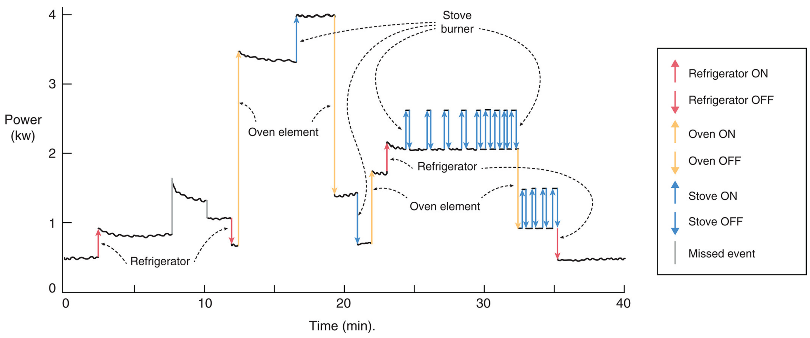


Event detection and classification for disaggregation of energy consumption metering data

Patryk Jan Mickiewicz

Thesis to obtain the Master of Science Degree in

Energy Engineering and Management

Supervisors: Prof. Pedro Manuel Santos de Carvalho
Prof. Hugo Gabriel Valente Morais

Examination Committee

Chairperson: Prof. Duarte de Mesquita e Sousa
Supervisor: Prof. Hugo Gabriel Valente Morais
Member of the Committee: Prof. Célia Maria Santos Cardoso de Jesus

June 2023

Declaration

I declare that this document is an original work of my own authorship and that it fulfills all the requirements of the Code of Conduct and Good Practices of the Universidade de Lisboa.

Dedicated to my mum, my dad, and my little brother.

Resumo

O aumento da eficiência energética é uma das principais soluções para mitigar o impacto das alterações climáticas. O uso de ferramentas de monitorização não intrusivas (NILM em Inglês) permite adquirir um conhecimento detalhado sobre o funcionamento dos equipamentos elétricos evitando a utilização massiva de equipamentos de medição.

Esta dissertação de mestrado tem como foco explorar e avaliar diversas abordagens para deteção e classificação de eventos usando diversos algoritmos de NILM. O trabalho desenvolvido começou por testar uma biblioteca existente, nomeadamente o NILMTK, que oferece um conjunto abrangente de algoritmos e ferramentas para desagregação de consumos. A partir dessa investigação inicial, foi testada uma abordagem mais recente baseada em redes neuronais profundas. Além destes métodos, foi desenvolvida e testada uma nova metodologia centrada na utilização de algoritmos existentes na libraria scikit-learn.

Nesta dissertação são destacadas as principais características de cada abordagem, assim como os pontos fortes e limitações de cada uma. Com o intuito de permitir uma compreensão abrangente dos desafios enfrentados no desenvolvimento de soluções NILM, são também apresentados os principais obstáculos encontrados ao longo do desenvolvimento deste trabalho. Além disso, são propostas algumas recomendações que podem ser adotadas para melhorar a performance dos métodos utilizados. Entre os algoritmos utilizados, é de destacar Random Forest, Gradient Boosting e Support Vector Regression que demonstraram um grande potencial de aplicação no âmbito do NILM.

Palavras-chave: Monitorização não intrusiva de consumo (NILM), desagregação de consumo, detecção de eventos, monitorização de consumo de energia elétrica

Abstract

One potential solution for addressing the pressing issue of global warming lies in the energy efficiency and power consumption of residential buildings. Non-intrusive load monitoring (NILM) emerges as a promising approach to optimize energy consumption by providing detailed insights into individual appliance usage.

This master's thesis focuses on exploring and evaluating various approaches for event detection and classification within NILM. The research begins by testing the NILMTK toolkit, which offers a comprehensive set of algorithms and tools for energy disaggregation. Building upon this initial investigation, a more recent approach based on deep neural networks is examined. Additionally, a novel methodology centered on leveraging the capabilities of scikit-learn is designed, developed, and presented.

The thesis highlights key findings from each approach, discussing their promises and limitations. Detailed accounts of the encountered obstacles throughout the development pathway are provided, allowing for a comprehensive understanding of the challenges faced in the field of NILM. Furthermore, potential improvements and enhancements are proposed in order to successfully perform the energy disaggregation.

In this research, a plethora of machine learning algorithms, both in the categories of classification and regression, are explored as potential solutions for NILM. A collection of solutions proposed in this dissertation - i.e. Random Forest Regression, Gradient Boosting Regression, and SVR - demonstrates a significant potential for advancing the NILM field and energy disaggregation. Moreover, these findings offer promising insights on feasibility of implementation of different approaches as well as the effectiveness of the methods that have been studied in the extent of this dissertation.

Keywords: Non-intrusive load monitoring, Load disaggregation, Event detection, Machine learning, Energy monitoring

Contents

List of Tables

List of Figures

Nomenclature

ADC	Analog to Digital Converter
AI	Artificial Intelligence
ANN	Analog neural networks
API	Application Programming Interface
AUC	Area Under Curve
CNN	Convolutional neural networks
CO	Combinatorial Optimization
CSV	Comma-Separated Values
FHMM	Factorial hidden Markov models
GBR	Gradient Boosting Regressor
GBS	Gradient Boosting Classifier
GPU	Graphics Processing Unit
GRU	Gated Recurrent Unit
HMM	Hidden Markov models
IoT	Internet of Things
LSTM	Long Short-term Memory
MAE	Mean Absolute Error
MSE	Mean Squared Error
NILM	Non-intrusive load monitoring
RMS	Root Mean Square
SNN	Spiking neural networks
STDP	Spike-timing-dependent plasticity

SVC	Support Vector Classifier
SVM	Support Vector Machine
SVR	Support Vector Regressor

Chapter 1

Introduction

Rapidly deteriorating climate and the ongoing process of global warming has spurred an interest and the development of areas of science that did not get so much attention before. One of those fields is demand side management and related to it energy efficiency. Development of those areas was aided by advancement of technologies such as artificial intelligence (AI), internet of things (IoT), smart meters, and smart grids. The common goal is to promote responsible use of resources, conservation of the environment, and improve societal well-being. Energy management is a key aspect of this development, which focuses on the efficient utilization of energy resources to achieve sustainability and self-reliance of energy systems. Nevertheless, as humanity is growing and developing, the consumption of energy is increasing, which has become a challenge, leading to climate change and global warming.

To mitigate these issues, efforts are being made to reduce carbon emissions in various sectors, including buildings, industrial, and transportation. The building sector contributes significantly to energy consumption, and optimizing energy usage in residential/commercial buildings would help reduce carbon emissions. Energy wastage (or just inefficiency) can be avoided by using advanced energy monitoring systems to monitor energy consumption and provide feedback to consumers. In recent years there could be observed a massive roll-out of smart meters being installed in buildings to monitor overall energy consumption. However, in most of the cases, these provide only information at the building level, which may not significantly impact consumer behavior.

One of the ways to address the above mentioned issues is through development and implementation of non-intrusive load monitoring (NILM). Monitoring appliance level energy consumption is expensive due to the installation of energy meters, additional cables, and space requirements for each appliance. By using NILM the cost of implementation and adaptation of energy efficiency measures could be diminished significantly, making it an accessible solution to the vast majority of the residential customers.

1.1 Motivation

The motivation behind this Master's thesis is rooted in a significant concern for the constantly worsening state of global warming and an unwavering commitment to making a meaningful contribution to halt this destructive trend before it reaches irreversible levels. This motivation resonates with the decision made 2 years ago to pursue a comprehensive master's degree in Sustainable Energy Development offered by EIT InnoEnergy[1].

At the core of this research lies the recognition that emission reduction through energy efficiency represents a powerful solution for combating carbon and methane emissions. In this context, investigating non-intrusive load monitoring methods becomes intriguing, as they hold significant promise for reducing energy consumption at the household level. While the emissions reduction achieved from a single household may appear relatively modest, when multiplied by the hundreds of millions of houses worldwide, the cumulative impact would become tangible and highly relevant. As one of the ingredients in this transition, the non-intrusive load monitoring methods could become a key enable towards making this projection a reality.

Moreover, the importance of energy efficiency measures is underscored by their integration into various policy frameworks, such as the European Union directives ”REpowerEU” and ”fit for 55”. These directives highlight the growing emphasis on energy efficiency in the public sector. Additionally, there is a notable surge of interest in energy monitoring, optimization, and efficiency enhancement within the private sector and business domains. In the past couple years prominent companies like Shell and bp are actively acquiring start-ups specializing in home energy monitoring, such as net2grid, bidgely, or auto-grid, to name a few. These companies leverage AI and big data to optimize processes, decrease inefficiencies, and enhance energy consumption of residential customers or distribution and transmission grid networks.

The personal dimension of motivation further reinforces the willingness for choosing this thesis subject. It stems from a recognition of the potential benefits that the entire research and development process can bring to my future career. Prior to starting this endeavor, I had limited experience in data engineering, computer science, machine learning, or neural networks. Taking this leap into uncharted territory has proven to be a rewarding decision, as it has provided me with invaluable knowledge and expanded my horizons immeasurably. I am immensely grateful for the opportunities for growth and learning that this undertaking has provided me.

By addressing the above mentioned dimensions of motivations, this master's thesis seeks to make a valuable contribution to the field of event detection and classification for disaggregation of energy consumption metering data which could ultimately lead to advancing sustainable energy practices and combating climate change.

1.2 Topic Overview

Non-intrusive load monitoring (NILM) is a technique used to determine the energy consumption of individual appliances in a building by analyzing the overall power consumption of the household.

[1] `https://www.innoenergy.com/for-students/master-school/master-s-in-sustainable-energy-systems/`

Although the first mention of NILM appeared around three decades ago, its biggest growth can be observed in the last 10 years. The most promising value that can be brought from applying NILM technology worldwide is the potential to significantly improve energy efficiency and reduce energy costs for all.

NILM is typically implemented using a combination of techniques from signal processing, machine learning, and data analysis. One of the key challenges in NILM is that the power consumption of individual appliances can be difficult to distinguish from the overall power consumption of the building. To overcome this challenge, many NILM algorithms use a variety of techniques such as feature extraction, clustering, and classification to identify patterns in the power consumption data that correspond to specific appliances. Some of the most common building blocks that are used in NILM include:

- Feature extraction: This involves identifying relevant features in the power consumption data that can be used to distinguish between different appliances. Some common features used in NILM include power spectral density, wavelet coefficients, and time-of-use patterns.
- Classification: This involves using the features extracted from the power consumption data to classify the energy consumption of individual appliances. Some common classification techniques used in NILM include decision trees and neural networks.
- Disaggregation: This is the process of breaking down the overall power consumption of the building into the specific energy consumption of individual appliances. Disaggregation can be performed using techniques such as combinatorial optimization, hidden Markov models, and neural networks.

Research in NILM has been growing rapidly in recent years. A lot of work has been done on NILM technique, several datasets have been proposed and a lot of algorithms have been developed in order to improve the accuracy of NILM.

In general, non-intrusive load monitoring is a viable solution that has the potential to significantly improve energy efficiency and reduce energy costs by providing detailed information about the energy consumption of individual appliances in a building. However there is still a gargantuan amount of work and research needed in order to improve the accuracy and applicability of those techniques.

1.3 Objectives and Deliverables

The main objective of this dissertation is to experiment and evaluate the feasibility of implementing non-intrusive load monitoring methodologies and performing the load disaggregation in a specific scenario. This particular scenario refers to a predisposed set of parameters, assumptions, and datasets that were provided at the beginning of this research endeavor. By employing advanced data analytics and machine learning techniques, the aim is to achieve a comprehensive understanding of energy consumption patterns and identify individual appliance-level load characteristics. Furthermore, an evaluation of the effectiveness of multiple classification and regression algorithms applied in the described process of load disaggregation is being executed.

To achieve the main objective, at first, a rigorous evaluation process of the existing NILM implementations is undertaken. These toolkits are meticulously assessed, examining their effectiveness in performing the energy consumption data disaggregating. Through a series of carefully designed experiments, an analysis of their viability for implementation is made. Building upon the aforementioned findings, an original solution tailored to the unique requirements of the studied households is proposed. This solution addresses the identified challenges and limitations encountered during the experimentation with previous toolkits.

In addition to that, this thesis provides a comprehensive introduction to the field of NILM and energy disaggregation. By examining the underlying principles, algorithms, and methodologies employed in NILM, a solid theoretical foundation can be established. This introductory section highlights the importance of NILM in the context of energy engineering, discussing its potential applications, benefits, and challenges. By understanding the current state-of-the-art techniques and approaches, a better understanding can be reached.

In order to successfully perform the necessary experiments and disaggregate given energy consumption data, various libraries, tools, and environments are harnessed. This dissertation provides a detailed overview of the specific libraries and software frameworks utilized, elaborating on their functionalities and explaining their relevance to the research objectives. On top of that and to ensure transparency and the ability for the implementation to be reproduced, the thesis includes a link to a dedicated GitHub repository[2] containing all the necessary code used in the research. This repository serves as a valuable resource, allowing interested parties to access and examine the implementation details, algorithms, and methodologies employed in the described experiments.

Finally, the findings and results obtained from the experiments are presented in both graphical and numerical formats. Through visual representations the thesis illustrates the performance and efficacy of the proposed solution. Additionally, comprehensive numerical analyses and statistical evaluations provide expanded insights into the accuracy, reliability, and efficiency of the solution. This ensures a comprehensive understanding of the experimental outcomes and enables meaningful comparisons between different approaches.

1.4 Thesis Outline

The content of this thesis is structured in the following order. First of all, the genesis of NILM is presented together with a brief demonstration of the process of development of that field. Secondly, a theoretical overview and in depth explanation of the theoretical models that are lying at the core of the non-intrusive load monitoring is shown. Following that an implementation phase is described meticulously in the form of a step-by-step description of the processes and experiments developed for the sake of this thesis. Afterwards comes a presentation of the result and the analysis of obtained values and metrics. The dissertation is finished by a chapter that presents conclusions including the

[2] `https://github.com/mickiewp/thesis_NILM`

main achievements of this study as well as a number of suggestions regarding possible pathways that could be taken as a future work.

Chapter 2

Background

2.1 Literature Review

The genesis of the topic of nonintrusive load monitoring is assigned by many researchers to a specific research paper titled ”Nonintrusive Appliance Load Monitoring” written by George William Hart, and published in the Proceedings of the IEEE in 1992 [1]. It focuses on the concept of nonintrusive load monitoring (NILM) and its potential applications in the field of energy consumption analysis.

As an introduction, the author provides an overview of the need for load monitoring techniques. The importance of understanding individual appliance energy usage within a building for effective energy management is highlighted. At that time, traditional methods of load monitoring involved intrusive techniques that required attaching sensors or meters to each individual appliance, which can be cumbersome and expensive. In contrast, NILM aims to estimate the energy consumption of individual appliances using only aggregate power measurements at the main power supply.

In [1], author describes the fundamental principles behind NILM and presents a comprehensive framework for its implementation. The paper discusses various signal processing techniques, such as time-domain and frequency-domain analysis, which are employed to extract appliance-specific features from the aggregate power signal. Feature extraction methods, such as transient detection, steady-state power estimation, and pattern recognition, are explored in detail.

The author emphasizes the importance of developing accurate and robust load signatures for different appliances, as these signatures serve as reference patterns for appliance identification and energy disaggregation. Hart discusses different approaches for signature acquisition, including supervised and unsupervised learning techniques, and provides examples of signature-based appliance recognition algorithms.

The paper also highlights the challenges and limitations associated with NILM, including the presence of noise, transient events, and overlapping power signatures. The author proposes possible solutions and discusses future research directions to address these challenges.

Furthermore, the author discusses the potential applications of NILM beyond energy monitoring, such as fault detection, security monitoring, and demand-side management. To support his arguments, Hart presents case studies and real-world examples to illustrate the effectiveness and practicality of NILM in these applications.

This paper serves as a foundational reference for researchers and practitioners interested in the field of NILM and its applications in energy management and beyond.

Another publication which is frequently cited and lays at the core of the NILM development is the paper titled: "NILMTK: An Open Source Toolkit for Non-Intrusive Load Monitoring". It introduces NILMTK, a comprehensive open-source toolkit designed for Non-Intrusive Load Monitoring. As already mentioned, NILM is a technique used to estimate the energy consumption of individual appliances within a building by analyzing the aggregate power signal [2].

The authors address the need for a standardized and open-source platform in the field of NILM and present NILMTK as a solution. The toolkit provides a collection of pre-processing, feature extraction, and disaggregation algorithms, along with evaluation metrics and visualization tools. It aims to simplify the development and evaluation of NILM algorithms, fostering collaboration and enabling reproducible research.

The paper discusses the architecture and design principles of NILMTK, highlighting its modularity and extensibility. It emphasizes the use of a common data format, facilitating data sharing and comparison between different datasets and algorithms. The toolkit supports various data types, including high-frequency voltage and current measurements, and accommodates both single- and multi-state appliance models.

Furthermore, the authors provide a detailed overview of the toolkit's core components and functionalities. They describe the data loading process, data preprocessing techniques such as resampling and noise removal, and feature extraction methods like transient detection and statistical features. The paper also presents different disaggregation algorithms supported by NILMTK, including factorial hidden Markov models (FHMM) and combinatorial optimization.

The evaluation metrics and visualization capabilities of NILMTK are discussed, enabling researchers to assess the accuracy and performance of their NILM algorithms. The authors also highlight the toolkit's integration with other open-source projects, such as Pandas and scikit-learn, further enhancing its capabilities and interoperability.

The paper concludes by showcasing the impact and adoption of NILMTK within the research community. It discusses several case studies and research projects that have utilized the toolkit, demonstrating its effectiveness in advancing the field of NILM.

Influenced by rapid development in other machine learning fields like speech recognition or image processing a novel approach to load disaggregation based on deep neural networks is being introduced. In [3], authors adapt existing solutions to non-intrusive load monitoring and load disaggregation purposes. In specific, three deep neural networks are redesigned and presented in this paper. First is based on a form of recurrent neural network called 'long short-term memory' (LSTM). Second utilizes a neural network which regresses the start and end time as well as the average power drawn by the specific appliance during an activation period. Third is an adaptation of denoising already existing solutions for autoencoders. Furtherly, these three neural networks are tested and evaluated on their accuracy and efficiency metrics. Last but not least, the performance scores of these novel disaggregation algorithms are juxtaposed against the conventional approaches like combinatorial optimization or hidden Markov models.

Authors of [3] suggest that non-intrusive load monitoring (NILM) or energy disaggregation are all interchangeable terms that describe a computational technique, whose main goal is to distinguish a specific energy consumption pattern of each electric appliance from a load recording of a whole house metering. It is pointed out that this technique is nothing new in modern research.

The first wave of interest about topics related to load disaggregation came as far back as the 1980s with the seminal work of George Hart [4]. Nevertheless, the technology approximately 40 years ago was on another level than it is now. For example, the first studies were devised using 1 Hz or slower data collection, whilst nowadays the sampling frequency can be performed at the rates of couple kHz and sometimes even reaching MHz values.

A couple of potential use-cases or future implementations are mentioned in this paper. Among them, the most promising are the following: increased performance in grid monitoring and demand side management capabilities, faster identification of faulty appliances inside a household or a factory, more informative way of communicating the end users their energy consumption and a possibility to create more suitable energy quotas from energy companies.

In this specific publication [3], authors want to examine whether the use of neural networks can be applied to the field of non-intrusive load monitoring. Three aforementioned approaches are studied in detail. The idea is to check if the algorithms used and developed in different fields of machine learning could be adapted for the purposes of energy disaggregation and what would be the performance and accuracy of such an adaptation.

One of the huge advantages of this neural network is that once it is trained on publicly available datasets it can be applied to any household without a need of a ground truth data for this building. After the initial training of the algorithms, the network should be capable of disaggregating the energy consumption and be able to locate the working periods of each appliance based on their characteristic working patterns. This feature is particularly useful, as it significantly lowers the effort that a homeowner would have to put into implementation of NILM solutions into his or her household, which is one of the major obstacles on a way of growth of this field.

The [3] highlights that in order to make the disaggregation algorithms more effective there should be more variety in databases and also the data collection should be performed in a more structured way. Using the example of a UK-DALE dataset, it is pointed out that for 5 households recorded, only 3 has a ground truth recording of a washing machine, and from these three examples consumption patterns of different machines and their varying load curves make it hard to disaggregate this appliance with a decent level of accuracy. Authors mention image classification field, and on this base, they suggest that a bigger variety in data available would greatly enhance the performance of the deep neural networks for non-intrusive load monitoring purposes.

Feature selection is essential for the classification section of the NILM methodology application [5]. It determines which elements of the recorded electric signal of the appliance will be used for the sake of distinguishing this specific appliance from other equipments inside the house. Examples of features that could be used include active or reactive power, peak voltage, maximum current, rms (root mean square) current, total harmonic distortion of current, enclosed area of voltage-current trajectories with consideration of trajectory direction [6]. In this publication 33 of such features are listed. Furtherly, they are divided into two groups, steady and transient features.

This paper presents two important contributions to the field of non-intrusive load monitoring. First is a comprehensive review of the existing feature selection approaches. Second identifies which features are bringing the biggest value to improving accuracy and efficiency of the load classification process. On top of that, combinations of a couple of features joined together are taken into consideration and tested on the publicly available PLAID dataset. This experiment concludes that some of the combinations can result in much better outcomes than taking only a single feature.

In general, the authors in [5] compellingly show that selecting the right features to be applied in the load classification process is a very important step.

The single features that are proved by this paper to have the greatest positive impact on load classification accuracy are the following (in the order of highest to lowest impact):

1. Total harmonic distortion of current
2. Enclosed area by VI (voltage-current) trajectory with consideration of trajectory direction
3. Energy of detail wavelet coefficients at i-th scale
4. j-th current harmonic coefficient
5. Normalized real power

It is also mentioned that things like the impedance of a nonlinear load or distinct power consumption patterns in different operating states of multi-state load appliances like a washing machine or a dryer, are furtherly deteriorating the accuracy of disaggregation and classification algorithms. These instances pose a significant obstacle on a way of non-intrusive load monitoring development. The paper lists washing machine, heater, fridge, air conditioner and laptop as the appliances with the worst accuracy, regardless of any factors.

Belonging to another subcategory of NILM methodologies are solutions that depend on event-based disaggregation. Reportedly, it brings greater efficiency and accuracy than the previously presented approaches [7]. Authors of this paper define breakthroughs in smart grids technology, especially smart meters, as a reason behind the increase in popularity of topics like non-intrusive load monitoring. It is mentioned that although the field is growing rapidly, there is still much room for improvement.

Data sampling rate is presented as one of the imperative culprits on the path of NILM development. As far as smart meters are able to collect data at several kilohertz frequency, the problem arises at the step of transmitting this data to the central station where it can be processed and utilized. One of the possible solutions is to send this data from a smart meter through the home area network to cloud based services.

Moving on, authors present that a major chunk of the present research related to non-intrusive load monitoring is still relying on the low sampling rates - 1 sample every 1 to 10 seconds. They are mostly eventless NILM algorithms that are based on probabilistic methods like for example hidden Markov models (HMM) and deep neural networks. Their main premise is trying to predict or guess a state of a specific appliance at each point in time. Nevertheless, this approach is in most of the cases only relatable to big electronic devices like washing machines or electric stove. The drawback of this method is that the computational power needed to perform disaggregation increases significantly with every appliance in the household.

To address the aforementioned issues, authors are pointing towards event-based NILM algorithms. In this scenario higher sampling rates could be harnessed. As a result, features like transient states during the ON and OFF periods, higher order harmonics or even electromagnetic interference could be detected.

Event-based NILM algorithms can be distinguished into two categories. First being event detection and second, event classification. Event detection is focused on finding transient states. For that part it is critical to apply signal filtering and normalization to accommodate for any irregularities, like voltage drops, that can disturb proper functioning of the algorithm. On the other hand, event classification is designed to learn specific load signatures and assign them to corresponding appliances afterwards. Active or reactive power waveform, current waveform of voltage-current trajectories are among the most popular features to be employed for that.

Authors in [7] point out that two identical appliances could have different load signatures, which further complicates the processes of disaggregation and classification. For example, a washing machine designed by one manufacturer could have a varying energy consumption pattern than a washing machine produced by some other company. Even if there would be two identical washing machines, but they were working in different washing modes, one could be mistakenly classified by the algorithm as some other appliance, like a notebook or an airfryer.

Although event detection algorithms incorporating probabilistic methods for load disaggregation can work more efficiently than other solutions, authors highlight that these still require a training process and labeling. These two aspects are increasing in a substantial way the burden for the end-user and complicates the implementation.

The methods implemented in [7] introduce a new way of creating a load signature of appliances. The novel method is based on active, reactive and distortion power trajectories. To compose this vector a transient state as well as adjacent steady states right before and right after the transient are taken into consideration. As it is proved in the paper on the example of around 100 distinctive households and appliance brands, this approach is able to reach 98 % recognition accuracy even if the same type of appliance is placed in a different household or manufactured by different companies. As a result, the algorithm is uncannily versatile and relatively easier to apply. Furthermore, the scalability of the NILM algorithm is greatly enhanced thanks to that.

In the similar subcategory of event-driven methodologies is a following research paper which brings a novel approach that enhances the usability and effectiveness of the non-intrusive load monitoring [8].

This method is based on a convolutional neural network (CNN). Additionally, the event-driven detection procedure helps to locate and classify start and stop points of electric appliances through.

The authors in [8] mention advancements in smart metering and increasing climate-consciousness as main reasons for growth in popularity of load monitoring and disaggregation. They confer that the development of the NILM processes is beneficial for both consumers and producers. From household perspective, by knowing their consumption patterns people can actively switch their most energy intensive to the off-peak hours, therefore lowering their expenditures for electricity. From the side of energy companies, either working in generation, transmission or distribution, the management of the energy and balancing the grid is facilitated.

What this paper is using as an input to feed into the NILM algorithm is the raw current data obtained from the electricity meter. These current waveforms are then transformed into gray-scale images for each appliance. Later, pattern recognition is applied to locate the start and stop moments in time per each appliance.

Authors are confident that thanks to using this approach the problems like data dimension reduction or information loss are avoided. Therefore the higher sampled current data can be utilized. That, as a consequence, brings the accuracy of the identification results to a higher level. Moreover, computation complexity and computation time are taken into careful consideration throughout the whole process of development of this new algorithm.

In opposition to the previous two academic publications - [7] and [8] - comes the paper, which points out that the majority of the existing publications is focusing on event-based approaches for disaggregation purposes. This means that the algorithm is looking for a special event, which for example can be a turning on of the appliance and based on the pattern of this particular transient state it can recognize this appliance and discern it from other electrical equipment in the house. On top of that, it is highlighted that these methods cannot reach high enough disaggregation accuracy without obtaining specific load consumption sub-metered data per each appliance for the training purposes. Therefore, it is not suitable for real-world applications [9].

To address these issues, the paper presents a non-event-based probabilistic model. This approach focuses on the whole load consumption patterns and not only on the specific points. It learns specific parameters of each appliance model and takes into consideration load signatures. In the final step, it utilizes those two components to recognize state transitions of appliances within the aggregated load consumption recording.

This study narrows down the scope of disaggregation to the office environment. Even more specifically, it examines a scenario where multiple computer screens, laptops, lamps and other desk level equipment are connected to the same power socket.

In this particular study, the authors decided to examine a model based on probabilistic methods, which is called the factorial hidden Markov model (FHMM). It is presented that is able to disaggregate desk level appliances of both characteristics, single-state (in this paper called 'binary state' or ON/OFF) like lamps, or multi-state like computers. The paper points out that the majority of existing publications focuses on detecting single-state appliances, which is not practical, because much of the equipment used in daily life is operating in more elaborated fashion.

Moreover, the method herein presented is using low-frequency data measurements. Consequently, scalability and introduction of this solution to the market would be greatly facilitated.

It is shown in the paper that this approach, after sufficient training of the model, would be capable of performing a disaggregation process in real time. It is a major distinguisher from other proposed solutions. This feature opens the door to deploying the system in a form of ready to use live application.

After performing intricate experimentation of the proposed non-event-based approach to load disaggregation, authors present that in case of single-state appliances the accuracy is around 90%, whereas with multi-load appliances this value drops to 80%. Furtherly, it is shown that the accuracy decreases significantly with every new equipment added to the load. This makes the approach impractical for a household application, as usually houses contain more than two or three appliances that would have to be disaggregated if non-intrusive load monitoring were to be implemented.

The main goal of the subsequent publication is to introduce a new NILM algorithm based on deep learning and benchmark it using publicly available datasets and compare the results with other already existing algorithms to show whether it is indeed more effective than the past approaches. This breakthrough method harnesses the possibilities of unsupervised learning based on spiking deep neural networks [10].

The authors, as a motivation for their work, point out three distinctive impediments of the current disaggregation solution. First is limited communication, which relates to a problem highlighted in other research papers, that is data acquisition and data transfer. Second shows that implementing non-intrusive load monitoring methods is not yet systematized therefore there is a necessity to adjust and accommodate the features with respect to particular household or facility. This significantly amplifies the amount of time and effort needed. Consequently, the cost also rises making the whole project uneconomic. Last but not least, the issue of priceyness of highly specialized hardware is raised. This in effect further expands the costs of implementation.

Authors of [10] ascribe a recent uptake in the NILM popularity to the emergence of the field of Internet of Things, and specifically the smart plugs, which are able to measure voltage and current usage per each device in the household. This fact makes the process of disaggregation much easier. Nevertheless, it is called to attention that implementing this on a large scale is unfeasible. The main reason being that to collect and process influx of data from dozens of smart plugs belonging to the local network would require powerful computational units, like graphic processing units and such. The solution to this part of a problem is proposed to be a simple component built from a resistor and capacitor, which imitates the process of neurons sending electric signals inside the brain.

Another problem is related to the field of machine learning and deep learning, specifically the topics of data compression and computation cost reduction. Although methods of reducing this load are already existing in case of NILM and sample frequency of 20 kHz for example would still be a considerable burden even if lowered by 80%. On top of that comes the struggle of training the neural network in a supervised learning manner so it would be adjusted to each specific household respectively.

The solution that is proposed by the authors is a new deep learning scheme, which is called 'spiking neural networks (SNN)'. It is based on a mechanism in which neurons in the human brain are sending signals between themselves. Specifically, this approach mimics the flow of an electric impulse that is discharged between synapse and dendrite and how it activates a specific neuron as an effect. Authors argue that this interplay that is happening in the brain can be resembled by an elementary electric circuit build from a resistor and a capacitor (RC circuit) and nothing else. Moving forward such a neural network could be processed by an integrated circuit like metal-oxide-semiconductor-field-effect transistor [11]. This would significantly bring the costs of implementing NILM methodology down, as the usage of sophisticated GPUs could be abandoned.

What is more, the training of this neural network would also become much easier. The only input that would be needed from the user would be a one time labeling post training (spike-timing-dependent plasticity (STDP)). In theory, having multiple neuron excitement patterns obtained from the process of training the neural network the final user would have to assign one pattern to one appliance and from this moment onward the algorithm would be able to perform disaggregation and classification of the load on itself.

Lastly, this approach addresses the problem of extremely high frequency of data collection and transmission that is necessary for developing non-intrusive load monitoring algorithms. Furtherly, this solution also facilitates to perform a more efficient distributed computation. By presented example, it is shown that it is easier to compress binary vectors used by SNN than an 8 or 16 bits of analog data used in currently prevalent analog neural networks (ANN). Thanks to this, a first layer of processing data could happen on a smart outlet and afterwards a refined/filtered data would be sent to a local server - the second layer.

A comprehensive review that presents the state-of-the art methodologies in the NILM field comes from [12]. This paper similarly to [2] links a development of non-intrusive load monitoring field to increased prevalence of smart meters. Moreover, it also adds noticeable progress in the areas of artificial intelligence, and smart grids as vital causes behind this growth. All in all, authors assign these components to a major category of smart sustainable cities, which promotes efficient use of resources, improving the overall well-being of its citizens and environmentally conscious solutions. Following that, it is highlighted how important is the development and improvement in this specific category in relation to the huge ongoing process of global warming and how it could help in its mitigation.

In short, this paper does not contribute any new method or algorithm to the field of non-intrusive load monitoring. It serves as a review and a big picture summary of what has been achieved up until this point, what are the state-of-the-art solutions being introduced and developed at this point in time as well as what are the outlooks for the future. It also provides a plethora of explanations of basic concepts, which in effect makes it easier for a reader who is not accustomed with the field of NILM to follow the contents of this paper.

In [12], authors are referring to findings published by [13], which implies that due to increase in global population the surge in energy consumption can be observed. This ever-growing demand would finally result in shortages or blackouts leading to overall energy crises, which will harm economies of the whole countries and its populations. Based on this line of reason the point is made of how important the field of energy management is. By in detail monitoring of present energy consumption and analyzing the past records an energy optimization algorithm can be applied. As a result, any inefficiencies or losses might be avoided. That is also where the topic of non-intrusive load monitoring would come into the picture.

There are two approaches to lowering the intensity of carbon emission related to the building sector. One is to create well insulted, energy efficient building, which are made from sustainable materials as 4-8% of total global CO2 emissions come from concrete [14]. The other option is to improve the energy efficiency of already constructed buildings. As calculated by [15], in 2021 emissions related to the operation and construction of buildings accounted for 30% of global final energy consumption.

The paper mentions a couple of thought-provoking findings, which points out that by monitoring energy usage and providing feedback to the consumers regarding their consumption patterns and how they can improve them there is a potential of lowering their energy usage. Nevertheless, there are different values presented in specific papers. In one of them, it is mentioned that this reduction could drop by 3% [16]. Meanwhile the other study calculates that this percentage of potential energy savings can grow to as much as 9% [17]. On the other hand, analyses made by [18] show that there is no reduction in energy consumption whatsoever.

As an introduction, the authors are presenting four building blocks of the non-intrusive load monitoring, which are data acquisition, event detection, feature extraction and finally energy disaggregation. It also highlights that most of the methods are based on artificial intelligence or machine learning. Moreover, the training of the algorithms can be divided into supervised or unsupervised. In order to detect a specific energy consumption waveform of an appliance, also called signature, one can refer to either steady state or transient features.

A vital point that this paper is raising is the role that sampling rate of the data is playing in the whole process. In a perfect scenario, without taking into consideration all other factors or things, the higher the sampling rate the better. Nevertheless, there is an issue related to that. To obtain higher sampling there would be a necessity of more advanced and therefore more expensive metering devices. On top of that, gathering, sending, and processing such huge amounts of data would additionally drive the expenses even higher. This stands in direct opposition to the basic premise of non-intrusive load monitoring, which intends to make it more accessible and affordable for the end-users to get feedback on their energy usage. As a name states itself, it should be 'non-intrusive'.

The benefits of a successful implementation of NILM are presented here from a two main points of view. As a household owner, by getting detailed feedback on an energy consumption for specific appliances in your house, you could adjust your consumption patterns and start applying energy efficiency practices. What's more, it would be possible to detect any deviations from a normal operation of any appliance. From a perspective of a utility or an energy service company introduction of NILM would potentially facilitate operations related to demand side response. Detecting faults in a grid would also happen quicker and it would be much easier to find the points of failure.

Beyond already mentioned applications of the NILM methodology in smart grids and distribution energy resources, authors also ponder about the opportunities for the future. Two particularly interesting and at first glance non obvious areas are mentioned. The first is related to the military and the second with elderly people. As for the military, the shipboard of the US Navy is given as an example. It is written that by installing proper measuring devices and applying NILM techniques any problem of the electrical system or an electrical machine could be found and repaired much faster. The other creative implementation would be into assisted living of elderly people. Through non-intrusive load monitoring of households in which seniors are dwelling alone, it could be monitored and detected if they are doing fine, by looking at anomalies in their daily usage of home appliances.

Besides numerous advantages of NILM and its vast potential for making a positive impact on human development, there are also a couple of drawbacks that should be mentioned:

1. Data Management Challenges: One of the primary challenges in NILM lies in the process of data collection, transmission, and storage. The ability to capture appliance signatures with extremely high sampling frequency is crucial for accurate energy disaggregation. However, the efficient transmission of such vast amounts of data to the cloud over wireless broadband remains a significant bottleneck. Furthermore, the potential for data losses during each step of the process raises concerns regarding the reliability and integrity of the collected data. Addressing these challenges is essential to ensure the effectiveness of NILM systems in real-world applications.

2. Technology Immaturity: Despite the rapid growth witnessed in the field of NILM in recent years, it can still be considered in the research phase of technology development. To achieve widespread adoption, there is a need for specialized hardware specifically designed for non-intrusive load monitoring purposes. Just as the introduction of GPUs revolutionized computer gaming, graphics, video processing, and artificial intelligence, a similar shift is required to propel NILM into the mass market. The ideal solution should be easily accessible, user-friendly, and cost-effective, enabling seamless integration into existing energy management systems.

3. Algorithmic Complexity: NILM heavily relies on machine learning algorithms to disaggregate energy consumption. However, the complexity and computational demands associated with training these models pose significant challenges. Training accurate and reliable disaggregation models can often take days, and user input may be required, making the process time-consuming and costly. Overcoming these challenges is crucial to develop more efficient algorithms that can deliver accurate appliance-level energy consumption without excessive computational requirements and user intervention.

4. Consistency in Data Formats and Methodologies: While efforts have been made to standardize data formats and methodologies in NILM research, the field remains somewhat chaotic. The lack of consistency poses challenges for researchers and practitioners in navigating the existing literature and developing meaningful contributions. Achieving a higher level of standardization will enable researchers to build upon each other's work, fostering collaboration and advancing the field more effectively. Establishing common protocols, data formats, and evaluation metrics will streamline the development and evaluation of NILM algorithms, ultimately leading to more robust and reliable solutions.

2.2 Experimental Setup

2.2.1 Dataset description

The data was collected and provided by a Portuguese electric utility. A set of hardware that could potentially be deployed for the sake of this project is presented on Figure 2.1. Presumably, using such devices it would be possible to collect, record and send the necessary measurements. In the top-left part of the figure are presented two smart plugs units that could collect specific data from each equipment inside a given household and provide the so called 'source of truth', a crucial element for performing load disaggregation. Next to them, in the top-right corner of the figure is a smart meter that could measure and record the data from the power consumption of the whole house providing a specific part of the dataset that is later in this thesis referred to as 'mains' or 'raw data'. At the bottom of the figure, a communication device can be seen, which possibly is used for preliminary validation of collected data, packaging this data into tangible files, and sending them to the assigned directory.

Figure 2.1: An ensemble of edp re:dy products that can be used for the home monitoring purposes. source: https://www.sato-project.eu/news/edp-new-completes-demonstration-activities-in-seixal-residential-pilot

The data was collected from two single family households. In the first one, called "House 0", there lives a family of four (2 adults and 2 kids). On the other hand, the second house, referred to as "House

1” is inhabited by three people (2 adults and 1 children). It was mentioned in the internal communication that both of the houses are located in Portugal. No further detail related to a more precise location had been provided.

In theory, the time frame of power consumption of each of the houses should span for exactly 1 week or 7 days. Meanwhile, reality turned out to be slightly different. The data was sent as a collection of folders containing recordings from each day from 30 September 2021 till 11 October 2021. According to datasheets there should be data recorded for the House 0 starting at 00:01 on 30 September 2021 and ending at 23:59 on 06 October 2021. For House 1 the start date of data collected is at 00:01 on 5 October 2021 and ends at 23:59 11 October 2021. Nevertheless, there was only one folder for a day of 05/10/2021 and one for 06/10/2021, instead of two pairs - one for House 0 and the other for House 1 respectively. As it turned out, the data contained for those two days indicated above was related to the recordings from House 1. Therefore, instead of having 7 day recording for House 0, there was only 5 days of data available for this household.

Separately to this recording from the mains, another set of data was provided. It was sent in the form of two csv files, first for House 0 and second for House 1. Each of the files contained energy measurements from special smart-plugs that were monitoring power consumption of specific appliances or groups of appliances inside the house. These recordings were meant to serve as the “source of truth” in the subsequent part of this research.

For House 0 these devices are bundled in the following groups:

1. ‘Fridge + Freezer’
2. ‘Oven + Microwave’
3. ‘Bimby / Kettle / Air fryer’
4. ‘Stove’

For House 1 these devices are:

1. ‘Fridge + Freezer’
2. ‘Washing Machine’
3. “Microwave / Coffee Machine / Kettle / Toaster’

Beside the above mentioned electric devices that were monitored, there were also other non-monitored equipment that are important to list, because they are also a part of the energy consumption of the whole house and therefore they might influence the experiments with disaggregation and classification undertaken on this house. For House 0 these devices are: TVs, TV Box, Sound Bar, 2x Laptops, Sandwich Press Grill, Toaster, Electric Blinds, Garage door electric monitor. On the other site, in the House 1 the unmonitored equipments are: TVs, TV Box, MiniPC, Game Console, 2x HVAC, Bathroom Heater, 2x Laptops, 2x Displays.

In the following part of this research the predicted values from either classification or regression experiments were juxtaposed to the "source of truth" files to evaluate effectiveness of aforementioned algorithms.

The time resolution of those data recordings is 1 sample per minute. On top of that, there data measurements in more sparse resolution of 1 sample per 15 minutes was also provided. Nevertheless the decision was made to omit them as such resolution would not give any valuable insight from the non-intrusive load monitoring and energy disaggregation perspective. Table 2.1 provides a brief overview, presentation and comparison of the most popular datasets in the field of NILM, which are listed next to the dataset incorporated in this dissertation.

Table 2.1: A collection of the most popular datasets used for NILM

Dataset	Country	Buildings	Duration	Sampling	Publication
REDD	United States	6	2-4 w	I,V: 15 kHz P: 1 Hz OA: 4 s	[19]
BLUED	United States	1	1 w	I,V: 12 kHz P,Q: 60 Hz OA: 1 Hz	[20]
UK-DALE	United Kingdom	5	655 d	I,V: 16 kHz P,Q: 1 Hz OA: 6 s	[21]
PLAID	United States	55	n/a	30 kHz	[22]
iAWE	India	1	74 d	1 Hz	[23]
DRED	the Netherlands	1	6 m	1 Hz, 1 min	[24]
this thesis	Portugal	2	5-7 d	I,V: 7.5 kHz OA: 1 min, 15 min	n/a

I = current, V = voltage, P = active power, Q = reactive power, OA = one appliance

The dataset obtained from the external entity was shared in compressed gzip file format. Gzip is a popular data compression and file packaging format. It is commonly used to compress single files or concatenate multiple files into a single compressed archive. Consequently, the initial task involved decompressing all the folders to obtain a readable dataset that could be utilized for the purpose of this dissertation. Additionally, it was necessary to execute the parsing step.

After performing aforementioned steps, the datasets were aligned in the following order. The whole recording from 30 September 2021 to 11 October 2021 was divided into folders containing 4 hours of data recording. First folder being called "20210930_00h_04h " and the last one "20211011_20h_24h". One folder contains either 240 elements, that are 1 minute recordings of a mains signal in a csv format.

An exemplary name of such a file is the following: "20211005_040250". As a sampling frequency is equal to 7.5 kHz, inside each singular file there are 450001 rows of data, with the first row being a description of each column. First column is a timestamp, which denotes a specific minute at which data was collected, written in a 10-digit epoch time format. Second, third and fourth columns contain measurements of instantaneous voltage, instantaneous current, and instantaneous power respectively.

One csv file has a size of approximately 10.8 MB. Having 240 files in one folder and a total number of 72 folders the aggregated size of all the data from mains recording collected accumulates to around 176.4 GB worth of data.

The initial observation and inspection of dataset quality is performed at the point of collection of data, which in this specific case is inside the given household. Among the actions that are performed to ensure a decent standard of data are: checking max absolute and delta values, performing continuity checks, constant monitoring of the frequency of mains.

It is useful to present the internal structure of each 1-minute file, so that a better understanding of it and its impact on the experiment can be reached. The file is divided into 12 columns named: 'id', 'collectionDate', 'macAddress', 'messageType', 'operationType', 'localId, date', 'ActiveEnergyConsumed', 'AverageVoltage', 'ActiveEnergyInjected', 'ActiveEnergyProduced', 'ActivePeakPowerConsumed'. Each column has a length of around 10060 rows, where the number of rows is varying slightly from appliance to appliance due to some missing recording in those 1-minute files. It is necessary to mention that from the 5 last columns that are directly related to the energy measurement indicators only one called 'ActiveEnergyConsumed' contains any data and the other 4 are empty.

On the other side, the structure of data recordings from the mains was divided into one column containing a timestamp signature and following that additional 3 columns called: 'voltage', 'current', 'power'. Each of those 3 specific energy measures recordings contains 450000 rows. As it is elaborated in the following chapter of this dissertation at the latter stage of the experiments the lengths of 1-minute files and the raw data file containing recordings from main had to be adjusted in a proper manner.

2.3 Tools and methods

2.3.1 NILMTK toolkit

One of the most popular tools in the field of non-intrusive load monitoring is a toolkit called: NILMTK. It is an open-source package of functions and algorithms, whose goal is to enable the comparison of energy disaggregation algorithms in a reproducible manner. It was developed and published in [2] by researchers from Indraprastha Institute of Information Technology Delhi, Imperial College London, University of Southampton, Columbia, and UCLA. Although the paper was published almost a decade ago, to this day it is one of the most widely used resources and references in the field of NILM.

This paper is a pioneering work on the possibility for comparing a variety of disaggregation approaches among multiple data sets, which are publicly available. On top of that, the toolkit presented in the paper is also equipped with supplementary tools and functionalities that are intended to facilitate the comparison process even further. In the initial version the tools at the disposal are:

- data preprocessing - checking continuity, deleting repeated values;
- parsers, which helps to adapt the datasets to format and proper layout that can be then inputted to the disaggregation algorithms;
- two reference benchmark algorithms: Combinatorial Optimization and Factorial Hidden Markov Models;
- accuracy metrics, which helps to assess how effective the algorithm and the process is (True Positive, Recall, Precision, etc.).

It is pointed out that although the topic of nonintrusive appliance load monitoring was primarily introduced in the 1990s, there still is an extensive field of unexplored opportunities and functionalities waiting to be discovered. It is mentioned that in the last couple years prior to the publication of this paper, the topic of non-intrusive load monitoring regained interest and attractiveness due to the fact of increased prevalence of smart meters as well as decreased cost of electricity sensors applicable to retail consumers. This rise in popularity could be quantified by the number of papers released on the topic as well as a number of startups being founded at that time.

The crucial problem that is singled out by the authors of this paper is that the comparison between different disaggregation algorithms is virtually impossible. Consequently, the speed and quality of further development of the non-intrusive load monitoring is significantly suppressed. There are three main issues presented as a component of this problem. Firstly, the variability of data sets used, appliances measured, sampling frequency, measurement location, time frame and a plethora of other aspects are all responsible for creating bigger and bigger divergence among obtained results. These differences make it excruciatingly difficult to reproduce given experiments. Another side to the problem is that there are no benchmark algorithms publicly available. As a result, there can be no juxtaposition of an algorithm used, and therefore it cannot be said whether a presented approach is more or less accurate and effective than an alternative one. Thirdly, the lack of commonly used accuracy and performance metrics for evaluation of a given approach prevents a direct and precise comparison between two or more disaggregation algorithms. On top of that, it was observed that due to the lack of some benchmark implementation examples with open-source access, a phenomenon of reimplementation is fairly common in the field. This directly leads to loss of time and resources. In effect it impedes the growth of the NILM area of research in general.

Having all these impediments in mind the authors are introducing an open source tool, which they call: NILMTK. The creation of such a toolkit is furtherly motivated by references of such tools already existent in other fields, such as 'PhysioToolkit' in the biomedical research community, or 'CRAWDAD' related to the field of wireless networks.

The authors mention 3 main attributes of the NILMTK toolkit, which are believed to be giving the tool its unique advantage. First is the fact that the toolkit and its documentation are released as open-source software. It can be easily accessed through the Github platform[1]. This way the barrier to entry to the NILM related topics is believed to be lowered, thus possibly spurring the interest and accelerating general development of the field. Second is a modular design and clear API design based on scikit-learn machine learning software. Third attribute is that most of the code is written in Python language, which is one of the most popular programming languages in the world.

Among a series of contributions of the NILMTK toolkit it is worth to highlight a few. One of them is an introduction of a uniform data format called NILMTK-DF, which thoroughly described and evaluated in [25] or [26]. A conception of this format is supported by the fact that there was, and still is, a major diversity of data formats in which measurements of energy consumption of whole households and then specific appliances are recorded and stored. Consequently, it is almost impossible to use one data set as an input to the disaggregation algorithm that was created for an entirely different data set. Another contribution is that inside the toolkit there are two disaggregation algorithms provided. The first is based on combinatorial optimization and the second on the factorial hidden Markov model. They can serve as a benchmark for any new disaggregation approach proposed by future research. On top of that a variety of functions needed to perform an initial data preprocessing are also provided. Lastly, a couple of accuracy metrics are proposed, which are also hoped to be serving as a benchmark for the sake of future development of the NILM field.

Combinatorial Optimization

Having in mind that combinatorial optimization (CO) is one of the two methods included in the NILMTK package it is worth briefly mentioning the basic logic behind it. In this way, this method could be superficially compared with other algorithms implemented in this experiment.

The main application of combinatorial optimization is to find the best solution among a set of possible solutions for a problem with discrete variables. This area of mathematics and computer science seeks to optimize an objective function over a discrete set of feasible solutions by finding the best combination of variables that allows to reach a desired output [27].

Examples of problems that can be solved using combinatorial optimization techniques include the traveling salesman problem, where the objective is to find the shortest possible route that visits a set of cities and returns to the starting point or the graph coloring problem, which aims to find the minimum number of colors required to color the vertices of a graph such that no two adjacent vertices have the same color [28].

Combinatorial optimization algorithms use various techniques to search the solution space, such as branch and bound, dynamic programming, genetic algorithms, and other methods. These algorithms can help solve complex optimization problems that would otherwise be impossible to solve using brute force methods.

[1] `https://github.com/nilmtk/nilmtk`

Factorial Hidden Markov Models

The other of the two methods on the disposal inside the NILMTK toolkit is Factorial Hidden Markov Models (FHMM). This algorithm originates from the one called Hidden Markov Models (HMM). To better understand the latter it is best to start with the HMMs.

A HMM is a statistical model that is used to model systems that are assumed to be Markov processes with hidden states. In other words, HMMs are models that are used to describe situations where you observe a sequence of outputs (e.g. a sequence of words), but the underlying state that generated those outputs is not directly observable. Each hidden state has an associated probability distribution over the possible observable outputs. The probability of moving from one hidden state to another is also governed by a set of probabilities [29].

A Markov process is a mathematical model that describes a system in which the future state of the system depends only on its present state and not on any of its previous states. It is a memoryless stochastic process which only takes into account the probability distribution of the future state. A Markov process is typically defined by a set of states and a set of probabilities that describe the probability of transitioning from one state to another. These probabilities are often represented as a transition matrix.

On top of that the FHMMs were created. Both FHMMs and HMMs are probabilistic models used for sequence modeling. However, there are some important differences between the two. The main difference is that FHMMs allow for multiple, parallel, and independent hidden state sequences, while HMMs only have one hidden state sequence. In other words, FHMMs can model interactions between multiple sequences, while HMMs can only model one sequence at a time. FHMMs are often used for modeling complex time series data, where there may be multiple sources of variation or multiple related time series that need to be modeled together.

2.3.2 Scikit-learn

Scikit-learn is a widely-used machine learning library in Python that provides a comprehensive set of tools for data analysis, predictive modeling, and pattern recognition [30]. It is built on top of NumPy, SciPy, and matplotlib, which are fundamental Python libraries. Scikit-learn is designed to be user-friendly, efficient, and accessible to both novice and experienced machine learning researchers.

The scikit-learn library offers a wide range of algorithms and functionalities for various machine learning tasks, including classification, regression, clustering, or model selection. It supports both supervised and unsupervised learning approaches, making it suitable for a diverse range of applications. With its extensive documentation, rich set of examples, and an active community, scikit-learn is an excellent resource that can be deployed with a considerable straightforwardness by anyone wanting to conduct research in the field of machine learning [31].

One of the key strengths of scikit-learn is its emphasis on simplicity and consistency. It provides a unified and intuitive interface for working with different algorithms, making it easy to create a variety of experiments and compare different models. The API follows a consistent structure, allowing users to seamlessly switch between algorithms without needing to rewrite their code. This consistency also extends to the representation of data, where scikit-learn relies on the widely-adopted NumPy array format, ensuring compatibility with other scientific computing libraries.

Scikit-learn incorporates a wide range of machine learning algorithms, spanning from traditional techniques to state-of-the-art methods. All the models are implemented with efficiency in mind, leveraging optimized numerical operations from underlying libraries. The library also includes preprocessing modules for data transformation, feature extraction, and scaling, which are crucial steps in the machine learning pipeline. On top of that, scikit-learn provides tools for model evaluation and validation. It offers metrics for assessing the performance of models, cross-validation techniques, and hyperparameter tuning methods to optimize model settings. These evaluation and validation procedures enable researchers to make informed decisions about their models and ensure reliable and reproducible results.

Scikit-learn is a powerful and versatile machine learning library that simplifies the development and deployment of machine learning models. Its ease of use, extensive documentation, and broad range of functionalities make it an essential tool for researchers and users in the field.

2.3.3 K-fold cross-validation

K-fold cross-validation is a popularly used approach for effectively evaluating and validating predictive models. It addresses the ubiquitous in machine learning challenge of increasing the performance of predictive models, which are crucially dependent on the quality and quantity of the available data. Such obstacles are especially pronounced when the dataset is limited. This specific cross-validation technique achieves the task by partitioning the available data into k subsets or ”folds” [32]. The underlying methodology that drives this technique can be described sequentially.

First step is the data partitioning in which the original dataset is divided into k roughly equal-sized subsets or folds. Each fold contains an approximately equal distribution of samples. Following that comes the iterative training and evaluation. At this point the model is trained and evaluated k times. During each iteration, the model is trained on k-1 folds, while the one remaining fold is left aside as a validation set to later on evaluate the model's performance. Thirdly, there is a step of performance aggregation. All of the performance measures (e.g. accuracy, precision, recall, or F1 score) obtained from each iteration are at this time averaged or combined and respectively an overall assessment of the model's performance is computed. Last but not least the model selection and hyperparameter tuning can be performed. By systematically evaluating models across multiple folds the k-fold cross-validation helps in selecting the best-performing model or finding optimal hyperparameter settings.

There are a number of advantages of k-fold cross-validation. First to mention would be its robustness in performance estimation. It provides a more reliable estimate of a model's performance compared to a single train-test split. It also reduces the risk of overfitting or underfitting due to the randomness of the train-test split. Another advantage would be the efficient use of data. As each sample gets to be part of the training set and the validation set in different folds this validation approach utilizes a larger portion of the available data for training. This is especially beneficial when the dataset is limited. One more advantage to mention is an overall better model generalization. By evaluating the model on multiple validation sets k-fold cross-validation gives a better indication of how well the model will perform on unseen data. It provides a more realistic assessment of the model's ability to generalize, which indicates how successful this model would be when applied to a new untrained dataset.

In essence, k-fold cross-validation is bringing an enormous value into evaluating and validating predictive models in machine learning. By partitioning the data, iteratively training and evaluating the model, and aggregating performance measures, it provides a reliable estimate of a model's performance and helps in selecting the best model for the specific task. Its efficient use of data and ability to assess generalization make it an exceptionally popular choice for model evaluation.

2.3.4 Classification

Classification is a process in machine learning that involves assigning data points into different classes or categories. It is a type of a supervised learning algorithm that is used to predict the class label of a new data point based on the information contained in a set of training data, which consists of a set of features or attributes and the corresponding class labels. The classification algorithm learns the relationship between the features and the class labels and uses this knowledge to predict the class label of new data points [33].

Classification algorithms can be broadly categorized into two types: binary classification and multi-class classification. In binary classification, the task is to predict the class label of a data point that belongs to one of two possible classes. In the context of NILM would be for example, predicting whether a specific device is turned on or off at a certain moment in time would be a binary classification problem. In multi-class classification, the task is to predict the class label of a data point that belongs to one of several possible classes. For example, predicting the type of an electrical equipment based on more than one characteristic, like current curve, active and reactive load profile, voltage drops, etc.

Classification is used in a wide range of applications, including image and speech recognition, fraud detection, sentiment analysis, and recommendation systems. It is a powerful tool that enables machines to make intelligent decisions based on the patterns and relationships in the data. However, the accuracy of the classification algorithm depends on the quality of the training data, the choice of features, and the selection of the appropriate classification algorithm for the specific problem at hand.

Logistic Regression

Logistic regression is a popular type of a classification algorithm, where the goal is to predict a binary outcome (e.g., true/false, 1/0) based on a set of input features. A simplified operation of classification can be seen on Figure 2.2.

The basic idea behind logistic regression is to use a logistic function (also called the sigmoid function) to transform a linear combination of the input features into a probability value between 0 and 1. The logistic function takes any real-valued number as input and returns a value between 0 and 1. The output of the logistic function can be interpreted as the predicted probability of the positive class [34].

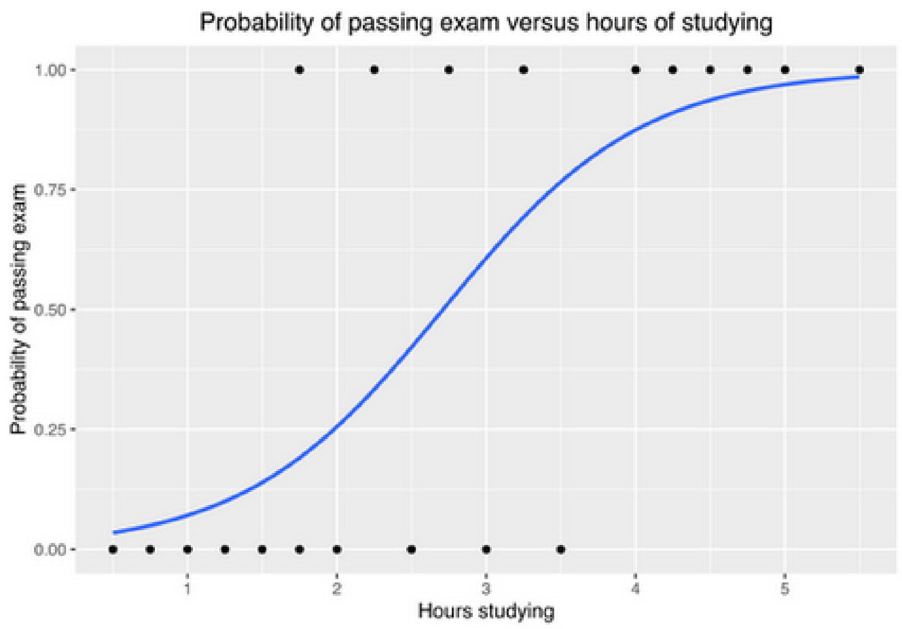


Figure 2.2: Example graph of a logistic regression curve fitted to data. The curve shows the probability of passing an exam versus hours spent studying. source: `https://en.wikipedia.org/wiki/Logistic_regression`

The goal of training a logistic regression model is to learn the optimal values of the weights of the logistic regression model assigned to each input value such that the predicted probabilities are as close as possible to the true values in the training data. This is typically done by minimizing a loss function such as the binary cross-entropy loss using gradient descent or maximizing the Area Under Curve (AUC).

Once the logistic regression model is trained, it can be used to predict the probability of the positive class for new input data points, and a decision threshold can be set to convert the probabilities into binary class labels.

Random Forest Classifier

Random forest is a type of ensemble learning algorithm that combines a multitude of decision trees to create a more accurate and robust classifier. The idea behind random forest is to build a large number of decision trees, each trained on a random subset of the training data and a random subset of the input features, as it can be seen on Figure 2.3 The final prediction of the random forest is then based on the majority vote or average of the predictions of the individual trees [35].

The first step in building a random forest classifier would be a random sampling process in which a random subset of the training data is selected to train each tree in the forest. This as an effect helps to reduce overfitting by introducing diversity into the training data for each tree.

Following that is a random feature selection. At this point at each node of each decision tree, a random subset of the input features is considered for categorizing the data. This helps to reduce the correlation between the individual trees and make the random forest more robust to noisy or irrelevant features.

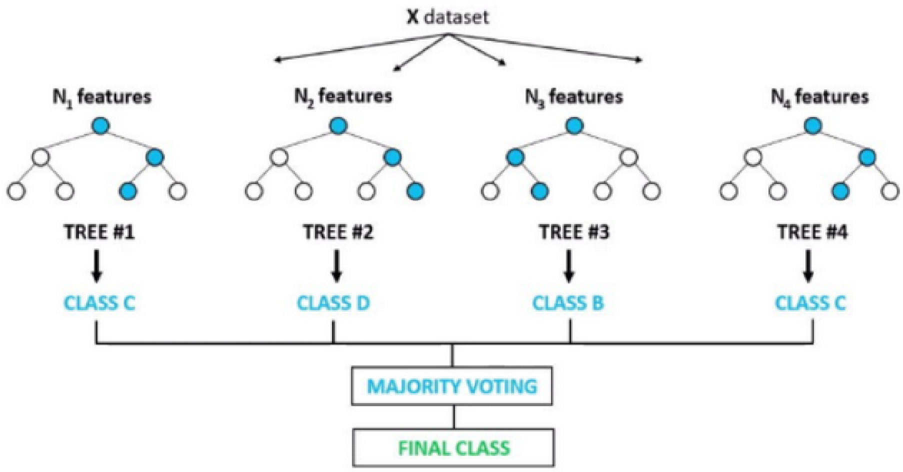


Figure 2.3: Graphic simplified representation of Random Forest algorithm. source: `https://www.freecodecamp.org/news/how-to-use-the-tree-based-algorithm-for-machine-learning/`

After that ensues a step of building decision trees which is built for each subset of the data and features using a specific criteria. The decision tree splits the data into smaller and smaller subsets based on the selected features. In that, each final node represents a class label.

As the last step comes a process of combining just generated decision trees. The predictions of all the decision trees in the forest are combined to make the final prediction. Each decision tree can cast one vote as to what a class label is according to it based on the preceding categorization of data. In a classification problem, the class with the highest number of votes from the individual trees is selected as the final prediction.

Gradient Boosting Classifier

Gradient Boosting Classifier (GBS) is another classification algorithm popular in the field of machine learning. It works by combining multiple weak prediction models to create a strong prediction model. Similarly to the previous method of classification, this one also is based on decision trees, nevertheless it uses them in a different manner. The algorithm starts by creating a simple decision tree that makes predictions based on a single feature. Then, it evaluates how well the tree performs by comparing its predictions to the actual outcomes. Next, it creates a new decision tree that focuses on the samples that were misclassified by the first tree. Following that proceeds a number of iterations where the process is repeated multiple times, with each new tree trying to correct the errors made by the previous ones. What can be also seen on Figure 2.4 is that in the end, all the trees are combined to create a final prediction model that is significantly more accurate than any of the individual trees [36].

The term ”gradient boosting” refers to the use of gradient descent optimization technique to minimize the loss function [37]. The loss function is a measure of how well the prediction model is performing on a set of given input data. During the training process, the algorithm calculates the gradient of the loss function with respect to the predicted values. It then uses this gradient to update the weights of the prediction model, so that it can make better predictions. The gradient descent process is repeated multiple times, with each iteration improving the performance of the prediction model.

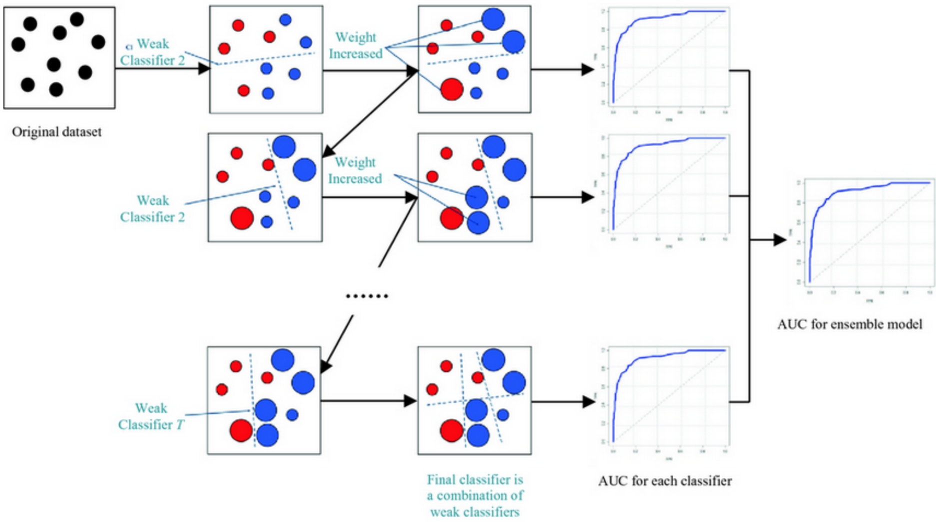


Figure 2.4: Graphic simplified representation of Gradient Boosting algorithm. source: `https://datascience.eu/machine-learning/gradient-boosting-what-you-need-to-know/`

Gradient Boosting Classifier is known for its high accuracy and ability to handle large and complex datasets. However, it can be computationally expensive and requires careful tuning. Overall, gradient boosting is a powerful tool for classification tasks and has become one of the most popular machine learning algorithms in computer science.

Support Vector Classifier

Support Vector Classifier (SVC) is a machine learning algorithm used for classification purposes. It is based on the idea of finding the hyperplane in the feature space that separates the data points into different classes. It achieves the task by finding the maximum margin. The margin is defined as the distance between the hyperplane and the closest data points from each class, as showcased on Figure 2.5. Ultimately, the goal of the SVC algorithm is to find the hyperplane that maximizes this margin while also minimizing the classification error [38].

In SVC, each data point is represented as a vector in a high-dimensional space. The algorithm then finds the hyperplane that separates the data points with the maximum margin. To achieve this, SVC uses a mathematical technique called Lagrange multipliers to optimize a cost function that balances the margin with respect to keeping the classification error as low as possible. The resulting hyperplane is called the maximum margin hyperplane and is used to classify new data points based on which side of the hyperplane they fall.

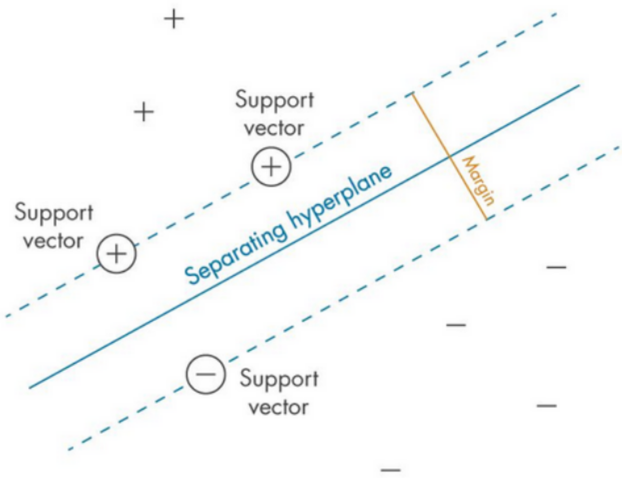


Figure 2.5: Graphic representation of a hyperplane, support vectors and the margin in a Support Vector Machine source: `https://ch.mathworks.com/discovery/support-vector-machine.html`

2.3.5 Regression

Regression is a statistical method used in computer science to determine the relationship between one or more independent variables and a dependent variable. In simpler terms, regression helps in predicting the value of one variable based on the values of other variables. It is a popular technique used in machine learning, data analysis, and other related fields. The aim of regression is to establish a mathematical relationship between the variables so that future values can be predicted with greater accuracy [39].

There are two main types of regression: linear and nonlinear. Linear regression assumes that the relationship between the predictor variables and the outcome variable is linear, which means it can be represented as a straight line. Nonlinear regression models, on the other hand, allow for more complex relationships between the variables, such as curves or other nonlinear shapes.

Regression analysis involves identifying the variables that have an impact on the dependent variable. These variables are known as predictors or independent variables. The other crucial component of a regression analysis involves determining the strength of the relationship between both the independent and the dependent variables.

Once a model is built, it can be used to make predictions about the outcome variable based on the predictor variables. The accuracy of the predictions depends on the quality of the model, which can be evaluated using various metrics such as the mean squared error or the R-squared value, which will be explained in more detail in the following part of this dissertation.

LinearRegression

Linear regression is a statistical modeling technique used to establish a linear relationship between a dependent variable and one or more independent variables. It is one of the simplest and most commonly used techniques for predictive modeling and is widely used in various fields such as finance, economics, social sciences, and engineering. The basic idea of linear regression is to find the best fit for a line (or hyperplane in higher dimensions) to the data points in such a way that the distance between the line and the data points is minimized. The line represents the best linear approximation of the relationship between the dependent and independent variables, as it is graphically presented on Figure 2.6.

In computer science, linear regression is often used as a supervised learning algorithm for predicting numerical values based on input data. For example, it can be used to predict the price of a house based on its size, number of bedrooms, location, etc. The algorithm learns from a training set of input-output pairs and then uses this knowledge to make predictions on new input data. The process of learning involves estimating the parameters of the linear equation that best fits the training data. This is typically done by minimizing the sum of squared errors between the predicted and actual output values using techniques such as gradient descent or the normal equation.

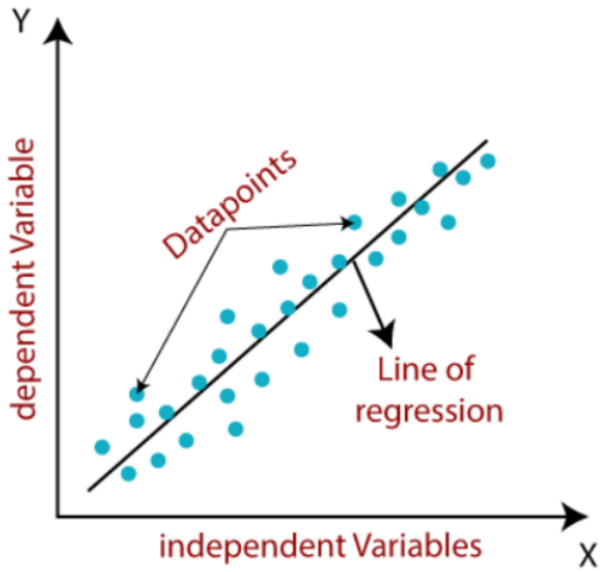


Figure 2.6: Graphic representation of an exemplary Linear regression model. source: `https://www.analyticsvidhya.com/blog/2021/10/everything-you-need-to-know-about-linear-regression/`

One of the strengths of linear regression is its simplicity and interpretability. The model can be easily visualized as a line or hyperplane, and the coefficients of the model provide insights into the relationships between the variables. However, linear regression also has some limitations. It assumes a linear relationship between the variables and may not capture complex nonlinear relationships. It is also sensitive to outliers and can be affected by multicollinearity (high correlation between independent variables). To overcome these limitations, more advanced regression techniques should be utilized.

RandomForestRegressor

The main difference between a Random Forest Regressor and a Random Forest Classifier is the type of problem they are used to solve. A Random Forest Regressor is used for regression tasks, where the goal is to predict a continuous output variable. For example, predicting the price of a house based on its features such as the number of bedrooms, square footage, and location. On the other hand, a Random Forest Classifier is used for classification tasks, where the goal is to predict a categorical output variable. For example, predicting whether an e-mail is spam or not based on its features such as the sender, subject line, and content [40].

While both algorithms use decision trees and ensemble methods to improve the accuracy of their predictions, there are some differences in how they work. In a Random Forest Regressor, the final prediction is the average of the predictions made by each decision tree. In contrast, in a Random Forest Classifier, the final prediction is based on the majority vote of the predictions made by each decision tree.

Another difference is the evaluation metric used to measure the performance of the algorithm. In a Random Forest Regressor, the most commonly used metric is the mean squared error (MSE), which measures the average squared difference between the predicted and actual values. While both Random Forest Regressor and Random Forest Classifier use similar techniques, they are used for different types of problems and have some differences in how they work and how they are evaluated.

GradientBoostingRegressor

Gradient Boosting Regressor (GBR) is another commonly used type of machine learning algorithm that falls under the category of ensemble learning methods. It is a popular technique for regression problems. GBR works by combining multiple weak regression models, typically decision trees, to create a stronger model that can make more accurate predictions. It builds a sequence of trees, where each following layer of trees corrects the errors made by the previous layer of trees. The process continues until a predefined stopping criterion is met, for example, by reaching a predefined number of iterations or achieving a minimum error rate [41].

At each step of the process, the algorithm calculates the gradient of the loss function with respect to the current prediction, which represents how much the prediction needs to change in order to reduce the error. Then, it fits a new tree to the negative gradient of the loss function, effectively minimizing the residual error of the previous tree.

GBR is a powerful and flexible algorithm that can handle various types of data and has been shown to produce high-quality predictions. However, it can be computationally intensive and may require careful tuning of hyperparameters to achieve optimal performance.

Support Vector Regressor (SVR)

There are many similarities between SVR and previously explained SVC. Both of these approaches are based on the operating framework of Support Vector Machine (SVM), which involves finding a hyperplane in a high-dimensional feature space. They both utilize support vectors, which are a subset of training examples that are utilized in order to determine this hyperplane. Both algorithms incorporate the concept of margin and regularization. They aim to find a hyperplane that maximizes the margin or tolerance for errors while also minimizing the model complexity. SVR similarly to SVC can handle nonlinear relationships in the data by using kernel functions. These specific functions allow for implicit mapping of the data into a higher-dimensional space, where a linear hyperplane can be better fitted to the transformed data [42].

The main difference is that SVC aims to find a hyperplane that best separates different classes in a dataset. Its main objective is to classify data points into different discrete categories or classes. On the other hand, SVR focuses on finding a hyperplane that best represents the relationship between input features and real numerical values. It is tailored for regression tasks and to deliver the output in the form of continuous numerical predictions.

Chapter 3

Implementation

3.1 Dataset preparation

Data preparation is a crucial step in conducting an in-depth analysis of energy consumption metering data for event detection and classification. To ensure the dataset's suitability for subsequent analyses, a series of steps were undertaken.

The data provided from an external entity was shared in a form of a series of compressed gzip files. Therefore, the first step to be performed in order to have a legible set of data that could be used for the sake of this dissertation was to decompress all the folders.

In the subsequent step, the raw data obtained from a 24-bit Analog to Digital Converter (ADC) required segmentation into meaningful chunks and transformation into real values representing timestamps, voltage, and current measurements of the mains record. This task was accomplished using a specialized parsing function. The parsing script was fine-tuned specifically for the purpose of this thesis, ensuring optimal extraction of relevant information.

To maintain data integrity and improve its quality, various algorithms were implemented as part of an initial data validity check. These algorithms focused on eliminating outliers and monitoring data continuity. Outliers, if present, were identified and removed to minimize their potential impact on subsequent analyses. Additionally, data continuity was assessed to identify any gaps or irregularities in the data stream, which could affect the accuracy of event detection, classification, and regression algorithms.

Following the initial preprocessing steps, the dataset was converted from its original text format to a more convenient and widely used CSV (Comma-Separated Values) format. This transformation enhanced the dataset's accessibility and facilitated seamless integration of this dataset in the succeeding stages of this research project.

Simultaneously, a separate file containing power consumption recordings per appliance, serving as a reference or 'source of truth' for the experiment, was also parsed into individual files. Each of these files underwent a thorough data continuity check. The focal parameter for data continuity was the timestamp, ensuring a reliable and continuous recording of information saved in the dataset. In cases where missing values were identified, these gaps were filled with zeros to maintain a consistent data structure. Furthermore, any duplicated values were eliminated during this stage.

After performing those steps, the structure of the dataset was as follows: 1 folder containing so called 'raw data' or measurements from the mains, which had a resolution of 7.5 k samples per second. On top of that there were also additional 4 files each having measurements of specific appliances at 1 sample per minute sampling rate. To proceed with machine learning and load disaggregation experiments a subsampling of data related to mains' power consumption had to be performed. This way a voltage or current at one point in time in the 'source of truth' files would be aligned with the measurement in the file containing 'raw data'. To achieve that a special function was written which iteratively computed Root-Mean-Square of instantaneous voltage, current, and power recorded and subsampled it to 1 sample per minute resolution, which would be perfectly correlated with the per appliance recordings.

These preparatory steps aimed to establish a robust and well-organized dataset, laying the foundation for performing classification and regression experiments and analyses. The transformed dataset, along with the verified appliance-level power consumption recordings, formed the basis for evaluating the accuracy and effectiveness of the disaggregation algorithms employed in this research.

3.2 Creating the library with NILMTK toolkit

In order to effectively utilize the functionalities offered by the NILMTK toolkit, it was essential to prepare and organize the dataset in a specific manner. To accomplish this, a dedicated dataset converter was developed. This step was achieved by drawing inspiration from an existing converter created for the iAWE dataset[1].

The key objective of the converter was to align the column names, structure the data appropriately, and subsequently convert it into the HDF5 data format. By adhering to the converter's instructions and layout, the dataset was transformed into a standardized and compatible format suitable for seamless integration with the NILMTK toolkit.

An additional advantage of utilizing the converter from the NILMTK package was its incorporation of data validation checks and the option for voltage and power normalization if required. This ensured the reliability and consistency of the dataset, enhancing the accuracy of subsequent analyses. As a result of these preprocessing steps, a neatly organized dataset was created, with file '1.csv' containing measurements of the mains' energy indicators, and files '[2, 3, 4, 5].csv' containing data on per-appliance power consumption. These individual appliance-level recordings would later serve as the 'source of truth' during the event detection and classification process.

Nevertheless, creating the converter was not a straightforward task and several challenges were encountered, primarily stemming from the limited documentation available for the converters within NILMTK. To overcome this obstacle, a systematic approach was adopted. The first step involved identifying the most similar dataset available within the toolkit that closely resembled the characteristics of the dataset under investigation. This selection allowed for a more informed understanding of the converter's functionalities and facilitated the subsequent adaptation to the specific requirements of the research

[1] available at `https://github.com/nilmtk/nilmtk/tree/master/nilmtk/dataset_converters/iawe`

project.

Upon identifying the closest match, which in this case was the ‘iAWE converter’, a detailed analysis of the converter script was performed. Each function within the converter script was thoroughly examined to comprehend its purpose and functionality. This overall understanding of the existing converter served as the foundation for creating a customized converter tailored to the specific dataset that was provided for this thesis.

The structure of the newly created library, representing the organized dataset, is summarized, and presented in Table 3.1. This table provides an overview of the dataset's organization, highlighting the file names, corresponding measurements, and the role they play in the subsequent stages of the research.

Table 3.1: Organisation of data and naming convention of files used for creating a library based on the NILMTK toolkit.

Equipment name in the Datasheet	Equipment name in ”labels.dat”	Input file name
	mains	1.csv
Fridge + Freezer	fridge	2.csv
Oven + Microwave	electric oven	3.csv
Bimby /Kettle /Air fryer	kettle	4.csv
Stove	stove	5.csv

After employing the NILMTK toolkit and customizing the dataset converter to align with the specific requirements of the research, a well-structured and standardized library was created. This library formed the foundation for further exploration and analysis, enabling the implementation of classification and regression algorithms on the provided mains' energy consumption data and in the end the disaggregation.

3.3 Attempt with NILMTK toolkit

Despite having successfully converted the dataset to meet the required standards of the NILMTK toolkit, attempts at recreating the combinatorial optimization or FHMM (Factorial Hidden Markov Model) algorithms within the toolkit proved unsuccessful. As a consequence, the approach had to be reconsidered and a thorough examination of available documentation and instructions from various sources followed. As a next step, an endeavor was made to replicate an experiment described in [2]. For this attempt it was decided that leaving for the moment the original dataset and utilizing a dataset sourced from within the NILMTK toolkit would be more suited for the task. Regrettably, this trial also concluded with unsatisfactory results.

The lack of success in reproducing the previous methods can be attributed to several factors. Firstly, the challenges faced may stem from the author's limited programming experience, which may have impeded the accurate implementation of the complex algorithms within the NILMTK framework. Moreover, it should be noted that the NILMTK toolkit was developed nearly a decade ago, with the last significant updates or changes to the repository occurring four years ago. This suggests that the toolkit may no longer receive active support or maintenance. Notably, several scripts within NILMTK contain comments indicating that certain sections require further development or improvement, but unfortunately, there appears to be no subsequent follow-up, resulting in unfinished or non-functional code segments.

In light of these challenges and limitations, a decision was made to make a pivot and explore an alternative solution that could still leverage the existing capabilities of the NILMTK toolkit. This shift in approach was motivated by the need to find a more viable path forward in achieving the disaggregation of the provided dataset. By acknowledging the encountered hurdles and the evolving nature of the NILMTK toolkit, this adaptation allowed for a more flexible and practical approach to address the research goals. This decision underscores the adaptability and problem-solving skills required in the field of energy engineering, as researchers must navigate complex software frameworks and make informed choices to progress their research effectively.

The unsuccessful attempts to recreate specific algorithms within NILMTK highlight the importance of continuous development, support, and community engagement in open-source projects. The findings of this exploration serve as valuable insights for future researchers and developers seeking to utilize the NILMTK toolkit or similar tools in the pursuit of energy disaggregation and event detection methodologies.

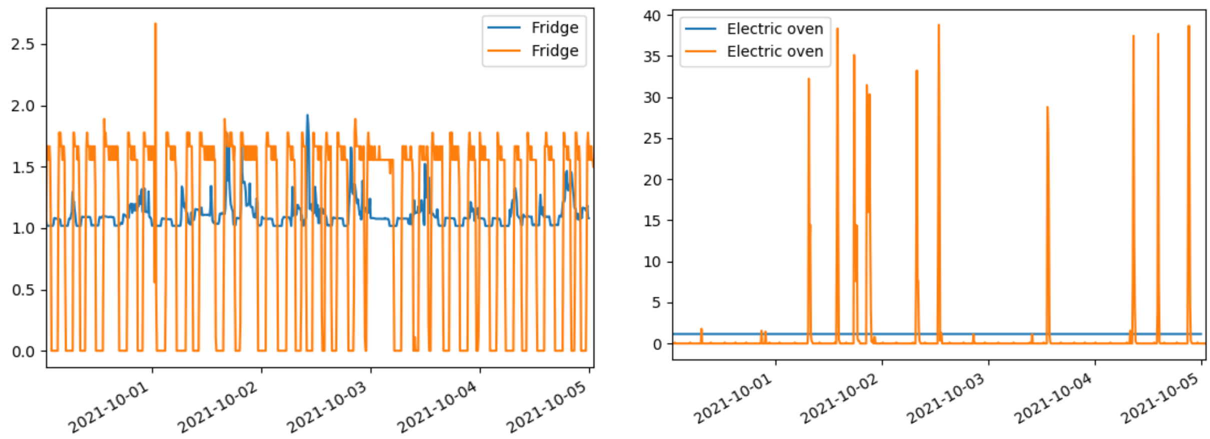
3.4 Attempt with external solution

The next approach that was tried in order to perform a successful disaggregation involved the implementation primarily developed by Odysseas Krystalakos[2], which is based on NILM disaggregation using neural networks. This proposition drew inspiration from the work and architecture presented in [3], providing a promising avenue for further exploration.

The selection of this method was largely driven by the successful recreation of the basic experiment described by the author, in contrast to previous attempts. Encouraged by this outcome, the subsequent step involved replacing the input energy consumption data with the specific dataset provided for this thesis. The possibility for leveraging the dataset that had already been developed using the NILMTK-based converter for the previous experiment was another advantage of this approach. However, despite completing the disaggregation process, the obtained results proved to be unsatisfactory. This observation was evident from the metrics and visual representations depicted on Figure 3.1.

One plausible reason contributing to the failure of this implementation is the highly unbalanced nature of the dataset. For instance, the kettle exhibits only 10 activations over a span of 5 days. Moreover, each period of activity is characterized by short-duration high peaks of power consumption lasting no more than 3 minutes. Besides that, the record of energy usage of this appliance is full of extended periods of zeros or values close to zero. Similar challenges are also observed in other appliances within the dataset. An exception to this pattern is the fridge, whose load profile demonstrates a continuous shape similar to a squared sinusoid. This discrepancy in load profiles poses an obstacle to the successful

[2]https://github.com/OdysseasKr/neural-disaggregator



(a) fridge disaggregation using GRU

(b) electric oven disaggregation using LSTM

Figure 3.1: Visual representation of an unsuccessful disaggregation based on a) Gated Recurrent Unit (GRU) method, and b) Long short-term memory (LSTM) method.

disaggregation process.

The highly unbalanced nature of the dataset, with varying activation patterns and periods of inactivity, presents a significant challenge for accurately disaggregating the energy consumption of individual appliances. The limited occurrence of activations for certain appliances, combined with extended periods of zero consumption, introduces complexity to the disaggregation algorithm's ability to accurately identify and classify energy events.

While the neural network-based disaggregation method initially showed promise in the end it turned out to be unsuccessful, therefore another strategic pivot had to be made and a new pathway had to be discovered in order to conclude this research with a successful disaggregation experiment.

3.5 Disaggregation with scikit-learn algorithms

After extensive effort and experimentation with the previous two approaches which unfortunately ended up unsuccessfully, a decision was made to approach the problem from a more mathematical perspective using scikit-learn for disaggregation and put the implementations based on the NILMTK toolkit on the sidetrack.

With a new wave of resolution and through a thorough review of the literature and examining successful implementations in related fields such as image classification, speech synthesis, and pattern recognition, inspiration was drawn for leveraging scikit-learn in this research. The popularity and well-documented nature of scikit-learn made it an attractive choice, as it provided a wide range of classification and regression methods that could be utilized for energy disaggregation. What's more, scikit-learn offered a comprehensive suite of tools for model evaluation and performance assessment. Various evaluation metrics, including accuracy, precision, recall, F1-score, and mean squared error, were employed to quantitatively measure the performance of the algorithms. These metrics provided valuable insights into the effectiveness of the disaggregation models, allowing for comparisons and informed decision-making.

The metrics that are used in this dissertation are:

$$TP = number\,of\,true\,positives \tag{3.1}$$

$$FP = number\,of\,false\,positives \tag{3.2}$$

$$TN = number\,of\,true\,negatives \tag{3.3}$$

$$FN = number\,of\,false\,negatives \tag{3.4}$$

$$y_i = appliance\,actual\,power \tag{3.5}$$

$$\hat{y}_i = appliance\,predicted\,power \tag{3.6}$$

$$\bar{y} = mean\,of\,predicted\,and\,actual\,values \tag{3.7}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{3.8}$$

$$Precision = \frac{TP}{TP + FP} \tag{3.9}$$

$$Recall = \frac{TP}{TP + FN} \tag{3.10}$$

$$AUC = Area\,Under\,[ROC]\,Curve \tag{3.11}$$

$$mean\,absolute\,error = \frac{1}{n}\sum_{i=1}^{n} |y_i - \hat{y}_i| \tag{3.12}$$

$$mean\,squared\,error = \frac{1}{n}\sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \tag{3.13}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{3.14}$$

Taking advantage of the plethora of instructional materials available on the internet and leveraging the knowledge shared on platforms like Stack Overflow, the algorithms and functions necessary for this research were developed and refined. This collaborative ecosystem and a vast amount of explanatory articles or videos facilitated the successful implementation of the classification and regression methods discussed in this thesis.

As the dataset was already well organized and validated several times the decision was made to benefit from that fact and utilize the same CSV files obtained from the previous attempts as input data. The dataset containing the raw recordings from the mains was divided into five subsets to enable the application of the k-fold cross-validation method. This approach ensured that each subset, representing a day's worth of measurements, served as a testing set, while the remaining four days were used for training the classification or regression algorithms. This process was iterated, with each day being used as a testing set, while the algorithms were trained on the remaining data, ensuring a comprehensive evaluation of the models.

In terms of the results obtained from the approach harnessing the functionalities of the scikit-learn, the generated code organizes the output into four distinct folders: 'electric_oven', 'fridge', 'kettle', and 'stove'. Each folder corresponds to a specific equipment or a group of equipment, as explained in the earlier sections of this thesis. The results are presented both numerically and graphically, providing a comprehensive evaluation of the disaggregation process.

Within each folder, there are four CSV files for classification and four for regression containing the predictions made by the network using varying time window sizes (5, 10, 15, or 20 samples). Additionally, summarizing files are included, which compares the performance of each method used and provides metrics that assess their accuracy and efficiency. In addition to that, the folders contain an array of graphs that visually illustrate the true values of a given device alongside the predicted values generated by the specific method employed. These graphs enable a comprehensive overview and facilitate a direct comparison of the different implementations.

Through the application of scikit-learn and its diverse array of machine learning algorithms, the objective was to attain accurate and reliable energy disaggregation results. By providing both numerical and graphical representations of the disaggregated data, the results section offers a comprehensive analysis of the performance and effectiveness of the employed methods. The variability of time window sizes and the inclusion of four different algorithms for both classification and regression approaches allow for a thorough evaluation of the disaggregation process for non-intrusive load monitoring applications. The results obtained from this part of the research are presented and discussed in detail in the following Chapter 4.

Chapter 4

Results

A full set of numerical and graphical results of the classification and the regression is available at a designated shared folder space in the google drive[1].

4.1 Classification Results

For the classification of the fridge, the results obtained reveal important insights. A comprehensive analysis is presented in Table 4.1, while Figure 4.1 provides a visual representation of the findings. One notable observation is that Logistic Regression exhibits significant deficit in performance when compared to the other classification methods employed in this experiment. This discrepancy highlights the limitations of using a relatively simplistic model like Logistic Regression in scenarios where the dataset contains nonlinear or time series dependencies.

In contrast, the three alternative classification methods employed, namely the Random Forest Classifier, Gradient Boosting Classifier, and SVC, showcase impressive performance metrics. Notably, all three methods surpass the desirable 90% threshold in terms of accuracy, precision, and recall. This achievement implies that the overall performance of these models is highly satisfactory for the task. Furthermore, their robustness highlights the fact that they are able to effectively discern patterns and capture dependencies within more complicated datasets like the energy consumption data incorporated in this research.

It is crucial to emphasize that all models achieved an average recall rate exceeding 0.98. This statistic implies that nearly all activation periods of the fridge were accurately detected and correctly assigned as an "ON" state by the classification algorithms. This level of performance further reinforces the reliability and effectiveness of the employed models in accurately identifying and characterizing the operational status of the fridge.

When comparing the classification results of the electric oven to the previous appliance, the fridge, it becomes evident that there are distinctive differences. Speaking more precisely, these dissimilarities are observed in terms of the number of non-zero samples present in the dataset. While the energy

[1] https://drive.google.com/drive/folders/1yhE7LKSrit9tUHhyXU52TMR6T6sFJkFF?usp=sharing

Table 4.1: Comparison of key classification performance metrics for the fridge.

	$time\_window = 5$				$time\_window = 20$			
	AUC	$accuracy$	$precision$	$recall$	AUC	$accuracy$	$precision$	$recall$
$LogisticRegression$	0.87	0.60	0.60	1.00	0.88	0.59	0.59	1.00
$RandomForestClassifier$	0.95	0.91	0.91	0.95	0.95	0.93	0.91	0.97
$GradientBoostingClassifier$	0.94	0.94	0.91	0.99	0.93	0.94	0.91	0.99
SVC	0.88	0.93	0.90	1.00	0.89	0.93	0.90	0.99

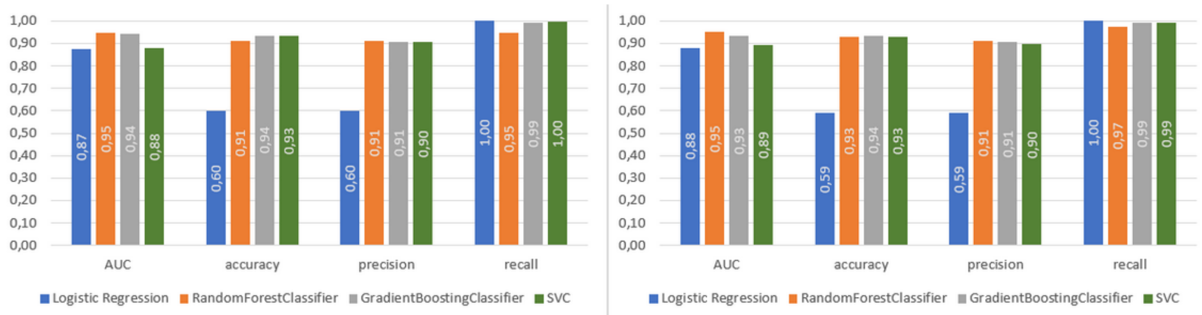


Figure 4.1: Classification performance of the fridge's energy consumption for time_window = 5 (left) and time_window = 20 (right).

consumption data of the fridge encompasses a substantial 4,354 values that correspond to the activation period (when the energy consumption is above 0), constituting approximately 60.52% of all samples, the electric oven data, on the other hand, comprises 10.61 times fewer values within the activation period. In fact, the electric oven data only consists of 410 non-zero values, which represents 5.70% share of the entire dataset.

The implications of this discrepancy are clearly reflected in the analysis presented on Figure 4.2. The figure provides a visual representation of the observed phenomenon, illustrating a significant decline in recall values across all classification methods. This decline serves to highlight the underlying issue at hand, which is the inability of all the models to accurately detect and locate the ON states of the electric oven. The significance of this predicament is further emphasized by the data presented in Table 4.2, where the recall performance has significantly dropped, reaching levels not exceeding 0.45.

These findings shed light on the challenges associated with accurately classifying the electric oven's energy consumption patterns. The scarcity of non-zero values poses a unique obstacle for the classification models, resulting in a decreased ability to precisely identify the ON states of this particular appliance.

In the context of the classification of the kettle and the stove, a significant challenge arises due to the scarcity of measurements during their activation periods. When compared to the previously mentioned appliance, the electric oven, this scarcity issue is further exacerbated for both the kettle and the stove. Specifically, the share of non-zero data points in relation to the entire dataset is found to be 1.46% (105 samples) for the kettle and 0.87% (63 samples) for the stove. Given that the obstacle of non-zero value scarcity is applicable to both appliances, it is feasible to discuss them simultaneously.

Table 4.2: Comparison of key classification performance metrics for the electric oven.

	$time\_window = 5$				$time\_window = 20$			
	AUC	$accuracy$	$precision$	$recall$	AUC	$accuracy$	$precision$	$recall$
$LogisticRegression$	0.89	0.89	0.92	0.23	0.90	0.90	0.91	0.33
$RandomForestClassifier$	0.85	0.91	0.87	0.36	0.90	0.89	0.85	0.27
$GradientBoostingClassifier$	0.86	0.91	0.83	0.44	0.90	0.91	0.85	0.42
SVC	0.79	0.88	0.96	0.13	0.88	0.88	0.96	0.12

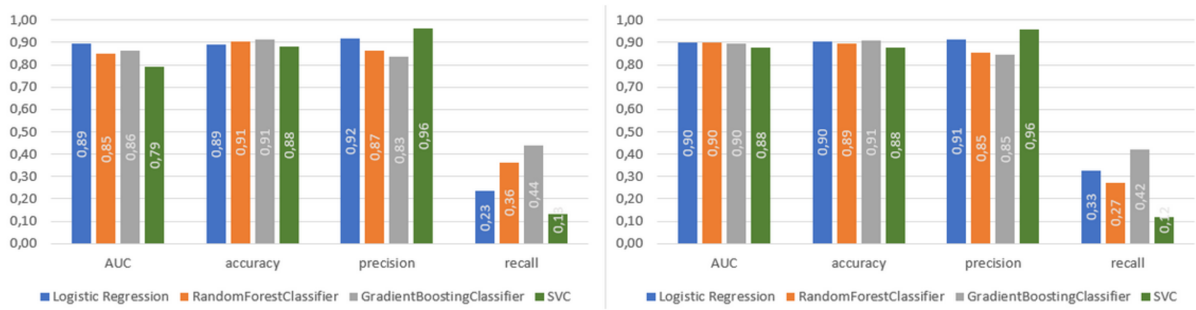


Figure 4.2: Classification performance of the electric_oven's energy consumption for time_window = 5 (left) and time_window = 20 (right).

In order to visually illustrate the implications of this dataset imbalance, Figure 4.3 and Figure 4.4 provide explicit representations of the precision and recall values for both the kettle and the stove, revealing that the values of performance are either 0 or very close to 0. However, upon examining Table 4.3 and Table 4.4, it becomes evident that the AUC and accuracy metrics achieve remarkable values of up to 100%. This outcome arises from the fact that the model constantly predicts the ”OFF” state, resulting in only 105 misclassifications for the kettle and 63 misclassifications for the stove. These numbers are negligible considering the total number of predictions made for each appliance, which amounts to almost 7,200 in total.

Addressing this scarcity issue requires alternative approaches. One potential solution would be to extend the duration of recording energy consumption data beyond the current five-day period. By increasing the length of the data collection phase, more measurements during the activation periods of the kettle and the stove can be obtained, thus mitigating the scarcity problem. Alternatively, gathering data from multiple households using different kettles. Another idea would be to conduct more frequent activations of these appliances within a household during the data acquisition period and as a result obtaining a larger dataset. However, it is worth noting that the last proposal contradicts the fundamental principle of non-intrusive load monitoring, which aims to minimize the intrusion and disruption to the people living inside the monitored households.

Table 4.3: Comparison of key classification performance metrics for the kettle.

	$time\_window = 5$				$time\_window = 20$			
	AUC	$accuracy$	$precision$	$recall$	AUC	$accuracy$	$precision$	$recall$
$LogisticRegression$	0.91	0.98	0.00	0.00	0.87	0.98	0.00	0.00
$RandomForestClassifier$	0.93	0.96	0.00	0.00	0.95	0.98	0.00	0.00
$GradientBoostingClassifier$	0.93	0.96	0.12	0.19	0.94	0.97	0.00	0.00
SVC	0.60	0.97	0.16	0.19	0.84	0.98	0.00	0.00

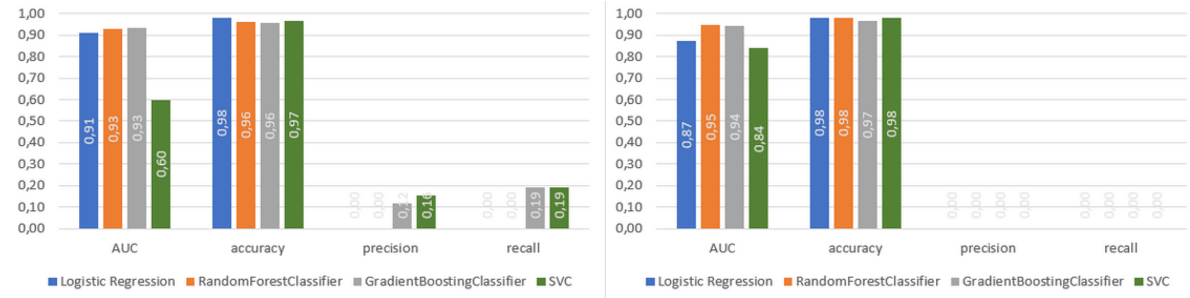


Figure 4.3: Classification performance of the kettle's energy consumption for time_window = 5 (left) and time_window = 20 (right).

Table 4.4: Comparison of key classification performance metrics for the stove.

	$time\_window = 5$				$time\_window = 20$			
	AUC	$accuracy$	$precision$	$recall$	AUC	$accuracy$	$precision$	$recall$
$LogisticRegression$	1.00	1.00	0.00	0.00	1.00	1.00	0.00	0.00
$RandomForestClassifier$	1.00	1.00	0.00	0.00	1.00	1.00	0.00	0.00
$GradientBoostingClassifier$	1.00	1.00	0.00	0.00	1.00	1.00	0.00	0.00
SVC	1.00	1.00	0.00	0.00	1.00	1.00	0.00	0.00

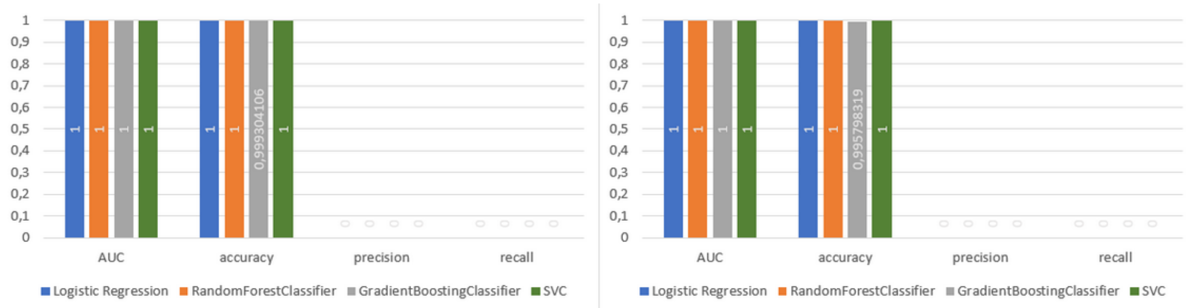


Figure 4.4: Classification performance of the stove's energy consumption for time_window = 5 (left) and time_window = 20 (right).

4.2 Regression Results

Moving on to the regression experiments in the context of load disaggregation, the first in line to discuss the performance of the regression algorithms stands the fridge. In this subsection of the thesis a correlation between the effectiveness of classification methods made previously and the performance of the following regression algorithms is observed. Certain similarities stand out more than others, such is in the case of the poor performance of the Linear Regression model, which operates using comparable mechanisms to the Logistic Regression model.

Figure 4.5, specifically panel (a), illustrates the predictions made by the Linear Regression model. As depicted, the model's predictions closely resemble a straight, continuous line with a value of 1. This indicates that the method is essentially making random guesses between the minimum and maximum values for the fridge, which are 0 and 2, respectively. This outcome is reminiscent of the previous unsuccessful attempt described in Chapter 3.4 in which GRU and LSTM methods were incorporated.

In contrast to the unsatisfactory performance of the Linear Regression model, the prediction results obtained from three alternative regression methodologies—Random Forest Regressor, Gradient Boosting Regressor, and SVR—prove to be more promising. The corresponding charts can be observed in Figure 4.5, specifically points (b), (c), and (d), respectively.

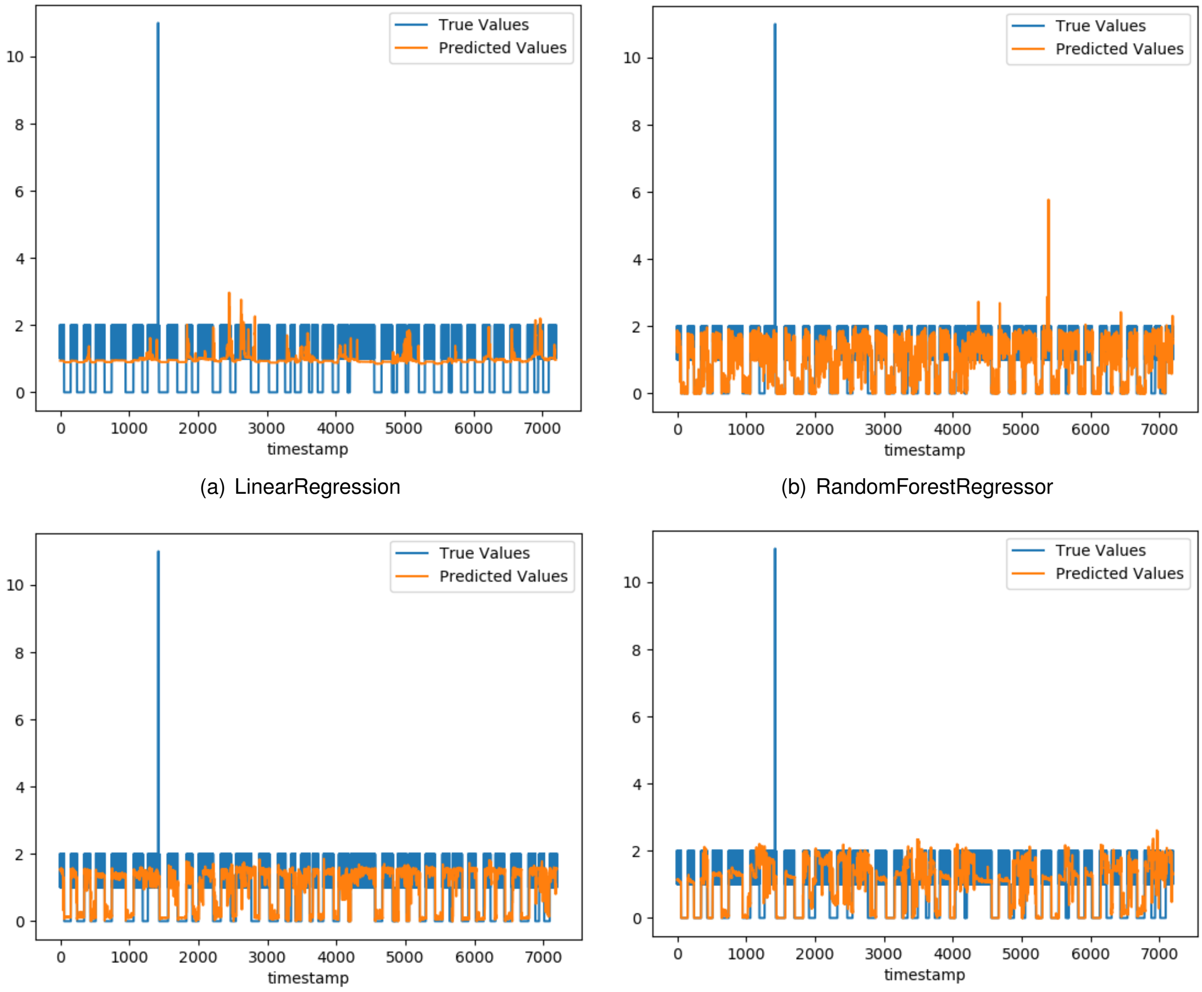
To further reinforce the observations made from the figures, numerical values are provided in Table 4.5. These values confirm that the Random Forest Regressor and Gradient Boosting Regressor outperform the other methods, with MSE values of 0.380 and 0.417, respectively. Although SVR performs slightly worse than the previous two methods, it still exhibits a reasonably acceptable level of effectiveness.

It is worth noting that the three latter methods excel at identifying and predicting 0 values, as well as detecting the transition moments when the energy consumption of the fridge shifts from zero to a non-zero level. However, they struggle to accurately estimate the power consumption values when the fridge is in the ON state. This difficulty arises from the fact that during this period, the true values oscillate between 1 and 2 at a relatively random rate, characterized by unpredictable dynamic changes.

Table 4.5: Comparison of key regression performance metrics for the fridge.

	MAE	MSE	R^2
$LinearRegression$	0.779	0.813	0.047
$RandomForestRegressor$	0.465	0.417	0.511
$GradientBoostingRegressor$	0.459	0.380	0.555
SVR	0.463	0.488	0.429

In the context of regression results for the electric oven, kettle, and stove, it is evident that although each of these appliances are different, their load consumption patterns at a resolution incorporated in this research exhibit many similarities. Figure 4.6 provides a clear visualization of the electric oven's energy usage, which differs significantly from the previous equipment studied - the fridge. In the cases of the kettle and stove, their energy consumption is predominantly equal to zero, with only occasional instances of usage. By comparing Figure 4.6, Figure 4.7, and Figure 4.8, one can observe the similarities in energy consumption patterns among these three appliances, as they frequently exhibit a continuous, straight line at a level of zero. Occasionally, small peaks in energy consumption are observed in the true values data visualization. Consequently, these three appliances can be discussed collectively as a



(a) LinearRegression

(b) RandomForestRegressor

(c) GradientBoostingRegressor

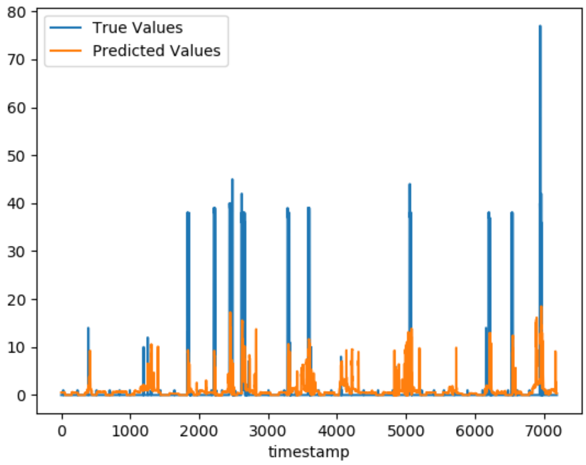
(d) SVR

Figure 4.5: Visual representation of disaggregation of energy consumption of the fridge based on various regression methodologies: a), b), c), and d).

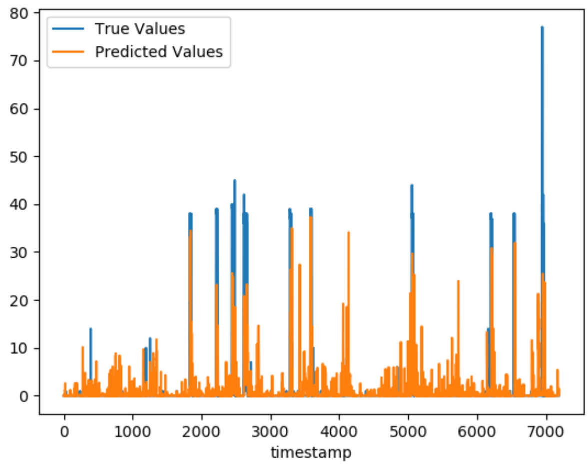
single group.

The scarcity of non-zero values and the highly unbalanced nature of the dataset have already been addressed in the previous Chapter 4.1. Therefore, there is no need to reiterate these issues. However, it should be mentioned that these challenges were overarching problems encountered throughout the research conducted for this dissertation, as well as in the field of non-intrusive load monitoring as a whole.

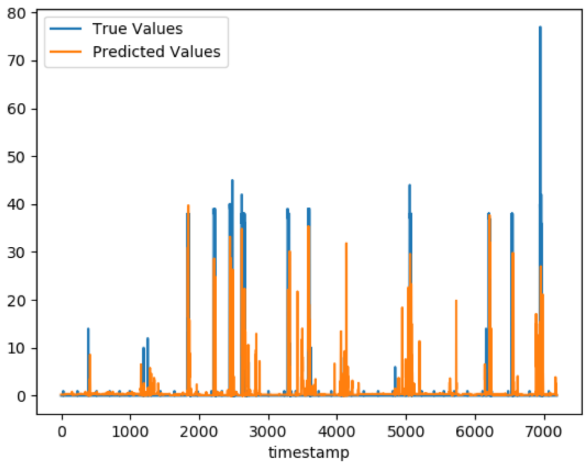
Remarkably, despite the difficulties posed by the challenging dataset, both the Random Forest Regressor and the Gradient Boosting Regressor demonstrate relatively positive performance. This can be observed most prominently when analyzing Figure 4.6, precisely points b) and c). The graphs illustrate that many peak values are accurately recognized, and the predicted values align well with the true values in terms of timing, occasionally even approaching the correct magnitude of the peaks. Of particular interest is the notable superiority of the Gradient Boosting Regressor over the Random Forest Regressor, as evidenced in Table 4.6. While both methods exhibit almost identical scores for MAE (0.435 vs 0.438, respectively), the MSE metric highlights the clear advantage (0.700 vs 1.908, respectively) of the Gradient Boosting Regressor. These findings are further supported by the graphs, particularly in Figure 4.6, points b) and c), and to some degree in Figure 4.7, points b) and c). This outcome was expected due to the principal of operation of the just mentioned regression methodology, as it is described in Chapter 2.3.5.



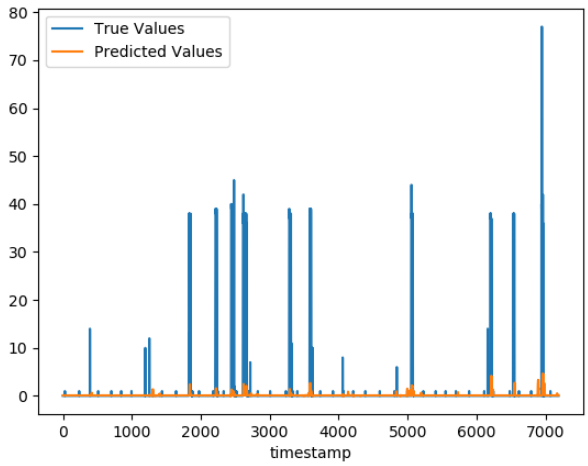
(a) LinearRegression



(b) RandomForestRegressor



(c) GradientBoostingRegressor



(d) SVR

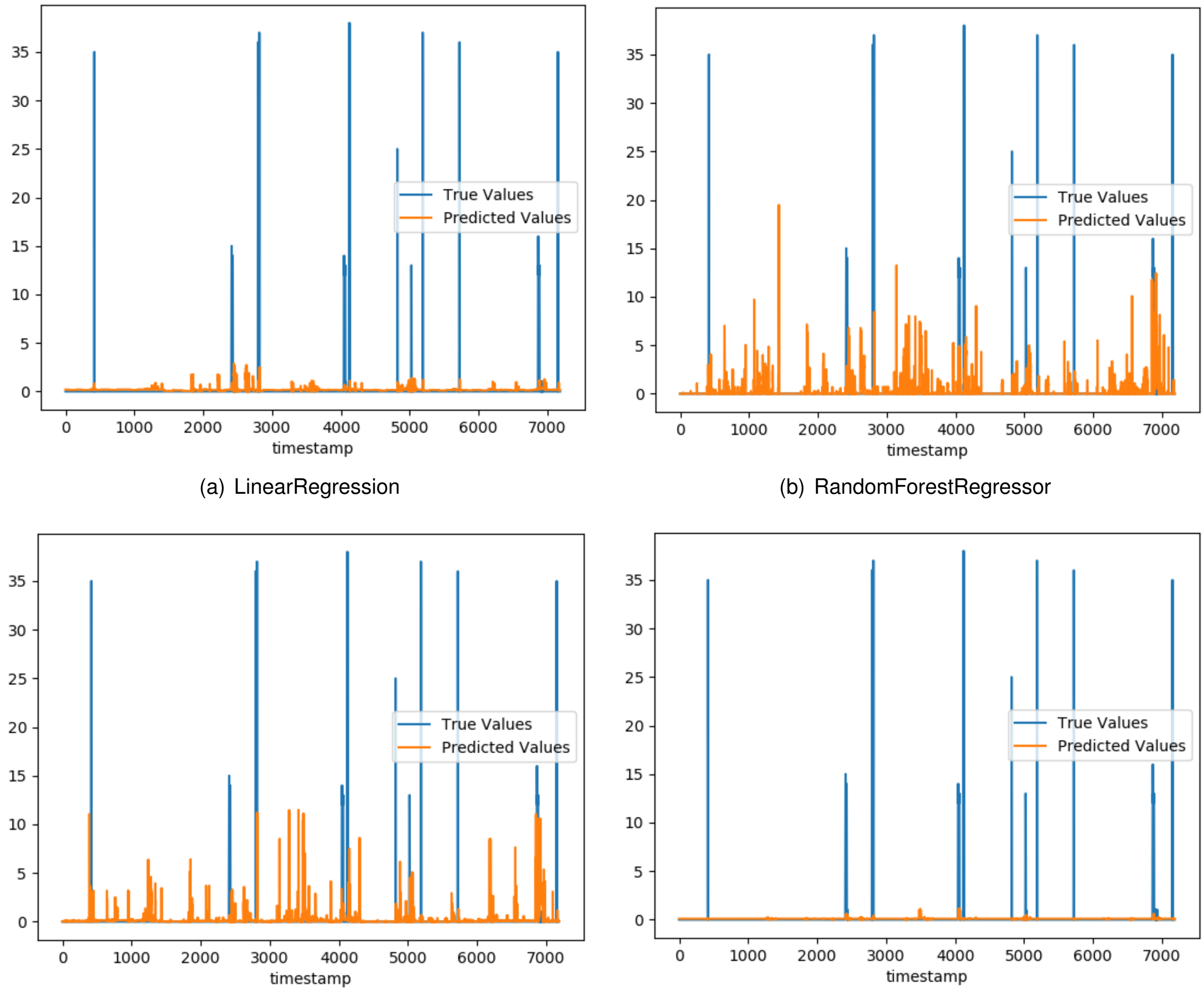
Figure 4.6: Visual representation of disaggregation of energy consumption of the electric oven based on various regression methodologies: a), b), c), and d).

Table 4.6: Comparison of key regression performance metrics for the electric oven.

	MAE	MSE	R^2
$LinearRegression$	0.700	2.049	-5.307
$RandomForestRegressor$	0.438	1.908	-4.874
$GradientBoostingRegressor$	0.435	0.700	-1.153
SVR	0.136	0.333	-0.024

It is important to note that the R^2 score for all three appliances is either close to zero or even negative, indicating the absence of a linear relationship between the independent and dependent variables. This outcome can be attributed to the discrepancy between the predicted values and the true values, highlighting the regression methods' limited ability to accurately capture underlying relationships and make precise predictions.

Furthermore, an interesting observation can be made regarding the SVR. Although, at first glance, it may appear as a continuous, straight line in close proximity to zero when examining points d) in Figure 4.6, Figure 4.7, or Figure 4.8, the metrics presented in Table 4.6 and Table 4.7 indicate that SVR exhibits the lowest MAE and MSE among all the methods employed in these regression experiments.



(a) LinearRegression

(b) RandomForestRegressor

(c) GradientBoostingRegressor

(d) SVR

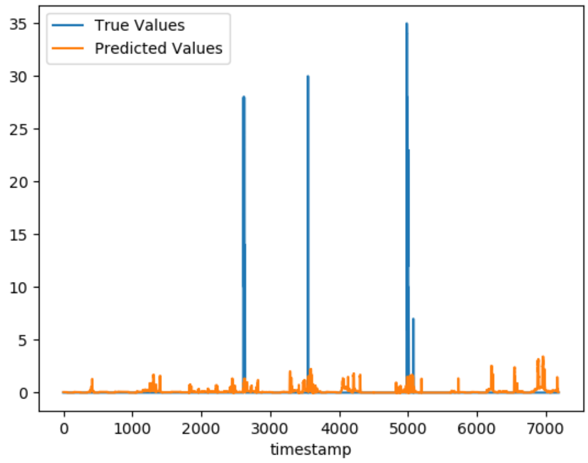
Figure 4.7: Visual representation of disaggregation of energy consumption of the kettle based on various regression methodologies: a), b), c), and d).

Table 4.7: Comparison of key regression performance metrics for the kettle.

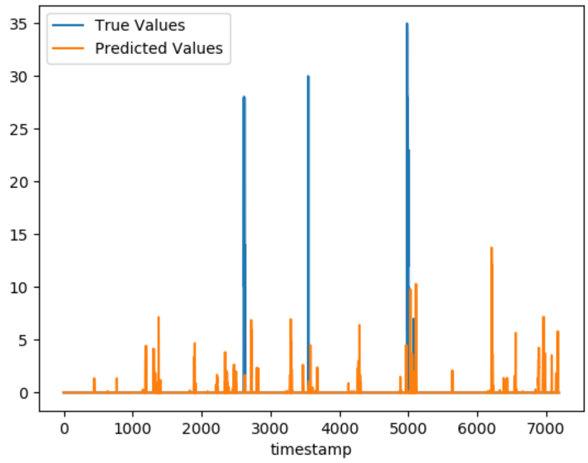
	MAE	MSE	R^2
$LinearRegression$	0.239	1.050	0.013
$RandomForestRegressor$	0.225	1.812	-0.704
$GradientBoostingRegressor$	0.254	1.437	-0.351
SVR	0.139	1.073	-0.009

Overall, these findings provide valuable insights into the regression results for the fridge, electric oven, kettle, and stove, providing a plethora of valuable insides on the performance of various regression methods and highlighting the complexities and challenges associated with accurately predicting energy consumption patterns for these appliances.

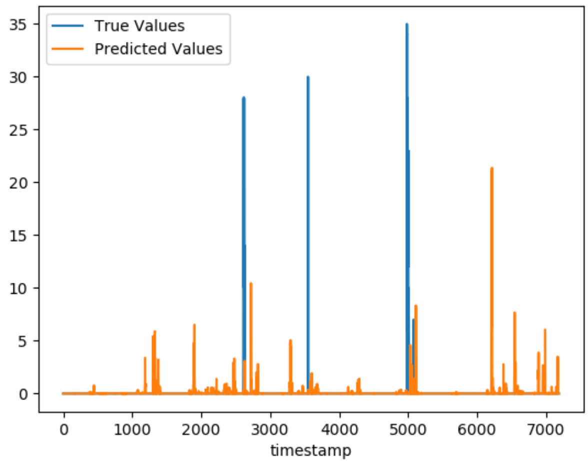
Taking into consideration all of the above performed and presented experiments, it can be seen that there exists a clear superiority of Random Forest Regressor and Gradient Boosting Regressor for accurately predicting energy consumption patterns of all of the appliances analyzed in this research. The findings demonstrate the potential of these methodologies to contribute significantly to load disaggregation techniques and improve energy management strategies by providing precise information about appliance usage.



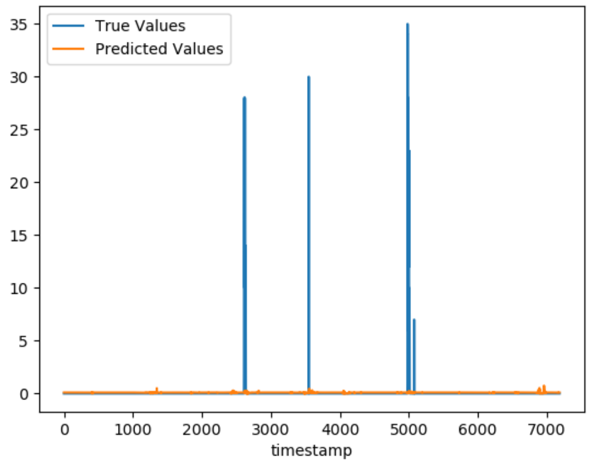
(a) LinearRegression



(b) RandomForestRegressor



(c) GradientBoostingRegressor



(d) SVR

Figure 4.8: Visual representation of disaggregation of energy consumption of the stove based on various regression methodologies: a), b), c), and d).

Table 4.8: Comparison of key regression performance metrics for the stove.

	MAE	MSE	R^2
$LinearRegression$	0.088	0.043	0.000
$RandomForestRegressor$	0.048	0.111	0.000
$GradientBoostingRegressor$	0.051	0.101	0.000
SVR	0.096	0.009	0.000

Chapter 5

Conclusions

The final chapter of this dissertation summarizes the key findings and accomplishments of the research on event detection and classification for disaggregation of energy consumption metering data, as well as proposes multiple pathways that could be taken as a further development of this topic. Throughout this master's thesis, significant efforts were dedicated to preprocessing the data, implementing various classification or regression machine learning algorithms, and evaluating their performance of disaggregation capabilities on different household devices. This chapter aims to provide an encapsulating overview of the processes undertaken, challenges encountered, and the noteworthy achievements attained in this study. By analyzing the outcomes and reflecting on the acquired knowledge and skills, valuable insights are gained into the effectiveness of the implemented methodologies.

5.1 Achievements

The first of the achievements obtained on the course of development of this thesis, which was also first in the chronological order is conducting the successful preprocessing and data quality improvement. This included unpacking and decoding the dataset, as well as addressing issues such as outliers and data continuity. The dataset was prepared to meet the standards required for further analysis.

Another aspect that was a considerable obstacle but at the same time it served as a potent reason to exert efforts even stronger was to keep progressing with limited information. Despite encountering limitations in terms of available information about the dataset or clarifications to some parts of it the work was not halted and the progress continued. Complementary information was sought to enhance the understanding and analysis of the data.

Next thing that could be counted as the achievement gained during the process was the implementation of NILM metadata that played a crucial role in converting the dataset into a specific type of a library in the HDF5 format. This process involved aligning the data according to the metadata specifications and instructions inside the NILMTK toolkit. At the end this step facilitated to a great extent a subsequent testing and experimentation with different disaggregation approaches.

Following that the successful disaggregation was performed despite limited power consumption measurements. Regardless of the scantness of the data recording and not so substantial amount of power measurements available, successful disaggregation methods were performed. This poses as an argument in favor of the effectiveness of the implemented approaches. Implementation and evaluation of classification and regression methods was conducted. It was a significant achievement as it compared various artificial intelligence and provided a comprehensive analysis allowing for the identification of suitable machine learning algorithms for non-intrusive load monitoring and disaggregation. Additionally, important parameters such as the size of the time window were evaluated to determine their impact on performance.

Last but not least, it is important to mention an aspect of personal growth and learning, which can also be accounted as the achievement of this dissertation. Throughout the research process, significant personal growth was achieved in various domains, including programming, machine learning, operating on big data, and applying artificial intelligence methods to address the real world problems. The acquisition of knowledge and skills in these areas represents a substantial personal accomplishment.

Overall, the research efforts in event detection and classification for disaggregation of energy consumption metering data have yielded successful preprocessing techniques, effective disaggregation methods, and valuable insights into the performance and suitability of various machine learning algorithms. The personal growth and learning experiences gained throughout the research further contribute to the significance of these experiments.

5.2 Future Work

In the topic of future work for event detection and classification in disaggregation of energy consumption metering data, there are several potential avenues that subjectively speaking would be interesting and worth exploring or possibly improving the threads that have already been opened in the experiments developed in this dissertation.

One important aspect that could be improved is related to the general size of the dataset. Recording, collecting, and compiling larger amounts of data from households available to the companies or other institutions having access to energy consumption information would bring benefits to the overall implementation of NILM. By working with a richer dataset, it would be possible to train the applied algorithms on a broader range of scenarios, leading to potentially more accurate disaggregation results, at least in theory. Gathering data from diverse households, including those located outside Portugal, could also provide insights into the universality and robustness of the developed techniques. Examining this pathway could also provide a sense of how feasible this solution would be for mass deployment and whether introducing it to a broader market would already be within the reach of private companies.

Another plausible area of future research would be to conduct experiments on multiple and different households, using one house for testing the neural networks and the remaining ones for training the the disaggregation algorithms. However, it should be noted that this approach has shown some challenges that are documented in the literature. One of such obstacles refers to the issue of similarity of load profiles among different appliances. For example a load profile of the computer could be so similar to the one generated by the air conditioning unit that the disaggregation algorithm yields results no better than a random guess. On the other hand, multiple research papers point out that there are also troubles related to the variability in load profiles even for the same appliances but with slightly different models or manufacturers. Further investigations and experimentation are needed to address these complexities and evaluate the performance of the algorithms in different household contexts.

Additionally, as technology advances, it is worth considering the potential impact of a new version of NILM toolkits, such as an API or web applications. Investigating whether these newly developed solutions offer enhanced capabilities and improved performance compared to the original version of the NILMTK repository used in this research. Related to that would be another possible pathway that could be taken in the future which would be to focus on lowering the barrier of entry for implementing NILM techniques. Simplifying the deployment of these techniques could empower ordinary homeowners to easily monitor their energy consumption and make informed decisions such as aligning the energy consumption to the off-peak hours in order to promote sustainability. By providing user-friendly tools and interfaces, more people would have the opportunity to track their energy usage, identify potential inefficiencies, and adopt more sustainable habits.

A collection of solutions proposed in this dissertation as well as presented pathways for future research demonstrate significant potential for advancing the field of non-intrusive load monitoring and energy disaggregation from metering data. Moreover, these findings offer promising inputs that combined with other research could serve as a potent way of addressing the pressing global issue of global warming, as they promote energy efficiency and empower individuals to actively engage in the development of sustainable energy practices. By collectively working towards this common goal, we can strive to solve the overarching challenge and contribute to a more sustainable future.

Bibliography

[1] G. W. Hart. Nonintrusive appliance load monitoring. *Proceedings of the IEEE*, 80(12):1870–1891, 1992.

[2] N. Batra, J. Kelly, O. Parson, H. Dutta, W. Knottenbelt, A. Rogers, A. Singh, and M. Srivastava. Nilmtk: An open source toolkit for non-intrusive load monitoring. In *Proceedings of the 5th international conference on Future energy systems*, pages 265–276, 2014.

[3] J. Kelly and W. Knottenbelt. Neural nilm: Deep neural networks applied to energy disaggregation. In *Proceedings of the 2nd ACM international conference on embedded systems for energy-efficient built environments*, pages 55–64, 2015.

[4] G. W. Hart. Prototype nonintrusive appliance load monitor. In *MIT Energy Laboratory Technical Report, and Electric Power Research Institute Technical Report*. 1985.

[5] N. Sadeghianpourhamami, J. Ruyssinck, D. Deschrijver, T. Dhaene, and C. Develder. Comprehensive feature selection for appliance classification in nilm. *Energy and Buildings*, 151:98–106, 2017.

[6] H. Y. Lam, G. Fung, and W. Lee. A novel method to construct taxonomy electrical appliances based on load signaturesof. *IEEE Transactions on Consumer Electronics*, 53(2):653–660, 2007.

[7] J. Alcalá, J. Ureña, Á. Hernández, and D. Gualda. Event-based energy disaggregation algorithm for activity monitoring from a single-point sensor. *IEEE Transactions on Instrumentation and Measurement*, 66(10):2615–2626, 2017.

[8] D. Yang, X. Gao, L. Kong, Y. Pang, and B. Zhou. An event-driven convolutional neural architecture for non-intrusive load monitoring of residential appliance. *IEEE Transactions on Consumer Electronics*, 66(2):173–182, 2020.

[9] A. Zoha, Q. H. Abbasi, and M. A. Imran. A non-event based approach for non-intrusive load monitoring. *Wireless Automation as an Enabler for the Next Industrial Revolution*, pages 173–191, 2020.

[10] Z. Zhou, Y. Xiang, H. Xu, Y. Wang, and D. Shi. Unsupervised learning for non-intrusive load monitoring in smart grid based on spiking deep neural network. *Journal of Modern Power Systems and Clean Energy*, 10(3):606–616, 2021.

[11] S. Dutta, V. Kumar, A. Shukla, N. R. Mohapatra, and U. Ganguly. Leaky integrate and fire neuron by charge-discharge dynamics in floating-body mosfet. *Scientific reports*, 7(1):8257, 2017.

[12] R. Gopinath, M. Kumar, C. P. C. Joshua, and K. Srinivas. Energy management using non-intrusive load monitoring techniques–state-of-the-art and future research directions. *Sustainable Cities and Society*, 62:102411, 2020.

[13] J. Morley and M. Hazas. The significance of difference: Understanding variation in household energy consumption. eceee, 2011.

[14] J. Lehne and F. Preston. Making concrete change: Innovation in low-carbon cement and concrete. 2018.

[15] Buildings. `https://www.iea.org/reports/buildings`, 2022.

[16] A. L. Davis, T. Krishnamurti, B. Fischhoff, and W. B. de Bruin. Setting a standard for electricity pilot studies. *Energy Policy*, 62:401–409, 2013.

[17] K. C. Armel, A. Gupta, G. Shrimali, and A. Albert. Is disaggregation the holy grail of energy efficiency? the case of electricity. *Energy policy*, 52:213–234, 2013.

[18] W. Janssen. Energy disaggregation: Nonintrusive load monitoring: The search for practical methods. 2018.

[19] J. Z. Kolter and M. J. Johnson. Redd: A public data set for energy disaggregation research. In *Workshop on data mining applications in sustainability (SIGKDD), San Diego, CA*, volume 25, pages 59–62. Citeseer, 2011.

[20] A. Filip et al. Blued: A fully labeled public dataset for event-based nonintrusive load monitoring research. In *2nd workshop on data mining applications in sustainability (SustKDD)*, volume 2012, 2011.

[21] J. Kelly and W. Knottenbelt. The uk-dale dataset, domestic appliance-level electricity demand and whole-house demand from five uk homes. *Scientific data*, 2(1):1–14, 2015.

[22] J. Gao, S. Giri, E. C. Kara, and M. Bergés. Plaid: a public dataset of high-resoultion electrical appliance measurements for load identification research: demo abstract. In *proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings*, pages 198–199, 2014.

[23] 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*, pages 1–8, 2013.

[24] A. S. 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*, pages 45–54, 2015.

[25] C. Klemenjak, A. Reinhardt, L. Pereira, S. Makonin, M. Bergés, and W. Elmenreich. Electricity consumption data sets: Pitfalls and opportunities. In *Proceedings of the 6th ACM international conference on systems for energy-efficient buildings, cities, and transportation*, pages 159–162, 2019.

[26] L. Pereira and N. Nunes. Performance evaluation in non-intrusive load monitoring: Datasets, metrics, and tools—a review. *Wiley Interdisciplinary Reviews: data mining and knowledge discovery*, 8 (6):e1265, 2018.

[27] C. H. Papadimitriou and K. Steiglitz. *Combinatorial optimization: algorithms and complexity*. Courier Corporation, 1998.

[28] B. H. Korte, J. Vygen, B. Korte, and J. Vygen. *Combinatorial optimization*, volume 1. Springer, 2011.

[29] L. Rabiner and B. Juang. An introduction to hidden markov models. *ieee assp magazine*, 3(1): 4–16, 1986.

[30] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011.

[31] scikit-learn user guide. `https://scikit-learn.org/stable/user_guide.html`.

[32] S. Raschka and V. Mirjalili. *Python machine learning: Machine learning and deep learning with Python, scikit-learn, and TensorFlow 2*. Packt Publishing Ltd, 2019.

[33] ”a beginner’s guide to binary classification”. `https://towardsdatascience.com/beginner/home`.

[34] A. C. Müller and S. Guido. *Introduction to machine learning with Python: a guide for data scientists*. ” O’Reilly Media, Inc.”, 2016.

[35] A. Cutler, D. R. Cutler, and J. R. Stevens. Random forests. *Ensemble machine learning: Methods and applications*, pages 157–175, 2012.

[36] T. Hastie, R. Tibshirani, J. H. Friedman, and J. H. Friedman. *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer, 2009.

[37] S. Ruder. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*, 2016.

[38] A. Géron. *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow*. ” O’Reilly Media, Inc.”, 2022.

[39] S. Weisberg. *Applied linear regression*, volume 528. John Wiley & Sons, 2005.

[40] M. R. Segal. Machine learning benchmarks and random forest regression. 2004.

[41] A. Natekin and A. Knoll. Gradient boosting machines, a tutorial. *Frontiers in neurorobotics*, 7:21, 2013.

[42] S. Suthaharan and S. Suthaharan. Support vector machine. *Machine learning models and algorithms for big data classification: thinking with examples for effective learning*, pages 207–235, 2016.