TÉCNICO LISBOA

Fuzzy Modelling for the Detection of Non-Technical Losses Using Time Variant and Invariant Features

Diogo Fermisson Caridade

Thesis to obtain the Master of Science Degree in

Mechanical Engineering

Supervisors: Prof. Susana Margarida da Silva Vieira

Eng. Joaquim Paul Laurens Viegas

Examination Committee

Chairperson: Prof. Paulo Jorge Coelho Ramalho Oliveira

Supervisor: Prof. Susana Margarida da Silva Vieira

Member of the Committee: Prof. Carlos Augusto Santos Silva

November 2018

Acknowledgements

I would like to express my sincere gratitude to those who in one way or another have supported me and contributed to the completion of this thesis.

I would first like to thank professor Susana Vieira for introducing me to machine learning and making it accessible and enticing to someone searching for a career path to follow. A very special thanks to engineer Joaquim Viegas without whom I would not have been able to come this far. His remarkable patience, guidance and availability gave me the confidence and knowledge to overcome every obstacle in my way.

I am also deeply thankful to my family for their unwavering support and patience not only during the course of this project, but throughout my entire academic journey.

Furthermore, I would like to thank my friends Rafael Santos and Rita Tomaz for the relaxing lunches and stimulating discussions. Thanks also to Melanie Rodrigues for sharing with me some of her passionate spirit and priceless stories.

Finally, a very warm and meaningful thank you to Joana Dias for sticking with me through thick and thin and for providing me with much needed distraction, comfort and inspiring words in the most stressful moments.

Resumo

Esta tese aborda a problemática das perdas não técnicas na indústria da produção de electricidade com o foco no consumo ilegal de electricidade por parte de certos consumidores. O objectivo é criar modelos de classificação usando modelação fuzzy para a detecção à base de informação de perdas não-técnicas. Mais especificamente, um novo processo de agrupamento, Mixed Fuzzy Clustering (MFC), é utilizado para potenciar tanto os atributos variáveis como os não variáveis no tempo na identificação de consumidores ilegais. De forma a avaliar o desempenho dos modelos criados, três casos de estudo são usados considerando diferentes tipos de variáveis baseadas na informação reunida a partir de consumos e questionários de cada consumidor. Para efeitos de comparação, o algoritmo Fuzzy C-Means, (FCM) foi também utilizado para derivar modelos. Apesar deste algoritmo não ter sido desenvolvido especificamente para lidar com informação variante no tempo, este tem-se revelado eficaz em diferentes aplicações, p. ex. processar o consumo energético de um consumidor de forma a caracterizá-lo. Os melhores classificadores foram construídos aplicando o algoritmo FCM à base de dados com atributos variantes e não variantes no tempo, indicando que o algoritmo MFC não será o mais indicado para lidar com a variáveis utilizadas. Os modelos desenvolvidos atingiram um bom desempenho na detecção de perdas não-técnicas, desempenho esse que é representado por uma taxa de positivos verdadeiros a chegar aos 80% e abaixo de uma taxa de positivos falsos de 21.9%, revelando que a modelação fuzzy é adequada para a função.

Palavras-chave: perdas não-técnicas, identificadores variantes no tempo, identificadores não-variantes no tempo, Fuzzy C-Means, Mixed Fuzzy Clustering

Abstract

This thesis addresses the ever-growing problem of non-technical losses in the electricity industry, with a focus on illegal consumption behaviour from consumers. The aim of this thesis is to develop classifying models using fuzzy modelling for data-based detection of Non-Technical Losses (NTLs). More specifically, a new clustering scheme, Mixed Fuzzy Clustering, is utilized to leverage both time variant and invariant features (gathered from surveys and electricity consumption records) in the identification of illegal consumers. To evaluate the performance of the developed models, three use cases are used considering different types of features based on collected consumption data and consumer surveys. For comparison purposes, the Fuzzy C-Means algorithm was also used to derive models. Although this algorithm was not specifically developed for dealing with time-variant data, it has proven effective in many different applications, such as dealing with consumption data for consumer profiling. The best overall classifiers were computed by applying FCM to the dataset with time variant and invariant features, indicating that the Mixed Fuzzy Clustering algorithm is not best suited to deal with the data features used. The models developed achieved a good performance in detection of non-technical losses, quantified by true positive rate of up to 80% under a false positive rate 21.9%, showing that fuzzy modelling is suited for the task.

Keywords: non-technical losses, time variant feature, time invariant feature, Fuzzy C-Means, Mixed Fuzzy Clustering

Contents

Acknowledgments . . . iii
Resumo . . . v
Abstract . . . vii
List of Tables . . . xi
List of Figures . . . xiii

1 Introduction **1**
1.1 Non-technical losses and the electricity industry . . . 1
1.2 Detection of non-technical losses . . . 3
1.3 Objectives and Contributions . . . 6
1.4 Outline . . . 7

2 Classification models for detection of Non-Technical Losses **9**
2.1 Fuzzy modelling . . . 10
2.1.1 Fuzzy logic . . . 10
2.1.2 Fuzzy Inference Systems . . . 10
2.2 Fuzzy clustering . . . 13
2.2.1 Fuzzy C-Means . . . 13
2.2.2 Mixed Fuzzy Clustering . . . 14
2.3 Takagi-Sugeno model for detection of NTLs . . . 17
2.4 Fuzzy modelling to detect Non-Technical Losses . . . 19

3 Data and threat model **21**
3.1 Used databases . . . 21
3.2 Threat model . . . 22
3.3 Feature engineering . . . 25
3.3.1 Static consumption indicators (Static CI) . . . 25
3.3.2 Time variant and invariant features . . . 27
3.3.3 Temporal consumption indicators (Temporal CI) . . . 31

4 Results and discussion **33**
4.1 Evaluation criteria . . . 33
4.1.1 Receiver operating characteristic . . . 34
4.1.2 Tests on threshold determination . . . 36
4.2 Tests and parameters . . . 37
4.2.1 Static Consumption Indicators . . . 37
4.2.2 Time variant and invariant features, raw data . . . 39
4.2.3 Study of temporal consumption indicators . . . 44
4.3 Results . . . 45
4.3.1 Static consumption indicators . . . 45
4.3.2 Time variant and invariant features . . . 48
4.3.3 Temporal consumption indicators . . . 55
4.4 Overall results . . . 57
4.5 Comparison to previous works . . . 59

5 Conclusions **61**
5.1 Future Work . . . 62

Bibliography **65**

Appendices **71**

A Cost analysis for Static Consumption Indicators using Fuzzy C-Means **72**
A.1 Implementation of the tests on False Positives and False Negatives . . . 72
A.2 Results . . . 73
A.3 Conclusions . . . 76

List of Tables

3.1 Features from survey: respondent information 29
3.2 Features from survey: household information 29
3.3 Features from survey: heating information 30
3.4 Features from survey: appliances information 30

4.1 Confusion Matrix 35
4.2 Table with results from tests done to $C = [2 : 19]$ and $m = [1.4 : 0.3 : 3.8]$ when applying FCM to the dataset Static CI with Method 2 applied and 5% ratio 46
4.3 Table with results from tests done to $C = [8 : 11]$ and $m = [1.4 : 0.3 : 3.8]$ when applying FCM to the dataset Static CI with Method 3 applied and 36% ratio 47
4.4 Table with results from tests done to $C = [8 : 11]$ and $m = [1.4 : 0.3 : 3.8]$ when applying FCM to the dataset of Static CI 49
4.5 Table with results for $C = 2$ when applying MFC to the raw dataset 50
4.6 Results of the best models using a 200M dataset when different balancing techniques are tested. also applied to 1000M 51
4.7 Results from the best models when using the 1000M dataset when different balancing techniques are tested 51
4.8 Table with the best models from 200M applied to the 1000M dataset computed using a new threat model 53
4.9 Table with the best models from the 1000M dataset computed using a new threat model . 53
4.10 Table with the best models from 200M applied to the 1000M dataset using only one time series 54
4.11 Table with the best models 1000M using only one time series 54
4.12 Table with the best models when applying FCM to raw data of 200 and 1000 meters . . . 55
4.13 Table with the best models when applying MFC to Temporal CI to a database of 1000 meters and varying the range of parameters m and q 56
4.14 Table with the best models when applying FCM to Temporal CI to a database of 1000 meters 57
4.15 Summarised table 58

4.16 Comparison between the algorithms used in the thesis and the ones used in the literature 59

A.1 Confusion Matrix with the respective Cost nomenclature 73

List of Figures

1.1 Thesis Graph 7

2.1 Fuzzy inference system 11
2.2 Defuzzification methods in Mamdani fuzzy logic 12
2.3 Graphical representation of the difference between a) Euclidean distance and b) DTW . . 16

3.1 Graphical representation of the Threat Model on Meter 40 25

4.1 Graphical representation of the difference between a) Euclidean distance and b) DTW . . 34
4.2 Graphical representation of the performance evaluation methods 37
4.3 Static CI Graph 38
4.4 Static CI Graph 40
4.5 Static CI Graph 44
4.6 ROC curve for dataset with a 20% distribution and Method 2 applied 48
4.7 ROC curve for dataset with a 47% distribution and Method 3 applied 48

A.1 The average and standard deviation of critical parameters 74
A.2 The average and standard deviation of critical parameters 75

Chapter 1

Introduction

This thesis addresses the ever-growing problem of non-technical losses in the electricity industry, with a focus on illegal consumption behaviour from consumers. The aim is to create models for detection of Non-Technical Losses (NTLs) built from supervised and interpretable techniques. These same models are computed using time variant and invariant information simultaneously and compared with models built solely from static or temporal features. This study uses data in different configurations to assess the potential of Computational Intelligence, more specifically fuzzy clustering and modelling, to discern between malicious and non malicious behaviour among consumers.

This data was taken from surveys to get static (demographic) information and from meters which registered temporal information (consumption patterns from both private and commercial users). The information was used in three distinct structures: in the same format that was taken from the surveys and meters (raw data), in the form of static indicators and in the form of temporal indicators, both computed from the raw data. These datasets were used to test the performance of an algorithm which is capable of handling both time variant and invariant data and has already been used in other fields yielding promising results.

1.1 Non-technical losses and the electricity industry

Throughout the entire power grid that starts with the production of electrical energy and ends at the moment of consumption there are power losses expected from any physical process. These can be estimated by computing the difference between the amount of electricity produced and the amount consumed by the end user and they are referred to in the literature as transmission and distribution (T&D) losses.

Since no process is 100% efficient, there are registered dissipations in distribution lines, capacitors, transformers and in every other device or tool used in the system. Despite the fact that the efficiencies of these components are measured and tabled, the losses they cause can only be estimated, as the intensity of load demand, load density and energy patterns fluctuate throughout the day and the ca-

pability and configuration of the transmission and distribution system vary among installations [1]. By knowing the amount of power a company produces, the electricity that is billed to the consumer (amount of electricity that the company registered as consumed) and the energy lost from technical losses even if estimated, it is possible to ascertain the existence of external power losses. If the resulting balance is not approximately null then it is a sign of the presence of Non-Technical Losses (NTLs) in the studied system.

In developed countries, NTLs correspond to a small percentage over the entire T&D losses, 1-2% out of the total 6% that T&D losses take up from the power generated back in 2000 [2]. However, NTLs play a more significant role in the electricity industry in developing countries, where it can take up 14% of the entire power generated (this was the case in Bangladesh) [2]. These losses are a result of consumers trying and succeeding in paying for less power than what they consume. This is frequently achieved through four distinct methods: fraud by tampering with meters; stealing power from the grid through illegal connections; billing irregularities by bribing electric company officials [2].

The existence of NTLs is well documented throughout the literature [2, 3, 4, 5, 6], and acknowledged by the industry as well [7, 8]. It has a significant impact in several points around the globe, like Jamaica [3], South Africa [2], Tanzania [5], India [5, 4], among others. The worldwide consumption of electricity has always been steadily increasing, from around 12,500 TWh in 2000 to around 20,000 TWh in 2015 [9]. In 2016, in the United States of America 3700 TWh of electricity were consumed resulting in 390$ billion in revenue [10]. Citing the same governmental body, the estimated losses in the electricity transmission and distribution averaged around 5% [11]. Considering the aforementioned data, 1-2% out of the total T&D losses are a result of NTLs in developed countries. Objectively speaking this seemingly small percentage equals several billion dollars in unpaid bills. In countries like India, where T&D losses can add up to 40% of the total generated electricity, its costs to electric companies 4,5$ billion every year [12].

These high costs lead to several negative consequences. Not only are some of the theft procedures dangerous for the ones executing them (risk of electrocution when bypassing the electric connection to a meter or when rigging a line from the power source) [6, 2], but they will also impact law abiding. Since companies produce energy that is not paid for, they end up charging the difference to legitimate consumers. This forces some to steal electricity because they cannot afford to pay the increased prices.

As prices cannot rise indefinitely, there will be a saturation point where companies will lose money from NTLs and will not be able to get it back. This is already seen in developing countries where electricity theft is widespread resulting in significant losses. With less money, maintenance and upgrades to current grids will be affected. Consequently, the overload of the components forming the distribution line, due to unmonitored consumption of electricity, will trigger more frequent blackouts (complete loss of power in a region) and brownouts (voltage supplied to the system falls below the specified operating range but there is no total loss of power) [3].

1.2 Detection of non-technical losses

As mentioned before, NTLs have had noteworthy impact in the industry leading to the acknowledgement of several companies and the attention and interest of researchers. In the last two decades this has culminated in studies focused on techniques and approaches to detect, estimate and analyse NTLs [13]. Overall, the research done can be sorted into three distinct categories: theoretical studies, hardware solutions and non-hardware solutions.

Theoretical studies

Theoretical studies centre on the use of statistical techniques to find the correlation between socio-demographic, economic, market variables and electricity fraud. These techniques have the advantage of producing results that can have a high impact by helping design policy and make decisions to reduce NTLs. However, these same studies can estimate and find the drivers of aggregate NTLs but are not adequate when it comes to finding specific cases of theft and faults in metering or billing.

An example of a theoretical study is the paper [5]. It examines the phenomenon of theft in two distinct developing contexts, Zanzibar, Tanzania, and the Sunderban Islands, India, by looking into the main factors connected to electricity fraud through surveys and ethnographic fieldwork.

Hardware solutions

Hardware solutions primarily focus on characterizing and deploying equipment (e.g. metering hardware and infrastructure, signal generation, sensors) that enables the detection or estimation of NTLs. Studies on metering hardware solutions propose various alternatives which are able to completely disable some theft options, such as meter reversal and disconnection. However, their implementation on a high number of households carries elevated costs that companies are, more often than not, not willing to pay.

In [14] researchers propose a new system based on ARM-Cortex M3 processor to protect the energy meter from phase line bypassing, neutral line disconnection, whole meter bypassing and meter tampering.

Solutions on metering infrastructures involve changes on a wider scale than the solutions concerning metering hardware and may even disrupt the current power grid. For these reasons, the costs involved are considerably higher and their implementation has to be well coordinated within the utility company.

In [15] threats facing Advanced Metering Infrastructures (AMIs) were studied so that attack techniques could be computed to identify and understand the requirements for a comprehensive intrusion detection solution. With the threat model established, it was then possible to infer the information that

would be required to effectively detect attacks. As an alternative to AMIs, researchers proposed a hybrid sensing infrastructure that uses both a centralized intrusion detection system and embedded meter sensors.

Improvements to inspections and metering systems, such as AMIs composed of smart meters (SM), would help better alert electricity providers to irregularities and overloads of certain areas in the power grid [12].

Another solution found in a limited number of studies deals with signal generation and processing. Despite needing the presence of smart metering systems, these methods can usually detect all types of NTLs.

One example of this particular approach is presented in [16] where researchers propose the use of a harmonic signal generator to detect illegal consumers. These signals are introduced into the feeder to destroy appliances of illegal consumers and their effects are estimated. A cost-benefit analysis for implementation of the proposed method in India is also evaluated.

Non-hardware solutions

In this category the methodologies propose computational approaches to detect or estimate the presence of NTLs. Most studies within this field, including this thesis, focus on describing classification techniques that infer the presence of NTLs from electricity consumption or other data. These methods require low investment (unlike hardware solutions) and, if the theoretical information is accurate and extensive, they can detect most types of NTLs. However, the presence of sources of NTLs upon identification is not guaranteed and these methods might not be capable of identifying individual sources. Therefore, these algorithms are better used as an add-on to improve the effectiveness of inspection resources. These types of solutions mainly encompass the following techniques: estimation, game theory and classification.

Some studies focus on estimating the amount of NTLs from a neighbourhood or a household by using state estimation to gauge irregularities and errors in customers' demand data or using technical loss modelling to estimate aggregated NTLs.

Researchers in [17] propose state estimation combined with attacker modelling to tackle the threat of false data injection enabling the detection of NTLs that would otherwise be undetected by traditional methods. State estimation is more precise than classification techniques but requires more accurate and complete data on the distribution network loads.

Game theory based techniques model all members behaviour involved in the chain of consumption, legitimate consumers, fraudsters and the relationships with the electricity utility. One main disadvantage of these techniques is the need for solid assumptions on how frauds are carried out, only providing estimates on the detection capabilities of techniques under those assumptions. However, this leads to precise detection capabilities.

A game-theoretic approach is proposed in [18] to model the adversarial nature of the electricity theft problem. It considers two settings, unregulated monopoly and perfect competition. This framework can help detect electricity theft through the observation of the power usage behaviour.

Classification algorithms are capable of predicting the presence of electricity fraud at an end-point. Their effectiveness is reliant on the data used, the type (e.g. electricity consumption, demographic information) and the form it takes (e.g. information directly taken from surveys and meters, indicators built from that information), and how fitting the algorithm is for the problem at hand. The studies with highest impact utilize support vector machines (SVM) to deduce the presence of fraud in a household.

Researchers in [19] propose an approach which uses customer load profile information and one additional attribute to expose abnormal behaviour which was regarded in the paper to be highly correlated with NTL activities. This feature was a credit worthiness rating given by the electrical company Tenaga Nasional Berhad to its costumers that was influenced by delayed payments and intentionally avoiding bills. The end result is a shortlist of potential suspects for onsite check-ups, integrating computational solutions with inspection resources. When applied to a utility company in Malaysia, the end result was an increase in detection hit-rates from 3% to 60%.

With the vast amount of information that SM provide, data processing and later machine learning techniques are needed to sort and analyse consumer details such as power consumption (electricity usage over time), demographic information (e.g. number of people per household, number of bedrooms) and/or behavioural information (social media activity, religious/political beliefs).

Several studies have been conducted in the areas of electricity consumption profiling and theft classification. In the case of electricity consumption profiling, the authors of [20] identified and analysed five lifestyle factors reflecting social and behavioural patterns such as air conditioning and personal computer and TV usage. In [21], psychological responses and predispositions, for instance acceptance of policy measures, were studied by applying a cluster analytic approach. As for theft detection, [22] present a consumption pattern-based energy theft detector that utilizes silhouette plots to distinguish clear distributions in the database and is able to adapt its performance depending on the aim of the application by adjusting the detection delay. One of the main issues within NTLs identification is the data unbalance that is inherent to most experiments done in the field as the number of fraudulent consumers is much smaller than the number of honest customers resulting in standard classifiers to discard the minority class as it is overwhelmed by the remaining one. To overcome this problem, researches in [23] tested the individual performance of several suitable classifiers (variations to SVM, path forest and decision tree) and combinations of them, concluding that the ensembles show slightly better results at the cost of high computational power. In [24], a clustering-based novelty detection scheme was proposed which applies fuzzy clustering to extracted indicators from a dataset of legitimate consumers. This form of clustering, more precisely the Gustafson-Kessel technique, proved to be more effective in grouping similar consumers creating a more solid base line for a distance-based novelty detection model to uncover irregular data received from consumers.

As it has been stated before, the types of data usually available in the energy research field are static and temporal attributes. Most machine learning techniques focus only on one type, as they are not appropriate when dealing with the complexity of datasets containing different data types. While most studies focus on socio-demographic datasets composed only of static or time invariant features, some, like [25, 26], study the effectiveness of time variant data (electricity consumption over time) in discerning fraudulent from regular behaviour. In [25] the authors propose a novel inspection algorithm which is able to detect malicious meters in one inspection step. In [26], power consumption without the seasonal component is analysed in order to detect fraud and illogical consumption.

So as to introduce a new perspective into this field, a recently proposed method that uses time variant (spatial component) and invariant (temporal component) data as input was chosen as the basis for this work. The algorithm, entitled Mixed Fuzzy Clustering (MFC), applies fuzzy clustering to spatio-temporal data [27] to learn *if-then* rules for identification of Takagi-Sugeno fuzzy models. The clustering of temporal attributes has seen increased interest as most real life problems are characterized by datasets containing variables that change over time however, studies on cluster analysis of these features are still limited [28]. Several clustering-based approaches have been proposed to extract fuzzy rules from data, where each cluster represents a region in the product space that contains enough information to support the existence of a fuzzy input/output relationship [29].

Fuzzy modelling provides transparent models and linguistic interpretations of the decision-making process, allowing for a clear understanding of the classification of certain consumers as fraudulent. Several works in the literature have used fuzzy modelling in the most diverse areas.

In [30], a study on the sustainability of biomass for energy purposes was conducted using fuzzy-set based methods since, according to researchers, these were proven to be able to handle uncertain and vague information in environmental topics. Particularly in the paper four input variables were considered (energy output, energy balance, fertilizers and pesticides). The first two variables were chosen on the need to include information about the energy dimension of sustainability. The remainder involve information about the chemical pressure from crop cultivation which should be considered in a process of sustainability assessment. In [31], a fault detection and isolation solution is proposed with application to a wind farm benchmark model and the solution relies on a set of piecewise affine Takagi–Sugeno models, which are identified from the noisy measurements acquired from the simulated wind park. More specifically, MFC has been used in medical settings for administration of vasopressors, identification of patients with high risk of mortality and readmissions in intensive care units [32, 33, 34].

1.3 Objectives and Contributions

This thesis analysed three databases Static Consumption Indicators (Static CI), time variant and invariant features (referred to as raw data throughout the document) and Temporal Consumption Indicators. The main focus was on the application of a method that combined spatio-temporal features to better

cluster data points, Mixed Fuzzy Clustering, thus creating more precise fuzzy models. In order to create a basis for comparison, the algorithm Fuzzy C-Means (FCM) was also applied to all three datasets as this method does not distinguish between temporal and static components. The work done in this thesis is schematically shown in Fig. 1.1. Throughout the thesis parts of this scheme will be used to better visualize what is being mentioned and explained. As shown in the picture, three distinct datasets were used, one containing time variant and invariant features (taken directly from surveys and consumption records), one containing static indicators and the latter, temporal indicators both built from the time variant and invariant features. The information provided to us was regarded as information from legitimate consumers (did not commit theft). Since the goal was the detection of NTL, a set of attack vectors were computed onto the benign set. This threat model was taken from [56, 22].

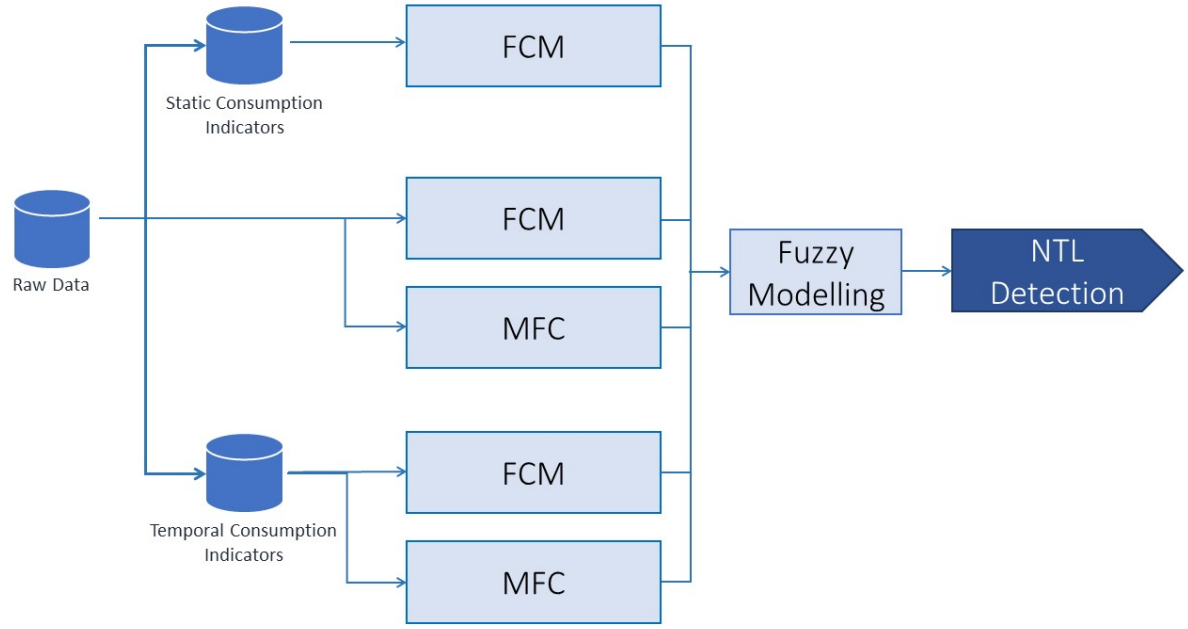


Figure 1.1: Graph of the methods applied to each dataset

The results show that the best performing models were computed by applying FCM to the dataset with time variant and invariant features. FCM also proved to yield good results with the dataset Static CI while on Temporal CI neither MFC nor FCM proved to be fitting clustering algorithms.

1.4 Outline

Chapter 2 establishes the theoretical background of the algorithms used throughout the thesis. An overview of fuzzy logic is presented followed by an exposition on the clustering methods and then the Takagi-Sugeno modelling approach. In chapter 3, the approaches taken to process the gathered information for the respective algorithms will be described. Depending on whether or not the data had time varying components, the resulting dataset would take a different configuration. Chapter 4 presents

the tests done to the three databases each having the two clustering methods applied, and evaluates the respective results. Chapter 5 analyses and discusses the results observed in the thesis. Conclusions and future work are drawn in Chapter 6. Appendix A presents a study done on the effects of varying the importance of misclassified positive cases as a deciding factor to compute a classifier. It is common for power companies to prioritize the reduction of false positives and thus it is relevant to evaluate how such concerns would influence the choice of classifiers.

Chapter 2

Classification models for detection of Non-Technical Losses

In this chapter the theoretical background of the methodologies used throughout this thesis is presented, more specifically fuzzy clustering and fuzzy modelling. The concepts of fuzzy logic and fuzzy inference systems are firstly introduced followed by fuzzy clustering and modelling.

Clustering focuses on grouping information based on their similarity. Data can be grouped into separate groups (crisp clustering) or present a degree of belonging to all (fuzzy clustering). This thesis utilizes fuzzy clustering, more specifically, the algorithms Fuzzy C-Means (FCM) and Mixed Fuzzy Clustering (MFC).

The partitioned data is then used to create classification models. The information is divided into a training set which is used to create the classifiers, and an evaluation set which contains different data points to check whether the trained classifier is able to reproduce the performance shown during training stage. So as to understand how likely each household is to be presenting an abnormal behaviour (as opposed to simply present a binary classification), and to keep the resulting classifiers as transparent and understandable as possible, fuzzy modelling, Takagi-Sugeno (TS) modelling, was chosen to compute the models. These fuzzy models are described by fuzzy IF-THEN rules which represent local input-output relations of a nonlinear system. The main feature of a TS fuzzy model is to express the local dynamics of each fuzzy implication (rule) by a linear system model. The overall fuzzy model of the system is achieved by combining the linear system models [35].

This chapter begins by explaining Fuzzy Logic, moves on to a description of the theoretical background of fuzzy clustering and modelling and the respective algorithms used.

2.1 Fuzzy modelling

2.1.1 Fuzzy logic

Using the approach of classical sets (crisp sets), an element either is or is not a member of a set. This formulation dates back to Aristotle during Classical Greece [36],

"But on the other hand, there cannot be an intermediate between contradictories, but of one subject we must either affirm or deny any one predicate.".

Despite providing a clear distinction between elements of different sets and what might characterize each grouping, by analysing the elements composing them, Boolean logic cannot solve all problems. This usually happens when expert knowledge is needed in the classification process, where there is an uncertainty in the membership degree of each element to any set. To better solve these specific problems, one can resort to Fuzzy Logic.

Fuzzy Logic was first introduced by [37] as a way to translate human decision making, for example translating ranges of possibilities between *YES* and *NO*, into a language comprehended by computers. In general terms, fuzzy sets allow their elements to possess a degree of membership to every other set. A practical demonstration of this concept can be seen when considering, for example, the rule *"a person is considered tall when their height is at least 180cm"* to evaluate two people, one being 181cm (person A) and another 179cm (person B). While one can more promptly agree that person A is tall, given the previous rule, the second individual should not be necessarily considered "small" (opposite of "tall") because they lack 1cm. In other words, it is intuitive to consider a level of belonging to the group "tall" where person A would have a much higher degree than person B. This is essentially Fuzzy Logic. The transition from "belonging" to "not belonging" is gradual and is characterized by membership functions that enable the modelling of linguistic concepts. The degree of membership ranges from 0 (not an element of the set) to 1 (a member of the set). It is worth noting that allowing only the values of 0 and 1 will turn the fuzzy classification into a crisp one.

2.1.2 Fuzzy Inference Systems

A fuzzy inference systems (FIS), also known as fuzzy rule-based systems or fuzzy models, constitute the process of mapping the inputs (features in the case of fuzzy classification) to the outputs (classes in the case of fuzzy classification) using fuzzy logic. The purpose of this approach is to compute a solution by decoding it from expert knowledge and linguistic terms through the use of fuzzy IF-THEN rules, which form the knowledge base and can effectively model human expertise in a specific field. These rules take the following structure,

If x is **A**, Then y is **B**

where **A** and **B** are linguistic variables defined by fuzzy sets on the universes of discourse x and y respectively.

A FIS is composed of 4 blocks: fuzzification interface, knowledge base (composed of the rule-base and database blocks), decision-making unit and defuzzification interface [38] which are schematically shown in Fig. 2.1

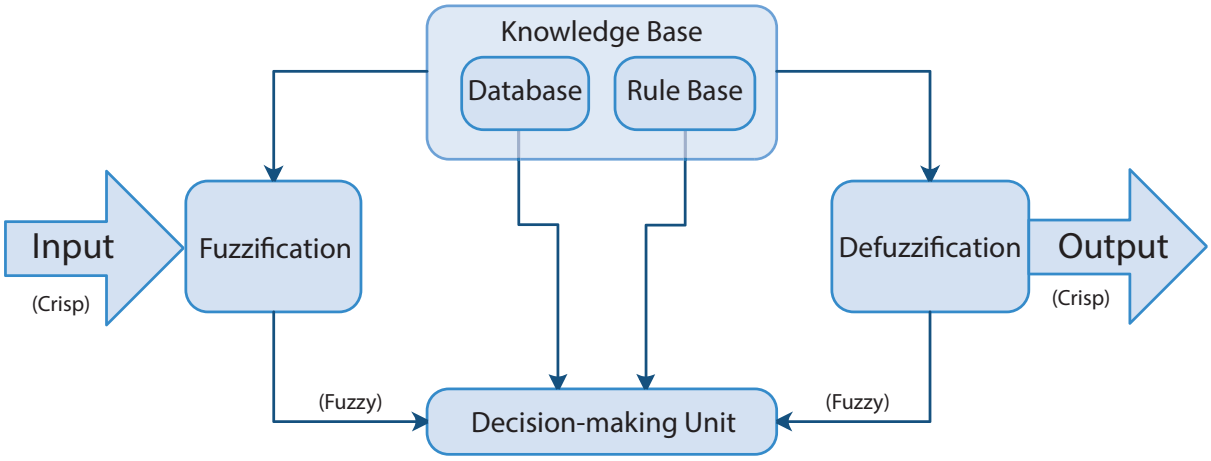


Figure 2.1: Fuzzy inference system

The first, fuzzification, transforms crisp values into grades of belonging, by means of a membership function, to fuzzy sets expressing linguistic terms. These membership functions are defined in the database block and together with the if-then rules that form the rule-base block comprise the knowledge base. The decision-making unit, also known as the inference engine (the core of the FIS) evaluates the input's degree of membership to the fuzzy output sets using the fuzzy rules. Depending on the relative importance of each rule, weight constants can be added to each of them in order to take into consideration this disparity when fuzzy output is calculated. The final stage consists on the defuzzification of the output by transforming it into a crisp value. The inference engine can reproduce the human decision-making process by performing approximate reasoning in order to achieve a control strategy. In the case of the fuzzy classifiers in this thesis, defuzzification is where the fuzzy output is turned into the binary classification of theft or no theft.

The two most well known types of fuzzy inference methods are the Mamdani and the Takagi-Sugeno-Kang inference systems.

In Mamdani-type FIS [39] both the antecedent and consequent are fuzzy propositions. As such, the output of the inference engine has a high degree of interpretability and its final crisp value is obtained through the defuzzification of the output surface. There are several methods that yield the single value from the resulting fuzzy set: centroid of the area, largest or smallest value which equals the maximum of the surface or even the mean value of those representing the surface's maximum.

The Takagi-Sugeno-Kang method [40] (or simply Takagi-Sugeno (TS) method)used in this thesis has fuzzy inputs and a crisp output (linear combination of the inputs). The consequence of each fuzzy rule

is a function, where output of TS systems is generated from a weighted average of the output functions from the fuzzy rules of the knowledge base.

Both inference systems share several similarities however the main difference lies on the fact that the output of the TS method is not a membership function, but a crisp number computed by a weighted average. By not having to handle a membership function as the output of the inference engine to compute the final crisp value, the computational power required for the process decreases making it efficient and suitable to work with optimization and adaptive techniques.

The scheme in Fig. 2.2 utilizes a two-rule two-input fuzzy inference system to show different types of fuzzy system mentioned above. While Type 1 was the first one to be developed, it received some criticism for the lack of uncertainty associated with it, something that is characteristic of Fuzzy Logic. Type 2 was then proposed in [41] and addressed the criticism by incorporating uncertainty about the membership function into fuzzy set theory. This second approach is the Mamdani type fuzzy system where the output function is determined based on overall fuzzy output. Type 3 is the Takagi-Sugeno type fuzzy system.

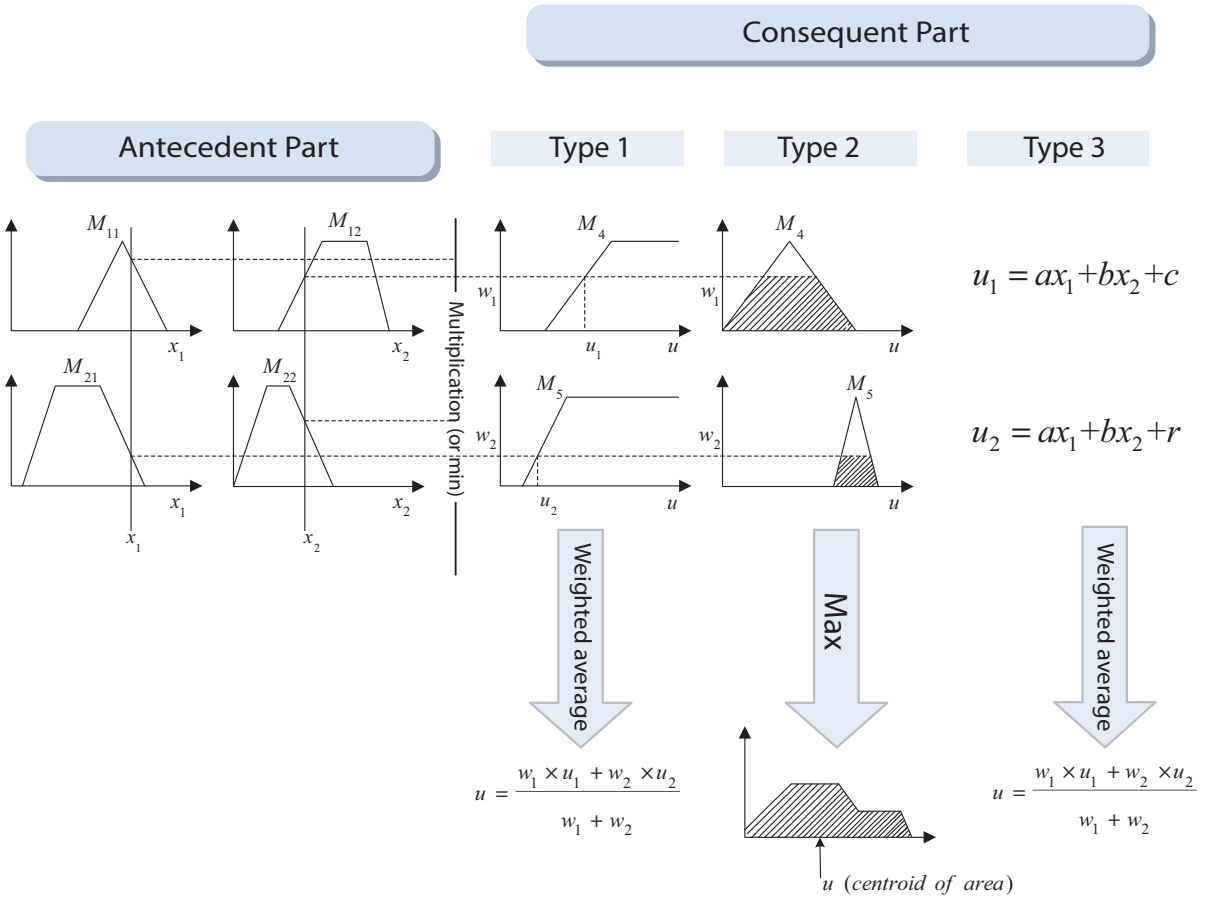


Figure 2.2: Defuzzification methods in Mamdani fuzzy logic

2.2 Fuzzy clustering

Cluster analysis or clustering is the task of grouping a set of unlabelled objects in such a way that the degree of similarity is higher among objects within the same group or cluster (intra-group similarity) than objects from different clusters (inter-group similarity).

It is one of the key tasks in exploratory data mining, and a common technique for statistical data analysis, used in many fields such as machine learning [42], pattern recognition [43], image analysis [44] among others.

2.2.1 Fuzzy C-Means

Fuzzy C-Means (FCM) is an unsupervised partitioning algorithm with frequent implementation in pattern recognition and image processing problems [44, 43, 45]. It was first proposed in [46] and later improved in [47]. This method introduces Fuzzy Logic to the hard clustering framework, seen in K-Means [48] for example, by allowing each data point $x_j = [x_{j1}, x_{j1}, ..., x_{jR}]$, R being the number of features in the input matrix, to belong to all clusters C by means of a membership degree as opposed to hard clustering where each point can only belong to one cluster.

FCM is an iterative optimization that minimizes the following objective function,

$$J_m = \sum_{i=1}^{N} \sum_{j=1}^{C} u_{ij}^m d_{ij}^2(x_j, c_i) \tag{2.1}$$

where N is the number of samples, C the number of clusters, m is the fuzzification parameter which has to range within $[1; \infty[$ and defines how fuzzy or crisp the end clusters will be, u_{ij}^m is the membership degree of element i to the cluster j, and $d_{ij}^2(x_j, c_i)$ is a similarity measure, commonly viewed as a proximity measure and thus handled as a distance variable, between the point x_j and the cluster centre, also known as prototype, c_i.

The algorithm starts with the initialization of the partition matrix U which is formed with the membership degrees of every element N to every cluster C.

$$U = \begin{bmatrix} u_{11} & \dots & u_{1C} \\ \vdots & \ddots & \vdots \\ u_{N1} & \dots & u_{NC} \end{bmatrix}$$

This initialization can be performed randomly by giving aleatory belonging values of each point to every cluster. However, the membership degree has to obey the following constraints

$$u_{ij} \in [0,1] \quad \forall i; \qquad 0 < \sum_{j=1}^{N} u_{ij} < N \quad \forall i; \tag{2.2}$$

$$\sum_{i=1}^{C} u_{ij} = 1 \quad \forall i. \tag{2.3}$$

Afterwards, through an iterative process the fuzzy partitioning is carried out and the objective function in 2.1 is optimized. This is done by computing the cluster centres with the equation

$$c_i = \frac{\sum\limits_{j=1}^{N} u_{ij}^m x_j}{\sum_{j=1}^{N} u_{ij}^m} \tag{2.4}$$

also known as prototypes. These are the mean of all points, weighted by their degree of membership to the cluster. Next the partition matrix is updated using equation (2.5) which calculates the membership degree of element i to cluster j.

$$u_{ij} = \frac{1}{\sum\limits_{k=1}^{C} \left(\frac{d_{ij}^2}{d_{kj}^2}\right)^{\frac{1}{m-1}}} \tag{2.5}$$

Once this is done a stopping condition is checked and if cleared, the iteration stops. In this project, the cycle halts when $max|u_{ij}^{z+1} - u_{ij}^{z}| < \epsilon$ is verified where z is the iteration step and ϵ is the stopping condition.

In the present study, each sample is assigned to each cluster with a certain degree of membership. This degree is proportional to the distance between the sample and the cluster prototype, which in a general way can be computed as

$$d_{ij}^2(x_j, c_i) = ||x_j - c_i||^2 = (x_j - c_i)^T A_i (x_j - c_i) \tag{2.6}$$

2.2.2 Mixed Fuzzy Clustering

Mixed Fuzzy Clustering is a novel clustering method based on Fuzzy C-Means [33] which allows the clustering of time variant (remain constant over time) and time invariant (change over time) features simultaneously. This algorithm aims at providing a solution for dealing with longitudinal misaligned data where the length of time variant features is different, and to account for misalignment through the use of Dynamic Time Warping (DTW) distance. This approach clusters the dataset using an augmented form of the FCM [49]. The main difference between them relies on the distance function. In the augmented version, a new parameter λ is calculated, weighting the importance to be given to each feature [50].

Each sample x_i, with $i = 1, ..., n$, is characterized by static features or time invariant, x^s, and by time variant features or time-series, X^t:

$$x_i = (x_i^s, X_i^t), \tag{2.7}$$

where x^s is a NxR matrix with N equal to the number of entities and R equal to the number of time invariant features, and X^t a NxP matrix with P being the number of time variant features. Each entry of X^t is an array of values, x_{ip}^t of length Q dependent on p,

$$x_{ip}^t = (x_{i1}^t, x_{i2}^t, ..., x_{iQ(p)}^t), \tag{2.8}$$

The time invariant cluster centres l, also known as the time invariant prototypes l, v_l^s, and the time variant cluster centres l, v_l^t for feature p are calculated through equations (2.9) and (2.10) respectively.

$$v_l^s = \frac{\sum_{i=1}^{N} u_{li}^m x_i^s}{\sum_{i=1}^{N} u_{li}^m} \tag{2.9}$$

$$v_{lp}^t = \frac{\sum_{i=1}^{n} u_{li}^m x_{ip}^t}{\sum_{i=1}^{N} u_{li}^m} \tag{2.10}$$

Each cluster l has its own set of feature weights λ, calculated separately for x^t and x^s in every dimension:

$$\lambda_{lr}^s = \frac{1}{\left(\sum\limits_{1<k\leq R} \frac{\sum\limits_{i=1}^{N} u_{li}^m ||(x_{ir}^s, v_{lr}^s)||^2}{\sum\limits_{i=1}^{N} u_{li}^m ||x_{ik}^s - v_{lk}^s||^2} + \sum\limits_{R<k\leq R+P} \frac{\sum\limits_{i=1}^{N} u_{li}^m ||(x_{ir}^s, v_{lr}^s)}{\sum\limits_{i=1}^{N} u_{li}^m \delta(x_{i(k-R)}^t, v_{l(k-R)}^t)} \right)^{\frac{1}{q-1}}} \tag{2.11}$$

$$\lambda_{lp}^t = \frac{1}{\left(\sum\limits_{1<k\leq R} \frac{\sum\limits_{i=1}^{N} u_{li}^m \delta(x_{ip}^t, v_{lp}^t)}{\sum\limits_{i=1}^{N} u_{li}^m ||x_{ik}^s - v_{lk}^s||^2} + \sum\limits_{R<k\leq R+P} \frac{\sum_{i=1}^{N} u_{li}^m \delta(x_{ip}^t, v_{lp}^t)}{\sum\limits_{i=1}^{N} u_{li}^m \delta(x_{i(k-R)}^t, v_{l(k-R)}^t)} \right)^{\frac{1}{q-1}}} \tag{2.12}$$

The variable q offers a degree of feature discrimination and its value ranges from 1 to ∞. According to [50], as *q* approaches 1, λ will tend to take binary values, meaning one feature will be labelled 1 for being the most relevant in the computation of the distance between samples and the prototypes. On the other hand, if q took higher and higher values, the same feature weights will have the same levels of relevancy thus making the process of feature selection irrelevant.

As previously mentioned, λ alongside a distance function δ was used to compute the distance between an entity and the time invariant and variant prototypes of a cluster *j*, through equation (2.13).

$$d_{ji}^2 = \sum_{r=1}^{R} \lambda_{jr}^s ||x_{ir}^s - v_{jr}^s||^2 + \sum_{p=1}^{P} \lambda_{jp}^t \delta^2(x_{ip}^t, v_{jp}^t) \tag{2.13}$$

The distance function δ between two vectors a and b of same length M, with $i = 1, 2, ..., M$ and

$l = 1, 2, ..., M$, is given by:

$$\begin{cases} \boldsymbol{\delta}^2(\mathbf{a}, \mathbf{b}) = (a_1 - b_1)^2 + ... + (a_M - b_M)^2, & \text{if EUC} \\ \boldsymbol{\delta}^2(\mathbf{a}, \mathbf{b}) = \gamma(M, M), & \text{if DTW} \end{cases} \tag{2.14}$$

where $\gamma(i, l) = ||a_i - b_l||^2 + min\{\gamma(i-1, l), \gamma(i-1, l-1), \gamma(i, l-1)\}$ and $\gamma(1, 1) = ||a_1 - b_1||^2$.

Although the calculation of the distance between a sample and the time invariant is only made through the Euclidean distance method, the distance to the time variant centres can be done through either Euclidean distance or Distance Time Warping (DTW). This latter approach fundamentally differs from the former on how it handles two time series. While Euclidean distance directly compares two points in the same time instance without considering differences in vector size (either from different time samples or different start/end recording) or misalignment between them, DTW takes both vectors and tries to align them so as to minimize their difference thus creating what is called a *warping path*. Taking two time series, $Q = q_1, ..., q_m$ and $R = r_1, ..., r_n$, for the computation of DTW, a distance matrix (mxn) is constructed where each (i, j) matrix element contains the distance value of point q_i to point r_j. A warping path is then created as a set of matrix elements that are bound by three rules: boundary condition, continuity, and monotonicity. The boundary condition dictates that the warping path starts and finishes in diagonally opposite corner cells of the matrix, namely $w_1 = (1, 1)$ and $w_1 = (m, n)$. The continuity constraint restricts the allowable steps to adjacent cells. The monotonicity constraint forces the points in the warping path to be monotonically spaced in time.

Both distance measures are depicted in Fig.2.3

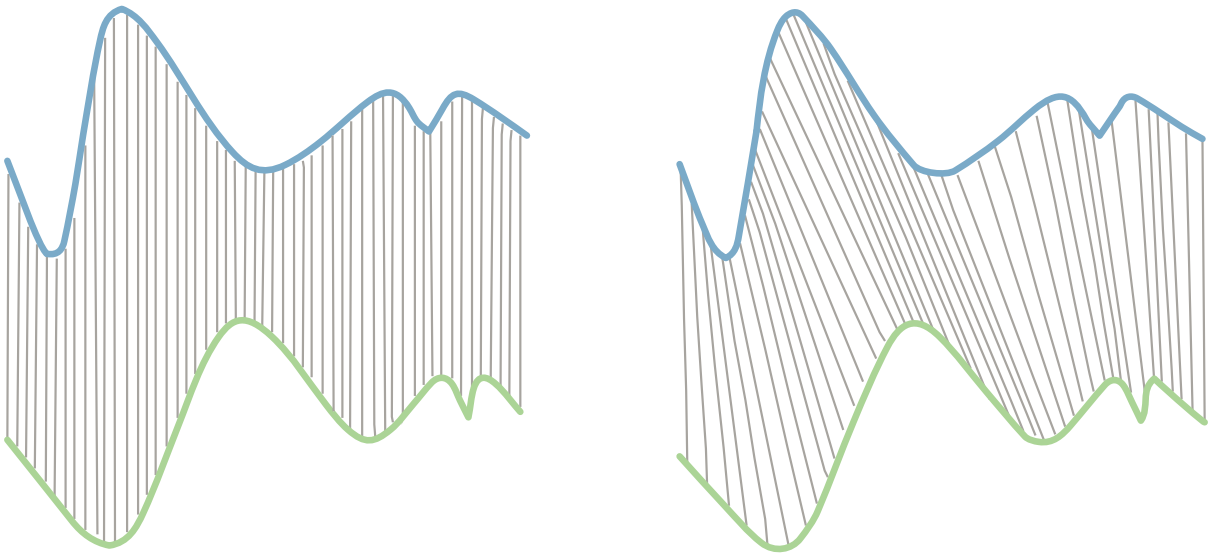


Figure 2.3: Graphical representation of the difference between Euclidean distance (on the left) and Dynamic Time Warping (on the right)

One of the inputs to the Mixed Fuzzy Clustering algorithm is the partition matrix $U = [u_{ij}]$ which is built from the degree of membership of each sample i to each and every cluster j. Given N samples

and C number of clusters, the matrix will be CxN and each entry is calculated using the equation (2.15)

$$u_{ij} = \frac{1}{\sum_{g=1}^{C} \left(\frac{d_{ji}^2}{d_{gi}^2}\right)^{\frac{1}{m-1}}} \tag{2.15}$$

and subject to the following constraints

$$u_{ij} \in [0,1] \quad \forall i; \qquad 0 < \sum_{i=1}^{N} u_{ij} < N \quad \forall i,j; \tag{2.16}$$

$$\sum_{j=1}^{C} u_{ij} = 1 \quad \forall j; \tag{2.17}$$

As previously mentioned, this is an augmented version of FCM and as such applies an augmented form of the objective function

$$J = \sum_{j=1}^{C} \sum_{i=1}^{N} u_{ij}^m d_{ji}^2 (v_j^s, v_{jp}^t, x_i) \tag{2.18}$$

2.3 Takagi-Sugeno model for detection of NTLs

A classifier is a mathematical model that assigns a class label to an object, based on the characteristics of the objects. In particular, a fuzzy classifier [51] is any classifier that uses fuzzy sets or fuzzy logic during the course of its training or operation. The input data (characteristics of each data point) comes in the form of a vector containing values for the features considered in the classification problem. In the field of electricity consumption, whether it is fraud detection or consumption profiling, these features are frequently related to demographic and/or psychological information, consumption patterns and others. In this thesis these features take form of raw data (data directly taken from survey and meters without any feature transformation) or indicators (created by transforming raw data into a smaller set of features that characterize the same reality differently).

Fuzzy models are transparent “grey box” that allow the approximation of previously unknown non-linear systems to be modelled using a number of linear and understandable sub-models responsible for distinct sub-domains. The advantage of using fuzzy models over other non-linear modelling methods, relies on not only providing transparency but also linguistic interpretation to the decision process in the form of if-then rules. These may help company employees understand the weight certain pieces of information have in the final identification of an illegal consumer as these fuzzy rules describe a local input-output relation. In the case of NTL detection studied in this thesis, it was found that the resulting rules were complex and difficult to read as they were dependent on many variables (demographic features for example) and characterized many clusters. Fuzzy models use a training set in order to discover potentially predictive relationships between inputs and outputs, and a test set (invariably smaller

than the training set) to validate said relationships. For the binary classification case, each discriminant function consists of rules,

$$\begin{aligned} R_j &: \text{If } x_1 \text{ is } A_{j1} \text{ and ... and } x_M \text{ is } A_{jM} \\ &\text{then } y_j(x) = f_j(x), j = 1, 2, \ldots K \end{aligned} \tag{2.19}$$

where f_j is the consequent function of rule R_j and $M = r+q$, the number of features used. The output of the discriminant function $y_j(x)$ can be interpreted as a score (or evidence) for the positive example given the input feature vector x.

The number of rules K of the type R_i and the antecedent fuzzy sets A_{jh} are determined by fuzzy clustering in the product space of the input variables. The consequent functions $f_i(x)$ are linear functions determined by ordinary-least squares (OLS) in the space of the input and output variables.

The degree of activation of the jth rule is given by

$$\beta_j = \prod_{h=1}^{M} \mu_{A_{jh}}(x), \tag{2.20}$$

where $\mu_{A_{jh}}(x) : \mathbb{R} \to [0; 1]$.

In the TS model, the overall output is a weighted average of individual rule outputs, β_j being the weight, and the inference is reduced to a simple algebraic expression:

$$y(x) = \frac{\sum_{j=1}^{K} \beta_i f_i(x)}{\sum_{j=1}^{K} \beta_i} \tag{2.21}$$

The output of the system is often continuous but since it needs to identify classes (in this case two groups, theft and regular consumption) a threshold has to be established. For that, the model uses the training data to set a threshold that creates the most accurate classifier for that specific set of data points. This particular model (with the threshold previously set) is then used with the test set to evaluate its performance.

A sample x_j is considered positive if the score is higher than a certain threshold γ, in other words, $y_j(x_j) > \gamma$.

$$y_j(x_j) > \gamma \tag{2.22}$$

The performance of the classifier is then judged using metrics such as AUC, Accuracy, TPR and FPR which all vary in the interval $[0, 1]$. These metrics will be explained in section 4.1

2.4 Fuzzy modelling to detect Non-Technical Losses

Several papers have been published on the implementation of fuzzy modelling regarding the work done on the detection of NTLs. In [52] researchers apply fuzzy logic to determine a suspicion level of fraud for each customer. In [53], the authors continue the work of [19] by implementing a FIS in an SVM based fraud detection model. Using onsite inspection as feedback, while SVM achieved a hit-rate or True Positive Rate of 60%, adding the FIS as a postprocessing scheme improved the hit rate to 70%. In [54], after using FCM to cluster consumers with similar electricity usage profiles, a fuzzy classification is performed to identify anomalies, such as non-technical losses, in consumption patterns of households.

This thesis focused on evaluating classifying models built using fuzzy modelling on both time variant and invariant features. Although these two variables are traditionally featured in the research pertaining to this field, with demographic features and electricity consumption records, a study has not been conducted on the use of both to detect NTL. The results gathered helped understand the importance of considering two distinct sets of features during the classification process.

Chapter 3

Data and threat model

3.1 Used databases

All methods studied in this project used data from the same source. This information was provided by the Commission for Energy Regulation who conducted surveys and collected data on the electricity consumption of around 5,000 Irish homes and businesses from the Summer of 2009 to the Winter of 2010 and made them available at the Irish Social Science Data Archive (ISSDA) [55]. The survey information (number of adults and children, type of house, heating provided through gas, ...) will be referred to as static information or time invariant features while the consumption patterns will be referred to as time series, time variant features or temporal features.

The clustering algorithms were applied to three distinct datasets, all built from the data provided by ISSDA. The first dataset was composed of time variant and invariant features taken directly from the surveys and records of Smart Meters (SM) provided by ISSDA. The remaining two datasets were built with indicators computed from the data present in the first dataset, surveys and SM records. One of these two databases had static indicators (indicators with one value per registered day) and the other had temporal indicators (these were computed during different times of the day allowing for a daily evolution of each indicator). These were the same indicators but were computed differently.

The algorithm MFC was designed to be applied to datasets containing both static and temporal features. As such, for the static indicator dataset MFC was not used. As explained in the previous chapter, FCM does not compute the time variant nature of features such as the electricity consumption records. So while this algorithm can still be applied onto these features, it was used as a direct way to compare the influence of each different dataset on the resulting classifiers. The use of FCM also helped determine the importance of considering the time variant nature of the studied features for the classification problem as it opposed MFC in its approach to time series.

3.2 Threat model

Despite the different approaches to the input data, the same attack vectors were computed onto the data points. For each sample representing a normal household, 8 attacks were applied. For this project, the provided information by ISSDA was treated as data from legitimate consumers and was used to apply a threat model which represents distinct ways a consumer might steal from electricity providers. All the attacks involved manipulating data sent from SM to utility companies, either with the gol of lowering the average consumption or shifting consumption from periods where tariffs are higher. The idea behind the application of these attack vectors is to create a flexible model with information regarding the same consumer interfering with his records through a variety of methods all the while keeping the unaltered consumption pattern for theft identification. The threat model used was taken from the work [56] which in turn was based off of [22].

According to [56], the attack procedures can be grouped in two different categories: attacks that started during the meters data gathering (non zero-day attacks) and attacks which had already started by the time the information was collected (zero-day attacks). The different attacks and scenarios (zero-day and non zero-day) are presented below with the first attack, for example, presenting the non zero-day scenario as h_1 and the zero-day scenario as h_{10}. The equations use the following notation: $m_i^{d,t}$ are the meter consumption readings from consumer i in day d for hour t. This results in m_i^d = $(m_i^{d,1}, m_i^{d,2}, ..., m_i^{d,24})$ representing the 24 hour vector of metered data of consumer i on day d. All parameters and variables used in each equation were taken directly from the aforementioned papers. Although some attacks clearly reduced the average consumption, others focus on lowering consumption in periods of the day where tariffs are higher (from 18h to 24h).

- h_1; h_{10} : constant random reduction of consumption.
  Every data point of the same time period is multiplied by the same constant, which is randomly chosen within the range [0;1].

$$h_1(m_i^{d,t}) = \alpha m_i^{d,t} \ , \qquad \alpha = random(0.1, 0.8) \tag{3.1}$$

- h_2; h_{20} : registering zero consumption for a random period of the day; zero day scenario.
  A random period of the day, a starting point and duration randomly chosen, is selected to have its

electrical pattern set to 0.

$$
\begin{aligned}
&h_2(m_i^{d,t}) = \beta^h m_i^{d,t} \\
&\beta_t = \begin{cases} 0, & t_{start} < t < t_{end} \\ 1, & \text{else} \end{cases} \\
&t_{start} = random(0, 20) \\
&\delta = random(4, 24) \\
&t_{end} = t_{start} + \delta, \quad t_{end} \leq 24
\end{aligned}
\tag{3.2}
$$

- h_3; h_{30} : random hourly reduction of consumption.
  Each entry belonging to the chosen time window is multiplied by a different randomly picked constant between [0;1]. This attack model is similar to the first except instead of one common constant for the whole time period, as mentioned in h_1 and h_{10}, each hourly record is combined with a different constant.

$$h_3(m_i^{d,t}) = \gamma_t m_i^{d,t} \ , \qquad \gamma_t = random(0.1, 0.8) \tag{3.3}$$

- h_4; h_{40} : random hourly consumption pattern with reduced average consumption.
  After calculating the average consumption in that period of time, each time sample of said period is multiplied by a distinct randomly picked constant within [0;1]. This method makes the consumption profile very different from other consumers with similar demographic characteristics as there are lower consumption peaks, which are normal at the last hours of each work day. This can be picked up by utility companies by checking, for example, the difference in consumption peaks between consumers and difference between daily lows and peaks of the same consumer.

$$h_4(m_i^{d,t}) = \gamma_t \, \mu(m_i^d) \ , \qquad \gamma_t = random(0.1, 0.8) \tag{3.4}$$

- h_5; h_{50} : constant hourly consumption equal to the average.
  The entire daily consumption pattern is replaced by the average of the electricity usage in that same day. Although the daily average is not changed, as it lowers the consumption during peak hours (normally from 18h to 24h) it will decrease the monthly bill since these hours tend to have higher tariffs. This is another attack which can be quickly picked up by companies as consumption profiles have lows and peaks throughout the day and this attacks changes it to a constant value.

$$h_5(m_i^{d,t}) = \mu(m_i^d) \tag{3.5}$$

- h_6; h_{60} : reversed hourly consumption.

The daily electrical pattern is reversed. Although, this might not change the average consumption it will move the electricity usage away from periods of high consumption. If the consumer has similar patterns to others who have high consumption levels at the end of the day for example, shifting those records to the beginning of the same day (early hours of the morning) means the consumers will end up paying less since prices are higher when most consumers are active and lower when they are not.

$$h_6(m_i^{d,t}) = m_i^{d,24-t} \tag{3.6}$$

- h_7; h_{70} : shift of consumption from peak hours to the rest of the day, turning peak hours consumption to the average.

  The peak of electricity consumption in each day and the consumption of the following X hours (a three hour window was used for this project, but different values of X could also have been considered) are computed and their difference to the average consumption of that day is calculated. From here, that difference is then distributed over the remaining data points of that day. The reason behind this model is the same as the previous one, a change in the electricity bill (change in the consumption during hours when the price is higher) despite maintaining the average consumption.

$$\begin{aligned} & h_7(m_i^{d,t}) = \begin{cases} m_i^{d,t} - \delta\, m_i^{d,t}, & p_{start} < t < p_{end} \\ m_i^{d,t} + \epsilon\,/21, & \text{else} \end{cases} \\ & p_{start} \text{ is the starting hour of the highest consumption three hour period} \\ & p_{end} = p_{start} + 3 \\ & \epsilon = \sum_{j=1}^{3} m_i^{d,p_{start}+j-1} \end{aligned} \tag{3.7}$$

- h_8; h_{80} : change the consumption data so that it copies the one from a consumer with lower average of electricity consumption.

  The electricity pattern of each day is replaced with one from another consumer with lower average. If during that day the fraudulent consumer is the one with the lowest average, then he does not alter his record.

$$\begin{aligned} & h_8(m_i^{d,t}) = m_r^{d,t} \\ & r \text{ is a random consumer with } \mu(m_r^{d,t}) < \mu(m_i^{d,t}) \end{aligned} \tag{3.8}$$

In order to better visualize the effects of the diverse ensemble of attack vectors on a normal consumption pattern the Fig. 3.1 shows the application of the Threat Model onto a specific meter on a particular day.

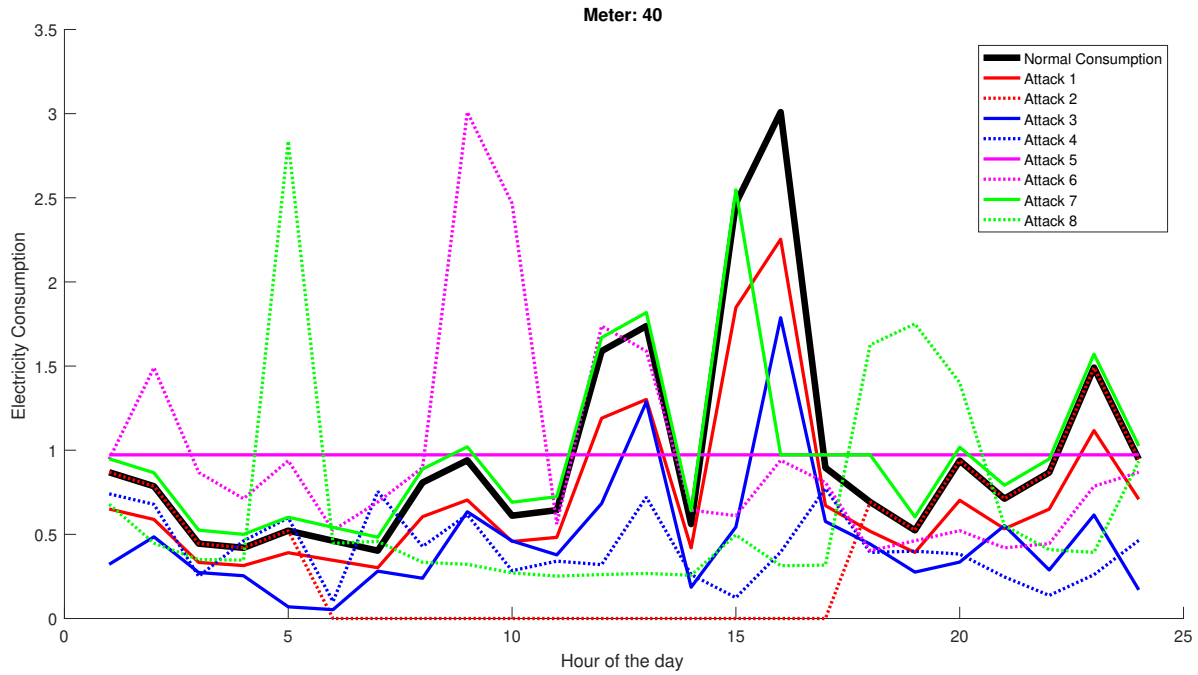


Figure 3.1: Graphical representation of the Threat Model on Meter 40

3.3 Feature engineering

Since the data for the TS and MFC algorithms had to be in different formats, the way they were processed to be ready for computation took distinct approaches.

3.3.1 Static consumption indicators (Static CI)

Rather than just keeping the same characterizing variables from the original set of time variant and invariant features, indicators were computed to make future models as transparent and interpretable as possible. These new features reflect changes from past consumption patterns and from other consumers with similar characteristics when comparing answers to the survey. The computed dataset presents the following information:

- Meter : Serial number that identifies each meter
- Date : The date in which the data was recorded
- Attack : The type of attack each sample is meant to represent ($None$ for normal consumption and h_i for the attack i)
- I_1 : Indicator of consumption variation. Ratio between current and past consumption.
  The first indicator, I_1, is a ratio between the consumption of the last α days and the last β periods of α days.

$$I_1(i,d) = \frac{\sum_{j=1}^{\alpha}\sum_{k=1}^{24} m_i^{d-j,k}}{\frac{1}{\beta}\sum_{l=1}^{\beta}\sum_{j=1}^{\alpha}\sum_{k=1}^{24} m_i^{d-j-\alpha\beta,k}} \tag{3.9}$$

- I_2^e and I_2^c : Indicators of hourly consumption pattern change.
  The indicator I_2^v relates the hourly pattern of a day with the mean hourly pattern of the α days before. If *v* is the euclidean distance (*v* = *e*), changes in absolute consumption will be the most relevant for the indicator. If *v* is the Pearson correlation (*v* = *c*), changes of dynamic can be detected.

$$I_2^v(i,d) = v(m_i^d, \mu(m_i^{d-1-\alpha}, ..., m_i^{d-1})) \tag{3.10}$$

- I_3 : Indicator of consumption difference in comparison to consumers with similar characteristics.
  I_3 portrays the difference of selected consumer to one $r \in R$ with highest similarity in terms of demographic information. Compares the mean consumption of the last β days to the mean consumption for the same days for the consumers with the most similar characteristics. *R* are the τ consumers in 1; 2; ...;*N* with lowest similarity between their characteristics s_i.

$$I_3(i,d) = \frac{\frac{1}{\beta}\sum_{l=1}^{\beta}\sum_{k=1}^{24} m_i^{d-j-\beta-1,k}}{\mu(\{\frac{1}{\beta}\sum_{l=1}^{\beta}\sum_{k=1}^{24} m_r^{d-j-\beta-1,k} \forall r \in R\})} \tag{3.11}$$

- I_4^e and I_4^c : Indicators of hourly consumption pattern difference in comparison to consumers with similar characteristics.
  The last indicators I_4^e and I_4^c represent the difference in the hourly consumption pattern between consumer *i* and others with the highest similarity by relating their mean hourly consumption on the α days.

$$I_4^v(i,d) = v(\mu(m_i^{d-\alpha}, ..., m_i^d), \mu(\{(m_r^{d-\alpha}, ..., m_r^d) \forall r \in R\})) \tag{3.12}$$

Because the data was compiled from a real-life setting, some samples display missing information in certain variables (this information was represented by NaN). Three approaches to this problem were considered: turn all values of NaN into 0; turn all NaN into the average value of the corresponding feature; turn only the NaN entries that characterize the attack models 5 and 50 into 0 and the rest into the average values. Recalling the explanation given in section 3.2 for the threat model, these two attack models in particular simulate the replacement of a regular consumption pattern with a constant consumption sequence. This leads the indicators I^2 and I^4 (specifically the indicators which use the Pearson correlation, I_c^2 and I_c^4) to present errors (NaN) as they characterize differences with past consumption patterns and in relation to similar consumers respectively. As these attacks replace consumption variations with a constant signal, there was no change to be evaluated by these indicators

hence the NaN values. Since the remaining NaN present in other features throughout the database cannot be easily explained (therefore resolved promptly), to deal with them, a methodology that disrupts as little as possible of the modelling process must be implemented. Thus, tests were run to determine the best course of action when dealing with erroneous data.

The last step in the data processing stage concerns the division of the database into training/testing sets. Seeing as the data points contain information of the same meter in different occasions of the year, the division method must ensure that every sample of each meter (from the 5 days in the 4 seasons of the year), both the normal consumption sample and its corresponding attack samples, has to be in the same set (either training set or testing set). The distribution between training/testing data is kept at 60%/40% throughout the thesis.

3.3.2 Time variant and invariant features

As the information provided by ISSDA contains data from both households and businesses, an early selection had to be made due to the entirely distinct consumption behaviours these two sets exhibit and thus, have to be dealt separately. To that end, the data points concerning private consumers were used and the ones regarding small and medium sized companies were discarded. The information available regarding private consumers is separated into two sections: one providing the results from the surveys and the other one the electricity consumption over the span of those two years.

Each of these two datasets, electrical consumption and survey results, presented different formats and, therefore, distinct challenges in their analysis.

Electricity consumption data

The electricity patterns of each meter were published in different .txt files, each containing information regarding distinct groups of meters. The number of recordings varied from meter to meter. Initially, each was set up to record consumption every 30 minutes (48 recordings per day), except for the days corresponding to the Daylight Saving Time (DST) schedule where they show 50 and 46 points. However, all meters had missing information throughout the day. Instead of points showing no consumed electricity, there were periods where the meter itself did not register, only to resume later on. As a result, there were, for instance, days with around 30 entries out of the 48. A similar procedure to what was used in [33] was also used in this project to handle absent data. Depending on the ratio between registered and missing information, the entire electricity record of a specific meter had to be discarded. On the other hand, if the ratio was above a certain threshold, the amount of data missing was sufficiently small relatively to the data points registered, the information missing could be filled automatically without compromising the database. With this in mind, when assessing the absent entries, these were replaced with information from the last recorded data point, effectively applying the method Zero Order Hold (ZOH). This approach has been applied by other researchers [33].

In order to manipulate a small but representative database due to computational constraints (more data to analyse would increase computation time), instead of using the complete set of data points from the two years of electricity records, the same three days of each season of the year (a total of 12 days per year) were picked from every consumer. Given that the records start at day 195 (July 14^{th} 2009) and end at day 730 (December 31^{st} 2010), information regarding Spring of the first year, the first month of Summer of the first year and the last two months of Winter of the second year were not registered and therefore not evaluated in this project. A time window was then applied to the selected days so as to provide a record of past behaviours. In spite of having information ranging from two years, the time window ranged from 5 days prior to each of the three days per season. This decision had in mind the procedure through which the algorithms from the electricity providers operate to detect fraudulent consumers, as officials, after deciding to inspect a certain client on a particular day, would only have access to the user's past information.

From the selected days (12 days per year) and the days within the respective time windows, if one was to belong to the DST calendar (a day with one hour added or removed), it would mean the time series of that day has a different size than the remaining days. This would create differently sized time series which would have to be handled with DTW. Despite being one of the advantages of using MFC when dealing with possible misaligned time variant features [32], the computational power needed to use DTW over Euclidean distance is greatly exacerbated leading to an increase of the time needed to run the experiments. As a solution, it was decided to discard the information from the added hour (49^{th} and 50^{th} entries) on the days when the clock would advance 60 minutes or apply ZOH to the last non 48^{th} entry, when an hour is removed. Although it is adding information to a period of time that never existed and deleting valid information, the impact on the results from the algorithm is negligible. In order to keep confidentiality in check, the data was resampled to 24 points per day by adding every two points of the time series. After analysing the consumption data over a weekly period, the difference between patterns from weekends and working days was substantial enough to justify discarding the weekend information from the database. Without this decision, the classification algorithm would end up comparing very distinct time series within the same time window making the job of discerning between theft and legitimate behaviour more complex and less accurate.

Survey data

With respect to the surveys, these consisted of 234 questions made to both private and corporate consumers. However certain questions were directed specifically to one of these groups. This, allied with the fact that certain variables or household characteristics could more heavily influence electricity consumption, justified the use of Feature Selection. Applying a rating from 0 (completely irrelevant for consumer profiling) to 3 (of the highest importance) to all 234 questions, choosing the ones with score above 1 and removing the ones that showed a skewed distribution in the answers (questions where over 85% gave the same answer) left 39 variables eligible to be used for the study. Skewed distribution

in answers would result in some features to not have representation to more than one answer during tests, specifically when dividing between training and testing data. As a practical example, despite the fact that washing machines consume a lot of power, since practically every household has one using this feature to distinguish between consumers is not viable. These 39 variables with the names given by ISSDA are shown in Tables 3.1 to 3.4.

Table 3.1: Features from survey: respondent information

Feature	Description
Age	Age of the respondent
Employment	Employment status of the respondent
Social_class	Social class of the respondent
Education	Education level of the respondent
Income	Income of the respondent before tax
Living_situation	Living situation (e.g. alone, adults, adults and children)
N_adults	Number of adults in the household
N_children	Number of children in the household

Table 3.2: Features from survey: household information

Feature	Description
Home_type	Type of home (e.g. apartment, terraced)
Home_age	Household age
Bedrooms	Number of bedrooms
CLF_lightbulbs	Fraction of CLF light bulbs (e.g. around a quarter, about half)
Doublegazed_windows	Fraction of doubleglazed windows
Attic_insulated	Number of years the attic has been insulated
Externalwalls_insulated	Insulation of external walls: yes, no, don't know
Internet	Access to the internet

One of the problems that surfaced after analysing the database was that the consumers did not fill the queries properly. Specifically, when asked to describe the people in the household, if the answer was "I live alone", the person surveyed did not answer 1 to the question related to the number of people over the age of 15 nor 0 to the question related to the number of people under the age of 15. Instead these questions show an invalid response. A similar situation happened when no one under the age of 15 was living in a household with 2 or more adults. Except in this case, the question with invalid answers was the number of people under the age of 15. Despite the problems created from these errors, they were mended by manually filling the invalid entries with the appropriate values. Another question that had analogous but less systematic occurrences was related to the age of the building.

Table 3.3: Features from survey: heating information

Feature	Description
Heat_gas	Use of gas for heating (YES or NO answer)
Heat_oil	Use of oil for heating
Heat_solidfuel	Use of solid fuel for heating
Heat_timer	Use of timer to control heating
Water_heat_central	Use of central heating to heat water
Water_heat_electric	Use of electric heating to heat water
Water_heat_gas	Use of gas to heat water
Water_heat_oil	Use of gas to heat water
Water_heat_solidfuel	Use of solid fuel to heat water

Table 3.4: Features from survey: appliances information

Feature	Description
Tumble_dryer	Number of tumble dryers in the household
Dishwasher	Number of dishwashers in the household
Electric_shower_1	Number of electric showers 1 in the household
Electric_shower_2	Number of electric showers 2 in the household
Electric_cooker	Number of electric cookers in the household
Electric_heater	Number of electric heaters in the household
Standalone_freezer	Number of standalone freezers in the household
Water_pump	Number of water pumps in the household
Immersion_heater	Number of immersion heaters in the household
Tv_21_less	Number of TVs smaller than 21” in the household
Tv_21_greater	Number of TVs bigger than 21” in the household
Desktop_computer	Number of desktop computers in the household
Laptop_computer	Number of laptop computers in the household
Game_console	Number of gaming consoles in the household

For this question there is no correlation between the consumers who did not answer like in the previous situations. Therefore, the samples which presented an invalid entry were discarded. This in conjunction with the private consumer selection previously mentioned led to a final database of 4233 samples.

3.3.3 Temporal consumption indicators (Temporal CI)

The third dataset was a combination of the last two, it had indicators computed from raw data but displaying a temporal evolution as opposed to being static. This scheme used the principle behind the Static CI (developing indicators of electricity consumption instead of feeding raw information directly into the clustering algorithms) but still contained the temporal nature that is the focal point of this thesis. As a result, besides the date of each registry and the meter and attack identification, it had the temporal evolution of the same 4 indicators presented in the first database, I_1, I_2^e and I_2^c, I_3 and I_4^e and I_4^c. The recorded evolution spanned from 9 days prior to the selected day (day of the attack in the case of the non zero-day threat model) to the day itself. The dataset used the threat model with 6 attacks with both zero-day and non zero-day scenarios. Each of the 6 attacks were computed onto each meter in a set of 1000.

Chapter 4

Results and discussion

In this chapter, different tests were undertaken in order to evaluate the performance of the modelling algorithms described in chapter 2 on the different datasets developed. Only FCM was applied to Static Consumption Indicators since it does not exhibit the temporal nature of the other two sets. MFC was then tested using the remaining sets, Raw Data and Temporal Consumption Indicators, which possess the time-related characteristic for which the clustering algorithm was developed for. Likewise, FCM was tested with these to assess the impact of considering the entries of the temporal datasets as values for static features. In other words, discarding their time varying essence and treating them as static data.

Firstly, the evaluation metrics used throughout the analysis will be presented with a theoretical overview on AUC, Accuracy, True Positive Rate, Negative Rate and the specific indexes used for the assessment of FCM on Static Consumption Indicators. The discussion on the criteria is followed by the reasoning behind why certain tests and parameters were examined for the three frameworks and then the conclusions taken from their results.

4.1 Evaluation criteria

Similar tests were performed across the different databases and thus similar metrics were used to evaluate the results. With respect to the Static Consumption Indicators dataset, a more thorough evaluation was done on the application of FCM when compared to the implementation of the same method on the two other datasets. Because this was the only implementation of a clustering procedure on this dataset, it was decided to explore several assessment tools in order to determine their viability for subsequent tests. The simulations done on the three databases were evaluated using the following criteria: Area Under the Receiver Operating Characteristics Curve (AUC); Accuracy; True Positive Rate (TPR); False Positive Rate (FPR); Difference between TPR and FPR. More specifically, for the dataset of static indicators, the threshold determination method was studied using indexes such as Youden Index, minimum distance and testing every possible threshold until one yields the lowest difference between specificity and sensitivity.

4.1.1 Receiver operating characteristic

A common tool used for visualizing, organizing and selecting binary classifiers is the Receiver Operating Characteristic (ROC) graph, Fig. 4.1. In the ROC space, each classifier with a given class distribution and cost matrix is represented by a point (FP, TP) on the ROC curve. In addition to being a generally useful graphing method for comparison between classifiers, these curves have properties that make them especially useful for domains with class unbalance and unequal classification error costs. In a ROC curve, the true positive rate (TPR), also known as sensitivity or recall, is plotted against the false positive rate (FPR), calculated through $1 - specificity$, for different cut-off points of a parameter, as seen in Fig. 4.1.

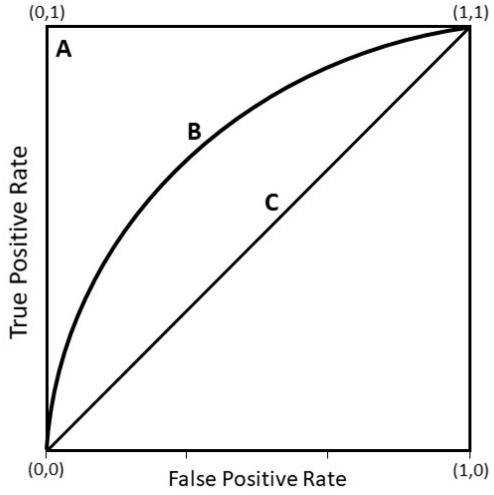


Figure 4.1

The ROC space can be divided in 3 general areas, the diagonal line, the upper triangle and the lower triangle. If the curve, like the one represented by the letter B in Fig. 4.1, overlaps with the diagonal line between points (0,0) and (1,1), the line C in Fig. 4.1 also known as chance line, it can be concluded that the classifier is no better than an algorithm that identifies the samples on a randomly basis. If the model's curve is nearer the point (0,1), point A in Fig. 4.1, than the diagonal than the model is better than random guessing. This point symbolizes a decision threshold that would correctly identify all positive ($TPR = 1$) and all negative ($FPR = 0$ or $specificity = 1$) samples. The closer the curve of the model gets to (0,1) the better the classifier. Lastly, if by contrast the curve is further away from the corner point than the diagonal line (the curve is located in the lower triangle shown in Fig. 4.1) the system is overall seen as having inadequate prediction power. Despite this, by inverting the classification outcome it is possible to mirror the respective curve into the upper portion of the ROC space, turning a once regarded inferior model into an acceptable one.

Each point on the ROC curve represents a sensitivity/specificity pair corresponding to a particular decision threshold. This results in a distinct confusion matrix, Table 4.1 for each of these points.

Table 4.1: Confusion Matrix

		Predicted	
		Positive	Negative
Actual	Positive	True Positive (TP)	False Negative (FN)
	Negative	False Positive (FP)	True Negative (TN)

It is through this table that metrics such as TPR, FPR and accuracy are calculated. All of these metrics vary between $[0, 1]$ and, as opposed to FPR, the closer they get to 1 the better.

True Positive Rate (TPR) or Sensitivity: True Positive Rate represents the percentage of positive cases correctly predicted as positive. In other words, the higher TPR, the fewer positive data points will be misclassified.

$$TPR = \frac{TP}{TP + FN} \tag{4.1}$$

False Positive Rate (FPR) or 1-Specificity: False Positive Rate represents percentage of negative cases incorrectly predicted (predicted as positive instead of negative). In other words, the higher FPR, the more negative data points will be misclassified.

$$FPR = \frac{FP}{FP + TN} \tag{4.2}$$

Accuracy: Accuracy represents the percentage of ”true” cases (both positive and negative) over all classified elements.

$$Accuracy = \frac{TN + TP}{TN + TP + FP + FN} \tag{4.3}$$

Despite evaluating the classifier on its ability to correctly identify positive and negatives classes within the test sample, accuracy does not take into account skewed class distribution. If a model is proficient at classifying the minority class but inadequate towards the majority group than accuracy will be low, since, objectively speaking, the system is misclassifying a high number of true cases. Another factor relevant to mention is the fact that accuracy does not consider the probability (be it 0.51 or 0.99) of the prediction, as long as the class with the largest probability estimation is the same as the target, it is regarded as correct. These points made it so a better criterion had to be used to estimate the predictive accuracy of classifiers and more precisely compare the different results gathered from the subsequent tests.

Area Under Curve (AUC)

Throughout the literature [24, 56, 34, 33, 57, 58], the area of the ROC graph or Area Under Curve (AUC) has been regarded as a well founded metric to reduce the whole ROC curve into a scalar measure making it an independent index on the decision threshold. This metric has an important statistical property: the AUC of a classifier is equivalent to the probability that the classifier will rank a randomly chosen positive instance higher than a randomly chosen negative instance [59]. The effectiveness of AUC at interpreting a model classification ability over other metrics such as accuracy has been studied and proven in the past [60, 61]. AUC is preferably used for comparison of models' performance [62] rather than for objective and standalone analysis. However, some have criticised the resounding support the adoption of this particular index has had throughout the years [63] but such criticism has also been counterpointed [64] leading to more conscious but still generalized uses of AUC.

Since this criterion is a representation of the whole curve, the values from AUC have a direct translation to different regions on the ROC space. As it is a unit graph (both axes range from 0 to 1 thus making its area 1), the value of AUC of a perfect model (the curve of a perfect model follows the y-axis from (0,0) to (0,1) and then ends at point (1,1)) is the same as the ROC space area, in other words, $AUC = 1$. The closer AUC gets to 1 the better the model. With the direct interpretation seen so far, it is easy to understand that $AUC = 0.5$ represents the random guessing model and any value of AUC around this is a sign of poor performance. As for values of $AUC < 0.5$, these represent models which have their classifications mixed and inverting them will give a new reversed AUC, AUC' ($AUC' = 1 - AUC$), resulting in $AUC' > 0.5$.

4.1.2 Tests on threshold determination

As mentioned above, to get the accuracy, TPR and FPR of a classifier a threshold needs to be defined. For the first tests developed in this project (analysis done to the Static CI dataset) a study on a few methodologies was done to understand their differences and determine which one fit best for this case study. The methods to determine the best threshold for the model relied on the ROC curve, more specifically, the Youden Index (Youden's J Statistic) [65] and the minimum distance to the point (0,1), Fig. 4.2. Beyond these, a more extensive method was used where every possible threshold was applied and the model with the best balance between accuracy and $TPR - FPR$ was then selected. This extensive framework (checking every threshold) will serve as baseline to compare the other two. Since this method takes longer, as it has to survey all possibilities, the purpose of it is to establish if any of the alternatives, Youden Index or minimum distance, can reach similar conclusions to the point of replacing checking all thresholds.

The Youden Index, as it was stated previously, is calculated using the following equation,

$$\begin{aligned} J &= max(\text{ Sensitivity} + \text{Specificity} - 1) \\ &= max(\text{ TPR} - \text{FPR})\,. \end{aligned} \tag{4.4}$$

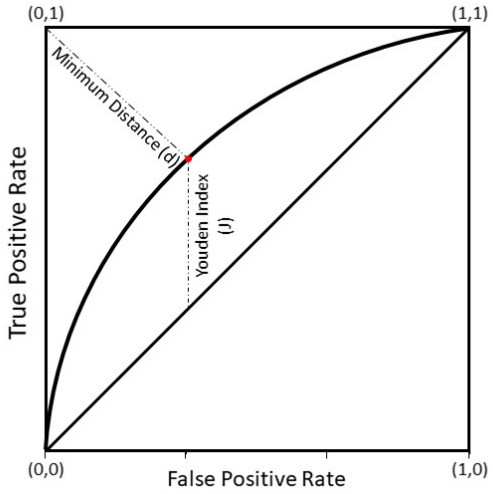


Figure 4.2: Graphical representation of the performance evaluation methods.

This index defines the maximum potential effectiveness of a particular threshold. The point in the ROC curve that maximizes this index is the cut-point that optimizes the threshold's differentiating ability when equal weight is given to sensitivity and specificity. Its value ranges from -1 ($AUC = 0$) to 1 ($AUC = 1$), and is zero when a diagnostic test gives the same proportion of positive results for groups with and without the data manipulation ($AUC = 0.5$). In this case the test is irrelevant.
The second method evaluates the distance, *d*, of the various ROC curve points to (0,1) and is defined by the following equation

$$d = \sqrt{(1 - \text{Sensitivity})^2 + (1 - \text{Specificity})^2} \tag{4.5}$$

Since the ideal model's ROC curve follows the lines that go from point (0,0) to (0,1) and finally to (1,1), in other words, $AUC = 1$, the closer a real model's curve gets to that the better. Therefore, it is relevant to look for the point in a ROC curve closest to (0,1) thus minimizing *d*.

4.2 Tests and parameters

4.2.1 Static Consumption Indicators

To the static consumption indicators only FCM was applied as a clustering method.

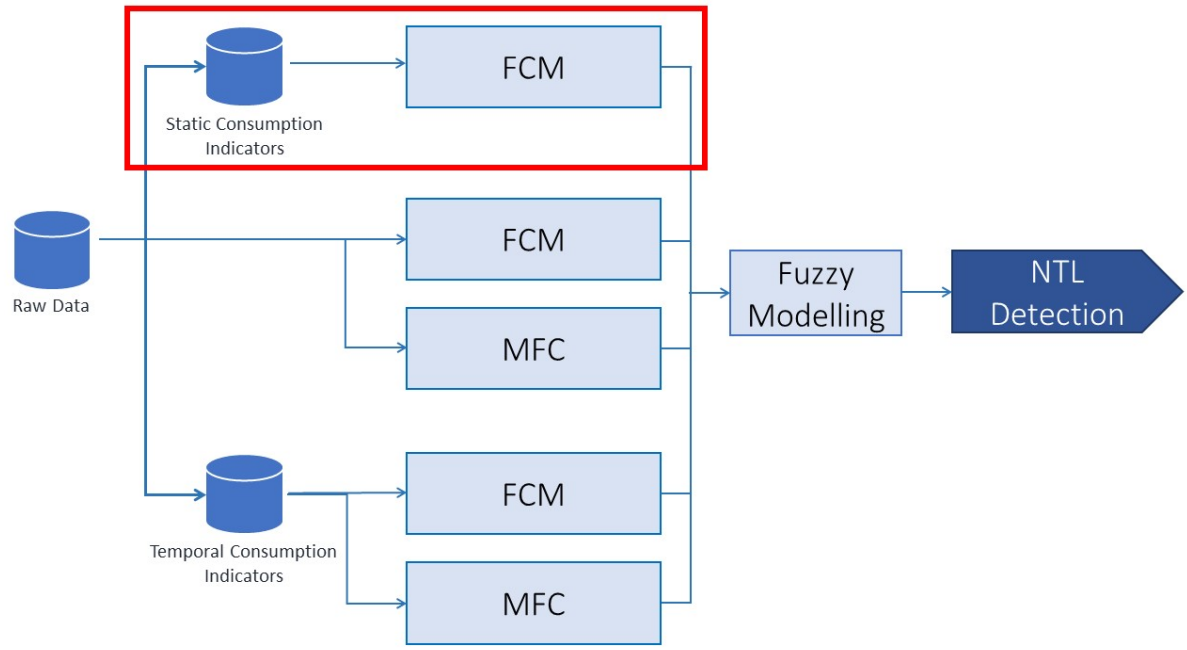


Figure 4.3: Graph of the methods applied to Static CI

There were four main parameters to be chosen in the tests done to this database: optimal number of clusters and fuzzification parameter, skewed class distribution, best method to deal with invalid entries in the dataset and lastly the effects of varying the cost associated with FP or FN.

With the intent of setting down the more specific details first, the parametrization of the classifiers (number of clusters, C, and fuzzy parameter, m) was done first through the use of a grid search. Given that the threat model presents 17 different samples (one element of non malicious behaviour plus 8 zero-day and 8 non zero-day attacks) the number of clusters was tested for $[2:19]$. On the off chance that the clustering algorithm could identify more than the 17 scenarios described, the variable C was set until 19 instead of 17. As for parameter m, it was tested for $[1.4:0.3:3.8]$.

During the data processing stage, the original database presented errors, null value or Not a Number (NaN), mostly for features I_c^2 and I_c^4, which had to be fixed before implementing the algorithms. Three methods, described in section 3.3.1, were designed to fix them and the best one for the task was to be found. As a reminder, the methods were: turn all values of NaN into 0; turn all NaN into the average value of the corresponding feature; turn only the NaN entries that characterize the attack models 5 and 50 into 0 and the remaining NaN into the average values. With the intention of simplifying the document these techniques will be referred to as Method 1, Method 2 and Method 3, respectively.

Another issue found during this early stage was the uneven ratio between regular/irregular electricity consumption patterns, for each data point of regular consumption, 16 attacks were computed onto them. This point in particular (skewed class balance) is common in machine learning and data mining research. As such, it became a point of interest to assess the influence of balancing techniques in the final result of the classifiers. The different ratios tested were built by copying the regular samples

several times until the ratio regular/irregular became what was needed. The original ratio (one sample of regular consumption to 16 samples of irregular) will also be referred in this thesis as the ratio of 5%. The different approaches studied reached values of 20% (4 regular samples to 16 attacks), 35% (8 regular samples to 16 attacks) and 47% (13 regular samples to 16 attacks).

These last two issues were combined during tests. This resulted in each specific method for dealing with NaN to have the four distributions implemented. As mentioned in the previous section, optimizing the determination of the threshold was also done with Static CI. To determine which approach (Youden Index, minimum distance or testing all possible thresholds) would result in the best classifier, all three were applied to each pair of Method/Balance and the accuracy, TPR and FPR were analysed and compared.

As an additional perspective, in appendix A a study on the relative costs of FP and FN was done as an alternative to find the optimal threshold for metric such as accuracy, FPR and TPR. In the field of theft and fraud, the costs of accusing and investigating incorrect cases are significantly high to the point where companies prioritize a high certainty in the detection of positive cases even if it means not looking into a few uncertain ones that would be positive. With this in mind, evaluating how would varying the costs fare in the determination of more appropriate thresholds for the case study.

4.2.2 Time variant and invariant features, raw data

The temporal component characteristic of the consumption patterns influenced the need to use a clustering method that took that into account. As a result, MFC was used. However, this dataset was also computed so that clustering methods for static features could also be used such as FCM. This was done to assess the end result of discarding the temporal component of the consumption patterns and whether that would result in better performing classifiers.

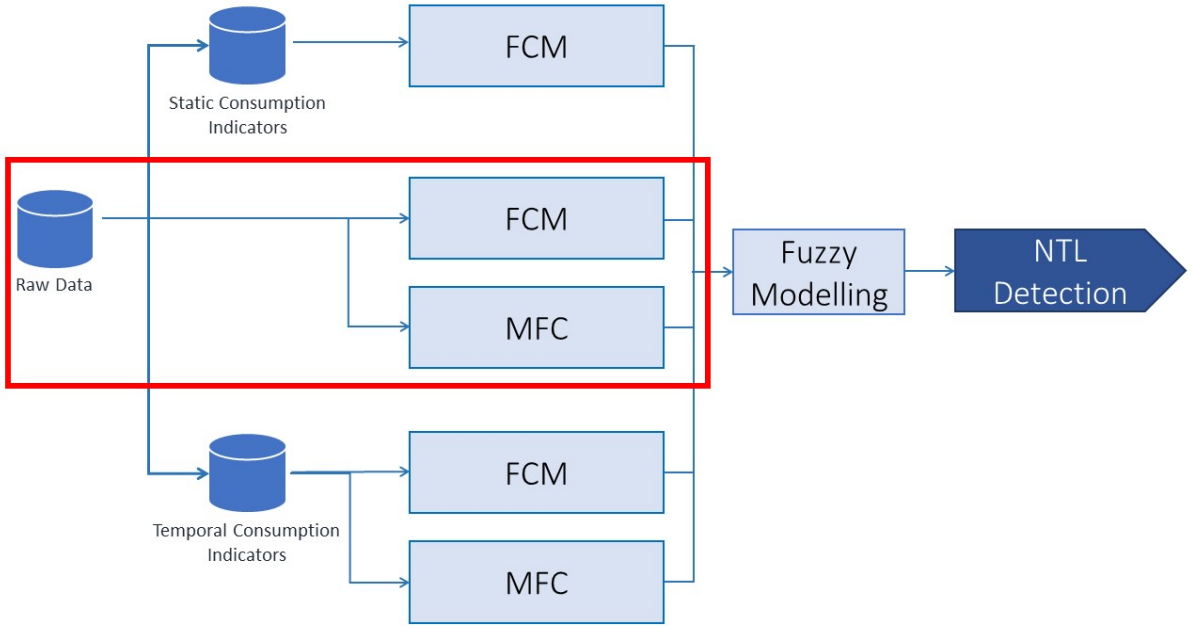


Figure 4.4: Graph of the methods applied to raw data

Despite the changes made to the original database, which were explained in section 3.3.2, contributing to a smaller sample size, it was still too large to train models over the several parameters chosen for the following tests. Not only there were different frameworks for the testing datasets (for example balancing ratios and a change in the threat model) but the chosen parameters (number of clusters, fuzzification parameter and feature discrimination parameter) were also tested for wide ranges. This resulted in extensive simulations that would only get their computing time increased the larger the number of samples in the datasets. The number of meters and, as a consequence, the number of samples, influenced the computing time the most. If made too small could hinder the clustering and classification process as a result. Two sets of meters were then chosen, with 200 and 1000, and they will be referred throughout this document as 200M and 1000M, respectively. The objective of this thesis is to provide a solution to the very real problem of identification of electricity theft. As such, the real life scenario presents an ever increasing amount of information for existing classification programs to sort through and analyse, meaning, the 1000 meter datasets more closely represents real life cases. Nonetheless, the hypothesis of overloading classification algorithms with information can be tested by comparing results from models computed from 1000M with ones built from 200M.

Experimental analysis of fuzzy model parameters

The first set of tests consisted in evaluating the influence of the fuzzification parameter and the feature discrimination parameter in the final predictive accuracy of the model. For that, the number of clusters used during clustering and classification did not vary and remained equal to 2. While for the set of 200M a grid search was set up for clustering and classification individually (for each set of clustering

parameters a grid search was performed for the modelling process), when it came to the 1000 meter dataset the parameters for classification were kept constant, $C = 2$ and $m = 2$. This was done because the computation time would increase considerably if the same methods were used. In addition to that, the results from the 200 meter showed that the range of performances on the final models could potentially be kept while maintaining the values of the classification model the same. These initial studies also focused on evaluating the modelling process when solely applying what is in theory the simplest attack vector to identify, h_5, and the zero-day correspondent h_{50}. This is believed to be true since these two particular cases change the fluctuating electricity consumption patterns to the daily average. The results from these two attacks methods were compared to the dataset where all attacks were computed to both 200M and 1000M. When applying grid search, the fuzzification parameter and the feature weight discriminant were examined for the values of $[1.1 : 0.3 : 3.5]$ and $[2, 10, 50, 100]$, respectively. The choice for the values of m was influenced by the paper [66] where researchers claim that tests to determine the effects of m should be done for values within the range $[2, 3.5]$ where, in theory, the optimal value will lie within $[2.5, 3]$. Keeping these findings in mind and understanding that electric fraud case studies were not in the scope of this study, these tests were done with lower values of the fuzzification parameter, namely $m = [1.1 : 0.3 : 2]$, to see if they would generate accurate models. The feature discriminant variable was tested in advance for lower values, $q < 10$, and it as shown that these yielded worse scores than higher ones. For that reason and the fact that higher values were showing faster simulations, contributed to the use of high values for this variable.

Balance data in preprocessing

These next tests addressed the mathematical models' abilities to identify illegal behaviour when presented with datasets containing only zero-day attacks, non zero-day attacks and both situations. For these experiments data balancing techniques were also analysed. These techniques consisted in replicating the sample of regular electricity consumption a certain amount of times until a specific balance between regular and irregular samples was achieved. However, this method carries an inherent problem, overfitting. If the same sample was repeated too many times, the mathematical model created from clustering the training data becomes biased towards those specific regular samples and unable to classify different regular ones as non theft. The end result was three arrays of tests for the dataset of 200M where the first had no balancing technique applied, the second had the non-malicious sample replicated twice (leading to ratios of 3 regular samples to 8 attack vectors in zero-day and non zero-day tables and 3 regular samples to 16 attack vectors in the joint dataset) and the last one had the regular sample replicated five times. When it came to the 1000 meter dataset, instead of doing grid search like in the previous experiments, the best models using the 200M were picked and their performances computed for this larger sample size. Only the unbalanced and the first balancing method were carried out for 1000M. In all these tests, neither the 200 nor the 1000M sets had grid search performed during the modelling process. The clustering and modelling parameters were set to the same values. With

the information gathered from the previous tests it was clear that setting both number of clusters and fuzziness parameter to 2 during the modelling step would yield suboptimal results. As a consequence, this new methodology of using the same values for clustering and modelling was implemented to still see if good results could be obtained while trying to accelerate the simulation time. For the tests with only zero-day or non zero-day attacks the number of clusters was tested for $[2:4, 7:10]$. The data bases were formatted to only discern between regular and irregular consumption, the attack labels only display 0 or 1 (two different clusters), and thus C is tested for values between 2 and 4 (there might be a few attack vectors that might display very distinct patterns so $C = [3, 4]$ was also tested). However, these datasets contain 8 different attack vectors plus the regular sample and, for that reason, higher number of clusters had to be examined hence the $C = [7:10]$. As for the dataset with all attacks vectors, the number of clusters was tested for $[2:4, 7:10, 15:19]$, higher number of attack vectors, higher number of clusters to be tested. As for the fuzziness parameter, m and the feature discriminant exponent, q, these were tested for $[1.1:0.3:3.8]$ and $[10, 50, 100, 500]$, respectively. The two balance ratios used were equivalent to the ones used in Static CI.

Threat model experiments

Afterwards, the predictive power of models using a slightly different attack vector was evaluated on three different array of samples, zero-day attacks, non zero-day attacks and both approaches together. Thus far, the threat model used in this project had followed the work of [24] but for these tests, as a means of comparison, an identical attack vector to the one presented in [22] was used with the addition of the zero-days scenario. In [22], the authors formulated a threat model composed of 6 distinct theft methodologies (the same first 6 non zero-day attacks used in [24] and, as a result, in this project so far). Researchers in [24] took that 6 vector threat model and added 2 new attacks while also implementing the zero-day scheme.

In addition to changing the threat model for these experiments, a balancing procedure was also analysed. It consisted in randomly choosing one attack per meter resulting in a balance of 1 regular consumption sample to 1 irregular sample. This balance framework was used only for the set of 1000M since it is the dataset that more closely depicts the situation in current electric companies (high volume of clients) and the resulting table, with the balance implemented, does not overload the classification algorithm requiring an impractical amount of time to run completely. For the unbalanced set of tests, a grid search was applied to the set of 200M for the clustering process, using the same values for the modelling process, and the best classifiers found were applied to the set of 1000M to examine if they maintained their capability in a more populated environment. To ascertain these particular models' relative performance in this new set, a grid search was then applied to 1000M to check the difference between the best overall models and the best models simulated with 200 meters. In the balanced scenario a grid search was only applied to the 1000M data. For the datasets with zero-day and non zero-day attacks separated the number of clusters was examined for $[2:4, 6:8]$ while for the table

with both attacks structures it runs through $[2:4, 6:8, 12:14]$. The fuzziness parameter is tested for $[2:0.3:3.8]$ and the feature weight for $[10, 50, 100, 500, 1000]$.

Practical case scenario

Up until this point, all 18 time series (3 distinct time windows per season of the year during the 6 registered seasons) had been used as input to the algorithm so as to provide enough context and information to what constituted each attack vector. Since it is the consumption patterns that change with each malicious procedure, not the demographic data, it was thought to be fundamental to include as many examples of each approach as possible without overloading the program. However, this does not exactly translate the current reality in the electric industry. Utility companies, once they had spotted a consumer who might have been illegally altering their consumption patterns, would only have data from that particular incident and not from other seasons of the year. It is being assumed that these consumers have made new contracts with the respective company as it has been a concurrent occurrence in several proven cases. For the next tests a different framework was used to replicate the information electric companies tend to gather for non-technical losses inspections. As such, instead of 18 time series, only one was randomly picked for each attack done to a every single meter. Since there are 17 individual samples per meter (8 zero-day and 8 non zero-day attacks and the non malicious sample) and 18 time series, each sample ended up having a different time window associated with it. Similarly to the last test, not only were the two threat models compared but also the approach to 200M and 1000M was the same (best models found in 200M were tested for 1000M and compared with the best models overall in the larger set). This was done for an unbalanced set (one non malicious sample to 8 or 6 attack vectors). A simple set of tests were performed using a balancing scheme to see if, like before, it would improve results. The regular consumption sample was replicated so that the layouts zero-day, non zero-day and both together had a ratio of 75/25 (irregular/regular samples). This was projected to be done firstly for the 200M set and see if the prediction accuracy improved before moving onto the larger set of meters. As for parameters, for the 8 attack threat model, C was tested for $[2:4, 8:10]$ and $[2:4, 8:10, 16:18]$, depending on whether it is for zero and non zero-day attacks or both at the same time. For the other threat model, C was tested for $[2:4, 6:8]$ and $[2:4, 6:8, 12:14]$. In either case both m and q were tested for $[2:0.3:3.8]$ and $[10, 50, 100, 500]$ respectively. The value $q = 1000$ was removed because it had been used previously to prove that higher q were yielding better performing models but it is a very high value to be used for this variable according to [32].

Modelling using FCM from raw datasets

To understand the effects of using raw data as opposed to static consumption indicators, it was decided to use FCM to create classifying models and compare the results with what was done with Static CI.

The unsatisfactory results from past tests using MFC also influenced the desire to implement Fuzzy C-Means. However, applying FCM to these datasets meant that the temporal component present in the

data would be lost. Each entry of the time series would be regarded by the clustering algorithm as a value for another time invariant feature. Using the threat model with 8 attacks and one randomly chosen time series per sample for 200M and 1000M, tests were run for $C = [2 : 9]$ and $C = [2 : 9, 12 : 14]$ (depending on the attack dataset) and $m = [1.4 : 0.3 : 3.8]$. These two parameters had a broader scrutiny due to the less computationally intensive method used, FCM instead of MFC.

4.2.3 Study of temporal consumption indicators

Similarly to the datasets in the previous section, Temporal CI also contain a temporal component making it eligible for MFC. And as a way to compare with the first dataset, SCI, and evaluating the impact of the temporal component in this particular dataset, FCM was also applied.

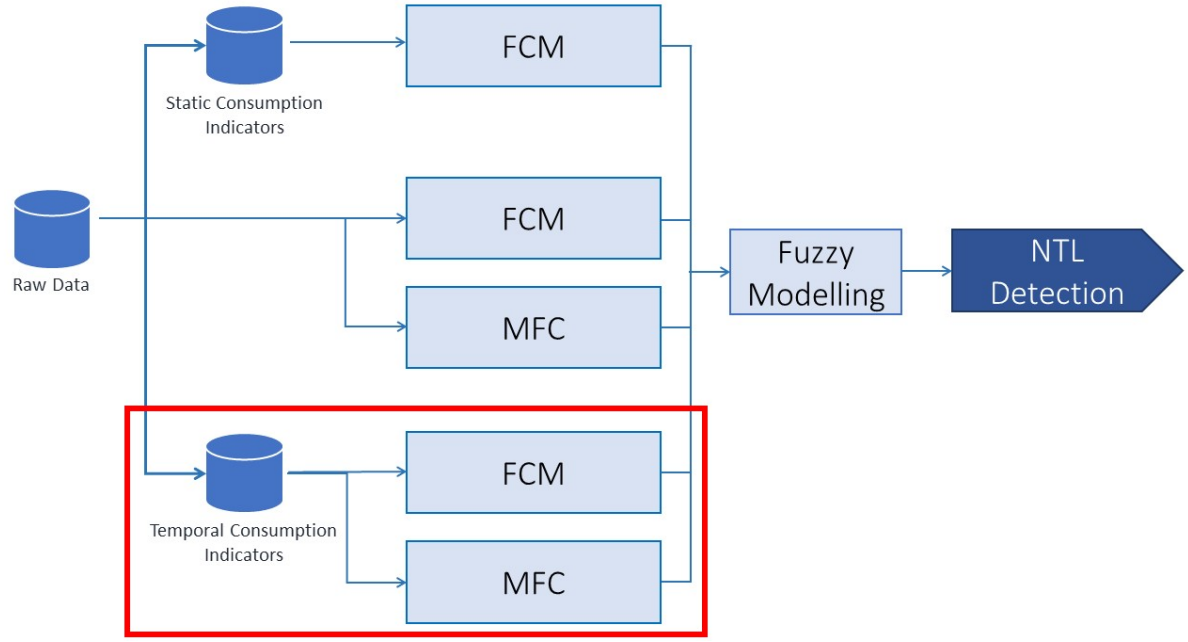


Figure 4.5: Graph of the methods applied to raw data

MFC with the Temporal Consumption Indicators

Applying the database directly to the MFC algorithm, a grid search was performed where $C = [2 : 4, 6 : 8]$ and $C = [2 : 4, 6 : 8, 12 : 14]$ (depending on the attack dataset), $m = [1.4 : 0.3 : 3.8]$ and $q = [3, 10, 50, 100, 500]$. The use of temporal indicators, despite also possessing a time variant nature, proved to accelerate the simulation time when compared to using raw data directly. Because of this and the fact that most of the best models were found to have $q = 3$, a new set of tests were done where q was set up for lower values, $[1.3, 1.8, 2.5, 5, 8]$. This test was thought to be relevant since it was the first time that low values of q were observed to compute the best performing models.

FCM with the Temporal Consumption Indicators

To serve as a base for comparison not only with the last test but also with the rest experiments with FCM, Temporal CI were also used to create FCM FM. This particular experiment is relevant since it will provide a means of comparison between static and temporal indicators and, overall, if in fact there is an advantage in keeping the time variant nature of certain features. Continuing with the 1000M set, C was tested for $[2:9]$ and $C = [2:9, 12:14]$ and m for $[1.4:0.3:3.8]$.

4.3 Results

In the following sections, the main results from the previously presented tests are shown and discussed. Here only the best models and their evaluation values are presented so as to keep the document brief and concise. When relevant, more extensive tables will be referred to the corresponding appendixes. In tables where clustering and/or modelling parameters are shown, they will be displayed with the format [C m q] and [C m], respectively.

4.3.1 Static consumption indicators

The tests done on the Static CI dataset are presented in the following order: analysis on cluster number and fuzzy parameter, balancing techniques and NaN solution and study on the relative costs of false classifications, shown in appendix A.

Modelling using FCM from Static Consumption Indicators

The grid search performed for C and m was applied onto a dataset with no balancing method implemented and with all the errors turned to the average. Table 4.2 presents the results with the models sorted by their AUC value. The parameter column displays [C m] as the number of clusters and fuzzy constant, respectively, considered for the specific model. The first five rows show the five worst models and the remaining show the best (skipped the rest in order to not overload the document). The grid search done to both parameters resulted in a total of 162 models and from the worst to the best the AUC only varied 0,04. Given that the range of values for m and especially C was rather extensive, the impact they had in the overall performance of the classifiers was inconsequential.

In terms of the parameters themselves, it is possible to see a clear predisposition for m where 1.4 and 1.7 yield the best classifiers. As for C, there does not seem be any correlation between either small, medium or high values and good performance. Although high numbers, such as $C = [16, 17, 18]$, seem to create better performing models, these also directly correlate to an increase in computation time so higher performance has to be balanced with practicality.

Taking only these conclusions into consideration, the set of variables to use in further evaluations would be $C = 9$ and $m = 1.4$ since this is the system with the lowest number of clusters in the group of

best classifiers (balancing performance and computation time). However, this was done to a particular database, one with Method 2 and no balancing procedure applied. To safely use these values on all following tests, similar conclusions on the impact of C and m have to be drawn from different datasets. To this end, another grid search was set up on other data points for these variables. This time C and m were tested for $[8 : 11]$ and $[1.4 : 0.3 : 3.8]$ respectively. The former results were taken into account prompting a smaller range for cluster number but the same for fuzzification. These variables were tested on a dataset with 36% of benign data where Method 3 was applied to deal with NaN. So as to validate previous claims, analogous trends have to be observed, AUC needs to take similar values overall and vary from higher to lower numbers as m increases.

By looking at Table 4.3 it is possible to confirm the previously registered tendency with AUC. Not only do lower values of m create better performing classifiers, but C also continues to show no clear correlation with AUC values. This metric shows slightly higher numbers overall, but it is still extremely close to the previous run. Once again, the pair $C = 9$ and $m = 1.4$ is amongst the models with highest AUC. This confirmation makes it so the use of these numbers for these variables on following test is now justified.

Table 4.2: Table with results from tests done to $C = [2 : 19]$ and $m = [1.4 : 0.3 : 3.8]$ when applying FCM to the dataset Static CI with Method 2 applied and 5% ratio

Parameters	AUC	Accuracy	TPR	FPR	TPR-FPR
[4 3,8]	0,741 $\pm$ 0	0,680 $\pm$ 0	0,679 $\pm$ 0	0,298 $\pm$ 0	0,380
[9 3,8]	0,742 $\pm$ 0	0,677 $\pm$ 0	0,675 $\pm$ 0	0,295 $\pm$ 0	0,381
[11 3,8]	0,742 $\pm$ 0	0,677 $\pm$ 0	0,675 $\pm$ 0	0,293 $\pm$ 0	0,382
[7 3,8]	0,743 $\pm$ 0	0,677 $\pm$ 0	0,675 $\pm$ 0	0,292 $\pm$ 0	0,383
[7 3,5]	0,743 $\pm$ 0	0,682 $\pm$ 0	0,681 $\pm$ 0	0,298 $\pm$ 0	0,382
$\vdots$					
[10 1,4]	0,777 $\pm$ 0	0,700 $\pm$ 0	0,697 $\pm$ 0	0,250 $\pm$ 0	0,448
[9 1,4]	**0,778 $\pm$ 0**	**0,710 $\pm$ 0**	**0,708 $\pm$ 0**	**0,259 $\pm$ 0**	**0,449**
[19 1,7]	0,778 $\pm$ 0	0,712 $\pm$ 0	0,711 $\pm$ 0	0,265 $\pm$ 0	0,446
[12 1,7]	0,778 $\pm$ 0	0,705 $\pm$ 0	0,702 $\pm$ 0	0,248 $\pm$ 0	0,453
[17 1,7]	0,778 $\pm$ 0	0,704 $\pm$ 0	0,702 $\pm$ 0	0,255 $\pm$ 0	0,447
[17 1,4]	0,780 $\pm$ 0	0,696 $\pm$ 0	0,693 $\pm$ 0	0,250 $\pm$ 0	0,443

As for the examination of balancing and error fixing methods, the results are shown in Table 4.4. Overall, there is not a substantial difference in AUC between the three methods for fixing NaN. From the best to the worst models, the difference does not reach 0.05. Method 2 computed the worst performing classifiers out of the three, while the remaining two do not show great difference aside from when they are paired with an unbalanced set (dataset with 5% of benign data). And even in this setting (Method 1 with 5% regular samples and Method 2 with 5%) the difference in AUC and $TPR - FPR$ are just 0.02

Table 4.3: Table with results from tests done to $C = [8 : 11]$ and $m = [1.4 : 0.3 : 3.8]$ when applying FCM to the dataset Static CI with Method 3 applied and 36% ratio

Parameters	AUC	Accuracy	TPR	FPR	TPR-FPR
[10 3,8]	0,766 $\pm$ 0	0,704 $\pm$ 0	0,693 $\pm$ 0	0,276 $\pm$ 0	0,416
[11 3,8]	0,767 $\pm$ 0	0,707 $\pm$ 0	0,702 $\pm$ 0	0,283 $\pm$ 0	0,419
[11 3,2]	0,768 $\pm$ 0	0,709 $\pm$ 0	0,707 $\pm$ 0	0,287 $\pm$ 0	0,420
[9 3,8]	0,768 $\pm$ 0	0,706 $\pm$ 0	0,697 $\pm$ 0	0,278 $\pm$ 0	0,419
$\vdots$					
[11 1,7]	0,783 $\pm$ 0	0,727 $\pm$ 0	0,712 $\pm$ 0	0,247 $\pm$ 0	0,465
[9 1,4]	**0,786 $\pm$ 0**	**0,725 $\pm$ 0**	**0,722 $\pm$ 0**	**0,268 $\pm$ 0**	**0,454**
[10 1,7]	0,787 $\pm$ 0	0,729 $\pm$ 0	0,716 $\pm$ 0	0,247 $\pm$ 0	0,469
[8 1,7]	0,788 $\pm$ 0	0,728 $\pm$ 0	0,713 $\pm$ 0	0,245 $\pm$ 0	0,468
[9 1,7]	0,788 $\pm$ 0	0,729 $\pm$ 0	0,718 $\pm$ 0	0,251 $\pm$ 0	0,467
[11 1,4]	0,799 $\pm$ 0	0,737 $\pm$ 0	0,720 $\pm$ 0	0,235 $\pm$ 0	0,486

and 0.03, respectively.

When comparing the balancing frameworks, both the original dataset and the one with 36% benign data consistently yielded the best results across the three NaN methods. Given that there is an inherent cost to revise the collected data points to create a more even set (the additional cost being an increase in computational power since the dataset is larger), working with information as is (no redistribution needed) seems a fitting choice for this specific database.

As for the methods to find the optimal threshold, no method stands out. To evaluate these, the metrics accuracy, but primarily $TPR - FPR$, will be used. The latter exhibits no significant difference amongst the three despite one of the methods being based on maximizing this metric. The biggest difference in $TPR - FPR$ for a single pair Method NaN / Balance was 0,014. Because all the three points (one for each threshold approach) belong to the same ROC curve, the similarity in $TPR - FPR$ means that the points are considerably close to each other. This can be confirmed in Fig. 4.6 and Fig. 4.7. With respect to accuracy, the strategy of checking all thresholds consistently lead to more accurate models contrasting with the Youden Index which computed the less accurate ones out of the three procedures. It can be concluded that focusing on maximizing $TPR - FPR$, which is what Youden index does, may compromise other indicators. It is however worth noting that the difference in accuracy is not substantial and the maximum registered was 0.025.

With all of this said, the best combination of parameters is Method 1 applied to a database with no redistribution procedure implemented and with the threshold being computed through the use of minimum distance. Reiterating what has previously been said, using no balance strategy helps in reducing the cost of simulations and, despite the extensive search of testing every threshold yielding more accurate points (by a small margin) than minimum distance, the added cost (increased computational time)

is not worth it. As a result, in the following tests, the result tables show classifiers picked using minimum distance.

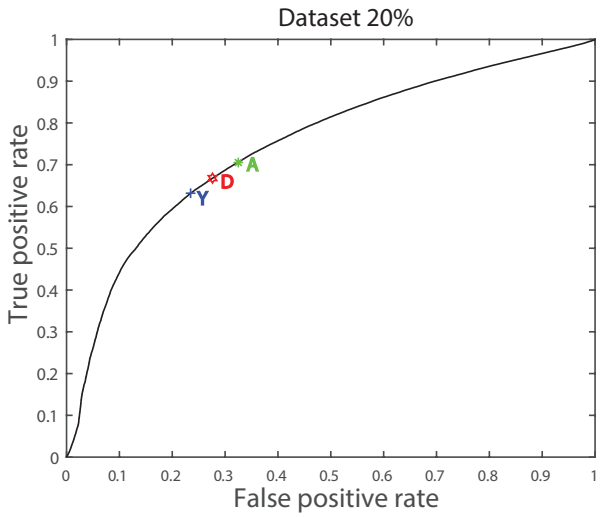


Figure 4.6: ROC curve for dataset with a 20% distribution and Method 2 applied

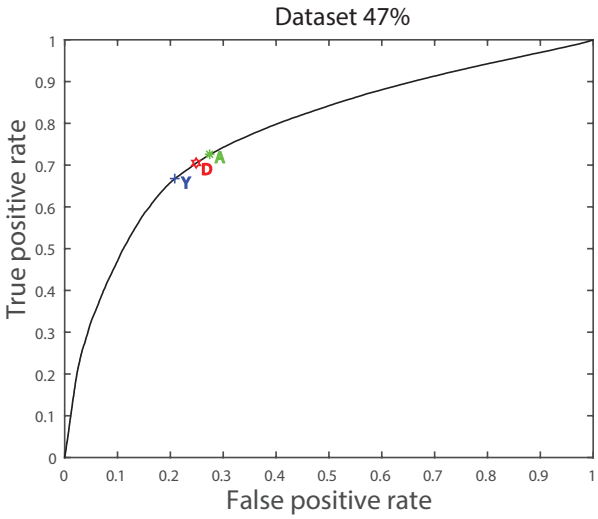


Figure 4.7: ROC curve for dataset with a 47% distribution and Method 3 applied

4.3.2 Time variant and invariant features

Experimental analysis of fuzzy model parameters, $C = 2$

Having the number of clusters remain the same, allowed for a more thorough investigation on the parameter estimation for clustering and modelling

Looking at the two different approaches regarding the modelling parameters it would be reasonable clear as to which produced better results were it not for the *All Attacks* set up. For *Attack 5 / Attack 50* and *Attack 50* models tend to perform better and a grid search was applied to the modelling process, as opposed to keeping said variables constant. However, for the remaining attack framework the difference was negligible.

The overall results gathered from this initial assessment, Table 4.5, outright revealed a poor predictive precision from the algorithm. While the 200m set confirmed that models created using solely Attack 50 had the best performance, in 1000M the same could not be said. As it was explained previously, this attack in particular replaced the daily consumption points with ones equal to the average of that day from the moment the meter started registering, zero-day scenario. In theory, this was the most distinguishable attack and so, the mathematical model would have been able to discern more easily regular and irregular behaviour, which did not happen. Even in the best situation the results were mediocre. Looking at the overall outcome of 200M and 1000M, the former built better performing models.

From the table 4.5 it can also be seen that MFC was able to compute better performing models using the set up with *Attack 50* on the 200M dataset than with the other two frameworks on either 200M and 1000M. In theory, this attack specifically produces the most distinguishable pattern out of the 8 attacks,

Table 4.4: Table with results from tests done to $C = [8 : 11]$ and $m = [1.4 : 0.3 : 3.8]$ when applying FCM to the dataset of Static CI

Method NaN	**Balance**	**Method Threshold**	**AUC**	**Accuracy**	**TPR**	**FPR**	**TPR-FPR**
Method 1	**5%**	**Minimum Distance**	**0,790 ± 0**	**0,732 ± 0**	**0,731 ± 0**	**0,263 ± 0**	**0,469**
		Youden Index	0,790 ± 0	0,722 ± 0	0,720 ± 0	0,250 ± 0	0,470
		All Thresholds	0,790 ± 0	0,747 ± 0	0,749 ± 0	0,286 ± 0	0,464
	20%	Minimum Distance	0,780 ± 0	0,723 ± 0	0,724 ± 0	0,279 ± 0	0,445
		Youden Index	0,780 ± 0	0,715 ± 0	0,710 ± 0	0,264 ± 0	0,446
		All Thresholds	0,780 ± 0	0,724 ± 0	0,725 ± 0	0,280 ± 0	0,445
	36%	Minimum Distance	0,789 ± 0	0,731 ± 0	0,724 ± 0	0,258 ± 0	0,466
		Youden Index	0,789 ± 0	0,727 ± 0	0,706 ± 0	0,236 ± 0	0,469
		All Thresholds	0,789 ± 0	0,734 ± 0	0,749 ± 0	0,291 ± 0	0,458
	47%	Minimum Distance	0,776 ± 0	0,716 ± 0	0,682 ± 0	0,244 ± 0	0,438
		Youden Index	0,776 ± 0	0,715 ± 0	0,657 ± 0	0,219 ± 0	0,438
		All Thresholds	0,776 ± 0	0,715 ± 0	0,713 ± 0	0,282 ± 0	0,431
Method 2	5%	Minimum Distance	0,772 ± 0	0,689 ± 0	0,686 ± 0	0,260 ± 0	0,425
		Youden Index	0,772 ± 0	0,650 ± 0	0,641 ± 0	0,213 ± 0	0,428
		All Thresholds	0,772 ± 0	0,719 ± 0	0,721 ± 0	0,304 ± 0	0,416
	20%	Minimum Distance	0,746 ± 0	0,683 ± 0	0,677 ± 0	0,293 ± 0	0,384
		Youden Index	0,746 ± 0	0,663 ± 0	0,642 ± 0	0,252 ± 0	0,390
		All Thresholds	0,746 ± 0	0,696 ± 0	0,701 ± 0	0,325 ± 0	0,376
	36%	Minimum Distance	0,766 ± 0	0,705 ± 0	0,680 ± 0	0,251 ± 0	0,429
		Youden Index	0,766 ± 0	0,697 ± 0	0,648 ± 0	0,215 ± 0	0,433
		All Thresholds	0,766 ± 0	0,710 ± 0	0,704 ± 0	0,279 ± 0	0,425
	47%	Minimum Distance	0,753 ± 0	0,698 ± 0	0,659 ± 0	0,258 ± 0	0,401
		Youden Index	0,753 ± 0	0,696 ± 0	0,615 ± 0	0,211 ± 0	0,404
		All Thresholds	0,753 ± 0	0,698 ± 0	0,684 ± 0	0,287 ± 0	0,397
Method 3	5%	Minimum Distance	0,771 ± 0	0,717 ± 0	0,717 ± 0	0,281 ± 0	0,435
		Youden Index	0,771 ± 0	0,709 ± 0	0,708 ± 0	0,272 ± 0	0,437
		All Thresholds	0,771 ± 0	0,711 ± 0	0,710 ± 0	0,273 ± 0	0,437
	20%	Minimum Distance	0,783 ± 0	0,715 ± 0	0,706 ± 0	0,250 ± 0	0,457
		Youden Index	0,783 ± 0	0,710 ± 0	0,698 ± 0	0,241 ± 0	0,457
		All Thresholds	0,783 ± 0	0,724 ± 0	0,723 ± 0	0,269 ± 0	0,454
	36%	Minimum Distance	0,786 ± 0	0,723 ± 0	0,713 ± 0	0,258 ± 0	0,455
		Youden Index	0,786 ± 0	0,718 ± 0	0,696 ± 0	0,243 ± 0	0,453
		All Thresholds	0,786 ± 0	0,726 ± 0	0,723 ± 0	0,269 ± 0	0,455
	47%	Minimum Distance	0,783 ± 0	0,728 ± 0	0,709 ± 0	0,250 ± 0	0,459
		Youden Index	0,783 ± 0	0,727 ± 0	0,671 ± 0	0,208 ± 0	0,463
		All Thresholds	0,783 ± 0	0,726 ± 0	0,726 ± 0	0,274 ± 0	0,452

changing the normally irregular profile into a constant equal to the daily average. This might explain the increased performance when compared to *Attack 5 / Attack 50* or *All Attacks*. Not only did the set up with *Attack 50* on the 200M yield models with higher AUC than than the remaining configurations but it also had the highest $TPR - FPR$. However, $AUC = 0,644$ was not a good result. Using FCM on Static CI yielded AUC values of $0,79$.

Table 4.5: Table with results for $C = 2$ when applying MFC to the raw dataset

Dataset	Attacks	Parameters		AUC	Accuracy	TPR	FPR	TPR-FPR
		Clustering	Modelling					
200M	Attack 5 / Attack 50	[2 2,3 50]	[2 2,9]	0,568 ± 0,004	0,483 ± 0,000	0,342 ± 0,000	0,233 ± 0,000	0,109
		[2 2,3 100]	[2 2,6]	0,568 ± 0,003	0,494 ± 0,000	0,392 ± 0,000	0,300 ± 0,000	0,092
		[2 2,9 50]	[2 2,6]	0,570 ± 0,001	0,522 ± 0,000	0,442 ± 0,000	0,317 ± 0,000	0,125
	Attack 50	[2 3,2 100]	[2 2,6]	0,641 ± 0,000	0,600 ± 0,000	0,467 ± 0,000	0,267 ± 0,000	0,200
		[2 1,4 100]	[2 1,7]	0,642 ± 0,001	0,600 ± 0,000	0,450 ± 0,000	0,250 ± 0,000	0,200
		[2 2,6 100]	[2 2,3]	0,644 ± 0,002	0,600 ± 0,000	0,450 ± 0,000	0,250 ± 0,000	0,200
	All attacks	[2 1,4 100]	[2 1,4]	0,587 ± 0,002	0,617 ± 0,000	0,626 ± 0,000	0,533 ± 0,000	0,093
		[2 3,2 100]	[2 2,3]	0,587 ± 0,002	0,627 ± 0,000	0,637 ± 0,000	0,533 ± 0,000	0,103
		[2 2,6 100]	**[2 2]**	**0,588 ± 0,000**	**0,627 ± 0,000**	**0,637 ± 0,000**	**0,533 ± 0,000**	**0,103**
1000M	Attack 5 / Attack 50	[2 1,7 50]	[2 2]	0,527 ± 0,000	0,496 ± 0,000	0,427 ± 0,000	0,367 ± 0,000	0,060
		[2 2,6 10]	[2 2]	0,528 ± 0,000	0,511 ± 0,000	0,495 ± 0,000	0,457 ± 0,000	0,038
		[2 1,4 100]	[2 2]	0,529 ± 0,000	0,479 ± 0,000	0,380 ± 0,000	0,324 ± 0,000	0,056
	Attack 50	[2 1,4 10]	[2 2]	0,548 ± 0,000	0,550 ± 0,000	0,413 ± 0,000	0,313 ± 0,000	0,100
		[2 1,4 100]	[2 2]	0,564 ± 0,000	0,560 ± 0,000	0,373 ± 0,000	0,253 ± 0,000	0,120
		[2 1,4 50]	[2 2]	0,564 ± 0,000	0,560 ± 0,000	0,373 ± 0,000	0,253 ± 0,000	0,120
	All attacks	[2 2,6 50]	[2 2]	0,582 ± 0,000	0,550 ± 0,000	0,600 ± 0,000	0,477 ± 0,000	0,123
		[2 2,9 100]	[2 2]	0,587 ± 0,000	0,560 ± 0,000	0,593 ± 0,000	0,470 ± 0,000	0,123
		[2 2,9 50]	**[2 2]**	**0,589 ± 0,000**	**0,560 ± 0,000**	**0,596 ± 0,000**	**0,470 ± 0,000**	**0,126**

Balance data in preprocessing

As explained in section 4.2.2, these test evaluate the approaches taken to data balancing. The Table 4.6 gathers the results from the best classifiers when using the 200 meters dataset and also the results of these same models when applied to the 1000M set. To contextualize these last results, the best models when using the larger set had to be computed, Table 4.7.

The results from the scenario with 200 meters showed an improvement in model performance when using balanced datasets. The more balanced the better the results. However, this time, the difference was consistent enough to exclude the possibility of it being caused by fluctuations from running the algorithm several times. A similar increasing trend was registered when using the set with 1000 meters except the increments were more noticeable.

The distinction used for both attack types proved to result in interesting outcomes. As expected, models based solely on *Zero-Day* attacks had the best results which clearly contrasted with the ones from *Non Zero-Day* attacks. *All Attacks*, as it is an assembly of both cases, has in between results. It is worth noting that for the ratio of 1 regular sample to 16 attacks on the 1000M set, *Non Zero-Day*

models showed very low Accuracy. Accuracy evaluates how well a system identifies ”true” cases (True Positives and True Negatives) and does not take into account skewed class distribution. In other words, if a model is proficient at classifying the minority class but inadequate towards the majority group than the Accuracy will be low, since objectively speaking the system is misclassifying a lot of true cases. The same thing does not happen with AUC. Being unable to identify either class will yield a bad score with this metric regardless. The low Accuracy values for the mentioned scenario is indicative that these specific models do not identify the majority class well, theft.

Table 4.6: Results of the best models using a 200M dataset when different balancing techniques are tested. also applied to 1000M

Dataset	Ratio	Attacks	Parameters		AUC	Accuracy	TPR	FPR	TPR-FPR
			Clustering	Modelling					
200M	1 regular / 16 attacks	Zero-Day	[10 2,3 1000]	[10 2,3]	0,682 ± 0,018	0,611 ± 0,089	0,605 ± 0,119	0,342 ± 0,153	0,264
		Non Zero-Day	[4 3,5 50]	[4 3,5]	0,519 ± 0,008	0,398 ± 0,029	0,365 ± 0,032	0,333 ± 0,000	0,031
		All Attacks	**[10 3,2 100]**	**[10 3,2]**	**0,592 ± 0,004**	**0,846 ± 0,001**	**0,889 ± 0,002**	**0,833 ± 0,000**	**0,055**
	3 regular / 16 attacks	Zero-Day	[9 2,9 500]	[9 2,9]	0,699 ± 0,007	0,549 ± 0,096	0,467 ± 0,203	0,233 ± 0,189	0,233
		Non Zero-Day	[2 2,3 50]	[2 2,3]	0,525 ± 0,033	0,609 ± 0,070	0,713 ± 0,380	0,667 ± 0,220	0,046
		All Attacks	**[8 3,8 1000]**	**[8 3,8]**	**0,598 ± 0,018**	**0,621 ± 0,068**	**0,639 ± 0,113**	**0,475 ± 0,177**	**0,164**
	6 regular / 16 attacks	Zero-Day	[8 3,8 100]	[8 3,8]	0,708 ± 0,012	0,645 ± 0,126	0,590 ± 0,323	0,267 ± 0,189	0,323
		Non Zero-Day	[3 3,2 500]	[3 3,2]	0,528 ± 0,014	0,497 ± 0,026	0,433 ± 0,370	0,400 ± 0,190	0,033
		All Attacks	**[16 2 50]**	**[16 2]**	**0,585 ± 0,014**	**0,717 ± 0,094**	**0,973 ± 0,108**	**0,967 ± 0,118**	**0,006**
1000M	1 regular / 16 attacks	Zero-Day	[9 3,5 500]	[9 3,5]	0,601 ± 0,022	0,579 ± 0,059	0,583 ± 0,110	0,457 ± 0,280	0,127
		Non Zero-Day	[4 3,8 10]	[4 3,8]	0,500 ± 0,002	0,116 ± 0,078	0,007 ± 0,002	0,007 ± 0,001	0,000
		All Attacks	**[18 3,5 100]**	**[18 3,5]**	**0,550 ± 0,014**	**0,648 ± 0,044**	**0,661 ± 0,350**	**0,560 ± 0,140**	**0,101**
	3 regular / 16 attacks	Zero-Day	[9 2,9 500]	[9 2,9]	0,674 ± 0,009	0,654 ± 0,008	0,687 ± 0,002	0,433 ± 0,094	0,253
		Non Zero-Day	[2 2,3 50]	[2 2,3]	0,512 ± 0,012	0,549 ± 0,079	0,596 ± 0,007	0,577 ± 0,022	0,020
		All Attacks	**[8 3,8 1000]**	**[8 3,8]**	**0,585 ± 0,026**	**0,667 ± 0,058**	**0,810 ± 0,005**	**0,717 ± 0,014**	**0,094**

Table 4.7: Results from the best models when using the 1000M dataset when different balancing techniques are tested

Dataset	Ratio	Attacks	Parameters		AUC	Accuracy	TPR	FPR	TPR-FPR
			Clustering	Modelling					
1000M	1 regular / 16 attacks	Zero-Day	[9 2,9 100]	[9 2,9]	0,655 ± 0,044	0,719 ± 0,014	0,754 ± 0,130	0,560 ± 0,140	0,194
		Non Zero-Day	[10 2 50]	[10 2]	0,512 ± 0,025	0,298 ± 0,012	0,237 ± 0,010	0,217 ± 0,240	0,020
		All Attacks	**[7 2,6 100]**	**[7 2,6]**	**0,588 ± 0,019**	**0,595 ± 0,030**	**0,599 ± 0,230**	**0,470 ± 0,070**	**0,129**
	3 regular / 16 attacks	Zero-Day	[9 1,4 50]	[9 1,4]	0,692 ± 0,050	0,683 ± 0,008	0,740 ± 0,106	0,433 ± 0,066	0,306
		Non Zero-Day	[4 3,5 100]	[4 3,5]	0,543 ± 0,022	0,502 ± 0,032	0,458 ± 0,040	0,383 ± 0,210	0,075
		All Attacks	**[7 3,5 500]**	**[7 3,5]**	**0,620 ± 0,024**	**0,660 ± 0,003**	**0,739 ± 0,100**	**0,550 ± 0,150**	**0,189**

Threat model experiments

In order to understand if introducing less information into the system would yield better results, the threat model proposed in [22] was used to compute the several malicious patterns for the databases. Analysing the AUC values from the best models for each situation, presented in Table 4.8, it is possible to observe that results did not actually improve. However, the scenario with one attack per meter, *Balance 50/50*, generated relatively better results for at least one scenario, *Zero-Day*.

In general, *Non Zero-Day* continued to yield the worst models as opposed to *Zero-Day*, which was expected. After all, one has the attacks start at the beginning of each time series, making the difference with regular consumption patterns clearer, while the other is composed of changes done only at the end of each time series. The Table 4.9 with the best models for 1000M showed worse results than their counterpart 200M. This difference could be explained by the fact that adding more meters may in fact add more variability making it harder for the algorithm to distinguish individual attack vectors. However, this time, the difference was consistent enough to make sense discarding the possibility of being fluctuations from running the algorithm several times.

For the *Unbalanced* framework, models created using the dataset with solely *Non Zero-Day* attacks showed very poor accuracy. In practical terms this means that the models clearly could not distinguish between fraudulent and non fraudulent consumers and ended up misclassifying both. This helps understand the reason for the low AUC values as this is an overall evaluation index and the models from *Non Zero-Day* are ineffective classifiers. The difference between using *Zero-Day* or *All Attacks* was clear using 200 meters but vanishes as more data is added to the program (with the use of 1000 meters).

Analysing the same models being tested in different datasets proved that each situation had to be, most likely, dealt separately. Looking at both tables, one could see that the best models in each framework are not the same and that the best models found using the 200 dataset had worse results in the 1000M dataset. This change was more noticeable in all metrics aside AUC.

Comparing unbalanced and the 50/50 balance made for the 1000 consumers, the results corroborate the theoretical improvement of the results. Performances improved across the board, with significant change to models using *Zero-Day* attack vectors. Accuracy greatly increased for the *Non Zero-Day* dataset and although the same can be said for TPR on the same data points, *Zero-Day* and *All Attacks* registered significant deterioration. Knowing that TPR evaluates the ability of a model to accurately identify positive cases and that AUC increased for all scenarios, the decrease in TPR proves that the models became more precise in classifying negative values. This conclusion can be corroborated by observing the decrease in FPR.

Practical case scenario

The overall results depicted in Table 4.10 show very poor performances in every scenario. Looking at tests done previously the decline in predictive power when limiting the amount of information used in this case consumption patterns is very clear. It is plausible to conclude that the algorithm becomes less adept at distinguishing a particular attack from another with a similar approach when the number of occurrences that depict the change each attack (the different time series) is lowered.

At first glance, one could assume that the overall predictive precision of the several classification systems are all practically the same. However, looking closely at accuracy and the other criteria it is possible to see a very clear distinction between the two attack vectors. From what was registered, the threat model with 8 attacks frequently presented lower values for this index. The fact that both

Table 4.8: Table with the best models from 200M applied to the 1000M dataset computed using a new threat model

Ratio	Dataset	Attacks	Parameters		AUC	Accuracy	TPR	FPR	TPR-FPR
			Clustering	Modelling					
Unbalanced	200M	Zero Day	[8 2,9 500]	[8 2,9]	0,726 ± 0,017	0,712 ± 0,069	0,765 ± 0,124	0,450 ± 0,094	0,315
			[7 3,8 500]	[7 3,8]	0,731 ± 0,006	0,558 ± 0,071	0,472 ± 0,094	0,183 ± 0,000	0,289
		Non Zero Day	[2 2 50]	[2 2]	0,526 ± 0,009	0,460 ± 0,012	0,392 ± 0,016	0,334 ± 0,000	0,058
			[6 3,5 500]	[6 3,5]	0,528 ± 0,029	0,583 ± 0,083	0,642 ± 0,145	0,592 ± 0,106	0,050
		All attacks	[7 2,9 500]	[7 2,9]	0,627 ± 0,005	0,686 ± 0,024	0,775 ± 0,035	0,633 ± 0,035	0,142
			[6 2 1000]	**[6 2]**	**0,628 ± 0,003**	**0,601 ± 0,123**	**0,616 ± 0,189**	**0,475 ± 0,224**	**0,141**
	1000M	Zero Day	[8 2,9 500]	[8 2,9]	0,611 ± 0,019	0,634 ± 0,038	0,660 ± 0,054	0,522 ± 0,059	0,138
			[7 3,8 500]	[7 3,8]	0,617 ± 0,012	0,621 ± 0,076	0,637 ± 0,103	0,475 ± 0,083	0,162
		Non Zero Day	[2 2 50]	[2 2]	0,506 ± 0,000	0,226 ± 0,000	0,114 ± 0,000	0,100 ± 0,000	0,014
			[6 3,5 500]	[6 3,5]	0,501 ± 0,002	0,654 ± 0,103	0,714 ± 0,145	0,708 ± 0,153	0,006
		All attacks	[7 2,9 500]	[7 2,9]	0,588 ± 0,022	0,608 ± 0,044	0,615 ± 0,033	0,477 ±0,035	0,139
			[6 2 1000]	**[6 2]**	**0,604 ± 0,039**	**0,681 ± 0,042**	**0,700 ± 0,035**	**0,557 ± 0,022**	**0,144**
Balance 50/50	1000M	Zero Day	[4 3,8 100]	[4 3,8]	0,719 ± 0,014	0,587 ± 0,087	0,289 ± 0,253	0,113 ± 0,128	0,176
			[4 3,2 1000]	[4 3,2]	0,723 ± 0,027	0,604 ± 0,018	0,357 ± 0,082	0,148 ± 0,047	0,208
		Non Zero Day	[2 2 500]	[2 2]	0,526 ± 0,001	0,531 ± 0,002	0,437 ± 0,003	0,404 ± 0,000	0,033
			[4 3,8 100]	[4 3,8]	0,526 ± 0,007	0,527 ± 0,013	0,675 ± 0,030	0,623 ± 0,024	0,051
		All attacks	[4 3,2 1000]	[4 3,2]	0,624 ± 0,034	0,585 ± 0,029	0,517 ± 0,042	0,402 ± 0,036	0,115
			[4 3,5 1000]	**[4 3,5]**	**0,625 ± 0,038**	**0,577 ± 0,021**	**0,420 ± 0,042**	**0,267 ± 0,030**	**0,153**

Table 4.9: Table with the best models from the 1000M dataset computed using a new threat model

Ratio	Dataset	Attacks	Parameters		AUC	Accuracy	TPR	FPR	TPR-FPR
			Clustering	Modelling					
Unbalanced	1000M	Zero Day	[2 2 1000]	[2 2]	0,644 ± 0,000	0,698 ± 0,000	0,737 ± 0,000	0,537 ± 0,000	0,201
			[8 2 500]	[8 2]	0,664 ± 0,004	0,637 ± 0,063	0,646 ± 0,091	0,420 ±0,104	0,226
		Non Zero Day	[2 2,3 50]	[2 2,3]	0,519 ± 0,005	0,205 ± 0,010	0,081 ± 0,012	0,050 ± 0,000	0,031
			[4 2,3 50]	[4 2,3]	0,521 ± 0,016	0,523 ± 0,042	0,525 ± 0,063	0,492 ± 0,083	0,033
		All attacks	[6 3,8 100]	[6 3,8]	0,609 ± 0,020	0,753 ± 0,036	0,785 ± 0,021	0,623 ± 0,037	0,161
			[14 3,2 50]	**[14 3,2]**	**0,611 ± 0,004**	**0,611 ± 0,011**	**0,845 ± 0,010**	**0,720 ± 0,019**	**0,125**

attack vectors show similar AUC but different accuracy values can be interpreted as a problem with class unbalance, a known problem in this project. The accuracy measure does not take into account this problematic while AUC does. As such, if the models are capable of identifying the majority class but completely inadequate in identifying the other one, their accuracy will be high, while their AUC will not. This can be confirmed by observing TPR. High TPR values are common to systems capable of identifying positive cases which are the majority class.

Using the best models found with 200 meters on the 1000M dataset once again proved each models' precision varies depending on the set used but comparing the best models for each showed 1000M to have better perfomring classifiers.

Table 4.10: Table with the best models from 200M applied to the 1000M dataset using only one time series

Dataset	Attacks	Threat Model	Parameters		AUC	Accuracy	TPR	FPR	TPR-FPR
			Clustering	Modelling					
200M	Zero-Day	6 Attacks	[3 2 100]	[3 2]	0,558 ± 0,015	0,607 ± 0,069	0,628 ± 0,100	0,517 ± 0,079	0,111
		8 Attacks	[4 3,5 50]	[4 3,5]	0,535 ± 0,024	0,298 ± 0,074	0,233 ± 0,090	0,178 ± 0,097	0,055
	Non Zero-Day	6 Attacks	[6 2 500]	[6 2]	0,519 ± 0,001	0,457 ± 0,037	0,433 ± 0,220	0,400 ± 0,132	0,033
		8 Attacks	[9 3,8 50]	[9 3,8]	0,511 ± 0,000	0,124 ± 0,000	0,015 ± 0,000	0,000 ± 0,000	0,015
	All attacks	**6 Attacks**	**[3 2,3 50]**	**[3 2,3]**	**0,529 ± 0,013**	**0,731 ± 0,037**	**0,769 ± 0,046**	**0,717 ± 0,050**	**0,052**
		8 Attacks	[4 3,2 100]	[4 3,2]	0,527 ± 0,015	0,720 ± 0,112	0,748 ± 0,150	0,733 ± 0,114	0,015
1000M	Zero-Day	6 Attacks	[3 2 100]	[3 2]	0,522 ± 0,039	0,506 ± 0,065	0,502 ± 0,200	0,470 ± 0,210	0,032
		8 Attacks	[4 3,5 50]	[4 3,5]	0,520 ± 0,021	0,476 ± 0,064	0,463 ± 0,200	0,423 ± 0,202	0,040
	Non Zero-Day	6 Attacks	[6 2 500]	[6 2]	0,503 ± 0,001	0,810 ± 0,081	0,934 ± 0,170	0,934 ± 0,180	0,001
		8 Attacks	[9 3,8 50]	[9 3,8]	0,500 ± 0,000	0,111 ± 0,000	0,000 ± 0,000	0,000 ± 0,000	0,000
	All attacks	6 Attacks	[3 2,3 50]	[3 2,3]	0,522 ± 0,002	0,410 ± 0,179	0,390 ± 0,211	0,350 ± 0,200	0,040
		8 Attacks	**[4 3,2 100]**	**[4 3,2]**	**0,523 ± 0,033**	**0,650 ± 0,110**	**0,667 ± 0,140**	**0,620 ± 0,153**	**0,047**

Table 4.11: Table with the best models 1000M using only one time series

Dataset	Attacks	Threat Model	Parameters		AUC	Accuracy	TPR	FPR	TPR-FPR
			Clustering	Modelling					
1000M	Zero-Day	6 Attacks	[8 2,9 10]	[8 2,9]	0,548 ± 0,007	0,854 ± 0,068	0,999 ± 0,210	0,960 ± 0,183	0,039
		8 Attacks	[3 3,5 500]	[3 3,5]	0,541 ± 0,030	0,733 ± 0,005	0,792 ± 0,190	0,740 ± 0,181	0,052
	Non Zero-Day	6 Attacks	[6 2,6 50]	[6 2,6]	0,511 ± 0,025	0,370 ± 0,009	0,313 ± 0,130	0,290 ± 0,117	0,023
		8 Attacks	[9 2,3 10]	[9 2,3]	0,505 ± 0,011	0,144 ± 0,102	0,043 ± 0,010	0,039 ± 0,008	0,004
	All attacks	6 Attacks	[7 2,9 500]	[7 2,9]	0,530 ± 0,008	0,568 ± 0,034	0,575 ± 0,040	0,518 ± 0,027	0,057
		8 Attacks	[8 2,6 500]	[8 2,6]	0,555 ± 0,009	0,421 ± 0,105	0,403 ± 0,180	0,293 ± 0,124	0,110

Modelling using FCM from raw datasets

Since it was clear that the best models on a specific dataset would not translate into the best ones on a different one, this approach was dropped. Instead the best models on 200M and 1000M were directly compared, assuming they would not be the same which proved to be true.

Going back to the tests done on Static CI, using FCM on raw data yielded similar results. In comparison with MFC, FCM produced much better classifying models, as seen in Table 4.12.

While MFC would repeatedly create worse models when using higher volume of data points, this did not happen when using FCM. With 1000M, models showed better values across all criteria, having the biggest difference being registered in AUC. Another point of contrast between the two methodologies is the performance improvement of the *Non Zero-Day* scenario with respect to the other two. Unlike before, this scenario proved to yield accurate classifiers using both set of meters (the best results out of the three scenarios for 200M). Not only did it have better AUC but the remaining metrics were also higher.

For the scenario of *All Attacks* on 200M, multiple models were registered to have the same values for all evaluation indexes. Analysing the output of these models it is clear that every single one had the same classification output, meaning, they identified the multiple samples with the same label. No matter the values for the number of clusters or fuzzification parameter most of their combination produced models that behaved the same way to the input information, something that is hard to explain and differs from what has been seen so far in this thesis.

Table 4.12: Table with the best models when applying FCM to raw data of 200 and 1000 meters

Dataset	Attacks	Parameters	AUC	Accuracy	TPR	FPR	TPR-FPR
200M	Zero Day	[6 1,7]	0,771 ± 0,032	0,778 ± 0,015	0,800 ± 0,016	0,356 ± 0,043	0,444
		[9 1,7]	0,795 ± 0,006	0,785 ± 0,019	0,805 ± 0,023	0,331 ± 0,019	0,473
	Non Zero Day	[9 3,8]	0,803 ± 0,000	0,786 ± 0,000	0,814 ± 0,000	0,383 ± 0,000	0,431
		[9 1,7]	0,804 ± 0,008	0,758 ± 0,014	0,769 ± 0,017	0,308 ± 0,010	0,460
	All attacks	[14 3,5]	0,747 ± 0,000	0,760 ± 0,000	0,765 ± 0,000	0,300 ± 0,000	0,465
		[14 3,8]	**0,747 ± 0,000**	**0,760 ± 0,000**	**0,765 ± 0,000**	**0,300 ± 0,000**	**0,465**
1000M	Zero Day	[8 1,7]	0,861 ± 0,032	0,799 ± 0,018	0,805 ± 0,018	0,235 ± 0,020	0,570
		[7 1,7]	0,871 ± 0,002	0,798 ± 0,009	0,801 ± 0,010	0,219 ± 0,007	0,582
	Non Zero Day	[6 1,7]	0,863 ± 0,003	0,806 ± 0,007	0,814 ± 0,009	0,238 ± 0,003	0,576
		[9 1,7]	0,863 ± 0,001	0,806 ± 0,004	0,813 ± 0,005	0,236 ± 0,005	0,577
	All attacks	[7 1,7]	0,842 ± 0,013	0,792 ± 0,004	0,796 ± 0,005	0,258 ± 0,012	0,538
		[13 1,7]	**0,844 ± 0,006**	**0,791 ± 0,003**	**0,794 ± 0,005**	**0,256 ± 0,023**	**0,539**

4.3.3 Temporal consumption indicators

MFC with the Temporal Consumption Indicators

Applying MFC to temporal indicators proved to yield better results than using time variant and invariant features, raw data.

The first tests used the usual ranges for the parameters C, m and q but it was seen that lower values of q were computing more reliable models than higher ones. Since this had not happened in previous results, in fact the exact opposite was the norm for almost every scenario run, it was decided to use a narrower range around smaller values of q. In Table 4.13, the results showed that models had peaked with the values of q already tested and that smaller ones did not yield better models. In fact, for the

All Attacks the best ones found were worse than in the previous test, though this might have been an outlier as there is no reasonable explanation to justify this deviation.

It is interesting to note that, despite having very bad AUC with both tests, *Non Zero-Day* attacks produced models with the best accuracy by far. Looking at the output of these models it becomes clear that they simply classified every single sample as ”fraudulent” which lead to 16 of the 17 samples per meter being correctly classified. This is another proof that accuracy cannot be solely used to evaluate the performance of a classifier.

Table 4.13: Table with the best models when applying MFC to Temporal CI to a database of 1000 meters and varying the range of parameters m and q

Range of m & q	Attacks	Parameters		AUC	Accuracy	TPR	FPR	TPR-FPR
		Clustering	Modelling					
Same	Zero Day	[7 1,4 3]	[7 1,4]	0,643 ± 0,001	0,645 ± 0,008	0,674 ± 0,012	0,532 ± 0,012	0,143
		[7 2 3]	[7 2]	0,646 ± 0,003	0,546 ± 0,013	0,523 ± 0,022	0,317 ± 0,038	0,206
	Non Zero Day	[8 3,8 100]	[8 3,8]	0,500 ± 0,000	0,857 ± 0,000	1,000 ± 0,000	1,000 ± 0,000	0,000
		[8 3,8 500]	[8 3,8]	0,500 ± 0,000	0,857 ± 0,000	1,000 ± 0,000	1,000 ± 0,000	0,000
	All attacks	[13 3,5 3]	[13 3,5]	0,679 ± 0,000	0,594 ± 0,009	0,587 ± 0,010	0,330 ± 0,014	0,257
		[13 3,8 3]	**[13 3,8]**	**0,680 ± 0,003**	**0,611 ± 0,033**	**0,608 ± 0,041**	**0,357 ± 0,066**	**0,251**
Focused	Zero Day	[8 1,4 2,5]	[8 1,4]	0,648 ± 0,002	0,634 ± 0,002	0,656 ± 0,004	0,502 ± 0,007	0,155
		[7 1,4 2,5]	[7 1,4]	0,650 ± 0,002	0,567 ± 0,035	0,553 ± 0,003	0,347 ± 0,057	0,206
	Non Zero Day	[8 3,8 5]	[8 3,8]	0,500 ± 0,000	0,857 ± 0,000	1,000 ± 0,000	1,000 ± 0,000	0,000
		[8 3,8 8]	[8 3,8]	0,500 ± 0,000	0,857 ± 0,000	1,000 ± 0,000	1,000 ± 0,000	0,000
	All attacks	[7 3,2 5]	[7 3,2]	0,579 ± 0,001	0,724 ± 0,004	0,760 ± 0,005	0,705 ± 0,007	0,055
		[6 3,5 2,5]	**[6 3,5]**	**0,581 ± 0,000**	**0,367 ± 0,000**	**0,332 ± 0,000**	**0,207 ± 0,000**	**0,126**

FCM with the Temporal Consumption Indicators

With the intent of being able to compare the three data types used in this work, FCM had to also be applied to Temporal CI.

Unlike what happened when this algorithm was applied to time variant and invariant features, the results in Table 4.14 did not show good performances across all scenarios. In fact, *Non Zero-Day* attacks had the same classification scores as the one from the previous experiment when MFC was used. As a consequence, *All Attacks* evaluation values worsened as it combines *Zero-Day* attacks which computed well performing models and *Non Zero-Day* attacks which computed very bad ones.

Table 4.14: Table with the best models when applying FCM to Temporal CI to a database of 1000 meters

Attacks	Parameters	AUC	Accuracy	TPR	FPR	TPR-FPR
Zero Day	[2 3,2]	0,868 ± 0,000	0,803 ± 0,000	0,799 ± 0,000	0,170 ± 0,000	0,629
	[2 3,5]	0,869 ± 0,000	0,804 ± 0,000	0,800 ± 0,000	0,170 ± 0,000	0,630
	[2 3,8]	0,869 ± 0,000	0,804 ± 0,000	0,799 ± 0,000	0,170 ± 0,000	0,629
Non Zero Day	[9 3,2]	0,500 ± 0,000	0,857 ± 0,000	1,000 ± 0,000	1,000 ± 0,000	0,000
	[9 3,5]	0,500 ± 0,000	0,857 ± 0,000	1,000 ± 0,000	1,000 ± 0,000	0,000
	[9 3,8]	0,500 ± 0,000	0,857 ± 0,000	1,000 ± 0,000	1,000 ± 0,000	0,000
All attacks	[2 2,9]	0,695 ± 0,000	0,695 ± 0,000	0,648 ± 0,000	0,367 ± 0,000	0,281
	[2 3,2]	0,696 ± 0,000	0,696 ± 0,000	0,648 ± 0,000	0,367 ± 0,000	0,281
	[2 3,5]	**0,696 ± 0,000**	**0,647 ± 0,000**	**0,648 ± 0,000**	**0,367 ± 0,000**	**0,281**

4.4 Overall results

Table 4.15 presents a summary of the results gathered from the tests made throughout the thesis. In each dataset, the *All Attacks* threat model is highlighted as it represents all possible attack vectors. Since there is no prior knowledge as to how a fraudulent consumer might interfere with their consumption patterns, *All Attacks* allows for a more inclusive and intricate threat model that represents all the forms of meter manipulation studied in this thesis. These reasons make this threat model advantageous to utility companies as they could introduce any household data and not worry if certain techniques were missing as this attack vector would include all.

From this global perspective, the difference in performances becomes clear when using MFC or FCM to cluster data points before applying Takagi-Sugeno modelling. Aside from some outliers, using Fuzzy C-Means, even in data containing a temporal component, results in more proficient classifying models. It is then plausible to conclude that maintaining the temporal nature of time variant features at the clustering stage does not directly translate into more accurate models.

For the most part, *Non Zero-Day* attacks showed worse results than the other scenarios. This can be attributed to the fact that the changes in the power consumption only alter the last day of the record. This fact makes it harder for clustering algorithms to identify certain samples as theft and discern them from normal consumption. Despite the good results gathered from *Zero-Day* attacks, the ones from *Non Zero-Day* made it so *All Attacks* frequently computed an average performing set of classifiers since this set of attacks combines the previous two scenarios.

Table 4.15: Summarised table

Dataset	Model	Parameters		AUC	TPR	FPR	TPR-FPR
SCI	FCM FM	Method 1 No Balance		**0,790**	**0,731**	**0,263**	**0,469**
Raw Data	MFC FM	Zero-Day	18 Time Series	0,725	0,590	0,267	0,323
			1 Time Series	0,558	0,628	0,517	0,111
		Non Zero-Day	18 Time Series	0,543	0,458	0,383	0,075
			1 Time Series	0,519	0,433	0,400	0,033
		All Attacks	**18 Time Series**	**0,620**	**0,739**	**0,550**	**0,189**
			1 Time Series	0,555	0,403	0,293	0,110
	FCM FM	Zero-Day	1 Time Series	0,871	0,800	0,219	0,581
		Non zero-day	1 Time Series	0,863	0,813	0,236	0,577
		All	**1 Time Series**	**0,844**	**0,798**	**0,233**	**0,565**
TCI	MFC FM	Zero-Day		0,646	0,523	0,317	0,206
		Non Zero-Day		0,500	1,000	1,000	0,000
		All Attacks		**0,680**	**0,608**	**0,357**	**0,251**
	FCM FM	Zero-Day		0,867	0,834	0,240	0,594
		Non Zero-Day		0,500	1,000	1,000	0,000
		All Attacks		**0,692**	**0,653**	**0,373**	**0,280**

4.5 Comparison to previous works

Despite the high values of AUC that were achieved in certain set ups during this thesis, it is necessary to put the work done into perspective by analysing the results from researchers who applied different tools.

The authors of [22], developed a consumption pattern-based energy theft detector by applying SVM to a synthetic attack dataset. This approach resulted in very accurate models, leading others to follow suit. The threat model seen in [56] was based off of the previously mentioned work. The modifications made (adding two new attack vectors and differentiating between zero-day and non zero-day approaches) made this threat model ideal to be used in this thesis. In addition to the revision of the threat model, the authors tested on the same dataset as the Static CI of this thesis a variety of methods, including FCM and SVM, being Gustafson-Kessel (GK) fuzzy clustering algorithm the one which showed the best results. In [53] the authors extended their previous research [19] by integrating human knowledge and expertise into the SVM-based fraud detection model with the implementation of a FIS.

A seen from the table 4.16, there are clear differences in the evaluation metrics and data used throughout these works. However, the results found in [56] can be compared to the work done in this thesis since both used the same dataset from ISSDA and applied the same metrics to evaluate the resulting classifiers. As for [22] despite having a similar input data (although it was also from ISSDA, the authors only used the consumption patterns and not the demographic information), the metrics used were different, only coinciding with FPR, and the threat model was much smaller leading to different balancing problems. Lastly,[19] not only used a completely different dataset but also only one matching metric, hit-rate or TPR.

Table 4.16: Comparison between the algorithms used in the thesis and the ones used in the literature

Reference	Method	Data Type	AUC	TPR	FPR	Accuracy	DR
Thesis	FCM	Raw data	0,844	0,798	0,233	0,790	-
		SCI	0,790	0,731	0,263	0,732	-
Viegas [56]	GK	SCI	0,752	0,643	0,241	-	-
Jokar [22]	SVM	Temporal features	-	-	0,110	-	0,940
Nagi [53]	SVM-FIS	Feature extraction	-	0,720	-	-	-

The table shows that as far as AUC is concerned, the model computed from FCM on the raw dataset was the most accurate and with the highest hit-rate. The Detection Rate (DR) used by [22] is calculated following the formula

$$DR = \frac{TP}{TN + TP + FP + FN} \quad . \tag{4.6}$$

Looking at that equation it is possible to conclude that one model will necessarily have higher accuracy than DR since the numerator of the equation that characterizes the former also contains the element TN. Since [22] used a smaller threat model, resulting in less attacks or samples from class 1 (as opposed to samples of regular consumption, class 0), it is not correct to immediately assume that SVM yielded better performing models then any of the algorithms used in this thesis. Nonetheless, the absolute value of DR indicates a very good classifier for the input data that was used.

Chapter 5

Conclusions

The goal of this thesis was to identify electricity theft not only through static features (demographic information) but also through time variant features (power consumption patterns). This was done by assessing the performance of classifiers built with algorithms that took into consideration the time varying component of crucial information, such as electricity consumption of households over a period of time. Many fields work with this type of information (time series) and, as such, need proper tools to handle and take into consideration the temporal nature and the inherent difference between time variant and invariant features.

The low computational power of FCM allowed for a thorough analysis of the dataset Static CI. The tests done to this set revealed that turning all NaN values to 0 (Method 1) and applying no balancing technique resulted in better performing models. Despite exploring more complex ways to deal with errors in the data and multiple balancing techniques, the simpler approaches ended up yielding the best outcomes. This was helpful for the following tests as it justified the used of smaller datasets thus shortening simulation time. As expected, the three threshold finding methods computed very close points in the ROC curve. Despite choosing the minimum distance method for future tests, as it showed overall slightly better results, it is believed the other methods would have computed similarly performing classifiers.

When using raw data and evaluating balance ratios, the conclusions were similar to the Static CI study. High ratios did not yield good enough results with the current set up (18 time series with a threat model of 16 attacks) to justify the increase in processing time required to run larger datasets. Overall, these tests yielded worse results than Static CI. It is possible that an overload of information on the algorithms could have caused these results. As a consequence, two different approaches were taken to assess whether this was indeed a problem. First the threat model was changed from 8 attacks to 6, followed by tests with only one time series from the 18 initial ones. The first set of tests (different threat model) showed an increase, although negligible, in performance in comparison to the balancing tests. As for the time series reduction tests, these showed a visible deterioration especially when using zero-day attacks. It was clear that reducing the information allocated to the algorithms regarding

consumption patterns hindered their ability to correctly discern between theft and regular consumption despite characterizing the live case scenario of every utility company. Lastly, the impact of the temporal component of the time series on the final classification was studied by applying FCM which disregards this component of the data. As previously mentioned, this clustering algorithm takes every column characterizing each sample as values of different features. In other words, instead of taking, for example, 144 entries of a time series as part of a temporal evolution of a household on a particular time window, the algorithm processes those 144 entries as values for distinct 144 features (for example the static features that represented the socio-demographic questions of the survey). This approach yielded the best results out of all the tests done to the three datasets. Although only one time series was used, the results exceeded all expectations. It is also worth noting that the difference in computational power when using FCM over MFC is clear. A grid search that can take 30 minutes to an hour, maximum, while using FCM can take several continuous days using MFC. The fact that the much faster algorithm also yielded much better results was surprising.

The last dataset, Temporal CI, was used as a middle ground between the other two. It is a database composed of the temporal evolution of the same indicators as Static CI. The algorithms MFC and FCM were applied to this dataset without any changes being made to it (for example changing the features used). While both had unsatisfactory results for the non zero-day scenario, FCM once again showed better results for the other two (substantial difference in zero-day scenario and a negligible one for all attacks). It is unclear why FCM performed so poorly for non zero-day. The tests were repeated multiple times but the outcome was always the same.

Across the three datasets, the best results came from applying FCM to time variant and invariant features. Keeping the temporal nature of the data might not be necessary for high performing classifiers and instead doing so may hinder their performance as it might overload the algorithms with unnecessary information.

5.1 Future Work

The results gathered from the use of MFC were not better than what has already been used in the field of NTL detection. Due to the ability of handling both time variant and invariant features, further investigation (with different algorithms and approaches) is required before completely dismissing this particular method as a viable strategy in electricity fraud identification.

Feature selection

One of the areas targeted in the tests was the number of time series used in the clustering process and how that affected the end result. However, a comprehensive study on the optimal number to use was not performed. A feature selection study was also not implemented when choosing the static features for the datasets of raw information. A smaller amount of time variant and invariant features might yield

faster simulations which were one of the main issues during this work.

Cross validation

With faster simulations it is then possible to apply cross validation which was not done in this thesis. This validation method is widely used as it validates the results from each classifier by applying different data points in each iteration and averaging the results. The more divided the data, the more reliable the final result and higher the computation time. A smaller original dataset allows for a more extensive cross validation.

Length of time series

The time window used for the power consumption patterns was set at 5 (excluding weekends since these present distinct behaviours from working days) before the randomly picked day. The goal was to get a week worth of data for the algorithm to notice a behavioural change on the last day, in the case of the non zero-day threat model. Studying the impact that varying the time window has on the performance of the classifiers might prove useful in understanding the overall weight of the time series on the detection of of NTLs.

Dynamic Time Warping

This approach was not used and instead another data processing method (zero order hold) was adopted to deal with irregularities of the time series. Early tests with DTW showed a significant increase in computation time and since the methods used, in theory, did not compromise the original data, DTW was applied. However, the research done on this approach [32] shows that MFC yields better results when applied with DTW to align the input data. As such, it is plausible that this algorithm may improve the results obtained in this thesis.

Non zero-day attacks

Across all tests, the non zero-day attack vector consistently showed unsatisfactory results. Consequently, the results from classifiers using the *All Attacks* threat model were compromised and were worse than the zero-day scenario. As such, studying different methods to improve the results using the non zero-day threat model would be beneficial and would likely have a significant impact on the performance of classifiers that use the *All Attacks* threat model.

Testing and comparing SVM with MFC

Given the results shown in [22], and the recorded improvements when applying FIS to SVM (as shown in [53]), using the same or a very similar algorithm with the data used in this thesis might prove to give better results than any of the approaches adopted. If it does not, at the very least it will be possible to properly compare FCM and MFC to SVM and any variation of the latter.

Bibliography

[1] M. Mahmood, O. Shivam, P. Kumar, and G. Krishnan. “Real Time Study on Technical Losses in Distribution System”. In: *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering* 3 (2014), pp. 131–137.

[2] T. B. Smith. “Electricity theft: a comparative analysis”. In: *Energy policy* 32.18 (2004), pp. 2067–2076.

[3] F. B. Lewis. “Costly ‘Throw-Ups’: Electricity Theft and Power Disruptions”. In: *The Electricity Journal* 28.7 (2015), pp. 118 –135.

[4] J. P. Navani, N. K. Sharma, and S. Sapra. “Technical and Non-Technical Losses in Power System and Its Economic Consequence in Indian Economy”. In: *International Journal of Electronics and Computer Science Engineering* 1.2 (2012), pp. 757–761.

[5] T. Winther. “Electricity theft as a relational issue: A comparative look at Zanzibar, Tanzania, and the Sunderban Islands, India”. In: *Energy for Sustainable Development* 16.1 (2012), pp. 111 –119.

[6] A. G. Leal and M. Boldt. “A big data analytics design patterns to select customers for electricity theft inspection”. In: *2016 IEEE PES Transmission & Distribution Conference and Exposition-Latin America*. 2016, pp. 1–6.

[7] Siemens. *More revenue by fighting non-technical losses*. URL: https://www.siemens.com/customer-magazine/en/home/energy/agility-in-energy/more-revenue-by-fighting-non-technical-losses.html.

[8] Forbes. *Electricity Theft: A Bigger Issue Than You Think*. URL: https://www.forbes.com/sites/peterdetwiler/2013/04/23/electricity-theft-a-bigger-issue-than-you-think/#56a6c0425ed7x.

[9] International Energy Agency. *IEA - Report*. URL: http://www.iea.org/statistics/statisticssearch/report/?year=2015&country=WORLD&product=ElectricityandHeat.

[10] US Energy Information Agency. *Total Electric Power Industry Summary Statistics, 2016 and 2015*. URL: https://www.eia.gov/electricity/annual/html/epa_01_01.html.

[11] US Energy Information Agency. *How much electricity is lost in transmission and distribution in the United States?* URL: https://www.eia.gov/tools/faqs/faq.php?id=105&t=3.

[12] S. S. Depuru, L. Wang, and V. Devabhaktuni. “Electricity theft: Overview, issues, prevention and a smart meter based approach to control theft”. In: *Energy Policy* 39.2 (2011), pp. 1007–1015.

[13] J. L. Viegas, P. R. Esteves, R. Melício, V. M. Mendes, and S. M. Vieira. “Solutions for detection of non-technical losses in the electricity grid: A review”. In: *Renewable and Sustainable Energy Reviews* 80 (2017), pp. 1256–1268.

[14] K. Dineshkumar, P. Ramanathan, and S. Ramasamy. “Development of ARM processor based electricity theft control system using GSM network”. In: *2015 International Conference on Circuits, Power and Computing Technologies* (2015), pp. 1–6.

[15] D. Grochocki, J. H. Huh, R. Berthier, R. Bobba, W. H. Sanders, A. A. Cárdenas, and J. G. Jetcheva. “AMI threats, intrusion detection requirements and deployment recommendations”. In: *2012 IEEE Third International Conference on Smart Grid Communications*. 2012, pp. 395–400.

[16] S. S. Depuru, L. Wang, and V. Devabhaktuni. “A conceptual design using harmonics to reduce pilfering of electricity”. In: *Power and Energy Society General Meeting, 2010 IEEE*. 2010, pp. 1–7.

[17] C.-H. Lo and N. Ansari. “CONSUMER: A novel hybrid intrusion detection system for distribution networks in smart grid”. In: *IEEE Transactions on Emerging Topics in Computing* 1.1 (2013), pp. 33–44.

[18] S. Amin, G. A. Schwartz, A. A. Cardenas, and S. S. Sastry. “Game-theoretic models of electricity theft detection in smart utility networks: Providing new capabilities with advanced metering infrastructure”. In: *IEEE Control Systems* 35.1 (2015), pp. 66–81.

[19] J. Nagi, K. S. Yap, S. K. Tiong, S. K. Ahmed, and M. Mohamad. “Nontechnical loss detection for metered customers in power utility using support vector machines”. In: *IEEE transactions on Power Delivery* 25.2 (2010), pp. 1162–1171.

[20] T. F. Sanquist, H. Orr, B. Shui, and A. C. Bittner. “Lifestyle factors in US residential electricity consumption”. In: *Energy Policy* 42 (2012), pp. 354–364.

[21] B. Sütterlin, T. A. Brunner, and M. Siegrist. “Who puts the most energy into energy conservation? A segmentation of energy consumers based on energy-related behavioral characteristics”. In: *Energy Policy* 39.12 (2011), pp. 8137–8152.

[22] P. Jokar, N. Arianpoo, and V. Leung. “Electricity Theft Detection in AMI Using Customers’ Consumption Patterns”. In: *IEEE Transactions on Smart Grid* 7.1 (2015), pp. 216–226.

[23] M. Di Martino, F. Decia, J. Molinelli, and A. Fernández. “Improving Electric Fraud Detection using Class Imbalance Strategies”. In: *International Conference on Pattern Recognition Applications and Methods*. 2012, pp. 135–141.

[24] J. L. Viegas and S. M. Vieira. “Clustering-based novelty detection to uncover electricity theft”. In: *2017 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)* (2017), pp. 1–6.

[25] X. Xia, W. Liang, Y. Xiao, and M. Zheng. “BCGI: a fast approach to detect malicious meters in neighborhood area smart grid”. In: *2015 IEEE International Conference on Communications*. 2015, pp. 7228–7233.

[26] J. V. Spirić, M. B. Dočić, and S. S. Stanković. “Fraud detection in registered electricity time series”. In: *International Journal of Electrical Power & Energy Systems* 71.C (2015), pp. 42–50.

[27] S. Kisilevich, F. Mansmann, M. Nanni, and S. Rinzivillo. “Spatio-temporal clustering”. In: *Data mining and knowledge discovery handbook*. 2009, pp. 855–874.

[28] T. W. Liao. “Clustering of time series data — a survey”. In: *Pattern Recognition* 38.11 (2005), pp. 1857–1874.

[29] R. Babuška and H. B. Verbruggen. “Constructing fuzzy models by product space clustering”. In: *Fuzzy model identification*. 1997, pp. 53–90.

[30] F. Cavallaro. “A Takagi-Sugeno Fuzzy Inference System for developing a sustainability index of biomass”. In: *Sustainability* 7.9 (2015), pp. 12359–12371.

[31] S. Simani, S. Farsoni, and P. Castaldi. “Residual generator fuzzy identification for wind farm fault diagnosis”. In: *IFAC Proceedings Volumes* 47.3 (2014), pp. 4310–4315.

[32] C. M. Salgado, M. C. Ferreira, and S. M. Vieira. “Mixed Fuzzy Clustering for Misaligned Time Series”. In: *IEEE Transactions on Fuzzy Systems* 25.6 (2017), pp. 1777–1794.

[33] M. C. Ferreira, C. Salgado, J. L. Viegas, H. Schäfer, C. S. Azevedo, S. M. Vieira, and J. M. Sousa. “Fuzzy modelling based on Mixed Fuzzy Clustering for health care applications”. In: (2015), pp. 1–5.

[34] C. M. Salgado, J. L. Viegas, C. S. Azevedo, M. C. Ferreira, S. M. Vieira, and J. M. Sousa. “Takagi–Sugeno fuzzy modeling using mixed fuzzy clustering”. In: *IEEE Transactions on Fuzzy Systems* 25.6 (2017), pp. 1417–1429.

[35] K. Mehran. “Takagi-Sugeno fuzzy modeling for process control”. In: *Industrial Automation, Robotics and Artificial Intelligence (EEE8005)* 262 (2008).

[36] W. D. Ross, ed. *Aristotle: Metaphysics*. 2 vols. Oxford: Sandpiper Books, 1997. Repr.

[37] L. A. Zadeh. “Fuzzy sets””. In: *Information and Control* 8.3 (1965), pp. 338 –353.

[38] J. M. Sousa and K. Uzay. *Fuzzy decision making in modeling and control*. Vol. 27. 2002.

[39] E. H. Mamdani and S. Assilian. “An experiment in linguistic synthesis with a fuzzy logic controller”. In: *International Journal of Man-Machine Studies* 7.1 (1975), pp. 1–13.

[40] T. Takagi and M. Sugeno. In: 15.1 (1985).

[41] L. A. Zadeh. “The concept of a linguistic variable and its application to approximate reasoning-III”. In: *Information sciences* 9.1 (1975), pp. 43–80.

[42] A. McGregor, M. Hall, P. Lorier, and J. Brunskill. “Flow clustering using machine learning techniques”. In: *International Workshop on Passive and Active Network Measurement*. 2004, pp. 205–214.

[43] M. N. Ahmed, S. M. Yamany, N. A. Mohamed, and A. A. Farag. “A modified fuzzy C-means algorithm for MRI bias field estimation and adaptive segmentation”. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. 1999, pp. 72–81.

[44] K.-S. Chuang, H.-L. Tzeng, S. Chen, J. Wu, and T.-J. Chen. “Fuzzy C-Means Clustering with Spatial Information for Image Segmentation”. In: *Computerized Medical Imaging and Graphics* 30.1 (2006), pp. 9–15.

[45] W. Cai, S. Chen, and D. Zhang. “Fast and robust fuzzy c-means clustering algorithms incorporating local information for image segmentation”. In: *Pattern Recognition* 40.3 (2007), pp. 825–838.

[46] J. C. Dunn. “A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters”. In: *Journal of Cybernetics* 3.3 (1973), pp. 32–57.

[47] J. C. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms*. 1^{st}. Springer US, 1981.

[48] S. Lloyd. “Least squares quantization in PCM”. In: *IEEE Transactions on Information Theory* 28.2 (1982), pp. 129–137.

[49] H. Izakian, W. Pedrycz, and I. Jamal. “Clustering spatiotemporal data: An augmented fuzzy c-means”. In: *IEEE Transactions on Fuzzy Systems* 21.5 (2013), pp. 855–868.

[50] H. Frigui and O. Nasraoui. “Unsupervised learning of prototypes and attribute weights”. In: *Pattern recognition* 37.3 (2004), pp. 567–581.

[51] L. I. Kuncheva. *Combining pattern classifiers: methods and algorithms*. John Wiley & Sons, 2004.

[52] J. V. Spirić, S. S. Stanković, and M. B. Dočić. “Identification of suspicious electricity customers”. In: *International Journal of Electrical Power & Energy Systems* 95 (2018), pp. 635–643.

[53] J. Nagi, K. S. Yap, S. K. Tiong, S. K. Ahmed, and F. Nagi. “Improving SVM-based nontechnical loss detection in power utility using the fuzzy inference system”. In: *IEEE Transactions on power delivery* 26.2 (2011), pp. 1284–1285.

[54] E. W. Angelos, O. R. Saavedra, O. A. Cortés, and A. N. de Souza. “Detection and identification of abnormalities in customer consumptions in power distribution systems”. In: *IEEE Transactions on Power Delivery* 26.4 (2011), pp. 2436–2442.

[55] ISSDA. *Data from the Commission for Energy Regulation*. URL: `http://www.ucd.ie/issda/`.

[56] J. L. Viegas, P. R. Esteves, and S. M. Vieira. “Clustering-based novelty detection for identification of non-technical losses”. In: *International Journal of Electrical Power & Energy Systems* 101 (2018), pp. 301–310.

[57] Andrew P Bradley. “The use of the area under the ROC curve in the evaluation of machine learning algorithms”. In: *Pattern Recognition* 30.7 (1997), pp. 1145–1159.

[58] Konstantina Kourou, Themis P Exarchos, Konstantinos P Exarchos, Michalis V Karamouzis, and Dimitrios I Fotiadis. “Machine learning applications in cancer prognosis and prediction”. In: *Computational and Structural Biotechnology Journal* 13 (2015), pp. 8–17.

[59] J. A. Hanley and B. J. McNeil. “The meaning and use of the area under a receiver operating characteristic (ROC) curve.” In: *Radiology* 143.1 (1982), pp. 29–36.

[60] Jin Huang and Charles X Ling. “Using AUC and accuracy in evaluating learning algorithms”. In: *IEEE Transactions on knowledge and Data Engineering* 17.3 (2005), pp. 299–310.

[61] Charles X Ling, Jin Huang, and Harry Zhang. “AUC: a better measure than accuracy in comparing learning algorithms”. In: *Conference of the Canadian society for computational studies of intelligence*. 2003, pp. 329–341.

[62] J. A. Hanley and B. J. McNeil. “A method of comparing the areas under receiver operating characteristic curves derived from the same cases.” In: *Radiology* 148.3 (1983), pp. 839–843.

[63] D. Hand. “Measuring classifier performance: a coherent alternative to the area under the ROC curve”. In: *Machine learning* 77.1 (2009), pp. 103–123.

[64] C. Ferri, J. Hernández-Orallo, and P. A. Flach. “A coherent interpretation of AUC as a measure of aggregated classification performance”. In: *Proceedings of the 28th International Conference on Machine Learning*. 2011, pp. 657–664.

[65] William J Youden. “Index for rating diagnostic tests”. In: *Cancer* 3.1 (1950), pp. 32–35.

[66] K. Zhou, C. Fu, and S. Yang. “Fuzziness parameter selection in fuzzy c-means: The perspective of cluster validation”. In: *Science China Information Sciences* 57.11 (2014), pp. 1–8.

[67] New Straits Times. *Six suspected to be behind electricity thefts arrested*. 1999.

Appendices

Appendix A

Cost analysis for Static Consumption Indicators using Fuzzy C-Means

One of the main solutions to electricity theft is investing in inspections and monitoring of power users. In the year 2000, CEMIG (a Brazilian power company) registered losses of $12 million but by investing $2.1 million on tests and inspections, they were able to recover $6.2 million [2]. In 1999, Malaysian inspection teams revealed that 587 out of 684 suspected cases were confirmed to have stolen electricity [67]. However, said inspections are costly so companies want to prioritize a high degree of certainty before deciding to proceed with an inspection. It may result in some cases of fraud to go unchecked but will allow utility companies to better manage inspection resources and solely go after those who are clearly committing theft. To replicate this strategy, it was decided to assess the impact of varying the importance given to false positives (FP). The bigger the importance of false positives, the more costly they will be to the company forcing the algorithm to lower their occurrences. Varying FP allows for different points in the ROC curve to be picked, similar to Youden Index and the minimum distance method.

A.1 Implementation of the tests on False Positives and False Negatives

As previously mentioned, depending on the application and the desired focus it is possible to focus on giving priority to lowering the misclassified positive and negative cases with the emphasis on false positives. The algorithms used allowed for certain “costs” (weights) to be attributed to each entry of the confusion matrix. For example, for the case where the predicted class is positive and the true class is negative, the cost nomenclature is read as ”cost of the negative points which are predicted to be positive”, Cost(P|N). The remaining costs can be seen in table A.1. One of the constraints of the method is that the weights of False Positive (FP) and False Negative (FN) need to be higher than

their counterparts (True Positive and True Negative). On the other hand, the relative weights between FP and FN are not bound and will be placed under scrutiny. To evaluate the influence of giving more importance to FP rather then FN, and vice versa, two experiments were run. One consisted in keeping the weight for FP constant at 2 and the weight for FN varying along $[2; 8]$, while the other test consisted on the opposite, varying FP and maintaining FN.

Table A.1: Confusion Matrix with the respective Cost nomenclature

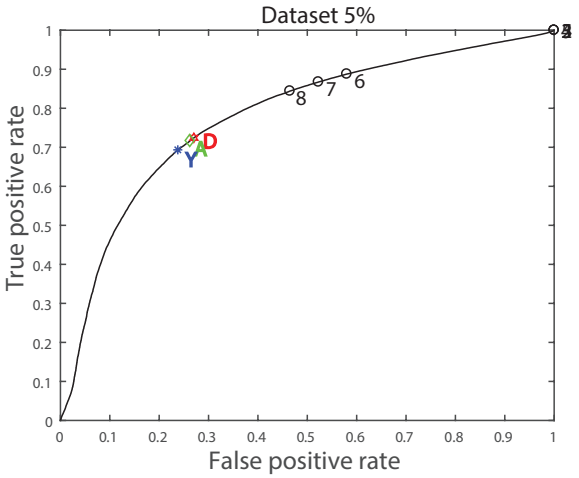
		Predicted	
		Positve	Negative
Actual	Positive	Cost (P|P)	Cost (N|P)
	Negative	Cost (P|N)	Cost (N|N)

A.2 Results

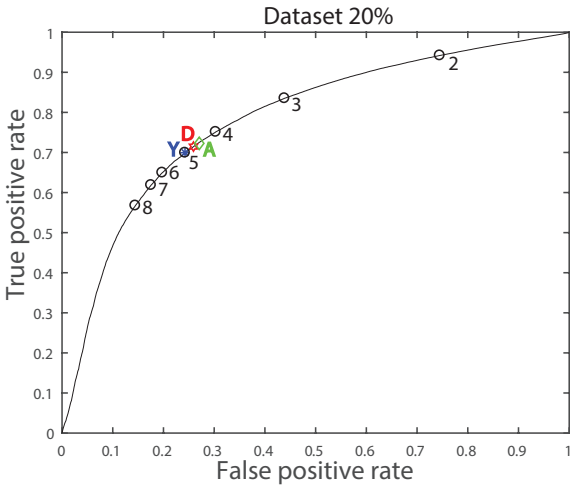
From Figures A.1 and A.2 it is possible to understand the effects of increasing the amount of non fraudulent samples when evaluating changing the relative weights of false positives. The more balanced the dataset, the closer the resulting thresholds get to point (0,0). This is shown when turning only the NaN entries that characterize the attack models 5 and 50 into 0 and the remaining NaN into the average values (Method 3) and turning all NaN into the average value of the corresponding feature (Method 2). Same thing happened with Method 1 but to keep this section brief it was omitted.

In theory, as the weight increases, the program further prioritizes decreasing the amount of FP. Given the usual shape of a ROC curve, this decrease will lead to points with also lower TP. The goal then becomes one of balancing the two, amount of FP with TP. In the unbalanced datasets, the increase in weights leads to a higher decrease of FP than TP. This is the ideal situation where a company that wants to better reallocate their resources is able to do so without letting many malicious consumers go unchecked. However, there comes a point where the decrease of FP no longer outweighs the one registered for TP. This breakpoint usually comes after the points computed using Youden index and the other previously mentioned methods. After this point, the classification registered from the model begins to show less and less positive cases but with such a low FPR the chances theses are accurate are quite high.

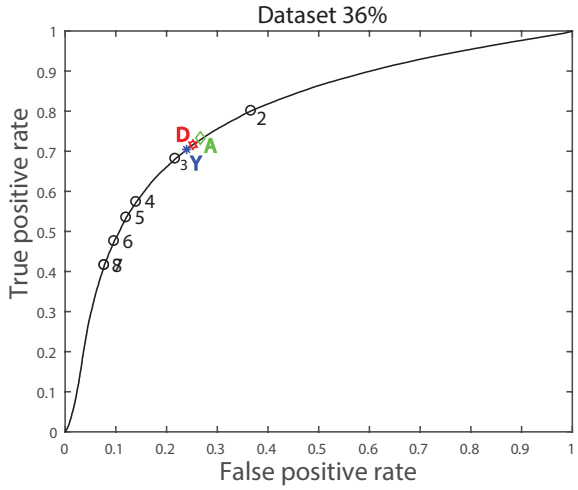
On the opposite end, when the weights result in points close or even over the graph point (1,1), the classifier is identifying every sample as positive which ends with a FPR of 1. This can be easily seen in the unbalanced dataset where low weights yield practically the same threshold.



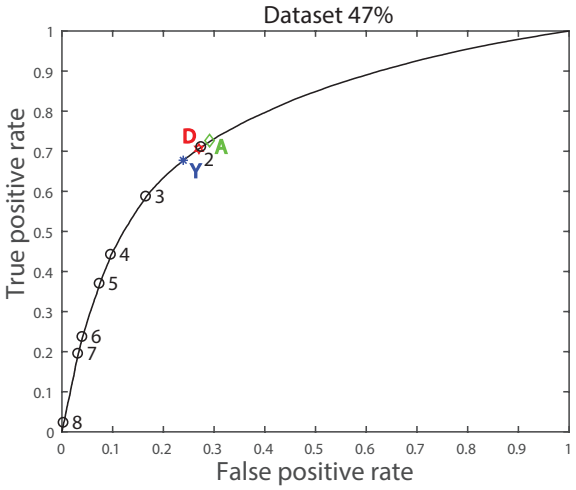
(a) Unbalanced dataset



(b) Dataset with 20% non fraudulent samples

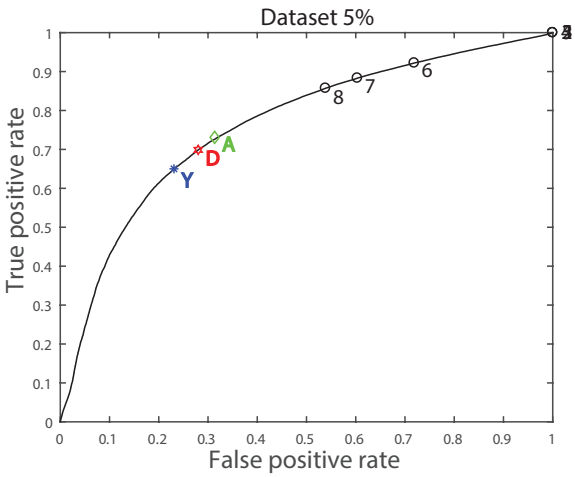


(c) Dataset with 36% non fraudulent samples

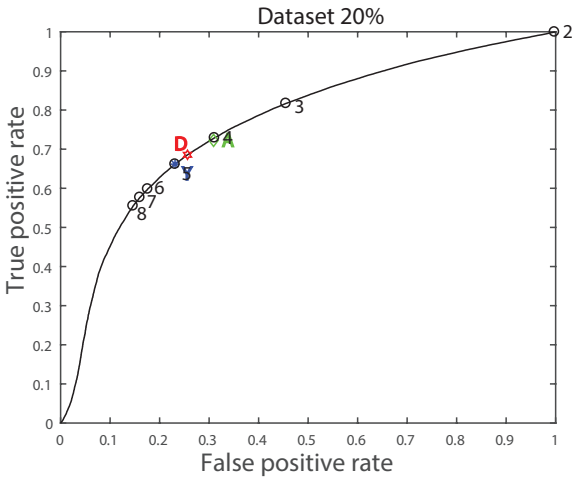


(d) Dataset with 47% non fraudulent samples

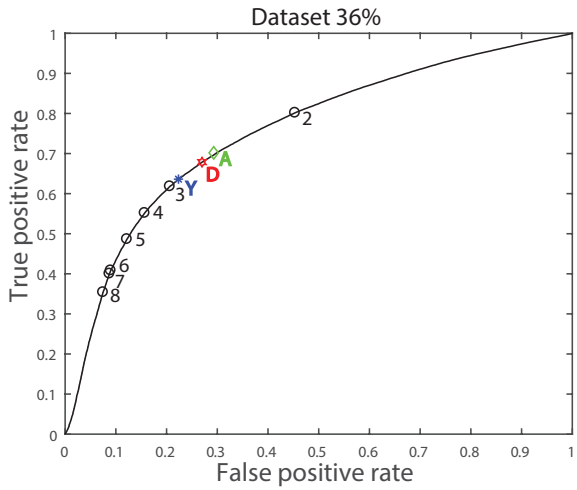
Figure A.1: Evaluation of FP weights when applying Method 3



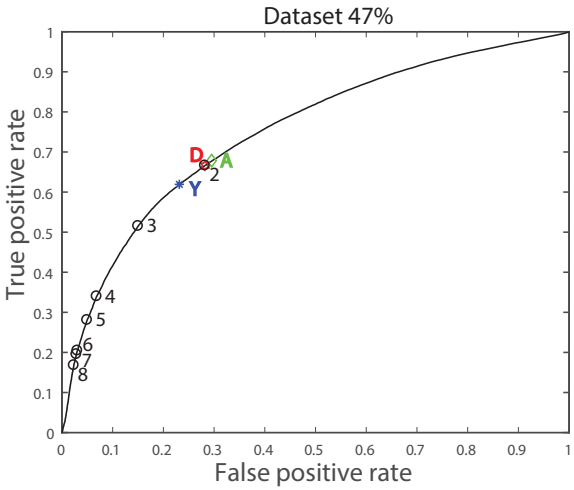
(a) Unbalanced dataset



(b) Dataset with 20% non fraudulent samples



(c) Dataset with 36% non fraudulent samples



(d) Dataset with 47% non fraudulent samples

Figure A.2: Evaluation of FP weights when applying Method 2

A.3 Conclusions

If no data balancing technique is used and there is an increased interest in reducing FP by attributing higher weights to the confusion matrix entry FP, then it is important to have that relative weight high (4 to 5 times higher) in order to reduce FPR and not compromise TPR. If lower values of FPR are required it is imperative to acknowledge the compromise in TPR inherent with the increased weights, and subsequent decrease in FPR.

Balancing techniques cause a drastic change in the results of weight testing. It is then critical to comprehend said techniques so as to not over weigh FP and cause for FPR and TPR to yield 0.