

FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Day ahead electricity market: identification of bidding strategies with information theoretic tools

Vítor Manuel Correia Navega



Mestrado em Engenharia Electrotécnica e de Computadores Major Energia

Supervisor: Professor Vladimiro Miranda

November 17, 2022

The following people collaborated with the supervising team during this work: Dr. José Villar, André Oliveira and Dr. Jean Sumalli.

Resumo

O mercado ibérico de eletricidade (MIBEL) funciona através de um sistema de ofertas de venda por parte dos produtores e ofertas de compra por parte de comercializadores e consumidores qualificados a participar em mercado de tipo pool. Neste mercado, o preço da energia resultante é o chamado preço marginal, que corresponde ao preço de venda da unidade mais cara. Embora num mercado com competência perfeita os produtores só têm incentivo a ofertar ao seu custo marginal de produção, na prática é possível aplicar comportamentos estratégicos para incrementar o lucro resultante da operação no mercado.

Neste trabalho serão desenvolvidas ferramentas para analisar a viabilidade da aplicação da metodologia Information Theoretical Mean Shift para analisar comportamentos estratégicos dos agentes do mercado. A ideia de base é identificar padrões de comportamento diferentes na sequência temporal das ofertas assim como os momentos em que esses padrões de comportamento mudam.

Abstract

The Iberian Electricity Market (MIBEL) operates through a system of offers to sell by generators and offers to buy by trading companies or consumers qualified to participate in the day-ahead market. The price at which electricity is traded in this market is called the marginal price and is equal to the marginal price of the most expensive generation unit whose offer has been accepted. In a perfect market, generators only have the incentive to bid at their marginal production price, but in real situations it is possible that they adopt strategic behaviors to change their bidding strategy, which allows them to gain market advantages and thus make larger profits. The objective of this work is to develop tools to analyze the feasibility of using Information Theoretic Learning methods, in particular the Theoretic Information Mean Shift, to identify these strategic behaviors of market participants, through an unexplored methodology for this application. The basic concepts behind this are the identification of different behavioral patterns of market agents. If this study proves successful, it will provide the ability to predict and classify different market behaviors, proving its value to companies in the energy market.

Agradecimentos

Estou grato ao meu supervisor, Professor Vladimiro Miranda, por me ter dado a oportunidade de realizar este trabalho e pela sua confiança em mim. Estou-lhe profundamente grato por me ter proporcionado este incentivo e disponibilidade. Não podia estar mais satisfeito por tê-lo ao meu lado no culminar desta longa viagem. Muito obrigado.

Gostaria de agradecer ao meu co-supervisor, Professor José Villar, pela sua ajuda e apoio na resolução dos problemas que surgiram, mesmo à custa de sacrificar algumas horas do meu tempo. Os seus ensinamentos e conselhos foram fundamentais para este trabalho.

Ao André Oliveira, que tem estado comigo e me guiado ao longo de todas as etapas deste projecto.

Ao Dr. Jean Sumaili por me ter disponibilizado o seu tempo e atenção para me ensinar de forma simples os conceitos complexos relacionados com esta tese bem como acompanhado na sua elaboração.

Aos meus pais, por todo o seu amor, palavras e conselhos, por lutarem para me verem feliz. E pela garantia de que tenho alguém que nunca me abandonará.

Ao meu irmão, com quem cresci. Mesmo que tomemos caminhos diferentes na vida, a nossa ligação é de sangue e eterna.

A minha avó Amelia. Por caminhar comigo enquanto crescia e pela sua dedicação e amor. Aos meus avós restantes, por tudo o que me deram. Se eu pudesse ter lá estado, teria ficado orgulhoso.

Aos meus amigos Ricardo, Rui e Tiago que me acompanharam de todo o coração através dos melhores e piores momentos. A nossa amizade é um privilégio que sempre acarinharei.

Vítor Navega

*“People always tell you, ‘Be humble. Be humble.’
When was the last time someone told you to be amazing?
Be great! Be great! Be awesome! Be awesome!”*

Ye

Contents

1	Introduction	1
1.1	Problem and motivation	2
2	State of the art	4
2.1	The market context: state of the art and background	5
2.2	The necessary tools: state of the art and background	10
2.2.1	Kernel Density Estimation (KDE) / Parzen Window	11
2.2.2	Renyi's quadratic entropy and information potential	14
2.2.3	Distance of Cauchy-Schwarz	15
2.2.4	Information Theoretic Mean Shift (ITMS)	16
2.2.5	Mutual Information	17
2.3	Conclusions	18
3	Data Preprocessing	20
3.1	Initial considerations and study limitations	21
3.2	Data model design: Power BI	22
3.2.1	Complete Data Model	24
3.3	Pre-Processing the Database: Python	25
3.4	Normalization	30
4	Data clustering: ITMS application	33
4.1	ITMS in energy markets- A new approach	33
4.2	Results validation	38
4.2.1	Representing the clusters	38
4.2.2	Curve Classification	43
4.2.3	Feature Selection	45
4.2.4	Cluster repetition for different periods	46
5	Analysis and interpretation of results	50
5.1	Results interpretation	50
5.1.1	ENDESA GENERACION, S.A	50
5.1.2	IBERDROLA GENERACION ESPANA S.A	56
6	Conclusions and Future Work	63
6.1	Achievements	63
6.2	Conclusions	64
6.3	Future Work	65
	References	67

A	Data Mining	71
A.1	Data Sources	71
A.2	OMIE file details	74
A.2.1	CAB Files	75
A.3	DET Files	76
A.4	PDBC Files	77
B	Power BI	78
B.1	CAB files query	78
B.2	DET files query	79
B.3	PDBC files query	81
B.4	Market Price	82
B.4.1	PDBC-STOTA files query	84
B.4.2	Other queries	86
C	Pre-processing: Python code	88
C.1	Data_Pre_Processing class	88
C.2	Filtering by hour and agent	88
C.3	Process class	89
C.4	Creating the cumulative curves	90
C.5	Multi-agents scenario	90
C.6	Multi-hour scenario	90
D	Pre-processing results	92
D.1	Commulative curves data framework	93
E	ITMS code	95
E.1	Update	95
E.2	best parameters	96
F	Data Validation	97
F.1	Cluster representation	97
F.1.1	Euclidean Distance	97
F.1.2	ITMS	97
F.2	Classification: Modified ITMS	98
F.3	Explanatory variables	99
F.3.1	Cross-entropy	99
F.4	trans period graphics	100
F.4.1	hourly sliding window	100
G	Scenario 1 results for IBERDROLA GENERACION ESPANA S.A	101
H	Scenario 1 results for ENDESA GENERACION, S.A	110

List of Figures

2.1	Aggregate supply and demand curves. Source: <i>MIBEL</i>	6
2.2	Day-time map showing the temporal evolution of electricity demand between 1/1/2003 and 31/3/2003 [41]	9
2.3	Decision tree to explain membership in a pattern based on price and demand [41]	9
2.4	Illustration the sample of 8 points D [21]	13
2.5	Estimation of the pdf with the Parzen windows method for a sample of 8 points, with $\sigma = 0,3$ [21].	13
2.6	Figure 6: Estimation of the true pdf with the Parzen windows method for a sample of 8 points, with $\sigma = 0,6$ [21].	14
2.7	Illustration of the relations among the individual entropies $H(X)$ and $H(Y)$, the joint entropy $H(X,Y)$ and the conditional entropies $H(X Y)$ and $H(Y X)$, for two random variables X and Y with mutual information $I(X,Y)$	18
3.1	Data mining process scheme [44]	20
3.2	ETL process diagram	23
3.3	Class diagram portraying all the necessary relationships of all the queries	24
3.4	Diagram of Power BI and DAX Studio relation	25
3.5	Example step-wise curves for <i>ENDESA GENERACION, S.A</i> at 09:00	29
3.6	Heatmap for <i>ENDESA GENERACION, S.A</i> at 09:00	30
3.7	Trigonometric and step-wise representations	31
4.1	Final curves using different λ and σ parameters for <i>ENDESA GENERACION, S.A</i>	35
4.2	Final curves using different λ and σ parameters for <i>IBERDROLA GENERACION ESPANA S.A</i>	36
4.3	Results for single agent and hour scenario for <i>ENDESA GENERACION, S.A</i>	37
4.4	Results for single agent and hour scenario for <i>IBERDROLA GENERACION ESPANA S.A</i>	38
4.5	Minimized clusters for single-agent and single-hour scenario using <i>ENDESA GENERACION, S.A</i> and <i>IBERDROLA GENERACION ESPANA S.A</i>	40
4.6	Results obtained in the <i>ITMS</i> method for single-agent and single-hour scenario using <i>ENDESA GENERACION, S.A</i> and <i>IBERDROLA GENERACION ESPANA S.A</i>	41
4.7	Minimized clusters for single-agent and single-hour scenario using <i>ENDESA GENERACION, S.A</i> and <i>IBERDROLA GENERACION ESPANA S.A</i>	41
4.8	New curves classification for <i>ENDESA GENERACION, S.A</i> for single agent and hour scenario	44
4.9	New curves classification for <i>IBERDROLA GENERACION ESPANA S.A</i> for single agent and hour scenario	44
4.10	Minimized clusters with new classified curves.	45

4.11	Hourly cluster evolution	47
4.12	Daily cluster evolution	48
5.1	Euclidean distance between filtered clusters	51
5.2	Euclidean distance between filtered clusters for different price ranges	52
5.3	Results of Scenario 3 ran for a single agent and multiple hours, for <i>ENDESA GENERACION, S.A</i>	55
5.4	Euclidean distance between filtered clusters	56
5.5	Euclidean distance between filtered clusters for different price ranges	57
5.6	Results for single agent and multiple hour, Scenario 3, for <i>IBERDROLA GENERACION ESPANA S.A</i>	62
A.1	List of files used represented in red boxes from the OMIE website [27]	72
A.2	Example of Yahoo finance table for Natural Gas historical data with red boxes around used variables [53]	73
A.3	SENDECO2 table for CO2 historical data with red boxes around used variables [42]	74
A.4	Format of <i>CAB</i> files and description of the variables [28]	75
A.5	Example of <i>CAB</i> files format [28]	76
A.6	Format of <i>DET</i> files and description of the variables [28]	76
A.7	Example of <i>DET</i> files format [28]	76
A.8	Format of <i>PDBC</i> files and description of the variables [28]	77
A.9	Example of <i>PDBC</i> files format [28]	77
B.1	Flowchart of the major actions taken while creating the <i>CAB</i> Query	78
B.2	Class diagram for the <i>CAB</i> query	79
B.3	Flowchart of the major actions taken while creating the <i>DET</i> Query	80
B.4	Class diagram for the <i>DET</i> query	80
B.5	Flowchart of the major actions taken while creating the <i>PDBC</i> Query	81
B.6	Class diagram for the <i>PDBC</i> query	81
B.7	Flowchart of the major actions taken while creating the <i>Market Price</i> Query	83
B.8	Class diagram for the <i>Market Price</i> query	83
B.9	Flowchart of the major actions taken while creating the <i>PDBC-STOTA</i> Query	85
G.1	Hour 00:00, <i>IBERDROLA GENERACION ESPANA S.A</i>	101
G.2	Hour 00:01, <i>IBERDROLA GENERACION ESPANA S.A</i>	101
G.3	Hour 00:02, <i>IBERDROLA GENERACION ESPANA S.A</i>	102
G.4	Hour 00:03, <i>IBERDROLA GENERACION ESPANA S.A</i>	102
G.5	Hour 00:04, <i>IBERDROLA GENERACION ESPANA S.A</i>	102
G.6	Hour 00:05, <i>IBERDROLA GENERACION ESPANA S.A</i>	103
G.7	Hour 00:06, <i>IBERDROLA GENERACION ESPANA S.A</i>	103
G.8	Hour 00:07, <i>IBERDROLA GENERACION ESPANA S.A</i>	103
G.9	Hour 00:08, <i>IBERDROLA GENERACION ESPANA S.A</i>	104
G.10	Hour 09:00, <i>IBERDROLA GENERACION ESPANA S.A</i>	104
G.11	Hour 10:00, <i>IBERDROLA GENERACION ESPANA S.A</i>	104
G.12	Hour 11:00, <i>IBERDROLA GENERACION ESPANA S.A</i>	105
G.13	Hour 12:00, <i>IBERDROLA GENERACION ESPANA S.A</i>	105
G.14	Hour 13:00, <i>IBERDROLA GENERACION ESPANA S.A</i>	105
G.15	Hour 14:00, <i>IBERDROLA GENERACION ESPANA S.A</i>	106
G.16	Hour 15:00, <i>IBERDROLA GENERACION ESPANA S.A</i>	106

G.17	Hour 16:00, <i>IBERDROLA GENERACION ESPANA S.A</i>	106
G.18	Hour 17:00, <i>IBERDROLA GENERACION ESPANA S.A</i>	107
G.19	Hour 18:00, <i>IBERDROLA GENERACION ESPANA S.A</i>	107
G.20	Hour 19:00, <i>IBERDROLA GENERACION ESPANA S.A</i>	107
G.21	Hour 20:00, <i>IBERDROLA GENERACION ESPANA S.A</i>	108
G.22	Hour 21:00, <i>IBERDROLA GENERACION ESPANA S.A</i>	108
G.23	Hour 22:00, <i>IBERDROLA GENERACION ESPANA S.A</i>	108
G.24	Hour 23:00, <i>IBERDROLA GENERACION ESPANA S.A</i>	109
H.1	Hour 00:00, <i>ENDESA GENERACION, S.A</i>	110
H.2	Hour 00:01, <i>ENDESA GENERACION, S.A</i>	110
H.3	Hour 00:02, <i>ENDESA GENERACION, S.A</i>	111
H.4	Hour 00:03, <i>ENDESA GENERACION, S.A</i>	111
H.5	Hour 00:04, <i>ENDESA GENERACION, S.A</i>	111
H.6	Hour 00:05, <i>ENDESA GENERACION, S.A</i>	112
H.7	Hour 00:06, <i>ENDESA GENERACION, S.A</i>	112
H.8	Hour 00:07, <i>ENDESA GENERACION, S.A</i>	112
H.9	Hour 00:08, <i>ENDESA GENERACION, S.A</i>	113
H.10	Hour 09:00, <i>ENDESA GENERACION, S.A</i>	113
H.11	Hour 10:00, <i>ENDESA GENERACION, S.A</i>	113
H.12	Hour 11:00, <i>ENDESA GENERACION, S.A</i>	114
H.13	Hour 12:00, <i>ENDESA GENERACION, S.A</i>	114
H.14	Hour 13:00, <i>ENDESA GENERACION, S.A</i>	114
H.15	Hour 14:00, <i>ENDESA GENERACION, S.A</i>	115
H.16	Hour 15:00, <i>ENDESA GENERACION, S.A</i>	115
H.17	Hour 16:00, <i>ENDESA GENERACION, S.A</i>	115
H.18	Hour 17:00, <i>ENDESA GENERACION, S.A</i>	116
H.19	Hour 18:00, <i>ENDESA GENERACION, S.A</i>	116
H.20	Hour 19:00, <i>ENDESA GENERACION, S.A</i>	116
H.21	Hour 20:00, <i>ENDESA GENERACION, S.A</i>	117
H.22	Hour 21:00, <i>ENDESA GENERACION, S.A</i>	117
H.23	Hour 22:00, <i>ENDESA GENERACION, S.A</i>	117
H.24	Hour 23:00, <i>ENDESA GENERACION, S.A</i>	118

List of Tables

3.1	Example dataset <i>PE</i>	27
3.2	3 equal-sized ['Price'] buckets	28
3.3	Discretized prices with matching energy offers	29
5.1	Amount of non clustered curves per cluster (pre formed and new) for the <i>ENDESA GENERACION, S.A</i> agent	53
5.2	Most relevant variables according to Mutual Information	54
5.3	Similarity between clusters from Scenario 1 and Scenario 3	56
5.4	Most similar and Least similar clusters	57
5.5	Number of non clustered curves per already formed and new clusters for the <i>IBERDROLA GENERACION ESPANA S.A</i> agent	60
5.6	Most relevant variables according to Mutual Information	61
5.7	Similarity between clusters from Scenario 1 and Scenario 3	62
B.1	Header and example first 10 rows of the <i>CAB</i> Query	79
B.2	Header and example first 10 rows of the <i>DET</i> Query	80
B.3	Header and example first 10 rows of the <i>PDBC</i> Query	82
B.4	Header and example first 10 rows of the <i>Market Price</i> Query	83
B.5	Header and example first 10 rows of the <i>Special Regime Production</i> Query	85
B.6	Header and example first 10 rows of the <i>Hydro Production</i> Query	86
B.7	Header and example first 10 rows of the <i>Demand</i> Query	86
B.8	Header and example first 10 rows of the <i>Natural Gas Price</i> Query	87
B.9	Header and example first 10 rows of the <i>Co2 Price</i> Query	87
D.1	Preprocessed dataframe for <i>ENDESA GENERACION, S.A</i> at hour 09:00	93
D.2	Preprocessed dataframe for <i>IBERDROLA GENERACION ESPANA S.A</i> at hour 09:00	94

Abreviaturas e Símbolos

GBMS	Gaussian Blurring Mean Shift
GMS	Gaussian Mean Shift
ITL	Information Theoretic Learning
ITMS	Information Theoretic Mean Shift
MIBEL	Mercado Ibérico da Energia Elétrica
MO	Market Operator
PDF	Probability Density Function
TSO	Technical System Operator
BI	Bussiness Inteligence
ETL	Extract, Transform, and Load

Chapter 1

Introduction

Historically, electric utilities have had vertically integrated monopolies that limited competition in markets and required full public control. These regulated monopolies have little incentive to increase efficiency and reduce costs to end users. With the goal of improving their economic efficiency, energy system activities have been privatized and liberalized. Unlike vertically integrated structures, liberalized markets promote competition, which benefits end users through lower or at least more cost-reflective prices.

Due to the physical nature of electricity, a true spot market, where immediate delivery of electricity occurs is not possible. Transactions must be scheduled some time in advance of physical delivery to keep the power system in balance and to compensate for the delay in the response of generation technologies.

Competitive pools or power exchanges are a substitute for a true spot market. In most existing pools, pool purchase prices and scheduled supply are set through auctions some time before physical delivery. Final market prices result from adding the cost of imbalances, ancillary services, and possibly other demand-related charges such as capacity payments to the pool purchasing price. Because prices in electricity markets are set based on supply and demand, they are referred to as ex ante pools. [14]

The Iberian Electricity Market (MIBEL) was created by integrating the Portuguese and Spanish electricity markets in response to the European incentive to create regional electricity markets. MIBEL operates with the Poolco model, an ex-ante pool, and offers the possibility for bilateral contracts, with OMIE [20] being the market operator (MO) that manages the Poolco day-ahead and intraday markets.

In poolco markets, the market price can be determined by the intersection of the demand and supply curves, and according to the theory of economic markets, under ideal conditions the intersection of these curves should equal the marginal operating cost or the opportunity cost of the most expensive selling unit cleared. As mentioned above, the notion that the market price is equal to the marginal operating cost is not always true under real conditions, as actors may engage in their own strategic behavior.

Considering an agent's entire production portfolio, strategic behavior can be defined as the

strategy an agent adopts to profit from its potential market power by not bidding at the marginal or opportunity cost of production and by exploiting and profiting from this practice. For example, when a market agent reduces production, it sells less energy, but the resulting price increase leads to an increase in net profit when the market agent's entire portfolio is considered. These behaviors lead to deviations that are not described in several approaches that use operating cost modeling to calculate the market price, resulting in inaccurate forecasting algorithms. By applying the theoretical mean shift method [35], clusters of strategic behaviors are identified to determine their differences and the timing of their occurrence.

Using publicly available data and the techniques explained in more detail in the following chapters, this work aims to explore a market area where there is not yet much published work. This work is limited to a short period of time, so its objective must be limited and precise, and it focuses only on one approach. In terms of results from which valuable information can be extracted, the study contributes to the state of the art since, to our knowledge, no published work has been found on this method applied to energy markets.

The following section defines the objectives of this study. A revision of Chapter 2 is recommended to provide a better understanding of the current fundamental concepts related to energy markets, as well as the mathematical principles and theories behind the required tools.

Prior to clustering the collected data using the *ITMS*, data pre-processing must be performed to obtain a usable data framework. Data pre-processing and data clustering are discussed in Chapters 3 and 4, respectively. Although the development of the work is a laborious process that has great practical value and is necessary for the study, the true scientific value of this study lies in the analysis of the results and the conclusions. In order to keep Chapters 3 and 4 clear and relevant, the practical steps have been distributed to separate appendices and should be analyzed if one wishes to reuse this work or learn about data mining and clustering techniques.

In this way, the most important scientific parts of this thesis are presented in Chapters 5 and 6, where an interpretation of the results and the conclusions drawn are included.

1.1 Problem and motivation

Reforms in the energy markets show a converging tendency toward the introduction of consumer choice of electricity supplier as the cornerstone of effective reform, i.e., the promotion of competition not only in generation but also in electricity trading and supply to consumers as a service [14].

Energy forecasting is essential for efficient scheduling of power distribution to the consumers. If a forecasting model or technique can precisely predict the direction of the market, investment risk and uncertainty can be minimized.

‘‘Nobody knows what the stock market is going to do or even what it ought to do. Hence, the most valuable asset in all business, which is knowledge, is necessarily absent.’’ - Edwin Lefevre

Applying this knowledge to a field where corporate decision makers base their actions on predictions that have value associated with them, it is necessary to understand how markets work in order to make these valuations. The companies with the most accurate predictions can better adjust their decisions and use and control their power in the markets to their advantage.

These concepts form the basis for most markets and apply particularly to energy markets as their reforms move toward introducing consumer choice of electricity supplier as the cornerstone of effective reform, i.e., promoting competition not only in generation but also in electricity trading and supply to consumers as a service. Although short-term forecasts are becoming more accurate, the high volatility of renewables and other external variables limit the ability of market price forecasting models when it comes to medium- or long-term forecasts.

In energy markets, the ability to predict the agents bidding strategies is an essential prerequisite for gaining market power. This study will approach this problem by identifying and characterizing the bidding patterns. This is a fundamental step, because before the application of predictive models can begin, it is first necessary to define what the behavior looks like. Therefore, the expected findings should serve as the basis for completing the desired predictive models. By analyzing the results, it should be determined whether there are consistent patterns in the behavior of agents, what the nature of these patterns are, and what potential they have for strategic analysis when formed using statistical analysis and information theory.

Chapter 2

State of the art

Ninety days after the market auctions completion, all the data contained in the offers are made publicly available in the MIBEL market. The volume of data in the files supplied is considerable, with multiple variables, increasing the complexity to find an appropriate vectorial representation of the data across its various dimensions. Considering the multiple market sessions that occur over time increases the problems complexity by adding a temporal dimension, essential to identify changing behaviors, but that requires more intricate methods for its analysis.

Assuming a vectorial representation for market offers, it is possible to represent them as points in a N dimensional space, where N is the number of variables considered to be relevant. As the initial available data comes in different documents with different types of variables, there is the need to do pre-process data to represent them with vectors that can be used in the Information Theoretic Mean Shift (*ITMS*).

After pre-processing the data, it is possible to use the *ITMS* to determine the bidding modes of market participants. The *ITMS* is a powerful method that allows analyzing the modes by tracking and identifying them. The method is a good adaptive solution to this problem, as it is possible to control clustering by controlling the standard deviation (σ) and the trade-off (λ).

To do so, each point of the N dimensional vector will be theoretically modeled through probability density functions (pdf). It is a common approach to use pdf to represent clusters of bids that allow to characterize a) its type or its distinguishing features, and b) its frequency, while the pdf modes represent the most frequent bid(s) of each cluster.

To our knowledge, no published papers applying this method to electricity markets have been published. However, its promising applicability to various other fields has already been proved and continues to be applied to new ones. This work does not aim to compare all the possible methodologies in the literature, but rather, to demonstrate the value of the Information Theoretic Learning in its application to the electricity markets. In this chapter we explore the existing literature and expand on relevant mathematical and theoretical concepts.

2.1 The market context: state of the art and background

According to [1], [5], there was a paradigm shift following the introduction of competitive unregulated markets in the electricity generation industry, from the companies operating for a utility-based with centralized procedures, to a decentralized PoolCo where the generation firms primarily focus on maximizing their profits.

The MIBEL works with the Poolco model and provides the option for bilateral contracts, with OMIE being the market operator (MO) entity that manages many different markets (day-ahead market, intra-daily, ancillary services as secondary and tertiary reserves, etc.), together with the Transmission System Operator TSO. [49]

The day-ahead market aims to set the daily energy program for the next day. It acts as a symmetric pool where eligible consumers send their purchase offers to the MO. The purchase offers are aggregated from the highest to the lowest price for each hour, resulting in the demand curve [5]. Similarly, the energy producers send their sell offers (or buy offers in the intraday markets) to the MO. The sell offers are ordered from the lowest to the highest price to form the supply curve for each hour.

The offers are defined by a limited number of hourly blocks, each with a different price, and can be simple bids when the blocks among different hours have no intertemporal constraints defined, or complex bids when additional technical or economic conditions, linking the block among different hours, are added. These conditions [23], [27] can be:

- *Indivisibility*: — This condition establishes the requirement to set a minimum operating value in the first block of each hour.
- *Load gradients*: — This condition establishes the maximum variation of energy produced in a unit in two consecutive hours.
- *Minimum income*: — This condition sets that for any bid submitted, the production unit only operates if its income level surpasses a set amount, plus the variable payment set in euros for each MWh matched.
- *Scheduled stop*: — This condition allows the unit to set a schedule for when it will not be running.

Figure 2.1 shows an example of the MO market clearing for the 1st hour of 06/01/2022 from MIBEL. Here it is possible to see how both demand and supply curves intersect on the PoolCo market. If all technical restrains are met, then the market price corresponds to the price of the last selling offer that was accepted.

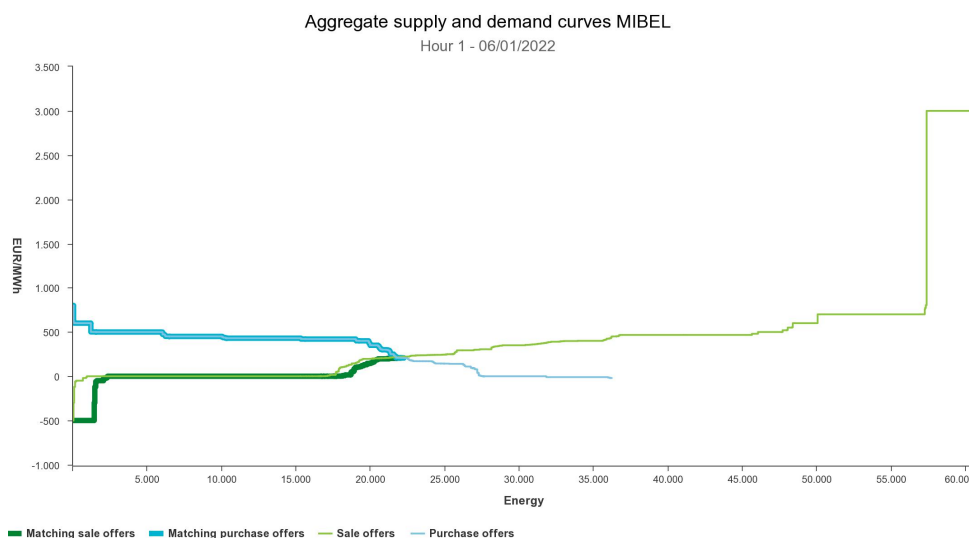


Figure 2.1: Aggregate supply and demand curves. **Source:** MIBEL

After clearing the market by crossing the supply and demand curves and filtering the bids whose complex conditions do not hold, the MO sends the resulting energy schedules (programs) to the transmission system operator (TSO, REN in Portugal, REE in Spain) that applies grid constraints management procedures. Redispatches can take place resulting in new schedules being issued by the TSO and being sent back to the MO. The balance and quality of the energy transmission and distribution system is the main priority when creating the redispatches, with rigid regulations imposed to keep deviations minimized.

With the increased presence of renewable sources for energy generation and, given that these energy sources have a natural dispatch priority in the market due to their lower marginal production costs (making offers at or closer to 0€/MWh), they create a more competitive environment at the cost of its uncertainty and intermittent nature [5]. In addition, the introduction of bilateral contracts allows agents to know demand in advance and better manage production by strategically bidding on the pool markets to maximize their profits. In bilateral markets, the marginal price is set near offer blocks where the slope of the supply curve is greatest. Strategic supply changes made by market participants cause this slope to shift, changing the supply curves patterns of the agents. This has the greatest impact on the market when it has a lower baseload supply or instrumental price [20]. Energy markets are highly regulated, with active maximum and minimum bid prices for the day-ahead and intra-day markets, with recent changes to the intra-day market.

As described in [6] the market design is defined by a variety of bidding structures that constrain the way agents strategically participate and operate their assets. According to this study they defined four basic bidding structures that help the comprehension of how markets work. These four structures are:

- *Simple quantity bidding:* — This is a cost-based approach where the agent participates in the market by bidding quantities of available energy, offered to the market at predetermined prices. Using this kind of approach, the agent might strategically withhold part of their

capacity if it translates into operational profits. This structure is dominant in cost-based markets as is the case of Central and South America [12].

- *Simple price bidding*: —The agent makes all its production available at self-determined prices. While theoretically this option is possible and there have been studies about it [39], it is not widely exercised in real markets.
- *Quantity-price pairs bidding*: — This structure is a combination of the two above approaches. The strategic agent can withhold part of its capacity and express its willingness to sell or hold by the price he presents on his bids. This strategy for bidding is widely used in several electricity markets including the European EPEX spot market and the MIBEL day-ahead market [6].
- *Complex bidding*: — In complex bidding the strategic agent presents its technical constraints to the market operator that must ensure that they are fulfilled when clearing the market. By limiting the strategic agent capacity, it is possible to introduce strategic behaviour in this structure. There is also the possibility for price-based and price-quantity complex bidding structures.

The presence of strategic behaviors was already studied for the secondary reserve markets in [3]. In order better understand how markets are modelled and what type of approaches were used to identify the strategic behaviors, seems relevant to briefly contextualize how energy reserve markets operate. The reserve markets function as a mechanism to keep the system balanced. This study focus especially secondary and tertiary markets, because they use different remuneration mechanisms than primary reserve, which is an automatic response to imbalances in the system [49], [41]. According to the revised literature, there are two main approaches for modelling the reserve markets [3], [18]: joint market models and sequential market models. In sequential models, energy or reserve are dispatched first, leaving the remaining commodity to be traded in subsequent markets. Joint models ensure the dispatch of both commodities optimizing them at the same time. As seen in [9], [33], [16], the joint models tend to present better results but, as they are complex methods, there is the necessity to assume important simplifications.

Regarding market modelling it is possible to find that in [5], [23], [27] the approaches use MIP models in order to model the main technical constraints of generation plants. On these articles it was concluded that this method offers robust solutions. This method can be applied to short term predictions, producing reliable results. There are other, more complex methods, as is the case of hybrid models that use of technical constraints and complement them with additional methodologies that extract further information from the available data, enabling to accurately forecast the competitors bidding curves. In [52], [48], [4] it is possible to further explore the use of hybrid models in energy markets, leading to the conclusion that their use provide better results than when compared to dispatch only algorithms. Since day-ahead markets could incorporate agents' strategic behavior, by extracting modes from agents' bids using the *ITMS*, it is possible to incorporate the results of this study into hybrid models to represent markets more accurately.

In [7] is presented a revision for another widely used model in this area, the conjecture equilibrium model. In summary, these models have for decision variables the produced quantities, and for the so-called conjectures the reaction of the competitors. As this method heavily rely in historical data, there is the need to attention for changes in regulation and technical variables. This makes the method more appropriate for long term predictions.

Although these methods, and many others, are applied sellers bidding curves prediction, the use of clustering techniques in order to classify bidding curves does not seem to be widely applied even though it could be used together with prediction models to further improve their accuracy. By finding strategic patterns in available data, it is expected to develop these approaches by completing them with information contained in the found patterns, to determine when to change the numeric part of hybrid models as well as to obtain samples of future scenarios that may help improving strategic bidding.

Strategic behaviors of market agents can be found when studying the secondary reserve bidding curve in the very short term, when the agents do not bid at marginal or opportunity production costs. If considering its whole portfolio of generation units the agent holds a big market power, then it can exploit the system needs, taking this advantage to profit. For example, withdrawing production would lead to less energy sold but at a higher price which could lead to a net profit increment.

In [3] it is possible to find a clustering technique applied to the secondary reserve market. The methodology aims at creating stepwise secondary reserve bidding curves for the very short term. This method is centered in identifying the patterns in residual demand curves, by applying Kernel Density Estimation (Parzen Window) and models that simulate the main technical constraints of the generation plants, by means of mixed integer models. This method has proven to be accurate and supplied functional predictions. By achieving such conclusions, it not merely demonstrated its significance to energy markets applications, as it proved to be possible to find patterns in markets curves. This article proves of foremost importance for this dissertation, as it makes use of Parzen Window to find cluster patterns, which is a base for the Mean-Shift method used in this thesis and highlights its possible value for these applications as it introduces a new, more fitting non-parametric approach.

In [41] it is possible to better understand the intricacies of mining and visualization of market data, the vast amount of data available as well as the richness and potential of the information that it may contain. It has also expanded on specific techniques for the treatment of supply curves and visualization of these data. The Linear Hinges Model [8] or in alternative the Sigmoid [19] are two supervised learning models proposed for the treatment of the supply curves. These methods extract information from the offer strategies of the different agents can be obtained by applying grouping techniques, facilitating their visualization. It is also proposed to use graphic day-hour maps, that represent in a matrix the hourly data where in the x axis is represented the day of the market and in the y axis the hour of the offer. This method can be seen in the picture bellow, extracted from [41].

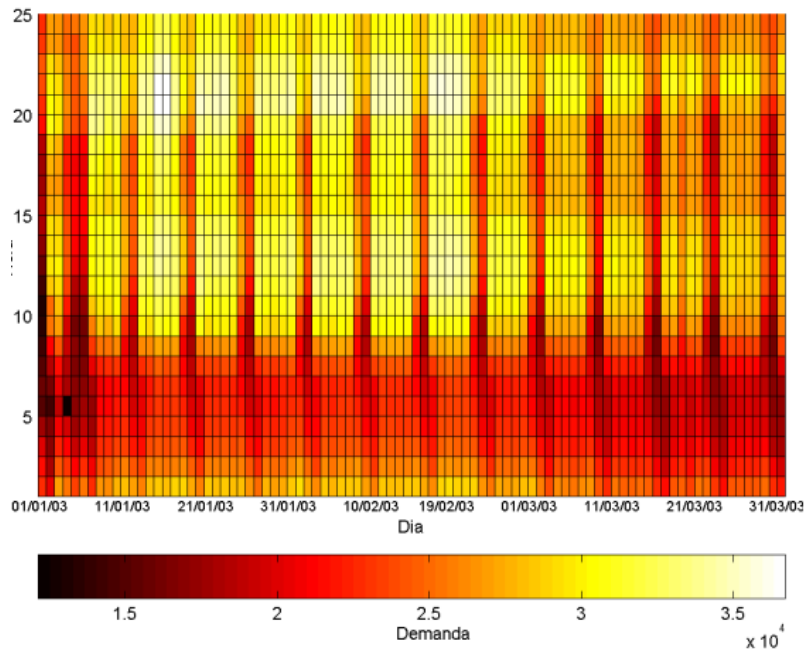


Figure 2.2: Day-time map showing the temporal evolution of electricity demand between 1/1/2003 and 31/3/2003 [41]

This method simplifies pattern recognition, being already widely used in convolutional neural networks and enables the use of decision trees for taking actions according to the demand and the price as can be seen in the picture bellow [41].

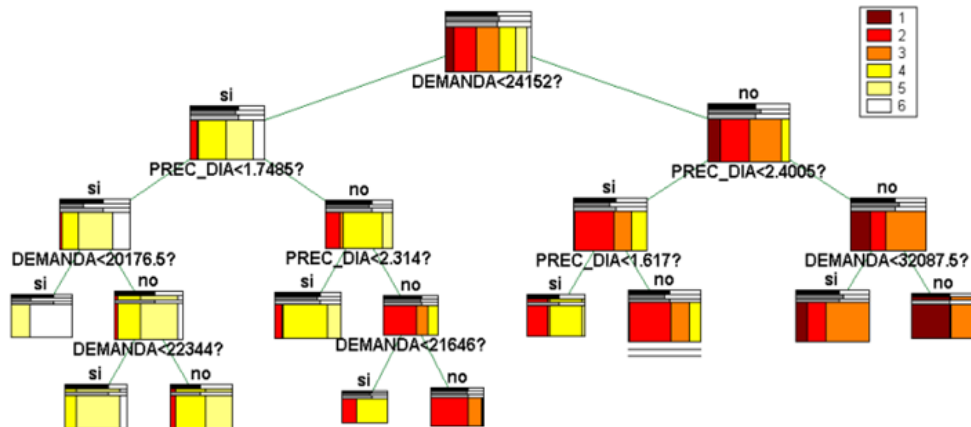


Figure 2.3: Decision tree to explain membership in a pattern based on price and demand [41]

The combination of these techniques is an example of methods already implemented in market studies and presents useful tools suitable for performing analyses of various kinds. In [19] and [1] a semi-functional partial linear model is applied, which is a non-parametric method. By using wind energy predictions as explanatory variable for the linear component, this methodology provides low error results when facing problematic predictive scenarios. It is also added that for less complicated predictive scenarios the naïve method is a more competitive solution. The naïve

method makes use of the hourly inverse residual demand curve of previous day in the training sample (for a particular unit the inverse residual demand curve represents the markets demand minus the supply offers of all other units).

2.2 The necessary tools: state of the art and background

As technology improves over the years, the exponential computing power growth enabled to explore heavy processing methods and theories that before were just theoretical concepts with limited application. As a data driven civilization, the development of methods that treat raw data and extract valuable information from them has a great importance for the present day.

Initially, an untreated dataset may not seem to be very valuable, as most of the times the information and relations that it contains are not specifically defined. As no conclusions can be drawn from an untreated dataset, it is needed to process it in ways that reveal how the data connects, transforming available data in useful information. Accordingly, it is possible to define information as data that has value. Information Theory uses mathematically precise terms to define this concept [32]. This acquired knowledge leads to a better understanding of what can be called the truth contained in the data that we consider relevant to our problem.

One basic concept that was adapted from Physics to Information Theoretic Learning (*ITL*) is the notion of entropy as an idea of unpredictability. This way it is possible to calculate the amount of unpredictability using the information contained in each point. In case a single point can contain the information of the entire dataset, the rest of the dataset becomes redundant as its depiction can be found in one single point [50].

The choice of methods should take into account the amount of information to be preserved and the type of structures to be represented.

Considering possible to discover clusters, an essential question that arises is how to represent them. Considering the two most extreme examples: one could represent a cluster by all the points contained in it, or represent each cluster by a single point, which is the simplest way. When applying clustering techniques, it's intended to gather data particles with similarities, minimizing the entropy.

Accordingly, the mean-shift algorithm, uses this same cluster logic iteratively, shifting particles to the modes of the distribution and it allows to represent both all the points contained in each cluster or a single point. In the *ITMS* there are two parameters that will control the shape of the clusters, being them the standard deviation of the kernel, and a trade-off. The standard deviation controls how details are retained in the clusters by changing the pdf shape. The trade-off controls how the initial dataset is portrayed in the clusters. Unlike the *ITMS*, the classical Mean Shift does not have the initial dataset memory and as a result, if the process runs indefinitely all the particles will converge to one single centroid, losing the relevance of the solution by not retaining any memory of the initial dataset [33].

The Mean Shift method has increased its popularity in recent times and across multiple fields. It became widely used in image processing [15] and vision community to perform clustering, segmentation, and tracking [34].

Over this chapter some of the basic concepts that allow to understand how the MS method works were already explained. Additionally, the *ITMS* makes use of these concepts and combines them with Information Theoretic Learning concepts. This new field helps to mathematically define relations between datapoints, creating a concept of information hidden in the dataset. This information will be useful to reach a higher level of understanding how each point clusters, and how to control those clusters. The MS method was developed into two initial slightly different approaches that began to incorporate ITL concepts:

- *The Gaussian Mean Shift*: — This method is a stable method that can be applied with a stopping criterion that is based on the change of the cost function through the iterations, minimizing Reiny's Cross Entropy.
- *The Gaussian Blurring Mean Shift*: — Since modes can not be defined as stationary or saddle points, the stopping criterion for this method must be heuristic, revealing its unstable nature for finding modes. This method only allows to directly minimize Reiny's Entropy.

The Information Theoretic Mean Shift improved over the limitations of these two methods as it released the need for supervision of the stopping criterion of the Gaussian Blurring Mean Shift and allowed for better control of the cluster formation than the Gaussian Mean Shift [21].

As presented in [40], [36], [38] the *ITMS* can also be used as an enhancement for forecasting. On these papers the *ITMS* was applied since there were scarce datasets that did not contain enough points to train a neural network. To mitigate this lack of useful data a special method referred as densification trick was applied creating new points that allow to improve training process of neural network. The results achieved on these papers verify the robustness of the *ITMS* in finding useful information and the potential use of virtual data for representing real scarce historical data. On [22] the same method was applied successfully on diagnosing faults in power transformers with Autoassociative Neural Networks (ANN) with positive results, once more emphasizing the versatility and potential of *ITMS*.

There are already multiple successful applications of *ITMS* to the energy field that add positive results accordingly to the ones announced before [21], [15], [36], [22], [31].

Concluding the study of the Mean Shift algorithm over the next sub-chapters the focus will be a more in-depth approach to the theoretical backgrounds of the *ITMS*. In the *ITMS* sub-chapter it is going to be defined how to iteratively update each centroid and how to control the trade-off in order to obtain different solutions.

2.2.1 Kernel Density Estimation (KDE) / Parzen Window

Till the 1950's, the histogram was the only nonparametric density estimator being used. The histogram is a powerful tool that allows numerical data analysis in a way that simplifies time series and signal processing.

The use of kernel estimators began with statisticians using them to average shifted histograms. Over the years a vast collection of approaches [39] to nonparametric estimation have been developed, being that choosing the right approach is crucial for problem solving, as not all approaches are equally powerful [24].

For discrete normalized datasets, each point is replaced by a probability, defined by a kernel function. Repeating this process over all the samples and interpolating them it is possible to obtain the density estimation. The function obtained from this method displays how each dataset point weights on the pdf, in accordance with the distance of the point where the kernel is centered. By applying this method to all points of the dataset, it is possible to obtain a complete pdf, a true pdf [9].

This method allows to work with continuous pdf functions of discrete datasets, making it faster and simpler to solve. Equation 2.1 proposed by Wand and Jones (1994) define this method

$$p(x) = \frac{1}{N} \sum_{n=1}^N K \left(\left\| \frac{x - x_n}{\sigma} \right\|^2 \right) \quad x \in R^{DS} \quad (2.1)$$

Where N is the number of points on a dataset DS , K is the Kernel chosen, x is the point where the kernel is centered, x_n is n 'th point of the DS and σ the standard deviation. The two most common Kernels used in the MS are the Gaussian and the Epanechnikov. In practice, the Gaussian Kernel proved to be more efficient than Epanechnikov's as it provides more stable and accurate results. [18]

For this work the Gaussian Kernel fits *ITMS* the best.

$$K(t) = e^{-t/2} \quad (2.2)$$

Represents the Gaussian Kernel and,

$$K(t) = 1 - t, \quad t \in [0, 1] \quad \vee \quad K(t) = 0, t \geq 1 \quad (2.3)$$

Represents the Epanechnikov kernel.

In Equation 2.1 the standard deviation affects how the pdf is shaped. Since the mean-shift method adapts the non-parametric density estimation technique of the Parzen window, it is important to understand how the choice of this parameter shapes the PDF.

The Figures 2.4, 2.5, 2.6 illustrates an example of the changes of the pdf according to different standard deviations using Parzen Window [21].

For a given discrete set of 8 points $D = -1, 1; -0,85; -0,8; 0; 0,2; 0,3; 1,4; 1,6$ where each point is represented by a Dyrac impulse:

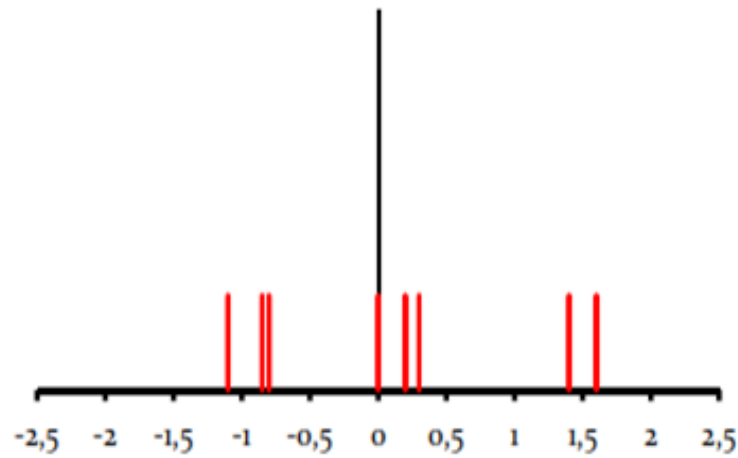
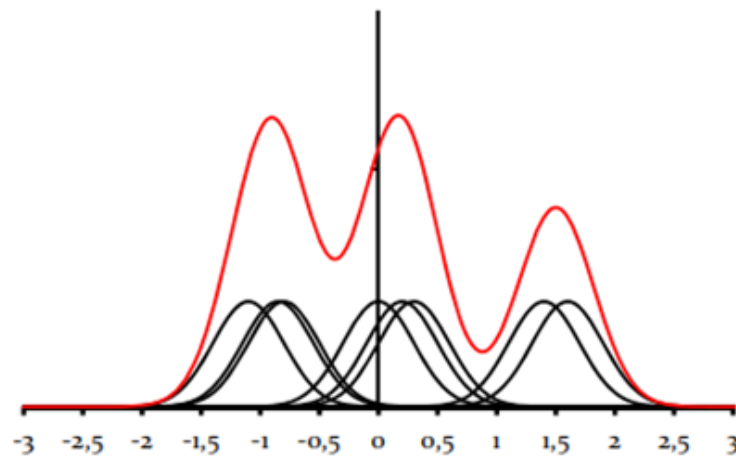


Figure 2.4: Illustration the sample of 8 points D [21]

The true pdf is the unknown actual distribution of probabilities for events to occur in a given situation. The true pdf is determined by placing the selected kernel (for this example the Gaussian kernel is used) over each point and interpolating them.

In Figure 2.5 and Figure 2.6 the Kernel functions are represented in black and the true pdf is the red function.

Figure 2.5: Estimation of the pdf with the Parzen windows method for a sample of 8 points, with $\sigma = 0,3$ [21].

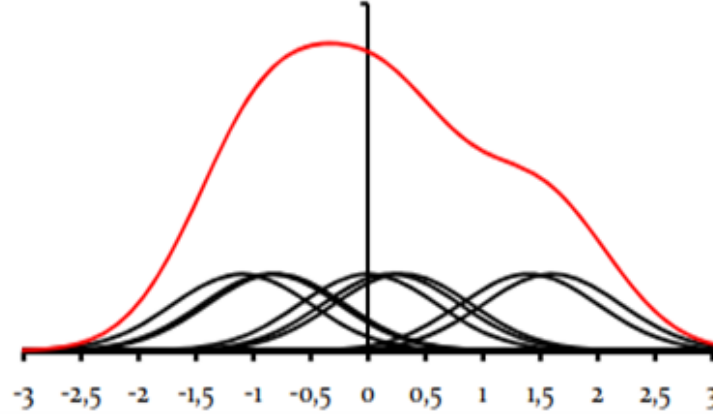


Figure 2.6: Figure 6: Estimation of the true pdf with the Parzen windows method for a sample of 8 points, with $\sigma = 0,6$ [21].

As exemplified by choosing different standard deviations the pdf shape is affected and the wrong choice of this parameter can result in overfitted pdf's if the standard deviation is too big, or under fitted functions if it's too small. In the Mean Shift this parameter will change the shape of the clusters. If the parameter is too small, the dataset retains most of its original form, preserving most of the information, but the dataset is not sufficiently simplified due to a high number of redundant clusters. If the parameter is too large, it causes the dataset clustering to the higher modes, losing information and oversimplifying the shape of the clusters.

2.2.2 Renyi's quadratic entropy and information potential

The Renyi cross entropy is a generalization of the Shannon cross entropy, which has recently been used as a loss function for generative deep learning networks [43].

Cross-entropy loss measures the performance of a classification model whose output is a probability value between 0 and 1. The cross-entropy loss increases when the predicted probability differs from the actual label.

The pdf estimate function defined at Equation 1 can be used to define Renyi's quadratic entropy for a pdf as defined in [16]:

$$H(X) = -\log \left(\int p^2(x) dx \right) \quad (2.4)$$

Using this result and the fact that the integral of the product of two Gaussians is another Gaussian with a variance equal to the sum of the variances, yields the non-parametric estimator for Renyi's quadratic entropy $V(x)$:

$$H(x) = -\log(V(x)) \quad (2.5)$$

$$V(x) = \frac{1}{N^2} \sum_{i=1}^N \sum_{j=1}^N G \left(\left\| \frac{x_i - x_j}{\sigma'} \right\| \right)^2 \quad (2.6)$$

Where $\sigma' = \sqrt{2}\sigma$.

The concept of information potential of the pdf $p(x)$ was named by Principe et al. and collaborators and the potential experienced by a single data point due to all other particles can be expressed as:

$$V(x) = \frac{1}{N^2} \sum_{j=1}^N G\left(\left\|\frac{x_i - x_j}{\sigma'}\right\|\right)^2 \quad (2.7)$$

The derivative of this contribution with respect to the value of this sample is easily evaluated to be:

$$\frac{\partial}{\partial x_i} V(x_i) = \frac{1}{N^2} \sum_{j=1}^N G\left(\left\|\frac{x_i - x_j}{\sigma'}\right\|\right)^2 \left(\frac{x_i - x_{aj}}{\sigma'}\right) \quad (2.8)$$

$$\sum_{j=1}^N F(x_i | x_j) \quad (2.9)$$

At [16] $F(x_i | x_j)$ is defined as an "information force" exerted by x_j on x_i , and $F(x_i)$ is the net force experienced by x_i . Similarly, the cross-entropy equations can be expressed as:

$$H(X; X_o) = -\log(V(X; X_o)) \quad (2.10)$$

$$V(X; X_o) = \int p(x)q(x)dx = \frac{1}{N^2} \sum_{i=1}^N \sum_{j=1}^N G\left(\left\|\frac{x_i - x_{oj}}{\sigma'}\right\|\right)^2 \quad (2.11)$$

It's possible to conclude the following equations for the information potential and force experienced by individual particle:

$$V(x_i; X_o) = \sum_{j=1}^N G\left(\left\|\frac{x_i - x_{oj}}{\sigma'}\right\|\right)^2 \quad (2.12)$$

$$F(x_i; X_o) = \sum_{j=1}^N F(x_i | x_{oj}) \quad (2.13)$$

2.2.3 Distance of Cauchy-Schwarz

The Cauchy-Schwartz distance is an important and widely used concept in mathematics. In *ITMS* it is used as a constant value that constraints the distance between the new and the original dataset.

The following inequation for the inner product space of two vectors is a known mathematical property from the Euclidean spaces [33]

$$(X^T X) (Y^T Y) \geq (X^T Y)^2 \quad (2.14)$$

It can also be written as,

$$\frac{|X \cdot Y|}{\sqrt{\|X\|^2 \|Y\|^2}} \leq 1 \quad (2.15)$$

From this we can deduce the Cauchy-Schwarz distance $D_{CS}(X, Y)$ can be expressed as [16], [2]:

$$D_{CS}(X, Y) = -\log \frac{\int p(x)q(x)dx}{\sqrt{\int p(x)dx \int q(x)dx}} \geq 0 \quad (2.16)$$

2.2.4 Information Theoretic Mean Shift (ITMS)

As mentioned earlier, the Information Theoretical Mean Shift (*ITMS*) algorithm is a two-criteria optimization algorithm. To control how much of the data structure it captures, it minimizes the Renyi Quadratic Entropy of the dataset, while maintaining the Cauchy-Schwarz distance between the new dataset and the original dataset, with a constant value k of $0 < k < H(X_0)$. The concept for this method was introduced by Rao et al in [35], [34] and can be expressed as [33]:

$$J(X) = H(X) + \lambda [-H(X) - (H(X_0) + 2H(X, X_0))] \quad (2.17)$$

The method creates its clusters by minimizing $J(x)$.

By differentiating $J(X)$ iteratively for each point x_i it is possible to find the minimum of $J(X)$:

$$x_i^{\tau+1} = \frac{c_1 \sum_{j=1}^n G\left(\left\|\frac{x_i^{\tau}-x_j^{\tau}}{\sqrt{2}\sigma}\right\|^2\right) x_j^{\tau} + c_2 \sum_{j=1}^n G\left(\left\|\frac{x_i^{\tau}-x_{Oj}}{\sqrt{2}\sigma}\right\|^2\right) x_{Oj}}{c_1 \sum_{j=1}^n G\left(\left\|\frac{x_i^{\tau}-x_j^{\tau}}{\sqrt{2}\sigma}\right\|^2\right) + c_2 \sum_{j=1}^n G\left(\left\|\frac{x_i^{\tau}-x_{Oj}}{\sqrt{2}\sigma}\right\|^2\right)} \quad (2.18)$$

Where,

$$c_1 = \frac{(1-\lambda)}{V(X)} \quad (2.19)$$

$$c_2 = \frac{\lambda}{V(X, X_0)} \quad (2.20)$$

V represents the information potentials and G the presence of the Gaussian Kernel. $V(X)$ represents the information potential of the clustered points in the τ iteration, and $V(X, X_0)$ represents the information potential between the clustered points and the initial dataset.

By varying the value of the trade-off λ , it is possible to obtain different representations of the modes by the way they are clustered. To better understand how to work with this parameter it is important to define three different cases that will ultimately modify the *ITMS* solution making it behave in different ways:

- $\lambda = 0$
- $\lambda = 1$

- $\lambda > 1$

As indicated in [38], when $\tau = 0$ it is possible to reach the following expression for the cost function $J(X)$ from the one previously presented:

$$J(X) = \min_X H(X) \quad (2.21)$$

The fixed-point update equation can now be written as:

$$x_i^{\tau+1} = \frac{\sum_{j=1}^n G\left(\left\|\frac{x_i^{\tau}-x_j^{\tau}}{\sqrt{2}\sigma}\right\|^2\right) x_j^{\tau}}{\sum_{j=1}^n G\left(\left\|\frac{x_i^{\tau}-x_j^{\tau}}{\sqrt{2}\sigma}\right\|^2\right)} \quad (2.22)$$

For this situation the *ITMS* will act as a *GBMS*, having no memory of the initial set and ultimately converging to a single centroid point if the method runs indeterminably. As previously stated, this application of Mean Shift has an unstable nature, demanding the careful choice of the stopping criterium.

For $\lambda = 1$ the cost function can be written as:

$$J(X) = \min_X H(X; X_0) \quad (2.23)$$

Similarly, the fixed-point update rule will have the following form:

$$x_i^{\tau+1} = \frac{\sum_{j=1}^n G\left(\left\|\frac{x_i^{\tau}-x_{0j}}{\sqrt{2}\sigma}\right\|^2\right) x_{0j}}{\sum_{j=1}^n G\left(\left\|\frac{x_i^{\tau}-x_{0j}}{\sqrt{2}\sigma}\right\|^2\right)} \quad (2.24)$$

On this case the algorithm behaves as a *GMS*. As previously stated, this is a stable solution, not requiring any attention when choosing stopping criterium.

When analyzing the situation where $\lambda > 1$, it is important to remember the cost function initial equation:

$$: J(X) = H(X) + \lambda [-H(X) - (H(X_0) + 2H(X, X_0))] \quad (2.25)$$

From this equation is possible to understand that as λ increases over 1 the value of the equality constraint imposed by Cauchy-Schwarz distance [38] also increases. This results in a more detailed representation of the denser regions of the pdf, and when λ tends to infinity, the resulting dataset will represent all the features of the original dataset.

2.2.5 Mutual Information

For two random variables X and Y , the mutual information shared between them can be defined by their probabilistic density function $f(x)$, $f(y)$ and $f(x, y)$ as follows:

$$I(X; Y) = \iint \frac{f(x, y)}{f(x)f(y)} dx dy \quad (2.26)$$

Thus allowing to establish relations between mutual information and entropy as: [50]

$$I(X;Y) = H(X) - H(X|Y) = H(Y) - H(Y|X) \quad (2.27)$$

where $H(X|Y)$ is the conditional entropy defined as

$$H(X|Y) = \int f(y)H(X|Y = Y) dy \quad (2.28)$$

The concepts above are inter-related in a way suggested by 2.7

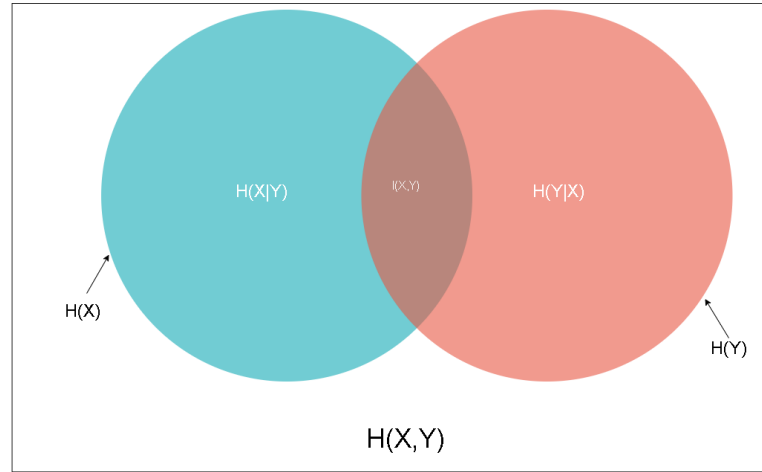


Figure 2.7: Illustration of the relations among the individual entropies $H(X)$ and $H(Y)$, the joint entropy $H(X,Y)$ and the conditional entropies $H(X|Y)$ and $H(Y|X)$, for two random variables X and Y with mutual information $I(X,Y)$

Usually, mutual information is calculated using Shannon's entropy. Estimation of this entropy is computationally intensive when nonparametric methods are used [17], with a computational complexity of $O(N^2)$, where N is the number of samples.

As stated in [17] and [13], the mutual information can be estimated as follows, replacing the Shannon entropy with the Renyi quadratic entropy previously presented in this chapter it is possible to reduce the computational complexity from $O(N)^2$ to $O(N)$.

2.3 Conclusions

For now, it is possible to conclude that there is the need for accurate forecasting of the energy markets, so the agents gain advantage margins over their competitors. These behaviors that aim to maximize one's profit can be defined as strategic behaviors. To make accurate predictions, past data must be analyzed to extract useful information that can be used in predictive models.

The markets have a vast amount of public data that can be analyzed, to extract useful information from it. The treatment of the data and the information extracting methods can be complex and present several problems in practical solutions. The first big decision is regarding the variables that are going to be used in the *ITMS* method, and how they are going to be treated to give

better results. For this the Information Theoretic Mean Shift seems to be an appropriate choice for this study, as it is a robust mode seeking algorithm that can cluster similar bidding curves and distinguish between different agent behaviors.

Chapter 3

Data Preprocessing

This chapter defines the technical strategy and decisions used in order to preprocess the data. Once the study goals are well defined, it is possible to create an initial strategy and the necessary tools to obtain results.

By detailing all the techniques, tools used and decisions made, the study and final results will be underpinned with the theoretical foundation and provide a deeper understanding of why certain tools developed are needed and what they are for. It is also intended to provide the means for future work to further develop this study, and to this end all tools developed and used for this study will be made available online at [25].

The data sources and files used in this work were obtained by data mining from publicly available information. To improve the capabilities of *ITMS* to obtain accurate results, the collected data must be preprocessed in several steps before the clustering method can be applied.



Figure 3.1: Data mining process scheme [44]

To obtain a data warehouse that contains all the necessary information for *ITMS* processing and for data validation it is first needed to select the data sources and integrate them into a complete system. Data extraction and cleaning are the first steps of pre-processing and is presented in the appendix A. This appendix is important to define which variables are used for this work and from which sources they come.

Since the raw data are untreated, they must be pre-processed to improve the chances of successfully extracting actionable information from the *ITMS*. Furthermore, the use of non-numeric variables in the *ITMS* requires additional special treatment for each individual situation. This data pre-processing is a crucial step because it can determine how good the final results will be and allows for a strategy that provides a visual representation of the results.

It is important to establish a well-structured initial strategy, as it will form the basis of the study, but also to understand that it should be adjusted as needed during the course of the study to better address any problems that may arise. This strategy must serve as a means to achieve the initial goal of this project: to find strategic behaviors in energy markets using the *ITMS* and to form clusters that will allow the identification and classification of these behaviors.

After obtaining usable data frames from data preprocessing, it is possible to apply the theoretical concepts already explained to obtain clustered events that are going to be possible representations of the strategic behaviors. In order to visually represent the obtained results, it is important to develop an approach that can represent the variables across all problem dimensions in a simple and intuitive way that provides a clear view of the results.

Moreover, in order to evaluate the information contained in the results and draw conclusions, it is necessary to validate and analyze them to verify whether or not the method is successful in discriminating between strategic behaviors.

3.1 Initial considerations and study limitations

The *MIBEL* [20] day-ahead market uses a PoolCo model and receives bids with energy quantities and corresponding prices for a given offering unit and hour of the day. Special days such as national holidays and major festivities provide special demand patterns that are difficult to analyze in clustering because they are events that do not occur as regularly, resulting in scarce available data. Therefore, for simplicity, holidays and weekends were not included in this study as they are significantly different from weekday demand.

To distinguish between strategic behaviors, it is important to keep in mind that each agent may have an extensive portfolio of units, each of which acts in a way that maximizes the agents' profit. Analyzing the strategic behavior of units may lead to more individualized and complex solutions that may not be able to accurately represent the behavior of the entire agents' portfolio.

One solution to this problem is to aggregate energy offered per price for each agent. To further reduce data complexity, prices can be discrete and have fixed values representative of the possible range of offer prices. This should improve the chances of finding representative clusters and minimize data noise.

It is also important to not consider agents whose portfolio is heavily represented by special regime units because their bids are priced at or close to zero to guarantee that non dispatchable renewable units always produce. The bids of agents with a high share of hydro and thermal assets are more interesting, as they can dispatch their production according to their most profitable strategy. In this way, the agents

In the absence of publicly available information on the marginal cost functions of units, it is difficult to define strategic behavior in terms of the price-quantity pair clusters formed by the *ITMS*, since it is not possible to extract information on whether agents are not offering at the marginal price from these results alone. To address this lack of information, other available variables related to bids and the market session are used to explain why and when clusters occur and to distinguish between them. In this way, it is expected that deeper relationships can be found in the data and possible explanatory variables can be defined.

The technical limitations of the units, e.g., ascent and descent ramps, minimum matched amount of energy, were not considered relevant for this stage of the study since the initial considered definition of behavior is the repeated pattern of price-quantity offers.

To test the capacities of *ITMS* for classification after clustering, it is interesting to divide the available data into a training set and a classification set. The training set should represent the majority of the curves and is used to create the clusters that represent the behaviour of the agents. The classification set is not used in the initial clustering process and should be used later when clusters have already been formed. In this way, it is possible to assign the unprocessed curves to the already formed clusters and test the capabilities of *ITMS* to predict new curves as known behaviours or outliers.

3.2 Data model design: Power BI

In this work, Microsoft Power BI was used to develop a data model that connects data from all sources and establishes relationships between them so that they can be further preprocessed by a Python script. Power BI is a suitable tool for this research because it is easy to use, it is free, and because several organisations already use this software, so developing a solution with this software will be more versatile for others. [30]

Since the formats of variables differ from file type to file type, it is necessary to consolidate all source data into a single data warehouse. By processing each data type individually so that they can all be correlated and used in the *ITMS*, it is possible to create a complete data warehouse linking all relevant source variables from the different files. The files were grouped by type and transformed into queries through the Extract, Transform, and Load (ETL) process. [45]

The basic principle of the ETL process is to connect to source data systems and then perform data transformations. Data transformation operations are the actions necessary to change the data from the source system to meet the requirements of the target system. During this step, it may also be necessary to perform data cleaning by removing invalid or irrelevant data from the result set. This is necessary to maintain the data quality of the solution.

The final step of the ETL process is then loading the data into the target system, in this case the data warehouse.

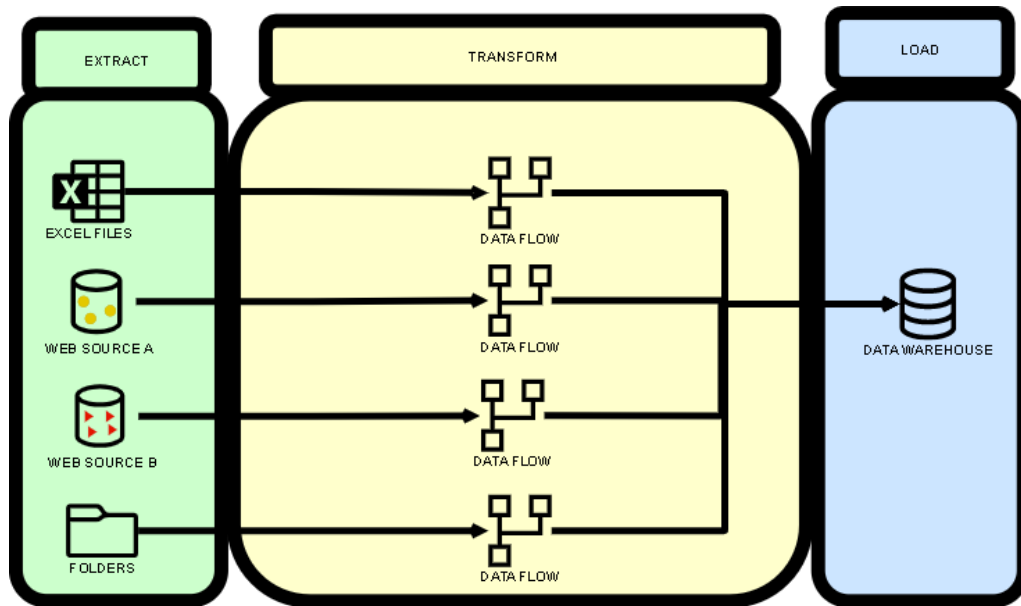


Figure 3.2: ETL process diagram

Appendix B expands on the content of the transformation process developed in *Power BI*, which is required to create the data warehouse. In this way, it is possible to provide a background for the integration and selection of relevant variables, as well as for the transformations required to build queries that will integrate the data warehouse.

After the ETL process, a BI solution is obtained. A BI solution can be briefly described as a visual projection of business operations that includes historical, current, and predictive views. [11]

To create the data model, the individual files must be stored in multiple tables that relate to each other and effectively link all the necessary information, as they have different formats and variables. A data model is simply a set of disjoint tables connected by relationships, with key variables represented by columns that share the connections between two tables. Primary keys are unique identifiers that represent each row in a table and represent it with a foreign key in another table. A well-designed data model improves the practicality of data preprocessing.

In the data model, relationships are defined by direction and cardinality. Direction is defined by setting one of the tables as the source where the value in the other table comes from, and the other table as the target where the search is performed. The cardinality of the relationship refers to the uniqueness of the key and target values. Based on cardinality alone, three different types of relationships can be distinguished:

- one-to-one
- one-to-many
- many-to-many

Optional relationships allow for no occurrences on both sides of the relationship, e.g., one-to-zero-or-one.

The process of data transformation is expanded in Appendix B, where all decisions that affect or complement the original data structures are examined in more detail. This provides a more intuitive understanding of what was done to create the full data model.

3.2.1 Complete Data Model

After establishing all the necessary relationships between all the queries, it is possible to obtain a complete database containing all the information needed to perform this study. Figure 3.3 is a class diagram showing all the relationships created.

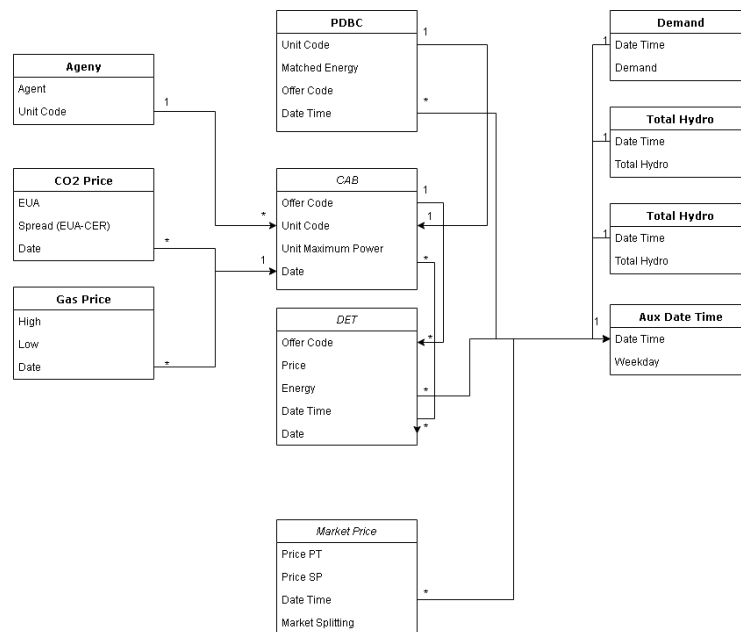


Figure 3.3: Class diagram portraying all the necessary relationships of all the queries

To conclude this section, the data model created was exported using *DAX Studio* to a csv file named *Query Tese*, where it is represented by a table containing all relevant variables according to their relationships. This file allows further preprocessing to obtain the bid curves of the agents for a given day and hour. For this purpose, a *Python script* was used that first discretizes the prices and aggregates the price-quantity pairs by agent.

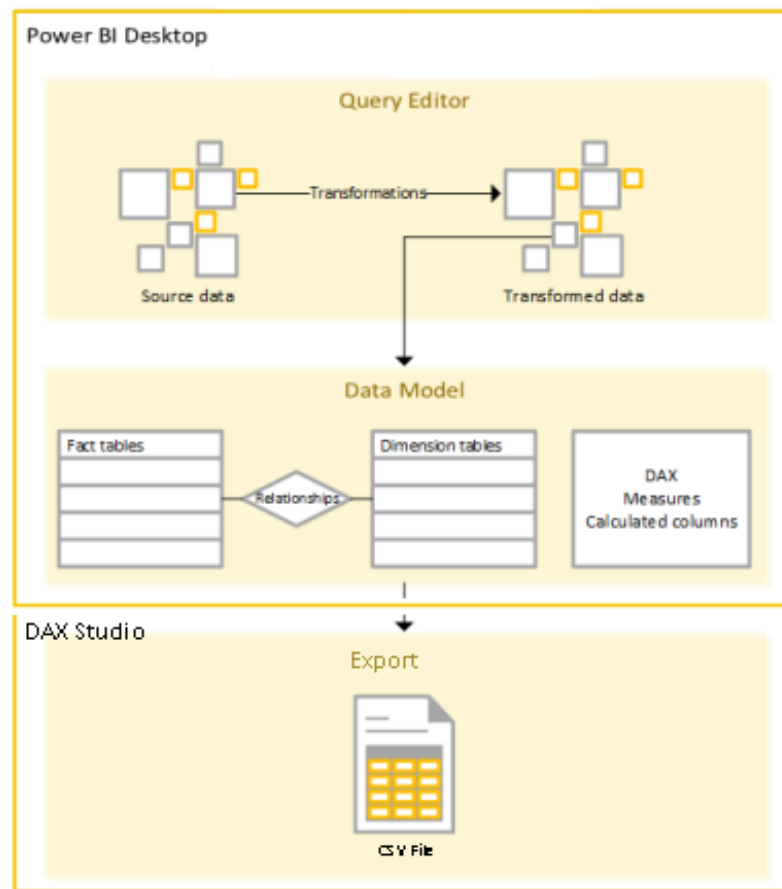


Figure 3.4: Diagram of Power BI and DAX Studio relation

3.3 Pre-Processing the Database: Python

In order to proceed with the preprocessing of the data, a data framework representing the bid curves of the agents must be created. For this purpose, a *Python* script named *Data_Pre_Processing.py* was developed to return this data frame with the aggregated energy values for the different discretized price values for a given agent, date, and time.

To improve this study, it was believed that analysing multiple scenarios would add value to the work and give a better idea of how the *ITMS* algorithm can be used in the energy market and how results vary when the *ITMS* is used with data frameworks that contain different information. To this end, 3 scenarios were developed for investigation:

- Scenario 1- Run the *ITMS* for a data frame containing the cumulative curves representing the data in a one-agent one-hour format. This scenario is expected to explain the agents behaviors for each hour. Since some behaviors might repeat themselves across multiple hours, by filtering the redundant behaviors, it is expected to obtain the relevant agent behaviors.
- Scenario 2- Run the *ITMS* for a data frame containing the aggregated curves of multiple-agents for a one-hour. On this scenario it is studied the possibility of assigning behaviors to

agents. When using the *ITMS*, if the clusters are formed separating the bidding curves by agents it proves important for this study.

- Scenario 3- Run the *ITMS* for a data frame containing the aggregated curves of a one-agent for multiple-hours. This scenario is intended to process all bidding curves of one agent in *ITMS* and discover its behaviors. It serves as a comparison to Scenario 1, as their end results should be similar.

The curves for these three scenarios were all aggregated by hour and agent, the difference between them being what the information contained in the various data frames represents. By extending the data processing capabilities of the *ITMS*, the final results should represent data that can be used to compare behaviors, distinguish outliers, and provide more consistent patterns between different agents. It also opens the possibility to determine if it is possible to assign clusters to dominant agents and if it is possible to detect patterns across hours.

The *ITMS* algorithm is used to cluster the aggregate energy bid curves for discretized prices. For this purpose, it is necessary to separate the ['Energy'] and ['Price'] columns from the other special variables to obtain a data frame that can be used in the algorithm. The variables used in *ITMS* must continue pre-processing until a new data frame is obtained containing the cumulative curves of discrete price and energy for a given date and time.

In the *Data Pre Processing.py* file it was created a *Data_Pre_Processing* class that contains all the necessary functions to finish the preprocessing. To supplement this information, please refer to the Appendix C Section C.1 to see how the class was initialized.

This class is initialized by reading the csv file created by exporting the data model developed in the last section to *DAX Studio*. Using the values in the ['Price'] column, it is possible to create price intervals to be used in the discretization of the variable.

The *ITMS* algorithm is used to cluster the aggregate energy bid curves for discretized prices. For this purpose, it is necessary to separate the ['Energy'] and ['Price'] columns from the other special variables to obtain a data frame that can be used in the algorithm. The variables used in *ITMS* must continue preprocessing until a new data frame is obtained containing the cumulative curves of discrete price and energy for a given date and time.(see Appendix C Section C.2 for the code application)

Price discretization was done to smooth the data and reduce noise introduced by small changes in markets or bids that may not be relevant to the solution. Although this process may involve a loss of detail as the shape of the curves is changed, it benefits the clustering process by reducing the complexity of the data set and increasing the likelihood of finding representative clusters. This is the case because stationary price dimensions that retain values between different agents and offers do not change the dimensions of the data frame, making it easier to draw inferences between them through standardized testing.

The stationary price values are defined to represent the range of offers in enough detail that one can still draw conclusions from them. The values are obtained by a quantile-based discretization of the bids and are defined as cut points, which are the boundaries that divide a probability distribution

into continuous intervals with equal probabilities. In this scenario, the ['Price'] values are sorted in ascending order and divided into intervals with an equal number of observations. The boundaries of the intervals are used as stationary values.

The number of stationary values representing the range is a parameter that can be chosen and will have a stationary value of 13. This value was chosen because it is thought to represent the range with enough detail that the curves formed with these values will still represent important information about the agents bidding behaviors. To demonstrate how this discretization works, the following example is given. For an made-up set of 12 price-quantity quotations $PE = \{(P_0, E_0), \dots, (P_{12}, E_{12})\}$ present at the table 3.1:

Price (€/MWh)	Energy (MWh)
1	20
5	50
44	88
12	32
7	17
3	5
21	10
55	74
31	20
180	100
1	26
47	11

Table 3.1: Example dataset PE

To convert the ['Price'] range into 4 discretized stationary values, 3 buckets of equal size must be created. Sorting the values in ascending order results in the following buckets:

Price (€/MWh)
1
1
3
5

Price (€/MWh)
7
12
21
31

Price (€/MWh)
44
47
55
180

Table 3.2: 3 equal-sized ['Price'] buckets

The new stationary ['Price'] values are [1,6,37.5,180]. The values were obtained by calculating the mean values between the buckets boundaries, so that it is possible to create buckets with as uniform a number of quotations as possible.

The equations 3.1 and 3.2 define how the offered energy was assigned to the new stationary values. Since the price offered is always between two stationary values, called upper and lower boundaries, the quantity is divided into weighted parts depending on the difference between the price offered and the boundaries.

For a offered energy quantity E , at price P , the weighted energy quantity assigned to the lower boundary, E_L :

$$E_L = (1 - (Price - Lower Limit) / (Upper Limit - Lower Limit)) * E \quad (3.1)$$

And the energy quantity assigned to the upper limit, E_U :

$$E_U = (Price - Lower Limit) / (Upper Limit - Lower Limit) * E \quad (3.2)$$

Price (€/MWh)	Energy (MWh)
1	$E_L = 20 + 26 + (1 - \frac{3-1}{6-1}) \cdot 5 - (1 - \frac{5-1}{6-1}) \cdot 50$
6	$E_U = \frac{5-1}{6-1} \cdot 50 + \frac{3-1}{6-1} \cdot 5$

Price (€/MWh)	Energy (MWh)
6	$E_L = (1 - \frac{7-6}{37.5-6}) \cdot 17 - (1 - \frac{12-6}{37.5-6}) \cdot 32 - (1 - \frac{21-6}{37.5-6}) \cdot 10 - (1 - \frac{31-6}{37.5-6}) \cdot 20$
37.5	$E_U = \frac{31-6}{37.5-6} \cdot 20 + \frac{7-6}{37.5-6} \cdot 17 + \frac{12-6}{37.5-6} \cdot 32 + \frac{21-6}{37.5-6} \cdot 10$

Price (€/MWh)	Energy (MWh)
44	$E_L = (1 - \frac{44-37.5}{180-37.5}) \cdot 88 - (1 - \frac{47-37.5}{180-37.5}) \cdot 11 - (1 - \frac{55-37.5}{180-37.5}) \cdot 74$
180	$E_U = 100 + \frac{44-37.5}{180-37.5} \cdot 88 + \frac{47-37.5}{180-37.5} \cdot 11 + \frac{55-37.5}{180-37.5} \cdot 74$

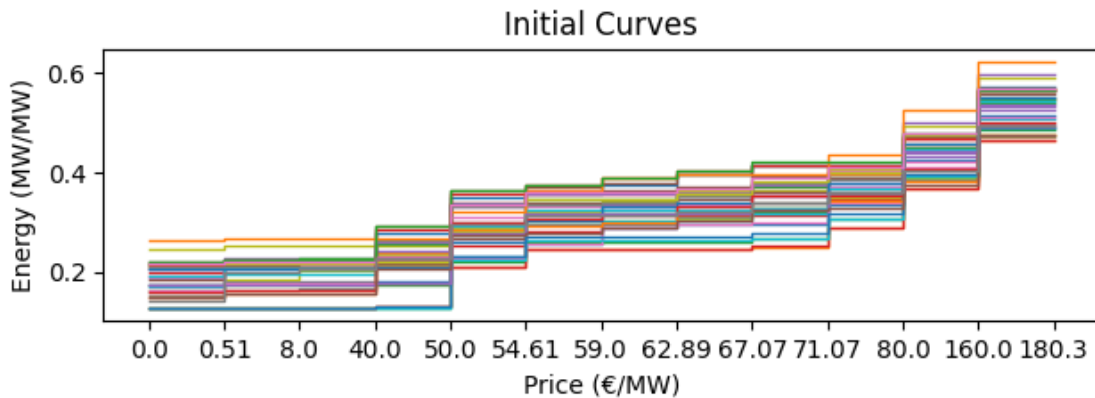
Table 3.3: Discretized prices with matching energy offers

In the *Process* function the discretization is done by the code presented at Appendix C Section C.3

After calculating the price-offer pairs, the energy values are added up cumulatively to represent the agent's supply curve. (for the code application refer to Appendix C Section C.4

The function developed so far returns a dataframe for individual agent and hour with the corresponding cumulative curves for the defined time period. Tables D.1 and D.2 in appendix D are used to illustrate the final data frames created for the selected agents, *ENDESA GENERACION, S.A* and *IBERDROLA GENERACION ESPANA S.A*, respectively, and represent all the curves present in the files for the period from 01/01/2019 to 28/02/2019 at the hour 09:00. In these tables, the energy quantities were reduced to 4 decimal places to better visualize the data.

Through this work, the data will be graphically represented by step-wise curves and heatmaps. For the e.g. it is possible to represent the curves from tables D.1 and D.2 as shown:

Figure 3.5: Example step-wise curves for *ENDESA GENERACION, S.A* at 09:00

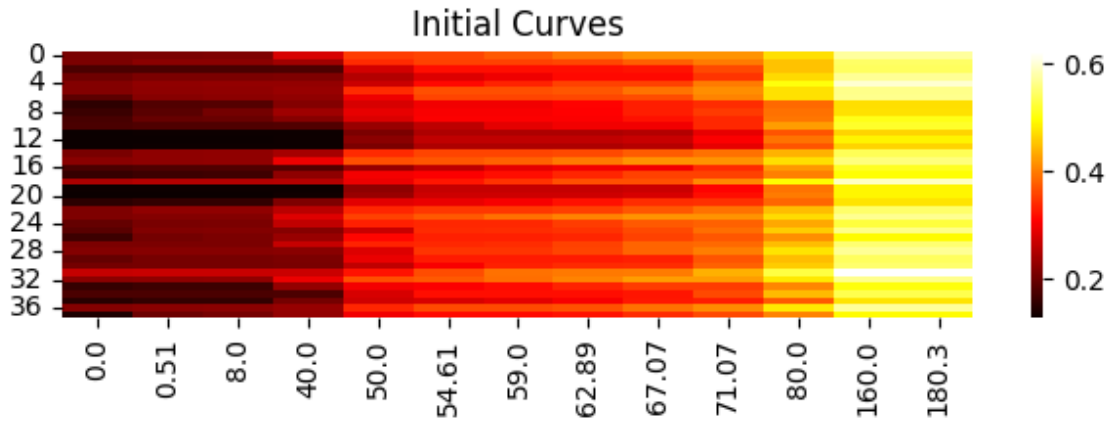


Figure 3.6: Heatmap for *ENDESA GENERACION, S.A* at 09:00

As mentioned earlier, three different scenarios were considered for this project. All scenarios share the same curve formation process: for a defined period of time the curves are formed by cumulatively summing the quantity offered for a certain discretized price, considering a particular agent and particular hour.

In scenario 1, the *ITMS* is run for the one-hour cumulative curves of a given agent. The results of *ITMS* are one-hour clusters for the different hours, which can be concatenated into a multi-hour data frame. This allows the analysis of the behavior of individual agents over different hours. Scenario 1 uses the functions *process* output for one-hour one-agent to run each iteration of the *ITMS*

For Scenario 2, the *ITMS* is run for one-hour cumulative curves of multiple- agents. These data frames are obtained by grouping the respective one-hour one-agent data frames for the agents under consideration. This approach allows to study the formation of clusters between agents and to detect behavioral similarities.

For Scenario 3, the one-hour cumulative curves for an agent created by the *process* function are grouped into a multi-hour data frame for an agent. By running these data frames for the selected agents with *ITMS*, it is possible to examine agent behavior over multiple time periods.

The code developed for the last two scenarios is shown at Appendix C Sections C.5 and C.6 respectively.

3.4 Normalization

In statistics, the most common definition of normalization is the process of adjusting values measured at different scales to a common scale, often before their aggregation or averaging. There are many other definitions, depending on the context or area of study [51]

In this work, the normalization techniques focus on the field of machine learning and can be considered as transforming raw data to obtain numerical and comparable criteria by using a

common scale with an interval of [0-1]. Several techniques have been used for this normalization, which are discussed in more detail in this chapter [29], [26], [47].

Data normalization is an important step in the data preprocessing phase, especially for distance-based methods such as *ITMS*. Without normalization, these methods tend to weight more variables with larger ranges, which can reduce the coherence and accuracy of the results.

In order to preserve the cyclic nature of the time and date variables, a simple min-max normalization would not smoothly map the continuity between cycles because it would simply show a gap that is not representative of its cyclic nature. To this end, it was used a normalization method that uses trigonometric expressions and represents each cyclic variable in two dimensions. For a given variable X with a period P , it is possible to calculate each new dimension X_1, X_2 with the range [0,1] by formula 3.3:

To preserve the cyclic nature of the time and date variables, the normalization method uses trigonometric expressions for these variables and represents each cyclic variable in two dimensions. For a given variable X with a period P , it is possible to calculate each new dimension X_1, X_2 with the range [0,1] by the formula 3.3:

$$\begin{cases} X_1 = 0.5 \cdot (1 + \sin(\frac{2 \cdot \pi \cdot X}{P})) \\ X_2 = 0.5 \cdot (1 + \sin(\frac{2 \cdot \pi \cdot X}{P})) \end{cases} \quad (3.3)$$

As can be seen in Figure 3.7, this plot describes the periodicity of the function more accurately than a stepwise plot because it smooths out the decay at the end of the cycle.

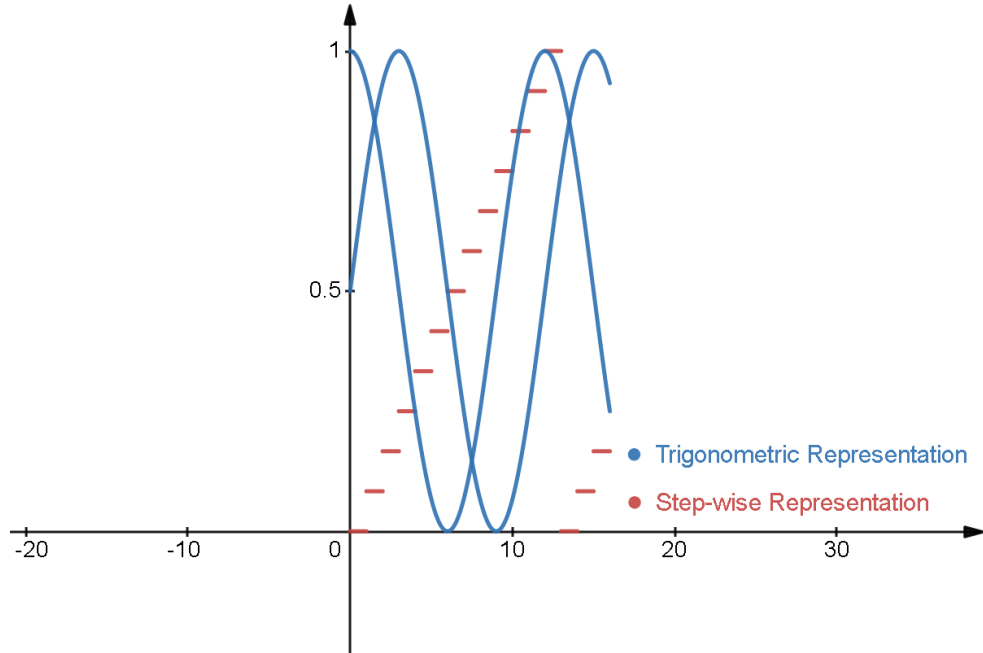


Figure 3.7: Trigonometric and step-wise representations

For all other special variables, a min-max normalization was used. In this method, the minimum and maximum are converted to 0 and 1, respectively, effectively reducing the range of

characteristics to [0,1]. In this method, an x variable can be scaled by the following equation:

$$x_{scaled} = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (3.4)$$

Given the difficulty of standardizing the amount of energy across all dimensions (also known as the z-score) and maintaining the shape of the curves, it was decided to perform normalization for all variables. This scaling only reduces the ranges of the variables and remains sensitive to outliers that are very different from the rest of the data, since it only compresses a poorly fitting range that was never representative of the variable.

To normalize the offered energy across all price dimensions, it is necessary to find parameters that span the different dimensions. The range of energy offered was set between 0, meaning that no quantity is offered for that price, and to the agent's maximum capacity for the offering units. By using these values, it is possible to reduce the range to [0,1]. A min-max normalizer that exploits the defined range having the smallest possible value of zero, effectively only a max-normalization.

$$x_{scaled} = \frac{x}{x_{max}} \quad (3.5)$$

For a given agent A , one can calculate its maximum capacity by summing the maximum capacities of the N offering units owned by this agent.

$$Cap_A = \sum_{i=1}^N Cap^N, \text{ for } N \text{ units} \quad (3.6)$$

After calculating the maximum agent capacity, it is possible to normalize the energy values across the 13 price dimensions x :

$$x_{i \text{ scaled}} = \frac{x_i}{Cap_A}, \text{ for } i \in [1, 13] \quad (3.7)$$

Chapter 4

Data clustering: ITMS application

4.1 ITMS in energy markets- A new approach

As mentioned in Chapter 2.2, *ITMS* is a clustering algorithm used to process the price-energy pair bids prepared in the pre-processing phase. The data frame is represented by a matrix x^0 of dimensions $NC \times 13$, with 13 columns representing the discretized prices and NC rows representing the cumulative bidding curves for a single agent and single hour.

$$x^0 = \begin{bmatrix} x_{1,1} & \dots & x_{1,13} \\ \vdots & \ddots & \vdots \\ x_{NC,1} & \dots & x_{NC,13} \end{bmatrix} \quad (4.1)$$

The data frame defined as x^0 (iteration $\tau = 0$) represents the initial curves obtained in the pre-processing phase. The pre-processed results for the three considered scenarios are going to be different giving rise to different x^0 . To determine the updated curves for iteration $\tau = 1$, the equation 2.18 is used. By iterating, it is possible to shift the curves according to their centroids and initial shape to obtain clusters. The clusters are groups of initial curves which, by shifting their values in the *ITMS*, lead to a similar new curve representing a behavior.

By treating each row as an array representing the energy offered across all discretized price dimensions, it is possible to determine the distance between x_i and x_j . The Euclidean distance can then be calculated for an array with dimensions $D = 13$ as in equation 4.2: [46]

$$t = \left\| \frac{x_i - x_j}{\sqrt{2}\sigma} \right\| = \sqrt{\sum_{D=1}^{13} \frac{(x_i^D - x_j^D)^2}{2\sigma^2}} \quad (4.2)$$

The distances are used as input to a Gaussian kernel $K(t)$ 2.2, which is used to determine $V(x)$, $V(x; x_o)$, and in the update equation, shown respectively in 2.7, 2.11, and 2.18.

The choice of the parameters σ and λ is a precondition that must be set before starting the clustering process. After this step, it is possible to run the *ITMS* and update the data frame as long

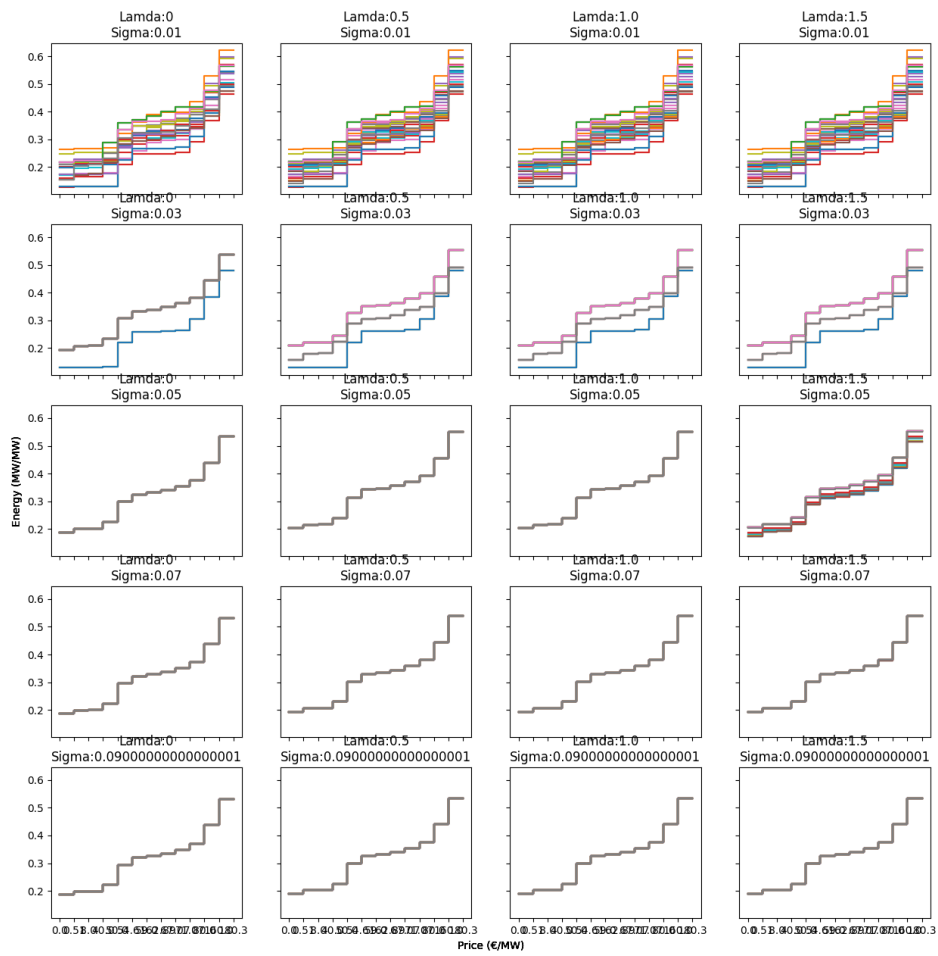
as the following conditions are met:

$$stop\ criteria = \begin{cases} \max(\|x_i^\tau - x_i^{\tau+1}\|, i \in NC) > 0.000001 \\ \tau < 50 \end{cases} \quad (4.3)$$

The maximum Euclidean distance between x_i^τ and $x_i^{\tau+1}$ is defined to stop the algorithm if the curves do not update significantly during an iteration, and to prevent the algorithm from running indefinitely, another stopping criteria was added to stop the algorithm when it reaches the 50th iteration.

To apply this algorithm, a *Python script* was created with an *ITMS* class that contains all the necessary functions for data processing. In the *Python script*, the update equation is computed by the function *update*. The code for this function is present at Appendix E Section E.1

Determining the parameters λ and σ is the final requirement to run the *ITMS* and effectively process the curves. Several combinations of these values were tested to determine how they would affect the clustering process and its ability to process the data in a meaningful way.

Figure 4.1: Final curves using different λ and σ parameters for ENDESA GENERACION, S.A

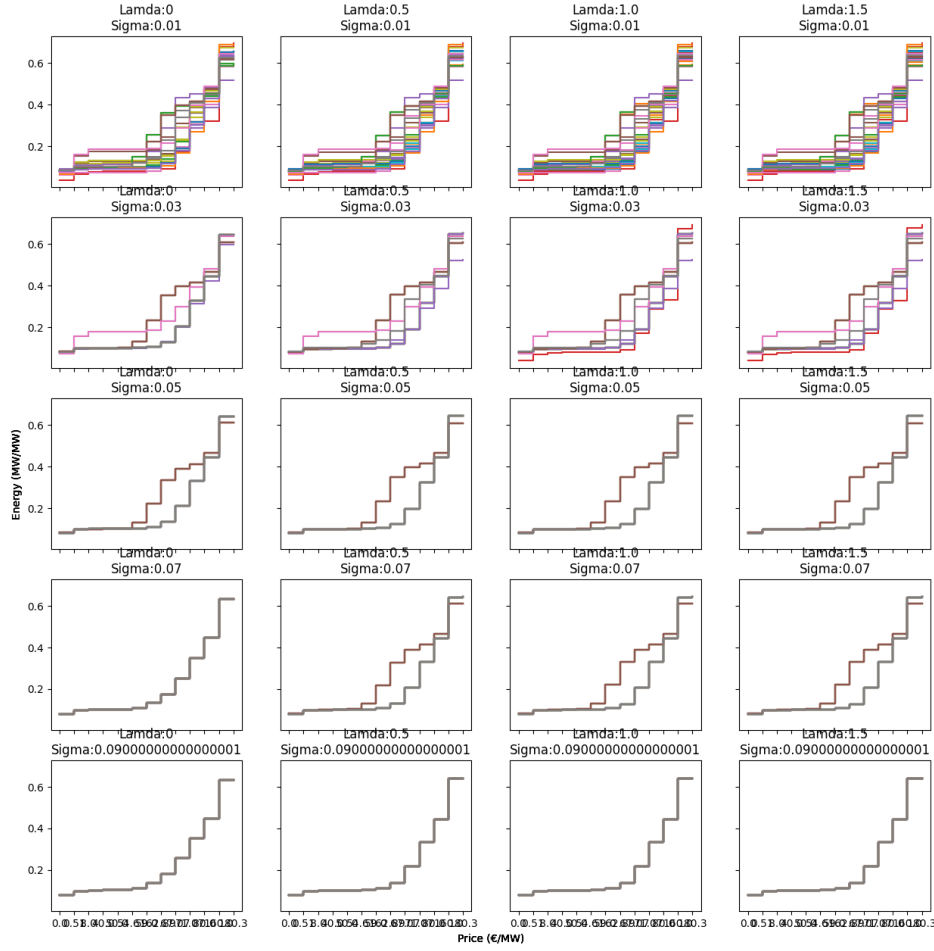


Figure 4.2: Final curves using different λ and σ parameters for IBERDROLA GENERACION ESPANA S.A

For the sample curves shown in the Tables D.1 and D.2 in the Appendix D, multiple combination of parameters was tested using the *ITMS* method for both agents, and the clusters formed are shown in Figures 4.1 and 4.2. In this way, each graph represents a clustering process using the *ITMS* and allows to understand how each parameter affects the process, through a graphical analysis. Looking at the influence of the λ parameter on the solution, we can see that for results with the same σ , more distinct clusters form as the *lambda* value increases. This is due to the original shape of the data set gaining more weight on the clustering process with increasing λ values. When analysing the results for both agents, it can be concluded that the best value for λ is 1.5, as it successfully clusters the curves into different patterns, gives more weight to the initial curves, and produces solutions that are more representative of the original data set, as was intended for this study.

As for the σ parameter, at a value of 0.01, most curves do not update sufficiently to form significant clusters thus performing poorly in the clustering process because it retains too much of the x^0 shape. On the other hand, solutions where $\sigma > 0.05$ tend to align all curves to the distribution centroid, representing oversimplified solutions that are not very useful for this case study. In light of this, it was decided that σ values should fall in a $[0.03, 0.05]$ range, with the chosen value being 0.04. For simplicity, these parameters are universal to all agents in this work. Better results could be obtained by using more complex automated methods to determine these parameters. This will be addressed in another subsection 6.3, where further suggestions for possible future work will be made.

The chosen parameters are as follows:

$$\begin{cases} \sigma = 0.04 \\ \lambda = 1.5 \end{cases} \quad (4.4)$$

To calculate these parameters, the *findbestparameters()* function was created to iteratively test all combinations of 0.5 λ -steps for $\lambda \in [0, 1.5]$ and 0.03 σ -steps for $\sigma \in [0.1, 0.09]$. (see code at Appendix E Section E.2)

For the data frame used to define the best parameters it is possible to depict the initial and final processed curves by Figures 4.3 and 4.4 for the *ENDESA GENERACION, S.A* agent and *IBERDROLA GENERACION ESPANA S.A* at hour 09:00.

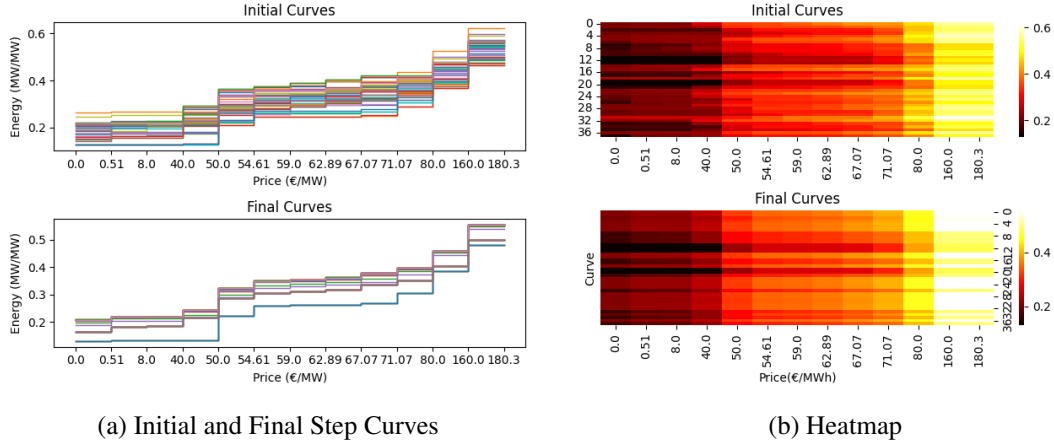


Figure 4.3: Results for single agent and hour scenario for *ENDESA GENERACION, S.A*

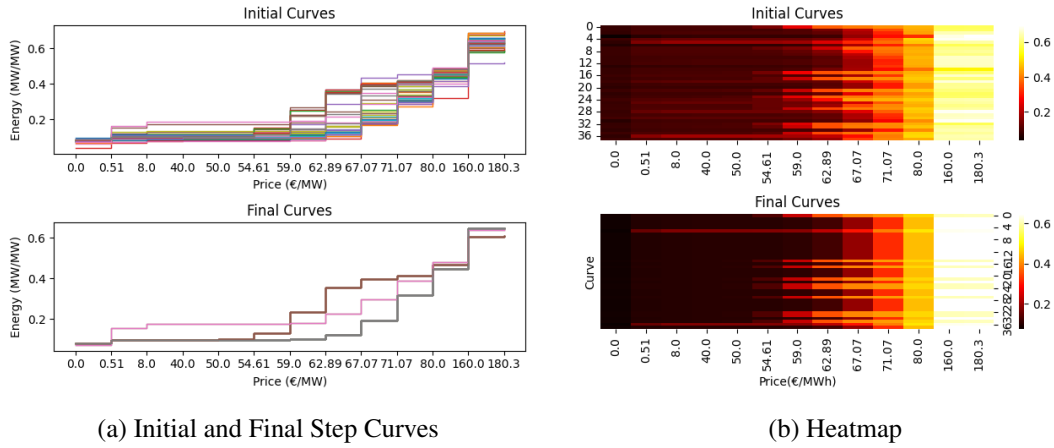


Figure 4.4: Results for single agent and hour scenario for *IBERDROLA GENERACION ESPANA S.A*

4.2 Results validation

After running the *ITMS* and obtaining clusters for the hourly bidding curves that might explain the agents' behaviors, it is important to find a way to validate the results. Since there is no public information about units marginal costs, it can be considered incorrect to define strategic behaviors by only analyzing the formed clusters. The results obtained so far refer only to quantity-price offers and represent a highly simplified solution that may not be representative of all variables involved in the process of creating a market offer. These possible explanatory variables can range from the date and time of the offer to external factors that could affect marginal production costs. The inclusion of these variables is intended to provide a better understanding of what may have led to the behavioral changes and a possible validation of the results obtained.

The validation of the results is a crucial step that allows to carry out a more in-depth study that could find explanatory scenarios and compensate for the lack of information on marginal costs.

This study aims to investigate the classification potential of the *ITMS* method by modifying the algorithm. In doing so, it is expected that new unprocessed curves can be assigned to already defined clusters or identified as new behaviors that have never been seen before. Various ways to graphically represent the data will also be explored to extract the most information from the results. Statistical analysis of the results will also be developed to examine the repetition of the behaviors.

Finally, the importance of the features in forming each cluster is analyzed using *ITL* concepts.

4.2.1 Representing the clusters

Once the clusters for each scenario have been identified, an important step is to find a way to represent the clusters by a single representative curve. This representation simplifies their comparison and makes it more intuitive. To this end, it was first used the centroid as the representative of a cluster. By calculating the Euclidean distance between multi-hour curves and using the maximum

limit defined in the subsection 4.2.1.1 it is possible to determine whether the curve belongs to a cluster or, if it does not converge with other curves, is an outlier.

The approach defined in 4.2.1.1 is a solution where clusters are interpreted as different if they are far apart. This method completely ignores the concept of clusters as curves that share similar information among themselves. The results obtained may not be representative of the pdf and may therefore give incorrect results.

For this purpose, an alternative method has been developed and presented in 4.2.1.2. In this method, the *ITMS* method is applied to all hourly clusters to obtain multi-hour clusters. In this way, it is possible to better control the clustering process through the *ITMS* parameters and get a better insight into the pdf.

4.2.1.1 Euclidean Distance

To display the clusters, a multi-hour data frame was created to reduce the clusters to a representative curve for each distinct cluster. This way it is possible to analyze all the hourly clustered curves and by measuring the euclidean distance between all curves it was possible to reduce the data frame dimensions. For x_i distinct clusters present in a data frame with N clustered curves, it is possible to define that a curve x_j is included in cluster x_i if the distance d_{ij} between x_i and x_j is less than 0.07. This value was selected after a trial-and-error experiment and was chosen as a suitable candidate.

$$\begin{cases} d_{ij} = \|x_i - x_j\| < 0.07, i \in [0, C] \cap j \in [0, N] \\ d_{ij} < 0.07 \end{cases} \quad (4.5)$$

This distance was chosen to include clusters that may not have fully converged during the *ITMS* process.

To do this operation it was created a *Python* function named *defineclusters* that will cycle through the processed curves and return the processed dataframe with a new column ['Cluster'] that shows the belonging x_i cluster to x_j curve.

This way it is possible to represent the distinct cluster by an heatmap that allows the comparison between them.

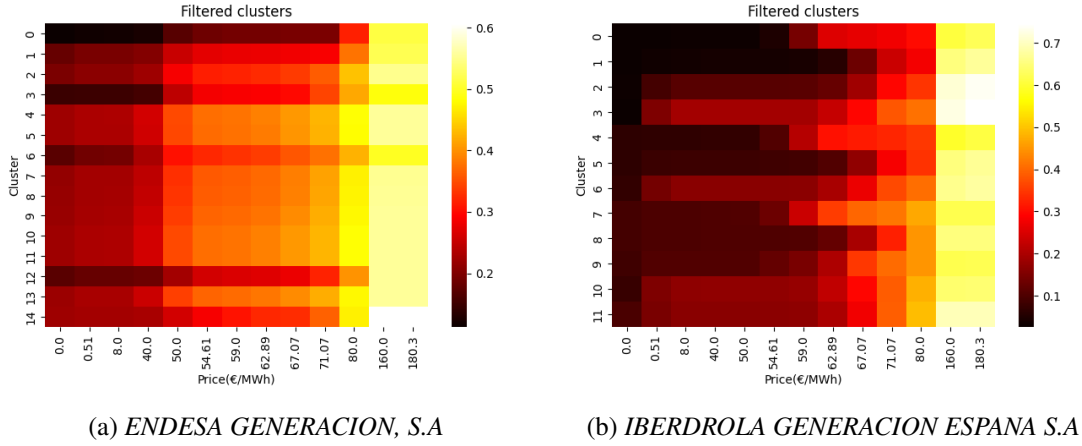


Figure 4.5: Minimized clusters for single-agent and single-hour scenario using *ENDESA GENERACION, S.A* and *IBERDROLA GENERACION ESPANA S.A*

4.2.1.2 Minimizing with ITMS

The hourly processing of the clusters means that the true pdf of the bidding curves is broken by hour. The clusters formed during this process represent the local maxima of the pdf for that hour. These values may repeat across hours and result in redundant clusters. The redundant clusters can be analyzed as observations that reduce the entropy of the set. These curves do not have much value because they do not represent new relevant information for the set. By using *ITMS* to clean the data, it is possible to increase the entropy of the dataset and ensure that any new clusters represent the distribution centroids of the clustered curves.

In Scenario 1, *ITMS* is run on the preprocessed curves hourly for a single agent. By merging the hourly clusters into one data frame, obtaining a full multi-hour data frame, and running *ITMS* again, it is possible to process obtain multi-hour clusters as the ones obtained in Scenario 3, with less computational effort.

This allows a comparison of the two results from Scenario 1 and Scenario 3 in terms of benefits and knowledge extraction.

The results obtained in *ITMS* using this method can be seen in Figure 4.6

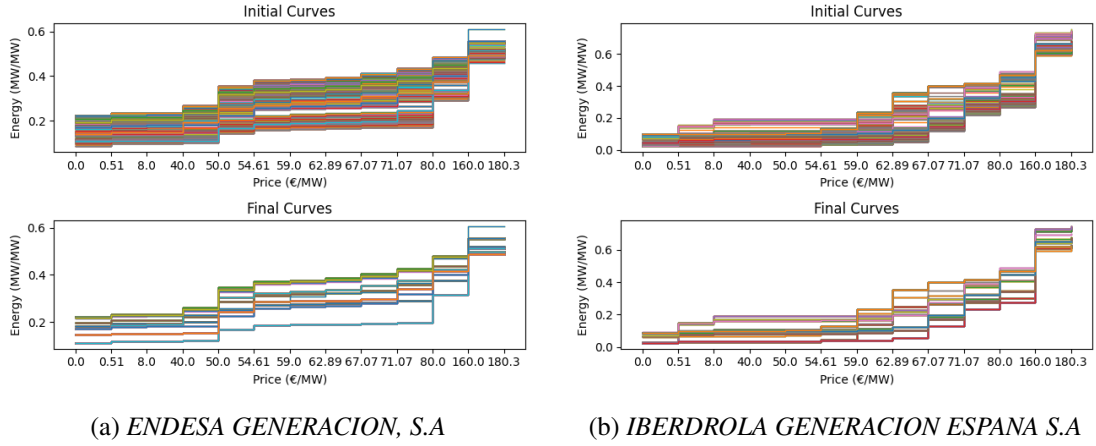


Figure 4.6: Results obtained in the *ITMS* method for single-agent and single-hour scenario using *ENDESA GENERACION, S.A* and *IBERDROLA GENERACION ESPANA S.A*

The cluster representation can be seen in Figure 4.7

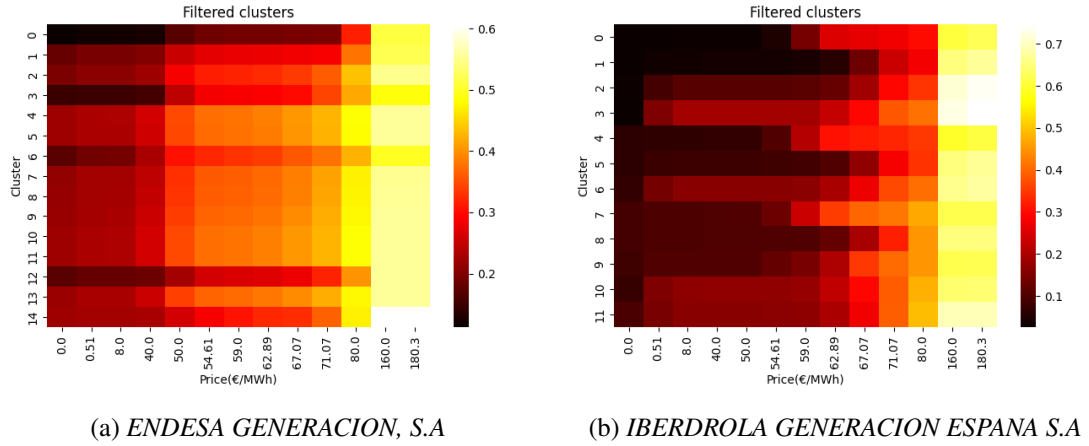


Figure 4.7: Minimized clusters for single-agent and single-hour scenario using *ENDESA GENERACION, S.A* and *IBERDROLA GENERACION ESPANA S.A*

Analyzing Figure 4.7 for *ENDESA GENERACION, S.A*, it is possible to note that for prices below 40€/MWh, clusters C_{E-0} , C_{E-3} , C_{E-12} show the most similarities among each other with a small amount of energy offered, less than 10% of the agent's production capacity. The remaining clusters all exhibit similar amounts of supply, with more quantity being offered, representing approximately 25% of the agent's production capacity.

For prices between 50€/MWh and 80€/MWh, clusters C_{E-4} , C_{E-5} , C_{E-7} - C_{E-11} , C_{E-13} show very similar offer quantities, with quantities up to 50% of the agents capacity being offered. The remaining clusters all have lower offered quantities, with the lowest offered quantity found in cluster C_{E-0} , where the offered quantity does not exceed 30% of agent capacity.

For high prices above 160€/MWh all clusters present identical quantities varying between 55% and 60% of the agents capacity.

Through a graphical analysis of the Figure 4.7 for *IBERDROLA GENERACION ESPANA S.A.*, it was possible to visually capture some of the key differences between the clusters. To this end, the following conclusions were drawn:

- For cluster C_{I0} , at prices of 54.61€/MWh or less, the quantity offered is very small and corresponds to less than 10% of the agents' production capacity. The quantity offered increases at 59€/MWh and 62.89€/MWh to about 30% of the agents' capacity and remains the same until 80€/MWh. At 160€/MWh, the quantity offered increases further to about 70% of the agent's capacity. With this behaviour, the prices of 59€/MWh and 62.89€/MWh are of particular interest, as these are the lowest offered values at which the agent shows interest in significantly increasing the amount of energy.
- For cluster C_{I1} , at prices of 62.89€/MWh or below, the quantity offered remains very small and corresponds to less than 10% of the agent's production capacity. At higher prices, the quantity is gradually increased until it reaches about 30% of the agent's capacity at 80€/MWh. In this behaviour, it can be observed that agent is not willing to offer at prices lower than 62.89€/MWh. Here we can observe that the prices 67,07€/MWh and 71,07€/MWh are decisive for this strategy.
- For cluster C_{I-2} , there is an increase in offered energy to over 10% of the agents' production capacity at the price of 0.51€/MWh and another increase at the price of 8€/MWh. There are new offers that start at 59€/MWh and gradually increase the offered energy to about 30% of the agent's capacity when the price reaches 80€/MWh. This behaviour shows the tendency of the agent to offer a small part of its capacity at low prices and gradually increase this share at prices above 62,89€/MWh.
- Cluster C_{I-3} has a similar shape to cluster C_{I-2} . It is possible to distinguish this cluster by a larger increase in the quantity offered for 0.51€/MWh, which causes a positive shift in the quantity axis compared to cluster C_{I2} .
- For cluster C_{I-4} , the quantity offered remains very low until the price reaches 50€/MWh. At prices 54,61€/MWh and 59€/MWh, the quantity offered increases twice. Up to 80€/MWh there are no more offers, and only at a price of 160€/MWh there is a strong increase.
- Cluster C_{I-5} shows very low supply at prices of 62€/MWh or below, with a gradual increase up to 80€/MWh to about 30% of the agents production capacity.
- The cluster C_{I-6} has a very similar shape to the cluster C_{I-3} and mostly differs only by a lower offered quantity starting from 160€/MWh.
- Cluster C_{I-7} has a low quantity offer for prices below 50€/MWh of about 10% of the agent's production capacity. From a price of 54.61€/MWh, it gradually increases until it reaches almost 50% of the agent's capacity at a price of 80€/MWh and increases to almost 60% of the agent's capacity at prices above 160€/MWh.

- Cluster C_{I-8} shows a matching pattern of offered energy as cluster C_{I-5} for prices below 80€/MWh. At this price point, the quantity offered is higher than in cluster C_{I-5} and rises above the 40% of the agent's production capacity.
- Cluster C_{I-9} shows values close to the energy offered in cluster C_{I-7} for prices below 59€/MWh. At this price point, it differs from cluster C_{I-7} by a slightly lower quantity offered.
- Cluster C_{I-10} shows identical supply quantities as in cluster C_{I-3} for prices up to 71.07€/MWh. At 80€/MWh, the offered energy increases to about 50% of the agent output, which is more than in C_{I-3} . At prices of 160€/MWh and 180.3€/MWh, it can be seen that the offered energy is slightly lower than in cluster C_{I-3} .
- Cluster C_{I-11} is very identical to cluster C_{I-10}

4.2.2 Curve Classification

The goal of this approach is to understand if it is possible to use the formed clusters to classify new curves by allocating them to one of the existing clusters, or to consider them as an outlier. This allows to update the known behaviors with new curves without having a computational intensive process of running the *ITMS* all over again for all curves. To obtain these new curves the *ITMS* only processes 90% of the pre-processed dataframe, and does not alter the other 10%. The processed curves for all hours were used together with the unprocessed curves in order to keep the most information about the clusters that were formed.

From equation 2.18 it is possible to represent the new x^0 by:

$$\begin{cases} x^0 = [x_0^0, x_1^0, x_2^0, \dots, x_j] \\ j \in [0, PC + NC] \end{cases} \quad (4.6)$$

Variable PC represents the number of processed curves and NC the number of new curves. By modifying the *ITMS* in order to only update the new curves it is possible to do the classification. The new update equation is:

$$x_i^{\tau+1} = \frac{c_1 \sum_{j=1}^n G\left(\left\|\frac{x_i^{\tau}-x_j^{\tau}}{\sqrt{2}\sigma}\right\|^2\right) x_j^{\tau} + c_2 \sum_{j=1}^n G\left(\left\|\frac{x_i^{\tau}-x_{Oj}}{\sqrt{2}\sigma}\right\|^2\right) x_{Oj}}{c_1 \sum_{j=1}^n G\left(\left\|\frac{x_i^{\tau}-x_j^{\tau}}{\sqrt{2}\sigma}\right\|^2\right) + c_2 \sum_{j=1}^n G\left(\left\|\frac{x_i^{\tau}-x_{Oj}}{\sqrt{2}\sigma}\right\|^2\right)}, \quad i \in [PC, PC + NC] \quad (4.7)$$

To this end the *classifyclusters* function was developed in order to apply this modified version of the *ITMS*. The function returns x^0 and the processed dataframe.

The results obtained for the single-agent and single-hour are depicted in Figures 4.8 and 4.9

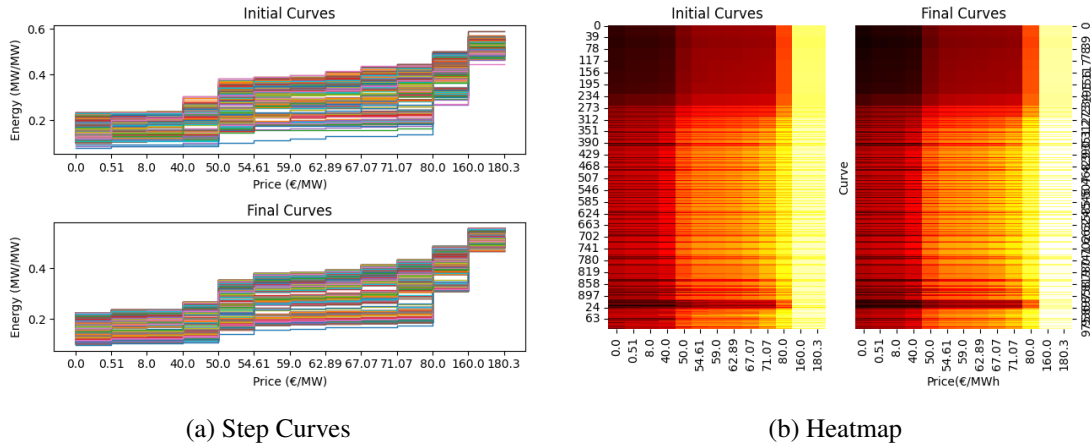


Figure 4.8: New curves classification for *ENDESA GENERACION, S.A* for single agent and hour scenario

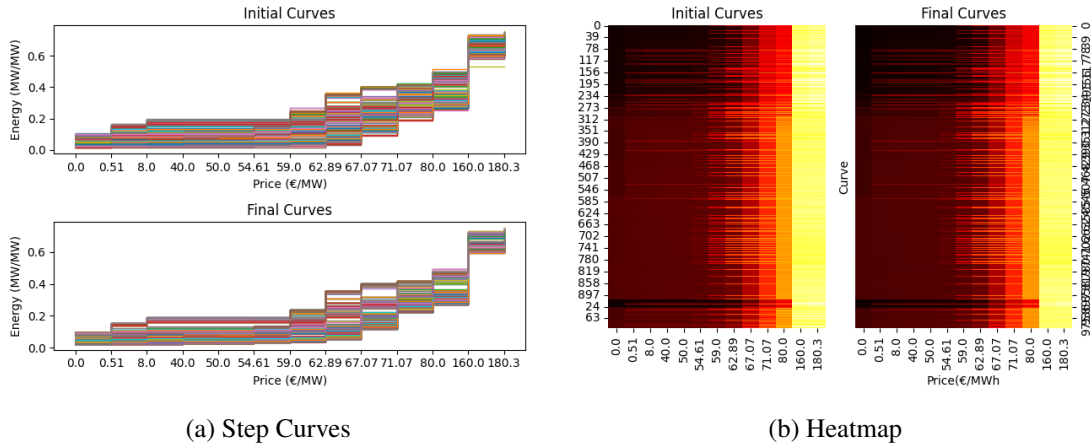


Figure 4.9: New curves classification for *IBERDROLA GENERACION ESPANA S.A* for single agent and hour scenario

To facilitate the analysis it was used the method defined at 4.2.1.2 to minimize the distinct clusters and define to what cluster each curve belongs. The results obtained from this operation can be seen in Figure 4.10

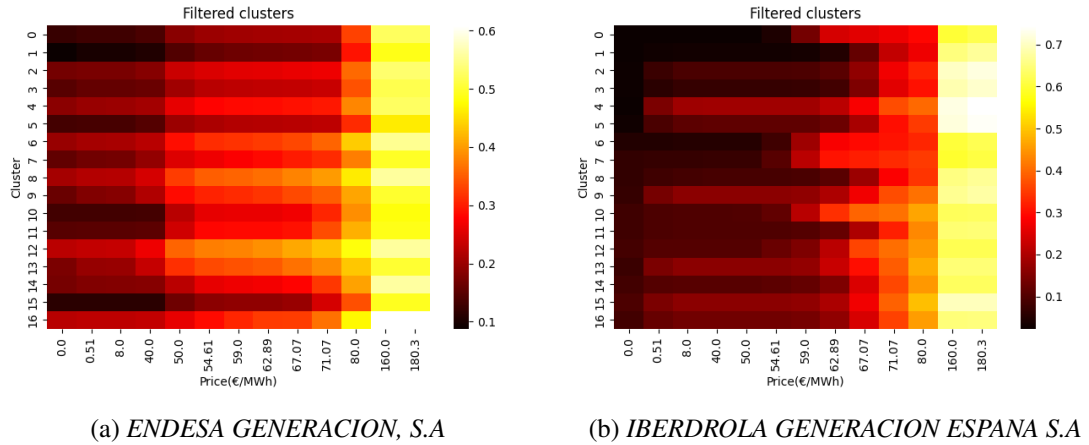


Figure 4.10: Minimized clusters with new classified curves.

4.2.3 Feature Selection

Traditional statistics is strong in developing methods for describing data and deriving distribution parameters from samples. In most cases, the information contained in the data is not obvious, so analysis must be performed. By using the 1emphITMS it was possible to obtain methods for classifying curves as behaviors, and thus decision makers can analyze a situation and adapt their strategies according to this new knowledge. Although this knowledge only defines what the behavior is like, it is the basis for understanding how agents formulate their strategies. Put simply, the work to this point answers the simple question "How do agents behave?" and explores the methods that best define those behaviors.

The next step would be to try to understand why the behavior occurs. This step can prove to be more complex, as causal inferences about a phenomenon are usually very complex and results can vary greatly depending on the method used. Due to the complexity of this type of study, it was decided to first analyze which variables might be most strongly related to a behavior.

By using *ITL* concepts, it is possible to formulate relational knowledge through numerous mathematical mechanisms. Combining the clustered curves with external variables opens the possibility to draw new relational inferences about a phenomenon.

Feature selection plays a fundamental role in pattern recognition and more generally in machine learning tasks [10], such as visualization, classification, recognition, and verification. Feature selection algorithms can be divided into two methods: filters and wrappers. [17]

The filter methods selects a subset of features in the pre-processing step, independent of the learning algorithm. This allows the subset to not inherit biases from a learning algorithm.

The wrapper methods search for an optimal feature subset tailored to a specific algorithm and domain, which assume a given learning algorithm and use its performance to evaluate and determine the selected features.

Firstly it was studied the possibility of using Renyi cross-entropy as a feature selection method, but since it is not correct to compare separate entropies between different variables method was

later replaced by the mutual information method. The mutual information is useful for finding and selecting features to use either for clustering or as input to increase the accuracy of predictive algorithms.

4.2.3.1 Renyi Cross-entropy between clusters and special variables

This approach as the goal is of finding probable explanatory variables for the cluster formed by calculating the Renyi's cross-entropy, present at Equation 2.6 between initial and final curves. By doing so it is possible to measure the difference between the two sets in the sense of statistical distinguishability. By adding special variables one by one to each set it is possible to see how the Renyi's cross-entropy changes, meaning the more it closes to zero, the more close the sets are. This allows to attribute the degree of relation of special variables to the clusters formed. It is needed to pay some attention to the date-time variables that were normalized into two dimensions as presented at Chapter 3.4

4.2.3.2 Mutual Information

In this work, feature selection methods based on mutual information are going to be employed as a measure of feature selection and redundancy.

Given a set of already selected features, the next selected feature should maximize information about the corrected class and have minimal information with the already selected feature. In this way, a greedy selection method is applied to successively select the most relevant features for this work.

In this work, the use of normalized mutual information based on Renyi's quadratic entropy is proposed for feature selection, which reduces the computational complexity of the method and relies on the efficient estimation of mutual information.

4.2.4 Cluster repetition for different periods

To better understand how clusters repeat over time, different visualization approaches have been adopted. This helps to relate dates or time of day to specific clusters and to approach the definition of possible strategic behaviors.

4.2.4.1 Cluster hourly evolution

In this approach, the occurrence of certain clusters is counted for each hour. By plotting the time and clusters in an heatmap it is possible to obtain information about frequency and visualize offer changes throughout the day.

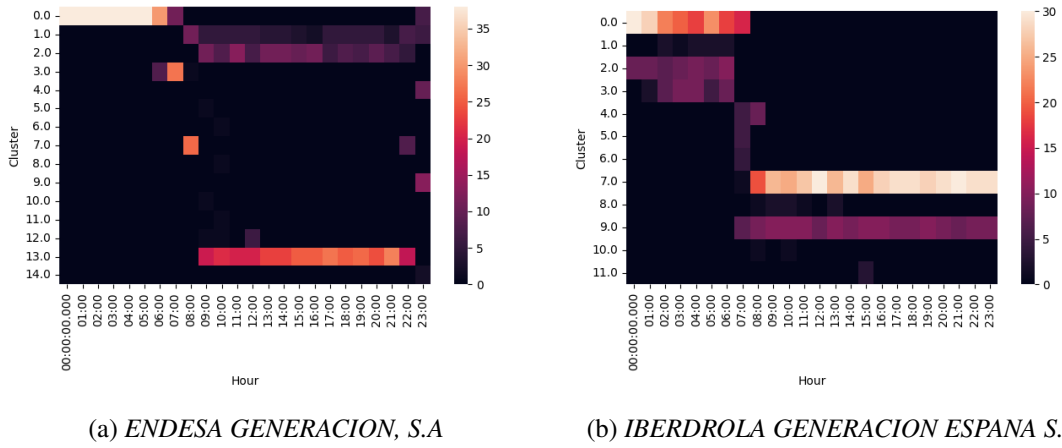


Figure 4.11: Hourly cluster evolution

To better understand the information contained in these diagrams, an in-depth analysis of cluster 0 is performed for both agents.

For *ENDESA GENERACION, S.A.*, it can be seen that cluster 0 occurs frequently between 00:00 and 05:00, with more than 35 occurrences counted for each hour of this period. Between 06:00 and 07:00, the cluster occurs less frequently, with about 30 occurrences in the first hour and less than 15 occurrences in the second hour. After that, the cluster almost does not occur at all, with only a few occurrences in the 23:00 hour.

For *IBERDROLA GENERACION ESPANA S.A.* it can be seen that cluster 0 occurs between 00:00 and 07:00. In the other hours its presence can be neglected. The cluster occurs more frequently for the hours 00:00 and 01:00, with about 30 counted events. In the following hours, the frequency decreases, always maintaining a rate above 15 occurrences per hour.

Analysis of the repetition of clusters across hours for *ENDESA GENERACION, S.A.* suggests that between 00:00 and 05:00, cluster C_{E-0} is the only cluster occurring at least 35 times in each hour. Between 06:00 and 07:00, the frequency of cluster C_{E-0} decreases and the occurrence of cluster C_{E-3} increases significantly. For the hour 08:00 it is possible to see that the only significant clusters are C_{E-1} and C_{E-7} , with C_{E-1} occurring almost 15 times and C_{E-7} more than 25 times. Between 09:00 and 22:00, three dominant clusters can be seen, namely C_{E-1} , C_{E-2} , and C_{E-13} . Of these three clusters, C_{E-13} occurs most frequently for all hours, then C_{E-2} and lastly C_{E-1} . At 23:00, it can be seen that cluster C_{E-9} occurs most frequently, followed by clusters C_{E-4} , C_{E-1} and C_{E-0} . The remaining clusters occur very infrequently and cannot be considered regular behavior for any hour of the day.

By analyzing the hourly repetitions of the clusters for *IBERDROLA GENERACION ESPANA S.A.*, it is possible to see that between 00:00 and 06:00 the clusters C_{I-0} , C_{I-1} , C_{I-2} and C_{I-3} are the most frequent, with the cluster C_{I-0} being the most repeated. For hour 07:00, it can be seen that cluster C_{I-0} is still the dominant behavior, but for this period clusters C_{I-4} , C_{I-5} , C_{I-6} and C_{I-9} also occur. From hour 08:00 to 23:00, clusters C_{I-7} and C_{I-9} show the most repetitions, with

the dominant behavior being C_{I-7} . The remaining clusters show few repetitions and cannot be precisely assigned to any time range.

4.2.4.2 Cluster daily repetition

The possibility that behaviors change during the course of the defined period was also considered. In the same way as for the hourly evolution of the clusters, an approach was developed to count the daily occurrence of different clusters.

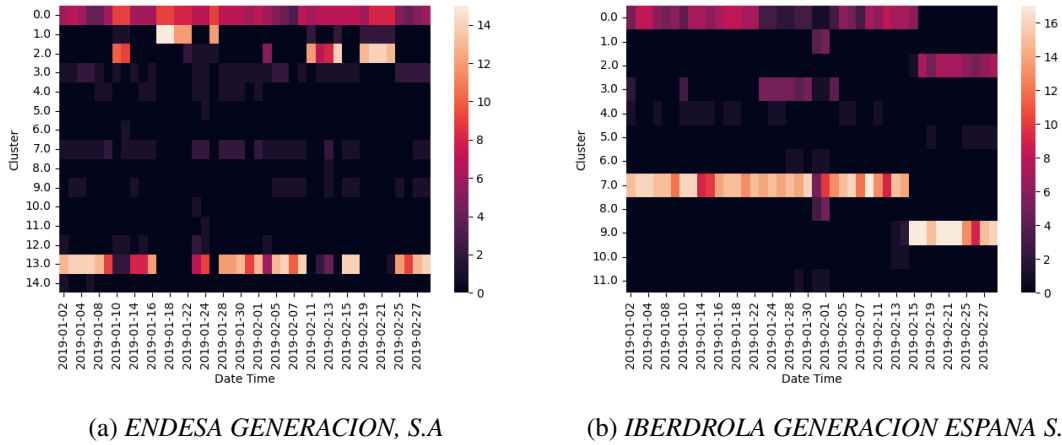


Figure 4.12: Daily cluster evolution

To better understand the significance of the results obtained from these charts, an in-depth analysis of the results for cluster 1 during the period from 02/01/2019 to 02/14/2019 for the two agents is provided.

For *ENDESA GENERACION, S.A.*, there was no occurrence of cluster 1 from 02/01/2019 to 09/01/2019, which could lead to the conclusion that the cluster is not as significant during this period as others that may occur more frequently. From 01/10/2019 to 01/13/2019, there was a slight increase in the frequency of the cluster, but it was not significant. From 01/16/2019 to 01/22/2019, there was a significant increase in the occurrence of Cluster 1 to approximately 15 daily occurrences. This means that cluster 1 occurred in about 15 different hours on each day. On the other days, there was almost no occurrence of this cluster.

At *IBERDROLA GENERACION ESPANA S.A.*, cluster 1 occurred only in the period from 01/31/2019 to 01/03/2019, with a daily frequency of about 5 different hours.

Furthermore, by analyzing the daily repetitions *ENDESA GENERACION, S.A.*, it can be seen that clusters C_{E-0} and C_{E-13} occur most frequently, clusters C_{E-1} and C_{E-2} seem to occur in the periods when cluster C_{E-13} has the fewest repetitions. Considering that it was previously found that these clusters occur most frequently at the same hours, it is interesting to extrapolate the possibility that this fact shows a clear choice between the agents' behaviors. The remaining clusters are more irregular behaviors with much less repetition than the above clusters. Although some clusters such as C_{E-3} , C_{E-7} , and C_{E-9} repeat over the analyzed period, their patterns are irregular,

and no relationships could be drawn from these graphs. In addition, clusters such as C_{E-6} , C_{E-10} , C_{E-11} , and C_{E-14} are good examples of outlier behavior that occur almost exclusively at a given event.

The analysis of daily repetitions for *IBERDROLA GENERACION ESPANA S.A* shows that, clusters C_{I-0} and C_{I-7} are the most frequent, with cluster C_{I-7} having the most repetitions. For this period, it can be seen that for the period between 01/ 31/2019 and 02/01/2019, there is an increase in the number of repetitions of clusters C_{I-1} , C_{I-6} and C_{I-8} , while clusters C_{I-0} and C_{I-7} significantly reduce their repetitions. The same is true for the period between 01/22/2019 and 01/30/2019, where cluster C_{I-3} occurs more frequently and cluster C_{I-0} reduces its recurrences. Since clusters C_{I-0} , C_{I-1} and C_{I-3} occur in the same hours, it can be concluded that their occurrence may influence each other. The same conclusions can be drawn for the C_{I-6} , C_{I-7} , and C_{I-8} clusters, which usually occur at the same hours and whose occurrence seems to be interconnected. In the period from 02/ 14/2019 to 02/27/2019, the C_{I-0} and C_{I-7} clusters no longer occur, and the C_{I-2} and C_{I-9} clusters replace these two behaviors as the dominant ones. Again, it seems possible to establish a relationship between these events, since the C_{I-0} and C_{I-2} clusters tend to occur in the same time range, and the same is true for the C_{I-7} and C_{I-9} clusters.

Chapter 5

Analysis and interpretation of results

In order to properly complete this study and answer the research questions, the results collected must be analysed so that conclusions can be drawn.

The results will be described, taking into account the available information, in order to draw conclusions about whether the methods used were successful and what information can be obtained from them. This chapter includes the analysis, presentation and interpretation of the results of this study.

The results obtained in Scenario 2 need special analysis because the hourly clusters contain curves from both agents.

5.1 Results interpretation

5.1.1 ENDESA GENERACION, S.A

Without considering the curves that were used for classification, the *ITMS* was run with 37 curves in all three scenarios defined at subsection 3.3. Since the *ITMS* complexity is $O(n^2)$, it is possible to compare the computational effort of the three scenarios.

To obtain hourly clusters, in Scenario 1 the *ITMS* was run with 37 curves for each hour of the day. This translates to $24 * 37^2$, given the *ITMS* complexity. With the hourly clusters formed, the *ITMS* was run again to obtain multi-hour clusters. Since the clustered curves contain a considerable number of duplicates, the *ITMS* computational complexity can be defined by the number of unique hourly clusters formed, which reduces the computational cost.

If this operation were not possible, the *ITMS* would involve a computational cost of 888^2 . By counting the duplicates for each cluster, the computational cost was reduced to 337^2 . This value could be further reduced by dropping the sets last decimal places, since in some occurrences they do not converge to the same value, even though their difference is minimal.

5.1.1.1 Similarity between clusters

The graphs and explanations given so far only attempt to show or analyze the clusters individually. It would be interesting to compare clusters with each other to draw conclusions about their similarity. The analysis of the similarities between the clusters is not always easy to perform with the graphs created so far. This is because there is not just one difference between the clusters, and sometimes the clusters formed can be very similar for one price range and very different for another, making it difficult to establish simple relationships between the clusters.

To address this, a new method for analyzing the clusters was developed.

The developed method measures the similarity of clusters by calculating the Euclidean distance between them. The results can be displayed in a two-dimensional heatmap, where the filtered clusters are arranged on each axis and the Euclidean distance is shown in the cells. In this way, it is possible to better define groups with identical behavior by comparing the values obtained.

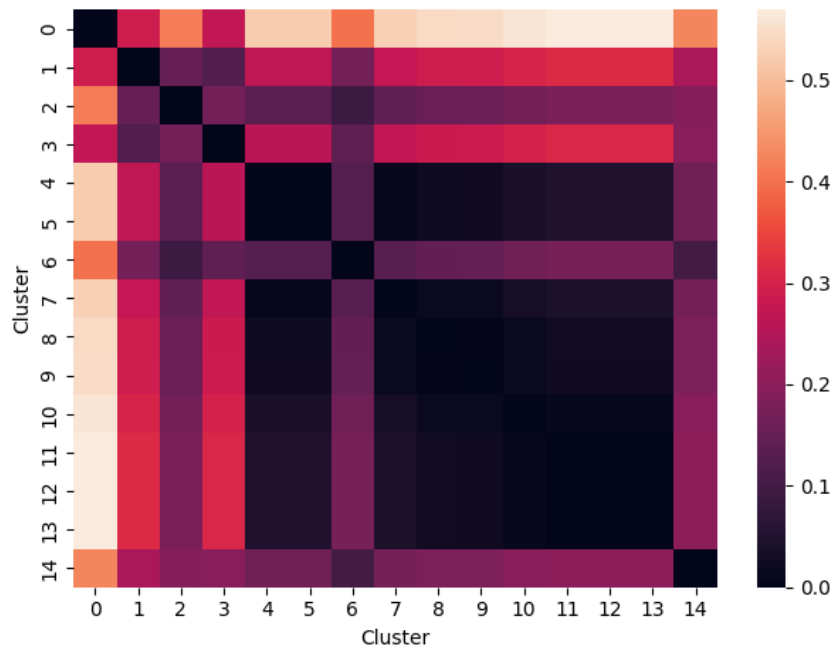


Figure 5.1: Euclidean distance between filtered clusters

To deepen the understanding of how the clusters differ along the price dimensions, an additional approach was also developed in which the prices were also divided into four different categories:

- *Low Prices*: Prices comprehended between 0€/MWh and 40€/MWh
- *Average Prices*: Prices comprehended between 50€/MWh and 62.07€/MWh
- *High prices*: Prices comprehended between 67.07€/MWh and 80€/MWh
- *Very High prices*: Prices comprehended between 160€/MWh and 180.3€/MWh

These categories were used for both agents.

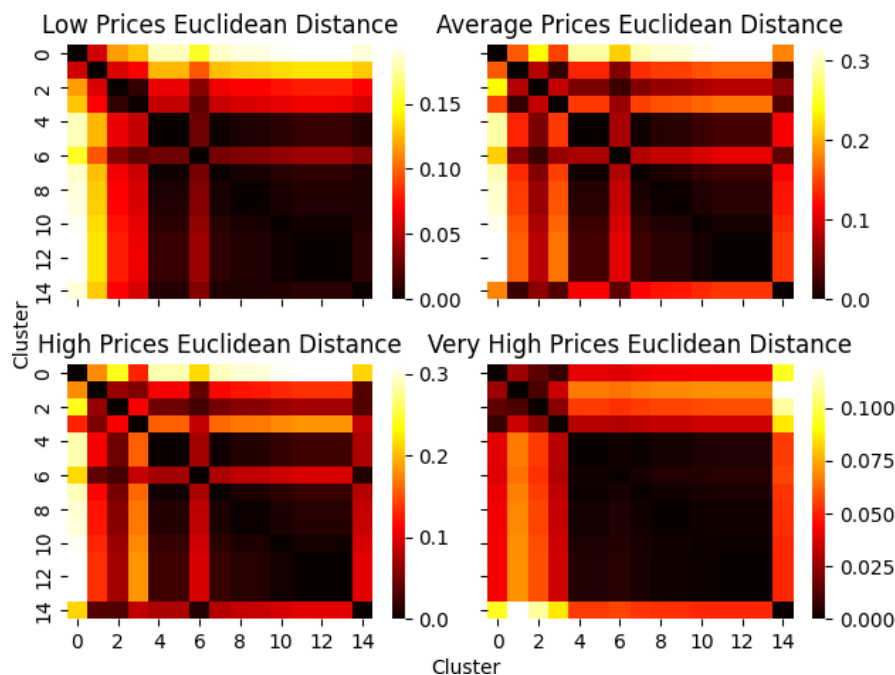


Figure 5.2: Euclidean distance between filtered clusters for different price ranges

For ENDESA GENERACION, S.A., clusters $C_{E-4}, C_{E-5}, C_{E-7}-C_{E-11}, C_{E-13}$ all offer very identical quantities, with only some price points slightly varying the quantity offered.

Clusters C_{E-1}, C_{E-2} , and C_{E-6} also show great similarities at prices below 71.07€/MWh and start to differ up to 80€/MWh, with C_{E-2} offering more quantity than C_{E-3} . Compared to the other clusters, they show lower offered quantity up to 71.07€/MWh.

Cluster C_{E-0} differs the most from the rest with the offered quantity being the lowest up to prices of 80€/MWh.

5.1.1.2 Curve classification

This work also explored the possibility of using clusters already formed to classify new curves into already known behaviors or outliers. For this purpose, 120 curves that were not clustered were used with the 15 clusters already identified. By using the modified version of *ITMS* presented in section 4.2.2, it was possible to obtain results corresponding to the statistics shown in Table 5.1.

	Number of curves
C_{E-0}	21
C_{E-1}	2
C_{E-2}	5
C_{E-3}	0
C_{E-4}	5
C_{E-5}	0
C_{E-6}	16
C_{E-7}	13
C_{E-8}	37
C_{E-9}	3
C_{E-10}	0
C_{E-11}	0
C_{E-12}	2
C_{E-13}	6
C_{E-14}	6
C_{E-15}	7
C_{E-16}	2

Table 5.1: Amount of non clustered curves per cluster (pre formed and new) for the *ENDESA GENERACION*, S.A agent

From these data, it can be concluded that the modified algorithm successfully assigned most of the curves to the existing clusters, with the exception of 9 curves that formed new behaviors (clusters C_{E-15} and C_{E-16}).

5.1.1.3 Feature Selection

Given the presence of large quantities of high dimensional data (which may come in a variety of noisy forms) and the lack of a comprehensive understanding at a finely detailed level, the feature selection plays a crucial roll in giving another level of explanation to why certain behaviors occur.

Using the concept of mutual information present in subsection 2.2.5, it is possible to find the variables that contribute most to reducing uncertainty in each cluster. If the mutual information is low, it indicates a small reduction, and a mutual information of zero between two random variables means that the variables are independent.

By computing the mutual information as explained in subsection 4.2.3, it is possible to obtain the two most relevant variables for each filtered cluster. In this way, the table 5.2 represents the results obtained with this method.

Cluster	Most relevant variables			
	First		Second	
	Mutual Information	Variable	Mutual Information	Variable
0	2.9751	[Day of the month]	2.17544	[Market Price]
1	1.669977	[Day of the month]	1.458782	['Total Hydro']
2	1.979818	[Day of the month]	1.523251	[Weekday]
3	2.722094	[Day of the month]	1.576051	[Weekday]
4	2.117962	[Day of the month]	1.714798	[Market Price]
5	N/A	N/A	N/A	N/A
6	N/A	N/A	N/A	N/A
7	2.687076	[Day of the month]	1.580358	['Total Special Regime']
8	N/A	N/A	N/A	N/A
9	2.341546	[Day of the month]	1.518981	[Weekday]
10	N/A	N/A	N/A	N/A
11	N/A	N/A	N/A	N/A
12	1.741553	[Day of the month]	1.519736	['Total Special Regime']
13	2.71841	[Day of the month]	1.663913	[Market Price]
14	0.693147	[Weekday]	0.693147	['EUA']

Table 5.2: Most relevant variables according to Mutual Information

From the results, it appears that the day of the month plays an important role in clustering, as it regularly appears as the most relevant feature in all clusters. Other relevant characteristics are market price, total hydropower production, weekday, total production in the special regime, and the EUA.

Some of the clusters consist of a single curve, effectively making them outliers. In order to compute the information contained in a cluster, several observations must be compared and the degree of relationship between them determined. For this purpose, the outliers do not have mutual information.

5.1.1.4 Scenario 1 vs Scenario 3: Clustering with single or multiple hours?

In Scenario 1, the *ITMS* was used for each agent to first cluster the hourly bid curves independently and repeat the process for each hour. With this procedure, it is possible to obtain hourly clusters of the agents. The clusters represent the modes of the bid curves and help define the most frequently recurring behaviours of the agents for each hour as well as outliers, which represent behaviours that do not occur as frequently and could indicate a strategic behaviour or a change in the behaviour of the agents.

Due to running the *ITMS* independently for each hour, there is a possibility that some clusters may represent the same behavior but not converge to the same curves. To get a complete view of an agent's behavior, one can use the *ITMS* for all hourly clusters, filtering out the redundant behaviors and leaving only the relevant behaviors.

In Scenario 3, the *ITMS* is used to cluster the agents' bidding curves for all hours in a single run. Due to the exponential nature of the algorithm, this solution tends to be slower than the one studied in Scenario 1. In Scenario 1, clusters have already been formed by the second pass of the *ITMS*, which significantly reduces the number of computations required.

In a similar representation to Scenario 1, Figure 5.3 shows the distinct clustered curves and heat maps obtained in Scenario 3.

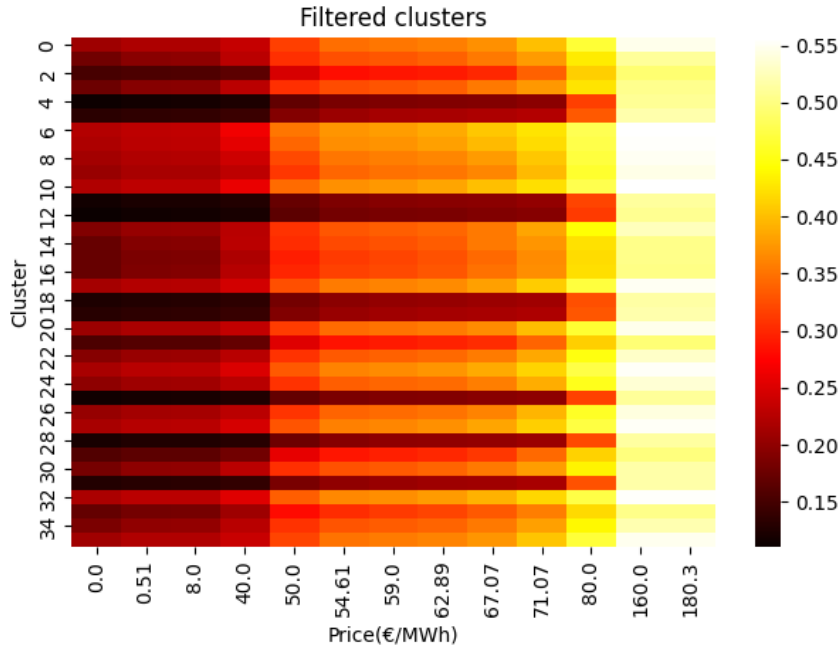


Figure 5.3: Results of Scenario 3 ran for a single agent and multiple hours, for *ENDESA GENERACION, S.A*

By visually analyzing the heat map, it is possible to identify multiple clusters that have similar values. By comparing the Scenario 3 clusters, it is possible to reduce the redundant behaviors. It is also interesting to compare these values with those of scenario 1. Theoretically, the final clusters should reflect the same information to some extent, since both scenarios contain the same pdf, provided that the parameters have been adjusted accordingly. Table 5.3 presents the relationships between the clusters of both scenarios. In this table, it can be seen that the clusters from scenario 3 have been greatly reduced and could be represented by only 7 curves. When possible to find corresponding Scenario 1 clusters they were also represented in the table.

Scenario 3	Scenario 1
$C_{E-4}, C_{E-5}, C_{E-11}, C_{E-12}, C_{E-18}, C_{E-19}, C_{E-25}, C_{E-28}, C_{E-31}$	C_{E-0}
$C_{E-20}, C_{E-21}, C_{E-29}, C_{E-33}$	C_{E-1}
$C_{E-6}, C_{E-7}, C_{E-8}, C_{E-9}, C_{E-10}, C_{E-20}, C_{E-23}, C_{E-27}, C_{E-32}, C_{E-35}$	C_{E-2}
$C_{E-22}, C_{E-24}, C_{E-26}, C_{E-34}$	None
$C_{E-1}, C_{E-15}, C_{E-16}$	C_{E-4}
C_{E-0}, C_{E-13}	None
C_{E-17}	None

Table 5.3: Similarity between clusters from Scenario 1 and Scenario 3

5.1.2 IBERDROLA GENERACION ESPANA S.A

5.1.2.1 Similarity between clusters

To measure the similarity of the clusters across all their dimensions, the Euclidean distance between the clusters is calculated. By comparing the obtained values, it is possible to define how the clusters compare to each other in terms of similarity.

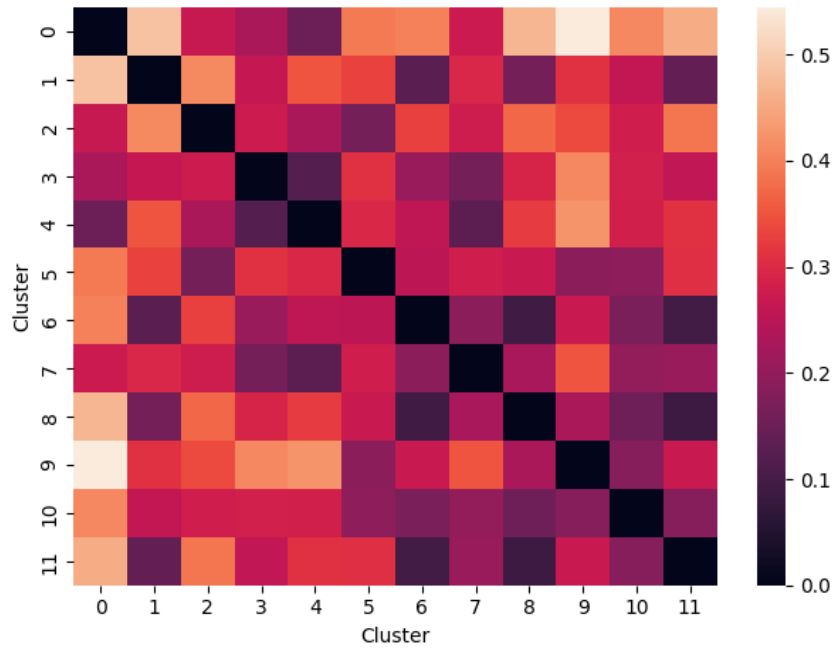


Figure 5.4: Euclidean distance between filtered clusters

The analysis of the similarities of the figure 5.4 can be found in Table 5.4, where the Euclidean distance is analyzed cluster by cluster, with the clusters with the smallest Euclidean distance being the most similar and the clusters with the largest Euclidean distance being the least similar.

	Most Similar	Least Similar
Cluster 0	$C_{I-5}, C_{I-7}, C_{I-10}$	$C_{I-1}, C_{I-3}, C_{I-4}$
Cluster 1	C_{I-3}, C_{I-4}	$C_{I-0}, C_{I-5}, C_{I-7}, C_{I-10}$
Cluster 2	$C_{I-3}, C_{I-5}, C_{I-6}$	C_{I-4}, C_{I-8}
Cluster 3	$C_{I-1}, C_{I-2}, C_{I-6}$	$C_{I-0}, C_{I-7}, C_{I-8}$
Cluster 4	$C_{I-1}, C_{I-8}, C_{I-9}$	C_{I-0}, C_{I-2}
Cluster 5	$C_{I-0}, C_{I-2}, C_{I-7}, C_{I-10}$	C_{I-1}
Cluster 6	C_{I-2}, C_{I-3}	C_{I-8}
Cluster 7	$C_{I-0}, C_{I-5}, C_{I-9}, C_{I-10}$	C_{I-1}, C_{I-3}
Cluster 8	C_{I-4}, C_{I-9}	$C_{I-2}, C_{I-3}, C_{I-6}, C_{I-4}, C_{I-7}, C_{I-8}$
Cluster 9	$C_{I-4}, C_{I-7}, C_{I-8}$	C_{I-0}
Cluster 10	$C_{I-0}, C_{I-5}, C_{I-7}$	C_{I-1}
Cluster 11	C_{I-10}	C_{I-0}

Table 5.4: Most similar and Least similar clusters

By dividing the prices into four different ranges, it is possible to further identify the areas with the largest differences between the clusters. The results obtained by this method are shown in Figure 5.5.

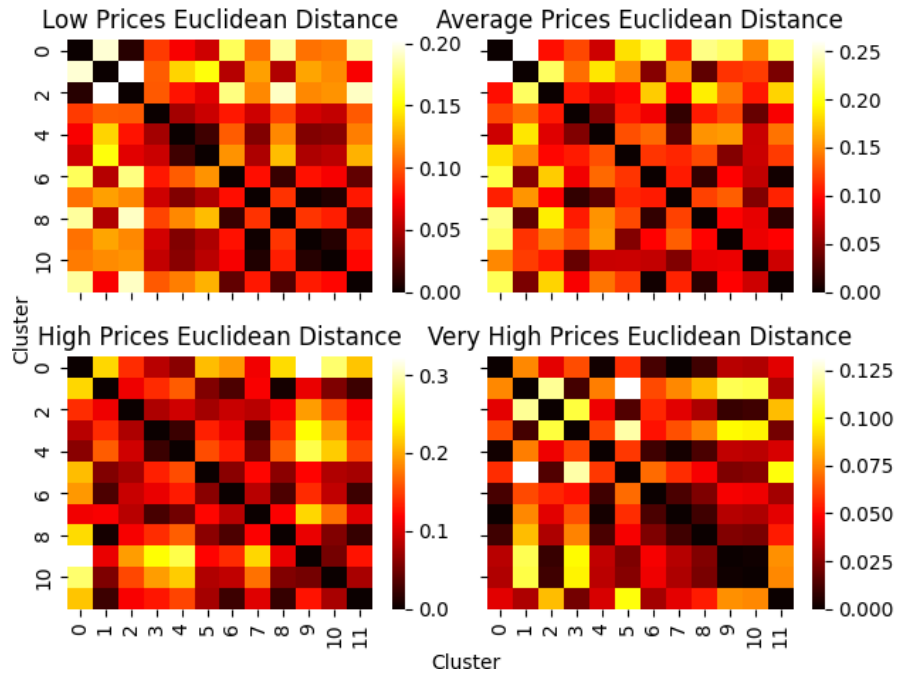


Figure 5.5: Euclidean distance between filtered clusters for different price ranges

For low prices, the following conclusions can be drawn:

	Most Similar	Least Similar
Cluster 0	$C_{I-5}, C_{I-7}, C_{I-10}$	$C_{I-1}, C_{I-3}, C_{I-4}$
Cluster 1	C_{I-3}, C_{I-4}	$C_{I-0}, C_{I-5}, C_{I-7}, C_{I-8}, C_{I-9}$
Cluster 2	$C_{I-3}, C_{I-8}, C_{I-9}$	C_{I-1}
Cluster 3	$C_{I-1}, C_{I-2}, C_{I-4}, C_{I-6}, C_{I-8}, C_{I-9}$	$C_{I-0}, C_{I-7}, C_{I-10}$
Cluster 4	$C_{I-1}, C_{I-3}, C_{I-6}, C_{I-8}, C_{I-9}$	$C_{I-0}, C_{I-5}, C_{I-7}, C_{I-10}$
Cluster 5	C_{I-0}, C_{I-10}	C_{I-1}, C_{I-4}
Cluster 6	$C_{I-3}, C_{I-4}, C_{I-8}, C_{I-9}$	C_{I-0}
Cluster 7	C_{I-0}, C_{I-10}	$C_{I-1}, C_{I-3}, C_{I-4}$
Cluster 8	$C_{I-2}, C_{I-3}, C_{I-4}, C_{I-6}, C_{I-9}$	C_{I-5}, C_{I-1}
Cluster 9	$C_{I-2}, C_{I-3}, C_{I-4}, C_{I-6}, C_{I-8}$	C_{I-0}, C_{I-1}
Cluster 10	$C_{I-0}, C_{I-5}, C_{I-7}$	$C_{I-1}, C_{I-3}, C_{I-4}$

For average prices:

	Most Similar	Least Similar
Cluster 0	$C_{I-5}, C_{I-7}, C_{I-10}$	$C_{I-1}, C_{I-2}, C_{I-23}, C_{I-4}, C_{I-6}$
Cluster 1	C_{I-11}	$C_{I-0}, C_{I-5}, C_{I-7}, C_{I-8}, C_{I-10}$
Cluster 2	$C_{I-3}, C_{I-6}, C_{I-9}$	C_{I-0}
Cluster 3	C_{I-2}, C_{I-6}	$C_{I-0}, C_{I-5}, C_{I-7}, C_{I-8}, C_{I-10}$
Cluster 4	C_{I-8}	C_{I-0}
Cluster 5	$C_{I-0}, C_{I-7}, C_{I-10}$	C_{I-1}, C_{I-3}
Cluster 6	$C_{I-2}, C_{I-3}, C_{I-9}$	C_{I-0}, C_{I-8}
Cluster 7	$C_{I-0}, C_{I-5}, C_{I-10}$	C_{I-1}, C_{I-3}
Cluster 8	C_{I-4}	$C_{I-1}, C_{I-3}, C_{I-6}$
Cluster 9	C_{I-2}, C_{I-6}	C_{I-0}
Cluster 10	$C_{I-0}, C_{I-5}, C_{I-7}$	C_{I-1}, C_{I-3}

For high prices:

	Most Similar	Least Similar
Cluster 0	$C_{I-4}, C_{I-5}, C_{I-6}, C_{I-7}, C_{I-9}, C_{I-10}$	C_{I-2}, C_{I-3}
Cluster 1	$C_{I-2}, C_{I-4}, C_{I-6}$	C_{I-8}, C_{I-9}
Cluster 2	$C_{I-1}, C_{I-3}, C_{I-6}$	C_{I-8}, C_{I-9}
Cluster 3	C_{I-2}, C_{I-6}	$C_{I-0}, C_{I-2}, C_{I-4}, C_{I-7}, C_{I-8}, C_{I-9}, C_{I-10}$
Cluster 4	$C_{I-0}, C_{I-1}, C_{I-5}, C_{I-7}$	C_{I-3}
Cluster 5	$C_{I-4}, C_{I-7}, C_{I-10}$	C_{I-1}
Cluster 6	$C_{I-0}, C_{I-1}, C_{I-2}, C_{I-3}, C_{I-10}$	C_{I-8}, C_{I-9}
Cluster 7	$C_{I-0}, C_{I-4}, C_{I-5}, C_{I-9}, C_{I-10}$	C_{I-3}
Cluster 8	C_{I-9}	$C_{I-1}, C_{I-2}, C_{I-3}, C_{I-6}$
Cluster 9	$C_{I-0}, C_{I-7}, C_{I-8}, C_{I-10}$	$C_{I-1}, C_{I-2}, C_{I-3}, C_{I-6}$
Cluster 10	$C_{I-0}, C_{I-5}, C_{I-6}, C_{I-7}, C_{I-9}$	C_{I-3}

For very high prices:

	Most Similar	Least Similar
Cluster 0	$C_{I-2}, C_{I-5}, C_{I-10}$	$C_{I-1}, C_{I-3}, C_{I-4}, C_{I-6}, C_{I-7}, C_{I-8}, C_{I-9}$
Cluster 1	$C_{I-3}, C_{I-4}, C_5, C_{I-7}, C_{I-8}, C_{I-9}$	$C_{I-0}, C_{I-2}, C_{I-10}$
Cluster 2	$C_{I-0}, C_{I-5}, C_{I-10}$	$C_{I-1}, C_{I-4}, C_{I-7}, C_{I-8}, C_{I-9}$
Cluster 3	$C_{I-1}, C_{I-5}-C_{I-49} 1,$	C_{I-0}
Cluster 4	$C_{I-1}, C_{I-8}, C_{I-9}$	$C_{I-0}, C_{I-2}, C_{I-5}, C_{I-10}$
Cluster 5	All except C_{I-4}, C_{I-5}	C_{I-4}, C_{I-5}
Cluster 6	$C_{I-3}, C_{I-5}, C_{I-9}$	C_{I-0}
Cluster 7	$C_{I-1}, C_{I-3}, C_{I-5}$	C_{I-0}, C_{I-2}
Cluster 8	$C_{I-1}, C_{I-3}, C_{I-4}, C_{I-5}, C_{I-6}, C_{I-7}$	$C_{I-0}, C_{I-2}, C_{I-10}$
Cluster 9	$C_{I-1}, C_{I-3}, C_{I-4}, C_{I-5}, C_{I-6}, C_{I-8}$	$C_{I-0}, C_{I-2}, C_{I-10}$
Cluster 10	$C_{I-0}, C_{I-2}-C_{I-5}$	$C_{I-1}, C_{I-4}, C_{I-8}, C_{I-9}$

5.1.2.2 Curve classification

As explained for the agent *ENDESA GENERACION, S.A*, the statistics in the table 5.1 were obtained by curve classification for 127 unclustered curves of the agent *IBERDROLA GENERACION ESPANA S.A*.

	Number of curves
C_{I-0}	6
C_{I-1}	18
C_{I-2}	1
C_{I-3}	2
C_{I-4}	1
C_{I-5}	1
C_{I-6}	0
C_{I-7}	0
C_{I-8}	0
C_{I-9}	1
C_{I-10}	19
C_{I-11}	63
C_{I-12}	1
C_{I-13}	4
C_{I-14}	7
C_{I-15}	1
C_{I-16}	2

Table 5.5: Number of non clustered curves per already formed and new clusters for the *IBER-DROLA GENERACION ESPANA S.A* agent

For this agent, it was also shown that it is possible to identify already known behaviors with new, non-clustered curves. In this particular case, 5 new clusters were found (C_{I-12} , C_{I-13} , C_{I-14} , C_{I-15} , and C_{I-16}).

5.1.2.3 Feature Relevance

The feature relevance for this agent was processed as previously described for ENDESA GENERACION, S.A and the results were gathered in table 5.6.

Cluster	Most relevant variables			
	First		Second	
	Mutual Information	Variable	Mutual Information	Variable
0	2.72583	[Day of the month]	2.040252	[Market Price]
1	0.680877	[Weekday]	0.680849	['Month SIN', 'Month COS']
2	1.95162	[Day of the month]	1.581264	[Market Price]
3	2.021889	[Day of the month]	1.897058	[Market Price]
4	2.248687	[Day of the month]	1.574551	[Weekday]
5	1.391196	[Day of the month]	1.272966	[Weekday]
6	1.386294	[Weekday]	1.201523	[Day of the month]
7	2.866911	[Day of the month]	1.504168	['Total Special Regime']
8	0.900716	['Demand']	0.75764	['Total Hydro']
9	2.072474	[Day of the month]	1.593378	[Weekday]
10	0.693147	[Market Price]	0.693147	['Total Special Regime']
11	1.098612	[Weekday]	1.098536	[Market Price]

Table 5.6: Most relevant variables according to Mutual Information

The results show that the offers dates have information that can reduce the uncertainty of the bidding curve prediction. Characteristics such as demand, market price, total hydraulics, and specific regime productions also contain information that can be used to better define and predict some of the clusters.

5.1.2.4 Scenario 1 vs Scenario 3: Clustering with single or multiple hours?

Similar to the study for ENDESA GENERACION, S.A., this subsection compares the results of Scenario 1 and Scenario 3. For this agent, the relevant clusters found for Scenario 3 are represented by a heat map in Figure 5.6.

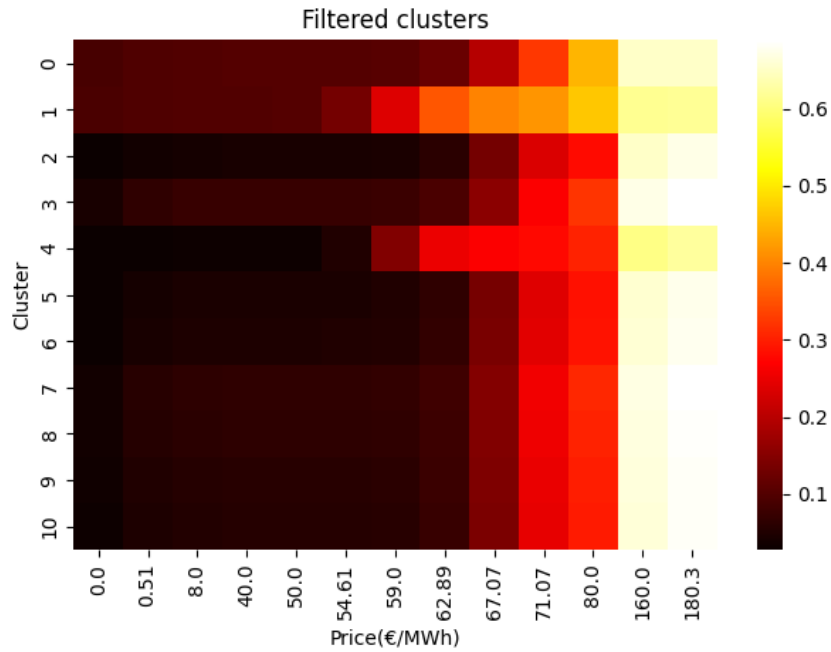


Figure 5.6: Results for single agent and multiple hour, Scenario 3, for *IBERDROLA GENERACION ESPANA S.A*

Table 5.7 portrays the most apparent relations between the clusters from Scenario 3 and Scenario 1.

Scenario 3	Scenario 1
2,5,6	1
0	8
1	4
4	0
3,7,8,9,10	2

Table 5.7: Similarity between clusters from Scenario 1 and Scenario 3

It can be inferred that the results of Scenario 3 and 1 are indeed related to some extent, and that Scenario 1 should be the preferred method for clustering the bidding curves because it provides hourly information of behaviours. In addition, Scenario 1 is less computationally intensive and the parameters used are easier to control because an additional clustering step allows for a more relevant representation of the intended data.

Chapter 6

Conclusions and Future Work

This final chapter addresses the main conclusions drawn from the case study and the findings presented in Chapter 5. Although this work already provides a solid basis for understanding this problem, further studies and tests can still be carried out to improve the state of the art regarding the application of *ITMS* in energy markets. With this in mind, Section 6.3 explores some possible ways to provide continuity to this work.

6.1 Achievements

The analysis of the energy market is based on the extraction of knowledge from the available data in order to show the agents how to compete with the rival companies. The main objective of this work was to test the capabilities of the *ITMS* algorithm to identify strategic behaviors in quantity-price offers of energy market agents.

ITL concepts are becoming extremely useful in knowledge discovery in datasets, The *ITMS* is a good example of how theory provides the solid foundations for discovering modes in probability density functions and detecting clusters in data.

The *ITMS* algorithm was conceived as a means of recognizing properties of data clusters based on a sound theoretical background, rather than merely a mechanical algorithmic procedure. Its value in pattern recognition has been proven in this application, particularly because of its flexibility in discovering cluster properties at different levels of aggregation through appropriate manipulation of the λ parameter: optimizing the problem to maximize the entropy between clusters and minimize the sum of the entropies of the individual clusters.

This study analyzed how to treat the available data so that it can be processed using statistical and *ITL* concepts to define the presence of strategic behaviors in the agents' bids. To this end, it was possible to define hourly clusters for *IBERDROLA GENERACION ESPANA S.A* and *ENDESA GENERACION, S.A* and find the most appropriate parameters for the *ITMS* to cluster the bids in a significant way for this work. The clusters are representative of different behaviors related to specific hours, and this work explored different ways to represent these data, concluding

that stepwise and heatmap graphs are two efficient ways to represent the offers in all their dimensions. In addition, it was possible to see that these clusters differed from each other, revealing the agents' intention to make a quantity adjustment for different price ranges. It was also possible to analyze the similarities in the offers for different prices to better understand in which ranges the offers usually fall.

It was determined that the best way to analyze the markets was to run the *ITMS* algorithm by separating the agents and the hours for which the offers are made. This made it possible to obtain information about hourly clusters and filter these clusters to obtain multi-hour clusters. When the *ITMS* algorithm is run for multiple hours at once, the information about hourly clusters is lost, and due to the exponential complexity of the algorithm, the computational cost is much greater.

Once the clusters were defined, it was also analyzed what might represent consistent patterns or what might be outliers that are not regular offers or do not occur frequently. For this purpose, it was not sufficient to analyze only the shapes or similarities of the clusters, but it was also necessary to examine how often they were repeated in different cases. In this study, the hourly and daily repetitions of clusters were analyzed. The analysis of hourly clusters showed that it is possible to define higher repetitions of certain clusters for specific day times and hourly ranges. The analysis of daily repetitions made it possible to identify outliers and the most frequent clusters, as well as to define clusters in relation to the dates on which they were offered, and showed that there are some obvious cases of relationships between the use of certain clusters.

This knowledge may be useful to define and predict agents objectives when performing strategic analysis.

The *ITMS* algorithm was also modified to classify new curves with respect to the clusters already created. This experiment shows that it is possible to create a usable database of previous cluster information and quickly run new tests on new data. The use of a reduced set of scenarios significantly reduces the computation time required by the program and opens the possibility of developing a solution to run this algorithm in real time.

In addition, an attempt was made to establish relationships between several different variables and the behaviors. For this purpose, the mutual information between the clusters and the different variables was calculated in a two-step process that allowed determining which variables contained the most information. This process is important to better understand the clusters and their possible causes.

Due to time restrictions during the development of this work further relations in data and integration with other algorithms were not analysed, falling on the future work field. To produce better results, more generations or more individuals could be used but this would require higher computational times.

6.2 Conclusions

This paper began by noting that *MIBEL* is an energy market that operates on a highly competitive model. Because some agents have significant market power, it is important to know how and

why they engage in certain strategic or tactical behaviors. If decision makers know how or why behaviors occur, they can respond and increase their profitability.

Before it is possible to predict behaviors, it is necessary to define what behaviors agents engage in. This study was applied to *ENDESA GENERACION, S.A* and *IBERDROLA GENERACION ESPANA S.A* because these two agents have a significant number of units in this market.

With this study, it was possible to identify patterns and confirm that they change depending on the agent, date and time. It was also possible to find that most of these behaviors are repeated over a period of time, while others are not frequent and are classified as non-normal behaviors.

For the same situation, it was also found that the agents' behaviors change differently, leading to the conclusion that the decision criteria vary from agent to agent and that their weighting also varies from behavior to behavior.

An initial analysis was also conducted to determine which external variables might be associated with the emergence of certain behaviors. This information can be valuable in understanding how strategies are formulated. Analyzing why the strategies or tactics used are employed is outside the goal of this thesis, as drawing conclusions about this would require a series of hypotheses that could not be adapted within the time constraints of this study.

6.3 Future Work

Although the time frame of this project did not allow to obtain results for all hypotheses of this problem, it was possible to perform a technical reflection of the requirements and future work that would benefit the state of the art for energy markets.

The main objective is to develop a robust model to predict the strategic behavior of agents based on the results obtained and publicly available information. To simplify the approach to this problem, it is possible to break it down into a list of steps required to achieve the desired results.

- *Definition of behaviors:* This step summarizes most of the work on this study. It concerns data collection and processing, as well as determining the best techniques for describing and identifying agent behaviors. Although this step does not allow for the extraction of the amount of specific information needed by decision makers to take informed actions.
- *Construction of a strong and tested portfolio of results:* In this step, it is important to develop different solutions and scenarios to test the robustness of using ITMS and ITL concepts for this problem. There is a solid theoretical argument that minimizing hourly clusters should be employed through the use of *ITMS*, nevertheless it would be interesting to demonstrate with more examples that an easily interpretable approximation of reality can be achieved through their use.
- *Building a Data-Based Predictive Model:* With proven foundations and extracted knowledge, it is possible to construct a robust predictive model that uses the available data to predict the bidding strategies of agents in specific situations. This is an important step for

strategy planning, as it provides incredible insight into how competitors operate in the market. It also provides the requirements to identify outliers and flag them in order to relay necessary tactics to counter unusual behaviors.

In order to predict the tactical or strategic behavior of agents, strong causal relationships would need to be established between external variables that might be associated with strategy making. These would be defined as decision criteria that change from agent to agent and are weighted differently in different situations. In this way, it would be possible to formulate a future work related to the circumstances around the offers to identify these criteria. After identifying the criteria, it would be possible to use data-based predictive models such as decision trees to define how the strategies could be formulated.

References

- [1] Germán Aneiros, Juan M. Vilar, Ricardo Cao, and Antonio Muñoz San Roque. Functional prediction for the residual demand in electricity spot markets. *IEEE Transactions on Power Systems*, 28(4):4201–4208, 2013.
- [2] Carlos Batlle and Julián Barquín. A strategic production costing model for electricity market price analysis. *IEEE Transactions on Power Systems*, 20(1):67–74, feb 2005.
- [3] Fco Alberto Campos, Antonio Munoz San Roque, Eugenio F. Sanchez-Ubeda, and Jose Portela Gonzalez. Strategic Bidding in Secondary Reserve Markets. *IEEE Transactions on Power Systems*, 31(4):2847–2856, jul 2016.
- [4] Alex Coronati, José R. Andrade, and Ricardo J. Bessa. A deep learning method for forecasting residual market curves. *Electric Power Systems Research*, 190, jan 2021.
- [5] Christopher J Day, Benjamin F Hobbs, and Jong-Shi Pang. Oligopolistic Competition in Power Networks: A Conjectured Supply Function Approach. *IEEE TRANSACTIONS ON POWER SYSTEMS*, 17(3):597, 2002.
- [6] Gustavo De Vivero-Serrano, Kenneth Bruninx, and Erik Delarue. Implications of bid structures on the offering strategies of merchant energy storage systems. *Applied Energy*, 251:113375, oct 2019.
- [7] C A Díaz, J Villar, F A Campos, and J Reneses. Electricity market equilibrium based on conjectural variations. *Electric Power Systems Research*, 80(12):1572–1579, 2010.
- [8] Eugenio Fco, S ´ Anchez- ´ Ubeda, and Louis Wehenkel. The Hinges model: A one-dimensional continuous piecewise polynomial model. *Information Processing and Management of Uncertainty in Knowledge-based Systems, IPMU*, pages 878–885, June 1998.
- [9] Pablo González, José Villar, Cristian A. Díaz, and Fco Alberto Campos. Joint energy and reserve markets: Current implementations and modeling trends. *Electric Power Systems Research*, 109:101–111, apr 2014.
- [10] Isabelle Guyon and Andre@tuebingen Mpg De. An Introduction to Variable and Feature Selection André Elisseeff. *Journal of Machine Learning Research*, 3:1157–1182, 2003.
- [11] Henri Halonen. *A Proposal for Business Intelligence Solution Based on Systems Integration and Enhanced Reporting Functionality*. PhD thesis, Metropolia Ammattikorkeakoulu, 2016.
- [12] T.J. Hammons, L.A. Barroso, and H. Rudnick. Latin America: deregulation in a hydro-dominated market. *Oil, Gas & Energy Law*, 1(3), jul 2003.
- [13] Simon S. Haykin. *Unsupervised adaptive filtering*. Wiley, 2000.

- [14] IEA, Organisation for Economic Co-operation, Organisation for Economic Co-operation, Development, Development, International Energy Agency, and International Energy Agency. Competition in Electricity Markets. Technical report, IEA, 2001.
- [15] Pengfei Li, Shaoru Wang, and Junfeng Jing. The segmentation in textile printing image based on mean shift. *Proceeding 2009 IEEE 10th International Conference on Computer-Aided Industrial Design and Conceptual Design: E-Business, Creative Design, Manufacturing - CAID and CD'2009*, pages 1528–1532, 2009.
- [16] Zuyi Li and Mohammad Shahidehpour. Security-Constrained Unit Commitment for Simultaneous Clearing of Energy and Ancillary Services Markets. *IEEE TRANSACTIONS ON POWER SYSTEMS*, 20(2):1079, 2005.
- [17] Can Tao Liu and Bao Gang Hu. Mutual information based on Renyi's entropy feature selection. *Proceedings - 2009 IEEE International Conference on Intelligent Computing and Intelligent Systems, ICIS 2009*, 1(November 2009):816–820, 2009.
- [18] Xingwang Ma, David Sun, and Kwok Cheung. Energy and reserve dispatch in a multi-zone electricity market. *IEEE Transactions on Power Systems*, 14(3):913–919, 1999.
- [19] A. Mateo, E. F. Sánchez-Úbeda, A. Muñoz, J. García-González, J. Villar, M. Casado, A. Sáiz, E. J. García, and R. González. Modeling bidding curves: The linear hinges model versus the sigmo model. *2001 IEEE Porto Power Tech Proceedings*, 1:420–425, 2001.
- [20] MIBEL Regulatory Council. Description of the operation of the mibel. Technical Report November, MIBEL, 2009.
- [21] Vladimiro Miranda, Adriana R. Garcez Castro, and Shigeaki Lima. Diagnosing faults in power transformers with autoassociative neural networks and mean shift. *IEEE Transactions on Power Delivery*, 27(3):1350–1357, 2012.
- [22] Vladimiro Miranda, Adriana R. Garcez Castro, and Shigeaki Lima. Diagnosing faults in power transformers with autoassociative neural networks and mean shift. *IEEE Transactions on Power Delivery*, 27(3):1350–1357, 2012.
- [23] Luis Ferreira Monteiro. *Analysis of the Auction and Continuous Intraday Electricity Markets of the MIBEL*. PhD thesis, Universidade do Porto, 2021.
- [24] Allen G. Morinec and F. Eugenio Villaseca. Optimal generator bidding strategies for power and ancillary services using game theory. *40th North American Power Symposium, NAPS2008*, 2008.
- [25] Vítor Navega. <https://drive.google.com/drive/folders/1aJjM1mBjjiFDX8EtSUwQ9KYcv0W7O7L5?usp=sharing>, 2022.
- [26] Rita Nazanin, Vafaei, Maria, Sarmento and Almeida Ribeiro. *Data Normalization in Decision Making Processes*. PhD thesis, Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa, 2021.
- [27] OMIE. Electricity market | OMIE.
- [28] S A Omie. Modelo de Ficheros para la distribución pública de Información del mercado de electricidad Versión 1.33.

- [29] Pavličić Dubravka M. Normalization affects the results of MADM methods. *Yugoslav Journal of Operations Research*, 2001.
- [30] Filip Petrovski. *Development of Information Management solution for Measuring People Performance Internship report submitted in partial fulfilment of the requirements for the Master's degree in Information Management*. PhD thesis, NOVA Information Management School, 2016.
- [31] Akash Pooransingh, Cathy Ann Radix, and Anil Kokaram. The path assigned mean shift algorithm: A new fast mean shift implementation for colour image segmentation. *Proceedings - International Conference on Image Processing, ICIP*, pages 597–600, 2008.
- [32] Jose C. Principe. Information-Theoretic Learning Tutorial.
- [33] R. Raineri, S. Ríos, and D. Schiele. Technical and economic aspects of ancillary services markets in the electric power industry: an international comparison. *Energy Policy*, 34(13):1540–1555, sep 2006.
- [34] Sudhir Rao, Allan de Medeiros Martins, and José C. Príncipe. Mean shift: An information theoretic perspective. *Pattern Recognition Letters*, 30(3):222–230, 2009.
- [35] Sudhir Rao, Liu Weifeng, Jose C. Principe, and Allan De Medeiros Martins. Information theoretic mean shift algorithm. *Proceedings of the 2006 16th IEEE Signal Processing Society Workshop on Machine Learning for Signal Processing, MLSP 2006*, pages 155–160, 2006.
- [36] Liviane Rego, Jean Sumaili, Vladimiro Miranda, Carlos Francês, Marcelino Silva, and Ádamo Santana. Mean shift densification of scarce data sets in short-term electric power load forecasting for special days. *Electrical Engineering*, 99(3):881–898, sep 2017.
- [37] REN. SISTEMA DE INFORMAÇÃO DE MERCADOS DE ENERGIA.
- [38] Tiago Rodrigues, Vladimiro Miranda, and Jean Sumaili. *Finding Representative Scenarios in Wind Power or Load Forecast*. PhD thesis, Faculdade de Engenharia da Universidade do Porto, 2013.
- [39] Carlos Ruiz and Antonio J. Conejo. Pool strategy of a producer with endogenous formation of locational marginal prices. *IEEE Transactions on Power Systems*, 24(4):1855–1866, 2009.
- [40] Daniel Carlos Silva Costa Sá. *Load Forecasting on Special Days*. PhD thesis, Faculdade de Engenharia da Universidade do Porto, 2013.
- [41] Eugenio F. Sánchez Úbeda, Antonio Muñoz, and José Villar. Minería y visualización de datos del mercado eléctrico español, 2006.
- [42] *SendeCO₂*. Precios CO₂ - Sendeco2.
- [43] Ferenc Cole Thierrin, Fady Alajaji, and Tamás Linder. On the rényi cross-entropy, 2022.
- [44] Tibco. What is Data Mining? | TIBCO Software.
- [45] Arttu Tolvanen. Developing an efficient business intelligence solution for day-to-day supply chain management : Case pulp and paper industry workshop. 2019.
- [46] TU and Towards Data Science. Optimising pairwise Euclidean distance calculations using Python, jan 2021.

- [47] Nazanin Vafaei, Rita A. Ribeiro, and Luis M. Camarinha-Matos. Data normalisation techniques in decision making: case study with TOPSIS method. *International Journal of Information and Decision Sciences*, 10(1):19–38, jan 2018.
- [48] Mariano Ventosa, Alvaro Baillo, Andres Ramos, Michel Rivier, Mariano Ventosa, Alvaro Baillo, Andres Ramos, and Michel Rivier. Electricity market modeling trends. *Energy Policy*, 33(7):897–913, 2005.
- [49] J. Villar, A. Muñoz, E. F. Sánchez-Úbeda, A. Mateo, M. Casado, A. Campos, J. Maté, E. Centeno, S. Rubio, J. J. Marcos, and R. González. SGO: Management information system for strategic bidding in electrical markets. *2001 IEEE Porto Power Tech Proceedings*, 1:414–419, 2001.
- [50] Vladimiro Miranda. INFORMATION THEORETIC LEARNING PRINCIPLES a short overview. In *CBA 2018, João Pessoa (PB), Brazil*, page 30, 2018.
- [51] Wikipedia. Normalization - Wikipedia.
- [52] Professor Frank A Wolak. An Empirical Analysis of the Impact of Hedge Contracts on Bidding Behavior in a Competitive Electricity Market. *International Economic Journal*, 14(2):1–39, 2000.
- [53] YAHOO. Natural Gas Jul 22 (NG=F) Stock Historical Prices & Data — Yahoo Finance.

Appendix A

Data Mining

A.1 Data Sources

This subsection identifies the data sources from which the relevant data were obtained. Based on the information contained in these sources, a data framework is created that can be used in *ITMS*, as well as a data framework on which possible explanatory variables can be found.

One source to find relevant information is the *OMIE* website [27]. Several files can be found there that contain information about day-ahead market bids. The files that will be used for this work are:

omie Publications Links FAQs Q es | pt | en

OMIE Spot today Spot market Agents Services OMIEData

Day-ahead Market

1. Prices

Day-ahead market hourly prices in Spain
Day-ahead market hourly price in Portugal

2. Programmes

Spanish daily basic matching process program
Daily basic matching process program by company
Portuguese daily basic matching process program
Daily functioning basic program
Daily final viable program
Total disaggregated power after Day-ahead market
Market, bilaterals and special regime negotiated total power afterd Day-ahead market
Market, bilaterals and special regime total power in the daily functioning basic program

3. Curves

Aggregate supply and demand curves of Day-ahead market
Monthly files with aggregate supply and demand curves of Day-ahead market including bid units

4. Bids

Header of bids for Day-ahead Market
Day-ahead market bids detail

Figure A.1: List of files used represented in red boxes from the OMIE website [27]

OMIE provides a model [28] containing all market information files with the format and description of all variables contained therein. For better illustration, the tables available in [28] for the files used are added in Appendices A.2.1, A.3, A.4. For the implementation of ITMS, the files Header of bids for Day-ahead Market (CAB) and Day-ahead market bids detail (DET) are needed to obtain the aggregated bid curves. There are several variables of importance in these files:

- Offer code
- Offering unit
- Market date and session
- Unit maximum power
- Price
- Energy

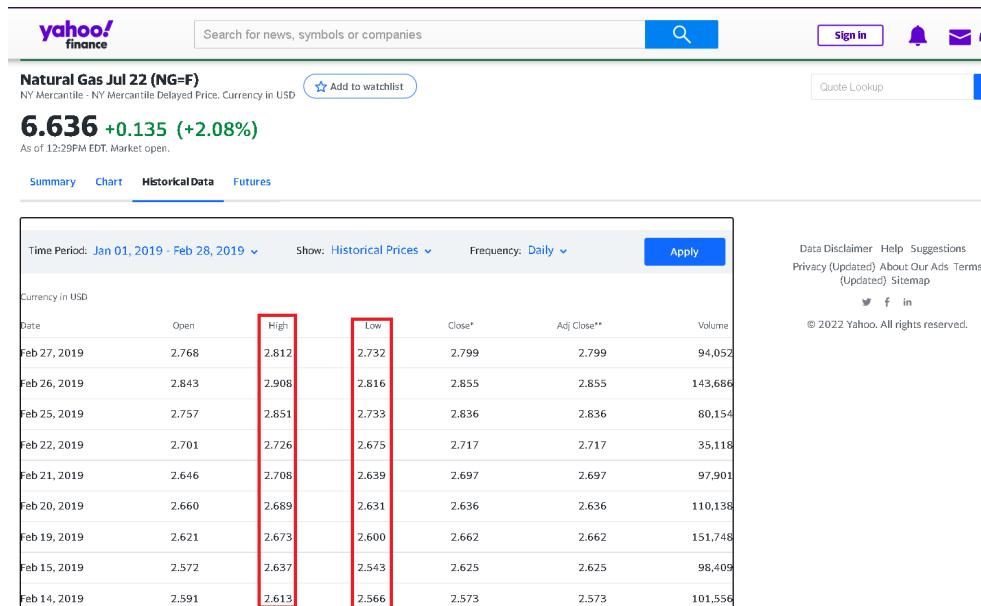
From [28], some of the possible explanatory variables can also be obtained from the Portuguese and Spanish daily basic matching process program (PDBC) and the total disaggregated

power by day-ahead market (*PBDC-SOTA*). The hourly variables included in these files are listed below:

- Cleared energy
- Total energy produced by hidro units
- Total energy produced by special regime units
- Demand

In addition, it is possible to find explanatory variables outside of *OMIE*, such as natural gas prices found on the [53] and CO2 prices found on the [42], which were also considered. From these files, the following variables can be obtained:

- Natural Gas lower price
- Natural Gas higher price
- CO2 European Emission Allowances
- CO2 spread between European Emission Allowances and Certified Emission Reductions



Natural Gas Jul 22 (NG=F)
 NY Mercantile - NY Mercantile Delayed Price. Currency in USD
6.636 +0.135 (+2.08%)
 As of 12:29PM EDT. Market open.

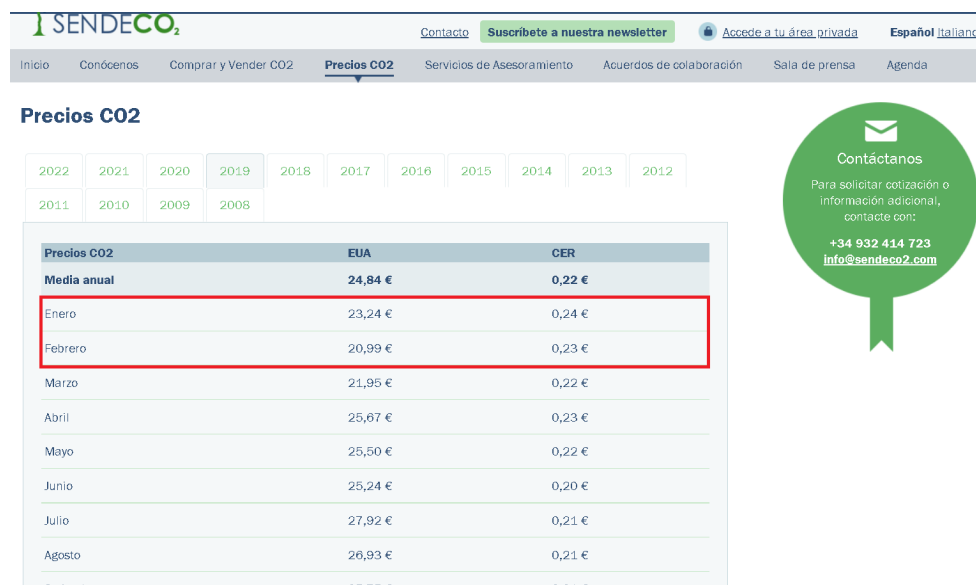
Summary Chart **Historical Data** Futures

Time Period: Jan 01, 2019 - Feb 28, 2019 Show: Historical Prices Frequency: Daily Apply

Date	Open	High	Low	Close*	Adj Close**	Volume
Feb 27, 2019	2.768	2.812	2.732	2.799	2.799	94,052
Feb 26, 2019	2.843	2.908	2.816	2.855	2.855	143,686
Feb 25, 2019	2.757	2.851	2.733	2.836	2.836	80,154
Feb 22, 2019	2.701	2.726	2.675	2.717	2.717	35,118
Feb 21, 2019	2.646	2.708	2.639	2.697	2.697	97,901
Feb 20, 2019	2.660	2.689	2.631	2.636	2.636	110,138
Feb 19, 2019	2.621	2.673	2.600	2.662	2.662	151,748
Feb 15, 2019	2.572	2.637	2.543	2.625	2.625	98,409
Feb 14, 2019	2.591	2.613	2.566	2.573	2.573	101,556

Data Disclaimer Help Suggestions
 Privacy (Updated) About Our Ads Terms (Updated) Sitemap
 © 2022 Yahoo. All rights reserved.

Figure A.2: Example of Yahoo finance table for Natural Gas historical data with red boxes around used variables [53]



Precios CO2

Precios CO2	EUA	CER
Media anual	24,84 €	0,22 €
Enero	23,24 €	0,24 €
Febrero	20,99 €	0,23 €
Marzo	21,95 €	0,22 €
Abril	25,67 €	0,23 €
Mayo	25,50 €	0,22 €
Junio	25,24 €	0,20 €
Julio	27,92 €	0,21 €
Agosto	26,93 €	0,21 €
Septiembre	25,75 €	0,21 €

Contactános
 Para solicitar cotización o información adicional, contacte con:
 +34 932 414 723
info@sendeco2.com

Figure A.3: SENDECO2 table for CO2 historical data with red boxes around used variables [42]

The market prices were found at [37]. From this website it is possible to export an Excel spreadsheet portraying the

To identify special days, an *Excel* spreadsheet was created that includes all fixed-date holidays and weekends from 2016 to 2019. The days included in this table are not taken into account when conducting the study.

The data collected by [23] contained a file in which the units registered in *OMIE* were assigned to the agent to which they belonged. This file was used to obtain the entire unit portfolio of each agent.

A.2 OMIE file details

This section presents the file format according to [28] and the examples included in the document. The examples given do not contain real data and should only be taken into account as examples of formats.

A.2.1 CAB Files

4.1.1 Fichero Cabecera de Ofertas que entran en Casación

El nombre del fichero será: cab_aaaamdd.v

Descripción de los campos:

CAMPO	DESCRIPCIÓN	FORMATO	POSICIÓN
CodOferta	Código de la oferta	I7	1
Version	Número de versión	I3	8
Código	Unidad Ofertante	A7	11
Descripción	Descripción de la unidad	A30	18
CV	Indicación de compra o venta	A1 C=compra V=venta	48
Int	(no utilizado)	A1	49
OferPlazo	Indicador de si la oferta proviene de una desagregación de una unidad de contratación a plazo	A1 O= Oferta normal P= Oferta proveniente de una desagregación de UCP	50
No utilizado	No utilizado	F17.3	51
No utilizado	No utilizado	F17.3	68
MaxRamSub	Máxima rampa de subida	F7.1	85
MaxRamBaj	Máxima rampa de bajada	F7.1	92
Fijoeuro	Término fijo en EUR a partir del 1/6/2010 (Término fijo en cEUR hasta el 31/5/2010)	F17.3	99
Vareuro	Término variable en EUR/MWh a partir del 1/6/2010 (Término variable en cEUR/kWh hasta el 31/5/2010)	F17.3	116
MaxPot	Potencia máxima	F7.1 – 0.0 a 99999.9	133
MaxRamArr	Máxima rampa de arranque	F7.1	140
MaxRamPar	Máxima rampa de parada	F7.1	147
CodInt	Código de Interconexión	I2	154
Año	Año de Inserción	I4	156
Mes	Mes de Inserción	I2	160

Versión 1.32 Modelo de Ficheros para la distribución de Información pública del OM Pág. 13



Día	Día de Inserción	I2	162
Hora	Hora de Inserción	I2	164
Minuto	Minuto de Inserción	I2	166
Segundo	Segundo de Inserción	I2	168

Figure A.4: Format of CAB files and description of the variables [28]

The given example for this type of file is:

1107802	PPAAV01	VENTA EXCED. AUTOPROD.	VNO	0.000	0.000	0.0	0.0	0.000	0.000	1094.3	0.0	0.0	120080303204925
412980	CGJ310	C.T. DISEÑO10	VNO	0.000	0.000	4.0	4.0	0.000	5.400	392.7	0.0	0.0	120080529114453
1173456	MMCC4	C.C. MMCC4C14	VNO	0.000	0.000	4.0	4.0	0.000	5.420	379.3	0.0	0.0	120080529114456
149826	TTTTT	C.T. TTTTTT EEE	VNO	0.000	0.000	2.0	2.0	0.000	6.429	365.2	0.0	0.0	120080529114500
149828	KKDK1	C.T. KKKDK1	VNO	0.000	0.000	3.0	3.0	0.000	6.529	563.2	0.0	0.0	120080529114502
149829	KKDK1	C.T. KKKDKR 1	VNO	0.000	0.000	0.0	0.0	0.000	10.424	60.0	0.0	0.0	120080529114507
149830	KKDK2	C.T. KKKDKR 2	VNO	0.000	0.000	1.5	1.5	0.000	7.354	154.4	0.0	0.0	120080529114507
149831	KKDK3	C.T. KKKDKR 3	VNO	0.000	0.000	2.0	2.0	0.000	6.202	364.1	0.0	0.0	120080529114509
464844	SSSA1	C.C. SSS A 1	VNO	0.000	0.000	4.0	4.0	0.000	5.310	394.1	0.0	0.0	120080529114511
464845	SSSA2	C.C. SSS A 2	VNO	0.000	0.000	4.0	4.0	0.000	5.330	389.2	0.0	0.0	120080529114515
1173457	SSSA3	C.C. SSS A 3	VNO	0.000	0.000	4.0	4.0	0.000	5.350	391.0	0.0	0.0	120080529114518
149832	NNNN1	C.T. NNN NNNR 1	VNO	0.000	0.000	1.5	1.5	0.000	7.043	270.0	0.0	0.0	120080529114521
149833	NNNN2	C.T. NNN NNNR 2	VNO	0.000	0.000	2.0	2.0	0.000	6.304	370.7	0.0	0.0	120080529114524
1173458	NNNA1	C.C. NNNA1A 1	VNO	0.000	0.000	4.0	4.0	0.000	5.320	417.3	0.0	0.0	120080529114526
1173459	NNNA2	C.C. NNNA1A 2	VNO	0.000	0.000	4.0	4.0	0.000	5.340	419.6	0.0	0.0	120080529114530
1173460	NNNA3	C.C. NNNA1A 3	VNO	0.000	0.000	4.0	4.0	0.000	5.360	418.6	0.0	0.0	120080529114533
1173461	SSSA3	C.C. SS AAA3	VNO	0.000	0.000	0.0	0.0	0.000	5.360	397.4	0.0	0.0	120080529114540
148167	HHHG	C.H. HHHEL GGGGG.	VNO	0.000	0.000	0.0	0.0	0.000	0.000	215.0	0.0	0.0	120080529114547
149837	KKR1	UGH. PRS KKR1	VNO	0.000	0.000	0.0	0.0	0.000	0.000	382.1	0.0	0.0	120080529114550
149838	KKR2	UGH. PRS KKR2	VNO	0.000	0.000	0.0	0.0	0.000	0.000	847.8	0.0	0.0	120080529114553
149839	KKR3	UGH. PRS KKR3	VNO	0.000	0.000	0.0	0.0	0.000	0.000	349.9	0.0	0.0	120080529114557
149834	PPR1	C.T. PPR1	VNO	0.000	0.000	0.0	0.0	0.000	14.202	115.6	0.0	0.0	120080529115009
1173462	PPPP1	PPR P 01	VNO	0.000	0.000	0.0	0.0	0.000	0.000	65.8	0.0	0.0	120080529120307
1221856	TTT0C1	TTU T C01	CMO	0.000	0.000	0.0	0.0	0.000	0.000	9.0	0.0	0.0	120080728091502
1205340	PPPP1	PPF PFFFFF 1	VNO	0.000	0.000	0.0	0.0	0.000	0.000	193.0	0.0	0.0	120081106175839
1339994	NNN18	NNNNN NNN 18	VNO	0.000	0.000	0.0	0.0	0.000	0.000	5.3	0.0	0.0	120081121174139
686850	NNN03	NNNNN NNN 03	VNO	0.000	0.000	0.0	0.0	0.000	0.000	13.1	0.0	0.0	120081127185033
627842	NNN1	NNNNN NNN 01	VNO	0.000	0.000	0.0	0.0	0.000	0.000	73.5	0.0	0.0	120081210163637
1375453	MMK	MMK	VNO	0.000	0.000	0.0	0.0	0.000	0.000	84.0	0.0	0.0	120081230163341
1291106	PPP2	PPP 2	VNO	0.000	0.000	0.0	0.0	0.000	0.000	408.4	0.0	0.0	120090324091644
1291105	PPP1	PPP 1	VNO	0.000	0.000	0.0	0.0	0.000	0.000	408.6	0.0	0.0	120090324091703
833025	LLL1	LLL 1	CMO	0.000	0.000	0.0	0.0	0.000	0.000	5500.0	0.0	0.0	120090325121113
605927	DDRR	DDRR	CMO	0.000	0.000	0.0	0.0	0.000	0.000	32.0	0.0	0.0	120090408093234

Figure A.5: Example of CAB files format [28]

A.3 DET Files

4.1.2 Fichero Detalle de Ofertas que entran en Casación

El nombre del fichero será: det_aaaaamdd.v

Descripción de los campos:

CAMPO	DESCRIPCIÓN	FORMATO	POSICIÓN
CodOferta	Código de la oferta	I7 – 1 a 9999999	1
Version	Número de versión	I3 – 0 a 999	8
Periodo	Período de la oferta	I2	11
NumBloq	Número de bloque	I2	13
No utilizado	No utilizado	F17.3	15
PrecEuro	Precio en EUR/MWh desde el 1/6/2010 (Precio en cEUR/kWh hasta el 31/5/2010)	F17.3	32
Energía	Energía de la oferta (MWh)	F7.1	49
BloqInd	Indicativo de bloque divisible	A1 S=sí N=no	56
BloqRet	Indicativo de bloque retirable	A1 S=sí N=no	57

Figure A.6: Format of DET files and description of the variables [28]

The given example for this type of file is:

1804728	9 9 1	0.000	5.420	1.0SS
1804893	8 1 1	0.000	81.420	203.3SS
1804893	8 1 2	0.000	180.300	2.9SS
1804893	8 2 1	0.000	81.420	203.3SS
1804893	8 2 2	0.000	180.300	2.9SS
1804893	8 3 1	0.000	0.000	15.0SS
1804893	8 3 2	0.000	180.300	191.2SS
1804893	8 4 1	0.000	0.000	25.0SS
1804893	8 4 2	0.000	180.300	181.2SS
1804893	8 5 1	0.000	0.000	40.0SS
1804893	8 5 2	0.000	180.300	166.2SS

Figure A.7: Example of DET files format [28]

A.4 PDBC Files

The *PDBC* files come divided by country (Portugal or Spain) and since they share the same format were used in the same query.



5.2 Programa diario básico de casación (PDBC)

Fichero con los resultados de la casación del Mercado Diario, conteniendo las energías por unidad ofertante que han resultado casadas en dicho proceso. Este fichero se publicará una vez cumplido el periodo de confidencialidad.

Nombre del fichero: *pdbc_aaaamddd.v* donde *aaaamddd* corresponde a la fecha de sesión y *v* es la versión del fichero.

Descripción de los campos:

CAMPO	DESCRIPCIÓN	VALORES VALIDOS
Año	Año	I4 – 20XX
Mes	Mes	I2 – 1 a 12
Día	Día	I2 – 1 a 31
Hora	Hora	I2 – 1 a 25
Código	Código de Unidad Ofertante	A7
Energía asignada	Energía asignada (MWh)	F7.1 – -99999.9 a 99999.9
No utilizado	Este valor será 0	0
Tipo de oferta	Indicará que tipo de oferta es	I2 – 0 a 99
NumOf	Indicará el número de oferta asociado a la unidad	I8 – 1 a 99999999

Figure A.8: Format of *PDBC* files and description of the variables [28]

The given example for this type of file is:

Ejemplo de contenido del fichero

```

PDBC;
2011;02;05;1;ABENV1;103.9;0;10;2072355;
2011;02;05;1;ABO2;520;0;1;2072217;
2011;02;05;1;ABRE005;8;0;10;2072356;
2011;02;05;1;ACAVADO;368;0;1;2071884;
2011;02;05;1;ACC1EBR;44.1;0;1;2072127;
2011;02;05;1;ACC2EBR;22;0;1;2072128;
2011;02;05;1;ACE3;229;0;1;2072093;
2011;02;05;1;ACE4;220;0;1;2072132;
2011;02;05;1;ADOUINT;800;0;1;2071886;
*
```

Figure A.9: Example of *PDBC* files format [28]

Appendix B

Power BI

B.1 CAB files query

To create the query *CAB*, it is first necessary to organize all the files with this format in a folder, so that it is easier to join them by concatenation. In this way, it is possible to obtain a complete query containing all the offer headings for the desired period between 01/01/2019 and 28/02/2019. Since these files contain information about all the sale and purchase offers, the query had to be filtered to contain only the rows whose values in the ['CV'] column are equal to "V", since these are the values that correspond to the sale offers. Another transformation that must be made is the creation of a ['Date'] column, since this variable is needed to establish relationships with other queries. Although these files have a column of dates, they indicate when the offers were made, which is not relevant to the study because they do not reflect the market session date of the offers. To obtain information about the date of the market session, the file name was used, since its format is fixed and written as "CAB_YYYYMMDD.1" (YYYY corresponds to the year, MM to the month, DD to the day), it is possible to obtain the date of the corresponding market session by querying this information. An example of this is the file CAB_20190101.1, which is translated as 2019-01-01 in the newly created ['Date'] column. Figure B.1 shows a flowchart of the actions performed.

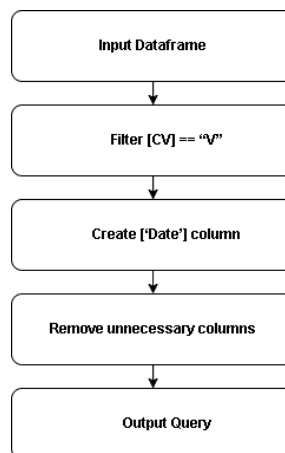


Figure B.1: Flowchart of the major actions taken while creating the *CAB* Query

The class diagram for the query *CAB* is represented in Figure B.2. Also, Table B.1 shows the field headings and first 10 rows of the query.

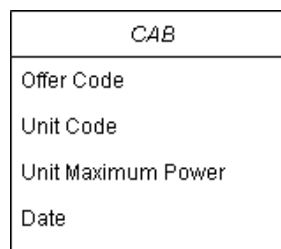


Figure B.2: Class diagram for the *CAB* query

Offer Code	Unit Code	Unit Max Power	Date
5659687	NEXVD21	10	Tuesday, January 1, 2019
5659691	NEXVD34	10	Tuesday, January 1, 2019
5659696	NEXVD45	10	Tuesday, January 1, 2019
5659715	NEXVD70	10	Tuesday, January 1, 2019
5659716	NEXVD71	10	Tuesday, January 1, 2019
5659717	NEXVD72	10	Tuesday, January 1, 2019
5657803	EGVD376	10	Tuesday, January 1, 2019
5659400	CEVD019	10	Tuesday, January 1, 2019
5659404	CEVD051	10	Tuesday, January 1, 2019
5660121	WMVD024	10	Tuesday, January 1, 2019
5660122	WMVD025	10	Tuesday, January 1, 2019

Table B.1: Header and example first 10 rows of the *CAB* Query

B.2 DET files query

To create the query *DET*, the files were first concatenated in the same way as the query *CAB*. In these files, the hour of the market session is represented in the ['Session'] column. This column represents the hours in a range of [1,25], and since *Power BI* requires that columns of type Time be represented in the format [0,24], 1 had to be subtracted from each value in this column.

Since the format of the filename is fixed and is written as "DET_YYYYMMDD.1" (YYYY corresponds to the year, MM to the month, DD to the day), it is possible to obtain the date of the corresponding market session by processing the name in the same way as in the query *CAB*. An example of this is the file DET_20190101.1, which is translated as 2019-01-01 in the newly created column ['Date']. It is then possible to create a ['Date Time'] column by merging the ['Session'] and ['Date'] columns together.

Figure B.3 presents a flowchart of actions taken to transform this query.

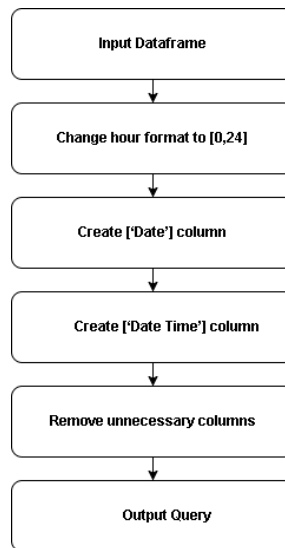


Figure B.3: Flowchart of the major actions taken while creating the *DET* Query

The class diagram for the query *DET* is represented in Figure B.4. Also, Table B.2 shows the field headings and first 10 rows of the query.

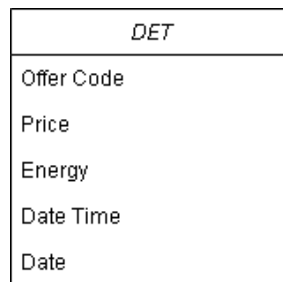


Figure B.4: Class diagram for the *DET* query

Offer Code	Price	Energy	Date Time	Date
781186	0	0.1	1/1/2019 0:00	1/1/2019
781186	0	0.1	1/1/2019 1:00	1/1/2019
781186	0	0.1	1/1/2019 2:00	1/1/2019
781186	0	0.1	1/1/2019 3:00	1/1/2019
781186	0	0.1	1/1/2019 4:00	1/1/2019
781186	0	0.1	1/1/2019 5:00	1/1/2019
781186	0	0.1	1/1/2019 6:00	1/1/2019
781186	0	0.1	1/1/2019 7:00	1/1/2019
781186	0	0.1	1/1/2019 8:00	1/1/2019
781186	0	0.1	1/1/2019 9:00	1/1/2019

Table B.2: Header and example first 10 rows of the *DET* Query

B.3 PDBC files query

To create the query *PDBC*, the files were concatenated with this format in the same way as was done previously for the queries *CAB* and *DET*. This query must also perform the same transformation as was done in the *DET* query in the ['Session'] column, so that the hourly values are represented in a [0,24] format.

In addition, the ['Year','Month','Day'] columns in the *PDBC* files were used to create a ['Date'] column. Figure B.5 shows a flowchart for the actions performed.

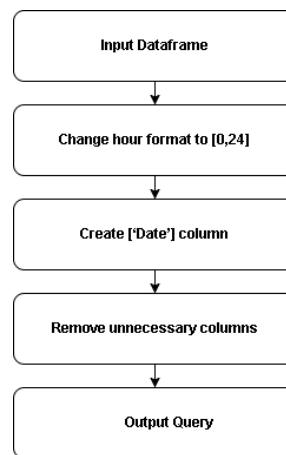


Figure B.5: Flowchart of the major actions taken while creating the *PDBC* Query

The class diagram for the query *PDBC* is represented in Figure B.6. Also, Table B.3 shows the field headings and first 10 rows of the query.

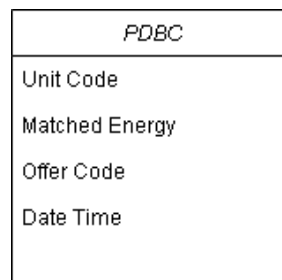


Figure B.6: Class diagram for the *PDBC* query

Unit Code	Matched Energy	Offer Code	Date Time	Date
FORZC01	-0.1	5369420	1/11/2019 0:00	1/1/2019
FORZC01	-0.1	5369420	1/11/2019 1:00	1/1/2019
FORZC01	-0.1	5369420	1/11/2019 2:00	1/1/2019
FORZC01	-0.1	5369420	1/11/2019 3:00	1/1/2019
FORZC01	-0.1	5369420	1/11/2019 4:00	1/1/2019
FORZC01	-0.1	5369420	1/11/2019 5:00	1/1/2019
FORZC01	-0.1	5369420	1/11/2019 6:00	1/1/2019
FORZC01	-0.1	5369420	1/11/2019 7:00	1/1/2019
FORZC01	-0.1	5369420	1/11/2019 8:00	1/1/2019
FORZC01	-0.1	5369420	1/11/2019 9:00	1/1/2019

Table B.3: Header and example first 10 rows of the *PDBC* Query

B.4 Market Price

The query *Market Price* is created using the file *Market Price* created with the information obtained in [37]. To filter the query *Market Price* to keep only the day-ahead session, only the rows where the column ['Session'] is 0 were kept.

The time variable represented by the ['Hour'] column must be transformed to a [0,24] format, similar to the transformation done in the *DET* query for the ['Session'] column.

The condition in which the transmission system is congested and Spanish and Portuguese market prices do not coincide is called market splitting. To represent when this condition occurs, a column with binary values ['Market Splitting'] was created (0 means there is no market splitting, 1 means there is market splitting).

Figure B.7 shows a flowchart for the decisions made during the creation and transformation of this query.

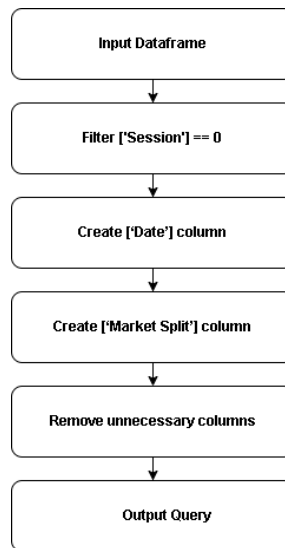


Figure B.7: Flowchart of the major actions taken while creating the *Market Price* Query

The class diagram for the query *Market Price* is represented in Figure B.8. Also, Table B.4 shows the field headings and first 10 rows of the query.



Figure B.8: Class diagram for the *Market Price* query

Price PT	Price SP	Date Time	Market Splitting	Date
24.57	24.57	1/1/2016 23:00	0	1/1/2019
40.66	40.66	1/2/2016 23:00	0	1/1/2019
21.01	21.01	1/3/2016 23:00	0	1/1/2019
23.9	23.9	1/4/2016 23:00	0	1/1/2019
32	32	1/5/2016 23:00	0	1/1/2019
18.11	18.11	1/6/2016 23:00	0	1/1/2019
28.76	28.76	1/7/2016 23:00	0	1/1/2019
36.1	36.1	1/8/2016 23:00	0	1/1/2019
18.1	18.1	1/10/2016 23:00	0	1/1/2019
28.1	28.1	1/11/2016 23:00	0	1/1/2019

Table B.4: Header and example first 10 rows of the *Market Price* Query

B.4.1 PDBC-STOTA files query

To create the query *PDBC-STOTA*, all files had to be concatenated, as had been done with other queries.

From this query, only renewable energy production and demand in *Mibel* for a given hour were considered as relevant variables. In this way, the query was filtered to retain only the following fields relevant to the study:

- **941**- This field contains information on the hourly production of all Hydroelectric plants in the MIBEL market. Both conventional hydroelectric power plants and pumped storage power plants are represented in this field.
- **943**- This field contains the total hourly production of special regime units for the MIBEL market. It contains information about market and distribution special regime units.
- **959**- This field contains the total hourly demand in the MIBEL market.

Further transformations were applied to this query to establish relationships with the other queries. The queries obtained so far represent the date-time by a single column. The original query *PDBC-STOTA* represents the values scattered over hourly columns. A way must be found to better represent these fields by a single column rather than multiple columns. To do this, it was necessary to unpivot the hourly columns. Unpivoting creates two new columns, ['Hour'] and ['Value'], so that the data can be represented by the ['Field'], ['Hour'] and ['Value'] columns. In this way it is possible to split the unique fields into separate columns with the corresponding values by pivoting the ['Field'] column.

The following columns represented by hourly values were determined:

- Hydro Production
- Special Regime Production
- Demand

Although the unpivot and pivot actions solved the data shape problem, each of the newly created columns has null values in different rows than the others. This makes it difficult to filter the values and shape them properly. To solve this problem, 3 new queries had to be created to filter these columns individually. Figure B.9 shows a flowchart of the main actions performed to get the desired queries.

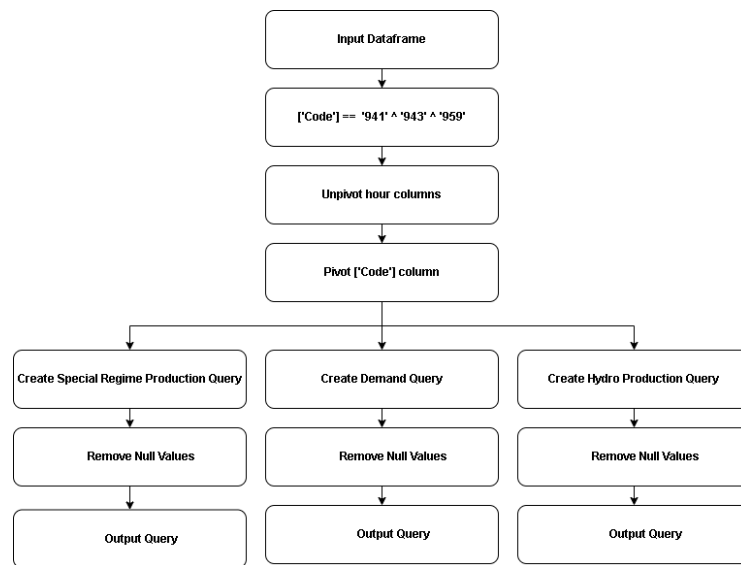


Figure B.9: Flowchart of the major actions taken while creating the *PDBC-STOTA* Query

The 3 newly created queries consist of the columns ['Field value'] and ['Date Time']. To illustrate the queries formed, the Tables B.5, B.6, B.7 show the headings and the first 10 rows of the queries.

Hour	Special Regime Production	Date
00:00	12868,3	1/1/2019
01:00	10439,2	1/1/2019
02:00	10069,4	1/1/2019
03:00	9890,9	1/1/2019
04:00	9724,5	1/1/2019
05:00	9585,5	1/1/2019
06:00	9455,3	1/1/2019
07:00	9476,0	1/1/2019
08:00	9494,6	1/1/2019
09:00	9614,6	1/1/2019

Table B.5: Header and example first 10 rows of the *Special Regime Production* Query

Hour	Hydro Production	Date
00:00	2307,3	1/1/2019
01:00	3299,3	1/1/2019
02:00	3111,7	1/1/2019
03:00	2467,2	1/1/2019
04:00	1421,0	1/1/2019
05:00	1012,4	1/1/2019
06:00	1235,2	1/1/2019
07:00	1213,4	1/1/2019
08:00	1359,5	1/1/2019
09:00	1389,3	1/1/2019

Table B.6: Header and example first 10 rows of the *Hydro Production* Query

Hour	Demand	Date
00:00	23188,8	1/1/2019
01:00	21044,9	1/1/2019
02:00	19792,4	1/1/2019
03:00	18724,4	1/1/2019
04:00	17310,8	1/1/2019
05:00	16687,7	1/1/2019
06:00	16597,0	1/1/2019
07:00	16741,7	1/1/2019
08:00	17097,4	1/1/2019
09:00	17374,2	1/1/2019

Table B.7: Header and example first 10 rows of the *Demand* Query

B.4.2 Other queries

The queries *Agent*, *Gas Price*, and *CO2 Price* did not require any special treatment, since all that was needed to create them was to import the data from the file sources and remove unnecessary columns.

Date	High	Low
4/29/2019	2.598	2.546
4/26/2019	2.58	2.477
4/25/2019	2.52	2.439
4/24/2019	2.487	2.45
4/23/2019	2.535	2.447
4/22/2019	2.535	2.484
4/18/2019	2.521	2.478
4/17/2019	2.582	2.503
4/16/2019	2.594	2.57
4/15/2019	2.653	2.585

Table B.8: Header and example first 10 rows of the *Natural Gas Price Query*

EUA	SPREAD (EUA - CER)	Date Formatted
24.98	2474	1/2/2019
23.03	2279	1/3/2019
23.49	2325	1/4/2019
22.1	2186	1/7/2019
22.62	2238	1/8/2019
21.89	2164	1/9/2019
21.94	217	1/10/2019
22.59	2235	1/11/2019
22.4	2217	1/14/2019
22.46	2222	1/15/2019

Table B.9: Header and example first 10 rows of the *Co2 Price Query*

In addition, an *Aux Date* query was created to import the information from the datasheet, which contains a calendar with binary values that have a value of 1 for all holidays and weekends, and 0 for all other events. In this way, it is possible to exclude these events from the other queries by filtering for ['Special Day']. A day of the week column has also been created, which displays the days of the week as ['Weekday']=['0': Monday, '1':Tuesday, '2':Wednesday, '3':Thursday, '4':Friday, '5':Saturday, '6':Sunday].

Appendix C

Pre-processing: Python code

C.1 *Data_Pre_Processing* class

```
class Data_Pre_Processing():
    def __init__(self, price_intervals):
        self.df = pd.read_csv('Query_Tese.csv')
        self.df['Buckets'] , self.interv = pd.qcut(
            x = self.df['Price'], q = price_intervals, precision = 0,
            duplicates = 'drop', retbins = True)
```

C.2 Filtering by hour and agent

```
#filtered_agent_df-filters dataframe per agent
#filtered_hour_df-filters the filtered_agent_df per hour
#filtered_days- array of unique days where offers have been made
#special_var_df- creates dataframe containing non offer price related variables
filtered_agent_df = self.df[self.df['AGENTE PROPIETARIO'].str.contains(agent)].
                        reset_index(drop=True)
filtered_hour_df = filtered_agent_df[filtered_agent_df['Date Time'].str.contains(
                        hour)]
filtered_date = filtered_hour_df['Date Time'].drop_duplicates().reset_index(drop=
                        True)
special_var_df = filtered_hour_df.iloc[:,2:].groupby(by=['Date Time'], as_index =
                        False).agg({'Code':'first', 'Unit Maximum
                        Power':'max' ,
                        'Month SIN':'first' , 'Month COS':'first' , 'Day SIN':'first' , 'Day COS':'first' ,
                        'Weekday Sin':'first' ,
                        'Weekday COS':'first' , 'Price PT':'first' , 'Price SP':'first' , 'Market Splitting
                        ':'first' , 'Co2 EUA':'first' ,
```

```
'SPREAD (EUA - CER)':'first' , 'Gas Price.High':'first' , 'Gas Price.Low':'first' ,
                                'Total Special Regime':'first' ,
'Demand':'first','Total Hydro':'first'}).drop_duplicates()
special_var_df['Agent'] = agent
special_var_df = special_var_df.drop(['Code'],axis=1)
```

C.3 Process class

```
for x in filtered_date:
    aux = filtered_hour_df.loc[filtered_hour_df['Date Time'] == x]
    aux = aux.groupby(by=['Date Time', 'Price'],as_index=False).sum('Energy').
            sort_values('Price').reset_index(drop
            =True)

    energy = aux['Energy'].to_numpy()
    price = aux['Price'].to_numpy()
    interval_df = pd.DataFrame(columns = ["Price", "Energy", "Date Time"])
    interval_df['Price'] = interv.tolist()
    interval_df.loc[:, 'Date Time'] = x
    for z in price:
        if z == interv[0]:
            lower_limit = interv[0]
            upper_limit = interv[1]

        else:
            lower_limit = interv[np.where(interv < z)][-1]
            upper_limit = interv[np.where(interv >= z)][0]

        pass

    index = np.where(price == z)
    quantity_lower_limit = (1-(z-lower_limit)/(upper_limit-lower_limit)) *
                            energy[index[0]]
    quantity_upper_limit = ((z-lower_limit)/(upper_limit-lower_limit)) * energy
                            [index[0]]
    l=interval_df.loc[interval_df['Price'] == lower_limit,['Energy']]
    u=interval_df.loc[interval_df['Price'] == upper_limit,['Energy']]
    interval_df.loc[interval_df['Price'] == lower_limit, ["Energy"]] = l['
        Energy'].fillna(0).to_numpy() + (
        quantity_lower_limit)/
        max_quantity
    interval_df.loc[interval_df['Price'] == upper_limit, ["Energy"]] = u['
        Energy'].fillna(0).to_numpy() + (
        quantity_upper_limit)/
        max_quantity
```

```
treated_DF = pd.concat([treated_DF, interval_df], ignore_index=True)
```

C.4 Creating the cumulative curves

```
filtered_day = treated_DF['Date Time'].drop_duplicates().reset_index(drop=True)
#for each date-hour present in the treated_DF calculates the commulative sum of the
#price offers and appends the array in
#data

#creating a commulative curves dataframe
for x in filtered_day:
    aux = treated_DF.loc[treated_DF['Date Time'] == x]
    aux = aux.sort_values('Price').reset_index(drop=True)
    r = aux['Energy'].cumsum().to_list()
    r.extend([x, agent])
    bb = pd.DataFrame([r], columns = column)
    data = pd.concat([data, bb], ignore_index=True)
#filling na values with the quantity assigned to the closest lower price
data = data.fillna(axis=1, method = 'ffill')
data = data.fillna(0).drop_duplicates().reset_index(drop = True)
```

C.5 Multi-agents scenario

```
def multiagents(self, agents, hour):

    for x in agents:
        aux, aux_spec = self.process(x, hour)
        if agents.index(x) == 0:
            data = pd.DataFrame().reindex_like(aux).dropna()
            special_data = pd.DataFrame().reindex_like(aux_spec).dropna()

        data = pd.concat([data, aux], ignore_index=True)
        special_data = pd.concat([special_data, aux_spec], ignore_index=True)

    return data, special_data
```

C.6 Multi-hour scenario

```
def multihours(self , agent , hours):  
  
    for x in hours:  
        aux, aux_spec = self.process(agent , x)  
        if hours.index(x) == 0:  
            data = pd.DataFrame(columns = aux.columns)  
            special_data = pd.DataFrame().reindex_like(aux_spec).dropna()  
  
        data = pd.concat([data,aux])  
        special_data = pd.concat([special_data, aux_spec])  
  
    return data, special_data
```


Appendix D

Pre-processing results

D.1 Commulative curves data framework

Curves	Price(€/MWh)												
	0	0.51	8	40	50	54.61	59	62.89	67.07	71.07	80	160	180.3
0	0.2077	0.2114	0.2117	0.2789	0.3514	0.3584	0.3734	0.3951	0.4163	0.4163	0.4724	0.5730	0.5730
1	0.2096	0.2217	0.2225	0.2374	0.3391	0.3559	0.3559	0.3596	0.3766	0.3975	0.4497	0.5463	0.5463
2	0.1741	0.1741	0.1741	0.1753	0.2738	0.3168	0.3168	0.3169	0.3218	0.3612	0.4506	0.5412	0.5412
3	0.2005	0.2103	0.2109	0.2124	0.2515	0.2817	0.2957	0.3129	0.3138	0.3469	0.4674	0.5694	0.5694
4	0.2165	0.2267	0.2275	0.2289	0.2681	0.3140	0.3324	0.3549	0.3550	0.3986	0.5005	0.5978	0.5978
5	0.2166	0.2246	0.2252	0.2326	0.3374	0.3651	0.3651	0.3708	0.3924	0.4119	0.4731	0.5638	0.5638
6	0.1870	0.2192	0.2214	0.2253	0.2951	0.3570	0.3585	0.3603	0.3639	0.4051	0.4744	0.5651	0.5651
7	0.1517	0.1801	0.1820	0.2166	0.2785	0.2972	0.2979	0.3101	0.3294	0.3396	0.3846	0.4714	0.4714
8	0.1629	0.1852	0.2040	0.2349	0.2897	0.3037	0.3037	0.3081	0.3277	0.3496	0.3896	0.4738	0.4738
9	0.1721	0.1955	0.1971	0.2182	0.2810	0.3050	0.3050	0.3051	0.3194	0.3371	0.3940	0.5065	0.5065
10	0.1742	0.1767	0.1769	0.1799	0.2311	0.2585	0.2882	0.2969	0.2977	0.3366	0.4236	0.5157	0.5157
11	0.1264	0.1289	0.1291	0.1301	0.2100	0.2474	0.2474	0.2474	0.2513	0.2902	0.3686	0.4638	0.4638
12	0.1282	0.1282	0.1282	0.1287	0.2220	0.2618	0.2618	0.2618	0.2664	0.3062	0.3921	0.4881	0.4881
13	0.1264	0.1289	0.1291	0.1301	0.2100	0.2474	0.2474	0.2474	0.2522	0.2903	0.3686	0.4637	0.4637
14	0.2082	0.2240	0.2251	0.2419	0.3381	0.3575	0.3575	0.3610	0.3758	0.3896	0.4382	0.5375	0.5375
15	0.2212	0.2249	0.2253	0.2939	0.3646	0.3717	0.3796	0.3965	0.4149	0.4153	0.4711	0.5573	0.5573
16	0.1742	0.1767	0.1769	0.1791	0.2297	0.2585	0.2882	0.2969	0.2983	0.3377	0.4205	0.5125	0.5125
17	0.1591	0.1648	0.1652	0.2232	0.2989	0.3057	0.3176	0.3336	0.3550	0.3558	0.4024	0.4972	0.4972
18	0.2460	0.2516	0.2520	0.2535	0.2929	0.3224	0.3435	0.3650	0.3650	0.4087	0.4946	0.5911	0.5911
19	0.1288	0.1288	0.1288	0.1292	0.2232	0.2641	0.2641	0.2641	0.2686	0.3084	0.3932	0.4888	0.4888
20	0.1283	0.1283	0.1283	0.1307	0.2318	0.2710	0.2710	0.2710	0.2790	0.3168	0.3979	0.4883	0.4883
21	0.1500	0.1561	0.1566	0.2120	0.2852	0.2915	0.3012	0.3183	0.3410	0.3418	0.3819	0.4752	0.4752
22	0.2123	0.2256	0.2265	0.2572	0.3347	0.3481	0.3481	0.3530	0.3708	0.3905	0.4464	0.5467	0.5467
23	0.2122	0.2159	0.2161	0.2848	0.3559	0.3740	0.3897	0.4039	0.4148	0.4148	0.4697	0.5662	0.5662
24	0.1933	0.2096	0.2107	0.2602	0.3311	0.3364	0.3371	0.3527	0.3736	0.3788	0.4316	0.5259	0.5259
25	0.1845	0.2142	0.2162	0.2181	0.2674	0.3387	0.3402	0.3450	0.3608	0.3889	0.4593	0.5637	0.5637
26	0.1646	0.2057	0.2100	0.2327	0.3117	0.3304	0.3304	0.3346	0.3549	0.3722	0.4114	0.4928	0.4928
27	0.2096	0.2108	0.2109	0.2643	0.3320	0.3372	0.3398	0.3552	0.3800	0.3853	0.4472	0.5506	0.5506
28	0.2161	0.2183	0.2185	0.2198	0.2813	0.3461	0.3467	0.3569	0.3809	0.4011	0.4749	0.5680	0.5680
29	0.1926	0.2067	0.2077	0.2103	0.2922	0.3260	0.3260	0.3260	0.3298	0.3688	0.4472	0.5424	0.5424
30	0.2049	0.2074	0.2076	0.2094	0.2615	0.3042	0.3309	0.3391	0.3403	0.3799	0.4579	0.5492	0.5492
31	0.2647	0.2672	0.2674	0.2692	0.3214	0.3630	0.3897	0.3965	0.3981	0.4370	0.5276	0.6219	0.6219
32	0.2213	0.2250	0.2254	0.2941	0.3652	0.3733	0.3885	0.4047	0.4231	0.4235	0.4789	0.5649	0.5649
33	0.1591	0.1633	0.1637	0.2275	0.2993	0.3059	0.3155	0.3317	0.3520	0.3528	0.4058	0.5015	0.5015
34	0.1741	0.1741	0.1741	0.1765	0.2776	0.3168	0.3168	0.3169	0.3237	0.3615	0.4440	0.5345	0.5345
35	0.1493	0.1555	0.1560	0.2052	0.2729	0.2792	0.2882	0.3021	0.3244	0.3295	0.3740	0.4752	0.4752
36	0.2165	0.2211	0.2214	0.2293	0.3346	0.3623	0.3623	0.3680	0.3902	0.4097	0.4791	0.5698	0.5698
37	0.1408	0.1997	0.2061	0.2280	0.2971	0.3158	0.3158	0.3204	0.3398	0.3617	0.4036	0.4937	0.4937

Table D.1: Preprocessed dataframe for *ENDESA GENERACION, S.A* at hour 09:00

Curves	Price(€/MWh)												
	0	0.51	8	40	50	54.61	59	62.89	67.07	71.07	80	160	180.3
0	0.0790	0.0934	0.0948	0.0967	0.0992	0.1462	0.2531	0.3659	0.3964	0.4153	0.4512	0.5820	0.5846
1	0.0797	0.0935	0.0941	0.0943	0.0943	0.0943	0.1036	0.1399	0.2108	0.3507	0.4439	0.5887	0.5911
2	0.0753	0.0968	0.1005	0.1027	0.1027	0.1027	0.1071	0.1229	0.1923	0.3091	0.4352	0.6438	0.6449
3	0.0363	0.0665	0.0762	0.0785	0.0785	0.0785	0.0785	0.0902	0.1709	0.2880	0.3194	0.6758	0.6950
4	0.0874	0.1208	0.1268	0.1291	0.1291	0.1291	0.1291	0.1429	0.2241	0.3594	0.4845	0.6865	0.6876
5	0.0733	0.1520	0.1727	0.1730	0.1730	0.1730	0.1815	0.2447	0.3119	0.3845	0.4731	0.6326	0.6337
6	0.0614	0.0732	0.0738	0.0741	0.0741	0.0741	0.0797	0.1222	0.1956	0.3057	0.4012	0.5808	0.5819
7	0.0845	0.1020	0.1026	0.1028	0.1028	0.1248	0.1490	0.2298	0.3733	0.3977	0.4465	0.6163	0.6174
8	0.0842	0.1014	0.1020	0.1022	0.1022	0.1037	0.1214	0.1570	0.2926	0.3879	0.4524	0.6252	0.6263
9	0.0828	0.0997	0.1003	0.1006	0.1006	0.1011	0.1148	0.1399	0.2157	0.3669	0.4437	0.6111	0.6122
10	0.0803	0.0929	0.0935	0.0938	0.0938	0.0938	0.0938	0.1155	0.1953	0.3332	0.4563	0.6578	0.6589
11	0.0817	0.0955	0.0968	0.0990	0.0990	0.0990	0.1041	0.1198	0.1782	0.3062	0.4483	0.6543	0.6554
12	0.0808	0.0956	0.0969	0.0991	0.0991	0.0991	0.1035	0.1193	0.1796	0.3106	0.4495	0.6548	0.6559
13	0.0693	0.0800	0.0813	0.0834	0.0834	0.0858	0.0897	0.1066	0.1809	0.3287	0.4451	0.6506	0.6517
14	0.0784	0.0941	0.0947	0.0949	0.0949	0.0949	0.1015	0.1397	0.1899	0.2882	0.3855	0.5167	0.5178
15	0.0798	0.0942	0.0956	0.0974	0.1002	0.1414	0.2675	0.3555	0.3855	0.4063	0.4508	0.5986	0.6012
16	0.0870	0.1016	0.1022	0.1025	0.1025	0.1025	0.1025	0.1169	0.1913	0.3253	0.4655	0.6585	0.6596
17	0.0791	0.0926	0.0938	0.0956	0.0964	0.1194	0.2207	0.3572	0.4056	0.4152	0.4762	0.6035	0.6036
18	0.0873	0.1132	0.1168	0.1191	0.1191	0.1191	0.1191	0.1267	0.2257	0.3398	0.4648	0.6717	0.6728
19	0.0756	0.0892	0.0905	0.0927	0.0927	0.0927	0.0971	0.1129	0.1769	0.3004	0.4298	0.6210	0.6221
20	0.0703	0.0827	0.0833	0.0836	0.0836	0.0836	0.0880	0.1162	0.1740	0.3001	0.4275	0.6234	0.6245
21	0.0848	0.0995	0.1007	0.1025	0.1035	0.1227	0.2210	0.3496	0.4029	0.4136	0.4761	0.6054	0.6055
22	0.0737	0.0883	0.0888	0.0891	0.0891	0.0909	0.1078	0.1299	0.2538	0.3692	0.4363	0.5875	0.5886
23	0.0796	0.0940	0.0954	0.0973	0.1042	0.1536	0.2538	0.3664	0.3970	0.4218	0.4518	0.5927	0.5954
24	0.0896	0.1081	0.1094	0.1115	0.1115	0.1288	0.1731	0.2866	0.4349	0.4522	0.4798	0.6408	0.6435
25	0.0825	0.1196	0.1264	0.1267	0.1267	0.1267	0.1354	0.1630	0.2345	0.3595	0.4566	0.6301	0.6312
26	0.0848	0.1008	0.1014	0.1016	0.1016	0.1288	0.1408	0.1827	0.3469	0.4095	0.4469	0.6147	0.6158
27	0.0871	0.1043	0.1046	0.1047	0.1050	0.1242	0.2171	0.3450	0.3969	0.4181	0.4805	0.6299	0.6325
28	0.0820	0.1261	0.1352	0.1355	0.1355	0.1355	0.1355	0.1629	0.2449	0.3667	0.4735	0.6866	0.6877
29	0.0828	0.0966	0.0979	0.1001	0.1001	0.1001	0.1052	0.1231	0.1950	0.3199	0.4524	0.6530	0.6541
30	0.0928	0.1135	0.1156	0.1159	0.1159	0.1159	0.1159	0.1282	0.2005	0.3103	0.4679	0.6589	0.6600
31	0.0662	0.0887	0.0928	0.0951	0.0951	0.0951	0.0951	0.1067	0.1674	0.2704	0.4133	0.6872	0.6883
32	0.0800	0.0944	0.0957	0.0976	0.1009	0.1498	0.2481	0.3557	0.3910	0.4129	0.4482	0.5789	0.5816
33	0.0814	0.0949	0.0963	0.0983	0.1012	0.1247	0.2234	0.3541	0.3982	0.4094	0.4687	0.6275	0.6276
34	0.0745	0.0893	0.0910	0.0931	0.0931	0.0931	0.0975	0.1164	0.1768	0.2972	0.4136	0.6422	0.6433
35	0.0861	0.1012	0.1015	0.1016	0.1016	0.1119	0.2223	0.3474	0.3888	0.4156	0.4849	0.6254	0.6280
36	0.0724	0.1622	0.1863	0.1866	0.1866	0.1866	0.1884	0.2142	0.2876	0.3987	0.4896	0.6432	0.6443
37	0.0849	0.1009	0.1015	0.1018	0.1018	0.1152	0.1410	0.1757	0.3323	0.4061	0.4471	0.6362	0.6373

Table D.2: Preprocessed dataframe for *IBERDROLA GENERACION ESPANA S.A* at hour 09:00

Appendix E

ITMS code

E.1 Update

```
def update(self):
    sigma_linha = self.sigma*np.sqrt(2)
    ms_df = pd.DataFrame(columns = self.data.columns)
    x = self.data.copy(deep = True)
    initial_x = self.data.copy(deep = True)
    initial_graph_x = initial_x.copy(deep = True)
    new_x = pd.DataFrame().reindex_like(self.data).fillna(0.1)
    stop_criteria = 0
    max_distance = 1
    while (stop_criteria < 50) and (max_distance > 0.000001):
        c1 = self.C1(x)
        c2 = self.C2(x, initial_x)
        for index, xi in x.iterrows():
            sum_1 = 0
            sum_2 = pd.Series(0, index = self.data.columns)
            sum_3 = 0
            sum_4 = pd.Series(0, index = self.data.columns)
            for index1, xj in x.iterrows():
                t = (np.linalg.norm(xi.to_numpy()-xj.to_numpy())/(sigma_linha))**2
                aux = self.K_x(t)
                sum_1 += aux
                sum_2 = sum_2.add(aux*xj)

            for index2, x0j in initial_x.iterrows():
                t2 = (np.linalg.norm(xi.to_numpy()-x0j.to_numpy())/(sigma_linha))**2
                aux2 = self.K_x(t2)
                sum_3 += aux2
                sum_4 = sum_4.add(aux2*x0j)

            new_x.iloc[index] = (c1*sum_2 + c2*sum_4)/(c1*sum_1 + c2*sum_3)
```

```

ms_df = pd.concat([ms_df,x])
distance = []
for n in range (len(x)):
    a = x.iloc[n]
    b = new_x.iloc[n]
    distance.append( np.linalg.norm(a.to_numpy()-b.to_numpy()))

x = new_x.copy(deep = True)
max_distance = max(distance)
stop_criteria += 1

final_x_graph = x.copy(deep=True)

return initial_graph_x, final_x_graph, ms_df

```

E.2 best parameters

```

def findbestparameters(self):
    sigma = 0.01
    lambd = 0
    ms_parameters = Mean_Shift(self.data,sigma,lambd)
    fig, axs = plt.subplots(5, 4, sharex=True, sharey=True, figsize = (15,15))

    for x in range (5):
        for y in range(4):
            initial_graph_x, final_x_graph, ms_df = ms_parameters.update()
            for xi in range (len(final_x_graph)):
                axs[x,y].step( final_x_graph.columns, final_x_graph.iloc[xi],
                               where='post')
                axs[x,y].set_title('Lamda:' +str(ms_parameters.lambd)+'\nSigma:' +
                                   str(ms_parameters.sigma))

            ms_parameters.lambd += 0.5
    ms_parameters.lambd = 0
    ms_parameters.sigma += 0.02
    fig.add_subplot(111, frame_on=False)
    plt.tick_params(labelcolor="none", bottom=False, left=False)
    plt.xlabel('Price (Eur/MW)')
    plt.ylabel('Energy (MW/MW)')
    return fig

```

Appendix F

Data Validation

F.1 Cluster representation

F.1.1 Euclidean Distance

```
def defineclusters(self, final_x):
    alphabet = list(string.ascii_uppercase)
    filtered_clusters = final_x.copy(deep = True)
    aux = final_x.copy(deep = True)
    i=0
    final_x['Cluster'] = None
    for ind, xi in aux.iterrows():
        cluster= i #alphabet[i]
        x= final_x.iloc[ind]['Cluster']
        if x == None:
            final_x.at[ind, 'Cluster'] = i
            for index1, xj in aux.iterrows():
                distance = np.linalg.norm(xi.to_numpy()-xj.to_numpy())
                y = final_x.iloc[index1]['Cluster']
                if (distance < 0.07) and (final_x.iloc[index1]['Cluster'] == None):
                    final_x.at[index1, 'Cluster'] = i
            i+=1
    filtered_clusters = final_x.copy(deep = True).drop_duplicates(subset=['Cluster']).
                                                                    reset_index(drop=True)
    return final_x, filtered_clusters
```

F.1.2 ITMS

.....

F.2 Classification: Modified ITMS

```
def classifyclusters(self, df_to_classify, df_clusters):
    #make sure both df have same columns
    df_to_classify.reset_index(drop=True, inplace=True)
    ms_df = pd.DataFrame(columns = df_to_classify.columns)
    sigma_linha = self.sigma*np.sqrt(2)
    x = pd.concat([df_clusters, df_to_classify])
    initial_x = x.copy(deep=True)
    new_x = pd.DataFrame(columns = df_to_classify.columns, index = pd.RangeIndex(0,
                                                                                  len(df_to_classify), step = 1))

    iteration = 0
    max_distance = 1
    while (iteration < 50) and (max_distance > 0.000001):
        c1 = self.C1(x)
        c2 = self.C2(x, initial_x)
        for index, xi in df_to_classify.iterrows():
            sum_1 = 0
            sum_2 = pd.Series(0, index = df_to_classify.columns)
            sum_3 = 0
            sum_4 = pd.Series(0, index = df_to_classify.columns)
            for index1, xj in x.iterrows():
                t = (np.linalg.norm(xi.to_numpy()-xj.to_numpy())/(sigma_linha))**2
                aux = self.K_x(t)
                sum_1 += aux
                sum_2 = sum_2.add(aux*xj)

            for index2, x0j in initial_x.iterrows():
                t2 = (np.linalg.norm(xi.to_numpy()-x0j.to_numpy())/(sigma_linha))**
                    2
                aux2 = self.K_x(t2)
                sum_3 += aux2
                sum_4 = sum_4.add(aux2*x0j)

            new_x.iloc[index] = (c1*sum_2 + c2*sum_4)/(c1*sum_1 + c2*sum_3)

    ms_df = pd.concat([ms_df, x])
    distance = []
    #new_x = pd.concat([df_clusters, x], ignore_index=True)
    x.reset_index(drop = True, inplace = True)
    #new_x.reset_index(drop= True, inplace = True)
    for n in range (len(new_x)):
        a = x.iloc[len(x)-len(new_x)+n]
        b = new_x.iloc[n]
        distance.append( np.linalg.norm(a.to_numpy()-b.to_numpy()))

    x.iloc[(len(x)-len(new_x)):] = new_x.copy(deep = True)
    max_distance = max(distance)
```

```
iteration += 1

final_x = x.copy(deep=True)

return initial_x, final_x, ms_df
```

F.3 Explanatory variables

F.3.1 Cross-entropy

```
def findcrossentropy(self, initial_data , special_variables , clusters):
    cross_entropy_df = pd.DataFrame(columns = ['Cross Entropy', 'Variables'])
    ini_entropy = self.V_x(initial_data, len(initial_data))
    clusters_entropy = self.V_x(clusters, len(clusters))
    clusters_initial_data_cross_entropy = -np.log(self.V_x_xo(initial_data,
                                                                clusters, len(initial_data)))

    agent = special_variables['Agent'].unique()
    special_variables['Agent'].replace(to_replace = agent, value = list(range(len(
        agent))), inplace = True)

    special_variables = special_variables.drop(['Date Time'],
        axis = 'columns')

    special_variables = special_variables.fillna(0)
    for x in range (len(special_variables.columns)):
        if x == 0:
            aux_best_variables_df = pd.concat([clusters , special_variables.iloc[:,
                x]], axis = 'columns')

            aux_initial_data = pd.concat([initial_data , special_variables.iloc[:,x
                ]], axis = 'columns')

        elif x == 2 or x == 4 or x==6:
            continue
        elif x == 3 or x == 5 or x ==7:
            aux_best_variables_df = pd.concat([clusters , special_variables.iloc[:,
                np.r_[x-1,x]]], axis = '
                columns')

            aux_initial_data = pd.concat([initial_data , special_variables.iloc[:,
                np.r_[x-1,x]]], axis = '
                columns')

        else:
            aux_best_variables_df = pd.concat([clusters , special_variables.iloc[:,
                x]], axis = 'columns')

            aux_initial_data = pd.concat([initial_data , special_variables.iloc[:,x
                ]], axis = 'columns')
```

```

new_cross_entropy = -np.log(self.V_x_xo(aux_initial_data,
                                         aux_best_variables_df, len(
                                             initial_data)))

aux = pd.DataFrame(columns = ['Cross Entropy', 'Variables'])
aux.loc[0] = [new_cross_entropy, aux_best_variables_df.columns]
cross_entropy_df = pd.concat([cross_entropy_df, aux])

aux.loc[0] = [clusters_initial_data_cross_entropy, initial_data.columns.astype(
                                                    str)]

cross_entropy_df = pd.concat([cross_entropy_df, aux])
aux.loc[0] = [ini_entropy, initial_data.columns.astype(str)]
cross_entropy_df = pd.concat([cross_entropy_df, aux])
aux.loc[0] = [clusters_entropy, initial_data.columns.astype(str)]
cross_entropy_df = pd.concat([cross_entropy_df, aux])

return cross_entropy_df

```

F.4 trans period graphics

F.4.1 hourly sliding window

```

def clusterevoplot(self, df, hours, clusters, p_pth):
    cluster_evo_df = pd.DataFrame(columns = ['Count', 'Cluster', 'Hour'])
    for hour in hours:
        for cluster in clusters:
            filtered = df.loc[(df['Date Time'].str.contains(hour)) & (df['Cluster']
                                                                    == cluster)]

            count = len(filtered)
            aux = pd.DataFrame(columns = ['Count', 'Cluster', 'Hour'])
            aux.loc[0] = [count, cluster, hour]
            cluster_evo_df = pd.concat([cluster_evo_df, aux])

    fig = plt.figure()
    cl_evo = pd.pivot_table(cluster_evo_df, index = 'Cluster', columns='Hour',
                            values='Count' )

    ax = sns.heatmap(cl_evo)
    fig.tight_layout()
    fig.savefig(p_pth + str('\\Cluster evolution heatmap.png'))
    plt.close(fig)

pass

```

Appendix G

Scenario 1 results for *IBERDROLA GENERACION ESPANA S.A*

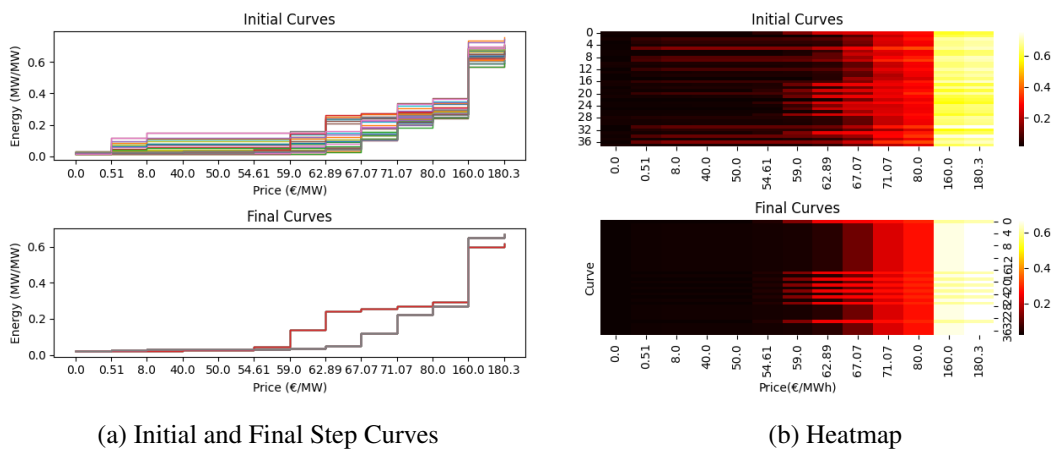


Figure G.1: Hour 00:00, *IBERDROLA GENERACION ESPANA S.A*

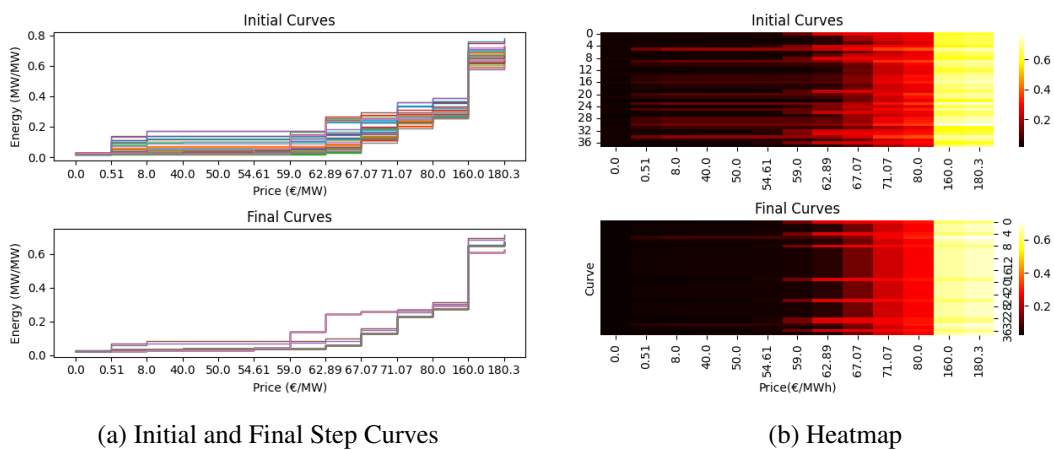
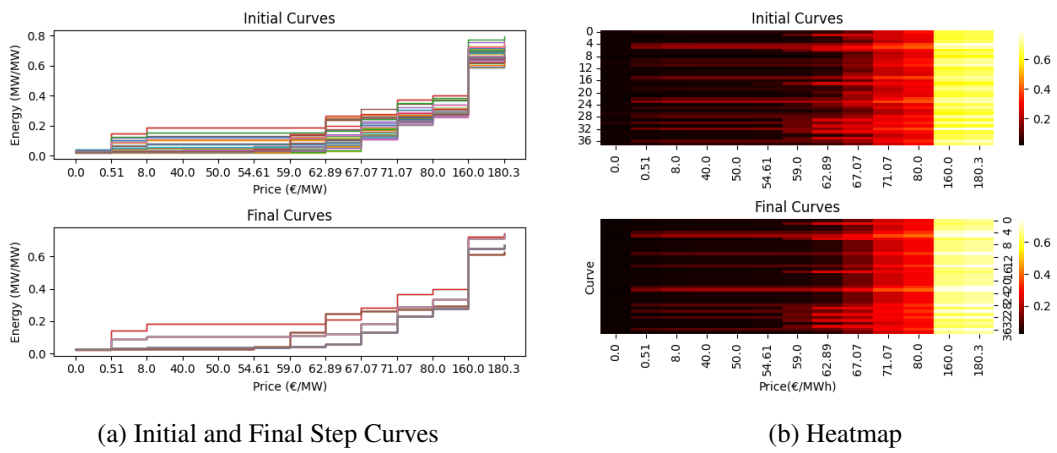
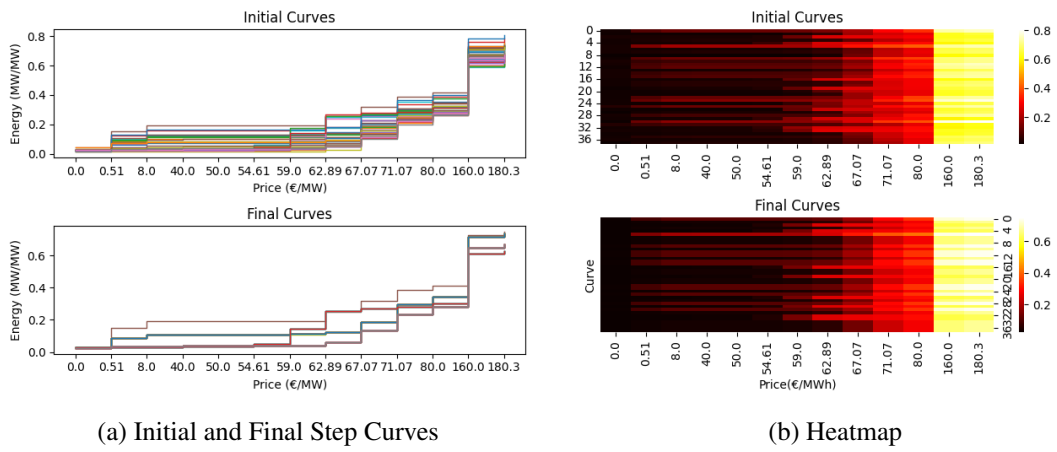
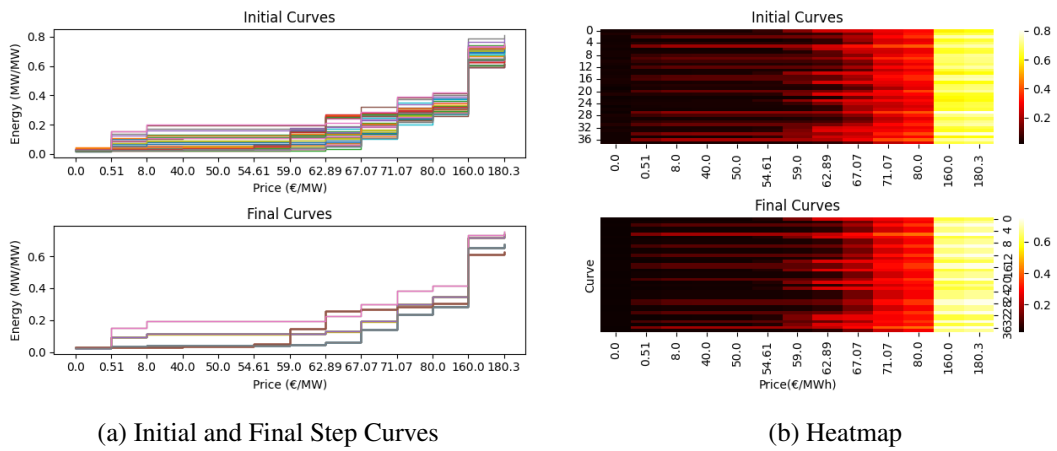
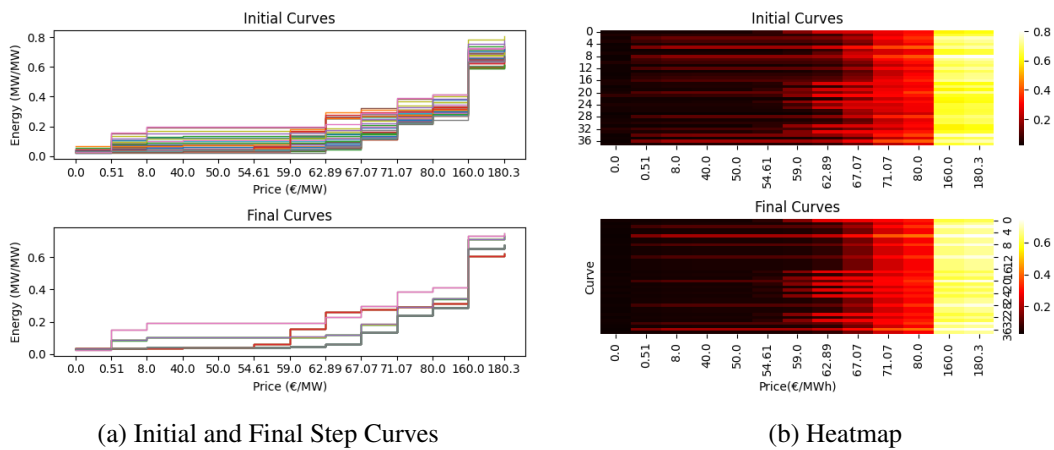
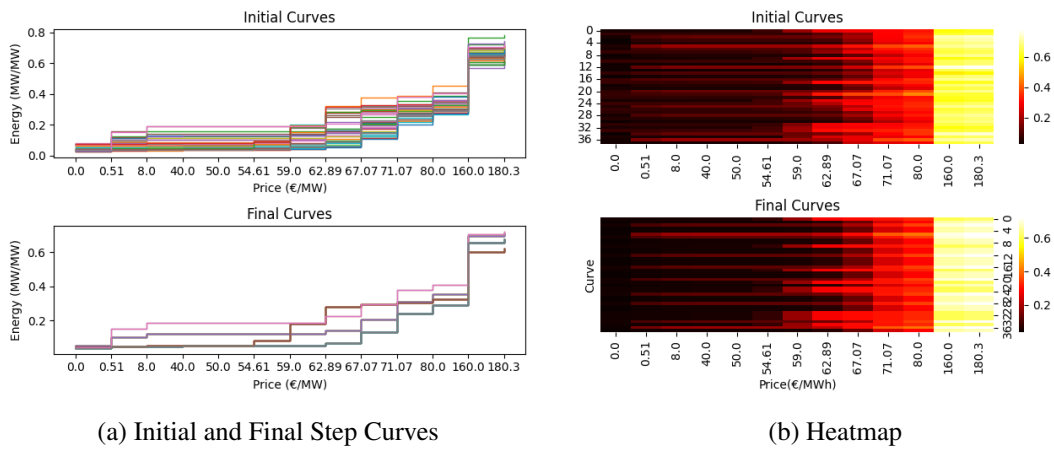
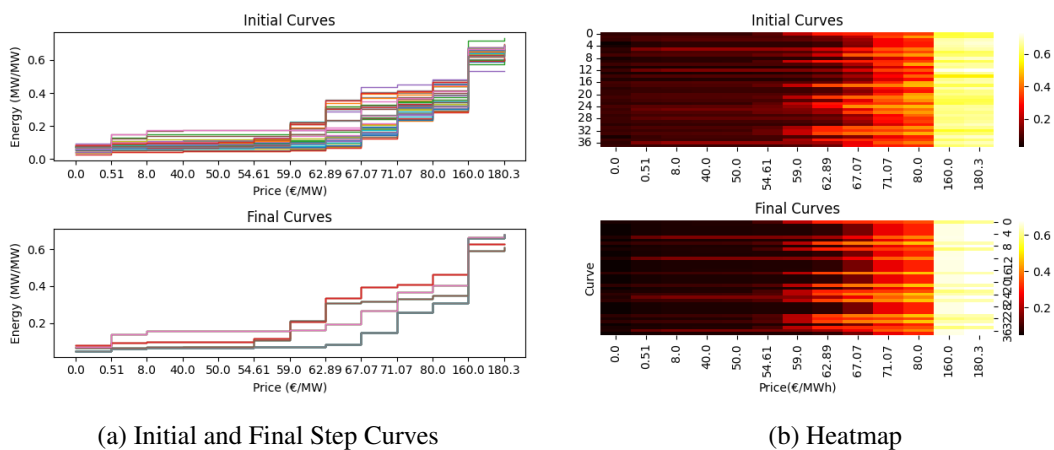


Figure G.2: Hour 00:01, *IBERDROLA GENERACION ESPANA S.A*

Figure G.3: Hour 00:02, *IBERDROLA GENERACION ESPANA S.A*Figure G.4: Hour 00:03, *IBERDROLA GENERACION ESPANA S.A*Figure G.5: Hour 00:04, *IBERDROLA GENERACION ESPANA S.A*

Figure G.6: Hour 00:05, *IBERDROLA GENERACION ESPANA S.A*Figure G.7: Hour 00:06, *IBERDROLA GENERACION ESPANA S.A*Figure G.8: Hour 00:07, *IBERDROLA GENERACION ESPANA S.A*

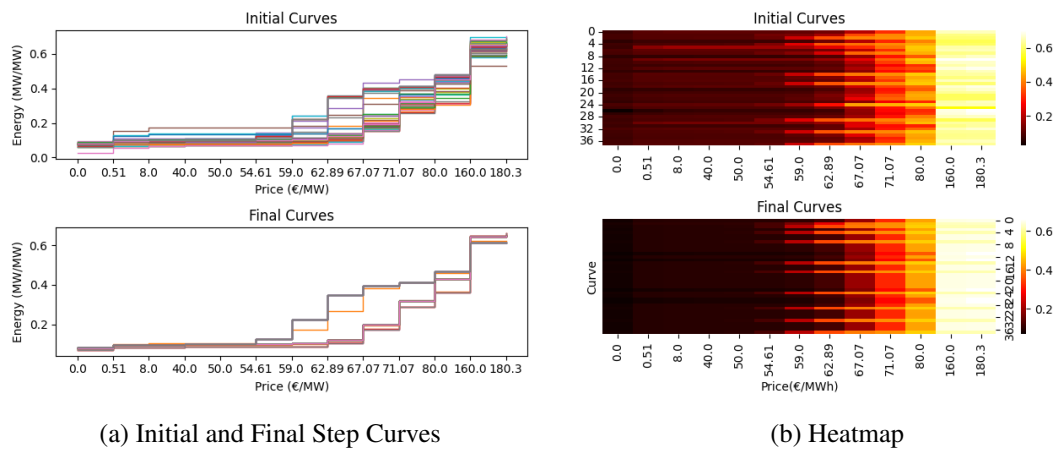


Figure G.9: Hour 00:08, IBERDROLA GENERACION ESPANA S.A

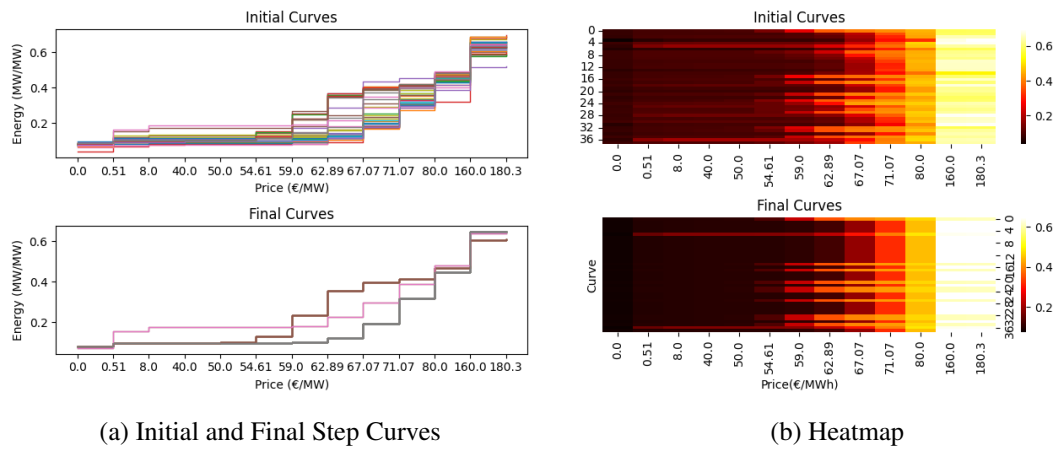


Figure G.10: Hour 09:00, IBERDROLA GENERACION ESPANA S.A

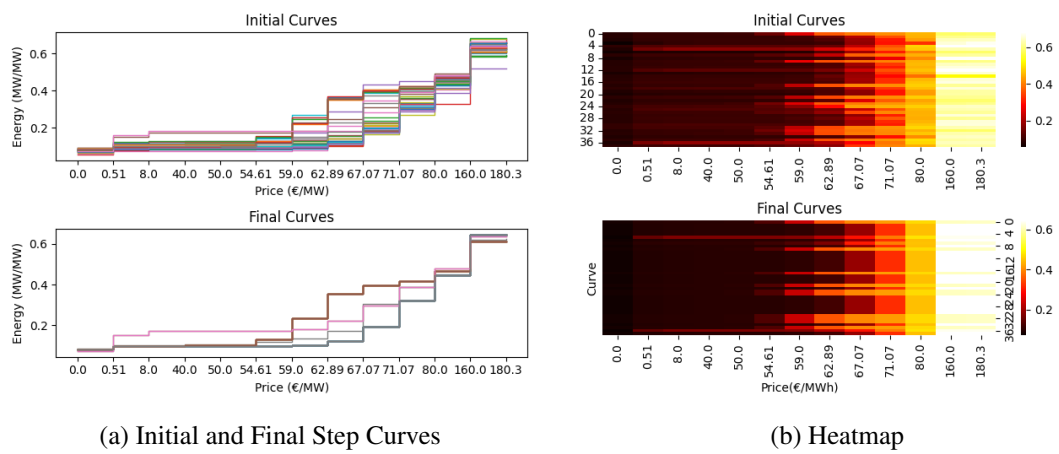


Figure G.11: Hour 10:00, IBERDROLA GENERACION ESPANA S.A

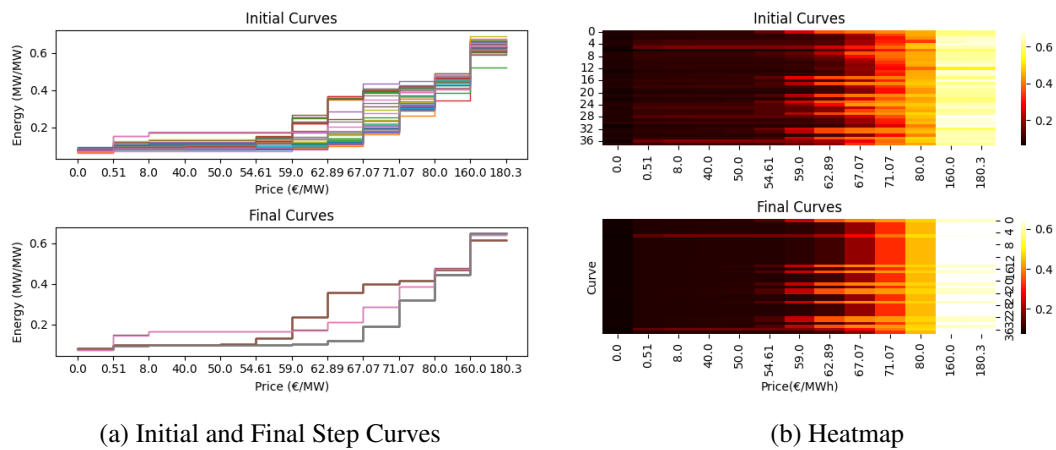


Figure G.12: Hour 11:00, IBERDROLA GENERACION ESPANA S.A

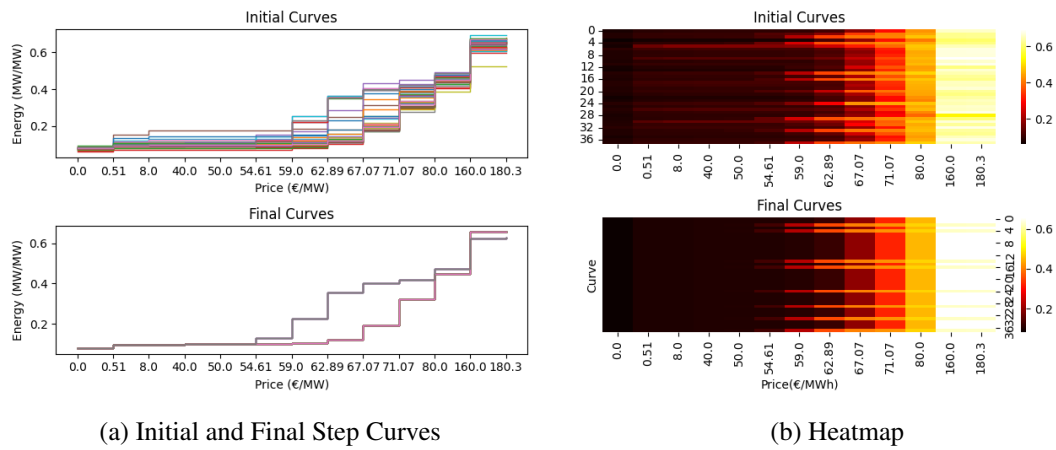


Figure G.13: Hour 12:00, IBERDROLA GENERACION ESPANA S.A

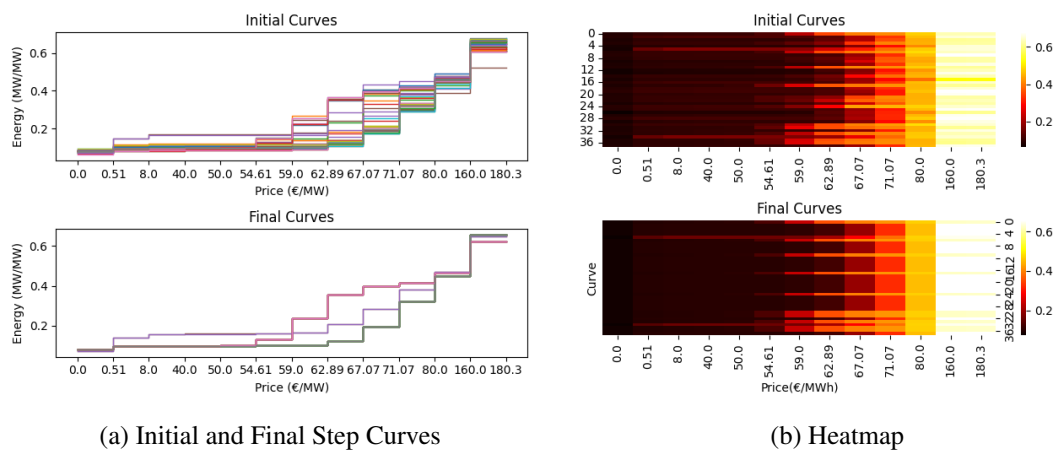


Figure G.14: Hour 13:00, IBERDROLA GENERACION ESPANA S.A

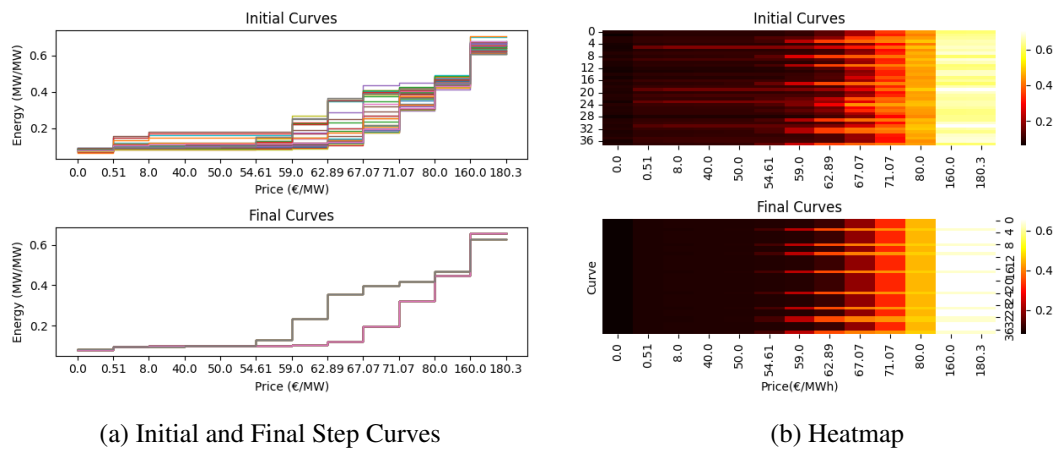


Figure G.15: Hour 14:00, IBERDROLA GENERACION ESPANA S.A

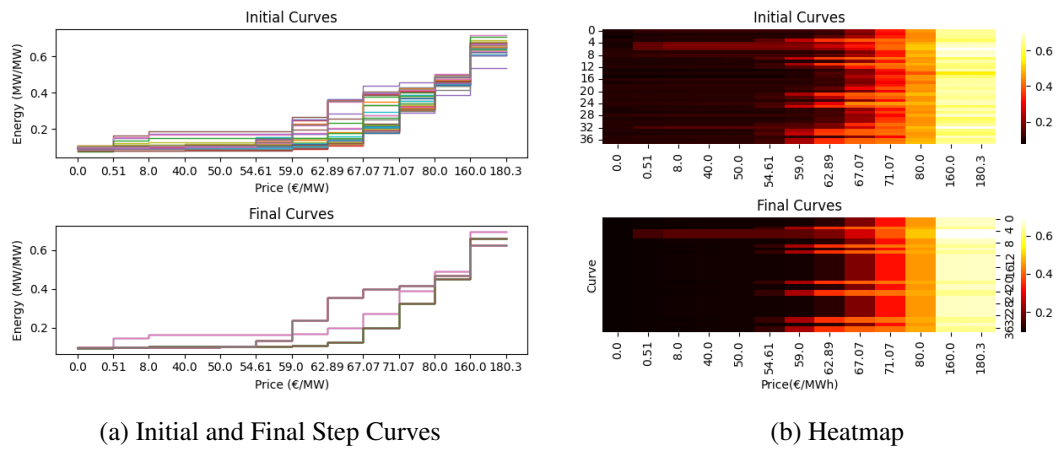


Figure G.16: Hour 15:00, IBERDROLA GENERACION ESPANA S.A

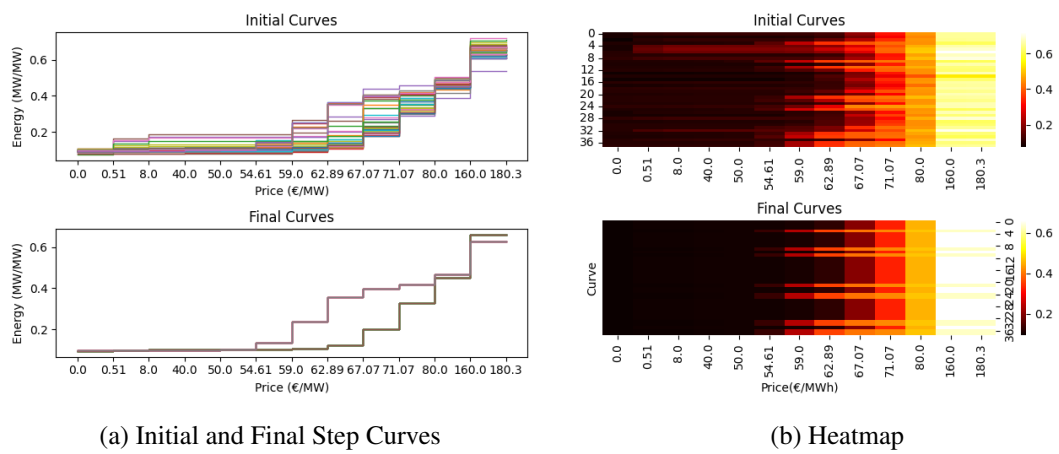


Figure G.17: Hour 16:00, IBERDROLA GENERACION ESPANA S.A

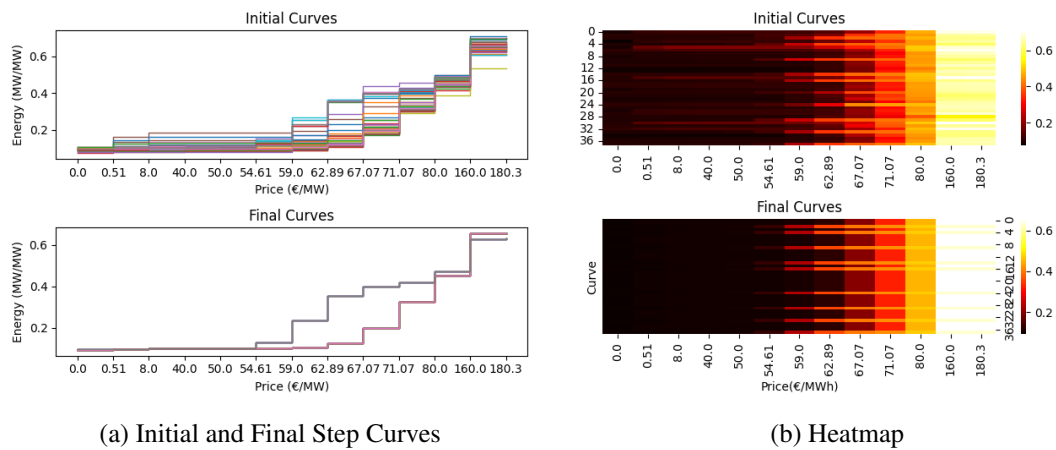


Figure G.18: Hour 17:00, IBERDROLA GENERACION ESPANA S.A

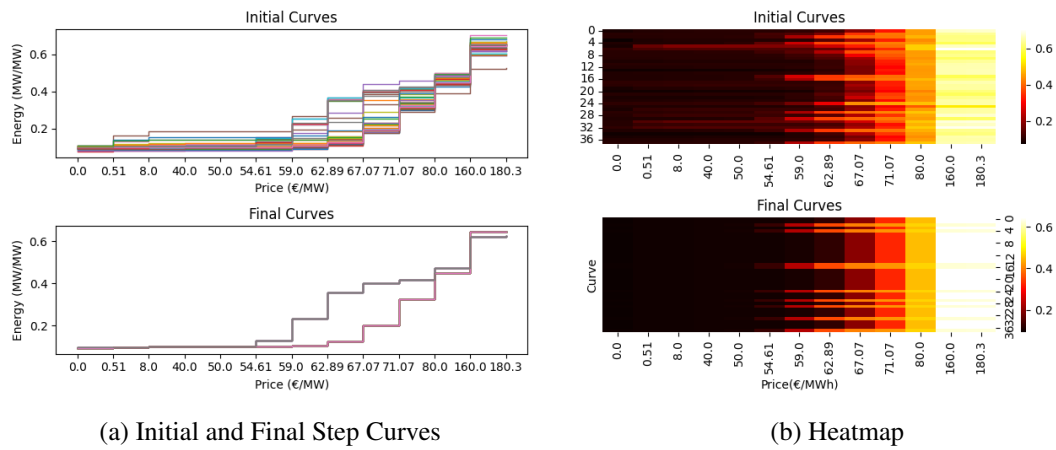


Figure G.19: Hour 18:00, IBERDROLA GENERACION ESPANA S.A

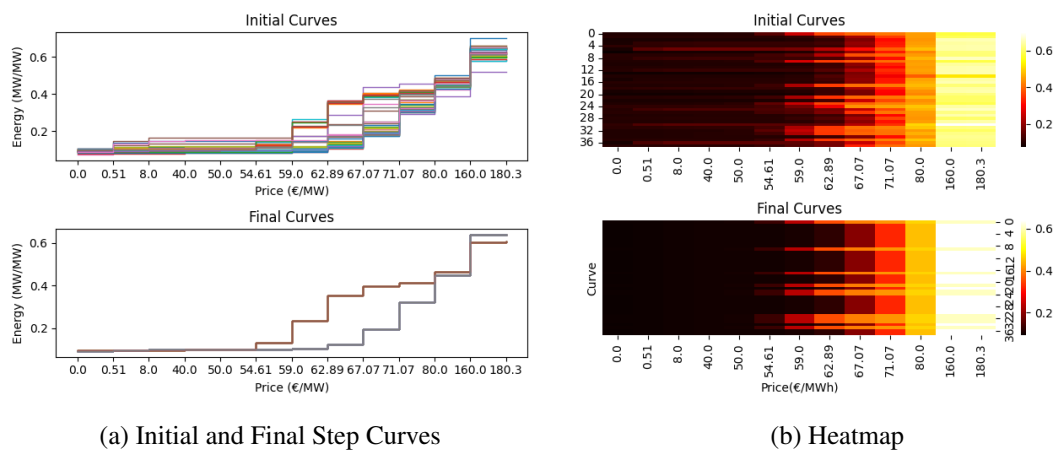


Figure G.20: Hour 19:00, IBERDROLA GENERACION ESPANA S.A

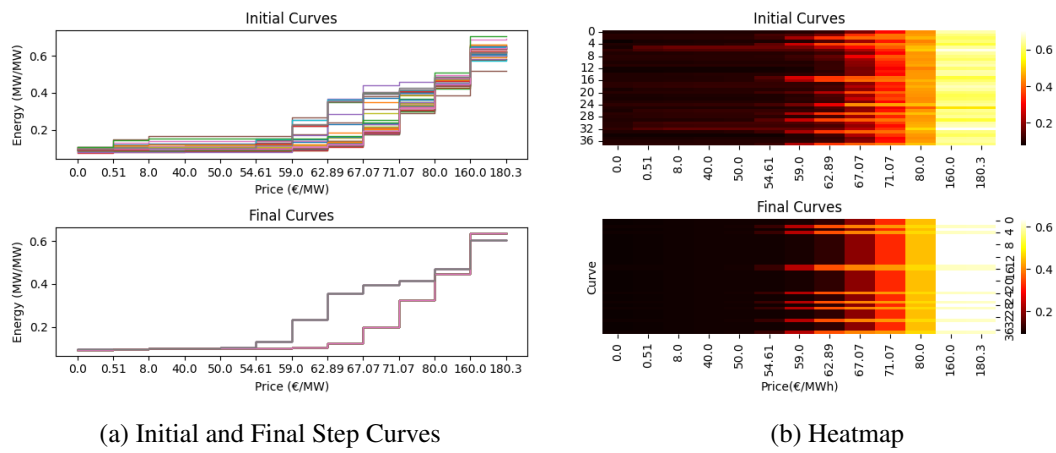


Figure G.21: Hour 20:00, IBERDROLA GENERACION ESPANA S.A

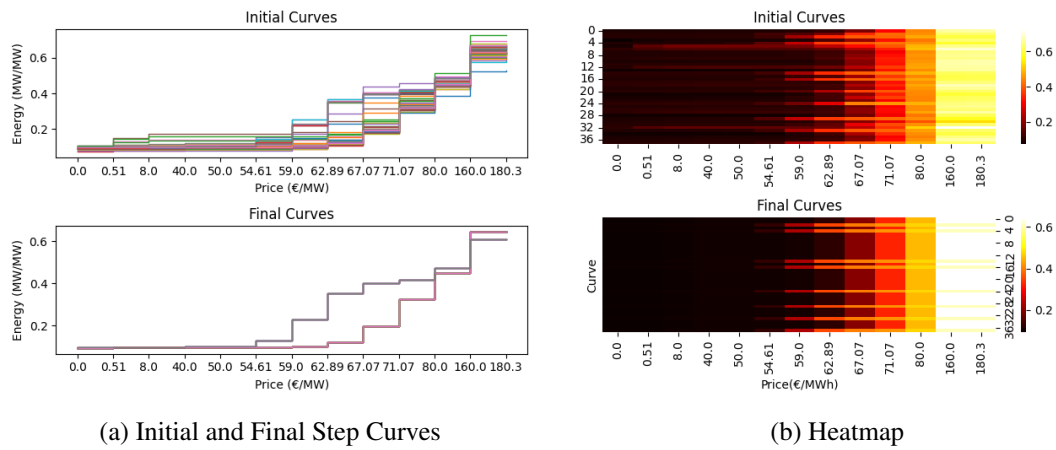


Figure G.22: Hour 21:00, IBERDROLA GENERACION ESPANA S.A

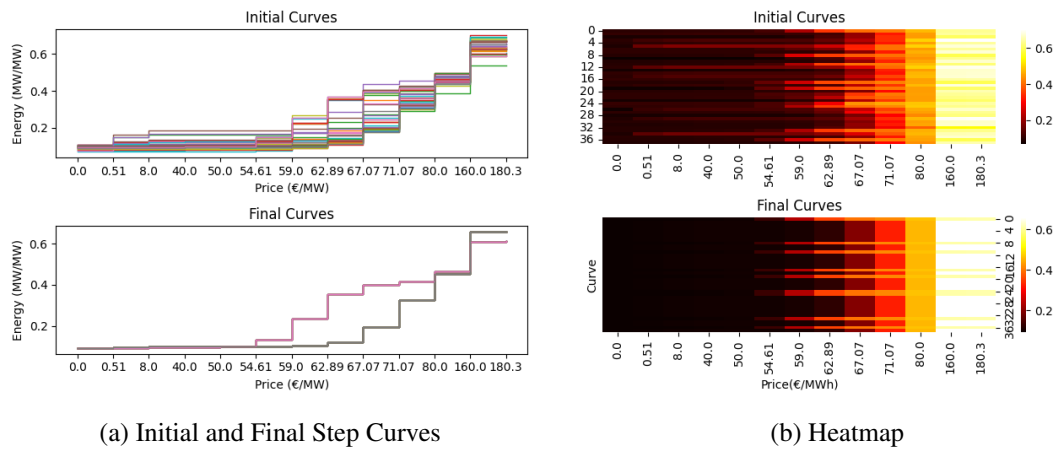


Figure G.23: Hour 22:00, IBERDROLA GENERACION ESPANA S.A

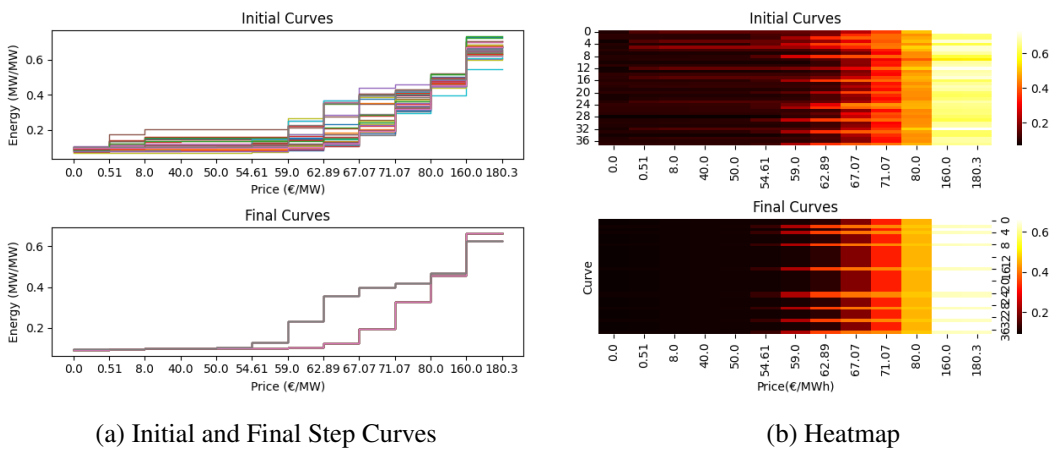


Figure G.24: Hour 23:00, IBERDROLA GENERACION ESPANA S.A

Appendix H

Scenario 1 results for *ENDESA GENERACION, S.A*

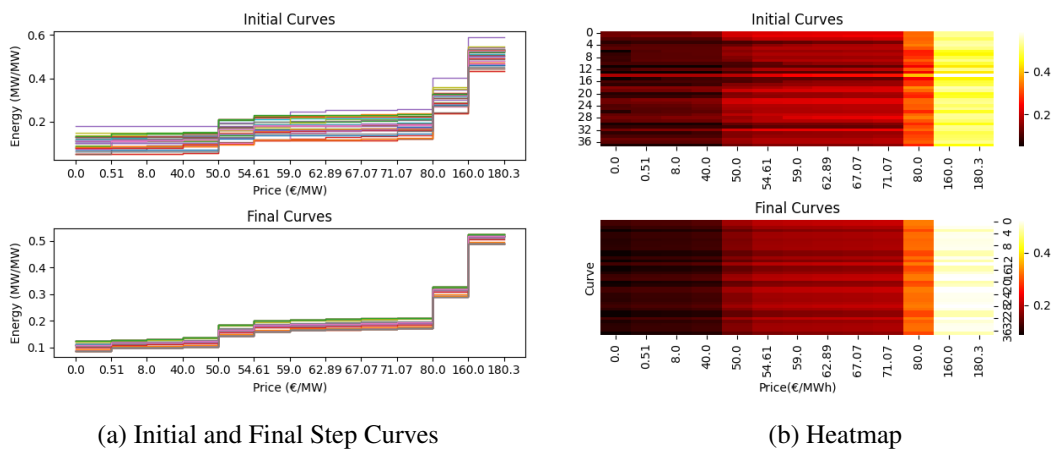


Figure H.1: Hour 00:00, *ENDESA GENERACION, S.A*

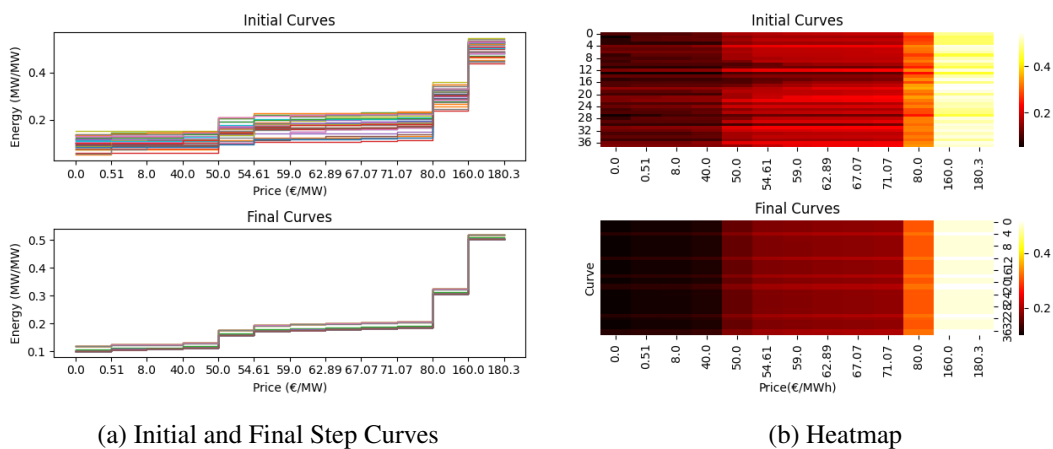


Figure H.2: Hour 00:01, *ENDESA GENERACION, S.A*

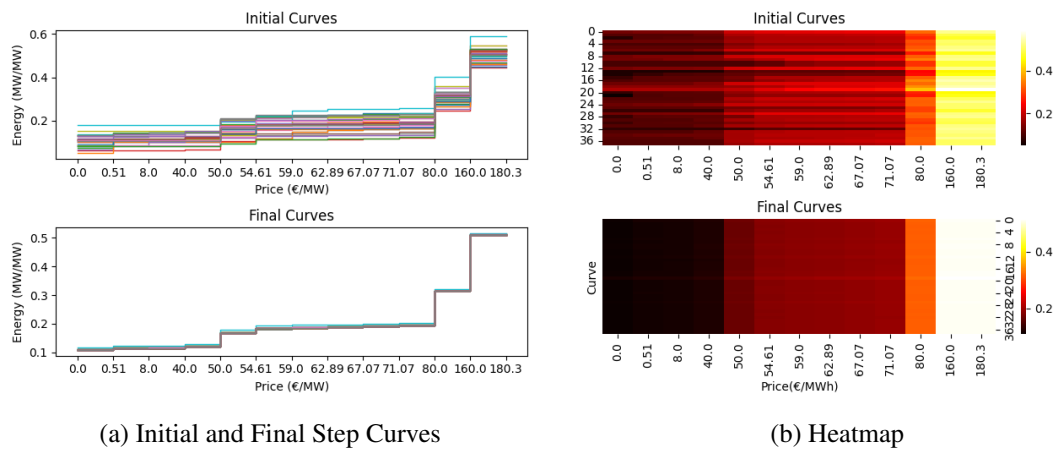


Figure H.3: Hour 00:02, ENDESA GENERACION, S.A

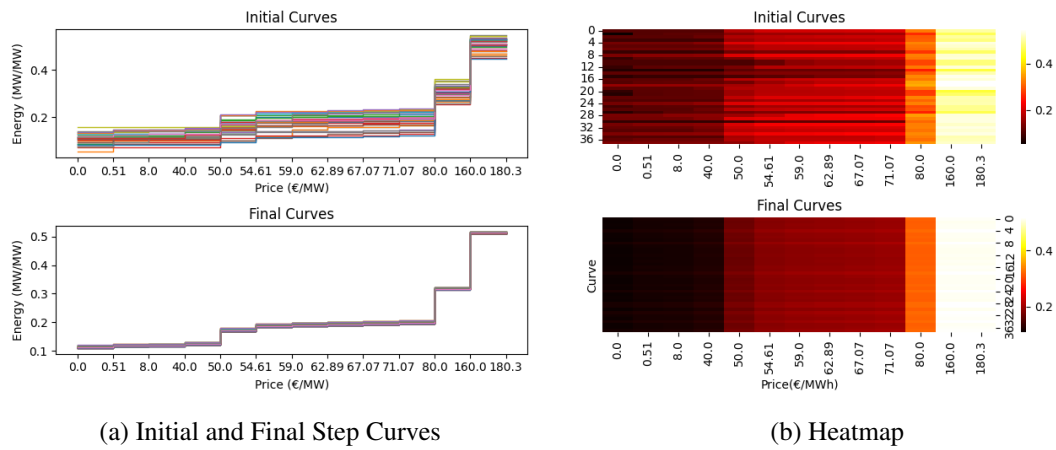


Figure H.4: Hour 00:03, ENDESA GENERACION, S.A

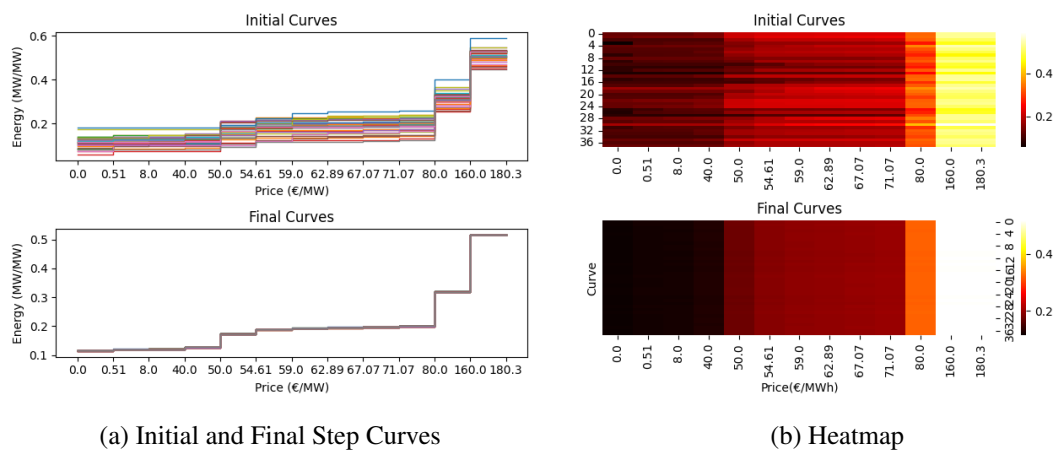
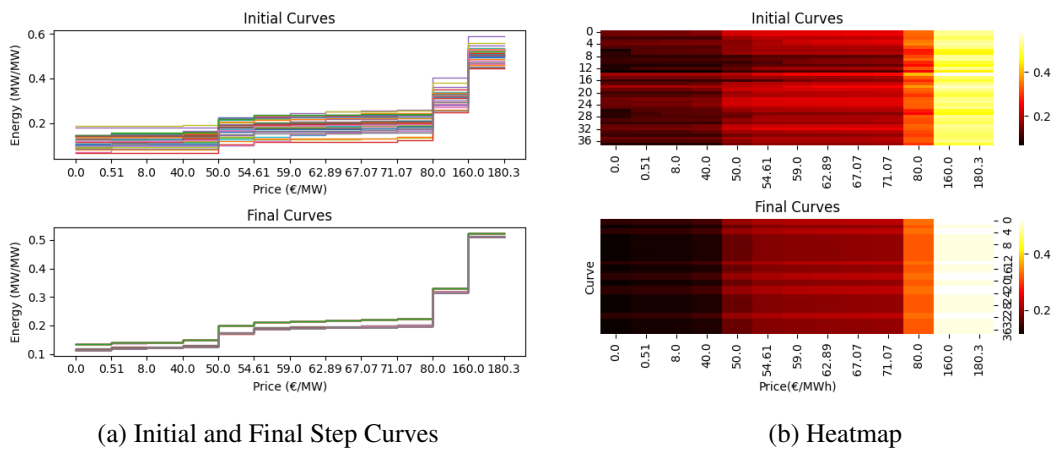
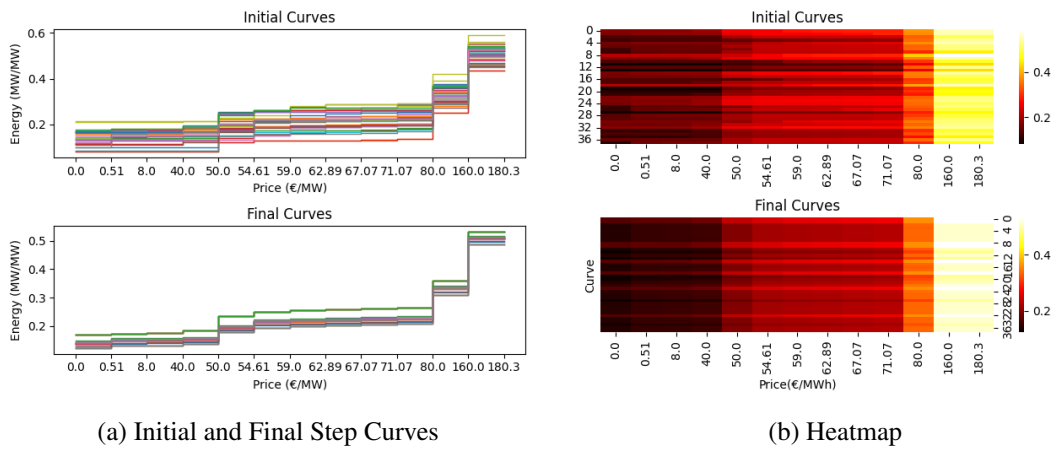
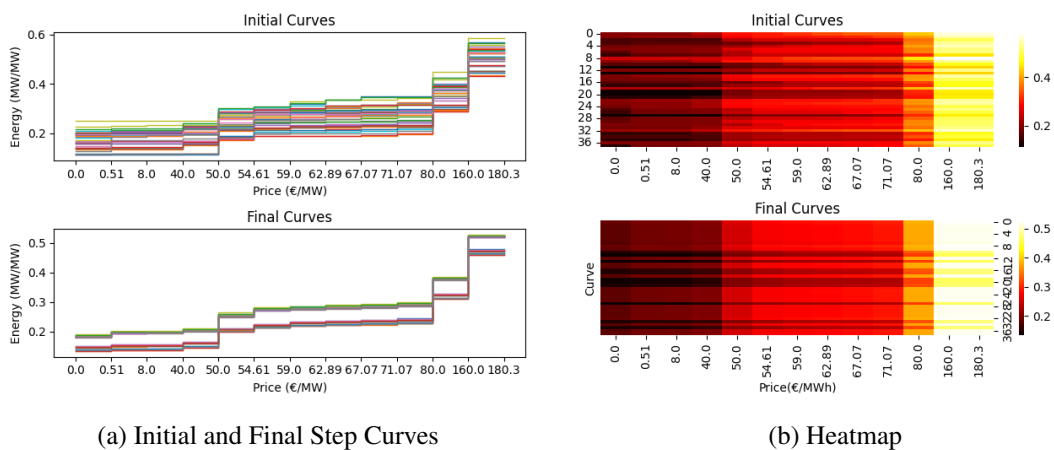


Figure H.5: Hour 00:04, ENDESA GENERACION, S.A

Figure H.6: Hour 00:05, *ENDESA GENERACION, S.A*Figure H.7: Hour 00:06, *ENDESA GENERACION, S.A*Figure H.8: Hour 00:07, *ENDESA GENERACION, S.A*

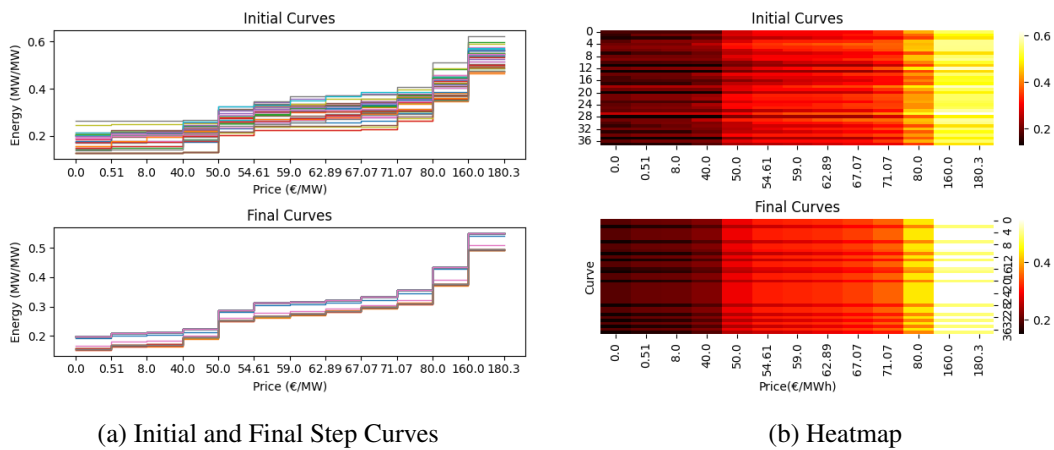


Figure H.9: Hour 00:08, ENDESA GENERACION, S.A

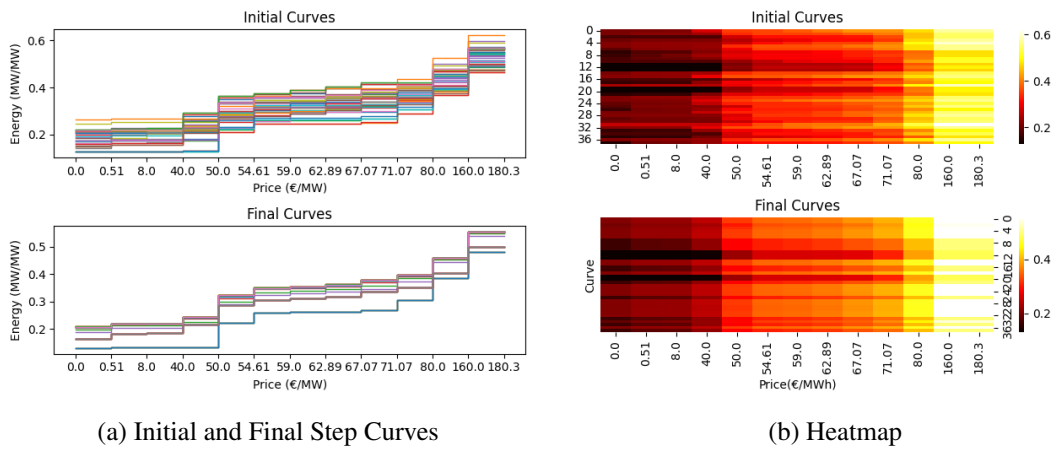


Figure H.10: Hour 09:00, ENDESA GENERACION, S.A

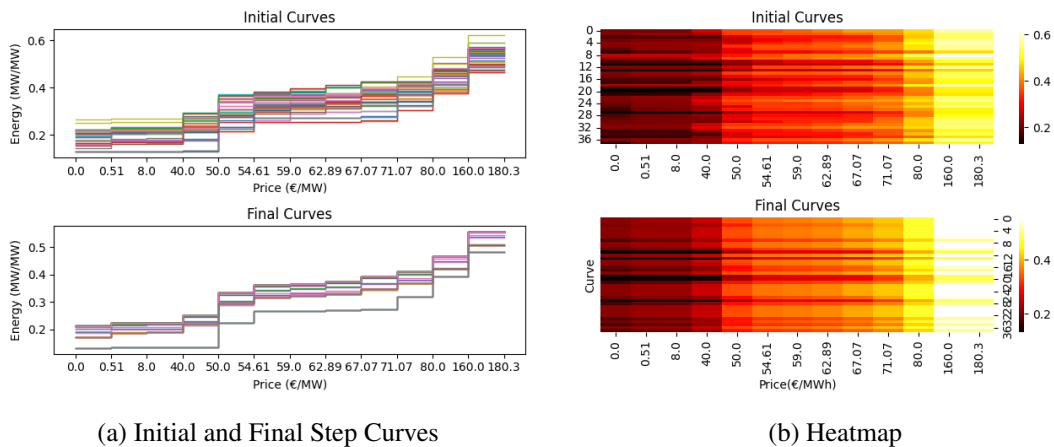


Figure H.11: Hour 10:00, ENDESA GENERACION, S.A

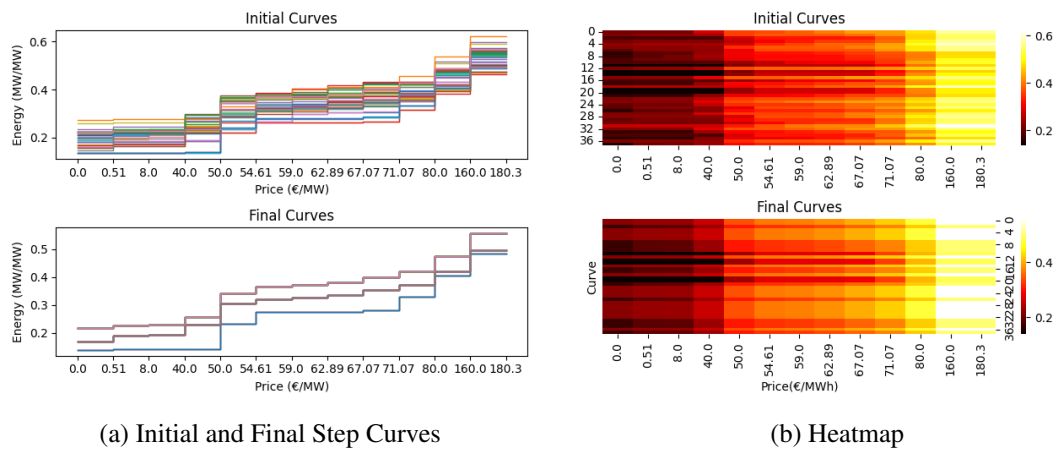


Figure H.12: Hour 11:00, ENDESA GENERACION, S.A

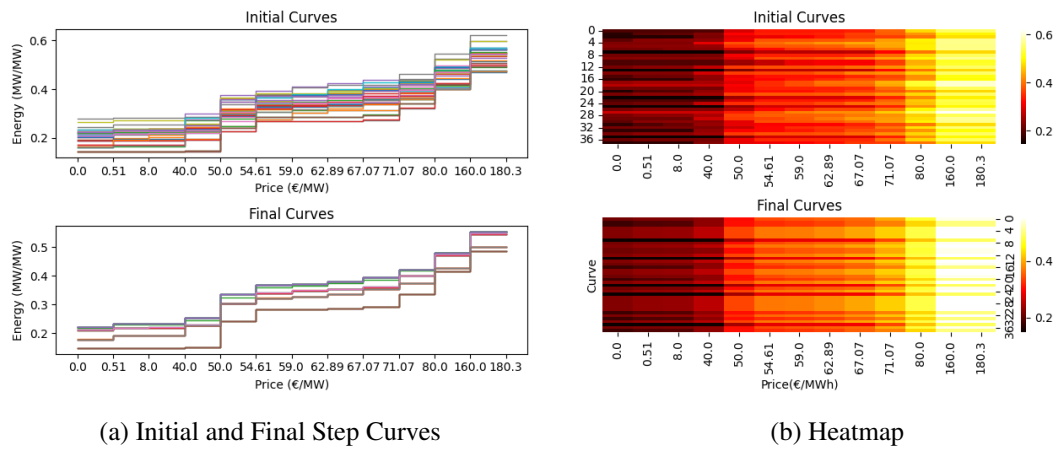


Figure H.13: Hour 12:00, ENDESA GENERACION, S.A

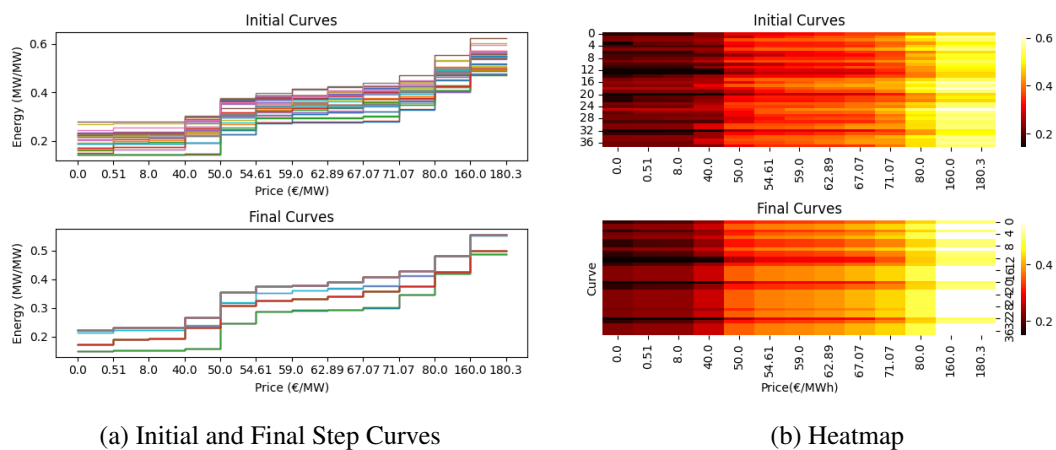


Figure H.14: Hour 13:00, ENDESA GENERACION, S.A

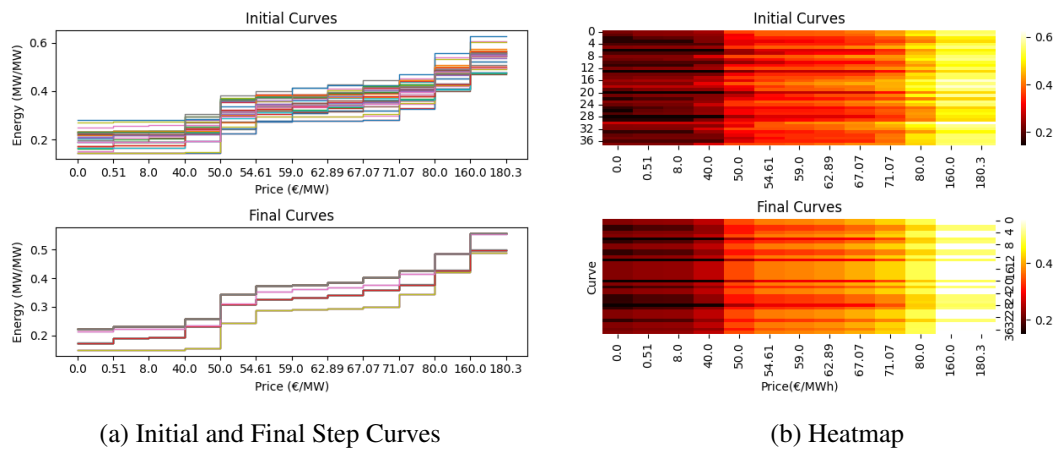


Figure H.15: Hour 14:00, ENDESA GENERACION, S.A

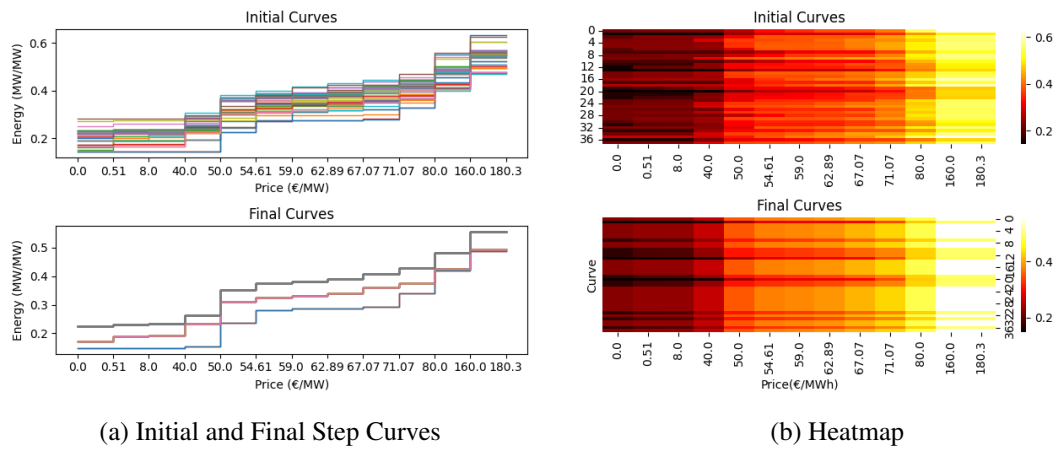


Figure H.16: Hour 15:00, ENDESA GENERACION, S.A

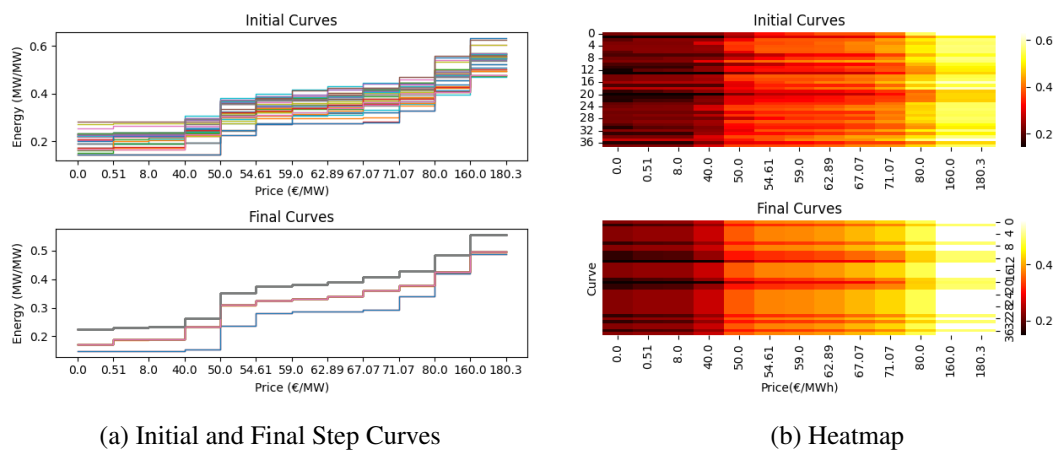


Figure H.17: Hour 16:00, ENDESA GENERACION, S.A

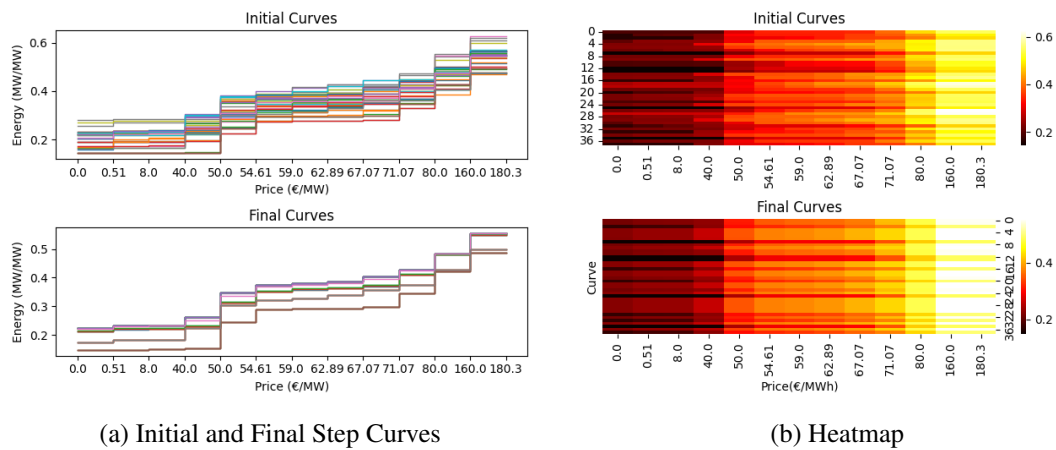


Figure H.18: Hour 17:00, ENDESA GENERACION, S.A

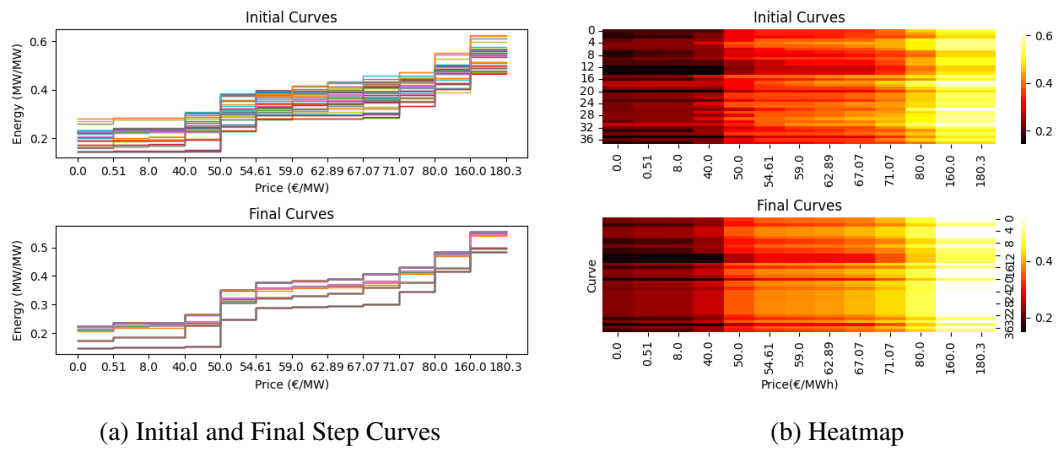


Figure H.19: Hour 18:00, ENDESA GENERACION, S.A

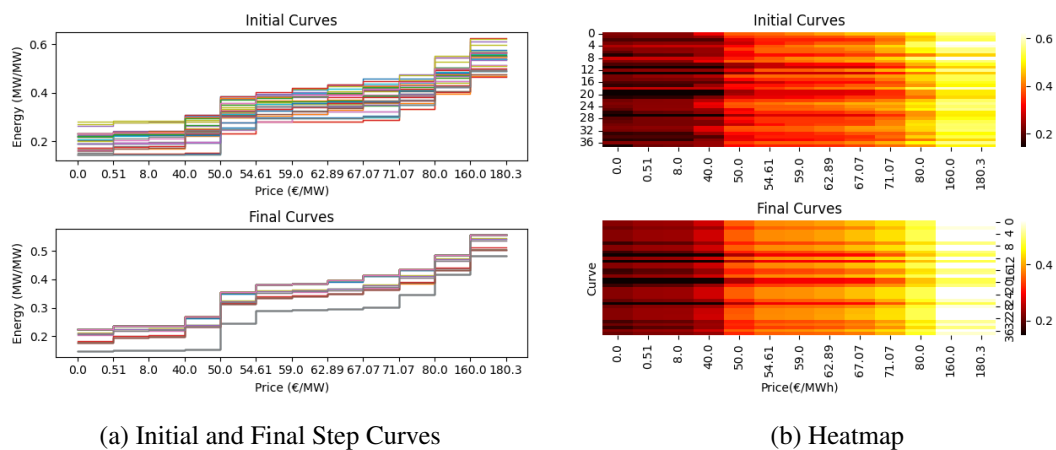


Figure H.20: Hour 19:00, ENDESA GENERACION, S.A

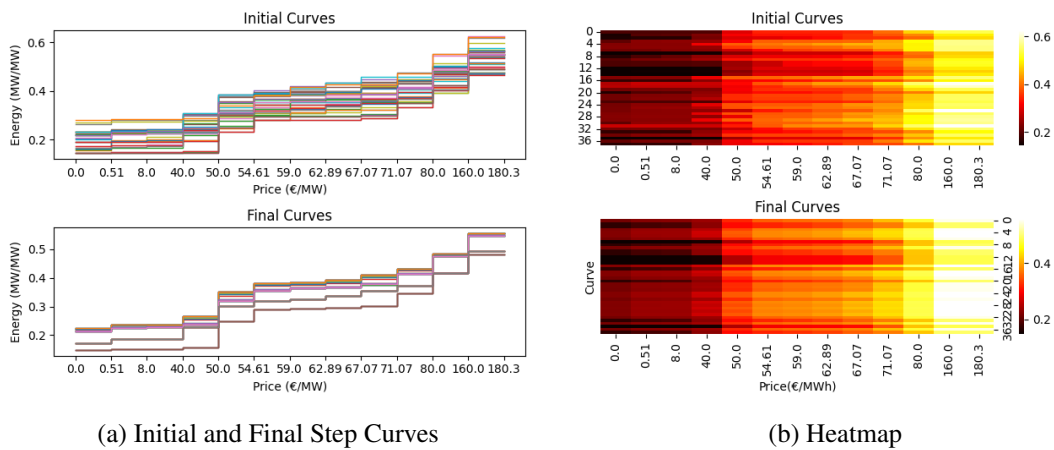


Figure H.21: Hour 20:00, ENDESA GENERACION, S.A

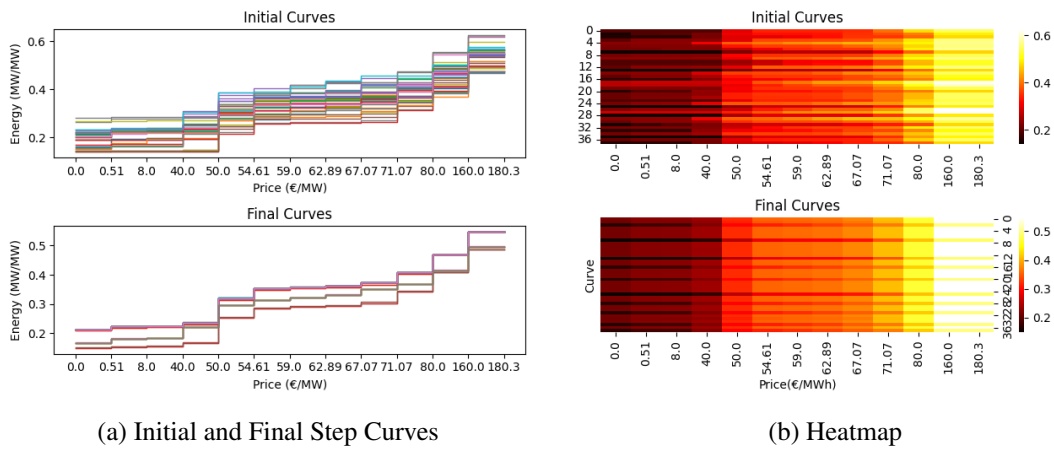


Figure H.22: Hour 21:00, ENDESA GENERACION, S.A

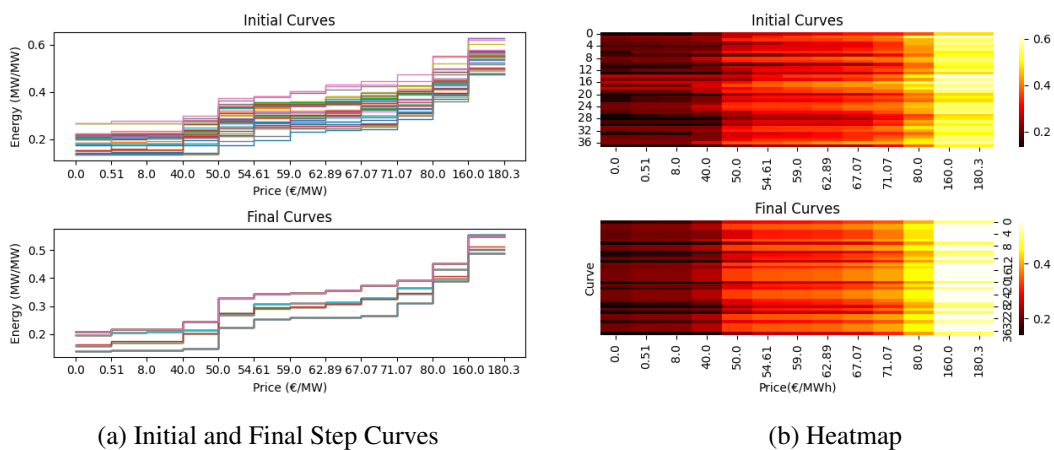


Figure H.23: Hour 22:00, ENDESA GENERACION, S.A

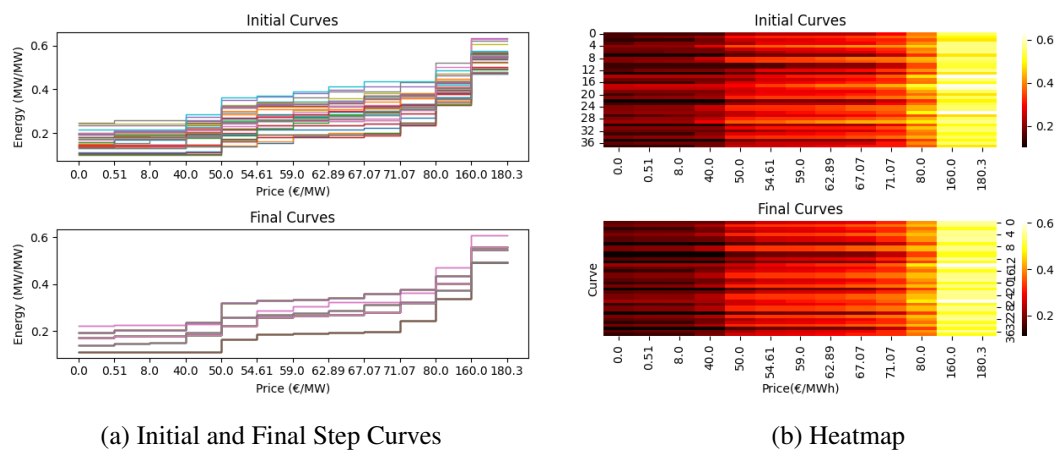


Figure H.24: Hour 23:00, ENDESA GENERACION, S.A