UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH

Adaptive load consumption modelling on the user side: contributions to load forecasting modelling based on supervised mixture of experts and genetic programming

Francisco Javier Giacometto Torres

Departament d'Enginyeria Electrònica
UNIVERSITAT POLITÈCNICA DE CATALUNYA

ADAPTIVE LOAD CONSUMPTION MODELLING ON THE USER SIDE

Contributions to load forecasting modelling based on supervised mixture of experts and genetic programming

Doctoral Thesis presented in partial fulfillment of the requirement for the PhD Degree issued by the Universitat Politècnica de Catalunya, in its Electronic Engineering Program.

Francisco Javier Giacometto Torres

Advisor: Dr. Jose Luis Romeral Martinez

July, 2017

UPC MCIA

AUTHOR'S DECLARATION & DISCLAIMER

I declare that the work in this dissertation was carried out in accordance with the requirements of the University's Regulations and Code of Practice for Research Degree Program. Also declare that this research has not been submitted for any other academic award, except the indicated by specific reference in the text. This work constitutes an intellectual property of the candidate's fruit of his labor in collaboration with, or with the assistance of, others, as is indicated on this document. Any views expressed on this document present the conceptual point of view of the author and cannot be linked with any organizations mentioned.

Signed .... Date ....

The most exciting phrase to hear in science,
The one that heralds new discoveries,
Is not "Eureka!" but
"That's odd...",
"That's funny...",
"Hey, wait a minute...".

—
Isaac Asimov

To the conditions accomplished to make me the self-conscious life form that I am now.
To everyone who trust on my capabilities and enhance them.
To everyone who provide me emotional support to enjoy every moment during my path of knowledge.
To readers, hoping that they enjoy the fruits of my labor and it would be useful for them.

ABSTRACT

The importance of the load forecasting has been acknowledge since the beginning of the XX century. The firsts on spotted its value were public organizations, then it didn't take long time until the estimation procedures were extrapolated to the generation and distribution applications in order to maximize the delivering of power.

Lately, forecasting procedures evolve in quality and complexity, linked to the rise of artificial intelligence algorithms; allowing to implement energy management activities relying on the low levels of uncertainty obtained from the forecast. Load forecasting plays a corner stone on the decision making processes carried out by energy management systems.

The accuracy of the forecast in an essential condition to obtain improvements on cost reduction tasks such as load scheduling, co-generation scheduling, and planning energy purchases. Nevertheless, accuracy on load forecasting is a difficult parameter to achieve; mostly because consumptions are influenced by many factors such as weather conditions, holidays, economy status, and idiosyncratic habits of individual customers.

Over recent years, the study of the short term load forecasting on the user side has been addressed using several types modelling strategies, the most successful ones are centered on the use of non-linear models due to their aptitude to shape strong but non-trivial and non-linear relation between future consumption and factors that produce it (climatic conditions, goals of production, labor at calendar, etc.).

Most of these approaches have considered strategies that benefit the general model accuracy over a specific data set, and leave it behind traits such as the model adaptiveness or the potential diversity captured. There are few studies which addressed the load modelling as a multi-scenario strategy, and are fewer which has developed an exhaustive measure of the model goodness.

This research work proposes three main contributions on the load forecasting field: the enhancement of the forecasting accuracy, the enhancement of the model adaptiveness, and the automatization on the execution of the load forecasting strategies implemented. On behalf the accuracy contribution, learning algorithms have been implemented on the basis of machine learning, computational intelligence, evolvable networks, expert systems, and regression approaches.

The options for increase the forecasting quality, through the minimization of the forecasting error and the exploitation of hidden insights and miscellaneous properties of the training data, are equally explored in the form of feature based specialized base learners inside of a modelling ensemble structure. Preprocessing and the knowledge discovery algorithms are also implemented in order to boost the accuracy trough cleaning of variables, and to enhance the autonomy of the modelling algorithm via non-supervised intelligent algorithms respectively.

The Adaptability feature has been enhanced by the implementation of three components inside of an ensemble learning strategy. The first one corresponds to resampling techniques, it ensures the replication of the global probability distribution on multiple independent training sub-sets and consequently the training of base learners on representatives spaces of occurrences.

The second one corresponds to multi-resolution and cyclical analysis techniques; through the decomposition of endogenous variables on their time-frequency components, major insights are acquired and applied on the definition of the ensemble structure layout. The third one corresponds to Self-organized modelling algorithms, which provides of fully customized base learner's.

The Autonomy feature is reached by the combination of automatic procedures in order to minimize the interaction of an expert user on the forecasting procedure. Experimental results obtained, from the application of the load forecasting strategies proposed, have demonstrated the suitability of the techniques and methodologies implemented, especially on the case of the novel ensemble learning strategy.

KEYWORDS

Energy management

Short time load forecasting

Computational intelligence

Machine learning

Ensemble learning

User side

Industrial user

Commercial user

Residential user

Accuracy

Adaptability

Autonomy

Multi-resolution component

Knowledge discovery Algorithms

Cyclical analysis

ACKNOWLEDGEMENTS

To my parents Francisco Giacometto and Zunilda Torres for raise me up and share their precious soul pieces with me.

To my advisor, mentor, and friend Luis Romeral for sharing pieces of his knowledge and dedication to me with the objective of cultivate and augment my academic and professional skills.

To all the members of the MCIA center for extend their comradery, support, cheerfulness and counseling to me.

To all the people that support me and were not fully addressed.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER

Thesis Delimitation

This chapter presents an overview of the research scope of this thesis. It fulfills the principal reader doubts related with the problem awareness conducted by this research. The contents follows a storyline directed by the problem statement, the statement of research aim and objectives, and the hypothesis delineate for this study.

Contents

1.1 Introduction

The importance of the load forecasting has been acknowledge since the beginning of the XX century. the first on spotted its value were public organizations [1]; they start to consider to make estimations of the energy on the consumption side based on historic registers [2] and weather conditions [3]. It didn't take long time until the estimation procedures were extrapolated to the generation and distribution applications in order to maximize the delivering of power [4]. lately, forecasting procedures evolve in quality and complexity, linked to the rise of artificial intelligence algorithms; allowing to implement energy management activities relying on the low levels of uncertainty obtained from the forecast. it scenario produce a sky-rocket of business activities around the energy services sector with special focus on energy management applications for the demand side [5].

On generation side, especially on the electric utility industry, many of them count with in-house load forecasting capability followed for incipient energy management systems. Meanwhile on the demand side; utilities, residential buildings, commercial buildings, and private householders may have to outsource load forecasting services as well as the energy management. The reason is the high cost implied on the planning, instrumentation and development of an energy management project up to the current state of art. Nevertheless, as the benefits include technological advances and cost-reduction operation, it may become economically justifiable to outsource the energy management.

This chapter presents a brief overview of the load forecasting, scope of research exposed on this thesis, awareness of the problem, problem statement, research aim and objectives, hypothesis, delimitations and motivation of the research assumptions. It also presents a draft of research design and methodologies. At the end of this chapter, the outline of the thesis and the conclusions are provided.

1.2 Thesis framework

20 years ago, we had assisted to the birth of a cultural awareness about the environment and the limited resources that the humanity have. Since then until now, Public and private sectors has been sharing efforts in order to construct a **policy framework** over energy strategies. Consequently, new trends of research on the energy field has flourish; as well as, the develop of new business of opportunities for the upcoming technologies.

The deregulation on the power utility industry was one of the most important political decisions that trigger the development of the energy sector. Its impact has opened a business competition in every aspect of power systems; at power generation, at transmission, at energy consumption; and the most important, the professional management of the electric energy with the correspondent efficiency enhancement.

As side effect, the **market liberalization and de-regularization** has rocket the technological developments on the energy field; the most important one is the enhancement on the operation of the power system networks. It has evolved to use integrate **information based tools** in the electric power generation, demand or load management; these are implemented in order to discovery knowledge based on historic data and support decision making. Between them, **load forecasting** in the most critical tool for the operation of power systems. Those tools make possible the **distributed and controlled generation (DG)**, together with the storing and **demand side management (DSM)**.

The forecast of the load profiles is crucial for network planning, energy scheduling, infrastructure investment, development, and **MANAGEMENT**. On terms of utilization side, load forecasting can be classified on two categories. **Utility based forecasting**, applied on to the power generation side provide assistance to management planning on strategic decisions such as co-ordination on the generation, interchange evaluation, security assessment and other planning tasks. On this side, forecast is applied at a high level of load aggregation, which produces a smooth profile; being the peak detection and consumption offset the most important parameter to forecast.

Consumer based forecasting is often used as a source of information for energy optimization procedures carried out on the transport networks and consumption users. On terms of economic value, the forecasting performance affects organizational process such as manage risk of the supply, infrastructure finance, reduction of operational cost, among others. **Energy service provider companies (ESPC's)** usually perform this kind of forecast, most of the time hired as outsourced services.

The advice of the load forecasting on the planning capacity is usually measured in terms of time resolution, the load forecasting is frequently classified on three categories; namely short, medium and long term forecasting. There is no consensus on the exact definition of the time horizon that covers these categories. However, based on the generalized opinion the time horizons are defined as following is presented. **Short-term load forecasting (STLF)** covers from one hour ahead to weekly forecast. These are often needed by intra-day operations on energy trading, management or power generation.

Medium-term load forecast (MTLF) covers from weeks, seasons with a maximum of one year ahead. This type of forecast is important for annual scheduling productivity and maintenance plans linked to seasonal factors. **Long-term load forecast (LTLF)** deals with horizons over one year ahead, it serves to help on capacity extension plans, long-term financial studies, capital investments and so on. Due to his short range, **Very short term load forecast (VSTLF)** is not useful to planning task, but instead make real time optimization tasks.

The assistance of long-term forecasting on financial decisions is always balanced by external factors that transform the socio-economic conditions such as politics, market behavior, and future uncertainties. The support on the decision making of the deployment, expansion, and enhancement on power system networks are another remarkable use the long-term forecasting.

However, the value of the support decision is subject to the accuracy itself, for that reason accuracy is the most critical feature on load forecasting. A lame forecast mislead customers to take non-efficient decisions with the correspondent financial cost. On the case of network planners and utilities, accuracy is important for distribution systems investments. On the user side, end-users and service providers depend entirely of the accuracy to perform their management strategies.

The accuracy severity on the load forecasting is biased towards the under-prediction energy profile. A negative error could severely affect the production levels at the consumption side, and trigger the expansion plans on delivery networks at short time. For that reason, short-term predictions always are preferred instead of medium-term.

In pursuit of accurate predictions, several methodologies have been emerged to shape load consumption profiles on the user-side and utility-side; being classified in terms of their complexity, flexibility and data requirement. The load forecasting on the user side have a difficulty factor incorporated due to the consumption peaks unforeseeably located. Otherwise, in generator side is possible to estimate a flatter consumption, due to the sum of different users. Statistic techniques used in this area, implement multiple regressions, analysis of time series and state space [6].

However, applications based on non-linear models obtain better results in forecasting, due to their aptitude to shape strong but non-trivial and non-linear relation between future consumption and factors that produce it (climatic conditions, goals of production, labor at calendar, etc.); as well as artificial intelligent (AI) based systems are capable of iteratively adjust their internal parameters until reduce the model error.

Applications based on artificial intelligence methods, neural networks [7] using learning with and without supervision [8]–[13] are example of it. The use of hybrid techniques has spread recently, due to the complexity of the forecast in real cases. This is the case of combination between neural-fuzzy inference systems (ANFIS) [14], pre-filter and optimization components

[15]–[18]. Intelligent methods are often favored for short and medium time load forecasting applications.

Recent authors have proposed an improvement of the accuracy on the load forecasting based on the implementation ensemble models [19]–[22]. They have took the ideas from its success on the weather prediction, which combines specialized models from different phenomena's (wind, temperature, humidity, irradiance, wave tides, pressure) to produce an integrated forecast [23].

Ensemble modelling used the combination of weak learners, usually based on supervised learning models, each one specialize his learning on specific sectors of the target according to their intrinsic particularities or the guided selection made by a human expert. They also can be used to construct lateral models, providing supplementary information to the main modelling structure [24], [25].

Some drawback of this method have been gathered by some authors [21], [26], [27]; the elevated training time cause that high specialized models can't be implemented for very short-term forecasting. The high computational effort to train the models; together with the big amounts of storage necessary to save the models, are other minors drawbacks.

This new trend represents a challenging framework on load forecasting on planning, design, control, and support to decision making on future power systems.

1.3 Problem awareness

Load forecasting plays a corner stone on the decision making processes carried out by energy management systems [28]. The accuracy of the forecast in an essential condition to obtain improvements on cost reduction task such as loads scheduling, co-generation scheduling, and planning energy purchases. Nevertheless, accuracy on load forecasting is a difficult parameter to achieve; mostly because consumptions are influenced by many factors such as weather conditions, holidays, economy status, and idiosyncratic habits of individual customers [7].

In the deregulated power system market, a minimal increase of few units in the prediction accuracy percentage would bring benefits of millions of dollars [29], which makes load forecasting become more important than ever before. Otherwise, inaccurate load forecasts may increase operating costs for the Energy Service Companies. Authors on [30] reported that a one percent error in the cumulative forecasting of the residential electricity demand has resulted on an increase of operating costs by worth of £10 million over the British power system.

On the industrial consumption-side the consequences of a poor accuracy can be listed as, wrong and expensive expansion plans, redundant reserve of electric power or failure in providing sufficient electric power for the manufacturing processes [31].

As we can see, for an energy planner, the **accuracy** feature is a problem that comprises not only an underestimation or overestimation of the load; on this scenario, forecasting techniques with high degree of accuracy need to be developed. Artificial intelligent algorithms presents some superiority on the modelling of non-linear relationships, but possibilities to improve associated drawbacks cannot be ruled out.

Among the most remarkable disadvantages of the AI are the dependence on initial parameters, the limitations of the forecasting due to topologies and the impossibility to extract all the information without human support. Therefore, **there is a need for development of optimal model structures for load forecasting in order to improve the forecast error**.

Based on the previous statements, the main research question of this thesis can be formulated as:

- How an ensemble based load forecasting model be rationally trained using supervised algorithms and variables product of cyclical and multiresolution analysis; and possibly obtain an optimal network structure for short-term **load forecasting**?

1.4 Problem statement

As general opinion, authors [2], [3], [7], [19], [29] define the load forecasting as an easy task when authors search describe the load profile by means of basic patterns; but, in terms of accuracy, forecast becomes a complex exercise due to the idiosyncratic habits of individual customers, the operation regimes of the loads, and the intrinsic uncertainty of environmental variables used to be inputs for models.

In view of electric load profiles are non-linear probabilistic functions with and high stochastic component associated, traditional forecasting methods are simply not suitable for the implementation of high accuracy models due to the lack of nonlinear mapping ability. Although artificial intelligent algorithms had proven be superior candidates for load forecasting compared to traditional techniques, the research and design of optimal network structures has not yet fully researched.

On the other hand, an enhancement on the accuracy and performance of load forecasting systems require optimal network structures and the complete exploitation of the information hidden on the learning data.

Problem statement: To develop an optimized ensemble-based models for medium and short-term Load Forecasting; to apply these models to a real life case of study to evaluate the performance of the proposed approach providing as result prediction with horizons of one day, one week, and one moth ahead.

1.5 Research aim and objectives

The scope of the research is develop novel **ensemble structures for STLF & MTLF**, perform network assessments on the basis of the forecast quality, and network performance using different algorithms approaches.

The models are trained using as input drivers a collection of preprocessed exogenous variables extracted from time and environmental information; all of them together with endogenous variables extracted from multiresolution, cyclical and entropy analysis.

On the other hand, **base learners** have been implemented on the basis of machine learning algorithms such as on neural networks, evolvable networks, expert systems, support vector machines, and trees topologies. The options for increase the forecasting quality, through the minimization of the forecasting error and the exploitation of hidden insights and miscellaneous properties of the training data, are equally explored in the form of clusters of sub-specialized base learners inside of the ensemble structure.

The algorithms and methodologies early mentioned have been selected in pursuit of some modelling features such as **Accuracy, Adaptability, and Autonomy**. Preprocessing and the knowledge discover algorithms are implemented in order to boost the **accuracy** trough cleaning of variables, and to enhance the autonomy of the modelling algorithm via non-supervised intelligent algorithms respectively.

The **Adaptability** feature has been reached using a base of three components. **Base learners** trained using resampling techniques [32], which ensures the replication of the global probability distribution on multiple independent training sub-sets and consequently the training of base learners on representatives spaces of occurrences; building general models that produce accurate predictions on new cases restrained by the global probability distribution.

Multiresolution and cyclical analysis: Through the decomposition of endogenous variables on their time-frequency components, major insights are acquired and applied on the definition of the ensemble structure layout. **Self-organized modelling algorithms** such as Cartesian genetic programing based algorithms, used as provider of customized weak learner's networks, are the last component on to the model adaptation over the training data.

The **Autonomy** feature is reached by the use of automatic procedures and experimental criteria's in order to minimize the interaction of an expert user on the forecasting procedure.

Aim: To develop novel ensemble models architectures for short & medium term load forecasting (STLF & MTLF), reinforcing and enhancing features such as accuracy, autonomy, and adaptability of the modelling algorithm through a solid methodology; to evaluate the performance of these models on the user-side in order to predict the load profiles one day, and one week in advance.

In order to accomplish this aim, the following objectives are intended to achieve:

1.5.1 Technical & Methodological objectives

- To review the state of art on load forecasting in order to discover variables that plays an important role on prediction of the electricity consumption at short, medium, and long term; along with the methods to discover, measure and rank those variables from the raw data.
- To make a thorough study of the state of the art of load forecasting to find out which modelling algorithms group has the greater number of benefits.
- To gather historical databases that includes the load consumption for a set of consumers with theirs exogenous variables (weather information), grouped in accordance of their user-side sector (state consumption, industrial consumer, commercial/residential buildings, house holding).
- To develop a strategy for detect and eliminate/replace the corrupted data, due to the possible existence of bad data in the load profiles as well as in the weather data can‘t unfortunately be discarded.
- To verify the modelling features former defined: accuracy, adaptability, and autonomy

1.5.2 Scientific objectives

- To develop novels load-forecasting structures based on the mixture of experts trough ensemble architectures and evaluate their performance using standard error measures, normality tests and other supplementary performance measure functions.
- Performs an exhaustive measure of the uncertainty associated with the forecasting error in terms of modelling approach and forecasting horizons selected.
- To apply resampling techniques in order to create independent sub-sets of training data; maximizing the replication of the global probability distribution into small sub-spaces. This provides a wide data specialization due to the training of the base learners on representative’s spaces of occurrences, and builds an ensemble model that produce accurate predictions with a low generalization error restrained by the global probability distribution.
- To define and measure the suitability of novel ensemble structure layouts, designed based on the insight of the multiresolution and cyclical analysis of the endogenous variables.
- To integrate non-supervised knowledge discovery algorithms in order to produces an effective segmentation of the data, helping to small-groups of base learners to obtain a regional specialization based on the clustering the load profile.
- To record the approaches presented on this thesis on a readable and easygoing way, in order to allow to neophyte users the replication of the models.

1.6 Hypothesis

During the presentation of this thesis, author will test the followings hypothesis ranging from general to specific.

- Ensemble modelling architectures can be implemented for STLF & MTLF with the added benefit of an improvement on the accuracy, adaptability, and autonomy of the forecasting system by means of artificial intelligence algorithms; tested over load profiles extracted from different consumption scenarios.
- Accuracy of STLF & MTLF can be improved by the use of base learners based on artificial intelligence algorithms, and structurally oriented in base of multiresolution and cyclical analysis.
- The Integration of non-supervised knowledge discovery algorithms produces a segmentation of the data, which helps on the increase of accuracy.
- The integration of statistical analysis, preprocessing techniques, and knowledge discovery algorithms allows extracting information to support the autonomy of the forecasting system. In parallel, it also helps to mitigate the uncertainty components carried by the drivers and source of inaccurate forecasting's.
- The adaptability of the load forecasting algorithm can be achieved by means of a forecasting methodology based on three components; usage of base learners trained using resampling techniques; usage of multiresolution and cyclical analysis over endogenous variables in order to construct a structural adapted model; implementation of self-organized modelling algorithms as provider of customized weak learner's networks.
- An automatic and non-supervised load forecasting algorithm, able to create in an exhaustive way accurate models tested over different consumption scenarios, can be a push on the state of the art for Energy Management Systems.

1.7 Research scope

This research work presents the novel implementation of **load forecasting algorithms on the user side**. The forecasting horizons cover on this research includes short-term and medium term load forecasting. Long-term horizon are not covered on this document due to the lack of technological importance for comitial applications rather than long term planning purposes, but this does not mean that methods described can't be equally useful for this horizon. On the other hand, very short-term load forecasting present a challenge for the training time of the algorithms only solved using large computational resources.

The forecasts are generated by Ensemble models (EM) using a mixture of novel structures, algorithms and platforms, i.e. ensemble model of cascade feed forward networks using seasonal components developed in MATLAB environment; or an ensemble model of neural network Cartesian genetic programming using cyclical and multi-resolution analysis developed in C++/MatLab environment.

Once time the models architectures are designed, they are trained with the historical data obtained from user such as buildings, industries and regional aggregation of consumers. The comparisons between forecast and real data are done merely as case study to validate the approach. The Figure 1 below attempts to clarify the focus of this research.

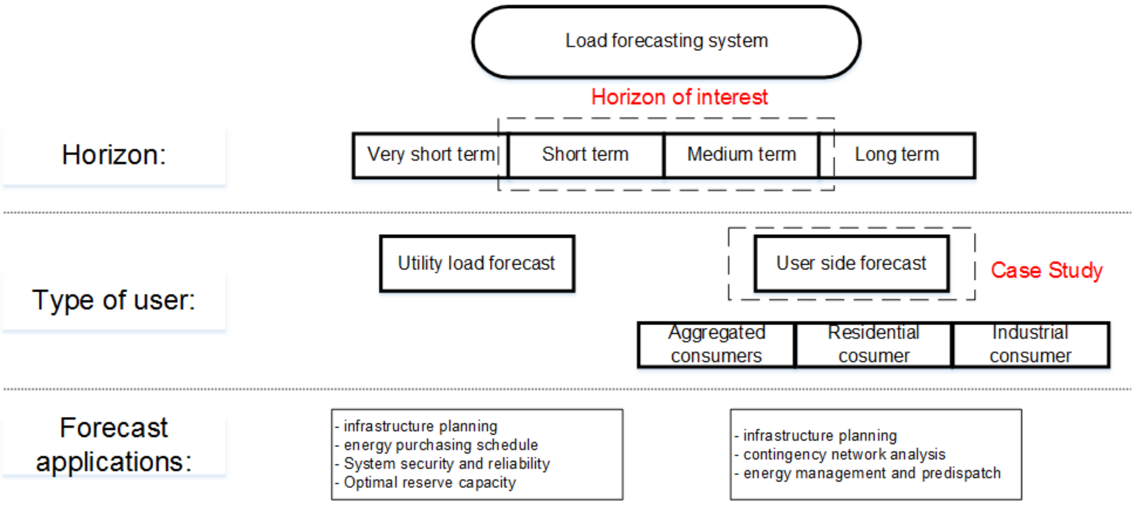


Figure 1. Types of load forecasting and focus of the research.

1.8 Project motivation

Load forecasting of energy profiles contains a very volatile, uncertainly located, and hardly predictable component. Most of the times, it is related with the idiosyncrasy of the human behavior or the environmental conditions, and always it suffer an increment with the expansion of the forecasting horizon.

Nevertheless, not only the characteristics of the target to analyze impact on the accuracy of load forecasting systems; the model limitation to simulate the load profile and extrapolate their behavior time ahead; the preprocessing of the variables; and knowledge discovery methods are another circumstances that can increase or reduce the performance of the forecasting system.

Solve the accuracy problem is the major subject of this research because load forecasting is the corner stone on energy industry. It assists to utilities on the planning network expansions, to calculate the optimal point of the reserve capacity, on the system security and reliability planning, on the energy purchasing, etc. On the user side, accuracy can affect also on the cost reduction task such as loads scheduling, co-generation scheduling, energy purchases, and investing criterion purposes.

Furthermore, this work presents an approximation to the user side load forecasting problem due to its high business profitability, on the current state of the energy management systems markets, together with the plenty scope for accuracy improvements to be applied on it.

Also, continuous supervision compared to load forecasting helps to detect malfunctions of equipment, or missed set points, thus improving the energy efficiency of the system. Figure 2 attempts to clear the case of study of this research, the first dimension deals with the forecasting horizon, and second one involves the user side as principal subject of the forecast.

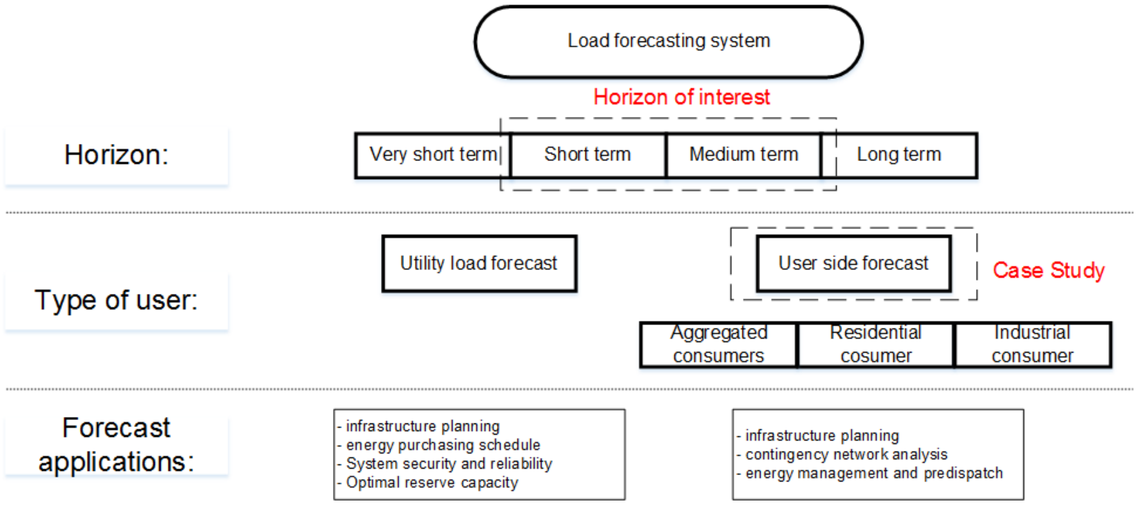


Figure 2. Load forecasting outline and some applications.

1.9 Thesis outline

This thesis document have been developed in order to introduce gradually the reader on the problematic of the energy management, highlighting the importance of the load forecasting among all the services running on an energy management system. Then, the elements and procedures carried out on the load forecasting process are analyzed based on real data sets.

This leads to the introduction of the customization process that must be carried out by human experts during the installation of an energy management system. Later, the modelling techniques necessary to obtain forecasting models are introduced, leaving space to the description of the novelties introduced by this thesis on the field of the load forecasting modelling.

The previous paragraphs corresponds to a general description of the thesis contents, on the other hand we will proceed to be more specific about the contents per chapter on the following paragraphs. **Chapter 2** presents the state of the art of the load forecasting on power systems. It start with a general description of the energy management systems, making a special emphasis on the forecasting system and the load modelling process.

It continues with a detailed description about the elements that affect the short time load forecast for a given electric consumption, and more important the state of the art of the load forecasting techniques. Based on this stat of the art the **Chapter 3** is presented, it comprises all the contributions implemented in order to fulfill the thesis objectives.

The chapter starts with the description of a load modelling algorithm based on expert systems and multi-resolution analysis, continues with the description of a load forecasting algorithm based on evolutionary modelling, and finalize with an ensemble learning strategy which embrace and satisfy the objectives of fast execution, accuracy, adaptability and Semi-automated deploy.

On **Chapter 4**, the conclusions about the load forecasting algorithms presented for a short time forecasting are presented per each objective stated on the thesis objectives, praising the goodness of our modelling strategies. **Chapter 5** presents the dissemination of the thesis results in form of scientific dissemination or participation on technology transfer projects.

Finally, **Appendix** section introduces the helpful information to understand the estimation of the errors on load forecasting, a detailed multivariable analysis of the experimental databases implemented to test our hypothesis, and a socio economical description of the impact of energy management in the European and Spanish markets.

CHAPTER

Load Forecasting on Power Systems

This chapter presents the state of the art of the load forecasting techniques, including the concepts related with them. The reader will be introduced on the energy management systems, known as the pillar of the energy management revolution; the definitions, processes and functions that make operative the EMS's, and the central pillar of all the energy savings actions "the modelling and forecasting system engine."

Contents

2.1 Introduction

On the chapter 1, the reasons and objectives to carry out a doctoral investigation around the load forecasting systems are presented. Thus far, we introduced the problem statement on an economical and scientific context resulting on a delimitation of the research plan to critical horizon of predictions.

In that order of ideas, this chapter introduces the reader on the areas where the load forecasting techniques are widely used, as well as his economic impact as element of the EMS's. Furthermore, this section describes the EMS's, the systems in charge of analyze, control, and optimize the energy consumption across different consumption levels.

Later, is introduced a complete description of the characteristics of the load profile, their origin, challenges and the objectives of the forecasting process. These concepts cause a classification on the implementation of the forecasting system, which leads to implementations specialized on the forecasting horizon, or the field of application. Finally, the characteristics of the ultimate forecasting system are depicted as a guide for future commercial implementations at the reader's discretion.

On the fourth section, the preprocessing techniques executed as previous step of the load forecasting are introduced. This section starts with a description of the driver's frequently employed on the modelling task; including the importance of the exogenous variables, and the endogenous variables and their generation. On the same section, interesting cyclical behaviors allocated on the load profile are described.

The section continues with the description of the preprocessing techniques implements with the aim of clean the signals that participate on the modelling process. Furthermore, knowledge discover and input selection techniques are introduced as helpers on the antecedent steps of the modelling.

Fifth section presents the state of the art of the load forecasting techniques from simple's regression methods to computational intelligence and machine learning approaches. This section also made an especial remark on the problems and limitations related to the previous techniques as an introduction to ensemble forecasting methods.

2.2 Energy Management Systems

One of the most spread and well-known technology, which can help on the energy delivering and consumption issues, is the energy management systems (EMS). EMS allows collect, analyze, and share critical information to understand, control, and optimize the energy consumption across different consumption levels. As example, EMS can help to consumers to flat the peak consumption hours avoiding high prices and keeping on a database a history of the actions carry out used for future references.

The research about EMS and its principal characteristics has been published since three decades and the latest advances bring to it specialized instrumentation for monitoring and a user-friendly configuration adequate by the software resellers. EMS lay on emerging technological advances and trends including:

- Non-intrusive load monitoring techniques applied to gather data from the energy consumption at any level of disaggregation or consumer.
- The availability and diversification of sensors, it allows gather several environmental variables that can be integrated in to EMS to increase the effectiveness and exploit the maximum information.
- The big data systems in charge to manipulate enormous quantities of data, storing, sorting, analyzing and visualize in a meaningful way to consumers.
- Cloud computing, which made possible to perform large-scale analytics on disaggregated energy data and offer real-time reports to operators without.
- The increased accuracy of the artificial intelligence algorithms applied to knowledge discovery and evidence-based learning. It support to the modelling, supervision and optimization of the demand to perform savings with a minimum of human intervention.
- The evolve of the market landscape on trends such as corporate awareness of the Internet of Things (IoT) and demand for data-driven decision support tools facilitate the adoption of EMS's.

In terms of the application sector, EMS can be classified as Building Energy Management Systems (BEMS) or Enterprise Energy Management Systems (EEMS). BEMS are generally referred to as Building Automatization Systems (BAS). By setting a goal on the operational performance of the energy facility while ensuring the comfort and safety of the occupants, BEMS help to reduce the operational cost over the life cycle of the facility.

BEMS realize functions at upper level, which are advance monitoring which provides data about the consumption pattern, extremely useful in intelligent decision making related with energy use and the smart control of the loads at the facility [33]–[38]. BEMS can support modularity and inter-operability; visualization and reporting; fault detection and diagnostics; predictive maintenance and continuous improvement; and their decision-making intelligence can integrate algorithms for dynamic control and system optimization.

Integration of more sophisticated tools and programming techniques on BEMS's, allows the implementation of controls to maximize optimally the energy conservation. Measures to manage mechanical, electrical, and plumbing systems; for example: installing carbon dioxide sensors; varying air handler fan speed; checking speed of circulating hot water and condensed water pumps; controlling enthalpy economizers; controlling intensity of light and many others.

The global market for BEMSs continues to grow with the maturity technologies and financially motivated high potential customers. Due to the cost of control devices is decreasing, and monitoring and control systems generate complementary strategic benefits such as greenhouse gas reductions and sustainability improvement, BEMSs are becoming more cost-effective options for a broader set of customers.

According to Navigant Research [39], the global BEMS market is expected to reach $2.4 billion in 2015 and grow to $10.8 billion by 2024. The Navigant Research report assesses the global market for BEMSs, including the software, services, and hardware components. On the other hand, according to the Mordor intelligence [40], The European Building Energy Management Systems market revenue is estimated to grow with a compound annual growth rate of 22.48 percent from 2014 to 2020 to reach at US$9.50 billion in 2020.

At the European Union, The EMS industry has been largely fueled by the use of smart grid services, industrial competition and the political framework of incentives in energy efficiency as far was commented previously [41], [42]. The Mordor intelligence report has segmented BEMS market by Software (Data Management, Asset Performance Optimization, Application Platform, HVAC system and Lighting system), by Technology (Wired and Wireless), by Services (Consulting & Training and Support & Maintenance Services) , by Industry (Manufacturing, Telecom & IT, Office & Commercial Buildings, Municipal, University, School & Hospital (MUSH) systems and Government) and by Countries.

Enterprises EMSs provides a complete and disaggregated view of the facility's operations. Unlike BEMSs, EEMSs is defined by software and services that support holistic energy management within an industrial facility or across an enterprise to achieve efficiency, cost savings, sustainability, and climate change targets while maintaining the optimal operational parameters for the production processes.

Based on the penetration level of the EEMS capabilities with the facilities infrastructure, the EEMS is able reproduce a strategic energy management at several levels. On the **basic level**, **monitoring and report generation** provide key performance indicators and help to track the energy performance goals by energy project participants.

On the **medium level**, the EEMS strategy support **modeling and forecasting** of the demand and generation profiles; benchmarking against historical consumption; cost analysis of energy use; and measurement and verification. At **high level**, the strategy support actions such as **dynamic control of the loads and system optimization; fault detection and diagnostics; productive planning; predictive maintenance and continuous improvement**.

The EMS strategies bring to organizations a comprehensive understanding of historical energy performance, planning and cost-effective selection over energy conservation measures, performance tracking of implemented measures, and saving verification.

On general terms, several authors [43]–[45] agreed to divide the EEMS on:

- **EEMS for industrial and commercial consumers**. An EEMS system can give energy consumers the ability to take ownership of the procurement and consumption of electrical energy. The system can help identify accurately enterprise-wide energy needs by aggregating and profiling usage patterns and by helping perform variable analysis against utility rate choices. It can help procure energy effectively, verify billing, and allocate energy costs to tenants, clients, departments, or processes.
- **EEMS for energy services**. EEMS give Energy Service Providers (ESPs) an economical and feature-rich way to offer competitive value added reporting, performance contracting, and consulting services to large numbers of customers, with multiple facilities spread across wide geographical areas. Intelligent devices can be located at the customer's service entrance and within their facilities, with head-end software at the offices of the ESPs.
- **EEMS for grid enterprises and utilities**. An EEMS enables demand response or load curtailment programs for ISO and utility enterprises by providing the high-speed communications necessary to efficiently contact customers, control distributed generators or loads, and verify operations. It also automatically acquires the energy logs from each location to support settlement and billing.

Today, EEMS adoption is increasing worldwide because of current industrial market dynamics, including customer demands for solutions that help them hedge risks and take advantage of opportunities. According to Navigant Research [46], global IEMS revenue is expected to grow from $13.5 billion in 2015 to $35.6 billion in 2024.

2.2.1 Energy Management System Features

On the practice, exist clear differences amongst the types of EMS that exist. Power generating companies have very complex needs to monitor and control energy conversion processes for the large amount of equipment and devices that exist in these systems. On the other hand, delivering and services companies need detailed controls to optimize the power flowing through the grid.

Regarding to end users, the EMS is able to diversify and scale their operations and characteristics in order to manage the energy needs and unique demand profiles on the user side. Different authors define the next set of features as the most relevant on the conception of an EMS [47]–[49].

- **Monitoring.** Systems must provide energy consumption information at various temporal frequencies such as 15 min., hourly, daily, and weekly. The feedback is most successful when it is provided frequently and over a long period.
- **Disaggregation**. systems must provide a perception about the energy consumed by individual appliances. The disaggregated data also highlights the impact of long-term changes such as switching to an energy-efficient appliance..
- **Availability and accessibility**. systems must make the information available to the consumer at all times through an easy-to-use interface, either in the form of a physical device, or through a web or mobile portal. EMS may also use push technology to send urgent notifications to consumers.
- **Information integration**. EMS must also integrate other types of information such as indoor temperature, humidity, acoustics, and light; and consumers historical data, usage data related to different appliances, as well as peers consumption data.
- **Affordability**. systems should allow easy installation without professional help. Its configuration and maintenance should be simple. It should consume minimal energy with a low running cost. These factors help reduce the entry barrier of the system and facilitate widespread adoption.
- **Control**. systems should be able to provide remote, programmable, and automatic control of devices. Generally, the consumer is expected to perform necessary control operations manually. However, a digital control option or automated actions are more effective.
- **Cyber-security and privacy**. systems must authenticate all transactions to ensure that consumers data and control operations are secure, and not accessible to third parties without explicit consent.
- **Intelligence and analytics**. **A desirable feature in new generation of EMS is the expert use of information**. Consumers often lack a deep understanding of electrical systems and have limited time to make energy-related decisions. Thus, it is desirable to have the system perform intelligent actions that balance energy consumption and consumer comfort. Those actions requires techniques from machine learning, human–computer interaction, and “big data” analytics to discern usage patterns and predictive actions. The following list collect the new trends of the computational intelligence on the energy management, these are equally presented graphically on the **Figure 3**:
    - **Load modelling and forecasting on the user side**, this feature is the central pillar of the intelligence energy management process. The consumption profiles of the loads, single or aggregated, can be modelled using technical specifications of loads and its programmed use; this approach is called parametric modelling.
    - Alternatively, the modelling of the load profiles based on historic data increases the automatization of the system and reduces the human intervention; this approach is often called data-driven modelling.

- In addition, a data-driven approach allow to benchmark the model against historical consumption models, measure their accuracy and verify the normal condition of the load.
- **Fault detection and diagnostics**, EMS can measure and verify the state of the loads using the registers gathered through its continued monitoring. Analyzing possible deviation on the load profiles respect to its historical behavior is possible detect failures or associate a probability to them.
- **Dynamic control of the loads and system optimization**, in order to find a solution that optimizing costs and minimize the risk of loss a load by over/under frequency operations; EMS should optimize the load operation based on the stochastic probability of several conditions (load schedule, forecasted production from renewable generators, prices frames).
- **Productive planning**, EMS can schedule an optimized planning according to cost-productive boundaries obtained from the optimization.
- **Predictive maintenance**, EMS can provide a planning of maintenance combining the information of the fault detection, the diagnosis results, and the optimized schedule of operation. This guarantee a cost-efficient planning over the use of the loads.

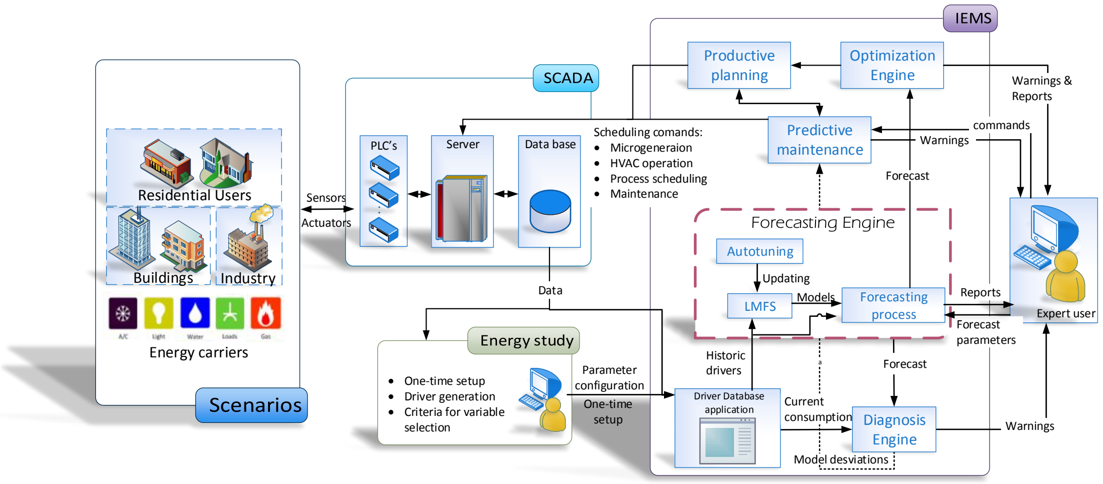


Figure 3. Modern IEMS diagram, includes the new trends of the computational intelligence on energy management.

It is clear that EMS research shows to be a trending topic in the scientific community and the industrial world. This fact is supported on the economic implication of the energy management reported on [39], [41], [42], [46] and the one-thousand publications yearly registered by the Institute of Electrical and Electronics Engineers. Although the scientific effort is being focused on tackle partial energy management problems, **the scientific community has identified the modelling and optimization as the most important challenges to allow a next generation of EMS** [33], [47], [49]–[55].

2.2.2 The Modelling and forecasting engine

Since 70's, techniques for energy consumption prediction and load forecasting are being applied to user-side including tertiary buildings and SMEs and large companies [33]. Initially, these have been used in order to build models of the energy usage through the gathering of information about energy consumption and to plot this against some variables (such as degree-days or production activity). These simple correlation methods where enough efficient to find the primary drivers among the consumptions.

Still today, these basic principles allow to human-supported systems find the most accurate technique and model on each scenario, let's check some of the basic principles: on buildings, is normally found direct relationships between the energy consumed by a building and degree-day measurements. On production processes, where energy use is largely determined by the physics of the process, there is normally a direct relationship between the energy consumed and production volume. On household applications the energy is largely drained from HVAC and white goods, who follows the human patters governed by temperature or holydays.

These simple relationships were used as base information to provide a model of consumptions by simple interpolation of acquired data [43]. As a consequence, on large implementations of EMS, The forecasting system allows the benchmarking as a straightforward method for comparing the energy consumption of different buildings or equipment against each other as a way to determine why some of the buildings are more efficient, which ultimately results in action to increase the efficiency in the under-performing building.

In terms of importance, the novel implementations of the forecasting systems have reach the electricity markets. It has penetrated in a large energy intensive enterprise in order to aim the purchase/selling operations on the electricity trading operators. In addition, Load Modelling Forecasting (LMF) has been supporting different areas of energy management, for example, for utilities in generation, transmission, and distribution of electricity; in energy markets for price forecasting; and in buildings for HVAC control and optimization.

In industries, the load modelling forecasting is important in the support of the decision making process. For example, to address the problems of economic scheduling of generating capacity, scheduling of fuel purchases, security assessment and planning for energy transactions. Energy security and stability rely on accurate planning of these items.

In utilities, Short-term load forecasting (STLF) is the most important type of forecasting. Commonly, the STLF methodology is used for the calculation of an energy forecast 24 hours ahead with time series sampled between 15 minutes to 1 hour. Another important kind of forecasting is the peak power load forecasting of a few days ahead, it is used as an operation index for unit commitment and scheduling.

In general for any energy management system, the load forecasting is greatly important because it provides the basis for the control and optimization of the loads. For example, when calculating the energy forecast demand for a day ahead, the BEMS can schedule the demand of the heating and ventilation equipment taking advantage of the slow thermal behavior of the buildings. Its importance has been such that there was a society founded in 1894 dedicated to HVAC systems for buildings called **ASHRAE**. This society, among its many lines of action, supports research on LMF systems for energy consumption in buildings. The potential of energy modelling in buildings is also presented in [38], especially in HVAC systems forecasting. It is not mentioned that in the industrial sector, the potential could be larger since the possibility of process and machine control exists.

As consequence, energy forecasting has attracted great attention in power system research. Alfares et al. [56] divided the existing energy consumption forecasting methods into two categories and nine sub-categories. Aggarwal [57] agreed with the upper categorization namely hard and soft computing techniques. Hard computer includes: multiple linear regression, exponential smoothing, iterative reweighted least-squares, mixed models, and autoregressive moving average with exogenous variable.

Soft computing includes approaches as: Bayesians nets, models based on genetic algorithms, fuzzy logic, fuzzy neural networks, neural networks, and expert systems. One of the most accurate methods was those based on an artificial neural network (ANN) technique. However, in scientific work, concerning to soft computing approaches, there is a lack of appropriate methods that allows the determination of the optimal structures on each scenario.

Indeed, there is a trend to use computational intelligent tools for load modelling forecasting. Neural networks, neuro-fuzzy networks and evolutionary algorithms are noteworthy of mention. This is mainly due to the capacities of the computational intelligence algorithms to model non-lineal behaviors, such as electrical consumption. In utilities, as well as in BEMS, the computational intelligence tools are well accepted for load modelling forecasting and for hybrid algorithms with computational intelligence and statistical algorithms.

On the user-side, particularly for industrial users, the LMF of energy consumption has also acquire high importance as in other areas. LMF has been used to support decision-making, looking to take advantage of available energy smart-meters database by means of data mining [58]. These applications can forecast the energy profiles based on some energy drivers, such as the daily total production, the work hours, and the running time of equipment in the plant. However, basics applications use the endogenous variables created from the overall consumption of the plant.

Hybrid proposals such as modelling combined with pre-processing methods, or evolutionary approaches in order to fix the internal parameters of models have been widely documented. Principal component analysis (PCA) mixed with support vector machine (SVM) for long-term energy consumption forecasting [59], support vector regression (SVR) for electric forecasting in combination with seasonal filtering [60], particle swarm optimization and adaptive-network based fuzzy inference system (ANFIS) for the energy consumption forecasting in Spain [61], or improved models by feature selection based on correlation analysis [62].

In this way, load forecasting systems applied to industrial users has adopt Computer Intelligence (CI) technologies used to enhance the efficiency on their plants. CI tools have great flexibility, proof of that is their capacity to be mixed with other CI tools, statistical or signal processing functions. This mix can be done in different levels and schemes. For example, we can mix different structures as neural networks and fuzzy logic, or evolutionary algorithms to train adaptive network structures as NN, NFN, ANFIS and get evolutive trained structures.

Furthermore, because of the cyclical behavior of demand load profiles, statistical and signal processing functions, or operators can be useful to highlight this kind of cyclical behavior and improve the results. Examples of these functions are wavelet transformations, Kalman filters, correlation functions, etc. Therefore, proposals based on hybrid algorithms by joining CI with statistical, signal processing functions or operators have presented the best results in the resent years [63]–[65].

2.2.2.1 Definition of the forecasting engine

The forecasting engine is in charge of the modelling, and consequent forecast, of the energy consumption on the scenario analyzed. It considers a mixture of external and internal parameters to the process that can affect the behavior of the load´s operation (i.e., weather data, working days, production process, etc.). These parameters are frequently addressed as energy.

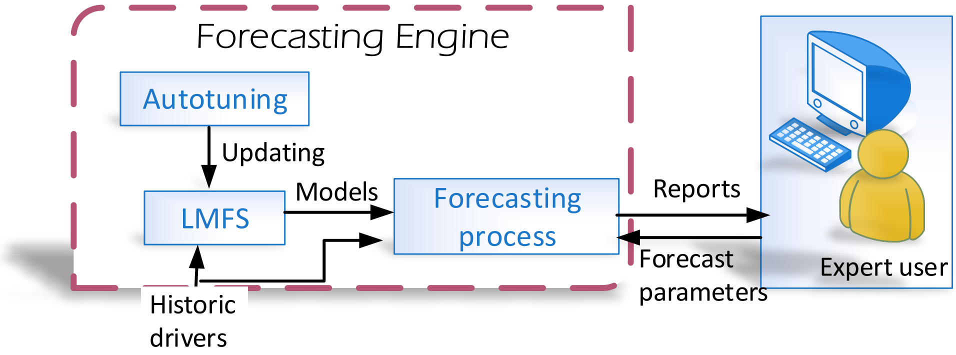


Figure 4. Forecasting engine

As is shown in the previous figure, the forecasting unit is divided into three principal parts: the load modeling and forecasting system (LMFS), the auto tuning process and finally the forecasting process.

2.2.2.1.1 *Load modeling and forecasting system (LMFS)*

The load modeling and forecasting system is in charge of request the required historical data of the energy drivers, and uses them to generate the mathematical models of the energy consumption. In order to generate them, the system has to process and analyze the data in order to find and describe the relationship between the consumption profile and the energy drivers. The energy database could include:

- The historical power consumption by load or by set of loads.

- The list of parameters that may affect the load profile.

2.2.2.1.2 *Forecasting process*

The forecasting process uses a supervised training algorithm which can acquire information and learn from the historical database. In order to complete a forecast load profile the following two main steps are required.

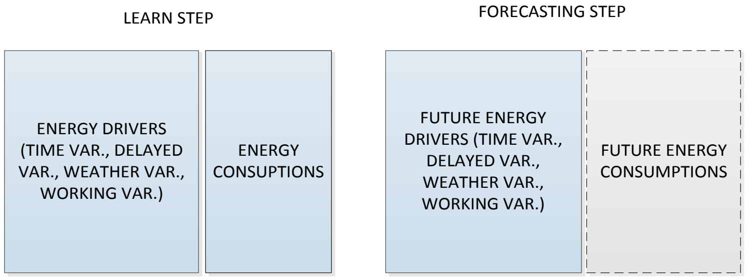

LEARN STEP
FORECASTING STEP
ENERGY DRIVERS (TIME VAR., DELAYED VAR., WEATHER VAR., WORKING VAR.)
ENERGY CONSUPTIONS
FUTURE ENERGY DRIVERS (TIME VAR., DELAYED VAR., WEATHER VAR., WORKING VAR.)
FUTURE ENERGY CONSUMPTIONS

Figure 5. Learn and forecasting steps.

During the first step, the algorithm, by means of its training method, finds and learns the different relationships between the energy drivers and the energy consumptions. The result of this process is a nonlinear model of multiple inputs and a single output.

In the second step, the model is evaluated with inputs independent of those previously used and the short-term energy forecast is calculated.

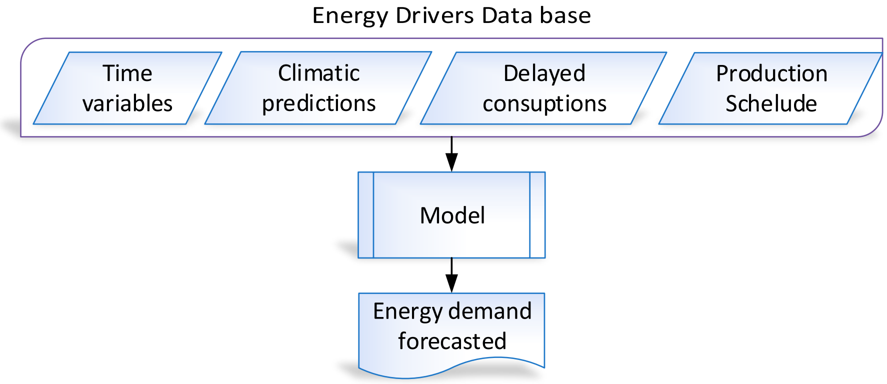


Figure 6. Flowchart of forecasting process.

To generate short-term energy forecasting, it is necessary to know the future values of the particular energy drivers that have been used during the training process. It is possible to obtain this information by using databases of that includes exogenous variables (climatic predictions, time variables, working schedules), and endogenous variables (delayed energy consumption, derivatives of weather signals). All of these parameters are stored in the energy driver database.

The time resolution and the forecasting horizon depend on the EMS features and function requirements. The EMS functions or upper level applications are the "users" of the forecasts and these forecasts are their initial input information which they use to carry out their tasks.

For example, the result of the forecasting process is used in the diagnosis unit to detect malfunctions or anomalies, in the optimization unit for taking decisions or adjustment of equipment or on energy prediction calculations for general reports.

2.2.2.1.3 *Auto tuning*

A second concept that is important in the EMS is the online modeling or also called "auto tuning" [66], [67]. It gives the LMSF the ability to closely match the operating conditions to the model. That means that the model can detect when there are unusual weather conditions, energy consumption averages in workshops, or other conditions and take them into account for the future predictions.

The auto tuning operation is a periodic execution of the LMFS using collected data. Its process is regularly executed (e.g., every two months), updating the energy driver database and updating the mathematical models of the energy consumption. As a minimum, the database has to include an appropriated amount of collected data to run the modeling (e.g., two months would be enough for a first modeling).

The results of the auto tuning consist of a fine adjustment of the model to predict the future load´s energy consumption with the right accuracy; meanwhile the energy database keeps growing.

The accuracy improvement of the modelling using auto tuning is mostly based on adaptability of the model structures to changes inside of energy drivers. For example, ANFIS or ANN as data driven models, are excellent candidates as basic model structures, due to theirs extraordinary adaptability to uncertainly events. Later in the document, these type of model structures are being explained in depth.

2.3 Classification of the load forecasting

THe forecast horizon is defined as the number of periods between today and the date that we want forecast. It could be annual, quarterly, monthly, weekly, hourly, etc. the horizon is important for at least two reasons. First, the forecast change with the forecast horizon what means that the accuracy is lost and the prediction intervals became width. Second, the best forecasting model will often change with the forecasting horizon as well (Update) [68].

On all the cases the forecast is limited to the quality and the quantity of information available when forecast are made. Sometimes, we use the **univariate** information set, composed by the endogenous variables including the present. Alternatively, we use the **multivariate** information set composed by the endogenous and exogenous variables.

Those ideas are fundamentals for the evaluation of a forecast, because we are sometimes interested in whether the forecast could be improved by using a given set of information more efficiently, or add more information to the current set. Furthermore, the selection of the forecasting tools is determined by the specifics of the situation will indicate the desirability of a specific method or modelling strategy.

The reader might guess that the complexity of this models is often associate with the complexity of the real phenomena, but far from be true, decades of literature have proven just the opposite. Simple parsimonious models tend to be the best for out-of-sample forecasting in complex matter such as finance and economics.

This originated the **parsimonious principle**, which present simple models with characteristics such as a better generalization, easily interpretation, and intuitive feel of their operation. Enforcing their simplicity with data mining techniques maximizes its fit with the historical data. Also, restrictions over the forecasting model known as **Shrinkage principle** helps to enhance the accuracy of the models by means of making them sharp under specific conditions. The models contained on this thesis try to follows the last principles and retain the **KISS principle**, keeping sophisticatedly simple all the approach presented.

All of the previous paragraphs have a close relation with the forecasting horizon. Because, they show clear clues to select the set of conditions or parameters necessary for each kind of horizon. For example, a very short term forecasting running in the matter of minutes, need to use the last samples of the target forecasted to produce the next points. Meanwhile, on a long term forecast we prefers extract the cyclical behavior of the exogenous variables.

2.3.1 Forecasting time ahead windows, which one is more suitable for each case?

The multitude of modelling methods could confuse to readers on their application and suitability depending of the given load scenario and forecast horizon required. In this section we will try to present some practical knowledge to determine the best modelling strategy.

2.3.1.1 Very short-term load forecasting (VSTLF)

There is no officially accepted definition for VSTLF, but literature has used the term to indicate load forecasting from one minute to half hour lead time. It principal function is provide a generation target for economic dispatch and load frequency control.

The VSTLF is closely related with the rise of the Smart Grid and Microgrid concepts, on the management of large operations at real-time a very short term forecast is required. This is mostly because the stochastic nature of renewable energy sources such as photovoltaic (PV) panels and wind farms, and the multiplicity of load profiles combined trough load-aggregation.

The techniques employed on this area comes from polynomial regressions, ARMA models, modified filters, space-state models, fuzzy logic, autoregressive neural networks [69]. But, experience has demonstrated that VSTLF method's accuracy is extremely variable, and basically governed by the statistical characteristics of the system it is applied to. This concludes that accuracy and tolerance of VSTLF system must be adapted according to the complexity of the model.

2.3.1.2 Short-term load forecasting (STLF)

Short-term load forecasting (STLF) covers from one hour ahead to weekly forecast and his quality impact has a significant importance on the operation efficiency of any electrical utility. This forecast support many operational decisions on the generation side such as economic scheduling for near-future trading, calculation of the generation capacity and availability of resources for energy request, scheduling of fuel purchase, security assessment of the production margins, and frequency control.

On the transmission grid, the STLF had a major impact on the restriction of the climate effects over the energy transactions, planning, and vertical integration of the services given by the Trading system operator (TSO) and data hub. It is often difficult to predict and can vary significantly even over a short period. Climate also varies in time: seasonally, annually and on a decadal basis.

On distribution grid, it serves to diverse functions such as peak demand identification, assessment of the load variability, confidence margins of distribution, frequency control for large consumers. In all the previous cases, STLF have been cataloged as the key factor on the enhancing of the energy management. This is because the weather variability and the corresponding energy consumption reach the maximum level of uncertainty in this short-term period of time.

In comparison, VSTLF is only limited to forecast period where the derivative of the profile are steady state or nearly zero. On the other hand, long-term forecast are mostly governed by activities scheduled on base of previous expansion plans or long-term economics. This fact and the cyclical behavior of the weather make easy stablish error boundaries an uncertainty.

2.3.1.3 Medium-term load forecasts (MTLF)

MTLF covers from weeks to a maximum of one year ahead. This type of forecast is important for annual scheduling productivity and maintenance plans linked to seasonal factors. Medium-term load forecasts enables companies to estimate the load demand for a longer time interval which helps them for example in the negotiation of contracts with distribution companies or service operators.

It application also covers the predictive maintenance due to most of the failures show their patterns on periods lower than one year. The MTLF mostly aims for the decrease of the general error thought an over fitting on the bias, this means that they prefers the follow the curve over the time period instead of fine details as peak of consumption. The input values for this prediction usually incorporate additional influences like demographic and economic factors.

2.3.1.4 Long-term load forecast (LTLF)

The LTLF deals with horizons over one year ahead, is usually found forecast of 20 years ahead. It serves to help on capacity extension plans, long-term financial studies, and capital investments. For these methods the population growth and gross domestic products have to be considered. Due to his long range, the differences in forecast comes visible and have consequences for the models and methods applied and for the input data available and selected.

The information of the section is summarized in the **Table 1**. Moreover, information about the use of the forecasting horizon is added on the APPENDIX section.

Table 1. Classification of load forecast.

	Weather variables	Economic variables	Updating cycle	Horizon
VSTLF	Optional	Optional	<= 1 hour	1 day
STLF	Required	Optional	1 day	2 weeks
MTLF	Simulated	Required	1 month	3 years
LTLF	Simulated	Simulated	1 year	30 years

Table 2. Availability of weather, economics, and land use variables.

	Accurate	Inaccurate	Unreliable
Weather	1 day	2 weeks	=> 2 weeks
Economics	1 month	3 year	=> 3 year
Land use	1 year	5 year	=> 5 year

2.3.2 Characteristics of Electric Load Series

A mixture of information about the load profiles is available for the forecasters by visual inspection. Being the easiest information to notice the cyclic patterns, and the difficult ones the slow changes on the nature of the consumption and/or the uncertainty attached to the consumption often called volatility. This information is usually called components and it study came when early economists try to understand the nature of a business cycle.

They began studying series in search of calendar effects (prior monthly and trading day adjustments), trends, cycles, seasonal, and irregular components. These components could be added or multiplied together to constitute the time series. The decomposition could be represented by the additive and multiplicative unobserved-components decomposition:

Eq. 1 $$\hat{Y}_a(t) = \hat{P}(t) + \hat{T}(t) + \hat{S}(t) + \hat{C}(t) + \hat{I}(t)$$

Where $\hat{Y}_a(t)$ represent the non-linear additively composed time series, $\hat{P}(t)$ the prior distribution of the target, $\hat{T}(t)$ trend, $\hat{S}(t)$ seasonality, $\hat{C}(t)$ cyclical components, and $\hat{I}(t)$ irregularities.

On this section we will introduce the unobserved components of an observed time series. The modelling of these components by separate constitutes a useful approach to obtain an uncorrelated residual error.

2.3.2.1 Trend component

Is the part of the series movement that corresponds to long-term slow evolution. The trend is extremely slow and is easy noticed on long runs, an example of this is the influence of the global warming over the weather variables. In business, finance, and economics areas, for example, the trend evolve slowly due to technology, demographic, preferences and politics.

The trend could manifest different natures, frequently it describe a linear or quadratic behavior, but in non-linear cases, the series must be analyzed on logarithmic scales. This type of trend is common on finance fields where it are associated with the name log-linear trend. The trend is easily detected on the Sample ACF correlogram as a slow decay.

Of course the trend is not a decisive factor on short forecasting horizons, but this concept can serves to the EMS to filter corrupted data. Sometimes, the key variables are mixed with a trend pattern as result of a defective sensor preprocessing. In this cases the identification of the trend is crucial to nullify the variable until it not being cleared.

2.3.2.2 Seasonal component

Seasonality is that part of the series pattern that repeat each year or less. If the repetition is exact we are speaking of a **deterministic seasonality**, other case we are in front of a **stochastic seasonality.** The special characteristic of the seasonality is that comes from a high correlation between the phenomena and the calendar. The weather, for example, is a very important seasonal series. It comprises a group of variables that change around the year following a solid pattern. Any technology highly influenced by the weather will exhibit a seasonal behavior.

A key technique to modelling seasonality is the use of **seasonal experts**, which stablish the creation of seasonal specialized models by means of a data segmentation, or the use of an input variable which indicate the season of interest. On temporal terms the seasonality could be expressed on patterns dependent of vacations, holidays, working days, weekends and weather seasons. It possible say that the calendar and the holiday effect are special category among the seasonal patterns.

2.3.2.3 Cyclical component

Cycle is catch-all phrase for various forms of dynamic behavior that link the present to the past and hence the future with the present [70]. Cycles in finance could be a pattern that describe a cyclicality totally uncorrelated of the temporal dimension, as the commodities prices or the

Brent oil barrel cost. Cycles may display a persistence in such way that any sample could be easily linked with the past on a finite period.

Although, the trend could be easily modeled using polynomial regression, and the seasonality using a mixture of experts; In terms of complexity, the cyclical dynamic is far complicated to capture due to the wide variety of cyclical patterns. In these terms, all the methods classified on the state of the art dig their roots on the demonstration of the covariance stationarity of the dependent and independent time series.

It means, probe that after a certain number of steps, the target and the key drivers are not white noise showing some degree of autocorrelation & cross correlation. The steps could be summarized as:

- Checking of the series mean: it must remain stable over time.
- Checking of the variance: it must be constant over time at a confidence interval.
- Checking of the covariance stationarity: covariance structure must be stable over time.
- The first two steps could be easily checked using visual inspection, on third step is necessary calculate the auto covariance function of the signal. Don't get lost by the term auto covariance, it is only a mean centered version of an autocorrelation. The idea of calculate a correlation over the signal deviations outside the mean, allow us to calculate the percentage of white noise inserted on the signal (see sample autocorrelation and Bartlett bands).
- On the practice, the characterization of the cyclical components is an iterative process. It is preceded with the analysis of the target signal by means of the PACF in order to find the seasonal components on the signal. Once time the model has been customized to integrate the seasonal dynamics, the residual error over the prediction is analyzed. The PACF is applied on them in order to obtain as much influential lags exhibit the series.
- In this iterative process, the stop is produced once time the 95% of the correlation falls inside the Bartlett bands. That indicates that the residuals only contains independent and uncorrelated noise.

2.3.3 Consumer classes

The nature of the operations carried out by the consumers are diverse, this is expressed clearly on the characteristics of the load profile such as: volatility, peak demand, load duration curve, and load curve. Through the classification of the diverse loads the TSO can perform a close track of the users aimed to reduce the non-observability of the grid and enhancing the unit commitment.

2.3.3.1 Large aggregation of consumptions

This consumption is represented as a large aggregation of loads on a territorial limit such as regions, countries, or interconnected communities. Examples of big aggregations profiles can be seen on the actors of the transmission grid, and high-end actors of the distribution grid.

These load profiles are characterized by smooth transitions and cyclic transitions. As a large aggregation, more than serve to the dispatch, control or optimization of the energy; it provide a valuable statistical measure of the total energy distributed. This information is key for the LTLF because it serves to planning actors to perform studies about the pertinence of a grid expansion or investment.

2.3.3.2 Commercial and Industrial users

The industrial users have a production activity defined, this fact implies that consumption is directly correlated with the activity carried out, and the schedule of the operation. Furthermore, the consumption exhibit cyclical and seasonal patterns, with slightly differences in comparison with the last time where the same pattern occur. The volatility the profile user is composed by uncorrelated peak of production or faulty machinery, being a good indicator for maintenance analysis.

As their industrial counterparts, commercial users usually have specific energy contracts, which specifies cheap prices on the time frames where they are most active. Their use of electricity goes from lighting, HVAC, and office machinery. Commercial users comes on a range among private building, mall centers, or government institutions.

Their diversity on activity purposes make hardly to differentiate the kind of profile that they exhibit. However, in general the profile express a high volatility on the working period, mostly due the high aggregation of loads. The patterns found on this profile are seasonal and cyclic, expressing a clear tendency to be influenced by human’s factors.

2.3.3.1 Residential buildings and household users

Their private nature make them susceptible to have a high volatility, mostly subject to human criteria based on uncertainty factor. In most of the cases human decisions can be forecasted based on their needs and the influence of the externalities, such as weather variables, calendar, or politics. These facts can help to determinate the activation of certain loads.

Residential consumption can be totally absent of intraday patterns except a variation on the consumption during the day and a peak during night. The seasonal patterns are mostly expressed on a week level on non-holiday season. According to the level of aggregation on residential buildings is possible to stablish a pattern for intraday behaviors, but accurate 1 day predictions are still unreliable unless they forecast the cumulated energy consumed per day.

Residential and commercial clients are traditionally implemented more EMS. This fact is easily explained due to the major concentration of the demand found on the industrial users (2 - 10%) in contrast with others sectors. other reason is the high level of specialization of the production activity on industrial user which lead specialized EMS features.

2.3.4 Requisites of a good STLF system

A forecast is a little bit more than guess about the future, is make a probabilistic statement about the future of a certain temporal series. A forecast is also defined as an especial case of the prediction, which can derivate on a projection when the forecast is immerse on a conditional scenario or hypothesis. Because the aim of a forecast is guide on the decision making, a good forecaster is the key to sharp and decisive early action. In this section we will present the features of a good quality forecast system.

2.3.4.1 Fast execution

Computational resources every time increase their efficiency, reducing the cost of new and advanced processors and memory. On the other hand computational methods have a long span of life on literature, inclusive are rebranded for new applications such the case of deep learning and the use of multiple layer neural networks. This leave us with the last influential factor, the size of the data necessary to make the prediction.

At major system complexity, more variables must be added in order to model the internal dynamics. This means, more space necessary to keep the data and more dimensions to consider by the model. But, the complexity of the modelling could be easily reduced by correct use of the forecasting horizon. For example, VSTLF require a fast modelling time usually solved by algorithms that exploit the recency effect.

STLF have a span of hours to produce a forecast, which is more than sufficient time for the state of the art forecast methodologies. However, complex models such as hierarquical models and probabilistic forecast require multiples iterations to complete a forecast. These used to parallelize as much operations they can in order to process threads of the forecast process on multicore CPU’s. Finally, MTLF & LTLF systems doesn’t have any hurry to produce forecast, and their data sources could be objectively trimmed on the study phase.

2.3.4.2 Accuracy

The accuracy term comprises the major challenge of any forecast horizon. Accuracy can be perceived on VLST as a little deviation, but for the others intervals came in company with a probability of error associated. In general terms the accuracy could be classified on two, the accuracy achieved on the training-test process (benchmark accuracy), and the obtained one once time our prediction is compared with the validation set (forecast accuracy).

Benchmark accuracy comprises the standard measures of the model previous to the forecast. It include the information about the bias and variance of the model on the form of statistical errors, and the residual analysis of the prediction over the test set. This accuracy concerns only to the set-up modelling process and gives hints about lack on the model.

Forecast accuracy is found over the validation set and can be extrapolated to forecast on the future samples of the target. It comprises the probabilistic nature of the forecast achieved by the model (conditional probability distribution), and gives a measure of the margin error of the prediction as a function of time. On STLF the accuracy is described in terms of the error distribution associated to the forecast, this gives the error margins founded on the forecast plots.

2.3.4.3 Automatic deploy

Autonomy is the condition where by the forecast system achieve a complete or partial autonomy of the human expertize. The industry dedicated to provide solutions on the forecasting sector limit the autonomy concept to the options that can be automatically processed by the system once time the forecasting system is running. It means humans experts are artifices of the driver preprocessing, driver identification, and modelling method design.

Once time the EMS is operative, the human decisions are already coded and no interaction are required except those ones with the system user. But the modelling automatization could be pushed forward, it means that the identification of the key drivers, the selection of best suitable modelling algorithm according to the scenario (user, forecasting horizon… etc.), and inclusive the selection of the modelling strategy could be carried out by a set-up configuration system.

This configuration system could contain the preprocessing and the knowledge discover algorithms in order to boost the accuracy trough cleaning of variables, previously codding of the human expertize. Then, the Autonomy feature is reached by the use of automatic procedures and experimental criteria's in order to minimize the interaction of an expert user on the forecasting procedure.

2.3.4.4 Adaptability

The adaptability refers to the ability of a forecast system to keep a high accuracy in any kind of implementation scenario, focusing the modelling method to counter the volatility of the load profile. The adaptability on a forecasting system can be achieved by the automatization of the error analysis, in his form of distribution or residuals.

Unless this feature hasn't been deeply discussed on the state of the art, some authors has already implement approaches to bring a solution on wind and electricity power forecasting. On the wind power forecast implementation, the authors propose an experimental method to handle the temporal limitation of the forecasting algorithm and the uncertainty associated to the physical location [71]. The authors seem to tackle the problem with an automated preprocessing and iterative modelling.

On the electricity power forecasting implementation [72], authors present the adaptability concept as a the combination of a preprocessing step to find and select the predictors on the weather-load model, and a modelling that combines the prediction of a weather-sensitive load component and his complement. However, these approaches are not so far away that a common multi-agent implementation on subjects governed by the same nature.

On this thesis the adaptability feature is presented on three components of the forecasting methodology. The first one refers to the implementation of an ensemble modelling approach, which serves to capture the global probability distribution of the target on multiple base models trained from independent sub-sets. The second one refers to the multiresolution and cyclical analysis of the target, it refers about the time-frequency decomposition of the endogenous variables. These analysis brings major insights about the ensemble structure layout, which increase the accuracy making the model insensible of the forecasting horizon requirement.

The last component refers to an experimental method for the layout organization at the base modelling level, it brings a third degree level trough the creation of structural adapted base-models. Self-organized modelling algorithms, such as Cartesian genetic programing, are used as provider of customized weak learner’s networks.

These components bring to the approaches implemented on this thesis the ability to be easily implemented on users governed by different phenomena’s, as for example the elements on a smart grid controlled by a multy-agent system.

2.4 Data collection and preprocessing

One of the biggest challenges on the pursuit of major accuracy on forecasting is the preprocessing. It comprehends the identification, creation and treatment of the variables that governs or give hits about the future of the target. This section present the influence of the preparations carried out before the modelling task, together with the techniques employed to maximize the extraction of key forecasting elements.

This section start with a description of the most influent variables over the load consumption, continues with the techniques carried out in order to customize the final model to particular incidences related with human influences over the load profile, and finish with the introduction on the preprocessing techniques implemented on this thesis

2.4.1 Introduction to the forecasting variables

The variables employed to obtain a forecast are usually called **key drivers**, and they can be originated form the target (**Endogenous variables**); or from phenomena's which behavior scape of any causality generated by the target (**Exogenous variables**). These variables must express some correlation degree with the target, and their importance will be measured according with this measure. On the following paragraphs the most influent variables on load forecasting are presented.

2.4.1.1 The load profile and the endogenous variables

The load profile itself can be a good regressor variable, this fact is exploited on algorithms that used lagged samples of their forecast as for example autoregressive models (AR) or the autoregressive neural network (NARNN). On this thesis, the benchmark is settled to explore the autoregressive models and their counter parts which integrate the lagged samples as input variables at the margin of the model topology.

The lagged versions of the target have been considered as strong key drivers by the literature. Due to they collect information about the characteristics of the load influenced by conditions such as cycles, season, trends or recency. **The number and importance of the lags** is usually determined by the peaks obtained on the partial autocorrelation of the target. But those strong peaks are usually closely followed by smaller peaks (mirror peaks) caused by the recency effect.

In order to not interfere with the customization procedure aimed to control the recency effect, the correlation peaks near to the origin and mirror peaks are discarded. The most widely implemented lags associated with the current load consumption are the equivalent to a day or a week samples.

2.4.1.2 Weather variables

The load behavior is strongly affected by the weather, especially on sectors where HVAC covers the majority of the consumption. Although some STLF methods doesn't require weather information most of the methods use them. The most frequent weather variables reported on literature include dry bulb temperature, wind speed, dew point temperature, wind direction, relative humidity, and cloud cover.

Among the variables mentioned before, the temperature (known as dry bulb temperature) is the most widely accepted. Variables derived of the temperature such as lagged versions of itself, windowed averages, and temperature derivate are the usual way to include the temperature on models.

2.4.1.3 Calendar variables

These variables can be used by the load forecasting system to extract information about periodical patterns that affect human or the process behaviors. In term of periodicity and grouping, the literature have been using a metric of season and moths; being the definition of season dependent of the climate on the service territory. This fact make the season classification, and the variables derivate, a case to consider on the customization of the model and the updating period.

For example, a typical load profile on the southern part of the world may have a longer summer, while northern localization have a longer winter. Linked to the seasonal specialization of the models, literature used to distinguish the transition between periods: summer (Jul 1- Sept 15), fade to fall (Sep 16 - CST), fall (CST- Nov 30), winter (Dec 1 – Feb 15), fade to spring (Feb 16 – DST), spring (DST – May 31), fade to summer (Jun 1 – Jun 30).

On STLF, the most important to define the calendar variable to use is recognize and classify the dynamics of the week. Factories used to work on working days and activate fewer loads on weekends. Household users tends to wake up late on weekends and provoke a displacement on weekend morning peaks. Furthermore, methods to group load profiles according to his temporal behavior may be influenced by the human factor as beliefs and laziness.

This leaves us with a visual method to identify the daily clusters on the load profiles, the optimal method will be introduced on the section referred to the **week profile and the weekend effect**. In terms of intraday behavior, the day could be divided at the engineering discretion. In base of the seasons the hours can be grouped in 6 groups, which aren’t necessarily match identically among seasons. But, in general terms, the simplicity of modelling four invariant hour profiles is well accepted.

The previous method is part of the customization procedure, considered as last step after the definition of the main modelling algorithm. For that reason, is usual get previous versions of the forecast by the use of input variables that gives hints to the learning algorithm to discover the correlation time-consumption. Those variables are usually expressed as integers that correspond to the **Hour of the day** and the **Day of the week**.

2.4.1.4 Labor driven & economic indicators

Variables related with the process carried out on the facilities of the user are directly correlated with the load profile. This variables gives a hint about the amount of energy necessary to carry out the process and are closely connected to the working calendar. Among them the most common are the **working days**, **the scheduled production** and **current production**.

Economic indicators also have proven a correlation with the load profile on long term. Population incremental ratio, land use, and others macro indicators and policies are in this group of drivers. On the household sector, labor variables as working calendar could has a major significance on the working days, but weekend are totally governed for stochastic process derived of the human factor.

2.4.2 Customizing the Benchmarking Model

On any energy management project, the modelling strategy must be designed based on the knowledge of the conditions, features and necessities of the specific implementation to be carried out. This characterization of the scenario (type of user) usually comes previous to the definition of the model to implement, in order to not oversize the forecasting application. But can be placed on the last stage of modelling in order to reduce the forecasting error.

This section introduce the theory behind the customizing procedure carried out on the benchmark of the models described on this thesis.

2.4.2.1 The recency effect

This effect refers to the “memory” of any sample of the target signal, respect to the recent past samples of influent variables such as temperature, dew point, irradiation, humidity, economics, etc. The following procedures are proposed to model the recency effect of the variables and sharp the original forecast.

Step 1, calculate a **simple moving average** of the key driver on the preceding 24 hours. This variable will be inserted as an input vector, as well as the original inputs, on a second layer. This model will make a close calibration of the prediction originated on the first layer by means of a bias correction. At the end of this step the model must be evaluated in order to measure the improvement on the MAPE.

If there is a negative increment on the accuracy, you can jump to the step 2. Vice versa, you can continue adding samples to the moving average until reach the forecast horizon desired.

Step 2, replace the moving average by a **weighted moving average** using the equation

Eq. 2 $$X_w(t) = \sum_{k=1}^{24} \propto^{k-1} X(t-k) / \sum_{k=1}^{24} \propto^{k-1}$$

The smoothing factor $\propto$ is determined by using values from 0.95 to 0.8, being 0.95 the closest sample to the forecasting point and 0.8 the most distant sample. Is important remember that steps above presented reach an accuracy improvement on the near forecast horizon (1-24 hours), and the accuracy tends to decrease with time.

2.4.2.2 The week profile and the weekend effect

Most of the loads profiles suffer a big change mostly influenced by social conventions. Office buildings close, factories close, commercial buildings increase their consumption, and household sector obtain major uncertainty. People used to wake up late than working days, shifting the morning peak one or two hours. Also Sunday and Saturday are completely different on the last two cases due to departments stores close early at the Sunday.

After a weekend, factories must resume the production line, which usually leads a peak of consumption than usual. Monday mornings also are different of the weekdays, due to the pronounced slope that they manifest. In order to customize the weekend effect have been proposed three steps, implemented as the discretion of the programmer and based on the best results over the generalization error.

Step 1, a variable called **weekday** is declared in the range of integer numbers from 1 to 7. This variable will be inserted as input together with the key variables in order to leave the classification of the different profiles in charge of the model. If the MAPE is reduced in comparison with the model without the weekday input proceed to Step 3. Otherwise proceed to Step 2. Alternatively on this step you can use a vector of decimals values between 1 to 7.

Step 2, this steps exploit the differences between the days and their characteristics patterns. In this step the prediction will be in charge of **five different models**, each one in charge of model each one of the following days: Monday, Tuesday-Thursday, Friday, Saturday, and Sunday. These groups have been stablished by the experience of the author through several cases scenarios and supported by literature benchmarks [73]–[75]. If the MAPE decrease you could still try the step 3 in order to compare results.

Step 3, based on the key drivers such as weather variables or production is possible make a clustering classification of the load consumption. The groups can be obtained using the **Lloyd algorithm** guessing how many cluster could be since the beginning, which must be contrast by the modelling implementation.

Alternatively, **hierarchical clustering** could make a progressive clustering between a maximum amounts of cluster to the initial population. This allows you to observe the progression (dendrogram) and prone where the number of branches became low but conserve the maximum like hood.

2.4.2.3 The holiday effect

Holidays can come fixed by date or fall on any day of the week. But, we can be sure that holidays are a back pain for the forecasters because the load profile is similar to weekend effect, however it is located on wrong places affecting the day classification process carried out before. Of course, the human factor (skip days between holidays or holidays to weekend) must be considered on detailed applications, but it will be skipped on this thesis implementations.

In case of the holidays occurs on the working days, it can be threaten as weekends and the surrounding days as Mondays and Fridays. This can be an easy task if the country year calendar is available, or the scenario-related calendar is available. Holidays can be classified on fix-date holidays and fix weekday ones. Fix-dates holidays are not easy to forecast because there is not similar days on the historic record.

On the next steps is presented the strategy to deal with holidays. This strategy can be easily obtained by close inspection of the load profile behavior.

Step 1, classify the holidays according to the effect of it has on the load profile. Use on the historic record for this task.

Step 2, Gather only those ones with major significance and classify them in terms of their week position. In parallel, observe the special cases (New Year, Columbus Day, thanksgiving, labor day, etc.), and classify then on a special class.

Step 3, follow this guideline:

- If the holyday is observed on a weekend, no especial action is carried out.
- If the holyday is on a working day (except Friday). The day is treated as a Sunday (day after as Monday, and day before as Saturday).
- If the holyday is on Friday, the day is treated as a Saturday.
- The treatment of special holiday's class must be copied of previous records or based on social conventions.

This strategy is compatible with the naïve model (created without take care of the holidays), but the inputs must be modified in order to forecast the holidays. Other option is reserve a model for holidays, but it will be against the KISS principle.

2.4.3 Data preprocessing

On load forecasting is usual that gathered data doesn't present the optimal conditions to work with. On this case the forecasting experts must perform an initial analysis to measure the importance of the variables related with the target to predict, at the same time that defines an automatic procedure for the cleaning of the key drivers.

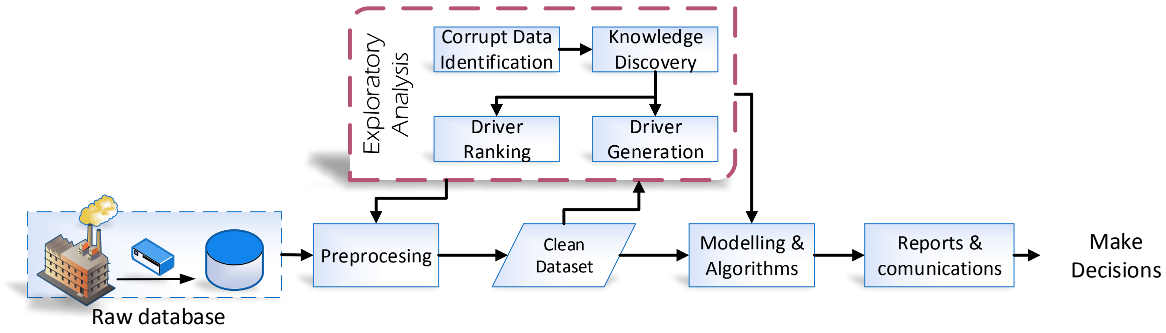


Figure 7. Preprocessing process flowchart.

The preprocessing approach presented on this thesis is introduced on the previous figure. The **first step** on the preprocessing task corresponds to the target signal treatment. On this step the experts clean irregularities on the signal such as outliers, gaps, and sensing errors without replacing the samples. Their objective is analyses the samples that belongs to the 95% of the Gaussian distribution and observe that irregularities don't present any cyclic pattern or suitable to be correlated.

This samples are represented as white noise on this stage, but will be saved as incidences to be analyzed on the diagnosis and predictive maintenance EMS functionalities. On the **second step**, knowledge discovery algorithms are executed in order to detect and measure the characteristics of the drivers such as trend, seasonal, cyclic effects, distribution, or correlation.

Based on the previous measures, the experts automatize the creation of the most common key drivers such as weather, calendar and endogenous variables. Variables derived from the target, such as the resulted of temporal or frequency transformations, are also added on this step. In parallel, a customized algorithm measure of the importance of the drivers as predictors. This last procedure is susceptible to be automatized, but always require of the human criteria in order to maintain the simplicity and not oversaturate the model with many inputs.

Once time the key variables are totally identified, on the **third step** the expert must codify the relevant procedures to obtain them automatically. On the preprocessing algorithm must be declared the procedures to detect and handle corrupt data, and the variable generation process. On the following sections we will introduce the preprocessing procedures implemented on this thesis.

2.4.3.1 Detection and handling of corrupt data

The data gathered from the SCADA regularly suffer of gaps, atypical values, or white noise. This data can induce error on the efficiency of the forecasting system without the proper treatment. For that reason statistical analysis are necessary in order to identify and remove the corrupted data automatically.

The first procedure is the verification of the covariance stationarity on the dependent (target) and independent (key drivers) time series. This procedures allows to the algorithm fulfill the first logical condition, the target or any of the signals are not pure white noise.

The second procedure includes: the identification of the gaps in order to fill them with NaN values; the identification and exclusion of the outlier data. The criteria to identify and remove them is based in the analysis of the standard deviation. In the case that any sample inside of a day exceed a boundary imposed by three times the value of the standard deviation of the same day, it is automatically removed.

This leads some problems on the exclusion of samples of quick change dynamics such as samples on early Saturday or late Sunday, whose inclination is higher than intraday variation. The solution is consider these few samples NaN and fill them with KNN.

The third procedure implemented is the gap filling of consecutives NaN samples lesser than the equivalent of one day samples. The decision of set this maximum amount of samples to fill was based on the premise of not introduce large modifications on the raw data collected because that could lead to biased predictions.

On the other hand, if the gap is as big as a week or a month, fill these with interpolations could cause an effect on the global and local average as well as on the signal distribution. For this reason, sectors with big gaps will remain as NaN on the modelling set.

The identified gaps to fill are set as missing values, and later introduced on a K nearest neighborhood algorithm. The most suitable predictors used on the KNN algorithm are set experimental test, those are the variables Time, WeekDay, WorkingDay.

2.4.3.2 Time series filtering

Instead of erase non desired components of the time series, the filtering process are implemented in order to highlight specific time or frequency zones where the cyclical or seasonal effects are remarkably visible. The filters implemented are presented attending to the primary objective of their implementation.

The **first group** are those implemented to discover knowledge from the data as well as to score the importance of the key drivers. The first technique usually observed on time series analysis are the sample autocorrelation function (**Sample ACF**) and the partial autocorrelation function (**PACF**). These functions serves originally to found the grade of auto regression that the system presents.

It means how many periodic signals are presented on the time series. Based on the number of peaks it is possible calculate how long is the period of the seasonal effect and his magnitude. This derivate on the number of lagged target versions added as inputs to the forecasting model.

Same result is easily obtained by the use of the **Fourier analysis** on the target signal. On the single side amplitude spectrum is easy to observe the frequencies who exhibit a seasonal effect and perfectly match with the previous analysis, despite of showing little differences on the lag ranking.

As you already notice, none of the previous analysis measure the importance of the exogenous variables against the target. Most basic techniques such as correlation matrix serves to rank the exogenous variables via the analysis of their variances, on this case the use of the covariance matrix is only plausible when the variables have similar range and scale.

In this scenario, exploratory data analysis (**EDA**) seems to be perfect for the study across multiple dimension of the entangled effects among the variables. MVA flagship visual technique called Principal component analysis (**PCA**), again the correlation based PCA is the indicate option if the series doesn't have similar range or scale.

On the other hand, PCA is extremely sensible to unit change and standardization, so their use must be limited to a human supervision. Furthermore, the graphical nature of this technique limit his use to a handy tool during the preliminary study

Although EDA techniques are widely used by forecast experts, the analysis of the results are totally dependent of the expert criteria. This means each preliminary study is custom for the application and where only the procedure can be repeated for future implementations. Fortunately, data mining provides a series of procedures to calculate the importance of the input drivers with low human intervention.

In **Data Mining**, Feature Selection is the task where we intend to reduce the dataset dimension by analyzing and understanding the impact of its variables on a model. Such analysis allows us to select a subset of the original variables, reducing the dimension and complexity. During a subset selection, we try to identify and remove as much of the irrelevant and redundant information as possible.

Techniques for Feature Selection can be divided in two approaches: feature ranking and subset selection [76]. In the first approach, variables are ranked by a given criteria and then variables above a defined threshold are selected. In the second approach, the techniques explore on a space of variables subsets for the optimal subset.

Moreover, the second approach can be split in three methodologies: **Filter approaches**, the variables are selected and are used as a subset to execute a classification algorithm. **Embedded approaches**, the feature selection occurs as part of a classification algorithm. **Wrapper approaches**, an algorithm for classification is applied over the dataset in order to identify the best variables.

On this study we are selected two algorithms from feature ranking and one from subset selection approaches respectively. **One Rule**: is a simple, yet accurate, classification algorithm that generates one rule for each consumption unit in the data and then selects the rule per each consumption unit with the smallest total error as its "one rule". **Correlation filter**: This algorithm finds the weights of the energy driver candidates basing on their correlation with the consumption unit.

Consistency-based filter: This algorithm finds a subset of energy drivers using consistency measure for continuous and discrete data. These techniques have been selected due to the low power of computing, the simple formulation of the algorithms, and the different types of scores to measure the weight of any predictor variable over the target.

Second group are those filter techniques implemented to create new variables from the original set. On this group we can found time-frequency analysis techniques such as **Wavelet transform**, the Hilbert-Huang transform, or the short-time Fourier transform (STFT). Discrete operators are also added on this thesis such as the Scaling.

On this thesis, our approaches make use of the wavelet transform in order to obtain smooth filtered versions of the target signal, which allows to the forecasting algorithm assign more weight to the waveform instead that try to model the withe noise.

Scaling is a discrete operator used to obtain multiple quantized versions of the target attending the window average of the signal. On load forecasting is usual that the longest seasonal effect moves with the weather season, practically looking like a trend for a STLF algorithm.

As a result, the solution to be implemented needs not only to follow the offset on the time series, but also to recognize, adaptively, the pattern exhibited. The quantization procedure start with the collection of the ranked lags. Then the algorithm perform an average on a number of samples equal to the lag number (window of samples), the procedure starts from the last sample acquired.

Once time the average is obtained, all the samples on the window are replaced by the average. The result are multiples stepped versions of the target signal according to the number of lags selected.

2.5 Load forecasting techniques

On this section are introduced the load forecasting algorithms implemented on this thesis. They have been selected due to capacity to maximize the adaptability, the accuracy and the automatization of the forecasting methodology proposed on this thesis as a novelty. Furthermore, the forecast algorithms have been grouped and introduced according to the algorithms complexity.

The first subsection describes a set of regression techniques implemented to calculate the prediction intervals of the most complex forecast algorithms such as computational intelligence and machine learning approaches. The subsection start with a family of multivariate spline regressor, and finalize with the generalized autoregressive conditional heteroskedasticity models (GARCH).

The second subsection introduce the computational intelligence (CI) methods used as forecast algorithms. These algorithm comes from the family of neural networks and have been selected due to the strong popularity on literature, and their ability to achieve a high accuracy.

The third and last subsection are dedicated to machine learning (ML) approaches on load forecasting. It start with an introduction to support vector machines (SVM), and continues with the description of an adaptive network-based fuzzy inference system (ANFIS). Previous ML approaches and CI methods constitute the base of the benchmark of this thesis, because they are ranked with medium complexity, interpretability, and relative high accuracy.

Packed on this subsection evolutionary computation is presented. It includes two multipurpose modelling techniques, Cartesian genetic programming (CGP) and Neural Cartesian genetic programming (NCGP), novels among the load forecasting implementations.

2.5.1 Regression techniques

Simplest regression methods usually achieve good generalization errors due their strong learning algorithms, and their simplicity on the description of the polynomial matrices governing over the regions on the data to forecast. However, regression methods are a good example to show the usual trade-off among the flexibility of the model and the interpretability, often called complexity.

Linear models poses a lower complexity but are too rigid on large datasets and/or non-linear dependences. Second and third degree interpolations covers partially the non-lineal dependences, but equally fails on large data sets. **Multivariate adaptive regressions spline (MARS)** [77] use tensor product splines, who are flexible and simple to interpret if the level of interaction on the hinge functions is small.

Time series fitting by polynomials adjusts on a data set or regions of itself presents two challenges, the first one is that piecewise linear model requires that a priori know the number of regions (pieces) needed to model the data avoiding overfitting. Second, the piecewise polynomial structure is itself sharpie; owning discontinuities in derivatives and data variability along the boundaries. A solution of these problems are introduced on the proposed **Bayesian Multivariate Linear Spline (BMLS)** [78].

From the point of view of uncertainty analysis, the previous modelling errors could be classified on errors due to parameters and structure of the model respectively. However, some errors come from the volatility/innovations present on the data. The terms volatility or innovation refers to disturbances that cannot be explained as corrupt data and neither product of trend, seasonal or cyclic effects.

Assuming the condition of non-independence on the volatility, the hypothesis that the innovation error is pure white noise is discarded and a heteroskedasticity forecast technique can be adjusted to the time-variant volatility. **Generalized autoregressive conditional heteroskedasticity (GARCH)** is the forecast technique selected to model the time-variant volatility.

These algorithms, prior their great features as predictors, have been selected because their low computational requirements and their great accuracy on large data bases. It grant them the possibility to be implemented in dedicated applications as the nodes of a multiagent system, which is the hot trend today with the awareness about the grid decentralization and the smart grid.

2.5.1.1 Multivariate spline interpolation

You might guess that the idea that a simple polynomial cannot model an entire data set seems obvious; but take a step more and guess the number of regions on the data to model, and the order of the polynomial to adjust each of those regions sounds as an optimization problem.

Well, the idea that a high order interpolation was already stated by Runge in 1901, and the natural solution was stablish a low order polynomials between equidistant points of the data. **Multivariate spline interpolation** are considered as piecewise function, an ensemble of polynomials models which adjust polynomials to pieces of the data set called regions.

Eq. 3
$$\hat{y}(t) = \begin{cases} a_{k1}t^n + b_{k1}t^{n-1} + \cdots + z_{k1} & if \quad 0 < t \leq t_{k1} \\ a_{k2}t^n + b_{k2}t^{n-1} + \cdots + z_{k2} & if \quad t_{k1} < t \leq t_{k2} \\ a_{km}t^n + b_{km}t^{n-1} + \cdots + z_{km} & if \quad t_{km-1} < t \leq t_{km} \end{cases}$$

The control parameters on a Spline algorithm are: the number of equidistant nodes (knots) which split the data on the regions (m), the degree of the polynomial which is usually stablished between linear and cubic order (n), and the penalty coefficient which ensure the fit or constrain level of the piecewise function to the time series (p).

Benefits of the splines implementations are: lower computational cost because tuning get focused only on two parameters, the degrees of freedom (m) and the penalty coefficient (p); custom control of the overfitting and smoothness by the possibility of increase the number of basis functions and the degrees of freedom (m) without risks; maximum interpretability of the model; discontinuities avoided on high order kernels such as cubic spline.

2.5.1.2 Multivariate adaptive regressions spline (MARS)

A second piecewise approach called **MARS** follows the steps of the previous approach but replacing the polynomial adjust for hinge functions [77]. Hinge functions $\boldsymbol{h_i(x)}$, has their mirror version on the activation function rectifier linear unit from neural networks, or the ramp function on signal processing. These are expressions who operator describe a line starting from a knot $\boldsymbol{t}$ on straight or reversed version [79].

Equations below presents a reflected pair of hinge functions.

Eq. 4 $$\boldsymbol{h(x-t)_{\pm} = \begin{cases} h_t(+(x-t)) = \begin{cases} x-t & if \quad x>t \\ 0 & otherwise \end{cases} \\ h_t(-(x-t)) = \begin{cases} t-x & if \quad x<t \\ 0 & otherwise \end{cases} \end{cases}}$$

Other forms of express hinge functions.

Eq. 5 $$\boldsymbol{h_t\big(+(x-t)\big) = max(0, x-t) = (x \cdot t)_+ = (x-t)_+ = h(x|t)_+ = h(x-t)_+}$$

A MARS model is defined as:

Eq. 6 $$\boldsymbol{\hat{y}(x) = \beta_0 + \textstyle\sum_{m=1}^{M} \beta_m \, h_m(x) \qquad , \qquad M = total\ knots}$$

Where $\boldsymbol{h_m(x)}$ could be any version of the reflected pair, and be a product of univariate hinge functions.

Eq. 7 $$\boldsymbol{h_m(x) = \textstyle\prod_{s=1}^{k_m} h\big(x_{i(s,m)} \big| t_{(s,m)}\big) \qquad , \qquad 1 \leq m \leq k}$$

Where the subscript $\boldsymbol{i}$ means a particular explanatory variable, $\boldsymbol{t}$ the knot where the hinge function fix the basis spline, and $\boldsymbol{k_m}$ are the maximum number of hinges function allowed to interact at one knot. Also, it can be defined as the number of interactions among the variables and knots. On this case of $\boldsymbol{k} = 1$, the model will be purely additive. Next equation presents an example of a linear MARS function, hinge functions can express quadratic an cubic terms also [80].

Eq. 8 $$\boldsymbol{\hat{y}(x_1, x_2) = \beta_0 + \textstyle\sum_{m=1}^{knots} \beta_m \, h_m(x) = \begin{cases} \beta_0 \\ + \beta_1 \, h(x_2 - t_1) \\ + \cdots + \ \beta_{m-1} \, h(x_1 - t_{m-1}) \, h(-(x_2 - t_{m-1})) \\ + \beta_m \, h(-(x_2 - t_m)) \end{cases}}$$

MARS models allows the product of several hinge functions in order to reproduce the correlation of two or more inputs. The two stage MARS training method consist on a **forwards pass** where a heuristic algorithm adds two mirrored hinge functions (reflected pair) in order to reduce the residual error. On each new term the new and previous parameters of the model must be adjusted, as well products of hinge functions tested.

This process continues until the residual error converge to a local minimum, or the maximum number of terms is reached. **Backward pass** serves to increase the model generalization, avoiding the overfitting imposed on the previous stage. It prune the model term by term searching the best sub model, the subsets are then compared using the Generalized cross validation partitioning method.

This last step itself correspond to a basic but efficient method of variable selection, because term discarded in pursuit of the generalization are usually the less important drivers. Others impressive features of this algorithm are: the incredibly lower computational cost which can handle large data sets in minutes, the interpretability and flexibility of the models, the introduction of a partitioning method to obtain the best suitable bias-variance tradeoff, and quick prediction results.

2.5.1.3 Bayesian Multivariate Linear Spline (BMLS)

BMLS was introduced on [80] carry the same model structure presented on MARS, but with an training strategy entirely focused on mimic the probability distribution on the data. As well as other interesting articles it not save his approach form plagiarism. Recently, some authors have made a semi plagiarism of the technique calling it BARS [81].

the algorithm assume a Gaussian distribution on the data set, implying that the number and distribution of the basis splines are treated as random on the prior conditions, to be later organized by a probabilistic method. It brings the model a **model spatially and data adaptive** feature, a characteristic desired on this thesis.

The BMLS algorithm inherit some characteristics of the MARS algorithm, as his use of basis functions (hinge functions) to construct the regression surface bringing continuity at the pieces boundaries at mean level, his soft regression surface due to interaction of the hinge functions, and intrinsic variable selection. But differs on the strategy to locate the basis functions and knots which is entirely driven by a Bayesian approach, and the method to simulate the probability distribution and the convergence which is performed by a Markov chain Monte Carlo (MCMC) sampling.

Let's start with the description of the algorithm. BMLS assume that the dependent variable $\boldsymbol{y}$ is a result of a function of the explanatory variables plus a Gaussian noise term. The fact that the model consider a Gaussian noise is important, because it represents the assumption of a homoscedasticity posterior distribution, making the probability density function completely Gaussian for any slice. This will be discussed later on the Appendix B, specifically on the probabilistic predictive interval estimation.

Eq. 9 $$\boldsymbol{y_i = f(x_i) + \epsilon_i \; = \; f(x_i) + N(0, \sigma^2)}$$

As we stated preciously the BMLS is based on a piecewise linear regression algorithm, then to model the piecewise structure of a data set we adopt the following basis function approach equation. Different from mars equation we reserve m variable to refer to the model, and t to refer to the variable time.

Eq. 10 $\hat{\boldsymbol{f}}(\boldsymbol{x_i}) = \boldsymbol{\beta_0} + \sum_{j=1}^{k} \boldsymbol{\beta_j}\,(\boldsymbol{x_j} \cdot \boldsymbol{\mu_j})_{+}\quad ,\ \boldsymbol{k = total\ knots,\ \mu = position\ of\ basis\ function}$

As we know the coefficient μ determine the position of the regression coefficient that determines the gradient (β). On typical piecewise linear models the parameters are set to single optimal values chosen according to some cost function, such a penalized likelihood or a cross-validation score on MARS. This procedure fails to merge uncertainty in the setting of these values.

The Bayesian approach of the BMLS, places the probability distribution on all unknown parameters. Let's start referring as a particular model structure and noise variance in function of their inner parameters $\boldsymbol{\mathcal{M}(k, \mu, \beta, \sigma^2)}$: number of knots, position of the basis function, gradient coefficient, and total variance. Where $\boldsymbol{\mu}$ is defined as the set of spline parameters $\boldsymbol{\mu} = (\boldsymbol{u_1}, \dots, \boldsymbol{u_n})$ same for $\boldsymbol{\beta}$. The BMLS use the Bayes's rule in order to set the parameters.

Eq. 11 $\boldsymbol{P(\mathcal{M}|\mathcal{D}) = P(\mathcal{D}|\mathcal{M}) * P(\mathcal{M})\ /\ P(\mathcal{D})}$

It calculate the posterior $P(\boldsymbol{\mathcal{M}}|\boldsymbol{\mathcal{D}})$ distribution of the model based on the prior distribution $P(\boldsymbol{\mathcal{M}})$ and the likelihood $P(\boldsymbol{\mathcal{D}}|\boldsymbol{\mathcal{M}})\ /\ P(\boldsymbol{\mathcal{D}})$. Based on the previous equation the point predictions under the probabilistic **posterior model space** can be given as expectations.

Eq. 12 $\boldsymbol{E(y_i|x_i) = \int \hat{f}_{\mathcal{M}}(x_i) P(\mathcal{M}|\mathcal{D}) d\mathcal{M}}$

Where $\boldsymbol{\hat{f}_{\mathcal{M}}(x_i)}$ refers to model $\boldsymbol{\hat{f}(x_i)}$ with $\boldsymbol{\mathcal{M}(k, \mu, \beta, \sigma^2)}$ parameter settings. For the piecewise linear models the number of planes $\boldsymbol{k}$ is one of the unknowns, and the dimension of the posterior density $P(\boldsymbol{\mathcal{M}}|\boldsymbol{\mathcal{D}})$ Is varying and typically complex. For that reason when making inferences on the model space is necessary employ simulation methods such as Markov chain Monte Carlo samplers (MCMC).

Now, let's introduce the Bayesian model estimated for the posterior distribution. As we say before the log-likelihood model of the posterior distribution follows the form of our assumptions of Gaussian noise.

Eq. 13 $\boldsymbol{l(\mathcal{M}|\mathcal{D}) = -nlog(\sigma) - \frac{1}{2\sigma^2} \sum_{i=1}^{n} \{y_i - \hat{f}_{\mathcal{M}}(x_i)\}^2}$

Through a mathematical description of the steps needed to present the probabilistic nature of the model in terms of parameters subject to be implemented on an optimization algorithm [78], the joint prior distribution of the model space can be written in a factorized form as:

Eq. 14 $\boldsymbol{P(k, \beta, \mu, \sigma^2, z, \gamma) = p(\beta|\sigma^2, k)\ p(\sigma^2)\ \ p(\mu|z, \gamma, k) p(\gamma|z, k) p(z|k) p(k)}$

Using this representation, the probabilities can be calculated based on known matrixes. On the other hand, the among the model parameters two new terms have been introduced. $\boldsymbol{z}$ Represent the maximum number of interactions or hinge functions are allowed in any piece (spline), and $\boldsymbol{\gamma}$ define the indicator vector that describe the interaction following the formula:

Eq. 15 $\boldsymbol{z_j = \sum_{d=1}^{p} \gamma_{jd}}$ $\boldsymbol{,\ z_j = number\ of\ interactions\ at\ jth\ spline,\ \gamma_{jd} = the\ dth\ element\ at\ j}$

Now, we wish to sample the posterior density (posterior distribution) $P(\boldsymbol{\mathcal{M}}|\boldsymbol{\mathcal{D}})$ in order to make an inference of the predictive conditional distribution and the mean regression (point forecast). But because conventional MCMC needs to know a priori the number of pieces or knots, the BLMS use the reversible MCMC jump sampler. This algorithm redistribute the size of the pieces making an optimization of the inertia or variance contained on it.

A simplified explanation of the MCMC process, by whom the basis (hinge) functions are added, removed or modified (actions) follows these steps: first an action is required, the model at time t has the following state $\boldsymbol{\mathcal{M}_t} = \{\boldsymbol{k}, \boldsymbol{\beta}, \boldsymbol{\mu}, \boldsymbol{\sigma^2}, \boldsymbol{z}, \boldsymbol{\gamma}\}$ at the Markov chain; second, for a random spline is stablished a basis functions and calculated the improvement on the posterior state.

The MCMC sampler is iterated until enough samples have been considered to have been collected with an initial portion discarded to allow for the chain to converge the sufficiently closer to its stationary distribution. The resulting samples are the BMLS models drawn from the posterior model space $P(\boldsymbol{\mathcal{M}}|\boldsymbol{\mathcal{D}})$. Is important know that the **mean regression surface** is reported as the average of the piecewise linear surfaces generated by the samples.

On summary, added to the spline modelling benefits the BLM algorithms provide an absolute control on the mathematical description of the kernel employed as basis function. The modelling surface constructed is soft and doesn't present discontinuities, this as result of the smooth kernels integrated. The last but not less important, the model avoid be over fitted due to the MCMC sampler iterates until the stationary distribution is reached, this means that optimal model and the number of basic functions are set in function of the global variance not the mean.

2.5.1.4 Stochastic time series models

Until now, the models presented has perform regressions made under pieces or slices of the dependent variable respect to the independent variables. They have assume Gaussian distribution on the errors, a stationary covariance, and a conditional distribution (only BMLS). But there is no really reason to believe that the errors are white noise, neither the conditional distribution remain constant and indifferent of the forecast horizon.

Remember, those assumptions comes from the theory that a valid regression model could model all the temporal effects until it possesses a constant error variance. On the early 80's authors such as Engle, Diebold, Lopez, Granger, and Kraft describe that under some circumstances the “error variance can change over the time and be predicted by the past forecast errors” [70], [82], [83].

They study financials econometrics processes with a high **autoregressive heteroskedasticity** associated such as the analysis of the inflation, which cause an increment on the volatility of the value of a stock option. Where error variance increase with the time the risk of investments increase (see more at [83]). These studies derivate on the assumption that in some cases **the value of the error variance can be a function of the time lag**.

In case of regression models, an autoregressive model with conditional heteroskedasticity error variable could be appropriate to model the financial risk or volatility. Based on the previous information, we assume that modelling the conditional heteroskedastic error variance presented on load forecasting can reduce the error due innovations. Finally producing an effective modelling of the non-normal time-dependent distributed errors. This characteristic is a strong point for the **adaptability feature** in order to reduce the volatility not measured.

Now let’s get introduced on the stochastic model algorithms using a combined description from [70], [82], [83]. Let’s suppose we have a time series from which any trend and seasonal effects have been removed and from which linear (short-term correlation, minor lags) effects may also have been removed. Thus r_t could, for example, be the series of residuals from a regression or autoregressive model

Eq. 16 $\boldsymbol{y_t = \beta_1 x_t + \varepsilon_t \quad and \quad \varepsilon_t \backsim N(0, \sigma_t^2)}$

Where ε_t denotes a sequence of independent modelling errors with zero mean and unit variance and σ_t may be thought of as the local conditional variance of the process. The errors here as well as other modelling approaches are assumed as normal, but this assumption is not necessary for much of the theory. In any case, the unconditional distribution of a data generated by a non-linear model will be generally fat-tailed rather than normal.

Let’s stop a minute, ε_t is white noise, some authors define white noise as strong or weak (independent or merely serially uncorrelated). When ε_t is independent, there is no distinction between the unconditional distribution of ε_t (PDF), and the conditional distribution of ε_t upon its past (CPDF). Hence, σ_t^2 is both the unconditional and the conditional variance of ε_t.

If ε_t is dependent, then its unconditional and conditional distribution differ. For that reason the Wold decomposition used as base to stablish the ARCH model denote the residuals as a conditional distribution. This conditional dynamic is explained on a **time-varying conditional distribution (time-varying volatility)** representation such as:

Eq. 17 $\boldsymbol{(\varepsilon_t | \varepsilon_{t-n}) \backsim N(0, \sigma_t^2)}$

Where n could be considered the lag order or as a model equation

Eq. 18 $\boldsymbol{r_t = \sigma_t \varepsilon_t}$

The notation r_t is used to emphasize that **models for changing variance (heteroskedastic prediction models)** are rarely applied directly to the observed data. Then, r_t should be (approximately) uncorrelated but may have a variance that changes through time, being represented in the form.

Once time we can observe the σ_t, various models can be assumed based on his time change. The **AutoRegressive Conditionally Heteroscedastic** model of order p, abbreviated **ARCH(p)**, assumes that σ_t^2 is linearly dependent on the last p squared values of the time series.

Eq. 19 $\boldsymbol{\sigma_t^2 = \alpha_0 + \sum_{j=1}^{q} \alpha_j \, \varepsilon_{t-q}^2}$

For example ARCH(1) shows how the conditional variance evolves through time according to the equation.

Eq. 20 $\boldsymbol{\sigma_t^2 = \alpha_0 + \alpha_1 \varepsilon_{t-1}^2}$

Notice the parallelism between the formula of an AR(1) and ARCH(1). The ARCH model has been generalized to allow linear dependence of the conditional variance σ_t^2 , on past values of σ_t^2 as well as on past (squared) values of the series. The **Generalized ARCH** (or **GARCH**) model of order (p, q) assumes the conditional variance depends on the squares of the last p values of the error series ε_t^2 and on the last q values of σ_t^2**.**

Eq. 21 $\boldsymbol{\sigma_t^2 = \alpha_0 + \sum_{j=1}^{q} \alpha_j \, \varepsilon_{t-j}^2 \; + \sum_{i=1}^{p} \beta_i \, \sigma_{t-i}^2}$

Where the parameters α, β must satisfy (α + β) < 1 for stationarity.

For example, the conditional variance of a GARCH(1, 1) model may be written

Eq. 22 $\boldsymbol{\sigma_t^2 = \alpha_0 + \alpha_1 \varepsilon_{t-1}^2 \; + \beta_1 \sigma_{t-1}^2}$

These models have been documented as viable algorithms to compute the **probability density function (PDF)** and the **conditional probability density function (CPDF)** on econometrics [84]. They are part of the set of parametric modelling algorithms that estimate the variance on the predictions such as the generalized linear models (GLM).

So, in order to **forecast the conditional variance one step ahead** follows directly from the model. **Forecasting more than one step ahead** is carried out by replacing future values of σ_t^2 and of ε_t^2 by their estimates.

Chatfield. [83] continues with some recommendations about the use of GARCH models. GARCH models have now been used in forecasting a variety of financial variables, where estimation of variance is important in the assessment of risk. These include share prices, financial indices and the price of derivatives such as options to buy a certain share at a pre-specified time in the future.

The evidence indicates that it is often important to allow a changing variance, but that GARCH models do not always outperform alternative models. Sometimes a random walk model for the variance is better than GARCH, while GARCH may not cope well with sudden changes in volatility or with asymmetry. In the latter case, something like EGARCH or a stochastic volatility model may be better.

An alternative to ARCH or GARCH models is to assume that σ_t^2 follows a stochastic process. This is usually done by modelling $\log(\sigma_t^2)$ to ensure that σ_t^2 remains positive. **Models of this type are called stochastic volatility or stochastic variance models**. It seems intuitively more reasonable to assume that σ_t changes stochastically through time rather than deterministically, and the resulting forecasts are often at least as good as those from GARCH models.

This statement open a new question about the nature of the load profile. Is stochastic the distribution of the variance/volatility?, it means at each interval of time the distribution presents a skewness an a kurtosis unique that will change following a behavior that cannot be follow by a linear equation such GARCH presents. Could be approximate by to a linear model which a certain grade of confidence?.

The degree of acceptance of the answer of this questions, rather than philosophical implications could be extrapolated to the existence of probabilistic models as Hidden Markov Chain models or Bayesian networks. Our expertise could notice that any attempt to model the stochastic variance component will continue under a mere exercise of modelling because some errors (such as the model structure error) will be impossible to avoid.

2.5.2 Computational Intelligence based models: Artificial neural networks topologies

In order to model the nature of the interactions presented among the key drivers and the target, this thesis has introduced alternatively the use of some members of the ANN family. The implementation of this topologies will highlight the effects of specifics effects such as recency, seasonality and trend. At the same time they will serve as a modelling algorithms to construct a benchmark, and compare our strategies.

2.5.2.1.1 *Non-linear Autoregresive Neural Network (NARNN)*

It is a neural network that forecasts a time series based on the past values, thus generating an autoregressive model. This method has been considered because the thermal convection follows a trend based on his past values [85]. This neural network will be implemented in order to check the dependency of the target with its own past. The following figure shows the structure of the NARNN model for a single output.

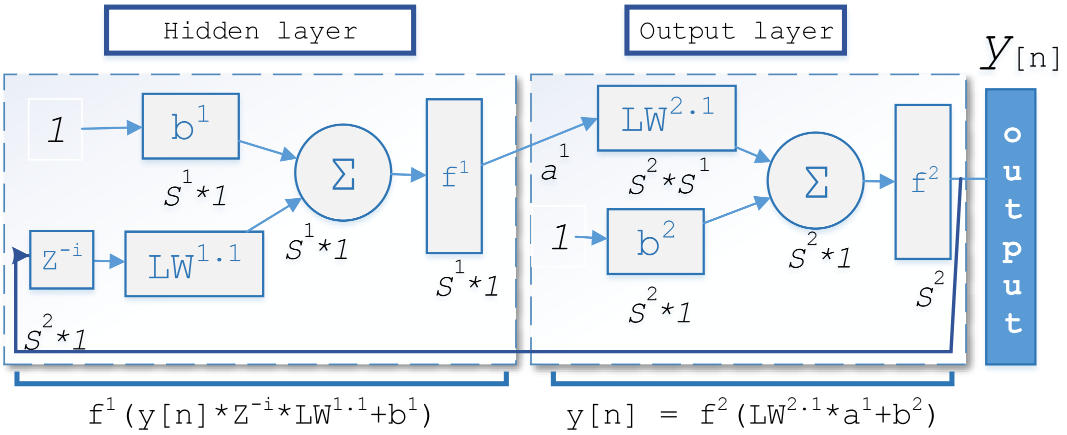


Figure 8. NARNN model structure.

NARNN is modelled as,

Eq. 23 $\boldsymbol{y[n] = f(y[n-i_1], y[n-i_2], \dots, y[n-i_n])}$

The output y[n] is a function of past values of outputs, where:

y[n-i1], y[n-i2],…,y[n-in] are the past output values at the ith sample.

uk[n-i1], uk[n-i2],…,uk[n-in] are the past input values at the ith sample.

f1 and f2 are the activation functions on the hidden and output layers.

IWq.k is the input weight matrix order s^2 * Rk, the superscript q denotes the layer number and k denotes the number of vector inputs entering the weight.

LWq.1 indicates the layer weight matrix of order s2*s1.

b2, b1 are the bias vectors of first and second layer respectively.

Rk denotes input vector of R elements.

Z-i indicates the number of lag samples to use.

The previous description presented will be considered as canonical, and is the work of the reader extrapolate it to future NN model structures.

2.5.2.1.2 *Non-linear Autoregresive Neural Network with exogenous inputs (NARXNN)*

NARXNN is a neural network used to forecast time series based on the past values. It has exogenous inputs, this means that the model uses a feedback version of its forecast and also current and lagged values of the inputs [85]. This is a basic algorithm that can replace an autoregressive polynomial, for that reason it is implemented to test the dependency of the target with it past and the exogenous variables employed on the forecast.

The following figure shows the internal structure of the NARXNN model.

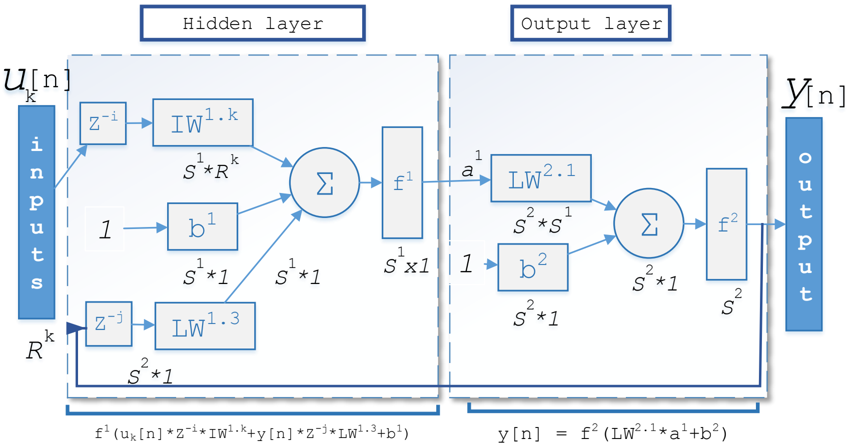


Figure 9. NARXNN model structure.

NARXNN is modelled as,

Eq. 24 $$\boldsymbol{y[n] = f(u_k[n-i_1], u_k[n-i_2], \ldots, u_k[n-i_n], y[n-i_1], y[n-i_2], \ldots, y[n-i_n])}$$

Where the output y[n] is a function of the past inputs and outputs values.

2.5.2.1.3 *Layer Recurrent Neural Network (LRNN)*

LRNN uses the specified signals as inputs, but also integrates a lagged version of the hidden layer outputs. This creates a directed feedback lagged circle also called internal memory, thus exhibiting a dynamic temporal behavior [85]. The load profile could have some behaviors that are activated by the combination of certain conditions of the inputs, and whose effect can persist on a fix time. This particular case could make ideal the use of the LRNN algorithm. The following figure shows the internal structure of the LRNN model.

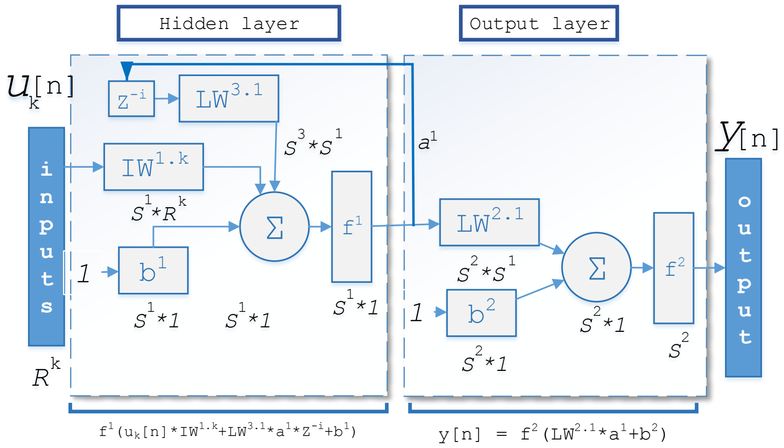


Figure 10. LRNN model structure.

The equation of the LRNN model is given as,

Eq. 25 $\boldsymbol{y[n] = f(u_k[n], a^1[n - i_1], a^1[n - i_2], \ldots, a^1[n - i_n])}$

The next value of the signal y[n] is regressed to the input and the previous values of the intermediate layer outputs.

2.5.2.1.4 *Feed Forward Neural Network (FFNN)*

It is a classical NN whose weights are adjusted through a back-propagation algorithm [85]. This type of network consists of multiple layers of computational units, usually interconnected in a feed-forward manner. By applying various techniques, the error is then feedback through the network. Using this information, the algorithm adjusts the weights of each connection in order to reduce the value of the error function by some small amount.

After repeating this process for a sufficiently large number of training cycles, the network will usually converge to some state in which the error is small. The following figure shows the internal structure of a NN model.

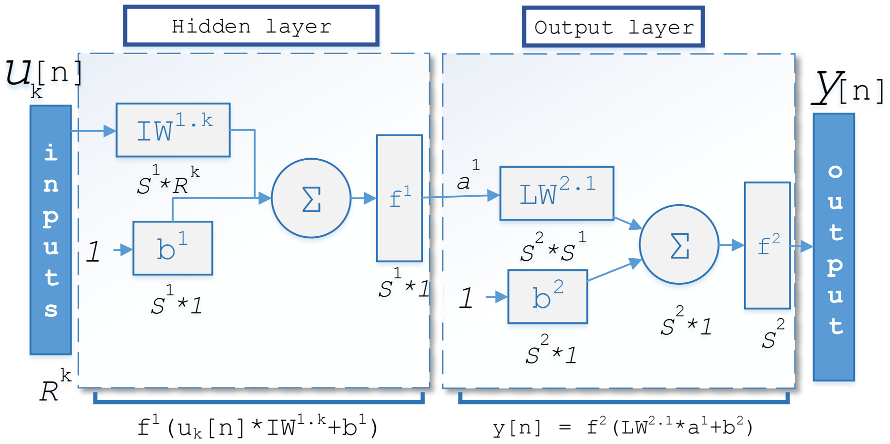


Figure 11. FFNN model structure.

The equation for the FFNN model is given as,

Eq. 26 $\boldsymbol{y[n] = f(u_k[n])}$

Where, the next value of the signal y[n] is regressed to the input.

2.5.2.1.5 *Cascade Feed Forward Neural Network (CFFNN)*

The following figure shows the internal structure of a CFFNN with a unique hidden layer. As in the FFNN, the back propagation algorithm adjusts the weights, but the architecture includes a connection from the inputs and every layer to following layers.

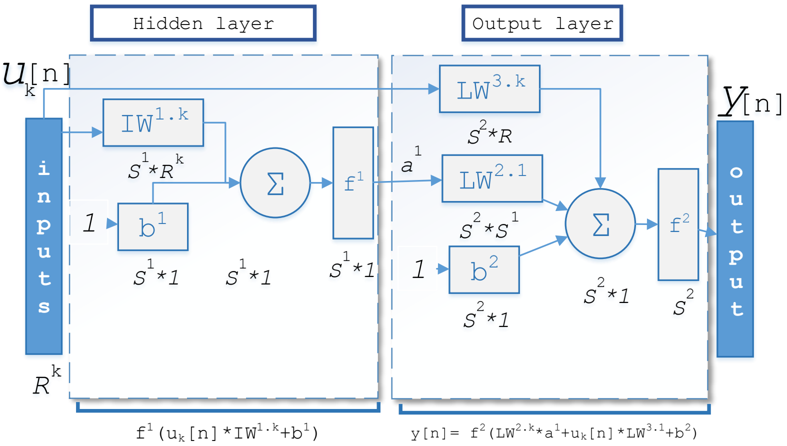


Figure 12. CFFNN model structure.

The equation for the CFFNN model is as,

Eq. 27 $\boldsymbol{y[n] = f(u_k[n], a^1[n])}$

Where the next value of the signal y[n] is regressed to the input and the values of the intermediate layer outputs.

2.5.3 Machine learning approaches

In competition with the artificial intelligence approaches, machine learning has introduced their highly adaptive candidates. On this section we will introduce a highly adaptable statistical learning algorithm called SVM, which compete on generalization capability with NN except that instead of find the local optima it searches for the global one.

As second algorithm is introduced an algorithm that mimic the linguistic expressions that humans use to take and communicate decisions. ANFIS algorithms are a mixture among soft codding of inputs realized by fuzzy logic, neural network structure, and piecewise weighted polynomials. Third algorithms covers evolutionary approaches in order to construct models driven by the force of convergent error evolution.

2.5.3.1 Support vector machines (SVM)

Support Vector Machine, proposed by V. N. Vapnik in 1995 through a statistical learning theory, is a comparatively new approach to the problems of classification, regression, ranking, etc. As a binary classifier, it tries to find an optimal hyperplane that maximizes the margin between data samples in two classes in a higher dimensional feature space derived from the original data space through a kernel function, while reducing the training errors.

As a linear regressor, the basic idea is to map the data into a high dimensional feature space by nonlinear mapping and then performing a linear regression in this feature space. The regression function performed by a vector machine is as follows:

Eq. 28 $\boldsymbol{y_t = \omega\, \Phi(x_t) + b}$

Being w, b estimated by minimizing the regularized risk function

Eq. 29 $\boldsymbol{R = \frac{1}{2}\|w\|^2 + C\frac{1}{l}\sum_{i=1}^{l}|y_i - f(x_i)|_\varepsilon}$

$\|w\|^2$ is the weights norm, which is used to constrain the model structure capacity in order to obtain better generalization performance. C is the regularized constant determining the trade-off between the empirical error and the regularization term. ε known as the intensive zone, is a linear loss function used as a measure for the empirical error.

The vector w can be written in terms of the data points

Eq. 30 $\boldsymbol{w = \sum_{i=1}^{l}(\alpha_i - \alpha_i^*)\phi(x_i)}$

With α_i , α_i^* being the solutions of the risk function. Considering the previous equations, we can get the regression function in the low dimensional input space.

Eq. 31 $\boldsymbol{f(x, \alpha_i, \alpha_i^*) = \sum_{i=1}^{l}(\alpha_i - \alpha_i^*)K(x_j, x_i) + b \qquad where \quad K(x_j, x_i) = \phi(x_i) \times \phi(x_j)}$

$K(x_j, x_i)$ is called kernel function, and is the inner product of the x vectors on the feature space $\phi(x_i), \phi(x_j)$. Popular kernels include the linear kernel

Eq. 32 $\boldsymbol{K(x_j, x_i) = \langle x_i, x_j \rangle}$

The polynomial kernel

Eq. 33 $\boldsymbol{K(x_j, x_i) = \langle x_i, x_j \rangle^n}$

And the Gaussian kernel, called radial basis function (RBF).

Eq. 34 $$K(x_j, x_i) = exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right)$$

The kernel trick, i.e., mapping the data point with a kernel in to another hiperplane and then accomplish the learning task on it, is a general strategy that can be incorporated into any learning algorithm that considers inner products among the input feature vectors. In principle, the only parameter used in SVM, besides the kernel function and his parameters, is a parameter C, which determines the trade-off between two conflicting goals: maximizing the margin and minimizing the training errors.

Neither ANNs nor SVMs are perfect**. SVMs are fast in training and guarantee a global optimum** if the kernel satisfies Mercer's condition, but requires an appropriate choice of kernel function. **ANNs are slow in training and can only guarantee local optima, but are robust to noise** and fast in classifying [86].

2.5.3.2 Adaptive network-based fuzzy inference system (ANFIS)

ANFIS is an algorithm of universal approximation composed by the supervised learning of the neuronal networks and functions based on linguistic expressions of fuzzy logic. it was proposed in 1993 by J.S.R. Jang [14]. ANFIS is based on fuzzy inference systems of the type Takagi-Sugeno [87].

Using the training data, ANFIS creates an inference fuzzy system consisting of one membership function input layer. The parameters on this layer are called antecedent parameters, they trained using backpropagation algorithm. For default authors use the same algorithm on the training of the output consequent parameters.

When least squared method is used to train the consequent parameters, the training algorithm is called hybrid. These algorithms allow to ANFIS to learn from the time series the characteristics on the target trend by means of mapping those trends in to regions where driven-fitted polynomials can be applied.

In order to introduce the ANFIS topology let us start with a simple fuzzy first order model Takagi-Sugeno. On this example, the system has two inputs and one output; each input also counts with two functions of membership by every variable (see **Figure 13**a). The equivalent architecture ANFIS of the first order inference system Takagi-Sugeno is shown in the **Figure 13**b.

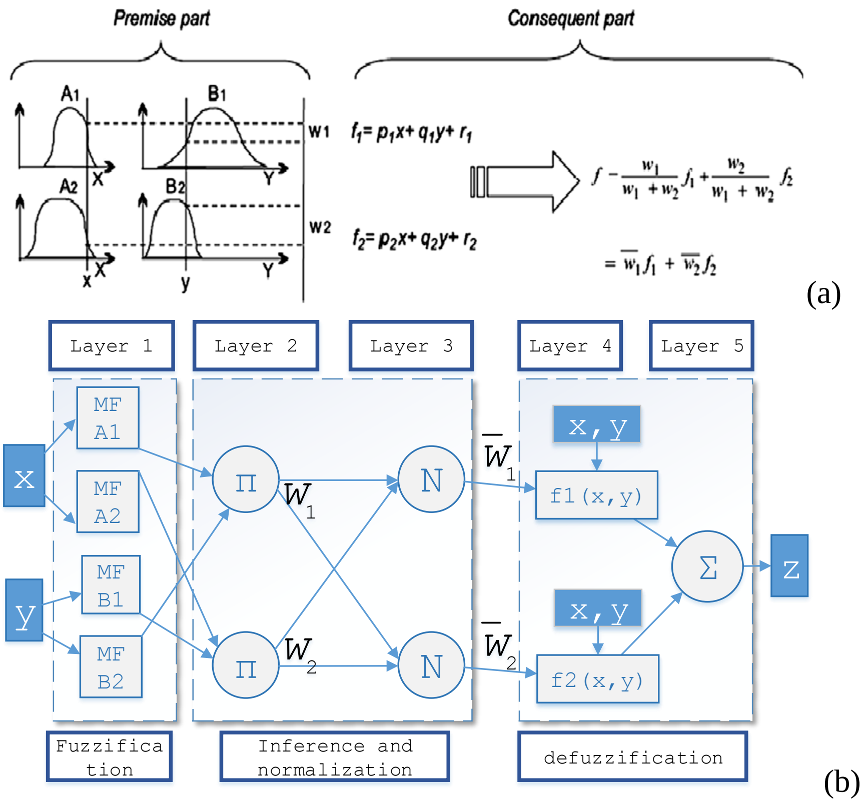


Figure 13. (a) If-then rules of the fuzzy model takagi-sugeno and mechanism of fuzzy reasoning, (b) ANFIS Structure.

The system consists of five layers, in which can exist several nodes. In order to explain the functioning of the system let us denote O_i^j as the exit for the i-th node in the layer j. In Layer 1, every node i is an adaptive node with node function.

Eq. 35 $\boldsymbol{O_i^1 = \mu A_i(x)}$ if $\boldsymbol{i = 1, 2}$ & $\boldsymbol{O_i^1 = \mu B_{i-2}(y)}$ if $\boldsymbol{i = 3, 4}$

Where *x* (o *y*) is the input to the *i*-th node and A_i, B_{i-2} is a linguistic label associated with this node. Membership functions for *A* and *B* are usually described by generalized bell functions, as for example:

Eq. 36 $\boldsymbol{\mu A_i(x) = 1/1 + \left|\frac{x - r_i}{p_i}\right|^{2q_i}}$

Where $\{p_i, q_i, r_i\}$ are the parameters to set, they control the width, shift, and shape/altitude of the membership function. Any continuous and piecewise differentiable functions, such as triangular-shaped membership functions, are also qualified candidates for node functions in this layer. In the layer 2, each node Π multiplies incoming signals and sends the product out. Here each node output represents the firing strength of a rule.

Eq. 37 $\boldsymbol{O_i^2 = W_i = \mu A_i(x) \mu B_i(x), \quad i = 1, 2}$

In layer 3, each node N computes the strength ratio of each i-th rule respect to the sum of all rules strengths. Outputs of this layer are called normalized firing strengths.

Eq. 38 $$\boldsymbol{O_i^3 = \overline{W}_i = \frac{W_i}{W_1+W_2}}, \qquad \boldsymbol{i = 1, 2}$$

In the layer 4, each node computes the contribution of the ith rule to overall output.

Eq. 39 $$\boldsymbol{O_i^4 = \overline{W}_i Z_i = \overline{W}_i (s_i x + t_i y + u_i)}, \qquad \boldsymbol{i = 1, 2}$$

Where $\overline{W}_i$ is the output of layer 3 and $\{\mathrm{s_i}, t_\mathrm{i}, u_\mathrm{i}\}$ are the coefficients to set. The parameters on this layer are referred to as consequent parameters. In layer 5, the single node Σ compute the final output as the sum of all incoming signals.

Eq. 40 $$\boldsymbol{O_i^5 = \sum_i \overline{W}_i Z_i = \frac{\sum_i W_i Z_i}{\sum_i W_i}}$$

Thus, an adaptive network is functionally equivalent to a sugeno-type fuzzy inference system. As was explaining at the beginning, this system obtains his expertise from the initial man-made configuration, where parameters such as the number of memberships functions per input, the number of rules to be interpreted according to their importance (this stablish the number of polynomials to use), the training method, and the number of epochs contribute to enhance the algorithm performance avoiding the over fitting.

The similarities between the soft mapping of polynomial regions according to inference rules makes to the ANFIS an algorithm comparable to an ensemble learning system, being capable to lead with the uncertainty of the signals easily; but leaving a big weakness floating around, his under capacity to generalize on the boundaries and outer limits of the variables analyzed.

2.5.3.3 Evolutionary computing

The adaptativeness of a model could be interpreted as the parameter fit made by the learning algorithm, or as the structural fit performed by an expert. The selection of the number of neurons, the membership function, the number of membership, and the order of the polynomials… all of those are structural decisions.

But, what if the structure together with the parameters were driven by a learning algorithm. On this section an experimental optimization method is presented, the method is able to construct models structural and parametrically fitted to the time series observed.

The two most common implementation of the genetic programming method are introduced: the Cartesian genetic programming, who constructs directed acyclic graphs based on pre-defied functions; and the Neural Cartesian genetic programming, who constructs directed acyclic graphs based on neurons.

2.5.3.3.1 *Cartesian genetic programing (CGP)*

CGP was originally proposed as a method for general genetic programming in [88]. While its name comes from its original application, evolving circuits on a two dimensional grid, modern CGP can represent any directed acyclic graph (DAG), and has been utilized in applications such as digital circuits [89], robot controllers [90], neural networks [91], image classifiers [92], and regression [93]–[95].

CGP represents DAGs using a linear genome of integer values. Each node in the DAG is encoded as a tuple of genes, with one gene specifying the function that the node applies to its inputs, and the remaining genes expressing where the node takes its inputs. Nodes can take input from either a problem input or any node preceding them in the linear genome.

Restricting connections in this way prevents the creation of cycles, while still allowing CGP to reuse values. This is in contrast to tree based GP, which must duplicate functionality anywhere the same value is needed. To complete the representation, a set of extra genes are included at the end of the genome to specify which nodes or input locations to use as function outputs.

As both output locations and information flow in the DAG are evolvable, often only a tiny fraction of the genome participates in creating the output values. These nodes are referred to as “active,” while the nodes not being used to create output values referred to as “inactive.” Inactive nodes allow for genetic drift, as individuals can be mutated without affecting their fitness.

These mutations can then be incorporated, as future mutations can change the DAG structure causing previously inactive nodes to become active. Previous work suggests CGP is most efficient when up to 95% of the genome is inactive [96], it may be a result of the parsimony on the error convergence due to the low mutation rate [97].

CGP uses very simple evolutionary mechanisms. The most common evolutionary strategy is $\mu + \lambda$. On this article, we have defined $\mu \leftarrow 1$ and $\lambda \leftarrow 4$; it means a total de 4 chromosomes to evolve, during each generation a single parent produces four offspring using mutation. The best offspring then compete with the parent replacing the parent if it is less fit.

This replacement strategy encourages neutral drift. In CGP mutation, each gene of each node could change randomly to some different valid value. For example, if a function gene is chosen for mutation, its new value is randomly chosen from all possible functions, excluding the gene’s current value.

As it is mentioned on [98], experience shows that in order to achieve a reasonably fast evolution, the mutation should arrange rate μ_r to be such that the number of genes chosen for mutation being a function of the genotype length. However, some authors [89]–[91], [98] prefer to fix the mutation rate to values between 1-10%. It is also recommended to maintain high mutation rates on small genotypes for fast evolution.

2.5.3.3.2 *Neural Cartesian genetic programing*

Neural Cartesian genetic programing (NCGP) is a natural driven evolution of CGP. It introduces the replacement of the node functions for neuronal operators, granting a similar look to the neuron functions on NN architectures. As well as the CGP, the driven force of the evolution lies on a genetic evolutionary mechanism which need time and patience for a moderate accuracy convergence.

The DAG evolved from the training process could be suitable for the exploited data, but for less computational effort a non-oversized NN could achieve better performances. Finally CGP and NCGP could be considered an excellent modelling strategy to be exploited for academic publications on the load forecasting, but they will never beat robust and well accepted methods such as NN or SVM.

NCGP can be considered an Evolutionary application towards the training of artificial neural networks. In the case of CGP it is referred to as Cartesian Genetic Programming of Artificial Neural Networks (CGPANN). NCGP makes use of the same training strategy presented on the CGP theory but exploit the weights on the node connections, which have existed on the CGP-Library but has been deliberately ignored for GCP implementations.

2.5.4 Hierarchical load forecasting and ensemble learning

One of the major concern related with the previously introduced computational intelligence techniques is their lack of detail on the track of components and effects that control the time series. Although they constitute the top tier techniques used on forecast they are not enough complex to generalize all the components on the dependent variable on to a single model.

Literature describe this problem as a **trade-off between generalization and piecewise specialization** [99]. The strategy followed for regression algorithms is to perform some type of decomposition of the search space on pieces to be regressed, limiting their inference to a finite number quadrants avoiding over map the regression surface.

But, regression errors are inherent to the learning algorithms so their description of the quadrants is not totally accurate. In order to complement their guess, more elements will be added and a strategy to weight and combine the elements must be defined.

Ensemble learning methods train multiple base-learners and then combine them using Boosting and Bagging as representatives. They ensembles is usually significantly more accurate than a single learner, and ensemble methods have already achieved great success in state-of-the art literature and many real-world tasks.

The origins of ensemble methods, referring to the basic idea of deploying multiple models, is fuzzy due to has been in use for a long time; however, it is clear that ensemble methods have become a hot topic since the 1990s [100], and various fields such as data mining, pattern recognition, machine learning, neural networks and statistics have explored ensemble methods from different aspects [21], [26], [27].

On this thesis the ensemble learning approaches has been introduced on a custom ensemble approach called **hierarchical load forecasting model**. It includes an ensemble learning approach made by combining different multi-resolution specialized forecasters, who are in turn containing another ensemble group of base learners.

2.5.4.1 Ensemble methods

An ensemble structure contains a number of learners called **base learners**. Base learners are usually generated from training data by a **base learning algorithm** which can be decision tree, neural network or other kinds of learning algorithms. Most ensemble methods use a single base learning algorithm to produce homogeneous base learners, i.e., learners of the same type, leading to **homogeneous ensembles**, but there are also some methods which use multiple learning algorithms to produce heterogeneous learners, i.e., learners of different types, leading to **heterogeneous ensembles** [24], [25].

The generalization ability best characteristic of an ensemble, it is often much stronger than that of base learners. In fact, Ensemble methods are able to boost **weak learners,** which are even just slightly better than random guess, as well as **strong learners** which can make very accurate predictions.

The current area of ensemble methods is a sum of three early threads of research; these are, combining classifiers, ensembles of weak learners and mixture of experts. **Combining classifiers** was vastly studied in the pattern recognition community, this thread was characterized by its focus on strong classifiers and the design of powerful combining rules to get stronger combined classifiers.

Ensembles of weak learners was mostly studied in the machine learning community. In this thread, researchers often work on weak learners and try to design powerful algorithms to boost the performance from weak to strong. This thread of work has led to the birth of famous ensemble methods such as AdaBoost, Bagging, etc., and theoretical understanding on why and how weak learners can be boosted to strong ones [101][102].

Mixture of experts was mostly studied in the neural networks community. In this thread, researchers generally consider a **divide-and-conquer** strategy, try to learn a mixture of parametric models jointly and use combining rules to get an overall solution. The basic approach to construct an ensemble consist on two steps, i.e., generating the base learners, and then combining them. To get a good ensemble, it is generally believed that the base learners should be as accurate as possible, and as diverse as possible.

2.5.4.1 Hierarchical load forecast

The novel approaches presented on this thesis are based on a hierarchical combination of ensemble learners, each one trained along a certain key multi-resolution element, i.e., one hour average load, wavelet decomposition level, etc. The base learners explored goes from so called weak ones, i.e., spline regressions, MARS regression, BLMS regression, regression trees, CGP, to strong ones, i.e., NCGP, NARXC, NARX, NN, ANFIS, SVM.

In this section we will introduce the ensemble theory [102] which provided the theoretical basis for the novel load forecasting architectures presented in the next chapter. The ensemble theory integrate the following design elements:

- **Ensemble learning methods** to train the base learners
- **Combination methods** for the base learners
- Ensemble diversity measurements
- **Pruning methods** and the identification of the optimal size of learners and
- **Clustering methods** for base learner specialization

2.5.4.1.1 *Ensemble learning methods*

Ensemble learning methods are closely related with sampling methodologies. Sampling refers to the selection of a subset of individuals from a population to estimate the characteristics of the entire population. The main advantage of these methods on ensemble modelling is the

statistical diversity founded at each subset. This is translated to a better capacity of generalization due to the diversity of weak learners.

On the other hand, sampling allows to create **probability density estimations** by the combination of the individual forecast made by the ensemble forecasters. The sampling method employed on this thesis is called **bagging**, it has been selected for the validation methods of the weak learners and the entire ensemble. Typical validation methods comprises the k–fold cross validation (K-FCV) and hold out validation (HOV). So, Lets describe the formal description of the ensemble learning approaches.

According to how the base learners are generated, there are two paradigms of ensemble methods, that is, sequential ensemble methods where the base learners are generated sequentially, with Ada-Boost as a representative, and parallel ensemble methods where the base learners are generated in parallel, with **Bagging** as a representative [29], [103][102].

The basic motivation of sequential methods is to exploit the dependence between the base learners, since the overall performance can be boosted in a residual-decreasing way. The basic motivation of parallel ensemble methods is **to exploit the independence between the base learners**, since the error can be reduced dramatically by combining independent base learners.

The name Bagging came from the abbreviation of Bootstrap AGGregatING. As the name implies, the two key ingredients of Bagging are bootstrap and aggregation. Bagging applies bootstrap sampling to obtain the data subsets to train and validate the base learners, generating a different distribution for each base learner.

Table 3. Bagging algorithm of a simple regression.

```
Input: Training data set $D^{Tr} = \{(x^{Tr}[1,\dots,m], y^{Tr}[1,\dots,m])\}$;
Base learner algorithm $\mathfrak{L}$ ;
Number of base learners $E$ ;
Process:
for e = 1,…,E:
    $h_e = \mathfrak{L}(D, D_{bs})\ or\ \mathfrak{L}(D_{ib}, D_{oob})$      % $(D, D_{bs})$ data set due to bootstrap distribution
end
Output: $GE_{oob}(x) = \frac{1}{E.m}\sum_{e=1}^{E}(h_e(x) - y)\ .\ \mathbb{I}(x \in D_{oob})$     % generalization error

$AVGMSE_{oob}(x) = \frac{1}{E.m}\sum_{e=1}^{E}(h_e(x) - y)^2\ .\ \mathbb{I}(x \in D_{oob})$ % average out of bag MSE error
```

The sampling process consist on generate bootstrap replicas, as much as weak learners, of the training data set (training part). **Each bootstrap replica is generated by sampling with replacement**, this create two new sets called “in-bag” and "out-of-bag" observations. **In-bag** observations is expected to have approximate 63% of unique samples, the rest 37% are duplicates.

This means that 37% of samples have been omitted. These are the so called **out-of-bag** observations. They are used to estimate the predictive accuracy of the entire ensemble or about the learners as **out-of-bag errors**, i.e., MSEOOB, RMSEOOB. **Out-of-bag average error** of the entire ensemble is often called generalization error, and can be obtained by average the errors of the base learners.

This is an attractive feature of bagging, inclusive without supply test data is possible obtain reliable estimates of the predictive power in the training process. **Bagging leads to "improvements for unstable procedures"**, which include, for example, artificial neural networks, classification and regression trees, and subset selection in linear regression [104]. Notice that bagging is not recommendable for base learners based on autoregressive structures such as NARXNN, NARNN, LRNN.

The out-of-bag samples can also be used to stablish the posterior probability of the prediction. Among the techniques employed to calculate the **conditional probability density function (CPDF)**, there are some ones based on prediction error approaches, such as Monte Carlo approaches. These ones execute several times a modelling algorithm in order to set the distribution of the prediction and stablish the prediction distribution at the same time a margin error zone.

On our case the due to the data set diversity has been gather on individual learner, and those have been trained using a stochastic sampling algorithm, every base-learner prediction could be accepted as a probable prediction. It means that gather all the predictions could leads to obtain the conditional distribution.

On this thesis, the **random sampling without replacement** is also explored in order to create in-bag and out-of-bag sets of observations, results are presented on Chapter 3.

2.5.4.1.1 *Combination methods*

Combination methods refers to the techniques employed to combine the set of base learners in order to achieve strong generalization ability. Combination methods are supported on three fundamental reasons: accuracy of the ensemble (statistical), finding the optima (computational), finding true hypothesis (representational).

The **statistical** issue refers to the mixture of base learners trained, each one representing a hypothesis. Due to the risk of choosing a wrong hypothesis for new data, its better combining them. The **computational** issue refers to the capacity of learners to get stuck on local optima, by combining the hypothesis (predictor functions) choose a wrong local minimum can be avoided. The **representation** issue presents a riddle, supposing that exist unknown hypothesis that cannot be generated form the limited training data, by a combination of hypothesis could be possible reach the unknown ones.

These three issues are among the most important factors for which the traditional learning approaches fail. A learning algorithm that suffers from the statistical issue is generally said to have a high "*variance*", a learning algorithm that suffers from the computational issue can be described as having a high "*computational variance*", and a learning algorithm that suffers from the representational issue is generally said to have a high "*bias*". Therefore, through combination, the variance as well as the bias of learning algorithms may be reduced [102].

The basic combination methods start with: simple/weighted averaging, majority/plurality/weighted/soft voting. But on this thesis we have been explored the **combination by learning** where the individual learners are combined by another learner, this technique is often called **Staking**.

2.5.4.1.1.1 Staking

The main idea is connect the base learners "*first-level learners*" by a meta-learner "*second-level learner*" [105]. The first-level learners are training using the original data set, and their predictions over new samples are then used as a new data set to train the second-level learner. The pseudo-code of a general stacking procedure is summarized below.

Table 4. General stacking procedure.

```
Input: Training data set $D^{Tr} = \{(x^{Tr}[1,\dots,m], y^{Tr}[1,\dots,m])\}$;
Number of base learners $E$ ;
First-level learning algorithm $\mathfrak{L}_E$ ;
Second-level learner algorithm $\mathfrak{L}$ ;
Process:
for e = 1,…,E:                    % Train a first-level learner by applying the
    $h_e = \mathfrak{L}_E(D_{Tr})$                      % first level learning algorithm
end
$D' = 0$;                         % Generate a data set for second-layer learner
for e = 1,…,E:
    for i = 1,…,m:
        $z_e[i] = h_e(x^{Val}[i])$;          % Prediction over the validation samples
    end
    $D' = D' \cup (z_E, y^{Val})$;
end
$h' = \mathfrak{L}(D')$;                         % Train the second-level learner h' using the $D'$
Output: $H(x) = h'(h_1(x^{Ts}),\dots,h_E(x^{Ts}))$   % Second level prediction over the test set
```

If the data used to train the first-level learner are also used to generate the new data set for training the second-level learner, there will be a high risk of overfitting. Hence, it is suggested cross validation or leave-one-out procedure. On this thesis, the validation data is used to train the second-learners.

2.5.4.1.1.2 Mixture of experts

Mixture of experts (ME) is an effective approach to train multiple learners as experts on a set of features. In contrast to typical approximation to ensemble learning, where individual learners are trained for the same problem, ME works in a divide-and-conquer strategy where a complex task is broken up into several simpler and smaller subtasks, and individual learners (called experts) are trained for different subtasks.

A function called **Gating** is usually employed to combine the experts by a system of weights. In the benchmarks of models presented on this thesis no gating function is presented because the 2nd learner is in enough capable to mix the 1st learners. The equation that define the procedure of the ME and the final output is presented below in a simple regression notation (f(x) = a*x+b).

Eq. 41 $$\boldsymbol{H(y|x;\Psi) = h'\left(h_{K,E}(y \mid x)\right) = \sum_{k=1}^{K}\sum_{e=1}^{E} w_{k,e} \cdot h_{k,e}(y \mid x;\theta_k) + b_{k,e}}$$

Where, $\boldsymbol{\Psi}$ includes the unknown parameters, the output y is a continuous variable. Given an input x, each local expert $\boldsymbol{h_{k,e}}$ tries to approximate the distribution of y and obtains a local output $\boldsymbol{h_{k,e}}(y \mid \boldsymbol{x};\boldsymbol{\theta_k})$, where $\boldsymbol{\theta_k}$ is the k-th feature used to build the ensemble. The 2nd level learner provides a set of coefficients: $\boldsymbol{w_{k,e}}$ that weigh the contributions of experts, and $\boldsymbol{b_{k,e}}$ is the parameter of bias. Thus, the final output of the ME is a weighted sum of all the local outputs produced by the experts.

2.5.4.1.2 *Ensemble diversity*

Ensemble diversity is known as the difference among the individual learners inserted by the ensemble learning strategy. As it was presented on the previous sections, diversity can be provided by **sampling methods**, **mixture of experts based on features**, or **mixture of experts based on clustering** [106]–[109]**.** Last one introduced on the next section.

But, the major obstacle for the diversity lies in the fact that the individual base learners are trained for the same task, and from the same training data, making them highly correlated. As a conclusion, a successful ensemble learning approach lies in achieving a good trade-off between the individual performance (learner's accuracy) and learner's diversity.

In fact the diversity is the holy grail of the field of ensemble learning, there is no well-accepted formal definition of it, but is crucial achieved it and measure it on this thesis.

2.5.4.1.2.1 Diversity generation

We already study some effective some heuristic mechanisms for diversity generation in the ensemble construction. On these, the basic idea is to inject some randomness into the learning process. On this section we will classify them based on their mechanics.

Data Sample Manipulation, consist on the training of individual learners from a re-sampled original data set, i.e. Bagging, AdaBoost. **Input Feature Manipulation**, consist on the description of the training data on a set of features. Different subsets of features, a.k.a. subspaces, provide different views on the data. Making the individual learners trained from different subspaces diverse. The searching for features could be based on inputs, i.e. day of the week, could be based on clustering, or be totally random.

Learning Parameter Manipulation, this mechanism consist on generate diverse individual learners by using different parameter settings for the base learning algorithm, i.e, different initial weights can be assigned to individual neural networks. **Output Representation Manipulation**, it consist on the manipulation of the base learners outputs in order to transform their nature as if other class of learner algorithm produce it, i.e., converts multi-class outputs to multivariate regression outputs to construct individual learners.

2.5.4.1.2.2 Error decomposition measures for ensemble methods

We could infer that the generalization error of an ensemble have a direct connection with the diversity concept, or a term related to. On this section two famous error decomposition schemes for ensemble methods will be studied, known as, the error-ambiguity decomposition and the bias-variance decomposition.

2.5.4.1.2.3 Error-ambiguity decomposition

It measure the ability of an ensemble and their learners to approximate a real function that describe the target, and it final prediction. A weighted ensemble average is defined by

Eq. 42 $\boldsymbol{H(x) = \sum_{e=1}^{E} w_e \cdot h_e(x)}$

Then, the error of the ensemble $\boldsymbol{H}$ and each one of the individual learner $\boldsymbol{h_e}$ and the, are respectively

Eq. 43 $err(H|x) = \big(y(x) - H(x)\big)^2$
Eq. 44 $\boldsymbol{err(h_e|x) = (y(x) - h_e(x))^2}$

Given an instance x, the ambiguity term measures the disagreement among the individual learners on instance x. the ambiguity of the individual learner he is defined as

Eq. 45 $\boldsymbol{ambi(h_e|x) = (h_e(x) - H(x))^2}$

And the average ambiguity of the ensemble is

Eq. 46 $\boldsymbol{\overline{ambı}(h|x) = \sum_{e=1}^{E} w_e \cdot ambi(h_e|x)}$

Then, the average ambiguity, which is the **variance** of the output over the ensemble is

Eq. 47 $\boldsymbol{\overline{ambı}(h|x) = \sum_{e=1}^{E} w_i \cdot err(h_e|x) - err(H|x) = \overline{err}(h|x) - \ err(H|x)}$

Being

Eq. 48 $\boldsymbol{\overline{err}(h|x) = \sum_{e=1}^{E} w_i \cdot err(h_e|x)}$

Where w is the weighted average of the individual errors. Based on the above notations, we can get the error-ambiguity decomposition. The term $\boldsymbol{err(H|}x)$ is the ensemble error.

Eq. 49 $\boldsymbol{err(H|x) = \overline{err}(h|x) - \overline{ambı}(h|x)}$

By averaging over the input distribution $P(x)$, and implicitly over the target outputs y(x), one obtain the ensemble **generalization error**.

Eq. 50 $err = \overline{err} - \overline{ambi}$

Notice that **the weights are simply impossible to get** on non-linear models, for that reason on the section we will introduce a more general notation that consider unitary weights.

2.5.4.1.2.4 Bias-variance decomposition

This decomposition is an important tool for analyzing the performance of ensemble methods and learning algorithms in general. Given a learning target and the size of training set, it divides the generalization error of a learner into three components, i.e., **intrinsic noise**, **bias** and **variance**.

The **intrinsic noise** is a lower bound on the expected error of any learning algorithm on the target; the **bias** measures how closely the average estimate of the learning algorithm is able to approximate the target; the **variance** measures how much the estimate of the learning approach fluctuates for different training sets of the same size.

Since the intrinsic noise is difficult to estimate, it is often subsumed into the bias term. Thus, the generalization error consist into the bias term which describes the error of the learner in expectation, and the variance term which reflects the sensitivity of the learner to variations in the training samples.

Let’s denote the target y(x) as y, and h denote the learner. For squared loss, the decomposition is

Eq. 51 $$err(h) = \mathbb{E}[(h-y)^2] = (\mathbb{E}[h]-y)^2 + \mathbb{E}[(h-\mathbb{E}[h])^2]$$
$$= bias(h)^2 + variance(h)$$

Where the bias and variance of the learner h is respectively

Eq. 52 $bias(h) = \mathbb{E}[h] - y;$ $\qquad variance(h) = \mathbb{E}(h-\mathbb{E}[h])^2$

For an ensemble of E learners, the decomposition can be further expanded, yielding the bias-variance-covariance decomposition. Without loss of generality, suppose that the individual learners are combined with equal weights. The averaged bias, averaged variance, and averaged covariance of the individual learners are defined respectively as

Eq. 53 $\overline{bias}(H) = \frac{1}{E}\sum_{e=1}^{E}(\mathbb{E}[h_e] - y)$

Eq. 54 $\overline{variance}(H) = \frac{1}{E}\sum_{e=1}^{E}(h_e - \mathbb{E}[h_e])^2$

Eq. 55 $\overline{covariance}(H) = \frac{1}{E(E-1)}\sum_{e=1}^{E}\sum_{j=1;j\neq e}^{E}(h_e - \mathbb{E}[h_e])(h_j - \mathbb{E}[h_j])$

Then, the bias-variance-covariance decomposition of squared error of ensemble is

Eq. 56 $$\boldsymbol{err(H) = \overline{bias}^2(H) + \frac{1}{E}\overline{variance}(H) + \left(1 - \frac{1}{E}\right)\overline{covariance}(H)}$$

It shows that the squared error of the ensemble depends heavily on the covariance term, which models the correlation between the individual learners. The smaller the covariance, the better the ensemble. It is obvious that if all the learners make similar errors, the covariance will be large, and therefore it is preferred that the individual learners make different errors.

Thus, the covariance term shows that the diversity is important for ensemble performance. Notice that the bias and variance terms are constrained to be positive, while the covariance term can be negative. As you will notice there is a connection between the error ambiguity decomposition and the bias-variance-covariance decomposition. For simplicity, assume that the individual learners are combined with equal weights.

Eq. 57 $$\boldsymbol{\overline{bias^2}(H) + \frac{1}{E}\overline{variance}(H) + \left(1 - \frac{1}{E}\right)\overline{covariance}(H) = \overline{err}(H) - \overline{ambi}(H)}$$

Eq. 58 $$\overline{err}(H) = \frac{1}{E}\sum_{e=1}^{E}(h_e - y)^2 = \overline{bias}^2(H) + \overline{variance}(H)$$

Eq. 59 $$\overline{ambi}(H) = \frac{1}{E}\sum_{e=1}^{E}(h_e - H)^2 = \overline{variance}(H) - \frac{1}{E}\overline{variance}(H) - \left(1 - \frac{1}{E}\right)\overline{covariance}(H)$$

Thus, we can see that the term variance appears in both the averaged squared error term and the average ambiguity term, and it cancels out if we subtract the ambiguity from the error term. Moreover, the fact that the term variance appears in both err and ambi terms indicates that it is hard to maximize the ambiguity term without affecting the bias term, implying that generating diverse learners is a challenging problem.

2.5.4.1.1 *Ensemble pruning*

Ensemble pruning tries to select a subset of individual learners, rather than combining all of them to comprise the ensemble. Among the advantages obtained are: smaller sizes, increase of the efficiency, better generalization error.

Originally, ensemble pruning was defined for the instance where the individual learners have already been generated, and no more individual learners will be generated from training data during the pruning process. Although, recent literature have extended it to all steps of ensemble construction.

Indeed, the central problem of ensemble pruning research is how to design practical algorithms leading to smaller ensembles without sacrificing or even improving the generalization performance contrasting to all-member ensembles. Ensemble pruning methods can be classified into three categories:

- **Ordering-based pruning**. They try to order the individual learners according to some criterion, and only the learners in the front-part will be put into the final ensemble.
- **Clustering-based pruning**. Those methods try to identify a number of representative prototype individual learners, from groups created by clustering, in order to constitute the final ensemble.
- **Optimization-based pruning**. Those methods formulate the ensemble pruning problem as an optimization problem which aims to find the subset of individual learners that maximizes or minimizes an objective related to the generalization ability of the final ensemble.

It is obvious that the boundaries between different categories are not crisp, and there are methods that can be put into more than one category.

2.5.4.1.2 *Clustering methods*

We already spoke about the possibility to use clustering to create diversity inside the ensemble, it could be considered a data sample manipulation. Clustering consist on the classification of the training data by grouping them into clusters based on features of the inherent data structure.

Table 5. General clustering procedure on ensemble algorithms.

```
Input: Training data set $D^{Tr} = \{(x^{Tr}[1,\dots,m], y^{Tr}[1,\dots,m])\}$;
Number of base learners $E$ ;
1st level learning algorithms $\mathfrak{L}_E$ ;
2nd level learning algorithm $\mathfrak{L}$ ;
Set of features to train expert $\theta_K$ ;
Number of features/subspaces/clusters $K$;
Process:
$\theta_K$ = Classify($D^{Tr}, K$);          % Classify the data set
for k = 1,…,K:                              % Select the feature
    for e = 1,…,E:                          % Select a 1st learner
        $f_{k,e} = RS(D^{Tr}|\theta_k)$      % Random sampling over the class
        $h_{k,e} = \mathfrak{L}_E(f_{k,e})$      % Train a 1st level learner for the set f
    end
end
$D' = 0$;                                   % Generate a data set for second-layer learner
for k = 1,…,K:                              % Select the feature
    for e = 1,…,E:
        for i = 1,…,m:
            $z_{k,e}[i] = h_{k,e}(x^{Val}[i])$;  % Prediction over the validation samples
        end
        $D' = D' \cup (z_{k,e}, y^{Val})$;
    end
end
$h' = \mathfrak{L}(D')$;                     % Train the second-level learner h' using the $D'$
Output: $H(x) = h'(h_{1,1}(x^{Ts}), \dots, h_{K,E}(x^{Ts}))$     % Second level prediction over the test set
```

Clustering can be used as a stand-alone exploratory tool to gain insights on the nature of the data, and it can also be used as a preprocessing stage to facilitate subsequent learning tasks. A lot of clustering methods have been developed and various taxonomies can be defined from different perspectives clustering methods could be divided on the following five categories.

Partitioning Methods. A partitioning method organizes the data space into k partitions by optimizing an objective partitioning criterion. The most well-known partitioning method is **k-means clustering** which optimizes the square-error criterion.

Hierarchical Methods. A hierarchical method creates a hierarchy of clusters on the data space at various granular levels, where a specific clustering can be obtained by thresholding the hierarchy at a specified level of granule.

Density-Based Methods. A density-based method constructs clusters on the data space based on the notion of density, where regions of instances with high density are regarded as clusters which are separated by regions of low density. **DBSCAN** is a representative density-based clustering method, which characterizes the density of the data space with a pair of parameters (radius, MinPts).

Grid-Based Methods. A grid-based method quantizes the data space into a finite number of cells forming a grid-structure, where the quantization process is usually performed in a multi-resolution style. **STING** is a representative grid-based method, which divides the data space into a number of rectangular cells.

Each cell stores statistical information of the instances falling into this cell, such as count, mean, standard deviation, minimum, maximum, type of distribution, etc. There are several levels of rectangular cells, each corresponding to a different level of resolution. Here, each cell at a higher level is partitioned into a number of cells at the next lower level, and statistical information of higher-level cells can be easily inferred from its lower-level cells with simple operations such as elementary algebraic calculations.

Model-Based Methods. A model-based method assumes a mathematical model characterizing the properties of the data set, where the clusters are formed to optimize the fit between the data and the underlying model. The most famous model-based method is **GMM-based clustering**, which works by utilizing the **Gaussian Mixture Model** (GMM).

CHAPTER

Study and Contributions to Load Forecasting

This chapter presents the contributions on the load forecasting techniques brought by the research made on this thesis. The contents follow a timeline directed by the problem statement. It starts with the implementation of single forecast algorithms, move through implementations of novel ensemble learning algorithms, and finish the compilation of the best strategies on an ensemble predictor.

Contents

3.1 Introduction

On the chapter 2 we sketched a variety of areas for which the load forecasting techniques are widely used, a complete description of the characteristics that defines each component of the load profile, the state of the art on the preprocessing techniques on the area, and the framework of algorithms that compose the techniques used for energy prediction on the literature.

On this chapter, we will introduce the contributions on the forecasting field proposed on this thesis. The contributions have been classified on three main categories attending to the standard classification of forecasting on temporal series.

The following sections will start with a machine learning approach, an ANFIS combined with a robust multi-resolution techniques in order to obtain an improvement variance error without the decrease the generalization error.

Continues with the introduction of a novel evolutive method (CGP) as result of the search for a method with high generalization and adaptiveness. On third section, approaches based on ensemble learning are presented. Although this technique is part of the machine learning algorithms, we have create a special section due the extensive develop of novel methods on this area.

3.2 Load forecasting algorithm based on expert systems and multi-resolution analysis

The implementation presented on this section makes use of an algorithm that possesses strong learning rules and human reasoning inspired rules; we are speaking about the expert systems. As was stated on the chapter 2, implementations based on non-linear models obtain better results on forecasting due their aptitude to shape non-trivial and non-trivial relations.

This relationship exists between future consumption and factors that produce it (climatic conditions, goals of production, labor at calendar, etc.), and precisely this has motivated the risen of hybrid techniques on the field of expert systems; the complexity of the forecast in real cases. This is the case of combination between ANFIS and pre-filter and optimization components [15], [110], [111].

Following this trend, we introduce an approximation to a STLF on the user-side to be presented from here onward. It employs an ANFIS algorithm as modelling core and the SWT as preprocessing component; the implementation is carried out based on the general electricity consumption of the car manufacturer.

3.2.1 Theoretical approach

The objective of the STLF algorithm proposed is introduce the wavelet preprocessing in order to decompose the target variables into a number of approximations and details of original signal. This would bring some useful insight on the construction of analytical relationships inside of the modelling algorithm.

The load forecasting process implemented over the industrial scenario, have been tested using the inputs and target addressed on the **Figure 14**. The inputs variables created has been based on temporal information of the series, weather information of the consumption location, and delayed versions of the target.

Decomposed signals are fed into an ANFIS algorithm, and finally when the model converges to the minimal error between the information of training and checking data, the best model is obtained. **Figure 14** present the flow diagram of the algorithm implement on the proposed approach.

On the **Step 0**, the drivers and target are gathered on a matrix. The drivers employed as predictors of the total electricity demanded by the car manufacturing company where: Temperature (Temp) in Celsius; Hour of the day (Hour), a integer number between 0 and 23, Day of the week (WD), a integer number between 1-7; and labor day (LD), a Boolean.

Once the input matrix is created, the data is cleaned from out layers and the remaining gaps are filled on the **Step 1**. Next, the endogenous variables such as one day (Elec_1Dd) and one week (Elec_1Wd) delayed consumptions are created in base of the target consumption. These variables are added to the modelling dataset. Then, the dataset is partitioned in three sets training (60%), tes (30%), and validation (10%). The normalization of the train-test data set to normal distribution using z-is carried out as final step. The original statistical properties for a future reuse.

On the **Step 2,** inputs are decomposed using the SWT. The endogenous variables are divided in six levels of approximations (A1-A6) and details (D1-D6); the temperature is divided in three levels; for this decomposition, the wavelet function Db10 is used. In this stage is applied thresholding to remove strong variations in the coefficients, which can cause noise in the reconstruction. The approximations of the original signals will be versions more smoothed of the original ones, whereas the details allow ANFIS make thin adjustments [112].

Later, the process of reconstruction applying ISWT is effectuated. Selecting A5, D1 and D5 to describe the delayed consumptions and A3, D1 and D3 to describe temperature. These signals retain most of the information of the original signals according to authors [15], [113]. The original Temperature and endogenous variables are replaced by their filter versions.

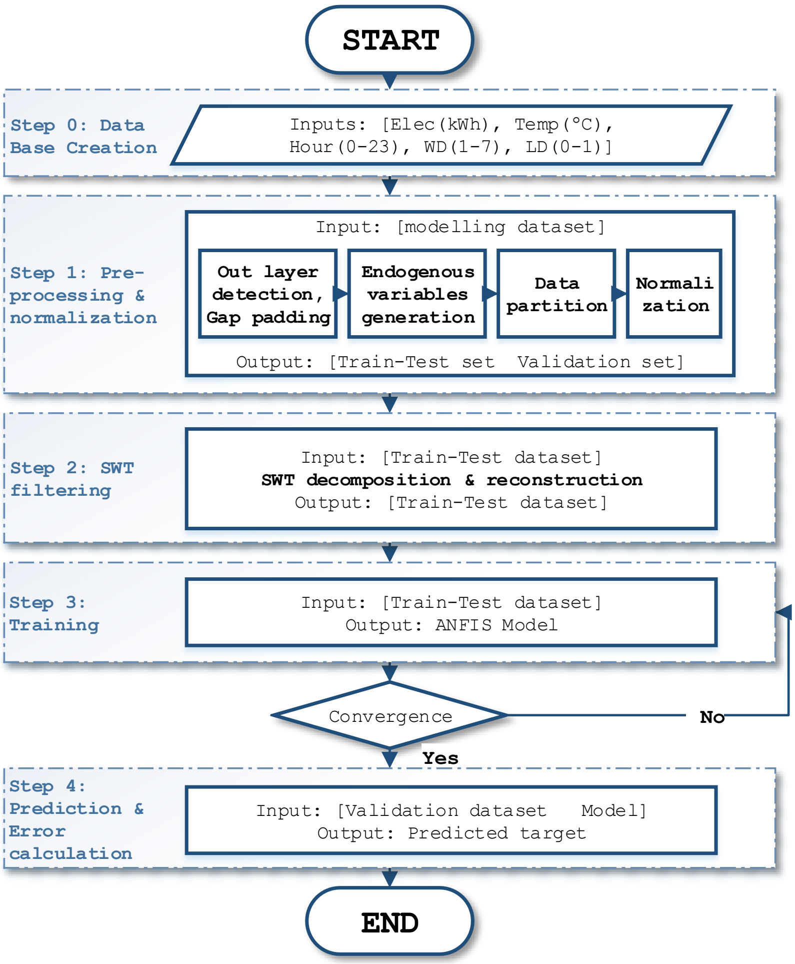

START
Step 0: Data Base Creation
Inputs: [Elec(kWh), Temp(°C), Hour(0-23), WD(1-7), LD(0-1)]
Step 1: Pre-processing & normalization
Input: [modelling dataset]
Out layer detection, Gap padding
Endogenous variables generation
Data partition
Normali zation
Output: [Train-Test set Validation set]
Step 2: SWT filtering
Input: [Train-Test dataset]
SWT decomposition & reconstruction
Output: [Train-Test dataset]
Step 3: Training
Input: [Train-Test dataset]
Output: ANFIS Model
Convergence
No
Yes
Step 4: Prediction & Error calculation
Input: [Validation dataset Model]
Output: Predicted target
END

Figure 14. Flow diagram of the Load forecasting algorithm based on expert systems and multi-resolution analysis.

On the **Step 3**, the elements of the modelling algorithm are set. In this case, two membership functions are selected by every input and their parameters are initialized using clustering. The ANFIS is trained using back propagation method. The maximum number of epochs to reach the minimum generalization error is fixed to 200 due to the fast convergence of the learning algorithm. If the convergence is not reach, the training task will be restarted.

Once the model is obtained, the validation set is employ to obtain the prediction on the **Step 4**. In this step, the validation set is normalized using the statistical information of the train-test dataset, and feed on to the model. In order to evaluate the precision in the forecasting of the load profile, the root means square error (RMSE) and the mean absolute percentage error (MAPE) are considered.

3.2.1.1 Wavelet transform (WT)

WT belong to time-frequency transforms, it decomposes the origin signal on a family of functions with zero average called wavelets. These are created from time-shift and time-expand on the function base called wavelet mother. WT's can be divided in two categories: discrete wavelet transforms (DWT) and continuous wavelet transforms (CWT). Any of the wavelet filtered signals (W(a,b)), of an original signal f(x), using the wavelet mother ϕ(x) is given by:

Eq. 60 $$\boldsymbol{W(a,b) = \frac{1}{\sqrt{a}}\int_{-\infty}^{+\infty} f(x)\phi\left(\frac{x-b}{a}\right)dx}$$

Where a=f0⁄f and f0 is the central frequency of function wavelet ϕ(x), a is known as scale factor and determines the width of ϕ(x); b is parameter of shift and determines central position of ϕ(x). The CWT needs a significant capacity of calculation, represented in time and resources. It happens with the majority of transformations in continuous time, on the other hand, DWT reduces significantly these disadvantages and is much simpler to implement. the DWT is defined as

Eq. 61 $$\boldsymbol{W(m,n) = \frac{1}{\sqrt{a}}\sum_{t=0}^{T-1} f(x)\phi\left(\frac{x-b}{a}\right)}$$

Where T is the length of the signal f(x). Shift and scale parameters are functions, on powers of 2, of the integer variables m and n (a=2^m, b=n2^m). It can be observed how the signal in discrete terms is linear combination of shifted and scaled functions. It makes us resemble the known time-frequency transformations, but do not get confused; wavelet transform has a non-linear axis on frequency. In fact, it gives more importance to a few frequencies instead of others.

Rapid algorithm DWT is based on Mallat's development [114]. It implementation use decimation to facilitate the computational process, but it causes losses of information in tasks of forecasting. In order to settle the problem we have been always use not decimated DWT known as SWT. In **Figure 15** the multilevel process, presented as a successive decomposition on the original sign and each of his approximations, proposed by Mallat is observed.

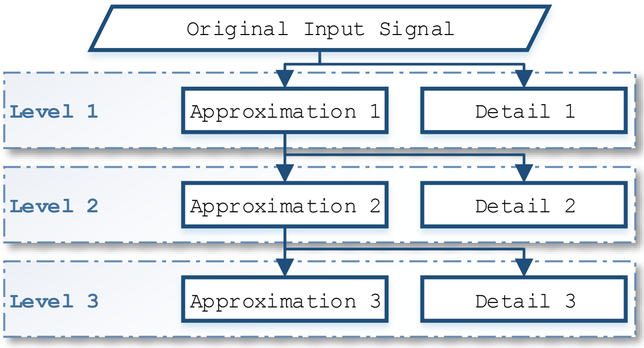


Figure 15. Process of multilevel decomposition realized by the Wavelet transforms.

In this thesis has been selected like wavelet mother ϕ(x) the Daubechies scaling function of order 10. This wavelet offers an appropriate balance between wavelength and smoothness. That allows analyze to simple sight the consumption profile to realize short-term forecast [15], [112]. It is possible due that approximations of the signal present a polynomial approximation of minor degree that the original ones.

3.2.2 Experimental results

The errors obtained for the current implementation are presented in comparison with a simple ANFIS implementation on the **Table 6**. Despite to appear being low, these error errors are significantly high when are measure on kWh.

Table 6. Error the Load forecasting algorithm based on expert systems and multi-resolution analysis.

	RMSE (%)	**MAPE (%)**
Current implementation	5.3	11.56
ANFIS	6.5	13.54

However, taking into account that the forecast was realized in the user-side, which is characterized by a fast fluctuation of load, the results are suitable for be used in an iEMS. It is reasonable to conclude that the necessary polynomials to predict the load profile in the user-side have to be more than polynomials to predict in the generator-side.

The database was gathered between 1/06/2010 and 25/03/2011. Numerical results of the forecast are showed on the **Figure 16** to **Figure 17**. These graphs present the predicted load profiles for weeks located in summer, autumn, winter, and spring respectively.

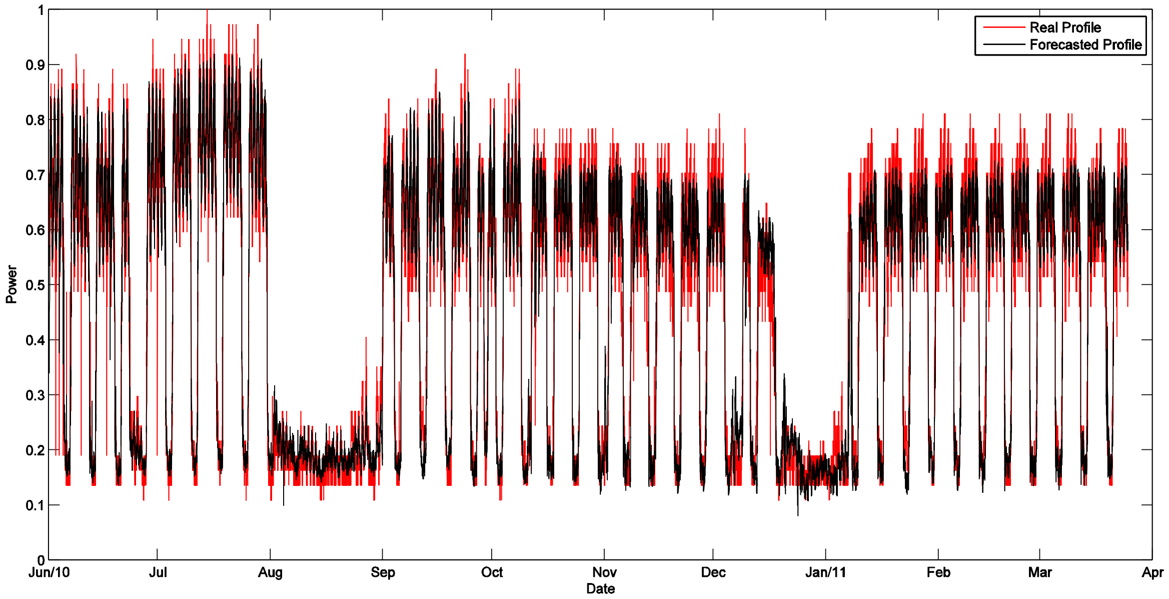
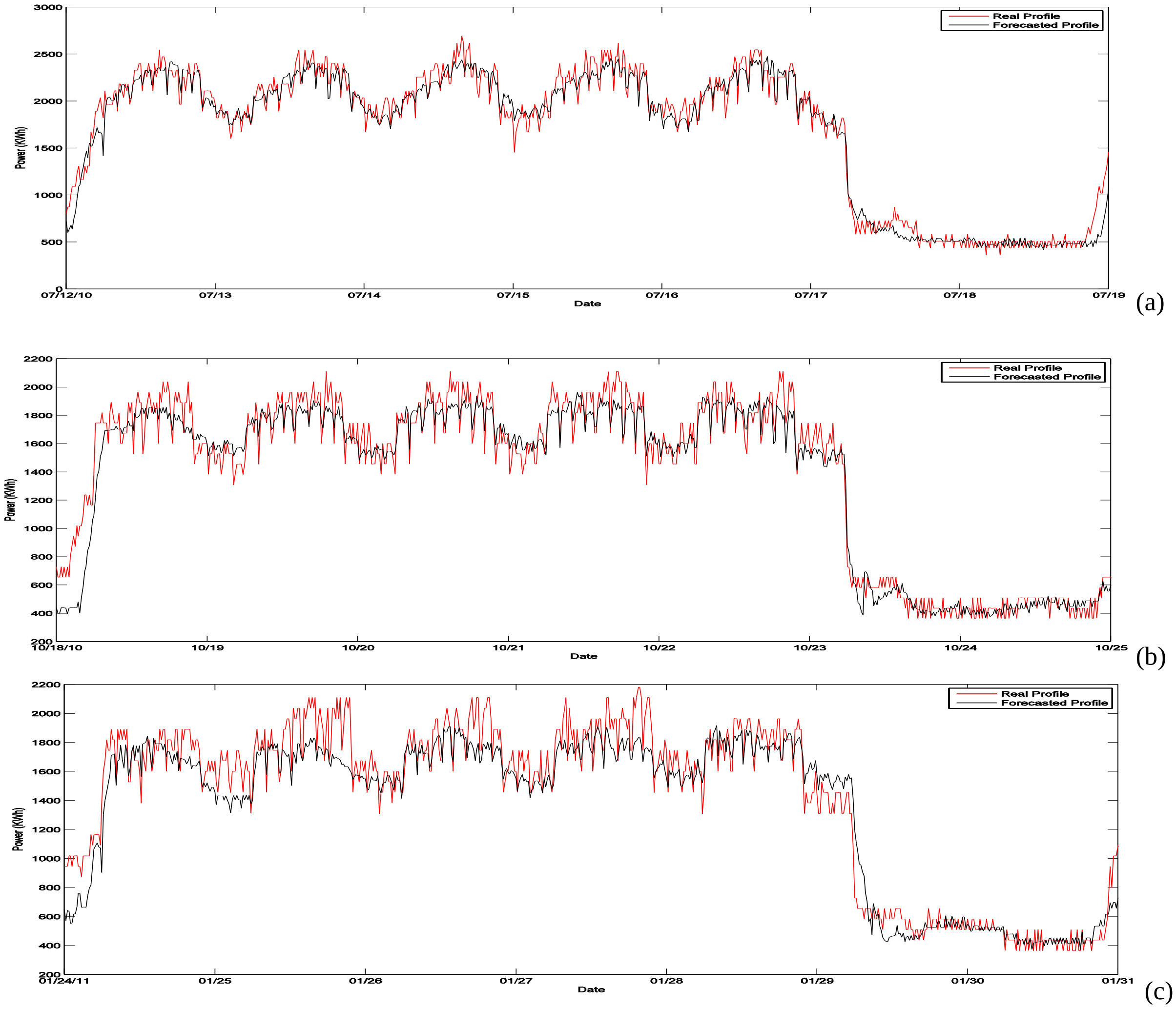


Figure 16. Annual consumption profile, original (red) and forecasted by the proposed Load forecasting algorithm based on expert systems and multi-resolution analysis (black).



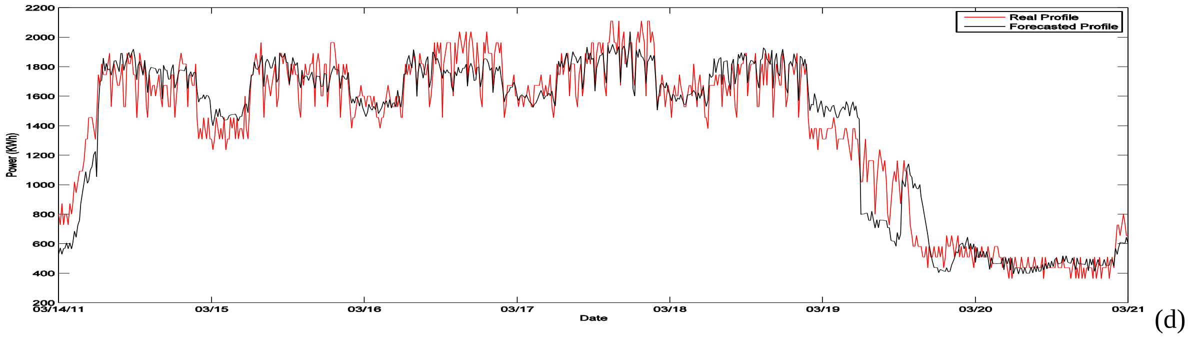


Figure 17. Weekly profile of power consumption, original (red) and forecasted by the proposed STLF approach (black) for: **(a)** the summer week 07/12/10 to 07/19/10, **(b)** the autumn week 10/18/10 to 10/25/10, **(c)** the winter week 01/24/11 to 01/31/11, **(d)** the spring week 03/14/11 to 03/21/11.

3.2.3 Discussion and conclusion

In this section, we proposed an STLF algorithm based on expert systems and multiresolution analysis. The multiresolution element refers to the stationary wavelet transform, employed to filter and detect the cyclical information on the target profile. This stage contributes to planning the incoming steps of the modelling strategy.

The families of wavelet functions implemented, Daubechies, are chosen because are characterized by maximum number of vanishing moments for a given support, which improves its ability to handle the information in load data. The performance of the proposed approach is compared with an ANFIS, and is found to be superior. This proves the effectiveness of proposed method on the treatment of the input variables characteristic as key on the reduction of the generalization error.

Another contribution on the effectiveness of the algorithm is made by the combination of the filter key drivers and the fuzzy reasoning – polynomial evaluation performed by the ANFIS. The first one allows to the second one to follow the trend of the profile and not the finest details which contribute to an over fitting of the model.

In presence of the obtained results with our current approach, we conclude that the STLF scheme proposed is suitable to be implemented on an EMS due to it meets with the requirement of enhance the global prediction accuracy by the mitigation of the uncertainty components carried out by the drivers and source of inaccurate forecasting's. In terms of autonomy the preprocessing part constitute a hard obstacle due the selection of the parameters selected on the filtering are part of the human expertise and must be corrected with the experience.

3.3 Load forecasting algorithm based on Genetic Cartesian Programing.

Most of the papers related with genetic programming found a niche on other application fields, being the load forecasting applications still at a congress level due to the experimental soul of the algorithm implementation. This constitutes a window of opportunity for reseach on the field.

Published genetic programming methodologies have been attempted to tackle the load forecasting problem from an evolutive point of view [115]. These novel model structures are known for their integration of pieces of functional code, assembled using genetic programming.

This section introduce an enhancement over a type of evolutionary programming algorithm called Cartesian Genetic Programming (CGP). The contributions are made in order to develop a model with one day ahead load forecasting capabilities. Regular CGP implementations fix its mutation ratio to low values causing a low convergence over a big number of generations [90], [115]–[117]. Also is known that CGP model produce overfitted models due to the use of a unique data set on the training phase.

The main contribution described on this section is the proposal of a novel approach for the efficient evolution of models based on CGP. Our approach is applied on a regression problem, specifically short-term load forecasting. In alignment to the previous statement, we will test the following hypotheses:

- A fitness function obtained from train-validation data sets can reduce the generalization error on the models.
- A mutation rate method based on the generation epochs and MAPE of the train set could reduce the number of generations necessary to converge at a low forecast error.

3.3.1 Theoretical approach

In order to test our hypotheses, we have performed a multistep procedure as shown in the following figure. The procedure starts with the construction of the data array used by the modelling algorithms.

The data array is randomly sorted in order to cancel the effects of the seasonality over the train and validation sets. Models are obtained from our proposed strategy and three relevant modeling methods and their forecast errors are calculated. The model errors are gathered over two hundred runs and the accuracy of the proposed strategy is compared with the others modelling methods.

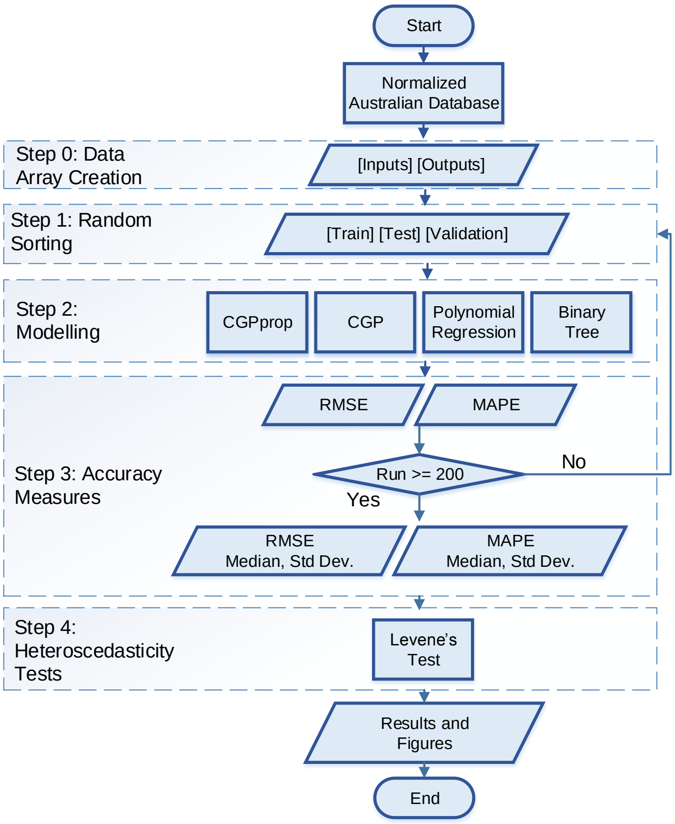


Figure 18. Flow chart of the procedure implemented in order to test the suitability of a load forecasting based on CGP.

Step 0-1: **Data array creation & Random sorting**. Based on the data set of ACT-NSW regions, the data array used for the one day ahead forecast consist of the followings variables: dry bulb temperature, dew point, hour, day of week, holidays, previous day demand, previous week demand, and current demand. A random sort is applied over the data. The models use pieces of the data array divided into train (60%), validation (30%) and test (10%) sets.

Step 2: **Modeling**. Based on the train-validation sets, the models learn the dynamics of the electric consumption. Each model adjusts its internal parameters using the train samples and verifies its accuracy over the validation ones.

We have implemented a novel evolutive strategy in order to enhance the basic form of the CGP algorithm. Our approach is compared with the basic implementation of the CGP algorithm [93], [118] obtained from “http://cgplibrary.co.uk”, a polynomic regression, a binary decision tree, a neural network and an ANFIS model. The characteristics of the models have been presented on the modeling algorithms section.

Step 3: **Accuracy measures**. Once the models are trained, we proceed to validate their accuracy evaluating them over the test partition. The Root Mean Square Error (RMSE) and MAPE have been calculated on each running. After two hundred executions, the mean and standard deviation of the error measures are calculated. This provides a general view of the error variance and the performance of the learning algorithm.

Step4: **Results analysis**. The distribution of the MAPE is studied using the Levene's test [19, 20]. It analyzes the observed data in order to compute the difference of the error distribution shape with respect to a Gaussian distribution, obtaining the p-value from these discrepancies. P-value allows to conclude if the convergence of our modelling approach is better than the basic CGP implementation in terms of a narrow distributed error.

3.3.1.1 Modelling Algorithms

In order to model the ACT-NSW electricity consumption a novel evolutive strategy based on CGP is implemented. This strategy is called: Fast double checked Cartesian genetic programming (FCD-CGP).

The proposed CGP implementation is based on the hypothesis that the generalization error can be reduced using a strategy that check the train and validation errors. This strategy provides robustness to the CGP method due to its double check on pairs of randomly sorted sets. The procedures described on this sections could be easily replicated by introducing the correspondent modifications on the original CGP implementation.

```
procedure New parent selection
   for all $i \in Chromosome$ do
      $fitTr \leftarrow fitnessFunctionOnTrainSet[Chromosome(i)]$
      $fitVa \leftarrow fitnessFunctionOnValidationSet[Chromosome(i)]$
      if $fitTr < previousFitTr$ then
         if $fitVa < previousFitVa$ then
            $bestFitTr \leftarrow fitTr$
            $bestFitVa \leftarrow fitVa$
            $bestGeneration \leftarrow Generation$
            $bestChromosome \leftarrow Chromosome(i)$
         end if
      end if
   end for
end procedure

procedure Fitness function
$error \leftarrow 0$
   for all $i \in Samples$ do
      $forecastOut \leftarrow evalChromosome(inputs(i))$
      $error \leftarrow error + \|Output(i) - forecastOut\|/Output(i)$
   end for
$error \leftarrow error/Samples$
end procedure
```

Figure 19. Algorithm to select a new parent.

The new parent selection algorithm performs the evaluation of the fitness function over N chromosomes using the train and validation sets as shown in previous figure. Only if a chromosome is able to obtain lower fitness in comparison with its parent, the new chromosome replaces the parent. Variables such as, the fitness value over the train and validation set and the generation number are saved.

The fitness function showed on previous figure is based on MAPE and is shared with all CGP implementations presented on this article. The mutation rate is calculated on the generations where an improvement of the fitness train error occurs, and persists until another calculation is made.

The follow figure shows the equation used to calculate the mutation rate. It depends on the fitness obtained from the train set and the generations where a new parent is found.

```
procedure Mutation rate funtion
    $mutationRate \leftarrow bestFitTr + \Delta bestFitTr + log(\Delta bestGeneration)/10$
end procedure
```

Figure 20. Algorithm to calculate the mutation rate at each generation.

The concept behind the mutation rate equation are meant to provide an inertia effect over the random mutation. The derivative term over the train fitness error adds the capacity of use the error momentum to force an adaptive fast convergence. The derivative term over the best generations provide a supplementary momentum too, especially when the errors reach a steady state.

3.3.1.2 Study of the variance on the forecasting error

Due to the genetic algorithm produce different models on each run, the estimation of the convergence accuracy must be evaluated. Some tests are introduced to validate the normal distribution on the forecast errors for the CGP modelling algorithms. These tests has been performed in order to measure the degree of disturbance introduced by our evolutive strategy on the normal error convergence of the CGP algorithm. In order to measure the disturbance degree, a large number of executions (200) over the model algorithms have been carried out. On the other hand, the test has only been applied over the CGP implementations as well as over the test set.

The heteroskedasticity tests indicates the variability of the variances for a group of data sets [119]. Levene's test is used for checking if k samples present heterogeneity of variances (heteroscedasticity) or homogeneity (homoscedasticity) [120]. On this test, the null hypothesis is the normality condition on the variances [121].

The homoscedasticity provides a measure of the goodness of the learning algorithm used to fix the internal model parameters [122]. In order to test our evolutive strategy, the random sort over the train, validation and test sets has been performed [123].

3.3.2 Experimental result

In order to demonstrate the convergence qualities of the proposed evolutive strategy on a CGP algorithm against other relevant algorithms, a normalized database based on the Australian electric market has been used.

As relevant information, the parameters used on the execution of the CGP algorithms where: stop criteria fixed to 20000 generations, number of nodes fixed to 100+1, nodes columns fixed to 100 and rows fixed to 1, arity per node fixed to 2, any input node can use any of the previous node outputs.

The allowed node functions were: square root, reciprocal, power, addition, subtraction, multiplication and division. On the case of the CGP regular approach, the mutation rate was set to 10%.

The polynomial regression, introduced as an example for comparison, consist on an eight-degree polynomial. In the case of the decision tree, it consist of 20 leaf nodes. The neural network consist of a single layer with 14 neurons. The ANFIS model consist of two membership functions per input.

The reason behind choosing a polynomial regression and a decision trees algorithm was to compare the accuracy of the CGP approach based on their simple structure. The neural network and the ANFIS model have been introduced in order to compare the accuracy with state of the art algorithms. The CGP is able to ensemble a model based on simple function nodes and using only an evolutive algorithm to set the model parameters.

Due to his simplicity, the models created act as mathematic functions easy to implement based on the DAG. This simplicity has led us to compare CGP with basic repressors with the quality of being easy to read such as polynomic regression and binary decision trees. The order and level used on those methods has been fixed proportionally to the median of nodes function obtained from the two hundred CGP models.

After two hundred executions of the modeling algorithms over the randomly sorted data sets, the accuracy measures have been obtained. The median (M) and standard deviation (D) of the experiments have been recorded on the **Table 7** for MAPE and RMSE over the training (Tr), validation (Va) and test (Ts) sets.

Table 7. Comparison of accuracy in terms of MAPE & RMSE values (median & std. Dev.) For models built over two hundred randomly sorted datasets.

Model			**MAPE**			**RMSE**	
		Tr	**Va**	**Ts**	**Tr**	**Va**	**Ts**
CGPprop	***M***	3.53	3.95	4.13	3.49	3.81	3.94
	D	0.69	0.75	0.60	0.79	0.83	0.92
CGP	***M***	4.54	4.73	4.91	4.35	4.18	4.29
	D	0.84	0.83	0.90	0.88	0.80	0.91
Tree	***M***	4.12	4.14	4.15	3.42	3.51	3.53
	D	≈ 0	≈ 0	≈ 0	≈ 0	≈ 0	≈ 0
Poly	***M***	4.91	5.04	5.01	4.53	4.65	4.56
	D	≈ 0	≈ 0	≈ 0	≈ 0	≈ 0	≈ 0
NN	***M***	3.12	3.04	2.77	2.86	2.92	2.78
	D	0.18	0.16	0.19	0.15	0.13	0.13
ANFIS	***M***	4.55	4.21	4.32	4.12	4.02	4..9

D	0.75	0.45	0.51	0.66	0.52	0.47

As it can be observed on the **Table 7**, the proposed evolutive strategy confers to the CGP modelling the ability to select the best chromosome with a low generalization error. This is demonstrated by the homogeneous values obtained from MAPE and RMSE over the data sets. Also, **Table 7** confirms that our evolutive strategy produce a remarkable reduction over the standard deviation of the error. It means that our approach helps the model converge to low errors.

On the other hand, we can check that the common CGP evolutive strategy that only uses the training set to train the model shows a clear tendency to be over trained. The standard deviation of the decision tree and polynomial model are close to zero due its deterministic learning algorithms.

3.3.2.1 Homoscedasticity test over the CGP algorithms.

Regarding to the homoscedasticity study, **Table 8** shows the result by applying Levenne's test based on MAPE obtained from the CGP model algorithms using the test set. The symbol 'b' indicates that the variances of the distributions of the CGP algorithms for the current data set are not homogenous. It implies that the null hypothesis, normality of the distribution, is rejected.

Table 8. Results of Levene homoscedascity test in terms of p-values for CGP algorithms using MAPE values from the test set.

Model	P-values
CGPprop	0,765
CGP	0,962[b]

We had applied the Levenne's tests by considering a significance confidence level of α=0.05. It means that P-values over 0.95 and under 0.05 would reject the null hypothesis. The two hundred MAPE values, obtained from the test set, have been splitted in two groups in order to apply the Levenne's test.

On **Table 8**, the CGP models do not satisfy the homoscedasticity test because the learning algorithm is entirely driven by a stochastic approach such as genetic programming.

The following figure introduces a graphical representation of the homoscedasticity condition over the models trained with our CGP approach. The histogram represents a MAPE by using bars, so that the area of each bar is proportional to the frequency of the represented value. The Q-Q graphic represents a confrontation between the quartiles from MAPE observed (blue) and those from the normal distribution (red).

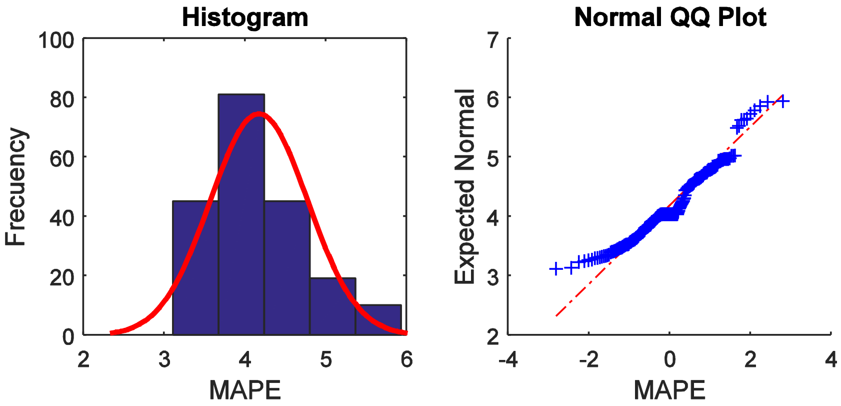


Figure 21. Models obtained from the CGP approach proposed: histogram and Q-Q graphic.

As it can be observed, the MAPE distribution over the test set follow a normal distribution. This confirms the results of the Levenne’s test; the learning algorithm of the CGP model produces a stochastic convergence and our evolutive strategy introduce a slightly measurable disturbance on the model convergence, as well as it is demonstrated on the standard deviations of the **Table 7**.

3.3.2.1 Load forecasting examples

As an example, **Figure 22**, **Figure 23**, and **Figure 24** presents the results obtained from the best model trained with the proposed CGP approach. **Figure 22** shows the architecture of the best model obtained with our CGP approach. On the graph: the inputs are represented by squares, the node functions by circles and output by an oval.

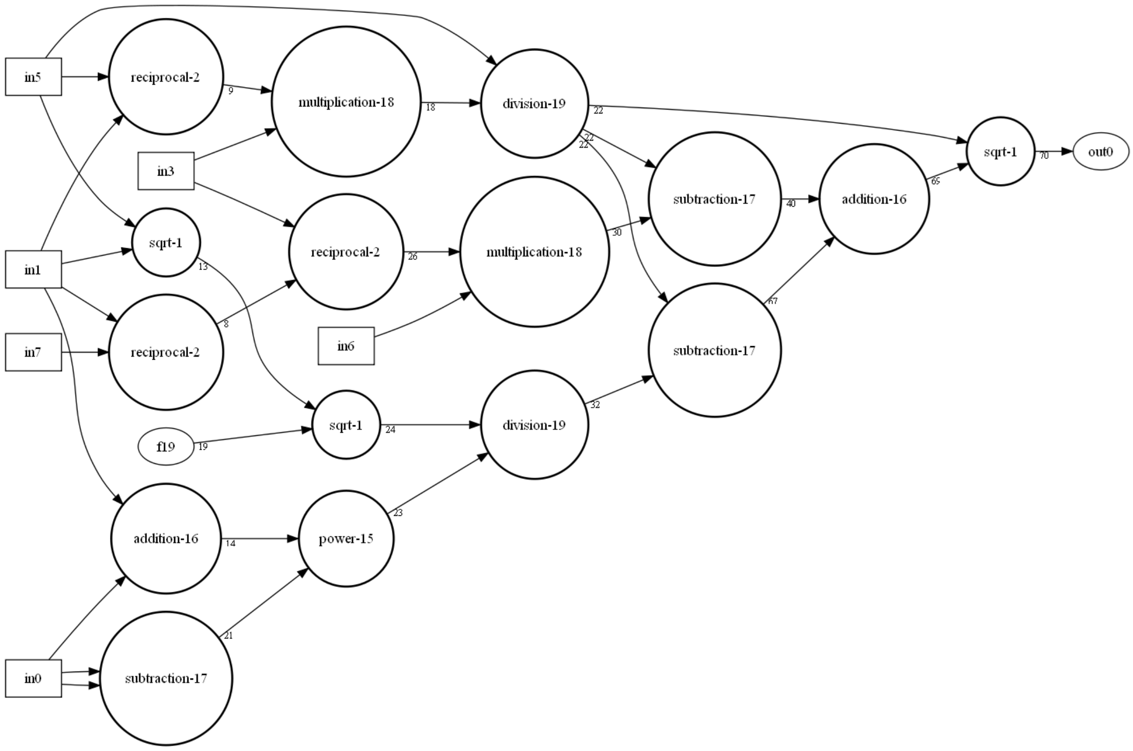


Figure 22. Results of CGP approach proposed: internal architecture of the best model obtained.

Figure 23 shows the one day ahead electricity consumption forecasting corresponding to the CGP model, presented on the **Figure 22**, over the train, validation and test sets. The plot shows a low absolute percentage error (less than 5%).

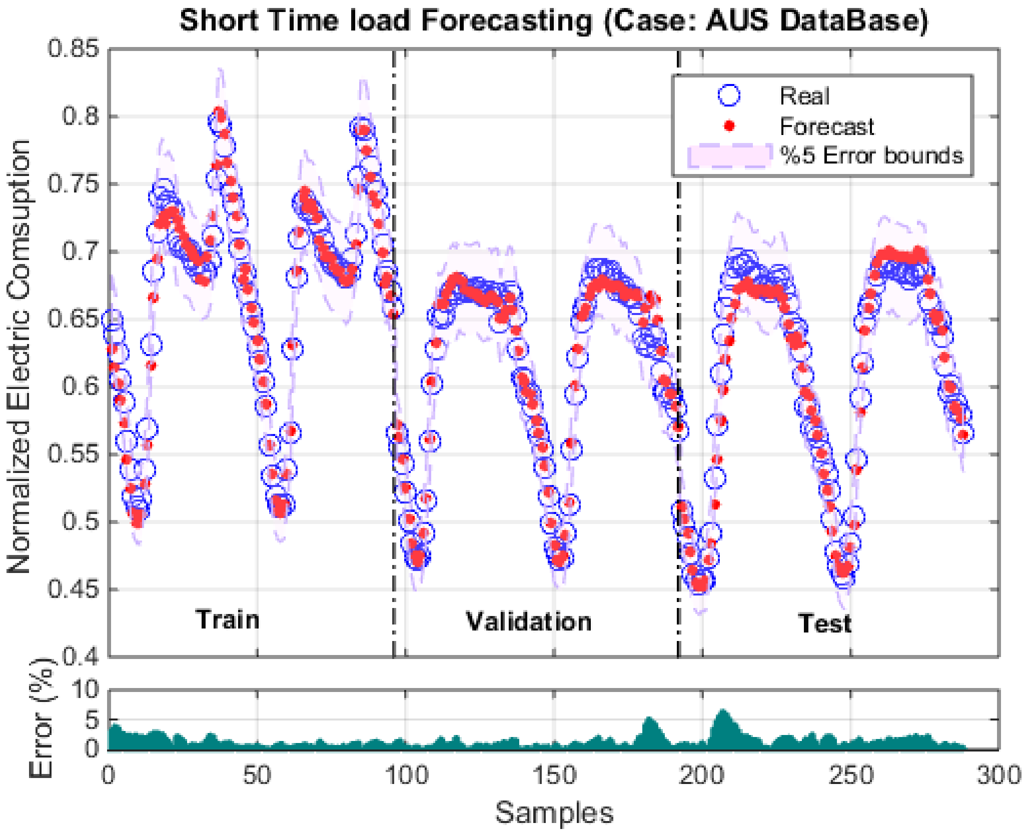


Figure 23. Results of CGP approach proposed: Real and forecasted electricity consumption using the best model trained.

Figure 24 Shows the boxplot of the one day ahead electricity consumption forecasting over the validation set. It presents a comparison between the forecasted samples obtained from the CGP approach presented and the real samples.

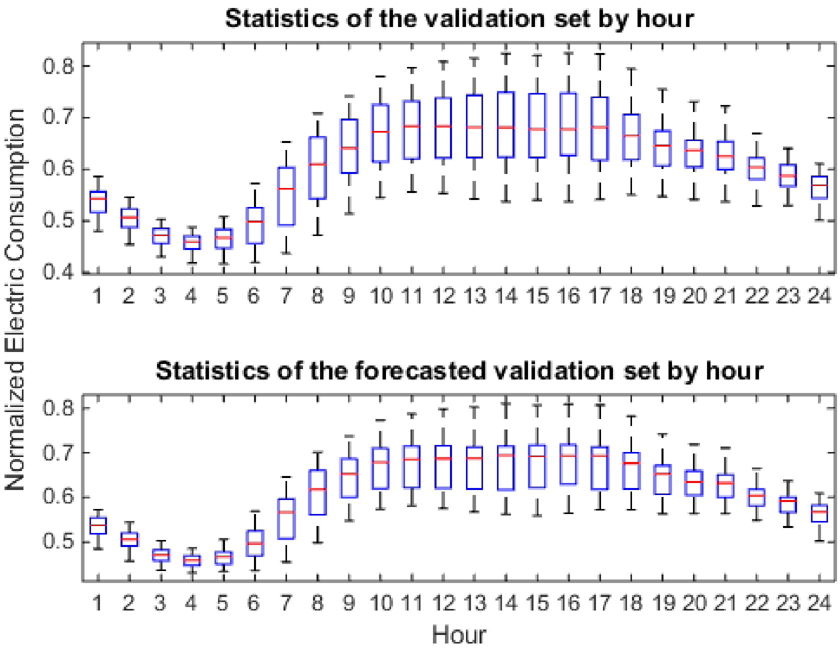


Figure 24. Results of CGP approach proposed: boxplot of the one day ahead load forecasting, over the validation set, carried out by the best model trained.

As we can see, the medians are quite similar, but the percentile intervals are shorter on the forecasted case. This means that the modelling algorithm can't mimic accurately the spread distribution of the electricity consumption, but it keeps the forecasted samples closer to the median than the real samples. It demonstrates the accuracy of the algorithm presented.

In order to validate the reduction of the computational time on the proposed strategy, the **Table 9** presents the statistics of the maximum number of generations reached by the CGP algorithms. Also includes the statistics of the number of node functions obtained from the models and the training time.

Table 9. Generations reached by CGP algorithms, number of nodes functions and training time.

Model		**Generation**	**# Node functions**	**training time (s)**
CGPprop	***M***	5625	20.51	170.38
	D	1851	8.69	40.35
CGP	***M***	14592	23.15	726.98
	D	3210	9.54	12.68

As we can see, the number of node functions are similar; but the generation, in which is achieved the last improvement before reach the maximum generation number, is almost three times lower in our proposed implementation than in the common one.

Regarding to the training time, it is near to five times lower on our implementation. Because our approach uses a double check, it should be slower than the CGP basic implementation that only uses a fitness calculation over the train set. The reason of the enhancement on the computational execution is based on the use of parallel processing. This characteristic is not presented on the original CGP implementation [88].

On our proposed approach, the calculation of the fitness function per chromosome is assigned to a thread. Due to the implementation of the CGP algorithms have been programmed on C++ [88], we have used the OpenMP API [124]. This API supports a multi-platform shared-memory parallel programming in C/C++. This API allows to the user control the number of iterations assigned to each thread and manage the shared and individual memory.

The algorithms have been implemented on a CPU with 8 GB of RAM, a quad-core processor running at 3.4 GHz and WINDOWS 7 as operating system. The CGP implementations have been compiled on the IDE Code::Blocks and the compiled executable file have been managed using MATLAB as well as the other model algorithms.

Because the fitness function was based on MAPE, **Figure 25** shows the MAPE values. They are obtained from the train and validation set, for the generations in which a mutation leads to an accuracy improvement. The mutation rate is also shown.

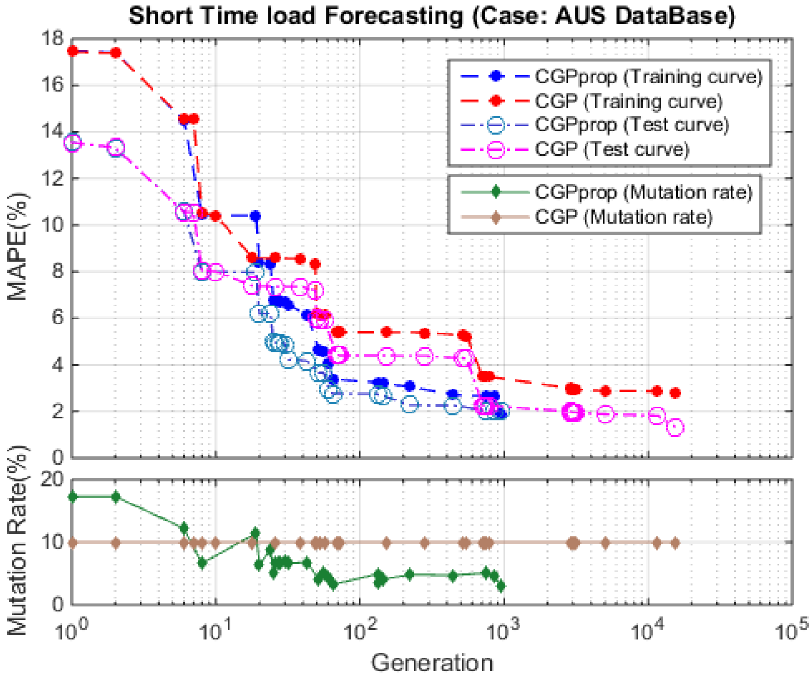


Figure 25. Results of CGP approach proposed: training, test and mutation rate curves carried out by the best model trained.

The **Table 9** and **Figure 25** demonstrates the fast convergence of the proposed evaluative strategy without any penalty over the accuracy.

3.3.3 Discussions and conclusion

On this section, we proposed a novel evolutive strategy based on CGP in order to improve the efficiency of the convergence of models. The methodology has been applied to the load forecasting Australian electric case in order to test it. The main goal of the proposed strategy was to demonstrate that it is possible to train models with a high generalization and accurate error convergence. To test this statement, we used a chromosome selection based on the check of train and validation set.

The fast convergence of the models was an additional goal. A variable mutation rate controlled by the training error and generation was implemented. In addition, the fitness evaluations was implemented using multithreading.

The implemented procedure to test the goals of our approach include: two hundred executions of the CGP approach and another five models over a randomly sorted data set; a comparison over statistics of the errors measured; a statistical test of the homoscedasticity on the error variance; and finally, the measure of the maximum number of generations reached with an error reduction and the number node functions for the CGP algorithms.

As conclusion, our approach achieved a faster convergence than the regular method without compromising its accuracy. It also obtained a low error variance and a low running time.

3.4 Ensemble learning strategies for load forecasting

On this section we will introduce the load forecasting experiments based on ensemble learning strategies. The theory behind the ensemble learning methods as well as the motivations behind the design of a custom hierarchical load forecasting on this thesis has been widely explained on 2.5.4 Hierarchical load forecasting and ensemble learning.

Continuing with the motives exposed on the previous chapter**, Ensemble learning strategies seems to be highly aligned with the objectives of this thesis**, granting the potential to achieve high adaptability-accuracy ratio and the flexibility to be configured with a semi-autonomous degree.

On this thesis, the ensemble learning approaches has been introduced on a custom ensemble approach called **hierarchical load forecasting model**. It includes an ensemble learning approach made by combining different multi-resolution specialized forecasters, who are in turn containing another ensemble group of base learners.

On our custom ensemble implementation the **diversity** is provided by sampling methods such as **bagging**, **mixture of experts based on features**, or **mixture of experts based on clustering**. But rather that train the 1st base learners using only a data set, which provokes highly correlated bagged training sets. On this thesis another diversity component is introduced as novel concept called **multi-resolution manipulation**.

Multi-resolution manipulation calls for a transformation of a data set by means of multi-resolution techniques such as **stationary wavelet transform** (SWT), or **time scaling**. They create non-linear related versions of the data set, which also serve to create parallel training, validation and test sets.

Our ensemble implementations are structured as a three level hierarchy, each the previous described these parallel data sets will create consequently 1st and 2nd level learners, henceforth called **parallel branches**. Being the 1st learner algorithms on each parallel branch affected for the data sample manipulation (bagging), and the input feature manipulation (clustering).

Due to each parallel branch of our scheme have been trained with statistically non-homogeneous data sets, the combination of their predictions must be performed by a non-linear learner. This introduced a 3rd level learner, which creates a new stack level.

On the other hand, because the non-homogeneous data sets possibly are not-statistically related with the target and among them, the more convenient way to measure the ensemble errors is calculate them globally and inside of each branches.

This means that errors based on the bias-variance decomposition, and error-ambiguity decomposition, will measure the accuracy and diversity of the branches, serving as parameters to estimate the relevance of the branch on the ensemble and score each one of the diversity methods implemented.

In order to assets this measures, 1st level error measures on each parallel branch are averaged. In this manner, the averaged errors will serve to compare the accuracy of the base learning algorithms and his potential to depict the introduced clustering diversity.

On this thesis, the pruning method lies on a **heuristic optimization** of the cluster organized base learners. Our heuristic optimization pruning, search in advance the correct number of 1st base learners per cluster which minimizes the general error of the parallel branch. Notice that this method exploit the best of the **clustering-based** and **optimization-based** pruning methods.

The heuristic optimization algorithm is based on a gradual increase of the base learners for each clustering, measuring at each step three key parameters that comprises the global accuracy and the generalization accuracy of the branch. An analysis of these parameters will provide the optimal number of base learners per cluster to be set on the branch.

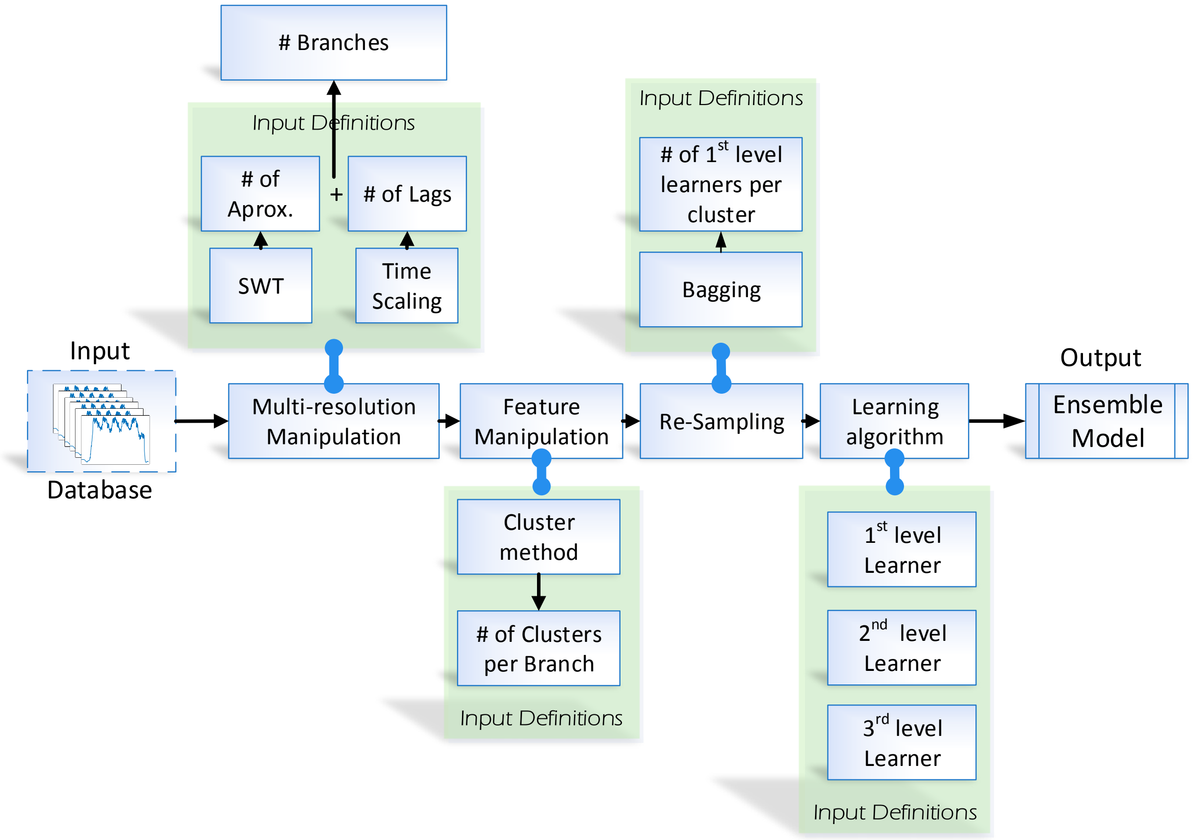


Figure 26. Flow diagram of the novel procedure implemented to create and train a hierarchical load forecasting models.

This heuristic procedure is accelerated using the fastest learning algorithm as base learner, this also allow to test different clustering techniques in order to identify the best combination among clustering methodologies and subspace size. Among the clustering algorithms explored we count with Fuzzy c-means clustering, Self-Organized Maps, Gaussian Mixture Models, K-Nearest Neighbors, and Hierarchical clustering.

Finally as a resume, the efforts introduced on this section in order to minimize the forecast error sources are presented on the following table. The table is based on the concepts introduced on 2.5.4 Hierarchical load forecasting and ensemble learning, and the appendix B.

Table 10. Thesis course of action to minimize the uncertainty due to modelling errors.

- Errors due to modelling structure
    - **Data preprocessing** has been introduced in order to obtain a complete knowledge of the dynamics presented on the load profiles.
    - **Ensemble learning** has been introduced to improve the general goodness of the modelling structure. It relies on the following components:
        - **Stacking,** a combinational method that allows combine base learners predictions by the use of a hierarchical architecture of 3 levels and non-linear learners.
        - **Statistical diversity**, an ensemble learner feature achieved by:
            - A mixture of **parallel branches** based on multi-resolution components.
            - A mixture of base learner algorithms.
            - **Bagging**, a random sampling method.
        - **Clustering** of the base learners based on features of the load profile.
        - **Heuristic optimization** of the cluster size as the clustering-based and optimization-based pruning methods.
- Error due to model parameters
    - Random initialization of the model parameters.
- **Goodness** model statistics
    - **Bias-variance** decomposition
    - Error-ambiguity decomposition
    - Error measures.
        - Accuracy measures
        - Residual statistics
        - Estimation of the **model sensibility** regards to parameters

3.4.1 Architectures based on a non-linear aggregation of multi-resolution components: theoretical approach

As was preciously introduced, the modelling approach proposed on this thesis is an ensemble learning architecture that combines the stacking method and the mixture of experts. The ensemble architecture combine three levels of staking, the 1st level is integrated by a base learner population, the 2nd level is integrated by a branch specialized model and the 3rd level is a single model that integrate all the branches base predictions.

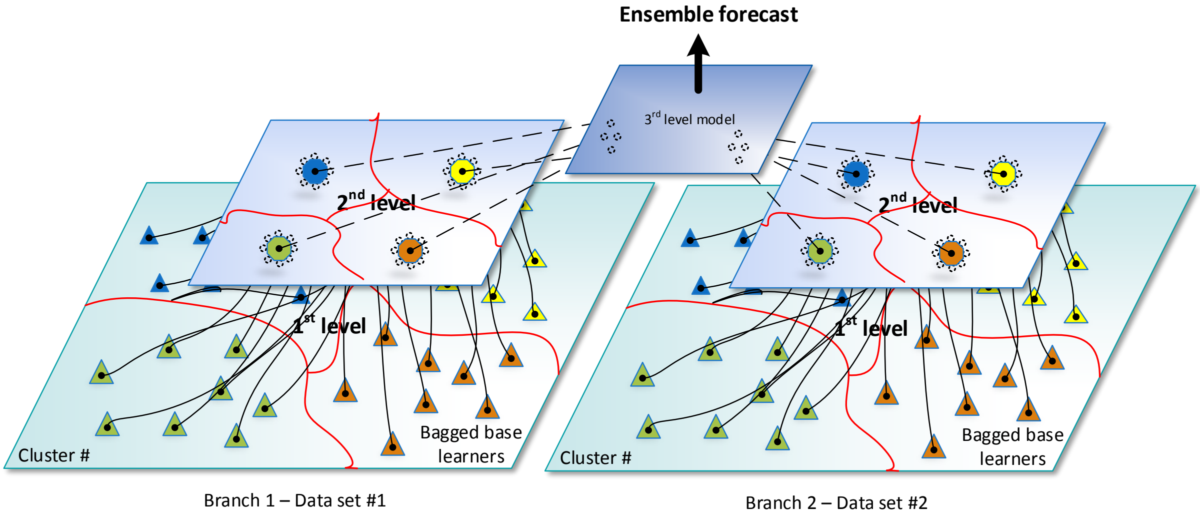


Figure 27. Ensemble learning model representation, 1st level: bagged models grouped by data cluster, 2nd level: model aggregator of base learners predictions, 3rd level: model aggregator of branch predictions.

The **Figure 27** illustrate the components of our ensemble architecture, on it is possible observe that non-homogeneous data sets are required to construct parallel branches. Each data set is also explored using clustering techniques in order to break the profile consumption on feature-based subsets.

This allow us to maximize the extraction of information by the feature specialization of the base learners. Furthermore, in order to increase the resilience of the model bagging is introduced on each data set. This creates high diversity on the prediction of the subset base learners. On the following subsections we make the technical description of our approach.

3.4.1.1 Description of the base learner algorithms

On this section we will describe the model parameters selected for the learners of the ensemble. It means that every one of the learning algorithms addressed have been serve as base learners of the ensemble models. As consequence, each one of these algorithms have been tested during the survey of the best learning algorithms to constitute the ensemble (see section 3.4.1.1.3).

Bayesian MARS and **BLMS** learners are trained using a maximum number of 1000 hinge functions and assume a Gaussian response variable using Markov chain Monte Carlo (MCMC).

Regression trees learners are trained based on the square error, allowing deep trees but making a prune of the tree when the predictive power remain stable. Larger leaves didn’t increase so much the predictive power for our datasets. In fact, a reduction of the tree sizes also reduces training and prediction time, as well as memory usage for the trained ensemble.

Neural Networks learners are trained using a maximum epoch’s equal to 200 and a number of neurons on the hidden layer equal to: 2 * #inputs +2. For autoregressive models as LRNN, NARXNN, and NARNN the delay introduced is equal to one sample.

ANFIS learners are trained using a maximum epoch's equal to 200 and 3 membership functions per cluster. These parameters have been chosen over systematically test in order to reduce the training time and the memory usage without loss accuracy. **SVM** learners are trained using a Gaussian radial basis function.

NCGP learners are trained based on the follow list of parameters: node arity = 5, probabilistic mutation rate = 1%, node function = logistic sigmoid of the weighted sum of inputs, recurrent connection probability = 0, target fitness = 1% or 0.5 million of generations, fitness function = MAPE, evolutionary strategy = 4 + 1.

3.4.1.2 Growth of parallel branches based on multi-resolution components

Multi-resolution components comes from the transformation of the original data set by means of multi-resolution techniques such as stationary wavelet transform, or time scaling. They create non-linear related versions of the original data set, which in turn, serve to create parallel branches.

These braches help to the ensemble to reduce the general bias error, by containing a better track of the conditional mean. Their function is basically provide smooth predictions to the 3rd level learner to decrease his dependency over the branches most affected by the white noise.

Scaling is a discrete operator used to obtain multiple quantized versions of the target attending the window average of the signal. The quantization procedure start with the collection of the ranked **lags**. Then the algorithm perform an average on a number of samples equal to the lag number (window of samples), the procedure starts from the last sample acquired.

Once time the average is obtained, all the samples on the window are replaced by the average. The result are multiples stepped versions of the target signal according to the number of lags selected. The number of lags and parallel branches to grow will be discussed on the section results.

On the case of the parallel branches created using the **SWT**, each branch is trained using as a target an approximation or detail. In order to extract the proper number of details, and guess the correct approximation that retain the minimum of identifiable information from the target, the SWT decomposition is monitored by the welch t-test.

3.4.1.3 Base learner specialization by means of feature classification

We already spoke about the possibility of use methods such as **mixture of experts based on selected features**, or **mixture of experts based on clustering** to create diversity inside the ensemble. These methods consist on the classification of the training data at the current branch into clusters. Finally, the base learners are trained over bagged versions of each cluster data set.

These methods create expert sets of base learns inside branches, increasing the diversity, adaptability, accuracy and the resilience of the ensemble. The concept behind the base learner specialization is graphically explained at **Figure 27**, and **Figure 28**. On those figures is possible distinguish the clusters obtained from the 1st level data set, and the model trained over bagged versions of each cluster.

The pruning method of the cluster-organized base learners lies on a **heuristic optimization** of the cluster. Our heuristic optimization pruning, search in advance the correct number of 1st base learners per cluster which minimizes the general error of the branch. This method exploit the best of the **clustering-based** and **optimization-based** pruning methods.

The heuristic optimization algorithm is based on a gradual increase of the base learners for each clustering, measuring at each step **three key** measures that comprised the global accuracy and the generalization accuracy of the branch. An analysis of the error will provide the optimal number of base learners per cluster to be set on the branch.

This heuristic procedure is accelerated using the most fast/accurate learning algorithm as base learner, this also allow to test different clustering techniques in order to identify the best combination among clustering methodologies and subspace size. Among the clustering algorithms explored we count with Fuzzy c-means clustering, Self-Organized Maps, Gaussian Mixture Models, K-Nearest Neighbors, and Hierarchical clustering, and Density-based spatial clustering.

3.4.1.3.1 *Clustering based specialization*

The classification techniques implemented belongs to supervised and unsupervised algorithms, the objective of these algorithms is bring added diversity to the ensemble on the first two layers making it more robust and adaptable. Due to is necessary stablish an optimal number of classes/features among the data set, on this thesis was develop a method to measure the convenience of a model with a certain number of clusters.

Ensemble pruning serves to identify those individual learners that are representative yet diverse among the given set individual learners, and then select only these individuals to constitute the ensemble. The most straightforward way to identify these individuals is by use of clustering techniques [101], [125], the last ones gives the name to the pruning method called clustering-based pruning.

Generally, the implementations of clustering-based pruning methods used to work in two steps. First step gather the individual learners into a number of clusters by means of a cluster technique. The following examples present some of the clustering techniques applied to ensemble base learner. On [126], authors used hierarchical agglomerative clustering and regarded the probability that the individual learners do not make coincident validation errors as the distance. [127] used KNN clustering based on Euclidean distance, and [128] used deterministic annealing for clustering.

The second step erase the contributions of those base learners not selected. Continuing with the authors previously introduced. For example, [126] select from each cluster the learners which more distant to other clusters; [127] iteratively remove individual learners from the least to the most accurate inside each cluster until the accuracy of the ensemble starts to decrease; and [128] select the closest elements to the centroid of each cluster.

Our heuristic optimization approach rather than abuse of the use of computational resources by means of genetic algorithms [101], is performed based on **explain the maximum variance that the cluster could retain**, because finally this is the objective of this diversity technique.

On this thesis the assessment of the optimal number of learners allowed per cluster is based on a combined custom criteria, this try to retain the maximum variance that could be explained by the cluster over the **validation data**. Our criteria consist on the observation of three key measures, the **first** key measure is **the accumulated averaged variance error** of the cluster calculated among the elements of the cluster,

Eq. 62 $$Acc\bar{V}E = \sum_{t=1}^{T} \overline{variance}_t = \frac{1}{K*E} \sum_{t=1}^{T} \sum_{k=1}^{K} \sum_{e=1}^{E} (h_{b,k,e}^{1}(D^{Ts}|x) - \overline{h_{b,k,e}^{1}(D^{Ts}|x)})^2$$

This formula comes from the generalization of the Bias-Variance-Covariance decomposition. The **second** and **third** key measures are based on t**he F-statistic**, which serves to measure the ratio

Eq. 63 $$F = \frac{explained\ variance\ between{-}group\ variability}{unexplained\ variance\ within{-}group\ variability}$$

It could be equally used to measure if an increment on the cluster models leads a better ability to fit a target, **respect to** a unique learner or the previous number of learners tested. Based on the description of the F-statistic introduced on the appendix B, The **second key measure** could be defined as

Eq. 64 $$F_{h|h_1} = \frac{(SSR_{h_1} - SSR_h)/(K*E-2)}{(SSR_h)/(T-K*E)}$$

It present an **F-test of a cluster with E base learners versus an cluster with an unique learner $\boldsymbol{h_1}$**, under the null hypothesis that the first model does not provide a significantly better fit than the second model, F will have an F distribution, with (K*E-2, T−K*E) degrees of freedom. The null hypothesis is rejected if the F calculated from the data is greater than the critical value of the F-distribution for the false-rejection probability equal to 0.05.

The third key measure is based on an F-test of a cluster with E base learners versus a cluster with E-1 base learner $\boldsymbol{h_1}$.

Eq. 65 $$F_{h|h_{E-1}} = \frac{(SSR_{h_{E-1}} - SSR_h)/K*(E-1)}{(SSR_h)/(T-K*E)}$$

Under the null hypothesis that the first model does not provide a significantly better fit than the second model, F will have an F distribution, with (K*(E-1), T−K*E) degrees of freedom. The null hypothesis is rejected if the F calculated from the data is greater than the critical value of the F-distribution for the false-rejection probability equal to 0.05. The term SSR correspond to the sum square of the residuals over the ensemble set of clusters, it is defined as:

Eq. 66 $$SSR_h = \sum_{t=1}^{T} err_{t,h}{}^2 = \frac{1}{K*E}\sum_{t=1}^{T}\sum_{k=1}^{K}\sum_{e=1}^{E}((D^{Ts}|y) - h^1_{b,k,e}(D^{Ts}|x)\,)^2$$

3.4.1.3.1 *Custom feature based specialization*

This specialization refers to create diversity implementing a **mixture of experts based on selected features.** The custom classification method make use of temporal features clearly defined at the data base, being the strongest one the weekday, and less important the Labor Day indicator. The reader can notice the importance of this variables based on the data base appendix graphics. On these, the daily load distribution per season exhibit the same pattern on all the data bases.

Being the weekends the days with the lower consumption and variance, and the labor days presenting opposite characteristics. Labor days and weekend days also present differences among elements of each set, making more convenient use a more general and strong feature such as the weekday. The method to select the optimal number of expert base learners is also surrogated to the heuristic optimization procedure.

3.4.1.4 Ensemble learning procedure

In this section we will explain the procedure carried out in order to build the ensemble model, the theoretical background is included on 2.5.4 Hierarchical load forecasting and ensemble learning. On **Table 11**, the pseudo-code of the modelling procedure for our proposed ensemble approach is explained. The first section of the table introduces the basic definitions to read the process, the designated acronyms for the elements of the data base, the set of multi-resolution operators and the model notation are introduced

As initial parameters these ones are fix: the data set to be regressed, the number of multi-resolution components / number of branches, the cluster method or the number of clusters to split the training sets, the number of base learners per cluster, and the designated learning algorithm per each level. In order to maintain the simplicity, the diversity on learning algorithms inside of a level is restricted.

The **training stage** start with the branch creation (**1**), the multi-resolution operator is applied over the original data set, except during the first execution which not modify the original data base. Each multi-resolution operation, like time scaling based on lags or time-frequency based on the SWT, create non-homogeneous versions of the original data set to be stored on a cell array (**2**).

The branch data set is divided on training, validation and test sets. Each set with 60, 30 and 10% of the original size respectively (**3**). Once time data set of the branch is created, the diversity generation process start with the input feature manipulation via clustering (**5**). The optimal cluster method and number of clusters has been obtained thanks to the theory presented on the previous section, the clustering results will be introduced on the result section. As result of the clustering operation, feature classified data set are created for the 1st level branch.

The diversity generation process continues with the data sample manipulation via bagging, and the **training of the 1st base learners** (**5-16**). In this step, each set of features is re-sampled with a ratio 63% in-bag to 37% out-of-bag. Then, the learning algorithm defined for the 1st level proceed to create a base learner per each in-bag set on the branch (**8**).

The learning algorithm make use of the in-bag set and the branch validation set. The overall procedure brings statistical diversity to base learners, independence that is manifested on the reduced error and the sharp density estimation.

Because the training set is compromised on the training of the 1st level learners, we had come up with a strategy to train the 2nd and 3rd level learners. In order to create statistical independent sets for the learners on these levels we have proceeded to use the validation and training set to train these levels respectively.

Using the 1st learner predictions over validation set (**11**) **the 2nd level learners are trained** (**13**). In this case the 2nd level learner will be our gating function employed to combine the cluster specialized base learners. This process produce a number of 2nd level learners equals to the number of features.

The FFNN algorithm has demonstrated be the most reliable gating function, the number of neurons in the hidden layer are fix to two times the number of 1st level learners per cluster. On the 3rd level the staking aggregates the mixtures of experts and the diversity principles by the combination of 2nd level learner's predictions.

Using the 1st & 2nd learner predictions over training set (**10**) **the 3rd level learner is trained** (**17**). In this case the 3rd level learner will be our general gating function, employed to combine the clustered branch predictions coming from 2nd level learners. At this point the reader could fear the overfitting of the models because the recurrent use of the training set, so let's make some considerations about this.

The principal method to introduce diversity on the ensemble is based on provide differences among the training sets, such as bagging sampling. If we planning to use a sampling technique to train 2nd & 3rd level, this leads more memory usage, make hard the observation of the error, and still we must sampling over the training and validation sets. These ones are no benefits at all.

But, the principal objections against sample to train upper layers are based on the nature of the ensemble. In order to normalize the prediction of many base learners, ensemble must count with a non-modified set which provide the guide to regularize the vote among the predictions. This means that gating functions needs sets that remain as the original signal to model.

On this thesis the validation set has been set as the guide to regularize the 1st base learner predictions on the 2nd level gating function per each cluster. Similarly, but avoiding use the validation dataset because was used to train 2nd level learners, the 3rd level learner is trained using the prediction of the lower levels over the training set. In this way, the overfitting is avoided due to the high diversity inserted among the lower levels.

The **prediction stage** (**19-26**) starts with the prediction of the test set using the 1st level learners on the branch (**22**). Consequently, the predictions are gathered and feed the 2nd level learner correspondent to the current branch (**24**). Once time all the branches had produced their predictions, these feed the 3rd level learner, producing the ensemble prediction (**27**).

As result of the ensemble procedure the following data is available: an array of the ensemble learners organized by layers and branches, ready to be used for a prediction; the ensemble prediction made over the test set, and the statistics about the model goodness.

A general diagram of the ensemble model architecture is presented on the **Figure 28**. It describes all operations carried inside the architecture starting at the multi-resolution transformation of the original inputs, the prediction carried out by the 1st level learners, the prediction combination carried out by the 2nd level learner, and the combination of the predictions carried out by the branches carried out by the 3rd level learner.

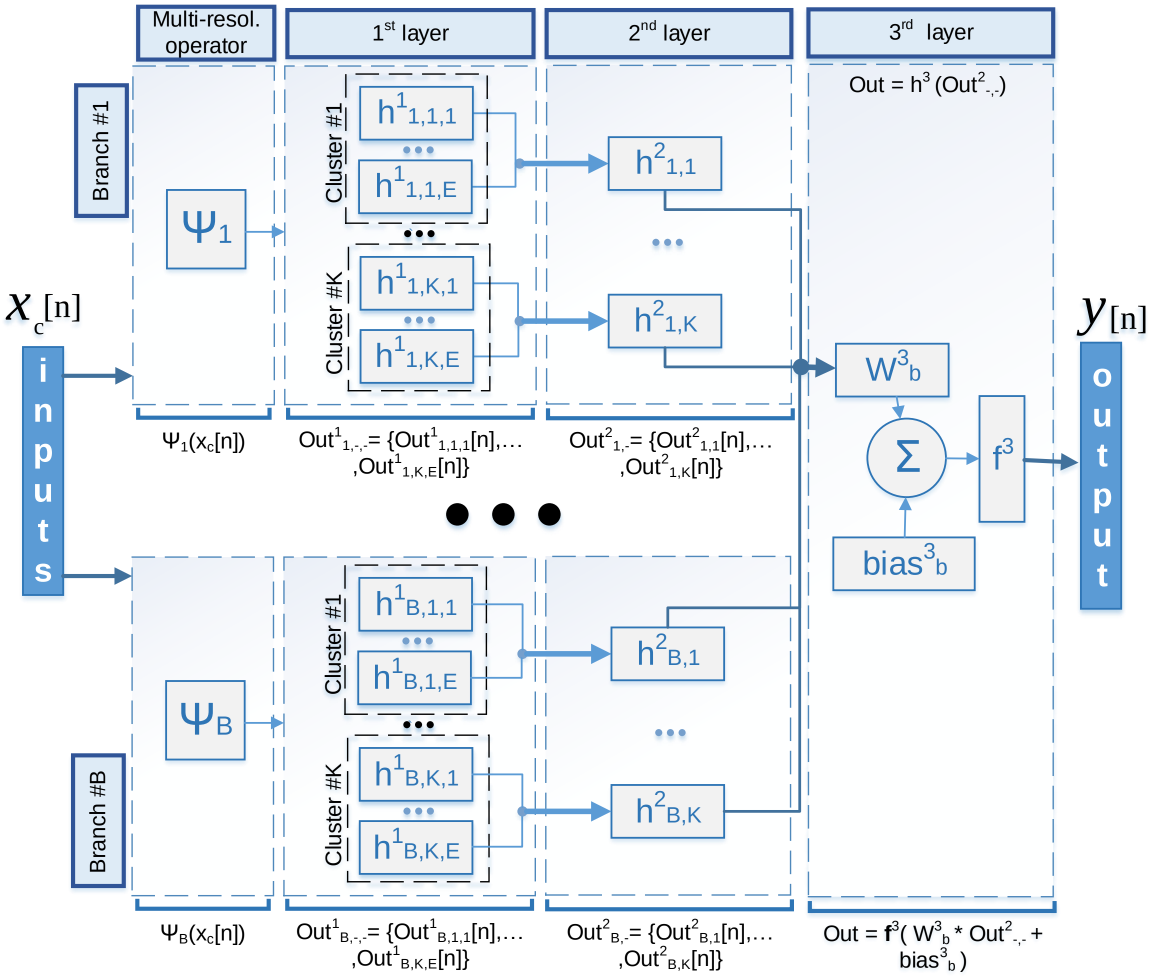

Multi-resol. operator
1st layer
2nd layer
3rd layer
Out = $h^3(\text{Out}^2_{-,-})$
Branch #1
$x_c[n]$
inputs
Ψ_1
Cluster #1
$h^1_{1,1,1}$
$h^1_{1,1,E}$
Cluster #K
$h^1_{1,K,1}$
$h^1_{1,K,E}$
$h^2_{1,1}$
$h^2_{1,K}$
W^3_b
Σ
f^3
bias^3_b
$y[n]$
output
$\Psi_1(x_c[n])$
$\text{Out}^1_{1,-,-} = \{\text{Out}^1_{1,1,1}[n], \ldots ,\text{Out}^1_{1,K,E}[n]\}$
$\text{Out}^2_{1,-} = \{\text{Out}^2_{1,1}[n], \ldots ,\text{Out}^2_{1,K}[n]\}$
Branch #B
Ψ_B
Cluster #1
$h^1_{B,1,1}$
$h^1_{B,1,E}$
Cluster #K
$h^1_{B,K,1}$
$h^1_{B,K,E}$
$h^2_{B,1}$
$h^2_{B,K}$
$\Psi_B(x_c[n])$
$\text{Out}^1_{B,-,-} = \{\text{Out}^1_{B,1,1}[n], \ldots ,\text{Out}^1_{B,K,E}[n]\}$
$\text{Out}^2_{B,-} = \{\text{Out}^2_{B,1}[n], \ldots ,\text{Out}^2_{B,K}[n]\}$
Out = $\mathbf{f}^3(W^3_b * \text{Out}^2_{-,-} + \text{bias}^3_b)$

Figure 28. Ensemble learning model - diagram representation.

Table 11. Pseudo-code procedure of the proposed ensemble learning model.

Definitions:

data set split $D = \{D^{Tr}, D^{Val}, D^{Ts}\}$;
Training data set $D^{Tr} = \{x^{Tr}[\text{row x col}], y^{Tr}[\text{row x 1}]\}$;
Set of multi-resolution operators $\Psi = \{0, \Psi_2, \Psi_3, \dots, \Psi_B\}$;
Set of features θ_K ;
Hypothesis/model h ;

Initial parameters:

Original data set D;
# of branches B $(B \geq 1)$;
Cluster method / # of clusters;
# of base learners per cluster E ;

1st level learning algorithm $\mathfrak{L}^1$;
2nd level learning algorithm $\mathfrak{L}^2$;
3rd level learning algorithm $\mathfrak{L}^3$;

Process:

```
for b = 1:B                                                  // Select  branch  #b
  $D_b = \Psi_b(D)$;                                         // Multi-resolution operator
  $\{D_b^{Tr}, D_b^{Val}, D_b^{Ts}\} = split(D_b)$           // Split of the branch data set
  $\theta_K$ = Classify( $D_b^{Tr}, K, cluster\ method$);    // Split of the training set #b on K set
  for k = 1:K                                                // Select the feature #k
    for e = 1:E                                              // Select the 1st base learner #e
      $[S_{b,k,e}^{inb}, S_{b,k,e}^{oob}] = bagging(\theta_k)$;    // Bagging the feature set #k
      $h_{b,k,e}^1 = \mathfrak{L}_E(S_{b,k,e}^{inb}|x, S_{b,k,e}^{inb}|y, D_b^{Val}|x, D_b^{Val}|y)$    // Training base learner
      $z_{b,k,e}^{oob} = h_{b,k,e}^1(S_{b,k,e}^{oob}|x)$;    // Predictions over the out-of-bag sets
      $z_{b,k,e}^{tr} = h_{b,k,e}^1(D_b^{tr}|x)$;            // Predictions over the training set
      $z_{b,k,e}^{val} = h_{b,k,e}^1(D_b^{Val}|x)$;          // Predictions over the validation set
    end
    $h_{b,k}^2 = \mathfrak{L}^2(z_{b,k,-}^{val}, D_b^{Val}|y)$;    // Train 2nd level learners over val. predictions
    $z_{b,k}^2 = h_{b,k}^2(z_{b,k,-}^{tr})$;                 // Prediction over 1st level prediction over the
                                                             training data set.
  end
end
$h^3 = \mathfrak{L}^3(z_{b,k}^2, D_1^{Tr}|y)$;               // Train 3rd level model
                                                             // Prediction procedure
for b = 1:B
  for k = 1:K
    for e = 1:E
      $out_{b,k,e}^1 = h_{b,k,e}^1(D_b^{Ts}|x)$;             // 1st learner predictions
    end
    $out_{b,k}^2 = h_{b,k}^2(out_{b,k,-}^1)$                 // 2nd learner predictions
  end
end
$\widehat{y_H} = h^3(out_{-,-}^2)$                           // 3rd level prediction
$G_H = Goodness\ statistics$                                 // calculation of the model goodness
```

Output:

$H(x^{Ts}) = Prediction$ // Ensemble prediction
$H = \{h^3, \{h_1^2 \dots h_B^2\}, \{h_{1,1,1}^1 \dots h_{B,K,E}^1\}\}$ // Ensemble model
G_H // Goodness model statistics

3.4.1.4.1 *Ensemble model equation*

Two reasons allows the simplification of our ensemble approach on a model equation: the variety of base learners algorithms implemented, which allows consider each algorithm as a black box, and the use of a simple NN algorithm for 2nd and 3rd layer. On the following equation the ensemble procedure is presented in function of the layer predictions using a regression notation (f(x) = a*x + bias).

Eq. 67 $$\boldsymbol{H(y|x) = \sum_{b=1}^{B} w_b^3 \cdot \left(\sum_{k=1}^{K}\sum_{e=1}^{E}\left(w_{b,k,e} \cdot h_{b,k,e}\big(\Psi_b(x)\big) + bias_{b,k,e}\right)\right) + bias_b^3}$$

Where, $\boldsymbol{\Psi}$ is the multi-resolution operator active on the branch, and the output y is a continuous variable. Given an input x, each local expert $\boldsymbol{h_{b,k,e}}$ tries to approximate the distribution of y and obtains a local output $\boldsymbol{h_{b,k,e}\big(\Psi_b(x)\big)}$. The 2nd level learners provides a set of coefficients: $\boldsymbol{w_{b,k,e}}$ that weigh the contributions of base learners, and $\boldsymbol{bias_{b,k,e}}$ is the parameter of bias. Thus, the final output of the ensemble model is a weighted sum of all the branch outputs produced on the gating functions of the 2nd level and combined on the 3rd level learner.

3.4.1.4.2 *Goodness statistics.*

The goodness of the ensemble model cover statistics like: the general accuracy of the 1st base learners, true authors of the regression task; the diversity among base learners, necessary to extract as much information the ensemble can; accuracy measures of the ensemble performance; or the analysis of the prediction errors.

Because the general theory of the goodness statistics has been introduced on section 2.5.4.1.2.2 and appendix B, we will not dig on deep explanations. The **first statistics** calculated correspond to the estimation of the **model sensibility** regards to parameters. These statistics will bring information about the accuracy of the base learns using the out-of-bag set.

Out-of-bag RMSE error over 1st learner predictions, based on a single branch ensemble.

Eq. 68 $$RMSE_{oob}^{1} = \frac{1}{E.K.N}\sum_{n=1}^{N}\sum_{k=1}^{K}\sum_{e=1}^{E}\left(z_{1,k,e}^{oob} - \left(S_{1,k,e}^{oob}|y\right)\right)^2$$

In the case of ensemble with more than one branch these formulas must be replaced by the Out-of-bag MAPE error. Together with the fact that data base has been normalized, this percentage error allows a sum of error through the non-homogeneous branches. Out-of-bag MAPE over 1st learner predictions, based on a single multi-branch ensemble.

Eq. 69 $$MAPE_{oob}^{1} = \frac{1}{E.K.B.n}\sum_{n=1}^{N}\sum_{b=1}^{B}\sum_{k=1}^{K}\sum_{e=1}^{E}\left| z_{b,k,e}^{oob} - \left(S_{b,k,e}^{oob}|y\right) \middle/ \left(S_{b,k,e}^{oob}|y\right)\right|$$

The **second's statistics** corresponds to the error decomposition measures. These measures has been customized for the proposed ensemble approach, being these a mere conduct to compare the fitness among the ensembles to test. The error decomposition measures are presented below.

The average ambiguity for the first branch of the ensemble is

Eq. 70 $\overline{ambi}(h_1^1|x) = \frac{1}{E.K.N}\sum_{n=1}^{N}\sum_{k=1}^{K}\sum_{e=1}^{E}\left(h_{1,k,e}^1(x) - H(x)\right)^2$

The average error of individual base learners is

Eq. 71 $\overline{err}(h_1^1|x) = \frac{1}{E.K.N}\sum_{n=1}^{N}\sum_{k=1}^{K}\sum_{e=1}^{E}(h_{1,k,e}^1(x) - y)^2$

The generalization error of the base learners on the first branch is

Eq. 72 $err(h_1^1|x) = \overline{err}(h_1^1|x) - \overline{ambi}(h_1^1|x)$

The bias error of the base learners on the first branch is

Eq. 73 $bias^2(h_1^1|x) = \frac{1}{E.K.N}\sum_{n=1}^{N}\sum_{k=1}^{K}\sum_{e=1}^{E}\left(\overline{h_{1,k,e}^1}(x) - y\right)^2$

The variance error of the base learners on the first branch is

Eq. 74 $variance(h_1^1|x) = \frac{1}{E.K.N}\sum_{n=1}^{N}\sum_{k=1}^{K}\sum_{e=1}^{E}\left(h_{1,k,e}^1(x) - \overline{h_{1,k,e}^1}(x)\right)^2$

These statistics allow to measure the suitability of the 1st learning algorithm. The selection of the 1st branch correspond with the fact that only this branch has been trained using a non-manipulated target, allowing a direct check of the base learners performance.

The **third measures** corresponds to the error measures calculated overall ensemble performance. Accuracy measures such as MAPE, SMAPE, Daily peak MAPE, RMSE, Error Variance, MAE are implemented. Furthermore, residuals statistics such as SSR, F-statistic, FVU, R^2, Dubin-Watson statistic, S^2, AIC, SBC and Theil-U statistics are included.

3.4.1.1 Experimental results

This section presents the procedures carried out in order train the proposed ensemble models and test their general goodness over the selected consumption scenarios. Because the amount and diversity of experiments realized, we will address models accordingly with the procedures employed on their training, creating a benchmark of ensemble models which are the central interest of this section.

The comparison among the techniques employed will show how successful were the procedures carried out to strength the model, and how much the error sources have been reduced. The data base used to make a benchmark of our implementations is widely described at the APPENDIX C.

The length of the data set has been reduced to 10 Weeks in order to accelerate the training and optimization procedures, being the length of the training, validation and test sets 6, 3 and 1 week correspondingly. This means that Goodness measures are calculated over a **test set equal to 1 week** despite the forecast is made 1 day-ahead.

3.4.1.1.1 *Optimization of the ensemble parameters based on clustering techniques: optimal number of clusters and base learners.*

For a moment, think about combination of techniques and parameters necessary to construct our model approaches, from the **Figure 26** to **Figure 28** is clearly stated that our particular ensemble method requires a study over a fair quantity of options. Consequently, we have decided divide the optimization process by stages, providing at each one the optimal solution for a set of techniques/parameters.

Our **first stage**, consist on the identification of the **optimal number of clusters and base learners per cluster**. Our heuristic optimization procedure is presented on the **Figure 29**, it gradually grows the set of base learners per cluster and the number of cluster at the same time switch among clustering methods; measuring at each iteration the global accuracy and the generalization accuracy of the branch.

The error measures to be observed correspond to the follow ones:

- Mean absolute percentage error (MAPE), low values secures a high accuracy on the model.
- The accumulated averaged variance error ($Acc\bar{V}E$): it measures how diverse are the 1st level predictions. High values means high diversity captured on the model.
- P-value of the F-test over a cluster with E base learners versus a cluster with a unique learner h_1 ($F_{h|h_1}$): it measures the accuracy of the 1st level predictions. Improvements over the model accuracy on the 1st level predictions will lead to a unitary p-value.
- P-values of the F-test over of a cluster with E base learners versus a cluster with E-1 base learner h_1 ($F_{h|h_{E-1}}$): it measures the accuracy of the 1st level predictions. Improvements over the model accuracy on the 1st level predictions will lead to a unitary p-value.

In order to find the optimal cluster size, our analysis weight the importance of each error measure following the values 0.5, 0.25, 0.125, and 0.125 respectively. It considers the whole performance of the ensemble for a fixed cluster size regardless the number of base learners.

The heuristic procedure is accelerated using one of the fastest, accurate, and consistent learning algorithm as base learner such as feed forward neural network (FFNN). On our study we also integrates several clustering techniques in order to minimize the possibilities of make a mistake on the guessing of the best combination among number of clusters and number of base learners.

The clustering algorithms explored are: K-Nearest Neighbors clustering, Density-based spatial clustering, Fuzzy c-means, Hierarchical clustering, Self-Organized Maps, Self-Organized Maps combined with KNN, and Gaussian Mixture Models.

Also, because our heuristic procedure is an exploratory analysis that only require a representative set of samples, we have decided to boost the speed of the iterations reducing the length of the data set to 10 Weeks. The length reduction has allow us to conduct our heuristic procedure over two consumption scenarios, and three different configurations of learning algorithms for the ensemble architecture.

This implies a six fold execution of the heuristic procedure under different conditions to test the strength and consistency of the results, as well as satisfy the acceptance of our null-hypothesis. Our null hypothesis assume that clusters sizes equals to 2, 5, and 7 will consistently exhibit ideal scores on the four error measures observed. This clusters corresponds with the class: Labour Day, Sunday/Monday/Tuesday-Thursday/Friday/Saturday, and weekdays.

The data bases implemented correspond to the industrial consumption scenario (Car manufacturing plant), and the regional consumption scenario (Australian data). The combinations of learning algorithms on for the 3 levels of the ensemble architecture has been respectively: Reg. Tree - Reg. Tree - Reg. Tree, Reg. Tree – FFNN - FFNN, and FFNN – FFNN - FFNN.

Given the similarities among the results obtained from the six executions of our heuristic procedure, we have decided to prioritize the presentation of the most interesting results. These results correspond to the analysis of high volatile consumption profiles: industrial scenario, and the ensemble architectures which present the most consistent accuracy measures among the experiments: Reg. Tree – FFNN – FFNN, FFNN – FFNN - FFNN.

The **Figure 29** introduces the steps carried out over an execution of our heuristic optimization, the process is equivalent to the pseudo code introduced on the **Table 11**. The procedure start with the definition of the number of branches to grow (**Step 1**). The lack of multi-resolution processing is equivalent to operate only with the original version of the data base, leading to the creation of the base branch.

On the **Step 2** clustering method is selected among the 7 available, also the number of clusters to grow is set form 1 to 10. On the **Step 3** each classified data set is bagged in a number of in-bag and out-bag pair of sub-sets equal to the number of base learners to grow, the number of bagging sets goes from 1:11. The **Step 4** the ensemble model is trained based on the selection of the learning algorithms per level.

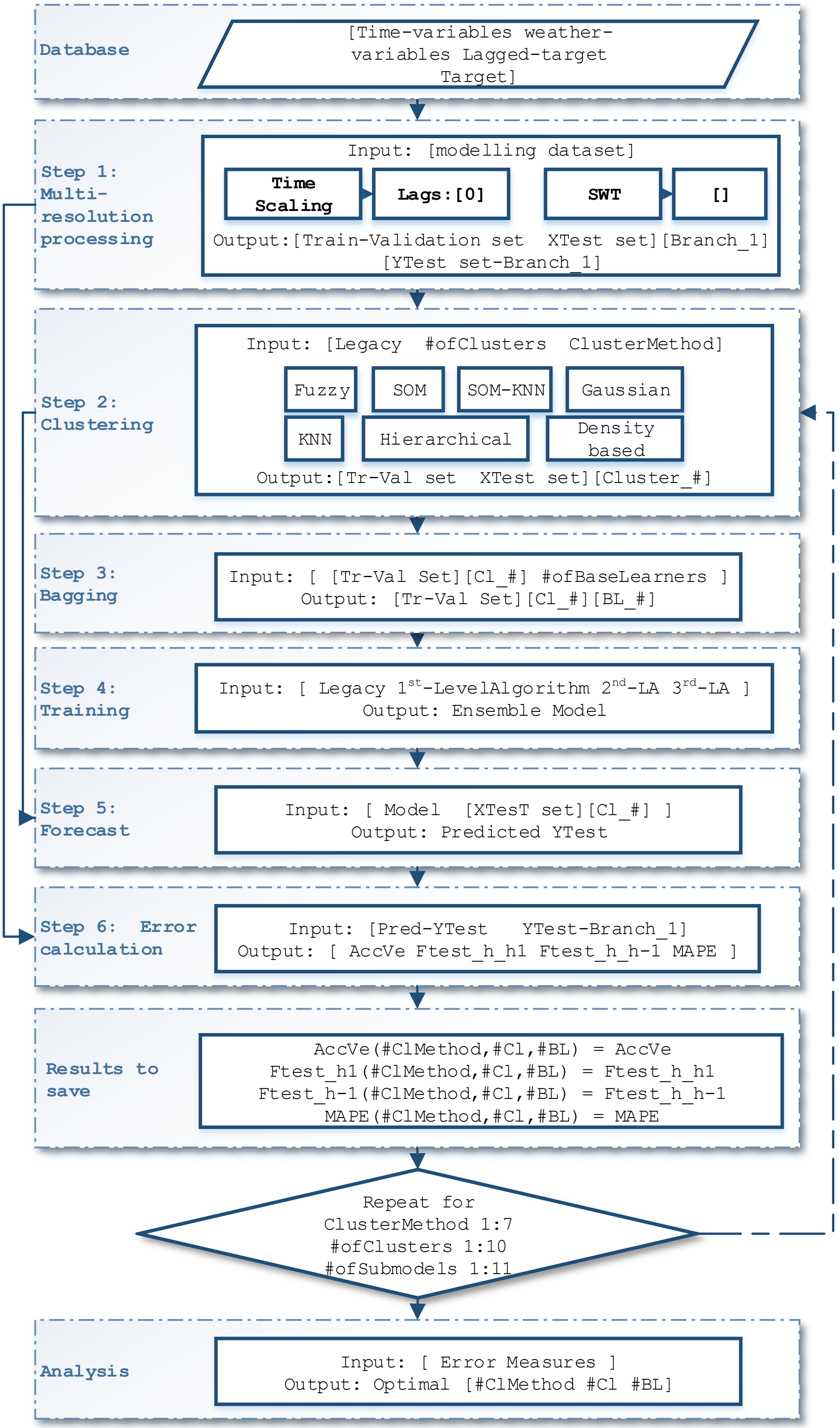

Database
[Time-variables weather-variables Lagged-target Target]
Step 1: Multi-resolution processing
Input: [modelling dataset]
Time Scaling
Lags:[0]
SWT
[]
Output:[Train-Validation set XTest set][Branch_1] [YTest set-Branch_1]
Step 2: Clustering
Input: [Legacy #ofClusters ClusterMethod]
Fuzzy
SOM
SOM-KNN
Gaussian
KNN
Hierarchical
Density based
Output:[Tr-Val set XTest set][Cluster_#]
Step 3: Bagging
Input: [[Tr-Val Set][Cl_#] #ofBaseLearners]
Output: [Tr-Val Set][Cl_#][BL_#]
Step 4: Training
Input: [Legacy 1st-LevelAlgorithm 2nd-LA 3rd-LA]
Output: Ensemble Model
Step 5: Forecast
Input: [Model [XTesT set][Cl_#]]
Output: Predicted YTest
Step 6: Error calculation
Input: [Pred-YTest YTest-Branch_1]
Output: [AccVe Ftest_h_h1 Ftest_h_h-1 MAPE]
Results to save
AccVe(#ClMethod,#Cl,#BL) = AccVe
Ftest_h1(#ClMethod,#Cl,#BL) = Ftest_h_h1
Ftest_h-1(#ClMethod,#Cl,#BL) = Ftest_h_h-1
MAPE(#ClMethod,#Cl,#BL) = MAPE
Repeat for
ClusterMethod 1:7
#ofClusters 1:10
#ofSubmodels 1:11
Analysis
Input: [Error Measures]
Output: Optimal [#ClMethod #Cl #BL]

Figure 29. Flow diagram of the heuristic optimization procedure carried out to find the optimal combination of techniques/parameters inside the ensemble architecture.

The forecast for the test set is made on the **Step 5**, and the error measures calculated on the **Step 6.** Each error measure is saved on an array of three dimensions pointing the clustering method, the number of clusters and the number of base learners employed. When the executions reach the end, a graphic analysis of the result array is done.

In order to facilitate the analysis of the error measures we have proceed to plot the error matrix obtained per clustering method, as a result the figures **Figure 30** - **Figure 36** are presented. On the description of each figure we have include the optimal cluster sizes found at the clustering method analyzed. In general terms, our null-hypothesis could be accepted due to the best scores has been obtained from models with 2, 5, and 7 classes.

Regarding to the optimal number of base learners per cluster, the figures shows that traits measured such as model accuracy (MAPE), Variance captured at 1st level (AccVe), and the Accuracy improvements regarding to variance captured at 1st level (F-tests) fails to deliver a consistent answer if all the classes are observed.

One easy explanation the reader could infer is related with the effects of data base length over the measures. This assumption is rejected because we had make our procedure insensitive as soon we use a relevant subset of the original load profile correspondent to dates among 11-Jan-2017 to 21-Mar-2017, which are stable working weeks of the spring season.

On the other hand, if we observe only the **clusters sizes 2, 5, and 7** we could notice that a low number of base learners leads implicitly to a low MAPE and a low AccVE, regardless the clustering techniques implemented. Among the number of clusters selected the lowest MAPE was 3.85 at the **KNN clustering** (**Figure 30**– MAPE: [1 2]).

Regarding to accumulated variance error is logical conclude that more base learners implies more diversity on the first level of the ensemble, remember that this error show the accumulate differences among the first level predictions against the averaged prediction inside of a cluster.

According to the AccVE the optimal number base learner could be set around 6-8 models per cluster. But helped by the p-values from the test $F_{h|h_1}$, and $F_{h|h_{E-1}}$ we decided to set the **optimal number of base learners to 1 & 8**. As a last comment, NaN scores and errors over 15% has been set to zero in order to maintain low the color bar palette on the figures.

Once time we identified the optimal number of clusters (2, 5, 7), and the optimal number of base learners (8); we proceed to compare the general goodness of the ensemble models trained. The **Table 12**, **Table 13** presents the goodness statistics of the model, the models are presented using the notation #Br_#Cl_#Bl which correspond to the number of branches, clusters and sub models implemented on the model.

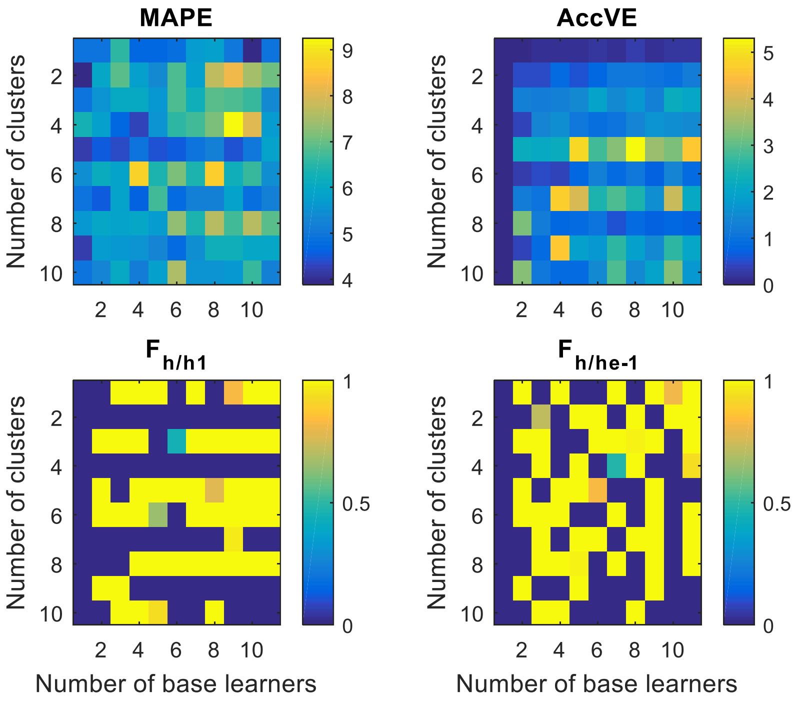


Figure 30. Results of heuristic optimization procedure obtained from the K-Nearest Neighbors clustering. Optimal number of clusters extracted from the graphs: starting from the highest score 5, 7, 3, and 1.

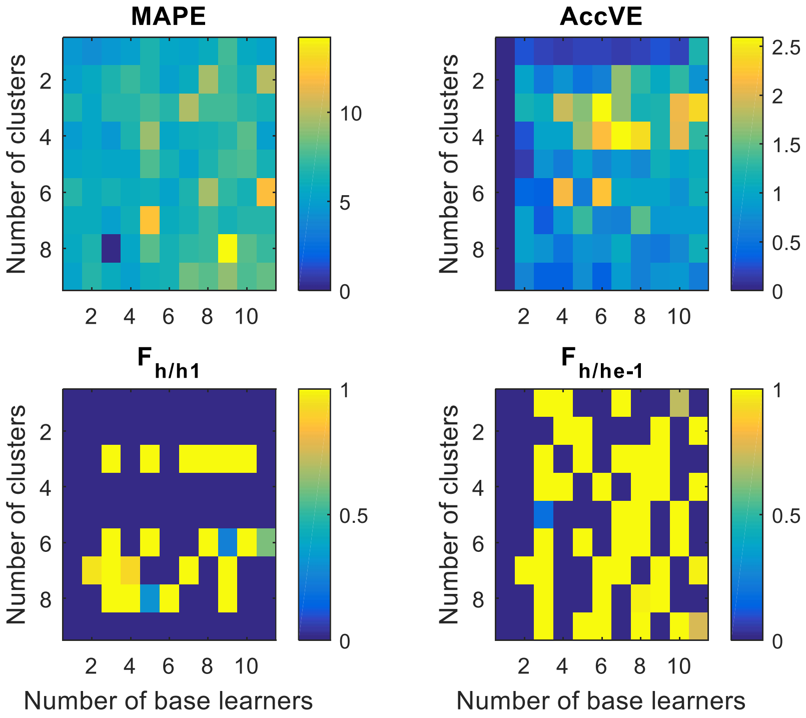


Figure 31. Results of heuristic optimization procedure obtained from the density-based spatial clustering. Optimal number of clusters extracted from the graphs: starting from the highest score 3, 4, 1, and 7.

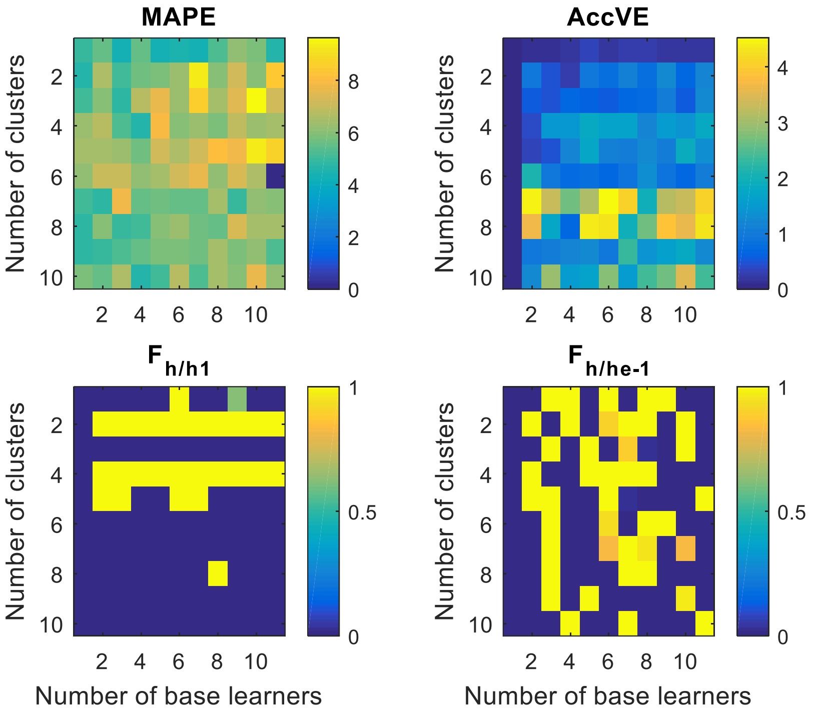


Figure 32. Results of heuristic optimization procedure obtained from the fuzzy c-means clustering. Optimal number of clusters extracted from the graphs: starting from the highest score 7, 8, 2, 4, and 5.

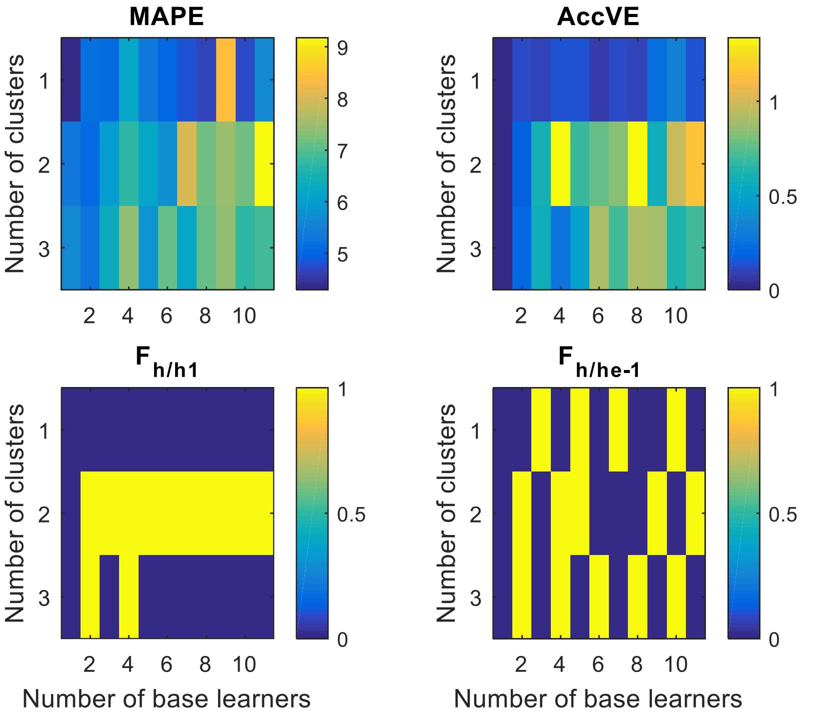


Figure 33. Results of heuristic optimization procedure obtained from the Hierarchical clustering. Optimal number of clusters extracted from the graphs: 2.

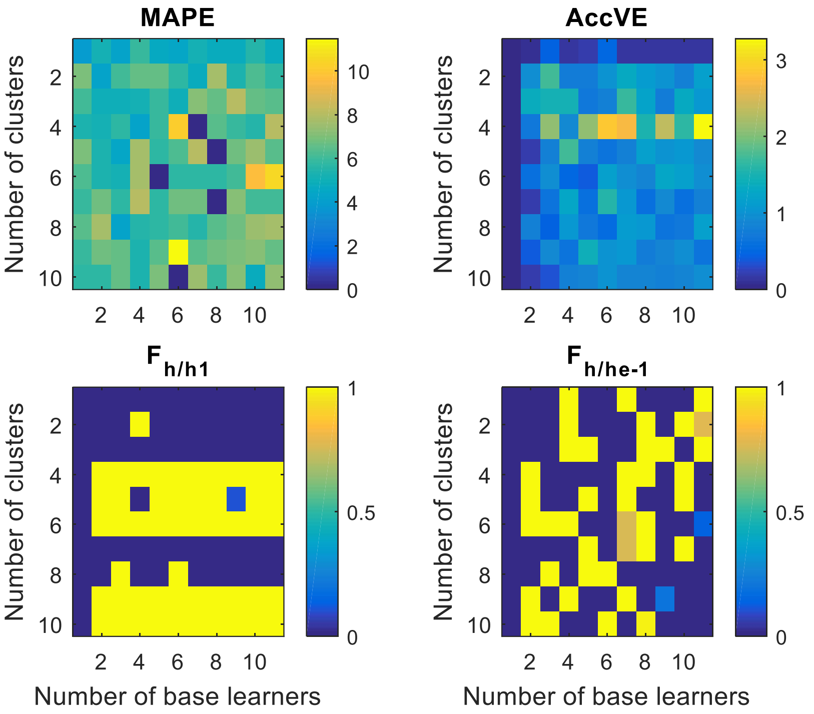


Figure 34. Results of heuristic optimization procedure obtained from the self-organized map clustering. Optimal number of clusters extracted from the graphs: starting from the highest score 4, 6, and 2.

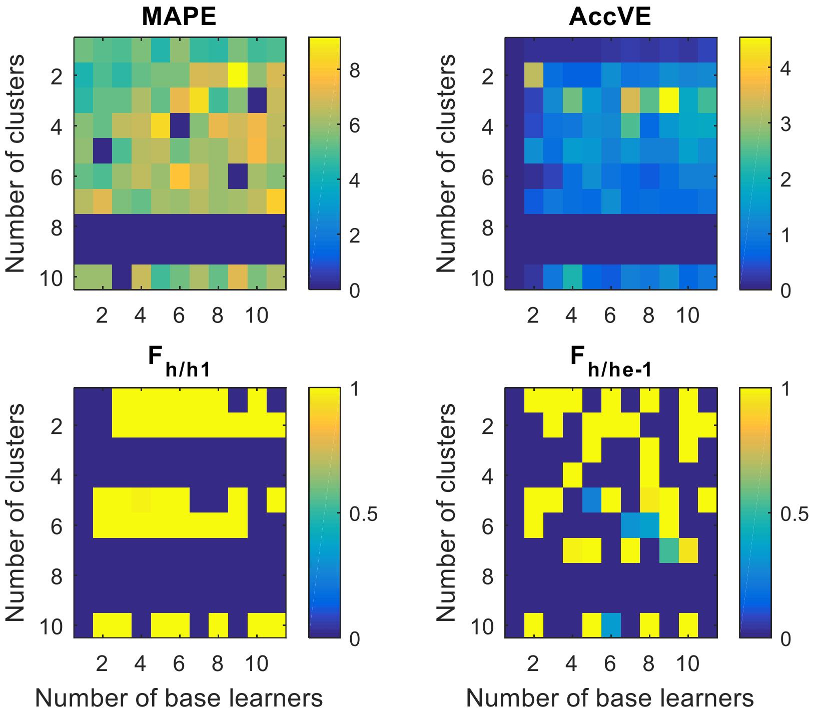


Figure 35. Results of heuristic optimization procedure obtained from the Self-Organized Maps – KNN clustering. Optimal number of clusters extracted from the graphs: starting from the highest score 2, 1, and 5.

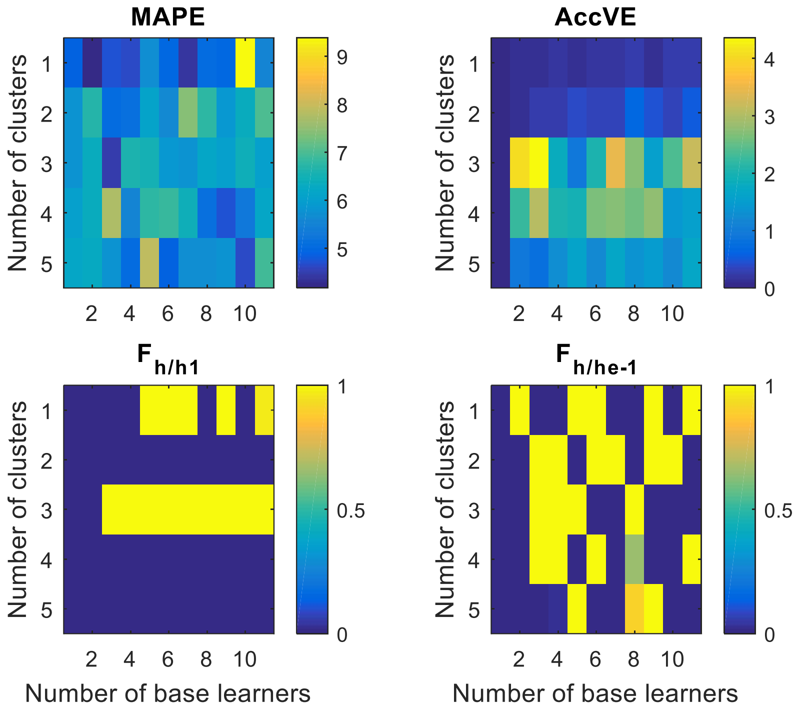


Figure 36. Results of heuristic optimization procedure obtained from the Gaussian mixture models clustering. Optimal number of clusters extracted from the graphs: starting from the highest score 3, 1, and 5.

Table 12. Goodness measures of the ensemble models with configuration: Reg. Tree – FFNN – FFNN, and trained with the optimal parameters obtained from the heuristic optimization.

Goodness measure	**EM 1_1_1**	**EM 1_1_8**	**EM 1_2_1**	**EM 1_2_8**	**EM 1_5_1**	**EM 1_5_8**	**EM 1_7_1**	**EM 1_7_8**
RMSEoob	2,41	2,06	1,73	1,91	0,23	0,18	0,53	0,22
MAPEoob								
Avg. Ambi	0,065	0,028	0,217	0,022	0,003	0,003	0,000	0,048
Avg. Error	0,760	0,788	0,706	0,788	0,030	0,017	0,015	0,020
Gen. Error	0,69	0,76	0,49	0,77	0,03	0,01	0,01	-0,03
Bias Error	0,76	0,79	0,71	0,79	0,03	0,02	0,02	0,02
Var. Error								
ME	-3,72E-06	-1,11E-05	-4,99E-06	-7,46E-06	-7,70E-06	-8,82E-06	-9,76E-06	-6,61E-06
EV	1,81E-06	2,05E-06	2,82E-06	1,30E-06	1,76E-06	1,47E-06	3,48E-06	1,84E-06
MSE	0,0012	0,0014	0,0019	0,0009	0,0012	0,0010	0,0023	0,0012
RMSE	0,0349	0,0371	0,0436	0,0296	0,0344	0,0314	0,0484	0,0351
MAE	0,0191	0,0175	0,0213	0,0155	0,0199	0,0156	0,0308	0,0174
MAPE	4,96	4,19	5,45	3,74	4,85	3,73	7,13	4,33
SMAPE	0,0204	0,0187	0,0228	0,0166	0,0213	0,0166	0,0329	0,0186
Daily Peak MAPE	3,58	2,72	3,06	2,83	5,44	3,03	5,62	5,30
FVU	0,060	0,068	0,094	0,043	0,058	0,049	0,115	0,061
R2	0,940	0,932	0,906	0,957	0,942	0,951	0,885	0,939
Durbin Watson	0,377	0,093	0,270	0,168	0,236	0,181	0,200	0,183
S2	0,001	0,001	0,003	0,003	0,007	0,007	0,011	0,011
S	0,6143	0,3045	0,5193	0,4094	0,4860	0,4251	0,4472	0,4283
AICc	4512	4428	4215	4735	4541	4661	4085	4515
General SBC	4517	4433	4226	4746	4568	4688	4123	4553
Theils U1	0,036	0,038	0,045	0,030	0,035	0,032	0,050	0,036
Theils U2	1,036	1,011	1,047	0,983	1,015	0,998	0,987	1,027

Table 13. Goodness measures of the ensemble models with configuration: FFNN – FFNN – FFNN, and trained with the optimal parameters obtained from the heuristic optimization.

Goodness measure	**EM 1_1_1**	**EM 1_1_8**	**EM 1_2_1**	**EM 1_2_8**	**EM 1_5_1**	**EM 1_5_8**	**EM 1_7_1**	**EM 1_7_8**
RMSEoob	1,89	2,05	1,17	1,25	0,20	0,21	0,28	0,23
MAPEoob	311,17	322,54	252,53	275,43	25,36	26,19	60,37	50,35
Avg. Ambi	0,018	12,936	4,133	2,700	0,009	0,015	0,223	0,143
Avg. Error	0,795	0,649	1,347	2,011	0,030	0,025	0,203	0,087
Gen. Error	0,78	-12,29	-2,79	-0,69	0,02	0,01	-0,02	-0,06
Bias Error	0,80	0,52	1,35	1,36	0,03	0,02	0,20	0,07
Var. Error	0,00	0,13	0,00	0,65	0,00	0,00	0,00	0,02
ME	-2,91E-06	-1,41E-06	-2,76E-07	4,01E-06	-1,87E-06	-3,22E-06	-1,74E-08	-7,85E-06
EV	1,84E-06	3,01E-05	1,83E-06	4,20E-06	2,14E-06	4,23E-06	4,75E-06	5,39E-06
MSE	0,0012	0,0202	0,0012	0,0028	0,0014	0,0028	0,0032	0,0036
RMSE	0,0351	0,1422	0,0351	0,0531	0,0379	0,0533	0,0565	0,0602
MAE	0,0199	0,1170	0,0187	0,0239	0,0217	0,0258	0,0282	0,0264
MAPE	4,89	36,89	4,86	5,71	5,71	6,28	8,20	6,54
SMAPE	0,0212	0,1246	0,0199	0,0253	0,0231	0,0275	0,0300	0,0282
Daily Peak MAPE	4,94	29,92	1,76	7,30	1,53	13,54	4,07	7,64
FVU	0,061	0,999	0,061	0,139	0,071	0,140	0,158	0,179
R2	0,939	0,001	0,939	0,861	0,929	0,860	0,842	0,821
Durbin Watson	0,160	0,011	0,231	0,382	0,139	0,851	0,175	0,404
S2	0,001	0,012	0,003	0,024	0,007	0,063	0,011	0,091
S	0,4004	0,1042	0,4803	0,6180	0,3722	0,9224	0,4185	0,6357
AICc	4503	2637	4507	3978	4408	4026	3876	3900
General SBC	4509	2681	4518	4065	4435	4241	3914	4198
Theils U1	0,036	0,148	0,036	0,054	0,039	0,054	0,058	0,061
Theils U2	0,999	1,000	0,979	0,952	0,988	0,973	0,976	0,961

These tables confirms that more base learners increases the accuracy and variance retained by the ensemble, being the optimal models those ones correspondent to **2, 5 clusters with 8 base learners**. A more detailed comparison is leaved to the reader which could address the theory of the goodness measures presented on the APPENDIX C.

3.4.1.1.2 *Selection of ensemble parameters based on custom features*

On the previous section we confirm that optimal number of clusters match with a temporal classification of the load profile (see null-hypothesis). On this section we will make a direct test of the null-hypothesis, forcing the clustering process to follow the features provided by us.

The on this test we have proceed to evaluate three different cluster sizes: 2, 7, and 24. These clusters corresponds with the classes: Labour Day, weekdays, and Day hour correspondingly. The **Table 14**, and **Table 15** presents the goodness statistics of the model, the models are presented using the notation #Br_#Cl_#Bl which correspond to the number of branches, clusters and sub models implemented on the model.

The ensemble architecture implemented is based on the learning algorithm configurations: Reg. Tree – FFNN – FFNN, and FFNN – FFNN - FFNN.

Table 14. Goodness measures of the ensemble models with configuration: Reg. Tree – FFNN – FFNN, and trained using the manual clustering option.

Goodness measure	**EM 1_2_1**	**EM 1_2_8**	**EM 1_7_1**	**EM 1_7_8**	**EM 1_24_1**	**EM 1_24_8**
RMSEoob	0,91	0,57	1,59	0,68	0,16	0,07
MAPEoob						
Avg. Ambi	0,014	0,017	0,003	0,022	0,136	0,004
Avg. Error	0,378	0,103	1,363	0,669	0,188	0,015
Gen. Error	0,36	0,09	1,36	0,65	0,05	0,01
Bias Error	0,38	0,10	1,36	0,67	0,19	0,02
Var. Error						
ME	-5,15E-06	-8,59E-06	-6,27E-06	-1,08E-05	1,02E-06	-4,87E-06
EV	2,88E-06	1,69E-06	3,72E-06	1,57E-06	4,02E-06	1,41E-06
MSE	0,0019	0,0011	0,0025	0,0011	0,0027	0,0009
RMSE	0,0440	0,0337	0,0500	0,0325	0,0520	0,0308
MAE	0,0226	0,0171	0,0296	0,0168	0,0244	0,0168
MAPE	5,87	4,18	8,48	4,00	6,96	4,30
SMAPE	0,0241	0,0183	0,0316	0,0180	0,0259	0,0180
Daily Peak MAPE	5,94	2,25	6,71	3,30	16,70	5,20
FVU	0,095	0,056	0,123	0,052	0,133	0,047
R2	0,905	0,944	0,877	0,948	0,867	0,953
Durbin Watson	0,240	0,122	0,142	0,164	0,604	0,205
S2	0,003	0,003	0,011	0,011	0,037	0,037
S	0,4898	0,3494	0,3767	0,4048	0,7770	0,4528
AICc	4203	4559	4040	4620	4024	4729
General SBC	4214	4570	4079	4658	4154	4859
Theils U1	0,045	0,035	0,051	0,033	0,053	0,031
Theils U2	1,052	0,979	1,001	0,991	0,984	1,020

Table 15. Goodness measures of the ensemble models with configuration: FFNN – FFNN – FFNN, and trained using the manual clustering option.

Goodness measure	EM 1_2_1	EM 1_2_8	EM 1_7_1	EM 1_7_8	EM 1_24_1	EM 1_24_8
RMSEoob	0,54	0,56	0,42	0,49	0,05	0,05
MAPEoob	73,37	75,31	90,47	97,76	8,69	10,10
Avg. Ambi	0,028	0,040	0,160	3,434	0,015	0,005
Avg. Error	0,136	0,131	1,679	2,213	0,013	0,011
Gen. Error	0,11	0,09	1,52	-1,22	0,00	0,01
Bias Error	0,14	0,12	1,68	0,89	0,01	0,01
Var. Error	0,00	0,01	0,00	1,32	0,00	0,00
ME	7,08E-06	-1,91E-05	-6,01E-06	1,22E-06	-5,58E-06	-6,79E-06
EV	1,59E-06	7,44E-06	3,97E-06	6,94E-06	2,83E-06	2,19E-06
MSE	0,0011	0,0050	0,0027	0,0047	0,0019	0,0015
RMSE	0,0327	0,0707	0,0517	0,0683	0,0436	0,0384
MAE	0,0193	0,0320	0,0253	0,0315	0,0216	0,0183
MAPE	5,31	8,15	6,56	8,25	5,22	4,40
SMAPE	0,0205	0,0345	0,0270	0,0335	0,0231	0,0196
Daily Peak MAPE	5,58	8,61	4,06	27,02	7,25	3,55
FVU	0,053	0,247	0,132	0,230	0,094	0,073
R2	0,947	0,753	0,868	0,770	0,906	0,927
Durbin Watson	0,179	0,438	0,097	0,539	0,351	0,234
S2	0,003	0,024	0,011	0,091	0,037	0,400
S	0,4231	0,6615	0,3112	0,7344	0,5928	0,4835
AICc	4599	3593	3996	3730	4260	4923
General SBC	4610	3680	4035	4027	4390	5824
Theils U1	0,033	0,073	0,053	0,069	0,045	0,039
Theils U2	0,979	1,001	1,000	1,013	0,978	0,963

Following the trend of the previous section, the results the diversity created via manual classification corroborate the strength of the temporal features. Being the strongest trait the load pattern per days, followed by the load pattern per labor day, and the load pattern per hour. The reader can notice the importance of this variables based on the data base appendix graphics. On these, the daily load distribution per season exhibit the same pattern on all the data bases.

On the other hand, the results obtained by means of our manual clustering are as good as their counterpart results obtained from the clustering techniques. Being the more accurate achieved on the ensemble model Reg. Tree – FFNN – FFNN **1_2_8** as consequence of the better classification on the clustering algorithm.

As a resume of the clustering procedures, the **Figure 45** presents the MAPE/RMSE of each test made. On it is possible observe that the lowest MAPE/RMSE are always obtained with high numbers of regression trees as base learners. On the contrary FFNN base learners obtain low

MAPE/RMSE with a single base learner per cluster. This clearly shows that weak base learners performs better than strong ones, at least for this data base.

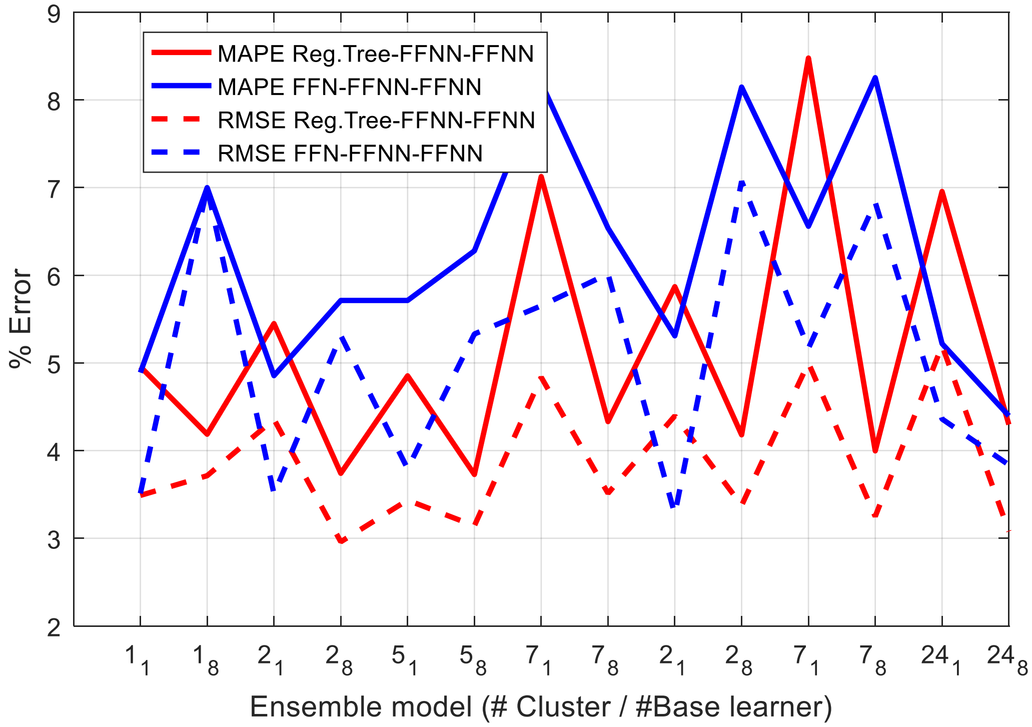


Figure 37. Comparison of the MAPE/RMSE among the ensemble models constructed from the clustering procedure and the manual classification.

Another interesting conclusion comes from the classification of the data according to the hour observed, under the condition of 24 classes and 8 base learners both ensembles achieve low errors. The meaning of such behavior is intuitive, as less samples classified on the cluster the strong base learner sharp his accuracy while the weak learner remain dumb.

3.4.1.1.3 *Survey of the best learning algorithms to constitute the ensemble*

The differences among the goodness of weak learners and strong learners is visible, so we must conduct a study among the best combinations of learning algorithms on the ensemble levels. We have created a pool of base learners for the first layer: Bayesian MARS regression, BLMS regression, regression trees, NCGP, NARXC, NARX, FFNN, CCNN, LRNN, ANFIS, SVM, and RBNN. On the second and third layer we had restricted the functions to FFNN and regression trees.

The best performances amongst the first layer learning algorithms where for regression trees, Bayesian MARS regression, CCNN, FFNN, and RBNN respectively. The best performances amongst the second and third layer where for FFNN, and CCNN. The best 10 combinations are presented on the **Table 16**, and **Table 17** It is easy to appreciate that weak learners such as regression trees and Bayesian MARS overwhelm the results obtained by the strong learners.

Furthermore, the tables corroborates that models based on neural network architectures performs better among the strong learners algorithms. On the other hand weak learners such as regression trees and Bayesian MARS stand out amongst the base learners.

Table 16. Goodness measures of the best 10 ensemble models created from the combinations of the learning algorithms.

Goodness measure	Reg. Tree-FFNN-FFNN 2_8	Reg. Tree-FFNN-FFNN 5_8	Reg. Tree-FFNN-FFNN 24_8	FFNN - FFNN-FFNN 24_8	CCNN - FFNN-FFNN 2_1
RMSEoob	1,91	0,18	0,07	0,05	0,556
MAPEoob	0,00	0,00	0,00	10,10	76,061
Avg. Ambi	0,022	0,003	0,004	0,005	0,007
Avg. Error	0,788	0,017	0,015	0,011	0,112
Gen. Error	0,77	0,01	0,01	0,01	0,104
Bias Error	0,79	0,02	0,02	0,01	0,112
Var. Error	0	0	0	0	0
ME	-7,46E-06	-8,82E-06	-4,87E-06	-6,79E-06	-4,65E-06
EV	1,30E-06	1,47E-06	1,41E-06	2,19E-06	1,30E-06
MSE	0,0009	0,0010	0,0009	0,0015	0,001
RMSE	0,0296	0,0314	0,0308	0,0384	0,030
MAE	0,0155	0,0156	0,0168	0,0183	0,016
MAPE	3,74	3,73	4,30	4,40	4,063
SMAPE	0,0166	0,0166	0,0180	0,0196	0,017
Daily Peak MAPE	2,83	3,03	5,20	3,55	2,161
FVU	0,043	0,049	0,047	0,073	0,043
R2	0,957	0,951	0,953	0,927	0,957
Durbin Watson	0,168	0,181	0,205	0,234	0,158
S2	0,003	0,007	0,037	0,400	0,003
S	0,4094	0,4251	0,4528	0,4835	0,398
AICc	4735	4661	4729	4923	4738,210
General SBC	4746	4688	4859	5824	4749,194
Theils U1	0,030	0,032	0,031	0,039	0,030
Theils U2	0,983	0,998	1,020	0,963	1,026

Table 17. Goodness measures of the best 10 ensemble models created from the combinations of the learning algorithms.

Goodness measure	CCNN - FFNN-FFNN 24_1	RFB_FFNN_FFNN 24_8	BMARS - FFNN-FFNN 2_1	BMARS - FFNN-FFNN 5_1	BMARS - FFNN-FFNN 24_1
RMSEoob	0,056	0,050	0,704	0,306	0,053
MAPEoob	10,925	9,872	95,251	43,629	10,686

Avg. Ambi	0,001	0,017	0,036	0,007	0,001
Avg. Error	0,008	0,011	0,212	0,085	0,012
Gen. Error	0,007	-0,006	0,177	0,077	0,011
Bias Error	0,008	0,011	0,212	0,085	0,012
Var. Error	0	0	0	0	0
ME	-7,70E-06	5,10E-07	-7,06E-06	-5,80E-06	-4,89E-06
EV	2,09E-06	1,91E-06	2,40E-06	1,61E-06	1,30E-06
MSE	0,001	0,001	0,002	0,001	0,001
RMSE	0,037	0,036	0,040	0,033	0,030
MAE	0,019	0,018	0,019	0,016	0,016
MAPE	4,474	4,668	4,499	4,200	3,975
SMAPE	0,020	0,020	0,020	0,017	0,017
Daily Peak MAPE	2,795	11,759	1,901	4,133	3,020
FVU	0,069	0,063	0,080	0,053	0,043
R2	0,931	0,937	0,920	0,947	0,957
Durbin Watson	0,186	0,508	0,203	0,189	0,206
S2	0,037	0,400	0,003	0,007	0,037
S	0,432	0,713	0,451	0,434	0,454
AICc	4464,526	5016,156	4324,107	4598,552	4784,178
General SBC	4594,760	5917,464	4335,092	4625,976	4914,412
Theils U1	0,038	0,036	0,041	0,034	0,030
Theils U2	1,036	0,998	1,029	1,024	1,034

3.4.1.1.4 *Growth of parallel branches based on multi-resolution components*

Multi-resolution components comes from the transformation of the original data set by means of multi-resolution techniques such as stationary wavelet transform, or time scaling. They create non-linear related versions of the original data set, which in turn, serve to create parallel branches.

On this section we will test the null-hypothesis which states that the general goodness of the ensemble model could be increased with the insertion of multi-resolution branch specialization. The multi-resolution techniques implemented are called Time scaling and the non-decimated Stationary wavelet transform.

Time scaling requires a set of lags, each one in charge of grow a branch, easily obtained from an analysis of the periodogram, Sample ACF, and PACF. An example of this analysis is presented on the APPENDIX C for the Australian load profile. The lags selected to be tested, in order of importance, corresponds to 2, 24, 8, 12, and 48 hours.

It’s important to remark that every load profile have specific scores among the autocorrelation peaks and periodogram peaks; in order to generalize the lags tested on this section corresponds to the first ranked scores obtained from the industrial load profile and the other scenarios. In the case of the SWT, The number of approximations implemented has been set among 3 to 5 levels of decomposition.

Based on the best ranked ensemble models presented on **Table 16**, and **Table 17**, we have proceeded to train each one of these models with the all the combinations possible amongst the time scaling lags, SWT number of approximations, or both. On the **Table 18**, the best ranked ensemble model configurations obtained from the multi-resolution pool are presented.

Table 18. MAPE and RMSE of the best 5 ensemble models created from the combinations of the multi-resolution elements.

Ensemble Model # Cluster_#Base leaner	# Clusters / # Base leaners	Time Scaling Lags	SWT Number of approximations	RMSE	MAPE
Reg. Tree-FFNN-FFNN	1 / 8		5	3,399	4,054
FFNN-FFNN-FFNN	24 / 8		5	3,884	4,672
FFNN-FFNN-FFNN	1 / 3		5	3,672	4,347
CCNN-FFNN-FFNN	2 / 1		5	2,820	4,003
CCNN-FFNN-FFNN	1 / 1	1, 2, 8		2,870	3,979

In general terms, the inclusion of multi-resolution branches on the optimal haven't introduced improvements, except in the case of the algorithm CCNN-FFNN-FFNN 2_1 which has increased his RMSE but maintained his MAPE.

Although, the new configuration FFNN-FFNN-FFNN 1_3 has reached a new minimum. It means that multi-resolution components doesn't necessary work together with previously optimized #cluster - #base learner ensemble models.

3.4.1.1.5 *Resume of the best ensemble model configurations*

The best five scored ensemble models has been gathered on the **Table 19**; also we have include an ANFIS, and FFNN models in order to compare the ensemble model accuracy with some simple control algorithms. The **Table 20** presents the goodness measures of the models presented on the **Table 19**.

Table 19. MAPE and RMSE of the best 5 ensemble models created from the combinations of: multi-resolution elements, number of clusters, number of base learners, and learning algorithms.

#	Ensemble Model	# Clusters / # Base leaners	Time Scaling Lags	SWT Number of approximations	RMSE	MAPE
1	Reg. Tree-FFNN-FFNN	5 / 8			3,142	3,728
2	BMARS -FFNN-FFNN	24 / 1			2.952	3.975
3	CCNN-FFNN-FFNN	2 / 1	1, 2, 8		2,870	3,979
4	CCNN-FFNN-FFNN	2 / 1		5	2,820	4,003
5	Reg. Tree-FFNN-FFNN	1 / 8		5	3,399	4,054
	ANFIS				6,952	13,016
	FFNN				5,369	6,838

Table 20. Goodness measures of the best 5 ensemble models created from the combinations of: multi-resolution elements, number of clusters, number of base learners, and learning algorithms.

Goodness measure	1	2	3	4	5
RMSEoob	0,18	0,053	0,553	0,558	1,977
MAPEoob	0	10,686	75,236	76,595	0
Avg. Ambi	0,003	0,001	0,110	0,034	0,282
Avg. Error	0,017	0,012	0,143	0,128	0,696
Gen. Error	0,01	0,011	0,033	0,094	0,414
Bias Error	0,02	0,012	0,143	0,128	0,696
Var. Error	0	0	0	0	0
ME	-8,82E-06	-4,89E-06	-6,78E-06	-1,42E-06	-1,08E-05
EV	1,47E-06	1,30E-06	1,23E-06	1,18E-06	1,72E-06
MSE	0,0010	0,001	0,001	0,001	0,001
RMSE	3,14	2,95	2,87	2,82	3,40
MAE	0,0156	0,016	0,016	0,017	0,016
MAPE	3,73	3,975	3,979	4,003	4,054
SMAPE	0,0166	0,017	0,017	0,018	0,018
Daily Peak MAPE	3,03	3,020	4,101	4,698	3,794
FVU	0,049	0,043	0,041	0,039	0,057
R2	0,951	0,957	0,959	0,961	0,943
Durbin Watson	0,181	0,206	0,299	0,315	0,334
S2	0,007	0,037	0,001	0,001	0,001
S	0,4251	0,454	0,547	0,561	0,578
AICc	4661	4784	4785	4821	4557
General SBC	4688	4914	4818	4886	4590
Theils U1	0,032	0,030	0,029	0,029	0,035
Theils U2	0,998	1,034	1,007	0,968	0,999

The best results are integrated by ensemble models which employ weak base learners such as Regression Trees and Bayesian MARS. These models are specialized on load profile features such as Day Type (Sunday, Saturday, Friday, Monday, Tuesday-Thursday), or day hour. These features corresponds to a number of classes equals to 5 and 24 and are fount on Model #1 & #2.

Model #3 & #4 use a strong learner as base learner, CCNN; it also make use of a calendar variable, Labor Day, as element to discern among the features of the load. In comparison his single branched version, presented at **Table 16**, the increment of the model accuracy suggest the convenience of multi-resolution elements on ensemble models with strong base learners.

Model #5 has been a surprised among the pool of ensemble models, previously we have noticed that ensemble models based on weak learners often requires larger numbers of base learners per cluster, but on this case the large number of clusters seems to be compensated by the number of branches (1 + 5 approximations).

In order to present the accuracy of the ensembles in a visual form, we have proceeded to plot the forecasts obtained from the 5 best models over the industrial scenario test set.

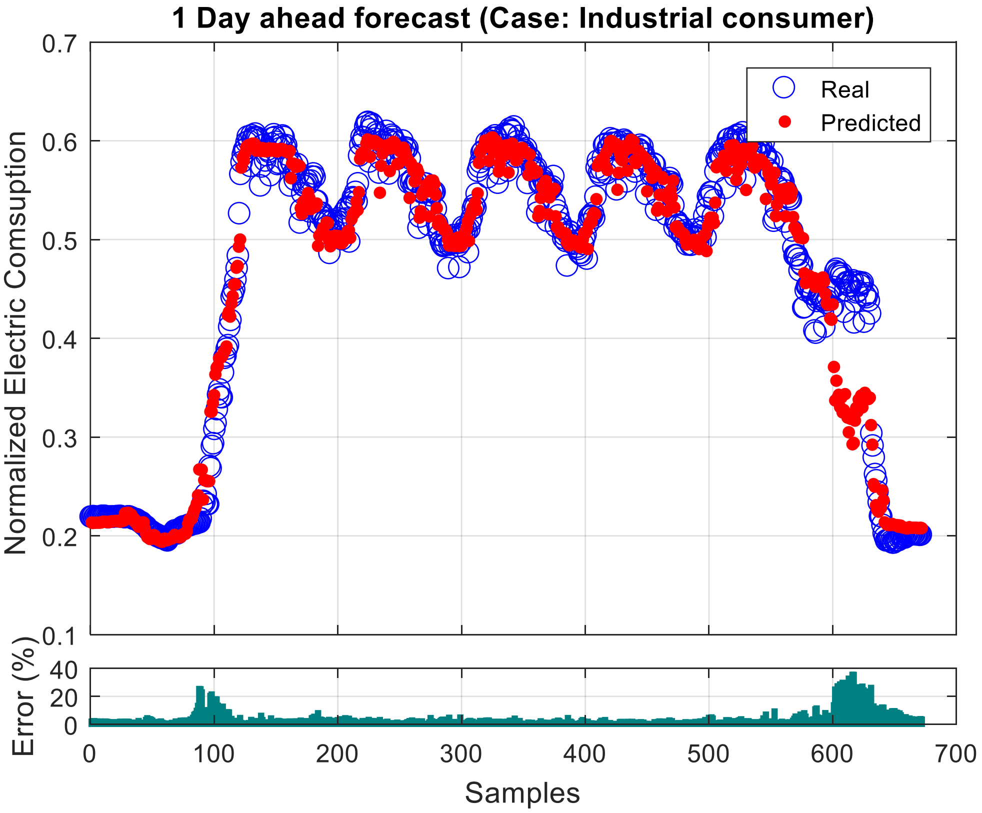


Figure 38. Load forecast correspondent to the ensemble model #1. Learning algorithm combination: Reg. Tree-FFNN-FFNN. Number of clusters/number of base learners: 5/8. Lags implemented: 0. Approximations implemented: 0.

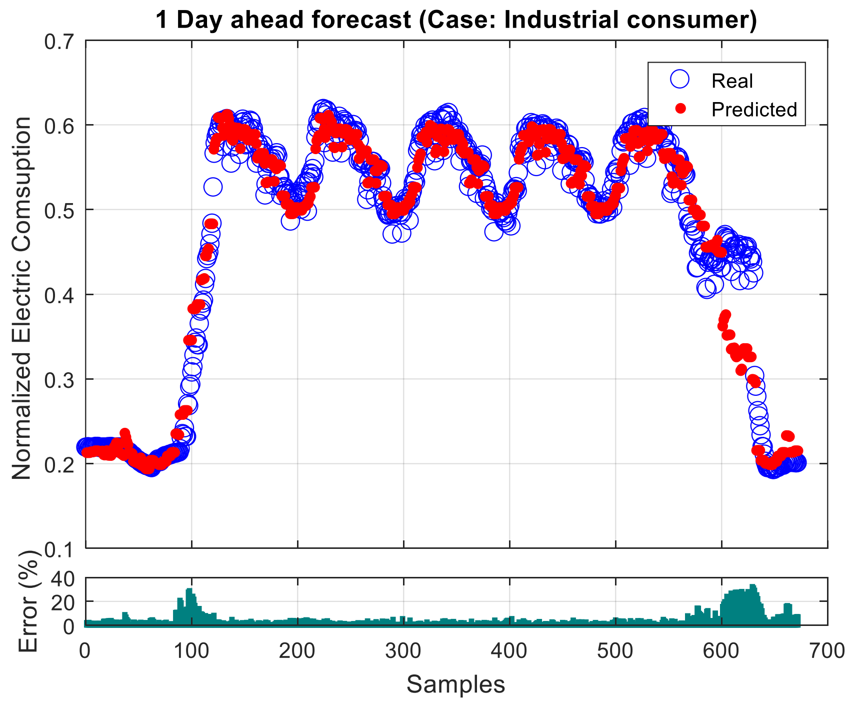


Figure 39. Load forecast correspondent to the ensemble model #1. Learning algorithm combination: BMARS -FFNN-FFNN. Number of clusters/number of base learners: 24/1. Lags implemented: 0. Approximations implemented: 0.

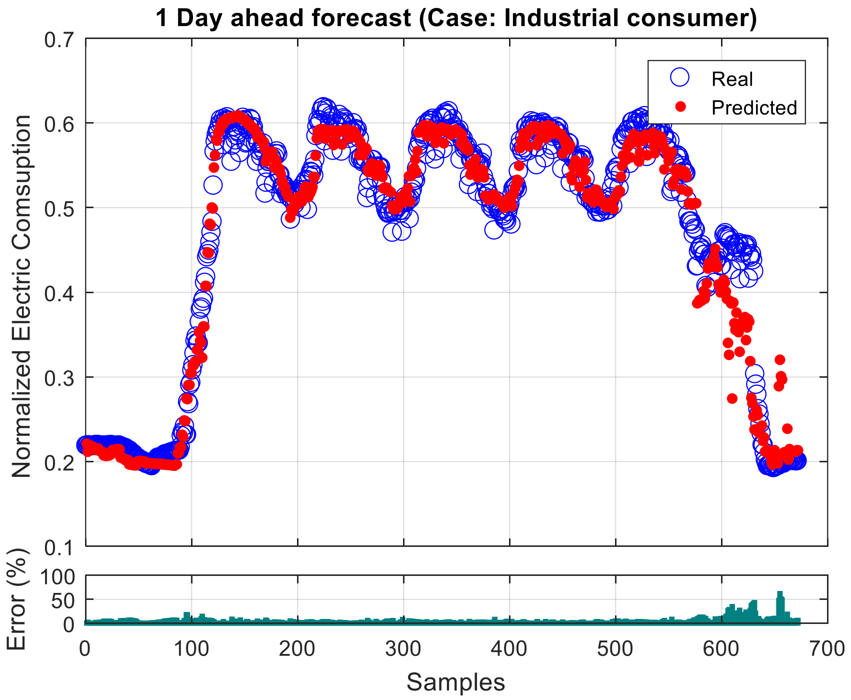


Figure 40. Load forecast correspondent to the ensemble model #1. Learning algorithm combination: CCNN-FFNN-FFNN. Number of clusters/number of base learners: 2/1. Lags implemented: 1, 2, 8. Approximations implemented: 0.

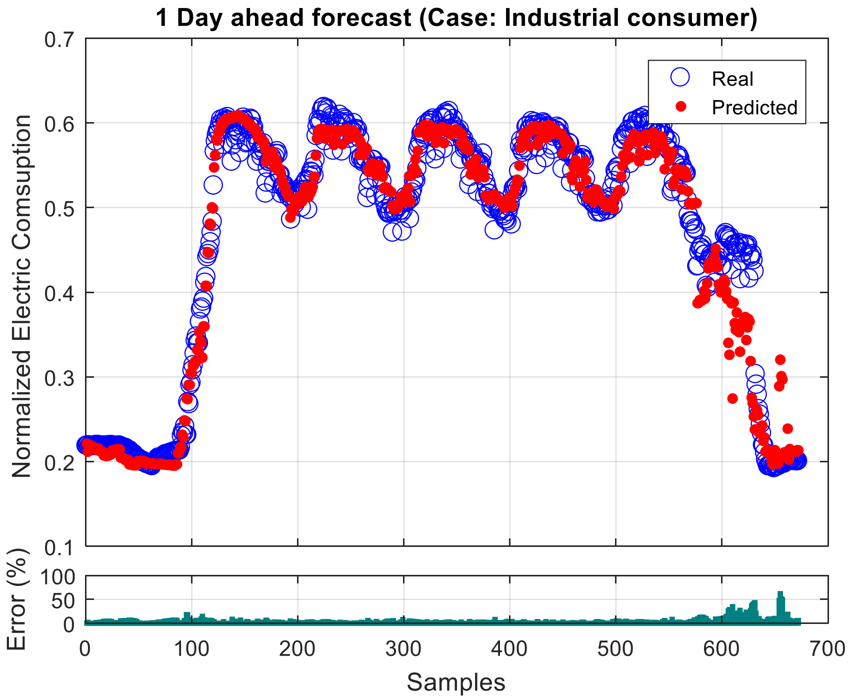


Figure 41. Load forecast correspondent to the ensemble model #1. Learning algorithm combination: CCNN-FFNN-FFNN. Number of clusters/number of base learners: 2/1. Lags implemented: 0. Approximations implemented: 5.

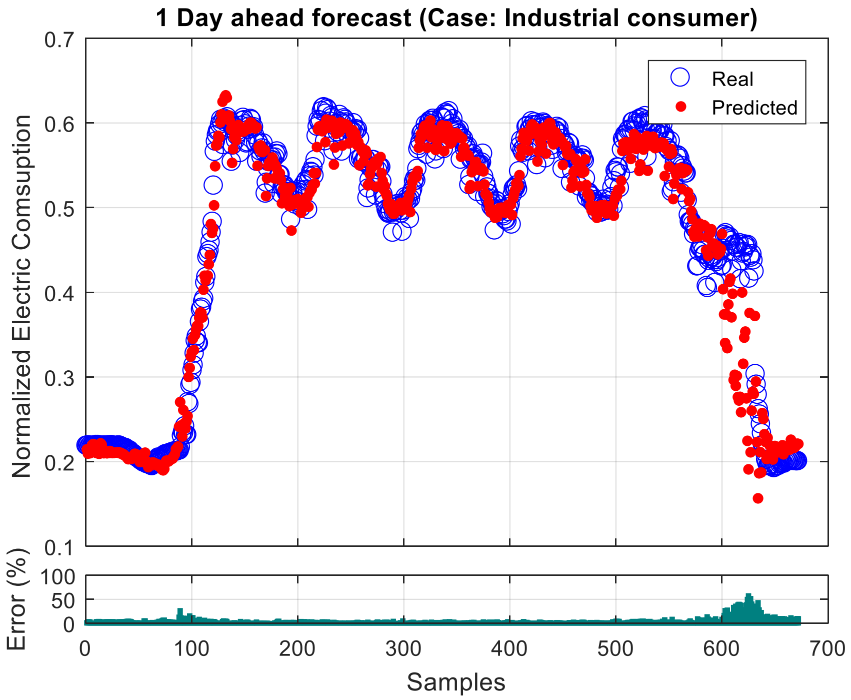


Figure 42. Load forecast correspondent to the ensemble model #1. Learning algorithm combination: Reg. Tree-FFNN-FFNN. Number of clusters/number of base learners: 1/8. Lags implemented: 0. Approximations implemented: 5.

The graphics gives a hint about the real problem behind the models, all of them lack of the ability to forecast the Friday-Saturday transition, and some of them are not so good tracking the slope transition Sunday-Monday. They exhibit tendencies most typical of office workers.

But there is a reason for the lack of accuracy on these two zones, the uncertainty implicit on the zones with switching dynamics. In order to make a hypothesis about this problem we have checked the slopes of weeks previous to the forecasted week, and they doesn't tend to remain stable neither, adopting a different twist every week.

These dynamics are not unique of an industrial scenario, it is also reflected on the representative data sets of the Spanish university case, and the residential building case. On the other hand this dynamic is not so strong at the regional consumption scenario: Australian data set.

Is important highlight that despite of the errors obtained on this two highly volatile zones, the weekly forecasting error remains low; being the last one the most important forecast for energy dispatch, energy purchase and monthly load forecasting decisions.

In demonstrate the adaptability of our ensemble learning strategy for load forecasting, we have conduct a series of forecasting on different scenarios. Using the best ensemble model configuration model #1: Reg. Tree-FFNN-FFNN 5/8, obtained for the industrial scenario, and adding some options such as the possibility to include lags or approximations we have created three models to test the evolution of the ensemble model trough scenarios.

We also have included the second model on accuracy, model #3: CCNN-FFNN-FFNN 2/1. It will serve as a control ensemble model, providing a second point of view about the ensemble learning adaptation to the current data set from the perspective of a model with strong base learners.

Simple learners, FFNN - ANFIS, are also implemented in order to highlight the ensemble model accuracy, and give a measure about the background difficulty to model the current set. Given the smoothness of the data set employed on the **Table 21** was natural expect low forecasting errors, the ensemble models have demonstrated the convenience of the multi-resolution approach at the same time achieve low scores.

Table 21. MAPE and RMSE of the ensemble models tested over the regional consumption scenario: Australian data set.

#	Ensemble Model	# Clusters / # Base leaners	Time Scaling Lags	SWT Number of approximations	RMSE	MAPE
1	Reg. Tree-FFNN-FFNN	5 / 8			2,19	2,62
2	Reg. Tree-FFNN-FFNN	5 / 8		5	1,67	2,24
3	Reg. Tree-FFNN-FFNN	5 / 8	1, 2, 8		2,69	2,85
4	CCNN-FFNN-FFNN	2 / 1	1, 2, 8		2,18	2,57
	ANFIS				3,63	4,21
	FFNN				3,59	3,89

The level of aggregation on the commercial consumption scenario employed on the **Table 22** is high but the random walk on the series becomes a serious problem for ensemble models and simple learners. The optimal modelling strategy for this set, rather than a fine grained feature specialization, seems to be a less grained clustering such as in the model #3.

The data set on the **Table 23** also possess a high uncertainty intraday but remain stable during the week. These results support the ensemble learning strategy proposed on this thesis at the same time that opens a window for more applications on the time series prediction.

Table 22. MAPE and RMSE of the ensemble models tested over the commercial consumption scenario: Spanish university case.

#	Ensemble Model	# Clusters / # Base leaners	Time Scaling Lags	SWT Number of approximations	RMSE	MAPE
1	Reg. Tree-FFNN-FFNN	5 / 8			7,06	7,75
	Reg. Tree-FFNN-FFNN	5 / 8		5	8,19	10,11
	Reg. Tree-FFNN-FFNN	5 / 8	1, 2, 8		7,18	8,04
3	CCNN-FFNN-FFNN	2 / 1	1, 2, 8		7,17	7,40
	ANFIS				9,84	11,71
	FFNN				9,04	8,70

Table 23. MAPE and RMSE of the ensemble models tested over the Residential consumption scenario: Bristol building.

#	Ensemble Model	# Clusters / # Base leaners	Time Scaling Lags	SWT Number of approximations	RMSE	MAPE
1	Reg. Tree-FFNN-FFNN	5 / 8			5,81	4,53
	Reg. Tree-FFNN-FFNN	5 / 8		5	5,25	4,40
	Reg. Tree-FFNN-FFNN	5 / 8	1, 2, 8		5,77	4,64
3	CCNN-FFNN-FFNN	2 / 1	1, 2, 8		5,49	4,26
	ANFIS				7,13	6,94
	FFNN				7,21	7,35

3.5 Conclusion

On this section major advances on the construction of adaptable, accurate and semiautomatic modelling approaches has been made. The approaches presented has comprised a diversity of techniques, starting at the signal processing of the data using a time-frequency decomposition technique in order to extract information for the modelling process. Continuing on the exploration of auto-generated models from a refined evolutionary programming technique, and finalizing with the development of a novel strategy to build, train and optimize ensemble methods.

All the approaches presented had follow a clear path of evolution in order to guarantee the accuracy of the ensemble and ensure the maximum extraction of knowledge form the data sets. The Wavelet-ANFIS modelling approach brings an improvement over the plain learner algorithm, corroborating that an exploration the time-frequency traits of the signals is a clear path to elaborates more accurate models.

The Cartesian genetic programming approach brings the opportunity to test the hypothesis of a fully adapted model to the data, since it grew/evolve of it, making the term “adaptability” the key term to judge the performance of this algorithm instead of the accuracy; which also obtain a good performance over a typical learning algorithm.

Inspired of these approaches, a novel ensemble learning strategy have been develop as a way to retain as much as possible traits contained on the signal, following the bias and volatility of the signal on a more precise and controlled way. In order to achieve this objectives was necessary implement clustering methods to control the features to model, subsampling to grow diversity among sets of base learners, and more important: declare a hierarchy to combine all the predictions.

Finally, in order capture more diversity among the elements of the ensemble model we have tested the introduction of multi-resolution techniques, which has corroborated the conclusions obtained from the Wavelet-ANFIS approach. In order to test the accuracy and diversity extracted by the model we have carried out a set of 24 goodness measures.

These measures has help us to conclude:

- Ensemble models based on weak base learners retain mayor diversity than those ones based on strong learners
- Accuracy improves with the use of multi-resolution elements but they must be selected previous analysis.
- Our ensemble learning strategy offers good results for series with high volatility on short time forecasting.

CHAPTER

Conclusion and future outlook

This chapter summaries the conclusions of the present dissertation, the conclusion may be divided in three sections an outlook for further research.

Contents

4.1 Conclusions

As the reader could notice, this thesis has presented an exercise of pattern recognition. We could say that time-series analysis, modelling algorithms, and forecast analysis, are procedures to find and identify elements susceptible to be parameterized and classified. An example of this statement is the usage of the time-series analysis for the identification of the trend and seasonal components on the signal, and the posterior parameterization of those using base-learners.

On the other hand, an example of hard parameterization is founded on the estimation of the conditional variance. This estimation require a residual error analysis, which counts residual errors as a mixture of errors such as: modelling errors due to innovations, modelling error due improper model structure, errors due to data sample and corrupted data.

The ensemble learning strategy presented on this thesis is complex during the energy study, which is a once-time implementation, but once time the best parameters are defined for the load profile the prediction is easy/fast to execute without almost no configuration. The ensemble strategy was also created to adapt to any volatile scenario, inclusive those ones with high dynamics loads.

Our novel ensemble strategy has been inspired by techniques original from time series forecasting field and weather prediction models, extending the validity to very short forecasting horizons. In fact, the training of a model could happen in less than quarter hour prediction interval, allowing to present a novel forecasting methodology with great accuracy and without over calculations.

The ensemble load forecasting approaches presented on this dissertation, have been designed in order to minimize the errors presented on three so-called features of the load forecasting methods. The conclusions obtained from the implementation presented are summarized on the following sections based on their respective association with the features called adaptability, accuracy, and automatization.

4.1.1 Adaptability in terms of the load profile volatility

Deterministic modelling algorithms generally perform a training based on the unconditional mean, except on the case of autoregressive models which can be considered as conditional mean regressor. Such methods can be improved by the implementation of resampling statistical methods such as Monte Carlo methods, or their particular cases bootstrapping and stochastic sampling methods such as MCMC could be also considered. These methods will serve to obtain the conditional variance through several run on the case of base-learners.

The measuring and modelling the conditional variance is the base of the adaptability, because the variance (volatility) on a load profile is defined accordingly to user scenario. Ensemble learning approaches based on boosting methods, rather than averaging methods, reduce effectively the bias and provides a better understand of the aggregation of explanatory models constructed for the dominant effects on the load profile.

But, the reduction of the bias is a consequence of the accurate conditional mean track. In order to calculate the conditional variance is necessary integrate a model suitable for this task. On this scenario, assumptions like implement a density estimator as base learner (BMLS - MARS), genetically evolved models (CGP - NCGP), or strong learners, are equally valid and have been discussed and compared on this thesis.

4.1.2 Accuracy in terms of Bias tracking

Accuracy is a term linked with the estimation of the conditional mean of a time series on a forecasting horizon. Although, the best methods to control the conditional mean tracking (active bias tracking, online forecasting) implies short-term sequential updating instead of recursive forecast, the decision on the implementation of any alternative is penalized according with the load forecasting algorithm requirements (training method, database size, algorithm complexity).

Results has demonstrated that short-time forecast made by NN family algorithms could be relative quicker than a fraction of the forecast horizon under large data sets. But apart of being possible, this leads to the question of the consequences of rely on an online training instead of use modern one-step ahead forecast.

The principal objective of the approaches presented on this thesis are define a series of structured models driven by the special features founded on the load profiles scenarios. Features such as seasonal components, cyclical effects, day patterns, and autoregressive terms linked to recency effect are some of the key elements used to build these ensemble structures.

However, in order to increase the precision of the model, linked to innovations that cannot be described on the previous elements, methods based on multi-resolution elements are discussed. It has been shown that models that aggregates these elements achieve a significant improvement but it always depend on the quality of the data available.

As we can see, errors due to innovations plays a central role on the analysis of the residuals, but the study is not limited to them. Our research performs an exploratory analysis of the errors inherent to the forecast algorithm chosen. The contributions to the residual error due to model structure are explored using a variety of model to compare their contributions on to an ensemble approach, and the errors due to model parameters are intended to be cancelled by the use of samplers.

Finally the errors due to corrupted data or quantization of the signal are considered elements susceptible to be largely filtered on the preprocessing stages, residuals of those errors must be considered part of the innovations.

4.1.3 Automatization

Energy study comprised as the actions necessary to install an EMS on a new user always will depend on human expertise, there is no possibility to codify the expertise, just make it more affordable to be used in form of software packages or documented procedures.

In order to catalog the procedures necessary to carry out a load forecast, it has been proposed a modelling strategy from the identification and filter of the variables, up to the presentation of the forecast results, and the measures necessary to validate the reliability of the model according to the dynamics of the user scenario. The packing of these procedures on a fully automated method is left to user discretion.

Although the main focus of this thesis is load forecasting, we wish have maintained a statistical spirit describing a way to understand the problems of the energy time-series on energy management. Following that motivation, the next sections will present some of the promissory research paths on the study of the time-series forecast with applications to energy profiles.

4.2 Future outlook

Time-series forecasting applied to load profiles have been mostly restrained to conditional mean regression models, and the expert supervision to achieve an accurate adaptation to data sets. On this section, based on the analysis of the current literature, the advance of the smart grids, and the multy-agent implementations, we propose two possible trend topics for the upcoming 10 years.

4.2.1 Enhancement of the stochastic modelling

The approaches presented on this thesis have been searching for the correct modelling of the volatility presented on the load profiles; but, as we stated along the previous chapters, there is no point on assume Gaussian distribution on the error variance (modelling methods presented) or assume that conditional prediction error could be correlated with previous states (GARCH models).

Although, we have presented a robust approach to minimize the sources of error on the modelling, we encourage to readers to test if probabilistic approaches such Markov Chain derivatives could increase the modelling accuracy. Readers can use those probabilistic approaches to compute a short term load forecast or a non-parametric density estimation using a similar procedure as the explained on the GARCH modelling.

Also, readers can combine such approaches with piecewise functions or artificial intelligence algorithms such as NN [129]. Although the recency of those approaches is a big attractor, these predict the conditional variance based on the previous stage, reducing their reliability for more than one step ahead. As recommendation, readers could use the conditional mean extracted from sampler methods as base for the conditional distribution.

Other methods focused on the probability density based on the data projections such as probabilistic principal components analyzers [130], used on image compression and hand written digital recognition could be also explored as an alternative to found the volatile in terms of artificial dimensions easy to analyze.

Referenced books on the field of stochastic forecasting [131]–[134] are introduced to reader in order to make their own research on stochastic modelling.

4.2.2 Dynamic recognition of modelling set up parameters using pattern recognition.

Deep learning is the current fashion… yes, as you tough, welcome to the rollercoaster of fashion. The 80's were a factory of trends such as Yuppie's, an acronym for 'Young Upwardly Mobile Professional Person', who style perfectly define the new hipster-informal way of dress. But, 80's was recognized also for diverse applications of multilayer perceptron's a.k.a multilayer neural networks.

This means that the large applications on the identification of complex parameter on a signal without effort was existing since 30 years ago and recently investigators decide give a second chance to a dusty technology. This rusty technology could also provide the ideal autonomy searched for the load forecasting process, from the time-series analysis passing by the modelling method and finishing on the custom forecast.

Future research works could use pattern recognition in order to measure key indicators such as mean, variance, mode, median, season effects, volatility, etc. in order to customize the load modelling method and the statistics provided by the forecasting process.

CHAPTER

5

Thesis results dissemination

This chapter

Contents

5.1 Related journal and conference publications

5.1.1 Journals

August/2016	Giacometto, F.; Capelli F.; Riba J. R.;Romeral, L.; Sala, E. “Thermal Response Estimation in Substation Connectors Using Data-Driven Models.” In Advances in Electrical And Computer Engineering. vol.16, no.3, pp.25-30, 2016-Aug. doi: 10.4316/AECE.2016.03004.

5.1.2 Conferences

November/2015	Giacometto, F.; Sala, E.; Kampouropoulos, K.; Romeral, L.; “Short Term Load Forecasting using Cartesian Genetic Programming: an Efficient Evolutive Strategy Case: Australian electricity market.” In 41st Annual Conference of the IEEE Industrial Electronics Society IECON 2015. pp.5087-5094, 2015-Nov., doi: 10.1109/IECON.2015.7392898. • **Best presentation recognition** in the Computational Intelligence session.
November/2015	Giacometto, F.; Romeral, L.; Sala, E.; Capelli F.; Riba J. R., “Temperature Rise Estimation of Substation Connectors Using Data-Driven Models Case: Thermal conveccion response.” In 41st Annual Conference of the IEEE Industrial Electronics Society IECON 2015. pp.3957-3962, 2015-Nov., doi: 10.1109/IECON.2015.7392717.
October/2012	Giacometto, F.; Cardenas, J.J.; Kampouropoulos, K.; Romeral, J.L., "Load forecasting in the user side using wavelet-ANFIS," in IECON 2012 - 38th Annual Conference on IEEE Industrial Electronics Society , pp.1049-1054, 25-28 Oct. 2012, doi: 10.1109/IECON.2012.6388575.

5.1.3 Collaborative work

November/2014	Sala, E.; Kampouropoulos, K.; Giacometto, F.; Romeral, L., "Smart multi-model approach based on adaptive Neuro-Fuzzy Inference Systems and Genetic Algorithms," in Industrial Electronics Society, IECON 2014 - 40th Annual Conference of the IEEE, pp.288-294, 2014-Nov., doi: 10.1109/IECON.2014.7048513.

November/2014	Sala, E.; Kampouropoulos, K.; Giacometto, F.; Romeral, L., "Smart multi-model approach based on adaptive Neuro-Fuzzy Inference Systems and Genetic Algorithms," in Industrial Electronics Society, IECON 2014 - 40th Annual Conference of the IEEE , vol., no., pp.288-294, Oct. 29 2014-Nov. 1 2014, doi: 10.1109/IECON.2014.7048513.
September/2012	Cardenas, J.J.; Giacometto, F.; Garcia, A.; Romeral, J.L., "STLF in the user-side for an iEMS based on evolutionary training of Adaptive Networks," in Emerging Technologies & Factory Automation (ETFA), 2012 IEEE 17th Conference on , pp.1-8, 17-21 Sept. 2012, doi: 10.1109/ETFA.2012.6489626

5.2 Collaborations in technologic transfer projects

October 2011 – March 2015	**EUROENERGEST**: Increase of automotive car industry competitiveness through an integral and artificial intelligence driven energy management system. Funded under FP7-ICT. Responsibilities: • Writing and coordination of the work package 4: "Theoretical models for the load forecasting system" • Analysis of the load consumption on the SEAT car manufacturing plant using data mining • Design and coding of data-driven models for a load forecasting service at an energy management application
January 2012 – July 2015	**OPTIENER**: energy efficiency optimization on building sector. Funded under IMPACTO Spanish projects. Responsibilities: • Writing and coordination of the work package 4: " Configuration of dynamic models for the energy load system" • Analysis of the load consumption on the buildings using data mining • Design and coding of data-driven models for a load forecasting service at an energy management application
August 2013 – August 2014	**EFINDPRO**: Process And Living Lab For Industry Energy Efficiency. Funded under KIC InnoEnergy projects. Responsibilities: • Writing and coordination of the work package 2: " theoretical automatic and tunable models for power consumption, ready for plant integration " • Analysis of the HVAC load consumption at industrial users using data mining • Design and coding of data-driven models for a load forecasting service at an energy management application

REFERENCES

[1] A. Wilstam, “Dividing load economically among power plants by use of the kilowatt &#x2014; Killowatt-hour curve,” *A.I.E.E., J.*, vol. 47, no. 6, pp. 430–432, 1928.

[2] R. F. Hamilton, “The Summation or Load Curves,” *Am. Inst. Electr. Eng. Trans.*, vol. 63, no. 10, pp. 729–735, 1944.

[3] H. A. Dryar, “The Effect of Weather on the System Load,” *Am. Inst. Electr. Eng. Trans.*, vol. 63, no. 12, pp. 1006–1013, 1944.

[4] R. B. Rowson, “Electricity supply&#2014;a statistical approach to some particular problems,” *Proc. IEE - Part II Power Eng.*, vol. 99, no. 68, pp. 151–167, 1952.

[5] J. G. Gruetter, “The Application of Business Machines to Electrical Utility Load Forecasting [includes discussion],” *Power Appar. Syst. Part III. Trans. Am. Inst. Electr. Eng.*, vol. 74, no. 3, p. 1, 1955.

[6] I. S. Moghram and S. Rahman, “Analysis and evaluation of five short-term load forecasting techniques,” *IEEE Trans. Power Syst.*, vol. 4, no. 4, pp. 1484–1491, 1989.

[7] H. S. Hippert, C. E. Pedreira, and R. C. Souza, “Neural networks for short-term load forecasting: a review and evaluation,” *Power Syst. IEEE Trans.*, vol. 16, no. 1, pp. 44–55, 2001.

[8] C.-N. Lu, H.-T. Wu, and S. Vemuri, “Neural network based short term load forecasting,” *Power Syst. IEEE Trans.*, vol. 8, no. 1, pp. 336–342, 1993.

[9] S. T. Chen, D. C. Yu, and A. R. Moghaddamjo, “Weather Sensitive Short-Term Load Forecasting Using Nonfully Connected Artificial Neural Network,” *IEEE Trans. Power Syst.*, vol. 7, no. 3, pp. 1098–1105, 1992.

[10] M. Hisham Choueiki, C. A. Mount-Campbell, and S. C. Ahalt, “Building a ‘quasi optimal’ neural network to solve the short-term load forecasting problem,” *IEEE Trans. Power Syst.*, vol. 12, no. 4, pp. 1432–1439, 1997.

[11] M. Hisham Choueiki, “Implementing a weighted least squares procedure in training a neural network to solve the short-term load forecasting problem,” *IEEE Trans. Power Syst.*, vol. 12, no. 4, pp. 1689–1694, 1997.

[12] J. Vermaak and E. C. Botha, “Recurrent neural networks for short-term load forecasting,” *Power Syst. IEEE Trans.*, vol. 13, no. 1, pp. 126–132, 1998.

[13] S. Rahman and R. Bhatnagar, “An expert system based algorithm for short term load forecast,” *IEEE Trans. Power Syst.*, vol. 3, no. 2, pp. 392–399, 1988.

[14] J. S. R. Jang, “ANFIS: Adaptive-Network-Based Fuzzy Inference System,” *IEEE Trans. Syst. Man Cybern.*, vol. 23, no. 3, pp. 665–685, 1993.

[15] J. P. S. Catalão, H. M. I. Pousinho, and V. M. F. Mendes, “Hybrid wavelet-PSO-ANFIS approach for short-term electricity prices forecasting,” *IEEE Trans. Power Syst.*, vol. 26, no. 1, pp. 137–144, 2011.

[16] S. Fan, L. Chen, and W. J. Lee, “Short-term load forecasting using comprehensive combination based on multimeteorological information,” *IEEE Trans. Ind. Appl.*, vol. 45, no. 4, pp. 1460–1466, 2009.

[17] M. Hanmandlu and B. K. Chauhan, “Load forecasting using hybrid models,” *IEEE Trans. Power Syst.*, vol. 26, no. 1, pp. 20–29, 2011.

[18] R. Noori, M. A. Abdoli, A. Farokhnia, and M. Abbasi, “Results uncertainty of solid waste generation forecasting by hybrid of wavelet transform-ANFIS and wavelet transform-neural network,” *Expert Syst. Appl.*, vol. 36, no. 6, pp. 9991–9999, 2009.

[19] J. Yokoyama and H. D. Chiang, “Short Term Load Forecasting Improved by Ensemble and its Variations,” *2012 Ieee Power Energy Soc. Gen. Meet.*, 2012.

[20] R. Zhang, Z. Y. Dong, Y. Xu, K. Meng, and K. P. Wong, “Short-term load forecasting of Australian National Electricity Market by an ensemble model of extreme learning machine,” *Iet Gener. Transm. Distrib.*, vol. 7, no. 4, pp. 391–397, 2013.

[21] A. Kaur, H. T. C. Pedro, and C. F. M. Coimbra, “Ensemble re-forecasting methods for enhanced power load prediction,” *Energy Convers. Manag.*, vol. 80, pp. 582–590, 2014.

[22] E. M. Burger and S. J. Moura, “Gated ensemble learning method for demand-side electricity load forecasting,” *Energy Build.*, vol. 109, pp. 23–34, 2015.

[23] A. Tuohy, J. Zack, S. E. Haupt, J. Sharp, M. Ahlstrom, S. Dise, E. Grimit, C. Mohrlen, M. Lange, M. G. Casado, J. Black, M. Marquis, and C. Collier, “Solar Forecasting: Methods, Challenges, and Performance,” *IEEE Power Energy Mag.*, vol. 13, no. 6, pp. 50–59, 2015.

[24] J. W. Taylor and R. Buizza, “Neural network load forecasting with weather ensemble predictions,” *Ieee Trans. Power Syst.*, vol. 17, no. 3, pp. 626–632, 2002.

[25] J. R. Dong, C. Y. Zheng, G. Y. Kan, M. Zhao, J. Wen, and J. Yu, “Applying the ensemble artificial neural network-based hybrid data-driven model to daily total load forecasting,” *Neural Comput. Appl.*, vol. 26,

no. 3, pp. 603–611, 2015.

[26] K. Siwek and S. Osowski, “Short term load forecasting model in the power system using ensemble of predictors,” *2007 Ieee Instrum. Meas. Technol. Conf. Vols 1-5*, pp. 512–517, 2007.

[27] K. Siwek, S. Osowski, and R. Szupiluk, “Ensemble Neural Network Approach for Accurate Load Forecasting in a Power System,” *Int. J. Appl. Math. Comput. Sci.*, vol. 19, no. 2, pp. 303–315, 2009.

[28] J. Casazza and F. Delea, “Electric Energy Consumption,” in *Understanding Electric Power Systems:An Overview of the Technology and the Marketplace*, Wiley-IEEE Press, 2004, pp. 41–53.

[29] S. Fan, L. Chen, and W. J. Lee, “Short-Term Load Forecasting Using Comprehensive Combination Based on Multimeteorological Information,” *Ieee Trans. Ind. Appl.*, vol. 45, no. 4, pp. 1460–1466, 2009.

[30] D. O. Jermain, “Comparative models for electrical load forecasting : D.W. Bunn and E.D. Farmer (Editors), John Wiley, New York (1985), 232 pp. £24.95 (hardback),” *Long Range Planning*, vol. 19, no. 6. 1986.

[31] P. F. Pai, “Hybrid ellipsoidal fuzzy systems in forecasting regional electricity loads,” *Energy Convers. Manag.*, vol. 47, no. 15–16, pp. 2283–2289, 2006.

[32] C. H. Yu, “Resampling methods: concepts, applications, and justification. ,” *Pract. Assessment, Res. Eval.*, 2003.

[33] A. E. Guntermann, “Are Energy Management Systems Cost Effective?,” *IEEE Trans. Ind. Appl.*, vol. IA-18, no. 6, pp. 616–625, 1982.

[34] S. Buchanan, R. Taylor, and S. Paulos, “The electricity consumption impacts of commercial energy management systems,” *IEEE Trans. Power Syst.*, vol. 4, no. 1, pp. 213–219, 1989.

[35] P. Du and N. Lu, “Appliance commitment for household load scheduling,” *IEEE Trans. Smart Grid*, vol. 2, no. 2, pp. 411–419, 2011.

[36] S. Lee, B. Kwon, and S. Lee, “Joint energy management system of electric supply and demand in houses and buildings,” *IEEE Trans. Power Syst.*, vol. 29, no. 6, pp. 2804–2812, 2014.

[37] P. B. Luh, “Building Energy Management: Integrated Control of Active and Passive Heating, Cooling, Lighting, Shading, and Ventilation Systems,” *IEEE Trans. Autom. Sci. Eng.*, vol. 10, no. 3, pp. 588–602, 2013.

[38] M. Manic, D. Wijayasekara, K. Amarasinghe, and J. J. Rodriguez-Andina, “Building Energy Management Systems: The Age of Intelligent and Adaptive Buildings,” *IEEE Ind. Electron. Mag.*, vol. 10, no. 1, pp. 25–39, 2016.

[39] Navigant Research, *Buildings Energy Management Systems. Software, Services, and Hardware for Energy Efficiency and Systems Optimization: Global Market Analysis and Forecasts*. .

[40] Mordor Intelligence, *European Building Energy Management Systems Market - Growth, Trends, and Forecasts (2015-2020)*. FEBRUARY 2016, 2016.

[41] P. Bertoldi, *ESCO Market Report 2013*. 2013.

[42] (IEA) International Energy Agency, *Energy efficiency Market Report 2015*. Paris: OECD Publishing, 2015.

[43] J. C. Van Gorp, “Enterprising energy management,” *IEEE Power Energy Mag.*, vol. 2, no. 1, pp. 59–63, 2004.

[44] Z. Zhiping Wang, X. Xiantang Liu, and B. Baojian Wu, “The study and application of steel enterprise energy management system,” in *2011 International Conference on Electric Information and Control Engineering*, 2011, pp. 4667–4670.

[45] Y. Yuting Yang, Q. Qiong Liu, and L. Ling Song, “Design and development of management system for enterprise energy consumption and cost,” in *2015 IEEE International Conference on Communication Problem-Solving (ICCP)*, 2015, pp. 327–330.

[46] Navigant Research, *Industrial Energy Management Systems. Software, Services, and Hardware for Energy Efficiency and Systems Optimization: Global Market Analysis and Forecasts*. Navigant Consulting, Inc.

[47] S. Aman, Y. Simmhan, and V. K. Prasanna, “Energy management systems: state of the art and emerging trends,” *IEEE Commun. Mag.*, vol. 51, no. 1, pp. 114–119, 2013.

[48] P. Palensky and D. Dietrich, “Demand Side Management: Demand Response, Intelligent Energy Systems, and Smart Loads,” *Ind. Informatics, IEEE Trans.*, vol. 7, no. 3, pp. 381–388, 2011.

[49] D. Lee and C.-C. Cheng, “Energy savings by energy management systems: A review,” *Renew. Sustain. Energy Rev.*, vol. 56, pp. 760–777, 2016.

[50] Q. Ding, H. Zhang, T. Huang, and J. Zhang, “A Holiday Short Term Load Forecasting Considering Weather Information,” *2005 Int. Power Eng. Conf.*, pp. 2–5, 2005.

[51] W. Dai and P. Wang, “Application of Pattern Recognition and Artificial Neural Network to Load Forecasting in Electric Power System,” *Third Int. Conf. Nat. Comput. (ICNC 2007)*, no. Icnc, pp. 381–385, 2007.

[52] F. Veltman, L. G. Marin, D. Saez, L. Gutierrez, and A. Nuñez, “Prediction interval modelling tuned by an improved teaching learning algorithm applied to load forecasting in microgrids,” *2015 IEEE Symp. Ser. Comput. Intell.*, pp. 651–658, 2015.

[53] J. W. Taylor and R. Buizza, “Neural network load forecasting with weather ensemble predictions,” *IEEE Trans. Power Syst.*, vol. 17, no. 3, pp. 626–632, 2002.

[54] G. Gross and F. D. Galiana, “Short-term load forecasting,” *Proc. IEEE*, vol. 75, no. 12, pp. 1558–1573, 1987.

[55] H. S. Hippert, C. E. Pedreira, and R. C. Souza, “Neural networks for short-term load forecasting: a review and evaluation,” *IEEE Trans. Power Syst.*, vol. 16, no. 1, pp. 44–55, 2001.

[56] H. K. Alfares and M. Nazeeruddin, “Electric load forecasting: literature survey and classification of methods,” *Int. J. Syst. Sci.*, vol. 33, no. 1, 2002.

[57] S. K. Aggarwal, L. M. Saini, and A. Kumar, “Electricity price forecasting in deregulated markets: A review and evaluation,” *Int. J. Electr. Power Energy Syst.*, vol. 31, no. 1, pp. 13–22, 2009.

[58] D. De Silva and X. Yu, “A data mining framework for electricity consumption analysis from meter data,” *IEEE Trans. Ind. Informatics*, vol. 7, no. 3, pp. 399–407, 2011.

[59] B. L. B. Liu and R. Y. R. Yang, “A novel method based on PCA and LS-SVM for power load forecasting,” *2008 Third Int. Conf. Electr. Util. Deregul. Restruct. Power Technol.*, no. 978, pp. 759–763, 2008.

[60] W. C. Hong, “Electric load forecasting by seasonal recurrent SVR (support vector regression) with chaotic artificial bee colony algorithm,” *Energy*, vol. 36, no. 9, pp. 5568–5578, 2011.

[61] H. M. I. Pousinho, V. M. F. Mendes, and J. P. S. Catalão, “Short-term electricity prices forecasting in a competitive market by a hybrid PSO-ANFIS approach,” *Int. J. Electr. Power Energy Syst.*, vol. 39, no. 1, pp. 29–35, 2012.

[62] T. Hong and S. Fan, “Probabilistic Electric Load Forecasting: A Tutorial Review,” *Int. J. Forecast. under Rev.*, vol. 32, no. 3, pp. 1–32, 2014.

[63] G. Giannakis, V. Kekatos, and N. Gatsis, “Monitoring and optimization for power grids: A signal processing perspective,” *IEEE Signal Process. Magzine*, no. August 2013, pp. 107–128, 2013.

[64] Q. Huang and D. O. Wu, “Flatten a curved space by Kernel [Applications Corner],” *IEEE Signal Process. Mag.*, vol. 30, no. 5, pp. 132–136, 2013.

[65] S. Amakali, “Development of models for short-term load forecasting using artificial neural networks,” *Thèse*, pp. 1–224, 2008.

[66] E. Sala, K. Kampouropoulos, F. Giacometto, and L. Romeral, “Smart multi-model approach based on adaptive Neuro-Fuzzy Inference Systems and Genetic Algorithms,” in *IECON 2014 - 40th Annual Conference of the IEEE Industrial Electronics Society*, 2014, pp. 288–294.

[67] K. Kampouropoulos, J. J. Cardenas, F. Giacometto, and L. Romeral, “An energy prediction method using Adaptive Neuro-Fuzzy Inference System and Genetic Algorithms,” in *Industrial Electronics (ISIE), 2013 IEEE International Symposium on*, 2013, pp. 1–6.

[68] E. A. Feinberg and D. Genethliou, “Load Forecasting,” *Appl. Math. Restructured Electr. Power Syst.*, pp. 269–285, 2006.

[69] K. Liu, S. Subbarayan, R. R. Shoults, M. T. Manry, C. Kwan, F. I. Lewis, and J. Naccarino, “Comparison of very short-term load forecasting techniques,” *IEEE Trans. Power Syst.*, vol. 11, no. 2, pp. 877–882, 1996.

[70] F. X. Diebold, *Elements of Forecasting*, 3rd editio. mason, ohio: South-Western College Pub, 2006.

[71] Man Xu, Zongxiang Lu, Ying Qiao, Ningbo Wang, Shiyuan Zhou, and Yanhong Ma, “Study on the adaptability of day-ahead wind power forecast system for on-site use,” in *2013 IEEE Power & Energy Society General Meeting*, 2013, pp. 1–5.

[72] O. Hyde and P. F. Hodnett, “An adaptable automated procedure for short-term electricity load forecasting,” *IEEE Trans. Power Syst.*, vol. 12, no. 1, pp. 84–94, 1997.

[73] M. T. Hagan and S. M. Behr, “The Time Series Approach to Short Term Load Forecasting,” *IEEE Trans. Power Syst.*, vol. 2, no. 3, pp. 785–791, 1987.

[74] N. F. Hubele and C.-S. Cheng, “Identification of seasonal short-term load forecasting models using statistical decision functions,” *Power Syst. IEEE Trans.*, vol. 5, no. 1, pp. 40–45, 1990.

[75] T. M. Peng, N. F. Hubele, and G. G. Karady, “Advancement in the application of neural networks for short-term load forecasting,” *IEEE Trans. Power Syst.*, vol. 7, no. 1, pp. 250–257, 1992.

[76] I. Guyon and A. Elisseeff, “An Introduction to Variable and Feature Selection,” *J. Mach. Learn. Res.*, vol. 3, no. 3, pp. 1157–1182, 2003.

[77] J. H. Friedman, “Multivariate Adaptive Regression Splines,” *Ann. Stat.*, vol. 19, pp. 1–67, 1991.

[78] C. C. Holmes, “Bayesian regression with multivariate linear splines,” 2001.

[79] I. E. Review and S. Fattahi, “A Comparative Study of Parametric and Nonparametric Regressions,” vol. 16, no. 30, 2011.

[80] J. H. Friedman, “MULTIVARIATE ADAPTIVE REGRESSION SPLINES*.”

[81] R. E. Kass, “Bayesian curve-fitting with free-knot splines,” pp. 1055–1071, 2001.

[82] A. B. Koehler, “Time Series Analysis and Forecasting with Applications of SAS and SPSS,” *Int. J.*

Forecast., vol. 17, no. 2, pp. 301–302, 2001.
[83] C. Chatfield, *Time-Series Forecasting*. 2000.
[84] R. N. Onody, G. M. Favaro, and E. R. Cazaroto, “Gaussian and Exponential GARCH models,” pp. 1–7.
[85] R. Rojas, “Neural networks: a systematic introduction,” *Neural Networks*, p. 502, 1996.
[86] J. Li, “An empirical comparison between SVMs and ANNs for speech recognition,” *First Instr. Conf. Mach. …*, pp. 4–7, 2003.
[87] T. Takagi and M. Sugeno, “Fuzzy Identification of Systems and Its Applications to Modeling and Control,” *IEEE Trans. Syst. Man Cybern.*, vol. SMC-15, no. 1, pp. 116–132, 1985.
[88] J. F. Miller and O. Cgp, “Cartesian Genetic Programming,” 2000.
[89] J. A. Walker and J. F. Miller, “The automatic acquisition, evolution and reuse of modules in Cartesian genetic programming,” *IEEE Trans. Evol. Comput.*, vol. 12, no. 4, pp. 397–417, 2008.
[90] S. Harding and J. F. Miller, “Evolution of robot controller using Cartesian Genetic Programming,” *Proc. 8th Eur. Conf. Genet. Program.*, pp. 120–131, 2005.
[91] M. Mahsal Khan, A. Masood Ahmad, G. Muhammad Khan, and J. F. Miller, “Fast learning neural networks using Cartesian genetic programming,” *Neurocomputing*, vol. 121, pp. 274–289, 2013.
[92] S. Harding, V. Graziano, J. Leitner, and J. Schmidhuber, “MT-CGP: Mixed Type Cartesian Genetic Programming,” in *Proceedings of the fourteenth international conference on Genetic and evolutionary computation conference - GECCO ’12*, 2012, p. 751.
[93] S. Harding, J. F. Miller, and W. Banzhaf, “Self modifying cartesian genetic programming: Parity,” in *2009 IEEE Congress on Evolutionary Computation, CEC 2009*, 2009, pp. 285–292.
[94] C. L. Alonso, J. L. Montana, J. Puente, and C. E. Borges, “A new Linear Genetic Programming approach based on straight line programs: some Theoretical and Experimental Aspects,” *Int. J. Artif. Intell. Tools*, vol. 18, no. 5, pp. 757–781, 2009.
[95] J. L. Montaña, C. L. Alonso, C. E. Borges, and J. De La Dehesa, “Penalty functions for genetic programming algorithms,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2011, vol. 6782 LNCS, no. PART 1, pp. 550–562.
[96] J. F. Miller and S. L. Smith, “Redundancy and computational efficiency in cartesian genetic programming,” *IEEE Trans. Evol. Comput.*, vol. 10, no. 2, pp. 167–174, 2006.
[97] B. W. Goldman and W. F. Punch, “Length Bias and Search Limitations in Cartesian Genetic Programming,” *Gecco’13 Proc. 2013 Genet. Evol. Comput. Conf.*, pp. 932–940, 2013.
[98] J. F. Miller and P. Thomson, “Cartesian genetic programming,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2000, vol. 1802, pp. 121–132.
[99] S. Z. S. Zhou and Z. S. Z. Sun, “Can Ensemble Method Convert a ‘Weak’ Evolutionary Algorithm to a ‘Strong’ One?,” *Int. Conf. Comput. Intell. Model. Control Autom. Int. Conf. Intell. Agents, Web Technol. Internet Commer.*, vol. 2, pp. 68–74, 2005.
[100] D. Opitz and R. Maclin, “Popular Ensemble Methods : An Empirical Study,” vol. 11, pp. 169–198, 1999.
[101] Q. Dai, “A competitive ensemble pruning approach based on cross-validation technique,” *Knowledge-Based Syst.*, vol. 37, pp. 394–414, 2013.
[102] Zhi-Hua Zhou, *Ensemble Methods*, vol. 2. 2013.
[103] B.-H. Mevik, V. H. Segtnan, and T. Næs, “Ensemble methods and partial least squares regression,” *J. Chemom.*, vol. 18, no. 11, pp. 498–507, 2004.
[104] L. Breiman, “Bagging predictors,” *Mach. Learn.*, vol. 24, no. 2, pp. 123–140, Aug. 1996.
[105] H. Lee, E. Kim, and W. Pedrycz, “A new selective neural network ensemble with negative correlation,” *Appl. Intell.*, no. March, pp. 1–11, 2012.
[106] R. E. Banfield, L. O. Hall, K. W. Bowyer, and W. P. Kegelmeyer, “Ensemble diversity measures and their application to thinning,” *Inf. Fusion*, vol. 6, no. 1, pp. 49–62, 2005.
[107] N. Li, Y. Yu, and Z. H. Zhou, “Diversity regularized ensemble pruning,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2012, vol. 7523 LNAI, no. PART 1, pp. 330–345.
[108] L. I. Kuncheva and C. J. Whitaker, “Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy,” *Mach. Learn.*, vol. 51, no. 2, pp. 181–207, 2003.
[109] Z. H. Zhou and N. Li, “Multi-information ensemble diversity,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2010, vol. 5997 LNCS, pp. 134–144.
[110] F. Shu and C. Luonan, “Short-term load forecasting based on an adaptive hybrid method,” *Power Syst. IEEE Trans.*, vol. 21, no. 1, pp. 392–401, 2006.
[111] M. Hanmandlu and B. K. Chauhan, “Load forecasting using hybrid models,” *IEEE Trans. Power Syst.*, vol. 26, no. 1, pp. 20–29, 2011.

[112] A. J. Conejo, M. a Plazas, R. Espínola, S. Member, and A. B. Molina, "Day-Ahead Electricity Price Forecasting Using the Wavelet Transform and ARIMA Models," *IEEE Trans. Power Syst.*, vol. 20, no. 2, pp. 1035–1042, 2005.

[113] A. J. R. Reis and A. P. A. Silva, "Feature Extraction via Multiresolution Analysis for Short-Term Load Forecasting," *IEEE Trans. Power Syst.*, vol. 20, no. 1, pp. 189–198, 2005.

[114] S. G. Mallat, "A Theory for Multiresolution Signal Decomposition: The Wavelet Representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 11, no. 7, pp. 674–693, 1989.

[115] S. S. S. Hosseini and A. H. Gandomi, "Short-term load forecasting of power systems by gene expression programming," *Neural Comput. Appl.*, vol. 21, no. 2, pp. 377–389, 2012.

[116] M. M. Khan, G. M. Khan, and J. F. Miller, "Evolution of neural networks using Cartesian Genetic Programming," in *IEEE Congress on Evolutionary Computation (CEC 2010)*, 2010.

[117] S. Harding, J. F. Miller, and W. Banzhaf, "Evolution, development and learning using self-modifying cartesian genetic programming," in *GECCO '09: Proceedings of the 11th Annual conference on Genetic and evolutionary computation*, 2009, pp. 699–706.

[118] S. L. Harding, J. F. Miller, and W. Banzhaf, "Self-modifying cartesian genetic programming," *Nat. Comput. Ser.*, vol. 43, pp. 101–124, 2011.

[119] G. W. Downs and D. Rocke, "Interpreting Heteroscedasticity," *Am. J. Pol. Sci.*, vol. 23, no. 4, pp. 816–828, 1979.

[120] G. Pan, "On a levene type test for equality of two variances," *J. Stat. Comput. Simul.*, vol. 63, no. 1, pp. 59–71, Apr. 1999.

[121] B. B. Schultz, "Levene's test for relative variation," *Syst. Zool.*, vol. 34, no. 4, pp. 449–456, 1985.

[122] B. R. Chang and H. F. Tsai, "Forecast approach using neural network adaptation to support vector regression grey model and generalized auto-regressive conditional heteroscedasticity," *Expert Syst. Appl.*, vol. 34, no. 2, pp. 925–934, 2008.

[123] B. Trawiński, M. Smętek, Z. Telec, and T. Lasota, "Nonparametric statistical analysis for multiple comparison of machine learning regression algorithms," *Int. J. Appl. Math. Comput. Sci.*, vol. 22, no. 4, pp. 867–881, 2012.

[124] B. Chapman, G. Jost, and R. Van Der Pas, "Using OpenMP: Portable Shared Memory Parallel Programming (Scientific and Engineering Computation)," *Book*, 2007.

[125] G. Kumar and K. Kumar, "The Use of Artificial-intelligence-based Ensembles for Intrusion Detection: A Review," *Appl. Comp. Intell. Soft Comput.*, vol. 2012, no. Id, p. 21:21-21:21, 2012.

[126] G. Giacinto, F. Roli, and G. Fumera, "Design of Effective Multiple Classifier Systems by Clustering of Classifiers," *15th Int. Conf. Pattern Recognit. - ICPR*, pp. 160–163, 2000.

[127] A. Lazarevic and Z. Obradovic, "Effective pruning of neural network classifier ensembles," *IJCNN'01. Int. Jt. Conf. Neural Networks. Proc. (Cat. No.01CH37222)*, vol. 2, pp. 796–801, 2001.

[128] B. Bakker and T. Heskes, "Clustering ensembles of neural network models," *Neural Networks*, vol. 16, no. 2, pp. 261–269, 2003.

[129] M. Sperandio, D. P. Bernardon, and V. J. Garcia, "Building forecasting Markov models with Self-Organizing Maps," *Univ. Power Eng. Conf. (UPEC), 2010 45th Int.*, 2010.

[130] M. E. Tipping and C. M. Bishop, "Mixtures of probabilistic principal component analyzers.," *Neural Comput.*, vol. 11, no. 2, pp. 443–482, 1999.

[131] D. Fusion, *No Title*. .

[132] H. Durrant-whyte, "Multi Sensor Data Fusion," *Methods*, pp. 1–153, 2006.

[133] J. P. Lesage, "Applied Econometrics using MATLAB," *Rev. Lit. Arts Am.*, vol. 204, no. 1, pp. 1–332, 1999.

[134] C. M. Bishop, *Pattern Recognition and Machine Learning*, vol. 53, no. 9. 2013.

[135] W. Gilchrist, *Statistical forecasting*. Wiley, 1976.

[136] W. H. Williams and M. L. Goodman, "A Simple Method for the Construction of Empirical Confidence Limits for Economic Forecasts," *J. Am. Stat. Assoc.*, vol. 66, no. 336, pp. 752–754, Dec. 1971.

[137] "ENERGY SECURITY FOR ALL AUSTRALIANS – Australian Energy Market Operator." [Online]. Available: http://www.aemo.com.au/. [Accessed: 09-Mar-2017].

[138] J. Zhai, R. Palmer, C. Brownlee, and M. Lyons, "POWER SYSTEM OPERATING PROCEDURE – LOAD FORECASTING," 2014.

[139] (IEA) International Energy Agency, *Energy Statistics of OECD Countries 2015*. Paris: OECD Publishing, 2015.

Appendix

This chapter

Contents

Appendix A: Symbols

x	Variable
$\boldsymbol{x}$	Vector
$\boldsymbol{A}$	Matrix
$\boldsymbol{I}$	Identity matrix
$\boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{Y}}$	Input and output spaces
$\mathcal{D}$	Probability distribution
D	Data sample (data set)
$\mathcal{N}$	Normal distribution
$\boldsymbol{\mathcal{U}}$	Uniform distribution
$\mathcal{H}$	Hypothesis space
$\boldsymbol{H}$	Set of hypotheses
$h(.)$	Hypothesis (learner)
$\boldsymbol{\mathcal{L}}$	Learning algorithm
$p(.)$	Probability density function
$\boldsymbol{p}(.\|.)$	Conditional probability density function
P (.)	Probability mass function
$\mathrm{P}(.\|.)$	Conditional probability mass function
$\mathbb{E}_{.\sim\mathcal{D}}[f(.)]$	Mathematical expectation of function f(·) to · under distribution $\mathcal{D}$. $\mathcal{D}$ and/or, · is ignored when the meaning is clear
$\mathbb{E}[.]$	Unconditional mean
$var_{.\sim\mathcal{D}}[f(.)]$	Variance of function $f(\cdot)$ to · under distribution D
$err(.)$	Error function
$\{\dots\}$	Set
$(\dots)$	Row vector
$(\dots)^T$	Column vector
$\|.\|$	Size of data set

$\|.\|$	L2-norm
$\mathbb{I}(.)$	Indicator function which takes 1 if . is true, and 0 otherwise
$\Psi(.)$	Custom function transform (time-frequency or time scaling)

Appendix B: Load forecasting error estimation

On the Chapter 2 the load forecasting theory was presented on general terms, concepts as the classification of the load forecasting and the loads to forecast, the identification of their characteristics, the preprocessing techniques and the methods to model the load behavior were presented. But, the methods to measure the significance of the modelling techniques or the load forecasting strategies presented where introduced.

We deliberately left the conceptual framework of the error estimation as a separate section due to the analysis of the error could be addressed from several points of view. An example of this could be stated on the representation of the generalization error for ensemble models, wich could be divided on bias, variance, and covariance, or in the average base learner error and the average ambiguity generated by them.

On this appendix we will introduce concepts that has been addressed on the novel approaches implemented on this thesis, and not have been fully explained. The first section describe the load forecasting error theory viewed from a structured combination of errors produced by a mixture of sources, i.e., model parameters, model structure, data innovation.

The second section will introduce the statistical tools used to measure the goodness of the forecast and model. The third section introduce the techniques employed to set the intervals where the prediction is likely supposed to be, knowing as confidence intervals. Last section is dedicated to show some statistical operators employed.

Forecasting error decomposition

First of all, let's remember the definition of the **residuals** or so called error, they are constituted by a sum of all those components that the model fails in their explanation whether through incapacity or lack of data. When we introduce the description of the BMLS algorithm, we consider the dependent variable $\boldsymbol{y}$ as result of a function of the explanatory variables plus a Gaussian noise term.

Eq. 75 $$\boldsymbol{y_t = f(x_t) + \varepsilon_t \qquad and \quad \varepsilon_t \backsim N(0, \sigma^2)}$$

The function f represents the real mathematical relationship that exist among the inputs and desired output, it can be only guessed or approximate by the forecasting approach. **The Gaussian noise** represents the assumption of a homoscedasticity posterior distribution, it means that BMLS present the **error due to innovation** as a term with **fix distribution**.

Later, on the description of the GARCH algorithm the equation changed to

Eq. 76 $\boldsymbol{y_t = f(x_t) + \varepsilon_t \quad and \quad \varepsilon_t \backsim N(0, \sigma_t^2)}$

Where the **error due to innovations** ε_t this time is **time dependent**. It denotes a sequence of independent modelling errors with zero mean and local conditional variance of the process σ_t. This guarantee the assumption of heteroskedasticity on the posterior distribution, it means that GACH models presents the error due to innovation as a term with time dependent distribution.

This formula is follows the same principle of a random walk, where $\boldsymbol{\phi_t}$ is a random variable that describe the probability for the next step, and $\boldsymbol{h}$ the time between subsequent intervals.

Eq. 77 $\boldsymbol{y_{t+h} = y_t + \phi_h \quad and \quad \phi_t \backsim N(0, \sigma_t^2)}$

If the direction and longitude of the step becomes only dependent of the last sample and no other previous position, the random walk is considered with a Markov property.

As you will notice, the function f is the ideal relationship that remains hidden, and the modelling procedure tries to mimic this function. Typically, a supervised learning algorithm is employed to map the samples of the **x** variables against the **y** variable. On this process the innovation error is assumed as another component to be modeled. But, more error is added when the fitted model must predict **y** using estimated **x** variables, i.e., the estimated temperature.

Based on this statement, we can consider the GARCH algorithm as the top performer on the study of the independent & uncorrelated residuals, consider it a way to understand the distribution of the innovation error on the predicted samples. Although, the innovation error comprised as a part of the modelling on next section we will study the use of the **kernel density estimators** (**KDE**), in charge of assert the distribution of the forecast.

Consequently, any modelling procedure, presents the following equation. Where the forecasted dependent variable is presented as the estimated function $\hat{f}$, it could be consider as an ensemble of models $\hat{f} = F(\hat{f}_1, \hat{f}_2, \dots, \hat{f}_n)$.

Eq. 78 $\boldsymbol{\hat{y}_t = \hat{f}(x_t)}$

Then, the square loss error is presented as

Eq. 79 $\boldsymbol{err = \mathbb{E}(y_t - \hat{y}_t)^2 = \mathbb{E}\big(f(x_t) - \hat{f}(x_t) + \varepsilon_t\big)^2 \quad and \quad \varepsilon_t \backsim N(0, \sigma_t^2)}$

And, the residuals presented as

Eq. 80 $\boldsymbol{r_t = y_t - \hat{y}_t = f(x_t) - \hat{f}(x_t) + \varepsilon_t \quad and \quad \varepsilon_t \backsim N(0, \sigma_t^2)}$

The based on this equation, the residual error could be divided on two terms, the residual error because modelling, called model uncertainty.

Eq. 81 $\mathrm{r}_t^m = \boldsymbol{f(x_t) - \hat{f}(x_t)}$

And the residual error because innovations, which must be generalized as uncertainty about the experimental framework (EF).

Eq. 82 $\mathrm{r}_t^{ef} = \boldsymbol{\varepsilon_t \quad and \quad \varepsilon_t \backsim N(0, \sigma_t^2)}$

On the following figure the behavior of the bias and variance associated with an ensemble model is presented. Notice how the model attempt to be close to the real function, but is deflected because his learning is conditioned to data already corrupt by the EF.

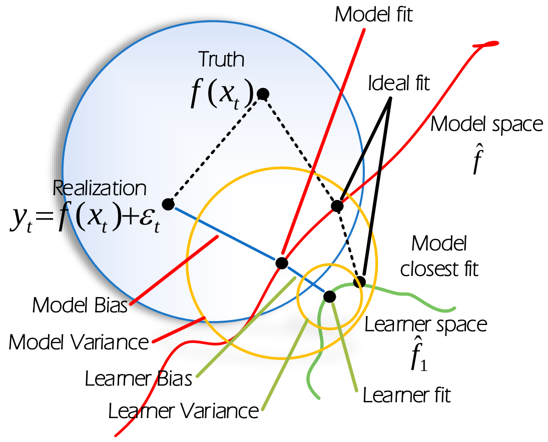


Figure 43. Schematic of the bias-variance behavior on an ensemble model.

Uncertainty decomposition on load forecasting

As it was introduced before, the main sources of uncertainty in any statistical could be categorized based on their importance.

- **Uncertainty due to modelling**: The primary and larger source of error comes from the modelling error. This also comprises two errors, the error about the structure of the model, and the error about the correct estimation of the model parameters.
    - **Uncertainty about the structure of the model**: Is related with the lack of knwoledge about the dinamic that will be modeled and for consequence choose a modelling algorithm with poor performance, or a bad simplification of the model, i.e., pruning. It error also include: the incorrect selection of the numebr of input variables, and a bad guess of the variable dependency.

  - **Uncertainty about estimates of the model parameters**: Assuming the model structure is known, a inadecuate asd use of the initial conditions or the learning algorithms could lead on this error.

Then, the residual error because modelling, is presented as

Eq. 83
$$\mathbf{r}_t^m = \boldsymbol{f}(x_t) - \hat{\boldsymbol{f}}(x_t) = \boldsymbol{\varepsilon}_t^{param} + \boldsymbol{\varepsilon}_t^{structure}$$

- **Uncertainty about the experimental framework (EF)**: This will include unexplained random variation in the observed variables, i.e., innovations, as well as measurement and recording errors such as corrupt data.

The errors due to the experimental framework could be considered as the intrinsic noise of the data set, being the lowest boundary on the expected error of any learning algorithm over the target. It should become clear to the reader that doubts about the model have a more serious effect on forecast accuracy than uncertainty arising from other sources. An example of this is the bias/variance decomposition, which is mostly influenced by modelling errors.

The bias measures how closely, the average estimation of, the learning algorithm is able to approximate the target; the variance measures how much, the estimation of, the learning approach fluctuates for different training sets of the same size.

Since the intrinsic noise is difficult to estimate, it is often subsumed into the bias term. Thus, the generalization error is broken into, the bias term which describes the error of the learner in expectation, and the variance term which reflects the sensitivity of the learner to variations in the training samples.

The errors due to innovations could be approached taking on consideration the innovations as a conditional variable that depends only of the last observation of the signal. It means that new values of the innovations follows a conditional probability distribution.

Probabilistic modelling techniques are focused on model the conditional probability distribution of the target at time t-1 in order to predict the next prediction interval, or measure the conditional probability distribution of the predictors employed for the regression. Algorithms as bayesian networks and markov models are examples of those approaches.

On this thesis the conditional probability distribution of the innovations is not considered individually due to his low significance on short-term forecast. This can be partially explained on the stationary covariance property of the load profile.

However, the thesis author recommends accept his high relevance is on very short-term forecast or economic forecast. This is due stochastic nature of the loads time series became excessively obvious at low ranges of time and economic forecast most of the time doesn’t have a stationary covariance.

Estimation of the prediction intervals

When we performs a forecast, we must decide if the forecast will be (1) a single number “the best punctual guess”, (2) a range of numbers into which the future value can be expected to fall at a certain percentage of the time, or (3) an entire probability distribution for the future value. These conditions define the type of forecast, and represent (1) a **point forecast**, (2) an **interval forecast**, (3) a **density forecast**.

Point forecast provides an easy digest information about the guess of the time series, however unpredictable “shocks” will produce errors on the prediction. Thus, we may want to know the degree of confidence that we have on a particular point forecast. More specifically, we want to know the uncertainty associated to the point forecast.

An **interval forecast** has several characteristics, first, the length of the interval conveys information about the forecast uncertainty. Second, it contains more information of a point forecast, you can construct several point forecast by using the media of the interval.

A **density forecast** gives the entire density (probability distribution) of the future value of the series. The density forecast also convey more information than interval forecast because, given a density function, any interval forecast at any confidence level could be easily created. For example, given a the future values of a series y, with a Gaussian distribution $N(\mu, \sigma^2)$, an interval forecast at 95% could follow the equation $y = \mu \pm 1.96\sigma$.

Notice that **density forecast** require as central component a **density function**, it represent the distribution that will follow the future values of the series. This functions could be, (1) **parametric**, such as Gaussian or Bayesian, which follows a defined equation and remain similar for all the point forecast. Or (2) **non-parametric**, functions obtained from regressions over the distribution of the time series, GARCH models and Kernel estimators are part of these group.

However, in practice point forecast are the most common forecast made, interval forecast are distant second, and density forecast are rarely made. There is at least two reason for this. First, the construction of interval and density forecast requires either, (1) additional and possibly incorrect assumptions relative to those required to for the construction of the point forecast, Or (2) advanced and computer-intensive methods involving extensive simulations, i.e., Monte Carlo methods.

Second, it is often easier to understand and take action based upon a set of point forecasts relative to an interval or density forecast. Another forecast type of particular relevance to event outcome and event timing forecasting is the **probability forecast**. On the following figure the differences among the forecast are obvious.

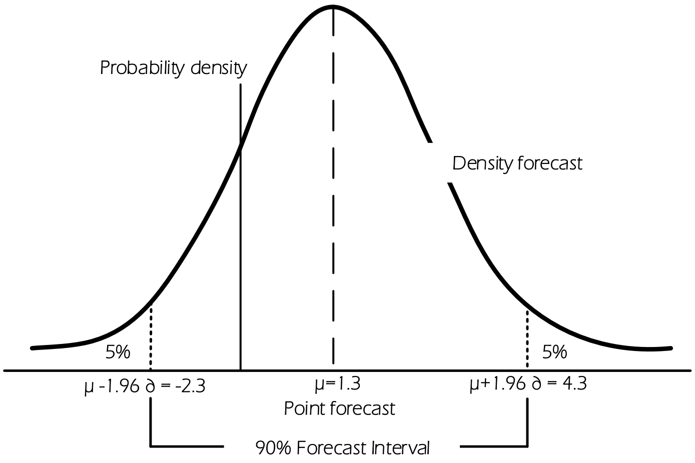


Figure 44. Types of forecast.

On the next subsections we will dig deeper on the consequences of have a prediction interval that relies on a density function, estimated by any sort of regression or conventional such as Gaussian distribution, conditioned by the forecast timing.

Parametric interval forecast (homoscedastic assumption)

Point forecast are made based on the mean estimation this means that **modelling algorithms are conditional mean experts**, because their base their forecast on events occurred at a time t. This mean forecast give us a hint about the learning method, the modelling algorithm must be learning from the inherent temporal distribution of the time series by means of certain rules.

As we notice on the description of the modelling algorithms, sometimes we start supposing that time series follows a normal distribution (Gaussian distribution). But non-linear algorithms made his way learning directly from the data without initial assumptions, lead on heteroskedasticity mean driven models. On the next section we will discuss this topic.

The most common procedure to calculate the forecast interval is assume a **homoscedastic distribution** for all the points of the prediction, and associate to them a predefined **probability density function** (**PDF**) such as a Gaussian function, T-student distribution etc. Most of the **forecast intervals** found on literature follows a Gaussian distribution. A $100(1-\propto)\%$ P.I. for a horizon *h* is given by:

Eq. 84 $$\boldsymbol{P}(\hat{\mathbf{y}}|\mathbf{t},\mathbf{h}) = \hat{\boldsymbol{y}}_{\boldsymbol{t+h}} \pm \boldsymbol{z}_{\propto/2}\sqrt{\boldsymbol{\sigma}^{2}(\hat{\boldsymbol{y}})}$$

Where $\boldsymbol{z}_{\propto/2}$ denotes the percentage point of an standard normal distribution with a proportion $\propto/\mathbf{2}$ above it, $\propto/\mathbf{2}$ area within each of the two tails, $1-\propto$ probability between the interval limits. Others author consider the equation for within-sample prediction intervals

Eq. 85 $$\boldsymbol{P}(\hat{\mathbf{y}}|\mathbf{t},\mathbf{h}) = \hat{\boldsymbol{y}}_{\boldsymbol{t+h}} \pm \boldsymbol{z}_{\propto/2}\sqrt{\boldsymbol{\sigma}^{2}(\hat{\boldsymbol{e}})}$$

This formula only apply for prediction because the future errors are unknown, also his value has been widely criticized (Chatfield [83], p. 201).

Based on the supposition of homoscedasticity we could infer that forecast present a distribution as the following graph shows.

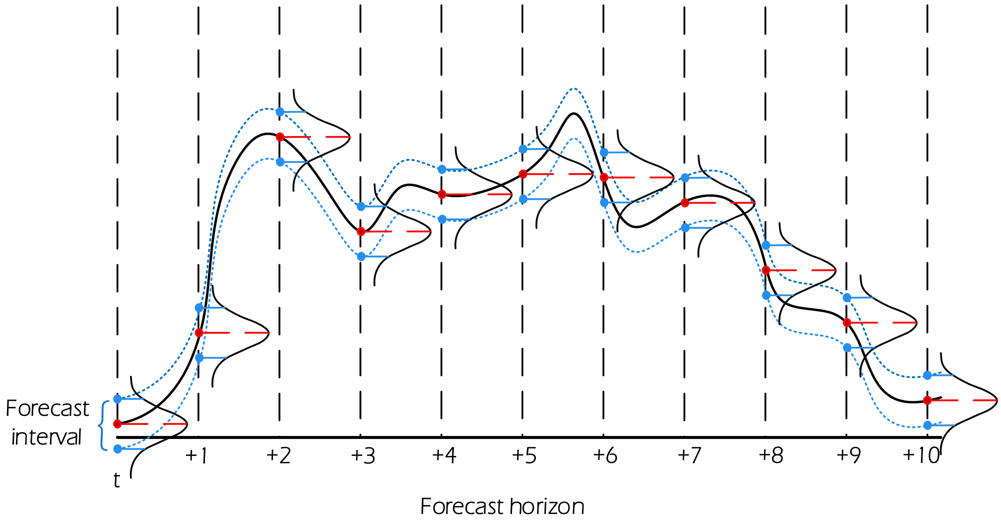


Figure 45. Representation of a homoscedastic forecast.

Non-parametric interval forecast (heteroskedastic assumption)

In nonparametric interval statistics no assumptions are made on the underlying probability of the forecast model, canceling the assumption of a normal distribution around the point forecast. The appeal of nonparametric methods lies in their ability to reveal structure in data that might be missed by classical parametric methods.

Then, we can assume that assumptions normality or a homoscedastic distribution is too naïve. The real time series presents distributions with extravagant skewness and kurtosis due to the so called “shocks”. If we perform a prediction over the time series past, we will find traces of a heteroskedastic distribution as the following figure shows.

Notice that the histogram of the time series, for a given time, could be translated on a probability function. This non-parametric regression methods is exploited to produce empirically based probabilistic intervals will be explained later.

On this section we will present some alternatives to construct a density function estimated $\widehat{\boldsymbol{P}}$ from the observed data with an unknown density function $\boldsymbol{P}$. The classical approach to estimate the probability density is to assume a parametric model, but inferences derived from it can lead to misleading interpretations of the prediction interval.

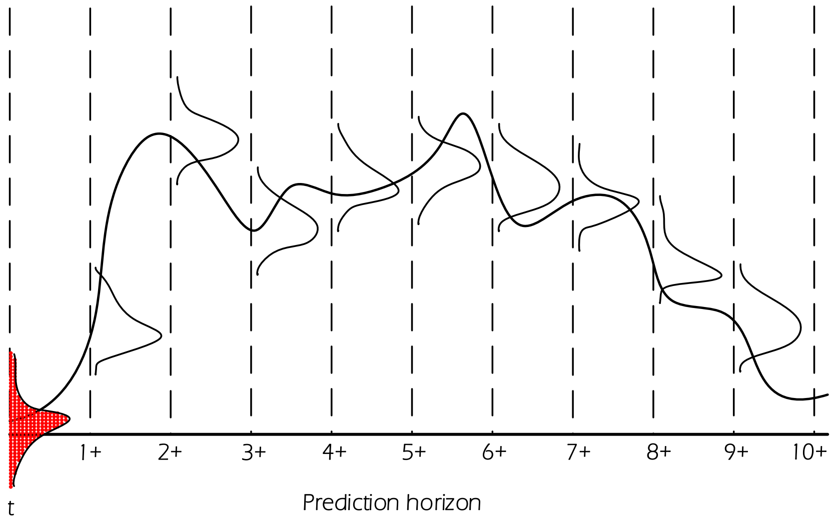


Figure 46. Prediction of a time series with heteroskedastic distribution.

In non-parametric density estimation less rigid assumptions are placed over the functional expression of the density. The resulting density estimator is more flexible and the time series are “allowed to speak for themselves”. This results in a powerful tool for exploratory data analysis: visualization of data sets, classification... and a natural framework to formally define and handle data sets.

Non-parametric models also has drawbacks; kernel estimators depend on a smoothing parameter, typically hard to select because there is no unique obvious “optimal choice” for it; they require large samples because the theoretical motivation for the estimators is usually asymptotic; they suffer dimensionality curse, as soon the data dimension grows the estimators require larger and larger sample sizes.

Since the multivariate nonparametric density estimation methods are generalizations of univariate ones, we will introduce first the univariate proposals. We are familiarized with the **simplest density estimator**, **the histogram**. On the last figure we could see how a density estimation could be inferred from dividing the histogram on infinite classes.

Kernel estimators

A simple estimator (based on a similar idea to that of the histogram) is the moving window estimator

Eq. 86 $$\boldsymbol{f_n(t) = \frac{1}{n\,2h_n}\sum_{i=1}^{n}(t - x_i)\cdot \mathbb{I}(-h_n, h_n)}$$

This can be generalized by replacing the normalized uniform density of the histogram interval $\mathbb{I}(-1,1)/2$ with another density function called kernel density function, or **kernel density estimator** (KDE).

Eq. 87 $$\boldsymbol{f_n(t) = \frac{1}{n\,h_n}\sum_{i=1}^{n} K\left(\frac{t-x_i}{h_n}\right)}$$

Notice that kernel provides a smoothed version of the histogram, among the possible kernel choices we have the Epanechnikov kernel

Eq. 88 $$K(x) = \frac{3}{4}(1 - x^2) \cdot \mathbb{I}(-1, 1)\ (x)$$

The Biweight kernel

Eq. 89 $$K(x) = \frac{15}{16}(1 - x^2)^2 \cdot \mathbb{I}(-1, 1)\ (x)$$

The Triweight kernel

Eq. 90 $$K(x) = \frac{35}{32}(1 - x^2)^3 \cdot \mathbb{I}(-1, 1)\ (x)$$

The Gaussian kernel

Eq. 91 $$K(x) = \frac{1}{\sqrt{2\pi}} exp(-\frac{x^2}{2}) \cdot \mathbb{I}(-1, 1)\ (x)$$

The Student's t-distribution kernel, with ν as the number of degrees of freedom

Eq. 92 $$K(x) = \frac{\Gamma\left(\frac{v+1}{2}\right)}{\sqrt{v\,\pi}\,\Gamma\left(\frac{v}{2}\right)}\left(1 + \frac{x^2}{v}\right)^{-\frac{v+1}{2}} \cdot \mathbb{I}(-1, 1)(x) = \frac{1}{\sqrt{v}\,\mathrm{B}\left(\frac{1}{2}, \frac{v}{2}\right)}\left(1 + \frac{x^2}{v}\right)^{-\frac{v+1}{2}} \cdot \mathbb{I}(-1, 1)\ (x)$$

And, finally The Uniform kernel

Eq. 93 $$K(x) = \frac{1}{2} \cdot \mathbb{I}(-1, 1)\ (x)$$

Other kernels could be based on the regression of the time series such as the Nadaraya-Watson estimator

Eq. 94 $$\widehat{m}(x) = \frac{n^{-1}\sum_{i=1}^{n} K_h(x - x_i) y_i}{n^{-1}\sum_{j=1}^{n} K_h(x - x_j)}$$

The difference between prediction intervals and confidence intervals

On the previous section, reader has observe the creation of the probability density functions based on the past data. A density function modeled from histograms and softened by some kernels. These are called confidence intervals, and we could say that share the homoscedastic assumption because it is based on the data available.

But, prediction intervals (PI) and confidence intervals (CI) are not the same thing. R. Hyndman make a clear distinction about them. A PI is an interval associated with a random variable yet to be observed, with a specified probability of the random variable lying within the interval. Prediction intervals can arise in Bayesian or frequentist statistics.

A CI is an interval associated with a parameter and is a frequentist concept. The parameter is assumed to be non-random but unknown, and the confidence interval is computed from data.

Because the data are random, the interval is random. A 95% CI will contain the true parameter with probability 0.95. That is, with a large number of repeated samples, 95% of the intervals would contain the true parameter.

Empirically based prediction intervals

The estimation of the density intervals on out-of-sample data have the same principles of a forecasting procedure. When theoretical function of the forecast interval is not available, instead of assume a normal distribution, the reader should use a more computationally intensive approach based either on (1) using the properties of the observed distribution of the within-sample prediction errors, or (2) based on simulation or resampling methods. Precursors on the method (1) will be following described.

Gilchrist [135] make a prediction on the within-sample data, finding the within-sample prediction errors at 1+,2+,3+, … steps ahead from all the available time origins, and then finding the variance of these errors at each lead time over the period of fit. Then assuming normality an approximate $100(1-\propto)\%$ P.I. for a h-step-ahead horizon h is given by:

Eq. 95 $$P(\hat{y}|\mathrm{N},\mathrm{h}) = \hat{y}_N(\mathbf{h}) \pm z_{\propto/2}\,\sigma_{e,h} = \hat{y}_N(\mathbf{h}) \pm z_{\propto/2}\sqrt{\sigma^2(\hat{e}_N(h))}$$

If N is small $z_{\propto/2}$ is replaced by the corresponding percentage point of the t-distribution with ν degrees of freedom, where σ is also based on these degrees of freedom. Then he extends these prediction intervals over the future of the time series. However, the values of variance are unreliable because are based on in-sample residuals rather than on out-of-sample forecast error.

Williams and Goodman [136], divide the past data in two parts: training and validation, the validation errors are used to estimate the density distribution of the series. The resulting errors are much more like true forecast errors because the use of out-of-sample data. Then, the model is refitted with one additional observation in the training data and one less on the validation data, and so on, making this a heuristic approach.

This heuristic approach could be consider as an intention to calculate the **conditional probability density function** (**CPDF**), because the forecast jumps each sample at time. The authors found that the distribution of the forecast errors tended to approximate a gamma distribution rather than a normal distribution.

PDF's were constructed using the percentage points of the empirical distribution, thereby avoiding any distributional assumptions, such as normality. Promising results were obtained corroborating that authors were ahead of its time. However, although the approach is attractive in principle, it seems to have been little used in practice, presumably because of the heavy penalization imposed by the computational demands.

Resampling methods, sometimes called *Monte Carlo* approaches, are often the most preferable way to obtain the empirical distribution function. These methods are explained as follows [83].

"Given a probability time-series model, it is possible to simulate both past and future behavior by generating an appropriate series of random innovations and hence constructing a sequence of possible past and future values. This process can be repeated many times, leading to a large set of possible sequences, sometimes called pseudo-data. From such a set it is possible to evaluate P.I.s at different horizons by finding the interval within which the required percentage of future values lie". Due to the use of a heuristic approach such that, is generally assumed that the model has been identified correctly.

As reader already notice, this concept correspond to a heuristic sampling of the innovations from some assumed parametric distribution, often normal. It literally make use of fragmentation of the data on sequences that no longer respect the temporal arrangement. An alternative is the *bootstrapping*, it effectively approximates the theoretical distribution of innovations, by the empirical distribution of the observed residuals, because it is a distribution-free approach.

On **Simulation**, the main idea is to use the knowledge about the primary structure of the model to generate a sequence of possible future values and find a forecast interval. Of course, because the nature of the model is deterministic, the use of resampling is necessary. Although, they are not based on a proper probability method, but rely instead on a set of recursive equations involving observed and forecast values.

In a time series context, resampling would make no sense because successive observations are not independent, but are correlated through time. This explains why time series data are usually bootstrapped by *resampling the residual errors rather than the actual observations*. Just because residuals are expected to be at least approximately independent.

However, the reader should be aware that it is generally more difficult to resample correlated data, such as time series, rather than resample independent observations, such as residual errors. Moreover, the effect of resampling the residual errors makes the procedure much more dependent on the choice of model which has been fitted. Finally, literature has demonstrated that bootstrapped prediction intervals are a useful non-parametric alternative to the usual Box-Jenkins intervals.

Measures of forecast goodness

In practice, it is unlikely that we will ever stumble upon a fully optimal forecast; instead, the most common situation is combine a number of suboptimal forecast. Even for very good forecast, the actual and forecasted values may be very different.

This highlights the inherent limits of the forecastibility, which depends on the process being forecast; some process are inherently easy to forecast, whereas others too difficult. In other words, sometimes the information on which the forecaster conditions is very valuable, a sometimes it isn't.

The crucial object in measuring the forecast accuracy is the loss function $L(y_{t+h}, \hat{y}_{t+h,t})$ often restricted to residuals $L(e_{t+h,t})$. In addition to the shape of the loss function, the notation $t +$

h, t has a deep meaning, it represents that the forecast horizon h at a certain time t is a conditional guess based on the current time t.

Accuracy measures

Rankings of forecast accuracy may be very different across different loss functions and different horizons. On this sections we will discuss few accuracy measures that are important and popular. The following notation will be introduced in order to reduce computational resources at the calculus of the error. Accuracy measures are usually defined on the **forecast errors**

Eq. 96 $$e_{t+h,t} = y_{t+h} - \hat{y}_{t+h,t}$$

Or percent errors

Eq. 97 $$p_{t+h,t} = {}^{y_{t+h} - \hat{y}_{t+h,t}}/_{y_{t+h}}$$

The **mean error** measures the **bias**, small bias is desired on any model.

Eq. 98 $$ME = \frac{1}{T}\sum_{t=1}^{T} e_{t+h,t}$$

The **error variance** measures the dispersion of the forecast errors, small variance are also preferred.

Eq. 99 $$EV = \frac{1}{T}\sum_{t=1}^{T}(e_{t+h,t} - ME)^2$$

Although these measures are components of the accuracy, neither provides an overall accuracy measure. The most common overall accuracy measure are the **mean square error**

Eq. 100 $$MSE = \frac{1}{T}\sum_{t=1}^{T}(e_{t+h,t})^2$$

And the mean squared percent error,

Eq. 101 $$MSPE = \frac{1}{T}\sum_{t=1}^{T}\left(p_{t+h,t}\right)^2$$

Often the square roots of these measures are used to preserve the units, yielding the **root mean squared error**,

Eq. 102 $$RMSE = \sqrt{\frac{1}{T}\sum_{t=1}^{T}\left(e_{t+h,t}\right)^2}$$

And the root mean squared percent error,

Eq. 103 $$RMSPE = \sqrt{\frac{1}{T}\sum_{t=1}^{T}\left(p_{t+h,t}\right)^2}$$

To understand the meaning of “preserving units”, and why it is sometimes helpful to do so, suppose that the forecast errors are in dollars. Then the MSE is measured on squared dollars, using a square root the units come s back to dollars. The opposite of preserving units is delete them, this condition is exploited when the errors of the **multi resolution branches** of an ensemble must be mixed.

Due to, each branches has been subdue to some transformation, the statistical properties of the original set are no longer there, although there units maybe is preserved. Somewhat less popular accuracy measures are the **mean absolute error**,

Eq. 104 $$\boldsymbol{MAE = \frac{1}{T}\sum_{t=1}^{T}\left|e_{t+h,t}\right|}$$

And the mean absolute percent error,

Eq. 105 $$\boldsymbol{MAPE = \frac{1}{T}\sum_{t=1}^{T}\left|p_{t+h,t}\right|}$$

Residual statistics

Residuals are the forecast errors, in ideal conditions after perform the correct decomposition of the signal components and the model of them, residual must express no autocorrelation, random distribution at each time t, and independence of any factor.

Eq. 106 $$\boldsymbol{r_t = e_t \quad \therefore \quad e_t \;\; iid\sim\,(0, \sigma_t^2)}$$

On previous sections we study the decomposition of the residual errors on the contributions made by the modelling procedures and the intrinsic noise of the data. On this section we will continue referring to the residuals as a unity. The statistics measure the degree of goodness of the model for a given data set, they will help us to improve the correct guess of the model features.

The **sum square of the residuals** serve as constrains to minimize on a least square estimation. CGP does use of this measure.

Eq. 107 $$\boldsymbol{SSR = \sum_{t=1}^{T} e_t{}^2}$$

The **F-statistic** could be used indistinctively for two purposes, measure the predictive value of the independent variables introduced on the model as a whole, or measure if parameters improvements on a model leads a better ability to fit a target. On this thesis the second option was employed to measure the optimal base learners per cluster on the ensemble model.

Eq. 108 $$\boldsymbol{F = \frac{(SSR_1 - SSR_2)/(p_2 - p_1)}{(SSR_2)/(T - p_2)}}$$

The terms SSR_1, SSR_2 are the sum of squared residuals from a restricted regression $\widehat{f_1}(x)$ with p_1 parameters, and a regression $\widehat{f_2}(x)$ with p_2 parameters. It is called restricted because parameters at the regression 1 are fewer than model 2, being $p_1 < p_2$ a norm.

Under the null hypothesis that model 2 does not provide a significantly better fit than model 1, F will have an F distribution, with (p2−p1, T−p2) degrees of freedom. The null hypothesis is rejected if the F calculated from the data is greater than the critical value of the F-distribution for some desired false-rejection probability (e.g. 0.05).

Fraction of variance Unexplained gives an unbiased estimation of how much variance in the response variable can be explained by the model. A value equal to 0 corresponds to a perfect fit.

Eq. 109 $$FVU = \frac{\sum_{t=1}^{T} e_t{}^2}{\sum_{t=1}^{T}(y_t - \bar{y})^2} = \frac{MSE}{\sigma^2}$$

The **R squared error** is the percent of the target variance explained by the variables included on the regression. It measures the in-sample success of the regression model in forecasting the target. It is widely used to quickly check the *goodness on fit*, or forecastability of y based on the input variables.

Eq. 110 $$R^2 = 1 - \frac{\sum_{t=1}^{T} e_t{}^2}{\sum_{t=1}^{T}(y_t - \bar{y})^2} = 1 - \frac{MSE}{\sigma^2} = 1 - FVU$$

Durbin-Watson statistic allows to examine the residuals in search of patterns, it performs a first order serial correlation over the residuals. If the errors are serially correlated, the model could improve his goodness by integrating the correlated lags. DW takes values in the interval [0,4], being values around 2 normal, and less than 1.5 a motive to worry.

Eq. 111 $$DW = \frac{\sum_{t=2}^{T}(e_t - e_{t-1})^2}{\sum_{t=1}^{T} e_t{}^2} = \frac{\sum_{t=2}^{T}(e_t - e_{t-1})^2}{\mathrm{SSR}}$$

Estimators of the model sensibility regards to parameters

Forecast accuracy can depend on the forecast horizon. Some forecasts are more stable than others. How far into the future this horizon extends and where it ends must be known. In general, the farther into the future the forecast horizon, the more difficult it is to forecast. The following estimators are joint functions of the minimum forecast error and some form of penalty for the number of free parameters (degrees of freedom) in the model.

As you already notice F-test stablish an analogy between the concept of degrees of freedom and parameters to tune on an algorithm. The literature make use of this analogy to estimate the goodness of the model; for example on [83], on the section "*fitting neural network models*", the neurons weights and the number of inputs on a NN model were counted as parameters to optimize based on the BIC and AIC.

On this thesis, analogously to the use of the F-test criteria, information criteria's such as S^2, AIC, BIC have been used to fit the ensembles, assuming as parameters to tune the number of base learners. These test can be extended to the number of clusters, branches or any other parameter that reader could infer as crucial for the model complexity.

S^2, **s-squared**, is the sample variance of the squared residuals. It estimate the dispersion of the regression disturbance, being a natural estimator of the $\boldsymbol{\sigma}^2$. The larger **S**^{2}**,** the worse the models fit.

Eq. 112 $$\boldsymbol{S^2} = \frac{\sum_{t=1}^{T} e_t{}^2}{\mathbf{T-p}} = \frac{\mathbf{SSR}}{\mathbf{T-p}}$$

Of course it exist the standard deviation of the residuals, which is easy to interpret because have the same units of the target. S must be no more than the 12-15 % of the target mean.

Eq. 113 $$\boldsymbol{S} = \sqrt{\frac{\sum_{t=1}^{T} e_t{}^2}{\mathbf{T-p}}} = \sqrt{\frac{\mathbf{SSR}}{\mathbf{T-p}}}$$

Akaike information criterion, or **AIC**, estimate the out-of-sample forecast error variance, as is **S**^{2}. But it penalizes harder the degrees of freedom. It is used to select competing models.

Eq. 114 $$\boldsymbol{AIC} = \frac{\boldsymbol{SSR}}{\mathbf{T}}\,\boldsymbol{exp}\left(\frac{\mathbf{2}(\boldsymbol{p}+\mathbf{1})}{\boldsymbol{T}}\right)$$

If we consider a perfect fit on the model, making AIC equals to zero, we will found the generalized form of the AIC

Eq. 115 $$\boldsymbol{AIC} = \boldsymbol{T}\,\boldsymbol{ln}\left(\frac{\boldsymbol{SSR}}{\mathbf{T}}\right) + \mathbf{2}\boldsymbol{p}$$

The AIC have a second extension, it the bias corrected AIC version. Fortunately, it is easy to calculate by adding a correction term to the AIC. This term is small when p is small compared with N, but can become large if p/N exceeds about 0.05.

Eq. 116 $$\boldsymbol{AIC_c} = \boldsymbol{T}\,\boldsymbol{ln}\left(\frac{\boldsymbol{SSR}}{\mathbf{T}}\right) + \mathbf{2}\boldsymbol{p} + \frac{\mathbf{2}(\boldsymbol{p}+\mathbf{1})(\boldsymbol{p}+\mathbf{2})}{\mathbf{T-p-2}}$$

The **Bayesian information criterion** (**BIC**), or the most known **Schwartz-Bayesian information criterion** (**SBC**), is another criteria for the selection of models among a finite set. It has a harsher degrees-of-freedom penalty.

Eq. 117 $$\boldsymbol{SBC} = \frac{\boldsymbol{SSR}}{\mathbf{T}}\boldsymbol{T}^{\frac{\boldsymbol{p}+\mathbf{1}}{\boldsymbol{T}}}$$

The generalized form of the SBC is

Eq. 118 $$\boldsymbol{SBC} = \boldsymbol{T}\,\boldsymbol{ln}\left(\frac{\boldsymbol{SSR}}{\mathbf{T}}\right) + \boldsymbol{p} + \boldsymbol{p}\,\boldsymbol{ln}(\mathbf{T})$$

Forecasts can also be evaluated in terms of their complexity or parsimony. The lesser the parameter redundancy and parameter uncertainty, the better the model used for forecasting. Simpler forecasts are preferred to complex forecasts, given the same level of accuracy [82].

Relative forecast ability might be assessed in terms of the comparative abilities of different approaches. One criterion of relative forecast ability is that of forecast efficiency. Forecast efficiency of a model involves comparing the mean square forecast error of the model to some baseline model.

The forecast error variance of the model under consideration may be derived from a baseline comparison. That baseline used is often the naïve forecast, a forecast formed by assuming that there is no change in the value of the latest observation. Theil developed a **U statistics** which compares forecasts.

Eq. 119

$$\boldsymbol{U_1} = \frac{\sqrt{\frac{1}{T}\sum_{t=1}^{T}(y_t - \bar{y}_t)^2}}{\sqrt{\frac{1}{T}\sum_{t=1}^{T}(y_t)^2} + \sqrt{\frac{1}{T}\sum_{t=1}^{T}(\bar{y}_t)^2}}$$

Eq. 120

$$\boldsymbol{U_2} = \sqrt{\frac{\sum_{t=1}^{T-1}\left(\frac{y_{t+1} - \bar{y}_{t+1}}{y_t}\right)^2}{\sum_{t=1}^{T-1}\left(\frac{y_{t+1} - y_t}{y_t}\right)^2}}$$

The more accurate the forecasts, the lower the value of the U1 statistic. The U1 statistic is bounded between 0 and 1, with values closer to 0 indicating greater forecasting accuracy. The U2 statistic will take the value 1 under the naive forecasting method: (y(t+1)-y(t))/ y(t). Values less than 1 indicate greater forecasting accuracy than the naive forecasting method, values greater than 1 indicate the opposite.

Other relevant concepts

On this section we will introduce the mathematical description of the statistical methods employed on the statistical characterization of the temporal series at the preprocessing and tuning modeling stages. Although, the description of the methods follows the mathematical cannon; the mathematical formulation and the concepts are personalized to highlight the application of them on the thesis subject.

Statistical moments

Far from being a simple introduction to some statistical descriptors, we will present the tools necessary to observe the uncertainty attached to the time series. At the end of this section we you will be familiarized with methods to describe the influence of the input variables on the statistical moments of the target.

Mean

Eq. 121

$$\boldsymbol{\bar{y}} = \frac{1}{\mathrm{T}}\sum_{t=1}^{T} \boldsymbol{y_t}$$

Standard deviation

Eq. 122 $$\boldsymbol{\sigma} = \sqrt{\frac{1}{\mathrm{T}}\sum_{t=1}^{T}(\boldsymbol{y}_t - \overline{\boldsymbol{y}})^2}$$

Skewness

Measures the amount of asymmetry in a distribution, a positive value means a long right tail and a negative value a long left tail. A zero value means a totally centered distribution.

Eq. 123 $$\boldsymbol{Skw} = \frac{\frac{1}{\mathrm{T}}\sum_{t=1}^{T}(\boldsymbol{y}_t - \overline{\boldsymbol{y}})^3}{\boldsymbol{\sigma}^3}$$

Kurtosis

Measures the thickness of the tails in respect to a normal distribution, a normal random variable has a kurtosis equals 3 (mesokurtic), large values create high tails (leptokurtic), lower values creates plain tails (platikurtic).

Eq. 124 $$\boldsymbol{Kur} = \frac{\frac{1}{\mathrm{T}}\sum_{t=1}^{T}(\boldsymbol{y}_t - \overline{\boldsymbol{y}})^4}{\boldsymbol{\sigma}^4}$$

These indicators are called statistical moments because serve to quantify the mass center of the data (mass), and the inertia of the series based on the distribution. This is especially important for the conditional forecast.

Jarque-Bera test statistic

It effectively aggregates the information about the skewness and the kurtosis to produce an over-all test for normality.

Eq. 125 $$JB = \frac{T}{6}\left(Skw^2 + \frac{1}{4}(Kur - 3)^2\right)$$

Under the null-hipothesis of independent normally distributed observations, JB test is distributed as a chi-square random variable with 2 degrees of freedom in large samples. other normality test [27], as Kolmogorov-Smirnov test could be also used.

Other statistical descriptors of the target could be consulted in order to address information of the distribution of it. The **sample standard deviation** of the dependent variable measures the dispersion of it.

Eq. 126 $$\boldsymbol{SSD} = \sqrt{\frac{1}{T-1}\sum_{t=1}^{T}(\boldsymbol{y}_t - \overline{\boldsymbol{y}})^2}$$

Signal processing operations

On this section we present the equations that define the signal processing operations carried out on this thesis. The equations have been ordered to offer to the reader a hint to compare the different operators and find their similarities and differences.

Cross correlation

Eq. 127 $$W_{UV}(k) = {}^{1}/_{N} \sum_{j} U(k+j) * V(k)$$

Autocorrelation function (ACF)

Eq. 128 $$W(k) = {}^{1}/_{N} \sum_{j} U(k+j) * U(k)$$

Autocovariance function (mean centered version of the autocorrelation)

Eq. 129 $$W(k) = {}^{1}/_{N} \sum_{j} (U(k+j) - \bar{U}) * (U(k) - \bar{U})$$

Covariance coefficient

Eq. 130 $$W = {}^{1}/_{N} \sum_{k} (U(k) - \bar{U}) * (V(k) - \bar{V}) = Cov(U,V)$$

Convolution

Eq. 131 $$W = \sum_{J} U(J) * V(k - J + 1)$$

Correlation coefficient

Scaled version of the covariance, also a normalized coefficient of the cross correlation.

Eq. 132 $$W = {}^{Cov(U,V)}/_{\sigma_U \sigma_V} = {}^{1}/_{N} \sum_{k} \left[\frac{(U(k) - \bar{U})}{\sigma_U} * \frac{(V(k) - \bar{V})}{\sigma_V} \right]$$

Sample autocorrelation function (SACF)

Eq. 133 $$r_y(k) = \frac{\sum_{t=1}^{n-k} (y(t) - \bar{y})(y(t+k) - \bar{y})}{\sigma_Y^2} \quad at \quad k \; Lag$$

Partial autocorrelation function (PACF)

Eq. 134 $$r_{k,k} = \begin{cases} r_1 & if \; k = 1 \\ \frac{r_k - \sum_{j=1}^{k-1} r_{k-1,j} \cdot r_{k-j}}{1 - \sum_{j=1}^{k-1} r_{k-1,j} \cdot r_k} & otherwise \end{cases} \; ; \; r_{kj} = r_{k-1,j} - r_{kk} r_{k-1,k-j}$$

Appendix C: Experimental databases

The modelling methodology presented on this thesis has been validated over some electrical profiles gathered from public databases, and particular research projects. On this section we will present the data bases information, together with a statistical analysis of them. The overall characteristics of the data base will be introduced on this section, followed for the description of the load profiles on other sections.

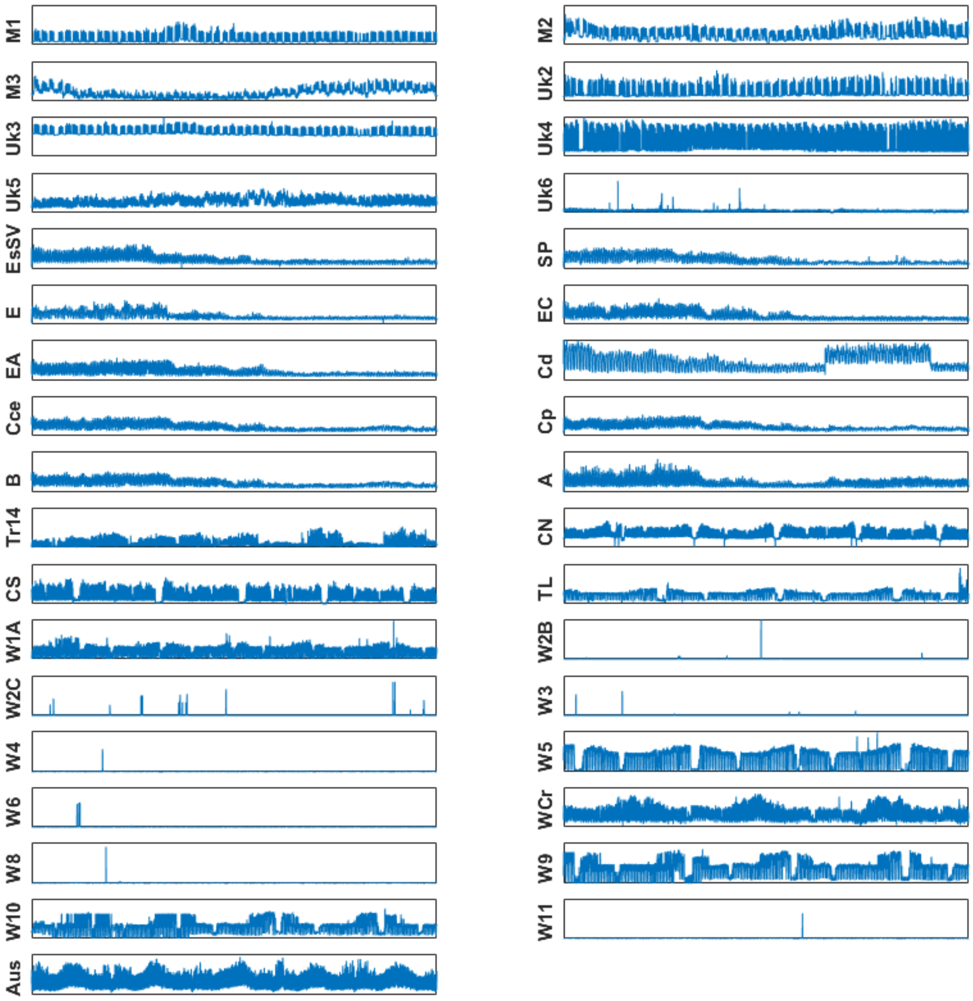


Figure 47. Raw Load profile graphs stored on the Db.

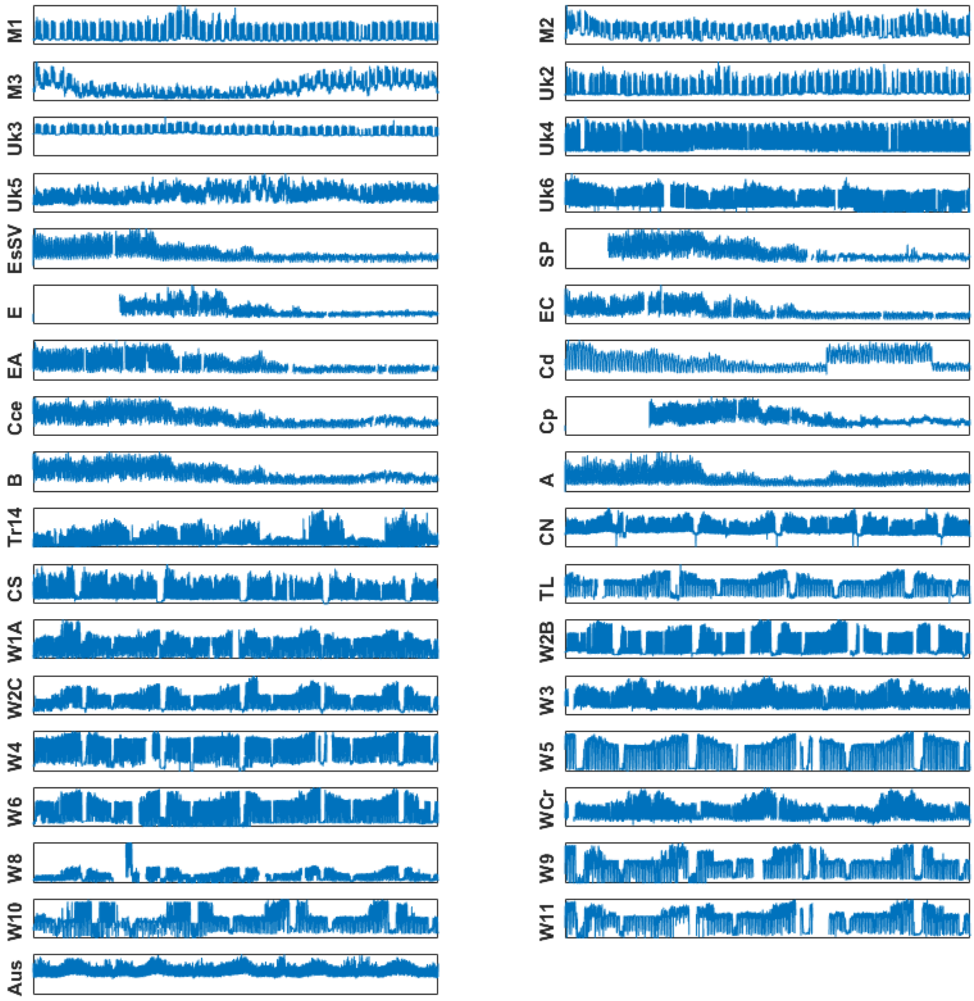


Figure 48. Cleaned and normalized load profile graphs stored on the Db.

For all the **normalized data bases** the following feature parameters has been calculated:

- **Lag**: Seasonal periods gather by an automatic analysis of the sample PACF peaks and Periodogram peaks
- **Ss**: Strength of the seasonality = 1- var(Rt)/var(Yt-St) (1 month period)
- **Sc**: Strength of the cyclical components (1 week, 1 day period)
- **Ee**: Espectral entropy
- **Obc**: Optimal Box-Cox transformation parameter
- **Ncp**: Number of crossing points of mean line
- **Acf1**: first order autocorrelation
- **Lu**: Lumpiness, variance of intraday variances
- **Spk**: Spikiness,variance of intraday residual variances.

- **Vch**: Variance change, max difference in variances of consecutive moving windows of day size.

Based on these parameters, a study of the data base features has been performed in order to select the most interesting scenarios to model. On the follow table the calculated parameters for all the data bases are presented. Due to the Lumpiness and Spikiness converge to similar results the last one was erased.

Table 24. Db feature parameters.

Db Name*	**Ss**	**Sc1**	**Sc2**	**Ee**	**Obc**	**Npc**	**Acf1**	**Lu (1e-4)**	**Vch (1e-3)**
Bld_Uk1M1	0,98	0,98	0,95	0,23	-0,60	558	0,98	3,32	85
Bld_Uk1M2	0,98	0,98	0,96	0,10	0,17	574	0,98	1,94	113
Bld_Uk1M3	0,96	0,98	0,98	0,19	0,36	1520	0,98	0,45	78
Bld_Uk2	0,96	0,95	0,87	0,11	-0,85	636	0,96	2,77	99
Bld_Uk3	0,99	0,99	0,98	0,11	-3,85	500	0,99	0,18	19
Bld_Uk4	0,97	0,97	0,80	0,22	0,05	678	0,96	5,24	86
Bld_Uk5	0,98	0,98	0,95	0,14	0,21	858	0,98	0,49	49
Bld_Uk6	0,98	0,98	0,95	0,24	0,39	2510	0,98	4,01	486
Bld_EsSV	0,96	0,98	0,95	0,18	-0,72	558	0,94	4,13	157
Bld_EsSP	0,92	0,96	0,93	0,10	-0,52	634	0,90	2,78	146
Bld_EsE	0,92	0,95	0,92	0,18	-0,84	850	0,90	1,95	198
Bld_EsEC	0,92	0,96	0,94	0,19	-0,25	548	0,91	1,70	182
Bld_EsEA	0,94	0,96	0,91	0,19	-0,64	550	0,90	4,77	1162
Bld_EsCd	0,97	0,97	0,95	0,13	0,07	324	0,94	2,79	133
Bld_EsCce	0,92	0,96	0,93	0,16	-0,14	630	0,91	1,79	59
Bld_EsCp	0,94	0,97	0,94	0,15	-0,62	618	0,93	2,06	2180
Bld_EsB	0,92	0,96	0,93	0,16	-0,14	636	0,91	1,77	59
Bld_EsA	0,95	0,96	0,85	0,19	-0,14	548	0,89	3,71	81
Bld_EsUpcTr14	0,92	0,90	0,83	0,23	0,16	5080	0,94	1,39	95
Ind_EsUpcCN	1,00	0,99	0,99	0,11	0,16	2380	0,97	1,56	119
Ind_EsUpcCS	0,99	0,99	0,99	0,25	0,31	2280	0,97	3,83	237
Ind_CMPTL	1,00	0,98	1,00	0,09	0,97	1676	1,00	0,36	22
Ind_CMPW1A	0,91	0,78	0,92	0,15	0,86	20912	0,92	0,39	176
Ind_CMPW2B	0,99	0,99	0,99	0,21	-0,20	2742	0,99	6,67	70
Ind_CMPW2C	0,99	0,97	0,99	0,13	0,69	3122	0,99	0,06	72
Ind_CMPW3	0,99	0,99	0,98	0,12	-0,41	2188	0,99	3,22	38
Ind_CMPW4	0,99	0,97	0,99	0,14	1,65	2066	0,99	2,91	466
Ind_CMPW5	0,99	0,98	1,00	0,13	0,80	968	1,00	3,75	46
Ind_CMPW6	0,99	0,96	0,99	0,17	0,73	2656	0,99	3,20	56
Ind_CMPWCr	0,95	0,93	0,91	0,11	-0,15	8056	0,94	1,81	336
Ind_CMPW8	0,99	0,99	0,99	0,19	0,39	3664	1,00	0,14	73
Ind_CMPW9	0,99	0,97	0,99	0,12	0,74	1798	0,99	2,23	79
Ind_CMPW10	0,98	0,96	0,98	0,10	0,55	2776	0,99	1,93	165

Ind_CMPW11	0,98	0,96	0,99	0,11	1,10	2606	0,99	1,88	999
State_Aus	0,99	0,99	0,98	0,08	0,71	4594	0,98	0,13	13

* Acronyms: Bld: building load profile, Ind: industrial load profile, State: Regional load profile

By perform a principal component analysis over the Db feature parameters gathered is possible classify the load profiles according to the random walk degree presented, and the seasonality strength.

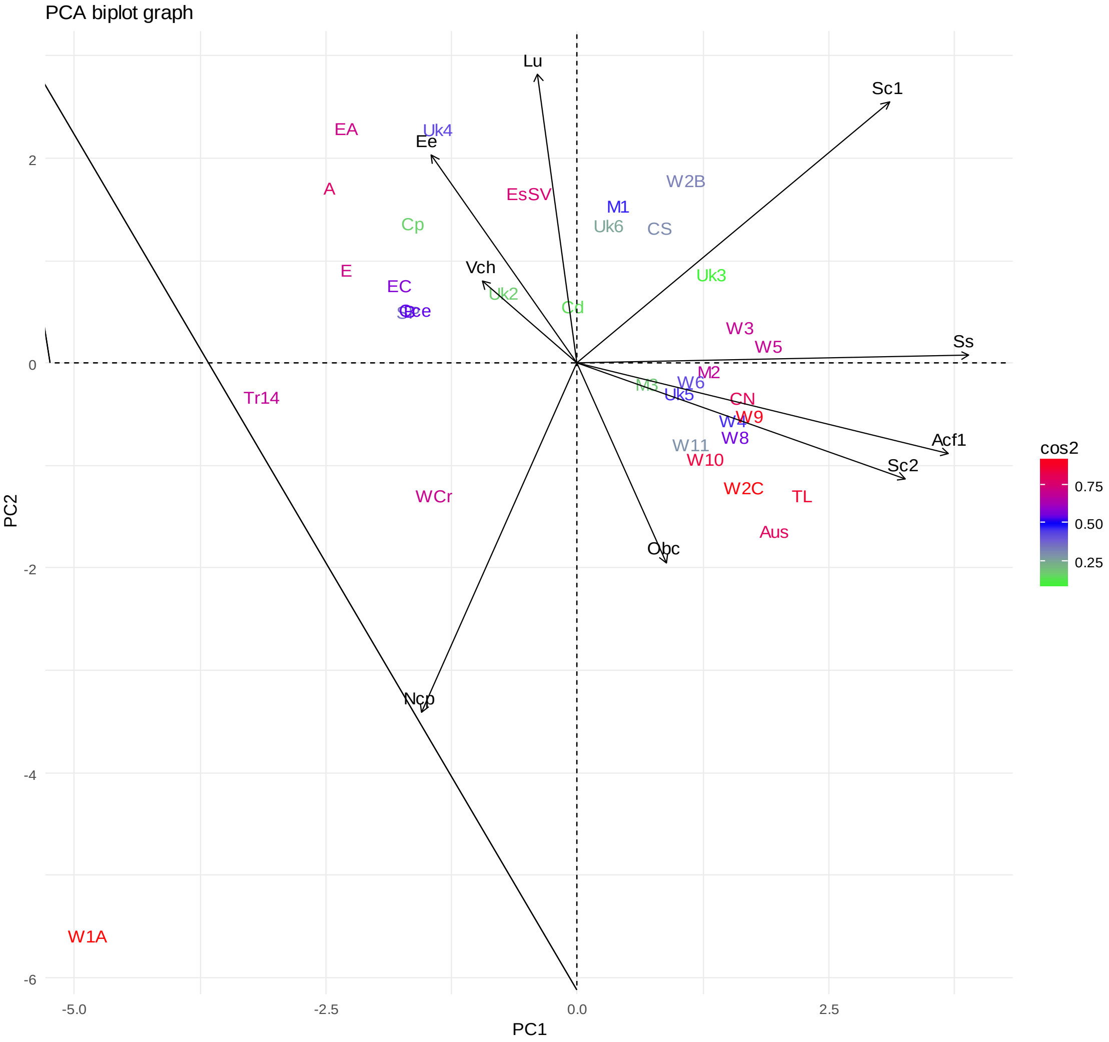


Figure 49. PCA graph of the database features.

On the PCA graph is intuitive say that first principal component represent the strength of the cyclical and seasonal components, and second represent the random walk derived from innovations. We can see some examples of that, Australia load profile is for example highly seasonal and smooth due to the level of load aggregations. On an opposite position, the Spanish building Elche Aljub, is presented as lowly seasonal driven, and highly charged with a random walk component.

Finally, we can resume the load profiles according to their quadrants. The second quadrant present load profiles with a large aggregation level such as industrial and regional profiles, showing a soft curve and well defined seasonal and cyclical patters as we move away from the origin on a diagonal line.

The fourth quadrant present load profiles with a lower aggregation level such as building load profiles, showing a spiky curve and not so well defined seasonal and cyclical patters as we move away from the origin on a diagonal line. First quadrant present a transition among the characteristics of the second and fourth quadrants, load profiles are characterized by a medium level aggregation.

Australian electricity case

The NEM is the Australian wholesale electricity market and the associated interconnected electricity transmission grid [137]. It provides electricity service for six states and territories in Australia, known as Queensland, New South Wales (NSW), Australian Capital Territory (ACT), Victoria, South Australia and Tasmania. In the practical operations of NEM, ACT is joined with NSW, leading to five regional networks. Among the five regions, NSW takes the largest share of the total electricity consumption.

The load forecasting activities in NEM consist of medium-term load forecasting (MTLF), short-term load forecasting (STLF), pre-dispatch load forecasting and dispatch load forecasting for each regional network [138]. NEM defines that MTLF begins from the eighth day up to 24 months ahead with a daily resolution. STLF and pre-dispatch load forecasting is run on the basis of calendar days from the half-hour start 00:30 am to half-hour ending 24:00 am (**48 points in per day**) ranging from 1 to 8 calendar days ahead and dispatch load forecasting aims at the load for the next 5 min during the real-time operation of the system.

The historical electricity consumption data of ACT-NSW regions from 1st January 2006 to 31st December 2011 (5 year, 87648 points in total) is used. The data set includes the electricity consumption of the regions and its weather variables such as dry bulb temperature, dew point. Variables related with the seasonality such as hour, day of week, and holidays; and related with historical electricity consumption such as previous day, and previous week consumption can be easily created.

Graphs of some features measured for this load profile are presented on following. According to a frequency and temporal analysis the most prominent lags are identified as: 2,4,12 hours, and weekdays.

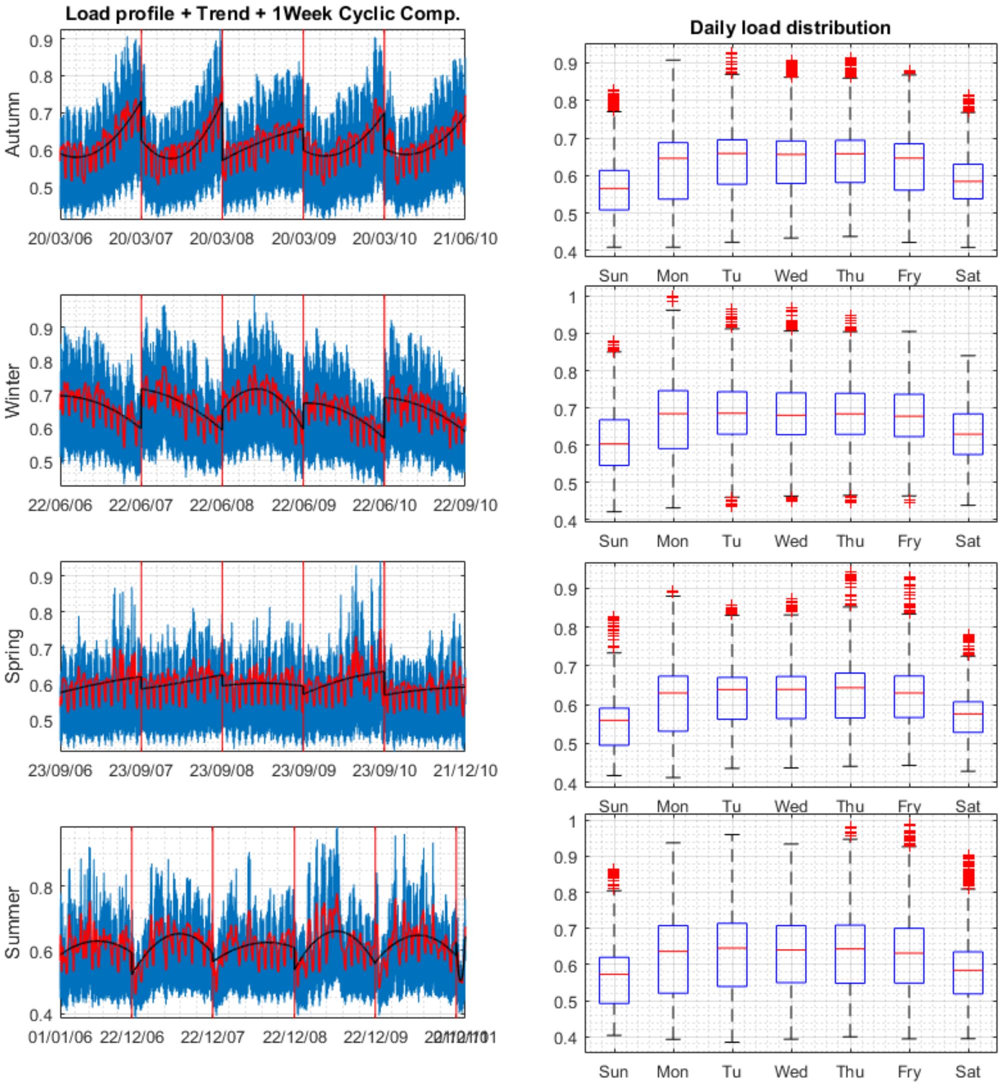


Figure 50. Graphs of the Australian load profile. Right column: Segmentation of Load profile per seasons (blue), First cyclical component of the seasons (red), trend component of the seasons (black). Left column: Daily load distribution per season.

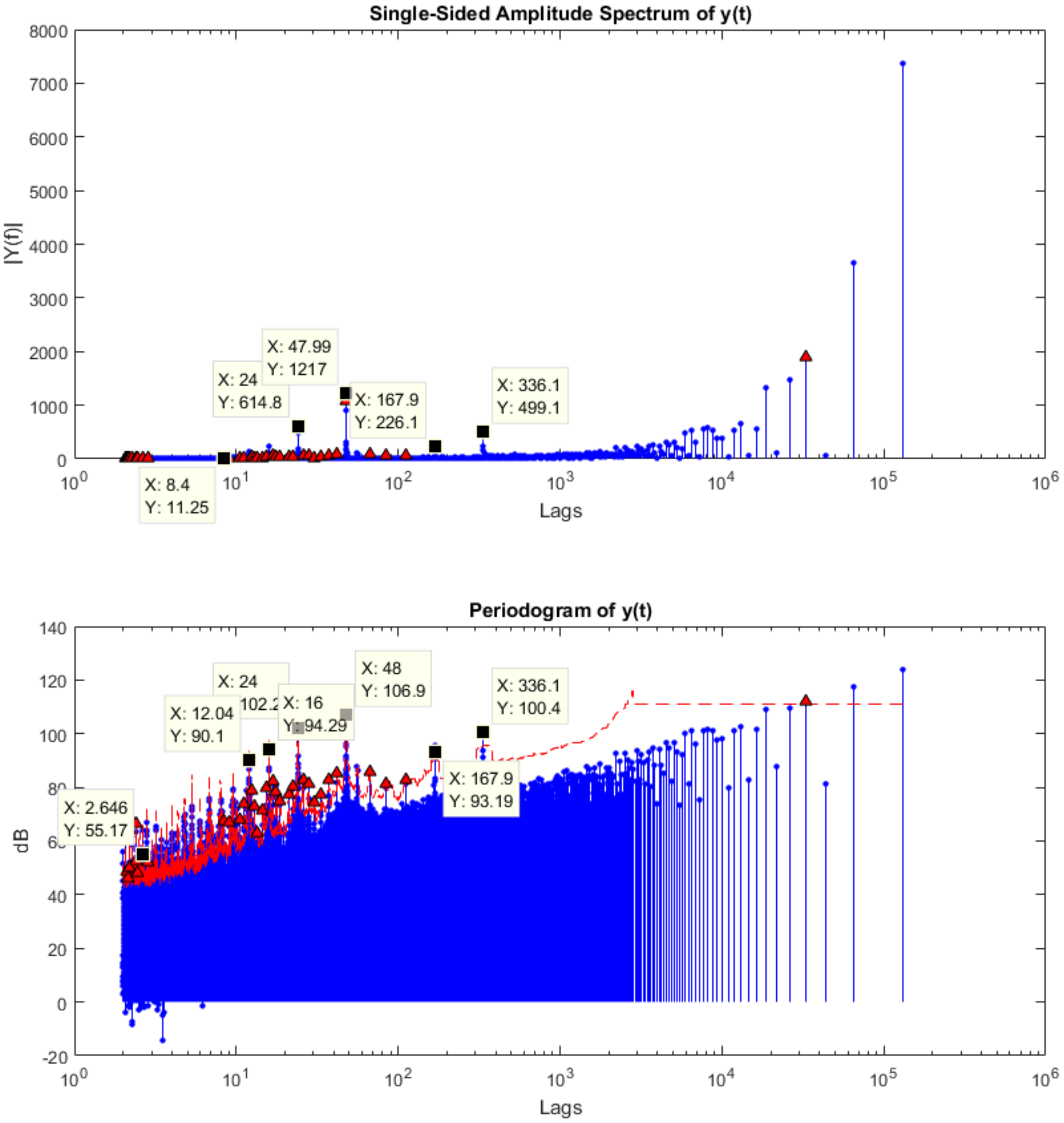


Figure 51. Periodogram of the Australian load profile.

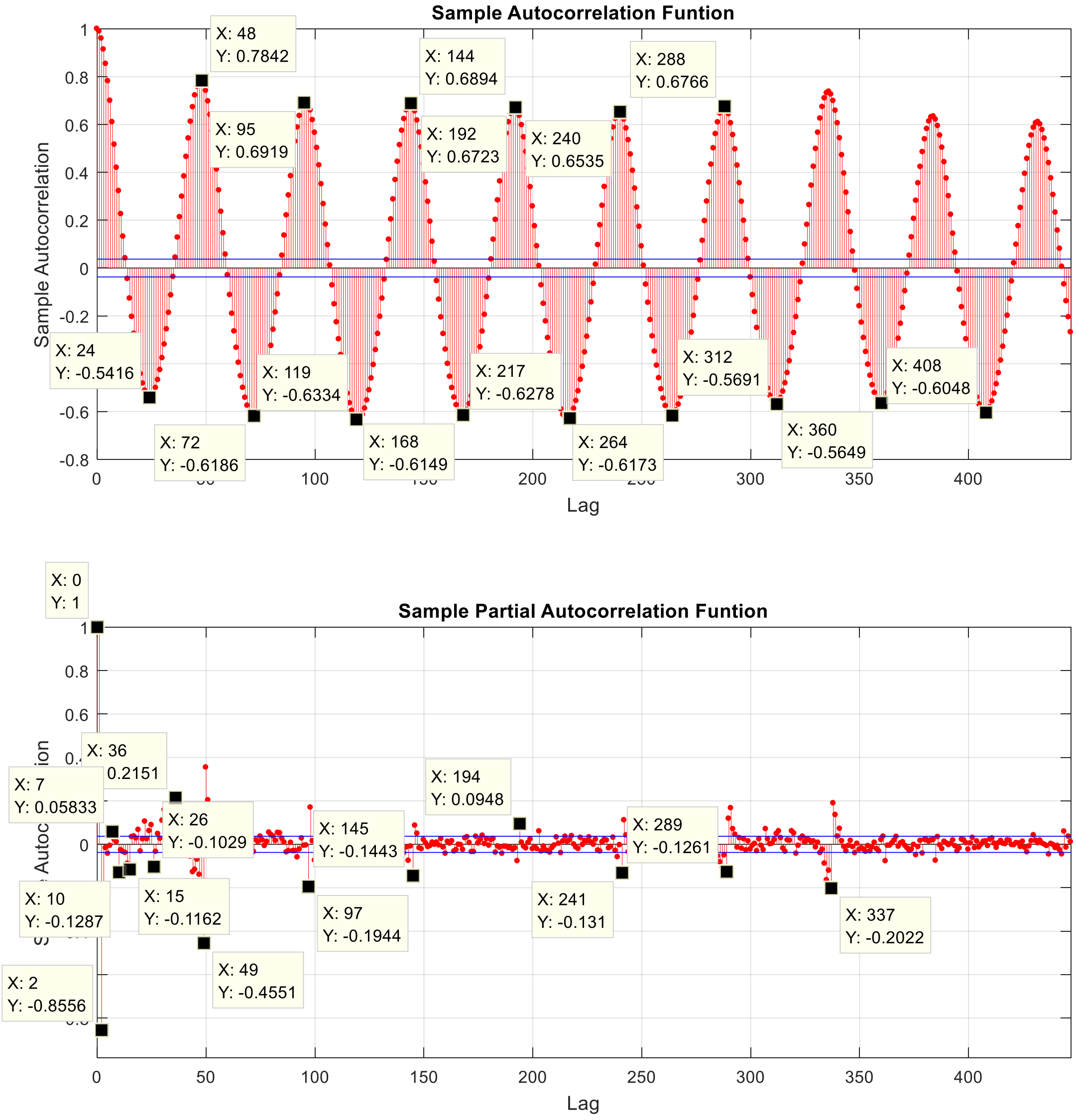


Figure 52. Sample ACF and PACF of the Australian load profile.

Industrial case: Car manufacturing company

Load profiles of the industrial user to be forecasted belongs to the car manufacturing company SEAT. Located in Martorell (Spain), it counts with numerous workshops specialized on different task along the car productive chain: Press (1A), Bodywork (1, 2, 6), Painting (4, 5, 2B), Assembly (8, 9, 10, 11), Logistics (14, 15), etc.

The data base collected counts with 12 workshops (1A, 2B, 2C, 3, 4, 5, 6, 8, 9, 10, 11, Cross section) and the **total factory consumption** which was selected as **target**. The following figure shows the distribution of SEAT Martorell plant. Each building is named with the letter T, which stands for Taller, ‘workshop’ in Spanish, and it has a number assigned.

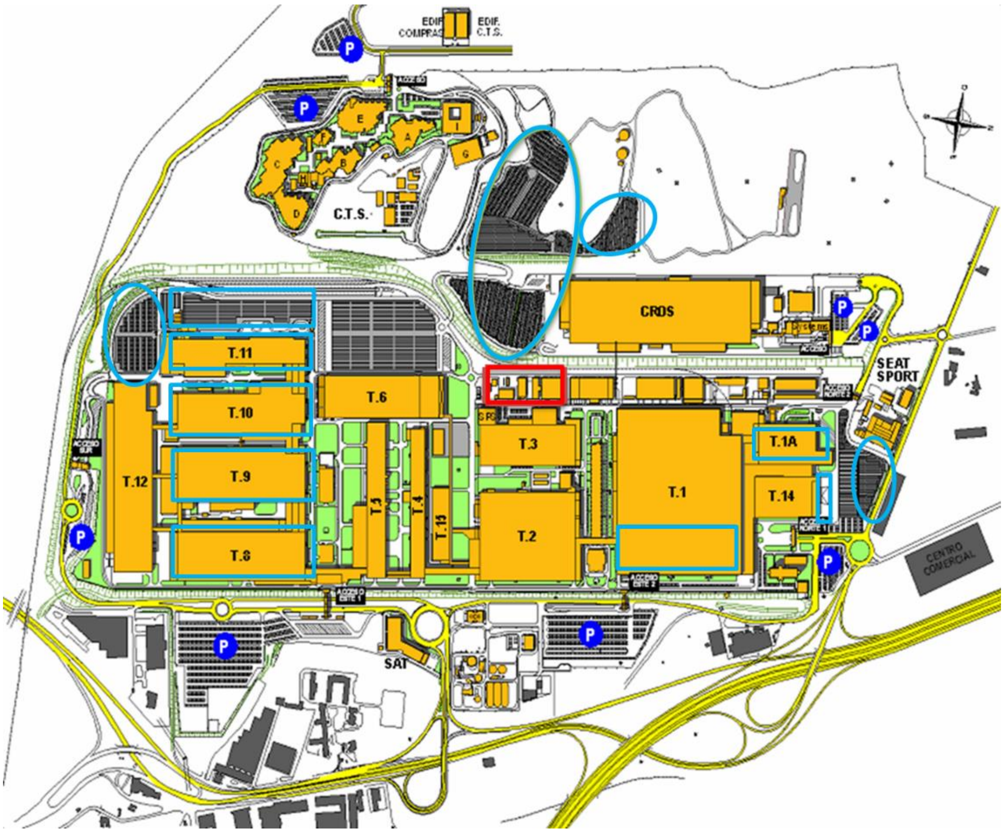


Figure 53. SEAT factory map. Source: SEAT.

The historical electricity consumption data goes from 1st January 2012 to 29st January 2017 (178173 points in total). The data set includes the electricity consumption of the workshops and its weather variables such as temperature. Variables related with the seasonality such as hour, day of week, and holidays; and related with historical electricity consumption such as previous day, and previous week consumption can be easily created.

STLF is run on the basis of calendar days from the quarter-hour start 00:15 am to 24:00 am (**96 points in per day**) According to a frequency and temporal analysis the most prominent lags are identified as: 8 hours, and 1, 7 days.

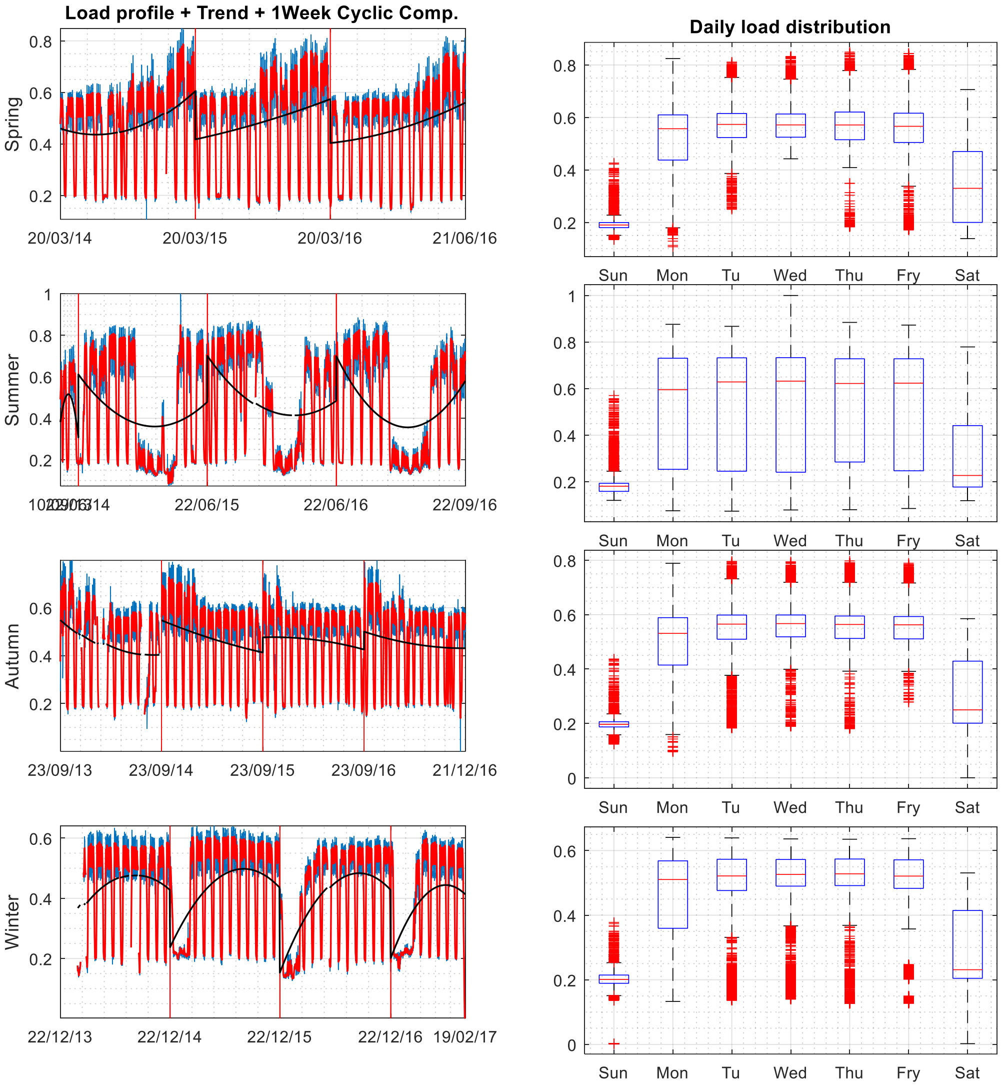


Figure 54. Graphs of the total electric consumption of the SEAT plant. Right column: Segmentation of load profile per seasons (blue), First cyclical component of the seasons (red), trend component of the seasons (black). Left column: Daily load distribution per season.

Spanish university case

A campus as well as their individual buildings could be considered as commercial users due to they are on the service industry, and their load profile is human driven. We have analyzed the consumption of the south and north campus of the UPC university located in Barcelona (Spain).

The data could be obtained at the web page of the energy management services of the UPC: http://sirenaupc.dexcell.com/dashboard/widgets.htm. The historical electricity consumption data goes from 29-Jan-2010 13:00:00 to 25-Dec-2014 23:00:00 (42995 points in total).

The selected **target** was the **south campus**, the calendar days from the one-hour start 01:00 am to hour ending 24:00 am (**24 points in per day**) According to a frequency and temporal analysis the most prominent lags are identified as: 2, 12 hours, and 1, 7 days.

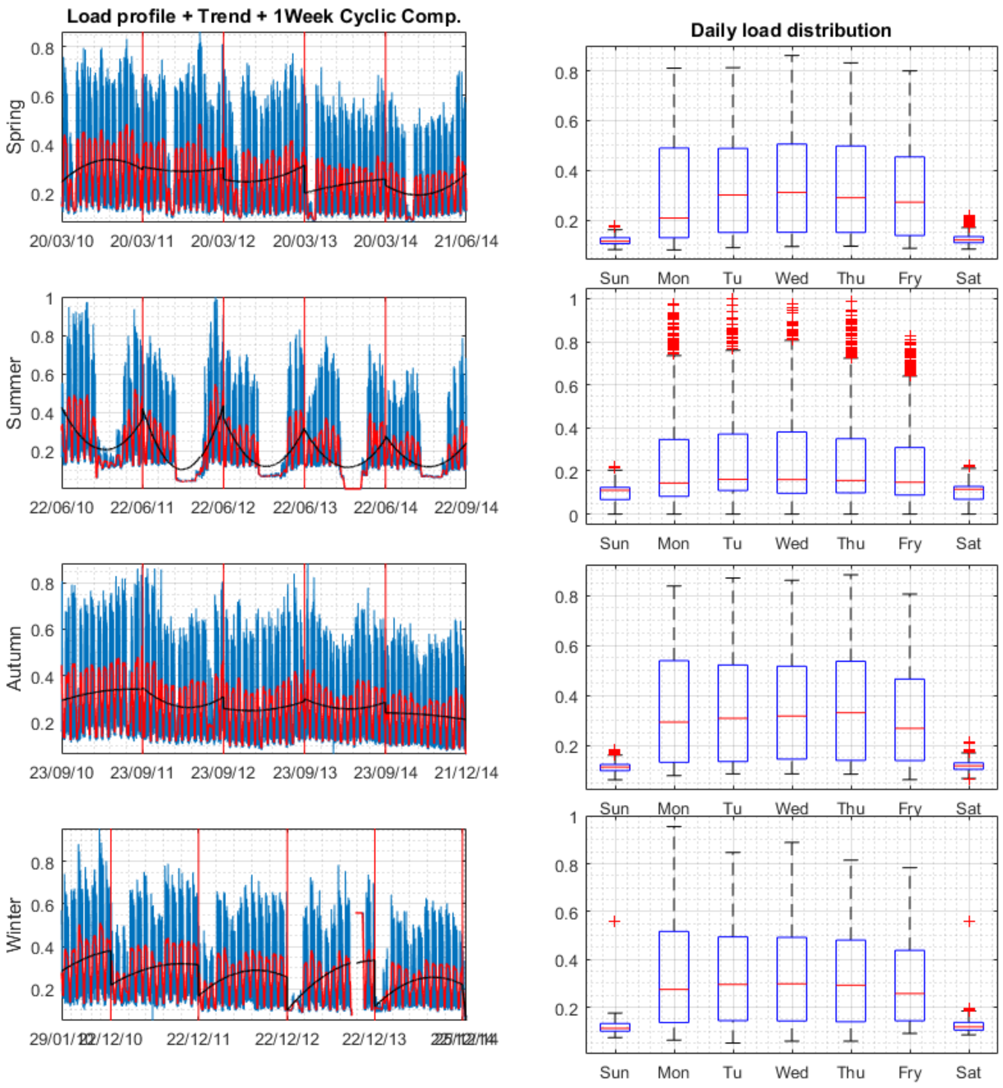


Figure 55. Graphs of the total electric consumption of the south campus. Right column: Segmentation of load profile per seasons (blue), First cyclical component of the seasons (red), trend component of the seasons (black). Left column: Daily load distribution per season.

Residential building cases

The residential load profiles has been gathered from public and private repositories. Their origins also are diverse, covering several geographic location on two countries. The first repository refers to 8 electric profiles measured on Bristol, England (https://data.gov.uk/dataset/energy-consumption-for-selected-bristol-buildings-from-smart-meters-by-half-hour). The acronyms of these buildings are: Uk1M1, M2, M3, Uk2, Uk3, Uk4, Uk5, and Uk6.

Second repository correspond to 10 electric profiles measured on diverse cities of Catalonia-Spain, The names of these buildings are: **E**s**S**an**V**icente, SantaPola, Elche, ElcheCarrus, ElcheAljub, ConDomina, ConCEntaina, CamPello, Babel, and Alcoy.

All of those building profiles counts with a high random walk component on their aggregated condition, characteristic of the residential purpose buildings. In order to not be repetitive we have selected as target only one building of this category.

The target selected is located on Bristol, The historical electricity consumption data goes from 29-Jan-2010 01:00:00 to 26-Dec-2014 20:45:00 (172112 points in total). The calendar days from the one-hour start 01:00 am to hour ending 24:00 am (**96 points in per day**) According to a frequency and temporal analysis the most prominent lags are identified as: 2, 12 hours, and 1, 7 days.

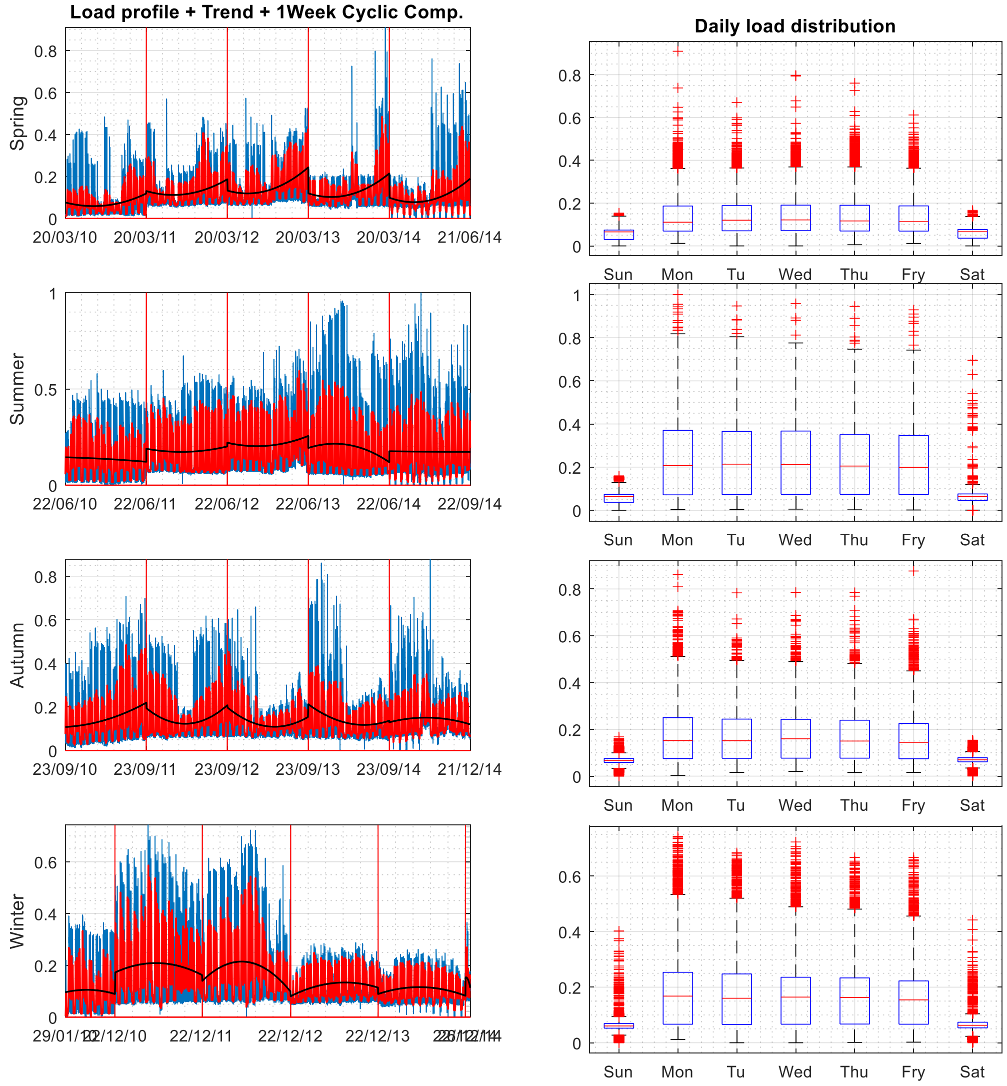


Figure 56. Graphs of the total electric consumption of the target residential building. Right column: Segmentation of load profile per seasons (blue), First cyclical component of the

seasons (red), trend component of the seasons (black). Left column: Daily load distribution per season.

Appendix D: A business oriented study of load forecasting for energy management

This Appendix introduces the **principal driver** on the development of this thesis: **the demand of energy efficiency and management services on the global energy market**. The market demand is also introduced under the point of view of the load forecasting activity, which is the supporter any energy management action. The contents follows a storyline starting from a general introduction to the world energy scenario, followed by a focus on the European market and Spanish scenario, and finally the advances on energy management technologies.

Context: World economic scenario

Nowadays, the technological process combined with the economical shifts caused by regulatory reforms and **geopolitical uncertainty has been key drivers on the transformation of the energy economic**. During the last ten years social and technological phenomena’s on the energy markets have challenged the theoretical basis of the energy prices forecasting, and the demand has returned to be the key driver of price evolution in short, medium and long term.

Among the remarkable list of events considered as game changers on the energy sector the following seems to be the most prevalent. The **decay of the interest about the nuclear energy** due the geopolitical interest in to reduce nuclear arsenal and concerns about the low return on investment of those energy generators. The **wavy nature on the price of the fossil fuels**, due to geopolitical uncertainty and organized actors as OPEC, has revealed an extreme dependence on those energies.

Also new generation of fuel such as shale gas and oil have produce a negative impact on the market and the public opinion due to their fast spread over several countries producing an oversupply and severe damage in ecosystems. Consequently, **Renewable energy and Smart grids have been taking an important role** due his low impact on the environment, easy scalability, fast development, and easy integration with existent infrastructure.

In order to prepare for a future where these key drivers will be increasingly influential, energy companies need **access to reliable data and long-term forecasts** for key economic and energy demand drivers.

In this scenario, the **computational intelligence** has been playing and important role. It has been providing the energy sector with projections and hypothesis for the world economy by zones, analysis of the future trends in global industry production with special attention on automotive and construction industries, exhaustive forecast on local consumer billing, energy prices projections, and drivers or triggers for economic and political risk.

To help to companies and government leaders to surf on the risk, opportunities and challenges on the energy economics markets, the **industry and financial experts lean on the most accurate models and analytical tools** to provide a valuable decision over the most critical energy issues, including:

- **Forecast for energy demand** covering 200 countries, 100 industrial sectors and 3000 cities around the world, with a monthly representation of the future markets trends and prices for commodities.
- **Global economics models** used to perform macroeconomic simulations, it cover fossil fuel demand and supply for 46 countries in detail and other 35 additional economies just to cover the 98% of the world GDP. Those models also aggregate the data form OPEC supply and other key zones in order to perform price forecasting.
- **Risk analysis** on an economic, political, and operational levels, mixed with the latest economic events and data releases that might rocketing investment speculation and influence on the commodity prices in short, medium and long term.
- **Evaluation of the economic impact** of companies to national markets, custom oil price scenarios, and tendencies in risk management for the energy and utilities sector.

In resume, computational intelligence applied to economic modelling and impact analysis provide to government, economist, and market actors with invaluable panoramas of the regulatory, economic and public decision making.

The forecasting for energy demand also supports other areas of the economic modelling such as:

- **Real options modelling** (risk-based decision tools), it provide strong elements of decision and support advice on the development of R&D strategies and portfolios.
- **Market modelling**, where the models are used for asset evaluations including risk assessment, the timing of new developments, gas and electricity procurement and contracting, forecast and scenario planning.

Nevertheless, more than only a buzz word used with a market motivation, the energy forecast is a scientific term broadly related with the energy efficiency field. The degree of importance of energy forecast and its impact on the world economy can be easily noticed in base of the statistics on energy efficiency performed by the international energy agency (IEA).

As their annual report document [42], It has measure over the **last 25 years a cumulative saving over USD 5.7 trillion in energy expenditures via energy efficiency**. This virtual supply of energy generates multiple benefits for governments, businesses, and households, including greater energy security from reduced dependence on energy imports and billions of tons of greenhouse gas emissions reductions.

Major facts, presented by the agency on its annual report, states about the impact of the energy efficiency over the world; making highlights on the following areas:

- Energy efficiency investment returns and markets outlook:
    - In 2014, IEA countries are estimated to have avoided primary energy imports of natural gas, oil and coal, totaling at least 7790 Petajoules (PJ) (190 Mtoe), and saving USD 80 billion in import bills.
    - In IEA countries, the avoided consumption generated by energy efficiency investments increased by 10% (1 930 PJ) in 2014 – the fastest rate in almost a decade.
- Tracking of the energy efficiency progress:
    - Energy efficiency has had the greatest impact in the residential sector, where the efficiency effect is estimated to have led to a cumulative TFC reduction of 19 exajoules (EJ) (463 Mtoe) between 2002 and 2012 in the IEA-18.
    - Improved data collection and analysis will help governments and other stakeholders to better track energy efficiency developments.
- Efficiency markets for buildings:
    - Global energy efficiency investment in buildings (excluding appliances) is estimated to have been USD 90 billion (+/- 10%) in 2014, with significant potential for additional profitable investments. Investment in three countries alone –China, Germany and the United States – is estimated to have been USD 59 billion.
    - Investment in energy efficiency in buildings globally is growing more rapidly than overall growth of building construction.
    - Global energy efficiency investment in buildings is projected to increase to over USD 125 billion (excluding appliances) by 2020.
- Energy efficiency in the electricity system and the outlook for utility efficiency investments:
    - Energy efficiency improvements since 1990 drove savings of 2200 TWh in 2014 in International Energy Agency (IEA) member countries, equaling about 24% of total electricity demand. Growth in electricity consumption has flattened

across OECD countries, from a peak of 9 385 TWh in 2007 to 9 355 TWh in 2013. Total electricity demand in the OECD is projected to increase by an average of 0.8% per year through 2020 [139].

- In addition to delivering electricity, utilities are important players in energy efficiency markets, spending over USD 13 billion in 2013 on end-use energy efficiency improvements. Utilities also invest in generation, transmission and distribution (T&D), and metering infrastructure that improves efficiency and reliability of the electricity grid.

The report continues with an analysis over the impact of lower electricity demand over the sales of traditional utility business model. This scenario push to Governments and utilities to renew their policies and business models in order to sustain investments, while also keeping an eye over long-term climate and energy demand challenges.

In non-OECD regions, electricity demand is still increasing. This can be translated to better opportunities to generate value from energy efficiency, maximizing the profits over utilities investment targeting the supply-side and end-use energy efficiency on news deployments due to population increases.

A side effect of energy efficiency policies is generated do to the transport and distribution investments. Those bring technical improvements to infrastructure reducing the losses, which implies an increase on the reliability of the power supply, giving the possibility to electricity access to more customers and reducing the cost of expanding the infrastructure.

As summary, the most important facts to takeaway on the report are:

- The energy intensity of countries belonging to the Organization for Economic Co-operation and Development (OECD) improved by 2.3% in 2014.
- Energy efficiency improvements in International Energy Agency (IEA) countries since 1990 have avoided a cumulative 10.2 billion tons of CO2 emissions.
- Investments worldwide in energy efficiency in buildings, which account for more than 30% of global energy demand, are estimated to be USD 90 billion (+/- 10%) and are set to expand.
- Electricity consumption in IEA countries has flattened partly as a result of energy efficiency improvements; energy efficiency investments since 1990 saved 2200 terawatt hours (TWh) in 2014.

These facts give a proof of the global importance of the energy efficiency at the economic and political level; also make clear that data collection and analysis is the pillar of present and future research, policies, and deployments.

The **energy modelling & forecasting** is inherently connected with energy efficiency analysis; because they provide accurate predictions, and the tools which support those ones, over the elements of the energy demand at many aggregated levels. Their results allows to analyst and energy players to obtain extrapolations of the future state of energy commodities and profiles.

IEA has standardize the level of the energy efficiency investment per country identifying three key drivers: a supportive policy environment, rising energy prices, and recent changes in energy efficiency indicators as a guide to momentum. **These key drivers have provide the fundament of this thesis, giving a credible economic and political motivational background**.

The **policy driver indicates the extent to which best practice policies are in place in a country** based on IEA Energy Efficiency Policies and Measures (EE PAMS). The sector categories analyzed are Cross-sectorial, Energy utilities, Industry, Existing buildings, New buildings, Appliances, Lighting, and Transport.

This driver has provide the research delimitations in the form of end-user sectors to analyze, it means that energy consumption databases used in this thesis are samples of the most important sectors observed.

The policy types listed in EE PAMS consist on Regulatory instruments, Policy support, Economic instruments, Information and education, Voluntary approaches (public-private and private sector), and Research, development, and deployment (RD&D) (research programme, demonstration project). Because the design of a forecasting system and the deployment of an energy management system is framed inside of the policy RD&D, the analysis and conclusions presented below only have been related with this policy category.

The **price driver** reflects the extent to which end-user prices can be expected to affect the potential for energy efficiency investment; the rate of increase, if significant, has an important impact on end-user behavior and thus on markets, while prices including taxes are used because these are the prices faced by end-users and which ultimately affect energy efficiency markets.

The **performance driver** provides quantified evidence of changes in energy intensity and efficiency in total final consumption (TFC) in 2012 relative to 2002, decomposed by factors; it is measured using the decomposition of IEA energy efficiency indicators.

The performance driver and price driver have been analyzed in order to make clear the economical context in which this thesis has been developed. In addition, those indicators provide valuable information about the suitability of new entrepreneurship in the area of energy efficiency.

On the following tables, five countries have been analyzed (Australia, Germany, Spain, Sweden, United states). Those countries have been recognized as the most open countries for the international surveillance on his energy sector and sharing of public data. Acknowledging that is a limited sample, the Snapshots do provide some interesting insights when examined collectively.

Table 25. Policy driver table (RD&D)

	Australia	Germany	Spain	Sweden	United states
Cross-sectoral	0	2	0	1	1
Energy utilities	0	1	0	0	1
Industry	0	2	0	0	1
Existing buildings	0	2	0	1	1
New buildings	0	1	0	1	1
appliances	0	1	0	1	1
lightning	0	1	0	1	1
transport	1	2	0	0	2

1 = several relevant policies are in place. 1 = at least one relevant policy is in place. 0 = no relevant policies have been identified.

Table 26. Price driver table (weigthed price of one unit of energy, percentage increases, 2012-2014)

	Australia	Germany	Spain	Sweden	United states
household	31 %	43 %	39 %	36 %	54 %
Industry	37 %	63 %	47 %	57 %	47 %

Table 27. Performance driver table (change in TFC in 2012 relative to 2002, measuring the efficiency effect)

AUSTRALIA					
	total	residential	Industry and services	Passenger transport	Freight transport
TFC	-12 %	13.43 %	6.81 %	13.29 %	32.41 %
Activity effect	30.9 %	16.94 %	35.14 %	22.48 %	45.74 %
Structure effect	-8.1 %	10.49 %	-16.15 %	-0.53 %	-4.59 %
Efficiency effect	-6.9 %	-12.2 %	-5.74 %	-7.74 %	-4.77 %
GERMANY					

	total	residential	Industry and services	Passenger transport	Freight transport
TFC	-5.4 %	-16.33 %	0.91 %	6.32 %	5.85 %
Activity effect	9.0 %	-0.68 %	15.11 %	5.11 %	23.17 %
Structure effect	-0.1 %	9.63 %	-5.31 %	-0.52 %	2.32 %
Efficiency effect	-13.5 %	-23.15 %	-7.41 %	-10.41 %	-16.00 %

SPAIN

	total	residential	Industry and services	Passenger transport	Freight transport
TFC	-7.0 %	-0.05 %	-5.63 %	14.43 %	-31.55 %
Activity effect	3.0 %	12 %	12 %	2 %	-20 %
Structure effect	-9.3 %	17 %	-20 %	-5 %	-1 %
Efficiency effect	-1.8 %	-24 %	5 %	18 %	-14 %

SWEDEN

	total	residential	Industry and services	Passenger transport	Freight transport
TFC	-12.5 %	-15.68 %	-14.14 %	-9.56 %	5.06 %
Activity effect	14.6 %	6.65 %	23.85 %	5.28 %	-1.16 %
Structure effect	-9.0 %	-2.31 %	-13.4 %	-1.66 %	-5.58 %
Efficiency effect	-16.7 %	-19.07 %	-19.94 %	-12.65 %	12.58 %

UNITED STATES

	total	residential	Industry and services	Passenger transport	Freight transport
TFC	-6.8 %	-5.67 %	-10.82 %	-4.62 %	-0.45 %
Activity effect	5.6 %	9.13 %	15.65 %	-5.4 %	5.48 %
Structure effect	-5.5 %	-7.17 %	-10.10 %	-1.41 %	2.79 %
Efficiency effect	-7.3 %	-6.89 %	-14.23 %	2.27 %	-8.19 %

Based on the results shown in the Snapshot for the overall of countries, is clearly presented the case that **all countries need to scale up investment in energy efficiency**, from both public and private sector sources (at EU level, much of this is being done by the member states). That presents an interesting opportunity for the growth of energy forecasting applications on Europe.

The RD&D policy Snapshot show that policies directed at the buildings sector are the most widely implemented, followed by cross-sectoral policies and transport policies. New buildings are almost as well covered as existing buildings, despite their much smaller share of the building stock. Energy utilities and lighting are the least well-covered sectors, maybe because the lack of private portfolios or public resources dedicated.

The IEA report remark that Information, education and economic instruments are the most widely implemented. About RD&D at Energy efficiency, it may be covered under broader research programs in many countries, which means that it is less visible and harder to delimitate and monitor.

There is a wide opinion that energy efficiency technologies are already fully mature. That is clearly showed because the market of energy efficiency continues without a dominant competitor neither regularization.

This implies that funding on R&D for energy efficiency technologies is still required, and **projects based on computational intelligence** instead to decrease his popularity continues and will be dominant on this areas. This statement may be demonstrated based on the poor performance obtained by Spain and Australia and their efforts to strength their energy management at research level.

Regarding to **price driver**, Countries have seen quite strong energy price increases over the period 2002-14, ranging from 33% for Australia up to 52% in the United States for the combined industry and households index. The price pillar may be a more important driver in the industry sector than in the residential sector: industry prices rose significantly faster than household prices in all countries except the United States. This is important to be consider because target the sector with more promissory for profits on energy efficiency actions.

Regarding to the **performance driver**, the countries had efficiency effect between 2002 and 2012 ranges from -16.7% in Sweden (the biggest improvement) to -1.8% in Spain. However, the decompositions at sector level are more informative. In the residential sector, for example, Spain had the largest efficiency effect at -24% but it obtains a bad result at industry and services 5%. This testifies that energy efficiency & management portfolios also public policies had only focused on the household sector.

The opposite case is presented on Germany case; it shows a high efficient level at residential sector and a medium result on industry. Otherwise, Sweden shows good statistics at all sectors except transport freight this is supported by the policy framework adopted by this country during the last years.

Market perspective of the energy efficiency: A glimpse of the European and Spanish ESCO's.

As it was mention before, The European Union and its member's states have dedicated large efforts in forms to policies framework and public funding in order to improve the energy efficiency at generation, transport, and demand sides for the purpose of to cut the energy waste and satisfy the growing electricity demand.

At the same time, on the demand side, users have been more concerned about the energy prices and try to avoid them using sustainable construction methods, and searching for financially viable long term solution in energy use.

This market demand have been supplied for **Energy Service Companies (ESCO's)** becoming integral part of the European energy efficiency market. They are able to offer financial solutions, technical and technological expertise, management creativity, market knowledge and communication abilities.

The Energy Efficiency Directive (EED, 2012/27/EU) defines an 'energy service provider' as a "natural or legal person who delivers energy services or other energy efficiency improvement measures in a final customer's facility or premises", while **'energy performance contracting' (EPC)** is understood as a "contractual arrangement between the beneficiary and the provider of an energy efficiency improvement measure, verified and monitored during the whole term of the contract, where investments (work, supply or service) in that measure are paid for in relation to a contractually agreed level of energy efficiency improvement or other agreed energy performance criterion, such as financial savings" [41].

On the European ESCO market report [41], the authors use a slightly different definition of an ESCO (an energy service provider, an energy efficiency provider, or energy service company), "a company that offers energy services which should include implementing energy-efficiency projects (and other sustainable energy projects)". The three main characteristics of the gainful activity on ESCO's delimited by the European Commission are:

- Guarantee of energy savings and/or provision of the same level of energy service at a lower cost.
- The remuneration of ESCOs is directly tied to the energy savings achieved.
- ESCOs can finance, or assist in arranging financing for the operation of an energy system by providing a savings guarantee.

Therefore, ESCOs can reduce their degree of risk via payments for the services delivered (either whole or partial) on the achievement of those energy efficiency improvements.

In principle, energy services include a wide range of activities, such as:

- Energy analysis and audits,

- Energy management,
- Project design and implementation,
- Maintenance and operation,
- Monitoring and evaluation of savings,
- Property/facility management,
- Energy and/or equipment supply,
- Provision of service (space heating/cooling, lighting, etc.),
- Advice and training

ESCO’s are not the only ones that offer energy efficiency services, others called **Energy Service Provider Companies (ESPC’s)** maintain in his portfolio options as supply and installation of energy-efficient equipment, the supply of energy, building refurbishment, maintenance operation, facility management, and educational services. They provide a service based on a fixed fee or in base to the added value obtained from the supply of energy of equipment. This fact demonstrate that the reduction of the energy consumption is not the first interest on their contracts.

The ESCO’s subscribe an EPC with their clients, their principles turn around a guarantee performance in terms of energy efficiency. An EPC may be two types of models, on the **guaranteed saving** model the client usually provide the project budget, therefore, the client will pay for the services of the ESCO and for performance guarantee in forms of energy savings.

The second model is called **shared savings**; on it, the ESCO provides financing for the investments who in returns obtain a share of the savings. The share of the savings is stablished based on the length of the contract, the payback time, and the risk taken. This models is more common in a starter market or after financial problems because the clients have limited access to capital and prefer an ESCO project over own financing. One variation of the shared savings is the first out, it declare the validity of the contract based on the level of savings achieved.

Other contracts different from EPC, could also be supply for ESCOS, between them exist the **Delivery contracting (DC)** model which is focused on the supply of a set of energy services such as HVAC mainly via outsourcing the energy supply. Chauffage, on this arrangement the supplier guarantee the service and his cost is based on the current bills minus certain level of monetary savings.

Inside of Europe, countries have different versions of those contracts. Some cases are UK and Ireland where the ESCO-type work is referred as Contract energy management (CEM). In italy, Chauffage contract is equivalent to “Energy Service Plus” contracts wich reduce the energy heating consumption in 10% during winter. In Nordic countries the DC are called comfort contracting, and beyond just deliver energy it take care of full maintenance and aesthetics.

In terms of policy framework, the European commission have been applied a set of actions to boost the European and national ESCO markets, it can be summarized as:

- Directives ESD (2006/32/EC), EED (2012/27/EU), EED (2003/54/EC)
- prEN15900 standard
- EU EPC campaign
- European Energy Efficiency Fund (EEE–F)
- ESCO market research (done regularly by the EC JRC)
- Database (JRC and Transparense)
- IEE projects, such as Eurocontract, EMEEES, ChangeBest, Permanent, Transparense, EESI, EESI2020, Combines, etc.
- FP7 projects: good examples, business models

The directives are the most important element because they impose the active supporting on the development of an ESCO market on EU. In response, regional authorities have prepared a "portfolio of flexible mechanisms" which included the formation of ESCO networks; customer oriented information, model contracts, credit lines, guidelines for contract process, calls to implement energy services in public buildings and project evaluations.

European ESCO market.

The European ESCO market was estimated on 2013 around 10.3-12.6 billion per year [41]. Countries such as France, Germany, Italy, Spain and UK seems to have the biggest and well stablished markets. On the other hand, countries such as Belgium, Bulgaria, Austria and Denmark shows a non-exploited and promissory market.

As general opinion, the EU-ESCO market is classified as demand driven, it means that potential ESCO clients actively search for suppliers. They define and communicate their needs and requirements for an energy services project, waiting for the adequate financial supported solution.

However, exist some business and technological barriers that delay the booming of this market at maximum levels. The principal one is the lack of trust by the clients in the markets due to inhomogeneous ESCO offers in the market, this statement is supported by the following reasons: lack of experience of clients, lack of information with visible references, lack of proper measurement and verification practices.

The second barrier was identified as the Lack of well-established partnerships between ESCOs and sub-contractors. This inevitably ends with failed projects, due to the poor execution or maintenance, and financial insecurities on behalf of the facilitators. On the pro-side, the financial crisis had increased the attention over cost reductions through energy efficiency and advantages of the flexible financing offered by ESCOs (such as third part financing and shared savings).

Other barriers are related with the non-homogenous and complex legislative framework, the international accounting rules that harden financing opportunities and the fluctuating and non-regulated energy prices, most of the time determined by a supply monopoly.

On the other side, the success factors were identified on the business and political framework, between them the more relevant are: the good will of the European countries on build a legislative framework to regularize the ESCO market, the market liberalization, the environmental awareness and the institutionalization of the energy efficient market as solution.

Spanish ESCO market

The Spanish ESCO market has been increasing their participation on the public sector, with partners as local and autonomies authorities, and in the private sectors. Authors [41] refer to it as dependent of large national programs during the periods 2005-2007 and 2007-2010, with a fast rise between 2011 and 2013.

The referred public ESCO projects have been focused primarily on public lighting and public buildings. Third parties on the private sector have been triggering projects on private non-residential buildings, industries involving cogeneration, audits and HVAC control systems.

The IDAE (Instituto para la Diversificación y Ahorro de la Energía, a Spanish National Energy Agency) estimate 800 ESCO companies on 2013. About 60-70% of the market players are local or national, while the other 30-40% are sister companies of large international ESCO giants [41]

The Spanish ESCO market were estimated at 2015 in €400-500 million and 2016 in €1 billion, calculating with all costs, i.e. energy supply costs, investments plus maintenance and considering all types of projects [41].

Main ESCO area is public lighting (installing LED solutions and control systems), it represent about 90% of all public projects. Public buildings and water supply renovations in the public sector are the second area in importance.

Private clients such as private hotels, corporate buildings, sports facilities, heating systems in apartment buildings and big industries in the private sector constitute the third area. Main demand side technologies affected by ESCO have been HVAC, lighting, refurbishment, automation systems, pumps, motors and control systems.

Technological perspective of the energy efficiency

On the energy efficient market, ESCO's and other energy services providers use technological resources to guarantee the success of the projects during the execution and maintenance stages. The essence of this continuous monitoring underlying on to facilitate the information exchange between the energy usage point and the system operator. These technologies not only impact the usage of the energy the demand side abut also help the financial and enterprise level operations.

On this scenario, technological companies and researchers collaborate to create a solution from the convergence of monitoring, decision-making, and automatization technologies; used for energy managing and cost efficient savings. The objective of these emerging technologies is the modification of the demand on key processes and equipment, optimizing the energy demand to the supply capacity and fitting it to the parameters established by the operator.

Actors involved on the control of moderns grids.

Modern grids (as it shown on **Figure 57**), and by consequence electricity markets, constitutes a complex framework of rights, obligations, and exchange of information; and in the center of the mapping exists the actors that fulfills the activities from the physical layer to the administrative ones.

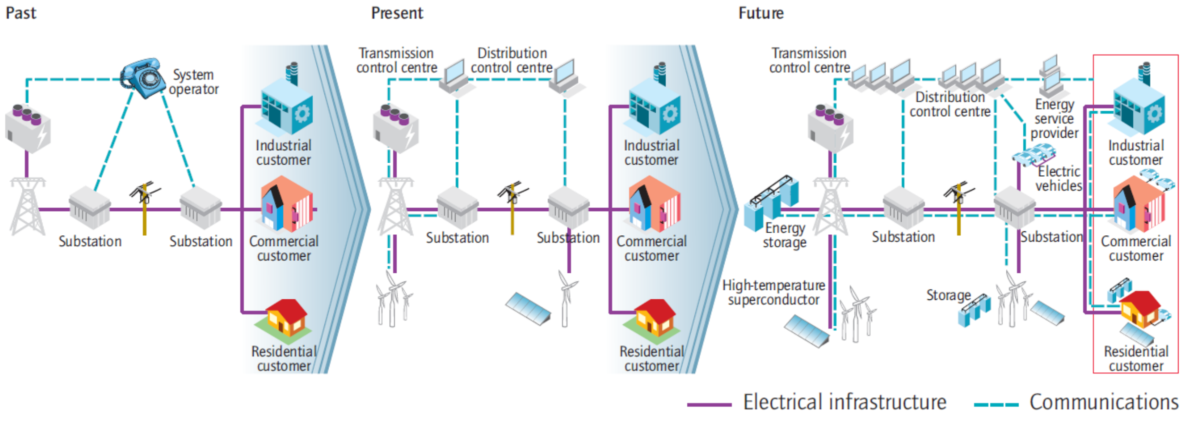


Figure 57. Evolution of the modern electricity grid (source IEA).

On the next paragraphs, we will cover the rights and obligations over the electricity market information exchange among the actors shown on **Figure 58**:

- Transmission System Operator (TSO)
- Distribution System Operator (DSO)
- Electricity Supplier
- Customer

- Data Exchange Platform
- Third Party, including Energy Service Companies

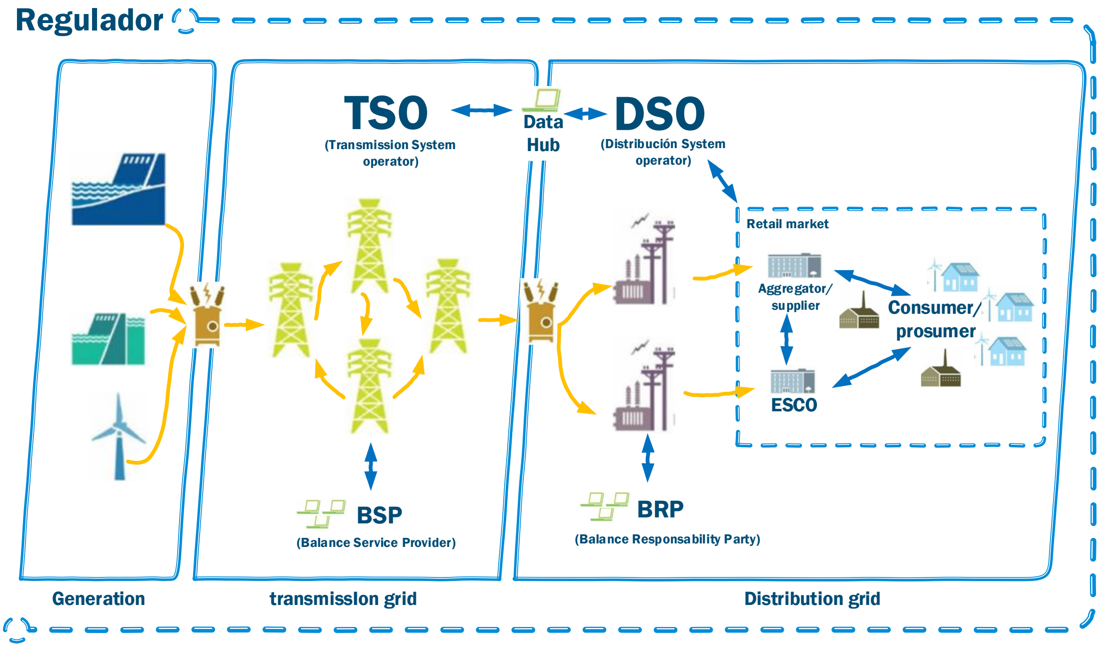


Figure 58. Actors of the electricity grid.

Due to the definitions about the rights and obligations in relation to the exchange of information among the actors are part of the national competences and the model adopted by countries, we will use the internationally harmonized role models. It allows covers the most important types of interactions between market actors. Besides, the roles will be solely used descriptively and not carry any normative consequence, being these ones a guideline will serve to compare the countries situations.

On the other hand, the rights and obligations will be described around the most relevant business processes among actors such as metering, Supplier switching, moving, end and start of supply, balance settlement, and access to customer data. Within these business processes, the following types of information exchange will be also addressed: meter data reading, data validation, obligation to deliver information, right to access information, operation of data exchange platform, data protection and integrity, and interaction between HUB's.

Information exchange regards to the communication of the electricity consumption parameters between network companies, TSO, suppliers, end-users, and other market players. The information shared can include consumption measured in kWh over a specified time, peak consumption, name, address of customer, and market and network data such as prices, network tariffs, power flows, outages, voltage quality.

In order to study and create a map of the roles and responsibilities of TSO and DSO regarding to communications between data-hubs and information exchange systems, we will develop the description of the actors around the followings principal points:

- The obligations of TSOs and DSOs concerning to the information exchange with electricity suppliers, energy service companies (ESCOs), customers and national communication systems such as hubs.
- The rights of suppliers, ESCOs and customers to access hubs/information exchange systems, both the types of information, accessibility and economic terms
- Rules of conduct for gaining access to a hub/information exchange system
- Interaction between hubs and information exchange systems across borders
- National plans for further harmonization in light of ongoing work at the EU level

TSO: Transmission System Operator

The TSO is responsible for operation and development of the Data-Hub, a mandatory centralized data exchange platform. Market players have the compulsory obligation to communicate solely with the Data-Hub (centralized), meaning that all information is being sent to and received from the Data-Hub. Data-Hub is thus the Metered Data Aggregator and the Metered Data Administrator.

The TSO is also imbalance settlement responsible and is final responsible party for the financial balancing of electricity consumption (and nomination). TSO receives data from the Data-Hub to fulfil this function. TSO are also responsible for development of data exchange platforms meaning that on some cases the operation of the data-hub could be allocated on the TSO too.

DSO: Distribution System Operator

DSO operates the distribution network and performs all meter readings (Metered Data Collector). They send all metering data and meter values to the Data-Hub. DSO remains responsible for connection of customers to the grid and for data validation (data quality). DSO can choose to be the Metering Point Administrator or outsource this to a third party; however, it will not be able to outsource the legal responsibility for meter administration.

DSOs are currently responsible for most of the functions regarding information exchange. This includes meter operation, data collection, data storage, meter data validation, and distribution of data to other market participants. Exceptions to this are the countries that have a data-hub in operation, here the data storage function lies with the operator of the data-hub. With the development of more data-hubs, it is expected that more responsibilities transfer from DSOs to data-hubs.

Supplier

A supplier sells electricity to customers (end-users) and is their main contact point. The supplier is billed by TSO and DSOs for grid operation costs. The supplier invoices the grid operation costs and electricity usage to the final customer including taxes. The supplier subsequently pays TSO and DSOs for grid operation costs. TSO and DSO remain responsible for payment of taxes to the tax authorities.

Suppliers are responsible for balance settlement for their customer portfolios. In case of data hubs and a supplier centric model, suppliers will operate as the main contact person for consumers. In all countries, customers have access to their own data. It differs per country if it is the responsibility of the supplier or the DSO to provide the consumer this access It depends mostly on the roll-out of smart meters what level of detail the consumption data is.

End user

The end-user (customer) has access to its own data (usage). The roll-out of smart meters will however impact the exchange of information between market players (including customers), this systems must provide final customers with information on actual time of use. Third parties (like ESCO or aggregators) should be granted access on behalf of the final customers for the purpose of comparing the consumption.

ESCO: Energy Service Company

ESCOs fall within the category of legitimate interested party and are not a market player. They can be allowed to access customers metering information via a written document (contract or attorney power).

Data Hub

The data-hub is developed and operated by the TSO while The DSO remains responsible for the physical meter, meter reading, and meter data validation. The data-hub serves as centralized data storage to which DSOs submit meter data, suppliers submit personal data of consumers and TSO submits balance settlement information.

Harmonization of the electricity roles and standards

Harmonization towards the opening of an european bilateral marketing have been a target since 2006. It pursues a harmonization in the definition of contracts, rules, and roles. The upper motivation will cause the creation of a “European energy transaction passport,” which is necessary to supply and demand a European scale consequently with a broader market, facilitating the selling energy on to European borders.

A key issue for harmonization is the development of efficient information exchange infrastructures between end users, DSOs, TSOs and other market agents, including new entrants such as Energy Service Companies (ESCOs).

Balancing on the market

Balancing refers to the capacity of the grid to keep in balance the demand and the supply, and it relies on the trading activities on four different time slots (day-ahead, intraday, gate closure time to real time and real time).

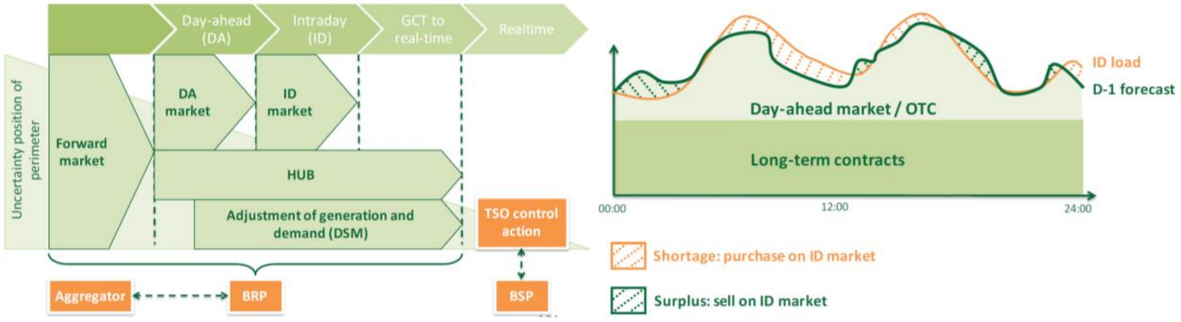


Figure 59. How the market is balanced based on the energy procurement strategy.

The actors responsible for this action are two: the balance responsible party (BRP) on the distribution side, and responsible for maintain a time-slot rescheduling on their portfolio each X minutes and do payments of imbalance tariffs to TSO; and the Balance service provider (BSP) which role is mostly assumed by the generator company but also could be the role of the prosumers. They deliver balance to TSO as a service whereby they obtain a remuneration.

In order perform this action, the TSO must be able to guarantee the procurement of balancing services, the settlement of imbalance volumes, and ensure a safe control zone for balance on their control area. Also, the DSO must to perform switching actions in base on demand and generation control, securing the distribution system operation and avoiding overloading the distribution grid.

Flexibility

Flexibility is the modification of generation injection and/or consumption pattern in reaction to an external signal (price signal activation, peak consumption, alarms…) in order to provide a service within the energy system. These actions are demanded in order to optimize the portfolio, provide balancing, and manage congestions usually caused by errors on the forecasting.

Could be equivalent to the capacity of the grid and their actors to fulfill the demand constrains in time, it is often associated to the capacity of the generators keep on track the demand but also counts the availability of decentralized generators and prosumers capacity. As example nuclear plants are categorized with low flexibility and gas/diesel engines have a high because these are able to operate as soon the requirement exist.

In order to increase the response to demand, the actors are forced to take the following actions; TSO must to prioritize balancing services in order to increase the efficiency on the match of generation technologies, the DSO must collaborate passively with the BRP to increase the variability on their portfolio.

Flexibility activities also could relies in another actor, the aggregator which manages the flexibility of a cluster of flexible devices with the purpose of offer demand responses services to the different power systems participant thought various markets. Local flexibility on demand side could comprises large or small scale from EV`s charging point at homes to parking's, from public lighting to industrial HVAC, but always it depends on the aggregation degree.

Business Needs of Load Forecasting

From the commercial point of view, the idea of energy management applied to industrial and large users has been around for few years now. Manufacturing and software companies have embraced this idea and have been trying to improve the energy management systems that can be used in manufacturing plants.

The current EMS market can be segmented into large software providers and small & medium software providers. The large players offer standard software solutions that are applicable to diverse manufacturing plants with little room for advanced customization. They offer broad features, robust mathematical analysis, and standard integration with corporate software. Powerful interfacing tools and a wide range of data presentations are also common features. The small and medium software providers, frequently oriented to home and tertiary buildings, offer closer customization, module approach to address specific problem, and agile response to customer needs.

Most of the current EMS offer address data acquisition, mathematical analytics, cloud-based software as a service (SaaS), financial analysis, and monitoring at different levels of sophistication. However, **there is a lack of available products based on advanced analysis supported by machine learning Intelligence & automated energy optimization without human intervention; energy-related maintenance by correlating the states and consumptions; and module oriented design that enables plug and play feature for many independent functions such as optimization, data mining, knowledge discover, reports & analytics, automatization and others.**

Load forecasting based on machine learning play a main role on the EMS features, it principal contributions are integrated on operations such as planning, control, optimization, maintenance and diagnosis. The most important business needs of the end users related with energy forecasting can be summarized on the following:

- **Supervision, monitoring & Productive planning.** On the house holding sector real time monitoring and supervision constitute the first step, and regularly the unique one, on the energy management process; this condition is caused because the control actions intended to reduce and optimize the energy consumption are limited by the execution of the client preferences.
- This means that house holding clients regularly perform the plot of energy experts following the current and forecasted load profiles and compare them with previous period.

- Due to the high level of automatization on enterprise users, the load forecasting are dropped on computational processes able to gather big amounts of data and perform an intelligent dimensional reduction of the data. The result are alarms, information about production and process which allows to human expert to supervise and planning.
- **Fault detection and diagnosis**. Industrial users and buildings improve the ratability of their operation in base of the find and correction of malfunctions on their equipment. The implementation of load forecasting algorithms as a way to compare anomalous behaviors on the loads allow to detect possible failures states associated with the problem.
- **Predictive maintenance.** Industrial users also can use the expected demand, expected price and productive information to create an automated response and suggest predictive maintenance in terms of the economical convenience.
- **Dynamic control of the loads and system optimization.** Large consumers with aggregated loads can optimize their energy performance of their loads by means of data driven modelling; using the forecasted consumption of the loads, Automated control systems can decide the optimal energy source regarding customizable criteria to supply the energy demand.

According to the lead time range of each business need described above, the minimum updating cycle and maximum horizon of the forecasts are summarized in Chapter 2.

Table 28. Needs of forecasts in large end users.

Business need	**Minimum updating cycle**	**Max horizon**
Supervision, monitoring & Productive planning.	5 min	1 years and above
Fault detection and diagnosis.	15 minutes	1 day and above
Predictive maintenance.	15 minutes	1 years and above
Dynamic control of the loads and system optimization.	5 min	10 years and above

Load forecasting is not only limited to energy management activities at the user side. It also constitute the pillar of the decisions made on inner departments of utilities such as planning, operations, trading, among others. The most important business needs of the utilities related with energy forecasting can be summarized on the following:

- **Energy purchasing.** Whether a utility purchases its own energy supplies from the market place, or outsources this function to other parties, load forecasts are essential for purchasing energy. The utilities can perform bi-lateral purchases and asset commitment in the long term, e.g., 10 years ahead. They can also do hedging and block purchases one month to 3 years ahead, and adjust (buy or sell) the energy purchase in the day-ahead market.

- **Transmission and distribution (T&D) planning**. Utilities need to maintain and upgrade the interconnection systems in order to satisfy the growth of demand and improve the reliability in the territory which they serve. This activities also include real state considerations about the placing the substations in the future.
- These planning decisions heavily rely on the forecasts, known as spatial load forecasts, that contain information about the when, where, and how many loads as well as customers will grow.
- **Operations and maintenance**. In daily operations, load patterns obtained during the load forecasting process guide the system operators to stablish switching and loading decisions, as well as schedule maintenance outages.
- **Demand side management (DSM)**. Although lots of DSM activities are belong to daily operations, it is worthwhile to separate DSM from the operations category due to its importance in this smart-grid world. A load forecast can support the decisions in load control and voltage reduction. On the other hand, through the studies performed during load forecasting, utilities can perform long term planning according to the characteristics of the end-use behavior of certain customers.
- **Financial planning**. The load forecasts can also help the executives of the utilities project medium and long term revenues, make decisions during acquisitions, approve or disapprove project budgets, plan human resources and technologies, among others.

According to the lead time range of each business need described above, the minimum updating cycle and maximum horizon of the forecasts are summarized in **Table 29**.

Table 29. Needs of forecasts in utilities.

Business need	Minimum updating cycle	Max horizon	VSTLF	STLF	MTLF	LTLF
Energy purchasing	1 hour	10 years and above	x	x	x	x
T&D planning	1 day	30 years		x	x	x
Operations	15 minutes	2 weeks	x	x		
DSM	15 minutes	10 years and above	x	x	x	x
Financial planning	1 month	10 years and above			x	x

Conclusions & takeaways

Mayor takeaways of this section are:

- Regulatory reforms and geopolitical uncertainty has been key drivers on the transformation of the energy economic. In this scenario, the computational intelligence has been playing and important role. It has been providing the energy sector with projections and hypothesis for the world economy, analysis of the future trends in global industry, exhaustive forecast on local consumer billing, energy prices projections, and drivers or triggers for economic and political risk.
- At European and global levels, the international energy agency is the organism in charge of present's major facts about the impact of the energy efficiency. The Energy Efficiency Policy framework is measured impact is measured trough the analysis of fundamental sector categories such as industry, building and transport applications.
- Data shows that energy efficiency technologies are already fully mature and the European market of energy efficiency continues without a dominant competitor neither regularization.
- The European ESCO's have been exploiting this market estimated on 2013 around 10.3-12.6 billion per year [41]. Countries such as France, Germany, Italy, Spain and UK seems to have the biggest and well stablished markets. However, exist some business and technological barriers that delay the booming of this market at maximum levels.
- The Spanish ESCO market were estimated at 2015 in €400-500 million and 2016 in €1 billion, calculating with all costs, i.e. energy supply costs, investments plus maintenance and considering all types of projects [41]. This market is considered still in growth due to the lack of advance on energy efficiency services and the big exploitation potential.
- One of the most spread and well-known technology, which can help on the energy delivering and consumption issues, is the energy management systems (EMS). EMS allows collect, analyze, and share critical information to understand, control, and optimize the energy consumption across different consumption levels.
- **Load modelling and forecasting** is the central pillar of any intelligent process inside energy management systems. It constitutes a helper to solve the lack of available products based on advanced analysis supported by machine learning.

In summary, actors at Energy efficiency and management business are not well stablished on European markets stills, giving to new competitors the opportunity to take a piece of cake and test new business proposals before policies changes.

In the center of this the energy management systems implemented don't count with any intellectual protection, and most of the time the merely protection is just the industrial secret. This gives a special chance to industry to improve their systems based on scientific literature.

This thesis have been planned on the frame of these problems, aiming for improve the load forecasting as the principal element of any energy management process. Using databases from large end users as industries and buildings, houses, and entire countries, we present several contributions to the state of the art with a scientific and economic orientation.