Clustering of Load Curves to Support Demand and Generation Forecast

Ana Catarina Constantino Vaz
a.catarina.c.vaz@tecnico.ulisboa.pt

Instituto Superior Técnico, Lisboa, Portugal

September 2017

Abstract

Each customer of an electricity company demands a certain amount of electricity at every instant. So, this amount needs to be generated in order to suit the customer's demands. In order to optimize the use of energy generation resources, it is necessary to know the demanded quantity in advance. Hence, this study is aimed at fitting models that explain customer's electricity consumption and that enable the forecast of one day ahead consumption. However, for a large customer portfolio, it becomes impractical to analyse each consumption individually. Therefore, we make use of clustering methods to group customers by similarity of consumption and we propose only one model that represents the consumption for each cluster. Two types of models will be compared based on their forecast accuracy: generalized additive models and autoregressive integrated moving average models.
Keywords: Load Forecasting, Time Series, Hierarchical Clustering, Generalized Additive Models

1. Introduction

In large scale and when the load generation is greater than the demand, there are still major technical challenges that make it impractical to store the excess of electricity. So, in order to avoid losses and to optimize the use of energy generation resources, it is important to reduce the mismatch between procured generation and demand. Having this in mind, given data that represents one customer's electricity consumption from previous years until today, our aim is to build a statistical model that best explains the data and to forecast future values of electricity consumption accurately.

On one hand, we will study models that only consider historical data of electricity consumption. On the other hand, we will study models that include as well external variables. In this latter case, understanding the underlying variables that motivate different electricity consumption behaviours might be essential in forecasting consumption. In particular, we will understand the influence of weather variables in electricity consumption, in order to introduce these in the model.

Now consider that we have millions of customer's consumption to forecast. It becomes impractical to analyse the demand of each customer individually. Finding groups of customers with similar consumption behaviours allows us to reduce the number of behaviours that need to be analysed. This grouping procedure, known as clustering, enable us to create only one statistical model for each group. In order to do the clustering, it is necessary to state what do we define by similarity of consumption behaviours and which are the best techniques that perform the grouping.

Finally, having obtained the right customer's partition allied with the best model that explains each cluster's data, we are able to foresee consumer behaviours in the near future.

2. State-of-the-Art

Load forecasting of a large set of different customers requires the realization of two main steps. The first step consists in the clustering of customer load profiles, whereas the second one consists in fitting a model that explains the consumption of each cluster and forecasting short term electricity load.

For the clustering procedure, Chicco [2012] proposed a pre-clustering phase, characterized by setting up a normalised representative load pattern (RLP), in order to get load patterns comparable in terms of their shape. The RLP was computed by dividing the typical daily load pattern by its reference power, where the reference power is defined as the peak value of the typical daily load pattern. The usual clustering methods were then applied to these RLP's.

For the modelling and forecasting short term electricity load, the second main step mentioned in the first paragraph of this section, Huang and Shih

[2003] proposed an ARMA model, whereas Taylor [2012] proposed the use of an exponential smoothing formulation. These two modelling approaches use only historical load data, so they are called univariate models. The following approach use models that explain electricity consumption based on different variables: Yannig Goude and Sinn [2012] proposed a weather-based generalized additive models (GAM) and evaluated the methodology on 5 years of electricity load data. The results showed that GAM outperforms the other state-of-the-art methods in terms of model tracking and prediction accuracy.

There is a continuing effort to improve the accuracy of the consumption forecasts. In this study, we begin by normalizing the load curves in order to do the clustering, based on Chicco [2012], with a modified formula in the way of normalizing though, and after that, we present a GAM for each cluster, based on Yannig Goude and Sinn [2012].

3. Background

Time series occur in a variety of fields, in particular, during this study, we encounter them in the electrical and also meteorological fields. In Subsection 3.1, a formal definition of time series and related concepts are presented.

Our objective is to identify a model that best fits observations of a given time series and to predict its future values. Having this is mind, we compare the performance of two kinds of models. On one hand, we only consider historical data by fitting time series models, which are explained in Subsection 3.2. Alternatively, in Subsection 3.3, we introduce generalized additive models, which allow the existence of external variables.

Since we are dealing with a large set of time series, it is impractical to identify models for each one of them. This requires that we group time series according to their similarity. Thus, a clustering method must be used. Subsection 3.4 describes clustering techniques used in time series.

3.1. Fundamental Concepts in Time Series

In order to properly define time series, we begin by defining stochastic processes, according to Zitkovic (2010).

Definition 3.1. Let $\mathcal{T}$ be a subset of $[0, \infty)$. A family of random variables $\{X_t : t \in \mathcal{T}\}$, indexed by $\mathcal{T}$, is called a stochastic (or random) process. When $\mathcal{T} = \mathbb{Z}$, $\{X_t : t \in \mathcal{T}\}$ is said to be a discrete-time process, and when $\mathcal{T} = [0, \infty)$, it is called a continuous-time process.

Thus, a time series is a sample function, or realization, from a certain stochastic process. In this study, we will deal with discrete-time processes. According to Brockwell and Davis (2002), a time series $\{X_t : t \in \mathbb{Z}\}$ may be decomposed as:

$$X_t = m_t + s_t + \epsilon_t, t \in \mathbb{Z} \tag{1}$$

which is called the classical decomposition model, and where m_t is a slowly changing function known as a trend component, s_t is a function with known period referred to as a seasonal component, and ϵ_t is a random noise component.

In general, for $h \in \mathbb{N}$, ϵ_t and ϵ_{t+h} are dependent, since, in practice, future events are influenced by previous events. So, it is necessary to introduce the concept of stationary. Loosely speaking, we say that a time series is stationary if it has statistical properties similar to those of the time-shifted series. The concept of stationarity leads us to the definition of autocovariance function (ACVF) of $\{X_t\}$ at lag h, which is given by

$$\gamma_h = Cov(X_t, X_{t+h}), h \in \mathbb{Z}. \tag{2}$$

In addition, since ACVF is sensitive to the units in which the observations are measured, we define the autocorrelation function (ACF) of $\{X_t\}$ at lag h as:

$$\rho_h = \frac{\gamma_h}{\gamma_0} = \frac{Cov(X_t, X_{t+h})}{Var(X_t)}, h \in \mathbb{Z}. \tag{3}$$

Besides ACVF and ACF, there is also the concept of partial autocorrelation function (PACF), which measures the effects of the relationship between y_t and y_{t-h}, when the effects of the lags $1, 2, 3, ..., h-1$ are removed (Hyndman and Athanasopoulos, 2014).

3.2. Time Series Models

We consider the general model for fitting non-seasonal and non-stationary time series: the AutoRegressive Integrated Moving-Average (ARIMA) model.

While the moving average process is a multiple regression with past errors as predictors (Hyndman and Athanasopoulos, 2014), in the autoregressive process, the predictors are lagged values $\{X_s : s < t\}$ of the time series $\{X_t\}$, plus a random component at instant t that is characterized to be white noise. An autoregressive moving average process combines the two previous models, in the sense that it has as predictors both lagged values of the time series and also lagged errors. The integrated part refers to the number of differencing needed to make the time series stationary.

Non-seasonal ARIMA models are generally denoted as $ARIMA(p, d, q)$ where the parameters p, d, and q are non-negative integers, p is the order (number of time lags) of the autoregressive model, d is the degree of differencing, and q is the order of the moving-average model.

The ARIMA process can also be used to model time series with a seasonal component. In this case, we have Seasonal ARIMA models, which are usually denoted as $SARIMA(p, d, q)(P, D, Q)m$, where m refers to the number of periods in each season, and P, D, Q refer to the autoregressive, differencing, and moving average terms for the seasonal part of the ARIMA model.

3.3. Generalized Additive Models

In this Subsection, we introduce a type of models, Generalized Additive Models (GAMs), that incorporate explanatory variables. Suppose that we have a response random variable Y and a set of predictor random variables $W_1, W_2, ..., W_p$. A set of n independent realizations of these random variables is denoted by $(y_1, w_{11}, ..., w_{1p}), ..., (y_n, w_{n1}, ..., w_{np})$.

Before introducing GAMs, we start by defining Generalized Linear Models (GLMs), which, according to Wood [2006], are linear models that allow for response distributions other than Normal and a degree of non-linearity in the model structure.

A Generalized Additive Model is a Generalized Linear Model with a linear predictor involving a sum of smooth functions of covariates,

$$g(\mu_i) = f_0 + f_1(W_1) + ... + f_p(W_p), \qquad (4)$$

where f_j are smooth functions of the covariates W_j for all $j = 0, ..., p$ and $W_0 = I$.

For simplicity, let us look at the case of a single predictor and where the link function is the identity function:

$$Y = f(W). \qquad (5)$$

Consider n samples (w_j, y_j), $j = 1, ..., n$, of covariates and dependent variables. In order to estimate f, this function is represented in such a way that equation 5 becomes a linear model. This can be done by choosing a basis, that is, by defining the space of functions of which f is an element. If $b_i(x)$ is the ith such basis function, then the smooth function f can be represented as

$$f(w) = \sum_{i=1}^{q} \beta_i b_i(w) = \beta^T b(w) \qquad (6)$$

where β_i are unknown parameters (the spline coefficients), β is a vector containing the spline coefficients and $b(w)$ is the vector containing the spline basis functions. We say that f is modelled by regression splines.

The model's smoothness can be controlled by adding a penalty to the least squares fitting objective, known as smoothing parameter, λ. For example, rather than minimizing the residual sum of squares

$$||y - \beta^T B(w)||^2, \qquad (7)$$

where B is the matrix with the rows $b(w_1)^T, b(w_2)^T, ..., b(w_n)^T$ containing the evaluated spline basis functions, we instead add to the residual sum of squares a penalty: some multiple λ of the integral of the squared second derivative of $f(w)$ with respect to w, which penalizes steep slopes. Consider a small interval δw over which the second derivative $f''(w)$ of the smoother $f(w)$ is approximately constant. The contribution of that interval to the penalty is then $\lambda f''(w)^2 \delta w$.

Then it is minimized

$$||y - \beta^T B(w)||^2 + \lambda \int_0^1 [f''(w)]^2 dw, \qquad (8)$$

where the constant λ is determined by cross correlation. If instead of a single variable, we have several variables, then the contributions of several variables can be added. There is then one λ_i, $i = 1, ..., p$, for each of the p variables.

It can be shown (see Wood [2006]) that the formal expression for the minimizer of equation 8, the penalized least squares estimator of β, is

$$\hat{\beta} = (B^T B + \lambda S)^{-1} B^T y \qquad (9)$$

and the hat matrix A of the model can be written as

$$A = B(B^T B + \lambda S)^{-1} B^T. \qquad (10)$$

The problem of estimating the degree of smoothness for the model is now the problem of estimating the smoothing parameter λ, which can be done by cross validation.

3.4. Clustering Time Series

Throughout the thesis, we make use of the hierarchical algorithm for the clustering procedure.

The choice of dissimilarity measure is very important, as it has a strong effect on the resulting dendrogram. We took into account several dissimilarity measures (see Montero and Vilar [2014]), but we end up choosing the one based on the spectral analysis of a time series proposed by Caiado et al. [2014].

According to Brockwell and Davis [2002], the spectral representation of a stationary time series $\{X_t\}$ essentially decomposes $\{X_t\}$ into a sum of sinusoidal components with uncorrelated random coefficients. In conjunction with this decomposition there is a corresponding decomposition into sinusoids of the autocovariance function of $\{X_t\}$. The spectral decomposition is thus an analogue for stationary processes of the more familiar Fourier representation of deterministic functions. The analysis of stationary processes by means of their spectral representation is often referred to as the "frequency domain analysis" of time series or "spectral analysis".

The frequency domain approach allows to measure the dissimilarity between time series. The key idea is to assess the dissimilarity between the corresponding spectral representations of the series, which is the base of periodogram-based distances.

Having applied the hierarchical clustering, a wide variety of indexes have been proposed to find the optimal number of clusters in a partitioning of a data set during the clustering process (see Charrad et al., 2014). We used mainly 4 indexes: Gamma, Silhouette, Dunn and Entropy.

4. Results

This Chapter is dedicated to state the results of clustering 25 load curves and forecasting one day ahead electricity consumption of the representative time series of each cluster. Firstly, Subsection 4.1 is intended to describe the given data for this study. In Subsection 4.2 it is presented the clustering of the time series given. Finally, in Sections 4.3 and 4.4 it is explained the procedure of fitting a Generalized Additive Model and a SARIMA model, respectively, to the representative time series of the cluster, as well as presenting the results of the forecast.

4.1. Description of the Data

We were given 25 time series, each containing observations of electrical consumption of a certain customer, measured in kilowatt (kW), at every XXX minutes, starting at 2015-05-05 05:15 and ending at 2017-03-19 09:00, which accounts for almost two years of observations.

Besides 25 load curves, we were also given weather time series, because, for the modelling using GAM, we intend to include weather variables in the model. Each of the 25 time series is from a different location, and, for each location, we were given 7 weather variables, namely, the radiance, atmospheric pressure, temperature, wind direction, wind speed, wind U and wind V (where a positive U component represents wind blowing to the East and a positive V represents wind to the North). The time period for which we have information about the weather is the same period as for the electrical consumption time series.

In the next section it is detailed the clustering procedure, which only makes use of the 25 load time series. Weather time series will be used in Section 4.3, when fitting a generalized additive model.

4.2. Time Series Clustering

This section is intended to state the results of grouping 25 load time series by similarity of consumption. In order to output this grouping, it is necessary to indicate what is the definition of similarity of consumption. First of all, note that, when plotting each of the 25 load time series, we could understand that these have different scales. This means that, if we perform the clustering without any transformation of the time series, these will be grouped by similarity of consumption in scale.

However, when clustering time series, we decided that we should give primarily attention to group customers with the same consumption pattern rather than focusing on scale, because we want to fit models that clearly translate daily, weekly or yearly seasonalities. So, first, it is necessary to normalize each time series, as presented in Figure 1. There are several ways to perform a normalization (see Juszczak et al. [7]), one of which is using the formula

$$Y_t = \frac{X_t - mean(X_t)}{sd(X_t)} \tag{11}$$

where Y_t denotes the normalized time series, X_t is the original time series and sd stands for the standard deviation. This was the one that performed better, so this was the one that we end up choosing.

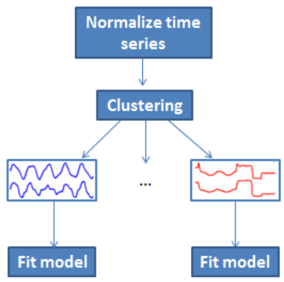

Figure 1: Approach for clustering and fitting models.

The time series will be clustered by applying hierarchical clustering algorithm with the distance based on the periodograms and we want to choose a partition that has the optimal number of clusters. This choice is based on the majority vote of the best number of clusters returned by 4 validation indexes: Dunn, Entropy, Silhouette and Gamma index. The best number of clusters returned was 6, which leads to partitioning the dendrogram as in Figure 2.

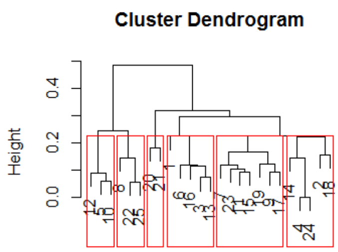

Figure 2: Partition of the 25 time series in 6 clusters.

Let the labels of the resulting clusters be C_1, C_2, C_3, C_4, C_5 and C_6, where $C_1 = \{1, 3, 6, 13, 16\}$, $C_2 = \{2, 4, 14, 18, 24\}$, $C_3 = \{5, 10, 12\}$, $C_4 = \{7, 9, 11, 15, 17, 19, 23\}$, $C_5 = \{8, 22, 25\}$ and $C_6 = \{20, 21\}$.

4.3. Modelling using GAM

Our aim is to fit a Generalized Additive model in order to model and forecast electricity load of the time series within one cluster. The advantage of this model is that it allows the inclusion of external variables in the model and it is able to capture non-linear effects of the variables.

We will describe in detail the procedure for modelling and forecasting time series in cluster 6. For the remaining clusters, modelling and forecasting follow the same reasoning.

For cluster C_6, which contains time series 20 and 21, the input and output of our procedure is the following:

- **input:** Timestamp, time series 20, time series 21, 7 distinct weather variables from the location of time series 20, 7 distinct weather variables from the location of time series 21;

- **output:** One day ahead forecast of electricity consumption of the representative time series of the cluster, for every 15 minutes.

Given the normalized time series within cluster C_6 and without performing further transformation to the data, we started by getting the representative time series. The model will be fitted to the obtained time series.

The variables that we identified as best to be included in the model are the following:

- Day type: categorical variable that represents the day type. Every week day get its own factor; also, it is important to set a factor for public holidays, for which consumption may be distinct from the remaining types of days. So, the factors of this variable are then 1 for Mondays, 2 for Tuesdays, 3 for Wednesdays, 4 for Thursdays, 5 for Fridays, 6 for Saturdays, 7 for Sundays, and 8 for public holidays;

- Time of day: categorical variable that represents the current time within the day (measured in quarter-hourly time steps). The total number of observation within one day is 96, because it is the result from multiplying 24 hours by 4, where 4 is the number of 15 minutes periods within one hour. So, the factors go from 0, representing hour 00 and minute 00 of the day, until 95, representing hour 23 and minute 45 of the day;

- Time of year: categorical variable that represents the current day and month within the year. The factors go from 0, representing the 1st of January, until 365, representing the 31st of December;

- Lags of electricity consumption. In order to find out which is the right lag of consumption, we make use of the cross correlation function (ccf). In general, given two time series, y_t and x_t, we want to find out whether the series y_t may be related to past lags of the x-series. The sample cross correlation function (ccf) is helpful for identifying lags of the x-variable that might be useful predictors of y_t. The sample ccf is defined as the set of sample correlations between x_{t+h} and y_t for $h = 0, \pm 1, \pm 2, \pm 3$, and so on. So, if we compute the sample ccf of the electricity consumption with it self, the lag with greater sample ccf should be the one to be included in the model. Lag 96 is the one with greater sample ccf, so this is the one that we should include in the model.

- Lags of weather variables. Notice that each time series within the cluster is from a different location and we are fitting a model for the aggregated model, which means that we need to decide from which location we should choose the weather variables. Our strategy to solve this problem is the following: we build one GAM using weather variables of the location of time series 20, another GAM using weather variables of the location of time series 21 and the final GAM will be an ensemble, i.e., the mean of the two last generalized additive models (for clusters that have more than 2 time series, we choose two time series at random to perform the ensemble).

We first consider the weather variables of time series 20. Recall that we are given seven weather variables. However, some of them might be correlated with each other, which means that, we shall detect which ones are correlated with and only include in the model not correlated variables. By analysing the sample ccf between temperature and the weather variables, we can conclude that the variables temperature and wind speed from the location of time series 20 are the ones that should be included in the first GAM. A similar study of the weather variables from the location of time series 21 resulted in the same choice of variables for the second GAM.

It remains to see, from the weather variables chosen, which are the lags that should be included in the model. The lags chosen are the ones that represent the greater sample ccf between the weather variables and the load. For the first GAM, the temperature lag should be 48, and the wind speed lag should be 48 as well. For the second GAM, the temperature lag should be 48, whereas the wind speed lag should be 198.

To sum up, since cluster C_6 is composed by 2 time series from different locations, we fit two generalized additive models, with covariates $x1_t$ and $x2_t$, respectively, given by:

$$x1_t = \{daytype_t, timeday_t, timeyear_t, y_{t-96}, \\ ts20.temperature_{t-48}, ts20.windspeed_{t-48}\} \quad (12)$$

$$x2_t = \{daytype_t, timeday_t, timeyear_t, y_{t-96}, \\ ts21.temperature_{t-48}, ts21.windspeed_{t-198}\} \quad (13)$$

where y_t denotes the values of the aggregated time series of electricity consumption at time t.

We split the aggregated time series y_t (which is the response variable) and the covariates $x1_t$ and $x2_t$ in a train set (used for fitting the model) and in a test set (used for compare forecast with true values).

Our aim is to create two models that fit the aggregated data until the 9th of January of 2017 and to predict the load curve for the 10th of January of 2017, which is a Tuesday. So the train set will be the observations from the 5th of May of 2015 to the 9th of January of 2017 and the test set will be from the 10th of January of 2017 until the 19th of March of 2017.

We fit the following two generalized additive models for the electricity load:

$$\textbf{Model 1: } y_t = \beta^{intercept} + f(trend_t) + \\ f(y_{t-96}) + f(ts20.temperature_{t-48}) + \\ f(ts20.windspeed_{t-48}) + f(timeyear_t) + \\ \sum_{l=1}^{8} I_{\{daytype_t = l\}} f(timeday_t) \quad (14)$$

$$\textbf{Model 2: } y_t = \beta^{intercept} + f(trend_t) + \\ f(y_{t-96}) + f(ts21.temperature_{t-48}) + \\ f(ts21.windspeed_{t-198}) + f(timeyear_t) + \\ \sum_{l=1}^{8} I_{\{daytype_t = l\}} f(timeday_t) \quad (15)$$

where:

- $\beta^{intercept}$ models the base load and $f(trend_t)$ captures non-linear trends;
- $f(y_{t-96})$ takes into account the electricity load of the previous day;
- $daytype_t$ and $f(timeday_t)$ capture the day-type specific effects of the time of the day;
- $f(ts20.temperature_{t-48})$ and $f(ts20.windspeed_{t-48})$ take into account, respectively, the temperature and the wind speed of the previous half day from the location of the client related to time series 20;
- $f(ts21.temperature_{t-48})$ and $f(ts21.windspeed_{t-198})$ take into account, respectively, the temperature of the previous half day and the wind speed of the previous two days from the location of the client related to time series 21, and
- $f(timeyear_t)$ represents yearly cycles.

Thin plate regression splines is the basis used to represent the smooth functions f, because these were the the optimal smoother of any given basis dimension. The final model, which we call ensemble model, will correspond to the mean at each observation between the fitted models 1 and 2.

Having the ensemble model, the next step aims at analysing the residuals, namely, its stationarity and normality.

KPSS test, which tests a null hypothesis that an observable time series is stationary, returns a p-value greater than 0.1. We reject the null hypothesis significance levels greater than the p-value, so, for the usual significance levels (1%, 5% and 10%), we do not reject the null hypothesis, meaning that the residuals are stationary.

However, by analysis of the histogram (Figure 3) and QQ-plot (Figure 4) of the aggregated time series, the aggregated time series does not seem to follow a Normal distribution. In spite of trying many transformations to the data, this continued to occur, so we proceed with the analysis.

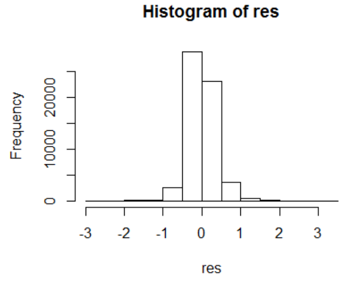

Figure 3: Histogram of the residuals of the ensemble model.

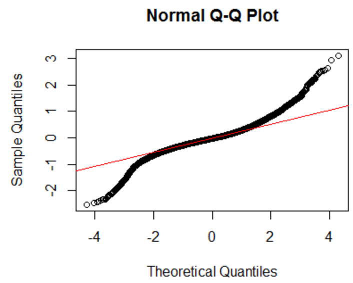

Figure 4: QQ-plot of the residuals of the ensemble model.

Figures 5 is the graphic of the residuals *versus* the linear predictor. The points appear randomly around zero, which indicates that the residuals are uncorrelated.

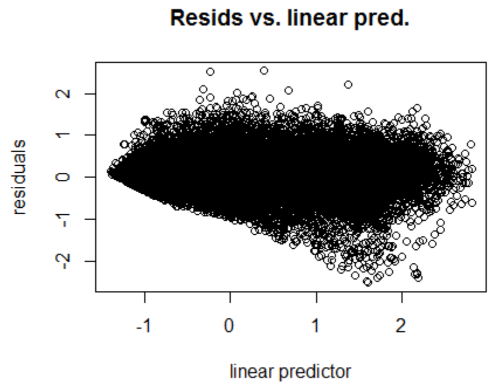

Figure 5: Plot of the linear predictor *versus* residuals.

The one day prediction for the aggregated normalized time series of cluster 6 is presented in Figure 6. Green lines represent the test values, whereas blue lines represent the forecast.

Figure 6: Confidential Annex: Forecast the normalized aggregated time series of cluster 6.

Notice that we are dealing with normalized values until now. Next, we can transform the final predictor vector to its original scale using the formula:

$$Z_t = Y_t * sd(X_t) + mean(X_t) \tag{16}$$

where Y_t denotes the normalized predictor vector, X_t is the non-normalized representative time series of the cluster and Z_t denotes the non-normalized predictor vector. Figure 7 compares the forecast with the true values of the load in its aggregated original scale.

Figure 7: Confidential Annex: Forecast the non-normalized aggregated time series of cluster 6.

Since we have the true values for the 10th of January of 2017, we intended to analyse the performance of the method, by comparing the forecast with the true values. We computed the MAPE of the forecast with the original scale and the returned MAPE for this forecast was 16.6%.

We noticed that the residuals of the model were not normal, which means that we cannot make use of the normality to create prediction intervals. Our way of contour this problem was to create, not only an ensemble model for the aggregated time series using the median, but also two more ensemble models: the first one will fit the aggregated time series where the function using to perform the aggregation is the minimum at each observation, and the second one will fit the aggregated time series where the function using to perform the aggregation is the maximum at each observation. The forecast of these two new ensemble models will give an idea of the maximum and minimum values of consumption.

The one day prediction for the aggregated normalized time series of cluster 6 using the median, minimum and maximum is presented in Figure 8. Green lines represent the test values, whereas red lines represent the predictor vector, with the dash lines representing the aggregated prediction using the minimum and the maximum function and the thick red line representing the aggregated prediction using the median.

Figure 8: Confidential Annex: Forecast the aggregated time series using median, minimum and maximum at each observation of cluster 6.

Following the procedure for fitting an ensemble GAM for the remaining clusters, we computed predictions for the 10th of January of 2017 (tuesday), the 1st of January of 2017 (public holiday), the 6th of January of 2017 (saturday) and finally for the 30th of August of 2016 (summer holidays). The MAPE for each cluster and day is presented in Tables 1 and 2.

	10 JAN 2017	1 JAN 2017
C1	11.36	288.35
C2	5.27	1.63
C3	8.90	9.02
C4	10.40	7.60
C5	4.86	6.86
C6	16.6	23.4

Table 1: MAPE (%) for the aggregated function of each cluster using the function median for a Tuesday on winter and for a public holiday.

	6 JAN 2017	30 AUG 2016
C1	18.71	5.57
C2	12.67	21.62
C3	8.99	3.76
C4	8.74	7.93
C5	4.70	3.51
C6	19.98	15.71

Table 2: MAPE (%) for the aggregated function of each cluster using the function median for a Saturday on winter and for a summer day.

For cluster 1, we can see that there is an anomalous MAPE in the 1st of January. This may be due to the fact of the consumption during this public holiday being completely different from the remaining time of year, which makes it hard to predict. For the remaining cases, we can see that the MAPE varies between around 4% and 23%.

4.4. Modelling using SARIMA

In order to compare the performance of the Generalized Additive Model, we fit a seasonal ARIMA model to the representative time series of the cluster, that is, a $SARIMA(p, d, q)(P, D, Q)[m]$ model. Once again, we illustrate the model procedure for cluster 6 and forecast the load for a typical Tuesday on winter.

The representative time series of the cluster is stationary, so there is no need to differentiate, so $d = D = 0$. The frequency of the time series is one day, that is, 96 observations, hence $m = 96$. Therefore, it remains to determine p, q, P and Q, using the auxiliary graphs of the sample ACF (Figure 9) and PACF (Figure 10).

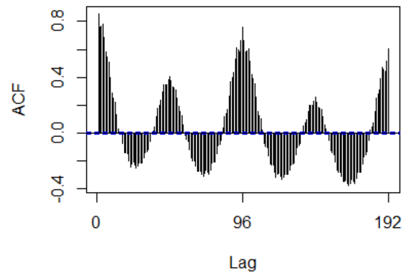

Figure 9: Sample ACF.

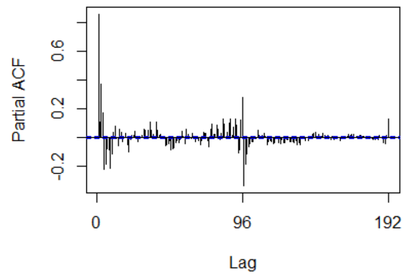

Figure 10: Sample PACF

By analysing Figures 9 and 10, our first proposal of model is $ARIMA(1, 0, 0)(1, 0, 0)[96]$, due to the peak at sample PACF at lag 96 and due to the sinusoidal pattern of the sample ACF. However, the incrementation of q was leading to a decreasing of the AICC score, which we want to be minimum. Hence, by analysis of the value of AICC, we reached the model $ARIMA(1, 0, 6)(1, 0, 0)[96]$.

Having the model, the next step aims at analysing the residuals, namely, its stationarity and normality.

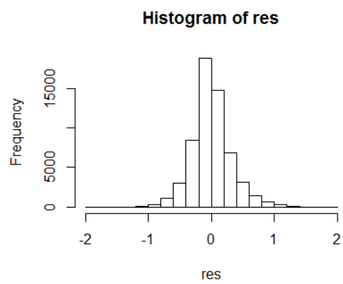

Figure 11: Histogram of the residuals of the SARIMA model.

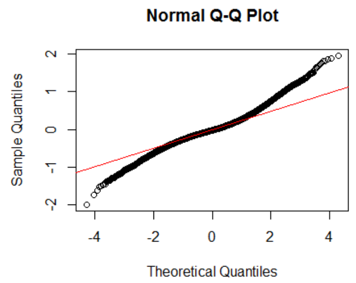

Figure 12: QQ-plot of the residuals of the SARIMA model.

KPSS test, which tests a null hypothesis that a time series is stationary, returns a p-value greater than 0.1. We reject the null hypothesis significance levels greater than the p-value, so, for the usual significance levels (1%, 5% and 10%), we do not reject the null hypothesis, meaning that the residuals are stationary.

However, by analysis of the histogram (Figure 11) and QQ-plot (Figure 12) of the aggregated time series, the aggregated time series seems to be close to follow a Normal distribution but still presenting heavy tails, so, once again, we are not able to produce forecast intervals for the prediction.

The one day prediction for the aggregated normalized time series of cluster 6 is presented in Figure 13. Green lines represent the test values, whereas blue lines represent the forecast.

Figure 13: Confidential Annex: Forecast the aggregated time series of cluster 6 using SARIMA.

After transforming the final predictor vector into its original scale, we computed the MAPE of the forecast. The result was 20.1%, which is higher value of MAPE than the one obtained for GAM, hence, in this case, the prediction is worst when fitting a SARIMA model.

For the remaining clusters, the procedure of fitting a SARIMA model is the same. The only step that must be taking care of and may be different from cluster to cluster is the choice the parameters of the model, which is done by analysing the sample ACF and PACF of each representative time series of the clusters.

Following the procedure for fitting a SARIMA model the remaining clusters, we computed predictions for the 10th of January of 2017 (Tuesday), the 1st of January of 2017 (public holiday), the 6th of January of 2017 (Saturday) and finally for the 30th of August of 2016 (summer holidays) by fitting a SARIMA model. The values of MAPE go from approximately 18% until 35%, and we could see an anomalous value of 70% during a Saturday for cluster 1. The value of MAPE is lower for only 2 cases out of 20 when comparing to the Generalized Additive Model. This means that, overall, fitting a SARIMA model is not better than fitting a Generalized Additive Model.

5. Conclusions

This chapter is dedicated to state the achievements obtained thought this study (subsection 5.1) and a suggestion of directions for future work (subsection 5.2).

5.1. Achievements

In this study, we were focused in forecasting electricity consumption, so we were given a set of 25 time series that contained measures of consumption of 25 different clients through almost 2 years.

It is impractical to create models and to forecast consumption for each of the time series given, so it was necessary to start by performing a clustering method, in order to group clients by similarity of consumption. The chosen method was hierarchical clustering and two approaches were analysed: either we could apply the clustering method to raw data or to normalized data. We concluded that, when the data were first normalized, then we obtained partitions that could best reflect the consumption patterns of the clients (hourly, daily or monthly behaviours); on the other hand, in the presence of raw data, the clustering would be focused mainly on scale. Since we were more interested in creating models posteriorly that would translate trends and seasonalities of the data, we conclude that it is necessary to perform a normalization. The distance that we applied when performing the clustering was based in the periodograms of the normalized time series, because this was the most appropriated one that goes along with our objective of extracting and comparing the patterns of time series. The best number of clusters was determined by analysis of several validation indexes.

Having the right partition of the 25 time series according to their similarity, we aimed at building a model that could represent the consumption of each cluster and comparing the forecast accuracy of two models: Generalized Additive Models and SARIMA models. The GAM has the advantage of including external variables and capturing nonlinear effects of the variables. Therefore, we were also given weather variables so that we could studied the correlation between these and the consumption. However, notice that each of the 25 clients are from different locations and recall that the model is being fitted to a time series that represents the consumption of the whole cluster, which means that we had to decide which locations should be included in the model. So, for the GAM, we created an ensemble model: given a cluster of clients, we chose the locations of two clients at random (two because of computational costs, but this number could be extended) and we created one model with weather variables from one client and another model with weather variables from the other. The ensemble is defined to be the mean of the two fitted models. By comparing the forecast accuracy of the GAM and the SARIMA models, we could understand that, in general, the GAM performs better.

5.2. Future Work

The proposed methods in this thesis were applied to 25 time series. The suggestion of future work is to apply the same methods to a larger set of time series. Assume that we have a set S_N that contains N load curves, with $N >> 25$. Then we apply the clustering methods to the set S_N and we fit a GAM to each cluster. Now suppose that we have a new load curve that was not in the set S_N. Then, we could apply classification methods, such as the K-Nearest Neighbours, to classify the new time series in one of the clusters, identified previously, and forecast its consumption by making use of the aggregated model of that cluster. In other words, we suggest not to apply the clustering methods to the whole customer portfolio, only part of it, and classify the remaining customers into the right clusters, based on classifying methods.

References

[1] Brockwell, P. J. and Davis, R. A. (2002). *Introduction to Time Series and Forecasting.* Springer, 2^{nd} edition. ISBN 0-387-95351-5.

[2] Caiado, J., Crato, N., and Pea, D. (2014). A periodogram-based metric for time series classification. *Elsevier*, 50:2668 2684. DOI:10.1016/j.csda.2005.04.012.

[3] Charrad, M., Ghazzali, N., Boiteau, V., and Niknafs, A. (2014). NbClust: An R package for determining the relevant number of clusters in a data set. *Journal of Statistical Software*, 61.

[4] Chicco, G. (2012). Overview and performance assessment of the clustering methods for electrical load pattern grouping. *Energy*, 42:68–80. DOI:10.1016/j.energy.2011.12.031.

[5] Huang, S. J. and Shih, K. R. (2003). Short-term load forecasting via arma model identification including non-gaussian process considerations. *IEEE Transactions on Power Systems*, 18:673 – 679. DOI: 10.1109/TPWRS.2003.811010.

[6] Hyndman, R. J. and Athanasopoulos, G. (2014). *Forecasting: Principles and Practice.* Otexts, 2^{nd} edition.

[7] Juszczak, P., Tax, D., and Duin, R. (2000). Feature scaling in support vector data description. DOI: 10.1.1.100.2524.

[8] Montero, P. and Vilar, J. A. (2014). Tsclust: An R package for time series clustering. *Journal of Statistical Software*, 62.

[9] Taylor, J. W. (2012). Short-term load forecasting with exponentially weighted methods. *IEEE Transactions on Power Systems*, 27:458 – 464. DOI: 10.1109/TPWRS.2011.2161780.

[10] Wood, S. N. (2006). *Generalized Additive Models: an introducing to R.* Chapman and Hall/CRC. ISBN 9781584884743.

[11] Yannig Goude, Pascal Pompey, A. B. and Sinn, M. (2012). Adaptive learning of smoothing functions: Application to electricity load forecasting. In *Advances in Neural Information Processing Systems 25*, pages 2519–2527. NIPS.

[12] Zitkovic, G. (2010). Introduction to stochastic processes - lecture notes.

6. Confidential Annex