

TD

Hardware and Software Platforms to Deploy and Evaluate Non-Intrusive Load Monitoring Systems

DOCTORAL THESIS

Amâncio Lucas de Sousa Pereira

DOCTORATE IN INFORMATICS ENGINEERING

SPECIALTY: SOFTWARE ENGINEERING



UNIVERSIDADE da MADEIRA

A Nossa Universidade

www.uma.pt

October | 2016

FCT Fundação para a Ciência e a Tecnologia

MINISTÉRIO DA CIÊNCIA, TECNOLOGIA E ENSINO SUPERIOR

Hardware and Software Platforms to Deploy and Evaluate Non-Intrusive Load Monitoring Systems

DOCTORAL THESIS

Amâncio Lucas de Sousa Pereira

DOCTORATE IN INFORMATICS ENGINEERING
SPECIALTY: SOFTWARE ENGINEERING

SUPERVISOR

Duarte Nuno Jardim Nunes

CO-SUPERVISOR

Mario E. Bergés González

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other University. This dissertation is the result of my own work and includes nothing, which is the outcome of work done in collaboration, except where specifically indicated in the text. This dissertation contains less than 200 pages, excluding indices, bibliography and appendixes.

Lucas Pereira

2016

Abstract

The work in this PhD thesis addresses the practical implications of deploying and testing Non-Intrusive Load Monitoring (NILM) and eco-feedback solutions in real-world scenarios. The contributions to this topic are centered around the design and development of NILM frameworks that have been deployed in the wild, supporting long-term research in eco-feedback and also serving the purpose of producing real-world datasets and furthering the state of the art regarding the performance metrics used to evaluate NILM algorithms.

This thesis consists of three main parts: i) the development of tools and datasets for NILM and eco-feedback research, ii) the design, implementation and deployment of NILM and eco-feedback technologies in real world scenarios, and iii) an experimental comparison of performance metrics for event detection and event classification algorithms.

In the first part we describe the Energy Monitoring and Disaggregation Data Format (EMD-DF) and the SustData and SustDataED public datasets.

In second part we discuss the development and deployment of two hardware and software platforms in real households, to support eco-feedback research. We then report on more than five years of experience in deploying and maintaining such platforms. Our findings suggest that the main practical issues can be divided in two categories, *technological* (e.g., system installation) and *social* (e.g., maintaining a steady sample throughout the whole study).

In the final part of this thesis we analyze experimentally the behavior of a number of performance metrics for event detection and event classification, identifying clusters and relationships between the different measures. Our results evidence some considerable differences in the behavior of the performance metrics when applied to the different problems.

Keywords: NILM, Event-Based, Eco-Feedback, Performance evaluation, Platforms, Real world scenarios

Resumo

O trabalho desenvolvido nesta tese de doutoramento aborda as implicações práticas da instalação e avaliação de soluções de monitorização não intrusiva de cargas elétricas (NILM) e eco-feedback em cenários reais. As contribuições para este tópico estão centradas em torno da concepção e desenvolvimento de plataformas NILM que foram instaladas em ambientes não controlados, suportando a pesquisa de longo termo em eco-feedback e servindo também o propósito de produzir conjuntos de dados científicos, bem como promover o avanço do estado da arte acerca das métricas de desempenho utilizadas para avaliar algoritmos NILM.

Esta tese é constituída por três partes principais: i) o desenvolvimento de ferramentas e conjuntos de dados científicos para investigação em NILM e eco-feedback, ii) a concepção, desenho e instalação de tecnologias NILM e eco-feedback em cenários reais, e iii) uma comparação experimental de métricas de desempenho para algoritmos de detecção e de classificação de eventos.

Na primeira parte descrevemos o *Energy Monitoring and Disaggregation Data Format* (EMD-DF) e os conjuntos de dados científicos SustData e SustDataED.

Na segunda parte discutimos o desenvolvimento e instalação de duas plataformas de hardware e software em residências atuais com a finalidade de suportar a investigação em eco-feedback. Aqui, reportamos sobre mais de cinco anos de experiência na instalação e manutenção destes sistemas. Os nossos resultados sugerem que as principais implicações práticas podem ser divididas em duas categorias, físicas (e.g., instalação do sistema) e sociais (e.g., manter uma amostra constante ao longo de todo o estudo).

Na terceira parte analisamos experimentalmente o comportamento de uma série de métricas de desempenho quando estas são utilizadas para avaliar algoritmos de detecção e de classificação de eventos. Calculamos as correlações lineares e não lineares entre os vários pares de métricas, e com base nesses valores procuramos agrupar as métricas que evidenciam

um comportamento semelhante. Os nossos resultados sugerem a existência de diferenças evidentes no comportamento das métricas quando aplicadas a ambos dos problemas.

Palavras-chave: NILM, Baseado-em-eventos, Eco-Feedback, Avaliação de performance, Plataformas, Ambientes reais

Thesis Contents

Thesis Contents	ix
List of Figures	xi
List of Tables	xv
Chapter 1 Introduction	1
1.1 Context	1
1.2 Motivation	4
1.3 Problem Statement	6
1.4 Thesis Outline	7
Chapter 2 Non-Intrusive Load Monitoring	9
2.1 Seminal Work	9
2.2 Event-Based and Event-Less Approaches	10
2.3 Existing Challenges	13
2.4 Literature Review on Load Disaggregation	15
2.5 Literature Review on Performance Evaluation	24
Chapter 3 Research Scope	41
3.1 Research Questions	41
3.2 Research Method	43
3.3 Research Contributions	45
3.4 Publications	48
Chapter 4 Tools and Datasets	53
4.1 EMD-DF: Energy Monitoring and Disaggregation Data Format	53
4.2 SustData: A Public Dataset for Electric Energy Research	71
4.3 SustDataED: A Public Dataset for Electric Energy Disaggregation Research	80
Chapter 5 NILM Deployments in Real World Scenarios	89
5.1 Introduction	90
5.2 Research Platform Overview	93
5.3 Practical Deployment Considerations	105
5.4 Conclusion	119

Chapter 6	Experimental Comparison of Performance Metrics for Event Detection and Classification Algorithms	125
6.1	Algorithms	126
6.2	Datasets	143
6.3	Performance Metrics	146
6.4	Experimental Design.....	152
6.5	Analysis of Results	168
6.6	Conclusion	187
Chapter 7	Conclusion and Future Work	193
7.1	Chapter Summaries	193
7.2	Future Work	194
Bibliography	199
Appendix A	Background Research.....	A-1
A.1	A Survey of Smart-Meters	A-1
A.2	A Survey of Public Household Energy Datasets	A-13
Appendix B	Research Datasets.....	B-1
B.1	UK-DALE.....	B-1
B.2	BLUED	B-3
B.3	PLAID	B-5
Appendix C	Performance Metrics	C-1
C.1	Event Detection.....	C-1
C.2	Event Classification	C-9
Appendix D	Additional Tables	D-1
D.1	Parameter and Feature Sweep Lookup Tables.....	D-1
D.2	Performance Metrics Pairwise Correlations	D-6
Appendix E	Additional Resources	E-1
E.1	Power Calculations	E-1

List of Figures

Figure 1.1 – Total electricity consumption by end-use sector: 2010 (left), 2040 (right).....	2
Figure 2.1 – Example of event-based energy disaggregation	12
Figure 2.2 – Example of event-less energy disaggregation	12
Figure 2.3 – General workflow for event-based NILM approaches	16
Figure 4.1 – EMD-DF: Data model overview	54
Figure 4.2 - RIFF chunk format definition	55
Figure 4.3 – RIFF-WAVE file format chunk structure.....	57
Figure 4.4 - Example of an Appliance Activity Annotation	62
Figure 4.5 - Markers for the refrigerator events in the P & Q file at 60 Hz (top), and one marker at the I & V file at 12 kHz (bottom)	63
Figure 4.6 - Example of user activity annotation.....	64
Figure 4.7 - Example of a Local Metadata Annotation	65
Figure 4.8 – Appliances custom metadata annotation	66
Figure 4.9 - User activities custom metadata annotation	66
Figure 4.10 – NILM Metadata project custom chunk.....	67
Figure 4.11 – Real power transient of a microwave turning <i>ON</i>	76
Figure 4.12 – Aggregate consumption vs. the sum of the individual appliances summarized by day and hour.....	86
Figure 4.13 – EMD-DF configuration information: raw voltage and current (left), processed waveforms (right).....	87
Figure 5.1 - Energy monitoring and eco-feedback research platform overview	91
Figure 5.2 – Sensing hardware: split-core current sensor (left), voltage transformer (center) and TRS splitter connectors (right).....	94
Figure 5.3 – Current and voltage sensors installed in the main power feed	95
Figure 5.4 – Energy eco-feedback is provided on-site using the netbooks’ built-in LCD screen	96
Figure 5.5 – Eco-feedback interfaces used in deployment one: version 1 (left), version 2 (right)	97

Figure 5.7 – General overview of the single-house version of the energy monitoring platform	98
Figure 5.8 – Multi-house platform installation: current sensors (left), voltage sensors and DAQ (right)	99
Figure 5.9 – Energy eco-feedback applications used in deployment two: energy awareness mode (left), detailed consumption mode (right)	100
Figure 5.10 -	101
Figure 5.11 – Example of a possible configuration of the multi-house energy monitoring platform	102
Figure 5.12 – Research platforms deployment timeline	103
Figure 5.13 – The energy monitors are attached to the main fuse door with sticky back Velcro straps	103
Figure 5.14 – Major milestones of deployment one (top); active installations over time (bottom)	104
Figure 5.15 – Multi-port DAQs installed in the main electric panel of one of the buildings	104
Figure 5.16 – Major milestones of deployments two and three (top); active installations over time (bottom)	105
Figure 5.17 – MySQL vs. MongoDB: database physical size	108
Figure 5.18 – MySQL vs. MongoDB: average query time	109
Figure 5.19 – Single- vs. Multi-House: Hardware costs associated with monitoring one house	113
Figure 5.20 – Single- vs. Multi-House vs. Multiple Sensors: Hardware costs associated with monitoring one house	115
Figure 5.21 – Differences in hardware costs projected up to 5000 houses	115
Figure 5.22 – Estimated energy costs of different energy monitoring solutions after one year	117
Figure 5.23 – Proportion of aggregate to power event data for one house after one year	119
Figure 6.1 – Illustration of the MEH event detection process. Real power and absolute power changes (top), threshold filter (center), elapsed time filter (bottom)	129
Figure 6.2 – Illustration of the LLD event detection process. Active power and detection statistic (top), voting procedure (center), and votes filtered by threshold (bottom)	133
Figure 6.3 – Illustration of the SLLD event detection process. Active power and detection statistics (top), detection statistics and local maxima (center), active power and power events (bottom)	136
Figure 6.4 – Flowchart of the algorithm used to create the contingency table of the event detection algorithms	163
Figure 6.5 – List of metric pairs with pairwise correlations above 0.9 in at least of one the coefficients	174
Figure 6.6 – Dendrograms showing ranks (left) and linear (right) correlations of the performance metrics across datasets	174

Figure 6.7 – SLLD _{Max} (UK-DALE – H1): Precision and Recall based metrics sorted in ascending order of detected events (line series). Top 10 models selected by the each metric (column series).....	176
Figure 6.8 – SLLD _{Max} (BLUED – A): Precision and Recall based metrics sorted in ascending order of detected events (line series). Top 10 models selected by the each metric (column series).	177
Figure 6.9 – Dendrograms showing ranks (left) and linear (right) correlations of the micro-average performance metrics across datasets	183
Figure 6.10 – TOP 5 models selected by the micro-average and probabilistic metrics.....	184
Figure 6.11 – Dendrograms showing ranks (left) and linear (right) correlations of the unweighted macro-average performance metrics across datasets	185
Figure 6.12 - Dendrograms showing ranks (left) and linear (right) correlations of the weighted macro-average performance metrics across datasets	185
Figure 7.1 –Single sensor (Left) and Multiple sensor (Right)	A-3
Figure 7.2 - The Energy Detective smart-meter solution: Packaged hardware (Left) and installation in the main breaker box (Right).	A-4
Figure 7.3 - EnerSure branch circuit power meter: Metering unit (Left) and installation in the main breaker box (Right).	A-5
Figure 7.4 - Multiple sensor smart-meters: Belkins Conserve (Left) and P3 Internationals Kill-a-Watt CO ₂ Wireless (Right)	A-8
Figure 7.5 – UK-DALE 1: Distribution of power events in terms of the absolute active power change; all events (left); events below 100 Watts (right)	B-1
Figure 7.6 – UK-DALE 1: Boxplot showing the distribution of the power events according to the absolute power change.	B-2
Figure 7.7 – UK-DALE 2: Distribution of power events in terms of the absolute active power change; all events (left); events below 100 Watts (right)	B-2
Figure 7.8 – UK-DALE 2: Boxplot showing the distribution of the power events according to the absolute power change.	B-3
Figure 7.9 – BLUED A: Distribution of power events in terms of the absolute active power change; all events (left); events below 100 Watts (right)	B-3
Figure 7.10 – BLUED A: Boxplot showing the distribution of the power events according to the absolute power change.	B-4
Figure 7.11 – BLUED B: Distribution of power events in terms of the absolute active power change; all events (left); events below 100 Watts (right)	B-4
Figure 7.12 – BLUED 2: Boxplot showing the distribution of the power events according to the absolute power change.	B-5
Figure 7.13 – PLAID: Appliance instances distribution in each dataset partition.....	B-6
Figure 7.14 – PLAID: Proportion appliance instances in each dataset partition	B-7

List of Tables

Table 2.1 – Appliance types definitions and examples.....	11
Table 2.2 – Overview of public household energy datasets	27
Table 2.3 – Performance metrics for NILM approaches. (EB: Event-Based; EL: Event-Less)	36
Table 4.1 – List of chunks that compose the RIFF-WAVE file format.....	58
Table 4.2 - List of chunks that compose EMD-DF.....	60
Table 4.3 - List of RIFF metadata chunks	67
Table 4.4 – List of functions available to create and maintain EMD-DF based datasets	68
Table 4.5 - Household demographic features	72
Table 4.6 – Summary of household demographics.....	72
Table 4.7 – Household member demographic features	73
Table 4.8 - Summary of household member demographics	73
Table 4.9 – Energy consumption measurements	74
Table 4.10 – Summary of the energy consumption data	75
Table 4.11 – Power event measurements.....	75
Table 4.12 - Power events summary	76
Table 4.13 – User event features.....	77
Table 4.14 – Summary of user events.....	77
Table 4.15 - Environmental data measurements.....	78
Table 4.16 - Electric energy production measurements.....	78
Table 4.17 – Summary of the individual appliance and occupancy data that can be found in SustDataED.....	84
Table 5.1 – Baseline hardware costs of the single- and multi-house energy monitors.....	112
Table 5.2 - Baseline hardware costs of the single- and multi-house energy monitors	114
Table 5.3 – Estimated energy costs of the components that compose the monitoring solutions	116
Table 5.4 – Projected amount of aggregate data that will be generated in one month and one year.....	118

Table 5.5 – Projected amount of power event data that will be generated after one month and one year	118
Table 6.1 – Event detection algorithms to be evaluated	126
Table 6.2 – Parameter space of Meehan s et al. expert heuristic event detector	128
Table 6.3 – Parameter space for the Log Likelihood Ratio event detector.....	131
Table 6.4 – Parameter space of the voting algorithm used in the LLR event detector	132
Table 6.5 – Parameter space of the maxima algorithm used in the SLLR detector.....	135
Table 6.6 – Selected classification algorithms.....	137
Table 6.7 – Summary of the datasets used to evaluate detection algorithms	144
Table 6.8 - Summary of the active power change and elapsed time between power events in the event detection datasets.....	145
Table 6.9 – Datasets used for event classification	145
Table 6.10 - Summary of performance metrics for event detection	147
Table 6.11 – Summary of rank metrics for event detection algorithms.....	148
Table 6.12 – Summary of domain specific metrics for event detection algorithms	149
Table 6.13 – Confusion matrix based metrics for event classification	150
Table 6.14 – Summary of rank metrics for event classification	151
Table 6.15 – Summary of probabilistic metrics for event classification	152
Table 6.16 – List of possible pairwise correlations per metric types.....	154
Table 6.17 – Parameter ranges for Meehan Expert Heuristic event detector	157
Table 6.18 – Parameter ranges for Log-Likelihood and Simplified Log-Likelihood detectors with <i>voting activation</i>	158
Table 6.19 – Parameter ranges for Log-Likelihood and Simplified Log-Likelihood detectors with <i>maxima activation</i>	159
Table 6.20 – Number of different models that will be evaluated across datasets.....	160
Table 6.21 – Tolerance values for event detection evaluation.....	161
Table 6.22 – List of the parameters that will be switched in each classification algorithm and respective values	164
Table 6.23 – Different feature sets used in the event classification algorithms	167
Table 6.24 – Rank (bottom-left) and linear (top-right) correlation results for all four datasets	170
Table 6.25 – Rank and linear correlations averaged by metric for the four event detection datasets	171
Table 6.26 – Clusters formed after cutting the dendrograms of the cross dataset non-linear and linear correlations	175
Table 6.27 – Micro average metrics: rank (bottom-left) and linear (top-right) correlation results for all datasets	179
Table 6.28 – Unweighted macro average metrics: rank (bottom-left) and linear (top-right) correlation results for all datasets	180

Table 6.29 - Weighted macro average metrics: rank (bottom-left) and linear (top-right) correlation results for all datasets	181
Table 6.30 – Rank and linear correlations averaged by metric for all datasets	182
Table 6.31 – Unweighted macro-average clusters	186
Table 6.32 – Weighted macro-average clusters	186
Table 7.1 – Shortlist of single sensor smart-meter alternatives	A-6
Table 7.2 – Shortlist of multiple sensor smart-meter alternatives	A-11
Table 7.3 – PLAID: Appliance instances distribution in each dataset partition	B-5
Table 7.4 – Different parameter configurations for the MEH algorithm (50 Hz and 60 Hz datasets).....	D-2
Table 7.5 – Different parameter combinations for the LLD algorithm (50 Hz datasets)	D-3
Table 7.6 – Different parameter combinations for the LLD algorithm (60 Hz datasets)	D-4
Table 7.7 – Different parameter configurations for the SLLD algorithm (50 Hz and 60 Hz datasets).....	D-5
Table 7.8 – Different parameter and feature configurations for the six event classification algorithms	D-5
Table 7.9 – Different parameters and possible values for each of the six classification algorithms	D-6
Table 7.10 – Different feature combinations for the six classification algorithms.....	D-6
Table 7.11 – MEH event detector: cross dataset pairwise correlations	D-7
Table 7.12 – SLLD _{Max} event detector: cross dataset pairwise correlations.....	D-7
Table 7.13 – LLD _{Max} event detector: cross dataset pairwise correlations	D-8
Table 7.14 – SLLD _{Vote} event detector: cross dataset pairwise correlations	D-8
Table 7.15 – LLD _{Vote} event detector: cross dataset pairwise correlations.....	D-9
Table 7.16 – K-NN classifier: cross dataset pairwise correlations for micro-average metrics	D-9
Table 7.17 – KStar classifier: cross dataset pairwise correlations for micro-average metrics	D-10
Table 7.18 – LWL with Naïve Bayes classifier: cross dataset pairwise correlations for micro-average metrics	D-10
Table 7.19 – Decision Trees classifier: cross dataset pairwise correlations for micro-average metrics	D-11
Table 7.20 – ANN classifier: cross dataset pairwise correlations for micro-average metrics	D-11
Table 7.21 – SVM classifier: cross dataset pairwise correlations for micro-average metrics	D-12
Table 7.22 – K-NN classifier: cross dataset pairwise correlations for unweighted macro-average metrics	D-12
Table 7.23 – KStar classifier: cross dataset pairwise correlations for unweighted macro-average metrics	D-13

Table 7.24 – LWL with Naïve Bayes classifier: cross dataset pairwise correlations for unweighted macro-average metrics	D-13
Table 7.25 – Decision Trees classifier: cross dataset pairwise correlations for unweighted macro-average metrics	D-14
Table 7.26 – ANN classifier: cross dataset pairwise correlations for unweighted macro-average metrics	D-14
Table 7.27 – SVM classifier: cross dataset pairwise correlations for unweighted macro-average metrics	D-15
Table 7.28 – K-NN classifier: cross dataset pairwise correlations for weighted macro-average metrics	D-15
Table 7.29 – KStar classifier: cross dataset pairwise correlations for weighted macro-average metrics	D-16
Table 7.30 – LWL with Naïve Bayes classifier: cross dataset pairwise correlations for weighted macro-average metrics	D-16
Table 7.31 – Decision Tree classifier: cross dataset pairwise correlations for weighted macro-average metrics	D-17
Table 7.32 – ANN classifier: cross dataset pairwise correlations for weighted macro-average metrics	D-17
Table 7.33 – SVM classifier: cross dataset pairwise correlations for weighted macro-average metrics	D-18
Table 7.34 – Power calculations: time vs. frequency domain	E-1

Chapter 1 Introduction

1.1 Context

The global demand for energy has been steadily increasing since 1990 and it is set to grow by 37% between 2012 and 2040 (from 13157 Mtoe in 2012 to an estimated 18419 Mtoe in 2040), according to the International Energy Agency [1]. This growth is mostly driven by the emerging economies in Asia (60% of the global total), Africa, Middle East and Latin America, contrasting the most developed nations in Europe, North America and Pacific that manage to maintain a near-steady demand during that period [1].

Likewise, electricity consumption has been experiencing a steady increase since 1990, in large part led by the BRICS¹ countries who shared among them 35% of the total world electricity consumption in 2012. As a matter of fact, these numbers are only a reflection of how the world evolved in the last couple of decades, with electricity emerging as the second most used end form of energy with a 17.7% share, only behind oil with 40.8% [2].

One of the leading factors for this growth in electricity demand is the change in energy consumption habits in domestic environments which was, in 2010, responsible for 28% of the final electricity consumption among all sectors (Figure 1.1 – left), a figure which represents an overall increase of almost 40% between 1990 and 2010 [3].

¹¹ BRICS is the acronym for the association of five major emerging national economies: Brazil, Russia, India, China and South Africa.

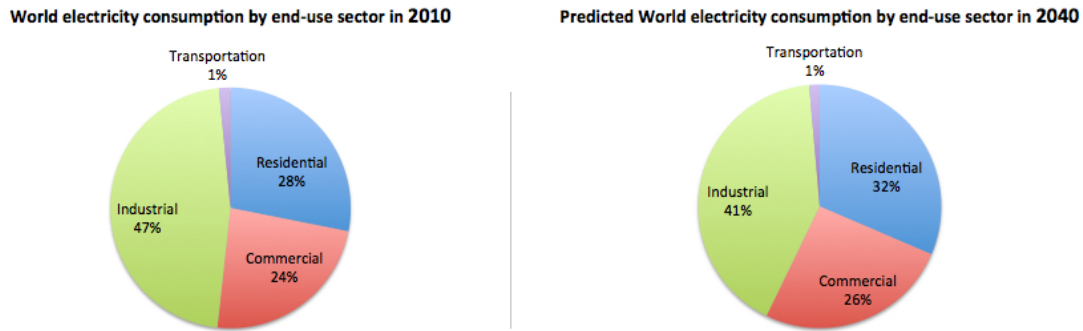


Figure 1.1 – Total electricity consumption by end-use sector: 2010 (left), 2040 (right)

One of the factors leading to the growth in electricity consumption in the last years is the notion of wellbeing based on personal ownership and mass consumption. As more people in developing countries have access to higher levels of comfort it is expected that the world's demand for electricity will continue to increase in the next couple of decades. In fact, according to the U.S. Energy Information Administration [3], it is expected that the demand for electricity in residential and commercial buildings will see an average annual percent increase of 2.6% and 2.5%, respectively, until 2040 reaching, by then, 32% and 26% of the final energy consumption (Figure 1.1 – right).

Nevertheless, improvements in the quality of life enabled by electricity (e.g., improved heating and ventilation systems, more and better electrical appliances, etc.) do not come without environmental costs. In fact, evidence shows that the carbon dioxide emissions from fuel combustion used to generate electrical energy, have also been steadily increasing since 1990. An increase that is particularly evident in the emerging economies (BRICS), which have shown an average yearly increase of 5.4% between 2000 and 2012.

Furthermore, with energy-related carbon dioxide emissions expected to grow 46% by 2040, it is expected that residential energy consumption will contribute significantly for the degradation of our eco-systems. Hence, the importance of domestic electric energy in the global context of energy overconsumption as outlined in [4] where the authors mention that residential buildings hold the potential for achieving one of the seven stabilization wedges required to reduce carbon emissions by 2054.

In particular, green building techniques are expected to play an important role in reducing the impact of space thermal comfort (HVAC²), lighting and water heating through smart construction strategies (e.g., effective window placement, wall and roof insulation and solar water heating). Whilst current technology already offers appliances that consume less power while delivering the same or better results, e.g. LED devices (lights and TVs) and energy-efficient appliances like refrigerators that are expected to use about 15% less energy than the traditional ones³. Yet, even though substantial savings can be achieved through technology that is currently available, most of these are still expensive or have payback times that are too prolonged in time to make them appealing to the general population of domestic consumers.

Moreover, research indicates that a scenario where infrastructures themselves require less energy would raise the problem of perverse incentives⁴ that suggest that the cheaper something is, the more it will be used [5]. This is supported by reports suggesting that the low price of electricity in some regions is one of the reasons behind the increase of electrical energy consumption. Therefore, despite the potential savings that can be achieved with these technological solutions, it does not necessarily mean that this would result in lower energy consumption and, consequently, reduced carbon dioxide emissions.

In fact, literature reveals that the real potential for reducing energy consumption lies with the consumers making a more efficient use of the house utilities and not so much on the buildings themselves. Many studies suggest that providing users with real-time and historical information about their consumption can lead to potential savings between 5% and 10% [6], [7], especially in the cases where the feedback is enhanced with individual appliance consumption information [8].

This is commonly known as **eco-feedback technology** and is defined as *the technology that provides feedback on individual or group behaviors with a goal of reducing environmental impact* [9]. The basic assumption behind eco-feedback technology is that

² HVAC is the acronym for Heating, Ventilations and Air Conditioning technology

³ Energy Star, www.energystar.gov

⁴ A perverse incentive is an incentive that produces unplanned or unwanted results.

people will be able to change their actions and consequently reduce their consumption if they are able to understand which appliances are responsible for their overall energy consumption breakdown.

This was especially noticed in [7], where Parker and colleagues evaluated two low-cost monitoring systems and found that users quickly discovered that by simply examining the differences in the overall demand by turning appliances *ON* and *OFF* they could easily approximate the energy usage of each individual appliance.

As a consequence, there has been a substantial effort to create monitoring solutions that are able to provide the consumption figures of individual appliances. Including, for instance, **electrical sub-metering** (i.e., installing individual sensors in each appliance) or the development of **smart appliances** that are able to communicate their own energy consumption to a central gateway.

Yet, despite the fact that the reported results are mostly positive regarding improved awareness and achieving savings in energy consumption [6], it has been also reported that after an initial period of exposure to this technology the tendency is towards a decrease in the attention given to the feedback leading to behavior relapse [10], [11]. This is defined in literature as the *response-relapse effect* and suggests that in order to properly assess the effectiveness of eco-feedback as a tool for promoting sustained energy saving, future studies should be carried for longer periods of time.

Furthermore, the intrinsically intrusive nature of such solutions implies that the information about the consumption of individual appliances will be associated with higher installation and maintenance costs that may obfuscate the potential savings [12].

1.2 Motivation

Against this background, there has been a significant research effort devoted to the development of **non-intrusive load monitoring** (NILM) techniques that are able to sense and disaggregate energy consumption from measurements taken at a limited number of locations

in the electric distribution grid, hence contrasting the more traditional intrusive monitoring technology that involve deploying multiple sensors throughout the house.

Early research in this topic dates back to 1985, when George Hart from the Massachusetts Institute of Technology (MIT) coined the term Non-Intrusive (Appliance) Load Monitoring (NILM) [13] [14]. In very simple terms we define NILM as *a set of signal-processing and machine-learning techniques that are used to estimate the aggregate and individual appliance electricity consumption from current and voltage measurements taken at a limited number of locations in the electric distribution of a house (optimally the mains, hence covering the demand of the entire house).*

Still, it was only in recent years that NILM gained renewed attentions from the research community, in part due to the potential of such advanced metering technologies in promoting *overall year-round* and *indirect short-term* energy saving strategies, which are expected to considerably reduce the carbon footprint associated with electric energy consumption. Furthermore, NILM is expected to serve as the backbone technology that will enable the creation of innovative smart-grid services that go beyond helping individuals saving energy, as it was recently observed in the 2016 edition of the international NILM workshop [15].

The potential benefits of NILM include for example:

- The lower costs of installation and maintenance of NILM systems, enables the deployment of long-term energy efficiency programs like eco-feedback, which are necessary in order to access to long-term effectiveness of such programs.
- Having energy consumption disaggregated by appliance also enables the creation of novel energy efficiency services. These include, the possibility of inferring and providing eco-feedback on the everyday activities, e.g., preparing meals or taking care of the laundry [16], [17].
- Additionally, NILM technology also enables the detection of anomalies in the electric loads [18], [19], which can result in the early detection of malfunctioning appliances, hence avoiding possible energy waste.

This being said, it is not totally unexpected that recent times have seen a considerable increase in the body of work in the field of energy disaggregation, which is naturally reflected in the exponential grow of published papers in the topic [20] as well as in the recent boost in the number of companies offering NILM products and services [21], [22].

Yet, and despite the growing body of work in this field, there are still many challenges that must be solved before it is possible to take full advantage of the potential benefits of low-cost and reliable NILM solutions. These are grouped according to two categories, namely: i) *load identification*, and ii) *training and supervision challenges* [23].

As the name suggests, the former encompasses the different issues related to the problem of *correctly identifying the loads* in the ever-growing complexity of the electric grid (e.g., different types of load and simultaneous load switching) and are naturally the focus of much of the undergoing research [24], [25]. On the other hand, the latter encompasses the issues related to the *replication and generalization of research findings* (e.g., the lack of proper test and training data and the inexistence of a formal agreement on how to report the disaggregation results) and only recently have become the focus of a smaller group of NILM researchers [26], [27].

Furthermore, and despite the abundance of literature, it was until only recently that we saw the first publications regarding the value proposition of NILM as a tool to reduce energy consumption, or trying to educate the research community about the practical issues of deploying such systems in real world scenarios [28]–[31], which we believe are of crucial importance to the large-scale adoption of NILM technology in years to come.

1.3 Problem Statement

As previously explained, despite the growing body of work in NILM research, there are still some underexplored areas. This is particularly evident when it concerns to the practical issues of deploying NILM systems or conducting formal evaluations and benchmarks of the proposed algorithms, which are the main topics of the work in this thesis. To state more concretely, our goals with this work are twofold:

Firstly, we will focus our attention on understanding the practical issues of deploying NILM and eco-feedback systems in the domestic environments. These practical issues include for instance, the ease of installation and use of the monitoring equipment, which may ultimately affect how such systems are received and adopted by the residential sector.

Secondly, we will focus our attention on understanding the challenges of defining a consistent set of performance metrics for the energy disaggregation problem. More concretely, we propose to study the compatibility of existing performance metrics with the nature and structure of the data generated by the different NILM algorithms, which may ultimately change the way the different metric are used to draw conclusions regarding the performance of such algorithms.

1.4 Thesis Outline

The remaining chapters of this thesis are organized as follows. In Chapter 2, we provide a comprehensive background and literature review on NILM. Then, in Chapter 3 we formalize the research questions addressed in this thesis and present the methods we will use to answer these questions.

Following that, the main body of this thesis will be divided in three chapters. First, in Chapter 4 we propose a data format to represent energy disaggregation datasets, and present two public datasets that emerged from the work in this thesis. Then, in Chapter 5 we describe two bespoke energy monitoring and eco-feedback platforms, and thoroughly discuss the practical considerations of deploying such platforms in real world scenarios. Lastly, in Chapter 6 we study the behavior of a number of performance metrics when they are used to evaluate two different types of NILM algorithms.

Finally, in Chapter 7 we summarize the contents of this thesis and discuss general ideas for future work in evaluating the performance of energy disaggregation systems.

Chapter 2 Non-Intrusive Load Monitoring

In this chapter we review the state of the art in Non-Intrusive Load Monitoring, which is the main focus point in the research scope of this thesis. We start with a review of the field, going from its early days to the many challenges that are currently found in literature. We then provide an extensive literature review of the on-going research efforts in this field. More particularly, we first review the current literature in the task of correctly identifying the different appliances in the aggregated load; we then report on the main research efforts that are being conducted with respect to the task of evaluating the performance of the different approaches that have been proposed to solve the problem of Non-Intrusive Load Monitoring.

2.1 Seminal Work

As previously mentioned, the first attempt to disaggregate energy consumption from a single location dates back to 1985 when Hart [13] proposed his prototype Non Intrusive Appliance Load Monitor (NIALM). The basic assumption behind the first NILM algorithms is that every change in the total electrical load of a building happens as a response to an electric device changing its state, e.g. a television turning *ON* or *OFF*. As such, early approaches were designed such that it was possible to detect the power changes in the household's electricity demand and extract features from the vicinity of the power changes that were then used to discriminate between the different appliances power demands through the application of machine learning algorithms.

Assuming this to be consistently true, the proposed algorithm worked by taking 1 second interval measurements of real and reactive power from both power legs of the electricity grid. These measurements were then normalized (to ensure that potential variation were accounted for), and used in the edge-detection step that looked at identifying when appliances changed

their working state. For each identified power change the total amount of change was computed by subtracting the steady power level prior to the change from the steady power level after the change ends.

The observed changes were then clustered according to the amount of change observed in each measurement, and the obtained clusters were subsequently used to match the *ON* and *OFF* clusters to each appliance according to their time of occurrence, assuming that each *ON* / *OFF* pair would correspond to a single cycle of appliance usage. The total energy consumption was then computed by multiplying the amount of (positive) power change with the time elapsed between the *ON* and *OFF* events (in hours). Ultimately, each *ON* / *OFF* cluster pair was matched to an appliance name by checking its characteristics against all the appliance classes provided in a separate table with all the operational characteristics of different appliance types (e.g. expected power changes, time of operation and number of cycles).

Building on these early findings, Hart continued his research and in 1992 an enhanced version of his NILM system was reported [14] in which multi-state appliance disaggregation was also addressed. To this end, Hart introduced the idea of modeling multi-state appliances as Finite State Machines in which the circles indicate the states that an appliance can be, and the arcs indicate the allowed state transitions. The original algorithm was then enhanced with two extra steps, one to build the actual appliance models and another to keep track of the behavior of the appliances according to their models.

2.2 Event-Based and Event-Less Approaches

After Hart's publications, very few research efforts were reported in the following 10 to 15 years. However, the foundations of the current research efforts were launched at that time, as some of Harts' original ideas are now cornerstones to many of the ongoing research efforts.

For instance, the early NILM assumption that every change in the aggregate power happens in response to an appliance changing its mode of operation, and the concept of appliance signatures are as of today the basis of the event-based approaches.

Furthermore, Hart also introduced the concept of appliance types (described in Table 2.1) that turned out to be the starting point for the creation of event-less approaches.

Table 2.1 – Appliance types definitions and examples

Appliance type	Description	Examples
ON / OFF	Appliances that are either running or not, i.e. either ON or OFF.	Light-bulb, toaster and water kettle
Finite State	Appliances that during their operation will pass through a finite number of operation modes.	Clothes washer and clothes drier
Variable Power	Appliances whose power draw is variable and no finite number of states or transitions can be observed.	Dimmer lights and power tools
Permanent Consumers	Appliances that normally run on the background 24/7 with constant power draw.	Alarms and surveillance cameras

To the best of our knowledge, the concepts of event-based and event-less approaches were first introduced at the 1st International NILM Workshop⁵ in 2012, and aim at providing a clear categorization of the ever-increasing approaches to the energy disaggregation problem.

On the one hand, *event-based* approaches are intrinsically related to the early days of NILM, and seek to disaggregate the total consumption by means of detecting and labeling every appliance transition in the aggregated signal (see in Figure 2.1) using previously trained supervised or semi-supervised learning algorithms. Consequently, approaches categorized under this category require a data collection step where a number of transitions (i.e., power events) from the appliances of interest are collected, labeled and stored, to be used later as training data.

⁵ 1st International NILM Workshop, www.ices.edu/psii/nilm

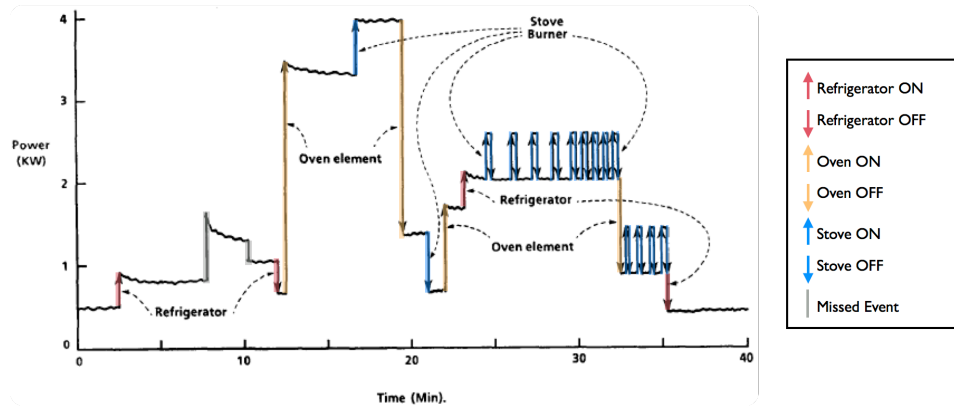


Figure 2.1 – Example of event-based energy disaggregation

Event-less approaches, on the other hand, do not rely on event detection and classification. Instead, these approaches attempt to match each sample of the aggregated power with the consumption of one specific appliance or a combination of different appliances (see Figure 2.2), by means of statistical (e.g., Bayesian methods) and probabilistic (e.g., Hidden Markov Models) machine-learning methods. Therefore, the training data does not require any labeled transitions. Instead, only the aggregated consumption of the loads of interest is required. Thus making the process of collecting training data for event-less approaches more straightforward than for event-based approaches.

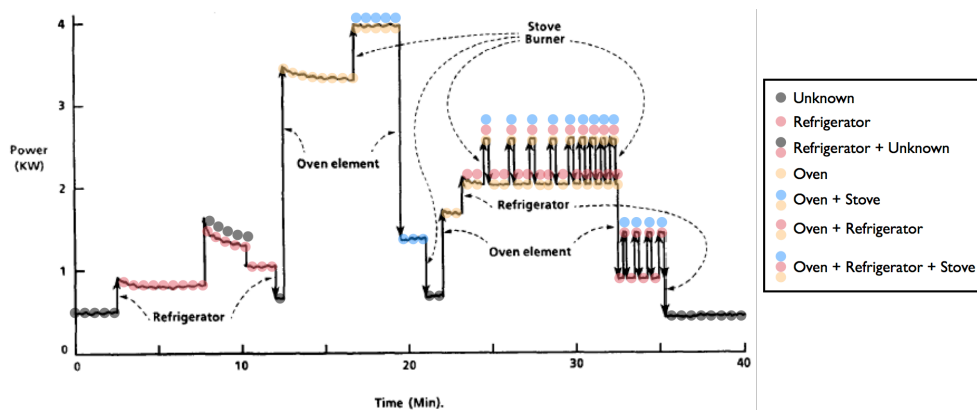


Figure 2.2 – Example of event-less energy disaggregation

Naturally, both approaches have advantages and disadvantages. For example, despite the fact that event-based approaches require the continuous execution of event detection algorithms, the appliance inference is only performed when events are detected, hence

making these approaches more computationally efficient. Yet, the success of the final energy estimation is heavily dependent on the detection and classification steps, consequently, any missed detections or erroneous classifications will be propagated in time, possibly leading to large energy estimation errors.

On the other hand, in the case of event-less approaches, the inference step is performed for every sample, making such approaches considerably more computationally intensive. Nevertheless, since all the data is taken into consideration at all times, errors are not expected to propagate. Instead, these will be corrected as the inference algorithms are being executed.

2.3 Existing Challenges

More than three decades after Hart first introduced Non-Intrusive Load Monitoring in his seminal work [13], this is still a very active field of research. Yet, despite all the research efforts, to date some considerable technical challenges are still present.

Currently, the different challenges are grouped according to two categories, namely: i) *load identification challenges*; and ii) *training and supervision challenges* [21]. These are described next in more detail.

2.3.1 Load identification challenges

Challenges under this category are related to the problem of correctly identifying the individual loads, and the first challenge is the ever-growing complexity of the domestic electric grid and the **very different load types** [24], [25] that NILM systems must account for. For instance, variable power loads (e.g. dimming lights), multistate loads (e.g. clothes washer) and always-on loads (e.g. security cameras and alarms).

Likewise, NILM algorithms also need to be able to discern between **appliances that draw the same power** [24], [25], independently of being similar appliances or just different devices working at the same power level. Furthermore, and more specifically for event-based approaches, researchers need to account for **simultaneous power events** (i.e., when loads are activate at the same time or nearly at the same time), which may introduce errors in the event

detection process that can propagate to the subsequent stages and result in large energy estimation errors [32].

Lastly, and perhaps the most important challenge, is the fact that researchers need to be fully aware of the **dynamic nature of the electric grid** [33], which makes the problem of energy disaggregation considerably different from many other classic machine-learning problems. In other words, in many of the classic machine-learning domains (e.g., speech recognition or hand writing recognition) training and testing datasets are assumed to have the same or nearly the same statistical properties of the future data that will be given to the learning algorithms. However, due to the dynamic nature of the electric grid, this is not likely to happen in NILM problems. Instead the learning algorithms must be robust against changes in the future data, like for example the presence of unknown and / or malfunctioning appliances or the **many different modes of operating and combining such appliances** [16], [34].

2.3.2 Training and supervision challenges

Challenges under this category encompass the many issues related to training and evaluating the performance of the different NILM solutions that are being proposed by the research community. For example, different algorithms require different training data, e.g., event-based approaches need labeled transitions while event-less approaches require historical traces of individual appliance consumption data. Yet, very little work was carried out in this direction, thus there are currently **no identified strategies to collect training data** [25], [33]. This is especially important in the case of event-based approaches as these rely on considerable amounts of labeled appliance transitions that will most probably require human intervention to get.

Furthermore, and despite the efforts to create public datasets that is being observed in the last couple of years, the **shortage of proper public datasets** is still considered one of the main caveats of NILM research, particularly in the case of fully labeled datasets that are required to train and validate event-based approaches. Moreover, the **currently available datasets have wide difference between each other** [26], [35] (e.g. data formats, available

measurements, data resolution and appliance types), which on the one hand makes the task of evaluating algorithms very time consuming, and on the other hand, adds considerable bias to the evaluation results, hence compromising any cross-dataset benchmarks.

Finally, and despite some efforts that have been made to carry out formal evaluations of the technology (e.g., [27], [31], [36]), as of today there is **no formal agreement on which metrics should be used to measure and report the performance of NILM algorithms and systems** [31]. Instead, most evaluations have focused solely on reporting the accuracies of the proposed methods without having previously studied the compliance between the used metrics and the NILM problem, like it is done in other machine-learning domains [37], [38].

2.4 Literature Review on Load Disaggregation

As it was mentioned previously, as of today the different approaches to the NILM problem are grouped according to two categories, namely: event-based and event-less approaches. As such, in the next two sub-sections we summarize some of the most relevant research to date according to these categories. Nevertheless, we should note that other categorizations can be found in the literature, for example [24] categorizes NILM research in terms of the metering feedback dimension, i.e. low frequency vs. high frequency, while [25] focuses on signature features and algorithms.

2.4.1 Event-based Approaches

Event-based approaches for energy disaggregation are intrinsically linked to the early work by Hart, and aim at computing individual appliance consumption by keeping track of every appliance state transition (e.g. kettle turning *ON* or *OFF*) by means of event detection and classification assuming that the system was previously trained.

A typical event-based NILM system workflow contains five consecutive steps, as shown in Figure 2.3: **data acquisition**, where signals representing the electrical energy flowing into the house are sensed, sampled and transformed into power-related measurements (e.g. real and reactive power); ii) **event detection**, which is the process of identifying the changes in

the consumption that are assumed to happen in response to appliances changing their mode of operation; iii) **feature extraction**, where different parameters are extracted from the vicinity of the power event, forming a power event signature that will be used in the process of identifying the loads responsible for each event; iv) **event classification**, where previously trained machine-learning algorithms are applied to the signatures of the previously detected power events to obtain a classification, i.e., the name of the appliances that triggered the events; and v) **energy estimation**, where the consumption of the individual loads is estimated based on the labeled power events and their distribution in time. Next, we provide a comprehensive literature review on the event detection, feature extraction, event classification and energy estimations steps.



Figure 2.3 – General workflow for event-based NILM approaches

2.4.1.1 Event detection

According to the literature in event detection for NILM [39], the different approaches are grouped in three categories: i) expert heuristics; ii) probabilistic models; and iii) matched filters.

Expert heuristics

Algorithms under the *expert heuristic* category are probably the less complex, and follow the basic principle of scanning the time series data looking for changes that are above a certain threshold, as defined by Hart in his seminal work [13].

For example, in [40] the power signal is first filtered to minimize the presence of noise and reduce the chance of false positives. On a second step, the power events are detected by means of computing the absolute differences between two consecutive samples and selecting the indexes where this difference is above a pre-defined threshold. In [41] a similar approach is proposed, yet instead of computing the absolute differences between two consecutive

samples, the differences are calculated between the current sample and the sample X seconds before. Moreover, in order to help reduce the number of false positives, an index with absolute value above the pre-defined threshold is only considered a power event if no power event was detected in the last Y seconds.

Probabilistic models

Another approach to event detection is by means of probabilistic methods. In this category of detectors, the event detection occurs in two steps, as described below:

In the first step it is necessary to calculate the chance of an event occurring at each sample of the power signal. This signal is normally referred to as the *detection statistic*, and is computed by applying either statistical tests (e.g., Generalized Likelihood Ratio (GLR) [42], Goodness-of-Fit (GOF) [43], CUmulative SUM (CUSUM) [44]) or other mathematical functions (e.g., Kernel Fisher Discriminant Analysis (KFDA) [45]), to the power measurements by means of sliding windows.

In the second step the power events are extracted from the resulting *detection statistic* signal. This is normally done via thresholding, i.e., whenever the detection statistic is above a certain threshold a power event is flagged in the power sample that corresponds to that index [42], [43]. Nevertheless, for the particular case of NILM, more robust strategies have been designed. For instance, in [46] and [47] the selection of the power events is done by applying either a voting algorithm or a maxima/minima locator algorithm to the *detection statistic* signal, respectively.

Match filters

In this category of algorithms, power events are detected by correlating a known, or template with an unknown signal to detect the presence of the former signal in the later. In other words, match filter event detectors work by trying to find known appliance transients (i.e., templates) in the aggregated consumption signal (i.e., unknown signal) by means of filtering techniques.

To the best of our knowledge, this was first attempted in the NILM domain in [44] and [45] where the authors propose an event detector that attempts to match segments of startup

transients (obtained from training) to the aggregated signal using two transversal filters in sequence. The first filter is used to find the transient shapes in the aggregate signal, and the second filter is used to enforce that the matches correspond to actual transients and not some fortuitous noise [49].

As of today, the match filters category also incorporates those detectors that use filters to transform the power measurements into signals that emphasize potential power events while depreciating the steady stage regions, similarly to what is done in the probabilistic models category. For example, in [50] the authors apply a Hilbert transform⁶ to the instantaneous current sampled at 20 kHz. This is followed by a combination of average and derivation filters on the transformed signal such that only the transitions of interest (i.e., power events) are represented.

Another example of event detection based in filter matching is the work of Baets et al. [51] that apply Cepstrum analysis⁷ to the power signal computed at 60 Hz. The resulting signal is then thresholded such that only the positions where the signal is above a certain value are considered power events.

2.4.1.2 Feature extraction

Feature extraction is the process of selecting the best features such that the power event signatures are robust and have enough discriminative power between different appliances. Overall, features can be categorized as being either *engineered features* or *data driven features*. The former encompasses features that are extracted by taking advantage of the domain knowledge that we have from electrical power and appliance characteristics while the latter refers to features that are learned directly from the data by means of techniques like unsupervised feature learning [52].

Engineered features are normally extracted from the samples surrounding the event of interest. The most common examples of these features are the amount of power change (also known as delta metrics), transient shapes, harmonic components [46] and voltage and current

⁶ Hilbert transform, <http://mathworld.wolfram.com/HilbertTransform.html>

⁷ Cepstrum analysis, <http://www.mathworks.com/help/signal/ug/cepstrum-analysis.html>

(V-I) trajectories [53], [54]. Additionally, several features have been drawn from the frequency domain content like Electromagnetic Interference that emanate from certain appliances [55], electric noise [56] using Fast Fourier Transforms (FFT) or in some cases the Wavelet transform to simultaneously extract time and frequency domain features [57].

With respect to data-driven features, these are also extracted from the measurements surrounding the power event. Yet, unlike engineered features, these are directly learned from the data. For example, in [54] Lam et al. proposed the application of Single Value Decomposition (SVD) techniques to extract features from the current waveforms, whereas Gao et al. propose the use of VI binary images (V-I trajectories that are amplitude normalized and converted to binary images) and Principal Component Analysis (PCA) on the V-I binary images [58].

2.4.1.3 Event classification

Current NILM literature is very rich in terms of supervised learning algorithms for event classification. These range from the more traditional learning algorithms, like the K-Nearest Neighbor (K-NN) [44], decision trees [58], [60], naïve Bayes (NB) [61], [62], artificial neural networks (ANN) [63], [64], [65] or support vector machines (SVM) [60], [66], [67] to more complex approaches like genetic algorithms (GA) [60], [63], [68] and Integer programming [69].

Some authors have also explored the feasibility of ensemble-based approaches where different algorithms are combined to enhance the overall classification performance [70], [71]. Likewise, the possibility of sequentially combining different classification algorithms was also explored. For instance, in [41] the authors propose a two-step approach for energy disaggregation. In the first step, one classifier attempts to discriminate the appliance by category (i.e., purely resistive, inductive or capacitive), such that in the second step another classifier is trained with only the features that are considered relevant to identify appliances under that category.

Lastly, Barsim and Yang have also experienced with semi-supervised approaches that attempt to make use of both labeled and unlabeled data for training classification algorithms

[72]. In very simple terms, the rationale behind semi-supervised approaches is the fact that in most machine-learning problems labeled data is scarce or very expensive to obtain. As such, semi-automatic learning methods attempt to leverage the potential of unlabeled data by using small sets of labeled examples to infer the labels of unlabeled examples and use them later as training data [73].

2.4.1.4 Energy estimation

Finally, in the energy estimation step the classified power events and associated timestamps are used to infer the consumption of the individual appliances. This topic was briefly explored in Harts original work where the author proposes to model the individual appliance consumption by means of expert heuristics like the Zero Loop Sum Constraint (ZLSC), which states that the sum of power changes in any cycle of state transitions is zero [14]. This method, however, is simplistic given that it assumes that power transitions of a given appliance are symmetrical, and that there are no simultaneous events.

Other attempts on the topic of energy estimation include the works of Baranski and Voss [68], [74] and Streubel and Yang [75]. The former presents a completely unsupervised method of estimating appliance behavior based on observed power differentials, and optimization of a quality function using genetic algorithms (GA). The latter proposes the modeling of appliance behavior using Finite State Machine (FSM) formulations by separating the power traces of a single appliance into transients and steady-state modes. However, these two approaches remain to be validated and some of the assumptions made by the authors have been shown to present some considerable drawbacks to both approaches as stated in [24] and [76].

Lastly, Giri and Bergés also proposed an approach to energy estimation [76]. The proposed framework that aims reducing the effects of outliers (incorrect labels) and missed state transitions, is composed of five sequential steps: i) clustering of the detected power events, ii) perturbation of the power events to enforce the ZLSC, i.e., correction of the missed detections, iii) creation of FSMs from the corrected transitions, iv) correction of any errors

that violate the ZLCS, i.e., correction of outliers, and v) estimation of energy consumption using the resulting FSMs of each individual appliance.

The proposed framework was tested on the BLUED [77] and REDD [78] datasets, using the percentage error in energy estimation (PEEE) [46] as the performance metric. The results have shown a considerable variation of the performance metric across the datasets, which in average ranged from 5.9% and 16.5 in BLUED to 22.4 in REDD.

In summary, the sequential nature of event-based approaches implies that each step in the process will result in affecting its successors. Consequently, it is safe to say that the ultimate goal of these solutions is to find the best combination of algorithms and features across the different steps such that the properly disaggregated energy is maximized. Furthermore, it is evident that these approaches require big volumes of labeled data for algorithm training, which in itself is another different problem for NILM researchers to solve.

To the best of our knowledge, as of today, only a few authors have attempted to tackle this issue. For example, Berges, in his user-centered approach, provides a mechanism that prompts users to provide appliance information whenever the system is not able to find a match with a power event [46]. Weiss [59] proposes the application of mobile apps in order for users to collect appliance labels and signatures in real time. Lastly, on the commercial side, Bidgely proposes the application of crowdsourcing techniques to collected appliance labels from their customers [79].

2.4.2 Event-less Approaches

Unlike event-based approaches, the event-less alternatives do not require that machine-learning algorithms be previously trained to identify every individual power change in the aggregated signal. Instead, these approaches rely mostly on the existing knowledge about individual appliance operation, through different techniques like motif mining, blind source separation and probabilistic graphical models.

2.4.2.1 Motif mining

A motif mining approach for energy disaggregation was proposed in [80] and works by mining the aggregated power signal for recurring episodes (i.e. individual appliance working cycles), that are composed of sequences of power events (e.g. + 1000 -400 -600) and match them to individual devices that are known to exhibit such behavior. Each episode must fulfill certain conditions in order to be considered as belonging to an appliance. This includes the minimal episode completion criterion that is used in order to identify episodes that are completed by a single device (e.g. the episode +600 -800 +400 -1000 + 800 would not be considered, whereas episodes +600 +400 —1000 and -800 +800 would be accepted).

2.4.2.2 Blind source separation

Blind source separation is the process of separating individual sources from a signal that is known to be composed of a set of mixed signals but very little or no information is provided regarding the source signals or the mixing process.

In [81], a blind source separation technique has been applied to the problem of energy disaggregation where the authors use the steady-state active and reactive power changes (ΔP and ΔQ) to create appliance clusters, each of which was assumed to correspond to one appliance state transition. A matching pursuit algorithm (MP) is then applied to reconstruct the original source (i.e. aggregated power) of each cluster (i.e. appliance). Another example of blind source separation is presented in [82] where the authors propose the application of discriminative sparse coding to find the sets of basis functions that best represent each individual appliance. Non-negative matrix factorization is then applied to find the optimal sparse set of basis function activations that best explain the household aggregate data.

2.4.2.3 Probability graphical models

In contrast to the methods we have seen so far, that require a separate event detection process, a new approach has emerged in which load disaggregation is attempted using probabilistic approaches based on Probabilistic Graphical Models (PGMs), which only take into account

the power consumption and eventually non-power features such as the duration and time of appliance usage.

The basic assumption behind such methods is that the aggregated electrical energy consumption (P) at a given instant (t) is characterized by the consumption of several appliances that are operating in a particular mode. Therefore the disaggregation problem can be formalized as the task of finding the best possible mode sequences (m) that explain the observed aggregated power (P). Given this, authors have attempted to develop such models of appliance behavior using several variations of Hidden Markov Models (HMM), in which the ultimate goal is to find the sequence of hidden states that best represent the model outputs (i.e. the aggregate power at a given instance in time).

An example of using HMM for energy disaggregation is the work of Parson et al. in [83] in which, for each appliance, a semi-supervised algorithm is used to determine its most likely sequence of states (i.e., model each appliance as a HMM). To that end, the authors feed their modeling algorithm with generic appliance models containing information about the operational characteristics of the appliance type they are modeling (e.g., aggregate consumption, state transition probabilities and the estimated consumption in each state). Lastly, using the state sequence of each appliance the disaggregation algorithm attempts to estimate its consumption and subtract that from the aggregated consumption before repeating the process for the next available state sequence, until there are no more state sequences remaining.

Likewise, Kolter and Jaakkola [84] also propose modeling appliances as HMMs, yet, in contrast to Parson et al. [83], the authors use an unsupervised algorithm to estimate the number of appliances and their consumption patterns taking only the aggregate consumption data as input. To this end, the algorithm works by extracting snippets of the aggregated consumption data that most likely correspond to an appliance's working cycle (defined as the period between the appliance's start-up and shutdown). Each extracted snippet is then modeled as an HMM and those that are most likely to belong to the same appliance are identified as such. This results in a factorial HMM (FHMM) (i.e., a composition of several independent HMMs), which the authors then use to estimate the consumption of the individual appliances [78].

Some NILM researchers have also explored the possibility of simultaneously modeling different aspects of the energy consumption data by means of combining different HMMs. For example, in [85] Kim et al. shows how Conditional Factorial Hidden Markov Models (CFHMM) can be combined with Hidden-Semi Markov Models (HSMM) and the Input-Output Hidden Markov Model (IOHMM) in the context of energy disaggregation. In this particular work, the CFHMM allows the dependencies between appliances to be modeled (e.g. dependency between computer and monitor), while the HSMM allows appliance usage durations to be modeled explicitly (e.g. length of washing machine cycle), and finally the IOHMM allows additional observations, which might influence appliance use to be built into the model (e.g. dependency of shower usage on time of day).

Lastly, Lange and Bergés have recently proposed a combination of event-less and event-based approaches by means of Dual-Emissions FHMM [86]. More concretely, in this work, the authors' combine the observed power readings (P) and a feature vector containing information extracted from the appliance transitions (i.e., power events).

2.5 Literature Review on Performance Evaluation

As previously explained, one of the current challenges of NILM research is the inexistence of a formal method to assess the performance of the many solutions proposed by the community. Moreover, unlike the load identification problem that already has a wide body of research, only a few researchers are now devoting their efforts towards creating such methods.

These efforts are summarized in the next three sub-sections. More specifically, we present and describe some of the available household energy datasets, the frameworks and toolkits that have been developed to leverage the potential of such datasets. Lastly, we review the performance metrics that have been used to report the accurateness of the proposed NILM algorithms and systems.

2.5.1 Public Household Energy Datasets

A household energy disaggregation dataset is a collection of electrical energy measurements taken from houses in real-world scenarios, without disrupting the everyday routines of the household, i.e., trying to keep the data as close to reality as possible.

These usually contain measurements from the whole-house consumption (taken at the mains) and of the individual loads (i.e., ground-truth data), which is obtained either by measuring each load at the plug-level or measuring the individual circuit to which the load is connected. In a real-world scenario, however, normally multiple loads are connected to the same circuit; therefore, this last method does not always ensure that the individual consumption of all the different loads is actually available.

Similarly to what happens with the different NILM approaches, the currently available datasets can also be categorized as event-based or event-less datasets. The major difference between the two categories of datasets lies in the fact that the latter does not require the identification of every power change. Consequently, collecting datasets for event-less approaches is more straightforward and less time consuming, which in part explains the higher availability of event-less public datasets, as we will see below.

Currently, there are to the best of our knowledge, 20 public household energy datasets. From these, 15 are suitable to evaluate event-less approaches and four to evaluate event-based approaches. The remaining dataset contains only aggregated whole-house consumption and therefore very little application to the energy disaggregation problem. Detailed descriptions of each dataset can be found in section A.2 of Appendix A.

In order to facilitate comparisons between the existing solutions, in Table 2.2 we provided a brief summary of the 20 datasets. The following characteristics are provided: Year of release, country, number of monitored households, if the data is continuous or not (continuous – C or not continuous – NC), i.e., if the data was collected in consecutive time periods. The approaches enabled by the dataset (event-based – EB or event-less – EL), types of smart-meters used in the collection (whole-house – WH, individual circuit – IC or individual appliance – IA and if a list of power event labels is available – LE). The available

electric energy features (current – I, voltage – V, real power – P, reactive power – Q, apparent power – S, others) and the dataset time resolution.

Table 2.2 – Overview of public household energy datasets

Dataset (Year)	Country (Houses)	Duration	Approach			Meters				Features							Resolution
			EB	EL	WH	IC	IA	LE	I	V	P	Q	S	Others			
REDD [78] (2011)	USA (6)	2-4 weeks (NC)	✗	✓	✓	✓	✓	✗	✗	✓	✓	✓	✓	✗	✗	---	I & V: 15 kHz; P: 1 Hz IC & IA: 3-4 seconds
AMPds [87] (2013)	Canada (1)	2 years ^a (C)	✗	✓	✓	✓	✓	✓	✗	✓	✓	✓	✓	✓	✓	Frequency Power Factor Energy	1 minute
TEALD ⁸ (2016)	Canada (1)	N/A ¹ (C)	✗	✓	✓	✓	✓	✗	✗	✓	✓	✓	✓	✓	✓	Power factor Frequency	1 Hz
Dataport [88] (2013)	USA (1400 ²)	4 years ^a (C)	✗	✓	✓	✓	✓	✓	✗	✗	✗	✗	✓	✗	✓	---	1 minute
UK-DALE [89] (2014)	England (4)	499 days ^a (NC)	✗	✓	✓	✓	✓	✓	✗	✓	✓	✓	✓	✓	✓	---	I & V: 16 kHz; P, Q & VRMS: 1 Hz (2 houses) WH & IA: 6 seconds
iAWE [90] (2013)	India (1)	74 days (C)	✗	✓	✓	✓	✓	✓	✗	✓	✓	✓	✓	✓	✓	Frequency Phase angle Energy	1 Hz
Smart * [91] (2011)	USA (3)	3-4 month (NC)	✗	✓	✓	✓	✓ ^b	✗	✓	✗	✗	✗	✓	✗	✓	---	1 Hz

⁸ TEALD dataset, www.teald.org

Dataset (Year)	Country (Houses)	Duration	Approach		Meters			Features							Resolution
			EB	EL	WH	IC	IA	LE	I	V	P	Q	S	Others	
BLUED [77] (2012)	USA (1)	1 week (C)	✓	✗	✓	✗	✗	✓	✓	✓	✓	✓	✗	---	I & V: 12 kHz P & Q: 60 Hz
ECO [92] (2014)	Switzerland (6)	8 month (NC)	✗	✓	✓	✗	✓	✗	✓	✓	✓	✗	✗	Phase angle	1 Hz
OCTES ⁹ (2012/13)	Fin., Ice., Sco. ^e (33)	4-13 month (NC)	✗	✗	✓	✗	✗	✗	✗	✗	✓	✗	✗	Energy price	6-7 seconds
IHEPCDS [93] (2013)	France (1)	4 years (C)	✗	✓	✓	✓	✗	✗	✓	✓	✓	✓	✗	---	1 minute
HES ¹⁰ [94] (2010/11)	UK (251)	1-12 month (C)	✗	✓ ^e	✗	✗	✓	✗	✗	✗	✗	✗	✗	Energy	2 minutes
REFIT [95] (2014)	UK (20)	2 years (C)	✗	✓	✓	✗	✓	✗	✗	✗	✗	✗	✓	On / Off status	8 seconds
ACS-Fx [96], [97] (2013)	Switzerland (N/A)	N/A (NC)	✗	✓ ^e	✗	✗	✓	✗	✓	✓	✓	✓	✗	Phase angle	10 seconds
DRED [98] (2015)	The Netherlands (1)	6 month	✗	✓	✓	✗	✓	✗	✗	✗	✗	✓	✗	---	1 Hz 1 minute

⁹ OCTES Dataset, <http://octes.oamk.fi/final>

¹⁰ HES Dataset, <http://tinyurl.com/HES-Dataset>

Dataset (Year)	Country (Houses)	Duration	Approach		Meters						Features						Resolution
			EB	EL	WH	IC	IA	LE	I	V	P	Q	S	Others			
Tracebase [99] (2012)	Germany (N/A)	1883 days (N/A)	✗	✓ ^e	✗	✗	✓	✗	✗	✗	✓	✗	✗	---	1-10 seconds		
GREEND [106] (2014)	Austria, Italy (9)	3-6 month (C)	✗	✓ ^e	✗	✗	✓	✗	✗	✗	✓	✗	✗	---	1 Hz		
PLAID [108] (2014)	USA (55)	N/A	✓ ^f	✗	✗	✗	✓	✗	✓	✓	✗	✗	✗	---	30 kHz		
WHITED [101] (2016)	Ger., Aus., Ind. ^d (N/A)	N/A	✓ ^f	✗	✗	✗	✓	✗	✓	✓	✗	✗	✗	---	44 kHz		
HFED [102] (2015)	India (N/A)	N/A	✓ ^f	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	EMI spectrum	10 kHz – 5 MHz		

^a New data is continuously being added

^b Individual circuit ground-truth is only available in house A

^c Finland, Iceland and Scotland

^d Germany, Austria and Indonesia

^e These datasets can only be used as training data. Evaluation must happen in datasets where whole house consumption is available

^f Only possible for event classification using either cross-validation or events from other datasets

Starting with the event-less datasets, it is possible to see that only 11 of them (REDD, AMPds, Dataport, UK-DALE, iAWE, Smart*, ECO, IHEPCDS, REFIT, TEALD and DRED) contain aggregated and individual appliance / circuit consumption. As such, these are the only datasets that can be used simultaneously as training and testing data.

As for the remaining four datasets (HES, ACS-Fx, Tracebase and GREEND), they only provide individual appliance consumption information. Therefore, they can only serve as training data. One possibility to use these four datasets for training and testing data would be to artificially generate the aggregate data by summing the power demand of each appliance. Still, this approach presents some caveats that would certainly affect the final results. For example, summing all the individual loads completely excludes the effects of appliances that were not sub-metered, hence resulting in very simplistic and optimistic datasets that fail to represent the complexities of the household electrical grid.

Regarding the event-based datasets, only BLUED contains whole-house consumption information and a list of appliance labels for all the identified power changes, hence making this the only dataset that can be used for event-detection evaluation. Still, since it only contains one week of data it is not very suitable to evaluate the remaining algorithms of the event-based pipeline.

On the other hand, PLAID, WHITED and HFED only contain data from the startup transients and spectral traces of several individual appliances. Consequently, these three dataset are only suitable to evaluate feature extraction and classification algorithms using cross-validation [58], [71]. Likewise, it is also possible to use PLAID and WHITED to classify power events from other datasets, still it should be noted that they only contain startup transients, therefore it is not possible to classify OFF transitions.

Lastly, it is also remarkable that none of the datasets can be used to evaluate event-based and event-less approaches. On the one hand, BLUED does not provide any individual appliance consumption. On the other hand, none of the event-less datasets provide the location of the power events in the aggregate data.

In our understanding, this clearly highlights the difficulty of creating fully labeled datasets that can be used to evaluate both types of approaches. While one of the main challenges is the complexity behind creating hardware setups to collect data that support fully labeling NILM datasets, we argue that the main challenge lies in the actual labeling stage which still relies heavily on a lengthy and error-prone manual inspection of the whole dataset.

This is the so-called process of labeling sensor data, which is a transversal problem to many domains of machine learning. On the one hand it is not possible to rely on fully automated labeling processes, since we need to achieve perfect labels. Yet, on the other hand it is not possible to rely only on humans since this task is very time-consuming and prone to mistakes.

2.5.2 Frameworks and Toolkits

There is a general consensus in the NILM research community on the importance of public datasets in furthering energy disaggregation research. Nevertheless, despite the tremendous efforts in releasing public data, not many steps have been taken towards homogenizing the way these are made available to the community or how to quickly access the data. In fact, the most common way of releasing publicly accessible data continues to be, after so many years, using text files that follow a certain structure that is then passed to the users in a disparate array of formats including CSV files and plain-text files. Consequently, before any evaluation step, researchers have to understand the underlying structure of the datasets and produce code to interface with them as well as to accommodate the different evaluation metrics.

Against this background, recent times have seen serious efforts to homogenize existing datasets and provide a single interface to run evaluations. In this section we introduce some of these projects, namely the NILM Metadata proposal [35], the open-source Non-Intrusive Load Monitoring Toolkit (NILMTK) [26], [103] and the NILM-Eval framework [92].

2.5.2.1 NILM Metadata

The NILM Metadata project¹¹ authors have proposed a metadata schema with the goal of homogenizing the representation of the elements that can be found in an energy disaggregation dataset. For instance, monitored appliances, used smart-meters and the actual buildings where the collection occurred.

The proposed schema is divided in two main sub-schemas: i) a schema that describes the actual dataset; and ii) central metadata that contains general information about the appliances represented in the first component. The first sub-schema aims at modeling each individual component of the dataset, which will always produce data that varies from component to component. The second sub-schema, known as central metadata, is common to all datasets and contains detailed information about each appliance that can be modeled in the first sub-schema.

2.5.2.2 NILMTK and NILMTK v0.2

The NILMTK, released in April 2014, is an open source toolkit that was created with the ultimate goal of enabling comprehensive dataset analysis and providing a unified framework for performing cross dataset NILM performance evaluation. To this end, the authors have come up with a common data format, the NILMTK-DF (data format), which can easily accommodate the existing datasets while enabling the quick implementation of reference NILM algorithms and metrics.

Overall, the toolkit is composed of several software components written in Python¹², including parsers for a range of existing datasets, dataset diagnosis functions (e.g. gap and dropout rate detection), dataset statistics (e.g. proportion of sub-metered energy) and data pre-processing functions (e.g. down sampling, voltage normalization and top-K appliances). Additionally, NILMTK contains implementations for two reference benchmark disaggregation algorithms (combinatorial optimization, proposed by Hart [14] and Factorial

¹¹ NILM Metadata, https://github.com/nilmtn/nilm_metadata

¹² Python Software Foundation, <https://www.python.org>

Hidden Markov Models [78], [85]) as well as for several performance metrics (e.g. error in the total energy assigned and the fraction of total energy assigned correctly).

In order to assess the feasibility of their toolkit the authors performed some evaluations, including several dataset statistical analysis and energy disaggregation tests. Regarding the latter, the two default benchmark algorithms were tested against six datasets (REDD, Smart*, PSRI, AMPds, iAWE and UK-Dale) at 1-minute resolutions with the results being expressed in terms of: i) the fraction of total energy assigned correctly (FTE), ii) the normalized error in assigned power (NEP), and iii) the F-score.

The obtained results indicated that FHMM performance was superior to CO across the three metrics for REDD, Smart* and AMPds, whilst for the remaining three datasets both CO and FHMM performed similarly. Furthermore, the authors also stressed the importance of considering the time required for the training and disaggregation steps, since this can serve to decide whenever algorithms perform similarly in terms of disaggregation.

2.5.2.3 NILM-Eval

The NILM-Eval is a Matlab-based open source framework for running comprehensive performance evaluations of NILM algorithms across multiple datasets. This is very similar in scope to the NILMTK in the sense that it allows evaluations across multiple datasets with common performance metrics. Yet it was designed to facilitate the design and execution of large experiments that consider several different parameter settings for the different algorithms in repeated experiments, therefore enabling the quick evaluation and benchmark of such algorithms under different settings.

The authors of NILM-Eval have also thoroughly tested their systems' ability to evaluate and benchmark disaggregation algorithms. To this end, they have used their own dataset (ECO) to evaluate four different algorithms, two of them event-based (Baranski [68] and Weiss [40]) and two event-less (Parson [83] and Kolter [84]). These algorithms were tested under different parameter configurations and the results were reported using the system default performance metrics. The results have shown, for instance, that the event-based approaches performed better than the event-less counterparts. Furthermore, it was also

possible to learn that a data granularity of at least 1 Hz is required to reliably detect switching events of appliances. Weiss algorithm, for instance, achieves F_1 scores up to 0.92 when detecting events of cooling appliances or appliances with high changes in the consumption patterns.

2.5.3 Performance Metrics

Throughout the years the term *disaggregation accuracy* has been widely used by NILM researchers when referring to the performance of their algorithms. However, *disaggregation accuracy* has a very loose definition in a sense that it refers only to the degree of proximity between the output of a particular NILM algorithm and the true value.

Consequently, as of today it is possible to find several forms of defining the proximity between NILM results to the actual real value. However, since all of these falls under the “*disaggregation accuracy* umbrella”, there is a lack of consensus about what is actually observed. For example, in his seminal work Hart [14] used both the fraction of correctly classified power events and the fraction of total energy explained as accuracy metrics. Whereas just a few years later [49] suggested that the difference between the estimated and the true power of each appliance should be used instead.

As previously mentioned, most of the early efforts in the NILM research were devoted to event-based approaches. Consequently, several *disaggregation accuracy* metrics have been proposed to evaluate such systems. For instance, in [25] the author defined accuracy metrics for event detection (e.g. failed detection and positive predictability), classification (e.g. individual appliance and global classification accuracy) and power computation (e.g. the difference between actual and predicted energy). A similar approach, i.e., that of considering different steps in the workflow, was followed by [104] where the authors proposed three different metrics to evaluate NILM systems. These take into consideration event detector type I (when no appliance changes its state but an event is detected) and type II (when an appliance is operated but no event is detected) errors. To this end the authors defined accuracy in terms of, detection accuracy (given by the ratio between the correct and all the

detected events), disaggregation accuracy (excluding detection errors) and overall accuracy (including detection errors).

The effects of event detection type I and type II errors play an important role in the overall disaggregation results. However the proposed metrics assume that all the events are of equal importance (i.e., all appliances consume the same), which is far from being a plausible assumption. Therefore, in an attempt to quickly understand the interaction between detection errors and the actual energy consumption Anderson et al. proposed two new metrics. The total power change and the average power change, which are the sum (average in the second case) of the power changes for all the type I and type II errors [39].

Event-less approaches on the other hand, rarely rely on a separate event detection process, and instead attempt to disaggregate the total load in separate time slices using some of the methods outlined in sub-section 2.4.2. Consequently, specific metrics were created to evaluate such methods. For example, in [78] the authors propose an accuracy metric that captures the total error in the assigned energy normalized by the actual energy consumption in each time slice averaged over all appliances. Whereas in [84] the authors present an equivalent metric, but this time considering the individual appliance error rather than the average between all the appliances, thus reducing the chance of having large errors in certain time slices just because a single appliance actually performed poorly in that exact time period.

Furthermore, authors working on eventless approaches have also “re-invented” the notions of False Positives, False Negatives, True Positives and True Negatives in terms of time slice results, such that common statistical metrics like accuracy, precision, recall, sensitivity, F_1 -Score, confusion matrices and Receiver Operating Characteristic (ROC) curves can also be used to evaluate event-less approaches [26].

Table 2.3 summarizes the performance metrics that are found more often in energy disaggregation literature.

Table 2.3 – Performance metrics for NILM approaches. (EB: Event-Based; EL: Event-Less)

Metric	Description	EB	EL	References
True Positive (TP)	Whenever the system detects something as being True and the actual output is True, e.g., a power event is labeled as being triggered by appliance X and it actually was (event-based) or a time slice consumption is attributed to appliance X which is actually responsible for it (event-less).	✓	✓	[26], [46], [92]
True Negative (TN)	Whenever the system detects something as being False and the actual output is also False, e.g., no power event is detected at a given instant and no appliance has changed its state during (event-based) or for a given time slice no consumption is attributed to an appliance when that appliance is actually not consuming.	✓	✓	[24], [44], [69]
False Positives (FP)	Whenever the system detects something as being True and the actual output is False, e.g., a power event is labeled as being triggered by appliance X and it actually was not (event-based) or a time slice consumption being attributed to appliance X which is actually not working (event-less).	✓	✓	[26], [46], [92]
False Negatives (FN)	Whenever the system detects something as being False and the actual output is True, e.g., no power event is detected at a given instant but an appliance changed its state in that instant (event-based) or for a given time slice no consumption is attributed to an appliance when that appliance is actually consuming (event-less).	✓	✓	[26], [46], [92]
Accuracy	Proportion of true results (TP + TN) against the all the results (TP + TN + FP + FN)	✓	✓	[14]
Precision	Proportion of true positives against positive results (TP + FP)	✓	✓	[26]
Recall / Sensitivity	Proportion of true positives against actual positive results (TP + FN). It is also know by True Positive Rate (TPR)	✓	✓	[26], [39]
False Positive Rate	Proportion of false positives against actual negative results (FP + TN)	✓	✓	[33], [68]
F ₁ -Score	The weighted average between precision and sensitivity	✓	✓	[26], [92]

Metric	Description	EB	EL	References
Receiver Operating Characteristics / Area Under Curve	The ROC metric finds the algorithm / parameter configurations that have the best tradeoff between its TPR and FPR. The area under the ROC curve measures accuracy. An area of 1 represents a perfect test; an area of .5 represents a random (therefore worthless) test.	✓	✓	[24], [46], [104]
Total Energy Explained	TBE is the ratio between total estimated energy and actual energy used.	✓	✓	[14]
Estimated and true power difference	ETPD is the difference between estimated and actual power of each individual appliance.	✓	✓	[46], [49]
Energy Identification Rate	EIR is the ratio between estimated and actual energy	✓	✓	[46]
Detection Accuracy	EDA is the event detection accuracy including the effects of False Positives	✓	✗	[104]
Disaggregation Accuracy	DA is the disaggregation accuracy excluding the effects of False Positives	✓	✗	[104]
Overall Accuracy	OA is the disaggregation accuracy including the effects of False Positives and False Negatives	✓	✗	[104]
True Positive Percentage	TPP is the percentage of the ratio between true positives and actual true results.	✓	✗	[39]
False Positive Percentage	FPP is the percentage of the ratio between false positives and actual true results	✓	✗	[39]
Total Power Change	TPC is the sum of the deltas for all the False Positives or False Negatives	✓	✗	[39]
Average Power Change	AOC is the average of the deltas for all the False Positives or False Negatives	✓	✗	[39]
Total error in assigned energy I	TEAE I is the total error in assigned energy normalized by the actual energy consumption in each time slice averaged over all appliances	✓ ^a	✓	[78]

Metric	Description	EB	EL	References
Individual appliance error in assigned energy	LATEAE is the same as the TEAE, but for individual appliances errors	✓ ^a	✓	[84]
Total error in assigned energy II	TEAE II is the total error in assigned energy consumed over the complete duration of the data set rather than per time slice, like the two previous metrics	✓ ^a	✓	[83]
Error in total energy assigned	ETEA is the difference between the total energy assigned and the actual energy consumed by a given appliance over the dataset	✓ ^a	✓	[26]
Fraction of total energy assigned correctly	FTEAC is the overlap between the fraction of energy assigned to each appliance and the actual fraction of energy consumed by each appliance over the dataset	✓ ^a	✓	[26]
Normalized error in assigned power	NEAP is the sum of the difference between the assigned power and the actual power of a given appliance in each time slice, normalized by the appliance s total energy consumption	✓ ^a	✓	[26]
Root Mean Square Error in assigned power	RMSEAP is the root mean square error between the assigned power and the actual power of a given appliance in each time slice	✓ ^a	✓	[26], [92]
Hamming Loss	Hamming Loss measures the total information loss when appliances are incorrectly classified over the entire dataset	✓ ^a	✓	[26]
Deviation	The deviation of the inferred energy from the actual energy of a given appliance over a period of time	✓ ^a	✓	[92]

^a Although some of these metrics were specifically designed for event-less approaches it is possible to apply them to event-based approaches as long as it is possible to split the individual and aggregate consumption in time intervals (slices in the event-less nomenclature).

To summarize, it is clear from Table 2.3 that, with the exception of event detector specific metrics, it is possible to generalize all the currently available metrics so that they can be used to evaluate both NILM approaches. However, those implementations need to be supported by the data under test, such that it is possible to calculate the different performance metrics. Yet, and despite the emergence of public datasets for energy disaggregation and the recent attempts to create a common interface to access the different datasets, there are still considerable differences that make it difficult, if not impossible, to generalize most of the existing metrics across datasets.

Chapter 3 Research Scope

In the previous chapters we motivated the importance of Non-Intrusive Load Monitoring technology for optimizing domestic energy consumption. We then provided an extensive review of this technology, where we have highlighted the major challenges that still need to be addressed before it is possible to take full advantage of Non-Intrusive Load Monitoring as a technology that is able to provide energy consumption figures disaggregated by individual appliance.

In this chapter we present the research problems that we will be addressing in this thesis and in particular with more detail the research questions, the proposed research methods and the different contributions that will emerge from this work.

3.1 Research Questions

Against the background of NILM expectations and challenges that we have presented in the two previous chapters, we are now drawn to the two research questions (RQ) of this PhD thesis, which happen to be intrinsically related to the real world applicability of Non-Intrusive Load Monitoring:

1. **What are the practical issues of deploying a NILM and eco-feedback solution in real-world settings?**
2. **How do performance metrics compare to each other when applied to event detection and event classification algorithms?**

What are the practical issues of deploying a NILM and eco-feedback solution in real-world settings?

As it was previously mentioned, advanced metering solutions such as NILM holds the potential to leverage the creation and deployment of novel energy efficiency services and programs. Yet, the current state of the art presents little or no evidence about how to develop and deploy this technology in real-world settings or about how the underlying algorithms would perform once deployed outside the controlled laboratory environment.

As such, in **RQ 1** we want to explore the potential of using a bespoke NILM solution to support the development and deployment of long-term energy efficiency programs using eco-feedback technology. More precisely we wish to:

- Understand the **technical** and **social constraints** of developing and deploying such a solution in real world settings. The former includes for example, hardware and software requirements, whereas the latter includes issues related to the security and intrusiveness of the different energy monitoring solutions.
- Identify and understand the possible **costs** associated with developing and deploying a solution like ours. These include, equipment acquisition costs, the energy consumed by such devices and the costs associated with storing the obtained data in the cloud.

How do performance metrics compare to each other when applied to event detection and event classification algorithms?

As it was already seen in the previous chapter, recent years have seen considerable research efforts being made towards producing meaningful comparisons between different NILM algorithms, which are greatly reflected by the emergence of public datasets for energy disaggregation, and the recent attempts to provide single interfaces to produce such evaluations and benchmarks.

Nevertheless, and despite all the reported efforts, very little research has targeted the fact that as of today it is still not possible to find a proven and formally accepted set of metrics to measure and report the performance of the many proposed energy disaggregation methods.

Consequently, in **RQ 2** we propose to explore and understand the challenges of defining a consistent set of performance metrics for event detection and event classification problems. More concretely, we propose to:

- Investigate the existence of **clusters** and **relationships** between performance metrics when these are applied to each individual problem.
- Investigate if, when applied to event detection and event classification problems, the same performance metrics show a **similar** or a **distinct** behavior.

3.2 Research Method

In this section we present the research methods that will be followed to provide the answers to the two research questions of this thesis.

Research Question 1

In order to answer the first research question, we have iteratively developed and deployed two hardware and software platforms for unobtrusive energy monitoring and eco-feedback research. These two platforms were created exploring the practical and technical feasibility of low-cost non-intrusive techniques to extract detailed consumption information from the aggregate source and provide eco-feedback to the householders.

These two platforms were deployed in a total of 50 homes for consecutive periods that lasted between **6** and **18 months**. During that period the system was constantly monitored and perfected and several eco-feedback studies, including qualitative interviews and surveys, were conducted in the context of a large sustainability research project¹³.

Here, however, we focus on the practical issues of building, deploying and maintaining such systems for long periods of time. By iteratively developing and deploying our sensing and eco-feedback infrastructures we managed to build upon previous findings and lessons learned to gain a deeper understanding on how to create, deploy and maintain such systems.

¹³ SINAIS research project, <http://sinais.m-iti.org>

Concurrently, we gained valuable insights regarding what are some of the most relevant costs associated with running such experiments, which are seldom reported in literature.

When taken together, the different insights and lessons learned from the three deployments represent an advancement in the state of the art in the live deployment of NILM systems, in particular when these are targeted at eco-feedback research.

Research Question 2

To the best of our knowledge, literature devoted to NILM performance evaluation reports mostly in the event-less approaches [26], [92], [105], which we argue, in part happens due to the lack of proper datasets to evaluate event-based approaches.

A few exceptions to this are the works of Beckel et al. [92] and Czarnek et al. [106] that managed to evaluate event-based approaches after manually extracting labeled data from the ECO and REDD datasets respectively. Lastly, in [39] Anderson et al. used BLUED to evaluate performance metrics for event detection algorithms.

As such, and given the fact that there is very little work done in evaluating event-based approaches, in this work we are interested in evaluating algorithms under this category with a particular focus in the high-frequency (≥ 50 Hz) approaches. Stated more precisely, we will analyze the behavior of performance metrics when applied to evaluate event detection and event classification algorithms over multiple datasets.

To this end, we will first train and evaluate five different event detection and six classification algorithms across different datasets using pre-defined sets of performance metrics. Regarding the event detection, we execute a parameter sweep of the five detection algorithms across four datasets. For the classification tasks we execute a parameter and feature sweep of six supervised classification algorithms against eleven datasets. Then, for each resulting model we compute the respective performance metrics.

Once all the performance metrics are calculated we investigate the existence of correlations between the results obtained with each performance metric. More particularly, we study the existence of linear (Pearson) and rank (Spearman) correlations between pairs of

metrics. The former indicates the existence and direction of any linear relationships, whereas the later assesses the existence of monotonic relationships (results tend to change together but not necessarily at a linear rate).

After the initial correlation analysis, we further explore the metrics correlation using hierarchical clustering. More precisely we explore the distances between metrics using *dendrograms* in which the different metrics are joined together in a hierarchical fashion from the closest, that is most similar, to the furthest apart, i.e., the most different.

Ultimately, the in-depth analysis of the pairwise correlations and the resulting clusters represents an advancement of the state of the art towards defining a consistent set of metrics to evaluate event detection and event classification algorithms. For example:

- The different clusters / groups formed by the performance metrics will unveil if there are performance metrics that will yield the same ranks (i.e., if they will choose the same models). Likewise, this will also unveil the cases in which the performance metrics rank the different models in totally different directions.
- The different metric clusters / groups will allow us to understand if the theoretical guarantees of such metrics hold true for these two machine-learning problems. In other words, it will be possible to understand to what extent the different performance metrics are compatible with event detection and event classification problems.

3.3 Research Contributions

When taken together, the contributions of this thesis fall under three broad categories: i) tools and datasets to enhance current energy disaggregation and eco-feedback research, ii) advances to the state of the art in the live deployment of NILM and eco-feedback technology, and iii) advances to the state of the art towards defining a consistent set of metrics for event detection and event classification algorithms.

Tools and Datasets

We present one tool and two datasets:

- The *Energy Monitoring and Disaggregation Data Format* (EMD-DF) [107] is a data model and file format that was created with the intention of providing a unique interface to create, manage and access energy disaggregation datasets. EMD-SF is open-source and can be freely accessed from <http://aveiro.m-iti.org/software>.
 - The *dataset for electric energy research* (SustData) [47], is a public dataset that gathers in the same place all the data that was collected during the three real world deployments of our energy monitoring and eco-feedback research platforms.
 - The *dataset for electric energy disaggregation research* (SustDataED) [108], is an extension to the original SustData dataset, and consists of aggregate and individual appliances electric energy consumption taken from a single-family residence in Portugal for the duration of 10 days.
- SustData and SustDataED are freely available in <http://aveiro.m-iti.org/data>.

Live Deployments of NILM and Eco-Feedback Technology

The research contributions under this category are threefold:

- We *designed and developed two hardware and software platforms for energy monitoring and eco-feedback research*. Our platforms include, among others, modules to interface with different data acquisition boards, perform power calculations and communicate those measurements.
- The second platform was later extended to support the creation of energy disaggregation datasets and the implementation of NILM performance evaluation pipelines. The implementations details are out of the scope of this thesis, and can be found in the following publications: [109]–[111]. It is also possible to access and download the source code from <http://aveiro.m-iti.org/software>.
- We report on more than five years of experience in deploying and maintaining such platforms in real world scenarios. More specifically, we *highlight the different*

technical and social challenges that NILM and eco-feedback researchers must address when conducting long-term studies.

- We *identify the costs associated with running real world experiments* with our two energy monitoring and eco-feedback platforms. More concretely, we investigate the costs associated with hardware acquisition, the energy required to run the energy monitors and the costs of storing the data that are generated. We then *compare our costs with those of two hypothetical solutions*, namely a multi-sensor solution and a NILM solution with an embedded microprocessor.

Performance Metrics for Event Detection and Event Classification

Algorithms

We present three research contributions in this category:

- In the first contribution, we *analyze experimentally the behavior of a number of performance metrics in several event detection and event classification scenarios*, identifying clusters and relationships between the different measures and problems.
To state more concretely, we analyze the behavior of 24 different performance metrics for event detection, which is done against five algorithms across four datasets. We then perform a similar analysis to 18 distinct performance metrics for classification algorithms. This is done for six classification algorithms against 11 datasets.
- In the second contribution, we *present a new probabilistic event detection algorithm*, the Simplified Log Likelihood Detector (SLLD). The proposed algorithm is described in subsection 6.1.1.3.
- In the third contribution we release the two manually labeled event detection datasets to the NILM research community. The two datasets can be downloaded from <http://aveiro.m-iti.org/data>. Additional details are available in section B.1 of Appendix B.

3.4 Publications

In this section we list the publications that emerged from the work in this thesis. We also present a list of publication that are currently under preparation.

2011

1. N. J. Nunes, L. Pereira, F. Quintal, and M. Bergés, "**Deploying and evaluating the effectiveness of energy eco-feedback through a low-cost NILM solution**", *International Conference on Persuasive Technology* (Persuasive '11), Columbus, OH, USA, 2011. [full paper]

2012

2. L. Pereira, and N. J. Nunes, "**Low cost framework for non-intrusive home energy monitoring and research**", *International Conference on Smart Grids and Green IT Systems* (SMARTGREENS '12), 1, vol. 1, Porto, Portugal, SciTePress, pp. 191-196, 04/2012. [short paper]
3. L. Pereira, F. Quintal, N. J. Nunes, and M. Bergés, "**The design of a hardware-software platform for long-term energy eco-feedback research**", *ACM SIGCHI symposium on Engineering Interactive Computing Systems* (EICS '12), Copenhagen, Denmark, ACM, pp. 221–230, 06/2012. [full paper, **chapter 5**]
4. F. Quintal, L. Pereira, and N. J. Nunes, "**A long-term study of energy eco-feedback using non-intrusive load monitoring**", *International Conference on Persuasive Technology* (Persuasive '12), pp. 49, 06/2012. [poster]
5. F. Quintal, V. Nisi, N. J. Nunes, M. Barreto, and L. Pereira, "**HomeTree – An art inspired mobile eco-feedback visualization**", *Advances in Computer Entertainment Conference* (ACE '12), Springer Berlin/Heidelberg, pp. 545–548, 11/2012. [demo]

6. L. Gouveia, L. Pereira, M. Scott, and I. Oakley, "**Eco-Avatars: Visualizing disaggregate home energy use**", *ACM conference on Designing Interactive Systems* (DIS '12), Newcastle, UK, 06/2012. [demo]

2013

7. L. Pereira, F. Quintal, M. Barreto, and N. J. Nunes, "**Understanding the Limitations of Eco-feedback: a One Year Long-term Study**", *International Conference on Human Factors in Computing & Informatics* (SouthCHI '13) [Acceptance Rate: 22%], vol. 7947, Maribor, Slovenia, Springer Berlin Heidelberg, pp. 237-255, 07/2013. [full paper]
8. L. Pereira, "**Towards Automating the Performance Evaluation of Non-Intrusive Load Monitoring Systems**", *International Conference on ICT for Sustainability* (ICT4S '13), Zurich, Switzerland, 02/2013. [doctoral consortium]
9. F. Quintal, L. Pereira, N. J. Nunes, V. Nisi, and M. Barreto, "**WATTSBurning: design and evaluation of an innovative eco-feedback system**", *IFIP TC13 Conference on Human-Computer Interaction* (INTERACT '13), vol. 8117, Cape Town, South Africa, Springer Berlin Heidelberg, pp. 453-470, 08/2013. [full paper]
10. F. Quintal, M. Barreto, N. J. Nunes, V. Nisi, and L. Pereira, "**WattsBurning on my mailbox: a tangible art inspired eco-feedback visualization for sharing energy consumption**", *IFIP TC13 Conference on Human-Computer Interaction* (INTERACT '13), Cape Town, South Africa, Springer Berlin Heidelberg, 08/2013. [short paper]

2014

11. L. Pereira, F. Quintal, R. Gonçalves, and N. J. Nunes, "**SustData: A Public Dataset for ICT4S Electric Energy Research**", *International Conference on ICT for Sustainability* (ICT4S '14), Stockholm, Sweden, Atlantis Press, 08/2014. [full paper, chapter 4]

12. L. Pereira, N. J. Nunes, and M. Bergés, "**SURF and SURF-PI: A File Format and API for Non-Intrusive Load Monitoring Public Datasets**", *ACM International Conference on Future Energy Systems (e-Energy '14)*, Cambridge, UK, ACM, 06/2014. [short paper, **chapter 4**]
13. M- Scott, L. Pereira, and I. Oakley, "**Show Me or Tell Me: Designing Avatars for Feedback**", *Interacting with Computers*, Oxford University Press, 03/2014. [journal]
14. F. Quintal, L. Pereira, N. J. Nunes, and V. Nisi, "**What-a-Watt : Where does my electricity comes from?**", *International Working Conference on Advanced Visual Interfaces (AVI '14)*, Como, Italy, 2014. [demo]
15. M. Barreto, A. Szóstek, E. Karapanos, N. J. Nunes, L. Pereira, and F. Quintal, "**Understanding families' motivations for sustainable behaviors**", *Computers in Human Behavior*, vol. 40, pp. 6 - 15, 11/2014. [journal]

2015

16. L. Pereira, and N. J. Nunes, "**Semi-Automatic Labeling for Non-Intrusive Load Monitoring Datasets**", *IFIP Conference on Sustainable Internet and ICT for Sustainability (SustainIT '15)*, Madrid, Spain, IEEE Explore, 04/2015. [poster, **chapter 7**]
17. L. Pereira, and N. J. Nunes, "**Towards Systematic Performance Evaluation of Non-Intrusive Load Monitoring Algorithms and Systems**", *IFIP Conference on Sustainable Internet and ICT for Sustainability (SustainIT '15)*, Madrid, Spain, IEEE Explore, 04/2015. [doctoral consortium, **chapter 6**]
18. N. J. Nunes, L. Pereira, and V. Nisi, "**Towards using Low-Cost Opportunistic Energy Sensing for Promoting Energy Conservation**", *Fostering Smart Energy Applications Workshop (FSEA 2015)*, Bamberg, Germany, University of Bamberg Press, 09/2015. [position paper]
19. F. Quintal, L. Pereira, C. Jorge, and N. J. Nunes, "**EnerSpectrum: Exposing the source of energy through plug-level eco-feedback**", *IFIP Conference on Sustainable*

Internet and ICT for Sustainability (SustainIT '15), Madrid, Spain, IEEE Explore, 04/2015. [poster]

20. F. Quintal, L. Pereira, N. Nunes, and V. Nisi, "**What-a-Watt: Exploring Electricity Production Literacy Through a Long Term Eco-Feedback Study**", *IFIP Conference on Sustainable Internet and ICT for Sustainability* (SustainIT '15), Madrid, Spain, IEEE Explore, 04/2015. [short paper]

2016

21. M. Ribeiro, L. Pereira, F. Quintal, and N. Nunes, "**SustDataED: A Public Dataset for Electric Energy Disaggregation Research**", *International Conference on ICT for Sustainability* (ICT4S '16), Amsterdam, The Netherlands, Atlantis Press, 08/2016. [poster, **chapter 4**]

Under preparation

22. L. Pereira, N. J. Nunes, and M. Bergés, "**EMD-DF: An Energy Monitoring and Disaggregation Data Format**". [chapter 4]
23. L. Pereira, M. Ribeiro, N. J. Nunes, "**SustDataED: A public dataset for energy disaggregation research**", [chapter 4]
24. L. Pereira, N. J. Nunes, and M. Bergés, "**Semi-Automatic Labeling of Public Energy Disaggregation Datasets**", [chapter 4]
25. L. Pereira, M. Ribeiro, N. J. Nunes, M. Bergés, "**Collaborative Labeling of Public Energy Disaggregation Datasets**", [chapter 4]
26. L. Pereira, M. Ribeiro, N. J. Nunes, M. Bergés, "**Hardware and Software Platforms to Collect and Label Energy Disaggregation Datasets**", [chapter 4]
27. L. Pereira, R. Gonçalves, and N. J. Nunes, "**Understanding the data management issues of deploying NILM and eco-feedback technology in real world scenarios**", [chapter 5]

28. L. Pereira, N. J. Nunes, and M. Bergés, **“Understanding the practical issues of deploying NILM technology in the real world: lessons learned from three long-term deployments”**. [chapter 5]
29. L. Pereira, M. Bergés, and N. J. Nunes, **“An experimental comparison of performance metrics for event detection algorithms in non-intrusive load monitoring systems”**. [chapter 6]
30. L. Pereira, M. Bergés, and N. J. Nunes, **“An experimental comparison of performance metrics for event classification algorithms in non-intrusive load monitoring systems”**. [chapter 6]

Chapter 4 Tools and Datasets

In this chapter we describe the software tools and datasets that emerged from this thesis. More concretely, we present the Energy Monitoring and Disaggregation Data Format (EMD-DF) [107], the SustData dataset for electric energy research [47] and the SustDataED dataset for energy disaggregation research [109].

4.1 EMD-DF: Energy Monitoring and Disaggregation Data Format

As it was already discussed in this thesis, only recently there has been a serious effort to homogenize the existing datasets and provide a single interface to run NILM evaluations [26], [35] to which we wish to contribute by proposing EMD-DF, a common file format and programming interface that supports the creation and manipulation of energy disaggregation datasets.

EMD-DF supports embedded annotations and metadata, and features an application-programming interface (API). Next we describe the underlying data model, the data structure of the current implementation and the corresponding API. We then highlight the limitations of the current version and outline future work.

4.1.1 Data Model

In the current version of EMD-DF, we have identified and modeled three main data entities that should be present in a dataset for energy disaggregation. They are: i) consumption data, ii) ground-truth data, and iii) data annotations. Figure 4.1 shows an illustration of the

proposed data model using the Unified Modeling Language (UML) notation; here we will give a brief overview and additional details will be provided along this section.

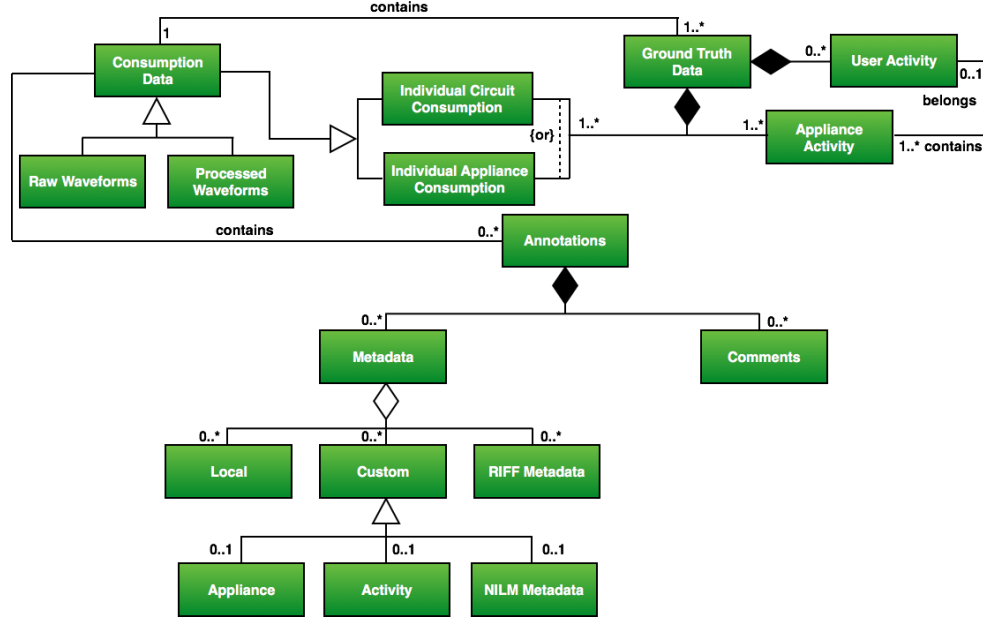


Figure 4.1 – EMD-DF: Data model overview

Starting from the left hand side of the diagram, we have the **consumption data** entity representing all the data elements that refer to energy consumption. Consumption data can be of two different types: i) raw waveforms, i.e., current and voltage; or ii) processed waveforms, i.e., different power metrics like real and reactive power.

Moving to the right-hand side of the diagram we have the **ground-truth data** entity. This entity is mandatory, and can be of four different types: i) individual appliance consumption, ii) individual circuit consumption, iii) appliance activity, and iv) user activity. Individual appliance and individual circuit consumption are themselves a special type of **consumption data** and are used to train, test and validate event-less approaches. Appliance activities, on the other hand, provide information about the power events that exist in the dataset (e.g., timestamp and label) and are required to train, test and validate event-based approaches.

We have also introduced the concept of user activities, which in very simple terms refer to actions that people perform involving the use of electric appliances, e.g., *doing the laundry*

(involves clothes washer, clothes dryer and iron) or *preparing a meal* (oven, stove, microwave, aid choppers, blenders, etc.). As it can be observed in the diagram, *one individual appliance activity can only be associated with one user activity*, otherwise the total consumption of the user activities will be larger than the total consumption of the individual appliances, thus introducing inconsistency to the data model.

Lastly, we have the **data annotations** entity. These can be either metadata or comments, and are not mandatory. We have defined three different types of metadata annotations, namely: i) local metadata, which refers to specific samples in the consumption data, ii) custom metadata that are defined by the dataset creator and can serve multiple purposes, and iii) RIFF Metadata, which is composed of metadata chunks defined by the RIFF file container format itself.

4.1.2 Data Structure

In order to represent the different entities in EMD-DF, we propose an extension of the well-known Waveform Audio File Format (WAVE¹⁴) that was originally created to store audio data. WAVE is an application of the Resource Interchange File Format (RIFF¹⁵) standard in which the file contents are grouped and stored in separate chunks, each of which following the pre-defined format, as shown in Figure 4.2.

FourCC
Size
Data Padding byte

Figure 4.2 - RIFF chunk format definition

The Four Character Code (**FourCC**) is the 4 bytes chunk identifier (e.g. ‘fmt ‘ and ‘data’). **Size** is an unsigned, little-endian 32-bit integer that contains the length of the actual chunk Data. A **padding byte** is added whenever the chunk length is not even.

¹⁴ WAVE file format: <http://fileformats.archiveteam.org/wiki/WAV>

¹⁵ RIFF file format, <http://fileformats.archiveteam.org/wiki/RIFF>

The idea of using an audio format to represent electric energy data was inspired by the single-house energy monitoring platform (read more about this in Chapter 5), where we use a soundcard to perform the acquisition of the current and voltage signals. From the widely available audio file formats¹⁶ (e.g., AIFF, WAV, AU, and FLAC), we have opted to extend the WAVE, since it has a number of properties that we believe are desirable in the context of energy disaggregation datasets. More particularly:

- The waveform data and annotations are all stored in a **single compact file**, thus limiting the number of artifacts to be managed;
- The waveform data is represented in individual channels, hence keeping a **clear separation between measurements**;
- The resulting **files are optimized** to have very little overhead. Furthermore, since the sampling rate is fixed, only the initial timestamp is necessary to obtain the time of the remaining samples;
- It is an **uncompressed lossless format**, i.e., all the original values of the data are kept untouched;
- It is **possible to extend** the format at any time with additional chunks without breaking the file consistency, i.e., it will always be recognized as a wave file, hence offering backwards compatibility;
- There are already a diversity of **mature programming interfaces** in many programming languages, thus facilitating the manipulation of the data elements in the datasets and eventually the expansion and portability of EMD-DF to other programming environments.

4.1.2.1 WAVE: File Format Definition

Overall, any file that follows the RIFF standards is itself a RIFF chunk, which then can contain further sub-chunks: hence, the first four bytes of a correctly formatted RIFF file will always spell out "R", "I", "F", "F" as shown in Figure 4.3.

¹⁶ Wikipedia entry for audio file formats: https://en.wikipedia.org/wiki/Audio_file_format

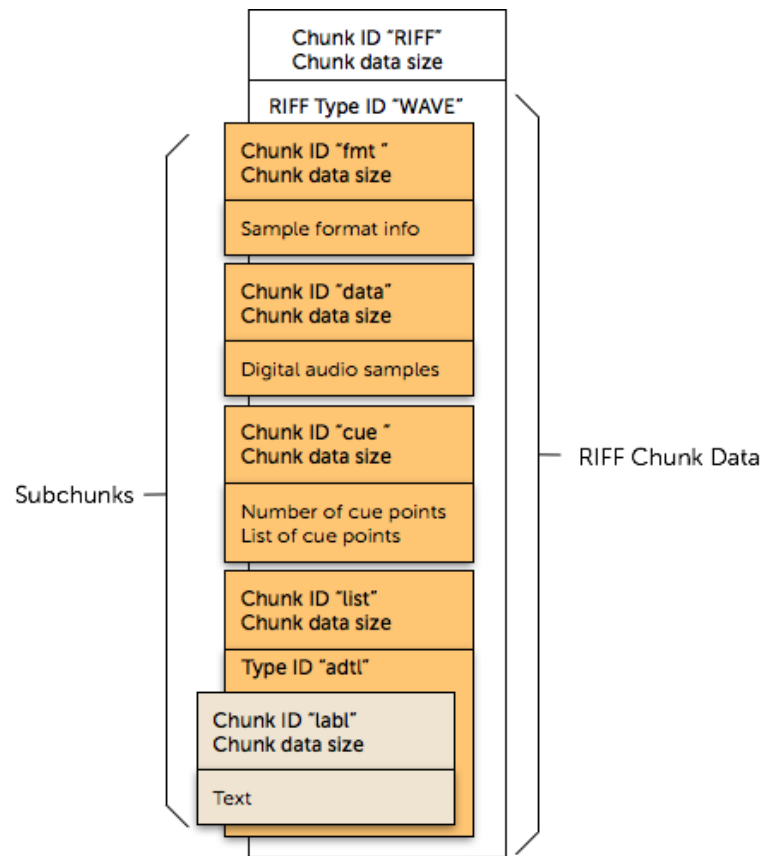


Figure 4.3 – RIFF-WAVE file format chunk structure

A wave file is composed of thirteen chunks, three of which are mandatory. All the WAVE chunks are briefly described in Table 4.1, with special focus to those that are reused in the EMD-DF data model. For more details about the RIFF standard please refer to the original project documentation in [112].

Table 4.1 – List of chunks that compose the RIFF-WAVE file format

Name	FourCC	Description	Parent	Mandatory	EMD-DF
RIFF	'RIFF'	This is the main chunk and is mandatory for every file that is based on the RIFF standard.	---	✓	✓
WAVE ^a	'WAVE'	Identifies the contents of the RIFF chunk as being of the type wave.	RIFF	N/A	✓
Format	'fmt '	Defines the data format, e.g. sampling rate, sample size and bits and the number of channels.	RIFF	✓	✓
Data	'data'	Waveform data can be stored as a single contiguous array of interleaved samples or as a discrete sequence of blocks of samples and silence wrapped in a 'wavl' chunk.	RIFF	✓	✓
Silent	'slnt'	Represents silence and is defined as a count of silence samples.	wavl	✗	✗
Wave List	'wavl'	Wraps sequences of data and silence chunks.	data	✗	✗
Fact	'fact'	Stores information about how the waveform data is organized. It is mandatory when the waveform data is stored in a 'wavl' chunk and for all compressed audio formats.	RIFF	✗	✗
Cue	'cue '	Identifies a series of positions in the waveform data as as having additional information associated with them. There is at most one queue chunk per wave file, and it is followed by a list of queue points.	RIFF	✗	✓
List	'list'	This is a wrapper for chunk, which in the particular case of Wave files is an associated data list ('adtl').	RIFF	✗	✓
Associated Data List ^a	'adtl'	Identifies a list that contains individual information attached to the cue points defined in the cue chunk.	list	N/A	✓
Label	'label'	Associates a text label to a specific cue point. Must be defined inside the associated data list chunk.	adtl list	✗	✓
Note	'note'	Same as label, but usually contains comment text for a specific cue point.	adtl list	✗	✓

Labeled Text Chunk	'text'	Associates a text comment to specific regions of waveform data. A region is a cue point whose duration in samples is defined in this chunk. Must be defined inside the associated data list chunk.	adt list	✗	✓
Embedded File Information	'file'	Contains information described in other file formats (e.g. ASCII text files) that is associated with a particular cue point.	adt list	✗	✗
Playlist	'plst'	Specifies a play order for a series of cue points.	RIFF	✗	✗
Info ^a	'INFO'	Identifies a list that contains the info chunks defined by the RIFF standard [112].	RIFF	N/A	✗

^a 'WAVE', 'adt1' and 'INFO' are chunk identifiers.

4.1.2.2 EMD-DF: File Format Definition

The EMD-DF data structure is currently composed of 18 chunks each one containing its own header and data bytes. One is inherited from the RIFF standard (Info), eight from the WAVE format (Format, Data, Cue, List, Associated Data List, Label, Note and Labeled Text Chunk), and the remaining are custom chunks. Table 4.2 provides a description of the custom chunks that can be added to EMD-DF data structure.

Table 4.2 - List of chunks that compose EMD-DF

Name	FourCC	Description	Parent	Mandatory
Config ^a	‘CNFG’	Identifies a list that contains ED3M specific configurations.	RIFF	✓
Timestamp	‘TMSP’	Unix timestamp of the first sample in the waveform data.	CNFG LIST	✓
Timezone	‘TMZN’	Timezone of the place where the data was collected.	CNFG LIST	✓
Sampling rate	‘SPRT’	Sampling rate of the waveform data (overwrites the original value in the format chunk if the actual sampling rate is lower than 1 Hz).	CNFG LIST	✓
Calibration Constants	‘CHCC’	Calibration constants to recreate the original values of the waveform data. One constant for each channel.	CNFG LIST	✓
Annotation ^a	‘ANNO’	Identifies a list that contains metadata and comment chunks.	RIFF	✗
Metadata	‘META’	This is metadata specific chunk. Must be contained in the ‘ANNO’ list.	ANNO LIST	✗
Comment	‘COMT’	This is a comment specific chunk and must be specified within the “ANNO” list.	ANNO LIST	✗

^a ‘CONFIG’, ‘ANNO’, and ‘META’ are chunk identifiers.

Next, we present in detail the different chunks that compose the EMD-DF data structure. We first describe how the data format is defined in the **Format** and **Config** chunks. Then we show how the actual power measurements are stored and supplemented with the different embedded annotations, namely: i) individual appliance activity, ii) user activities and iii) comments and metadata.

4.1.2.2.1 Waveform data format

In the EMD-DF data structure the waveform data (i.e., consumption data) must be defined in the **format** chunk ('fmt '). This is inherited from the WAVE format and consists of the following fields: i) *sample size in bits* (8, 16, 24, 32 or 64 bits); and ii) *number of individual channels* (greater or equal to 1).

Additionally, all the sub-chunks defined in the **Config** list chunk ('CNFG') are mandatory. More precisely: i) *timezone* (the time zone of the location where the data was collected), ii) *timestamp* (the Unix timestamp of the first sample in the waveform data), iii) *sampling rate* (the number of samples per second in the waveform data), and iv) *calibration constants* (zero or one for each waveform channel).

The calibration constant chunks are associated to each channel in ascending order, and for the model to be valid the number of chunks must be zero (i.e., no calibration is needed) or equal to the number of individual channels.

4.1.2.2.2 Consumption data

The waveform data is stored uncompressed in the **Data** chunk. If only one metric needs to be represented (this is the case in most individual appliance and circuit ground-truth data), the samples are stored consecutively; otherwise the samples are stored interleaved.

Each sample S is represented by an integer with a value between -1 and 1. The size of each sample is equal to the smallest number of bytes required to represent the sample size specified in the format chunk. Samples are stored in little endian format (i.e., the least significant byte is stored first). The bits that represent the sample amplitude are stored in the most significant bits of S , and the remaining bits are set to zero.

4.1.2.2.3 Individual appliance activity

Individual appliance activities correspond to the changes in the power consumption that are triggered by different appliance turning *ON*, *OFF*, or changing their working mode (e.g., low

to high). Each activity has a corresponding timestamp that is mapped to a position in the waveform data using equation (4.1).

$$position = \frac{actual_timestamp - initial_timestamp}{\frac{1}{f} \times 1000} \quad (4.1)$$

Where **actual_timestamp** is the timestamp in milliseconds that we want to map to an audio position, **initial_timestamp** is the timestamp in milliseconds of the first sample in the dataset and **f** is the sampling rate of the waveform data. For example, if the elapsed time since the initial sample is 3500 milliseconds (3.5 seconds) and the signal frequency is 60 Hz, the position in samples will be 210, but if the signal frequency is 12 kHz the position will be 42000.

In order to embed these activities the **Cue**, **Associated data list** and **Label** chunks are used as follows: First, for each individual appliance activity an entry is added to the **Cue** chunk. Then, for each entry in the cue chunk, a **Label** chunk is added to the **Associated Data List** chunk. Each label chunk consists of a sample position in the waveform data and a JSON formatted string with the details of that activity. For example, the following JSON string corresponds to a refrigerator activity that was mapped to position 19394633:

```
{
  "ID": 1101,
  "Type": 1,
  "Position": 19394633,
  "Timestamp": "2011-10-24 05:45:57.040",
  "App_ID": 111,
  "App_Label": "Refrigerator"
}
```

Figure 4.4 - Example of an Appliance Activity Annotation

Where **ID** is the unique identifier of the appliance activity in the whole dataset (independently of the sampling rate used), **Type** identifies if there was an increase in consumption (1) or a decrease (-1), the **Position** refers to the position in the waveform (in samples), the **Timestamp** is the date and time of the power change, the **App_ID** is the identifier of the appliance that is responsible for this event and the **App_Label** is the corresponding appliance name.

Figure 4.5 shows a graphical representation of one of such activities in the BLUED dataset. In the top figure the activity is supplemented in the real and reactive power traces at 60 Hz. The bottom image shows the same activity (**ID: 1101**) supplemented on the respective current and voltage model, sampled at 12 kHz.

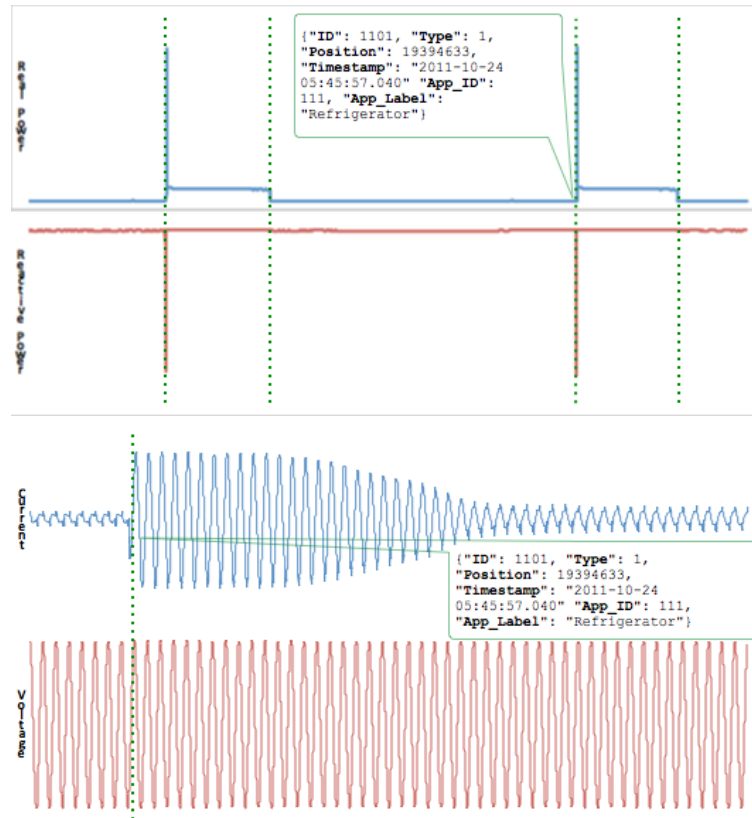


Figure 4.5 - Markers for the refrigerator events in the P & Q file at 60 Hz (top), and one marker at the I & V file at 12 kHz (bottom)

4.1.2.2.4 User activity

Current research related to the human side of energy monitoring suggests that householders tend to associate their consumption with everyday activities (e.g., cooking, leisure, cleaning) [113], [114]. Consequently, providing a way to extract such activity information from energy disaggregation datasets would leverage research on top of the electricity data focusing on disaggregating the energy consumption by more meaningful human activities instead of individual appliances.

In this context, each user activity refers to an action that people perform involving the use of electric appliances, e.g., doing the laundry (involves clothes washer, clothes dryer and iron) or preparing a meal (oven, stove, microwave, aid choppers, blenders, etc.).

Such activities are represented using the **Cue**, **Associated data list** and **Labeled text** chunks. Each user activity contains two timestamps (start and end), which are also mapped to audio samples using equation (4.1). JSON is used to encode the activity details, as shown in the following snippet for the “*working on the computer*” activity that involves using the desktop computer, one monitor and a printer.

```
{
  "ID": 10020,
  "Activity_ID": 111,
  "Activity_Label": "Working on the computer",
  "Start_Position": 19394633,
  "End_Position": 19394633,
  "Start_Timestamp": "2011-10-24 05:45:57.040",
  "End_Timestamp": "2011-10-24 06:23:18.056",
  "Appliance_Activity_IDs": [1101, 1109, 1203],
  "Total_Power": 1000
}
```

Figure 4.6 - Example of user activity annotation

Where **ID** is the unique identifier of each user activity in the whole dataset, **Activity_ID** identifies this activity, **Activity_Label** is the corresponding activity name, **Start_Position** is where the activity starts (reciprocally for the **End_Position**), **Start_Timestamp** is the date and time when the activity starts (reciprocally for **End_Timestmap**), **Appliance_Activity_IDs** is an array with the *ids* of all the individual appliance activities that constitute this activity and the **Total_Power** refers to the amount power consumed during this activity (sum of the power consumed by each individual appliance activity).

4.1.2.2.5 Metadata

Local metadata

Local metadata is used to supplement specific waveform samples with custom annotations. These are created using the **Cue**, **Additional Data List** and **Note** chunks as follows: First, for each local annotation that we wish to create, an entry is added to the **Cue** chunk. Then, for each entry in the **cue** chunk, a **note** chunk is added to the **associated data list** chunk.

Local metadata annotations can be used for instance to supplement datasets with details like the instant when a new appliance is added or removed from the electric circuit. Each local annotation is associated to a position in the waveform data and its content is encoded in a JSON string as shown in Figure 4.7.

```
{
  "ID": 1101,
  "Position": 19394633,
  "Timestamp": "2011-10-24 05:45:57.040",
  "Text": "New refrigerator was added"
}
```

Figure 4.7 - Example of a Local Metadata Annotation

Custom metadata

These custom chunks can be used to enrich datasets with custom metadata according to the author needs. These chunks are added using the **Annotation List** and **Metadata** chunks. The content of such chunks do not follow any specific rule, yet it must be encoded in JSON and always include the **ID** and **Metadata_Label** fields. Currently EMD-DF fully supports three different custom metadata types: i) appliances; ii) user activities; and iii) NILM metadata project annotations.

Appliances metadata keeps a list of the appliances that co-exist in the dataset, including the appliance characteristics like brand, model, energy consumption and energy efficiency rating. Figure 4.8 shows a possible JSON string for the appliances metadata chunk.

```

{
  "ID": 10021,
  "Label": "Dataset Appliances",
  "Appliances": [
    {
      "ID": 111,
      "Label": "Refrigerator",
      "Brand": "Some brand",
      "Model": "Some model",
      "Energy_Consumption": 200,
      "Energy_Efficiency_Rating": "E"
    },
    {...}
  ],
}

```

Figure 4.8 – Appliances custom metadata annotation

User activities metadata keeps a list of the user activities that are present in the dataset, including a list of the appliances that can be associated with each activity. An example is provided in Figure 4.9.

```

{
  "ID": 10022,
  "Label": "User Activities",
  "User_Activities": [
    {
      "ID": 111,
      "Label": "Working on the computer",
      "Appliance_IDs": [110, 211, 105]
    },
    {...}
  ],
}

```

Figure 4.9 - User activities custom metadata annotation

Lastly, it is also possible to supplement datasets with annotations from the NILM metadata project. To this end we have defined the **NILM Metadata Project** annotation that can be used to embed the content of the different YAML files that compose the NILM Metadata project, in a metadata chunk (see Figure 4.10 below). Alternatively, we could embed only the references to the YAML files. Yet, this would add undesired external

dependencies to the datasets and consequently violate our principle of keeping the datasets with the minimum number possible of files.

```
{
  "ID": 10023,
  "Label": "NILM Metadata Project",
  "Dataset": "YAML content",
  "Meter_Devices": "YAML content",
  "Building": [
    "Building_1": "YAML content",
    "Building_2": "YAML content"
  ]
}
```

Figure 4.10 – NILM Metadata project custom chunk

RIFF metadata

Being a direct application of the RIFF standard, EMD-DF supports by default all the RIFF metadata sub-chunks, which are defined in the **Info** chunk. Yet, most of these chunks are targeted at audio and other media file types. Consequently only a subset of these chunks is considered in our data model. These are listed in Table 4.3.

Table 4.3 - List of RIFF metadata chunks

Name	FourCC	Description
File Creator	'IART'	The name of the file creator
Commissioner	'ICMS'	The name of the dataset commissioner
Comments	'ICMT'	Free text comment
Copyright	'ICOP'	Dataset copyright notice
Creation Date	'ICRD'	Data of dataset creation
Keywords	'IKEY'	A list of keyword that can help describe the dataset contents
Name	'INAM'	The name of the dataset
Product	'IPRD'	Original propose of the file
Subject	'ISBJ'	Contents of the file (e.g. current and voltage waveforms)
Software	'ISFT'	The name of the software that was used to create the file

Name	FourCC	Description
Source	'ISRC'	Original (person / organization) source of the file
Source Form	'ISRF'	Original form of material (.csv /.txt)

4.1.2.2.6 Comments

Custom comment chunks consist of free form text and are created using the **Annotation List** and **Comment** chunks. These can be used to add any kind of comments, for example, add a comment containing the historic of previous performance evaluations results on that particular file or dataset. Another example would be, adding a comment regarding some external event that could have affected the data.

4.1.3 Application Programming Interface

In order to facilitate the creation and manipulation of dataset that follow the proposed model, we implemented a number of functions that are summarized in Table 4.4.

Table 4.4 – List of functions available to create and maintain EMD-DF based datasets

Format Chunk
SetFormat(sampleRate, channels, bitsPerSample) <i>Set the format of the waveform data samples.</i>
Data Chunk
WriteWaveformData(dataArrayBuffer) <i>Writes the data array into the data chunk.</i>
ReadWaveformData(samplesToRead, samplesOffSet) <i>Reads the amount of samples in samplesToRead, starting at of samplesOffSet sample.</i>
Label and Note Chunks

`SetLabel(position, jsonString) / SetNote(...)`

Adds a new label (or note) chunk in the Cue and Associated data list chunks.

`DeleteLabel(position) / DeleteNote(...)`

Deletes an existing label (or note) from the underlying Cue and Associated data list chunks.

`GetLabels() / GetNotes()`

Returns a list with all the existing labels (or notes).

Labeled text Chunk

`SetRegion(start_position, end_position, jsonString)`

Adds a new Labeled text chunk (we refer to this as a region) in the Cue and Associated data list chunks.

`DeleteRegion(start_position, end_position)`

Deletes an existing region from the underlying chunks.

`GetRegions()`

Reads and returns a list with all the existing regions.

Metadata – Info Chunk

`SetAuthor(author) / SetTitle(title) / SetCreationDate(creationDate) / SetComment(comment) / SetCopyright(copyright)`

Update the several metadata fields in the Info chunk.

Metadata – Custom Chunks

`SetInitialTimestamp(timestamp)`

Set the initial date and time of the wave file.

`SetCalibrationConstants(constantsArray)`

Set the calibration constants for each of the channels.

`SetExternal(jsonString)`

Set the external chunk content.

`SetAppliance(id, label, extraFields) / SetActivity (id, label, extraFields)`

Add an appliance (or activity) to the corresponding chunk.

`DeleteAppliance(id) / DeleteActivity (id)`

Deletes an appliance (or activity) from the corresponding chunk.

`GetAppliances() / GetActivities ()`

Returns all the appliances (or activities) in the corresponding chunk.

4.1.4 Limitations and Future Work

We now discuss some of the limitations of the current version of EMD-DF and outline future work.

First of all, since the proposed data format always assumes a constant sampling rate, (1 Hz to 4.3 GHz) missing data are not supported by default. Instead, missing data is currently handled either by: i) resampling whenever possible, i.e., when the number of missed samples is short and sparse, ii) break the datasets in different files when missing big blocks of data, and iii) resampling and breaking into multiple files when the missed data is both sparse and dense.

Still, this solution goes against one of the main motivations for the development of this file format, which was to keep the number of required items to a minimum. As such, future work will look at solving this issue by exploring the concept of silent chunks (see Data, Silent, Wave List, and Fact in Table 4.1), which enable the representation of missing data as a count of silence samples.

Second, EMD-DF files are limited to a maximum of 4 GB. This happens because the original WAVE specification, from which EMD-DF was inherited, uses a 32-bit unsigned integer to record the file size header. This is equivalent to about 248 days of two 16-bit channels sampled at 50 Hz; still, we have seen that some datasets have sample rates in the order of the kHz. For example, BLUED was sampled at 12 kHz, meaning that each EMD-DF file can only represent about 16 hours of the three 16-bit channels (i.e., two currents phases and one voltage phase).

Consequently, in future versions, EMD-DF will extend the RF64¹⁷ format that has been created to solve this limitation. This will lead to a new maximum file size of approximately 16 *exabytes*, which is equivalent to roughly 21 years of three 16-bits channels sampled at 4.3 GHz). Furthermore, since RF64 is also an extension of the RIFF / WAVE format, much of the existing code base will be preserved across versions.

¹⁷ RF64 File Format, <http://fileformats.archiveteam.org/wiki/RF64>

Finally, we also plan to make EMD-DF fully compatible with NILMTK, which will in principle require two steps: i) provide a Python implementation of the EMD-DF data structure, and ii) interface this with the NILMTK's `DataStore`¹⁸ class.

4.2 SustData: A Public Dataset for Electric Energy Research

The SustData dataset emerged from the three energy monitoring and eco-feedback deployments reported in Chapter 5. During this period, of almost five years, we have collected and stored a considerable amount of electric energy and eco-feedback related data, which we are now making publicly available to the research community.

4.2.1 Dataset Description

The SustData dataset contains over 50 million individual records of electric energy related data. This includes individual records of energy consumption, power events and user events, to which we added the demographics of all the monitored houses. Additionally, we have created a record of the electric energy generated by the local electric utility, and compiled weather information from a public weather web-service. Next we describe the different data records that compose our dataset and for each record we provide some descriptive statistics of the underlying data.

4.2.1.1 Demographics

The demographics data is a record that describes the participating households, their homes and the periods for which consumption and user event data is available. Table 4.5 lists the demographic features of the participating households.

¹⁸ NILMTK API Documentation: <http://nilmtk.github.io/nilmtk/master/index.html>

Table 4.5 - Household demographic features

Field	Description	Units
home_id	Monitored home unique identifier	-
building_id	Building identifier	-
begin_monitoring	Date and time of the first measurement	datetime
end_monitoring	Date and time of the last measurement	datetime
begin_feedback	Date and time of feedback deployment	datetime
end_feedback	Date and time of feedback removal	datetime
type	Type of residence. Apartment or house	-
rented	Household is rented or not	-
bedrooms	Number of bedrooms	-
adults	Number of adults	-
children	Number of children	-
contracted_power	Contracted power with the provider	kWh

Table 4.6 provides a summary of the household demographic data, where **SH** stands for single house and **A** for apartment.

Table 4.6 – Summary of household demographics

Deploy	Houses	Typology		Rented		Adults	Children
		SH	A	Yes	No		
1	23	6	17	1	22	46	24
2	17 ^a	0	17	2	4	12	9
3	10 ^a	0	10	1	4	8	4
---	50	6	44	4	30	66	36

^a In deployments 2 and 3 a number of houses were part of a control ground (11 and 5 respectively) and no demographic data is available.

In addition to this, the demographics data contains a record with individual householder information. The features of this record are presented in Table 4.7.

Table 4.7 – Household member demographic features

Field	Description
home_id	Monitored home unique identifier
gender	Gender
age	Age at deployment time
role	Role in family, e.g. father, mother, son
education	Education level, e.g. bachelor, doctorate
employment	Date and time of feedback removal

Table 4.8 below gives a summary of the household members demographic data, where **M** stands for male, **F** for female, **E** for employed, **NE** for not employed, **S** for student, **R** for retired and **O** for other employment situations, e.g., too young to attend school / work.

Table 4.8 - Summary of household member demographics

Deploy	Adults		Children		Total		Employment				
	M	F	M	F	M	F	E	NE	S	R	O
1	21	25	13	8	34 ^a	33 ^a	38	2	17	3	10
3	6	6	4	5	10	11	10	0	8	0	3
4	4	4	2	2	6	6	7	1	2	0	2
---	31	35	19	15	50	50	55	3	27	3	15

^a In deployments 1 the gender is missing for 3 participants.

4.2.1.2 Energy Consumption

The energy consumption data is a record of different energy consumption measurements (e.g. real, reactive and apparent power) aggregated at **one-minute time intervals**. Table 4.9 lists the energy consumption measurements that are available in SustData.

Table 4.9 – Energy consumption measurements

Field	Description	Units	D1-a	D1-b	D2	D3
home_id	Monitored home unique identifier	-	✓	✓	✓	✓
timestamp	Date and time of the measurement	datetime	✓	✓	✓	✓
deploy	Deployment identifier	-	✓	✓	✓	✓
Imin	Minimum current	A	✗	✓	✓	✓
Imax	Maximum current	A	✗	✓	✓	✓
Iavg	Average current	A	✓	✓	✓	✓
Vmin	Minimum voltage	V	✗	✓	✓	✓
Vmax	Maximum voltage	V	✗	✓	✓	✓
Vavg	Average voltage	V	✓	✓	✓	✓
Pmin	Minimum real power	W	✗	✓	✓	✓
Pmax	Maximum real power	W	✗	✓	✓	✓
Pavg	Average real power	W	✓	✓	✓	✓
Qmin	Minimum reactive power	VAR	✗	✗	✓	✓
Qmax	Maximum reactive power	VAR	✗	✗	✓	✓
Qavg	Average reactive power	VAR	✗	✗	✓	✓
PFmin	Minimum power factor	-	✗	✗	✓	✓
PFmax	Maximum power factor	-	✗	✗	✓	✓
PFavg	Average power factor	-	✓	✓	✓	✓
miss_flag	If this record is missing	-	✓	✓	✓	✓

Table 4.10 summarizes the energy consumption data, which contains near 22 million individual records. Here, **Min Days** refers to the shortest number of days available for a single house and vice-versa for the **Max Days**. **First Day** is the date of the very first measurement in each deployment. The same applies to **Last Date**.

Table 4.10 – Summary of the energy consumption data

Dep.	Samples	Days	Min Days	Max Days	First Day	Last Day
1-a	3474.557	123	51	119	2010/07/10	2010/11/10
1-b	12481.536	504	240	511	2010/11/25	2012/04/20
3	5671.576	298	237	297	2012/08/01	2013/05/25
4	2884.512	219	187	217	2013/07/31	2014/03/10
---	21627.669	1144	---	1144	---	---

4.2.1.3 Power Events

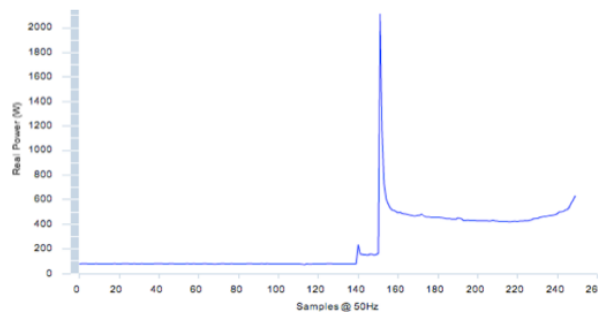
The power event data is a record of all the load changes with an average real power change of at least ± 30 Watts. Table 4.11 lists the measurements that characterize each power event.

Table 4.11 – Power event measurements

Field	Description	Units	D1-a	D1-b	D2	D3
home_id	Monitored home unique identifier	-	✓	✓	✓	✓
timestamp	Date and time of the measurement	datetime	✓	✓	✓	✓
deploy	Deployment identifier	-	✓	✓	✓	✓
delta_P	Real power change	W	✓	✓	✓	✓
delta_Q	Reactive power change	VA	✗	✗	✓	✓
trace_P	Real power trace (50 Hz)	W	✗	✗	✓ ^a	✓
trace_Q	Reactive power trace (50 Hz)	VAR	✗	✗	✓ ^a	✓

^a Only available after a few days

The event trace is the collection of all the power values in the vicinity of the power event edge. In the particular case of SustData the power event traces contain 150 measurements before the edge and 100 after, which at 50 Hz correspond to 3 and 2 seconds respectively. An example of a power event trace from a microwave turning *ON* is shown in Figure 4.11.

Figure 4.11 – Real power transient of a microwave turning *ON*

Currently the dataset contains over 11 million individual power events across all the four deployments. The power event data is summarized in Table 4.12 where **Min Events** is the minimum number of power events in a single house (reciprocate to **Max Events**), **Avg** is the average number of power events between all the houses, and **SD** is the standard deviation.

Table 4.12 - Power events summary

Deploy	Events	Min. Events	Max. Events	Avg.	SD
1-a	1.487.945	17.656	203.336	71	49
1-b	6.057.701	51.564	722.813	288	172
3	1.822.903	18.610	495.596	130	154
4	1.745.232	58.315	485.933	175	123
---	11.113.781	---	---	---	---

4.2.1.4 User Events

This is a record of the frequency at which the householders used the eco-feedback and is only available for the houses that had access to eco-feedback devices (some of the houses were used as control group and no eco-feedback was provided). Also no interactions are available during the baseline periods of the different studies (period where electricity consumption was collected but no eco-feedback was provided). Table 4.13 lists the features that describe each user event.

Table 4.13 – User event features

Field	Description	D1-a	D1-b	D2	D3
home_id	Monitored home unique identifier	✓	✓	✓	✓
timestamp	Date and time of the measurement	✓	✓	✓	✓
deploy	Deployment identifier	✓	✓	✓	✓
type	Type of interaction. Either mouse or touch	✓	✓	✓	✓
view_id	Identifier of the visualized screen	✗	✗	✓ ^a	✓
view_name	Name of visualized screen	✗	✗	✓ ^a	✓

^a Only available after a few days

Table 4.14 summarizes the power event data, where **Interaction** is the total of user events in each deployment and **min.** is the smallest number of user events from a household in the dataset (conversely for the **max**).

Table 4.14 – Summary of user events

Deploy	Houses	Interactions	Min.	Max.	Avg.	SD
1-a	23	3.739	3	548	178	155
1-b	23	7.682	14	928	366	226
2	6	1.424	20	732	237	247
3	5	1.182	136	356	236	72
---	34	14.027	---	---	---	---

4.2.1.5 Environmental and Electricity Production Data

The environmental data was obtained from an online repository of environmental data (wunderground¹⁹), and consists of several measurements (listed in Table 4.15) recorded **every 30 minutes**.

¹⁹ Weather Underground, www.wunderground.com

Table 4.15 - Environmental data measurements

Field	Description	Units
timestamp	Date and time of the measurement	datetime
temperature	Outside temperature	°C
humidity	Relative humidity	%
pressure	Relative pressure	hPa
wind_dir	Wind direction	-
wind_speed	Wind speed	km/h
precipitation	Precipitation levels	mm
events	Relevant events, e.g. rain or thunder	-
conditions	Sky conditions, e.g. partly cloudy	-

Additionally, we also keep a record of the electricity generated in Madeira Island. This is done using a web-service provided by the local electricity company (Electricidade da Madeira²⁰). The overall production data is **recorded at fifteen-minute intervals** along with the generation by individual source. Table 4.16 lists the available measurements.

Table 4.16 - Electric energy production measurements

Field	Description	Units
timestamp	Date and time of the measurement	datetime
total	Total production	MWh
thermal_fuel	Electricity produced by burning fuel	MWh
hydro	Hydro electricity produced	MWh
eolic	Wind farms production	MWh
photovoltaic	Solar electricity produced	MWh
thermal_waste	Electricity produced by burning waste	MWh

²⁰ Empresa de Electricidade da Madeira, www.eem.pt

4.2.2 Download and Explore

SustData is a free and publicly available dataset for all researchers to use and can be accessed from <http://aveiro.m-iti.org/data/>. The full dataset is available in CSV format and via the *OpenDataHub*²¹ [115] our bespoke Dataset Management System (DsMS).

The CSV files are organized in a number of folders named after its contents, i.e., `power_sample`, `power_event`, `power_event_simple` (power events without the transients), `user_event`, `demographic`, `environment` and `production`.

Power samples, power events and user events are further organized in sub-folders, each containing data from the different deployments (`d1_a`, `d1_b`, `d2` and `d3`). The power sample, power event and user event data are organized by household (`ps_<h>.csv`, `pe_<h>.csv`, `pes_<h>.csv`, `ue_<h>.csv`, where `h` is the home identifier).

Environment and demographics data are also organized in a single CSV file per deployment (`household<d>.csv`, `householder<d>.csv`, `environment<d>.csv` and `<production><d>.csv`, where `d` is the deployment identifier).

4.2.3 Future Work

SustData is one of the visible outcomes of the energy eco-feedback deployments that are presented in Chapter 5 of this thesis. SustData is expected to grow as we plan to move beyond traditional eco-feedback systems and anticipate distributed micro-generation scenarios, which will ultimately result in extending this dataset with micro-generation data, weather and other environmental parameters on top of the already existing macro-scale electricity generation and environmental data.

²¹ OpenDataHub DsMS, <http://aveiro.m-iti.org/data/sustdata/opendatahub.html>

4.3 SustDataED: A Public Dataset for Electric Energy Disaggregation Research

The SustDataED dataset is an extension to the original SustData dataset. This dataset consists of aggregated and individual appliance electric energy consumption measurements.

Here we describe the data collection setup and briefly explore the contents of the current version of the dataset. We then provide some details about the underlying structure of the collected data and outline some future work.

4.3.1 Data Collection Setup

In this section we describe the setup that is used to collect the aggregate (i.e., whole-house data) and individual appliance consumption data that compose SustDataED.

4.3.1.1 Aggregate Consumption

The hardware setup used to measure and collect aggregate consumption data is an extension of the multi-house energy monitoring and eco-feedback platform, which is detailed in section 5.2.2 of Chapter 5.

The setup consists of a multi-channel data acquisition board (LabJack U6²²), one processing unit, and a combination of current transformers (CT) and voltage transformers (VT), according to number and type of channels to sample and the desired sampling rate (see Figure 5.2). For example, the LabJack U6 enables the simultaneous monitoring of 14 analog channels with a 16-bits resolution and a maximum sampling rate of 3.2 kHz per channel.

In our setup, the Current and Voltage waveforms are sampled and stored in the EMD-DF data format in one-hour files. Furthermore, in order to minimize the effects of synchronization issues that may occur due to the differences in the internal clocks of the data acquisition devices²³, we have decided to perform a hardware-timed data acquisition (i.e., the

²² LabJack U6 DAQ, www.labjack.com/U6

²³ LabJack Forums, <http://forums.labjack.com/index.php?showtopic=6171&p=20718>

DAQ hardware is responsible for acquiring the requested number of samples) and a software-timed storage of the data (i.e., a new file is created only when exactly one hour has gone since the first samples were acquired by the DAQ, independently of the fact that the expected number of samples for that period was reached or not).

The downside of this approach is the possibility of ending up with files that have a number of samples that is different from what was originally expected (i.e., less samples when the clock is ahead and more samples when the clock lags behind). Therefore, the resulting files are resampled to guarantee that all the one-hour files have the same number of samples. In our particular case, we have resorted to Matlab's default `resample` function²⁴.

Lastly, since the storage format only enables values between -1 and 1, the Current and Voltage measurements have to be scaled before being stored. To this end, the measurements are scaled according to the maximum expected value and a confidence interval that varies with the sensor characteristics. An example of this is provided in subsection 4.3.2.

4.3.1.2 Individual Appliance Consumption

For the appliance-level data collection we have used the Plugwise system (this was also used in [92], [100]), which is a commercially available, distributed sub-metering platform. The system is composed of three main components namely the Circle (also called Module), the Stick and the Source software. Each circle is connected between the appliance being measured and the outlet. The "Stick" is used to wirelessly (using the ZigBee wireless protocol) interface each deployed circle with a computer that processes and displays the consumption of each individual circle using the source software.

The Plugwise source system aggregates the appliance level measurements (e.g., by hour, day or month) and generates multiple consumption reports for the users. This is, however, limited when we consider the level of granularity required to label a NILM dataset, e.g., knowing if an appliance is *ON* or *OFF* and when these transitions occurred. Consequently we

²⁴ Matlab `resample`, <http://www.mathworks.com/help/signal/ref/resample.html>

extended an open-source Python package²⁵ that allows direct access to the raw measurements of the Plugwise modules.

The original version of the software maintains a *hard-coded* list with the mac-address of the circles to be monitored and sequentially scans each module every 10 seconds. If for a given module no response is return within six attempts (each attempt is assumed to take one second) an exception is thrown and the system moves to the next plug in the list. This solution however presents some caveats for the collection of datasets, for instance: i) in the best possible scenario (when all the modules are accessible), individual appliance measurements are only available once every 10 seconds, and ii) the sampling interval increases by six seconds for each module that happens to be offline.

Against this background, we modified the original scripts to make the data collection process more suitable for energy disaggregation datasets. The following changes were made:

- The 10-second interval between scans was set to zero. In other words, a new scanning round-trip begins right after the previous one is concluded. Our tests have shown that it takes about one second to scan 10 plugs (1 Hz) and two seconds for 19 plugs (0.5 Hz). This assumes that all the plugs are online.
- While it was not possible to remove the six seconds timeout or make parallel scans, we have added the possibility of changing the list of modules to be scanned in runtime. This is particularly important, since it is very plausible that some appliances will be offline most of the time. For example, a vacuum cleaner is normally not connected to the grid unless it is in use.
- We developed a web application to facilitate the configuration during the deployment and maintenance phases. This includes enabling / disabling modules and checking the real time and historic consumption.

The measurements obtained from each module (apparent power and timestamp) are continuously stored in comma-separated-value (CSV). A new file for each appliance is created everyday at 12 AM.

²⁵ Plugwise Python, <https://github.com/SevenW/Plugwise-2-py>

4.3.2 Dataset Description

The version of SustDataED that we are now describing consists of 10 days of electric energy consumption and room occupancy measurements taken from a single-family residence in Portugal, composed of four householders. The monitored house is an apartment from the early 2000s and is comprised of seven divisions. The collected measurements include aggregated (whole-house) consumption and the individual consumption of 17 appliances.

4.3.2.1 Aggregated Data

The current and voltage waveforms were sampled from the mains at 12.8 kHz. The resulting waveforms are stored in one-hour files. Each one-hour file contains exactly 46.080.000 (forty six millions and eighty thousand) samples per channel and takes 184.3 MB of disk space.

The current waveforms were sensed using a 30 A to 1 V current transformer with a peak instantaneous voltage of about 1.47 V. Therefore, to avoid clipping when storing in the EMD-DF format, the data was scaled by a factor of 1.5 V ($1.47 \text{ V} \pm 5 \%$). The voltage waveforms were sensed using a 230 V to 0.5 V voltage transformer with a peak voltage of 0.7 V. As such, no scaling was necessary. The following formulas can be used to calculate the real current and voltage from the EMD-DF files:

$$\text{volts} = \text{value_from_file} \times 460 \quad (4.2)$$

$$\text{amps} = \text{value_from_file} \times 30 \times 1.5 \quad (4.3)$$

The resulting current and voltage files were later post-processed to generate additional power measurements that are relevant for energy disaggregation research. More concretely, we created 50 Hz EMD-DF files containing the following measurements: i) active power, ii) reactive power, iii) fundamental voltage RMS, iv) fundamental current RMS, v) total voltage RMS, and vi) total current RMS. The measurements were calculated using the power equations presented in section E.1 of Appendix E.

The resulting one-hour files were then concatenated to form 24-hour files. Each 24-hour file contains exactly 4.320.000 (four million three hundred and twenty thousand) samples per channel and takes 51.8 MB of disk space. The formulas in (4.2) and (4.3) can be used to obtain




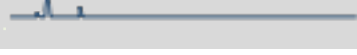






the real current and voltage values. As for the active and reactive power, the real values can be obtained using the formula defined in equation (4.4).

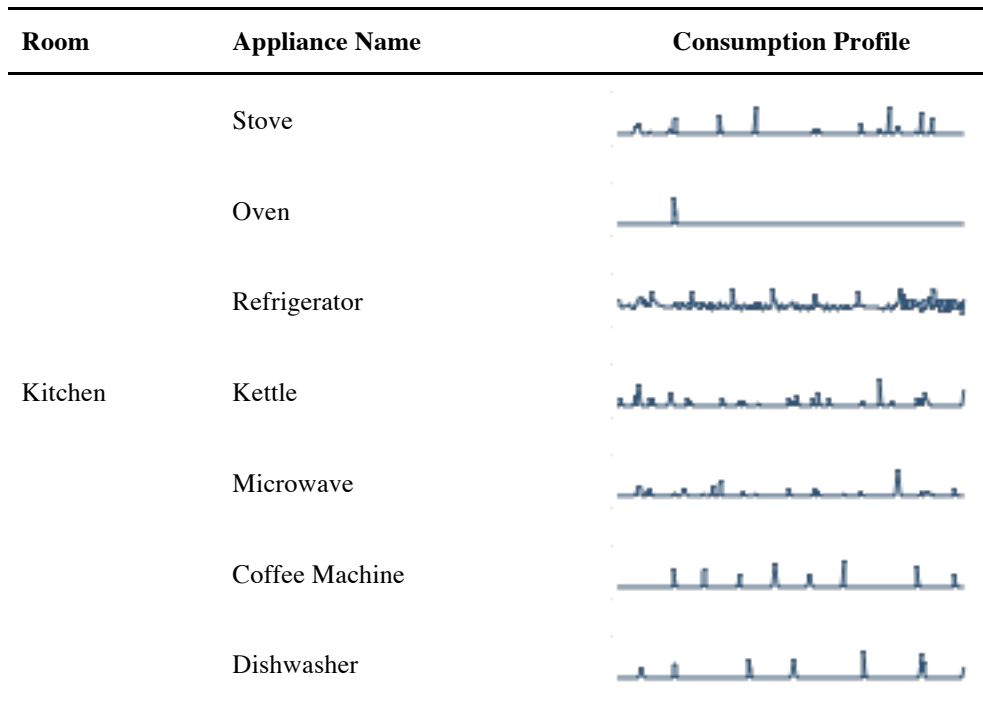
$$power = value_from_file \times 30 \times 1.5 \times 460 \quad (4.4)$$

4.3.2.2 Individual Appliance Consumption

The dataset contains the individual consumption of 17 appliances, measured roughly every two seconds (0.5 Hz). Table 4.17 summarizes the individual appliance consumption data. The appliances are listed according to the divisions in the house, and for each appliance we show the respective consumption profile (aggregated by date and hour).

Table 4.17 – Summary of the individual appliance and occupancy data that can be found in SustDataED

Room	Appliance Name	Consumption Profile
Bedroom 1	TV 1	
Bedroom 2	TV 2	
	Laptop 1	
Bedroom 3	TV 3	
Laundry Room	Washing Machine	
	Freezer	
	Water Heater	
Living room	TV 4	
	PlayStation	
	Laptop 2	



In Figure 4.12 we show a plot of the aggregated and the sum of the individual appliances consumption grouped by date and hour. We also plot the percentage of energy explained – %EE – (i.e., the % of aggregate energy for which there is appliance-level data), showing that in average the individual appliances are able to explain 82% of the aggregate consumption.

Note that there are occasions when the %EE is above 100%. This happens because in some occasions the whole-house data acquisition hardware stopped working. As such, when summarizing the data (e.g., by hour) it is possible that the sum of the appliances is higher than the aggregated consumption, which will be reflected in a %EE above 100%.

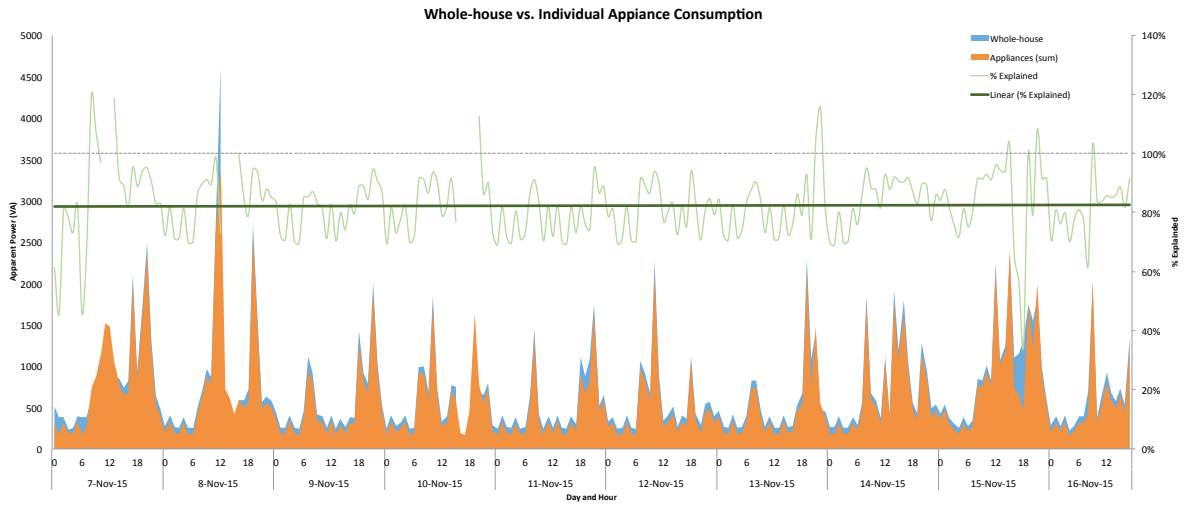


Figure 4.12 – Aggregate consumption vs. the sum of the individual appliances summarized by day and hour

4.3.3 Download and Explore

SustDataED is a free and publicly available dataset for all researchers to use and can be accessed from <http://aveiro.m-iti.org/data/>.

The aggregate data is stored in two different directories, named SustDataED_VI and SustDataED_PQVIVI. The former contains the raw voltage and current waveforms at 12.8 kHz whereas the later contains the processed waveforms at 50 Hz. Each directory contains a number of subdirectories named after the initial timestamp of the first file in it. Finally, the files in each directory are also named after the timestamp of their first sample (`<initial timestamp>.wav`).

All the data files are stored in the EMD-DF format, each of which containing a number of chunks with configuration information as per Figure 4.13.

RIFF DEFAULT - Channels: 2 - Rate: 12800.0 - BitsPerSample: 16 EMD-DF SPECIFIC - InitialTimestamp: 2015-11-07 02:08:44.599 - Timezone: WET - Rate: 12800.0 - Calibration: [460.0, 45.0]	RIFF DEFAULT - Channels: 6 - Rate: 50.0 - BitsPerSample: 16 EMD-DF SPECIFIC - InitialTimestamp: 2015-11-06 20:08:44.599 - Timezone: WET - Rate: 50.0 - Calibration: [20700.0, 20700.0, 460.0, 45.0, 460.0, 45.0]
--	---

Figure 4.13 – EMD-DF configuration information: raw voltage and current (left), processed waveforms (right)

The individual appliance data is provided in an SQLite database with three tables: i) room, ii) device, and iii) measurement.

- Room: (id: Integer, name: Text)
- Device: (id: Integer, name: Text, room_id: Integer)
- Measurement: (id: Integer, device_id: Integer, capture_time: Text, power: Real), where the capture_time is a string in datetime format (i.e., YYYY-MM-DD HH:MM:SS)

4.3.4 Future Work

Regarding the future of SustDataED, we are currently collecting data from a second household. Furthermore, we are also attempting to integrate new sensors for the different appliances (e.g., high-frequency measurements for fast switching appliances like washers and driers [116], [117]). Lastly, we are also looking at ways to introduce human activity detection, either by means of additional sensing technology like proximity beacons [118] or human-computer interaction techniques like the daily reconstruction methods through diary studies [119].

As a final remark, we should note that this dataset was released near the completion of the writing of this PhD thesis, and that the labeling is not yet completed. As such, it was not possible to use it along with the other event detection datasets in Chapter 6.

Chapter 5 NILM Deployments in Real World Scenarios

This chapter discusses the deployment of the NILM technology “in the wild” mostly for the purpose of supporting eco-feedback research. More concretely, we report on the many engineering challenges behind building and deploying NILM and eco-feedback technology in real word scenarios, which despite being relevant to the research community are seldom reported in the literature.

To do this, we rely on more than 5 years of experience developing and improving a research platform that combines low-cost non-intrusive monitoring of energy in households to enable the quick deployment of long and short-term studies of eco-feedback technology and at the same time serve as a research platform for developing and evaluating NILM algorithms.

The remaining of this chapter is organized as follows. First we present the rationale behind our NILM and eco-feedback deployments in the wild. Then, in section 5.2, we thoroughly describe the two research platforms that were developed in the process, and provide details of the three live deployments that were performed using such platforms. In section 5.3 we provide a comprehensive discussion of the many practical considerations of deploying and maintaining such systems. Finally, in section 5.4, we conclude this chapter by summarizing the answer to the first research question of this thesis, and highlighting the implications of our findings for future research in this or similar fields, where live deployments of monitoring and feedback technology are required. Lastly, we describe the limitations of this work and outline possible improvements.

5.1 Introduction

As discussed in previous chapters, although most of the existing studies in eco-feedback technology show very promising results in promoting energy savings, there is still very little evidence that these savings will sustain over time. Furthermore, and despite current literature is abundant in research that studies the effects of deploying eco-feedback in real world scenarios, the practical issues of deploying such systems, e.g., optimizing prototype costs and ensuring easy access to the collected data, are very seldom reported in literature.

We argue that reporting on the practical issues of real world deployments can be of crucial importance for assessing research findings in larger and longer deployments. As such, in this chapter we focus our attention in exploring the practical implications of deploying NILM and eco-feedback technology outside of the controlled laboratory environments.

More particularly, we report on the technical (e.g., hardware / software requirements, installation), social (e.g., devices security and intrusiveness) and financial (e.g., prototype costs) challenges of designing, deploying and maintaining hardware and software platforms to support long-term real word studies on energy monitoring and eco-feedback technology.

5.1.1 Vision

The research platform described here is part of the Sustainable Interaction with social Networks, context Awareness and Innovative Services (SINAIS) research project, which involved a team of multidisciplinary researchers looking at using sensing, social networking and context awareness to understand and motivate people to reduce their energy consumption in the residential and transportation sectors.

Under the umbrella of this project, we were required to develop a hardware and software platform to simplify and reduce the costs of deploying and maintaining energy monitoring and eco-feedback solutions in the wild during long periods of time. The envisioned energy monitoring and eco-feedback research platform is illustrated in Figure 5.1.

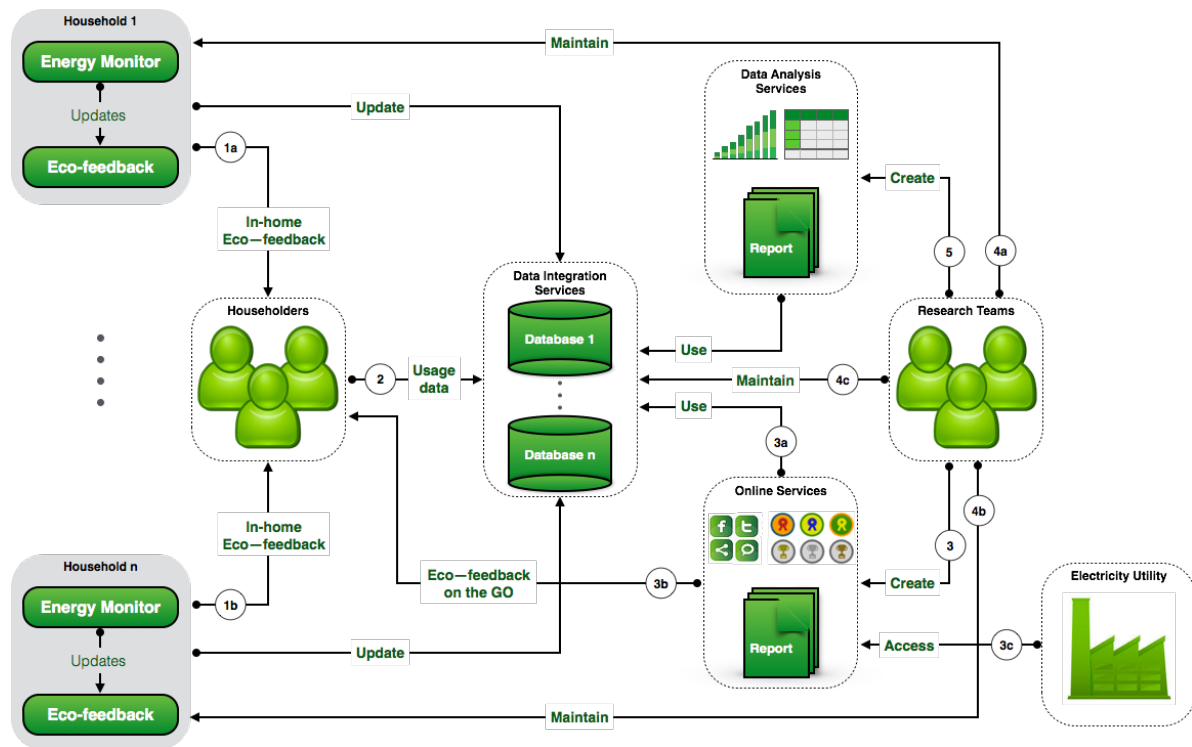


Figure 5.1 - Energy monitoring and eco-feedback research platform overview

Starting from the left side of the diagram, each house would be equipped with a non-intrusive energy monitor and different devices capable of providing eco-feedback to the householders. All the acquired data would be stored locally and in a centralized database to be used in a set of different scenarios.

One of such scenarios is the possibility of deploying different eco-feedback solutions and keeping track of how the householders actually interact with such devices. This scenario is depicted in the diagram (*arrows 1a, 1b and 2*) in which householders are given feedback through the “in-home” eco-feedback devices and their usage data (e.g., visualized screens) is stored in the central database.

Another scenario (*arrows 3, 3a, 3b and 3c*) is the one in which the collected data is leveraged to create online services such that one can study other domains of eco-feedback technology like for instance information sharing in social networks or the application of *gamification* (*arrow 3b*). Likewise, this would also enable the creation of extended

consumption reports that can be used by electric utility companies to understand their consumers and leverage on that to offer new services (*arrow 3c*).

Lastly, in the right side of the diagram there is the research team responsible for maintaining the platform and analyzing the data items that will be generated throughout time (*arrows 4a, 4b, 4c, 5 and 5a*). To this end, the envisioned framework should also incorporate data analysis servers with dedicate software as well as remote access and file sharing facilities such that the deployed applications can be easily and transparently updated by the research team members.

5.1.2 Requirements

Considering the envisioned scenarios just mentioned, from a hardware standpoint an energy monitoring solution for this platform would require the following elements [120]:

- A **data acquisition** module capable of reading current and voltage signals,
- A **processing element** capable of calculating the energy and also events from the acquired signals,
- A **visualization component** to enable some form of visual and audio feedback,
- **Additional sensors** capable of detecting human activities (motion, etc.),
- A **connection** element to upload data to online repositories.

Given this set of requirements we performed an extensive survey of the available commercial metering solutions (see section A.1 of Appendix A), which revealed that none of the existing solutions was capable of offering all the required elements at reasonable costs.

For example, the most affordable solutions only measure current and assume a fixed reference voltage, while others only report power measurements at granularities that are not suitable for providing real time eco-feedback or creating robust NILM implementations (e.g., most event-based approaches require power measurements at granularities equal or greater than 1 Hz). As for the most expensive solutions some offered all the required data, from instantaneous current and voltage to power factor and harmonics content (e.g., circuit-level meters). Yet, they were too expensive and difficult to install for our purposes.

Moreover, most systems rely on proprietary protocols or specific applications to communicate the measurements, thus making the integration with third party systems (e.g., eco-feedback devices and social network applications) extremely complex. Lastly, none of the surveyed solutions seemed to offer a built-in method for inferring human activity, which was an important requirement for our studies.

5.2 Research Platform Overview

The failure to find a viable commercial solution led to the creation of the two custom end-to-end unobtrusive energy monitoring and eco-feedback platforms that are described in this section.

As of the time of this writing, the two platforms have been successfully deployed in three long-term energy monitoring and eco-feedback studies that lasted between six and 18 consecutive months.

The three deployments involved a total of 50 different households that had the system installed and running continuously in their houses to acquire data that was then used by an interdisciplinary team of researchers to monitor and understand how people react and adopt eco-feedback technologies, which so far resulted in two additional PhD theses [121], [122] and three master thesis [115], [123], [124].

In the first version of the platform (used in the first deployment) the energy meters are installed in the main power feed of the house and the eco-feedback is provided using a built-in display. In the second version of the platform (used in deployments two and three) the energy monitors are installed in the main lobby of the apartment buildings, hence enabling us to measure the energy consumption of multiple homes from a single sensing location, and the eco-feedback is provided using bespoke mobile applications.

We refer to the former as *single-house* in a sense that every house needs to have its own energy monitor, whereas the later is referred to as *multi-house* since multiple houses can be monitored from a single energy monitor.

5.2.1 Single-House Energy Monitoring and Eco-Feedback Platform

Taking together the hardware requirements that were previously mentioned, it becomes clear that a flexible research platform could easily reach hundreds of Euros (each element costs between 50 to 100 Euros without the cost of integration).

As such, after several attempts with custom hardware we decided to use a netbook that provided all of the above-mentioned elements in a compact package that could cost between 200 and 300 Euros. The soundcard serves as the data acquisition module (two channels, one for current and another for voltage) using the built-in Analog to Digital Converter (ADC). The mini display and the speakers provide the feedback, while the Wi-Fi and Ethernet cards enable communication over the Internet. Lastly, the built-in camera and microphone can act as low- cost sensors for human activity sensing.

5.2.1.1 Data Acquisition and Load Monitoring

In European countries, most residential buildings have 50 Hz single-phase alternate current (AC) systems with a 230 V voltage. Consequently, only two sensors are required to measure consumption in most households, i.e., one sensor for current and another for voltage.

In our platform the current waveforms are sensed using standard non-invasive split-core (clam-on) AC current sensors, similar to the one shown on the left side of Figure 5.2. The voltage is measured with a custom-made voltage transformer that steps down the 230 V input voltage to 0.5 V, such that it can be correctly sampled by the soundcard. The two sensors are then connected to the soundcard using 3.5 mm TRS splitters (Figure 5.2 – right).



Figure 5.2 – Sensing hardware: split-core current sensor (left), voltage transformer (center) and TRS splitter connectors (right).

The netbook and the sensors are installed in the main power feed (see Figure 5.3), thus covering the entire house consumption and eliminating the need for additional sensing

locations. The current and voltage waveforms are continuously sampled at a pre-defined sampling rate (8 kHz in our deployments) using the netbook built-in soundcard. The digitized waveforms are then processed and transformed into common power metrics that are representative of the energy consumption (e.g. apparent, real and reactive power).

The power metrics are calculated at a rate of 50 samples per second (i.e. the mains frequency), and subsequently used for event detection, event classification and, ultimately, the breakdown of consumption into individual appliances. In the meantime, all the metrics are stored in a local database (aggregated at 1 measurement per minute) along with the detected power events for feedback and future data analysis purposes.



Figure 5.3 – Current and voltage sensors installed in the main power feed

5.2.1.2 Energy Eco-Feedback

The energy eco-feedback is provided on-site using the built-in display of the netbook (see Figure 5.4) through different custom made applications that provide historical and real-time information on energy consumption and power events.

The historic consumption data is obtained by directly querying the local database, whereas the real-time information is obtained by connecting to one of the two multi-threaded Internet socket servers that run in parallel with the energy monitoring software. Additionally, the eco-feedback software is able to record how often householders interact with the different visualizations by keeping a log of every mouse click and screen change on the user interface.



Figure 5.4 – Energy eco-feedback is provided on-site using the netbooks’ built-in LCD screen

During the deployment of this platform, the householders were given access to two eco-feedback user interfaces. The first interface consisted mostly of traditional column charts to display the consumption information. The system displays a column chart with the total energy consumption over the current day, and also the consumption of all the previous days. It is also possible to compare the consumption of the current week against last week based on a daily average. In Figure 5.5 (left) we present an example of the daily consumption in a column chart, where each column represents the different hours of the day.

The second version was designed based on feedback we received from the deployment of the first version. In this interface we used a gauge analogy to display consumption information to the user. The interface displays information for the hour, week, month and year’s consumption and is organized in a tabbed menu. The consumption levels are mapped using a color scale going from green to dark red, and if the mouse cursor hovers over the gauge it displays information about CO₂ emissions and cost associated with that time slot.

Both versions are able to display the current consumption information, but only the second one is able to display power events. These are displayed in the hour view because displaying the events in the day or month view would result in a very confusing interface (the hour view is refreshed every hour, meaning that only events for the current hour are displayed). Every time a power event is detected a small dot is added to the interface as close as possible to the time of occurrence. The size of the dot is used to indicate the amount of power change, and a click on it reveals the appliance that has the highest probability of having triggered that event. Additionally the user can confirm or correct the system’s estimate. In Figure 5.5 (right) we present a screenshot of the hourly consumption screen with dots that represent power events.

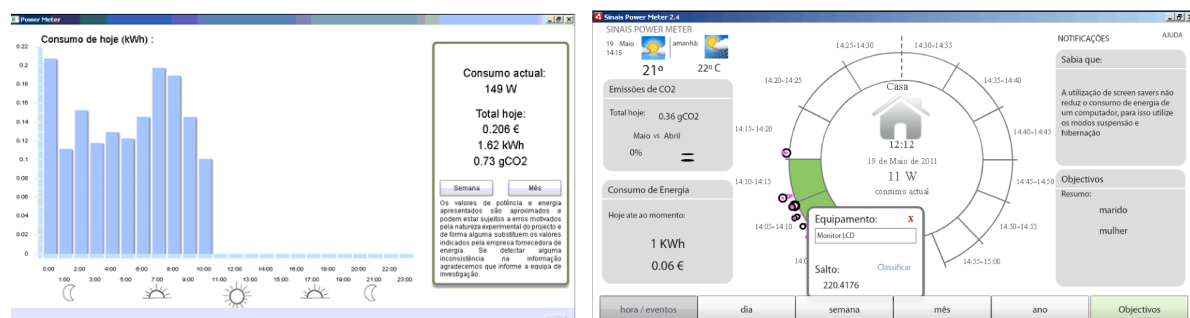


Figure 5.5 – Eco-feedback interfaces used in deployment one: version 1 (left), version 2 (right)

5.2.1.3 System Installation and Data Integration

In order to install this version of the platform the current sensor must be placed in the main fuse box while the voltage transformer and netbook need to be connected to a power supply.

Regarding the data integration, each individual monitor stores all the data locally using an SQLite ³²⁶ database. All the databases are synchronized using a Dropbox²⁷ folder, which is linked to an account shared among all the houses in the deployment. Finally, the individual databases are integrated into a single data warehouse using the SQL Server²⁸ Integration Services running in a machine that is also linked to the same Dropbox account. A general overview of the single-house platform installation and data integration process is provided in Figure 5.6.

²⁶ SQLite , www.sqlite.org

²⁷ Dropbox, www.dropbox.com

²⁸ SQLServer, www.microsoft.com/en-us/server-cloud/products/sql-server/default.aspx

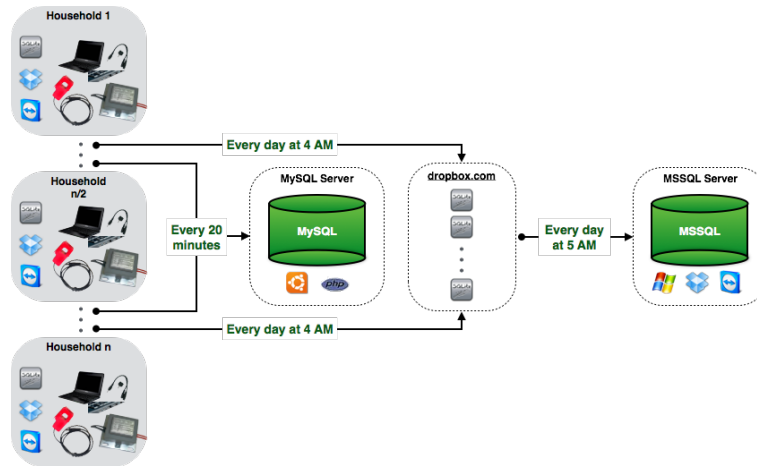


Figure 5.6 – General overview of the single-house version of the energy monitoring platform

5.2.2 Multi-House Energy Monitoring and Eco-Feedback Platform

Our hardware and software platform evolved according to the limitations found on the first deployment. As mentioned previously, our initial setup (sensing and eco-feedback) was installed in the main breaker box, raising some issues of limited accessibility for some household members (especially children) and some questions regarding the security and intrusiveness of the system. Furthermore, in some households the main fuse box may be located in the basement or in the attic, thereby making this solution unfeasible for providing eco-feedback unless this is done using alternative means such as web or mobile applications.

As a consequence of these facts, we made some changes to the original monitoring platform. The most significant one involved the replacement of the netbook soundcard with a more capable Data Acquisition System (DAQ).

The new DAQ system is installed in the main lobby of the apartment buildings where all the meters from the electric company are mounted. This solution enables us to measure the energy consumption of multiple homes from a single sensing location, hence reaching a new level of unobtrusiveness and security since no hardware needs to be installed inside the monitored houses. Furthermore, using a multi-channel DAQ we are able to deploy our platform in two or three-phase electric power systems (the previous version is limited to

single phase systems due to limitations in the number of channels in the soundcard – only two channels, right and left, are available).

5.2.2.1 Data Acquisition and Load Monitoring

Current and voltage signals for all the monitored houses are acquired from the building main electric panel (Figure 5.7 – left) and processed by a single computer using a dedicated DAQ board (Figure 5.7 – right).



Figure 5.7 – Multi-house platform installation: current sensors (left), voltage sensors and DAQ (right)

The computer, to which we refer to as Energy Monitoring Base Station (EMBS), is also responsible for storing and providing remote access to consumption data. To this end, all the data is stored in a single MySQL database, and a layer of REST²⁹ web-services was implemented to enable easy access to the data.

Regarding the data acquisition hardware, in this particular case we use the LabJack U6 DAQ, which is able to scan 14 analog input signals with a bit resolution up-to 16 bit and a maximum sampling rate of 50 kHz (to be shared among all the active input channels). The LabJack DAQ connects to the EMBS via USB 2.0.

5.2.2.2 Energy Eco-Feedback

The multi-house energy monitoring and eco-feedback platforms enable householders to access the eco-feedback in different places of the house, or even outside the household premises as long as there is an Internet connection available. As such, in the particular case of

²⁹ Representational State Transfer (REST), www.ics.uci.edu/~fielding/pubs/dissertation/top.htm

the two deployments of this platform, the eco-feedback was provided using custom made mobile applications running on 7'' Android tablets, as shown in Figure 5.8.

The developed applications receive real-time and historical consumption data from the sensing platform. The-real time data is provided through the TCP socket servers that were preserved across versions, whereas the historical data is loaded using the REST web-services provided by the EMBS. The historical data is stored locally in the tablet such that the users can still check past consumption without an Internet connection. Furthermore, this strategy reduces processing on the server side and the payload of the HTTP communications between the client applications and the EMBS webserver since only new data needs to be processed and transferred.

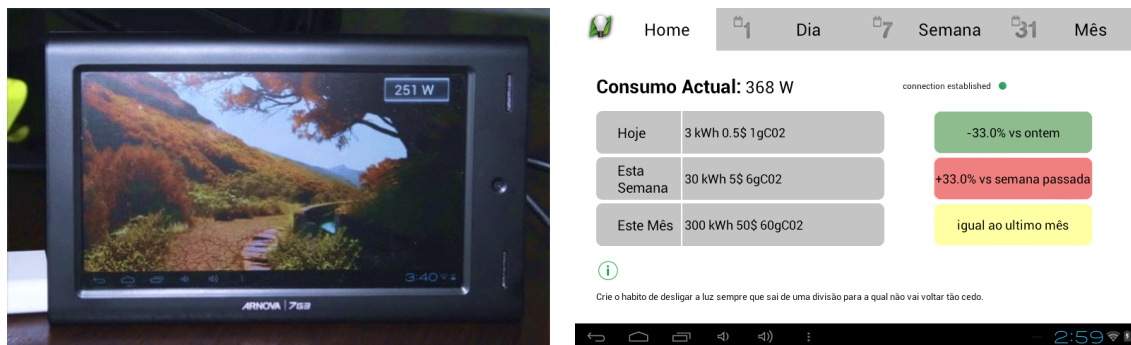


Figure 5.8 – Energy eco-feedback applications used in deployment two: energy awareness mode (left), detailed consumption mode (right)

The eco-feedback system used in the second deployment involves two main modes of operation. When it is not used for two minutes it goes into the *Energy Awareness* mode that shows the consumption mapped as a digital illustration of the local endemic forest (see Figure 5.8 – left). Once the user interacts with the tablet, by pressing the back soft key, the system goes to *Detailed Consumption* mode and shows daily, weekly and monthly information about the home energy use as shown in Figure 5.8 – right.

In the eco-feedback system used in the third deployment the energy awareness mode was replaced with information about energy generation at the island-level. The developed application is composed of a set of tabs that represent the production information and summary of the consumption on a daily, weekly and monthly basis.

The production view is the default mode of the application and the system reverts to this visualization when no interaction happens for a pre-defined period of time. The electricity production is represented using a “cumulative” chart of all the sources of energy used during the day, their quotas relative to each other, and a prediction of which sources would be available for the rest of the day (Figure 5.9 – left). The summary view (Figure 5.9 – right) contains two charts representing the consumption of the current day, week and month, and a comparison between homologous periods.



Figure 5.9 – Energy eco-feedback applications used in deployment three: energy generation information (left), consumption summary (right)

For additional information about the different eco-feedback studies please refer to the following publications: i) [11], [109] and [125] for the first deployment; ii) [126]–[128], for the second deployment; and iii) [129], for the third deployment.

5.2.2.3 System Installation and Data Integration

The physical installation of the multi-house platforms can be done in a number of different ways, depending on the number of houses that need to be monitored and the desired sampling frequency. For example, a building with 11 apartments or less can be fully monitored using a single DAQ as depicted in Figure 5.10.

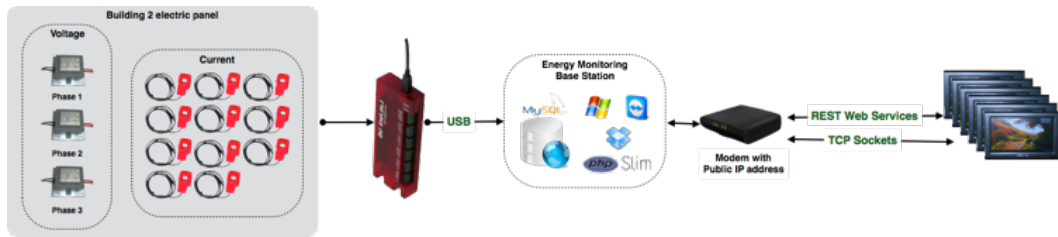


Figure 5.10 – Example of a possible configuration of the multi-house energy monitoring platform

All the three voltage phases and the each of the 11 current signals are sensed and digitized using the multi-port DAQ. The digitized waveforms are then fed to the energy monitoring software that computes the different power metrics by combining the corresponding voltage and current waveforms of each house.

Regarding the data integration task, the MySQL database in each the EMBS are available for remote access. This way it is possible to remotely access and integrate the individual MySQL databases whenever new data is required. Moreover, to further optimize the interactions with the databases, we maintain summary tables (e.g., hourly, daily, weekly and monthly energy consumption averages).

5.2.3 Summary of Deployments

As it was already mentioned, the two research platforms have been successfully deployed in three long-term energy monitoring and eco-feedback studies that took place in the city of Funchal, the capital of Madeira Island in Portugal. Deployment one was done using the single-house monitoring platform, whereas the second and third deployments were done using the multi-house version.

Altogether, the three deployments involved 50 different households. In Figure 5.11 we present the timeline of the three deployments where the start and end dates of each deployment are relative to the date of the first and last obtained measurements, respectively.

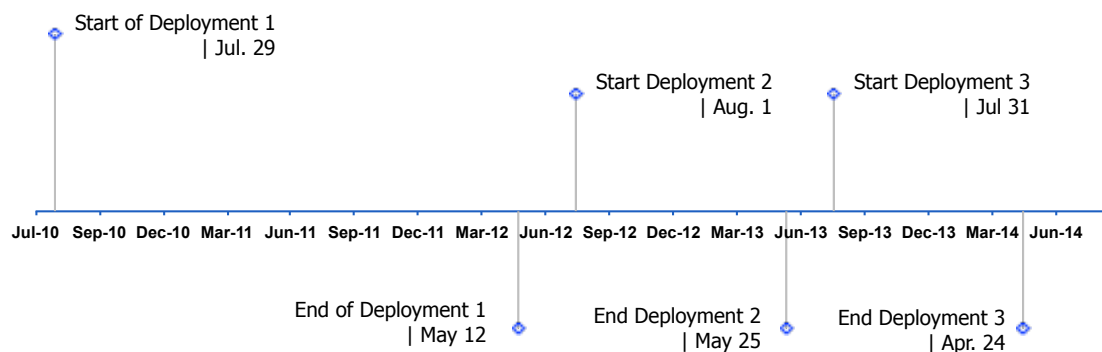


Figure 5.11 – Research platforms deployment timeline

In the first deployment the sensors and the netbook had to be installed directly in the main fuse box of the house. To this end, and since this involved specialized work, all the installations were performed by certified electricians from the local electricity utility.

Furthermore, during the installations the teams made sure that all the sensor and netbook cables were hidden from sight, with only the output ends of the current and voltage sensors passing to the front. Also, since both sensors had very short cables we decided to attach our meter to the fuse box door with Velcro as shown in Figure 5.12.



Figure 5.12 – The energy monitors are attached to the main fuse door with sticky back Velcro straps

Altogether, and considering each day with at least one installation, it took 16 days to complete the installation of the 23 energy monitors that comprise the deployment. Ultimately, the deployment lasted for 658 consecutive days between the end of July 2010 when the first device was installed and mid-May 2012 when the last one was removed. In Figure 5.13 we present the major milestones of the entire deployment including an overview of how the number of participating households evolved over time.

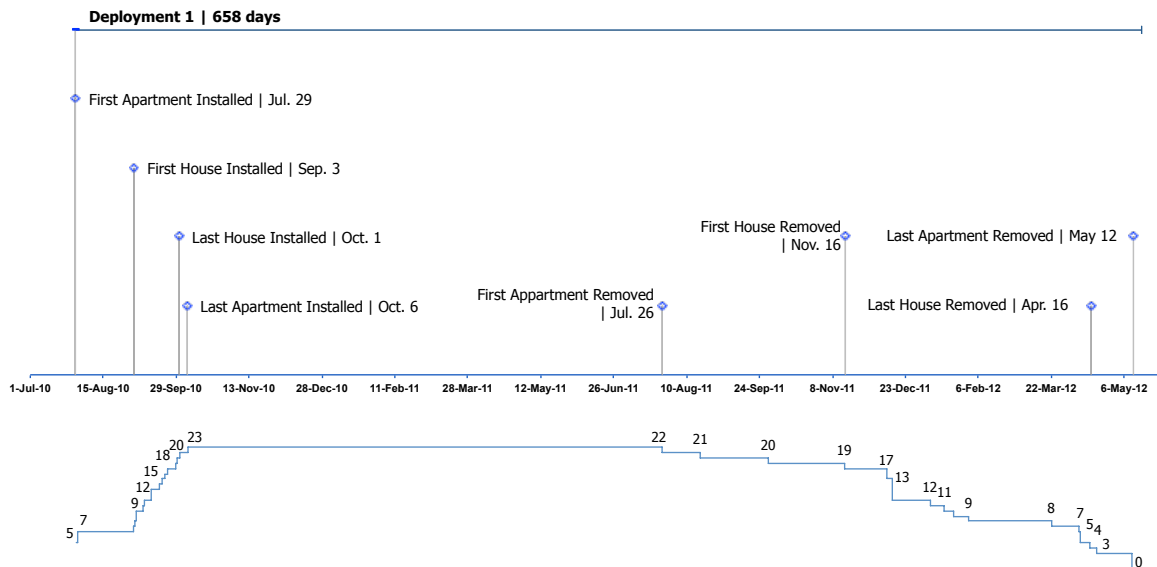


Figure 5.13 – Major milestones of deployment one (top); active installations over time (bottom)

Regarding the second and third deployments, the sample was recruited from a number of apartment blocks, since the second version of the energy monitoring and eco-feedback platform is installed in the building main electric panel (see Figure 5.14). Furthermore, due to the considerable complexity of the buildings main electricity panels all the installations had to be performed by qualified electricians.



Figure 5.14 – Multi-port DAQs installed in the main electric panel of one of the buildings

The second deployment started in the beginning of August 2012 and lasted until the end of May 2013 for a total of 298 consecutive days. As for the third deployment, it started in the beginning of August 2013 lasting until then end of April 2014 when the platform was totally removed from the building. Overall, this deployment lasted 268 consecutive days. In Figure 5.15 we present the major milestones of the two deployments including an overview of how the number of monitored apartments evolved over time.



Figure 5.15 – Major milestones of deployments two and three (top); active installations over time (bottom)

5.3 Practical Deployment Considerations

Having provided a concise description of the development and deployment of two energy monitoring and eco-feedback platforms in real world settings we will now attempt to answer research question number one of this thesis: *“what are the practical issues of deploying NILM and eco-feedback systems in real-world scenarios?”*

More concretely, we will reflect on more than five years of experience developing and deploying unobtrusive energy monitoring solutions in real-world scenarios to identify and clarify what we consider to be the practical issues behind deploying and maintaining research platforms like the ones presented in this chapter.

To do this, we will first explore the different *technical challenges* that researchers are presented like physical installation constraints and data management issues. Secondly, we discuss the different *social challenges*, which involve for example maintaining a steady sample during the entire deployment. Finally, we discuss the costs associated with deploying energy monitoring systems for NILM and eco-feedback research purposes.

5.3.1 Technical Considerations

Technological (or physical) issues refer to the different challenges that research teams are presented when developing and deploying this kind of systems. In our particular case, we have identified three main categories of technological issues, namely: i) installation and maintenance; ii) communication; and iii) data management.

5.3.1.1 Installation and maintenance

Regarding the *installation* of the system, the main challenges are related to the location of the breaker boxes, particularly when deploying the first version of the system. For example, despite all of the homes in the first deployment had the fuse box next to main door or in the kitchen there were cases in which the fuse box was located inside a bedroom (see Figure 5.3 - right), thus making the whole process unviable due to the extreme intrusiveness.

Likewise, it is also expected that in some older houses the breaker box will be located in the basement or in the attic, which in any of the cases invalidates solutions with built-in eco-feedback. Furthermore, we should remark that most energy monitoring systems requires a constant power source to connect the metering device. However, as we observed in our experiments, it was not very common to find power outlets near the break box. Consequently, when deploying single-house monitoring systems it is important to take into consideration that some extra work might be required to install all the necessary equipment.

With respect to the multi-house platform, the issues of accessing the breaker box are naturally avoided. However, installing the system in the building breaker box is by far a more challenging task that must be conducted by experienced electricians. Nevertheless, the biggest challenge of deploying the second platform is also related to the actual physical location of the electrical panels and the circumstance that they are not prepared for the installation of this kind of systems. This fact was particularly evident in the third deployment was no space to store the necessary hardware. Lastly, the *maintenance* of such long-term deployments was considerably challenging. In particular the need to constantly monitor all the installations to ensure that everything is working smoothly.

Moreover, it should be taken into consideration that constantly monitoring the status of the deployment will not necessarily mean that all the failures are detected in useful time. As such, we argue in favor of following a pro-active maintenance strategy in which the meters themselves are responsible for at least detecting and notifying the system administrators in case of failure.

5.3.1.2 Connectivity

In the wild research platforms rely heavily on the availability of stable network connectivity, for a number of reasons including data transmission and system maintenance. This dependency was particularly evident in our second sensing platform, since everything was done remotely.

Yet, contrary to what one would expect, Internet connections and particularly Wi-Fi are not widely available or easily accessible. This was the case of many of the homes monitored in deployment one, which lead to the creation of a had-hoc Local Area Network to provide Internet to participant households. Likewise, Internet access was also a constraint in the second and third deployments, since we had to contract Internet connections from a local provider in order to connect our energy monitoring base stations to the Internet.

Consequently, when deploying this kind of systems, it is important to take into consideration that Internet connections may represent extra costs. This will become even more important if the deployments happen in remote places where the only available connections are mobile (e.g., 3G or 4G) since these are normally more expensive than traditional DSL or cable connections.

5.3.1.3 Data Management

One of the most considerable challenges in our deployments was to cope with the rate at which data was generated by the deployed energy monitors. Taking as an example the power readings that are stored at one sample per minute and considering the 50 households, after just one week there will be 504 000 records in the database, and 2 160 000 after 30 days.

Consequently, making the right choice of database technology is a crucial step when deploying systems like this. In particular, there are two aspects that we consider of great importance, namely the *query performance* and the *physical size of the data*. The former is expected to greatly affect the performance of any systems that rely on the stored data, like for example, the eco-feedback applications, whereas the latter plays an important role regarding the selection of the hosting services.

Given the relevance of this issue in all the three deployments we have decided to go beyond the theoretical guarantees of data storage technologies and performed a benchmark between SQL and NoSQL database management systems. Regarding the former, we selected MySQL, which is probably the most widely used database management system and normally the first option of most researchers, including us. As for the later, we selected MongoDB since it is one of the fastest growing NoSQL solutions at that time.

The benchmark was performed using the SustData dataset. In one of the tests we wanted to evaluate how much disk space would be necessary to store the same amount of data in both technologies. To this end we performed the sequential insertion of 10 million power samples in MySQL and MongoDB and the results have shown that just after two million records (i.e., one month of power samples for 50 households) the size of the MongoDB database almost doubled the size of its MySQL counterpart (~0,95 GB vs. ~0.4 GB). Figure 5.16 shows a graphical representation of the obtained results.

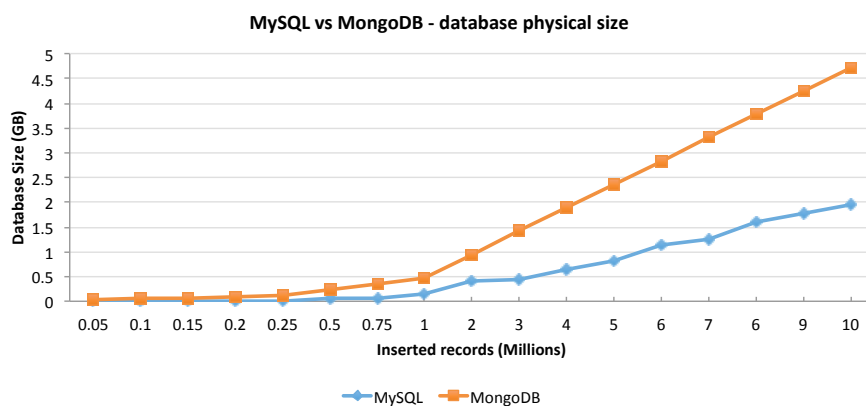


Figure 5.16 – MySQL vs. MongoDB: database physical size

In another test we wanted to measure how long it would take to query the data in each engine. To this end we selected the top five queries that were performed in our eco-feedback applications and executed each one of them using the same database sizes of the previous test, i.e., from 0.5 million up to 10 million records. Figure 5.17 shows a graphical representation of the average time it took to complete the five queries in each of the different database sizes. As it can be observed MongoDB clearly outperforms MySQL, a trend that becomes particularly evident after there is half a million records in the database (~7 days considering the same 50 households).

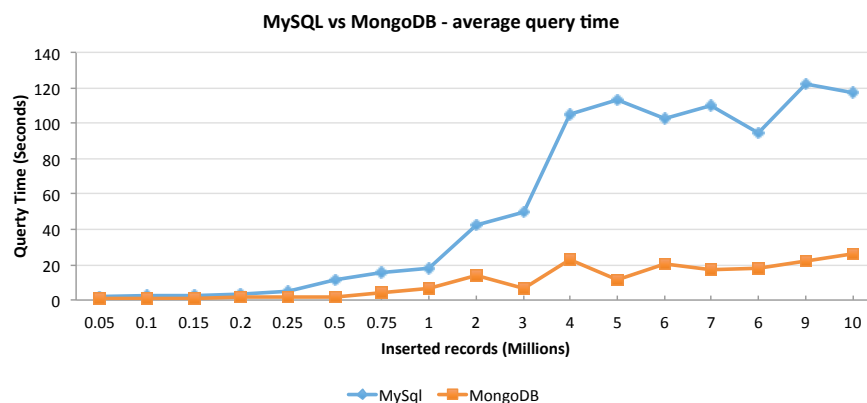


Figure 5.17 – MySQL vs. MongoDB: average query time

As it can be observed from the results of the two tests, there is a clear trade-off between the two database technologies. On the one hand MySQL takes considerably less disk space, but just after 250 k records the performance of the queries starts to degrade (4.71 seconds in MySQL against 1.78 seconds in MongoDB in average). On the contrary, MongoDB more than doubles the required disk space but manages to keep the query times in average 5 times faster than MySQL.

This said it is clear that at the end of the day, the most suitable database technology (or combination of technologies) is highly dependent on the type of application. For instance, in our particular case we are interested in providing the information to the user in the shortest period of time possible, thus the query time is much more relevant than the disk space.

Lastly, it is important to remark that this benchmark was done as part of a master thesis, and that here we are only summarizing the experiments that we consider relevant for this work. For further details please refer to the following publication [115].

5.3.2 Social Considerations

By social issues we refer to the different human-computer interaction related challenges that researchers face when running long-time projects. More particularly, in our research we have identified two mains categories of social issues, namely: i) maintaining a steady sample, and ii) physical location and security of the deployed systems.

5.3.2.1 Installing and maintaining a steady sample

From our experience deploying these systems in real world scenarios we realize that one of the most interesting challenges was to deal with the very different agendas of everyone involved. This fact become particularly clear during the installation phase of the first platform, which as it was already mentioned took 16 days to complete (and other 16 to remove) due to the difficulties in scheduling the visits to the houses.

Likewise, during the different deployments the research teams also experienced these difficulties when they had to engage with the householders to conduct the different eco-feedback research studies. Furthermore, we have also observed that with time participants tend to lose interest in the topic and opt to leave the experiment earlier, thus further limiting the size of the sample [11].

Consequently, we argue that field experiments like examining the effects of eco-feedback in the householders may be hard to implement and validate, unless the sample size is large enough to account for caveats like the inability to start and conclude all the experiments at the same time.

5.3.2.2 Physical location, security and intrusiveness

The fact that the first version of the monitoring platform was implemented using a netbook that had to be installed at the entrance of the monitored houses, presented some limitations.

Firstly, the system was not easily accessible to all family members in particular children, as one of the mothers shared with us: *She didn't reach it (youngest daughter 7 years old)*". In addition, the location of the netbook near the main power feed made it harder for family members to interact with the eco-feedback as some users were afraid of either dropping it on the floor or damaging the equipment since they considered it to be very fragile (the computer was stuck to the wall with sticky Velcro) and they did not own the system [11].

Likewise, some families also expressed concerns regarding the intrusiveness and safety of the system, even though it was properly and securely installed by a qualified electricians. For instance, some families did not allow their kids to come nearby or interact with the devices, fearing the risk of electric shock.

Finally, with regards to the second platform, since all the measurements were taken from the main electrical panel of the building and the eco-feedback was provided using mobile applications, we did not observe any major concerns regarding the security and intrusiveness of the equipment.

5.3.3 Cost Considerations

As it was already mentioned in the course of this thesis, we argue that one of the main reasons for absence of long-term studies on energy monitoring and eco-feedback are the costs associated with deploying the different systems. Thus, in order to provide an overview of how much studies of this nature can cost, we also report on the costs involved in our deployments.

More concretely, we explore the costs associated with the monitoring hardware, the energy required to run such energy monitors and the costs associated with storing the large amounts of data that are generated when running such experiments.

5.3.3.1 Hardware

Regarding the hardware costs associated with the single- and multi-house energy monitors the baseline costs were estimated based on the acquisition prices of the different components that comprise each solution. We also consider that the multi-house platform can monitor up to 10 houses.

The individual costs of each component are presented in Table 5.1 showing a comparing between the single-house and two different versions of the multi-house energy monitor (with and without tablet). A graphical representation is also provided in Figure 5.18.

Table 5.1 – Baseline hardware costs of the single- and multi-house energy monitors

Item	Unit Cost	Single-House		Multi-House	
		Qt.	Total	Qt.	Total
Netbook	230	1	230	1	230
Current Sensor	10	1	10	10	100
Voltage Sensor	15	1	15	3	45
Audio Splitter	3	1	3	-	-
LabJack U6 DAQ	338	-	-	1	338
Tablet	99	-	-	10	999
			258	1703 ^a	
^a Total per house: 170.3 (with tablet); 71.3 (without tablet)					

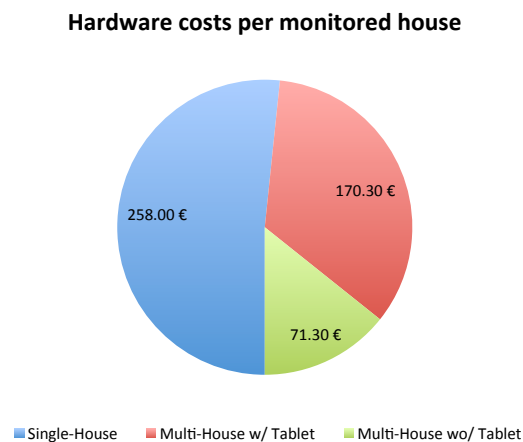


Figure 5.18 – Single- vs. Multi-House: Hardware costs associated with monitoring one house

As it was already expected, monitoring multiple houses from one single location is significantly cheaper than installing hardware in every house. Furthermore, it can also be observed that a substantial part of the costs is associated with the need to provide eco-feedback, hence indicating that the solutions will become much more cost effective when eco-feedback is provided using channels that do not require additional hardware.

In order to better understand the costs associated with energy monitoring deployments, we also compare a multi-house solution with a single-house monitor based on a low-cost credit card-size embedded computer, and an hypothetical multiple-sensor smart-meter.

With regards to the multi-house solution, we consider that the Raspberry Pi 3³⁰ embedded computer is used as a processing unit. As for the single-house solution, we consider that the system requires a dedicated processing unit and a dedicated DAQ board. More precisely, we assume that the single-house monitor is comprised of the BeagleBone Black³¹ rev. C and the PRUDAQ high-speed ADC³². Lastly, with respect to the multiple-sensor smart-meter, we consider that the system is able to monitor 10 different loads and the aggregate consumption by means of an additional whole house smart-meter. More precisely, we consider the

³⁰ Raspberry Pi, <https://www.raspberrypi.org/>

³¹ BeagleBone Black, <https://beagleboard.org/black>

³² PRUDAQ ADC, <https://github.com/google/prudaq/wiki>

CurrentCost smart-meter, which happens to be the less expensive solution in our benchmark. The baseline costs for the different solutions are presented in Table 5.2.

Table 5.2 - Baseline hardware costs of the single- and multi-house energy monitors

Item	Unit Cost EUR	Single-House		Multi-House		Multiple- Sensors	
		Qt.	Total	Qt.	Total	Qt.	Total
Raspberry Pi 3	40	-	-	1	40	-	-
BeagleBone Black rev C	45	1	45	-	-	-	-
LabJack U6 DAQ	338	-	-	1	338	-	-
PRUDAQ	60	1	60	-	-	-	-
Whole house meter	75	-	-	-	-	1	75
Individual Plug	31	-	-	-	-	10	310
Current Sensor	10	1	10	10	100	-	-
Voltage Sensor	15	1	15	3	45	-	-
		130		523^a		385	

^a This is the price for 10 houses.

In Figure 5.19 we show a comparison of the three solutions. As it can be easily observed, the multi-house solution is much more cost effective than the other solutions (e.g., costs 60% less than the single-house option). On the contrary, in a multiple-sensor solution the information comes at much higher costs. For example, even if we consider only two individual plugs, the cost per house would still be higher than the single-house version (137 EUR vs. 130 EUR).



Figure 5.19 – Single- vs. Multi-House vs. Multiple Sensors: Hardware costs associated with monitoring one house

In Figure 5.20 we compare the three different monitoring options with a projection up to 5000 houses.

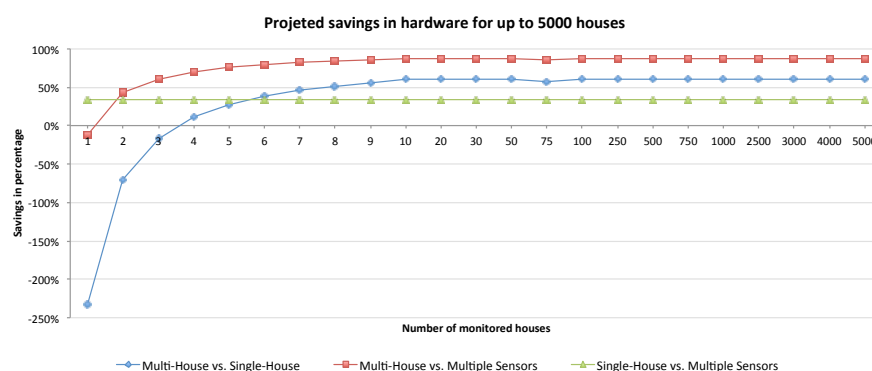


Figure 5.20 – Differences in hardware costs projected up to 5000 houses.

As it can be observed, the multi-house system costs in average less 60% than the single-house version and 86% when compared to a multiple sensor solution with 10 loads. Likewise, the single-house system is in average 34% less costly than the multiple sensors solution. Also noteworthy, is the fact that due to the higher costs associated with the data acquisition hardware, the multi-house option only becomes more cost-effective than the single-house monitor after at least four houses have been installed.

5.3.3.2 Consumed Energy

In this work we also look at the energy needed to run the energy monitoring devices, as this will impact the overall conclusions regarding the savings produced by the eco-feedback interventions.

To this end we considered the instantaneous power usage of each solution and projected the total consumption in kWh and EUR / kWh for different periods of time. To calculate the monetary cost we assume a baseline value of 16 cents per kWh, which is the current rate of the local provider in Madeira Island. The obtained results are presented in Table 5.3.

Table 5.3 – Estimated energy costs of the components that compose the monitoring solutions

Item	Watts	Day		Month		Year	
		kWh	EUR	kWh	EUR	kWh	EUR
Netbook	30	0.72	0.12	21.6	3.46	259.2	42.05
Embedded Computer / Single Sensor	5	0.12	0.02	3.6	0.58	43.2	6.91
Individual Plug	1	0.002	0.004	0.72	1.18	8.64	1.40

Then, given these estimates, we projected the costs in energy after one year of providing eco-feedback with a number of different energy monitoring solutions. More concretely, we considered the following scenarios:

1. Single-house with a notebook (used in the first deployment),
2. Multi-house with netbook (used in the second and third deployments),
3. Single-house with an embedded computer,
4. Multi-house with an embedded computer,
5. Multiple sensors, considering ten plugs, and
6. Ten individual plugs and a single-house meter.

The obtained estimates for each case are presented in Figure 5.21.

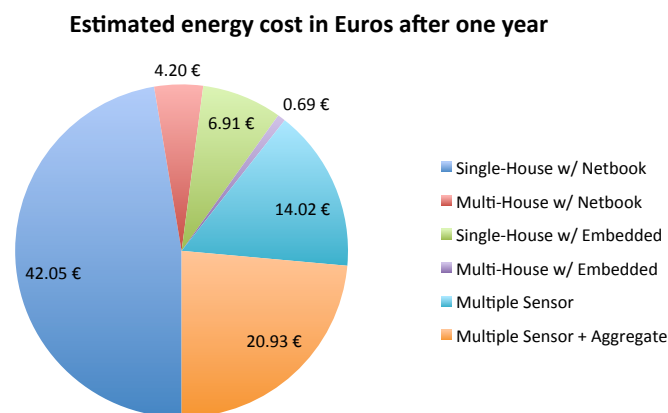


Figure 5.21 – Estimated energy costs of different energy monitoring solutions after one year

As it can be observed, and despite the fact that the notebook provides all the components needed to conduct eco-feedback research studies, the amount of energy that is consumed by that device is much higher than all the other solutions. This is particularly evident in the single-house solution where each monitored house represents a monthly energy cost of 3.5 Euros.

Another relevant observation is the fact that the ability to constantly monitor the energy consumption of 10 individual appliances will cost 1.7 Euros per month. The higher cost associated with multiple sensor solutions become even more evident when compared with those associated with the NILM solutions based on embedded computers. For example, the single-house solution has a monthly cost of 57 cents and the multiple house solution costs only about 5 cents a month.

5.3.3.3 Data Storage

Lastly, we looked at the cost associated with storing the data generated by such technologies. To this end, we consider a number of energy monitoring scenarios by assuming a fixed number of power events per day and varying the frequency at which the measurements are stored in the database.

In order to obtain a fair estimate of the number of power events per day we computed the daily average of power events from SustData, using the data from the third deployment. This

resulted in a daily average of 666 power events (SD: 645, $n=3347$). In Table 5.4 we show an estimate of the space that would be necessary to store the aggregate consumption of 1, 50 and 5000 houses after one and 12 months.

Note that we do not consider the overheads of the different storage technologies. Instead, we focus only on the size of the actual energy related records. More particularly, we consider that each record is composed of an identifier of type long (8 bytes), a date / time field (8 bytes), and five floating-point values (4 bytes each).

Table 5.4 – Projected amount of aggregate data that will be generated in one month and one year

Freq.	1 Month			1 Year		
	1 House	50 Houses	5000 Houses	1 House	50 Houses	5000 Houses
1 Hz	135 MB	6.7 GB	670	1.64 GB	82.24 GB	8.22 TB
1/6 Hz	22.5 MB	1.25 GB	112,5 GB	274 MB	13.7 GB	1.37 TB
1/15 Hz	9 MB	450 MB	45 GB	109 MB	5.48 GB	54.8 GB
1/60 Hz	2.3 MB	115 MB	11.5 GB	27.4 MB	1.37 GB	137 GB
15 Min	0.02 MB	10 MB	1 GB	1.8 MB	90 MB	9 GB

In Table 5.5 we show a projection of the space that necessary to store only the power event data generated for the same number of houses and time periods. To do this we consider three different sizes for the power event records. More specifically, we consider that each record contains an identifier and a data/time field (8 bytes each) and a varying number of power features represented as floating points (32, 112 and 328 bytes).

Table 5.5 – Projected amount of power event data that will be generated after one month and one year

Bytes per Event	1 Month			1 Year		
	1 House	50 Houses	5000 Houses	1 House	50 Houses	5000 Houses
48	0.9 MB	46.8 MB	4.6 GB	11 MB	558 MB	55.8 GB
128	2.46 MB	122.8 MB	12.28 GB	29.7 MB	1.49 GB	149 GB
344	6.5 MB	327 MB	32.7 GB	79 MB	3.99 GB	399 GB

In order to better understand the size of the data for each monitored household in Figure 5.22 we plot the projected costs for storing the energy consumption data of one household during two years. To this end we consider that the power events are stored in the 344 Bytes format, and that the monthly cost of storage is fixed in 3 Euro cents per GigaByte³³.

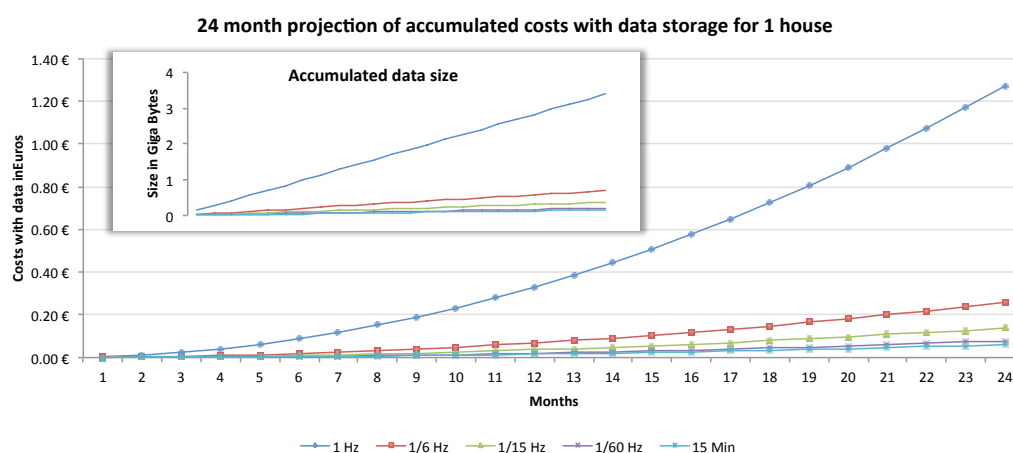


Figure 5.22 – Proportion of aggregate to power event data for one house after one year

As it can be observed, after two years an energy monitor that stores one power measurement every second will reach a hard disk quota of nearly 4 GB, but due to the current low prices of storage this will represent less than 1.40 Euros in total. These low costs are naturally even more evident in solutions that store less data, for example, a solution that stores one sample every 6 seconds (1/6 Hz) will cost less than 30 cents after two years.

5.4 Conclusion

In this chapter we have reported on the different technical, social and financial challenges behind building and deploying energy monitoring and eco-feedback systems in real world scenarios. We now summarize the answer to research question number one and then discuss the implications of our findings for future research in this field.

³³ Amazon S3 Storage, <https://aws.amazon.com/s3/pricing/> (visited on 4/8/2016)

5.4.1 Research Question

We now summarize the answer to the research question number one: “*what are the practical issues of deploying NILM and eco-feedback systems in real-world scenarios?*”

According to the experience we have gathered with incrementally developing and deploying energy monitoring and eco-feedback systems in real world experiments, we believe that the main practical issues can be divided in two categories, *technological* and *social* issues.

The former can be further categorized in three different dimensions, more precisely:

1. System installation and maintenance
2. Communication Infrastructure (i.e., Internet access)
3. Data management

And the later can be categorized according to two different dimensions:

4. Recruiting, installing and maintaining a steady sample throughout the whole study
5. Device location, security and intrusiveness

Lastly, concerning the financial costs of running such experiments, we believe that these should appear associated with the different technological and social issues. For example, hardware acquisition costs are an obvious technological issue. On the other hand, costs associated with running the smart-meters is more of a social issue in a sense that such costs may affect how households perceive smart-meters and their potential benefits, which to the best of our knowledge, is a topic that was not yet mentioned in literature.

5.4.2 Implications

We now draw some implications of this work to future research in this particular or similar fields that involve monitoring and feedback on the monitored variables. More particularly, the drawn implications are threefold: i) *technological*, ii) *social*, and iii) *financial*.

On the technical side, we have found that *Internet plays a key role* in research studies like this one, and that unfortunately it is not always readily available, which in the end can represent a considerable increase in the maintenance costs.

Furthermore, we have also learned that on the contrary to what many early vendors of smart-meters claim, the installation is not necessarily straight forward, and will most likely require the work from professional electricians. Additionally, we have observed that conducting studies where variables are constantly monitored will inevitably result in an explosion of data, and that as a consequence of this *the choice of the right database technology is of crucial importance* particularly in terms of user experience (time necessary to query the data).

On the social side, we have found from our deployments that security and intrusiveness are a major concern of the participating families, thus letting us believe that the best way to have participants engage in similar research studies is by providing a combination of *transparent energy monitoring and ubiquitous eco-feedback*. In other words, people would prefer not to see any monitoring hardware and eco-feedback should be provided using different modalities. Furthermore, we have learned that *conducting and validating real-world experiments can be very challenging* due to caveats that result mostly from the busy agendas of the participants.

Lastly, on the financial side we have observed that *NILM solutions tend to be significantly less expensive than multiple-sensor technologies*, not only in the initial acquisition costs but also in terms of the energy needed to run the systems. For example, we have seen that a multiple sensor solution with only two individual plugs will still have a slightly higher acquisition cost and will consume 30% more energy than a single-house NILM system.

Likewise, we have also learned that despite being limited to apartment buildings, the *multi-house platform presents several advantages when compared to the single-house platform*, in particular the fact that more houses can be monitored from a single location and also the fact that it has the potential to be further expanded to monitor two and three-phase electric systems without extra hardware.

Lastly, we have also observed that contrariwise to what would be expected, the *costs associated with storing the data that results from such longer-term experiments, are not so significant* when compared with the other costs. For example, in our scenarios, monitoring one house for five consecutive years using the single-house NILM will have an acquisition cost of 130 Euros and consume about 34 Euros of energy, but will cost less than two euros to store one power sample every six seconds and less than eight Euros in the more extreme case of storing one record per second.

In conclusion and taking into consideration all the practical issues reported above, we believe that in order to conduct successful long-term research studies not only in energy monitoring and eco-feedback but also in other domains where sensing and feedback is involved, it is important that the deployed systems enable **transparent monitoring** (i.e., all the monitoring hardware should be out of sight) and **ubiquitous feedback** (i.e., the householders should be able to access the monitored information in different modalities, independently of where they are).

Furthermore, considering all the costs associated with the different solutions, we believe that it is safe to say that in the long-term, even the more conservative NILM solutions (i.e., with a very limited number of disaggregated appliances) tend to have higher potential as tools to save energy than the multiple sensor solutions.

5.4.3 Limitations and Future Work

Although we have reached the goals that were initial set in this chapter, there are some limitations to this work that we would like to acknowledge.

First of all, the fact that the first deployment was conducted using a very unconventional smart-meter may have a direct influence in how the device was received by the householders. Furthermore, the second and third deployments were conducted without the need to install any hardware inside the households, which of course did not pose any issues related to the installation and security of the devices. As such, in future work we should seek to further

validate the social issues related with the physical location, security and intrusiveness of conventional smart-meters.

A second limitation of this work is the fact that our deployments were only targeted a very specific segment of consumers living in a modern city, which may also have implications on how these technologies are received and perceived by the participants. Consequently, in future live deployments of NILM and eco-feedback technologies we should target different consumer segments as these may have different needs and perceptions regarding smart-meters and eco-feedback technology. These new segments include for example, consumers from rural areas and most of all, consumers with micro-production installations, e.g., solar PV systems.

A third limitation of this work is directly related to the “in the wild” nature of our deployments, which prevented us from conducting more controlled experiments targeted specifically at the NILM problem. For instance, in one of the few NILM related experiments we deployed an interface to label power events with the goal of understanding how users would react to the possibility of labelling their own power consumption. Yet, after just one month into this experiment, we have noticed that none of the members from the selected households managed to label power events on their own. We believe that the main reason for this was the high number of events that would be shown in the user interface, as stated by one of the family members: *“I think the most complicated thing to do is the consumption per device (...) it’s complicated to manage such a large number of devices”*.

Consequently, future work should look at the possibility of conducting controlled deployments of NILM technology where it is possible to evaluate not only the social aspects of this technology, but also assess the performance of the underlying algorithms and systems.

Chapter 6 Experimental Comparison of Performance Metrics for Event Detection and Classification Algorithms

As discussed in section 2.5.3 of Chapter 2, many performance metrics have been defined in the literature with the aim of assessing the quality of NILM algorithms. Yet, many of those metrics are derived from other application domains in machine learning and thus it is not always clear how they will behave once applied to different NILM algorithms. In addition, machine-learning methods that perform well on one specific metric will not necessarily perform well on the others [37].

Consequently, it is a common practice to analyze how the different performance metrics correlate when applied to sub-sets of machine learning problems, such that it is possible to ascertain to what extent and in which situations the results and conclusions obtained using one particular metric can be extended to others.

In this chapter we extend the works from [37], [38] to the domain of event-based Non-Intrusive Load Monitoring. More concretely, we empirically analyze the behavior of a number of performance metrics across several event detection and event classification algorithms and datasets in order to identify and explain any observed similarities or dissimilarities between the different measures and algorithms.

The remaining of this chapter is organized as follows. In section 6.1 we provide extensive details of the algorithms used in this work. Then, in section 6.1.2.3.1, we present the datasets

against which the algorithms are executed, and in section 6.3 we thoroughly describe the performance metrics used to evaluate the two categories of algorithms. In section 6.4 we provide in-depth details of the overall research methodology that is followed in this work. Then, in section 6.5, we provide an extensive analysis of the obtained results, before we conclude this chapter in section 6.6. There we summarize the answer to the second research question of this thesis and highlight the implications of this work for future research in performance evaluation of NILM technology. We then describe the limitations of the work in this chapter and outline possibilities of improvement in future work.

6.1 Algorithms

In this section we thoroughly describe the different event detection and classification algorithms that are used in this experiment.

6.1.1 Event Detection

In subsection 2.4.1.1, we presented a review of event detection algorithms. However, many of the proposed solutions lack the implementation details. As such, it is unrealistic to implement and evaluate all the existing approaches in a timely manner.

Instead, we perform an in-depth analysis of five different algorithms for which the implementation details are available: one expert heuristic and four probabilistic detectors, which are summarized in Table 6.1. Next we provide detailed descriptions of the three base algorithms.

Table 6.1 – Event detection algorithms to be evaluated

Name	Type	Symbol
Meehan's et al. Expert Heuristic Detector [41]	Heu.	MEH
Log Likelihood Ratio Detector with Voting [31]	Prob.	LLD _{Vote}
Simplified Log Likelihood Ratio Detector with Maxima [47]	Prob.	SLLD _{Max}

Name	Type	Symbol
Log Likelihood Ratio Detection with Maxima	Prob.	LLD_{Max}
Simplified Log Likelihood Ratio Detector with Voting	Prob.	$SLLD_{Vote}$

6.1.1.1 Meehan's et al. Expert Heuristic Detector

The original version of Meehan's et al. heuristic event detector (MEH) was presented in [59]. It is based on a moving window that identifies changes in the root mean square (RMS) of the current signal through an expert heuristic that defines when a power event should be triggered. According to this algorithm, two criteria are required in order to trigger an event: i) the absolute amplitude of the RMS current in the second under test must be greater by a threshold value than the RMS current four seconds before, and ii) the previous event must not have occurred in the last three seconds.

Overall, the original algorithm contains three parameters that comprise its parameter space Ψ : a power threshold P_{thr} , which is set by default to 75% of the smallest appliance current RMS; the number of seconds before the second under evaluation G_{pre} , which is set to four seconds by default; and the minimum elapsed time between events T_{elap} , originally set to three seconds.

In our implementation of this algorithm we have extended the parameter space Ψ with four additional parameters. More precisely, pre- and post-event window lengths, w_{pre} and w_{post} , respectively; a power metric M_{pwr} ; and an event edge E_{edge} .

The pre- and post-event window lengths enable us to set the number of samples that will be averaged in order to find the difference in amplitude between different instants in time. This allows the creation of parameter combinations that are more robust to random noise that otherwise could be considered power events. The power metric parameter will allow us to test the algorithm using other power metrics, thus enabling us to compare the outcomes of this algorithm against detectors that use other power metrics (e.g., apparent and real power).

Lastly, the event edge parameter is used to enable the evaluation of the obtained results against the ground truth data. In other words, the event edge is the sample index inside the

second where the event occurred, e.g., an event edge of zero means that the event happened in the first sample of that second (assuming zero-indexing). It should be noticed that the event edge should always be a value between 0 and the frequency of the signal minus one (50 Hz or 60 Hz). Table 6.2 below summarizes the parameter space Ψ_{MEH} of this algorithm.

Table 6.2 – Parameter space of Meehan s et al. expert heuristic event detector

Parameter	Symbol	Description
Pre-event gap	G_{pre}	Seconds before the second under test (Seconds)
Power threshold	P_{thr}	Minimum absolute power change of interest (Watts)
Elapsed time	T_{elap}	Minimum elapsed time before last power event (Seconds)
Pre-event window	w_{pre}	Seconds to average before the second under test (Seconds)
Post-event window	w_{post}	Seconds to average in the second under test (Seconds)
Power metric	M_{pwr}	Power metric against which the detector will be executed
Event edge	E_{edge}	Index of the sample inside the power event (Sample)

In Figure 6.1 we show an illustration of the event detection process using the MEH algorithm. The above parameters are set to the following values: $P_{thr} = 100$ W, $T_{elap} = 3$ seconds, $w_{pre} / w_{post} = 50$ samples, $M_{pwr} = \text{active power}$, and $E_{edge} = 0$.

In the first step the absolute active power changes between consecutive windows of 50 samples (i.e., one second) are calculated. In Figure 6.1 – top we show the active power and the obtained absolute power changes.

Then, in the second step, only the power changes above the P_{thr} parameter are selected. In this particular example, only six power changes are selected (Figure 6.1 – centre).

Finally, in the third step, the valid power changes are filtered according to the T_{elap} parameter. As it can be observed, ultimately only three power events were considered valid (Figure 6.1 – bottom).

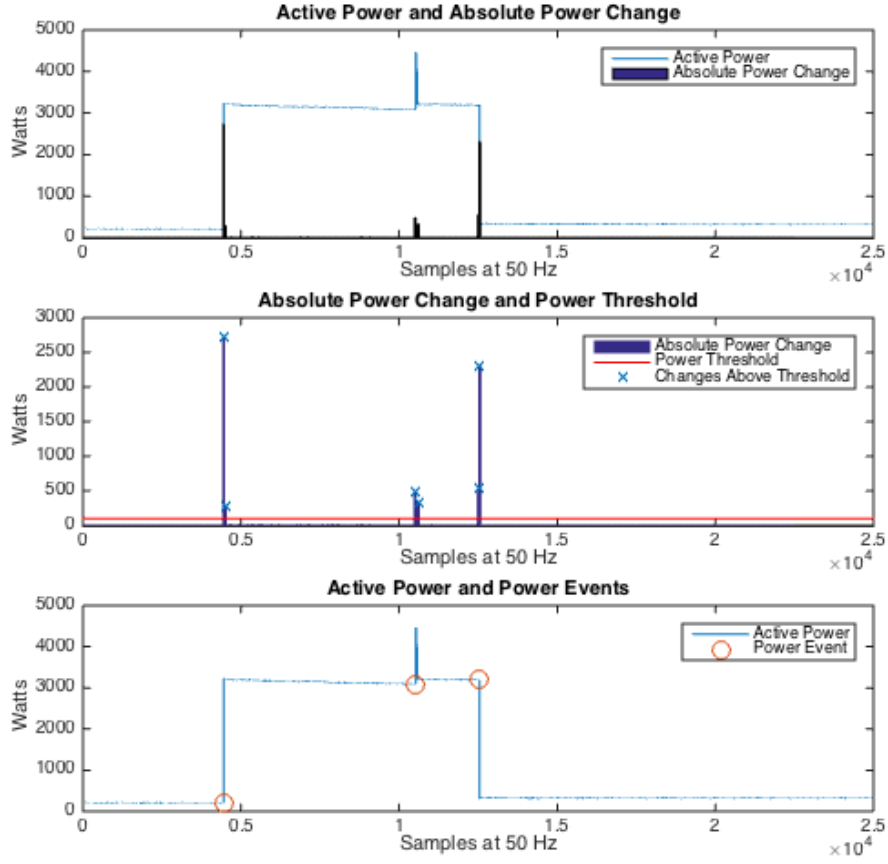


Figure 6.1 – Illustration of the MEH event detection process. Real power and absolute power changes (top), threshold filter (center), elapsed time filter (bottom)

6.1.1.2 Log-Likelihood Ratio Detector

The Log-Likelihood Ratio event detector (LLR) [46] is an extension of the Generalized Likelihood Ratio event detector (GLR) [42] already mentioned in sub-section 2.4.1 (Event-based Approaches).

Like the GLR, the LLR event detector makes use of the log-likelihood ratio test [42] to calculate the likelihood of a potential change in the mean value of two sequential windows (pre- and post-event windows, respectively). However, unlike the GLR algorithm that thresholds the detection statistic to find the potential power events, (i.e., a power event is signaled once a pre-defined threshold is exceeded in the detection statistics), the LLE detector employs a voting scheme on the detection statistic output in order to signal potential power events.

Our implementation of the LLR detector (and other probabilistic detectors like the SLLR [47] that will be presented next) consists of two different algorithms. The first, to which we refer to as *Detection Statistic*, is used to calculate the power event likelihood. The second, referred to as *Detection Activation*, is used to extract the power events from the signal generated by the Detection Statistic algorithm.

6.1.1.2.1 Detection Statistics

The LLR detection statistics algorithm works with one sliding window (*detection statistics window - dws*) that is used to calculate the likelihood of a change of mean happening at a given sample.

The *dsw* is composed of two separate windows, a pre-event (w_0) and a post-event (w_1) window, and for each sample n in the power metric P the detection statistic $S[n]$ is given by equation (6.1).

$$S[n] = \ln \left(\frac{\sigma_{0,n}}{\sigma_{1,n}} \right) + \frac{(P[n] - \mu_{0,n})^2}{2 \times \sigma_{0,n}^2} - \frac{(P[n] - \mu_{1,n})^2}{2 \times \sigma_{1,n}^2} \quad (6.1)$$

Where $\mu_{0,n}$, $\sigma_{0,n}^2$, $\mu_{1,n}$, and $\sigma_{1,n}^2$ are the sample mean and variance of the pre- and post-event windows, respectively. These values are calculated using equations (6.2) to (6.5), where w_0 and w_1 are the pre- and post-event windows lengths.

$$\mu_{0,n} = \frac{1}{w_0} \times \sum_{k=n-w_0}^{n-1} P[k] \quad (6.2)$$

$$\mu_{1,n} = \frac{1}{w_1} \times \sum_{k=n+1}^{n+w_1} P[k] \quad (6.3)$$

$$\sigma_{0,n}^2 = \frac{1}{w_0 - 1} \times \sum_{k=n-w_0}^{n-1} (P[k] - \mu_{0,n})^2 \quad (6.4)$$

$$\sigma_{1,n}^2 = \frac{1}{w_1 - 1} \times \sum_{k=n+1}^{n+w_1} (P[k] - \mu_{1,n})^2 \quad (6.5)$$

Lastly, from the detection statistics signal, $S[n]$, the authors created a modified log-likelihood ratio $l[n]$ by forcing it to be equal to zero when the absolute difference between w_0 and w_1 is below a threshold P_{thr} . Ultimately, the log-likelihood ratio of a power event occurring at sample n is given by the equation (6.6).

$$l[n] = \begin{cases} S[n], & |\mu_{1,n} - \mu_{0,n}| > P_{thr} \\ 0, & otherwise \end{cases} \quad (6.6)$$

Overall, the detection statistics algorithm for the LLR detector parameter space Ψ_{LLR} is constituted of three adjustable parameters: a pre-event window size w_0 ; a post-event window size w_1 ; and a power threshold P_{thr} as summarized in Table 6.3 below:

Table 6.3 – Parameter space for the Log Likelihood Ratio event detector

Parameter	Symbol	Description
Pre event window	w_0	Length of the pre-event window (Samples)
Post event window	w_1	Length of the post-event window (Samples)
Power threshold	P_{thr}	Minimum absolute power change of interest (Watts)

6.1.1.2.2 Detection Activation

The original implementation of the voting algorithm presented in [46] works by sliding a voting window (w_v) across the log-likelihood $l[n]$ and assigning a vote in each shift of the window to the point with the largest absolute magnitude that is greater than zero. Next, for each sample in the log-likelihood the votes are accumulated and the samples with a number of votes greater than a voting threshold (V_{thr}) are signaled as being power events.

In our implementation of the voting algorithm we have added an extra parameter to the parameter space Ω_{VA} , more precisely a log-likelihood threshold (l_{thr}). This parameter is used to enhance the voting schema by not allowing votes in samples below a specific threshold. Furthermore, by setting the likelihood threshold to a value greater than zero and the voting threshold to zero we end up with a parameter configuration that is equivalent to the original GLR event detector. The parameter space, Ω_{VA} , of the voting algorithm is presented in the Table 6.4:

Table 6.4 – Parameter space of the voting algorithm used in the LLR event detector

Parameter	Symbol	Description
Voting window	w_v	Length of voting window (Samples)
Voting threshold	V_{thr}	Minimum votes necessary to trigger an event (Count)
Log-likelihood threshold	I_{thr}	Minimum absolute log-likelihood value that must be met to cast a vote to sample n (Likelihood)

In Figure 6.2 we show an illustration of the event detection process using the LLD algorithm with voting. The different parameters are set to the following values: $P_{thr} = 100$ W, $w_{pre} / w_{post} = 50$ samples, $w_v = 10$ samples, $V_{thr} = 9$, and $I_{thr} = 0$.

In the first step, a detection statistic is calculated for each power sample. This is done using equation (6.1). The resulting detection statistic for each sample is depicted on the top of Figure 6.2 (scaled by a factor of $10^{-1.5}$ for better visualization).

Next, a voting schema, with a 10-samples window, is applied to the detection statistic signal. In this step the votes are filtered (V_{thr}), and only detection statistic values with more than nine votes are considered events. In this example, six samples have more than 9 votes, as it can be observed from the representation in the center of Figure 6.2.

Finally, in the third step, the valid detection statistic indexes are mapped to the power signal, and the respective power events are extracted (see Figure 6.2 – bottom).

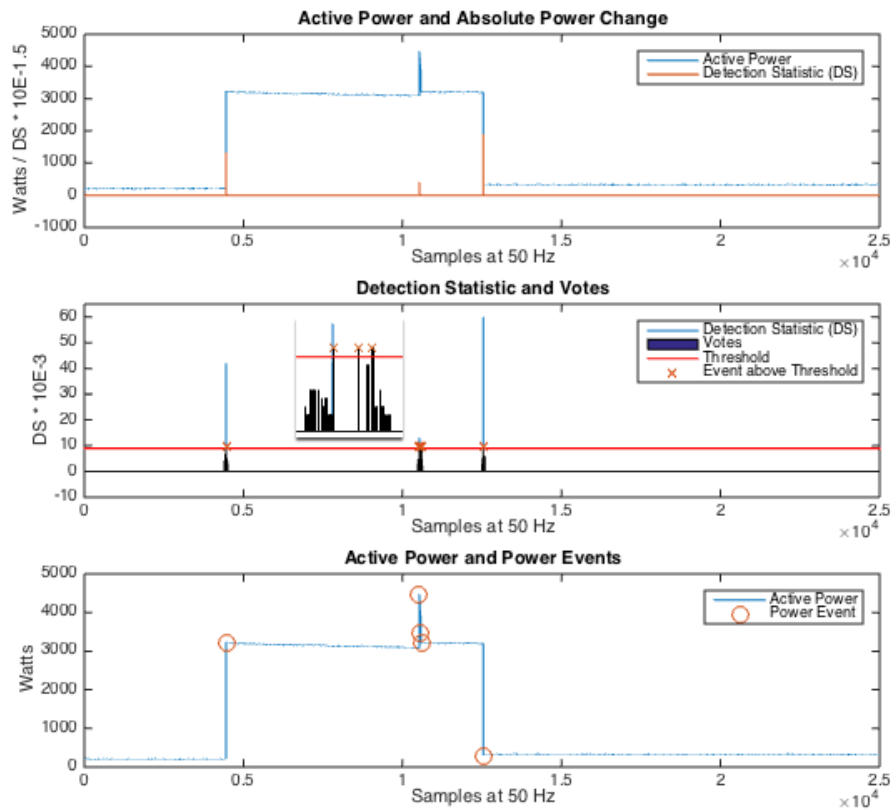


Figure 6.2 – Illustration of the LLD event detection process. Active power and detection statistic (top), voting procedure (center), and votes filtered by threshold (bottom)

6.1.1.3 Simplified Log-Likelihood Ratio Detector

Like the GLR and the LLR, the SLLR detector also makes use of the log likelihood ratio test to calculate the likelihood of a potential change in the mean value of two sequential windows. More specifically, this algorithm uses a simplified version of the likelihood equation and employs a maxima / minima locator algorithm on the detection statistics output in order to identify potential power events.

6.1.1.3.1 Detection Statistics

The implementation of the SLLR algorithm is similar to the one of the LLR, with the exception of the way that the detection statistic is calculated. More precisely, for each sample

n in the power metric P , the detection statistics of the SLLR algorithm is given by equation (6.7).

$$S[n] = \frac{\mu_{1,n} - \mu_{0,n}}{\sigma_n^2} \times |P[n] - \mu_n| \quad (6.7)$$

Where $u_{0,n}$ and $u_{1,n}$ are the sample mean of the pre- and post-event windows, respectively; And u_n and σ_n^2 are the mean and variance of the detection statistics window, correspondingly.

Likewise, in this algorithm we also force the log-likelihood ratio $l[n]$ to be equal to zero when the absolute difference between w_0 and w_1 is below a threshold P_{thr} using equation (6.6). Lastly, it is important to remark that the parameter space for this algorithm, Ψ_{SLLR} , is the same as the parameter space of the LLR algorithm, Ψ_{LLR} .

6.1.1.3.2 Detection Activation

The detection activation of the SLLR algorithm works by sliding a maxima/minima finder window (w_M) across the absolute value of the log-likelihood $l[n]$ looking for the local maxima. The length of the window is equal to twice the maxima precision plus one ($2M_{pre} + 1$). For each shift of the window the sample in the middle will be signaled as a power event if its absolute value is larger than the absolute value of all the M_{pre} samples to its left and right.

The parameter space, Ω_{MAX} , of the maxima finder algorithm consists of two tunable parameters: a maxima precision M_{pre} ; and a log-likelihood threshold l_{thr} . The M_{pre} is used to make the process of finding the maximum values more stable, whereas the l_{thr} can be used to prevent power events with an absolute value lower than a specific value from being signaled. The parameter space Ω_{MAX} is summarized in Table 6.5 below:

Table 6.5 – Parameter space of the maxima algorithm used in the SLLR detector

Parameter	Symbol	Description
Maxima Precision	M_{pre}	Number of consecutive samples that must be lower than sample n for it to be considered a maximum (Seconds)
Log-likelihood threshold	l_{thr}	Minimum absolute log-likelihood value that must be met to call a maximum at sample n (Likelihood)

It is important to remark that, like the T_{elap} parameter in the MEH algorithm, the M_{pre} parameter will prevent power events separated by less than M_{pre} samples from being signaled. Also, it should be noticed that if M_{pre} is set to zero and l_{thr} is set to a value greater than zero this algorithm will be similar to the original GLR, with the exception of the likelihood function.

In Figure 6.3 we show an illustration of the event detection process using the SLLD algorithm with voting. The different parameters are set to the following values: $P_{thr} = 100$ W, $w_{pre} / w_{post} = 50$ samples, $M_{pre} = 50$ samples, and $l_{thr} = 0$.

In the first step, a detection statistic is calculated for each power sample. This is done using equation (6.7). The resulting detection statistic for each sample is depicted on the top of Figure 6.3 (scaled by a factor of 10^3 for better visualization).

Next, a maxima/minima location algorithm, with a tolerance of 50 samples, is applied to the detection statistic signal. In this example, two local maxima and one local minimum are found, as it can be observed from the representation in the center of Figure 6.3.

Finally, in the third step, the maxima / minima indexes of the detection statistic signal are mapped to the power signal, and the respective power events are extracted (see Figure 6.3 – bottom).

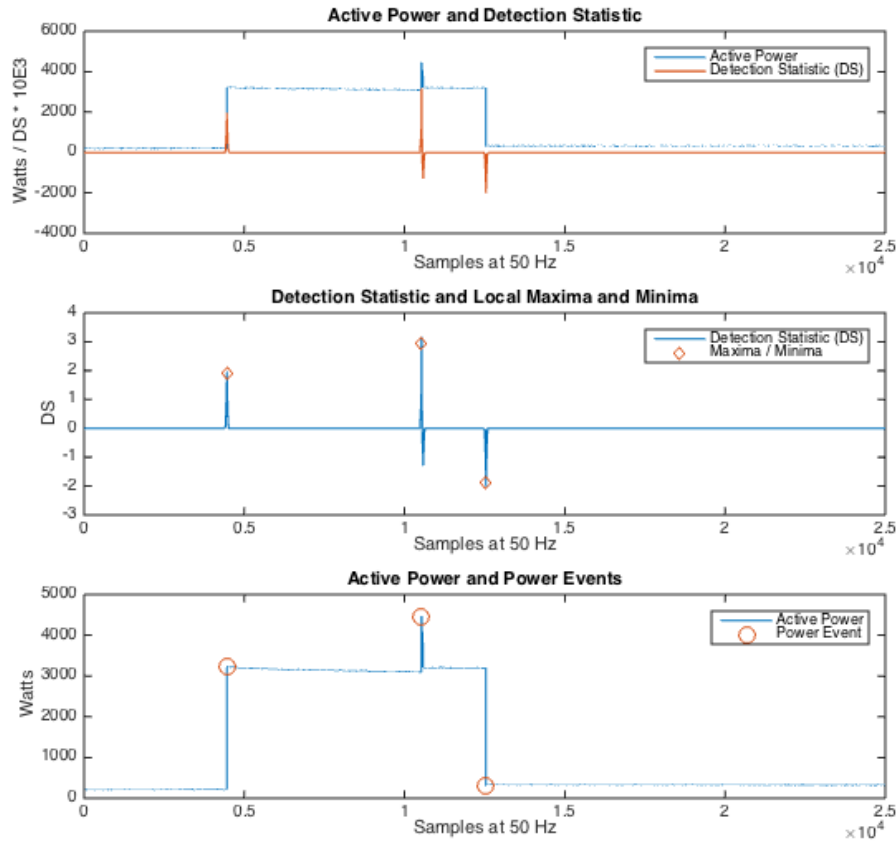


Figure 6.3 – Illustration of the SLLD event detection process. Active power and detection statistics (top), detection statistics and local maxima (center), active power and power events (bottom)

6.1.2 Event Classification

As it was mentioned in Chapter 2, many supervised learning algorithms have been already applied to the NILM problem. Here, and again due to the fact that is impossible to implement and / or evaluate all the existing approaches, we have decided to implement the supervised learning algorithms shown in Table 6.6.

In order to proceed with the evaluations we use the Waikato Environment for Knowledge Analysis (Weka) [130] machine learning software which already incorporates multi-class implementations of the six algorithms. Next we briefly describe the selected algorithms.

Table 6.6 – Selected classification algorithms

Lazy Learning	Eager Learning
K-Nearest Neighbor (K-NN) [131]	Decision Trees (DT) [132]
K-Star (K*) [133]	Artificial Neural Networks (ANN) [134]
LWL with Naïve Bayes (LWL-NB) [135]	Support Vector Machines (SVM) [136]

6.1.2.1 Lazy Learners

Lazy learning is a learning method in which the processing of the examples is deferred until an explicit request is received (e.g., a new unlabeled instance is given to a classifier) [137].

The main advantage of such methods is that since the generalization step is only performed when an unlabeled example is provided, they can successfully adapt to previously unseen data. For example, in the particular case of NILM, when a new appliance is added to the grid, a lazy classifier will be able to correctly classify new instances of that appliance when enough labeled examples of it are added to the signatures database.

On the other hand, the disadvantages of lazy learners include the large size of the training database since the training data is always needed in the inference phase. Another disadvantage is the fact that these methods usually have a slower evaluation phase, which tends to degrade as more examples are added to the signature database.

6.1.2.1.1 K-Nearest Neighbor

The K-Nearest Neighbor (K-NN) is a very simple supervised lazy learning algorithm that classifies new instances based on a similarity measure (e.g., distance functions). In other words, a new instance is classified by a majority vote of its neighbors, with the label being assigned to the class most common amongst its K nearest neighbors. If $K = 1$, then the case is simply assigned to the class of its nearest neighbor.

Regarding the parameter space of this algorithm, there are a number of different parameters that must be set in advance. These include, the K number of neighbors, the distance function D , and an optional distance weighting function (D_{weight}).

6.1.2.1.2 K-Star

The K-Star (K^*) is another supervised lazy learning algorithm that classifies new instances based on a similarity measure. However, instead of relying on traditional distance functions, in K^* the distance between two instances is motivated by information theory. The basic intuition behind K^* is that the distance between instances can be defined as the complexity of transforming one instance into another [133].

Ultimately, in the WEKA implementation of this algorithm, only one parameter needs to be defined, which is the number of neighbors that should be considered when performing the instances transformation.

Stated more concretely, in the K^* algorithm, the number of “important” neighbors is specified using the blending parameter (b), which varies between $b = 0\%$ and $b = 100\%$. The $b\%$ nearest neighbors are then selected using the Euclidian distance function. Finally, the entropic distance between the unlabeled instances to the $b\%$ nearest neighbors is calculated and a majority-voting scheme is applied to select the closest. As such, when selecting $b = 0\%$, the K^* algorithm behaves exactly like the nearest neighbor algorithm ($K = 1$), and selecting $b = 100\%$ gives equally weighted instances, i.e., all instances are equally important.

6.1.2.1.3 Locally Weighted Learning with Naïve Bayes

Locally Weighted Learning (LWL) is a class of lazy learning algorithms, where predictions are made using an approximated local model around the current point of interest instead of building global models for the entire training data [138].

Regarding the WEKA implementation of LWL, two parameters must be set in advance: i) the nearest neighbor search algorithm, and ii) a learning model to be fit to the selected neighborhood. Stated more concretely, in this work, we use a combination of K-NN to set the neighborhood, and a Naïve Bayes classifier for classification.

6.1.2.2 Eager Learners

Eager learning is a learning method in which the system tries to construct a general target function from the training data [137]. Then, when a new unlabeled instance requests a classification, its features are fed to the eager classifier that will output a possible label based on the provided inputs.

The main advantage of such methods is that since the generalization is done in advance, it is not necessary to store the training data. As such, eager learners require much less space than lazy learners.

On the other hand, one of the main disadvantages of such methods is that they do not cope well with previously unseen data. For example, in the case of NILM, whenever a new appliance is added, the system must be re-trained with training data that includes labeled examples of the new appliance.

6.1.2.2.1 Decision Trees

Classification and regression trees are eager machine-learning methods for constructing prediction models from data. The models are obtained by recursively partitioning the data space and fitting a simple prediction model within each partition. As a result, the partitioning can be represented graphically as a decision tree. Decision tree learning (or induction) is the process of constructing decision trees from labeled sets of training data.

In very simple terms, a decision tree can be seen as an inverted tree structure that represents the partitioning of the training data. The topmost node is the root node, the internal nodes denote tests on particular attributes, each branch represents a test outcome, and the terminal node (or leafs) hold the possible outputs of the learner model (class labels in the case of classification trees, or real numbers in the case of regression trees).

There are many types of decision tree algorithms, some of which are available in the WEKA platform [139]. In this work we implement the J48 algorithm, which is a slight variation of the C4.5 algorithm [140].

Regarding the parameter space of this algorithm, only a few parameters must be set in advance. These include the minimum number of instances required to open a new branch (*minNumObjs*), if tree pruning is activated (*prun*) and the confidence factor of the pruning process (C_{prun}) that is only used if tree pruning is activated.

6.1.2.2.2 Artificial Neural Networks

Artificial Neural Networks (ANNs) are another eager learning method for constructing predictive models from labelled training data. A key feature of neural networks is the iterative learning process in which the training data are presented to the network one at a time, and the weights associated with the input values are adjusted each time. During this learning phase, the network learns by adjusting the weights such that it is able to predict the correct class label of the training samples.

In the case of ANNs, there are a considerable number of parameters that must be set in advance, some of which may result in considerable differences in the network structure and the learning process. The more common parameters include the number of layers (*numLayers*), the number of hidden neurons (*numHiddenNeurons*), the learning rate (*learningRate*) and the momentum (*momentum*). The first two affect mostly the network structure, whereas the latter two have a direct influence in the learning process. For example, the learning rate parameter defines the amount at which the weights and bias are updated at each step of the training phase.

6.1.2.2.3 Support Vector Machines

Support Vector Machines (SVMs) are a category of eager and supervised algorithms that can be used to perform linear and non-linear classification. Algorithms under this category work by constructing hyperplanes in high- or infinite- dimensional spaces by implicitly mapping the inputs (features) to high-dimensional feature spaces by means of Kernel functions [136].

One important characteristics of SVMs is that they always return the hyperplane (or hyperplanes) with the largest distance to the nearest training-data of each class (support

vectors), under the general assumption that the larger the margin the lower the generalization error of the classifier. To achieve this, in the training phase, SVMs allows some examples to be misclassified, such that the margin can be more easily maximized [136].

The number of misclassification is implicitly controlled by the misclassification cost parameter (C). Smaller values of C mean that a small penalty is given to the misclassifications, hence resulting in larger margins. On the contrary, larger values of C will result in smaller margins, since the algorithm will try to make as few classification errors as possible to avoid the high penalizations.

In the current version of WEKA, a number of classification SVMs algorithms and kernel functions are readily available for classification problems. Still, in this work we only implement the C -Support Vector Classification formulation (C-SVC [141]) with a Radial Basis Function (RBF) kernel as suggested in [142].

Gamma (γ) is the only parameter of the RBF kernel, and is used to control the level of influence of each training example in its neighborhood. A low value of gamma means high-influence and reduces the number of misclassifications, whereas, a high gamma means low-influence and favors misclassifications.

6.1.2.3 Learning Features

Regarding the learning features, in this work we evaluate each classification algorithm using 30 feature sets, which are composed of 13 different features. Next we briefly describe the individual features. Additional details about the 30 feature sets can be found in 6.4.3.2 (Feature sweep).

6.1.2.3.1 Delta features

Delta features are by far the simplest features in event-based classification algorithms. Quantitatively speaking, delta features measure the average amount of change of a particular power metric, and are extracted by computing the difference between the average values of

the samples in a post- and a pre-event window – see equation (7.23). In this particular case we are using the delta features for real power (P), reactive power (Q) and current RMS (I).

6.1.2.3.2 Harmonic features

Despite the presence of harmonic powers in the grid is not the most desirable situation, as it can degrade the mains efficiency, they provide a very attractive method of characterizing the different electric loads. Here we are using current ($H_{I, n}$) and instantaneous power ($H_{IV, n}$) harmonics up to the 21st component.

6.1.2.3.3 Raw and quantized waveforms features

Raw waveform features consist of a number of measurements of a particular metric taken from within the vicinity of the power event. As for the quantized waveforms, these are down-sampled versions of the raw waveforms that are obtained by quantizing the raw data into n bins.

In this work we are using raw and quantized measurements taken from one period of instantaneous current (I_{WF} and I_{QWF}) as well as quantized measurements taken from one period of instantaneous current combined with one period of instantaneous voltage (IV_{QWF}). For the quantization procedure we set $n = 20$ and each bin is represented by the respective median.

6.1.2.3.4 Data-driven features

As it was mentioned in sub-section 2.4.1, data driven features are learned directly from the data without the necessity of incorporating any domain knowledge. Here we use the set of features that was identified and explored in [58]. More particularly we use VI binary images (VI_{BIN}) and a number of principal components extracted from the binary images (BIN_{BIN_PCA}) and the raw and quantized waveforms (I_{WF_PCA} , I_{QWF_PCA} and IV_{QWF_PCA}).

6.2 Datasets

In this section we briefly describe the datasets that will be used to evaluate the different event detection and classification algorithms. Additional details can be found in Appendix B.

6.2.1 Event detection

As we mentioned in section 2.5.1, at the time of writing of this thesis only BLUED [77] contains a comprehensive list of appliance labels thus making it the unique dataset that can be used to benchmark event detection algorithms.

Consequently, in order to proceed with our work, we had to manually label some of the already existing datasets. More precisely, we provided labels for one week of data from two houses in the UK-DALE dataset [89] that contain high frequency measurements at 16 kHz (houses 1 and 2). Table 6.7 and

Table 6.8 below summarize the four datasets that will be used to evaluate the event detection algorithms. See Appendix B for additional details.

Table 6.7 – Summary of the datasets used to evaluate detection algorithms

Dataset Name	IV Freq.	PQ Freq.	Duration	Events
UK-DALE house 1	12.8 kHz	50 Hz	7 days	5440
UK-DALE house 2	12.8 kHz	50 Hz	7 days	2842
BLUED phase A	16 kHz	60 Hz	7 days	887
BLUED phase B	16 kHz	60 Hz	7 days	1562

Table 6.8 – Summary of the active power change and elapsed time between power events in the event detection datasets

Dataset Name	Active Power Change (W)				Time Between Events (S)			
	Mean	25 %	50 %	75 %	Mean	25 %	50 %	75 %
UK-DALE house 1	268	48	100	273	111	4	7	28
UK-DALE house 2	365	45	74	137	212	6	15	172
BLUED phase A	274	84	116	582	690	18	294	892
BLUED phase B	351	40	170	428	383	7	35	83

6.2.2 Event classification

Regarding the event classification algorithms benchmark we decided to use the PLAID dataset [143], which contains current and voltage measurements for 11 appliance types measured across 55 houses.

Recent works with this dataset have considered each house in PLAID as if it was a different dataset [58], [71]. In what can be considered a variation of the 1-fold cross validation technique, i.e., the data from one house is used as test data while the remaining 54 houses are used to train the learning algorithms. Here instead we have decided to split PLAID into eleven different datasets where each one is constituted by the data of five houses.

The reasons behind this decision are twofold: i) to compensate for the fact that the number of instances can be considerable different between houses (e.g., in the most extreme case we have one house with only two events and another with thirty six, and ii) to have a more manageable number of datasets when performing the different benchmarks. Table 6.9 below summarizes the datasets that will be used to evaluate the event classification algorithms. The subscripts x - y under the dataset name identifies the houses in each dataset, e.g., the houses 1 to 5 constitute PLAID₁₋₅.

Table 6.9 – Datasets used for event classification

ID	Name	Appliances	Events
----	------	------------	--------

		Types	Instances	
1	PLAID ₁₋₅	8	24	112
2	PLAID ₆₋₁₀	7	18	86
3	PLAID ₁₁₋₁₅	8	24	117
4	PLAID ₁₆₋₂₀	9	19	95
5	PLAID ₂₁₋₂₅	10	22	107
6	PLAID ₂₆₋₃₀	9	19	90
7	PLAID ₃₁₋₃₅	10	30	129
8	PLAID ₃₆₋₄₀	8	11	66
9	PLAID ₄₁₋₄₅	7	15	61
10	PLAID ₄₆₋₅₀	10	20	87
11	PLAID ₅₁₋₅₅	9	26	124
---	Total	---	228	1074

6.3 Performance Metrics

In this section we thoroughly described the performance metrics that are explored in this work. We first describe the metrics for event detection and then for event classification.

6.3.1 Event Detection

A characteristic of residential power data is that appliance activity is sparsely distributed. As a consequence, for an event detector with reasonable performance it is expected that the number of true negatives (TN) will be much higher than the number of true positives (TP), false positives (FP) and false negatives (FN) [39]. This effect is not exclusive to the NILM problem. It is, for example, common in information retrieval problems where the number of irrelevant documents than can be returned after a specific query is much higher than the number of actual relevant items. As such, it is not totally unexpected that NILM researchers

have adapted performance metrics used in the information retrieval domain to evaluate event detection algorithms, like for example, *precision* and *recall* [144].

In this work we will look at these two performance metrics as well as other *confusion matrix* and *rank / score* based metrics that are commonly used to evaluate this class of problems. Furthermore, we will also look at performance metrics that were specifically created for event detection problems [39] that we refer to as *Domain Specific Metrics*.

6.3.1.1 Confusion matrix based metrics

As the name suggests, confusion based metrics are directly derived from the values in the confusion matrix. In Table 6.10 we summarize the confusion matrix based performance metrics used to evaluate the event detection algorithms. Here *Best* and *Worst* refer to the best and worst values that each metric can report.

Table 6.10 - Summary of performance metrics for event detection

Metric	Symbol	Best	Worst
Accuracy	A	1	0
Error rate	E	0	1
Precision	P	1	0
Recall	R	1	0
F_1 -Measure	F_1	1	0
$F_{0.5}$ -Measure	$F_{0.5}$	1	0
F_2 -Measure	F_2	1	0
P-R Distance to Perfect	DTP_{PR}	0	2
False Positive Rate	FPR	0	1
TPR – FPR Distance to Perfect	DTP_{Rate}	0	2
True Positive Percentage	TPP	1	0
False Positive Percentage	FPP	0	1
TPP-FPP Distance to Perfect	DTP_{Perc}	0	*

Metric	Symbol	Best	Worst
Standardized MCC	<i>SMCC</i>	0	1
* Since the FPP metric can return a value greater than one it is not possible to define a fixed lower bound.			

6.3.1.2 Rank metrics

Rank (or ordering) metrics can be thought of as summaries of the performance of a learned model over varying decision criteria. One of such measures is the area under the *ROC* curve (*AUC*), which is drawn by varying the discrimination threshold of a classifier, and calculated by using the trapezoidal rule.

However, for discrete algorithms where fixed labels are produced (the case of event detection), the *AUC* should not be measured by employing the trapezoidal rule since the eventual presence of outliers could lead to distorted results [145]. Instead, the nonparametric Wilcoxon statistic should be used, as shown by Hanley and Mcneil [146]. Table 6.11 below summarizes the rank metrics that we will use in this work. A more detailed explanation of each metric can be found in [145].

Table 6.11 – Summary of rank metrics for event detection algorithms

Metric	Symbol	Best	Worst
Wilcoxon statistics based ROC AUC	<i>WAUC</i>	1	0
Wilcoxon statistics based ROC AUC Balanced	<i>WAUCB</i>	1	0
Biased AUC	<i>BAUC</i>	1	0
Geometric mean AUC	<i>GAUC</i>	1	0

6.3.1.3 Domain specific metrics

Domain specific metrics for event detection were first introduced in [68] motivated by the fact that metrics based solely on the confusion matrix implicitly assume that all power events are of equal importance. An assumption that as argued by the authors is not a fair since

different appliances have different consumption levels and consequently more or less weight in the final energy estimation. Table 6.12 below summarizes the domain specific metrics that will be under study in this work.

Table 6.12 – Summary of domain specific metrics for event detection algorithms

Metric	<i>Symbol</i>	Best	Worst
Total Power Change – False Positives	TPC_{FP}	0	*
Total Power Change – False Negatives	TPC_{FN}	0	*
Average Power Change – False Positive	APC_{FP}	0	*
Average Power Change – False Negative	APC_{FN}	0	*
TPC-FP – TPC-FN Distance to Perfect	DTP_{TPC}	0	**
APC-FP – APC-FN Distance to Perfect	DTP_{APC}	0	**

* The worst result is proportional to the number of events and size of the erroneous events; thus it is not possible to define a fixed lower bound

** Since we cannot define a fixed lower bound to the individual metrics it is also not possible to set a lower bound to the DTP metric.

6.3.2 Event Classification

In NILM the classification task is a multi-class problem, i.e., each power event can be classified into more than two different appliances. As such, most of the performance metrics available for this kind of problems were adapted from their binary classification counterparts that we have just described.

Multi-class classification metrics can be calculated over the entire class collection, which is called *micro-averaging*, or by averaging the performance of each individual class, which is called *macro-averaging*.

In micro-averaging, each class counts the same for the average, as such larger classes dominate the measure; In macro-averaging, first the average for each class is determined, and only then each class counts the same for the final average. This difference is particularly important when the collection is skewed, which is indeed the case of NILM, since in a household it is expected that some appliances will trigger much more power events than others.

Macro-average metrics are not without their own caveats. For instance, one evident issue with macro-averaging is that it doesn't consider the number of samples in each class. Hence if there are very few examples of one appliance then the metric values for that appliance will be unreliable since it will tend to have a large variance that will necessarily affect the statistical significance of the final per-class average. Consequently, it is common practice to weight the individual class metrics by the respective number of instances, thus making the final average less sensitive to smaller classes. This is known as *weighted macro-average*.

In this work we will look at the micro- and macro- versions of the confusion matrix and rank metrics that were described in the previous section. Likewise, we will also look at probabilistic metrics, that is, metrics that measure how far the predictions are from the true result. More precisely, we will investigate two of them, namely: i) Mean Absolute Error, and ii) Root Mean Squared Error.

6.3.2.1 Confusion matrix based metrics

Table 6.13 below summarizes the confusion matrix based metrics that we use to evaluate event classification algorithms.

Note that in the case of the Micro average we are not considering Precision, Recall, $F_{0.5}$ -score and F_2 -score, which happen due to the fact that the resulting confusion matrix for all the classes will have the same number of False Positives and False Negatives. Consequently, all these metrics will have the same value as F_1 .

Table 6.13 – Confusion matrix based metrics for event classification

Metric	Micro	Macro	Weighted
Precision	✗	✓	✓
Recall	✗	✓	✓
F_1 -Measure	✓	✓	✓
$F_{0.5}$ -Measure	✗	✓	✓

Metric	Micro	Macro	Weighted
F ₂ -Measure	✗	✓	✓
Accuracy	✓	✓	✓
Error rate	✓	✓	✓
DTP_{PR}	✗	✓	✓
FPR	✓	✓	✓
DTP_{Rate}	✓	✓	✓
FPP	✓	✓	✓
DTP_{Perc}	✓	✓	✓
SMCC	✓	✓	✓

6.3.2.2 Rank / ordering metrics

Table 6.14 below summarizes the rank metrics that are used to evaluate event classification algorithms. Note that we are not using the *BAUC* metric, which happens due to the fact that we do not have a majority class in any of the eleven datasets for event classification.

Table 6.14 – Summary of rank metrics for event classification

Metric	Micro	Macro	Weighted
<i>WAUC</i>	✓	✓	✓
<i>GAUC</i>	✓	✓	✓
<i>WAUCB</i>	✓	✓	✓

6.3.2.3 Probabilistic metrics

Table 6.15 below summarizes the two probabilistic measures that are used in the performance evaluation of classification algorithms.

Table 6.15 – Summary of probabilistic metrics for event classification

Metric	Symbol	Best	Worst
Mean Absolute Error	<i>MAE</i>	0	*
Root Mean Squared Error	<i>RMSE</i>	0	*
* The worst result is proportional to the number of miss-classifications and the distance to the correct classification, thus it is not possible to define a fixed lower bound			

6.4 Experimental Design

In this section we thoroughly describe our experimental design. We start with a general overview of the research method and then provide details for the two individual problems that we are studying.

6.4.1 Research Methodology

Here we provide a general overview of the research methodology that is followed. So state more concretely, we describe how the experimental data for both problems is generated (training, testing and evaluation data), processed (pairwise correlations, average correlation matrices and hierarchical clustering) and analyzed.

6.4.1.1 Algorithms training and testing

In order to gain deeper insights on the nature and structure of the data that is generated by event detection and event classification algorithms we first perform a parameter sweep on the

selected event detection algorithms and a parameter and feature sweeps on the selected classification algorithms.

A parameter sweep refers to a controlled variation of a number of parameters in a particular algorithm (i.e., structural changes) and provides insights into how the different parameters affect the final results. As for the feature sweep, it refers to a controlled variation of the learning features of a particular classification algorithm and it is aimed at providing insights into how the different learning features affect the classification results. Detailed descriptions of the parameter and feature sweeps are provided in subsections 6.4.2 and 6.4.3.

6.4.1.2 Algorithm evaluation

In this step we compute the performance metrics for each of the models that result from the parameter and feature sweeps. To do this, we first count the true positives, false positives, true negatives and false negatives (i.e., the contingency matrix) for each of the tested models. Then, the resulting contingency matrices are used to calculate the performance metrics described in section 6.3.

Concerning the calculation of the contingency matrices, we should remark that the process is distinct for event detection and classification models. Regarding the former, this is done by comparing the events triggered by each model to the true events in the corresponding dataset (i.e., ground-truth). As for the later, we use the *one-vs-all* approach for multi-class classification problems [147]. Detailed explanations of the contingency matrices calculation are provided in sub-sections 6.4.2 and 6.4.3.

6.4.1.3 Computation of metrics pairwise correlations

As it was mentioned in Chapter 3, our goal is to analyze how different metrics compare to each other when applied to event detection and classification algorithms. To accomplish this, we need to compute the linear (Pearson) and rank (Spearman) pairwise correlations between the performance metrics that are used to evaluate the different models.

At this stage it is important to note that unlike the Pearson correlation coefficient that is based on the raw data, the Spearman correlation coefficient is based on the ranked values of

each performance metric. In this work, we decided to calculate the ranks following the *competition ranking strategy*, i.e., metrics that are equal receive the same ranking number and a gap is left in the ranking numbers, thus guarantying that ties won't modify the ranks given to the remaining metrics.

The selection of this ranking strategy is particularly relevant because we are trying to assess the consistency of the ranks across models and datasets. As such, it is important that the range of the rankings (worst to best) remains consistent independently of ties that may happen. This would not happen if, for example, we had chosen the *dense ranking strategy* in which the next element always receives the immediately following ranking number independently of ties.

6.4.1.4 Computation of average correlation matrices

In this step we compute the average correlation matrices for each problem. This process is described below.

Considering that for one particular algorithm X metrics are calculated, this allows $X * (X - 1) / 2$ unique pairwise correlations per correlation coefficient. Thus, in our particular case there are 812, 342 and 552 possible pairwise correlations for event detection metrics, micro and macro classification metrics, respectively, as summarized in Table 6.16.

Table 6.16 – List of possible pairwise correlations per metric types

	Metrics*	Pairwise Correlations	Total**
Event Detection	27	351	702
Event Classification - Micro	19	171	342
Event Classification - Macro	22	231	462

* Here we also consider the counts of TP, FP, TN and FN as individual metrics

** Each pairwise correlation is computed for two coefficients, namely the Pearson correlation coefficient (linear) and the Spearman correlation coefficient (non linear)

Regarding the event detection algorithms, each model is evaluated ten times in each of the four datasets, meaning that there are 200 different correlation matrices for each coefficient ($10 \text{ tolerance values} * 5 \text{ algorithms} * 4 \text{ datasets}$). These 200 matrices are averaged (arithmetically) based on the tolerance value, forming 20 new correlation matrices. Then, in order to evaluate the correlations on each individual dataset, these matrices are averaged by algorithm to create four matrices. Finally, a cross-dataset correlation matrix is computed by averaging the resulting correlation matrices.

As for the event classification algorithms, every produced model is evaluated once against each of the eleven datasets, meaning that there are 66 different correlations matrices per coefficient ($6 \text{ algorithms} * 11 \text{ datasets}$). Then, in order to evaluate the correlations of each individual dataset the 66 matrices are aggregated by algorithm (leading to a total of 11 matrices). Finally, in order to explore the performance metrics across the different algorithms and datasets, we produce one final cross-dataset correlation matrix by averaging the correlation matrices for each individual dataset.

Note that, under no circumstances we merge the evaluation results obtained from each of the different model-dataset pairs. Instead, we merge only the pairwise metric correlations. The reason for this is the fact that there is evidence that event detection and classification algorithms depend heavily on the datasets [79]. Thus, producing cross-datasets averages can lead to biased conclusions since it is possible that good results in one dataset compensate for poor results in other datasets and vice-versa.

6.4.1.5 Hierarchical clustering

In this step we build clusters from the resulting average pairwise correlation matrices using hierarchical clustering. To do so we first define the *dissimilarity function*, i.e., a function that defines the distance between two clusters (or metrics), which in this particular case depends only on the pairwise correlation coefficients. Then, we define the *linkage function* that is used to join (i.e., cluster) the different pairs of metrics and clusters.

Regarding the former, in this work we use the dissimilarity function that is defined in equation (6.8), where D is the distance and $|C|$ is the absolute value of the correlation between the clusters.

$$D = 1 - |C| \quad (6.8)$$

This dissimilarity measure is known to discriminate well between all correlated pairs, independently of the direction since the pairs with "stronger" correlation are ordered correctly from the bottom ($|C|=1.0$) to the top ($|C| = 0.0$). Hence making this measure more suitable for graphical representation using dendrograms.

As for the linkage distance, we use the *average-group distance*, which joins an existing group to the element (or group) whose average distance to the group is minimum. This method is also known as **Un-weighted Pair Group Method with Arithmetic Mean (UPGMA)** and the distance between two groups A and B is given by equation (6.9).

$$D_{AB} = \frac{1}{|A||B|} \sum_{a \in A} \sum_{b \in B} d(a, b) \quad (6.9)$$

Where d is a distance function (in our case the Euclidean distance) and $|A|$ and $|B|$ are the size of groups A and B , respectively.

6.4.1.6 Comparison strategy

The comparative analysis is done separately for each problem, and it is based on observations taken from three different representations of the pairwise correlations: i) the raw pairwise correlations, ii) the average pairwise correlations between all the performance metrics, and iii) the hierarchical clustering of the correlation results.

To do this, we first perform a more general analysis of the correlation results. Then, on a second stage we provide a more in-depth analysis by looking at specific metrics or pairs of metrics. Lastly, we explore the possibility of creating groups of metrics (i.e., clusters) in the two learning problems. To do so we will investigate how cutting the resulting dendrograms at different distances will affect the grouping of the performance metrics.

6.4.2 Event Detection Algorithms

Here we provide additional details about how the data for evaluating event detection algorithms is generated. More precisely, we thoroughly describe the parameter sweep that is applied to each algorithm and the method that is used to evaluate the generated models.

6.4.2.1 Parameter sweep

Regarding the event detectors parameter sweep we decided to keep some common parameters across the different algorithm unchanged, namely the power metric (M_{pwr}) and respective threshold (P_{thr}).

More precisely, we have set the power metric to real power since it is probably the most widely used metric in event detection literature. We also set the power threshold to 30 Watts, because it is the minimum power change for which there are labeled events in any of the four datasets. Next we present the parameters that were switched in each algorithm.

6.4.2.1.1 Meehan Expert Heuristic

In the MEH we decided to switch all the remaining parameters using the ranges presented in Table 6.17 below:

Table 6.17 – Parameter ranges for Meehan Expert Heuristic event detector

Parameter	Min	Max	Increment
G_0	0	5	1 (Seconds)
w_0	1	5	1 (Seconds)
w_l	1	5	1 (Seconds)
T_{elap}	0	5	0.5 (Seconds)
E_{edge}	1, 0.5 F_s , F_s		

The pre- and post-event windows (w_0 and w_l) vary between one and five seconds with one-second intervals, i.e., for each second in the pre-event window there are five different

sizes for the post-event window. Regarding the pre-event gap (G_0), it varies between zero and five seconds with one-second intervals.

Likewise, the elapsed time parameter (T_{elap}) varies between zero and five seconds, yet this was done with half-a-second intervals since this parameter can have a heavy impact on the final detection results (e.g., when set to zero all changes above the 30 Watts threshold will be considered power events, but when set to five only events that are detected five seconds apart will be triggered).

Lastly, we have decided that for each detected power event there were three possible values for the event edge (E_{edge}), namely: i) the first sample of the second, ii) the half-a-second sample, and iii) the last sample in the second.

Overall, with this parameter range there are a total of 4950 instances of this algorithm ($5 w_0 * 5 w_I * 6 G_0 * 11 T_{elap} * 3 E_{edge}$) each of which was executed against each of the four labeled datasets described in section 6.1.2.3.1.

6.4.2.1.2 Log-Likelihood Detector and Simplified Log-Likelihood Detector with voting activation

As mentioned previously, we will evaluate the two log-likelihood detectors using both detection activation algorithms. As such, we will first execute the LLD and SLLD with *voting activation* according to the parameter ranges defined in Table 6.18 below:

Table 6.18 – Parameter ranges for Log-Likelihood and Simplified Log-Likelihood detectors with *voting activation*.

Parameter	Min	Max	Increment
w_0	0,5	5	0,5 (Seconds)
w_I	0,5	5	0,5 (Seconds)
w_v	0,5	5	0,5 (Seconds)
V_{thr}	5	*	15 (Votes)

Here the pre-, post-event and voting windows (w_o , w_l) vary between half-a-second and five seconds with half-a-second intervals, i.e., for each half-a-second in the pre-event window there are ten different sizes for the post-event window. For each combination of pre- and post-event there are 10 possible voting windows, in a total of 1000 different combinations ($w_o * w_l * w_v$). Lastly, the voting threshold (V_{thr}) was set such that it is never larger than the voting window; otherwise no events would be triggered. Therefore, we set this parameter to a minimum of five votes that is incremented in intervals of fifteen votes up to a maximum that is never larger than the voting window size.

Regarding the total number of tests that will result from this parameter sweep, it is important to note that it will vary with the sampling rate of the dataset being considered (50 Hz and 60 Hz in our case). For example, in a 60 Hz dataset at each half-a-second increment it will always be possible to increment the voting threshold by fifteen samples two times, which is not always true for 50 Hz datasets. Overall, after the calculations are made, this parameter sweep will result in eleven thousand (11000) tests for the 60 Hz datasets and nine thousand five hundred (9500) when 50 Hz datasets are used.

6.4.2.1.3 Log-Likelihood Detector and Simplified Log-Likelihood Detector with maxima activation

Here we will evaluate the LLD and SLLD with maxima voting activation. This will be done according to the parameter ranges defined in Table 6.19 below:

Table 6.19 – Parameter ranges for Log-Likelihood and Simplified Log-Likelihood detectors with *maxima* activation.

Parameter	Min	Max	Increment
w_o	0,5	5	0.5 (Seconds)
w_l	0.5	5	0.5 (Seconds)
M_{pre}	0.5	5	0.5 (Seconds)

As one can see, we will use the same window sizes for pre- and post-event windows. As for the *maxima* precision we decided to use a wide range of values for this parameter ranging

from half-a-second to five seconds with half-a-second intervals. Overall, this parameter sweep will result in 1000 possible event detectors ($10 w_{0,n} * 10 w_{I,n} * 10 M_{pre}$).

In summary, executing all the above-mentioned models (each parameter combination of a different algorithm is considered a model) on the four datasets gives a total of 109800 different model-dataset pairs as shown in Table 6.20.

Table 6.20 – Number of different models that will be evaluated across datasets

Algorithm	Individual Models	Model-Dataset Pairs
MEH	4950	19800
SLLD + Maxima	1000	4000
SLLD + Voting	11000 + 9500	22000 + 19000
LLD + Maxima	1000	4000
LLD + Voting	11000 + 9500	22000 + 19000
---	47950	109800

6.4.2.2 Evaluation

In order to benchmark the different metrics, we first have to build the confusion matrix that result from the execution of each algorithm configuration. To accomplish this, we need to define a tolerance interval in which the detected events must fall in order to be considered correct detections. The detection interval is defined by equation (6.10) and is based on a tolerance value that was added to account for certain ambiguity in defining exactly where an event occurs when labeling a dataset [148].

$$\psi = [\text{ground truth position} - \text{tolerance}, \text{ground truth position} + \text{tolerance}] \quad (6.10)$$

This parameter is particularly important in our case since we are working with datasets at line frequency (50 Hz and 60 Hz), which necessarily increases the labeling ambiguity. In previous work on this topic [94] the authors varied this parameter from one to six seconds (in one-second steps) and found that no improvements were observed with more than three

seconds of tolerance. Consequently, we decided to set this parameter to range between zero and three seconds with variable steps as shown in Table 6.21. F_s represents the sampling frequency of the dataset.

Table 6.21 – Tolerance values for event detection evaluation

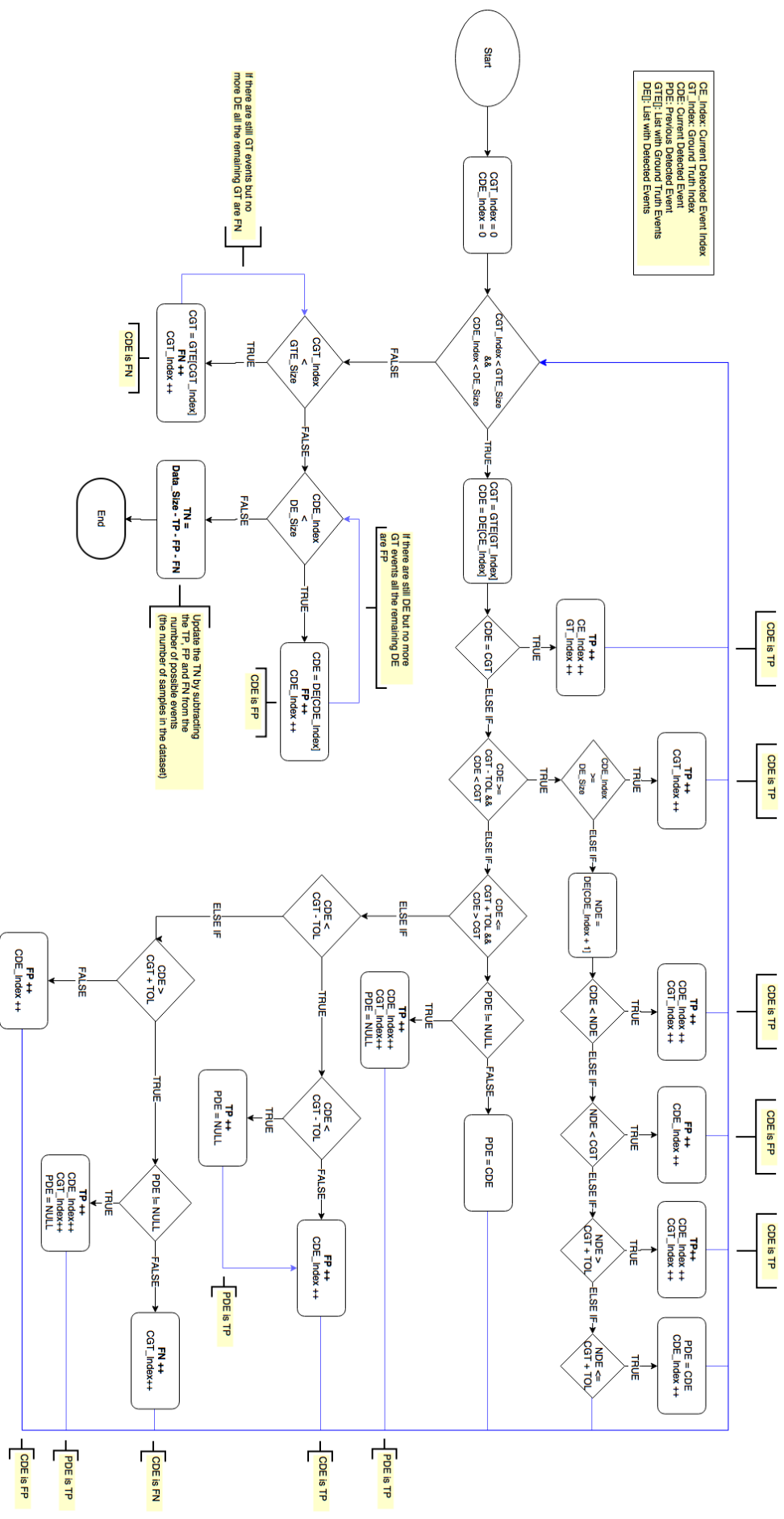
Tolerance	Description
0	Zero samples
1	One sample
5	Five samples
15	Fifteen samples
$0.5 F_s$	Half of the sampling frequency (25 samples for 50 Hz or 30 for 60 Hz)
F_s	Sampling rate (either 50 or 60 samples)
$1.5 F_s$	One-and-a-half times the sampling frequency (75 or 90 samples)
$2 F_s$	Twice the sampling frequency (100 or 120 samples)
$2.5 F_s$	Two-and-a-half times the sampling frequency (125 or 150 samples)
$3 F_s$	Three times the sampling frequency (150 or 180 samples)

Regarding the process of creating the confusion matrix, we had to develop an algorithm that follows the logic presented in Figure 6.4. Given a list of detected events and another with the ground-truth data, the algorithm works as follows:

For each ground-truth event, if there are detections that fall within the interval ψ given by equation (6.10), the event that is closer to the ground-truth position (in absolute distance) or the one that was detected first (in the case of equidistant detections) is considered a **True Positive**, whereas the others must be compared with the next ground-truth event.

Otherwise, if no detections happened within the specified interval, a **False Negative** is added. Likewise, detections that do not fall within any of the possible intervals ψ (one per each ground-truth event) are considered **False Positives**.

Lastly, when all the detected and ground-truth events have been processed, the ***True Negatives*** are calculated by subtracting the TP , FN and FP from the number of samples in the dataset, i.e., all the positions where an event could have happened.



6.4.3 Event Classification Algorithms

Here we provide additional details about how the data for evaluating event classification algorithms is generated. More precisely, we thoroughly describe the parameter and feature sweeps that are applied to each algorithm and the method that is used to create the necessary confusion matrices.

6.4.3.1 Parameter sweep

Regarding the parameter sweep of the event classification algorithms, we decided to switch only one parameter of each algorithm while leaving the remaining parameters set to their default values. Using this strategy we ensure that each classification algorithm is tested the same number of times, but more importantly, we assure that changes in the obtained results are fully justified by one single parameter and the set of learning features.

In Table 6.22 we list the parameter that will be switched in each algorithm. Further details are provided in the following subsections.

Table 6.22 – List of the parameters that will be switched in each classification algorithm and respective values

Algorithm	Parameter	Symbol	Values
K-Nearest Neighbor	Number of neighbors	K	$1, 3, \sqrt{\frac{N}{2}}, N$
KStar	Blending parameter	b	1, 20, 50, 75, 100
LWL with Naïve Bayes	Number of neighbors	K	$\frac{N}{2}, \sqrt{\frac{N}{2}}, N$
Decision Trees	Min. number of instances per leaf	$minNumObjs$	1, 2, 5, 10, 15
Artificial Neural Networks	Learning rate	$learningRate$	0.1, 0.2, 0.3, 0.4, 0.5
Support Vector Machines	Cost parameter	C	0.01, 0.1, 1, 10, 100

6.4.3.1.1 K-Nearest Neighbor

In the K-NN classifier we decided to vary only the number of neighbors (k). More specifically, we use the following values: $1, 3, \sqrt{n_{sc}}, \frac{n_{sc}}{2}$ and n_{sc} . Where n_{sc} is the smallest number of examples from a particular class present in the dataset. For example, if in the dataset the smallest class has 50 examples, n_{sc} is set to that value.

6.4.3.1.2 KStar

Regarding the KStar algorithm, we decided to vary the blending parameter (b), which is used to define the size of the neighborhood. More concretely, the following values are used for b : 1%, 20% (default value), 50%, 75% and 100%.

6.4.3.1.3 LWL with Naïve Bayes

For the Locally Weighted Learning with a Naïve Bayes classifier we change the number of neighbors (k) that defines region where classifier will be applied.

More particularly, we vary k in a way that is very similar to the K-NN algorithm. The single exception is that the case where $k = 1$ is replaced with a value of k that is equal to the number of samples in the training data.

6.4.3.1.4 Decision Trees

Concerning the decision trees, we have decided to tweak the minimum number of instances per leaf parameter (*minNumObj*) of a pruned J48 decision tree. The *minNumObj* was tweaked using the following set of values: 1, 2 (default), 5, 10 and 15.

6.4.3.1.5 Artificial Neural Networks

In the case of the ANN algorithm we decided to tweak the learning rate parameter (*learningRate*) of a two-layer network with n hidden neurons, where n is the number of

classes in the dataset (11 in the case of PLAID). The *learningRate* was set to take the following values: 0.1, 0.2, 0.3 (default), 0.4 and 0.5.

6.4.3.1.6 Support Vector Machines

Lastly, concerning the SVM algorithms we have decided to tweak the cost parameter (C) of an SVM with and RBF kernel with γ set to its default value. The C parameter was set to take the following values: 0.01, 0.1, 1, 10, and 100.

6.4.3.2 Feature sweep

Regarding the feature sweep we have decided to evaluate each classification algorithm using 30 feature sets. The feature sets are presented in Table 6.23 and were created from thirteen learning features presented in 6.1.2.3 (Learning Features).

We refer to the features sets from 1 to 12 as *single-feature* since they either contain a single feature (sets 3 to 5 and 7 to 11) or combine features of the same type (sets 1, 2, 6 and 12). The remaining 18 sets are referred to as *multi-feature* since they combine features from different types.

Naturally we did not attempt each possible combination of features, since: i) with thirteen individual features there will be a combinatorial explosion of the possible features sets, hence making this task extremely time consuming, and, ii) some of the features contain similar information (e.g., raw and quantized waveforms), which could easily result in over-fitting. Instead, we decided to choose some feature combinations that are complementary. For example, the feature sets 13 to 16 combine delta features with harmonic and waveforms features. Lastly, it is important to remark that the multi-feature sets are scaled before feeding the learning algorithms, thus avoiding that learning features with higher values have more preeminence in the final results.

Table 6.23 – Different feature sets used in the event classification algorithms

Category	Features
Delta	1. P, Q 2. P, Q, I
Harmonics	3. $H_{I,n}$ 4. $H_{IV,n}$
Raw Waveforms	5. I_{WF}
Quantized Waveforms [100]	6. I_{QWF} 7. IV_{QWF}
Data Driven Features [100]	8. VI_{BIN} 9. VI_{BIN_PCA} 10. I_{WF_PCA} 11. I_{QWF_PCA} 12. IV_{QWF_PCA}
Combined	13. P, Q, $H_{I,n}$ 14. P, Q, $H_{IV,n}$ 15. P, Q, I_{WF} 16. P, Q, IV_{WF} 17. P, Q, VI_{BIN_PCA} 18. P, Q, $H_{I,n}$, I_{QWF} 19. P, Q, $H_{I,n}$, IV_{QWF} 20. P, Q, $H_{IV,n}$, I_{QWF} 21. P, Q, $H_{IV,n}$, IV_{QWF} 22. P, Q, $H_{I,n}$, VI_{BIN_PCA} 23. P, Q, $H_{IV,n}$, VI_{BIN_PCA} 24. $H_{I,n}$, VI_{BIN_PCA} 25. $H_{IV,n}$, VI_{BIN_PCA} 26. I_{QWF} , VI_{BIN_PCA} 27. IV_{QWF} , VI_{BIN_PCA} 28. I_{QWF_PCA} , VI_{BIN_PCA} 29. IV_{QWF_PCA} , VI_{BIN_PCA} 30. I_{WF} , VI_{BIN_PCA}

In summary, considering the parameter and feature sweep, there will be $5 \times 30 = 150$ different models for each of the six classification algorithms. Each algorithm will then be evaluated against the 11 datasets leading to 1650 confusion matrices per algorithm. Hence, considering all the possible combinations, in the end there will be a total of $1650 \times 6 = 9900$ matrices.

6.4.3.3 Evaluation

Regarding the evaluation procedure for event classification we have decided to split the training and testing sets by individual dataset. In other words, all the measurements from one dataset will be used as testing data while the data from the remaining ten datasets will be used to train the models. This process is repeated once for each of the 11 datasets.

By following this approach, the models will always be tested in previously unseen data, thus reducing the chance of *over-fitting* during training. Likewise, using this approach all the models are trained with a large and diverse set of examples, which we believe can help reduce classification bias.

Regarding the creation of the confusion matrix we followed the *one-vs-all* approach. To do this, one binary confusion matrix is created for each class on the training data, where one class is considered the positive class and the combination of the remaining classes make up the negative class. The resulting binary matrices are then added together to form the final *one-vs-all* confusion matrix that is used to compute micro- and macro-average metrics.

6.5 Analysis of Results

In this section we analyze the obtained results for each individual problem. To this end, we first look at the individual pairwise correlations, after which we examine possible metric clusters that may emerge from the correlation matrices.

6.5.1 Event detection

The average pairwise correlations across event detection datasets are presented in Table 6.24, showing the rank and linear correlations in the lower and upper triangles, respectively. Metrics with pairwise correlations (in absolute value) closer to one (above 0.9) appear highlighted as they are expected to behave more similarly than others.

Table 6.25 shows the average rank and linear correlations of each metric across the four datasets. Metrics with average correlations above 0.6 appear highlighted in green tones, and when above 0.7 a bold font-face is used.

Table 6.24 – Rank (bottom-left) and linear (top-right) correlation results for all four datasets

		Ranks																									
	TP	FP	TN	FN	FPP	FPR	A	E	P	R	F05	F1	F2	SMCC	DTPrp	DTPrate	DTperc	W_AUC	W_AUCB	G_AUC	B_AUC	TPC_FN	TPC_FP	DTpApc	APC_FN	APC_FP	DTpApc
TP		0.54	-0.54	-1.00	0.54	0.54	-0.46	0.46	-0.35	1.00	-0.29	-0.11	0.32	-0.01	0.02	-0.97	0.42	1.00	0.99	0.99	1.00	-0.91	0.48	0.25	-0.28	0.17	-0.22
FP	-0.65		-1.00	-0.54	1.00	1.00	-0.99	0.99	-0.80	0.54	-0.80	-0.73	-0.43	-0.66	0.63	-0.47	0.95	0.54	0.55	0.53	0.54	-0.53	0.85	0.70	-0.29	0.15	-0.21
TN	-0.65	1.00		0.54	-1.00	-1.00	0.99	-0.99	0.80	-0.54	0.80	0.73	0.43	0.66	-0.63	0.47	-0.95	-0.54	-0.55	-0.53	-0.54	0.53	-0.85	-0.70	0.29	-0.15	0.21
FN	1.00	-0.65	-0.65		-0.54	-0.54	0.46	-0.46	0.35	-1.00	0.29	0.11	-0.32	0.01	-0.02	0.97	-0.42	-1.00	-0.99	-0.99	-1.00	0.91	-0.48	-0.25	0.28	-0.17	0.22
FPP	-0.65	1.00	1.00	-0.65		1.00	-0.99	0.99	-0.80	0.54	-0.80	-0.73	-0.43	-0.66	0.63	-0.47	0.95	0.54	0.55	0.53	0.54	-0.53	0.85	0.70	-0.29	0.15	-0.21
FPR	-0.65	1.00	1.00	-0.65	-0.65		-0.99	0.99	-0.80	0.54	-0.80	-0.73	-0.43	-0.66	0.63	-0.47	0.95	0.54	0.55	0.53	0.54	-0.53	0.85	0.70	-0.29	0.15	-0.21
A	-0.46	0.92	0.92	-0.46	0.92	0.92		-1.00	0.81	-0.46	0.81	0.77	0.51	0.71	-0.67	0.40	-0.95	-0.46	-0.48	-0.45	-0.46	0.47	-0.84	-0.73	0.28	-0.15	0.20
E	-0.46	0.92	0.92	-0.46	0.92	0.92	1.00		-0.81	0.46	-0.81	-0.77	-0.51	-0.71	0.67	-0.40	0.95	0.46	0.48	0.45	0.46	-0.47	0.84	0.73	-0.28	-0.15	-0.20
P	-0.36	0.85	0.85	-0.36	0.85	0.85	0.83	0.83		-0.35	0.99	0.92	0.60	0.89	-0.86	0.29	-0.68	-0.35	-0.36	-0.34	-0.35	0.39	-0.65	-0.49	0.27	-0.11	0.22
R	1.00	-0.65	-0.65	1.00	-0.65	-0.65	-0.46	-0.46	-0.36		-0.29	-0.11	0.32	-0.01	0.02	-0.97	0.42	1.00	0.99	0.99	1.00	-0.91	0.48	0.25	-0.28	0.17	-0.22
F05	-0.27	0.79	0.79	-0.27	0.79	0.79	0.84	0.84	0.98	-0.27		0.96	0.67	0.93	-0.90	0.22	-0.69	-0.29	-0.30	-0.27	-0.28	0.32	-0.65	-0.52	0.25	-0.13	0.18
F1	-0.04	0.61	0.61	-0.04	0.61	0.61	0.76	0.76	0.85	-0.04	0.92		0.83	0.98	-0.96	0.03	-0.67	-0.10	-0.12	-0.09	-0.10	0.15	-0.61	-0.55	0.16	-0.17	0.09
F2	0.37	0.20	0.20	0.37	0.20	0.20	0.41	0.41	0.48	0.37	0.58	0.79		0.86	-0.85	-0.39	-0.45	0.32	0.31	0.33	0.32	-0.26	-0.35	-0.44	-0.04	-0.14	-0.10
SMCC	0.01	0.57	0.57	0.01	0.57	0.57	0.72	0.72	0.82	0.01	0.89	0.98	0.82		-0.98	-0.05	-0.60	-0.01	-0.03	0.00	-0.01	0.07	-0.54	-0.50	0.12	-0.17	0.04
DTPrp	0.00	0.55	0.55	0.00	0.55	0.55	0.70	0.70	0.81	0.00	0.88	0.97	0.81	0.98		0.05	0.58	0.01	0.03	0.00	0.01	-0.07	0.52	0.47	-0.11	0.21	-0.02
DTPrate	0.99	-0.65	-0.65	0.99	-0.65	-0.65	-0.45	-0.45	-0.36	0.99	-0.26	-0.04	0.38	0.02	0.01	-0.36	-0.97	-0.97	-0.97	-0.98	-0.97	0.89	-0.41	-0.18	0.25	-0.11	0.21
DTperc	-0.48	0.90	0.90	-0.48	0.90	0.90	0.98	0.98	0.81	-0.48	0.82	0.75	0.40	0.70	0.69	-0.47	0.78	0.42	0.43	0.41	0.42	-0.42	0.82	0.75	-0.24	0.14	-0.17
W_AUC	0.99	-0.65	-0.65	0.99	-0.65	-0.65	-0.45	-0.45	-0.36	0.99	-0.26	-0.04	0.38	0.02	0.01	1.00	-0.47		0.99	0.99	1.00	-0.91	0.48	0.25	-0.28	0.17	-0.22
W_AUCB	0.99	-0.66	-0.66	0.99	-0.66	-0.66	-0.47	-0.47	-0.37	0.99	-0.27	-0.05	0.37	0.01	0.00	-0.48		0.99		0.99	1.00	-0.91	0.49	0.26	-0.28	0.18	-0.22
G_AUC	0.99	-0.65	-0.65	0.99	-0.65	-0.65	-0.45	-0.45	-0.36	0.99	-0.26	-0.04	0.38	0.02	0.01	0.99	-0.47	0.99	0.99		0.99	-0.91	0.47	0.24	-0.27	0.17	-0.21
B_AUC	0.99	-0.65	-0.65	0.99	-0.65	-0.65	-0.45	-0.45	-0.36	0.99	-0.26	-0.04	0.38	0.02	0.01	1.00	-0.47	1.00	0.99	0.99		-0.91	0.48	0.25	-0.28	0.17	-0.22
TPC_FN	0.92	-0.70	-0.70	0.92	-0.70	-0.70	-0.53	-0.53	-0.43	0.92	-0.35	-0.14	0.28	-0.07	-0.09	0.92	-0.55	0.92	0.92	0.92	0.92	-0.48	-0.23	0.48	-0.16	0.36	0.66
TPC_FP	-0.63	0.86	0.86	-0.63	0.86	0.86	0.79	0.79	0.73	-0.63	0.69	0.53	0.15	0.49	0.48	-0.63	0.78	-0.63	-0.64	-0.63	-0.63	-0.66	0.39	0.88	-0.30	0.48	-0.07
DTpApc	0.03	0.26	0.26	0.03	0.26	0.26	0.42	0.42	0.27	0.03	0.33	0.44	0.50	0.44	0.41	0.03	0.42	0.03	0.03	0.03	0.03	0.06	0.39	-0.14	-0.10	0.39	0.04
APC_FN	0.23	-0.34	-0.34	0.23	-0.34	-0.34	-0.35	-0.35	-0.27	0.23	-0.26	-0.16	0.02	-0.13	-0.12	0.23	-0.33	0.23	0.23	0.23	0.23	0.43	-0.31	-0.07	-0.09	-0.10	0.76
APC_FP	-0.16	-0.04	-0.04	-0.16	-0.04	-0.04	-0.02	-0.02	-0.04	-0.16	-0.02	0.02	0.01	0.01	0.05	-0.17	0.00	-0.17	-0.16	-0.17	-0.17	-0.15	0.31	0.38	-0.09	-0.10	0.32
DTpApc	0.16	-0.29	-0.29	0.16	-0.29	-0.29	-0.28	-0.28	-0.24	0.16	-0.22	-0.12	0.05	-0.08	-0.07	0.16	-0.27	0.16	0.16	0.16	0.16	0.29	-0.11	0.13	0.76	0.25	

When examining the correlation results shown in Table 6.24, a first interesting observation is that the Average Power Change (APC) metrics do not correlate well with any of the other metrics. Furthermore, if we recall that APC_{FP} and APC_{FN} are just the TPC_{FP} and TPC_{FN} metrics normalized by the number of events it is possible to conclude that APC metrics will evidence strong variations depending of the number and size of the power events in the dataset. For example, if event detector **A** fails to detect (false negatives) all the power events bellow 50 Watts but only fails to detect one event of 100 Watts, it will still have an APC_{FN} of about 50 Watts. On the other hand, an event detector (**B**) that only misses one event with 100 Watts will have an APC_{FN} of 100.

Another general observation concerns to the relatively strong correlation (> 0.5 in absolute value) between most of the other metrics in Table 6.25. The only exceptions to this trend are F_1 , F_2 , DTP_{PR} and $SMCC$ than have an average correlation of only 0.46. This is particularly interesting since three out of the four metrics were designed to balance Precision and Recall (F_1 , F_2 and DTP_{PR}), and still they do not correlate well with their “parent” metrics. For example, the F_2 metric does not have any pairwise correlation above 0.65, and perhaps even more surprising, it does not correlate at all (< 0.5) with either P or R .

A more specific observation concerns to the very strong (0.92) pairwise rank and linear correlations between TPC_{FN} , Recall (R) and all the four rank metrics (AUC). A possible explanation for this is that most missed power events (false negatives) have similar delta values (possibly near te minimum power threshold), hence the strong linear and non-linear correlations. Similarly, if we consider the TPC_{FP} , it is possible to observe some correlation (0.63) in the non-linear coefficient that is not followed by a linear correlation. Hence, it is expected that some TPC_{FP} ranks will be relatively close to those obtained with the other metrics, in particular those that are derived from the False Positives. Still, the lack of a strong linear correlation is a good indicator that the delta values of the FP are heavily dependent on the dataset characteristics.

The results in Table 6.24 also reveal that all the four rank-based metrics are very well correlated between themselves (0.99), as well as with Recal (0.99). However, this is just a reflection of the fact that specificity (or True Negative Rate) is always close to one since the number of true negatives is much higher than the number of false positives. Consequently, AUC metrics are only reflecting variations on the Recall, meaning that the selected models will be the selected based on that metrics only.

Moreover, it is possible to observe that the DTP_{Rate} metric is also very well correlated with the rank-based metrics in both coefficients. In this case this is a reflection of the fact that the FPR is always close to zero, meaning that the metrics is fully controled by Recall.

Overall, it is possible to find 44 metric pairs where at least one of the coefficients is above 0.9 in absolute value. These are summarized in Figure 6.5, where it is possible to quickly identify three groups, covering 15 of the 20 studied metrics (without considering the base metrics and three redundant AUC metrics):

1. $R, WAUC, TPC_{FN}, DTP_{Rate}$
2. $FPR, FPP, DTP_{Perc}, A, E$
3. $P, F_{0.5}, F_1, SMCC, DTP_{PR}$

Additionally, it is possible to observe that the metrics in the first group are all correlated with the TP and FN whereas the metrics in the second group show strong correlations with the TN and FP .

		Rank	Linear
TP	FN	1,00	-1,00
TP	R	1,00	1,00
TP	DTP_Rate	0,99	-0,97
TP	W_AUC	0,99	1,00
TP	TPC_FN	0,92	-0,91
FP	TN	1,00	-1,00
FP	FPR	1,00	1,00
FP	FPP	1,00	1,00
FP	DTP_Perc	0,90	0,95
FP	A	0,92	-0,99
FP	E	0,92	0,99
TN	FPR	1,00	-1,00
TN	FPP	1,00	-1,00
TN	DTP_Perc	0,90	0,95
TN	A	0,92	-0,99
TN	E	0,92	-0,99
FN	R	1,00	-1,00
FN	DTP_Rate	0,99	0,97
FN	W_AUC	0,99	-1,00
FN	TPC_FN	0,92	0,91
Average		0,96	0,98

		Rank	Linear
R	W_AUC	0,99	-1,00
R	TPC_FN	0,92	0,91
R	DTP_Rate	0,99	-0,97
DTP_Rate	W_AUC	1,00	-0,97
DTP_Rate	TPC_FN	0,92	0,89
WAUC	TPC_FN	0,92	-0,91
FPR	FPP	1,00	1,00
FPR	DTP_Perc	0,91	0,95
FPR	A	0,92	-0,99
FPR	E	0,92	0,99
FPP	DTP_Perc	0,91	0,95
FPP	A	0,92	-0,99
FPP	E	0,92	0,99
DTP_Perc	A	0,98	-0,95
DTP_Perc	E	0,98	0,95
A	E	1,00	-1,00
P	F05	0,98	0,99
P	F1	0,85	0,92
SMCC	F1	0,98	0,98
F05	F1	0,92	0,96
F05	DTP_PR	0,88	-0,90
F05	SMCC	0,89	0,93
DTP_PR	SMCC	0,99	-0,98
DTP_PR	F1	0,97	-0,96
Average		0,94	0,96

Figure 6.5 – List of metric pairs with pairwise correlations above 0.9 in at least of one the coefficients

In order to further understand the different possible metric arrangements, we performed clustering analysis to the correlation values. Figure 6.6 shows the dendrograms obtained from the ranks (non-linear) and linear pairwise correlations.

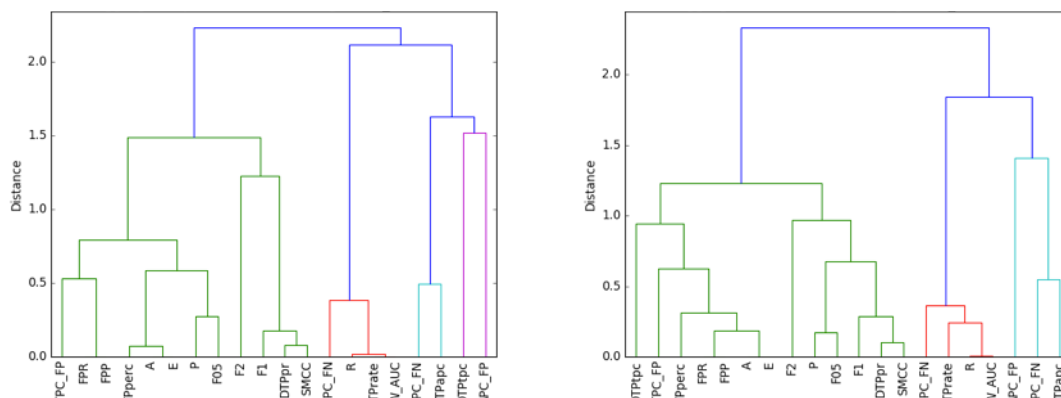


Figure 6.6 – Dendrograms showing ranks (left) and linear (right) correlations of the performance metrics across datasets

The resulting dendrograms were then split at different cut-off values (i.e., distances) to determine the number of clusters that the metrics would form. More precisely, after visual inspection, we ended up cutting the dendrograms at 0.05 and 0.1. The obtained clusters are listed in Table 6.26 below.

Table 6.26 – Clusters formed after cutting the dendrograms of the cross dataset non-linear and linear correlations

Dist.	Rank	Linear
0.05	1. $R, WAUC, DTP_{Rate}$	1. $R, WAUC$
	2. FPP, FPR	2. FPP, FPR
	3. A, E	3. A, E
0.1	1. $R, WAUC, DTP_{Rate}$	1. $R, WAUC$
	2. A, E, DTP_{Perc}	2. A, E
	3. FPP, FPR	3. FPP, FPR
	4. $SMCC, DTP_{PR}$	

A first general observation is the fact that linear and non-linear correlations return very similar clusters. In fact, the only difference is the absence of DTP metrics in the linear correlation clusters, which is not necessarily surprising given the quadratic nature of such metrics. Consequently, in the following discussion we only consider the clusters obtained from the non-linear correlations.

This being said, a first specific observation is that with a cut-off distance of 0.05 only 8 out of 20 metrics will belong to a cluster. More precisely, only metrics with pairwise correlation of at least 0.99 get clustered together. Correspondingly, with a cut-off distance of 0.1 only metrics with pairwise correlation of at least 0.985 get clustered, and so on until all the metrics are clustered at a maximum distance around 2.5.

For example, with a cut-off distance slightly below 0.25 F_1 joins $SMCC$ and DTP_{PR} in cluster 4, and with a cut-off distance of about 0.3 P and $F_{0.5}$ are joined in a 5th cluster. Lastly, it is important to remark that F_2 remains isolated until very late in the clustering process and that the same happens with all the domain specific metrics with the exception of the TPC_{FN} metric.

To conclude this section of event detection algorithms, we also provide a more in-depth look at the metrics that balance *Precision* and *Recall* ($F_{0.5}$, F_1 , F_2 and DTP_{PR}), which as we have seen above, do not have a standard behavior when applied to event detection problems.

To this end, in Figure 6.7 and Figure 6.8 we simultaneously plot the number of events detected by the different models and the value of the metrics in each case (line series). The tests are ordered from the least sensitive to the most sensitive model (i.e., ascending order of detected events). Finally, we also highlight the Top 10 models according to each metric using column series, where the height represents the rank of the model (in descending order).

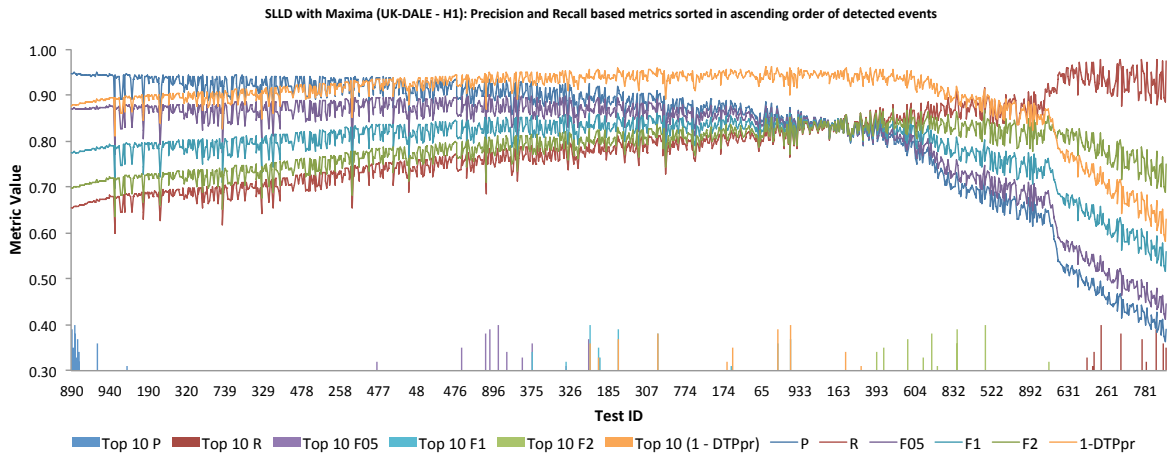


Figure 6.7 – $SLLD_{Max}$ (UK-DALE – H1): Precision and Recall based metrics sorted in ascending order of detected events (line series). Top 10 models selected by the each metric (column series).

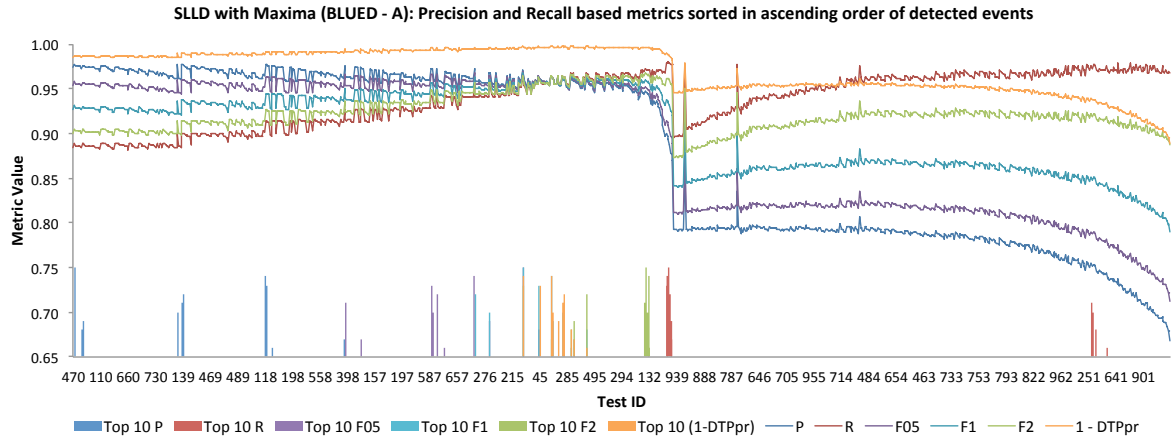


Figure 6.8 – SLLD_{Max} (BLUED – A): Precision and Recall based metrics sorted in ascending order of detected events (line series). Top 10 models selected by the each metric (column series).

As it can be observed, the models selected by Precision and Recall tend to appear in opposite corners. In other words, higher true positive counts come at the expense of a high number of false positives, whereas the lower false positive counts come at the expense of lower true positive counts. Hence, if the objective is to select the model that does the best possible job in finding power events independently of the number of false positives (i.e., a liberal model) the Recall metric should be selected. On the other hand, if the goal is to avoid to the maximum the number of erroneously detected events (i.e., a conservative model), Precision should be used.

As for the “balance metrics”, the two examples show a clear propensity to select models that are neither closer to the models selected by Precision or Recall alone, which we believe helps explain the low correlation values with their “parent” metrics. Furthermore, this is also an indicator of the “virtually unbounded” nature of the event detection problem, i.e., the positive cases (actual events) are in a much lower number when compared to what can possibly go wrong (missed and false detections).

Still, and despite the low pairwise correlation values, it is possible to see from in Figure 6.7 and Figure 6.8, that the models selected by $F_{0.5}$ and F_2 show a slight tendency to select models where the more prevalent parent metric prevails, i.e., $F_{0.5}$ selects models where P is higher, whereas F_2 selects the models with higher R . Hence, these two metrics should be used

whenever the goal is to select a model that maximizes one of the metrics without inflicting too much “damage” in the other.

Regarding F_1 , it is possible to see a clear tendency towards selecting the models where P is considerable higher than R , i.e., F_1 favors models with less false detections. Lastly, in the case of the DTP_{PR} metric, the tendency seems to be more towards selecting models where P and R are closer to each other, thus making this an interesting metric for situations where the goal is to select the algorithms with the best tradeoff between correct and erroneous detections.

6.5.2 Event classification

Regarding the performance metrics for event classification we look at the micro-, unweight macro- and weighted macro-averages separately and compare them with the remaining metrics.

The resulting correlation matrices are presented in Table 6.27, Table 6.28, and Table 6.29, showing the rank and linear correlations in the lower and upper triangle, respectively. Metrics with pairwise correlations (in absolute value) closer to one (above 0.9) appear highlighted as they are expected to behave more similarly than others. Table 6.30 shows the average rank and linear correlations between all the performance metric.

Table 6.28 – Unweighted macro average metrics: rank (bottom-left) and linear (top-right) correlation results for all datasets

	Linear																							
	TP	FP	TN	FN	P	R	A	E	FPR	FPP	SMCC	F1	F05	F2	DTPpr	DTPperc	DTPrate	WAUC	GAUC	WAUCB	MAE	RMSE		
TP		-0.99	0.93	-0.99	0.88	0.94	0.97	-0.97	-0.86	-0.63	0.94	0.92	0.90	0.94	-0.90	-0.70	-0.91	0.95	0.92	0.95	-0.95	-0.91		
FP	0.99		-0.91	1.00	-0.88	-0.94	-0.97	0.97	0.87	0.63	-0.94	-0.92	-0.90	-0.94	0.90	0.70	0.91	-0.95	-0.92	-0.95	0.95	0.91		
TN	0.99	0.99		-0.91	0.80	0.87	0.90	-0.90	-0.79	-0.57	0.87	0.85	0.83	0.87	-0.82	-0.63	-0.84	0.88	0.84	0.88	-0.88	-0.85		
FN	0.99	1.00	0.99		-0.88	-0.94	-0.97	0.97	0.87	0.63	-0.94	-0.92	-0.90	-0.94	0.90	0.70	0.91	-0.95	-0.92	-0.95	0.95	0.91		
P	0.86	0.86	0.86	0.86		0.93	0.88	-0.88	-0.82	-0.57	0.97	0.98	0.99	0.95	-0.97	-0.65	-0.94	0.93	0.95	0.91	-0.87	-0.79		
R	0.93	0.93	0.93	0.93	0.91		0.92	-0.92	-0.83	-0.56	0.98	0.98	0.95	0.99	-0.96	-0.66	-0.97	0.99	0.98	0.99	-0.93	-0.86		
A	0.97	0.97	0.97	0.97	0.87	0.91		-1.00	-0.94	-0.69	0.94	0.92	0.90	0.92	-0.89	-0.73	-0.90	0.94	0.90	0.94	-0.94	-0.89		
E	0.97	0.97	0.97	0.97	0.87	0.91	0.99		0.94	0.69	-0.94	-0.92	-0.90	-0.92	0.89	0.73	0.90	-0.94	-0.90	-0.94	0.94	0.89		
FPR	0.85	0.86	0.85	0.86	0.80	0.81	0.92	0.92		0.72	-0.86	-0.84	-0.83	-0.84	0.81	0.69	0.82	-0.86	-0.82	-0.85	0.85	0.80		
FPP	0.64	0.64	0.63	0.64	0.58	0.57	0.70	0.70	0.76		-0.61	-0.58	-0.57	-0.57	0.55	0.84	0.54	-0.58	-0.54	-0.58	0.58	0.60		
SMCC	0.93	0.93	0.93	0.93	0.96	0.98	0.93	0.93	0.85	0.61		0.99	0.98	0.99	-0.97	-0.68	-0.97	0.98	0.98	0.97	-0.93	-0.86		
F1	0.91	0.91	0.91	0.91	0.97	0.97	0.91	0.91	0.82	0.59	0.99		0.99	0.99	-0.98	-0.67	-0.98	0.98	0.98	0.96	-0.91	-0.83		
F05	0.88	0.89	0.88	0.89	0.99	0.94	0.89	0.89	0.81	0.58	0.98	0.99		0.97	-0.98	-0.66	-0.96	0.95	0.97	0.93	-0.89	-0.81		
F2	0.93	0.93	0.93	0.93	0.94	0.99	0.92	0.92	0.82	0.58	0.99	0.99	0.96		-0.97	-0.67	-0.98	0.99	0.99	0.98	-0.92	-0.85		
DTPpr	0.88	0.88	0.88	0.88	0.97	0.95	0.87	0.87	0.79	0.56	0.97	0.98	0.98	0.97		0.67	0.98	-0.96	-0.99	-0.94	0.89	0.79		
DTPperc	0.78	0.78	0.78	0.78	0.73	0.76	0.82	0.82	0.77	0.86	0.78	0.76	0.75	0.76	0.76		0.67	-0.67	-0.67	-0.67	0.66	0.59		
DTPrate	0.89	0.89	0.89	0.89	0.93	0.97	0.88	0.88	0.79	0.54	0.97	0.97	0.95	0.98	0.98	0.75		-0.97	-0.99	-0.96	0.90	0.81		
WAUC	0.94	0.94	0.94	0.94	0.92	0.99	0.93	0.93	0.84	0.59	0.98	0.97	0.94	0.99	0.95	0.77	0.97		0.98	0.99	-0.93	-0.87		
GAUC	0.90	0.90	0.90	0.90	0.94	0.98	0.89	0.89	0.80	0.55	0.98	0.98	0.96	0.98	0.98	0.75	0.99	0.98		0.96	-0.91	-0.82		
WAUCB	0.94	0.94	0.94	0.94	0.89	0.98	0.93	0.93	0.83	0.58	0.97	0.96	0.92	0.98	0.93	0.76	0.95	0.99	0.96		-0.93	-0.87		
MAE	0.94	0.94	0.94	0.94	0.85	0.91	0.93	0.93	0.83	0.58	0.91	0.90	0.87	0.91	0.87	0.75	0.88	0.92	0.89	0.92		0.88		
RMSE	0.92	0.92	0.92	0.92	0.78	0.85	0.90	0.90	0.80	0.60	0.85	0.83	0.81	0.85	0.79	0.71	0.81	0.86	0.82	0.86	0.88			

Table 6.29 - Weighted macro average metrics: rank (bottom-left) and linear (top-right) correlation results for all datasets

Linear																							
TP	FP	TN	FN	P	R	A	E	FPR	FPP	SMCC	F1	F05	F2	DTPpr	DTPperc	DTPrate	WAUC	GAUC	WAUCB	MAE	RMSE		
	-0.99	0.93	-0.99	0.89	0.99	0.96	-0.96	-0.79	-0.87	0.98	0.98	0.93	0.99	-0.94	-0.84	-0.96	0.99	0.97	0.98	-0.95	-0.91		
TP																							
FP	0.99	-0.91	1.00	-0.90	-0.99	-0.95	0.95	0.80	0.87	-0.98	-0.98	-0.93	-0.99	0.94	0.84	0.96	-0.99	-0.97	-0.98	0.95	0.91		
TN	0.99	0.99		0.82	0.92	0.89	-0.89	-0.73	-0.79	0.90	0.91	0.86	0.92	-0.87	-0.77	-0.89	0.92	0.90	0.91	-0.88	-0.85		
FN	0.99	1.00	0.99		-0.90	-0.99	-0.95	0.80	0.87	-0.98	-0.98	-0.93	-0.99	0.94	0.84	0.96	-0.99	-0.97	-0.98	0.95	0.91		
P	0.88	0.88	0.88		0.90	0.89	-0.89	-0.82	-0.83	0.95	0.93	0.99	0.93	-0.97	-0.78	-0.93	0.91	0.94	0.88	-0.88	-0.79		
R	0.99	0.99	0.99	0.88		0.96	-0.96	-0.79	-0.87	0.98	0.98	0.93	0.99	-0.94	-0.84	-0.96	0.99	0.97	0.98	-0.95	-0.92		
A	0.95	0.95	0.95	0.87	0.95		-1.00	-0.89	-0.92	0.97	0.96	0.92	0.96	-0.91	-0.84	-0.94	0.97	0.95	0.96	-0.92	-0.88		
E	0.95	0.95	0.95	0.87	0.95	1.00		0.89	0.92	-0.97	-0.96	-0.92	-0.96	0.91	0.84	0.94	-0.97	-0.95	-0.96	0.92	0.88		
FPR	0.80	0.80	0.80	0.82	0.80	0.88	0.88		0.90	-0.85	-0.84	-0.83	-0.81	0.80	0.70	0.81	-0.84	-0.82	-0.82	0.80	0.73		
FPP	0.85	0.86	0.85	0.86	0.82	0.85	0.90	0.90	0.91	-0.89	-0.89	-0.85	-0.87	0.84	0.85	0.85	-0.89	-0.86	-0.87	0.84	0.80		
SMCC	0.97	0.97	0.97	0.97	0.94	0.97	0.96	0.96	0.85	0.88	0.99	0.97	0.99	-0.97	-0.84	-0.97	0.99	0.98	0.97	-0.95	-0.89		
F1	0.98	0.98	0.98	0.98	0.92	0.98	0.96	0.96	0.84	0.88	0.99		0.96	-0.97	-0.85	-0.98	0.99	0.98	0.97	-0.95	-0.90		
F05	0.93	0.93	0.93	0.93	0.98	0.93	0.91	0.91	0.84	0.85	0.97	0.96		-0.98	-0.81	-0.95	0.94	0.97	0.92	-0.92	-0.84		
F2	0.99	0.99	0.99	0.99	0.91	0.99	0.96	0.96	0.82	0.86	0.99	0.99	0.95	-0.96	-0.85	-0.97	0.99	0.98	0.98	-0.95	-0.90		
DTPpr	0.94	0.94	0.93	0.94	0.95	0.94	0.90	0.80	0.83	0.96	0.96	0.97	0.96		0.85	0.98	-0.94	-0.98	-0.92	0.91	0.83		
DTPperc	0.90	0.90	0.90		0.81	0.90	0.88	0.75	0.88	0.89	0.90	0.85	0.90	0.88		0.87	-0.84	-0.86	-0.83	0.80	0.72		
DTPrate	0.96	0.95	0.95	0.95	0.91	0.96	0.92	0.80	0.84	0.97	0.98	0.95	0.97	0.98	0.90		-0.97	-0.99	-0.95	0.93	0.85		
WAUC	0.99	0.99	0.99	0.99	0.90	0.99	0.97	0.84	0.87	0.98	0.98	0.94	0.99	0.94	0.89	0.96		0.98	0.99	-0.96	-0.91		
GAUC	0.97	0.97	0.97	0.97	0.93	0.97	0.94	0.82	0.85	0.98	0.98	0.96	0.98	0.98	0.90	0.99	0.98		0.96	-0.94	-0.87		
WAUCB	0.98	0.98	0.98	0.98	0.86	0.98	0.96	0.81	0.86	0.96	0.97	0.91	0.98	0.91	0.89	0.94	0.98	0.96		-0.95	-0.91		
MAE	0.94	0.94	0.94	0.94	0.87	0.94	0.92	0.79	0.82	0.94	0.94	0.91	0.94	0.91	0.86	0.92	0.95	0.94	0.94		0.88		
RMSE	0.92	0.92	0.92	0.92	0.81	0.92	0.90	0.75	0.80	0.90	0.90	0.85	0.91	0.86	0.83	0.88	0.92	0.89	0.91	0.88			

Table 6.30 – Rank and linear correlations averaged by metric for all datasets

Micro		TP	FP	TN	FN	P	R	A	E	FPR	FPP	SMCC	F1	F05	F2	DTPPr	DTPperc	DTPrate	WAUC	GAUC	WAUCB	MAE	RMSE	Avg.
	Ranks	0.98	0.98	0.98	0.98			0.99	0.99	0.99	0.99	0.99	0.99	0.99		0.99	0.99	0.99	0.99	0.99	0.99	0.94	0.92	0.98
	Linear	0.97	0.97	0.91	0.97			0.98	0.98	0.98	0.98	0.98	0.98	0.98		0.96	0.96	0.96	0.98	0.97	0.97	0.94	0.90	0.96
	Avg.	0.97	0.97	0.94	0.97			0.98	0.98	0.98	0.98	0.98	0.98	0.98		0.97	0.97	0.97	0.98	0.98	0.98	0.94	0.91	0.96
U. Macro	Ranks	0.91	0.91	0.91	0.91	0.87	0.91	0.91	0.91	0.83	0.64	0.92	0.91	0.89	0.92	0.89	0.78	0.89	0.92	0.90	0.91	0.88	0.84	0.88
	Linear	0.91	0.91	0.84	0.91	0.88	0.91	0.91	0.91	0.84	0.62	0.92	0.91	0.89	0.91	0.89	0.69	0.90	0.92	0.90	0.91	0.89	0.83	0.87
	Avg.	0.91	0.91	0.87	0.91	0.87	0.91	0.91	0.91	0.83	0.63	0.92	0.91	0.89	0.91	0.89	0.73	0.89	0.92	0.90	0.91	0.88	0.83	0.87
	W. Macro	Ranks	0.94	0.94	0.94	0.94	0.88	0.94	0.93	0.93	0.82	0.86	0.95	0.95	0.92	0.95	0.92	0.88	0.93	0.95	0.94	0.94	0.91	0.88
Linear	0.94	0.94	0.88	0.94	0.89	0.94	0.93	0.93	0.82	0.87	0.95	0.95	0.92	0.95	0.92	0.83	0.93	0.95	0.94	0.93	0.93	0.91	0.86	0.91
Avg.	0.94	0.94	0.91	0.94	0.88	0.94	0.93	0.93	0.82	0.86	0.95	0.95	0.92	0.95	0.92	0.85	0.93	0.95	0.94	0.93	0.93	0.91	0.87	0.91

A first general observation is that for any of the averaging techniques the resulting correlations are very strong. This is particularly manifested in the micro-average case as shown by the 0.97 average correlations between all the metrics. As expected, the very strong correlations are also expressed in the two dendrograms in Figure 6.9, where it can be seen that only the probabilistic metrics appear outside the main cluster.

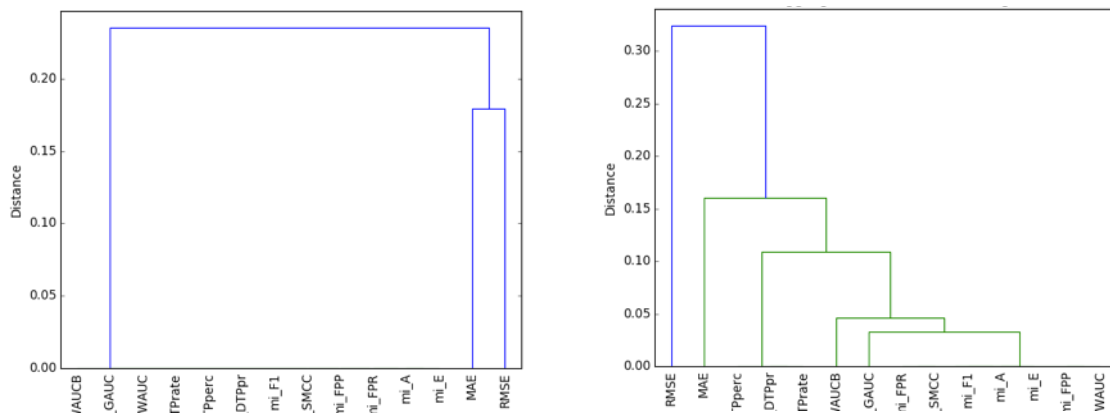


Figure 6.9 – Dendrograms showing ranks (left) and linear (right) correlations of the micro-average performance metrics across datasets

To illustrate this, in Figure 6.10 we plot the top 5 models selected by each cluster (vertical series) against the number of true positives and false positives (line series). For this purpose we use the KNN algorithm and select the results from the first dataset (PLAID_{1.5}).

As it can be seen, the Top 5 results of each metric are very similar, hence reflecting the proximity of the two clusters. Likewise, it is possible to see that the selected models are shifted to the right, i.e., towards maximizing the number of true positives.

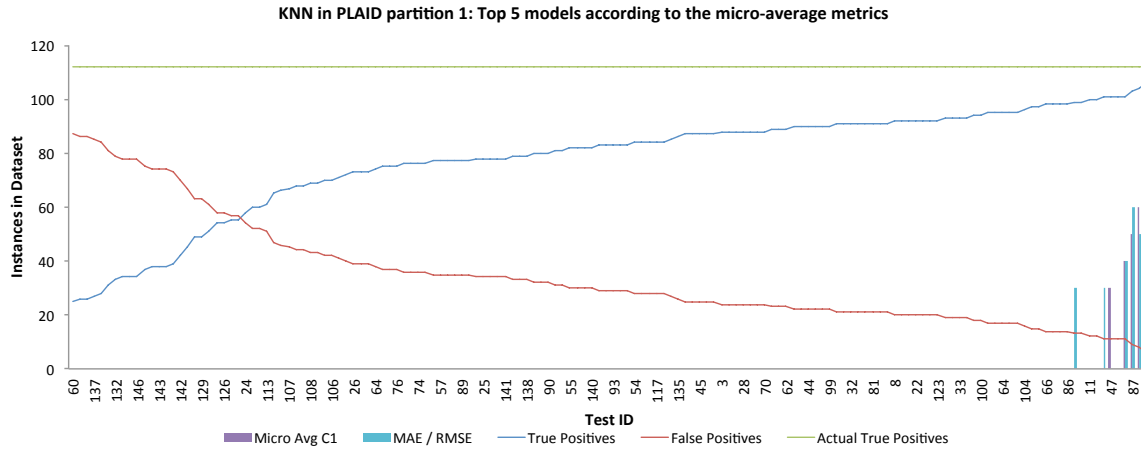


Figure 6.10 – TOP 5 models selected by the micro-average and probabilistic metrics

With regard to the macro-average metrics, a general observation is that the *weighted macro-average* metrics show better average correlation values than their *unweighted* counterparts. As expected, these differences are also reflected in the dendrograms through the vertical axis scale (distances), which is considerably smaller in the weighted macro-average metrics.

Another general observation is the high pairwise correlations between all the rank metrics (both coefficients above 0.95). Still, these values are lower than those observed in the event detection problem, which indicates that despite they tend to select the same models there might be a number of occasions when this won't happen. For example, it can be seen from the dendrograms that $GAUC$ is closely correlated to DTP_{Rate} , whereas $WAUC$ is more correlated with F_2 .

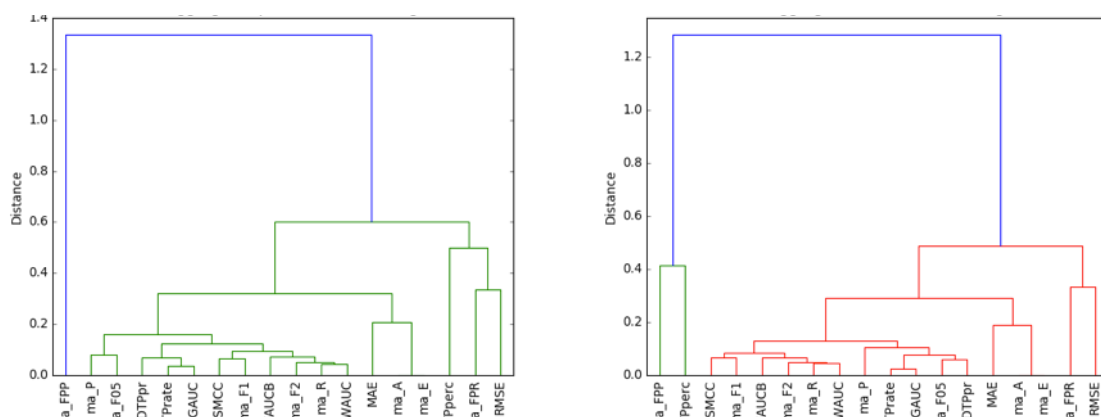


Figure 6.11 – Dendrograms showing ranks (left) and linear (right) correlations of the unweighted macro-average performance metrics across datasets

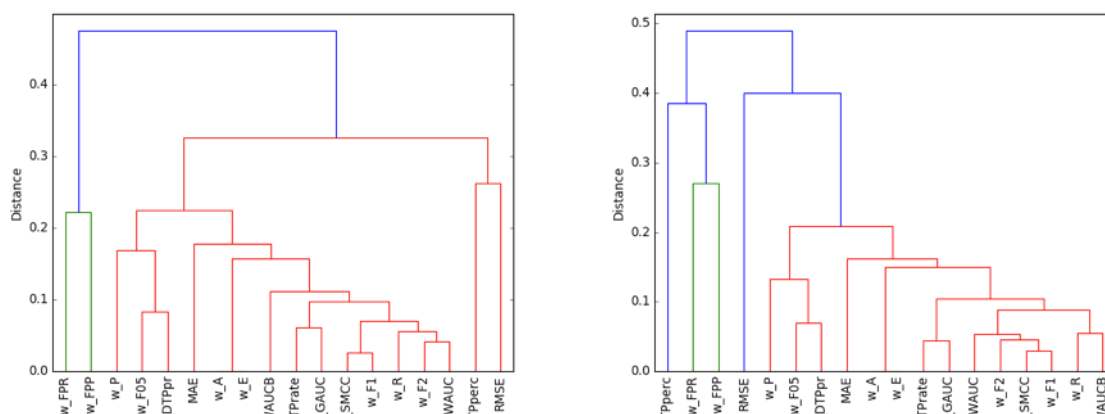


Figure 6.12 - Dendrograms showing ranks (left) and linear (right) correlations of the weighted macro-average performance metrics across datasets

From a more individual perspective, and considering only the *weighted* and *unweighted* macro-average metrics, we observe that some metrics do not correlate well with any of the other metrics. This is the case of FPR , FPP and DTP_{perc} , which as it can be seen from the dendrograms in Figure 6.11 and Figure 6.12, will not join another cluster until a cut-off distance of 0.2.

Likewise, it is possible to see that the *weighted* macro-average precision is also isolated (it only joins other metrics with a cut-off distance around 0.17), which is an indicator that macro-average Precision tends to be more sensitive to unbalanced datasets than the remaining metrics. On the contrary, it is also possible to see that Accuracy and Error-rate will always be in the same cluster, and more importantly, that they won't join any other metric until at a cut-off distance above 0.15.

Lastly, on a final note, it is important to remark the strong pairwise correlation between $SMCC$ and the three F -measures both in their *weighted* and *unweighted* versions, which ultimately reflects the theoretical guarantees that $SMCC$ is a balanced measure and can be applied to problems with balanced and unbalanced datasets [149].

To summarize, in Table 6.31 and Table 6.32 we list the clusters obtained from cutting the above dendrograms with cutoff distances of 0.05 and 0.1.

Table 6.31 – Unweighted macro-average clusters

Dist.	Rank	Linear
0.05	1. $DTP_{Rate}, GAUC$	1. $DTP_{Rate}, GAUC$
	2. $R, WAUC$	2. $F_2, WAUC, R$
	3. A, E	3. A, E
0.1	1. $DTP_{Rate}, DTP_{PR}, GAUC$	1. $DTP_{PR}, DTP_{Rate}, F_{0.5}, GAUC$
	2. $F_1, F_2, SMCC, WAUC, WAUCB, R$	2. $F_1, F_2, SMCC, WAUC, WAUCB, R$
	3. $F_{0.5}, P$	3. A, E
	4. A, E	

Table 6.32 – Weighted macro-average clusters

Dist.	Rank	Linear
0.05	1. $F_1, SMCC$	1. $F_1, F_2, SMCC$
	2. $F_2, WAUC$	2. $DTP_{Rate}, GAUC$
	3. A, E	3. A, E
0.1	1. $F_1, SMCC, F_2, WAUC, GAUC, DTP_{Rate}, R$	1. $F_1, F_2, SMCC, WAUC, WAUCB, R$
	2. $DTP_{PR}, F_{0.5}$	2. $DTP_{Rate}, GAUC$
	3. A, E	3. $DTP_{PR}, F_{0.5}$
		4. A, E

6.6 Conclusion

We now summarize the answer to research question number two and discuss the implications of our results for future research. We then highlight some of the limitations in this work and outline some possible steps to enhance the work done in this chapter.

6.6.1 Research question

We now attempt to provide the answer to the research question number two: “*How do performance metrics compare to each other when applied to event detection and event classification algorithms?*”

To do this, we first look at the differences and similarities in the metrics that are used in the two problems, namely the confusion matrix based and rank based metrics.

1. In event detection problems there is very little correlation between *Precision*, *Recall* and any version of the *F-measure*. This is particularly evident in the case of F_2 that does not have a strong correlation with any of the studied metrics.
 - a. Ultimately, this is a reflection of the nature of the data that can be generated by event detectors. For instance, it is possible to have detectors with very high *Precision* and very low *Recall* (conservative detectors) and detectors with very high *Recall* at the expense of very low *Precision* (liberal detectors). As such, the harmonic mean of *Precision* and *Recall* in these detectors will be distant from both values, which is then reflected in low pairwise correlation values.
2. On the contrary, in the case of event classification algorithms, there is a clear tendency for the existence of strong pairwise correlations between all the metrics that balance *Precision* and *Recall*.
 - a. Micro-average metrics are all extremely correlated, meaning that the same models will be selected independently on the performance metric. Ultimately, when using micro-average metrics the larger classes will dominate the metric, which in the NILM problem can become problematic given the unbalanced nature of the problem. For example, a classifier that does a great job with

refrigerators but misses the coffee machine all the time will be considered a very good algorithms because the number of refrigerator events is much higher than the number of coffee machine events.

- b. There are some subtle differences between the unweighted and weighted macro-average metrics. One of such differences is that the weighted macro-average *Precision* tends to appear isolated from the remaining metrics, which may indicate a different treatment of the false positives.
3. In event detection, F_1 , $SMCC$ and DTP_{PR} have very low average correlation to the remaining metrics (< 0.5). Still, they are very highly correlated between themselves (≥ 0.97). Hence, it is expected that these metrics will mostly select the same models.
4. We have also seen that metrics that derive only from the true positive cases behave similarly in both problems. This can be easily observed by the strong correlations between *Recall*, *AUC* and DTP_{Rate} in both event detection and classification algorithms.
5. Rank metrics are of very little use in event detection problems since they are dominated by *Recall*. In other words, the models selected by the *AUC* metrics will be the same as the ones selected by *Recall*.
6. In the event classification problem the *AUC* metrics also have high correlation between themselves. However, since they form different clusters it is very unlikely that they can be used interchangeably.
 - a. This result confirms the early finding from [38] that *AUC* metrics tend to correlate well between themselves.
 - b. Furthermore, in the case of the event classification, it also confirms the observation that *AUC* metrics are very correlated with most performance metrics (> 0.85) [37], [38].
7. Lastly, it is also possible to see that *Accuracy* and *Error-rate* appear in the same cluster in both problems. Still, we should stress that this result has its own peculiarities. More concretely, in the case of event detection, these two metrics only report based on the number of false positives and true negatives (see Table 6.24).

Therefore, if we consider the much larger number of true negative when compared to the number of false positive cases, two situations will occur:

- a. *Accuracy* and *error-rate* will always have values very close to one and zero, respectively. Hence, making this two metrics of very little use to report results.
- b. *Accuracy* and *error-rate* will report mostly based on the variations in the number of false positives. Hence, the considerably strong pairwise correlations with *FPP* and *FPR* (0.92).

We now look at the metrics that are exclusive to each problem.

8. Regarding the domain specific metrics used for event detection, it becomes clear that these metrics rely heavily on the datasets. In particular the *APC* metrics that depend both on the number of power events and respective amplitudes.
 - a. The only exception to this is the high correlation (linear and rank) between the TPC_{FN} and *Recall*, which indicate that most of the missed events have similar amplitudes in absolute value.
9. Concerning the probabilistic metrics used in event classification, they do not correlate well with the remaining metrics. In other words, these metrics evaluate the performance in a totally different manner than the remaining metrics; as such they should be taken into consideration when evaluating event classification algorithms.
 - a. This result is also in accordance with previous work in performance metrics for classification problems [37].

6.6.2 Implications

In this chapter, we proposed ourselves to study the relationship between a number of well-studied performance metrics when they are used to assess the performance of event detection and event classification algorithms.

We now draw some implications of this work to future research in this topic. More concretely, the major implications of this work are twofold: i) *uncovering important differences and similarities between measures*, and ii) *highlighting niches for which additional metrics should be studied and new ones created*.

Our results have uncovered important differences and similarities between performance measures. This can be helpful when choosing the most adequate metric or set of metrics to meet a specific goal. For example, we have seen that in event detection problems using only Precision or Recall alone can lead to completely different conclusions, since they tend to report solely based on minimizing the false positives (Precision) or maximizing the true positives (Recall).

On the other hand, we have seen that the micro-average metrics for event classification all tend to select the same models, independently of the underlying characteristics of the data (i.e., classes with more examples dominate the metric). Hence, any conclusions drawn from them should be supported by sufficient evidence from other metrics.

This work also highlights some areas in which additional existing metrics should be studied and possibly new ones created. For instance, our study in event detection reveals that only the DTP_{PR} metric was able to select models where *Precision* and *Recall* are closer to each other. Therefore, it would be important to study other metrics related with the *Precision-Recall (PR)* curve. This includes, for example, the Mean Average Precision (*MAP*), which is equivalent to the area under the PR curve; and the break-even point (*BEP*), i.e., the point(s) where Precision and Recall are the same [156 - chapter 8].

Additionally, this work has also highlighted the potential of DSM to unveil important characteristics of the evaluated algorithms (e.g., TPC_{FN} and TPC_{FP}). As such, it is important conduct further study using this metrics. Likewise, future work should aim at defining new metrics that take into account other characteristics of the data (e.g., the number of simultaneous or near simultaneous events), and disaggregation information such as the number of missed cycles (both *ON* and *OFF* are missed).

Lastly, it would also be relevant to introduce metrics that take into consideration concepts from cost-sensitive learning [151]. For instance, power events from small appliances may be less relevant than those from larger appliances, hence they should have a smaller miss-detection / miss-classification cost. On the other hand, appliances that are rarely used should

have a high miss-detection / miss-classification cost, since failing to correctly identify the few occurrences of such appliances will result in a significant underestimation of their consumption.

6.6.3 Limitations and Future Work

As it was mentioned several times in this thesis, one of the main challenges to the evaluation of event-based NILM approaches is the lack of labeled datasets.

This issue is especially prominent when it concerns to datasets for event detection, and despite we have managed to contribute to minimize this issue with two additional weeks of fully labeled data, we are aware that this process introduces some limitations to our work. First, the two datasets were labeled from ground-truth data that was collected every 6 seconds, and there was no ground-truth for all the appliances. Second, the same person labeled the two datasets; as such the labels are subject to the interpretation of one single agent.

Consequently, to better generalize the results from this work, future iterations should look at incorporating properly curated datasets, i.e., dataset whose labels have been previously validated by other researchers. Likewise, future work should incorporate other event detection algorithms, in particular those under the matched filters category that are not represented in this work. Finally, as it was mentioned in the previous sub-section, future work should also incorporate additional performance measures.

We should also note that the datasets used in event classification also add some shortcomings to the results in this work. One of such limitations is the fact that datasets only contain positive transitions, i.e., loads going from the *OFF* to the *ON* state. Furthermore, we should mention that all the examples in the dataset were carefully commissioned such that the extracted features were the best possible, which is naturally far from the conditions that NILM algorithm will face when deployed in real houses.

Consequently, future analyses of performance metrics for event classification algorithms should be conducted using scenarios that are closer to those that event classification will face in real world deployments (e.g., erroneous detections and previously unseen appliances). A

simple method to achieve this would be by deliberately introducing examples with erroneous and previously unseen classes, hence mimicking the presence of false positives and previously unseen appliances. Another possibility would be by introducing the concept of ceiling analysis, where event detection, feature extraction and event classification are executed in parallel and the output of one algorithm is the input of the other (i.e., the output of the event detection algorithm is the input of the feature extraction and the extracted features are the input of the classification algorithm).

Future iterations of this work should also consider the performance metrics used to evaluate energy estimation algorithms. Still, we should remark that this task is particularly challenging for a number of reasons. First, only a few authors have addressed the energy disaggregation stage of the event-based NILM pipeline, hence only a very limited number of energy estimation algorithms may be available. Second, energy estimation algorithms require fully labeled datasets with enough data to serve as training and testing sets. Third, some of the proposed approaches for energy estimation also need appliance level sub-metered data in order to automatically learn the appliance models.

Finally, we would also like to acknowledge the existence of other methods that could have been used in this work. These include the application of different rank correlation coefficients, like *Kendall's tau* and *Spearman's footrule* [152]–[154]; or graphical representation methods such as multidimensional scaling (MDS) and non-linear mapping (NLM) [155]–[157]. To the best of our knowledge, to date only MDS has been used as a tool to analyze the behavior of performance metrics [37]. As such, in future work it would be also relevant to compare our results with those obtained using these methods.

Chapter 7 Conclusion and Future Work

In this chapter we summarize the contents of this thesis and discuss general directions for future work in the deployment and evaluation of NILM systems.

7.1 Chapter Summaries

In Chapter 1, we motivated the importance of NILM technology for optimizing domestic energy consumption and briefly discussed its major challenges. We then highlighted the research problems addressed in this thesis, which are intrinsically related to the real world applicability of NILM.

Then, in Chapter 2, we provided a comprehensive background of this technology and reviewed existing work in the field.

In Chapter 3, we formalized the two research questions of this thesis and described the research methodology that was followed in each individual case. To state more concretely, in the first research question we focused our attention on understanding the practical issues of deploying NILM and eco-feedback systems in the context of domestic environments. As for the second research question, we proposed to study the behavior of a number of performance metrics when applied to event detection and event classification algorithms.

In Chapter 4, we presented one tool and two datasets that emerged from the work in this thesis. More concretely, we presented the Energy Monitoring and Disaggregation Data Format (EMD-DF), which aims at providing a unified interface to represent, store and handle energy disaggregation datasets. The proposed file format is an extension of the Resource Interchange File Format (RIFF), and was widely used in the work that was carried out as part

of our second research question. In this chapter we also presented SustData and SustDataED, two public dataset for energy monitoring and eco-feedback research that emerged from the work done as part of our research question number one.

Chapter 5 was dedicated to the first research question of this thesis. In this chapter we described two energy monitoring and eco-feedback platforms that were developed to support the deployment of short- and long-term energy monitoring and eco-feedback research studies. We then described three live deployments of such platforms, and thoroughly discussed the practical issues of deploying and maintaining such platforms for extended periods of time. We then concluded Chapter 5 with our answer to the research question, and a discussion on the implication of the work done for future research.

Finally, Chapter 6 was devoted to our second research question. There we presented and described the selected event detection and classification algorithms as well as the datasets and performance metrics that were used in the two cases. We then described the experimental design and thoroughly discussed the obtained results. Chapter 6 concluded with our answer to the proposed research question, and a discussion on the implications of our findings for future work in the performance evaluation of event-based NILM approaches.

7.2 Future Work

In this section we provide some general ideas to further extend the research topics addressed in this thesis.

7.2.1 Semi-Automatic Labeling of Energy Disaggregation Datasets

In this thesis we have seen that one of current challenges to NILM research is the shortage of labeled datasets to support extensive evaluations and benchmarks of event-based NILM approaches.

We argue that the main reason behind the lack of such datasets is that the actual labeling process still relies on a heavy, lengthy, and error-prone manual inspection of the whole

dataset, thus preventing the emergence of other labeled datasets. Against this background, we believe that future work should address the possibility of developing semi-automatic labeling tools for energy disaggregation datasets. Such tools are expected to significantly reduce the manual annotations effort by enabling the users to verify automatically generated annotations instead of providing them from scratch.

In the concrete case of energy disaggregation data, semi-automatic labeling can be done in three general steps:

1. Event detection algorithms are used to detect individual appliance transitions in the ground truth data. Then, individual expert users can supervise the labeling process by confirming, rejecting or correcting the system guesses. Finally, the sets of labels from the different expert users are joined and the level of agreement is measured, hence minimizing the effects of labeling ambiguity.
2. The resulting labels should then be mapped into the aggregated data following a second semi-automatic labeling session. Algorithms are responsible to find the best possible matches between the ground-truth labels and the aggregated data. Expert users are responsible for verifying the automatically generated matches.
3. Finally, the remaining events in the aggregated data should be processed using unsupervised machine-learning algorithms to find possible matches within the available labels. Whenever there is the possibility of labeling additional events, expert users should be asked to verify the suggestions. On the other hand, power events without a label should be clustered based on their characteristics (i.e., features) and presented to the expert users in a final supervision session.

Semi-automatic labeling of machine learning data is not a new topic and was already attempted in other domains like context aware driving [158], image segmentation [159] and video annotations [159], [160]. Regarding the field of energy disaggregation, in our own work, we propose and evaluate an initial version of our own semi-automatic labeling prototype [161].

7.2.2 Controlled Deployments of NILM Technology

In Chapter 5, we have seen that despite being necessary to learn how people react to new technologies, real-world deployments pose significant challenges when it is necessary to conduct and validate studies that require a greater level of engagement from the users.

Consequently, we believe that in order to conduct more accurate studies of NILM technology when deployed in the wild, future studies should be conducted in more controlled environments. Next, we briefly describe some of the potential benefits of such controlled deployments in furthering NILM research:

1. Assess the value proposition of NILM as a tool that helps users save energy. This can be achieved by means of A/B testing [162]. In such tests, two randomized groups of households should be provided with energy eco-feedback through the same communication channel, but only one of the groups would have access to disaggregated consumption information.
2. Assess the performance of NILM as a tool to disaggregate energy by means of the simultaneous deployment of NILM and multiple-sensor technology.
 - a. Such deployments would enable the creation and evaluation of novel user interfaces for NILM systems. For example, eco-feedback user interfaces that combine aggregated and disaggregated information, and user interfaces to assist end-users in the training phase of NILM systems.
 - b. The simultaneous deployment of both technologies would also enable the creation of new, and possibly longer, dataset for NILM research.

7.2.3 Benchmark Event-Based NILM Algorithms

In this thesis we performed an experimental comparison of performance evaluation metrics for event detection and classification algorithms. Still, we did not perform any benchmarks of the actual algorithms.

Therefore, we believe that the next obvious step is to benchmark NILM algorithms using the metrics that were discussed in this thesis. Next we briefly describe two possibilities of future work in this direction:

1. Perform an in-depth analysis of the results obtained from different parameters and features sweeps. This is commonly known as sensitivity analysis, and is used to evaluate how a learned model responds to changes on its inputs [163]. This technique is in general used when creating the model and provides a number of benefits to the modelers, including: i) understand which parameters contribute the most to the output variability, ii) identify parameters that are insignificant to the model and therefore can be held constant or eliminated from the final model; and iii) identify if parameter interactions are present, and if so, which parameters (or group of) interacts with each other [164].
2. Benchmark the different algorithms using statistical significance tests to gather mathematical evidence that the obtained results are representative of the general behavior of the algorithms and not due to certain characteristics of the datasets or other random factors [165]. Literature is rich in examples of such evaluations when it comes to traditional machine learning problems. For example in [165], [166] the authors identify possible statistical testing scenarios (e.g., one algorithm vs. multiple datasets or multiple algorithms vs. multiple datasets) and describe which tests are most suitable in each situation. Still, to the best of our knowledge, to date, it is not possible to find any published research showing the application of statistical testing in energy disaggregation algorithms.

Bibliography

- [1] International Energy Agency, “World Energy Outlook 2014,” International Energy Agency, Nov. 2014.
- [2] International Energy Agency, “2013 Key World Energy Statistics,” International Energy Agency, 2014.
- [3] U.S. Energy Information Administration, “International Energy Outlook 2013,” U.S. Energy Information Administration, Jul. 2013.
- [4] Pacala and R. Socolow, “Stabilization wedges: solving the climate problem for the next 50 years with current technologies.,” *Science*, vol. 305, no. 5686, pp. 968–972, 2004.
- [5] T. Theis and J. Tomkin, *Sustainability: A Comprehensive Foundation*. University of Illinois, 2012.
- [6] C. Fischer, “Feedback on household electricity consumption: a tool for saving energy?,” *Energy Effic.*, vol. 1, no. 1, pp. 79–104, Feb. 2008.
- [7] D. Parker, D. Hoak, A. Meier, and R. Brown, “How much energy are we using? Potential of residential energy demand feedback services,” in *2006 Summer Study on Energy Efficiency in Buildings*, Asilomar, CA - USA, 2006.
- [8] K. Carrie Armel, A. Gupta, G. Shrimali, and A. Albert, “Is disaggregation the holy grail of energy efficiency? The case of electricity,” *Energy Policy*, vol. 52, pp. 213–234, Jan. 2013.
- [9] J. Froehlich, L. Findlater, and J. Landay, “The Design of Eco-feedback Technology,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, New York, NY, USA, 2010, pp. 1999–2008.
- [10] G. Peschiera, J. E. Taylor, and J. A. Siegel, “Response–relapse patterns of building occupant electricity consumption following exposure to personal, contextualized and occupant peer network utilization data,” *Energy Build.*, vol. 42, no. 8, pp. 1329–1336, Aug. 2010.
- [11] L. Pereira, F. Quintal, M. Barreto, and N. J. Nunes, “Understanding the Limitations of Eco-feedback: A One-Year Long-Term Study,” in *Human-Computer Interaction and Knowledge Discovery in Complex, Unstructured, Big Data*, Maribor, Slovenia, 2013, pp. 237–255.

-
- [12] M. Berges, E. Goldman, H. Matthews, L. Soibelman, and K. Anderson, "User-Centered Nonintrusive Electricity Load Monitoring for Residential Buildings," *J. Comput. Civ. Eng.*, vol. 25, no. 6, pp. 471–480, 2011.
 - [13] G. W. Hart, "Prototype Nonintrusive Appliance Load Monitor," MIT Energy Laboratory Technical Report, and Electric Power Research Institute Technical Report, Sep. 1985.
 - [14] G. W. Hart, "Nonintrusive appliance load monitoring," *Proc. IEEE*, vol. 80, no. 12, pp. 1870–1891, 1992.
 - [15] B. Townson, "NILM: Vehicle or Destination?," presented at the International Workshop on Non-Intrusive Load Monitoring, Vancouver, BC, Canada, May-2016.
 - [16] I. Richardson, M. Thomson, D. Infield, and C. Clifford, "Domestic electricity use: A high-resolution energy demand model," *Energy Build.*, vol. 42, no. 10, pp. 1878–1887, Oct. 2010.
 - [17] H.-Â. Cao, T. K. Wijaya, and K. Aberer, "Estimating Human Interactions with Electrical Appliances for Activity-based Energy Savings Recommendations: Poster Abstract," in *ACM Conference on Embedded Systems for Energy-Efficient Buildings*, New York, NY, USA, 2014, pp. 206–207.
 - [18] C. Laughman *et al.*, "Power signature analysis," *IEEE Power Energy Mag.*, vol. 1, no. 2, pp. 56–63, Mar. 2003.
 - [19] P. Armstrong, C. Laughman, S. Leeb, and L. Norford, "Fault Detection Based on Motor Start Transients and Shaft Harmonics Measured at the RTU Electrical Service," in *International Refrigeration and Air Conditioning Conference*, 2004.
 - [20] O. Parson, "Disaggregated Homes: Overview of the NILM field." .
 - [21] O. Parson, "Disaggregated Homes: NIALM in industry." .
 - [22] "Companies offering NILM products and services - NILM Wiki." [Online]. Available: http://wiki.nilme.eu/index.php?title=Companies_offering_NILM_products_and_services. [Accessed: 14-Jul-2016].
 - [23] Bergés, Mario and Kolter, Zico, "Non-Intrusive Load Monitoring: A Review of the State of the Art," presented at the International Workshop on Non-Intrusive Load Monitoring, Pittsburgh, PA, USA, Jul-2012.
 - [24] M. Zeifman and K. Roth, "Nonintrusive appliance load monitoring: Review and outlook," *IEEE Trans. Consum. Electron.*, vol. 57, no. 1, pp. 76–84, 2011.
 - [25] A. Zoha, A. Gluhak, M. A. Imran, and S. Rajasegarar, "Non-Intrusive Load Monitoring Approaches for Disaggregated Energy Sensing: A Survey," *Sensors*, vol. 12, no. 12, pp. 16838–16866, Dec. 2012.
 - [26] N. Batra, J. Kelly, and O. Parson, "NILMTK: An Open Source Toolkit for Non-intrusive Load Monitoring," in *e-Energy '14: Proceedings of the 5th International Conference on Future Energy Systems*, Cambridge, UK, 2014.

-
- [27] R. S. Butner, D. J. Reid, M. G. Hoffman, G. Sullivan, and J. Blanchard, "Non-Intrusive Load Monitoring Assessment: Literature Review and Laboratory Protocol," Pacific Northwest National Laboratory (PNNL), Richland, WA (US), PNNL-22635, Jul. 2013.
- [28] J. Kelly and W. Knottenbelt, "Does disaggregated electricity feedback reduce domestic electricity consumption? A systematic review of the literature," in *3rd International NILM Workshop*, 2016.
- [29] N. Batra, A. Singh, and K. Whitehouse, "Exploring The Value of Energy Disaggregation Through Actionable Feedback," presented at the International Workshop on Non-Intrusive Load Monitoring, 2016.
- [30] H. Kosonen and A. Kim, "Quantifying Plug Load Energy Use in a LEED Gold Building—Lessons Learned in the Installation Phase," in *Construction Research Congress 2016*, American Society of Civil Engineers, pp. 1234–1243.
- [31] E. Mayhorm, R. Butner, M. Baechler, G. Sullivan, and H. Hao, "Characteristics and Performance of Existing Load Disaggregation Technologies," Pacific Northwest National Laboratory, Richland, WS, USA, PNNL-24230, Apr. 2015.
- [32] Y. F. Wong, Y. Ahmet Sekercioglu, T. Drummond, and V. S. Wong, "Recent approaches to non-intrusive load monitoring techniques in residential settings," in *2013 IEEE Symposium on Computational Intelligence Applications In Smart Grid (CIASG)*, 2013, pp. 73–79.
- [33] P.-H. Lai, M. Trayer, S. Ramakrishna, and Y. Li, "Database Establishment for Machine Learning in NILM," in *International Workshop on Non-Intrusive Load Monitoring*, Pittsburgh, PA, 2012.
- [34] S. Firth, K. Lomas, A. Wright, and R. Wall, "Identifying trends in the use of domestic appliances from household electricity consumption measurements," *Energy Build.*, vol. 40, no. 5, pp. 926–936, 2008.
- [35] J. Kelly and W. Knottenbelt, "Metadata for Energy Disaggregation," *ArXiv14035946 Cs*, Mar. 2014.
- [36] Electric Power Research Institute, "Non-Intrusive Load Monitoring (NILM) Technologies for End-Use Load Disaggregation: Laboratory Evaluation I." [Online]. Available: <http://tinyurl.com/kloo5wq>.
- [37] R. Caruana and A. Niculescu-Mizil, "Data Mining in Metric Space: An Empirical Analysis of Supervised Learning Performance Criteria," 2004, pp. 69–78.
- [38] C. Ferri, J. Hernández-Orallo, and R. Modroiu, "An experimental comparison of performance measures for classification," *Pattern Recognit. Lett.*, vol. 30, no. 1, pp. 27–38, Jan. 2009.
- [39] K. D. Anderson, M. E. Berges, A. Ocneanu, D. Benitez, and J. M. F. Moura, "Event detection for Non Intrusive load monitoring," presented at the Annual Conference on IEEE Industrial Electronics Society, 2012, pp. 3312–3317.
- [40] M. Weiss, A. Helfenstein, F. Mattern, and T. Staake, "Leveraging smart meter data to recognize home appliances," presented at the Proceedings of PERCOM 2012, 2012.

-
- [41] P. Meehan, C. McArdle, and S. Daniels, "An Efficient, Scalable Time-Frequency Method for Tracking Energy Usage of Domestic Appliances Using a Two-Step Classification Algorithm," *Energies*, vol. 7, no. 11, pp. 7041–7066, Oct. 2014.
 - [42] D. Luo, L. K. Norford, S. B. Leeb, and S. R. Shaw, "Monitoring HVAC Equipment Electrical Loads from a Centralized Location Methods and Field Test Results," *ASHRAE Trans.*, vol. 108, pp. 841–857, 2002.
 - [43] Y. Jin, E. Tebekaemi, M. Berges, and L. Soibelman, "Robust adaptive event detection in non-intrusive load monitoring for energy aware smart facilities," in *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2011, pp. 4340–4343.
 - [44] K. T. Nguyen, E. Dekneuveld, B. Nicoll, O. Zammit, C. N. Van, and G. Jacquemod, "Event Detection and Disaggregation Algorithms for NIALM System," presented at the International Workshop on Non-Intrusive Load Monitoring (NILM), 2014.
 - [45] B. Wild, K. S. Barsim, and B. Yang, "A new unsupervised event detector for non-intrusive load monitoring," in *2015 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, 2015, pp. 73–77.
 - [46] M. Berges, "A Framework for Enabling Energy-Aware Facilities Through Minimally-Intrusive Approaches," CARNEGIE MELLON UNIVERSITY, 2010.
 - [47] L. Pereira, F. Quintal, R. Gonçalves, and N. J. Nunes, "SustData: A Public Dataset for ICT4S Electric Energy Research," in *Proceedings of ICT for Sustainability 2014*, Stockholm, Sweden, 2014.
 - [48] S. B. Leeb, S. R. Shaw, and J. L. Kirtley, "Transient event detection in spectral envelope estimates for nonintrusive load monitoring," *IEEE Trans. Power Deliv.*, vol. 10, no. 3, pp. 1200–1210, Jul. 1995.
 - [49] L. K. Norford and S. B. Leeb, "Non-intrusive electrical load monitoring in commercial buildings based on steady-state and transient load-detection algorithms," *Energy Build.*, vol. 24, no. 1, pp. 51–64, 1996.
 - [50] J. M. Alcalá, J. Ureña, and Á. Hernández, "Event-based detector for non-intrusive load monitoring based on the Hilbert Transform," in *2014 IEEE Emerging Technology and Factory Automation (ETFA)*, 2014, pp. 1–4.
 - [51] L. De Baets, "Event detection in NILM using Cepstrum smoothing," 2016. .
 - [52] Q. V. Le *et al.*, "Building high-level features using large scale unsupervised learning," in *International Conference on Machine Learning*, 2012. 103.
 - [53] T. Hassan, F. Javed, and N. Arshad, "An Empirical Investigation of V-I Trajectory based Load Signatures for Non-Intrusive Load Monitoring," *IEEE Trans. Smart Grid*, vol. 5, no. 2, pp. 870–878, Mar. 2014.
 - [54] H. Y. Lam, G. S. K. Fung, and W. K. Lee, "A Novel Method to Construct Taxonomy Electrical Appliances Based on Load Signaturesof," *IEEE Trans. Consum. Electron.*, vol. 53, no. 2, pp. 653–660, May 2007.

-
- [55] S. Gupta, M. S. Reynolds, and S. N. Patel, "ElectriSense: Single-point Sensing Using EMI for Electrical Event Detection and Classification in the Home," in *Proceedings of the 12th ACM International Conference on Ubiquitous Computing*, New York, NY, USA, 2010, pp. 139–148.
- [56] S. N. Patel, T. Robertson, J. A. Kientz, M. S. Reynolds, and G. D. Abowd, "At the Flick of a Switch: Detecting and Classifying Unique Electrical Events on the Residential Power Line (Nominated for the Best Paper Award)," in *UbiComp 2007: Ubiquitous Computing*, 2007, pp. 271–288.
- [57] W. L. Chan, A. T. P. So, and L. L. Lai, "Harmonics load signature recognition by wavelets transforms," in *International Conference on Electric Utility Deregulation and Restructuring and Power Technologies, 2000. Proceedings. DRPT 2000*, 2000, pp. 666–671.
- [58] J. Gao, E. Can Kara, S. Giri, and Bergés, Mario, "A feasibility study of automated plug-load identification from high-frequency measurements," 2015.
- [59] M. Weiss, "Ubiquitous computing technologies for residential energy conservation," PhD, ETH, Zurich, 2012.
- [60] G. Lin, S. Lee, J. Y.-J. Hsu, and W. Jih, "Applying power meters for appliance recognition on the electric panel," in *2010 the 5th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, 2010, pp. 2254–2259.
- [61] L. Farinaccio and R. Zmeureanu, "Using a pattern recognition approach to disaggregate the total electricity consumption in a house into the major end-uses," *Energy Build.*, vol. 30, no. 3, pp. 245–259, Aug. 1999.
- [62] A. Marchiori, D. Hakkarinen, Q. Han, and L. Earle, "Circuit-Level Load Monitoring for Household Energy Management," *IEEE Pervasive Comput.*, vol. 10, no. 1, pp. 40–48, Jan. 2011.
- [63] A. Ruzzelli, C. Nicolas, A. Schoofs, and G. M. P. O'Hare, "Real-Time Recognition and Profiling of Appliances through a Single Electricity Sensor," in *2010 7th Annual IEEE Communications Society Conference on Sensor Mesh and Ad Hoc Communications and Networks (SECON)*, 2010, pp. 1–9.
- [64] D. Srinivasan, W. S. Ng, and A. Liew, "Neural-network-based signature recognition for harmonic source identification," *IEEE Trans. Power Deliv.*, vol. 21, no. 1, pp. 398–405, Jan. 2006.
- [65] M. Baranski and J. Voss, "Nonintrusive appliance load monitoring based on an optical sensor," in *Power Tech Conference Proceedings, 2003 IEEE Bologna*, 2003, vol. 4, p. 8 pp. Vol.4-.
- [66] M. Mittelsdorf, A. Hüwel, T. Klingenberg, and M. Sonnenschein, "Submeter based Training of Multi-class Support Vector Machines for Appliance Recognition in Home Electricity Consumption Data," presented at the 2nd International Conference on Smart Grids and Green IT Systems, 2013, pp. 151–158.
- [67] T. Kato, H. S. Cho, D. Lee, T. Toyomura, and T. Yamazaki, "Appliance Recognition from Electric Current Signals for Information-Energy Integrated Network in Home Environments," in *Ambient Assistive Health and Wellness Management in the Heart of*

- the City*, M. Mokhtari, I. Khalil, J. Bauchet, D. Zhang, and C. Nugent, Eds. Springer Berlin Heidelberg, 2009, pp. 150–157.
- [68] M. Baranski and J. Voss, “Genetic algorithm for pattern detection in NIALM systems,” in *2004 IEEE International Conference on Systems, Man and Cybernetics*, 2004, vol. 4, pp. 3462–3468 vol.4.
- [69] K. Suzuki, S. Inagaki, T. Suzuki, H. Nakamura, and K. Ito, “Nonintrusive appliance load monitoring based on integer programming,” in *SICE Annual Conference, 2008*, 2008, pp. 2742–2747.
- [70] O. Kramer *et al.*, “On Ensemble Classifiers for Nonintrusive Appliance Load Monitoring,” in *Proceedings of the 7th International Conference on Hybrid Artificial Intelligent Systems - Volume Part I*, Berlin, Heidelberg, 2012, pp. 322–331.
- [71] K. Barsim, L. Mauch, and B. Yang, “Neural Network Ensembles to Real-time Identification of Plug-level Appliance Measurements,” presented at the International Workshop on Non-Intrusive Load Monitoring, Vancouver, BC, Canada, 2016.
- [72] K. S. Barsim and B. Yang, “Toward a semi-supervised non-intrusive load monitoring system for event-based energy disaggregation,” in *2015 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, 2015.
- [73] X. Zhu and A. B. Goldberg, “Introduction to Semi-Supervised Learning,” *Synth. Lect. Artif. Intell. Mach. Learn.*, vol. 3, no. 1, pp. 1–130, Jan. 2009.
- [74] M. Baranski and J. Voss, “Detecting Patterns of Appliances from Total Load Data Using a Dynamic Programming Approach,” in *2013 IEEE 13th International Conference on Data Mining*, Los Alamitos, CA, USA, 2004, vol. 0, pp. 327–330.
- [75] R. Streubel and B. Yang, “Identification of electrical appliances via analysis of power consumption,” in *2012 47th International Universities Power Engineering Conference (UPEC)*, 2012, pp. 1–6.
- [76] S. Giri and M. Bergés, “An energy estimation framework for event-based methods in Non-Intrusive Load Monitoring,” *Energy Convers. Manag.*, vol. 90, pp. 488–498, Jan. 2015.
- [77] K. Anderson, A. Ocneanu, D. Benitez, D. Carlson, A. Rowe, and M. Berges, “BLUED: A Fully Labeled Public Dataset for Event-Based Non-Intrusive Load Monitoring Research,” presented at the Workshop on Data Mining Applications in Sustainability (SustKDD), Beijing, China, 2012.
- [78] Z. Kolter and J. Matthew, “REDD: A public data set for energy disaggregation research,” presented at the Workshop on Data Mining Applications in Sustainability (SustKDD), San Diego, CA, USA, 2011.
- [79] A. Shyr, A. Gupta, and V. Garud, “Crowdsourcing Appliance Labels for Energy Disaggregation,” presented at the International Workshop on Non-Intrusive Load Monitoring, Vancouver, BC, Canada, 2016.

-
- [80] H. Shao, M. Marwah, and N. Ramakrishnan, "A Temporal Motif Mining Approach to Unsupervised Energy Disaggregation: Applications to Residential and Commercial Buildings," presented at the AAAI Conference on Artificial Intelligence, 2013.
 - [81] H. Gonçalves, A. Ocneanu, and M. Berges, "Unsupervised disaggregation of appliances using aggregated consumption data," presented at the Data Mining Applications in Sustainability (SustKDD '11), San Diego, CA, USA, 2011.
 - [82] Z. Kolter, S. Batra, and A. Ng, "Energy Disaggregation via Discriminative Sparse Coding," presented at the Advances in Neural Information Processing Systems, 2010, pp. 1153–1161.
 - [83] O. Parson, S. Ghosh, M. Weal, and A. Rogers, "Non-intrusive load monitoring using prior models of general appliance types," presented at the Conference on Artificial Intelligence (AAAI-12), 2012, pp. 356–362.
 - [84] Z. Kolter and T. Jaakkola, "Approximate Inference in Additive Factorial HMMs with Application to Energy Disaggregation," presented at the International Conference on Artificial Intelligence and Statistics, 2012.
 - [85] H. Kim, M. Marwah, M. Arlitt, G. Lyon, and J. Han, *Unsupervised Disaggregation of Low Frequency Power Measurements*. .
 - [86] H. Lange and M. Bergés, "Efficient Inference in Dual-Emission FHMM for Energy Disaggregation," in *Workshops at the Thirtieth AAAI Conference on Artificial Intelligence*, 2016.
 - [87] S. Makonin, F. Popowich, L. Bartram, B. Gill, and I. V. Bajic, "AMPds: A Public Dataset for Load Disaggregation and Eco-Feedback Research," presented at the Electrical Power and Energy Conference (EPEC), 2013 IEEE, Halifax, NS, Canada, 2013.
 - [88] C. Holcomb, "Pecan Street Inc.: A Test-bed for NILM," presented at the International Workshop on Non-Intrusive Load Monitoring, Pittsburgh, PA.
 - [89] J. Kelly and W. Knottenbelt, "'UK-DALE': A dataset recording UK Domestic Appliance-Level Electricity demand and whole-house demand," in *arXiv:1404.0284 [cs]*, 2014.
 - [90] N. Batra, M. Gulati, A. Singh, and M. B. Srivastava, "It's Different: Insights into Home Energy Consumption in India," in *Proceedings of the 5th ACM Workshop on Embedded Systems For Energy-Efficient Buildings*, New York, NY, USA, 2013, p. 3:1–3:8.
 - [91] Sean Barker, Aditya Mishra, David Irwin, Emmanuel Cecchet, and Prashant Shenoy, "Smart*: An Open Data Set and Tools for Enabling Research in Sustainable Homes."
 - [92] C. Beckel, K. Kleiminger, R. Cicchetti, T. Staake, and S. Santini, "The ECO Data Set and the Performance of Non-Intrusive Load Monitoring Algorithms," in *1st ACM International Conference on Embedded Systems for Energy-Efficient Buildings*, Memphis, TN. USA, 2014.
 - [93] K. Bache and M. Lichman, "Individual Household electric power consumption dataset." Irvine, CA: University of California, School of Information and Computer Science., 2013.

-
- [94] Intertek Testing & Certification Ltd, "Household Electricity Survey A study of domestic electrical product usage," UK, R66141, May 2012.
- [95] D. Murray *et al.*, "A data management platform for personalised real-time energy feedback," 2015.
- [96] C. Gisler, A. Ridi, D. Zufferey, O. A. Khaled, and J. Hennebert, "Appliance consumption signature database and recognition test protocols," in *2013 8th International Workshop on Systems, Signal Processing and their Applications (WoSSPA)*, 2013, pp. 336–341.
- [97] A. Ridi, C. Gisler, and J. Hennebert, "ACS-F2: A new database of appliance consumption signatures," in *2014 6th International Conference of Soft Computing and Pattern Recognition (SoCPaR)*, 2014, pp. 145–150.
- [98] A. S. N. Uttama Nambi, A. Reyes Lua, and V. R. Prasad, "LocED: Location-aware Energy Disaggregation Framework," in *Proceedings of the 2Nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments*, New York, NY, USA, 2015, pp. 45–54.
- [99] A. Reinhardt *et al.*, "On the accuracy of appliance identification based on distributed load metering data," in *Sustainable Internet and ICT for Sustainability (SustainIT)*, 2012, 2012, pp. 1–9.
- [100] A. Monacchi, D. Egarter, W. Elmenreich, S. D'Alessandro, and A. M. Tonello, "GREEND: An Energy Consumption Dataset of Households in Italy and Austria," in *In proceedings of the 5th IEEE International Conference on Smart Grid Communications*, Venice, Italy, 2014.
- [101] M. Kahl, A. UI Haq, T. Kriechbaumer, and J. Hans-Arno, "WHITED - A Worldwide Household and Industry Transient Energy Data Set," presented at the International Workshop on Non-Intrusive Load Monitoring, Vancouver, BC, Canada, 2016.
- [102] M. Gulati, S. S. Ram, and A. Singh, "An in Depth Study into Using EMI Signatures for Appliance Identification," in *Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings*, New York, NY, USA, 2014, pp. 70–79.
- [103] J. Kelly *et al.*, "NILMTK V0.2: A Non-intrusive Load Monitoring Toolkit for Large Scale Data Sets: Demo Abstract," in *Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings*, New York, NY, USA, 2014, pp. 182–183.
- [104] J. Liang, S. K. K. Ng, G. Kendall, and J. W. M. Cheng, "Load Signature Study Part I: Basic Concept, Structure, and Methodology," *IEEE Trans. Power Deliv.*, vol. 25, no. 2, pp. 551–560, Apr. 2010.
- [105] S. Makonin and F. Popowich, "Nonintrusive load monitoring (NILM) performance evaluation," *Energy Effic.*, pp. 1–6, Oct. 2014.
- [106] N. Czarnek, K. Morton, L. Collins, R. Newell, and K. Bradbury, "Performance comparison framework for energy disaggregation systems," in *2015 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, 2015, pp. 446–452.

-
- [107] L. Pereira, N. Nunes, and M. Bergés, “SURF and SURF-PI: A File Format and API for Non-Intrusive Load Monitoring Datasets,” presented at the International Conference on Future Energy Systems, Cambridge, UK, 2014.
 - [108] M. Ribeiro, L. Pereira, F. Quintal, and N. Nunes, “SustDataED: A Public Dataset for Electric Energy Disaggregation Research,” in *Proceedings of ICT for Sustainability 2016*, Amsterdam, The Netherlands, 2016.
 - [109] L. Pereira, F. Quintal, N. Nunes, and M. Bergés, “The Design of a Hardware-software Platform for Long-term Energy Eco-feedback Research,” in *ACM SIGCHI Symposium on Engineering Interactive Computing Systems*, Copenhagen, Denmark, 2012, pp. 221–230.
 - [110] L. Pereira, “Towards Automating the Performance Evaluation of Non- Intrusive Load Monitoring Systems,” *ICT 4 Sustain.*, 2013.
 - [111] L. Pereira and N. J. Nunes, “Towards systematic performance evaluation of non-intrusive load monitoring algorithms and systems,” in *Sustainable Internet and ICT for Sustainability (SustainIT)*, 2015, 2015, pp. 1–3.
 - [112] “Multimedia Programming Interface and Data Specifications 1.0,” IBM Corporation and Microsoft Corporation, Aug. 1991.
 - [113] E. Costanza, S. D. Ramchurn, and N. R. Jennings, “Understanding Domestic Energy Consumption Through Interactive Visualisation: A Field Study,” in *ACM Conference on Ubiquitous Computing*, New York, NY, USA, 2012, pp. 216–225.
 - [114] S. Rollins, N. Banerjee, L. Choudhury, and D. Lachut, “An In Situ System for Annotation of Home Energy Data,” presented at the ACM Workshop on Embedded Systems For Energy-Efficient Buildings, Rome, Italy, 2013, p. 23:1–23:2.
 - [115] R. Gonçalves, “OpenDataHub: An Open Dataset Management System,” Master, University of Madeira, Funchal, 2016.
 - [116] T. Kriechbaumer, A. U. Haq, M. Kahl, and H.-A. Jacobsen, “Towards a Cost-Effective High-Frequency Energy Data Acquisition System for Electric Appliances - Google Search,” presented at the International Workshop on Non-Intrusive Load Monitoring, 2016.
 - [117] M. N. Meziane, T. Picon, P. Ravier, G. Lamarque, J. C. L. Bunetel, and Y. Raingeaud, “A measurement system for creating datasets of on/off-controlled electrical loads,” in *2016 IEEE 16th International Conference on Environment and Electrical Engineering (EEEIC)*, 2016, pp. 1–5.
 - [118] “How do beacons work? The physics of beacon tech,” *Reality matters*. [Online]. Available: <http://blog.estimote.com/post/106913675010/how-do-beacons-work-the-physics-of-beacon-tech>. [Accessed: 21-Sep-2016].
 - [119] A. B. K. Daniel Kahneman, “A survey method for characterizing daily life experience: The day reconstruction method,” *Science*, vol. 306, no. 5702, pp. 1776–80, 2005.
 - [120] F. Quintal, N. J. Nunes, A. Ocneanu, and M. Berges, “SINAIS: Home Consumption Package: A Low-cost Eco-feedback Energy-monitoring Research Platform,” in

- Proceedings of the 8th ACM Conference on Designing Interactive Systems*, New York, NY, USA, 2010, pp. 419–421.
- [121] M. Barreto, “Understanding Family Motivations for...,” PhD, University of Madeira, Funchal, Portugal, 2014.
- [122] Filipe Quintal, “Eco-feedback in the wild,” PhD, University of Madeira, Funchal, Portugal, 2016.
- [123] A. L. S. Pereira, “Low cost non-intrusive home energy monitoring,” MSc, University of Madeira, Funchal, Portugal, 2011.
- [124] A. R. Viana, “Designing Services for Sustainability: The case of home energy monitoring,” MSc, University of Porto, Porto, Portugal, 2011.
- [125] N. J. Nunes, L. Pereira, F. Quintal, and M. Berges, “Deploying and evaluating the effectiveness of energy eco-feedback through a low-cost NILM solution,” in *Proceedings of the 6th International Conference on Persuasive Technology*, Columbus, OH, USA, 2011.
- [126] F. Quintal, V. Nisi, N. Nunes, M. Barreto, and L. Pereira, “HomeTree – An Art Inspired Mobile Eco-feedback Visualization,” in *Advances in Computer Entertainment*, A. Nijholt, T. Romão, and D. Reidsma, Eds. Springer Berlin Heidelberg, 2012, pp. 545–548.
- [127] F. Quintal, L. Pereira, N. Nunes, V. Nisi, and M. Barreto, “WATTSBurning: Design and Evaluation of an Innovative Eco-Feedback System,” in *Human-Computer Interaction – INTERACT 2013*, P. Kotzé, G. Marsden, G. Lindgaard, J. Wesson, and M. Winckler, Eds. Springer Berlin Heidelberg, 2013, pp. 453–470.
- [128] F. Quintal, M. Barreto, N. Nunes, V. Nisi, and L. Pereira, “WattsBurning on My Mailbox: A Tangible Art Inspired Eco-feedback Visualization for Sharing Energy Consumption,” in *Human-Computer Interaction – INTERACT 2013*, 2013, pp. 133–140.
- [129] F. Quintal, L. Pereira, N. J. Nunes, and V. Nisi, “What-a-Watt: exploring electricity production literacy through a long term eco-feedback study,” in *2015 Sustainable Internet and ICT for Sustainability (SustainIT)*, 2015, pp. 1–6.
- [130] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, “The WEKA Data Mining Software: An Update,” *SIGKDD Explor Newsl*, vol. 11, no. 1, pp. 10–18, Nov. 2009.
- [131] T. Cover and P. Hart, “Nearest neighbor pattern classification,” *IEEE Trans. Inf. Theory*, vol. 13, no. 1, pp. 21–27, Jan. 1967.
- [132] J. R. Quinlan, “Induction of Decision Trees,” *Mach Learn*, vol. 1, no. 1, pp. 81–106, Mar. 1986.
- [133] J. G. Cleary and L. E. Trigg, “K*: An Instance-based Learner Using an Entropic Distance Measure,” in *In Proceedings of the 12th International Conference on Machine Learning*, 1995, pp. 108–114.

-
- [134] C. M. Bishop, *Neural Networks for Pattern Recognition*. New York, NY, USA: Oxford University Press, Inc., 1995.
- [135] E. Frank, M. Hall, and B. Pfahringer, “Locally Weighted Naive Bayes,” in *Proceedings of the Nineteenth Conference on Uncertainty in Artificial Intelligence*, San Francisco, CA, USA, 2003, pp. 249–256.
- [136] C. Cortes and V. Vapnik, “Support-vector networks,” *Mach. Learn.*, vol. 20, no. 3, pp. 273–297.
- [137] P. Domingos, *The Master Algorithm: How the Quest for the Ultimate Learning Machine Will Remake Our World*. New York: Basic Books, 2015.
- [138] C. G. Atkeson, A. W. Moore, and S. Schaal, “Locally Weighted Learning,” *Artif Intell Rev*, vol. 11, no. 1–5, pp. 11–73, Feb. 1997.
- [139] Y. Zhao and Y. Zhang, “Comparison of decision tree methods for finding active objects,” *Adv. Space Res.*, vol. 41, no. 12, pp. 1955–1959, Jan. 2008.
- [140] X. Wu *et al.*, “Top 10 algorithms in data mining,” *Knowl. Inf. Syst.*, vol. 14, no. 1, pp. 1–37, Dec. 2007.
- [141] C.-C. Chang and C.-J. Lin, “LIBSVM: A Library for Support Vector Machines,” *ACM Trans Intell Syst Technol*, vol. 2, no. 3, p. 27:1–27:27, May 2011.
- [142] C. Hsu, C. Chang, and C. Lin, *A practical guide to support vector classification*. 2010.
- [143] J. Gao, S. Giri, E. C. Kara, and M. Bergés, “PLAID: A Public Dataset of High-resolution Electrical Appliance Measurements for Load Identification Research: Demo Abstract,” in *Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings*, New York, NY, USA, 2014, pp. 198–199.
- [144] Y. Kagolovsky and J. R. Moehr, “Current Status of the Evaluation of Information Retrieval,” *J. Med. Syst.*, vol. 27, no. 5, pp. 409–424.
- [145] H. Iba, Y. Hasegawa, and T. K. Paul, *Applied Genetic Programming and Machine Learning*, 1st ed. Boca Raton, FL, USA: CRC Press, Inc., 2009.
- [146] J. A. Hanley and B. J. McNeil, “The meaning and use of the area under a receiver operating characteristic (ROC) curve,” *Radiology*, vol. 143, no. 1, pp. 29–36, Apr. 1982.
- [147] R. Rifkin and A. Klautau, “In Defense of One-Vs-All Classification,” *J. Mach. Learn. Res.*, vol. 5, no. Jan, pp. 101–141, 2004.
- [148] K. Anderson, “Non-Intrusive Load Monitoring: Disaggregation of Energy by Unsupervised Power Consumption Clustering,” 2014.
- [149] B. W. Matthews, “Comparison of the predicted and observed secondary structure of T4 phage lysozyme,” *Biochim. Biophys. Acta BBA - Protein Struct.*, vol. 405, no. 2, pp. 442–451, Oct. 1975.
- [150] M. G. Kendall, “A New Measure of Rank Correlation,” *Biometrika*, vol. 30, no. 1/2, pp. 81–93, 1938.
- [151] C. Spearman, “‘Footrule’ for Measuring Correlation,” *Br. J. Psychol. 1904-1920*, vol. 2, no. 1, pp. 89–108, Jul. 1906.

-
- [152] R. Kumar and S. Vassilvitskii, "Generalized Distances Between Rankings," in *Proceedings of the 19th International Conference on World Wide Web*, New York, NY, USA, 2010, pp. 571–580.
 - [153] J. D. Carroll and P. Arabie, "Multidimensional Scaling," *Annu. Rev. Psychol.*, vol. 31, no. 1, pp. 607–649, 1980.
 - [154] *Modern Multidimensional Scaling*. New York, NY: Springer New York, 2005.
 - [155] V. Kumar and R. P. Leone, "Nonlinear mapping: An alternative to multidimensional scaling for product positioning," *J. Acad. Mark. Sci.*, vol. 19, no. 3, pp. 165–176.
 - [156] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*, 1 edition. New York: Cambridge University Press, 2008.
 - [157] C. Elkan, "The Foundations of Cost-sensitive Learning," in *Proceedings of the 17th International Joint Conference on Artificial Intelligence - Volume 2*, San Francisco, CA, USA, 2001, pp. 973–978.
 - [158] K. Torkkola, C. Schreiner, M. Gardner, and K. Zhang, "Development of a Semi-Automatic Data Annotation Tool for Driving Data," in *IEEE Intelligent Transportation Systems Conference, 2006. ITSC '06*, 2006, pp. 642–646.
 - [159] S. Bianco, G. Ciocca, P. Napoletano, and R. Schettini, "An interactive tool for manual, semi-automatic and automatic video annotation," *Comput. Vis. Image Underst.*, vol. 131, pp. 88–99, Feb. 2015.
 - [160] J. Niño-Castañeda *et al.*, "Scalable Semi-Automatic Annotation for Multi-Camera Person Tracking," *IEEE Trans. Image Process. Publ. IEEE Signal Process. Soc.*, vol. 25, no. 5, pp. 2259–2274, May 2016.
 - [161] L. Pereira, N. J. Nunes, and M. Bergés, "Semi-Automatic Labeling for Public Non-Intrusive Load Monitoring Datasets," presented at the SustainIT, 2015.
 - [162] R. Kohavi, "Online Controlled Experiments: Lessons from Running A/B/N Tests for 12 Years," in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, NY, USA, 2015, pp. 1–1.
 - [163] H. C. Frey and S. R. Patil, "Identification and review of sensitivity analysis methods," *Risk Anal. Off. Publ. Soc. Risk Anal.*, vol. 22, no. 3, pp. 553–578, Jun. 2002.
 - [164] H. Wang, I. Rish, and S. Ma, "Using sensitivity analysis for selective parameter update in Bayesian network learning," in *Proceedings of the AAAI Spring Symposium on Information Refinement and Revision for Decision Making: Modeling for Diagnostics, Prognostics and Prediction*, 2002, pp. 29–36.
 - [165] N. Japkowicz and M. Shah, *Evaluating Learning Algorithms: A Classification Perspective*. New York, NY, USA: Cambridge University Press, 2011.
 - [166] J. Demšar, "Statistical Comparisons of Classifiers over Multiple Data Sets," *J Mach Learn Res*, vol. 7, pp. 1–30, Dec. 2006.
 - [167] F. Quintal, N. J. Nunes, A. Ocneanu, and M. Berges, "SINAIS: Home Consumption Package: A Low-cost Eco-feedback Energy-monitoring Research Platform," in

-
- Proceedings of the 8th ACM Conference on Designing Interactive Systems*, New York, NY, USA, 2010, pp. 419–421.
- [168] M. Buevich, A. Rowe, and R. Rajkumar, “SAGA: Tracking and Visualization of Building Energy,” in *2011 IEEE 17th International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA)*, 2011, vol. 2, pp. 31–36.
- [169] Y. Kim, T. Schmid, Z. M. Charbiwala, and M. B. Srivastava, “ViridiScope: Design and Implementation of a Fine Grained Power Monitoring System for Homes,” in *Proc. Ubicomp09*, 245–254, 2009.
- [170] M. Bergés, L. Soibelman, H. S. Matthews, and E. Goldman, “Evaluating the Electric Consumption of Residential Buildings: Current Practices and Future Prospects,” 2010, pp. 71–80.
- [171] D. Powers, “Evaluation: From precision, recall and f-measure to roc., informedness, markedness & correlation,” *J. Mach. Learn. Technol.*, vol. 2, no. 1, pp. 37–63, 2011.
- [172] T. K. Paul, K. Ueno, K. Iwata, T. Hayashi, and N. Honda, “Genetic Algorithm Based Methods for Identification of Health Risk Factors Aimed at Preventing Metabolic Syndrome,” in *Proceedings of the 7th International Conference on Simulated Evolution and Learning*, Berlin, Heidelberg, 2008, pp. 210–219.
- [173] K. D. Lee, “Electric load information system based on non-intrusive power monitoring,” Thesis, Massachusetts Institute of Technology, 2003.

Appendix A Background Research

In this appendix we present a survey of smart electricity meters and another on public datasets for energy disaggregation research.

A.1 A Survey of Smart-Meters

A.1.1 Introduction

As of today, with the advent of sensing and feedback technologies, new and more advanced energy monitoring solutions are becoming available for both the utility companies and the electricity consumers. These energy meters are commonly referred to as smart-meters, and promise to go one step further than their predecessors by providing next to real-time feedback and detailed historical information on electrical energy consumption.

Broadly speaking, the term smart-meter refers to electric devices that record the electric energy consumption in pre-defined intervals and communicates the measurements to the Internet or to an in-home display. Smart-meters greatly contrast with traditional electricity meters in a sense that the latter only exist to measure the total consumption for billing purposes, whereas smart-meters record when and how much of a resource is consumed.

Smart-meters are available in many forms and can be categorized according to three dimensions, namely **time** that refers to the meters' ability to sample the waveforms and compute the power measurements (also known as throughput rate), **space**, which refers to the points in the power distribution circuit to be measured (whole-house, circuit or individual loads / appliances) and **feedback** referring to what information is provided (e.g. power consumed in kilowatt-hour, monetary cost or environmental effects like CO₂ emissions), how

it is made available to the user (e.g. built-in LCD displays, dedicated websites and mobile apps) and how frequently (e.g. close to real time, every five second, on a hourly basis, etc.).

The combination of these three dimensions is of crucial importance when deciding which kind of smart-meter to use given a specific situation. For example, if we want to provide only the aggregated consumption of a single appliance there is no need for a very high sampling rate and probably a 1 Hz throughput (**time**) plug-level meter (**space**) with a built-in LCD display updated every other second (**feedback**) will suffice. On the other hand, if we want to study the individual loads of specific home divisions we will probably need to use a circuit level meter with multiple channels (**space**), a sampling rate in the order of several kHz (**time**) and enable real time access to the raw data (**feedback**).

Yet, the most common way of categorizing smart-meters, which is directly related to the **space** dimension, is according to the number and type of sensors involved. On one hand we have the **single sensor** approaches that are used to measure the energy consumption of the whole house (or individual circuits), and on the other hand we have the **multiple sensor** approaches that enable monitoring the energy usage of individual loads. Figure 7.1 illustrates the differences between these two types, which are described in the next two sub-sections.

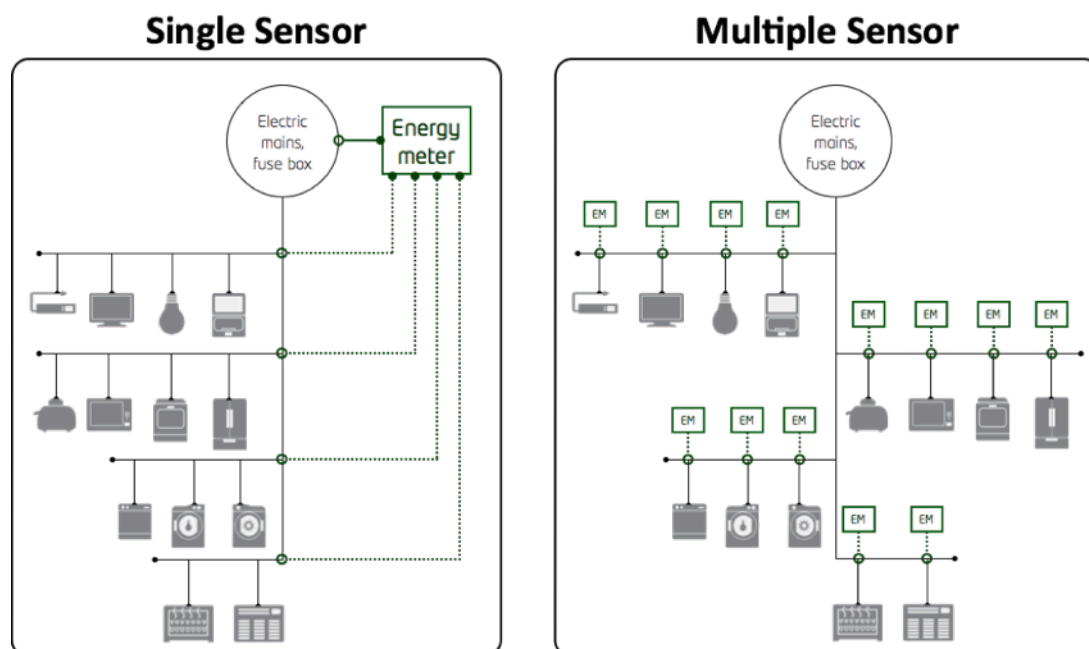


Figure 7.1 –Single sensor (Left) and Multiple sensor (Right)

A.1.2 Single sensor smart-meters

This category encompasses smart-meters that are specifically designed for monitoring and displaying the aggregate consumption of the whole house (we will be referring to these as whole house smart-meters), and in some cases the demand of each individual circuit in the house electricity installation as well (we will refer to these as circuit level smart-meters).

Overall, the installation of such smart-meters follows the schema in Figure 7.1 – left, where the smart-meter is installed in the main breaker box (solid line) to measure the whole-house demand. As for the circuit-level smart-meters, these usually have several input channels, hence also allowing for monitoring of the demand of the individual circuits (dotted lines).

Motivated by the findings that consumers could benefit from having access to their energy consumption and the constant advances in sensing technology, several models of smart-meters have reached the market in the last few years, and perhaps the whole-house

smart-meters are the easier to find with products such as The Energy Detective³⁴ (Figure 7.2), Current Cost³⁵, PowerCost Monitor³⁶ and Wattson³⁷, all of which are composed of a transmitter connected to a current sensor and a visualization element that communicates wirelessly with the latter.

Most of these smart-meter vendors claim that their products do not require professional installation (by assuming a constant voltage only a current sensor needs to be installed on the main feed, as shown in Figure 7.2 – right). Still, our own preliminary research revealed that most users are not familiar or comfortable with attempting this installation on their own [167]. This can be considered a first indicator that despite the readily availability of such systems, the cost of an overhead installation may keep people away from having their own smart-meters.



Figure 7.2 - The Energy Detective smart-meter solution: Packaged hardware (Left) and installation in the main breaker box (Right).

Regarding the circuit level smart-meters, these have a more limited availability with just a few marketed products, such as the EnerSure³⁸ Branch Circuit Power Meter (see Figure 7.3) or the SiteStage from Powerhouse Dynamics³⁹. These are also considerably more difficult to

³⁵ Current Cost, www.currentcost.com

³⁶ Blue Line Innovations Inc., www.bluelineinnovations.com

³⁷ DIY Kyoto, www.diykyoto.com

³⁸ TrendPoint Inc., www.trendpoint.com

³⁹ Powerhouse Dynamics Inc., www.powerhousedynamics.com

install and most probably will require the presence of a certified electrician for a proper and secure installation.

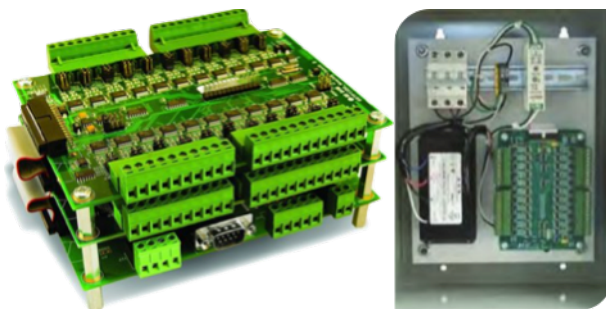


Figure 7.3 - EnerSure branch circuit power meter: Metering unit (Left) and installation in the main breaker box (Right).

There are several single sensor smart-metering solutions currently on the market, all of which have different time and feedback characteristics. In Table 7.1 we summarize some of the most relevant solutions that we found according to these two dimensions, as well as their origin (commercial - **C**, open-source - **OS** or research projects - **R**), sub-type (whole-house – **WH** or circuit level - **CL**) and price range.

A careful inspection of Table 7.1 highlights what we believe are important trends in this kind of systems. First the fact that no time dimension information is available in any of the solutions, which is probably due to the fact that although relevant for the system manufacturers, this information is not important for the system vendors and end-users.

Second, it is possible to see that most of the surveyed systems only monitor the current flowing into the house (assuming a constant Voltage of either 110 V or 230 V depending on the location) and that the tendency is to provide feedback through web-based dashboards and mobile apps.

Lastly, and although not presented in this table, most of the examined products also offer the possibility of monitoring micro-production as well (e.g., solar and wind), which indicates a clear effort from the vendors to work in proximity with this new market segment where consumers are also producers, i.e. *prosumers*.

Table 7.1 – Shortlist of single sensor smart-meter alternatives

Product	Origin	Sub-type	Sensor Technology	Time Dim.	Feedback Dimension	Price Range (€)
Current cost	C	WH	Current transformer	N/A	Real and historical energy consumption (kWh/€) in LCD displays, desktop, web and mobile applications	60 - 75
PowerCost	C	WH	Optical reader	N/A	Real and historical energy consumption; 30-day energy projection (kWh/€) through an LCD display, a web and mobile apps. Updated every 32 seconds	55 - 60
Watson	C	WH	Current transformer	N/A	Real time consumption (W) and yearly cost prediction (\$) in the Ambient display; Daily historical consumption in a desktop application via USB connection or web dashboard; Updated every 3,6,12 or 24 seconds	150 – 200
Efergy ⁴⁰	C	WH	Current transformer or optical reader	N/A	Instant energy (kW/€), historical and average energy (kWh/€) either online or using portable displays; Updated every 6, 12 or 18 seconds	50 - 70
OWL ⁴¹	C	WH	Current transformer	N/A	Instant and accumulative consumption (kW/kWh/€); Instant and accumulative CO2 emissions; Average and comparative historical (day, week, month) using portable displays, desktop applications (via USB connection) and online dashboards. Updated every 12 seconds.	35 - 75

⁴⁰ Efergy Technologies Limited, www.efergy.com⁴¹ theowl.com, www.theowl.com

Product	Origin	Sub-type	Sensor Technology	Time Dim.	Feedback Dimension	Price Range (€)
OEM ⁴²	OS	WH / CL	Current and Voltage transformers	N/A ^a	Current, voltage, apparent, real power and power factor through a portable LCD display or an online platform, which includes mobile apps	195 ^b
Flukso ⁴³	C / OS	WH	Current transformer	N/A	Averaged and Accumulative consumption (W, Wh, kWh, kWh/year) in an online dashboard updated every minute in the most expensive version	95 - 110
SEGmeter ⁴⁴	C / OS	WH / CL	Current Transformer	N/A ^a	Several gadgets (e.g. heat maps and gauges) on an online dashboard providing instant and historical energy consumption	265 – 380
EnerSure	C	WH / CL	Current and Voltage transformers	N/A	Current, voltage, power factor, real power and energy. No default feedback mechanism is available but the system offers integration with 3 rd party systems, e.g. Building Management Systems	600
SticStage	C	WH / CL	Current transformer	N/A	Instant and accumulative energy consumption of up to 44 circuits, offering also the possibility of monitoring up to three-phase whole house circuits. Feedback is provided in a web-based dashboard	380 - 690

^a The feedback dimension is not available but it is dependent on the used micro-controllers, in this cases Arduinos⁴⁵ that we know that can sample up to 10 kHz, with 10 bit samples in its most simple versions.

^b This value was calculated considering the Open Energy Monitor online shop prices for the parts required to assemble a 4 circuits monitor (75 € for the energy monitor – *emonTx V3*, 60 € for the LCD display – *emonGLCD*, 48 € for the 4 current transformers and 12 € for the AC adapter).

⁴² Open Energy Monitor, www.openenergymonitor.org

⁴³ Flukso, www.flukso.net

⁴⁴ Smart Energy Groups Pty Ltd, www.smartenergygroups.com

⁴⁵ Arduino, www.arduino.cc

A.1.3 Multiple sensor smart-meters

Multiple sensor smart-meters, on the other hand, are used to monitor the consumption of individual loads normally at the plug-level.

Most of these smart-meters have a very simple mode of operation in which the appliance to be monitored is connected to the smart-meter which is in turn connected to a wall socket (this is also known as *direct sensing* because the meter is directly connected to the load), providing basic and detailed information about the connected appliance power demands, e.g. consumption in kilowatt-hour, Voltage, Amperage, Line frequency and Power Factor.

In the less expensive models the feedback is usually provided through built-in LCD displays, like for instance the Kill a Watt P4400⁴⁶ and Belkins' Conserve Insight⁴⁷ (see Figure 7.4 – left), while in the most advanced cases we can find meters that are able to communicate wirelessly with a portable display, e.g. the Kill a Watt CO₂ Wireless (see Figure 7.4 – right).



Figure 7.4 - Multiple sensor smart-meters: Belkins Conserve (Left) and P3 Internationals Kill-a-Watt CO₂ Wireless (Right)

Most of these are, however, limited to individual appliance consumption, as they are not able to aggregate the consumption of each plug-level meter. While combining both approaches is a possible solution for this, it would not offer an integrated way of visualizing all the information.

⁴⁶ Belkin International inc, www.belkin.com

⁴⁷ P3 International Corporation, www.p3international.com

Therefore, some products are now offering this integration out-of-the-box. For example, the Plugwise⁴⁸ distributed sub-metering platform has the ability to interface each individual plug-level meter with a computer running a proprietary software that processes and displays the consumption of each appliance as well as that of groups of appliances (e.g., all the connected appliances in the house or only those in the same division).

Another limitation of such systems is that they are not able to monitor the consumption of appliances that are not plugged in a power socket (e.g., ceiling lamps) or those that have their own dedicated circuits (e.g., a water heater). To overcome this limitation, some academic work focused on what is commonly known as *indirect sensing*, in which several ambient sensors are deployed alongside the smart-meters to make it possible to find the consumption of otherwise inaccessible loads.

The FireFly wireless sensor network [168], for instance, combines plug-level smart-meters and environmental sensors (e.g. light intensity, sound level and humidity) to monitor and actuate on individual appliances. On the other hand the Viridiscopes [169] system aims at reducing the number of plug-level meters deployed in the house, and to this end the authors have combined whole-house smart-meter data with data streams from several ambient sensors to infer device-level power consumption from this information.

In Table 7.2 we summarize some of the most relevant solutions that we could find according to the two dimensions, as well as origin (commercial - **C**, open-source - **OS** or research projects - **R**), sub-type (standalone – **S** or networked - **N**) and price range.

Once again, a close inspection of the table shows that only the FireFly system provides some information for the time dimension, in this case a sampling rate of 1 kHz for each channel although no information about the processing rate (i.e. how many samples are processed per second) is given. Furthermore, it is possible to see that most of these solutions offer voltage information, power factor and line frequency (note that no reactive power information is provided whatsoever). With regard to the price ranges we have noticed some

⁴⁸ Plugwise, www.plugwise.com

discrepancies between vendors, for example the Kill a Watt P 4400 offers more information than Belkin's Conserve for less than half the price.

In summary, with the current state of home energy monitoring technology there is an important trade off that needs to be taken into consideration. Information comes at a price and if homeowners want to get individual and aggregate consumption details they will most probably have to acquire and integrate different systems. Furthermore, at some stage the potential savings that can be obtained with the help of such mechanisms will extend the payback period of these investments to unacceptable time periods [170].

Table 7.2 – Shortlist of multiple sensor smart-meter alternatives

Product	Origin	Sub-type	Time Dim.	Feedback Dimension	Price Range (€)
Current Cost	C	S	N/A	Individual appliance and whole-house instant and historical energy consumption (kWh/€) in portable LCD displays, desktop, web and mobile applications	31 ^a
Efergy	C	S	N/A	Cost and total cost in programmable units. Energy, power, voltage, current, frequency, lower and maximum power rates. All is displayed in a built-in LCD screen.	25
Belkin	C	S	N/A	Average cost (€), energy (kWh) and CO ₂ emissions (lb) in a built-in LCD display.	40
P3 International	C	S / N	N/A	Cumulative kWh, voltage, current, active power, apparent power and line frequency in the built-in LCD display (standalone version) or in the portable LCD display (in the networked version up-to 8 sensors).	18 - 62
Plugwise	C	N	N/A	Summaries of consumption by appliance and groups of appliances in a desktop or a mobile application.	105 - 420
Cloogy ⁴⁹	C	N	N/A	Instant and historical energy, power, current, voltage, frequency and power factor through a portable LCD display or online dashboard.	99 ^b
Weiss System	OS / A	N	N/A	Instant and historical energy, power, current, voltage, frequency and power factor for each plug or groups of plugs. Feedback is provided through a mobile application or an online dashboard.	N/A ^c
FireFly	OS / A	N	1 kHz ^d	Web and mobile based dashboards where it is possible to see the real and apparent power, power factor, current and voltage RMS for individual and appliance groups.	N/A

⁴⁹ Cloogy Shop, <http://shop.cloogy.pt>

Product	Origin	Sub-type	Time Dim.	Feedback Dimension	Price Range (€)
Viridiscopes	A	N	N/A	N/A ^e	N/A

^a 31 € is the cost of each plug-level meter. To this we still have to add the price of the Current Cost monitor (see **Table 7.1**).

^b 99€ is the price of a whole kit that includes two smart outlets.

^c This system uses the Plogg energy meter, which was discontinued by the manufacturer.

^d This value is for the plug-level meters. No information was found for ambient sensors.

^e This was mainly a research project and no information about the actual feedback is provided.

A.2 A Survey of Public Household Energy Datasets

A.2.1.1 REDD: Reference Energy Disaggregation Dataset

The first publicly available NILM dataset was the Reference Energy Disaggregation Data Set (REDD), which was primarily released for the event-less approaches. It includes whole-house and individual circuit consumption data collected over several months from six households in the state of Massachusetts, USA. The authors made available high-frequency (15 kHz) current and voltage for both mains phases and low-frequency active power for the individual circuits (every three to four second). Moreover, some of the circuits contain only one appliance, thus providing appliance level ground-truth data for such devices.

A.2.1.2 AMPDs: Almanac of Minutely Power dataset

The Almanac of Minutely Power Dataset (AMPDs), released in 2013, is a two-year long dataset that contains whole-house consumption (two phases) and 18 sub-metered individual circuits reported at one sample per minute. The data was collected from an individual household in the province of British Columbia, Canada.

Each record includes measurements of voltage, current, line frequency, displacement power factor, apparent power factor, real power, real energy, reactive power, reactive energy, apparent power and apparent energy. Furthermore, and similar to REDD, some individual circuits contain a single appliance, thus making this dataset appropriate for event-less approaches. Lastly, in AMPDs the electrical energy measurements are supplemented with measurements for gas and water consumption, also at 1-minute intervals.

A.2.1.3 TEALD: Tautological Energy AnaLog Dataset

The TEALD dataset, to be released in 2016, is a 1 Hz sampling frequency version of the AMPDs dataset. Currently only one house is being monitored, but according to the dataset creators a second house will be added to both AMPDs and TEALD in the near future.

A.2.1.4 Dataport Energy Dataset

In 2013 the Pecan Street Research Institute (PSRI) announced the release of a sample of their own large-scale dataset consisting of seven days of data from ten houses in Texas, USA. This dataset contained both aggregate and circuit level power readings at one and fifteen minute intervals (apparent and real power).

In the meantime, Pecan Street Inc has released a much larger dataset via the Dataport initiative. At the time of writing, the dataset contains domestic energy measurements from over 650 homes. The measurements consist of aggregate power demand and individual appliances power demands and are available in intervals of 1 Hz and one minute.

A.2.1.5 UK-DALE: UK Domestic Appliance-Level Electricity Dataset

The UK-Dale dataset, released in 2014, is a record of electric energy consumption from five homes in the United Kingdom. Overall, the dataset contains whole-house apparent power and appliance-level active power, measured every six seconds. Additionally, in three houses, the whole-house current and voltage are made available at 16 kHz along with the real power, reactive power and voltage RMS calculated at 1 Hz. For one of these homes the dataset contains over two years of aggregate consumption and individual consumption records for almost every single appliance in the home (54 loads). The other three homes were monitored for several months and each of them contains between five and 29 individually monitored appliances.

A.2.1.6 iAWE: Indian data for Ambient, Water and Electricity Sensing Dataset

The Indian data for Ambient, Water and Electricity Sensing (iAWE) dataset was released in 2013 and contains aggregated and sub-metered electricity and water measurements from one house in the city of Delhi, India for a period of 73 days. The measurements were taken at one second intervals and contain information about the whole-house and of 10 individual

appliances. Additionally, the authors also made water consumption measurements available (at the frequency of 5 Hz) and other environmental phenomena like motion, light and temperature across five rooms in the house.

A.2.1.7 Smart *: UMASS Smart * Home Dataset

The UMASS Smart* Home Data Set provides electric energy consumption data from three sub-metered houses in the state of Massachusetts, USA. For one of the houses, the power measurements are taken from the mains and from individual circuits at the frequency of one measurement per second together and listings of the start-up times of the individual appliances in each circuit. With respect to the other two houses, only aggregate whole-house consumption is available.

A.2.1.8 BLUED: Building-Level Fully-Labeled Dataset for Electric Energy Disaggregation Dataset

The Building-Level fully-labeled dataset for Electricity Disaggregation (BLUED) was, to the best of our knowledge, the second public NILM dataset to be made publicly available. Unlike its predecessor, REDD, this dataset is particularly tailored at the evaluation of event-based approaches and consists of one week of whole-house current and voltage measurements (at 12 kHz) and real and reactive power (at 60 Hz) from one house in the state of Pennsylvania, USA. Individual appliance activity is reported through a list containing the timestamps and appliances names for all the power transitions that are observed in the whole-house consumption data.

A.2.1.9 ECO: Electricity Consumption and Occupation Dataset

The Electricity Consumption and Occupation (ECO) dataset is a collection of electric energy consumption data from 6 households in Switzerland, collected over a period of eight months. This dataset contains whole-house consumption for the three phases and that of six to ten individual appliances, covering between 16% and 94% of the overall consumption. The whole-house electricity demand was collected at 1 Hz, and each measurement contains

information about the voltage, current and phase shift between them thus enabling the quick calculation of other power measurements. The active power measurements of the individual appliances were taken with varying frequency, and these were then resampled to 1 Hz to maintain consistency across the dataset.

A.2.1.10 OCTES: Opportunities for Community Groups Through Energy Storage Dataset

The OCTES dataset is the final outcome of a trial study that happened in the period between January 2012 and February 2013, during which energy monitors were installed in 69 homes from remote areas in Finland (16), Iceland (13) and Scotland (40) for consecutive periods varying between four and 13 months, depending on the house monitored.

The data has a granularity of one measurement every six to seven seconds, and each record contains information about the active power of each of the three phases, the total power (sum of the phases) and the energy price at that given instant. This is, to the best of our knowledge, the only dataset that contains the actual price of energy, which we believe will become common practice in the near future as a consequence of the constant advances and dissemination of micro-generation in households all over the world.

A.2.1.11 IHEPCDS: Individual Household Electric Power Consumption Data Set

The Individual household electric power consumption Dataset (IHEPCD) was released in 2012 and is composed of one-minute average measurements of aggregate active power, reactive power, voltage and current collected from one household in France over a period of four years. Despite the fact there is no data for individual appliances, the average active power, at the same 1-minute resolution, is made available for three individual circuits (kitchen, laundry room and the electric water heater and air-conditioner circuit).

A.2.1.12 HES: Household Electricity Use Study Dataset

This dataset contains individual appliance consumption from 251 owner-occupied households across England, and was collected between April 2010 and April 2011. The consumption of each individual appliance (13 to 51 appliances depending on the house) was monitored at two-minute intervals for periods between one month (225 households) and one year (26 households).

A.2.1.13 REFIT: Electrical Load Measurements

The REFIT dataset was released in 2014, and contains active power measurements from the aggregate consumption and of 9 individual appliances from 20 homes in the Loughborough area of the UK. The electric energy data is available at a resolution of one sample every eight seconds, and it is supplemented with gas consumption measurements recorded at 30-minute intervals.

A.2.1.14 ACS-F2: Appliance Consumption Signature Database

The Appliance Consumption Signature Database contains approximately two-hours of electrical energy consumption measurements from about 225 home appliances divided into 15 categories. The consumption was measured in terms of real power (W), reactive power (var), RMS current (A) and phase of voltage relative to current (ϕ) and is reported at ten-second intervals.

A.2.1.15 DRED: Dutch Residential Energy Dataset

The DRED dataset, released in 2015, contains both house level and appliance energy consumption information from one house in the Netherlands. The electric energy data is made available in intervals of one-second or one-minute and is supplemented with ambient, occupancy and household information (e.g., environmental parameters, room-level location information of occupants and appliance-location mapping).

A.2.1.16 Tracebase Dataset

This dataset was released at the end of 2012 and, unlike the datasets we have seen so far, does not contain any aggregated consumption data. Instead, Tracebase contains only the individual appliance consumption of 158 appliance models from 43 different appliance types, recorded between one and 10-second intervals. The data was collected in Germany and spans a total of 1883 days of power readings from an undetermined number of houses.

A.2.1.17 GREEND: Green Electric Energy Dataset

The Green Electric Energy Dataset (GREEND), released in 2014, contains appliance level consumption from nine households, five in Austria and four in Italy. For each house, nine plug-level energy monitors were deployed for periods of between three and six months, measuring the active power at the frequency of 1 Hz. Additionally the dataset creators also provide some information about the monitored spaces as well as the residents, e.g. number of floors, number of residents and an overview of the periods when they are home or away.

A.2.1.18 PLAID: Plug Load Appliance Identification Dataset

The PLAID dataset, released in 2014, is different from every other dataset we have seen so far in the sense that it does not provide actual consumption information. PLAID includes current and voltage measurement sampled at 30 kHz from 11 different appliance types present in 55 households in Pittsburgh, Pennsylvania, USA. For each measured appliance, a two-second containing both the start-up transient (if present) and the steady state operation is extracted. At the time of writing, 1074 instances are available.

A.2.1.19 WHITED: Worldwide Household and Industry Transient Energy Data Set

The WHITED dataset was introduced in 2016 and is very similar to the PLAID dataset. At time of writing, WHITED contains the appliance start-up measurements from 110 different

appliances, amounting to 47 different appliance types. The measurements were collected in three different countries (Germany, Austria and Indonesia) and consist of current and voltage waveforms sampled at 44 kHz for 5 seconds, using a custom sound card meter.

A.2.1.20 HFED: High Frequency EMI Data Set

The HFED dataset was released in 2015 and it similar to PLAID and WHITED in a sense that the available measurements are from appliance start-up operation. The difference is that instead of current and voltage, HFED consists of the transient spectral traces sampled at up to 5 MHz. The dataset is available in two collections; one from a controlled laboratory experiment, and a second set that was collected from one residential apartment.

Appendix B Research Datasets

In this appendix we provide additional details about the different datasets that are used in Chapter 6.

B.1 UK-DALE

B.1.1 House 1

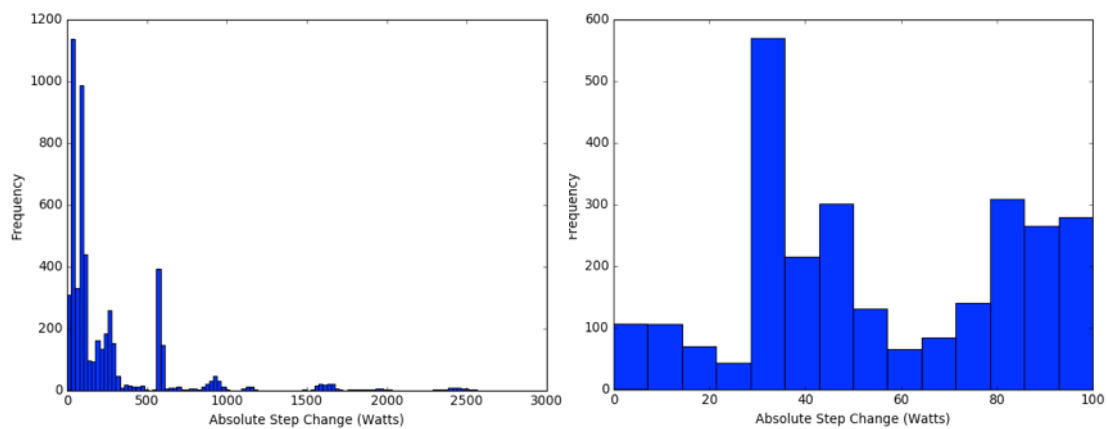


Figure 7.5 – UK-DALE 1: Distribution of power events in terms of the absolute active power change; all events (left); events below 100 Watts (right)

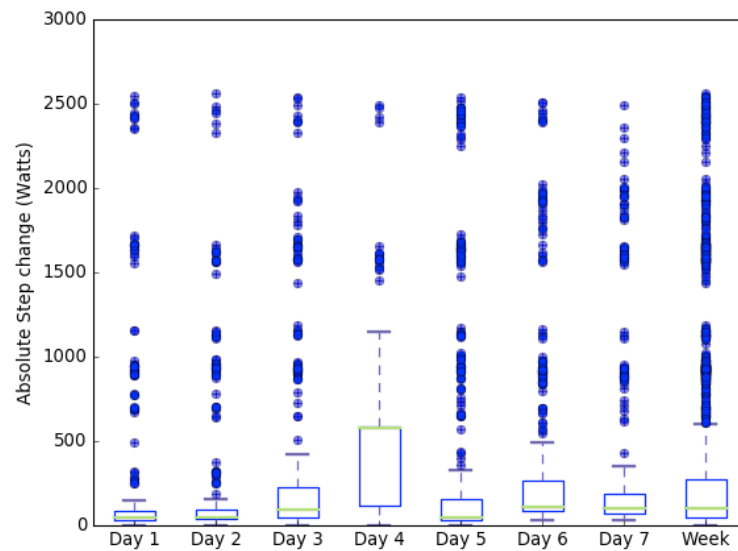


Figure 7.6 – UK-DALE 1: Boxplot showing the distribution of the power events according to the absolute power change.

B.1.2 House 2

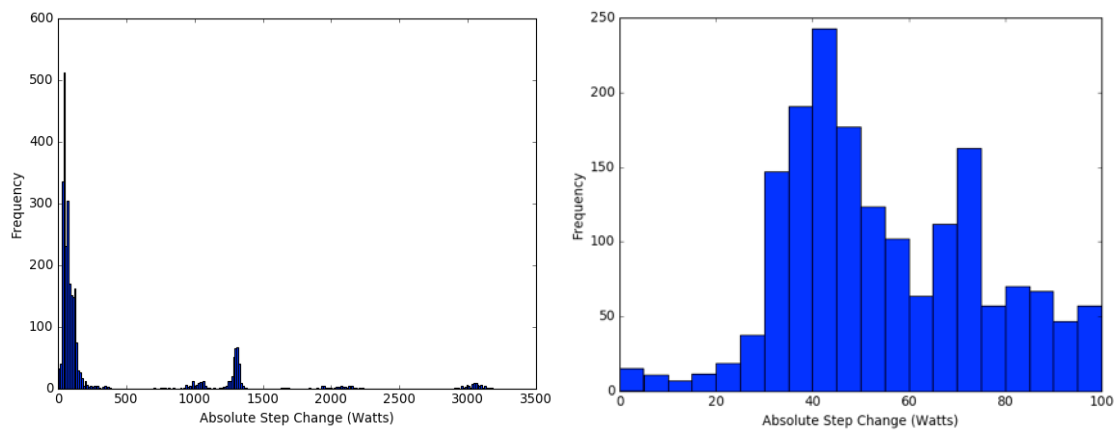


Figure 7.7 – UK-DALE 2: Distribution of power events in terms of the absolute active power change; all events (left); events below 100 Watts (right)

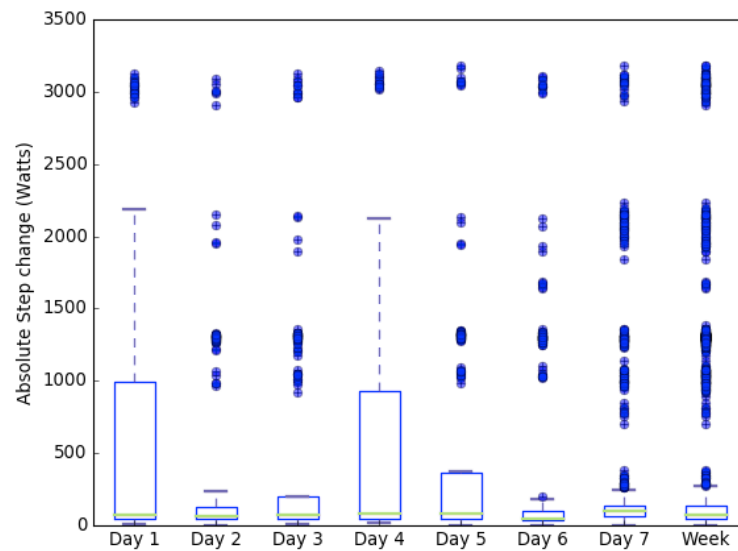


Figure 7.8 – UK-DALE 2: Boxplot showing the distribution of the power events according to the absolute power change.

B.2 BLUED

B.2.1 Phase A

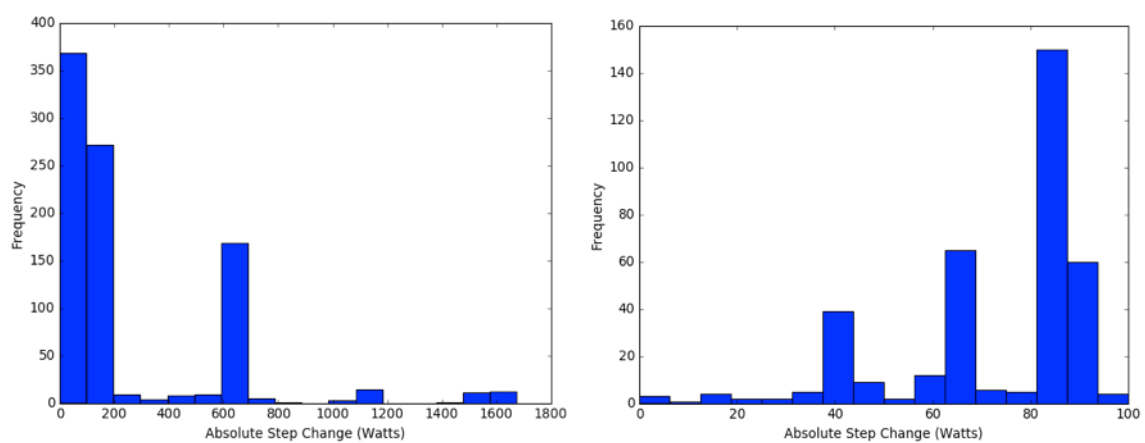


Figure 7.9 – BLUED A: Distribution of power events in terms of the absolute active power change; all events (left); events below 100 Watts (right)

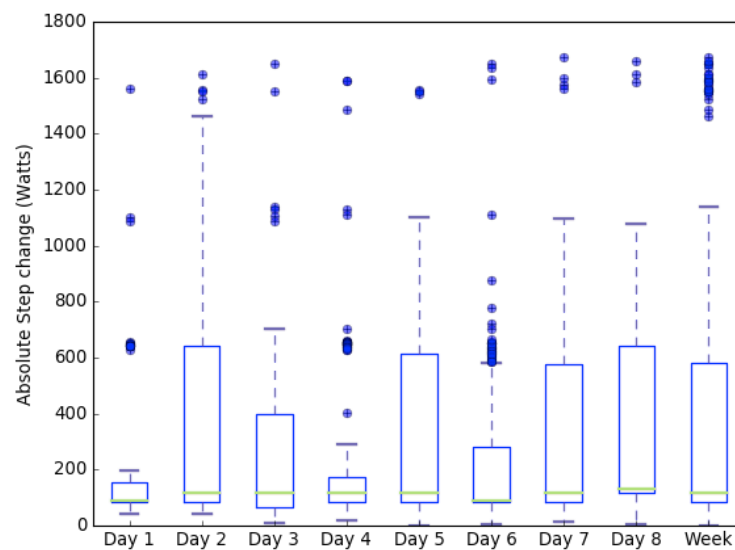


Figure 7.10 – BLUED A: Boxplot showing the distribution of the power events according to the absolute power change.

B.2.2 Phase B

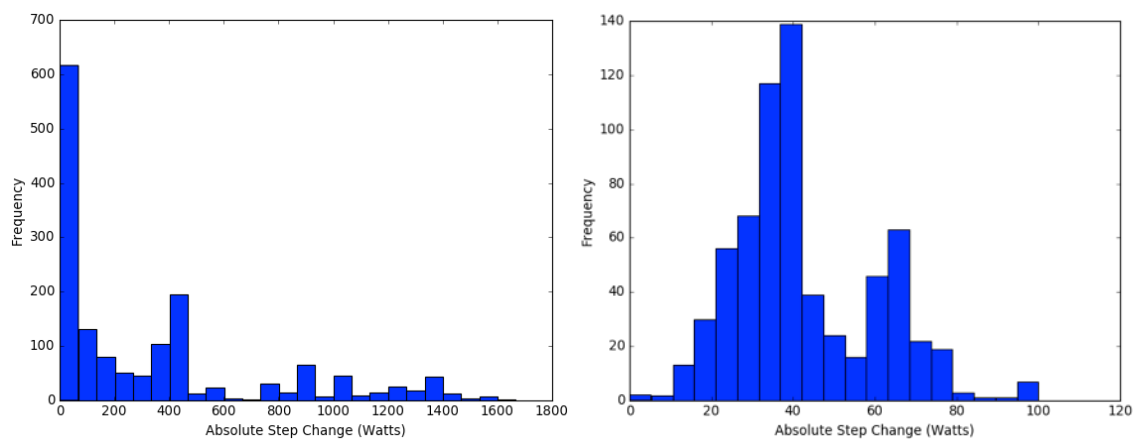


Figure 7.11 – BLUED B: Distribution of power events in terms of the absolute active power change; all events (left); events below 100 Watts (right)

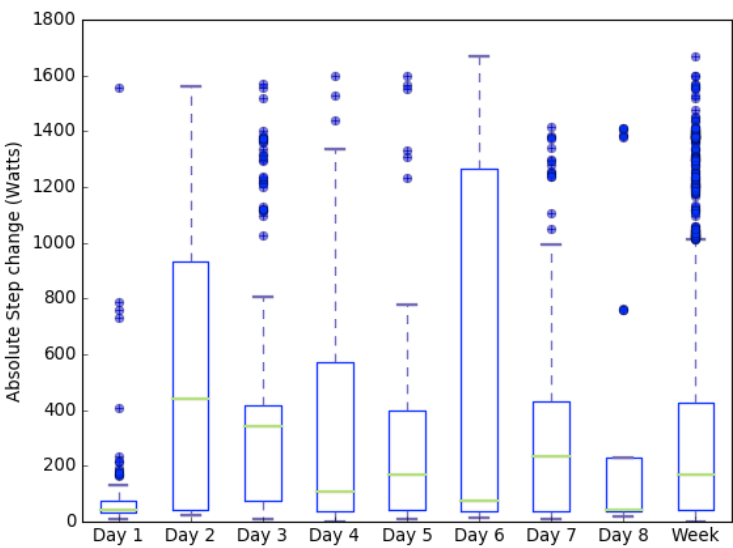


Figure 7.12 – BLUED 2: Boxplot showing the distribution of the power events according to the absolute power change.

B.3 PLAID

Table 7.3 – PLAID: Appliance instances distribution in each dataset partition

Appliance	Partition											
	1	2	3	4	5	6	7	8	9	10	11	
Air conditioner	0	9	24	5	3	0	5	0	8	10	10	74
Comp. Fluorescent Lamp	19	0	19	10	10	20	22	30	0	10	27	167
Fan	15	20	15	15	5	10	10	5	0	5	15	115
Fridge	6	2	5	5	6	2	2	0	3	2	5	38
Hair dryer	19	0	9	10	20	15	23	5	15	14	26	156
Heater	9	10	0	0	0	10	0	0	0	0	6	35
Incandescent Light Bulb	5	10	15	15	15	5	20	10	5	5	9	114

Appliance	Partition											
	1	2	3	4	5	6	7	8	9	10	11	
Laptop	15	15	15	15	17	10	20	5	15	25	20	172
Microwave	24	20	15	10	20	10	15	5	10	10	0	139
Vacuum Cleaner	0	0	0	0	5	8	10	5	5	5	0	38
Washing machine	0	0	0	10	6	0	2	1	0	1	6	26
	112	86	117	95	107	90	129	66	61	87	124	1074

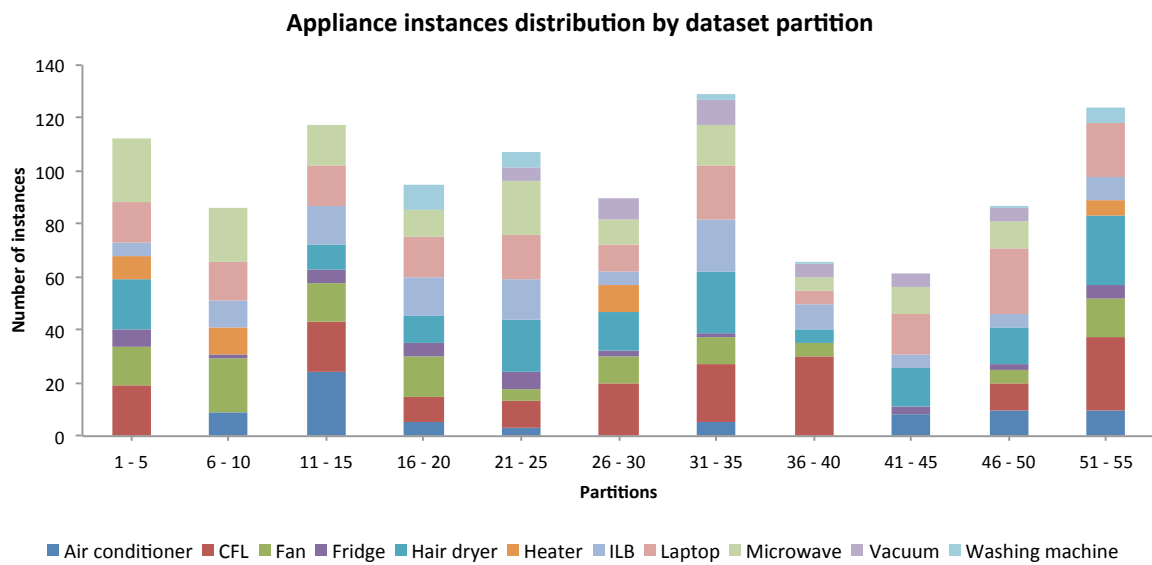


Figure 7.13 – PLAID: Appliance instances distribution in each dataset partition

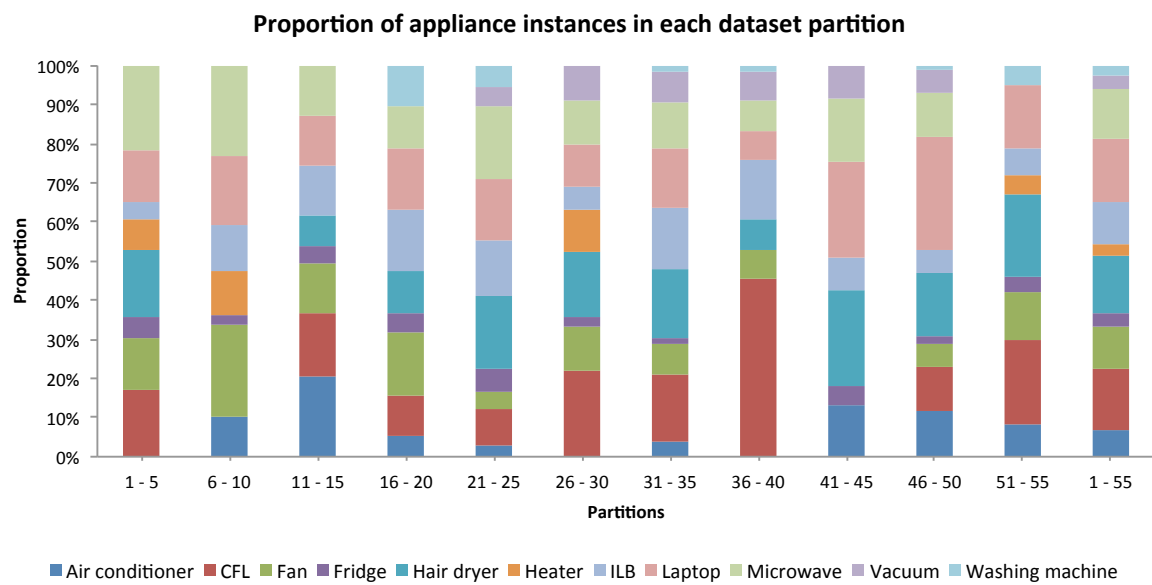


Figure 7.14 – PLAID: Proportion appliance instances in each dataset partition

Appendix C Performance Metrics

In this appendix we provide additional details about the performance metrics that are used in Chapter 6.

C.1 Event Detection

C.1.1 Confusion-matrix based performance metrics

C.1.1.1 Accuracy and Error-rate

Accuracy is the proportion of true results (TP + TN) against all the obtained results (TP + TN + FP + FN). In other words, accuracy is used to describe how close the results of an experiment are to the true values. Accuracy is formally defined by equation (7.1).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (7.1)$$

The error-rate on the other hand measures the fraction of false results (FP + FN) against all the obtained results. In other words, it is used to describe how far the obtained results are from the true values. It is formally defined by equation (7.2).

$$Error_rate = \frac{FP + FN}{TP + TN + FP + FN} = 1 - Accuracy \quad (7.2)$$

C.1.1.2 Precision and Recall

In information retrieval problems, *precision* (also called *Positive Predictive Value* – PPV) reports the fraction of retrieved instances that are relevant. In other words, *precision* is the proportion of relevant instances that were reported as being relevant (TP) against all the instances that were reported as relevant (TP + FP). Precision is formally defined by the following equation.

$$Precision = \frac{TP}{TP + FP} \quad (7.3)$$

Recall (also called sensitivity or True Positive Rate – TPR) on the other hand reports the fraction of relevant instances that were actually retrieved. In other words, recall is the proportion of relevant instances that were reported as being relevant (TP) against all the truly relevant instances (TP + FN). *Recall* is formally defined by equation (7.4).

$$Recall = \frac{TP}{TP + FN} \quad (7.4)$$

Regarding the event detection problem, *precision* reports the fraction of power events that were correctly detected among all the detected events, whereas *recall* reports the fraction of existing power events that were found by the event detector algorithm.

C.1.1.3 F β - Measure

As we can see from the two previous equations, both precision and recall report on the events that are correctly detected but at different costs. More specifically, False Positives in the case of precision and False Negatives when recall is considered.

Consequently, there is a trade-off between the two metrics that often makes the step of determining which algorithm is superior difficult to achieve when considering the combination of this two metrics. Considering the case of event detection algorithms, if we have a more liberal detector (i.e., a detector that triggers a lot of power events at a cost of a high number of wrong detections) recall will increase with the decrease of the FN but

precision will decrease due to the high number of FP. Conversely, a more conservative detector (i.e., a detector that triggers very few power events with very high probability of being correct) will have a higher precision at the expense of a much lower recall.

An alternative is to use a metric that is able to assess this trade-off. This metric is the F_β -*Measure*, which reports results in terms of the weighted harmonic mean of precision and recall. Mathematically, the F Measure is represented by equation (7.5).

$$F_\beta = (1 + \beta^2) \times \frac{\text{Precision} \times \text{Recall}}{(\beta^2 \times \text{Precision}) + \text{Recall}} \quad (7.5)$$

Where β is the weighing factor that is used to attach β times as much importance to recall as to precision. For example, if β is equal to 2 (**F₂ Measure**) recall will be twice as important as precision, whereas if β is equal to 0.5 (**F_{0.5} Measure**) precision will be twice as important as recall. Finally, in the case where β is equal to 1 (**F₁ Measure**) recall and precision have the same weight.

C.1.1.4 False Positive Rate

In information retrieval problems, the *False Positive Rate* (FPR – also called *false alarm rate*) is the fraction of irrelevant instances that were retrieved as if they were relevant. In other words, *FPR* is the proportion of irrelevant instances reported as being relevant (FP) against all the truly irrelevant instances. Mathematically, *FPR* is defined by the following equation:

$$FPR = \frac{FP}{FP + TN} \quad (7.6)$$

Regarding the event detection problem, the *FPR* reports the fraction of power events that were wrongfully triggered among all the truly negative examples (i.e., all the samples in the power signal that are not labeled as power events).

C.1.1.5 True Positive Percentage and False Positive Percentage

The True Positive Percentage (TPP) and False Positive Percentage (FPP) metrics were first introduced by [39] as an alternative to the *TPR* and *FPR* metrics. These are intended to compare TP and FP to the number of ground truth events in the data, and are formally defined as follows:

$$TPP = \frac{TP}{E} \quad (7.7)$$

$$FPP = \frac{FP}{E} \quad (7.8)$$

Where E is the number of ground truth events, which is given by the sum of true positives and false negatives. It should be noted that FPP might not always be a percentage because it is possible that the number of false positives is larger than the number of ground truth events.

C.1.1.6 Mathews Correlation Coefficient

The Mathews Correlation Coefficient (MCC) is the correlation between the predictions and the ground-truth data [149]. MCC ranges between -1 and +1 where +1 represents a perfect prediction, 0 a prediction that is not better than random and -1 indicates total disagreement between the predictions and the ground-truth data.

MCC is generally regarded as a balanced measure, i.e., insensitive to the class size [171], thus the relevance in studying this metric in the NILM domain. For a binary problem, the MCC is defined by equation (7.9).

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TN + FN) \times (TN + FP) \times (TP + FN) \times (TP + FP)}} \quad (7.9)$$

When the denominator of (7.9) is zero, MCC is set to zero as well. The standardized Mathews Correlation Coefficient ($SMCC$) was defined by [145] and is calculated as follows:

$$SMCC = \frac{1 + MCC}{2} \quad (7.10)$$

Unlike MCC that ranges between -1 and +1, the *SMCC* ranges between 0 and +1, the higher values being the better, 0 representing total disagreement and 0,5 indicating predictions that are not better than random.

C.1.1.7 Precision – Recall distance to perfect vector

The “perfect detector” in terms of precision and recall would have 100% performance in both metrics. In other words, the optimal detectors according to this two metrics are the ones where the Precision - Recall vector (P, R) is closer to the vector (1,1).

Consequently, another possible metric to combine precision and recall is the distance to the perfect detector. We will refer to this metric as distance to perfect (DTP_{PR}) and, it is defined by the following equation:

$$DTP_{PR} = \|(1, 1) - (P, R)\|^2 = \|(1 - P, 1 - R)\|^2 = P^2 + R^2 - 2 \times (P + R) + 2 \quad (7.11)$$

This metric can then be generalized such that it can be applied to other pairs of metrics and, it is formally defined by equation (7.12).

$$DTP_{\psi} = \|\vec{\theta} - \vec{\psi}\|^2 \quad (7.12)$$

Where $\vec{\psi}$ is the vector obtained by combining the new metrics and, $\vec{\theta}$ is the vector of the best possible result using such metrics (see below for more examples of this metric).

C.1.1.8 TPR – FPR distance to perfect vector

When considering this two metrics the perfect detector has a *TPR* (or sensitivity / recall) of 1 and a *FPR* of 0. This then results in the DTP_{Rate} equation presented below:

$$DTP_{Rate} = \|(1, 0) - (TPR, FPR)\|^2 = \|(1 - TPR, 0 - FPR)\|^2 = TPR^2 + FPR^2 - 2 \times TPR + 1 \quad (7.13)$$

Note that in the original paper that refers to this metric [39], the authors went a step further by neglecting the effect of the *FPR* under the assumption that due to the nature of the power signal this metric would always have a value that is very close to zero. Yet, and despite this is a fair assumption, in this work we use the original equation.

C.1.1.9 TPP – FPP distance to perfect vector

Similar to the case of the distance to perfect metric for TPR and FPR (DTP_{Rate}), the perfect detector will have a TPP equal to 1 and FPP equal to 0. Therefore, the distance to perfect metric (DTP_{perc}) can be expressed as follows:

$$DTP_{perc} = \|(1, 0) - (TPP, FPP)\|^2 = \|(1 - TPP, 0 - FPP)\|^2 = TPP^2 + FPP^2 - 2 \times TPP + 1 \quad (7.14)$$

C.1.2 Rank based performance metrics

C.1.2.1 Wilcoxon Statistics based AUC

The AUC calculated using the Wilcoxon statistics ($WAUC$) for a single point in ROC space is given by equation (7.15), where, Recall is equivalent to the True Positive Rate and Specificity to the True Negative Rate.

$$WAUC = \frac{1}{2} \times (\text{Recall} + \text{Specificity}) \quad (7.15)$$

C.1.2.2 Wilcoxon Statistics based AUC Balanced

Lastly, we also look at the balanced version of the Wilcoxon Statistics based AUC ($WAUCB$). The $WAUCB$ was introduced in [172] and it is a function of the $WAUC$ and the amount of balancing between recall and specificity. Mathematically, the $WAUCB$ is defined as follows:

$$WAUCB = WAUC \times \text{Balancing_Factor} \quad (7.16)$$

Where the balancing factor is given by:

$$\text{Balancing_Factor} = 1 - |\text{Recall} - \text{Specificity}| \quad (7.17)$$

C.1.2.3 Biased AUC

Unlike *WAUC* and *GAUC* that treat recall and specificity equally, the biased *AUC* metric (*BAUC*) assigns higher weight to the measurement that corresponds to the majority class in the dataset. Mathematically, *BAUC* is defined as follows:

$$BAUC = \frac{1}{2} \times (Recall \times Specificity) + Bias \quad (7.18)$$

Where the bias is given by:

$$Bias = \begin{cases} \frac{Recall}{2}, & \text{If the target class is the majority class} \\ \frac{Specificity}{2}, & \text{If the target class is the minority class} \end{cases} \quad (7.19)$$

The first term balances recall and specificity equally, whereas the second term biases the metric towards the majority class.

Regarding its applicability to event detection problems (see sub-section 6.3.1), event detectors are expected to have a number of true negatives much larger than the number of true positives (TP), false positives (FP) and false negatives (FN). Hence, when applied to event detection algorithms the target is the minority class, i.e., the bias will be half of the specificity.

C.1.2.4 Geometric Mean AUC

Like the name suggests, the Geometric Mean AUC (*GAUC*) is based on the geometric mean between recall and specificity. The *GAUC* is mathematically defined as follows:

$$GAUC = \sqrt[2]{Recall \times Specificity} \quad (7.20)$$

C.1.3 Domain specific performance metrics

C.1.3.1 Total Power Change

The total power change (*TPC*) is an attempt to quantify the amount of power missed (FN) or erroneously considered (FP) by event detection algorithms. As such, the total power change

is defined by two different metrics, namely the total power change of all the false negatives (TPC_{FN}) and the total power change of all the false positives (TPC_{FP}). These are formally defined according to equations (7.21) and (7.22), where M and F are the sets of all false negatives and false positives, respectively.

$$\Delta P_{FN} = \sum_{m \in M} |\Delta P_m| \quad (7.21)$$

$$\Delta P_{FP} = \sum_{f \in F} |\Delta P_f| \quad (7.22)$$

The amount of power change for an event is given by equation (7.23).

$$\Delta X_e = \frac{1}{w_3} \sum_{i=e+w_2+1}^{e+w_2+w_3} X(i) - \frac{1}{w_1} \sum_{i=e-w_1}^{e-1} X(i) \quad (7.23)$$

Where w_1, w_2 and w_3 are window lengths; w_1 and w_3 referring to the *pre-event*, *post-event* windows, and w_2 to a *delay* window that is used to allow the start-up transient to end, thus achieving a more steady state and therefore more stable delta values.

C.1.3.2 Average Power Change

The Average Power Change (APC) metric was also introduced by [39] and, as the name suggests, reports the average power change of the false negatives and the false positives. The two metrics are formally defined according to the equations below:

$$\overline{\Delta P_{FN}} = \frac{1}{|M|} \times \sum_{m \in M} |\Delta P_m| = \frac{1}{|M|} \times \Delta P_{FN} \quad (7.24)$$

$$\overline{\Delta P_{FP}} = \frac{1}{|F|} \times \sum_{f \in F} |\Delta P_f| = \frac{1}{|F|} \times \Delta P_{FP} \quad (7.25)$$

C.1.3.3 Distance to Perfect Vector

It is also possible to define the distance to perfect vector for these metrics where the perfect algorithm will have a TPC_{FN} and TPC_{FP} equal to zero. The DTP_{TPC} is therefore defined according to equation (7.26).

$$DTP_{TPC} = \|(TPC_{FN}, TPC_{FP})\|_2^2 = TPC_{FN}^2 + TPC_{FP}^2 \quad (7.26)$$

Likewise, the perfect detector according to the APC metrics will have APC_{FN} and APC_{FP} equal to zero. Consequently, by analogy, the distance to perfect metric (DTP_{APC}) can be obtained by replacing TPC for APC in equation (7.26).

C.2 Event Classification

C.2.1.1 Mean Absolute Error and Root Mean Squared Error

The mean absolute error (MAE) and root mean squared error (RMSE) are used to measure how close forecasts or predictions are to the eventual outcomes. In other words, these measures look at the average differences between those two values. MAE and RMSE are calculated according to the equations (7.27) and (7.28), where $\hat{\theta}$ is the obtained prediction and θ is the true value.

$$MAE = \frac{1}{N} \times \sum_{i=1}^N |\hat{\theta}_i - \theta_i| \quad (7.27)$$

$$RMSE = \sqrt{\frac{1}{N} \times \sum_{i=1}^N (\hat{\theta}_i - \theta_i)^2} \quad (7.28)$$

Note that the two metrics are in the same scale of the data under test, i.e., a value of one in either MAE or RMSE represent a distance of one between the prediction and the real value. Note also that we did not use probabilistic measures to evaluate detection algorithms since the absolute distances between the predictions and the true results will always be equal

to zero for true detections or equal to one otherwise. As such, the MAE (and the RMSE) will be equivalent to the error-rate defined in equation (7.2).

Appendix D Additional Tables

D.1 Parameter and Feature Sweep Lookup Tables

In section 6.4 of Chapter 6 we presented a number of parameter and feature sweeps to be performed on top of event detection and event classification algorithms. Here we provide lookup tables, such that given the identifier of the parameter sweep (i.e., the model), the reader can quickly identify the corresponding combination of parameters.

Table 7.4 to Table 7.7 refer to the event detection algorithms, whereas Table 7.8 to Table 7.10 refer to the event classification algorithms. A number of practical examples are available for each table.

D.1.1 Event Detection Lookup Tables

Table 7.7 – Different parameter configurations for the SLLD algorithm (50 Hz and 60 Hz datasets)

Tests	w0	w1										Mpre									
		0,5	1	1,5	2	2,5	3	3,5	4	4,5	5	0,5	1	1,5	2	2,5	3	3,5	4	4,5	5
[1, 100]	0,5	[1, 10]										1	2	3	4	5	6	7	8	9	10
		[11, 20]										11	12	13	14	15	16	17	18	19	20
		[21, 30]										21	22	23	24	25	26	27	28	29	30
		[31, 40]										31	32	33	34	35	36	37	38	39	40
		[41, 50]										41	42	43	44	45	46	47	48	49	50
		[51, 60]										51	52	53	54	55	56	57	58	59	60
		[61, 70]										61	62	63	64	65	66	67	68	69	70
		[71, 80]										71	72	73	74	75	76	77	78	79	80
		[81, 90]										81	82	83	84	85	86	87	88	89	90
		[91, 100]										91	92	93	94	95	96	97	98	99	100
[101, 200]	1																				
[201, 300]	1,5																				
[301, 400]	2																				
[401, 500]	2,5																				
[501, 600]	3																				
[601, 700]	3,5																				
[701, 800]	4																				
[801, 900]	4,5																				
[901, 1000]	5																				

Lookup examples

- Test ID = 1: $w_0 = 0.5$, $w_l = 0.5$, $M_{pre} = 0.5$
- Test ID = 666: $w_0 = 3.5$, $w_l = 3.5$ ($666 - 600 = 66$), $M_{pre} = 3$
- Test ID = 1000: $w_0 = 5$, $w_l = 5$ ($1000 - 900 = 100$), $M_{pre} = 5$

D.1.2 Event Classification Lookup Tables

Table 7.8 – Different parameter and feature configurations for the six event classification algorithms

Test ID	Parameter Index	Feature Index
[1, 12]	1	Test ID
[61, 78]		Subtract 48
[13, 24]	2	Subtract 12
[79, 96]		Subtract 66
[25, 36]	3	Subtract 24
[97, 114]		Subtract 84
[37, 48]	4	Subtract 36
[115, 132]		Subtract 102
[49, 60]	5	Subtract 48
[133, 150]		Subtract 120

Lookup examples

- Test ID = 1: $P_{index} = 1$, $F_{index} = 1$
- Test ID = 33: $P_{index} = 3$, $F_{index} = 9$ ($33 - 24$)
- Test ID = 66: $P_{index} = 1$, $F_{index} = 18$ ($66 - 48$)
- Test ID = 149: $P_{index} = 5$, $F_{index} = 29$ ($149 - 120$)

Table 7.9 – Different parameters and possible values for each of the six classification algorithms

Algorithm	Parameter Name	Parameter Index				
		1	2	3	4	5
KNN	K	1	3	5	13	26
Kstar	b	0	20	50	75	100
LWL-NB	K	3	5	13	26	175
DTREE	minObjects	1	2	5	10	15
ANN	learningRate	0.1	0.2	0.3	0.4	0.5
SVM	C	0.01	0.1	1	10	100

Table 7.10 – Different feature combinations for the six classification algorithms

Index	Features	Index	Features	Index	Features
1	PQ	11	PCA_Q_WF_IV	21	PQ_H_IV_Q_WF_IV
2	PQI	12	PCA_BINARY_IV	22	PQ_H_I_PCA_BINARY_VI
3	WF_I	13	PQ_H_I	23	PQ_H_IV_PCA_BINARY_VI
4	Q_WF_I	14	PQ_H_IV	24	H_I_PCA_BINARY_IV
5	Q_WF_IV	15	PQ_Q_WF_I	25	H_IV_PCA_BINARY_IV
6	H_I	16	PQ_Q_WF_IV	26	Q_WF_I_PCA_BINARY_VI
7	H_IV	17	PQ_PCA_BINARY_IV	27	Q_WF_IV_PCA_BINARY_VI
8	BINARY_VI	18	PQ_H_I_Q_WF_I	28	PCA_Q_WF_I_PCA_BINARY_IV
9	PCA_WF_I	19	PQ_H_I_Q_WF_IV	29	PCA_Q_WF_IV_PCA_BINARY_IV
10	PCA_Q_WF_I	20	PQ_H_IV_Q_WF_I	30	PCA_WF_I_PCA_BINARY_IV

D.2 Performance Metrics Pairwise Correlations

In section 6.5 of Chapter 6 we presented the pairwise correlation matrices averaged by algorithm and dataset. Here we present the pairwise correlations matrices averaged by algorithm. Table 7.11 to Table 7.15 refer to the pairwise correlations of the event detection performance metrics. Table 7.16 to Table 7.33 refer to the pairwise correlations of the event classification performance metrics.

Table 7.11 – MEH event detector: cross dataset pairwise correlations

		Linear																												
		TP	FP	TN	FN	FPP	FPR	A	E	P	R	F05	F1	F2	SMCC	DTPpr	DTPRate	DTPperc	W_AUC	W_AUCB	G_AUC	B_AUC	TPC_FN	TPC_FP	DTPtpc	APC_FN	APC_FP	DTPapc		
Ranks	TP		0.53	-0.53	-1.00	0.53	0.53	-0.45	0.45	0.19	1.00	0.25	0.38	0.63	0.51	-0.55	-0.97	0.49	1.00	0.99	0.99	1.00	-0.93	0.58	0.42	-0.35	0.34	-0.12		
	FP	-0.58		-1.00	-0.53	1.00	1.00	-0.99	0.99	-0.51	0.53	-0.48	-0.38	-0.13	-0.23	0.14	-0.45	0.96	0.53	0.55	0.50	0.53	-0.53	0.88	0.84	-0.43	0.18	-0.30		
	TN	-0.58	1.00		0.53	-1.00	-1.00	0.99	-0.99	0.51	-0.53	0.48	0.38	0.13	0.23	-0.14	0.45	-0.96	-0.53	-0.55	-0.50	-0.53	0.53	-0.88	-0.84	0.43	-0.18	0.30		
	FN	1.00	-0.58	-0.58		-0.53	-0.53	0.45	-0.45	-0.19	-1.00	-0.25	-0.38	-0.63	-0.51	0.55	0.97	-0.49	-1.00	-0.99	-0.99	-1.00	0.93	-0.58	-0.42	0.35	-0.34	0.12		
	FPP	-0.58	1.00	1.00	-0.58		1.00	-0.99	0.99	-0.51	0.53	-0.48	-0.38	-0.13	-0.23	0.14	-0.45	0.96	0.53	0.55	0.50	0.53	-0.53	0.88	0.84	-0.43	0.18	-0.30		
	FPR	-0.58	1.00	1.00	-0.58	1.00		-0.99	0.99	-0.51	0.53	-0.48	-0.38	-0.13	-0.23	0.14	-0.45	0.96	0.53	0.55	0.50	0.53	-0.53	0.88	0.84	-0.43	0.18	-0.30		
	A	-0.49	0.98	0.98	-0.49	0.98	0.98		1.00	0.56	-0.45	0.53	0.44	0.20	0.30	-0.20	0.38	-0.96	-0.45	-0.47	-0.42	-0.45	0.46	-0.86	-0.84	0.41	-0.16	0.29		
	E	-0.49	0.98	0.98	-0.49	0.98	0.98	1.00		-0.56	0.45	-0.53	0.44	0.20	-0.30	0.20	0.38	0.96	0.45	0.47	0.42	0.45	-0.46	0.86	0.84	-0.41	-0.16	-0.29		
	P	0.15	0.50	0.50	0.15	0.50	0.50	0.57	0.57		0.19	0.99	0.95	0.79	0.90	-0.84	-0.28	-0.42	0.19	0.17	0.22	0.19	-0.13	-0.32	-0.37	0.28	0.10	0.37		
	R	1.00	-0.58	-0.58	1.00	-0.58	-0.58	-0.49	-0.49	0.15		0.25	0.38	0.63	0.51	-0.55	-0.97	0.49	1.00	0.99	0.99	1.00	-0.93	0.58	0.42	-0.35	0.34	-0.12		
	F05	0.20	0.46	0.46	0.20	0.46	0.46	0.53	0.53	0.99	0.20		0.98	0.84	0.93	-0.88	-0.34	-0.40	0.25	0.22	0.28	0.25	-0.19	-0.29	-0.35	0.26	0.09	0.34		
	F1	0.30	0.36	0.36	0.30	0.36	0.36	0.44	0.44	0.96	0.30	0.98		0.92	0.98	-0.94	-0.46	-0.32	0.39	0.36	0.41	0.39	-0.32	-0.20	-0.29	0.19	0.08	0.28		
	F2	0.47	0.17	0.17	0.47	0.17	0.17	0.27	0.27	0.85	0.47	0.88	0.94		0.97	-0.97	-0.69	-0.12	0.63	0.61	0.64	0.63	-0.56	0.00	-0.11	0.02	0.09	0.13		
	SMCC	0.38	0.27	0.27	0.38	0.27	0.27	0.36	0.36	0.92	0.38	0.94	0.98	0.97		-0.98	-0.58	-0.19	0.51	0.48	0.53	0.51	-0.44	-0.07	-0.17	0.12	0.09	0.22		
	DTPpr	0.38	0.23	0.23	0.38	0.23	0.23	0.32	0.32	0.90	0.38	0.93	0.97	0.97	0.99		0.63	0.10	-0.55	-0.52	-0.57	-0.55	0.49	0.00	0.10	-0.09	-0.07	-0.18		
	DTPRate	0.99	-0.58	-0.58	0.99	-0.58	-0.58	-0.48	-0.48	0.16	0.99	0.21	0.30	0.47	0.38	0.38		-0.42	-0.97	-0.95	-0.98	-0.97	0.91	-0.51	-0.35	0.31	-0.30	0.09		
	DTPperc	-0.55	0.99	0.99	-0.55	0.99	0.99	0.99	0.99	0.54	-0.55	0.49	0.40	0.21	0.31	0.28	-0.54		0.49	0.51	0.46	0.49	-0.49	0.85	0.85	-0.37	0.18	-0.25		
	W_AUC	0.99	-0.58	-0.58	0.99	-0.58	-0.58	-0.48	-0.48	0.16	0.99	0.21	0.30	0.47	0.38	0.38	1.00	-0.54		0.99	0.99	0.99	1.00	-0.93	0.58	0.42	-0.35	0.34	-0.11	
	W_AUCB	0.99	-0.59	-0.59	0.99	-0.59	-0.59	-0.50	-0.50	0.15	0.99	0.20	0.30	0.47	0.38	0.38	0.99	-0.55	0.99		0.99	0.99	0.98	0.99	-0.93	0.59	0.44	-0.36	0.34	-0.12
	G_AUC	0.99	-0.58	-0.58	0.99	-0.58	-0.58	-0.49	-0.49	0.16	0.99	0.21	0.30	0.47	0.38	0.38	0.99	-0.54	0.99	0.99		0.99	0.99	0.93	0.56	0.39	-0.34	0.35	-0.10	
	B_AUC	0.99	-0.57	-0.57	0.99	-0.57	-0.57	-0.48	-0.48	0.16	0.99	0.21	0.30	0.47	0.38	0.38	1.00	-0.54	1.00	0.99	0.99		0.99	-0.93	0.58	0.42	-0.35	0.34	-0.11	
	TPC_FN	0.94	-0.63	-0.63	0.94	-0.63	-0.63	-0.55	-0.55	0.06	0.94	0.11	0.21	0.39	0.29	0.30	0.94	-0.60	0.94	0.94	0.94	0.94		-0.53	-0.37	0.54	-0.24	0.28		
	TPC_FP	-0.70	0.81	0.81	-0.70	0.81	0.81	0.78	0.78	0.26	-0.70	0.22	0.14	0.01	0.06	0.04	-0.69	0.80	-0.69	-0.70	-0.70	-0.69	-0.68		0.94	-0.36	0.53	-0.05		
	DTPtpc	-0.40	0.66	0.66	-0.40	0.66	0.66	0.71	0.71	0.34	-0.40	0.32	0.27	0.19	0.22	0.19	-0.40	0.68	-0.40	-0.40	-0.40	-0.40	-0.37	0.80		-0.27	0.43	-0.03		
	APC_FN	0.41	-0.48	-0.48	0.41	-0.48	-0.48	-0.46	-0.46	-0.33	0.41	-0.29	-0.22	-0.10	-0.17	-0.16	0.40	-0.47	0.40	0.41	0.40	0.40	0.40	0.58	-0.35	-0.23	-0.11	0.65		
	APC_FP	-0.51	0.16	0.16	-0.51	0.16	0.16	0.13	0.13	-0.11	-0.51	-0.12	-0.13	-0.15	-0.14	-0.12	-0.52	0.15	-0.52	-0.51	-0.52	-0.52	-0.44	0.63	0.53	-0.13	0.43	0.42		
	DTPapc	0.11	-0.31	-0.31	0.11	-0.31	-0.31	-0.30	-0.30	-0.37	0.11	-0.34	-0.28	-0.16	-0.24	-0.22	0.10	-0.30	0.10	0.11	0.10	0.10	0.22	0.01	0.15	0.60	0.38			

Table 7.12 – SLLD_{Max} event detector: cross dataset pairwise correlations

		Linear																										
		TP	FP	TN	FN	FPP	FPR	A	E	P	R	F05	F1	F2	SMCC	DTPpr	DTPRate	DTPperc	W_AUC	W_AUCB	G_AUC	B_AUC	TPC_FN	TPC_FP	DTPtpc	APC_FN	APC_FP	DTPapc
Ranks	TP		0.58	-0.58	-1.00	0.58	0.58	-0.43	0.43	-0.41	1.00	-0.32	-0.03	0.63	0.05	0.00	-0.98	0.46	1.00	0.99	0.99	1.00	-0.89	0.39	-0.07	-0.28	-0.02	-0.31
	FP	-0.64		-1.00	-0.58	1.00	1.00	-0.97	0.97	-0.91	0.58	-0.88	-0.73	-0.16	-0.66	0.68	-0.52	0.96	0.58	0.59	0.57	0.58	-0.57	0.72	0.40	-0.33	0.04	-0.27
	TN	-0.64	1.00		0.58	-1.00	-1.00	0.97	-0.97	0.91	-0.58	0.88	0.73	0.16	0.66	-0.68	0.52	-0.96	-0.58	-0.59	-0.57	-0.58	0.57	-0.72	-0.40	0.33	-0.04	0.27
	FN	1.00	-0.64	-0.64		-0.58	-0.58	0.43	-0.43	0.41	-1.00	0.32	0.03	-0.63	-0.05	0.00	0.98	-0.46	-1.00	-0.99	-0.99	-1.00	0.89	-0.39	0.07	0.28	0.02	0.31
	FPP	-0.64	1.00	1.00	-0.64		1.00	-0.97	0.97	-0.91	0.58	-0.88	-0.73	-0.16	-0.66	0.68	-0.52	0.96	0.58	0.59	0.57	0.58	-0.57	0.72	0.40	-0.33	0.04	-0.27
	FPR	-0.64	1.00	1.00	-0.64	1.00		-0.97	0.97	-0.91	0.58	-0.88	-0.73	-0.16	-0.66	0.68	-0.52	0.96	0.58	0.59	0.57	0.58	-0.57	0.72	0.40	-0.33	0.04	-0.27
	A	-0.30	0.84	0.84	-0.30	0.84	0.84		1.00	0.91	-0.43	0.91	0.81	0.32	0.75	-0.76	0.37	-0.97	-0.43	-0.44	-0.42	-0.43	0.44	-0.73	-0.47	0.33	-0.07	0.25
	E	-0.30	0.84	0.84	-0.30	0.84	0.84	1.00		-0.91	0.43	-0.91	-0.81	-0.32	-0.75	0.76	-0.37	0.97	0.43	0.44	0.42	0.43	-0.44	0.73	-0.47	0.33	0.07	-0.25
	P	-0.43	0.92	0.92	-0.43	0.92	0.92	0.84	0.84		-0.41	0.98	0.86	0.31	0.82	-0.82	0.36	-0.84	-0.41	-0.42	-0.40	-0.41	0.44	-0.67	-0.38	0.29	-0.08	0.19
	R	1.00	-0.64	-0.64	1.00	-0.64	-0.64	-0.30	-0.30	-0.43		-0.32	-0.03	0.63	0.05	0.00	-0.98	0.46	1.00	0.99	0.99	1.00	-0.89	0.39	-0.07	-0.28	-0.02	-0.31
	F05	-0.27	0.84	0.84	-0.27	0.84	0.84	0.89	0.89	0.96	-0.27		0.92	0.42	0.88	-0.89	0.26	-0.84	-0.32	-0.33	-0.31	-0.32	0.36	-0.66	-0.43	0.25	-0.12	0.14
	F1	0.07	0.56	0.56	0.07	0.56	0.56	0.81	0.81	0.73	0.07	0.86		0.69	0.98	-0.98	-0.02	-0.75	-0.03	-0.05	-0.02	-0.03	0.09	-0.56	-0.52	0.13	-0.21	0.00
	F2	0.63	-0.02	-0.02	0.63	-0.02	-0.02	0.34	0.34	0.18	0.63	0.36	0.69		0.74	-0.70	-0.67	-0.27	0.63	0.62	0.64	0.63	-0.52	-0.14	-0.42	-0.09	-0.16	-0.21
	SMCC	0.13	0.52	0.52	0.13	0.52	0.52	0.77	0.77	0.70	0.13	0.83	0.98	0.73		-0.98	-0.11	-0.68	0.05	0.03	0.06	0.05	0.01	-0.50	-0.51	0.08	-0.20	-0.06
	DTPpr	0.10	0.51	0.51	0.10	0.51	0.51	0.77	0.77	0.69	0.10	0.82	0.98	0.71	0.98		0.06	0.71	0.00	0.02	0.00	0.00	-0.06	0.53	0.51	-0.08	0.26	0.06
	DTPRate	0.99	-0.63	-0.63	0.99	-0.63	-0.63	-0.29	-0.29	-0.43	0.99	-0.26	0.07	0.64	0.13	0.10		-0.39	-0.98	-0.97	-0.98	-0.98	0.87	-0.32	0.14	0.24	0.08	0.31
	DTPperc	-0.30	0.80	0.80	-0.30	0.80	0.80	0.97	0.97	0.80	-0.30	0.85	0.80	0.34	0.76	0.78	-0.29		0.46	0.47	0.45	0.46	-0.45	0.74	0.50	-0.28	0.10	-0.22
W_AUC	0.99	-0.63	-0.63	0.99	-0.63	-0.63	-0.29	-0.29	-0.43	0.99	-0.26	0.07	0.64	0.13	0.10	1.00	-0.29		0.99	0.99	0.99	1.00	-0.89	0.39	-0.07	-0.28	-0.02	-0.31
W_AUCB	0.99	-0.64	-0.64	0.99	-0.64	-0.64	-0.30	-0.30	-0.44	0.99	-0.28	0.06	0.63	0.12	0.09	0.99	-0.30	0.99		0.99	0.99	0.99	-0.89	0.40	-0.06	-0.28	-0.01	-0.31
G_AUC	0.99	-0.63	-0.63	0.99	-0.63	-0.63	-0.29	-0.29	-0.43	0.99	-0.26	0.07	0.64	0.13	0.10	1.00	-0.29	1.00	0.99		0.99	0.99	-0.89	0.38	0.08	-0.27	-0.03	-0.31
B_AUC	0.99	-0.63	-0.63	0.99	-0.63	-0.63	-0.29	-0.29	-0.43	0.99	-0.26	0.07	0.64	0.13	0.10	1.00	-0.29	1.00	0.99	1.00		-0.89	0.39	-0.07	-0.28	-0.02	-0.31	
TPC_FN	0.88	-0.66	-0.66	0.88	-0.66	-0.66	-0.38	-0.38	-0.49	0.88	-0.35	-0.03	0.51	0.02	-0.01	0.88	-0.38	0.88	0.88	0.88	0.88		-0.42	0.08	0.50	-0.01	0.44	
TPC_FP	-0.54	0.85	0.85	-0.54	0.85	0.85	0.71	0.71	0.84	-0.54	0.77	0.53	0.00	0.50	0.50	-0.54	0.68	-0.54	-0.54	-0.54	-0.54	-0.59		0.74	-0.42	0.56	-0.15	
DTPtpc	0.49	-0.14	-0.14	0.49	-0.14	-0.14	0.15	0.15	0.01	0.49	0.11	0.36	0.57	0.37	0.34	0.49	0.16	0.49	0.49	0.49	0.49	0.55	-0.03		-0.05	-0.51	0.19	
APC_FN	0.20	-0.29	-0.29	0.20	-0.29	-0.29	-0.31	-0.31	-0.23	0.20	-0.23	-0.10	0.08	-0.07	-0.07	0.20	-0.28	0.20	0.20	0.20	0.20	0.42	-0.31	0.02		-0.15	0.80	
APC_FP	0.13	-0.32	-0.32	0.13	-0.32	-0.32	-0.25	-0.25	-0.23	0.13	-0.19	-0.05	0.04	-0.06	-0.01	0.12	-0.20	0.12	0.12	0.12	0.12	0.10	0.04	0.31	-0.13		0.21	
DTPapc	0.22	-0.30	-0.30	0.22	-0.30	-0.30	-0.30	-0.30	-0.21	0.22	-0.19	-0.04	0.14	0.00	0.00	0.22	-0.27	0.22	0.22	0.22	0.22	0.38	-0.20	0.15	0.84	0.12		

Table 7.13 – LLD_{Max} event detector: cross dataset pairwise correlations

		Linear																										
		TP	FP	TN	FN	FPP	FPR	A	E	P	R	F05	F1	F2	SMCC	DTPpr	DTPRate	DTPperc	W_AUC	W_AUCB	G_AUC	B_AUC	TPC_FN	TPC_FP	DTPtpc	APC_FN	APC_FP	DTPapc
Ranks	TP		0.63	-0.63	-1.00	0.63	0.63	-0.55	0.55	-0.54	1.00	-0.47	-0.26	0.26	-0.19	0.22	-0.98	0.49	1.00	0.99	0.99	1.00	-0.91	0.55	0.29	-0.24	0.24	-0.23
	FP	-0.67		-1.00	-0.63	1.00	1.00	-0.99	0.99	-0.91	0.63	-0.91	-0.83	-0.41	-0.77	0.78	-0.57	0.94	0.63	0.63	0.62	0.63	-0.67	0.87	0.68	-0.34	0.24	-0.25
	TN	-0.67	1.00		0.63	-1.00	-1.00	0.99	-0.99	0.91	-0.63	0.91	0.83	0.41	0.77	-0.78	0.57	-0.94	-0.63	-0.63	-0.62	-0.63	0.67	-0.87	-0.68	0.34	-0.24	0.25
	FN	1.00	-0.67	-0.67		-0.63	-0.63	0.55	-0.55	0.54	-1.00	0.47	0.26	-0.26	0.19	-0.22	0.98	-0.49	-1.00	-0.99	-0.99	-1.00	0.91	-0.55	-0.29	0.24	-0.24	0.23
	FPP	-0.67	1.00	1.00	-0.67		1.00	-0.99	0.99	-0.91	0.63	-0.91	-0.83	-0.41	-0.77	0.78	-0.57	0.94	0.63	0.63	0.62	0.63	-0.67	0.87	0.68	-0.34	0.24	-0.25
	FPR	-0.67	1.00	1.00	-0.67	1.00		-0.99	0.99	-0.91	0.63	-0.91	-0.83	-0.41	-0.77	0.78	-0.57	0.94	0.63	0.63	0.62	0.63	-0.67	0.87	0.68	-0.34	0.24	-0.25
	A	-0.43	0.89	0.89	-0.43	0.89	0.89		1.00	0.91	-0.55	0.92	0.87	0.50	0.82	-0.82	0.48	-0.95	-0.55	-0.55	-0.54	-0.55	0.61	-0.86	-0.70	0.32	-0.23	0.23
	E	-0.43	0.89	0.89	-0.43	0.89	0.89	1.00		-0.91	0.55	-0.92	-0.87	-0.50	-0.82	0.82	-0.48	0.95	0.55	0.55	0.54	0.55	0.61	0.86	0.70	-0.32	0.23	-0.23
	P	-0.51	0.94	0.94	-0.51	0.94	0.94	0.91	0.91		-0.54	0.99	0.90	0.47	0.88	-0.87	0.49	-0.78	-0.54	-0.55	-0.53	-0.54	0.63	-0.77	-0.54	0.33	-0.22	0.24
	R	1.00	-0.67	-0.67	1.00	-0.67	-0.67	-0.43	-0.43	-0.51		-0.47	-0.26	0.26	-0.19	0.22	-0.98	0.49	1.00	0.99	0.99	1.00	-0.91	0.55	0.29	-0.24	0.24	-0.23
	F05	-0.39	0.88	0.88	-0.39	0.88	0.88	0.94	0.94	0.96	-0.39		0.94	0.55	0.92	-0.92	0.42	-0.79	-0.47	-0.48	-0.47	-0.47	0.57	-0.77	-0.58	0.31	-0.25	0.20
	F1	-0.13	0.67	0.67	-0.13	0.67	0.67	0.86	0.86	0.80	-0.13	0.89		0.76	0.99	-0.98	0.20	-0.77	-0.26	-0.27	-0.25	-0.26	0.37	-0.71	-0.62	0.20	-0.28	0.08
	F2	0.35	0.21	0.21	0.35	0.21	0.21	0.46	0.46	0.37	0.35	0.50	0.75		0.79	-0.78	-0.32	-0.46	0.26	0.25	0.27	0.26	-0.13	-0.35	-0.46	-0.12	-0.19	-0.19
	SMCC	-0.08	0.65	0.65	-0.08	0.65	0.65	0.84	0.84	0.79	-0.08	0.88	0.99	0.77		-0.98	0.13	-0.71	-0.19	-0.20	-0.18	-0.19	0.31	-0.65	-0.58	0.16	-0.26	0.05
	DTPpr	-0.09	0.63	0.63	-0.09	0.63	0.63	0.83	0.83	0.77	-0.09	0.87	0.98	0.77	0.99		-0.15	0.74	0.21	0.22	0.21	0.21	-0.33	0.68	0.60	-0.17	0.31	-0.05
	DTPRate	0.99	-0.67	-0.67	0.99	-0.67	-0.67	-0.43	-0.43	-0.50	0.99	-0.38	-0.12	0.35	-0.08	-0.08		-0.43	-0.98	-0.97	-0.98	-0.98	0.88	-0.48	-0.22	0.20	-0.17	0.22
	DTPperc	-0.41	0.85	0.85	-0.41	0.85	0.85	0.98	0.98	0.87	-0.41	0.91	0.87	0.48	0.84	0.85	-0.41		0.49	0.50	0.49	0.49	-0.53	0.85	0.74	-0.28	0.24	-0.19
	W_AUC	0.99	-0.67	-0.67	0.99	-0.67	-0.67	-0.43	-0.43	-0.50	0.99	-0.38	-0.12	0.35	-0.08	-0.08	1.00	-0.41		0.99	0.99	1.00	-0.91	0.55	0.29	-0.24	0.24	-0.23
	W_AUCB	0.99	-0.67	-0.67	0.99	-0.67	-0.67	-0.44	-0.44	-0.51	0.99	-0.39	-0.13	0.35	-0.09	-0.09	0.99	-0.42	0.99		0.99	0.99	-0.91	0.56	0.30	-0.24	0.25	-0.23
	G_AUC	0.99	-0.67	-0.67	0.99	-0.67	-0.67	-0.43	-0.43	-0.50	0.99	-0.38	-0.12	0.35	-0.08	-0.08	1.00	-0.41	1.00	0.99		0.99	-0.91	0.55	0.28	-0.23	0.23	-0.23
	B_AUC	0.99	-0.67	-0.67	0.99	-0.67	-0.67	-0.43	-0.43	-0.50	0.99	-0.38	-0.12	0.35	-0.08	-0.08	1.00	-0.41	1.00	0.99	1.00		-0.91	0.55	0.29	-0.24	0.24	-0.23
	TPC_FN	0.90	-0.75	-0.75	0.90	-0.75	-0.75	-0.55	-0.55	-0.62	0.90	-0.51	-0.27	0.22	-0.22	-0.23	0.90	-0.53	0.90	0.90	0.90	0.90		-0.59	-0.30	0.43	-0.22	0.37
	TPC_FP	-0.63	0.87	0.87	-0.63	0.87	0.87	0.77	0.77	0.86	-0.63	0.80	0.61	0.18	0.59	0.59	-0.62	0.74	-0.62	-0.63	-0.62	-0.62	-0.67		0.87	-0.30	0.55	-0.09
	DTPtpc	0.13	0.17	0.17	0.13	0.17	0.17	0.35	0.35	0.25	0.13	0.34	0.50	0.58	0.49	0.49	0.13	0.38	0.13	0.13	0.13	0.13	0.14	0.28		-0.10	0.48	0.08
	APC_FN	0.10	-0.32	-0.32	0.10	-0.32	-0.32	-0.33	-0.33	-0.30	0.10	-0.30	-0.17	0.06	-0.15	-0.15	0.10	0.10	0.10	0.10	0.10	0.10	0.31	-0.26	0.00	-0.02	0.84	
	APC_FP	-0.13	-0.06	-0.06	-0.13	-0.06	-0.06	-0.05	-0.05	-0.01	-0.13	0.00	0.05	0.05	0.04	0.08	-0.13	-0.02	-0.13	-0.13	-0.13	-0.13	-0.08	0.25	0.39	0.00		0.29
	DTPapc	0.13	-0.29	-0.29	0.13	-0.29	-0.29	-0.29	-0.29	-0.23	0.13	-0.23	-0.09	0.11	-0.07	-0.06	0.13	-0.26	0.13	0.13	0.13	0.13	0.28	-0.14	0.14	0.86	0.23	

Table 7.14 – SLLD_{Vote} event detector: cross dataset pairwise correlations

		Linear																										
		TP	FP	TN	FN	FPP	FPR	A	E	P	R	F05	F1	F2	SMCC	DTPpr	DTPRate	DTPperc	W_AUC	W_AUCB	G_AUC	B_AUC	TPC_FN	TPC_FP	DTPtpc	APC_FN	APC_FP	DTPapc
Ranks	TP		0.47	-0.47	-1.00	0.47	0.47	-0.43	0.43	-0.52	1.00	-0.47	-0.33	0.01	-0.25	0.24	-0.97	0.33	1.00	0.99	0.99	1.00	-0.90	0.43	0.31	-0.28	0.13	-0.25
	FP	-0.70		-1.00	-0.47	1.00	1.00	-0.99	0.99	-0.85	0.47	-0.87	-0.89	-0.77	-0.85	0.81	-0.40	0.95	0.47	0.48	0.46	0.47	-0.43	0.88	0.80	-0.18	0.18	-0.11
	TN	-0.70	1.00		0.47	-1.00	-1.00	0.99	-0.99	0.85	-0.47	0.87	0.89	0.77	0.85	-0.81	0.40	-0.95	-0.47	-0.48	-0.46	-0.47	0.43	-0.88	-0.80	0.18	-0.18	0.11
	FN	1.00	-0.70	-0.70		-0.47	-0.47	0.43	-0.43	0.52	-1.00	0.47	0.33	-0.01	0.25	-0.24	0.97	-0.33	-1.00	-0.99	-0.99	-1.00	0.90	-0.43	-0.31	0.28	-0.13	0.25
	FPP	-0.70	1.00	1.00	-0.70		1.00	-0.99	0.99	-0.85	0.47	-0.87	-0.89	-0.77	-0.85	0.81	-0.40	0.95	0.47	0.48	0.46	0.47	-0.43	0.88	0.80	-0.18	0.18	-0.11
	FPR	-0.70	1.00	1.00	-0.70	1.00		-0.99	0.99	-0.85	0.47	-0.87	-0.89	-0.77	-0.85	0.81	-0.40	0.95	0.47	0.48	0.46	0.47	-0.43	0.88	0.80	-0.18	0.18	-0.11
	A	-0.53	0.93	0.93	-0.53	0.93	0.93		1.00	-0.85	0.43	0.87	0.89	0.79	0.85	-0.82	0.36	-0.95	-0.43	-0.44	-0.42	-0.43	0.39	-0.88	-0.81	0.18	-0.18	0.10
	E	-0.53	0.93	0.93	-0.53	0.93	0.93	1.00		-0.85	0.43	0.87	0.89	0.79	0.85	-0.82	0.36	0.95	0.43	0.44	0.42	0.43	-0.39	0.88	0.81	-0.18	0.18	-0.10
	P	-0.55	0.94	0.94	-0.55	0.94	0.94	0.93	0.93		-0.52	0.99	0.95	0.72	0.92	-0.89	0.44	-0.69	-0.52	-0.53	-0.51	-0.52	0.51	-0.77	-0.62	0.24	-0.23	0.13
	R	1.00	-0.70	-0.70	1.00	-0.70	-0.70	-0.53	-0.53	-0.55		-0.47	-0.33	0.01	-0.25	0.24	-0.97	0.33	1.00	0.99	0.99	1.00	-0.90	0.43	0.31	-0.28	0.13	-0.25
	F05	-0.47	0.90	0.90	-0.47	0.90	0.90	0.93	0.93	0.98	-0.47		0.97	0.78	0.95	-0.92	0.39	-0.73	-0.47	-0.48	-0.46	-0.46	0.45	-0.79	-0.65	0.21	-0.24	0.09
	F1	-0.25	0.74	0.74	-0.25	0.74	0.74	0.86	0.86	0.86	-0.25	0.92		0.89	0.99	-0.96	0.24	-0.77	-0.33	-0.34	-0.32	-0.33	0.32	-0.80	-0.68	0.14	-0.27	0.02
	F2	0.18	0.34	0.34	0.18	0.34	0.34	0.53	0.53	0.50	0.18	0.59	0.79		0.92	-0.91	-0.10	-0.74	0.01	0.00	0.02	0.01	0.00	-0.69	-0.63	-0.01	-0.28	-0.14
	SMCC	-0.19	0.70	0.70	-0.19	0.70	0.70	0.83	0.83	0.84	-0.19	0.90	0.99	0.80		-0.98	0.17	-0.74	-0.25	-0.27	-0.24	-0.25	0.25	-0.76	-0.66	0.11	-0.29	-0.01
	DTPpr	-0.21	0.70	0.70	-0.21	0.70	0.70	0.82	0.82	0.83	-0.21	0.89	0.98	0.80	0.98		-0.15	0.72	0.24	0.25	0.23	0.24	-0.24	0.73	0.62	-0.09	0.33	0.05
	DTPRate	0.99	-0.70	-0.70	0.99	-0.70	-0.70	-0.52	-0.52	-0.55	0.99	-0.46	-0.24	0.18	-0.19	-0.21		-0.27	-0.97	-0.97	-0.98	-0.97	0.88	-0.35	-0.25	0.26	-0.06	0.26
	DTPperc	-0.56	0.93	0.93	-0.56	0.93	0.93	0.98	0.98	0.92	-0.56	0.92	0.85	0.51	0.81	0.81	-0.55		0.33	0.34	0.32	0.33	-0.30	0.81	0.80	-0.14	0.13	-0.09
W_AUC	0.99	-0.70	-0.70	0.99	-0.70	-0.70	-0.52	-0.52	-0.55	0.99	-0.46	-0.24	0.19	-0.19	-0.21	1.00	-0.55		0.99	0.99	0.99	1.00	-0.90	0.43	0.31	-0.28	0.13	-0.25
W_AUCB	0.99	-0.71	-0.71	0.99	-0.71	-0.71	-0.54	-0.54	-0.56	0.99	-0.47	-0.26	0.17	-0.20	-0.22	0.99	-0.56	0.99		0.99	0.99	0.99	-0.90	0.44	0.32	-0.28	0.14	-0.24
G_AUC	0.99	-0.70	-0.70	0.99	-0.70	-0.70	-0.52	-0.52	-0.55	0.99	-0.46	-0.24	0.18	-0.19	-0.21	1.00	-0.55	1.00	0.99		0.99	0.99	-0.90	0.42	0.30	-0.27	0.12	-0.25
B_AUC	0.99	-0.70	-0.70	0.99	-0.70	-0.70	-0.52	-0.52	-0.55	0.99	-0.46	-0.24	0.19	-0.19	-0.21	1.00	-0.55	1.00	0.99	1.00		-0.90	0.43	0.31	-0.28	0.13	-0.25	
TPC_FN	-0.92	-0.72	-0.72	0.92	-0.72	-0.72	-0.57	-0.57	-0.59	0.92	-0.51	-0.31	0.10	-0.26	-0.28	0.92	-0.60	0.92	0.92	0.92	0.92		-0.40	-0.28	0.48	-0.15	0.38	
TPC_FP	-0.65	0.99	0.99	-0.65	0.99	0.99	0.85	0.85	0.89	-0.65	0.86	0.72	0.35	0.69	0.69	-0.65	0.85	-0.65	-0.66	-0.65	-0.65	-0.68		0.95	-0.28	0.21	0.42	-0.02
DTPtpc	0.03	0.30	0.30	0.03	0.30	0.30	0.45	0.45	0.38	0.03	0.44	0.57	0.64	0.57	0.54	0.03	0.44	0.03	0.03	0.03	0.03	0.06	0.38		-0.14	0.33	-0.01	
APC_FN	0.25	-0.32	-0.32	0.25	-0.32	-0.32	-0.32	-0.32	-0.32	0.25	-0.24	0.16	0.03	-0.13	-0.11	0.25	0.30	0.25	0.25	0.25	0.25	0.44	-0.31	-0.09		-0.12	0.77	
APC_FP	-0.13	-0.01	-0.01	-0.13	-0.01	-0.01	0.00	0.00	0.05	-0.13	0.07	0.12	0.10	0.12	0.17	0.14	0.03	-0.14	-0.13	-0.14	-0.14	-0.15	0.26	0.35	-0.10		0.30	
DTPapc	-0.18	-0.25	-0.25	0.18	-0.25	-0.24	-0.24	-0.24	-0.18	0.18	-0.15	-0.05	0.12	-0.02	0.00	0.17	-0.21	0.17	0.18	0.17	0.17	0.30	-0.15	0.11	0.78	0.10	0.23	

Table 7.15 – LLD_{Vote} event detector: cross dataset pairwise correlations

		Linear																											
		TP	FP	TN	FN	FPP	FPR	A	E	P	R	F05	F1	F2	SMCC	DTpPr	DTpPerc	W_AUC	W_AUCB	G_AUC	B_AUC	TPC_FN	TPC_FP	DTpPrc	APC_FN	APC_FP	DTPapc		
Ranks	TP		0.50	-0.50	-1.00	0.50	0.50	-0.46	0.46	-0.48	1.00	-0.43	-0.30	0.09	-0.20	0.18	-0.97	0.35	1.00	0.99	0.99	1.00	-0.92	0.46	0.31	-0.25	0.17	-0.18	
	FP	-0.68		-1.00	0.50	1.00	0.50	-0.99	0.99	-0.82	0.50	-0.84	-0.85	-0.69	-0.79	0.74	-0.43	0.93	0.50	0.51	0.49	0.50	-0.46	0.88	0.80	-0.17	0.09	-0.12	
	TN	-0.68	1.00		0.50	-1.00	-0.99	0.99	-0.82	-0.50	0.84	0.85	-0.69	0.79	-0.74	-0.43	-0.93	-0.50	-0.51	-0.49	-0.50	0.46	-0.88	-0.80	0.17	-0.09	0.12		
	FN	1.00	-0.68	-0.68		-0.50	-0.50	0.46	-0.46	-0.48	-1.00	-0.43	0.30	-0.09	0.20	-0.18	0.97	-0.35	-1.00	-0.99	-0.99	1.00	0.92	-0.46	-0.31	-0.25	-0.17	0.11	
	FPP	-0.68	1.00	1.00	-0.68		1.00	-0.99	0.99	-0.82	0.50	-0.84	-0.85	-0.69	-0.79	0.74	-0.43	0.93	0.50	0.51	0.49	0.50	-0.46	0.88	0.80	-0.17	0.09	-0.12	
	FPR	-0.68	1.00	1.00	-0.68	1.00		-0.99	0.99	-0.82	0.50	-0.84	-0.85	-0.69	-0.79	0.74	-0.43	0.93	0.50	0.51	0.49	0.50	-0.46	0.88	0.80	-0.17	0.09	-0.12	
	A	-0.54	0.96	0.96	-0.54	0.96	0.96		-1.00	0.82	-0.46	0.84	0.86	0.72	-0.80	-0.75	0.39	-0.94	-0.46	-0.47	-0.45	-0.46	-0.43	-0.88	-0.81	-0.17	-0.09	-0.12	
	E	-0.54	0.96	0.96	-0.54	0.96	0.96	1.00		-0.82	-0.46	-0.84	-0.86	-0.72	-0.80	-0.75	-0.39	0.94	0.46	0.47	0.45	-0.46	-0.43	0.88	0.81	-0.17	0.09	-0.12	
	P	-0.49	0.92	0.92	-0.49	0.92	0.92	0.92	-0.48		0.99	0.95	0.69	0.92	-0.88	-0.42	-0.64	-0.48	-0.49	-0.48	-0.48	0.50	-0.72	-0.56	0.22	-0.14	-0.16		
	R	1.00	-0.68	-0.68	1.00	-0.68	-0.54	-0.54	-0.49	-0.43	-0.30	-0.09	-0.20	0.18	-0.97	0.35	1.00	0.99	0.99	1.00	-0.92	0.46	0.31	-0.25	0.17	-0.18			
	F05	-0.42	0.89	0.89	-0.42	0.89	0.89	0.92	0.92	0.99	-0.42	0.97	0.75	0.95	-0.91	0.37	-0.67	-0.43	-0.44	-0.43	-0.43	0.44	-0.74	-0.58	0.20	-0.15	0.13		
	F1	-0.42	0.75	0.75	-0.42	0.75	0.75	0.85	0.85	0.89	-0.22	0.94	0.87	0.98	-0.95	0.22	-0.72	-0.30	-0.31	-0.29	-0.30	0.31	-0.75	-0.62	0.13	-0.17	0.06		
	F2	0.24	0.31	0.31	0.24	0.31	0.31	0.46	0.46	0.51	0.24	0.58	0.77	0.90	-0.89	-0.17	-0.66	0.09	0.08	0.10	0.09	-0.07	-0.61	-0.57	-0.04	-0.15	-0.11		
	SMCC	-0.13	0.69	0.69	-0.13	0.69	0.69	0.79	0.79	0.85	-0.13	0.90	0.98	0.82	-0.98	0.12	-0.67	-0.20	-0.21	-0.19	-0.20	0.22	-0.70	-0.58	0.11	-0.18	0.03		
	DTpPr	-0.14	0.67	0.67	-0.14	0.67	0.67	0.77	0.77	0.84	-0.14	0.89	0.96	0.79	0.98	-0.10	0.63	0.18	0.19	0.17	0.18	-0.20	0.65	0.54	-0.09	0.22	0.00		
	DTpRate	0.99	-0.68	-0.68	0.99	-0.68	-0.68	-0.54	-0.54	-0.49	0.99	-0.41	-0.21	0.24	-0.13	-0.13	-0.29	-0.97	-0.97	-0.98	-0.97	0.90	-0.39	-0.25	0.23	-0.12	-0.19		
DTpPerc	0.57	0.95	0.95	0.57	0.95	0.95	0.95	0.99	0.91	0.57	0.91	0.83	0.44	0.78	0.76	-0.56	0.35	0.36	0.34	0.35	-0.31	0.84	0.86	-0.12	0.06	-0.08			
W_AUC	0.99	-0.68	-0.68	0.99	-0.68	-0.68	-0.54	-0.54	-0.49	0.99	-0.41	-0.21	0.24	-0.13	-0.13	0.10	-0.56		0.99	0.99	1.00	-0.92	0.46	0.31	-0.25	0.17	-0.18		
W_AUCB	0.99	-0.69	-0.69	0.99	-0.69	-0.69	-0.55	-0.55	-0.49	0.99	-0.42	-0.21	0.24	-0.14	-0.14	0.99	-0.57	0.99		0.99	0.99	-0.92	0.47	0.32	-0.26	0.18	-0.18		
G_AUC	0.99	-0.68	-0.68	0.99	-0.68	-0.68	-0.54	-0.54	-0.49	0.99	-0.41	-0.21	0.24	-0.13	-0.13	1.00	-0.56	1.00	0.99		0.99	-0.92	0.45	0.30	-0.25	0.17	-0.18		
B_AUC	0.99	-0.68	-0.68	0.99	-0.68	-0.68	-0.54	-0.54	-0.49	0.99	-0.41	-0.21	0.24	-0.13	-0.13	1.00	-0.56	1.00	0.99	1.00		-0.92	0.46	0.31	-0.25	0.17	-0.18		
TPC_FN	0.93	-0.72	-0.72	0.93	-0.72	-0.72	-0.60	-0.60	-0.55	0.93	-0.48	-0.29	0.16	-0.21	-0.23	0.93	-0.62	0.93	0.93	0.93		-0.43	-0.28	0.45	-0.16	-0.33			
TPC_FP	-0.64	0.85	0.85	-0.64	0.85	0.85	0.82	0.82	0.81	-0.64	0.79	0.67	0.26	0.61	0.60	0.64	0.82	-0.64	-0.65	-0.64	-0.64	-0.67	0.93		-0.22	0.34	-0.04		
DTpPrc	-0.08	0.35	0.35	-0.08	0.35	0.35	0.45	0.45	0.40	-0.08	0.44	0.53	0.54	0.53	0.48	0.07	0.34	-0.07	-0.08	-0.07	-0.07	-0.06	0.51	-0.14	0.22	-0.02			
APC_FN	0.21	0.31	0.31	0.21	0.31	0.31	-0.33	-0.33	-0.24	0.21	-0.23	-0.15	0.05	-0.13	-0.12	0.21	-0.31	0.21	0.21	0.21	0.21	-0.40	-0.31	-0.09	-0.10	0.74			
APC_FP	-0.18	0.02	0.02	-0.18	0.02	0.02	0.02	0.02	0.08	-0.18	0.10	0.11	0.03	0.10	0.16	-0.18	0.04	-0.18	-0.18	-0.18	-0.17	0.38	0.34	-0.10	0.30		0.36		
DTPapc	0.16	0.31	0.31	0.16	0.31	0.31	-0.30	-0.30	-0.23	0.16	-0.21	-0.12	0.07	-0.09	-0.06	0.16	-0.29	0.16	0.16	0.16	0.16	0.29	-0.11	0.12	0.73	0.30			

Table 7.16 – K-NN classifier: cross dataset pairwise correlations for micro-average metrics

[illegible]

Table 7.17 – KStar classifier: cross dataset pairwise correlations for micro-average metrics

[illegible]

Table 7.18 – LWL with Naïve Bayes classifier: cross dataset pairwise correlations for micro-average metrics

[illegible]

Table 7.19 – Decision Trees classifier: cross dataset pairwise correlations for micro-average metrics

[illegible]

Table 7.20 – ANN classifier: cross dataset pairwise correlations for micro-average metrics

[illegible]

Table 7.21 – SVM classifier: cross dataset pairwise correlations for micro-average metrics

		Linear																	
		TP	FP	TN	FN	A	E	FPR	FPP	SMCC	F1	DTPpr	DTPperc	DTPrate	WAUC	GAUC	WAUCB	MAE	RMSE
Rank	TP		-1,00	1,00	-1,00	1,00	-1,00	-1,00	-1,00	1,00	1,00	-0,97	-0,97	-0,97	1,00	0,99	0,99	-1,00	-0,99
	FP	1,00		-1,00	1,00	-1,00	1,00	1,00	1,00	-1,00	-1,00	0,97	0,97	0,97	-1,00	-0,99	-0,99	1,00	0,99
	TN	1,00	1,00		-1,00	1,00	-1,00	-1,00	-1,00	1,00	1,00	-0,97	-0,97	-0,97	1,00	0,99	0,99	-1,00	-0,99
	FN	1,00	1,00	1,00		-1,00	1,00	1,00	1,00	-1,00	-1,00	0,97	0,97	0,97	-1,00	-0,99	-0,99	1,00	0,99
	A	1,00	1,00	1,00	1,00		-1,00	-1,00	-1,00	1,00	1,00	-0,97	-0,97	-0,97	1,00	0,99	0,99	-1,00	-0,99
	E	1,00	1,00	1,00	1,00	1,00		1,00	1,00	-1,00	-1,00	0,97	0,97	0,97	-1,00	-0,99	-0,99	1,00	0,99
	FPR	1,00	1,00	1,00	1,00	1,00	1,00		1,00	-1,00	-1,00	0,97	0,97	0,97	-1,00	-0,99	-0,99	1,00	0,99
	FPP	1,00	1,00	1,00	1,00	1,00	1,00	1,00		-1,00	-1,00	0,97	0,97	0,97	-1,00	-0,99	-0,99	1,00	0,99
	SMCC	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00		1,00	-0,97	-0,97	-0,97	1,00	0,99	0,99	-1,00	-0,99
	F1	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00		-0,97	-0,97	-0,97	1,00	0,99	0,99	-1,00	-0,99
	DTPpr	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00		1,00	1,00	-0,97	-0,99	-0,94	0,97	0,93
	DTPperc	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00		1,00	-0,97	-0,99	-0,94	0,97	0,93
	DTPrate	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00		-0,97	-0,99	-0,94	0,97	0,93
	WAUC	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00		0,99	0,99	-1,00	-0,99
	GAUC	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00		0,96	-0,99	-0,96
	WAUCB	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00		-0,99	-0,99
MAE	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00		0,99	
RMSE	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00		

Table 7.22 – K-NN classifier: cross dataset pairwise correlations for unweighted macro-average metrics

Rank	Linear																			
	TP	FP	TN	FN	P	R	A	E	FPR	FPP	SMCC	F1	F05	F2	DTPpr	DTPperc	DTPrate	WAUC	GAUC	WAUCB
TP		-1,00	1,00	-1,00	0,87	0,95	0,98	-0,98	-0,91	-0,61	0,95	0,93	0,90	0,95	-0,89	-0,71	-0,92	0,96	0,92	0,96
FP	1,00		-1,00	1,00	-0,87	-0,95	-0,98	0,98	0,91	0,61	-0,95	-0,93	-0,90	-0,95	0,89	0,71	0,92	-0,96	-0,92	-0,96
TN	1,00	1,00		-1,00	0,87	0,95	0,98	-0,98	-0,91	-0,61	0,95	0,93	0,90	0,95	-0,89	-0,71	-0,92	0,96	0,92	0,96
FN	1,00	1,00	1,00		-0,87	-0,95	-0,98	0,98	0,91	0,61	-0,95	-0,93	-0,90	-0,95	0,89	0,71	0,92	-0,96	-0,92	-0,96
P	0,84	0,84	0,84	0,84		0,92	0,88	-0,88	-0,83	-0,54	0,96	0,98	0,99	0,95	-0,97	-0,68	-0,94	0,92	0,95	0,90
R	0,93	0,93	0,93	0,93	0,90		0,94	-0,94	-0,87	-0,55	0,98	0,98	0,95	0,99	-0,95	-0,70	-0,97	0,99	0,98	0,99
A	0,98	0,98	0,98	0,98	0,85	0,92		-1,00	-0,96	-0,66	0,95	0,92	0,90	0,94	-0,89	-0,74	-0,91	0,95	0,91	0,95
E	0,98	0,98	0,98	0,98	0,85	0,92	0,99		0,96	0,66	-0,95	-0,92	-0,90	-0,94	0,89	0,74	0,91	-0,95	-0,91	-0,95
FPR	0,92	0,92	0,92	0,92	0,81	0,86	0,96	0,96		0,67	-0,89	-0,86	-0,85	-0,87	0,83	0,70	0,84	-0,89	-0,85	-0,89
FPP	0,66	0,66	0,66	0,66	0,56	0,58	0,70	0,70	0,73		-0,59	-0,56	-0,55	-0,56	0,52	0,86	0,51	-0,57	-0,52	-0,57
SMCC	0,93	0,93	0,93	0,93	0,95	0,98	0,93	0,93	0,88	0,62		0,99	0,98	0,99	-0,97	-0,71	-0,97	0,99	0,98	0,98
F1	0,91	0,91	0,91	0,91	0,97	0,97	0,90	0,90	0,85	0,58	0,99		0,99	0,99	-0,98	-0,70	-0,98	0,98	0,98	0,96
F05	0,87	0,87	0,87	0,87	0,99	0,93	0,88	0,88	0,83	0,57	0,97	0,98		0,97	-0,98	-0,69	-0,96	0,95	0,97	0,93
F2	0,93	0,93	0,93	0,93	0,93	0,99	0,92	0,92	0,86	0,58	0,99	0,98	0,96		-0,97	-0,70	-0,98	0,99	0,99	0,98
DTPpr	0,85	0,85	0,85	0,85	0,96	0,94	0,85	0,85	0,79	0,54	0,96	0,97	0,97	0,96		0,70	0,98	-0,95	-0,99	-0,93
DTPperc	0,77	0,77	0,77	0,77	0,70	0,74	0,79	0,79	0,75	0,87	0,76	0,74	0,72	0,75	0,73		0,70	-0,71	-0,70	-0,70
DTPrate	0,87	0,87	0,87	0,87	0,92	0,97	0,86	0,86	0,80	0,53	0,96	0,96	0,94	0,97	0,97	0,73		-0,97	-0,99	-0,96
WAUC	0,94	0,94	0,94	0,94	0,90	0,99	0,93	0,93	0,88	0,60	0,98	0,97	0,94	0,99	0,94	0,75	0,96		0,98	0,99
GAUC	0,89	0,89	0,89	0,89	0,94	0,97	0,88	0,88	0,82	0,54	0,97	0,98	0,96	0,98	0,98	0,73	0,99	0,97		0,96
WAUCB	0,94	0,94	0,94	0,94	0,88	0,98	0,93	0,93	0,88	0,60	0,97	0,95	0,91	0,98	0,91	0,75	0,94	0,99	0,95	
MAE	0,94	0,94	0,94	0,94	0,82	0,90	0,93	0,93	0,88	0,59	0,90	0,88	0,85	0,90	0,83	0,71	0,85	0,91	0,87	0,91
RMSE	0,89	0,89	0,89	0,89	0,71	0,80	0,89	0,89	0,83	0,64	0,80	0,77	0,74	0,79	0,71	0,71	0,73	0,81	0,75	0,82

Table 7.23 – KStar classifier: cross dataset pairwise correlations for unweighted macro-average metrics

	Linear																						
	TP	FP	TN	FN	P	R	A	E	FPR	FPP	SMCC	F1	F05	F2	DTPpr	DTPperc	DTPrate	WAUC	GAUC	WAUCB	MAE	RMSE	
Rank	TP	-0,98	0,62	-0,98	0,92	0,95	0,98	-0,98	-0,81	-0,62	0,95	0,94	0,93	0,95	-0,93	-0,85	-0,94	0,95	0,94	0,95	-0,93	-0,85	
	FP	0,98		-0,47	1,00	-0,93	-0,95	-0,98	0,98	0,83	0,62	-0,95	-0,95	-0,94	-0,95	0,94	0,86	0,94	-0,96	-0,94	-0,96	0,93	0,84
	TN	0,99	0,97		-0,47	0,47	0,52	0,56	-0,56	-0,37	-0,29	0,51	0,50	0,49	0,51	-0,50	-0,44	-0,50	0,51	0,50	0,51	-0,50	-0,48
	FN	0,98	1,00	0,97		-0,93	-0,95	-0,98	0,98	0,83	0,62	-0,95	-0,95	-0,94	-0,95	0,94	0,86	0,94	-0,96	-0,94	-0,96	0,93	0,84
	P	0,91	0,91	0,90	0,91		0,97	0,93	-0,93	-0,83	-0,54	0,98	0,99	0,99	0,98	-0,99	-0,80	-0,98	0,97	0,98	0,96	-0,94	-0,78
	R	0,95	0,95	0,94	0,95	0,96		0,95	-0,95	-0,83	-0,53	0,99	0,99	0,98	0,99	-0,98	-0,80	-0,98	0,99	0,99	0,99	-0,95	-0,82
	A	0,98	0,98	0,98	0,98	0,92	0,95		-1,00	-0,87	-0,65	0,95	0,95	0,94	0,95	-0,93	-0,87	-0,94	0,96	0,94	0,95	-0,93	-0,85
	E	0,98	0,98	0,98	0,98	0,92	0,95	0,99		0,87	0,65	-0,95	-0,95	-0,94	-0,95	0,93	0,87	0,94	-0,96	-0,94	-0,95	0,93	0,85
	FPR	0,82	0,84	0,82	0,84	0,83	0,84	0,86	0,86		0,70	-0,85	-0,83	-0,83	-0,84	0,83	0,73	0,83	-0,84	-0,83	-0,84	0,82	0,74
	FPP	0,61	0,62	0,61	0,62	0,54	0,53	0,63	0,63	0,70		-0,56	-0,54	-0,54	-0,54	0,52	0,72	0,51	-0,54	-0,51	-0,54	0,52	0,60
	SMCC	0,94	0,95	0,94	0,95	0,98	0,99	0,95	0,96	0,85	0,55		0,99	0,99	0,99	-0,99	-0,81	-0,98	0,99	0,99	0,99	-0,96	-0,83
	F1	0,94	0,94	0,93	0,94	0,98	0,98	0,94	0,94	0,84	0,54	0,99		0,99	0,99	-0,99	-0,81	-0,99	0,99	0,99	0,98	-0,95	-0,80
	F05	0,92	0,92	0,92	0,92	0,99	0,97	0,93	0,93	0,83	0,54	0,98	0,99		0,99	-0,99	-0,81	-0,98	0,98	0,99	0,97	-0,95	-0,79
	F2	0,94	0,95	0,94	0,95	0,97	0,99	0,95	0,95	0,84	0,54	0,99	0,99	0,98		-0,99	-0,81	-0,99	0,99	0,99	0,99	-0,95	-0,81
	DTPpr	0,92	0,92	0,92	0,92	0,98	0,98	0,93	0,93	0,83	0,53	0,98	0,99	0,99	0,98		0,82	0,99	-0,98	-0,99	-0,97	0,95	0,77
	DTPperc	0,89	0,90	0,89	0,90	0,85	0,87	0,90	0,90	0,77	0,72	0,88	0,87	0,86	0,87	0,86	0,83	0,83	-0,81	-0,82	-0,81	0,78	0,61
	DTPrate	0,93	0,93	0,93	0,93	0,97	0,99	0,99	0,94	0,94	0,83	0,52	0,98	0,98	0,97	0,99	0,99	0,86		-0,98	-0,99	-0,98	0,95
WAUC	0,95	0,95	0,95	0,95	0,96	0,99	0,96	0,96	0,85	0,54	0,99	0,98	0,97	0,99	0,97	0,87	0,98		0,99	0,99	-0,95	-0,83	
GAUC	0,93	0,93	0,93	0,93	0,97	0,99	0,99	0,94	0,94	0,83	0,52	0,99	0,99	0,98	0,99	0,99	0,86	0,99	0,99		0,98	-0,95	-0,78
WAUCB	0,95	0,95	0,95	0,95	0,95	0,99	0,99	0,95	0,96	0,85	0,54	0,98	0,98	0,96	0,99	0,97	0,87	0,98	0,99	0,98		-0,95	-0,83
MAE	0,92	0,92	0,91	0,92	0,92	0,94	0,92	0,93	0,82	0,51	0,94	0,94	0,93	0,94	0,93	0,84	0,94	0,94	0,94	0,94			0,80
RMSE	0,88	0,88	0,88	0,88	0,82	0,85	0,88	0,88	0,78	0,60	0,86	0,84	0,83	0,85	0,83	0,77	0,84	0,85	0,84	0,85	0,81		

Table 7.24 – LWL with Naïve Bayes classifier: cross dataset pairwise correlations for unweighted macro-average metrics

		Linear																					
		TP	FP	TN	FN	P	R	A	E	FPR	FPP	SMCC	F1	F05	F2	DTPpr	DTPperc	DTPrate	WAUC	GAUC	WAUCB	MAE	RMSE
Rank	TP		-1,00	1,00	-1,00	0,90	0,96	0,98	-0,98	-0,92	-0,66	0,96	0,94	0,92	0,96	-0,92	-0,75	-0,94	0,97	0,94	0,96	-0,94	-0,92
	FP	1,00		-1,00	1,00	-0,90	-0,96	-0,98	0,98	0,92	0,66	-0,96	-0,94	-0,92	-0,96	0,92	0,75	0,94	-0,97	-0,94	-0,96	0,94	0,92
	TN	1,00	1,00		-1,00	0,90	0,96	0,98	-0,98	-0,92	-0,66	0,96	0,94	0,92	0,96	-0,92	-0,75	-0,94	0,97	0,94	0,96	-0,94	-0,92
	FN	1,00	1,00	1,00		-0,90	-0,96	-0,98	0,98	0,92	0,66	-0,96	-0,94	-0,92	-0,96	0,92	0,75	0,94	-0,97	-0,94	-0,96	0,94	0,92
	P	0,89	0,89	0,89	0,89		0,94	0,91	-0,91	-0,87	-0,61	0,97	0,98	0,99	0,97	-0,98	-0,74	-0,96	0,94	0,97	0,93	-0,91	-0,80
	R	0,96	0,96	0,96	0,96	0,94		0,95	-0,95	-0,90	-0,61	0,99	0,98	0,96	0,99	-0,97	-0,74	-0,98	0,99	0,98	0,99	-0,94	-0,88
	A	0,98	0,98	0,98	0,98	0,90	0,95		-1,00	-0,96	-0,71	0,96	0,94	0,92	0,95	-0,92	-0,78	-0,93	0,96	0,94	0,96	-0,94	-0,91
	E	0,98	0,98	0,98	0,98	0,90	0,95	0,99		0,96	0,71	-0,96	-0,94	-0,92	-0,95	0,92	0,78	0,93	-0,96	-0,94	-0,96	0,94	0,91
	FPR	0,92	0,92	0,92	0,92	0,86	0,90	0,96	0,96		0,72	-0,91	-0,89	-0,88	-0,90	0,87	0,75	0,88	-0,92	-0,89	-0,91	0,89	0,84
	FPP	0,69	0,69	0,69	0,69	0,63	0,63	0,73	0,73	0,75		-0,65	-0,62	-0,62	-0,62	0,59	0,87	0,59	-0,63	-0,59	-0,63	0,64	0,61
	SMCC	0,95	0,95	0,95	0,95	0,97	0,98	0,95	0,95	0,91	0,67		0,99	0,98	0,99	-0,98	-0,76	-0,98	0,99	0,99	0,98	-0,94	-0,87
	F1	0,93	0,93	0,93	0,93	0,98	0,98	0,93	0,93	0,89	0,64	0,99		0,99	0,99	-0,99	-0,76	-0,98	0,98	0,99	0,97	-0,93	-0,84
	F05	0,91	0,91	0,91	0,91	0,99	0,96	0,92	0,92	0,87	0,64	0,98	0,99		0,98	-0,99	-0,75	-0,97	0,96	0,98	0,95	-0,92	-0,82
	F2	0,95	0,95	0,95	0,95	0,96	0,99	0,95	0,95	0,90	0,64	0,99	0,99	0,97		-0,98	-0,75	-0,99	0,99	0,99	0,98	-0,94	-0,86
	DTPpr	0,90	0,90	0,90	0,90	0,98	0,96	0,90	0,90	0,85	0,61	0,97	0,98	0,98	0,97		0,76	0,99	-0,96	-0,99	-0,95	0,91	0,81
	DTPperc	0,84	0,84	0,84	0,84	0,80	0,82	0,85	0,85	0,82	0,88	0,84	0,82	0,81	0,83	0,82		0,75	-0,75	-0,75	-0,75	0,74	0,64
DTPrate	0,92	0,92	0,92	0,92	0,95	0,98	0,92	0,92	0,87	0,61	0,98	0,98	0,96	0,98	0,98	0,82		-0,98	-0,99	-0,97	0,92	0,83	
WAUC	0,96	0,96	0,96	0,96	0,94	0,99	0,96	0,96	0,91	0,65	0,99	0,98	0,96	0,99	0,96	0,83	0,97		0,98	0,99	-0,94	-0,88	
GAUC	0,93	0,93	0,93	0,93	0,96	0,98	0,92	0,92	0,87	0,61	0,98	0,99	0,97	0,99	0,99	0,82	0,99	0,98		0,97	-0,92	-0,84	
WAUCB	0,96	0,96	0,96	0,96	0,92	0,99	0,96	0,96	0,91	0,65	0,97	0,96	0,94	0,98	0,94	0,83	0,96	0,99	0,97		-0,94	-0,89	
MAE	0,94	0,94	0,94	0,94	0,89	0,93	0,94	0,94	0,89	0,65	0,93	0,92	0,90	0,93	0,89	0,80	0,90	0,93	0,91	0,93			
RMSE	0,93	0,93	0,93	0,93	0,80	0,88	0,92	0,92	0,87	0,62	0,87	0,85	0,82	0,87	0,81	0,76	0,84	0,88	0,84	0,89	0,89		

Table 7.25 – Decision Trees classifier: cross dataset pairwise correlations for unweighted macro-average metrics

	Linear																						
	TP	FP	TN	FN	P	R	A	E	FPR	FPP	SMCC	F1	F05	F2	DTPpr	DTPperc	DTPrate	WAUC	GAUC	WAUCB	MAE	RMSE	
Rank	TP	-1,00	1,00	-1,00	0,72	0,86	0,95	-0,95	-0,83	-0,60	0,87	0,82	0,76	0,85	-0,74	-0,52	-0,77	0,88	0,78	0,89	-0,94	-0,90	
	FP	1,00		-1,00	1,00	-0,72	-0,86	-0,95	0,95	0,83	0,60	-0,87	-0,82	-0,76	-0,85	0,74	0,52	0,77	-0,88	-0,78	-0,89	0,94	0,90
	TN	1,00	1,00		-1,00	0,72	0,86	0,95	-0,95	-0,83	-0,60	0,87	0,82	0,76	0,85	-0,74	-0,52	-0,77	0,88	0,78	0,89	-0,94	-0,90
	FN	1,00	1,00	1,00		-0,72	-0,86	-0,95	0,95	0,83	0,60	-0,87	-0,82	-0,76	-0,85	0,74	0,52	0,77	-0,88	-0,78	-0,89	0,94	0,90
	P	0,69	0,69	0,69	0,69		0,84	0,75	-0,75	-0,71	-0,52	0,94	0,96	0,99	0,90	-0,94	-0,40	-0,87	0,85	0,89	0,79	-0,70	-0,61
	R	0,85	0,85	0,85	0,85	0,82		0,82	-0,82	-0,71	-0,49	0,96	0,95	0,89	0,99	-0,91	-0,45	-0,94	0,99	0,96	0,98	-0,83	-0,76
	A	0,94	0,94	0,94	0,94	0,73	0,80		-1,00	-0,95	-0,71	0,87	0,81	0,78	0,83	-0,73	-0,58	-0,74	0,86	0,75	0,84	-0,91	-0,87
	E	0,94	0,94	0,94	0,94	0,73	0,80	0,99		0,95	0,71	-0,87	-0,81	-0,78	-0,83	0,73	0,58	0,74	-0,86	-0,75	-0,84	0,91	0,87
	FPR	0,81	0,81	0,81	0,81	0,70	0,68	0,94	0,94		0,73	-0,80	-0,74	-0,73	-0,73	0,66	0,56	0,65	-0,77	-0,67	-0,73	0,81	0,75
	FPP	0,58	0,58	0,58	0,58	0,51	0,46	0,70	0,70	0,73		-0,58	-0,52	-0,52	-0,51	0,46	0,87	0,43	-0,53	-0,43	-0,51	0,54	0,56
	SMCC	0,85	0,85	0,85	0,85	0,92	0,95	0,85	0,85	0,78	0,56		0,99	0,96	0,98	-0,94	-0,49	-0,93	0,97	0,95	0,93	-0,84	-0,76
	F1	0,81	0,81	0,81	0,81	0,95	0,94	0,80	0,80	0,72	0,50	0,98		0,98	0,98	-0,96	-0,44	-0,94	0,96	0,96	0,92	-0,80	-0,71
	F05	0,74	0,74	0,74	0,74	0,99	0,87	0,76	0,76	0,72	0,51	0,96	0,98		0,94	-0,96	-0,42	-0,90	0,90	0,93	0,85	-0,75	-0,66
	F2	0,84	0,84	0,84	0,84	0,88	0,99	0,81	0,81	0,71	0,48	0,97	0,97	0,92		-0,94	-0,45	-0,95	0,99	0,97	0,96	-0,83	-0,75
	DTPpr	0,74	0,74	0,74	0,74	0,93	0,91	0,73	0,73	0,65	0,45	0,95	0,96	0,95	0,94		0,43	0,97	-0,91	-0,97	-0,85	0,72	0,61
	DTPperc	0,60	0,60	0,60	0,60	0,50	0,53	0,68	0,68	0,66	0,91	0,59	0,54	0,52	0,54	0,53		0,43	-0,48	-0,42	-0,46	0,46	0,45
	DTPrate	0,76	0,76	0,76	0,76	0,84	0,94	0,72	0,72	0,63	0,41	0,93	0,93	0,89	0,95	0,96	0,51		-0,94	-0,99	-0,89	0,75	0,64
WAUC	0,87	0,87	0,87	0,87	0,83	0,99	0,84	0,84	0,74	0,50	0,96	0,95	0,88	0,99	0,91	0,56	0,93		0,95	0,98	-0,85	-0,78	
GAUC	0,78	0,78	0,78	0,78	0,87	0,95	0,74	0,74	0,64	0,41	0,95	0,96	0,91	0,97	0,97	0,50	0,98	0,95		0,91	-0,77	-0,66	
WAUCB	0,88	0,88	0,88	0,88	0,76	0,97	0,82	0,82	0,71	0,49	0,92	0,90	0,83	0,95	0,85	0,54	0,88	0,97	0,90		-0,86	-0,80	
MAE	0,94	0,94	0,94	0,94	0,67	0,81	0,90	0,90	0,78	0,51	0,82	0,78	0,72	0,81	0,71	0,54	0,74	0,83	0,75	0,84		0,85	
RMSE	0,90	0,90	0,90	0,90	0,59	0,75	0,86	0,86	0,74	0,53	0,74	0,70	0,64	0,73	0,62	0,53	0,64	0,77	0,66	0,78	0,84		

Table 7.26 – ANN classifier: cross dataset pairwise correlations for unweighted macro-average metrics

		Linear																					
		TP	FP	TN	FN	P	R	A	E	FPR	FPP	SMCC	F1	F05	F2	DTPpr	DTPperc	DTPrate	WAUC	GAUC	WAUCB	MAE	RMSE
Rank	TP		-1,00	1,00	-1,00	0,90	0,95	0,98	-0,98	-0,88	-0,57	0,95	0,94	0,92	0,95	-0,92	-0,52	-0,93	0,96	0,94	0,96	-0,94	-0,92
	FP	1,00		-1,00	1,00	-0,90	-0,95	-0,98	0,98	0,88	0,57	-0,95	-0,94	-0,92	-0,95	0,92	0,52	0,93	-0,96	-0,94	-0,96	0,94	0,92
	TN	1,00	1,00		-1,00	0,90	0,95	0,98	-0,98	-0,88	-0,57	0,95	0,94	0,92	0,95	-0,92	-0,52	-0,93	0,96	0,94	0,96	-0,94	-0,92
	FN	1,00	1,00	1,00		-0,90	-0,95	-0,98	0,98	0,88	0,57	-0,95	-0,94	-0,92	-0,95	0,92	0,52	0,93	-0,96	-0,94	-0,96	0,94	0,92
	P	0,87	0,87	0,87	0,87		0,94	0,90	-0,90	-0,83	-0,55	0,97	0,98	0,99	0,96	-0,98	-0,48	-0,95	0,94	0,96	0,92	-0,89	-0,85
	R	0,94	0,94	0,94	0,94	0,91		0,93	-0,93	-0,85	-0,51	0,98	0,98	0,96	0,99	-0,97	-0,48	-0,98	0,99	0,98	0,99	-0,93	-0,89
	A	0,96	0,96	0,96	0,96	0,88	0,91		-1,00	-0,95	-0,63	0,95	0,93	0,92	0,94	-0,91	-0,56	-0,92	0,95	0,93	0,94	-0,93	-0,90
	E	0,96	0,96	0,96	0,96	0,88	0,91	0,99		0,95	0,63	-0,95	-0,93	-0,92	-0,94	0,91	0,56	0,92	-0,95	-0,93	-0,94	0,93	0,90
	FPR	0,84	0,84	0,84	0,84	0,81	0,80	0,93	0,93		0,66	-0,87	-0,85	-0,84	-0,85	0,84	0,57	0,84	-0,87	-0,85	-0,86	0,86	0,81
	FPP	0,60	0,60	0,60	0,60	0,58	0,54	0,69	0,69	0,76		-0,56	-0,54	-0,54	-0,52	0,53	0,85	0,50	-0,53	-0,51	-0,52	0,52	0,51
	SMCC	0,94	0,94	0,94	0,94	0,96	0,97	0,94	0,94	0,84	0,60		0,99	0,98	0,99	-0,98	-0,50	-0,98	0,98	0,98	0,97	-0,93	-0,89
	F1	0,93	0,93	0,93	0,93	0,97	0,97	0,92	0,92	0,82	0,57	0,99		0,99	0,99	-0,99	-0,49	-0,98	0,98	0,99	0,97	-0,92	-0,88
	F05	0,90	0,90	0,90	0,90	0,99	0,94	0,90	0,90	0,82	0,58	0,98	0,99		0,97	-0,98	-0,49	-0,96	0,96	0,97	0,94	-0,91	-0,86
	F2	0,94	0,94	0,94	0,94	0,94	0,99	0,92	0,92	0,81	0,56	0,99	0,99	0,96		-0,98	-0,49	-0,98	0,99	0,99	0,98	-0,93	-0,89
	DTPpr	0,90	0,90	0,90	0,90	0,96	0,96	0,90	0,90	0,80	0,56	0,98	0,98	0,97	0,98		0,51	0,99	-0,97	-0,99	-0,95	0,91	0,85
	DTPperc	0,70	0,70	0,70	0,70	0,68	0,69	0,75	0,75	0,78	0,90	0,72	0,70	0,70	0,70	0,71		0,51	-0,49	-0,50	-0,49	0,47	0,42
	DTPrate	0,91	0,91	0,91	0,91	0,93	0,97	0,89	0,89	0,79	0,53	0,97	0,97	0,95	0,98	0,98	0,69		-0,98	-0,99	-0,97	0,92	0,85
WAUC	0,94	0,94	0,94	0,94	0,92	0,99	0,93	0,93	0,83	0,56	0,98	0,98	0,95	0,99	0,96	0,70	0,97		0,98	0,99	-0,94	-0,89	
GAUC	0,92	0,92	0,92	0,92	0,94	0,98	0,90	0,90	0,80	0,54	0,98	0,98	0,96	0,99	0,98	0,69	0,99	0,98		0,97	-0,92	-0,86	
WAUCB	0,94	0,94	0,94	0,94	0,90	0,98	0,93	0,93	0,82	0,56	0,97	0,96	0,93	0,98	0,95	0,70	0,96	0,99	0,97		-0,94	-0,90	
MAE	0,92	0,92	0,92	0,92	0,84	0,90	0,90	0,90	0,81	0,56	0,90	0,89	0,87	0,90	0,88	0,68	0,88	0,91	0,89	0,91		0,92	
RMSE	0,92	0,92	0,92	0,92	0,81	0,86	0,89	0,89	0,78	0,54	0,86	0,86	0,84	0,86	0,83	0,62	0,83	0,87	0,84	0,86	0,89		

Table 7.27 – SVM classifier: cross dataset pairwise correlations for unweighted macro-average metrics

	Linear																						
	TP	FP	TN	FN	P	R	A	E	FPR	FPP	SMCC	F1	F05	F2	DTPpr	DTPperc	DTPrate	WAUC	GAUC	WAUCB	MAE	RMSE	
Rank	TP	-1,00	1,00	-1,00	0,95	0,98	0,95	-0,95	-0,84	-0,69	0,97	0,97	0,96	0,98	-0,97	-0,82	-0,98	0,98	0,98	0,97	-1,00	-0,99	
	FP	1,00		-1,00	1,00	-0,95	-0,98	-0,95	0,95	0,84	0,69	-0,97	-0,97	-0,96	-0,98	0,97	0,82	0,98	-0,98	-0,98	-0,97	1,00	0,99
	TN	1,00	1,00		-1,00	0,95	0,98	0,95	-0,95	-0,84	-0,69	0,97	0,97	0,96	0,98	-0,97	-0,82	-0,98	0,98	0,98	0,97	-1,00	-0,99
	FN	1,00	1,00	1,00		-0,95	-0,98	-0,95	0,95	0,84	0,69	-0,97	-0,97	-0,96	-0,98	0,97	0,82	0,98	-0,98	-0,98	-0,97	1,00	0,99
	P	0,95	0,95	0,95	0,95		0,96	0,92	-0,92	-0,82	-0,67	0,98	0,99	0,99	0,98	-0,99	-0,79	-0,97	0,96	0,98	0,95	-0,95	-0,94
	R	0,98	0,98	0,98	0,98	0,96		0,94	-0,94	-0,83	-0,66	0,99	0,98	0,97	0,99	-0,98	-0,79	-0,99	0,99	0,99	0,99	-0,98	-0,98
	A	0,96	0,96	0,96	0,96	0,93	0,96		-1,00	-0,94	-0,79	0,96	0,94	0,93	0,95	-0,94	-0,88	-0,95	0,96	0,95	0,96	-0,95	-0,95
	E	0,96	0,96	0,96	0,96	0,93	0,96	0,99		0,94	0,79	-0,96	-0,94	-0,93	-0,95	0,94	0,88	0,95	-0,96	-0,95	-0,96	0,95	0,95
	FPR	0,80	0,80	0,80	0,80	0,79	0,80	0,89	0,89		0,85	-0,86	-0,83	-0,83	-0,84	0,83	0,84	0,85	-0,86	-0,85	-0,86	0,84	0,83
	FPP	0,68	0,68	0,68	0,68	0,66	0,66	0,76	0,76	0,85		-0,69	-0,68	-0,67	-0,68	0,67	0,87	0,68	-0,69	-0,68	-0,69	0,69	0,68
	SMCC	0,98	0,98	0,98	0,98	0,98	0,99	0,96	0,96	0,82	0,68		0,99	0,99	0,99	-0,99	-0,81	-0,99	0,99	0,99	0,99	-0,97	-0,98
	F1	0,97	0,97	0,97	0,97	0,99	0,98	0,95	0,95	0,81	0,67	0,99		0,99	0,99	-0,99	-0,80	-0,99	0,98	0,99	0,98	-0,97	-0,97
	F05	0,96	0,96	0,96	0,96	0,99	0,97	0,94	0,94	0,80	0,66	0,99	0,99		0,98	-0,99	-0,80	-0,98	0,97	0,98	0,96	-0,96	-0,95
	F2	0,98	0,98	0,98	0,98	0,97	0,99	0,96	0,96	0,80	0,67	0,99	0,99	0,98		-0,99	-0,80	-0,99	0,99	0,99	0,99	-0,98	-0,98
	DTPpr	0,97	0,97	0,97	0,97	0,98	0,98	0,95	0,95	0,80	0,66	0,99	0,99	0,99	0,99		0,80	0,99	-0,98	-0,99	-0,97	0,97	0,96
	DTPperc	0,89	0,89	0,89	0,89	0,86	0,88	0,92	0,92	0,83	0,86	0,89	0,88	0,87	0,88	0,88		0,83	-0,81	-0,83	-0,80	0,82	0,78
	DTPrate	0,98	0,98	0,98	0,98	0,97	0,99	0,96	0,96	0,81	0,67	0,99	0,99	0,98	0,99	0,99	0,89		-0,99	-0,99	-0,98	0,98	0,97
WAUC	0,98	0,98	0,98	0,98	0,96	0,99	0,97	0,97	0,82	0,68	0,99	0,98	0,97	0,99	0,98	0,89	0,99		0,99	0,99	-0,98	-0,98	
GAUC	0,98	0,98	0,98	0,98	0,97	0,99	0,96	0,96	0,81	0,67	0,99	0,99	0,98	0,99	0,99	0,89	0,99	0,99		0,98	-0,98	-0,97	
WAUCB	0,98	0,98	0,98	0,98	0,96	0,99	0,97	0,97	0,82	0,68	0,98	0,98	0,97	0,99	0,98	0,89	0,99	0,99	0,99		-0,97	-0,98	
MAE	1,00	1,00	1,00	1,00	0,95	0,98	0,96	0,96	0,80	0,68	0,98	0,97	0,96	0,98	0,97	0,89	0,98	0,98	0,98	0,98		0,99	
RMSE	1,00	1,00	1,00	1,00	0,95	0,98	0,96	0,96	0,80	0,68	0,98	0,97	0,96	0,98	0,97	0,89	0,98	0,98	0,98	0,98	1,00		

Table 7.28 – K-NN classifier: cross dataset pairwise correlations for weighted macro-average metrics

		Linear																					
		TP	FP	TN	FN	P	R	A	E	FPR	FPP	SMCC	F1	F05	F2	DTPpr	DTPperc	DTPrate	WAUC	GAUC	WAUCB	MAE	RMSE
Rank	TP		-1,00	1,00	-1,00	0,89	1,00	0,97	-0,97	-0,88	-0,91	0,98	0,98	0,93	0,99	-0,93	-0,90	-0,96	0,99	0,97	0,99	-0,96	-0,91
	FP	1,00		-1,00	1,00	-0,89	-1,00	-0,97	0,97	0,88	0,91	-0,98	-0,98	-0,93	-0,99	0,93	0,90	0,96	-0,99	-0,97	-0,99	0,96	0,91
	TN	1,00	1,00		-1,00	0,89	1,00	0,97	-0,97	-0,88	-0,91	0,98	0,98	0,93	0,99	-0,93	-0,90	-0,96	0,99	0,97	0,99	-0,96	-0,91
	FN	1,00	1,00	1,00		-0,89	-1,00	-0,97	0,97	0,88	0,91	-0,98	-0,98	-0,93	-0,99	0,93	0,90	0,96	-0,99	-0,97	-0,99	0,96	0,91
	P	0,87	0,87	0,87	0,87		0,89	0,89	-0,89	-0,86	-0,84	0,94	0,93	0,99	0,92	-0,97	-0,84	-0,93	0,90	0,94	0,87	-0,87	-0,76
	R	1,00	1,00	1,00	1,00	0,87		0,97	-0,97	-0,88	-0,91	0,98	0,98	0,93	0,99	-0,93	-0,90	-0,96	0,99	0,97	0,99	-0,96	-0,91
	A	0,97	0,97	0,97	0,97	0,86	0,97		-1,00	-0,94	-0,94	0,98	0,97	0,92	0,97	-0,92	-0,90	-0,95	0,98	0,96	0,97	-0,94	-0,87
	E	0,97	0,97	0,97	0,97	0,86	0,97	1,00		0,94	0,94	-0,98	-0,97	-0,92	-0,97	0,92	0,90	0,95	-0,98	-0,96	-0,97	0,94	0,87
	FPR	0,87	0,87	0,87	0,87	0,87	0,87	0,93	0,93		0,95	-0,91	-0,91	-0,88	-0,89	0,86	0,84	0,88	-0,90	-0,89	-0,89	0,86	0,77
	FPP	0,92	0,92	0,92	0,92	0,83	0,92	0,94	0,94	0,92		-0,92	-0,92	-0,87	-0,91	0,86	0,89	0,88	-0,92	-0,89	-0,91	0,88	0,83
	SMCC	0,98	0,98	0,98	0,98	0,93	0,98	0,97	0,97	0,91	0,92		0,99	0,97	0,99	-0,96	-0,90	-0,98	0,99	0,99	0,97	-0,95	-0,88
	F1	0,98	0,98	0,98	0,98	0,91	0,98	0,96	0,96	0,90	0,92	0,99		0,96	0,99	-0,96	-0,91	-0,98	0,99	0,99	0,98	-0,95	-0,87
	F05	0,93	0,93	0,93	0,93	0,98	0,93	0,91	0,91	0,89	0,88	0,97	0,96		0,96	-0,98	-0,87	-0,95	0,94	0,97	0,91	-0,91	-0,81
	F2	0,99	0,99	0,99	0,99	0,90	0,99	0,96	0,96	0,88	0,92	0,99	0,99	0,95		-0,96	-0,90	-0,97	0,99	0,98	0,98	-0,96	-0,89
	DTPpr	0,93	0,93	0,93	0,93	0,95	0,93	0,89	0,89	0,84	0,86	0,96	0,96	0,97	0,95		0,91	0,98	-0,94	-0,98	-0,92	0,91	0,80
	DTPperc	0,93	0,93	0,93	0,93	0,80	0,93	0,89	0,89	0,79	0,90	0,91	0,91	0,85	0,92	0,89		0,93	-0,91	-0,92	-0,90	0,87	0,81
	DTPrate	0,95	0,95	0,95	0,95	0,90	0,95	0,91	0,91	0,85	0,87	0,97	0,97	0,94	0,96	0,98	0,92		-0,97	-0,99	-0,95	0,93	0,84
WAUC	0,99	0,99	0,99	0,99	0,88	0,99	0,97	0,97	0,89	0,92	0,98	0,98	0,93	0,99	0,93	0,92	0,95		0,98	0,99	-0,96	-0,90	
GAUC	0,97	0,97	0,97	0,97	0,92	0,97	0,94	0,94	0,87	0,90	0,98	0,98	0,96	0,98	0,98	0,92	0,98	0,97		0,96	-0,94	-0,85	
WAUCB	0,98	0,98	0,98	0,98	0,85	0,98	0,97	0,97	0,87	0,92	0,96	0,97	0,90	0,98	0,90	0,92	0,93	0,98	0,95		-0,95	-0,90	
MAE	0,94	0,94	0,94	0,94	0,85	0,94	0,92	0,92	0,85	0,87	0,93	0,93	0,90	0,94	0,89	0,87	0,91	0,94	0,93	0,93		0,87	
RMSE	0,89	0,89	0,89	0,89	0,75	0,89	0,88	0,88	0,77	0,83	0,87	0,87	0,81	0,88	0,80	0,84	0,84	0,89	0,85	0,89	0,84		

Table 7.29 – KStar classifier: cross dataset pairwise correlations for weighted macro-average metrics

Rank	Linear																							
	TP	FP	TN	FN	P	R	A	E	FPR	FPP	SMCC	F1	F05	F2	DTPpr	DTPperc	DTPrate	WAUC	GAUC	WAUCB	MAE	RMSE		
TP		-0,98	0,62	-0,98	0,92	0,99	0,97	-0,97	-0,74	-0,83	0,97	0,98	0,95	0,99	-0,96	-0,89	-0,96	0,97	0,95	0,97	-0,93	-0,85		
FP	0,98		-0,47	1,00	-0,93	-0,99	-0,96	0,96	0,75	0,85	-0,97	-0,98	-0,95	-0,99	0,96	0,90	0,96	-0,97	-0,96	-0,97	0,93	0,84		
TN	0,99	0,97		-0,47	0,48	0,56	0,55	-0,55	-0,34	-0,36	0,52	0,53	0,50	0,54	-0,51	-0,45	-0,52	0,54	0,52	0,54	-0,50	-0,48		
FN	0,98	1,00	0,97		-0,93	-0,99	-0,96	0,96	0,75	0,85	-0,97	-0,98	-0,95	-0,99	0,96	0,90	0,96	-0,97	-0,96	-0,97	0,93	0,84		
P	0,91	0,92	0,91	0,92		0,93	0,93	-0,93	-0,80	-0,80	0,97	0,95	0,99	0,95	-0,98	-0,87	-0,96	0,95	0,97	0,94	-0,93	-0,76		
R	0,99	0,99	0,99	0,99	0,91		0,97	-0,97	-0,74	-0,84	0,97	0,99	0,95	0,99	-0,96	-0,90	-0,96	0,98	0,96	0,98	-0,94	-0,85		
A	0,96	0,96	0,96	0,96	0,93	0,97		-1,00	-0,84	-0,87	0,98	0,98	0,95	0,97	-0,95	-0,88	-0,97	0,98	0,97	0,98	-0,94	-0,84		
E	0,96	0,96	0,96	0,96	0,93	0,97	1,00		0,84	0,87	-0,98	-0,98	-0,95	-0,97	0,95	0,88	0,97	-0,98	-0,97	-0,98	0,94	0,84		
FPR	0,78	0,80	0,77	0,80	0,83	0,79	0,84	0,84		0,89	-0,82	-0,79	-0,80	-0,77	0,77	0,67	0,80	-0,81	-0,80	-0,80	0,78	0,69		
FPP	0,83	0,85	0,82	0,85	0,82	0,84	0,85	0,85	0,94		-0,84	-0,85	-0,82	-0,84	0,82	0,81	0,82	-0,84	-0,82	-0,84	0,81	0,74		
SMCC	0,96	0,97	0,96	0,97	0,96	0,97	0,98	0,98	0,84	0,85		0,99	0,98	0,98	-0,98	-0,87	-0,98	0,99	0,98	0,99	-0,95	-0,84		
F1	0,97	0,97	0,97	0,97	0,95	0,98	0,98	0,98	0,83	0,85	0,99		0,97	0,99	-0,98	-0,89	-0,98	0,99	0,98	0,98	-0,95	-0,85		
F05	0,94	0,94	0,94	0,94	0,99	0,94	0,95	0,95	0,83	0,84	0,98	0,97		0,97	-0,99	-0,89	-0,98	0,97	0,98	0,96	-0,94	-0,78		
F2	0,98	0,98	0,98	0,98	0,95	0,98	0,98	0,98	0,82	0,85	0,99	0,99	0,99		-0,98	-0,91	-0,98	0,98	0,98	0,98	-0,94	-0,83		
DTPpr	0,95	0,95	0,95	0,95	0,98	0,95	0,95	0,95	0,82	0,83	0,98	0,98	0,99	0,97		0,92	0,98	-0,97	-0,99	-0,96	0,94	0,77		
DTPperc	0,96	0,96	0,95	0,96	0,89	0,96	0,94	0,94	0,78	0,85	0,94	0,95	0,92	0,95	0,94		0,91	-0,88	-0,90	-0,88	0,84	0,64		
DTPrate	0,96	0,96	0,96	0,96	0,95	0,96	0,97	0,97	0,83	0,83	0,98	0,98	0,97	0,98	0,98	0,94		-0,98	-0,99	-0,98	0,94	0,78		
WAUC	0,97	0,97	0,97	0,97	0,94	0,97	0,98	0,98	0,83	0,85	0,99	0,99	0,97	0,99	0,97	0,94	0,98		0,98	0,99	-0,95	-0,84		
GAUC	0,96	0,96	0,96	0,96	0,96	0,96	0,97	0,97	0,83	0,84	0,99	0,99	0,98	0,98	0,99	0,94	0,99	0,99		0,98	-0,94	-0,78		
WAUCB	0,97	0,97	0,96	0,97	0,92	0,97	0,98	0,98	0,82	0,84	0,98	0,98	0,95	0,98	0,96	0,94	0,98	0,99	0,98		-0,95	-0,84		
MAE	0,92	0,92	0,91	0,92	0,92	0,92	0,93	0,93	0,80	0,81	0,94	0,94	0,93	0,94	0,94	0,90	0,94	0,94	0,94	0,94		0,80		
RMSE	0,88	0,88	0,88	0,88	0,82	0,89	0,88	0,88	0,74	0,78	0,88	0,88	0,85	0,88	0,85	0,84	0,87	0,88	0,87	0,87	0,81			

Table 7.30 – LWL with Naïve Bayes classifier: cross dataset pairwise correlations for weighted macro-average metrics

Rank	Linear																							
	TP	FP	TN	FN	P	R	A	E	FPR	FPP	SMCC	F1	F05	F2	DTPpr	DTPperc	DTPrate	WAUC	GAUC	WAUCB	MAE	RMSE		
TP		-1,00	1,00	-1,00	0,91	1,00	0,97	-0,97	-0,85	-0,92	0,98	0,99	0,94	0,99	-0,95	-0,91	-0,97	0,99	0,97	0,99	-0,94	-0,92		
FP	1,00		-1,00	1,00	-0,91	-1,00	-0,97	0,97	0,85	0,92	-0,98	-0,99	-0,94	-0,99	0,95	0,91	0,97	-0,99	-0,97	-0,99	0,94	0,92		
TN	1,00	1,00		-1,00	0,91	1,00	0,97	-0,97	-0,85	-0,92	0,98	0,99	0,94	0,99	-0,95	-0,91	-0,97	0,99	0,97	0,99	-0,94	-0,92		
FN	1,00	1,00	1,00		-0,91	-1,00	-0,97	0,97	0,85	0,92	-0,98	-0,99	-0,94	-0,99	0,95	0,91	0,97	-0,99	-0,97	-0,99	0,94	0,92		
P	0,91	0,91	0,91	0,91		0,91	0,91	-0,91	-0,86	-0,87	0,95	0,94	0,99	0,95	-0,98	-0,89	-0,95	0,92	0,96	0,90	-0,90	-0,80		
R	1,00	1,00	1,00	1,00	0,91		0,97	-0,97	-0,85	-0,92	0,98	0,99	0,94	0,99	-0,95	-0,91	-0,97	0,99	0,97	0,99	-0,94	-0,92		
A	0,98	0,98	0,98	0,98	0,90	0,98		-1,00	-0,90	-0,94	0,98	0,98	0,93	0,97	-0,94	-0,91	-0,96	0,98	0,96	0,98	-0,92	-0,90		
E	0,98	0,98	0,98	0,98	0,90	0,98	1,00		0,90	0,94	-0,98	-0,98	-0,93	-0,97	0,94	0,91	0,96	-0,98	-0,96	-0,98	0,92	0,90		
FPR	0,89	0,89	0,89	0,89	0,89	0,89	0,93	0,93		0,91	-0,88	-0,87	-0,86	-0,86	0,85	0,82	0,86	-0,88	-0,87	-0,87	0,84	0,77		
FPP	0,92	0,92	0,92	0,92	0,87	0,92	0,94	0,94	0,94		-0,93	-0,93	-0,89	-0,92	0,89	0,91	0,91	-0,93	-0,91	-0,92	0,89	0,84		
SMCC	0,98	0,98	0,98	0,98	0,95	0,98	0,97	0,97	0,92	0,93		0,99	0,97	0,99	-0,97	-0,92	-0,98	0,99	0,99	0,98	-0,95	-0,90		
F1	0,98	0,98	0,98	0,98	0,94	0,98	0,97	0,97	0,91	0,93	0,99		0,97	0,99	-0,97	-0,93	-0,99	0,99	0,99	0,98	-0,94	-0,90		
F05	0,95	0,95	0,95	0,95	0,99	0,95	0,93	0,93	0,91	0,90	0,98	0,97		0,97	-0,99	-0,90	-0,96	0,95	0,98	0,93	-0,92	-0,84		
F2	0,99	0,99	0,99	0,99	0,94	0,99	0,97	0,97	0,90	0,93	0,99	0,99	0,97		-0,97	-0,93	-0,98	0,99	0,99	0,98	-0,94	-0,91		
DTPpr	0,95	0,95	0,95	0,95	0,97	0,95	0,93	0,93	0,88	0,89	0,97	0,97	0,98	0,97		0,93	0,98	-0,95	-0,99	-0,93	0,91	0,83		
DTPperc	0,95	0,95	0,95	0,95	0,88	0,95	0,93	0,93	0,86	0,93	0,95	0,95	0,91	0,95	0,93		0,94	-0,92	-0,94	-0,91	0,87	0,80		
DTPrate	0,97	0,97	0,97	0,97	0,94	0,97	0,94	0,94	0,88	0,91	0,98	0,98	0,96	0,98	0,98	0,95		-0,97	-0,99	-0,96	0,93	0,87		
WAUC	0,99	0,99	0,99	0,99	0,92	0,99	0,98	0,98	0,91	0,93	0,99	0,99	0,95	0,99	0,96	0,95	0,97		0,98	0,99	-0,94	-0,92		
GAUC	0,98	0,98	0,98	0,98	0,95	0,98	0,96	0,96	0,90	0,92	0,99	0,99	0,97	0,99	0,98	0,95	0,99	0,98		0,97	-0,93	-0,88		
WAUCB	0,99	0,99	0,99	0,99	0,90	0,99	0,98	0,98	0,89	0,93	0,97	0,98	0,93	0,98	0,94	0,95	0,96	0,99	0,97		-0,94	-0,93		
MAE	0,94	0,94	0,94	0,94	0,90	0,94	0,93	0,93	0,87	0,89	0,95	0,94	0,93	0,95	0,92	0,91	0,93	0,95	0,94	0,94		0,87		
RMSE	0,93	0,93	0,93	0,93	0,83	0,93	0,92	0,92	0,84	0,87	0,92	0,92	0,87	0,92	0,88	0,89	0,90	0,93	0,91	0,93	0,89			

Table 7.31 – Decision Tree classifier: cross dataset pairwise correlations for weighted macro-average metrics

	Linear																						
	TP	FP	TN	FN	P	R	A	E	FPR	FPP	SMCC	F1	F05	F2	DTPpr	DTPperc	DTPrate	WAUC	GAUC	WAUCB	MAE	RMSE	
Rank	TP	-1,00	1,00	-1,00	0,76	1,00	0,93	-0,93	-0,74	-0,83	0,96	0,96	0,86	0,99	-0,87	-0,77	-0,92	0,99	0,95	0,98	-0,94	-0,90	
	FP	1,00		-1,00	1,00	-0,76	-1,00	-0,93	0,93	0,74	0,83	-0,96	-0,96	-0,86	-0,99	0,87	0,77	0,92	-0,99	-0,95	-0,98	0,94	0,90
	TN	1,00	1,00		-1,00	0,76	1,00	0,93	-0,93	-0,74	-0,83	0,96	0,96	0,86	0,99	-0,87	-0,77	-0,92	0,99	0,95	0,98	-0,94	-0,90
	FN	1,00	1,00	1,00		-0,76	-1,00	-0,93	0,93	0,74	0,83	-0,96	-0,96	-0,86	-0,99	0,87	0,77	0,92	-0,99	-0,95	-0,98	0,94	0,90
	P	0,73	0,73	0,73	0,73		0,76	0,75	-0,75	-0,79	-0,77	0,88	0,85	0,98	0,83	-0,91	-0,57	-0,81	0,79	0,86	0,71	-0,76	-0,66
	R	1,00	1,00	1,00	1,00	0,73		0,93	-0,93	-0,74	-0,83	0,96	0,96	0,86	0,99	-0,87	-0,77	-0,92	0,99	0,95	0,98	-0,94	-0,90
	A	0,92	0,92	0,92	0,92	0,71	0,92		-1,00	-0,88	-0,90	0,94	0,94	0,83	0,93	-0,80	-0,73	-0,85	0,95	0,89	0,92	-0,90	-0,84
	E	0,92	0,92	0,92	0,92	0,71	0,92	1,00		0,88	0,90	-0,94	-0,94	-0,83	-0,93	0,80	0,73	0,85	-0,95	-0,89	-0,92	0,90	0,84
	FPR	0,72	0,72	0,72	0,72	0,76	0,72	0,86	0,86		0,87	-0,85	-0,82	-0,82	-0,78	0,73	0,58	0,72	-0,79	-0,76	-0,73	0,76	0,66
	FPP	0,81	0,81	0,81	0,81	0,75	0,81	0,88	0,88	0,87		-0,89	-0,88	-0,83	-0,86	0,78	0,81	0,79	-0,86	-0,82	-0,82	0,81	0,75
	SMCC	0,95	0,95	0,95	0,95	0,85	0,95	0,93	0,93	0,82	0,87		0,99	0,95	0,98	-0,93	-0,75	-0,94	0,97	0,97	0,93	-0,93	-0,86
	F1	0,96	0,96	0,96	0,96	0,82	0,96	0,92	0,92	0,80	0,86	0,99		0,92	0,98	-0,93	-0,78	-0,95	0,97	0,98	0,94	-0,93	-0,86
	F05	0,84	0,84	0,84	0,84	0,97	0,84	0,81	0,81	0,79	0,81	0,94	0,91		0,91	-0,95	-0,65	-0,89	0,88	0,93	0,81	-0,84	-0,76
	F2	0,99	0,99	0,99	0,99	0,80	0,99	0,92	0,92	0,75	0,83	0,98	0,97	0,90		-0,91	-0,76	-0,94	0,99	0,97	0,96	-0,94	-0,89
	DTPpr	0,86	0,86	0,86	0,86	0,89	0,86	0,78	0,78	0,70	0,76	0,92	0,93	0,94	0,90		0,73	0,96	-0,87	-0,96	-0,80	0,84	0,74
	DTPperc	0,79	0,79	0,79	0,79	0,60	0,79	0,75	0,75	0,60	0,83	0,79	0,81	0,69	0,79	0,76		0,79	-0,77	-0,77	-0,73	0,71	0,68
	DTPrate	0,91	0,91	0,91	0,91	0,78	0,91	0,82	0,82	0,69	0,76	0,93	0,95	0,87	0,93	0,96	0,81		-0,92	-0,97	-0,86	0,88	0,80
WAUC	0,99	0,99	0,99	0,99	0,75	0,99	0,94	0,94	0,77	0,84	0,97	0,97	0,86	0,99	0,87	0,79	0,91		0,96	0,97	-0,95	-0,90	
GAUC	0,95	0,95	0,95	0,95	0,82	0,95	0,87	0,87	0,73	0,80	0,97	0,97	0,91	0,97	0,96	0,80	0,97	0,95		0,91	-0,92	-0,84	
WAUCB	0,97	0,97	0,97	0,97	0,67	0,97	0,91	0,91	0,71	0,78	0,91	0,92	0,78	0,95	0,79	0,75	0,84	0,97	0,90		-0,93	-0,90	
MAE	0,94	0,94	0,94	0,94	0,72	0,94	0,88	0,88	0,73	0,78	0,92	0,92	0,82	0,93	0,83	0,74	0,86	0,94	0,91	0,92		0,85	
RMSE	0,90	0,90	0,90	0,90	0,64	0,90	0,85	0,85	0,66	0,74	0,86	0,86	0,75	0,89	0,75	0,71	0,80	0,90	0,84	0,89	0,84		

Table 7.32 – ANN classifier: cross dataset pairwise correlations for weighted macro-average metrics

		Linear																					
		TP	FP	TN	FN	P	R	A	E	FPR	FPP	SMCC	F1	F05	F2	DTPpr	DTPperc	DTPrate	WAUC	GAUC	WAUCB	MAE	RMSE
Rank	TP		-1,00	1,00	-1,00	0,94	1,00	0,96	-0,96	-0,82	-0,87	0,98	0,98	0,96	0,99	-0,96	-0,73	-0,97	0,99	0,98	0,98	-0,94	-0,92
	FP	1,00		-1,00	1,00	-0,94	-1,00	-0,96	0,96	0,82	0,87	-0,98	-0,98	-0,96	-0,99	0,96	0,73	0,97	-0,99	-0,98	-0,98	0,94	0,92
	TN	1,00	1,00		-1,00	0,94	1,00	0,96	-0,96	-0,82	-0,87	0,98	0,98	0,96	0,99	-0,96	-0,73	-0,97	0,99	0,98	0,98	-0,94	-0,92
	FN	1,00	1,00	1,00		-0,94	-1,00	-0,96	0,96	0,82	0,87	-0,98	-0,98	-0,96	-0,99	0,96	0,73	0,97	-0,99	-0,98	-0,98	0,94	0,92
	P	0,91	0,91	0,91	0,91		0,94	0,92	-0,92	-0,86	-0,87	0,97	0,96	0,99	0,96	-0,98	-0,69	-0,96	0,95	0,97	0,92	-0,90	-0,85
	R	1,00	1,00	1,00	1,00	0,91		0,96	-0,96	-0,82	-0,87	0,98	0,98	0,96	0,99	-0,96	-0,73	-0,97	0,99	0,98	0,98	-0,94	-0,92
	A	0,95	0,95	0,95	0,95	0,90	0,95		-1,00	-0,91	-0,91	0,97	0,96	0,94	0,96	-0,93	-0,71	-0,94	0,97	0,95	0,96	-0,91	-0,89
	E	0,95	0,95	0,95	0,95	0,90	0,95	1,00		0,91	0,91	-0,97	-0,96	-0,94	-0,96	0,93	0,71	0,94	-0,97	-0,95	-0,96	0,91	0,89
	FPR	0,80	0,80	0,80	0,80	0,85	0,80	0,90	0,90		0,87	-0,87	-0,86	-0,86	-0,84	0,83	0,60	0,84	-0,87	-0,85	-0,84	0,82	0,76
	FPP	0,84	0,84	0,84	0,84	0,84	0,84	0,89	0,89	0,88		-0,89	-0,90	-0,88	-0,88	0,87	0,79	0,87	-0,89	-0,88	-0,87	0,84	0,80
	SMCC	0,98	0,98	0,98	0,98	0,95	0,98	0,96	0,96	0,86	0,87		0,99	0,98	0,99	-0,97	-0,71	-0,97	0,99	0,98	0,97	-0,94	-0,91
	F1	0,98	0,98	0,98	0,98	0,95	0,98	0,96	0,96	0,85	0,87	0,99		0,98	0,99	-0,98	-0,74	-0,98	0,99	0,99	0,98	-0,94	-0,91
	F05	0,94	0,94	0,94	0,94	0,99	0,94	0,93	0,93	0,85	0,85	0,98	0,97		0,97	-0,98	-0,71	-0,97	0,96	0,98	0,94	-0,92	-0,88
	F2	0,99	0,99	0,99	0,99	0,93	0,99	0,95	0,95	0,82	0,85	0,99	0,98	0,96		-0,97	-0,73	-0,98	0,99	0,99	0,98	-0,94	-0,92
	DTPpr	0,94	0,94	0,94	0,94	0,96	0,94	0,90	0,90	0,80	0,83	0,96	0,97	0,97	0,96		0,76	0,99	-0,96	-0,99	-0,94	0,92	0,86
	DTPperc	0,82	0,82	0,82	0,82	0,77	0,82	0,80	0,79	0,69	0,89	0,82	0,83	0,79	0,82	0,83		0,78	-0,72	-0,76	-0,72	0,66	0,59
DTPrate	0,95	0,95	0,95	0,95	0,93	0,95	0,91	0,91	0,80	0,83	0,96	0,97	0,95	0,96	0,98	0,84		-0,97	-0,99	-0,96	0,92	0,87	
WAUC	0,99	0,99	0,99	0,99	0,92	0,99	0,96	0,96	0,84	0,85	0,98	0,98	0,95	0,99	0,94	0,81	0,95		0,98	0,98	-0,94	-0,92	
GAUC	0,97	0,97	0,97	0,97	0,94	0,97	0,93	0,93	0,82	0,84	0,98	0,98	0,97	0,98	0,98	0,83	0,98	0,98		0,97	-0,93	-0,89	
WAUCB	0,98	0,98	0,98	0,98	0,89	0,98	0,96	0,96	0,82	0,84	0,97	0,97	0,92	0,98	0,92	0,80	0,93	0,98	0,96		-0,94	-0,92	
MAE	0,92	0,92	0,92	0,92	0,87	0,92	0,88	0,88	0,78	0,80	0,91	0,92	0,89	0,92	0,90	0,78	0,91	0,92	0,92	0,91		0,92	
RMSE	0,92	0,92	0,92	0,92	0,84	0,92	0,89	0,89	0,76	0,77	0,91	0,90	0,87	0,92	0,87	0,73	0,88	0,92	0,90	0,91	0,89		

Table 7.33 – SVM classifier: cross dataset pairwise correlations for weighted macro-average metrics

		Linear																					
		TP	FP	TN	FN	P	R	A	E	FPR	FPP	SMCC	F1	F05	F2	DTPpr	DTPperc	DTPrate	WAUC	GAUC	WAUCB	MAE	RMSE
Rank	TP		-1,00	1,00	-1,00	0,95	1,00	0,94	-0,94	-0,74	-0,84	0,99	0,99	0,96	0,99	-0,97	-0,85	-0,98	0,99	0,98	0,99	-1,00	-0,99
	FP	1,00		-1,00	1,00	-0,95	-1,00	-0,94	0,94	0,74	0,84	-0,99	-0,99	-0,96	-0,99	0,97	0,85	0,98	-0,99	-0,98	-0,99	1,00	0,99
	TN	1,00	1,00		-1,00	0,95	1,00	0,94	-0,94	-0,74	-0,84	0,99	0,99	0,96	0,99	-0,97	-0,85	-0,98	0,99	0,98	0,99	-1,00	-0,99
	FN	1,00	1,00	1,00		-0,95	-1,00	-0,94	0,94	0,74	0,84	-0,99	-0,99	-0,96	-0,99	0,97	0,85	0,98	-0,99	-0,98	-0,99	1,00	0,99
	P	0,95	0,95	0,95	0,95		0,95	0,92	-0,92	-0,75	-0,82	0,97	0,97	0,99	0,97	-0,98	-0,84	-0,96	0,95	0,97	0,93	-0,95	-0,93
	R	1,00	1,00	1,00	1,00	0,95		0,94	-0,94	-0,74	-0,84	0,99	0,99	0,96	0,99	-0,97	-0,85	-0,98	0,99	0,98	0,99	-1,00	-0,99
	A	0,95	0,95	0,95	0,95	0,93	0,95		-1,00	-0,88	-0,93	0,96	0,96	0,93	0,94	-0,93	-0,88	-0,95	0,97	0,95	0,97	-0,94	-0,94
	E	0,95	0,95	0,95	0,95	0,93	0,95	1,00		0,88	0,93	-0,96	-0,96	-0,93	-0,94	0,93	0,88	0,95	-0,97	-0,95	-0,97	0,94	0,94
	FPR	0,73	0,73	0,73	0,73	0,75	0,73	0,84	0,84		0,88	-0,79	-0,78	-0,75	-0,74	0,74	0,73	0,77	-0,80	-0,76	-0,80	0,74	0,75
	FPP	0,81	0,81	0,81	0,81	0,79	0,81	0,88	0,88	0,91		-0,86	-0,86	-0,83	-0,84	0,83	0,90	0,85	-0,87	-0,85	-0,87	0,84	0,83
	SMCC	0,98	0,98	0,98	0,98	0,97	0,98	0,97	0,97	0,77	0,83		0,99	0,98	0,99	-0,98	-0,86	-0,99	0,99	0,99	0,98	-0,99	-0,98
	F1	0,99	0,99	0,99	0,99	0,97	0,99	0,96	0,96	0,75	0,82	0,99		0,98	0,99	-0,98	-0,86	-0,99	0,99	0,99	0,99	-0,99	-0,98
	F05	0,97	0,97	0,97	0,97	0,99	0,97	0,94	0,94	0,75	0,80	0,98	0,98		0,98	-0,99	-0,85	-0,98	0,97	0,98	0,95	-0,96	-0,95
	F2	0,99	0,99	0,99	0,99	0,97	0,99	0,95	0,95	0,74	0,81	0,99	0,99	0,98		-0,99	-0,87	-0,99	0,99	0,99	0,98	-0,99	-0,98
	DTPpr	0,98	0,98	0,98	0,98	0,98	0,98	0,94	0,94	0,74	0,80	0,98	0,99	0,99	0,99		0,87	0,99	-0,97	-0,99	-0,96	0,97	0,95
	DTPperc	0,94	0,94	0,94	0,94	0,91	0,94	0,95	0,95	0,77	0,88	0,94	0,94	0,92	0,94	0,94		0,89	-0,87	-0,89	-0,86	0,85	0,81
DTPrate	0,99	0,99	0,99	0,99	0,96	0,99	0,96	0,96	0,75	0,81	0,99	0,99	0,98	0,99	0,99	0,95		-0,99	-0,99	-0,98	0,98	0,96	
WAUC	0,99	0,99	0,99	0,99	0,96	0,99	0,97	0,97	0,77	0,83	0,99	0,99	0,97	0,99	0,98	0,95	0,99		0,98	0,99	-0,99	-0,98	
GAUC	0,99	0,99	0,99	0,99	0,97	0,99	0,96	0,96	0,75	0,81	0,99	0,99	0,98	0,99	0,99	0,95	0,99	0,99		0,98	-0,98	-0,96	
WAUCB	0,99	0,99	0,99	0,99	0,95	0,99	0,97	0,97	0,76	0,82	0,98	0,99	0,96	0,99	0,97	0,95	0,99	0,99	0,99		-0,99	-0,98	
MAE	1,00	1,00	1,00	1,00	0,95	1,00	0,95	0,95	0,73	0,81	0,98	0,99	0,97	0,99	0,98	0,94	0,99	0,99	0,99	0,99		0,99	
RMSE	1,00	1,00	1,00	1,00	0,95	1,00	0,95	0,95	0,73	0,81	0,98	0,99	0,97	0,99	0,98	0,94	0,99	0,99	0,99	0,99	1,00		

Appendix E Additional Resources

E.1 Power Calculations

The power flow in Alternating Current (AC) systems is a complex quantity that has three main components: the apparent power (S) measured in volt-ampere (VA), the real power (P) measured in watts (W) and the reactive power (Q) measured in reactive volt-amperes (VAR).

For purely resistive loads real power is equal to apparent power whilst for all other loads real power is less than apparent power due to the presence of non-linear circuits components (e.g. inductors and capacitors) that affect the electrical current thus causing the alternate storage and release of reactive power as well as the addition of harmonic components of the fundamental frequency, which ultimately result in harmonic powers that spread across the power spectrum.

Implementation wise, power calculation can be performed in either time or frequency domain. Still, in the time domain only fundamental powers can be calculated, whereas in the frequency domain it is possible to calculate fundamental and harmonic powers.

In Table 7.34 we summarize the electric power measurements that can be calculated in time and frequency domain.

Table 7.34 – Power calculations: time vs. frequency domain

Measurement	Symbol	Unit	Time	Frequency
Apparent Power	S	VA	✓	✓
Real Power	P	W	✓	✓
Reactive Power	Q	VAR	✓	✓

Measurement	Symbol	Unit	Time	Frequency
RMS voltage	V_{RMS}	V	✓	✓
RMS current	I_{RMS}	A	✓	✓
Power Factor	PF	N/A	✓	✓
Phase voltage	$\angle V$	rad	✗	✓
Phase current	$\angle I$	rad	✗	✓
Phase difference	\emptyset	rad	✗	✓
Real Harmonic Power	$P_k (k > 1)$	W	✗	✓
Reactive Harmonic Power	$Q_k (k > 1)$	VAR	✗	✓

At this stage, we should remark that with the exception of real power, to date there is still no universally accepted method to calculate fundamental reactive power or to define harmonic powers (either real or reactive). As such, in this work we follow the approach proposed in [173] to define fundamental and harmonic powers. Next we provide the implementations details for each of the different power metrics.

E.1.1 Apparent Power

Apparent power is a measure of the maximum power that can be transmitted and supplied to the load. It is the result of the product between the Root-Mean-Squared (RMS) average of the voltage and the current waveforms, as shown in equation (7.29).

$$S = V_{RMS} \times I_{RMS} \quad (7.29)$$

E.1.2 RMS Voltage and Current

In AC systems, the RMS value is the measure of the magnitude of a signal. Mathematically the RMS voltage is the square root of the definite integral of the voltage $v(t)$ squared, as seen in equation (7.30).

$$V_{RMS} = \sqrt{\frac{1}{T} \times \int v^2(t) dt} \quad (7.30)$$

In discrete time, an RMS value is defined as the square root of the mean value of the square of the instantaneous value of a periodically varying quantity, averaged over one, or more, complete cycles. The discrete time equation for calculating voltage RMS is therefore defined as follows:

$$V_{RMS} = \sqrt{\frac{\sum_{n=0}^{N-1} v^2(n)}{N}} \quad (7.31)$$

Where $v(n)$ is the sampled instance of $v(t)$ and N is the number of samples. By analogy the RMS current can also be computed using equations (7.30) and (7.31).

E.1.3 Real Power and Power Factor

Real power (also known as active power) is defined as the power used by a device to produce useful work. Mathematically it is the definite integral of the voltage $v(t)$ multiplied the current $i(t)$, as shown in equation (7.32).

$$P = \frac{1}{T} \int v(t) \times i(t) dt \equiv V_{RMS} \times I_{RMS} \times \cos(\phi) \equiv S \times \cos(\phi) \quad (7.32)$$

Where $\cos(\phi)$ is the power factor (PF), which measures how much the mains efficiency is affected by phase difference ϕ and the harmonic content of the input current. By convention, the phase difference is the angle by which voltage leads current, i.e., $\phi = \angle V - \angle I$.

The power factor is the ratio of real power to apparent power and an equivalent function for its calculation is as follows:

$$PF = \frac{P}{S} \quad (7.33)$$

The discrete time equivalent to calculate real power is given by equation (7.34).

$$P \equiv \frac{1}{N} \sum_{n=0}^{N-1} v(n) * i(n) \quad (7.34)$$

Where $v(n)$ is the sampled instance of $v(t)$, $i(n)$ the sampled instance of $i(t)$, and N is the number of samples in one period.

The real power is therefore obtained by taking the product of the voltage and the current waveform and computing its one cycle average.

E.1.4 Reactive Power

Reactive power (or imaginary power) is a measure of the power going back and forth between the load and the supply that does no useful work.

Mathematically it is the definite integral of the voltage $v(t)$ multiplied the 90° out-of-phase current $i(t)$ that is obtained by shifting the voltage waveform by a quarter cycle before integration. Reactive power is therefore defined as follows:

$$Q = \frac{1}{T} \int v\left(t - \frac{T}{4}\right) \times i(t) dt \equiv V_{RMS} \times I_{RMS} \times \sin(\phi) \equiv S \times \sin(\phi) \quad (7.35)$$

The discrete time equivalent to calculate reactive power is given by equation (7.36).

$$Q \equiv \frac{1}{N} \sum_{n=0}^{N-1} v\left(n - \frac{N}{4}\right) * i(n) \quad (7.36)$$

Where $v\left(n - \frac{N}{4}\right)$ is the sampled instance of $v\left(t - \frac{T}{4}\right)$, $i(n)$ is the sampled instance of $i(t)$ and N is the number of samples in one period.

The reactive power is therefore obtained by taking the product of the voltage (shifted by a quarter of a cycle) and the current waveform and computing its one cycle average.

E.1.5 Harmonic Powers

As it was previously mentioned, we are using the approach presented in [173] to compute real and reactive current harmonic powers.

The basic premise behind this method is that since the amount of voltage harmonic distortion is typically very small at the end users' sites, one may use a properly shifted harmonic voltage waveform (period, magnitude and phase) as a reference to compute power at a harmonic frequency. More precisely, the authors use the fundamental voltage waveform to represent the current harmonic powers in terms of the Discrete Fourier Transform (DFT) of the sampled current waveform, as shown in equations (7.37) and (7.38).

$$P_k = \frac{V}{N} \text{Re}(I_k) \quad (7.37)$$

$$Q_k = -\frac{V}{N} \text{Im}(I_k) \quad (7.38)$$

Where V is the peak fundamental voltage, $\text{Re}(I_k)$ is the real part of the transform at the k^{th} frequency component, $\text{Im}(I_k)$ is the imaginary part of the transform at the k^{th} frequency component and N is the length of the corresponding period.