	17.705 ± 1.312	1.400 ± 0.369	0.793 ± 0.277
$\Lambda = 200$ TeV				
10	114.92 ± 1.57	77.680 ± 1.279	27.584 ± 0.752	42.735 ± 0.940
50	76.298 ± 1.170	109.864 ± 1.417	8.970 ± 0.393	8.115 ± 0.374
100	50.963 ± 0.700	72.451 ± 0.839	10.241 ± 0.310	10.649 ± 0.316
200	28.623 ± 0.731	45.878 ± 0.930	10.183 ± 0.434	8.667 ± 0.400
400	13.728 ± 0.504	26.131 ± 0.698	6.801 ± 0.354	4.501 ± 0.288
600	7.881 ± 0.382	17.992 ± 0.578	5.208 ± 0.310	3.118 ± 0.240
800	5.868 ± 0.330	14.341 ± 0.517	3.824 ± 0.266	2.076 ± 0.196
1000	4.286 ± 0.292	13.082 ± 0.512	3.194 ± 0.252	1.740 ± 0.186
1200	4.742 ± 0.295	9.934 ± 0.428	2.497 ± 0.214	1.117 ± 0.143
10000	0.731 ± 0.119	3.071 ± 0.244	0.584 ± 0.106	0.377 ± 0.085
$\Lambda = 250$ TeV				
10	36.596 ± 0.508	26.370 ± 0.428	10.777 ± 0.269	15.581 ± 0.325
50	22.890 ± 0.283	37.505 ± 0.368	4.592 ± 0.124	3.191 ± 0.104
100	15.715 ± 0.198	23.754 ± 0.245	5.358 ± 0.114	3.988 ± 0.098
200	8.960 ± 0.229	15.690 ± 0.304	5.030 ± 0.171	3.560 ± 0.143
400	4.347 ± 0.153	7.689 ± 0.205	3.178 ± 0.131	2.013 ± 0.104
600	2.417 ± 0.113	5.312 ± 0.167	2.282 ± 0.109	1.230 ± 0.080
800	1.839 ± 0.098	4.276 ± 0.150	1.842 ± 0.098	1.013 ± 0.073
1000	1.424 ± 0.086	3.250 ± 0.130	1.426 ± 0.086	0.753 ± 0.063
1200	1.022 ± 0.073	2.974 ± 0.125	1.183 ± 0.079	0.658 ± 0.059
10000	0.177 ± 0.032	0.636 ± 0.060	0.249 ± 0.038	0.081 ± 0.021
$\Lambda = 300$ TeV				
10	13.716 ± 0.165	13.289 ± 0.162	5.744 ± 0.104	6.069 ± 0.107
50	8.151 ± 0.085	18.023 ± 0.130	2.629 ± 0.048	1.201 ± 0.032

100	5.643 ± 0.070	11.056 ± 0.100	3.118 ± 0.052	1.713 ± 0.038
200	3.287 ± 0.078	6.738 ± 0.113	2.726 ± 0.071	1.448 ± 0.052
400	1.552 ± 0.054	3.526 ± 0.081	1.732 ± 0.057	0.885 ± 0.040
600	0.952 ± 0.042	2.314 ± 0.066	1.147 ± 0.046	0.595 ± 0.033
800	0.600 ± 0.033	1.761 ± 0.057	0.971 ± 0.042	0.431 ± 0.028
1000	0.491 ± 0.030	1.412 ± 0.051	0.847 ± 0.040	0.339 ± 0.025
1200	0.383 ± 0.027	1.253 ± 0.048	0.600 ± 0.033	0.246 ± 0.021
10000	0.068 ± 0.012	0.229 ± 0.021	0.135 ± 0.017	0.028 ± 0.007
$\Lambda = 350\,\text{TeV}$				
10	5.143 ± 0.069	7.106 ± 0.082	3.151 ± 0.053	2.400 ± 0.046
50	6.016 ± 0.048	6.509 ± 0.050	1.053 ± 0.019	0.986 ± 0.019
100	4.039 ± 0.040	3.999 ± 0.039	1.251 ± 0.022	1.234 ± 0.021
200	2.355 ± 0.042	2.381 ± 0.043	1.090 ± 0.029	1.102 ± 0.029
400	1.034 ± 0.032	1.181 ± 0.034	0.667 ± 0.025	0.662 ± 0.025
600	0.663 ± 0.022	0.858 ± 0.025	0.501 ± 0.019	0.422 ± 0.018
800	0.484 ± 0.020	0.679 ± 0.023	0.374 ± 0.017	0.351 ± 0.017
1000	0.350 ± 0.016	0.491 ± 0.019	0.322 ± 0.015	0.282 ± 0.014
1200	0.304 ± 0.015	0.441 ± 0.018	0.281 ± 0.014	0.202 ± 0.012
10000	0.040 ± 0.005	0.099 ± 0.009	0.050 ± 0.006	0.029 ± 0.005
$\Lambda = 400\,\text{TeV}$				
10	2.082 ± 0.027	3.747 ± 0.037	1.795 ± 0.025	0.952 ± 0.018
50	2.534 ± 0.022	3.681 ± 0.027	0.636 ± 0.011	0.424 ± 0.009
100	1.733 ± 0.026	2.284 ± 0.030	0.768 ± 0.017	0.593 ± 0.015
200	1.001 ± 0.021	1.257 ± 0.023	0.611 ± 0.016	0.498 ± 0.015
400	0.436 ± 0.012	0.629 ± 0.014	0.405 ± 0.011	0.271 ± 0.009
600	0.271 ± 0.010	0.461 ± 0.013	0.296 ± 0.010	0.181 ± 0.008
800	0.182 ± 0.008	0.314 ± 0.011	0.220 ± 0.009	0.141 ± 0.007
1000	0.137 ± 0.007	0.278 ± 0.010	0.188 ± 0.008	0.112 ± 0.006
1200	0.129 ± 0.006	0.230 ± 0.009	0.149 ± 0.007	0.089 ± 0.005
10000	0.021 ± 0.003	0.050 ± 0.004	0.022 ± 0.003	0.010 ± 0.002

Table 7.20: Predicted signal yields in bins A, B, C, and D using the event selection for the single-photon category in 2018.

$c\tau$ (cm)	Yield in bin A	Yield in bin B	Yield in bin C	Yield in bin D
		Λ = 100 TeV		
10	834.99 ± 23.09	1779 ± 33	148.69 ± 9.72	66.180 ± 6.483
50	1262 ± 29	2560 ± 42	65.499 ± 6.679	28.325 ± 4.391
100	1050 ± 38	2595 ± 60	143.573 ± 14.198	96.214 ± 11.621
200	770.93 ± 31.53	1984 ± 50	209.47 ± 16.40	103.31 ± 11.51
400	478.53 ± 17.51	1433 ± 30	174.71 ± 10.57	66.605 ± 6.526
600	358.13 ± 15.22	1091 ± 26	135.65 ± 9.36	57.355 ± 6.086
800	303.86 ± 13.96	819.10 ± 22.97	109.65 ± 8.38	37.864 ± 4.925
1000	255.14 ± 12.93	755.64 ± 22.31	83.971 ± 7.418	44.810 ± 5.418
1200	210.40 ± 16.346	722.89 ± 30.36	89.925 ± 10.681	43.098 ± 7.393
10000	89.391 ± 10.938	170.79 ± 15.12	15.506 ± 4.554	16.705 ± 4.727
		Λ = 150 TeV		
10	79.549 ± 3.284	210.32 ± 5.36	25.793 ± 1.866	9.157 ± 1.111
50	108.53 ± 3.15	247.32 ± 4.78	14.761 ± 1.158	9.558 ± 0.932
100	90.606 ± 2.541	233.37 ± 4.09	29.533 ± 1.447	17.060 ± 1.100
200	73.560 ± 2.231	209.71 ± 3.78	31.407 ± 1.456	15.294 ± 1.015
400	41.092 ± 2.371	152.44 ± 4.58	34.144 ± 2.161	10.844 ± 1.217
600	33.295 ± 2.108	112.63 ± 3.88	26.282 ± 1.872	9.643 ± 1.133
800	25.569 ± 1.315	90.217 ± 2.475	23.207 ± 1.252	9.810 ± 0.814
1000	21.093 ± 1.193	75.857 ± 2.268	16.350 ± 1.051	8.291 ± 0.748
1200	17.460 ± 1.534	68.114 ± 3.035	16.495 ± 1.491	5.963 ± 0.896
10000	2.862 ± 0.449	17.215 ± 1.101	2.024 ± 0.377	1.524 ± 0.327
		Λ = 200 TeV		
10	21.895 ± 0.752	56.307 ± 1.214	9.650 ± 0.498	4.365 ± 0.335
50	22.748 ± 0.527	56.305 ± 0.835	5.731 ± 0.264	3.274 ± 0.199
100	18.906 ± 0.771	51.227 ± 1.277	9.958 ± 0.558	5.269 ± 0.406
200	15.083 ± 0.633	45.657 ± 1.107	14.523 ± 0.621	5.758 ± 0.390
400	10.816 ± 0.377	33.654 ± 0.669	12.274 ± 0.402	4.614 ± 0.246
600	7.336 ± 0.439	25.849 ± 0.827	9.332 ± 0.496	4.268 ± 0.335
800	5.146 ± 0.260	22.306 ± 0.544	8.572 ± 0.336	3.446 ± 0.213
1000	4.485 ± 0.347	18.386 ± 0.705	6.864 ± 0.430	2.535 ± 0.261
1200	3.965 ± 0.253	15.367 ± 0.500	7.103 ± 0.339	2.219 ± 0.189

10000	0.625 ± 0.132	3.959 ± 0.333	1.118 ± 0.177	0.341 ± 0.098
		Λ = 250 TeV		
10	6.532 ± 0.162	24.035 ± 0.314	5.028 ± 0.142	1.550 ± 0.079
50	6.354 ± 0.148	21.815 ± 0.278	3.291 ± 0.106	0.922 ± 0.056
100	5.646 ± 0.136	18.282 ± 0.247	4.774 ± 0.125	1.591 ± 0.072
200	4.719 ± 0.133	16.857 ± 0.253	6.515 ± 0.156	2.246 ± 0.092
400	2.791 ± 0.102	12.118 ± 0.214	6.085 ± 0.151	1.641 ± 0.078
600	2.114 ± 0.089	9.359 ± 0.188	5.276 ± 0.141	1.332 ± 0.070
800	1.747 ± 0.114	8.384 ± 0.251	4.509 ± 0.184	1.109 ± 0.091
1000	1.642 ± 0.078	6.731 ± 0.159	3.689 ± 0.117	0.970 ± 0.060
1200	1.143 ± 0.065	5.899 ± 0.148	3.160 ± 0.108	0.830 ± 0.056
10000	0.170 ± 0.025	1.352 ± 0.071	0.415 ± 0.039	0.142 ± 0.023
		Λ = 300 TeV		
10	2.049 ± 0.074	10.374 ± 0.168	2.276 ± 0.078	0.484 ± 0.036
50	2.107 ± 0.050	9.354 ± 0.106	1.585 ± 0.043	0.378 ± 0.021
100	1.804 ± 0.054	7.556 ± 0.111	2.326 ± 0.061	0.623 ± 0.032
200	1.412 ± 0.044	6.900 ± 0.097	3.283 ± 0.067	0.741 ± 0.031
400	0.921 ± 0.035	5.210 ± 0.083	3.044 ± 0.064	0.552 ± 0.027
600	0.677 ± 0.030	4.134 ± 0.075	2.584 ± 0.059	0.480 ± 0.025
800	0.541 ± 0.027	3.518 ± 0.068	2.410 ± 0.056	0.430 ± 0.024
1000	0.540 ± 0.038	2.979 ± 0.089	1.867 ± 0.070	0.528 ± 0.037
1200	0.387 ± 0.023	2.661 ± 0.061	1.636 ± 0.048	0.292 ± 0.020
10000	0.096 ± 0.016	0.687 ± 0.043	0.286 ± 0.027	0.038 ± 0.010
		Λ = 350 TeV		
10	1.259 ± 0.026	4.191 ± 0.048	1.043 ± 0.024	0.330 ± 0.013
50	1.249 ± 0.025	3.870 ± 0.044	0.715 ± 0.019	0.227 ± 0.010
100	0.988 ± 0.022	3.008 ± 0.038	1.040 ± 0.022	0.343 ± 0.013
200	0.796 ± 0.021	2.809 ± 0.039	1.357 ± 0.027	0.410 ± 0.015
400	0.551 ± 0.017	2.113 ± 0.034	1.270 ± 0.026	0.360 ± 0.014
600	0.439 ± 0.015	1.773 ± 0.031	1.142 ± 0.025	0.308 ± 0.013
800	0.348 ± 0.014	1.435 ± 0.028	0.972 ± 0.023	0.243 ± 0.011
1000	0.314 ± 0.013	1.244 ± 0.026	0.829 ± 0.021	0.231 ± 0.011
1200	0.257 ± 0.012	1.088 ± 0.024	0.857 ± 0.021	0.196 ± 0.010
10000	0.060 ± 0.008	0.293 ± 0.018	0.100 ± 0.010	0.028 ± 0.005
		Λ = 400 TeV		
10	0.420 ± 0.010	2.025 ± 0.022	0.546 ± 0.011	0.107 ± 0.005

50	0.460 ± 0.013	1.824 ± 0.026	0.381 ± 0.012	0.094 ± 0.006
100	0.379 ± 0.009	1.485 ± 0.018	0.510 ± 0.011	0.137 ± 0.005
200	0.292 ± 0.012	1.325 ± 0.025	0.658 ± 0.017	0.184 ± 0.009
400	0.157 ± 0.019	1.000 ± 0.047	0.617 ± 0.037	0.140 ± 0.018
600	0.148 ± 0.006	0.820 ± 0.014	0.607 ± 0.012	0.142 ± 0.006
800	0.132 ± 0.008	0.725 ± 0.018	0.528 ± 0.016	0.093 ± 0.007
1000	0.115 ± 0.007	0.614 ± 0.017	0.451 ± 0.014	0.088 ± 0.006
1200	0.097 ± 0.005	0.545 ± 0.011	0.388 ± 0.009	0.070 ± 0.004
10000	0.030 ± 0.003	0.128 ± 0.005	0.057 ± 0.004	0.011 ± 0.002

Table 7.21: Predicted signal yields in bins A, B, C, and D using the event selection for the diphoton category in 2018.

$c\tau$ (cm)	Yield in bin A	Yield in bin B	Yield in bin C	Yield in bin D
		Λ = 100 TeV		
10	6017 ± 63	9738 ± 81	533.31 ± 18.43	407.63 ± 16.11
50	5806 ± 64	7913 ± 75	89.355 ± 7.801	107.51 ± 8.55
100	3753 ± 73	5367 ± 88	179.31 ± 15.86	177.61 ± 15.79
200	1866 ± 49	3036 ± 63	170.36 ± 14.78	138.54 ± 13.33
400	860.93 ± 23.53	1754 ± 33	139.24 ± 9.43	82.779 ± 7.275
600	563.01 ± 19.10	1362 ± 29	81.875 ± 7.272	63.470 ± 6.402
800	435.60 ± 16.73	975.57 ± 25.09	88.341 ± 7.525	44.351 ± 5.331
1000	345.11 ± 15.05	900.77 ± 24.37	78.377 ± 7.166	50.011 ± 5.724
1200	330.97 ± 20.51	650.30 ± 28.78	49.815 ± 7.949	24.696 ± 5.596
10000	63.384 ± 9.209	135.97 ± 13.49	5.586 ± 2.734	4.199 ± 2.370
		Λ = 150 TeV		
10	453.56 ± 7.94	1234 ± 13	97.586 ± 3.639	40.142 ± 2.329
50	447.38 ± 6.47	990.04 ± 9.82	32.041 ± 1.708	17.369 ± 1.257
100	300.20 ± 4.66	669.49 ± 7.05	47.053 ± 1.828	25.718 ± 1.350
200	154.65 ± 3.24	402.05 ± 5.28	54.074 ± 1.912	25.175 ± 1.303
400	71.950 ± 3.141	221.85 ± 5.54	32.022 ± 2.092	14.462 ± 1.405
600	42.904 ± 2.393	155.11 ± 4.57	24.525 ± 1.808	9.463 ± 1.123
800	35.211 ± 1.543	126.83 ± 2.93	21.206 ± 1.197	7.817 ± 0.726
1000	28.936 ± 1.398	107.186 ± 2.699	15.445 ± 1.021	6.320 ± 0.653

1200	22.943 ± 1.759	86.923 ± 3.431	14.338 ± 1.390	4.525 ± 0.780
10000	4.539 ± 0.565	20.760 ± 1.209	1.866 ± 0.362	0.864 ± 0.246
		$\Lambda = 200\,\text{TeV}$		
10	83.145 ± 1.482	257.401 ± 2.691	28.410 ± 0.858	8.911 ± 0.479
50	75.188 ± 0.968	202.014 ± 1.627	10.491 ± 0.357	4.958 ± 0.245
100	52.322 ± 1.291	129.922 ± 2.066	15.070 ± 0.688	8.202 ± 0.507
200	30.510 ± 0.903	82.916 ± 1.503	14.681 ± 0.624	6.937 ± 0.428
400	14.689 ± 0.440	45.184 ± 0.777	10.797 ± 0.377	4.530 ± 0.244
600	8.317 ± 0.468	31.601 ± 0.916	8.064 ± 0.461	3.147 ± 0.287
800	5.592 ± 0.272	24.254 ± 0.568	5.911 ± 0.279	1.880 ± 0.157
1000	4.552 ± 0.350	18.727 ± 0.712	4.721 ± 0.356	1.624 ± 0.209
1200	3.555 ± 0.240	17.003 ± 0.526	4.037 ± 0.256	1.114 ± 0.134
10000	0.585 ± 0.128	3.131 ± 0.296	0.557 ± 0.125	0.132 ± 0.061
		$\Lambda = 250\,\text{TeV}$		
10	22.950 ± 0.307	85.877 ± 0.618	12.800 ± 0.228	3.287 ± 0.115
50	21.854 ± 0.278	67.676 ± 0.505	5.581 ± 0.139	2.032 ± 0.083
100	15.116 ± 0.224	43.739 ± 0.390	7.324 ± 0.155	3.086 ± 0.101
200	9.311 ± 0.187	27.468 ± 0.326	7.544 ± 0.168	3.017 ± 0.106
400	3.898 ± 0.121	13.390 ± 0.225	4.565 ± 0.131	1.876 ± 0.084
600	2.397 ± 0.095	8.997 ± 0.184	3.442 ± 0.113	1.018 ± 0.062
800	1.606 ± 0.109	7.435 ± 0.236	2.748 ± 0.143	0.859 ± 0.080
1000	1.293 ± 0.069	5.532 ± 0.144	2.018 ± 0.087	0.640 ± 0.049
1200	1.190 ± 0.066	4.888 ± 0.135	1.692 ± 0.079	0.545 ± 0.045
10000	0.108 ± 0.020	0.977 ± 0.060	0.261 ± 0.031	0.027 ± 0.010
		$\Lambda = 300\,\text{TeV}$		
10	7.810 ± 0.145	38.254 ± 0.339	6.139 ± 0.128	1.192 ± 0.056
50	7.465 ± 0.095	30.202 ± 0.199	3.258 ± 0.062	0.819 ± 0.031
100	5.274 ± 0.093	19.244 ± 0.182	4.554 ± 0.086	1.368 ± 0.047
200	3.454 ± 0.068	12.184 ± 0.130	4.171 ± 0.075	1.272 ± 0.041
400	1.527 ± 0.045	5.858 ± 0.089	2.637 ± 0.059	0.667 ± 0.030
600	0.947 ± 0.036	3.798 ± 0.072	1.934 ± 0.051	0.531 ± 0.027
800	0.673 ± 0.030	3.189 ± 0.065	1.522 ± 0.045	0.233 ± 0.017
1000	0.537 ± 0.038	2.406 ± 0.080	1.269 ± 0.058	0.262 ± 0.026
1200	0.385 ± 0.023	2.209 ± 0.056	0.920 ± 0.036	0.235 ± 0.018
10000	0.064 ± 0.013	0.415 ± 0.033	0.123 ± 0.018	0.018 ± 0.007

		Λ = 300 TeV		
10	4.571 ± 0.050	16.435 ± 0.101	2.932 ± 0.040	0.877 ± 0.022
50	4.727 ± 0.049	13.040 ± 0.084	1.617 ± 0.028	0.548 ± 0.016
100	3.285 ± 0.040	8.597 ± 0.067	2.129 ± 0.032	0.852 ± 0.020
200	1.962 ± 0.032	4.933 ± 0.052	1.912 ± 0.032	0.852 ± 0.021
400	0.855 ± 0.021	2.402 ± 0.036	1.253 ± 0.026	0.462 ± 0.016
600	0.556 ± 0.017	1.626 ± 0.030	0.910 ± 0.022	0.318 ± 0.013
800	0.400 ± 0.015	1.220 ± 0.026	0.687 ± 0.019	0.253 ± 0.012
1000	0.286 ± 0.012	0.941 ± 0.022	0.536 ± 0.017	0.205 ± 0.010
1200	0.221 ± 0.011	0.894 ± 0.022	0.492 ± 0.016	0.154 ± 0.009
10000	0.031 ± 0.006	0.152 ± 0.013	0.060 ± 0.008	0.014 ± 0.004
		Λ = 400 TeV		
10	1.824 ± 0.021	8.127 ± 0.047	1.578 ± 0.019	0.343 ± 0.009
50	1.936 ± 0.027	7.103 ± 0.054	0.962 ± 0.019	0.245 ± 0.009
100	1.362 ± 0.017	4.454 ± 0.033	1.291 ± 0.017	0.376 ± 0.009
200	0.780 ± 0.019	2.515 ± 0.035	1.043 ± 0.022	0.338 ± 0.013
400	0.370 ± 0.029	1.190 ± 0.052	0.714 ± 0.040	0.217 ± 0.022
600	0.230 ± 0.007	0.887 ± 0.014	0.505 ± 0.011	0.135 ± 0.006
800	0.156 ± 0.008	0.623 ± 0.017	0.382 ± 0.013	0.103 ± 0.007
1000	0.125 ± 0.008	0.527 ± 0.016	0.312 ± 0.012	0.068 ± 0.006
1200	0.093 ± 0.005	0.413 ± 0.010	0.264 ± 0.008	0.066 ± 0.004
10000	0.013 ± 0.002	0.083 ± 0.004	0.036 ± 0.003	0.007 ± 0.001

Table 7.22: Predicted background yields in bins B, C, and D from observed data in bin A using the shape templates described in Sec. 7.6 for 2016.

Bin boundary	Yield in bin A	Yield in bin B	Yield in bin C	Yield in bin D
0.0 ns, 250 GeV	26020 ± 161	72.6 ± 5.8	85.380 ± 6.905	30620.3 ± 324.2
1.5 ns, 100 GeV	54626 ± 234	2058.1 ± 47.3	0.603 ± 0.151	16.0 ± 4.0
1.5 ns, 150 GeV	55943 ± 237	745.7 ± 27.8	0.218 ± 0.055	16.4 ± 4.1

7.9 Summary

We present a search for long-lived particles that decay to a photon and a weakly interacting particle, where the expected exclusion limits are set on the long-lived

Table 7.23: Predicted background yields in bins B, C, and D from observed data in bin A using the shape templates described in Sec. 7.6 for 2017.

Bin boundary	Yield in bin A	Yield in bin B	Yield in bin C	Yield in bin D
Single-photon category				
0.5 ns, 300 GeV	373442 ± 611	233.4 ± 14.5	28.005 ± 1.747	44816.3 ± 240.6
1.5 ns, 200 GeV	417372 ± 646	1019.9 ± 32.2	0.505 ± 0.039	206.9 ± 14.7
1.5 ns, 300 GeV	418132 ± 647	261.3 ± 16.2	0.129 ± 0.012	207.2 ± 14.7
Diphoton category				
0.5 ns, 300 GeV	33575 ± 183	72.3 ± 8.1	8.413 ± 0.954	3905.8 ± 71.5
1.5 ns, 200 GeV	37284 ± 193	237.9 ± 15.6	0.163 ± 0.035	25.5 ± 5.2
1.5 ns, 300 GeV	37440 ± 193	80.6 ± 9.0	0.055 ± 0.013	25.6 ± 5.2

Table 7.24: Predicted background yields in bins B, C, and D from observed data in bin A using the shape templates described in Sec. 7.6 for 2018.

Bin boundary	Yield in bin A	Yield in bin B	Yield in bin C	Yield in bin D
Single-photon category (eras ABC)				
0.5 ns, 300 GeV	263589 ± 513	157.3 ± 12.5	1.760 ± 0.144	2950.0 ± 56.2
1.5 ns, 200 GeV	265975 ± 516	669.1 ± 25.9	0.139 ± 0.020	55.4 ± 7.6
1.5 ns, 300 GeV	266485 ± 516	159.0 ± 12.6	0.033 ± 0.005	55.5 ± 7.6
Single-photon category (era D)				
0.5 ns, 300 GeV	285132 ± 534	222.2 ± 14.9	0.866 ± 0.064	1110.7 ± 34.2
1.5 ns, 200 GeV	285495 ± 534	923.1 ± 30.5	0.172 ± 0.025	53.2 ± 7.5
1.5 ns, 300 GeV	286195 ± 535	223.0 ± 14.9	0.042 ± 0.006	53.3 ± 7.5
Diphoton category (eras ABC)				
0.5 ns, 300 GeV	24167 ± 155	44.5 ± 6.6	0.497 ± 0.081	269.7 ± 17.2
1.5 ns, 200 GeV	24312 ± 156	166.0 ± 13.0	0.029 ± 0.015	4.3 ± 2.1
1.5 ns, 300 GeV	24433 ± 156	45.0 ± 6.7	0.008 ± 0.004	4.3 ± 2.1
Diphoton category (era D)				
0.5 ns, 300 GeV	27163 ± 165	58.8 ± 7.7	0.220 ± 0.037	101.8 ± 10.5
1.5 ns, 200 GeV	27089 ± 165	234.0 ± 15.4	0.019 ± 0.013	2.2 ± 1.5
1.5 ns, 300 GeV	27264 ± 165	59.0 ± 7.7	0.005 ± 0.003	2.2 ± 1.5

particles as functions of their masses and proper decay lengths. The search is based on proton-proton collisions at a center-of-mass energy of 13 TeV collected by the CMS experiment during Run 2 over the span of three years, from 2016 to 2018, corresponding to a total integrated luminosity of 136.3 fb^{-1}.

This search relies on the unique kinematic features of the photons decayed from the long-lived particles, where the photons interacts with ECAL from a non-normal impact angles and with delayed times, resulting in different signatures in the ECAL clusters compared to those of prompt photons. These signatures are exploited with

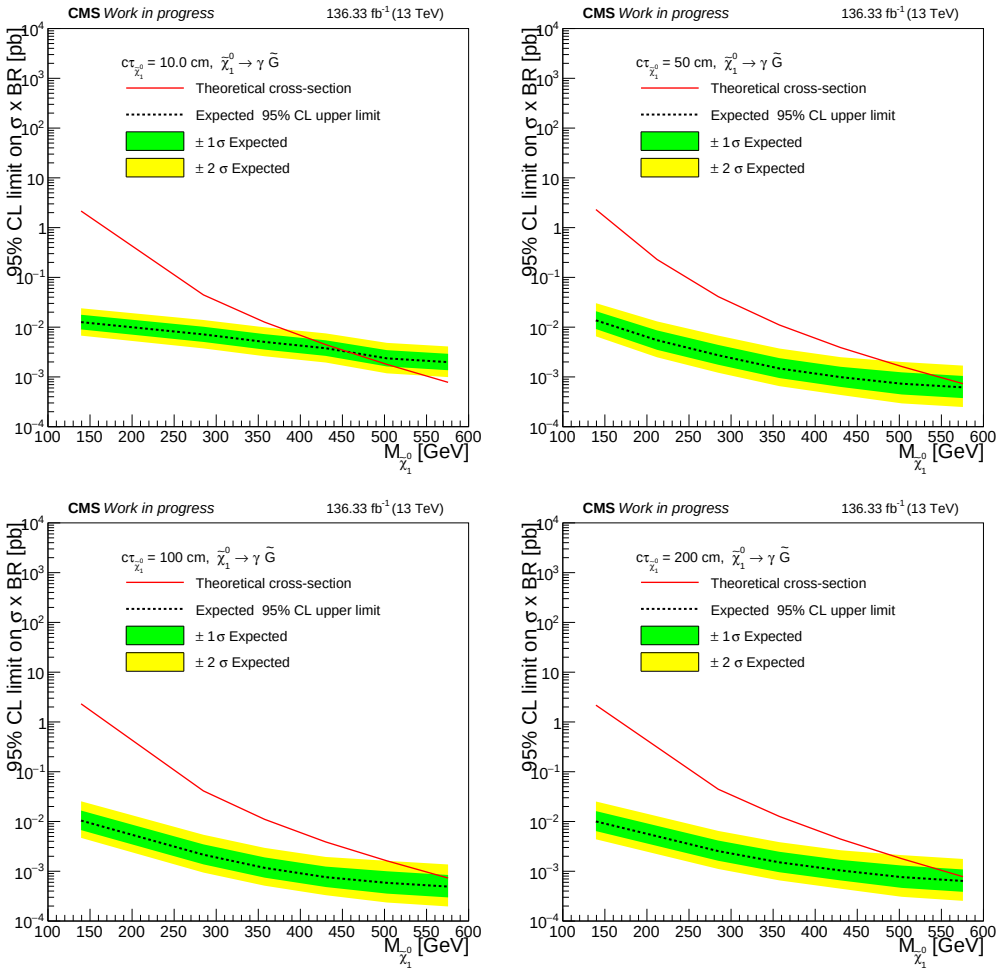


Figure 7.20: The expected 95% CL upper limit on the product of GMSB SPS8 neutralino production cross section and its branching ratio as a function of the neutralino mass for $c\tau$ between 10 cm and 200 cm, obtained from the full Run 2 data.

a dedicated deep-neural-network (DNN) based identifier to separate signal photons from the background. This DNN offers a significant improvement over the cut-based identification technique used in previous searches in CMS.

The results are interpreted in the context of supersymmetry with gauge-mediated supersymmetry breaking, using the SPS8 benchmark model. Expected exclusion limits of the neutralinos are set at 95% confidence level in terms of the neutralino masses, which is linearly proportional to the supersymmetry breaking scale, and the neutralino proper decay lengths. The previous best limits are extended by approximately 5 times in the neutralino proper decay length and by over 100 GeV in the neutralino mass.

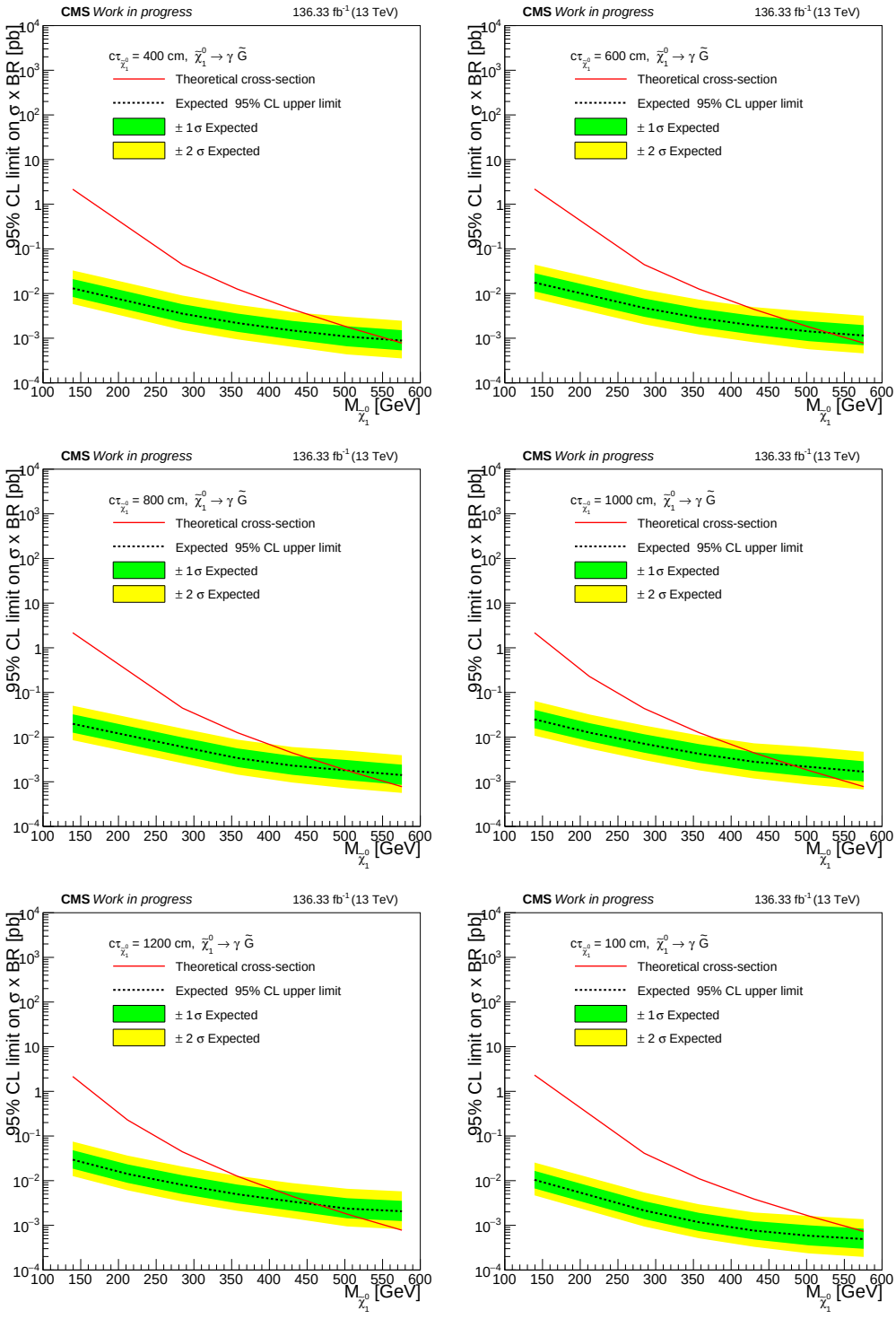


Figure 7.21: The expected 95% CL exclusion limit on the product of GMSB SPS8 neutralino production cross section and its branching ratio as a function of the neutralino mass for $c\tau$ above 200 cm, obtained from the full Run 2 data.

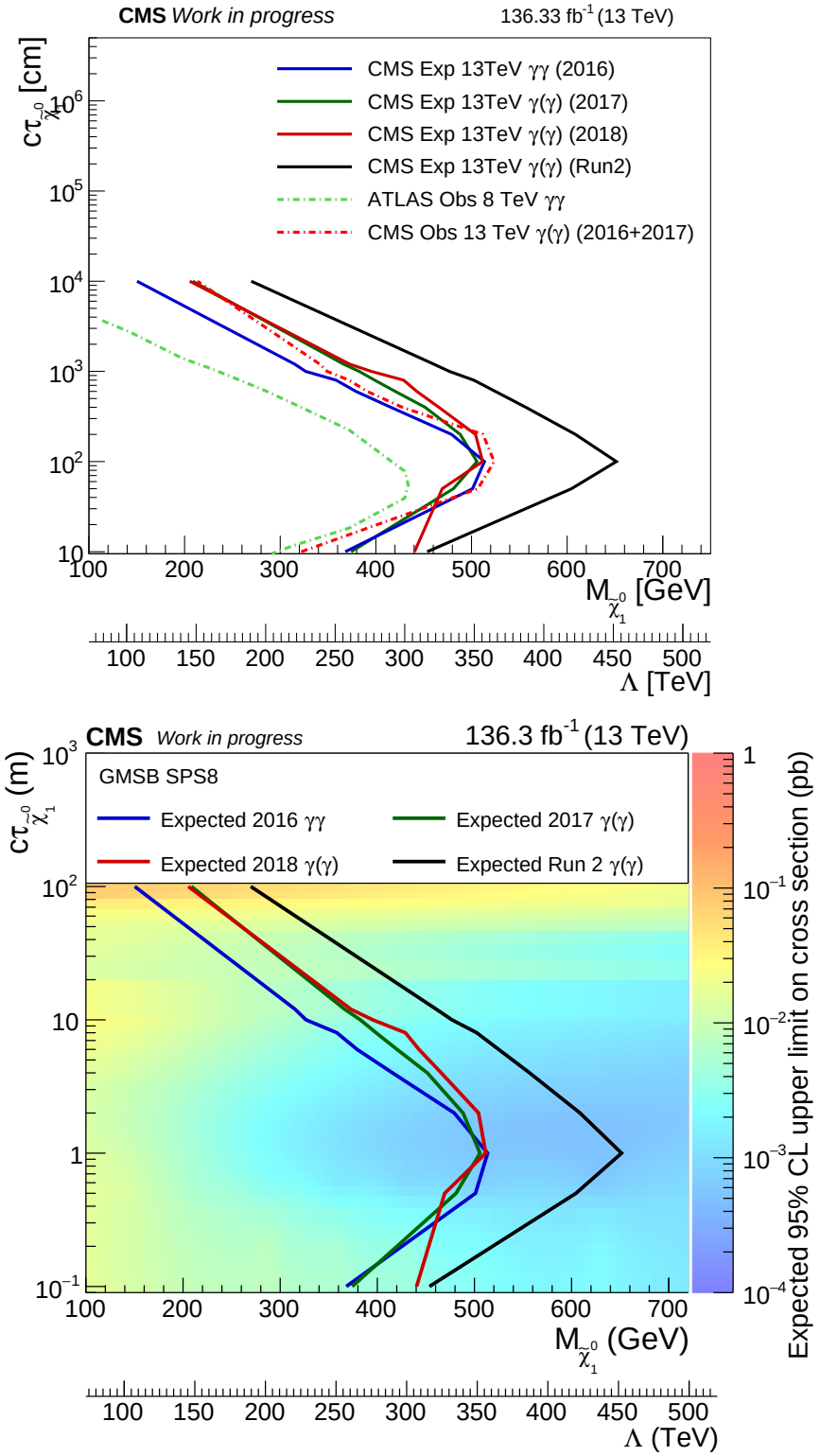


Figure 7.22: The expected 95% CL exclusion contours for the GMSB SPS8 model as functions of the SUSY breaking scale Λ and the neutralino's lifetime $c\tau$ from different years along with the combined Run 2 result. Top: The exclusion boundaries from this search compared to previously published analyses from ATLAS and CMS. Bottom: The exclusion boundaries along with a color map of the expected 95% CL upper limit on the signal cross section obtained from full Run 2 data.

Chapter 8

TOWARD MODEL-INDEPENDENT NEW PHYSICS SEARCHES WITH UNSUPERVISED LEARNING

Using variational autoencoders trained on known physics processes, we develop a one-sided threshold test to isolate previously unseen processes as outlier events. Since the autoencoder training does not depend on any specific new physics signature, the proposed procedure does not make specific assumptions on the nature of new physics. An event selection based on this algorithm would be complementary to classic LHC searches, typically based on model-dependent hypothesis testing. Such an algorithm would deliver a list of anomalous events, that the experimental collaborations could further scrutinize and even release as a catalog, similarly to what is typically done in other scientific domains. Event topologies repeating in this dataset could inspire new-physics model building and new experimental searches. Running in the trigger system of the LHC experiments, such an application could identify anomalous events that would be otherwise lost, extending the scientific reach of the LHC.

8.1 Introduction

One of the main motivations behind the construction of the CERN Large Hadron Collider (LHC) is the exploration of the high-energy frontier in search for *new physics* phenomena. New physics could answer some of the standing fundamental questions in particle physics, e.g., the nature of dark matter or the origin of electroweak symmetry breaking. In LHC experiments, searches for physics beyond the Standard Model (BSM) are typically carried on as fully-supervised data analyses: assuming a new physics scenario of some kind, a search is structured as a hypothesis test, based on a profiled-likelihood ratio [140]. These searches are said to be *model dependent*, since they depend on considering a specific new physics model.

Assuming that one is testing the *right* model, this approach is very effective in discovering a signal, as demonstrated by the discovery of the Standard Model (SM) Higgs boson [242, 243] at the LHC. On the other hand, given the (so far) negative outcome of many BSM searches at particle-physics experiments, it is possible that a future BSM model, if any, is not among those typically tested. The problem is more profound if analyzed in the context of the LHC big-data problem: at the

LHC, 40 million proton-beam collisions are produced every second, but only ∼1000 collision events/sec can be stored by the ATLAS and CMS experiments, due to limited bandwidth, processing, and storage resources. It is possible to imagine BSM scenarios that would escape detection, simply because the corresponding new physics events would be rejected by a typical set of online selection algorithms.

Establishing alternative search methodologies with reduced model dependence is an important aspect of future LHC runs. Traditionally, this issue was addressed with so-called model-independent searches, performed at the Tevatron [244, 245], at HERA [246], and at the LHC [247, 248], as discussed in Section 8.2.

In this paper, we propose to address this need by deploying an unsupervised algorithm in the online selection system (trigger) of the LHC experiments.[1] This algorithm would be trained on known SM processes and could be able to identify BSM events as anomalies. The selected events could be stored in a special stream, scrutinized by experts (e.g., to exclude the occurrence of detector malfunctions that could explain the anomalies), and even released outside the experimental collaborations, in the form of an open-access catalog. The final goal of this application is to identify anomalous event topologies and inspire future supervised searches on data collected afterwards.

As an example, we consider the case of a typical single-lepton data stream, selected by a hardware-based Level-1 (L1) trigger system. In normal conditions, the L1 trigger is the first of a two-steps selection stage. After a coarse (and often local) reconstruction and loose selection at L1, events are fully reconstructed in the High Lever Trigger (HLT), where a much tighter selection is applied. The selection is usually done having in mind specific signal topologies, eg., specific BSM models. In this study, we imagine to replace this model-dependent selection with a variational autoencoder (VAE) [251, 252] looking for anomalous events in the incoming single-lepton stream. The VAE is trained to compress the input event representation into a lower-dimension latent space and then decompress it, returning the shape parameters describing the probability density function (pdf) of each input quantity given a point in the compressed space. In addition, a VAE allows a stochastic modeling of the latent space, a feature which is missing in a simple AE architecture. The highlighted procedure is not specific of the considered single-lepton stream and could be easily extended to other data streams.

[1] A description of the ATLAS and CMS trigger systems can be found in Ref. [249] and Ref. [250], respectively. In this study, we take the data-taking strategy of these two experiments as a reference. On the other hand, the proposed strategy could be adapted to other use cases.

The distribution of the VAE's reconstruction loss on a validation sample is used to define a threshold, corresponding to a desired acceptance rate for SM events. All the events with loss larger than the threshold are considered as potential anomalies and could be stored in a low-rate anomalous-event data stream. In this work, we set the threshold such that $\sim$ 1000 SM events would be collected every month under typical LHC operation conditions. In particular, we took as a reference 8 months of data taking per year, with an integrated luminosity of ~ 40 fb^{-1}. Assuming an LHC duty cycle of 2/3, this corresponds to an average instantaneous luminosity of $\sim 2.9 \times 10^{33}$ cm^{-2} s^{-1}.

We then evaluate the BSM production cross section that would correspond to a signal excess of 100 BSM events selected per month, as well as the one that would give a signal yield $\sim 1/3$ of the SM yield. For this, we consider a set of low-mass BSM resonances, decaying to one or more leptons and light enough to be challenging for the currently employed LHC trigger algorithms.

This paper is structured as follows: we discuss related works in Section 8.2. Section 8.3 gives a brief description of the dataset used. Section 8.4 describes the VAE model used in the study, as well as a set of fully-supervised classifiers used for performance comparison. Results are discussed in Section 8.5. In Section 8.6 we discuss how such a procedure could be deployed in a typical LHC experiment while relying exclusively on data. Conclusions are given in Section 8.9.

8.2 Related work

Model-independent searches for new physics have been performed at the Tevatron [244, 245], HERA [246], and the LHC [247, 248]. These searches are based on the comparison of a large set of binned distributions to the prediction from Monte Carlo (MC) simulations, in search for bins exhibiting a deviation larger than some predefined threshold. While the effectiveness of this strategy in establishing a discovery has been a matter of discussion, a recent study by the ATLAS collaboration [248] rephrased this model-independent search strategy into a tool to identify interesting excesses, on which traditional analysis techniques could be performed on independent datasets (e.g., the data collected after running the model-independent analysis). This change of scope has the advantage of reducing the trial factor (i.e., the so-called *look-elsewhere* effect [253, 254]), which would otherwise wash out the significance of an observed excess.

Our strategy is similar to what is proposed in Ref. [248], with two substantial differences: (i) we aim to process also those events that could be discarded by the online selection, by running the algorithm as part of the trigger process; (ii) we do so exploiting deep-learning-based anomaly detection techniques.

Applying deep learning at the trigger level has been proposed in Ref. [119]. Recent works [255–258] have investigated the use of machine-learning techniques to setup new strategies for BSM searches with minimal or no assumption on the specific new-physics scenario under investigation. In this work, we use VAEs [251, 252] based on high-level features as a baseline. Previously, autoencoders have been used in collider physics for detector monitoring [259, 260] and event generation [261]. Autoencoders have also been explored to define a jet tagger that would identify new physics events with anomalous jets [262, 263], with a strategy similar to what we apply to the full event in this work.

Anomaly detection has been a traditional use case for one-class machine learning methods, such as one-class Support Vector Machine [264] or Isolation Forest [265, 266]. A review of proposed methods can be found in Ref. [267]. Variational methods have been shown to be effective for novelty detection, as for instance is discussed in Ref. [268]. In particular, VAEs [251] have been proposed as an effective method for anomaly detection [252].

8.3 Data samples

The dataset used for this study is a refined version of the high-level-feature (HLF) dataset used in Ref. [119]. Proton-proton collisions are generated using the `PYTHIA8` event-generation library [103], fixing the center-of-mass energy to the LHC Run-II value (13 TeV) and the average number of overlapping collisions per beam crossing (pileup) to 20. These beam conditions loosely correspond to the LHC operating conditions in 2016.

Events generated by `PYTHIA8` are processed with the `DELPHES` library [269], to emulate detector efficiency and resolution effects. We take as a benchmark detector description the upgraded design of the CMS detector, foreseen for the High-Luminosity LHC phase [270]. In particular, we use the CMS HL-LHC detector card distributed with `DELPHES`. We run the `DELPHES` *particle-flow* (PF) algorithm, which combines the information from different detector components to derive a list of reconstructed particles, the so-called PF candidates. For each particle, the algorithm returns the measured energy and flight direction. Each particle is associated

to one of three classes: charged particles, photons, and neutral hadrons. In addition, lists of reconstructed electrons and muons are given.

Many SM processes would contribute to the considered single-lepton dataset. For simplicity, we restrict the list of relevant SM processes to the four with the highest production cross sections, namely:

- Inclusive W production, with $W \to \ell\nu$ ($\ell = e,\ \mu,\ \tau$).
- Inclusive Z production, with $Z \to \ell\ell$ ($\ell = e,\ \mu,\ \tau$).
- $t\bar{t}$ production.
- QCD multijet production.[2]

These samples are mixed to provide a SM cocktail dataset, which is then used to train autoencoder models and to tune the threshold requirement that defines what we consider an anomaly. The cocktail is built scaling down the high-statistics samples ($t\bar{t}$, W, and Z) to the lowest-statistics one (QCD, whose generation is the most computing-expensive), according to their production cross-section values (estimated at leading order with `PYTHIA`) and selection efficiencies, shown in Table 8.1.

Events are filtered at generation requiring an electron, muon, or tau lepton with $p_T > 22$ GeV. Once detector effects are taken into account through the `DELPHES` simulation, events are further selected requiring the presence of one reconstructed lepton (electron or muon) with transverse momentum $p_T > 23$ GeV and a loose isolation requirement Iso < 0.45. If more than one reconstructed lepton is present, the highest p_T one is considered. The isolation for the considered lepton ℓ is computed as:

$$\text{Iso} = \frac{\sum_{p \neq \ell} p_T^p}{p_T^\ell} \,, \tag{8.1}$$

where the index p runs over all the photons, charged particles, and neutral hadrons within a cone of size $\Delta R = \sqrt{\Delta\eta^2 + \Delta\phi^2} < 0.3$ from ℓ.[3]

[2] To speed up the generation process for QCD events, we require $\sqrt{\hat{s}} > 10$ GeV, the fraction of QCD events with $\sqrt{\hat{s}} < 10$ GeV and producing a lepton within acceptance being negligible but computationally expensive.

[3] As common for collider physics, we use a Cartesian coordinate system with the z axis oriented along the beam axis, the x axis on the horizontal plane, and the y axis oriented upward. The x and y axes define the transverse plane, while the z axis identifies the longitudinal direction. The azimuth angle ϕ is computed from the x axis. The polar angle θ is used to compute the pseudorapidity $\eta = -\log(\tan(\theta/2))$. We fix units such that $c = \hbar = 1$.

Table 8.1: Acceptance and L1 trigger (i.e. p_T^ℓ and Iso requirement) efficiency for the four studied SM processes and corresponding values for the BSM benchmark models. For SM processes, we quote the total cross section before the trigger, the expected number of events per month and the fraction in the SM cocktail. For BSM models, we compute the production cross section corresponding to an average of 100 BSM events per month passing the acceptance and L1 trigger requirements. The monthly event yield is computed assuming an average luminosity per month of 5 fb^{-1}, corresponding to the running conditions discussed in Section 8.1.

Standard Model processes					
Process	Acceptance	L1 trigger efficiency	Cross section [nb]	Event fraction	Events /month
W	55.6%	68%	58	59.2%	110M
QCD	0.08%	9.6%	$1.6 \cdot 10^5$	33.8%	63M
Z	16%	77%	20	6.7%	12M
$t\bar{t}$	37%	49%	0.7	0.3%	0.6M

BSM benchmark processes				
Process	Acceptance	L1 trigger efficiency	Total efficiency	Cross-section 100 BSM events/month
$A \to 4\ell$	5%	98%	5%	0.44 pb
$LQ \to b\tau$	19%	62%	12%	0.17 pb
$h^0 \to \tau\tau$	9%	70%	6%	0.34 pb
$h^\pm \to \tau\nu$	18%	69%	12%	0.16 pb

The 21 considered HLF quantities are:

- The absolute value of the isolated-lepton transverse momentum p_T^ℓ.
- The three isolation quantities (ChPFIso, NeuPFIso, GammaPFIso) for the isolated lepton, computed with respect to charged particles, neutral hadrons and photons, respectively.
- The lepton charge.
- A Boolean flag (isEle) set to 1 when the trigger lepton is an electron, 0 otherwise.
- S_T, i.e. the scalar sum of the p_T of all the jets, leptons, and photons in the event with $p_T > 30$ GeV and $|\eta| < 2.6$. Jets are clustered from the reconstructed PF candidates, using the `FASTJET` [194] implementation of the anti-k_T jet algorithm [57], with a jet-size parameter R=0.4.

- The number of jets entering the S_T sum (N_J).
- The invariant mass of the set of jets entering the S_T sum (M_J).
- The number of these jets being identified as originating from a b quark (N_b).
- The missing transverse momentum, decomposed into its parallel ($p_{T,\parallel}^{\text{miss}}$) and orthogonal ($p_{T,\perp}^{\text{miss}}$) components with respect to the lepton ℓ direction. The missing transverse momentum is defined as the negative sum of the PF-candidate p_T vectors:
$$\vec{p}_T^{\,\text{miss}} = -\sum_q \vec{p}_T^{\,q} \ . \tag{8.2}$$
- The transverse mass, M_T, of the isolated lepton ℓ and the $\vec{p}_T^{\,\text{miss}}$ system, defined as:
$$M_T = \sqrt{2 p_T^\ell E_T^{\text{miss}} (1 - \cos\Delta\phi)} \ , \tag{8.3}$$
with $\Delta\phi$ the azimuth separation between the $\vec{p}_T^{\,\ell}$ and $\vec{p}_T^{\,\text{miss}}$ vectors, and E_T^{miss} the magnitude of $\vec{p}_T^{\,\text{miss}}$.
- The number of selected muons (N_μ).
- The invariant mass of this set of muons (M_μ).
- The absolute value of the total transverse momentum of these muons ($p_{T,TOT}^{\mu}$).
- The number of selected electrons (N_e).
- The invariant mass of this set of electrons (M_e).
- The absolute value of the total transverse momentum of these electrons ($p_{T,TOT}^{e}$).
- The number of reconstructed charged hadrons.
- The number of reconstructed neutral hadrons.

This list of HLF quantities is not defined having in mind a specific BSM scenario. Instead, it is conceived to include relevant information to discriminate the various SM processes populating the single-lepton data stream. On the other hand, it is generic enough to allow (at least in principle) the identification of a large set of new physics scenarios.

In addition to the four SM processes listed above, we consider the following BSM models to benchmark anomaly-detection capabilities:

- A leptoquark LQ with mass 80 GeV, decaying to a b quark and a τ lepton.
- A neutral scalar boson with mass 50 GeV, decaying to two off-shell Z bosons, each forced to decay to two leptons: $A \to 4\ell$.
- A scalar boson with mass 60 GeV, decaying to two tau leptons: $h^0 \to \tau\tau$.
- A charged scalar boson with mass 60 GeV, decaying to a tau lepton and a neutrino: $h^\pm \to \tau\nu$.

For each BSM scenario, we consider any direct production mechanism implemented in `PYTHIA8`, including associate jet production. We list in Table 8.1 the leading-order production cross section and selection efficiency for each model.

Figures 8.1 and 8.2 show the distribution of HLF quantities for the SM processes and the BSM benchmark models, respectively.

8.4 Model description

We train VAEs on the SM cocktail sample described in Section 8.3, taking as input the 21 HLF quantities listed there. The use of HLF quantities to represent events limits the model independence of the anomaly detection procedure. While the list of features is chosen to represent the main physics aspects of the considered SM processes and is in no way tailored to specific BSM models, it is true that such a list might be more suitable for certain models than for others. In this respect, one cannot guarantee that the anomaly-detection performance observed on a given BSM model would generalize to any BSM scenario. We will address in a future work a possible solution to reduce the residual model dependence implied by the input event representation.

In this section, we present both the best-performing autoencoder model, trained to encode and decode the SM training sample, and a set of four supervised classifiers, each trained to distinguish one of the four BSM benchmark models from SM events. We use the classification performance of these supervised algorithms as an estimate of the best performance that the VAE could get to.

Autoencoders

Autoencoders are algorithms that compress a given set of inputs variables in a latent space (encoding) and then, starting from the latent space, reconstruct the HLF input values (decoding). The loss distribution of an AE is used in the context of anomaly

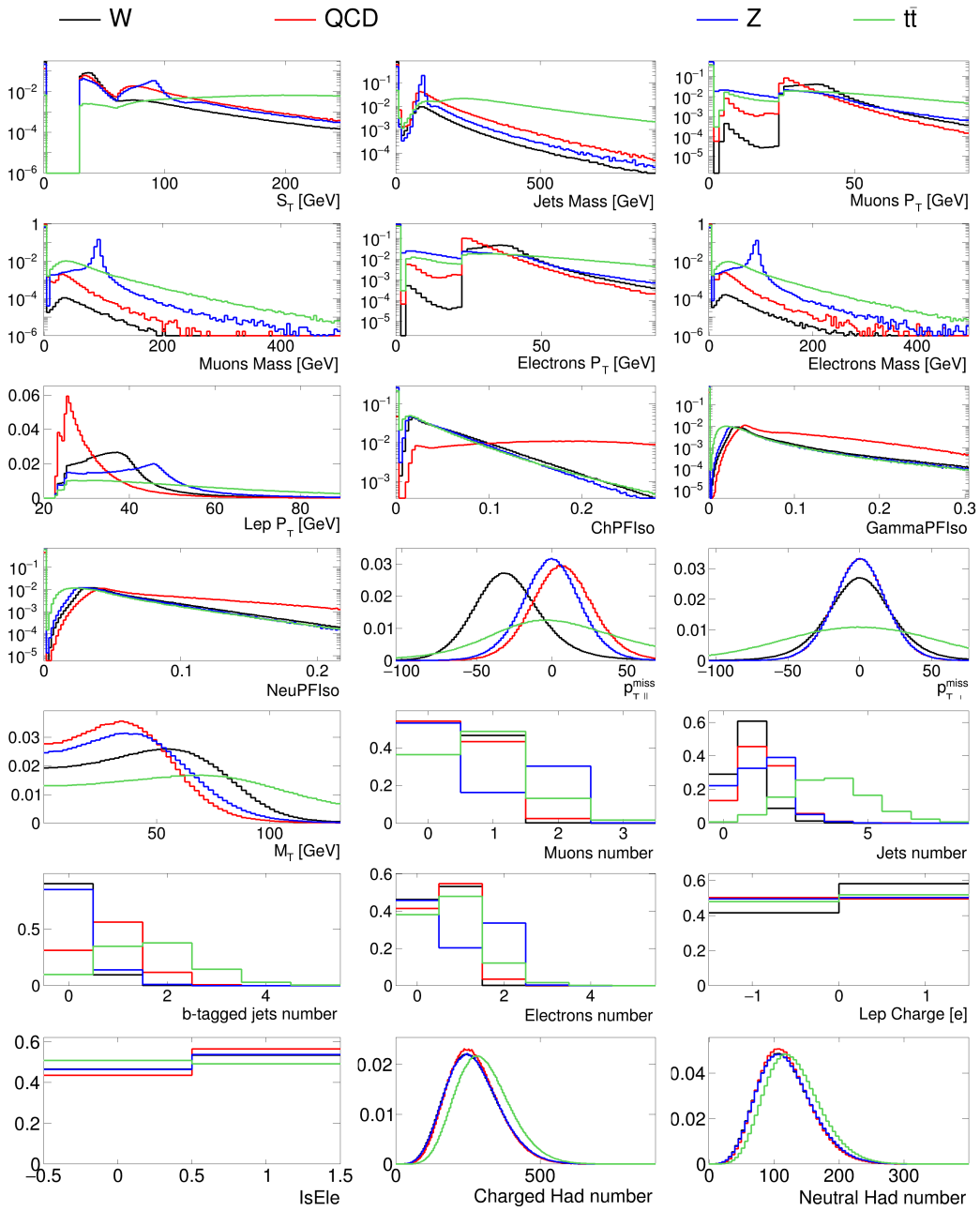


Figure 8.1: Distribution of the HLF quantities for the four considered SM processes.

detection to isolate potential anomalies. Since the compression capability learned on a given sample does not typically generalize to other samples, the tails of the loss distribution could be enriched by new kinds of events, different than those used to train the model. In the specific case considered in this study, the tail of the loss distribution for an AE trained on SM data might be enriched with BSM events.

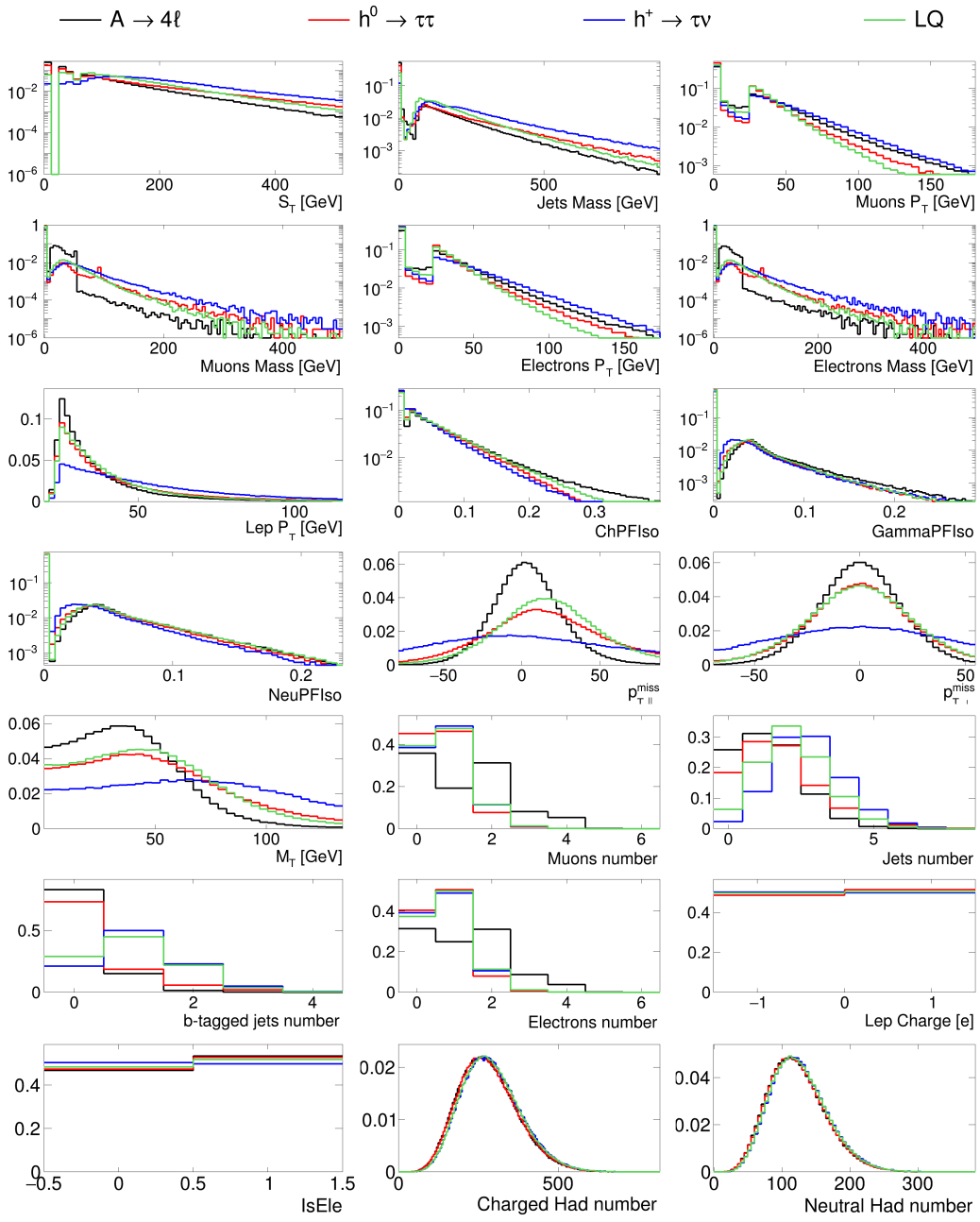


Figure 8.2: Distribution of the HLF quantities for the four considered BSM benchmark models.

In this work, we focus on VAEs [251]. For each event, a plain AE predicts an encoded point in the latent space and a decoded point in the original space. In other words, AEs are point-estimate algorithms. VAEs, instead, associate to each input event an estimated probability distributions in the latent space and in the original space. Doing so, VAEs provide both a best-point estimate and an estimate of the

associated statistical noise. Besides this conceptual difference, VAEs have been shown to provide competitive performances for novelty [268] and anomaly [252] detection.

We consider the VAE architecture shown in Fig. 8.3, characterized by a four-dimensional latent space. Each latent dimension is associated to a Gaussian pdf and its two degrees of freedom (mean μ_z and RMS σ_z). The input layer consists of 21 nodes, corresponding to the 21 HLF quantities described in Section 8.3. This layer is connected to the latent space through a stack of two fully connected layers, each consisting of 50 nodes with ReLU activation functions. Two four-node layers are fully connected to the second 50-node layer. Linear activation functions are used for the first of these four-node layers, interpreted as the set of four μ_z of the four-dimension Gaussian pdf $p(z)$. The nodes of the second layer are activated by the functions:

$$\text{p-ISRLu}(x) = 1 + 5 \cdot 10^{-3} + \Theta(x)x + \Theta(-x)\frac{x}{\sqrt{1+x^2}} \, . \tag{8.4}$$

This activation allows to improve the training stability, being strictly positive defined, non linear, and with no exponentially growing term (which might have created instabilities in the early epochs of the training). The four nodes of this layer are interpreted as the σ_z parameters of $p(z)$. After several trials, the dimension of the latent space has been set to 4 in order to keep a good training stability without impacting the VAE performances. The decoding step originates from a point in the latent space, sampled according to the predicted pdf (green oval in Fig. 8.3). The coordinates of this point in the latent space are fed into a sequence of two hidden dense layers, each consisting of 50 neurons with ReLU activation functions. The last of these layers is connected to three dense layers of 21, 17, and 10 neurons, activated by linear, p-ISRLu and clipped-tanh functions, respectively. The clipped-tanh function if written as:

$$C_{\tanh}(x) = \frac{1}{2}(1 + 0.999 \cdot \tanh x) \, . \tag{8.5}$$

Given the latent-space representation, the 48 output nodes represent the parameters of the pdfs describing the input HLF probability, i.e., the α parameters of Eq. 8.8.

The total VAE loss function Loss_{Tot} is a weighted sum of two pieces [271]: a term related to the reconstruction likelihood ($\text{Loss}_{\text{reco}}$) and the Kullback-Leibler divergence (D_{KL}) between the latent space pdf and the prior:

$$\text{Loss}_{\text{Tot}} = \text{Loss}_{\text{reco}} + \beta D_{\text{KL}} \, , \tag{8.6}$$

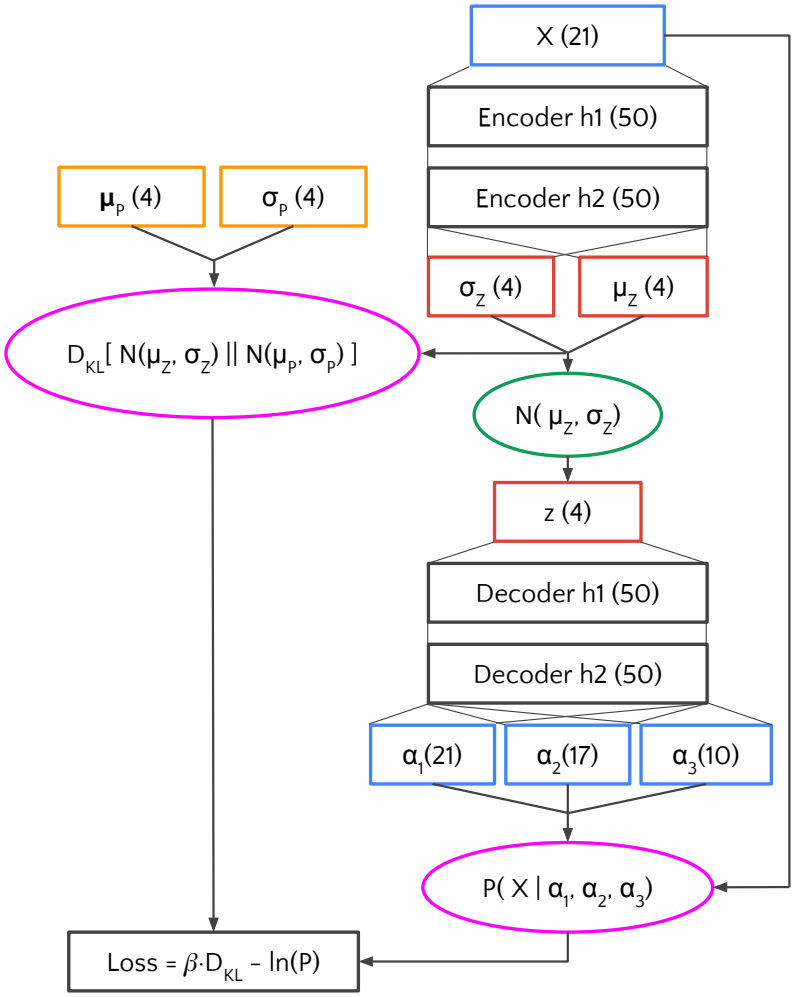


Figure 8.3: Schematic representation of the VAE architecture presented in the text. The size of each layer is indicated by the value within brackets. The blue rectangle X represents the input layer, which is connected to a stack of two consecutive fully connected layers (black boxes). The last of the two black box is connected to two layers with four nodes each (red boxes), representing the μ_z and σ_z parameters of the encoder pdf $p(z|x)$. The green oval represents the sampling operator, which returns a set of values for the 4-dimensional latent variables z. These values are fed into the decoder, consisting of two consecutive hidden layers of 50 nodes each (black boxes). The last of the decoder hidden layer is connected to the three output layers, whose nodes correspond to the parameters of the predicted distribution in the initial 21-dimension space. The pink ovals represent the computation of the two parts of the loss function: the KL loss and the reconstruction loss (see text). The computation of the KL requires 8 additional learnable parameters (μ_p and σ_p, represented by the orange boxes on the top-left part of the figure), corresponding to the means and RMS of the four-dimensional Gaussian prior $p(z)$. The total loss in computed as described by the formula in the bottom-left black box (see Eq. (8.6)).

where β is a free parameter. We fix $\beta = 0.3$, for which we obtained good reconstruction performances.[4] The prior $p(z)$ chosen for the latent space is a four-dimension Gaussian with a diagonal covariance matrix. The means (μ_P) and the diagonal terms of the covariance matrix (σ_P) are free parameters of the algorithm and are optimized during the back-propagation. The Kullback-Leibler divergence between two Gaussian distributions has an analytic form. Hence, for each batch, D_{KL} can be expressed as:

$$\begin{aligned} D_{\mathrm{KL}} &= \frac{1}{k}\sum_i D_{\mathrm{KL}}\left(N(\mu_z^i, \sigma_z^i) \;||\; N(\mu_P, \sigma_P)\right) \\ &= \frac{1}{2k}\sum_{i,j}\left(\sigma_P^j \sigma_z^{i,j}\right)^2 + \left(\frac{\mu_P^j - \mu_z^{i,j}}{\sigma_P^j}\right)^2 + \ln\frac{\sigma_P^j}{\sigma_z^{i,j}} - 1 \;, \end{aligned} \tag{8.7}$$

where k is the batch size, i runs over the samples and j over the latent space dimensions. Similarly, $\mathrm{Loss}_{\mathrm{reco}}$ is the average negative-log-likelihood of the inputs given the predicted α values:

$$\begin{aligned} \mathrm{Loss}_{\mathrm{reco}} &= -\frac{1}{k}\sum_i \ln\left[P(x \mid \alpha_1, \alpha_2, \alpha_3)\right] \\ &= -\frac{1}{k}\sum_{i,j} \ln\left[f_j(x_{i,j} \mid \alpha_1^{i,j}, \alpha_2^{i,j}, \alpha_3^{i,j})\right] \;. \end{aligned} \tag{8.8}$$

In the equation, j runs over the input space dimensions, f_j is the functional form chose to describe the pdf of the j-th input variable and $\alpha_m^{i,j}$ are the parameter of the function. Different functional forms have been chosen for f_j, to properly describe different classes of HLF distributions:

- **Clipped Log-normal + δ function**: used to describe S_T, M_J, p_T^{μ}, M_{μ}, p_T^{e}, M_e, p_T^{ℓ}, ChPFIso, NeuPFIso and GammaPFIso:

$$P(x \mid \alpha_1, \alpha_2, \alpha_3) = \begin{cases} \alpha_3\delta(x) + \frac{1-\alpha_3}{x\alpha_2\sqrt{2\pi}}\exp\left(-\frac{(\ln x - \alpha_1)^2}{2\alpha_2^2}\right) \;\; for\; x \geq 10^{-4} \\ 0 \;\; for\; x < 10^{-4} \end{cases} . \tag{8.9}$$

- **Gaussian**: used for $p_{T,\parallel}^{\mathrm{miss}}$ and $p_{T,\perp}^{\mathrm{miss}}$:

$$P(x \mid \alpha_1, \alpha_2) = \frac{1}{\alpha_2\sqrt{2\pi}}\exp\left(-\frac{(x-\alpha_1)^2}{2\alpha_2^2}\right) \;. \tag{8.10}$$

[4] Following Ref. [271], we tried to increase the value of β up to 4 without observing a substantial difference in performance.

- **Truncated Gaussian**: a Gaussian function truncated for negative values and normalized to unit area for $X > 0$. Used to model M_T:

$$P(x \mid \alpha_1, \alpha_2) = \Theta(x) \cdot \frac{1 + 0.5 \cdot (1 + \mathrm{erf}\frac{-\alpha_1}{\alpha_2\sqrt{2}})}{\alpha_2\sqrt{2\pi}} \exp\left(-\frac{(x-\alpha_1)^2}{2\alpha_2^2}\right) . \quad (8.11)$$

- **Discrete truncated Gaussian**: like the truncated Gaussian, but normalized to be evaluated on integers (i.e. $\sum_{n=0}^{\infty} P(n) = 1$). This function is used to describe N_μ, N_e, N_b and N_J. It is written as:

$$P(n \mid \alpha_1, \alpha_2) = \Theta(x) \left[\mathrm{erf}\left(\frac{n + 0.5 - \alpha_1}{\alpha_2\sqrt{2}}\right) - \mathrm{erf}\left(\frac{n - 0.5 - \alpha_1}{\alpha_2\sqrt{2}}\right)\right] \mathcal{N} , \quad (8.12)$$

  where the normalization factor $\mathcal{N}$ is set to:

$$\mathcal{N} = 1 + \frac{1}{2}\left(1 + \mathrm{erf}\left(\frac{-0.5 - \alpha_1}{\alpha_2\sqrt{2}}\right)\right) . \quad (8.13)$$

- **Binomial**: used for (isEle) and lepton charge:

$$P(n \mid p) = \delta_{n,m} p + \delta_{n,l}(1 - p) , \quad (8.14)$$

  where m and l are the two possible values of the variable (0 or 1 for (isEle) and -1 or 1 for lepton charge) and $p = C_{\tanh}(\alpha_1)$.

- **Poisson**: used for charged-particle and neutral-hadron multiplicities:

$$P(n \mid \mu) = \frac{\mu^n e^{-\mu}}{\Gamma(n+1)} , \quad (8.15)$$

  where μ = p-ISRLu(α_1).

These custom functions provide an improved performance with respect to the standard choice of an MSE loss. When using the MSE loss, one is implicitly writing the likelihood of the input quantities as a product of Gaussian functions with equal variance. This choice is clearly a poor description of the input distributions at hand in this application and it results in a poor representation of the cores and the tails of the input distributions. Instead, the use of these tailored functions allows to correctly describe the distribution cores and to improve the description of the tails.

We point out that the final performance depends on the choice of the p(x|z) functional form (i.e., on the modeled dependence of the observed features on the latent

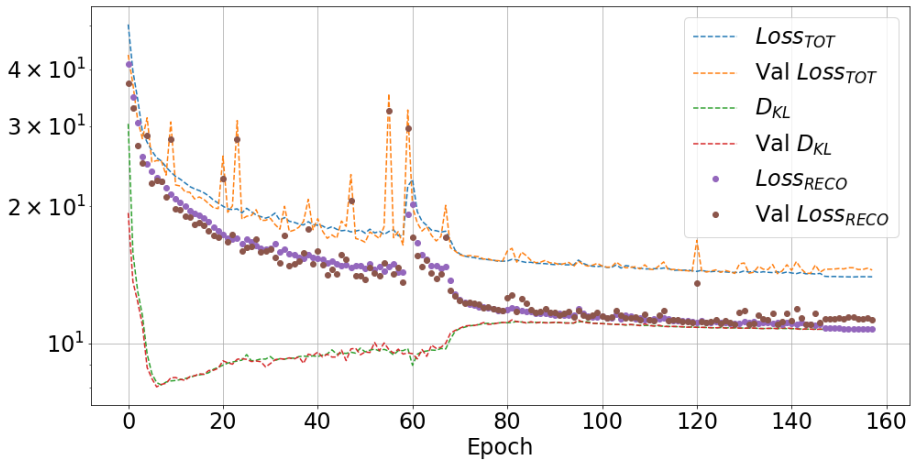


Figure 8.4: Training history for VAE. Total loss, reconstruction negative-log-likelihood ($\text{Loss}_{\text{reoc}}$) and KL divergence ($D_{KL}$) are shown separately for training and validation set though all the training epochs.

variables) and the p(z) prior function. The former was tuned looking at the distributions for SM events. The latter is arbitrary. We explored techniques to optimize the choice of p(z), learning it from the data [272]. In this case, no practical advantage in terms of anomaly detection was observed. An improved choice of p(x|z) and the possibility of learning p(z) during the train could potentially further boost the performances of this algorithm and will be the subject of future studies with real LHC collision data.

The model shown in Fig. 8.3 is implemented in `Keras+TensorFlow` [120, 121], trained with the Adam optimizer [212] on a SM dataset of 3.45M events, equivalent to an integrated luminosity of $\sim$ 100 pb^{-1}. The SM validation dataset is made of 3.45M of statistically independent examples. Such a sample would be collected in about ten hours of continuous run, under the assumptions made in this study (see Section 8.1). In training, we fix the batch size to 1000. We use early stopping with patience set to 20 and $\delta_{\text{min}} = 0.005$, and we progressively reduce the learning rate on plateau, with patience set to 8 and $\delta_{\text{min}} = 0.01$.

The model's training history is shown in Fig. 8.4. Figure 8.5 shows the comparison of the input and output distributions for the 21 HLF quantities in the validation dataset. A general good agreement is observed on the bulk of the distributions, even if some of the distributions are not well described on the tails. These discrepancies do not have a sizable impact on the anomaly-detection strategy, as shown in

Section. 8.5. Nevertheless, alternative architectures were tested, in order to reduce these discrepancies. For instance, we increased or decreased the dimensionality of the latent space, we changed the value of β in Eq. 8.6, we changed the number of neurons in the hidden layers, tried the RMSprop optimizer, and used plain Gaussian functions to describe the 21 input features. Some of these choices improved the encoding-decoding capability of the VAE, with up to a 10% decrease of the loss function at the end of the training. On the other hand, none of these alternative models provided a sizable improvement in the anomaly-detection performance. For simplicity, we decided to limit our study to the architecture in Fig. 8.3 and dropped these alternative models.

Supervised classifiers

For each of the four BSM benchmark models, we train a fully-supervised classifier, based on a Boosted Decision Tree (BDT). Each BDT receives as input the same 21 features used by the VAE and is trained on a labeled dataset consisting of the SM cocktail (the background) and one of the four BSM benchmark models (the signal). The implementation is done through the Gradient Boosted Classifier of the scikit-learn library [273]. The algorithm was tuned with up to 150 estimators, minimum samples per leaf and maximum depth equal to 3, a learning rate of 0.1, and a tolerance of 10^{-4} on the validation loss function (choose to be the default deviance). Each BDT, tailored to a specific BSM model, is trained on 3.45M SM events and about 0.5M BSM events, consistently up-weighted in order to match the size of the SM sample during the training.

Table 8.2: Classification performance of the four BDT classifiers described in the text, each trained on one of the four BSM benchmark models. The two set of values correspond to the area under ROC curve (AUC), and to the true positive rate (TPR) for a SM false positive rate $\epsilon_{SM} = 5.4 \cdot 10^{-6}$, i.e., to $\sim$ 1000 SM events accepted every month.

Process	AUC	TPR [%]
$A \to 4\ell$	0.98	5.4
$LQ \to b\tau$	0.94	0.2
$h^0 \to \tau\tau$	0.90	0.1
$h^\pm \to \tau\nu$	0.97	0.3

We show in Table 8.2 and in Figure 8.6 the classification performance of the four supervised BDTs, which set a qualitative upper limit for VAE's results. Overall,

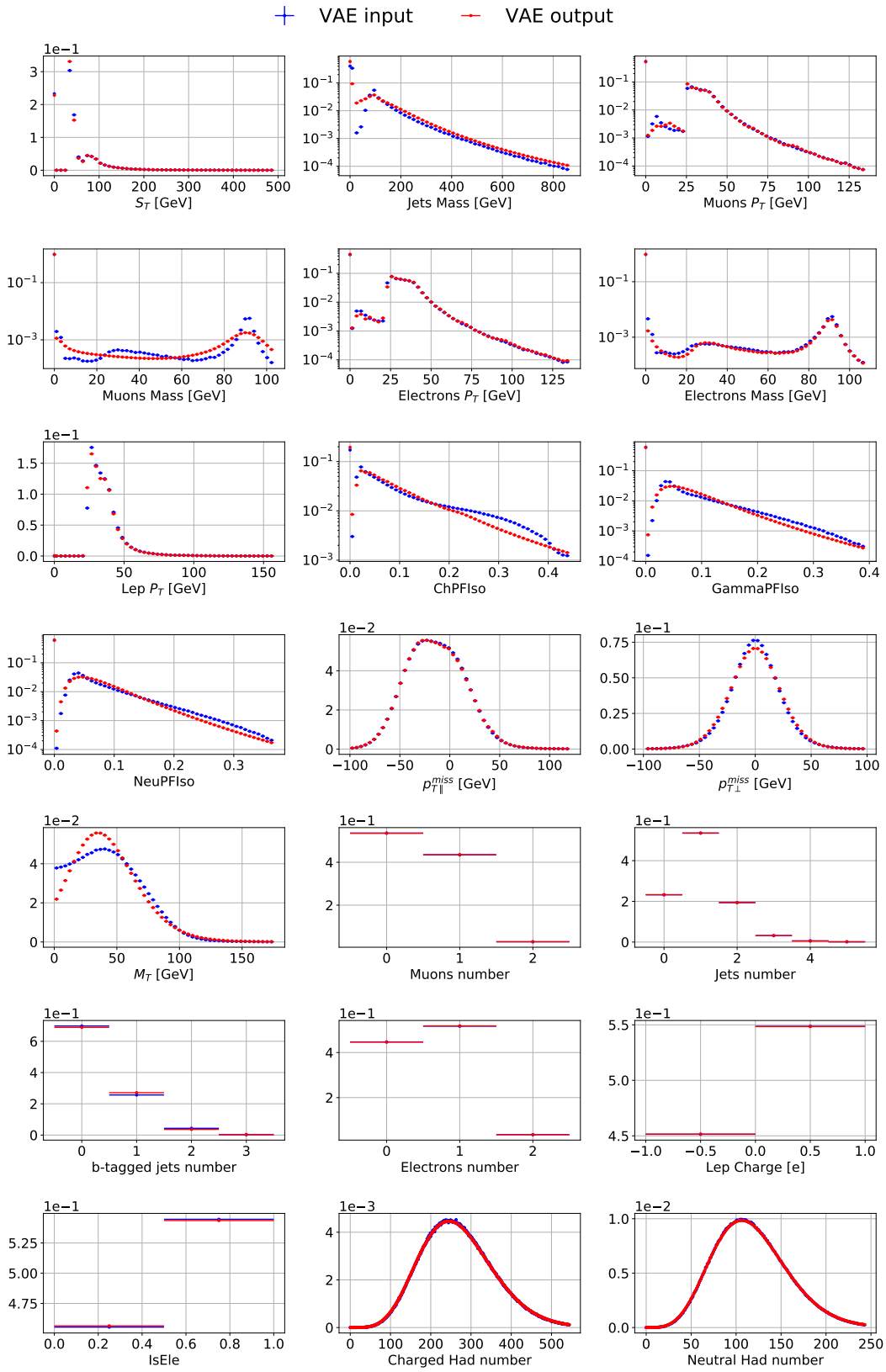


Figure 8.5: Comparison of input (blue) and output (red) probability distributions for the HLF quantities in the validation sample. The input distributions are normalized to unity. The output distributions are obtained summing over the predicted pdf of each event, normalized to the inverse of the total number of events (so that the total sum is normalized to unity).

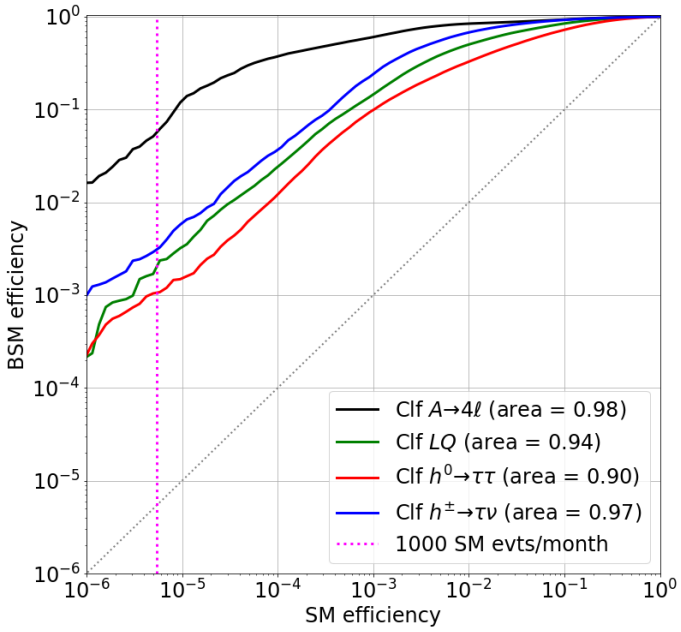


Figure 8.6: ROC curves for the fully-supervised BDT classifiers, optimized to separate each of the four BSM benchmark models from the SM cocktail dataset.

the four models can be discriminated with good accuracy, with some loss of performance for those models sharing similarities with specific SM processes (e.g., $h^0 \to \tau\tau$ exhibiting single- and double-lepton topology with missing transverse energy, typical of $t\bar{t}$ events). In the table, we also quote the true-positive rate (TPR) for each BSM model corresponding to a working point of SM false positive rate $\epsilon_{SM} = 5.4 \cdot 10^{-6}$, corresponding to an average of $\sim$ 1000 SM events accepted every month.

In addition to BDTs, we experimented with fully-connected deep neural networks (DNNs) with two hidden layers. Despite trying different architectures, we did not find a configuration in which the DNN classifiers could outperform the BDTs. This is due to the fact that, given the limited complexity of the problem at hand, a simple BDT can extract the maximum discrimination power from the 21 inputs. The limiting factor preventing to reach larger auc values is not to be found in the model complexity but in the discriminating power of the 21 input features. Not being tailored on the benchmark BSM scenarios, these features do not carry all the needed information for an optimal signal-to-background separation. While certainly one could obtain a better performance with more tailored classifiers, the purpose

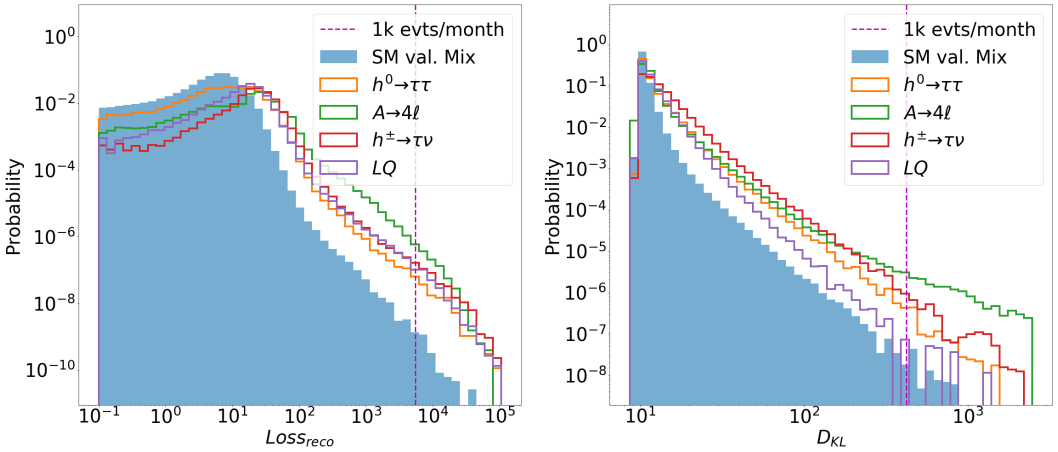


Figure 8.7: Distribution of the VAE's loss components, $\text{Loss}_{\text{reco}}$ (left) and D_{KL} (right), for the validation dataset. For comparison, the corresponding distribution for the four benchmark BSM models are shown. The vertical line represents a lower threshold such that $5.4 \cdot 10^{-6}$ of the SM events would be retained, equivalent to $\sim$ 1000 expected SM events per month.

of this exercise was to provide a fair comparison for the VAE. In view of these considerations, we decided to use the BDTs as reference supervised classifiers.

8.5 Results with VAE

An event is classified as anomalous whenever the associated loss, computed from the VAE output, is above a given threshold. Since no BSM signal has been observed by LHC experiments so far, it is reasonable to expect that a new-physics signal, if any, would be characterized by a low production cross section and/or features very similar to those of a SM process. In view of this, we decided to use a tight threshold value, in order to reduce as much as possible any SM contribution.

Figure 8.7 shows the distribution of the $\text{Loss}_{\text{reco}}$ and D_{KL} loss components for the validation dataset. In both plots, the vertical line represents a lower threshold such that a fraction $\epsilon_{SM} = 5.4 \cdot 10^{-6}$ of the SM events would be retained. This threshold value would result in $\sim$ 1000 SM events to be selected every month, i.e., a daily rate of $\sim$ 33 SM events, as illustrated in Table 8.3. The acceptance rate is calculated assuming the LHC running conditions listed in Section 8.1. Table 8.3 also reports the by-process VAE selection efficiency and the relative background composition of the selected sample.

Figure 8.7 also shows the $\text{Loss}_{\text{reco}}$ and D_{KL} distributions for the four benchmark BSM models. We observe that the discrimination power, loosely quantified by the

integral of these distributions above threshold, is better for $\text{Loss}_{\text{reco}}$ than D_{KL} and that the impact of the D_{KL} term on Loss_{Tot} is negligible. Anomalies are then defined as events laying on the right tail of the expected $\text{Loss}_{\text{reco}}$ distribution. Due to limited statistics in the training sample, the p-value corresponding to the chosen threshold value could be uncalibrated. This could result in a deviation of the observed rate from the expected value, an issue that one can address tuning the threshold. On the other hand, an uncalibrated p-value would also impact the number of collected BSM events, and the time needed to collect an appreciable amount of these events.

Once the $\text{Loss}_{\text{reco}}$ selection is applied, the anomalous events do not cluster on the tails of the distributions of the input features. Instead, they tend to cover the full feature-definition range. This is an indication of the fact that the VAE does more than a simple selection of feature outliers, which is what is done by traditional single-lepton trigger or by dedicated cross triggers (e.g., triggers that select events with soft leptons and large missing transverse energy, S_T, etc.). This is shown in Fig. 8.8 for SM events. A similar conclusion can be obtained from Fig. 8.9, showing the distribution of the 21 input HLF quantities for the $A \rightarrow 4\ell$ benchmark model, before and after applying the threshold requirement on the VAE loss.

The left plot in Fig. 8.10 shows the ROC curves obtained from the $\text{Loss}_{\text{reco}}$ distribution of the four BSM benchmark models and the SM cocktail, compared to the corresponding BDT curves of Section 8.4. As expected, the results obtained with the supervised BDTs outperform the VAE. On the other hand, the VAE can probe at the same time the four scenarios with comparable performances. This is a consequence of the trade off between precision and model independence and an illustration of the complementarity between the approach presented in this work and traditional supervised techniques. The right plot in Fig. 8.10 shows the one-sided p-value computed from the cocktail SM distribution, both for the SM events themselves (flat by construction) and for the four BSM processes. As the plot shows, BSM processes tend to concentrate at small p-values, which allows their identification as anomalies.

Table 8.4 summarizes the VAE's performance on the four BSM benchmark models. Together with the selection efficiency corresponding to $\epsilon_{SM} = 5.4 \cdot 10^{-6}$, the table reports the effective cross section (cross section after applying the trigger requirements) that would correspond to 100 BSM events selected in a month (assuming an integrated luminosity of 5 fb^{-1}). Similarly, we quote the cross section that would result in a signal-to-background ratio of 1/3 on the sample of events selected by the VAE. The VAE can probe the four models down to small cross section values,

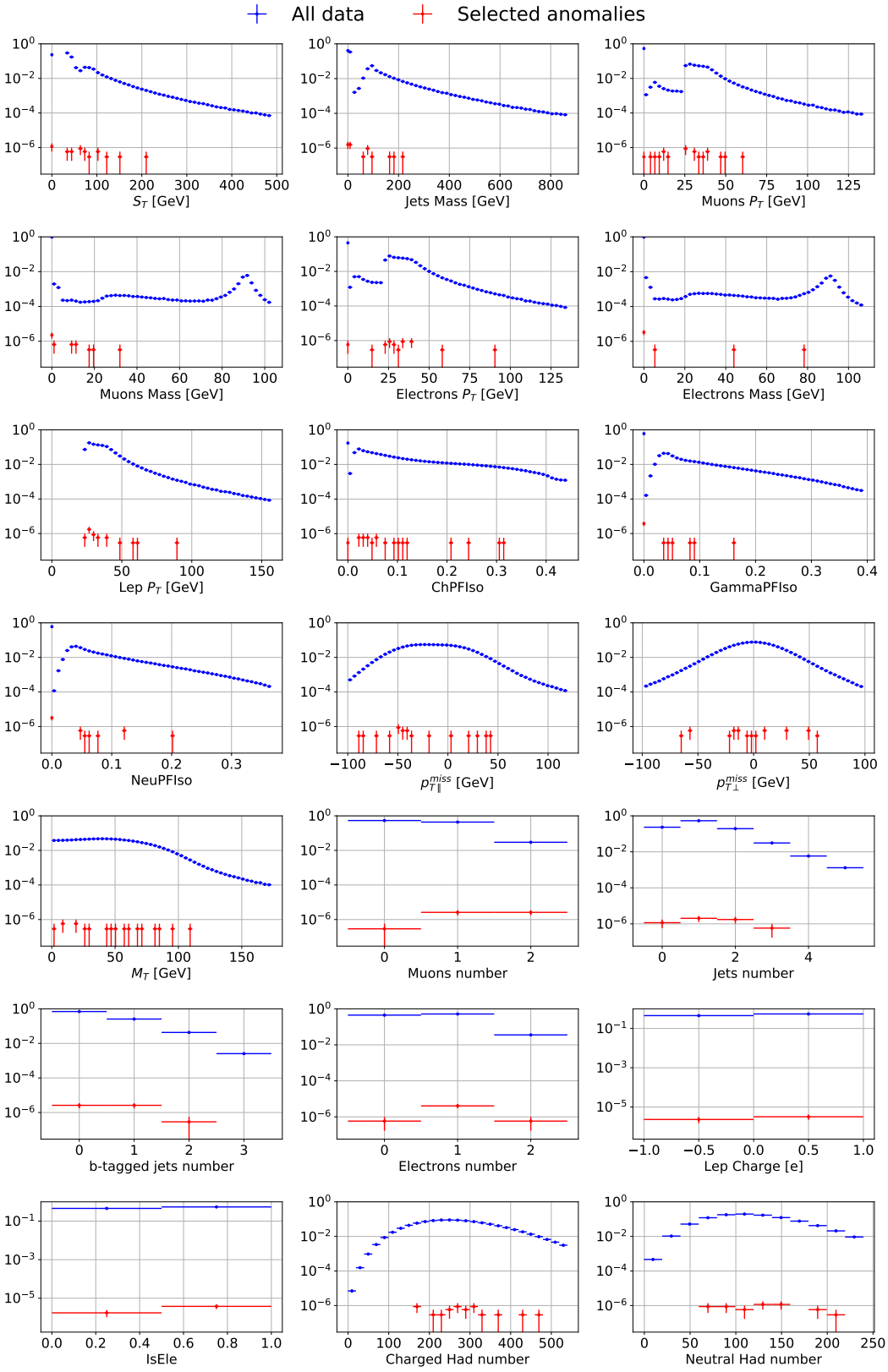


Figure 8.8: Comparison between the input distribution for the 21 HLF of the validation dataset (blue histograms) and the distribution of the SM outlier events selected from the same sample by applying the $\text{Loss}_{\text{reco}}$ threshold (red dots). The outlier events cover a large portion of the HLF definition range and do not cluster on the tails.

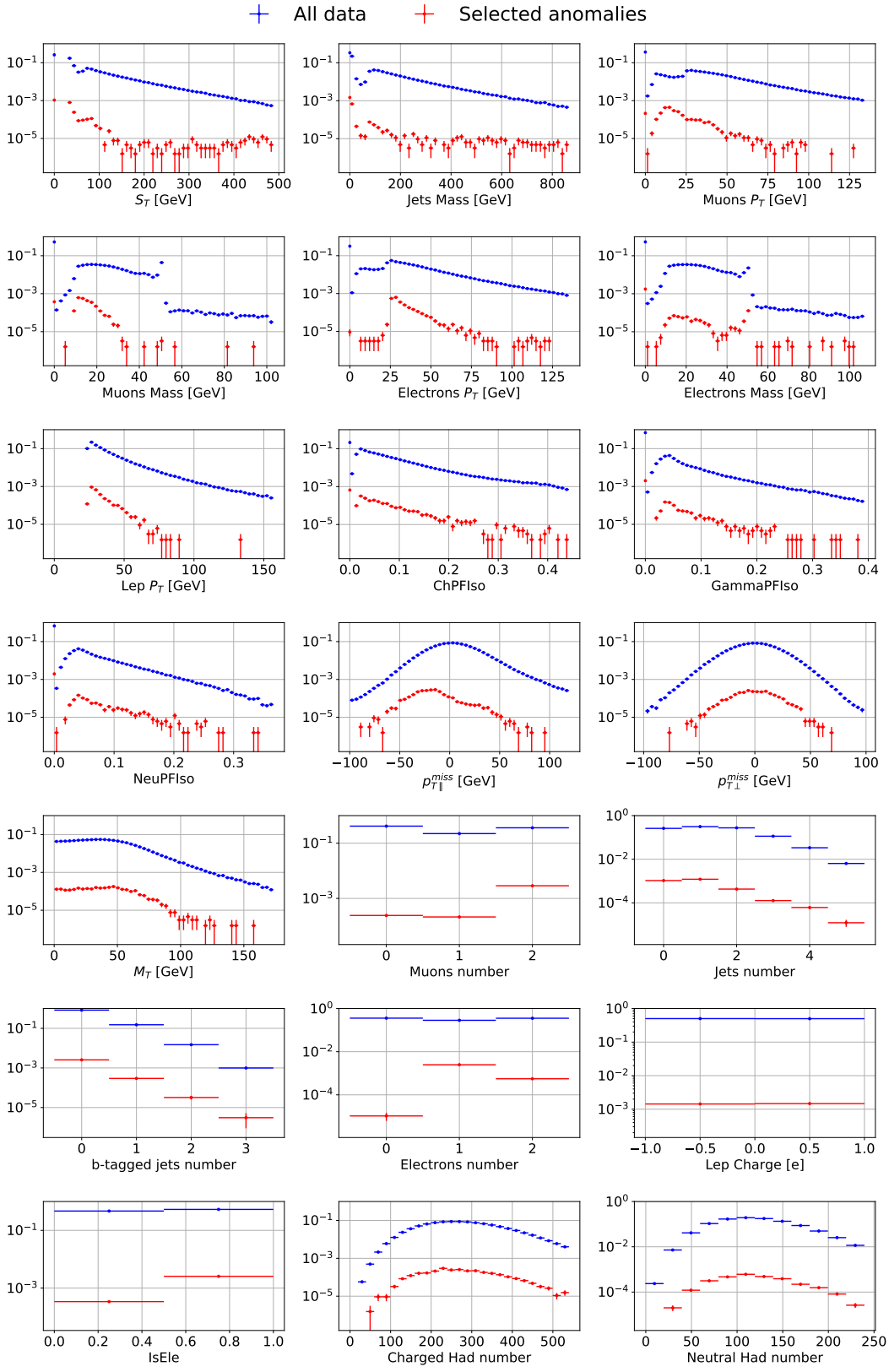


Figure 8.9: Comparison between the distribution of the 21 HLF distribution for $A \rightarrow 4\ell$ full dataset (blue) and $A \rightarrow 4\ell$ events selected by applying the $\text{Loss}_{\text{reco}}$ threshold (red). The selected events are not trivially sampled from the tail.

Table 8.3: By-process acceptance rate for the anomaly detection algorithm described in the text, computed applying the threshold on $\text{Loss}_{\text{reco}}$ shown in Fig. 8.7. The threshold is tuned such that a fraction of about $\epsilon_{SM} = 5.4 \cdot 10^{-6}$ of SM events would be accepted, corresponding to $\sim$ 1000 SM events/month, assuming the LHC running conditions listed in Section 8.1. The sample composition refers to the subset of SM events accepted by the anomaly detection algorithm. All quoted uncertainties refer to 95% CL regions.

Standard Model processes			
Process	VAE selection	Sample composition	Events/month
W	$3.6 \pm 0.7 \cdot 10^{-6}$	32%	379 ± 74
QCD	$6.0 \pm 2.3 \cdot 10^{-6}$	29%	357 ± 143
Z	$21 \pm 3.5 \cdot 10^{-6}$	21%	256 ± 43
$t\bar{t}$	$400 \pm 9 \cdot 10^{-6}$	18%	212 ± 5
Total			1204 ± 167

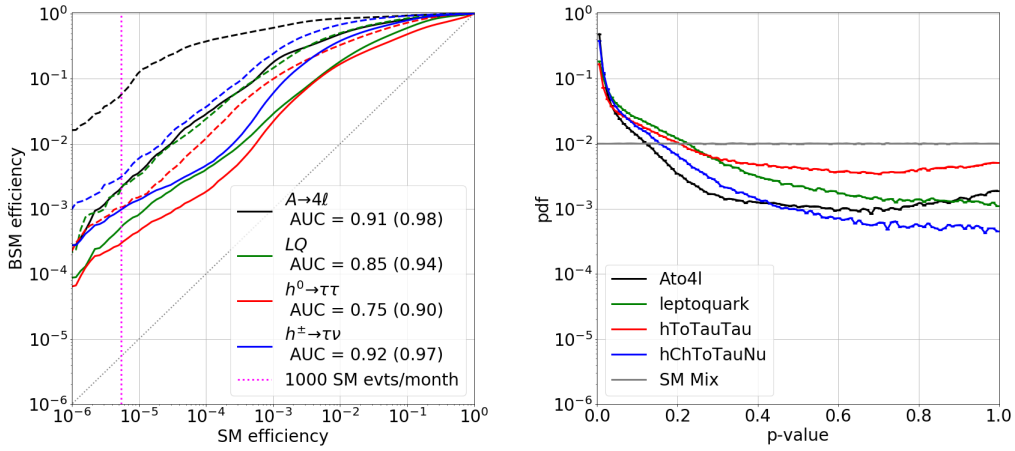


Figure 8.10: Performance of the VAE for different BSM scenarios. Left: ROC curves for the VAE trained only on SM events (solid), compared to the corresponding curves for the four supervised BDT models (dashed) described in Section 8.4. Right: Normalized p-value distribution distribution for the SM cocktail events and the four BSM benchmark processes.

comparable to the existing exclusion bounds for these mass ranges. As an example, Ref. [274] excludes a $LQ \rightarrow \tau b$ with a mass of 150 GeV and production cross section larger than $\sim$ 10 pb, using 4.8 fb^{-1} at a center-of-mass energy of 7 TeV, while most recent searches [275] cannot cover such a low mass value, due to trigger limitations.

Table 8.4: Breakdown of BSM processes efficiency, and cross section values corresponding to 100 selected events in a month and to a signal-over-background ratio of 1/3 (i.e., an absolute yield of $\sim$ 400 events/month). The monthly event yield is computed assuming an average luminosity per month of 5 fb^{-1}, computing by taking the LHC 2016 data delivery ($\sim$ 40 fb^{-1} collected in 8 months). All quoted efficiencies are computed fixing the VAE loss threshold $\epsilon_{SM} = 5.4 \cdot 10^{-6}$.

BSM benchmark processes			
Process	VAE selection efficiency	Cross-section 100 events/month [pb]	Cross-section S/B = 1/3 [pb]
$A \to 4\ell$	$2.8 \cdot 10^{-3}$	7.1	27
$LQ \to b\tau$	$6.7 \cdot 10^{-4}$	30	110
$h^0 \to \tau\tau$	$3.6 \cdot 10^{-4}$	55	210
$h^\pm \to \tau\nu$	$1.2 \cdot 10^{-3}$	17	65

Unlike a traditional trigger strategy, a VAE-based selection is mainly intended to select a high-purity sample of interesting event, at the cost of a typically small selection efficiency. To demonstrate this point, we consider a sample selected with the VAE and one selected using a typical inclusive single lepton trigger (SLT), consisting on a tighter selection than the one described in section 8.3. In particular, we require $p_T^\ell > 27$ GeV and ISO < 0.25. We consider the signal-over-background ratio (SBR) for the VAE's threshold selection and the SLT. While these quantities depend on the production cross section of the considered BSM model, their ratio

$$\frac{\mathrm{SBR_{VAE}}}{\mathrm{SBR_{SLT}}} = \left(\frac{\epsilon_{\mathrm{SLT}}}{\epsilon_{\mathrm{VAE}}}\right)_{SM} \cdot \left(\frac{\epsilon_{\mathrm{VAE}}}{\epsilon_{\mathrm{SLT}}}\right)_{BSM} \tag{8.16}$$

is only a function of the selection efficiency for the SLT (ϵ_{SLT}) and the for the VAE ϵ_{VAE} for SM and BSM events. Table 8.5 shows how the SBR reached by the VAE is about two order of magnitude larger than what a traditional inclusive SLT could reach.

8.6 Deployment in high-level triggers

The work presented in this paper suggests the possibility of deploying a VAE as a trigger algorithms associated to dedicated data streams. This trigger would isolate anomalous events, similarly to what was done by the CMS experiment at the beginning of the first LHC run. With early new physics signal being a possibility at the LHC start, the CMS experiment deployed online a set of algorithms (collectively called *hot line*) to select potentially interesting new-physics candidates. At that time, anomalies were characterized as events with high-p_T particles or high

Table 8.5: Selection efficiencies for a typical single lepton trigger (SLT) and the proposed VAE selection, shown for the four benchmark BSM models and for the SM cocktail. The last row quotes the corresponding BSM-to-SM ratio of signal-over-background ratios (SBRs), quantifying the purity of the selected sample.

	SM	$A \to 4\ell$	$LQ \to b\tau$	$h^0 \to \tau\tau$	$h^\pm \to \tau\nu$
ϵ_{VAE}	$5.3 \cdot 10^{-6}$	$2.8 \cdot 10^{-3}$	$6.7 \cdot 10^{-4}$	$3.6 \cdot 10^{-4}$	$1.2 \cdot 10^{-3}$
ϵ_{SLT}	0.6	0.5	0.6	0.7	0.6
$\epsilon_{SLT}/\epsilon_{\text{VAE}}$	$1.1 \cdot 10^5$	$1.8 \cdot 10^2$	$9.0 \cdot 10^2$	$1.7 \cdot 10^3$	$5.8 \cdot 10^2$
$\text{SBR}_{\text{VAE}}/\text{SBR}_{\text{SLT}}$	-	625	125	70	191

particle multiplicities, in line with the kind of early-discovery new physics scenarios considered at that time. The events populating the hot-line stream were immediately processed at the CERN computing center (as opposed to traditional physics streams, that are processed after 48 hours). The hot-line algorithms were tuned to collect $O(10)$ events per day, which were then visually inspected by experts.

While the focus of the work presented in this paper is not an early discovery, the spirit of the application we propose would be similar: a set of VAEs deployed online would select a limited number of events every day. These events would be collected in a dedicated dataset and further analyzed. The analysis technique could go from visual inspection of the collisions to detailed studies of reconstructed objects, up to some kind of model-independent analysis of the collected dataset, e.g. a deep-learning implementation of a model-independent hypothesis testing [255] directly on the loss distribution (provided a reliable sample of background-only data).

While a pure SM sample to train VAEs could only be obtained from a MC simulation, the presence of outlier contamination in the training sample has typically a tiny impact on performance. One could then imagine to train the VAE models on so-far collected data and use them on the events entering the HLT system. Such a training could happen offline on a dedicated dataset, e.g., deploying triggers randomly selecting events entering the last stage of the trigger system. The training could even happen online, assuming the availability of sufficient computing resources. As it happens with normal triggers, at the very beginning one would use some MC sample or some control sample from previously collected data to estimate the threshold corresponding to the target SM rate. Then, as it happens normally during HLT operations, the threshold will have to be monitored on real data and adjusted if needed.

To demonstrate the feasibility of a train-on-data strategy, we enrich the dataset used in Section 8.4 with a signal contamination of $A \to 4\ell$ events. As a starting point, the amount of injected signal is tuned to a luminosity of 100 pb^{-1} and a cross section of 7.1 pb, corresponding to the value at which the VAE in Section 8.4 would select 100 $A \to 4\ell$ events in one month. This results into about 700 $A \to 4\ell$ events added to the training sample. The VAE is trained following the procedure outlined in Section 8.4 and its performance is compared to that obtained on a signal-free dataset of the same size. The comparison of the ROC curves for the two models is shown in Fig. 8.11. In the same figure, we show similar results, derived injecting a $\times 10$ and $\times 100$ signal contamination. A performance degradation is observed once the signal cross section is set to 710 pb (i.e., 100 times larger than the sensitivity value found in Section 8.4). At that point, the contamination is so large that the signal becomes as abundant as $t\bar{t}$ events and would have easily detectable consequences. For comparison, at a production cross section of 27 pb a third of the events selected by the VAE in Section 8.4 would come from $A \to 4\ell$ production (see Table 8.4). Such a large yield would still have negligible consequences on the training quality. This test shows that a robust anomaly-detecting VAE could be trained directly on data, even in presence of previously undetected (e.g., at Tevatron, 7 TeV and 8-TeV LHC) BSM signals.

The possibility of training the VAE on data would substantially simplify the implementation of the strategy proposed in this work, since any possible systematic bias in the data would be automatically taken into account during the training process. In addition, it would make the procedure robust against other systematic effects (e.g., energy scale, efficiency, etc.) that would affect a MC-based training.

8.7 Deployment in Level-1 triggers

While the primary focus of anomaly detection in this chapter is on the HLT, this strategy would be more effective if deployed in the Level-1 trigger (L1T), i.e. before any selection bias is introduced. For instance, the CMS L1T reduces the input event rate from 40 MHz to 100 kHz. Each event has to be processed within a few microseconds. Due to the extreme latency and resource constraints of the L1T, only relatively simple, theory-motivated selection algorithms are deployed. These usually include requirements on the minimum energy of a physics object, such as a reconstructed lepton or a jet, effectively excluding lower-energy events from further processing. Instead, by deploying an unbiased algorithm which selects events based on their degree of abnormality, rather than on the amount of energy present in the

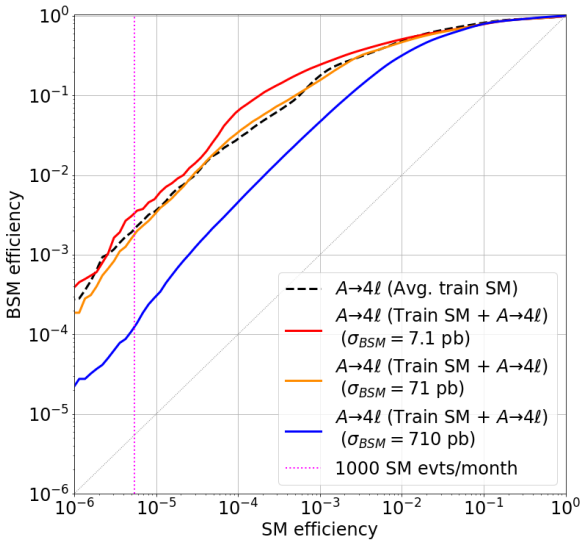


Figure 8.11: ROC curves for the VAE trained on SM contaminated with and without $A \to 4\mu$ contamination. Different levels of contamination are reported corresponding to 0.02% (σ = 7.15 pb - equal to the estimated one to have 100 events per month), 0.19% (σ = 71.5 pb) and 1.89% (σ = 715 pb) of the training sample.

event, we can collect data in a signal-model-independent way. Such an anomaly detection (AD) algorithm is required to have extremely low latency because of the restrictions imposed by the L1T.

Recent developments of the `hls4ml` library allow us to consider the possibility of deploying an AD algorithm on the FPGAs mounted on the L1T boards. The `hls4ml` library is an open-source software, developed to translate neural networks [276–280] and boosted decision trees [281] into FPGA firmware. A fully on-chip implementation of the machine learning model is used in order to stay within the 1 μs latency budget imposed by a typical L1T system. Additionally, the initiation interval of the algorithm should be within 150 ns, which is six times the bunch-crossing time at LHC for the upcoming period of the LHC operations. Since there are several L1T algorithms deployed per FPGA, each of them should take much less than the full FPGA resources. With its interface to QKeras [282], `hls4ml` supports quantization-aware training (QAT) [283], which makes it possible to drastically reduce the FPGA resource consumption while preserving accuracy. Using `hls4ml` we can compress neural networks to fit the limited resources of an FPGA.

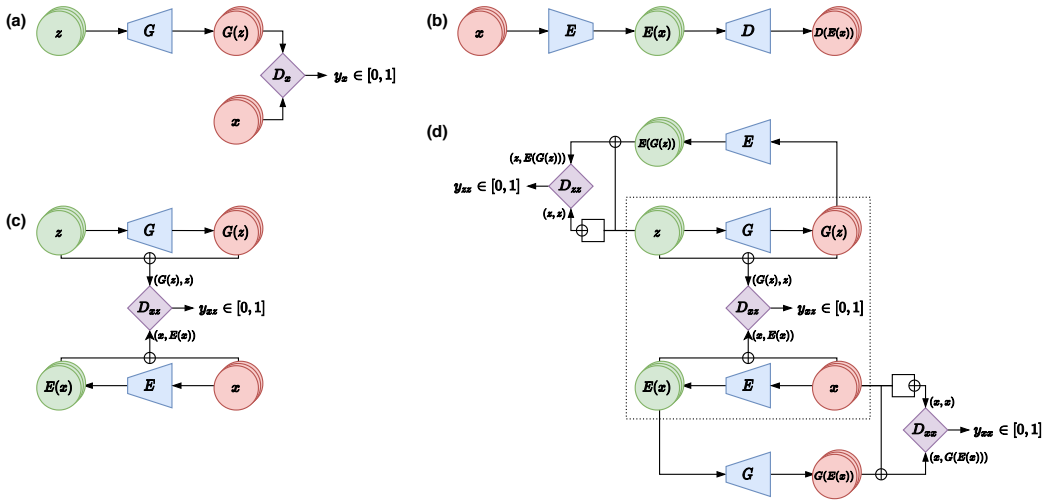


Figure 8.12: Comparison of Deep Network architectures: (a) In a GAN, a generator G returns samples $G(z)$ from latent-space points z, while a discriminator D_x tries to distinguish the generated samples $G(z)$ from the real samples x. (b) In an autoencoder, the encoder E compresses the input x to a latent-space point z, while the decoder D provides an estimate $D(z) = D(E(x))$ of x. (c) A BiGAN is built by adding to a GAN an encoder to learn the z representation of the true x, and using the information both in the real space $\mathcal{X}$ and the latent space $\mathcal{Z}$ as input to the discriminator. (d) The ALAD model is a BiGAN in which two additional discriminators help converging to a solution which fulfils the cycle-consistency conditions $G(E(x)) \approx x$ and $E(G(z)) \approx z$. The $\oplus$ symbol in the figure represents a vector concatenation.

8.8 An alternative model

As an alternative to the VAE, we present another anomaly detection technique using an Adversarially Learned Anomaly Detection (ALAD) algorithm [284], which combines the strength of generative adversarial networks with that of autoencoders.

The ALAD algorithm is based on Generative Adversarial Network (GAN) [285] specifically designed for anomaly detection. The basic idea underlying GANs is that two artificial neural networks compete against each other during training, as shown in Fig. 8.12. One network, the generator $G : \mathcal{Z} \to \mathcal{X}$, learns to generate new samples in the data space (e.g., proton-proton collisions in our case) aiming to resemble the samples in the training set. The other network, the discriminator $D_x : \mathcal{X} \to [0, 1]$, tries to distinguish real samples from generated ones, returning the score of a given sample to be real, as opposed of being generated by G. Both G and D_x are expressed as neural networks, which are trained against each other in a

saddle-point problem:

$$\min_{G}\max_{D_x}\mathbb{E}_{x\sim p_{\mathcal{X}}}[\log\ D_x(x)] + \mathbb{E}_{z\sim p_{\mathcal{Z}}}[\log\,(1 - D_x(G(z))]\ , \tag{8.17}$$

where $p_{\mathcal{X}}(x)$ is the distribution over the data space $\mathcal{X}$ and $p_{\mathcal{Z}}(z)$ is the distribution over the latent space $\mathcal{Z}$. The solution to this problem will have the property $p_{\mathcal{X}} = p_G$, where p_G is the distribution induced by the generator [285]. The training typically involves alternating gradient descent on the parameters of G and D_x to maximize for D_x (treating G as fixed) and to minimize for G (treating D_x as fixed).

Deep learning for anomaly detection [286] is usually discussed in the context of (variational) autoencoders [251, 252]. With autoencoders (cf. Fig. 8.12), one projects the input x to a point z of a latent-space through an encoder network $E : \mathcal{X} \to \mathcal{Z}$. An approximation $D(z) = D(E(x))$ of the input information is then reconstructed through the decoder network, $D : \mathcal{Z} \to \mathcal{X}$. The intuition is that the decoder D can only reconstruct the input from the latent space representation z if $x \sim p_{\mathcal{X}}$. Therefore, the reconstruction for an anomalous sample, which belongs to a different distribution, would typically have a higher reconstruction loss. One can then use a metric D_{R} defining the output-to-input distance (e.g., the one used in the reconstruction loss function) to derive an anomaly-score A:

$$A(x) \sim \mathrm{D}_{\mathrm{R}}(x, D(E(x))). \tag{8.18}$$

While this is not directly possible with GANs, since a generated $G(z)$ does not correspond to a specific x, several GAN-based solutions have been proposed that would be suitable for anomaly detection, as for instance in Refs. [284, 286–289].

In this work, we focus on the ALAD method [284], built upon the use of bidirectional-GANs (BiGAN) [290]. As shown in Fig. 8.12, a BiGAN model adds an encoder $E : \mathcal{X} \to \mathcal{Z}$ to the GAN construction. This encoder is trained simultaneously to the generator. The saddle point problem in Eq. 8.17 is then extended as follows:

$$\begin{aligned}\min_{G,E}\max_{D_{xz}} V(D_{xz}, E, G) = \min_{G,E}\max_{D_{xz}} &\mathbb{E}_{x\sim p_{\mathcal{X}}}[\log\ D_{xz}(x, E(x))] \\ &+ \mathbb{E}_{z\sim p_{\mathcal{Z}}}[\log\,(1 - D_{xz}(G(z), z)],\end{aligned} \tag{8.19}$$

where D_{xz} is a modified discriminator, taking inputs from both the $\mathcal{X}$ and $\mathcal{Z}$. Provided there is convergence to the global minimum, the solution has the distribution matching property $p_E(x, z) = p_G(x, z)$, where one defines $p_E(x, z) = p_E(z|x)p_{\mathcal{X}}(x)$ and $p_G(x, z) = p_G(x|z)p_{\mathcal{Z}}(z)$ [290]. To help reaching full convergence, the ALAD model is equipped with two additional discriminators: D_{xx} and

D_{zz}. The former discriminator together with the value function

$$V(D_{xx}, E, G) = \mathbb{E}_{x\sim p_X}[\log\, D_{xx}(x,x)] + \mathbb{E}_{x\sim p_X}[\log\,(1 - D_{xx}(x, G(E(x))))] \quad (8.20)$$

enforces the cycle-consistency condition $G(E(x)) \approx x$. The latter is added to further regularize the latent space through a similar value function:

$$V(D_{zz}, E, G) = \mathbb{E}_{z\sim p_Z}[\log\, D_{zz}(z,z)] + \mathbb{E}_{z\sim p_Z}[\log\,(1 - D_{zz}(z, E(G(z))))], \quad (8.21)$$

enforcing the cycle condition $E(G(z)) \approx z$. The ALAD training objective consists in solving:

$$\min_{G,E}\max_{D_{xz},D_{xx},D_{zz}} V(D_{xz}, E, G) + V(D_{xx}, E, G) + V(D_{zz}, E, G)\ . \quad (8.22)$$

Having multiple outputs at hand, one can associate the ALAD algorithm to several anomaly-score definitions. Following Ref. [284], we consider the following four anomaly scores:

- A "logit" score, defined as: $A_L(x) = \log(D_{xx}(x, G(E(x)))$.
- A "feature" score, defined as: $A_F(x) = ||f_{xx}(x,x) - f_{xx}(x, G(E(x)))||_1$, where $f_{xx}(\cdot,\cdot)$ are the activation values in the last hidden layer of D_{xx}.
- The L_1 distance between an input x and its reconstructed output $G(E(x))$: $A_{L_1}(x) = ||x - G(E(x))||_1$.
- The L_2 distance between an input x and its reconstructed output $G(E(x))$: $A_{L_2}(x) = ||x - G(E(x))||_2$.

We train our ALAD model on the SM cocktail and subsequently apply it to a test dataset, containing a mixture of SM events and events of physics beyond the Standard Model (BSM). As a starting point, we consider the ALAD architecture [284] used for the `KDD99` dataset, which has similar dimensionality as our input feature vector. In this configuration, both the D_{xx} and D_{zz} discriminators take as input the concatenation of the two input vectors, which is processed by the network up to the single output node, activated by a sigmoid function. The D_{xz} discriminator has one dense layer for each of the two inputs. The two intermediate representations are concatenated and passed to another dense layer and then to a single output node with sigmoid activation, as for the other discriminators. The hidden nodes of the

generator are activated by ReLU functions [209], while Leaky ReLU [291] are used for all the other nodes. The slope parameter of the Leaky ReLU function is fixed to 0.2. The network is optimized using the Adam [212] minimizer and minibatches of 50 events each. The training is regularized using dropout layers in the three discriminators.

Starting from this baseline architecture, we adjust the architecture hyperparameters one by one, repeating the training while maximizing a figure of merit for anomaly detection efficiency. We perform this exercise using as anomalies the benchmark models described in previously in this chapter and looking for a configuration that performs well on all of them. To quantify performance, we consider both the area under the receiver operating characteristic (ROC) curve and the positive likelihood ratio LR_+. We define the LR_+ as the ratio between the BSM signal efficiency, i.e., the true positive rate (TPR), and the SM background efficiency, i.e., the false positive rate (FPR). The training is performed on half of the available SM events (3.4M events), leaving the other half of the SM events and the BSM samples for validation. From the resulting anomaly scores, we compute the ROC curve and compare it to the results of the VAE. We further quantify the algorithm performance considering the LR_+ values corresponding to an FPR of 10^{-5}.

The optimized architecture, adapted from Ref. [284], is summarized in Table 8.6. This architecture is used for all subsequent studies. We consider as hyperparameters the number of hidden layers in the five networks, the number of nodes in each hidden layer, and the dimensionality of the latent space, represented in the table by the size of the E output layer.

Having trained the ALAD on the training dataset, we compute the anomaly scores for the validation samples as well as for the four BSM samples, where each BSM process has $O(0.5\mathrm{M})$ samples. Figure 8.13 shows the ROC curves of each BSM benchmark process, for the four considered anomaly scores. The best VAE result is also shown for comparison. In the rest of this paper, we use the L_1 score as the anomaly score. Similar results would have been obtained using any of the other three anomaly scores. Figure 8.14 compares the A_{L_1} distribution for each BSM process with the SM cocktail. One can clearly see that all BSM processes have an increased probability in the high-score regime compared to the SM cocktail. We further verified that the anomaly score distributions obtained on the SM-cocktail training and validation sets are consistent. This test excludes the occurrence of over-training issues.

Table 8.6: Hyperparameters for the ALAD algorithm. Parameters in bold have been optimized for. No Dropout layer is applied wherever a dropout rate is not specified.

Operation	Units	Activation	Batch Norm.	Dropout Rate
$\boldsymbol{E(x)}$				
Number of hidden layers	**2**			
Dense	**64**	Leaky ReLU	×	-
Dense	**64**	Leaky ReLU	×	-
Output	**16**	Linear	×	-
$\boldsymbol{G(z)}$				
Number of hidden layers	**2**			
Dense	**64**	ReLU	×	-
Dense	**64**	ReLU	×	-
Output	39	Linear	×	-
$\boldsymbol{D_{xz}(x,z)}$				
Number of hidden layers	**2**			
Only on x				
Dense	**128**	Leaky ReLU	√	-
Only on z				
Dense	**128**	Leaky ReLU	×	0.5
Concatenate outputs				
Dense	**128**	Leaky ReLU	×	0.5
Output	1	Sigmoid	×	-
$\boldsymbol{D_{xx}(x,\hat{x})}$				
Concatenate x and x'				
Number of hidden layers	**1**			
Dense	**128**	Leaky ReLU	×	0.2
Output	1	Sigmoid	×	-
$\boldsymbol{D_{zz}(z,\hat{z})}$				
Concatenate z and z'				
Number of hidden layers	**1**			
Dense	**128**	Leaky ReLU	×	0.2
Output	1	Sigmoid	×	-

Training Parameter	Value
Optimizer	Adam ($\alpha = 10^{-5}$, $\beta_1 = 0.5$)
Batch size	50
Leaky ReLU slope	0.2
Spectral norm	√
Weight, bias init.	Xavier Initializer, Constant(0)

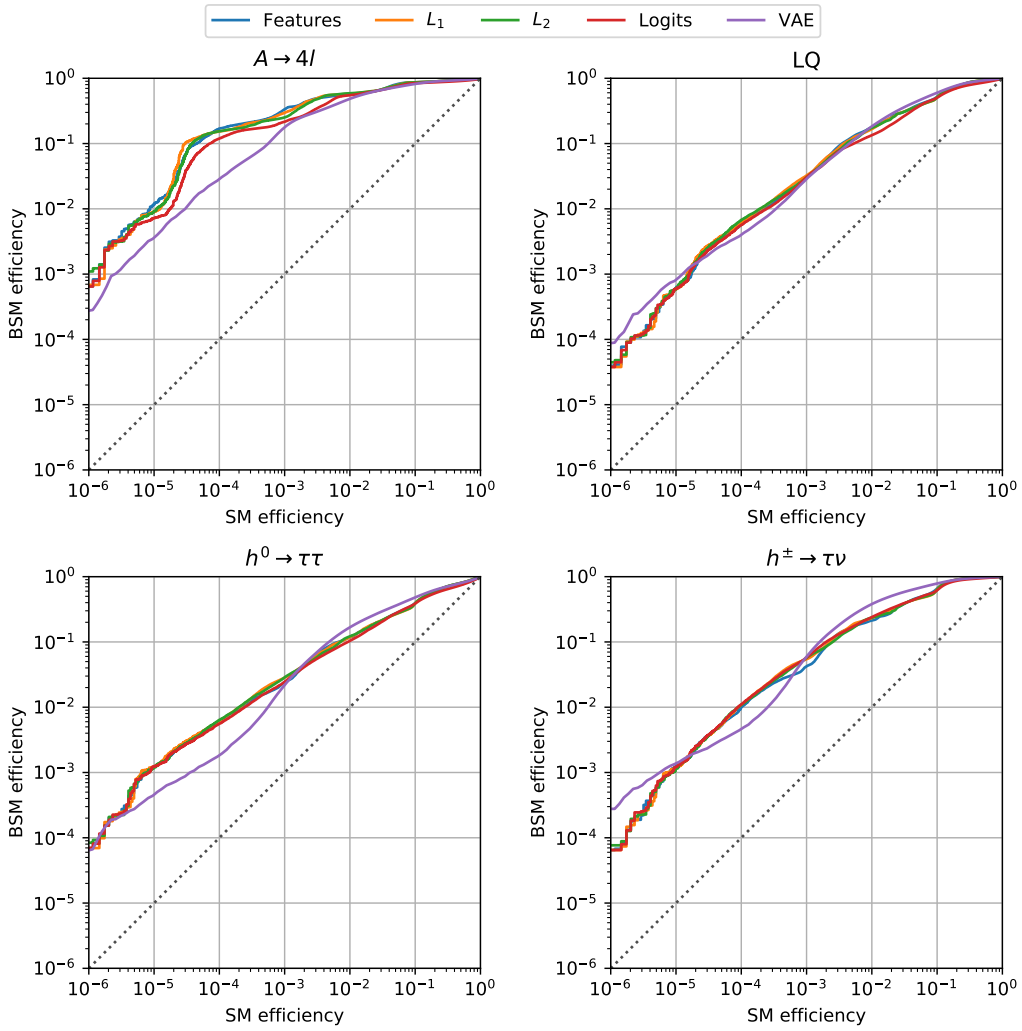


Figure 8.13: ROC curves for the ALAD trained on the SM cocktail training set and applied to SM+BSM validation samples. The VAE curve corresponds to the best result of the VAE, which is shown here for comparison. The other four lines correspond to the different anomaly score models of the ALAD.

The ALAD algorithm outperforms the VAE by a substantial margin on the $A \to 4\ell$ sample, providing similar performance overall, and in particular for FPR $\sim 10^{-5}$, the same working point chosen for the VAE. We verified that the uncertainty on the TPR at fixed FPR, computed with the AgrestiCoull interval [292], is negligible when compared to the observed differences between ALAD and VAE ROC curves, i.e., the difference is statistically significant.

The left plot in Fig. 8.15 provides a comparison across different BSM models. As for the VAE, ALAD performs better on $A \to 4\ell$ and $h^\pm \to \tau\nu$ than for the other two BSM processes. The right plot in Figure 8.15 shows the LR_+ values as a function

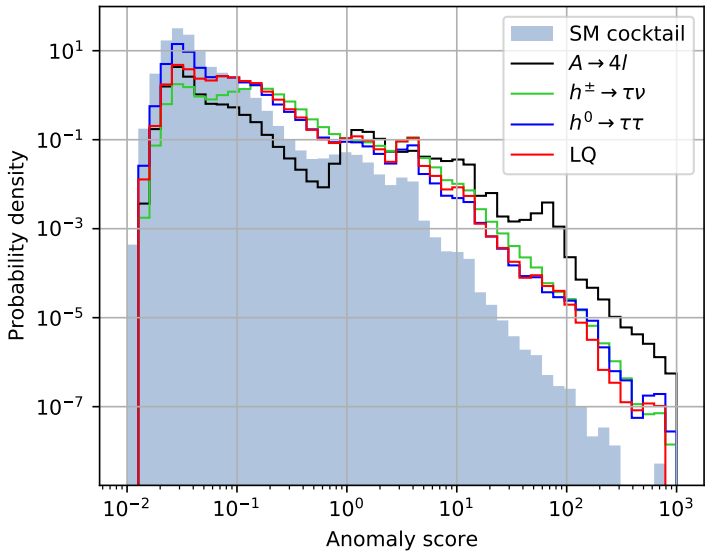


Figure 8.14: Distribution for the A_{L_1} anomaly score. The "SM cocktail" histogram corresponds to the anomaly score for the validation sample. The other four distributions refer to the scores of the four BSM benchmark models.

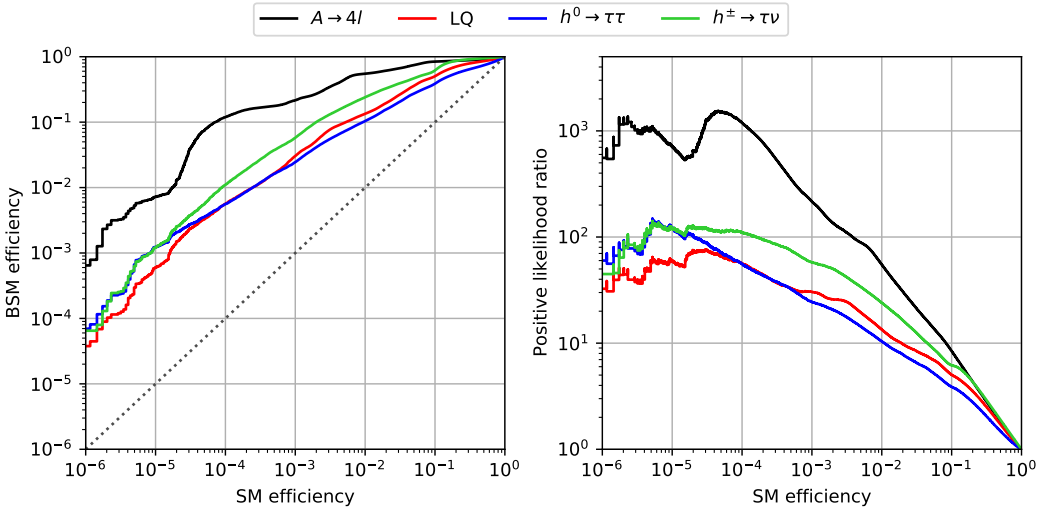


Figure 8.15: Left: ROC curves for each BSM process obtained with the ALAD L_1-score model. Right: LR_+ curves corresponding to the ROC curves on the left.

of the FPR ones. The LR_+ peaks at a SM efficiency of $O(10^{-5})$ for all four BSM processes and is basically constant for smaller SM-efficiency values.

8.9 Summary

We present a strategy to isolate potential BSM events produced by the LHC, using variational autoencoders trained on a reference SM sample. Such an algorithm

could be used in the trigger system of general-purpose LHC experiments to identify recurrent anomalies, which might otherwise escape observation (e.g., being filtered out by a typical trigger selection). Taking as an example a single-lepton data stream, we show how such an algorithm could select datasets enriched with events originating from challenging BSM models. We also discuss how the algorithm could be trained directly on data, with no sizable performance loss, more robustness against systematic uncertainties, and a big simplification of the training and deployment procedure.

The main purpose of such an application is not to enhance the signal selection efficiency for BSM models. Indeed, this application is tuned to provide a high-purity sample of potentially interesting events. We showed that events produced by not-yet-excluded BSM models with cross sections in the range of $O(10)$ to $O(100)$ pb could be isolated in a $\sim 30\%$ pure sample of ~ 43 events selected per day. The price to pay to reach such a purity is a relatively small signal efficiency and a strong bias in the dataset definition, which makes these events marginal and difficult to use in a traditional data-driven and supervised search for new physics.

The final outcome of this application would be a list of anomalous events, that the experimental collaborations could further scrutinize and even release as a catalog, similarly to what is typically done in other scientific domains. Repeated patterns in these events could motivate new scenarios for beyond-the-standard-model physics and inspire new searches, to be performed on future data with traditional supervised approaches.

We stress the fact that the power of the proposed approach is in its generality and not in its sensitivity to a particular BSM scenario. We show that a simple BDT could give a better discrimination capability for a given BSM hypothesis. On the other hand, such a supervised algorithm would not generalize to other BSM scenarios. The VAE, instead, comes with little model dependence and therefore generalizes to unforeseen BSM models. On the other hand, the VAE cannot guarantee an optimal performance in any scenario. As typical of autoencoders used for anomaly detection, our VAE model is trained to learn the SM background at best, but there is no guarantee that the best SM-learning model will be the best anomaly detection algorithm. By definition, the anomaly detection capability of the algorithm does not enter the loss function, as well as, by construction, no signal event enters the training sample. This is the price to pay when trading discrimination power for model independence.

We believe that such an application could help extending the physics reach of the current and next stages of the CERN LHC. The proposed strategy is demonstrated for a single-lepton data stream coming from a typical L1 selection. On the other hand, this approach could be generalized to any other data stream coming from any L1 selection, so that the full $\sim$ 100 Hz rate entering the HLT system of ATLAS or CMS could be scrutinized. While the L1 selection still represents a potentially dangerous bias, an algorithm running in the HLT could access 100 times more events than the $\sim$ 1 kHz stream typically available for offline studies. Moreover, thanks to progresses in the deployment of deep neural networks on FPGA boards [276], it is conceivable that VAEs for anomaly detection could be also deployed in the L1 trigger systems in a near future. In this way, the VAE would access the full L1 input data stream.

Part V

Machine-learning solutions for the High-Luminosity LHC era

Chapter 9

THE PHASE-II UPGRADE

After major successes with the LHC since its commission in 2010, most notably the discovery of the Higgs boson [242, 243], which opened up exciting opportunities to better understand the Standard Model and discover new physics, the LHC will continue to lead the energy frontier for at least the next two decades. To fully exploit the physics potential of the LHC, the high-luminosity LHC project (HL-LHC) aims to increase the peak luminosity by 5 folds with levelling operation, with the goal of 3000 fb^{-1} collected by 2040 [293]. Near the end of 2024, the LHC will enter the third long-shutdown period (LS3) to prepare for this Phase-II upgrade, as illustrated by the timeline in Fig. 9.1.

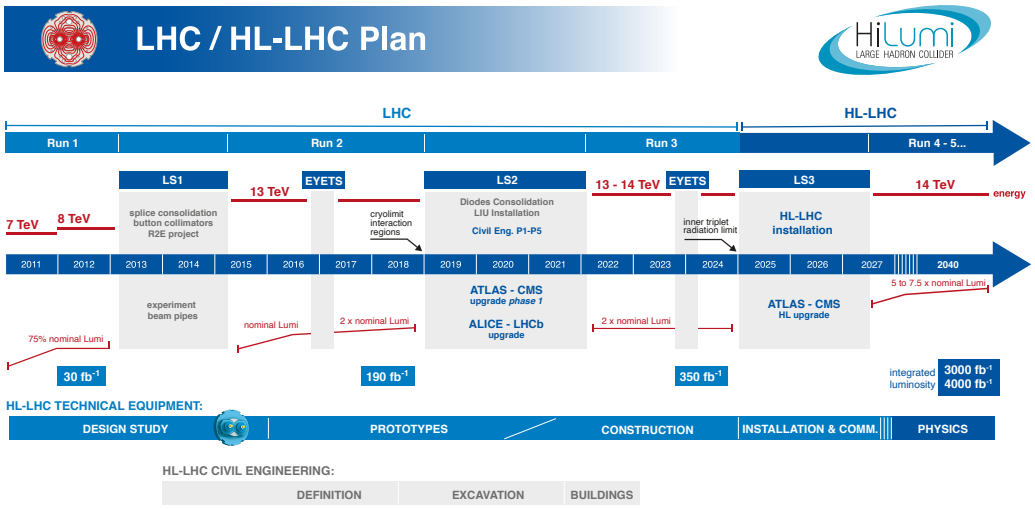


Figure 9.1: The baseline plan of the LHC for the next decade and beyond. After LS3 with the installation of the HL-LHC and the high luminosity upgrade for CMS, the machine will start collect data near the end of 2027 with the target peak luminosity of 5×10^{34} $\text{cm}^{-2}\text{s}^{-1}$ and an integrated luminosity of 250 fb^{-1} per year, with the goal of 3000 fb^{-1} by 2040 [293].

The substantial luminosity of HL-LHC will make it possible to study the Higgs sector through precision measurements and observations of rare processes involving the Higgs boson. Most remarkably, the target luminosity of 3000 fb^{-1} by 2040 can potentially allow us, for the first time, to observe the double Higgs production.

9.1 The challenges of CMS computing in the HL-LHC era

With the drastic increase of peak luminosity, the HL-LHC poses serious challenges to the computational resources, which cannot scale linearly along with the resources needed due to limited budget. In particular, both ATLAS and CMS experiments are expected to increase their trigger selection rate from 1 kHz to at least 7.5 kHz [73]. Given the 7x from the selection rate and the 5x from luminosity scaling, there is a drastic gap between the projected resources needed and actual available resources, since no adiabatic technology improvement can realistically provide an increase of over 30 times for the LHC computing capacity in the next 7 years. Additionally, with new detector installed, such as High-Granularity Calorimeter (HGCAL) [294] and the new pixel detector [295] for CMS with many more data acquisition channels, the raw event size can increase 7 folds, adding pressure to the storage and downstream processing. Fig. 9.2 illustrates the projected gap between resources needed versus availability for the CPU time and disk space by the CMS experiment [296]. Two scenarios are considered. The first scenario, historically considered by CMS, assumes an integrated luminosity of 275 fb^{-1} per year during Run 4, with the HLT rate of 7.5 kHz. The second scenario is more commonly considered in agreement with the Worldwide LHC Computing Grid group (WLCG) and ATLAS, where the integrated luminosity for each year during Run 4 is assumed to be 500 fb^{-1}, with the HLT rate of 10 kHz. Under both scenarios, there are large gaps between the requirements and the projected available resources, which are assumed to increase between 10% and 20% each year.

9.2 Heterogeneous platform for the future of CMS computing

Over the past few years, CMS has evolved the computing model to address these challenges. While traditional CPUs are still powerful processors for data centers, the slow down of Moore's law, along with the rise of hardware accelerators, makes heterogeneous platform an attractive option for WLCG sites and high-performance computing (HPC) centers.

Heterogeneous computing platform contains more than one kind of processor. Typically, it combines the traditional CPUs with one or more types of hardware accelerators, such as the General-Purpose Graphics Processing Units (GP-GPUs), Field-Programmable Gate Arrays (FPGAs), or Tensor Processing Units (TPUs). Significant efforts in CMS have been made to adopt heterogeneous architectures for running CMS offline software (CMSSW). During Run 3, a heterogeneous HLT farm will be deployed by CMS with the acceleration from NVIDIA GPUs along with

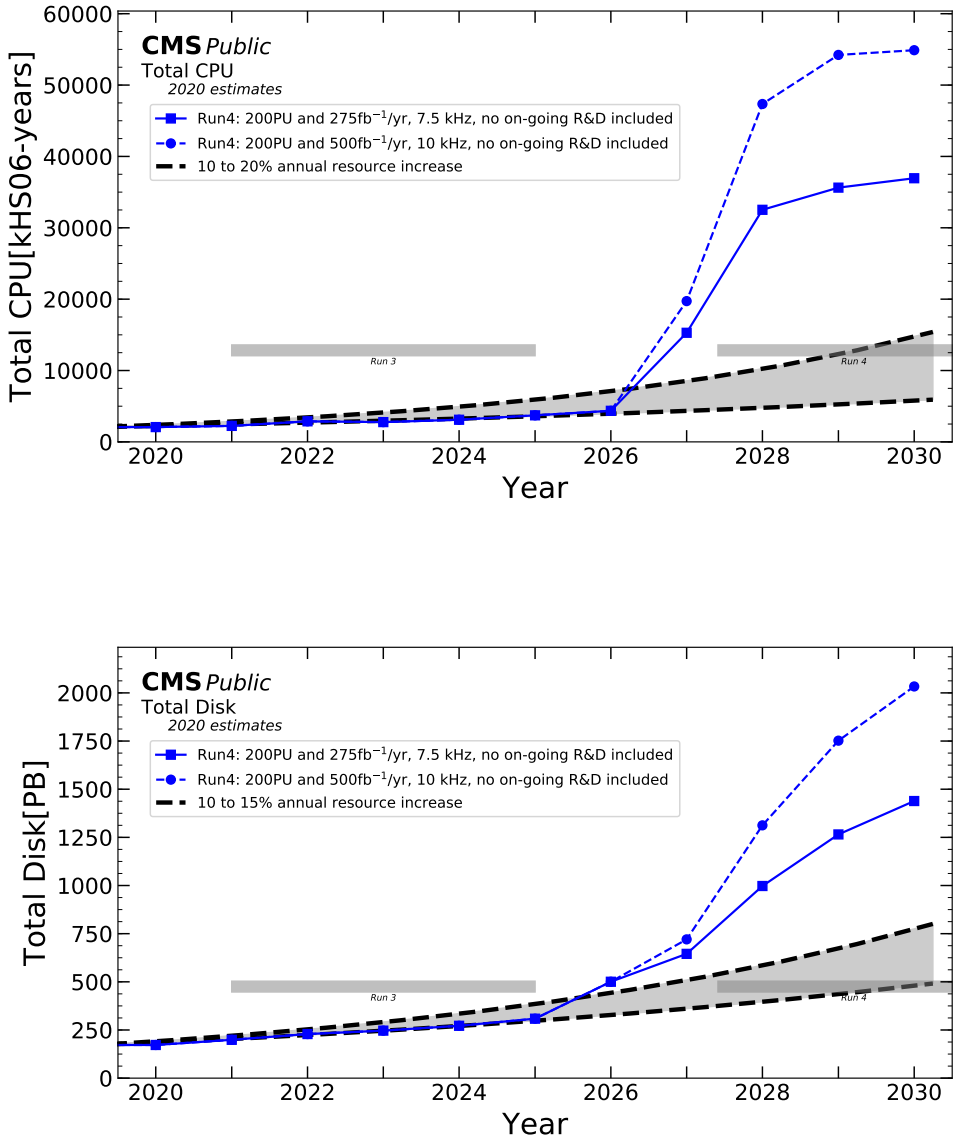


Figure 9.2: CMS experiment's projection for the CPU time (top) and disk space (bottom) requirements needed annually for CMS processing and analyses versus resource availability under 2 scenarios: (1) a running scenario of 275 fb^{-1} per year during Run 4, with a trigger rate of 7.5 kHz, as shown by the solid blue line; (2) a running scenario of 500 fb^{-1} per year during Run 4, with a trigger rate of 10 kHz, as shown by dashed blue line. The projected resources needed are summed across Tier-0, Tier-1, and Tier-2 resources. The black curves show the projected resources availability assuming an annual increase between 10% and 20% [296].

traditional CPUs [297]. A significant portion of the track reconstruction algorithms during HLT has been re-engineered in CMSSW to run on accelerators with CUDA, showing substantial improvements with respect to traditional algorithms running on CPUs [298].

9.3 Machine-learning solutions for the HL-LHC era

Even though the heterogeneous computing platform originally aims to re-engineer the classical algorithms to boost their performances on hardware accelerators, the availability of GP-GPUs and FPGAs open up numerous opportunities for the deployments of machine-learning algorithms at the LHC to boost the computing efficiency as well as to extend the reach of the New Physics search program.

The deep learning era in the past decade has flourished since the realization that GP-GPU, traditionally designed for graphics processing, could be used for deep learning algorithms, which essentially consist of matrix operations. Backpropagation [164, 299] and stochastic gradient descent [300] proved to be a powerful combination for optimizing a learning model with even more than a trillion parameters.

In high energy physics, the advantages of deep learning algorithms are threefold. First, they have the flexibility to incorporate multiple complicated data structures, which are ubiquitous in high energy physics. For example, as shown in Ch. 6, input data can be represented as graphs, where the elements are permutation invariant and the relations between elements can be learned. Furthermore, multiple data structures can go simultaneously into a neural network (also known as a multimodal neural network), such as sequential data and tabular data, as shown by the $t\bar{t}H$ discriminator in Ch. 5. Second, due to their large number of parameters, supervised deep learning algorithms typically outperform other traditional classification algorithms in the big data regime, which is the case in most high energy physics applications. Third, by the universal approximation theorem [301], a deep learning algorithm can approximate any function in high dimensions. It is therefore possible to substitute traditionally computationally expensive tasks, such as event simulation, with deep learning models. Generative models, such as generative adversarial networks and autoencoders, are particularly useful for these tasks.

This part of the thesis will discuss a series of machine-learning solutions to address the aforementioned challenges in the HL-LHC era. In particular, Ch. 10 introduces a new module using multimodal neural networks with supervised learning to clean up the event stream after the trigger selection, reducing the amount of downstream

resources wasted in processing false positive events. Ch. 11 proposes a new approach to alternate full event simulation with generative adversarial networks. Ch. 12 targets analysis-specific simulation, where a deep neural networks can be used to replace the simulation and reconstruction step for Monte Carlo simulated events.

Chapter 10

TRIGGER IMPROVEMENTS WITH TOPOLOGY CLASSIFIER

We show how an event topology classification based on deep learning could be used to improve the purity of data samples selected in real time at the Large Hadron Collider. We consider different data representations, on which different kinds of multi-class classifiers are trained. Both raw data and high-level features are utilized. In the considered examples, a filter based on the classifier's score can be trained to retain $\sim 99\%$ of the interesting events and reduce the false-positive rate by more than one order of magnitude. By operating such a filter as part of the online event selection infrastructure of the LHC experiments, one could benefit from a more flexible and inclusive selection strategy while reducing the amount of downstream resources wasted in processing false positives. The saved resources could translate into a reduction of the detector operation cost or into an effective increase of storage and processing capabilities, which could be reinvested to extend the physics reach of the LHC experiments.

10.1 Introduction

The CERN Large Hadron Collider (LHC) collides protons every 25 ns. Each collision can result in any of hundreds of physics processes. The total data volume exceeds by far what the experiments could record. This is why the incoming data flow is typically filtered through a set of rule-based algorithms, designed to retain only events with particular signatures (e.g., the presence of a high-energy particle of some kind). Such a system, commonly referred to as *trigger*, consists of hundreds of algorithms, each designed to accept events with a specific topology. The ATLAS [249] and CMS [56] trigger systems are based on this idea. In their current implementation, given the throughput capability and the typical event size, these two experiments can write on disk ~ 1000 events/sec. A few processes, e.g., QCD multijet production, constitute the vast majority of the produced events. One is typically interested to select a fraction of these events for further studies. On the other hand, the main interest of the LHC experiments is related to selecting and studying the many rare processes which occur at the LHC. In a typical data flow, these events are overwhelmed by the large amount of QCD multijet events. The

trigger system is put in place to make sure that the majority of these rare events are part of the stored $\sim$ 1000 events/sec.

Trigger algorithms are typically designed to maximize the efficiency (i.e., the true-positive rate), resulting in a non-negligible false-positive rate and, consequently, in a substantial waste of resources at trigger level (i.e., data throughput that could have been used for other purposes) and downstream (i.e., storage disk, processing power, etc.).

The most commonly used selection rules are *inclusive*, i.e., more than one topology is selected by the same requirement. The so-called isolated lepton triggers are a typical example of this kind of algorithms. These triggers select events with a high-momentum electron or muon and no surrounding energetic particle, a typical signature of an interesting rare process, e.g., the production of a W boson decaying to a neutrino and an electron or muon. With such a requirement, one can simultaneously collect W bosons produced in the primary interaction (W events) or from the cascade decay of other particles, e.g., top quarks (mainly in $t\bar{t}$ events where a top quark-antiquark pair is produced). A sample selected this way is dominated by W events but it retains a substantial ($> 10\%$) contamination from QCD multijet. The $t\bar{t}$ contribution is smaller than 1%. Events from $t\bar{t}$ production are sometimes triggered by a set of dedicated lepton+jets algorithms, capable of using looser requirements on the lepton at the cost of introducing requirements on jets.[1] Due to this additional complexity, the use of these triggers in a data analysis comes with additional complications. For instance, the applied jet requirements produce distortions on offline distributions of jet-related quantities. To avoid having this effect, any typical data analysis applies a tighter offline selection. This means that many of the selected events close to the online-selection threshold are discarded. This is not necessarily the most cost-effective way to retain an unbiased dataset for offline analysis.

In this chapter, we investigate the possibility of using machine learning to classify events based on their topologies, serving as an additional clean-up algorithm at the trigger level. Doing so, one could customize the trigger-selection strategy on individual processes (depending on the physics goals) while keeping the selection loose and simple. As a benchmark case, we consider a stream of data selected by requiring the presence of one electron or muon with transverse momentum

[1] A jet is a spray of hadrons, typically originating from the hadronization of gluons and quarks produced in the proton collisions.

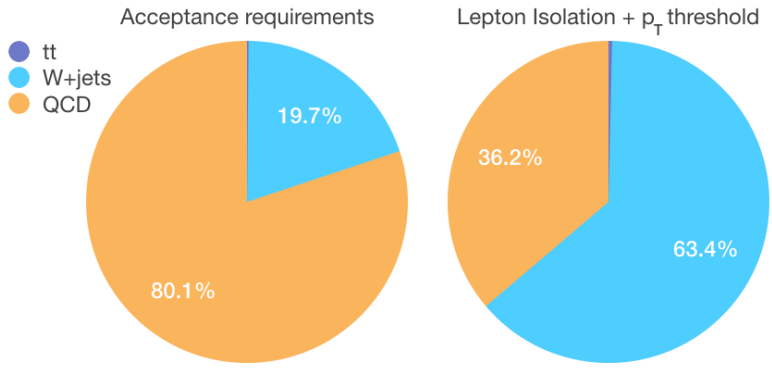


Figure 10.1: Relative composition of the isolated-lepton sample after the acceptance requirement (left) and the trigger selection (right), as described in the text.

$p_T > 23$ GeV [2] and a loose requirement on the isolation. Details on the applied selection can be found in Sec. 10.2.

The considered benchmark sample is dominated by direct W production, with a sizable contamination from QCD multijet events and a small contribution of $t\bar{t}$ events. Other interesting processes (e.g., WW, WZ, and ZZ production) are usually selected with more exclusive and dedicated trigger algorithms (e.g., di-muon or di-electron triggers), or share the same kinematic properties of the two main interesting processes (W and $t\bar{t}$). For the sake of simplicity, we ignore these sub-leading processes in our study, without compromising the validity of our conclusions. Fig. 10.1 shows the composition of a sample with one electron or muon within the defined acceptance ($p_T > 22$ GeV and pseudorapidity $|\eta| = |-\log[\tan(\theta/2)]| < 2.6$, where θ is the polar angle), before and after applying the trigger requirements ($p_T > 23$ GeV and loose isolation).

Such a loose set of requirements would translate into an event acceptance rate of ~ 690 Hz for a luminosity of 2×10^{34} cm^{-2} s^{-1}, well beyond the currently allocated budget for these triggers (typically ~ 200 Hz). We suggest that, using the score of our topology classifier, one could tune the amount of each process to be stored for further analysis, within the boundaries of the allocated resources. For instance,

[2] In this chapter, we set units in such a way that $c = \hbar = 1$.

one might be interested to retain all the $t\bar{t}$ events and some fraction of W events, while rejecting the QCD multijet events. We envision two main applications: for a given total rate, one could loosen the baseline trigger requirements, increasing the acceptance efficiency at no cost. Or, for a given acceptance efficiency (true positive rate), one could save resources by reducing the overall rate, rejecting the contribution of unwanted topologies (see Sec. 10.7).

We consider several topology classifiers based on deep learning model architectures: fully-connected deep neural networks (DNNs), convolutional neural networks (CNNs) [302], and recurrent neural networks such as Long-Short-Term-Memory networks (LSTMs) [166] and gated recurrent units (GRUs) [167]. We consider four different representations of the collision events: (i) a set of physics-motivated high-level features, (ii) the raw image of the detector hits, (iii) a sequence of particles, characterized by a limited set of basic features (energy, direction, etc.), and (iv) an *abstract* representation of this list of particles as an image.

The chapter is structured as follows. In Sec. 10.2 we describe the four data representations. In Sec. 10.3 we describe the corresponding classification models. Results are discussed in Sec. 10.4. In Sec. 10.5 we investigate the generalization properties of the four classifiers to scenarios of other topologies. We study the robustness of our classifiers against Monte-Carlo simulation inaccuracy with pseudo-data in Sec. 10.6. In Sec. 10.8 we briefly discuss applications of machine learning algorithms to similar problems. Conclusions are given in Sec. 10.9. Appendix A describes a different scenario, in which the classifier is used to save resources by reducing the trigger acceptance rate, as opposed to using it to sustain a loose trigger selection that could otherwise require too many resources.

10.2 Dataset

Synthetic data corresponding to W, $t\bar{t}$ and QCD multijet production topologies are generated with 10^5 events per process ($3 \cdot 10^5$ events in total) using the `PYTHIA8` event generation library [103]. The setup of the proton-beam simulation is loosely inspired by the LHC running configuration in 2015-2016: two proton beams, each with 6.5 TeV, generate on average 20 proton-proton collisions per crossing following a Poisson distribution.

Generated samples are processed with the `DELPHES` library [269], which applies a parametric model of a detector response. Detector performances is tuned to the CMS upgrade design foreseen for the High-Luminosity LHC [270], as implemented in the

corresponding default card provided with DELPHES. We run the DELPHES *particle-flow* (PF) algorithm, which combines the information from all the CMS detector components to derive a list of reconstructed particles, the so-called PF candidates. For each particle, the algorithm returns the measured energy and flight direction. Each particle is associated to one of three classes: charged particles, photons, and neutral hadrons. Jets are clustered from the reconstructed PF candidates, using the FASTJET [194] implementation of the anti-k_T jet algorithm [57], with jet-size parameter R = 0.4. The jet's b-tagging efficiency is parametrized as a function of jet's p_T and η in the default DELPHES CMS upgrade design card. The parametrized b-tagging efficiency is shown to provide a reasonable agreement with CMS [269].

The basic event representation consists of a list of reconstructed PF candidates. For each candidate q, the following information is given: (i) The particle four-momentum in Cartesian coordinates (E, p_x, p_y, p_z); (ii) The particle three-momentum, computed from (i), in cylindrical coordinates: the transverse momentum p_T, the pseudorapidity η, and the azimuthal angle ϕ; (iii) The Cartesian coordinates (x_{vtx}, y_{vtx}, z_{vtx}) of the particle point of origin. For all neutral particles, (0, 0, 0) is used in the absence of pointing information; (iv) The electric charge; (v) The particle isolation with respect to charged particles (ChPFIso), photons (GammaPFIso), or neutral hadrons (NeuPFIso). For each particle class, the isolation is quantified as

$$\texttt{ISO} = \frac{\sum_{p \neq q} p_T^p}{p_T^q} \,, \tag{10.1}$$

where the sum extends over all the particles of the appropriate class with angular distance $\Delta R = \sqrt{(\Delta\eta)^2 + (\Delta\phi)^2} < 0.3$ from the particle q.

The particle identity is categorized via a one-hot-encoded representation (isChPar, isNeuHad, isGamma), corresponding to a charged particle, a neutral hadron, or a photon. In addition, two boolean flags are stored (isEle and isMu) to identify if a given particle is an electron or a muon. In total, each particle is then described by 19 features.

The trigger selection is emulated by requiring all the events to include one isolated electron or muon with transverse momentum $p_T > 23$ GeV and particle-based isolation ChISO + GammaISO + NeuISO < 0.45. This baseline selection, which follows the typical requirements of an inclusive single-lepton trigger algorithm, accepts ≈ 100 QCD multijet events and ≈ 176 W events for every $t\bar{t}$ event. Despite its large W and $t\bar{t}$ efficiency, this trigger selection comes with a large cost in terms of

QCD multijet events written on disk and processed offline. The cost is even larger if the main physics target is $t\bar{t}$ events and the W contribution is seen as an additional source of background (e.g., in a high-statistics scenario, with all measurements of W properties limited in precision by systematic uncertainties).

All particles are ranked in decreasing order of p_T. For each event, the isolated lepton is the first entry of the list of particles. To avoid double counting of this isolated lepton ℓ as a charged particle, each charged particle q is required to have $\Delta R(q, \ell) > 10^{-4}$. In addition to the isolated lepton, we consider the first 450 charged particles, the first 150 photons, and the first 200 neutral hadrons. This corresponds to a total of 801 particles per event, each characterized by the 19 features described above. The choice of the numbers of particles is made such that, on average, only 5% charged particles, 5% neutral hadrons and 1% photons are ignored. Thanks to p_T ordering by particle category, what we remove carries small information. In early stages of this work we experimented with tighter cuts on particle multiplicity without observing substantial difference. We verified that the particles we ignore have typical p_T below 1 GeV. If fewer particles are found in the event, zero padding is used to guarantee a fixed length of the particle list across different events. The events are then stored as NumPy arrays in a set of compressed HDF5 files. The dataset is planned to be released on the CERN OpenData portal, accessible at `opendata.cern.ch`.

In addition to this raw-event representation, we provide a list of physics-motivated high-level features, computed from the full event (the `HLF` dataset):

- The scalar sum, S_T, of the p_T of all the jets, leptons, and photons in the event with $p_T > 30$ GeV and $|\eta| < 2.6$.
- The missing transverse energy E_T^{miss}, defined as the absolute value of the missing transverse momentum, computed summing over the full list of reconstructed PF candidates:

$$E_T^{\text{miss}} = \left|\vec{p}_T^{\,\text{miss}}\right| = \left|-\sum_q \vec{p}_T^{\,q}\right| . \tag{10.2}$$

- The squared transverse mass, M_T^2, of the isolated lepton ℓ and the E_T^{miss} system, defined as:

$$M_T^2 = 2 p_T^{\ell} E_T^{\text{miss}} (1 - \cos\Delta\phi) \tag{10.3}$$

with p_T^{ℓ} the transverse momentum of the lepton and $\Delta\phi$ the azimuthal separation between the lepton and $\vec{p}_T^{\,\text{miss}}$ vector.

- The azimuthal angle of the $\vec{p}_T^{\,\text{miss}}$ vector, ϕ^{miss}.
- The number of jets entering the S_T sum.
- The number of these jets identified as originating from a b quark.
- The isolated-lepton momentum, expressed in polar coordinates (p_T, η, ϕ)
- The three isolation quantities (`ChPFIso`, `NeuPFIso`, `GammaPFIso`) for the isolated lepton.
- The lepton charge.
- The $isEle$ flag for the isolated lepton.

The list of 801 particles is used to generate two visual representations of the events: *raw representation* and *abstract representation*. In the *raw representation*, the (η, ϕ) plane corresponding to the detector acceptance is divided into a barrel region ($|\eta| < 1.5$), two end-cap regions ($1.5 \leq \eta < 3.0$ and $-3.0 < \eta \leq -1.5$), and two forward regions ($3.0 \leq \eta < 5.0$ and $-5.0 < \eta \leq -3.0$). The barrel and endcap regions of the electromagnetic calorimeter, as well as the endcap of the hadronic calorimeter (HCAL), are binned in cells of size 0.0187×0.0187. The barrel region of the HCAL is binned with cells of size 0.087×0.087. The forward regions are binned with cells of size 0.175 in η, while the dimension in ϕ varies from 0.175 to 0.35. Each cell is filled with the scalar sum of the p_T of the particles pointing to that cell. The three classes of particles (charged particles, photons, and neutral hadrons) are considered separately, resulting in three channels. An example is shown in Fig. 10.2 for a $t\bar{t}$ event. This representation corresponds to the raw image recorded by the detector.

Recently, it was proposed to represent LHC collision events as abstract images where reconstructed physics objects (jets, in that case) are represented as geometric shapes whose size reflects the energy of the particle [303]. We generalize this *abstract representation* approach by applying it to the full list of particles. Each particle is represented as a unique geometric shape, centered at the particle's (η, ϕ) coordinates and with size proportional to its log p_T. The geometric shapes are chosen as follow: (i) pentagons for the selected isolated electron or muon; (ii) triangles for photons;

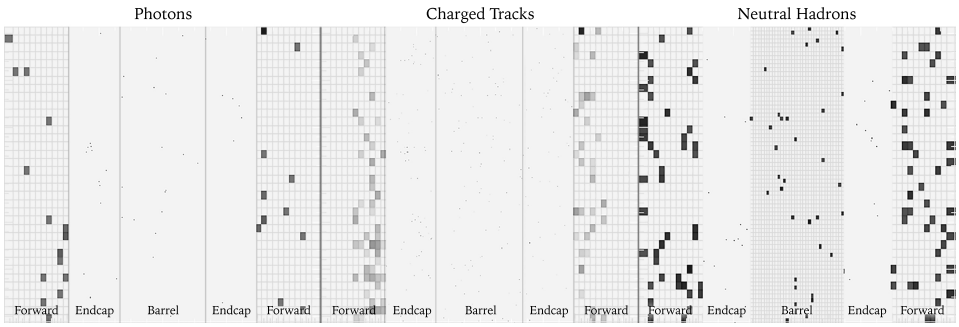


Figure 10.2: An example of a $t\bar{t}$ event as the input of the raw-image classifier. Vertical and horizontal axess are the ϕ and η coordinates, respectively, of the sub-detectors.

(iii) squares for charged particles; (iv) hexagons for neutral hadrons. The images are digitized as arrays of size $5\times150\times94$, where each of the first four channels contains a separated particle class, and the last channel contains the E_T^{miss}, represented as a circle. As an example, the abstract representation for the event in Fig. 10.2 is shown in Fig. 10.3.

This abstract representation allows mitigating the sparsity problem of the raw images. On the other hand, there is no guarantee that the physics information is fully retained in this translation. As a result, there could be a reduction of discrimination power. This is one of the points we aim to investigate in this study.

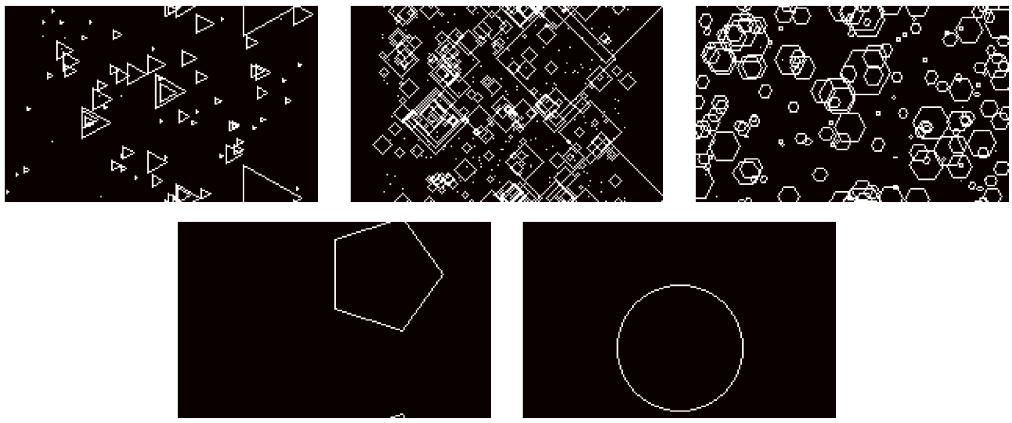


Figure 10.3: Example of a $t\bar{t}$ event, represented as a 5-channel abstract images of photons (top-left), charged hadrons (top-center), neutral hadrons (top-right), the isolated lepton (bottom-kleft), and the event E_T^{miss} (botton-right).

10.3 Model description

In this section, we describe five types of multi-class classifiers, trained on the four data representations described in the previous section. We start by considering a state-of-the art HEP application, based on the high-level features listed in Sec. 10.2. We then consider a convolutional neural network taking as input the raw images. This model offers the baseline point of comparison for the classifier using the abstract images. In order to have a fair comparison between the two approaches, the same kind of network architecture is used for the two sets of images. Next, we consider recurrent neural networks based on LSTMs and GRUs, trained directly on the lists of 801 particles. Finally, we consider a classifier taking both the high-level features and the list of 801 particles as inputs, using a combination of recurrent neural networks and fully connected neural networks.

The CNNs are implemented in `PyTorch` [206]. The recurrent neural networks and feed-forward neural networks are implemented in `Keras` [120] and trained using `Theano` [304] as a back-end. The Adam optimizer [212] is used to adapt the learning rate. The training is capped at 50 epochs, and can be stopped early if there is no improvement in terms of validation loss after 8 epochs. Categorical cross entropy is used as the loss function. All trainings are performed on a cluster of GeForce GTX 1080 GPUs. In an early stage of this work, experiments on the recurrent models were performed on the CSCS Piz Daint super computer, using the `mpi-learn` library [305] for multiple-GPU training.

High-level-feature classifier

A fully connected feed-forward DNN based on a set of high-level features (*HLF classifier*) is the closest approach to the currently used rule-based trigger algorithms. We train a model of this kind taking as input the 14 features contained in the HLF dataset (see Sec. 10.2). The 14 features are normalized to take values between 0 and 1.

The final network configuration is the result of an optimization process performed using the `scikit-learn` optimizer [273], which performs an exhaustive cross-validated grid-search over a set of hyperparameters related to the network architecture and the training setup. The number of layers, the number of nodes in each layer, and the choice of optimizer have been considered in the scan. For a given number of layers, discrimination performances were found to be constant over the considered range of number of nodes per layer. We believe that this is a direct consequence of

the simple problem at hand: even a relatively small networks achieve good classification performances. We then took the smallest network as the best compromise between performance and architecture minimality.

The chosen architecture consists of three hidden layers with 50, 20, and 10 nodes, activated by rectified linear units (ReLU) [209]. The output layer consists of 3 nodes, activated by a softmax activation function.

Raw-image classifier

To classify events represented as raw calorimeter images (*raw-image classifier*), we use DenseNet-121, a model based on the Densely Connected Convolutional Network [306]. The DenseNet-121 architecture includes 4 dense blocks, each of which contains 6, 12, 24, 16 dense layers, respectively. Each dense layer contains two 2D convolutional layers preceded by batch normalization layers. A dropout rate of 0.5 is applied after each dense layer. Between two subsequent dense blocks is a transition layer consisting of a batch normalization layer, a 2D convolutional layer, and an average pooling layer.

Abstract-image classifier

We use the same DenseNet-121 architecture above to classify the abstract image representation. We refer to this model as *abstract-image classifier*.

Particle-sequence classifier

A *particle-sequence classifier* is trained using a recurrent network, taking as input the 801 candidates. To feed these particles into a recurrent network, particles are ordered according to their increasing or decreasing distance from the isolated lepton. Different physics-inspired metrics are considered to quantify the distance (ΔR, $\Delta\phi$, $\Delta\eta$, k_T [144], or anti-k_T [57]). The best results are obtained using the ΔR decreasing distance ordering.

We use gated recurrent units (GRU) to aggregate the input sequence of particle flow candidate features into a fixed size encoding. The fixed encoding is fed into a fully connected layer with 3 softmax activated nodes. Input data is standardized so that each feature has zero mean and unit standard deviation. The zero-padded entries in the particle sequence are skipped with the Masking layer. The best internal width of the recurrent layers was found to be 50, determined by k-fold cross validation on a training set of 210,000 events. We also considered using long short-term memory

networks (LSTM) to replace the GRU, but we found that the GRU architecture outperformed the LSTM architecture for the same number of internal cells.

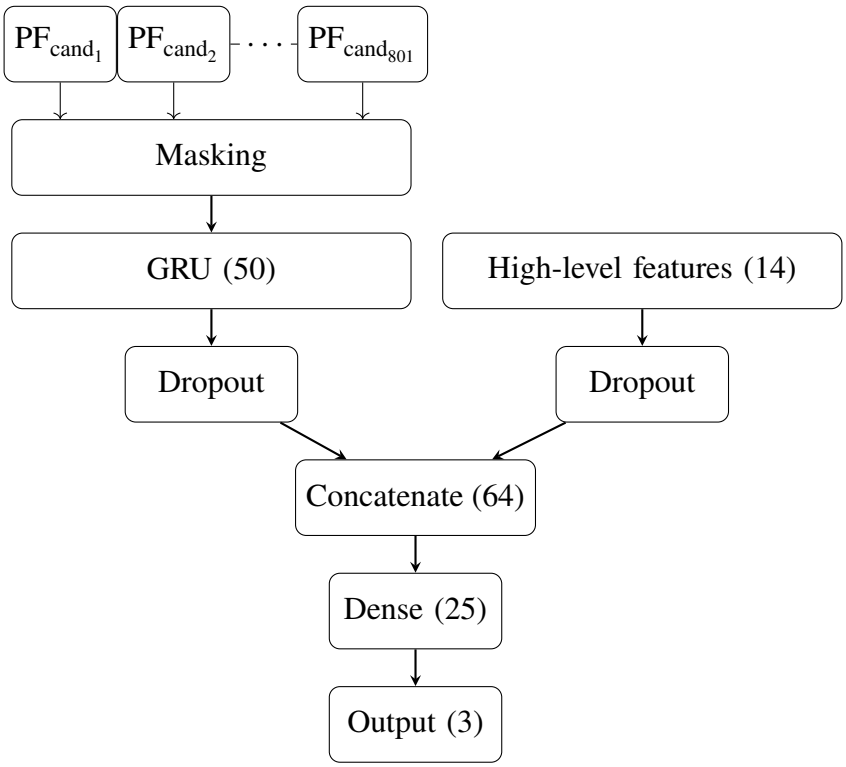


Figure 10.4: Network architecture of the inclusive classifier.

Inclusive classifier

In order to inject some domain knowledge in the GRU classifier, we consider a modification of its architecture in which the 14 features of the HLF dataset are concatenated to the output of the GRU layer after some dropout (see Fig. 10.4). As for the other classifiers, the final output layer consists of 3 nodes, activated by a softmax activation function. We refer to this model as *inclusive classifier*.

10.4 Results

Each of the models presented in the previous section returns the probability of each event to be associated to a given topology: y_{QCD}, y_W, and $y_{t\bar{t}}$. By applying a threshold requirement on y_W or $y_{t\bar{t}}$, one can define a W or a $t\bar{t}$ classifier, respectively. By changing the threshold value, one can build the corresponding receiver operating characteristic (ROC) curve. Fig. 10.5 shows the comparison of the ROC curves for five classifiers: the DenseNets based on raw images and abstract images, the GRU using the list of particles, the DNN using the HLFs, and the inclusive classifier using

both the HLFs and the list of particles. Results for both a $t\bar{t}$ and W selectors are shown.

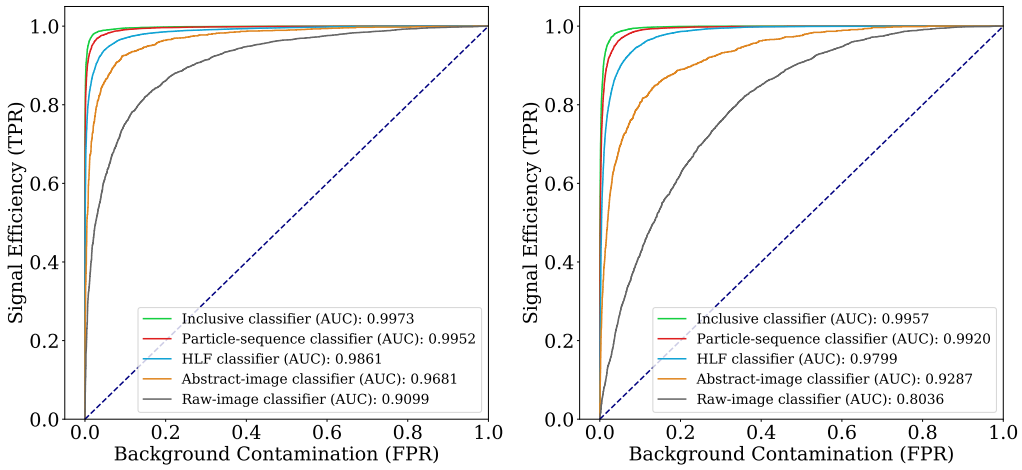


Figure 10.5: ROC curves for the $t\bar{t}$ (left) and W (right) selectors described in the chapter.

Acceptable results are obtained already with the raw-image classifier. On the other hand, the use of abstract images allows us to reach better performances. A further improvement is observed for those models not using an image-based representation of the event. The fact that the HLF selectors perform so well does not come as a surprise, given a considerable amount of physics knowledge implicitly provided by the choice of the relevant features. On the other hand, the fact that the particle-sequence classifier reaches better performances compared to the HLF selector is remarkable, as is the further improvement observed by merging the two approaches in the inclusive classifier. In some sense, the GRU layer is gaining a good part of the physics intuition that motivated the choice of the HLF quantities, but not entirely. Fig. 10.6 shows the Pearson correlation coefficients between the GRU scores ($y_{t\bar{t}}$ and y_W) and the HLF quantities. As one would expect, $y_{t\bar{t}}$ exhibits a stronger correlation with those features that quantify jet activity (n_{jets} in Fig. 10.6), as well as with the b-jet multiplicity ($n_{\text{b-jets}}$). On the contrary, W events shows an anti-correlation with respect to jet quantities, since the production of associated jets in W events is much more penalized than for $t\bar{t}$ events. As expected, both scores are anti-correlated to the isolation quantities, which takes larger values for non-isolated leptons.

The performance of each of the five classifiers is summarized in Tab. 10.1 in terms of false-positive rate (FPR) and trigger rate (TR) as a function of the true-positive rate (TPR). The best QCD rejection is obtained by the inclusive classifier, which can retain 99% of the $t\bar{t}$ or W events with a false-positive rate of $\sim 5.2\%$.

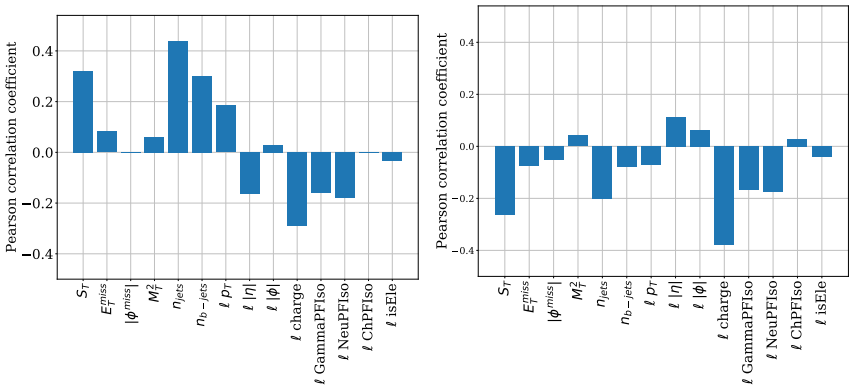


Figure 10.6: Pearson correlation coefficients between the $y_{t\bar{t}}$ (left) and y_W (right) scores of the Particle-sequence classifier and the 14 quantities of the HLF dataset.

Table 10.1: False positive rate (FPR) and trigger rate (TR) at different values of the true positive rate (TPR), for a $t\bar{t}$ (top) and W selector. Rate values are estimated scaling the TPR and process-dependent FPR values by the acceptance and efficiency, assuming a leading-order (LO) production cross section and luminosity of $2{\times}10^{34}$ cm^{-2} s^{-1}. TR values should be taken only as suggestions of the actual rates, since the accuracy is limited by the use of LO cross sections and a parametric detector simulation.

$t\bar{t}$ selector	Raw-image (DenseNet)	Abstract-image (DenseNet)	HLF (DNN)	Particle-sequence (GRU)	Inclusive (DNN+GRU)
FPR @99% TPR	76.5 ± 0.2%	50.1 ± 0.2%	28.6 ± 0.2%	9.2 ± 0.1%	5.2 ± 0.1%
FPR @95% TPR	41.3 ± 0.2%	15.7 ± 0.1%	6.1 ± 0.1%	1.7 ± 0.1%	0.7 ± 0.0%
FPR @90% TPR	26.5 ± 0.2%	7.4 ± 0.1%	2.7 ± 0.1%	0.6 ± 0.0%	0.2 ± 0.0%
TR @99% TPR	382.0 ± 0.9 Hz	250.9 ± 1.0 Hz	143.9 ± 0.9 Hz	48.1 ± 0.6 Hz	28.4 ± 0.4 Hz
TR @95% TPR	207.8 ± 1.0 Hz	80.3 ± 0.7 Hz	32.4 ± 0.5 Hz	11.0 ± 0.3 Hz	6.0 ± 0.2 Hz
TR @90% TPR	134.2 ± 0.9 Hz	39.0 ± 0.5 Hz	15.5 ± 0.3 Hz	5.2 ± 0.2 Hz	3.5 ± 0.1 Hz
W selector	Raw-image (DenseNet)	Abstract-image (DenseNet)	HLF (DNN)	Particle-sequence (GRU)	Inclusive (DNN+GRU)
FPR @99% TPR	79.0 ± 0.2%	61.8 ± 0.2%	23.5 ± 0.2%	10.2 ± 0.1%	6.3 ± 0.1%
FPR @95% TPR	60.5 ± 0.2%	36.0 ± 0.2%	9.7 ± 0.1%	3.7 ± 0.1%	1.8 ± 0.1%
FPR @90% TPR	48.1 ± 0.2%	22.8 ± 0.2%	5.1 ± 0.1%	1.8 ± 0.1%	0.9 ± 0.0%
TR @99% TPR	488.9 ± 0.3 Hz	462.3 ± 0.5 Hz	301.9 ± 0.6 Hz	268.2 ± 0.5 Hz	259.7 ± 0.4 Hz
TR @95% TPR	454.5 ± 0.6 Hz	365.1 ± 0.8 Hz	259.2 ± 0.5 Hz	242.6 ± 0.4 Hz	238.0 ± 0.4 Hz
TR @90% TPR	408.2 ± 0.8 Hz	301.8 ± 0.8 Hz	235.0 ± 0.5 Hz	225.4 ± 0.5 Hz	223.3 ± 0.5 Hz

The trigger baseline selection we use in this study, looser than what is used nowadays in CMS, gives an overall trigger rate (i.e., summing electron and muon events) of $\sim$ 690 Hz, more than a factor two larger than what is currently allocated. Using the 99% working points of the two classifiers, one would reduce the overall rate to $\sim$ 270 Hz (counting the overlap between the two triggers). This would be comparable to what is currently allocated for these triggers, but with a looser selection, i.e., with a less severe bias on the offline analysis. In addition, the trigger efficiency (the

TPR) is so high that the bias imposed on offline quantities is quite minimal. This is illustrated in Fig. 10.7, where the dependence of the TPR on the most relevant HLF quantities is shown. In our experience, any rule-based algorithm with the same target trigger rate would result in larger inefficiencies at small values of at least some of these quantities, e.g., the lepton p_T. One should also consider that the principle of a topology classifier could be generalized to other physics cases, as well as to other uses (e.g., labels for fast reprocessing or access to specific subsets of the triggered samples).

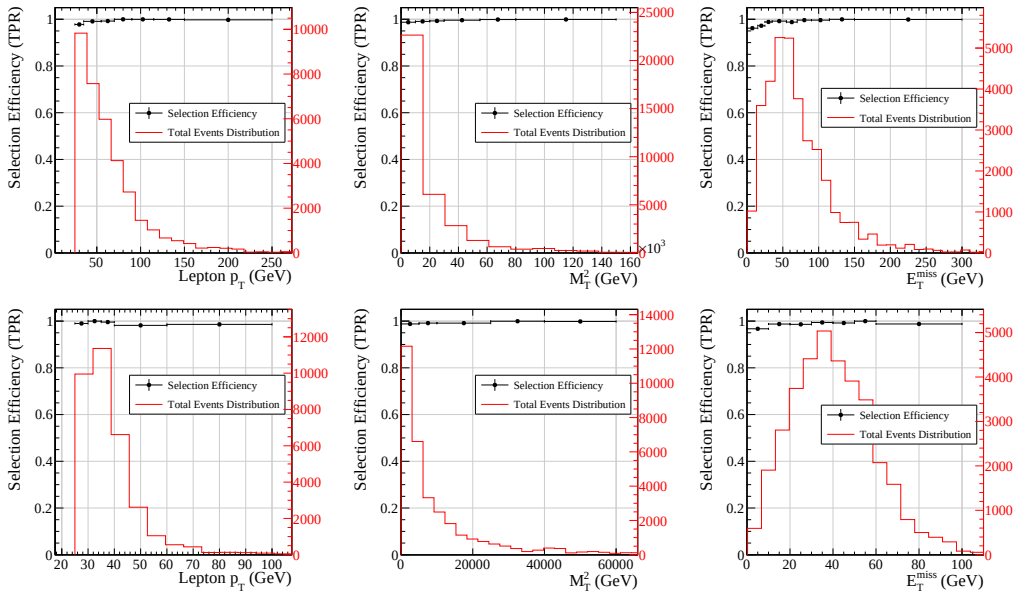


Figure 10.7: Selection efficiency using 99% TPR working point as functions of lepton p_T, M_T^2, and E_T^{miss} for the $t\bar{t}$ selector on $t\bar{t}$ events (top) and the W selector on W events (bottom).

Figure 10.8 shows the TPR and FPR of the inclusive $t\bar{t}$ selector when applying the 99% TPR working-point threshold, as a function of the number of vertices in the event, which quantifies the amount of pileup. The TPR is fairly insensitive to PU until $PU \sim 35$, (the average PU recorded by the LHC in 2018), where the TPR drops to 97%. At the same time, the FPR increases mildly, resulting in a rate increase from ~ 34 Hz (at the average PU value ~ 20) to ~ 48 Hz at $PU \sim 35$. In other words, the algorithm trained on 2016 conditions would have been sustainable until 2018 with $\sim 15\%$ rate increase (with respect to the average value) or it would have required a threshold adjustment along the way, a pretty standard operation when designing a trigger menu at the beginning of the year. We believe that, in view of these facts, the

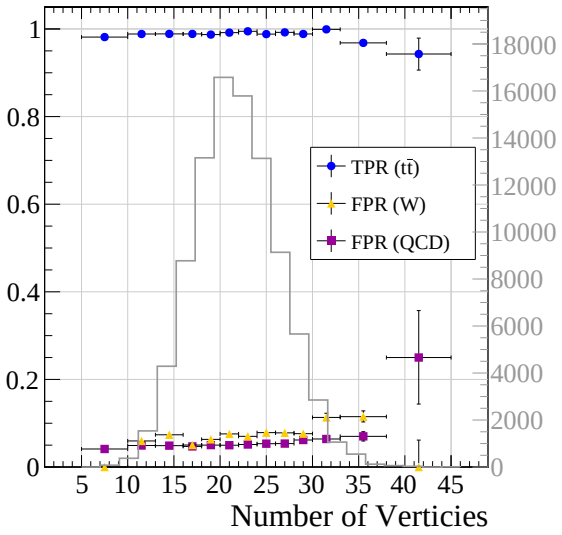


Figure 10.8: Dependence of TPR and FPR on the amount of pileup in the event (estimated through the number of vertices) for the inclusive $t\bar{t}$ selector when applying the 99% TPR working-point threshold. The gray histogram shows the distribution of the number of vertices in the training dataset, covering a wide range from $\sim$ 10 to $\sim$ 40 following a Poisson distribution with mean value of 20.

proposed algorithm would be as robust as many state-of-the-art algorithms operated at the LHC experiments.

10.5 Impact on other topologies

While reducing the resource consumption of standard physics analyses is the main motivation behind this study, it is important to evaluate the impact of the proposed classifiers on other kind of topologies. For this purpose, we consider a handful of beyond-the-standard-model (BSM) scenarios, and we compute the TPR as a function of the most relevant kinematic quantities, similar to what was done in Fig. 10.7 for the standard topologies.

We consider the following BSM processes:

- $A \rightarrow H^{+}W$: a heavy Higgs boson A with mass 425 GeV decaying to a charged Higgs boson H^{+} of mass 325 GeV and a W^{-} boson. The H^{+} then decays to a $W^{+}H^{0}$ final state, where H^{0} is the 125 GeV Higgs boson, which we force to decay to a bottom quark-antiquark pair. This model, introduced in Ref. [307], generates a $2b2W$ topology similar to that given by $t\bar{t}$ events.

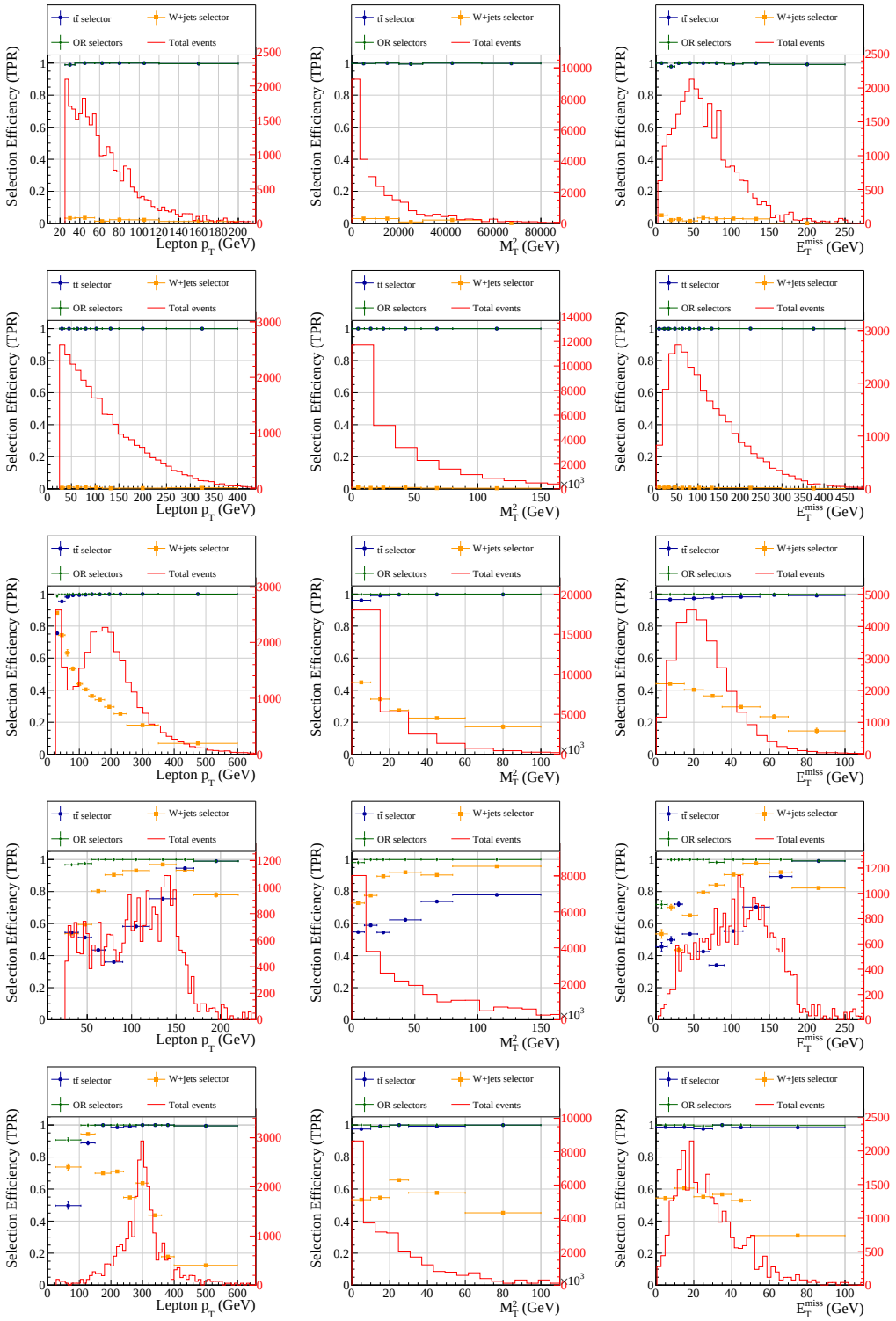


Figure 10.9: Selection efficiencies of different BSM models using 99% TPR working point as functions of lepton p_T, M_T^2, and E_T^{miss}. From top to bottom, $A \to H^+W^-$, High-mass $A \to H^+W^-$, $A \to 4\ell$, W', and Z'.

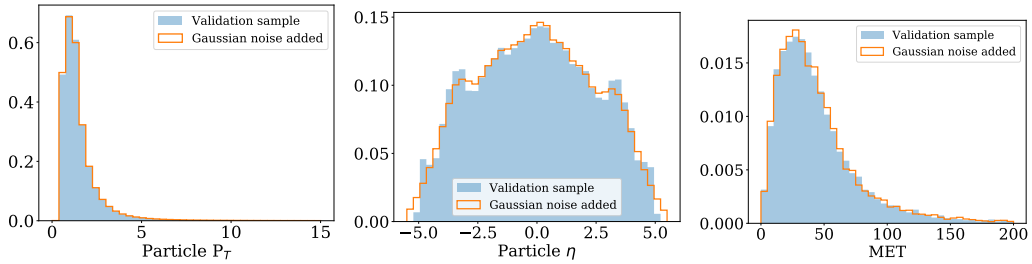


Figure 10.10: Distributions of the validation sample and pseudo-data. The pseudo-data is created by adding a Gaussian noise of mean zero and standard deviation of 10% to the validation sample's particle momenta. The high-level features are then recomputed with the new list of particles.

- High-mass $A \to H^+W$: a high-mass variation of the previous model, in which the A and H^+ masses are set to 1025 GeV and 625 GeV, respectively.
- $A \to 4\ell$: a light neutral scalar particle A with mass 20 GeV, decaying to two neutral scalars of 5 GeV each, both decaying to muon pairs, for a total of four muons in the final state.
- W' resonance with mass 300 GeV, decaying inclusively with W-like couplings.
- Z' resonance with mass 600 GeV, decaying to a pair of electrons or muons.

These events are filtered with the baseline selection described in Sec. 10.2.

For each of these models, we consider the inclusive classifier and apply the 99%-TPR thresholds on $y_{t\bar{t}}$ and y_W. We then consider the fraction of events passing at least one of the two selectors. Results are shown in Fig. 10.9 for the most relevant kinematic quantities. While the individual selectors might show local inefficiencies, the combination of the two trigger paths is perfectly capable of retaining any event with features different from that of a QCD multijet event. In this respect, the logical `OR` of our two exclusive topology classifiers is robust enough to also select a large spectrum of BSM topologies. On the other hand, one cannot guarantee that QCD-like topologies (e.g., a dark photon produced in jet showers and decaying to lepton pairs) would not be rejected, a limitation which also affects traditional inclusive trigger strategies.

10.6 Robustness study

As the classifier is trained on Monte-Carlo simulation samples, one needs to consider the discrepancy between Monte-Carlo and real data when deploying the classifier

Table 10.2: Signal efficiency (TPR) at different values of the false positive rate (FPR) for the *inclusive classifier* selecting $t\bar{t}$ evaluated on the validation sample and the pseudo-data.

FPR	TPR on validation sample	TPR on pseudo-data
5.2%	99.0 ± 0.1%	97.6 ± 0.1%
0.7%	95.0 ± 0.1%	90.9 ± 0.2%
0.2%	90.0 ± 0.2%	83.5 ± 0.2%

in the trigger. We investigate the robustness of our topology classifiers against this discrepancy by creating a pseudo-data sample, which attempts to emulate real data by adding a Gaussian noise to the particles' momenta in the simulation samples. The Gaussian noise has mean of zero and standard deviation of 10% of the variable's values being applied. Fig. 10.10 shows some comparisons between the Monte-Carlo samples and the pseudo-data with this Gaussian noise added.

We evaluate the performance of our fully-trained inclusive classifier on the new pseudo-data. Tab. 10.2 shows a slight reduction of signal efficiency: at the same background contamination rate of 5.2%, the signal efficiency reduces by only 1.4%. This demonstrates that our classifiers can be robust against some augmentation that mimics the discrepancy between data and Monte-Carlo simulation. A comprehensive study on full simulation and data in proper control regions would be needed when deploying this classifier into production.

10.7 An alternative use case

In this chapter, we showed how one could use a topology classifier to keep the overall trigger rate under control while operating triggers with otherwise unsustainable loose selections. In this appendix we discuss how topology classifiers could be used to save resources for a pre-defined baseline trigger selection by rejecting events associated to unwanted topologies. In this case, the main goal is not to reduce the impact of the online selection. Instead, we focus on reducing resource consumption downstream for a given trigger selection.

To this purpose, we consider a copy of the dataset described in Sec. 10.2, obtained tightening the p_T threshold from 23 to 25 GeV and the isolation requirement from `ISO` < 0.45 to `ISO` < 0.20. Doing so, the sample composition changes as follow: 7.5% QCD; 92% W; 0.5% $t\bar{t}$. With such selections, the trigger acceptance rate would decrease from 690 Hz to 390 Hz, closer to what is currently allocated for these triggers in the CMS experiment.

Following the procedure described in Sec. 10.3 and 10.4, we train the same topology classifiers on this dataset. The corresponding ROC curves are presented in Fig. 10.11 for a $t\bar{t}$ and a W selector.

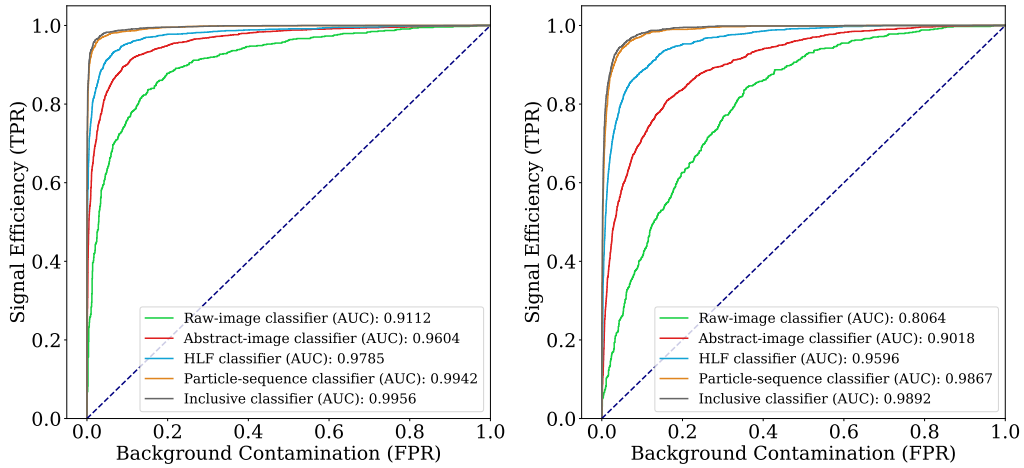


Figure 10.11: ROC curves for the $t\bar{t}$ (left) and W (right) selectors described in the chapter, trained on a dataset defined by a tighter baseline selection.

We then define a set of trigger filters applying a lower threshold to the normalized score of the classifier, choosing the threshold value that corresponds to a certain TPR value. The result is presented in Table 10.3, in terms of the FPR and the trigger rate.

The trigger baseline selection we use in this study, close to what is used nowadays in CMS for muons, gives an overall trigger rate (i.e., summing electron and muon events) of $\sim$ 390 Hz (i.e., 190 Hz per lepton flavor). If one was willing to take (as an example) half the W events and all the $t\bar{t}$ events, this number could be reduced to $\sim$ 200 Hz using the inclusive selectors presented in this study (taking into account the partial overlap between the two triggers). A more classic approach would consist in prescaling the isolated lepton triggers, i.e. randomly accepting half of the events. The effect on W events would be the same, but one would lose half of the $t\bar{t}$ events while still writing 15 times more QCD than $t\bar{t}$ events. In this respect, the strategy we propose would allow a more flexible and cost-effective strategy.

10.8 Related works

Machine learning is traditionally used in high-energy physics as part of data analysis, and was an important ingredient to the discovery of the Higgs boson, as discussed in [308]. Several classification algorithms have been studied in the context of LHC physics application, notably for jet tagging [172–179] and event topology

Table 10.3: False positive rate (FPR) and trigger rate (TR) corresponding to different values of the true positive rate (TPR), for a $t\bar{t}$ (top) and W selector. Rate values are estimated scaling the TPR and process-dependent FPR values by the acceptance and efficiency, assuming a leading-order (LO) production cross section and luminosity of $2{\times}10^{34}$ cm^{-2} s^{-1}. TR values should be taken only as a loose indication of the actual rates, since the accuracy is limited by the use of LO cross sections and a parametric detector simulation.

$t\bar{t}$ selector	Raw-image (DenseNet)	Abstract-image (DenseNet)	HLF (DNN)	Particle-sequence (GRU)	Inclusive (DNN+GRU)
FPR @99% TPR	76.7 ± 0.2%	55.5 ± 0.3%	44.3 ± 0.3%	13.4 ± 0.2%	10.2 ± 0.2%
FPR @95% TPR	43.5 ± 0.3%	20.2 ± 0.2%	9.1 ± 0.2%	2.1 ± 0.1%	1.5 ± 0.1%
FPR @90% TPR	24.8 ± 0.3%	9.9 ± 0.2%	4.2 ± 0.1%	0.6 ± 0.0%	0.5 ± 0.0%
TR @99% TPR	285.8 ± 0.9 Hz	230.4 ± 1.0 Hz	219.6 ± 1.0 Hz	56.7 ± 0.7 Hz	42.4 ± 0.6 Hz
TR @95% TPR	148.9 ± 1.0 Hz	84.6 ± 0.9 Hz	37.2 ± 0.6 Hz	9.9 ± 0.3 Hz	8.3 ± 0.3 Hz
TR @90% TPR	72.9 ± 0.8 Hz	41.6 ± 0.6 Hz	18.6 ± 0.4 Hz	3.9 ± 0.2 Hz	3.8 ± 0.2 Hz
W selector	Raw-image (DenseNet)	Abstract-image (DenseNet)	HLF (DNN)	Particle-sequence (GRU)	Inclusive (DNN+GRU)
FPR @99% TPR	81.3 ± 0.2%	68.9 ± 0.3%	45.7 ± 0.3%	17.3 ± 0.2%	14.9 ± 0.2%
FPR @95% TPR	58.4 ± 0.3%	43.9 ± 0.3%	19.6 ± 0.2%	6.1 ± 0.1%	5.2 ± 0.1%
FPR @90% TPR	46.9 ± 0.3%	30.2 ± 0.3%	11.7 ± 0.2%	3.0 ± 0.1%	2.5 ± 0.1%
TR @99% TPR	385.9 ± 0.2 Hz	384.3 ± 0.2 Hz	376.3 ± 0.2 Hz	363.1 ± 0.2 Hz	362.8 ± 0.2 Hz
TR @95% TPR	367.5 ± 0.5 Hz	360.8 ± 0.5 Hz	349.7 ± 0.5 Hz	344.2 ± 0.4 Hz	343.9 ± 0.5 Hz
TR @90% TPR	343.6 ± 0.6 Hz	336.6 ± 0.6 Hz	323.8 ± 0.6 Hz	325.0 ± 0.6 Hz	324.7 ± 0.6 Hz

identification [303, 307, 309] using feed-forward neural networks, convolutional neural networks or physics-inspired architectures. Lists of particles have been used to define jet and event classifiers starting from a list of reconstructed particle momenta [168–170]. These studies typically consider data analysis as the main use case, focusing on small FPR selections. This is the main difference with respect to this study, which focuses on the optimization of real-time data-taking procedure.

In parallel, machine learning techniques have also been used in online event selection. For example, the LHCb experiment used a decision-tree based approach for the high-level trigger in the first LHC run [310] and re-optimized it with MatrixNet algorithm for Run II [311]; ATLAS uses BDT in its multi-step tau trigger for Run II [312]; a BDT was also deployed on FPGA cards of the hardware-level trigger of the CMS experiment [313]. These triggers are mainly based on high-level features related to specific parts of a collision event. We propose instead to define an algorithm that is based on a raw-event representation and considers the full event collision at once. To our knowledge, this is the first demonstration of how a recurrent neural network could perform a successful inference on a full event and improve topology identification based on object-specific features.

In addition, traditional triggers based on machine learning run in *tagging mode*, i.e., are used to identify certain types of particles. Instead, we propose to use our topology classifier in *veto mode*: the trigger algorithm running downstream would be a classic trigger with loose selection, which would normally be unsustainable due to high throughput. The topology classifier would subsequently remove a majority of background events, sustaining the trigger rate and saving downstream computing resources.

10.9 Summary

We show how deep neural networks can be used to train topology classifiers for LHC collision events, which could be used as a cleanup filter to select or reject specific event topologies in a trigger system. We consider several network architectures, applied to different representations of the same collision datasets.

The best results are obtained by combining a set of physics-motivated high-level features with the output of a GRU unit applied to a list of particle-level features. For the most difficult case, i.e., selecting rare $t\bar{t}$ events, we show how a trigger based on this concept would retain 99% of the $t\bar{t}$ events while reducing the FPR by more than ~ 10 times.

The information given as input to the GRU, the abstract-image CNN and the raw-image CNN is the same, but coded differently. The difference in performance is then a combination of two effects: the encoding of this information in the input event representation and the way the network architecture exploits it. The DNN case is different. The DNN uses in principle less information. On the other hand, the list of HLFs given as input to the DNN is based on domain knowledge that the other networks have to learn by themselves. This is why the DNN model is very competitive despite using less information and why the inclusive classifier (GRU+DNN) improves on the GRU-based particle sequence classifier. Nevertheless, it is remarkable that the score of the particle sequence classifier learns interesting correlation patterns with the HLF features, showing that (to some extent) the GRU is learning some of this domain knowledge.

We show that such a trigger would have a minimal impact on the main kinematic features of the event topologies under consideration. The effect of operating this topology classifier as a final filter of a given single-lepton trigger would result in small decrease of trigger efficiency by few percentage (depending on the TPR of the chosen working point). On the other hand, such a filter would allow for a looser

selection, efficiently including non-isolated leptons with low p_T without downstream consequences in terms of computational power and storage. In addition, the logic `OR` of the $t\bar{t}$ and W selections would also catch a broad class of new-physics topologies, on which the classifiers were not trained.

The advantages of running these types of algorithms comes at the cost of computational resources to train the models. In our case, a single training of the *inclusive classifier* took 4 hours on a cluster consisting of 6 GeForce GTX 1080 GPUs. Building a cluster of a few tens of GPUs of this kind, to be used as a training facility, is well within the budget of big-experiment computing projects. For this reason, dedicated studies are ongoing to integrate train-on-demand services in the computing infrastructures of LHC experiments [305, 314]. In view of the challenging trigger environment foreseen for the High-Luminosity LHC, it would be important to test this trigger strategy as a way to preserve a good experimental reach with a substantial reduction of computational resources. In this respect, we look forward to the LHC Run III as an opportunity to experiment with this technique using full simulation and study its impacts on real-time event selection.

Chapter 11

GENERATIVE ADVERSARIAL NETWORKS FOR FULL-EVENT SIMULATION

We investigate how a Generative Adversarial Network could be used to generate a list of particle four-momenta from LHC proton collisions, allowing one to define a generative model that could abstract from the irregularities of typical detector geometries. As an example of application, we show how such an architecture could be used as a generator of LHC parasitic collisions (pileup). We present two approaches to generate the events: unconditional generator and generator conditioned on missing transverse energy. We assess generation performances in a realistic LHC data-analysis environment, with a pileup mitigation algorithm applied.

11.1 Introduction

The simulation of subatomic particle collisions, their subsequent detector interaction, and their reconstruction is a computationally demanding task for the computing infrastructure of the experiments operating at the CERN Large Hadron Collider (LHC). The high accuracy of state-of-the-art Monte Carlo (MC) simulation software, typically based on the GEANT4 [111], has a high cost: MC simulation amounts to about one half of the experiments' computing budget and to a large fraction of the available storage resources [315], the other half being largely used to process simulated and real data (event reconstruction).

Following their invention in 2014, GANs [285] gained traction as generative models, often superior to Variational Autoencoders [251] and with very impressive results in image production [316, 317]. Due to their high inference speed, GANs can be used as fast-simulation libraries. This approach has been successfully investigated with proof-of-principle studies related to particle showers in multilayer calorimeters [318–320] and particle jets [321], as well as in similar application to different HEP domains [322–327]. All these studies formalized the simulation task in terms of either image generation or analysis-specific high-level features.

In this work, we present pGAN, a full-event particle-based generative model that can be used to emulate pileup simulation at the LHC.

11.2 Pileup simulation

The majority of LHC proton-proton collisions result in so-called *minimum-bias* (MB) events, i.e., in low-energy (*soft*) interactions between proton constituents. These events are characterized by low-p_T particles, as opposed to the head-on collision processes typically studied at the LHC (so-called *hard* or high-p_T interactions). Any hard interaction happens simultaneously to many parasitic MB events, generically referred to as *pileup*. Pileup simulation is a fundamental aspect of a realistic LHC simulation software. The current implementation of pileup simulation consists in overlapping a set of MB events to the main high-p_T collision. Events could be generated on demand or be sampled from a pre-generated library. The former is computationally expensive, while the latter is inflexible and suffers from I/O issue.

GAN emerges as a possible solution to speed up the on-demand generation of MB events and remove the need for a pre-generated library. To our knowledge, the only application of machine learning to pileup simulation is the work presented in Ref. [328], where pileup images are generated using a Deep Convolutional GAN model (DCGAN) [329]. However, an image-based event representation cannot be used as input for downstream reconstruction algorithms. In addition, our proposed pGAN, which uses particle-based event representation, can abstract from the details of the detector geometry (e.g., its irregularities) and better scales with the foreseen increase of detector complexity.

11.3 Dataset

Synthetic MB events from proton-proton collisions are produced using the `PYTHIA8` [103] event generator. The center-of-mass energy of the collision is set to 13 TeV, corresponding to the LHC Run II (2015-2018). All soft QCD processes are activated, allowing for both initial- and final-state radiation as well as multiple parton interactions. The produced events are passed to `DELPHES` [269], to emulate the detector response. We take as a reference the upgraded design of the CMS detector, foreseen for the High-Luminosity LHC phase. The `DELPHES` particle-flow reconstruction algorithm is applied, returning the list of charged particles, photons, and neutral hadrons in the event. Minimum bias events are then combined to simulate a per-event pileup contribution, with mean number of MB events $\overline{n}_{\mathrm{PU}} = 20$ following a Poisson distribution. We randomly sample n_{PU} events from the MB dataset and mix them by merging the list of charged particles, neutral hadrons, and photons across the events.

Due to the complexity of training on long sequences, we restrict a maximum of 150 particles per event: the 50 charged particles, 50 photons and 50 neutral hadrons with the highest p_T value. This choice is mainly due to technical limitations that a more powerful training setup might help to overcome. On the other hand, cutting the sequence after p_T ordering is a well motivated simplification of the problem: a typical physics analysis would be based on a pileup mitigation algorithm, which usually removes the majority of the soft pileup contamination.

11.4 Network architectures

A GAN consists of two neural networks, a generator $\mathcal{G}$ and a discriminator $\mathcal{D}$. Given a set of samples x, the aim of a GAN training is to learn the function $p_{\text{data}}(x) \in A$ under which the x samples are distributed. We define an n-dimensional prior of input noise $z \sim p_z(z) \in \mathbb{R}^n$. The generator $\mathcal{G}$ is a differentiable function with trainable parameters $\theta_{\mathcal{G}}$, mapping $\mathbb{R}^n$ to A. The discriminator $\mathcal{D}$, with trainable parameters $\theta_{\mathcal{D}}$, is a map between A and $[0, 1]$, returning the probability that a given sample belongs to the set of real samples rather than originating from $\mathcal{G}$. $\mathcal{D}$ is trained to assign the correct probability to both real and generated ("fake") data; $\mathcal{G}$ is trained to produce samples such that they maximize the probability of them being real $\mathcal{D}(\mathcal{G}(z))$.

We develop 2 models for pGAN: an unconditional pGAN where the generator starts from purely random noise z, and a conditional pGAN [330] where a label acts as an extension of z to allow for generation of events based on an initial condition. In our use case, the missing transverse energy $p_{\text{T}}^{\text{miss}}$ is chosen as the label due to its importance in most physics analyses.

The loss function to train $\mathcal{G}$ and $\mathcal{D}$ in an unconditional pGAN is described as:

$$\mathcal{L}^{\text{uncond}} \equiv \mathcal{L}_{\text{adv}} = \mathbb{E}_{z\sim p_z(z)}[\log(\mathbb{P}(D(G(z)) = 0)] + \mathbb{E}_{I\sim A}[\log(\mathbb{P}(D(I) = 1))] \,, \tag{11.1}$$

while for conditional pGAN, the loss function becomes:

$$\mathcal{L}^{\text{cond}} = \mathcal{L}_{adv} + \alpha\mathcal{L}_{aux} \,, \tag{11.2}$$

where

$$\begin{aligned} \mathcal{L}_{\text{aux}} = \mathbb{E}_{(z,p_{\text{T}}^{\text{miss}})\sim(p(z)\times f(A))} &\left[\frac{|p_{\text{T}}^{\text{miss}} - \hat{p}_T^{\text{miss}}(G(z|p_{\text{T}}^{\text{miss}}))|}{p_{\text{T}}^{\text{miss}}}\right] \\ + \mathbb{E}_{x\sim A} &\left[\frac{|p_{\text{T}}^{\text{miss}}(x) - \hat{p}_T^{\text{miss}}(x)|}{p_{\text{T}}^{\text{miss}}(x)}\right] , \end{aligned} \tag{11.3}$$

$\hat{p}_T^{\text{miss}}$ is the missing transverse energy value computed from list of particles input to the discriminator, $f(A)$ is the empirical $p_{\text{T}}^{\text{miss}}$ distribution of the dataset. In the conditional GAN setting, the dataset is transformed such that $\phi(\vec{p}_T^{\text{miss}}) = 0$ and the ϕ value of each particle is computed as azimuth angle between its momenta and $\vec{p}_T^{\text{miss}}$. This way a single scalar value $p_{\text{T}}^{\text{miss}}$ can describe the full vector $\vec{p}_T^{\text{miss}}$ since the direction is chosen as the coordinate axis.

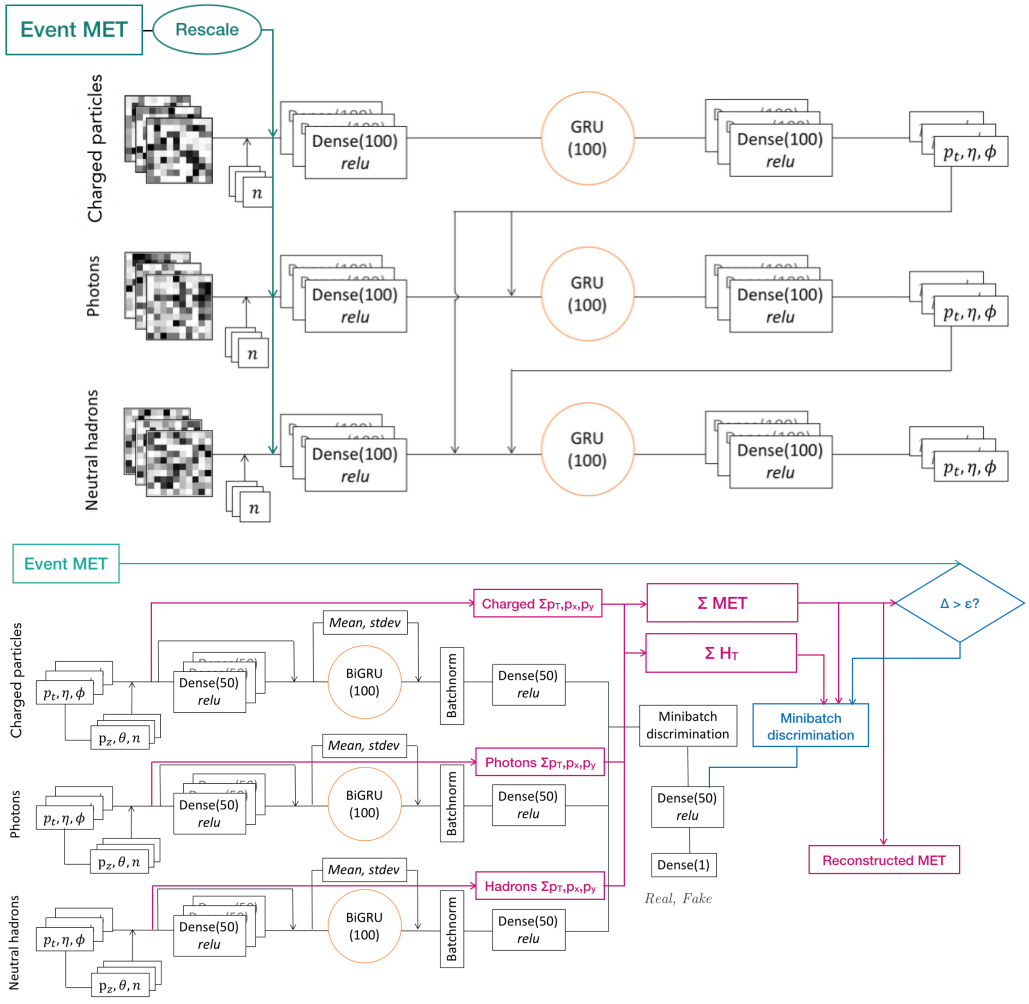


Figure 11.1: The architecture of conditional pGAN: generator $\mathcal{G}^{\text{cond}}$ (top) and discriminator $\mathcal{D}^{\text{cond}}$ (bottom). Arrows signify concatenation. Details are described in the text.

Unconditional pGAN

Generator

The $\mathcal{G}$ model takes as input a set of time steps, corresponding to the number of particles to generate, sampled from a uniform distribution from 0 to 50 for each

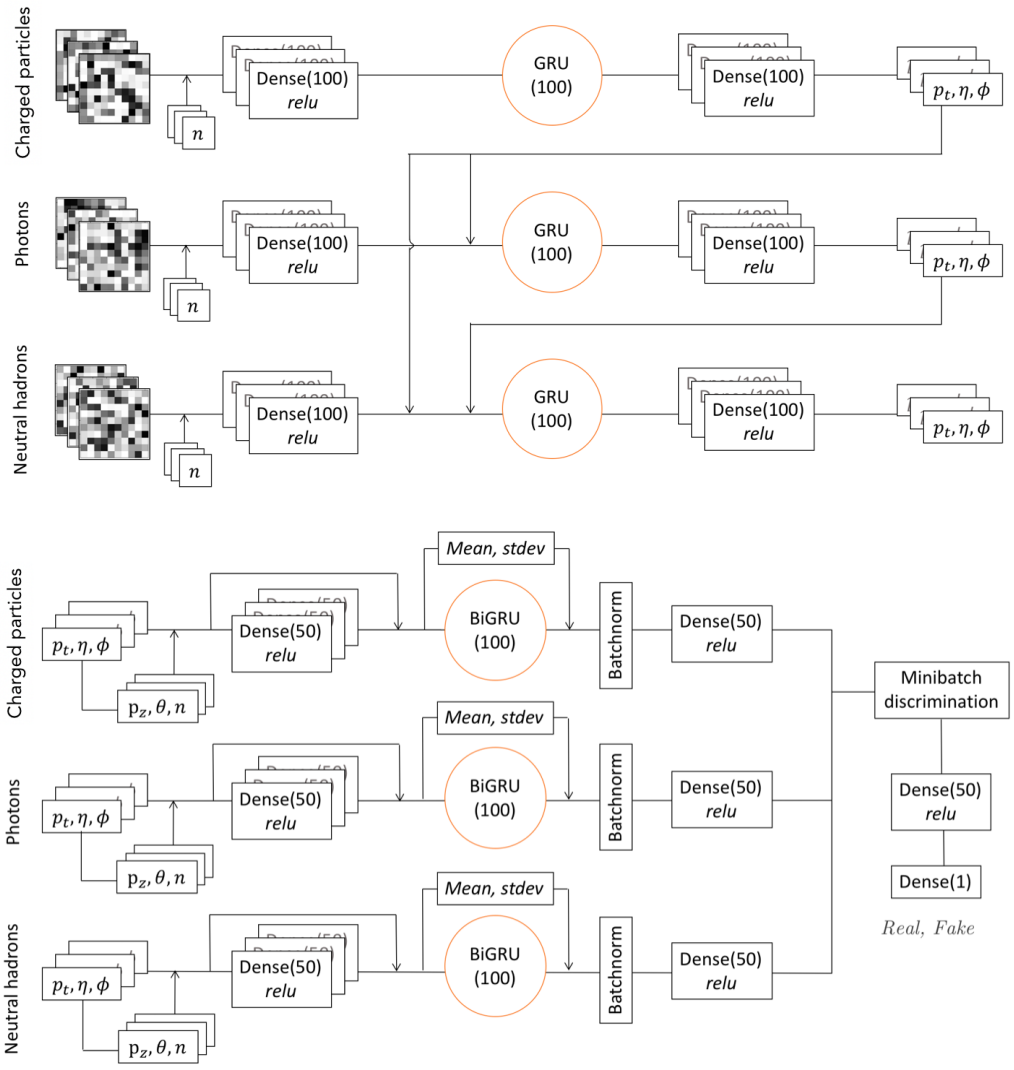


Figure 11.2: The architecture of unconditional pGAN: generator $\mathcal{G}^{\text{cond}}$ (top) and discriminator $\mathcal{D}^{\text{cond}}$ (bottom). Arrows signify concatenation. Details are described in the text.

particle class. Along with the noise, the current particle number n is given to the network. We represent each particle as a (p_T, ϕ, η) tuple. We enforce the ϕ-symmetry constraint through a mod-2π activation function for ϕ. The input in each branch from the noise input is processed via a block of Dense layers, a GRU layer, and another block of Dense layers as the final outputs. Each block of Dense consists 3 concatenated Dense layers that represent the particle's p_T, ϕ, and η, respectively.

Due to the peculiar shapes of the η distributions, we pre-train a small dense network that fed with a Gaussian-distributed input mimics the η distribution as output. This network is used to process the η output of the generator. This can be viewed

as an activation function parametrizing the η distribution in an unbiased way and decouples learning the distribution from the already complex particle generation task.

Discriminator

The discriminator $\mathcal{D}$ is of a binary classifier whose first step consists in a Physics layer, which takes each particle's defining features (p_T, ϕ, η, n) and concatenates to them other redundant features (θ, p_z, p). This Physics layer is introduced to maximize the information given to the discriminator without adding redundant information to the particle representation returned by the generator. This prevents the generator for having to learn dependencies between different features of the particle representation, while allowing the discriminator to exploit them. The discriminator makes use of a Bidirectional recurrent layer, which reduces the dependence on the long term memory of the GRU cell. A layer that calculates the mean and standard deviation of the features in a dense layer is included, in a similar spirit to feature matching [331]. Mini-batch discrimination is used to prevent mode collapse.

Conditional pGAN

Generator

The generator $\mathcal{G}^{\text{cond}}$ for the conditional pGAN, as shown in Fig. 11.1, is built on top of the unconditional generator $\mathcal{G}$. An initial value of the $p_{\text{T}}^{\text{miss}}$, sampled from real data, is injected as the input of the generator along with the noise after being rescaled to same range of the noise. This initial $p_{\text{T}}^{\text{miss}}$ is also used as the input to the discriminator. The rest of the generator architecture is similar to the unconditional pGAN's $\mathcal{G}$.

Discriminator

The discriminator $\mathcal{D}^{\text{cond}}$, as shown in Fig. 11.1, takes as inputs the 3 lists of particles along with the event $p_{\text{T}}^{\text{miss}}$, which is either the initial condition $p_{\text{T}}^{\text{miss}}$ (in the case of the generated lists of particles) or the actual $p_{\text{T}}^{\text{miss}}$ (in the case of sampling the lists of particles from the training data). In addition to usual computation flow in the unconditional $\mathcal{D}$ described in Sec. 11.4, $\mathcal{D}^{\text{cond}}$ also computes a few high-level features out of the input particle lists, in particular the reconstructed $p_{\text{T}}^{\text{miss}}$ and H_T (the scalar sum of all input particles' momenta), which are used as inputs to the final prediction. A binary flag Δ, which returns 1 if the absolute difference between

reconstructed $p_{\mathrm{T}}^{\mathrm{miss}}$ and the initial $p_{\mathrm{T}}^{\mathrm{miss}}$ is greater than ϵ and returns 0 otherwise, is also concatenated to the input of the final prediction. This can be viewed as an attempt to let the discriminator to learn some global kinematic features of the inputs. Additionally, the reconstructed $\hat{p}_T^{\mathrm{miss}}$ are compared with the initial input $p_{\mathrm{T}}^{\mathrm{miss}}$ to construct the additional term $\mathcal{L}_{aux}$ in the loss function, as described in Eq. 11.3.

Implementation details

Adam [212] is used for optimization with a batch size of 32, a learning rate of 1×10^{-4} for the generator and of 2×10^{-4} for the discriminator. For the conditional pGAN, we choose $\epsilon = 10$ GeV and $\alpha = 0.05$. We train on cropped sequences of variable length, forcing the discriminator to learn to distinguish even very short arrays, and make use of dropout in the discriminator. Batch normalization [230] is included in the discriminator but not in the generator. Hyperparameter optimization was performed on the learning rate of both networks. We find that Least-Squarse GAN [332] offers the most stable training, outperforming both Wasserstein GAN [333] and the original GAN implementation.

11.5 Experimental evaluation

GAN performance evaluation is a challenging task. Due to their unsupervised nature, domain and task-specific metrics are often required. Our evaluation technique is three-fold: (i) we plot distributions of relevant physical features and quantify the matching between the ground-truth (GT), `PYTHIA8 + DELPHES`, and our network; (ii) we make use of high-level global event-shape variables such as the transverse thrust and the $p_{\mathrm{T}}^{\mathrm{miss}}$; (iii) we evaluate the effect of using our proposed generation technique in a real analysis environment. To this purpose, we apply a state-of-the-art pileup removal algorithm (*SoftKiller* [334]) and cluster the remaining particles in the event, characterizing the agreement between real simulation and pGAN on jet kinematic properties (e.g., the jet p_T). These three different performance assessments are discussed in the rest of this section.

Histogram matching

Fig. 11.3 shows how the pGAN generators succeed in learning the main aspects of the particles' p_T, η and ϕ distributions. The comparison is limited to the first 50 high-p_T particles of each class, in order for the representation of the generated event to be consistent with the GAN generator output. We observe a remarkable agreement in p_T and ϕ: the long tail of the transverse momentum distribution is well described

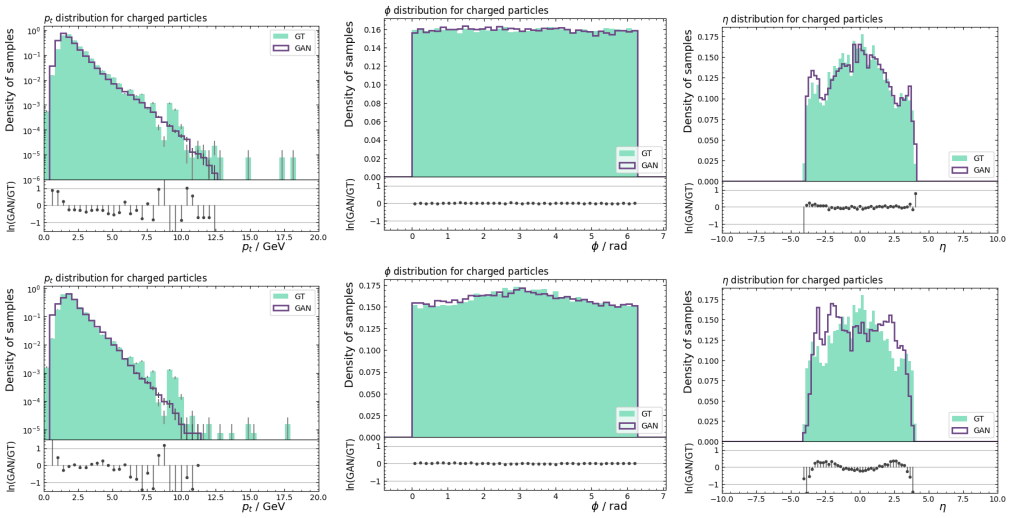
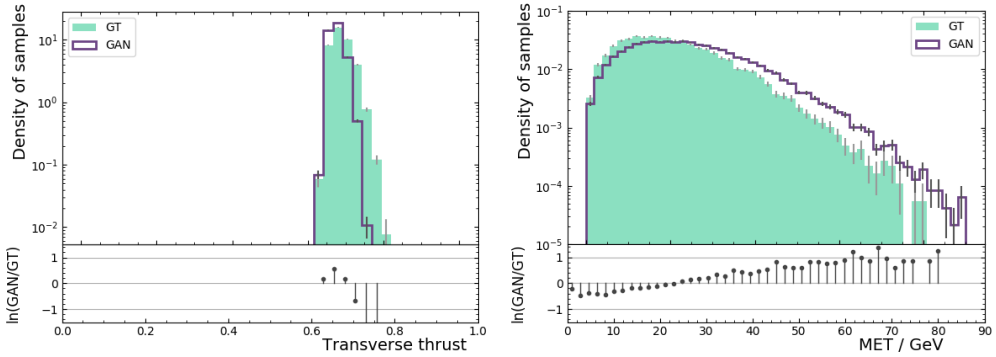


Figure 11.3: Comparison of the transverse momentum p_T (left), azimuth angle ϕ (center) and pseudorapidity η (right) for charged particles between the test data and the events generated by unconditional pGAN (top) and conditional pGAN (bottom). For the conditional pGAN, ϕ is transformed to be the azimuth angle between the particles' momenta and $\vec{p}_T^{\,\text{miss}}$.

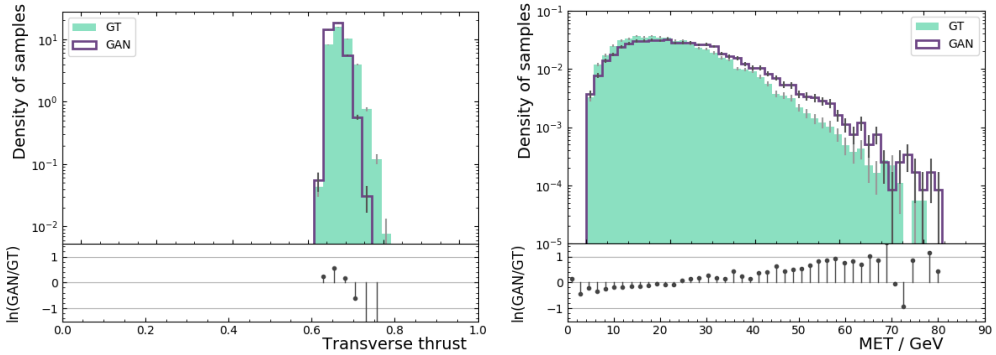
across four orders of magnitude and the ϕ rotation invariance of the physical process is reproduced. On the other hand, while the qualitative features of the pseudorapidity distribution are learned, the agreement is not completely satisfactory. A better match should be the goal of future work. Fig. 11.4 shows comparisons for some of the event-related high level features used in model evaluation: transverse thrust and $p_{\text{T}}^{\text{miss}}$. We observe some discrepancy being associated to overall scale shifts, related to the different truncation criteria applied to the two sets of events.

Wasserstein metric

We use the previously described histograms to quantify the difference between the target and generated distributions through the Wasserstein or Earth Mover's (EM) distance. The EM distance can be understood as the amount of "work" (probability density $\times$ distance) required to transform one distribution into the other. While other choices are possible (*e.g.*, the Kolgomorov-Smirnov test), the EM distance is usually more suited for long-tailed distributions such as in p_T and rewards local improvement better. We rescale the target and generated distributions of the various features so that they are all of the same order of magnitude, since the EM distance depends upon the scale choice.



(a) Unconditional pGAN



(b) Conditional pGAN

Figure 11.4: Comparison of the transverse thrust and $p_{\mathrm{T}}^{\mathrm{miss}}$ distributions between the test data and the generated events by pGANs.

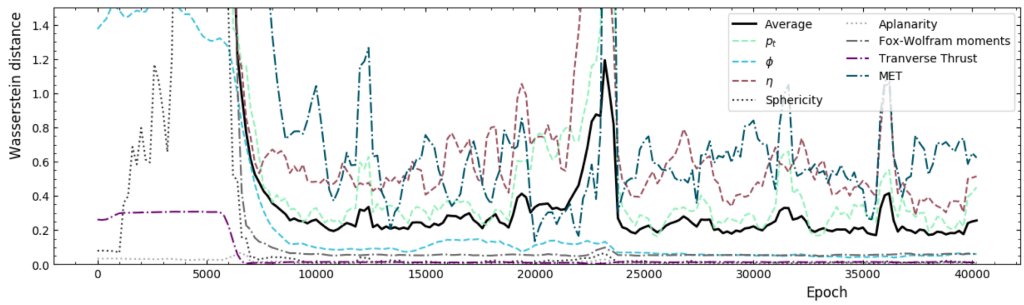


Figure 11.5: Evolution of our performance metric (solid black) as a function of training. EM distances for some of the individual quantities are superposed.

We define our performance metric as the weighted average of the EM distances over the feature distributions: p_T, η, ϕ for all three particle types, sphericity, transverse thrust, aplanarity, $p_{\mathrm{T}}^{\mathrm{miss}}$, and the first and second transverse Fox-Wolfram moments [335]. The use of this metric tackles the problem of lack of interpretability of the loss function: we observe that the metric decreases steadily as the training progresses, as shown in Fig. 11.5, providing a way of monitoring progress, performing early-stopping and tracking training failure. Based on this metric, we perform model comparison, hyperparameter tuning, and the final best-model choice.

Pileup subtraction

Typical LHC analyses are performed after applying a pileup removal algorithm, which aims to subtract soft radiation from QCD. It is then important to demonstrate that the pGAN is good in modeling the residual pileup contribution, after such a subtraction algorithm is applied. Since this residual pileup contribution is the only relevant effect for physics analyses, it is acceptable for a pileup emulation software to have a non-accurate pileup simulation as long as the pileup effect is well model after applying a pileup mitigation technique. For this purpose, we consider a sample of $Z \to \nu\overline{\nu}$ events, generated using `Pythia8`. Events are processed with `Delphes`, using the same setup adopted to generate the pileup reference sample, both with and without activating the pileup emulation at $\overline{n}_{\mathrm{PU}} = 20$. The generated no-pileup events are mixed with the pileup emulation returned by the generator. The *SoftKiller* [334] algorithm with a grid size of $a \approx 0.5$ is then applied to both these events and those with a full pileup simulation.

Table 11.1: Mean leading-jet p_T for events with no pileup and pileup generated by `Pythia8` and by the network (pGAN), before and after running *SoftKiller*.

	$\langle p_T \rangle$ / GeV		$\langle p_T \rangle$ / GeV
No PU	136.8		
Pileup—GT	146.6	Pileup—GT—subtracted	135.0
Pileup—pGAN	141.1	Pileup—pGAN—subtracted	135.7

Fig. 11.6 shows the p_T distribution of the highest-p_T jet in the event in various configurations. The main effect of pileup contamination is a shift in the p_T distribution towards larger values. The shift is underestimated when the pileup is described through the pGAN generator, which is due to the fact that pGAN only returns the first 150 particles per event, instead of the usual ~ 900. After processing the event with

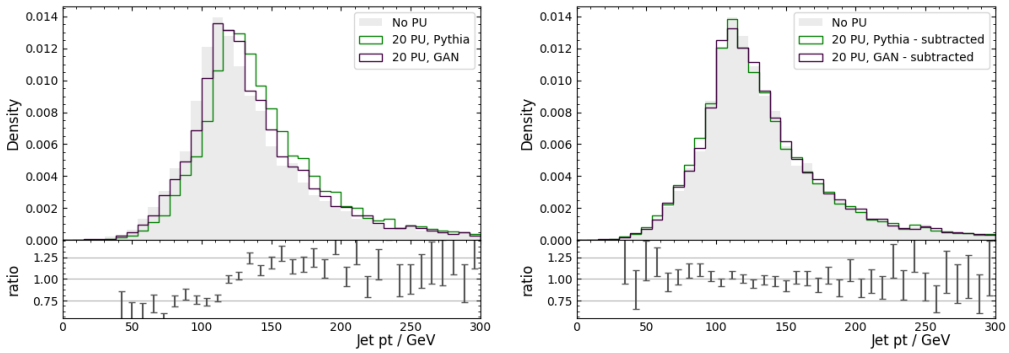


Figure 11.6: Comparison between leading jet p_T distributions for events with no pileup (solid black) and pileup generated by `Pythia8` (green) and by the network (magenta). Distributions are shown both before (left) and after (right) running the *SoftKiller* pileup mitigation algorithm. The bottom plot shows the Pythia/GAN ratio.

the *SoftKiller* algorithm, the leading jet p_T distribution for `Pythia8` and our GAN match within 0.7 GeV. The agreement could be further improved by increasing the number of neutral hadrons and charged particles returned by the pGAN.

Computational performance

We measure the average inference time with pGAN in a single thread of an `Intel® Xeon®` CPU E5-2650 v4 @ 2.20GHz to be 3.19 ms per event. The average event simulation time in CMS is $\sim O(100\,\text{s})$ per event [326]. This corresponds to an improvement factor of 10^5 in terms of computational resources with our pGAN solution.

11.6 Distributed training

Training a GAN is computationally expensive and benefits from supercomputing clusters with the coordination of multiple General-Purpose Graphics Processing Unit (GP-GPU) in different host machines. In this section, we present several ways for parallel computation of the gradients needed for training with Stochastic Gradient Descent (SGD) [300, 336, 337]. One can leverage these levels of parallelism on high performance computing (HPC) centers composed of many nodes with high bandwidth connectivity. The computation and communication is orchestrated using the MPI framework [338], abstracting the communication protocols from the computation. An MPI program is executed over multiple processes, running on multiple physical hosts on the cluster. Each process is called a worker, and it does not matter a priori if they get executed on the same physical node. Depending on the

topology of the HPC, there can be more than one GP-GPU per physical host, and we enforce to not get more than one process associated with one GP-GPU. Therefore, each worker in the following refers to a process with at least one dedicated GP-GPU attached.

In training a deep learning model, there are two main parallelism methods to consider:

- **Data parallelism**: Minibatches are split among multiple workers to compute the gradient evaluated on different fractions of the minibatches in parallel. Each worker keeps a copy of the model parameters. A master process will then collect the gradient results, compute the model parameter update, and then send the update back to the workers to update their models. In a synchronous setting, the master process waits until all worker processes finish their computations, then collect the result, compute the update, and broadcast the update to all workers at once. This synchronous method, while preserving the accuracy, suffers from high latency due to the inefficient use of workers with typically high idle time. In practice, parallel SGD is often done in an asynchronous fashion, where the master process receives information from each worker, computes the update, and then sends it back immediately to the worker without waiting for other workers to finish. Popular asynchronous training algorithms include `Downpour SGD` [339] and `Elastic Averaging SGD` [340].
- **Model parallelism**: For a gigantic model with billions of parameters, it is virtually impossible to store the whole model into a single GP-GPU's memory. Model parallelism is the process of splitting a model up between multiple devices and creating an efficient pipeline to train the model across these device to maximize GPU utilization [341]. The first step is to partition the model, either by modules or by types of operations, and then store each partition in one single device. The second step is to build an efficient pipeline to orchestrate the execution of the model training flow, including the forward computations and backward propagation, in different devices to maximize the utilization. In this study, we use the native functionality of `TensorFlow` to partition the computational graph into different devices [121].

11.7 Summary

We presented GAN models based on a recurrent unit, capable of generating lists of particles with variable lengths and also with an initial condition on $p_{\mathrm{T}}^{\mathrm{miss}}$. Such a model could be used for particle-based simulation software, such as those of experiments using particle-flow reconstruction algorithms. This model could be used to replace ordinary rule-based algorithms in specific aspects of jet generation. In this paper, we show its application to pileup emulation in LHC collisions. While technical limitations forced us to reduce the length of the returned particle chain, the network is capable of emulating the effect of pileup on a realistic data analysis, after applying a pileup mitigation algorithm, reducing the computational resource for event simulation by five orders of magnitude.

Chapter 12

GENERATIVE MODELS FOR ANALYSIS-SPECIFIC FAST SIMULATION

We present a fast-simulation application based on a Deep Neural Network, designed to create large analysis-specific datasets. Taking as an example the generation of W+jet events produced in $\sqrt{s}$ = 13 TeV proton-proton collisions, we train a neural network to model detector resolution effects as a transfer function acting on an analysis-specific set of relevant features, computed at generation level, i.e., in absence of detector effects. Based on this model, we propose a novel fast-simulation workflow that starts from a large amount of generator-level events to deliver large analysis-specific samples. The adoption of this approach would result in about an order-of-magnitude reduction in computing and storage requirements for the collision simulation workflow. This strategy could help the high energy physics community to face the computing challenges of the future High-Luminosity LHC.

12.1 Introduction

At the CERN Large Hadron Collider (LHC), high-energy proton-proton (pp) collisions are studied to consolidate our understanding of physics at the energy frontier and possibly to search for new phenomena. While these studies are typically conducted according to a *data driven* methodology, synthetic data from simulated pp collisions are a key ingredient to a robust analysis development. Particle physicists rely extensively on an accurate simulation of the physics processes under study, including a detailed description of the response of their detector to a given set of incoming particles. These large sets of synthetic data are typically generated with experiment-specific simulation software, based on the `GEANT4` [111] library. Through Monte Carlo techniques, `GEANT4` provides the state of the art in terms of simulation accuracy. The first two runs of the LHC highlighted the remarkable agreement between data and simulation, with discrepancies observed at the level of a few percent. On the other hand, running `GEANT4` is demanding in terms of resources. As a consequence of this, delivering synthetic data at the pace at which the LHC delivers real data is one of the most challenging tasks for the computing infrastructures of the LHC experiments. It is then more and more common for LHC physics analyses to be affected by large systematic uncertainties due to the limited

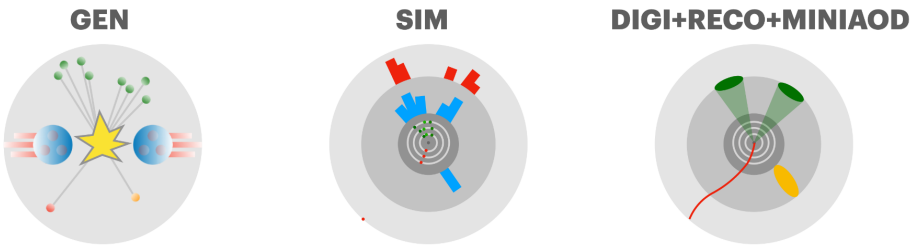


Figure 12.1: The event generation workflow of the CMS experiment. The pp collision process is simulated up to the production of stable (hence observable) particles (GEN). The simulation of the detector response is modelled by the `GEANT4` library (SIM). The resulting energy deposits are turned into digital signals (DIGI) that are then reconstructed by the same software used to process real collision events (RECO). At this stage, high-level objects such as jets are reconstructed. Starting from the RECO data format, a reduced analysis data format (MINIAOD) is derived.

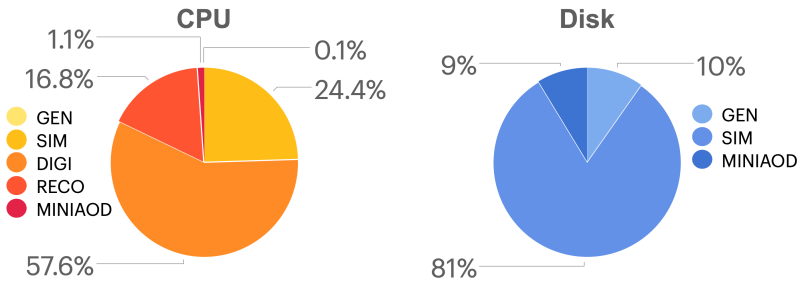


Figure 12.2: Computing resource breakdown for the generation workflow of the CMS experiment, in terms of CPU (left) and storage disk (right). See Sec. 12.6 for details.

amount of simulated data. This is particularly true for precise measurements of Standard Model processes for which large datasets are already available today. In the future, with the high-luminosity LHC upgrade, this will become a serious problem for most of the LHC data analyses [342]. Our community is called to reduce the computing resources needed for central simulation workflows by at least one order of magnitude, not to jeopardize the accuracy gain expected when operating the LHC at a high luminosity.

To give a concrete example, we consider the event simulation workflow of the CMS experiment, schematically represented in Fig. 12.1. The first step (GEN) consists in running an event generator library, simulating a pp collision, the production of high-mass particles from it, and the decay of these particles to those stable particles which are then seen by the detector. This step creates the so-called generator-

level view of a collision event, corresponding to what a perfect detector would see. The simulation of the detector response (SIM) translates this flow of particles into a set of *detector hits*, taking into account detector imperfections and the limited experimental resolution. These hits are converted to the same digital format (DIGI) produced by the detector electronics and then reconstructed by the same software used to process real collision events (RECO). At this stage, high-level objects such as jets are created. Starting from the RECO data format, a reduced analysis data format (MINIAOD) is derived [343]. Figure 12.2 provides a breakdown of CPU and disk resources for each of these steps. Details on the procedure followed to measure these values are given in Sec. 12.6.

Recently, generative algorithms based on Deep Learning (DL) techniques have been proposed as a possible solution to speed up `GEANT4`. When following this approach, one typically focuses on an image representation of LHC collisions (e.g., energy deposits in a calorimeter) and develops some kind of generative model [251, 285, 333, 344, 345] to by-pass `GEANT4` when simulating the detector response to individual particles [318, 320, 346–348] or to groups of particles, such as jets [321, 323, 349] or cosmic rays [322]. Generative models were considered also for similar applications in HEP, such as amplitude [350] and full event topology [326, 351, 352] generation. While these studies demonstrate the potential of generative models for HEP, more work is needed to fully integrate this new methodology in the centralized computing system of a typical LHC experiment. In particular, one needs to work beyond the collision-as-image paradigm so that the DL-based simulation accounts for the irregular geometry of a typical detector while delivering a dataset in a format compatible with downstream reconstruction software.

Other studies [327, 353, 354] investigated a more extreme approach: rather than training models to perform generic generation tasks in a broader software framework (e.g., a DL-based shower generator in GEANT), one could design analysis-specific generators, with the limited scope of delivering arrays of values for physics quantities which are relevant to a specific analysis. Reducing the event representation to a vector of meaningful quantities, one could obtain a large amount of events in short time and with small storage requirements by skipping all the intermediate steps of the data processing. The considered features could be the fundamental quantities used by a given analysis (e.g., the four-momenta of the final-state reconstructed objects in a search for new particles). In this context, both generative adversarial networks (GANs) [327, 353] and variational autoencoders (VAEs) [353] were considered.

In this case, one learns the N-dimensional probability density function (N-dim pdf) of the event, in a space defined by the quantities of interest for a given analysis. Sampling from this function, one can then generate new data. The open question with this approach stands with the trade-off between statistical precision (which decreases with the increase amount of generated events) and the systematic uncertainty that could be induced by a non accurate description of the N-dim pdf. When training both VAEs and GANs, one learns how to interpolate between the samples provided in the training dataset. The limited amount of data in the training dataset is the ultimate precision-limiting factor, as discussed in Ref. [355], but generative models retain amplification capability similarly to what a fitting function does, as shown in Ref. [356] for GANs. Ultimately, one needs to balance the statistical uncertainty (i.e., the amplification factor when augmenting the dataset) and systematic uncertainties associated to the accuracy with which the generative model interpolates between the training data points. The balance will be reached tuning, among other things, the training dataset size. The optimal configuration, intrinsically application specific, determines whether a generative model is computationally convenient.[1]

In this paper, we propose to rephrase the problem of analysis specific dataset generation. Rather than morphing a distribution in a latent space into a target distribution, we want to start from the ideal-detector distribution and morph it into the actual-detector distribution, learning a fast-and-accurate detector response model. We do so combining the strength of multi dimensional deep neural regressors to the adaptive power of kernel density estimation, which has a long and successful tradition in particle physics [357]. A similar goal is presented in [354] in which invertible neural networks are utilized with a focus on being able to perform unfolding (morphing from reconstructed level information to generator level distributions). For a given physics study, we assume that the interesting features can be represented by a limited set of high-level quantities (the feature vector $\vec{x}$). We assume that a training dataset is provided. For each collision event in the dataset, the feature vector is computed at three stages: (i) *at generator level* $\vec{x}_G$, i.e., before applying any detector simulation. This view of the collision event corresponds to the perfect-resolution ideal detector case; (ii) *at reconstruction level* $\vec{x}_R$, i.e. after the simulation of the

[1] Here, we are assuming that `GEANT4` will be used to generate the training dataset and the generative model will then be used to scale up the simulated dataset size. If the desired accuracy can be reached only at the price of more training data to be generated, the net gain of this approach would be reduced.

detector response, modelled with GEANT4; (iii) at the output of the DL model $\vec{x}_{DL}$ [2]. We model the detector response as a function of the generator-level feature vector:

$$x^i_{DL} = \mathcal{N}(\mu^i_R(\vec{x}_G), \sigma^i_R(\vec{x}_G)) \ , \tag{12.1}$$

where $\mathcal{N}(\mu, \sigma)$ is a one-dimension Normal function centered at μ with variance σ^2 and the index i runs over the components of the feature vector $\vec{x}$. We train a DL model to simultaneously learn the functions $\vec{\mu}_R(\vec{x}_G)$ and $\vec{\sigma}_R(\vec{x}_G)$, and then use the Normal model of Eq. (12.1) to generate $\vec{x}_{DL}$ from $\vec{x}_G$. Under the assumption that large sets of $\vec{x}_G$ values can be obtained in relatively short time (which is typically the case for High Energy Physics applications), this strategy would result in a sizable save of computing resources. On one hand, one would reduce computing time bypassing the more intense steps of the generation workflow. In addition, one would reduce the need for large storage elements: rather than storing individual collision data, which demands an event storage allocation between $\mathcal{O}$(1MB) (for raw data) and $\mathcal{O}$(10kB) (for analysis-ready object collections), one would directly handle a few relevant quantities for a given analysis. One could save resources by utilizing analysis-specific fast simulation models for data augmentation, e.g., generating 10% of the required data with the traditional GEANT4 workflow and the remaining 90% only up to the GEN step. These data, shared among the $\mathcal{O}(100)$ analyses, would be used to create analysis-specific training and inference datasets. Even considering that $\mathcal{O}(100)$ analysis teams would have to train $\mathcal{O}(100)$ specific generative models, the strategy we propose would result in an important resource gain, provided a large enough training facility.[3]

We demonstrate this strategy at work on a concrete example, namely the generation of W+1 jet events produced in $\sqrt{s}$ = 13 TeV pp collisions, similar to those recorded at the LHC. We discuss the model design and training, its performance and its accuracy for factor-ten data augmentation.

This paper is structured as follows: Section 12.2 provides a full description of the input dataset and its feature-vector representation. Section 12.3 describes the model architecture and the training setup. Sections 12.4 and 12.5 discuss the model

[2] Given the limited computing resources at hand, it was not possible to carry on this study on a GEANT4-based dataset. Instead, we used the DELPHES [269], which provides a realistic setup to demonstrate the proposed strategy.

[3] The model presented in this work was trained on a RTX2080 GPU by NVIDIA in 30 minutes. Even a small-size GPU cluster with O(10) GPUs dedicated to this use case could then serve the needs of a large collaboration. Its cost is negligible on the scale of the large computing infrastructures built for the LHC experiments.

performance in terms of accuracy and resource utilization, respectively. Conclusions and outlook are given in section 12.9.

12.2 Benchmark dataset

As a benchmark problem, we consider the generation of W+1 jet events produced in $\sqrt{s}$ = 13 TeV pp collisions. The starting point is the inclusive production of $W \to \mu\nu$ events using `PYTHIA8` [103]. At this stage, we require each event to have at least one muon with a transverse momentum $p_T > 22$ GeV.[4] Detector effects are modelled using `DELPHES` v3.4.2 [269]. We consider the CMS detector model for the HL-LHC upgrade, distributed with `DELPHES`. At this stage, the event is overlaid to minimum-bias events to model the effect of pileup, i.e., those parasitic pp collisions happening at the same beam crossing as the interesting event. For each collision, the number of pileup collisions is sampled from a Poisson distribution with expectation value set at 200, in order to match the expected conditions for HL-LHC.

At generator level (GEN), jets are clustered using the `anti-kt` algorithm [57] with jet-size parameter $R = 0.5$, taking the four-momenta of all the stable particles in the event as input. We consider events with one clustered jet, with $p_T > 30$ GeV and $|\eta| < 2.4$. In order to avoid the double counting of muons as jets, we require $\Delta R = \sqrt{\Delta\eta^2 + \Delta\phi^2} > 0.5$ between the muon and the jet in each event.

At reconstruction level, jets are clustered from the list of particles returned by the `DELPHES` particle-flow algorithm. As for the GEN jets, we consider `anti-kt` jets with $R = 0.5$. Both the muon and jet are matched to the corresponding generator-level object, selecting the reconstructed object (e.g., a muon) with the smallest ΔR from the corresponding generator-level object. Since our final state is composed of one jet and one muon, this simple algorithm does not generate ambiguity in the association. When generalizing this approach to more complex event topologies, one might modify the matching algorithm to prevent that the same gen-level object is associated to multiple reconstructed objects. In addition, we discard events with mismatched muons by requiring that the relative residual of the muon p_T to be $|p_T^G - p_T^R|/p_T^G < 10\%$. This requirement allows us to remove a small fraction of events ($\sim 0.5\%$ of the total) in which the muon from the W boson is not reconstructed

[4] We use a Cartesian coordinate system with the z axis oriented along the beam axis, the x axis on the horizontal plane, and the y axis oriented upward. The x and y axes define the transverse plane, while the z axis identifies the longitudinal direction. The azimuth angle ϕ is computed with respect to the x axis. The polar angle θ is used to compute the pseudorapidity $\eta = -\log(\tan(\theta/2))$. The transverse momentum (p_T) is the projection of the particle momentum on the (x, y) plane. We fix units such that $c = \hbar = 1$.

but another muon is found. In DELPHES, inefficiency in muon reconstruction happens through an uncorrelated *hit-or-miss* procedure based on pseudo-random numbers. Working in an experimental environment, one would retain the whole dataset from a more accurate simulation, based on specific physic requirements that would induce learnable correlations.

The feature vector $\vec{x}$ is built considering the following nine quantities:

- The muon momentum in Cartesian coordinates: p_x^μ, p_y^μ, and p_z^μ.
- The jet momentum in Cartesian coordinates: p_x^j, p_y^j, and p_z^j.
- The logarithm of the jet mass $\log(M_j)$.
- The missing transverse energy in Cartesian coordinates: E_x^{miss} and E_y^{miss}.

In addition, we consider a set of 12 auxiliary features, computed from the input feature vector $\vec{x}$:

- The muon momentum in longitudinal-boost-invariant coordinates: p_T^μ, η^μ, and ϕ^μ.
- The jet momentum in longitudinal-boost-invariant coordinates: p_T^j, η^j, and ϕ^j.
- The missing transverse energy in polar coordinates: E_T^{miss} and ϕ_{miss}.
- The transverse mass M_T, i.e., the mass of the four momentum obtained summing the the muon transverse momentum $(E_T^\mu, p_x^\mu, p_y^\mu, 0)$ to the missing transverse energy $(E_T^{\text{miss}}, E_x^{\text{miss}}, E_y^{\text{miss}}, 0)$.
- S_T, i.e., the scalar sum of E_T^{miss}, p_T^μ, and p_T^j.
- The jet mass: M_j.

These quantities are computed at generator and reconstruction level and are used to assess how well the correlation between the generated quantities is modeled. Unlike the feature-vector quantities, they do not enter the definition of the loss function.

The model training and performance assessment is done on a dataset of 2M events, which we separate in a test and a learning datasets, containing 20% and 80% of the

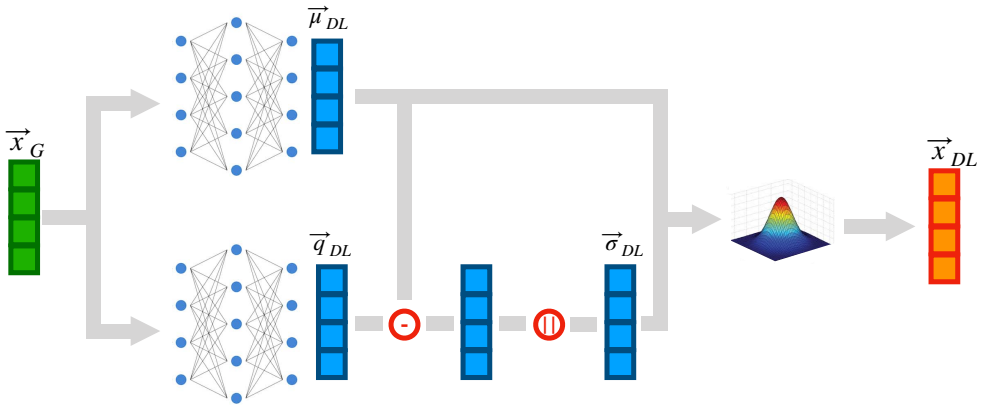


Figure 12.3: Model architecture: a feature vector at generator level $\vec{x}_G$ is given as input to two regression models, returning vectors of central values ($\vec{\mu}_{DL}$) and RMS ($\vec{\sigma}_{DL}$), from which the reconstructed feature vector predicted by the DL model $\vec{x}_{DL}$ is generated.

events, respectively. The learning dataset is further split into a training (70%) and a validation (30%) dataset. In order to test the data augmentation properties of the proposed strategy, we also consider a larger test dataset, containing 10M events.

Both the training and large-size testing datasets are published on Zenodo [358, 359].

12.3 Model description and training

Our model architecture is represented in Fig. 12.3. The input vector $\vec{x}_G$ of generator-level features is passed to two regressive models, each returning a vector with the same dimensionality of $\vec{x}_G$. One is interpreted as a vector of mean values $\vec{\mu}_{DL}$. The other one is interpreted as the "$\pm 1\sigma$" quantile $\vec{q}_{DL}$. By taking the absolute difference between each mean value and its corresponding quantile, we compute the RMS values $\vec{\sigma}_{DL}$.

Each regressive model consists of a six-layer dense neural network. The first and last layers have nine nodes each, while the intermediate layers have 100 nodes. All layers except the last one are activated by LeakyReLU [291] functions, with $\alpha = 0.05$. Linear activation functions are used for the last layer. The model output is then computed as $\vec{x}_{DL} = \vec{\mu}_{DL} + \vec{\sigma}_{DL} \cdot \vec{\epsilon}$, where the vector $\vec{\epsilon}$ contains random numbers sampled from a Normal function centered at 0 with unit variance. In addition to the main features $\vec{x}$, we compute a set of auxiliary features (see section 12.2) used for a further post-training validation.

The loss function is defined as the sum of a mean absolute error on $\vec{\mu}_{DL}$ and a quantile regression on $\vec{q}_{DL}$:

$$\mathcal{L}_{RECO} = \left\langle \left\| \vec{\mu}_{DL} - \vec{x}_R \right\|_1 + QR(\vec{q}_{DL}, \vec{x}_R) \right\rangle \ , \tag{12.2}$$

where the average is done over a training subset, and the quantile regression loss QR is defined as:

$$QR(\vec{x}, \vec{y}) = \sum_{i=1}^{k} \Theta(x_i, y_i)|x_i - y_i| \ . \tag{12.3}$$

where

$$\Theta(x, y) = (1 - \gamma)\theta(x - y) + \gamma\theta(y - x) \ . \tag{12.4}$$

The step function $\theta(t)$ is set to one (zero) for positive (negative) values of t and $\gamma = 0.841$. This choice of γ guarantees that the loss is minimized to learn the quantile corresponding to one standard deviation.

We implement the model in `KERAS` [120] and train it with the Adam [212] optimizer, with batches of 128 and an epoch-dependent learning rate $lr = 0.001/(1 + n_{\text{epoch}})$. The model is trained for 100 epochs, but convergence is typically reached between 30 epochs. The network parameter values corresponding to the smallest validation loss are taken as the optimal configuration.

12.4 Results

The trained model is used to generate samples of reconstructed events from generator-level events. We evaluate the training performance by comparing the output distributions with those obtained by `DELPHES` for the same generator-level events.

A comparison is shown in Fig. 12.4 for the feature-vector quantities. The sample derived from the DL model is similar to the the one obtained running a classic generation workflow. We train the model ten times and produce ten distributions. The bin-by-bin spread of these distributions is considered as a systematic uncertainty associated to the DL model, which is summed in quadrature to the statistical uncertainty in the same bin to compute the total uncertainty, shown by the error bars of the DL model in the figure. These systematic uncertainties are included to all the DL distributions shown in this paper. Only the statistical uncertainty is shown for the corresponding distributions of reconstructed quantities.

The model can account for small perturbations and major distortions of the GEN distribution, as well as the default detector simulation workflow. The agreement is

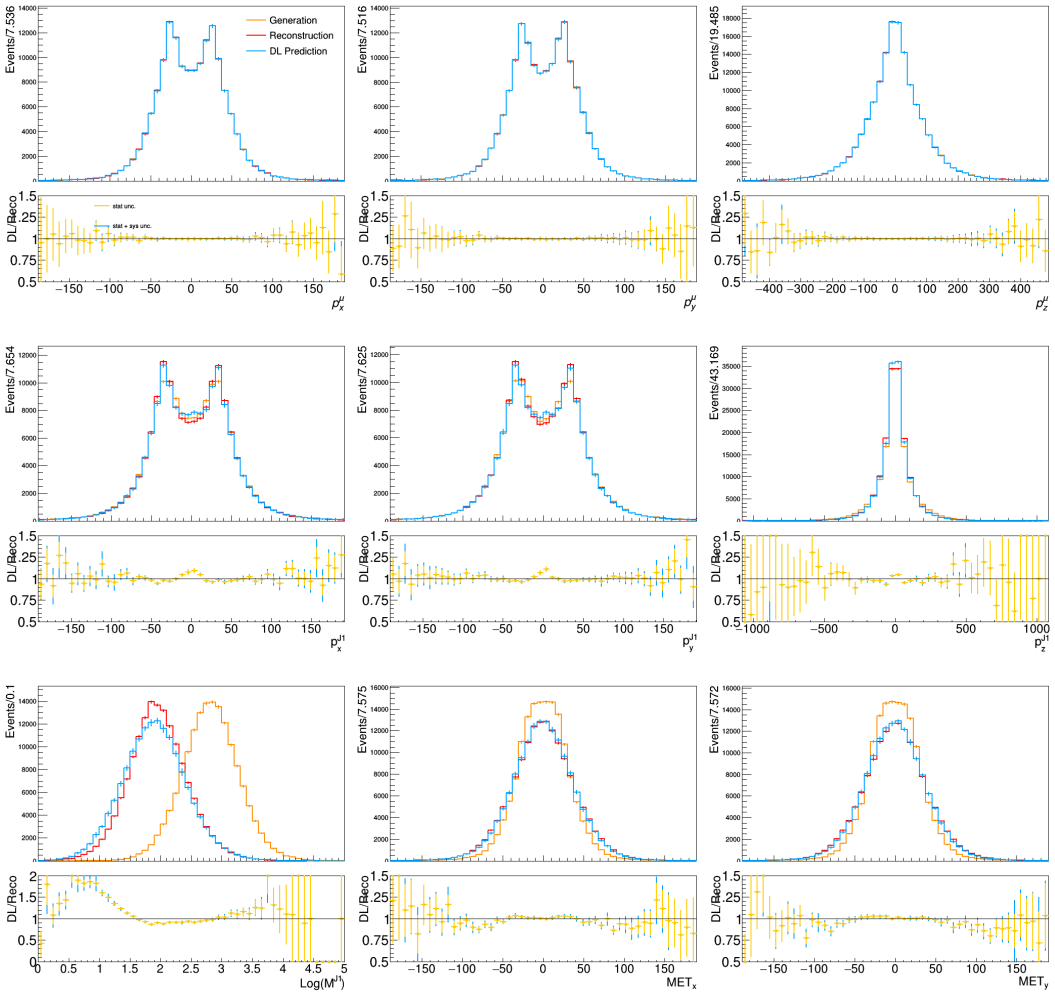


Figure 12.4: Distribution of reconstructed and model-predicted quantities for the feature-vector quantities, compared to the corresponding quantities from generator-level quantities provided as input to the model. The bottom panel below each plot shows the bin-by-bin ratio of the model-predicted over reconstructed distribution for each quantity, labelled DL/Reco. The error bars on the model-predicted quantities is composed of the statistical uncertainty and systematic uncertainty associated with model training, represented by the different colors.

not perfect, and certainly the model can be improved. Nevertheless, the reached accuracy is comparable to that of a typical data-to-simulation comparison and certainly sufficient to support the novel procedure that we want to put forward in this study. The observed agreement goes beyond one-dimensional projections of the input features. The distributions of auxiliary quantities, computed as a function of the feature-vector quantities, are also modelled to a good precision (see Fig. 12.5).

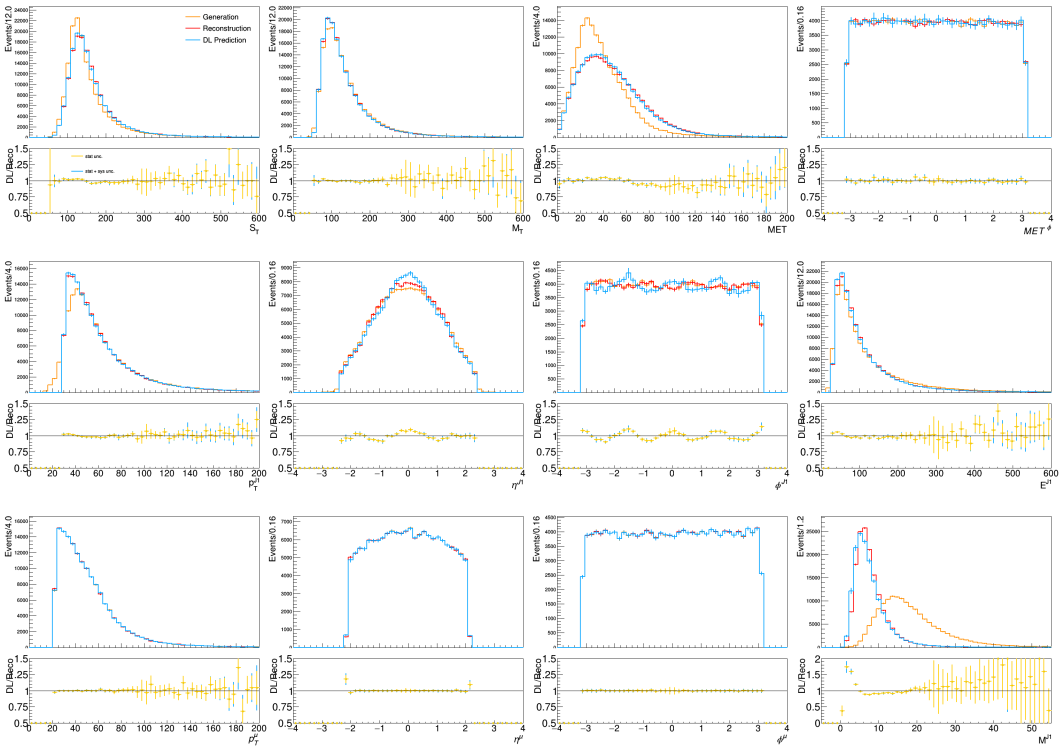


Figure 12.5: Distribution of reconstructed and model-predicted auxiliary quantities, compared to the corresponding generator-level quantities. The bottom panel below each plot shows the bin-by-bin ratio of the model-predicted over reconstructed distribution for each quantity, labelled DL/Reco. The error bars on the model-predicted quantities is composed of the statistical uncertainty and systematic uncertainty associated with model training, represented by the different colors.

This demonstrates that the DL-based generator accounts for correlations between quantities, as much as the traditional DELPHES workflow does.

While a comparison of dataset distribution gives a confidence of the quality achieved by the DL model, one can further test the achieved precision by looking at relative residual distributions. Our DL model does not sample events from a latent space (like a GAN or a plain VAE). Instead, it works as a fast simulation of a given generator-level event, preserving the correspondence between the reconstructed and the generated event, which allows us to compare event-by-event relative residual distributions. These distributions, which quantify the detector effects on the analysis-specific interesting quantities, are shown in Fig. 12.6. There, we compare the relative residuals between reconstructed and generated quantities, for the DL-based and the traditional simulation workflow. An overall agreement is observed, despite a bias on the muon and jet momentum coordinates. While the distribution ratio shown in

the bottom panel tends to magnify the effect that the distribution shift has on the tails, this residual difference between the target and learned resolution model has little impact on the simulation quality downstream, as one could judge by looking at the corresponding distributions in Fig. 12.4.

Figure 12.7 shows the same comparison for the auxiliary quantities. As the plot shows, a correct modeling of the residuals is obtained for energies, masses, and momenta. On the other hand, the model struggles to account for the high-resolution detector response on the η and ϕ coordinates. While this has little impact on the modeling of the ϕ and η distributions (see Fig. 12.4), this is certainly an aspect to improve in real-life applications. Deeper models on larger training data could learn the function better. In addition, one could modify the loss function to force the network to learn specific auxiliary quantities (e.g., the jet mass) with critic networks (as done in the context of GAN training) and explore non-Gaussian response functions. To this extent, working in Cartesian coordinates might be a better choice, in order to facilitate the calculation of the auxiliary quantities in the loss function. We did not expand our study in these directions, for which a target dataset based on a full detector simulation would be more appropriate.

Sec. 12.7 provides further assessments of the generation quality, showing 2D distributions of quantities derived from the DL-based generator vs the traditional one.

While our method relies on a Gaussian smearing function, it could be generalized to more complex functions if needed. In that case, one would have to learn more quantiles to model response functions with more than two parameters and then express these parameters as a function of the learned quantiles. On the other hand, it should be stressed that the response functions learned by our method are the result a convolution of the $\vec{\mu}$ and $\vec{\sigma}$ distribution (approximated by the Neural Network) and the Gaussian sampling function. Since the former is typically described by a non-Gaussian distribution, our model can learn non-Gaussian detector response even when relying on a simple Gaussian sampling. This is the case, for instance, of the asymmetric tail of the $p_{\mathrm{T}}^{\mathrm{miss}}$ residual distribution or the ϕ_{miss} double-peak structure shown in Fig. 12.7. On a practical side, a Gaussian sampling was adequate for this study, based on `DELPHES` data, but one might have to consider more complex sampling functions when trying to emulate with `GEANT`-based simulation.

In order to test the scaling of model accuracy with the inference dataset size, we apply our DL-based fast simulation strategy to a dataset five times bigger than what used for training. Figures 12.8 and 12.9 show the comparison of the distributions

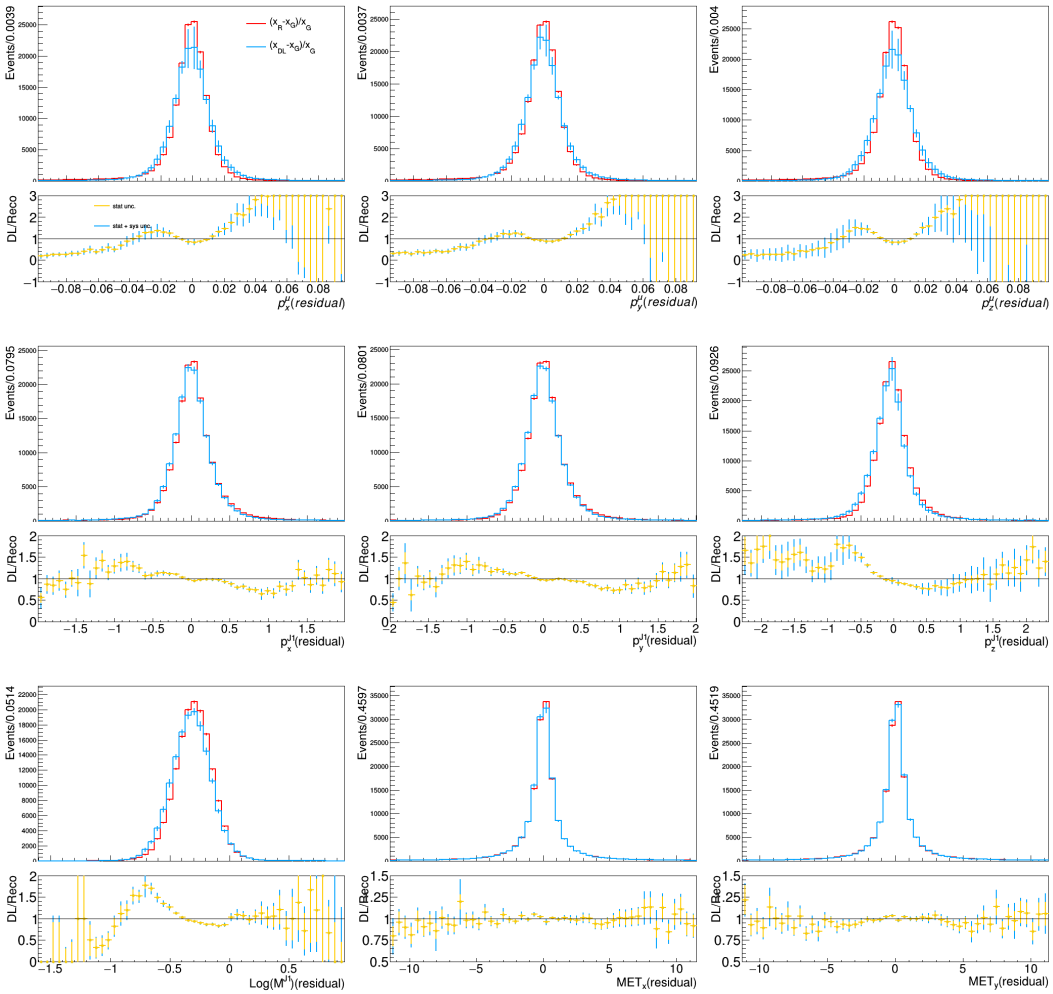


Figure 12.6: Relative residual distribution for reconstructed and model-predicted quantities in the feature vector, computing with respect to the reference input. The bottom panel of each plot shows the ratio between the two relative residuals, expected to be consistent with 1 for a DL model which correctly models the detector response of the traditional workflow. The error bars on the model-predicted quantities is composed of the statistical uncertainty and systematic uncertainty associated with model training, represented by the different colors.

obtained in this case, compared to what is obtained with DELPHES, respectively, for the input vector and the auxiliary features. The corresponding relative residual distributions are shown in Sec. 12.8. Figure 12.10 shows the differential double ratio distribution (high-statistics over low-statistics) for the reco-to-DL ratios. In presence of a systematic effect masked at low statistics, the reduction of the uncertainty in the high-statistics sample would unveil the problem. Instead we do observe flat double ratios, i.e. a similar behavior of the DL model for the small and the large sample.

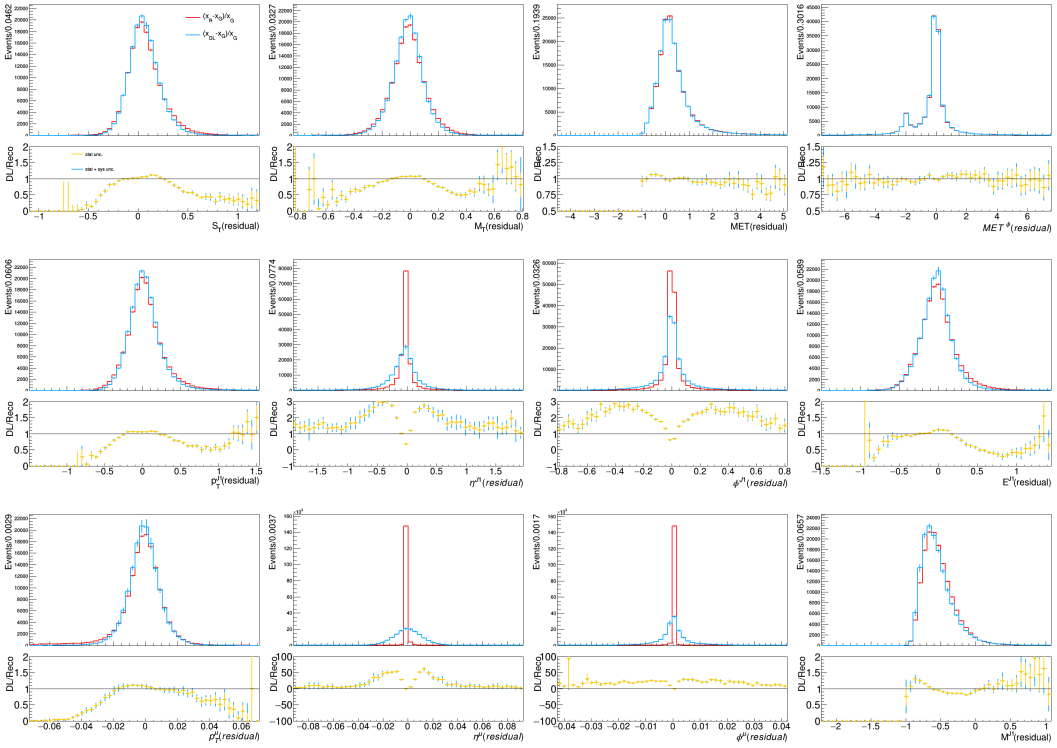


Figure 12.7: Relative residual distribution for reconstructed and model-predicted auxiliary quantities, computing with respect to the reference input. The bottom panel of each plot shows the ratio between the two relative residuals, expected to be consistent with 1 for a DL model which correctly models the detector response of the traditional workflow. The error bars on the model-predicted quantities is composed of the statistical uncertainty and systematic uncertainty associated with model training, represented by the different colors.

In view of this empirical observation, we are confident that the DL model accuracy would scale at much larger dataset size than what is used for training.

These distributions agree with those obtained when the training and inference dataset size agree, i.e., no accuracy deterioration is observed due to the scaling of the dataset size. This fact suggests that the proposed methodology scales adequately with the inference dataset size.

12.5 Computing resources

In order to fully assess the advantage of the proposed generation workflow, we consider the following use case: an analysis team requests N events to be centrally produced by the central computing infrastructure of their experimental collaboration. Instead, the central system would deliver N events at generator level (GEN step of

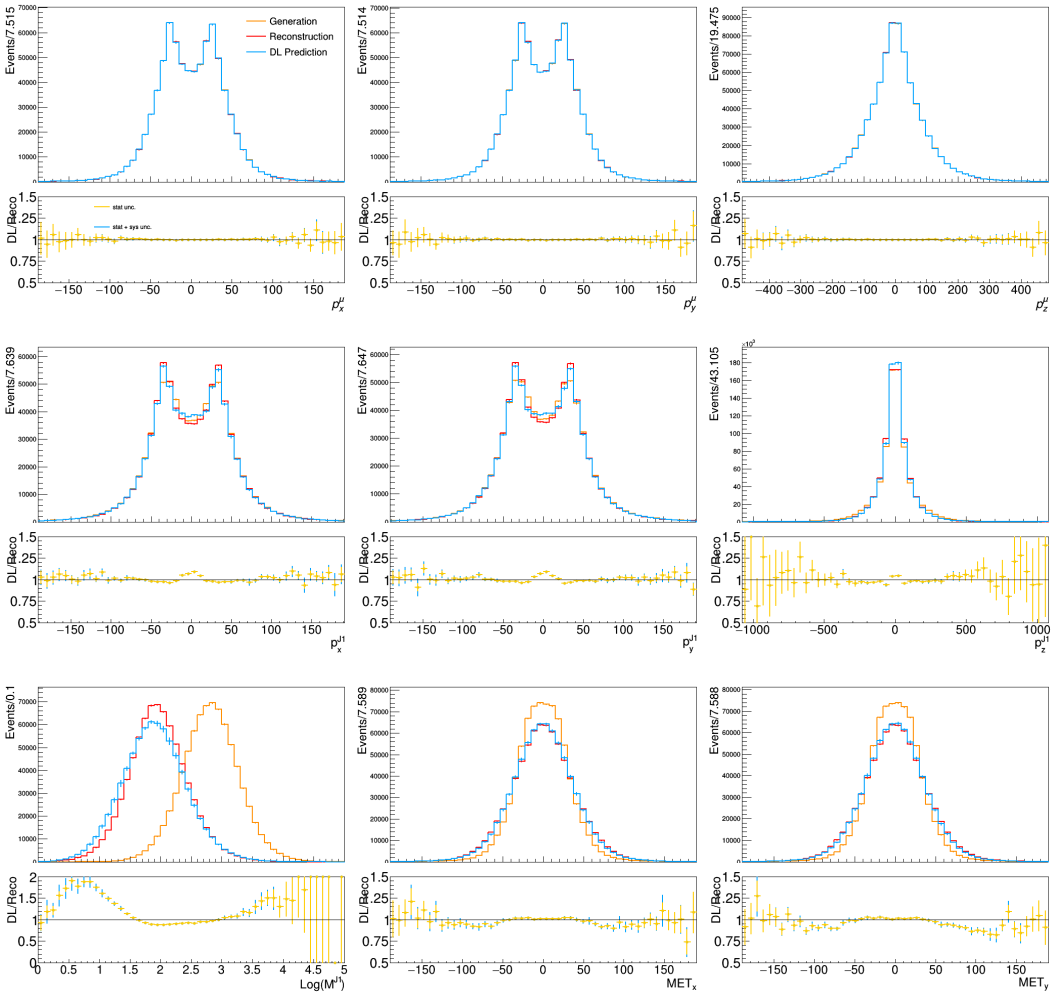


Figure 12.8: Distribution of reconstructed and model-predicted quantities for the feature-vector quantities, compared to the corresponding quantities from generator-level input. In this case, the model is applied to a dataset five times larger than the training dataset. The error bars on the model-predicted quantities is composed of the statistical uncertainty and systematic uncertainty associated with model training, represented by the different colors.

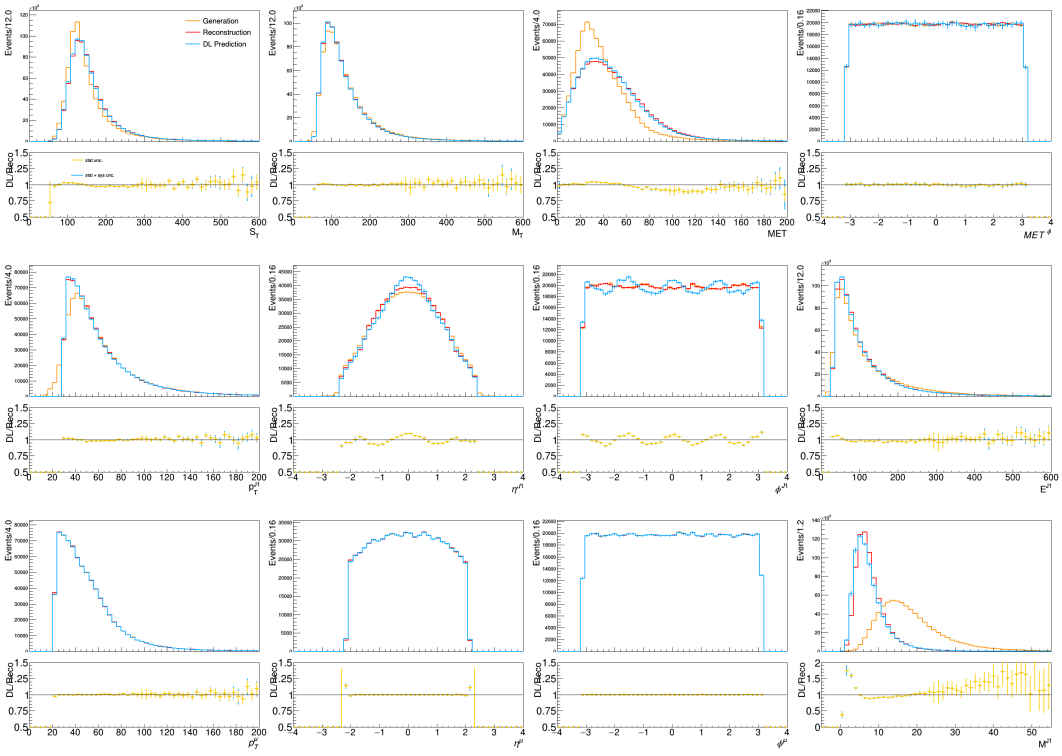


Figure 12.9: Distribution of reconstructed and model-predicted quantities in the auxiliary quantities, compared to the corresponding quantities from generator-level input. In this case, the model is applied to a dataset five times larger than the training dataset. The error bars on the model-predicted quantities is composed of the statistical uncertainty and systematic uncertainty associated with model training, represented by the different colors.

Fig. 12.1), while processing only $n < N$ of them through the full chain. The analysis team would then (i) run their data analysis software on the n events, and (ii) train on these data a DL-based fast-simulation like the one presented in Section 12.3. With this model, they would then (iii) process the other $(N - n)$ generator-level events and produce the dataset required for their analysis.

In order to assess the resource savings, we point out that step (iii) comes with negligible computational costs. Model inference on a CPU requires 100 sec to run on 100000 events (i.e., $O(1)$ msec/event), which results in a 8 MB file (saved as a compressed HDF5 file) for the example use case we discussed. While these details would change depending on the analysis-specific event representation, the quoted values give a reasonable order-of-magnitude estimate of the expected resource needs. Step (ii) can run at a minimal cost: our model could train within 30 minutes when running on a commercial GPU. The residual cost is then entirely driven by step (i).

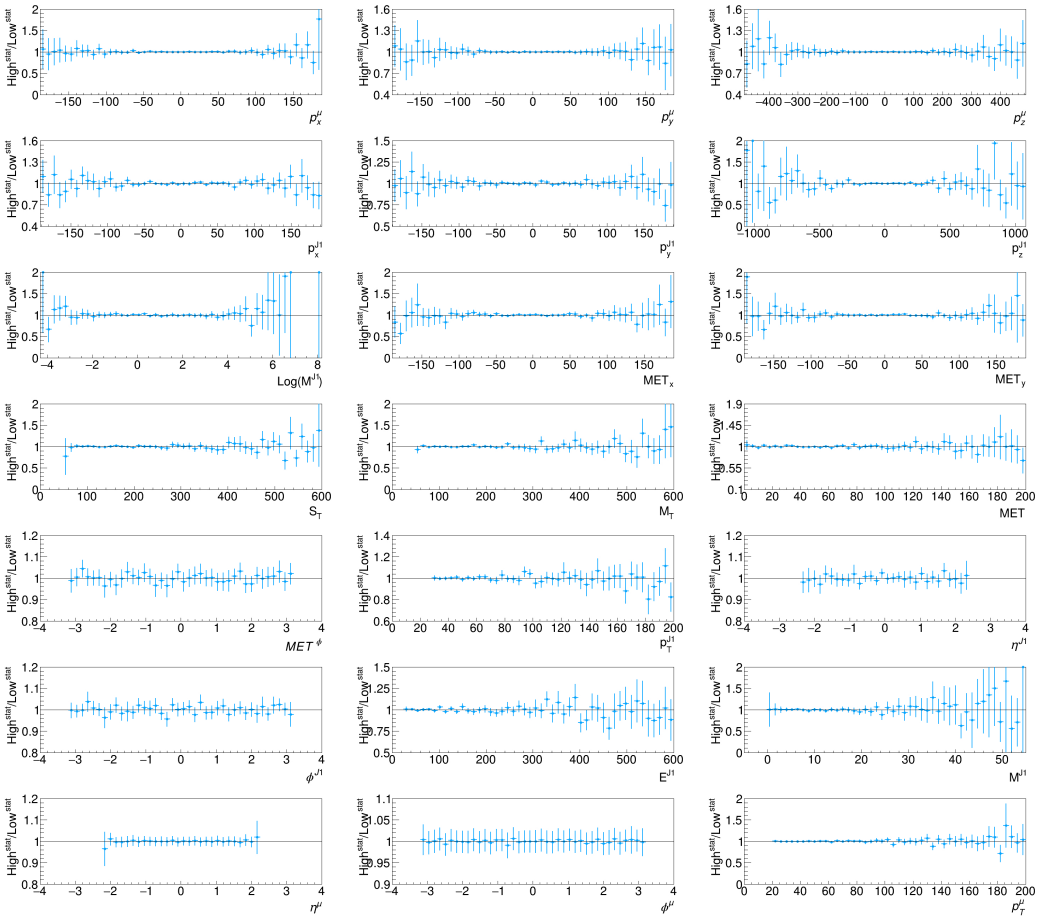


Figure 12.10: Differential double ratio distribution (high-statistics over low-statistics) for the reco-to-DL ratios shown in Figs. 12.4 and 12.8 and in Fig. 12.5 and Fig. 12.9.

While a traditional workflow requires O(100) sec/event of CPU time and occupies O(1) MB/event of storage, producing the same statistics (N events) of GEN-only events would require 10% disk allocation with a negligible CPU cost, as shown in Fig. 12.2.

As a consequence, by adopting the strategy outlined above, one would save a factor N/n in CPU (i.e., only spend sizable CPU resources to produce the training dataset, which would remain generic and could serve more than one analysis). The storage allocation would result from the sum of n events in full format and $(N-n)$ GEN-only events, for a total saving of $N/(n+10\%(N-n))$. For instance, considering $N = 1M$ events and $n = 10\%N$, one would save 90% of the CPU resources and 79% of the disk storage, almost equally shared among the full-format training data and the $(N-n)$ GEN-only data.

In principle, the adoption of Next-to-Leading order precision as a default for event generators could make the cost of the GEN step more relevant in the future. On the other hand, the upgrade of the detectors towards more granularity will also substantially increase the SIM. We then expect that the SIM step would still be the dominant consumer of CPU time, unless acceleration strategies like those proposed here will introduce beyond-GEANT alternatives. In addition, we do expect progresses to speed up the GEN step as well, e.,g., moving the computation to GPUs or similar accelerators [360], or using deep learning in phase-space integration [361–364].

12.6 Resource utilization for a standard GEANT4-based generation workflow

In this section, we describe how we derived the values quoted in Fig. 12.2. We take as a reference the CMS experiment. In absence of a published reference with a breakdown of CPU and disk resources for GEN, SIM, and DIGI+RECO steps, we derived the quantities quoted in Fig. 12.2 by generating QCD events on CPU, through the CERN batch system. To do so, we relied on the open-source CMSSW software [365] and followed the instructions provided by the CMS collaboration on the CERN Open Data portal [366].

We consider the same setup used to generate one of the QCD Run II samples published on the CERN Open Data portal [367] and the software installation available on CERN `cvmfs` distributed file system.

For each step, we ran jobs with 100 and 10 events. For each job, we recorded CPU time and output file size. Each step is repeated 10 times and the average of each quantity is considered. The typical uncertainty on these mean values, measured by the standard deviation of the 10 values, is found to be at most of a few percent and hence considered negligible. After computing the average for each set of jobs, we take the difference between the 100-event and 10-event job of each kind, in order to remove the overhead CPU time and file size that does not originate from per-event tasks. By dividing these differences by 90, we derive the per-event quantities quoted in Fig. 12.2.

12.7 Further validation of the deep learning generation workflow

Figures 12.11 and 12.12 show the distribution predicted by the model as a function of the corresponding quantities from detector simulation, respectively, for input and auxiliary features. Both the reconstruction techniques start from the generator-level information and model the detector response through a set of random degrees of

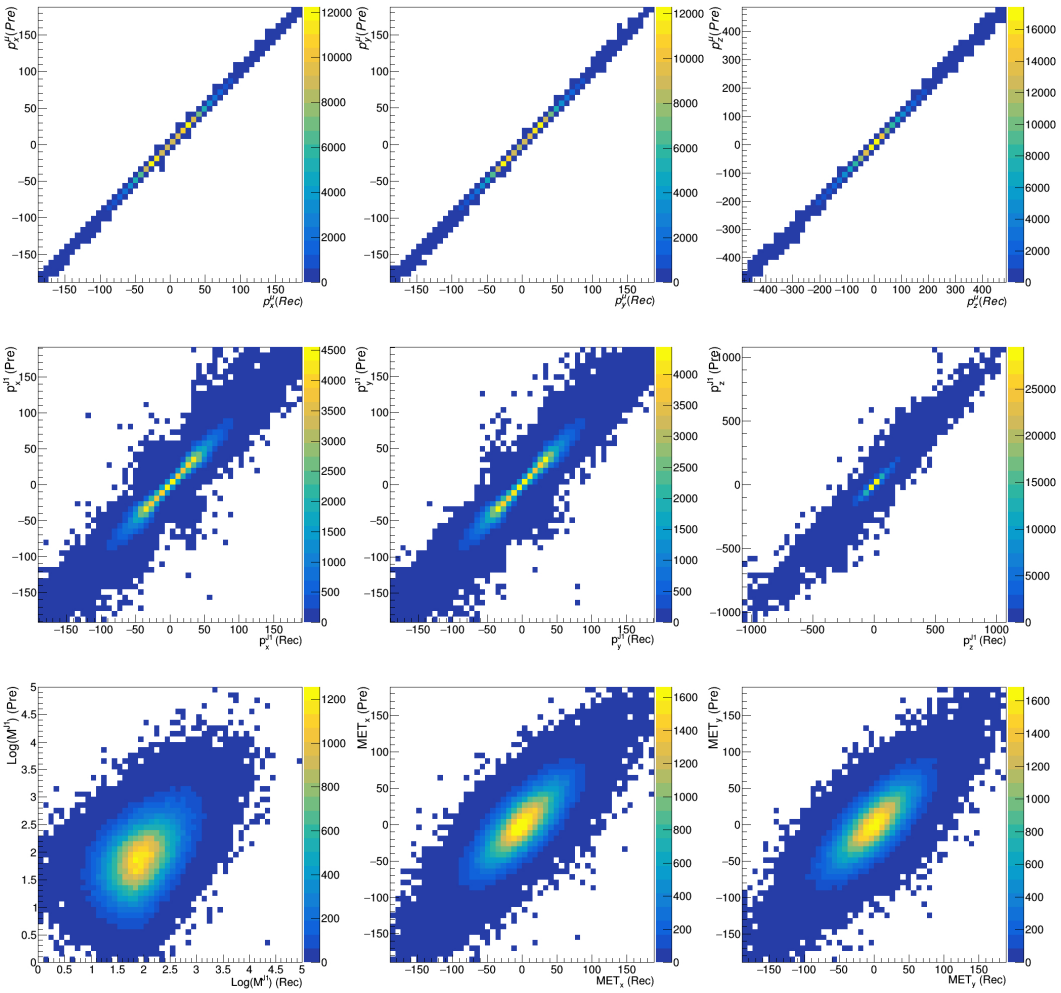


Figure 12.11: Distribution of input features predicted by the model as a function of the corresponding quantities from detector simulation.

freedom. The strong correlation and the symmetric distribution around the diagonal demonstrate that, to a large extent, the two event representations are equivalent.

12.8 Scaling with dataset size

Figures 12.13 and 12.14 show the comparison between reconstructed and generated quantities with five times more data, computed from detector simulation and processing the generator-level event with our model. Qualitatively, these distributions agree with those of Figs. 12.6 and 12.7, i.e., no accuracy deterioration is observed due to the scaling of the dataset size. This fact proves the robustness of the proposed methodology and its effectiveness for data augmentation.

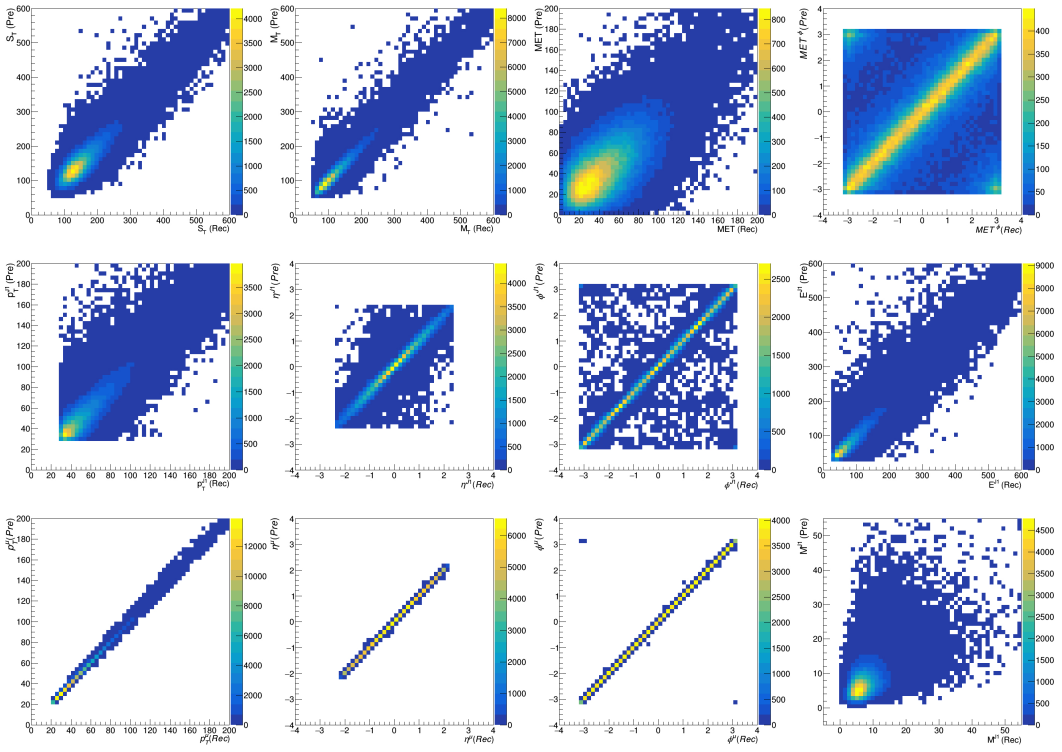


Figure 12.12: Distribution of auxiliary features predicted by the model as a function of the corresponding quantities from detector simulation.

12.9 Summary

We presented a proposal for a new data augmentation strategy for fast simulation workflows at LHC experiments, which exploits a generative Deep Learning model to convert an analysis-specific representation of collision events at generator level to the corresponding representation at reconstruction level. Following this procedure, one could replace any request of N simulated events with an $n < N$ request, providing the residual $(N - n)$ events at generator level. Bypassing the detector simulation and reconstruction process for the $(N - n)$ events, one would benefit of a substantial reduction in terms of required resources.

We demonstrated that a simple mean-and-variance regression model with a Gaussian sampling function allows to reach a good performance, producing a dataset which resembles that from a traditional workflow. We showed that the accuracy is preserved when applying our strategy to a test dataset much larger than the training dataset.

The proposed model is much simpler than a generative model, e.g., a GAN. The architecture is easier to train and the task it learns to solve is simpler than generative realistic events from random points in a latent space. The generator-level input

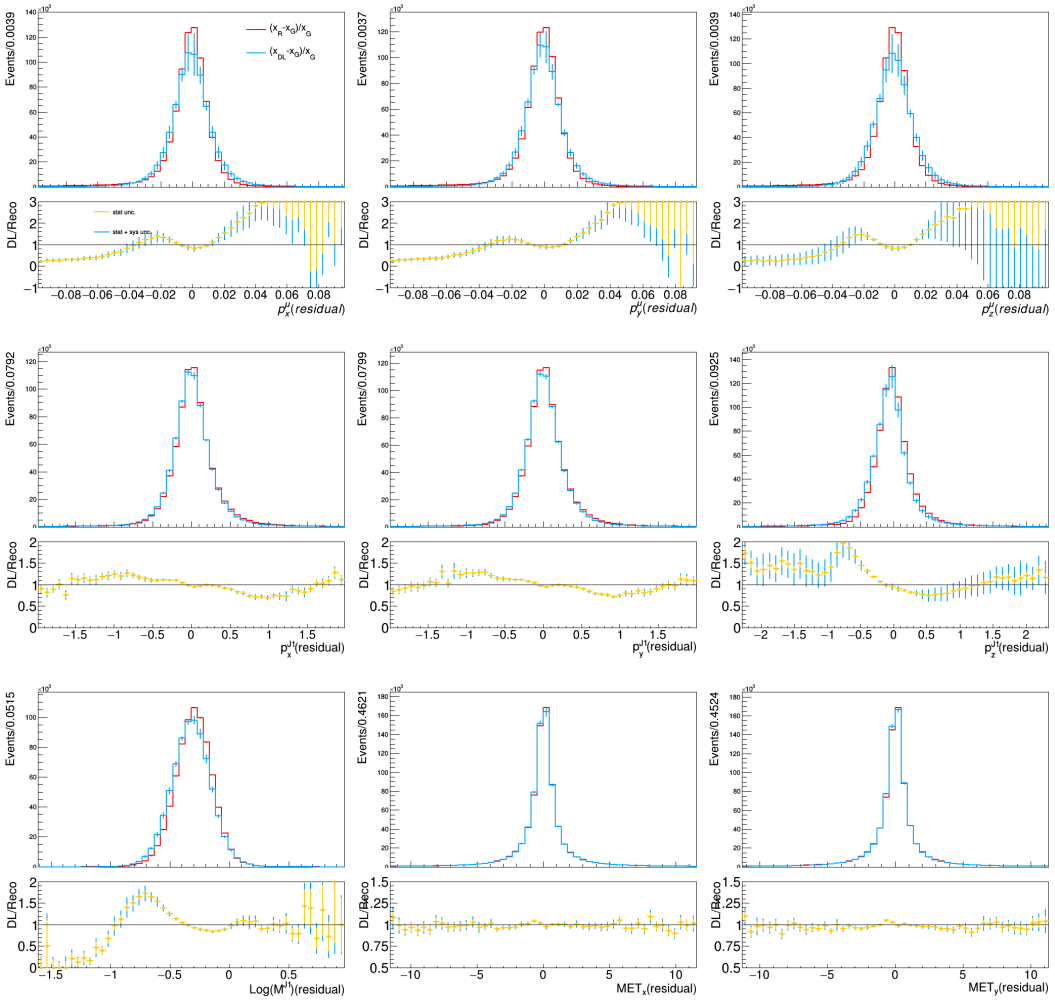


Figure 12.13: Predict on an inference dataset five times larger than the training dataset. Relative residual distribution for reconstructed and model-predicted quantities in the feature vector, computing with respect to the reference input.

carries much of the domain knowledge and the statistical fluctuations of the target dataset size. In addition, thanks to the light computational weight of the training and inference steps, one could consider to train several models and apply them to the same test dataset, using the spread of predictions to evaluate a simulation systematic uncertainty.

We believe that the LHC experiments could benefit from adopting the proposed procedure, particularly for the high-precision measurement era during the High-Luminosity LHC phase.

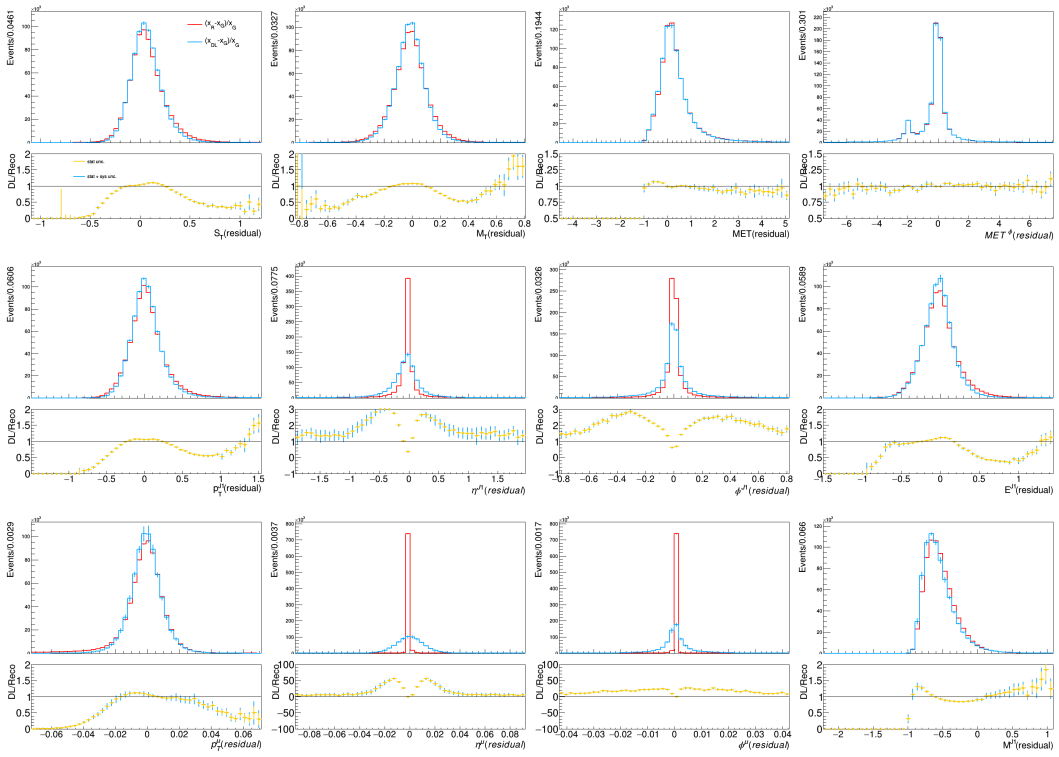


Figure 12.14: Predict on an inference dataset five times larger than the training dataset. Relative residual distribution for reconstructed and model-predicted auxiliary quantities, computing with respect to the reference input.

CONCLUSION

This thesis presented a brief overview of the Standard Model and the theoretical motivation to measure the Higgs self-coupling, which encodes information about possible deviation from the SM as well as the shape of the Brout-Englert-Higgs potential, which has direct impact on the stability of the universe. We presented a search for nonresonant Higgs pair production in the $HH \rightarrow b\bar{b}\gamma\gamma$ final states with CMS Run 2 data at $\sqrt{s}$ = 13 TeV to set limit on the Higgs boson self-coupling modifier κ_λ. The observed limit is set within a range $-3.3 < \kappa_\lambda < 8.5$, which is the most sensitive limit to date. The 95% CL upper limit on the product of the Higgs boson pair production cross section and branching fraction into $b\bar{b}\gamma\gamma$ is observed (expected) to be 7.7 (5.2) times the SM prediction. A novel technique to identify boosted $H \rightarrow b\bar{b}$ jets using graph neural networks was also presented to improve sensitivity for future Higgs searches. This technique was shown to outperform the existing `DeepDoubleB` algorithm, popularly used in CMS, while using fewer parameters and being more resource efficient.

We also discussed the hierarchy problem that motivated the supersymmetry extension beyond the Standard Model. In particular, we discussed the gauge-mediated supersymmetry framework, where the neutralino is the long-lived particle that decays into a photon and a gravitino. We presented a search for this new physics process, using the cluster information from the ECAL in combination with a deep neural network to identify photons not coming from the interaction vertices, which is the salient signature for this physics process. Results were shown as the expected exclusion limits in terms of the proper decay lengths and masses of the neutralinos, which are also proportional to the effective SUSY breaking scale. The expected exclusion limits extend the limits from previous searches by five times for the neutralino proper decay length and up to 100 GeV for the neutralino mass.

We introduced the concept of anomaly trigger in pursue of model-independent searches for new physics. The anomaly detection algorithm trained on the Standard Model cocktail can assign an anomaly score for a physics event, saving a dedicated anomalous data stream for further scrutiny. We also showed how it can be trained directly on data, and deployed in both HLT and L1 trigger, with the latter being implemented on FPGA.

We addressed the enormous computational challenges in the High-Luminosity LHC era and proposed several machine-learning solutions to tackle these challenges. First, we proposed a cleanup layer based on event topology to reduce the background rate of the trigger selection by more than one order of magnitude while retaining 99% of the signal events. The saved resources in terms of downstream processing and storage allow us to either relax the trigger threshold or reinvest in additional triggers for other physics programs. Second, we described a possible substitution for full event simulation based on recurrent generative adversarial networks, which has the potential to speed up the event simulation by five orders of magnitude. Lastly, we described a practical fast simulation solution for a specific analysis, where the number of physics objects are predetermined. Using an encoder-decoder architecture, we can emulate the detector response without running the simulation step, directly transform Monte Carlo generator event information into reconstructed event information, which can result in about an order-of-magnitude reduction in computing and storage requirements for the collision simulation workflow.

BIBLIOGRAPHY

[1] *The Standard Model of Particle Physics*. URL: https://www.symmetrymagazine.org/standard-model/ (visited on 08/24/2021).

[2] Murray Gell-Mann. *The Eightfold Way: A theory of strong interaction symmetry*. Tech. rep. Mar. 1961. DOI: 10.2172/4008239.

[3] Myron Bander. "Theories of quark confinement." In: *Physics Reports* 75.4 (1981), pp. 205–286. ISSN: 0370-1573. DOI: 10.1016/0370-1573(81)90026-0.

[4] David J. Gross and Frank Wilczek. "Ultraviolet behavior of non-abelian gauge theories." In: *Physical Review Letters* 30 (26 June 1973), pp. 1343–1346. DOI: 10.1103/PhysRevLett.30.1343.

[5] Sheldon L. Glashow. "Partial-symmetries of weak interactions." In: *Nuclear Physics* 22.4 (1961), pp. 579–588. ISSN: 0029-5582. DOI: 10.1016/0029-5582(61)90469-2.

[6] Peter W. Higgs. "Broken symmetries, massless particles and gauge fields." In: *Physics Letters* 12.2 (1964), pp. 132–133. ISSN: 0031-9163. DOI: 10.1016/0031-9163(64)91136-9.

[7] Gerald Guralnik, Carl Hagen, and Thomas W. B. Kibble. "Global conservation laws and massless particles." In: *Physical Review Letters* 13 (20 Nov. 1964), pp. 585–587. DOI: 10.1103/PhysRevLett.13.585.

[8] Peter W. Higgs. "Broken symmetries and the masses of gauge bosons." In: *Physical Review Letters* 13 (16 Oct. 1964), pp. 508–509. DOI: 10.1103/PhysRevLett.13.508.

[9] Francois Englert and Robert Brout. "Broken symmetry and the mass of gauge vector mesons." In: *Physical Review Letters* 13 (9 Aug. 1964), pp. 321–323. DOI: 10.1103/PhysRevLett.13.321.

[10] Thomas W. B. Kibble. "Symmetry breaking in non-abelian gauge theories." In: *Physical Review* 155 (5 Mar. 1967), pp. 1554–1561. DOI: 10.1103/PhysRev.155.1554.

[11] Peter W. Higgs. "Spontaneous symmetry breakdown without massless bosons." In: *Physical Review* 145 (4 May 1966), pp. 1156–1163. DOI: 10.1103/PhysRev.145.1156.

[12] Steven Weinberg. "A model of leptons." In: *Physical Review Letters* 19 (21 Nov. 1967), pp. 1264–1266. DOI: 10.1103/PhysRevLett.19.1264.

[13] Abdus Salam. "Weak and electromagnetic interactions." In: *Selected Papers of Abdus Salam* (1994), pp. 244–254. DOI: 10.1142/9789812795915_0034.

[14] Tommi Markkanen, Arttu Rajantie, and Stephen Stopyra. "Cosmological aspects of Higgs vacuum metastability." In: *Frontiers in Astronomy and Space Sciences* 5 (Dec. 2018). ISSN: 2296-987X. DOI: 10.3389/fspas.2018.00040.

[15] Michael E. Fisher. "The renormalization group in the theory of critical behavior." In: *Reviews of Modern Physics* 46 (4 Oct. 1974), pp. 597–616. DOI: 10.1103/RevModPhys.46.597. URL: https://link.aps.org/doi/10.1103/RevModPhys.46.597.

[16] Steven R. White. "Density matrix formulation for quantum renormalization groups." In: *Physical Review Letters* 69 (19 Nov. 1992), pp. 2863–2866. DOI: 10.1103/PhysRevLett.69.2863.

[17] Sidney Coleman. "Fate of the false vacuum: Semiclassical theory." In: *Physical Review D* 15 (10 May 1977), pp. 2929–2936. DOI: 10.1103/PhysRevD.15.2929.

[18] Curtis G. Callan and Sidney Coleman. "Fate of the false vacuum. II. First quantum corrections." In: *Physical Review D* 16 (6 Sept. 1977), pp. 1762–1768. DOI: 10.1103/PhysRevD.16.1762.

[19] Stephen P. Martin. "A supersymmetry primer." In: *Advanced Series on Directions in High Energy Physics* (July 1998), pp. 1–98. ISSN: 1793-1339. DOI: 10.1142/9789812839657_0001.

[20] Porter Williams. "Naturalness, the autonomy of scales, and the 125 GeV Higgs." In: *Studies in History and Philosophy of Science Part B: Studies in History and Philosophy of Modern Physics* 51 (2015), pp. 82–96. ISSN: 1355-2198. DOI: 10.1016/j.shpsb.2015.05.003.

[21] Mark Levinson. *Particle fever.* 2013.

[22] Pierre Fayet. "Mixing between gravitational and weak interactions through the massive gravitino." In: *Physics Letters B* 70 (1977), p. 461. DOI: 10.1016/0370-2693(77)90414-2.

[23] Howard Baer et al. "Signals for the minimal gauge-mediated supersymmetry breaking model at the Fermilab Tevatron collider." In: *Physical Review D* 55 (1997), p. 4463. DOI: 10.1103/PhysRevD.55.4463. eprint: hep-ph/9610358.

[24] Howard Baer et al. "Reach of Tevatron upgrades in gauge-mediated supersymmetry breaking models." In: *Physical Review D* 60 (1999), p. 055001. DOI: 10.1103/PhysRevD.60.055001. eprint: hep-ph/9903333.

[25] Savas Dimopoulos et al. "Sparticle spectroscopy and electroweak symmetry breaking with gauge-mediated supersymmetry breaking." In: *Nuclear Physics B* 488 (1997), p. 39. DOI: 10.1016/S0550-3213(97)00030-8. eprint: hep-ph/9609434.

[26] John Ellis et al. “Analysis of LEP constraints on supersymmetric models with a light gravitino.” In: *Physics Letters B* 394 (1997), p. 354. DOI: 10.1016/S0370-2693(97)00019-1. eprint: hep-ph/9610470.

[27] Michael Dine et al. “New tools for low energy dynamical supersymmetry breaking.” In: *Physical Review D* 53 (1996), p. 2658. DOI: 10.1103/PhysRevD.53.2658. eprint: hep-ph/9507378.

[28] Gian Francesco Giudice and Riccard Rattazzi. “Gauge-mediated supersymmetry breaking.” In: *Perspectives on supersymmetry*. World Scientific, Singapore, 1998, p. 355.

[29] Luis Alvarez-Gaume, Mark Claudson, and Mark B. Wise. “Low-energy supersymmetry.” In: *Nuclear Physics B* 207.1 (1982), pp. 96–110. ISSN: 0550-3213. DOI: 10.1016/0550-3213(82)90138-9.

[30] Gian Francesco Giudice and Riccardo Rattazzi. “Theories with gauge-mediated supersymmetry breaking.” In: *Physics Reports* 322 (1999), p. 419. DOI: 10.1016/S0370-1573(99)00042-3. arXiv: hep-ph/9801271 [hep-ph].

[31] Savas Dimopoulos et al. “Experimental signatures of low energy gauge-mediated supersymmetry breaking.” In: *Physical Review Letters* 76 (19 May 1996), pp. 3494–3497. DOI: 10.1103/PhysRevLett.76.3494.

[32] Abdelhamid Albaid and Kaladi S. Babu. “Higgs boson of mass 125ăGeV in gauge mediated supersymmetry breaking models with matter-messenger mixing.” In: *Physical Review D* 88.5 (Sept. 2013). ISSN: 1550-2368. DOI: 10.1103/physrevd.88.055007.

[33] Lyndon Evans and Philip Bryant. “LHC machine.” In: *Journal of Instrumentation* 3.08 (Aug. 2008), S08001–S08001. DOI: 10.1088/1748-0221/3/08/s08001.

[34] Esma Mobs. *The CERN accelerator complex - 2019. Complexe des accélérateurs du CERN - 2019.* General Photo. July 2019. URL: http://cds.cern.ch/record/2684277 (visited on 08/10/2021).

[35] CMS Collaboration. *CMS Luminosity: Public results*. URL: https://twiki.cern.ch/twiki/bin/view/CMSPublic/LumiPublicResults (visited on 08/17/2021).

[36] Oliver Sim Brüning et al. *LHC design report.* CERN Yellow Reports: Monographs. Geneva: CERN, 2004. DOI: 10.5170/CERN-2004-003-V-1. URL: https://cds.cern.ch/record/782076.

[37] CMS Collaboration. “Measurement of the inelastic proton-proton cross section at $\sqrt{s}$ = 13 TeV.” In: *Journal of High Energy Physics* 07 (2018), p. 161. DOI: 10.1007/JHEP07(2018)161. arXiv: 1802.02613 [hep-ex].

[38] CMS Collaboration. *CMS physics: Technical design report Volume 1: Detector performance and software.* CMS Technical Design Report CERN-LHCC-2006-001. 2006. URL: http://cds.cern.ch/record/922757.

[39] Tai Sakuma. “Cutaway diagrams of CMS detector.” May 2019. URL: http://cds.cern.ch/record/2665537.

[40] David Barney. “CMS detector slice.” CMS Collection. Jan. 2016. URL: http://cds.cern.ch/record/2120661.

[41] CMS Collaboration. “Precise mapping of the magnetic field in the CMS barrel yoke using cosmic rays.” In: *Journal of Instrumentation* 5.03 (Mar. 2010), T03021–T03021. DOI: 10.1088/1748-0221/5/03/t03021.

[42] J. C. Lottin et al. “Conceptual design of the CMS 4 Tesla solenoid.” In: *Advances in Cryogenic Engineering: Part A*. Ed. by Peter Kittel. Boston, MA: Springer US, 1996, pp. 819–826. DOI: 10.1007/978-1-4613-0373-2_106.

[43] F. Kircher et al. “Final design of the CMS solenoid cold mass.” In: *16th International Conference on Magnet Technology (MT-16)*. Oct. 1999.

[44] Vyacheslav Klyukhin et al. “Measuring the magnetic flux density in the CMS steel yoke.” In: *Journal of Superconductivity and Novel Magnetism* 26.4 (Dec. 2012), pp. 1307–1311. ISSN: 1557-1947. DOI: 10.1007/s10948-012-1967-5.

[45] CMS Collaboration. *CMS Tracker Detector Performance Results.* URL: https://twiki.cern.ch/twiki/bin/view/CMSPublic/DPGResultsTRK (visited on 08/17/2021).

[46] Martin Lipinski. *The Phase-1 upgrade of the CMS pixel detector.* Tech. rep. Geneva: CERN, May 2017. DOI: 10.1088/1748-0221/12/07/C07009.

[47] The CMS Collaboration. “Description and performance of track and primary-vertex reconstruction with the CMS tracker.” In: *Journal of Instrumentation* 9.10 (Oct. 2014), P10009–P10009. DOI: 10.1088/1748-0221/9/10/p10009.

[48] Cristina Biino. “The CMS electromagnetic calorimeter: Overview, lessons learned during Run 1 and future projections.” In: *Journal of Physics: Conference Series* 587 (Feb. 2015), p. 012001. DOI: 10.1088/1742-6596/587/1/012001.

[49] Nadja Strobbe. “The upgrade of the CMS hadron calorimeter with silicon photomultipliers.” In: *Journal of Instrumentation* 12.01 (Jan. 2017), pp. C01080–C01080. DOI: 10.1088/1748-0221/12/01/c01080.

[50] CMS Collaboration. “Performance of the CMS hadron calorimeter with cosmic ray muons and LHC beam data.” In: *Journal of Instrumentation* 5.03 (Mar. 2010). DOI: 10.1088/1748-0221/5/03/t03012.

[51] Federico De Guio and. “First results from the CMS SiPM-based hadronic endcap calorimeter.” In: *Journal of Physics: Conference Series* 1162 (Jan. 2019), p. 012009. DOI: 10.1088/1742-6596/1162/1/012009.

[52] CMS Collaboration. "Performance of the CMS muon detector and muon reconstruction with proton-proton collisions at $\sqrt{s}$ = 13 TeV." In: *Journal of Instrumentation* 13.06 (June 2018), P06015–P06015. ISSN: 1748-0221. DOI: 10.1088/1748-0221/13/06/p06015.

[53] CMS Collaboration. "Performance of the CMS Level-1 trigger in proton-proton collisions at $\sqrt{s}$ = 13 TeV." In: *Journal of Instrumentation* 15.10 (Oct. 2020), P10017–P10017. ISSN: 1748-0221. DOI: 10.1088/1748-0221/15/10/p10017.

[54] R Frazier et al. "A demonstration of a Time Multiplexed Trigger for the CMS experiment." In: *Journal of Instrumentation* 7.01 (Jan. 2012), pp. C01060–C01060. DOI: 10.1088/1748-0221/7/01/c01060.

[55] G. Hall et al. "A time-multiplexed track-trigger architecture for CMS." In: *Journal of Instrumentation* 9.10 (Oct. 2014), pp. C10034–C10034. DOI: 10.1088/1748-0221/9/10/c10034.

[56] CMS Collaboration. "The CMS high level trigger." In: *European Physical Journal C* 46 (2006), pp. 605–667. DOI: 10.1140/epjc/s2006-02495-8. arXiv: hep-ex/0512077 [hep-ex].

[57] Matteo Cacciari, Gavin P. Salam, and Gregory Soyez. "The anti-k_t jet clustering algorithm." In: *Journal of High Energy Physics* 04 (2008), p. 063. DOI: 10.1088/1126-6708/2008/04/063. arXiv: 0802.1189.

[58] Valentina Gori. "The CMS high level trigger." In: *International Journal of Modern Physics: Conference Series* 31 (2014), p. 1460297. DOI: 10.1142/S201019451460297X.

[59] CMS Collaboration. "Identification of heavy-flavour jets with the CMS detector in pp collisions at 13 TeV." In: *Journal of Instrumentation* 13 (2018), P05011. DOI: 10.1088/1748-0221/13/05/P05011. arXiv: 1712.07158 [physics.ins-det].

[60] Michael Aderholz et al. *Models of networked analysis at regional centres for LHC experiments (MONARC), Phase 2 report, 24th March 2000.* Tech. rep. Geneva: CERN, Apr. 2000. URL: https://cds.cern.ch/record/510694.

[61] *The worldwide LHC computing grid.* URL: https://wlcg.web.cern.ch/ (visited on 08/23/2021).

[62] Antonio Pérez-Calero Yzquierdo et al. "Evolution of the CMS global submission infrastructure for the HL-LHC era." In: *EPJ Web of Conferences* 245 (2020), 03016. 8 p. DOI: 10.1051/epjconf/202024503016.

[63] Edgar Fajardo Hernandez et al. "A new era for central processing and production in CMS." In: *Journal of Physics: Conference Series* 396.4 (Dec. 2012), p. 042018. DOI: 10.1088/1742-6596/396/4/042018.

[64] Antonio Pérez-Calero Yzquierdo et al. “Evolution of CMS workload management towards multicore job support.” In: *Journal of Physics: Conference Series* 664.6 (Dec. 2015), p. 062046. DOI: 10.1088/1742-6596/664/6/062046.

[65] Igor Sfiligoi et al. “The pilot way to grid resources using glideinWMS.” In: *2009 WRI World Congress on Computer Science and Information Engineering*. Vol. 2. 2009, pp. 428–432. DOI: 10.1109/CSIE.2009.950.

[66] Douglas Thain, Todd Tannenbaum, and Miron Livny. “Condor and the Grid.” In: *Grid Computing: Making the Global Infrastructure a Reality*. Ed. by Fran Berman, Geoffrey Fox, and Tony Hey. John Wiley & Sons Inc., Dec. 2002.

[67] Jim Basney and Miron Livny. “Deploying a high throughput computing cluster.” In: *High Performance Cluster Computing: Architectures and Systems, Volume 1*. Ed. by Rajkumar Buyya. Prentice Hall PTR, 1999.

[68] Douglas Thain, Todd Tannenbaum, and Miron Livny. “Distributed computing in practice: the Condor experience.” In: *Concurrency and Computation: Practice and Experience* 17.2-4 (2005), pp. 323–356.

[69] Pérez-Calero Yzquierdo, Antonio et al. “Exploring GlideinWMS and HTCondor scalability frontiers for an expanding CMS global pool.” In: *EPJ Web of Conferences* 214 (2019), p. 03002. DOI: 10.1051/epjconf/201921403002.

[70] Kenneth Bloom. “CMS use of a data federation.” In: *Journal of Physics: Conference Series* 513 (2014), p. 042005. DOI: 10.1088/1742-6596/513/4/042005.

[71] *XRootD*. URL: https://xrootd.slac.stanford.edu/ (visited on 08/23/2021).

[72] Ian Bird et al. “Architecture and prototype of a WLCG data lake for HL-LHC.” In: *EPJ Web of Conferences* 214 (2019), 04024. 6 p. DOI: 10.1051/epjconf/201921404024.

[73] Tommaso Boccali. “Computing models in high energy physics.” In: *Reviews in Physics* 4 (Oct. 2019), p. 100034. DOI: 10.1016/j.revip.2019.100034.

[74] Julien Baglio et al. “The measurement of the Higgs self-coupling at the LHC: theoretical status.” In: *Journal of High Energy Physics* 1304 (2013), p. 151. DOI: 10.1007/jhep04(2013)151. arXiv: 1212.5581 [hep-ph].

[75] Stefan Dittmaier et al. *Handbook of LHC Higgs cross sections: 2. Differential distributions.* Tech. rep. Jan. 2012. DOI: 10.5170/CERN-2012-002. arXiv: 1201.3084 [hep-ph].

[76] Daniel de Florian and Javier Mazzitelli. “Higgs boson pair production at next-to-next-to-leading order in QCD.” In: *Physical Review Letters* 111 (2013), p. 201801. DOI: 10.1103/PhysRevLett.111.201801. arXiv: 1309.6594 [hep-ph].

[77] Alexandra Carvalho et al. *Analytical parametrisation and shape classification of anomalous HH production in EFT approach.* Tech. rep. Feb. 2016. URL: https://cds.cern.ch/record/2130724.

[78] CMS Collaboration. "Combined measurements of Higgs boson couplings in protonproton collisions at $\sqrt{s} = 13\,\text{TeV}$." In: *European Physical Journal C* 79 (2019), p. 421. DOI: 10.1140/epjc/s10052-019-6909-y. arXiv: 1809.10733 [hep-ex].

[79] ATLAS Collaboration. "Search for Higgs boson pair production in the $\gamma\gamma b\bar{b}$ final state with 13 TeV pp collision data collected by the ATLAS experiment." In: *Journal of High Energy Physics* 11 (2018), p. 040. DOI: 10.1007/jhep11(2018)040. arXiv: 1807.04873 [hep-ex].

[80] ATLAS Collaboration. "Combination of searches for Higgs boson pairs in pp collisions at $\sqrt{s}$ =13 TeV with the ATLAS detector." In: *Physics Letters B* 800 (2020), p. 135103. DOI: 10.1016/j.physletb.2019.135103. arXiv: 1906.02025 [hep-ex].

[81] ATLAS Collaboration. "Search for Higgs boson pair production in the $\gamma\gamma b\bar{b}$ final state using pp collision data at $\sqrt{s} = 8$ TeV from the ATLAS detector." In: *Physical Review Letters* 114 (2015), p. 081802. DOI: 10.1103/PhysRevLett.114.081802. arXiv: 1406.5053 [hep-ex].

[82] ATLAS Collaboration. "Search for Higgs boson pair production in the $b\bar{b}\,b\bar{b}$ final state from pp collisions at $\sqrt{s} = 8$ TeV with the ATLAS detector." In: *European Physical Journal C* 75 (2015), p. 412. DOI: 10.1140/epjc/s10052-015-3628-x. arXiv: 1506.00285 [hep-ex].

[83] ATLAS Collaboration. "Search for pair production of Higgs bosons in the $b\bar{b}b\bar{b}$ final state using proton–proton collisions at $\sqrt{s} = 13\,\text{TeV}$ with the ATLAS detector." In: *Physical Review D* 94 (2016), p. 052002. DOI: 10.1103/PhysRevD.94.052002. arXiv: 1606.04782 [hep-ex].

[84] ATLAS Collaboration. "Searches for Higgs boson pair production in the $HH \to b\bar{b}\tau\tau, \gamma\gamma WW^*, \gamma\gamma b\bar{b}, b\bar{b}b\bar{b}$ channels with the ATLAS detector." In: *Physical Review D* 92 (2015), p. 092004. DOI: 10.1103/PhysRevD.92.092004. arXiv: 1509.04670 [hep-ex].

[85] CMS Collaboration. "Search for Higgs boson pair production in the $b\bar{b}\tau\tau$ final state in proton-proton collisions at $\sqrt{s} = 8$ TeV." In: *Physical Review D* 96 (2017), p. 072004. DOI: 10.1103/PhysRevD.96.072004. arXiv: 1707.00350 [hep-ex].

[86] CMS Collaboration. "Search for resonant and nonresonant Higgs boson pair production in the $b\bar{b}\ell\nu\ell\nu$ final state in proton-proton collisions at $\sqrt{s} = 13$ TeV." In: *Journal of High Energy Physics* 01 (2018), p. 054. DOI: 10.1007/jhep01(2018)054. arXiv: 1708.04188 [hep-ex].

[87] CMS Collaboration. "Search for two Higgs bosons in final states containing two photons and two bottom quarks in proton-proton collisions at 8 TeV." In: *Physical Review D* 94 (2016), p. 052012. DOI: 10.1103/PhysRevD.94.052012. arXiv: 1603.06896 [hep-ex].

[88] CMS Collaboration. "Search for Higgs boson pair production in events with two bottom quarks and two tau leptons in proton–proton collisions at $\sqrt{s}$ = 13 TeV." In: *Physics Letters B* 778 (2018), p. 101. DOI: 10.1016/j.physletb.2018.01.001. arXiv: 1707.02909 [hep-ex].

[89] CMS Collaboration. "Search for Higgs boson pair production in the $\gamma\gamma b\bar{b}$ final state in pp collisions at $\sqrt{s}$ = 13 TeV." In: *Physics Letters B* 788 (2019), p. 7. DOI: 10.1016/j.physletb.2018.10.056. arXiv: 1806.00408 [hep-ex].

[90] CMS Collaboration. "Combination of searches for Higgs boson pair production in proton-proton collisions at $\sqrt{s}$ = 13 TeV." In: *Physical Review Letters* 122 (2019), p. 121803. DOI: 10.1103/PhysRevLett.122.121803. arXiv: 1811.09689 [hep-ex].

[91] ATLAS Collaboration. "Search for the HH $\rightarrow b\bar{b}b\bar{b}$ process via vector-boson fusion production using proton-proton collisions at $\sqrt{s}$ = 13 TeV with the ATLAS detector." In: *Journal of High Energy Physics* 07 (2020), p. 108. DOI: 10.1007/jhep07(2020)108. arXiv: 2001.05178 [hep-ex].

[92] Daniel de Florian et al. *Handbook of LHC Higgs cross sections: 4. Deciphering the nature of the Higgs sector.* CERN Report CERN-2017-002-M. 2016. DOI: 10.23731/CYRM-2017-002. arXiv: 1610.07922 [hep-ph].

[93] Emanuele Bagnaschi et al. "Higgs production via gluon fusion in the POWHEG approach in the SM and in the MSSM." In: *Journal of High Energy Physics* 02 (2012), p. 088. DOI: 10.1007/jhep02(2012)088. arXiv: 1111.2854 [hep-ph].

[94] Gudrun Heinrich et al. "NLO predictions for Higgs boson pair production with full top quark mass dependence matched to parton showers." In: *Journal of High Energy Physics* 08 (2017), p. 088. DOI: 10.1007/jhep08(2017)088. arXiv: 1703.09252 [hep-ph].

[95] Gudrun Heinrich et al. "Probing the trilinear Higgs boson coupling in di-Higgs production at NLO QCD including parton shower effects." In: *Journal of High Energy Physics* 06 (2019), p. 066. DOI: 10.1007/jhep06(2019)066. arXiv: 1903.08137 [hep-ph].

[96] Gudrun Heinrich et al. "A non-linear EFT description of $gg \rightarrow HH$ at NLO interfaced to POWHEG." In: *Journal of High Energy Physics* 10 (2020), p. 021. DOI: 10.1007/jhep10(2020)021. arXiv: 2006.16877 [hep-ph].

[97] Stephen Philip Jones and Silvan Kuttimalai. "Parton shower and NLO-matching uncertainties in Higgs boson pair production." In: *Journal of High Energy Physics* 02 (2018), p. 176. DOI: 10.1007/jhep02(2018)176. arXiv: 1711.03319 [hep-ph].

[98] Gerhard Buchalla et al. “Higgs boson pair production in non-linear Effective Field Theory with full m_t-dependence at NLO QCD.” In: *Journal of High Energy Physics* 09 (2018), p. 057. DOI: 10.1007/jhep09(2018)057. arXiv: 1806.05162 [hep-ph].

[99] Johan Alwall et al. “The automated computation of tree-level and next-to-leading order differential cross sections, and their matching to parton shower simulations.” In: *Journal of High Energy Physics* 07 (2014), p. 079. DOI: 10.1007/jhep07(2014)079. arXiv: 1405.0301 [hep-ph].

[100] Benoit Hespel, David Lopez-Val, and Eleni Vryonidou. “Higgs pair production via gluon fusion in the Two-Higgs-Doublet model.” In: *Journal of High Energy Physics* 09 (2014), p. 124. DOI: 10.1007/jhep09(2014)124. arXiv: 1407.0281 [hep-ph].

[101] Rikkert Frederix et al. “Higgs pair production at the LHC with NLO and parton-shower effects.” In: *Physics Letters B* 732 (2014), p. 142. DOI: 10.1016/j.physletb.2014.03.026. arXiv: 1401.7340 [hep-ph].

[102] Tanju Gleisberg et al. “Event generation with SHERPA 1.1.” In: *Journal of High Energy Physics* 02 (2009), p. 007. DOI: 10.1088/1126-6708/2009/02/007. arXiv: 0811.4622 [hep-ph].

[103] Torbjörn Sjöstrand et al. “An introduction to PYTHIA 8.2.” In: *Computer Physics Communications* 191 (2015), p. 159. DOI: 10.1016/j.cpc.2015.01.024. arXiv: 1410.3012.

[104] Simone Alioli et al. “A general framework for implementing NLO calculations in shower Monte Carlo programs: the POWHEG BOX.” In: *Journal of High Energy Physics* 06 (2010), p. 043. DOI: 10.1007/jhep06(2010)043. arXiv: 1002.2581 [hep-ph].

[105] Stefano Frixione, Paolo Nason, and Carlo Oleari. “Matching NLO QCD computations with Parton Shower simulations: the POWHEG method.” In: *Journal of High Energy Physics* 11 (2007), p. 070. DOI: 10.1088/1126-6708/2007/11/070. arXiv: 0709.2092 [hep-ph].

[106] Paolo Nason. “A new method for combining NLO QCD with shower Monte Carlo algorithms.” In: *Journal of High Energy Physics* 11 (2004), p. 040. DOI: 10.1088/1126-6708/2004/11/040. arXiv: hep-ph/0409146.

[107] CMS Collaboration. “Event generator tunes obtained from underlying event and multiparton scattering measurements.” In: *European Physical Journal C* 76 (2016), p. 155. DOI: 10.1140/epjc/s10052-016-3988-x. arXiv: 1512.00815.

[108] CMS Collaboration. “Extraction and validation of a new set of CMS PYTHIA8 tunes from underlying-event measurements.” In: *European Physical Journal C* 80 (2020), p. 4. DOI: 10.1140/epjc/s10052-019-7499-4. arXiv: 1903.12179 [hep-ex].

[109] Richard D. Ball et al. "Parton distributions for the LHC Run II." In: *Journal of High Energy Physics* 04 (2015), p. 040. DOI: 10.1007/jhep04(2015)040. arXiv: 1410.8849 [hep-ph].

[110] Richard D. Ball et al. "Parton distributions from high-precision collider data." In: *European Physical Journal C* 77 (2017), p. 663. DOI: 10.1140/epjc/s10052-017-5199-5. arXiv: 1706.00428 [hep-ph].

[111] GEANT4 Collaboration. "Geant4—a simulation toolkit." In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 506.3 (2003), pp. 250–303. ISSN: 0168-9002. DOI: 10.1016/S0168-9002(03)01368-8.

[112] Barbara Jäger et al. "Parton-shower effects in Higgs production via vector-boson fusion." In: *European Physical Journal C* 80 (2020), p. 756. DOI: 10.1140/epjc/s10052-020-8326-7. arXiv: 2003.12435 [hep-ph].

[113] CMS Collaboration. "Particle-flow reconstruction and global event description with the CMS detector." In: *Journal of Instrumentation* 12 (2017), P10003. DOI: 10.1088/1748-0221/12/10/P10003. arXiv: 1706.04965 [physics.ins-det].

[114] CMS Collaboration. "Measurements of Higgs boson properties in the diphoton decay channel in proton-proton collisions at $\sqrt{s} = 13$ TeV." In: *Journal of High Energy Physics* 11 (2018), p. 185. DOI: 10.1007/jhep11(2018)185. arXiv: 1804.02716 [hep-ex].

[115] CMS Collaboration. *Performance of the DeepJet b tagging algorithm using 41.9* fb^{-1} *of data from proton-proton collisions at 13 TeV with Phase 1 CMS detector.* CMS Detector Performance Note CMS-DP-2018-058. 2018. URL: https://cds.cern.ch/record/2646773.

[116] Emil Bols et al. "Jet flavour classification using DeepJet." In: *Journal of Instrumentation* 15 (P12012 2020). DOI: 10.1088/1748-0221/15/12/P12012. arXiv: 2008.10519 [hep-ex].

[117] CMS Collaboration. "A deep neural network for simultaneous estimation of b jet energy and resolution." In: *Computing and Software for Big Science* 4 (2020), p. 10. DOI: 10.1007/s41781-020-00041-z. arXiv: 1912.06046 [hep-ex].

[118] Nilanjana Kumar and Stephen P. Martin. "LHC search for di-Higgs decays of stoponium and other scalars in events with two photons and two bottom jets." In: *Physical Review D* 90 (2014), p. 055007. DOI: 10.1103/PhysRevD.90.055007. arXiv: 1404.0996 [hep-ph].

[119] Thong Q. Nguyen et al. "Topology classification with deep learning to improve real-time event selection at the LHC." In: *Computing and Software for Big Science* 3 (2019), p. 12. DOI: 10.1007/s41781-019-0028-1. arXiv: 1807.00083.

[120] François Chollet et al. *Keras*. https://keras.io. 2015. URL: https://keras.io.

[121] Martin Abadi et al. *TensorFlow: Large-scale machine Learning on heterogeneous distributed systems.* 2016. arXiv: 1603.04467 [cs.DC]. URL: https://arxiv.org/abs/1603.04467.

[122] Tianqi Chen and Carlos Guestrin. "XGBoost: A scalable tree boosting system." In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD. San Francisco, California, USA: ACM, 2016, p. 785. ISBN: 978-1-4503-4232-2. DOI: 10.1145/2939672.2939785.

[123] CMS Collaboration. *Performance of quark/gluon discrimination in 8 TeV pp data.* CMS Physics Analysis Summary CMS-PAS-JME-13-002. 2013. URL: https://cds.cern.ch/record/1599732.

[124] CMS Collaboration. *Jet algorithms performance in 13 TeV data.* CMS Physics Analysis Summary CMS-PAS-JME-16-003. 2017. URL: https://cds.cern.ch/record/2256875.

[125] Fabio Maltoni et al. "Trilinear Higgs coupling determination via single-Higgs differential measurements at the LHC." In: *European Physical Journal C* 77 (2017), p. 887. DOI: 10.1140/epjc/s10052-017-5410-8. arXiv: 1709.08649 [hep-ph].

[126] CMS Collaboration. "Measurements of $t\bar{t}H$ production and the CP structure of the Yukawa interaction between the Higgs boson and top quark in the diphoton decay channel." In: *Physical Review Letters* 125 (2020), p. 061801. DOI: 10.1103/PhysRevLett.125.061801. arXiv: 2003.10866 [hep-ex].

[127] CMS Collaboration. "Search for direct production of supersymmetric partners of the top quark in the all-jets final state in proton-proton collisions at $\sqrt{s} = 13$ TeV." In: *Journal of High Energy Physics* 10 (2017), p. 005. DOI: 10.1007/jhep10(2017)005. arXiv: 1707.03316 [hep-ex].

[128] Mark Joseph Oreglia. "A study of the reactions $\psi' \to \gamma\gamma\psi$." SLAC Report SLAC-R-236. PhD thesis. Stanford University, 1980. URL: http://www.slac.stanford.edu/cgi-wrap/getdoc/slac-r-236.pdf.

[129] Paul Dauncey et al. "Handling uncertainties in background shapes: the discrete profiling method." In: *Journal of Instrumentation* 10 (2015), P04015. DOI: 10.1088/1748-0221/10/04/P04015. arXiv: 1408.6865 [physics.data-an].

[130] CMS Collaboration. "Observation of the diphoton decay of the Higgs boson and measurement of its properties." In: *European Physical Journal C* 74 (2014), p. 3076. DOI: 10.1140/epjc/s10052-014-3076-z. arXiv: 1407.0558 [hep-ex].

[131] CMS Collaboration. "Performance of photon reconstruction and identification with the CMS detector in proton-proton collisions at $\sqrt{s} = 8$ TeV." In: *Journal of Instrumentation* 10 (2015), P08010. DOI: 10.1088/1748-0221/10/08/P08010. arXiv: 1502.02702 [physics.ins-det].

[132] CMS Collaboration. "Jet energy scale and resolution in the CMS experiment in pp collisions at 8 TeV." In: *Journal of Instrumentation* 12 (2017), P02014. DOI: 10.1088/1748-0221/12/02/P02014. arXiv: 1607.03663.

[133] CMS Collaboration. "Measurement of the inclusive W and Z production cross sections in pp collisions at $\sqrt{s} = 7$ TeV." In: *Journal of High Energy Physics* 10 (2011), p. 132. DOI: 10.1007/jhep10(2011)132. arXiv: 1107.4789 [hep-ex].

[134] CMS Collaboration. *CMS luminosity measurements for the 2016 data-taking period.* CMS Physics Analysis Summary CMS-PAS-LUM-17-001. 2017. URL: https://cds.cern.ch/record/2257069.

[135] CMS Collaboration. *CMS luminosity measurement for the 2017 data-taking period at* $\sqrt{s} = 13$ TeV. CMS Physics Analysis Summary CMS-PAS-LUM-17-004. 2018. URL: https://cds.cern.ch/record/2621960.

[136] CMS Collaboration. *CMS luminosity measurement for the 2018 data-taking period at* $\sqrt{s} = 13$ TeV. CMS Physics Analysis Summary CMS-PAS-LUM-18-002. 2019. URL: https://cds.cern.ch/record/2676164.

[137] Alexander Lincoln Read. "Presentation of search results: the CL_s technique." In: *Journal of Physics G* 28 (2002), p. 2693. DOI: 10.1088/0954-3899/28/10/313.

[138] Thomas Junk. "Confidence level computation for combining searches with small statistics." In: *Nuclear Instruments and Methods in Physics Research A* 434 (1999), p. 435. DOI: 10.1016/S0168-9002(99)00498-2. arXiv: hep-ex/9902006 [hep-ex].

[139] Glen Cowan et al. "Asymptotic formulae for likelihood-based tests of new physics." In: *European Physical Journal C* 71 (2011), p. 1554. DOI: 10.1140/epjc/s10052-011-1554-0. arXiv: 1007.1727 [physics.data-an].

[140] ATLAS and CMS Collaborations. *Procedure for the LHC Higgs boson search combination in summer 2011.* 2011. URL: http://cdsweb.cern.ch/record/1379837.

[141] CMS Collaboration. "A measurement of the Higgs boson mass in the diphoton decay channel." In: *Physics Letters B* 805 (2020), p. 135425. DOI: 10.1016/j.physletb.2020.135425. arXiv: 2002.06398 [hep-ex].

[142] CMS Collaboration. "Measurement of the Higgs boson production rate in association with top quarks in final states with electrons, muons, and hadronically decaying tau leptons at $\sqrt{s} = 13$ TeV." In: *European Physical*

Journal C 81 (378 2021). DOI: 10.1140/epjc/s10052-021-09014-x. arXiv: 2011.03652 [hep-ex].

[143] Yuri L. Dokshitzer et al. "Better jet clustering algorithms." In: *Journal of High Energy Physics* 08 (1997), p. 001. DOI: 10.1088/1126-6708/1997/08/001. arXiv: hep-ph/9707323 [hep-ph].

[144] Stefano Catani et al. "Longitudinally-invariant $k_\perp$-clustering algorithms for hadron-hadron collisions." In: *Nuclear Physics B* 406 (1993), pp. 187–224. DOI: 10.1016/0550-3213(93)90166-M.

[145] Andrew J. Larkoski, Ian Moult, and Benjamin Nachman. "Jet substructure at the Large Hadron Collider: A review of recent advances in theory and machine learning." In: *Physics Reports* 841 (2020), p. 1. ISSN: 0370-1573. DOI: 10.1016/J.Physrep.2019.11.001. arXiv: 1709.04464.

[146] Mathias Niepert, Mohamed Ahmed, and Konstantin Kutzkov. "Learning convolutional neural networks for graphs." In: *Proceedings of the 33rd International Conference on Machine Learning*. Ed. by Maria Florina Balcan and Kilian Q. Weinberger. Proceedings of Machine Learning Research. New York: PMLR, 2016, p. 2014. URL: http://proceedings.mlr.press/v48/niepert16.html.

[147] Thomas N. Kipf and Max Welling. "Semi-supervised classification with graph convolutional networks." In: *5th International Conference on Learning Representations, International Conference on Learning Representations 2017, Conference Track Proceedings*. Amherst: OpenReview, 2017. arXiv: 1609.02907. URL: https://openreview.net/forum?id=SJU4ayYgl.

[148] Charles Ruizhongtai Qi et al. "PointNet: Deep learning on point sets for 3D classification and segmentation." In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. New York: IEEE, 2017. DOI: 10.1109/CVPR.2017.16. arXiv: 1612.00593.

[149] Yue Wang et al. "Dynamic graph CNN for learning on point clouds." In: *ACM Transactions on Graphics* 38 (2019), p. 146. DOI: 10.1145/3326362. arXiv: 1801.07829.

[150] Aditya Grover, Aaron Zweig, and Stefano Ermon. "Graphite: Iterative generative modeling of graphs." In: *Proceedings of the 36th International Conference on Machine Learning*. Ed. by Kamalika Chaudhuri and Ruslan Salakhutdinov. Vol. 97. Proceedings of Machine Learning Research. Long Beach: PMLR, 2019, p. 2434. arXiv: 1803.10459. URL: http://proceedings.mlr.press/v97/grover19a.html.

[151] Jiaxuan You et al. "GraphRNN: Generating realistic graphs with deep auto-regressive models." In: *Proceedings of the 35th International Conference on Machine Learning*. Vol. 80. Proceedings of Machine Learning Research. Stockholm: PMLR, 2018, p. 5708. arXiv: 1802.08773. URL: http://proceedings.mlr.press/v80/you18a.html.

[152] Joan Bruna et al. “Spectral networks and locally connected networks on graphs.” In: *2nd International Conference on Learning Representations, International Conference on Learning Representations 2014, Conference Track Proceedings*. Banf: International Conference on Learning Representations, 2014. arXiv: 1312.6203.

[153] David Zheng et al. “Unsupervised learning of latent physical properties using perception-prediction networks.” In: *Proceedings of the 34th Conference on Uncertainty in Artificial Intelligence*. Ed. by Amir Globerson and Ricardo Silva. Corvallis: AUAI Press, 2018, p. 497. arXiv: 1807.09244.

[154] Peter W. Battaglia et al. “Relational inductive biases, deep learning, and graph networks.” 2018. URL: https://arxiv.org/abs/1806.01261.

[155] Yann LeCun et al. “Gradient-based learning applied to document recognition.” In: *Proceedings of the IEEE* 11 (1998), p. 2278. DOI: 10.1109/5.726791.

[156] Kaiming He et al. “Deep residual learning for image recognition”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. New York: IEEE, 2016, p. 770. DOI: 10.1109/CVPR.2016.90. arXiv: 1512.03385.

[157] Michael M. Bronstein et al. “Geometric deep learning: Going beyond Euclidean data.” In: *IEEE Signal Processing Magazine* 34 (2017), p. 18. DOI: 10.1109/MSP.2017.2693418. arXiv: 1611.08097.

[158] Yujia Li et al. “Gated graph sequence neural networks.” In: *4th International Conference on Learning Representations, International Conference on Learning Representations 2016, Conference Track Proceedings*. San Juan: International Conference on Learning Representations, 2016. arXiv: 1511.05493.

[159] Peter W. Battaglia et al. “Interaction networks for learning about objects, relations and physics.” In: *Advances in Neural Information Processing Systems 29*. Ed. by D. D. Lee et al. Red Hook, New York: Curran Associates, Inc., 2016, p. 4502. arXiv: 1612.00222. URL: http://papers.nips.cc/paper/6418-interaction-networks-for-learning-about-objects-relations-and-physics.

[160] Simon Haykin. *Neural networks: A comprehensive foundation.* Prentice Hall PTR, 1994.

[161] Kunihiko Fukushima. “Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position.” In: *Biological Cybernetics* 36 (1980), pp. 193–202.

[162] Yann LeCun et al. “Handwritten digit recognition with a back-propagation network.” In: *Advances in Neural Information Processing Systems 2* (1990). Ed. by David S. Touretzky, pp. 396–404.

[163] Alexander Waibel et al. “Phoneme recognition using time-delay neural networks.” In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 37.3 (1989), pp. 328–339. DOI: 10.1109/29.21701.

[164] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. “Learning representations by back-propagating errors.” In: *Nature* 323 (1986), pp. 533–536. DOI: 10.1038/323533a0.

[165] Michael I. Jordan. *Serial order: a parallel distributed processing approach.* Tech. rep. AD-A-173989/5/XAB; ICS-8604. May 1986. URL: https://www.osti.gov/biblio/6910294.

[166] Sepp Hochreiter and Jürgen Schmidhuber. “Long short-term memory.” In: *Neural Computation* 9.8 (1997), p. 1735. DOI: 10.1162/neco.1997.9.8.1735.

[167] Kyunghyun Cho et al. “On the properties of neural machine translation: Encoder–decoder approaches.” In: *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*. Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 103–111. DOI: 10.3115/v1/W14-4012.

[168] Gilles Louppe et al. “QCD-aware recursive neural networks for jet physics.” In: *Journal of High Energy Physics* 2019.57 (2019). DOI: 10.1007/jhep01(2019)057. arXiv: 1702.00748.

[169] Shannon Egan et al. “Long Short-Term Memory (LSTM) networks with jet constituents for boosted top tagging at the LHC.” 2017. URL: https://arxiv.org/abs/1711.09059.

[170] Taoli Cheng. “Recursive neural networks in quark/gluon tagging.” In: *Computing and Software for Big Science* 2 (2018), p. 3. DOI: 10.1007/s41781-018-0007-y. arXiv: 1711.02633.

[171] Patrick T. Komiske, Eric M. Metodiev, and Jesse Thaler. “Energy flow networks: Deep sets for particle jets.” In: *Journal of High Energy Physics* 01 (2019), p. 121. DOI: 10.1007/jhep01(2019)121. arXiv: 1810.05165.

[172] Luke de Oliveira et al. “Jet-images – deep learning edition.” In: *Journal of High Energy Physics* 07 (2016), p. 069. DOI: 10.1007/jhep07(2016)069. arXiv: 1511.05190.

[173] Daniel Guest et al. “Jet flavor classification in high-energy physics with deep neural networks.” In: *Physical Review D* 94 (2016), p. 112002. DOI: 10.1103/PhysRevD.94.112002. arXiv: 1607.08633.

[174] Sebastian Macaluso and David Shih. “Pulling out all the tops with computer vision and deep learning.” In: *Journal of High Energy Physics* 10 (2018), p. 121. DOI: 10.1007/jhep10(2018)121. arXiv: 1803.00107.

[175] Kaustuv Datta and Andrew J. Larkoski. “Novel jet observables from machine learning.” In: *Journal of High Energy Physics* 03 (2018), p. 086. DOI: 10.1007/jhep03(2018)086. arXiv: 1710.01305.

[176] Anja Butter et al. “Deep-learned top tagging with a Lorentz layer.” In: *SciPost Physics* 5 (2018), p. 028. DOI: 10.21468/SciPostPhys.5.3.028. arXiv: 1707.08966.

[177] Gregor Kasieczka et al. “Deep-learning Top Taggers or The End of QCD?” In: *Journal of High Energy Physics* 05 (2017), p. 006. DOI: 10.1007/jhep05(2017)006. arXiv: 1701.08784.

[178] Patrick T. Komiske, Eric M. Metodiev, and Matthew D. Schwartz. “Deep learning in color: towards automated quark/gluon jet discrimination.” In: *Journal of High Energy Physics* 01 (2017), p. 110. DOI: 10.1007/jhep01(2017)110. arXiv: 1612.01551.

[179] Ariel Gustavo Schwartzman et al. “Image processing, computer vision, and deep learning: new approaches to the analysis and physics interpretation of LHC events.” In: *Journal of Physics: Conference Series* 762 (2016), p. 012035. DOI: 10.1088/1742-6596/762/1/012035.

[180] Gregor Kasieczka et al. “The machine learning landscape of top taggers.” In: *SciPost Physics* 7 (2019), p. 014. DOI: 10.21468/SciPostPhys.7.1.014. arXiv: 1902.09914.

[181] Javier Duarte et al. “Fast inference of deep neural networks in FPGAs for particle physics.” In: *Journal of Instrumentation* 13 (2018), P07027. DOI: 10.1088/1748-0221/13/07/P07027. arXiv: 1804.06913.

[182] Isaac Henrion et al. “Neural message passing for jet physics”. In: *Deep Learning for Physical Sciences Workshop at the 31st Conference on Neural Information Processing Systems*. Long Beach: dl4physicalsciences.github.io, 2017. URL: https://dl4physicalsciences.github.io/files/nips_dlps_2017_29.pdf.

[183] Huilin Qu and Loukas Gouskos. “ParticleNet: Jet tagging via particle clouds.” In: *Physical Review D* 101 (2020), p. 056019. DOI: 10.1103/PhysRevD.101.056019. arXiv: 1902.08570.

[184] Murat Abdughani et al. “Probing stop pair production at the LHC with graph neural networks.” In: *Journal of High Energy Physics* 08 (2019), p. 055. DOI: 10.1007/jhep08(2019)055. arXiv: 1807.09088.

[185] Nicholas Choma et al. “Graph Neural Networks for IceCube Signal Classification”. 2018. URL: https://arxiv.org/abs/1809.06166.

[186] Steven Farrell et al. “Novel deep learning methods for track reconstruction.” In: *4th International Workshop Connecting The Dots*. 2018. arXiv: 1810.06111.

[187] Jesus Arjona Martínez et al. “Pileup mitigation at the Large Hadron Collider with graph neural networks.” In: *European Physical Journal Plus* 134 (2019), p. 333. DOI: 10.1140/epjp/i2019-12710-3. arXiv: 1810.07988.

[188] Shah Rukh Qasim et al. “Learning representations of irregular particle-detector geometry with distance-weighted graph networks.” In: *European Physical Journal C* 79 (2019), p. 608. DOI: 10.1140/epjc/s10052-019-7113-9. arXiv: 1902.07987.

[189] Evan Coleman et al. “The importance of calorimetry for highly-boosted jet substructure.” In: *Journal of Instrumentation* 13 (2018). DOI: 10.1088/1748-0221/13/01/T01003. arXiv: 1709.08705 [hep-ph].

[190] Javier Mauricio Duarte et al. *hls4ml LHC jet dataset (30 particles).* Jan. 2020. DOI: 10.5281/zenodo.3601436.

[191] Javier Mauricio Duarte et al. *hls4ml LHC jet dataset (50 particles).* Jan. 2020. DOI: 10.5281/zenodo.3601443.

[192] Javier Mauricio Duarte et al. *hls4ml LHC jet dataset (100 particles).* Jan. 2020. DOI: 10.5281/zenodo.3602254.

[193] Duarte, Javier Mauricio and others. *hls4ml LHC jet dataset (150 particles).* Jan. 2020. DOI: 10.5281/zenodo.3602260.

[194] Matteo Cacciari, Gavin P. Salam, and Gregory Soyez. “FastJet user manual.” In: *European Physical Journal C* 72 (2012), p. 1896. DOI: 10.1140/epjc/s10052-012-1896-2. arXiv: 1111.6097.

[195] Jannicke Pearkes et al. “Jet constituents for deep neural network based top quark tagging.” In: arXiv:1704.02124 (2017). arXiv: 1704.02124 [hep-ex].

[196] *CERN Open Data Portal*. http://opendata.cern.ch. 2014.

[197] Lisa Randall and Raman Sundrum. “Large mass hierarchy from a small extra dimension.” In: *Physical Review Letters* 83 (1999), p. 3370. DOI: 10.1103/PhysRevLett.83.3370. arXiv: hep-ph/9905221.

[198] Richard D. Ball et al. “Parton distributions with LHC data.” In: *Nuclear Physics B* 867 (2013), p. 244. DOI: 10.1016/j.nuclphysb.2012.10.003. arXiv: 1207.1303.

[199] CMS Collaboration. *Sample with jet, track and secondary vertex properties for Hbb tagging ML studies.* CERN Open Data Portal. 2019. DOI: 10.7483/OPENDATA.CMS.JGJX.MS7Q.

[200] CMS Collaboration. “Particle-flow reconstruction and global event description with the CMS detector.” In: *Journal of Instrumentation* 12 (2017), P10003. DOI: 10.1088/1748-0221/12/10/P10003. arXiv: 1706.04965.

[201] Mrinal Dasgupta et al. “Towards an understanding of jet substructure.” In: *Journal of High Energy Physics* 09 (2013), p. 029. DOI: 10.1007/jhep09(2013)029. arXiv: 1307.0007.

[202] Jonathan M. Butterworth et al. "Jet substructure as a New Higgs Search Channel at the LHC." In: *Physical Review Letters* 100 (2008), p. 242001. DOI: 10.1103/PhysRevLett.100.242001. arXiv: 0802.2470.

[203] Andrew J. Larkoski et al. "Soft Drop." In: *Journal of High Energy Physics* 05 (2014), p. 146. DOI: 10.1007/jhep05(2014)146. arXiv: 1402.2657.

[204] CMS Collaboration. *Performance of deep tagging algorithms for boosted double quark jet topology in proton-proton collisions at 13 TeV with the Phase-0 CMS detector.* CMS Detector Performance Note CMS-DP-2018-046. 2018. URL: http://cds.cern.ch/record/2630438.

[205] Daniele Bertolini et al. "Pileup per particle identification." In: *Journal of High Energy Physics* 10 (2014), p. 059. DOI: 10.1007/jhep10(2014)059. arXiv: 1407.6013.

[206] Adam Paszke et al. "PyTorch: An imperative style, high-performance deep learning library." In: *Advances in Neural Information Processing Systems 32*. Ed. by H. Wallach et al. Red Hook, New York: Curran Associates, Inc., 2019, p. 8026. URL: http://papers.nips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.

[207] The GPyOpt authors. *GPyOpt: A Bayesian optimization framework in Python.* http://github.com/SheffieldML/GPyOpt. 2016.

[208] GPy. *GPy: A Gaussian process framework in Python.* http://github.com/SheffieldML/GPy. 2012.

[209] Vinod Nair and Geoffrey E. Hinton. "Rectified linear units improve restricted Boltzmann machines." In: *Proceedings of International Conference on Machine Learning*. Vol. 27. June 2010, pp. 807–814.

[210] Djork-Arne Clevert, Thomas Unterthiner, and Sepp Hochreiter. "Fast and accurate deep network learning by Exponential Linear Units (ELUs)." In: *CoRR* arXiv:1511.07289 (2015). arXiv: 1511.07289.

[211] Gunter Klambauer et al. "Self-normalizing neural networks." In: *CoRR* arXiv:1706.02515 (2017). arXiv: 1706.02515.

[212] Diederik P. Kingma and Jimmy Ba. "Adam: A method for stochastic optimization." In: *3rd International Conference on Learning Representations, International Conference on Learning Representations 2015*. San Diego: International Conference on Learning Representations, 2015. arXiv: 1412.6980.

[213] Matthew D. Zeiler. "ADADELTA: An adaptive learning rate method." In: *CoRR* arXiv:1212.5701 (2012). arXiv: 1212.5701.

[214] Jesse Thaler and Ken Van Tilburg. "Identifying boosted objects with N-subjettiness." In: *Journal of High Energy Physics* 03 (2011), p. 015. DOI: 10.1007/jhep03(2011)015. arXiv: 1011.2268 [hep-ph].

[215] Junjie Bai, Fang Lu, Ke Zhang, et al. *ONNX: Open Neural Network Exchange.* https://github.com/onnx/onnx. 2019.

[216] Yann LeCun, John S. Denker, and Sara A. Solla. "Optimal brain damage." In: *Advances in Neural Information Processing Systems 2*. Ed. by D. S. Touretzky. Morgan-Kaufmann, 1990, pp. 598–605. URL: https://openreview.net/pdf?id=OM0jvwB8jIp57ZJjtNEZ.

[217] Song Han, Huizi Mao, and William J. Dally. "Deep compression: Compressing deep neural network with pruning, trained quantization and Huffman coding." In: *CoRR* arXiv:1510.00149 (2015). arXiv: 1510.00149.

[218] Yu Cheng et al. "A survey of model compression and acceleration for deep neural networks." In: *CoRR* arXiv:1710.09282 (2017). arXiv: 1710.09282.

[219] Suyog Gupta et al. "Deep learning with limited numerical precision." In: *CoRR* arXiv:1502.02551 (2015). arXiv: 1502.02551.

[220] Yuan Yao, Lorenzo Rosasco, and Andrea Caponnetto. "On early stopping in gradient descent learning." In: *Constructive Approximation* 26 (2007), p. 289. ISSN: 1432-0940. DOI: 10.1007/s00365-006-0663-2.

[221] Hana Ajakan et al. "Domain-adversarial neural networks." In: *2nd Workshop on Transfer and Multi-Task Learning: Theory meets Practice at the 28th Conference on Neural Information Processing Systems*. Montreal: TMTL, 2014. arXiv: 1412.4446.

[222] Yaroslav Ganin et al. "Domain-adversarial training of neural networks." In: *Journal of Machine Learning Research* 17.59 (2016), p. 1. arXiv: 1505.07818. URL: http://jmlr.org/papers/v17/15-239.html.

[223] Gilles Louppe, Michael Kagan, and Kyle Cranmer. "Learning to pivot with adversarial networks." In: *Advances in Neural Information Processing Systems 30*. Ed. by I. Guyon et al. Red Hook, New York: Curran Associates, Inc., 2017, p. 981. arXiv: 1611.01046. URL: https://papers.nips.cc/paper/6699-learning-to-pivot-with-adversarial-networks.

[224] Chase Shimmin et al. "Decorrelated jet substructure tagging using adversarial neural networks." In: *Physical Review D* 96 (2017), p. 074034. DOI: 10.1103/PhysRevD.96.074034. arXiv: 1703.03507.

[225] ATLAS Collaboration. *Performance of mass-decorrelated jet substructure observables for hadronic two-body decay tagging in ATLAS.* ATLAS Public Note ATL-PHYS-PUB-2018-014. 2018. URL: http://cds.cern.ch/record/2630973.

[226] Layne Bradshaw et al. "Mass Agnostic Jet Taggers." In: *SciPost Physics* 8 (2020), p. 011. DOI: 10.21468/SciPostPhys.8.1.011. arXiv: 1908.08959.

[227] James Dolen et al. "Thinking outside the ROCs: Designing decorrelated taggers (DDT) for jet substructure." In: *Journal of High Energy Physics* 05 (2016), p. 156. DOI: 10.1007/jhep05(2016)156. arXiv: 1603.00027.

[228] Jerome H Friedman. "Stochastic gradient boosting." In: *Computational Statistics and Data Analysis* 38 (2002), p. 367. DOI: 10.1016/S0167-9473(01)00065-2.

[229] Jerome H Friedman. "Greedy function approximation: a gradient boosting machine." In: *Annals of Statistics* (2001), p. 1189. DOI: 10.1214/aos/1013203451.

[230] Sergey Ioffe and Christian Szegedy. "Batch normalization: Accelerating deep network training by reducing internal covariate shift." In: *Proceedings of the 32nd International Conference on Machine Learning*. Ed. by Francis Bach and David Blei. Vol. 37. Lille: PMLR, 2015, p. 448. arXiv: 1502.03167. URL: http://proceedings.mlr.press/v37/ioffe15.html.

[231] Serkan Kiranyaz et al. "Convolutional neural networks for patient-specific ECG classification." In: *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. New York: IEEE, 2015, p. 2608. DOI: 10.1109/EMBC.2015.7318926.

[232] Nitish Srivastava et al. "Dropout: A simple way to prevent neural networks from overfitting." In: *Journal of Machine Learning Research* 15 (2014), p. 1929. URL: http://jmlr.org/papers/v15/srivastava14a.html.

[233] Savas Dimopoulos et al. "Experimental signatures of low-energy gauge mediated supersymmetry breaking." In: *Physical Review Letters* 76 (1996), p. 3494. DOI: 10.1103/PhysRevLett.76.3494. arXiv: hep-ph/9601367 [hep-ph].

[234] Ben Allanach et al. "The Snowmass points and slopes: Benchmarks for SUSY searches." In: *European Physical Journal C* 25 (2002), p. 113. DOI: 10.1007/s10052-002-0949-3. arXiv: hep-ph/0202233 [hep-ph].

[235] CMS Collaboration. "Search for long-lived particles using delayed photons in proton-proton collisions at $\sqrt{s} = 13$ TeV." In: *Physical Review D* 100 (11 Dec. 2019), p. 112003. DOI: 10.1103/PhysRevD.100.112003.

[236] ATLAS Collaboration. "Search for nonpointing and delayed photons in the diphoton and missing transverse momentum final state in 8 TeV pp collisions at the LHC using the ATLAS detector." In: *Physical Review D* 90 (2014), p. 112005. DOI: 10.1103/PhysRevD.90.112005. arXiv: 1409.5542 [hep-ex].

[237] Howard Baer et al. "ISAJET 7.69: A Monte Carlo event generator for pp, $\bar{p}p$, and e^+e^- reactions." 2003. URL: https://arxiv.org/abs/hep-ph/0312045.

[238] Scott M Lundberg and Su-In Lee. "A unified approach to interpreting model predictions." In: *Advances in Neural Information Processing Systems 30*. Ed. by I. Guyon et al. Curran Associates, Inc., 2017, pp. 4765–4774. URL: http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf.

[239] Lloyd S. Shapley. "A value for n-person games." In: *Contributions to the Theory of Games (AM-28), Volume II*. Ed. by Harold William Kuhn and Albert William Tucker. Princeton University Press, 2016, pp. 307–318. DOI: 10.1515/9781400881970-018.

[240] CMS Collaboration. "Time reconstruction and performance of the CMS electromagnetic calorimeter." In: *Journal of Instrumentation* 5.03 (Mar. 2010), T03011–T03011. DOI: 10.1088/1748-0221/5/03/t03011.

[241] CMS Collaboration. "Performance of photon reconstruction and identification with the CMS detector in proton-proton collisions at $\sqrt{s} = 8$ TeV." In: *Journal of Instrumentation* 10.08 (Aug. 2015), P08010–P08010. DOI: 10.1088/1748-0221/10/08/p08010.

[242] ATLAS Collaboration. "Observation of a new particle in the search for the Standard Model Higgs boson with the ATLAS detector at the LHC." In: *Physics Letters B* B716 (2012), pp. 1–29. DOI: 10.1016/j.physletb.2012.08.020. arXiv: 1207.7214 [hep-ex].

[243] CMS Collaboration. "Observation of a new boson at a mass of 125 GeV with the CMS experiment at the LHC." In: *Physics Letters B* B716 (2012), pp. 30–61. DOI: 10.1016/j.physletb.2012.08.021. arXiv: 1207.7235 [hep-ex].

[244] CDF Collaboration. "Global search for new physics with 2.0 fb^{-1} at CDF." In: *Physical Review D* D79 (2009), p. 011101. DOI: 10.1103/PhysRevD.79.011101. arXiv: 0809.3781 [hep-ex].

[245] D0 Collaboration. "Model independent search for new phenomena in $p\bar{p}$ collisions at $\sqrt{s} = 1.96$ TeV." In: *Physical Review D* D85 (2012), p. 092015. DOI: 10.1103/PhysRevD.85.092015. arXiv: 1108.5362 [hep-ex].

[246] H1 Collaboration. "A general search for new phenomena at HERA." In: *Physics Letters B* B674 (2009), pp. 257–268. DOI: 10.1016/j.physletb.2009.03.034. arXiv: 0901.0507 [hep-ex].

[247] CMS Collaboration. *MUSiC, a Model Unspecific Search for New Physics, in pp Collisions at* $\sqrt{s} = 8$ TeV. Tech. rep. CMS-PAS-EXO-14-016. Geneva: CERN, 2017. URL: https://cds.cern.ch/record/2256653.

[248] ATLAS Collaboration. "A strategy for a general search for new phenomena using data-derived signal regions and its application within the ATLAS experiment." In: *European Physical Journal C* 79.2 (2019), p. 120. DOI: 10.1140/epjc/s10052-019-6540-y. arXiv: 1807.07447 [hep-ex].

[249] ATLAS Collaboration. "Performance of the ATLAS trigger system in 2015." In: *European Physical Journal C* 77.5 (2017), p. 317. DOI: 10.1140/epjc/s10052-017-4852-3. arXiv: 1611.09661 [hep-ex].

[250] CMS Collaboration. "The CMS trigger system." In: *Journal of Instrumentation* 12.01 (2017), P01020. DOI: 10.1088/1748-0221/12/01/P01020. arXiv: 1609.02366 [physics.ins-det].

[251] Diederik P. Kingma and Max Welling. "Auto-Encoding variational Bayes." In: *CoRR* (2013). arXiv: 1312.6114.

[252] Jinwon An and Sungzoon Cho. "Variational autoencoder based anomaly detection using reconstruction probability." In: *Special Lecture on IE* 2 (2015), pp. 1–18.

[253] Louis Lyons. "Open statistical issues in Particle Physics." In: *The Annals of Applied Statistics* 2.3 (Sept. 2008). ISSN: 1932-6157. DOI: 10.1214/08-aoas163.

[254] Eilam Gross and Ofer Vitells. "Trial factors for the look elsewhere effect in high energy physics." In: *European Physical Journal C* 70 (2010), pp. 525–530. DOI: 10.1140/epjc/s10052-010-1470-8. arXiv: 1005.1891 [physics.data-an].

[255] Raffaele Tito D'Agnolo and Andrea Wulzer. "Learning new physics from a machine." In: *Physical Review D* D99.1 (2019), p. 015014. DOI: 10.1103/PhysRevD.99.015014. arXiv: 1806.02350 [hep-ph].

[256] Jack H. Collins, Kiel Howe, and Benjamin Nachman. "Anomaly detection for resonant new physics with machine learning." In: *Physical Review Letters* 121.24 (2018), p. 241803. DOI: 10.1103/PhysRevLett.121.241803. arXiv: 1805.02664 [hep-ph].

[257] Andrea De Simone and Thomas Jacques. "Guiding new physics searches with unsupervised learning." In: *The European Physical Journal C* 79.4 (Mar. 2019). ISSN: 1434-6052. DOI: 10.1140/epjc/s10052-019-6787-3. URL: http://dx.doi.org/10.1140/epjc/s10052-019-6787-3.

[258] Jan Hajer et al. "Novelty detection meets collider physics." In: *Physical Review D* 101.7 (Apr. 2020). ISSN: 2470-0029. DOI: 10.1103/physrevd.101.076015.

[259] Adrian Alan Pol et al. *Detector monitoring with artificial neural networks at the CMS experiment at the CERN Large Hadron Collider.* 2018. arXiv: 1808.00911 [physics.data-an].

[260] Adrian Alan Pol et al. "Anomaly detection using Deep Autoencoders for the assessment of the quality of the data acquired by the CMS experiment." In: *23rd International Conference on Computing in High Energy and Nuclear Physics (CHEP 2018)*. Geneva, July 2018. DOI: 10.1051/epjconf/201921406008.

[261] ATLAS Collaboration. *Deep generative models for fast shower simulation in ATLAS.* Tech. rep. ATL-SOFT-PUB-2018-001. Geneva: CERN, July 2018. URL: http://cds.cern.ch/record/2630433.

[262] Theo Heimel et al. "QCD or What?" In: *SciPost Physics* 6.3 (2019), p. 030. DOI: 10.21468/SciPostPhys.6.3.030. arXiv: 1808.08979.

[263] Marco Farina, Yuichiro Nakai, and David Shih. "Searching for new physics with deep autoencoders." Apr. 2020.

[264] Bernhard Schölkopf et al. "Estimating the support of a high-dimensional distribution." In: *Neural computation* 13.7 (2001), pp. 1443–1471.

[265] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. "Isolation forest." In: (2008). DOI: 10.1109/ICDM.2008.17.

[266] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. "Isolation-based anomaly detection." In: *ACM Transactions on Knowledge Discovery from Data (TKDD)* 6.1 (2012), p. 3.

[267] Charu C Aggarwal. "Outlier analysis." In: *Data mining*. Springer. 2015, pp. 237–263.

[268] Mevlana Gemici et al. "Generative temporal models with memory." In: *CoRR* abs/1702.04649 (2017). arXiv: 1702.04649.

[269] DELPHES Collaboration. "DELPHES 3, A modular framework for fast simulation of a generic collider experiment." In: *Journal of High Energy Physics* 02 (2014), p. 057. DOI: 10.1007/jhep02(2014)057. arXiv: 1307.6346 [hep-ex].

[270] CMS Collaboration. "Technical proposal for the Phase-II upgrade of the CMS detector." In: (2015). eprint: CERN-LHCC-2015-010,LHCC-P-008,CMS-TDR-15-02. URL: https://cds.cern.ch/record/2020886.

[271] Irina Higgins et al. "beta-VAE: Learning basic visual concepts with a constrained variational framework." In: *International Conference on Learning Representations*. 2016. URL: https://openreview.net/forum?id=Sy2fzU9gl.

[272] Jakub M. Tomczak and Max Welling. "VAE with a VampPrior." In: Proceedings of Machine Learning Research 84 (2018). Ed. by Amos J. Storkey and Fernando Pérez-Cruz, pp. 1214–1223. URL: http://proceedings.mlr.press/v84/tomczak18a.html.

[273] Fabian Pedregosa et al. "Scikit-learn: Machine learning in Python". In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830. URL: http://jmlr.org/papers/v12/pedregosa11a.html.

[274] Serguei Chatrchyan et al. "Search for pair production of third-generation leptoquarks and top squarks in pp collisions at $\sqrt{s} = 7$ TeV." In: *Physical Review Letters* 110.8 (2013), p. 081801. DOI: 10.1103/PhysRevLett.110.081801. arXiv: 1210.5629 [hep-ex].

[275] CMS Collaboration. "Search for third-generation scalar leptoquarks and heavy right-handed neutrinos in final states with two tau leptons and two jets in proton-proton collisions at $\sqrt{s}$ = 13 TeV." In: *Journal of High Energy Physics* 07 (2017), p. 121. DOI: 10.1007/jhep07(2017)121. arXiv: 1703.03995 [hep-ex].

[276] Javier Duarte et al. "Fast inference of deep neural networks in FPGAs for particle physics." In: *Journal of Instrumentation* 13.07 (2018), P07027. DOI: 10.1088/1748-0221/13/07/P07027. arXiv: 1804.06913 [physics.ins-det].

[277] Jennifer Ngadiuba et al. "Compressing deep neural networks on FPGAs to binary and ternary precision with hls4ml." In: *Machine Learning: Science and Technology* (2020). DOI: 10.1088/2632-2153/aba042. arXiv: 2003.06308 [cs.LG].

[278] Yutaro Iiyama et al. "Distance-weighted graph neural networks on FPGAs for real-time particle reconstruction in high energy physics." In: *Frontiers in Big Data* 3 (2020), p. 598927. DOI: 10.3389/fdata.2020.598927. arXiv: 2008.03601 [hep-ex].

[279] Thea Aarrestad et al. "Fast convolutional neural networks on FPGAs with hls4ml." In: *Machine Learning: Science and Technology* 2 (2021), p. 045015. DOI: 10.1088/2632-2153/ac0ea1. arXiv: 2101.05108 [cs.LG].

[280] Aneesh Heintz et al. "Accelerated charged particle tracking with graph neural networks on FPGAs." In: *34th Conference on Neural Information Processing Systems*. Nov. 2020. arXiv: 2012.01563 [physics.ins-det].

[281] Sioni Summers et al. "Fast inference of Boosted Decision Trees in FPGAs for particle physics." In: *Journal of Instrumentation* 15.05 (2020), P05026. DOI: 10.1088/1748-0221/15/05/P05026. arXiv: 2002.02534 [physics.comp-ph].

[282] Claudionor N. Coelho. *QKeras*. 2019. URL: https://github.com/google/qkeras.

[283] Claudionor N. Coelho et al. "Automatic heterogeneous quantization of deep neural networks for low-latency inference on the edge for particle detectors." In: *Nature Machine Intelligence* (2021). DOI: 10.1038/s42256-021-00356-5. arXiv: 2006.10159 [physics.ins-det].

[284] Houssam Zenati et al. "Adversarially learned anomaly detection." In: *Proceedings of the 20th IEEE International Conference on Data Mining (ICDM)*. 2018. eprint: arXiv:1812.02288.

[285] Ian Goodfellow et al. "Generative adversarial nets." In: *Advances in Neural Information Processing Systems*. Ed. by Z. Ghahramani et al. Vol. 27. Curran Associates, Inc., 2014. URL: https://proceedings.neurips.cc/paper/2014/file/5ca3e9b122f61f8f06494c97b1afccf3-Paper.pdf.

[286] Thomas Schlegl et al. “Unsupervised anomaly detection with generative adversarial networks to guide marker discovery.” In: *Information Processing in Medical Imaging*. Cham: Springer International Publishing, 2017, pp. 146–157.

[287] Antonia Creswell and Anil Anthony Bharath. “Inverting the generator of a generative adversarial network.” In: *CoRR* abs/1611.05644 (2016). arXiv: 1611.05644. URL: http://arxiv.org/abs/1611.05644.

[288] Yuhuai Wu et al. “On the quantitative analysis of decoder-based generative models.” In: *CoRR* abs/1611.04273 (2016). arXiv: 1611.04273. URL: http://arxiv.org/abs/1611.04273.

[289] Elies Gherbi et al. “An encoding adversarial network for anomaly detection.” In: *11th Asian Conference on Machine Learning (ACML 2019)*. Vol. 101. JMLR: Workshop and Conference Proceedings. Nagoya, Japan, Nov. 2019, pp. 1–16. URL: https://hal.archives-ouvertes.fr/hal-02421274.

[290] Jeff Donahue, Philipp Krähenbühl, and Trevor Darrell. “Adversarial feature learning.” In: *International Conference on Learning Representations*. 2017. eprint: arXiv:1605.09782.

[291] Andrew L. Maas, Awni Y. Hannun, and Andrew Y. Ng. “Rectifier nonlinearities improve neural network acoustic models.” In: *International Conference on Machine Learning Workshop on Deep Learning for Audio, Speech and Language Processing*. 2013.

[292] Alan Agresti and Brent A. Coull. “Approximate is better than ‘exact’ for interval estimation of binomial proportions.” In: *The American Statistician* 52.2 (1998).

[293] Oliver Aberle et al. *High-Luminosity Large Hadron Collider (HL-LHC): Technical design report*. CERN Yellow Reports: Monographs. Geneva: CERN, 2020. DOI: 10.23731/CYRM-2020-0010. URL: http://cds.cern.ch/record/2749422.

[294] Christophe Ochando. “HGCAL: A high-granularity calorimeter for the endcaps of CMS at HL-LHC.” In: *Journal of Physics: Conference Series* 928 (2017), 012025. 4 p. DOI: 10.1088/1742-6596/928/1/012025.

[295] Giacomo Sguazzoni. “The CMS pixel detector for the High Luminosity LHC”. In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 958 (2020), p. 162089. ISSN: 0168-9002. DOI: 10.1016/j.nima.2019.04.043.

[296] CMS Collaboration. *CMS Offline and Computing: Public results*. URL: https://twiki.cern.ch/twiki/bin/view/CMSPublic/CMSOfflineComputingResults (visited on 08/10/2021).

[297] Andrea Bocci et al. “Bringing heterogeneity to the CMS software framework.” In: *EPJ Web of Conferences* 245 (2020), p. 05009. ISSN: 2100-014X. DOI: 10.1051/epjconf/202024505009.

[298] Andrea Bocci et al. “Heterogeneous reconstruction of tracks and primary vertices with the CMS pixel tracker.” In: *Frontiers in Big Data* 3 (2020), p. 49. ISSN: 2624-909X. DOI: 10.3389/fdata.2020.601728.

[299] Henry J Kelley. “Gradient theory of optimal flight paths.” In: *Ars Journal* 30.10 (1960), pp. 947–954.

[300] Herbert Robbins and Sutton Monro. “A stochastic approximation method.” In: *The Annals of Mathematical Statistics* 22.3 (1951), pp. 400–407. DOI: 10.1214/aoms/1177729586.

[301] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. “Multilayer feedforward networks are universal approximators.” In: *Neural Networks* 2.5 (1989), pp. 359–366. ISSN: 0893-6080. DOI: 10.1016/0893-6080(89)90020-8.

[302] Yann LeCun and Yoshua Bengio. “Convolutional networks for images, speech, and time series.” In: *The Handbook of Brain Theory and Neural Networks*. Ed. by Michael A. Arbib. Vol. 3361. Cambridge, Massachusetts: MIT Press, 1995, p. 255. ISBN: 0262511029.

[303] Celia Fernández Madrazo et al. “Application of a convolutional neural network for image classification for the analysis of collisions in high energy physics.” In: *EPJ Web of Conferences* 214 (2019). Ed. by A. Forti et al., p. 06017. DOI: 10.1051/epjconf/201921406017. arXiv: 1708.07034 [cs.CV].

[304] Rami Al-Rfou et al. “Theano: A Python framework for fast computation of mathematical expressions.” In: (2016). arXiv: 1605.02688 [cs.SC]. URL: http://arxiv.org/abs/1605.02688.

[305] Dustin Anderson, Maria Spiropulu, and Jean-Roch Vlimant. “An MPI-based Python framework for distributed training with Keras.” In: (2017). arXiv: 1712.05878 [cs]. URL: https://arxiv.org/abs/1712.05878.

[306] Gao Huang et al. “Densely connected convolutional networks.” In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017. arXiv: 1608.06993.

[307] Pierre Baldi et al. “Searching for exotic particles in high-energy physics with deep learning.” In: *Nature Communication* 5 (July 2014), p. 4308. URL: http://dx.doi.org/10.1038/ncomms5308.

[308] Alexander Radovic et al. “Machine learning at the energy and intensity frontiers of particle physics.” In: *Nature* 560.7716 (2018), pp. 41–48. DOI: 10.1038/s41586-018-0361-2.

[309] Wahid Bhimji et al. “Deep neural networks for physics analysis on low-level whole-detector data at the LHC.” In: *Journal of Physics: Conference Series* 1085.2 (2018). arXiv: 1711.03573 [hep-ex]. URL: https://inspirehep.net/record/1635524/files/arXiv:1711.03573.pdf.

[310] Vladimir Vava Gligorov and Michael Williams. “Efficient, reliable and fast high-level triggering using a bonsai boosted decision tree.” In: *Journal of Instrumentation* 8.02 (2013), P02013. DOI: 10.1088/1748-0221/8/02/p02013.

[311] Tatiana Likhomanenko et al. “LHCb topological trigger reoptimization.” In: *Journal of Physics: Conference Series* 664.8 (2015), p. 082025. DOI: 10.1088/1742-6596/664/8/082025.

[312] Pierre-Hugues Beauchemin. “Real time data analysis with the ATLAS Trigger at the LHC in Run-2.” In: *21st IEEE Real Time Conference (RT2018) Williamsburg, Virginia, June 11-15, 2018*. 2018. arXiv: 1806.08475 [hep-ex].

[313] CMS Collaboration. “Boosted decision trees in the level-1 muon endcap trigger at CMS.” In: CMS-CR-2017-357. Geneva, Oct. 2017. URL: http://cds.cern.ch/record/2290188.

[314] Valentin Kuznetsov. *TensorFlow as a Service.* https://github.com/vkuznet/TFaaS.

[315] Chris Allton et al. *Computing resources scrutiny group report.* Tech. rep. CERN-RRB-2017-125. Geneva: CERN, Sept. 2017. URL: http://cds.cern.ch/record/2284575.

[316] Tero Karras et al. “Progressive growing of GANs for improved quality, stability, and variation.” In: *International Conference on Learning Representations* (2018). arXiv: 1710.10196.

[317] Jun-Yan Zhu et al. “Unpaired image-to-image translation using cycle-consistent adversarial networks.” In: *CoRR* (2017). arXiv: 1703.10593.

[318] Michela Paganini et al. “CaloGAN : Simulating 3D high energy particle showers in multilayer electromagnetic calorimeters with generative adversarial networks.” In: *Physical Review D* 97.1 (2018), p. 014021. DOI: 10.1103/PhysRevD.97.014021. arXiv: 1712.10321 [hep-ex].

[319] Federico Carminati et al. “Calorimetry with deep learning: Particle classification, energy regression, and simulation for high-energy physics.” In: (2017). URL: https://dl4physicalsciences.github.io/files/nips_dlps_2017_15.pdf.

[320] Martin Erdmann, Jonas Glombitza, and Thorben Quast. “Precise simulation of electromagnetic calorimeter showers using a Wasserstein generative adversarial network.” In: *Computing and Software for Big Science* 3 (4 2019).

DOI: 10.1007/s41781-018-0019-7. arXiv: 1807.01954 [physics.ins-det].

[321] Pasquale Musella and Francesco Pandolfi. "Fast and accurate simulation of particle detectors using generative adversarial neural networks." In: *Computing and Software for Big Science* 2 (8 2018). DOI: 10.1007/s41781-018-0015-y. arXiv: 1805.00850 [hep-ex].

[322] Martin Erdmann et al. "Generating and refining particle detector simulations using the Wasserstein distance in adversarial networks." In: *Computing and Software for Big Science* 2.1 (2018), p. 4. DOI: 10.1007/s41781-018-0008-x. arXiv: 1802.03325 [astro-ph.IM].

[323] Luke de Oliveira, Michela Paganini, and Benjamin Nachman. "Learning particle physics by example: Location-aware generative adversarial networks for physics synthesis." In: *Computing and Software for Big Science* 1.1 (2017), p. 4. DOI: 10.1007/s41781-017-0004-6. arXiv: 1701.05927 [stat.ML].

[324] Viktoria Chekalina et al. "Generative models for fast calorimeter simulation: The LHCb case." In: *EPJ Web of Conferences* 214 (2019). Ed. by A. Forti et al., p. 02034. ISSN: 2100-014X. DOI: 10.1051/epjconf/201921402034.

[325] Joshua Lin, Wahid Bhimji, and Benjamin Nachman. "Machine learning templates for QCD factorization in the search for physics beyond the Standard Model." In: *Journal of High Energy Physics* 2019.5 (May 2019). ISSN: 1029-8479. DOI: 10.1007/jhep05(2019)181.

[326] Riccardo Di Sipio et al. "DijetGAN: a generative-adversarial network approach for the simulation of QCD dijet events at the LHC." In: *Journal of High Energy Physics* 2019.8 (Aug. 2019). ISSN: 1029-8479. DOI: 10.1007/jhep08(2019)110.

[327] Bobak Hashemi et al. "LHC analysis-specific datasets with generative adversarial networks." In: (2019). eprint: 1901.05282. URL: https://arxiv.org/abs/1901.05282.

[328] Steve Farrell et al. "Next generation generative neural networks for HEP." In: *2018 International Conference on Computing in High-Energy and Nuclear Physics, Sofia, Bulgaria* (2018). DOI: 10.1051/epjconf/201921409005.

[329] Alec Radford, Luke Metz, and Soumith Chintala. "Unsupervised representation learning with deep convolutional generative adversarial networks." In: *CoRR* (2015). arXiv: 1511.06434.

[330] Mehdi Mirza and Simon Osindero. "Conditional generative adversarial nets." In: (2014). arXiv: 1411.1784 [cs.LG].

[331] Tim Salimans et al. "Improved techniques for training GANs." In: *Conference on Neural Information Processing Systems 29* (2016), pp. 2234–2242. arXiv: 1606.03498.

[332] Xudong Mao et al. “Multi-class generative adversarial networks with the L2 loss function.” In: *CoRR* (2016). arXiv: 1611.04076.

[333] Martin Arjovsky, Soumith Chintala, and Léon Bottou. “Wasserstein generative adversarial networks.” In: *International Conference on Machine Learning 34*. Vol. 70. PMLR, Aug. 2017, pp. 214–223. arXiv: 1701.07875.

[334] Matteo Cacciari, Gavin P. Salam, and Gregory Soyez. “SoftKiller: A particle-level pileup removal method.” In: *European Physical Journal* 75.2 (2015), p. 59. DOI: 10.1140/epjc/s10052-015-3267-2. arXiv: 1407.0408 [hep-ph].

[335] Geoffrey C Fox and Stephen Wolfram. “Event shapes in $e^+ e^-$ annihilation.” In: *Nuclear Physics B* 149.3 (1979), pp. 413–496.

[336] Jack Kiefer and Jacob Wolfowitz. “Stochastic estimation of the maximum of a regression function.” In: *The Annals of Mathematical Statistics* 23.3 (1952), pp. 462–466. ISSN: 00034851. URL: http://www.jstor.org/stable/2236690.

[337] Leon Bottou, Frank E. Curtis, and Jorge Nocedal. *Optimization methods for large-scale machine learning*. 2018. arXiv: 1606.04838 [stat.ML].

[338] Message Passing Forum. *MPI: A message-passing interface standard.* Tech. rep. USA, 1994. URL: https://www.mpi-forum.org/docs/mpi-3.0/mpi30-report.pdf.

[339] Jeffrey Dean et al. “Large scale distributed deep networks.” In: *NIPS*. 2012.

[340] Sixin Zhang, Anna Choromanska, and Yann LeCun. “Deep learning with elastic averaging SGD.” In: *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*. NIPS’15. Montreal, Canada: MIT Press, 2015, pp. 685–693.

[341] Amazon SageMaker Team. *Introduction to model parallelism.* URL: https://docs.aws.amazon.com/sagemaker/latest/dg/model-parallel-intro.html.

[342] Johannes Albrecht et al. “A roadmap for HEP software and computing R&D for the 2020s.” In: *Computing and Software for Big Science* 3.1 (2019), p. 7. DOI: 10.1007/s41781-018-0018-8. arXiv: 1712.06982 [physics.comp-ph].

[343] CMS Collaboration. “Mini-AOD: A new analysis data format for CMS.” In: *Journal of Physics: Conference Series* 664 (2015), p. 072052. DOI: 10.1088/1742-6596/664/7/072052. arXiv: 1702.04685.

[344] Ishaan Gulrajani et al. *Improved training of Wasserstein GANs.* 2017. arXiv: 1704.00028.

[345] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. "Stochastic backpropagation and approximate inference in deep generative models." In: *Proceedings of the 31st International Conference on Machine Learning*. Vol. 32. Proceedings of Machine Learning Research. 2014. URL: http://proceedings.mlr.press/v32/rezende14.html.

[346] Dalila Salamani et al. "Deep generative models for fast shower simulation in ATLAS." In: *14th International Conference on e-Science*. 2018, p. 348. DOI: 10.1109/eScience.2018.00091.

[347] Dawit Belayneh et al. "Calorimetry with deep learning: Particle simulation and reconstruction for collider physics." In: *European Physical Journal C* 80.7 (2020), p. 688. DOI: 10.1140/epjc/s10052-020-8251-9. arXiv: 1912.06794 [physics.ins-det].

[348] Erik Buhmann et al. "Getting high: High fidelity simulation of high granularity calorimeters with high speed." In: (May 2020). arXiv: 2005.05334 [physics.ins-det].

[349] Stefano Carrazza and Frédéric A. Dreyer. "Lund jet images from generative and cycle-consistent adversarial networks." In: *European Physical Journal C* 79.11 (2019), p. 979. DOI: 10.1140/epjc/s10052-019-7501-1. arXiv: 1909.01359 [hep-ph].

[350] Fady Bishara and Marc Montull. *(Machine) learning amplitudes for faster event generation.* 2019. arXiv: 1912.11055 [hep-ph].

[351] Anja Butter, Tilman Plehn, and Ramon Winterhalder. "How to GAN LHC events." In: *SciPost Physics* 7 (2019), p. 075. DOI: 10.21468/SciPostPhys.7.6.075. arXiv: 1907.03764 [hep-ph].

[352] Jesus Arjona Martínez et al. "Particle Generative Adversarial Networks for full-event simulation at the LHC and their application to pileup description." In: *Journal of Physics: Conference Series* 1525 (Apr. 2020), p. 012081. ISSN: 1742-6596. DOI: 10.1088/1742-6596/1525/1/012081.

[353] Sydney Otten et al. "Event generation and statistical sampling for physics with deep generative models and a density information buffer." In: *Nature Communications* 12.2985 (2021). DOI: 10.1038/s41467-021-22616-z. arXiv: 1901.00875 [hep-ph].

[354] Marco Bellagente et al. "Invertible networks or partons to detector and back again." In: *SciPost Physics* 9 (2020), p. 074. DOI: 10.21468/SciPostPhys.9.5.074. arXiv: 2006.06685 [hep-ph].

[355] Konstantin T. Matchev and Prasanth Shyamsundar. *Uncertainties associated with GAN-generated datasets in high energy physics.* Feb. 2020. arXiv: 2002.06307 [hep-ph].

[356] Anja Butter et al. "GANplifying event samples." In: *SciPost Physics* 10.6 (June 2021). ISSN: 2542-4653. DOI: 10.21468/scipostphys.10.6.139.

[357] Kyle S. Cranmer. "Kernel estimation in high-energy physics." In: *Computer Physics Communications* 136 (2001), pp. 198–207. DOI: 10.1016/S0010-4655(00)00243-5. arXiv: hep-ex/0011057.

[358] Maurizio Pierini and Cheng Chen. *Data augmentation at the LHC through analysis- specific fast simulation with deep learning: W+jet training/test dataset.* Oct. 2020. DOI: 10.5281/zenodo.4080943.

[359] Maurizio Pierini and Cheng Chen. *Data augmentation at the LHC through analysis- specific fast simulation with deep learning: W+jet large test dataset.* Oct. 2020. DOI: 10.5281/zenodo.4080968.

[360] Kaoru Hagiwara et al. "Fast computation of MadGraph amplitudes on graphics processing unit (GPU)." In: *European Physical Journal C* 73 (2013), p. 2608. DOI: 10.1140/epjc/s10052-013-2608-2. arXiv: 1305.0708 [physics.comp-ph].

[361] Matthew D. Klimek and Maxim Perelstein. "Neural network-based approach to phase space integration." In: *SciPost Physics* 9 (2020), p. 053. DOI: 10.21468/SciPostPhys.9.4.053. arXiv: 1810.11509 [hep-ph].

[362] Christina Gao, Joshua Isaacson, and Claudius Krause. "i-flow: High-dimensional integration and sampling with normalizing flows." In: *Machine Learning: Science and Technology* 1.4 (2020), p. 045023. DOI: 10.1088/2632-2153/abab62. arXiv: 2001.05486 [physics.comp-ph].

[363] Christina Gao et al. "Event generation with normalizing flows." In: *Physical Review D* 101.7 (2020), p. 076002. DOI: 10.1103/PhysRevD.101.076002. arXiv: 2001.10028 [hep-ph].

[364] Stefano Carrazza and Juan M. Cruz-Martinez. "VegasFlow: accelerating Monte Carlo simulation across multiple hardware platforms." In: *Computer Physics Communications* 254 (2020), p. 107376. DOI: 10.1016/j.cpc.2020.107376. arXiv: 2002.12921 [physics.comp-ph].

[365] CMS Collaboration. *CMSSW framework.* URL: https://github.com/cms-sw/cmssw (visited on 08/24/2021).

[366] CMS Collaboration. *CMS Open Data.* 2015. URL: http://opendata.cern.ch/search?experiment=CMS (visited on 09/21/2021).

[367] CMS Collaboration. *Simulated dataset QCD_Pt_470to600_TuneCUETP8M1_-13TeV_pythia8 in MINIAODSIM format for 2016 collision data.* CERN Open Data Portal. 2019. DOI: 10.7483/OPENDATA.CMS.HBBW.LTT4.