

POLITECNICO DI TORINO

**Corso di Laurea Magistrale
in Ingegneria Energetica e Nucleare**

Tesi di Laurea Magistrale

Development of a tool for anomaly detection and power load forecasting: the case of Politecnico di Torino



Relatore

prof. Alfonso Capozzoli

firma del relatore

.....

Co-relatore

ing. Silvio Brandi

Candidato

Edoardo Chiabrera

firma del candidato

.....

A.A. 2017/2018

Acknowledgements

I wish to express my sincere thanks to the members of the staff of Politecnico di Torino who supported me during all the development of this thesis. I would like to pay a special thankfulness to ing. Giovanni Carioni and to the Living Lab for providing me all the information and data needed to achieve the goal of this dissertation.

Table of contents

Acknowledgements	2
1 Introduction	2
1.1 Forward models.....	3
1.2 Inverse models.....	3
2 Literature survey on the state of the art inverse modelling for energy and buildings .7	7
2.1 Forecasting.....	10
2.2 Model Predictive Control (MPC)	14
2.3 Demand response.....	17
2.4 Fault detection.....	18
2.5 Benchmarking	20
3 Data mining and forecasting methods	24
3.1 Data mining methods.....	24
3.1.1 Clustering	25
3.1.2 Association rule mining	27
3.2 Forecasting methods	28
3.2.1 Support vector regression.....	28
3.2.2 Artificial neural network.....	31
3.2.3 Classification and regression trees	36
3.2.4 ARIMA models.....	38
4 Description of the dataset.....	41
4.1 Row data analysis	42
4.2 Corrected data analysis	56
5 Data pre-processing.....	75
5.1 Clustering analysis	75
5.2 Classification tree analysis	85
5.3 Refrigeration unit analysis	88
6 Anomaly detection and energy benchmarking	92
6.1 General framework	92
6.2 Regressive CART implementation.....	93
6.3 Non-autoregressive artificial neural network.....	95
6.4 Model results	98
6.5 Anomalies and rare events detection and labelling.....	102
7 Forecasting model for the enhancement of building power load management and diagnosis.....	108
7.1 Electrical consumption forecasting	108
7.2 Consumption at final service level.....	113
8 Conclusions.....	119
9 Bibliography.....	122
List of figures.....	126
List of tables	129
10 Appendix.....	130

Chapter 1:

Energy Estimating and Modeling Methods

1 Introduction

The building sector is one of the most energy consuming ones. As can be seen from Figure 1.1, in 2015 the building sector accounted for the 30% of total final energy use, divided as 22% for residential buildings and 8% for non-residential ones (1).

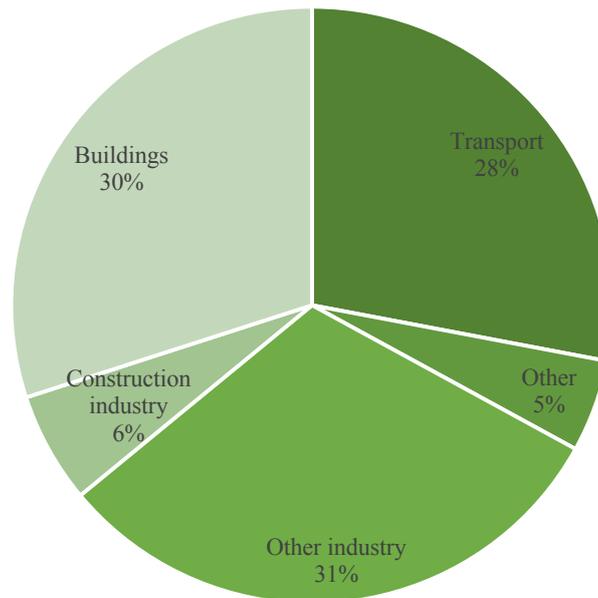


Figure 1.1: Global final energy consumption by sector, 2015. taken from (1) with changes.

As pointed out by González and Zamarreño (2), in order to set up a good operation of the energy management system (EMS) it's necessary to have information about the future building energy demand. To do so one can adopt a solution called short term load forecasting (STLF), which consists in the energy demand prediction during a period of minutes, hours, days and also weeks.

The importance of enhancing the building automation and control system (BACS) and the technical building management (TBM) has been recognised as a crucial factor for controlling the building energy consumption (3). As the authors underline, the attention given to the automation, control and monitoring systems is growing also at European level, since the European Performance in Building Directive (EPBD) 2010/31/EU (4) encourages the installation of active control systems and intelligent metering systems which are able to guarantee energy savings for new-constructions or for buildings undergoing a major renovation.

Energy forecasting has proven to be an effective tool also in the fields of fault detection, demand response, energy benchmarking and MPC (Model Predictive Control). In each case the prediction of the energy demand can be crucial as well as its accuracy.

As remarked in (5) starting from the 1990s researchers developed a large quantity of simulation tools to predict the energy demand in buildings. These methods can be categorized as forward or inverse.

1.1 Forward models

Forward, engineering or white-box models have the goal “to predict the output variables of a specified model with known structure and known parameters when subject to specified input variables” (6). Such models are usually employed during the preliminary phases of a project, when the building, or the building system have not been built yet, in fact, they do not require any real measured data. On the other hand, they necessitate of a significant amount of information about the interactions between the system and the external environment, including their nature and their extent. Apart from being applicable before the effective construction of the building, these models have the advantages of being clearly interpretable, since all the input parameters can be controlled by the analyst, and of giving the possibility to simulate the effects of energy efficiency measures before their implementation.

As stated in ASHRAE Handbook Fundamentals (6), all the forward models, can be divided in three development phases. First of all, the estimation of the space load, the amount of thermal energy which is required to maintain acceptable indoor conditions, is needed. As a second stage, one has to calculate the load on the secondary equipment which corresponds to the space load just obtained, taking into account all the equipment which stand between the central plant and the zones. Finally, the primary equipment, which usually includes generation and storage systems, is considered and its energy need is calculated starting from the ones of the secondary equipment. As reported in (6), some of the most famous simulation tools exploiting such models are TRNSYS, DOE2, ESP-r and Energy Plus.

1.2 Inverse models

Inverse or data-driven models are aimed at finding a mathematical relation which is able to describe the system and evaluate its parameters, while real measured input and output variables are at disposal. By definition, such models are useful only when the system is already in operation, so

that actual data are available to develop the model. As stated in (6), inverse models are often simpler and more precise than forward ones in predicting the future behaviour of the system. These tools require a limited number of parameters to be set by the analyst, since the information are grouped in aggregated parameters which are obtained from actual data and which are more likely to reflect the real system performance. By contrast, these models are less versatile when they have to be used to evaluate different design solutions as well as different energy efficiency measures.

Inverse models can be divided into three categories, according to (6): black-box, calibrated simulation or grey-box approaches. Empirical or black-box models include all the algorithms which identify a simple or multivariate regression model which links the output, usually the measured amount of energy used, with a number of inputs, including historical energy consumption data, weather conditions, building use, schedules etc. They have the peculiarity that their coefficients cannot be correlated with any physical meaning. According to ASHRAE Handbook Fundamentals (6) these are the most widely used inverse models.

Calibrated simulation approach refers to the models which use building simulation programs, by tuning their input parameters, in order to obtain simulated energy consumptions that closely match the actual measured ones. The main drawback of such models is that the calibration procedure can be very laborious and requires a user which has deep skills in building simulation and operation. Moreover, the results obtained often depend on the person who has carried out the calibration procedure, thus they are not always reproducible.

Grey-box approach refers to hybrid methods which put together a physical model, representing the structure of the system, and a statistical one, that is able to identify important aggregated parameters representative for the whole building. By doing so, they inevitably carry around some of the shortcomings of the previously discussed models (5), but they have a great potential for what concerns particularly fault detection and diagnosis as well as online control (6).

Another distinction among inverse models can be done according to their capacity of considering temperature transients. Under this perspective data-driven models can be divided into steady-state and dynamic models. Steady-state inverse models do not consider quantities like thermal mass or capacitance, for this reason they are not suitable for assessing more than monthly, weekly or daily energy consumption. By contrast, dynamic data-driven models use hourly or even sub-hourly data and take into consideration effects like the delay of the heat losses or gains, load peaks and warm-up or cool-down transients.

To clarify the distinction among the described models one could look at Table 1.1. in which some models that are used to analyse the energy consumption of buildings are classified.

Table 1.1: Classification of some energy analysis methods. Adapted from ASHRAE Handbook Fundamentals (6).

Method	Model category	Model category	sub-	Model type
Modified degree-day method	Forward	-		Steady-state
Transient simulation: TRNSYS, HVACSIM+	Forward	-		Dynamic
Simple linear regression	Data-driven	Black-box		Steady-state
Multiple linear regression	Data-driven	Black-box		Steady-state
Artificial neural network	Data-driven	Black-box		Dynamic
Support Vector Machine	Data-driven	Black-box		Dynamic
Computer simulation: DOE-2, EnergyPlus, ESP-r	Forward/ Data-driven	Calibrated simulation		Dynamic
Differential equation	Data-driven	Gray-box		Dynamic

This work will be focused on inverse black-box models because of their reliability and high accuracy. Moreover, when the analysis has to be carried out on an existing system for which a consistent amount of data is at disposal, the choice of an inverse model, instead of a forward model, can be the right one. In fact, it is possible to achieve very high accuracy in a reproducible way, since in this case the results will not be affected by the calibration procedure as in the case of calibrated simulation models. First of all, a literature review will be carried out to underline the principal and most promising fields of application of data-driven black-box models. Then a description of some of the principal data analytics models will be performed. Finally, the case study of the electrical substation C of Politecnico di Torino will be presented and some of the previously described techniques will be used to develop a simulation tool that, on one hand, is able to detect anomalies and rare events and tag them with a label, on the other hand, can forecast the hourly 24 hours ahead power load giving the possibility to improve the energy control strategies and to increase the users' awareness about the behaviour of their system.

Chapter 2:

State of the art and current trends in inverse
modelling for energy and buildings

2 Literature survey on the state of the art inverse modelling for energy and buildings

A considerable number of articles about data-driven building energy forecasting has been published in the last years. In order to clarify the nomenclature which will be used in the rest of this section it's useful to look at Figure 2.1. in which the algorithms used in the papers reviewed in this section are classified. The diagram shows that black-box method can be split in four categories according to either their purpose or their nature.

Clustering methods are a particular kind of AI-based algorithms with the aim of grouping elements which show analogous behaviours, separating them from the ones which have other similarities among them, a more in-depth analysis can be found in section 3.1.1.

Again, with regard to their purpose, the algorithms can be split into classification or regression ones. Classification problems are the ones in which an input variable or a number of input variables are used to estimate a function which gives discrete output variables, which in this case are called labels. On the other hand, a regression algorithm still uses an input variable or a number of input variables to estimate a function which gives output variables, but this time those outputs are continuous. Some of the methods used in classification problems can also be used in regression ones, in an adapted way.

One of the most famous algorithms used in both cases is Artificial Neural Network (ANN). This method is inspired to the functioning of the human brain and is composed by a structure made of neurons organised in layers, connected one with each other. Each neuron, or node, contains a specific activation function which determines its outputs given a number of inputs. A mathematical explanation of the functioning of ANNs can be found in section 3.2.2. At this point it can be useful to distinguish among the main types of neural networks. The simplest ones are Feedforward ANNs in which the information travel only in the direction that goes from the input layer to the output one. Another kind of ANNs is represented by the Radial Basis Function networks, which have the peculiarity of using radial basis activation functions for their neurons. Recurrent neural networks (RNN) are able to propagate information forward and then backward keeping memory of previous instances, thus they are a more sophisticate kind of networks. Autoregressive neural networks exploit the statistical autoregressive model, so that the output variable becomes dependent linearly on its own past values and also on a random quantity. Finally, the term Deep Neural Network (DNN) refers to each network which has more than one hidden layer, as well as an input and an output layer.

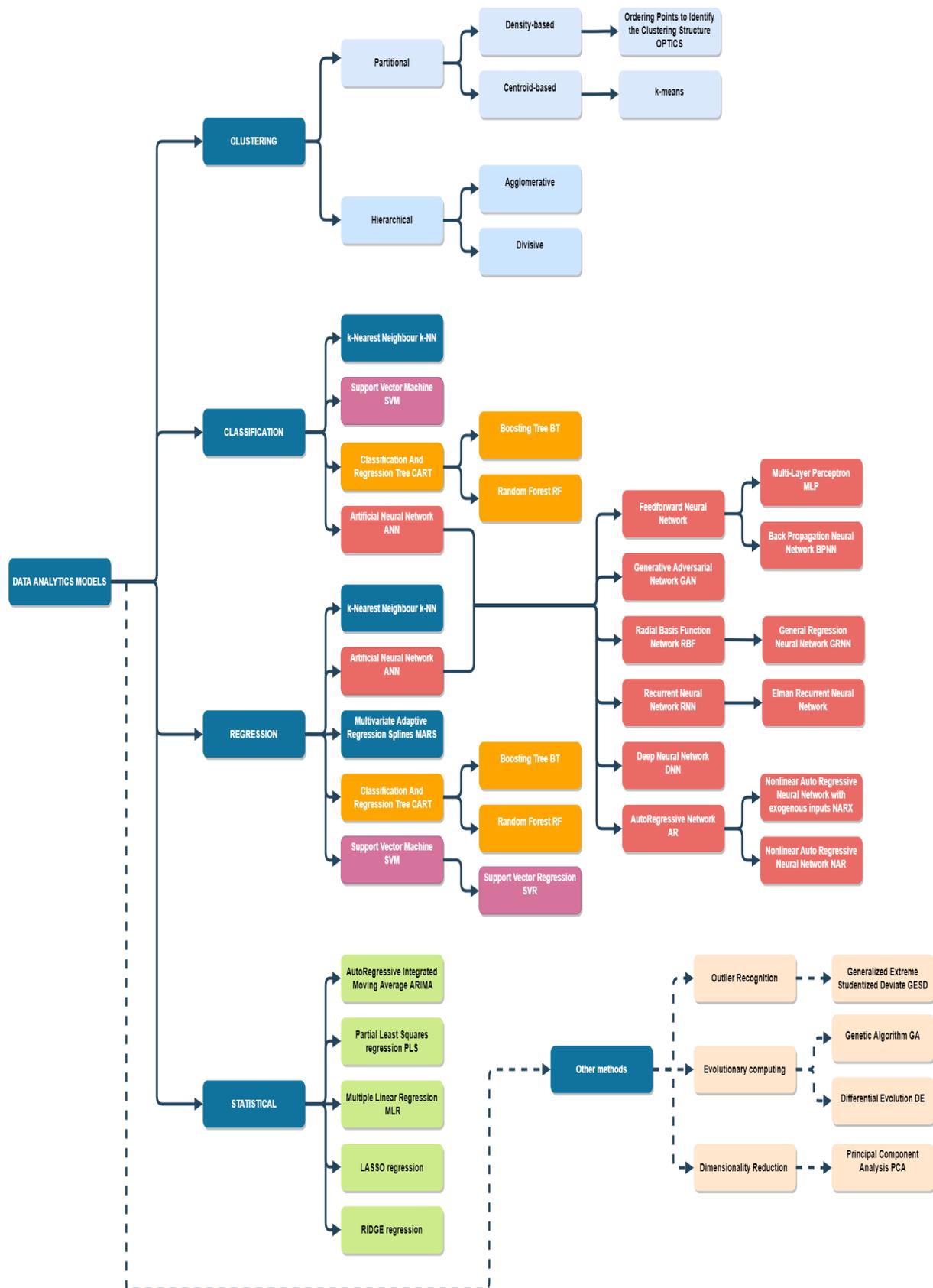


Figure 2.1: Diagram of the inverse models reviewed in section 2.

Another popular algorithm used both for classification and for regression purposes is Support Vector Machine (SVM). The idea, in this case, is to find a hyperplane able to distribute the data into two subspaces. This hyperplane is found by searching for the largest gap between the sets (7). An improved version of this algorithm called Support Vector Regression (SVR) has been developed to address regression problems. More information can be found in section 3.2.1.

Classification and regression trees are models obtained by iteratively subdividing the data space trying to fit a prediction model inside each subset (8). Depending on the kind of predicted variable, either discrete or continuous, they can be used both for regression and for classification. An in-depth analysis of these algorithms is reported in section 3.2.3.

K-Nearest Neighbour (KNN) is an algorithm employed both for regression and classification. It is based on the similarity of the objects of a group which have as neighbours some individuals with assumed similar properties (7). In classification problems the new samples are assigned to the nearest class using a set of previously classified points. For regression the result comes out as an average of the sample k-nearest neighbours.

Statistical models as Multiple Linear Regression, Autoregressive Integrated Moving Average, RIDGE, LASSO and Partial Least Squares regressions are a kind of inverse models used for regression purposes. They are all based on the concept of statistical modelling and they express the result with a mathematical equation relating some random variables to the predicted variable.

Finally, in Figure 2.1, there are some complementary methods which are not strictly speaking black-box methods, but which have been employed in the reviewed papers with different purposes, in order to improve the accuracy of the results or to simplify the algorithm. For example, Generalized Extreme Studentized Deviate (GESD) is a test used to identify the outliers in a univariate data set whose data are approximately normally distributed. Principal Component Analysis (PCA) is an algorithm used to perform a dimensionality reduction of a data set composed by a great quantity of variables which are not uncorrelated. This reduction is performed in such a way that preserves the variation of the data set. This procedure is done by converting the data set into a new one which is made of variables called principal components. Those are uncorrelated and are ordered by importance, so that in the first variables the majority of the data set variation is kept (9). Evolutionary computing is a branch of computer science which is made of algorithms inspired by the Darwinian principles of natural selection that in a group of individuals, called population, the environment can cause a pressure that leads to natural selection which brings to an increase of the population fitness. In mathematics, an objective function is maximized by creating a random group of candidate solutions and by using, as the population fitness measure, the

objective function itself (10). These algorithms can be used also to find the weights of each model in an ensemble, assigning the right coefficient to each solution, in order to maximize the prediction accuracy. The following is a presentation of some of the main topics related to energy prediction and it is organized in five sections depending on the main purpose of the papers.

2.1 Forecasting

In Figure 2.2 the methods reviewed in this section are classified depending on their purpose. The prediction models are the principal models which are used for estimating the value of a variable at a certain future time. The other models have different purposes, for instance some researchers use them to perform outlier recognition and removal, to decompose the time series making it easier to predict, to perform the dimensionality reduction of a data set, to select the optimal parameters for a prediction model or to find the right weights of an ensemble method.

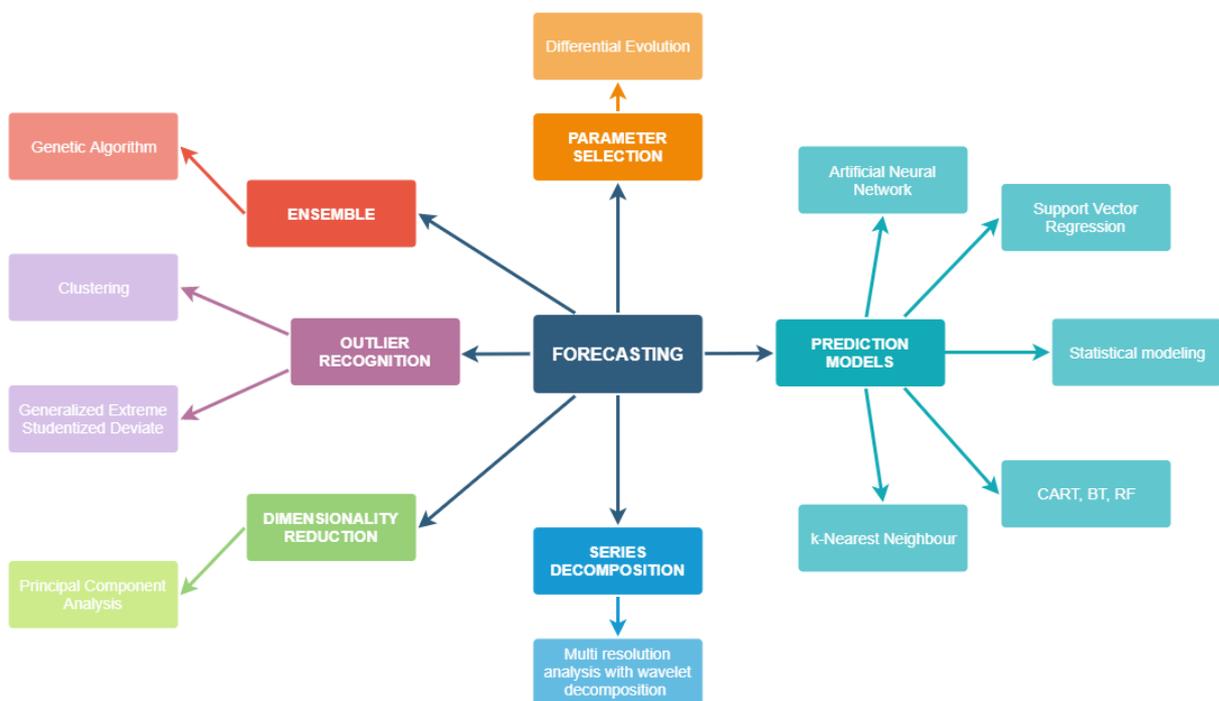


Figure 2.2: Diagram of the forecasting methods reviewed in section 2.1.

In (11), the authors demonstrated the validity of Nonlinear Auto Regressive Neural Networks (NAR) for monthly electricity energy consumption for 3, 6 and 12 months ahead, when there is only one time series involved. This model gives the opportunity to make a working analysis and diagnosis tool, for building energy demand, when the only data available are the past energy

consumptions. González and Zamarreño (2) developed 1-hour-ahead electric load forecasting in buildings using feedback (FB) artificial neural network with a short training period of 21 days. They predicted the external air temperature of the following hour and used it to forecast the electric load. The proposed method is easy to apply and can be used having at disposal only the atmospheric temperature and the load measurements. This study touches upon one of the most sensitive issues of AI-based methods, demonstrating that in some cases a quite limited number of data is sufficient to start making reasonably accurate predictions. Kelo and Dudul (12) showed the good performance of wavelet and Elman Recurrent Neural Networks (Elman RNN) for forecasting 1-day-ahead electrical power load under the influence of temperature. They added gaussian noise to the real temperature time series to verify that the predictions were still good enough. In (13), the authors coupled Artificial Neural Network (ANN) with Genetic Algorithm (GA) to avoid slow convergence and local minimum stagnation and provided a tool for daily energy consumption forecasting. In particular, Nonlinear Auto Regressive Neural Network (NAR), Nonlinear Auto Regressive Neural Network with exogenous inputs (NARX) and Elman Neural Network (ENN) are compared showing the better performance of ENN. Choueiki, Mount-campbell and Ahalt (14) demonstrated the superior performance of their “quasi optimal” neural network over a more traditional autoregressive integrated moving average (ARIMA) model for the STLF problem of hourly 24-hours-ahead load forecasting. They obtained, notably, a RMSE between 2.01% and 3.87%. As discussed in (15), for some application it can be useful to predict both half-hourly and daily electricity consumption. The authors developed a model, starting from one year of half-hourly data from an institutional building, using ν and ϵ Support Vector Regression (SVR) and selecting their parameters with Differential Evolution (DE). In this way daily and half-hourly energy consumption have been predicted with a single model without changing the parameters thanks to the DE which can assign the right weight to each SVR. The proposed tool showed better accuracy than single SVR with other evolutionary algorithms (Genetic Algorithm: GA, Particle Swarm Optimization: PSO). In (16), five different data-driven forecasting models including Multiple Regression (MR), Genetic Programming (GP), ANN, Deep Neural Network (DNN) and Support Vector Machine (SVM) have been compared. The task was to forecast the daily electrical energy consumption of an administration building starting from its historical values and from weather data. The work showed that the lowest general MAPE of 6% was generated by the ANN and the highest by the DNN that reached 11%. In any case, by aggregating the results to monthly values, each model forecasted electricity consumption of working days within a range of 3%. Chen, Tan and Song (17), offered a feasible method for one hour ahead energy prediction, based on the historical meter readings from a hotel. They used Multiresolution Wavelet Decomposition (MWD)

to remove random disturbances of the historical electric power series and illustrate the special periodic features and then they developed a model based on SVR. Their results pointed out that in an optimized SVR model, where all the parameters have been finely tuned, the MWD process seems to be redundant. Ahmad, Mourshed and Rezgui (18) compared the forecasting performance of ANN and Random Forest (RF) for 1-hour-ahead HVAC (Heating Ventilation and Air Conditioning) energy demand forecasting. The reported RMSE were 4.97% for ANN and 6.10% for RF. Despite of the fact that ANN performed better than RF in testing, RF could handle the missing values, since it is an ensemble-based algorithm. Both methods resulted to be suitable for predicting HVAC hourly energy demand with good accuracy.

Some researchers applied data-driven forecasting models to external air temperature because, as investigated in (2), it can be used to improve the accuracy in load forecasting. For instance in (19), Multi-Layer Perceptron (MLP) and Radial Basis Function (RBF) artificial neural network are used to predict external air temperature for the following 24 hours with better results than empirical equations. In particular, MLP showed better performance with respect to RBF artificial neural network. Yokoyama, Wakui and Satake (20) predicted the 3-hours-ahead energy demand, of a tertiary sector building, using an ANN which predicts the temperature and the relative humidity. The results were fed to a second ANN which forecasted the energy demand. Finally, global optimization and periodic differential operation were used to improve the forecasting accuracy.

Having at disposal on-line predictions of energy demand can lead to considerable energy savings, as discussed in section 0. Yang, Rivard and Zmeureanu (21) forecasted the 1-hour-ahead electrical energy demand of a chiller comparing a static ANN and two different types of adaptive ANNs (accumulative training and sliding window training). Adaptive ANNs use a training dataset that is constantly updated to adapt the network to unexpected changes of pattern: for this reason, they concluded that these models can be successfully employed for on-line building load prediction.

For some applications, having a good prediction of the daily load peak can be truly helpful. Fan, Xiao and Wang (22) predicted the 24-hours-ahead electricity load profile and the daily peak using an ensemble model generated via GA, which incorporates eight different machine-learning algorithms: multiple linear regression (MLR), ARIMA, SVR, RF, MLP, boosting tree (BT), multivariate adaptive regression splines (MARS), and kNN. With the ensemble model they reached a MAPE of 2.32% for the next-day energy consumption and of 2.85% for the load peak. For both predictions SVR and RF produced the most accurate results, but the ensemble model made each algorithm complement with another, leading to better generalization performance.

An important issue regarding HVAC systems management is represented by occupancy. Knowing the number of occupants, for each space and each time, can lead to large energy savings. For this purpose, in many cases, simplifying assumption are adopted, in order to make the prediction easier. Chen and Jiang (23) created a building occupancy model without any simplifying assumption for a room in an educational building. They used a model called Generative Adversarial Network (GAN), composed by a generative neural network and a discriminative one. The generative network produces an occupancy time series starting from random inputs while the discriminative network tries to distinguish the occupancy time series generated by the previous network and the real ones. The goal of the process is to generate occupancy time series that are indistinguishable from the real ones. Their outcomes demonstrated that the proposed model had better accuracy with respect to traditional ones (agent-based model ABM and inhomogeneous Markov chain IMC).

In some cases, the time series acquired as training data for a forecasting model can also be used for assessing some of the building properties, in order to improve the value of the prediction itself. In (24), the authors compared different methods for energy demand prediction and suggested a methodology for evaluating the thermal response time of buildings. The forecasts were then provided for the time ahead which was equal to the estimated building thermal response, in that case 40 minutes. The proposed models were Extreme Learning Machine (ELM), Back Propagation Neural Network (BPNN), SVR and MLR. The lowest RMSE equal to 3.824, in testing, was given by the ELM model optimized by feature selection.

Energy forecasting is a key factor for efficiently control the HVAC system serving a building. For this reason, many efforts have been done by the researchers to reconcile energy prediction and system control and optimization. Ben-Nakhi and Mahmoud (25) tried to optimize HVAC thermal energy storage in public buildings as well as office buildings using forecasts of hourly cooling load for the next day and total cooling load for entire day. They developed a model, based on a General Regression Neural Network (GRNN), which was fed with the external dry bulb air temperatures of the 24 hours before the one to be predicted. The authors stated that, using only these inputs, the controllers which would have used this method would have been easy to make and trustworthy. They obtained a coefficient of determination R^2 equal to 0.9552 for the hourly cooling load and of 0.949 and 0.964 respectively for the daily cooling load of N-S and E-W oriented zones. Zhao and Liu (26) developed a short term load forecasting model, to adjust the HVAC system to the actual load and enhance the operational safety. They made 1, 2, 3 and 24-hours-ahead load predictions, to detect the change of the cooling and heating load and to choose the operational strategy which leads to the most efficient management of the HVAC system. For predicting the load, the authors

used wavelet transform coupled with SVM and Partial Least Squares regression (PLS). The proposed model (wavelet-PLS-SVM) showed better accuracy than the single PLS, SVM and wavelet-SVM. The forecast error of the input weather variables was also considered: when the forecast error of temperature was 1.68°C and the forecast error of relative humidity was 11.7%, the Mean Absolute Relative Error (MARE) increased by 1.56%, as regards the 24-hours-ahead load prediction. The forecasts at 1, 2 or 3 hours ahead were less influenced by the forecast of the weather parameters because of their higher precision. In order to provide good internal temperature prediction, for improving the performance of temperature controllers, Al-Obeidat, Spencer and Alfandi (27) created a model capable of predicting 48-hours-ahead internal temperature with a time granularity of 15 minutes. For doing this they used ridge and lasso regression having the following inputs: six environmental factors, including inside and outside temperature, inside and outside humidity, windspeed and rain; 26 circuits; 4 motion detectors; 1 light switch. Each forecast was very close to its real value since the predicted internal temperatures for the next two days were always within 1.8°C from the actual ones.

2.2 *Model Predictive Control (MPC)*

As mentioned in section 2.1, the possibility of knowing the future energy consumption linked to the building can facilitate the smart control of the technical building systems as well as enhance the operational safety. Many applications of the forecasting tools to the control systems, known as Model Predictive Control, have been developed with different kinds of prediction methods. In any case, the energy savings related to the implementation of MPC are substantial and deserve to be mentioned as one of the most promising uses of energy forecasting. Moreover, as stated by Lauro et al. (28) these approaches do not require a huge number of data and can lead to significantly high energy savings. As an example of the considerable efficiency improvement that can be achieved with MPC, Pospíšil, Špiláček and Kudela (29) calculated the potential of such control for increasing the Seasonal Coefficient of Performance (SCOP) of an air-to-water heat pump. They gathered 24 and 48 hours-ahead external air temperature predictions and, starting from these information, they simulated a heat pump that was operated during the hours of the day with highest outdoor air temperature, to satisfy the building heating energy need and to fill a heat accumulator which would have been used during the remaining hours, given that, if it was empty, the heat pump would have been started despite the low outdoor temperature. Their results proved that the SCOP of the heat pump increased, with respect to the one without predictive control, of 19% with 24 hours temperature prediction and of 23% with 48 hours temperature prediction. The increased SCOP value corresponded to around 20% decrease in yearly power consumption and the authors stated

that the 48-hours-ahead predictive control could lead to a significant reduction of the starts of the heat pump, corresponding to an increase in its lifetime.

Figure 2.3 represent a diagram which summarize the models used in the reviewed papers to implement a Model Predictive Control system depending on the main objectives proposed by the authors.

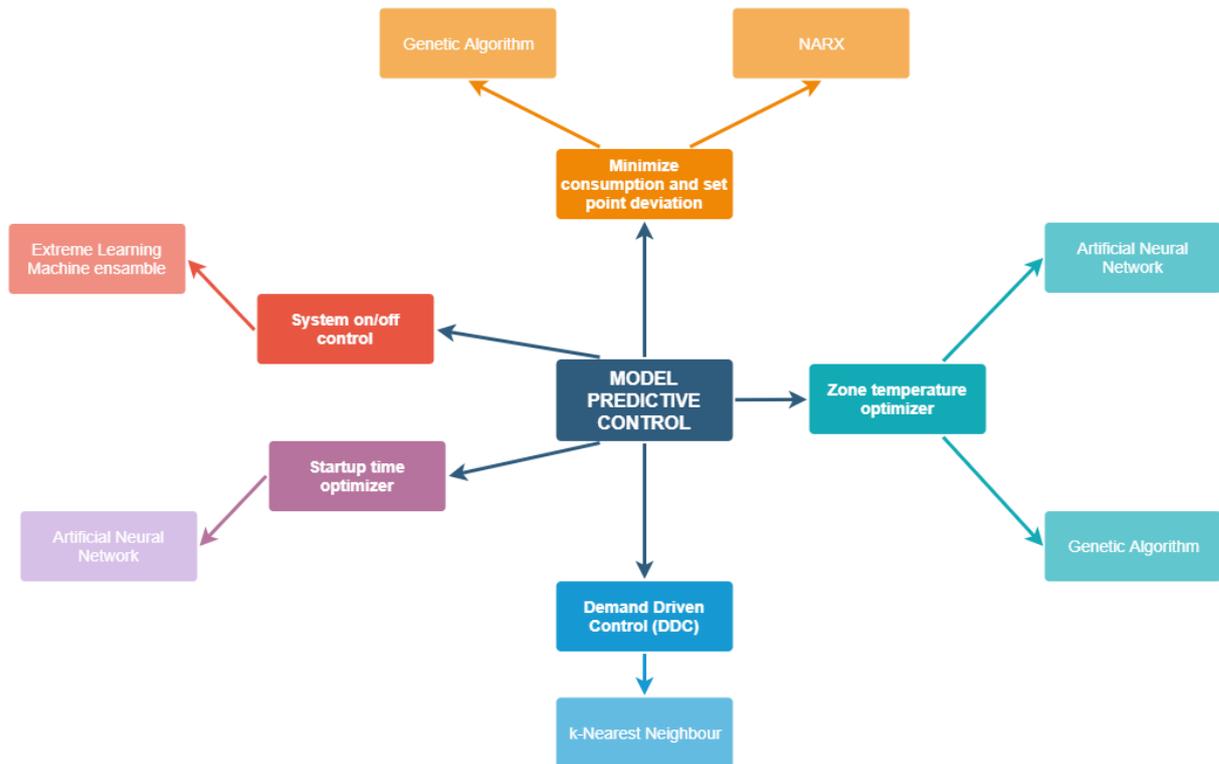


Figure 2.3: Diagram of the MPC methods reviewed in section 2.2.

In (30), Reynolds et al. developed a zone-level heating set-point scheduler which minimized the energy consumption using hourly 24 hours ahead energy consumption and indoor temperature predictions. An ANN was used to predict the energy consumption and the indoor temperature, then a GA was used to optimize the zone temperature set-point for one day, minimizing the energy consumption while guaranteeing suitable indoor environmental conditions. As an additional case the authors considered also a MPC where the set-point schedule was uploaded each hour for the next 24 hours so that the system could consider also the feedback from the building. The outcomes showed that the system could achieve energy savings around 25% with electricity flat tariff, either with day-ahead schedule or with MPC, and energy savings of 23.31% and 21.28% with electricity time of use tariff, respectively with day-ahead schedule and MPC. The corresponding money savings with electricity time of use tariff was estimated as 27.94 % for day-ahead schedule and

27.26% for MPC. The proposed solution required a small amount of additional hardware and brought to significant savings with the advantage of simulating the building behaviour with a simple ANN model that is fast to run.

Predictive control can also be used to determine the right time at which a system should be started-up, it is the case discussed in (31) where the authors implemented a MPC in a commercial building, to determine the optimal time at which turning on the boiler leads to achieve the target temperature at 8:00 am. They constructed an ANN which received as inputs the average internal temperature, the external temperature and the water heating system temperature and gave as output the time required, in quarters of an hour, for conditioning the building at 20°C at 8:00 am. The results showed that the gas energy consumption reduced by 19.69% with respect to on-off regulated by the energy manager and were better than those reached in similar studies proposing control strategies based on commercial BEMS (Building Energy Management System).

Another possible application of MPC is described in (32), where Costanzo et al. developed a model which determined if the system had to be switched on or off, to control indoor temperature. The behaviour of indoor temperature was predicted with an ELM ensemble model and the control policy was evaluated three times a day, as new weather forecasts arrived, and dispatched on 5 min basis. This framework led to reach a performance of 90% of a mathematical optimum in around 20 days. A different outline was proposed in (33), where Erfani, Rajabi-Ghahnaviyeh and Boroushaki optimized a multi-zone Air Handling Unit's (AHU) energy consumption implementing a Nonlinear Model Predictive Control (NMPC) which minimized the energy consumption and the deviation from the set-point temperature. The AHU had three zones and served an educational building, while the data was gathered each 17 minutes. The authors set the control horizon and the prediction horizon to 119 minutes and used NARX and discrete GA whose outputs were chilled water flow rate, damper position of the zones and air handling unit's status. The model brought to a reduction of consumption of NMPC with respect to no control of 55.1% on electricity and 43.7% on gas and to a reduction of consumption of NMPC with respect to on/off control of 18.5% on electricity and 17.4% on gas. The future set-point started 2 hours before the deviation between the internal temperature and the set-point took place. This demonstrated the ability of the model to act in advance, taking into consideration the thermal inertia of the building.

In (34) the authors developed a Demand driven Cooling Control (DCC) to increase the efficiency of HVAC systems in accommodating occupants' behavior in real time. The information about occupancy learned by the algorithms were used to control room setpoints in real-time as well as the cooling system. Subsequently, the Rule Based Control (RBC) was allowed to change the temperature set-point with three different possibilities plus a set-back. The experiment was carried

out in eleven rooms of a commercial building form which a global dataset was obtained. Local training datasets were the result of a clustering analysis of the global training dataset on the specified number of occupancy patterns in individual rooms, so that each training dataset corresponded to a kind of occupancy pattern in the global one. The time of the occupant's next presence and total presence duration in the remaining day were the outputs of the kNN. The achieved energy savings were different, depending on the final use of each monitored room: in the multi-person offices they reached 7% because of the high occupancy; in the single person offices, with medium occupancy rates, they were equal to 21%; in the meeting room, they were 52%; globally the DCC saved 21% of the sensible cooling energy. Moreover, the proposed method was able to guarantee short-presence detection since it took 12 minutes to activate the cooling system in a room. In this way unnecessary activations of the cooling system were avoided.

2.3 Demand response

Demand response (DR) can be defined as a group of programs “which aim to modify the demand patterns for electricity by encouraging its use during peak generation and discouraging its use at times when the load on the grid is highest” (35). This goal is achieved by acting directly on the demand side, as a consequence of the variation of the price at which electrical energy is sold or to receive incentives. Figure 2.4 represents the diagram of the demand response papers reviewed in this section.

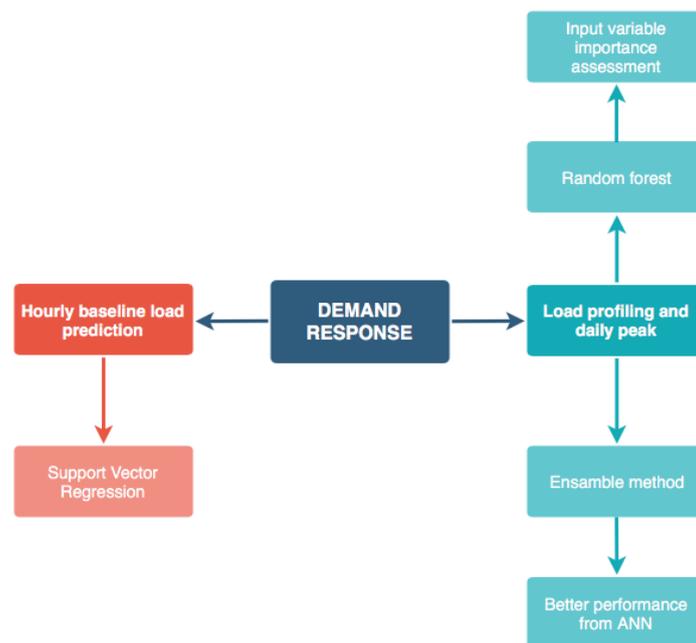


Figure 2.4: Diagram of the demand response methods reviewed in section 2.3.

In this framework data-driven forecasting can find one of its more promising applications. In fact, in order to implement an efficient DR program an estimate of the energy that would have been consumed without such program is needed. This guess is called DR baseline and is used to calculate the demand response performance, as the difference between the baseline and actual load during a DR event. For instance, Chae, Horesh, Hwang and Lee (36) predicted the 24-hours-ahead electricity load profile and the daily peak for a tertiary building complex, using weather data acquired every three hours from a local weather station. They used random forest to assess the input variable importance and then they compared the results obtained by nine different data-driven models. The best model resulted to be ANN and the authors suggested that it could be used to control the daily load profile, to control demand response for money savings and to control the priorities of the end-used energy consumption. The issue of providing a good DR baseline is also the aim of (37), in which a SVR model was used to provide the prediction of hourly load for the 8 working hours of the following day, in four office buildings, in the absence of a demand response program. The obtained results showed better performance than the ones obtained with the traditional methods.

2.4 *Fault detection*

Another possible field of application of the AI-based methods is fault detection. In fact, it is possible to identify the kind of fault as well as its severity so to be able to intervene or at least report it or stop the system operation. Figure 2.5 shows the algorithms used in the reviewed papers linked with their functions.

In(38), the authors developed a fault diagnosis method for photovoltaic (PV) arrays for improving their reliability, efficiency and safety. They used both data acquired in laboratory and from Simulink modelling approach, then Eagle strategy-based Hybrid Adaptive Nelder-Mead Simplex algorithm (EHA-NMS) was adopted to extract the model parameters based on the single diode model structure. The measured or simulated I-V curves were fed as inputs to the Kernel based Extreme Learning Machine (KELM) whose parameter were adjusted according to another model called Nelder-Mead Simplex (NMS). The KELM itself had 5 different outputs, corresponding to the normal condition plus the 4 considered types of fault. The reached fault detection testing accuracy on 100 random runs, with mixed simulated and laboratory data, was equal to 98.88%.



Figure 2.5: Diagram of the fault detection methods reviewed in section 2.4.

Li, Zhou, Hu and Spanos (39) investigated the performance of a tree-structured learning method for Fault Detection and Diagnosis (FDD) of a building cooling system, with the aim of detecting both the fault type and its severity. Wavelet-based De-noising was used to remove the data periodic patterns, while Modified Thompson's Tau method was adopted for outlier detection. The chosen fault detection model was the Tree-structured Fault Dependence Kernel method (TFDK) which had 29 nodes: the normal situation corresponded to node 1, and the seven considered types of faults, with their four severity levels, corresponded to nodes 2 to 29. TFDK showed better results compared with other state-of-the-art methods: Multi-class SVM (MSVM) with RBF kernel, Decision Tree (DT), Neural Network (NN), Ada Boost (AB), Quadratic Discriminant Analysis (QDA) and Logistic Regression (LR). The proposed model brought to 1.49% - 9.19% improvement in accuracy and 10.69% - 75% decrease in testing cost, with respect to the method at the second place.

Data-driven based fault detection can also be addressed to detect AHU sensors faults. It is the case of (40), where supply air temperature sensor, supply and return chilled water temperature sensors, supply air flow rate meter and supply water flow rate meter were monitored using PCA

and OPTICS (Ordering Points to Identify the Clustering Structure) clustering algorithm. The proposed model was able to detect single and multiple AHU sensor faults by identifying the spatial and temporal separation of the measured data.

Presenting an effective diagnostics tool to improve the energy efficiency and thermal comfort of buildings through removing faults was the task of (41), where the authors analysed the data obtained from an HVAC simulator based on TRNSYS. They decided to use two BPNNs, one having supply air temperature as output and the other as an auxiliary network. Wavelet analysis was used to process the raw data, removing useless and disturbing ones, while a sensitivity analysis was carried out on the input parameters, to choose the right output for the auxiliary BPNN: the chosen one was return chilled water temperature. Finally, PCA was used to determine the weighting factors for the two BPNNs. Clustering analysis was used to create a library of faults, while the BPNNs were used to determine if a fault was occurring, measuring the relative error with respect to the predicted data. Clustering analysis was also needed to determine which was the kind of fault and to add new faults to the library. The detection threshold was determined by analysing the false alarm ratio and set at 2%. The results showed that examining 6 different types of fault, the false alarm ratio was between 0.6 and 7.7%, while the missing alarm ratio was between 0 and 8.3%.

2.5 *Benchmarking*

As stated by Li, Han and Xu (42), since building energy performance always tends to decrease during time, a good energy benchmarking system would be extremely useful to keep a record of the energy consumption and to observe and point out the anomalous ones. Chung (43) notes that energy consumption benchmarking is carried out by comparing the consumption of several similar buildings, both to encourage poor-performers to increase their efficiency and to furnish an instrument to the regulators which can be released to the media to increase the people awareness and concern about the subject. Capozzoli et al. (44) claim that energy benchmarking is needed as a tool for regulators which can be used to assess the good or bad performance of a building and to establish feasible goals to increase the energy performance or to point out anomalous consumptions. Figure 2.6 shows the purposes of the reviewed papers and the related algorithms.

In (45), Roth and Rajagopal developed a building energy use benchmarking system to rank buildings. They started from a dataset of a thousand of buildings with 57 types of features and 2 years of data. Their framework consisted in creating a quantile regression model to identify the high and low performing buildings.



Figure 2.6: Diagram of the benchmarking methods reviewed in section 2.5.

They also used a cumulative distribution function (CDF), created by the aggregation of the totality of the models, in order to rank the buildings with a so-called QuantRank score. Finally, they analysed the influence plots of several quantities: for instance, they studied the influence of cooling degree days (CDD) on building energy consumption. Their approach was able to give a probabilistic interpretation of building energy consumption, taking into account the influence of different external factors (such as CDD, HDD, number of employees, etc.). In particular, they found out that CDD had a strong effect on bad performing buildings. Furthermore, their work proved to be versatile since it can incorporate unsupervised algorithms, nonlinear modelling dynamics as well as variable selection. Building owners could use this method to analyse their investments and to determine, for instance, whether increasing the number of employees would lead to a benefit or to an added cost concerning building energy consumption.

In (46), the authors developed a benchmarking methodology to assess the energy performance of a building starting from the shape of its electrical load. With such a purpose they created a tool which enabled to identify potential operational improvements, retrofits and savings and to reduce the energy supply cost. The process required a clustering analysis of the electrical load data obtained

by thousands of small or medium commercial buildings, sampled at 15-minutes time intervals. The clustering analysis was carried out on normalized profiles and, for instance, the obtained centroids were used to determine the starting and ending operating hours. The main merit of this work was to overcome the issue of the common Energy Performance Indicators (EPI), which are only able to consider the total or global building energy consumption, or their intensity, but which cannot give any information about the changes in the energy consumption patterns. In order to overcome this issue, the proposed model created a statistical method which considered 24 hours of power shape benchmarking and made a comparison between different buildings.

A model which was able to analyse building energy consumption data and to detect the ones with similar “temporal energy performance patterns” was developed in (47). The authors started from 100 types of features taken from thousands of buildings and performed a missing values and outliers removal. Subsequently, they carried out a k-means clustering analysis with normalized profiles and silhouette score and Dunn index were used to assess the right k value. Furthermore, they applied a logistic regression to predict the cluster to which each profile belonged. In the end two temporal patterns of building energy performance have been recognised: an increasing and a decreasing one. Energy consumption decrease showed to be mainly related to office buildings, which was larger, newer, and with higher-value and which presented improvements for what concerns the consumption in the years 2011-2016. The authors also suggested that this study could be used by the authorities, as a basis of suitable and more fair energy policies.

Capozzoli et al. (44) proposed a methodology to perform a benchmarking analysis addressed, in particular, to buildings with heterogeneous datasets. They used energy consumption data from 100 outpatient Healthcare Centres and building features collected by energy auditing. Firstly, they performed a dataset segmentation by means of a CART algorithm, to clarify the individual classes, then they carried out a second segmentation according to the operating time by means of expert analysis. Three Linear Mixed Effects Model (LMEM) and three distributions of energy consumption was performed for each identified class. The input variables used in LMEM was selected by correlation analysis. Then to assess the frequency distribution of the outcomes of the model, a Monte Carlo simulation was used, and the input variables were varied with a perturbation, each one in its possible variation interval. In this way the authors obtained a variance of the model of 96% with a MAPE around 15%. Thanks to this framework the end-user could have at disposal the median of the frequency distribution, which has been simulated, and by using that information he could obtain a value that can be used as a reference. The authors also stated that with this tool, having at disposal also the information about the distribution, the user would be able to select the parameter which best reflects its needs, strongly enhancing the energy performance of his building.

Chapter 3:

Methods

3 Data mining and forecasting methods

As discussed in section 1, one of the most promising ways of improving energy savings in buildings is the installation of a Building Automation and Control System. Furthermore, as demonstrated by the numerous examples presented in section 2, many studies have focused on technical building systems control, fault detection and demand response. As a consequence of the development of BACS and, similarly, because of the other applications described in the previous section, a large amount of data regarding building operation and weather conditions are available and have to be managed. Hence there is the opportunity to extract some knowledge from these data and, in order to do so, some AI-based techniques have arisen. One of the most auspicious solutions is data mining (DM) “a multi-disciplinary subject, integrating techniques from statistics, machine learning, artificial intelligence, high performance computing and etc” (48). Being AI-based methods, the DM algorithms require a certain amount of data to learn some information and extract knowledge. On the other hand, robust forecasting tools are required to develop a STLF framework able to improve the efficiency of BACS. For this reasons in the current section some of the mostly used data mining and forecasting models are presented both from a mathematical and from a practical point of view.

3.1 *Data mining methods*

DM can be subdivided into supervised and unsupervised analytics depending on the role played by the data scientist implementing the model. In supervised analytics the expert directs the algorithm towards the outcomes it should obtain, while in unsupervised one it is the algorithm itself that makes the computer learn some information about the data, without any human guide.

Unsupervised DM has the aim of discovering the hidden information that are present in raw data, for instance patterns, connections and relations among them. It employs a so called forward approach, in which no data is taken as output since they are all considered as inputs. In this way, the algorithm autonomously selects the target knowledge to extract. Regardless of the subjectivity and the challenge that can derive from the application of such models, there are several literature examples of their use in relation to building data (49). In the following sections, some of the most popular unsupervised DM algorithms are presented.

3.1.1 Clustering

Clustering analysis is an unsupervised classification model which has the aim of grouping observations that have similar patterns, separating them from others that have similarities among them. In a two-dimensional data frame each column corresponds to a variable that concerns the studied phenomenon, while each row represents an observation. As an unsupervised classifier, a clustering algorithm tags a class label to an object without any previous relation between the two have been imposed by the data scientist.

Starting from the set of data S , the algorithm creates a number of subsets which have the following structure

$$S_1 \cap S_2 \cap S_3, \dots, \cap S_k = \emptyset \quad (3.1)$$

In such a way any observation will belong to only one subset and will be excluded from all the others. In Figure 3.1 a simple example of clustering is reported. As can be seen, the data points have been divided in two different clusters called A and B.

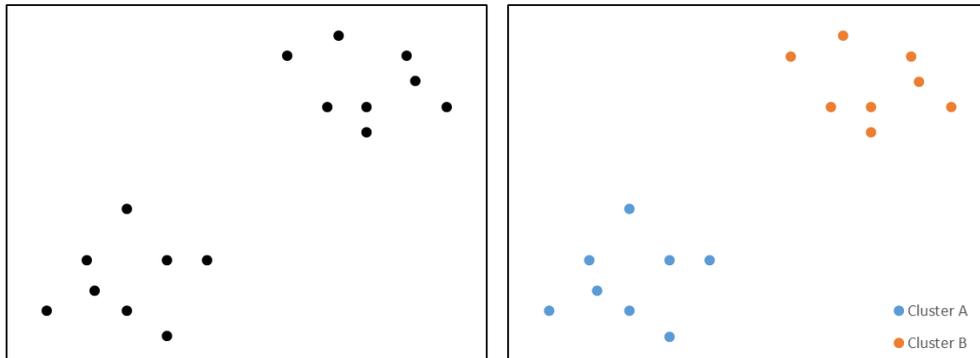


Figure 3.1: Simple clustering example.

Clustering algorithms are usually divided into hierarchical and partitional (50). Hierarchical clustering can be subdivided again into agglomerative or divisive clustering. Agglomerative one follows an approach called bottom-up, that consists in starting from single observations and join them in clusters that get always bigger. The process is ended when all the elements are in a single cluster or when particular conditions are satisfied. On the other hand, divisive clustering follows a top-down approach, which starts from a cluster containing all the observations and divide it into smaller ones until each element belongs to a cluster composed only by himself or until some

conditions are met. Finally, hierarchical clustering can be divided in three categories depending on the similarity measure it employs (50):

- Single-linkage clustering: also called nearest neighbour method, makes the link between two clusters using two elements, each one belonging to a different cluster, that are the closest ones. The distance between the clusters A and B can be defined as

$$\min\{dist(a, b) : a \in A, b \in B\} \quad (3.2)$$

- Complete-linkage clustering: also called furthest neighbour method, is similar to the previous one but it considers the two elements with the largest distance, as follows

$$\max\{dist(a, b) : a \in A, b \in B\} \quad (3.3)$$

- Average-linkage clustering: also called minimum variance method, employs a distance which is determined as the average distance of each element of the cluster A, to each element of the cluster B, as reported below

$$\frac{1}{|A||B|} \sum_{a \in A} \sum_{b \in B} dist(a, b) \quad (3.4)$$

Partitional clustering follows a quite different approach with respect to hierarchical one. In fact, it divides the observations into a number k of clusters, on the basis of a criterion function. The most used criterion function is the Euclidean distance (51), which calculates the distance between an element and each cluster and assign it to the one which has the minimum distance. One of the most used partitional clustering algorithms is the k -means clustering. In this algorithm the data scientist selects a number k of clusters that he wants to be classified. Then k random points are selected as cluster centroids and each point is assigned to the cluster with nearest centroid. Once every observation has been classified the k centroids are recomputed and the points are reassigned iteratively, until the centroids does not move anymore. The objective function J can be written as in Equation 3.5, where $\|x_i^{(j)} - c_j\|^2$ is a distance measure between an observation and a cluster centroid respectively.

$$\text{Minimize } J = \sum_{j=1}^k \sum_{i=1}^N \|x_i^{(j)} - c_j\|^2 \quad (3.5)$$

3.1.2 Association rule mining

Association rule mining (ARM) is an algorithm that allows to recognise all associations and correlations among attributes. The method provides a list of association rules between elements that are likely to be correlated. An association rule can be written as $A \rightarrow B$, in which A and B are sets of elements such that $A \cap B = \emptyset$. A is called premise or antecedent and B is called consequence. The association rule means that when A happens B will happen too.

In order to be more explicative, let's consider the example reported in (52). The authors consider 100 occupants living in 100 rooms of the same building and equipped with a computer and a table lamp, one each. Let's assume that 40 occupants turn on the computers while 20 occupants turn on the lamps. Now let's consider 10 occupants which turned on both their computer and their lamps, they represent the 10% of all the building occupants and the 25% of the ones that turned the computers on. The association rule that express the tendency of occupants who turn on the computer to turn on also the table lamp is expressed as follows

$$\textit{turn on computer} \rightarrow \textit{turn on table lamp} \quad (3.6)$$

$$[\textit{support} = 10\%, \textit{confidence} = 25\%]$$

ARM consists in finding all the association rules of a given dataset that accomplish a user defined criterion of support and confidence. The support expresses the validity of the association rule and is expressed as

$$\textit{support}(A \rightarrow B) = P(A \cup B) \quad (3.7)$$

The confidence, by contrast, is a measure of the certainty of the association rule and is given by

$$\textit{confidence}(A \rightarrow B) = P(B|A) \quad (3.8)$$

A third common indicator, similar to confidence, is the lift which can be expressed as follows

$$\textit{lift}(A \rightarrow B) = \frac{P(A \cup B)}{P(A)P(B)} = \frac{P(B|A)}{P(B)} \quad (3.9)$$

A lift value greater than 1 corresponds to a positive correlation, while a value smaller than 1 stands for a negative correlation. If the lift is equal to 1 there is no correlation between A and B, thus their occurrence is independent.

3.2 *Forecasting methods*

Forecasting building energy consumption is one of the key features to have good results in building systems control, fault detection, demand response and energy benchmarking. In this section some of the most common forecasting techniques are described, including supervised black-box models and statistical ones.

Supervised DM is actually more known than unsupervised one in the building industry (48). The aim of such methods is describing the relation that occurs between inputs and outputs. The structure of supervised DM algorithms requests that the possible outputs have been already presented by the data scientist and, therefore, that they are already known: this is the so called backward approach. Moreover, the data used to train the model must be already labelled and consistent with the results one wants to obtain. For instance, when an algorithm has to distinguish the image of a dog from the image of a cat, it has to be trained with different images that have already been correctly labelled as images of dogs or images of cats.

As a preliminary consideration it must be said that when data scientists have at disposal a set of data it is common use to split it into two subsets which are called training and testing datasets. For instance, in (13,19,22) the authors use 70% of data for training and 30% for testing while in (25) the split is done as 80% for training and 20% for testing. The training data are used to fit the model which learns from them, instead, the testing data are used to evaluate the model performance on data it has never seen before so that are more similar to the ones it could face in case of real-time use.

3.2.1 Support vector regression

Support vector regression SVR is a supervised algorithm and a generalised form of support vector machine (SVM) which is a model developed for solving binary classification problems. The mathematical model will be described in a row starting from a one-dimensional example. The complete mathematical demonstration can be found in (53).

The simplest SVR model is the one-dimensional linear one which is represented in Figure 3.2. As can be seen from the abovementioned figure, the SVR algorithm tries to approximate a given function using an optimization problem attempting to identify a tube that is the tightest one around the surface. The minimized quantity is the prediction error, which is basically the difference between predicted and real points.

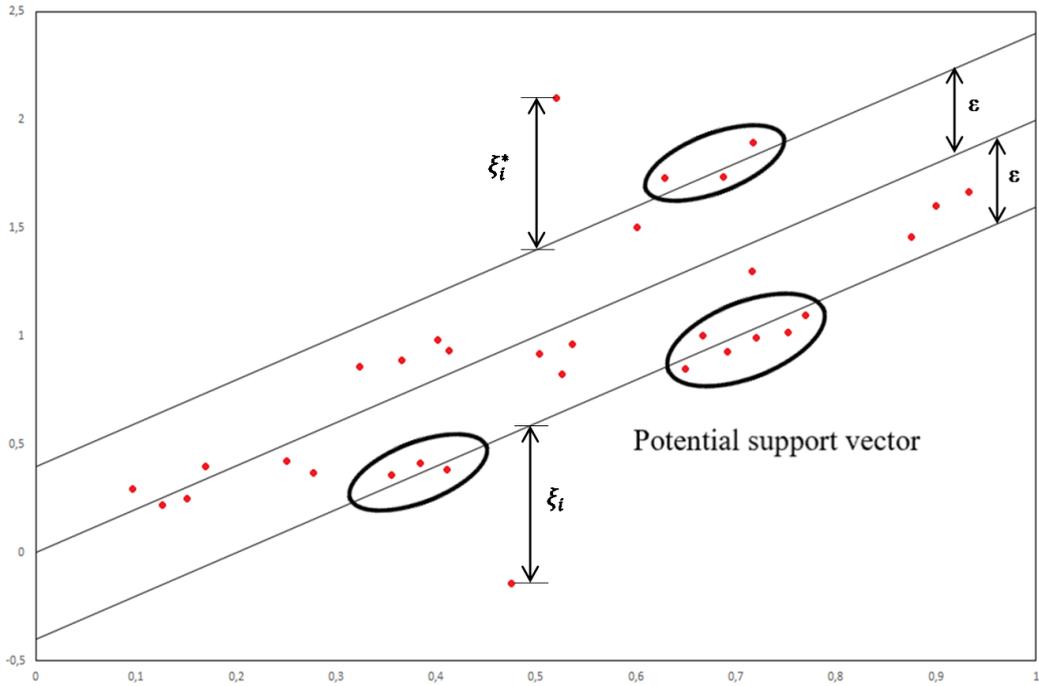


Figure 3.2: Linear support vector regression in one dimension. Figure taken from (52) with changes.

The target function, for multi-dimensional data, is called multivariate regression and is reported in Equation 3.10

$$f(x) = w^T x + b \quad x, w \in \mathbb{R}^{M+1} \quad (3.10)$$

The target function of the optimization problem is written as

$$\min_w \frac{1}{2} \|w\|^2 \quad (3.11)$$

Where $\|w\|$ is the magnitude of the vector which is normal to the surface to approximate. This magnitude can control the flatness of the solution, if it is too small the solution can have large discrepancy with respect to the desired outcomes and lead to underfitting, while if it is too high the model could overfit the solution.

Overfitting is one of the problems that can occur in developing a DM model. When the algorithm models too closely the training data, it may happen that it has also learned their noise and their details. In this case, when trying to apply the model to new data, it will lead to very poor results since it has lost its generalisation ability. Underfitting, for its part, is referred to the behaviour of a

model that is not able to generalise to data that it has never seen before and, moreover, that cannot produce good results even with training data.

To minimize the prediction error and solve the optimization problem, SVR algorithms adopt a so called ϵ -insensitive loss function. It means that the model penalizes all the forecasts that have a distance from the desired outcome which is higher than ϵ . By looking at Figure 3.2, it can be seen that ϵ determines the width of the boundary of the tube. If ϵ decreases the tube becomes narrower and the number of points around its boundary increases, leading to more support vectors. On the contrary, if ϵ increases the tube becomes wider and the number of points around its boundary decreases, in a way that determines a smaller number of support vectors. Several loss functions can be chosen, including linear, quadratic, and Huber ϵ . By looking once more at Figure 3.2 one could notice that a soft-margin approach is adopted, using the slack variables ξ and ξ^* to identify the outliers.

The final optimization problem can be derived considering Equation 3.11 and the chosen loss function and it is illustrated in Equation 3.12

$$\min_w \frac{1}{2} \|w\|^2 + C \sum_{i=1}^N (\xi_i + \xi_i^*) \quad (3.12)$$

$$y_i - w^T x_i \leq \epsilon + \xi_i^* \quad i = 1 \dots N$$

$$w^T x_i - y_i \leq \epsilon + \xi_i \quad i = 1 \dots N$$

$$\xi_i, \xi_i^* \geq 0 \quad i = 1 \dots N$$

Where C is a regularization term, which is a user-selected parameter that gives more weight to the minimization of the error when becoming larger.

This optimization problem can be solved by finding its Lagrangian and then taking its partial derivatives and setting them to zero. As stated by Awad and Khanna (53), the Lagrange multipliers which equal zero can be identified with the data inside the tube, the support vectors, instead, have Lagrange multipliers which are not zero. The approximation of the function, in which α and α^* are Lagrange multipliers, can be written as

$$f(x) = \sum_{i=1}^{N_{SV}} (\alpha_i^* - \alpha_i) x_i^T \quad x, \alpha_i, \alpha_i^* \in [0, C] \quad (3.13)$$

All the formulas seen so far have been written assuming that the function $f(x)$ to be approximated was linear. If the function is nonlinear, as it happens in the majority of cases and, of course when

it comes to predict power consumption or weather variables, it can be useful to map the data into a kernel space, a higher dimensional space to facilitate the computation. This process is referred to as “kernel trick”(54). In this way the final approximation of the function can be written as

$$f(x) = \sum_{i=1}^{N_{SV}} (\alpha_i^* - \alpha_i) k(x_i, x) \quad (3.14)$$

Where $k(\cdot)$ is the kernel, defined in Equation 3.15, in which $\varphi(\cdot)$ is the transformation to the kernel space

$$k(x_i, x) = \varphi(x_i) \cdot \varphi(x) \quad (3.15)$$

All the details of the demonstration can be found in (53).

3.2.2 Artificial neural network

Neural networks are a type of supervised algorithms, inspired from the functioning of human brain. In such a way, neurons are replicated, in a simplified manner, and represent the base unit for the neural network mathematical model as shown in Figure 3.3.

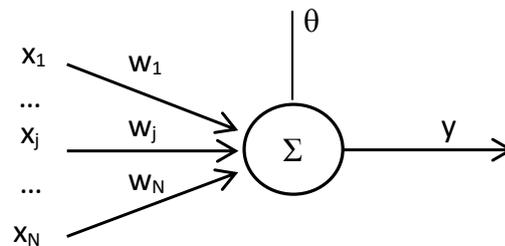


Figure 3.3: Neuron or perceptron.

The neurons or perceptrons are grouped in layers and connected with the ones in the other layers through links. The simplest ANN structure one can think of is a three-layered neural network with an input layer a hidden layer and an output layer. The links connecting perceptrons are called synaptic weights and the training process has the aim of calculating these weights (55).

For the sake of simplicity, let's consider a three-layered neural network with N input neurons, one hidden neuron and one output neuron. The total input to the perceptron will be the weighted

sum of the outputs arriving from each of the connected units to which is added a bias θ (56). Its mathematical expression is reported in Equation 3.16, where w_j are the synaptic weights and x_j are the inputs to the considered neuron, as shown in Figure 3.3.

$$s(t) = \sum_{j=1}^N w_j(t)x_j(t) + \theta(t) \quad (3.16)$$

When the synaptic weight w_j is positive it can be interpreted as an excitation of the neuron, while if it is negative it can be seen as an inhibition. At this point a function F able to combine the current activation $y(t)$ and the total input to the neuron $s(t)$ is needed. This function will give the next value of the activation as shown in Equation 3.17.

$$y(t + 1) = F(y(t), s(t)) \quad (3.17)$$

This function, called activation function, is usually a nondecreasing function of the only total input s to the perceptron (56). One of the most used ones, is the sigmoid function, whose expression is reported in Equation 3.18 and whose graph is shown in Figure 3.4.

$$F(s) = \frac{1}{1+e^{-s}} \quad (3.18)$$

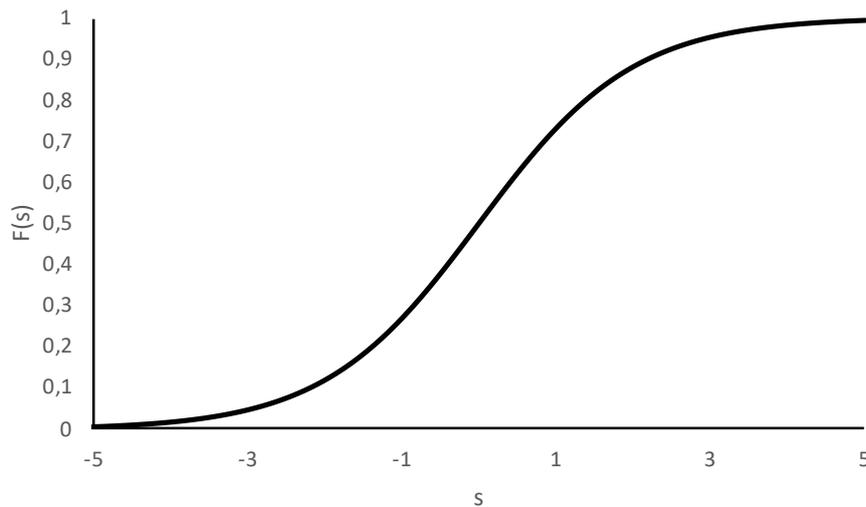


Figure 3.4: Sigmoid activation function.

Other types of activation functions can be used, including linear and threshold or Heaviside function, whose expression can be found in Equation 3.19

$$F(s) = \begin{cases} 1 & \text{if } s > 0 \\ -1 & \text{if } s \leq 0 \end{cases} \quad (3.19)$$

This specific activation function is used for classification purposes, to decide whether an object belongs or not to a specific class. The convergence of the learning process is reached using the perceptron learning rules which can be summarized, for a classification network, in the four following steps (56):

- 1) The process starts using random weights w_j
- 2) An input vector \mathbf{x} of observations is selected from the training data
- 3) If the network result is different from the desired one, hence $y \neq d(\mathbf{x})$, where d is the desired output, then the network weights are modified by a $\Delta w = d(\mathbf{x})x_j$ and the bias is set to $d(\mathbf{x})$
- 4) The process is repeated starting from point 2

If the network result is correct the weights are not modified, and the bias is set to zero.

To overcome the limitation linked to such a structure multi-layer feed-forward neural networks are employed. They consist in networks composed by layers which are composed, in turn, by neurons which receive signals from the upwards layer and send signals to the downwards one. The perceptrons of the same layer are not connected with each other. This kind of network use non-linear activation functions like the sigmoid one.

The least means square (LMS) error function, shown in Equation 3.20, is minimized with a method called gradient descent.

$$E = \frac{1}{2} \sum_p (d^p - y^p)^2 \quad (3.20)$$

The index p indicates a specific set of input pattern, d is the desired output and y the network output. The gradient descent changes the j -th weight in a proportional way to the negative of the derivative of the error with respect to the j -th weight itself, as reported in Equation 3.21 where γ is a constant called learning rate.

$$\Delta_p w_j = -\gamma \frac{\partial E^p}{\partial w_j} \quad (3.21)$$

The partial derivative of the LMS error of pattern set p can be rewritten as in Equation 3.22

$$\frac{\partial E^p}{\partial w_j} = \frac{\partial E^p}{\partial s^p} \frac{\partial s^p}{\partial w_j} \quad (3.22)$$

Then from Equation 3.16 the partial derivative of s^p can be expressed as in Equation 3.23

$$\frac{\partial s^p}{\partial w_j} = y_j^p \quad (3.23)$$

Now if we define a new quantity as in Equation 3.24

$$\delta^p = -\frac{\partial E^p}{\partial s^p} \quad (3.24)$$

The gradient descent of Equation 3.21 can be rearranged as

$$\Delta_p w_j = \gamma \delta^p y_j^p \quad (3.25)$$

At this point the generalized delta rule allows to compute the value of δ^p by back-propagating the error signals. By applying once again the chain rule one can rewrite Equation 3.24 as Equation 3.26

$$\delta^p = -\frac{\partial E^p}{\partial s^p} = \frac{\partial E^p}{\partial y^p} \frac{\partial y^p}{\partial s^p} \quad (3.26)$$

The partial derivative of the network output with respect to the input signal can be rewritten as follows by looking at Equation 3.17

$$\frac{\partial y^p}{\partial s^p} = F'(s^p) \quad (3.27)$$

The equations from 3.21 to 3.27 are valid for each k -th unit in the network but the subscript was omitted to simplify the notation. If the unit of interest is an output unit the subscript k can be substituted with o and the partial derivative of the LMS error with respect to the network output can be calculated as

$$\frac{\partial E^p}{\partial y_o^p} = -(d_o^p - y_o^p) \quad (3.28)$$

By substituting Equation 3.28 and Equation 3.27 in Equation 3.26 we can get the expression of δ^p for an output neuron as follows

$$\delta_o^p = (d_o^p - y_o^p)F'_o(s_o^p) \quad (3.29)$$

In the case in which the unit of interest is a hidden unit the subscript k can be substituted with h and the error can be evaluated as a function of each input from the hidden layer to the output one using once again the chain rule, as shown in Equation 3.30

$$\frac{\partial E^p}{\partial y_h^p} = \sum_{o=1}^{N_o} \frac{\partial E^p}{\partial s_o^p} \frac{\partial s_o^p}{\partial y_h^p} = \sum_{o=1}^{N_o} \frac{\partial E^p}{\partial s_o^p} \frac{\partial}{\partial y_h^p} \sum_{j=1}^{N_h} w_{ko} y_j^p = \sum_{o=1}^{N_o} \frac{\partial E^p}{\partial s_o^p} w_{ho} = - \sum_{o=1}^{N_o} \delta_o^p w_{ho} \quad (3.30)$$

When substituting Equation 3.30 in Equation 3.26 one can get the following result

$$\delta_h^p = F'(s_h^p) \sum_{o=1}^{N_o} \delta_o^p w_{ho} \quad (3.31)$$

This process is called back-propagation and can be intuitively described as in (56). The network gets some inputs and propagate them through its layers to the output units. Then the desired and the obtained results are compared and an error e_o is calculated for a specific output neuron. The aim of the back-propagation is to bring e_o to zero and to do so the incoming weights are modified by a quantity expressed in Equation 3.32 together with the modification of the weights between input and hidden neurons obtained once again with the delta rule.

$$\Delta w_{ho} = (d_o - y_o)y_h \quad (3.32)$$

Since we do not know the δ for the hidden units, the chain rule is applied, and the error of an output neuron is distributed to all the hidden neurons connected to it, considering also the activation function whose derivative must be applied to δ before the process can go on.

The learning algorithm described above, the so called back-propagation, is not the only possible algorithm for neural networks training. Other algorithms including resilient backpropagation (RPROP) with or without weight backtracking, its modified globally convergent version (GRPROP) or the Levenberg-Marquardt training algorithm can be used. Moreover, feed-forward neural networks are only the simplest structure of multi-layered networks, in fact Recurrent neural

networks (RNNs) including Elman networks, Jordan networks and Hopfield networks are a more sophisticated possibility. Finally, it must be said that the term Deep neural networks (DNNs) refer to all the neural networks with more than one hidden layer, including also Convolutional neural networks (CNNs).

3.2.3 Classification and regression trees

Classification and regression trees (CART) are another kind of supervised algorithms which can be used both for classification and for regression. We can start to analyse the classification purposes. Let's suppose we have a set of predictors variables X_1, \dots, X_n which correspond to a sample of a class variable Y we want to predict, and which can take the classes value of $1, 2, \dots, k$. For instance, the class variable Y could be the gender 1 for female and 2 for male and the predictors could be the weight X_1 and the height X_2 . The solution is a partition of X into a number of sets k equal to the number of classes. In this way if the variable space X belongs to the partition A_q , for q that goes from 1 to k , the predicted value of Y will be q . In our example if we assume that the height and the weight fall into the partition A_1 , then the value of the predicted Y will be 1 and consequently female.

In Figure 3.5 an example of CART used for classification is shown. One should notice that the tree starts from a single node and at each node a binary split is created with the left side corresponding to the condition expressed in the node.

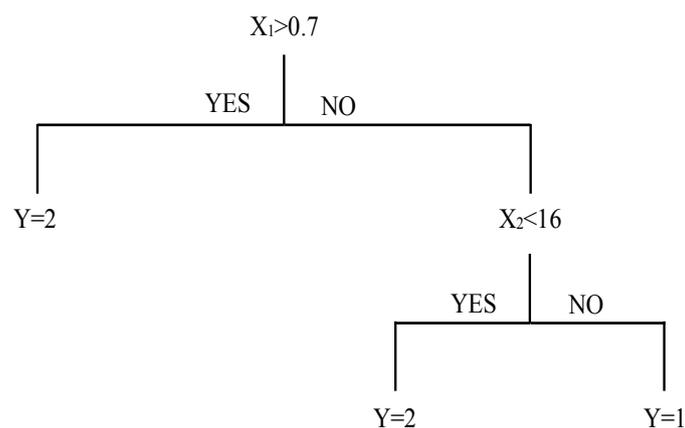


Figure 3.5: Classification tree example.

The tree ends with leaf nodes containing the predicted class. By starting from the top, it is possible to identify the Y class starting from a set of predictor variables X , for instance if X_1 is equal to 3 and X_2 is equal to 7 at the first split we get a NO, while at the second we get a YES so, in the end Y is equal to 2. Thanks to this example, one can understand the necessity of having a methodology which is able to identify the most suitable input variables to use and the position of the split points.

Following the mathematical explanation given by Morgan in (57) we could say that the aim of a CART algorithm is to find a function $d(X)$ which is able to correctly classify the domain X to the response variable Y . The error measure used to select this function is the mean squared prediction error reported in Equation 3.33.

$$E = (d(X) - E(Y|X))^2 \quad (3.33)$$

At this point the Gini node impurity criterion is used to define the splits in a way that maximizes the impurity decrease in each node as in Equation 3.34

$$1 - \sum_{c=1}^C p^2(c|l) \quad (3.34)$$

Where l is a leaf node, c is a training sample and $p(c|l)$ is the probability that an observation l belongs to class c .

If we focus now on regression trees, the differences are very few. One may consider that, in this case, “the Y variable takes ordered values and a regression model is fitted to each node to give the predicted values of Y ” (8). This time the mathematical model is given by Equation 3.35 which express the mean of the variable to be predicted in a specific node.

$$\hat{Y} = \frac{1}{c} \sum_{c=1}^C Y_c \quad (3.35)$$

Finally, a stopping criterion is needed to decide when to end the procedure described above. The most common solution is to assign a minimum number of instances resulting in each leaf node, otherwise the model could end up with a number of leaf nodes equal to the number of the Y samples. A technique called pruning can be used to further reduce the size of the tree, by removing some of the leaf nodes, with the effect of reducing the complexity of the tree.

3.2.4 ARIMA models

ARIMA is an acronym which stands for Autoregressive Integrated Moving Average and indicates a statistical algorithm obtained by coupling autoregressive and moving average models, which are two kinds of multiple regression models. Let's consider a univariate time series y_t , the first order differenced time series can be defined as the difference between an observation at time t and the previous one, as in Equation 3.36

$$y'_t = y_t - y_{t-1} \quad (3.36)$$

The noise ε_t associated with a differenced series which is purely white noise can be written as in Equation 3.37

$$\varepsilon_t = y_t - y_{t-1} \quad (3.37)$$

The model can be rewritten by adding a term c , which is the average of the changes between two consecutive observations, as follows

$$y_t = c + y_{t-1} + \varepsilon_t \quad (3.38)$$

The meaning of c is that if it is positive the changes in y_t tend to make it increase, in average, while if it is negative the time series will tend to decrease.

Using the notation of (58) the non-seasonal ARIMA(p,d,q) model can be expressed as in Equation 3.39

$$y'_t = c + \phi_1 y'_{t-1} + \dots + \phi_p y'_{t-p} + \theta_1 \varepsilon'_{t-1} + \dots + \theta_q \varepsilon'_{t-q} + \varepsilon_t \quad (3.39)$$

In which p is the order of the autoregressive part, q is the order of the moving average part and d is the degree of first differencing involved. The same equation can be expressed using the backshift notation as

$$(1 - \phi_1 B - \dots - \phi_p B^p)(1 - B)^d y_t = c + (1 + \theta_1 B + \dots + \theta_q B^q) \varepsilon_t \quad (3.40)$$

Where $(1 - B)^d y_t$ is the d -th order difference and B is the backshift operator which shifts the data back of one time-lag, for instance $B y_t = y_{t-1}$ and $B^2 y_t = y_{t-2}$. Now that we have defined the model,

its necessary to determine the parameters $c, \phi_1, \dots, \phi_p, \theta_1, \dots, \theta_q$ and this can be done using the maximum likelihood estimation MLE. Other measures of the goodness of the ARIMA model are Akaike's Information Criterion (AIC), corrected Akaike's Information Criterion (AICc) and Bayesian Information Criterion (BIC) whose definitions are reported respectively in Equations 3.41, 3.42 and 3.43.

$$AIC = -2 \log(L) + 2(p + q + k + 1) \quad (3.41)$$

$$AICc = AIC + \frac{2(p+q+k+1)(p+q+k+2)}{T-p-q-k-2} \quad (3.42)$$

$$BIC = AIC + [\log(T) - 2](p + q + k + 1) \quad (3.43)$$

In the previous three equations L is the likelihood of the data while k is a parameter which is equal to one when c is non-zero and zero otherwise. These parameters can be used to select p and q but not d , since the likelihood of the data depends on the order of differencing itself.

It is also possible to take into consideration the seasonality of data by developing a seasonal ARIMA model with the parameters $(p, d, q)(P, D, Q)_m$. An example of ARIMA $(1, 1, 1)(1, 1, 1)_4$ model is reported in Equation 3.44 where $m=4$ is the number of seasons.

$$(1 - \phi_1 B)(1 - \phi_1 B^4)(1 - B)(1 - B)^4 y_t = (1 + \theta_1 B)(1 + \theta_1 B^4) \varepsilon_t \quad (3.44)$$

Chapter 4:

Case study

4 Description of the dataset

The data that will be analysed in this study come from the electrical substation C of Politecnico di Torino, which is linked to various services whose identification codes are reported in Table 4.1, for a total floor area larger than 20,000 m². Both hot and chilled water circuits are integrated in the electrical substation, including their own pumps with a global nominal electrical power of 120 kW. The refrigeration system is composed by two chillers, which have a nominal power of 220 kW and a rated cooling capacity of 1120 kW together, and a reversible water-water heat pump with nominal power and cooling capacity of respectively 165 kW and 590 kW. These systems are connected in parallel and they both use a closed-loop geothermal water pool as heat rejection method and they are operated to guarantee a chilled water temperature of 7°C for the supply and of 12 °C for the return. The hot water, instead, is produced by a district heating exchanger which is placed in another zone.

Table 4.1: Electrical substation C services and respective identification codes.

Service	Identification code
Total electrical power	X1226
DIMAT	X294
Refrigeration unit n°2	X1022
Data centre	X1045
Bar Ambrogio	X1046
Canteen	X1047
Rectory	X1085
Print shop	X1086
Not allocated	Unknown

In Figure 4.1 the pie chart of the services of electrical substation C is reported for the biennium 2016-2017. The energy consumption in kWh has been calculated by multiplying the power load at time t for the time interval between t and $t+1$. As can be noticed, the greatest consumption is due

to the not allocated part with the 49%. Then canteen, data centre and refrigeration unit n°2 follow with 15%, 13% and 12% respectively. Finally, the consumption due to rectory, DIMAT and print shop are very low, each one under 4% of the total.

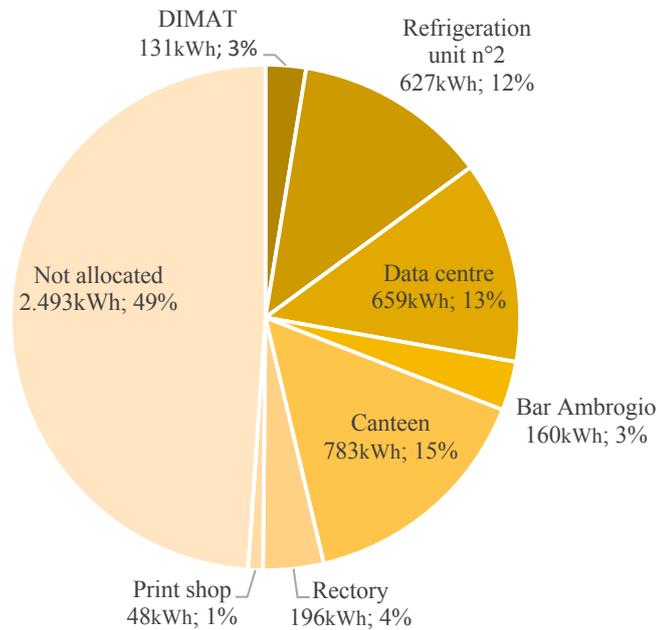


Figure 4.1: Electrical substation C services pie chart for biennium 2016-2017.

All the power measurements are collected by the Living Lab of Politecnico di Torino and the time interval between an observation and the following one is of 15 minutes. The not allocated or unknown part of the total power is the quantity that is not used for any of the specified services, thus it represents a load without a specific final use.

At first, the data will be examined as they are collected by the system, then some considerations about outliers and anomalous data will be performed in order to refine the study.

4.1 Row data analysis

The data examination takes into consideration the biennium 2016-2017 and, by involving 15 minutes measurements for two years, the dataset is composed by around $7.0 \cdot 10^4$ observations. Having to analyse such a large amount of data, it is useful to resort to carpet plots, box plots and histograms to visualize the maximum quantity of information in the simplest and clearest way.

Carpet plots are 2D plots which represent three dimensions by means of two axes and a colour gradient scale. In this case on the x axis the hour of the day is reported while on the y axis there is

the date and the colour gradient refers to the power load in kW. In Figure 4.2 the behaviour of the total power for the years 2016 and 2017 separately is shown. It is worthwhile to mention that since from these graphs some outliers can be detected, in fact, in March 2016 there is a short peak higher than 1 MW while a period of missing values is clearly visible in September 2017. The same peak of March 2016 can be recognised only in the unknown load as shown in Figure 4.3, so it is obvious that the problem in the data acquisition has occurred in the not allocated part of the monitoring system. From Figure 4.4 to Figure 4.10 the carpet plots for the remaining services are shown.

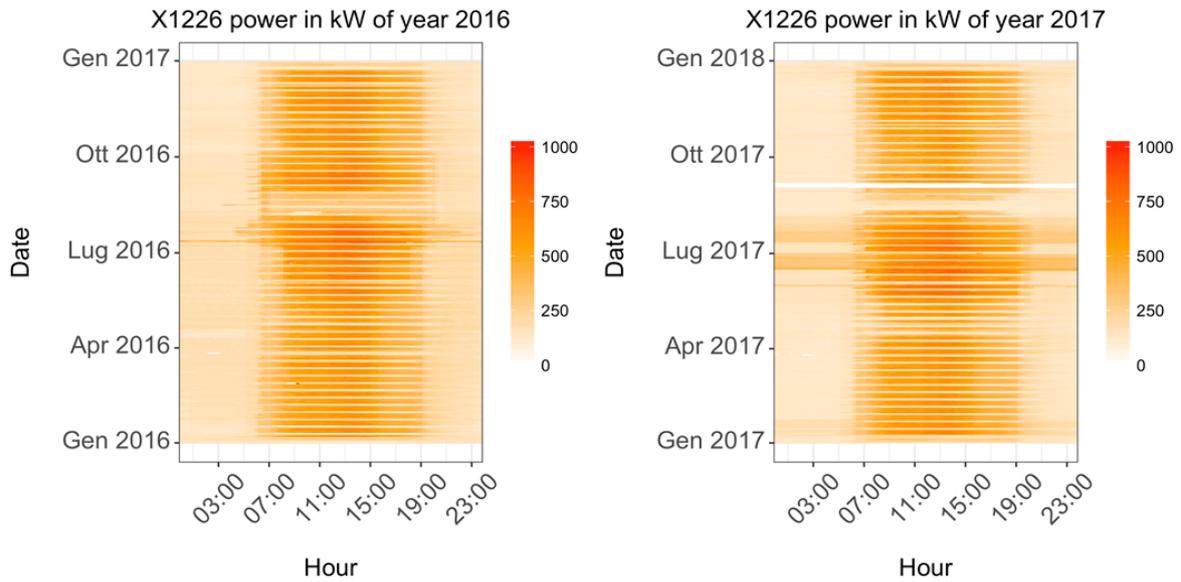


Figure 4.2: Total power carpet plot.

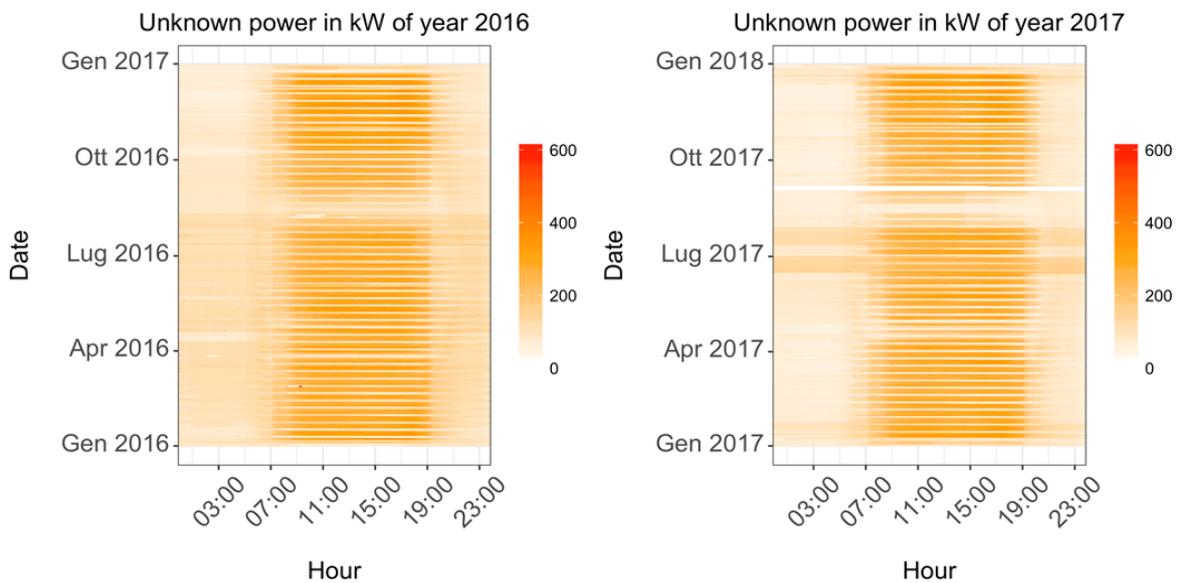


Figure 4.3: Not allocated power carpet plot.

By looking at Figure 4.4 we can state that the electrical consumption linked to the DIMAT appears to be uniform across the whole biennium, except for the month of august, in which the Politecnico is not open to the public. The same applies to Figure 4.7, Figure 4.8 and Figure 4.9 respectively for bar Ambrogio, canteen and rectory.

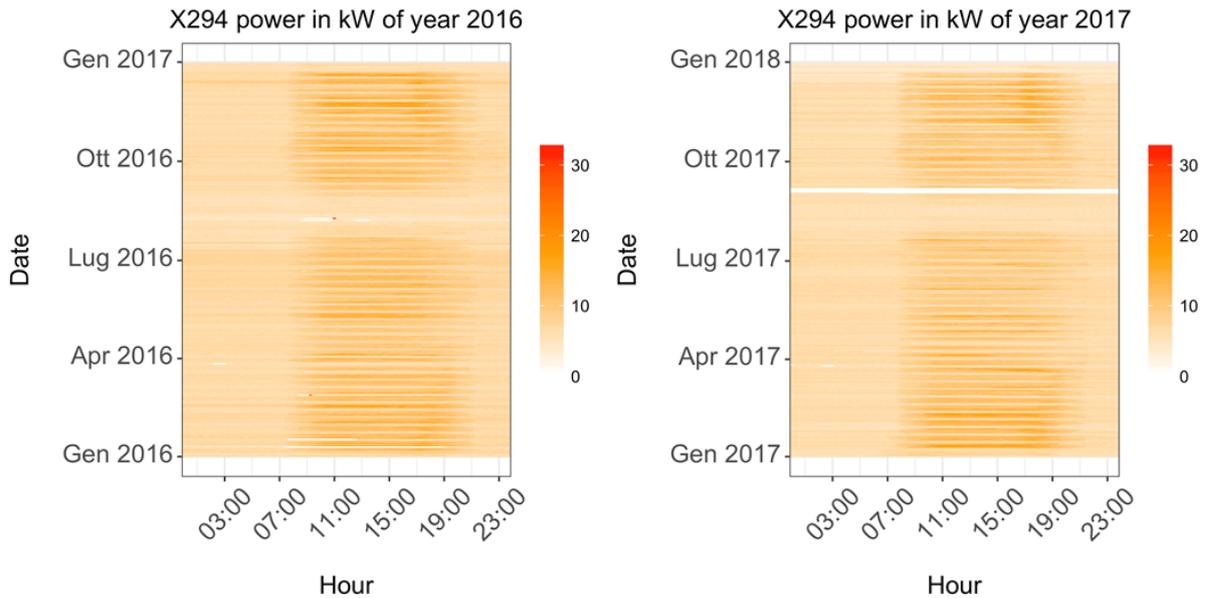


Figure 4.4: DIMAT power carpet plot.

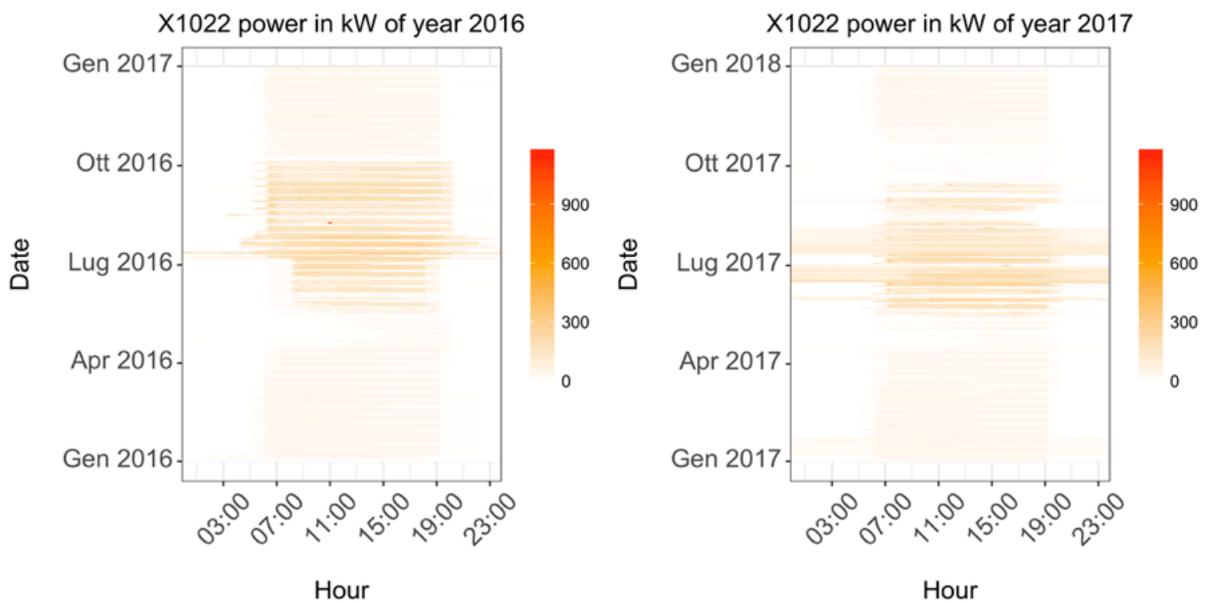


Figure 4.5: Refrigeration unit n°2 power carpet plot.

The electrical load linked to the refrigerant station n°2 reported in Figure 4.5 is higher in the period that goes from May to September, consistently with the increase of the cooling need which occurs in the same period. Finally, we could observe that while for the data centre the power need is quite constant at each time of the day, as shown in Figure 4.6, for all the others services the load is higher between 7:00 a.m. and 7:00 p.m., in accordance with the business hours of Politecnico.

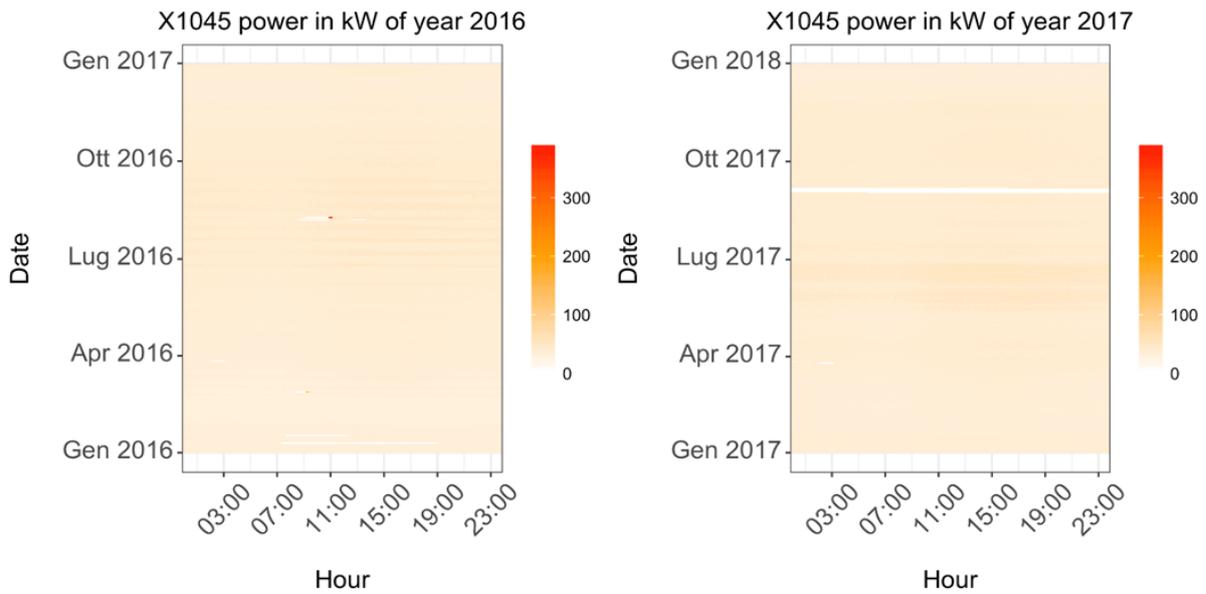


Figure 4.6: Data centre power carpet plot.

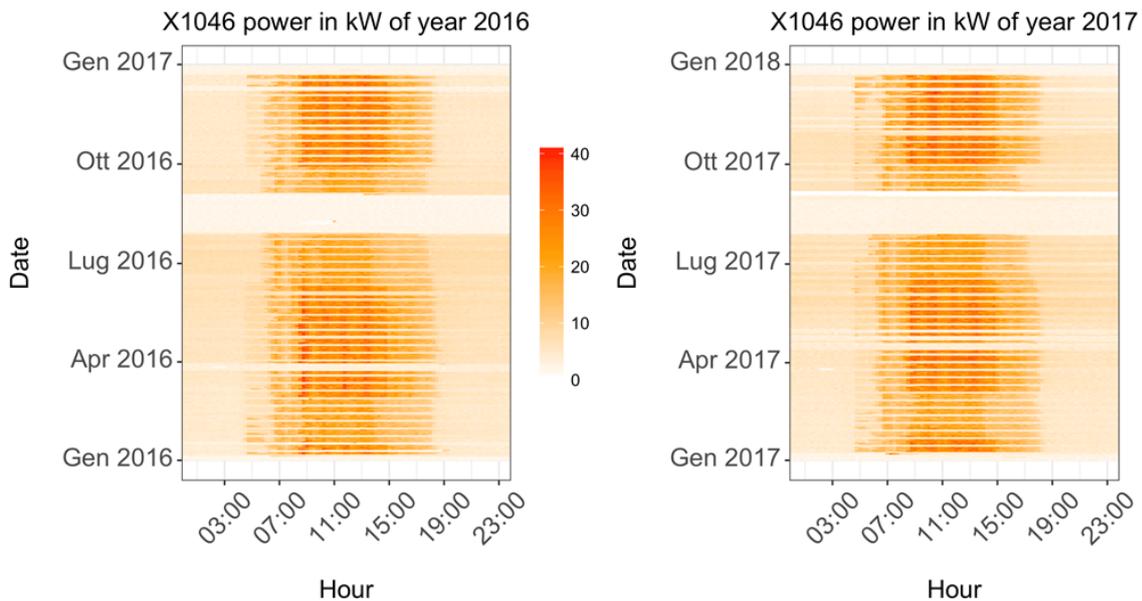


Figure 4.7: Bar Ambrogio power carpet plot.

In Figure 4.10 one can observe that while in 2016 the consumption in both August and September was very low, it increased, during the month of September in the year 2017.

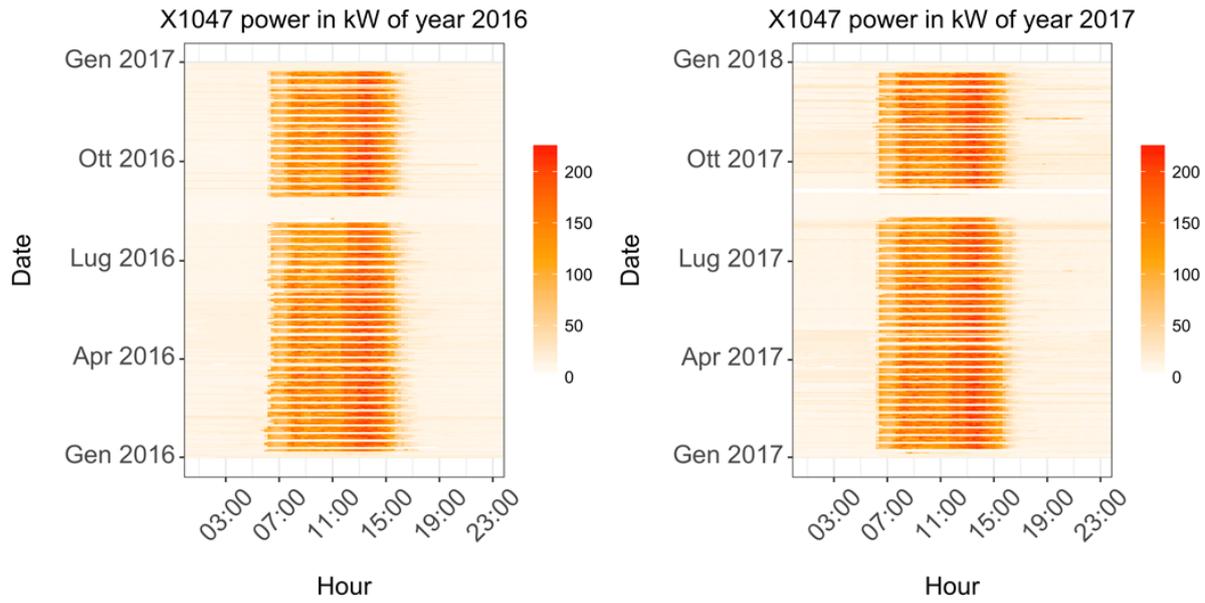


Figure 4.8: Canteen power carpet plot.

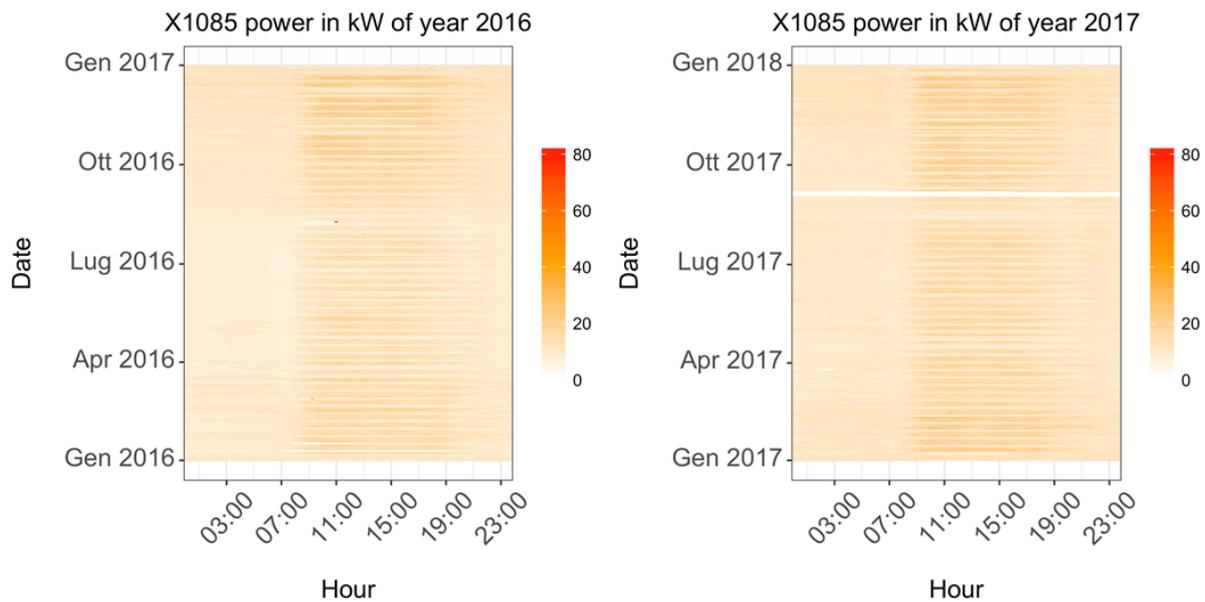


Figure 4.9: Rectory power carpet plot.

Since from this very initial phase, it is obvious that there are some ambiguous data which are far away from the median power consumption of the same service. To be more precise one can look at Figure 4.6, in which there is a short peak, in August 2016, that almost reaches 400 kW, while, during the rest of the year, the power consumption is around 40 kW. By looking at the carpet plots for DIMAT, bar Ambrogio, canteen, rectory and print shop, the same anomaly can be recognized, so it is probable that it was caused by an error in the measurement system and can be definitely identified as an outlier. In order to be more precise, we use the box plots to further investigate the presence of outliers.

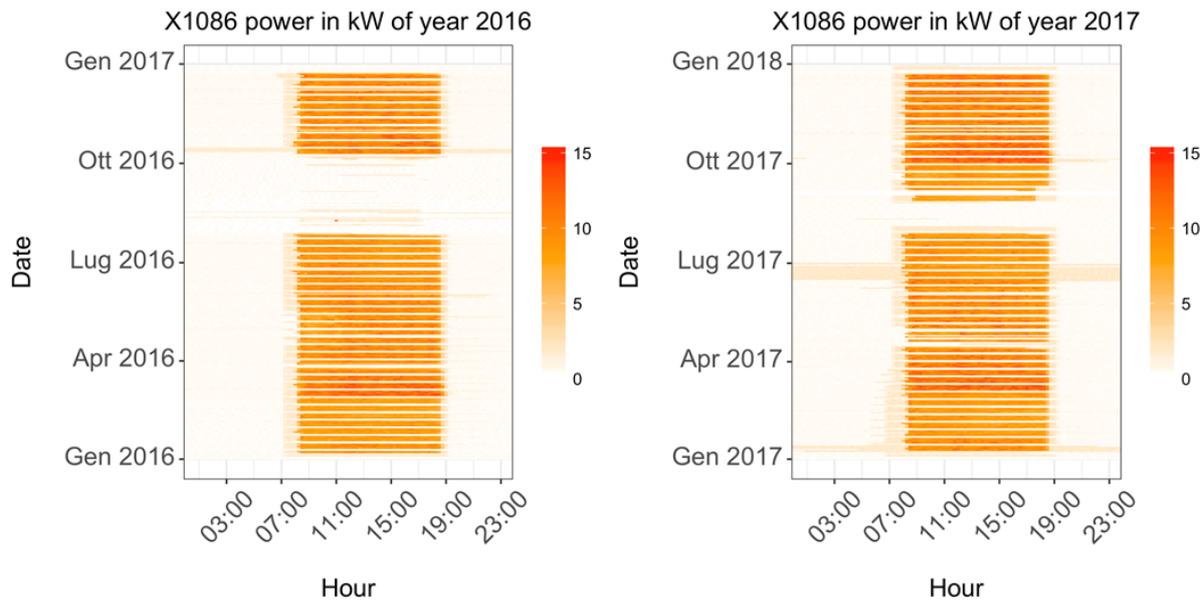


Figure 4.10: Print shop power carpet plot.

Box plot is a kind of plot which represent a “standard technique for presenting the 5-number summary which consists of the minimum and maximum range values, the upper and lower quartiles, and the median” (59). In Figure 4.11 the basic structure of a box plot is represented. The minimum and the maximum range values are the lowest and the highest values that the variable can take while the median, or second quartile, is the middle value of the set which can separate it in two parts, the higher half and the lower one. The lower, or first, quartile is the number which divide in two halves the dataset between the minimum and the median, i.e. it splits the lowest 25% of the distribution from the remaining part. Similarly, the upper, or third, quartile is the value which stays in the middle between the median and the maximum value, i.e. it separates the lowest 75% of the distribution from the rest.

Even if this type of box plot can be very useful in order to visualize the key points of the distribution of a quantity, in this case of an electrical consumption, this is not the kind of plot that will be used in this work. In fact, this time the focus is not only on the distribution, which can be also visualized by means of an histogram, rather on the points which are too far from the others and that should be labelled as outliers. It is the case of the box plot of the total power load represented in Figure 4.12.

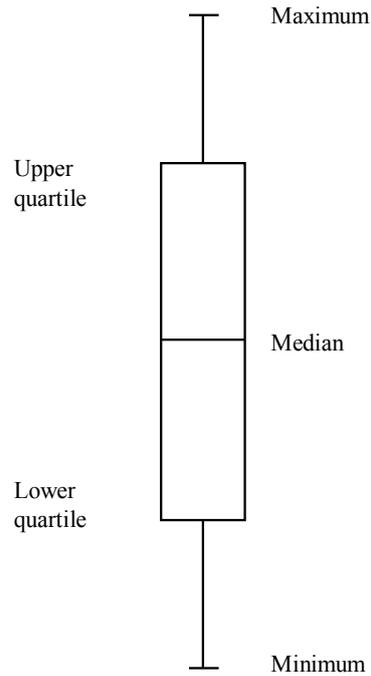


Figure 4.11: Basic structure of the box plot. Figure taken from (58) with changes.

This kind of box plot is slightly different from the one described in Figure 4.11, in fact, the upper whisker does not represent the maximum but the highest value which is within $1.5 \cdot \text{IQR}$ of the third quartile, where IQR is the inter-quartile range, equals to the distance between upper and lower quartiles. Similarly, the lower whisker does not represent the minimum but the lowest value within $1.5 \cdot \text{IQR}$ of the first quartile. This kind of architecture, also known as Tukey box plot is able to identify as outliers the data beyond the aforementioned whiskers, by plotting them as single points. As can be clearly seen from Figure 4.12 there is a number of outliers that have been identified during Saturdays and Sundays. For what concerns the weekdays, one could observe that the median is quite constant as well as the upper and lower quartiles, demonstrating that globally there is not a tendency of consuming more in specific days of the week, as we could have expected. Furthermore, one should notice that the electrical load is lower during weekends, but it is not null, this can be also stated by looking at the carpet plot in Figure 4.2. From Figure 4.13 to Figure 4.20 the boxplots of the power load of biennium 2016-2017 are reported for each service.

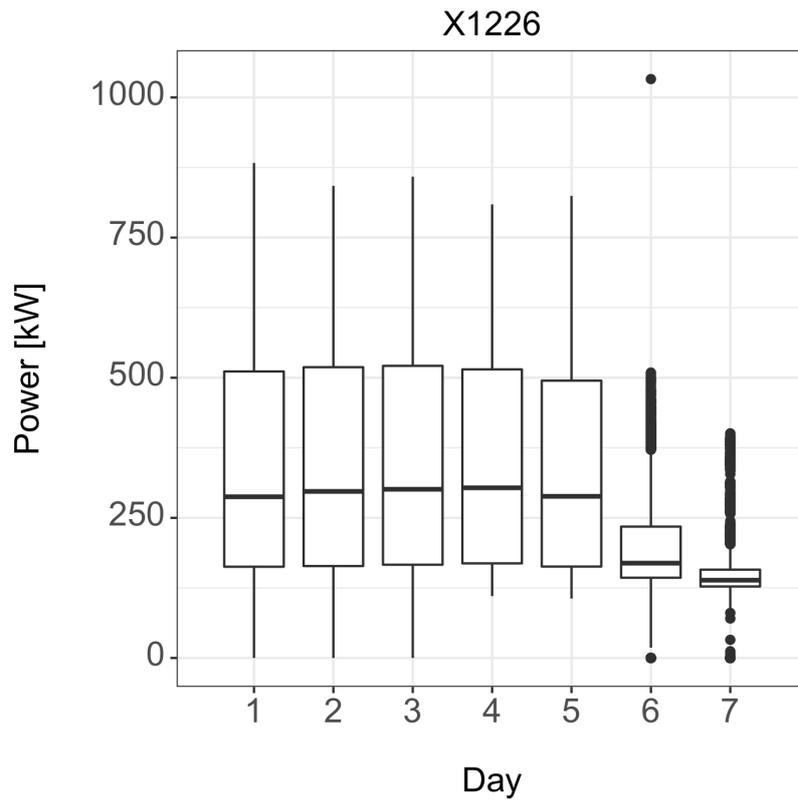


Figure 4.12: Box plot of the total power load for the biennium 2016-2017.

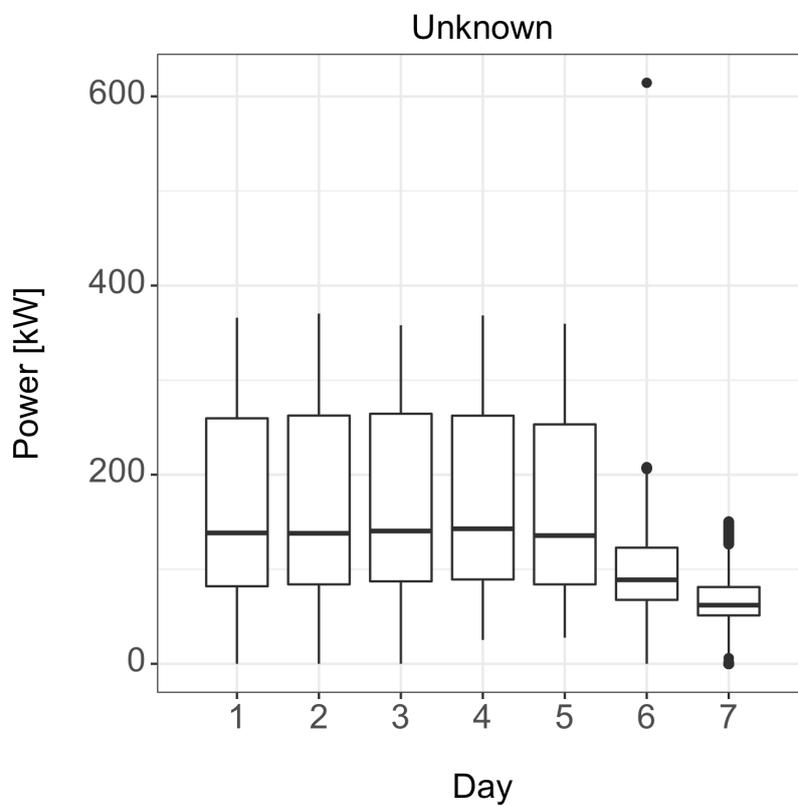


Figure 4.13: Box plot of the not allocated power load for the biennium 2016-2017.

While the not allocated power consumption and the load of the canteen and the print shop show outliers only during weekends, as depicted in Figure 4.13, Figure 4.18 and Figure 4.19 respectively, all the remaining services present outliers both during weekends and weekdays.

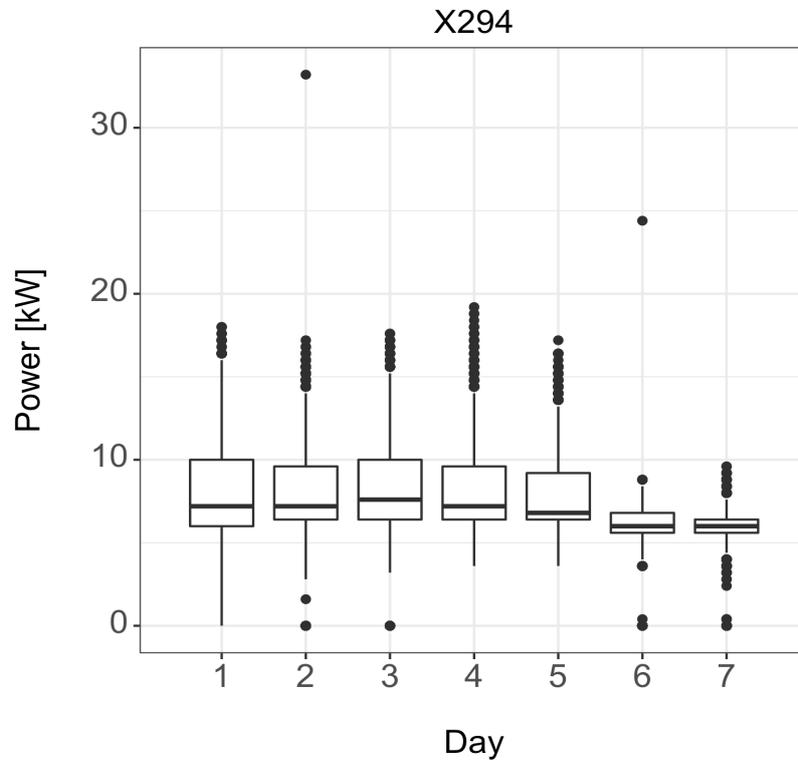


Figure 4.14: Box plot of the DIMAT power load for the biennium 2016-2017.

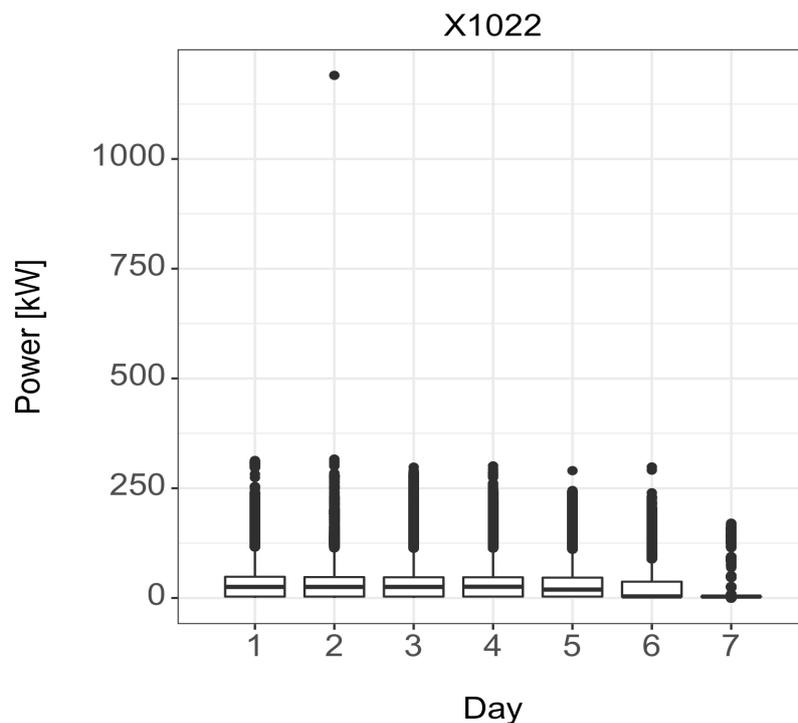


Figure 4.15: Box plot of the refrigeration unit n°2 power load for the biennium 2016-2017.

The box plot of the power consumption of Bar Ambrogio reported in Figure 4.17 clearly demonstrate that the energy need for this service is strongly driven by the day type, and so does the plot of the canteen consumption of Figure 4.18.

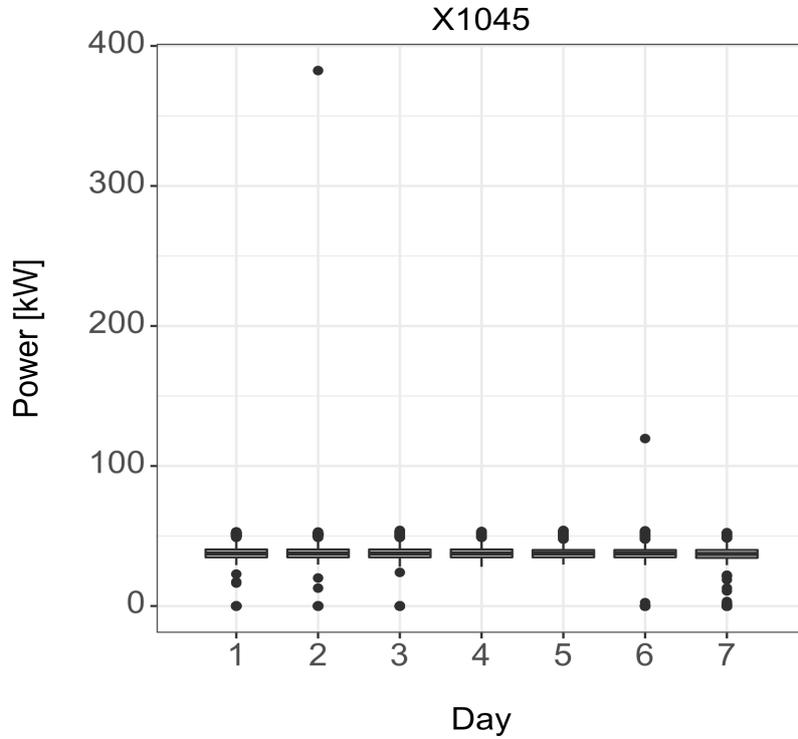


Figure 4.16: Box plot of the data centre power load for the biennium 2016-2017.

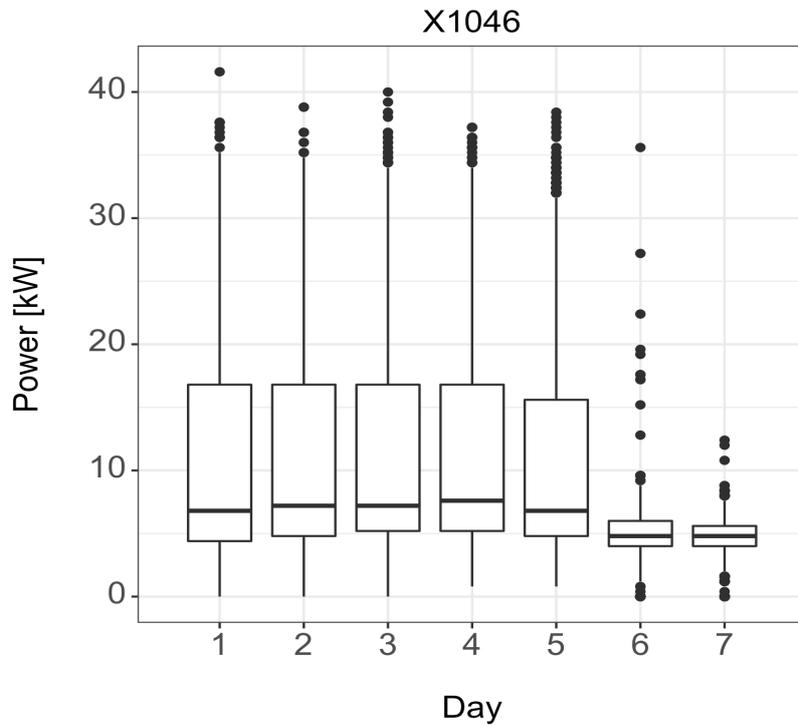


Figure 4.17: Box plot of the bar Ambrogio power load for the biennium 2016-2017.

Furthermore, by looking at Figure 4.20, one could say that the power consumption of the print shop has a very low median with respect to its upper quartile, meaning that the frequency of occurrence of values higher than the median is low compared to the ones below.

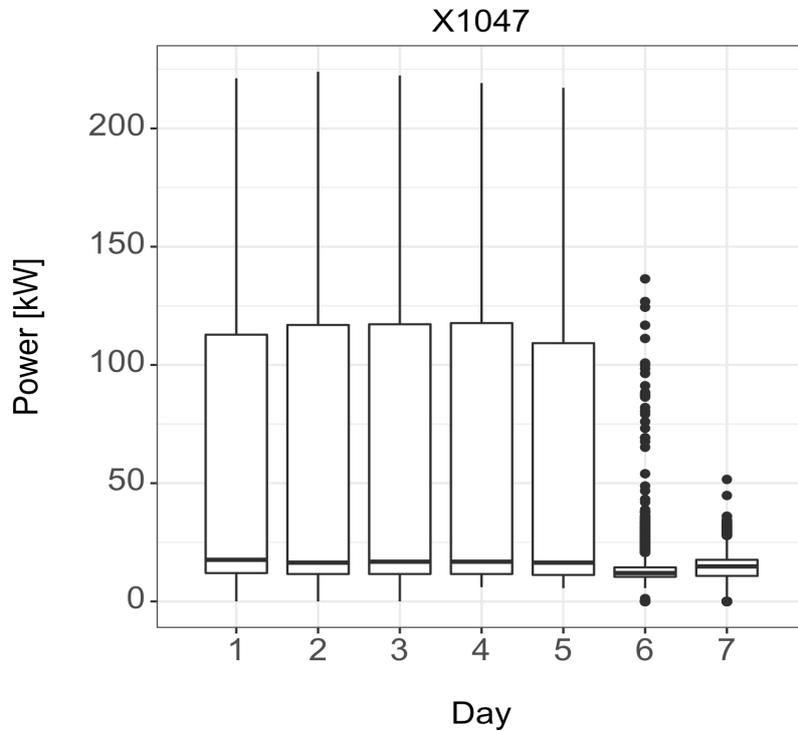


Figure 4.18: Box plot of the canteen power load for the biennium 2016-2017.

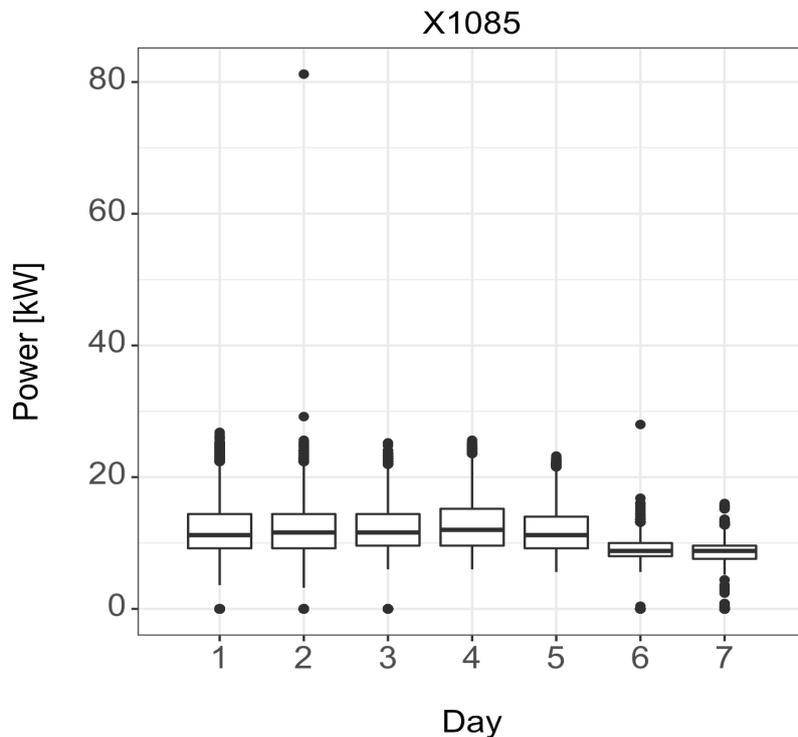


Figure 4.19: Box plot of the rectory power load for the biennium 2016-2017.

The reason for that is clear if we look at the carpet plot in Figure 4.10, where for the majority of the time the power load is nearly equal to zero. A similar consideration can be made on the power consumption of bar Ambrogio and of the canteen, whose box plots are reported in Figure 4.17 and Figure 4.18 respectively.

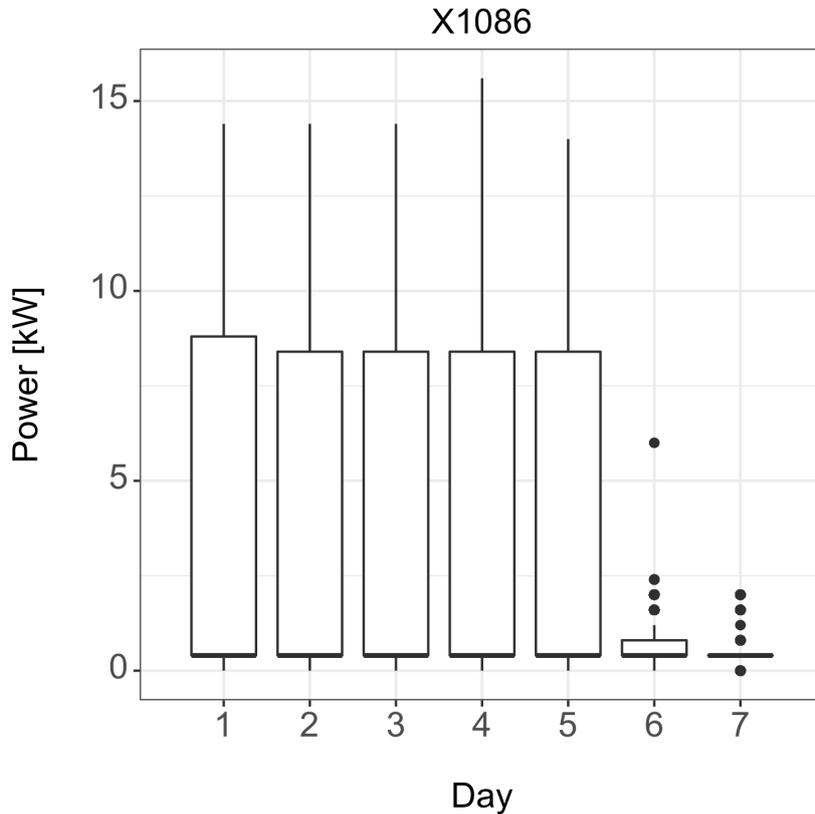


Figure 4.20: Box plot of the print shop power load for the biennium 2016-2017.

To better look at the power distribution of each service it is useful to print the histograms related to their load. From Figure 4.21 it can be observed that the distribution has a peak in the interval 100-175 kW meaning that the most probable power load is in that range. The distribution is quite constant between 175 and 625 kW and then decreases. The outlier identified by the box plot in Figure 4.12 is clearly visible also in the histogram, since the x axis goes above 1 MW with a very low number of instances. Figure 4.22 represents the total power load for the biennium 2016-2017 for each month. The general behaviour is similar to the one observed in Figure 4.21, with an increase of power consumption in the months of June and July probably related to the higher cooling load.

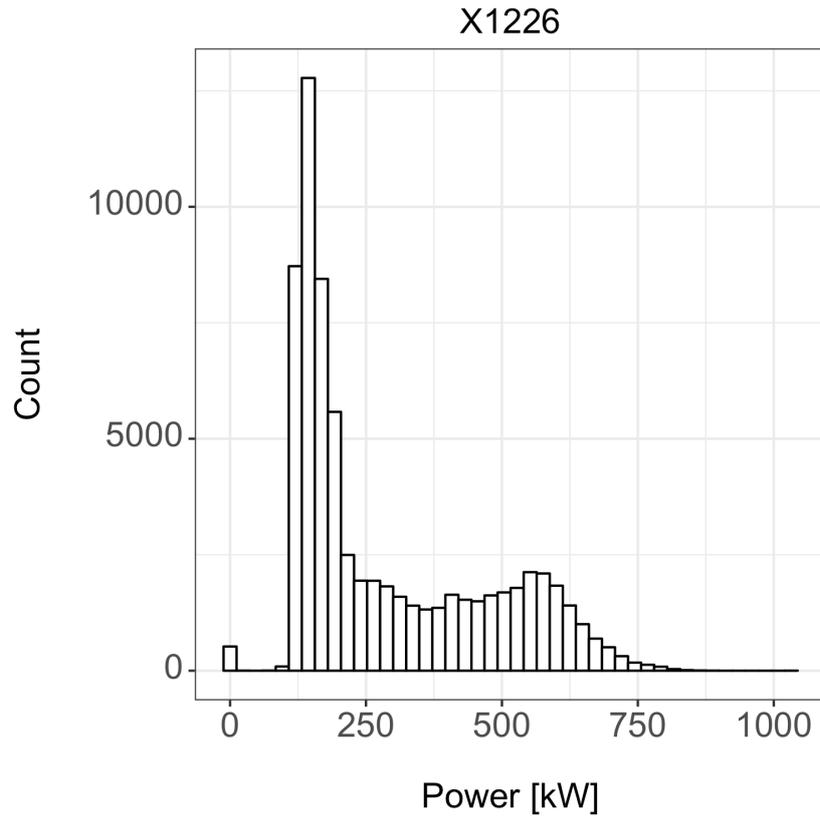


Figure 4.21: Histogram of the total power load for the biennium 2016-2017.

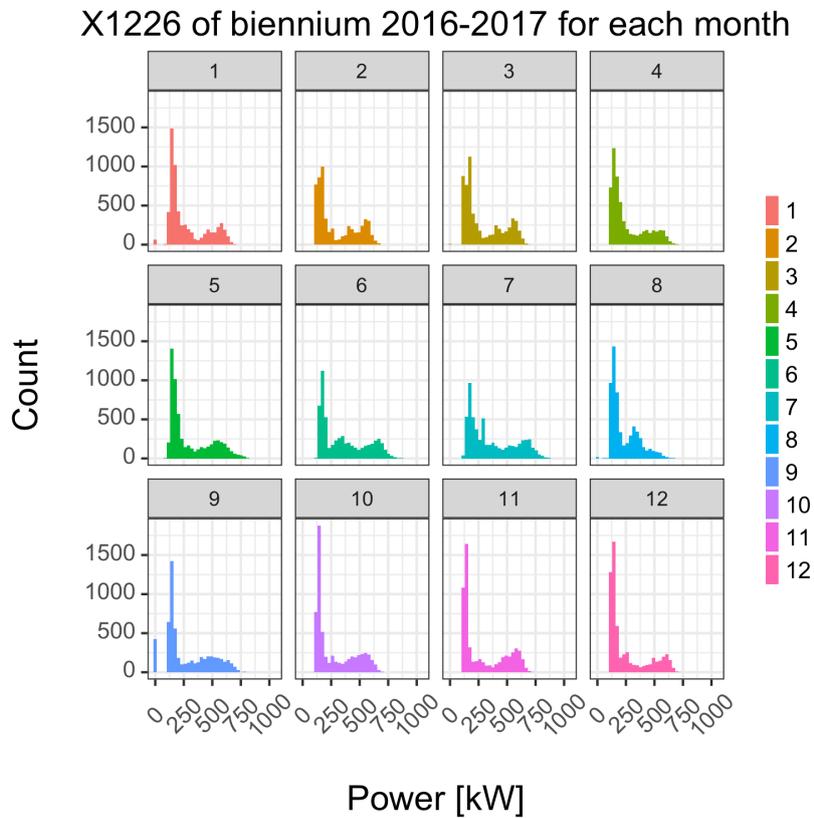


Figure 4.22: Histograms of the total power load for each month of the biennium 2016-2017.

The histograms of the total electrical load of the biennium 2016-2017 are reported once again in Figure 4.23. In this image the consumptions are divided by day type where 1 stays for Monday 2 for Tuesday and so on. As expected, the power load does not significantly change from a weekday to another, as already noticed in Figure 4.12. The consumption decreases in the weekend, but it does not reach zero.

As a final consideration, it can be noticed the electrical load of Saturdays is slightly higher than the one of Sundays. This fact is related both to the different occupancy schedule of these two day types and to the operation of some service in the morning hours which is performed during Saturdays and that will be addressed by this analysis in the following chapters.

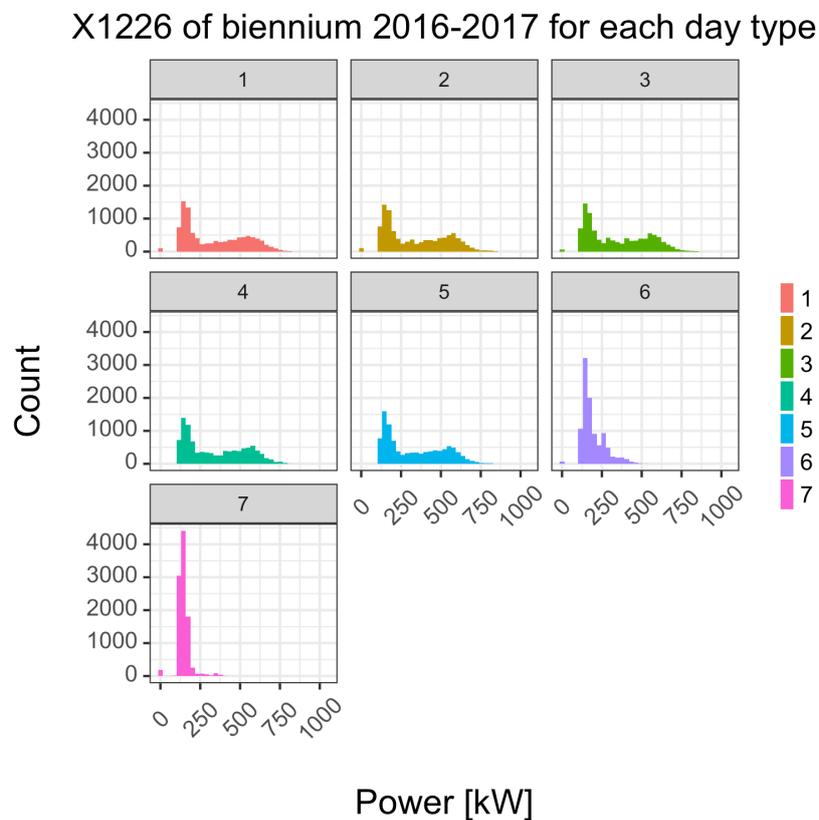


Figure 4.23: Histograms of the total power load for each day type of the biennium 2016-2017.

The presence of the outliers makes the histograms related to each service scarcely comprehensible, thus the remaining plots will be reported only after their removal in section 4.2.

4.2 Corrected data analysis

The data have been analysed as described in section 4.1 and some outliers and missing values have been discovered. The outliers have been subsequently removed by linear interpolation, while the period of missing values can be excluded from the training data set. In this section the plots of power load are represented with the corrected data. From Figure 4.24 to Figure 4.32 the power load carpet plots have been reported.

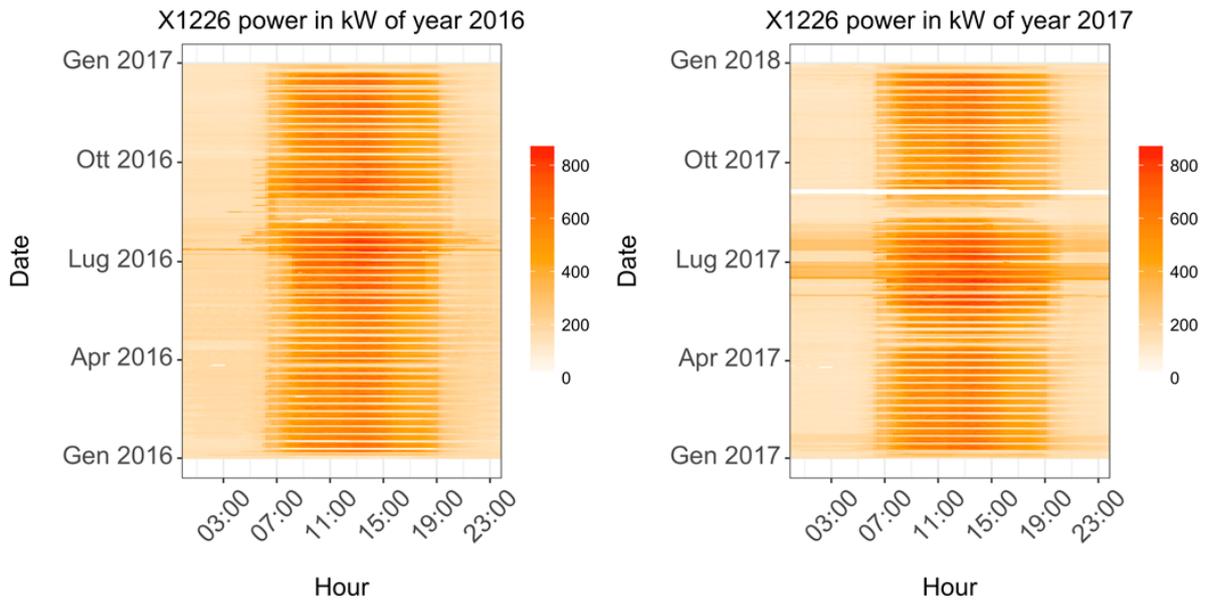


Figure 4.24: Total corrected power carpet plot.

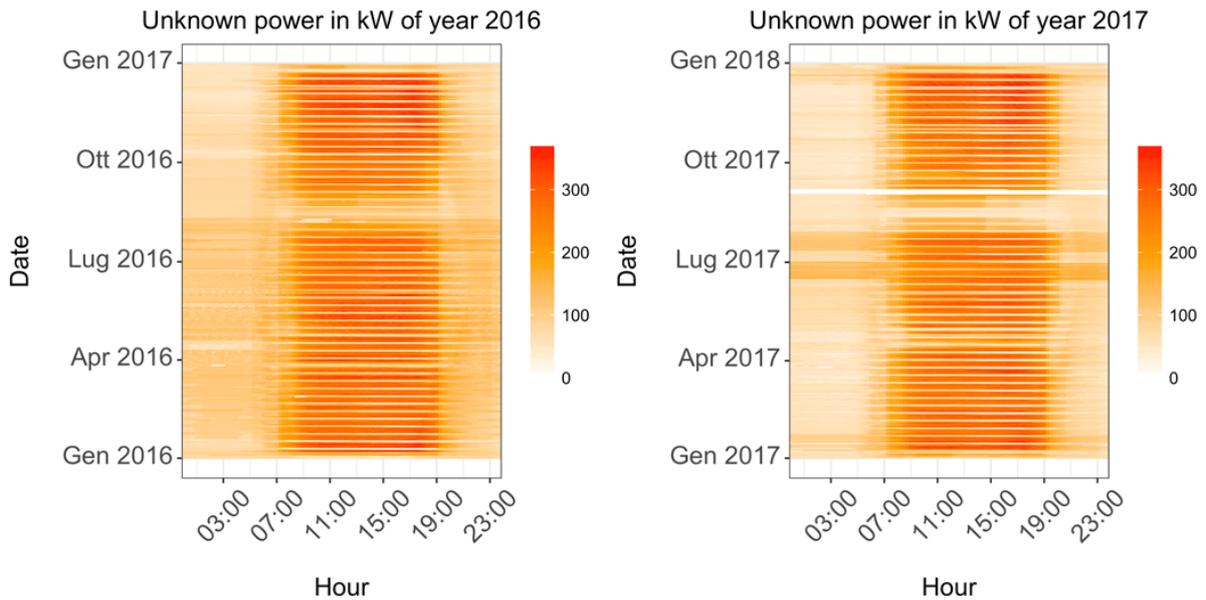


Figure 4.25: Not allocated corrected power carpet plot.

Now that the outliers have been replaced with interpolated values, we can better appreciate the behaviour of the power load represented by the colour gradient.

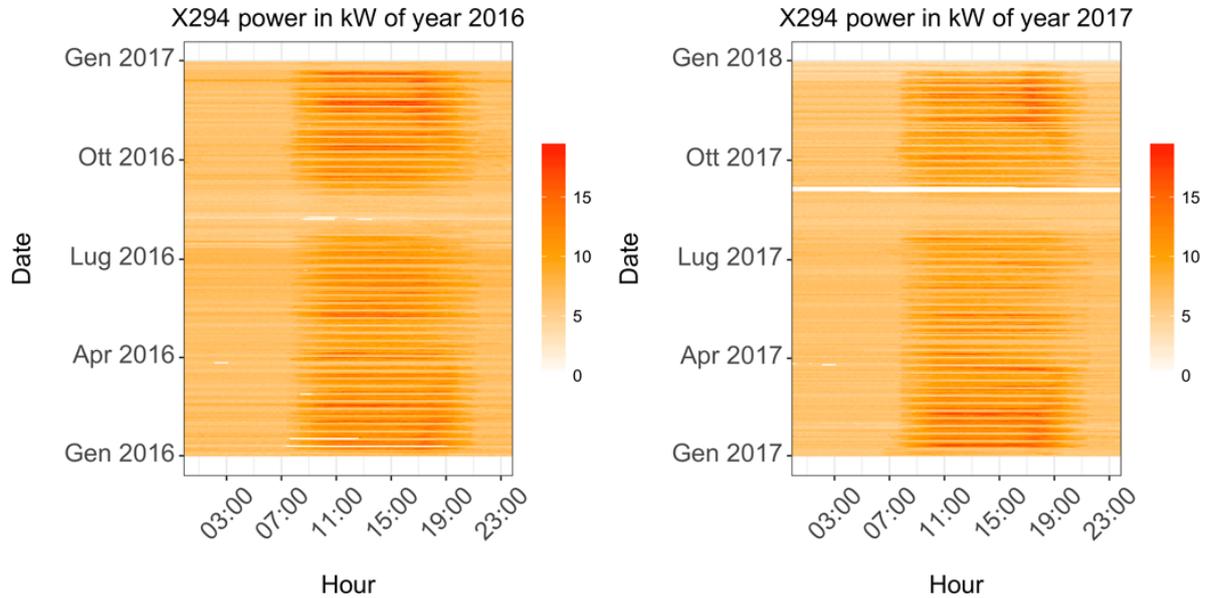


Figure 4.26: DIMAT corrected power carpet plot.

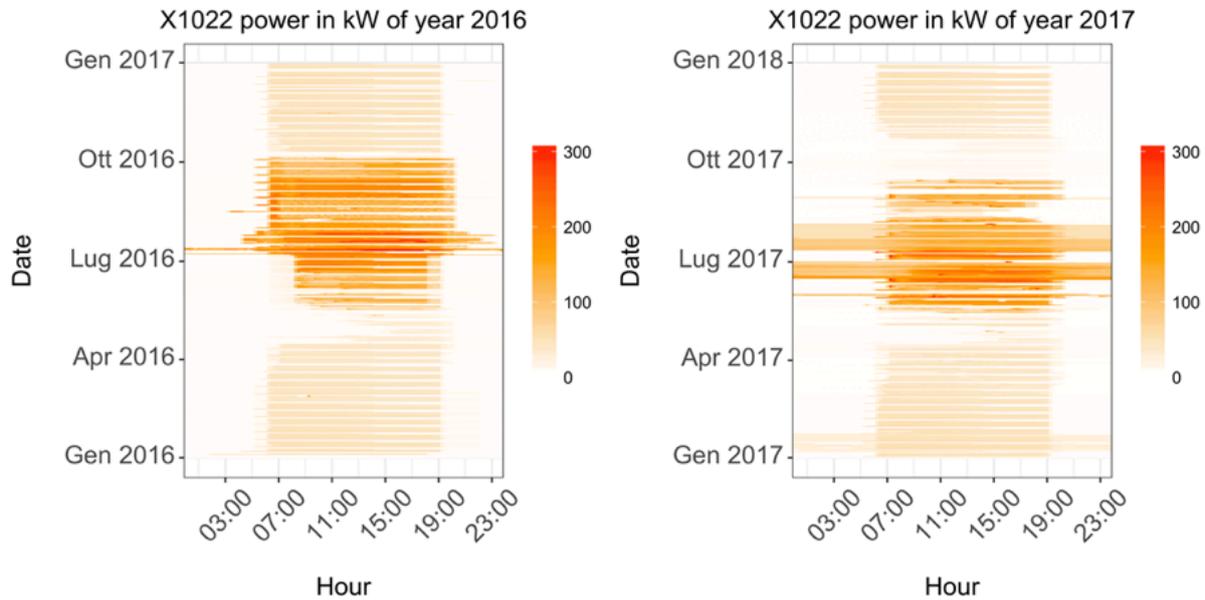


Figure 4.27: Refrigeration unit n°2 corrected power carpet plot.

The power consumption of refrigeration unit n°2 that was previously depicted in Figure 4.5 is now clearly visible in Figure 4.27 which shows a non-null consumption during the whole year, even in the colder months. Such behaviour can be explained because the power load allocated to this

service does not only include the one of refrigerators, heat pump and chilled water pumps but also the electrical consumption of the hot water circuit pumps.

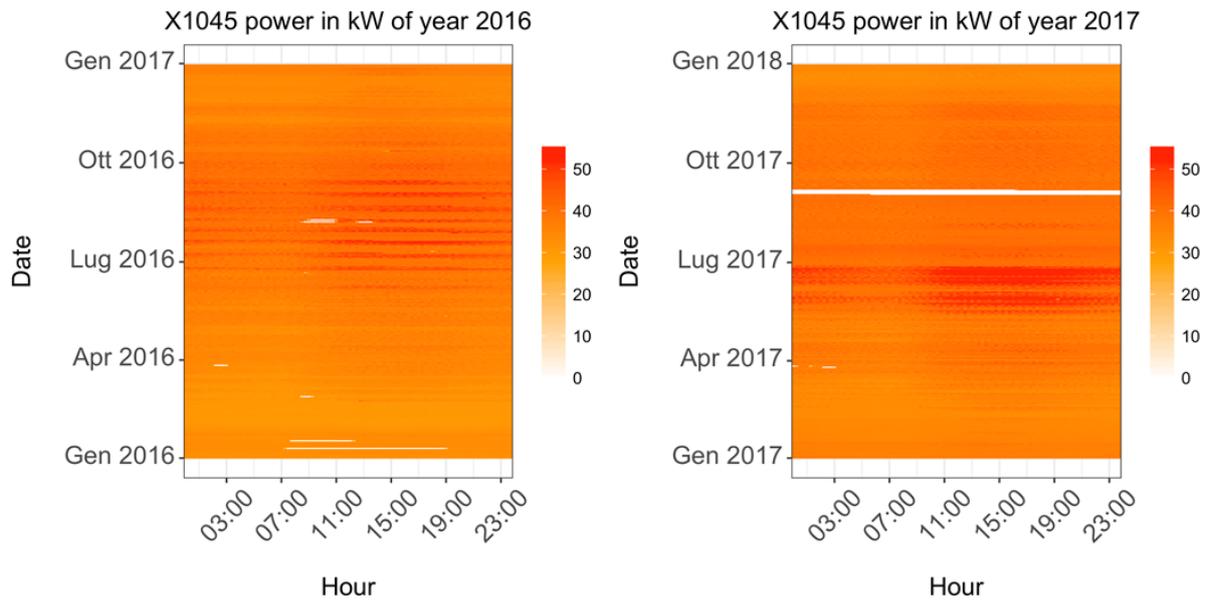


Figure 4.28: Data centre corrected power carpet plot.

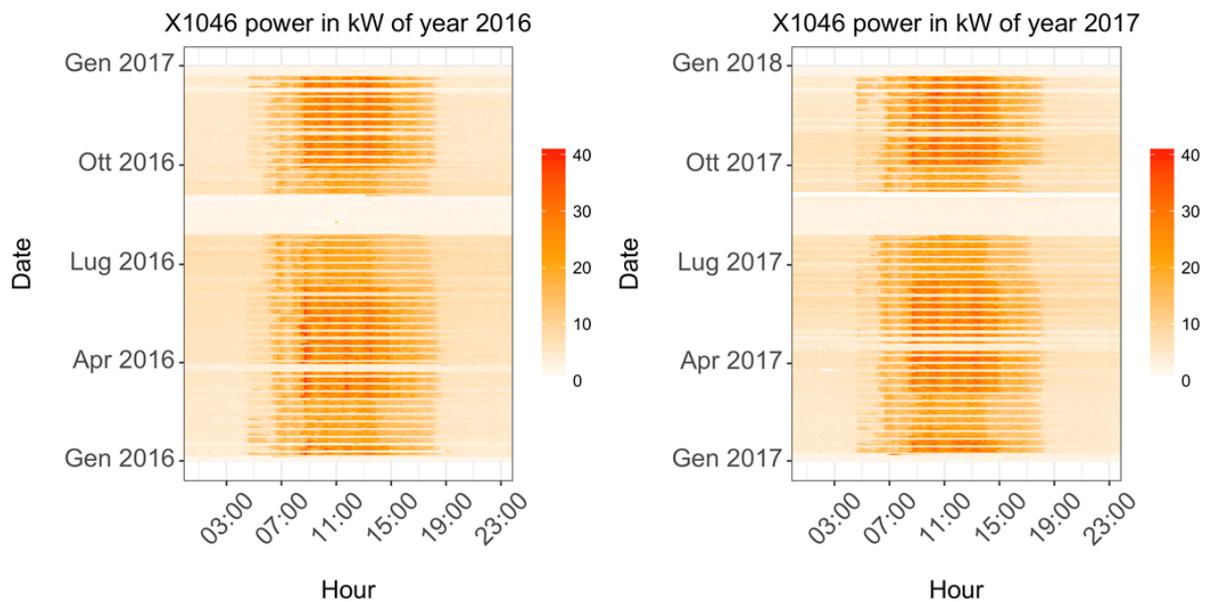


Figure 4.29: Bar Ambrogio corrected power carpet plot.

Also the carpet plots of data centre and rectory in Figure 4.28 and Figure 4.31 can be better represented without the outliers. All the considerations made in section 4.1 are still valid. It is worthwhile noticing that the consumption related to the data centre accounts for the 13% of the total, as demonstrated by Figure 4.1, even if its peak power does not exceed 55 kW. This is due to

the fact that its power load, although not so high in comparison to the total, is constant during the whole day, resulting in a large energy consumption.

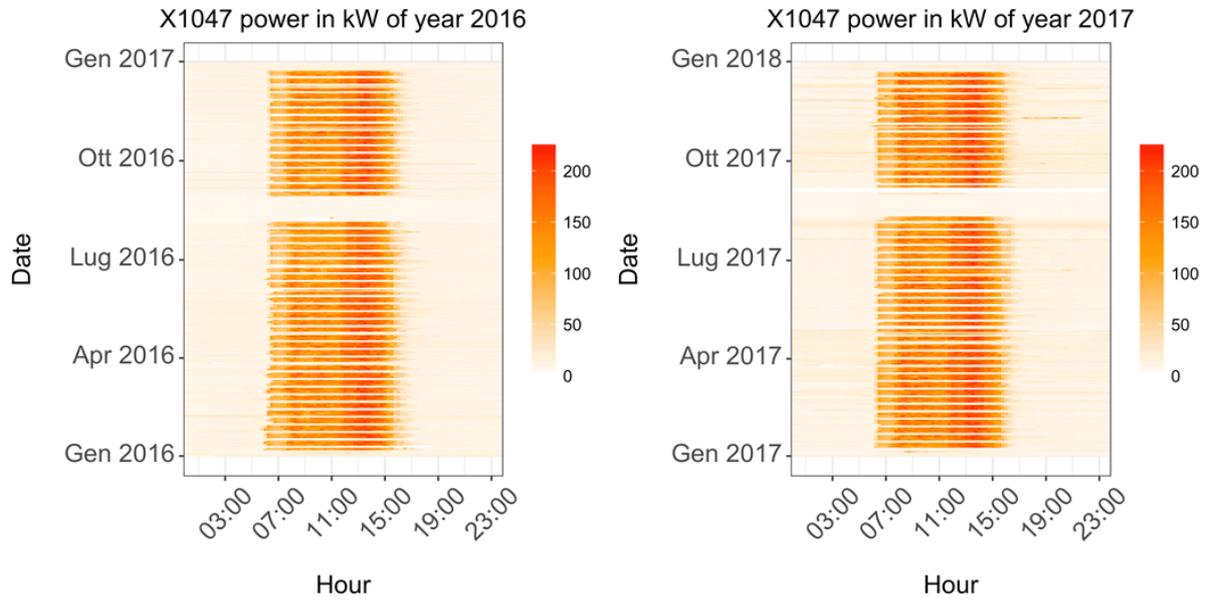


Figure 4.30: Canteen corrected power carpet plot.

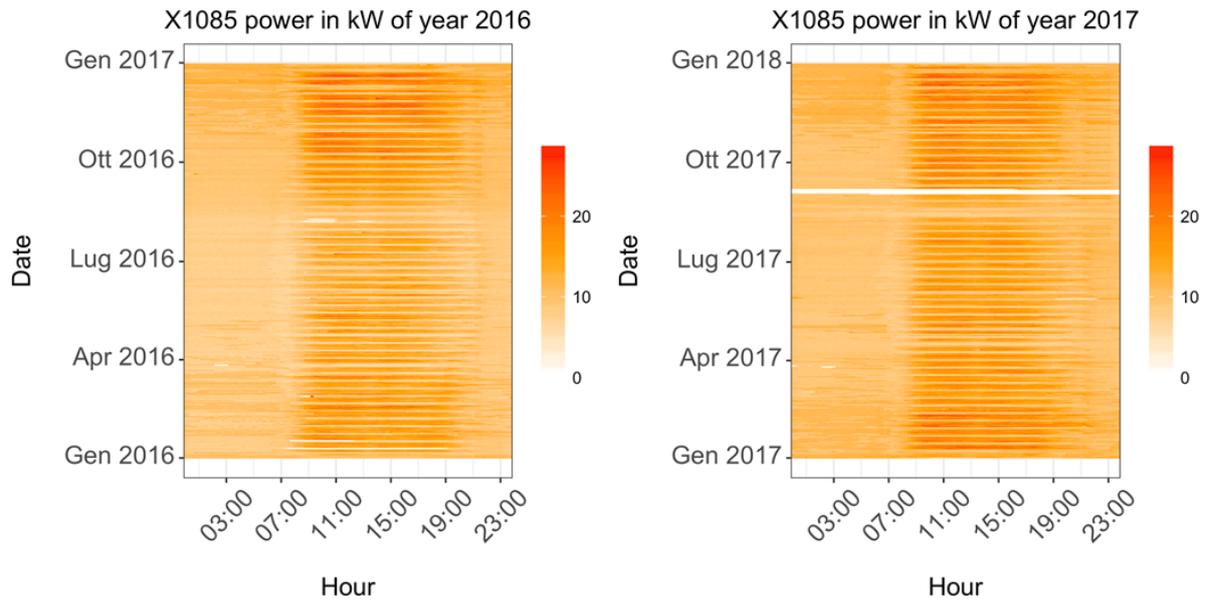


Figure 4.31: Rectory corrected power carpet plot.

In Figure 4.32 the carpet plot of the corrected power load of the print shop is reported. As it can be seen, such service does not represent a very significant part of the total consumption, but still deserves attention. Finally, one should notice that during the month of September 2016 its consumption was nearly zero while the next year it was similar to the one of the other months.

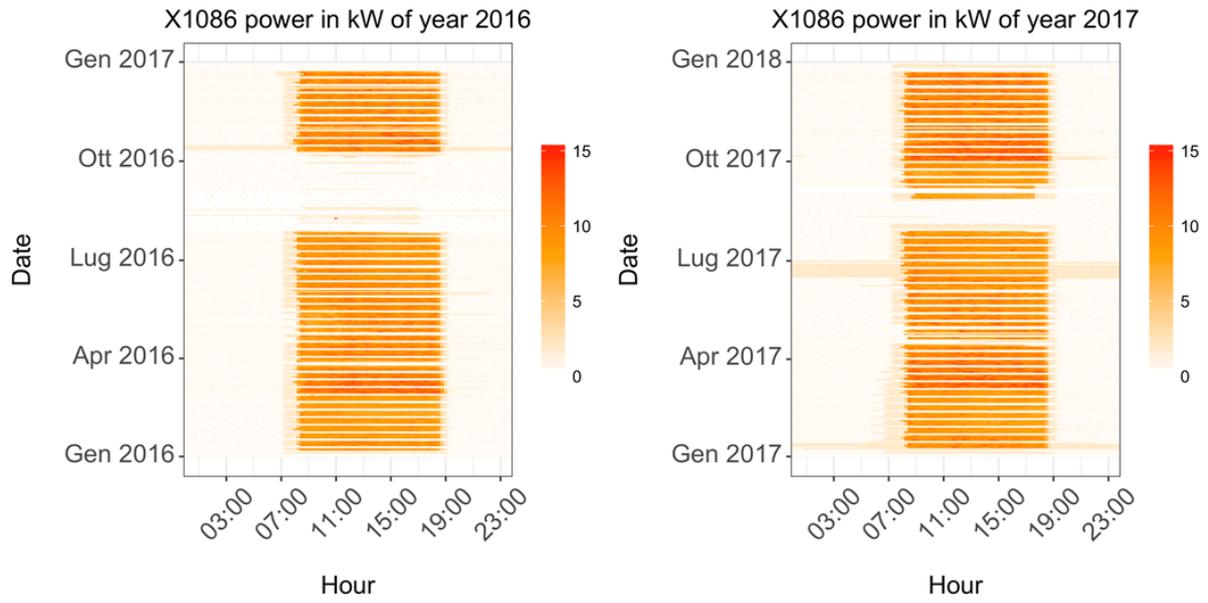


Figure 4.32: Print shop corrected power carpet plot.

Figure 4.33 to Figure 4.59 represent the histograms of the power load for each energy service, in the biennium 2016-2017 and on its months or day types.

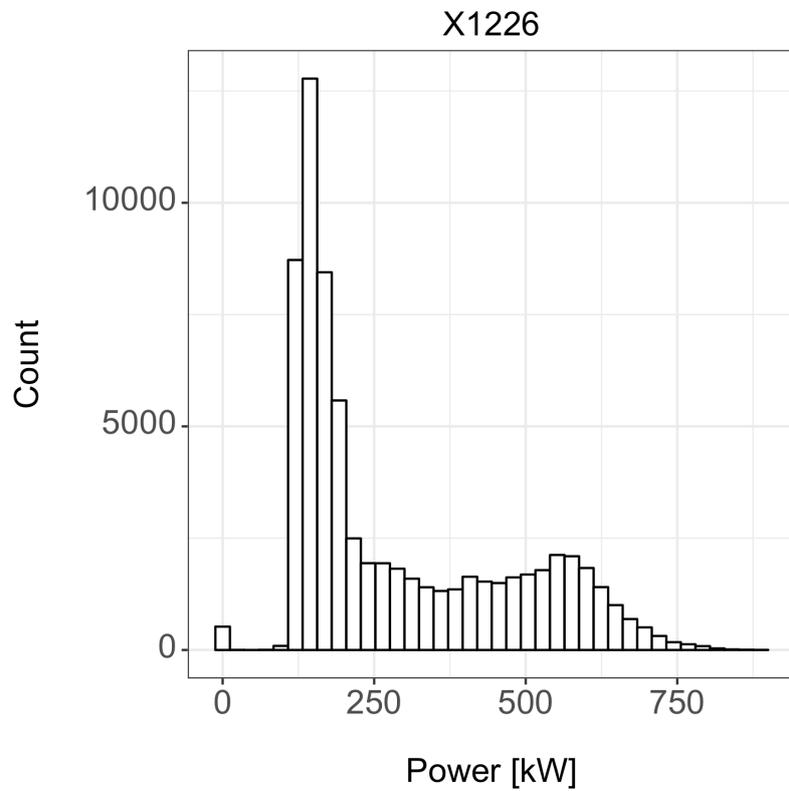


Figure 4.33: Histogram of the total corrected power load for the biennium 2016-2017.

The most probable power consumption lies around 175 kW, since for the majority of the hours most of the services are not requiring a significant amount of energy and when they do, the demand is not simultaneous.

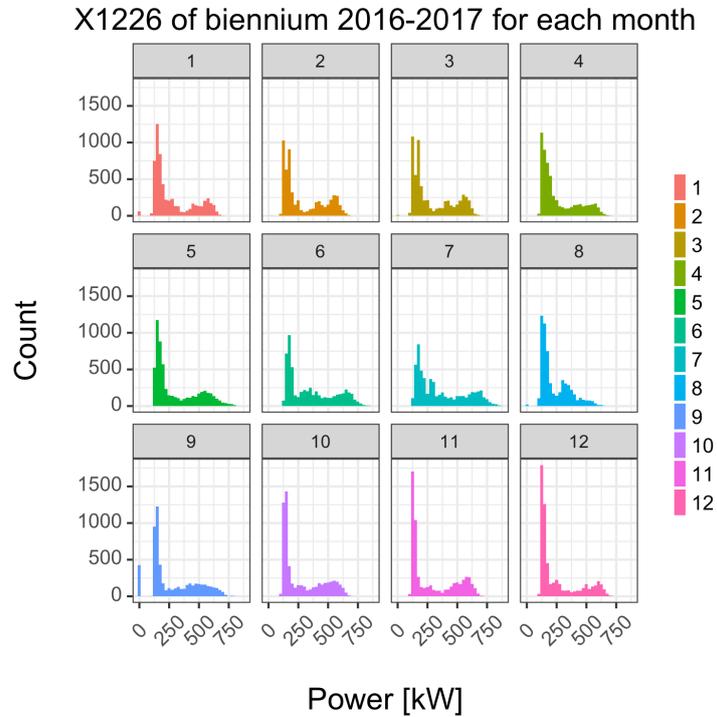


Figure 4.34: Histograms of the total corrected power load for each month of the biennium 2016-2017.

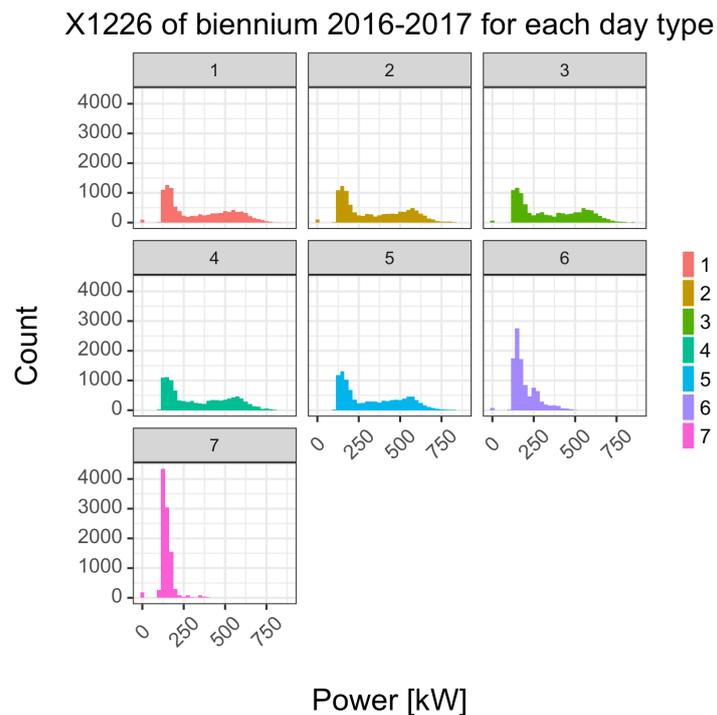


Figure 4.35: Histograms of the total corrected power load for each day type of the biennium 2016-2017.

In Figure 4.34 The consumption of the months of December appears to be more shifted towards low power values, as well as during the month of August in which there is also a small peak around 300 kW.

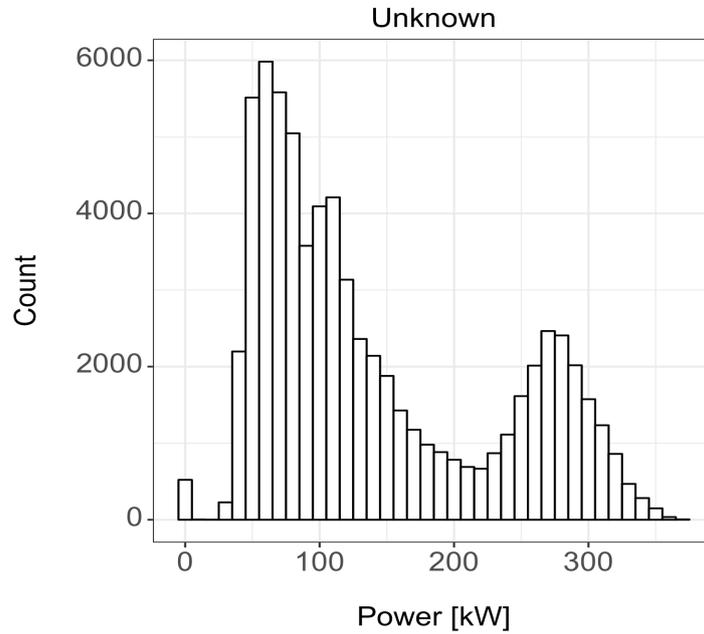


Figure 4.36: Histogram of the not allocated corrected power load for the biennium 2016-2017.

In Figure 4.37 the not allocated power load appears higher during the central months of the year from May to July, while it is minimum in August.

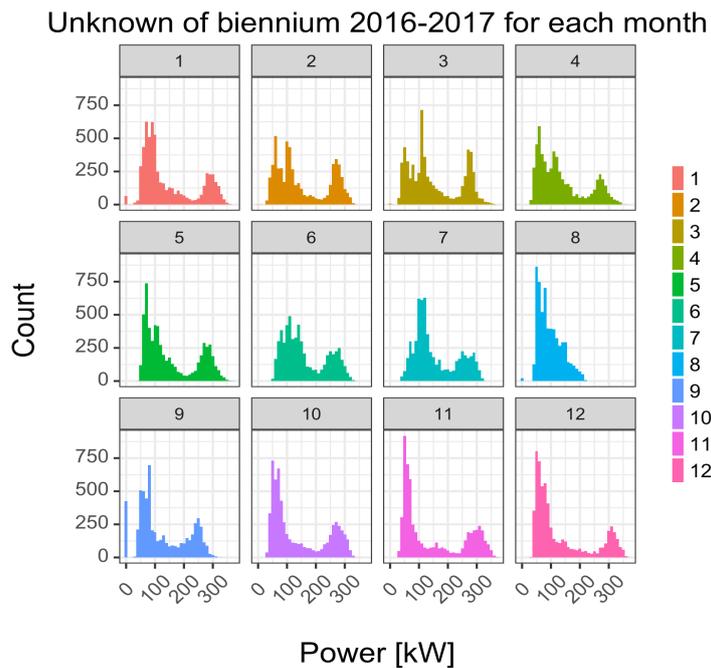


Figure 4.37: Histograms of the not allocated corrected power load for each month of the biennium 2016-2017.

As for the total power load, Figure 4.38 clearly says that the consumption of not allocated power does not vary significantly from a weekday to another, while it decreases during weekends.

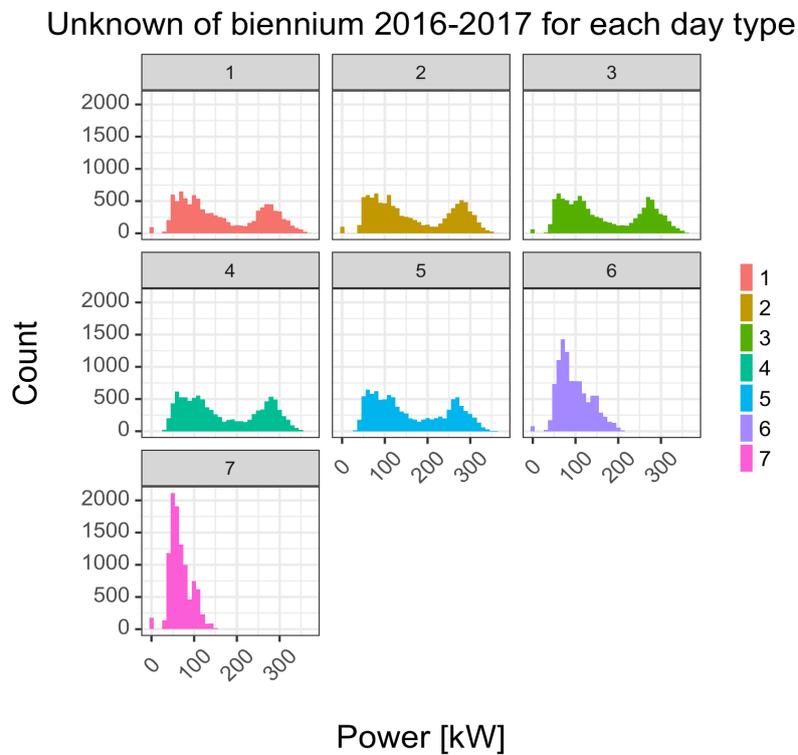


Figure 4.38: Histograms of the not allocated corrected power load for each day type of the biennium 2016-2017.

The DIMAT power consumption appears to be higher during hot months like May, June and July, as shown in Figure 4.40.

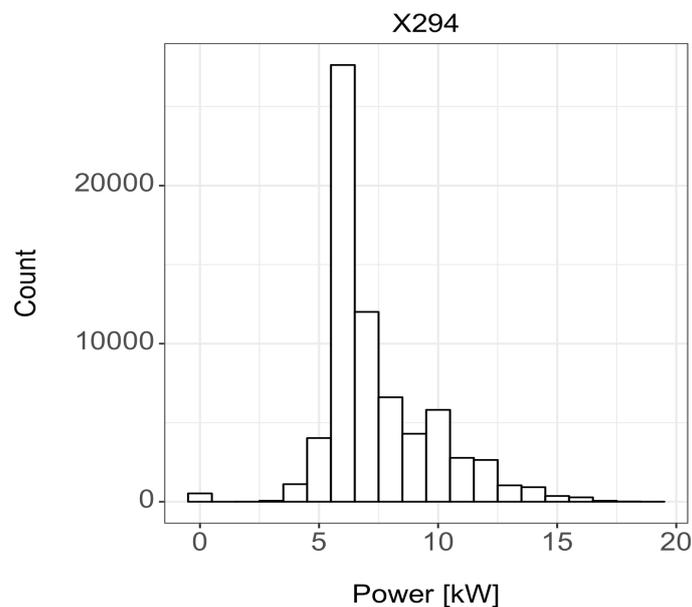


Figure 4.39: Histogram of the DIMAT corrected power load for the biennium 2016-2017.

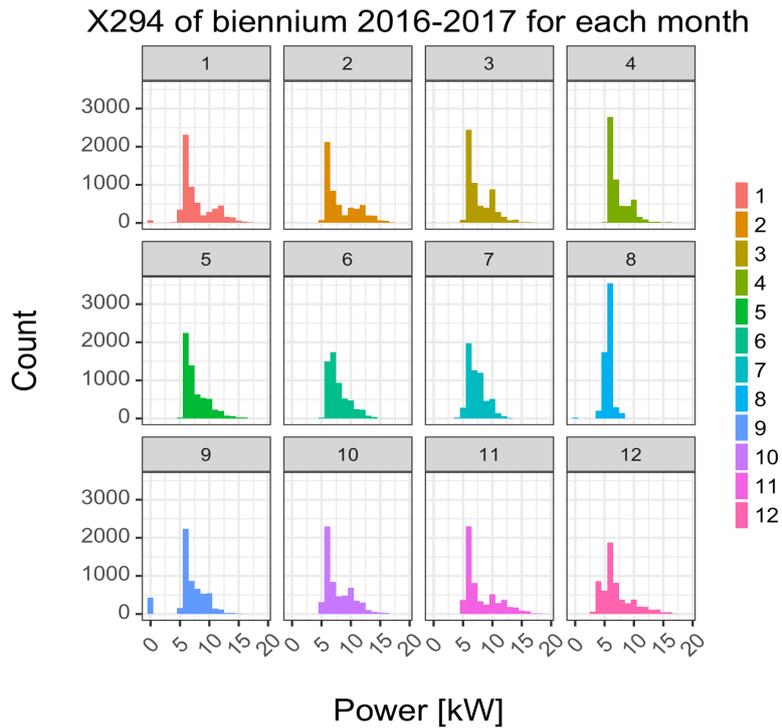


Figure 4.40: Histograms of the DIMAT corrected power load for each month of the biennium 2016-2017.

The DIMAT power load during Mondays is slightly lower compared to the other weekdays, while during the weekend, once again, the load is minimum, as reported in Figure 4.41.

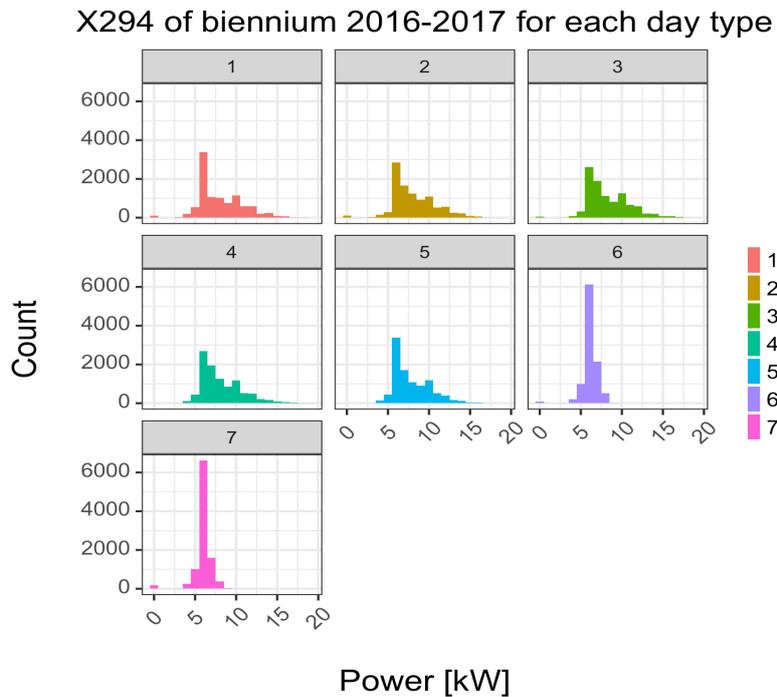


Figure 4.41: Histograms of the DIMAT corrected power load for each day type of the biennium 2016-2017.

From Figure 4.42 one can notice that the refrigeration unit n°2 usually has an electricity demand which stays between 0 and 50 kW, while occasionally it can reach higher values, up to 300 kW.

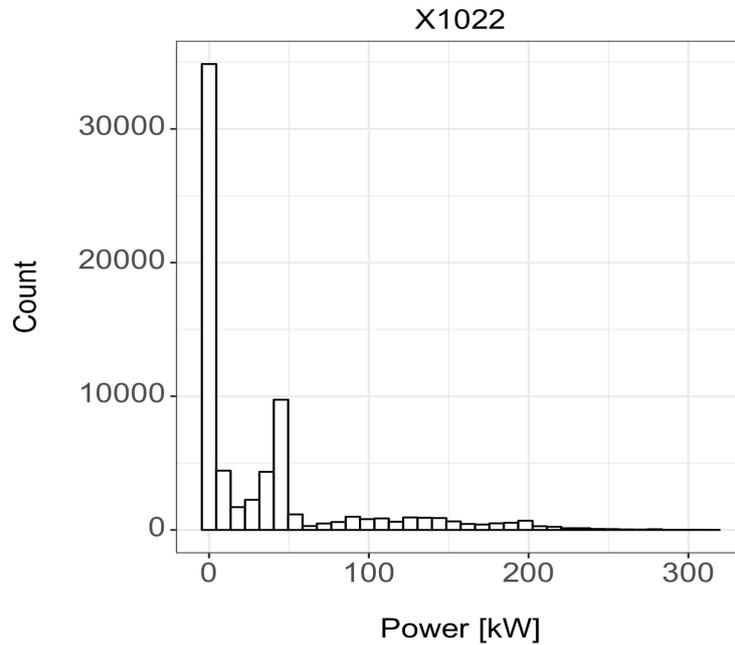


Figure 4.42: Histogram of the refrigeration unit n°2 corrected power load for the biennium 2016-2017.

By looking at Figure 4.43 the reason of such a behaviour can be explained, in fact, only during the months which go from May to September, the refrigeration unit requires more than 50 kW for a certain period of time.

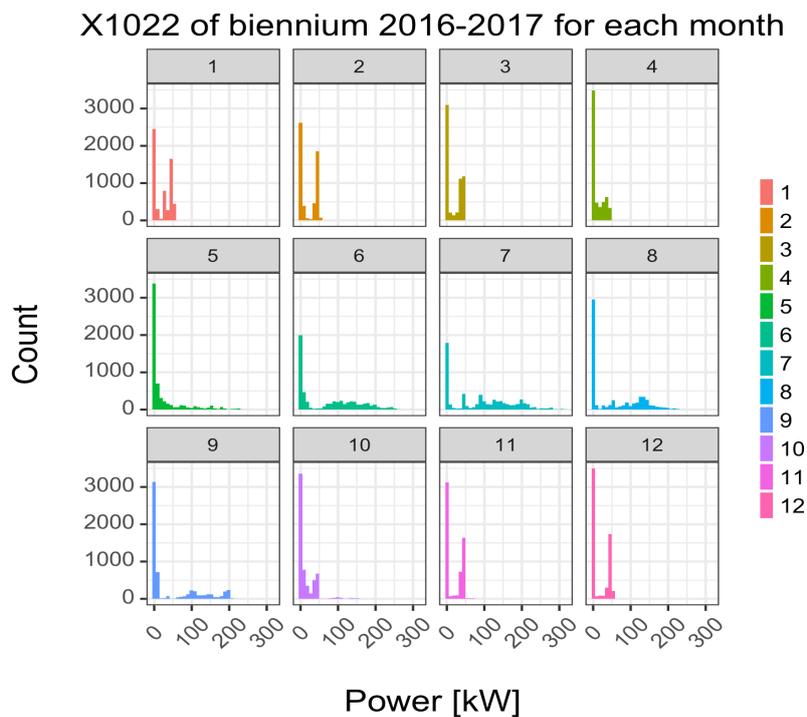


Figure 4.43: Histograms of the refrigeration unit n°2 corrected power load for each month of the biennium 2016-2017.

Also, in this case the power load is not dependent on the weekday, but it decreases during weekends, in particular during Sundays, as shown in Figure 4.44.

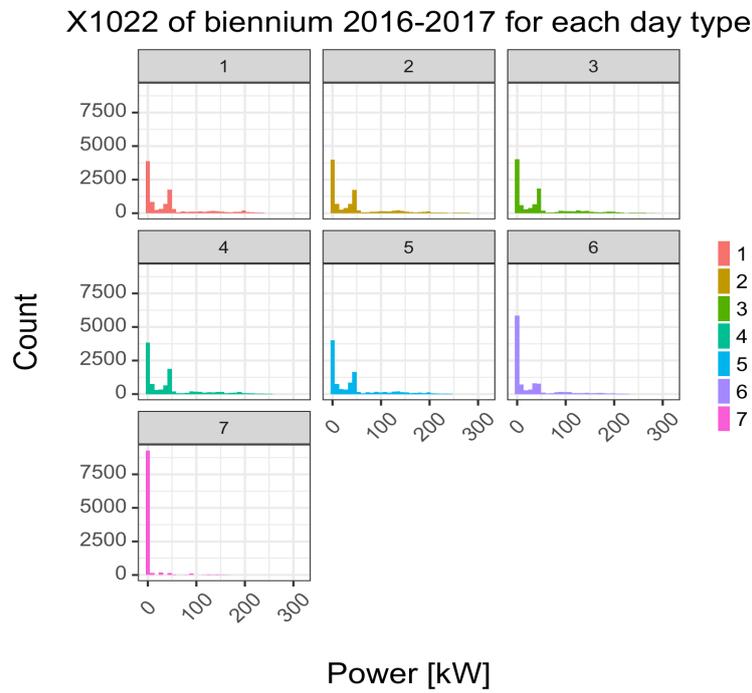


Figure 4.44: Histograms of the refrigeration unit n°2 corrected power load for each day type of the biennium 2016-2017.

The data centre, in Figure 4.45, displays a quite constant power consumption between 30 and 40 kW. Once again, the consumption is higher during hotter months, but in this case, it is maximum in August, since it is not turned off, the demonstration of this can be found in Figure 4.46.

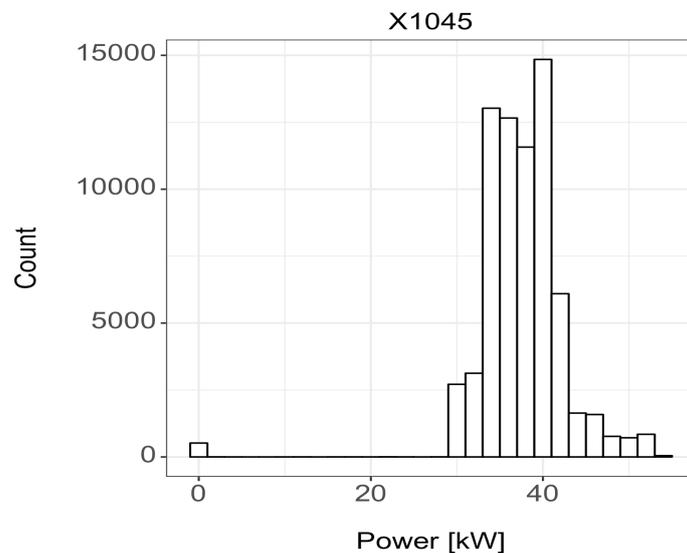


Figure 4.45: Histogram of the data centre corrected power load for the biennium 2016-2017.

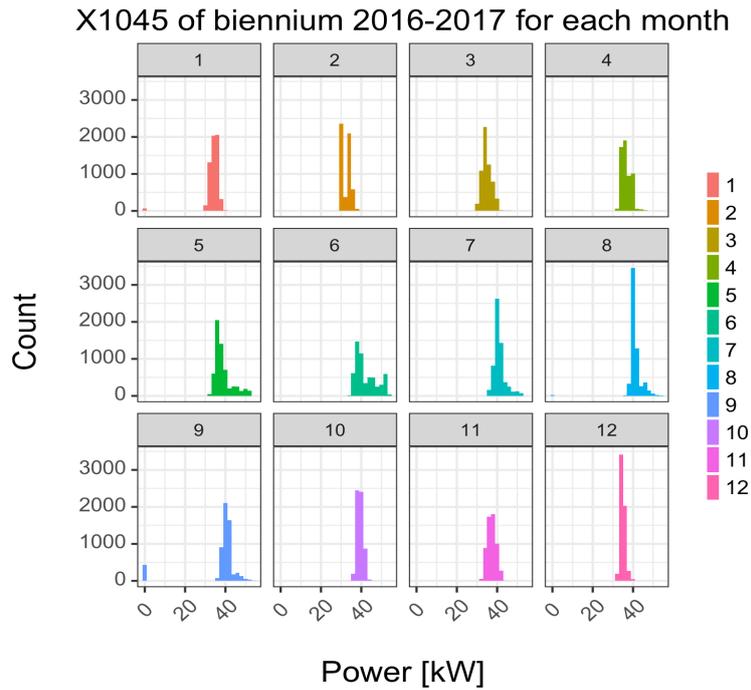


Figure 4.46: Histograms of the data centre corrected power load for each month of the biennium 2016-2017.

For what concerns the daily histograms in Figure 4.47, they clearly show that the data centre requires the same amount of power during weekdays and weekends indistinctly.

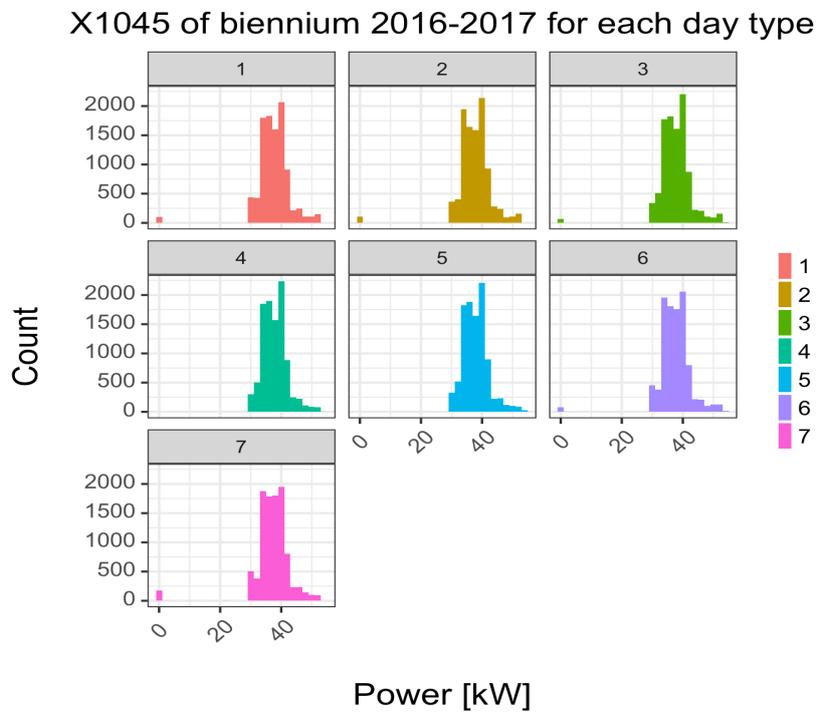


Figure 4.47: Histograms of the data centre corrected power load for each day type of the biennium 2016-2017.

Figure 4.48 represents the histogram of the power load of Bar Ambrogio.

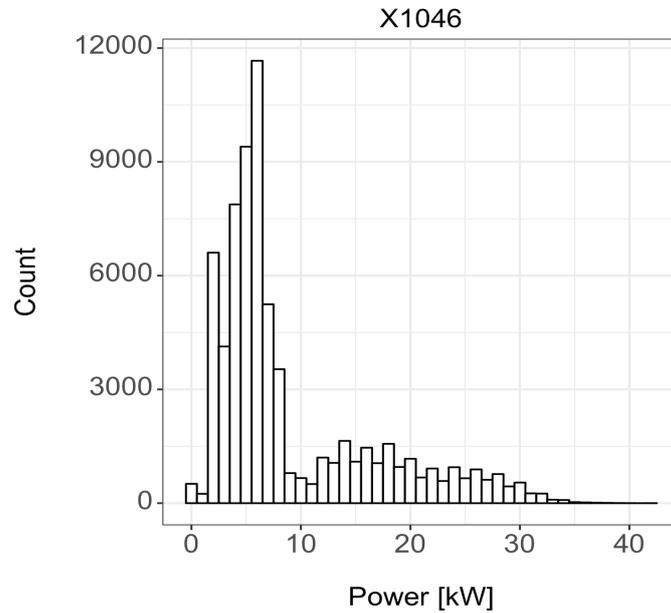


Figure 4.48: Histogram of the Bar Ambrogio corrected power load for the biennium 2016-2017.

As can be seen from Figure 4.49 and Figure 4.50 the consumption does not depend neither on the month nor on the weekday.

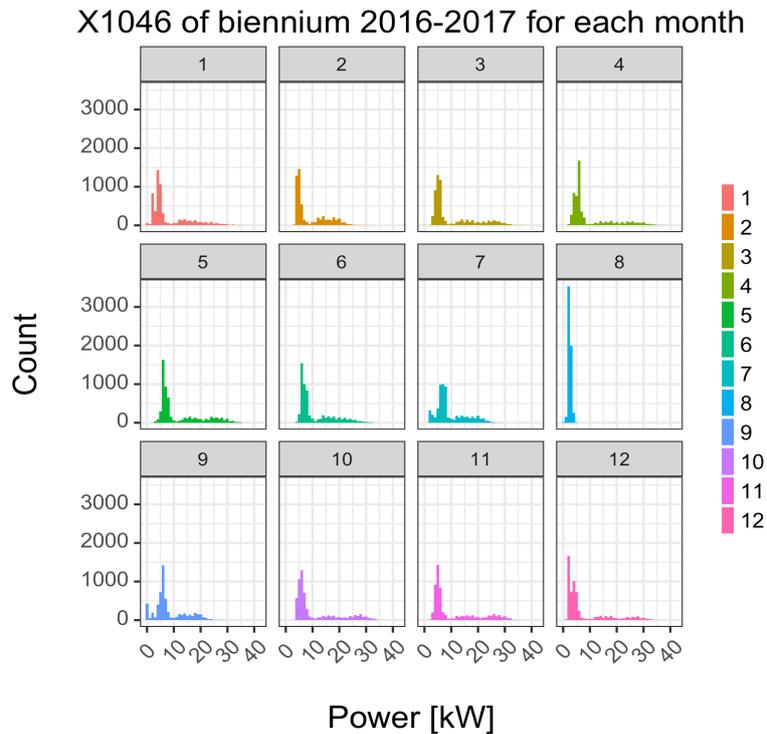


Figure 4.49: Histograms of the Bar Ambrogio corrected power load for each month of the biennium 2016-2017.

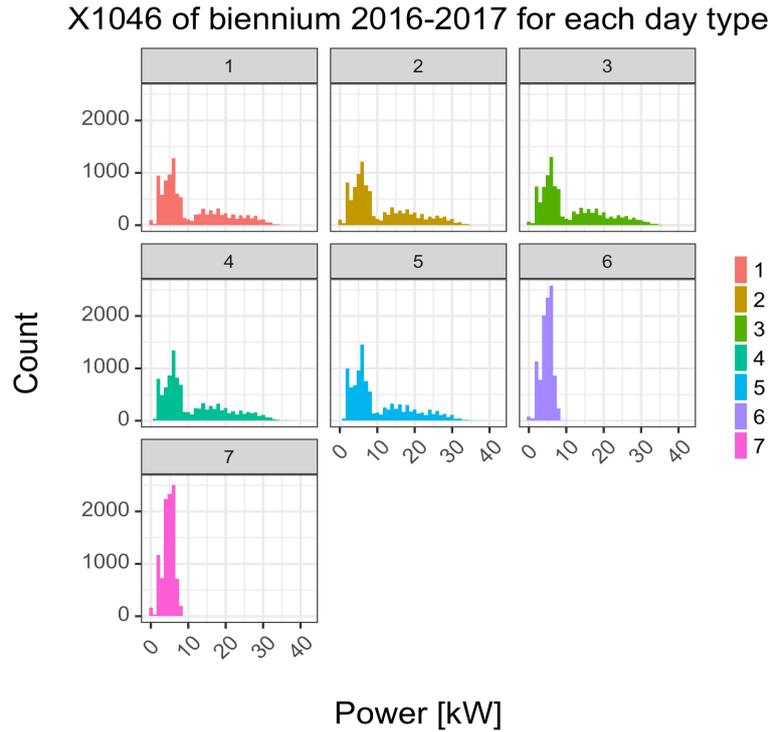


Figure 4.50: Histograms of the Bar Ambrogio corrected power load for each day type of the biennium 2016-2017.

As can be noticed from Figure 4.51 the power load of the canteen has a high probability of being under 50 kW, while between 75 kW and 200 kW it appears to be quite uniformly distributed.

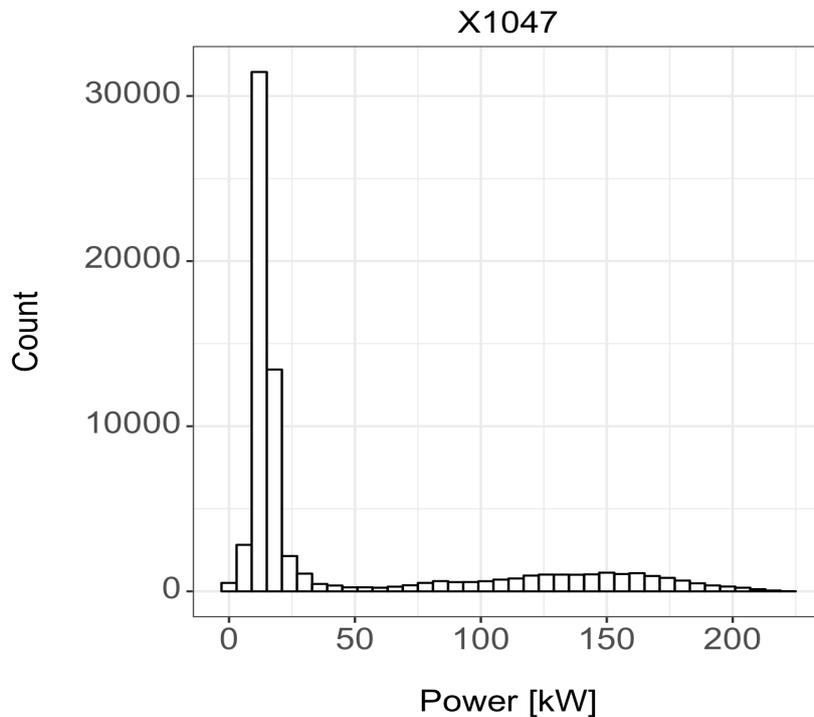


Figure 4.51: Histogram of the canteen corrected power load for the biennium 2016-2017.

Figure 4.52 and Figure 4.53 show that it does not depend on the month and day type, just like the consumption of Bar Ambrogio.

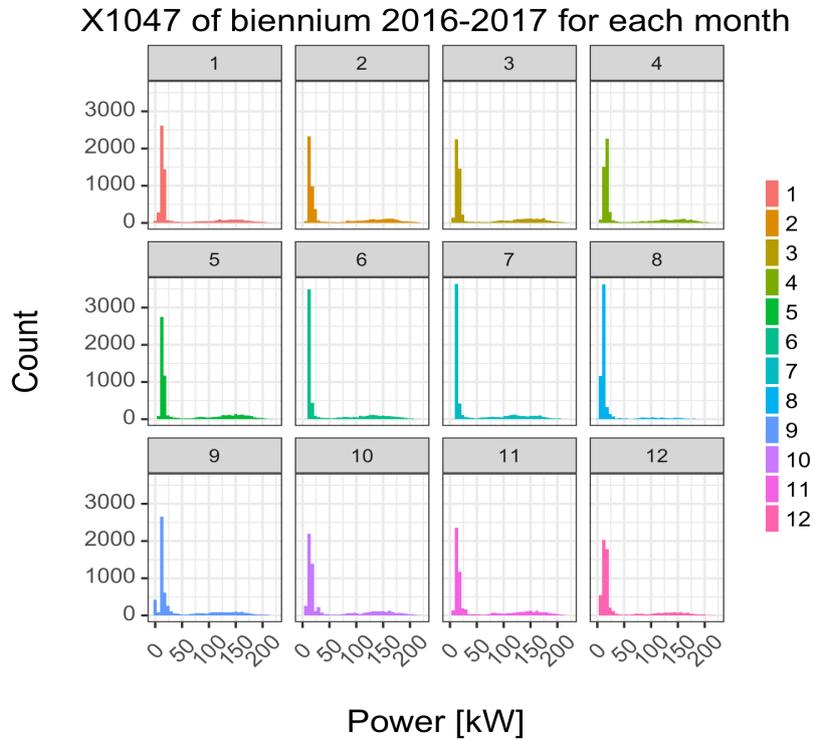


Figure 4.52: Histograms of the canteen corrected power load for each month of the biennium 2016-2017.

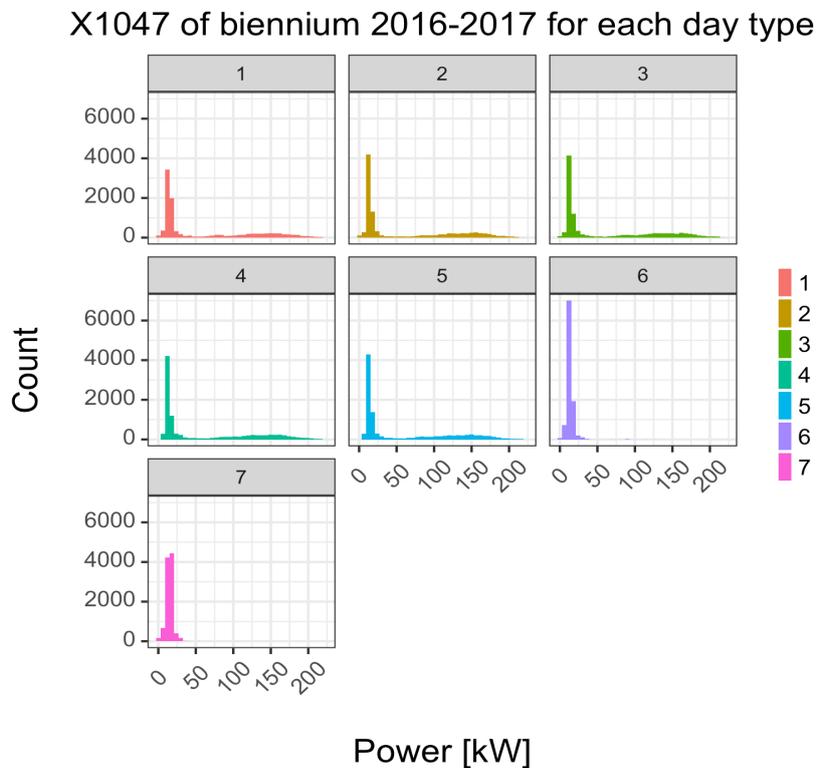


Figure 4.53: Histograms of the canteen corrected power load for each day type of the biennium 2016-2017.

Rectory power consumption is reported in Figure 4.54, Figure 4.55 and Figure 4.56.

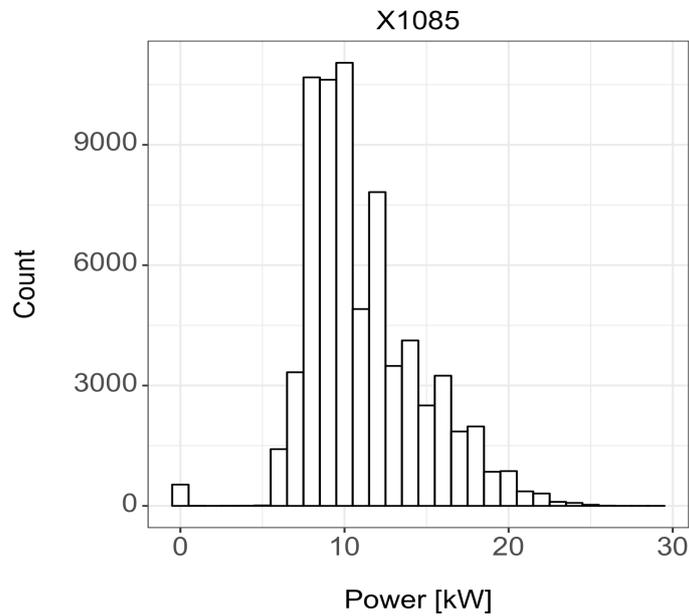


Figure 4.54: Histogram of the rectory corrected power load for the biennium 2016-2017.

In particular, by looking at Figure 4.55 one can observe that the distribution is different in the months of January, February and March.

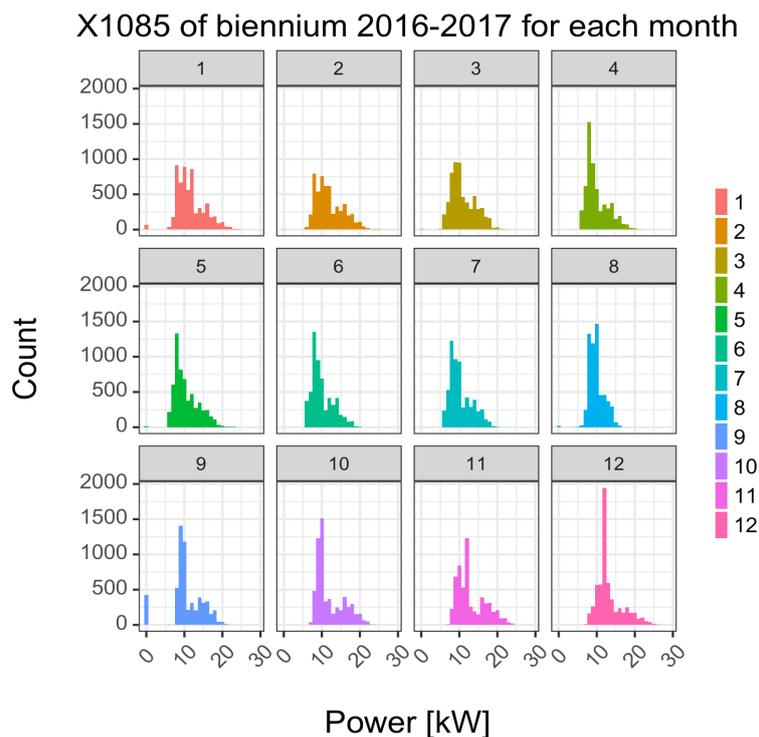


Figure 4.55: Histograms of the rectory corrected power load for each month of the biennium 2016-2017.

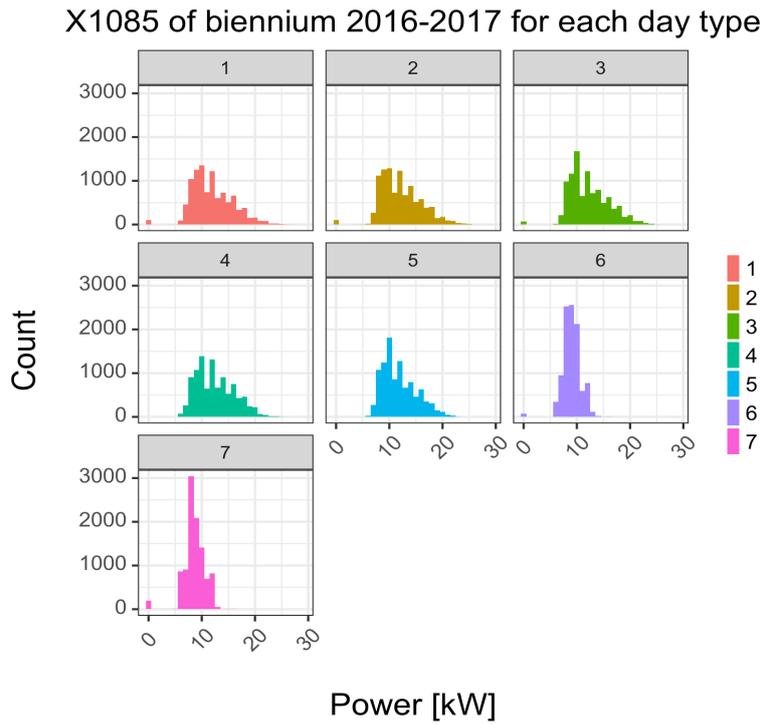


Figure 4.56: Histograms of the rectory corrected power load for each day type of the biennium 2016-2017.

Finally, the power load of the print shop, reported in Figure 4.57, Figure 4.58 and Figure 4.59 seems to be uniform not only during the different months but also during the weekdays, while it is minimum in the weekends.

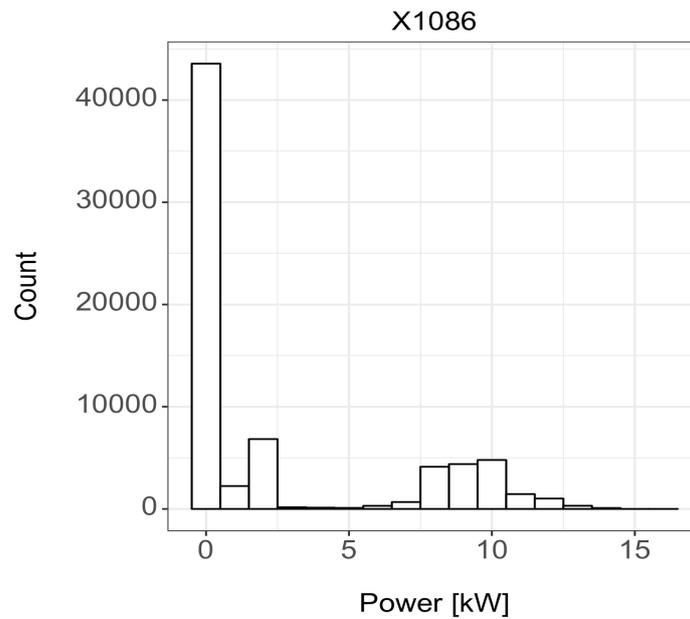


Figure 4.57: Histogram of the print shop corrected power load for the biennium 2016-2017.

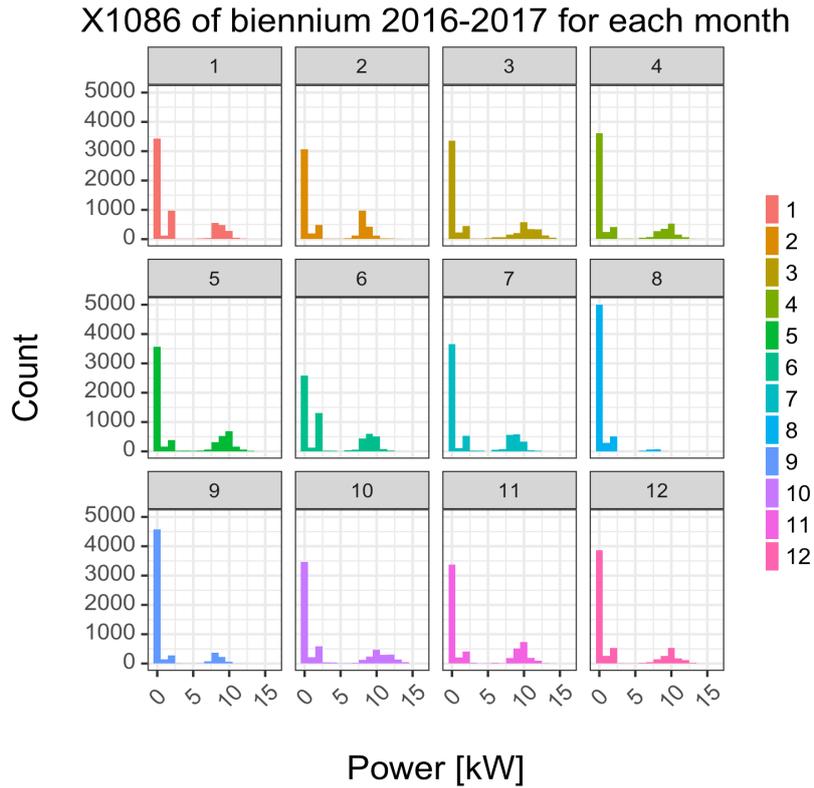


Figure 4.58: Histograms of the print shop corrected power load for each month of the biennium 2016-2017.

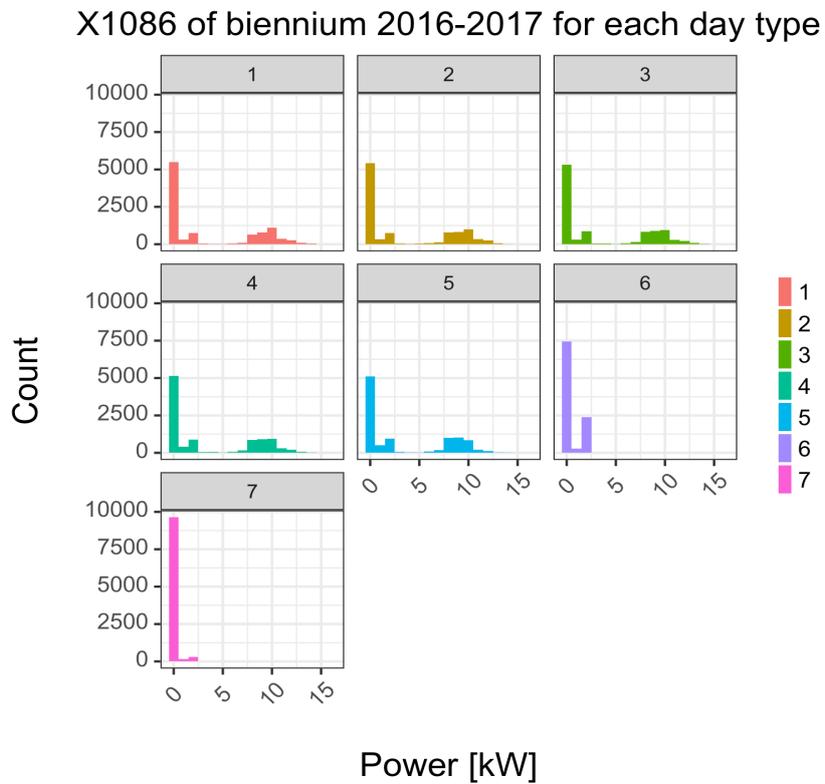


Figure 4.59: Histograms of the print shop corrected power load for each day type of the biennium 2016-2017.

Chapter 5:
Data preparation

5 Data pre-processing

Having at disposal a significant amount of data does not necessarily represent a sufficient starting point to develop a strong building model. In fact, some of them can carry on not representative events or anomalies that have nothing to do with the real behaviour of the system to be modelled. At this point, in order to develop a suitable inverse model, it is mandatory to get rid of those strange data not to affect it with their presence.

This procedure falls into the category of data pre-processing and can be performed in many ways, either by hand or using one of the described data mining techniques. In this case, some data mining techniques will be applied to the data with the intent to discover the presence of anomalies and to categorize them. In this phase, the supervision of a domain expert is necessary in order to distinguish between usual and unusual load profiles and to decide whether some of them should be considered separately or if a lower level of detail could benefit the model development.

5.1 *Clustering analysis*

In this section a clustering analysis is applied to the data presented in chapter 4. The outdoor air temperature data that will be used in this section and in the following ones have been collected by several weather stations located in the surroundings of the building and have been averaged to fill the eventual gaps. The aim of this procedure is to find the hidden correlation existing among the data, in order to mine a significant amount of information which can be used both with a diagnostic purpose and to obtain the right dataset starting from which one can build a reliable model of the building. It is worth noticing that carrying out this procedure manually, without the use of any data mining technique, would require a significantly longer amount of time, and would not necessarily produce sufficiently good results. Capozzoli, Piscitelli and Brandi (60) state that, in order to perform a pattern recognition analysis on time series, it is mandatory to subdivide them as a preliminary step and, moreover, they argue that for most of the application related to the building energy consumption a good choice for such division is represented by daily profiles.

With the previously discussed purposes, the first step of this procedure is to perform a daily profile clustering analysis with a method called “Follow the leader”, firstly introduced in (61). This algorithm is a particular type of hierarchical clustering which does not require the initial choice of the number of clusters, but the selection of a distance threshold called ρ . Firstly k clusters are generated and each cluster can contain a different number of profiles, then an iterative process takes place and each profile is reallocated into the cluster with the more similar behaviour, the

distance ρ can be determined with a trial and error approach (62). In this phase the skills of a domain expert are required in order to determine whether the profiles which fall into the same cluster can be considered as similar or if some subdivision should be done, maybe because of the difference in building operational conditions, in occupancy schedules, in technical building system operation or in energy final use. In Figure 5.1 the results of this procedure are shown.

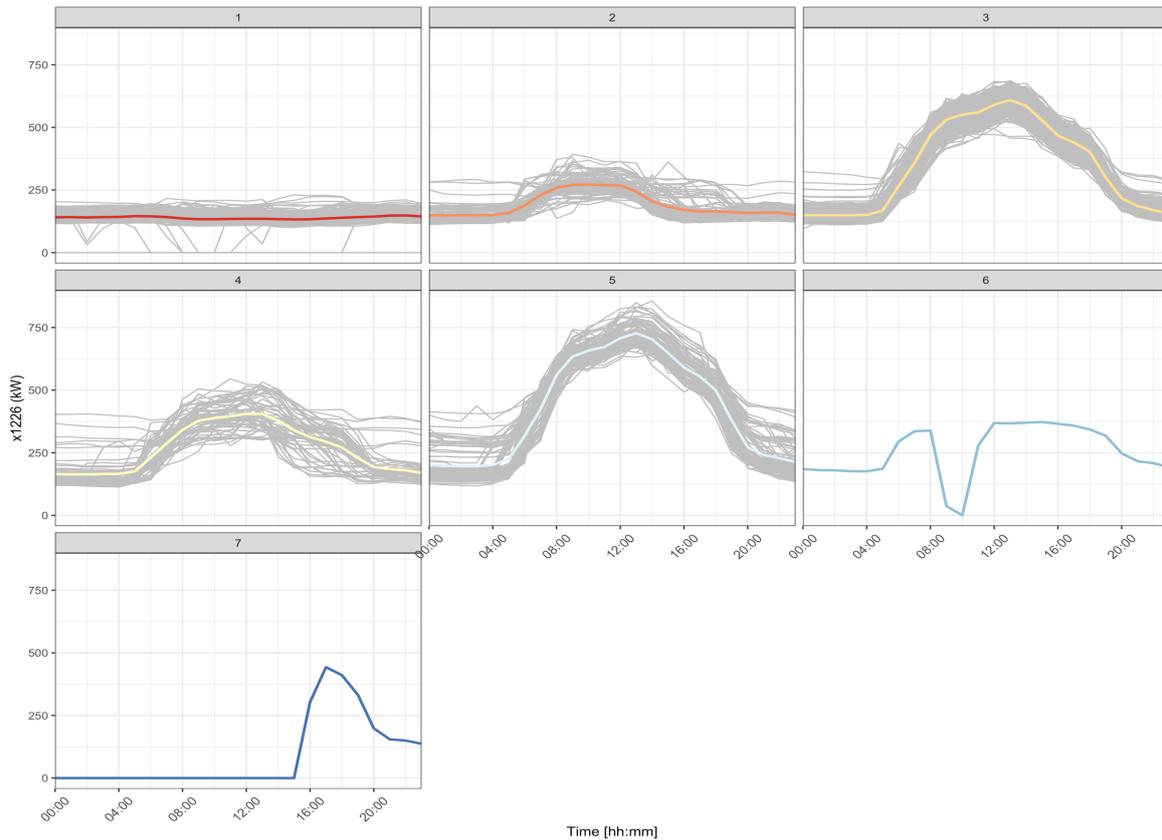


Figure 5.1: Follow the leader clustering results on the total power load.

By setting the threshold parameter $\rho = 720$ the algorithm selects 7 clusters among the profiles of the total power load, as can be seen from Figure 5.1. Clusters 6 and 7 clearly represent acquisition anomalies and the algorithm consistently isolate them. By looking at the others, one can see that none of them is well clustered, since the centroid, symbolised by the coloured profile, is not representative for the totality and some profiles have a great deviation from it, mainly during the night hours. Moreover, one should observe that in each cluster some rare behaviours are present: for instance, in cluster 5 there is a significant number of profiles which during the period which goes from 00:00 a.m. to 05:00 a.m. are stuck at 300 kW, while the majority of them is between 125 and 200 kW.

A good way to examine the temporal distribution of the different clusters is to look at their calendar plot, reported in Figure 5.2. In this figure the colours and the numbers of each cluster are the same of Figure 5.1 in order to make them more readable. At first sight it is clear that cluster 1 is representative for almost the totality of Sundays, while clusters 2 and 4 are located in correspondence of Saturdays, Christmas and Easter holidays. The working days of the heating season are all classified in cluster 3, while during the cooling season there is an alternation of cluster 5 and 4. It is worth noticing that clusters 4, 3 and 5 present a very similar shape, but they have increasing intensity. During the coldest months the power need is lower, so the profiles fall into cluster 3, when the temperature rises the amount of power required by the refrigeration unit increases and so does the total power profile. Finally, during the month of august the occupancy patterns lead to the change from cluster 4 to clusters 3 and 1 in such a way that cannot be explained with the only information about power consumption and climatic variables, which are the ones that are at disposal.

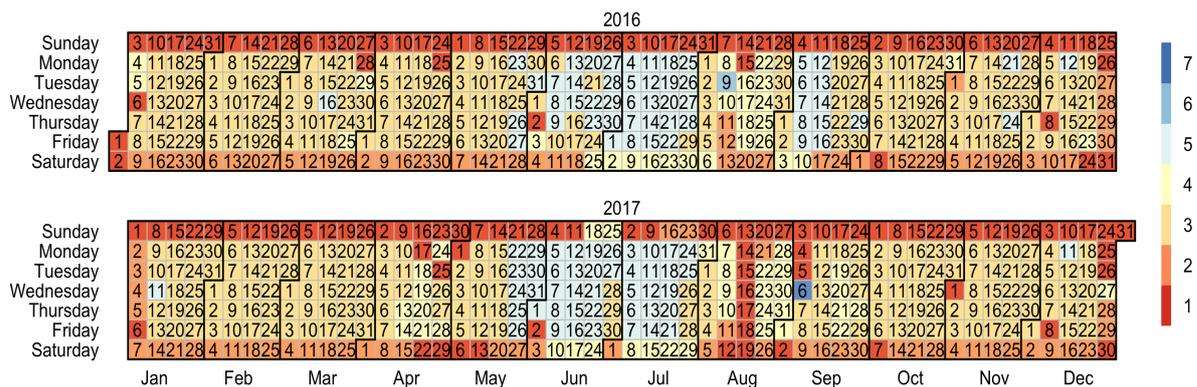


Figure 5.2: Calendar plot of the clusters

In order to investigate the reasons of the previously discussed strange behaviours, a clustering analysis is performed for a second time. A similar result could have been obtained also by changing the parameters of the clustering algorithm, reducing the distance threshold, but, in this way the achieved division is better, since it is able to clearly distinguish each type of profile and to highlight the ambiguous ones, so this is the chosen procedure. This second clustering is carried out individually on each initial cluster, then some consideration is made to analyse the nature of the different profiles. As an example, one could look at Figure 5.3, in which the result of the second clustering analysis performed only on the initial cluster 5 are shown.

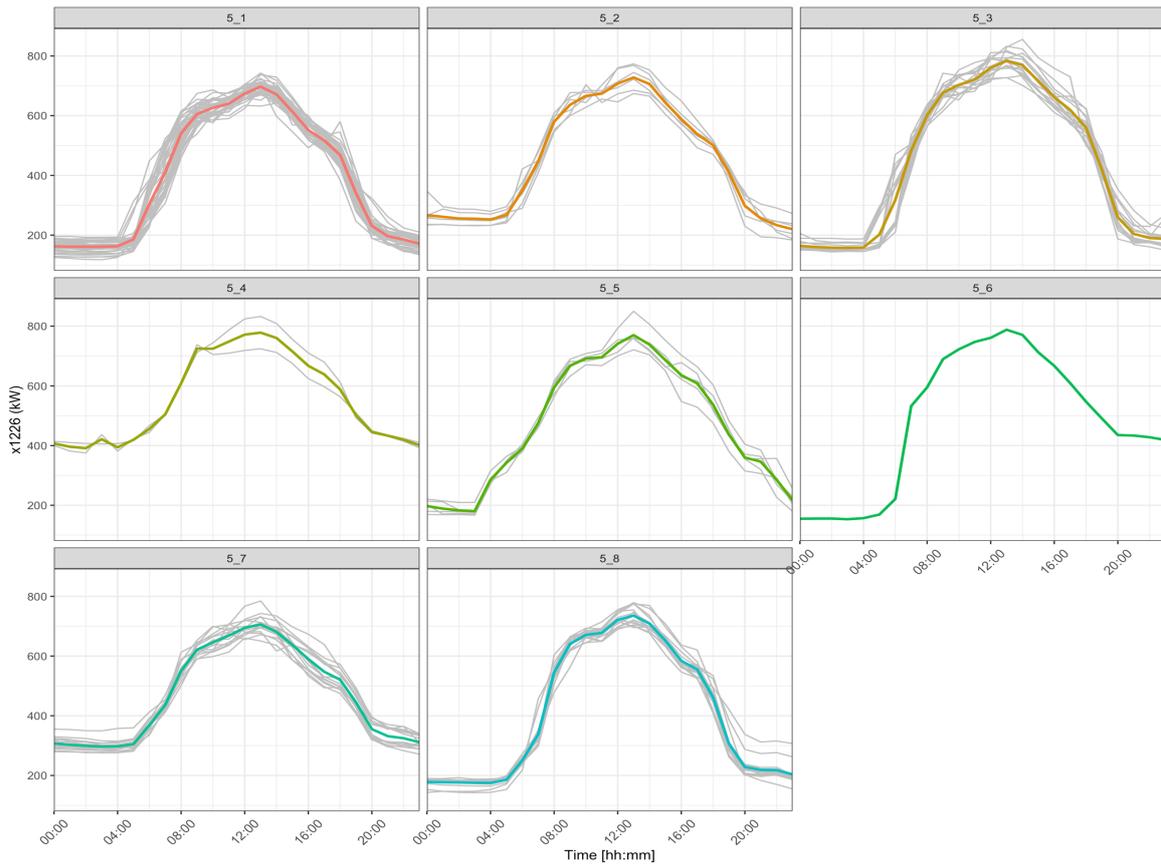


Figure 5.3: Follow the leader clustering performed only on the initial cluster 5

The second clustering procedure performed on cluster 5 identifies 8 new clusters labelled with the nomenclature 5_i with $i = 1, \dots, 8$. The majority of the daily profiles fall into cluster 5₁ and it can be observed that clusters 5₈ and 5₃ are really similar to it. A singular anomalous profile is consigned in cluster 5₆ while the remaining ones have a clearly different behaviour. As stated by Fan, Xiao, Zhao and Wang (63), such anomalies does not necessarily have to be labelled as faults, but they can be regarded as “rare events”. In particular, a profile analysis carried out by a domain expert is necessary to discover the nature of such faults. For instance, let’s consider cluster 5₂: the profiles show a night plateau which is 50 kW above the usual consumption of cluster 5₁. By looking at the partial energy uses the cause of such a behaviour can be easily detected, as shown in Figure 5.4. In this figure, which represent four plots having on the y axis the power load of the refrigeration unit n°2, the upper profiles refer to days with normal or usual operation, while the lower ones are some of the profiles contained in cluster 5₂. Thanks to this plot one can easily see that the refrigeration unit n°2 has not been turned off during the period between 00:00 a.m. and 04:00 a.m., resulting in a higher consumption. For this reason, cluster 5₂ can be labelled as “CF2 night plateau” where CF2 is the Italian acronym for refrigeration unit n°2.

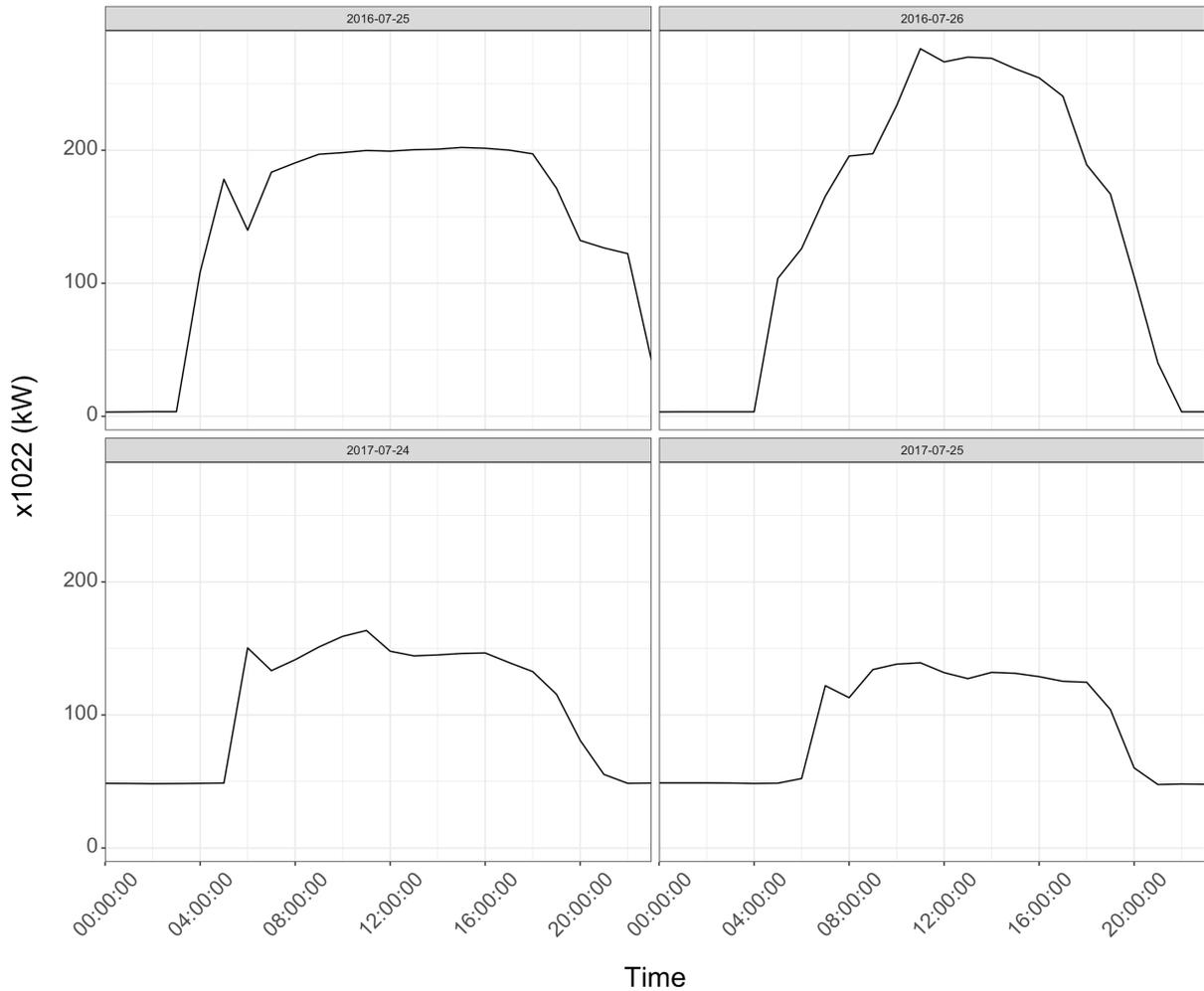


Figure 5.4: Refrigeration unit n°2 daily profiles. The two profiles above refer to usual operation while those under are the rare ones of cluster 5_2.

With an analogous approach also cluster 5_5 can be analysed and the reason of its shape can be recognised as a delayed shutdown of the refrigeration unit, thus it is labelled as “CF2 late off”. For what concerns clusters 5_4 and 5_7 it should be noticed that they are translations of the usual profiles, caused by a flat consumption of the refrigeration unit, as shown in Figure 5.5, for this reason they can be categorised as “CF2 flat”.

Similarly, all the initial clusters have been analysed and some acquisition anomalies and rare events have been identified. For instance, from the second clustering performed on cluster 4, the rare event called “Everything flat” can be identified. Such behaviour corresponds to a power profile which is slightly constant during all day, around 350 kW. The results of this process can be used to construct a rare event and acquisition anomaly library, inspired by the work of Du et al. (41). This library is shown in Figure 5.6.

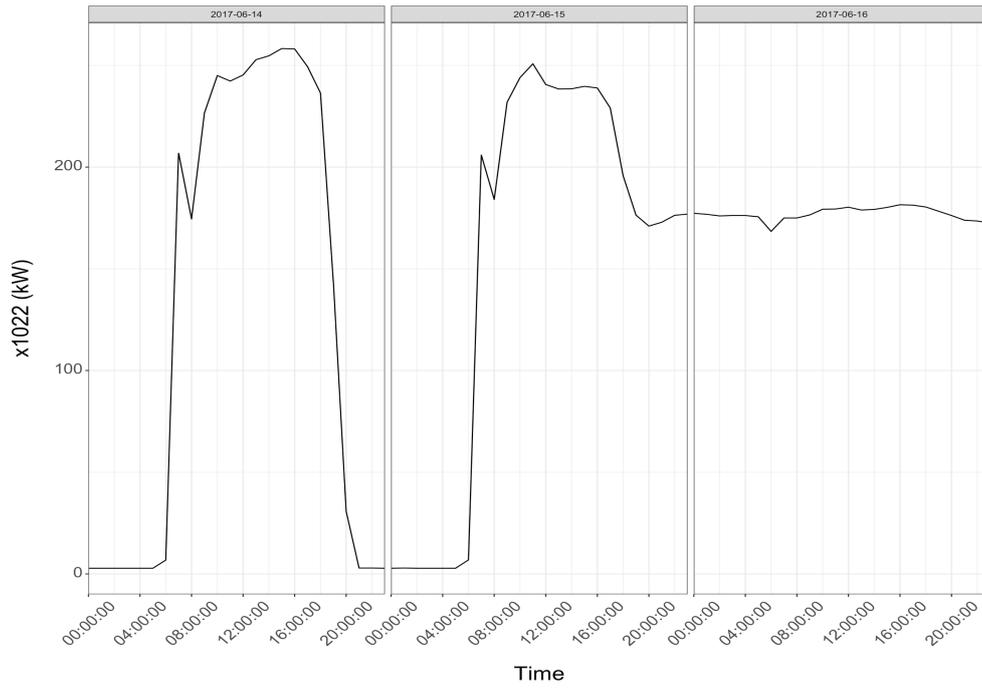


Figure 5.5: Refrigeration unit n°2 daily profiles. The rare event of cluster 5_4 starts between 2017-06-05 and 2017-06-16.

The library contains 55 days divided into 19 days of acquisition anomalies and 36 days of rare or faulty occurrences. The temporal distribution of such events can be visualised in Figure 5.7, where the calendar plot of the library is shown.

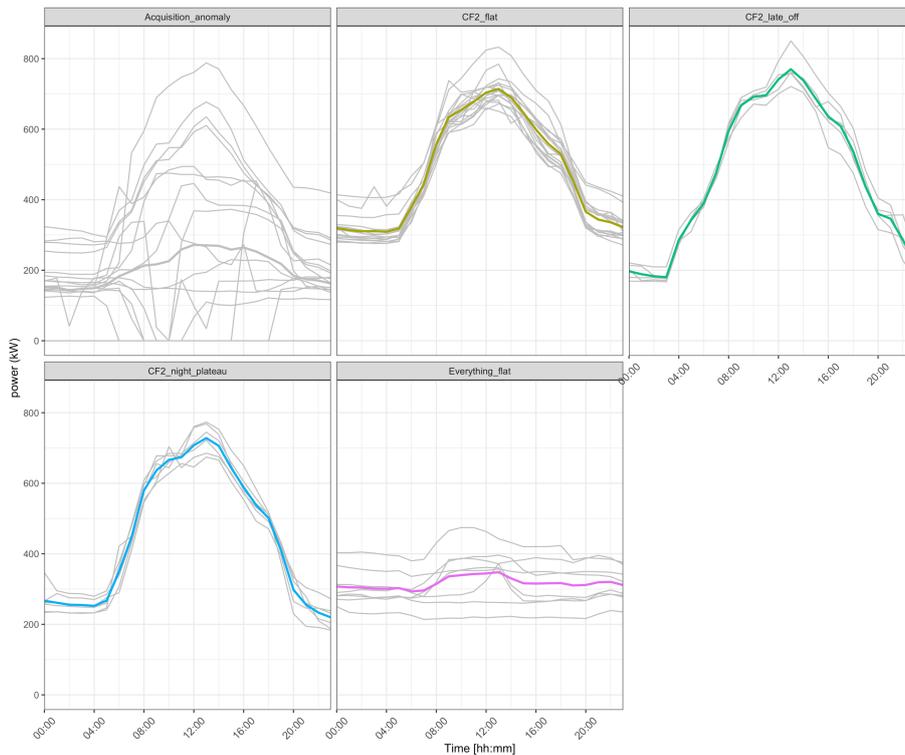


Figure 5.6: Rare events and acquisition anomalies library.

This correlation coefficient is a measure of the linear correlation that exists between X and Y. It is equal to 1 when Y is completely linearly correlated with X and to -1 if completely negatively correlated. When R equals 0 Y and X are not correlated (64). By calculating the correlation coefficient between the clusters, a correlation matrix can be constructed, as in Figure 5.9.

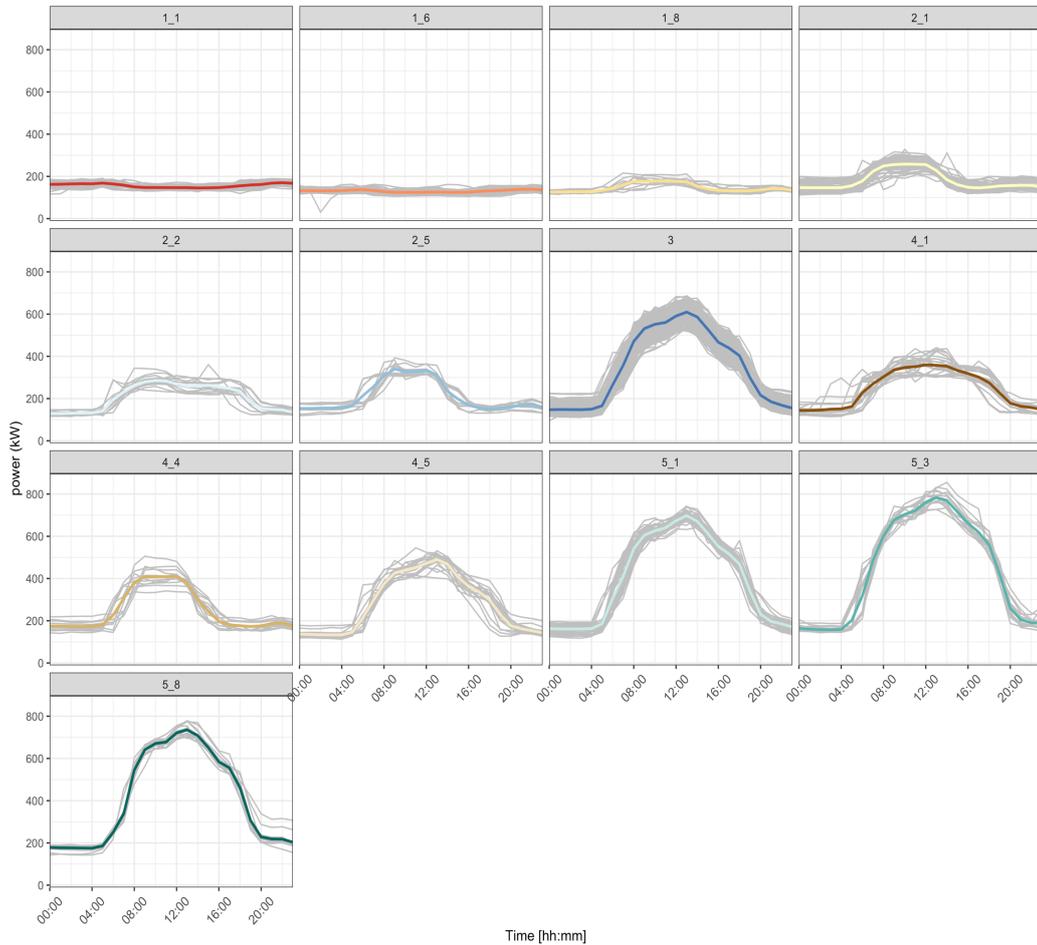


Figure 5.8: Intermediate clustering result without anomalies and rare events.

The correlation matrix is a symmetric matrix which has only 1 on the diagonal, since each cluster is perfectly linearly positively correlated with itself. By entering a specific row of the matrix, one can read the Pearson's correlation coefficient of the cluster represented by the row with the cluster reported in each column. For instance, element (1,3) contains the correlation coefficient between cluster 1_1 and cluster 1_8. It is worth noticing that even if these clusters have initially been assigned to the same cluster, i.e. number 1, they are not strongly correlated. On the contrary, cluster 1_8 seems to be strongly positively correlated with cluster 2_1, as can be also noticed by looking at the two centroid profiles in Figure 5.8.

The analysis of the profiles carried on both with the useful information provided by the Pearson's correlation matrix and by the profiles visual analysis leads to the possibility of joining some of the clusters, reassigning them in a way that is similar to the one described above. This procedure can be seen as a mixture of unsupervised data analytics and domain expertise that, if assembled, can lead to better results than if they are taken individually.

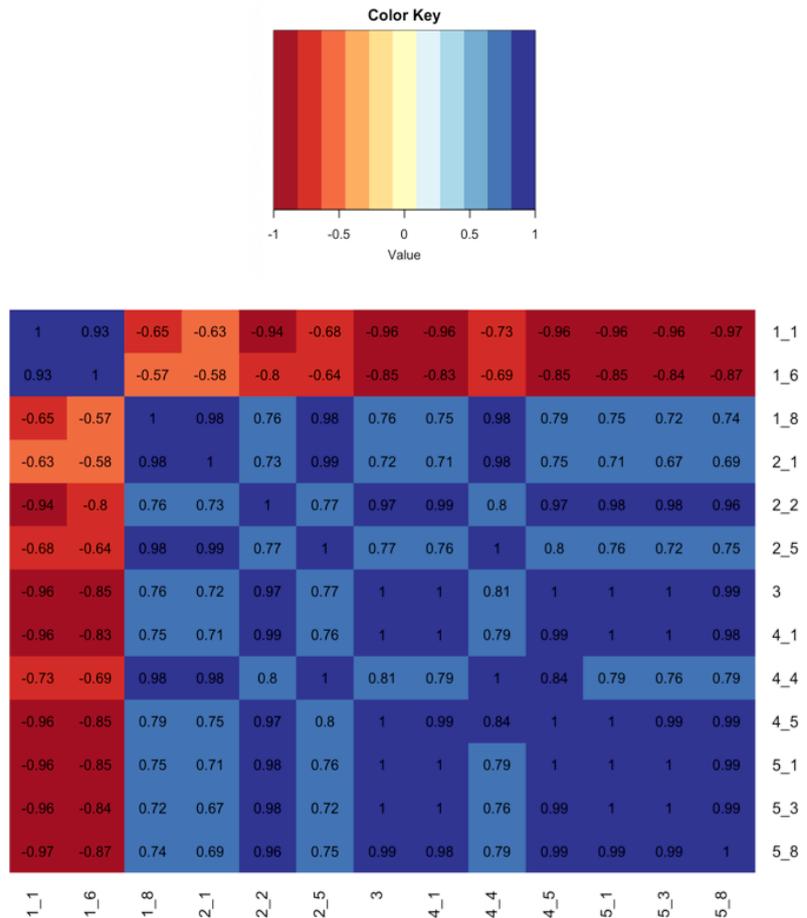


Figure 5.9: Correlation matrix.

In Table 5.1 the summary of the clusters merging process is presented. Apart from the already mentioned cluster 1_8 and 2_1, it is interesting to notice that clusters 1_6 and 1_1 present a very similar behaviour, with a slightly different profile intensity. Moreover, those clusters both refer to Sundays, so it could be a good choice to merge one of them in a unique cluster which takes its name from the reference one, in this case cluster 1_1. Very similar considerations may be done on clusters 2_2 and 4_1. In particular, cluster 2_2 incorporates all those profiles of the initial cluster 2 which ramped down around 05:00 p.m., one of the considerations that have been done to justify

the need of this second clustering analysis. Clusters 2_5 and 4_4 have both a very similar shape and a strong correlation according to Pearson’s matrix. Furthermore, they contain, for a very significant extent, the profiles which start to ramp up at 06:00 a.m. and to ramp down around 01:00 p.m. In the end, clusters 5_8 and 5_1 have the very same shape and a little difference in their maximum value, but for our goals they can be easily considered as a unique group.

Table 5.1: Cluster merging summary

Reference cluster	Merged cluster
1_1	1_6
2_1	1_8
4_1	2_2
4_4	2_5
5_1	5_8

In Figure 5.10 the final clustering results are shown. In particular, eight clusters of non-anomalous data have been identified.

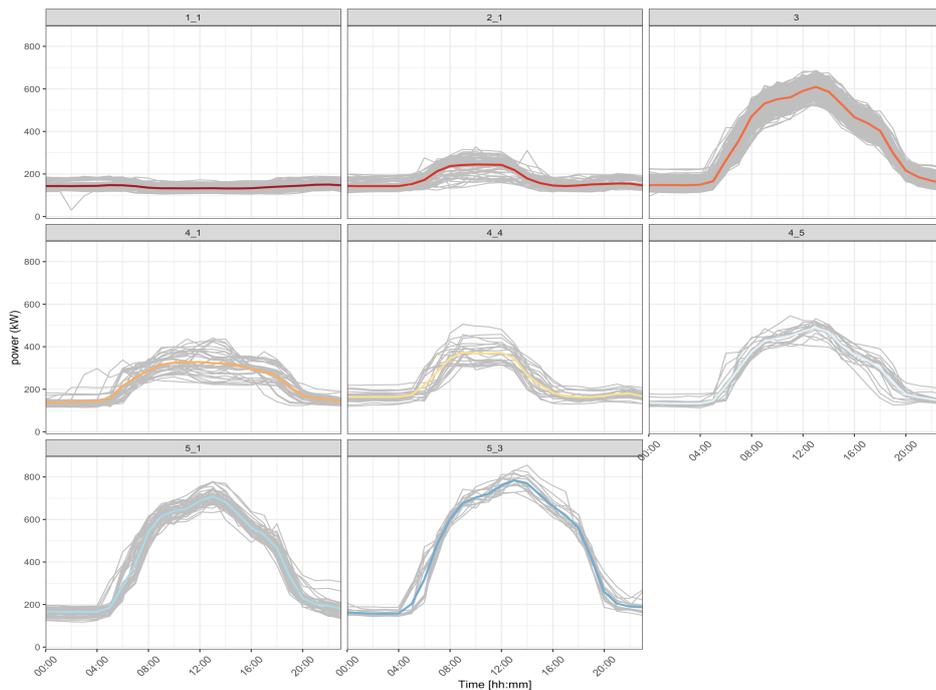


Figure 5.10: Final clustering results on the total power load.

Cluster 3 is the only initial cluster that have been kept in its original form, apart from the removal of some acquisition anomalies. Clusters 4_5 and 5_3 are results of the second clustering process which have not been merged with any of the other clusters. The remaining clusters are the ones who underwent the merging process summarized in Table 5.1.

As for the initial clustering and for the acquisition anomalies and rare events library one could look again at the temporal distribution of the 676 considered profiles and their relative final clusters depicted in Figure 5.11. As before cluster 3 is dominating all the working days during the heating season while Sundays are mainly occupied by cluster 1_1. The important difference between this calendar plot and the one represented in Figure 5.2 is that this time the represented cluster are purer, in fact they do not contain more than one general behaviour and each of them can describe a clear and unique profile. In this way one could have, in advance, an indication for the expected load shape and intensity for a precise day of the year. In conclusion, Saturdays are predominantly referred to clusters 2_1 and 4_4, while the weekdays of the cooling season are divided among clusters 4_5, 5_1 and 5_3, depending on the occupancy profile which we don't have at disposal.

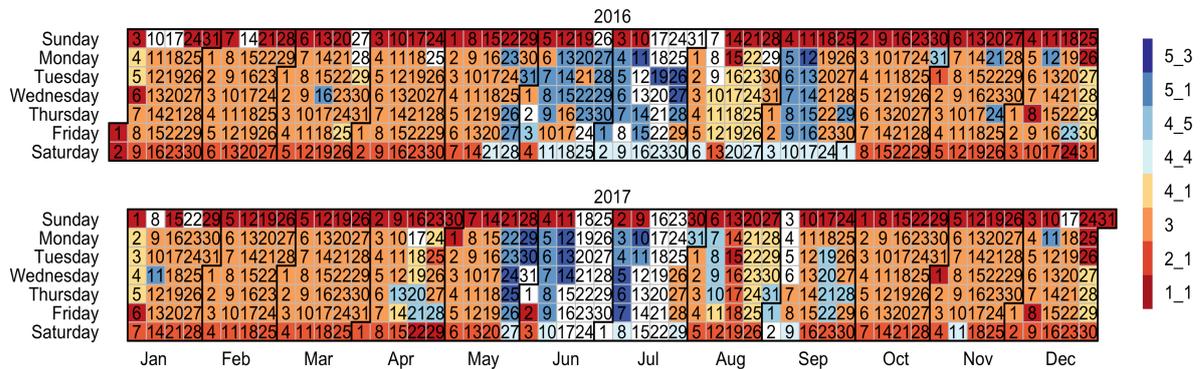


Figure 5.11: Calendar plot of the final clusters. White cells represent anomalies and rare events.

5.2 Classification tree analysis

Another possible way of interpreting the meaning of the discovered clusters is to use a CART, whose theory was presented in section 3.2.3. With this tool it is possible to classify the clusters depending on a chosen set of variables. In Figure 5.12, the classification tree for the final clusters is shown. The classification variables which have been employed are the day type, the month and the mean daily outdoor air temperature. The day type can vary between 1 and 7, where 1 stands for Monday, 2 for Tuesday and so on until 7 which represents both Sundays and holidays. This

tool is very useful if one wants to know in advance which is the probability that in a certain kind of day, the power load will follow the behaviour of a specific cluster. For instance, if we start from the top of the tree and follow the leftmost branches we see that if it is Sunday there is 100% probability of having a profile which stands in cluster 1_1. This information refers to the data with which the clustering and CART analysis were carried out, in fact its meaning is that if we choose a Sunday profile among the analysed ones it will surely fall in cluster 1_1, as can be easily seen also from Figure 5.11. If we assume that the behaviour of the building will be consistent with the one it had during the studied period, we could state that, with good approximation, the Sundays ahead will have a very similar load shape.

If we look again at the CART representation of Figure 5.12, we can observe that Saturdays have two dominant clusters: the first is cluster 2_1, which occurs during coldest days, in which the mean outside temperature is lower than approximately 19.6 °C; the second is cluster 4_4 which is concentrated in those days with higher external air temperature. In these cases, the probability distribution of having a certain cluster is slightly more complicated than in the previous case. In fact, there is a significant 21% of cases in which the air temperature is higher than 19.6°C and the cluster is 2_1 anyway. For all the working days, instead, if the outside temperature stands below 20.5 °C the expected cluster is 3 with a probability of 83% among the 360 considered working days, while if it is higher, a distinction between the months should be done. More precisely, during the months which go from May to September, except for August, the predominant clusters are 5_1 and 5_3. The first has a 70% probability of occurrence when the outdoor air temperature stands below 25.4 °C, while if it is higher there is a 55% of having a day in cluster 5_3. As one should notice, the probabilities in such months are definitely lower than in the previously discussed cases. This can be explained if we notice that the information about day, month and outdoor air temperature are not sufficient to describe and justify the difference between a cluster and another, during these months. In fact, there are sudden changes from a day and the following and significant differences between a year and another which could be only justified having at disposal some information about the occupancy patterns. For instance, during the months of July and August the power load due to the canteen can have a drastic influence on the total power profile and such behaviour can be predicted or expected only having occupancy information. The very same considerations can be done for the month of August, in which if the outdoor air temperature rises above 26.2 °C there is a 39% probability of falling in cluster 3, a 32% probability of occurrence of cluster 4_1 and a 19% for cluster 2_1. When the air temperature decreases, cluster 4_1 has a 58% percent probability of occurrence while clusters 3 and 4_5 follow with respectively 20% and 18%.

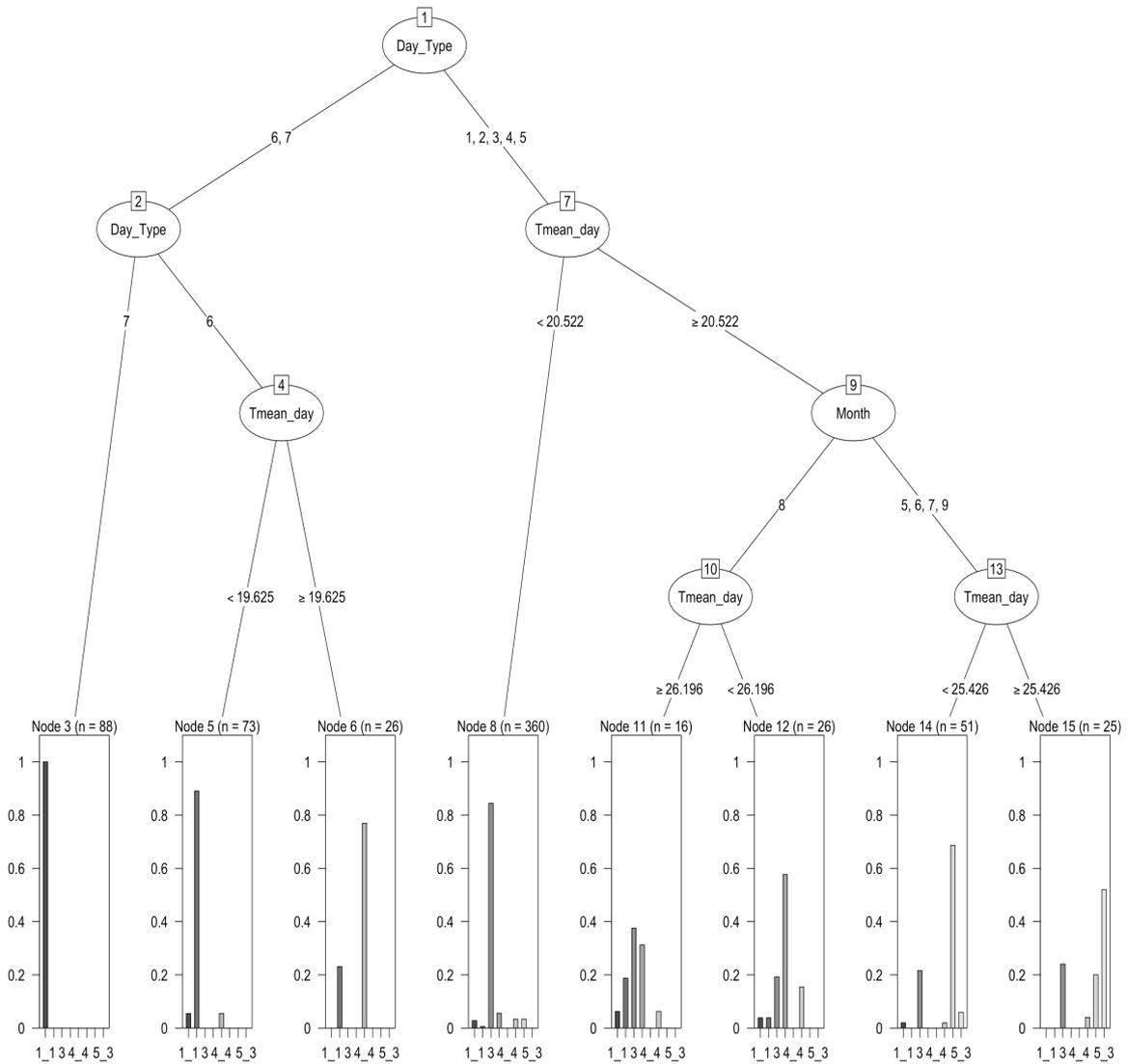


Figure 5.12: Clusters classification tree

As a final consideration it should be said that the developed model has a definitely high precision on weekends, holidays and working days of the heating season, while if the clustering analysis was able to effectively separate the behaviours which occurred during the cooling season, the CART was not equally good in describing their occurrence because of the lack of information about occupancy. Fortunately, thanks to the visual analytics and the calendar plots we were able to clearly describe their distribution and to go beyond the simple clustering results which, without the right interpretation, could lack of some really fundamental information.

5.3 Refrigeration unit analysis

Consistently with the analysis carried out on the total consumption, in this section, the fault-free data which have been identified are used to generate an archetypal load profiles classification for the refrigeration unit only, starting from the methodology proposed by Capozzoli, Piscitelli and Brandi (60).

From initial clustering analysis carried out on the refrigeration unit consumption, it was noticed that it presented very similar load shapes with different intensity levels. For this reason, a normalization was carried out on the daily profiles so that each of them was enclosed in the interval [0,1]. Then a hierarchical clustering analysis using Ward's linkage method was performed on the normalised profiles, and the number of clusters was set to 3. Finally, a classification tree was implemented, using as explanatory variable the day type. The results are shown in Figure 5.13, where both the CART and the clusters are reported.

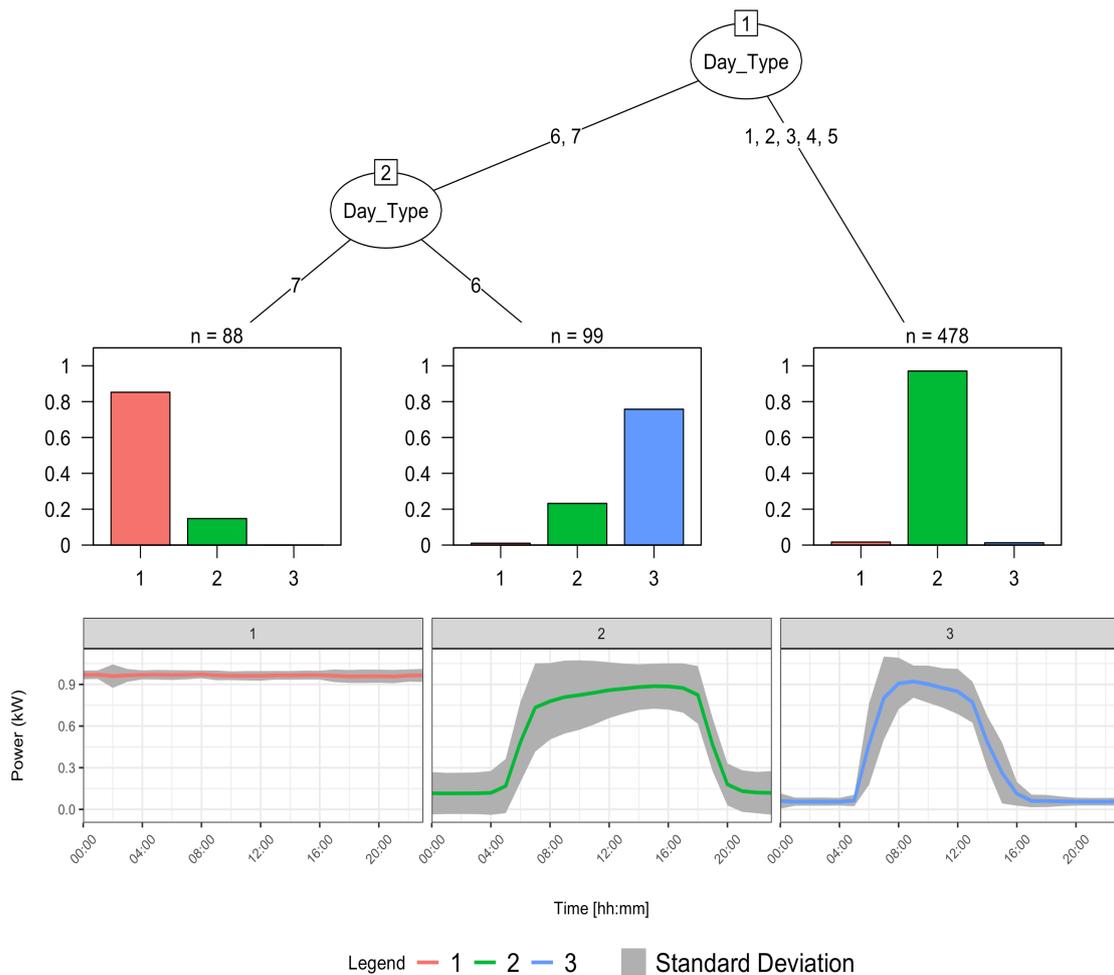


Figure 5.13: CART of the normalised clusters for refrigeration unit n°2 (above). Refrigeration unit normalised centroids with relative interval of two times the cluster standard deviation (below).

For what concerns Sundays, they are well described, for more than the 80% by cluster 1, as could have been expected against the background of the previous studies carried out on the total power load. On the other hand, Saturdays are mostly linked to cluster 3, whose shape tends to decrease in the early afternoon. During all the weekdays, the reference profile is the centroid 2, which ramps up around 05:00 a.m. and ramps down at about 05:00 p.m. As it can be noticed the two standard deviation range represented by the grey area is quite large, even if the general behaviour is well summarised by the green centroid. This suggested to perform a second clustering analysis which considered also the real profiles intensity restricting the data only to cluster 2.

Once again, a hierarchical clustering analysis using Ward's linkage method was performed, this time only on the initial cluster 2 and a number of three final cluster was chosen as a good way to summarise the differences in intensity enclosed in the initial one. Figure 5.14 represents the results of the clustering analysis as well as a classification tree having as explanatory variables the mean daily outdoor air temperature and the month.

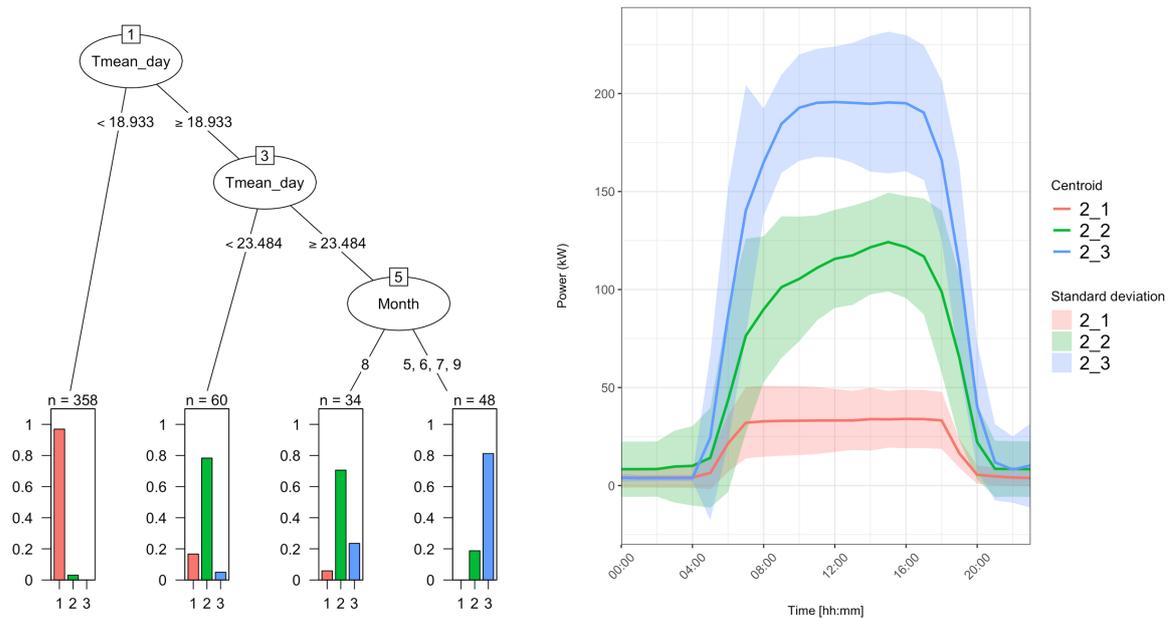


Figure 5.14: CART of cluster 2 profiles intensity (left). Cluster 2 profiles intensity with relative interval of two times the sub-cluster standard deviation (right).

The three profiles are well distanced during the operating hours and the CART is able to correctly distinguish them. In fact, almost the totality of the profiles belonging to cluster 2_1 is referred to cold days with average outdoor air temperature below 18.9 °C and which have a very low power consumption. Such days probably include the consumption of the hot water circuit pumps as well

as the consumption related to cooling season days with very low outdoor air temperature but still with the need of cooling, maybe because of the internal heat gains. The rest of the profiles belongs to cluster 2_2 when the temperature rises but is still below 23.5 °C, corresponding to mid-season days where the consumption is still a middle ground between high and low consumption. Finally, during hotter days, if the month is August the cluster is again 2_2, probably because of the low occupancy caused by the summer vacations, while for the months of May, June, July and September the consumption is the highest, corresponding to cluster 2_3. Such behaviour is consistent with the high thermal load that have to be managed by the refrigeration unit during those very hot days.

In conclusion, it can be said that the proposed analysis carried out on the refrigeration unit consumption only, was able to identify three principal load shapes, corresponding to Sundays, where the consumption is flat, Saturdays, when it ramps down around 01:00 p.m., probably because of the low occupancy and the time scheduling, and weekdays, in which the unit is operated from 05:00 a.m. to 05:00 p.m. Moreover, by a further analysis performed on the largest cluster, corresponding to the weekdays one, three intensity levels have been identified, corresponding to cold days, mid-season days and hot days. Each of them has been well classified by a CART which with the only information about the month and the outdoor air temperature was able to explain the magnitude difference from a profile to another. Such methodology can be useful to monitor the refrigeration unit operation and to think about possible operational improvements, maybe regarding the shut-down or the switch-on times of the refrigeration system as well as its operation during cold days in which also the possibility of free cooling could be exploited.

Chapter 6:

Results: anomaly detection and energy
benchmarking

6 Anomaly detection and energy benchmarking

In this section the development of an anomaly and rare events detection methodology is presented. This model can be also used to provide a benchmark between expected and measured electrical energy consumption. Capozzoli et al. (65), state that such tools can be employed to support the interested parties to analyse the building behaviour, to delineate suitable administration policies and to communicate to the users the presence of anomalous or rare events.

At first the description of the steps which lead to the model implementation are described, then some applications are shown to help in better understanding the potentialities that it implies. Finally, the results of this method are presented in accordance with the data analysed in section 5 and its performance is quantified.

6.1 General framework

The main purpose of this model is to develop a benchmarking profile which, if compared with the real load profile, could underline the anomalies and the rare events. Under this perspective, it is mandatory that the abovementioned model does not follow the last changes of power but that it gives as an output the load shape as it should have been in case of usual operation. To know if a profile should be considered usual or not, is not the task of this model, in fact, since it will be trained only with the data that have been identified as usual in the previous section, through the use of clustering analysis and domain expertise, it will only know those data that we labelled as fault-free. For this reason, the model will not be able to mimic the behaviour of anomalies and, not having at disposal any information about previous hourly power load it will not follow the anomalies or the rare events.

The general framework of the model is summarized in Figure 6.1. The process starts with the estimation of a profile which works as a benchmarking model. At first the model has been developed with a single non-autoregressive artificial neural network, but some more information was needed to make it more precise. So, it was useful to develop a regressive CART model whose output was subsequently fed into the ANN, making it prediction far more accurate, with a decrease of about 4 percentage points of the mean absolute percentage error (MAPE) in training, whose formula is reported in Equation 6.1, where n is the number of observations, x is the actual value and \tilde{x} is the predicted one.

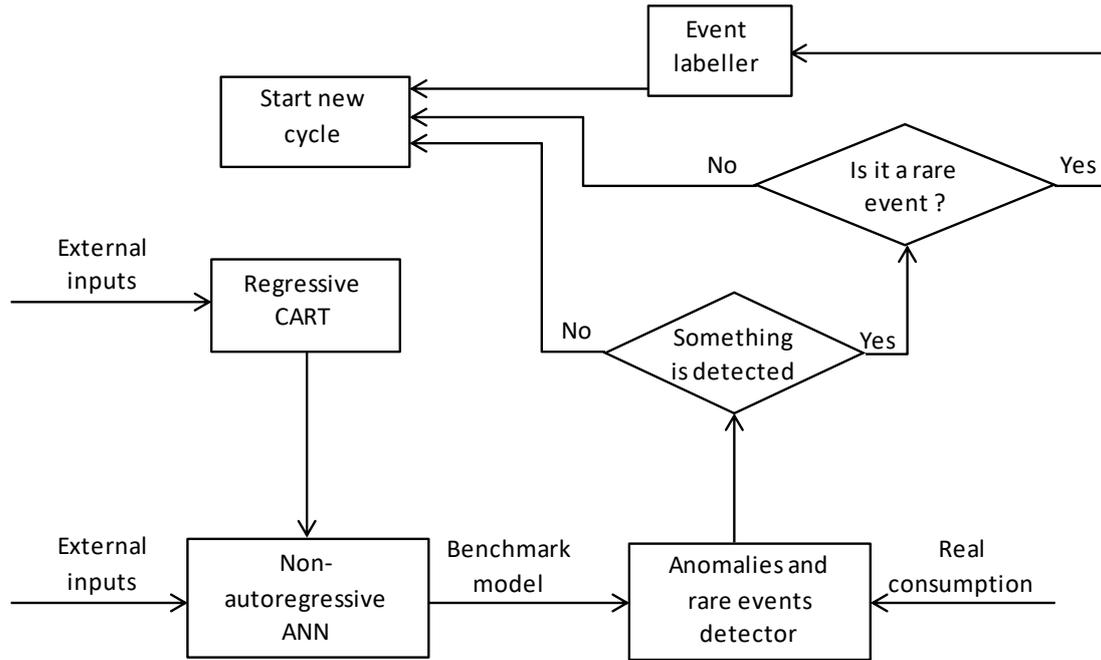


Figure 6.1: Model general framework.

Then the results provided by the joint action of the CART and the ANN are fed into an anomalies and rare events detector which receives also the real consumption data of an entire day. From the comparison between actual and expected data the detector identifies the presence of anomalies and rare events. In the case of a measurement anomaly detection this is simply counted and displayed as such. If instead a rare event is detected it passes through an event labeller which compares it with the known library of faults and select the right category for it.

$$MAPE = \frac{1}{n} \sum_{k=1}^n \left| \frac{x_k - \tilde{x}_k}{x_k} \right| \quad (6.1)$$

6.2 Regressive CART implementation

The first step of the fault and anomalies detection model is the prediction of the electrical energy consumption of the next day. This prediction is made by means of a regressive CART, whose structure can be seen in Figure 6.2. The predictor variables are the same of the classification tree of section 5.2, but this time the purpose is completely different. Using the predicted energy consumption of a certain day as an input for the benchmark ANN, gives the opportunity of knowing in advance if that precise day will have a significative power need, or if it will demand a quite low amount of electricity. Furthermore, this tool broadly divides the days into 6 categories which present different levels of energy consumption, according to simple variables which are the day type, the month and the daily mean outdoor air temperature. The first two of them are straight

forward, while the third can be easily acquired from a weather forecast service with very high accuracy or can be obtained by measurements if the estimation is done at the end of the day.



Figure 6.2: Regressive CART.

6.3 Non-autoregressive artificial neural network

In this section the construction of the non-autoregressive artificial neural network is described. We refer to this network as non-autoregressive since it has no input information about past hourly power consumption but still it is aimed at predicting the hourly load profile for an entire day. The main task of this ANN is to learn the function that relates external inputs and energy consumption with the load profile. Under this perspective this tool can be regarded as a model trying to estimate the power consumption of the building without taking into consideration the real past evolution of its power profile.

If we look at Figure 6.3, we can see the general structure of this network. It is a feedforward deep neural network with an input layer containing 5 input neurons, two hidden layers, whose neurons number decision process will be described later, and a single neuron output layer. The hidden neurons employ the rectifier activation function, thus they can be seen as ReLUs (rectified linear units), while the output layer employs a linear activation function. The general formulation of the rectifier function, defined as the positive part of x , is reported in Equation 6.2.

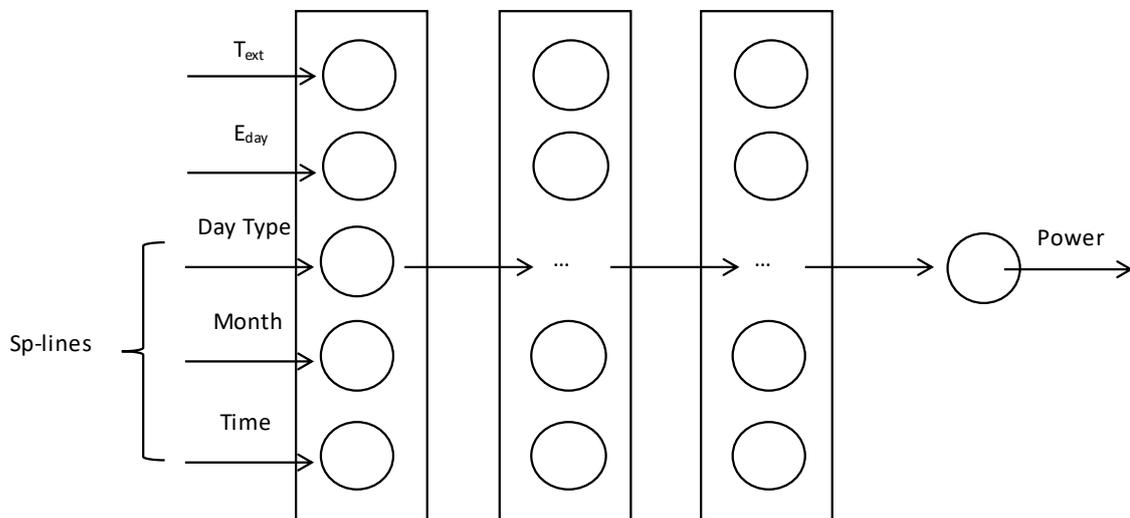


Figure 6.3: Non-autoregressive artificial neural network structure.

One of the model inputs is the mean hourly outdoor air temperature T_{ext} of the hour at which the prediction is carried out: once again this temperature can be obtained from a weather forecast service with very high accuracy, or it can be assumed as known, as in this case study, since the

power estimation can be performed at the end of the day, to verify if some anomalous or rare event has occurred.

$$f(x) = x^+ = \max(0, x) \quad (6.2)$$

The expected daily electrical energy consumption of the day at which the estimation is performed is indicated with E_{day} and represents the very output of the regressive CART described in section 6.2. Finally, the remaining inputs are day type, month and time of the hourly load to be estimated. These three variables are passed to the network after a B-splines transformation with 4 degrees of freedom is performed on them. A spline of order n , is a function in x made of polynomials and defined piecewise with meeting points t known as knots. If the spline is of order n , then it will be a function of degree $n-1$. Equation 6.3 reports the condition which must be respected by the b-spline of order n for a set of knots t in $i, \dots, i+n$, while a more complete mathematical explanation can be found in (66).

$$B_{i,n}(x) \begin{cases} = 0 & \text{if } x \leq t_i \quad \text{or} \quad t \geq t_{i+n} \\ \neq 0 & \text{otherwise} \end{cases} \quad (6.3)$$

The number of hidden nodes can be selected according to different techniques, in this case the random search has been applied. This technique consists in selecting a possible interval in which the number of nodes of each layer can vary, then an iterative process is started and each time a different random combination of the number of hidden neurons is sampled. According to Bergstra and Bengio (67) this technique is more effective than simple manual tuning or even grid search, both from an empirical and a theoretical point of view.

The results of this random search are then evaluated in terms of a chosen metric on the testing dataset. This is done to avoid the possibility of overfitting when evaluating the results on the training test only. The network is trained, as already discussed, only with fault-free data of the years 2016 and 2017, while as testing period the months from January to July 2018 are considered. In Figure 6.4 the results of a first random search run are represented using testing MAPE, whose definition was given in Equation 6.1, and mean absolute error (MAE), whose definition is given by Equation 6.4, where once again n is the number of observations, x is the actual value and \tilde{x} is the predicted one.

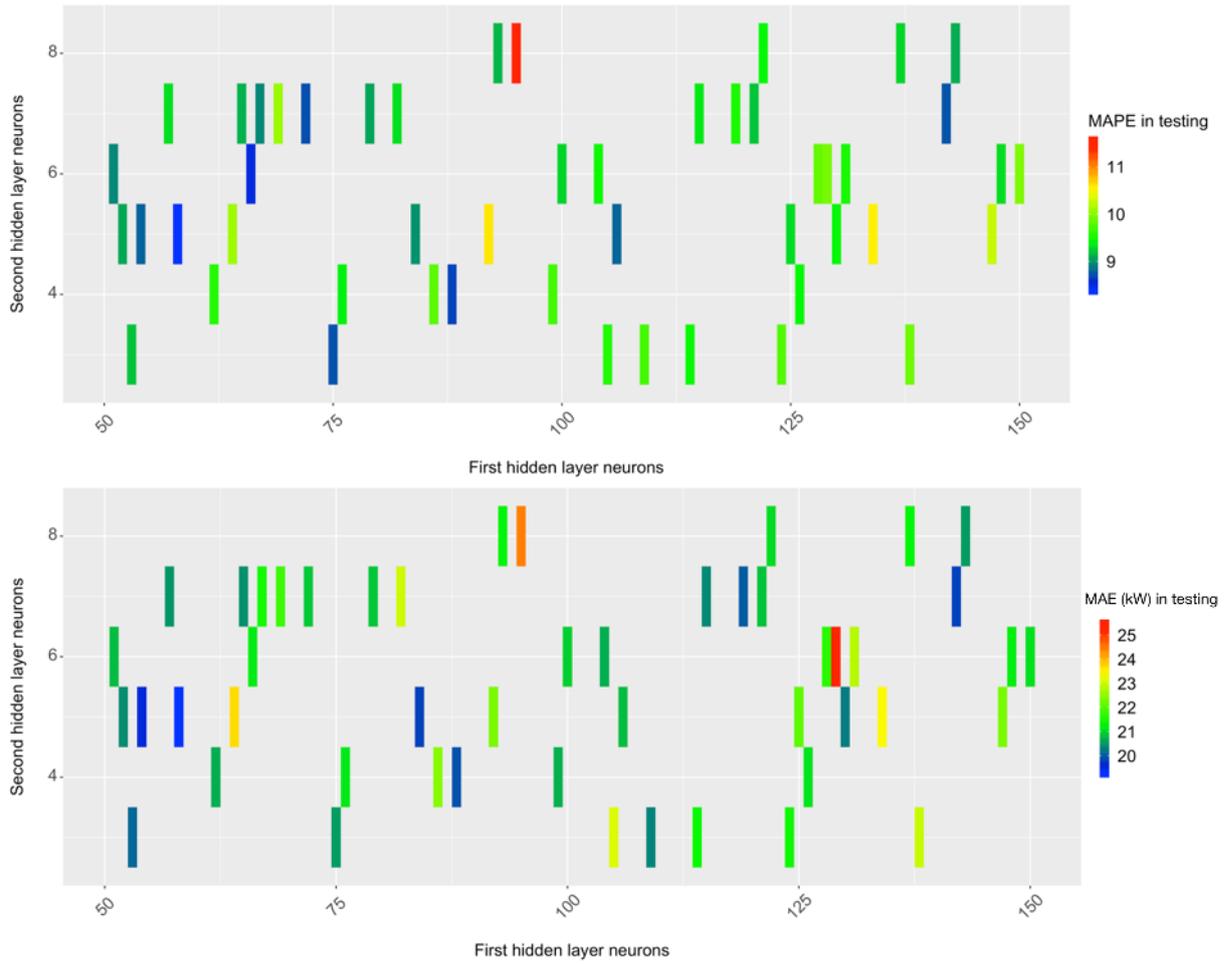


Figure 6.4: First random search results.

In the case of Figure 6.4, the limits for the number of hidden neurons in the first and second hidden layers was set respectively to the intervals [50,150] and [3,8]. Thanks to the color scale, one can easily see that the best results both in terms of testing MAPE and MAE are located in the region which has between 50 and 90 neurons in the first hidden layer and between 3 and 7 in the second. Not to increase the computational load, instead of performing a denser search in the initial region, it could be useful to restrict the area to a smaller interval.

$$MAE = \frac{1}{n} \sum_{k=1}^n |x_k - \widetilde{x}_k| \quad (6.4)$$

In Figure 6.5, the chosen region is in the interval [45,65] for the number of neurons in the first hidden layer, while for the neurons in the second layer it is in [4,6]. Also this time the results are shown both in terms of testing MAPE and MAE and the optimum is found at a number of hidden

neurons of 54 for the first hidden layer and 6 for the second, with a resulting MAPE of 8.44% and MAE of 19.31 kW on the testing data.

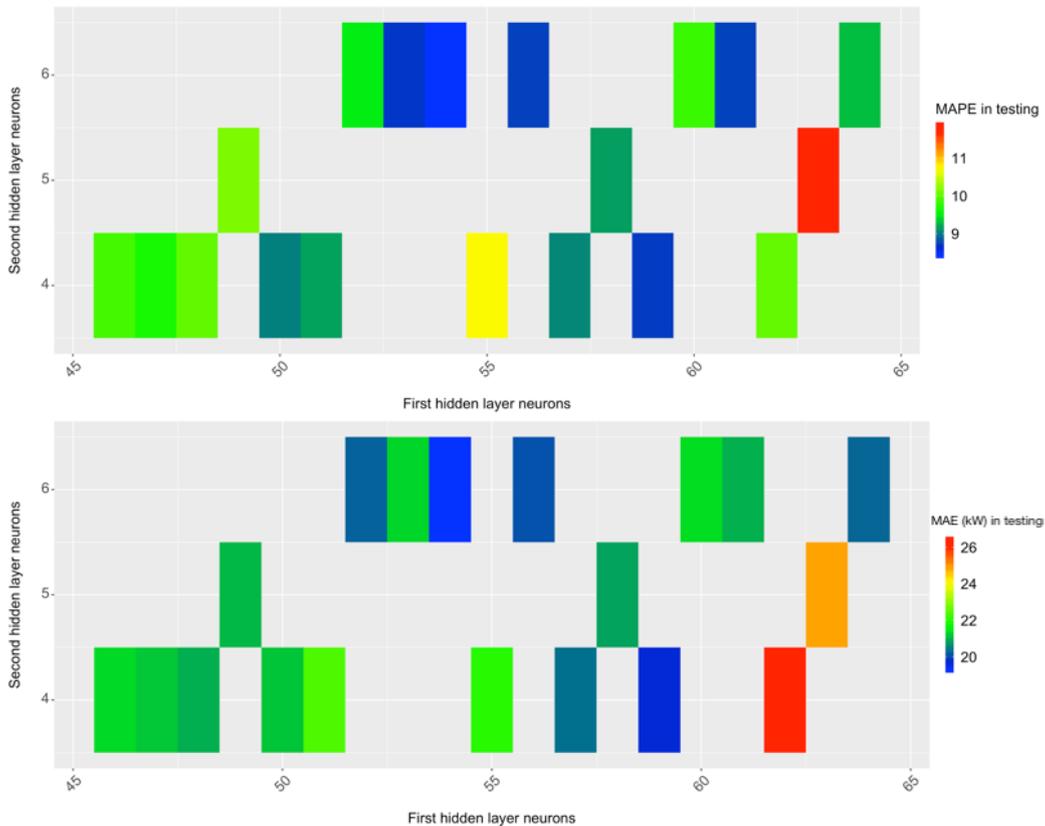


Figure 6.5: Second random search results.

It can be useful to notice that such results are valid for any hour of the day, since there is no information about historical power consumption, so assuming that the outdoor air temperature at a certain hour is well known, either from a weather forecast service or because it has been measured, the mean accuracy of the model will be the same for any hour of the day.

6.4 Model results

As pointed out in the previous paragraphs, the main objective of this framework is not to provide a good power load forecast, but to serve as a benchmark which is able to highlight the rare events which may occur during a certain day. This model can be either used in advance to predict the expected normal behaviour of the system, or at a certain point of the day in order to be compared with the real consumption data. In general, there are two possibilities: the model estimation and the real consumption closely match, or the real consumption has deviations from the expected one. In the first case the meaning of the result is that the system is behaving as expected by the model, which has been trained on non-anomalous data. In the second case there are many possibilities:

the deviation can be caused by a large estimation error of the model which was not able to interpret the system behaviour in the right way; the system is actually passing through a fault, a rare event, or an anomaly; there has been some acquisition problem resulting in a strange measured power load profile.

Since the results of the proposed model are quite good in terms of MAPE and MAE both in testing and training, it can be assumed that, if there is a great discrepancy between expected and real consumption, something rare should be happening. Let's consider now the anomalies and rare events identified in section 5. Since, these data have not been used to train the model they were used as an anomalous testing dataset . For instance, let's look at Figure 6.6, in which the event "CF2 flat" is represented.

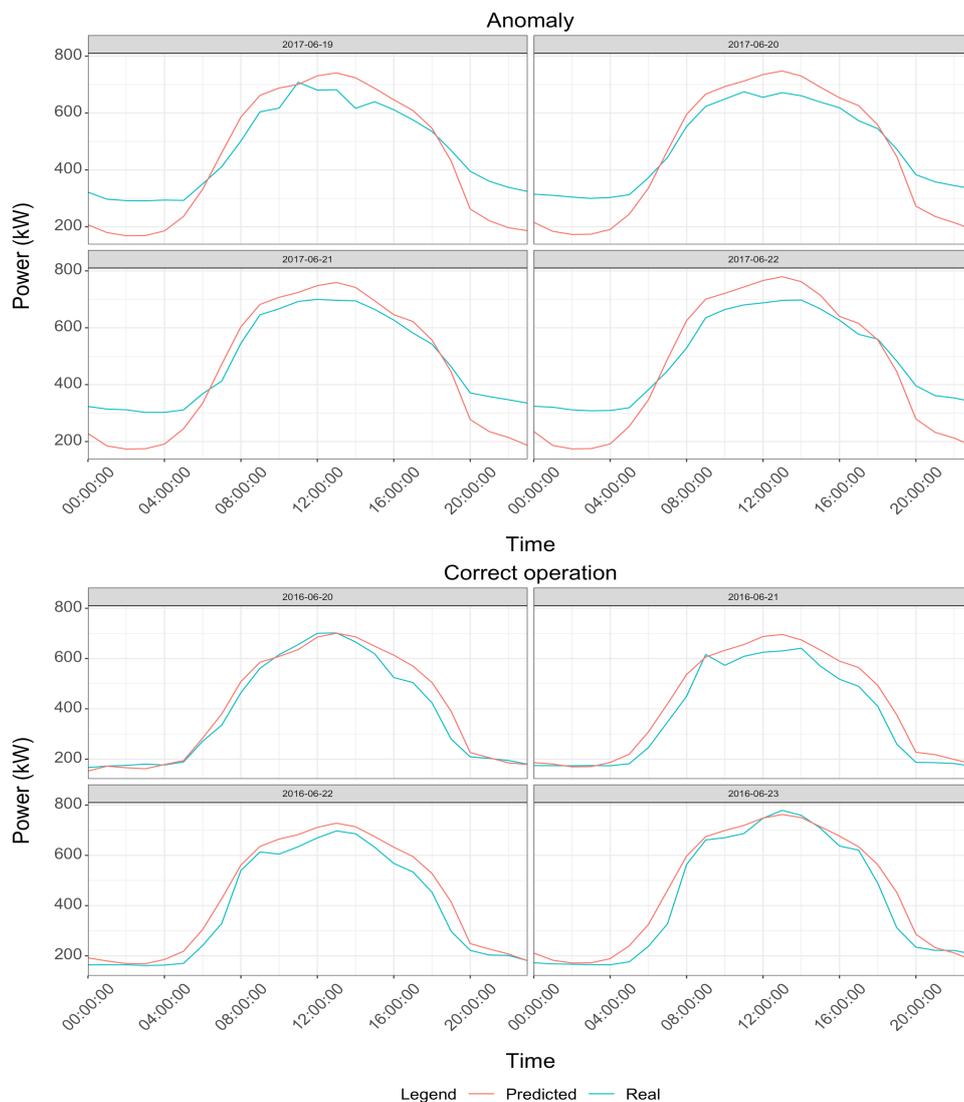


Figure 6.6: Comparison of expected and real consumption during the event "CF2 flat" (above) and during normal operation (below).

In the lower graphs four days of normal operation are represented and, as one can see, there is a very good accordance between the real and the expected consumption. On the contrary, if we look at the upper four profiles, we notice that during night hours the model expects the power load to decrease around 200 kW, while the measured data say that it did not go under 300 kW. As we know thanks to the analysis carried out previously, such behaviour is due to the refrigeration unit n°2 which stayed on during all day with a nearly constant consumption. The model cannot predict the strange and sudden change of behaviour of the refrigeration unit, thus it works exactly as we wanted, highlighting the difference between expected and measured consumption. Some analogous consideration can be made by looking at Figure 6.7 in which the event “Everything flat” is represented.

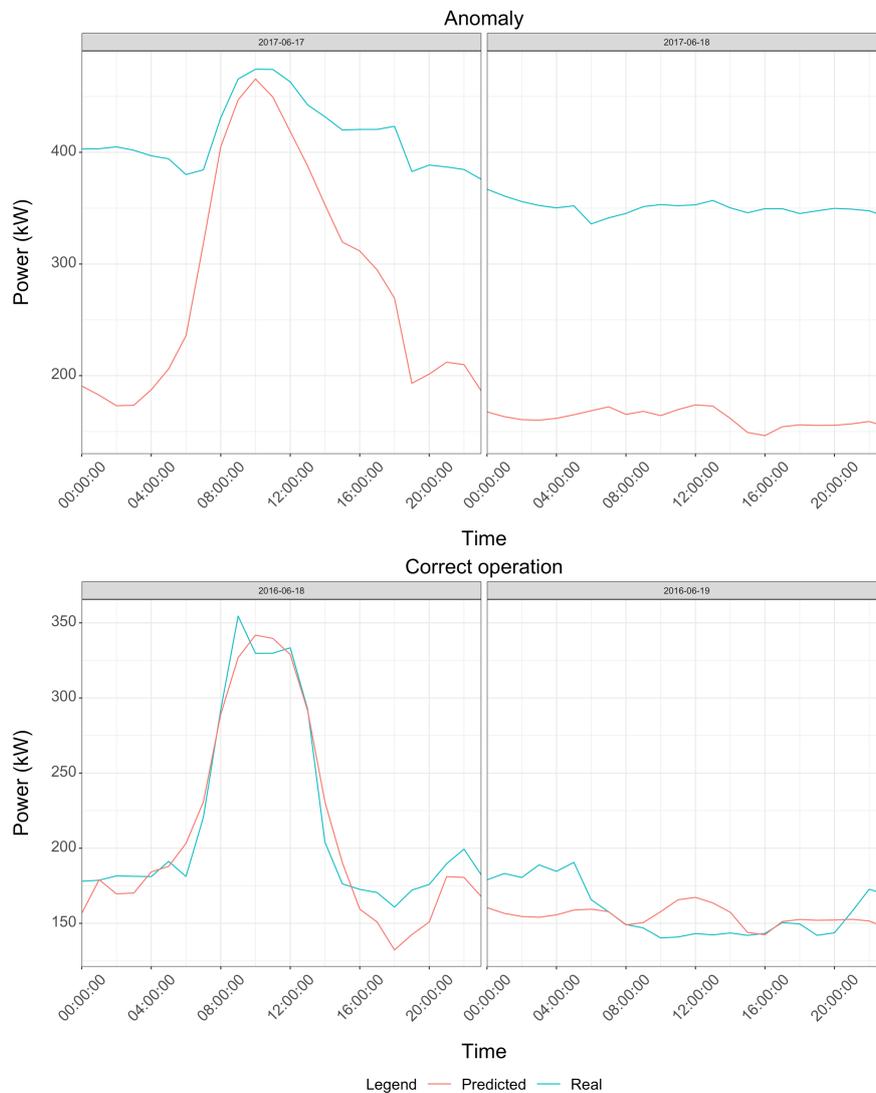


Figure 6.7: Comparison of expected and real consumption during the event "Everything flat" (above) and during normal operation (below).

As before if we look at the lower figures we can see that there is a good accordance between the two profiles with absolute errors rarely exceeding 25 kW, while if we look at the plots on the top, we see that there are gaps of 200 kW which last for several hours in succession. In this case the rarity of the event is even more evident than in the previous case and the model does not shrink from underlining that. Finally, one could be interested in looking at the performance of the model in case of the event “CF2 late off”, which is represented in Figure 6.8.

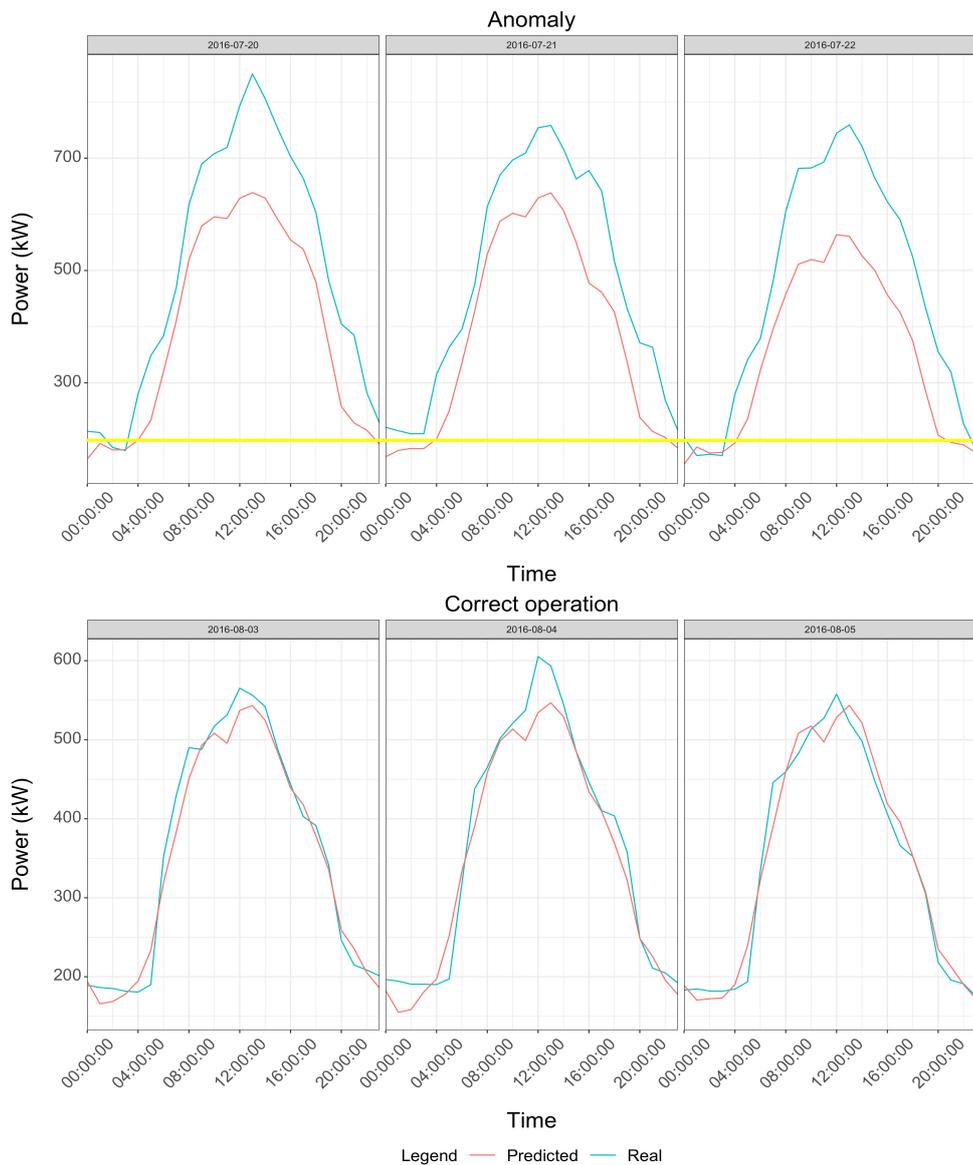


Figure 6.8: Comparison of expected and real consumption during the event "Everything flat" (above) and during normal operation (below).

This time the deviation between expected and real consumption is less evident, since the type of anomaly is less severe than in the previous cases. Still the model success in correctly simulating the load profile during days which are similar to the faulty ones but with normal operation. If we look at the figures on top we notice, at first, that there is a peak consumption which is definitely higher than the expected one, but if we concentrate only on the event of interest, we notice that while in the top left plot the expected consumption reaches 200 kW (yellow line) at approximately 10:30 p.m. to rise again at 04:00 a.m., the real profile does not even manage to reach that value. Similar considerations can be done on the other days in which the rare event verifies, noticing that there is always a time difference of about 2 hours from the moment in which the expected load reaches 200 kW and the one in which the real consumption does the same.

6.5 *Anomalies and rare events detection and labelling*

As the final step of this framework we want our model to detect the presence of anomalies or rare events and possibly to tag them with a label taken from the library obtained in section 5. Many efforts have been done in selecting the right approach for detecting anomalies, for instance Mavromatidis, Acha and Shah (68) stated that if 5 consecutive hours were outside their confidence bound then they could be tagged as faults with high probability of being right. In their case the percentage difference between actual and predicted consumption was taken as the confidence measure. As already discussed, Du et al. (41) used as metric the relative error combining the results of two different neural networks, setting the threshold so to have a good compromise between detections and false alarms. As another possibility, Chou and Telaga (69) stated that an anomaly can be detected if the actual consumption falls outside two times the standard deviation from the predicted value. The formula of the standard deviation is reported in Equation 6.5, where n is the number of samples, x_i is an individual value and \bar{x} is the mean value.

$$\sigma = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2} \quad (6.5)$$

Starting from the study of Chou and Telaga (69), we try to improve it by calculating the standard deviation in a more sophisticate way. If we think about the clusters identified in section 5.1, we certainly notice that while the holidays present a relatively low standard deviation during all day, working days instead present a significant variation of profile intensity passing from night to day. Under this perspective, to use a unique value of standard deviation in order to distinguish from normal and rare events seems reductive. Another interesting possibility is to use an hourly standard

deviation calculated on the hourly consumption of each day, in such a way that its value at 01:00 a.m. can be computed as the standard deviation of all the consumption which occurred at that time and so on. Furthermore, to develop a model which is more sensitive to anomalies also during holidays it is necessary to calculate separately the standard deviation for working days and non-working days.

As proposed by Du et al. (41), the confidence interval tuning was carried out in a way that tries to maximise the sensitivity of the anomaly detector while not increasing too much the number of false alarms. The performance evaluation of the model was executed in a way that simulates real operation, in fact, firstly 24 hours of data corresponding to a day were acquired, then the model was applied in order to calculate the expected consumption, then the residuals between the actual and the expected value were computed as in Equation 6.6 and they were compared with the aforementioned standard deviation, calculated only on the training data, corresponding to the fault-free data of years 2016-2017.

$$r_i = |x_i - \tilde{x}_i| \quad (6.6)$$

Not to decrease the number of faults that were detected, the number of residuals outside the confidence range which allowed to tag a profile as non-usual was set to 1. The confidence interval was set to two times the hourly standard deviation since it was a good compromise between the detected anomalies and the false alarms. Two testing datasets were used. The first, which will be called “rare-testing dataset”, contained all those data which were tagged as rare events in section 5, while the second, which will be named “fault-free-testing dataset”, was taken as the month of February 2018, after a careful visual profile analysis was carried out against the background of the knowledge of years 2016 and 2017.

In Figure 6.9 the red line, represents two times the value of the hourly standard deviation. This line is always higher than the light blue line, which represents the residual between actual and expected hourly consumption. In this case the model does not detect any rare event and does not alert the user.

On the contrary, in Figure 6.10, the opposite case is depicted. The residuals exceed the limit value several times and, in this case, the model will detect the anomaly. It is also interesting to notice that thanks to the aforementioned standard deviation calculation, the latter varies a lot from night time to day time and this allows to detect also those faults which occur only during a limited period of time, either night or day.

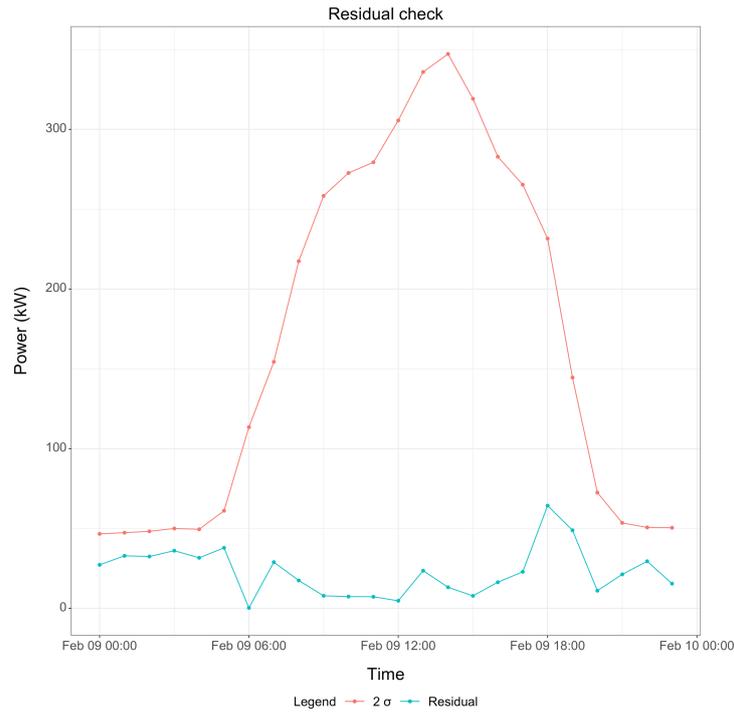


Figure 6.9: Anomaly detector functioning during normal operation.

It is worthwhile to notice that the model has no anomaly in input so all the anomalies and rare events that it detects are new to it, since it is analysing them for the first time.

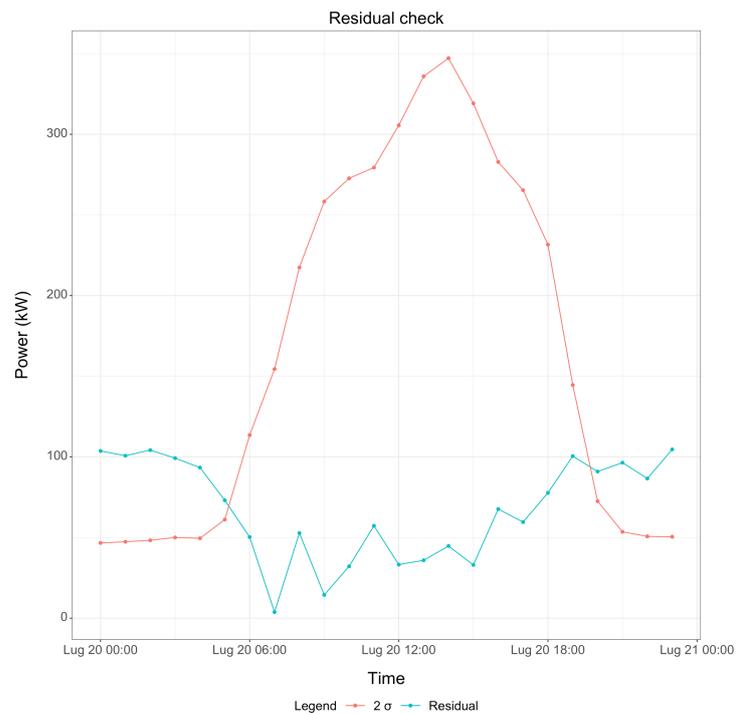


Figure 6.10: Anomaly detector functioning during a rare event.

As a final step the model is also capable to tag all the known events with the right label, chosen from the library proposed in section 5. To achieve this goal when a rare event is detected, according to the aforementioned procedure, it is then compared with the known events of the library. Firstly, the hourly absolute difference between real consumption and the centroid of each type of known rare event is taken, then its mean value is calculated on the 24 hours and the smallest one is chosen. In this way the anomalous profile can be assigned to a determined class from the library. The process is summarised in Equation 6.7, where $x(t)$ is the actual consumption at hour t , $x_i(t)$ is the consumption at hour t of the i -th rare event and n_{RE} is the number of the known rare events.

$$\min \left(\frac{1}{24} \sum_{t=1}^{24} |x(t) - x_i(t)| \right), \quad i = 1, \dots, n_{RE} \quad (6.7)$$

In Table 6.1 the results of the detection tool in testing are shown. The model was able to detect the 97% of the anomalies and rare events while it managed in labelling correctly the 83% of the detected rare events. Finally, only the 3.6% of the fault-free days gave a false alarm, resulting in only 1 day of error out of 28.

Table 6.1: Rare events detection and labelling results.

Dataset	Number of days	Percentage	Task
Rare-testing dataset	55	97.0%	Detected anomalies and rare events
Rare-testing dataset	36	83.3%	Detected rare events which have been labelled correctly
Fault-free testing dataset	28	3.6%	False alarm

As a final consideration it should be said that the model was able to highlight the majority of the rare events just at the end of the day in which they occurred. Since from the analysis carried out in section 5 it can be seen that the rare events usually occur during consecutive days, having at disposal such a tool would enable to promptly detect them and act to stop their repetition. In order to make an approximated evaluation of the influence of these events on the energy consumption, it is possible to assume that the expected consumption calculated by the model could have been reached if no rare event occurred. There are 36 days of high consumption which could have been lower, in particular there have been a difference of around 54,800 kWh more than expected.

Assuming a price of electrical energy of 0.15 €/kWh such consumption leads to an additional cost of about 8,220 € most of which could have been avoided if an anomaly detection system was in operation. To visualize the difference between expected and actual consumption during rare events one could look at Figure 6.11, where on the x axis a time index is reported to have a continuous visualisation of non-contiguous days.

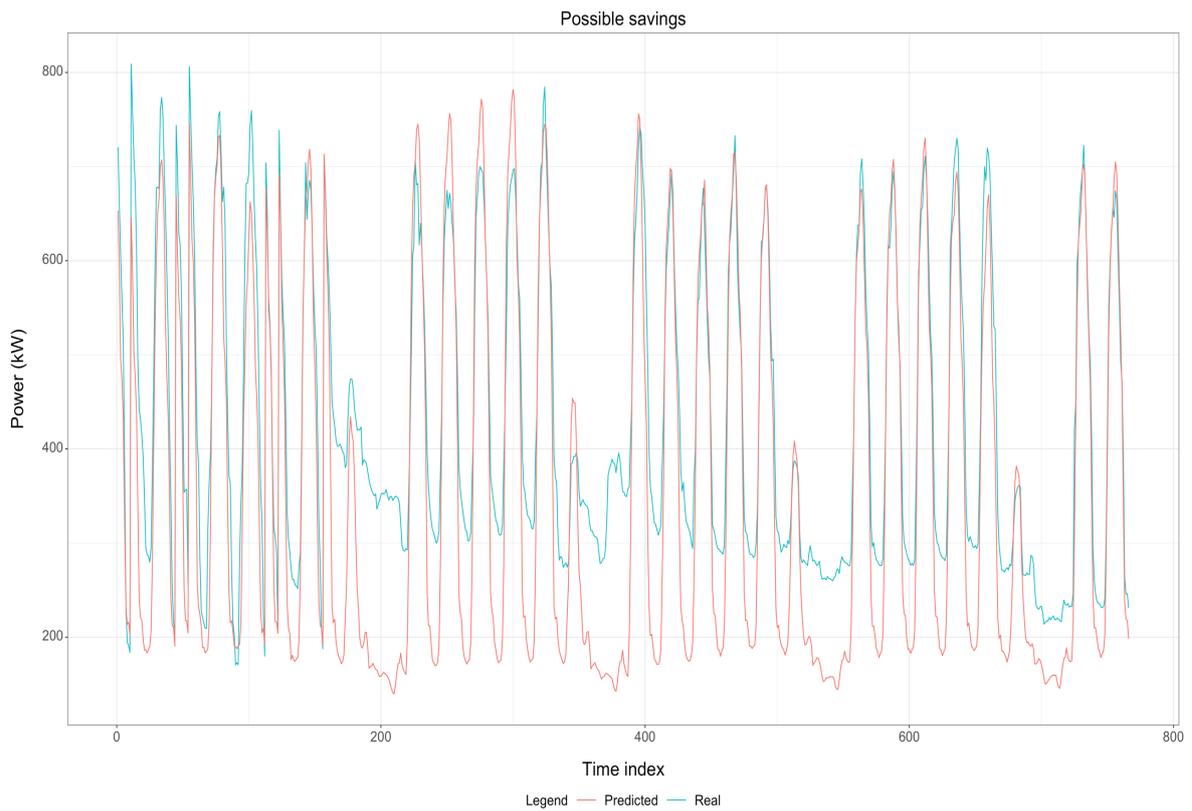


Figure 6.11: Rare events consumption and expected one.

Chapter 7:

Results: forecasting model

7 Forecasting model for the enhancement of building power load management and diagnosis

According to Kelo and Dudul (12), short term load forecasting (STLF) plays a key role in guaranteeing the right security and savings in the power systems. Furthermore Chae et al. (36) state that such predictions can be very useful both for suppliers and for consumers. In fact, the latter can use STLF for improving their power use management. According to Ruiz et al. (13) such methods enable to reach interesting cost savings and energy use improvements without compromising indoor thermal comfort. As stated by Dhillon et al. (70) STLF can be used to supervise future energy consumption and to decide which actions could be performed in order to reduce it. Capozzoli, Grassi and Causone (71) claim that the rapid and consistent prediction of the consumption of buildings is gaining more and more interest, particularly in the context of authorities and organisations which possess and administer great sized constructions. Since many efforts have been done to provide good short-term load forecasting frameworks and the interest in this field keeps increasing, this work provides a tool which forecasts the hourly total power consumption of the electrical substation C of Politecnico di Torino for a time interval of 24 hours. Furthermore, this model offers a methodology which is aimed at quantifying the consumption due to each service provided by the substation.

7.1 *Electrical consumption forecasting*

The electrical consumption forecasting is performed by means of a set of autoregressive neural networks. In Figure 7.1 the general framework of the model is represented. As it can be seen there are 24 independent artificial neural networks each one predicting the power load at a specific time, so that ANN1 forecasts the one hour ahead load, ANN2 the one at two hours ahead and so on until ANN24 which is aimed at predicting the one day ahead electrical load. Moreover it can be noticed that there are 24 different input sets and that each one is used with the corresponding model.

This architecture is called multi-model forecasting and is aimed at avoiding the error propagation which would derive from the choice of an iterative forecasting. The latter implies that the output of the model which provides the one hour ahead power forecast is then used to predict the two hours ahead load and so on. Clearly such framework has the drawback that the prediction error is propagated through each prediction step and so the process may diverge. On the other hand, multi-

model forecasting is a good solution which is able to avoid this issue still preventing overfitting, thanks to the relatively small size of the networks. The model was trained using the fault-free dataset of years 2016-2017 obtained in section 5.

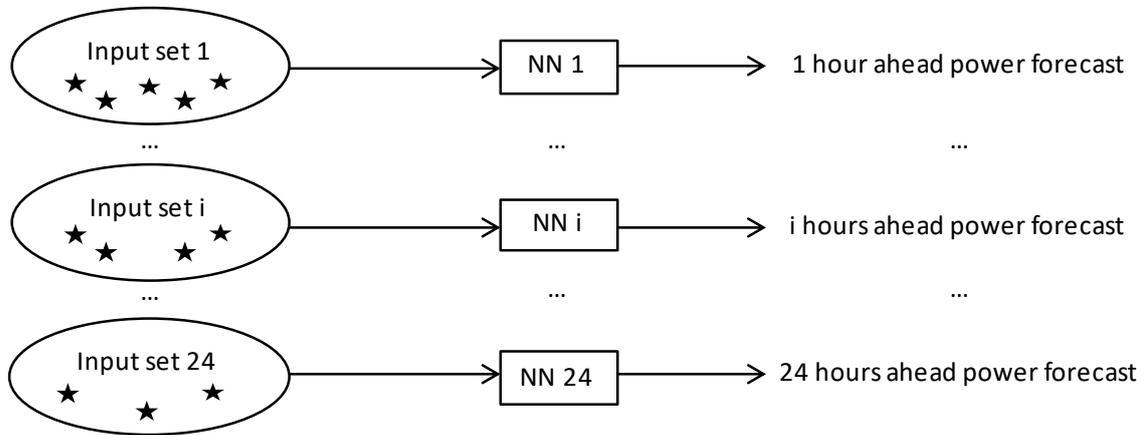


Figure 7.1: Forecasting model general framework.

In Table 7.1 a complete overview of the input sets for each model is reported. The used input variables are timestamp, holiday, power and temperature. The term timestamp refers to the information about month, time and day type. Holiday is a dummy variable which is set equal to one for holidays and to zero otherwise. P represents the power load in kW, while T is the outdoor dry bulb air temperature. The time at which the prediction is carried out is indicated with t , while k is an integer indicating the hours ahead at which the forecast is performed. The model inputs have been selected by looking at the ACF of the power load and trying to minimize the resulting MAPE evaluated on the testing dataset, which was taken as the months of 2018 which goes from January to July.

Table 7.1: Model inputs for each forecasting horizon. Timestamp includes month, time and day type.

Inputs	Forecasting horizon
Timestamp($t+k$), Holiday($t+k$), $P[t-(24-k)]$, $P[t-(23-k)]$, $P(t)$, $T(t)$	$t+k$ with $k=1, \dots, 7$
Timestamp($t+k$), Holiday($t+k$), $P[t-(24-k)]$, $P[t-(23-k)]$	$t+k$ with $k=8, \dots, 23$
Timestamp($t+k$), Holiday($t+k$), $P(t)$, $P(t-1)$, $T(t)$	$t+k$ with $k=24$

The autocorrelation function (ACF), is the linear correlation of a signal with itself at a delayed point in time. The partial autocorrelation function (PACF), instead, is the autocorrelation which exists between two signals which does not take into consideration the points between the two (22).

In Figure 7.2 the ACF of the power load is represented. As it can be seen there is a large positive autocorrelation with the power at time $t-1$ and time $t-2$. Furthermore, it can be noticed that some highly autocorrelated values of power are located at 23 and 24 hours before the current time. Under this perspective and thanks to a trial and error procedure the inputs for each one of the 24 models have been set.

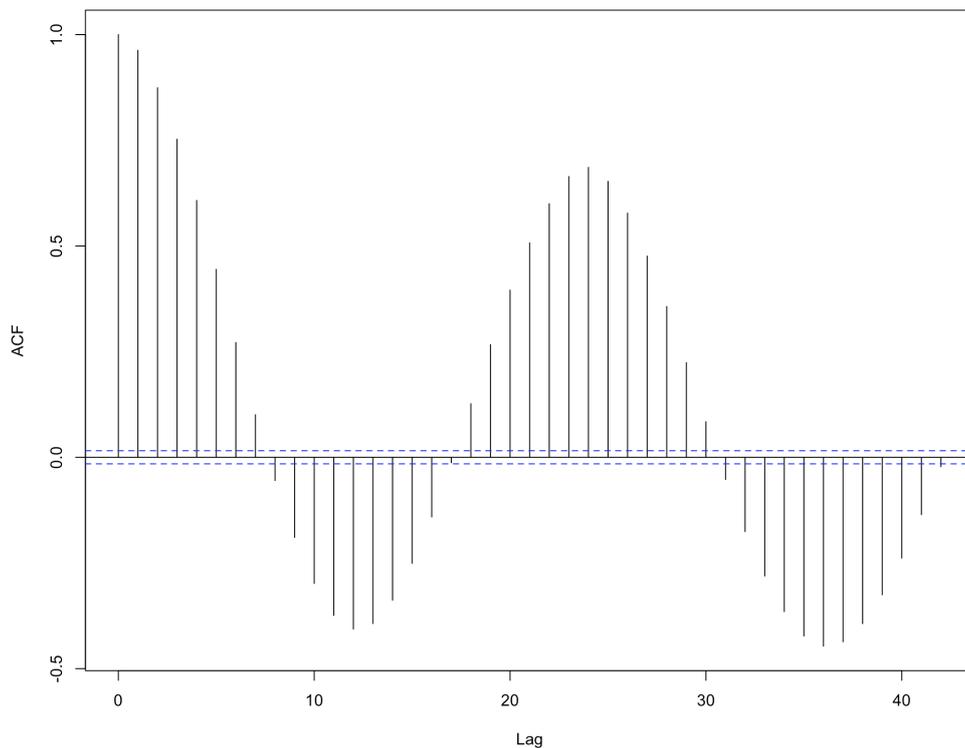


Figure 7.2: Autocorrelation of the power load.

Deciding to include in the model some information about the past 24 hours may lead to a lower prediction accuracy. For instance, if it is 00:00 a.m. of Monday and the model tries to predict the hourly load profile for the next 24 hours, then it looks at the past power values which belonged to a Sunday and it tries to predict the future ones with very poor results. In fact, the consumption of these days is usually very different both in terms of shape and magnitude. In order to overcome this issue two different models, each one made of 24 independent ANNs, are set up: the first is trained only with working days and the second only with weekends and holidays. Consistently each model will be used on the proper testing dataset, in other words the working days model will predict

the power load for working days, while the holiday model will do the same for holidays. As a clarifying example, in Figure 7.3 the ACF of the power load of the only working days is shown. In contrast with Figure 7.2, it can be said that this time the autocorrelations at 24 and 23 hours before the current time are comparable with the one at one hour. This indicates that such values could be effectively used to forecast the power load, as already described in Table 7.1.

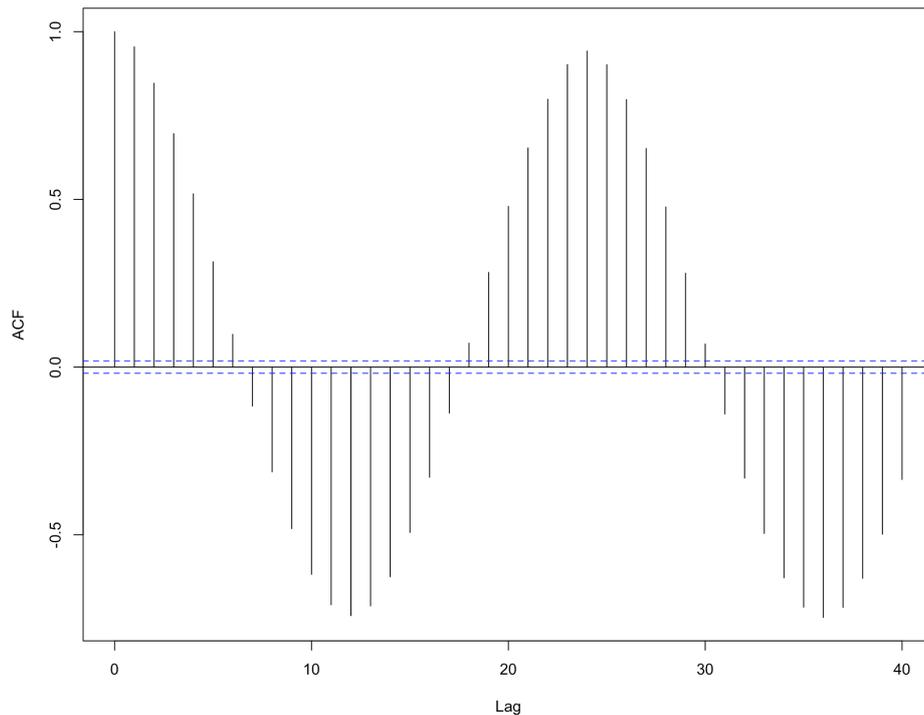


Figure 7.3: Autocorrelation function of the power load of working days.

All the ANNs employed in each one of the two models are feedforward deep neural networks with two hidden layers. The architecture of each neural network, except for the input layer, is the same: in fact, there are 12 neurons in the first hidden layer and 8 in the second. The chosen training algorithm is resilient back-propagation with weight backtracking while all the hidden neurons use the logistic activation function and the output neuron employs the linear one.

As depicted in Figure 7.4 the power load forecast MAPE in testing never reaches 14% with a value of one hour ahead forecast testing MAPE of 5.9 % for weekdays and 4.8 % for holidays. It is interesting to notice that, thanks to the analysis carried out on the power ACF, the MAPE from 5 to 24 hours ahead in working days remains between 8.6 % and 10.7 % demonstrating the reliability of the model. The MAPE of the holidays and weekends model, instead, slightly increases in that forecasting horizon. This can be due to the diversity of profiles which occurs between

Saturdays, Sundays and generic holidays, as highlighted by the analysis of section 5. Further studies could be addressed at reducing this MAPE, maybe by creating separate models for weekends and holidays, able to manage these differences.

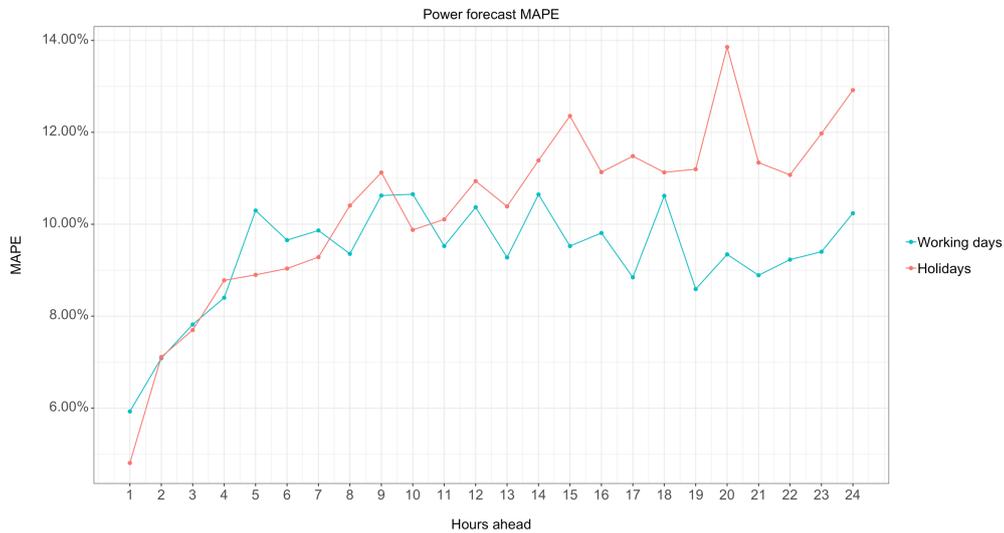


Figure 7.4: Power load forecast MAPE in testing for working days and holidays.

As stated by Fan et al. (63), the performance of a model can be somewhat estimated by looking at the distribution of the residuals which should be normal or almost. In Figure 7.5 the residual distribution is reported for each forecasting horizon in the case of the working days networks.

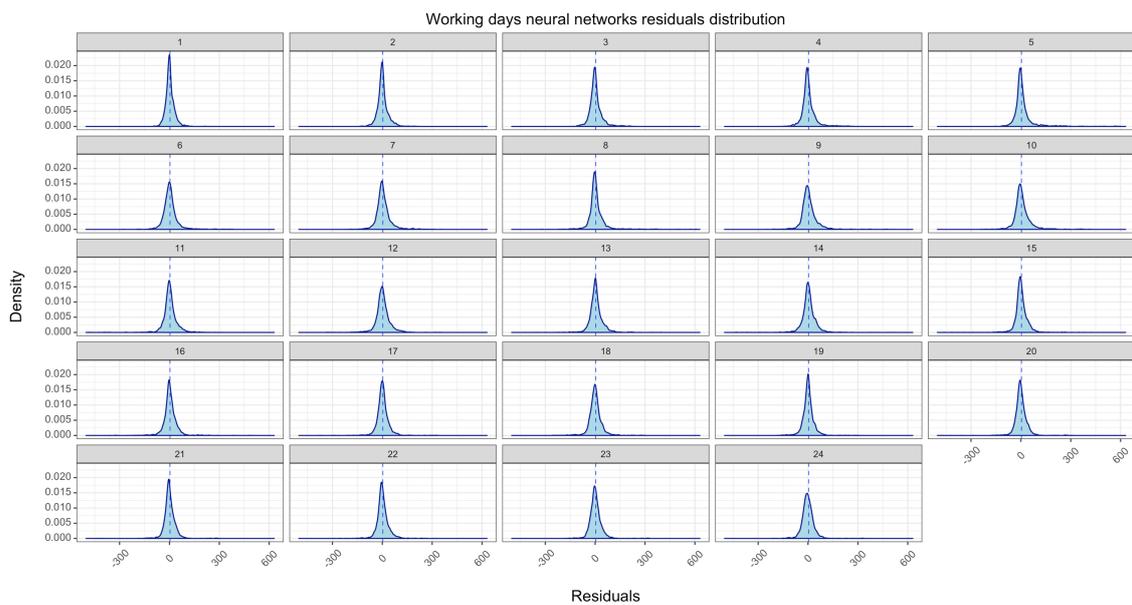


Figure 7.5: Residual distribution of the working days neural networks for each forecasting horizon.

As it can be noticed, even if the shape is not exactly the same for each hour, they are all almost normally distributed. Another common way to assess the validity of a prediction model is to look at the ACF of its residuals evaluated on the training dataset. In Figure 7.6 the residuals of each network are reported for the corresponding forecasting horizon. We can notice that the ACF is almost never above the significance range, indicated by the blue dotted lines, so that each residual appears to be correlated only with itself. Thanks to the analysis carried out both on the residual distribution and ACF and on the testing MAPE it is possible to conclude that the model is effectively able to forecast the power consumption on the desired horizon.

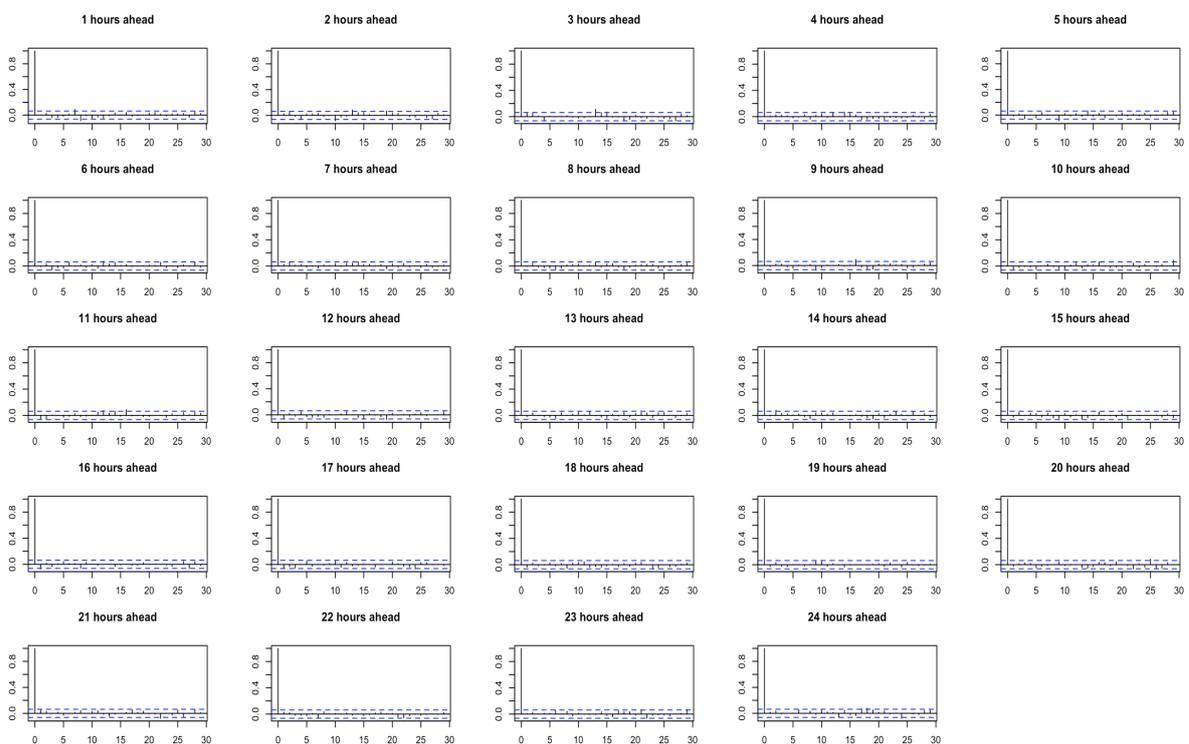


Figure 7.6: Residual ACF of the working days neural networks for each forecasting horizon.

7.2 Consumption at final service level

In order to make this tool even more useful, it is worthwhile to look at the partial power consumption, since being aware of the load fragmentation can effectively improve the energy management, allowing to act on the right final service. The clustering analysis performed on the historical data and the identification of a fault-free dataset can be very useful to overcome this issue. In fact, by categorizing the forecasted power load in one of the final clusters of Figure 5.10, it is possible to discover the percentage due to each single final service for each hour of the cluster.

In Figure 7.7 an example of the differences that can exist between the service intensity of each cluster is shown. In particular clusters 1_1, 3 and 5_3 have been depicted.

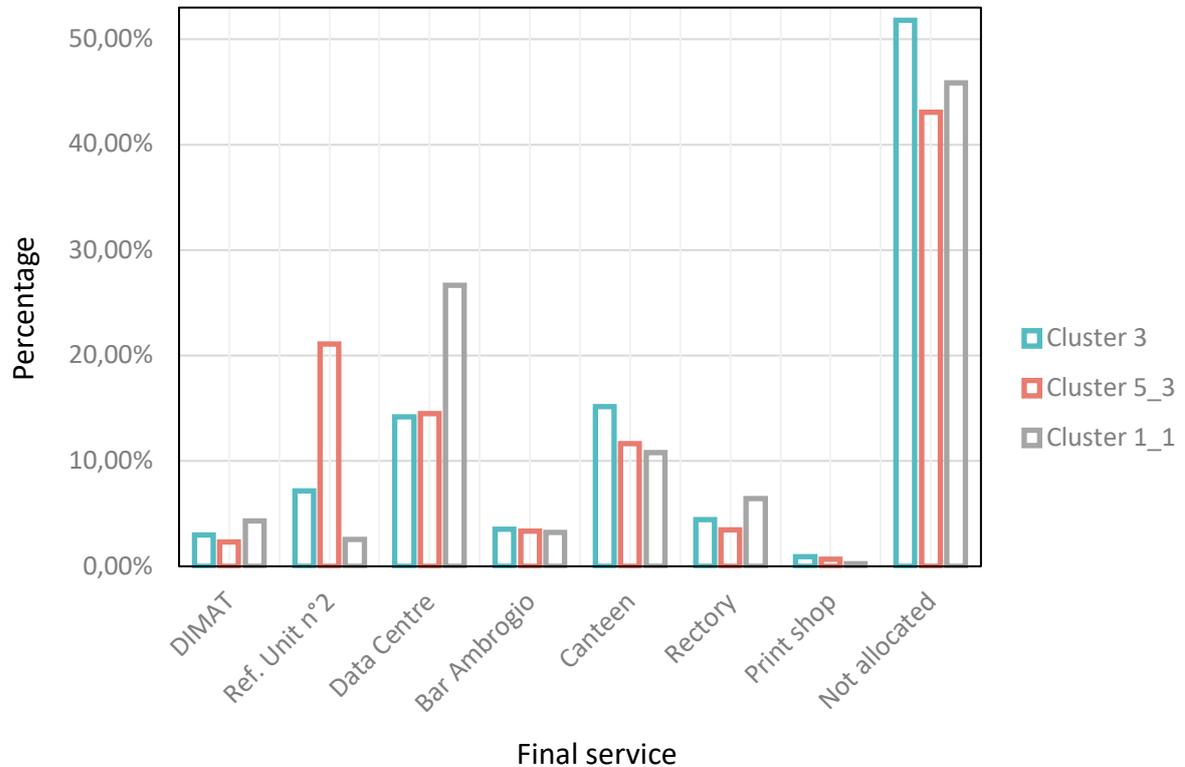


Figure 7.7: Final service mean percentage comparison for clusters 1_1, 3 and 5_3.

The percentages have been calculated as a mean on the whole day of the cluster centroid. As it can be noticed, the refrigeration unit n°2 accounts for less than 8% both for clusters 3 and 1_1, while for cluster 5_3 it represents almost the 28% of the total consumption. This can be easily explained by thinking about the clusters temporal distribution. In fact, cluster 3 is typical of working days belonging to the heating season, while cluster 1_1 is the one of Sundays. In both cases the percentage consumption due to the refrigeration unit is lower than during the very hot days corresponding to cluster 5_3. Similarly, it can be noticed that the part of consumption due to the data centre is higher in cluster 1_1, since during Sundays all the other services require a minimum amount of energy. Finally, a quite significant difference of about 5 percentage points can be noticed in the not allocated power, which is higher for cluster 3.

In order to apply these considerations to each hourly power load which the model predicts, the total 24 hours load shape is compared with each one of the eight centroids, corresponding to the

fault-free clusters. Then the most similar shape, in terms of mean distance, is chosen as the reference cluster for that prediction and the fractions of the cluster, for each energy service and for each hour, are applied to the hourly forecasted energy consumption. To avoid that the forecasts performed at the middle of the day took into consideration the values for the next day, in calculating the most similar cluster, the process is performed independently for each day, so that the predictions for the next day are compared separately to the corresponding hours of each cluster. An example of the resulting power load forecast is shown in Figure 7.8, which simulates the on-line output of the model. The black line represents the measured power load of the past 20 hours, which is useful to give an indication of the past real consumption which the model use to predict the future ones. The green line represents the real future power which at the moment of the prediction is not available and that is reported on the graph just to compare it with the forecasted one. The time of the prediction is represented by the shifting point from the black to the blue curve which is the forecasted hourly power load. It can be noticed that there is a very good accordance between real and forecasted values on the entire forecasting horizon.

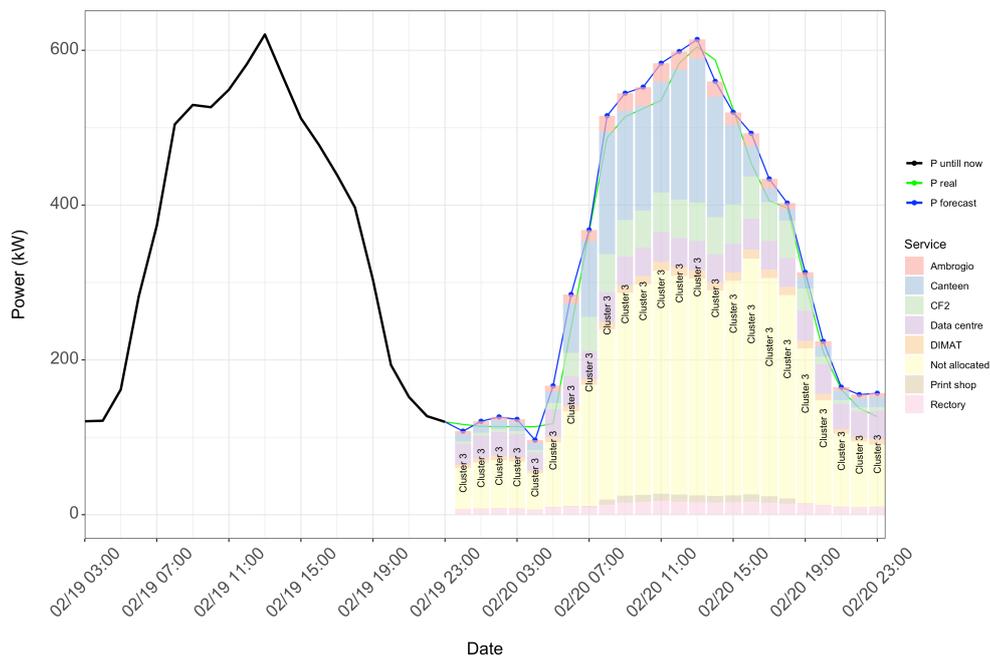


Figure 7.8: 24 hours ahead power load shape forecasting, with final service percentage from clusters for a working day.

The graph shows also the fractions of the hourly forecasted power load for each final energy service, by means of the methodology described above. In particular it can be seen that the consumption related to the canteen grows around the lunch time, the data centre electricity need

is quite constant during the whole day, the consumption of refrigeration unit n°2, rectory, bar Ambrogio, DIMAT, print shop and the not allocated one increase during the operating hours of Politecnico. The predicted cluster is 3, as could have been expected against the background of the cluster temporal analysis of section 5.

In Figure 7.9 the power load forecast for a weekend is reported. Also in this case there is a good accordance between predicted and real consumption.

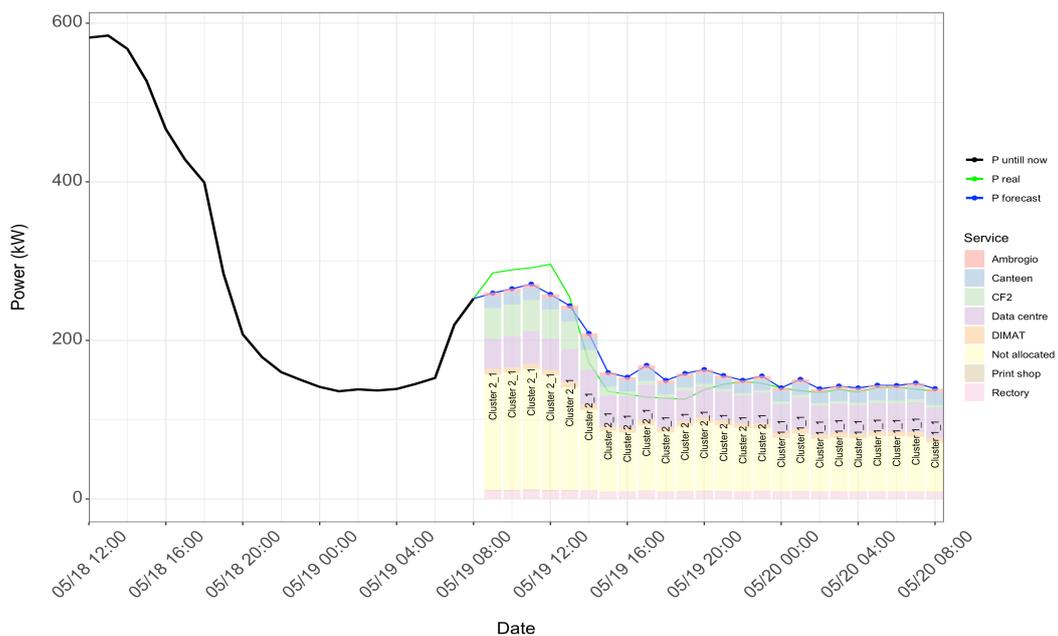


Figure 7.9: 24 hours ahead power load shape forecasting, with final service percentage from clusters for a weekend.

This time both canteen, data centre, rectory, print shop, bar Ambrogio and DIMAT consumption appear to be quite constant, while the consumption of refrigeration unit n°2 and the not allocated one are higher during the central hours of Saturday 05/19/2018. The expected cluster for Saturday is 2_1, while for Sunday is 1_1, showing again a good accordance with the considerations made about Figure 5.11, representing the temporal distribution of the fault-free clusters.

The proposed tool managed in providing a power load forecast on a horizon of 24 hours both for working days and for weekends and holidays. The obtained results were good both in terms of MAPE and in terms of residuals distribution, and the ad-hoc input selection performed by means of the ACF was a key factor to increase the whole model accuracy. The distinction between working and non-working days enabled to further improve the model results, thanks to the better selection of the input variables. The comparison between the reference final service fraction represented by

the clusters of the fault-free centroids made the model able to quantify the distribution of power for each energy service related to the forecasted total power consumption of the electrical substation C of Politecnico di Torino. In particular, at each hour the model looks at the fractions of the corresponding clusters and translates them into fractions of the predicted consumption. This procedure increases the user's awareness about the behaviour of the building and it could be really helpful both for monitoring the partial consumption and for making important management decisions aimed at reducing the demand of a specific service or at shifting it at a more convenient hour of the day, when it is possible.

Chapter 8:
Conclusions

8 Conclusions

The need of limiting the energy consumption is a crucial point involving every country all over the world. The final energy use related to the building sector accounts for a very significant part of the total final energy uses and the need of reducing it is a real matter of concern. Some very promising instruments can be identified in the building automation and control system (BACS) as well as in the technical building management (TBM). The good functioning of these instruments requires a deep knowledge about the building behaviour and the capability of predict future energy consumption, in order to manage them. In this framework, prediction tools are needed to take care of this issue and, among them, a very promising branch, whose importance is growing up, is represented by data-driven black-box models. Such models are often simpler and more precise than forward ones and have the advantage of avoiding the calibration procedure needed by the calibrated simulation tools.

In this work, an exhaustive review of some of the most used data-driven black-box models together with data mining techniques have been carried out, in order to clarify their main fields of application, their potentialities and their shortcomings. Under this perspective five main tasks related to the use of such models have been identified, including forecasting, model predictive control, demand response, fault detection and benchmarking. Forecasting of building energy consumption is both a very useful way to achieve interesting cost savings and energy use improvements without the drawback of compromising indoor thermal comfort, and a starting point for developing the other four applications. Model predictive control (MPC) exploits the aforementioned simulation tools to evaluate the future energy need of the building and then tries to minimize some chosen quantity, such as the total consumption, the cost or both, in order to reach a quasi-optimal solution by actively controlling the technical building system. Demand response consists in acting directly on the demand side, modifying its pattern, and shifting the energy use to the hours in which the grid load is lower. Under this perspective a good forecasting tool can act both as a baseline with which the real modified energy consumption can be compared, and as an effective way to know in advance the power peak hour and its intensity, making more feasible to successfully manage it. A good fault detection system can lead the users to an improved awareness of their building operation, while it is a promising tool both for energy auditing and to carry out some energy efficiency measures, leading to a significant saving in terms of energy and costs. Energy benchmarking, for its part, is a very useful technique which allows the comparison

between different buildings which have some similar characteristics and it can be performed by means of both data-driven and data mining models.

A novel tool aimed at detecting anomalies and rare events and at forecasting the power load of the electrical substation C of Politecnico di Torino has been developed. Firstly, the data have been studied thanks to visual analytics techniques. Then a clustering and CART analysis have been carried out to extract useful information about the most usual load profiles of the building and to identify a library of measurement anomalies and rare events. The fault-free data have been subsequently employed to train two different models. The first model was aimed at detecting rare events and at labelling them and showed a testing accuracy of anomaly detection of 97% and a 3.6% of false alarms. This tool could be used to monitor the functioning of the system identifying automatically anomalous behaviours in a quasi-real time way. Moreover, knowledge about the nature of the occurred fault could be leveraged by the user to avoid its repetition, resulting in a significant energy efficiency improvement. The second model was aimed at forecasting the hourly 24 hours ahead load profile for the electrical substation and resulted in a testing MAPE between 4.8% and 13.9% depending on the day type and on the forecasting horizon. An estimation of the load profile for the next 24 hours is a useful resource to improve building energy management providing the opportunity to allocate electrical loads to more convenient hours in a Demand Side Management perspective. Finally, the developed model was also able to extract the most probable partial consumption of each energy service linked to the electrical substation starting from the historical information gathered on the biennium 2016 – 2017, further increasing its utility and enhancing the energy management, passing from a total consumption level to a local one.

Further studies may focus on the use of other types of algorithms to develop an ensemble model which can improve the forecasting accuracy. Moreover, it would be interesting to add a functionality to the tool in order to recognise new kinds of anomalies or rare events and to add them to the existing library in an on-line functioning and to temporize the training of the models, adapting them to the changes which will happen in the building.

Chapter 9:
Bibliography

9 Bibliography

1. Environment U, IEA. Renewables Global Status Report 2017. 2017.
2. González PA, Zamarreño JM. Prediction of hourly energy consumption in buildings based on a feedback artificial neural network. *Energy Build.* 2005;37(6):595–601.
3. Ippolito MG, Sanseverino ER, Zizzo G. Impact of building automation control systems and technical building management systems on the energy performance class of residential buildings : An Italian case study. 2014;69:33–40.
4. Directive 2010/31/EU of the European Parliament and of the Council of 19 May 2010 on the energy performance of buildings.
5. Ahmad T, Chen H, Guo Y, Wang J. Energy & Buildings A comprehensive overview on the data driven and large scale based approaches for forecasting of building energy demand : A review. *Energy Build* [Internet]. 2018;165:301–20. Available from: <https://doi.org/10.1016/j.enbuild.2018.01.017>
6. Ashrae. Energy Estimating and Modeling Methods. In: ASHRAE, editor. ASHRAE Handbook Fundamentals. Inc.: Atalanta, GA, USA; 2013.
7. Deb C, Zhang F, Yang J, Eang S, Wei K. A review on time series forecasting techniques for building energy consumption. 2017;74(July 2016):902–24.
8. Loh WY. Classification and regression trees. *Wiley Interdiscip Rev Data Min Knowl Discov.* 2011;1(1):14–23.
9. Jolliffe I. T. Principal Component Analysis. In: Springer, editor. *Encyclopedia of Statistics in Behavioral Science*. Second Edi. 2002. p. 487.
10. Eiben AE, Schoenauer M. Evolutionary computing. 2002;82:1–6.
11. Ferlito S, Atrigna M, Graditi G, Vito S De, Salvato M, Buonanno A, et al. Predictive models for building ' s energy consumption : an Artificial Neural Network (ANN) approach . 2015;3–6.
12. Kelo S, Dudul S. Electrical Power and Energy Systems A wavelet Elman neural network for short-term electrical load prediction under the influence of temperature. *Int J Electr Power Energy Syst* [Internet]. 2012;43(1):1063–71. Available from: <http://dx.doi.org/10.1016/j.ijepes.2012.06.009>
13. Ruiz LGB, Rueda R, Cuéllar MP, Pegalajar MC. Energy consumption forecasting based on Elman neural networks with evolutive optimization. *Expert Syst Appl.* 2018;92:380–9.
14. Choueiki MH, Mount-campbell CA, Ahalt SC. Building a quasi-optimal neural network to solve the short-term-load forecasting. 1997;12(4):1432–9.
15. Zhang F, Deb C, Eang S, Yang J, Wei K. Time series forecasting for building energy consumption using weighted Support Vector Regression with differential evolution optimization technique. *Energy Build* [Internet]. 2016;126:94–103. Available from: <http://dx.doi.org/10.1016/j.enbuild.2016.05.028>
16. Amber KP, Ahmad R, Aslam MW, Kousar A, Usman M, Khan MS. Intelligent techniques for forecasting electricity consumption of buildings. 2018;157:886–93.
17. Chen Y, Tan H, Song X. Day-ahead Forecasting of Non-stationary Electric Power Demand in Commercial Buildings : Hybrid Support Vector Regression Based. 2017;105:2101–6.
18. Ahmad MW, Mourshed M, Rezgui Y. Trees vs Neurons : Comparison between random forest and ANN for high-resolution prediction of building energy consumption. *Energy Build* [Internet]. 2017;147:77–89. Available from: <http://dx.doi.org/10.1016/j.enbuild.2017.04.038>
19. Rezaeian-Zadeh M, Zand-Parsa S, Abghari H, Zolghadr M, Singh VP. Hourly air temperature driven using multi-layer perceptron and radial basis function networks in arid

- and semi-arid regions. *Theor Appl Climatol.* 2012;109(3–4):519–28.
20. Yokoyama R, Wakui T, Satake R. Prediction of energy demands using neural network with model identification by global optimization. *Energy Convers Manag* [Internet]. 2009;50(2):319–27. Available from: <http://dx.doi.org/10.1016/j.enconman.2008.09.017>
 21. Yang J, Rivard H, Zmeureanu R. On-line building energy prediction using adaptive artificial neural networks. *Energy Build.* 2005;37(12):1250–9.
 22. Fan C, Xiao F, Wang S. Development of prediction models for next-day building energy consumption and peak power demand using data mining techniques. 2014;127:1–10.
 23. Chen Z, Jiang C. Building Occupancy Modeling Using Generative Adversarial Network. *Energy Build* [Internet]. 2018; Available from: <https://doi.org/10.1016/j.enbuild.2018.06.029>
 24. Guo Y, Wang J, Chen H, Li G, Liu J, Xu C. Machine learning-based thermal response time ahead energy demand prediction for building heating systems ☆. *Appl Energy* [Internet]. 2018;221(April):16–27. Available from: <https://doi.org/10.1016/j.apenergy.2018.03.125>
 25. Ben-Nakhi AE, Mahmoud MA. Cooling load prediction for buildings using general regression neural networks. *Energy Convers Manag.* 2004;45(13–14):2127–41.
 26. Zhao J, Liu X. A hybrid method of dynamic cooling and heating load forecasting for office buildings based on artificial intelligence and regression analysis. *Energy Build.* 2018;174:293–308.
 27. Al-obeidat F, Spencer B, Alfandi O. Consistently accurate forecasts of temperature within buildings from sensor data using ridge and lasso regression. *Futur Gener Comput Syst* [Internet]. 2018; Available from: <https://doi.org/10.1016/j.future.2018.02.035>
 28. Lauro F, Moretti F, Capozzoli A, Panzieri S. Model predictive control for building active demand response systems. *Energy Procedia* [Internet]. 2015;83:494–503. Available from: <http://dx.doi.org/10.1016/j.egypro.2015.12.169>
 29. Pospíšil J, Špiláček M, Kudela L. Potential of predictive control for improvement of seasonal coefficient of performance of air source heat pump in Central European climate zone. *Energy.* 2018;154:415–23.
 30. Reynolds J, Rezgui Y, Kwan A, Piriou S. A zone-level, building energy optimisation combining an artificial neural network, a genetic algorithm, and model predictive control. *Energy.* 2018;151:729–39.
 31. Macarulla M, Casals M, Forcada N, Gangoles M. Implementation of predictive control in a commercial building energy management system using neural networks. *Energy Build* [Internet]. 2017;151:511–9. Available from: <http://dx.doi.org/10.1016/j.enbuild.2017.06.027>
 32. Costanzo GT, Iacovella S, Ruelens F, Leurs T, Claessens BJ. Sustainable Energy , Grids and Networks Experimental analysis of data-driven control for a building heating system. *Sustain Energy, Grids Networks* [Internet]. 2016;6:81–90. Available from: <http://dx.doi.org/10.1016/j.segan.2016.02.002>
 33. Erfani A, Rajabi-ghahnaviyeh A, Boroushaki M. Design and construction of a non-linear model predictive controller for building ' s cooling system. *Build Environ* [Internet]. 2018;133(February):237–45. Available from: <https://doi.org/10.1016/j.buildenv.2018.02.022>
 34. Peng Y, Rysanek A, Nagy Z, Schlüter A. Using machine learning techniques for occupancy-prediction-based cooling control in office buildings. *Appl Energy* [Internet]. 2018;211(November 2017):1343–58. Available from: <https://doi.org/10.1016/j.apenergy.2017.12.002>
 35. Srivastava A, Van Passel S, Laes E. Assessing the success of electricity demand response programs: A meta-analysis. *Energy Res Soc Sci.* 2018;40(January 2017):110–7.
 36. Chae YT, Horesh R, Hwang Y, Lee YM. Artificial neural network model for forecasting sub-hourly electricity usage in commercial buildings. *Energy Build* [Internet]. 2016;111:184–

94. Available from: <http://dx.doi.org/10.1016/j.enbuild.2015.11.045>
37. Chen Y, Xu P, Chu Y, Li W, Wu Y, Ni L, et al. Short-term electrical load forecasting using the Support Vector Regression (SVR) model to calculate the demand response baseline for office buildings. 2017;195:659–70.
 38. Chen Z, Wu L, Cheng S, Lin P, Wu Y, Lin W. Intelligent fault diagnosis of photovoltaic arrays based on optimized kernel extreme learning machine and I-V characteristics q. *Appl Energy* [Internet]. 2017;204:912–31. Available from: <http://dx.doi.org/10.1016/j.apenergy.2017.05.034>
 39. Li D, Zhou Y, Hu G, Spanos CJ. Fault detection and diagnosis for building cooling system with a tree-structured learning method &. *Energy Build* [Internet]. 2016;127:540–51. Available from: <http://dx.doi.org/10.1016/j.enbuild.2016.06.017>
 40. Yan R, Ma Z, Kokogiannakis G, Zhao Y. Automation in Construction A sensor fault detection strategy for air handling units using cluster analysis. *Autom Constr* [Internet]. 2016;70:77–88. Available from: <http://dx.doi.org/10.1016/j.autcon.2016.06.005>
 41. Du Z, Fan B, Jin X, Chi J. Fault detection and diagnosis for buildings and HVAC systems using combined neural networks and subtractive clustering analysis. 2014;73.
 42. Li Z, Han Y, Xu P. Methods for benchmarking building energy consumption against its past or intended performance: An overview. *Appl Energy* [Internet]. 2014;124:325–34. Available from: <http://dx.doi.org/10.1016/j.apenergy.2014.03.020>
 43. Chung W. Review of building energy-use performance benchmarking methodologies. *Appl Energy* [Internet]. 2011;88(5):1470–9. Available from: <http://dx.doi.org/10.1016/j.apenergy.2010.11.022>
 44. Capozzoli A, Savino M, Neri F, Grassi D, Serale G. A novel methodology for energy performance benchmarking of buildings by means of Linear Mixed Effect Model : The case of space and DHW heating of out-patient Healthcare Centres. *Appl Energy* [Internet]. 2016;171:592–607. Available from: <http://dx.doi.org/10.1016/j.apenergy.2016.03.083>
 45. Roth J, Rajagopal R. Benchmarking building energy efficiency using quantile regression. *Energy*. 2018;152:866–76.
 46. Luo X, Hong T, Chen Y, Piette MA. Electric load shape benchmarking for small- and medium-sized commercial buildings. *Appl Energy* [Internet]. 2017;204(June):715–25. Available from: <http://dx.doi.org/10.1016/j.apenergy.2017.07.108>
 47. Papadopoulos S, Bonczak B, Kontokosta CE. Pattern recognition in building energy performance over time using energy benchmarking data. *Appl Energy* [Internet]. 2018;221(February):576–86. Available from: <https://doi.org/10.1016/j.apenergy.2018.03.079>
 48. Fan C, Xiao F, Li Z, Wang J. Unsupervised data analytics in mining big building operational data for energy efficiency enhancement : A review. *Energy Build* [Internet]. 2018;159:296–308. Available from: <http://dx.doi.org/10.1016/j.enbuild.2017.11.008>
 49. Miller C, Nagy Z, Schlueter A. A review of unsupervised statistical learning and visual analytics techniques applied to performance analysis of non-residential buildings. *Renew Sustain Energy Rev* [Internet]. 2018;81(June 2017):1365–77. Available from: <http://dx.doi.org/10.1016/j.rser.2017.05.124>
 50. Jain AK, Murty MN, Flynn PJ. Data clustering: a review. *ACM Comput Surv*. 1999;31(3):264–323.
 51. Saxena A, Prasad M, Gupta A, Bharill N, Patel OP, Tiwari A, et al. A review of clustering techniques and developments. *Neurocomputing*. 2017;267:664–81.
 52. Yu Z, Fung BCM, Haghghat F. Extracting knowledge from building-related data - A data mining framework. *Build Simul*. 2013;6(2):207–22.
 53. Awad M, Khanna R. Support Vector Regression. In: *Efficient Learning Machines*. Apress, Berkeley, CA; 2015.
 54. Jain RK, Smith KM, Culligan PJ, Taylor JE. Forecasting energy consumption of multi-family

- residential buildings using support vector regression : Investigating the impact of temporal and spatial monitoring granularity on performance accuracy. *Appl Energy* [Internet]. 2014;123:168–78. Available from: <http://dx.doi.org/10.1016/j.apenergy.2014.02.057>
55. Deb C, Siew L, Yang J, Santamouris M. Forecasting diurnal cooling energy load for institutional buildings using Artificial Neural Networks. *Energy Build* [Internet]. 2016;121:284–97. Available from: <http://dx.doi.org/10.1016/j.enbuild.2015.12.050>
 56. Kröse B, van der Smagt P. *Introduction to Neural Networks*. 8th ed. University of Amsterdam Press, University of Amsterdam; 1996.
 57. Morgan J. *Classification and Regression Tree Analysis*. 2014;(1).
 58. Hyndman RJ, Athanasopoulos G. *Forecasting: principles and practice*. OTexts, editor. 2018. 380 p.
 59. Potter K. Methods for Presenting Statistical Information: The Box Plot. *Vis Large Unstructured Data Sets*. 2006;4:97–106.
 60. Capozzoli A, Savino M, Brandi S. ScienceDirect ScienceDirect Mining typical load profiles in buildings to support energy management in the smart city context. *Energy Procedia* [Internet]. 2017;134:865–74. Available from: <https://doi.org/10.1016/j.egypro.2017.09.545>
 61. Pao YH, Sobajic DJ. Combined use of unsupervised and supervised learning for dynamic security assessment. *IEEE Trans Power Syst* [Internet]. 1992;7(2):878–84. Available from: [10.1109/59.141799](http://dx.doi.org/10.1109/59.141799)
 62. Chicco G, Napoli R, Piglion F. Comparisons among clustering techniques for electricity customer classification. *IEEE Trans Power Syst*. 2006;21(2):933–40.
 63. Fan C, Xiao F, Zhao Y, Wang J. Analytical investigation of autoencoder-based methods for unsupervised anomaly detection in building energy data. *Appl Energy* [Internet]. 2018;211(September 2017):1123–35. Available from: <https://doi.org/10.1016/j.apenergy.2017.12.005>
 64. Zhang C, Cao L, Romagnoli A. On the feature engineering of building energy data mining. *Sustain Cities Soc* [Internet]. 2018;39(February):508–18. Available from: <https://doi.org/10.1016/j.scs.2018.02.016>
 65. Capozzoli A, Piscitelli MS, Brandi S, Grassi D. Automated load pattern learning and anomaly detection for enhancing energy management in smart buildings. *Energy* [Internet]. 2018;157:336–52. Available from: <https://doi.org/10.1016/j.energy.2018.05.127>
 66. Carlson BC. B-splines, hypergeometric functions, and Dirichlet averages. *J Approx Theory*. 1991;67(3):311–25.
 67. Bergstra J, Bengio Y. Random Search for Hyper-Parameter Optimization. *J Mach Learn Res*. 2012;13:281–305.
 68. Mavromatidis G, Acha S, Shah N. Diagnostic tools of energy performance for supermarkets using Artificial Neural Network algorithms. *Energy Build* [Internet]. 2013;62:304–14. Available from: <http://dx.doi.org/10.1016/j.enbuild.2013.03.020>
 69. Chou J, Telaga AS. Real-time detection of anomalous power consumption. *Renew Sustain Energy Rev* [Internet]. 2014;33:400–11. Available from: <http://dx.doi.org/10.1016/j.rser.2014.01.088>
 70. Dhillon J, Rahman SA, Ahmad SU, Hossain MJ. Peak Electricity Load Forecasting Using Online Support Vector Regression. 2016 IEEE Can Conf Electr Comput Eng. 2016;1–4.
 71. Capozzoli A, Grassi D, Causone F. Estimation models of heating energy consumption in schools for local authorities planning. *Energy Build* [Internet]. 2015;105(January 2006):302–13. Available from: <http://dx.doi.org/10.1016/j.enbuild.2015.07.024>
 72. Brandi S. Analysis of electrical energy consumption of Politecnico di Torino through data analytics methods: the case of the substation C. Tesi magistrale, Politecnico di Torino. 2016

List of figures

Figure 1.1: Global final energy consumption by sector, 2015. taken from (1) with changes.	2
Figure 2.1: Diagram of the Black-box methods reviewed in section 2.	8
Figure 2.2: Diagram of the forecasting methods reviewed in section 2.1.	10
Figure 2.3: Diagram of the MPC methods reviewed in section 2.2.	15
Figure 2.4: Diagram of the demand response methods reviewed in section 2.3.	17
Figure 2.5: Diagram of the fault detection methods reviewed in section 2.4.	19
Figure 2.6: Diagram of the benchmarking methods reviewed in section 2.5.	21
Figure 3.1: Simple clustering example.	25
Figure 3.2: Linear support vector regression in one dimension. Figure taken from (52) with changes.	29
Figure 3.3: Neuron or perceptron.	31
Figure 3.4: Sigmoid activation function.	32
Figure 3.5: Classification tree example.	36
Figure 4.1: Electrical substation C services pie chart for biennium 2016-2017.	42
Figure 4.2: Total power carpet plot.	43
Figure 4.3: Not allocated power carpet plot.	43
Figure 4.4: DIMAT power carpet plot.	44
Figure 4.5: Refrigeration unit n°2 power carpet plot.	44
Figure 4.6: Data centre power carpet plot.	45
Figure 4.7: Bar Ambrogio power carpet plot.	45
Figure 4.8: Canteen power carpet plot.	46
Figure 4.9: Rectory power carpet plot.	46
Figure 4.10: Print shop power carpet plot.	47
Figure 4.11: Basic structure of the box plot. Figure taken from (58) with changes.	48
Figure 4.12: Box plot of the total power load for the biennium 2016-2017.	49
Figure 4.13: Box plot of the not allocated power load for the biennium 2016-2017.	49
Figure 4.14: Box plot of the DIMAT power load for the biennium 2016-2017.	50
Figure 4.15: Box plot of the refrigeration unit n°2 power load for the biennium 2016-2017.	50
Figure 4.16: Box plot of the data centre power load for the biennium 2016-2017.	51
Figure 4.17: Box plot of the bar Ambrogio power load for the biennium 2016-2017.	51
Figure 4.18: Box plot of the canteen power load for the biennium 2016-2017.	52
Figure 4.19: Box plot of the rectory power load for the biennium 2016-2017.	52
Figure 4.20: Box plot of the print shop power load for the biennium 2016-2017.	53
Figure 4.21: Histogram of the total power load for the biennium 2016-2017.	54
Figure 4.22: Histograms of the total power load for each month of the biennium 2016-2017.	54
Figure 4.23: Histograms of the total power load for each day type of the biennium 2016-2017.	55
Figure 4.24: Total corrected power carpet plot.	56
Figure 4.25: Not allocated corrected power carpet plot.	56
Figure 4.26: DIMAT corrected power carpet plot.	57
Figure 4.27: Refrigeration unit n°2 corrected power carpet plot.	57
Figure 4.28: Data centre corrected power carpet plot.	58
Figure 4.29: Bar Ambrogio corrected power carpet plot.	58
Figure 4.30: Canteen corrected power carpet plot.	59
Figure 4.31: Rectory corrected power carpet plot.	59
Figure 4.32: Print shop corrected power carpet plot.	60
Figure 4.33: Histogram of the total corrected power load for the biennium 2016-2017.	60

Figure 4.34: Histograms of the total corrected power load for each month of the biennium 2016-2017.	61
Figure 4.35: Histograms of the total corrected power load for each day type of the biennium 2016-2017.	61
Figure 4.36: Histogram of the not allocated corrected power load for the biennium 2016-2017.	62
Figure 4.37: Histograms of the not allocated corrected power load for each month of the biennium 2016-2017.	62
Figure 4.38: Histograms of the not allocated corrected power load for each day type of the biennium 2016-2017.	63
Figure 4.39: Histogram of the DIMAT corrected power load for the biennium 2016-2017.	63
Figure 4.40: Histograms of the DIMAT corrected power load for each month of the biennium 2016-2017.	64
Figure 4.41: Histograms of the DIMAT corrected power load for each day type of the biennium 2016-2017.	64
Figure 4.42: Histogram of the refrigeration unit n°2 corrected power load for the biennium 2016-2017.	65
Figure 4.43: Histograms of the refrigeration unit n°2 corrected power load for each month of the biennium 2016-2017.	65
Figure 4.44: Histograms of the refrigeration unit n°2 corrected power load for each day type of the biennium 2016-2017.	66
Figure 4.45: Histogram of the data centre corrected power load for the biennium 2016-2017. ..	66
Figure 4.46: Histograms of the data centre corrected power load for each month of the biennium 2016-2017.	67
Figure 4.47: Histograms of the data centre corrected power load for each day type of the biennium 2016-2017.	67
Figure 4.48: Histogram of the Bar Ambrogio corrected power load for the biennium 2016-2017.	68
Figure 4.49: Histograms of the Bar Ambrogio corrected power load for each month of the biennium 2016-2017.	68
Figure 4.50: Histograms of the Bar Ambrogio corrected power load for each day type of the biennium 2016-2017.	69
Figure 4.51: Histogram of the canteen corrected power load for the biennium 2016-2017.	69
Figure 4.52: Histograms of the canteen corrected power load for each month of the biennium 2016-2017.	70
Figure 4.53: Histograms of the canteen corrected power load for each day type of the biennium 2016-2017.	70
Figure 4.54: Histogram of the rectory corrected power load for the biennium 2016-2017.	71
Figure 4.55: Histograms of the rectory corrected power load for each month of the biennium 2016-2017.	71
Figure 4.56: Histograms of the rectory corrected power load for each day type of the biennium 2016-2017.	72
Figure 4.57: Histogram of the print shop corrected power load for the biennium 2016-2017. ...	72
Figure 4.58: Histograms of the print shop corrected power load for each month of the biennium 2016-2017.	73
Figure 4.59: Histograms of the print shop corrected power load for each day type of the biennium 2016-2017.	73
Figure 5.1: Follow the leader clustering results on the total power load.	76
Figure 5.2: Calendar plot of the clusters	77
Figure 5.3: Follow the leader clustering performed only on the initial cluster 5.	78

Figure 5.4: Refrigeration unit n°2 daily profiles. The two profiles above refer to usual operation while those under are the rare ones of cluster 5_2.....79

Figure 5.5: Refrigeration unit n°2 daily profiles. The rare event of cluster 5_4 starts between 2017-06-05 and 2017-06-16.80

Figure 5.6: Rare events and acquisition anomalies library.....80

Figure 5.7: Calendar plot of the acquisition anomalies and rare events library.....81

Figure 5.8: Intermediate clustering result without anomalies and rare events.....82

Figure 5.9: Correlation matrix.....83

Figure 5.10: Final clustering results on the total power load.....84

Figure 5.11: Calendar plot of the final clusters. White cells represent anomalies and rare events.85

Figure 5.12: Clusters classification tree.....87

Figure 5.13: CART of the normalised clusters for refrigeration unit n°2 (above). Refrigeration unit normalised centroids with relative interval of two times the cluster standard deviation (below)...88

Figure 5.14: CART of cluster 2 profiles intensity (left). Cluster 2 profiles intensity with relative interval of two times the sub-cluster standard deviation (right).....89

Figure 6.1: Model general framework.....93

Figure 6.2: Regressive CART.....94

Figure 6.3: Non-autoregressive artificial neural network structure.....95

Figure 6.4: First random search results.97

Figure 6.5: Second random search results.....98

Figure 6.6: Comparison of expected and real consumption during the event "CF2 flat" (above) and during normal operation (below).....99

Figure 6.7: Comparison of expected and real consumption during the event "Everything flat" (above) and during normal operation (below). 100

Figure 6.8: Comparison of expected and real consumption during the event "Everything flat" (above) and during normal operation (below). 101

Figure 6.9: Anomaly detector functioning during normal operation..... 104

Figure 6.10: Anomaly detector functioning during a rare event..... 104

Figure 6.11: Rare events consumption and expected one..... 106

Figure 7.1: Forecasting model general framework..... 109

Figure 7.2: Autocorrelation of the power load..... 110

Figure 7.3: Autocorrelation function of the power load of working days. 111

Figure 7.4: Power load forecast MAPE in testing for working days and holidays..... 112

Figure 7.5: Residual distribution of the working days neural networks for each forecasting horizon. 112

Figure 7.6: Residual acf of the working days neural networks for each forecasting horizon.... 113

Figure 7.7: Final service mean percentage comparison for clusters 1_1, 3 and 5_3..... 114

Figure 7.8: 24 hours ahead power load shape forecasting, with final service percentage from clusters for a working day. 115

Figure 7.9: 24 hours ahead power load shape forecasting, with final service percentage from clusters for a weekend..... 116

List of tables

Table 1.1: Classification of some energy analysis methods. Adapted from ASHRAE Handbook Fundamentals (6).....	5
Table 4.1: Electrical substation C services and respective identification codes.....	41
Table 5.1: Cluster merging summary.....	84
Table 6.1: Rare events detection and labelling results.	105
Table 7.1: Model inputs for each forecasting horizon. Timestamp includes month, time and day type.....	109

10 Appendix

The following table summarizes the majority of the papers which have been reviewed in this work. They are organised depending on their main purpose and field of application and, for each of them, the key features are reported, with a particular focus on the goal of the paper, the utilised methods and relative parameters and the resulting implications.

Ref.	Category	Goal	Dataset and horizon	Model	Pre-Processing	Input	Parameters	Results	Implications
(37)	Demand Response	Providing a baseline useful to implement DR (Demand Response) programs	4 office buildings Data sampled each 15 minutes Hourly load for the 8 working hours of following day	SVR	Various combinations of previous temperatures are tested to select the best solution	Time, pre-two hourly average temperature, historical load of the 8 working hours of the previous day	10 days before the DR event for training and 8 hours for testing (corresponding to the DR event)	Best result: MAE = 1.57%	SVR can be used to evaluate a DR baseline which is used to measure the energy that would have been consumed in absence of a DR program The performance of this method is higher than the one of traditional methods
(36)	Demand Response	Predict the electricity load profile and the daily peak	Tertiary building complex (3 different buildings) 15-minutes time intervals Weather data taken from a local forecasting service every 3 hours 24 hours ahead electricity demand prediction	Random forests 9 different machine learning models are compared and the one with the best result is chosen: ANN	Weather data interpolated to 15-minutes intervals Time indicators variable transformed into integer values from 0 to 95 Categorical variables transformed into a set of indicator variables (ex. 1=1,0,0 for weekdays) Random forests algorithm used to assess the importance of the variables, minimizing the MSE of the prediction	Input variables selected with the Random forests algorithm: operational condition, time indicator, day type, outdoor dry-bulb temperature, outdoor relative humidity, previous electricity usage	Bayesian regularized neural network with LM backpropagation algorithm and 3 layers 50 neurons in the hidden layer and 6 time delayed input variables (up to 90 minutes before) selected by varying neurons number from 10 to 90 and time delay from 1 to 8 Testing period: August and September 2012 Training: - static training (4 weeks of July) - accumulative training (from 1st of July to the day before the target one) - sliding window training (4 weeks updated daily)	Daily electricity usage pattern: CV(RMSE) \approx 10% for weekdays Daily peak: averaged APE \approx 5% for weekdays	The model is sufficiently accurate and could be used to control the daily load profile of a building, and to control demand-response for money savings. It could be used also to control the priorities of the end-used energy consumption.

(39)	Fault Detection	Investigating the performance of tree-structured learning method for FDD (Fault Detection and Diagnosis) of a building cooling system (detecting the fault type and severity)	90-ton centrifugal water-cooled chiller data provided by ASHRAE and collected every 10 s	TFDK (Tree-structured Fault Dependence Kernel method) Wavelet-based De-noising Modified Thompson's Tau method	Wavelet-based De-noising is used to remove the periodic patterns Modified Thompson's Tau method is used for outlier detection	24 variables, including direct sensor measurement and calculated physical indexes	Nodes 1–29 are classification categories: the normal situation corresponds to node 1, and the seven considered types of faults, with their four severity levels, corresponds to nodes 2–29 The model has a first training phase with an ad hoc algorithm and then is trained on-line 8 different sample sizes: 6, 12, 18, 30, 48, 90, 120, and 180	TFDK has 1.49–9.19% improvement in accuracy and 10.69–75% decrease in testing cost with respect to the second-best method	TFDK shows better results compared with other state-of-the-art methods: Multi-class SVM (MSVM) with RBF kernel, Decision Tree (DT), Neural Network (NN), Ada Boost (AB), Quadratic Discriminant Analysis (QDA) and Logistic Regression (LR). With small sample size the accuracy of TFDK and MSVM is similar but increasing the sample size the accuracy of TFDK is definitely higher
(38)	Fault Detection	Developing a fault diagnosis method of photovoltaic (PV) arrays for improving its reliability, efficiency and safety	Accurate Simulink based PV modelling approach simulates a laboratory PV array to ease the acquisition of data and the simulation of the faults 4800 simulation data samples from Simulink and 2500 from laboratory	KELM (Kernel based Extreme Learning Machine) NMS (Nelder-Mead Simplex) EHA-NMS (Eagle strategy based Hybrid Adaptive Nelder-Mead Simplex algorithm)	EHA-NMS is used to extract the model parameters based on the single diode model structure Parameters from I-V curves normalized by dividing them for their ideal values	Measured or simulated I-V curves (open-circuit voltage, short-circuit current, MPP voltage and current, ideality factor, series resistance) and RMSE of the parameter identification	Considered faults: degradation fault, short-circuit fault, open-circuit fault and partial shading condition NMS is used to optimize the parameters of KELM KELM has 5 outputs corresponding to the normal condition plus the 4 considered faults	Testing accuracy on 100 random runs: Only simulated data: 100% Only experimental data: 98.77-99.40% Mixed data: 98.88%	KELM has the advantage of fast learning and generalization with respect to other machine learning methods. The results show very high accuracy in fault detection
(40)	Fault Detection	Presenting an unsupervised learning-based strategy for fault detection of AHU sensors	Data collected from a simulation system	PCA (Principal Component Analysis) OPTICS (clustering algorithm Ordering Points to Identify the Clustering Structure)	Feature extraction based on PCA	Supply air temperature sensor, supply and return chilled water temperature sensors, supply air flow rate meter and supply water flow rate meter	OPTICS: minimum number of data points set to 15, radius set to 1.2	The proposed model is able to detect single and multiple sensor faults in AHUs	The faults are detected by identifying the spatial and temporal separation of the measured data OPTICS fault detection is not sensitive to input parameters defined by the user (radius and minimum number of data points)

(41)	Fault Detection	Presenting an effective diagnostics tool to improve the energy efficiency and thermal comfort of buildings through removing faults	HVAC simulator based on TRNSYS	Wavelet analysis BPNN (Back-propagation Neural Network) PCA (Principal Component Analysis) Subtractive clustering	Wavelet analysis is used to process the raw data, removing useless and disturbing ones Sensitivity analysis on the input parameters to choose the right output for the auxiliary BPNN	BPNN: set-point temperature, supply chilled water temperature, return chilled water temperature, chilled water flow rate and chilled water valve position Auxiliary BPNN: set-point temperature, supply chilled water temperature, supply air temperature, chilled water flow rate and chilled water valve position	BPNN: LM training algorithm, the output is supply air temperature Auxiliary BPNN: LM training algorithm, the output is return chilled water temperature PCA is used to determine the weighting factors for the 2 BPNNs Clustering is used to create a library of faults BPNNs are used to determine if a fault is occurring (looking at the relative error), then the clustering analysis is needed to determine which is the kind of fault or to add a new fault to the library The detection threshold set at 2%	Six different faults are examined: false alarm ratio: 0.6-7.7% missing alarm ratio: 0-8.3%	With the clustering analysis the faults can be isolated accurately The 2 BPNNs together show good results and short fault detection time
(11)	Forecasting	Demonstrate the validity of NAR for time series forecasting when there is only one time series involved	3 years of monthly electricity energy consumption of a tertiary sector building Prediction of monthly electricity energy consumption for 3, 6 and 12 months ahead	NAR (Nonlinear Auto Regressive Neural Network)	Last 5,6 or 7 values of the time series, according to ACF	1 hidden layer with 6 neurons Sigmoid transfer function in the hidden layer Levenberg-Marquardt training algorithm 50% training, 45% validation, 5% testing	RMPSE = 15.7% for 3 months ahead RMSPE = 17.97% for 6 months ahead RMSPE = 14.59% for 12 months ahead	NAR gives the opportunity to make a working analysis and diagnosis tool for building energy demand, when the only data available are the past energy consumptions	
(19)	Forecasting	Show that ANN temperature prediction has better results than empirical equations	Daily maximum and minimum air temperature and hourly air temperature of 1 year (3 data sets from 3 weather station in different locations) 24 hours ahead temperature forecasting	MLP (Multi-Layer Perceptron) RBF ANN (Radial Basis Function)	Data normalized in the interval [0.05, 0,95] to avoid output signal saturation	Maximum and minimum temperatures at day t Maximum and minimum temperatures at day t and t-1	MLP: Levenberg-Marquardt training algorithm, tangent sigmoid transfer function in the hidden layer, 4 neurons in the input, 6 in the hidden, 24 in the output RBF ANN: 4 neurons in the input layer, 75 in the hidden layer and 24 in the output layer 70% training, 30% testing	R ² = 0.96; 0.98; 0.97 in testing depending on the considered data set MAE = 2.98; 0.99; 3.23 in testing depending on the considered data set	Using the maximum and minimum temperatures of day t-1 as inputs leads to a small improvement in the results. MLP performs better than RBF

(2)	Forecasting	Short-term electric load forecasting in buildings using a short training period	21 days for training Hourly data Building scale From ASHRAE competition 1 hour ahead load forecasting	FB (feedback) ANN	Data normalized in the interval [-0.9,0.9] adding ± 10 °C to the maximum and minimum temperatures and ± 50 kW to the maximum and minimum loads Hour and day are coded by means of their sine and cosine values	For the prediction of temperature at (t+1): temperature at time t and the hour t For the prediction of load at (t+1): the ΔT between T(t) and the predicted T(t+1), the hour t, the day of the week and the load at time t	Two FB ANN: one for the prediction of T(t+1) and one for the prediction of load(t+1) 5 neurons in the hidden layer, hyperbolic tangent activation function number of epochs: 1000, first day 750, the rest of the days	First dataset: CV(RMSE) = 1.4423 MBE = 0.0033 Second dataset: CV(RMSE) = 2.55 MBE = 0.0123 MAPE = 1.945	The proposed method is easy to apply for short-term load forecasting and can be used having just the atmospheric temperature and the load measurement available. Further studies may analyse the optimal dimensions of the training window and the parameters of the training algorithm
(12)	Forecasting	Show the good performance of short term load prediction using wavelet and Elman RNN	Hourly total load demand of 10 months and weather data from Indian Meteorological department Forecasting 1-day-ahead electrical power load under the influence of temperature	MRA (Multi-Resolution analysis) Daubechies wavelet Elman RNN (Recurrent Neural Network)	MRA with Daubechies wavelet is used to decompose the load series into 3 components (Approximation A2, Detail D1, Detail D2) Data are normalized in the interval [0,1]	Each component is modelled separately with an Elman RNN (Recurrent Neural Network) The final prediction is obtained by summing the partial ones The number of inputs is chosen to minimize MSE, NMSE and MAPE and to maximize R The inputs are the wavelet of the component (A2 or D1 or D2) and one of the following possibilities: temperature, humidity or temperature and humidity	3 models for different conditions: - Summer (3 months training, 7 days testing) - Rainy (2 months training, 7 days testing) - Winter (2 $\frac{1}{2}$ months training, 7 days testing) A2: 5 inputs, linearaxon transfer function in output layer, 10 PEs (processing elements) in the hidden layer, BP (Back Propagation) learning rule D2: 3 inputs, linearaxon transfer function in output layer, DBD (Delta Bar Delta) learning rule, integrator axon transfer function of PEs in the context unit D1: 3 inputs, tanh transfer function in output layer, 10 PEs in the hidden layer, DBD learning rule	Summer: MAPE = 1.22 STD = 0.40 Rainy: MAPE = 2.83 STD = 0.90 Winter: MAPE = 1.71 STD = 0.75	The best results are obtained when using temperature as the only weather variable. Gaussian noise has been added to temperature in order to verify that the prediction was still good enough.

(20)	Forecasting	Energy demand forecasting for the operation of distributed energy supply plants	Building scale Tertiary sector Hourly data 23 weekdays 3 hours ahead	ANN	Trend and periodic change are removed from the data (using ordinary and periodic differential operation) before passing them to the neural network	A first ANN predicts the temperature and relative humidity and then the results are fed to another ANN which predicts the energy demand	Global optimization by modal trimming method The number of neurons is determined by means of the global optimization method The ANN has an input layer, a hidden layer and an output layer	Relative error = 8.01%	Global optimization and periodic differential operation improve the forecasting accuracy. Taking into consideration temperature and relative humidity can improve the accuracy of the identification of the values of the parameters needed for the ANN, but that it may decrease the accuracy of the prediction
(21)	Forecasting	Compare two types of adaptive ANN for on-line energy forecasting in buildings	Simulated hourly data for tertiary sector building (outdoor dry-bulb and wet-bulb temperatures, supply water temperature, chiller electric demand) Measured hourly data for another tertiary sector building 1 hour ahead electrical energy demand of the chiller	Static ANN Adaptive-ANN: - accumulative training (1) - sliding window training (2) PCA (Principal Component Analysis)	Non-working hours removed from the data set, since one does not need predictions in such periods All data normalized between [-1,1]	Simulated data: dry and wet-bulb temperatures of past 6 hours, supply temperature of past hour PCA can be used to remove potential redundancy in the data Measured data: temperature of the water entering ice tank, temperature of the water at the evaporator input and output, outdoor relative humidity, outdoor temperature, electric current used by the chiller, weekday schedule, holiday indicator, percentage of chilled water in the ice tanks, logical variable for chilled water in the ice tanks ready or not	Logistic sigmoid activation function in each neuron, linear transfer function in the output layer Accumulative training: data are updated each day by adding new data to all the past ones, then the ANN is retrained Sliding window training: data are updated each day by moving the sliding window (20 days) one day ahead, then the ANN is retrained	1) Simulated data: Static ANN with PCA: CV = 0.07 Accumulative training ANN with PCA: CV = 0.15 Sliding window training: CV = 0.40 Sliding window training with PCA: CV = 0.15 2) Measured data: Static ANN with PCA: CV = 0.26 Accumulative training ANN: CV = 2.53 Sliding window training ANN: CV = 0.26	Adaptive ANN use a training dataset that is constantly updated to adapt the network to unexpected changes of pattern: for this reason, they can be used for on-line building load prediction.

(13)	Forecasting	Use ANN (coupled with GA to avoid slow convergence and local minimum stagnation) to analyse historical energy consumption and make daily energy forecasting	Daily energy consumption and temperature for 8 tertiary sector buildings (one model for each building) 1 day ahead energy consumption forecast	NAR (Nonlinear Auto Regressive Neural Network) NARX (Nonlinear Auto Regressive Neural Network with exogenous inputs) ENN (Elman Neural Network) GA (Genetic Algorithm)	All data are normalized between [0,1]	Previous energy consumption and temperature (depending on the model)	Random split of data for training (70%) and testing (30%) LM (Levenberg-Marquardt) training algorithm Hyperbolic tangent sigmoid transfer function for hidden layers For NAR and NARX the number of neurons in the input and hidden layers depend on the building, for ENN there are 10 hidden neurons and a memory of 5 past values	The methods are ranked like this in all but one case: 1) ENN with temperature (MSE = 0.004413) 2) ENN (MSE = 0.005085) 3) NAR 4) NARX with temperature	NAR(X)+GA brings to an average 35% improvement while GA+ENN brings to an average 61% improvement with respect to simple NAR and NARX models
(14)	Forecasting	To study the performance of the neural network by varying more parameters at the same time, to consider their interactions. To demonstrate the superior performance of the 'quasi optimal' neural network over an ARIMA model for the STLF (Short Term Load Forecasting) problem	Six years of hourly load and daily temperature for an electric utility Hourly 24 hours ahead load forecast	Multi-layer feedforward neural network	Several different configurations for the neural network model is tested to find some rules that lead to the 'quasi-optimal' one	20 input nodes for time variables (12 for month, 7 for day, 1 for holiday with values 0 or 1) 18 input nodes for temperature variables 48 nodes for historical load input variables	1 year for cross validation, 1 year for testing and 2 or 4 years for training Varying parameters: number of hidden layers type of transfer function learning algorithm adding noise to training input vector stopping rule feedback (ENN) training data set dimension time of peak (winter or summer) industrial load% (high or low)	RMSE = 2.01% - 3.87%	ANN outperforms ARIMA The following rules have to be applied with the same input format of this work and simultaneously: Sigmoid transfer function for the output layer Sinusoid transfer function in the hidden layers One hidden layer with a number of nodes determined with the Cascade Correlation algorithm Cumulative backpropagation rule Concurrent Descent method as stopping criteria Use ENN Network retrained monthly Training data should include load behaviour during both normal and abnormal weather conditions Learning momentum rates and learning schedules as described in a mentioned paper Noise added to input data to avoid local minima

(22)	Forecasting	Predict the electricity consumption and the daily peak improving the prediction accuracy	One year of data collected at 15-minutes intervals 24 hours ahead	Feature extraction Clustering (entropy weighted k-means, EWKM) GESD (Generalized extreme studentized deviate) RFE (recursive feature elimination) LGOCV (leave-group-out cross validation) GA (genetic algorithm) 8 predictive algorithms: multiple linear regression (MLR), autoregressive integrated moving average (ARIMA), support vector regression (SVR), random forests (RF), multi-layer perceptron (MLP), boosting tree (BT), multivariate adaptive regression splines (MARS), and k-nearest neighbours (kNN)	Feature extraction to remove seasonal effect Daily power consumption centred by daily mean and then scaled by daily standard deviation Outlier detection: clustering analysis, and GESD (outliers are deleted from the training dataset)	12 meteorological variables, 5 time variables (month, day, weekday, hour and minute) Model's input determined by RFE up to 15 days before For next-day energy consumption, the numbers of inputs are 24, 29, 31, 36, 33, 12, and 27 for MLR, SVR, RF, MLP, BT, MARS, and kNN, respectively. For the next-day peak power demand the numbers are 27, 30, 34, 35, 33, 15, and 30	70 % training and 30 % testing Parameters optimized using the LGOCV (ex. For MLP the hidden neurons are 20 and 14 for daily energy consumption and peak power respectively) Ensemble model generated using the GA (with the population size, number of iterations, mutation chance, and the proportion of elitism equal to 250, 1000, 0.1 and 0.2, respectively) and minimizing the MAPE	Next-day energy consumption: MAPE = 2.32% (ensemble) MAPE = 3.11% (SVR) Peak load demand: MAPE = 2.85% (ensemble) MAPE = 3.34% (SVR)	For both predictions SVR and RF produce the most accurate results Ensemble models make each model complement with another and this leads to better generalization performance RFE process can efficiently select the input variables and reduce the computation load while improving prediction accuracy
(15)	Forecasting	Forecast both, half-hourly and daily electricity consumption time series data	Institutional building 1 year of data Energy consumption data each 30 minutes 1 Day ahead or half an hour	nu-SVR epsilon-SVR DE (Differential Evolution)	The outliers are removed	Past energy consumption (daily or half-hourly)	Half-hourly energy forecast 384 data points for training and 96 points for testing. Daily energy forecast 209 data points for training and 52 points for testing DE is used to select the SVRs parameters and their weights	MAPE = 5.843 for daily forecast MAPE = 3.767 for half-hourly forecast	Daily and half-hourly energy consumption are predicted with a single model without changing the parameters thanks to the DE which assigns to each SVR the right weight The proposed model has better accuracy than single SVR with other evolutionary algorithms (Genetic Algorithm: GA, Particle Swarm Optimization: PSO)

(16)	Forecasting	Comparing prediction capabilities of five different data driven models used to forecast the electricity consumption of a building	Administration building Data: daily electricity usage, daily mean surrounding temperature, daily mean global irradiance, daily mean humidity, daily mean wind velocity, weekday Index (0,1) Daily electrical energy consumption	MR (Multiple Regression) GP (Genetic Programming) ANN DNN (Deep Neural Network) SVM	-	Solar radiation, temperature, wind speed, humidity and weekday index	4 years for training and 1 for testing	MAPE: 6% ANN 8.5% MR 8.7% GP 9% SVM 11% DNN	ANN has the best performance Each model forecasts electricity consumption of working days within a range of 3% (aggregating the results to monthly values)
(17)	Forecasting	Offering a feasible method for energy prediction based on the historical meter readings	Hotel 1 hour ahead	MWD (Multiresolution Wavelet Decomposition) Reconstruction and feature extraction SVR	MWD removes random disturbances of the historical electric power series and illustrates the special periodic features (dividing into low and high-frequency parts: $\Lambda 3$, D1, D2 and D3) The decomposed parts and feature extraction matrix are superposed Input parameters normalized in the range [-1, 1]	Historical electric power records, meteorological attributes and day type	Equilibrium factor (C) and σ^2 are equal to 1 The non-insensitive error threshold (ϵ) varies from 0.05 to 1 for testing the sensitivity of the MWD	The optimal daily mean relative error of a typical day is around 5.6%	Results show that the MWD processing can reduce the deviations slightly only when ϵ is higher than 0.1 In an optimized SVR model the MWD process seems to be redundant

(18)	Forecasting	Compare the forecasting performance of ANN and RF for hourly HVAC energy demand forecasting	Hotel 5 min historical values of HVAC electricity consumption Total daily number of guests and rooms booked 30 min weather conditions from weather station located nearby: outdoor air temperature, dew point temperature, wind speed and relative humidity 1 hour ahead	ANN RF (Random Forest)	Input and output parameters were normalized between 0 and 1	The wind speed demonstrated to decrease the prediction performance while previous hour consumption increased it Input parameters selected with the variable importance plot	6 days for testing ANN: 1000 epochs, Broyden–Fletcher–Goldfarb–Shano (BFGS) training algorithm, 10 hidden neurons, 1 hidden layer RF: max depth = 10, number of features = 5	RMSE: 4.97% ANN 6.10% RF MAPE: 4.09% ANN 4.60% RF	ANN performs better than RF in testing, however RF can handle the missing values, since it is an ensemble-based algorithm Both ANN and RF can be used to predict HVAC hourly energy demand
(23)	Forecasting	Creating a building occupancy model without any simplifying assumption	Room of educational building 15 minutes time interval 70 days of data 15 minutes ahead	GAN (Generative Adversarial Network) composed by a generative and a discriminative network	data generated: 700 days	Random variables: time of first arrival, time of last departure, cumulative occupied duration, and number of occupied/unoccupied transitions	The generative network produces an occupancy time series starting from random inputs; the discriminative network tries to distinguish the occupancy time series generated by the previous network and the real ones. The goal is to generate occupancy time series that are indistinguishable from the real ones Parameters optimized with grid search Output: mean occupancy Hidden nodes: 240; learning rate: 8.0×10^{-5}	NRMSD = 0.1424	The proposed model shows better results with respect to traditional ones (agent-based model ABM and inhomogeneous Markov chain IMC)

(24)	Forecasting and Diagnosis	Compare different methods for energy demand prediction and suggest a methodology for assessing the thermal response time of buildings	Tertiary sector building with GSHP (Ground Source Heat Pump) Data acquired for 14 days during winter Sampling interval of 5 minutes Time ahead of energy demand prediction taken as the thermal response time of the building (40 minutes selected as the time lag between local minima of heat pump power and indoor temperature)	ELM (Extreme Learning Machine) BPNN (Back Propagation Neural Network) SVR (Support Vector Regression) MLR (Multiple Linear Regression)	Correlation analysis and LASSO feature selection are used to optimize the input variables ACF and PACF are used to determine the influence of previous energy consumption on the future one	The parameters which can be selected by the feature selection methods are: 7 meteorological parameters, operating parameters, time and indoor temperature 5 sets of variables are selected The variables that are present in each set are: hour, outdoor temperature (integrated or not), supply temperature and indoor temperature	7 days for training and 7 days for testing BPNN: 3 layers; different combinations of neurons number and n+1 hidden neurons; learning rate = 0.1; number of iterations = 200; training accuracy = 0.00001 SVR: RBF function as kernel function, Gaussian kernel parameter and penalty coefficient optimized with grid search and ten-fold cross validation ELM: 3 layers; different combinations of neurons number, SLFN (Single Layer Feedforward Network) learning rule, sigmoidal activation function	Results on testing: ELM: RMSE = 3.824; MAE = 2.4; MAPE = 0.116 BPNN: RMSE = 4.105; MAE = 2.76; MAPE = 0.132 SVR: RMSE = 4.464; MAE = 2.974; MAPE = 0.146 MLR: RMSE = 3.857; MAE = 2.297; MAPE = 0.113	The best result in testing in terms of RMSE is given by the ELM model optimized by feature selection The thermal response time of the building can be assessed with the described methodology
(25)	Forecasting for control	Optimize HVAC thermal energy storage in public buildings as well as office buildings	5 years of hourly data generated by EPS-r simulation software Tertiary sector Building scale Forecast of: hourly cooling load for the next day total cooling load for entire day	GRNN (General Regression Neural Network)	Linear scaling factor [-1,1] for N-S buildings Logistic scaling function (0,1) for E-W buildings	External dry bulb temperature of the previous 24 hour	Logistic activation function City-Block distance metric 80% randomly selected data for training and 20% for testing 24 input neurons, 1450 neurons in the hidden layer, 1 output neuron	R ² = 0.9552 hourly cooling load R ² = 0.949 daily cooling load (N-S) R ² = 0.964 daily cooling load (E-W)	The optimum GRNN design can be different from one building to another. Using only external dry-bulb temperature as an input, the controllers which use this ANN method will be easy to make and trustworthy

(26)	Forecasting for control	Short term load forecasting to adjust the HVAC system to the actual load and enhance the operational safety 24 hours ahead load forecasting to detect the change of the cooling and heating load and choose the operational strategy which leads to the most efficient management of the HVAC systems	Tertiary building Hourly historical consumption simulated with DesignBuilder and weather data from meteorological station 1 hour ahead, 2 hours ahead, 3 hours ahead and 24 hours ahead load forecasting	Wavelet transform SVM (Support Vector Machine) PLS (Partial Least Squares regression) PSO (Particle Swarm Optimization) Sensitivity analysis Correlation analysis	The wavelet transform is used to extract load features (dividing data into low and high-frequency components) PLS is used for modelling the low-frequency component A2 (with strong periodicity and linearity) AVM is used for modelling the high-frequency components D1, D2 (with strong randomness and nonlinearity)	Sensitivity analysis is used to determine the meteorological parameters in input Inputs for cooling load prediction: historical load, dry-bulb temperature (from 3 hours before to current hour and also the value at prediction time) , relative humidity, solar direct normal radiation and solar diffuse horizontal radiation Inputs for heating load prediction: historical load, dry-bulb temperature (from 3 hours before to current hour and also the value at prediction time), solar direct normal radiation and solar diffuse horizontal radiation Different combination of past values of each input parameters are selected for the different forecasting horizons: correlation analysis used for further selection of meteorological parameters (past values can influence the model because of thermal inertia) and historical load:	PSO is used to optimize the parameters for SVM ($\epsilon = 0.1$, $C \in [0.01, 100]$, $\gamma \in [0.1, 100]$) Cooling load: 3 months for training and 2 for testing Heating load: 3 months for training and 25 days for testing	1 hour ahead: MARE = 2.60% (cooling) MARE = 3.99% (heating) 2 hours ahead: MARE = 6.17% (cooling) MARE = 6.77% (heating) 3 hours ahead: MARE = 8.54% (cooling) MARE = 6.84% (heating) 24 hours ahead: MARE = 9.87% (cooling) MARE = 12.19% (heating)	The proposed model (wavelet-PLS-SVM) shows better accuracy than the single PLS, SVM and wavelet-SVM The forecast error of weather variables is also considered: when the forecast error of temperature is 1.68°C and the forecast error of relative humidity is 11.7%, the MARE increases by 1.56% (24 hours ahead) The prediction at 1,2 or 3 hours ahead are less influenced by the forecast of the weather parameters because of the higher precision of these forecasts
------	-------------------------	---	--	---	---	---	---	--	---

(27)	Forecasting for control	Providing good internal temperature forecast for improving the performance of temperature controllers	Public dataset Data collected every 15 minutes 48 hours ahead each 15 minutes	Ridge regression Lasso regression	-	six environmental factors: inside and outside temperature, inside and outside humidity, windspeed and rain 26 circuits 4 motion detectors 1 light switch	Different number of lagged values for each case (0,1,2,4,8,16,32,64,96) 1 model for each 15 minutes interval up to 48 hours	Forecasting temperature Ridge regression MAE = 1.77 Lasso regression MAE = 1.73 Forecasting temperature difference Ridge regression MAE = 0.042 Lasso regression MAE = 0.0472	Forecasts for the next two days within 1.8 °C
(30)	MPC	Develop a zone-level heating set-point scheduler which minimize the energy consumption	Tertiary building Generated with a software Hourly values Hourly 24 hours ahead energy consumption and indoor temperature prediction and hourly 24 hours ahead set-point schedule	ANN (used to predict the energy consumption and the indoor temperature) GA (used to optimize zone temperature set-point: minimizes the energy consumption while guaranteeing suitable indoor environmental conditions) As an additional case one can use also a MPC (the set-point schedule is uploaded each hour for the next 24 hours so that the system considers also the feedback from the building)	Data are divided to be fed to an ANN for each zone	ANN input parameters: outdoor temperature, relative humidity and solar irradiance (from weather station), hour, set-point temperature, binary occupancy factor, indoor temperature of previous 3 time steps (predicted indoor temperature at time t+1 used as an input for time t+2 and so on)	ANN: structures differ from a zone to another (either 5-20-20-2 or 6-20-20-2), LM training algorithm, tangent sigmoid transfer function GA: 24 variables (temperature set-point for each hour), population size = 200, maximum number of generations = 2400) MPC: building simulated with energy plus Occupancy and weather conditions forecast assumed as 100% accurate	ANN energy prediction: CVRMSE around 30 % ANN indoor temperature prediction: CVRMSE < 2% Energy savings with electricity flat tariff: around 25 % Energy savings with electricity time of use tariff: 23.31 % (day ahead), 21.28% (MPC) Money savings with electricity time of use tariff: 27.94 % (day ahead), 27.26% (MPC)	This solution requires a small amount of additional hardware and brings to significant simulated energy savings with the advantage of simulating the building behaviour with a simple ANN model that is fast to run
(29)	MPC	Increasing the SCOP (Seasonal Coefficient Of Performance) of an air-to-water HP (Heat Pump) using predictive control	Hourly temperature values for 10 years in Central Europe 24-48 hours ahead temperature prediction	Predictive control for air-to-water HP	-	Outdoor air temperature forecast	The HP is operated during the hours of the day with highest outdoor air temperature, to satisfy the building heating energy need and to fill a heat accumulator which will be used during the remaining hours	SCOP increase with respect to the HP without predictive control: 19% (24 hours T prediction) 23% (24 hours T prediction)	The increased SCOP value corresponds to around 20% decrease in yearly power consumption and the 48 hours ahead predictive control leads to a significant reduction of the HP's starts, corresponding to an increase in its lifetime

(31)	MPC	Implementing a model predictive control in a commercial building, to determine the optimal time at which turning on the boiler leads to achieve the target temperature at 8:00 am	Commercial building Data from 2 heating seasons 15 minutes ahead	ANN	-	ANN inputs: average internal temperature, external temperature and water heating system temperature.	ANN output: time required in quarters of an hour for conditioning the building at 20°C at 8:00 am Hyperbolic tangent activation function in the hidden layer and linear function in the output layer 1 hidden layer with 10 neurons	Gas energy consumption reduced by 19.69% with respect to on-off regulated by the energy manager	Savings increase in the months in which the external temperature is closer to the base temperature (17°C) The system is automatic and does not require any action from the energy manager The savings obtained are higher than those reached in similar studies proposing control strategies based on commercial BEMS (Building Energy Management System)
(32)	MPC	Evaluate the efficiency of a model predictive control on a heating system: the model determines if the system must be switched ON/OFF to control indoor temperature (which behaviour is predicted with ELM ensemble)	Living lab Data points are collected every 5 min; external forecasts are provided three times a day with a granularity of 15 min and a prediction horizon of 8 h The control policy is evaluated three times a day, as new forecasts arrive, and dispatched on 5 min basis	MABRL (Model-Assisted Batch Reinforcement Learning) FQI (Fitted Q-Iteration) ELM (Extreme Learning Machine) Triangular membership functions (for policy shaping)	-	Time, indoor air temperature, outside temperature, solar irradiance and control action at previous timestep (HVAC system ON/OFF)	ELM: one output (change of internal temperature), regularization term: $C = 100$ Ensemble of 40 single-output ELMs obtained as a weighted average of the single outputs	A performance of 90% of a mathematical optimum is reached in around 20 days	The HVAC system switches on at the beginning of a low-price period and for two different outside temperature regimes. Qualitative results show that adding virtual data from a support model (ELM) can improve the efficacy of the control policies when there are few of real tuples in the batch, but this leads only to a small increase of performance
(33)	MPC	Optimize a multi-zone Air Handling Unit's (AHU) energy consumption implementing a Nonlinear Model Predictive Control (NMPC) which minimizes the energy consumption and the deviation from the set-point temperature	AHU with 3 zones serving an educational building 17 minutes time step Control horizon and prediction horizon equal to 119 minutes	NARX (Nonlinear Auto Regressive Neural Network with exogenous inputs) Discrete GA (Genetic Algorithm)	-	NARX: chilled water flow rate, damper position of the zones, air handling unit's status, ambient temperature, fresh air damper position, previous temperatures of the zones	GA output: chilled water flow rate, damper position of the zones, air handling unit's status NARX: the output is internal temperature at next time step for each zone, the number of output delays is 2, 35 hidden neurons 5 days for training and 2 for testing	The optimal NARX accuracy is equal to 97.71%. Reduction of consumption NMPC vs. no control: 55.1% electricity 43.7% gas Reduction of consumption NMPC vs. on/off: 18.5% electricity 17.4% gas	The future set-point starts 2 h before the deviation between the internal temperature and the set-point takes place. This demonstrate the ability of the model to act in advance, taking into consideration the thermal inertia of the building. The model shows better performance with respect to the traditional on/off control

(34)	MPC	Create a DDC (Demand Driven cooling Control) to increase the efficiency of HVAC systems in accommodating occupants' behaviour in real time The information about occupancy learned by the algorithms is used to control room setpoints in real-time and the cooling system The RBC (Rule Based Control) can change the temperature set-point (3 possibilities plus a set-back)	11 rooms of a commercial building 10 minutes time delay 1 stands for a monitored room which is occupied and 0 for one that is not occupied 2 datasets: historical weekday dataset (updated daily) and a current-day dataset	Feature selection Clustering KNN (k-Nearest Neighbour)	A global training dataset is created from the weekday dataset. Local training datasets are the result of a clustering analysis of the global training dataset on the specified number of occupancy patterns in individual rooms (each training dataset corresponds to a kind of occupancy pattern in the global one)	Chosen by feature selection: time of the daily first arrival, time of daily last departure and daily maximum vacancy duration during working hours	Clustering defines 4 occupancy patterns KNN: Hamming distance computation, threshold value set to 0, outputs: time of the occupant's next presence and total presence duration in the remaining day	The energy savings in the multi-person offices are 7% because of the high occupancy The energy savings in the single person offices with medium occupancy rates are 21% The energy savings in the meeting room are 52% Globally the DCC saved 21% of the sensible cooling energy	The method shows very good results (between 7% and 52% energy savings) with respect to the conventionally-scheduled cooling systems The difference in the means of temperature deviations in the baseline and DCC tests is less than 0.1 °C This method takes 12 minutes to activate the cooling system in a room (short-presence detection). In this way unnecessary activations of the cooling system are avoided
(45)	Benchmarking	Develop a building energy use benchmarking system to rank buildings	1000 commercial buildings 57 types of features 2 years of data	Create quantile regression models Construct CDFs to rank the buildings determining the QuantRank score Analyse influence plots	Dataset cleaning to remove buildings that contained missing values, to combine repetitions and to correct erroneous information about building characteristics.	-	92 unique quantile regressions for all values of tau between 0.05 and 0.95 with a step-size of 0.01	-	This method can give a probabilistic interpretation of building energy consumption, taking into account the influence of different external factors (HDD, number of employees, etc.) The proposed framework is versatile and can incorporate unsupervised algorithms, nonlinear modelling dynamics and variable selection CDD have a great effect for bad performing buildings Building owners could use this method to analyse their investments

(46)	Benchmarking	Develop a benchmarking methodology to assess the energy performance of a building starting from the shape of its electrical load. Create a tool which enables to identify potential operational improvements, retrofits and savings and reduce the energy supply cost	Electrical load data from thousands of small or medium commercial buildings at 15-minutes intervals	Clustering	Cluster profiles normalization between 0 and 1	-	Centroids used to determine the starting and ending operating hours	-	Common EPI (Energy Performance Indicators) take into account only the total or global building energy consumption or their intensity, but they cannot give information about the changes in energy consumption patterns. In order to overcome this issue, the proposed model creates a statistical method which considers 24 hours of power shape benchmarking and makes a comparison between different buildings
(47)	Benchmarking	Create a model to analyse building energy consumption data and to detect the ones with similar "temporal energy performance patterns"	100 types of features for thousands of buildings	k-means clustering Logistic regression to predict the cluster to which the profile belongs	Missing values and outliers removal Cluster profiles normalization between 0 and 1	-	Silhouette score and Dunn index used to assess the right k value for clustering	-	Two temporal patterns of building energy performance have been recognised. An increasing and a decreasing one. . Energy consumption decrease showed to be mainly related to office buildings, which was larger, newer, and with higher-value and which presented improvements for what concerns the consumption in the years 2011-2016. This study could be used as a basis of suitable and more fair energy policies

(44)	Benchmarking	Propose a methodology to perform a benchmarking analysis addressed, in particular, to buildings with heterogeneous datasets	dataset of 100 outpatient Healthcare Centres Consumption data and building features collected by energy auditing	Linear Mixed Effects Model Monte Carlo simulation CART	Dataset segmentation by CART to clarify the individual classes Second segmentation according to operating time by expert analysis	Selected by correlation analysis	Three LMEM for each dimensional class Three distributions of energy consumption for each class 70% training and 30% for testing of LMEM Monte Carlo simulation is used to "evaluate the frequency distribution of possible values of the model response following the perturbation of the input variables within their respective ranges of variation"	Variance of the model = 96% with a MAPE around 15%	The end-user can have at disposal the median of the frequency distribution which has been simulated and by using that he can obtain a reference value. Having at disposal also the distribution the user can select the parameter which best reflects its needs
------	--------------	---	--	--	---	----------------------------------	--	--	---