NOVA SCHOOL OF
SCIENCE & TECHNOLOGY

DEPARTMENT OF PHYSICS

UNCERTAINTY IN MACHINE LEARNING

A SAFETY PERSPECTIVE ON BIOMEDICAL APPLICATIONS

MARÍLIA DA SILVEIRA GOUVEIA BARANDAS

Master in Biomedical Engineering

DOCTORATE IN BIOMEDICAL ENGINEERING

NOVA University Lisbon
March, 2023

DEPARTMENT OF PHYSICS

UNCERTAINTY IN MACHINE LEARNING

A SAFETY PERSPECTIVE ON BIOMEDICAL APPLICATIONS

MARÍLIA DA SILVEIRA GOUVEIA BARANDAS

Master in Biomedical Engineering

Adviser: Hugo Filipe Silveira Gamboa
Associate Professor with Habilitation, NOVA School of Science and Technology

Examination Committee:

Chair:	Orlando Manuel Neves Duarte Teodoro *Full Professor, NOVA School of Science and Technology*
Rapporteurs:	Eyke Hüllermeier *Full Professor, Ludwig-Maximilians-Universität München*
	André Ribeiro Lourenço *Assistant Professor, Instituto Superior de Engenharia de Lisboa*
Adviser:	Hugo Filipe Silveira Gamboa *Associate Professor with Habilitation, NOVA School of Science and Technology*
Members:	Ricardo Nuno Pereira Verga e Afonso Vigário *Associate Professor with Habilitation, NOVA School of Science and Technology*
	Orlando Manuel Neves Duarte Teodoro *Full Professor, NOVA School of Science and Technology*

DOCTORATE IN BIOMEDICAL ENGINEERING

NOVA University Lisbon
March, 2023

Uncertainty in Machine Learning

Acknowledgements

I would like to express my sincere gratitude to those who have supported and guided me throughout my Ph.D. journey.

First and foremost, I am deeply grateful to my supervisor, Hugo Gamboa, for his invaluable guidance and unwavering availability. He has been pivotal to my academic development, consistently showing confidence in my abilities and fostering an environment that allows me to thrive. Moreover, I want to thank him for encouraging me to make this document more inspirational.

I would also like to extend my heartfelt appreciation to Fraunhofer Portugal for providing me with the incredible opportunity to pursue my Ph.D. Their unwavering support and readiness to help whenever needed have been invaluable. A special thank you goes to Liliana Ferreira and Inês Sousa for making this opportunity possible.

My journey would not have been the same without Duarte Folgado, who shared this experience with me. I am grateful for our scientific discussions, his unwavering support, and the inspiration he lent me throughout our time together. The memories and research achievements we have shared have made this research experience truly unforgettable.

I am also indebted to my lab colleagues at Fraunhofer for their stimulating research discussions and the fantastic work environment they helped create. Similarly, I would like to express my gratitude to MUDILab in Milan for their warm welcome and support during a crucial period of my Ph.D.

To my dear friend Bárbara Viotty, I cannot thank you enough for always being there. Your motivational quote, "A verdadeira viagem de descobrimento não consiste em procurar novas paisagens, mas em ter novos olhos (Marcel Proust)", deserves to be documented in this document.

Last but not least, I would like to express my love and appreciation to my family, whose unwavering support has been a constant source of strength throughout this journey.

“All models are wrong, but some are useful .” (George Box)

Abstract

Uncertainty is an inevitable and essential aspect of the world we live in and a fundamental aspect of human decision-making. It is no different in the realm of machine learning. Just as humans seek out additional information and perspectives when faced with uncertainty, machine learning models must also be able to account for and quantify the uncertainty in their predictions. However, the uncertainty quantification in machine learning models is often neglected. By acknowledging and incorporating uncertainty quantification into machine learning models, we can build more reliable and trustworthy systems that are better equipped to handle the complexity of the world and support clinical decision-making.

This thesis addresses the broad issue of uncertainty quantification in machine learning, covering the development and adaptation of uncertainty quantification methods, their integration in the machine learning development pipeline, and their practical application in clinical decision-making.

Original contributions include the development of methods to support practitioners in developing more robust and interpretable models, which account for different sources of uncertainty across the core components of the machine learning pipeline, encompassing data, the machine learning model, and its outputs. Moreover, these machine learning models are designed with abstaining capabilities, enabling them to accept or reject predictions based on the level of uncertainty present. This emphasizes the importance of using classification with rejection option in clinical decision support systems. The effectiveness of the proposed methods was evaluated across databases with physiological signals from medical diagnosis and human activity recognition. The results support that uncertainty quantification was important for more reliable and robust model predictions.

By addressing these topics, this thesis aims to improve the reliability and trustworthiness of machine learning models and contribute to fostering the adoption of machine-assisted clinical decision-making. The ultimate goal is to enhance the trust and accuracy of models' predictions and increase transparency and interpretability, ultimately leading to better decision-making across a range of applications.

Keywords: Machine learning; Uncertainty quantification; Classification with rejection option; Interpretability; Clinical decision making;

Resumo

A incerteza é um aspeto inevitável e essencial do mundo em que vivemos e um aspeto fundamental na tomada de decisão humana. Não é diferente no âmbito da aprendizagem automática. Assim como os seres humanos, quando confrontados com um determinado nível de incerteza exploram novas abordagens ou procuram recolher mais informação, também os modelos de aprendizagem automática devem ter a capacidade de ter em conta e quantificar o grau de incerteza nas suas previsões. No entanto, a quantificação da incerteza nos modelos de aprendizagem automática é frequentemente negligenciada. O reconhecimento e incorporação da quantificação de incerteza nos modelos de aprendizagem automática, irá permitir construir sistemas mais fiáveis, melhor preparados para apoiar a tomada de decisão clinica em situações complexas e com maior nível de confiança.

Esta tese aborda a ampla questão da quantificação de incerteza na aprendizagem automática, incluindo o desenvolvimento e adaptação de métodos de quantificação de incerteza, a sua integração no pipeline de desenvolvimento de modelos de aprendizagem automática e a sua aplicação prática na tomada de decisão clínica.

Nos contributos originais, inclui-se o desenvolvimento de métodos para apoiar os profissionais de desenvolvimento na criação de modelos mais robustos e interpretáveis, que tenham em consideração as diferentes fontes de incerteza nos diversos componentes-chave do pipeline de aprendizagem automática: os dados, o modelo de aprendizagem automática e os seus resultados. Adicionalmente, os modelos de aprendizagem automática são construídos com a capacidade de se abster, o que permite aceitar ou rejeitar uma previsão com base no nível de incerteza presente, o que realça a importância da utilização de modelos de classificação com a opção de rejeição em sistemas de apoio à decisão clínica. A eficácia dos métodos propostos foi avaliada em bases de dados contendo sinais fisiológicos provenientes de diagnósticos médicos e reconhecimento de atividades humanas. As conclusões sustentam a importância da quantificação da incerteza nos modelos de aprendizagem automática para obter previsões mais fiáveis e robustas.

Desenvolvendo estes tópicos, esta tese pretende aumentar a fiabilidade e credibilidade dos modelos de aprendizagem automática, promovendo a utilização e desenvolvimento dos sistemas de apoio à decisão clínica. O objetivo final é aumentar o grau de confiança e a fiabilidade das previsões dos modelos, bem como, aumentar a transparência e interpretabilidade, proporcionando uma melhor tomada de decisão numa variedade de aplicações.

Palavras-chave: Aprendizagem automática; Quantificação de incerteza; Classificação com opção de rejeição; Interpretabilidade; Tomada de decisão clinica.

Contents

List of Figures **xiii**

List of Tables **xix**

Acronyms **xxii**

1 Introduction **1**

1.1 Motivation 1

1.2 Applications of uncertainty 2

1.3 Research paths and contributions 3

1.4 Thesis structure 4

2 The Language of Uncertainty **7**

2.1 Definition of uncertainty 7

2.2 Sources of uncertainty 9

2.2.1 Classification example 11

2.3 Representing uncertainty 13

2.3.1 Uncertainty view 14

2.3.2 Anomaly view 14

2.3.3 Novelty view 15

2.3.4 Abstention view 16

2.4 Final remarks 17

3 Uncertainty Quantification in Machine Learning **18**

3.0.1 Notation 19

3.1 Uncertainty estimation: Single methods 20

3.1.1 Internal approaches 20

3.1.2 External approaches 22

3.2 Uncertainty estimation: Bayesian methods 25

3.2.1 Stochastic methods 28

3.2.2 Ensemble methods . . . 30
3.3 Uncertainty evaluation measures . . . 31
3.4 Calibration . . . 33
3.5 Experimental analysis . . . 35
3.5.1 Datasets . . . 35
3.5.2 Classification with rejection option . . . 36
3.5.3 Out-of-distribution detection . . . 39
3.5.4 Discussion . . . 44
3.6 Final remarks . . . 45

4 Uncertainty for Model Design **47**
4.1 Uncertainty-based rejection . . . 48
4.1.1 Preliminaries . . . 48
4.1.2 Uncertainty for model selection . . . 52
4.1.3 Uncertainty for models' combination . . . 57
4.1.4 Interpretability of rejection via uncertainty visualization . . . 60
4.2 Explainability meets uncertainty quantification . . . 62
4.2.1 Preliminaries . . . 64
4.2.2 Proposed approach . . . 67
4.2.3 Datasets . . . 71
4.2.4 Experimental setup . . . 71
4.2.5 Results and Discussion . . . 72
4.2.6 Conclusion . . . 79
4.3 Final remarks . . . 81

5 Uncertainty for Clinical Decision Making **83**
5.1 Preliminaries . . . 84
5.1.1 Multi-label classification . . . 84
5.1.2 Dataset shift . . . 85
5.1.3 Active Learning . . . 86
5.1.4 Electrocardiogram (ECG) classification . . . 87
5.1.5 Summary . . . 89
5.2 Methods . . . 91
5.2.1 ECG data preparation . . . 91
5.2.2 Uncertainty methods training . . . 95
5.2.3 Validation approach . . . 96
5.3 Experimental Results . . . 99
5.3.1 External validation . . . 99
5.3.2 Calibration . . . 103
5.3.3 Uncertainty quantification . . . 105
5.3.4 Classification with rejection-option . . . 108

5.3.5 Data quality . . . 112
5.3.6 Active learning . . . 116
5.4 Discussion . . . 117
5.5 Final remarks . . . 120

6 Conclusions **122**
6.1 Broader Impact . . . 123
6.2 Review of major contributions . . . 123
6.2.1 List of publications . . . 127
6.3 Future work . . . 128
6.3.1 Current challenges . . . 129
6.3.2 Specific future research directions . . . 129

Bibliography **131**

Appendices

A KUE Experiments: Dataset Details and Additional Results **147**
A.1 Supplementary details for Bacteria dataset . . . 147
A.2 KUE using different classifiers . . . 148
A.3 Area Under the Precision-Recall (AUPR)-In and AUPR-Out Results . . . 149
A.4 Accuracy Rejection Curves . . . 152

B Multimodal Experiments: Preprocessing and selected models **154**

C Multi-label ECG Classification: Annotations details and statistical analysis **159**
C.1 Datasets annotations . . . 159
C.2 Statistical analysis . . . 160

List of Figures

1.1 Overview of the thesis structure, illustrating the primary research works that were adapted into sections or combined with other texts within the respective chapters. 5

2.1 Schematic supporting the main idea of this research work. Uncertainty is represented in all Machine Learning (ML) core components and the decision can be either accepting the ML prediction or abstaining from giving the prediction due to the presence of high uncertainty. 8

2.2 Illustration of the potential sources of uncertainty in a machine learning pipeline for an ECG diagnosis classification problem with three classes (NSR: Normal, AF: Atrial fibrillation, LBBB: Left bundle branch block). The sources of uncertainty are depicted in dashed boxes. Variations in real-world factors, such as the use of different medical devices or patient populations, can impact the testing data since machine learning models are only trained on limited information. The measured data is prone to errors and noise introduced by the measurement systems, while labels may be impacted by inter-rater variability. The assumptions made by the chosen machine learning model and its final approximation also contribute to the overall predictive uncertainty. Examples of testing data with uncertainty include noisy data, shifts in data distribution, and unseen data (out-of-distribution). 10

2.3 Artificial dataset used to illustrate the different sources of uncertainty in a binary classification task. 11

2.4 Illustration of aleatoric and knowledge uncertainty regions (in gray) for the artificial dataset. 12

2.5 Illustration of model uncertainty region (in gray) with the increased sample size for the artificial dataset. The dashed lines represent the maximum and minimum decision boundaries obtained using a bootstrap approach. *N* in the lower right corner indicates the dataset size. 13

3.1 Overview of uncertainty estimation methods. Single methods are categorized into Internal and External approaches, with the latter using an additional component for uncertainty estimation. Bayesian methods are further divided into stochastic and ensemble approximations. 19

3.2 Uncertainty values for different classification algorithms using the maximum probability (upper plots) and predictive entropy (lower plots) as uncertainty measures. The sepal length and petal length features of the Iris dataset are used. The colorbars are set to the minimum and maximum theoretical values of each uncertainty measure. 21

3.3 Overview of the main steps of the proposed knowledge uncertainty measure (Knowledge Uncertainty Estimation (KUE)). 23

3.4 Uncertainty values for Iris dataset using KUE measure and four classification algorithms: Naïve Bayes, Random Forest, K-Nearest Neighbors (KNN) and Support Vector Machine (SVM). High uncertainty is consistently observed in low-density regions. 24

3.5 Uncertainty values for different classification algorithms using the probability variance (upper plots) and variation ratios (lower plots) as uncertainty measures. The sepal length and petal length features of the Iris dataset are used. The colorbars are set to the minimum and maximum theoretical values of each uncertainty measure. 26

3.6 Uncertainty values for different classification algorithms using the decomposition of entropy measures. From top to down total uncertainty, aleatoric uncertainty, and epistemic uncertainty. 28

3.7 Representation of a confidence curve derived from an uncertainty measure depicted by a solid line, the oracle confidence curve depicted by a dashed line, and the Area Under the Confidence-Oracle (AUCO) depicted in grey. . . . 31

3.8 Uncertainty evaluation measures for distinguishing in-distribution and out-of-distribution samples. On the left, the Receiver Operating Characteristic (ROC) curve is depicted by a solid line, the random predictor is depicted by a dashed line, and the Area Under the ROC (AUROC) is shown in grey. On the right, the Precision-Recall (PR) curve is depicted by a solid line, the random predictor is depicted by a dashed line, and the AUPR is shown in grey. . . . 32

3.9 Reliability diagram showing an underconfident model (accuracy is larger than the corresponding confidence), overconfident model (accuracy is smaller than the corresponding confidence), and a perfect calibrated model (accuracy is equal to confidence). 34

3.10 Relation between TPR, FPR, FPR* and train percentiles. FPR* stands for an adjusted FPR, where misclassified inputs were removed from the FPR. . . . 38

3.11 Accuracy-rejection curves for aleatoric, epistemic, and total uncertainty for the Bacteria dataset. The curve for perfect rejection is included as a baseline. The name in each plot represents the antibiotic name used for each Out-of-Distribution (OOD) inputs combination. 39

3.12 ROC curves for OOD detection using our KUE method and baseline methods on Caspofungin and Ciprofloxacin OOD combinations of the Bacteria dataset. Caspofungin and Ciprogloxacin represent the best and the worst performances of KUE method, respectively. 43

3.13 Ablation study of KUE method using Kernel Density Estimation (KDE), for the four datasets. The legend represents each OOD inputs combination, and the title of each plot represents the dataset used. 44

4.1 Uncertainties' rejection fraction and obtained accuracies using a k-fold cross-validation with an increasing number of training samples for 3 different models. The vertical line represents a training size that obtained a similar baseline accuracy for all models. 54

4.2 Uncertainties' rejection fraction and obtained accuracies with the increasing number of training samples for the Human Activity Recognition (HAR) dataset. 56

4.3 Uncertainties' rejection fraction and obtained accuracies using a k-fold cross-validation with an increasing number of training samples for Gaussian Naive Bayes (NB) and KDE Bayes models. 58

4.4 Performance measures for classification with rejection for different training sizes. 59

4.5 Human Activity Recognition (HAR) dataset uncertainty overview. 60

4.6 HAR dataset uncertainty overview by class. 61

4.7 Prediction uncertainty. A confidence value of 0 represents the obtained threshold for rejecting a sample by the uncertainty source. Bars' sizes are normalized between the maximum theoretical confidence/uncertainty. 62

4.8 Schematic of our framework. (upper) An early fusion model learns with features from different modalities. It often results in a complex high-dimensional explanation output that hinders human analysis (lower). Our proposed approach with model aggregation using uncertainty quantification that rejects the most uncertain models. The reduced number of modalities and features yields simpler explanations. 63

4.9 Overview of the proposed machine learning pipeline. 72

4.10 Rejection rate as a function of normalized uncertainty values for (a) aleatoric and (b) epistemic uncertainties. The uncertainty threshold is normalized to the maximum theoretical value of each uncertainty measure. Results are shown for both datasets using the best-performing uncertainty measures when applying majority voting, with p_{max} for aleatoric uncertainty and vr for epistemic uncertainty. 75

4.11 Diagram illustrating the performance of the best-performing model with uncertainty-based aggregation methods when handling missing modalities. The gray scale indicates the obtained performance for each combination of modalities (black for highest, white for lowest). The diagram starts with five modalities and ends with a combination of two modalities. The performance values for each combination are presented alongside the modalities in the figure. 77

4.12 Relationship between the mean explanation complexity and mean performance for the individual modalities, ALL, and aggregation approaches. The horizontal and vertical error bars represent the standard deviation across all folds for explanation complexity and performance, respectively. 78

4.13 Number of modalities and features of the minimum relevant feature subset. We report the median number of features and modalities and the error bar with a 95% confidence interval. 78

4.14 The impact of the threshold in the minimum relevant feature size. We multiplied $\left|p(y|x) - \mathbb{E}[f(x)]\right|$ from Eq. 4.13 by the scalar α and calculated the resulting number of features, modalities, and complexity for a range of values of α. Each point represents the mean value across all folds. (a) WESAD dataset and (b) CSL-SHARE dataset. 79

5.1 Workflow for a clinical decision support system (CDSS) that includes uncertainty quantification techniques. The process begins with a data quality assessment, followed by the machine learning model generating a prediction or diagnosis and estimating the associated uncertainty. The incorporation of Uncertainty Quantification (UQ) allows clinicians to make more informed decisions. Rejected predictions are used to retrain the model using an active learning workflow, improving the overall accuracy and reliability of the CDSS. 90

5.2 Overview of the methodology used for uncertainty methods evaluation. Data is based on 12-lead ECG signals, and uncertainty methods are divided into three main categories: Single, Bayesian, and Ensemble. Validation is done using three test sets (internal, external, and OOD) evaluated in terms of performance, calibration, and uncertainty measures. 92

5.3 Uncertainty methods and corresponding uncertainty measures selected for this analysis. The acronyms used throughout this work are represented in bold. 97

5.4 Binary multi-class multi-label confusion matrices for DeepEnsemble method in internal (IN) and external (EXT) validation sets. 100

5.5 Class performance drop as a function of Wasserstein distance between training and each represented test set. Each point is annotated with the class name abbreviation and the color represents the dataset. The linear regression is obtained with all datasets and represented in gray. The Pearson correlation coefficient (r) and p-value (p) for testing non-correlation are annotated in the graph area. 101

5.6 Correlation between Wasserstein distance and F1-score using different datasets combinations for internal and external datasets. The Pearson correlation coefficient (r) and p-value (p) for testing non-correlation are annotated in the graph area. 102

5.7 Reliability diagrams for internal (IN) and external (EXT) validation sets. The diagonal dashed line represents the perfect calibration. 103

5.8 DeepEnsemble uncertainty measures distributions for internal (CPSC), external (G12EC and PTB-XL), and Out-of-Distribution (OOD) datasets. Uncertainty measures are normalized with their maximum theoretical value, where 1 represents the maximum possible uncertainty. The OOD label contains data from both OOD-MI and OOD-HYP sets. 105

5.9 Comparison of uncertainty value distributions for correctly and incorrectly classified samples using DeepEnsemble model. (a) label dependence approach, where the entire label combination is considered as either correct or incorrect, and (b) label independence approach, where each class is treated as a separate binary classification problem. 106

5.10 Ranking performance evaluation using Area Under the Confidence-Oracle (AUCO) for all datasets and uncertainty methods. 107

5.11 Uncertainty performance measures for varying threshold values across different datasets using four uncertainty estimation methods with epistemic uncertainty. The chosen threshold for each method is denoted by a data point superimposed on each line plot. 109

5.12 Uncertainty performance measures for varying threshold values across different datasets using four uncertainty estimation methods with aleatoric uncertainty. The chosen threshold for each method is denoted by a data point superimposed on each line plot. 110

5.13 Example of reconstruction ECG signals using the autoencoder. On the left side, the ECG signal was classified as acceptable, and on the left side, the signal was classified as unacceptable. 113

5.14 Aleatoric and epistemic uncertainty values distribution for unacceptable and acceptable ECG signals and for internal and external datasets. 114

5.15 ECG lead II examples of signals classified as acceptable for both approaches. From the top row to the down row, the ECG signals are from the CPSC, G12EC, and PTB-XL datasets. 115

5.16 ECG lead II examples of signals classified as unacceptable for at least one of the tested approaches. From the top row to the down row, the ECG signals are from the CPSC, G12EC, and PTB-XL datasets. ECG signals with a red square on the upper right corner represent the signals not rejected using the classification with rejection option. The filename on the upper corner is displayed for reproducibility. 115

5.17 Classification performance as a function of the number of samples used to retrain the DeepEnsemble. In the upper plots data from the G12EC dataset was used to retrain the model, and in the lower plots, PTB-XL data was used. The gray background represents the dataset used to retrain the models. . . 117

6.1 Overview of the main contributions of this research project. 124

A.1 Accuracy-rejection curves for aleatoric, epistemic, and total uncertainty for HAR dataset. The curve for perfect rejection is included as a baseline. The name in each plot represents the activity used for each OOD input combination. 152

A.2 Accuracy-rejection curves for aleatoric, epistemic, and total uncertainty for the Cardio dataset. The curve for perfect rejection is included as a baseline. The name in each plot represents the OOD input combination. 153

A.3 Accuracy-rejection curves for aleatoric, epistemic, and total uncertainty for the Digits dataset. The curve for perfect rejection is included as a baseline. The name in each plot represents the digits used for each OOD input combination. 153

C.1 F1-Score for three subsets of the PTB-XL dataset. $NORM \cup SR$ represents the full dataset, $NORM$ is the subset with Normal Class and $\overline{NORM} \cap SR$ is the subset with only SR annotations. 160

List of Tables

3.1 Datasets used and their characteristics. 36

3.2 AUROC for detecting OOD test inputs using two variants of KUE (KDE and Gaussian) and other baseline methods on 4 datasets. The Mean and Standard Deviation (SD) over OOD combinations are presented after each dataset. All values are averages over 10 consecutive repetitions. 42

3.3 Qualitative AUROC evaluation over all OOD combinations. *Excellent*: 90–100%, *Good*: 80–90%, *Fair*: 70–80%, *Poor*: 60–70%, *Fail*: 50–60%, ↓ *Random*: < 50% 43

4.1 Performance measures (mean ± standard deviation) for different models using a training size of 7692 samples. The highlighted baseline accuracies represent the selected models that were considered for further analysis, since the models attained similar accuracy values. 53

4.2 Performance measures for different models using a training size of 7352 samples and the Human Activity Recognition (HAR) dataset. 56

4.3 Performance measures for individual models (Gaussian naive Bayes and KDE Bayes) and a combination of both models. The results were obtained using a rejection fraction of 10% and a training size of 90,000 samples. 59

4.4 Summary of combination strategies weighted by uncertainty measures. AU: Aleatoric Uncertainty, EU: Epistemic Uncertainty, TU: Total Uncertainty, MR: Model Reliability. $*$ combination of other aleatoric and epistemic uncertainty measures. 70

4.5 Performance evaluation of the best models for each modality comparing the results from Schmidt et al. [138] and ours. F1-score with macro average is used as the performance measure. We report the mean and standard deviation of LOSO. The standard deviation of the LOSO was not reported by Schmidt et al. [138]. The best results are highlighted in bold. 73

4.6 Performance evaluation of the best models for each modality. Accuracy is used as the performance measure. We report the mean and standard deviation of LOSO. 73

4.7 Performance evaluation of baseline aggregation methods compared with the proposed weighted version using uncertainty measures. The best result per dataset and aggregation strategy is highlighted in bold. 74

5.1 A summary of related studies on ECG classification using uncertainty quantification measures. 88

5.2 Overview of multi-label datasets statistics. 94

5.3 Global performance of uncertainty methods in internal (IN) and external (EXT) validation sets. The highest scores are represented in bold. 100

5.4 Comparison of metrics over the three test sets based on the embeddings extracted from the neural network. For each setting, values were averaged over every test set. 101

5.5 Performance comparison of different combinations of internal (IN) and external (EXT) validation sets. 102

5.6 Expected Calibration Error (ECE) for internal (IN) and external (EXT) validation sets. The lowest errors are represented in bold. 104

5.7 OOD detection performance comparison using all uncertainty methods and measures. OOD datasets are composed of two superclasses sets (MI and HYP) with only unknown classes from the PTB-XL dataset. 108

5.8 Performance evaluation metrics for classification with rejection option using DeepEnsemble method. F1-score is presented without rejection (baseline) and with rejection (Non-rejected F1-score). The rejection threshold was set to 15% rejection on the training set. Predictive uncertainty evaluation measures used the same threshold. 111

5.9 Percentage of samples and classification performance of unacceptable and acceptable subsets using the feature-based approach. 112

5.10 Percentage of samples and classification performance of unacceptable and acceptable subsets using the non-feature-based approach (autoencoder). . 113

5.11 Statistical analysis to compare the unacceptable and acceptable subsets distributions for feature-based and non-feature-based approaches. The values represent the obtained p-values using the non-parametric Wilcoxon-Mann-Whitney test. 114

A.1 Bacterial classes used as OOD for each antibiotic. 147

A.1 *Cont.* 148

A.2 AUROC for detecting OOD inputs using KUE method with KDE applied to 4 different classifiers and the correspondent accuracy (ACC) in % of the classifiers on 4 datasets. All values are averages over 10 consecutive repetitions. 148

A.2 *Cont.* 149

A.3 AUPR-Out for detecting OOD test inputs using two variants of KUE (KDE and Gaussian) and other baseline methods on 4 datasets. The Mean and Standard Deviation (SD) over OOD combinations is presented after each dataset. All values are averages over 10 consecutive repetitions. 150
A.4 AUPR-In for detecting OOD test inputs using two variants of KUE (KDE and Gaussian) and other baseline methods on 4 datasets. The Mean and Standard Deviation (SD) over OOD combinations is presented after each dataset. All values are averages over 10 consecutive repetitions. 151

B.1 Pre-processing and features extracted for each modality of CSL-SHARE dataset. 155
B.2 Pre-processing and features extracted for each modality of WESAD dataset. 156
B.3 Selected models for each modality with corresponding hyperparameters and the number of selected features for WESAD dataset. Hyperparameters not referenced were set to the default values of scikit-learn implementation. . . 157
B.4 Selected models for each modality with corresponding hyperparameters and the number of selected features for CSL-SHARE dataset. Hyperparameters not referenced were set to the default values of scikit-learn implementation. . . 158

C.1 Statistical comparison of average uncertainty values for correctly classified and wrongly classified samples using the non-parametric Mann-Whitney U test. P-values, P-values adjusted with Benjamini-Hochberg procedure, and the absolute value of Cohen's d effect sizes (Δ) are shown for each comparison. **Internal CPSC dataset**. 161
C.2 Statistical comparison of average uncertainty values for correctly classified and wrongly classified samples using the non-parametric Mann-Whitney U test. P-values, P-values adjusted with Benjamini-Hochberg procedure, and the absolute value of Cohen's d effect sizes (Δ) are shown for each comparison. **External G12EC dataset**. 161
C.3 Statistical comparison of average uncertainty values for correctly classified and wrongly classified samples using the non-parametric Mann-Whitney U test. P-values, P-values adjusted with Benjamini-Hochberg procedure, and the absolute value of Cohen's d effect sizes (Δ) are shown for each comparison. **External PTB-XL dataset**. 161

Acronyms

AI	Artificial Intelligence
AUCO	Area Under the Confidence-Oracle
AUPR	Area Under the Precision-Recall
AUROC	Area Under the ROC
BNN	Bayesian Neural Network
CNN	Convolutional Neural Network
DL	Deep Learning
ECE	Expected Calibration Error
ECG	Electrocardiogram
FPR	False Positives Rate
HAR	Human Activity Recognition
KDE	Kernel Density Estimation
KNN	K-Nearest Neighbors
KUE	Knowledge Uncertainty Estimation
ML	Machine Learning
MMD	Maximum Mean Discrepancy
NB	Naive Bayes
OOD	Out-of-Distribution

OSR	Open Set Recognition
PR	Precision-Recall
ROC	Receiver Operating Characteristic
SHAP	SHapley Additive exPlanations
SVM	Support Vector Machine
TPR	True Positives Rate
TSFEL	Time Series Feature Extraction Library
TSSEARCH	Time Series Subsequence Search Library
UCI	University of California Irvine
UQ	Uncertainty Quantification

1

Introduction

1.1 Motivation

Machine learning algorithms hold the potential to revolutionize the way humans approach critical decision-making, particularly in the field of medicine. As we stand at the cusp of a new era, harnessing the power of these algorithms has the potential to unlock unprecedented advancements in diagnostics, prognosis, and patient care. However, the very essence of learning from data is intertwined with the concept of uncertainty, and thus, it is crucial to address this inherent aspect of machine learning to build robust, reliable, and trustworthy models.

The ability to quantify uncertainty in machine learning predictions is paramount, as it enables models to abstain from providing decisions when faced with high levels of uncertainty. In doing so, we can effectively integrate human expertise, aligning with the instinctual practice of physicians seeking second opinions or additional input to reduce uncertainty and make more accurate decisions in unusual clinical cases [1].

In recent years, numerous studies have demonstrated that diagnostic, prognostic, and predictive models developed using machine learning can achieve comparable accuracy to gold-standard methods [2–4]. Consequently, there is growing interest in harnessing Artificial Intelligence (AI) systems to enhance clinical practice by improving workflow efficiency and providing physicians with decision-support tools. As machine learning continues to make significant advancements in these domains, particularly in the field of medicine, it becomes increasingly important to ensure the robustness and trustworthiness of AI models by addressing the inherent uncertainty present in the machine learning process [5].

In addition to its importance in cost-sensitive decision-making domains, UQ is also a key concept within the machine learning methodology itself. By quantifying uncertainty, practitioners can identify and understand the flaws in their models, which can help in the development of new or improved models. UQ is also critical for building more interpretable models.

Therefore, the development of a systematic and formal discipline for UQ in AI-based approaches is essential not only for decision support in safety-critical domains but also for practitioners' decision-making in general. Embracing uncertainty in machine learning can unlock the full potential of AI in biomedical applications, leading to safer and more reliable solutions that truly transform the field of medicine.

1.2 Applications of uncertainty

One of the primary applications of uncertainty in AI is its role in the field of AI safety [6, 7]. This area of research concentrates on developing technical solutions that ensure AI systems function safely and reliably. To achieve reliability, a machine learning system must operate effectively under a diverse range of conditions. By incorporating the ability to quantify uncertainty in its predictions, the system can reduce the likelihood of failure when encountering situations it is ill-equipped to handle. As machine learning continues to advance, AI systems are increasingly deployed in real-world scenarios where safety requirements must be taken into account. For instance, in medical applications such as automated decision-making or recommendation systems, the potential for life-threatening consequences increases if AI models are not reliable [1, 7].

In healthcare applications, AI models should be able to report their own uncertainty in predicting a given sample, so that healthcare workers know when the model is (or is not) confident in its decision. In addition to relying on models' uncertainty in their predictions, the ability to abstain from predicting a sample is also an important application of UQ in AI. This can be useful in situations where the model is faced with unknown classes[1] or adversarial examples[2], as it is likely that the model will make unreasonable decisions (essentially guessing at random) that could introduce biases and affect the judgment of experts. It is important to note that our knowledge will be always incomplete and it is likely that unknown data is submitted to the model after it has been deployed. In such cases, it is important to have safety mechanisms in place that can handle uncertainty, which is where UQ becomes particularly valuable.

In this context, after a machine learning model has been deployed in the real-world setting, UQ has also an important role in detecting dataset shifts [8]. It is important to understand if any dataset shift has occurred, as the conditions in which the model is used may differ from the conditions in which it was created. For example, a dataset shift may occur when a model that was trained on data from one hospital is validated in data from a different hospital due to differences in the patient population, the devices used, or the time frame of the data collection [9]. Understanding and addressing dataset shift is therefore an important aspect of applying machine learning models in the medical field.

In addition to the direct applications of AI safety, it is important to consider the need for continuous training after deploying an AI model [6]. This is due to the continuously

[1] Unknown classes refers to classes that were not previously seen in training data.

[2] Adversarial examples are inputs to machine learning models that an attacker has intentionally designed to cause the model to make a mistake.

changing environment in which AI models operate, where concept drifts and the appearance of unknown medical conditions may occur. For model retraining, it is necessary to label data that requires expert knowledge. Obtaining large amounts of labeled data can be unfeasible during clinical practice. One approach to reducing this effort is to use active learning to select the most informative unlabeled data for the model and ask an expert annotator to label only these selected samples. In this scenario, the ability to separately quantify uncertainty can be a useful criterion for selection within the active learning concept [10, 11].

1.3 Research paths and contributions

The field of uncertainty quantification is vast, and our primary research path is to contribute to the development of more trustworthy and robust models for use in the medical domain. Through exploring the use of uncertainty quantification in this context, our research project's primary goal is to advance the field of uncertainty quantification and promote the safe deployment of machine learning models in various applications.

Our research project contributes to three distinct research paths, which, although connected, can be divided into three main areas: 1) *Uncertainty quantification*; 2) *Uncertainty for model design*; and 3) *Uncertainty for clinical decision-making*.

1. *Uncertainty quantification*: There is currently no standard approach for uncertainty estimation, nor is there a commonly agreed-upon taxonomy for different types of uncertainty. The main goal of our research on this topic is to explore the most notable state-of-the-art methods for uncertainty quantification and identify gaps in their ability to estimate and separate different sources of uncertainty.

   Contributions. Introduced a knowledge uncertainty estimation measure; Extensive evaluation and adaptation of uncertainty quantification methods tailored to multi-label classification setting.

2. *Uncertainty for model design*: Previous research in the field of uncertainty quantification has largely focused on the development of methods for characterizing and quantifying uncertainty to build reliable and robust AI models. However, less attention has been given to leveraging uncertainty to improve model performance and interpretability. This research path aims to explore the use of uncertainty estimation methods during the machine learning development process to support practitioners in making more informed decisions and developing models more transparent and reliable. For instance, this includes using uncertainty to select the most suitable model for a given classification task by assessing the quality of model fit and evaluating the need for additional training samples, as well as combining models using uncertainty estimations to build more robust and interpretable models.

Contributions. Comprehensive study of uncertainty-based rejection for enhancing the machine learning development process and interpretability; Proposed a novel approach to lower the complexity of feature-based explanations through an uncertainty-weighted model combination approach.

3. *Uncertainty for clinical decision making*: Rather than focusing solely on developing better models or improving their accuracy in a given classification task, this research path seeks to identify the practical usefulness of uncertainty estimation methods throughout the entire lifecycle of a deployed machine learning model. This includes exploring how uncertainty can be used to address a range of topics, such as classification with a rejection option, dataset shift, active learning, data quality assessment, model calibration, and adaptability to other classification settings, like the multi-label setting. By investigating the practical applications of uncertainty estimation methods in these areas, we aim to develop a better understanding of how to deploy machine learning models in a responsible and trustworthy manner.

   Contributions. Conducted an in-depth study of uncertainty estimation techniques, exploring their application from the initial stage of the machine learning pipeline to their integration in clinical decision-making.

1.4 Thesis structure

This thesis is divided into six chapters, as summarized in Figure 1.1. Elements of our research have been published in journal articles and have been adapted into chapters, sections, or combined with other texts in this manuscript. In particular, the following works are noteworthy, and their connections to specific chapters are represented in Figure 1.1:

1. C. Pires[†], M. Barandas[†], L. Fernandes, D. Folgado, Hugo Gamboa. "Towards Knowledge Uncertainty Estimation for Open Set Recognition." *Machine Learning and Knowledge Extraction* 2(4): 505-532, 2020.

2. M. Barandas, D. Folgado, R. Santos, R. Simão, H. Gamboa. "Uncertainty-Based Rejection in Machine Learning: Implications for Model Development and Interpretability." *Electronics* 11(3): 396, 2022.

3. D. Folgado[†], M. Barandas[†], L. Famiglini, R. Santos, F. Cabitza, H. Gamboa. "Uncertainty Quantification Meets Explainability: Insights from Model Combination on Multimodal Time Series." *Information Fusion* 100: 101955, 2023.

[†] These authors contributed equally to this work.

4. M. Barandas, L. Famiglini, A. Campagner, D. Folgado, R. Simao, F. Cabitza, H. Gamboa. "Evaluation of uncertainty quantification methods in multi-label classification: a case study with automatic diagnosis of electrocardiogram." *Information Fusion* 101: 101978, 2024.

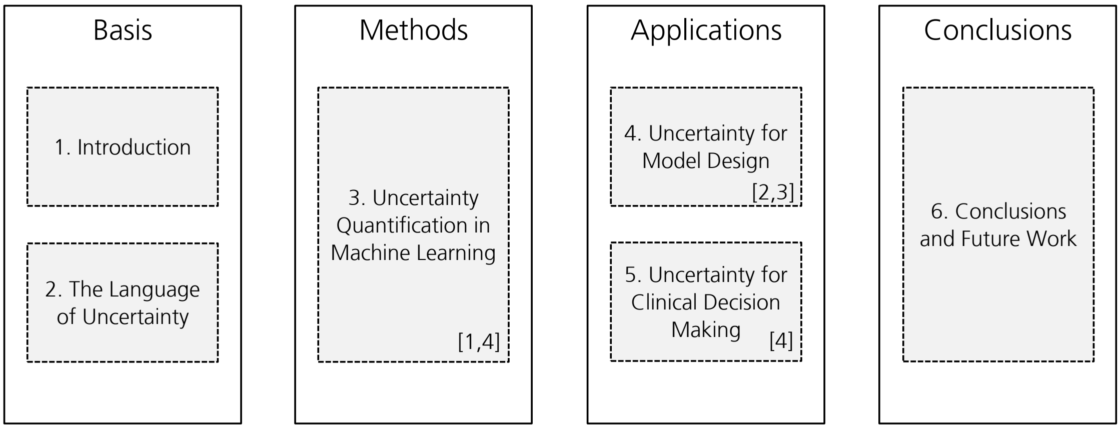

Basis
1. Introduction
2. The Language of Uncertainty
Methods
3. Uncertainty Quantification in Machine Learning
[1,4]
Applications
4. Uncertainty for Model Design
[2,3]
5. Uncertainty for Clinical Decision Making
[4]
Conclusions
6. Conclusions and Future Work

Figure 1.1: Overview of the thesis structure, illustrating the primary research works that were adapted into sections or combined with other texts within the respective chapters.

The present chapter (**Chapter 1**) introduces the problem, providing insights into the importance of uncertainty quantification in machine learning. This concept is particularly useful in decision-critical domains and within the machine learning methodology. The main research paths of this thesis are also presented.

Chapter 2 focuses on the theoretical formalization of our problem as a supervised classification problem with a rejection option. It discusses the definition of uncertainty, as well as the primary sources of uncertainty, *aleatoric uncertainty* and *epistemic uncertainty* in machine learning. Additionally, various perspectives on representing uncertainty are introduced, depending on the AI area.

Chapter 3 outlines the primary methods for estimating uncertainty, categorizing them into two main groups based on how uncertainty is modeled: *Single methods* and *Bayesian methods*. The chapter discusses specific uncertainty measures derived from each estimation method, as well as our proposed measure for estimating *knowledge uncertainty* [12]. Additionally, we introduce uncertainty evaluation measures to assess the quality and impact of uncertainty estimates and explain calibration methods, which are crucial for ensuring the reliability of probability-based uncertainty measures. The chapter concludes with an experimental analysis using our proposed measure to estimate knowledge uncertainty, providing a comparison with state-of-the-art methods.

Chapter 4 and **Chapter 5** cover the primary applications of uncertainty quantification addressed in this thesis. Chapter 4 discusses uncertainty applications for model design, aiming to help practitioners develop more robust and reliable models. This chapter emphasizes the importance of uncertainty-based rejection and the use of uncertainty quantification measures for combining different models. Chapter 5 presents an in-depth study of the different applications of uncertainty quantification within the workflow of a clinical decision support system, using an ECG classification problem as a domain example.

Chapter 6 brings the thesis to a close by offering a comprehensive overview of the key conclusions and the significance of our research. Additionally, it highlights the contributions made throughout the work and provides a summary of potential future research directions.

Included in the appendices are further details regarding the analysis of experimental results presented in Chapters 3, 4, and 5.

2 The Language of Uncertainty

To formally address the topic of uncertainty quantification in machine learning, this chapter starts with a definition of uncertainty followed by a brief introduction of recent literature in the field. The representation of uncertainty in machine learning models' predictions is also discussed within different areas of AI. Then, we introduce the two main sources of uncertainty in predictions: *aleatoric uncertainty* and *epistemic uncertainty*, in the context of classification tasks. This chapter concludes with the formulation of the research problem as a supervised machine learning classification setting with a rejection option where uncertainty quantification is modeled along the machine learning pipeline.

To provide a foundation for our discussion in this chapter, we begin by presenting Figure 2.1, which illustrates the main idea of our research. The figure depicts the different sources of uncertainty that can arise at various stages of the machine learning pipeline. Furthermore, machine learning models are designed with abstaining capabilities, which allow them to either accept or reject predictions based on the degree of uncertainty present. We will provide a more detailed explanation of these concepts throughout this chapter.

2.1 Definition of uncertainty

Uncertainty is ubiquitous and occurs in every single event we encounter in the real world. In general terms, uncertainty can be defined as the *lack of sureness about something or someone, ranging from just short of complete sureness to an almost complete lack of conviction about an outcome* [13].

Uncertainty is a concept that has its roots in various fields and has been present since the early days of the scientific method. The idea of measurement uncertainty started to gain prominence when people realized that measurements are always subject to some degree of error or imprecision. The development of methods to deal with measurement uncertainty has been an ongoing process over the centuries, with numerous mathematicians, scientists, and statisticians contributing to this area. For example, Laplace's work

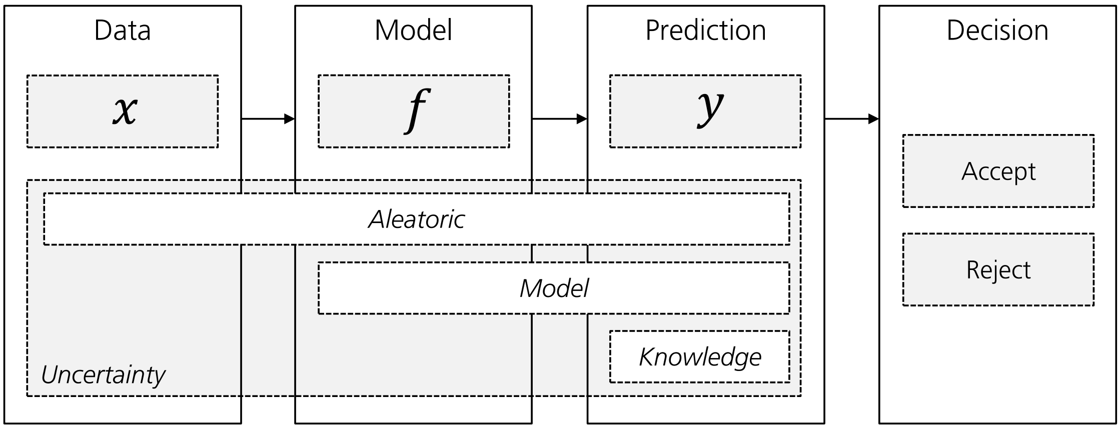


Figure 2.1: Schematic supporting the main idea of this research work. Uncertainty is represented in all ML core components and the decision can be either accepting the ML prediction or abstaining from giving the prediction due to the presence of high uncertainty.

on probability theory and normal distribution helped lay the groundwork for understanding and quantifying measurement errors and uncertainty. The concept of standard deviation, although not initially referred to by that name, can be traced back to the work of mathematicians and statisticians in the 18^{th} and 19^{th} centuries who introduced the idea of quantifying the dispersion or variability of data in numerous research works.

However, the term uncertainty became more prominent in the context of physics with the development of quantum mechanics in the early 20^{th} century. The Heisenberg Uncertainty Principle, formulated by Werner Heisenberg in 1927, is one of the most well-known examples of uncertainty in physics.

Uncertainty also has connections to philosophy, where the term is associated with the limits of human knowledge and the challenges of making decisions or judgments based on incomplete or imperfect information.

Machine learning is an interdisciplinary field that borrows concepts and techniques from various domains and is heavily based on statistical concepts and methods. Uncertainty in statistics often arises from randomness, noise, or sampling variability, which are also present in the data used to train machine learning models. In addition, in the context of machine learning, uncertainty also arises due to the limitations of learning algorithms and the imperfect nature of the models used to represent complex real-world phenomena. Machine learning, as a field that aims to develop algorithms that can learn from data, grapples with issues of uncertainty similar to the philosophical aspects of uncertainty, as models attempt to generalize from limited training data to make predictions in new, unseen situations.

In summary, machine learning models are built to learn from data and make predictions or decisions based on the patterns and relationships they capture from that data. However, the data used for training these models are imperfect and limited, which results in uncertainty in the model's predictions.

Uncertainty is a fundamental concept that permeates various fields of study. While specific definitions and interpretations of uncertainty may vary across these disciplines, the core idea revolves around the notion of incomplete knowledge, variability, or ambiguity in understanding, predicting, or estimating a particular situation, outcome, or event.

2.2 Sources of uncertainty

In order to understand the problem of uncertainty quantification in ML it is first necessary to understand the sources of uncertainty, as different types of uncertainty require different modeling approaches.

The uncertainty is classified in different ways by different communities. However, conceptually there are two fundamental sources of uncertainty categorized into *aleatoric uncertainty* and *epistemic uncertainty*. *Aleatoric uncertainty* refers to the notion of randomness arising from the data complexity, multi-modality, and noise. *Aleatoric uncertainty*, also known as *data uncertainty*, cannot be reduced because it is a property of the underlying distribution that generated the data, rather than a property of the model. On the other hand, *epistemic uncertainty* represents the uncertainty caused by a lack of knowledge of the underlying process being modeled, either due to the uncertainty associated with the model or the lack of data. In principle, this uncertainty can be reduced by providing more knowledge, i.e extending the training data, better modeling, or better data analysis.

Figure 2.2 provides an example of an ECG diagnosis classification problem and illustrates the potential sources of uncertainty in the machine learning model process, from the acquisition of raw information to the final machine learning prediction. Following the example, the process begins with the data acquisition of a set of users, which will serve as the training data for a machine learning model. However, the model is limited to learning patterns from the training set and may be impacted by factors such as the population type, medical devices used, or environmental conditions during acquisition. Nevertheless, since most environments are constantly changing, factors like the introduction of new medical devices or changes in the test population can affect the machine learning model and lead to a significant decrease in performance. This variability of real-world situations can lead to the propagation of (epistemic) uncertainty through the model affecting its final prediction.

The second block in Figure 2.2 pertains to the measurements used for training the models, which constitutes a source of aleatoric uncertainty. The measured data can be imprecise due to noise from the sensors or movement during acquisition. Additionally, limited information for directly mapping data to targets (e.g., ECG features to diagnosis) is also a source of aleatoric uncertainty in the process. For instance, similar pathologies can lead to high aleatoric uncertainty as a result of the overlap between classes.

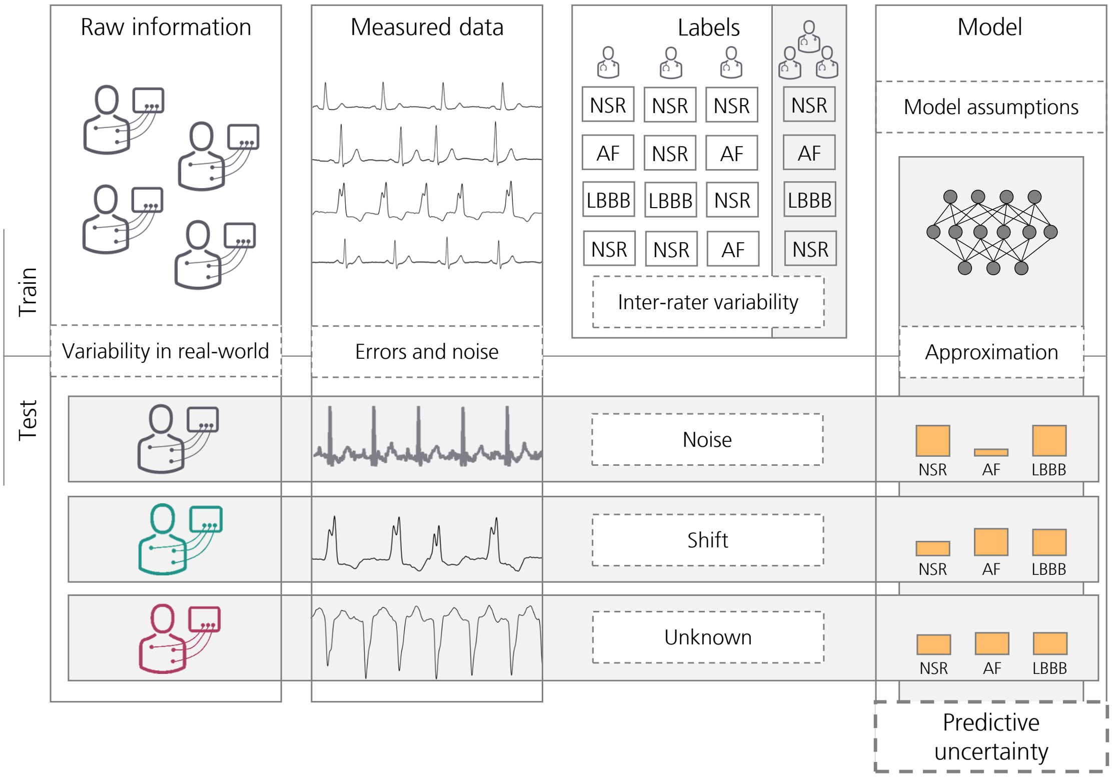


Figure 2.2: Illustration of the potential sources of uncertainty in a machine learning pipeline for an ECG diagnosis classification problem with three classes (NSR: Normal, AF: Atrial fibrillation, LBBB: Left bundle branch block). The sources of uncertainty are depicted in dashed boxes. Variations in real-world factors, such as the use of different medical devices or patient populations, can impact the testing data since machine learning models are only trained on limited information. The measured data is prone to errors and noise introduced by the measurement systems, while labels may be impacted by inter-rater variability. The assumptions made by the chosen machine learning model and its final approximation also contribute to the overall predictive uncertainty. Examples of testing data with uncertainty include noisy data, shifts in data distribution, and unseen data (out-of-distribution).

Furthermore, the labels used for mapping data to targets can also be a source of uncertainty when false labeling occurs. In the medical field, it has been observed that variability among raters can directly impact the model's uncertainty, as the same data may have different ground truth annotations depending on the rater [14]. This source of uncertainty is referenced as label noise or label uncertainty and reduces the model's confidence in the true class prediction during training.

Machine learning extracts models from data through the process of induction, which is inherently linked to uncertainty [5]. Incorrect model assumptions are therefore another source of uncertainty in the machine learning pipeline. We refer to this type of uncertainty as model uncertainty, which is a part of epistemic uncertainty. The inference process results in an approximation that combines the various sources of uncertainty,

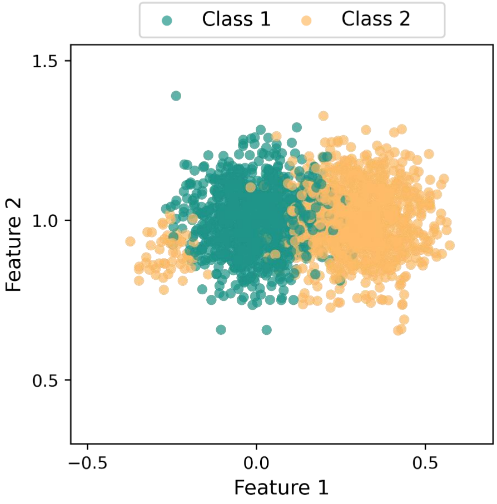


Figure 2.3: Artificial dataset used to illustrate the different sources of uncertainty in a binary classification task.

contributing to the final predictive uncertainty.

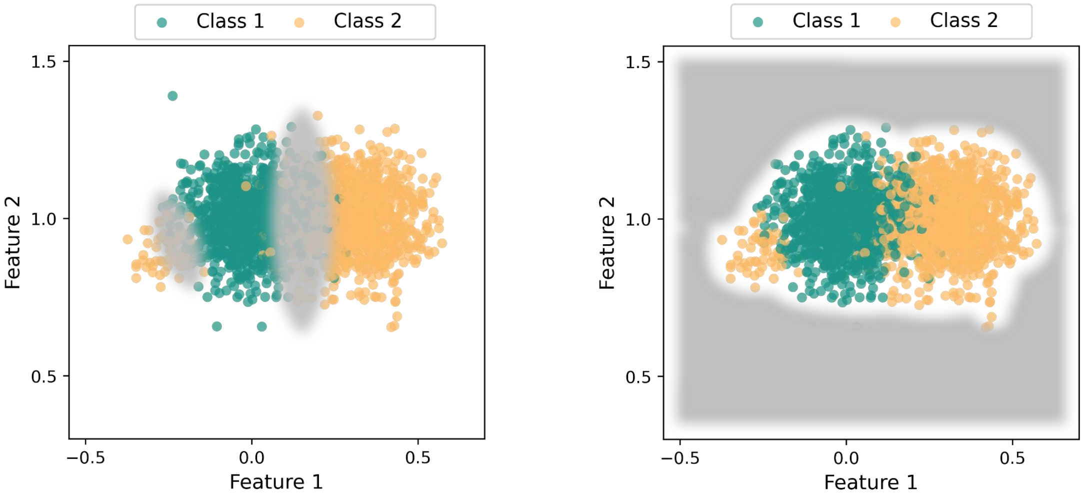
In the testing phase, various sources of uncertainty can arise. We will describe the three test scenarios depicted in Figure 2.2. An uncertainty-aware model designed to detect cardiac pathologies using ECG data should exhibit high uncertainty in all these scenarios. For instance, in the noisy data scenario, the test sample comes from the same distribution as the training data, but the measurement system generates a noisy ECG signal due to factors such as improper electrode placement or movement during the examination. In this case, the model should indicate high aleatoric uncertainty. The second example presents a test sample from a different distribution than the training data, such as a shift in the testing distribution caused by the introduction of a new electrocardiograph, resulting in high epistemic uncertainty. Similarly, high epistemic uncertainty is evident in the third example when the model is tasked with predicting out-of-distribution test data. For instance, an unseen pathology not used during model training would lie outside the data distribution on which the model was trained.

2.2.1 Classification example

To illustrate the concept of aleatoric uncertainty and epistemic uncertainty more concretely, consider an illustrative example using an artificial binary classification dataset, shown in Figure 2.3.

This artificial dataset consists of a two-dimensional dataset with two classes, where features from Class 1 were modeled with an unimodal Gaussian distribution, and features from Class 2 were modeled as a bimodal distribution with a mixture of two Gaussian distributions with highly unequal mass. The minor mode is approximately 5.5% of the probability mass of the major mode.

Observing Figure 2.3, an intuitive conclusion is that no matter the learning algorithm used and its performance capabilities, in the overlap class regions it is not possible to decide between the two classes without having a high uncertainty in that decision. Moreover, the regions in space where there are few or no data should also represent a high uncertainty if a model is sought to predict in these regions. In Figure 2.4 a representation of the mentioned regions is illustrated in gray, where aleatoric uncertainty is represented by the two regions in space where there is an overlap between both classes and epistemic uncertainty by the regions in space where it is little or no evidence of any class regardless of being far/near from the decision boundary. In this example, the epistemic uncertainty is only represented by the lack of data, however, the epistemic uncertainty also occurs due to uncertainties associated with the learning algorithm used to model a given task. In this work, we refer to knowledge uncertainty as the uncertainty related to the lack of data, and we use the term model uncertainty to refer to the uncertainty related to the model itself, i.e. the quality of the model fit on known data or uncertainty about the model parameters. Thus, in Figure 2.4(b), only knowledge uncertainty is represented.



(a) Aleatoric uncertainty

(b) Knowledge uncertainty

Figure 2.4: Illustration of aleatoric and knowledge uncertainty regions (in gray) for the artificial dataset.

The representation of model uncertainty is not trivial and it depends on the learning algorithm used to model the task. For illustration purposes, let us assume that a given model produces a decision boundary for different bootstrap samples[1]. The differences between decision boundaries of the same sample size, i.e. the maximum and minimum decision boundary values, are represented by the dashed lines in Figure 2.5. The gray region between the dashed lines can be seen as a representation of model uncertainty. Note that with the increased number of samples, the uncertainty region representing model uncertainty decreases and tends to near zero. Since decision boundaries are highly dependent on the available data, slight differences in the bootstrap samples have a high

[1] Bootstrap samples are new datasets created by sampling with replacement from the original dataset.

impact on model fit, especially in regions with little evidence. Although, increasing the sample size results in a well-defined decision boundary.

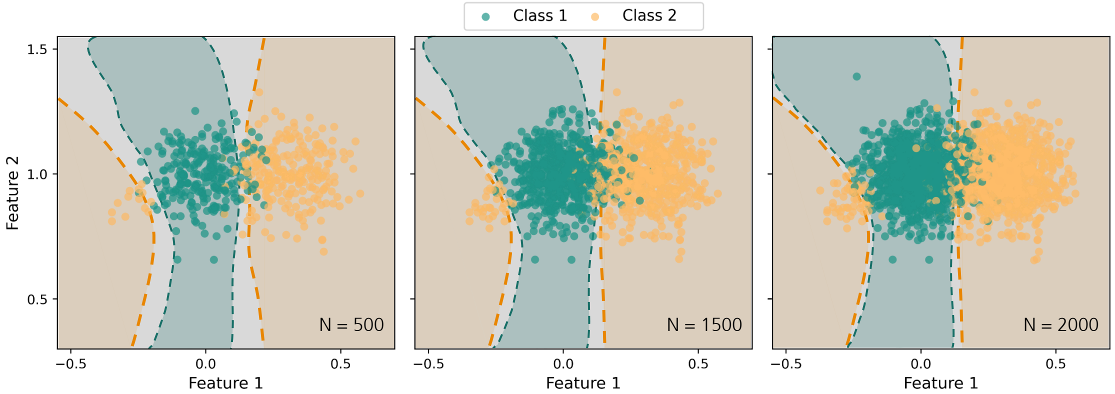


Figure 2.5: Illustration of model uncertainty region (in gray) with the increased sample size for the artificial dataset. The dashed lines represent the maximum and minimum decision boundaries obtained using a bootstrap approach. N in the lower right corner indicates the dataset size.

2.3 Representing uncertainty

In recent years, researchers have shown an increased interest in estimating uncertainty in machine learning. The most common method for estimating the uncertainty of a machine learning prediction, known as predictive uncertainty, involves separately modeling the uncertainty caused by data, referred to as aleatoric uncertainty, and the uncertainty caused by the model or knowledge, referred to as epistemic uncertainty. The methods for separating these uncertainties will be discussed in Chapter 3.

Although different types of uncertainty should be measured differently, this distinction in ML has only received attention recently [5]. In particular, in the medical domain, Senge et al. [15] proposed a method for quantifying the aleatoric and epistemic uncertainty showing the usefulness and reasonableness of their approach. Also, in the literature on deep learning, this distinction has been studied due to the limited awareness of neural networks of their own confidence. The focus has been more on epistemic uncertainty quantification since deep learning models are known as being overconfident with out-of-distribution examples or even adversarial examples [16, 17]. Motivated by such scenarios, several works have been developed for uncertainty quantification showing the usefulness of distinguishing both types of uncertainty in the context of AI safety [15, 18–20]. Likewise, the ability to separately quantify uncertainty has been used in active learning as a selection criterion for uncertainty sampling, where the selection of samples with high epistemic uncertainty will provide more knowledge to the model, which should improve the quality of the model and its capacity to generalize. Contrarily, the selection of samples with high aleatoric uncertainty is pointless, as aleatoric uncertainty is inherently irreducible. In this context, Nguyen et al. [10] compared the performance of both sources of uncertainty using binary classification datasets and their results showed that epistemic uncertainty outperforms aleatoric uncertainty as a selection criterion for active learning. Gal et al. [21] and Sadafi et al. [11] obtained similar conclusions when using uncertainty measures for active learning purposes.

To bridge the gap between the discussions on different types of uncertainty and their applications in various fields of AI, we highlight the different representations of uncertainty in a prediction between the different areas of AI. While each field is concerned with a specific uncertainty source, none offers a complete view of all uncertainties present in a typical ML problem. Structuring the methods in one common view is not possible and there are various possibilities for providing a coherent division. In this section, we adopted the broad division provided by Shafaei et al. [22] to briefly introduce these closely related domains.

2.3.1 Uncertainty view

Currently, in dealing with uncertainty, probability theory is at a dominant position [13]. Probability theory provides a consistent framework for the quantification and manipulation of uncertainty and forms one of the central foundations for pattern recognition [23]. Based on probability theory, methods such as Monte Carlo, Bayesian, and Dempster-Shafer evidence theory were developed.

Probability theory can deal with different sources of uncertainty in different ways. While aleatoric uncertainty is measured in terms of the variability in the outcome of an experiment due to inherently random effects, epistemic uncertainty is commonly measured in terms of disagreement between potentially viable hypotheses.

Classical statistics and probability estimation are well-established in machine learning to represent uncertainty in a prediction. Methods trained for inducing one single hypothesis or following basic frequentist techniques are mostly concerned with the aleatoric part of the overall uncertainty. Naive Bayes or nearest neighbor classifiers are examples of such methods. In these kinds of methods, it is important to apply calibration techniques to turn their scores into well-calibrated probabilities [5]. Otherwise, Logistic regression, Bayesian networks, or Gaussian processes are examples of well-known methods that output probability estimates from a hypothesis space instead of point predictors. Therefore, these methods provide information about both aleatoric and epistemic uncertainty [5]. A more detailed explanation of different methods under this uncertainty view will be described in Chapter 3.

2.3.2 Anomaly view

An intuitive idea to measure knowledge uncertainty is to use methods that focus on anomaly, outlier, and/or novelty detection. Anomaly or outlier detection is the process of identifying samples that deviates from the training dataset.

Density estimation is commonly used by these approaches to reject test inputs located in low-density regions. These low-density regions, where no training inputs have been encountered so far, represent a high knowledge uncertainty. Traditional methods, such as KDE, can be used to estimate densities, and often threshold-based methods are applied on top of the density where a classifier can abstain from predicting a test input in that region [24]. Distance-based methods are also well established in the anomaly and outlier detection field. These methods rely on a distance measure within the train data to define a threshold for what is considered an anomaly or outlier. Examples of popular approaches within anomaly detection include Local Outlier Factor, Isolation Forest, and one-class SVM algorithms.

The research field of anomaly and outlier detection is large and numerous methods are continuously being proposed in literature [25]. Although important, we will not describe such methods in detail, but highlight this research field due to its tight connection with knowledge uncertainty. In particular, in the area of deep learning, the problem of outlier detection is usually referred to as OOD detection. This topic is particularly important in deep neural networks and has been recognized by several studies showing that deep neural networks usually predict OOD inputs with high confidence [26, 27].

2.3.3 Novelty view

Novelty detection tries to identify novel or unusual data from a dataset. This particular task is often referred to as the Open Set Recognition (OSR) scenario. The OSR approach is similar to OOD detection and can be viewed as tackling both the classification and novelty detection problem at the same time. Contrary to the OOD detection, the novel classes that are not observed during training are often made up of the remaining classes in the same dataset. This task is probably harder because the statistics of a class are often very similar to the statistics of other classes in the dataset [28]. In each case, the goal is to correctly classify inputs that belong to the same distribution as the training set and to reject inputs that are outside this distribution.

A number of approaches have been proposed in the literature for OSR problem [29, 30]. A popular approach is the one-class classification since it focuses on the known class and ignores any additional class. Binary classification with the one-vs-all approach can also be applied to the open-set recognition [31]. In this scenario, when there is no in-distribution classification from the binary classifiers, the test sample is classified as unknown. Different adaptions of one-class classification and variations of the SVM have been applied aiming at minimizing the risk of the unknown classification [32–34].

Distance-based approaches are considered more suitable to open-world scenarios since the addition of new classes to existing classes can be made at near-zero cost [35]. Distance-based classifiers with a rejection option are easily applied to OSR because the classifiers can create a bounded known space in the feature space, rejecting test inputs that are far away from training data. For instance, the Nearest Class Mean classifier is a distance-based classifier that represents classes by the mean feature vector of their elements [35]. The problem for most of the methods dealing with rejection by thresholding the similarity score is the difficulty of determining such a threshold that defines whether a test input is an outlier or not. In this context, Júnior et al. [36] extended the traditional close-set nearest neighbor classifier by applying a threshold on the ratio of similarity scores of the two most similar classes and called it Open Set Nearest Neighbors.

2.3.4 Abstention view

The process of abstaining from producing an answer or discarding a prediction when the system is not confident enough is more than 60 years old and was introduced by Chow [37]. The purpose of abstention is to incur a lower cost than the cost of misclassification and it is primarily associated with the uncertainty view where a change of the learning problem gives models the choice of abstaining from prediction at the cost of a penalty. Chow's theory suggests that objects are rejected for which the maximum posterior probability is below a threshold. If the classifier is not sufficiently accurate for the task at hand, then one can take the approach not to classify all examples, but only those whose posterior probability is sufficiently high. Chow's theory is suitable when a sufficiently large training sample is available for all classes and when the training sample is not contaminated by outliers [38]. Fumera et al. [39] showed that Chow's rule does not perform well if a significant error in the probability estimation is present. In that case, a different rejection threshold per class has to be used. In classifiers with a rejection option, the key parameters are the thresholds that define the rejection area, which may be hard to define and may vary significantly in value, especially when classes have a large spread.

Using these methods, the rejection is mostly applied to samples with high aleatoric uncertainty, since it has been argued that probability distributions are less suitable for representing ignorance in the sense of a lack of knowledge [5]. Alternatively, more recent works [40–42], included the classification with rejection with a distinction between aleatoric and epistemic uncertainty using ensemble techniques and/or deep learning approaches. For the classification with rejection, a confidence threshold value needs to be defined indicating the rejection point. Different cost-based rejection methods have been proposed to minimize the classification risk [43, 44]. In probabilistic classifiers, risk can derive from the observation of the output probabilities employing different metrics, such as the least confidence, margin of confidence, variation ratios, and predictive entropy [6]. Thus, although the abstention view is directly applied to the uncertainty view, both anomaly and novelty views can be easily applied under the concepts of abstaining where a rejection function is learned to reject unknown inputs.

2.4 Final remarks

This chapter discussed the main concepts of uncertainty quantification in machine learning and their intrinsic relationship with classification with rejection option. We formulate the problem of this research as a supervised machine learning classification setting with a rejection option where uncertainty quantification is modeled along the ML pipeline. Besides the typical division between aleatoric and epistemic uncertainty, we further divided the epistemic uncertainty into two additional categories, namely knowledge uncertainty and model uncertainty. Although these terms are commonly used to refer to the broad view of epistemic uncertainty, we refer to knowledge uncertainty as the uncertainty related to the lack of data, i.e. to the regions in space where there is little or no evidence of any class regardless of being far/near from the decision boundary. On the other hand, we refer to model uncertainty as the uncertainty related to the model itself, i.e. the quality of the model fit on known data or uncertainty about the model parameters.

Figure 2.1 summarizes the main idea of our research work, where uncertainties are present in all components of ML systems under different sources. More specifically, *Data (x)* used to feed ML models are limited in their accuracy and potentially affected by various kinds of quality issues, which limits the models from being applied under optimal conditions [45, 46]. For example, the uncertainty caused due to errors in the measurement might affect the performance of a given classification task. Although the aleatoric uncertainty is supposed to be irreducible for a specific dataset, incorporating additional features or improving the quality of the existing features can assist in its reduction [47]; For a given classification task, several machine learning *Models (f)* can be applied and developed. The choice of a model is arguably important and is often based on the degree of error in the model's outcomes. However, besides the models' accuracy, the use of uncertainty quantification methods during model development can provide important elements to choosing the right model for the problem at hand. Moreover, understanding the model's uncertainty during training can give us insights into the specific limitations of each model and help in developing more robust models; After the model's training, estimating and quantifying uncertainty in a transductive way, in the sense of tailoring it to individual instances, is arguably relevant, all the more in safety-critical applications. For instance, in the context of computer-aided diagnosis systems, a *Prediction (y)* with high uncertainty shall justify either disregarding its output or conducting further medical examinations of the patient. In the latter, the goal is to retrieve additional evidence that supports or contradicts a given hypothesis. In the former, it is the case of classification with rejection, where a *Decision* can be either the acceptance of the ML prediction or its rejection, since the presence and cost of errors can be detrimental to the performance of automated classification systems [48].

3 Uncertainty Quantification in Machine Learning

In the previous chapter, we formulate our problem as a supervised machine learning classification task with a rejection option through an uncertainty quantification approach. This chapter will provide an overview of various methods for estimating uncertainty in machine learning, followed by the relevant uncertainty measures for assessing the quality and impact of uncertainty estimates. Throughout this section, we will reference notable works in the field of uncertainty estimation.

For a comprehensive analysis of uncertainty estimation methods, we propose a distinction between two main groups of uncertainty learning methods: *Single Methods* and *Bayesian Methods*. *Single methods*, encompasses uncertainty techniques that rely solely on a single model for the prediction task and can be either deterministic or generative. These methods can be further categorized into internal and external approaches depending if an additional component is used for uncertainty estimation (see Figure 5.3). Under the category of external approaches, we will present our proposed method for estimating knowledge uncertainty. *Bayesian Methods* describes uncertainty techniques that adopt a Bayesian perspective, including various approximation techniques. These methods are further divided into stochastic and ensemble approaches. The former covers all kinds of stochastic models, where each forward pass of the same input generally produces different results, while the latter encompasses ensemble methods that combine the predictions of multiple ensemble members to produce a final prediction and uncertainty estimation. Figure 5.3 summarizes these two primary methods of uncertainty estimation, which are further divided into two categories with illustrations of the underlying reasoning behind each uncertainty technique. It should be noted that while a neural network is used for visualization, the methods described in this chapter are not specific to neural networks or deep learning, and can be applied in a more general context to both traditional machine learning and deep learning.

After uncertainty estimation methods, we will introduce performance metrics to assess the quality of uncertainty estimates, and we will cover calibration techniques that

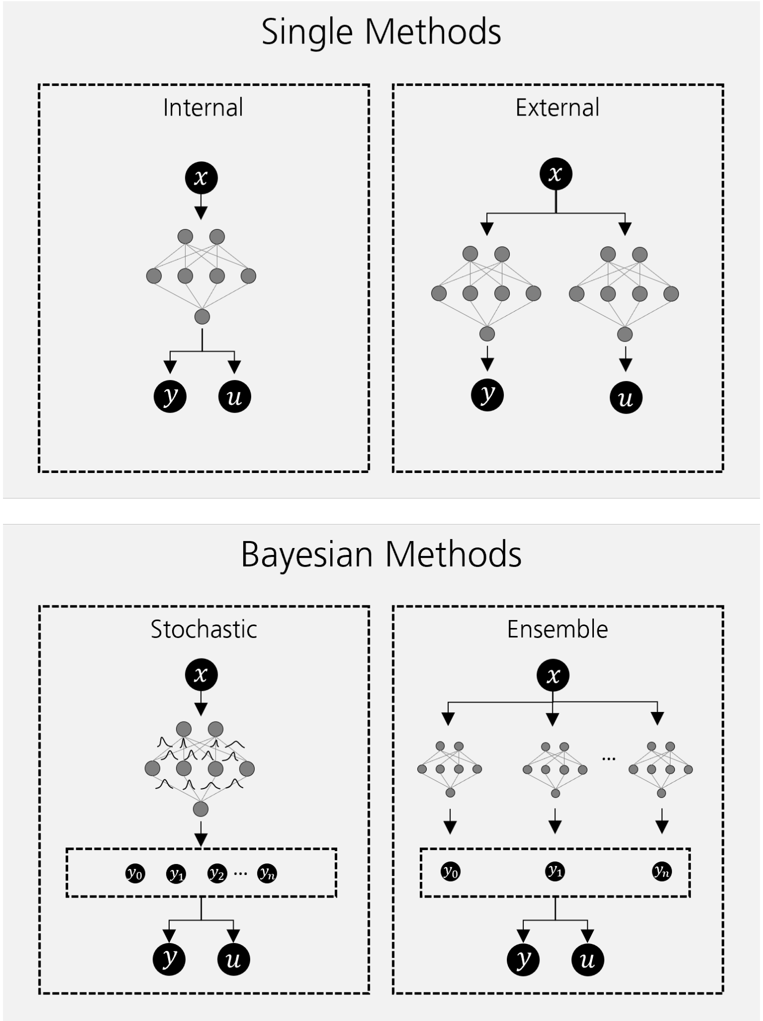


Figure 3.1: Overview of uncertainty estimation methods. Single methods are categorized into Internal and External approaches, with the latter using an additional component for uncertainty estimation. Bayesian methods are further divided into stochastic and ensemble approximations.

are important for ensuring that models are well-calibrated and that uncertainty measures based on probabilities can be used reliably. The chapter will conclude with an experimental analysis comparing our proposed uncertainty estimation method with other state-of-the-art methods for two experimental classification tasks: 1) Classification with rejection option based on a combination of different uncertainty measures; 2) Effectiveness of uncertainty estimation measures in distinguishing in- and out-distribution inputs.

3.0.1 Notation

As standard notation to introduce uncertainty estimation methods throughout this chapter, let us consider a standard setting of supervised learning with a finite training dataset, $D = \{(x_i, y_i)\}_i^N \subset \mathcal{X} \times \mathcal{Y}$, with N samples, composed of pairs of input instances x and outcomes y, where $\mathcal{X}$ is an instance space and $\mathcal{Y}$ the set of outcomes that can be associated with an instance. For the purpose of visualization, we will use the Iris flower dataset which is commonly employed as a benchmark dataset for classification tasks. The dataset consists of four features and three different Iris species. To simplify the visualization process, we will only use two features: the sepal length and the petal length. Regarding the classification learning algorithms[1], we will use a Naïve Bayes using a Gaussian distribution, a Random Forest (maximum depth of 3), a KNN where k was set to 7 neighbors and SVM with the RBF kernel.

3.1 Uncertainty estimation: Single methods

We define single methods as methods that, for the same input, always give the same prediction and are based on a single hypothesis. Following the taxonomy proposed in the context of neural networks by Gawlikowski et al. [49] we also divide these methods into two categories: Internal and External. External methods are based on additional components that evaluate the uncertainty separate from the prediction task, while internal methods depend directly on the hypothesis of the underlying predictor.

3.1.1 Internal approaches

For the internal methods, the most straightforward way of quantifying uncertainty is by using the output of the classification task that represents the class probabilities. Therefore, in a classification task, a simple uncertainty measure given by the confidence in a prediction x can be obtained by the probability of the predicted class, or maximum probability, by the following equation

$$p(\hat{y}|x) = \max_{y \in \mathcal{Y}} p(y|x) \tag{3.1}$$

Additionally, the entropy of the predictive posterior modeled by the (Shannon) entropy, is the most well-known measure of uncertainty of a single probability distribution. For discrete class labels is given by Equation 3.2:

$$H[p(y|x)] = -\sum_{y \in \mathcal{Y}} p(y|x) \log_2 p(y|x) \tag{3.2}$$

Both maximum probability and entropy of the predictive posterior distribution can be seen as measures of the total uncertainty in predictions [19]. These measures of uncertainty for probability distributions primarily capture the shape of the distribution and, hence, are mostly concerned with the aleatoric part of the overall uncertainty and their reliability depends on the probabilistic predictor used. For instance, methods that follow basic frequentist techniques, such as Naïve Bayes or nearest neighbor, return scores

[1] The non-mentioned algorithms' hyperparameters were set to the default values of the python module for machine learning scikit-learn (version 1.2.1).

that must be calibrated to represent well-calibrated probabilities [5]. Neural networks are also known for their overconfident predictions due to poorly calibrated outputs [49].

Figure 3.2 displays the uncertainty values obtained for both maximum probability and predictive entropy using the Iris dataset and four different classification algorithms. Each model provides different uncertainty regions due to different decision boundaries. Regardless of the algorithm used, regions where there is an overlap between classes always produce high uncertainty values. In areas with low density, the behavior varies depending on the algorithm used. However, all algorithms have low uncertainty in regions without data, demonstrating the limitation of using only these uncertainty measures. Comparing maximum probability (upper plots) and predictive entropy (lower plots), the obtained uncertainty values are similar with slight differences in the uncertainty value ranges. Both colorbars are set to the minimum and maximum theoretical values of each uncertainty measure.

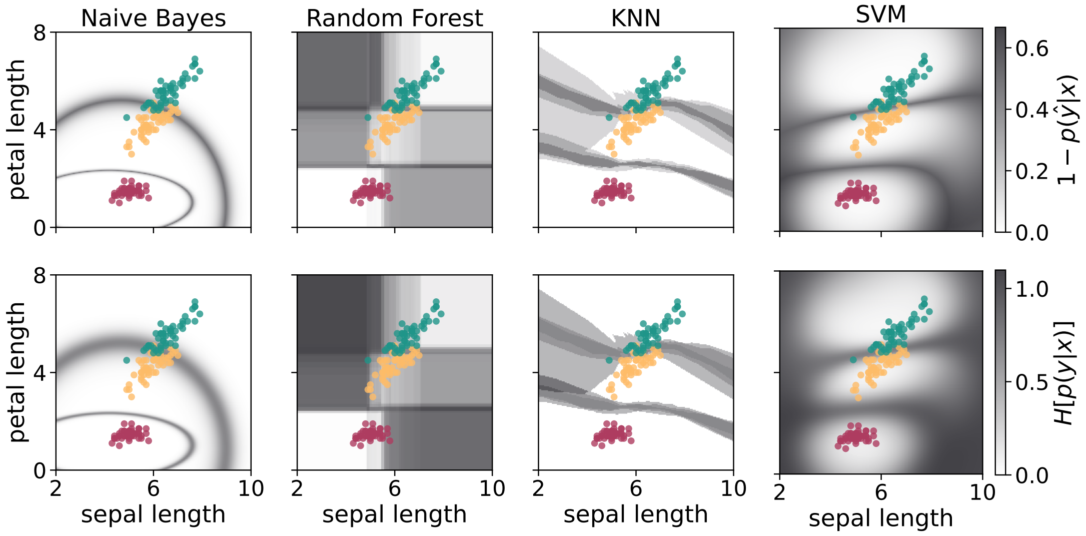


Figure 3.2: Uncertainty values for different classification algorithms using the maximum probability (upper plots) and predictive entropy (lower plots) as uncertainty measures. The sepal length and petal length features of the Iris dataset are used. The colorbars are set to the minimum and maximum theoretical values of each uncertainty measure.

Within the area of deep learning, there are several approaches where neural networks are explicitly modeled and trained to quantify both aleatoric and epistemic uncertainties [19, 20]. In these approaches, along with the uncertainty quantification, the training procedure and network's predictions are affected. More in the realm of anomaly or outlier detection, different approaches tailored for neural networks have been proposed to detect OOD inputs without changing the underlying predictor model and taking advantage of its logits values (the unnormalized predictions of the model), or embeddings representations (low-dimensional representations of discrete data as continuous vectors). We highlight a few representative works in the context of deep learning, such as the Maximum Logit score [50], Mahalanobis distance-based confidence score [51] and energy

[52] or joint energy for multilabel setting [53]. Although these approaches are not developed to explicitly quantify uncertainty, they can be seen as a measure of knowledge uncertainty.

3.1.2 External approaches

External methods are trained directly to quantify uncertainty and therefore are independent of the prediction task. There are some studies that argue that uncertainty quantification and prediction tasks should be two separate tasks for uncertainty quantification to be unbiased [54]. In this context, anomaly, outlier, or novelty methods can be used as a measure of knowledge uncertainty. Generative models that typically rely on densities are an intuitive idea to access knowledge uncertainty. In traditional machine learning, popular approaches are based on the one-class classification and its variations with SVM classifier. Additionally, some more popular approaches in this area include isolation forests [55], auto-encoders [56], and local outlier factor [57].

Alternative approaches to uncertainty quantification have been proposed in the literature, such as conformal prediction [58, 59]. Conformal predictions is a classical frequentist technique centered around hypothesis testing that provides error bounds on a per-instance basis without requiring prior probabilities. For classification problems, conformal predictions transform single-class predictions into set predictions. This transition from point estimation to set estimation inherently involves a sense of uncertainty. For example, a sample with multiple classes prediction indicates that the classifier is uncertain about the correct class, while a prediction with a single class implies confidence in the prediction. In this context, two common metrics for uncertainty quantification applied to conformal predictions are credibility, which measures the likelihood that a sample comes from the training set, and confidence, which estimates the level of certainty the model has that the prediction is a singleton.

3.1.2.1 Knowledge Uncertainty Estimation

During the course of this project, we introduced an agnostic measure named Knowledge Uncertainty Estimation (KUE) [12] in the context of external approaches.[1]

To introduce KUE, let us assume an OSR problem, where a model is trained only over in-distribution data, denoted by the distribution Q_{in}, and tested on a mixture distribution with in- and out-distribution inputs, drawn from Q_{in} and Q_{out} where the latter represents the out-distribution data. Thus, a model is trained to correctly identify the class label from Q_{in} and to reject unknown classes not seen in training from Q_{out}. KUE measure acts in combination with every ML model, measuring the uncertainty associated with each prediction to reject samples with high uncertainty. In Figure 3.3 an overview of the main steps of KUE is presented.

[1] KUE measure reflects a joint work with Catarina Pires as part of her Master's project, and it has already been published in the *Machine Learning and Knowledge Extraction Journal*.

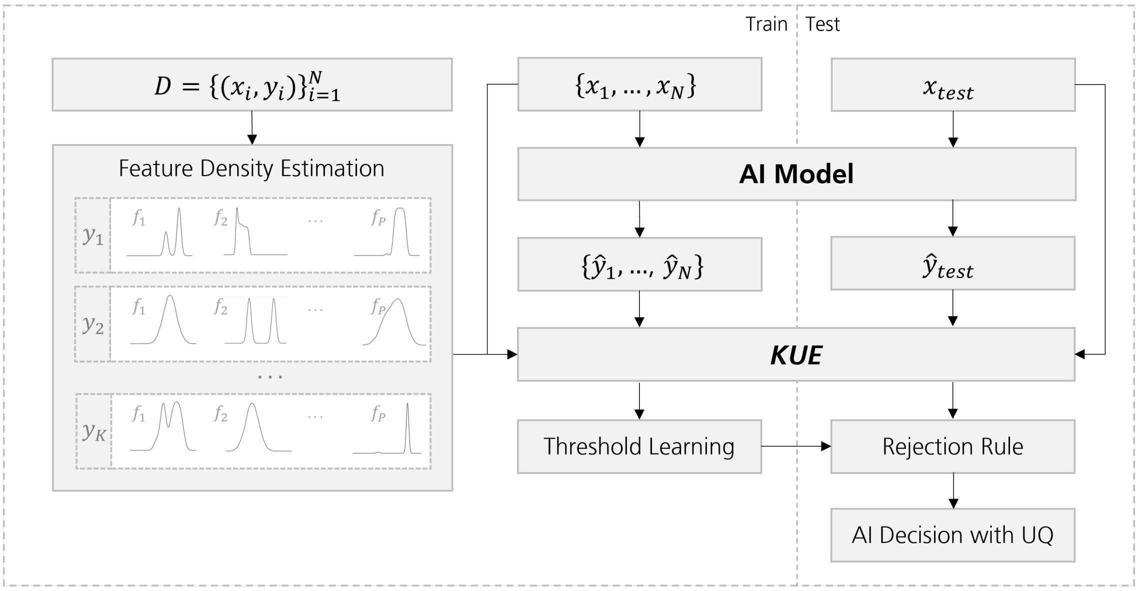


Figure 3.3: Overview of the main steps of the proposed knowledge uncertainty measure (KUE).

KUE is a distance-based measure derived from a normalization of the training features density distributions. During the training phase, the input feature distributions are learned and used for uncertainty estimation during testing. More specifically, uncertainty is modeled through a combination of a normalized density estimation over input feature space for each known class. Assuming an input x_i represented by P-dimensional feature vectors, where $f_j \in \{f_1, \dots, f_P\}$ is the feature vector in a bounded area of the feature space, an independent density estimation of the P features conditional by the class label is estimated and normalized by its maximum density, in order to set all values in the interval $[0, 1]$. Thus, each feature density is transformed on an uncertainty distance, d_{unc}, assuming values in $[0, 1]$, where 1 represents the maximum density seen in training, and near-zero values represent low-density regions where no training inputs were observed during training. The combination between each feature distance is computed by the product rule over whole features. Thus, given a test input x_i from class y_k its associated uncertainty, $KUE(x_i|y_k)$, is calculated by Equation 3.3:

$$KUE(x_i|y_k) = 1 - \left(\prod_{j=1}^{P} d_{unc}(f_j|y_k, x_i) \right)^{\frac{1}{P}} \tag{3.3}$$

Figure 3.4 shows the uncertainty values obtained from the Iris dataset using the KUE measure. While KUE is an external method that does not affect the prediction task, its results depend on the predictive performance of the classification algorithm. The slight variations in the obtained uncertainty values using different learning algorithms are due to differences in the predicted class for the same input. However, high uncertainty values are consistently observed in low-density regions.

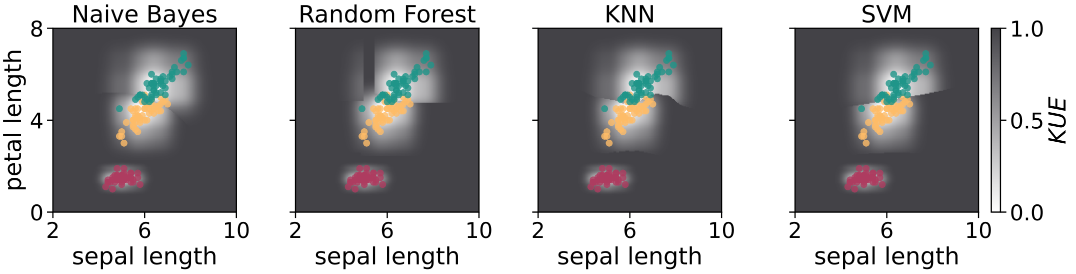


Figure 3.4: Uncertainty values for Iris dataset using KUE measure and four classification algorithms: Naïve Bayes, Random Forest, KNN and SVM. High uncertainty is consistently observed in low-density regions.

The decision to reject a sample is done by an uncertainty threshold. A common approach to defining a threshold for OOD samples is to use a certain amount of OOD data as a validation set. However, this approach is unrealistic due to the proper definition of OOD samples that come from an unknown distribution, leading to a compromised performance in real-world applications, as Shafei et al. [60] showed in their study. Therefore, we argue that a more realistic approach is to learn a threshold only from in-distribution data. Due to the differences between data from different datasets, learning a global threshold for all datasets is not a reliable approach. Therefore, our hypothesis is that if we learn the training uncertainty distribution for each class within a dataset, there is a specific threshold for each distribution that will bound our uncertainty space, so that input samples that fall outside the upper bound threshold are rejected. In our work, the upper bound threshold is defined based on a predefined percentile from the training uncertainty distribution. The percentile choice is defined according to different application scenarios, whether the end-user is willing to reject more or less in-distribution samples. As train and test in-distribution data come from the same distribution it is expected that the percentage of reject samples from test data will represent approximately 10% if the chosen percentile is set to 90%. From this 10% we can also argue that a certain percentage can represent classification errors or, if rejected samples were correctly classified, the classification was done under limited evidence so that a high uncertainty is associated with that decision. Thus, the rejection rule for input sample x_i for in- and out-distribution is given by $g(x_i|y_k)$ in Equation (3.4), where $P_r[U(y_k)]$ represents the uncertainty value for the r-th percentile of the train uncertainty data distribution associated with class y_k. The output values -1 and 1 mean that the input sample x_i is rejected or accepted, respectively:

$$g(x_i|y_k) = \begin{cases} -1 & \text{if } KUE(x_i|y_k) > P_r[KUE(y_k)] \\ 1 & otherwise \end{cases} \tag{3.4}$$

3.2 Uncertainty estimation: Bayesian methods

Bayesian inference can be seen as the main representative of probabilistic methods and provides a coherent framework for statistical reasoning that is well-established in machine learning [5]. The Bayesian interpretation of probability views probability as expressing a degree of belief or information (knowledge) about an event.

For a detailed description, suppose a hypothesis space $\mathscr{H}$ of probabilistic predictors, where a hypothesis h maps instances x to probability distributions on outcomes y, each hypothesis can be considered as an explanation of how the world works. Samples from the posterior distribution should yield explanations consistent with the observations of the world contained within the training data, D [61].

From a Bayesian perspective, each hypothesis is equipped with a prior distribution $p(h)$, and the posterior distribution, $p(h|D)$, can be computed via the Bayes rule,

$$p(h|D) = \frac{p(D|h)p(h)}{p(D)} \tag{3.5}$$

where $p(D|h)$ is the probability of the data given h.

The representation of uncertainty about a prediction is given by the posterior distribution, where the belief about the outcome y is represented by a second-order probability: a probability distribution of probability distributions [42]. In this type of Bayesian inference, a given prediction is obtained through model averaging, i.e., different hypotheses h provide predictions, which are aggregated in terms of a weighted average. The predictive posterior distribution is given by:

$$p(y|x) = \int p(y|x,h)dP(h|D) \tag{3.6}$$

Therefore, the predicted probability of an outcome y is the *expected* probability $p(y|x,h)$, where the expectation over the hypotheses is taken with respect to the posterior distribution, $P(h|D)$. However, since model averaging is often difficult and computationally costly, in machine learning, it is common to use the highest posterior probability to make predictions that consider a single hypothesis [5] or to apply approximation techniques such as the Monte Carlo methods or ensemble methods.

In order to assess the quality of these approximations, uncertainty measures have to be applied to the derived uncertainty estimation methods to quantify the different predicted types of uncertainty.

Assuming a single hypothesis, i.e a single probability distribution, the maximum class probability (Equation 3.1) and Shannon entropy (Equation 3.2) measures can be applied to obtain a measure of aleatoric uncertainty.

However, taking advantage of approximation techniques, where predictive posterior distribution is approximated by a finite set of Monte Carlo samples or by the individual ensemble members' predictions, the predictive variance of the M predictions is a measure of epistemic uncertainty given by:

$$\sigma[p(y|x)]^2 = \frac{1}{M}\sum_{i=1}^{M}(p(y|h_i,x)-\bar{p})^2 \tag{3.7}$$

where $\bar{p}$ is defined as $\bar{p} = \frac{1}{M}\sum_{i=1}^{M} p(y|h_i,x)$

Additionally, instead of considering the probability variance, one can consider the variation ratios that measure the variability of predictions by computing the fraction of samples with the correct output. This heuristic is a measure of the dispersion of the predictions around its mode [40]. For a given instance x, with M output predictions, the variation ratios is calculated as follows,

$$vr(x) = 1 - \frac{\sum_{i=1}^{M}[\![\hat{y}_i = \hat{y}]\!]}{M} \tag{3.8}$$

where $\hat{y}$ corresponds to the sampled majority class obtained and $[\![\hat{y}_i = \hat{y}]\!]$ is an indicator function that takes the value 1 if the expression is true, and to 0 otherwise.

In Figure 3.5, we present the obtained uncertainty values for the Iris dataset using both probability variance and variation ratios. Since only Random Forest is an ensemble method, we used a bootstrap approach to obtain a set of 20 Monte Carlo samples for Naïve Bayes, KNN, and SVM. Comparing both measures, although they are similar, the variation ratio has a greater impact on the uncertainty values than the probability variance. Changes in the predicted label have a significant impact on the variation ratio measure, whereas the impact on the probability variance measure is lower. In variation ratios, we are merely counting changes in the predictions, whereas, in probability variance, we are averaging the differences in the prediction probabilities. For instance, in the KNN

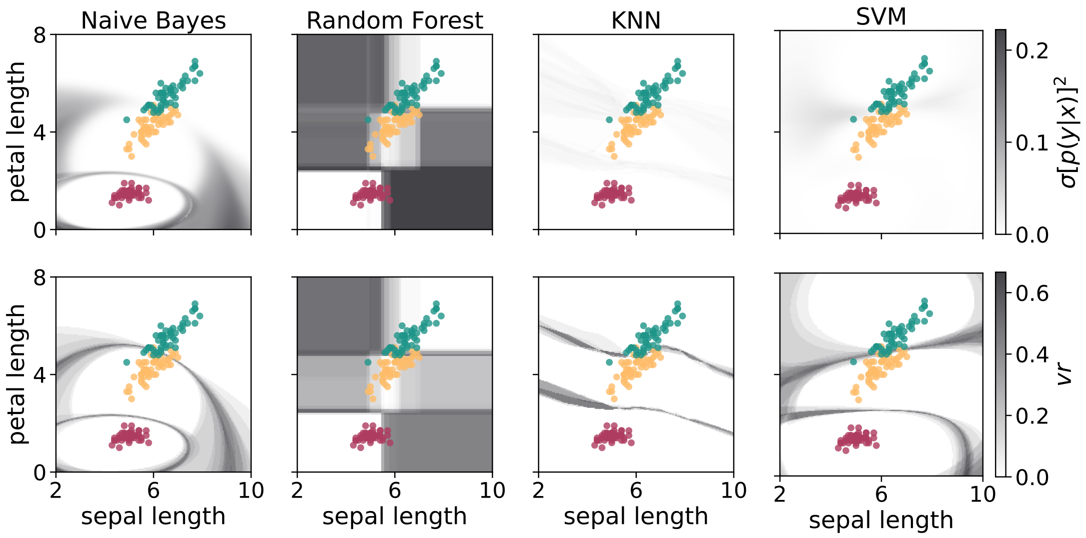


Figure 3.5: Uncertainty values for different classification algorithms using the probability variance (upper plots) and variation ratios (lower plots) as uncertainty measures. The sepal length and petal length features of the Iris dataset are used. The colorbars are set to the minimum and maximum theoretical values of each uncertainty measure.

and SVM classifier, the differences between probability variation and variation ratios are quite evident. We note that both measures have their colorbars fixed to the minimum and maximum theoretical values. Moreover, although these measures are referred to as epistemic uncertainty measures, they are both measuring the uncertainty related to the model, which we refer to as model uncertainty.

Furthermore, an explicit attempt at measuring and separating aleatoric and epistemic uncertainty was made by Depeweg et al. [62] who proposed an approach to quantify and separate uncertainties with classical information-theoretic measures of entropy. Although the approach was proposed in the context of neural networks for regression, the authors' idea was more general and can also be applied to other settings, such as in the work of Shaker et al. [41], where measures of entropy were applied using a random forest classifier, or the work of Malinin et al. [61], who adopted these measures in the context of gradient boosting models. In more detail, the total uncertainty is measured in terms of the entropy of the predictive posterior distribution approximated by:

$$u_t(x) := -\sum_{y\in\mathcal{Y}}\left(\frac{1}{M}\sum_{i=1}^{M}p(y|h_i,x)\right)\log_2\left(\frac{1}{M}\sum_{i=1}^{M}p(y|h_i,x)\right) \tag{3.9}$$

The aleatoric uncertainty is measured considering the average entropy of each individual prediction in terms of the expectation over the entropies of distributions. The idea is that by fixing a hypothesis h, the epistemic uncertainty is essentially removed. Its approximation is given by Equation 3.10:

$$u_a(x) := -\frac{1}{M}\sum_{i=1}^{M}\sum_{y\in\mathcal{Y}}p(y|h_i,x)\log_2 p(y|h_i,x) \tag{3.10}$$

Then, epistemic uncertainty is measured in terms of mutual information between hypotheses and outcomes and can be expressed as the difference between the total uncertainty, captured by the entropy of expected distribution, and the expected data uncertainty, captured by the expected entropy of each individual prediction [19],

$$u_e(x) := u_t(x) - u_a(x) \tag{3.11}$$

Thus, epistemic uncertainty is high if the distribution $p(y|h)$ varies a lot for different hypotheses h with high probability but leading to quite different predictions.

Figure 3.6 shows the decomposition of uncertainty using the classic information-theoretic measures of entropy. Although the range of uncertainty values varies depending on the uncertainty measure used, the conclusions are similar to those drawn from the previous visualizations. Moreover, the epistemic uncertainty obtained from this decomposition is also a measure of model uncertainty and exhibits similar behavior to that observed in Figure 3.5.

Note that in the context of neural networks, epistemic uncertainty is commonly associated with uncertainty about the model parameters, normally referred to as weights

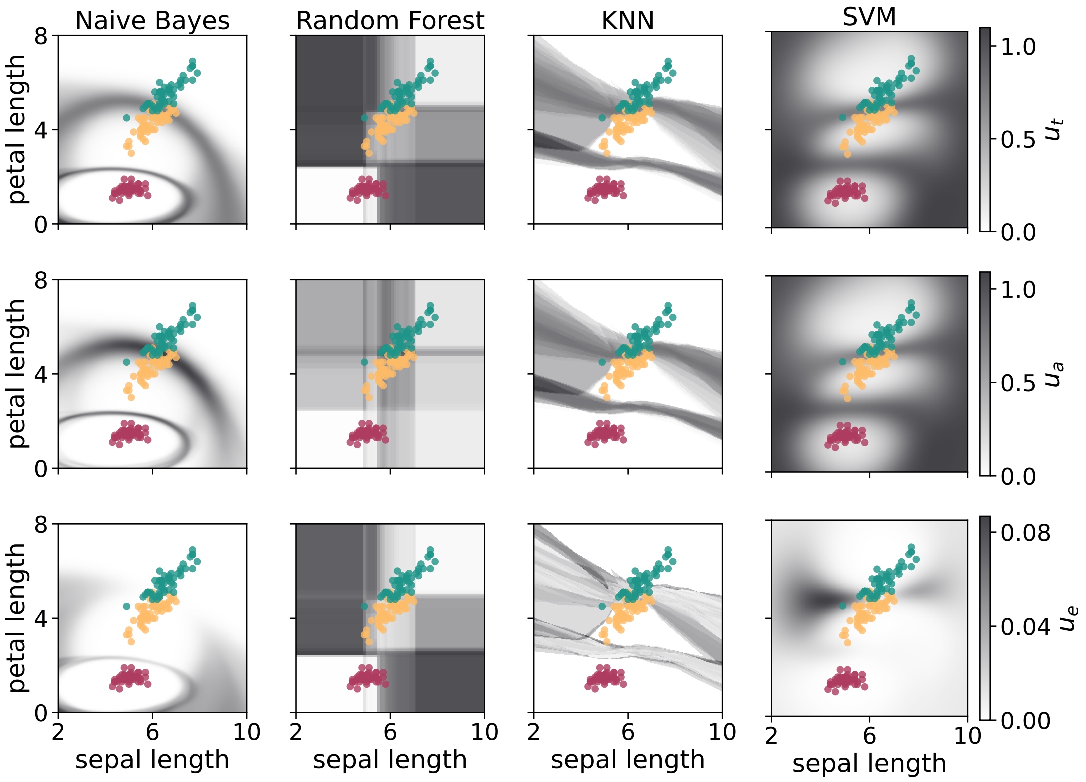


Figure 3.6: Uncertainty values for different classification algorithms using the decomposition of entropy measures. From top to down total uncertainty, aleatoric uncertainty, and epistemic uncertainty.

w. In a classical neural network, the weights are a point estimate. However, a Bayesian extension of deep neural networks was proposed [63, 64], known as Bayesian Neural Network (BNN), where a probability distribution is assigned to each weight, instead of a real number. In this setting, the mathematical formulation of the posterior predictive distribution is as follows:

$$p(y|x, D) = \int p(y|x, w)p(w|D)dw \tag{3.12}$$

In the previous equations, we use the notation of hypothesis to make the interpretation more general. Nonetheless, in the case of neural networks, a hypothesis h can be interpreted by a weight vector w.

3.2.1 Stochastic methods

Most Bayesian inference involves the approximation of integrals that are analytically intractable. This is particularly relevant for Bayesian models that involve the evaluation of complex, high-dimensional integrals. With the advancements in computing technology, a variety of methods have emerged for performing Bayesian inference using different types of approximations. These include analytical approximations such as the Laplace approximation and variational methods, as well as Monte Carlo methods, which are now widely used in Bayesian machine learning.

In *Variational inference* [65], the goal is to approximate a distribution that is close to the posterior distribution obtained by the model, using a predefined parameterized family of distributions, called the *variational distribution*. Variational inference transforms the problem of finding the posterior into an optimization problem. This optimization is performed by minimizing the Kullback-Leibler divergence between the variational distribution and the true posterior.

Peterson and Anderson [66] is arguably the first variational procedure for a particular model, a neural network [65]. Later, Hinton and Van Camp [67] proposed a variational algorithm for a similar neural network model, where the authors derived a diagonal Gaussian approximation to the posterior distribution of neural networks. Several modern approaches can be viewed as extensions of these early works [49].

Another important example in variational methods is Monte Carlo Dropout (MC Dropout) [68], which approximates the posterior with a product of Bernoulli distributions. The method consists of training a neural network with dropout layers, i.e. in each iteration a randomly selected subset of weights is set to zero, and then at the testing time, the predictive uncertainty can be obtained by keeping the dropout active. This work lies at the intersection of variational inference and sampling methods as it reformulates the inherent stochastic elements in neural networks as a form of variational inference.

Monte Carlo methods (also known as *sampling approaches*), represent uncertainty without a parametric model, using a set of samples drawn from the distribution [69]. This kind of approximation has the advantage of not being restricted by the type of distribution. Examples of Monte Carlo methods include Markov Chain Monte Carlo, importance sampling, rejection sampling, and particle filtering [49]. In the context of BNN, Neal [70] introduced the Hybrid Monte Carlo method as a way to perform Bayesian inference in neural networks. The Hybrid Monte Carlo method combines gradient-based optimization with Markov Chain Monte Carlo sampling to overcome the difficulties associated with high-dimensional posterior distributions in deep learning models. This work is considered one of the earliest contributions to Bayesian deep learning and has since inspired many follow-up studies and extensions.

Finally, the *Laplace approximation* approximates the posterior distribution of model parameters using a multivariate Gaussian distribution. The idea behind Laplace approximation is to find the mode of the posterior distribution and then fit a Gaussian distribution to the posterior distribution in the neighborhood of the mode. This approximation is based on the assumption that the posterior distribution is locally Gaussian near its mode, which is often a reasonable assumption in practice. The Laplace approximation is fast and easy to implement, and it is often used as a computationally efficient alternative to more complex approximation methods. It is applied as a post-hoc method, meaning that it can be used after a model has been trained to provide an estimate of the posterior distribution. Mackay [63] and Denker et al. [71] have pioneered the Laplace approximation for neural networks, and several modern methods provide an extension to deep neural networks [49].

3.2.2 Ensemble methods

Ensemble methods have long been recognized as very effective in improving the performance of machine learning and deep learning models with better generalization capabilities due to the use of synergy effects among different models, arguing that a group of decision-makers tends to make better decisions that a single decision-maker [49].

Ensemble methods were originally not introduced to explicitly handle and quantify uncertainties. Although using ensembles as an approximation of Bayesian methods is a concept that has been applied in several machine learning and statistical models [72]. The motivation behind using ensembles as a Bayesian approximation is that they can provide a way to incorporate model uncertainty into the predictions. By aggregating the predictions of multiple models, an ensemble can provide a robust estimate of the posterior distribution, capturing different sources of uncertainty that might be present in individual models.

Instead of directly approximating the posterior $p(h|D)$, as performed with Bayesian methods, the aim of ensembles is to obtain multiple modes of the posterior, where the set of hypotheses can be used as the posterior samples obtained with Bayesian techniques [73]. Therefore, within a Bayesian framework, the variance of the predictions generated by an ensemble is inversely related to the "peakedness" of a posterior distribution. As a result, an ensemble can be viewed as an approximation of the second-order distribution in a Bayesian setting [5].

Thus, the output of an ensemble is given by the mean of the predictions, while the variance corresponds to the epistemic uncertainty. The intuition behind ensemble uncertainty is simple. Assuming that the predictive posterior distribution is approximated by a finite ensemble of M hypotheses, different hypotheses will tend to output similar values when the inputs are similar to the observed training data. However, as inputs become less similar to the training data, the outputs of each hypothesis tend to be more affected by the specificities of the sub-optimal solution reached, thus the higher variance [74].

Different approaches can be applied to create an ensemble. In the context of deep learning, one popular approach was introduced by Lakshminarayanan et al. [72] where the same network is trained M independently times using different parameter initialization on the whole dataset. On the contrary, Bootstrapping, also known as Bagging, is another popular technique that instead of training a model on the whole dataset, varies the distribution of the used training set by sampling new sets of training samples from the original set. Each bootstrap sample is obtained by sampling from the training data uniformly and with replacement [75]. In order to maximize the variety among the members of an ensemble, the combination of different algorithms (or different network architectures) is also commonly applied [76].

Ensemble methods can be computationally expensive, as they require training multiple models, but they can provide a powerful approximation of Bayesian methods in cases where the posterior distribution is complex or high-dimensional.

3.3 Uncertainty evaluation measures

The empirical evaluation of methods for quantifying uncertainty is a non-trivial problem due to the lack of ground truth uncertainty information. A common approach for indirectly evaluating the predicted uncertainty measures is by accessing their usefulness to improve classification performance. In this sense, ranking-based methods can be used to evaluate the uncertainty measures' capability of ordering predictions based on their own uncertainty estimation. The idea is to evaluate how the classification performance varies as a function of the percentage of rejections. If a measure is able to quantify its own uncertainty well, the classification performance should improve with an increasing percentage of rejections. As an example, *accuracy-rejection curves*, which depict the accuracy of a predictor as a function of the percentage of rejections are commonly used in literature [77, 78]. However, this approach can only be directly applied to compare different uncertainty measures using the same predictive model since the classification performance curves depend not only on the uncertainty ordering but also on the predictive model performance.

Therefore, for a fair comparison between uncertainty measures obtained from different predictive models, the Area Under the Confidence-Oracle (AUCO) error introduced by Scalia et al. [74] is more suitable. AUCO computes the area between the theoretically perfect ordering (obtained from the oracle confidence curve) and the ordering made by each uncertainty measure. The oracle confidence curve represents the best possible

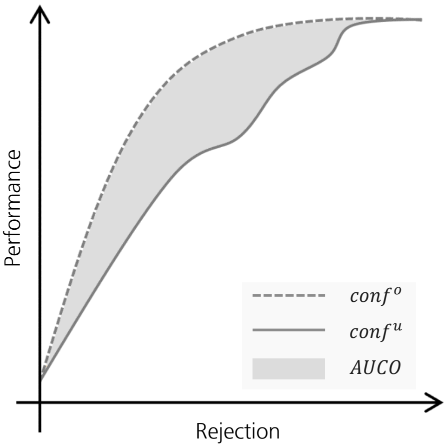


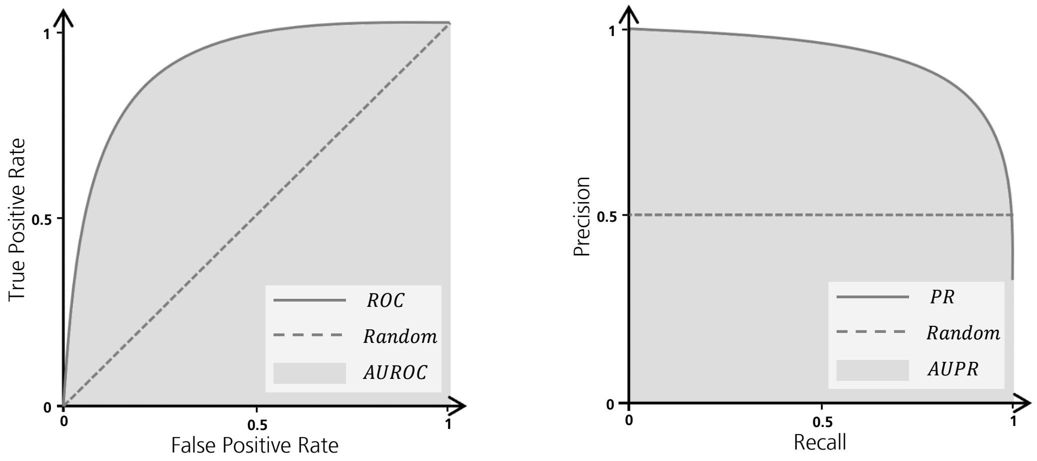
Figure 3.7: Representation of a confidence curve derived from an uncertainty measure depicted by a solid line, the oracle confidence curve depicted by a dashed line, and the Area Under the Confidence-Oracle (AUCO) depicted in grey.

ordering of predictions by their confidence, with the true error imposing the ordering. The AUCO value is calculated as the area under the curve representing the difference between the given uncertainty estimation and the oracle confidence curve. Smaller values of AUCO indicate that the given uncertainty estimation is closer to the oracle confidence curve and therefore is a better predictor of uncertainty. In Figure 5.10 an example of an oracle confidence curve $conf^o$, a confidence curve derived from an uncertainty measure $conf^u$ and the corresponding AUCO is shown. The formula of AUCO is as follows:

$$AUCO = \int_0^1 (conf_r^u - conf_r^o)\,dr \tag{3.13}$$

where $conf^u$ is the confidence curve for a given uncertainty estimation, $conf^o$ is the oracle confidence curve and r is the fraction of rejections. Thus, the integration is performed over the range of confidence values.

Additionally, in recent studies, the two most common measures to evaluate uncertainty methods in the task of distinguishing in-distribution and out-of-distribution samples are the ROC curve and the PR curve. Both methods generate curves based on different thresholds of the underlying measure. The ROC curve depicts the relationship between the True Positives Rate (TPR) and False Positives Rate (FPR), while the PR curve plots the precision against the recall. These curves give a visual idea of how well the underlying measures are suited to distinguishing in-distribution and out-of-distribution samples, but for a quantitative evaluation, the AUROC and AUPR metrics are commonly applied. Both metrics have the advantage of being threshold-independent performance metrics. A



(a) AUROC

(b) AUPR

Figure 3.8: Uncertainty evaluation measures for distinguishing in-distribution and out-of-distribution samples. On the left, the ROC curve is depicted by a solid line, the random predictor is depicted by a dashed line, and the AUROC is shown in grey. On the right, the PR curve is depicted by a solid line, the random predictor is depicted by a dashed line, and the AUPR is shown in grey.

representation of ROC and PR curves, as well as the corresponding AUROC and AUPR metrics, can be seen in Figure 3.8. AUROC can be interpreted as the probability that a positive example is assigned a higher detection score than a negative example. Consequently, a random positive example detector corresponds to a 50% AUROC, and a perfect detector corresponds to an AUROC score of 100% [79].

For the interpretation of AUPR, the baseline detector has an AUPR approximately equal to the precision [80], and a perfect detector has an AUPR of 100%. Consequently, the base rate of the positive class greatly influences the AUPR, so the AUPR-In and AUPR-Out are commonly used, where in-distribution and out-distribution inputs are specified as negatives and positives, respectively. The AUPR is sometimes deemed as more informative than AUROC because the AUROC is not ideal when the positive class and negative class have greatly differing base rates.

As threshold-dependent measures, recent studies in the literature are evaluating uncertainty estimations based on the concept of binary confusion matrix [81, 82]. In this context, predictions are classified as correct or incorrect, and, depending on a threshold, predictions are also classified as certain or uncertain. As a result, four combinations are identified: (i) True Certainty (TC): correct and certain; (ii) True Uncertainty (TU): incorrect and uncertain; (iii) False Uncertainty (FU): correct and uncertain; and (iv) False Certainty (FC): incorrect and certain. Based on these combinations, the following metrics can be calculated: Uncertainty Accuracy (UAcc), Uncertainty Sensitivity (USens), Uncertainty Specificity (USpec), and Uncertainty Precision (UPrec).

$$UAcc = \frac{TU + TC}{TU + TC + FU + FC} \tag{3.14}$$

$$USen = \frac{TU}{TU + FC} \tag{3.15}$$

$$USpec = \frac{TC}{TC + FU} \tag{3.16}$$

$$UPrec = \frac{TU}{TU + FU} \tag{3.17}$$

3.4 Calibration

The previously introduced uncertainty evaluation measures are an important measure to compare different uncertainty estimations, however, they do not take into consideration the actual values expressed by uncertainty. Therefore, calibration methods are important to ensure that the models are well-calibrated and uncertainty measures based on probabilities can be used reliably. If the predicted confidence level accurately reflects the true probability of being correct, i.e. if observed empirical frequencies are consistent with outputting probability distributions, the predictor is considered well-calibrated [83].

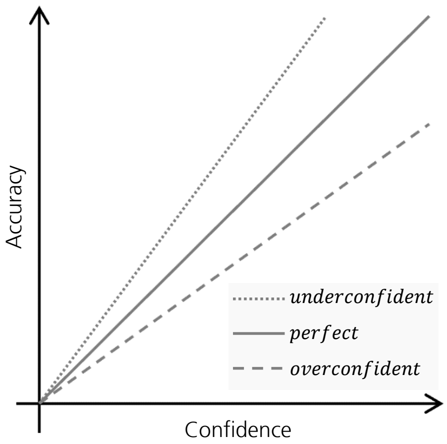


Figure 3.9: Reliability diagram showing an underconfident model (accuracy is larger than the corresponding confidence), overconfident model (accuracy is smaller than the corresponding confidence), and a perfect calibrated model (accuracy is equal to confidence).

To assess if a model is under-confident or over-confident, a reliability diagram (or calibration plot) can be used as a visualization method. Reliability diagrams depict accuracy on the y-axis and average confidence on the x-axis. A perfectly calibrated model outputs probabilities that match up with the accuracy, yielding a diagonal line, where confidence is equal to accuracy. In Figure 3.9 an example of a reliability diagram showing an underconfident, an overconfident, and a well-calibrated model is shown.

For quantitative evaluation of models' calibration, several calibration measures can be considered. A widely used measure, based on binning, is called Expected Calibration Error (ECE) [84]. ECE is calculated as the weighted average of the bin-wise calibration errors given by the following equation:

$$ECE = \sum_{i=1}^{K} p(i) \cdot ||acc(b_i) - conf(b_i)|| \tag{3.18}$$

where K is the number of bins, b_i is the i^{th} bin, and $p(i)$ is the fraction of the predictions that fall into the bin. acc and $conf$ are the average bin accuracy and the average bin confidence, respectively.

In recent years, the evaluation of calibration in neural networks has received attention because it has been demonstrated that as the accuracy of neural networks increases, they become less calibrated [83]. Calibration errors are generally caused by factors related to model uncertainty. There are different calibration methods that intend to improve calibration without compromising accuracy. Indeed, some of the previously introduced uncertainty estimation methods, such as Bayesian or ensemble methods, improve the model's calibration by reducing the model uncertainty [72, 85]. Calibration methods can be divided into three main groups: Regularization methods, post-processing methods, and uncertainty estimation approaches [49]. Regularization methods are applied during the training phase with the objective of obtaining well-calibrated models by modifying the objective function or augmenting the training data. Popular regularization approaches within the uncertainty quantification field are the label smoothing or the direct exposure of models to OOD examples. Post-processing methods are applied after the training procedure and require an independent calibration set to adjust the prediction scores. Examples include histogram binning [86], isotonic regression [87], bayesian binning into quantiles [84], ensemble of near isotonic regression [88] and temperature scaling [83].

Finally, uncertainty estimation approaches can also be applied as calibration methods, since there are some approaches (e.g. Bayesian and ensembling) that while reducing model uncertainty, also lead to better-calibrated models.

3.5 Experimental analysis

In this section, we present the results of experimental analysis to validate our proposed uncertainty measure, KUE, against state-of-the-art methods discussed earlier in the chapter. These results were previously published in the *Machine Learning and Knowledge Extraction Journal* [12]. The two main objectives of the experiment were: (1) to evaluate the performance of our proposed KUE method combined with classical information-theoretic measures of entropy in a classification with rejection option setting, and (2) to assess the effectiveness of KUE in distinguishing between in-distribution and out-of-distribution samples.

3.5.1 Datasets

We designed experiments on different data modalities to evaluate our method and compare it with state-of-the-art methods. As the datasets do not explicitly contain out-of-distribution samples, we adopted a common approach seen in literature to simulate a OSR problem, by re-labeling some of the known classes as unknown [38]. The datasets, instances, attributes, classes, and OOD combinations are summarized in Table 3.1. In the following, a brief description of each dataset is given:

- **Bacteria**[1]: This dataset includes bacterial Raman spectra of 30 common bacterial pathogens treated by eight antibiotics. For the feature extraction, we split each Raman spectra into 125 equal-sized windows corresponding to different wavenumber ranges. For each range we extracted minimum, maximum, and mean features and applied a feed-forward feature selection algorithm, obtaining a set of 50 features. Due to the high number of possible combinations for known and unknown classes, we grouped the 30 classes by empiric antibiotic treatment, resulting in eight OOD combinations that vary in the number of known and unknown classes. Details of the different combinations are available in Appendix A.1.

[1] https://github.com/csho33/bacteria-ID (accessed on June 2020)

- **HAR**[1]: This dataset contains six different human activities (walking, walking upstairs, walking downstairs, sitting, standing and laying) recorded with accelerometer and gyroscope smartphone sensors. This dataset has a set of 561 features available for which we applied a feed-forward feature selection algorithm. For the known and unknown classes split, we defined nine OOD combinations, considering each of the six individual classes as unknown and three additional combinations of classes defined as stairs (walking upstairs and walking downstairs), dynamic (walking, walking upstairs and walking downstairs), and static (sitting, standing, and laying).

- **Digits**[2]: This dataset is composed of 10 handwritten digits (from 0 to 9) and 64 attributes. We used each class as unknown resulting in a total of 10 OOD combinations.

- **Cardio**[3]: This dataset contains measurements of fetal heart rate and uterine contraction on cardiotocograms. The dataset has 10 classes and additional labeling as (Normal, Suspicious, and Pathologic). Thus, we trained the model using only classes labeled as Normal and considered the unknown classes from the labeling Suspicious and Pathologic.

Table 3.1: Datasets used and their characteristics.

Dataset	# Instances	# Attributes	# Classes	# OOD Combinations
Bacteria	3000	50	30	8
HAR	1800	9	6	9
Digits	5620	64	10	10
Cardio	2126	23	10	2

3.5.2 Classification with rejection option

3.5.2.1 Baseline methods

The methods used for the classification with rejection option through uncertainty measures are the following:

1. **Knowledge uncertainty** measured by our proposed KUE (Equation 3.3) using KDE for the probability density function of each feature and Scott's rule [89] for the kernel bandwidth;

[1] https://archive.ics.uci.edu/ml/datasets/human+activity+recognition+using+smartphones (accessed on February 2020)

[2] https://archive.ics.uci.edu/ml/datasets/Optical+Recognition+of+Handwritten+Digits (accessed on July 2020)

[3] https://archive.ics.uci.edu/ml/datasets/Cardiotocography (accessed on July 2020)

2. **Total uncertainty** approximated by the entropy of the predictive posterior using Equation 3.9;
3. **Aleatoric uncertainty** measured with the average entropy of each model in an ensemble using Equation 3.10;
4. **Epistemic uncertainty** expressed as the difference between the total uncertainty and aleatoric uncertainty given by Equation 3.11.

Although KUE can be applied to any ML model with feature-level representation, the measures of total, aleatoric, and epistemic uncertainty are approximated using an ensemble approach. Therefore, a Random Forest classifier with 50 trees and a bootstrap approach to create diversity between the trees of the forest was used for this experiment.

3.5.2.2 Experimental results

As previously introduced, to use KUE in a classification with a rejection option, a threshold must be established to serve as the classification rule. This threshold is determined based on a pre-selected percentile of the uncertainty values in the training data, and the specific percentile chosen may vary depending on the specific application.

As we hypothesized that the percentage of the reject in-distribution data depends on the chosen percentile, we computed the TPR and FPR for a range of train percentiles, as shown in Figure 3.10, considering the positive samples being the samples classified as out-distribution and the negative samples the ones classified as in-distribution. Additionally, as in- and out-distribution detection does not consider the prediction error, we also computed an adjusted FPR where the classification errors were removed from FPR, i.e., the in-distribution inputs that have an uncertainty value higher than the chosen percentile and were misclassified by the model were removed from the FPR. This adjusted FPR is represented in Figure 3.10 by FPR*. This metric has an important meaning for our method since our method depends on the classification performance, where the uncertainty of the misclassified inputs is computed using the probability densities of a different class. Therefore, it is expected that the uncertainty value is high for both OOD and for misclassified inputs.

In Figure 3.10, the variation of TPR, FPR and FPR* according to the train percentile (which defines the uncertainty threshold) for each of the four datasets is presented. Each graph comprises the average and the standard deviation of all OOD combinations for each dataset. As expected, the increase of the train percentile represented almost a linear decrease in FPR, since the distributions of the train data were similar to the distributions of the in-distribution test data. We can see that the FPR* was also linear in all datasets, and both FPR and FPR* converged to 0. This means that depending on the application and on the risk associated with decisions, we can define the train percentile based on how many in-distribution test samples we are willing to reject. On the other hand, TPR followed a different behavior, where a high percentile could reject most of the OOD samples and a few in-distribution test samples or reject a minor percentage of both in- and out-distribution inputs.

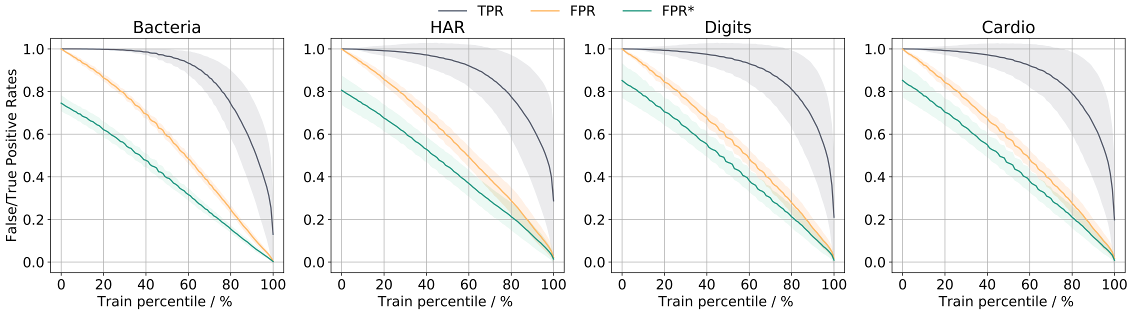


Figure 3.10: Relation between TPR, FPR, FPR* and train percentiles. FPR* stands for an adjusted FPR, where misclassified inputs were removed from the FPR.

Since our proposed approach only deals with knowledge uncertainty, we also quantified the uncertainty in terms of total, aleatoric, and epistemic uncertainty by means of ensemble techniques. Although epistemic uncertainty is a combination of model and knowledge uncertainty, its quantification is limited to the use of ensemble approaches. Moreover, specialized OOD detection methods would probably perform better for the knowledge uncertainty quantification. As our approach is only specialized in OOD detection, and total uncertainty encapsulates the uncertainty of the entire distribution, a combination between them should ideally perform better for the overall classification accuracy. Thus, we combined uncertainties by first rejecting input samples based on our method until the chosen percentile and then rejecting samples based on total uncertainty.

For evaluation, we used accuracy-rejection curves where the prediction uncertainty can be assessed indirectly by the improved prediction as a function of the percentage of the rejection. If we have a reliable measure of uncertainty involved in the classification of test inputs, then uncertainty estimation should correlate with the probability of making a correct decision so that the accuracy should be improved with increasing rejection percentage, and accuracy-rejection curves should be monotone increasing. The comparison between different methods using accuracy-rejection curves should be based on the required accuracy level and/or the appropriate rejection rate [90]. Since we are comparing methods derived from the same classifier, the accuracy-rejection curves always had the same starting accuracy for all methods. Consequently, the relevant variable for the empirical evaluation is the rejection rate. Thus, we moved vertically over the graph to see which method had a higher accuracy for a certain rejection rate. The accuracy-rejection curves were obtained by varying the rejection threshold, where samples with the highest uncertainty values were rejected first.

In Figure 3.11, the average rejection rate against the average accuracy for KUE, total, aleatoric, and epistemic uncertainty is presented. The proposed combination is also shown in black, and the optimal rejection is represented by the dashed line. The optimal accuracy-rejection curve was computed by rejecting all OOD samples as well as misclassified samples in a row. In order to obtain the accuracy-rejection curves we ran 10 random repetitions using 15% of OOD inputs and using an uncertainty train percentile for our proposed combination of 95%. As we can see in Figure 3.11, almost every curve over the different OOD combinations increased the accuracy with the increase of the rejection rate percentage. It is also interesting to note that even with only 15% of OOD inputs, our method always presented the monotone dependency between reject rate and classification accuracy, which means that our method also behaved quite well over the misclassified inputs. Regarding the proposed combination, the accuracy-rejection curve was always better or similar to the total uncertainty. Besides that, we observe that the tendency of the accuracy-rejection curve for the KUE method did not vary much between different OOD combinations, contrary to the aleatoric and epistemic uncertainty. The accuracy-rejection curves for the other datasets can be found in Appendix A.4.

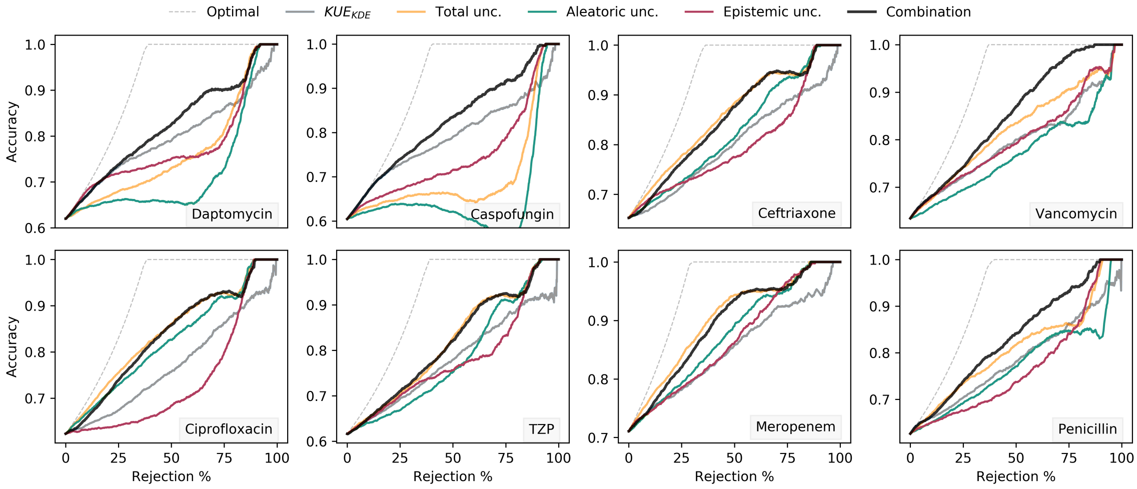


Figure 3.11: Accuracy-rejection curves for aleatoric, epistemic, and total uncertainty for the Bacteria dataset. The curve for perfect rejection is included as a baseline. The name in each plot represents the antibiotic name used for each OOD inputs combination.

3.5.3 Out-of-distribution detection

3.5.3.1 Baseline methods

For comparison purposes, besides the two variations (*KDE* and *Gauss*) of our proposed measures, we selected common approaches to measure uncertainty (2-5) and detect anomalies and/or outliers detection (6-11).

1. $\boldsymbol{KUE_{KDE}}$: Our proposed method for knowledge uncertainty estimation using KDE as the probability density function and Scott's rule [89] for the kernel bandwidth;

2. $\boldsymbol{KUE_{Gauss}}$: Our proposed method for knowledge uncertainty estimation using Gaussian distribution as a probability density function;

3. $\boldsymbol{p(\hat{y}|x)}$: Maximum class probability. Although standard probability estimation is more akin to the aleatoric part of the overall uncertainty, OOD data tend to have lower scores than in-distribution data [79].

4. $\boldsymbol{H[p(y|x)]}$: Total uncertainty modeled by the (Shannon) entropy of the predictive posterior distribution. High entropy of the predictive posterior distribution, and therefore a high predictive uncertainty, suggests that the test input may be OOD [5].

5. $\boldsymbol{I[y,h]}$: Epistemic uncertainty measured in terms of the mutual information between hypotheses and outcomes. High epistemic uncertainty means that $\boldsymbol{p(y|x,h)}$ varies a lot for different hypotheses $\boldsymbol{h}$ with high probability. The existence of different hypotheses, all considered probable but leading to quite different predictions, can indeed be seen as a sign of OOD input [5].

6. **OCSVM**: One-Class SVM introduced by Schölkopf et al. [91] using a radial basis function kernel to allow a non-linear decision boundary. OCSVM learns a decision boundary in feature space to separate in-distribution data from outlier data.

7. **SVM$^{\mathbf{ovo}}$**: Multiclass SVM with one-vs-one approach and calibration across classes using a variation of Platt's extended by [92].

8. **SVM$^{\mathbf{ova}}$**: One-vs-all multiclass strategy fitting one SVM per class.

9. **NCM**: Nearest Class Mean classifier using a probabilistic model based on multiclass logistic regression to obtain class conditional probabilities [93].

10. **OSNN**: Open Set Nearest Neighbors introduced by Júnior et al. [36] using a distance ratio based on the Euclidean distance of two most similar classes.

11. **IF**: Isolation Forest introduced by Liu et al. [55] for anomaly detection using an implementation based on an ensemble of an extremely randomized tree regressor.

Note that epistemic uncertainty is approximated by means of ensemble techniques, which is the representation of the posterior distribution by a finite ensemble of hypotheses. For this reason, to make the comparison fair between baseline methods 1–5, we chose a Random Forest classifier for the analysis of the experiments. Nevertheless, and since different classifiers have different accuracies for the classification of the very same data, a comparison study was carried out on a set of classical algorithms, namely Random Forest, KNN, NB, SVM and Logist Regression. The obtained results can be consulted in Appendix A.2.

3.5.3.2 Experimental results

The training process was done using only in-distribution inputs, ignoring the OOD inputs during training. For the final evaluation, we randomly selected the same number of in-distribution and out-distribution inputs from the test set. Therefore, the main performance measure used for the evaluation of OOD samples was the AUROC, since it is a threshold-independent performance metric applied by most of the recent studies [94]. Table 3.2 compares KUE using two variants of the feature modeling (KDE and Gaussian) with the 9 methods previously mentioned. The OOD names shown in Table 3.2 indicate the assumed unknown classes for each dataset. Regarding the Bacteria dataset, the names are the antibiotic treatments used to group the unknown classes, which are detailed in Appendix A.1. The AUROC represents the average results over 10 random repetitions for a total of 29 OOD combinations over 4 different datasets. Additional details about AUPR-In and AUPR-Out can be found in Appendix A.3.

From a detailed analysis of Table 3.2 we notice that, in the majority of the OOD combinations, our method obtained better or comparable AUROC with other methods. Moreover, the proposed method performed more consistently for different OOD combinations, unlike the other methods that showed unstable behaviors, where the standard deviation was very large over all combinations considered. For instance, the OCSVM presented the highest performance on the Digits and Cardio datasets. However, in the other datasets its performance varied a lot depending on the assumed unknown classes, with poor performance on several OOD combinations. As an example in Figure 3.12, we show the ROC curves for the Caspofungin and Ciprofloxacin OOD combinations of the Bacteria dataset, representing the best and the worst performance of our method in the Bacteria dataset, respectively. It is interesting to note that, after our method, OCSVM presented the highest performance for Caspofungin. However, for Ciprofloxacin the OCSVM performance was lower than random. A similar behavior happened with the maximum class probability, $p(y|x)$, and the total uncertainty, $H[p(y|x)]$, which are the best methods to detect OOD samples on Ciprofloxacin combination and the worse in the case of antibiotic Caspofungin. Both methods had the same behavior over all combinations due to their intrinsic dependency. Maximum class probability can also be seen as a measure of the total uncertainty in predictions. Regarding epistemic uncertainty, although it obtained a few poor performances, it seemed to have more consistent behavior than the other methods. Additionally, it can be seen that all methods obtained high AUROC and comparable performance for all combinations of the Digits dataset. Comparing our two feature modeling strategies (KDE and Gaussian), we observed that results were similar, probably due to the fact that the feature modeling using the KDE in our datasets was approximated to a Gaussian distribution.

Additionally, a more qualitative interpretation of AUROC values presented by Hendrycks and Gimpel [79] was also used to ease the process of comparison. Hendrycks and Gimpel defined the intervals for evaluation as follows: excellent: 90–100%, good: 80–90%, fair: 70–80%, poor: 60–70%, fail: 50–60%. The results are presented in Table 3.3, where the values represent the number of occurrences in each AUROC interval over all datasets. From this table, we can easily conclude that the KUE method was at least more robust to changes in OOD combinations/datasets than compared to state-of-the-art methods.

Table 3.2: AUROC for detecting OOD test inputs using two variants of KUE (KDE and Gaussian) and other baseline methods on 4 datasets. The Mean and Standard Deviation (SD) over OOD combinations are presented after each dataset. All values are averages over 10 consecutive repetitions.

		AUROC										
	OOD	KUE_{KDE}	KUE_{G}	$p(\hat{y}\|x)$	$H[p(y\|x)]$	$I[y,h]$	$OCSVM$	SVM^{ovo}	SVM^{ova}	NCM	$OSNN$	IF
Bacteria	Daptomycin	0.91	0.90	0.67	0.68	0.88	0.57	0.89	0.79	0.60	0.59	0.66
	Caspofungin	0.98	0.98	0.37	0.31	0.87	0.98	0.56	0.62	0.32	0.44	0.93
	Ceftriaxone	0.82	0.81	0.82	0.85	0.85	0.50	0.91	0.83	0.82	0.77	0.35
	Vancomycin	0.87	0.87	0.67	0.66	0.80	0.73	0.82	0.73	0.64	0.61	0.74
	Ciprofloxacin	0.74	0.74	0.86	0.91	0.71	0.35	0.88	0.80	0.81	0.75	0.22
	TZP	0.88	0.88	0.76	0.75	0.89	0.65	0.89	0.80	0.84	0.80	0.51
	Meropenem	0.77	0.77	0.87	0.87	0.78	0.48	0.87	0.84	0.83	0.78	0.43
	Penicillin	0.77	0.77	0.73	0.74	0.67	0.60	0.81	0.83	0.70	0.69	0.65
	Mean	0.84	0.84	0.72	0.72	0.81	0.61	0.83	0.78	0.70	0.68	0.56
	SD	0.08	0.08	0.15	0.18	0.08	0.18	0.11	0.07	0.17	0.12	0.21
HAR	Walking	0.69	0.67	0.77	0.78	0.80	0.21	0.73	0.70	0.74	0.73	0.23
	Upstairs	0.86	0.86	0.79	0.82	0.85	0.42	0.71	0.67	0.80	0.72	0.44
	Downstairs	0.84	0.84	0.72	0.70	0.72	0.88	0.55	0.73	0.45	0.70	0.89
	Sitting	0.73	0.75	0.52	0.52	0.66	0.69	0.50	0.43	0.51	0.46	0.66
	Standing	0.54	0.50	0.62	0.67	0.82	0.58	0.54	0.70	0.55	0.44	0.58
	Laying	0.99	0.99	0.25	0.26	0.39	0.99	0.14	0.20	0.79	0.51	1.00
	Stairs	0.90	0.89	0.54	0.58	0.73	0.78	0.25	0.39	0.49	0.43	0.80
	Dynamic	1.00	1.00	0.72	0.76	0.75	0.82	0.57	0.58	0.87	0.92	0.86
	Static	1.00	0.98	0.70	0.69	0.75	0.99	0.29	0.58	0.50	0.81	0.99
	Mean	0.84	0.83	0.63	0.64	0.72	0.71	0.48	0.55	0.63	0.64	0.72
	SD	0.15	0.16	0.16	0.16	0.13	0.25	0.19	0.17	0.15	0.17	0.25
Digits	0	0.93	0.95	0.90	0.90	0.97	1.00	0.95	0.90	0.90	0.98	0.80
	1	0.68	0.81	0.88	0.89	0.84	0.95	0.78	0.87	0.85	0.91	0.63
	2	0.90	0.92	0.90	0.89	0.87	0.99	0.90	0.90	0.90	0.95	0.84
	3	0.75	0.81	0.90	0.87	0.82	0.97	0.86	0.82	0.86	0.95	0.64
	4	0.94	0.95	0.88	0.89	0.96	0.99	0.85	0.92	0.84	0.93	0.94
	5	0.87	0.88	0.90	0.89	0.89	0.98	0.84	0.85	0.88	0.96	0.67
	6	0.92	0.93	0.88	0.88	0.97	0.99	0.95	0.85	0.93	0.97	0.81
	7	0.91	0.92	0.94	0.95	0.91	0.99	0.89	0.87	0.93	0.96	0.86
	8	0.90	0.87	0.97	0.98	0.94	0.97	0.96	0.95	0.92	0.96	0.45
	9	0.88	0.89	0.89	0.87	0.84	0.94	0.90	0.87	0.86	0.96	0.61
	Mean	0.87	0.89	0.90	0.90	0.90	0.98	0.89	0.88	0.89	0.95	0.73
	SD	0.08	0.05	0.03	0.03	0.05	0.02	0.05	0.04	0.03	0.02	0.14
Cardio	Suspect	0.67	0.65	0.33	0.31	0.45	0.75	0.48	0.50	0.31	0.67	0.71
	Pathologic	0.83	0.85	0.36	0.31	0.51	0.98	0.23	0.75	0.30	0.66	0.94
	Mean	0.75	0.75	0.34	0.31	0.48	0.86	0.36	0.62	0.30	0.66	0.82
	SD	0.08	0.10	0.01	0.00	0.03	0.12	0.12	0.12	0.01	0.01	0.11
	Mean	0.84	0.85	0.73	0.73	0.79	0.78	0.71	0.73	0.72	0.76	0.68
	SD	0.11	0.11	0.20	0.20	0.14	0.23	0.24	0.17	0.20	0.18	0.21

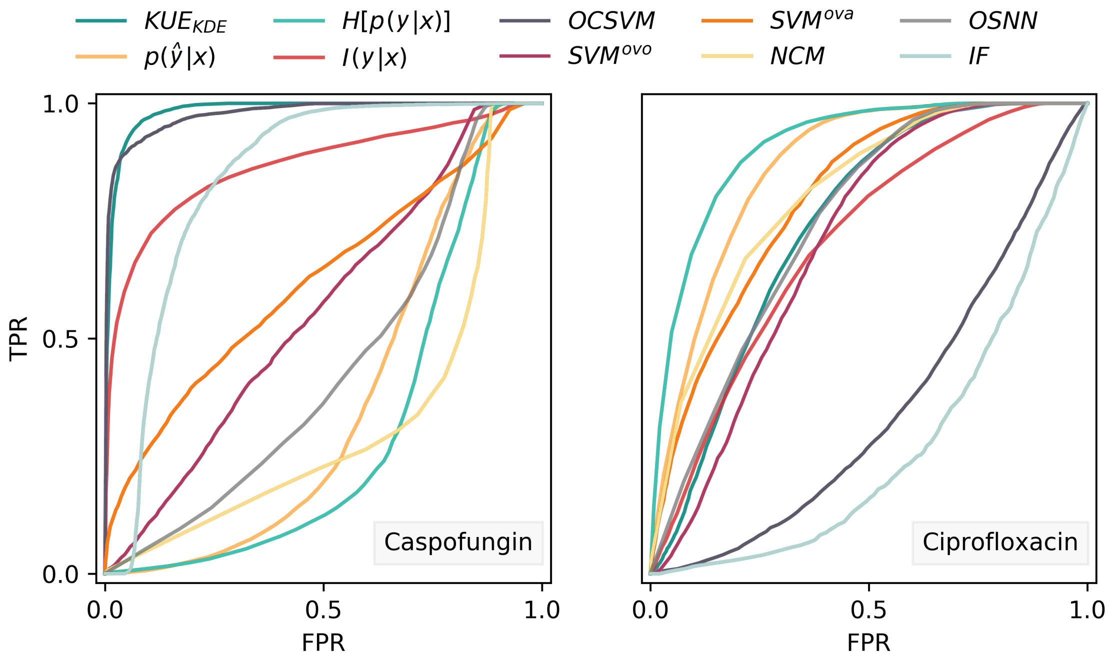


Figure 3.12: ROC curves for OOD detection using our KUE method and baseline methods on Caspofungin and Ciprofloxacin OOD combinations of the Bacteria dataset. Caspofungin and Ciprogloxacin represent the best and the worst performances of KUE method, respectively.

Unlike the other methods, our method did not obtain any OOD worse than random. We can also see that OCSVM had more occurrences of an excellent qualitative evaluation, but also one of which had more fail and random classifications.

Table 3.3: Qualitative AUROC evaluation over all OOD combinations. *Excellent*: 90–100%, *Good*: 80–90%, *Fair*: 70–80%, *Poor*: 60–70%, *Fail*: 50–60%, ↓ *Random*: < 50%

	KUE_{KDE}	KUE_G	$p(\hat{y}\|x)$	$H[p(y\|x)]$	$I[y,h]$	OCSVM	SVM^{ovo}	SVM^{ova}	NCM	OSNN	IF
Excellent	9	10	5	4	5	14	4	3	5	11	5
Good	11	12	8	10	12	2	12	11	11	2	7
Fair	5	4	6	5	7	3	3	7	2	5	2
Poor	3	2	4	4	2	2	0	2	2	5	7
Fail	1	1	2	2	1	4	5	3	4	2	2
↓ Random	0	0	4	4	2	4	5	3	5	4	6

Since our proposed approach for OOD detection is based on a density estimation technique, and density estimation typically requires a large sample size, we performed an ablation study to evaluate how the AUROC's results change with the number of train samples used for modeling. In Figure 3.13, we present the results of the ablation study for the four datasets used, where we rejected gradually 5% of the original number of train samples in each iteration, making a total of 20 iterations for each OOD combination. We can see that the AUROC values did not change significantly with the number of train samples. This means that the number of training samples caused small changes in feature modeling, resulting in minor variations in the performance of our method.

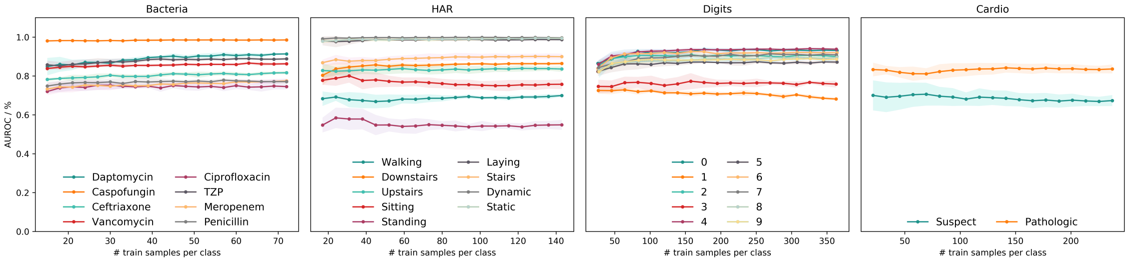


Figure 3.13: Ablation study of KUE method using KDE, for the four datasets. The legend represents each OOD inputs combination, and the title of each plot represents the dataset used.

3.5.4 Discussion

In this experiment analysis, our goal was to demonstrate the performance of the proposed knowledge uncertainty estimation method, KUE, for both classification with rejection option and the detection of out-of-distribution samples.

The proposed KUE method is based on a feature level density estimation of in distribution train data, and it does not rely on out-distribution inputs for hyperparameters tuning nor for threshold selection. In literature the idea of using densities to detect out-of-distribution examples is prominent [95, 96], however, the available methods are usually developed to do both the predictive task and the uncertainty estimation. Following the reasoning of Raghu et al. [54] which states that uncertainty quantification and prediction tasks should be two separate tasks for uncertainty quantification to be unbiased and since different classifiers have different accuracies for the classification of the very same data, we proposed a method that, although dependent on the classification accuracy, can be easily applied to any feature-level model without changing the underlying classification methodology. Another property of KUE is the independence between classes. This means that the training and uncertainty prediction do not depend on the other classes, allowing for the addition and removal of classes without the need to repeat the feature density estimation for all classes. This property also allows the definition of the r^{th} percentile per class if required for the task.

Moreover, as a parametric model for the data is often difficult to determine, we proposed the use of a KDE method for feature density estimation. However, due to the computational cost of KDE with the increase in training size, we also compared the proposed method using a Gaussian distribution assumption. For the four different datasets used for evaluation, Gaussian estimation showed similar results with KDE, which can significantly reduce the computational cost on large datasets. Nevertheless, if possible, the train data distribution can be calculated to choose the best parametric model to be applied.

One potential limitation of our proposed KUE measure is the naïve assumption of independence between features. In real-world scenarios, features often interact with each other, and ignoring these dependencies can lead to suboptimal performance and inaccurate uncertainty estimates. However, in some cases, assuming independence between features can simplify the modeling process and still yield good results, as demonstrated in our experimental analysis. Future work could explore efficient methods to incorporate feature dependencies into our measure and test it on large-scale datasets. Addressing this limitation may also require the development of more complex models that can capture feature interactions while still being computationally feasible.

Regarding the task of distinguishing in- and out-distribution inputs, our KUE method showed competitive performance results comparable to state-of-the-art methods using the AUROC as a performance evaluation measure. Furthermore, we also defined a threshold for OOD input rejection that is chosen based on the percentage of in-distribution test samples that we are willing to reject. We showed its dependency on FPR and also demonstrated that misclassified inputs tend to have high uncertainty values. Although the proposed threshold selection strategy effectively controlled the FPR, the TPR had a high variability between different datasets, and it was not possible to estimate its behavior for unknown inputs. For future research, this limitation should be addressed by combining KUE with different methods adopting a hybrid generative discriminative model perspective.

The aleatoric, epistemic, and total uncertainty produced by measures of entropy showed a monotone dependency between reject rate and classification accuracy, which confirmed that these measures of uncertainty are a reliable indicator of the uncertainty involved in a classification decision. Moreover, the proposed uncertainty measures combination between the proposed KUE method and total uncertainty outperformed the individual entropy measures of uncertainty for the classification with a rejection option. Future research includes the study of different combination strategies of uncertainty measures for classification with a rejection option. In addition, expanding the testing scenarios with more datasets should provide more indications about the robustness of the measures used. If more specialized OOD detection methods are able to properly quantify their own uncertainty, different combinations between existing methods and other sources of uncertainty should also be explored.

3.6 Final remarks

The field of uncertainty quantification in machine learning has gained significant attention in recent years. However, despite the growing number of approaches, there is still no standard method for uncertainty estimation, and there is no consensus for a common taxonomy. In this chapter, we proposed a classification scheme that divides uncertainty quantification methods into two main groups, namely Single Methods and Bayesian Methods, which are further subdivided into additional categories.

While the available methods for uncertainty quantification have been shown to be valuable for separating and quantifying aleatoric and epistemic uncertainty, they still present some limitations, particularly in the estimation of knowledge uncertainty. In the context of knowledge uncertainty, there are several methods dedicated to anomaly, outlier, or novelty detection. However, these methods can be too complex or do not generalize well for different settings. Furthermore, many of these methods are often tailored to specific types of data and may not be easily adaptable to different applications.

To address these challenges and since different classifiers have different accuracies for the classification of the very same data, we proposed an agnostic method for knowledge uncertainty estimation that, although dependent on the classification performance, can be easily applied to any feature-level model without changing the underlying classification methodology.

4

Uncertainty for Model Design

Although UQ plays an important role in safety-critical domains, such as medicine [97], it is also an important concept within the machine learning methodology itself. UQ is important across several stakeholders of the ML lifecycle. It helps developers debug their models, in understanding their flaws so they can be used for model improvement. For the users of AI systems, UQ increases interpretability and trust in model predictions, answering the question: *Can I trust this model?* For regulators and certification bodies, it contributes to algorithm auditing and quality control as a path towards the effective and reliable application of ML systems [98].

Previous research has been focused on the development of techniques to characterize and quantify uncertainty. In this context, recent uncertainty frameworks have been proposed that provide different capabilities to quantify and evaluate uncertainty in the AI development lifecycle [99, 100]. However, few studies addressed a comprehensive analysis of how UQ can be used to improve model performance and its interpretability. Interpretability is a crucial aspect of machine learning systems, as it enables the provision of not only predictions but also explanations of their outputs. By facilitating an enhanced understanding of the rationale behind the prediction, interpretable machine learning systems play a vital role in fostering trust and safety. This is achieved by providing insights into the reasonableness of the prediction, thereby enabling users to identify any areas of concern or potential risks. Additionally, designing a model for a new and unexplored research domain can be challenging, where being able to understand how the model is working can assist in the development process.

This chapter focused on leveraging the outcome from uncertainty quantification to improve the model development process. We applied the UQ concept in practice, giving insights into why it can be an effective procedure to improve model development and its interpretability.

This chapter aims to address three main research questions in which access to UQ measures can aid in model selection, model combination, and model interpretability tasks. We formulate the research questions as follows:

1. How can UQ contribute to choosing the most suitable model for a given classification task?
2. Can UQ be used to combine different models in a principled manner?
3. Can UQ be employed to enhance models' interpretability?

This chapter is organized into two main research works that, along with the validation of experimental results, provide insights into the practical usefulness of UQ in addressing the aforementioned research questions. In the first section, we demonstrate the importance of uncertainty-based rejection for model selection, model combination, and the interpretability of classifiers with a rejection option. In this section, we validate our experiments using a synthetic dataset and a HAR dataset. The results presented in this section were already published in [101]. Subsequently, in the second section, we build upon the preliminary results of the first section by using a multimodal dataset, focusing specifically on model combination and interpretability[1]. We propose a novel measure to assess explanation complexity and present evidence that an uncertainty-weighted model combination can reduce feature-based explanation complexity. The chapter concludes with final remarks on the obtained results.

4.1 Uncertainty-based rejection

In conventional classification tasks, classifiers are typically forced to predict a label. However, for difficult samples, this might result in misclassification, which can pose challenges in risk-sensitive applications. In such situations, it may be more appropriate to refrain from making decisions on difficult cases, thereby anticipating a lower error rate on the examples for which a classification decision is made [1]. Classification with a rejection option allows models to abstain from making predictions when uncertain, leading to improved performance and robustness in practical applications.

In this section, we introduce the utility of uncertainty-based rejection during the development process of machine learning models. The preliminaries section begins by presenting the basic concepts related to uncertainty-based rejection and providing details on the experimental setup. Subsequently, the following three sections discuss the experimental results concerning model selection, model combination, and interpretability of rejection.

4.1.1 Preliminaries

4.1.1.1 Baseline methods

To address the research questions previously introduced, different uncertainty measures can be used to model the different types of uncertainty. However, in the current chapter

[1] The results presented in this section were submitted for an open-access journal and are part of a collaboration with a Ph.D. candidate in the area of explainable AI.

our main objective is not to compare different uncertainty measures, but instead show how different sources of uncertainty can be used, in general, to help practitioners in the development of more robust models.

Thus, aleatoric, model, and knowledge uncertainties will be modeled using the following measures throughout this chapter:

- **Aleatoric uncertainty**: The (Shannon) entropy is the most notable measure of uncertainty for probability distributions. Although it can be seen as a measure of the total uncertainty in predictions [19], this measure is primarily capturing the shape of the distribution and, hence, is mostly concerned with the aleatoric part of the overall uncertainty [5]. Thus, we will measure aleatoric uncertainty using Equation 3.2;

- **Model uncertainty**: Variation ratios defined by Equation 3.8 was selected as a primary uncertainty quantification method, to estimate model uncertainty. As we are interested in evaluating the quality of the model fit, changes in the predicted label have a significant impact on the variation ratio measure. Contrarily, measures based on entropies, which are commonly used, can also be used, but the impact on the measure is lower, since in variation ratios, we are merely counting changes in the predictions, and in entropy measures, we are averaging the prediction probabilities [40];

- **Knowledge uncertainty**: Although the majority of works addressed the quantification of knowledge uncertainty with measures such as the mutual information (see Equation 3.11), we argue that these kinds of measures are more akin to model uncertainty. The uncertainty related to the lack of data might be poorly modeled by these measures. In this perspective, we considered density estimation methods, commonly used for outlier or novelty detection, as more prone to model knowledge uncertainty. Thus, our proposed KUE measure defined in Equation 3.3 was used to model knowledge uncertainty.

To measure model uncertainty, ensemble techniques will be used as an approximation approach. For cases where the learning algorithm is not an ensemble model, we will use the bootstrap method [102] to approximate the sampling distribution and compute uncertainty measures. The bootstrap method uses Monte Carlo simulation to approximate the sampling distribution by repeatedly simulating bootstrap samples, which are new datasets created by sampling with replacement from the uniform distribution over the original dataset. To bootstrap a supervised learning algorithm, one would need to sample M bootstrap datasets and run the learning procedure from scratch each time.

Each uncertainty measure was used as a rejection measure for the classification with a rejection option setting. The problem of choosing the optimal rejection point is not trivial and simple criteria were applied in this work. For a more comprehensive analysis of optimal rejection thresholds, the works from Condessa et al. [48] and Fisher et al. [103] can be consulted. Therefore, in our classification setting, the final prediction is given by the following rejection rule:

$$\hat{\omega} = \begin{cases} reject & \text{if } \Phi(x) > 0 \\ f(x) & otherwise \end{cases} \tag{4.1}$$

where $f(x)$ is the classifier without rejection and $\Phi(x)$ is a function on the input that evaluates the uncertainty of the prediction model. This uncertainty function is given by the set of uncertainties—aleatoric (a), model (m), and knowledge (k)—through the following equation:

$$\Phi(x) = \sum_{u \in U} 1[\phi_u(x) > \tau_u] \tag{4.2}$$

where $U \in [a, m, k]$ is the set of available uncertainties, ϕ_u is an uncertainty function that evaluates uncertainty u, and τ_u is a threshold for the rejection point for uncertainty u.

For aleatoric uncertainty, ϕ_a represents the Equation (3.2) and the optimal threshold, τ_a, was obtained using the following equation:

$$\tau_a = arg \max_{\theta} \left(\mathscr{E}_\theta - \frac{b}{1-b} \cdot \mathscr{L}_\theta \right) \tag{4.3}$$

where θ is a threshold in the interval $[0, 1]$, representing a normalized entropy value measured with Equation (3.2), and b is a rejection cost, here set to 0.5. $\mathscr{E}_\theta$ and $\mathscr{L}_\theta$ represent the subset of true rejects and false rejects for the threshold θ, respectively.

The uncertainty function used for model uncertainty, ϕ_m, is equal to Equation (3.8), and τ_m was set to zero, which means that a prediction must be equal in all bootstraps samples to not be rejected. This assumption was made because if a sample is predicted differently using slightly different datasets, the model in that particular region will still have some uncertainty associated.

For knowledge uncertainty, ϕ_k is equal to Equation (3.3). To define τ_k, we used a 95% value of the training uncertainty values, meaning that $\tau_k = P_{95\%}[KUE]$. A detailed description of this approach is available in [12].

In summary, our proposed approach was developed in the context of classification with rejection where rejection was obtained through measures of uncertainty. These uncertainty measures were distinguished by three different sources: aleatoric, model, and knowledge uncertainty. For the uncertainty quantification, we used an entropy measure for aleatoric uncertainty (Equation (3.2)), the variation ratio measure for model uncertainty (Equation (3.8)), and KUE to quantify the knowledge uncertainty. Regarding the rejection setting, we applied the rejection rule from Equation (4.1), where each source of uncertainty has an uncertainty function given by Equation (4.2). For the training procedure, a bootstrap approach with 20 bootstrap samples was used, and the uncertainty measures were calculated.

4.1.1.2 Rejection-based evaluation

As previously introduced in Chapter 3.3, the empirical evaluation of methods for quantifying uncertainty is usually done through standard metrics, such as accuracy, to obtain an accuracy rejection curve [77]. However, for evaluating the performance of classifiers with rejection, in addition to non-rejected accuracy, additional metrics should be considered. Condessa et al. [48] expanded the set of performance measures for classification with rejection and, besides the non-rejected accuracy, proposed two novel performance measures to evaluate the best rejection point, namely classification quality and rejection quality. These measures were employed in this analysis, and a brief explanation is presented below.

Considering a partition of a set of samples in subsets A, M, N, and R, where A is a subset of accurately classified samples, M is a subset of misclassified samples, N is a subset of nonrejected samples, and R is a subset of the rejected samples, each metric can be derived as follows:

- **Nonrejected accuracy** measures the ability of the classifier to accurately classify nonrejected samples, and it is computed as,

$$NRA = \frac{|A \cap N|}{|N|}; \tag{4.4}$$

- **Classification quality** measures the ability of the classifier with rejection to accurately classify nonrejected samples and to reject misclassified samples. It is computed as,

$$CQ = \frac{|A \cap N| + |M \cap R|}{|N| + |R|}; \tag{4.5}$$

- **Rejection quality** measures the ability of the classifier with rejection to make errors on rejected samples only, and it is computed as,

$$RQ = \frac{|M \cap R||A|}{|A \cap R||M|}. \tag{4.6}$$

The nonrejected accuracy and the classification quality are bounded in the interval $[0,1]$. Unlike these measures, the rejection quality has a minimum value of zero, and its maximum is unbounded by construction. Nonetheless, the higher the values, the better the metric performs for rejection.

4.1.1.3 Datasets

Predicted uncertainties are often evaluated indirectly, as data usually do not contain information about any form of "ground truth" uncertainties. For this reason, using a synthetic dataset can more readily offer insight into the different types of uncertainties and their quantification. Moreover, in a controllable setting, we can alter the size of the datasets, evaluate the models' performance and uncertainties under various conditions, and introduce noise into the data to assess the models' robustness

Considering the reasons mentioned above, we first validate our experiments on a synthetic dataset to provide an intuition of the potential use of UQ in addressing the research questions, and then apply the same reasoning to a real-world dataset of HAR.

- **Synthetic dataset**: We developed a dataset generator to facilitate our experimental analysis. This generator includes the option to define the number of classes, the number of features, and feature distribution-related parameters. The features can be informative (independently drawn), redundant (random linear combinations of informative features), or useless (randomly drawn). The features can also be modeled using different feature distributions, such as Gaussian, Uniform, or Exponential distribution. Additionally, the features can be modeled as unimodal or bimodal distributions.

- **HAR**: A benchmark dataset for Human Activity Recognition (HAR) [104] from the University of California Irvine (UCI) repository [105] was selected for these experiments. This dataset contains six different human activities (*walking*, *walking upstairs*, *walking downstairs*, *sitting*, *standing*, and *laying*) collected with a group of 30 volunteers and recorded using accelerometer and gyroscope smartphone sensors. The time series signals were pre-processed by applying noise filters and sampled in fixed-width sliding windows of 2.56 seconds with a time shift of 1.28 seconds. For this analysis, the 561 features provided with the dataset were used.

4.1.2 Uncertainty for model selection

In machine learning, various criteria can be employed in the problem of model selection. Model selection involves choosing a final model from a set of candidate models for a given training dataset. This process can be applied to different types of algorithms or the same type configured with different hyperparameters. The primary goal of model selection is to achieve the best predictive performance for modeling learning data and for making predictions for new examples that were not part of the learning process [106, 107].

In supervised learning, predictive performance is usually considered the most critical criterion for model selection. However, various criteria for the predictive model quality, such as interpretability or computational cost, can also play a significant role in model selection. To the best of our knowledge, uncertainty has not been considered as a criterion for model selection. Therefore, in this section, we explore how uncertainty might contribute to model characterization by providing valuable quantitative information, either by describing the quality of the model's fit or evaluating if sufficient training data were provided to generate reliable predictions.

Table 4.1: Performance measures (mean ± standard deviation) for different models using a training size of 7692 samples. The highlighted baseline accuracies represent the selected models that were considered for further analysis, since the models attained similar accuracy values.

Model	**Baseline Accuracy**	**Nonrejected Accuracy**	**Rejection Fraction**
Gaussian Naive Bayes	**0.838 ± 0.004**	0.861 ± 0.004	0.056 ± 0.006
KDE Naive Bayes	0.918 ± 0.004	0.929 ± 0.004	0.050 ± 0.007
Exponential Naive Bayes	**0.848 ± 0.012**	0.894 ± 0.011	0.109 ± 0.041
KDE Bayes	**0.845 ± 0.003**	0.914 ± 0.004	0.178 ± 0.004
Logistic Regression	0.717 ± 0.003	0.788 ± 0.005	0.198 ± 0.006
Decision Tree	0.764 ± 0.024	0.884 ± 0.004	0.328 ± 0.111
Random Forest	0.806 ± 0.004	0.871 ± 0.006	0.169 ± 0.004
k-Nearest Neighbors	0.820 ± 0.004	0.902 ± 0.007	0.202 ± 0.005
Support Vector Machines	0.744 ± 0.004	0.806 ± 0.005	0.173 ± 0.010

In the following sections, we start with experiments on a synthetic dataset and then apply the same methodology to the HAR dataset to address our first research question: *How can UQ contribute to choosing the most suitable model for a given classification task?*

4.1.2.1 Experiments on synthetic data

A dataset composed of a total of 150,000 ten-dimensional points corresponding to six different classes equally distributed was generated. Features from each class were modeled using Gaussian, exponential, and uniform distributions. The distributions were randomly selected and could be unimodal or bimodal distributions. To evaluate the behavior of uncertainty estimations with the increasing number of training samples, the models were trained for different training sizes using a k-fold cross-validation as the validation strategy where k was set to 5. An exponential growth of training samples was applied, starting with 50 samples per class (training size equals 300 samples).

For model training, different classifiers using a training size of 7692 samples were tested as presented in Table 4.1. Since features data were simulated using Gaussian, exponential, and uniform distributions, a focus on Bayesian models using Gaussian, KDE, and exponential distributions was employed. As expected, Bayesian models obtained higher baseline accuracies than the other tested classifiers, since part of the features likelihood was modeled with the true data distribution. The three classifiers highlighted in Table 4.1, with a similar baseline accuracy, were selected to continue the analysis. These classifiers were: (1) the NB classifier where the features likelihood was assumed to be Gaussian; (2) the NB classifier where the features likelihood was assumed to be exponential; (3) the Bayes classifier where the features likelihood was based on KDE. Additionally, the selected classifiers were trained using a bootstrap procedure with 20 bootstrap samples.

For this analysis, only aleatoric and model uncertainty measures were considered, since a synthetic dataset without outliers was used. Therefore, KUE would be near zero and would not bring relevant information for this analysis.

Figure 4.1 shows the rejection fraction and accuracy with the increasing number of training samples for the different tested models. The rejection fraction was obtained using both aleatoric and model uncertainty measures independently, and the nonrejected accuracy was obtained by rejecting all samples with aleatoric and/or model uncertainty (see Equation (4.1)).

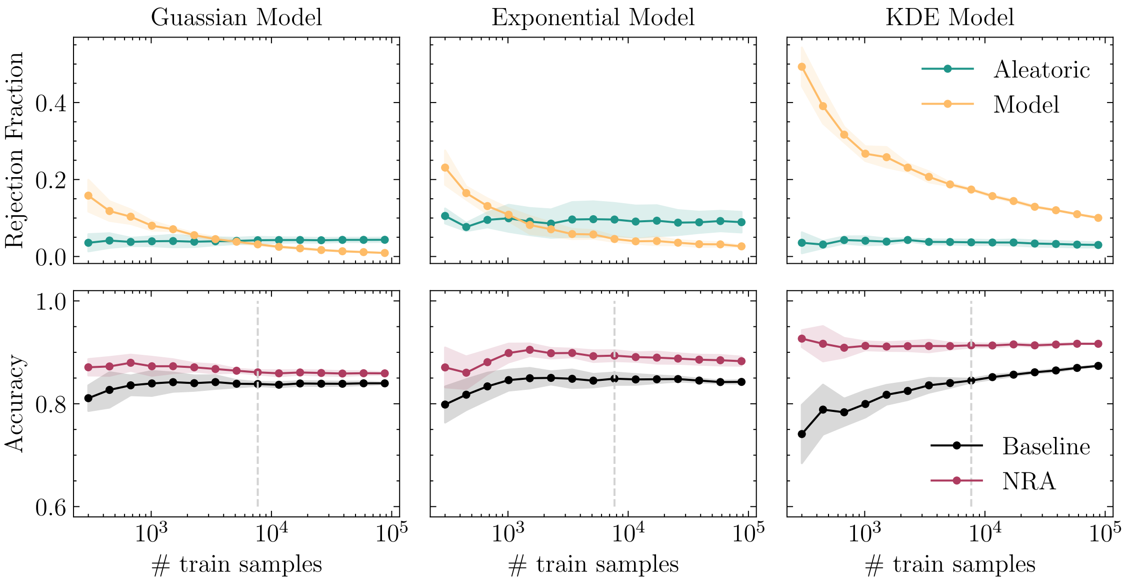


Figure 4.1: Uncertainties' rejection fraction and obtained accuracies using a k-fold cross-validation with an increasing number of training samples for 3 different models. The vertical line represents a training size that obtained a similar baseline accuracy for all models.

As previously mentioned, the model's accuracy is often one of the most important elements to model selection. However, we argue that uncertainty quantification methods should also be evaluated during the model's training, to help us choose the right model. Observing Figure 4.1, different models can achieve the same accuracy, but with different degrees of uncertainty. For example, for a training size of 7692 samples (dashed gray line in Figure 4.1), the three models obtained a baseline accuracy of 84%, approximately. Seen only from this point of view, the decision between the three models would be equal. However, observing the rejection fraction from uncertainty measures, it is easy to understand that the KDE model had higher model uncertainty compared with the other two models. The reason for this difference is that the KDE model is more complex, which means that it needs more data to correctly model the data distribution. Therefore, the differences in the bootstrap samples have a high impact on the model fit, meaning that the same sample is classified differently depending on the bootstrap sample used to fit the model. Additionally, observing the standard deviation with the increasing number of training samples, we can note a slight decrease in both the rejection fraction and accuracy values, except from the exponential model, which seemed to have an almost constant value across the different training sizes. Using this information and since the accuracy was approximately equal for the three models, the choice of a Gaussian NB would be probably preferable due to its low aleatoric and model uncertainty.

Nonetheless, if the rejection of samples or the addition of new samples is an option, a different analysis can be performed. By definition, aleatoric uncertainty is irreducible for the same dataset, which was verified with these experimental results. Increasing the number of training samples did not change the aleatoric uncertainty, making the rejection fraction mostly constant across the different training sizes. Contrarily, model uncertainty decreased with the increase in the number of training samples, tending towards zero when the model fit was equal for all bootstrap samples. Thus, the analysis of model uncertainty can give us insights about the usefulness of adding more samples for the model's training. In Gaussian and KDE models, the decrease of model uncertainty had a clear increase in the baseline accuracy. For the Gaussian NB model, from 10^3 training samples, the baseline accuracy was mostly constant and the decrease of model uncertainty was not significant. This means that the model fit did not change using different bootstrap samples, and the addition of new data did not improve the model's performance. However, observing the KDE model, due to its high rejection fraction of model uncertainty, the addition of new samples still increased the model's performance. Furthermore, the nonrejected accuracy was always higher than the baseline accuracy, and it was mostly constant across the different training sizes. This means that the model uncertainty measure was in fact detecting the regions in the feature space responsible for a high number of misclassifications due to a poor model fit.

4.1.2.2 Experiments on HAR dataset

In order to broaden our analysis, we conducted an additional experiment with the HAR benchmark dataset from the UCI repository [105]. Besides the importance of UQ for trustworthy ML systems, the use of uncertainty measures for human movements analysis plays also an important role in the recognition of abnormal human activities or the analysis, diagnosis, and monitoring of neurodegenerative conditions [108]. Furthermore, the high number of available samples (10,299 samples) in this dataset allowed us to make a similar evaluation to the synthetic data. For the data split into training and test sets, we used the available partition in the repository, where 70% of the volunteers were selected for generating the training data and 30% the test data. Regarding the feature vector, the original 561-feature vector with time and frequency domain variables was reduced using features correlation and the sequential forward feature selector, resulting in a 17-dimensional feature vector.

Similar to the previous section with synthetic data, we applied a training size exponential growth, starting with 300 samples (50 per class) until the maximum training size

Table 4.2: Performance measures for different models using a training size of 7352 samples and the Human Activity Recognition (HAR) dataset.

Model	Baseline Accuracy	Nonrejected Accuracy	Rejection Fraction
Gaussian Naive Bayes	0.89	0.90	0.03
KDE Bayes	0.88	0.92	0.12
Logistic Regression	0.89	0.92	0.06
Decision Tree	0.82	0.92	0.23
Random Forest	0.84	0.91	0.16
k-Nearest Neighbors	0.87	0.94	0.13
Support Vector Machines	0.91	0.93	0.06

of 7352 samples. For model training, we tested different classifiers with 20 bootstrap samples. Table 4.2 shows the obtained baseline accuracy, as well as the nonrejected accuracy and the rejection fraction for each of the tested classifiers. To visualize the behavior of accuracy and the corresponding rejection fraction for each type of uncertainty, we selected the 4 models that obtained higher baseline accuracy. Figure 4.2 shows these performance measures with the increased number of samples used to train the classifiers.

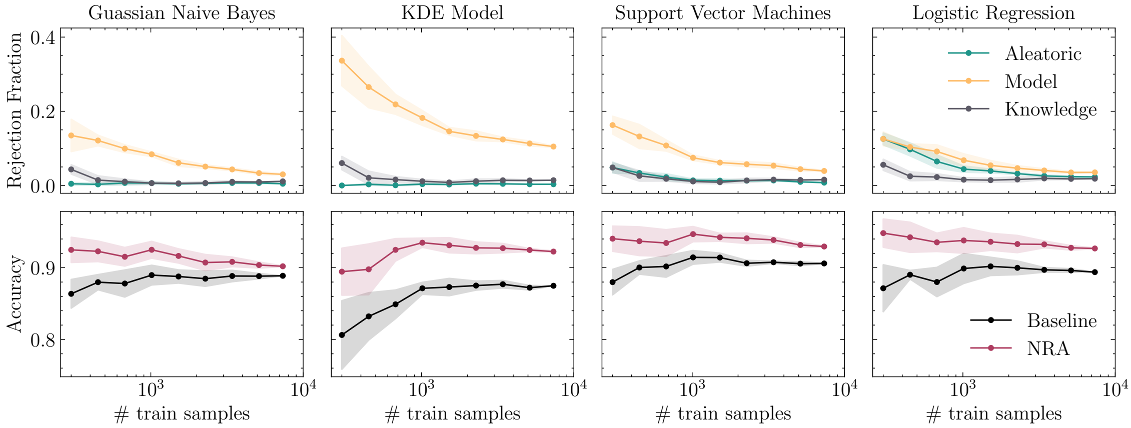


Figure 4.2: Uncertainties' rejection fraction and obtained accuracies with the increasing number of training samples for the Human Activity Recognition (HAR) dataset.

For the HAR dataset, the rejection fraction obtained with both the aleatoric and knowledge uncertainty measures presented a low value for all training sizes and classifiers being analyzed. As expected, regarding the model uncertainty, the rejection fraction decreased with the increasing number of training samples for all classifiers, where more complex classifiers had a higher rejection fraction than simpler classifiers. Due to the low obtained uncertainty (rejection fraction < 4%) and satisfactory accuracy (baseline accuracy of 89%), the Gaussian NB classifier was selected.

4.1.3 Uncertainty for models' combination

In particularly complex classification problems, it is often found that performance can be improved by combining multiple models instead of just using a single one. It had been observed, that although one model would yield the best performance for a given classification task, the sets of observations misclassified by the different models would not necessarily overlap. This suggested that different models potentially offered complementary information about the observations to be classified which could be harnessed to improve the performance of the selected model [109]. In general, ensembles require heterogeneity of predictions to be successful, regardless of the combination rule. This can be achieved through different feature sets or parameter settings for identical learning models, or through different learning algorithms using the same features. The key is to avoid identical erroneous decisions on the same observation instances so that the individual classifiers provide complementary information.

There are several combination rules to train and combine different models. Some rules address models' combination using the average of the predictions or the class probabilities. Nevertheless, the uncertainty of multiple models is seldom considered. Thus, we address how uncertainty can be taken into account for model combination, in the second research question *Can UQ be used to combine different models in a principled manner?*.

4.1.3.1 Experiments on synthetic data

From the analysis of the previous section, we observed that different models had different degrees of uncertainty for the same training size. Since different models are based on different assumptions, we hypothesized that uncertainty measures can be used to combine different models, producing a more robust model. In order to validate this hypothesis, a new synthetic dataset composed of 150,000 ten-dimensional points corresponding to six different classes equally distributed and modeled as a bimodal Gaussian distribution was generated.

A Gaussian NB classifier and a KDE Bayes classifier were trained using a bootstrap approach with 20 bootstrap samples. Figure 4.3 illustrates the rejection fraction and accuracy of both models with an increasing number of training samples. Since the Gaussian NB model fits the feature data with unimodal distributions, while the dataset contains features modeled as bimodal Gaussian distributions, the Gaussian NB classifier has a high rejection fraction due to aleatoric uncertainty, caused by high overlap between the fitted distributions. In contrast, the KDE Bayes classifier is well-suited for bimodal distributions, resulting in low overlap between classes and a low rejection fraction due to aleatoric uncertainty. With regard to model uncertainty, both models initially have high rejection fractions, but the Gaussian NB classifier reached an almost zero rejection rate at 10^5 training samples, while the KDE model, due to its complexity, had a rejection rate of approximately 10%. In summary, the Gaussian NB classifier exhibits high aleatoric uncertainty and low model uncertainty, while the KDE Bayes classifier has low aleatoric uncertainty and high model uncertainty.

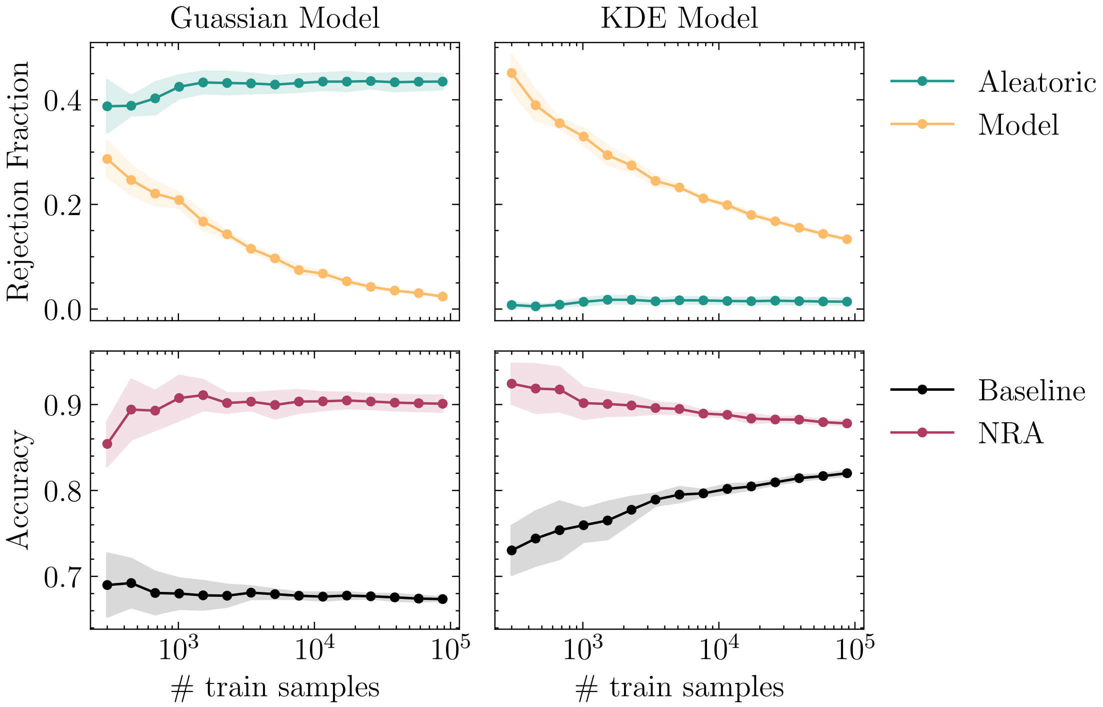


Figure 4.3: Uncertainties' rejection fraction and obtained accuracies using a k-fold cross-validation with an increasing number of training samples for Gaussian NB and KDE Bayes models.

To verify the potential for combining both models using uncertainty measures, the following combination rules were applied:

$$\hat{\omega} = \begin{cases} f_{c_1}(x) & \text{if } \Phi_{c_1}(x) = 0 \text{ and } \Phi_{c_2}(x) > 0 \\ f_{c_2}(x) & \text{if } \Phi_{c_1}(x) > 0 \text{ and } \Phi_{c_2}(x) = 0 \\ f_{c_1}(x) & \text{if } \Phi_{c_1}(x) = 0 \text{ and } \Phi_{c_2}(x) = 0 \text{ and } f_{c_1}(x) = f_{c_2}(x) \\ reject & otherwise \end{cases} \tag{4.7}$$

where c_1 and c_2 represent the Gaussian NB and KDE Bayes classifier and Φ_c is the uncertainty function defined in Equation (4.2).

To validate that the proposed combination strategy performed better than the individual models, we applied the performance measures proposed in the work of Condessa et al. [48]. To compare the performance of the classifiers with rejection, we used 10% of the rejected samples with the highest available training size (approximately 90,000 training samples).

Table 4.3 shows the obtained results for the three models using a 10% rejection fraction. The combination strategy using the uncertainties of both individual models resulted in higher values for the three performance measures for classifiers with rejection, namely the non-rejected accuracy, classification quality, and rejection quality.

These preliminary results demonstrate that access to uncertainty estimations during the model development process may be a useful source of information to develop more robust models. Although a simpler model, such as a NB classifier, may have lower performance compared to more complex models, the use of uncertainty estimations can provide information about the specific regions where the model has low uncertainty. Using this information in combination with more powerful models can increase overall model performance.

Table 4.3: Performance measures for individual models (Gaussian naive Bayes and KDE Bayes) and a combination of both models. The results were obtained using a rejection fraction of 10% and a training size of 90,000 samples.

Model	Nonrejected Accuracy	Classification Quality	Rejection Quality
Gaussian Naive Bayes	0.72	0.72	2.60
KDE Bayes	0.85	0.82	5.84
Model's Combination	0.86	0.83	6.89

4.1.3.2 Experiments on HAR dataset

To validate the combination strategy using the HAR dataset, we decided to combine the two trained models with low accuracy and high uncertainty (see Figure 4.2). Thus, we combined the KDE Bayes model and logistic regression for the different training sizes. To ensure the same rejection fraction for the three classifiers, we employed the obtained rejection fraction for the models' combination, given by Equation (4.7), for both the KDE Bayes and logistic regression classifiers.

Figure 4.4 shows the performance measures for classification with rejection for the individual models and their combination. The results show that the combination strategy outperformed the individual classifiers for almost all training sizes and performance measures. Notably, the combination strategy always resulted in a lower rejection fraction than the obtained rejection fraction for the individual classifiers, as can be confirmed by analyzing Figures 4.2 and 4.4.

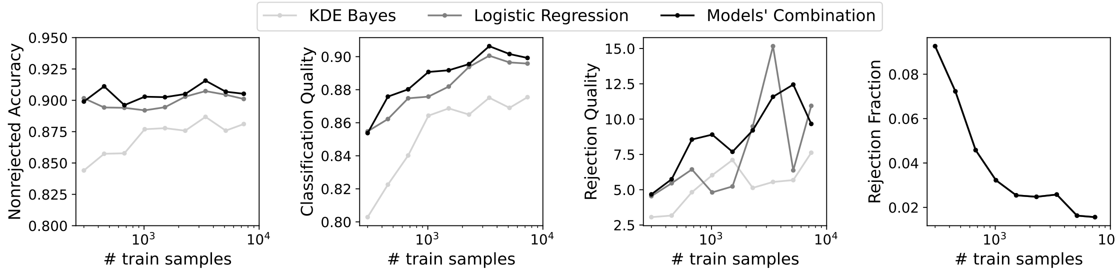


Figure 4.4: Performance measures for classification with rejection for different training sizes.

4.1.4 Interpretability of rejection via uncertainty visualization

In high-stakes applications where machine learning models are employed, auditing tools are essential to building confidence in the models and their decisions. In addition to quantification metrics, visualization techniques have been utilized to support the interpretability of classification models. In this context, Neto et al. [110] proposed a visualization explainable matrix applied to random forests, focusing on global and local explanations where confidence scores were used as an interpretability measure. However, regarding uncertainty visualization for a specific prediction or applied to the model itself, there have been few, if any, studies addressing this aspect in the literature. Consequently, to tackle our final research question, *Can UQ be employed to enhance models' interpretability?*, we quantify the different sources of uncertainty using visualization methods. These visualizations aim to aid in interpreting the models' uncertainty during model development and to audit individual decisions.

For an overview of the uncertainty estimation obtained during the model's development of HAR dataset, a representation of the overall uncertainty is shown in Figure 4.5. In this visualization, the x-axis represents the number of samples where samples are ordered by uncertainty. Using this ordering scheme, it was possible to interpret the overall dataset uncertainty (upper bar), as well as the proportion of the different sources of uncertainty across the dataset (lower bars). The size of each bar represents the number of samples rejected by each type of uncertainty. Although in this dataset only 4% of the test samples were rejected, we can make some observations about the uncertain samples. The majority of uncertain samples rejected by aleatoric uncertainty were also rejected by model uncertainty. Regions with an overlap between classes (aleatoric uncertainty) were also regions where it was expected that the model fit would change between bootstrap samples. In the case of knowledge uncertainty, it was expected that samples with knowledge uncertainty would not have aleatoric uncertainty. However, for model uncertainty, it is possible that some samples shared both model and knowledge uncertainty, which is also verified with Figure 4.5.

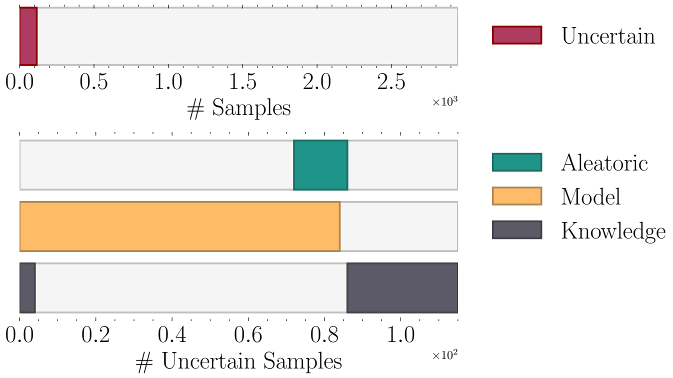


Figure 4.5: HAR dataset uncertainty overview.

Similarly, this representation can be applied to each individual class. Figure 4.6 shows the uncertainty distribution by class. From it, we can conclude that aleatoric uncertainty was presented only in *walking*, *walking upstairs* and *walking downstairs*, which makes perfect sense due to the similarity of these three classes. It is also possible to note that *laying* class did not have aleatoric or model uncertainty. However, it was the class with the highest knowledge uncertainty. Both *sitting* and *standing* classes had a similar pattern in terms of uncertainty, where the *sitting* class was the one with the highest number of uncertain samples.

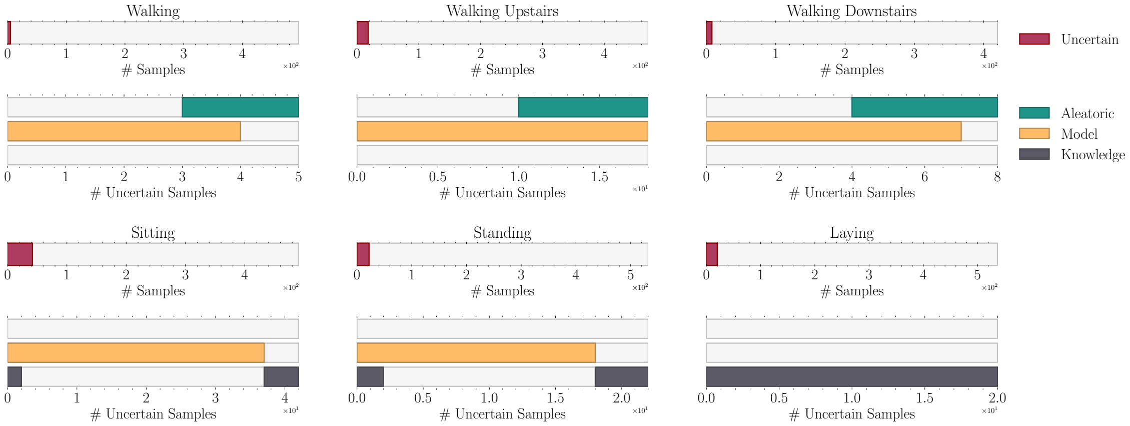
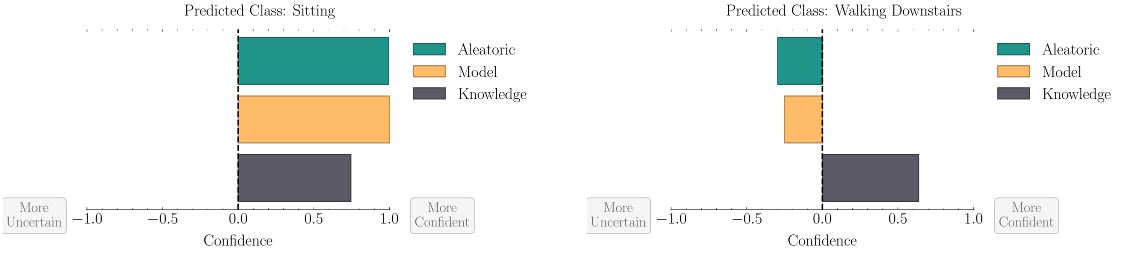


Figure 4.6: HAR dataset uncertainty overview by class.

Besides the visualization applied to the overall classifier's uncertainty, an alternative is to audit the reliability of a given prediction, answering questions such as: *Can I trust this prediction? Why did I reject this sample?*

For this purpose, using the uncertainty estimations for each type of uncertainty, Figure 4.7 was obtained. In this visualization, the bar's size represents how much the model is confident or uncertain about a prediction by uncertainty type. To make the visualization more intuitive, 0 confidence/uncertainty represents the obtained threshold for rejecting a sample. Then, the bars' sizes were normalized between 0 and 1 by the maximum/minimum theoretical value for each uncertainty.

Note that, in the aleatoric uncertainty, we visualize the prediction's expected data entropy, meaning that in Figure 4.7(a), the prediction probability was near 100% (entropy of 0). In the case of Figure 4.7(b), the obtained entropy was greater than the defined threshold for rejection and its value represents approximately 1/3 of the entropies that range between 1 and the rejection threshold. In the case of model uncertainty, we evaluated if a given prediction changes between different bootstrap samples, i.e., the bar's size represents the normalized variation ratios. In Figure 4.7(a), the prediction was the same in all bootstrap samples, obtaining a maximum confidence value, and in Figure 4.7(b), the prediction changed half the total number of possibilities, which are given by the number of bootstrap samples and the number of classes. For instance, this dataset had 6 classes, and 20 bootstrap samples were used, meaning that the maximum variation ratio was 0.8, and the prediction from Figure 4.7(b) obtained a variation ratio of 0.4. Finally, the knowledge uncertainty represents how much a prediction is similar to the training dataset, in terms of probability density. Thus, both Figures 4.7(a) and (b) represent predictions wherein the combination of feature densities led to an uncertainty distance comparable to several samples from the training data. In other words, using KDE as a density estimator, these predictions share similar feature densities with several training samples.



(a) Prediction accepted by aleatoric, model, and knowledge uncertainty

(b) Prediction rejected by aleatoric and model uncertainty.

Figure 4.7: Prediction uncertainty. A confidence value of 0 represents the obtained threshold for rejecting a sample by the uncertainty source. Bars' sizes are normalized between the maximum theoretical confidence/uncertainty.

4.2 Explainability meets uncertainty quantification

Feature importance evaluation is one of the prevalent approaches to interpreting black box ML models. These techniques allow for visualizing the importance ranking of each feature and measuring how much a given feature contributes to the prediction. However, these methods often require a sort of cutoff to get the most informative feature set, which is usually a non-trivial task. Insights from social sciences [111] support simple model explanations that involve the minimum number of features. Therefore, it is important to research methodologies for diminishing the complexity of the explication. This issue becomes particularly relevant in the context of explaining multimodal machine learning models, where the complexity of the problem is compounded by an increase in both the total number of features and the variety of data modalities being considered. An example scenario is the use of wearable data, where multiple sensors are worn across multiple body regions and retrieve information from multiple data modalities. A single model that learns from these high-dimensional datasets is often particularly complex and might attempt to model dozens of features from a wide spectrum of data modalities.

Multimodal learning attempts to model the combination of different modalities of data. However, it is often found that improved performance can be obtained by combining multiple models together instead of just using a single model in isolation. The most straightforward approach to combining models is calculating the mode from the predictions of a group of individual models. Such a methodology can be justified from a frequentist viewpoint by weighing the balance between bias and variance. However, such a combination process assumes that all models perform equally well regardless of their reliability or predictive uncertainty. Alternative aggregation methods can incorporate dynamic weights based on UQ for the aggregation process. Dynamic weighting enables the ensemble model to consider the uncertainty level of each individual model prediction, attributing more weight to the predictions of more certain models and less weight to those more uncertain.

In this section, we propose a novel approach to lower the explanation complexity of multimodal data using uncertainty quantification. Figure 4.8 outlines the intuition of the proposed approach. A baseline model uses all the data modalities of the dataset. We use UQ to aggregate specialized models for each modality by accounting for their respective uncertainty in predicting a given sample. This process allows for discarding less confident models and using a subset of modalities for making predictions. By reducing the number of modalities taken into account, the complexity of the explication is also minimized.

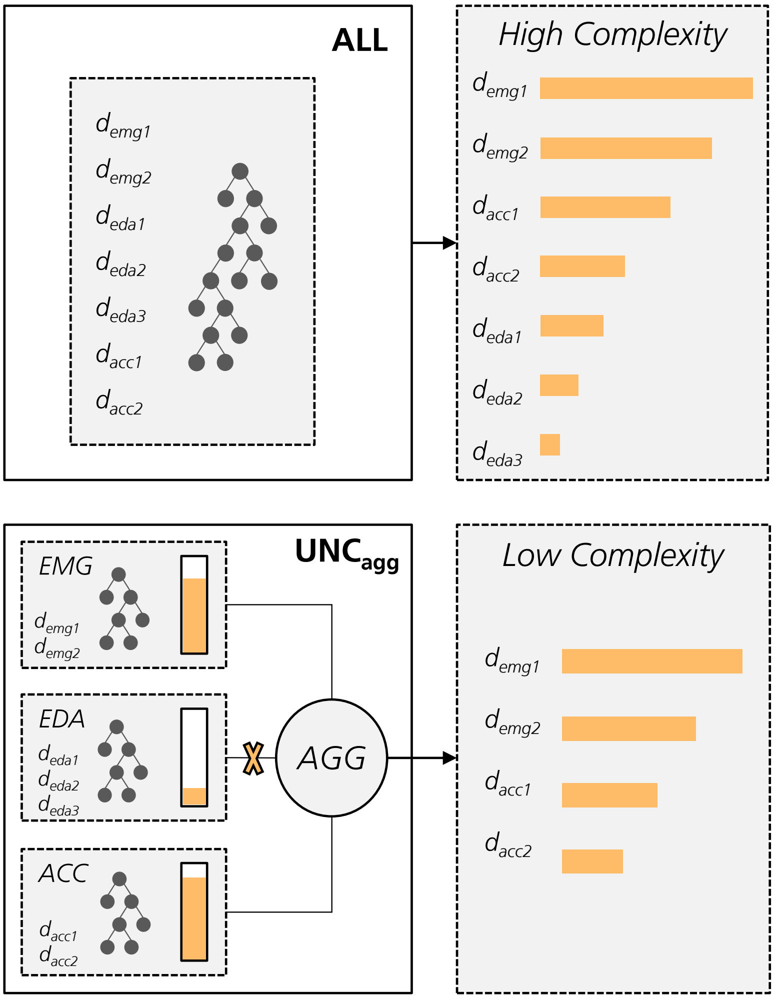


Figure 4.8: Schematic of our framework. (upper) An early fusion model learns with features from different modalities. It often results in a complex high-dimensional explanation output that hinders human analysis (lower). Our proposed approach with model aggregation using uncertainty quantification that rejects the most uncertain models. The reduced number of modalities and features yields simpler explanations.

Another important aspect of our proposed approach is handling missing data, a common issue in multimodal time series analysis [112]. Our proposed model combination approach allows the models trained on the other modalities to handle the missing data in one modality. For example, if one of the modalities is missing data for a particular time window, the other models can still provide predictions based on the available data. This can help improve the model's robustness and ensure that it continues to provide accurate predictions even in the presence of missing modalities.

In the subsequent subsections, we start with an overview of essential background information on aggregation methods and feature-based explanations. Next, we detail the proposed approach and conclude the section with the experimental results.

4.2.1 Preliminaries

4.2.1.1 Aggregation strategies

The main idea behind combination approaches is to exploit the characteristics of several independent classifiers by combining them to achieve higher performance than the best single model. This approach appears in the literature under several names such as multi-classifiers combination, multi-classifiers fusion, mixture of experts, ensemble-based classification systems, among others [113].

Despite the idea of combining models is not new, an interesting issue in the research community is to find the best combination rule for the task at hand. There are several combination rules to train and combine different models. Some rules address model combinations using voting mechanisms based on individual classifiers' predictions, while others use aggregation techniques based on the classifiers' class probabilities. The former is commonly called hard voting (also known as majority voting), and the latter is soft voting.

In the literature, it is common to use weights in the model combination process since models often exhibit varying performance levels. Some studies, such as those by [114] and [115], that leverage weights based on models' reliability scores outperform baseline methods with equal weights. However, dynamic weighting schemes tend to yield better results than fixed weights [113]. This superior performance can be attributed to the dynamic combination's ability to update weights assigned to individual classifiers before making the final decision. For example, Poh et al. [116] proposed a quality-based combination approach for multimodal biometrics. The underlying concept is that quality issues affecting one modality (e.g., signal noise) often do not impact other modalities. Consequently, the proposed combination method assigns higher weights to more reliable classifiers under specific conditions.

Different combination strategies exist, but the predictive uncertainty of the individual classifiers in the ensemble is seldom considered. In the following section, we will discuss different aggregation strategies and present various measures of uncertainty quantification that we proposed to use as weights for the aggregation methods.

Consider a standard setting of supervised learning with a finite training dataset, $D = \{(x_i, y_i)\}_i^N \subset \mathcal{X} \times \mathcal{Y}$, with N samples, composed of pairs of input instances x and outcomes y, where $\mathcal{X}$ is an instance space, $\mathcal{Y}$ the set of outcomes that can be associated with an instance. Suppose a hypothesis space $\mathcal{H}$ composed by a finite ensemble of M hypothesis, where a hypothesis h maps instances x to outcomes y. An individual model can be seen as a hypothesis of the ensemble.

One of the most common forms of aggregation is majority voting. As the name suggests, the predicted class label is obtained by considering the vote of each individual classifier with equal importance, i.e., the final prediction is the most frequently predicted class label. This method offers the benefit of directly handling the outputs of individual classifiers without the need for probabilistic modeling. However, it assumes that all classifiers perform equally well regardless of their reliability or predictive uncertainty [117]. In general, the individual models' performances are not similar, so it is reasonable to assign higher weights to the decision made by the more accurate classifiers using a weighted majority voting defined as:

$$agg_{vote}(x) = arg \max_{y \in \mathcal{Y}} \sum_{h \in \mathcal{H}} w_h * [\![\hat{y}_h = y]\!] \tag{4.8}$$

where $w_h \in \mathbb{R}^+$ denotes the weight associated with hypothesis h and $[\![\hat{y}_h = y]\!]$ is an indicator function that takes the value 1 if the expression is true, and to 0 otherwise. If $w_h = 1$ for all M hypotheses in the ensemble, we recover standard majority voting.

Instead of using the predicted class label, aggregation based on the class probabilities of each individual classifier can be made. One of the most straightforward methods in this soft-level aggregation is the average or sum of class probabilities. Although these soft aggregation methods consider more information in the combination process, they require different models to approximate the same function. Otherwise, the predictions are incomparable, and averaging is not a meaningful operation [118]. Moreover, calibrating individual classifiers' probabilities can be challenging in the combination process. These aggregations can also be turned into a weighted version. As an example, the sum rule quantifies the likelihood of a hypothesis by combining the class probabilities generated by the individual ensemble members using a weighted sum rule defined as:

$$agg_{sum}(x) = arg \max_{y \in \mathcal{Y}} \sum_{h \in \mathcal{H}} w_h * p(y|x, h) \tag{4.9}$$

where $p(y|x, h)$ is the probability of outcome y given x predicted by hypothesis h.

This definition can be generalized to other rules such as the average, product, maximum, minimum, median, etc.

4.2.1.2 Feature-based Explanations

Research on improving the interpretability of black-box machine learning models through post-hoc explanations has attracted considerable attention over the last few years. Feature-based explanations are popular among practitioners who want to understand their model better to ensure its adequate behavior when deploying in real-world applications. These techniques often involve visualizing the importance ranking of each feature and how the feature values affect the model's prediction. Formally, feature-based methods assign a scalar attribution value, sometimes called "relevance" or "contribution" to each input sample's input feature. The goal is to determine the contribution $\boldsymbol{R} = [R_1, R_2, \ldots, R_d] \in \mathbb{R}^d$ of each input feature x_i to the output $\hat{y}$.

Several approaches are available to measure each feature's relevance across different data types. Gradient-based methods compute the gradient of the model's output to its inputs and use those values to represent in salience maps [119]. Perturbation-based methods modify or remove parts of the input and measure the impact on the model's output [120]. In the previous setting, Local Interpretable Model-Agnostic Explanations (LIME) is a well-known technique that approximates the model locally with a simpler surrogate model from which several perturbations are performed [121]. Another well-known method is SHapley Additive exPlanations (SHAP), which translates the Shapley values from cooperative game theory into the context of machine learning [122].

The evaluation of the quality of explanations is a non-trivial task due to its subjective nature. Standardizing such evaluation is an open research topic that has received significant attention [123–125]. Insights from social sciences point out that interpretability has properties of clarity and parsimony. Clarity implies that the explanation is unambiguous, while parsimony means that the explanation is presented in a simple and compact form [126]. Lombrozo [111] argues that good explanations are simple and broad. The author's observation is supported by user research studies involving academics across several disciplines that identify consistency in their judgment regarding what constitutes good explanations. Appeals to simplicity were ubiquitous, and some participants also emphasized the significance of generality or comprehensiveness.

In the literature, model complexity is often used as a proxy for explainability complexity. In addition to the number of features, some model-specific metrics are used, such as the number of decision tree rules, tree depth, and the number of non-zeros coefficients in linear models [127–129]. This approach assumes that the more parameters a model has, the more complex it is. Another popular approach is to use information criteria such as the Akaike information criterion or the Bayesian information criterion [130]. These criteria provide means to compare the relative complexity of different models while also considering their goodness of fit. L1 regularization (Lasso) is a popular method for feature selection, as it shrinks the less important feature's coefficient to zero, thus, removing some features altogether. Prior studies have employed a technique of modifying the method by increasing the emphasis on minimizing particular modalities by assigning differential weights to features based on their modalities or aggregating features belonging to the same modality and subjecting the group to a penalty [131–134].

Bhatt et al. introduced in [135] three criteria for evaluating feature-based explanations: sensitivity, faithfulness, and complexity. Sensitivity measures the change in the explanation after perturbation in the model input; faithfulness concerns the capacity of an explanation method to select the truly relevant features; and complexity concerns the extent of the simplicity of the explanation. The authors present a desideratum for good explanations: low sensitivity, high faithfulness, and low complexity. Batterman and Rice studied the complexity of explanations in [136] and argued for minimal model explanations that contain only relevant and representative features. In fact, humans cannot process a high volume of information at once, therefore, explanations should have reduced complexity (i.e., use few features). In order to examine the influence of explanation intricacy on users' understanding, Lage et al. [137] investigated the impact of explanation length and complexity on response time, accuracy, and subjective satisfaction of users. Their findings indicated that heightened explanation complexity led to a decrease in subjective user satisfaction.

4.2.2 Proposed approach

This section describes the methods of the main contributions of this research work. The section starts with a description of our proposed approach to measure the complexity explanation of a single instance, i.e., local explanation, using the feature importance. Then we describe the proposed aggregation strategies based on uncertainty and finish with a summary of the proposed approach.

4.2.2.1 Measuring explanation complexity from feature importance

The rationale for our approach resides in the considerations associated with multimodal problems. In this type of problem, features are associated with different modalities. We argue that a simpler explanation uses the smallest possible set of modalities and features. Therefore, let us consider a dataset D composed of a set of R features and M disjoint modalities. Let us consider that to explain the instance x_i, only a subset of $r \subseteq R$ features and $m \subseteq M$ models are required. We define complexity as the fraction of features and modalities required to explain a given instance in relation to the total number of features and modalities.

Given an explanation function $\boldsymbol{g}$ that depends on the subset of r features and the subset of m modalities to explain the instance x_i, the complexity of $\boldsymbol{g}(r,m)$ at x_i is:

$$c(\boldsymbol{g};x_i) = \frac{1}{2}\left(\frac{|r|}{|R|} + \frac{|m|}{|M|}\right). \tag{4.10}$$

A complex explanation uses $|R|$ features and $|M|$ modalities in its explanation. Although the explanation is probably faithful to the model, it is difficult for the user to understand the relationships between the high number of features and different modalities contributing to a given prediction.

To find the subset of r features, different feature importance methods can be used. In this work, we chose to do it using SHAP. The Shapley values are a solution concept in cooperative game theory. They denote the marginal contribution of a player to the payoff of a coalitional game. Let T be a set of players and let $v : 2^T \rightarrow \mathbb{R}$ be the characteristic function, where $v(S)$ denotes the contribution of the players in $S \subseteq T$. The Shapley value of player j's contribution (i.e., averaging player j's marginal contributions to all possible subsets S) is:

$$\phi_j(v) = \frac{1}{|T|} \sum_{S \subseteq T \setminus \{j\}} \binom{|T|-1}{|S|}^{-1} (v(S \cup \{j\}) - v(S)). \tag{4.11}$$

In feature-based explanations, the problem is formulated similarly, and the game's payoff is the model's output $\hat{y} = \boldsymbol{f}(\boldsymbol{x})$, the players are the d features of $\boldsymbol{x}$, and the ϕ_j represent the contribution of x_i to the game $\boldsymbol{f}(\boldsymbol{x})$.

SHAP calculates Shapley value explanations with an additive feature attribution method:

$$g(z') = \phi_0 + \sum_{j=1}^{Z} \phi_j z'_j \tag{4.12}$$

where $\boldsymbol{g}$ is the explanation model, $z' \in \{0,1\}^Z$ is the coalition vector, Z is the maximum coalition size, and $\phi_j \in \mathbb{R}$ is the feature importance for feature j, i.e., the Shapley value.

Thus, to obtain the r subset, suppose the SHAP values are rescaled from the log odds to the probability space. In that case, their sum equals the difference between the posterior probability, $p(y|x)$, and the expected base value $\mathbb{E}[f(x)]$. The base value is generally the average of the outcome variable in the training set. We define the minimum number of relevant features, as the cardinality of the minimum subset from which the sum of Shapley values, ordered by their absolute value, $\bar{\phi_j}$, equals or surpasses the difference between the posterior probability and the base value.

Given the set of R features, and their Shapley values ordered by their absolute value, $\bar{\phi_j}$, the minimum relevant feature subset, r, to explain the instance x_i, is the one that satisfies the following:

$$\min \left\{ r \subseteq R : \sum_{j=1}^{d} \bar{\phi_j} \right\} \geq \left| p(y|x) - \mathbb{E}[f(x)] \right| . \tag{4.13}$$

To approximate a global explanation from local explanations, usually, all the local explanations are aggregated. Therefore, the global explanation complexity can be defined as the combination of local explanation complexities.

Given the set of c explanation complexities of N samples, the global explanation complexity of $\boldsymbol{g}$ is:

$$\mathbb{C}(c) = \frac{1}{N} \sum_{i=1}^{N} c_i \, . \tag{4.14}$$

4.2.2.2 Uncertainty-weighted model combination strategies

We propose using both soft and hard aggregation strategies, as described in Section 4.2.1.1, for the task of model combination, weighted at two main levels:

- **Model-based**: The predictions of each individual model are weighted by the model's classification performance. This approach is grounded in the principle that the more accurate models should be given greater weight than the less accurate ones, as they are more likely to provide reliable predictions.
- **Instance-based**: The predictions of each individual model are dynamically weighted using measures of uncertainty (see Chapter 3) for a given instance. Dynamic weighting enables the ensemble model to consider the confidence level in each individual model prediction. Although one model may generally be more accurate than the others, it does not necessarily mean it will always be more certain in its decisions. In some cases, the most accurate model may exhibit higher uncertainty in its predictions than the less accurate model.

Although all uncertainty measures can be applied to soft and hard aggregation strategies, aleatoric uncertainty measures will be only used as weights for hard voting methods. Since aleatoric measures are derived from class probabilities and soft voting methods rely on these probabilities, we have chosen to apply only epistemic uncertainty measures to these methods.

Additionally, we proposed a combination strategy based on classification with a rejection option for individual models. For instance-based weights, a model with high uncertainty in a given observation will correspond to a low weight in the combination process. Therefore, its contribution to the decision process will be minimal.

However, in the case of model-based weights, the abstaining capabilities of individual models can be advantageous for the combination process. Thus, we proposed to use both aleatoric and epistemic uncertainty as rejection measures for hard voting strategies and only epistemic uncertainty for soft voting.

For the rejection threshold, we have established the rejection threshold by considering a percentage of the maximum theoretical value for each uncertainty measure. Since all the uncertainty measures used in this study are upper and lower bounded, we have computed the maximum theoretical value for each measure and set the rejection threshold at 90% of that value.

Table 4.4 summarizes the combination strategies that will be studied in the experimental analysis.

Table 4.4: Summary of combination strategies weighted by uncertainty measures. AU: Aleatoric Uncertainty, EU: Epistemic Uncertainty, TU: Total Uncertainty, MR: Model Reliability. $*$ combination of other aleatoric and epistemic uncertainty measures.

Aggregation	**Uncertainty Source**	**Metric**	**Abbreviation**
Hard Voting	Aleatoric	$p(\hat{y}\|x)$ $H[p(y\|x)]$ $u_a(x)$	AU_{max} AU_{entropy} AU_{bayes}
	Epistemic	$vr(x)$ $u_e(x)$	EU_{vr} EU_{bayes}
	Total	$*$	TU_*
	Reliability	$F1$-score $F1$-score w/ aleatoric and epistemic rejection	MR MR_{rej}
Soft Voting	Epistemic	$vr(x)$ $u_e(x)$	EU_{vr} EU_{bayes}
	Reliability	$F1$-score $F1$-score w/ epistemic rejection	MR MR_{rej}

4.2.2.3 Lowering explanation complexity via model combination using uncertainty

We hypothesize that reducing the number of modalities and features used in the explanation can simplify the model's complexity. When modeling multimodal data, one can either use a single model for all available features and modalities or use separate models for each modality and combine their strengths and consider them using the combination strategies previously defined in Section 4.2.1.1.

For the model aggregation strategy, while the individual models might produce accurate predictions in general, in certain circumstances, this may not be the case. For example, there could be certain regions of feature space where some models referring to some modalities struggle to differentiate among the different classes. Using the combination strategy with a rejecting option proposed in Section 4.2.2.2, these models would abstain from making a prediction when they are likely to misclassify. Another advantage of this approach is that since the model rejects uncertainty modalities, it decreases the number of modalities and features required to explain a particular instance, thus lowering the complexity of the explanation.

For each instance, we decrease the number of modalities and features to explain by: (1) rejecting the models with high uncertainty and (2) using only the individual models that are in agreement.

To determine the subset r for the proposed approach, an output score must be calculated. The computation varies depending on the aggregation strategy employed. In hard voting strategies, the class probability is determined by the fraction of votes of each modality, whereas in soft voting strategies, the class probabilities are computed as the mean predicted class probabilities of each model in the ensemble.

In the following sections, we show that this approach reduces the explainability complexity without compromising the overall model performance.

4.2.3 Datasets

We tested our proposed approach in two public datasets composed of multimodal physiological data: WESAD [138] and CSL-SHARE [139].

The WESAD (Wearable Stress and Affect Detection) dataset was introduced and made publicly available by Schmidt et al. [138]. Their study aimed to elicit different affective states in the participants. A total of 15 participants experienced three different affective states, namely baseline, amused, and stressed conditions. This multimodal dataset contains motion and physiological data collected by equipment placed on participants' wrists and chests. For our study, only chest sensors were used, which include a 3-axis accelerometer sensor (ACC) and physiological data from electrocardiogram (ECG), electrodermal activity (EDA), skin temperature (TEMP), electrocardiography (EMG) and respiration (RESP). The data were collected at a sampling rate of 700 Hz.

The CSL-SHARE (Cognitive Systems Lab Sensor-based Human Activity REcordings) dataset contains 22 classes of activities of daily living and sports collected from 20 subjects and was made publicly available by Liu et al. [139]. Participants wore a knee bandage with two triaxial accelerometers (ACC), two triaxial gyroscope sensors (GYRO), four surface electromyography (sEMG) sensors, one biaxial electrogoniometer (GONIO), and one airborne microphone (MIC). The data were collected at a sampling rate of 1000 Hz.

4.2.4 Experimental setup

Figure 4.9 shows the pipeline employed in this study. Each individual modality is first trained separately, and a baseline model using all modalities is also trained for comparison purposes. Data preprocessing varies for each modality but typically includes methods such as signal filtering for noise reduction, as well as transformation methods (such as transforming the ECG signal to heart rate variability) and normalization to reduce inter-subject variability. For the WESAD dataset, the signals were segmented using a sliding window with a window shift of 6 seconds and a size of 60 seconds. While, for the CSL-SHARE dataset, the provided annotations were used to segment the signals. Feature extraction includes statistical, temporal, spectral, and fractal features that depend on the modality used. Additional details on data preparation, including signal preprocessing and feature extraction, are provided in Appendix B.2.

Regarding model training, we considered five machine learning models: Decision Tree, Random Forest, AdaBoost, Naive Bayes, and Support Vector Machines. We used

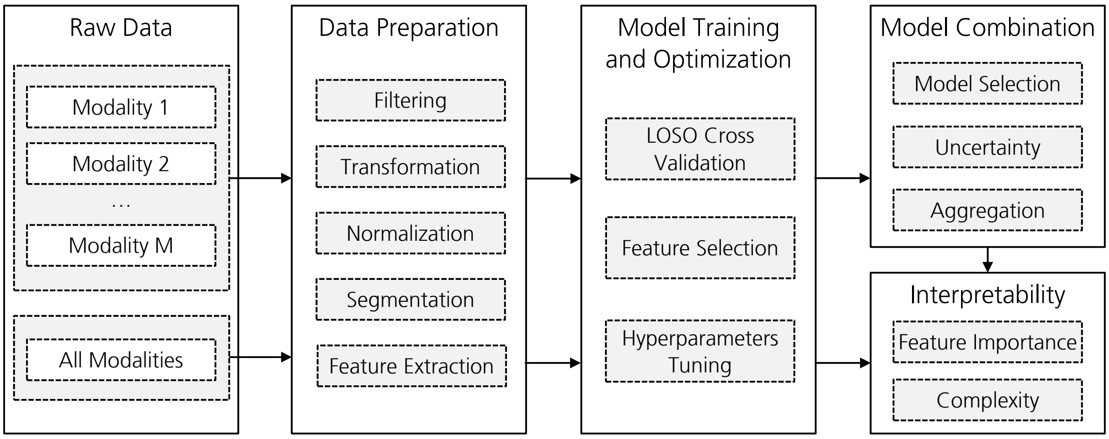


Figure 4.9: Overview of the proposed machine learning pipeline.

a Leave One Subject Out (LOSO) cross-validation approach. A sequential feature selection and a grid search hyperparameter tuning were also applied for model optimization. To evaluate the performance of the classifiers, we used different measures depending on skewed class proportions. For the imbalanced dataset (WESAD), we used F1-score with macro average, while for the balanced dataset (CSL-SHARE), we used accuracy. We selected the classifier with the best performance as the final model for each modality. A comprehensive description of the selected models for each modality, their respective hyperparameters, and the number of selected features is provided in the Appendix (see Tables B.3 and B.4).

In the final step of the pipeline, the best models for each modality are combined using the aggregation methods described in Section 4.2.1.1 weighted by uncertainty measures. The feature importance for each model was measured using SHAP. We used the TreeExplainer [140] for the Decision Tree and Random Forest models and the KernelExplainer for the remaining models with $K = 100$ samples summarized using k-means. The feature importance and the complexity of explanations were measured for the individual models with the proposed combined model and the baseline model.

4.2.5 Results and Discussion

To conduct a preliminary performance evaluation, we assessed the performance of our best models using the pipeline illustrated in Figure 4.9. For the WESAD dataset, we compared our results with the performance reported by Schmidt et al. [138]. The performance comparison is shown in Table 4.5. Since we did not find baseline performance results for the CSL-SHARE dataset, we only present the obtained results in Table 4.6.

Our proposed processing and modeling pipeline yielded improved results compared to Schmidt et al. [138], except for the ECG and RESP. The higher F1-score can be explained by changes we adopted to the original pipeline design. During preprocessing, we

Table 4.5: Performance evaluation of the best models for each modality comparing the results from Schmidt et al. [138] and ours. F1-score with macro average is used as the performance measure. We report the mean and standard deviation of LOSO. The standard deviation of the LOSO was not reported by Schmidt et al. [138]. The best results are highlighted in bold.

Modality	**F1-score**	
	Schmidt et al. [138]	**Ours**
ACC	0.443	**0.671** ± 0.178
EDA	0.483	**0.616** ± 0.195
TEMP	0.425	**0.518** ± 0.188
EMG	0.381	**0.402** ± 0.126
ECG	**0.560**	0.555 ± 0.148
RESP	**0.618**	0.613 ± 0.111
ALL	0.725	**0.777** ± 0.155

Table 4.6: Performance evaluation of the best models for each modality. Accuracy is used as the performance measure. We report the mean and standard deviation of LOSO.

Modality	**Accuracy**
ACC	0.685 ± 0.039
GYRO	0.728 ± 0.043
GONIO	0.650 ± 0.035
EMG	0.375 ± 0.019
MIC	0.257 ± 0.015
ALL	0.835 ± 0.036

normalized feature values for some modalities per subject. This approach aided in reducing model overfitting and eliminating residual bias stemming from differences in basal subject-specific values. It is important to note that there were differences in the feature set used in our study and that of Schmidt et al. [138]. Specifically, we utilized open-source libraries to extract ECG, RESP, and EDA data [141, 142]. Our LOSO's standard deviation was moderate, which reflects some inter-subject variability. Nevertheless, this value was consistent with findings from other studies utilizing the WESAD dataset [143].

4.2.5.1 Aggregation methods

In Table 4.7, we report the results of combining the individual models for each modality using the baseline aggregation methods and our proposed weighted version using uncertainty measures. Note that for soft aggregation strategies, only the best-performing aggregation method was considered for the weighted variant of soft aggregation strategies.

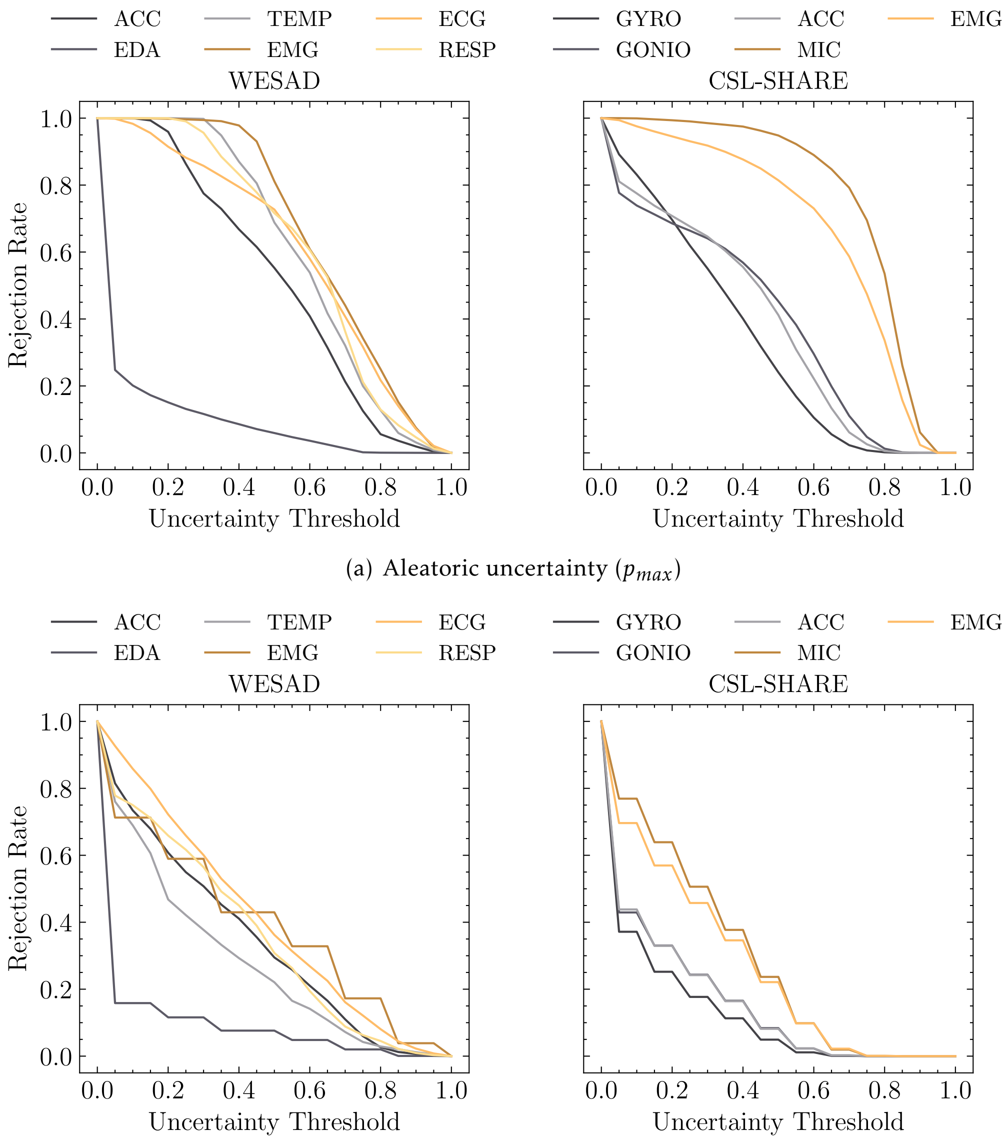
From Table 4.7, we can conclude that majority voting outperformed all equal-weighted

Table 4.7: Performance evaluation of baseline aggregation methods compared with the proposed weighted version using uncertainty measures. The best result per dataset and aggregation strategy is highlighted in bold.

Aggregation	**Weight**	**Performance**	
		WESAD	**CSL-SHARE**
Majority Vote	1	**0.752** ± 0.123	0.755 ± 0.035
Sum Rule	1	0.699 ± 0.137	**0.824** ± 0.032
Mean Rule	1	0.699 ± 0.137	0.824 ± 0.032
Max Rule	1	0.625 ± 0.175	0.783 ± 0.034
Prod Rule	1	0.663 ± 0.211	0.806 ± 0.024
Min Rule	1	0.656 ± 0.209	0.706 ± 0.016
Majority Vote	AU_{max}	0.774 ± 0.130	**0.800** ± 0.030
	AU_{entropy}	0.691 ± 0.146	0.792 ± 0.026
	AU_{bayes}	0.697 ± 0.140	0.791 ± 0.025
	EU_{vr}	**0.781** ± 0.131	0.786 ± 0.031
	EU_{bayes}	0.772 ± 0.128	0.783 ± 0.023
	$TU_{\{AU_{max};EU_{vr}\}}$	0.776 ± 0.132	0.795 ± 0.030
	MR	0.765 ± 0.119	0.787 ± 0.026
	$MR_{\text{-rej}}$	0.766 ± 0.115	0.788 ± 0.026
Sum Rule	EU_{bayes}	0.705 ± 0.156	**0.827** ± 0.032
	EU_{vr}	0.692 ± 0.155	0.823 ± 0.034
	MR	**0.711** ± 0.153	0.827 ± 0.033
	$MR_{\text{-rej}}$	0.710 ± 0.150	0.826 ± 0.033

soft voting strategies on the WESAD dataset, while the opposite was observed for the CSL-SHARE dataset. As a result, the best weighted aggregation strategy for WESAD dataset was majority voting weighted by EU_{vr}, while for the CSL-SHARE dataset, the sum rule weighted by EU_{bayes} obtained the highest score. Overall, the weighted majority voting demonstrated better performance compared to its unweighted counterpart across both datasets, except for AU_{entropy} and AU_{bayes} in the WESAD dataset. Concerning soft voting strategies, while the use of uncertainty-based weighting led to higher scores, the differences compared to the unweighted approach were minimal.

When comparing the performance improvements of aleatoric and epistemic uncertainty as weighted aggregation measures, we observed that epistemic uncertainty had a more pronounced effect on the WESAD dataset, while aleatoric uncertainty yielded higher scores on the CSL-SHARE. Figure 4.10 displays the rejection rate as a function of the dataset uncertainty values, which are normalized by their maximum theoretical value. For instance, when the uncertainty threshold is set to 1, it corresponds to the maximum theoretical value, and as a result, no samples are rejected, leading to a rejection rate of zero. The figure presents the results for both datasets using the best-performing uncertainty measures when applying majority voting, namely p_{max} for aleatoric uncertainty and vr for epistemic uncertainty.



(a) Aleatoric uncertainty (p_{max})

(b) Epistemic uncertainty (vr)

Figure 4.10: Rejection rate as a function of normalized uncertainty values for (a) aleatoric and (b) epistemic uncertainties. The uncertainty threshold is normalized to the maximum theoretical value of each uncertainty measure. Results are shown for both datasets using the best-performing uncertainty measures when applying majority voting, with p_{max} for aleatoric uncertainty and vr for epistemic uncertainty.

Upon evaluating the amounts of epistemic and aleatoric uncertainty in both datasets, we found that CSL-SHARE generally exhibited greater aleatoric uncertainty than WESAD and the opposite for epistemic uncertainty. In fact, CSL-SHARE has two modalities (MIC and EMG) with high aleatoric uncertainty throughout the entire dataset. This observation may have contributed to the obtained results.

Finally, compared to models trained using all modalities, the top-performing weighted aggregation strategy attained comparable performance. It is worth noting that aggregation methods use less information than the models trained with all modalities, as they do not consider the dependence between modalities.

4.2.5.2 Missing data

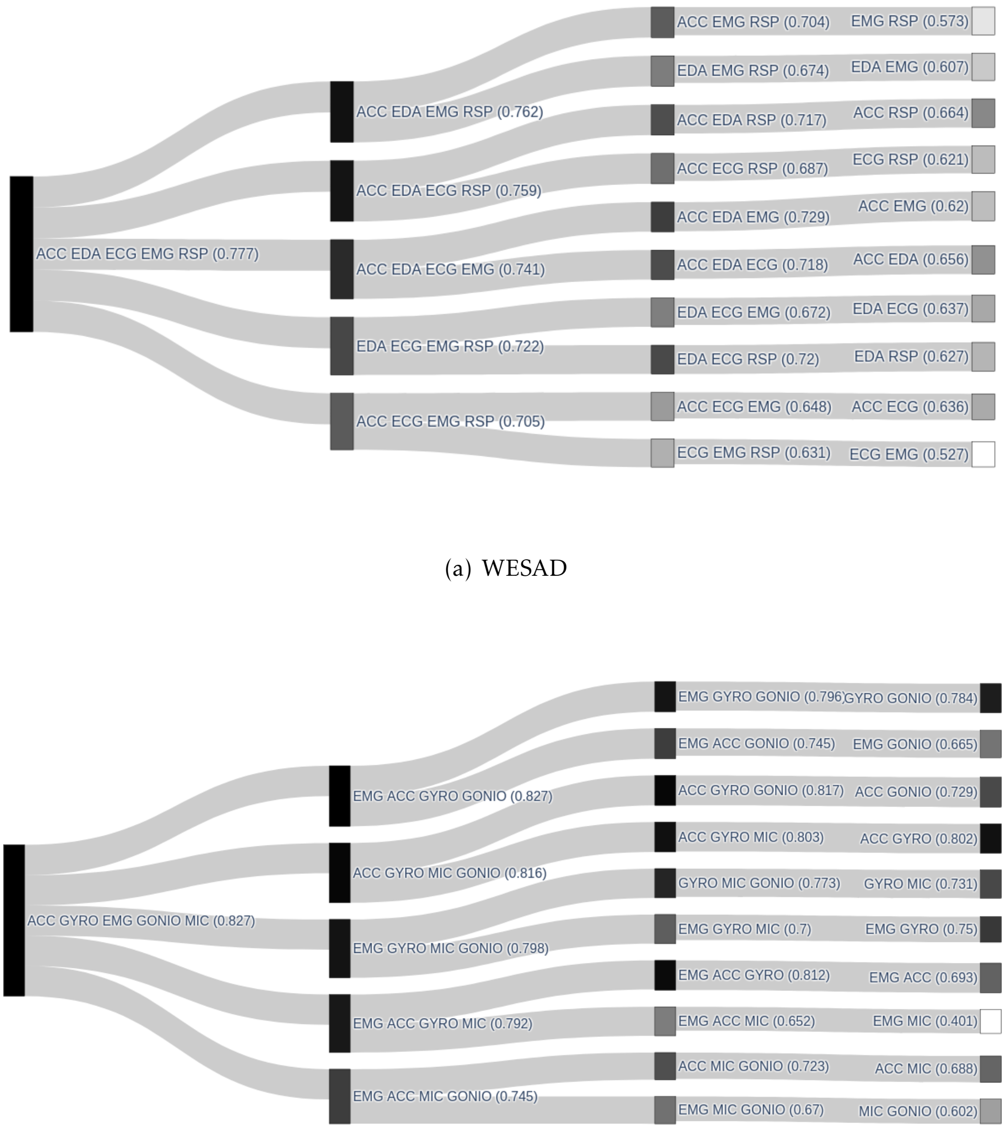
In Figure 4.11, we present an analysis of the model's performance with missing data, focusing on the best-performing model from Table 4.7, i.e., the majority voting weighted by EU_{vr} for WESAD dataset and the sum rule weighted by EU_{bayes} for CSL-SHARE dataset. The diagram in the figure demonstrates the model's performance with varying combinations of modalities, using grayscale to represent the obtained performance (dark for highest and white for lowest). For simplicity and due to the minimal impact of the temperature model on the ensemble performance of the WESAD dataset, the analysis begins with five modalities instead of the available six and concludes with combining two modalities. The drop in performance varies depending on the missing modality. For instance, in the WESAD dataset, the removal of the ECG modality results in a minor performance decrease from 0.777 to 0.762, whereas the removal of the EDA modality leads to a more significant drop from 0.777 to 0.705.

4.2.5.3 Explanation complexity

Figure 4.12 shows the relationship between the mean explanation complexity and mean performance for the models learned on individual modalities, the models learned with all the available modalities (ALL), and the aggregation approaches. Only the best-performing aggregation strategies are presented for the aggregation approaches: non-weighted denoted as BASE_{agg} and weighted denoted as UNC_{agg}.

For both datasets, the models learned on individual modalities exhibited a similar relationship between performance and explanation complexity scores, demonstrating a consistent pattern. As expected, using individual modalities led to lower explanation complexities but with a decreased model performance. The ALL and BASE_{agg} share a similar behavior, with higher performance but increased explanation complexity. In both datasets, the BASE_{agg} attained slightly lower performance than ALL and a slightly lower explanation complexity than ALL for the WESAD and higher complexity in the CSL-SHARE.

The UNC_{agg} model showed an interesting relationship, exhibiting similar performance to ALL but with a lower complexity score with an intermediate value between the individual models and ALL (0.40 for WESAD and 0.53 for CSL-SHARE). This behavior was consistent among the two datasets. Notably, UNC_{agg} achieved slightly higher performance than BASE_{agg}, with a notably lower explanation complexity. The explanation complexities for BASE_{agg} were 0.75 for WESAD and 0.90 for CSL-SHARE. This lower complexity arises from UNC_{agg} using, on average, fewer modalities and features than ALL and BASE_{agg}, as illustrated in Figure 4.13.

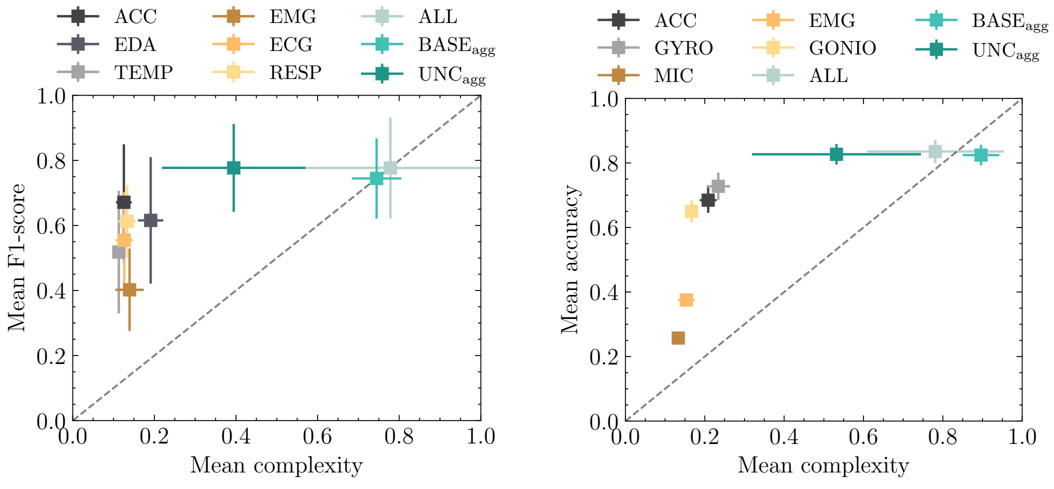


(a) WESAD

(b) CLS-SHARE

Figure 4.11: Diagram illustrating the performance of the best-performing model with uncertainty-based aggregation methods when handling missing modalities. The gray scale indicates the obtained performance for each combination of modalities (black for highest, white for lowest). The diagram starts with five modalities and ends with a combination of two modalities. The performance values for each combination are presented alongside the modalities in the figure.

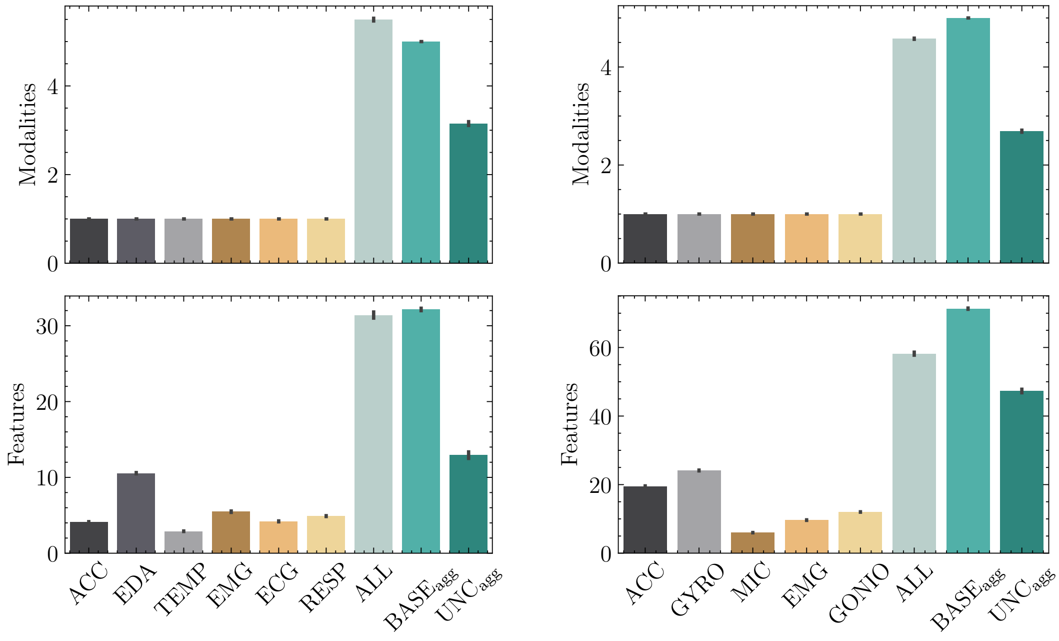
To calculate the number of modalities and features from the minimum relevant features set, we use the threshold defined in Eq. 4.13. We studied the impact of this threshold on the results by analyzing how its value relates to the complexity results. Specifically,



(a) WESAD

(b) CLS

Figure 4.12: Relationship between the mean explanation complexity and mean performance for the individual modalities, ALL, and aggregation approaches. The horizontal and vertical error bars represent the standard deviation across all folds for explanation complexity and performance, respectively.

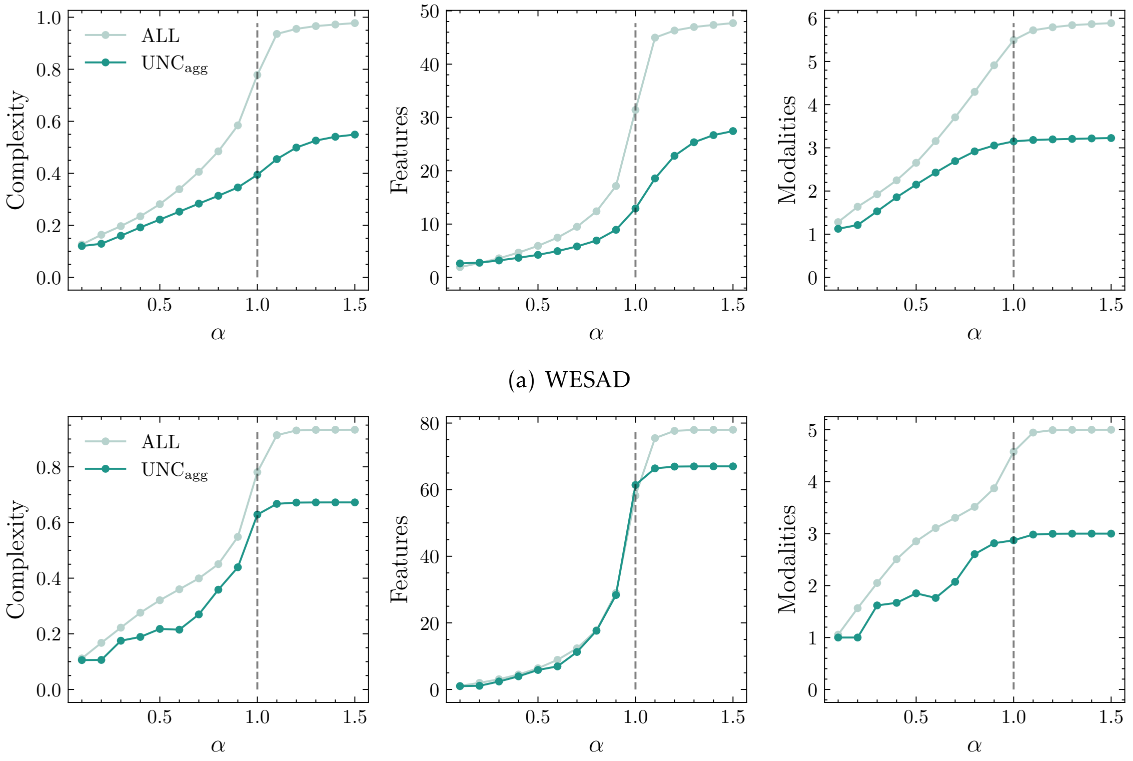


(a) WESAD

(b) CSL-SHARE

Figure 4.13: Number of modalities and features of the minimum relevant feature subset. We report the median number of features and modalities and the error bar with a 95% confidence interval.

we multiplied $\left|p(y|x) - \mathbb{E}[f(x)]\right|$ by the scalar α and calculated the resulting complexity for a range of values of α. The results are presented in Figure 4.14. As expected, lower values of α lower the cardinality of the minimum relevant feature set, thus lowering the explanation complexity. Larger values of α lead to increased complexity. Generally, the growth rate for the number of features, modalities, and complexity stabilizes on a plateau for $\alpha \geq 1$. For the WESAD dataset, the number of features, modalities, and complexity was lower for UNC_{agg} compared to ALL, irrespective of α. Similar behavior was observed in the CSL-SHARE, except for the number of features, which remained approximately equal to ALL until $\alpha \leq 1$.



(a) WESAD

(b) CSL-SHARE

Figure 4.14: The impact of the threshold in the minimum relevant feature size. We multiplied $\left|p(y|x) - \mathbb{E}[f(x)]\right|$ from Eq. 4.13 by the scalar α and calculated the resulting number of features, modalities, and complexity for a range of values of α. Each point represents the mean value across all folds. (a) WESAD dataset and (b) CSL-SHARE dataset.

4.2.6 Conclusion

We propose a novel approach to lower the explanation complexity of feature-based time series models based on reducing the number of modalities and features used to explain multimodal data. Specifically, we use a model aggregation approach weighted by UQ measures that consider the most certain modalities to predict a sample, leading to a lower explanation complexity. We argue that this reduction in explanation complexity yields less complex local explanations without compromising the models' performance.

Using an uncertainty-based model combination is a fundamental aspect of our approach. It allows us to dynamically select the most reliable and certain models for each instance. By incorporating uncertainty quantification measures as weights during the aggregation process, our method can adaptively assess the confidence level of each model's predictions. As a result, our approach outperformed static weighting strategies and yielded better overall performance. The use of confidence or quality measures as weighted metrics for model combinations is supported by other studies in the literature, which have reached similar conclusions on the effectiveness of employing such measures [116].

In the comparison between soft and hard voting strategies, distinct conclusions were reached. Although many weight measures used in majority voting aggregation surpassed the baseline majority voting with equal weights, the same could not be observed for soft voting aggregation. Although soft aggregation methods incorporate more information during the combination process, the predictions generated by different classifiers are only compatible if the output classifier scores represent well-calibrated probabilities. Otherwise, averaging or other combination methods may not yield meaningful results [118]. In future research, we plan to explore the effects of individual model calibration on ensemble performance.

Furthermore, our findings suggest that aleatoric and epistemic uncertainty measures resulted in improved overall performance on datasets containing a higher proportion of aleatoric and epistemic uncertainty samples, respectively. While this conclusion may seem apparent, it emphasizes the importance of integrating both uncertainty sources for a more effective weighted aggregation strategy. Additionally, gaining a deeper understanding of the models during development can enhance the robustness and performance of such models.

The study also highlights the importance of addressing missing data, demonstrating that our approach remains robust and reliable even when faced with incomplete data. This adaptability makes the method suitable for real-world applications where data completeness cannot always be guaranteed.

Regarding the proposed explanation complexity measure, a general assumption in previous literature is that linear models with fewer parameters or rule-based models with few rules are less complex than models with many parameters and rules [144, 145]. Our work extends this previous literature assumption to the multimodal data setting, arguing that more concise explanations have fewer modalities and features. This notion is supported by previous research in social sciences [111, 126], and user research studies focusing on model explainability [137]. In previous studies, metrics for post-hoc interpretability based on the number of features and interactions were proposed [135, 146]. However, these methods are more generalized and do not account for the impact of modalities on the explanation. Therefore, our technical approach differs, and we have introduced a novel measure for assessing model complexity appropriate for multimodal scenarios.

While the current study provides valuable insights and advances, it is important to acknowledge its limitations. This work relies on calculating a representative feature set to reduce the complexity of the explanation. The method to select the representative feature set was based on a heuristic that measured the difference between the posterior and prior probabilities. Some features were ignored since their marginal contribution was deemed minor. As a consequence, the local accuracy of the explanation may have been reduced, but this approach provided a net benefit in terms of lower complexity. Regarding the selected threshold, we conducted an experiment to understand better the impact of the threshold that selects the minimum relevant feature set. In general, irrespective of its value, our proposed approach yields explanations with fewer modalities and lower complexity. The reduction in the number of features was more evident in the WESAD dataset and marginal in the CSL-SHARE. Nevertheless, in the CSL-SHARE dataset, although the number of features from our approach was similar to the model trained with all modalities, the complexity was lower. Explaining an instance with the same number of features but referent to fewer modalities is simpler than with a high number of modalities.

While the current study used a heuristic to select the threshold to determine the representative feature set, it is possible that a user research study could provide further insights into the needs of users and better inform the selection of the optimal feature set.

4.3 Final remarks

As ML models are increasingly being integrated into safety-critical applications, incorporating uncertainty quantification estimates should become a required part of the ML methodology. Uncertainty quantification can be used for "uncertainty-informed" decisions and to support developers and end-users by increasing the interpretability of and trust in model predictions.

In this chapter, we introduced a complete study focused on how uncertainty quantification can be used in practice through three research questions: (1) How can UQ contribute to choosing the most suitable model for a given classification task? (2) Can UQ be used to combine different models in a principled manner? (3) Can UQ be employed to enhance models' interpretability?

Regarding the first question, we showed that uncertainty quantification in combination with the model's accuracy can give us important elements to choose the most suitable model. For instance, the decision between different classifiers with the same accuracy can benefit from the uncertainty quantification methods, whereas classifiers with lower degrees of uncertainty can be preferable. Furthermore, if model uncertainty is high and the addition of new samples is possible, the increase of training samples can reduce the model uncertainty and consequently increase the model's accuracy. By using uncertainty as a complement to performance measures, we can make more informed decisions in model selection.

The combination of multiple models, instead of using a single one, is often touted as a more robust solution. In this sense, uncertainty estimation can play an important role as part of the combination rule. By incorporating uncertainty quantification measures as weights during the aggregation process, the ensemble model can adaptively assess the confidence level of each individual model's predictions. We show empirical evidence that model combination using UQ methods outperforms static weighting strategies and yields better overall performance. Also, an uncertainty-based model combination helps reduce the complexity of explanations provided by feature importance scores. Another advantage of using dynamic weights as the basis for model combination is the ability to handle missing data. In real-world applications, it is not uncommon for some data to be missing or to exhibit poor quality due to various factors, such as sensor failures or noisy environments. This model combination approach offers a significant advantage in such scenarios, as it can still provide reliable predictions even when some modalities are missing, albeit with decreased performance. Therefore, by leveraging the uncertainty-based weighting scheme, the proposed approach can dynamically adjust the weights assigned to the available data. This capability not only increases the robustness of our method but also makes it more suitable for deployment in practical situations where data completeness cannot be guaranteed.

In the third question, we explored visualization techniques to assist in interpreting classifiers' uncertainty during the model's development and also to audit a given decision. Understanding which type of uncertainty is present during the model's development can give us insights into the limitations of each model and allow us to take actions in accordance. In the context of prediction reliability, the proposed visualization techniques were used to assess the interpretability of the rejection option in which a rejection may correspond to a low prediction probability (aleatoric uncertainty), a poor model fit (model uncertainty), or an outlier (knowledge uncertainty). In addition, we propose a novel approach to lower the explanation complexity of feature-based time series models based on reducing the number of modalities and features used to explain multimodal data. Specifically, we use the model aggregation approach weighted by UQ measures that consider the most certain modalities to predict a sample, leading to a lower explanation complexity. We argue that this reduction in explanation complexity yields less complex local explanations without compromising the models' performance. This conclusion is supported by a general assumption in previous literature that mentions that linear models with fewer parameters or rule-based models with few rules are less complex than models with many parameters and rules [144, 145]

In a broader view of research on machine learning, this study also introduces an innovative example of how we can leverage the intersection between the disciplines of uncertainty quantification and model explainability towards the topic of responsible AI Our main motivation with this chapter is to spark future research on how to consider uncertainty quantification as a tool to improve the ML model development lifecycle.

5 Uncertainty for Clinical Decision Making

Machine learning has made significant progress in a variety of decision-critical domains, including medicine. However, as these advancements are applied in real-world safety-critical applications, it is crucial to consider the inherent uncertainty present in the ML process as a path toward trustworthy AI [5]. While AI research has achieved promising results across various domains, the adoption of AI in the medical field remains a challenge [7]. This can be attributed to various factors, including the lack of trust in AI decisions. In medical AI, it is essential to have the ability to abstain from providing a decision when there is a high level of uncertainty associated with it. This mirrors the clinical practice of seeking a second opinion in unusual or complex cases. However, the quantification and communication of uncertainty are not routinely addressed in the current literature, yet they are crucial in healthcare applications [1]. Therefore, the development of a systematic and formal discipline for UQ in AI-based approaches is vital for machine-assisted medical decision-making.

In this chapter, we address the use of UQ for decision-making using ECG classification as a domain example. We choose to center our analysis on ECG classification due to the following reasons:

- There are several (and large) publicly available datasets from different populations, including different countries, different recording devices, and different time intervals;

- Research studies on ECG classification have only been focused on classification performance neglecting the robustness of models' results, which are a key element for their implementation in clinical practice;

- Uncertainty quantification has only been studied in single-label tasks, however, ECG classification (and many other medical tasks) should be treated as multi-label tasks since it is likely that a patient has more than one cardiac pathology at the same time;

- Publicly available datasets contain more than 100 diagnoses classes, which allows studying the robustness of UQ to handle unknown medical conditions (commonly referred to as OOD samples in literature) and data containing mixtures of known and unknown medical conditions (unexplored in literature);

This chapter is divided into five sections, beginning with important background for the experimental analysis, including multi-label classification, dataset shift, active learning, and the current research in the ECG classification domain. Then, the methods used and experimental results are presented. The chapter concludes with a discussion of the results obtained and final remarks.

5.1 Preliminaries

5.1.1 Multi-label classification

Multi-label classification is a classification task where multiple nonexclusive labels may be assigned to each instance, as opposed to multi-class or binary classification where a single label is assigned to each instance.

Formally, multi-label classification is the task of finding a hypothesis h that maps input instance x to binary output vector y. Considering a dataset $D = \{(x_i, y_i)\}_i^N \subset \mathcal{X} \times \mathcal{Y}$ with N samples, $\mathcal{X}$ is an instance space and $\mathcal{Y} \in \{0,1\}^K$ a set of K-dimensional binary vectors with $y = (y^1, \dots, y^K)$ called a multi-label and the components y^k micro-label [147]. For a pair (x_i, y_i) the micro-label $y_i^k = 1$ means that the k^{th} class is assigned to the i^{th} example [148].

Multi-label classification approaches can be grouped into two main categories: *data transformation* and *method adaptation*. *Data transformation* approaches solve the multi-label tasks by transforming them into multiple single-class problems. Popular transformation approaches are *Binary Relevance* that convert a multi-label sample into several single-label samples and *Label Powerset* that considers every label combination as a separate class in a multi-class classification problem [149]. *Method adaptation* approach focuses on changing existing single-class classification algorithms to solve multi-label cases. For example, a multi-label version of KNN algorithm, named *ML-kNN*, uses the KNN for each of the C labels independently [149]. In neural networks, by applying the sigmoid function as an activation function to the output layer with C neurons, the algorithm is transformed into a binary classification for each class [147].

The applications of multi-label include many real work problems such as text classification, music information retrieval, image classification, and time series analysis problems (such as ECG classification). Although the applications are vast, the multi-label studies that include UQ in their analysis are mainly associated with image recognition [150] or text classification [151]. Still, even in the mentioned applications, UQ for multi-label classification remains underexplored and uses rudimentary techniques [53].

5.1.2 Dataset shift

Commonly, machine learning relies on the assumption that training and test data are independent and identically distributed. Therefore, good performance on validation data is expected to translate to good performance in deployment. However, in real-world applications, the final application distributions often differ from training data, leading to poor model performance and generalization. In practice, data are subject to a wide range of possible distributional shifts where the greater the degree of shift, the poorer the model's performance [8]. For instance, machine learning models applied in the medical domain are often trained on data from a few hospitals and then deployed broadly to hospitals not included in the training set [9]. Data from different populations or medical equipment can introduce shifts that affect the machine learning models' performance. Assessing a model's robustness to distribution shifts and its ability to estimate predictive uncertainty is crucial for detecting these shifts.

Although there has been significant research done on developing methods for improving the robustness of dataset shift and uncertainty estimation, few studies provide a comprehensive evaluation of uncertainty estimation under dataset shift. However, in this context, Ovadia et al. [152] studied the impact of dataset shift on the accuracy and calibration of deep neural networks. For a diverse classification task tested on benchmark datasets, results revealed that models with the best accuracy and calibration do not necessarily perform well under dataset shift. Regardless of the method, the quality of uncertainty estimation degrades with increasing dataset shift, leading to incorrect predictions with high confidence on out-of-distribution data. On the tested uncertainty methods, the authors concluded that deep ensembles were more robust to dataset shift.

There are limited datasets and benchmarks for evaluating uncertainty estimation and robustness to dataset shift or even OOD. Some datasets, that synthetically added noise, natural adversarial attacks, or unseen classes, exist but are typically limited to image classification or text. The use of data similarity to quantify the degree of OOD or datasets similarity, associated with generalization, is an old and important question in the ML literature, as several ML methods implicitly rely on properties related to similarity (e.g., the large margin assumption in SVM learning) to guarantee good generalization performance [153]. The potential relationship between data similarity and the generalization properties of ML models was first investigated from an empirical point of view in the work of Bousquet et al. [154], where the authors discovered that datasets found to be substantially dissimilar likely stemmed from different distributions. Based on these findings, Kouw et al. [155] demonstrated that information about similarity can be used to understand why a model performs poorly on a validation set, while the same information can be used to understand when and how to successfully perform domain adaptation. To that end, several metrics for measuring data similarity have been proposed in the literature. Bousquet et al. [154] developed a measure (Data Agreement Criterion, DAC) based on the Kullback–Leibler divergence, which has since become frequently used to assess the similarity of distributions. More recently, Cabitza et al. [156] proposed instead a different approach based on a multivariate statistical testing procedure to obtain a hypothesis test for OOD data, the degree of correspondence, and also studied the correlation between degree of correspondence scores and the generalization of ML models. By contrast, in the Deep Learning literature, approaches based on the use of statistical divergence measures, such as the Wasserstein distance [157] or the Maximum Mean Discrepancy (MMD) [158], have become increasingly popular to design methods for OOD detection. See also, the recent review by Shen et al. [159].

5.1.3 Active Learning

Machine learning models need a large amount of labeled data for proper learning. The complexity of the problem or input data increases the amount of labeled data needed. This is particularly true in the medical field. For instance, to automate the analysis of medical exams, a significant amount of exams must be annotated by an expert to indicate the presence of a certain condition. Obtaining this amount of labeled data is time-consuming and costly.

One solution to this challenge is active learning [160], where the model selects the most informative unlabeled data for training and requests the label from an external oracle (e.g. medical expert). The selection of data is made by an acquisition function, which ranks data points based on their informativeness [6]. Many acquisition functions use uncertainty quantification to determine the informativeness of unlabeled data points. The more informative the selected data, the fewer labeled examples are required for improved classifier accuracy. Therefore, uncertainty quantification can play a central role in active learning.

The use of uncertainty quantification in active learning is common, as it has been demonstrated that measures of uncertainty, particularly those that quantify epistemic uncertainty, are effective criteria for model retraining. For example, Gal et al. [21] introduced a Bayesian Convolutional Neural Network (CNN)-based active learning framework that utilized Monte Carlo Dropout for approximate inference, and showed that the Bayesian CNN outperformed a deterministic CNN in selecting data to be labeled. Sadafi et al. [11] also developed an active learning framework that leveraged confidence scores from dropout variation inference to choose the most informative samples for labeling and found that using uncertainty quantification outperformed random sampling. Nguyen et al. [10] investigated the usefulness of distinguishing different sources of uncertainty and compared their performance in active learning. The results indicated that the framework that used epistemic uncertainty performed better than the framework using aleatoric uncertainty

5.1.4 ECG classification

Over the past decade, the automatic interpretation of ECG records has been widely investigated. Automated classification pipelines have been proposed for classifying individual heartbeats [161–163] and longer intervals containing multiple heartbeats [164–166]. While traditional ML models have been successful in classifying some medical conditions [167], Deep Learning (DL) methods have gained increasing attention in recent years, motivated by their superior performance without requiring significant effort in feature engineering [168]. In either case, research on the automatic interpretation of ECG has focused on single-label classification, where each ECG record is assigned to a label from a disjoint set. This approach, however, might be too brittle to answer real-world application scenarios. At the heart of ECG interpretation lies the interest to determine whether the record is normal in terms of wave morphology, intervals, and rhythm. Thus, depending on the case complexity, reference to arrhythmias, myocardial ischemia and infarction, conduction defects, and cardiac hypertrophies might coexist in clinical interpretation reports. The clinical interpretation resembles more closely multi-label classification where nonexclusive labels may be assigned to each record.

Prior studies in ECG classification have often overlooked the evaluation and management of uncertainty associated with their estimations, focusing primarily on classification performance without considering practical implementation in real-world applications. Hong et al. conducted a systematic review of the PhysioNet/CinC Challenge 2020 [169], highlighting the importance of handling unknown classes and interpretability for real-world implementation. Surprisingly, none of the top 10 methods in the Challenge 2020 addressed these critical topics.

While research on UQ for ECG classification remains limited, some recent works have addressed this area, and are summarized in Table 5.1.

Belen et al. [170] employed a variational encoder network to classify atrial fibrillation using the MITBIH Atrial Fibrillation database. Their method used KL Divergence as a loss function and estimated uncertainty by running the input through the network multiple times and computing the standard deviation of softmax probabilities. Vranken et al. [171] explored various uncertainty estimation methods, including Monte Carlo dropout, variational inference, ensemble, and snapshot ensemble. They evaluated the quality of uncertainty estimations using rank-based metrics, calibration evaluation, and OOD detection. Their results showed that variational inference with Bayesian decomposition and ensemble with auxiliary output outperformed other methods in terms of ranking and calibration across datasets and in both in-distribution and OOD settings. Aseeri et al. [172] developed a gated recurrent neural network trained using three types of datasets and estimated uncertainty using Monte Carlo dropout and deep ensemble methods. They also evaluated the uncertainty calibration of these methods and demonstrated that their proposed network achieved comparable results with state-of-the-art methods while having a strong capability of rejecting low-confidence examples. Elul et al. [173] presented a

Table 5.1: A summary of related studies on ECG classification using uncertainty quantification measures.

Study	Data	Labels	External Validation	OOD	Calibration
Belen et al. [170] (2020)	MITBIH AF	Single	No	No	No
Vranken et al. [171] (2021)	UMCU-Triage UMCU-Diagnose CPSC2018	Single	No	Yes	Yes
Asseri et al. [172] (2021)	MITBIH ARR INCART BIDMC	Single	No	No	Yes
Elul et al. [173] (2021)	MITBIH NSR Long-Term AF MITBIH ARR MITBIH AF THEW CinC 2017	Multi	Yes	Yes	No
Zhang et al. [174] (2022)	CPSC2018	Single	No	No	No
Jahmunah et al. [175] (2023)	PTB-XL	Single	No	No	No
Park et al. [176] (2023)	MITBIH ARR CinC 2017 INCART	Single	No	No	No

comprehensive study on integrating AI into clinical practice, emphasizing the importance of uncertainty estimation for handling OOD examples or multilabel diagnosis. They developed a DL model consisting of 10 binary classifiers for each trained ECG pathology, enabling the model to output any combination of known rhythms and handle unknown classes when the model outputs a negative prediction for every binary class. They employed the Monte Carlo dropout method to assess the confidence in predictions. Zhang et al. [174] employed a Bayesian neural network with Monte Carlo dropout for arrhythmia classification with a rejection option. They computed total uncertainty using an entropy-based decomposition of data and model uncertainty and explored different uncertainty thresholds to improve classification performance by rejecting high uncertainty samples. Jahmunah et al. [175] trained a Dirichlet DenseNet with reverse KL divergence to compute predictive entropy for model uncertainty in a multi-class classification task. The authors argue that their approach is faster and computationally lightweight compared to previous uncertainty quantification methods. Additionally, they included noisy ECG in their analysis. Recently, Park et al. [176] proposed a self-attention-based LSTM-FCN deep learning architecture using a deep ensemble approach to quantify uncertainty. Their results achieved state-of-the-art performance, showing that epistemic uncertainty is reliable for classifying the six arrhythmia types.

Even though some multi-label datasets were used in the previously presented studies, all of them employed a single-label classification approach, except for Elul et al. [173]. To the best of our knowledge, Elul et al.'s work [173] is the only one that applied an UQ method under the multi-label approach in ECG classification. While this study offers a comprehensive interpretation of the importance of handling a mixture of classes and demonstrates that their model is prepared to deal with the multi-label setting, no performance evaluation was conducted on multi-label datasets, making it difficult to thoroughly assess the performance of their model in such settings. Additionally, in this study, only the Monte Carlo Dropout method was used as the uncertainty quantification method.

Additionally, some studies focus on calibration metrics, others on OOD detection, and a few on external validation. However, we argue that a good uncertainty quantification measure should comply with all three validation procedures. In this sense, we focus our work on multi-label datasets, evaluating not only internal validation sets but also external sets, OOD, and calibration.

5.1.5 Summary

There is currently a gap in the literature considering multi-label and UQ for automated ECG interpretation. Both topics are crucial for successfully implementing automated decision support systems in real clinical practice. The scarcity of studies UQ in the context of multi-label classification goes beyond the applicability to ECG interpretation since there has been limited attention on both topics.

In this chapter, we provide an extensive comparison of several UQ methods in the multi-label classification approach. We choose to center our systematic analysis on the ECG classification since there are publicly available large multi-label datasets. The large volume of available datasets allows for studying the robustness of ML/DL models on external validation sets, including data with unknown medical conditions and heterogeneous mixtures of conditions. Furthermore, the integration of UQ and multi-label techniques might help translate decision support systems for ECG interpretation into clinical practice. In this sense, besides the comparison of UQ methods, we include in our work a clinical simulation scenario to assess the benefit of integrating AI uncertainty estimation methods into the practice of cardiology and provide an illustration in Figure 5.1.

The proposed system is based on an uncertainty-aware AI model, trained to detect cardiac pathologies based on 12-lead ECG signals. To prevent incorrect ECG interpretations, ECG data is quality checked before being used for diagnostic classification. In addition to the classification of cardiac pathologies, the model provides its overall confidence in

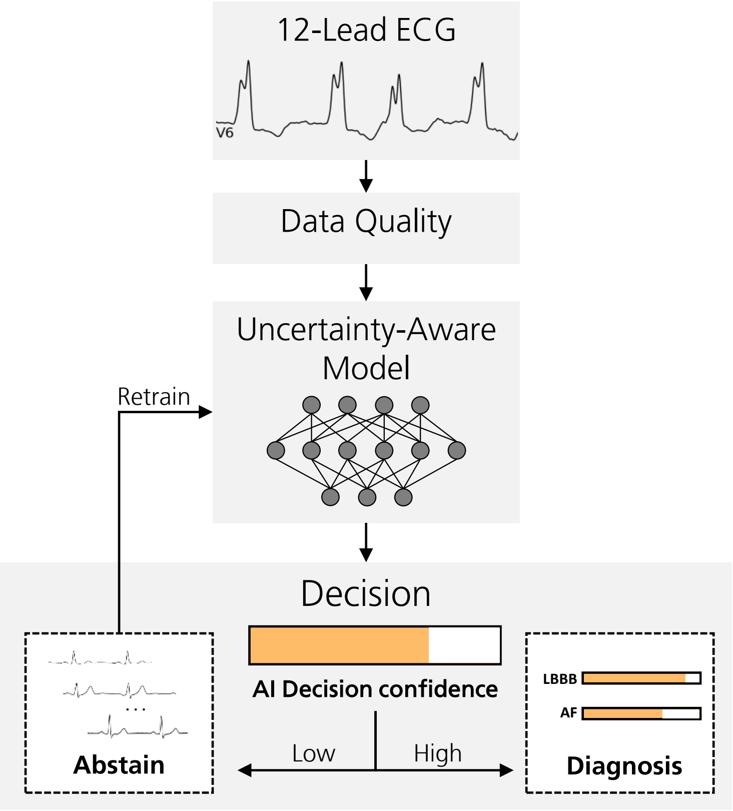


Figure 5.1: Workflow for a clinical decision support system (CDSS) that includes uncertainty quantification techniques. The process begins with a data quality assessment, followed by the machine learning model generating a prediction or diagnosis and estimating the associated uncertainty. The incorporation of UQ allows clinicians to make more informed decisions. Rejected predictions are used to retrain the model using an active learning workflow, improving the overall accuracy and reliability of the CDSS.

predicting a given sample which is used to abstain from providing a diagnosis when there is a large amount of uncertainty. In the case of a prediction with low uncertainty, an independent confidence score is provided for each predicted diagnosis. With this ability, additional human expertise can be sought on those rejected samples that later can be used to retrain the model, improving its performance capabilities. Detecting dataset shifts and continuous training after a model is deployed is of high importance since the environment is continuously changing, and concept drifts are likely to occur. In this scenario, and due to the cost associated with data labeling, uncertainty estimation plays an essential role in selecting the most informative samples to be labeled.

Contributions. We present a comprehensive comparison of UQ methods in a multi-label setting, focusing on ECG classification scenarios. Our evaluation of UQ methods across various validation scenarios highlights the importance of external validation and its influence on performance, the quality of uncertainty estimates, and calibration. Furthermore, we provide empirical evidence that incorporating UQ throughout the machine learning pipeline brings advantages in classification with a rejection option, dataset shift detection, and active learning.

In summary, this research work aims to address the following research questions that will be explored in the experimental results section.

- Is the performance of internal validation consistently reproduced on external validation? (Section 5.3.1: **External validation**)
- How does external validation affect the calibration of models' predictions? (Section 5.3.2: **Calibration**)
- How reliable are uncertainty methods in a multi-label setting under different validation strategies? (Section 5.3.3: **Uncertainty quantification**)
- What is the impact of using sample rejection on ECG classification performance? (Section 5.3.4: **Classification with rejection-option**)
- How does the presence of low-quality data impact the performance of uncertainty-aware models? Is there a correlation between low-quality data and high uncertainty estimates provided by these models? (Section 5.3.5: **Data quality**)
- Are uncertainty measures suitable as selection criteria for active learning? (Section 5.3.6: **Active learning**)

5.2 Methods

We conducted a systematic analysis of various uncertainty quantification methods, following the steps illustrated in Figure 5.2 and dividing this section accordingly. We begin by discussing the datasets employed and the considerations for data preprocessing and data quality assessment. Subsequently, we provide details on the neural network architecture and its variations for uncertainty estimation. The section concludes with an explanation of the validation, which involved three distinct sets (internal, external, and OOD) to assess the methods, the implemented evaluation measures, and particular applications of uncertainty methods (classification with rejection option, dataset shift and active learning).

5.2.1 ECG data preparation

An ECG sensor measures the electrical activity of the heart using skin-placed electrodes and is a noninvasive tool for diagnosing heart problems such as arrhythmias. However, accurate detection of heart problems depends on a clean ECG reading, but the noise from internal or external sources can impact the reading and lead to false interpretations. Therefore, ECG preprocessing is essential to eliminate noise and improve interpretability before ECG classification.

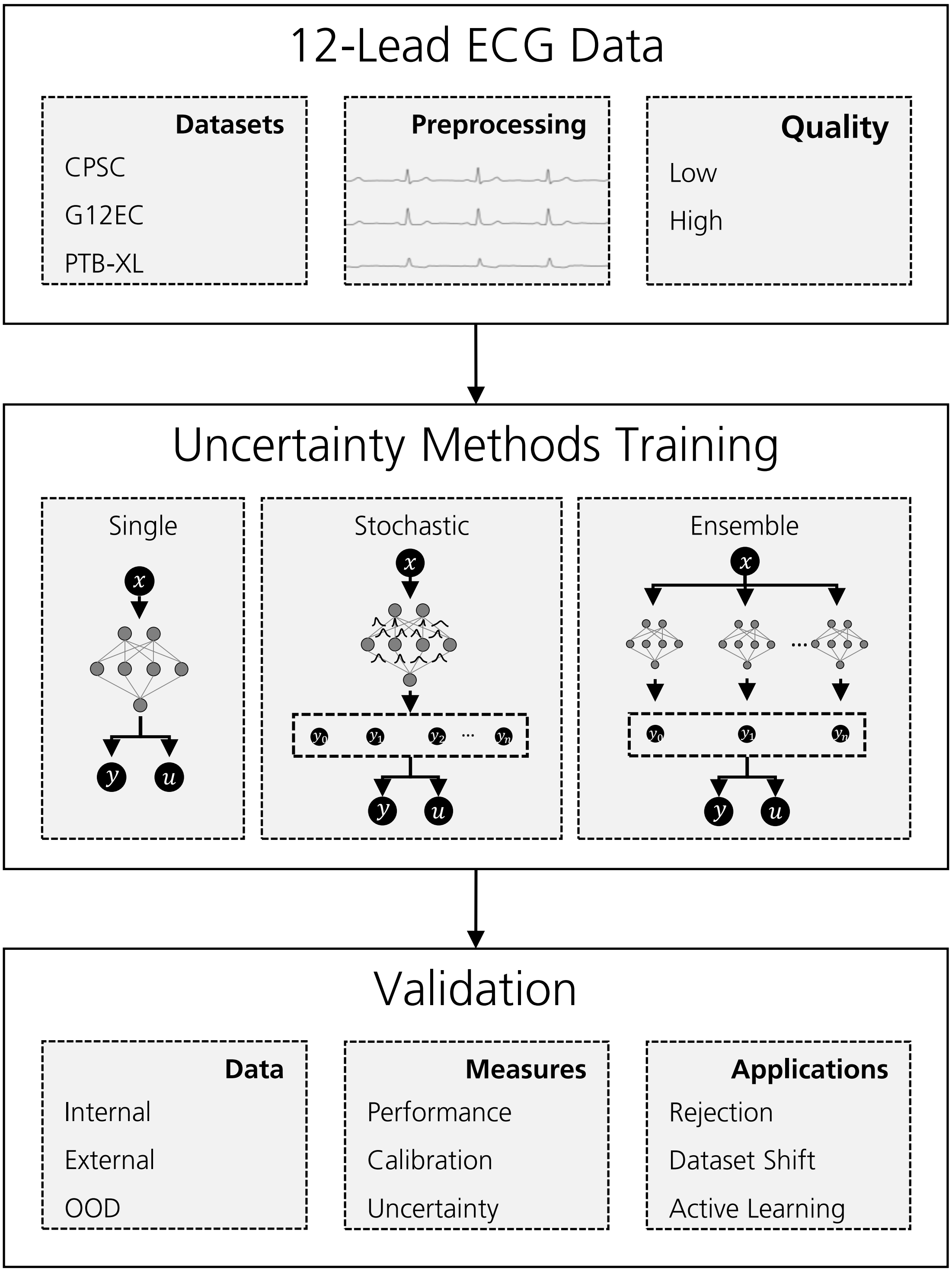


Figure 5.2: Overview of the methodology used for uncertainty methods evaluation. Data is based on 12-lead ECG signals, and uncertainty methods are divided into three main categories: Single, Bayesian, and Ensemble. Validation is done using three test sets (internal, external, and OOD) evaluated in terms of performance, calibration, and uncertainty measures.

5.2.1.1 Datasets

For dataset selection, our primary criterion was to choose datasets that included 12-lead ECG data. The PhysioNet/CinC Challenge 2020 provided 12-lead multi-label ECG datasets from four different data sources. However, due to our validation procedure, which involved internal and external validation using different data sources, we could not use the standard 27 classes (out of 111 classes) selected by PhysioNet/CinC Challenge 2020, as not all classes were present in every dataset.

As a result, we decided to utilize only the classes that were common among the datasets. This approach yielded nine classes (NSR, AF, I-AVB, LBBB, RBBB, PAC, VEB, STD, and STE) that were represented across the entire CPSC dataset, enabling us to conduct consistent validation across the different datasets. Furthermore, these nine classes are available in three different data sources. The first source is the China Physiological Signal Challenge 2018 (CPSC) [177], the second is the Physikalisch Technische Bundesanstalt XL (PTB-XL) [178] from Brunswick, Germany, and the third is the Georgia 12-lead ECG Challenge (G12EC) [179] Database, Emory University, Atlanta, Georgia, USA. The three datasets contain data from the 12-leads ECG signals, demographic information (age and gender), and multi-label annotations. The annotations between databases were previously standardized by PhysioNet/CinC Challenge 2020. However, following the evaluation procedure of PhysioNet/CinC Challenge 2020, we relabeled the class CRBBB in G12EC and PTB-XL dataset to RBBB.

5.2.1.2 Preprocessing

As we are using different datasets with different characteristics, the data preprocessing includes the following steps: resampling, window function, filtering, and normalization.

To improve the computational efficiency of the experiments, the ECG signals were downsampled from 500 Hz to 250 Hz. A 10-second window size was selected as the standard 12-lead ECG is a 10-second strip. For ECG signals longer than 10 seconds, the 10 seconds in the center were selected. This choice was made due to poor signal quality at the beginning and end of some ECG signals. ECG signals shorter than 10 seconds were discarded as they were few in number (25 signals in the G12EC dataset and 6 in the CPSC dataset). Additionally, the ECG signals were filtered using a 2nd order band-pass Butterworth filter between 1 and 40 Hz and normalized through a z-normalization over the complete training dataset.

In Table 5.2, a summary of ECG data used per class and dataset is presented. The Table contains information about the number of labels and recordings per dataset.

5.2.1.3 Quality assessment

As previously stated ECG recordings are often corrupted by the noise that resembles ECG waveforms. To prevent incorrect ECG interpretations, ECG data should be quality

Table 5.2: Overview of multi-label datasets statistics.

Class	**CPSC**	**G12EC**	**PTB-XL**	**Total**
AF	1220	568	1514	3302
I-AVB	722	766	795	2283
LBBB	235	231	536	1002
NSR	918	1735	18058	20711
PAC	614	636	398	1648
RBBB	1857	554	542	2953
STD	868	38	1009	1915
STE	220	134	28	382
VEB	699	41	1153	1893
# Labels	7353	4703	24033	36089
# Recordings	**6871**	**4301**	**20214**	**31386**

checked before being used for diagnostic purposes. This process should remove low-quality data to be sent for the classification models preventing erroneous diagnosis. ECG quality assessment methods are commonly divided in feature-based and non-feature-based (deep learning approaches) categories [180]. Feature-based methods typically depend on detecting constant signals (e.g. missing lead), low signal-to-noise ratio, or QRS detection. On the other hand, non-feature-based methods, rely on supervised deep learning models with pre-labeled datasets. Autoencoders can also be used in this category, where reconstruction error is used to detect low-quality signals. This approach is broader in scope, as it can detect different signal characteristics beyond low quality, such as a previously unseen pathology.

Uncertainty-aware classification models with a rejection option should theoretically reject low-quality data by outputting predictions with high uncertainty. In this case, the quality assessment before the classification model could be considered neglected. However, current uncertainty quantification methods still exhibit unpredictable behavior in the presence of unknown data. Furthermore, in contexts of active learning, it is crucial to distinguish between low-quality signals and high-quality signals with high uncertainty. For instance, if both low- and high-quality signals have high epistemic uncertainty, selecting high-quality data for model retraining is important, as low-quality data may contain no useful information for learning.

Thus, an exploratory evaluation was performed to assess ECG quality, using both a feature-based approach and an autoencoder. Three common quality check features were used in the feature-based approach: stationary signal, heart rate, and signal-to-noise ratio (as used in [181]). The stationary signal feature detects if an ECG signal is stationary within a predefined time window. The heart rate feature uses the Pan-Tompkins algorithm [182] to identify the QRS complex and count heartbeats. Acceptability is determined based on upper and lower thresholds. The signal-to-noise feature is the ratio of signal spectral power to noise spectral power, with signals having a signal-to-noise ratio below a threshold considered low quality.

For the non-feature-based approach, a one-dimensional CNN-based autoencoder was implemented for each ECG lead. The encoder comprised 5 blocks, each with a 1D CNN layer, batch normalization layer, and ReLU activation function. The filter size was 20 until the third block, then reduced to 10. The decoder was set to appear symmetrically with the encoder.

Since the datasets utilized do not offer information on data quality, an exploratory assessment was conducted using visualization and by evaluating the classification performance of acceptable versus unacceptable signals, labeled through visual inspection. For a quantitative evaluation, we employed the PhysioNet/Computing in Cardiology Challenge 2011 dataset [183]. This dataset comprises quality assessment annotations reviewed by a group of annotators with varying levels of expertise in ECG analysis. The annotations involve classifying a 12-lead ECG recording as acceptable or unacceptable. The dataset encompasses a total of 1000 recordings, including 225 unacceptable and 775 acceptable signals.

5.2.2 Uncertainty methods training

The goal of this chapter is not to develop better models or improve the accuracy of existing methods, but to examine the potential of uncertainty measures as a safety mechanism in practical ECG classification. To that end, we will provide a brief description of the baseline architecture and its application to uncertainty methods.

5.2.2.1 Baseline architecture

As baseline architecture, we decided to use the proposed neural network architecture, which was ranked first in the China Physiological Signal Challenge [166]. The model is a combined architecture of five CNN blocks, followed by a bidirectional gated recurrent unit (GRU), an attention layer, and a finally dense layer. For more details, please refer to Chen et al. [166]. The training was done using the Adam optimizer with a learning rate of 0.001. To counteract class imbalance in the data, the binary focal loss was used as the loss function with the focusing parameter set to 1. The training was performed for 100 epochs using mini-batches of size 64. The best model, which was the one with the smallest loss on the validation set, was selected as the baseline for the uncertainty methods.

5.2.2.2 Uncertainty methods

Following the split of uncertainty estimation methods discussed in Chapter 3, we selected the most common methods to estimate uncertainty. Based on the baseline model, we measure aleatoric and/or epistemic uncertainty using a total of seven measures. For aleatoric uncertainty estimation, both maximum probability and (Shannon) entropy were employed. For epistemic uncertainty, we selected baseline measures developed to improve OOD uncertainty estimation, namely Joint Energy [53], Mahalanobis distance-based confidence score [51], Maximum Logit [50], Isolation Forest [55] and Local Outlier Factor [57].

Regarding Bayesian methods, we selected both MC Dropout and Laplace approximation to their easy implementation with slight changes in training logic. For MC Dropout, the same trained network was used without retraining since the baseline architecture contains dropout layers. In the testing, dropout layers were kept active, and 15 MC samples were used. For the Laplace approximation, the same trained network was also used since this method can be applied post-hoc to trained neural networks that use an exponential family loss function and piece-wise linear activation functions [150]. Therefore, to approximate the intractable posterior distribution over the parameters of neural networks, we used the implementation of Rewicki et al. [150] developed under the multi-label scenario and publicly available[1]. Similar to MC Dropout, 15 samples were used for testing.

For ensemble methods, the popular approach introduced by Lakshminarayanan et al. [72] where the same network is trained M independently times using different parameter initialization was selected. We will refer to this approach as DeepEnsemble. Additionally, an ensemble based on bootstrapping approach was also trained. Both approaches are composed of 15 individual ensemble members.

Figure 5.3 presents a summary of the uncertainty estimation methods and corresponding uncertainty measures applied on top of it.

It is important to note that for the calculation of uncertainty measures that are directly dependent on class probabilities, we considered independence between labels. For instance, entropy measures are applied in a binary setting scenario for each label, which results in an uncertainty measure per label. To take into account the joint uncertainty across labels, we summed the measure of label uncertainties.

5.2.3 Validation approach

To evaluate the generalization capabilities of the trained models, we considered three different test sets where classification performance, calibration, and uncertainty measures quality were evaluated. Applications for classification with rejection option, dataset shift, and active learning are also explored in the following subsections.

5.2.3.1 Training, validation, and test sets

To perform the external validation, instead of the typical procedure of training and testing with data from the same data source, a unique data source from the CPSC dataset was used for models' training and G12EC and PTB-XL for external validation. The CPSC data

[1] https://github.com/ferewi/tf-laplace

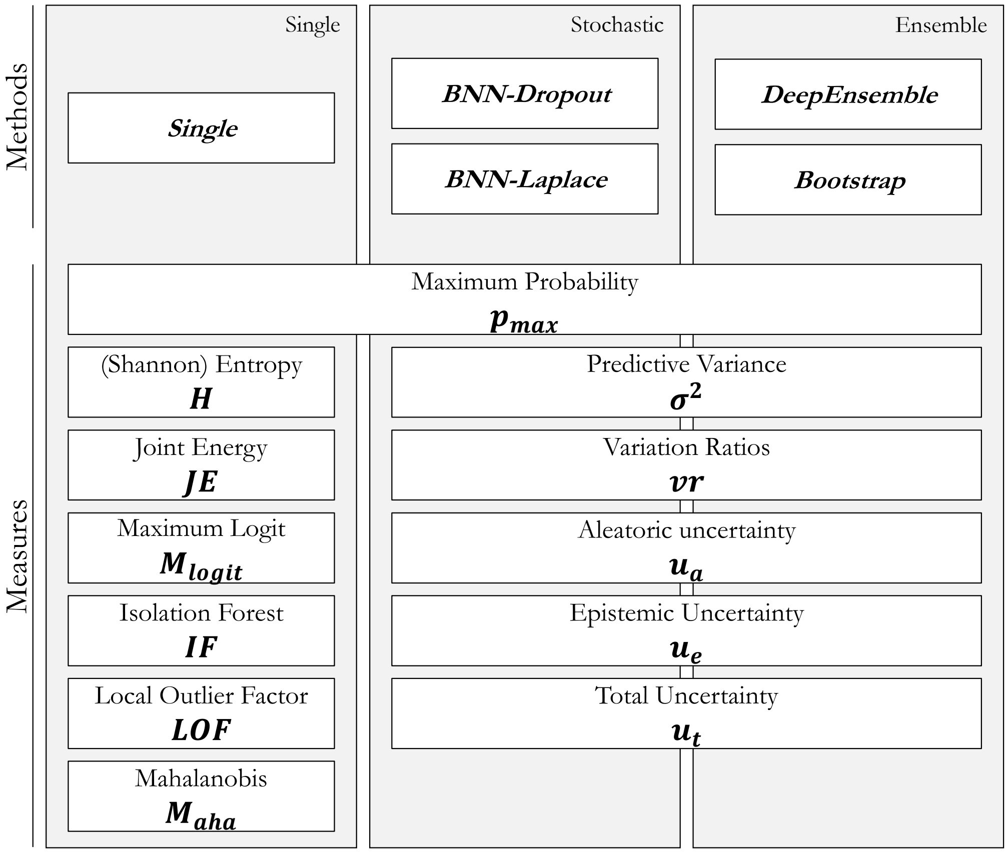


Figure 5.3: Uncertainty methods and corresponding uncertainty measures selected for this analysis. The acronyms used throughout this work are represented in bold.

was divided using an 80-10-10% train-val-test split. The class labels, gender, and age information were used as splitting criteria to ensure each set contains the same distribution of each criterion. Additionally, two OOD datasets were also considered for uncertainty quantification evaluation. Since we are not using all datasets' available classes for models' training, we selected a group of unknown classes as OOD. For this purpose, the hierarchical organization in terms of coarse superclasses and subclasses for the diagnostic labels provided by the PTB-XL dataset [178] was used. To reduce the similarity between the diagnostic labels used, we selected the Myocardial Infraction (MI) superclass and the Hypertrophy (HYP) superclass as OOD datasets. As the heterogeneous mixture of known and unknown classes can be presented in this set of labels, we removed all records that contain known classes mixed with these sets of unknown classes to ensure that OOD dataset contained only unknown classes.

Thus, the following test sets were used for evaluation purposes:

- IN (CPSC): Test set used for internal validation, i.e, an independent test from the same data source as the training set. This set contains a total of 687 recordings with the same proportion of class labels identified in Table 5.2;

- EXT (G12EC): The entire dataset from the G12EC dataset was used for external validation, containing a total of 4301 recordings;

- EXT (PTB-XL): The entire dataset from PTB-XL dataset was used for external validation, containing a total of 20214 recordings;

- OOD-MI: OOD dataset containing IMI, AMI, LMI and PMI diagnostic labels from PTB-XL dataset, totaling 2214 records.

- OOD-HYP: OOD dataset containing LVH, LAO/LAE, RVH, RAO/RAE and SEHYP diagnostic labels from PTB-XL dataset, totaling 1553 records.

The provided abbreviations will be used to refer to each test set during the experimental analysis.

5.2.3.2 Evaluation metrics

For classification performance evaluation, F1-score and AUROC were selected. Additionally, a binary multi-class, multi-label confusion matrix using the implementation provided by PhysioNet/CinC Challenge 2020 was used. Assuming a collection of diagnoses $C = [c_i]$, a confusion matrix is given by $A = [a_{ij}]$ where a_{ij} is the number of recordings in a database that were classified as belonging to class c_i but belong to class c_j.

Calibration evaluation was assessed by calculating ECE and reliability diagrams. Besides the calibration evaluation of uncertainty methods, post-hoc calibration methods were applied and compared with the base models. The post-hoc methods considered were: histogram binning, isotonic regression, bayesian binning into quantiles, ensemble of near isotonic regression, and temperature scaling.

Uncertainty estimation methods were evaluated and compared using threshold independent metrics, namely the AUCO for internal and external datasets, and the AUROC for OOD datasets. Additionally, we also employed threshold dependent measures, based on the concept of binary confusion matrices [81, 82]. Uncertainty accuracy, specificity, sensitivity, and precision were used for the evaluation of classification with the rejection option. A detailed description of all uncertainty measures employed in this study can be found in Section 3.3.

5.2.3.3 Applications

For the classification with rejection option, the uncertainty measures were used as a measure for rejection. The rejection threshold was obtained using the training data, where a given uncertainty training percentile is selected to reject samples on test data. Thus, for each test sample, the uncertainty is computed and compared with the defined threshold. If the obtained uncertainty value is greater than the threshold the sample is rejected and no prediction is made. On the other hand, if the uncertainty is lower than the threshold the model accepts the prediction, and a confidence level is also returned.

For dataset shift validation, two popular statistical divergence measures in the deep learning literature, the Wasserstein distance, and MMD, were applied to measure dataset similarity between internal and external validation. The Wasserstein-1 version of Wasserstein distance [184] was used and is given by:

$$W_1(X,Y) = \inf_{\pi \in \Gamma(X,Y)} \int_{\mathbb{R}\times\mathbb{R}} |x-y| \mathrm{d}\pi(x,y), \tag{5.1}$$

where $\Gamma(X,Y)$ is the set of distributions whose marginals are X and Y on the first and second factors, respectively. The variables x and y are samples from each distribution $\pi(x,y)$ from the set. Intuitively, the distance is given by the optimal cost of moving a distribution until it overlaps with the other. In our experiments, x and y are the feature representations of subsets of the train and test data; thus, W_1 represents the cost of mapping the distribution of x into the distribution of y (or vice versa).

Regarding the MMD, is a kernel-based statistical divergence measure that determines whether two given datasets come from the same distribution [185]. Given a fixed *kernel* function $k : X \times X \mapsto \mathbb{R}$ and two datasets X, Y with sizes $|X| = n$, $|Y| = m$, the MMD can be estimated as:

$$MMD(X,Y) = \frac{1}{n(n-1)} \sum_{i \neq j} k(x_i, x_j) + \frac{1}{m(m-1)} \sum_{i \neq j} k(y_i, y_j) - \frac{2}{nm} \sum_{i,j} k(x_i, y_j) \tag{5.2}$$

Intuitively, the MMD measures the distance between X and Y by computing the average similarity in X and Y separately and then subtracting the average cross-similarity between the two datasets, where the similarity between two instances is quantified by means of the selected kernel k. In this work, a simple linear kernel was selected. Furthermore, as for the Wasserstein distance, x and y represent the feature representations of subsets of the train and test data. Thus, MMD quantifies the average kernel similarity among instances in x and y, discounted by the cross-similarity between the two datasets. Both similarity measures were computed on the latent feature space, i.e., the embeddings extracted from the neural network, between the training set and each of the test sets.

For active learning validation, the samples are sorted based on their uncertainty values and the highest n uncertain samples are used for retraining the model. This process is performed using different uncertainty sources and compared to random sampling. The evaluation is based on the improvement of classification performance metrics.

5.3 Experimental Results

5.3.1 External validation

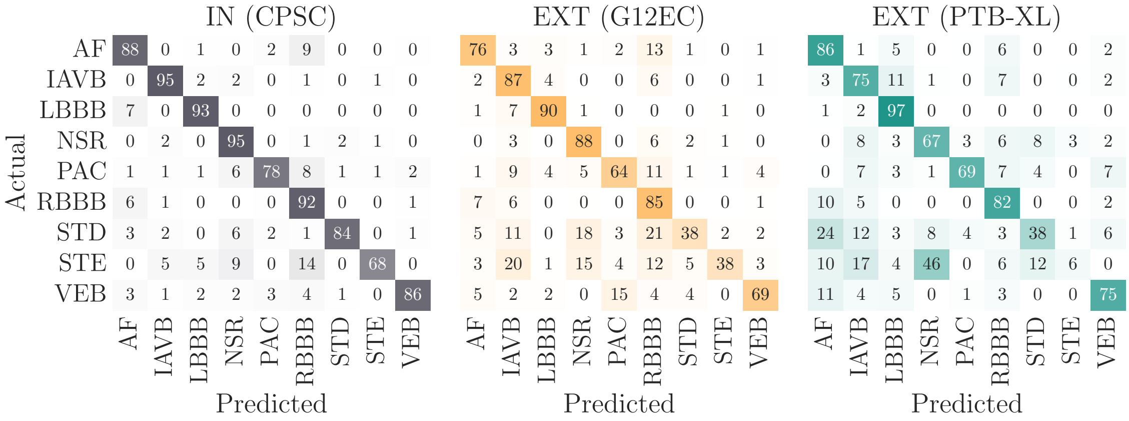
Although all the uncertainty methods share the same deep learning architecture, differences in training or testing procedures between them might affect not only the uncertainty estimation but also the predictive performance. To properly assess these uncertainty

Table 5.3: Global performance of uncertainty methods in internal (IN) and external (EXT) validation sets. The highest scores are represented in bold.

Model	IN (CPSC)		EXT (G12EC)		EXT (PTB-XL)	
	AUROC	F1-Score	AUROC	F1-Score	AUROC	F1-Score
Single Network	0.896	0.826	0.830	0.715	**0.734**	**0.567**
BNN-Dropout	0.890	0.833	0.811	0.699	0.700	0.516
BNN-Laplace	0.896	0.830	0.830	0.715	**0.735**	**0.568**
DeepEnsemble	**0.903**	**0.856**	**0.831**	**0.736**	0.724	0.559
Bootstrap	**0.903**	0.851	0.821	0.718	0.717	0.548

methods, we first present the classification performance for each method. Table 5.3 compares the AUROC and F1-score for each method during internal and external validation. The comparison indicates that the DeepEnsemble method performs slightly better than the other methods. However, the performance achieved within the same test set is similar across all methods. Table 5.3 also reveals a significant drop in performance during external validation, particularly in the PTB-XL dataset.

To analyze the class level performance between datasets, a binary multi-class, multi-label confusion matrix for each dataset was computed using the implementation provided by the PhysioNet/CinC Challenge 2020. As all methods demonstrated comparable performance measures, we only present the confusion matrices for the DeepEnsemble method in Figure 5.4. These confusion matrices reveal that in both external datasets, STE and STD diagnoses are not accurately recognized. In contrast, the Bundle Branch Blocks (LBBB and RBBB) maintain consistent performance across both internal and external datasets.



IN (CPSC)	AF	IAVB	LBBB	NSR	PAC	RBBB	STD	STE	VEB
AF	88	0	1	0	2	9	0	0	0
IAVB	0	95	2	2	0	1	0	1	0
LBBB	7	0	93	0	0	0	0	0	0
NSR	0	2	0	95	0	1	2	1	0
PAC	1	1	1	6	78	8	1	1	2
RBBB	6	1	0	0	0	92	0	0	1
STD	3	2	0	6	2	1	84	0	1
STE	0	5	5	9	0	14	0	68	0
VEB	3	1	2	2	3	4	1	0	86

EXT (G12EC)	AF	IAVB	LBBB	NSR	PAC	RBBB	STD	STE	VEB
AF	76	3	3	1	2	13	1	0	1
IAVB	2	87	4	0	0	6	0	0	1
LBBB	1	7	90	1	0	0	0	1	0
NSR	0	3	0	88	0	6	2	1	0
PAC	1	9	4	5	64	11	1	1	4
RBBB	7	6	0	0	0	85	0	0	1
STD	5	11	0	18	3	21	38	2	2
STE	3	20	1	15	4	12	5	38	3
VEB	5	2	2	0	15	4	4	0	69

EXT (PTB-XL)	AF	IAVB	LBBB	NSR	PAC	RBBB	STD	STE	VEB
AF	86	1	5	0	0	6	0	0	2
IAVB	3	75	11	1	0	7	0	0	2
LBBB	1	2	97	0	0	0	0	0	0
NSR	0	8	3	67	3	6	8	3	2
PAC	0	7	3	1	69	7	4	0	7
RBBB	10	5	0	0	0	82	0	0	2
STD	24	12	3	8	4	3	38	1	6
STE	10	17	4	46	0	6	12	6	0
VEB	11	4	5	0	1	3	0	0	75

Figure 5.4: Binary multi-class multi-label confusion matrices for DeepEnsemble method in internal (IN) and external (EXT) validation sets.

The correlation between data similarity and generalization properties across datasets has been previously identified as a strong indicator that the datasets originate from different distributions. Consequently, information about similarity can offer valuable insights into understanding why a machine learning model exhibits poor performance on an

Table 5.4: Comparison of metrics over the three test sets based on the embeddings extracted from the neural network. For each setting, values were averaged over every test set.

Metric	**IN (CPSC)**	**EXT (G12EC)**	**EXT (PTB-XL)**
Wasserstein	1.513 ± 0.007	2.199 ± 0.017	2.311 ± 0.023
MMD	0.019 ± 0.004	0.931 ± 0.023	1.238 ± 0.050

external dataset [156].

Table 5.4 presents a comparison of dataset similarity between internal and external datasets using Wasserstein distance and MMD. It is important to note that the raw values between metrics have different scales and are not comparable, only the ordering of test sets can be analyzed. Both metrics align with the test set order, which is also consistent with the classification performance in Table 5.3.

To delve deeper into the differences between test sets, a class-labeled dataset distance analysis was conducted using Wasserstein distance (as both metrics produced the same results for overall dataset similarity). Figure 5.5 illustrates the correlation between the performance drop and Wasserstein distance using the three datasets. The worst class performances observed in confusion matrices (Figure 5.4) also correspond to those with

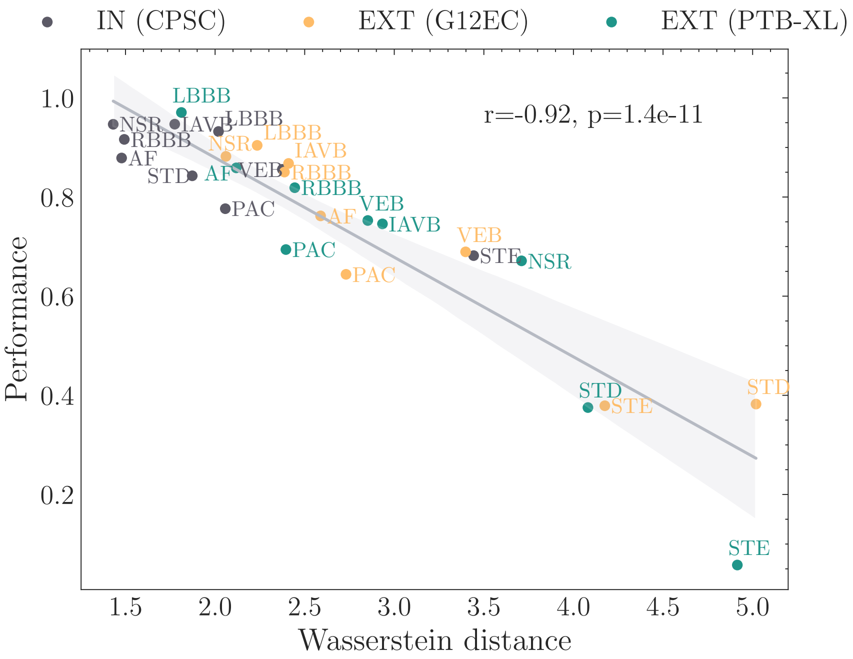


Figure 5.5: Class performance drop as a function of Wasserstein distance between training and each represented test set. Each point is annotated with the class name abbreviation and the color represents the dataset. The linear regression is obtained with all datasets and represented in gray. The Pearson correlation coefficient (r) and p-value (p) for testing non-correlation are annotated in the graph area.

Table 5.5: Performance comparison of different combinations of internal (IN) and external (EXT) validation sets.

CPSC		**G12EC**		**PTB-XL***	
Validation	F1-score	Validation	F1-score	Validation	F1-score
IN	0.856	EXT	0.736	EXT	0.699
IN	0.807	EXT	0.741	IN	0.891
IN	0.849	IN	0.818	EXT	0.728
EXT	0.722	IN	0.832	IN	0.885
IN	0.826	IN	0.815	IN	0.884

higher Wasserstein distances. The calculated Pearson correlation coefficient ($r = -0.92$) suggests that there is a potential shift (label-dependent) in external datasets, and the Wasserstein distance proves to be useful in detecting it. In addition to the STE and STD classes, the NSR (Normal Sinus Rhythm) class from the PTB-XL dataset also exhibits a higher distance and a significant drop when compared to the same class in the CPSC and G12EC datasets. Based on this observation, we carried out a thorough examination of the NSR class label and discovered a significant difference in NSR annotations across the three datasets. To align with the annotation of the training dataset, only a subset of the NSR class from the PTB-XL dataset will be utilized for the remainder of the analysis. We refer to this subset as PTB-XL*. A comprehensive explanation and the results obtained can be found in C.1.

Table 5.5 presents the performance results for various combinations of internal and external sets and Figure 5.6 the correlation between Wasserstein distance and global model

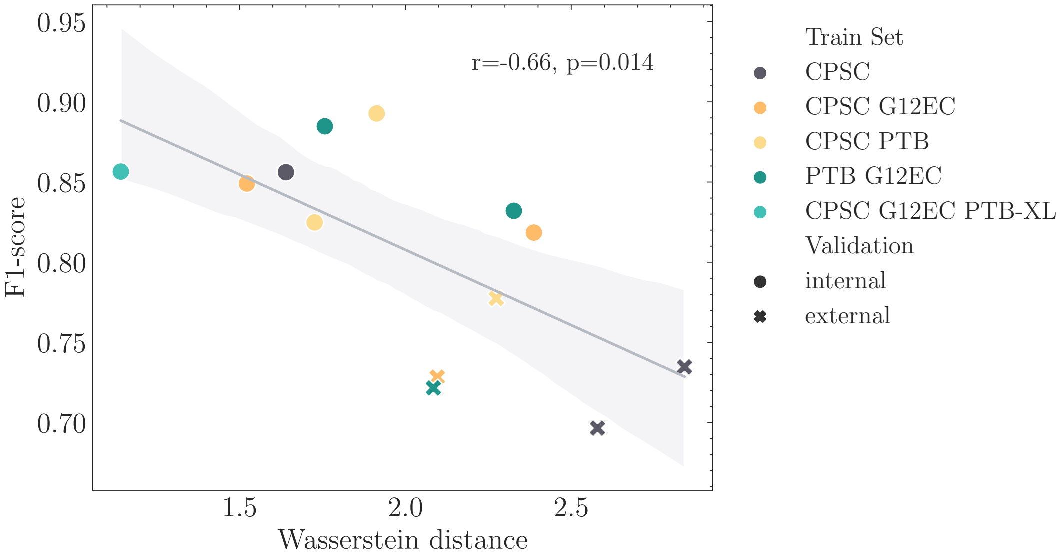


Figure 5.6: Correlation between Wasserstein distance and F1-score using different datasets combinations for internal and external datasets. The Pearson correlation coefficient (r) and p-value (p) for testing non-correlation are annotated in the graph area.

performance of different combinations. All models followed the same training procedure, as detailed in section 5.2.2. Independent validation sets were utilized for internal validation, either using the publicly available data partition or an 80-10-10% train-val-test split, with class labels, gender, and age serving as splitting criteria. Regardless of the combination, internal validation sets consistently achieved a performance higher than 0.80, while external validation sets showed performance below 0.75. Nevertheless, incorporating additional datasets for training led to enhanced performance on external datasets.

5.3.2 Calibration

As introduced in Section 3.4 uncertainty estimation methods such as deep ensembles and BNN can be seen as calibration methods. Besides the analysis of calibration errors with these uncertainty measures, post-hoc calibration methods were also applied. As post-hoc methods, Histogram Binning (Hist), Isotonic Regression (IsoReg), Bayesian Binning into Quantiles (BBQ), Ensemble of Near Isotonic Regression (ENIR), and Temperature Scaling (Temp) were applied. Note that to apply these post-hoc calibration methods in a multi-label setting, it is necessary to assume the independence between probability estimates for each class.

Table 5.6 shows the ECEs for all uncertainty methods, post-hoc calibration methods, and datasets using 10 bins. All uncertainty methods achieved equal or lower ECEs compared to the Single Network, with BNN-Dropout on the internal validation being the only exception. The DeepEnsemble model obtained the lowest ECE in CPSC and G12EC datasets, while BNN-Dropout obtained the lowest ECE in PTB-XL. BNN-Laplace was the least effective uncertainty method, exhibiting similar results to the Single Network.

Regarding post-hoc calibration methods, none of the methods were able to improve calibration in the external test sets. In the case of ensemble methods, even in the internal test set, all post-hoc methods obtained higher ECE than the default uncertainty method. For the Single Network, BNN-Dropout, and BNN-Laplace, only BBQ improved the calibration in the three models.

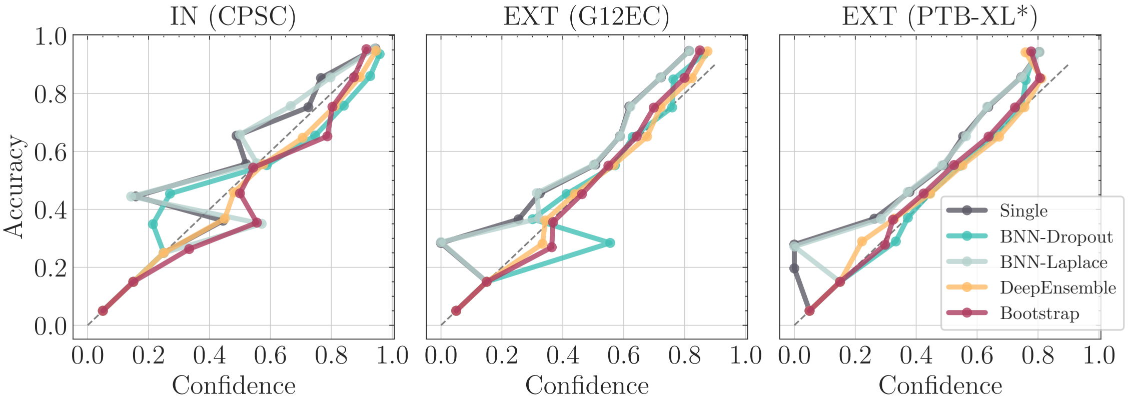


Figure 5.7: Reliability diagrams for internal (IN) and external (EXT) validation sets. The diagonal dashed line represents the perfect calibration.

Based on these results, reliability diagrams were calculated using the default probabilities of each model, i.e without using any post-hoc calibration method. Figure 5.7 shows the reliability diagrams for each test set and model using 10 bins. From the reliability diagrams, we can observe that the Single Network and BNN-Laplace exhibit similar behavior, with their estimates being overconfident across all datasets. Both ensemble methods display similar behavior in all datasets, with the DeepEnsemble appearing to be more robust across the various datasets.

Table 5.6: Expected Calibration Error (ECE) for internal (IN) and external (EXT) validation sets. The lowest errors are represented in bold.

Model	**Post-hoc**	**IN (CPSC)**	**EXT (G12EC)**	**EXT (PTB-XL*)**
Single Network	-	0.047	0.121	0.115
	Hist	0.047	0.151	0.220
	IsoReg	0.050	0.125	0.163
	BBQ	0.041	0.163	0.243
	ENIR	0.053	0.129	0.164
	Temp	0.046	0.122	0.131
BNN-Dropout	-	0.066	0.043	**0.037**
	Hist	0.038	0.130	0.174
	IsoReg	0.039	0.101	0.148
	BBQ	0.041	0.107	0.163
	ENIR	0.051	0.109	0.157
	Temp	0.049	0.129	0.146
BNN-Laplace	-	0.050	0.120	0.114
	Hist	0.056	0.124	0.159
	IsoReg	0.048	0.116	0.155
	BBQ	0.034	0.145	0.194
	ENIR	0.063	0.125	0.150
	Temp	0.051	0.117	0.133
DeepEnsemble	-	**0.026**	**0.034**	0.045
	Hist	0.054	0.089	0.142
	IsoReg	0.067	0.091	0.146
	BBQ	0.054	0.107	0.172
	ENIR	0.064	0.088	0.146
	Temp	0.045	0.106	0.126
Bootstrap	-	0.045	0.048	0.062
	Hist	0.059	0.100	0.167
	IsoReg	0.060	0.086	0.126
	BBQ	0.053	0.092	0.132
	ENIR	0.059	0.093	0.133
	Temp	0.055	0.133	0.161

5.3.3 Uncertainty quantification

As an initial illustrative visualization, we present the overall uncertainty of internal, external, and OOD datasets using the DeepEnsemble method in Figure 5.8. The uncertainty values were normalized to their maximum theoretical values, ensuring that all uncertainty measures are bounded within the range of 0 and 1. Noticeably, all measures consistently increase the overall uncertainty, regardless of the uncertainty measure employed. Ideally, we would like the increase in uncertainty values to coincide with the reduction in classification performance observed earlier. In other words, we would expect the uncertainty values to remain as consistent as possible under different external validations (as well as on OOD data) to indicate that models are uncertain when predicting a given input.

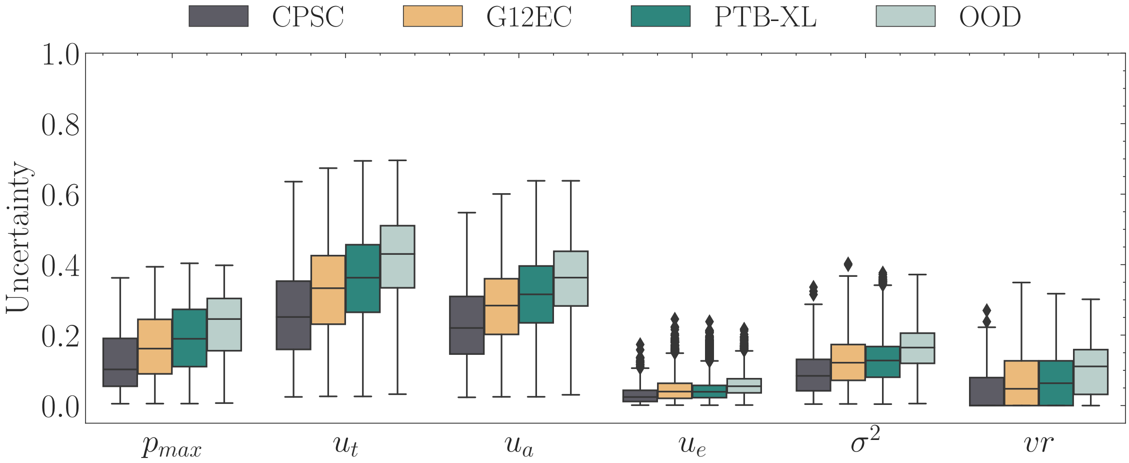
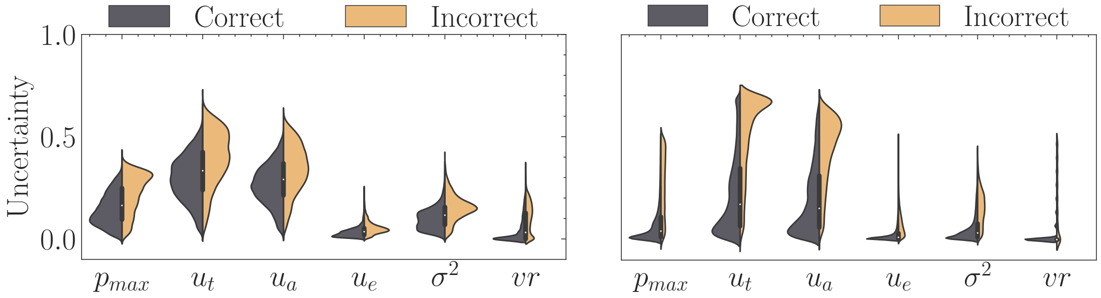


Figure 5.8: DeepEnsemble uncertainty measures distributions for internal (CPSC), external (G12EC and PTB-XL), and Out-of-Distribution (OOD) datasets. Uncertainty measures are normalized with their maximum theoretical value, where 1 represents the maximum possible uncertainty. The OOD label contains data from both OOD-MI and OOD-HYP sets.

In addition to assessing the uncertainty between datasets, it is also possible to statistically evaluate the relationship between the distributions of uncertainty values for correctly and incorrectly classified samples. In a multi-label setting, we can consider two scenarios: 1) a label dependence scenario, in which the entire label combination is treated as either correct or incorrect, and 2) a label independence scenario, in which each class is addressed as a separate binary classification problem. Figure 5.9 illustrates the distributions of these two scenarios, using DeepEnsemble as an example. The uncertainty distributions for the label independence scenario display a more pronounced distinction between correctly and incorrectly classified samples compared to the label dependence scenario. When applying the non-parametric Mann-Whitney U statistical test [186] for unpaired groups, both approaches were found to be statistically significant at a .05 significance level (even with Benjamini-Hochberg p-values correction [187]). In addition to evaluating the practical significance using Cohen's d effect size [188], we also computed the effect sizes for each method. The label independence approach was found to yield larger effect sizes. The complete results can be found in the C.2.



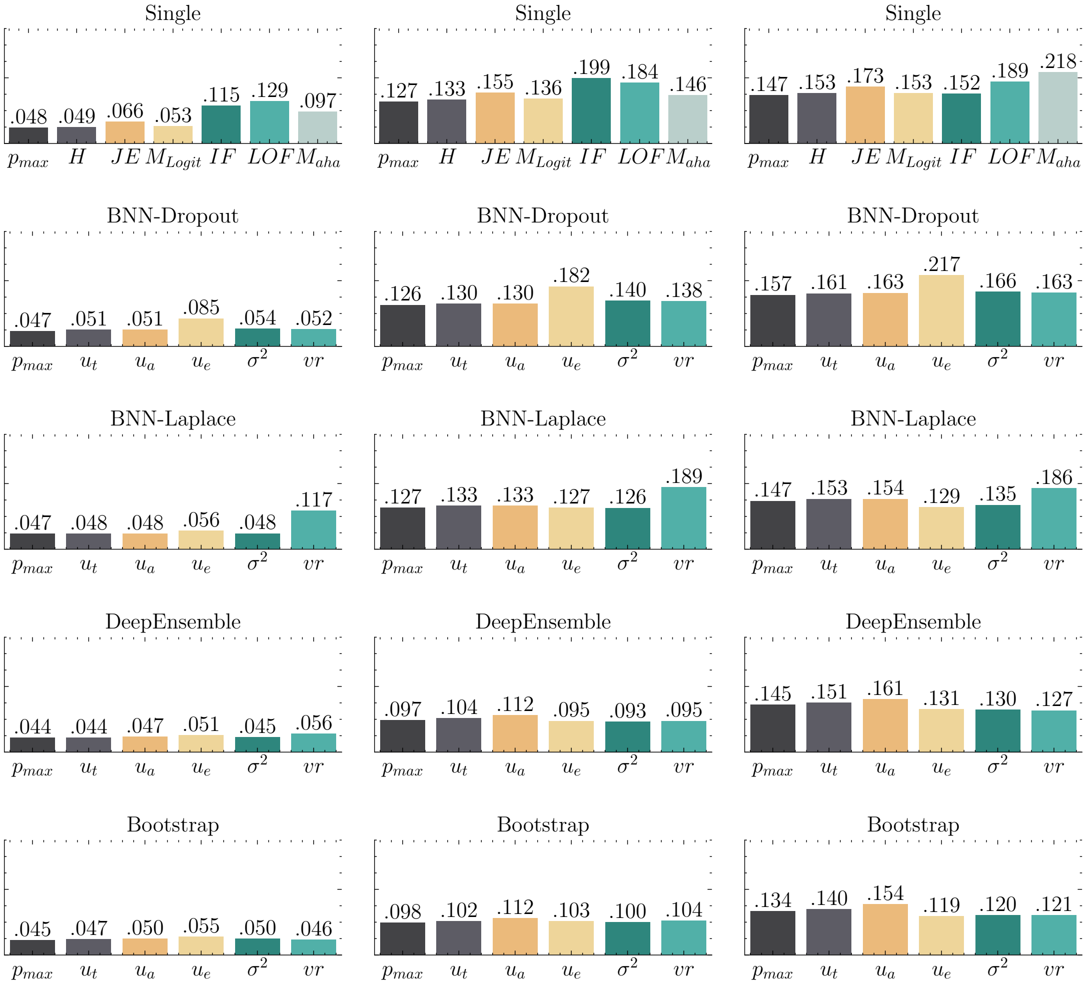
(a) Label dependence.

(b) Label independence.

Figure 5.9: Comparison of uncertainty value distributions for correctly and incorrectly classified samples using DeepEnsemble model. (a) label dependence approach, where the entire label combination is considered as either correct or incorrect, and (b) label independence approach, where each class is treated as a separate binary classification problem.

To enable a fair comparison among all uncertainty methods and their corresponding measures, the AUCO metric was calculated for both internal and external test sets. Smaller AUCO values indicate better performance. Figure 5.10 presents the results, with the same color representing the same uncertainty measure across different methods, except for the Single Network method where uncertainty measures are different. In addition to the differences in performance between internal and external validation, Figure 5.10 also clearly illustrates that uncertainty estimation measures are affected in external validation. In general, ensemble-based uncertainty measures appear more robust in preserving the correct uncertainty ordering compared to other methods, and epistemic uncertainty measures outperform aleatoric uncertainty measures in external validation. In internal validation, maximum probability (p_{max}) achieved the lowest AUCO across all methods. This is somewhat expected, as the internal dataset exhibits low epistemic uncertainty, unlike the external validation datasets. It is also worth noting that OOD detection measures (JE, M_{Logit}, IF, LOF, M_{aha}) do not perform as well as other methods in this rank-based analysis. Although there is a close relationship between uncertainty estimation and OOD detection, ordering uncertainty values and detecting OOD are not the same problem, which might explain the lower performance of these methods.

In regard to OOD detection, the two superclass sets (MI and HYP) from the PTB-XL dataset, consisting solely of unknown classes, were employed as OOD samples. The AUROC was calculated with the OOD samples as positive instances and the internal CPSC test samples as negative instances. The obtained results are presented in Table 5.7. In line with the previous analysis, ensemble methods surpassed other approaches in terms of AUROC. For the MI set, total uncertainty u_t (or entropy H for Single methods) achieved the highest AUROC across all methods. In contrast, for the HYP set, epistemic



(a) IN (CPSC) (b) EXT (G12EC) (c) EXT (PTB-XL)

Figure 5.10: Ranking performance evaluation using Area Under the Confidence-Oracle (AUCO) for all datasets and uncertainty methods.

uncertainty u_e yielded higher AUROC values for ensemble methods and BNN-Laplace. As for the methods specifically designed for OOD detection (JE, IF, LOF), their performance in distinguishing OOD samples was surprisingly poor.

While the OOD problem typically refers to anomaly and/or outlier detection, where OOD samples come from entirely different distributions, in our setting, the OOD samples consist of classes from the same datasets and are thus more related to novelty detection associated with the OSR scenario. Although OSR is similar to OOD detection, it is likely more challenging to address, as the statistics of the new classes often resemble those of existing classes within the dataset [28].

Table 5.7: OOD detection performance comparison using all uncertainty methods and measures. OOD datasets are composed of two superclasses sets (MI and HYP) with only unknown classes from the PTB-XL dataset.

Model	**Uncertainty**	**AUROC**	
		OOD-MI	**OOD-HYP**
Single Network	p_{max}	0.758	0.702
	H	0.767	0.703
	JE	0.749	0.715
	M_{Logit}	0.761	0.716
	IF	0.524	0.617
	LOF	0.502	0.633
	M_{aha}	0.614	0.569
BNN-Dropout	p_{max}	0.763	0.717
	vr	0.671	0.647
	σ^2	0.645	0.648
	u_a	0.773	0.719
	u_e	0.450	0.529
	u_t	0.767	0.717
BNN-Laplace	p_{max}	0.759	0.704
	vr	0.574	0.575
	σ^2	0.742	0.727
	u_a	0.767	0.704
	u_e	0.710	0.730
	u_t	0.767	0.704
DeepEnsemble	p_{max}	0.781	0.752
	vr	0.721	0.736
	σ^2	0.778	0.790
	u_a	0.787	0.740
	u_e	0.751	0.787
	u_t	**0.794**	0.764
Bootstrap	p_{max}	0.775	0.757
	vr	0.735	0.747
	σ^2	0.783	0.793
	u_a	0.776	0.730
	u_e	0.764	**0.794**
	u_t	0.791	0.767

5.3.4 Classification with rejection-option

In the previous sections, we compared various uncertainty estimation methods using threshold-independent measures. However, to evaluate the benefits of integrating AI uncertainty estimation methods in supporting medical decision-making within cardiology, a confidence threshold must be established. This threshold enables the classifier to abstain in situations with high uncertainty. The selection of a threshold restricts the comparison

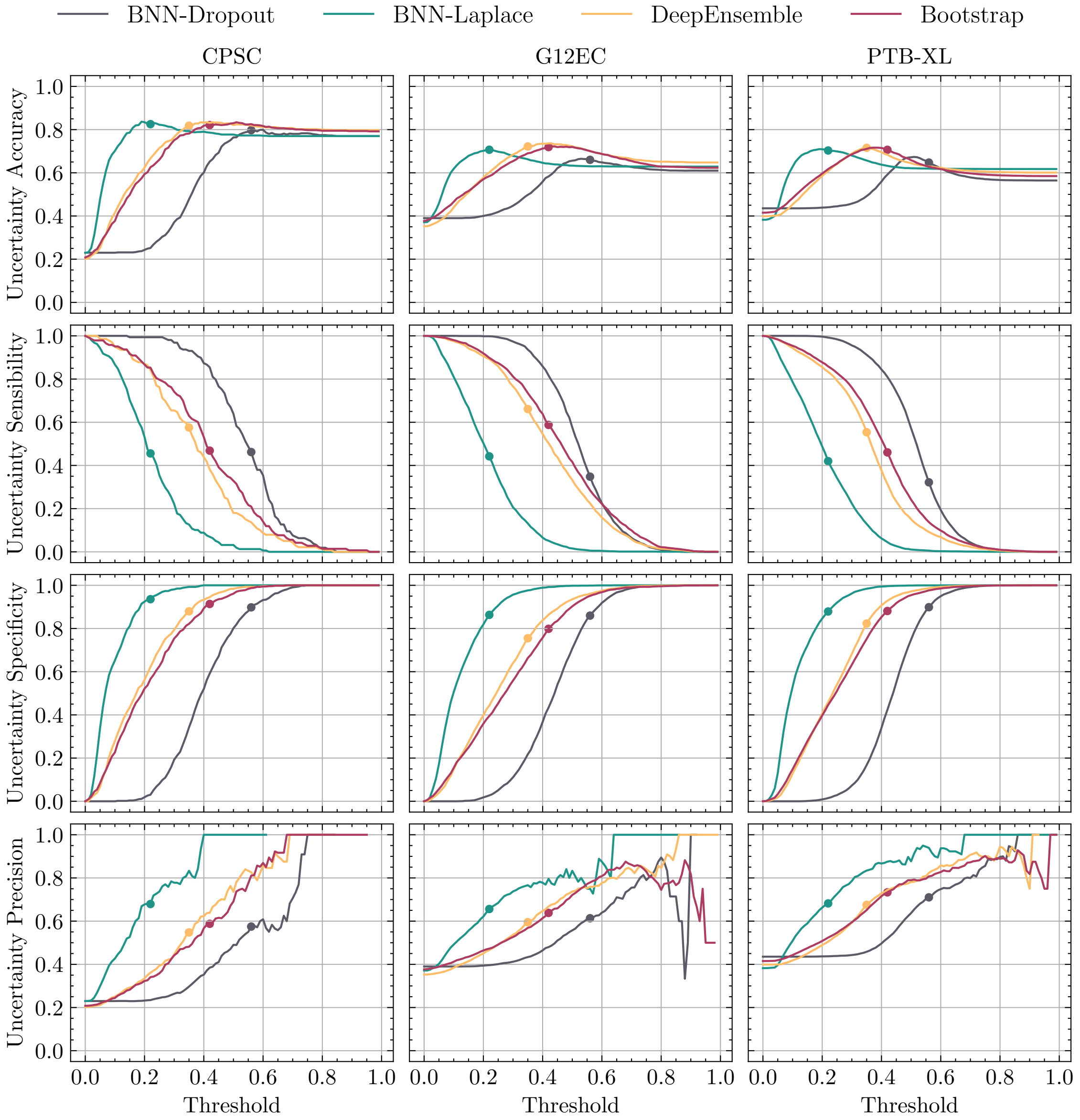


Figure 5.11: Uncertainty performance measures for varying threshold values across different datasets using four uncertainty estimation methods with epistemic uncertainty. The chosen threshold for each method is denoted by a data point superimposed on each line plot.

among methods, as each method may have a varying optimal threshold.

Figure 5.11 depicts the predictive uncertainty performance evaluation metrics for the three datasets, using the uncertainty estimation methods while varying the uncertainty threshold. For visualization purposes, we selected the epistemic uncertainty (σ^2) to demonstrate the differences in thresholds for the various methods. Although these differences are more pronounced when using epistemic uncertainty measures, aleatoric uncertainty also presents some disparities between methods (see Figure 5.12). Regardless of the chosen threshold, we can observe from the figure that there is a degradation

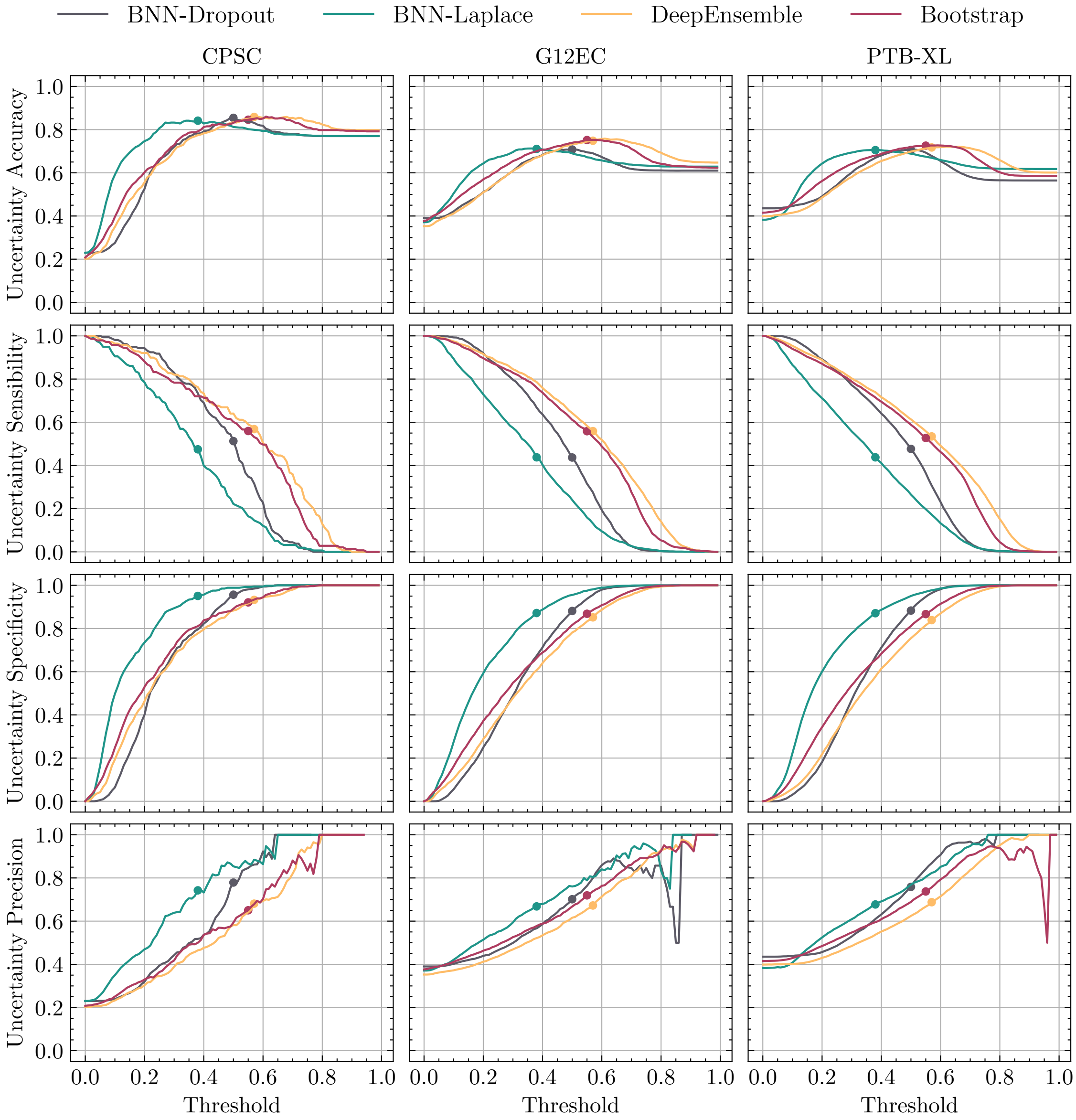


Figure 5.12: Uncertainty performance measures for varying threshold values across different datasets using four uncertainty estimation methods with aleatoric uncertainty. The chosen threshold for each method is denoted by a data point superimposed on each line plot.

of performance metrics in the external datasets. For instance, the uncertainty accuracy in internal validation reaches a maximum of over 0.8, while in external datasets, the maximum is approximately 0.70.

Since the proper definition of an uncertainty threshold is beyond the scope of this work, we opted for an analysis based on a given rejection rate obtained in training. Consequently, we defined a 15% rejection rate on the training set and used the corresponding uncertainty value to reject samples on the testing sets. Each threshold is represented by a data point placed on top of each line plot in Figure 5.11, emphasizing that the threshold

Table 5.8: Performance evaluation metrics for classification with rejection option using DeepEnsemble method. F1-score is presented without rejection (baseline) and with rejection (Non-rejected F1-score). The rejection threshold was set to 15% rejection on the training set. Predictive uncertainty evaluation measures used the same threshold.

Metric	**IN (CPSC)**	**EXT (G12EC)**	**EXT (PTB-XL*)**
F1-score (Baseline)	0.856	0.736	0.699
Non-rejected F1-score	0.915	0.844	0.795
Rejection Rate	0.207	0.385	0.317
Uncertainty Accuracy	0.818	0.722	0.716
Uncertainty Sensibility	0.576	0.661	0.554
Uncertainty Specificity	0.880	0.755	0.823
Uncertainty Precision	0.548	0.595	0.675

value differs for each method and selecting the same threshold for all methods does not provide a fair comparison between methods.

To simplify the analysis of experimental results for classification with a rejection option, Table 5.8 presents a summary of the complete performance results with rejection for the best uncertainty method, the DeepEnsemble. The uncertainty rejection measure used was the variance of ensemble members' probabilities, σ^2, as it achieved better performance measures in the previous analysis.

The first observation from Table 5.8 is that rejecting highly uncertain samples improves classification performance across all datasets. For the same threshold, the rejection rate varies considerably between datasets. As expected, the internal dataset CPSC exhibits a lower rejection rate (similar to the 15% applied in training), but for the external datasets, the rejection rate more than doubles, reaching 0.385 and 0.317 for G12EC and PTB-XL datasets, respectively. This observation aligns with the results obtained so far, in which the external test sets contain more uncertain samples. Applying the same threshold for the OOD datasets results in rejection rates of 0.634 and 0.666 for OOD-MI and OOD-HYP, respectively. While the performance of non-rejected samples can be considered acceptable (at least comparable to the internal validation), more than 30% of OOD samples were not rejected, which might be a substantial proportion of OOD samples. Naturally, lowering the uncertainty threshold would reject more OOD samples. However, this comes at the cost of rejecting more samples from known classes.

Table 5.8 also highlights acceptable uncertainty accuracy. However, with the selected threshold, all models exhibit higher specificity than sensitivity. This means that if we want to increase sensitivity (while decreasing specificity), the rejection rate will also increase.

5.3.5 Data quality

As the datasets do not have signal quality information, an exploratory evaluation was done by assessing the performance and uncertainty estimation of signals classified as acceptable or unacceptable signals. Due to the absence of ground truth, this analysis was simplified to use only lead II instead of the 12-leads used previously. The DeepEnsemble model and the best uncertainty measures for aleatoric and epistemic uncertainty were considered for this analysis. Based on the results of AUCO, the maximum probability p_{max} was selected for aleatoric uncertainty and the variance of ensemble members' probabilities σ^2 for epistemic uncertainty.

For the feature-based approach using stationary, heart-rate, and signal-to-noise features, the hyperparameters and thresholds used were set to be equal to those in the work of Kramer et al. [181]. Table 5.9 shows that this approach classified 2.1%, 6.7%, and 5.5% of CPSC, G12EC, and PTB-XL samples as unacceptable, respectively. When comparing the classification performance of unacceptable versus acceptable samples, we observe a decrease in performance within the unacceptable subset. However, due to the small percentage of samples classified as unacceptable, the performance of the acceptable subset shows no improvement.

Table 5.9: Percentage of samples and classification performance of unacceptable and acceptable subsets using the feature-based approach.

	% samples		**F1-score**	
	unacceptable	acceptable	unacceptable	acceptable
IN (CPSC)	2.2	97.8	0.77	0.86
EXT (G12EC)	6.7	93.3	0.72	0.74
EXT (PTB-XL)	5.5	94.5	0.65	0.70

For the non-feature-based approach, a one-dimensional CNN-based autoencoder was implemented for ECG lead II. The training data was the same as that used in the previous experimental analysis, i.e., the training partition of the CPSC dataset. However, to enhance the data quality utilized for autoencoder training, samples classified as unacceptable by the feature-based approach were excluded from the training set, resulting in the rejection of 180 training samples (3.3%). The reconstruction error was employed to establish a rejection threshold. The 99^{th} percentile of training samples reconstruction error was set as the upper threshold for rejecting a test sample. Figure 5.13 displays an example of autoencoder reconstruction of an acceptable signal (left) and an unacceptable signal (right).

The results obtained for the percentage of samples classified as unacceptable and the corresponding classification performance are presented in Table 5.10. Similar to the feature-based approach, the classification performance remained unchanged for the acceptable subset. However, the proportion of unacceptable data decreased to 1.2%, 1.8%,

Table 5.10: Percentage of samples and classification performance of unacceptable and acceptable subsets using the non-feature-based approach (autoencoder).

	% samples		**F1-score**	
	unacceptable	acceptable	unacceptable	acceptable
IN (CPSC)	1.2	98.8	0.62	0.86
EXT (G12EC)	1.8	98.2	0.48	0.74
EXT (PTB-XL)	1.3	98.7	0.49	0.70

and 1.3% for CPSC, G12EC, and PTB-XL samples, respectively. Regarding the classification performance of unacceptable data, the autoencoder exhibited a more significant decrease in performance compared to the feature-based approach.

In principle, unacceptable data should be associated with high uncertainty. Based on this hypothesis, Figure 5.14 compares the distribution of unacceptable versus acceptable data for the feature-based approach (Figure 5.14(a)) and the non-feature-based approach (Figure 5.14(b)). The visualization clearly demonstrates that the unacceptable samples classified by the autoencoder exhibit high uncertainty. To quantitatively verify this, we employed the non-parametric Mann-Whitney U statistical test to compare the distributions of unacceptable and acceptable samples. As anticipated, the unacceptable and acceptable subsets were significantly different for all datasets with a p-value lower than 0.01 when using the autoencoder. However, the results for the feature-based approach were not statistically significant for all datasets. The detailed results can be found in Table 5.11.

The results suggest that the samples classified as unacceptable by the autoencoder are likely to have low quality, but a definitive conclusion cannot be made without expert

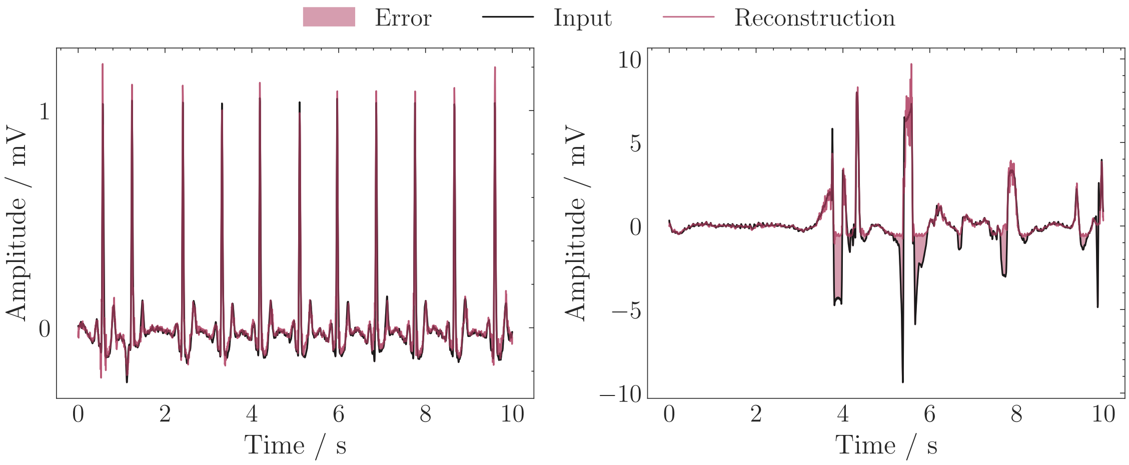
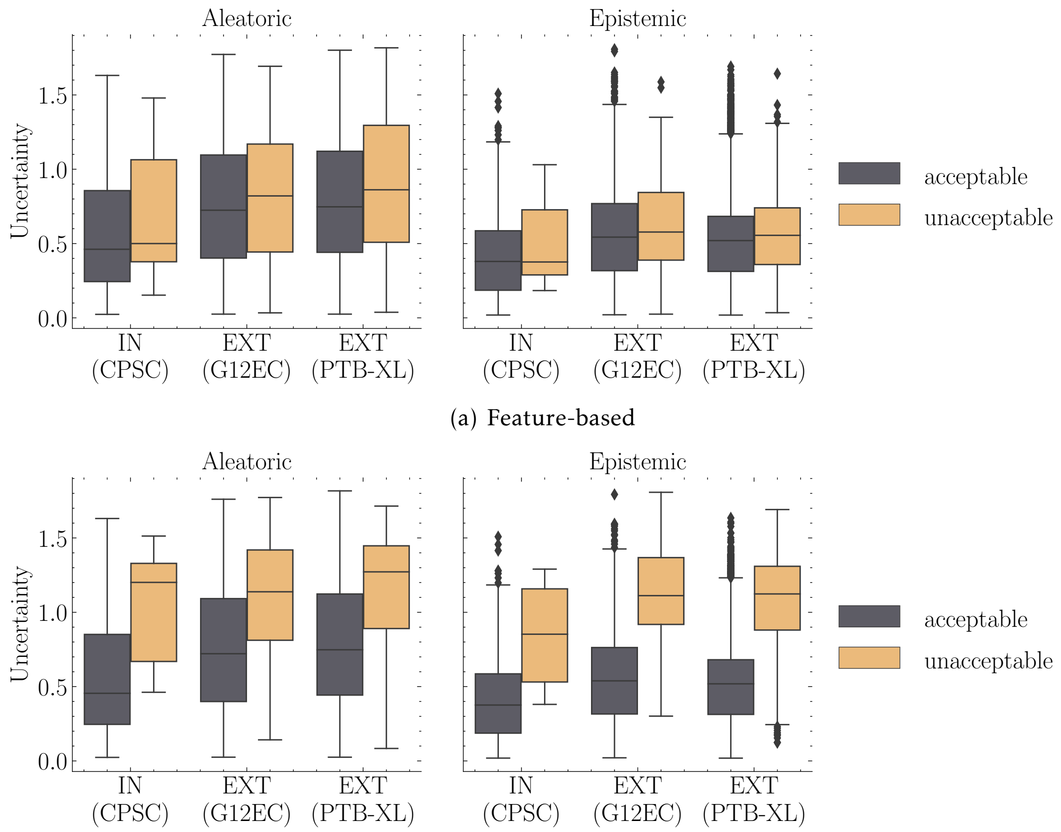


Figure 5.13: Example of reconstruction ECG signals using the autoencoder. On the left side, the ECG signal was classified as acceptable, and on the left side, the signal was classified as unacceptable.



(a) Feature-based

(b) Non-feature-based (autoencoder)

Figure 5.14: Aleatoric and epistemic uncertainty values distribution for unacceptable and acceptable ECG signals and for internal and external datasets.

annotation. Since the used datasets do not include annotations for data quality assessment, we validate our approach using the PhysioNet/Computing in Cardiology Challenge 2011 dataset [183], which contains quality assessment annotations reviewed by a group of annotators with varying levels of expertise in ECG analysis. Using this dataset, we obtained a sensitivity of 72% and a specificity of 99%, indicating that our approach does not detect all samples considered unacceptable, but few acceptable samples are deemed unacceptable.

Table 5.11: Statistical analysis to compare the unacceptable and acceptable subsets distributions for feature-based and non-feature-based approaches. The values represent the obtained p-values using the non-parametric Wilcoxon-Mann-Whitney test.

Approach	**Uncertainty**	**IN (CPSC)**	**EXT (G12EC)**	**EXT (PTB-XL)**
Feature-based	Aleatoric	2.31e-01	2.36e-02	4.28e-10
	Epistemic	1.90e-01	9.00e-03	8.24e-06
Non-featured-based	Aleatoric	2.36e-03	1.89e-11	3.12e-22
	Epistemic	9.74e-04	3.59e-30	3.06e-60

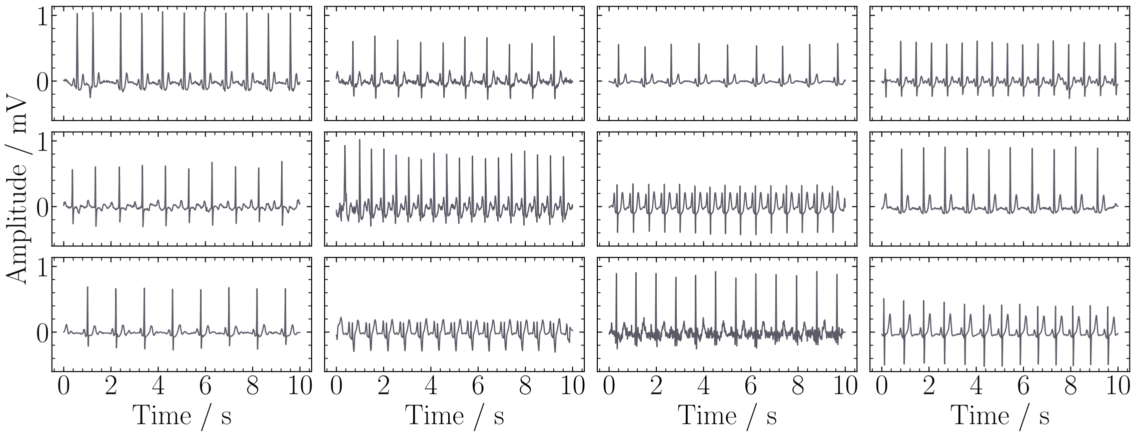


Figure 5.15: ECG lead II examples of signals classified as acceptable for both approaches. From the top row to the down row, the ECG signals are from the CPSC, G12EC, and PTB-XL datasets.

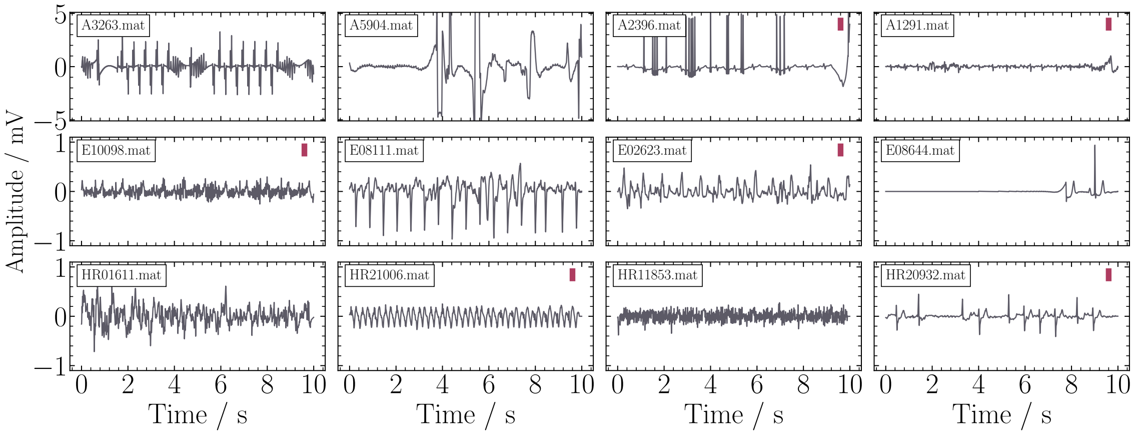


Figure 5.16: ECG lead II examples of signals classified as unacceptable for at least one of the tested approaches. From the top row to the down row, the ECG signals are from the CPSC, G12EC, and PTB-XL datasets. ECG signals with a red square on the upper right corner represent the signals not rejected using the classification with rejection option. The filename on the upper corner is displayed for reproducibility.

Although this validation supports our approach, we also conducted a visual inspection to better validate our results using the CPSC, G12EC, and PTB-XL datasets. Figure 5.15 displays ECG signals classified as acceptable by both approaches. The ECG signals from the first row were randomly selected from the CPSC dataset, the second row from the G12EC dataset, and the last row from PTB-XL. Although some signals exhibit more or less noise, it is possible to observe the common pattern of an ECG signal in all signals from Figure 5.15. On the other hand, Figure 5.16 presents ECG signals classified as unacceptable by at least one approach. The signals were visually selected to showcase the low quality of some ECG signals in the used datasets. The signals in each row are from the CPSC, G12EC, and PTB-XL datasets, respectively, and the filename is displayed for reproducibility.

Additionally, the ECG signals with a red square in the upper right corner represent signals not rejected using the previously defined threshold for the classification with rejection option (from Section 5.3.4), meaning that they do not exhibit high uncertainty values. Although the signals definitely have low quality, it can be debated whether this low-quality signal associated with lead II is affecting the classification of the 12-lead trained model since not all leads are required to diagnose heart disease.

Since this experimental analysis resulted in a low rejection rate of signals classified as unacceptable and was both quantitatively and qualitatively evaluated, the samples deemed unacceptable by the autoencoder were discarded for subsequent experiments.

5.3.6 Active learning

An essential procedure after deploying a model in clinical practice is continuous training to respond to changes in the data and prevent models from becoming unreliable and inaccurate. For model retraining, it is necessary to label data that requires expert knowledge. Obtaining large amounts of labeled data can be unfeasible during clinical practice. One possible approach to reduce this effort is to rely on active learning to select what unlabeled data would be most informative to the model and ask an expert annotator for a label on only these selected samples.

Following this reasoning, we retrained the DeepEnsemble method using data from PTB-XL and G12EC datasets. The retraining procedure consisted of selecting the rejected samples with higher uncertainty from one of the datasets and retraining the model with these new samples. For comparison purposes, we repeated this process 5 times using different uncertainty measures and random sampling. The process consisted of retraining the model using 400 new samples and repeating the process eight more times with a step of 400 new samples, totaling 3200 samples at the end of the process. For this analysis, we split the external datasets into train and test sets and present the results always using the test set for a fair comparison. Since PTB-XL has an available 10-fold split provided by PhysioNet, we used the last fold, as proposed by PhysioNet, for the test set and the other folds for training. For the G12EC dataset, since there is no proposed split, we used a 90-10% train-test split using classes, gender, and sex as group criteria for data splitting. The obtained performance in these test sets was similar to the performance using the entire dataset and represented the first point (0 samples) in the plots of Figure 5.17.

Figure 5.17 shows the evolution of classification performance with the increased number of samples used to retrain the models. In the first row, data from the G12EC dataset was used to retrain the model, and in the second row, PTB-XL data was used. The gray background represents the dataset used to retrain the models. Besides the performance evolution within the dataset used for retraining, we also show the classification performance in the other datasets to ensure that the increase in performance in one dataset does not represent a performance degradation in the other datasets. Observing Figure 5.17 we note that adding new samples from external datasets does not affect the performance in the internal CPSC dataset. Contrary, adding new samples from one of the external datasets increased not only the performance on that dataset but also the performance in the other external dataset. Comparing the random sampling with the different uncertainty measures, we conclude that every uncertainty measure performs better than using random samples to retrain the model. Even though random sampling also increases the classification performance but at a slower rate. As for the uncertainty measure used to retrain the model, aleatoric, epistemic, and total uncertainty obtained similar results on the G12EC dataset. Otherwise, on the PTB-XL dataset, epistemic uncertainty obtained a higher improvement compared to aleatoric and total uncertainty, with the only exception on the first 400 samples of the PTB-XL dataset.

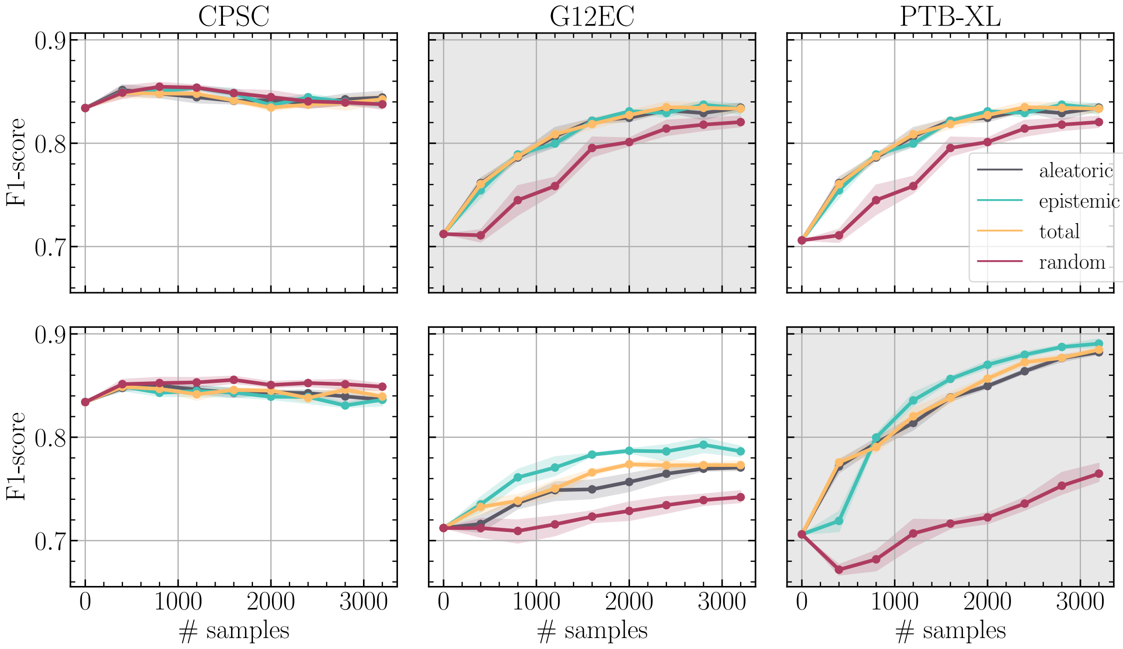


Figure 5.17: Classification performance as a function of the number of samples used to retrain the DeepEnsemble. In the upper plots data from the G12EC dataset was used to retrain the model, and in the lower plots, PTB-XL data was used. The gray background represents the dataset used to retrain the models.

5.4 Discussion

This study addresses the importance of uncertainty quantification in multi-label ECG classification to develop a practical approach suitable for implementation in clinical practice.

The external validation of machine learning models is becoming increasingly important, particularly in the medical domain. Although it offers a more reliable validation compared to internal validation, the results do not necessarily guarantee reliability on their own [156]. Our findings on external validation align with studies in the literature

[156, 189], where models trained in one setting (data from the same source) do not generalize well to other external data sources. Furthermore, incorporating more data sources into the training scheme improves overall performance on both internal and external data sources. However, it still does not guarantee the same level of performance as with internal datasets.

We showed that a trained model that performs well on internal validation (with comparable classification performance with similar studies in literature [166]) might be highly affected when validated on an external set. Different factors, such as concept shift, a low agreement between annotators, or difficulty in handling a mixture of known and unknown medical conditions, can be associated with low performance on external validation. Our results showed a drop of F1-Score from 0.86 to a range between 0.74 and 0.70 on external validation, depending on the dataset used. Besides being from a completely different source, the external datasets included not only the known classes for the model but also a mixture with unknown classes, i.e., since a multi-label setting is being used, a sample can be labeled with a known and an unknown class. In fact, in the external datasets, 50% of samples include unknown classes, and on the remaining 50%, only 20% of samples are not from the Normal class. Therefore, the majority of cardiac pathologies in the external datasets represent a heterogeneous mixture of medical conditions, which can be one of the major reasons for the performance drop. We showed that there is a strong correlation ($r = -0.92$) between class performance drop and the distance between the train and test sets using the Wasserstein distance. Nonetheless, independently of the performance drop reason, the mentioned factors will always be presented after a model is deployed in clinical practice, and appropriate methods must be taken into account to reduce unreliable or inaccurate predictions.

In this context, uncertainty quantification methods serve as a promising approach to assess the level of uncertainty associated with a given prediction and abstain from providing a diagnosis when high uncertainty is present. Various methods and measures exist in the literature for uncertainty quantification; however, their application in multi-label settings is limited or nonexistent [53]. Consequently, we systematically investigate the feasibility of existing methods applied to multi-label ECG classification. For the evaluation of uncertainty measures, our results demonstrated that ensemble-based methods yielded more robust uncertainty estimations compared to single or Bayesian methods. In terms of calibration analysis, MC-Dropout and ensemble methods achieved lower ECE values than the baseline network. Therefore, the uncertainty measures not only provide an assessment of uncertainty but also offer an improved and better-calibrated probability measure. While, to the best of our knowledge, no studies in the literature compare uncertainty methods using a multi-label setting, in single-label ECE analysis scenarios, Vranken et al. [171] obtained similar conclusions. In different application modalities (image, text, categorical), Ovadia et al. [152] conducted a comprehensive comparison of uncertainty methods under dataset shift and also reported better results for ensemble-based methods.

For the uncertainty source, aleatoric uncertainty estimations achieved better results in internal validation, while epistemic uncertainty estimations yielded superior results in external validation in terms of rank-based measures. Regarding OOD detection, ensemble-based methods using epistemic or total uncertainty outperformed other methods, achieving approximately 0.80 AUROC. Surprisingly, the methods designed for OOD detection and with proven results in other studies in the literature [50, 51, 53] obtained poor results in our ECG classification problem. Although OOD and OSR are similar concepts and OOD is often used in literature to represent a broad view of anomaly, outlier, or novelty detection, in our setting, OOD datasets are more related to the OSR problem since samples are composed of classes from the same datasets. For this reason, OSR problems are typically associated with more challenging scenarios where the statistics of unseen classes can be similar to the statistics of known classes in the dataset.

While acknowledging the importance and valuable use of available measures for quantifying uncertainty, our results showed that in external validation, the quality of all uncertainty measures was also affected. This indicates that the available measures are not fully capable of handling the multi-label setting and dataset shift, at least in the context of ECG classification.

While uncertainty evaluation measures are important to compare different uncertainty estimates, they do not take into consideration the real impact of using said measures when implementing new technologies into clinical practice. The notion of uncertainty and the ability to abstain from predicting a sample should be considered key features of any ML model to be used in clinical practice. Although, in the ECG classification field, none or few works address this important concept. In our analysis, we showed that by using such techniques, the ML-based models were able to abstain from predicting samples with high uncertainty, reducing the wrongly classified samples and consequently increasing the overall classification performance. Applying a 15% rejection threshold in the training set leads to more than double the rejection rate in external datasets, along with a 10% increase in classification performance. This indicates that the samples from external datasets indeed have more samples with high uncertainty, and the models are not fully prepared to classify every sample. The high rejection rate could serve as an indicator of dataset shift effects and the need to retrain the models.

In the context of rejection, low-quality data should be thoroughly checked before being used for diagnostic analysis to avoid misclassification. Our exploratory analysis revealed that the reconstruction error of an autoencoder can assist in this detection. Using a data quality assessment dataset, the autoencoder achieved a sensitivity of 72% and a specificity of 99%. Since the datasets used do not contain quality assessment annotations, we applied a statistical test and discovered that the unacceptable samples classified by the autoencoder were statistically different (p-value < 0.01) from the acceptable group in terms of uncertainty values and resulted in lower performances. Although uncertainty-aware models might reject low-quality data, in applications such as active learning, it is crucial to differentiate between low- and high-quality samples with high uncertainty to select the samples for labeling.

In this sense, after deploying a ML model, we need to take into account that the environment where the model is operating is continuously changing, and concept drifts are likely to occur as well as the appearance of unknown medical conditions that can be submitted to the model during testing. In this scenario, and due to the cost associated with the data labeling, it is very important to request an expert annotator to label the most informative samples to the model. Following this reasoning, we showed that an uncertainty estimation is also a viable option for being used as a selection criterion within the active learning concept. After retraining the DeepEnsemble model using the rejected samples with higher uncertainty, the model was able to learn the new data and obtain a similar performance to the internal validation after adding 2000 new samples, approximately.

5.5 Final remarks

In this chapter, we emphasize the crucial role of uncertainty quantification in clinical decision-making, with a specific focus on multi-label classification, a largely overlooked topic in the literature. We use ECG classification as a case study.

As a key contribution, we present the adaptation and evaluation of state-of-the-art uncertainty estimation methods for multi-label classification, which has broad practical applications [149]. Our results demonstrate that uncertainty estimation methods can aid in the machine learning process. However, current methods still have limitations in accurately quantifying uncertainty, particularly in the case of dataset shift. On external validation, a significant decrease in performance was noticed, accompanied by a decline in the quality of uncertainty estimates. Nevertheless, incorporating uncertainty estimates with a classification with rejection option improves the ability to detect such changes. After deploying a ML model, the data may change rapidly due to various reasons, such as a shift in the population, use of different medical equipment, or limited or unrepresentative training data. These changes often occur when new technologies are introduced in clinical practice, and retraining the ML models may become necessary. In such situations, where labeling a large amount of data may be impractical, we demonstrated that using uncertainty estimates as a criterion for sample selection can significantly reduce the number of samples that need to be labeled, and therefore, the frequency of model retraining compared to random sampling. Additionally, it is important to assess the quality of data before using it for classification, not only to prevent misclassifications when models are unable to abstain but also to avoid selecting a sample with high uncertainty to be labeled, which would be of low quality and unsuitable for classification. However, more research is required to assess the feasibility of using uncertainty measures for this purpose.

Despite the fact that uncertainty estimation is a fundamental feature for every ML model to be applied to clinical practice and there is a wealth of research on multi-label ECG analysis, few studies address uncertainty estimations in their methodology. Our aim is to provide novel mechanisms to evaluate the decision under uncertainty among the community regarding the importance of a comprehensive approach to uncertainty estimation for multi-label classification, by highlighting its evaluation not only in small-scale classification tasks but also in terms of robustness against dataset shifts in large-scale tasks. We believe that this work will serve as a crucial stepping stone towards the proper evaluation of uncertainty quantification methods and contribute to advancing this field, ultimately promoting the safe deployment of ML in various applications.

6

Conclusions

The central theme of this research work has been the exploration of methods for deriving measures of uncertainty in machine learning models to enhance their robustness, reliability, and trustworthiness. The use of uncertainty estimations led us to two main directions: firstly, the awareness of uncertainty throughout the entire development process of machine learning models, and secondly, the importance of knowing the associated uncertainty in predictions to support clinical decision-making.

In our work, we have demonstrated that uncertainty estimations can guide AI practitioners to better understand the predictions for reliable decision-making. We have leveraged the outcome from uncertainty quantification to improve the model development process and interpretability. We present empirical evidence that the uncertainty-weighted model combination improves upon unweighted aggregation strategies in terms of performance and helps reduce the complexity of feature-based explanations. The improvement in models' robustness and interpretability opens the doors for the wider adoption of machine learning models in clinical practice.

In this context, we explored the benefits of integrating uncertainty estimation methods into a real-world safety-critical application: a cardiovascular disease diagnosis classification problem using ECG data. In this topic, our research covered crucial scientific topics under the umbrella of uncertainty for AI safety, such as the quality of uncertainty estimation under different validation strategies (in-distribution, out-of-distribution, and dataset shift), the use of uncertainty estimations as a labeling criterion for active learning, and the benefits of classification with a rejection option to minimize machine learning model misclassifications.

This chapter serves as a conclusion to this thesis, highlighting the societal impact of our research, current challenges in evaluating uncertainty estimation methods, the main scientific contributions, and proposing future research directions.

6.1 Broader Impact

Our research focuses on demonstrating the importance of uncertainty quantification in machine learning, with a particular emphasis on clinical decision-making. The complexity and lack of interpretability of machine learning models have hindered their adoption in healthcare [7], leading in some cases to distrust among clinicians [1]. Uncertainty quantification can help improve the interpretability and trustworthiness of models by providing estimates of uncertainty associated with predictions. This ability to derive uncertainty estimates is not only important for improving model adoption but also essential for the safe deployment of machine learning models.

Accurately quantifying uncertainty in models' predictions enables abstaining from providing an output when a high level of uncertainty is present, escalating uncertain decisions to appropriate human decision-makers. This capability is particularly crucial in situations where models encounter unknown classes or adversarial examples. In these cases, models may make unreasonable decisions that could introduce biases and impact the judgment of experts.

Our research focuses on biomedical applications, but we believe that our work is foundational and has the potential to impact a variety of application areas. This approach will benefit fields that try to mitigate the risk in domains with potentially critical consequences and are prone to automation, including healthcare applications (e.g. medical diagnosis or rare disease identification), autonomous driving, robotics, or finance.

Our goal is to promote awareness among the community about the importance of a comprehensive approach to uncertainty estimation. We hope that our work will serve as a crucial stepping stone towards the proper evaluation of uncertainty quantification methods and contribute to advancing this field, ultimately promoting the safe deployment of machine learning in various applications.

6.2 Review of major contributions

The focus of this thesis is on the development of robust and trustworthy machine learning models, with a particular focus on their application in safety-critical domains such as medicine. Accurate uncertainty quantification is crucial in such domains and therefore plays an essential role in our work. As discussed in Chapter 1, this work is organized into three main research topics that span different stages of the machine learning pipeline: 1) uncertainty quantification, 2) uncertainty for model design, and 3) uncertainty for clinical decision making. While these three topics are central to our work, we also developed two Python libraries, Time Series Feature Extraction Library (TSFEL) and Time Series Subsequence Search Library (TSSEARCH), at the beginning of our research. Although these libraries are not directly related to the main research topics, they indirectly contribute to the development of different outcomes.

To summarize the contributions of this work, we have presented Figure 6.1, which highlights the three main topics discussed earlier in the machine learning pipeline. Each topic is accompanied by a list of scientific publications that contributed to that particular topic. It is worth noting that some publications contributed to multiple research topics and are therefore overlapping more than one research topic in the diagram. Each scientific work is represented by an abbreviation, and their full description can be found in subsection 6.2.1. In the following, we describe the main outcomes of this thesis in more detail:

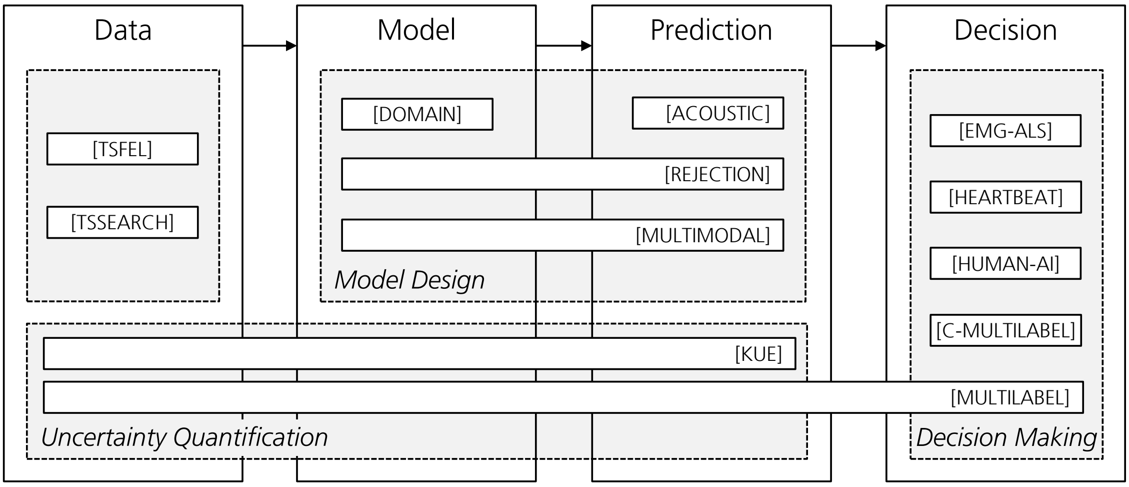


Figure 6.1: Overview of the main contributions of this research project.

Time series libraries: Two open source python libraries were developed, namely TSFEL [190] and TSSEARCH [191].

- TSFEL: Feature extraction is one of the preliminary steps of conventional machine learning pipelines. Quite often, this process ends up being a time-consuming and complex task as data scientists must consider a combination between a multitude of domain knowledge factors and coding implementation. Therefore, we developed TSFEL, which provides support for fast exploratory analysis supported by an automated process of feature extraction on multidimensional time series. We applied TSFEL on various of our preliminary outcomes, such as [101, 192, 193].

- TSSEARCH: The subsequence search is one of the most important subroutines for time series pattern mining. Subsequence search is used across different stages of the machine learning stack. In the initial stages, subsequence search can segment windows of interest, further characterized downstream using feature extraction methods. Furthermore, measuring the distance between a query and the segmented intervals provides quantitative data to perform downstream data mining tasks, such as clustering or supervised classification. In the context of this work, we used TSSEARCH for stride segmentation in the context of HAR tasks and to segment electrocardiography data. Also, an exploratory analysis of how uncertainty in time series can help in the subsequence search tasks was conducted and is available in the library. The subsequence search can therefore be computed using a query sample weights that assigns weights to each point of the query based on its relative uncertainty compared with the other points. For instance, we can search for a query where we are more uncertain about the shape of some intervals than others, thus assigning lower weights to specific intervals.

Uncertainty Quantification: This work explored the most notable state-of-the-art methods for uncertainty quantification in machine learning. The identified gaps in current methods led us to make two main contributions: 1) proposing a new method for knowledge uncertainty quantification and 2) adapting and studying current state-of-the-art methods in a multi-label classification setting.

1. We proposed an agnostic knowledge uncertainty estimation measure named KUE, based on feature-level density estimation of in-distribution data. Our proposed measure has the following properties: 1) Assumes independence between classes, allowing the addition or removal of classes with low cost; 2) Allows the definition of the r^{th} percentile per class for threshold learning; 3) Assumes independence between features; 4) Does not rely on out-of-distribution data for hyperparameter tuning nor threshold selection. The measure and its results were published in [12].

2. We introduced adaptations of single-label uncertainty quantification methods for multi-label classification. We presented a comprehensive comparison of UQ methods in a multi-label setting, focusing on ECG classification scenarios, demonstrating the quality of uncertainty estimations and calibration across various validation scenarios. The obtained results are reported in Chapter 5 and were published in [194].

Uncertainty for model design: The main purpose of this topic was to develop methods to support practitioners in making more informed decisions and develop models that are more transparent and reliable. The specific contributions include: 1) Strategies to help select the most suitable model for a given classification task; 2) Model combination approaches based on uncertainty quantification; 3) Leveraging uncertainty to enhance model interpretability. A comprehensive study of uncertainty-based rejection for enhancing the machine learning development process and interpretability was published in [101] and a novel approach for uncertainty-weighted model fusion was published in [195].

1. We introduced the concept of leveraging uncertainty quantification as a novel criterion for model selection. In addition to choosing the most suitable model for a specific classification task, we further investigated scenarios in which augmenting the dataset with more samples could potentially enhance the quality of the model fit and, consequently, its performance [101]. Within this context, we also explored the

generalization capabilities of various model types using both traditional machine learning approaches and deep learning methods (details can be found in [192]).

2. We propose employing uncertainty estimates to enhance the performance and interpretability of machine learning models. In a preliminary study published in [101], we demonstrated that merging two models based on their respective uncertainty estimates resulted in an improvement in terms of performance-based rejection metrics. In the context of multimodal time series, we proposed employing an uncertainty-weighted model combination technique. This approach enables the model to adapt to the most certain modalities for each instance, outperforming conventional aggregation strategies in terms of performance. The obtained results were published in [195].

3. We developed a visualization tool for classification with rejection to facilitate the interpretation of classifiers' uncertainty during model development and to audit specific decisions. This tool enables practitioners to make better-informed decisions throughout the model development process. The findings were published in [101]. Moreover, we proposed an innovative approach to reduce the explanation complexity of feature-based time series models by reducing the number of modalities and features used for explaining multimodal data, employing the uncertainty-weighted model combination technique (research work published in [195]).

Uncertainty for clinical decision making: The contributions in this topic focused on the practical usefulness of uncertainty estimation methods throughout the entire lifecycle of a deployed machine learning model. A key contribution was addressing the classification with rejection option based on uncertainty estimates, which can aid in decision-making. Although the developed knowledge applies to a wide range of applications, we mainly focused on time series data and biomedical applications, with notable contributions in ECG classification and HAR tasks.

1. A key contribution of our research involved developing various machine learning models with abstaining capabilities, emphasizing the importance of employing classification with rejection options in clinical decision support systems. Published works focusing on uncertainty-based rejection spanned diverse domains, including Amyotrophic Lateral Sclerosis diagnosis using electromyography signals [193], cardiac pathologies classification employing ECG [194, 196], human activity recognition through inertial sensing [101], and quality control using acoustic data in industrial scenarios [197].

2. In the ECG classification application domain, we conducted an in-depth study of uncertainty estimation techniques, exploring their application from the initial stage of the machine learning pipeline to their integration in clinical decision-making. Preliminary findings were published in [196], and an extension of this work was

published in [194]. Additionally, our published works [163, 198] contribute to ECG classification in terms of interpretability and human-AI protocols.

3. In the HAR domain, we published different works that highlighted the use of uncertainty for HAR, including those published in [12, 101, 192]. Moreover, the research detailed in Section 4.2 underscores the importance of uncertainty estimation in multimodal HAR tasks.

6.2.1 List of publications

Throughout the period of this thesis, the work developed has been disseminated through scientific publications. Although the main publications have already been mentioned in this manuscript, we summarize them here and include other co-author works that are also outcomes of this research project.

6.2.1.1 Journal Papers

- **[TSFEL]** - **M. Barandas**, D. Folgado, L. Fernandes, S. Santos, M. Abreu, P. Bota, H. Liu, T. Schultz, H. Gamboa. *TSFEL: Time series feature extraction library.* SoftwareX 11: 100456, 2020.

- **[KUE]** - C. Pires, **M. Barandas**, L. Fernandes, D. Folgado, Hugo Gamboa. *Towards Knowledge Uncertainty Estimation for Open Set Recognition.* Machine Learning and Knowledge Extraction 2(4): 505-532, 2020.

- **[HEARTBEAT]** - I. Neves, D. Folgado, S. Santos, **M. Barandas**, A. Campagner, L. Ronzio, F. Cabitza, Hugo Gamboa. *Interpretable heartbeat classification using local model-agnostic explanations on ECGs.* Computers in Biology and Medicine 133: 104393, 2021.

- **[REJECTION]** - **M. Barandas**, D. Folgado, R. Santos, R. Simão, H. Gamboa. *Uncertainty-Based Rejection in Machine Learning: Implications for Model Development and Interpretability.* Electronics 11(3): 396, 2022.

- **[TSSEARCH]** - D. Folgado, **M. Barandas**, M. Antunes, M. Nunes, H. Liu, Y. Hartmann, T. Schultz, Hugo Gamboa. *TSSEARCH: Time Series Subsequence Search Library* SoftwareX 18: 101049, 2022.

- **[DOMAIN]** - N. Bento, J. Rebelo, **M. Barandas**, A. Carreiro, A. Campagner, F. Cabitza, H. Gamboa. *Comparing Handcrafted Features and Deep Neural Representations for Domain Generalization in Human Activity Recognition.* Sensors 22(19): 7324, 2022.

- **[EMG-ALS]** - M. Antunes, D. Folgado, **M. Barandas**, A. Carreiro, C. Quintão, M. Carvalho, Hugo Gamboa. *A morphology-based feature set for automated Amyotrophic*

Lateral Sclerosis diagnosis on surface electromyography. Biomedical Signal Processing and Control 79: 104011, 2023.

- [**HUMAN-AI**] - F. Cabitza, A. Campagner, L. Ronzio, M. Cameli, G. Mandoli, M. Pastore, L. Sconfienza, D. Folgado, **M. Barandas**, H. Gamboa. *Rams, Hounds and White Boxes: Investigating Human-AI Collaboration Protocols in Medical Diagnosis*. Artificial Intelligence In Medicine: 102506, 2023

- [**MULTIMODAL**] - **M. Barandas**, D. Folgado, L. Famiglini, R. Santos, F. Cabitza, H. Gamboa. *Uncertainty Quantification Meets Explainability: Insights from Model Combination on Multimodal Time Series* Information Fusion 100: 101955, 2023.

- [**MULTILABEL**] - **M. Barandas**, L. Famiglini, A. Campagner, D. Folgado, R. Simao, F. Cabitza, H. Gamboa. *Evaluation of uncertainty quantification methods in multi-label classification: A case study with automatic diagnosis of electrocardiogram.* Information Fusion 101: 101978, 2024.

6.2.1.2 Conference Proceedings

- [**ACOUSTIC**] - M. L. Nunes, **M. Barandas**, H. Gamboa, F. Soares. *Acoustic structural integrity assessment of ceramics using supervised machine learning and uncertainty-based rejection.* ACM SIGKDD Explorations Newsletter 24(2): 105-113, 2022.

- [**C-MULTILABEL**] - R. Simao, **M. Barandas**, D. Belo, H. Gamboa. *Study of Uncertainty Quantification Using Multi-Label ECG in Deep Learning Models.* In Proceedings of the 16th International Joint Conference on Biomedical Engineering Systems and Technologies - BIOSIGNALS: 252-259, 2023.

6.3 Future work

With the increasing integration of artificial intelligence into various aspects of our lives, the importance of understanding and managing uncertainty has never been more crucial. As we continue to push the boundaries of our knowledge, the potential for uncovering novel methodologies and techniques for dealing with uncertainty in machine learning is vast. This new age of uncertainty invites us to not only refine existing approaches but also to venture into uncharted territory, laying the groundwork for the development of more robust, reliable, and interpretable AI systems.

In this final chapter, we explore the current challenges and future research directions in the field of uncertainty in machine learning, as well as outline specific areas of investigation that we plan to pursue in our ongoing quest to better understand and harness the power of uncertainty for the improvement of AI-driven clinical practice.

6.3.1 Current challenges

Even though many advances in uncertainty quantification have been made over the last years, the evaluation of uncertainty estimations in machine learning remains a challenge. In the following, we highlight three main challenges that impact the evaluation of uncertainty estimations: the lack of ground truth uncertainties, the absence of benchmark datasets for evaluating uncertainty estimation methods, and the lack of standardized evaluation protocols.

- **Lack of ground truth**: The empirical evaluation of methods for quantifying uncertainty is a non-trivial problem due to the lack of ground truth uncertainty information. A common approach for indirectly evaluating the predicted uncertainty measures is by accessing their usefulness to improve classification performance. However, this approach may not always provide a complete or accurate picture of the uncertainty associated with machine learning models and their predictions. Additionally, the lack of ground truth data makes it difficult to determine the appropriate level of uncertainty for different applications.

- **Absence of benchmark datasets**: The lack of ground truth uncertainties poses a challenge to the availability of benchmark datasets to evaluate uncertainty estimation methods. While synthetic datasets with out-of-distribution or dataset shifts exist, such as those with added noise, natural adversarial attacks, or unseen classes, these are often limited to image classification or text. Thus, the development of benchmark datasets for evaluating uncertainty estimation in other domains and applications remains a challenge.

- **Lack of standardized evaluation protocol**: There is no standardized approach for evaluating uncertainty estimates. Different methods may have different assumptions, requirements, and limitations, making it difficult to compare and select appropriate uncertainty estimation techniques for different applications. Moreover, there may be inconsistencies in reporting and interpreting uncertainty estimates across different studies, making it challenging to evaluate the effectiveness of different methods.

6.3.2 Specific future research directions

The research presented raises some unresolved issues and opens new questions that we plan to investigate further in future research. In the following paragraphs, we outline some of the ideas that we intend to explore in our future research efforts.

- **Extend KUE validation**: Although we conducted a thorough validation process and compared our uncertainty estimation method with other state-of-the-art methods across almost 30 different scenarios, we only used four datasets for validation.

Therefore, further validation using benchmark datasets is necessary to study the robustness of our approach to different kinds of data. Additionally, we aim to explore more efficient methods to incorporate feature dependencies into our measure and test it on large-scale datasets in future research.

- **Uncertainty based combination**: Regarding the utilization of uncertainty estimates for model combination, we achieved promising results with our proposed approaches. However, we identified several challenges related to soft aggregation methods. In future research, we aim to investigate the impact of individual model calibration on the ensemble's overall performance. Furthermore, we plan to explore more advanced combination strategies that consider both aleatoric and epistemic uncertainties.

- **Uncertainty in multi-label**: In our work, we adapted current uncertainty estimation methods for single label tasks to multi-label settings by assuming independence between classes. However, this assumption may not always hold in real-world scenarios, where dependence between some classes can improve uncertainty estimates. Therefore, there is a need to extend multi-label uncertainty estimation to better represent class dependencies and obtain more accurate uncertainty estimates. Additionally, we validated our approach using an ECG classification problem, but it is important to consider other application domains in future research.

- **Impact on clinical decision-making**: Research on the impact of uncertainty-aware machine learning models on clinical practice has not yet been conducted. However, this research is crucial for the introduction of these models in clinical practice. It is important to address the potential ethical and practical challenges associated with their implementation in future research.

We expect that this research will inspire additional research into the role of uncertainty quantification as a means of improving the machine learning development process and its application as a decision support tool. By adopting a responsible AI strategy, our objective is to raise awareness within the community regarding the need to develop reliable and trustworthy mechanisms that facilitate the integration of AI in safety-critical domains, such as medicine.

Bibliography

[1] B. Kompa, J. Snoek, and A. L. Beam. “Second opinion needed: communicating uncertainty in medical machine learning”. In: *NPJ Digital Medicine* 4.1 (2021), pp. 1–6.

[2] H. A. Haenssle et al. “Man against machine: diagnostic performance of a deep learning convolutional neural network for dermoscopic melanoma recognition in comparison to 58 dermatologists”. In: *Annals of oncology* 29.8 (2018), pp. 1836–1842.

[3] D. Nam et al. “Artificial intelligence in liver diseases: Improving diagnostics, prognostics and response prediction”. In: *JHEP Reports* (2022), p. 100443.

[4] P. Rajpurkar et al. “Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning”. In: *arXiv preprint arXiv:1711.05225* (2017).

[5] E. Hüllermeier and W. Waegeman. “Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods”. In: *Machine Learning* 110.3 (2021), pp. 457–506.

[6] Y. Gal. “Uncertainty in deep learning”. In: *University of Cambridge* 1.3 (2016).

[7] E. Begoli, T. Bhattacharya, and D. Kusnezov. “The need for uncertainty quantification in machine-assisted medical decision making”. In: *Nature Machine Intelligence* 1.1 (2019), pp. 20–23.

[8] A. Malinin et al. “Shifts: A dataset of real distributional shift across multiple large-scale tasks”. In: *arXiv preprint arXiv:2107.07455* (2021).

[9] J. R. Zech et al. “Variable generalization performance of a deep learning model to detect pneumonia in chest radiographs: a cross-sectional study”. In: *PLoS medicine* 15.11 (2018), e1002683.

[10] V.-L. Nguyen, M. H. Shaker, and E. Hüllermeier. “How to measure uncertainty in uncertainty sampling for active learning”. In: *Machine Learning* 111.1 (2022), pp. 89–122.

[11] A. Sadafi et al. “Multiclass deep active learning for detecting red blood cell subtypes in brightfield microscopy”. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2019, pp. 685–693.

[12] C. Pires et al. “Towards Knowledge Uncertainty Estimation for Open Set Recognition”. In: *Machine Learning and Knowledge Extraction* 2.4 (2020), pp. 505–532.

[13] Y. Li, J. Chen, and L. Feng. “Dealing with uncertainty: A survey of theories and practices”. In: *IEEE Transactions on Knowledge and Data Engineering* 25.11 (2012), pp. 2463–2482.

[14] B. Lambert et al. “Trustworthy clinical AI solutions: a unified review of uncertainty quantification in deep learning models for medical image analysis”. In: *arXiv preprint arXiv:2210.03736* (2022).

[15] R. Senge et al. “Reliable classification: Learning classifiers that distinguish aleatoric and epistemic uncertainty”. In: *Information Sciences* 255 (2014), pp. 16–29.

[16] Z. Huang, H. Lam, and H. Zhang. “Quantifying Epistemic Uncertainty in Deep Learning”. In: *arXiv preprint arXiv:2110.12122* (2021).

[17] J. Mukhoti et al. “Calibrating Deep Neural Networks using Focal Loss”. In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle et al. Vol. 33. Curran Associates, Inc., 2020, pp. 15288–15299. URL: https://proceedings.neurips.cc/paper/2020/file/aeb7b30ef1d024a76f21a1d40e30c302-Paper.pdf.

[18] K. R. Varshney and H. Alemzadeh. “On the safety of machine learning: Cyber-physical systems, decision sciences, and data products”. In: *Big data* 5.3 (2017), pp. 246–255.

[19] A. Malinin and M. Gales. “Predictive uncertainty estimation via prior networks”. In: *Advances in neural information processing systems* 31 (2018).

[20] M. Sensoy, L. Kaplan, and M. Kandemir. “Evidential deep learning to quantify classification uncertainty”. In: *Advances in neural information processing systems* 31 (2018).

[21] Y. Gal, R. Islam, and Z. Ghahramani. “Deep bayesian active learning with image data”. In: *International Conference on Machine Learning*. PMLR. 2017, pp. 1183–1192.

[22] A. Shafaei, M. Schmidt, and J. J. Little. “Does Your Model Know the Digit 6 Is Not a Cat? A Less Biased Evaluation of"Outlier"Detectors.” In: (2018).

[23] C. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[24] M. Perello-Nieto et al. “Background Check: A general technique to build more reliable and versatile classifiers”. In: *2016 IEEE 16th International Conference on Data Mining (ICDM)*. IEEE. 2016, pp. 1143–1148.

[25] M. Goldstein and S. Uchida. “A comparative evaluation of unsupervised anomaly detection algorithms for multivariate data”. In: *PloS one* 11.4 (2016), e0152173.

[26] A. Nguyen, J. Yosinski, and J. Clune. “Deep neural networks are easily fooled: High confidence predictions for unrecognizable images”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 427–436.

[27] D. Heaven et al. “Why deep-learning AIs are so easy to fool”. In: *Nature* 574.7777 (2019), pp. 163–166.

[28] R. Roady et al. “Are out-of-distribution detection methods effective on large-scale datasets?” In: *arXiv preprint arXiv:1910.14034* (2019).

[29] C. Geng, S.-j. Huang, and S. Chen. “Recent advances in open set recognition: A survey”. In: *IEEE transactions on pattern analysis and machine intelligence* 43.10 (2020), pp. 3614–3631.

[30] O. Gune et al. “Generalized Zero-shot Learning using Open Set Recognition.” In: *BMVC*. 2019, p. 213.

[31] A. Rocha and S. K. Goldenstein. “Multiclass from binary: Expanding one-versus-all, one-versus-one and ecoc-based approaches”. In: *IEEE Transactions on Neural Networks and Learning Systems* 25.2 (2013), pp. 289–302.

[32] W. J. Scheirer et al. “Toward open set recognition”. In: *IEEE transactions on pattern analysis and machine intelligence* 35.7 (2012), pp. 1757–1772.

[33] W. J. Scheirer, L. P. Jain, and T. E. Boult. “Probability models for open set recognition”. In: *IEEE transactions on pattern analysis and machine intelligence* 36.11 (2014), pp. 2317–2324.

[34] P. R. M. Júnior et al. “Specialized support vector machines for open-set recognition”. In: *arXiv preprint arXiv:1606.03802* (2016).

[35] T. Mensink et al. “Distance-based image classification: Generalizing to new classes at near-zero cost”. In: *IEEE transactions on pattern analysis and machine intelligence* 35.11 (2013), pp. 2624–2637.

[36] P. R. Mendes Júnior et al. “Nearest neighbors distance ratio open-set classifier”. In: *Machine Learning* 106.3 (2017), pp. 359–386.

[37] C. Chow. “On optimum recognition error and reject tradeoff”. In: *IEEE Transactions on information theory* 16.1 (1970), pp. 41–46.

[38] D. M. Tax and R. P. Duin. “Growing a multi-class classifier with a reject option”. In: *Pattern Recognition Letters* 29.10 (2008), pp. 1565–1570.

[39] G. Fumera, F. Roli, and G. Giacinto. “Reject option with multiple thresholds”. In: *Pattern recognition* 33.12 (2000), pp. 2099–2101.

[40] J. Mena, O. Pujol, and J. Vitrià. “Uncertainty-based rejection wrappers for black-box classifiers”. In: *IEEE Access* 8 (2020), pp. 101721–101746.

[41] M. H. Shaker and E. Hüllermeier. "Aleatoric and epistemic uncertainty with random forests". In: *International Symposium on Intelligent Data Analysis*. Springer. 2020, pp. 444–456.

[42] M. H. Shaker and E. Hüllermeier. "Ensemble-based Uncertainty Quantification: Bayesian versus Credal Inference". In: *PROCEEDINGS 31. WORKSHOP COMPUTATIONAL INTELLIGENCE*. Vol. 25. 2021, p. 63.

[43] B. Hanczar. "Performance visualization spaces for classification with rejection option". In: *Pattern Recognition* 96 (2019), p. 106984.

[44] N. Charoenphakdee et al. "Classification with rejection based on cost-sensitive classification". In: *International Conference on Machine Learning*. PMLR. 2021, pp. 1507–1517.

[45] M. Kläs. "Towards identifying and managing sources of uncertainty in AI and machine learning models-an overview". In: *arXiv preprint arXiv:1811.11669* (2018).

[46] A. Campagner, F. Cabitza, and D. Ciucci. "Three-way decision for handling uncertainty in machine learning: A narrative review". In: *International Joint Conference on Rough Sets*. Springer. 2020, pp. 137–152.

[47] A. S. Sambyal, N. C. Krishnan, and D. R. Bathula. "Towards Reducing Aleatoric Uncertainty for Medical Imaging Tasks". In: *arXiv preprint arXiv:2110.11012* (2021).

[48] F. Condessa, J. Bioucas-Dias, and J. Kovačević. "Performance measures for classification systems with rejection". In: *Pattern Recognition* 63 (2017), pp. 437–450.

[49] J. Gawlikowski et al. "A survey of uncertainty in deep neural networks". In: *arXiv preprint arXiv:2107.03342* (2021).

[50] D. Hendrycks et al. "A benchmark for anomaly segmentation". In: (2019).

[51] K. Lee et al. "A simple unified framework for detecting out-of-distribution samples and adversarial attacks". In: *Advances in neural information processing systems* 31 (2018).

[52] W. Liu et al. "Energy-based out-of-distribution detection". In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 21464–21475.

[53] H. Wang et al. "Can multi-label classification networks know what they don't know?" In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 29074–29087.

[54] M. Raghu et al. "Direct uncertainty prediction for medical second opinions". In: *International Conference on Machine Learning*. PMLR. 2019, pp. 5281–5290.

[55] F. T. Liu, K. M. Ting, and Z.-H. Zhou. "Isolation-based anomaly detection". In: *ACM Transactions on Knowledge Discovery from Data (TKDD)* 6.1 (2012), pp. 1–39.

[56] I. Goodfellow, Y. Bengio, and A. Courville. *Deep learning*. MIT press, 2016.

[57] M. M. Breunig et al. "LOF: identifying density-based local outliers". In: *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*. 2000, pp. 93–104.

[58] V. Vovk, A. Gammerman, and G. Shafer. *Algorithmic learning in a random world*. Vol. 29. Springer, 2005.

[59] A. N. Angelopoulos and S. Bates. "A Gentle Introduction to Conformal Prediction and Distribution-Free Uncertainty Quantification". In: *CoRR* abs/2107.07511 (2021). arXiv: 2107.07511. URL: https://arxiv.org/abs/2107.07511.

[60] A. Shafaei, M. Schmidt, and J. J. Little. "A less biased evaluation of out-of-distribution sample detectors". In: *arXiv preprint arXiv:1809.04729* (2018).

[61] A. Malinin, L. Prokhorenkova, and A. Ustimenko. "Uncertainty in gradient boosting via ensembles". In: *arXiv preprint arXiv:2006.10562* (2020).

[62] S. Depeweg et al. "Decomposition of uncertainty in Bayesian deep learning for efficient and risk-sensitive learning". In: *International Conference on Machine Learning*. PMLR. 2018, pp. 1184–1193.

[63] D. J. MacKay. "A practical Bayesian framework for backpropagation networks". In: *Neural computation* 4.3 (1992), pp. 448–472.

[64] R. M. Neal. *Bayesian learning for neural networks*. Vol. 118. Springer Science & Business Media, 2012.

[65] D. M. Blei, A. Kucukelbir, and J. D. McAuliffe. "Variational inference: A review for statisticians". In: *Journal of the American statistical Association* 112.518 (2017), pp. 859–877.

[66] J. R. Anderson and C. Peterson. "A mean field theory learning algorithm for neural networks". In: *Complex Systems* 1 (1987), pp. 995–1019.

[67] G. E. Hinton and D. Van Camp. "Keeping the neural networks simple by minimizing the description length of the weights". In: *Proceedings of the sixth annual conference on Computational learning theory*. 1993, pp. 5–13.

[68] Y. Gal and Z. Ghahramani. "Dropout as a bayesian approximation: Representing model uncertainty in deep learning". In: *international conference on machine learning*. PMLR. 2016, pp. 1050–1059.

[69] C. P. Robert, G. Casella, and G. Casella. *Monte Carlo statistical methods*. Vol. 2. Springer, 1999.

[70] R. M. Neal. *Bayesian training of backpropagation networks by the hybrid Monte Carlo method*. Tech. rep. Citeseer, 1992.

[71] J. Denker and Y. LeCun. "Transforming neural-net output levels to probability distributions". In: *Advances in neural information processing systems* 3 (1990).

[72] B. Lakshminarayanan, A. Pritzel, and C. Blundell. "Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles". In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon et al. Vol. 30. Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper/2017/file/9ef2ed4b7fd2c810847ffa5fa85bce38-Paper.pdf.

[73] A. F. Psaros et al. "Uncertainty quantification in scientific machine learning: Methods, metrics, and comparisons". In: *arXiv preprint arXiv:2201.07766* (2022).

[74] G. Scalia et al. "Evaluating scalable uncertainty estimation methods for deep learning-based molecular property prediction". In: *Journal of chemical information and modeling* 60.6 (2020), pp. 2697–2717.

[75] C. M. Bishop and N. M. Nasrabadi. *Pattern recognition and machine learning*. Vol. 4. 4. Springer, 2006.

[76] E. J. Herron, S. R. Young, and T. E. Potok. "Ensembles of networks produced from neural architecture search". In: *International Conference on High Performance Computing*. Springer. 2020, pp. 223–234.

[77] M. S. A. Nadeem, J.-D. Zucker, and B. Hanczar. "Accuracy-rejection curves (ARCs) for comparing classification methods with a reject option". In: *Machine Learning in Systems Biology*. PMLR. 2009, pp. 65–81.

[78] J. C. Hühn and E. Hüllermeier. "FR3: A fuzzy rule learner for inducing reliable classifiers". In: *IEEE Transactions on Fuzzy Systems* 17.1 (2008), pp. 138–149.

[79] D. Hendrycks and K. Gimpel. "A baseline for detecting misclassified and out-of-distribution examples in neural networks". In: *arXiv preprint arXiv:1610.02136* (2016).

[80] T. Saito and M. Rehmsmeier. "The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets". In: *PloS one* 10.3 (2015), e0118432.

[81] H. Asgharnezhad et al. "Objective evaluation of deep uncertainty predictions for covid-19 detection". In: *Scientific Reports* 12.1 (2022), pp. 1–11.

[82] P. Tabarisaadi et al. "An Optimized Uncertainty-Aware Training Framework for Neural Networks". In: *IEEE transactions on neural networks and learning systems* (2022).

[83] C. Guo et al. "On calibration of modern neural networks". In: *International conference on machine learning*. PMLR. 2017, pp. 1321–1330.

[84] M. P. Naeini, G. Cooper, and M. Hauskrecht. "Obtaining well calibrated probabilities using bayesian binning". In: *Twenty-Ninth AAAI Conference on Artificial Intelligence*. 2015.

[85] P. Izmailov et al. "Subspace inference for Bayesian deep learning". In: *Uncertainty in Artificial Intelligence*. PMLR. 2020, pp. 1169–1179.

[86] B. Zadrozny and C. Elkan. “Obtaining calibrated probability estimates from decision trees and naive bayesian classifiers”. In: *Icml*. Vol. 1. 2001, pp. 609–616.

[87] N. Chakravarti. “Isotonic median regression: a linear programming approach”. In: *Mathematics of operations research* 14.2 (1989), pp. 303–308.

[88] M. P. Naeini and G. F. Cooper. “Binary classifier calibration using an ensemble of near isotonic regression models”. In: *2016 IEEE 16th International Conference on Data Mining (ICDM)*. IEEE. 2016, pp. 360–369.

[89] D. W. Scott. *Multivariate density estimation: theory, practice, and visualization*. John Wiley & Sons, 2015.

[90] M. R. Abbas et al. “Accuracy rejection normalized-cost curves (ARNCCs): A novel 3-dimensional framework for robust classification”. In: *IEEE Access* 7 (2019), pp. 160125–160143.

[91] B. Schölkopf et al. “Estimating the support of a high-dimensional distribution”. In: *Neural computation* 13.7 (2001), pp. 1443–1471.

[92] T.-F. Wu, C.-J. Lin, and R. Weng. “Probability estimates for multi-class classification by pairwise coupling”. In: *Advances in Neural Information Processing Systems* 16 (2003).

[93] A. Bendale and T. Boult. “Towards open world recognition”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 1893–1902.

[94] J. Davis and M. Goadrich. “The relationship between Precision-Recall and ROC curves”. In: *Proceedings of the 23rd international conference on Machine learning*. 2006, pp. 233–240.

[95] W. Morningstar et al. “Density of states estimation for out of distribution detection”. In: *International Conference on Artificial Intelligence and Statistics*. PMLR. 2021, pp. 3232–3240.

[96] Y. Hechtlinger, B. Póczos, and L. Wasserman. “Cautious deep learning”. In: *arXiv preprint arXiv:1805.09460* (2018).

[97] A. Holzinger et al. “Causability and explainability of artificial intelligence in medicine”. In: *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 9.4 (2019), e1312.

[98] L. Oala et al. “Machine Learning for Health: Algorithm Auditing & Quality Control”. In: *Journal of medical systems* 45.12 (2021), pp. 1–8.

[99] S. Ghosh et al. “Uncertainty Quantification 360: A Holistic Toolkit for Quantifying and Communicating the Uncertainty of AI”. In: *arXiv preprint arXiv:2106.01410* (2021).

[100] Y. Chung et al. “Uncertainty toolbox: an open-source library for assessing, visualizing, and improving uncertainty quantification”. In: *arXiv preprint arXiv:2109.10254* (2021).

[101] M. Barandas et al. "Uncertainty-Based Rejection in Machine Learning: Implications for Model Development and Interpretability". In: *Electronics* 11.3 (2022), p. 396.

[102] B. Efron and R. Tibshirani. "Bootstrap methods for standard errors, confidence intervals, and other measures of statistical accuracy". In: *Statistical science* (1986), pp. 54–75.

[103] L. Fischer, B. Hammer, and H. Wersing. "Optimal local rejection for classifiers". In: *Neurocomputing* 214 (2016), pp. 445–457.

[104] D. Anguita et al. "A public domain dataset for human activity recognition using smartphones." In: *Esann*. Vol. 3. 2013, p. 3.

[105] A. Asuncion and D. Newman. *UCI machine learning repository*. 2007.

[106] Z. Bosnić and I. Kononenko. "An overview of advances in reliability estimation of individual predictions in machine learning". In: *Intelligent Data Analysis* 13.2 (2009), pp. 385–401.

[107] A. Tornede et al. "Algorithm selection on a meta level". In: *arXiv preprint arXiv:2107.09414* (2021).

[108] C. Buckley et al. "The role of movement analysis in diagnosing and monitoring neurodegenerative conditions: Insights from gait and postural control". In: *Brain sciences* 9.2 (2019), p. 34.

[109] J. Kittler et al. "On combining classifiers". In: *IEEE transactions on pattern analysis and machine intelligence* 20.3 (1998), pp. 226–239.

[110] M. P. Neto and F. V. Paulovich. "Explainable Matrix-Visualization for Global and Local Interpretability of Random Forest Classification Ensembles". In: *IEEE Transactions on Visualization and Computer Graphics* 27.2 (2020), pp. 1427–1437.

[111] T. Lombrozo. "Explanatory Preferences Shape Learning and Inference". en. In: *Trends in Cognitive Sciences* 20.10 (Oct. 2016), pp. 748–759. ISSN: 13646613. DOI: 10.1016/j.tics.2016.08.001. URL: https://linkinghub.elsevier.com/retrieve/pii/S136466131630105X (visited on 03/11/2023).

[112] R. Greer et al. "Multi-View Ensemble Learning With Missing Data: Computational Framework and Evaluations using Novel Data from the Safe Autonomous Driving Domain". In: *arXiv preprint arXiv:2301.12592* (2023).

[113] M. Mohandes, M. Deriche, and S. O. Aliyu. "Classifiers combination techniques: A comprehensive review". In: *IEEE Access* 6 (2018), pp. 19626–19639.

[114] E. Fersini, F. A. Pozzi, and E. Messina. "Detecting irony and sarcasm in microblogs: The role of expressive signals and ensemble classifiers". In: *2015 IEEE international conference on data science and advanced analytics (DSAA)*. IEEE. 2015, pp. 1–8.

[115] M. Shahhosseini, G. Hu, and H. Pham. “Optimizing ensemble weights and hyperparameters of machine learning models for regression problems”. In: *Machine Learning with Applications* 7 (2022), p. 100251. ISSN: 2666-8270. DOI: https://doi.org/10.1016/j.mlwa.2022.100251. URL: https://www.sciencedirect.com/science/article/pii/S2666827022000020.

[116] N. Poh and J. Kittler. “A unified framework for biometric expert fusion incorporating quality measures”. In: *IEEE transactions on pattern analysis and machine intelligence* 34.1 (2011), pp. 3–18.

[117] S. Chitroub. “Classifier combination and score level fusion: concepts and practical aspects”. In: *International Journal of Image and Data Fusion* 1.2 (2010), pp. 113–135.

[118] A. Tornede et al. “Algorithm selection on a meta level”. In: *Machine Learning* (2022), pp. 1–34.

[119] R. R. Selvaraju et al. “Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization”. In: *2017 IEEE International Conference on Computer Vision (ICCV)*. Venice: IEEE, Oct. 2017, pp. 618–626. ISBN: 978-1-5386-1032-9. DOI: 10.1109/ICCV.2017.74. URL: http://ieeexplore.ieee.org/document/8237336/ (visited on 03/11/2023).

[120] M. Ivanovs, R. Kadikis, and K. Ozols. “Perturbation-based methods for explaining deep neural networks: A survey”. en. In: *Pattern Recognition Letters* 150 (Oct. 2021), pp. 228–234. ISSN: 01678655. DOI: 10.1016/j.patrec.2021.06.030. URL: https://linkinghub.elsevier.com/retrieve/pii/S0167865521002440 (visited on 03/11/2023).

[121] M. T. Ribeiro, S. Singh, and C. Guestrin. “"Why Should I Trust You?": Explaining the Predictions of Any Classifier”. en. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. San Francisco California USA: ACM, Aug. 2016, pp. 1135–1144. ISBN: 978-1-4503-4232-2. DOI: 10.1145/2939672.2939778. URL: https://dl.acm.org/doi/10.1145/2939672.2939778 (visited on 03/11/2023).

[122] S. M. Lundberg and S.-I. Lee. “A Unified Approach to Interpreting Model Predictions”. In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon et al. Vol. 30. Curran Associates, Inc., 2017.

[123] F. Doshi-Velez and B. Kim. “Towards A Rigorous Science of Interpretable Machine Learning”. en. In: *arXiv:1702.08608 [cs, stat]* (Feb. 2017). arXiv: 1702.08608. URL: http://arxiv.org/abs/1702.08608 (visited on 04/07/2019).

[124] J. Zhou et al. “Evaluating the Quality of Machine Learning Explanations: A Survey on Methods and Metrics”. en. In: *Electronics* 10.5 (Mar. 2021), p. 593. ISSN: 2079-9292. DOI: 10.3390/electronics10050593. URL: https://www.mdpi.com/2079-9292/10/5/593 (visited on 03/11/2023).

[125] M. Nauta et al. “From Anecdotal Evidence to Quantitative Evaluation Methods: A Systematic Review on Evaluating Explainable AI”. en. In: *arXiv:2201.08164 [cs]* (Jan. 2022). arXiv: 2201.08164. URL: http://arxiv.org/abs/2201.08164 (visited on 02/21/2022).

[126] A. F. Markus, J. A. Kors, and P. R. Rijnbeek. “The role of explainability in creating trustworthy artificial intelligence for health care: A comprehensive survey of the terminology, design choices, and evaluation strategies”. en. In: *Journal of Biomedical Informatics* 113 (Jan. 2021), p. 103655. ISSN: 15320464. DOI: 10.1016/j.jbi.2020.103655. URL: https://linkinghub.elsevier.com/retrieve/pii/S1532046420302835 (visited on 03/11/2023).

[127] I. Askira-Gelman. “Knowledge discovery: comprehensibility of the results”. In: *Proceedings of the thirty-first Hawaii international conference on system sciences*. Vol. 5. IEEE. 1998, pp. 247–255.

[128] Y. Zhang and B. Wallace. “A Sensitivity Analysis of (and Practitioners’ Guide to) Convolutional Neural Networks for Sentence Classification”. In: *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Taipei, Taiwan: Asian Federation of Natural Language Processing, Nov. 2017, pp. 253–263. URL: https://aclanthology.org/I17-1026.

[129] B. Ustun and C. Rudin. “Supersparse linear integer models for optimized medical scoring systems”. In: *Machine Learning* 102 (2016), pp. 349–391.

[130] K. P. Burnham and D. R. Anderson. “Multimodel inference: understanding AIC and BIC in model selection”. In: *Sociological methods & research* 33.2 (2004), pp. 261–304.

[131] L. Zhao, Q. Hu, and W. Wang. “Heterogeneous feature selection with multi-modal deep neural networks and sparse group lasso”. In: *IEEE Transactions on Multimedia* 17.11 (2015), pp. 1936–1948.

[132] G. Plumb et al. “Regularizing black-box models for improved interpretability”. In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 10526–10536.

[133] S. M. Alghowinem et al. “Interpretation of depression detection models via feature selection methods”. In: *IEEE transactions on affective computing* (2020).

[134] S. L. Buchner. “Multimodal Feature Selection to Unobtrusively Model Trust, Workload, and Situation Awareness”. PhD thesis. University of Colorado at Boulder, 2022.

[135] U. Bhatt, A. Weller, and J. M. F. Moura. “Evaluating and Aggregating Feature-based Model Explanations”. en. In: *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*. Yokohama, Japan: International Joint Conferences on Artificial Intelligence Organization, July 2020, pp. 3016–3022.

ISBN: 978-0-9992411-6-5. DOI: 10.24963/ijcai.2020/417. URL: https://www.ijcai.org/proceedings/2020/417 (visited on 03/11/2023).

[136] R. W. Batterman and C. C. Rice. "Minimal Model Explanations". en. In: *Philosophy of Science* 81.3 (July 2014), pp. 349–376. ISSN: 0031-8248, 1539-767X. DOI: 10.1086/676677. URL: https://www.cambridge.org/core/product/identifier/S0031824800007145/type/journal_article (visited on 03/12/2023).

[137] I. Lage et al. "Human evaluation of models built for interpretability". In: *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*. Vol. 7. 2019, pp. 59–67.

[138] P. Schmidt et al. "Introducing wesad, a multimodal dataset for wearable stress and affect detection". In: *Proceedings of the 20th ACM international conference on multimodal interaction*. 2018, pp. 400–408.

[139] H. Liu, Y. Hartmann, and T. Schultz. "CSL-SHARE: A Multimodal Wearable Sensor-Based Human Activity Dataset". In: *Frontiers in Computer Science* 3 (Oct. 2021), p. 759136. ISSN: 2624-9898. DOI: 10.3389/fcomp.2021.759136. (Visited on 03/29/2023).

[140] S. M. Lundberg et al. "From local explanations to global understanding with explainable AI for trees". In: *Nature Machine Intelligence* 2.1 (2020), pp. 2522–5839.

[141] D. Makowski et al. "NeuroKit2: A Python toolbox for neurophysiological signal processing". In: *Behavior Research Methods* 53.4 (Feb. 2021), pp. 1689–1696. DOI: 10.3758/s13428-020-01516-y. URL: https://doi.org/10.3758%2Fs13428-020-01516-y.

[142] S. Föll et al. "FLIRT: A feature generation toolkit for wearable data". In: *Computer Methods and Programs in Biomedicine* 212 (2021), p. 106461.

[143] M. Yan et al. "Emotion classification with multichannel physiological signals using hybrid feature and adaptive decision fusion". In: *Biomedical Signal Processing and Control* 71 (2022), p. 103235.

[144] A. A. Freitas. "Comprehensible classification models: a position paper". en. In: *ACM SIGKDD Explorations Newsletter* 15.1 (Mar. 2014), pp. 1–10. ISSN: 1931-0145, 1931-0153. DOI: 10.1145/2594473.2594475. URL: https://dl.acm.org/doi/10.1145/2594473.2594475 (visited on 03/18/2023).

[145] J. Huysmans et al. "An empirical evaluation of the comprehensibility of decision table, tree and rule based predictive models". en. In: *Decision Support Systems* 51.1 (Apr. 2011), pp. 141–154. ISSN: 01679236. DOI: 10.1016/j.dss.2010.12.003. URL: https://linkinghub.elsevier.com/retrieve/pii/S0167923610002368 (visited on 03/18/2023).

[146] C. Molnar et al. "General pitfalls of model-agnostic interpretation methods for machine learning models". In: *xxAI-Beyond Explainable AI: International Workshop, Held in Conjunction with ICML 2020, July 18, 2020, Vienna, Austria, Revised and Extended Papers*. Springer. 2022, pp. 39–68.

[147] M. R. Boutell et al. "Learning multi-label scene classification". In: *Pattern recognition* 37.9 (2004), pp. 1757–1771.

[148] X.-Z. Wu and Z.-H. Zhou. "A unified view of multi-label performance measures". In: *international conference on machine learning*. PMLR. 2017, pp. 3780–3788.

[149] G. Tsoumakas and I. Katakis. "Multi-label classification: An overview". In: *International Journal of Data Warehousing and Mining (IJDWM)* 3.3 (2007), pp. 1–13.

[150] F. Rewicki and J. Gawlikowski. "Estimating Uncertainty of Deep Learning Multi-Label Classifications Using Laplace Approximation". In: *IGARSS 2022-2022 IEEE International Geoscience and Remote Sensing Symposium*. IEEE. 2022, pp. 1560–1563.

[151] J.-Y. Jiang et al. "Uncertainty in Extreme Multi-label Classification". In: *arXiv preprint arXiv:2210.10160* (2022).

[152] Y. Ovadia et al. "Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift". In: *Advances in neural information processing systems* 32 (2019).

[153] M.-F. Balcan and A. Blum. "On a theory of learning with similarity functions". In: *Proceedings of the 23rd international conference on Machine learning*. 2006, pp. 73–80.

[154] N. Bousquet. "Diagnostics of prior-data agreement in applied Bayesian analysis". In: *Journal of Applied Statistics* 35.9 (2008), pp. 1011–1029.

[155] W. M. Kouw et al. "Learning an MR acquisition-invariant representation using Siamese neural networks". In: *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*. IEEE. 2019, pp. 364–367.

[156] F. Cabitza et al. "The importance of being external. methodological insights for the external validation of machine learning models in medicine". In: *Computer Methods and Programs in Biomedicine* 208 (2021), p. 106288.

[157] E. Tzeng et al. "Deep domain confusion: Maximizing for domain invariance". In: *arXiv preprint arXiv:1412.3474* (2014).

[158] F. Zhou et al. "Domain generalization with optimal transport and metric learning". In: *arXiv preprint arXiv:2007.10573* (2020).

[159] Z. Shen et al. "Towards out-of-distribution generalization: A survey". In: *arXiv preprint arXiv:2108.13624* (2021).

[160] B. Settles. “Active learning literature survey”. In: (2009).

[161] A. M. Alqudah et al. “ECG heartbeat arrhythmias classification: a comparison study between different types of spectrum representation and convolutional neural networks architectures”. In: *Journal of Ambient Intelligence and Humanized Computing* 13.10 (2022), pp. 4877–4907.

[162] Z. Ahmad et al. “ECG heartbeat classification using multimodal fusion”. In: *IEEE Access* 9 (2021), pp. 100615–100626.

[163] I. Neves et al. “Interpretable heartbeat classification using local model-agnostic explanations on ECGs”. In: *Computers in Biology and Medicine* 133 (2021), p. 104393.

[164] Q. Yao et al. “Multi-class arrhythmia detection from 12-lead varied-length ECG using attention-based time-incremental convolutional neural network”. In: *Information Fusion* 53 (2020), pp. 174–182.

[165] A. H. Ribeiro et al. “Automatic diagnosis of the 12-lead ECG using a deep neural network”. In: *Nature communications* 11.1 (2020), pp. 1–9.

[166] T.-M. Chen et al. “Detection and classification of cardiac arrhythmias by a challenge-best deep learning neural network model”. In: *Iscience* 23.3 (2020), p. 100886.

[167] V. Gupta et al. “A critical review of feature extraction techniques for ECG signal analysis”. In: *Journal of The Institution of Engineers (India): Series B* 102.5 (2021), pp. 1049–1060.

[168] A. Y. Hannun et al. “Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network”. In: *Nature medicine* 25.1 (2019), pp. 65–69.

[169] S. Hong et al. “Practical Lessons on 12-Lead ECG Classification: Meta-Analysis of Methods From PhysioNet/Computing in Cardiology Challenge 2020”. In: *Frontiers in Physiology* (2022), p. 2505.

[170] J. Belen et al. “An uncertainty estimation framework for risk assessment in deep learning-based AFib classification”. In: *2020 54th Asilomar Conference on Signals, Systems, and Computers*. IEEE. 2020, pp. 960–964.

[171] J. F. Vranken et al. “Uncertainty estimation for deep learning-based automated analysis of 12-lead electrocardiograms”. In: *European Heart Journal-Digital Health* 2.3 (2021), pp. 401–415.

[172] A. O. Aseeri. “Uncertainty-aware deep learning-based cardiac arrhythmias classification model of electrocardiogram signals”. In: *Computers* 10.6 (2021), p. 82.

[173] Y. Elul et al. “Meeting the unmet needs of clinicians from AI systems showcased for cardiology with deep-learning–based ECG analysis”. In: *Proceedings of the National Academy of Sciences* 118.24 (2021), e2020620118.

[174] W. Zhang et al. "A Deep Bayesian Neural Network for Cardiac Arrhythmia Classification with Rejection from ECG Recordings". In: *arXiv preprint arXiv:2203.00512* (2022).

[175] V. Jahmunah et al. "Uncertainty quantification in DenseNet model using myocardial infarction ECG signals". In: *Computer Methods and Programs in Biomedicine* 229 (2023), p. 107308.

[176] J. Park et al. "Self-Attention LSTM-FCN model for arrhythmia classification and uncertainty assessment". In: *Artificial Intelligence in Medicine* 142 (2023), p. 102570.

[177] F. Liu et al. "An open access database for evaluating the algorithms of electrocardiogram rhythm and morphology abnormality detection". In: *Journal of Medical Imaging and Health Informatics* 8.7 (2018), pp. 1368–1373.

[178] P. Wagner et al. "PTB-XL, a large publicly available electrocardiography dataset". In: *Scientific data* 7.1 (2020), pp. 1–15.

[179] E. A. P. Alday et al. "Classification of 12-lead ecgs: the physionet/computing in cardiology challenge 2020". In: *Physiological measurement* 41.12 (2020), p. 124003.

[180] K. van der Bijl, M. Elgendi, and C. Menon. "Automatic ECG Quality Assessment Techniques: A Systematic Review". In: *Diagnostics* 12.11 (2022), p. 2578.

[181] L. Kramer, C. Menon, and M. Elgendi. "ECGAssess: A Python-Based Toolbox to Assess ECG Lead Signal Quality". In: *Frontiers in Digital Health* (2022), p. 79.

[182] L. Sathyapriya, L. Murali, and T. Manigandan. "Analysis and detection R-peak detection using Modified Pan-Tompkins algorithm". In: *2014 IEEE International Conference on Advanced Communications, Control and Computing Technologies*. IEEE. 2014, pp. 483–487.

[183] I. Silva, G. B. Moody, and L. Celi. "Improving the quality of ECGs collected using mobile phones: The Physionet/Computing in Cardiology Challenge 2011". In: *2011 Computing in Cardiology*. IEEE. 2011, pp. 273–276.

[184] M. Arjovsky, S. Chintala, and L. Bottou. "Wasserstein generative adversarial networks". In: *International conference on machine learning*. PMLR. 2017, pp. 214–223.

[185] A. Gretton et al. "A kernel two-sample test". In: *The Journal of Machine Learning Research* 13.1 (2012), pp. 723–773.

[186] P. E. McKnight and J. Najab. "Mann-Whitney U Test". In: *The Corsini encyclopedia of psychology* (2010), pp. 1–1.

[187] D. Thissen, L. Steinberg, and D. Kuang. "Quick and easy implementation of the Benjamini-Hochberg procedure for controlling the false positive rate in multiple comparisons". In: *Journal of educational and behavioral statistics* 27.1 (2002), pp. 77–83.

[188] S. S. Sawilowsky. "New effect size rules of thumb". In: *Journal of modern applied statistical methods* 8.2 (2009), p. 26.

[189] E. W. Steyerberg and F. E. Harrell. "Prediction models need appropriate internal, internal–external, and external validation". In: *Journal of clinical epidemiology* 69 (2016), pp. 245–247.

[190] M. Barandas et al. "TSFEL: Time series feature extraction library". In: *SoftwareX* 11 (2020), p. 100456.

[191] D. Folgado et al. "Tssearch: Time series subsequence search library". In: *SoftwareX* 18 (2022), p. 101049.

[192] N. Bento et al. "Comparing Handcrafted Features and Deep Neural Representations for Domain Generalization in Human Activity Recognition". In: *Sensors* 22.19 (2022), p. 7324.

[193] M. Antunes et al. "A morphology-based feature set for automated Amyotrophic Lateral Sclerosis diagnosis on surface electromyography". In: *Biomedical Signal Processing and Control* 79 (2023), p. 104011.

[194] M. Barandas et al. "Evaluation of uncertainty quantification methods in multi-label classification: A case study with automatic diagnosis of electrocardiogram". In: *Information Fusion* 101 (2024), p. 101978.

[195] D. Folgado et al. "Explainability meets uncertainty quantification: Insights from feature-based model fusion on multimodal time series". In: *Information Fusion* 100 (2023), p. 101955.

[196] R. Simão. et al. "Study of Uncertainty Quantification Using Multi-Label ECG in Deep Learning Models". In: *Proceedings of the 16th International Joint Conference on Biomedical Engineering Systems and Technologies - BIOSIGNALS*, INSTICC. SciTePress, 2023, pp. 252–259. ISBN: 978-989-758-631-6. DOI: 10.5220/0011680700003414.

[197] M. L. Nunes et al. "Acoustic structural integrity assessment of ceramics using supervised machine learning and uncertainty-based rejection". In: *ACM SIGKDD Explorations Newsletter* 24.2 (2022), pp. 105–113.

[198] F. Cabitza et al. "Rams, hounds and white boxes: Investigating human-AI collaboration protocols in medical diagnosis". In: *Artificial Intelligence in Medicine* (2023), p. 102506.

[199] C.-S. Ho et al. "Rapid identification of pathogenic bacteria using Raman spectroscopy and deep learning". In: *Nature communications* 10.1 (2019), pp. 1–8.

[200] C. H. Lubba et al. "catch22: CAnonical Time-series CHaracteristics: Selected through highly comparative time-series analysis". In: *Data Mining and Knowledge Discovery* 33.6 (2019), pp. 1821–1852.

[201] A. Phinyomark, P. Phukpattaranont, and C. Limsakul. “Feature reduction and selection for EMG signal classification”. In: *Expert systems with applications* 39.8 (2012), pp. 7420–7431.

[202] A. Greco et al. “cvxEDA: A convex optimization approach to electrodermal activity processing”. In: *IEEE Transactions on Biomedical Engineering* 63.4 (2015), pp. 797–804.

[203] D. Makowski et al. “NeuroKit2: A Python toolbox for neurophysiological signal processing”. In: *Behavior research methods* (2021), pp. 1–8.

A

KUE Experiments: Dataset Details and Additional Results

A.1 Supplementary details for Bacteria dataset

Bacteria dataset from the work of Ho et al. [199] is publicly available at https://github.com/csho33/bacteria-ID. The combinations used for OOD were defined based on the antibiotic treatment for a specific set of bacteria described on the work of Ho et al. The correspondence between antibiotic and bacteria names can be consulted in Table A.1, where each combination antibiotic treatment correspond to the OOD combination in the experiments analysis of KUE measure.

Table A.1: Bacterial classes used as OOD for each antibiotic.

Antibiotic	**Bacteria**
Daptomycin	*Enterococcus faecium*
Caspofungin	*Candida albicans* *Candida glabrata*
Ceftriaxone	*Streptococcus pneumoniae* 1 *Streptococcus pneumoniae* 2
Vancomycin	Methicillin-sensitive *Staphylococcus aureus* 1 Methicillin-sensitive *Staphylococcus aureus* 2 Methicillin-sensitive *Staphylococcus aureus* 3 Methicillin-resistant *Staphylococcus aureus* 1 Methicillin-resistant *Staphylococcus aureus* 2 *Staphylococcus epidermidis* *Staphylococcus lugdunensis*
Ciprofloxacin	*Salmonella enterica*
TZP	*Pseudomonas aeruginosa* 1 *Pseudomonas aeruginosa* 2

Table A.1: *Cont.*

Antibiotic	**Bacteria**
Meropenem	*Klebsiella aerogenes*
	Escherichia coli 1
	Escherichia coli 2
	Enterobacter cloacae
	Klebsiella pneumoniae 1
	Klebsiella pneumoniae 2
	Proteus mirabilis
	Serratia marcescens
Penicillin	*Enterococcus faecalis* 1
	Enterococcus faecalis 2
	Streptococcus sanguinis
	Group A *Streptococcus*
	Group B *Streptococcus*
	Group C *Streptococcus*
	Group G *Streptococcus*

A.2 KUE using different classifiers

The experimental evaluation of KUE method for OOD detection was performed using a Random Forest classifier to provide a fair comparison between methods that required the use of ensemble techniques. Nevertheless, we provide detailed results using a set of classical algorithms, namely KNN, NB, SVM and Logist Regression, for both AUROC and accuracy performance measures. In Table A.2 we report both AUROC and the respective accuracy of each method on different OOD combinations.

Table A.2: AUROC for detecting OOD inputs using KUE method with KDE applied to 4 different classifiers and the correspondent accuracy (ACC) in % of the classifiers on 4 datasets. All values are averages over 10 consecutive repetitions.

		KUE_{KDE}								
		KNN		**NB**		**SVM**		**LR**		**Mean**
	OOD	**AUROC**	**ACC**	**AUROC**	**ACC**	**AUROC**	**ACC**	**AUROC**	**ACC**	**AUROC**
Bacteria	Daptomycin	0.92	79.0	0.89	71.5	0.92	84.1	0.91	83.7	0.91 ± 0.01
	Caspofungin	0.98	76.3	0.99	71.2	0.98	84.3	0.97	83.3	0.98 ± 0.01
	Ceftriaxone	0.80	81.7	0.78	73.5	0.79	86.8	0.83	86.7	0.80 ± 0.02
	Vancomycin	0.83	79.0	0.84	74.1	0.83	81.5	0.86	84.0	0.84 ± 0.01
	Ciprofloxacin	0.73	80.4	0.71	72.4	0.72	82.7	0.71	83.7	0.72 ± 0.01
	TZP	0.88	77.9	0.88	71.2	0.88	83.7	0.88	83.7	0.88 ± 0.00
	Meropenem	0.75	85.7	0.76	78.9	0.75	88.3	0.76	88.9	0.75 ± 0.01
	Penicillin	0.76	79.4	0.76	70.9	0.77	83.2	0.77	84.9	0.76 ± 0.01

Table A.2: *Cont.*

		KUE_{KDE}								
		KNN		**NB**		**SVM**		**LR**		**Mean**
	OOD	**AUROC**	**ACC**	**AUROC**	**ACC**	**AUROC**	**ACC**	**AUROC**	**ACC**	**AUROC**
HAR	Walking	0.69	86.2	0.67	82.1	0.72	89.6	0.71	86.6	0.67 ± 0.02
	Upstairs	0.85	86.9	0.84	85.6	0.87	88.6	0.86	87.1	0.85 ± 0.01
	Downstairs	0.83	84.8	0.82	82.9	0.83	87.9	0.83	85.8	0.83 ± 0.01
	Sitting	0.75	87.3	0.77	88.6	0.76	88.8	0.74	86.6	0.76 ± 0.01
	Standing	0.55	86.6	0.54	89.2	0.54	89.9	0.56	86.0	0.55 ± 0.01
	Laying	0.99	78.0	1.00	76.7	0.99	81.4	0.99	80.1	0.99 ± 0.01
	Stairs	0.89	88.9	0.90	86.1	0.90	91.5	0.90	88.4	0.90 ± 0.00
	Dynamic	1.00	84.5	1.00	81.1	1.00	87.9	0.99	88.4	1.00 ± 0.00
	Static	0.99	79.3	1.00	80.2	0.99	82.0	0.99	83.5	0.99 ± 0.01
Digits	0	0.95	97.4	0.93	78.4	0.96	95.8	0.95	94.2	0.95 ± 0.01
	1	0.66	98.3	0.63	82.1	0.65	96.3	0.65	95.2	0.65 ± 0.01
	2	0.90	97.6	0.83	79.7	0.90	96.4	0.91	94.1	0.88 ± 0.03
	3	0.76	98.1	0.74	80.1	0.76	96.9	0.77	94.5	0.76 ± 0.01
	4	0.95	98.0	0.95	81.7	0.94	95.6	0.94	95.4	0.95 ± 0.01
	5	0.86	97.5	0.81	80.0	0.87	96.3	0.89	95.1	0.86 ± 0.03
	6	0.94	98.0	0.89	77.7	0.92	95.8	0.92	94.9	0.91 ± 0.02
	7	0.92	97.8	0.87	78.2	0.92	96.7	0.92	94.9	0.91 ± 0.02
	8	0.90	98.5	0.87	83.7	0.90	97.3	0.90	95.5	0.89 ± 0.01
	9	0.88	98.3	0.84	82.7	0.88	96.9	0.86	94.9	0.87 ± 0.02
Cardio	Suspect	0.65	80.1	0.64	60.8	0.67	87.1	0.67	84.5	0.66 ± 0.01
	Pathologic	0.83	79.3	0.80	62.1	0.83	87.0	0.82	94.9	0.82 ± 0.01

By the analysis of the results from different classifiers, we notice that AUROC are very similar between the different algorithms. However, algorithms with higher accuracy tend to have also higher AUROC, which makes sense due to the dependency on the classification accuracy of our proposed method.

A.3 AUPR-In and AUPR-Out Results

In Tables A.3 and A.4 we present the detailed results for AUPR-Out and AUPR-In, where in-distribution and out-distribution inputs are specified as negatives and positives, respectively. The conclusions drawn for the AUPR are analogous to the AUROC analysis since we used 50% of in- and out-distribution inputs.

Table A.3: AUPR-Out for detecting OOD test inputs using two variants of KUE (KDE and Gaussian) and other baseline methods on 4 datasets. The Mean and Standard Deviation (SD) over OOD combinations is presented after each dataset. All values are averages over 10 consecutive repetitions.

		AUPR-Out										
	OOD	KUE_{KDE}	KUE_G	$p(\hat{y}\|x)$	$H[p(y\|x)]$	$I[y,h]$	$OCSVM$	SVM^{ovo}	SVM^{ova}	NCM	$OSNN$	IF
Bacteria	Daptomycin	0.86	0.86	0.61	0.66	0.87	0.52	0.87	0.75	0.57	0.59	0.59
	Caspofungin	0.92	0.97	0.40	0.38	0.89	0.98	0.54	0.65	0.40	0.46	0.94
	Ceftriaxone	0.76	0.75	0.76	0.83	0.82	0.53	0.90	0.81	0.83	0.73	0.40
	Vancomycin	0.87	0.87	0.66	0.66	0.82	0.79	0.81	0.72	0.69	0.61	0.81
	Ciprofloxacin	0.66	0.66	0.82	0.89	0.66	0.40	0.85	0.76	0.79	0.71	0.35
	TZP	0.81	0.86	0.68	0.71	0.90	0.73	0.89	0.80	0.85	0.77	0.53
	Meropenem	0.74	0.74	0.81	0.84	0.75	0.48	0.85	0.81	0.80	0.73	0.44
	Penicillin	0.63	0.76	0.70	0.72	0.65	0.67	0.80	0.81	0.70	0.66	0.71
	Mean	0.78	0.81	0.68	0.71	0.80	0.64	0.81	0.76	0.70	0.66	0.60
	SD	0.10	0.09	0.13	0.15	0.09	0.18	0.11	0.05	0.14	0.09	0.20
HAR	Walking	0.59	0.59	0.72	0.71	0.71	0.35	0.69	0.68	0.65	0.68	0.35
	Upstairs	0.80	0.81	0.74	0.79	0.81	0.43	0.64	0.65	0.74	0.66	0.44
	Downstairs	0.78	0.76	0.65	0.63	0.64	0.84	0.54	0.70	0.44	0.65	0.88
	Sitting	0.75	0.72	0.49	0.49	0.70	0.66	0.46	0.43	0.50	0.45	0.64
	Standing	0.50	0.47	0.52	0.58	0.76	0.51	0.47	0.66	0.53	0.45	0.52
	Laying	0.85	0.93	0.36	0.38	0.47	1.00	0.32	0.35	0.71	0.49	1.00
	Stairs	0.83	0.83	0.48	0.52	0.67	0.74	0.33	0.42	0.49	0.42	0.79
	Dynamic	0.02	0.23	0.64	0.69	0.70	0.78	0.52	0.52	0.76	0.88	0.85
	Static	0.65	0.69	0.66	0.66	0.74	0.98	0.37	0.58	0.44	0.78	0.99
	Mean	0.64	0.67	0.58	0.61	0.69	0.70	0.48	0.55	0.58	0.61	0.72
	SD	0.25	0.20	0.12	0.12	0.09	0.22	0.12	0.12	0.12	0.15	0.23
Digits	0	0.84	0.89	0.82	0.82	0.96	1.00	0.93	0.46	0.87	0.96	0.72
	1	0.63	0.72	0.86	0.84	0.79	0.94	0.78	0.62	0.81	0.89	0.60
	2	0.70	0.78	0.86	0.82	0.83	0.99	0.88	0.52	0.87	0.94	0.82
	3	0.68	0.74	0.86	0.80	0.75	0.95	0.81	0.61	0.83	0.93	0.60
	4	0.73	0.85	0.86	0.87	0.96	0.99	0.84	0.61	0.82	0.91	0.91
	5	0.73	0.82	0.90	0.88	0.89	0.98	0.81	0.59	0.85	0.94	0.65
	6	0.80	0.86	0.84	0.84	0.96	0.99	0.94	0.52	0.91	0.95	0.75
	7	0.78	0.86	0.92	0.93	0.86	0.99	0.89	0.49	0.91	0.95	0.84
	8	0.79	0.78	0.96	0.98	0.93	0.97	0.95	0.81	0.90	0.95	0.44
	9	0.80	0.85	0.84	0.82	0.80	0.91	0.87	0.73	0.82	0.95	0.57
	Mean	0.75	0.82	0.87	0.86	0.87	0.97	0.87	0.60	0.86	0.94	0.69
	SD	0.06	0.05	0.04	0.05	0.07	0.03	0.06	0.10	0.04	0.02	0.14
Cardio	Suspect	0.60	0.59	0.42	0.42	0.52	0.67	0.47	0.48	0.41	0.59	0.62
	Pathologic	0.79	0.80	0.42	0.42	0.53	0.98	0.36	0.72	0.41	0.61	0.93
	Mean	0.70	0.70	0.42	0.42	0.52	0.82	0.42	0.60	0.41	0.60	0.78
	SD	0.10	0.11	0.00	0.00	0.01	0.15	0.05	0.12	0.00	0.01	0.16
	Mean	0.72	0.76	0.70	0.71	0.77	0.78	0.70	0.63	0.70	0.73	0.68
	SD	0.16	0.15	0.17	0.17	0.13	0.21	0.21	0.13	0.17	0.18	0.19

Table A.4: AUPR-In for detecting OOD test inputs using two variants of KUE (KDE and Gaussian) and other baseline methods on 4 datasets. The Mean and Standard Deviation (SD) over OOD combinations is presented after each dataset. All values are averages over 10 consecutive repetitions.

		AUPR-In										
	OOD	KUE_{KDE}	KUE_G	$p(\hat{y}\|x)$	$H[p(y\|x)]$	$I[y,h]$	$OCSVM$	SVM^{ovo}	SVM^{ova}	NCM	$OSNN$	IF
Bacteria	Daptomycin	0.93	0.93	0.71	0.71	0.88	0.64	0.91	0.82	0.60	0.63	0.72
	Caspofungin	0.99	0.99	0.52	0.48	0.82	0.98	0.53	0.61	0.47	0.55	0.93
	Ceftriaxone	0.85	0.85	0.86	0.87	0.86	0.51	0.93	0.85	0.83	0.82	0.43
	Vancomycin	0.87	0.87	0.65	0.63	0.73	0.65	0.82	0.76	0.58	0.62	0.66
	Ciprofloxacin	0.79	0.80	0.89	0.92	0.73	0.43	0.90	0.83	0.83	0.80	0.36
	TZP	0.89	0.89	0.82	0.81	0.89	0.58	0.90	0.80	0.85	0.84	0.50
	Meropenem	0.79	0.79	0.90	0.90	0.82	0.51	0.89	0.86	0.86	0.83	0.47
	Penicillin	0.78	0.78	0.72	0.73	0.69	0.53	0.80	0.85	0.64	0.73	0.59
	Mean	0.86	0.86	0.76	0.76	0.80	0.60	0.84	0.80	0.71	0.73	0.58
	SD	0.07	0.07	0.12	0.14	0.07	0.16	0.12	0.08	0.14	0.11	0.17
HAR	Walking	0.74	0.72	0.81	0.82	0.84	0.36	0.77	0.73	0.78	0.77	0.39
	Upstairs	0.88	0.88	0.83	0.85	0.88	0.45	0.76	0.71	0.83	0.76	0.49
	Downstairs	0.87	0.87	0.76	0.75	0.77	0.90	0.59	0.70	0.57	0.74	0.90
	Sitting	0.73	0.75	0.59	0.60	0.68	0.66	0.64	0.54	0.60	0.55	0.63
	Standing	0.55	0.56	0.70	0.72	0.82	0.68	0.68	0.73	0.62	0.50	0.68
	Laying	0.99	1.00	0.36	0.36	0.42	1.00	0.37	0.35	0.86	0.59	1.00
	Stairs	0.92	0.91	0.63	0.65	0.73	0.78	0.50	0.48	0.57	0.51	0.80
	Dynamic	1.00	1.00	0.80	0.81	0.77	0.85	0.67	0.68	0.91	0.94	0.87
	Static	1.00	0.99	0.74	0.74	0.76	0.99	0.44	0.55	0.59	0.83	1.00
	Mean	0.85	0.85	0.69	0.70	0.74	0.74	0.60	0.61	0.70	0.69	0.75
	SD	0.14	0.14	0.14	0.14	0.13	0.21	0.13	0.13	0.13	0.15	0.21
Digits	0	0.94	0.96	0.93	0.94	0.98	1.00	0.96	0.95	0.93	0.98	0.84
	1	0.69	0.85	0.91	0.91	0.86	0.96	0.80	0.92	0.88	0.92	0.66
	2	0.90	0.93	0.92	0.92	0.90	1.00	0.91	0.95	0.92	0.96	0.84
	3	0.79	0.86	0.93	0.91	0.87	0.97	0.88	0.90	0.89	0.97	0.66
	4	0.95	0.96	0.90	0.90	0.96	0.99	0.87	0.96	0.86	0.94	0.95
	5	0.88	0.91	0.91	0.90	0.90	0.98	0.86	0.92	0.90	0.97	0.66
	6	0.93	0.94	0.90	0.90	0.98	0.99	0.96	0.92	0.95	0.97	0.83
	7	0.93	0.94	0.96	0.96	0.93	0.99	0.90	0.93	0.95	0.97	0.88
	8	0.92	0.88	0.97	0.98	0.96	0.98	0.96	0.97	0.93	0.97	0.48
	9	0.90	0.92	0.92	0.90	0.87	0.96	0.92	0.92	0.90	0.97	0.64
	Mean	0.88	0.92	0.92	0.92	0.92	0.98	0.90	0.93	0.91	0.96	0.74
	SD	0.08	0.04	0.02	0.03	0.04	0.01	0.05	0.02	0.03	0.02	0.14
Cardio	Suspect	0.71	0.67	0.39	0.38	0.44	0.80	0.59	0.60	0.39	0.71	0.75
	Pathologic	0.86	0.88	0.40	0.39	0.50	0.98	0.41	0.81	0.41	0.71	0.95
	Mean	0.78	0.78	0.40	0.38	0.47	0.89	0.50	0.70	0.40	0.71	0.85
	SD	0.08	0.10	0.01	0.01	0.03	0.09	0.09	0.11	0.01	0.00	0.10
	Mean	0.86	0.87	0.77	0.77	0.80	0.80	0.76	0.78	0.76	0.79	0.71
	SD	0.10	0.10	0.17	0.18	0.14	0.21	0.18	0.16	0.17	0.16	0.19

A.4 Accuracy Rejection Curves

In this section, we present the accuracy-rejection curves for HAR (Figure A.1), Cardio (Figure A.2) and Digits (Figure A.3) and datasets. In the three figures, the average rejection rates against the average accuracy for our method, total, aleatoric, and epistemic uncertainty are presented. The proposed combination is also shown in black, and the optimal rejection is represented by the dashed line.

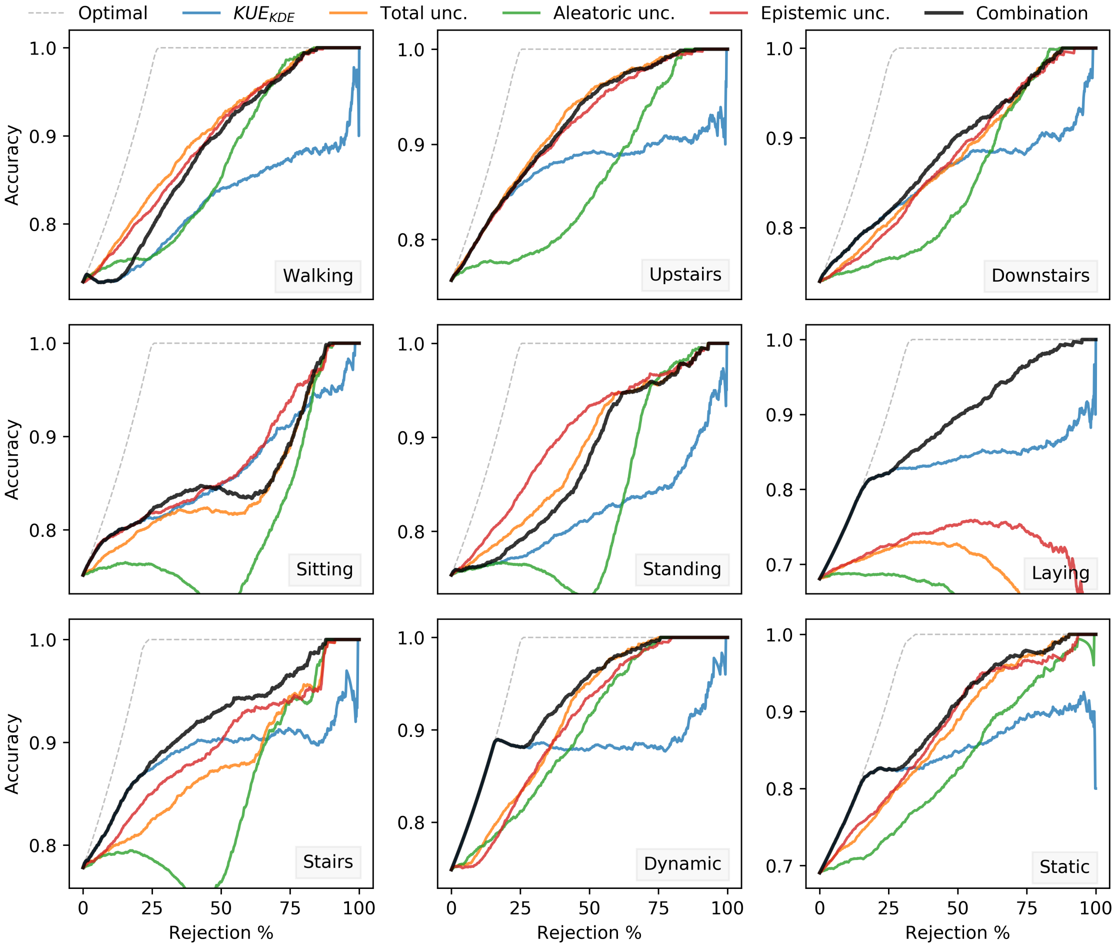


Figure A.1: Accuracy-rejection curves for aleatoric, epistemic, and total uncertainty for HAR dataset. The curve for perfect rejection is included as a baseline. The name in each plot represents the activity used for each OOD input combination.

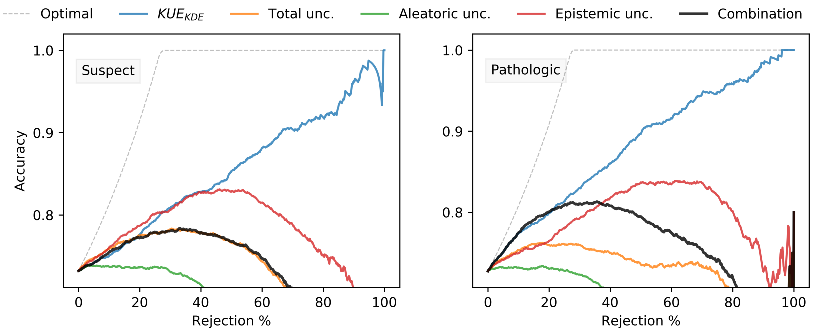


Figure A.2: Accuracy-rejection curves for aleatoric, epistemic, and total uncertainty for the Cardio dataset. The curve for perfect rejection is included as a baseline. The name in each plot represents the OOD input combination.

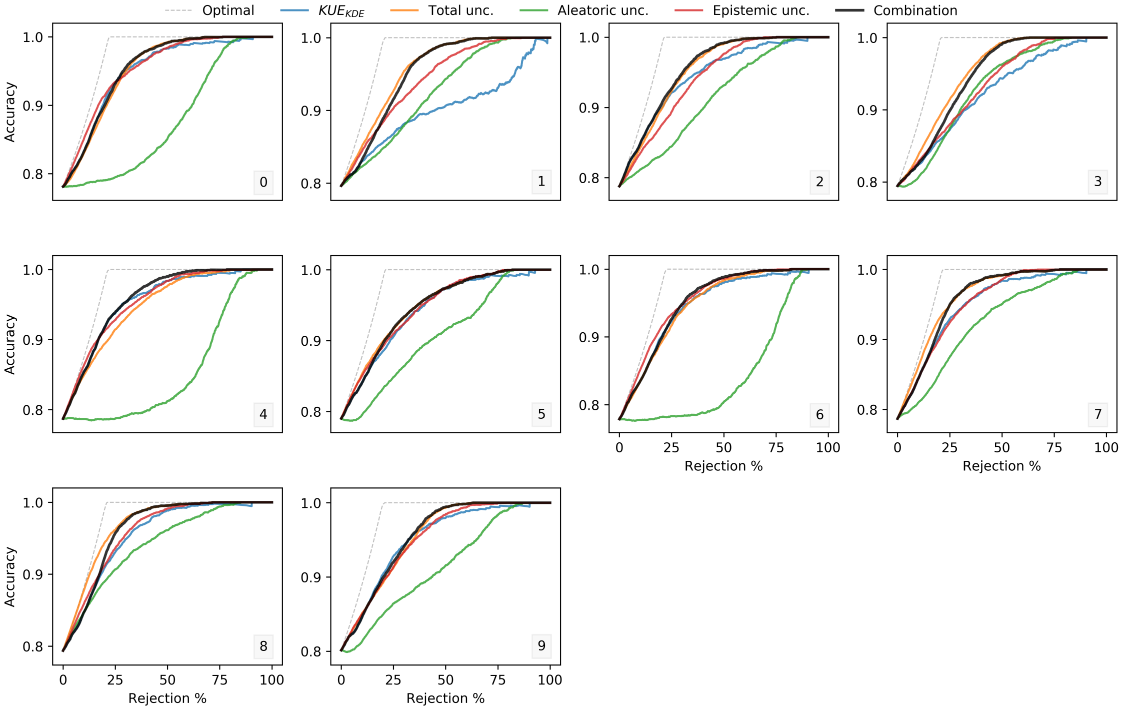


Figure A.3: Accuracy-rejection curves for aleatoric, epistemic, and total uncertainty for the Digits dataset. The curve for perfect rejection is included as a baseline. The name in each plot represents the digits used for each OOD input combination.

B

Multimodal Experiments: Preprocessing and selected models

The proposed approach to lower the explanation complexity of feature-based time series models using an uncertainty-weighted model aggregation strategy was tested in two public datasets composed of multimodal physiological data: WESAD [138] and the CSL-SHARE [139].

The details of signal preprocessing and feature extraction, are provided in Tables B.2 and B.1 for WESAD and CSL-SHARE datasets, respectively. Regarding the selected models for each modality, their respective hyperparameters, and the number of selected features are presented in Tables B.3 and B.4, respectively.

Table B.1: Pre-processing and features extracted for each modality of CSL-SHARE dataset.

Modality	**Pre-processing**	**Feature**
ACC	Magnitude	Catch22 [200] ($D = 114$) Statistical, temporal and spectral domain features of 3-axial accelerometer sensors (2 sensors)
GYRO	Magnitude	Catch22 [200] ($D = 144$) Statistical, temporal and spectral domain features of 3-axial gyroscope sensors (2 sensors)
GONIO	Baseline removal	Catch22 [200] ($D = 38$) Statistical, temporal and spectral domain features of 2-axial goniometer sensor (1 sensor)
EMG	Baseline removal 3th order 10-350 Hz bandpass Butterworth	Phinyomark et al. [201]($D = 80$) Statistical, temporal and spectral domain features of filtered EMG signals (4 sensors)
MIC	Baseline removal	Catch22 [200] ($D = 144$) Statistical, temporal and spectral domain features of 3-axial airborne sensor (1 sensors)

Table B.2: Pre-processing and features extracted for each modality of WESAD dataset.

Modality	**Pre-processing**	**Feature**
ACC	Magnitude	FLIRT [142] ($D = 22$) Statistical and temporal domain features of magnitude signal
EDA	2nd order lowpass Butterworth filter with cutoff of 5 Hz Decomposition in phasic and tonic component using *cvxEDA* [202] Min-Max normalization	FLIRT [142] ($D = 42$) Statistical and temporal domain features of phasic and tonic components
TEMP	Moving average with window size of $2 \times f_s$ Min-Max normalization	($D = 6$) Mean, std; min; max; dynamic range; slope
EMG	Baseline removal 3th order 10-350 Hz bandpass Butterworth	Phinyomark et al. [201] ($D = 19$) Statistical, temporal and spectral domain features of filtered EMG signal
ECG	Interbeat interval	NeuroKit2 [203] ($D = 72$) Time, frequency and non-linear domains of interbeat interval
RESP	2nd order 0.1-0.35 Hz bandpass Butterworth filter Constant detrending	NeuroKit2 [203] ($D = 9$) Statistical domain features of inspiration and expiration cycles.

Table B.3: Selected models for each modality with corresponding hyperparameters and the number of selected features for WESAD dataset. Hyperparameters not referenced were set to the default values of scikit-learn implementation.

Modality	**Classifier**	**Hyperparameters**	**# Features**
ACC	Random Forest	*n_estimators=100* *max_depth=4* *min_samples_split=20* *class_weight='balanced'*	7
EDA	Naive Bayes	-	14
TEMP	Random Forest	*n_estimators=100* *max_depth=3* *min_samples_split=20* *class_weight='balanced'*	4
EMG	SVM	*kernel='sigmoid'* *gamma=0.1* *C=0.1* *class_weight='balanced'*	10
ECG	Random Forest	*n_estimators=100* *max_depth=5* *min_samples_split=20* *class_weight='balanced'*	7
RESP	Random Forest	*n_estimators=100* *max_depth=3* *min_samples_split=20* *class_weight='balanced'*	7
ALL	SVM	*kernel='sigmoid'* *gamma=0.001* *C=0.1 class_weight='balanced'*	49

Table B.4: Selected models for each modality with corresponding hyperparameters and the number of selected features for CSL-SHARE dataset. Hyperparameters not referenced were set to the default values of scikit-learn implementation.

Modality	**Classifier**	**Hyperparameters**	**# Features**
ACC	Random Forest	*n_estimators=200* *max_depth=9* *min_samples_split=30*	23
GYRO	Random Forest	*n_estimators=200* *max_depth=10* *min_samples_split=15*	30
GONIO	Random Forest	*n_estimators=200* *max_depth=10* *min_samples_split=15*	14
EMG	Random Forest	*n_estimators=200* *max_depth=9* *min_samples_split=15*	15
MIC	Random Forest	*n_estimators=200* *max_depth=6* *min_samples_split=40*	8
ALL	Random Forest	*n_estimators=200* *max_depth=10* *min_samples_split=15*	78

C

Multi-label ECG Classification: Annotations details and statistical analysis

C.1 Datasets annotations

The performance differences observed between internal and external datasets, particularly for the NSR (Normal Sinus Rhythm) class of PTB-XL, prompted us to investigate the potential causes of these discrepancies. Upon closer examination, we identified a substantial difference in NSR annotations across the three datasets. We found that CPSC and G12EC datasets do not contain multi-label annotations with NSR class, unlike the PTB-XL dataset that contains 2704 multi-label annotations associated with NSR class. To avoid the differences related to different annotation protocols, the annotations provided by PhysioNet/CinC Challenge 2020 were used. However, originally PTB-XL had both NORM (normal ECG) and SR (Sinus Rhythm) label annotations that were merged and relabeled to NSR (Normal Sinus Rhythm). Contrarily, the CPSC dataset had originally only the Normal label that was relabeled to NSR. For the G12EC dataset, since it was first used on PhysioNet/CinC Challenge 2020, no additional information was found.

Following this finding, we proceed with an evaluation of a subset of the PTB-XL dataset that contains only Normal labels to understand whether the mentioned differences in annotation affected the classification performance. Figure C.1 compares the F1-Score using the entire dataset ($NORM \cup SR$), a subset with Normal class ($NORM$) containing 13932 recordings and the subset without Normal class ($\overline{NORM} \cap SR$) that contains 8544 recordings. The sum of recordings exceeds the number of PTB-XL records because of multi-label annotations per record. As expected, the subset with only Normal classes resulted in a significant improvement in performance across all methods. With this subset, both external validation sets obtained comparable performance. This subset is referred to as PTB-XL* in the subsequent results analysis of Chapter 5.

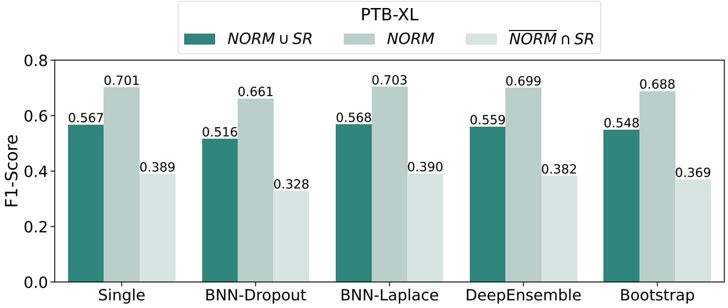


Figure C.1: F1-Score for three subsets of the PTB-XL dataset. $NORM \cup SR$ represents the full dataset, $NORM$ is the subset with Normal Class and $\overline{NORM} \cap SR$ is the subset with only SR annotations.

C.2 Statistical analysis

A statistical analysis was performed to assess the relationship between the distributions of uncertainty values for correctly and incorrectly classified samples. In a multi-label setting, we can consider two scenarios: 1) a label dependence scenario, in which the entire label combination is treated as either correct or incorrect, and 2) a label independence scenario, in which each class is addressed as a separate binary classification problem.

Tables C.1, C.2, C.3 present a statistical analysis comparing the distribution of uncertainty values for correctly and incorrectly classified samples using the Deep Ensemble as an example. The analysis employs the Mann-Whitney U test to assess the differences in the distributions. Before applying Mann-Whitney U test we performed the analysis for validating the assumptions for the two samples t-test. Firstly, the Kolmorov-Smirnov test was performed for normality assumptions (in both scenarios the normality is met). Secondly, we computed the Levene test to evaluate if there was equal variance between the analyzed groups. In this case the equal variance assumption is not met. For this reason we performed the non parametric Mann-Whitney U test. Ultimately, Benjamini-Hochberg correction was applied to the p-values within each dataset. In a multi-label setting, we consider two scenarios: a label dependence scenario, where the entire label combination is either correct or incorrect, and a label independence scenario, where each class is treated as a binary classification problem. Both approaches are included in the table.

Table C.1 reports the statistical analysis for the internal dataset CPSC:

Table C.2 reports the statistical analysis for the external dataset G12EC:

Table C.3 reports the statistical analysis for the external dataset PTB-XL:

Table C.1: Statistical comparison of average uncertainty values for correctly classified and wrongly classified samples using the non-parametric Mann-Whitney U test. P-values, P-values adjusted with Benjamini-Hochberg procedure, and the absolute value of Cohen's d effect sizes (Δ) are shown for each comparison. **Internal CPSC dataset**.

Uncertainty	**Label Independence**			**Label Dependence**		
	p_{value}	$p_{valueadj}$	Δ	p_{value}	$p_{valueadj}$	Δ
p_{max}	$< .001$	$< .001$	1.700	$< .001$	$< .001$	1.485
u_t	$< .001$	$< .001$	1.893	$< .001$	$< .001$	1.344
u_a	$< .001$	$< .001$	1.835	$< .001$	$< .001$	1.283
u_e	$< .001$	$< .001$	1.060	$< .001$	$< .001$	1.022
σ^2	$< .001$	$< .001$	1.490	$< .001$	$< .001$	1.246
vr	$< .001$	$< .001$	1.147	$< .001$	$< .001$	1.292

Table C.2: Statistical comparison of average uncertainty values for correctly classified and wrongly classified samples using the non-parametric Mann-Whitney U test. P-values, P-values adjusted with Benjamini-Hochberg procedure, and the absolute value of Cohen's d effect sizes (Δ) are shown for each comparison. **External G12EC dataset**.

Uncertainty	**Label Independence**			**Label Dependence**		
	p_{value}	$p_{valueadj}$	Δ	p_{value}	$p_{valueadj}$	Δ
p_{max}	$< .001$	$< .001$	1.566	$< .001$	$< .001$	1.100
u_t	$< .001$	$< .001$	1.689	$< .001$	$< .001$	0.977
u_a	$< .001$	$< .001$	1.584	$< .001$	$< .001$	0.849
u_e	$< .001$	$< .001$	1.050	$< .001$	$< .001$	0.906
σ^2	$< .001$	$< .001$	1.442	$< .001$	$< .001$	1.062
vr	$< .001$	$< .001$	1.184	$< .001$	$< .001$	1.117

Table C.3: Statistical comparison of average uncertainty values for correctly classified and wrongly classified samples using the non-parametric Mann-Whitney U test. P-values, P-values adjusted with Benjamini-Hochberg procedure, and the absolute value of Cohen's d effect sizes (Δ) are shown for each comparison. **External PTB-XL dataset**.

Uncertainty	**Label Independence**			**Label Dependence**		
	p_{value}	$p_{valueadj}$	Δ	p_{value}	$p_{valueadj}$	Δ
p_{max}	$< .001$	$< .001$	1.409	$< .001$	$< .001$	0.877
u_t	$< .001$	$< .001$	1.503	$< .001$	$< .001$	0.762
u_a	$< .001$	$< .001$	1.439	$< .001$	$< .001$	0.672
u_e	$< .001$	$< .001$	0.965	$< .001$	$< .001$	0.736
σ^2	$< .001$	$< .001$	1.310	$< .001$	$< .001$	0.875
vr	$< .001$	$< .001$	1.050	$< .001$	$< .001$	0.936