Prediction techniques based on non-parametric methods. Application to energy series

by

KIANA KATY PACO ANDRADE

A thesis submitted in conformity with the requirements for the MSc in Economics, Finance and Computer Science

University of Huelva & International University of Andalusia

September 2017

Prediction techniques based on non-parametric methods. Application to energy series

Kiana Katy Paco Andrade

Máster en Economía, Finanzas y Computación

José Manuel Bravo Caro

Universidad de Huelva y Universidad Internacional de Andalucía

Septiembre 2017

Abstract

The prediction of the electric energy demand is a problem of great importance for the electric industry, considering that, given the results of these predictions, different market agents take the most appropriate decisions. This is especially relevant for power companies that generate electric energy, because this way they're able to generate the amount needed in order to supply the market without exposing themselves to overproduction, which supposes a huge saving in economic costs.

This paper proposes a method to predict the electric power demand using techniques based on non-parametric regression. An empirical study focusing on the UK electricity market is presented.

The main objective of this research is to obtain a solid predictor for the electric power demand that essentially captures the intrinsic particularities of the electric power demand series and serves us to obtain future predictions. The methodology followed in order to obtain said predictor is based in using a training set in order to make and empirical adjustment of the predictor. The adjustment is obtained by selecting the set of hyperparameters that minimize the prediction error. The proposed predictor is validated and compared with others predictors through a validation set. The results support the goodness of the proposal made in this work.

JEL classification: C14, C51, C53, C63, L94, Q41

Key words: electricity markets, electric power demand, prediction models, non-parametric regression, functional data, smoothing methods.

Resumen

La predicción de la demanda de energía eléctrica es un problema de gran importancia para el sector eléctrico, considerando que, a partir de sus resultados, los agentes del mercado toman las decisiones más adecuadas. Esto es especialmente relevante para las empresas productoras de electricidad, ya que de esta manera son capaces de producir la cantidad necesaria para abastecer al mercado sin incurrir en sobreproducción, lo cual supone un enorme ahorro en costes económicos.

Este trabajo propone un método de predicción de la demanda de energía eléctrica utilizando técnicas basadas en regresión no paramétrica. Se presenta un estudio empírico centrado en el mercado eléctrico del Reino Unido.

El objetivo principal de esta investigación es el de obtener un predictor sólido de la demanda de energía eléctrica, que capte en esencia las particularidades intrínsecas de las series de demanda de energía eléctrica y nos sirva para obtener predicciones a futuro. La metodología seguida para conseguir dicho predictor se basa en utilizar un conjunto de datos de entrenamiento para realizar un ajuste empírico del predictor. El ajuste se consigue seleccionando el conjunto de hiperparámetros que minimizan el error de predicción. El predictor propuesto se valida y compara con otros predictores mediante un conjunto de datos de validación. Los resultados avalan la bondad de la propuesta realizada en este trabajo.

Clasificación JEL: C14, C51, C53, C63, L94, Q41

Palabras clave: mercados de electricidad, demanda de energía eléctrica, modelos de predicción, regression no paramétrica, datos funcionales, métodos de suavizado

Acknowledgements

The elaboration of this work is thanks to the knowledge acquired in the MSc, providing me each one of the subjects with the necessary notions to carry out this research.

Every step in the development of this work was guided by the professor Dr. José Manuel Bravo Caro who, with his talent, experience, patience and continuous dedication has given me guidance and suggestions, contributing with valuable comments for the properly elaboration of the work.

At the same time, I would like to thank the opportunity the University of Huelva and International University of Andalusia have given me to broaden my educational and professional horizons, which opened the doors for me to enter the fascinating world of data analysis.

To my family, for giving me unconditional support at every moment, and those who guide every one of my steps from heaven.

Agradecimientos

La elaboración de este trabajo es gracias a los conocimientos adquiridos en el Máster, aportándome cada una de las asignaturas las nociones necesarias para llevar a cabo esta investigación.

Cada paso del desarrollo fue guiado por el profesor Dr. José Manuel Bravo Caro, quien con su talento, experiencia, paciencia y dedicación continua me ha brindado orientación y sugerencias, aportando valiosas observaciones en la elaboración del presente trabajo.

A la vez agradecer la oportunidad de ampliar mis horizontes educacionales y profesionales a la Universidad Internacional de Andalucía y Universidad de Huelva, que me abrieron las puertas para adentrarme al fascinante mundo del análisis de datos.

A mi familia, por brindarme su apoyo incondicional en cada momento, y aquellos que guían cada uno de mis pasos desde el cielo.

Contents

List of Tables

List of Figures

Introduction

Due to the nature of the electric power, it is not possible for it to be stored. Because of this, power companies that generate electric energy must anticipate to future demands in order to avoid inadequate power production, so it seems clear that a good prediction is an important issue.

Since electric power demand is a key point for strategic planning and decision making in these kind of companies, a bad estimation of the electric power needed in order to satisfy the market would result in high economic costs. That's why these kind of companies should have proper techniques to provide reliable demand predictions.

Given this scenario, one approach to foresee the necessary electric power production to adequately satisfy the demand requiremente, is to obtain short term predictions. That is, given historical data for the daily electric power demand, predict one day ahead demand. By using this approach companies could benefit of a good way of estimating future demands in the short term, which would ultimately lead to the maximization of their profits.

Many different techniques were developed to predict electric power demand in the short term. Two different statistical methods were used fo this matter. The first of them consists on a model to derive the load profile, see [1,2]. The second one proposes 24 different models to obtain one day ahead hourly predictions, using a different time series for each hour [3,4]. In some publications temperature is incorporated as a exogenous variable in the model [5].

In this paper, different short term prediction models are proposed for the estimation of one day ahead hourly electric power demand. To do so, we will make use of different models such as the so called Naïve in its standard form as well as in its seasonal form and non-parametric regression models. Moreover, in this analysis we can observe the effect of considering different date formats through a combination of themselves, including hyperparameters for which the optimal values are obtained. In this way we aim to obtain a solid predictor with the least possible error.

In order to achieve the objective of this research and with the aim of presenting the reader a wide perspective of the prediction techniques used in this paper, it was considered necessary to divide this document in four main chapters. The first chapter is devoted to describe the contextual framework of the research, while the second chapter is focused on giving a theoretical framework of the basic concepts in the topic of prediction techniques. The third chapter consists on a practical application of prediction models development. The fourth and last chapter is devoted to the final conclusions.

The first chapter, contextual framework, pretends to give a general outlook of the context of this research. In the first part of the contextual framework a general outlook of the electric power market structure in the United Kingdom is given, from the market share of the main companies to the actual regulation of the market. In the second part of this contextual framework, a view of the situation of the electric power demand in the United Kingdom is given,

In the second chapter, a detailed view of the conceptual part needed to the development of prediction models based on non-parametric regression technique is given. In the first part of this chapter, we highlight the different prediction techniques, emphasizing the concept of regression. Next, a general outlook of the so-called Naïve models is given, both in its standard form and its seasonal form. Then we offer a detailed explanation of non-parametric regression, that are the essence of this research.

The third chapter consits in the implementation of the theoretical concepts shown in the second chapter. Firstly, we describe the database used and we detail the cleaning and transformation process of the data in order to adapt them to a suitable format that allows us to properly work. Later, a detailed development of each one of the predictors used in this paper is offered, providing the results of the predictions for these predictors. Finally, once we obtain the best predictor for the training set, we test this predictor in a test set that comprises the first six months of the year 2017.

The fourth chapter consits in a final conclusion.

Chapter 1. Contextual Framework

1.1 Electric power market structure in the United Kingdom

The United Kingdom has a fully liberalized and privatized electricity market. The UK was at the forefront of the liberalization of its electricity sector from the mid-1980s when the 1983 Energy Act opened the supply market beyond the 12 area councils that existed at the time. In the privatization programs that followed in the 1980s and 1990s, England and Wales were also pioneers in the creation of a wholesale market where electric power generating companies could sell electricity in near real time to meet demand on the supply side (the Wales Pool system between 1990 and 2001). Privatization programs were carried out throughout the jurisdictions of England and Wales. The restructuring and privatization program began in the early 1990s and saw the opening of the retail market introduced in phases in the period up to 1999.

In Scotland, vertically integrated energy consortia were privatized in 1991 and nuclear interests were privatized in 1996. In Northern Ireland, the electricity industry was privatized between 1992 and 1993. After privatization, the electricity market in The United Kingdom has changed for a number of reasons, especially to keep pace with changes arising from EU legislation and in particular the principles of free competition, transparency and free access to the network.

According to data from the Department of Business, Energy and Industrial Strategy, the energy sector in the United Kingdom is of vital importance in the country's economy, as it produces, transforms and supplies energy in its various forms to all sectors. Specifically for 2016, the energy sector reached

2.3% of the Gross Domestic Product of the country, concentrating 10% of total investment for the country and 34% of industrial investment.

Specifically, in relation to the share of the energy sector corresponding to electricity, it accounted for 17.5% of total energy consumption IN 2016. The sources of electricity generation were: gas with 42% , renewable energy with 24.5%, nuclear energy with 21%, coal with 9% and other energy sources with 3.1%.

Electricity is a product that can not be stored on a large scale, so demand and supply must be satisfied at all times. In the UK this is mainly done by suppliers, generators, merchants and customers operating in the competitive wholesale electricity market.Trade can take place bilaterally or in exchanges, and contracts for electricity can be given over time scales ranging from several years to intraday trading markets. Electricity can also be imported or exported through interconnections. There are currently electrical interconnections between the UK, France, the Netherlands and Ireland.

National Grid Electricity Transmission (NGET) has overall responsibility as the "residual balancer" of the electricity system, and its goal is to ensure that supply and demand for electricity coincide at every second. NGET has a number of tools to do this, including a balancing mechanism.The Equilibrium Mechanism allows NGET to accept electricity offers (increases in generation and reductions in demand) and electricity tenders (generation reductions and demand increases) at very short notice.If a market participant generates or consumes more or less electricity than it has hired, it is exposed to the price of imbalance, or "withdrawal", by the difference. The withdrawal price is the incentive for market participants to ensure that consumer demand for electricity is met and the NGET's "residual balance effect" is minimized. The starting price is based on the NGET costs of balancing the system.

On the other hand, Ofgem the Office of Gas and Electricity Markets, which depends on the Gas and Electricity Markets Authority (GEMA), is responsible for regulating the gas and electricity markets in Great Britain. They provide information on the evolution of demand and supply in retail markets.

With reference to the share of the electricity market in the UK, the main supply companies for the year 2016 are, in order of importance and as a percentage of the total, the following: British Gas 23%, SSE 15%, E.ON 15%, EDF 12%, Scottish Power 11%, npower 10%, which together represent 86% of the total electricity supplied [6]

1.2 Electric power demand in the United Kingdom

As for the situation of the electric power demand demand in the United Kingdom, the total amount of demand for the year 2016 was 357 TWh. Of the total demand, 30% corresponded to domestic consumption, this being the most significant sector of demand agents, followed closely by the industrial sector, which accounted for 26% of the total demand for electricity. The third sector with the highest demand for electric power was commercial, with a 21% of the total.

For this research, an analysis of the demand for electric power will be carried out, for which we consider the 2016 data series and the first 6 months of 2017 for the validation of the prediction model. The data for this purpose were obtained from the website of National Grid, a British multinational utility company based in the United Kingdom and the northeastern United States [7].

The 2016 part of the series of data to be worked covers the 12 months of 2016 from January 1 to December 31 (366 days), with hourly electrical demand data for each day. There will then be 24 observations of the electric power demand for each day, so the predictions will be hourly for each day of the year. This complete series is illustrated in Figure 1.1.

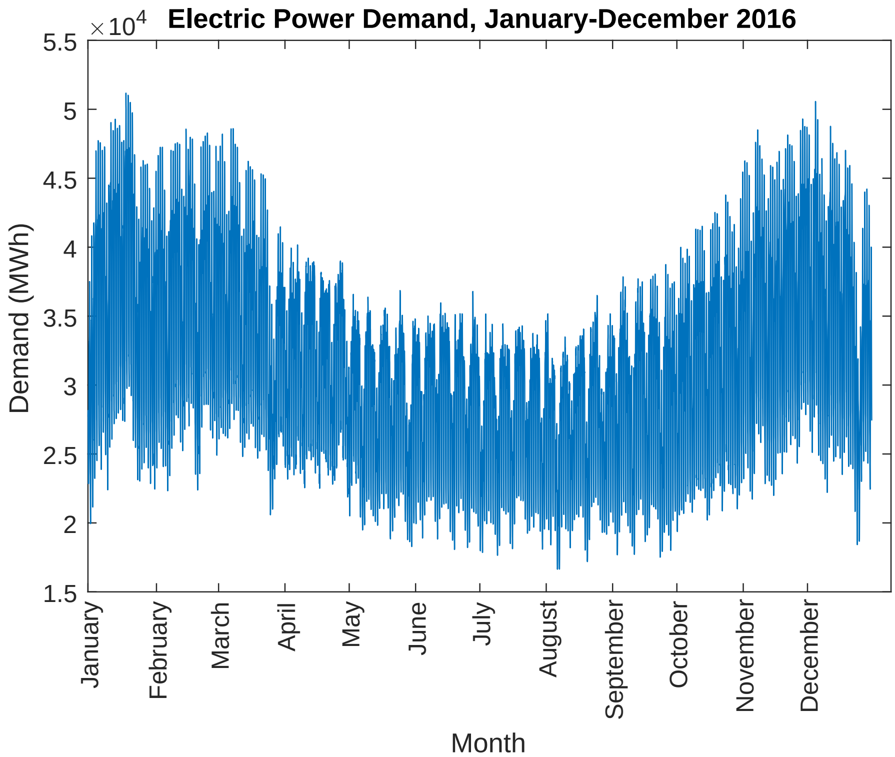


Figure 1.1: Electric Power Demand 2016

In Figure 1.1 we can observe, as several researches indicate over the years, that there are some common characteristics in the series of demand such as trend, seasonality and others. It can also be observed that the demand for electricity is high both for the first weeks of the year and for the last weeks of December, then it can be evidenced that there is a greater consumption of electricity in winter in relation to other seasons of the year, evidencing the existence of a certain annual seasonality of our series of electric power demand data.

To perform a more detailed analysis of our data series and to see the weekly seasonality pattern we will represent the weekly data considering the month of May 2016, as shown in Figure 1.2.

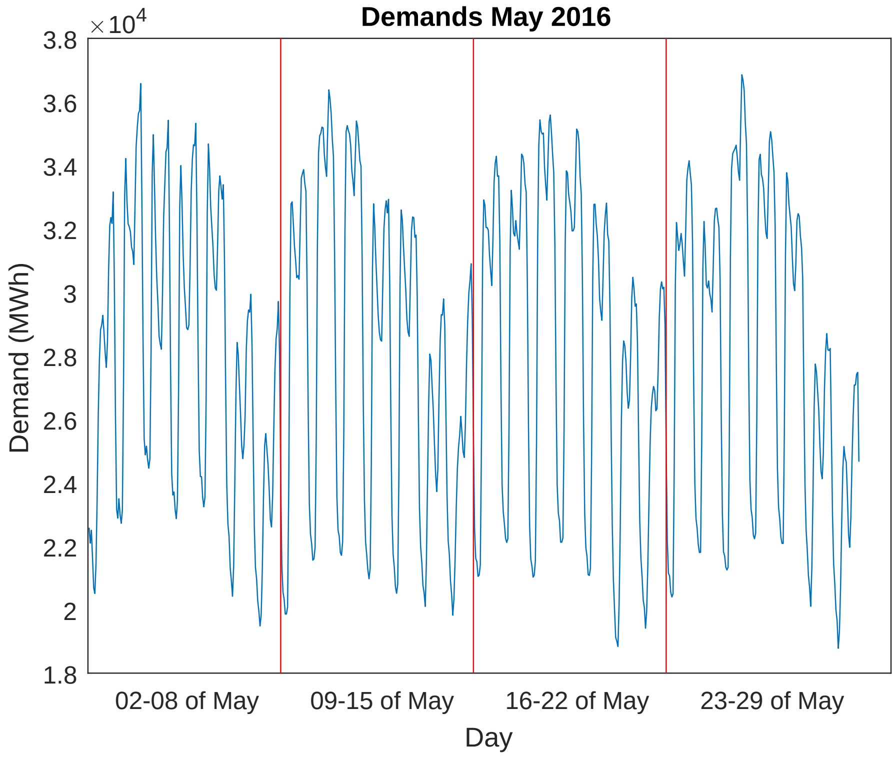


Figure 1.2: Electric Power Demand, May 2016

In Figure 1.2 we can observe the weekly seasonality of our time series for electric power demand, following a similar pattern each week. Thus we can also observe that the days that present higher demand correspond to the central days of the week, that is, Tuesday, Wednesday and Thursday; while the days with the lowest demands are on the weekends. If we consider the behavior of each day of the week and as we have seen these have a similar pattern between each week, this shows that if we take the characteristics of a particular day to predict the same, but for the following week, it would be a good predictor because of this similar pattern between weeks, as well as the influence of the previous day to predict the next day of a week. Therefore, the weekly seasonal effect must be taken into account when developing a predictor.

Chapter 2. Prediction Methods: Theorical Concepts

In recent times, the massive and continuous generation of data of all types coming from very different sources has allowed us to glimpse the potential of obtaining valuable information through the use of increasingly sophisticated data analysis techniques. For this reason, one of the fields that has recently experienced a huge growth is the field of Machine Learning.

Machine Learning is a branch within the field of Artificial Intelligence. The term has been present since the 50s, but has been in recent years when it has taken on great importance thanks to the enormous increase in computing capacity and the large volume of data that companies start to manage. In this sense, what it offers is a set of algorithms whose goal is to equip computers with the ability to learn without the need to be explicitly programmed to do so.

These algorithms can be classified into two main groups:

- **Supervised Learning**: where we have previous knowledge that can help us understand the new data that would come. That is, it allows us to make predictions of future data from past and present data, considering risk and uncertainty as key aspects of the analysis. Supervised learning methods can be classified in turn into classification methods and regression methods. The difference between the classification and regression methods is that for the case of classification the output data is discrete, whereas in the case of regression the output data is continuous.

- **Unsupervised Learning**: where we do not have previous experience to analyze the new data. it is more oriented to pattern searching and grouping of data according to similar characteristics and patterns of behavior. The most commonly used non-supervised learning methods are: Principal Components Analysis and Clustering [8].

In the case of predictive analysis, in order to generate predictions from the available data, we will use supervised learning techniques in this research, more specifically regression methods. In addition we considered a functional data analysis approach where our data will be functions of daily demand curve, as we will develop later. Therefore the data to be used will be daily demand curve functions, so the most appropriate method to carry out our energy series prediction application will be the regression method.

Being the purpose of the regression models to construct mathematical models that allow to explain the relation of dependence that might exist between a response variable and one or more independent variables, we can use these models as a tool to predict new values of the response variable from a certain particular value that the explanatory variable could take. It is imperative to use non-parametric regression techniques when it is intended to predict a variable response that is impossible or very costly to measure.

Generally, given n observations of two variables x and y, we consider the following regression model:

$$y_i = m(x_i) + \epsilon_i \quad \text{for} \quad i = 1, \ldots, n$$

Being x the input variable (regressor), $m(x)$ an unknown regression function and ϵ the random error, which would represent the non-reducible part of the total prediction error. The classification of regression models, taking into account the assumptions made in the regression function $m(x)$ are:

- **Parametric regression model**: which assumes that the regression function has a predetermined form.

- **Non-parametric regression model**: which only assumes hypothesis of smoothness (in the sense of continuity and differentiability) on the regression function $m(x)$. It does not assume any predefined form as above for the regression function.

Among the purposes of non-parametric regression to estimate regression curves are to provide a versatile method to study the general relationship between two variables and to give predictions based on past observations, even if they have no reference to any fixed parametric model.

Non-parametric methods are more appropriate when there is no prior knowledge of the relationship between the variables under study since they only start from soft assumptions about the regression function. These non-parametric methods are computationally expensive due to the large number of operations involved and are only applicable in practice with the help of a computer program [9].

For this research we will compare the predictions obtained with non-parametric methods with those resulting from the Naïve methods.

The following is a conceptual description of the main basic concepts that will help us to understand the structure and purpose of this work.

2.1 Functional data

To collect the information contained on demand curves we will use functional data analysis techniques, which is a branch of statistics that makes use of information from curves or other forms that the data could have. These curves can reflect very important information about the data itself.

Due to the nature of the data, non-parametric smoothing methods are very useful tools for the analysis of functional data. These will be the methods used in the present research. Specifically, we consider that the demand of electric power for a particular day is a function, of which we only know 24 points, those corresponding to hourly demands. The predictors that will be developed have as input a history of these functions and as output a function of electric power demand, also of 24 values, which will be the prediction obtained for that particular day. Thus, this predicted function is essentially a weighted sum of the electric power demand functions of the past.

To illustrate the above, a power demand function generated from the 24 known points, that is, from the 24 hour demands for that particular day, is presented below.

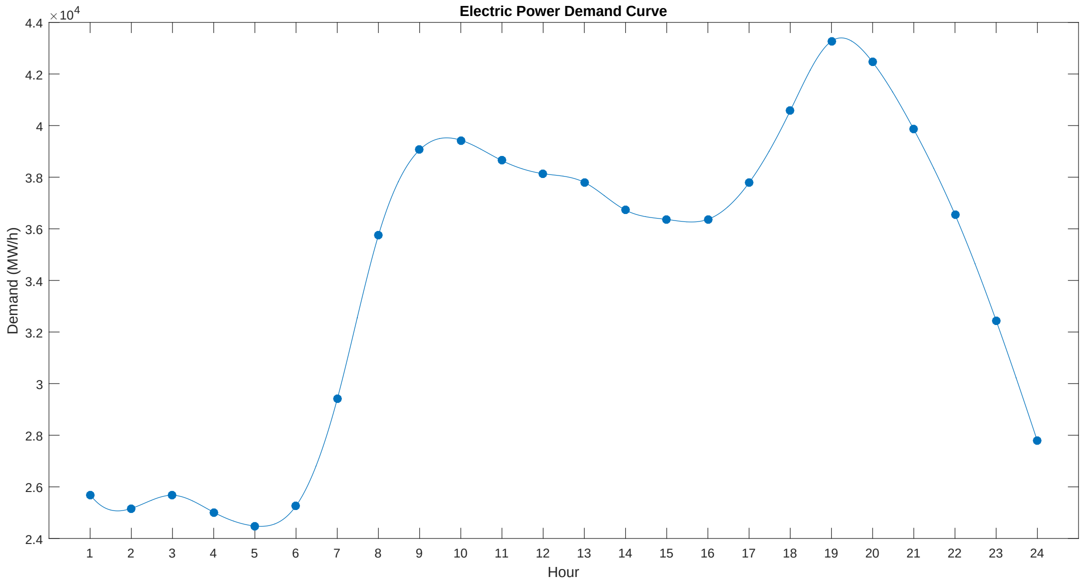


Figure 2.1: Hourly demand

In the figure we can see a curve with 24 points that refer to the hourly demands for that particular day. This will be the functional form that each of our daily demands will have.

2.2 Naïve models

This method corresponds to the family of probabilistic classifiers, based on the application of Bayes' theorem, with strong (naïve) assumptions of independence between the characteristics. Naïve forecasts are the most profitable forecast model and provide a benchmark with which to compare more sophisticated models.

The Naïve method is a supervised learning technique since it needs previous examples to help predict future values. It is one of the most used algorithms due to its simplicity and easy implementation when making predictions. In this method the predictions are generated by an automatic mechanism established a priori, that is, it is a prediction procedure that mechanically repeats a past behavior.

2.2.1 Simple Naïve

This is the most simple form of Naïve models sice it makes predictions assuming that the values obtained by the variable at time k coincide with the past value of the variable at time $k - 1$.

$$\hat{y}_k = y_{k-1}$$

2.2.2 Seasonal Naïve

If the series is considered to have seasonality, the Seasonal Naïve approach may be more appropriate when forecasts are equal to the value of the previous season. Just as in the following equation.

$$\hat{y}_k = y_{k-p}$$

Where p is the length of the seasonal period, which in our case is equal to 7 days, as shown in Figure 1.2, in the previous chapter.

This method explains the seasonality by establishing that each prediction is equal to the last observed value of the season. The main advantage of this method is its ease of implementation. The disadvantage presented by these types of models is that their memory is limited, so they are only used as a reference to evaluate the quality of more complex prediction methods: if a more complex method does not show important reductions of the error with respect to the Naïve model, it should not be considered a good predictor and therefore, neither should it be implemented.

For this work the Naïve prediction models will be a reference to be compared with the different non-parametric models that will be developed in later sections of the work.

2.3 Non-parametric regression

A non-parametric regression is a form of regression analysis in which the predictor does not have a predetermined form, but is constructed according to the information derived from the data, giving structure to the model. The non-parametric methods to be estimated in this section consider that $\hat{y}$ is a weighted average of the known past y_i. It does not make assumptions about the distribution that the data follows, unlike, for example, a linear model, so the best model of the data are the data itself.

The estimator to be used for parametric regression in this research is a Nadaraya-Watson type estimator, which uses a Kernel method for estimating density functions [10, 11]. A very common Kernel is the Gaussian Kernel.

2.3.1 Kernel based estimation

The non-parametric estimation of density functions, using the Kernel method, is clever way of estimating a density function that does not follow a known model (Normal, Binomial, Exponential, etc.). It has an enormous flexibility and what it does is to construct a function of density rotating around the sample values. Therefore, the density estimation by Kernels is no more than an average weighted by the distance of the observations to the point to be estimated. The greater the distance from the point to an element of observations, the lower its weight in the estimate. The weight will be determined by the chosen Kernel function and the value of a γ component. The higher the value of the latter, the lower the weight of those elements of the observations that are far from the point, so γ could be called a locality hyperparameter, where γ is the inverse relation of h (bandwidth or window).

The estimation of the Kernel density function is represented in the following function:

$$\hat{f}_\gamma(x_k) = \gamma \cdot \frac{1}{n} \sum_{i=1}^{k-1} K(\gamma \cdot d(x_i, x_k)) = \gamma \cdot \frac{1}{n} \sum_{i=1}^{k-1} K(u)$$

Where γ is the locality hyperparameter, $d(x_i, x_k)$ is a distance function and $K(u)$ is a Kernel function.

For most applications it is necessary that the Kernel function meets the following conditions

- Normalization:

$$\int_{-\infty}^{\infty} K(u)du = 1$$

- Symmetry:

$$K(-u) = K(u)$$

The choice of the Kernel function is of secondary importance. However, both γ as the distance function are relevant aspects to provide a good asymptotic and practical behavior of the Nadaraya-Watson estimator.

For analysis purposes, the Gaussian kernel function will be used to determine the relative importance of the data in the prediction.

2.3.2 Nadaraya-Watson estimator

This estimator is one of the most used prediction mechanisms in the field of non-parametric regression. It uses the Kernel method of the density function, being defined as shown in the following equation:

$$\hat{m}_\gamma(x_k) = \sum_{i=1}^{k-1} w_\gamma(x_i, x_k) \cdot y_i$$

Where

$$w_\gamma(x_i, x_k) = \frac{K(\gamma \cdot d(x_i, x_k))}{\sum_{j=1}^{n} K(\gamma \cdot d(x_j, x_k))}$$

In the Nadaraya-Watson estimator the function $w(x_i, x_k)$ corresponds to a normalized weight distribution that has the effect of giving more or less importance to the known data in function of its similarity with the data to be predicted, at the time to make an estimation. This distribution of weights is normalized and its values are obtained through the use of the Kernel density function, which will give greater values to those data that have a greater similarity with the data to predict and smaller values to the data that have a smaller similarity.

2.3.3 Gaussian Kernel Function

The Gaussian distribution is a form of Kernel distribution that acts like the density function of a random variable x with Normal distribution, being defined by the following equation

$$K(u) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}u^2} \quad \text{where } u = \gamma \cdot d(x_i, x_k)$$

Now we have chosen a specific Kernel function, we would have to choose a value for the hyperparameter γ and develop a distance function with the objective of specifying the way of obtaining the weight distribution of the Nadaraya-Watson estimator.

2.3.4 Locality Hyperparameter, γ

As previously indicated, the hyperparameter γ is a hyperparameter that gives locality to the distribution of the density of the Kernel function. This hyperparameter is closely related to the badwidth or window (h) of Kernel functions, sice it is the inverse value. In addition, since the hyperparameter γ gives locality in a positive sense, it is expected that if the value of γ increases, also increase the locality of the Kernel function, whereas at low values of γ, the locality will be smaller. This has a direct impact in the functional form that the Kernel function will take: as the values of γ get smaller the function is flattening, which causes the distribution of weights to be increasingly proportional and not penalize the values that are further away. On the contrary, as the values of γ are increasing the function becomes more agressive and much more centered at the origin, so much importance will be given to the data that are closer and very little importance to the data tha is going further away.

The above mentioned can be observed in the following figure, in which the

Gaussian Kernel function is represented for values of greater, smaller and equal to the unit.

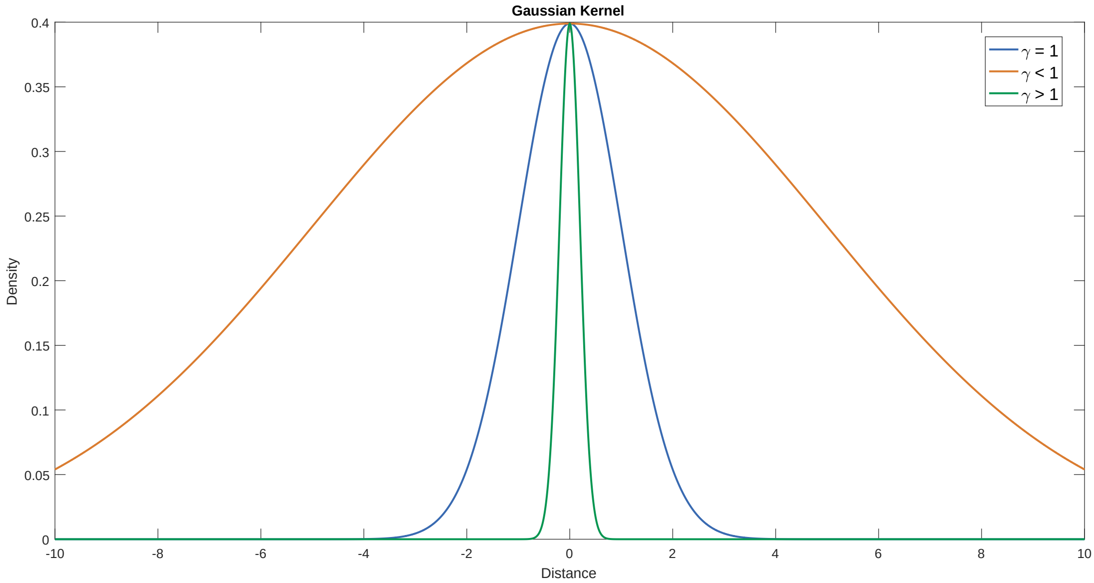


Figure 2.2: Kernel behaviour depending on the values of γ

As discussed above, for values of γ smaller than the unit we have a much wider Kernel function, while for values of γ greater than the unit the Kernel function is distributed much more locally. We can also appreciate the previously commented relation of γ with the bandwidth or window: for values of γ smaller than the unit we see that the Kernel function presents a greater bandwidth, whereas for values of γ greater than the unit, the function Kernel is more centered around the closest values, presenting a much smaller bandwidth.

To illustrate this in a more compact form, the figure below shows different values for γ, and the density distribution that corresponds to it. The Gaussian Kernel distribution gives a Normal distribution for a probability density function of a random variable x. When $\gamma = 1$ we have the Standard Normal distribution with $\mu = 0$ and $\sigma^2 = 1$, or $N \sim (0, 1)$, as represented by the blue line in Figure 2.3.

Figure 2.3 we can observe that when $\gamma = 0.3$ the density distribution is smoother in relation to the Gaussian Kernel distribution function. At the same time, we can see that the bandwidth is larger, i.e. the distribution of weights is less local. In the right panel we can observe that, when $\gamma = 3$, almost all the values of the distribution are at a distance of $\sigma < 1$, unlike what happened in a Standard Normal distribution in which most of the values were concentrated at a distance of $\sigma < 3$. This is how we can show that the bandwidth is reduced when the locality increases, that is, when the values of γ become larger.

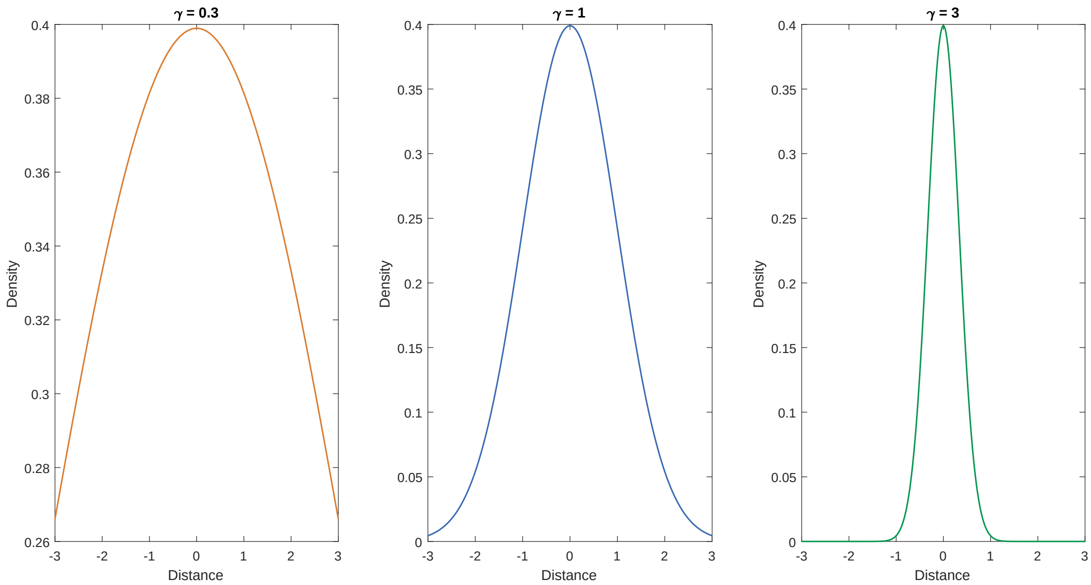


Figure 2.3: Different values for γ

2.3.5 Distance function

The last key aspect to be taken into account, once the Kernel distribution function is chosen and once a value for the hyperparameter γ is obtained, is the definition of a suitable distance function for the specific application that is being carried out, that allows somehow to discriminate between past data similar to the data to be predicted and past data that are not similar to this data to be predicted. This distance function will have its repercussion in the Kernel function, since, if the distance is small, the values that will be obtained of the Kernel function will be those that are placed in the upper part of the density curve of the function, whereas those values that present greater distances as established in the distance function, will be placed in lower parts of the Kernel distribution function.

In general, for this research two types of different distance functions have been considered: those that only incorporate elements in reference to the temporal distance and those that incorporate both elements of temporal distance and distance elements in relation to similarity in terms of the load profile curves. Distances are based on the l^2 norm, but taking into account several distance elements rather than one alone. In subsequent sections a detailed development of all the distance functions considered in the predictors developed for the application in question will be made.

Chapter 3. Practical Application

In this section the different prediction models will be developed, which will consist of those models based on non-parametric regression methods and Naïve models for comparison purposes, as stated in previous chapters.

The prediction to be performed with the training set will cover the last 6 months of 2016, while the prediction for the validation set will comprise the data set of the first 6 months of 2017, using the Matlab mathematical program tools. The form of evaluation of results will be through the comparison of the Mean Absolute Percentage Error (MAPE) obtained by each model.

To obtain the prediction models, a sample of training data will be taken, another sample of validation data and a data historic that will be used for both training and validation. In the training set, the optimal values of the hyperparameters will be obtained, which will minimize the average error of the whole period for each predictor, and then we will apply the predictor with the smallest error to perform predictions in a validation set.

Specifically, the set of training that has been used has been the one corresponding to the hourly demands of electric power for each of the days of the period from July to December of the year 2016 and the set of validation that has been used, corresponds to the hourly demands of electric power for each of the days from January to June 2017. Finally, to generate the predictions of training and validation sets we a historic of hourly demands of electric power for the first 6 months of 2016, that is, each one of the days from January to June.

Once the best prediction model is obtained for each one of the developed predictors, the predictor with the lowest error is selected, given the optimal hyperparameters for each predictor. This will be the predictor that will be used to make predictions in the validation dataset, also using the Naïve models as a reference point to evaluate the performance of the selected predictor.

3.1 Database description

The database used for the UK's electric power demand prediction, has been obtained from the website of National Grid, an electricity transmission operator based on the United Kingdom and the United States. On its website, National Grid has a section dedicated to publications of data related to the commercial operation of the electricity transmission system, in which you can find a historic of the demands of electric power for the United Kingdom. For our study, data for the year 2016, from January 1 to June 30, have been used for the data set used by the National Grid's electrical energy demands database for the set of historical data used by the training sets and validation. From July 1 to December 31 for the training set, the data for the months of January to June 2017 were used for the validation of the best model.

National Grid's database of electric power demands has a 30-minute time interval between each of the observations. That is, the first observation of the database corresponds to the demand for electrical energy for January 1^{st} at 12:00 a.m., while the second observation of the database would correspond to the demand for electric energy for the same day, but at 12:30 a.m., and so on.

Of all the variables that appear in the above mentioned database we will only use three of them: the date, the hour and the total electric power demand for the United Kingdom. The date has a format of dd-month-yyyy, the time has a range of values from 1 to 48 because the temporality is of the database is half an hour, and the electric power demand is measured in MW/h.

The year 2016 was a leap year, so the number of days in the database will be 366.

3.2 Data cleaning and transformation

Once the data is collected, the next step is to clean and transform it in a format that is more suitable for processing than the default format presented by the raw data.

In this research hourly predictions of the demand for electric power are made instead of predictions every half an hour, so it is necessary to filter the original database to only keep the observations that correspond to intervals of one hour. For this, a Python script was designed that, given the original database, generated a new database with one-hour intervals between each of the observations.

For the process of problem modeling and generation of predictions we use Matlab, so for practical purposes, the default date format contained in the database was not the most appropriate, since the date was in a string of characters forma and its handling in Matlab is more complicated in this way. Therefore, the first step in data transformation is the conversion of the date to a numerical format, which is done through the function "datenum()" which is built-in Matlab by default.

Once the dates are converted to a numeric format, the next step is to check if the database we are working with has missing values. As the database contains hourly demands for electric power and there are 366 days in 2016, the number of observations would have to be $366 \cdot 24 = 8784$. If we inspect the number of actual observations that the database in question has, we see that it actually coincides with the number of observations that we expected. However, this does not mean that everything is completely correct, because by doing a exhaustive analysis of the observations, we see that some days have more than 24 observations (surplus values) and that other days have less than 24 observations (missing values).

To solve this problem it is systematically checked, through an automated process that was coded, that for each day there are 24 demand data. In the case that a day is found to have fewer than 24 demands, the missing observations are completed with the average value of the demand for the whole year, whereas if it is detected that there is excess data for a day, we only take into account the first 24 demands.

Once the problem of excess or defect of time demands for each of the days is solved, we continue to have the database in a format where each observation corresponds to each one of the hourly demands for every day of the period considered in the study. Although the predictions we will make will be of hourly, we are interested in making daily predictions, so we will have to change the structure of the database to another more suitable for this purpose. Therefore, the next step is a transformation of the database from a hourly time structure to a daily temporal structure, where each observation is a day and the 24 hour demands are now contained in the columns of each of the observations.

Specifically, the original structure of the database had the following format

Day 1	Hour 1	Electric Power Demand
...	...	...
Day 1	Hour 24	Electric Power Demand
Day 2	Hour 1	Electric Power Demand
...	...	...
Day 2	Hour 24	Electric Power Demand
...	...	...
Day n	Hour 1	Electric Power Demand
...	...	...
Day n	Hour 24	Electric Power Demand

Table 3.1: Original Database Structure

Once the transformation is made, we get a database in the folling format

Day 1	Hourly Electric Power Demands
Day 2	Hourly Electric Power Demands
Day 3	Hourly Electric Power Demands
...	...
Day n	Hourly Electric Power Demands

Table 3.2: New Database Structure

Where the day column has a numerical coding of the date and "Hourly electric power demands" represents a function containing the electrical demands for each hour of the day.

Now the data is in a format suitable to work with them and to be able to elaborate the different models of prediction. Subsequently, since each predictor will use different ways of establishing the similarity between the data to be predicted and each one of the data of the sample, new transformations will be made on the data to adapt them to the format required by each of the predictors. detailed below.

3.3 Prediction accuracy evaluation

Since our goal is to estimate electric power demand through non-parametric techniques we will obtain predictions for each hour of the day. The accuracy of each model and forecast is measured in daily errors in percent for each day of the week. The daily errors (DE) are defined by a variation of the mean absolute percentage error (MAPE) that measures the error size in percentage terms, is defined by the following equation:

$$DE_i = \frac{1}{i}\frac{1}{24}\sum_{d=1}^{i}\sum_{h=1}^{24} e_{dh} \quad \text{for} \quad i = 1, \ldots, n-1$$

Where i is the day to be predicted, n is the number of total days in the prediction period and e_{dh} denotes the relative error rate per hour for a particular day, and is defined as:

$$e_{dh} = 100 \, \frac{|y_d(h) - \hat{y}_d(h)|}{y_d(h)}$$

Where $y_d(h)$ is the actual value of the electric power demand and $\hat{y}_d(h)$ is the prediction made for day d.

The variables $y_d(h)$ and $\hat{y_d}(h)$ correspond to the $\mathbb{R}^{24}$ space, where each function is associated to a particular day.

3.4 Naïve models prediction

The Naïve models are the most basic form of predictions, since they only take the value of previous data values to estimate future values, so they are a good reference when comparing predictions made with more sophisticated methods, as a way of evaluating the performance of these. Therefore, as mentioned previously, we will use predictions generated with Naïve models for the purpose of comparison with the predictions made with non-parametric regression models that correspond to each one of the predictors that have been developed.

The Naïve models considered for this comparison are two: a simple Naïve model, which only considers the value of the immediately previous instant as an estimate; and a Seasonal Naïve model, which considers the value of a temporal instant of 7 days in the past as an estimate.

3.4.1 Simple Naïve model

As it has been detailed above, this method is quite simple when making predictions. It is based on predicting the value of a variable at time t assuming that it coincides with the value of the variable at time $t-1$. Since if we want to predict the demand for a particular day, for example Monday, we will assume that it will be the same as the day before (Sunday). If we want to predict the electric power demand on Sunday, it will be assumed that it will be the value of the previous day i.e. Saturday, and so on. We will assume this behavior for every day of the week.

Next, we are going to represent the predictions with Naïve model in a graph.

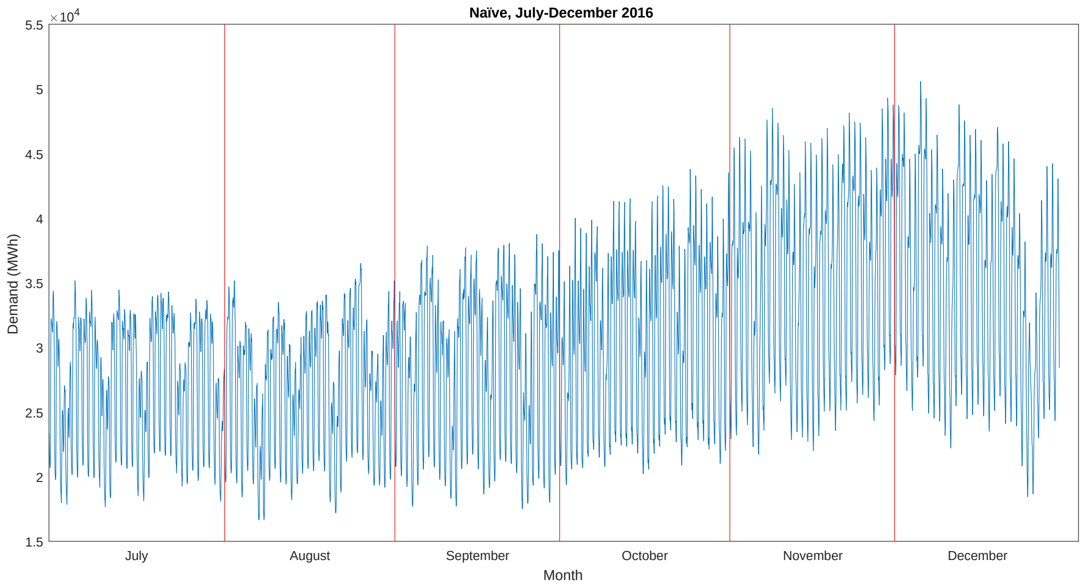


Figure 3.1: Naïve Model Predictions

In Figure 3.1 we can see the behavior of the predictions for the first Naïve model, for each of the days included in the prediction horizon of the 6 months from July to December 2016.

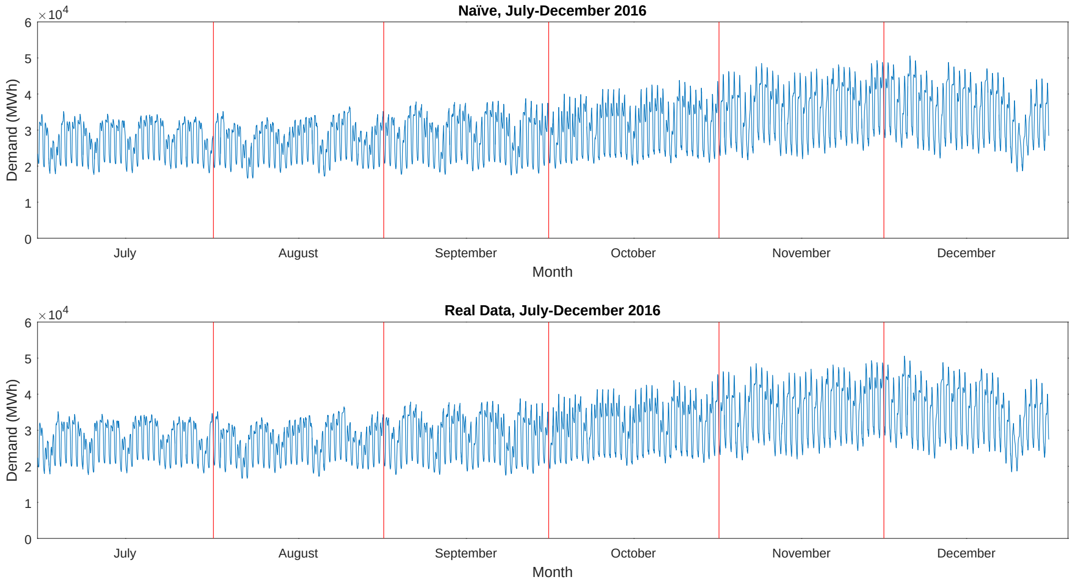


Figure 3.2: Naïve Predictions Compared to Real Data

Although Figure 3.2 allows us to observe how similar Naïve model prediction is with the actual data in a generic way, it would be interesting to have an idea of how much is the margin of difference between both and where the larger and smaller errors occur for the whole of the prediction period. Therefore, a graph of the daily errors for the Naïve model is shown below, which allows us to observe the distribution of errors more easily throughout the study period.

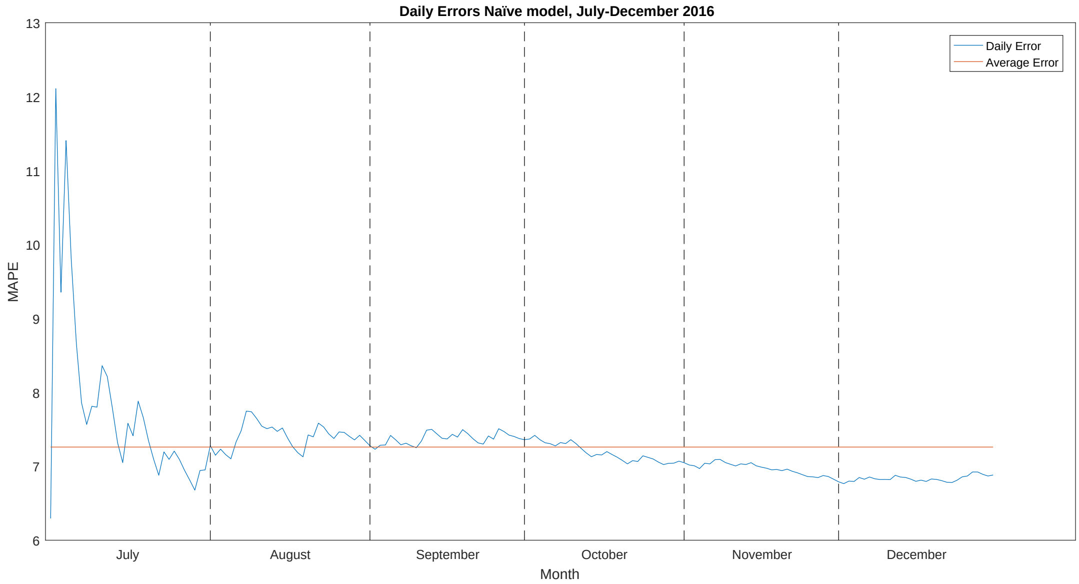


Figure 3.3: Daily Errors for Naïve Model

In Figure 3.3 we obtain the daily prediction errors for the Naïve predictor. We can observe where the estimation error with this predictor is bigger and smaller: the error is higher in the first days of July and starts to decrease as the days elapse until obtaining values more or less stable from the month of August, where the errors are around 6

Next, the table associated with the prediction errors for the first Naïve predictor is shown.

	July	August	September	October	November	December	Total
Monday	8.7132	7.4969	7.4774	7.2376	6.9937	6.8556	7.4624
Tuesday	8.191	7.4443	7.4422	7.2367	6.9883	6.8563	7.3598
Wednesday	7.6833	7.386	7.3809	7.1956	6.9589	6.8582	7.2438
Thursday	7.2715	7.3147	7.3375	7.1558	6.9733	6.8246	7.1462
Friday	6.893	7.3005	7.3105	7.1154	6.9467	6.8064	7.0621
Saturday	8.3279	7.4352	7.3446	7.1913	6.9719	6.8256	7.3494
Sunday	7.7224	7.4394	7.3481	7.1873	6.9611	6.8168	7.2458
						Period Average	7.2671

Table 3.3: Mean Absolute Percentage Error (MAPE) Naïve

In Table 3.1 we can observe the prediction errors for each day of the week for each of th months predicted from July to December 2016. We see that the

Naïve prediction method shows us prediction errors that are not too big. As seen in the error table, the values obtained from the errors are quite good, so this is a reference for comparing with the prediction error values obtained with the predictors that were developed.

3.4.2 Seasonal Naïve model

In the same way that we have represented the Naïve model, we estimate the second model, Seasonal Naïve which uses the value of the previous week as the estimation, assuming that the value of the electric power demand for a given day is equal to the value of the electric power demand for the same weekday but a week before.

In the next figure, we will represent the electric power demand predictions from July to December 2016 for the Seasonal Naïve model.

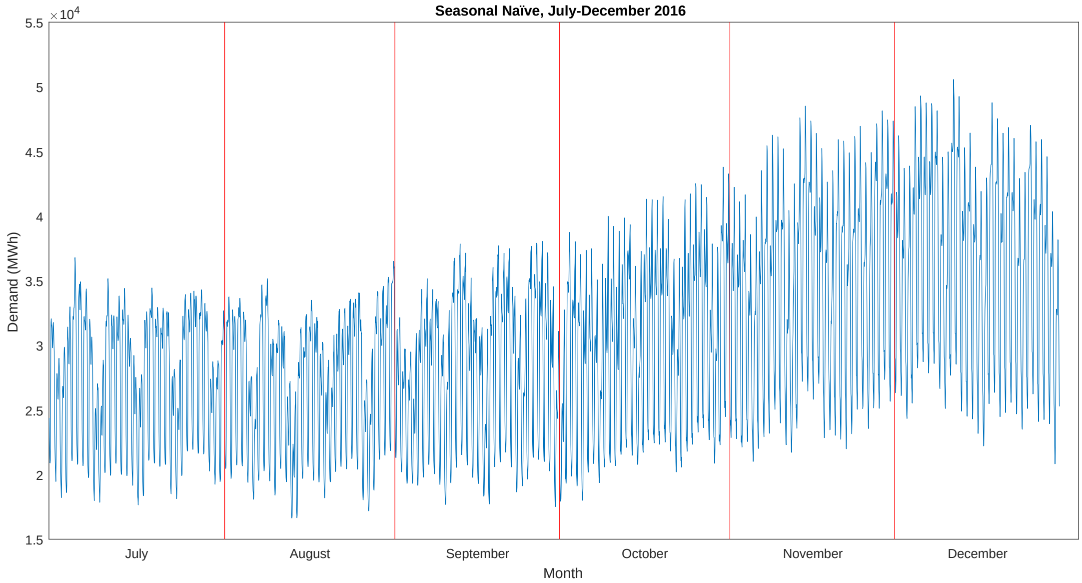


Figure 3.4: Seasonal Naïve Model Predictions

In Figure 3.4 we can see the behavior of the predictions for the Seasonal Naïve model for each one of the 6-month forecast July-December 2016. In this graph we can observe the evolution of the prediction of the electric power demand series. We see that in the months from July to September the series shows an apparent constant behavior, then rising from the month of October to the last month of prediction, reaching a peak in the month of December.

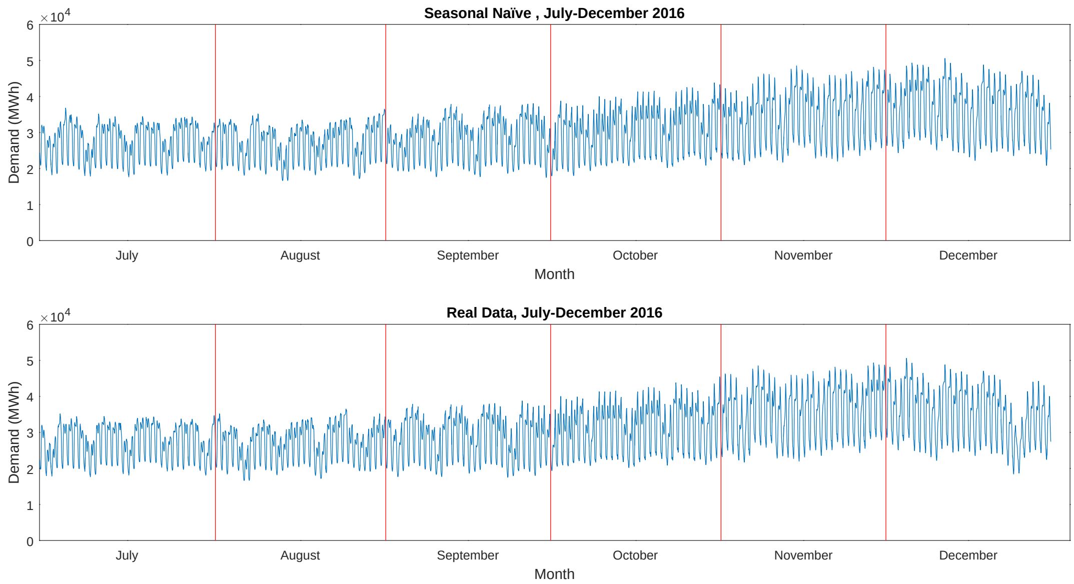


Figure 3.5: Seasonal Naïve Model Predictions Compared to Real Data

Although Figure 3.5 allows us to observe how similar the prediction of the Seasonal Naïve model is compared to the actual data in a generic way, it would be interesting to have an idea of how much is the margin of difference between both of them. Thus, a graph of the daily errors for the Seasonal Naïve model is shown, allowing us to observe in a more compact way the distribution of errors over the study period.

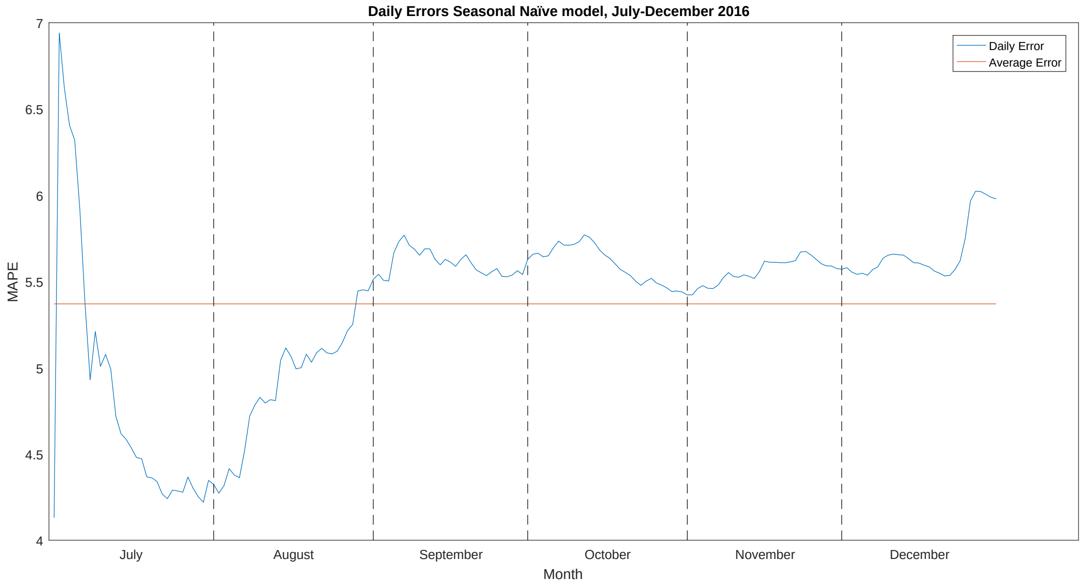


Figure 3.6: Daily Errors for Seasonal Naïve Model

In Figure 3.6 we obtain the daily prediction errors for the Seasonal Naïve model, allowing us to observe where the estimation error with this predictor is greater and lower. The highest error, as well as was the case of the Naïve model, is presented in the first days of the month of July, constantly decreasing as the days elapse until it reaches minimum values between day 20 and 40 of our data, which would be between the end of July and the beginning of August, to rise again from day 40, and reach a point of stabilization, oscillating between 5 and 6 percent error.

Next, the table associated with the prediction errors for the Seasonal Naïve model is shown.

In Table 3.2 we can observe the prediction errors for each day of the week for each of th months predicted from July to December 2016. We see that the Seasonal Naïve prediction method shows us even lower prediction errors than those obtained using the simple Naïve model. As seen in the error table, the values obtained from the errors are better, so this would be an even better reference for comparing with the prediction error values obtained with the predictors that were developed.

Thus our first two models obtained with the Naïve and Seasonal Naïve

	July	August	September	October	November	December	Total
Monday	5.0599	4.9428	5.6351	5.5904	5.5501	5.6824	5.4101
Tuesday	4.9894	4.9322	5.624	5.6234	5.5424	5.6899	5.4002
Wednesday	4.8408	4.9328	5.6174	5.6198	5.5677	5.6886	5.3778
Thursday	4.6582	4.8265	5.5924	5.6208	5.5683	5.6608	5.3212
Friday	4.4328	4.8544	5.583	5.6135	5.5663	5.6756	5.2876
Saturday	5.0293	4.8548	5.5757	5.5925	5.5593	5.6798	5.3819
Sunday	4.9489	4.9751	5.5986	5.5886	5.5502	5.634	5.3826
						Period Average	5.3659

Table 3.4: Mean Absolute Percentage Error (MAPE) Seasonal Naïve

methodology will be a benchmark to test against our developed prediction models.

3.5 Non-parametric regression models prediction

As we mentioned in the previous section, non-parametric regression is a form of regression in which the predictor does not have a predetermined form, but is constructed according to the information derived from the data, giving structure to the model.

Since the developed predictors are based on non-parametric regression methods, as in the previous chapter it was indicated, a Nadaraya-Watson estimator will be used. Therefore, the way to obtain predictions with the proposed predictors is, given a day to predict, to use all previous days and to obtain a weighted mean that will be our prediction.

In general, the prediction for a given day, k, would be expressed as follows:

$$\hat{y}_k = \sum_{i=1}^{k-1} w_\gamma(x_i, x_k) \cdot y_i$$

Where $y_i \in \mathbb{R}^{24}$ is the function of real demands for day i, and where $\hat{y}_i \in \mathbb{R}^{24}$ is the demand function to be predicted for day k

Being the weight function defined as:

$$w_\gamma(x_i, x_k) = \frac{K(\gamma \cdot d(x_i, x_k))}{\sum_{j=1}^{n} K(\gamma \cdot d(x_j, x_k))}$$

As it was already mentioned, the Kernel function that has been considered for our study has been the Gaussian Kernel.

This will be the general prediction equation used by each of the developed predictors. The difference between each one of our predictors will be in the value of the hyperparameters, the form of the regressor and, therefore, the definition of the distance function that depends on the form of the regressor. Since this equation gives the generic way of making the predictions through a Nadaraya-Watson estimator, the following sections will detail the form of the regressor and the distance function for each one of the different developed predictors.

For the development of each prediction model, two dimensions have been considered, which will be taken into account when establishing similarities between the data to be predicted and the previous data, as a way of distributing the weights for each of the data in a suitable way. These two dimensions are as follows:

3.5.1 Time dimension

The time dimension tries to capture the similarity between observations taking into account the variables that somehow collect the temporal essence of the problem of prediction of the electric power demand. This is because, as discussed above throughout the present investigation, the data present weekly but also annual seasonal structure.

Thus, the time dimension will be composed of 3 different components:

- **Day similarity**: it tries to establish how similar is an observation with respect to another taking into account the day of the week. It can only have two values, a value if the day to predict is equal to the observation with which it is being compared (1), and another value if the day to predict is different from that of the observation with which it is being compared (0).

The variable that will be used to establish the day similarity will be coded in a binary format composed by 7 columns, one for each day of the week, and will be called "binary day". In this variable, each column will represent a day of the week, starting on Monday and ending on Sunday, and being each of the columns coded in binary format only one of the columns will have the value 1 while all the other columns will have the value 0. The column containing the value 1 will be the one that tells us on which day of the week we are.

For example, Monday would be represented as [1 0 0 0 0 0], while Friday would be represented as [0 0 0 0 1 0 0].

- **Time distance**: it tries to establish how similar is an observation with respect to another depending on the number of periods that exist between them. Periodicity will be established in two different ways, which will result in different ways of measuring the time distance between observations.
  - **Monthly distance**: the time distance is measured by the difference between the months of the day to predict and each of the observations used to make the prediction. The variable that will be used to establish the monthly distance between observations will be a numeric variable that will show in which month of the year the observation is located. This variable will be called **"decimal month"** and will take values from 1 (representing January) to 12 (representing December).
  - **Daily distance**: the time distance is measured by the difference between the day that occupies the observation to be predicted in the time series and each of the positions that occupy the observations that are used to make the prediction. The variable that will be used to measure the daily distance between the observations will be a variable of numeric type that will indicate in which day of the established time horizon is the prediction. For our specific case, day 1 of the time horizon would correspond to January 1, 2016, while day 366 would correspond to December 31, 2016. This variable will be called **"relative day"**.
  - **Annual distance**: it tries to establish how similar is an observation with respect to another by taking into account the difference in years that exists between observations.

3.5.2 Functional dimension

The functional dimension tries to capture the similarity between the observations in terms of the functional structure that the load profile curves associated with the electrical demand present.

Since the functional dimension is extracted from the function of hourly demands for each day, we enconunter a problem regarding this issue: for purposes of comparison of similarity between the day to be predicted and the observations of the sample, we have no load curve for the day to predict, since this is precisely what we want to obtain with our predictions. To solve this problem, the load curve that is associated with the day to predict is taken from another day which is assumed to have a similar behavior in terms of functional structure. In this sense, we propose two different options:

- **Load Profile Curve of the previous day**: it is considered that the functional structure of the load curve of the day to be predicted has to be very similar to the functional structure of the load curve of the previous day. The variable that will capture the functional structure of the load curve of the previous day will be a functional variable of dimension $\mathbb{R}^{24}$ that will contain the hourly demands of the previous day to the day being predicted. The name of this variable will be **"load profile curve of the previous day"**.

- **Load profile curve of the previous week**: it is considered that the functional structure of the load curve of the day to be predicted has to be very similar to the functional structure of the load curve of the same day, but of the previous week. The variable that will capture this functional structure will be a functional variable of dimension $\mathbb{R}^{24}$ that will contain the hourly demands of the day of the week previous to the day that is being predicted. The name of this variable will be the **"load profile curve of the previous week"**.

3.5.3 Time dimension based models

Under this dimension two predictors has been developed in this research that are detailed below:

3.5.3.1 Predictor 1

This predictor contains the binary day, decimal month and the year as components of the regressor of variable we want to estimate, electric power demand. Therefore, the regressor would have the following form:

$$x_k = [binary\ day(x_k), decimal\ month(x_k), year(x_k)]$$
$$x_i = [binary\ day(x_i), decimal\ month(x_i), year(x_i)]$$
$$x_k, x_i \in \mathbb{R}^9$$
$$\text{where:} \quad binary\ day \in \mathbb{R}^7,\ decimal\ month \in \mathbb{R} \text{ and } year \in \mathbb{R}$$

This predictor, in order to make estimates of future values of the electric power demand, takes only into account the time dimension and determines the similarity between the observations taking into account if each one of the data coincides in day of the week with the day to predict, as well as how much time distance measured in months and years exists between each observation and the day to predict.

Therefore, the distance function for this predictor will be as follows:

$$d(x_i, x_k) = \|p_1 \cdot (binary\ day(x_i) - binary\ day(x_k)) + p_2 \cdot (decimal\ month(x_i) - decimal\ month(x_k)) + (year(x_i) - year(x_k))\|_2$$

The hyperparameters p_1 and p_2 are scaling factors that give more or less importance to the components of binary day and decimal month on the result of the distance function.

Predictor 1 Results

Next, the predictions of the electric power demand for the period from July to December of 2016 obtained with Predictor 1 are shown.

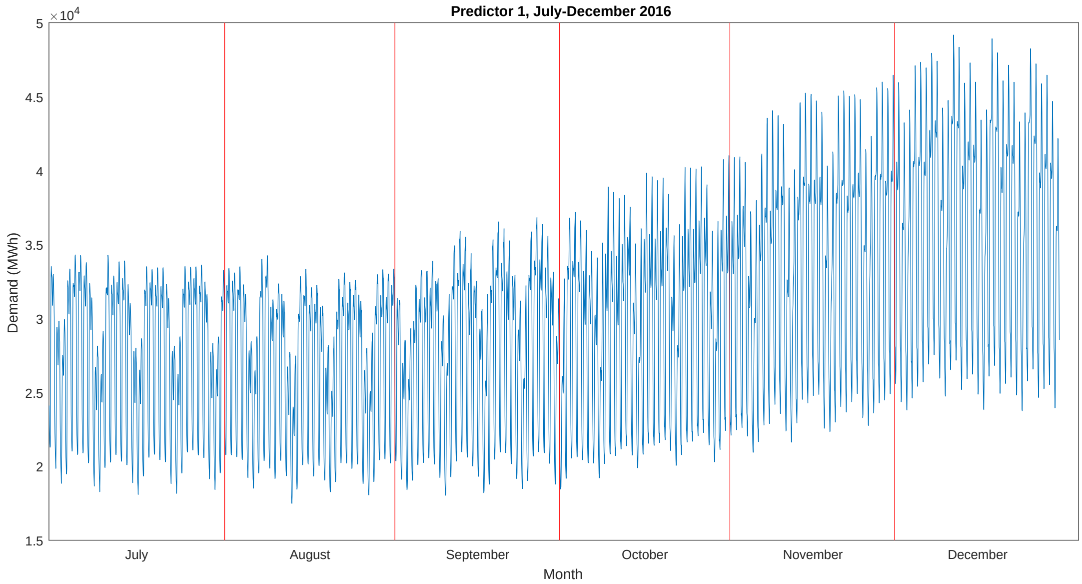


Figure 3.7: Predictor 1 predictions

In Figure 3.7 we can see the predictions of the electric power demand for for the chosen time horizon of prediction, which is from July to December 2016. As we can see, the forecasts show a constant trend over the first three months, and from the month of October, a growing trend can be distinguished in the estimation of the demand for electric power, which reaches its peak and stabilizes in the month of December. As for the volatility of the demand, we can observe that from the month of October the volatility tends to be higher.

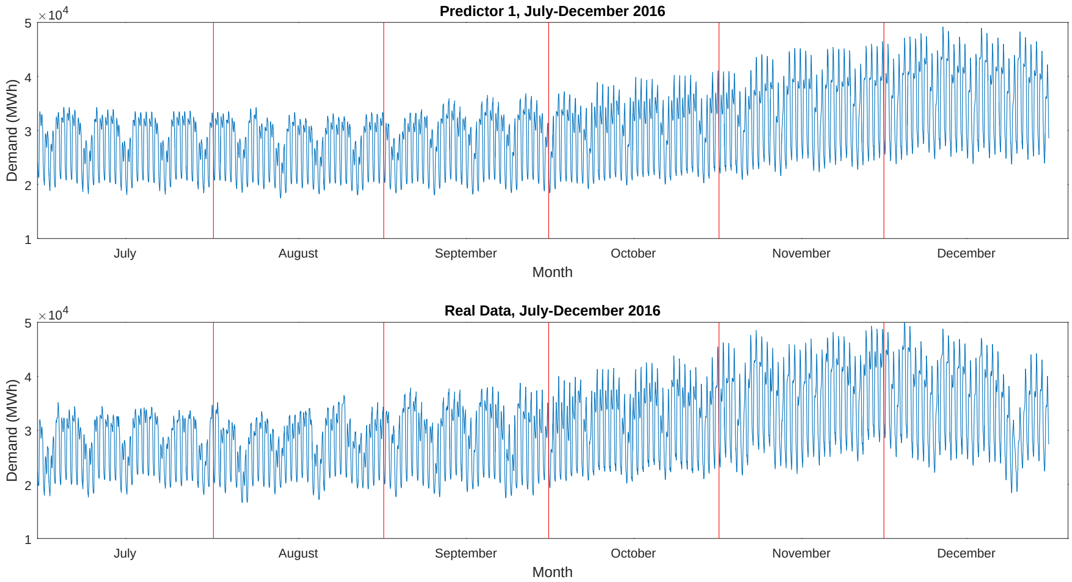


Figure 3.8: Predictor 1 Predictions Compared to Real Data

In Figure 3.8, it can be seen how similar is the prediction obtained with Predictor 1 and the data except for some months, where Predictor 1 achieves an almost uniform trend for the chosen months of the electric power demand series, while the actual data present some volatility in some weeks. We could then say that Predictor 1 captures a constant trend but fails to completely simulate the actual data. What might also be of importance is to have an idea about the error associated with the prediction. Therefore a graph of daily errors for Predictor 1 is shown, which allows us to observe in a more compact way the distribution of errors over the study period.

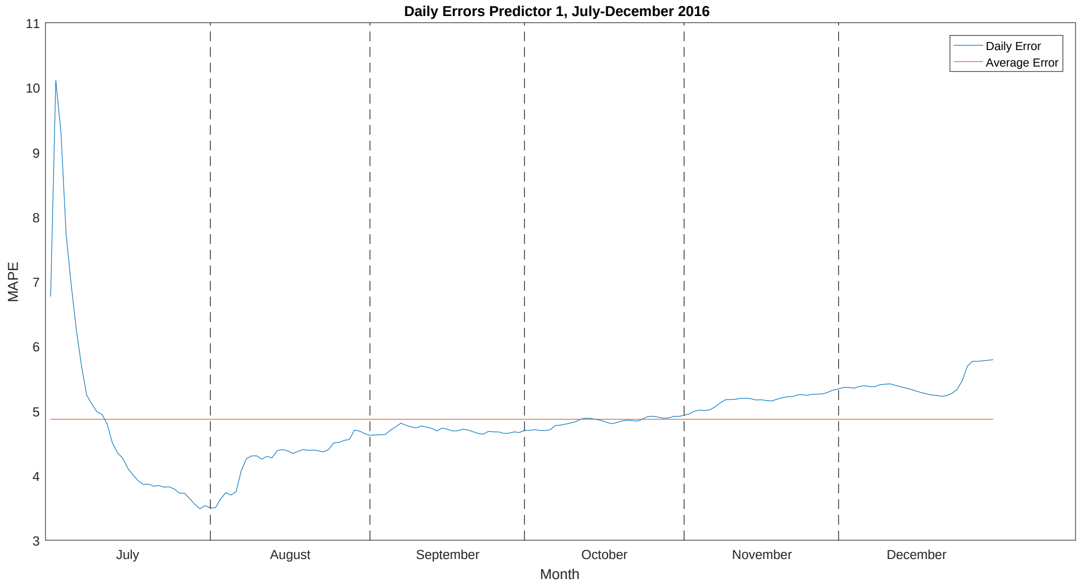


Figure 3.9: Daily Errors for Predictor 1

Figure 3.9 shows that the percentage of error starts with a peak around 10% in the first days of July, decreasing rapidly to reach its minimum value around 3.5% in the last days of July and early days of the month of August. From here the error grows gradually to stabilize around 5% on average for the following months.

The following table shows in detail the average prediction errors for each of the days of the week and for each month of the prediction period.

	July	August	September	October	November	December	Total
Monday	5.0976	4.2487	4.7096	4.8267	5.1818	5.4267	4.9152
Tuesday	4.831	4.2477	4.7093	4.8113	5.1455	5.4421	4.8645
Wednesday	4.5855	4.2667	4.7077	4.8257	5.1731	5.4294	4.8314
Thursday	4.3797	4.216	4.6919	4.8359	5.144	5.4107	4.7797
Friday	4.735	4.227	4.6802	4.8487	5.1465	5.4251	4.8438
Saturday	5.3255	4.2385	4.6842	4.8162	5.1582	5.4355	4.943
Sunday	5.1329	4.3517	4.6922	4.82	5.168	5.3743	4.9232
						Period Average	4.8715

Table 3.5: Mean Absolute Percentage Error (MAPE) Predictor 1

We can observe in the previous table that the average prediction errors have decreased with respect to the Seasonal Naïve model, obtaining error values for each day of the week below 5% in average. The 5% error is the maximum recommended limit in the literature by authors such as Ranaweera, Karady & Farmer (1997), so we have started to have more reliable prediction methods that could be incorporated into real prediction systems.

3.5.3.2 Predictor 2

This predictor contains as components the binary day, relative day and year as part of our variable, electric power demand. Therefore, the regressor would have the following form:

$$x_k = [binary\ day(x_k), relative\ day(x_k), year(x_k)]$$
$$x_i = [binary\ day(x_i), relative\ day(x_i), year(x_i)]$$
$$x_k, x_i \in \mathbb{R}^9$$
$$\text{where:} \quad binary\ day \in \mathbb{R}^7, \ relative\ day \in \mathbb{R} \text{ and } year \in \mathbb{R}$$

This predictor, in order to make estimates of future values of the electric power demand, takes only into account the time dimension and determines the similarity between the observations taking into account if each one of the data coincides in day of the week with the day to predict, as well as how much time distance measured in days and years exists between each observation and the day to predict.

Therefore, the distance function for this predictor will be as follows:

$$d(x_i, x_k) = \|p_1 \cdot (binary\ day(x_i) - binary\ day(x_k)) \ + \ p_2 \cdot (relative\ day(x_i) - relative\ day(x_k)) + (year(x_i) - year(x_k))\|_2$$

As for predictor 1, the hyperparameters p_1 and p_2 are scaling factors that will determine the weight of both the binary day and the relative day in the value of the distance function.

Predictor 2 Results

Next, we will present the evolution graph of predictions obtained with predictor 2 for our series of electric power demand in the time horizon of research, July to December 2016.

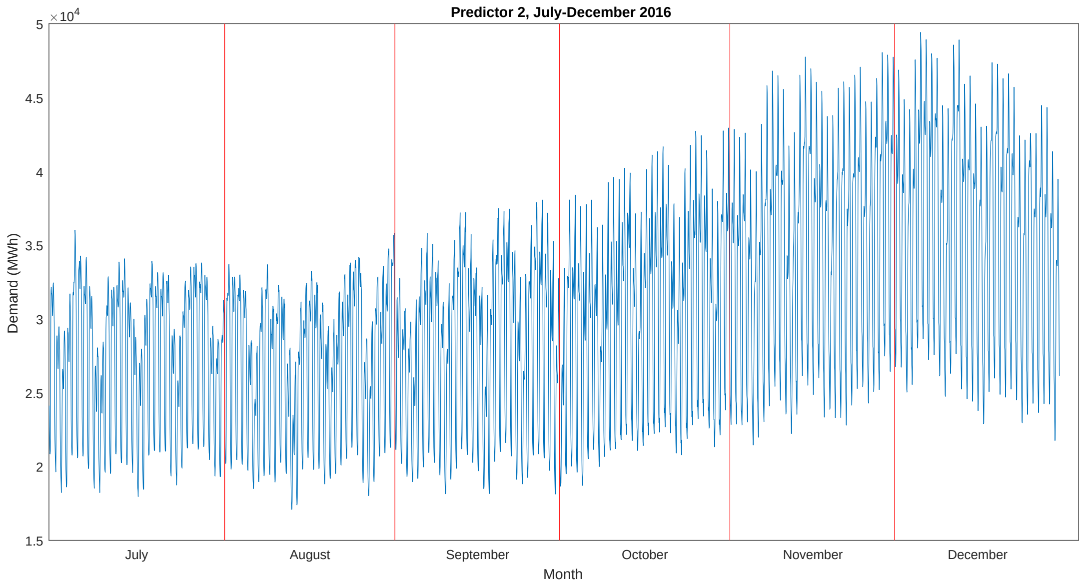


Figure 3.10: Predictor 2 predictions

In Figure 3.10 it can be seen that the prediction has a stable trend for the first 3 months of prediction from July to September of 2016 as in Predictor 1, to then increase from the last week of September and reach a peak in the first week of December to subsequently have a decreasing behavior until the end of the prediction period.

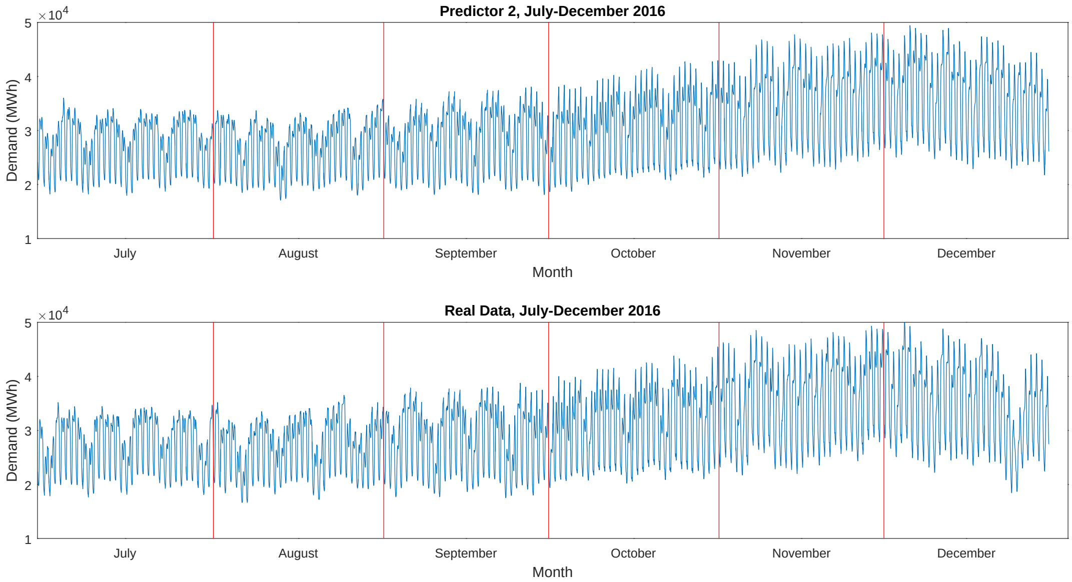


Figure 3.11: Predictor 2 Predictions Compared to Real Data

In Figure 3.11 we can see with greater clarity the evolution of the real data and those predicted with the predictor 2 for our electric power demand series. The predictor shows a trend that is almost similar to the real data in the first three months from July to September, similar to the behavior of Predictor 1, but this new predictor fails to represent the behavior of our demand series in the last month of the study period. With these results in view what could also be of importance is to have an idea of the error associated with the prediction. For this reason, a graph of the daily errors for Predictor 2 is shown, which allows us to observe in a more compact way the distribution of errors over the study period.

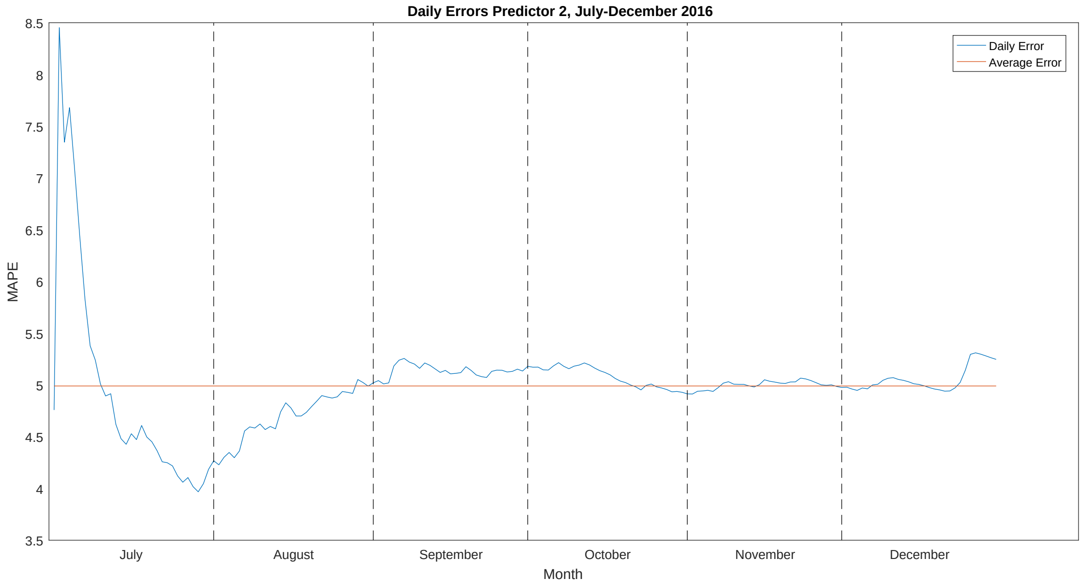


Figure 3.12: Daily Errors for Predictor 2

As shown in Figure 3.12, for our Predictor 2, the error for the prediction period presents its highest value in the first days of July, to show a decreasing behavior in the same month until dropping to a low error level of around 4%. From here it begins to have an increasing trend from August and to present some stability in error values of 5% in averagefor the months from September to December of our study period.

The following table shows in detail the average prediction errors for each of the days of the week and for each month of the prediction period, for the predictor 2

	July	August	September	October	November	December	Total
Monday	5.3257	4.728	5.173	5.0754	4.9951	5.0703	5.0612
Tuesday	5.1364	4.6998	5.1653	5.1025	4.9931	5.0675	5.0274
Wednesday	4.9004	4.6972	5.1513	5.094	5.0091	5.0672	4.9865
Thursday	4.6726	4.6256	5.1232	5.093	5.0104	5.0433	4.928
Friday	4.5579	4.642	5.1118	5.0835	5.0046	5.0525	4.9087
Saturday	5.3034	4.6645	5.1035	5.0829	4.9985	5.0576	5.0351
Sunday	5.046	4.7641	5.1239	5.0678	4.9859	5.0336	5.0035
						Period Average	4.9929

Table 3.6: Mean Absolute Percentage Error (MAPE) Predictor 2

We can observe in Table 3.6 that the mean prediction errors have decreased with respect to the Seasonal Naïve model, obtaining error values for each day of the week around 4.9 and 5.06%. These results still continue to show a good evolution and improvement in prediction for our series of electric power demand, so we could also say that this model could be a good reference of implementation in the field of reliable prediction.

3.5.4 Time and functional dimension based models

Under this dimension four predictors has been developed in this research that are detailed below:

3.5.4.1 Predictor 3

This predictor contains the binary day, decimal month, year and load profile curve of the previous day as components. This last component is a functional variable that will represent the functional structure of the load curve of the previous day, that will be defined in $\mathbb{R}^{24}$ and will contain the hourly demands of the previous day to the day that is being predicted. Therefore the regressor would have the following form:

$$x_k = [binary\ day(x_k), decimal\ month(x_k), year(x_k), load\ profile\ curve(x_{k-1})]$$

$$x_i = [binary\ day(x_i), decimal\ month(x_i), year(x_i), load\ profile\ curve(x_i)]$$

$$x_k, x_i \in \mathbb{R}^{33}$$

$$\text{where:} \quad binary\ day \in \mathbb{R}^7,\ decimal\ month \in \mathbb{R}\ ,\ year \in \mathbb{R} \text{ and}$$

$$load\ profile\ curve \in \mathbb{R}^{24}$$

The predictor 3 to make estimates of future values of the electric power demand takes into account both the time dimension and the functional dimension, and determines the similarity between the observations taking into account if each one of the data coincides in day of the week with the day to predict , as well as how much temporal distance measured in months and years, and the functional similarity of the load curve of the previous day that exists between each observation and the day to be predicted.

Therefore, the distance function for this predictor will take the following form:

$$d(x_i, x_k) = \|p_1 \cdot (binary\ day(x_i) - binary\ day(x_k)) + p_2 \cdot (decimal\ month(x_i) - decimal\ month(x_k)) + (year(x_i) - year(x_k)) + p_3 \cdot \frac{1}{24} \sum_{h=1}^{24} (load\ curve_h(x_i) - load\ curve_h(x_{k-1}))\|_2$$

The hyperparameters p_1, p_2 and p_3 are scaling factors to give more or less importance to the components of binary day, decimal month and load curve of the previous day on the result of the distance function.

Predictor 3 Results

The following graph will show the prediction results obtained with Predictor 3.

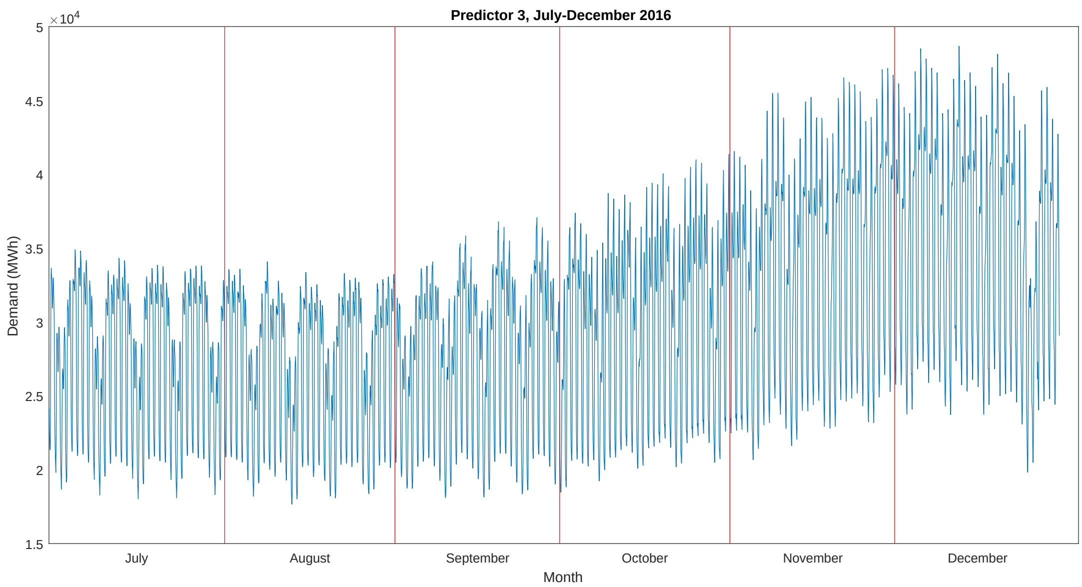


Figure 3.13: Predictor 3 predictions

The predictions of electric power demand with Predictor 3 as we can see in Figure 3.13 shows a stable trend in the first three months from July to

September. We could say that the behavior is similar to the behavior of the other two predictors because they somehow follow the same pattern of behavior of the real data in the first months and then show a similar behavior to the actual data for the months from October to December. Considering that these new predictions already show us a better behavior and predictions of the electric power demand series are more similar to the real data, this is an aspect to emphasize due to the tendency that the latter picks from the data like none of the previous two predictors , this being a very important aspect when considering a good model as we can observe in Figure 3.14.

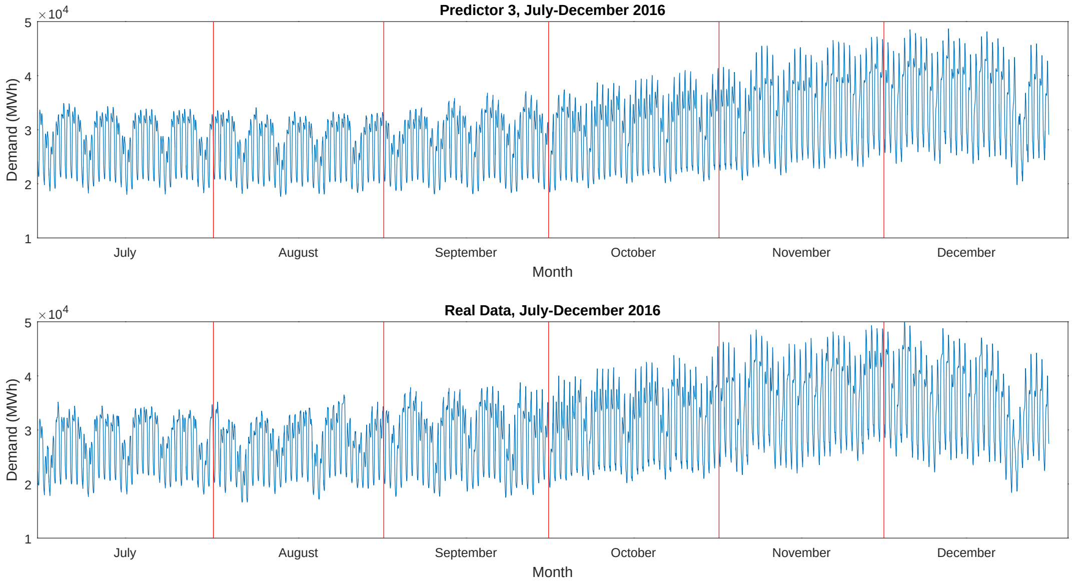


Figure 3.14: Predictor 3 Predictions Compared to Real Data

Due to the large amount of data in Figure 3.14, we cannot clearly see where are the smaller and larger prediction errors, so in the following figure we show in a the prediction errors for the period of study

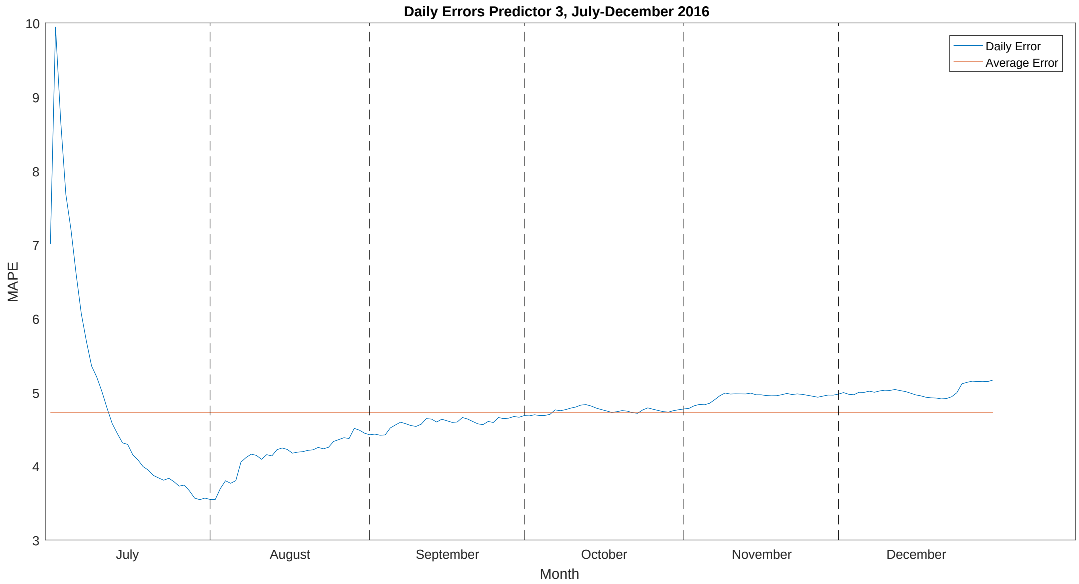


Figure 3.15: Daily Errors for Predictor 3

The behavior of the errors for our Predictor 3 takes is very similar to the behavior shown by the two predictors with temporal dimension as far as it refers to the error in the first days of July, to later show a decreasing behavior and smaller error in a percentage of 3.5%. From the August to December it also presents a constant error value around 4% and 5.5%.

The following table shows in detail the mean prediction errors for each of the days of the week and for each month of the prediction period, for Predictor 3.

	July	August	September	October	November	December	Total
Monday	5.1414	4.1357	4.6193	4.7502	4.9554	5.0242	4.771
Tuesday	4.9273	4.1312	4.6189	4.7505	4.9251	5.0232	4.7294
Wednesday	4.7171	4.1438	4.6117	4.7551	4.9355	5.0203	4.6973
Thursday	4.5113	4.1047	4.5758	4.7598	4.93	5.0053	4.6478
Friday	4.8837	4.1195	4.5643	4.7646	4.9311	5.0125	4.7126
Saturday	5.3902	4.1351	4.5383	4.7351	4.9271	5.0224	4.7914
Sunday	5.0876	4.2181	4.5449	4.7354	4.932	5.0098	4.7546
						Period Average	4.7292

Table 3.7: Mean Absolute Percentage Error (MAPE) Predictor 3

In the prediction error table for our Predictor 3, we can see that the error values are even lower than the two previous models, this behavior had already been observed in the figure of predictions for Predictor 3, in which we have observed the good simulation that makes this predictor in relation to the actual data of our electric power demand series.

3.5.4.2 Predictor 4

This predictor contains the binary day, relative day, year and load profile curve of the previous day as components. This last component is a functional variable that will represent the functional structure of the load curve of the previous day, that will be defined in $\mathbb{R}^{24}$ and will contain the hourly demands of the previous day to the day that is being predicted. Therefore the regressor would have the following form:

$$x_k = [binary\ day(x_k), relative\ day(x_k), year(x_k), load\ profile\ curve(x_{k-1})]$$

$$x_i = [binary\ day(x_i), relative\ day(x_i), year(x_i), load\ profile\ curve(x_i)]$$

$$x_k, x_i \in \mathbb{R}^{33}$$

$$\text{where:}\quad binary\ day \in \mathbb{R}^7,\ relative\ day \in \mathbb{R}\ ,\ year \in \mathbb{R} \text{ and}$$

$$load\ profile\ curve \in \mathbb{R}^{24}$$

This predictor takes into account both the time dimension and functional dimension in order to make estimates for future values of the electric power demand series and also determines the similarity between the observations taking into account if each of the data coincides in day of the week with the day to predict, as well as how much temporal distance measured in days and year, and the functional similarity of the load curve of the previous day that exists between each observation and the day to predict.

Therefore, the distance function for this predictor will be as follows:

$$d(x_i, x_k) = \|p_1 \cdot (binary\ day(x_i) - binary\ day(x_k))\ +\ p_2 \cdot (relative\ day(x_i)$$

$$-\ relative\ day(x_k))\ +\ (year(x_i) - year(x_k))$$

$$+ p_3 \cdot \frac{1}{24} \sum_{h=1}^{24} (load\ curve_h(x_i) - load\ curve_h(x_{k-1}))\|_2$$

The hyperparameters p_1, p_2 and p_3 as well as for Predictor 3 are scaling factors to give more or less importance to the components of binary day, relative day and load curve of the previous day on the result of the distance function.

Predictor 4 Results

In the following graphs we can observe the predictions obtained with Predictor 4 for our series of electric power demand for the months of July to December of 2016.

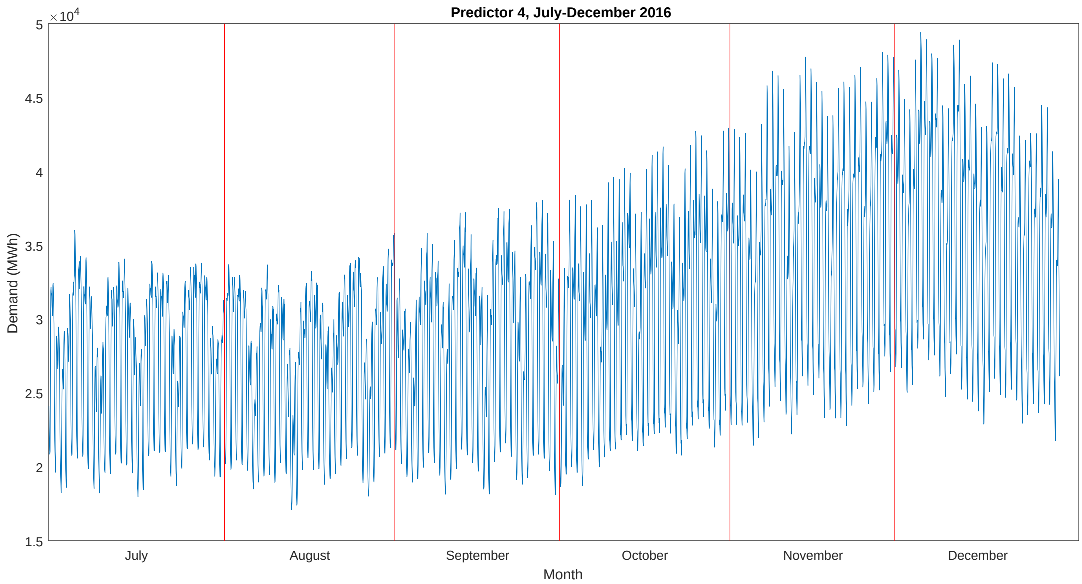


Figure 3.16: Predictor 4 predictions

The evolution of the predictions estimated with Predictor 4 presents a constant trend in the first months and shows the same behavior as our previous predictors for the first months of prediction. It is important to highlight that unique tendency that has been obtained until now with the predictors that have been developed. In the following graph we will show in a more compact way what the evolution of our Predictor 4 has been in relation to the actual data.

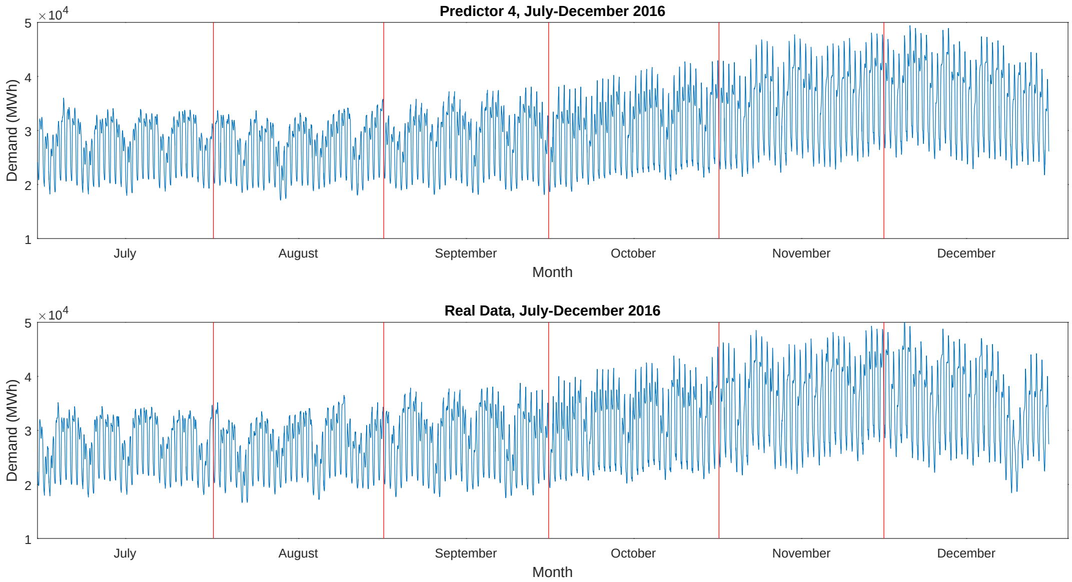


Figure 3.17: Predictor 4 Predictions Compared to Real Data

This new figure that has been obtained with predictions from Predictor 4 for our study series, apparently shows similar behavior in the first few months as we mentioned, but fails to obtain good predicted values in some months, so it does not pick up the trend of the last month especially in the last weeks of December. To get more information on the relationship between our estimated and real values we show the graph of daily prediction errors.

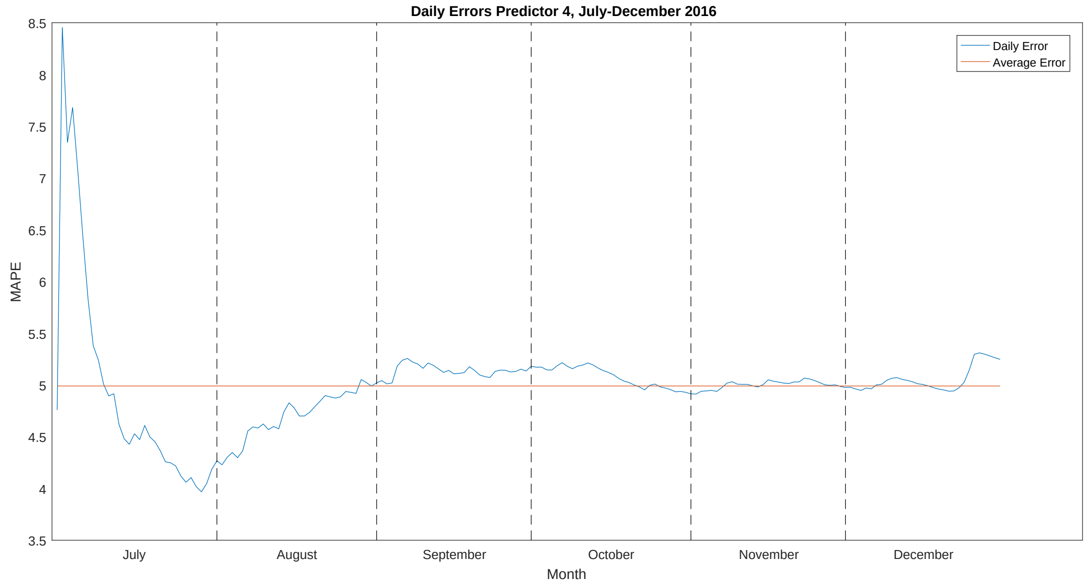


Figure 3.18: Daily Errors for Predictor 4

The errors for our Predictor 4 takes an initial value at around 8.5% in the first month of prediction, then presents a decreasing trend and the error takes values at around 4%, from the month of August the error again takes on increasing values and then shows a constant trend from the month of September to December, showing a 5% error on average.

The following table shows in detail the mean prediction errors for each of the days of the week and for each month of the prediction period, for the Predictor 4.

	July	August	September	October	November	December	Total
Monday	5.3257	4.728	5.173	5.0754	4.9951	5.0703	5.0612
Tuesday	5.1364	4.6998	5.1653	5.1025	4.9931	5.0675	5.0274
Wednesday	4.9004	4.6972	5.1513	5.094	5.0091	5.0672	4.9865
Thursday	4.6726	4.6256	5.1232	5.093	5.0104	5.0433	4.928
Friday	4.5579	4.642	5.1118	5.0835	5.0046	5.0525	4.9087
Saturday	5.3034	4.6645	5.1035	5.0829	4.9985	5.0576	5.0351
Sunday	5.046	4.7641	5.1239	5.0678	4.9859	5.0336	5.0035
						Period Average	4.9929

Table 3.8: Mean Absolute Percentage Error (MAPE) Predictor 4

In this prediction error table, we can see that the error values are still low in relation to those obtained with the Naïve and Naïve seasonal models. This behavior had already been observed in the figure of predictions comparison for the Predictor 4 and real data, in which we have observed the behavior of this predictor in relation to the actual data of our electric power demand series.

3.5.4.3 Predictor 5

This predictor contains the binary day, decimal month, year and load profile curve of the previous week as components. This last component is a functional variable that will represent the functional structure of the load curve of the same day but a week ago, that will be defined in $\mathbb{R}^{24}$ and will contain the hourly demands of the same day a week before to the day that is being predicted. Therefore the regressor would have the following form:

$$x_k = [binary\,day(x_k), decimal\,month(x_k), year(x_k), load\,profile\,curve(x_{k-7})]$$

$$x_i = [binary\ day(x_i), decimal\ month(x_i), year(x_i), load\ profile\ curve(x_i)]$$

$$x_k, x_i \in \mathbb{R}^{33}$$

$$\text{where:} \quad binary\ day \in \mathbb{R}^7,\ decimal\ month \in \mathbb{R}\ ,\ year \in \mathbb{R} \text{ and}$$

$$load\ profile\ curve \in \mathbb{R}^{24}$$

This predictor, to make estimates of the future values of the series, takes into account the time dimension and the functional dimension, and also determines the similarity between the observations taking into account if each of the data coincides in day of the week with the day to predict, as well as how much time distance measured in month and year and the functional similarity of the load curve of the previous week that exists between each observation and the day to predict.

Therefore, the distance function for this predictor will be as follows:

$$d(x_i, x_k) = \|p_1 \cdot (binary\ day(x_i) - binary\ day(x_k))\ +\ p_2 \cdot (decimal\ month(x_i)$$

$$-\ decimal\ month(x_k))\ +\ (year(x_i) - year(x_k))$$

$$+ p_3 \cdot \frac{1}{24} \sum_{h=1}^{24} (load\ curve_h(x_i) - load\ curve_h(x_{k-7}))\|_2$$

The hyperparameters p_1, p_2 and p_3 as well as for the other predictors are scaling factors to give more or less importance to the components of binary day, decimal month and load curve of the previous week on the result of the distance function.

Predictor 5 Results

In the following graphs we can observe the predictions estimated by Predictor 5 for our study period

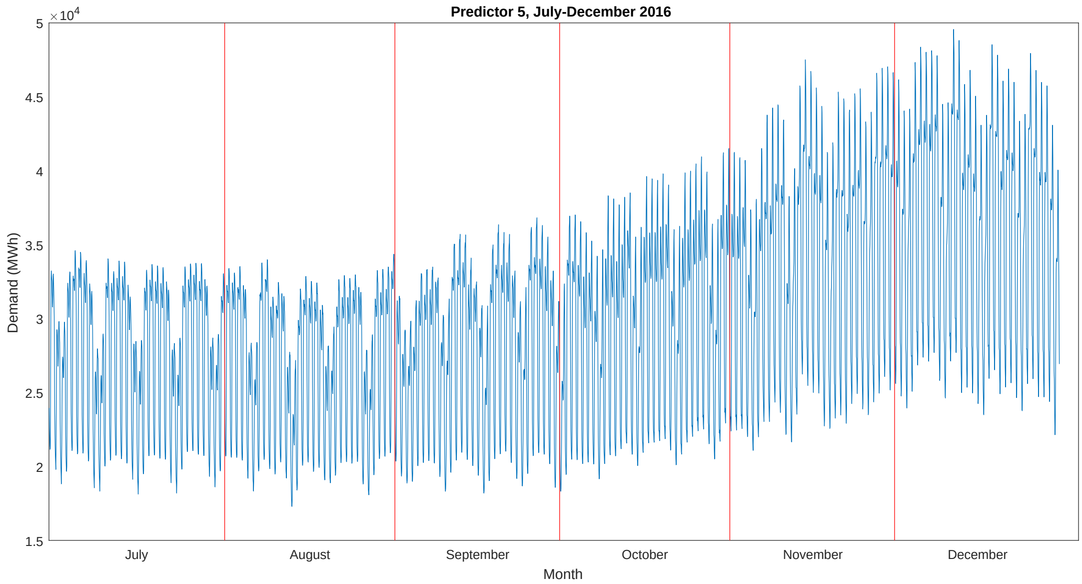


Figure 3.19: Predictor 5 predictions

We can observe that, as it happens for the previous predictions, in the first months the predictions present a constant trend until the end of the month of September. From there, the predictions present a growing trend, reaching its maximum at the beginning of the month of December, where it starts to decrease again.

For comparison purposes, a graph with predictions for Predictor 5 and the actual data for the study period is presented

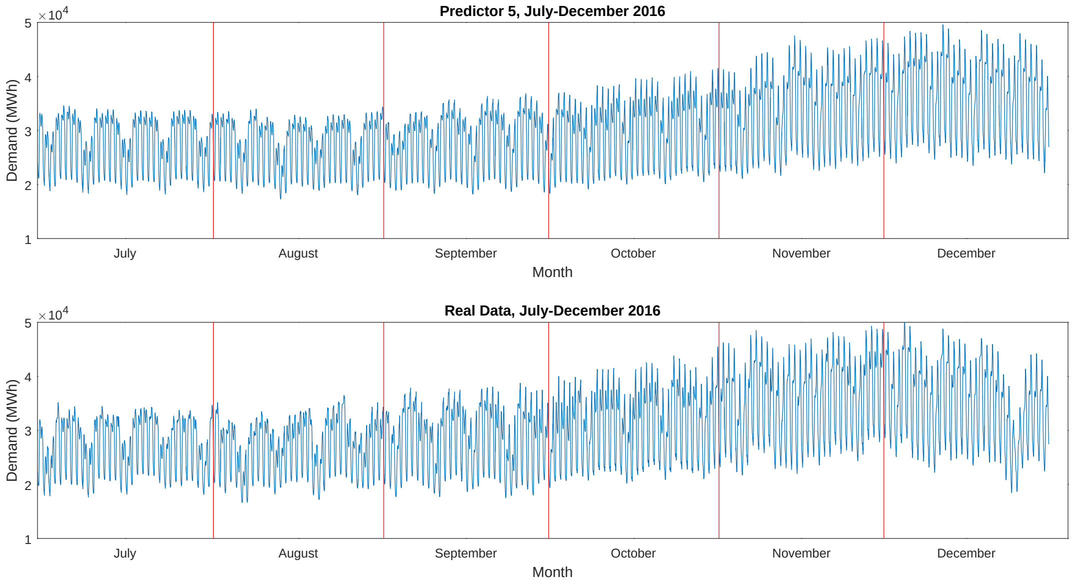


Figure 3.20: Predictor 5 Predictions Compared to Real Data

This new graph apparently shows a similar behavior in the first few months as we mentioned, but fails to obtain good enough predicted values in some months, especially for the month of December, so in order to get more information on the relationship between our estimated and real values we show the daily errors graph.

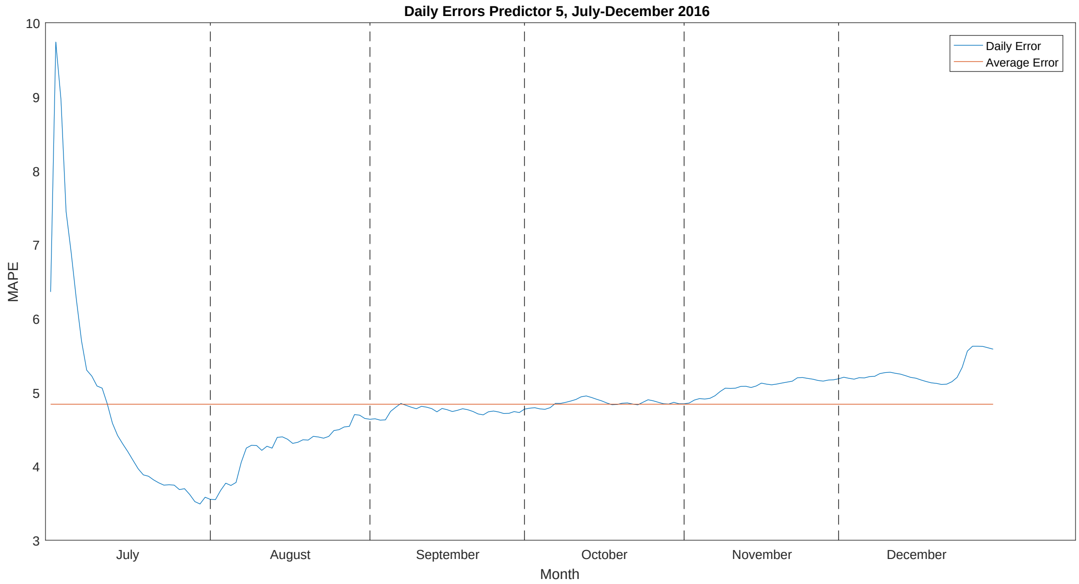


Figure 3.21: Daily Errors for Predictor 5

The errors for our Predictor 5 take a start value around 9.8% in the first month of July prediction and then presents a decreasing trend getting as low as 3.5% by the end of this month. From August, the error again takes increasing values and then shows a constant trend from the month of September to December with an error value of 5% on average.

The following table shows in detail the mean prediction errors for each of the days of the week and for each month of the prediction period, for Predictor 5.

	July	August	September	October	November	December	Total
Monday	5.0534	4.2585	4.7636	4.8485	5.0777	5.2859	4.8813
Tuesday	4.8241	4.254	4.7641	4.8532	5.0521	5.2968	4.8407
Wednesday	4.5997	4.2627	4.7612	4.8581	5.081	5.2923	4.8092
Thursday	4.3864	4.1987	4.7381	4.8658	5.0652	5.2672	4.7536
Friday	4.6522	4.2164	4.726	4.8711	5.0659	5.2794	4.8018
Saturday	5.2778	4.2282	4.72	4.8429	5.07	5.2823	4.9036
Sunday	5.0931	4.347	4.7358	4.8462	5.072	5.2342	4.888
						Period Average	4.8397

Table 3.9: Mean Absolute Percentage Error (MAPE) Predictor 5

In this error table for predictor 5, we can see that the error values are still low relative to those obtained with the Naïve and Seasonal Naïve models. This behavior had already been observed in the predictions and real data comparison graph, in which we have observed the behavior of this predictor in relation to the actual data of our electric power demand series. This table is then a good reference for showing us the good behavior of this predictor.

3.5.4.4 Predictor 6

This predictor contains the binary day, relative day, year and load profile curve of the previous week as components. This last component is a functional variable that will represent the functional structure of the load curve of the same day but a week ago, that will be defined in $\mathbb{R}^{24}$ and will contain the hourly demands of the same day a week before to the day that is being predicted. Therefore the regressor would have the following form:

$$x_k = [binary\ day(x_k), relative\ day(x_k), year(x_k), load\ profile\ curve(x_{k-7})]$$

$$x_i = [binary\ day(x_i), relative\ day(x_i), year(x_i), load\ profile\ curve(x_i)]$$

$$x_k, x_i \in \mathbb{R}^{33}$$

$$\text{where:} \quad binary\ day \in \mathbb{R}^7,\ relative\ day \in \mathbb{R}\ ,\ year \in \mathbb{R} \text{ and}$$
$$load\ profile\ curve \in \mathbb{R}^{24}$$

This predictor, to make estimates of the value of the series, takes into account the time dimension and the functional dimension, and also determines the similarity between the observations taking into account if each of the data coincides in day of the week with the day to predict, as well as how much time distance measured in days and year and the functional similarity of the load curve of the previous week that exists between each observation and the day to predict.

Therefore, the distance function for Predictor 6 will be as follows:

$$d(x_i, x_k) = \|p_1 \cdot (binary\ day(x_i) - binary\ day(x_k))\ +\ p_2 \cdot (relative\ day(x_i)$$
$$-\ relative\ day(x_k))\ +\ (year(x_i) - year(x_k))$$

$$+ p_3 \cdot \sum_{h=1}^{24} (load\ curve_h(x_i) - load\ curve_h(x_{k-7}))\|_2$$

The hyperparameters p_1, p_2 and p_3 as well as for the other predictors are scaling factors to give more or less importance to the components of binary day, relative day and load curve of the previous week on the result of the distance function.

Predictor 6 Results

In this section will be shown the resulting graphs for the values estimated by Predictor 6 of the electric power demand series for the entire study period from July to December 2016.

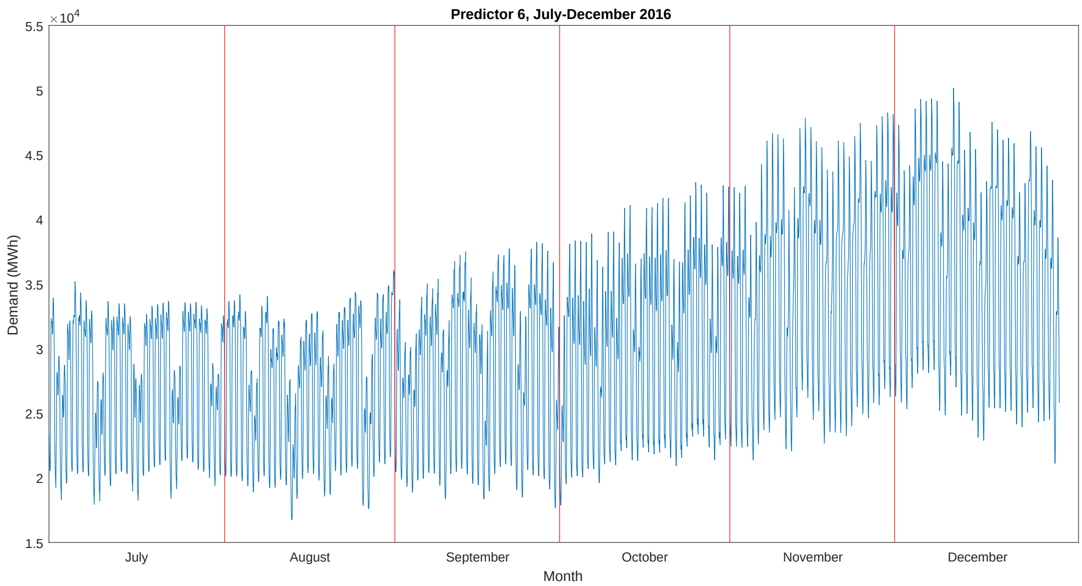


Figure 3.22: Predictor 6 predictions

In the previous figure we can observe that, as it happens for the previous predictions, in the first months predictions present a constant tendency until the end of the month of September. From there, predictions presents a growing trend, reaching its highest value in the month of December, and then descending until the end of the prediction period.

For comparison purposes, the prediction graph for Predictor 6 and the actual data is shown below

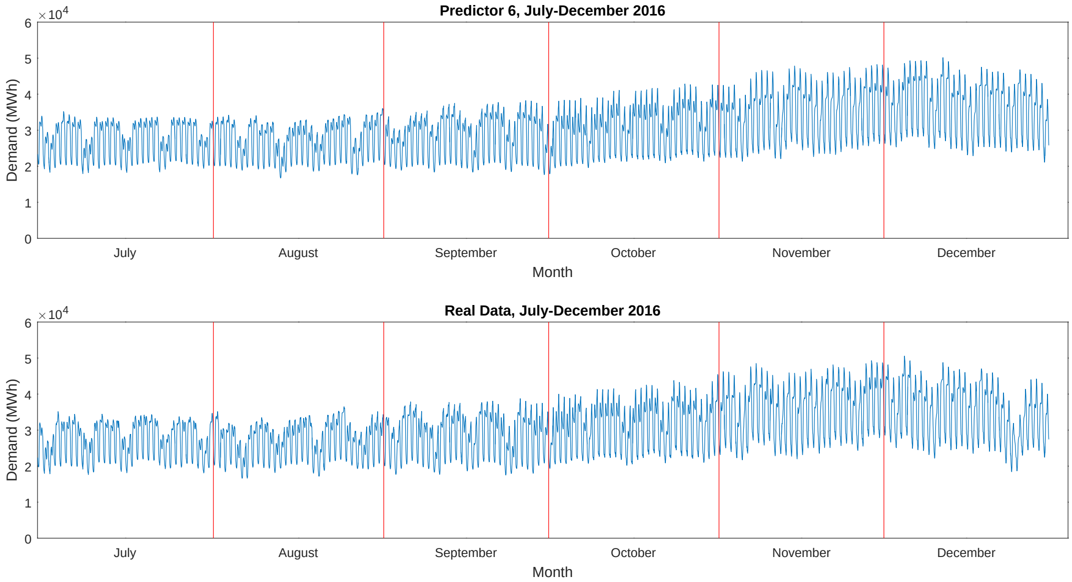


Figure 3.23: Predictor 6 Predictions Compared to Real Data

In this figure, predictions apparently show a similar behavior in the first months as we could observe in the previous figure, but as it was the case with Predictor 5, this predictor fails to obtain good predicted values in some months, especially for the month of December. To observe the relationship between our estimated and actual values in more detail we show the daily prediction error for this predictor

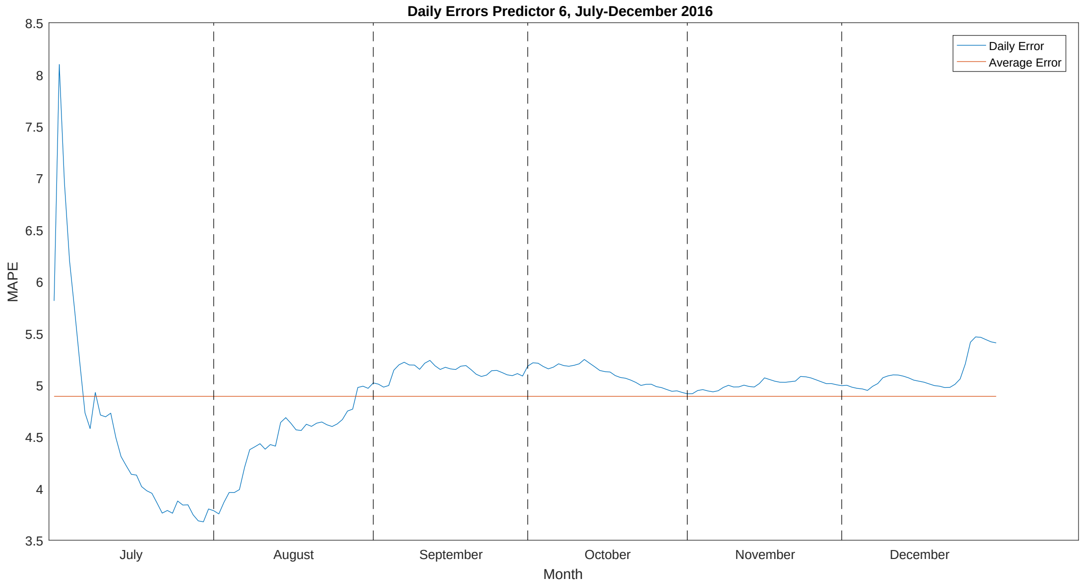


Figure 3.24: Daily Errors for Predictor 6

We see that the errors take a starting value around 8.2% in the first month, falling to a value of 3.8% by the end of July. In August, the error again takes increasing values and then shows a constant trend from September to December, presenting an average error of 5%.

The following table shows in detail the average prediction errors for each of the days of the week and for each month of the prediction period, for Predictor 6.

	July	August	September	October	November	December	Total
Monday	4.6969	4.4924	5.1722	5.0918	4.9922	5.1161	4.9269
Tuesday	4.5655	4.478	5.1565	5.1203	4.9912	5.1211	4.9054
Wednesday	4.3742	4.4864	5.1406	5.1146	5.013	5.1227	4.8752
Thursday	4.1608	4.381	5.1151	5.1055	5.0144	5.0936	4.8117
Friday	4.4121	4.4176	5.1074	5.096	5.0103	5.1053	4.8581
Saturday	4.9251	4.4366	5.1043	5.0945	5.0042	5.1115	4.946
Sunday	4.672	4.5597	5.1317	5.0924	4.994	5.0692	4.9198
						Period Average	4.8919

Table 3.10: Mean Absolute Percentage Error (MAPE) Predictor 6

In this table, we can observe that the error values are still low in relation to those obtained with the Naïve and Seasonal Naïve models. This had already been observed in Figure 3.23, in which we have observed the behavior of this predictor in relation to the actual data. This table is then a good reference that shows us the good behavior of this predictor.

3.5.5 Hyperparameter tuning

As already discussed above, in both the Kernel function and the distance functions for each of the different predictors, there are certain hyperparameters that affect the final result that will be obtained for each of the different weights. Three of these hyperparameters, p_1, p_2 and p_3, affect the behavior of the distance function by acting as multipliers of each of the components of the distance function: the hyperparameter p_1 is a scaling factor for the difference that exists on the day of the week between the day to predict and each of the observations, the hyperparameter p_2 is a scaling factor for the time distance (decimal month or relative day) that exists between the day to predict and each of the observations of the sample and, finally, the hyperparameter p_3 is a scaling factor for the difference between the functional structure of the load curves assimilated to the day to be predicted and each of the load curves of the days contained in the sample.

On the other hand, the other hyperparameter that affects the final result of the weights obtained for each one of the observations of the sample, is the hyperparameter γ, which as previously mentioned it is a locality hyperparameter that determines the functional structure of the Kernel distribution function.

Thus, it is necessary to establish a value for each of the aforementioned hyperparameters so that each predictor has the best possible performance and presents a lower prediction error value when estimating future electric power demands. To obtain the optimum value of each of the hyperparameters, an exhaustive method has been used, which is detailed below.

First, a range of values to test for each hyperparameter has been established. Subsequently all combinations of possible hyperparameters have been generated given this range and for each combination of hyperparameters a daily prediction (of 24 hourly predictions) has been obtained for each predictor, for each of the days of the established prediction period. Once the daily prediction matrix has been obtained, the daily error associated with each prediction has been obtained and finally the mean error for the entire prediction period for that specific combination for each of the predictors has been obtained. Finally, a table is generated in which each combination is systematically stored and its mean prediction errors for the whole period for each one of the predictors, obtaining a table with the following format:

Hyperparameters Values				Average Predictor Errors (MAPE) for the prediction period							
p_1	p_2	p_3	γ	Naïve	Seasonal Naïve	Predictor 1	Predictor 2	Predictor 3	Predictor 4	Predictor 5	Predictor 6
Combination 1				Error Comb. 1	Error Comb. 1	Error Comb. 1	Error Comb. 1	Error Comb. 1	Error Comb. 1	Error Comb. 1	Error Comb. 1
Combination 2				Error Comb. 2	Error Comb. 2	Error Comb. 2	Error Comb. 2	Error Comb. 2	Error Comb. 2	Error Comb. 2	Error Comb. 2
Combination 3				Error Comb. 3	Error Comb. 3	Error Comb. 3	Error Comb. 3	Error Comb. 3	Error Comb. 3	Error Comb. 3	Error Comb. 3
...				...	...	...	...	...	...	...	...
Combination n				Error Comb. n	Error Comb. n	Error Comb. n	Error Comb. n	Error Comb. n	Error Comb. n	Error Comb. n	Error Comb. n

Table 3.11: Average Predictor Errors (MAPE)

The domain values that have been tested for each hyperparameter have been:

$$p_1 \in \{0, 0.5, 1, ..., 6\} \rightarrow p_1 \in \{a_k\}_{k=1}^{13} \quad \text{with} \quad a_k = 0.5 \cdot (k-1)$$

$$p_2 \in \{0, 0.5, 1, ..., 6\} \rightarrow p_2 \in \{a_k\}_{k=1}^{13} \quad \text{with} \quad a_k = 0.5 \cdot (k-1)$$

$$p_3 \in \{0, 0.5, 1, ..., 6\} \rightarrow p_3 \in \{a_k\}_{k=1}^{13} \quad \text{with} \quad a_k = 0.5 \cdot (k-1)$$

$$\gamma \in \{0, 0.1, 0.2, \dots, 2\} \rightarrow \gamma \in \{a_k\}_{k=1}^{21} \quad \text{with} \quad a_k = 0.1 \cdot (k-1)$$

Given the domains of previous values for each of the hyperparameters, we obtain a total of combinations of:

$$\text{Total Combinations} = 12 \cdot 12 \cdot 12 \cdot 20 = 34560$$

Once all the mean errors of the prediction period for each combination and for each of the predictors are obtained, the next step is to identify which combination of hyperparameters obtains the least error for each predictor. After filtering this information we obtain the following table where the optimal values for each of the hyperparameters are indicated:

	p_1	p_2	p_3	γ	Error
Naïve					7.27
Seasonal Naïve					5.36
Predictor 1	3.5	3.5		0.6	4.87
Predictor 2	5	1		1.1	4.99
Predictor 3	1.6	4.5	1	0.4	4.73
Predictor 4	5	1	0	1.1	4.99
Predictor 5	5.5	5.5	1.5	0.3	4.84
Predictor 6	4.5	1.5	5	0.2	4.89

Table 3.12: Optimal values for hyperparameters

3.6 Analysis of results

Having detailed all the predictors that have been developed for this study, and having observed the accuracy of the predictions of each of them in an individualized way, it is interesting to make comparisons among them to determine which one is the predictor that generates the predictions that are closest to the actual data.

To do this, we first present graphically, together, all the predictions made by each of the predictors and the actual data for several prediction periods, as detailed below.

First, the predictions and actual data are shown for a one-week time horizon, for each of the training set prediction months. In particular, the week of the second Monday of each month has been chosen for comparison, since it has been considered that the second week of each month would be representative because it covers the central days of each month and we would be able to capture the monthly essence of the data, considering also that not all the first days of each month begin in Monday. In this way we can see the behavior of the predictors in a standardized way, for a whole week from Monday to Sunday.

July

Next, the graph corresponding to the week of the second Monday of July for the actual data and all the predictions obtained with each predictor

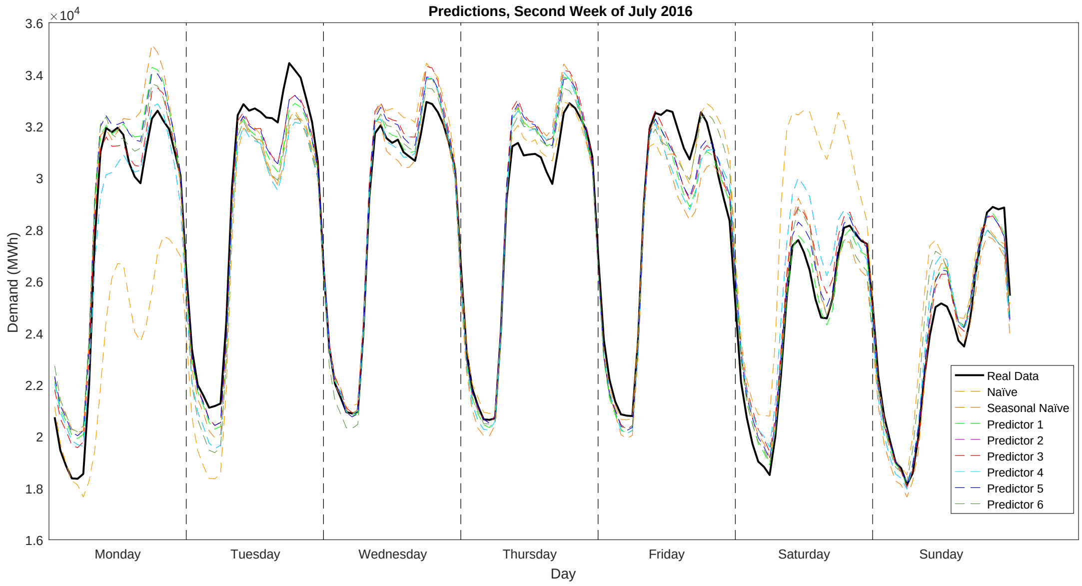


Figure 3.25: Second Week of July, All Predictors

In Figure 3.25, it is not possible to have a good visualization of the predictions for purposes of comparison with the real data. This is because there are many different predictors on the same graph and it is not easy to distinguish them.

Given this problem, the proposed solution has been to group and show different predictors with similar characteristics and visualize them together with the Naïve models and the real data. Thus, we will show the predictors that only take into account the time dimension (Predictors 1 and 2) on the one hand, the predictors that take into account the time dimension and that to incorporate the functional dimension using the load curve of the previous day (Predictors 3 and 4) on the other hand, and finally, those predictors that incorporate the time dimension and the functional dimension through the incorporation of the load curve of the previous week (Predictors 5 and 6). This is the graphical presentation format that will be used in the following months of prediction.

Next, we show the prediction graphs for the week of the second Monday of July for the different types of predictors

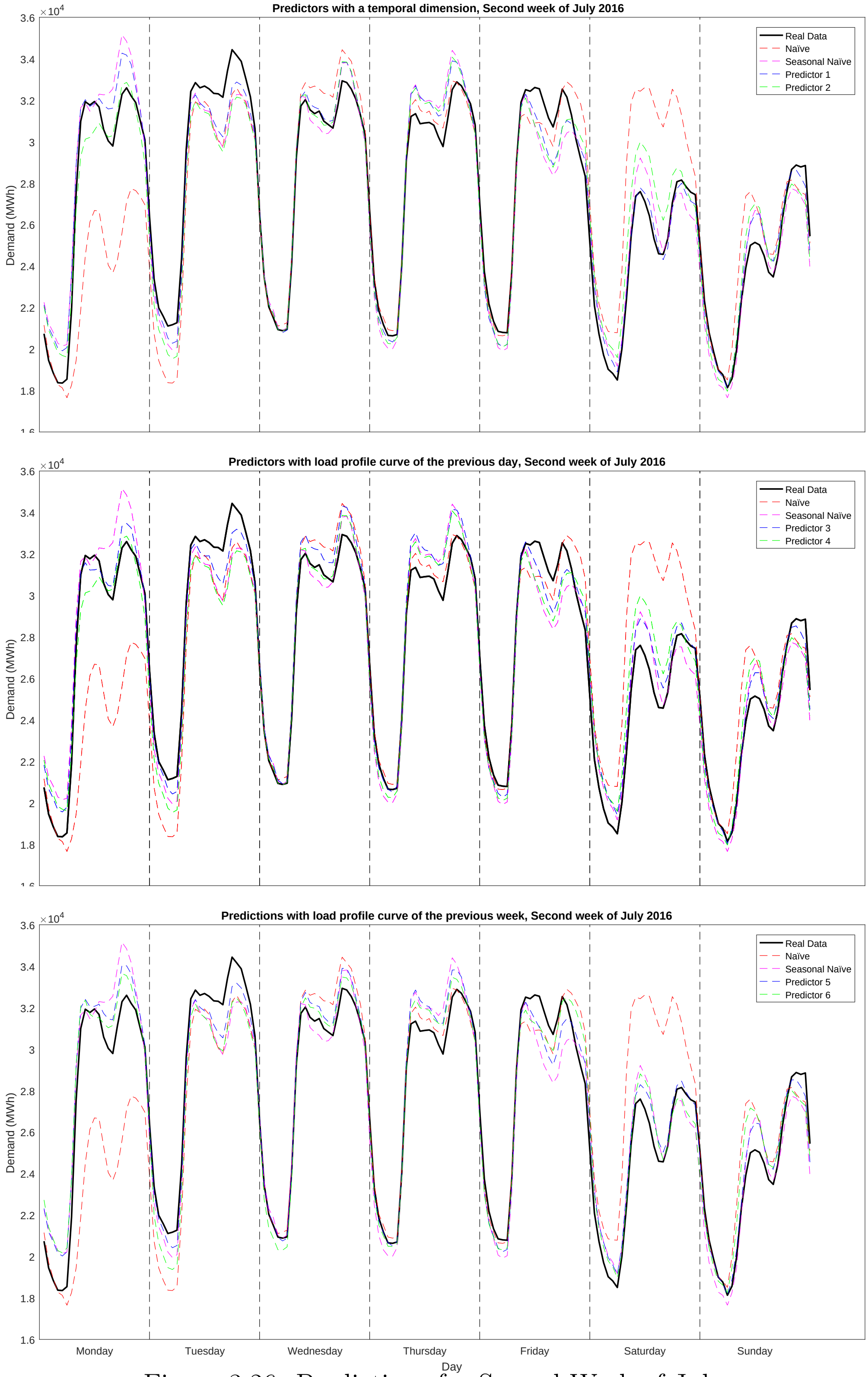


Figure 3.26: Predictions for Second Week of July

Of the different predictors shown in the previous graphs, it seems that the predictors that best fit the actual data for each of the comparisons are Predictor 1, Predictor 3 and Predictor 6, but it is not clear at first sight which of the three predictors is closer to the actual data.

To see exactly for the whole period which predictors are the ones that present the best behavior in terms of the accuracy of their predictions, a graph that shows the average errors for each of the days of the week, per predictor is shown below.

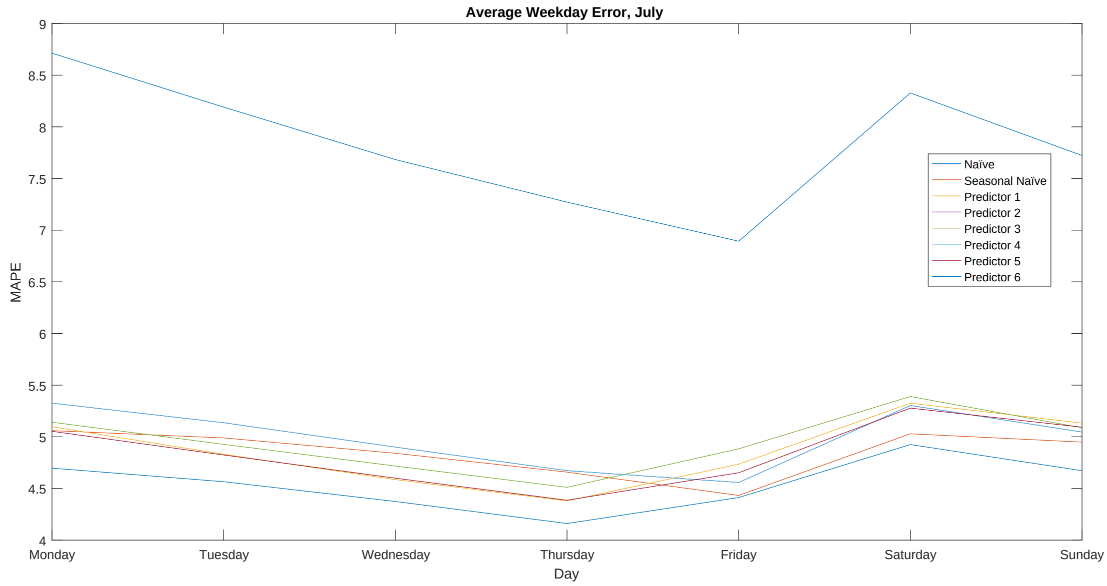


Figure 3.27: Average Errors Second Week of July

From the previous graph we can verify that, on average, the smallest errors are given for Predictor 6, followed by the predictor 5 and the Naïve Seasonal model. We can also observe that the errors for the Naïve model are disproportionately high compared to the other predictions, which could already be anticipated when visualizing the prediction graphs for the week of the second Monday of July, since the adjustment of the Naïve model was quite deficient. It is interesting to note that the range of values that make the errors, except the Naïve model, ranges between 4% and 5.5%.

To verify more accurately the mean value of the errors by predictor and day for the month of July, the following table is presented below.

	Naïve	Seasonal NaÏve	Predictor 1	Predictor 2	Predictor 3	Predictor 4	Predictor 5	Predictor 6
Monday	8.7132	5.0599	5.0976	5.3257	5.1414	5.3257	5.0534	4.6969
Tuesday	8.191	4.9894	4.831	5.1364	4.9273	5.1364	4.8241	4.5655
Wednesday	7.6833	4.8408	4.5855	4.9004	4.7171	4.9004	4.5997	4.3742
Thursday	7.2715	4.6582	4.3797	4.6726	4.5113	4.6726	4.3864	4.1608
Friday	6.893	4.4328	4.735	4.5579	4.8837	4.5579	4.6522	4.4121
Saturday	8.3279	5.0293	5.3255	5.3034	5.3902	5.3034	5.2778	4.9251
Sunday	7.7224	4.9489	5.1329	5.046	5.0876	5.046	5.0931	4.672
Average Error	7.8289	4.8513	4.8696	4.9917	4.9512	4.9917	4.841	4.5438

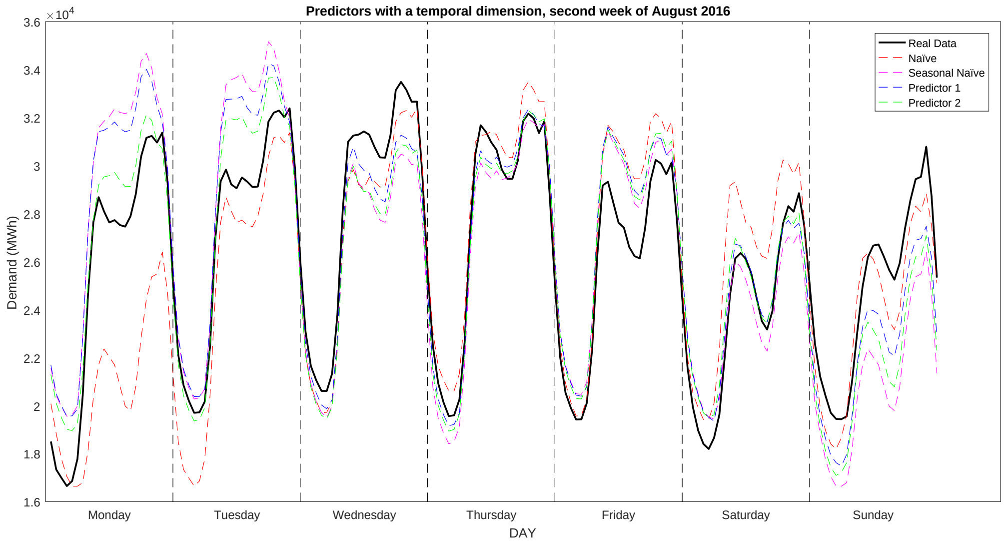
Table 3.13: Average error by Predictor, month of July

In Table 3.13, we see that the predictor with the smallest mean error for the whole month of prediction is Predictor 6, as we have already mentioned in the prediction and error graphs.

Therefore, we can conclude that for the month of July, the predictor that best adjusts the predictions to the real data is Predictor 6.

August

We proceed to analyze the second month of prediction of the training set August. In the same way that for the month of July the graph corresponding to the week of the second Monday of the month of August is shown for the



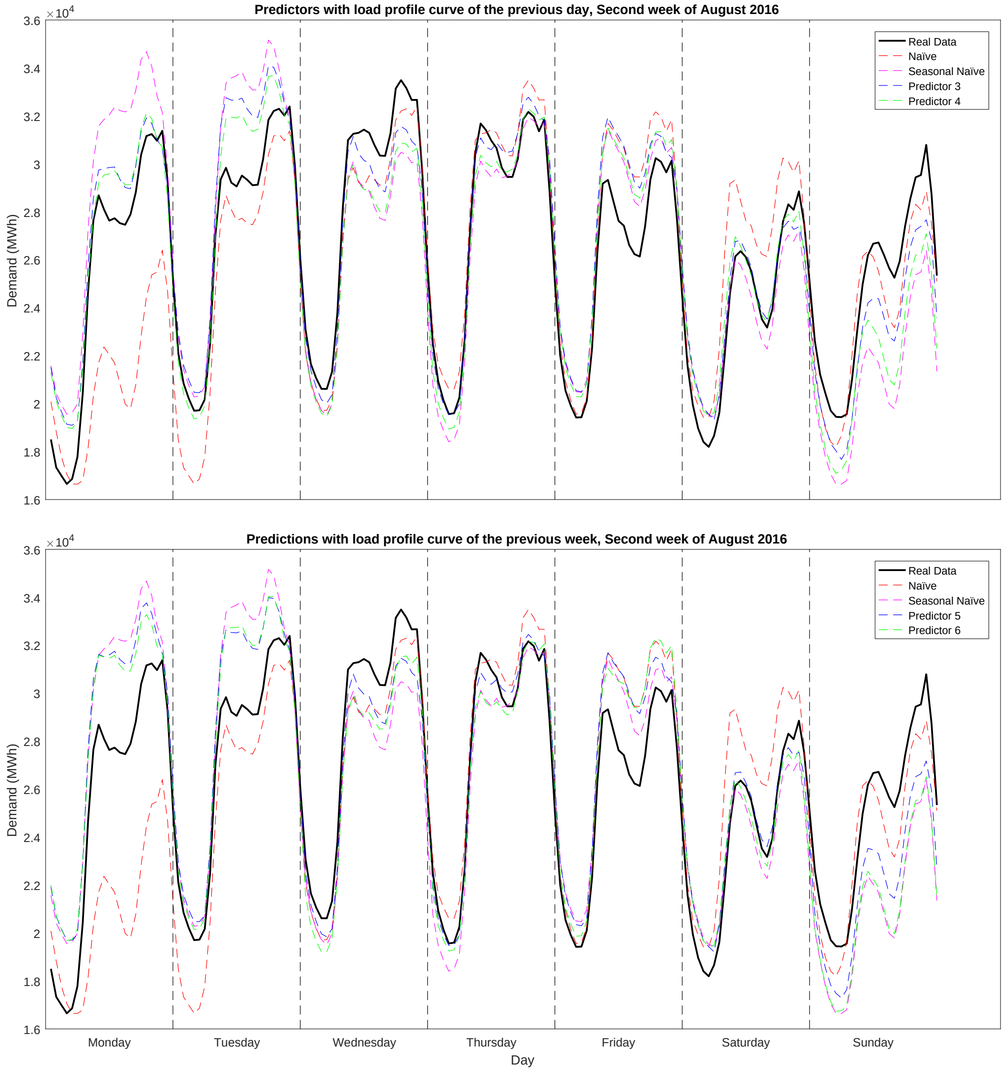


Figure 3.28: Predictions for Second Week of August

From the above graphs we can see that all the predictors from 1 to 6 present better predictions than the Naïve and Naïve Seasonal models, although it is not clear which of them are the ones that present the best fit to the real data.

In order to better appreciate the accuracy of the predictions generated by each of the predictors, a graph with average errors per day of the week of each one of the predictors considered for the month of August is shown below.

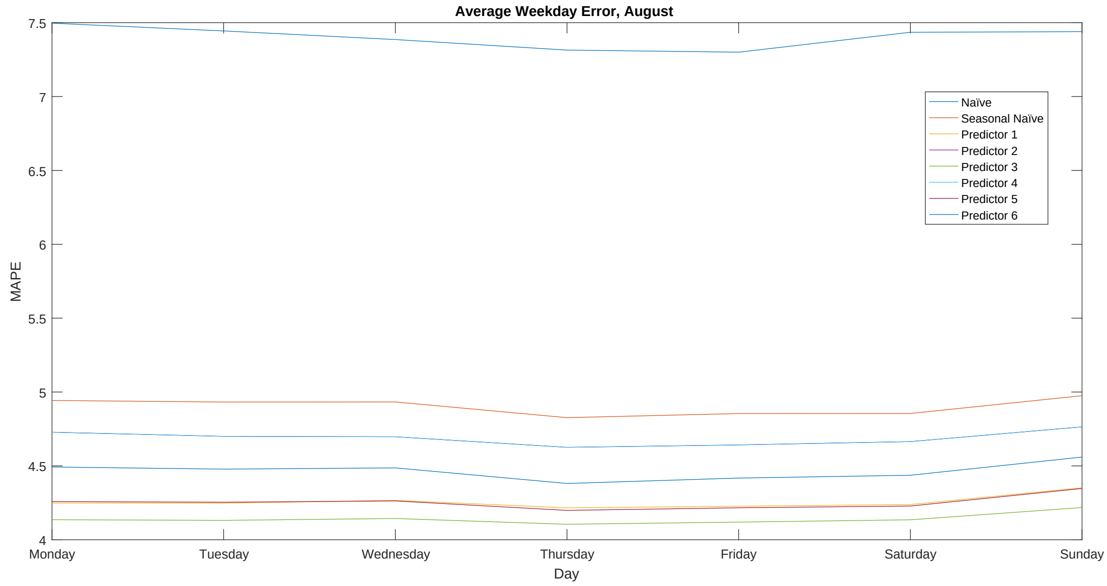


Figure 3.29: Average Errors Second Week of August

We can clearly see that the predictor that least error presents for all days of the week is Predictor 3. After Predictor 3 it is not at all clear what would be the best predictor since there are two overlapping error lines, which correspond to Predictors 5 and 1. It is interesting to note that the errors for all predictors, except for the Naïve model, are in the range of 4 to 5%.

To see the exact values of the average error per day and for the whole month of August, the corresponding table of errors is presented below.

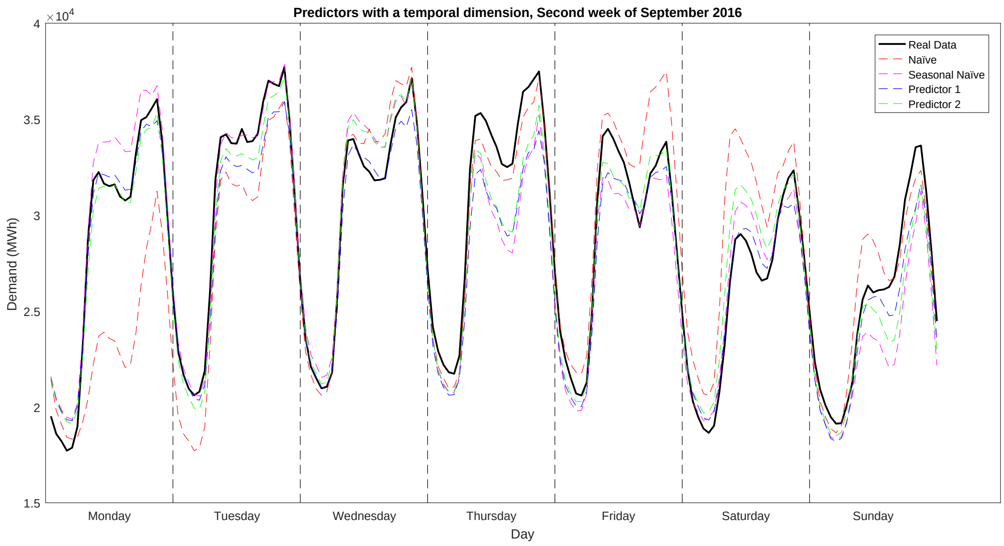
	Naïve	Seasonal Naïve	Predictor 1	Predictor 2	Predictor 3	Predictor 4	Predictor 5	Predictor 6
Monday	7.4969	4.9428	4.2487	4.728	4.1357	4.728	4.2585	4.4924
Tuesday	7.4443	4.9322	4.2477	4.6998	4.1312	4.6998	4.254	4.478
Wednesday	7.386	4.9328	4.2667	4.6972	4.1438	4.6972	4.2627	4.4864
Thursday	7.3147	4.8265	4.216	4.6256	4.1047	4.6256	4.1987	4.381
Friday	7.3005	4.8544	4.227	4.642	4.1195	4.642	4.2164	4.4176
Saturday	7.4352	4.8548	4.2385	4.6645	4.1351	4.6645	4.2282	4.4366
Sunday	7.4394	4.9751	4.3517	4.7641	4.2181	4.7641	4.347	4.5597
Average Error	7.4024	4.9027	4.2566	4.6887	4.1412	4.6887	4.2522	4.4645

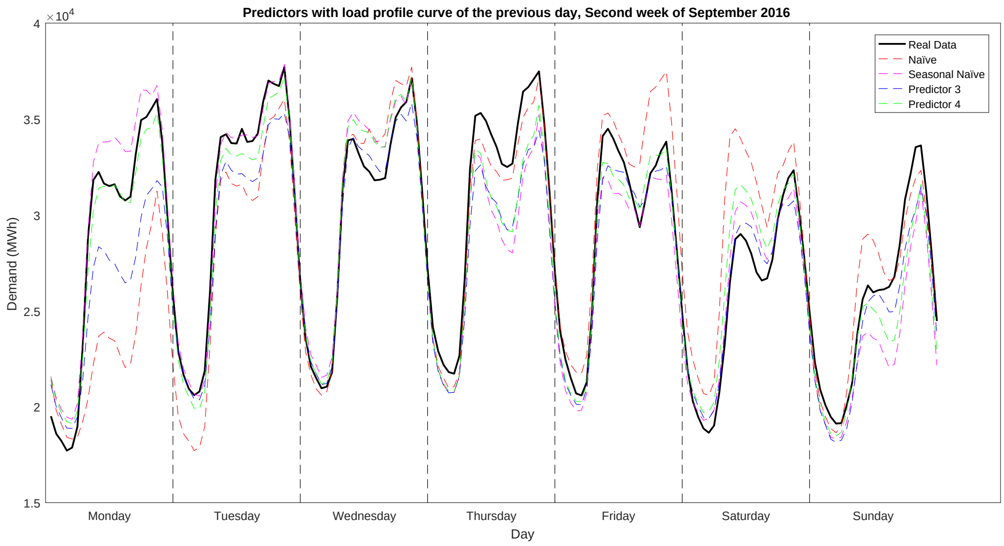
Table 3.14: Average error by Predictor, month of August

The previous table confirms that the predictor that the smallest average error presents for each of the days of the week is Predictor 3, followed by Predictor 5 and Predictor 1.

September

We now analyze the behavior of the predictors for the month of September, the third month of analysis of the training set, the graph corresponding to the week of the second Monday for the month of September is shown for the real demand data and all the predictions obtained with each predictor.





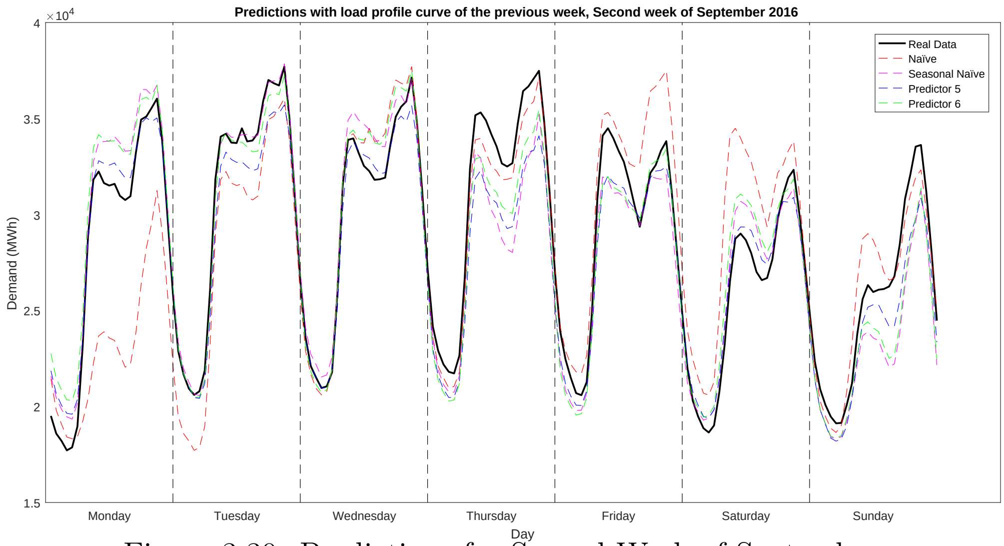


Figure 3.30: Predictions for Second Week of September

For this month we can observe in the figures that the predictions obtained with the six predictors manage to capture the behavior pattern of the real demand data. We can however also observe that our model Naïve is the one that obtains worse predictions and fails to reach a minimum distance between the real data, so we could say that as for the two previous months analyzed, for the month of September this model does not make good predictions.

To better observe this, we analyze the average errors per day of the week for each of the predictors considered, for the month of September study.

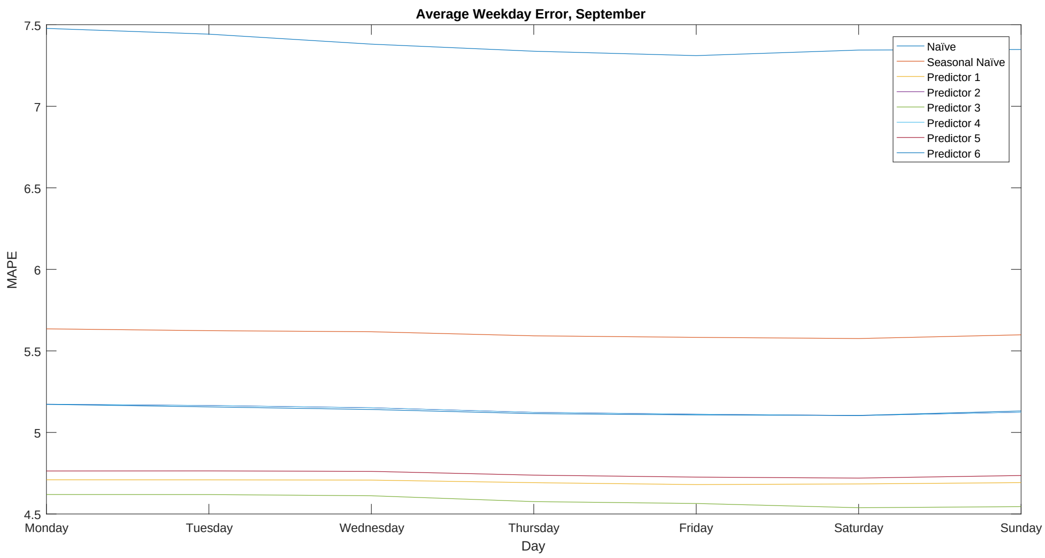


Figure 3.31: Average Errors Second Week of September

In the figure we can see that the predictor that least error presents for all the days of the week is Predictor 3. After Predictor 3 the second predictor that presents the least error for all the days of the week is Predictor 1, followed by the Predictor 5. For this particular month, the errors for all predictors, except for the Naïve model, are in the range of 4.5 to 5.7%.

Next, we will analyze the exact values of the average error per day of the week and for the whole month of Septembe. The corresponding table of errors is presented below.

	Naïve	Seasonal Naïve	Predictor 1	Predictor 2	Predictor 3	Predictor 4	Predictor 5	Predictor 6
Monday	7.4774	5.6351	4.7096	5.173	4.6193	5.173	4.7636	5.1722
Tuesday	7.4422	5.624	4.7093	5.1653	4.6189	5.1653	4.7641	5.1565
Wednesday	7.3809	5.6174	4.7077	5.1513	4.6117	5.1513	4.7612	5.1406
Thursday	7.3375	5.5924	4.6919	5.1232	4.5758	5.1232	4.7381	5.1151
Friday	7.3105	5.583	4.6802	5.1118	4.5643	5.1118	4.726	5.1074
Saturday	7.3446	5.5757	4.6842	5.1035	4.5383	5.1035	4.72	5.1043
Sunday	7.3481	5.5986	4.6922	5.1239	4.5449	5.1239	4.7358	5.1317
Average Error	7.3773	5.6037	4.6964	5.136	4.5819	5.136	4.7441	5.1325

Table 3.15: Average error by Predictor, month of September

In the table above, we can observe that the predictor that presents the smallest error in the predictions is Predictor 3. As for the month of August this is the Predictor that presents a better adjustment of prediction in relation to the actual data, followed by Predictor 1 and Predictor 5.

October

Then we proceed to analyze our fourth study month October. In the same way that for the previous 3 months we analyze the prediction graph for this month. The graph shown corresponds to the week of the second Monday of October for the actual demand data and predictions obtained with each of the predictors

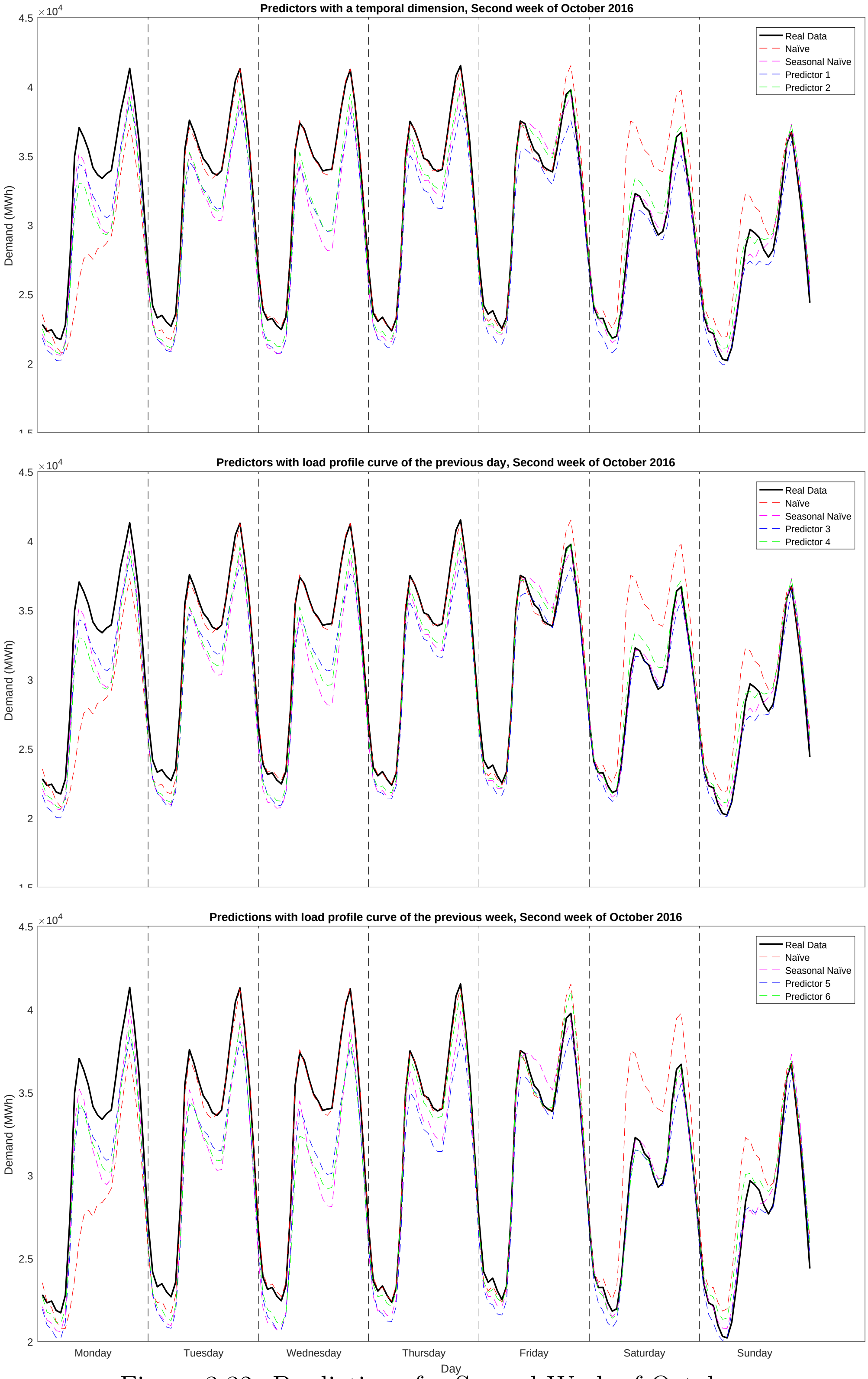


Figure 3.32: Predictions for Second Week of October

An interesting aspect of the previous graphs is that the shape of the daily demand curves changes slightly with respect to the previous months. In particular, they have a less rounded shape with more pronounced peaks. In spite of this, we see that the predictions seem to capture this change in the data.

As for all previous months, we can see that all predictors from 1 to 6 present better predictions than Naïve and Seasonal Naïve models, although it is not clear which of them are the ones that present the best fit to the real data.

To better appreciate the accuracy of the predictions generated by each of the predictors, we will present a graph with the average errors per day of the week of each of the predictors considered, for the month of October.

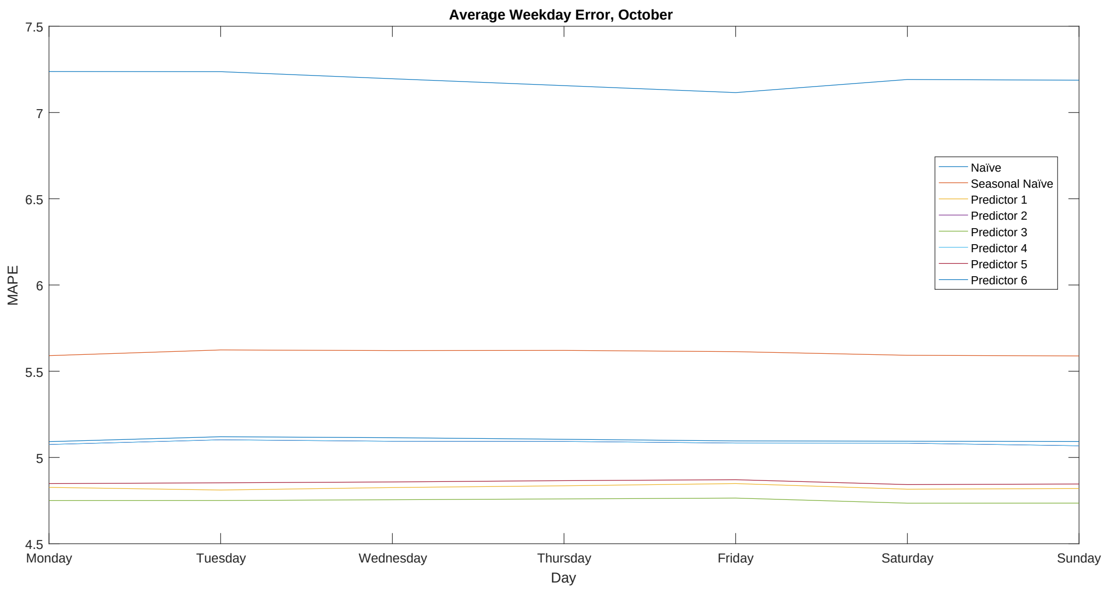


Figure 3.33: Average Errors Second Week of October

We can see that the predictor that presents the lowest average error per day of the week for the month of October is Predictor 3, followed by Predictor 1 and Predictor 5 respectively. For this month the errors for all predictors, except for Naïve model, are in the range of 4.7 to 5.6%.

For comparison purposes, and in order to be able to have a more accurate result as far as the average error per day is concerned and for the whole month of October, we proceed to analyze the table of errors by predictor for our study series of electrical energy demand.

	Naïve	Seasonal Naïve	Predictor 1	Predictor 2	Predictor 3	Predictor 4	Predictor 5	Predictor 6
Monday	7.2376	5.5904	4.8267	5.0754	4.7502	5.0754	4.8485	5.0918
Tuesday	7.2367	5.6234	4.8113	5.1025	4.7505	5.1025	4.8532	5.1203
Wednesday	7.1956	5.6198	4.8257	5.094	4.7551	5.094	4.8581	5.1146
Thursday	7.1558	5.6208	4.8359	5.093	4.7598	5.093	4.8658	5.1055
Friday	7.1154	5.6135	4.8487	5.0835	4.7646	5.0835	4.8711	5.096
Saturday	7.1913	5.5925	4.8162	5.0829	4.7351	5.0829	4.8429	5.0945
Sunday	7.1873	5.5886	4.82	5.0678	4.7354	5.0678	4.8462	5.0924
Average Error	7.1885	5.607	4.8264	5.0856	4.7501	5.0856	4.8551	5.1022

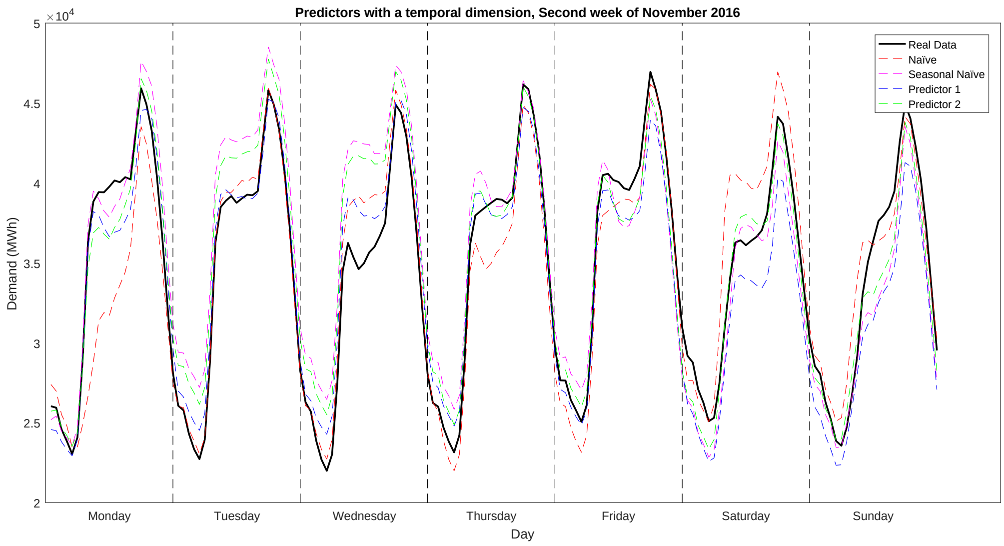
Table 3.16: Average error by Predictor, month of October

The above table confirms that the best predictor for the month of October is Predictor 3, as it happens from the month of August. This predictor is the one with the lowest prediction error, followed by Predictor 1 and Predictor 5.

Therefore, we consider that the best predictor for the month of October is Predictor 3 once again.

November

Then we proceed to analyze the fifth month of study of the training set, November. In the same way that for the previous months we analyzed the



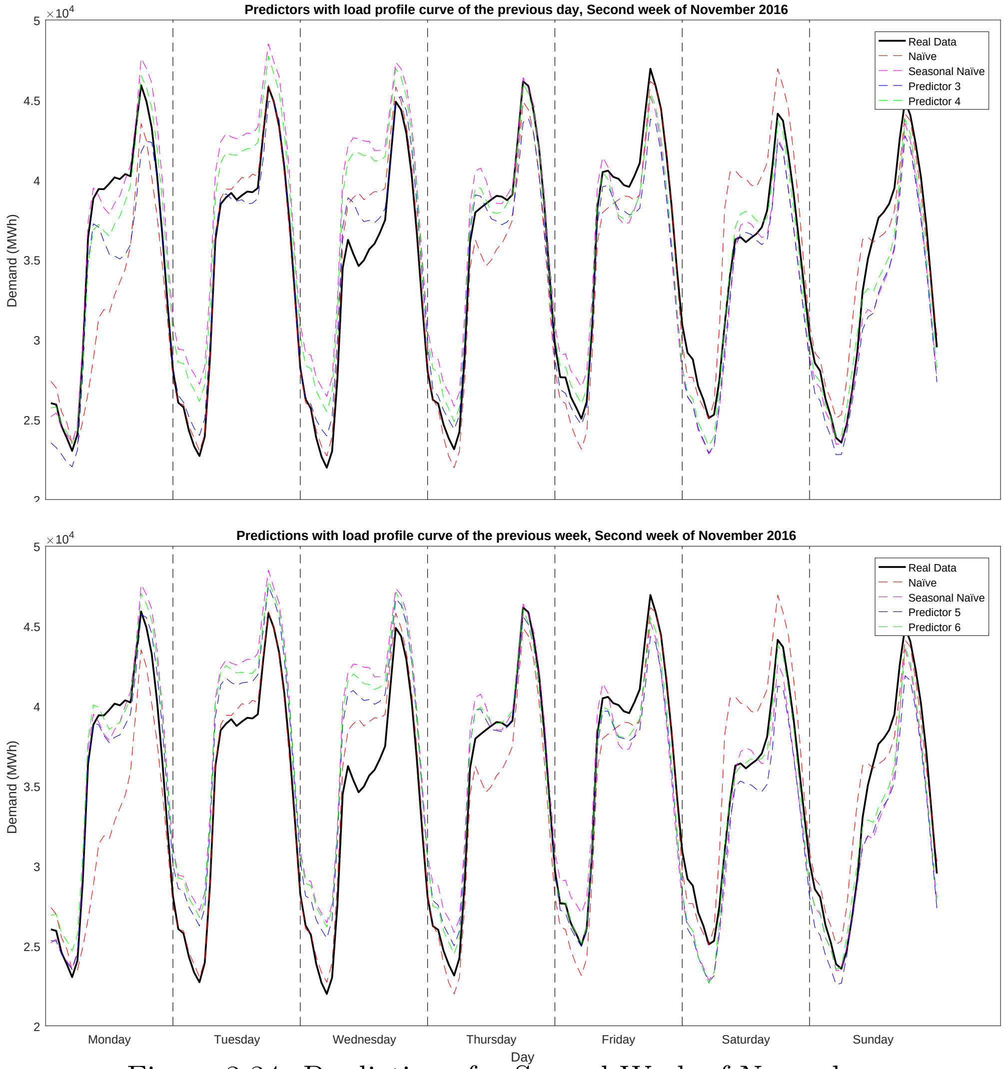


Figure 3.34: Predictions for Second Week of November

For this month we can observe that the shape of the curves of daily demand again suffer a new change. Now they have a more pointed shape and also have only one peak instead of two, which corresponds to the last hours of each day of the week. Since the change in the shape of the curves is rather abrupt, we see in the graphs that the predictors capture this change quite well, but they do not fit as well as in previous months. This will have its repercussion in the average daily error, since it will lead to greater errors.

We see that all the developed predictors succeed in overcoming the fit of the predictions generated by the Naïve models. To see in more detail this aspect, we show the graph of average errors for each of the days of the week, for each predictor.

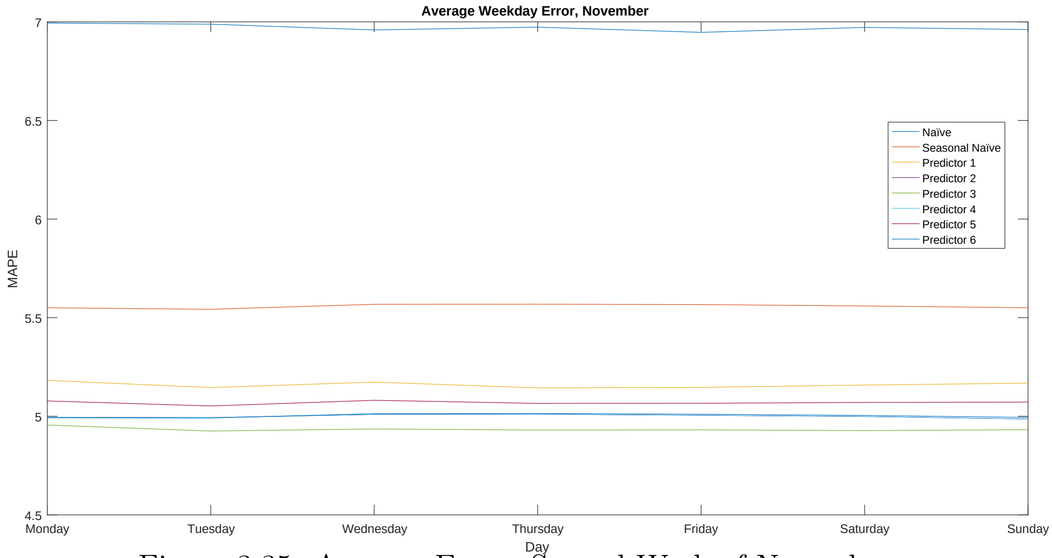


Figure 3.35: Average Errors Second Week of November

We can see that the predictor with the smallest mean error of prediction is Predictor 3. Secondly, there are lines of error that overlap and we can not see exactly which predictors are the ones with the smallest errors after Predictor 3. Therefore, The average daily error for the month of November is shown in detail in the following table.

	Naïve	Seasonal Naïve	Predictor 1	Predictor 2	Predictor 3	Predictor 4	Predictor 5	Predictor 6
Monday	6.9937	5.5501	5.1818	4.9951	4.9554	4.9951	5.0777	4.9922
Tuesday	6.9883	5.5424	5.1455	4.9931	4.9251	4.9931	5.0521	4.9912
Wednesday	6.9589	5.5677	5.1731	5.0091	4.9355	5.0091	5.081	5.013
Thursday	6.9733	5.5683	5.144	5.0104	4.93	5.0104	5.0652	5.0144
Friday	6.9467	5.5663	5.1465	5.0046	4.9311	5.0046	5.0659	5.0103
Saturday	6.9719	5.5593	5.1582	4.9985	4.9271	4.9985	5.07	5.0042
Sunday	6.9611	5.5502	5.168	4.9859	4.932	4.9859	5.072	4.994
Average Error	6.9705	5.5578	5.1596	4.9995	4.9337	4.9995	5.0691	5.0028

Table 3.17: Average error by Predictor, month of November

In the previous table we can see that Predictor 3 is the one with the smallest average error for the whole month. After Predictor 3, both Predictor 2 and 4 are the ones with the lowest error and correspond to the lines of error that overlapped in the graph of average daily errors. It is interesting to note that the average errors for Predictor 2 and Predictor 4 are exactly the same for each of the days of the week.

December

Now we go on to analyze the last month of our prediction period, December.

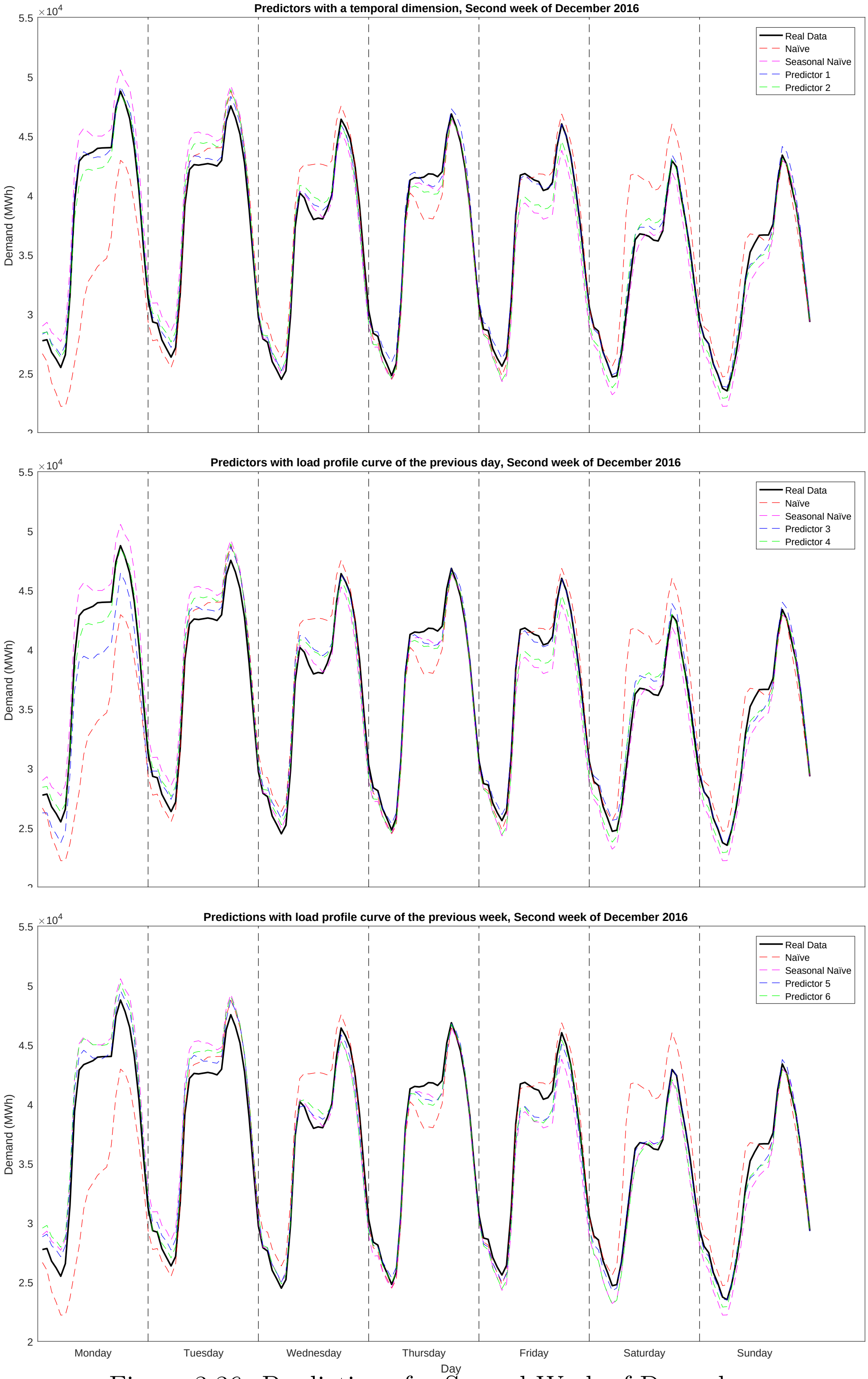


Figure 3.36: Predictions for Second Week of December

We can see in the graphs that the shape of daily demand curves presents a behavior similar to that of November, although now the daily behavior is somewhat more irregular, so that the predictors fail to capture the behavior pattern of the real data as well as in previous months. This fact will be reflected in the errors of predictions.

As with all previous months, the predictors that have been developed succeed in adjusting predictions to actual data better than Naïve models.

Then, to see exactly which predictors have the best settings, the graph corresponding to the average daily errors for the last forecast month, December,

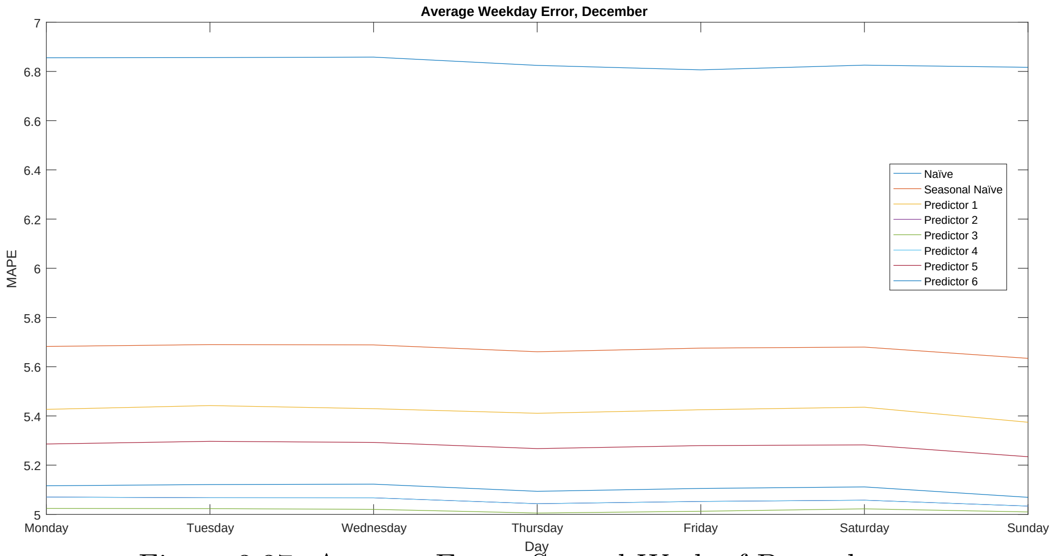


Figure 3.37: Average Errors Second Week of December

We can see in the above graph that Predictor 3 is the smallest prediction error presented. We can also appreciate that for this month there are two predictors that overlap, as it did in the month of November. In addition, it is interesting to note that the range of errors is higher than for previous months, taking values ranging from 5% to 5.7%, except for the Naïve model, which we have seen that for all months it presented much greater errors than the other predictors.

To better see the value of the average errors for each of the days of the week, for each one of the predictors, the table of average errors is shown next.

We can see that the predictor with the smallest error is again Predictor 3, as it happens in all previous months except for the first month, July. We again

	Naïve	Seasonal Naïve	Predictor 1	Predictor 2	Predictor 3	Predictor 4	Predictor 5	Predictor 6
Monday	6.8556	5.6824	5.4267	5.0703	5.0242	5.0703	5.2859	5.1161
Tuesday	6.8563	5.6899	5.4421	5.0675	5.0232	5.0675	5.2968	5.1211
Wednesday	6.8582	5.6886	5.4294	5.0672	5.0203	5.0672	5.2923	5.1227
Thursday	6.8246	5.6608	5.4107	5.0433	5.0053	5.0433	5.2672	5.0936
Friday	6.8064	5.6756	5.4251	5.0525	5.0125	5.0525	5.2794	5.1053
Saturday	6.8256	5.6798	5.4355	5.0576	5.0224	5.0576	5.2823	5.1115
Sunday	6.8168	5.634	5.3743	5.0336	5.0098	5.0336	5.2342	5.0692
Average Error	6.8348	5.673	5.4205	5.056	5.0168	5.056	5.2769	5.1056

Table 3.18: Average error by Predictor, month of December

see exactly the same behavior between Predictor 2 and Predictor 4, which have the second lowest error after Predictor 3.

Once a thorough analysis has been performed for each month of the training set prediction period, we can state that, in average terms, the predictor with the smallest errors presents, and therefore, which better predictions generates in terms of fit with respect to the real data is Predictor 3. This predictor is the best in all months except in the first month, but on average it is the predictor that least error presents by far. To better illustrate this fact, the graph of daily errors for the entire forecast period from July to December and the table of total mean errors for that period are shown below.

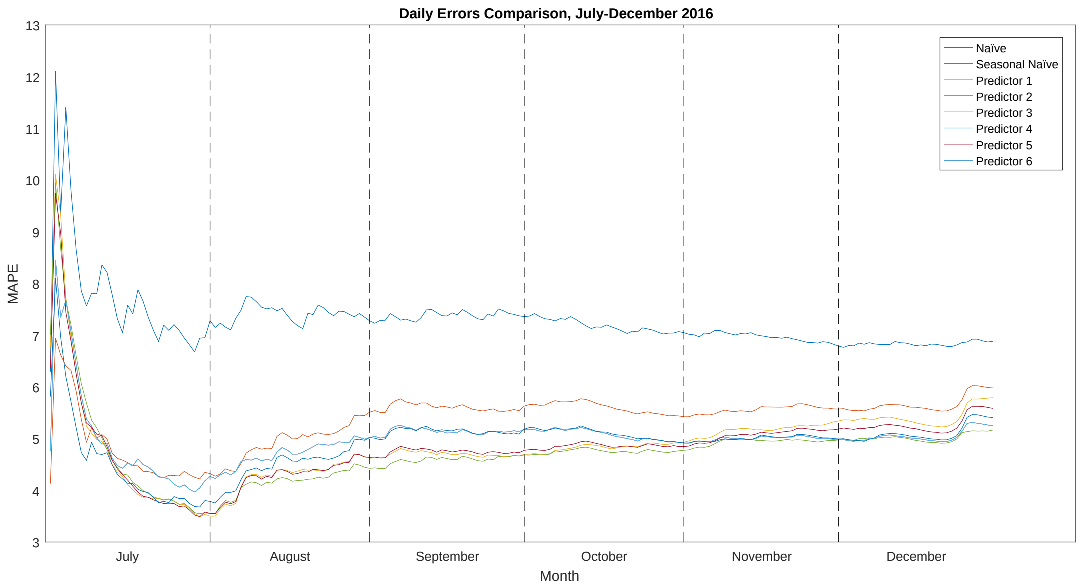


Figure 3.38: Average Errors Comparison, July-December 2016

	Naïve	Seasonal Naïve	Predictor 1	Predictor 2	Predictor 3	Predictor 4	Predictor 5	Predictor 6
Average Error	7.2671	5.3659	4.8715	4.9929	4.7292	4.9929	4.8397	4.8919

Table 3.19: Average Total Error July-December 2016

As we have discussed above and as can be seen in the average total error table for the entire forecast period, our best predictor for electric power demand is **Predictor 3**.

Recall that Predictor 3 had the following structure:

- **Temporal dimension**: it includes binary day, decimal month and year when looking for similarities between the data to be predicted and the other data of the sample.

- **Functional dimension**: it considers the load curve of the previous day for the day to be predicted and each one of the load curves associated to all the other data of the sample that are compared.

Specificaly,

$$x_k = [binary\ day(x_k), decimal\ month(x_k), year(x_k), load\ profile\ curve(x_{k-1})]$$

$$x_i = [binary\ day(x_i), decimal\ month(x_i), year(x_i), load\ profile\ curve(x_i)]$$

$$x_k, x_i \in \mathbb{R}^{33}$$

$$\text{where:}\quad binary\ day \in \mathbb{R}^7,\ decimal\ month \in \mathbb{R}\ ,\ year \in \mathbb{R} \text{ and}$$

$$load\ profile\ curve \in \mathbb{R}^{24}$$

And the distance function had the following structure:

$$d(x_i, x_k) = \|p_1 \cdot (binary\ day(x_i) - binary\ day(x_k)) + p_2 \cdot (decimal\ month(x_i) - decimal\ month(x_k)) + (year(x_i) - year(x_k)) + p_3 \cdot \frac{1}{24} \sum_{h=1}^{24} (load\ curve_h(x_i) - load\ curve_h(x_{k-1}))\|_2$$

	p_1	p_2	p_3	γ	$Error$
Predictor 3	1.6	4.5	1	0.4	4.73

Table 3.20: Optimal values Predictor 3

As for the optimal values of the hyperparameters γ, p_1, p_2 and p_3, we recall that they took the following values:

Once we have obtained a valid prediction model for the original research problem, the next step is to validate this predictor with data from the validation set corresponding to the first 6 months of the year 2017. For this, we will make predictions of electric power demand for the 24 hours of each day of the months from January to June of the year 2017.

In order to predict the 2017 data, 2016 data will be used, using the the optimal hyperparameters previously obtained for the distance function. For the cleaning and transformation process of the data in order to make predictions, the same procedure has been used as for the predictions of 2016, as detailed in the corresponding section.

3.7 Predictions for 2017

To finalize the practical development of the present study and apply the best prediction model obtained for the training data, covering the period from July to December 2016, we will perform a validation process of Predictor 3 with the optimal hyperparameters obtained in the training set.

We proceed to show the predictions obtained with the aforementioned predictor for the validation set that covers the months from January to June 2017.

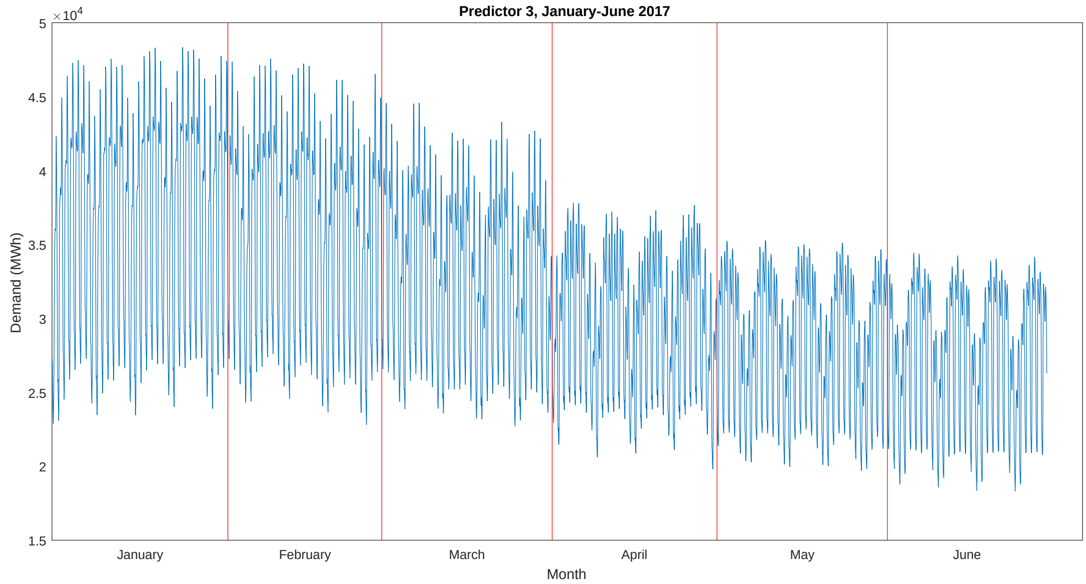


Figure 3.39: Predictor 3 predictions, January-June 2017

We can observe from Figure 3.39 that the predicted data present a decreasing trend in the months from January to May, where it stabilize and begin to show a constant trend until the June. Let's see if this trend is also reflected in the actual data.

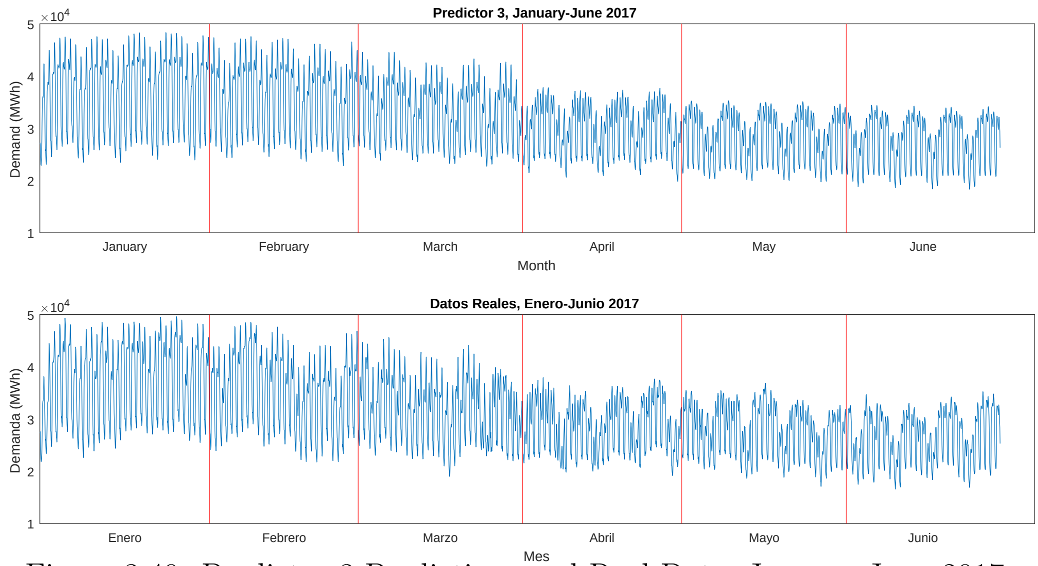


Figure 3.40: Predictor 3 Predictions and Real Data, January-June 2017

In fact, the decreasing trend can also be seen in the real data in the months of January to May, stabilizing until the month of June. At first sight it seems that Predictor 3 captures quite well the pattern of behavior of the actual data.

To see how good the Predictor 3 settings are compared to the actual data, a graph of daily errors is presented below.

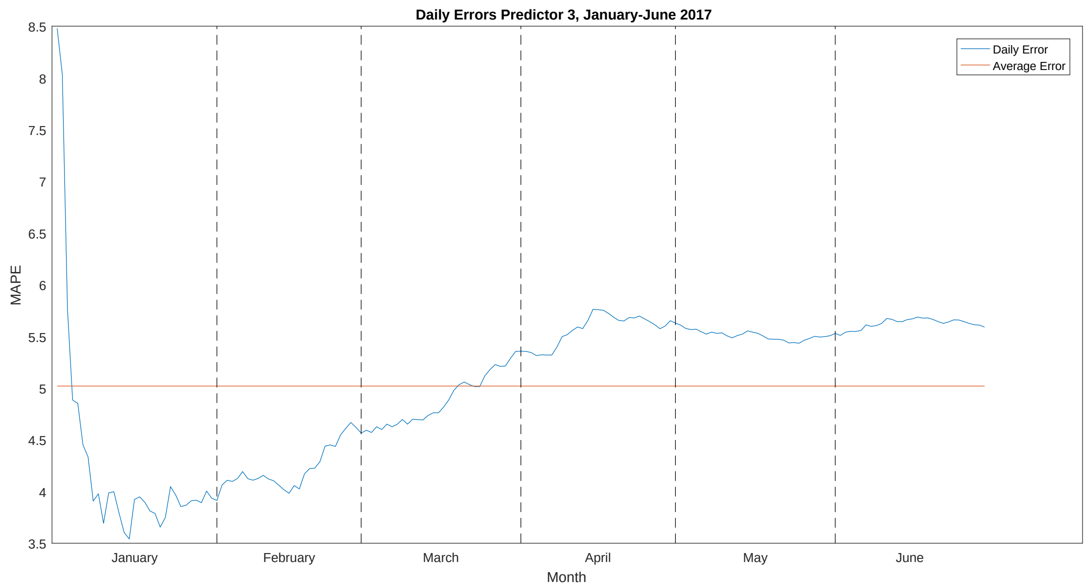


Figure 3.41: Daily Errors for Predictor 3, January-June 2017

As shown in Figure 3.41 the smallest errors occur for the first three months, always being below 5% error. As of the end of March, errors exceed the threshold of 5% but stabilize around 5.5% until the end of the validation period. On average, the error is very close to 5%.

As we have seen, we can see that for the months of January, February and March the prediction errors are relatively low, being in all cases below 5%. The other months have slightly higher errors, but they average around 5.5%, as we mentioned earlier. On average for the entire validation period for our predictor we have a total error of 5.01%, which is quite acceptable.

To make sure the correct behavior of the predictor in question, we will see it in relation to the performance of the Naïve models. For this purpose, the

	January	February	March	April	May	June
Monday	4.7972	4.2871	4.9013	5.5775	5.5364	5.6339
Tuesday	4.2621	4.2629	4.9079	5.5665	5.523	5.6225
Wednesday	4.1559	4.1115	4.845	5.5605	5.5176	5.6275
Thursday	4.1326	4.1746	4.871	5.5406	5.5027	5.6047
Friday	3.986	4.1817	4.882	5.5496	5.4985	5.6031
Saturday	3.8776	4.2345	4.8325	5.5601	5.5017	5.6283
Sunday	4.7141	4.2657	4.8633	5.5886	5.5045	5.6395
Average Error	4.2751	4.2169	4.8719	5.5634	5.512	5.6228
Total Average Error	5.01035					

Table 3.21: Average Errors January-June 2017, Predictor 3

graph that illustrates the average monthly errors for the prediction period of the validation set, for the predictor 3, the Naïve Simple model and the Naïve Seasonal model, is shown below.

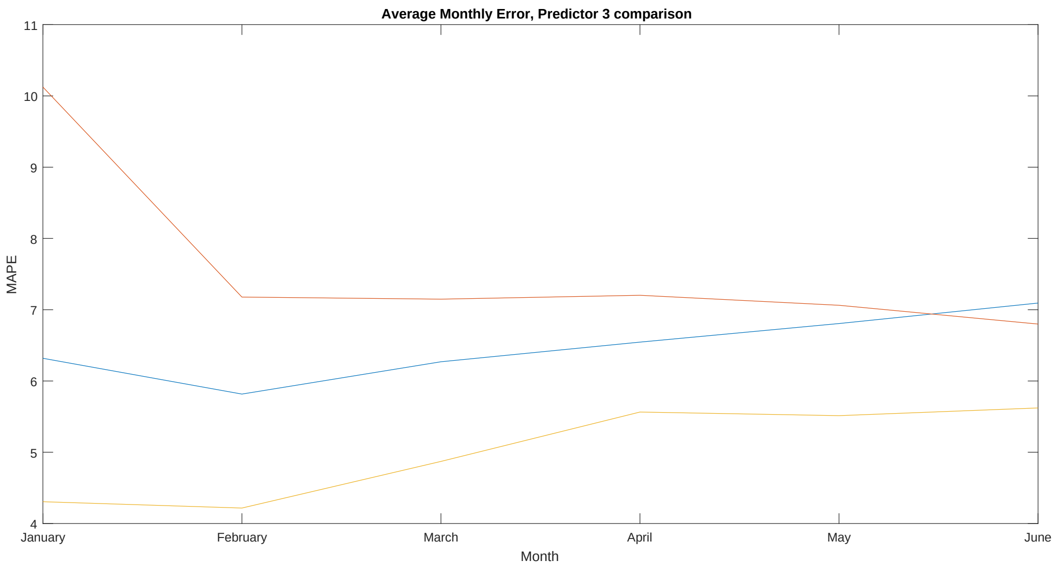


Figure 3.42: Average Monthly Errors, Predictor 3 Comparison

We can see that Predictor 3 is the one with the lowest prediction error for all months. In fact, the difference between Predictor 3 and the next predictor with the smallest error, which is the Naïve model, is quite broad reaching a difference of about 2% at the beginning of the prediction period, as it seems to be seen in the graph. To see in detail the exact average error values for each of the months, the table below shows the average monthly and total error for the entire forecast period, taking into account the two Naïve models and our Predictor 3.

	Naïve	Seasonal Naïve	Predictor 3
January	6.3040	10.1070	4.2751
February	5.8162	7.1774	4.2169
March	6.2733	7.1460	4.8719
April	6.5448	7.2032	5.5634
May	6.8034	7.0628	5.5120
June	7.0959	6.7983	5.6228
Total Period	6.4729	7.5825	5.0104

Table 3.22: Average Monthly Error January-June 2017

As we can see in the table and it was already noticed in the previous graph, Predictor 3 is the one with the smaller prediction error for the validation period. In fact, as we have seen, in the first two months there is a difference of 2 percentage points with respect to the Naïve model, which supposes a great decrease of the prediction error. In generic terms for the whole period, we see that Predictor 3 improves the Naïve model by almost 1.5% and the Seasonal Naïve model by approximately 2.5%.

We conclude therefore that Predictor 3 is a solid predictor, since with the optimum value for the hyperparameters obtained from the training set, we see that it performs well in the validation set, going from an average error of 4.73% in the training set an average error of 5.01% in the validation set, which represents a difference of only 0.28%.

Chapter 4. Final Conclusions

The main objective of this research was to obtain a predictor that would generate good enough predictions of electric power demand so that they could be able to be implemented in a real setting of the energy industry. To carry out this research 6 different predictors were developed, two of them composed of elements of time dimension and the remaining 4 composed by elements of both time dimension and functional dimension, which were compared with the so-called Naïve models, as reference for the evaluation of their performance.

For the generation of predictions, we first defined the components of each of the predictors in detail, and then proceed to obtain the optimal values of the hyperparameters for each predictor in the training set, which in our case covered the months of July to December of the year 2016. In particular, all the data of 2016 were used to generate, gradually, daily predictions from the month of July until December of this same year. Once the optimal values for each of the hyperparameters for each predictor were obtained, we compared the goodness of fit of every predictor in respect to the real data. For this comparison between each of the predictors and the actual data a mean percentage error measure (MAPE), measured for each day of the week and afterwards, was used for the whole period.

Among all predictors, we saw that the predictor that incorporated the binary day, the decimal month and the year as elements of the time dimension; and the load curve of the previous day as element of the functional dimension, which we had called as Predictor 3, is the one that gets the best predictions.

To be sure of the validity and proper functioning of this predictor, predictions were made in a validation set that comprised the first 6 months of 2017. The result obtained for this validation set confirms that the predictor in question is apt to be able to be used to make future predictions of the electric power demand, so we could say that we satisfied the main objective of this research.

As future work is proposed to extend the present work to predictors based on non-parametric polynomial regression and comparison with other techniques such as Support Vector Machines or Neural Networks.

Bibliography

[1] Michael Smith. Modeling and short-term forecasting of new south wales electricity system load. *Journal of Business & Economic Statistics*, 18(4):465–478, 2000.

[2] James W Taylor, Lilian M De Menezes, and Patrick E McSharry. A comparison of univariate methods for forecasting electricity demand up to a day ahead. *International Journal of Forecasting*, 22(1):1–16, 2006.

[3] Ramu Ramanathan, Robert Engle, Clive WJ Granger, Farshid Vahid-Araghi, and Casey Brace. Short-run forecasts of electricity loads and peaks. *International journal of forecasting*, 13(2):161–174, 1997.

[4] Remy Cottet and Michael Smith. Bayesian modeling and forecasting of intraday electricity load. *Journal of the American Statistical Association*, 98(464):839–849, 2003.

[5] Angel Pardo, Vicente Meneu, and Enric Valor. Temperature and seasonality influences on spanish electricity load. *Energy Economics*, 24(1):55–70, 2002.

[6] Ofgem. *European markets*. [Online]. Available from: `https://www.ofgem.gov.uk/`. [Accessed 25th August 2017].

[7] National Grid plc. *About us*. [Online]. Available from: `http://www2.nationalgrid.com/About-us/`. [Accessed 15th August 2017].

[8] Manuel Zaforas. *Machine Learning, la forma de hacer Big Data inteligente*. [Online]. Available from: `https://www.paradigmadigital.com/dev/machine-learning-la-forma-big-data-inteligente/`. [Accessed 20th August 2017].

[9] Teresa Zamora García. *Métodos de regresión no paramétrica en muestreo en poblaciones finitas.* [Online]. Available from: `http://masteres.ugr.es/moea/pages/tfm0910/metodosderegresionnoparametricaenmuestreoenpoblacionesfinitas/!` [Accessed 23th August 2017].

[10] Elizbar A Nadaraya. On estimating regression. *Theory of Probability & Its Applications*, 9(1):141–142, 1964.

[11] Geoffrey S Watson. Smooth regression analysis. *Sankhyā: The Indian Journal of Statistics, Series A*, pages 359–372, 1964.