January 2018

A Multi-Stage Electricity Price Forecasting For Day-Ahead Markets

Radhakrishnan Angamuthu Chinnathambi

A MULTI-STAGE ELECTRICITY PRICE FORECASTING FOR DAY-AHEAD MARKETS

by

Radhakrishnan Angamuthu Chinnathambi,
Bachelor of Engineering, Anna University

A Thesis

Submitted to the Graduate Faculty

of the

University of North Dakota

in partial fulfillment of the requirements

for the degree of

Master of Science

Grand Forks, North Dakota
May
2018

This thesis, submitted by Radhakrishnan Angamuthu Chinnathambi, in partial fulfillment of the requirements for the Degree of Master of Science from the University of North Dakota, has been read by the Faculty Advisory Committee under whom the work has been done and is hereby approved.

____

Prakash Ranganathan, Ph.D., Chairperson

____

Hossein Salehfar, Ph.D.

____

Saleh Faruque, Ph.D.

This thesis meets the standards for appearance, conforms to the style and format requirements of the Graduate School of the University of North Dakota, and is hereby approved.

____

Grant McGimpsey
Dean of the Graduate School

____

Date

PERMISSION

Title	A Multi-stage electricity price forecasting for day-ahead markets
Department	Electrical Engineering
Degree	Master of Science

In presenting this thesis in partial fulfillment of the requirements for a graduate degree from the University of North Dakota, I agree that the library of this University shall make it freely available for inspection. I further agree that permission for extensive copying for scholarly purposes may be granted by the professor who supervised my thesis work or, in his absence, by the chairperson of the department or the dean of the Graduate School. It is understood that any copying or publication or other use of this thesis or part thereof for financial gain shall not be allowed without my written permission. It is also understood that due recognition shall be given to me and to the University of North Dakota in any scholarly use which may be made of any material in my thesis.

Radhakrishnan Angamuthu Chinnathambi
May 2018

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

First of all, I want to thank my parents, A. Chinnathambi and C. Padmavathi and my brother A.C. Senthilvelan for supporting me throughout my life for my happiness. I am forever indebted to them.

I would like to express my sincere gratitude to my advisor, Dr. Prakash Ranganathan for his continuous guidance, support, and advice which he had provided from the day one. I'm very fortunate to have him as my advisor.

I am thankful to the members of my committee, Dr. Hossein Salefar and Dr. Saleh Faruque for their acceptance of being on my committee as well as their guidance. I would also like to acknowledge the role of the Department of Electrical Engineering at University of North Dakota (UND), Grand Forks, North Dakota for providing me the opportunity to study and conduct research.

My thanks also goes to my friends and colleagues here in Grand Forks for supporting me both academically and socially.

Above all, I am grateful to the Almighty God for giving me the capability and the chance to finish my thesis.

ABBREVIATIONS

MIBEL	Iberian Electricity Markets
RF	Random Forest
ARIMA	Auto Regressive Integrated Moving Average
GARCH	Generalized Autoregressive Conditional Heteroskedasticity
RRTP	Residential Real-time pricing
LOWESS	Locally Weighted Scatterplot Smoothing
GLM	Generalized Linear Model
SVM	Support Vector Machine
DNN	Deep Neural network
ANN	Artificial Neural Network
FFNN	Feed Forward Neural Network
MLP	Multilayer Perceptron
SDA	Stacked Denoising Auto Encoders
ELU	Exponential linear unit
RELU	Rectified linear unit
WMA	Weighted Moving Average
ARMA	Auto Regressive Moving Average
LASSO	Least Absolute Shrinkage and Selection Operator
SVR	Support Vector Regression

FL	Fuzzy Logic
LR	Linear Regression
MARS	Multivariate Adaptive Regression Splines
GCV	Generalized Cross Validation
RSS	Residual Sum of Squares
RF	Random Forest
AR	Auto Regressive
I	Integrated
MA	Moving Average
AIC	Akaike Information Criterion
BIC	Bayesian Information Criterion
SR	Standardized residuals
R	Residuals
MAPE	Mean absolute percentage error
TF	Tensor flow
SD	Standard Deviation

ABSTRACT

Forecasting hourly spot prices for real-time electricity usage is a challenging task. This thesis work investigates a series of price forecasting methods for day-ahead Iberian Electricity Markets (MIBEL). The dataset from MIBEL was used to train and test multiple forecast models. A hybrid combination of Auto Regressive Integrated Moving Average (ARIMA) and Generalized Linear Model (GLM) was proposed and its Mean Percentage Error (MAPE) values were compared against several methods. For example, ARIMA, GLM, Random forest (RF) and Support Vector Machines (SVM) methods are investigated. The results indicate a significant improvement in MAPE and correlation coefficient values for the proposed hybrid ARIMA-GLM method.

Forecasting hourly spot prices for real-time electricity markets are key activities in energy trading operations. This thesis work specifically develop a novel two-stage approach that uses a combination of Auto-Regressive Integrated Moving Average (ARIMA) with other models to improve residual errors in predicting the hourly spot prices. In Stage-1, the day-ahead price is forecasted using ARIMA, and then the resulting residuals are fed to another forecasting method in Stage-2. This approach was successfully tested with multiple duration periods ranging from one-week to ninety days for variables such as price, load, and temperature. A comprehensive set of 17 variables were included in the proposed model to predict the day-ahead electricity price. The results indicate a significant improvement in the Mean Absolute Percentage Error (MAPE) values compared to other present approaches.

To reduce the prediction error, three types of variable selection techniques such as Relative importance using Linear Regression (LR), Multivariate Adaptive Regression Splines (MARS), and Random forest (RF) were used. Four datasets (Three months, Six months, weekday, and weekend) were used to validate the performance of the model. Three different set of variables (17, 4, 2) were used in this study. At last, three common variables selected from these feature selection approaches were tested with all these datasets. Considerable reduction in MAPE for both three and six-month dataset were achieved by these variable selection approaches.

In addition, the work also investigate the application of a multi-layered deep neural network to the Iberian electric market (MIBEL) price forecasting task. A 3-month and 6-month of energy data are used to train the proposed model. The 3-month and 6-month period is treated as a historical dataset to train and predict the price for day-ahead markets. The network structure is implemented using Google's machine learning TensorFlow platform. Activation function such as Rectifier linear unit (ReLU) were tested to achieve a better Mean Absolute percentage error (MAPE) considering the weekday and weekend variations.

1 INTRODUCTION

The electricity markets are becoming sophisticated because of the recent changes in the trading structure for market bids on prices. These market usually include two instruments for trading: the pool and bilateral contract[1]. In the pool, both the consumers and producers submit bids which get cleared by the market operator. These operators announce the prices for the next day. The companies might also want to use bilateral contacts for a hedge against the risk of price volatility.

For both these instruments, price forecasting for the next day or next few months is vital for adjusting their bids to maximize the profit or for schedule outage, design load response and various decision-making process. The market clearing prices are publicly available for all the electricity market as it is the case of the day-ahead pool of mainland Spain (www.omel.es), the Californian pool (www.caiso.com), or the Australian national electricity market (www.aemo.com.au)

Therefore, an accurate price forecast will greatly help the consumers or producers in the bidding strategies and also in the price negotiation of the bilateral contract. This work focuses on the day-ahead price forecast of a daily electricity market using various statistical and computational intelligence models. This work provides models to forecast the next 24-hour market clearing prices for next day. These models provide reliable estimates of forecasts of prices in the Iberian electricity market of mainland Spain and Portugal.

1.1 The Electricity Market Representation

The electricity spot market is a day-ahead market in which the prices for the next day is finalized before a particular market closing time. This is different from the commodity or financial markets which allows continuous trading[2]. The system operators require advanced notice to check whether the schedule falls under transmission constraints. The agents usually submit their bids for each hour of the next day in the day-ahead electricity market before a particular market closing time. We define average of the 24 hourly prices as daily spot price or the base-load price. The average for the on-peak hourly prices typically range from 8 am to 8 pm is called as the peak load price.

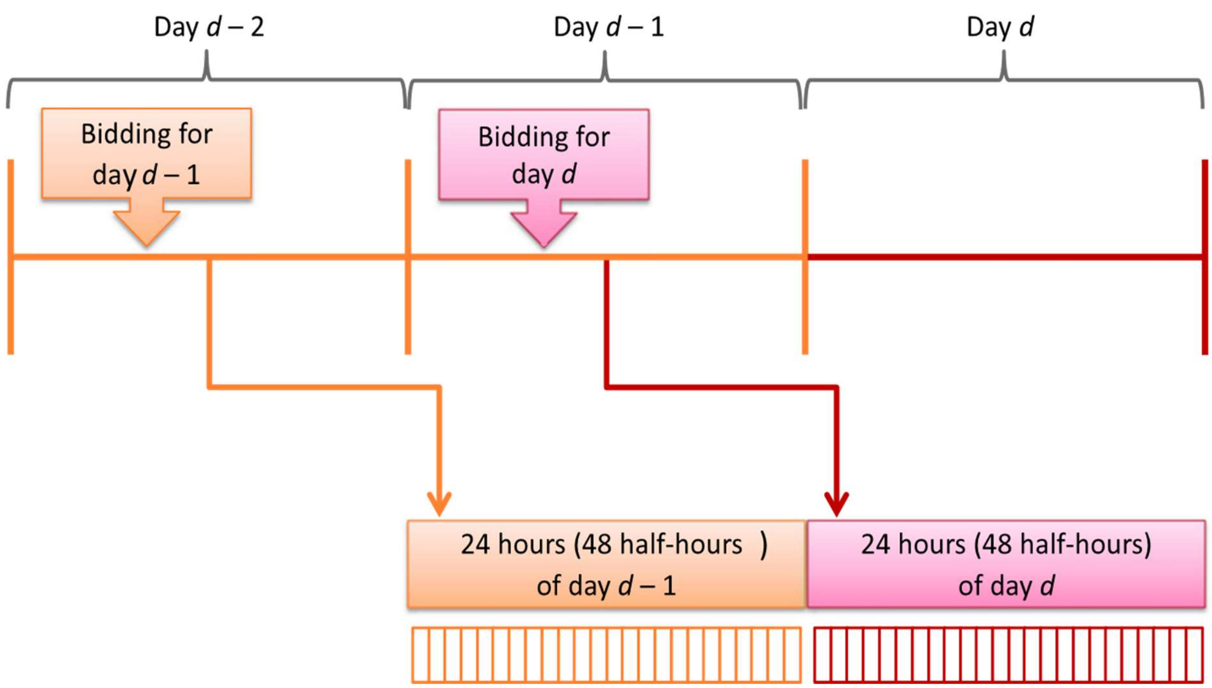

Figure 1. Electricity Market representation[3]

1.2 Thesis Contributions:

The following are the objectives of this work:

Objective Develop forecasting models for time series price data from Iberian electricity market (MIBEL) which serves the mainland areas of Spain and Portugal.

To accomplish this objective, following tasks were carried out:

Task 1 Conduct literature review of various price forecasting algorithms suitable for time-series datasets.

Task 2 Test various statistical and machine learning techniques such as auto regressive integrated moving average (ARIMA), random forest(RF), Support vector machines (SVM), locally weighted scatterplot smoothing (LOWESS), Generalized linear models(GLM) to Iberian market datasets.

Task 3 Different multi-stage hybrid forecasting techniques were tested along with ARIMA. This thesis focusses on the day-ahead price forecast for the Iberian electricity market using different hybrid techniques such as ARIMA-RF, ARIMA-SVM, ARIMA-GLM, ARIMA-ARIMA, and ARIMA-LOWESS.

Task 4 These techniques were investigated for various duration of datasets such as one week, two weeks, three weeks, one month, 45 days, 60 days, 75 days and 90 days. These techniques were also tested for weekday and weekend datasets for one month, two months, three months and six months duration of datasets. This two-stage ARIMA model is also tested for a dataset with/ without explanatory variables in stage-2 to understand the influence of the residual prediction. Finally, the results were compared with the existing literature for the same Iberian market to strengthen the fact that this hybrid model is a promising technique for short term price forecasting.

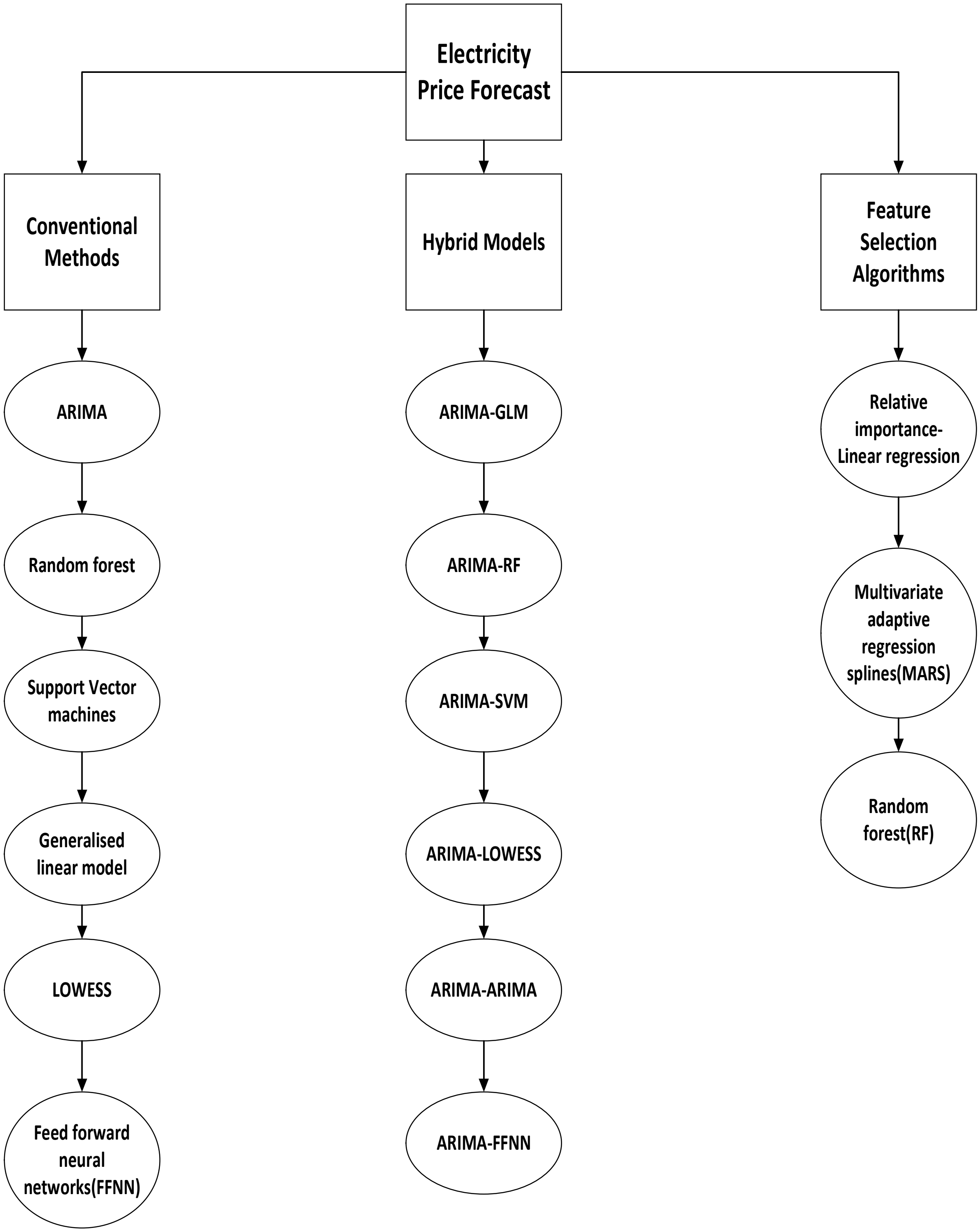

Figure 2: List of techniques used in this study

Task 5 Test state of the art tool to forecast the day-ahead price of the Iberian electricity market. A multi-layered deep neural network was tested using Google's machine learning TensorFlow platform.

Task 6 Reduce the prediction error using different variable selection approaches such as Relative importance using Linear Regression (LR), Multivariate Adaptive Regression Splines (MARS), and Random forest. Three common variables selected from these feature selection approaches were tested with these datasets to reduce the error.

Task 7 A statistical comparison of the above-mentioned methods are done using an index such as mean absolute percentage error (MAPE).

1.3 Thesis Organization

The thesis is organized as follows. **Chapter 1** introduces the price forecasting problem. **Chapter 2** presents the literature review of the existing forecasting methods. **Chapter 3** talks about the use of the multi-stage techniques for the day-ahead price forecast of the Iberian electricity market. **Chapter 4** discusses the multi-layered deep neural network applied to the Iberian electricity market. **Chapter 5** presents the input variable selection using different techniques. Finally, **Chapter-6** summarizes the conclusion and provides direction for the future work.

2 LITERATURE REVIEW

2.1 Forecasting techniques

An excellent state-of-the-art review of electricity price forecasting that includes methods can be found in [2]. This paper discusses various modeling approaches such as concepts from multi-agent theory, reduced-form, statistical and computational intelligence. Weron in [2] discusses the strengths and weakness of the existing forecast methods and enforces the need for a robust error evaluation procedure. For smart grid applications, deployment of Auto Regressive Integrated Moving Average (ARIMA) techniques have been used for load forecasting and has some effectiveness considering the seasonality on weather, and also used in predicting the short term electricity price [4]–[10]. Datasets that have used ARIMA are for Spanish, Californian and EPEX power Markets [1]. In [11], an ensemble learning method known as Random Forest (RF) has been applied to predict next day price for New York electricity market. In [12], a Support Vector Machine (SVM) method has been applied in Australian Market. This technique was used as a hybrid model along with ARMAX and Least Square [13]–[15].

ARIMA has been extensively used for load forecasting applications and as the key method for forecasting short-term electricity price predictions [6-13]. In [16], authors combine ARIMA with wavelet transform and GARCH to investigate forecasting accuracies. In [17], authors used ARIMA and GARCH to predict real-time market price in

ComEd/PJM Residential Real-time pricing(RRTP) for the purpose of minimizing home electricity costs. In [18], ARIMA was combined with SVM and Neural Networks. In [4], authors used ARIMA to capture the impacts of economical, technical, and strategical risk factors in intra-day prices. In [5], authors combined a time-varying regression model with ARMA in a two-stage model after accounting for the impact of system load and wind power generation in the Western Danish price area of Nord Pool's Elspot. In [6], authors used the double seasonal ARIMA as a univariate method along with exponential smoothing and both are used as a benchmark for comparison with the multivariate methods such as feed-forward neural networks which include the explanatory variables such as wind generation and weekdays. In [7], authors proposed an improved forecasting model for New South Wales in Australia that detaches high volatility and daily seasonality based on empirical mode decomposition, seasonal adjustment and ARIMA. In [9], authors compared the accuracies of twelve time-series methods for California and Nordic markets. These methods include standard auto regression and their extension as well as mean-reverting jump diffusion. In [10], author has used ARMAX and Takagi-Sugeno-Kang model to predict the short-term electricity price in the Colombian electricity market. In [19], authors have used random forest method and compared it with ARMA for New York electricity market. This random forest adaptive model provided confidence intervals associated with the prediction and adjusts itself to the latest forecasting scenarios.

Deep neural network has been used in load and price forecasting by the researchers. In [20], authors have used a Stacked Denoising Auto encoders (SDA) model, a class of deep neural networks for the short-term price forecasting. In this paper, authors have found that SDA performs better than classical neural networks, support vector machines (SVM), multivariate adaptive regression splines (MARS) and least absolute shrinkage and selection operator (Lasso) for on-line forecasting and compared with industrial results for day-ahead forecasting. These models were tested for data collected from Nebraska, Arkansas, Louisiana, Texas, and Indiana hubs in U.S.

In [21], multivariable mutual information is applied for feature selection to select the appropriate features for the price forecasting. In this paper, support vector regression (SVR) is applied and the experimental results show that this feature selection methods perform accurate prediction than other feature selection methods. In [22] , feature selection techniques are compared and analyzed. It is used as a filter prior to forecasting method. The popular search methods such as Best-First Search, Greedy-Step Wise Search, Exhaustive Search, Genetic Search, Random Search, and Ranker is compared with the proposed feature selection technique. In[23], new feature selection method is presented and a hybrid filter-wrapper approach is proposed.

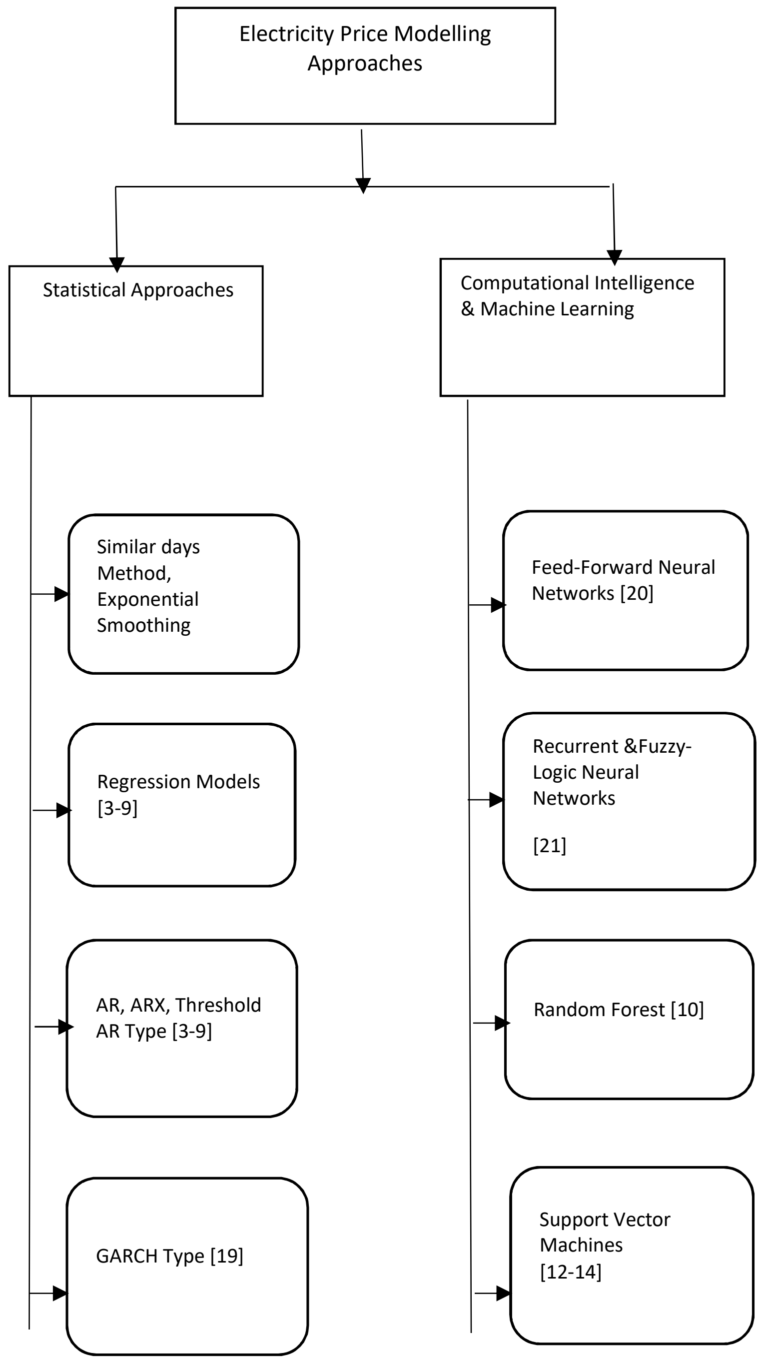

Figure 3: Taxonomy of Electricity Price Modelling Approach

3 MODELLING OF ELECTRICITY PRICE

The Electricity price forecast process involves multiple stages that include data cleansing, data preparation, and data evaluation. The following are steps involved:

Step 1- (Gather Load data): We collected the load consumption data through the web-link provided by the Iberian Electricity market. We had collected all the data into a single file for easy computation of the price.

Step 2 – Glean and order the data: We used the three & six months of the dataset to build the model and evaluate how well the model generalizes to future results.

Step 3 – Training a model on the data

To model the relationship between the predictor variables used in modeling and the electricity price, we used several Statistical and Machine Learning packages in Open source R software which provides a standard and easy-to-use implementation of such models.In this study, we have used four techniques. The following R packages were used in implementing the model. Auto-arima' function in forecast package in R helps us identify the best fit ARIMA model. 'Ksvm' function in 'kernlab' package in R helps us to fit the SVM model. 'randomForest' function in randomForest package in R helps us to fit the RF model. 'glm' function under guassian family in 'stats' package in R helps us to fit the GLM model.

Step 4 – Evaluating model performance: We must measure the correlation between our predicted electricity price and the true value. This correlation values help us in evaluating the model and helps us to find the direct relationship between the two variables.

Correlations close to 1 indicate strong linear relationships between actual and forecasted price. Therefore, the correlation of more than 0.9 for 14 variables for two different datasets shown in table 2 indicates a fairly strong relationship. These values show that our model is fitting well to predict the future values.

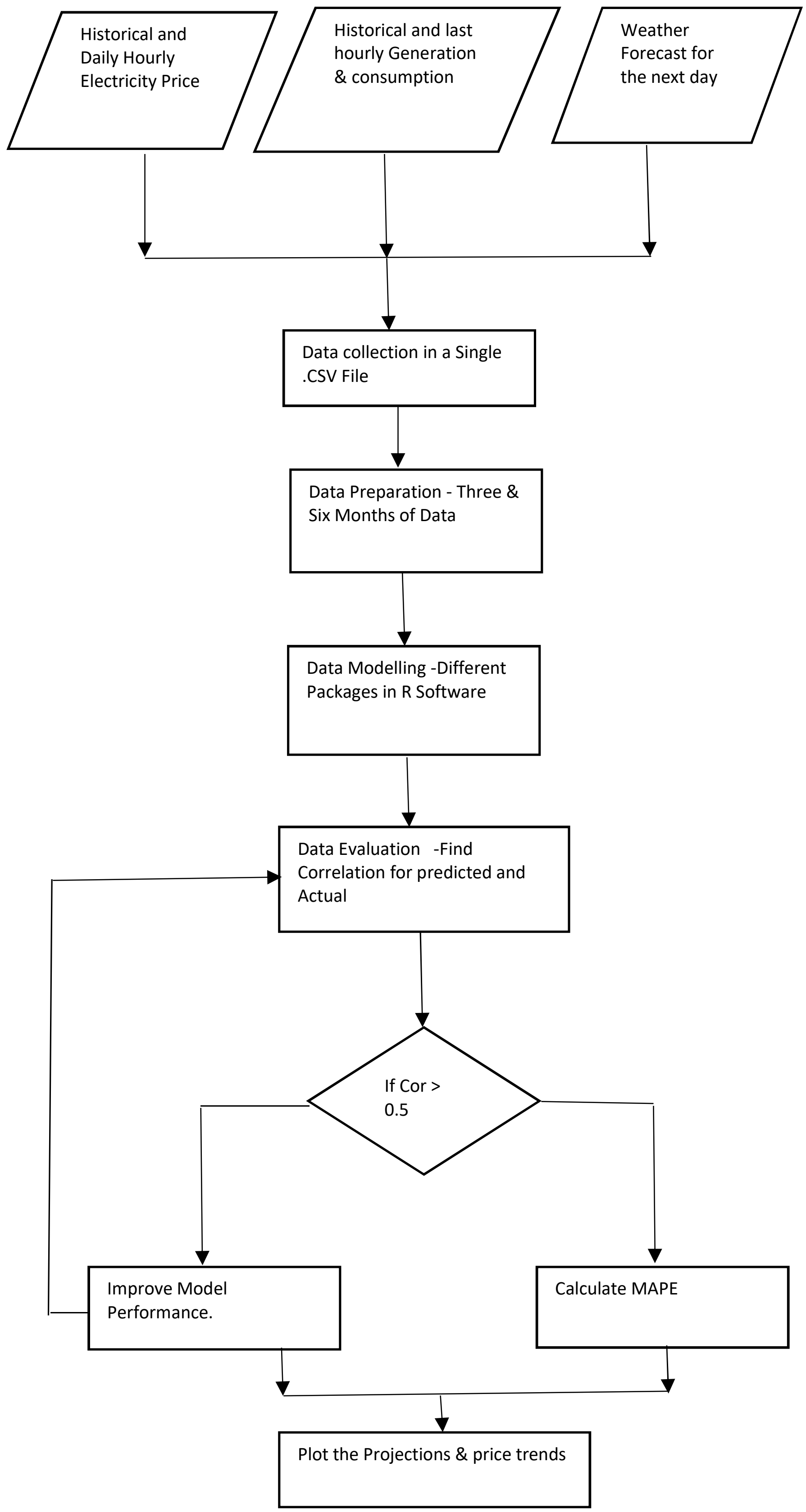

Figure 4: Flowchart showing modelling process of Electricity Price Forecasting

3.1 ARIMA

ARIMA method is a stochastic process used to analyze time series data. ARIMA is a mixture of three-time series components i.e. AR (Autoregressive), I (Integrated), and MA (Moving Average). A convenient notation for ARIMA model is ARIMA (p,d,q) [15] Here p, d, and q represents AR, I, and MA components. Each of these components is used to reduce the final residuals display white noise or no residuals at all.

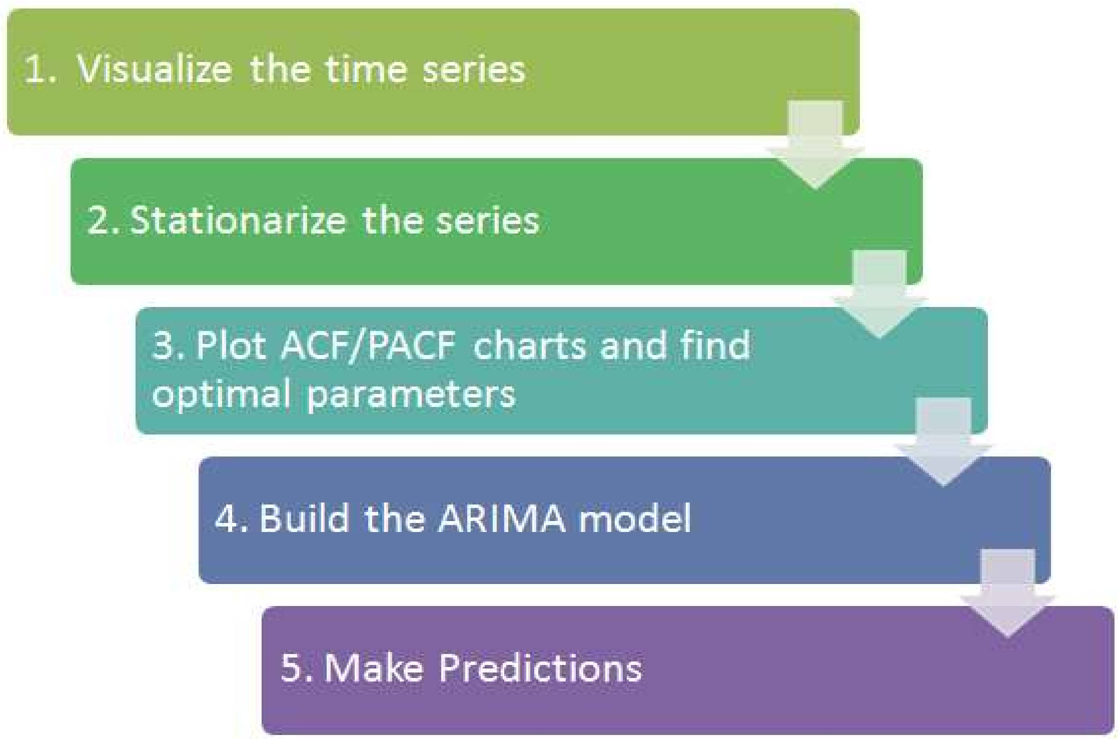

Figure 5: ARIMA flowchart[24]

1st step of ARIMA to extract Information:

Integrated (I) – subtracting the data from the previous or lagged one to extract trends to make it stationery [15]. This step is basically used to extract trend from the original time series data. Differencing is one of the most popularly used methods for extraction of trends. Here, the original series is subtracted from its lagged or previous.

The residues of most time series data become trendless after differencing for the first time which is represented as ARIMA (0, 1, 0). If the time series data has trends still, it is further differenced to remove the trend which is denoted as ARIMA (0, 2, 0). This is called 2nd order differencing. This trend-less series is called as stationery on mean series. This shows that the mean does not change over time.

2nd step of ARIMA to extract Information:

Auto Regressive (AR) – uses the previous value influence on the current value. After we difference the data to make it stationery, then the AR component of the ARIMA starts. As we mentioned earlier, it takes the previous value influence on the current values. This is done through obtaining a simple multiple linear regression .model with the independent or predictor variables as time lagged values. The general notation of the equation for this multiple linear regression is shown below. Here Y_t presents the price at time't', ϕ_1 denotes regression co-efficient and e_t denotes error term.

$$Y_t = c + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \ldots.\phi_p Y_{t-p} + e_t \tag{1}$$

AR model of order 1 i.e. p=1 or ARIMA (1, 0, 0) is denoted with the following regression equation

$$Y_t = c + \phi_1 Y_{t-1} + e_t \tag{2}$$

3rd step of ARIMA to Extract error terms

Moving Average (MA) – uses the previous value error term influence on the current value error. After we take the auto-regression is performed, here we form the relationship between the error term of the previous and current values as shown in the equation (3).This component of the ARIMA is formed with the simple multiple linear regression with the predictor variables as lagged error terms.

$$Y_t = c + e_t + \theta_1 e_{t-1} + \theta_2 e_{t-2} + ...\theta_q e_{t-q} \tag{3}$$

ARIMA Implementation:

Step 1: Identification of best fit ARIMA model

Auto-arima function in R under the forecast package help us in finding the best fit for the ARIMA model. It gives the best fit by giving the value of the three components (p,d,q) which we can use it for prediction. The best fit model obtained from the Auto-arima function is based on the lowest values of the Akaike Information criterion (AIC) and Bayesian Information Criterion (BIC).

Step 2: Forecast using the best fit ARIMA model

The next day hourly spot prices are forecasted using the Function Forecast in R. After finding the ARIMA model from the auto.arima function, arima function is used to predict the price using the given set of variables.

3.2 Support Vector Machines (SVM)

Support Vector Machines (Support Vector network) are supervised learning model that analyses data for regression analysis in this work. SVM is assumed to be a surface that has a boundary between numerous points in a data that represents example plotted in a multidimensional space [16]. The main aim of the SVM is to create a flat boundary that leads to the equal partition of data on both sides. This boundary helps in creating SVM to model complex relationship.

The mathematics behind the SVM has been there for a long time but it has become extremely popular due to the availability of these algorithm in various software. These algorithms were well supported in open source software like R that is implemented in libraries. Availability of these packages in open source software has increased the usage of this algorithm which is otherwise quite complex to implement. SVM can be used for both Classification and Prediction. In this case, we use this algorithm for predicting the prices.

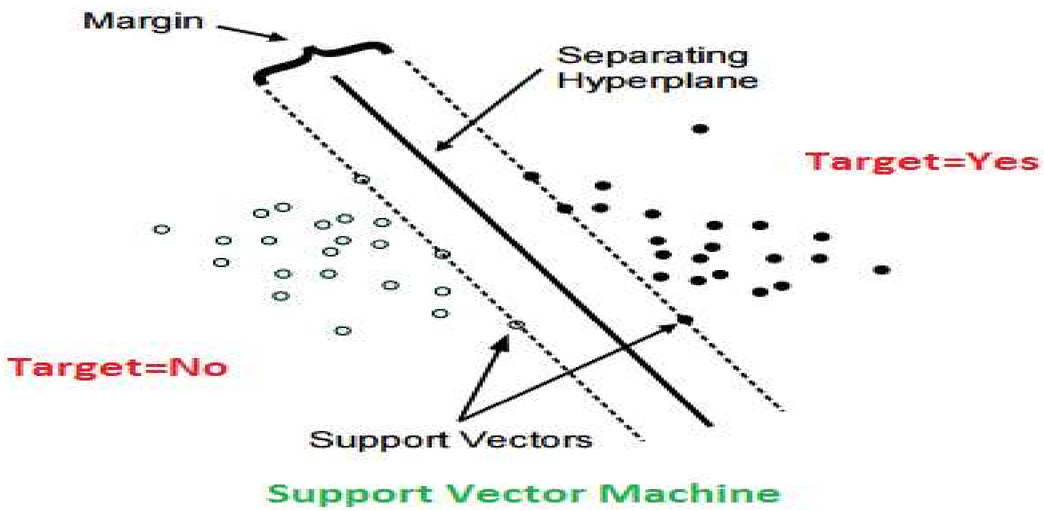

Figure 6: Support Vector Machine[25]

3.3 Random forest (RF)

Ensemble-based method called random forests (or decision tree forests) emphasis only on ensembles of decision trees [16]. This method combines the base principles of bagging with random feature selection to add additional variation to the decision tree models. The model uses a vote to combines the trees prediction after the ensembles of trees or forest is generated.

Random forest brings both versatility and power to this machine learning approach. Because the ensembles use the small portion of the larger dataset, it is extremely effective in handling the large dataset which might cause other methods to fail because of the dimensionality problems. Also, the error rate for the learning tasks is on par or equal to other machine learning approaches.

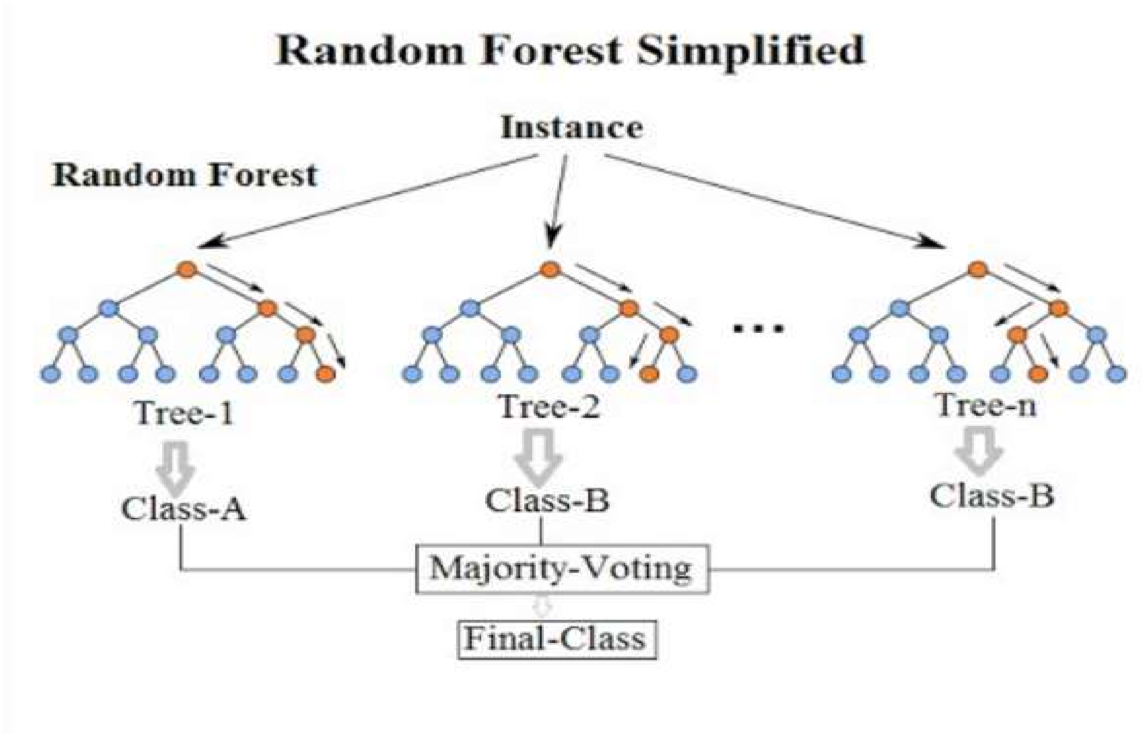

Figure 7: Random forest[26]

3.4 Generalized Linear Model (GLM)

In statistics, the Generalized linear model is a simplified generalization of the normal linear regression that allows the response variable to have an error distribution model rather than normal distribution [17]. It generalizes the simple linear regression by allowing the model to be related to the response variable through a link function. This is achieved by allowing the variance of each sample to be a function of its forecasted value. In this model, each outcome Y of the dependent variable is assumed to be generated from a family of probability distribution that includes the normal, binomial, Poisson, and gamma distribution. The mean depends upon on the independent variables, X through

$$E(Y) = \mu = g^{-1}(X\beta) \tag{4}$$

Where E(Y) is the expected value of Y; Xβ is the linear predictor, a linear combination of unknown parameters β; g is the link function.

$$\mathrm{Var}(Y) = V(\mu) = V(g^{-1}(X\beta)) \tag{5}$$

In this framework, the variance is typically a function, V, of the mean: It is convenient if V follows from the exponential family distribution, but it may simply be that the variance is a function of the predicted Value. The unknown parameters, β are typically computed with likelihood, maximum quasi-likelihood, or Bayesian techniques.

3.5 Proposed Hybrid 2-Stage Model

The following flowchart in Fig. 8 shows that the variable selection is carried out by ARIMA deployed in Stage-1, and residuals are then computed before Stage-2 begins. In Stage-2, residuals are fed as input to the collection of other forecasting methods.

The two-step residual extraction method has been briefly reported using ARIMA with GLM in our previous paper [27] and will not be repeated here. In the present paper, we are applying the same process to ARIMA-GLM, ARIMA-SVM, ARIMA-RF, and ARIMA-LOWESS. Some details on the proposed two-stage model are provided next.

3.5.1 Stage-1: Initial Price forecast (F) using ARIMA:

Step 1. In Stage-1, ARIMA was used to predict the day-ahead prices. Input variables that are considered include historical electricity prices, generation and consumption load and weather data like solar irradiance, temperature and wind speed. These variables are fed as time-series data to the ARIMA model. The relationship between the predictor variables and forecasted variables is then initialized through this model.

Step 2. An 'auto-arima' function built in R-software was used to identify the best-fit by inputting the residual values (p, d, q) of the three-time series components I, AR, and MA. After identifying the best-fit model, the 'forecast' function is used to predict the day-ahead price.

Step 3. The same process is repeated for other datasets. In this study, one week, two weeks, three weeks, one month, 45 days, 60 days, and 75 days of datasets from the Iberian electricity price market are used to predict the day-ahead electricity prices.

Step 4. After the price predictions, residuals are calculated by differencing the predicted value (f) from the actual value (A).

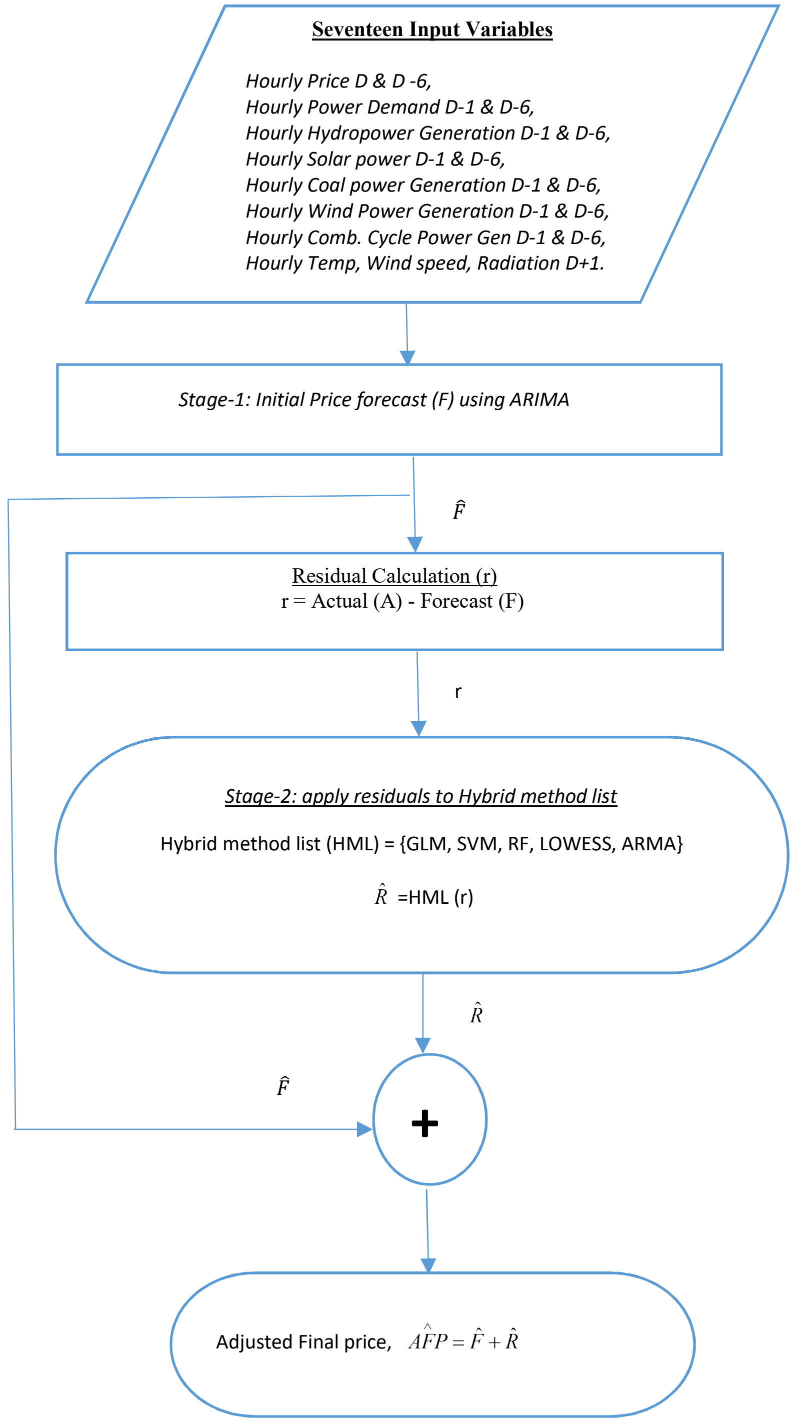

Figure 8: Flowchart of the proposed two-stage approach

3.5.2 Stage-2: Input residuals to the Hybrid Model

3.5.2.1 ARIMA-SVM

The steps involved in the two-stage residual extraction method, that uses combinations of ARIMA and SVM, are as follows:

Step 1. In Stage-2, the residual dataset is fed as an input to the SVM model. SVM model is then initialized by calling the function '*ksvm*' which available in the kernlab package. The SVM model is then used to predict the residual for the next day by calling the '*predict*' function.

Step 2. Finally, the calculated residual (R) from Step 1 is then added to the predicted price from the ARIMA method (P) to get the final price.

3.5.2.2 ARIMA-RF

The following are the steps in deploying the hybrid combination of ARIMA and RF methods to forecast the next day-ahead price:

Step 1. In Stage- 2 of the hybrid model, the residuals from the ARIMA model are fed as time series input data to the RF model. The *'random forest'* function in the Random Forest package of R helps in fitting the RF model.

Step 2. The RF model is then used to predict the future residuals (R) which are added to the earlier predictions to obtain the adjusted final price forecast.

3.5.2.3 ARIMA-LOWESS

The following are the steps in deploying the hybrid combination of ARIMA and LOWESS methods to forecast the next day-ahead prices:

Step 1. In the Stage-2 of the hybrid model, the residual dataset from the ARIMA model is fed as time series input data to the LOWESS model. The *'loess'* function in the Stats package of R helps in fitting the LOWESS model.

Step 2. The loess model is then used to predict the future residual (R) which is added along with the predicted price to get the final price forecast.

3.5.2.4 ARIMA-ARIMA

The following are the steps in deploying the combination of ARIMA and ARIMA to forecast the next day-ahead price:

Step 1. In Stage-2 of the hybrid model, the residuals from the ARIMA model are fed as input data to the same ARIMA model. The *'auto-arima'* function in the Stats package of R helps in fitting the ARIMA model in Stage-2.

Step 2. The ARIMA model is then used to predict the future residual (R) which is added along with the predicted price from the ARIMA method in the first stage to get the final price forecast.

3.6 Explanatory Variables for Day-Ahead Price Forecast

3.6.1 Data Explanation

The day-ahead electricity prices are greatly influenced by several explanatory variables [28] as shown in TABLE 1. They are as follows:

(a) Hourly electricity price for day D and day D-6.

(b) Hourly load data, including total load demand, hydro power demand, solar power demand, coal power demand, wind power demand, and combined cycle power demand for day D and day D-6.

(c) Hourly weather data, including temperature, wind speed, and solar irradiance.

All the fundamental price variables are taken into consideration for all the hybrid methods. The day-ahead price forecast results for different durations are discussed next

Table 1: Seventeen decision variables

Variable No.	**Description**
1,2	Hourly Price D , Hourly Price D-6
3,4	Hourly Power Demand D-1 & D-6
5,6	Hourly Hydropower Generation D-1 & D-6
7,8	Hourly Solar Power D-1 & D-6
9,10	Hourly Coal Power Generation D-1 & D-6
11,12	Hourly Wind Power Generation D-1 & D-6
13 ,14	Hourly Combined Cycle Power Generation D-1 & D-6
15,16,17	Hourly Temperature, Wind speed, Radiation D+1

3.7 Results and discussion

As discussed above, several two-stage hybrid models have been used to predict the electricity prices of the Iberian Markets in this study. The hybrid models include ARIMA-GLM, ARIMA-RF, ARIMA-SVM, GLM, and ARIMA-LOWESS. The hybrid models are trained and tested using datasets ranging from one-week to three months.

The dataset durations include one-week, two-weeks, three-weeks, one month, 45 days, 60 days, 75 days, and 90 days. The specific data durations are shown in Figs. 9 and 10. We evaluate the performance of our forecast models through a statistical measure known as MAPE (Mean Average Percentage Error) which represents the daily error in price predictions.

$$MAPE_{day} = \frac{1}{24}\sum_{i=1}^{24}\frac{p^{actual} - p^{pred}}{p^{actual}} \tag{6}$$

where $MAPE_{day}$ is the daily error. Table II shows a numerical comparison of the MAPE values for various data durations. Figs. 9 to 22 graphically show the MAPE comparison of hybrid models. Each figure shows the MAPE comparison of ARIMA, ARIMA-GLM, ARIMA-RF, and ARIMA-SVM. All the variables have been taken into consideration.

However, in Figs. 23 and 24 only four variables are considered since LOWESS can be modeled only with a maximum of four variables. In the last dataset (90 days), all methods use the following specific four variables: *Price, Price D-6, Power demand D-1, and Power demand D-6*.

Table 2: Comparison of MAPE values for different duration

MAPE	**ARIMA**	**ARIMA-GLM**	**ARIMA-SVM**	**ARIMA- RF**
Short-term price forecast (day-ahead)				
$MAPE_{1week}$	5.36	5.00	3.73	5.24
$MAPE_{2weeks}$	4.23	4.43	3.98	4.01
$MAPE_{3weeks}$	4.07	4.14	3.64	3.69
$MAPE_{1month}$	5.64	5.54	5.05	5.44
$MAPE_{45days}$	2.7	2.54	2.49	2.38
$MAPE_{60days}$	1.99	1.92	2.037	2.027
$MAPE_{75days}$	1.99	1.92	2.009	2.2263

Figs. 9 and 10 indicate that ARIMA-SVM combination outperforms other methods using a one-week dataset. In addition, from Table II it is also evident that the ARIMA-SVM model gives a better prediction with lesser durations of data such as for one week, two weeks, three weeks, and one month.

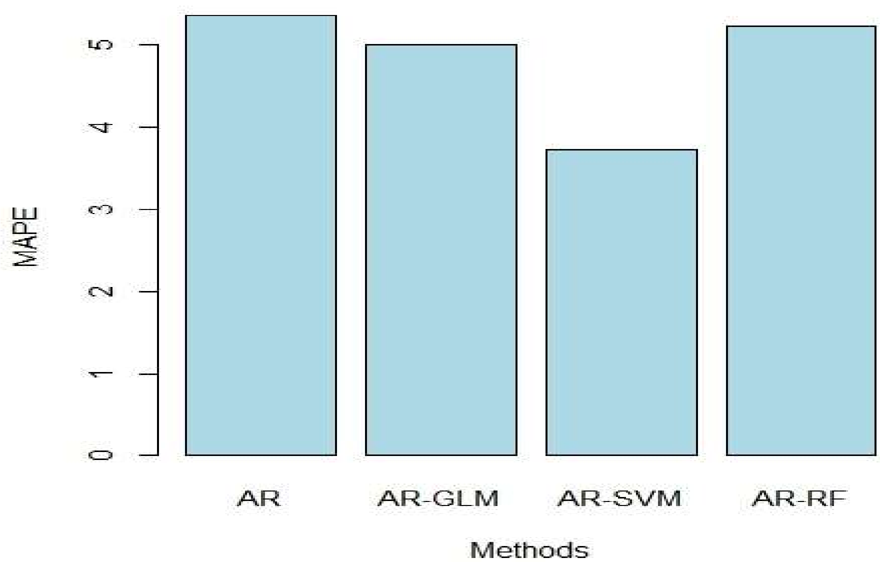

Figure 9: Comparison of MAPE for one week (July 24, 2015 to July 30, 2015) to predict day-ahead price (July 31, 2015).

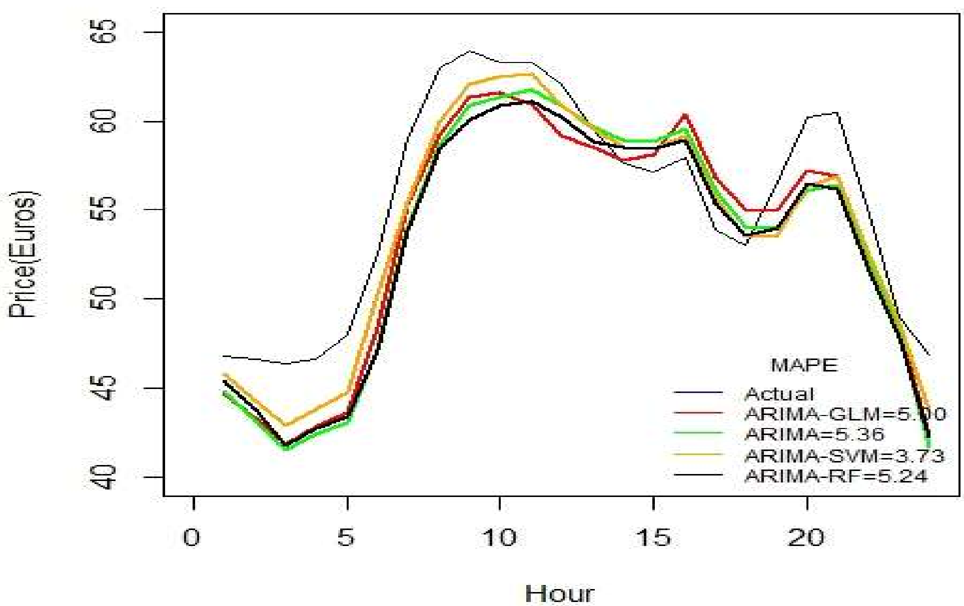

Figure 10: Comparison of MAPE for one week (July 24, 2015 to July 30, 2015) to predict day-ahead price (July 31, 2015).

From Figs. 11 and 12, MAPE errors for two weeks are reduced, but not substantially. The goal here is to test multiple durations of the datasets and observe how MAPE changes with the duration of datasets.

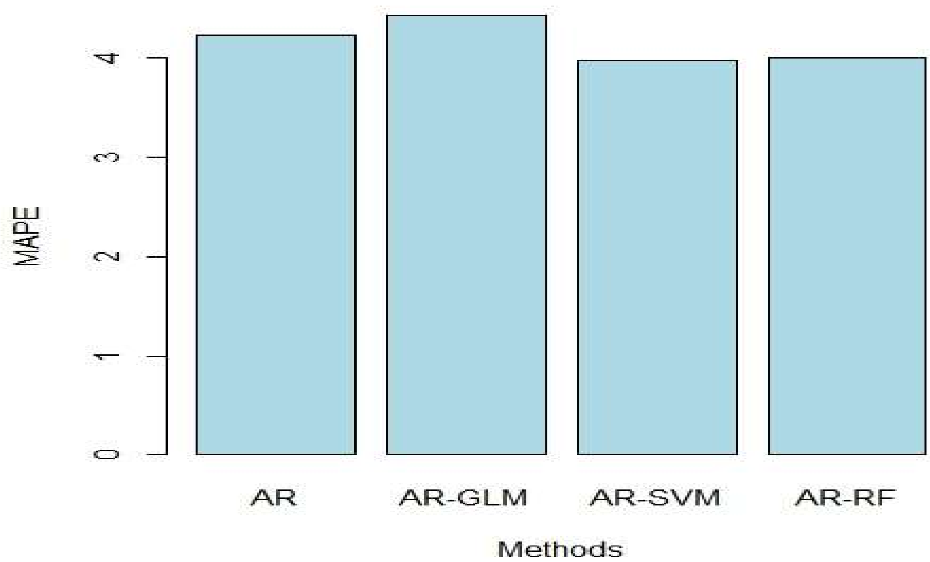

Figure 11: Comparison of MAPE for two weeks (July 17, 2015 to July 30, 2015) to predict day-ahead price (July 31, 2015)

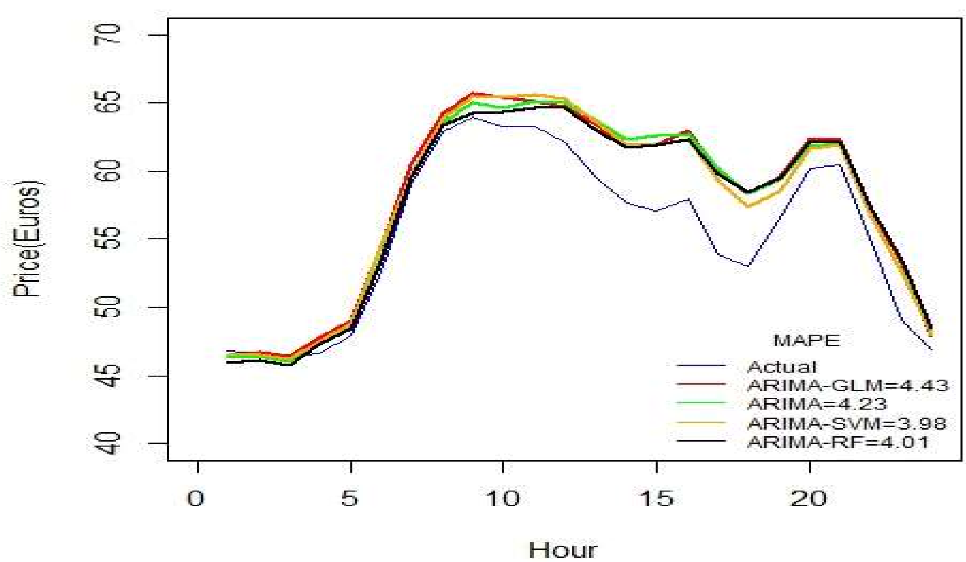

Figure 12: Comparison of MAPE for two weeks (July 17, 2015 to July 30, 2015) to predict day-ahead price (July 31, 2015)

From Figs.13 and 14, it is yet again evident that the combination of ARIMA-SVM performs better than other methods with 17 variables.

From above, it is noted that the MAPE values are reduced as the duration of the datasets is increased. Using 17 variables in the above case studies, there seems to be a linear reduction in MAPE values starting from one to three weeks. Another important inference from these results is that there is a sharp reduction in MAPE for ARIMA-RF combination than other methods. This is a strong evidence that the random forest is well-suited for larger datasets.

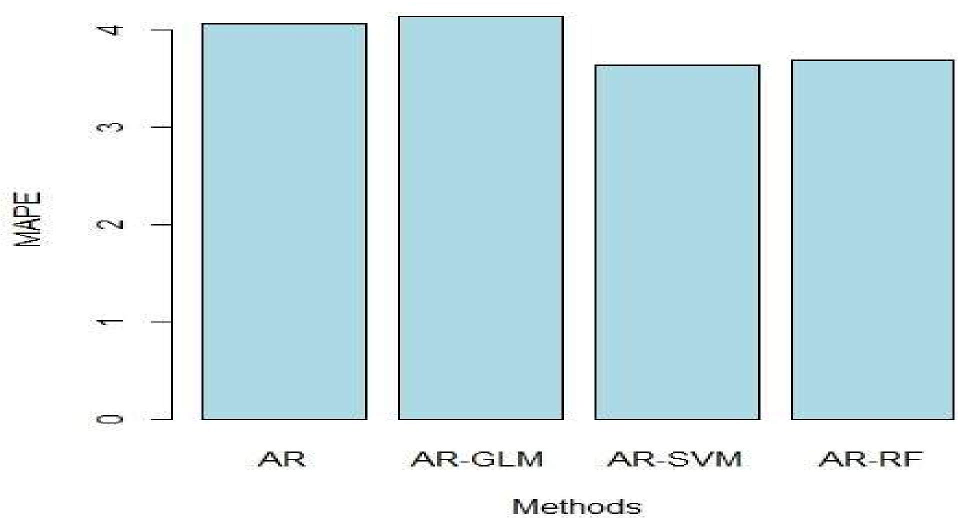

Figure 13: Comparison of MAPE for three weeks (July 10, 2015 to July 30, 2015) to predict day-ahead price (July 31, 2015).

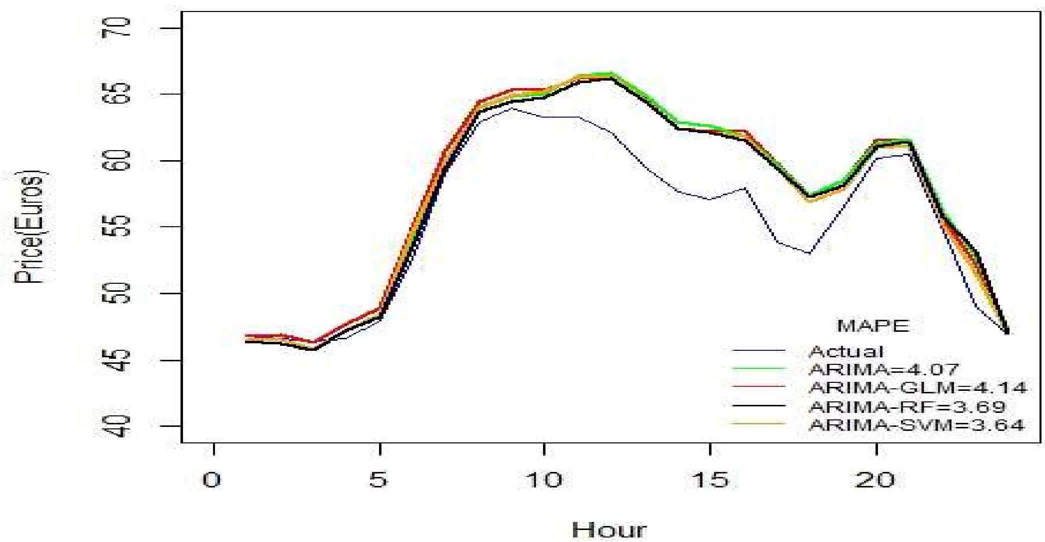

Figure 14: Comparison of MAPE for three weeks (July 10, 2015 to July 30, 2015) to predict day-ahead price (July 31, 2015).

For the 1-month duration of data seen in Figs. 15 and 16, error values seem to be increase compared to the 3-week data set. This may be due to some irrelevancy or missing fields in data. This might also due to the fact that price variable is not highly correlated with the predictor variables.

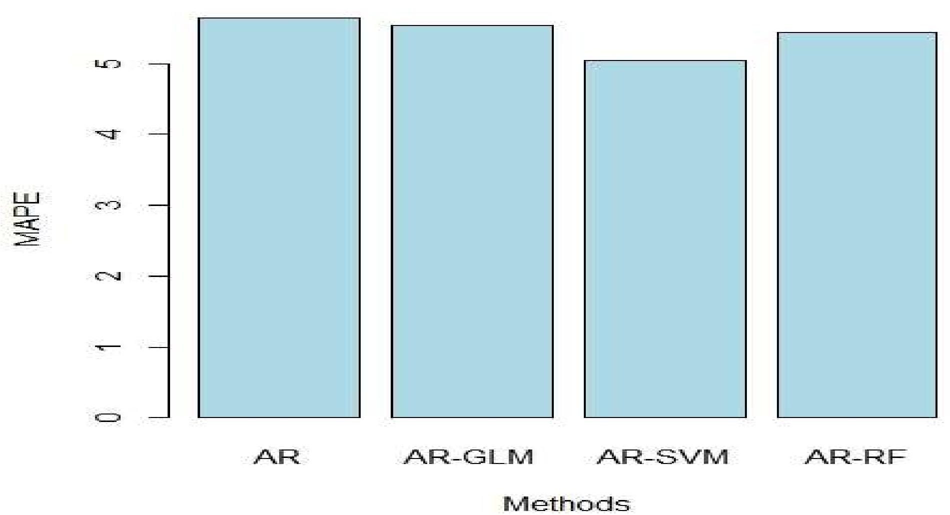

Figure 15: Comparison of MAPE for one month (July 01, 2015 to July 30, 2015) to predict day-ahead price (July 31, 2015).

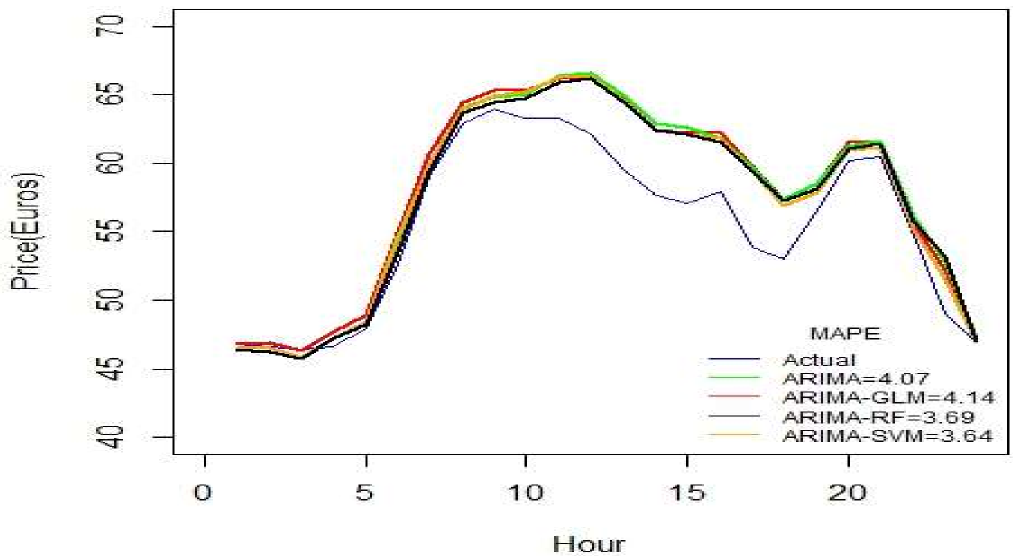

Figure 16: Comparison of MAPE for one month (July 01, 2015 to July-30 2015) to predict day-ahead price (July 31, 2015.

From Figs 17 and 18, the accuracy of the calculated values has improved substantially considering all the seventeen variables. If one includes the important variables such as price D and price D-6, one can then greatly reduce the forecasting error. Here, ARIMA-RF is considered effective for larger datasets, because the ensembles take a small portion of the dataset.

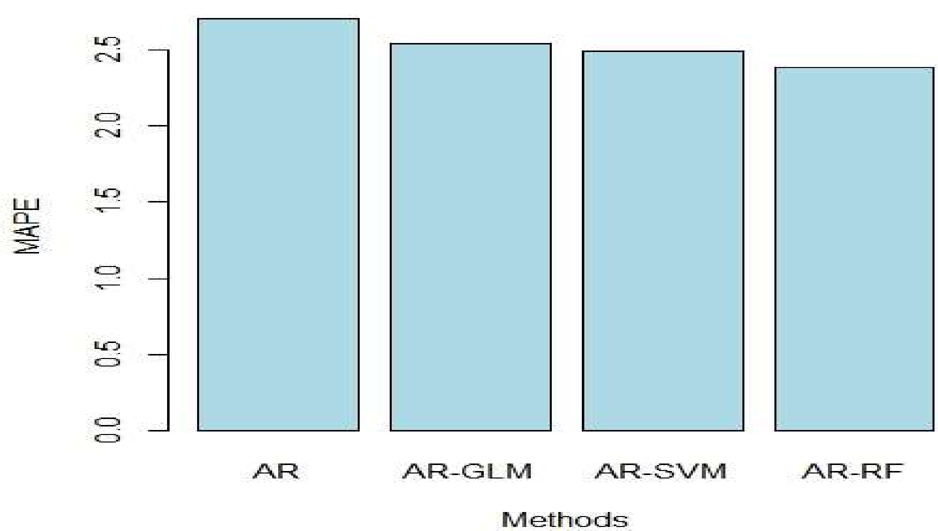

Figure 17: Comparison of MAPE for 45 days (June 16, 2015 to July 30, 2015) to predict day-ahead price (July 31, 2015)

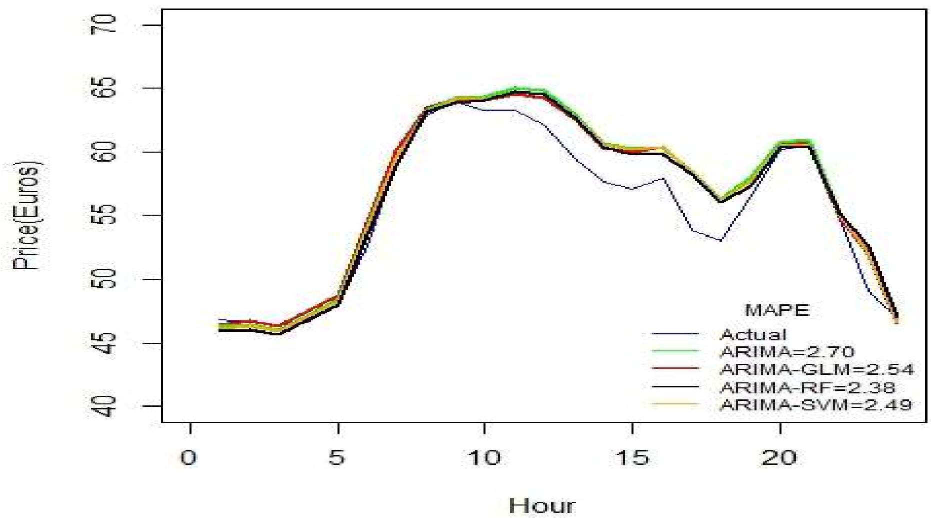

Figure 18: Comparison of MAPE for 45 days (June 16, 2015 to July 30, 2015) to predict day-ahead price (July 31, 2015).

As seen from Figs. 19 and 20, these hybrid models work better for durations greater than 45 days. All the proposed hybrid combinations closely predict the pattern of price-spikes, while matching with the actual data.

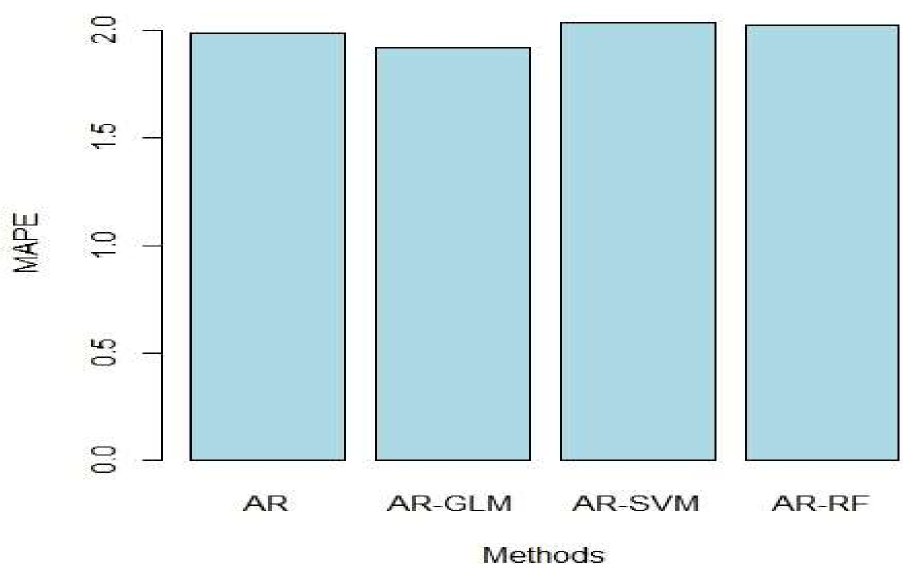

Figure 19: Comparison of MAPE for 60 days (June 01, 2015 to July 30, 2015) to predict day-ahead price (July 31, 2015).

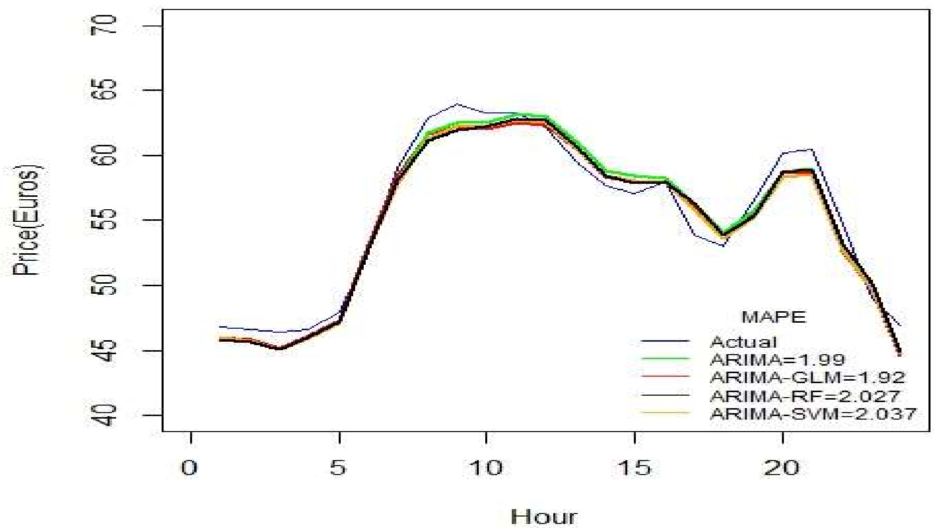

Figure 20: Comparison of MAPE for 60 days (June 01, 2015 to July 30, 2015) to predict day-ahead price (July 31, 2015)

A similar conclusion can be inferred from Figs. 21and 22, as the duration of datasets increases.

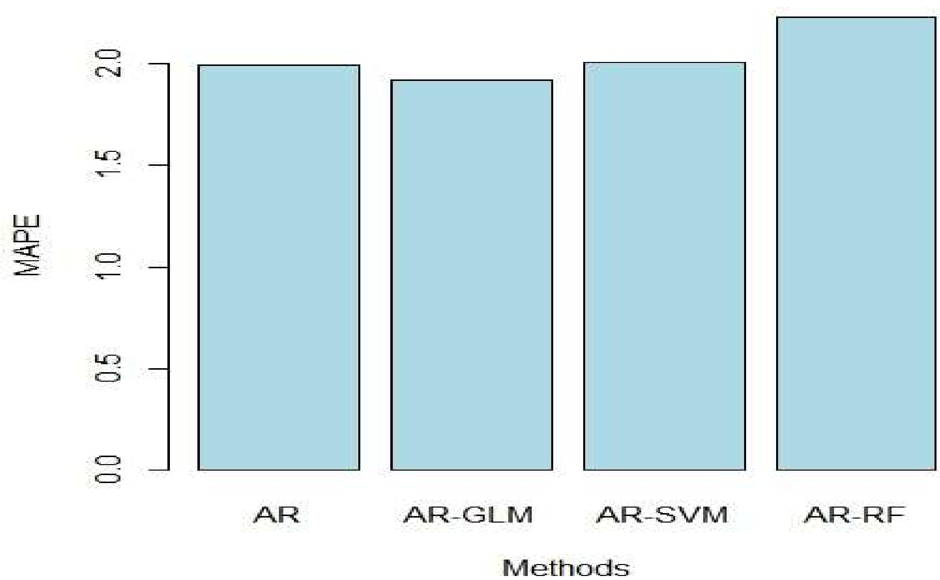

Figure 21: Comparison of MAPE for 75 days (May 17, 2015 to July 30, 2015) to predict day-ahead price (July 31, 2015).

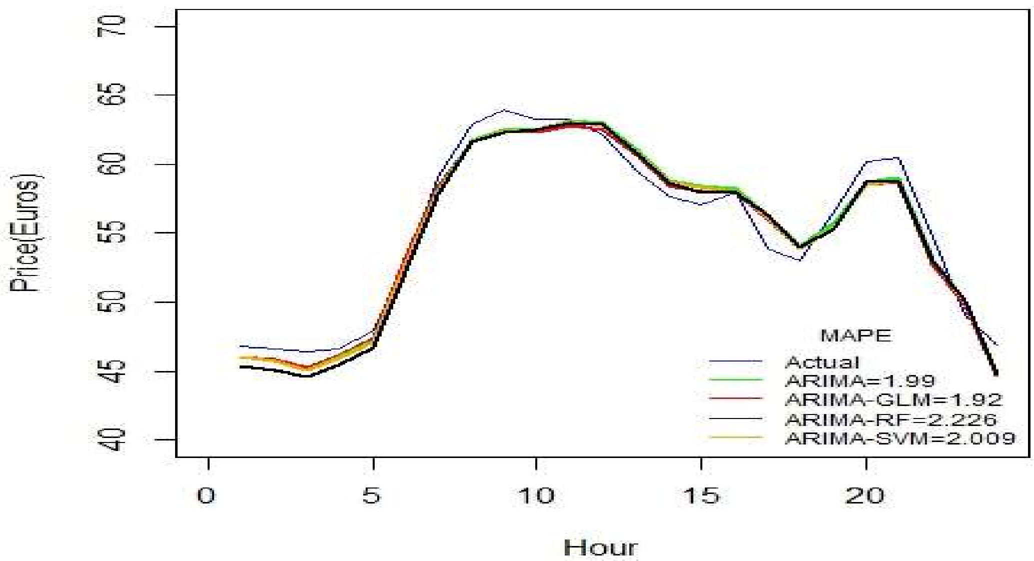

Figure 22: Comparison of MAPE for 75 days (May 17, 2015 to July 30, 2015) to predict day-ahead price (July 31, 2015).

Table 3: Comparison of day-ahead forecasting performance of several hybrid models for 90 days of dataset using 4 variables (hourly price D, hourly price D -6, hourly power demand D-1 & D-6)

Parameter	ARIMA	ARIMA -GLM	ARIMA - SVM	ARIMA LOWESS	ARIMA- RF
$MAPE_{90days}$	2.80	2.59	2.73	2.66	3.12

Table 3 shows the MAPE results for the 90 days dataset using four variables (Hourly Price D, Hourly Price D -6, Hourly Power Demand D-1 and D-6). Since LOWESS cannot be used with more than four variables, the ARIMA-LOWESS model is compared with the other hybrid models also using the same four variables.

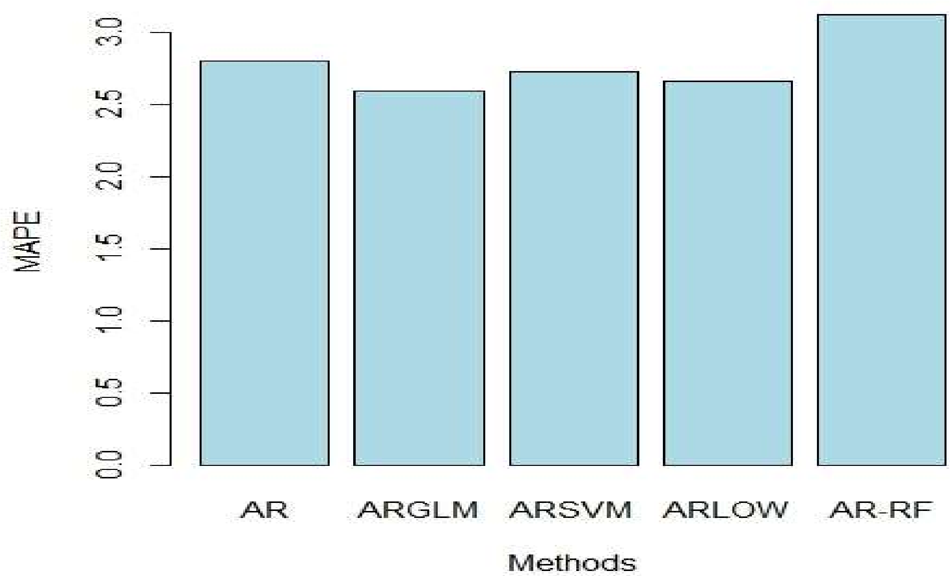

Figure 23: Comparison of MAPE for 90 days (May 01, 2015 to July 30 2015) to predict day-ahead price (July 31, 2015).

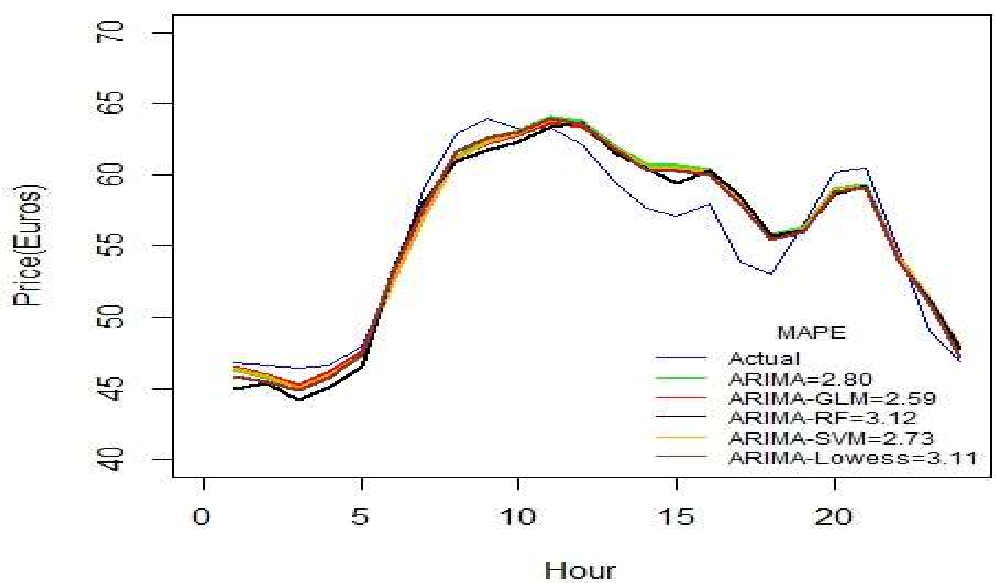

Figure 24: Comparison of MAPE for 90 days (May 01, 2015 to July 30, 2015) to predict day-ahead price (July 31, 2015)

Fig. 25 compares the MAPE values from one-week to 75 days. From Fig. 18, it is concluded that the models may need to be tested with additional data durations for scalability. For such models, variables such as price, load and temperature values have been considered. The MAPE error can be significantly reduced by considering only those important variables that highly correlate with the price.

The electricity market has to be studied thoroughly to consider which variable significantly impacts the electricity price. The larger penetration of renewable energy sources such as wind and solar resources into the grid might impact the price significantly. The weekday and weekend patterns were also studied by the authors and the results are summarized in Tables 4 and 5.

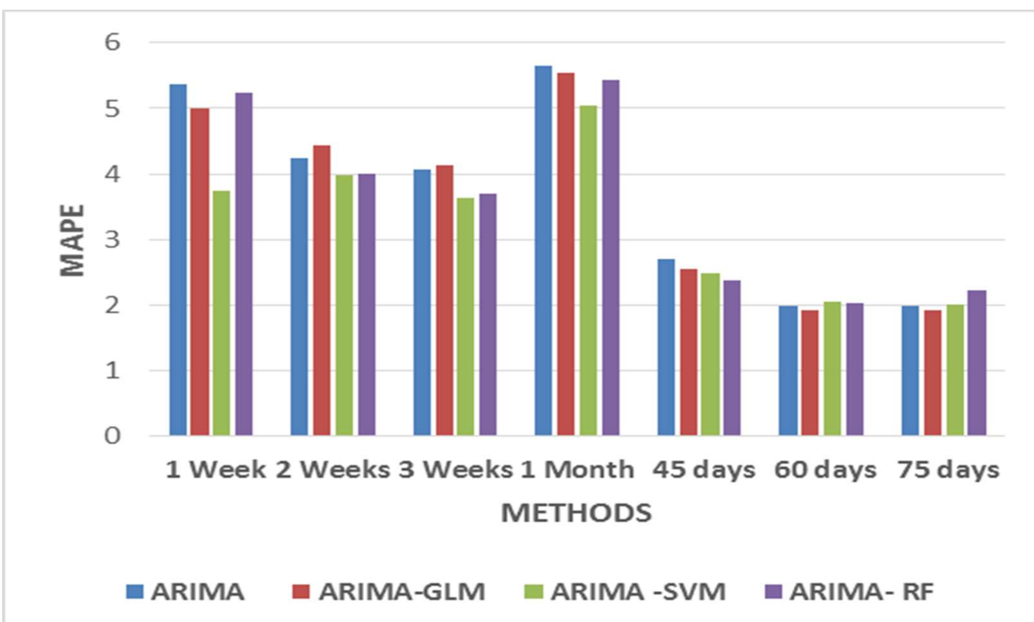

Figure 25: Comparison of MAPE for all dataset from one week to 90 days to predict day-ahead price (July 31, 2015)

Table 4: Comparison of day-ahead forecasting performance of several hybrid models for weekday dataset using 17 variables.

Parameter	ARIMA	ARIMA-GLM	ARIMA -SVM	ARIMA- RF
$MAPE_{1month}$	8.16	8.30	7.41	7.01
$MAPE_{2months}$	1.81	1.86	1.84	2.33
$MAPE_{3months}$	3.58	3.83	3.82	4.72
$MAPE_{6months}$	4.48	4.54	4.62	5.78

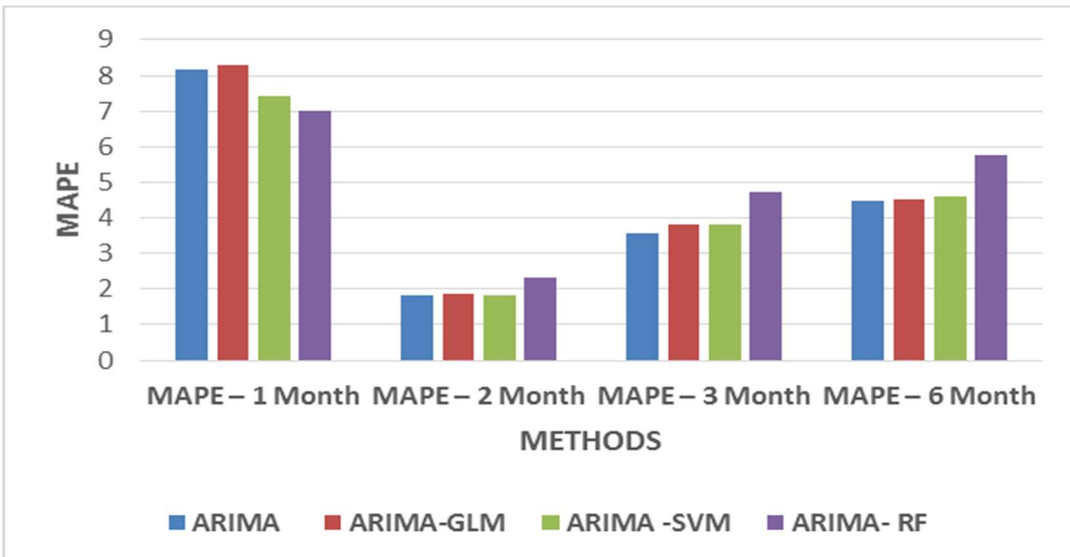

Figure 26: Comparison of MAPE for 1, 2, 3, and 6-month weekdays dataset to predict day-ahead price (July 31, 2015)

From Table-4, one infers that two months of weekday datasets give a better prediction, as this dataset highly correlates with the predicted price. For weekend datasets, only 10 variables are considered to be of importance.

The variables that take previous day’s influence into consideration were removed from the datasets. Thus taking into consideration only variables such as hourly price D-6, hourly power demand D-6, hourly hydropower generation D-6, hourly solar power D-6, hourly coal power generation D-6, hourly wind power generation D-6, hourly combined cycle power generation D-6, temp, wind speed, radiation D+1, etc.

Table 5: Comparison of day-ahead forecasting performance of several hybrid models for weekend dataset using 10 variables

Parameter	ARIMA	ARIMA-GLM	ARIMA-SVM	ARIMA- RF
$MAPE_{1month}$	13.07	12.4	12.01	13.7
$MAPE_{2months}$	9.94	9.15	9.26	9.52
$MAPE_{3months}$	9.73	9.22	9.15	9.19
$MAPE_{6months}$	9.91	9.63	9.53	9.88

Figs. 26 and 27 show MAPE values for weekday and weekend datasets. The results do not significantly improve the MAPE values, but they certainly indicate that the models may require additional data to identify patterns for better forecasts.

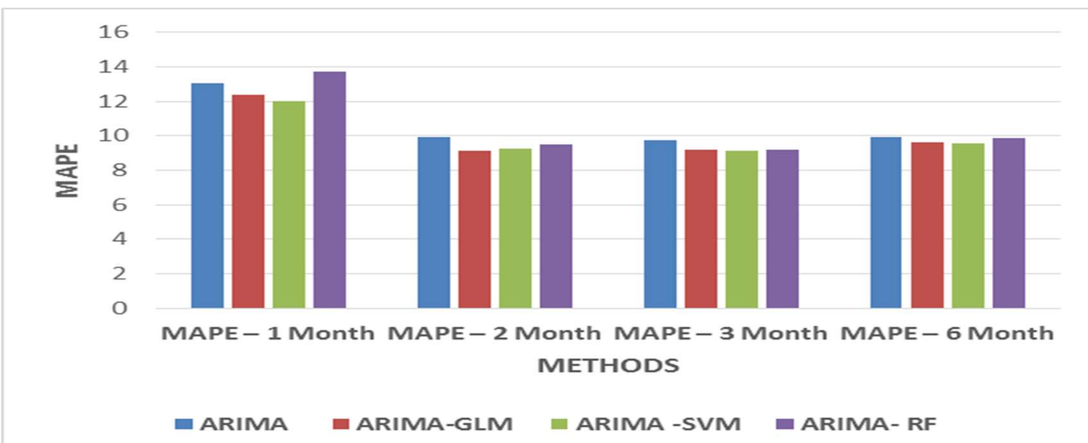

Figure 27: Comparison of MAPE for 1, 2, 3, and 6-month weekend dataset to predict day-ahead price (July 26, 2015)

Table 6, shows the MAPE results for the two-stage ARIMA models with and without explanatory variables in the Stage-2. From these results, one can clearly infer that the inclusion of the explanatory variables in Stage-2 has great influence on the residual predictions

Table 6: Comparison of MAPE results for two-stage ARIMA model with/ without explanatory variables in stage-2

MAPE	ARIMA	ARIMA-ARIMA (with explanatory variables in Stage-2)	ARIMA-ARIMA (without explanatory variables in Stage-2)
$MAPE_{1week}$	5.36	4.66	5.34
$MAPE_{2weeks}$	4.23	4.44	3.79
$MAPE_{3weeks}$	4.07	4.14	4.02
$MAPE_{1month}$	5.64	5.54	5.65
$MAPE_{45days}$	2.7	2.54	2.73
$MAPE_{60days}$	1.99	1.78	1.91
$MAPE_{75days}$	1.99	1.84	1.98

Table 7: Comparison of MAPE results for Iberian electricity market with published literature

Methods	**MAPE**
Mixed Model [29]	14.90
ARIMA with 2 Variables [1]	13.39
Neural Network [30]	11.40
Weighted Nearest Neighbor [31]	10.89
Wavelet- ARIMA with 4 Variables [32]	10.70
Fuzzy Neural Network [33]	9.84
Adaptive Wavelet Neural Network with 2 variables [34]	9.64
Neural network Wavelet Transform with 1 variable [35]	9.5
WNF with 1 variable [36]	9.47
Elman Network [37]	9.09
Hybrid Intelligent systems with 3 Variables	7.47
Wavelet –ARIMA-RBFN	6.76
Hybrid wavelet-PSO-ANFIS [38]	6.50

Cascaded Neuro- evolutionary Algorithm with 2 variables[39]	5.79

Table 7 presents and compares the MAPE results of the Iberian electricity market as published in the literature. This comparative table clearly strengthens the fact that the ARIMA–based two-stage model is a promising forecasting method to improve the accuracies in residual training for short term price forecasting.

4 DEEP LEARNING FOR ELECTRICITY PRICE FORECASTING

Deep learning [40] is a subfield of machine learning based on the algorithms greatly inspired by the structure and function of the human brain called artificial neural network. Deep generally refers to a large number of the layers. One of the important benefits of deep learning is its scalability. The results get better with more data and larger models, which in turn requires computation to train. The performance of the deep learning continues to increase while other machine learning algorithms reach a plateau in performance. In addition to its scalability, deep learning has the ability to extract features from raw data also called feature learning.

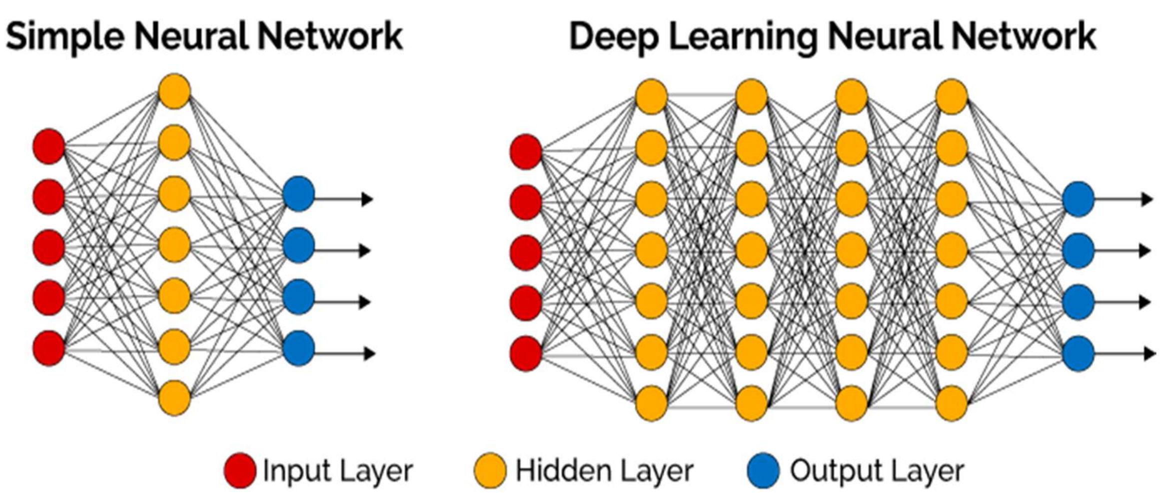

Figure 28. Comparison of deep learning with simple neural network[41]

Geoffrey Hinton is a pioneer in the field of artificial neural networks and co-published a first paper on the popular backpropagation algorithm. He started the introduction of the phrasing “deep” to describe the large artificial neural network. The Recent increase in the computing power and the accessibility to the larger datasets has unleashed the untapped capability of the artificial neural networks when used at a very large scale.

“Deep learning allows computational models that are composed of multiple processing layers to learn representations of data with multiple levels of abstraction”.

The most popular deep learning techniques are:

- Multilayer Perceptron Networks.
- Convolutional Neural Networks.
- Long Short-Term Memory Recurrent Neural Networks.

In this work, Multilayer Perceptron Networks which is a class of feed forward neural network was used.

4.1 Multi-Layer Perceptron Neural Networks

A Perceptron is a single neuron model which was a precursor to the large neural networks. It is a field which explains how a simple brain model can be used for computational tasks like the forecasting problem we have. The power comes from the ability to learn the representation of the training data and how well it relates it to predict the output variable. Neural network learn a mapping and mathematically they are proven to learn any mapping function and they are considered to be a universal approximation algorithm. The predictive capability of the MLP neural networks comes from their multi-layered structure which can pick out features at different scales or resolutions and combines them into higher order features.

Neurons:

The building blocks for the artificial neural networks are artificial neurons.

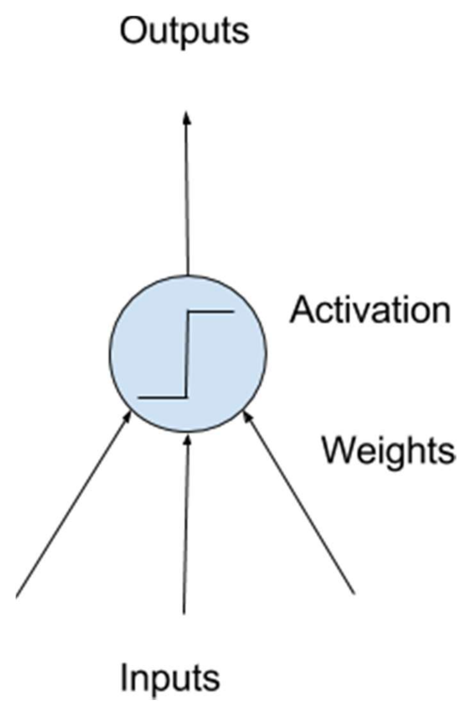

Figure 29.Model of a simple neuron[42]

These are simple computational units that take the weighted input signal and produce an output signal using activation function. The weights on the input are similar to the coefficients in the linear regression equation. Like linear regression, each neuron also has a bias and it can be assumed to have a value of 1.0 which must be weighted. Weights are often initialized to values from 0 to 0.3 randomly and sometimes, complex initialization schemes will be used.

These weighted inputs are added and passed through an activation function or transfer function. The activation simply maps the weighted input signal to the output signal. It is called activation function since it governs the threshold at which a neuron can be activated.

Networks of Neurons

A row of neurons are called layers and one network can have multiple layers. The bottom layer that takes the input from the dataset is called as input or visible layer.

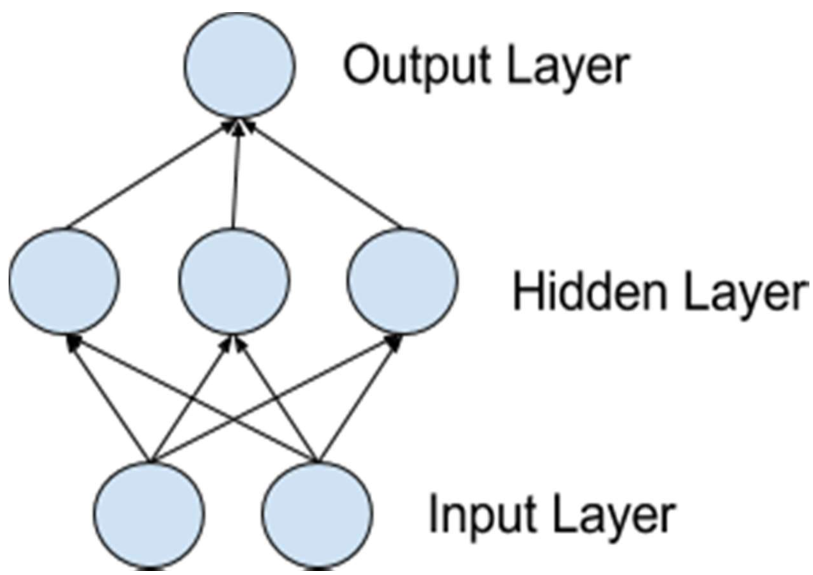

Figure 30. Model of a simple network[43]

The layers after the input layer are called hidden layers since it is not exposed to the network. The simplest ANN will have a single neuron in the hidden layer which directly outputs the value. Given, the increase in the computing power and variety of libraries, deep neural network with multiple hidden layers can be constructed. A Neural network with multiple hidden layers can be referred as deep neural network. They are called deep since they are unimaginably slow to train in the past, but it takes few minutes with the new hardware.

The final layer in the ANN is called as output layer since it is responsible for the output. The activation function in the output layer is strongly decided by the type of problem we are working on. A regression problem may have one neuron in the output layer and the layer may not have any activation function.

Training Networks:

Once the above parameters, the neural network can be trained on our dataset. ANN requires the data to be in a consistent way. This can be done using the technique known as normalization.

The classical and most used training algorithm still used is known as stochastic gradient descent algorithm. In this algorithm, one row of data is exposed at a time to the input of the network. The network processes the input data upwards activating the neurons thereby producing the output value. It is called as forward pass on the network. It is also the pass that is used to make predictions on the new data after the training the network.

The output of the network is compared with the actual output value. The error produced is back-propagated to the input of the network, one layer at a time. The weights are also adjusted based on the amount of error they produce. This type of algorithm is known as a back-propagation algorithm. This process is repeated for the whole training data in the network. The process of updating the entire network for the entire training dataset is known as an epoch. It may be repeated for tens, hundreds or even thousands of times until the error is reduced.

Weight Updates:

The weights in the network are updated based on the error produced after every training example. This process is known as online learning. It is very fast but it can result in chaotic changes in the network. Alternatively, the error produced after every training example is saved and it is updated at the end of the whole training. This efficient process is known as batch learning and it is more stable than the online learning.

Due to computational complexities and the bigger dataset, the number of examples the network is shown before the network is updated is often reduced to smaller batch typically in hundreds or thousands of training examples.

The amount by which the weights are updated is controlled by a certain parameter known as learning rate. Learning rate is the rate at which the weights are updated in the neural network. It is also known as step size since it controls the steps at which the weights are updated in the network for a particular error. The weight sizes are often very small in the range of 0.1, 0.001 or even lesser.

This update equation can be complemented with an additional parameters such as Momentum and Learning Rate Decay. Momentum allows the weights to get updated in the same direction even there is a less amount of error. Learning rate decay reduces the learning rate over the epochs at the beginning of the training where large changes occur in the weight and smaller changes occur at the end of the training.

Prediction:

Once the dataset is trained, the model can be used to make a prediction on the new data or unseen data. These data are validating by comparing it with the actual data using different statistical indices such as MAPE, MAE or RMSE.

4.2 Modeling of Electricity price through Deep learning algorithms

This model was developed using Tensor Flow deep learning platform. Tensor Flow is an open source library for fast numerical computation created by Google. It is a foundation library where it can be used directly or by using other wrapper libraries built on top of Tensor Flow. It can run on one CPU systems, GPUs as well as mobile devices and large distributed systems.

In general, deep learning model is divided into the following steps

i) Prepare the training, test data.

   The size of the dataset is an important factor in the accuracy, training and learning within the deep neural network. In this study, three-months, six-months, weekday/weekend dataset from the same Iberian electricity market (MIBEL). Single day (July-31, 2015) was selected to forecast and validate the performance of various dataset duration.

ii) Select the number of features/variables in the input and target data.

   In this study, 17 variables were used as mentioned in section 3.2. Price, load and temperature variables were included as variables in this study. Since deep neural network is good in modeling complex relationship between the dependent and independent variables, we included all the variables in the study. This was done by increasing the number of hidden layers from 2 to 4. Also, this model was tested for a lesser number of variables after performing variable selection which is clearly explained in the next chapter. Selecting important variables in the predictive model greatly reduces the computational time and improves the accuracy of the model.

iii) Normalize the input data along the features.

   Since the variables used in this study comprises of price, load and temperature, it is important to normalize the data along the features. These variables differ in the range, so it is important to convert all the variables within the 0-1 range. This normalization helps in avoiding one variable with the greater numeric range having great influence on the response variable which impacts the prediction accuracy. This normalization is done by performing the following computation for all the variables.

Normalized input variables= $\frac{x - x_{\min}}{x_{\max} - x_{\min}}$ (7)

iv) Use the training parameters such as epochs, learning rate and mini_batch size.

Epochs are the no of times (iterations), the dataset is trained to predict the unknown test dataset. No of epochs were kept uniform to compare the performance of the different cases. In this study, epochs were kept as 500.

Leaning rate is the rate at which the weights are updated in the neural network. In this study, learning rate was kept as 0.001.

Mini-batch size is the size at which the training data is split-up. In this study, it was kept as 100. Neurons are the building blocks of ANN. In this work, it is kept uniformly at 10 to compare the performance of the different cases.

v) Building the model with multiple hidden layers using Keras

Keras can be used to build and modify the neural network model. Before we can train the model, it is necessary to specify the loss function and the optimization algorithm. In this work, Adaptive Moment Estimation (ADAM) optimizer was used to train the model and minimize the mean square error loss function.

ADAM optimizer uses a stochastic gradient descent algorithm and currently, it is one of the best optimizer performing better than other adaptive learning algorithms. It computes the adaptive learning rate for each parameter.

vi) Training the model

In this work, mini-batch gradient descent is used. Here, the entire dataset is not used at once. Instead, a subset of the dataset known as mini-batch is used during one training iteration. While the model is training, the validation dataset is used parallel to compute the validation error and gauge how well the model is learning.

vii) Validate the deep learning model.

Finally, the model is validated using the actual data. The model is validated using a Statistical index called Mean Absolute Percentage Error (MAPE).

$$MAPE_{day} = \frac{1}{24}\sum_{i=1}^{24}\frac{p^{actual} - p^{pred}}{p^{actual}} \tag{8}$$

Where $MAPE_{day}$ is the daily error. p^{actual} is the actual price for the day, while p^{pred} is the predicted priced for the day

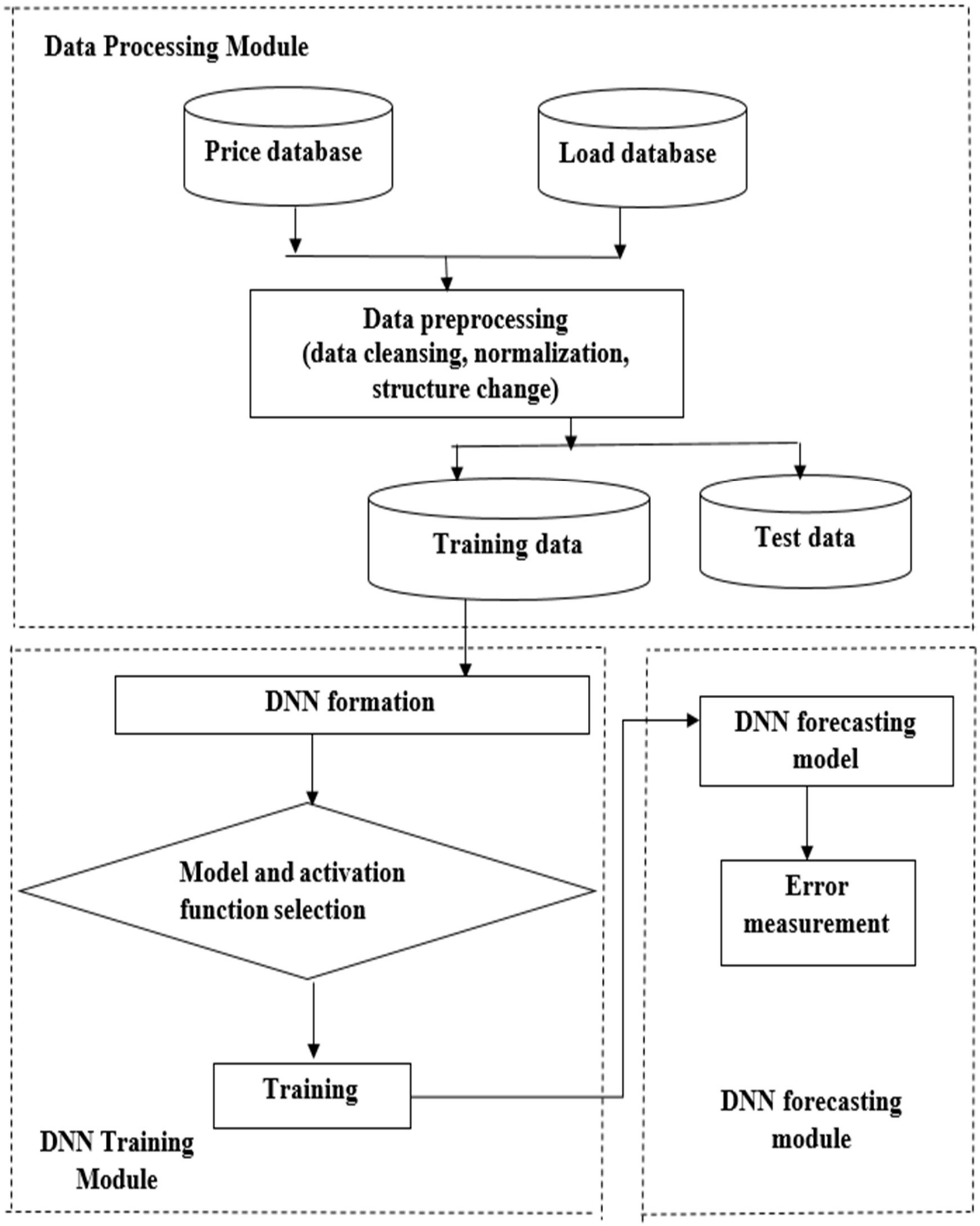

Figure 31:DNN based day ahead price forecast[44]

4.3 Hybrid ARIMA- Deep learning Model

Step 1. In Stage-1, ARIMA was used to predict the day-ahead prices. Input variables that are considered include historical electricity prices, generation and consumption load and weather data like solar irradiance, temperature and wind speed. These variables are fed as time-series data to the ARIMA model. The relationship between the predictor variables and forecasted variables is then initialized through this model.
Step 2. An '*auto-arima'* function built in R-software was used to identify the best-fit by inputting the residual values (p, d, q) of the three time series components I, AR, and MA. After identifying the best-fit model, the '*forecast' function* is used to predict the day-ahead price.

Step 3. The same process is repeated for other datasets. In this study, three- month and six-month of datasets from the Iberian electricity price market are used to predict the day-ahead electricityprices.
Step 4. After the price predictions, residuals are calculated by differencing the predicted value (f) from the actual value (A).

B. Stage-2: Input residuals to the Hybrid Model

1) ARIMA-FFNN
The steps involved in the two stage residual extraction method, that uses combinations of ARIMA and FFNN, are as follows:

Step 1. In Stage-2, the residual dataset is fed as an input to the FFNN model

Step 2. Finally, the calculated residual (R) from Step 1 is then added to the predicted price from the ARIMA method (P) to get the final price.

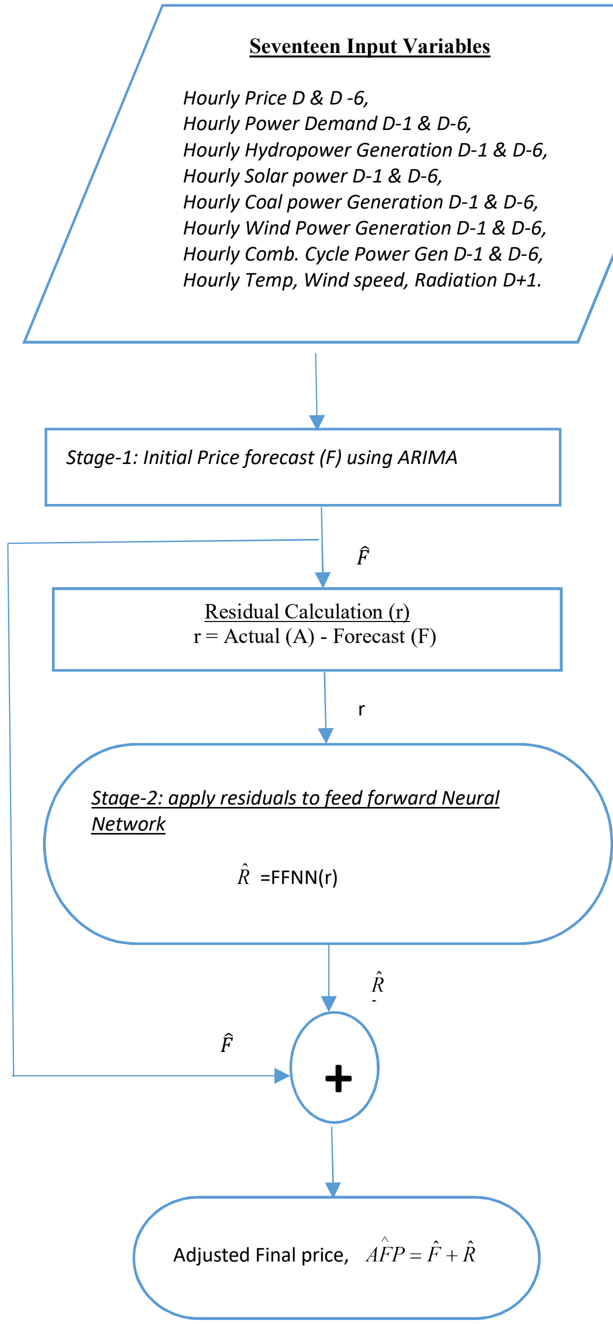

Figure 32: Flowchart of the proposed ARIMA-FFNN Model

4.4 Deep learning results & Conclusion

Multi-layered Perceptron Neural Networks were used to predict the day-ahead price of the Iberian Energy Market Operator (MIBEL). Two datasets (Three months & Six months) were used to validate the performance of the model. Single day (July-31, 2015) was selected to forecast and validate the performance of the deep neural network model. Three different set of variables (17, 4, 2) were used in this problem. Relu activation function was used in most of the cases in this study since it is easy to train when compared with the other activation function. Also, Relu is not subjected to vanishing gradient problem. Epochs (No of iterations) and no of neurons were kept uniform to compare the different cases in this problem. Different layers were used to train and test the dataset.

Case-1: (Activation function= Relu, epochs=500, Relative importance method)

In this case, three months dataset with different sets of variables were used to predict the day-ahead price of the electricity market. These variables are considered based on the variable selection method explained in the next chapter. 17 variables used in this case study were clearly mentioned in the chapter- 2. 4 variables used are Price D.6, Price D, P.DD.6 and C.GD.6. 2 variables used are Price D.6 and Price D. Three different layers (2, 3, and 4) were used to understand the behavior of the model. For 17 variables, as we increased the no of layers model performed well with the complex relationship between the 17 independent and dependent variables. This can be seen in the reduction of MAPE. For 4 variables, as we increase the layers, inconsistencies were observed. This can be reduced by increasing the no of epochs. But due to more computational time, this task was not performed. For 2 variables, the model performed with lesser no of layers.

Table 8: MAPE evaluation for three months dataset using Feed forward neural network

MAPE	17 variables	4 variables	2 variables
Relative importance variable selection approach			
2 layers	11.03	5.43	2.35
3 Layers	8.89	3.70	4.61
4 Layers	8.24	4.52	3.81
Multivariate Adaptive Regression Splines variable selection approach			
2 layers	11.03	6.92	2.35
3 Layers	8.89	6.01	4.61
4 Layers	8.24	4.64	3.81
Random forest variable selection approach			
2 layers	11.03	4.79	5.9
3 Layer	8.89	5.33	6.96
4 Layer	8.24	4.96	4.84

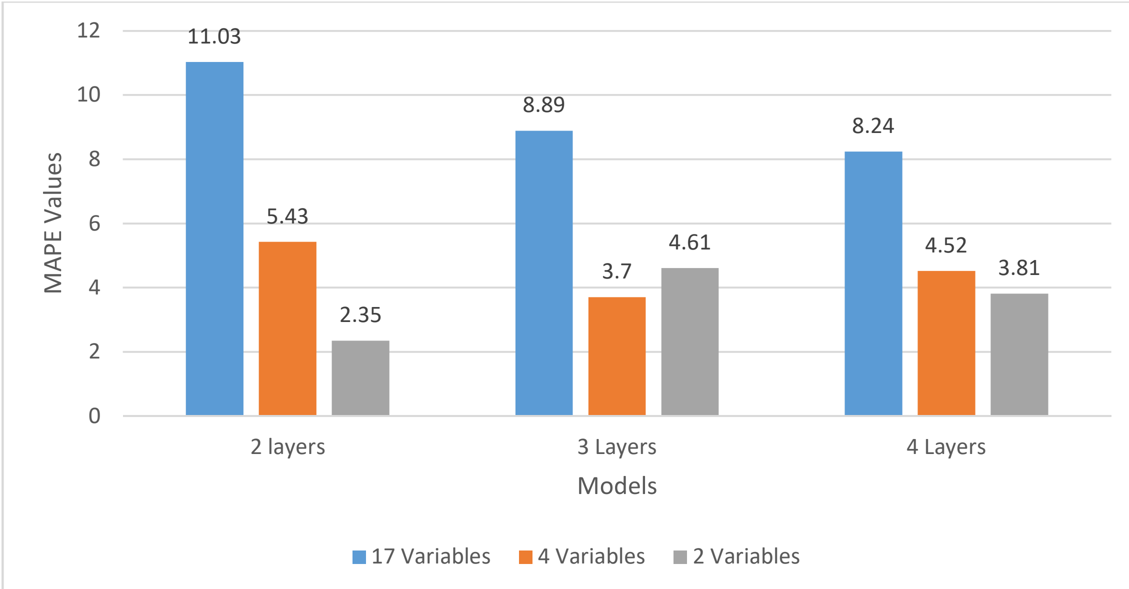

Figure 33: Comparison of MAPE for different layers of deep neural networks using Relative importance variable selection approach for three months duration dataset to predict day-ahead price (July 31, 2015)

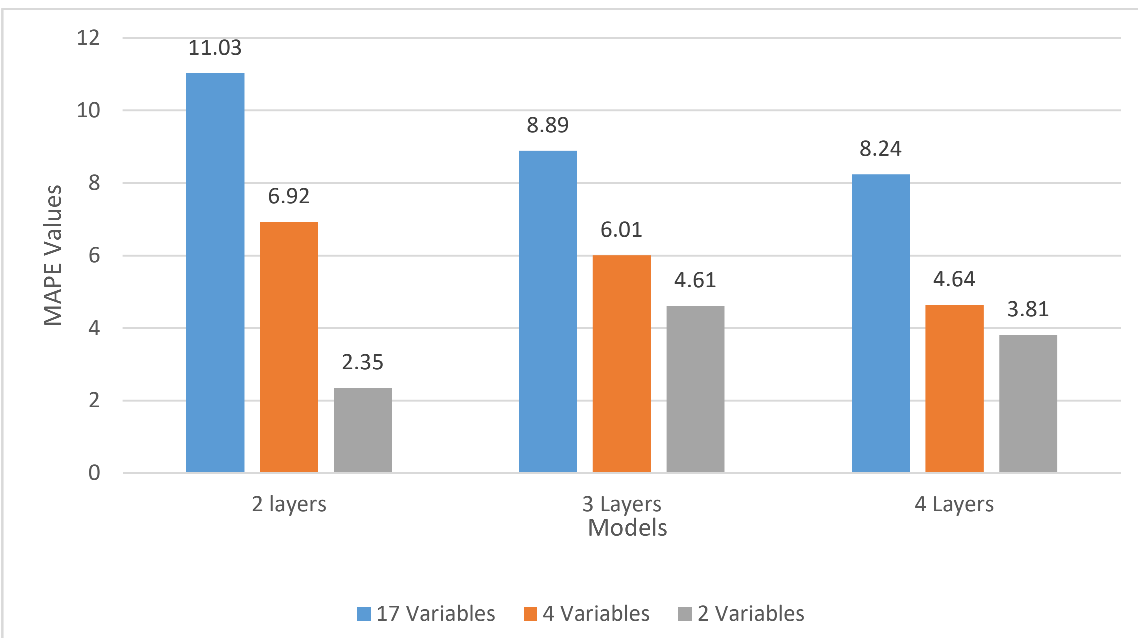

Figure 34: Comparison of MAPE for different layers of deep neural networks using MARS variable selection approach for three months duration dataset to predict day-ahead price (July 31, 2015)

Case-2: (Activation function= Relu, epochs=500, Multivariate Adaptive Regression Splines method)

In this case, three months dataset with same sets of variables were used to predict the day-ahead price of the electricity market. These variables are considered based on the Multivariate Adaptive Regression Splines method explained in the next chapter. 17 variables used in this case study were clearly mentioned in the chapter- 2. Same three different layers (2, 3, and 4) were used to understand the behavior of the model. For 17 variables, as we increased the no of layers model performed well with the complex relationship between the 17 independent and dependent variables. For 4 variables, as we increase the layers, the model performed with no of hidden layers. For 2 variables, as we increase the layers, inconsistencies were observed. This can be reduced by increasing the no of epochs.

Case-3: (Activation function= Relu, epochs=500, Random forest method)

In this case, three months dataset with same sets of variables were used to predict the day-ahead price of the electricity market. These variables are considered based on the Random forest method explained in the next chapter. Same 17 variables used in this case study were clearly mentioned in the chapter- 2. Same three different layers (2, 3, and 4) were used to understand the behavior of the model. For 17 variables, as we increased the no of layers model performed well with the complex relationship between the 17 independent and dependent variables. For 4 & 2 variables, as we increase the layers, same inconsistencies were observed. This can be reduced by increasing the no of epochs.

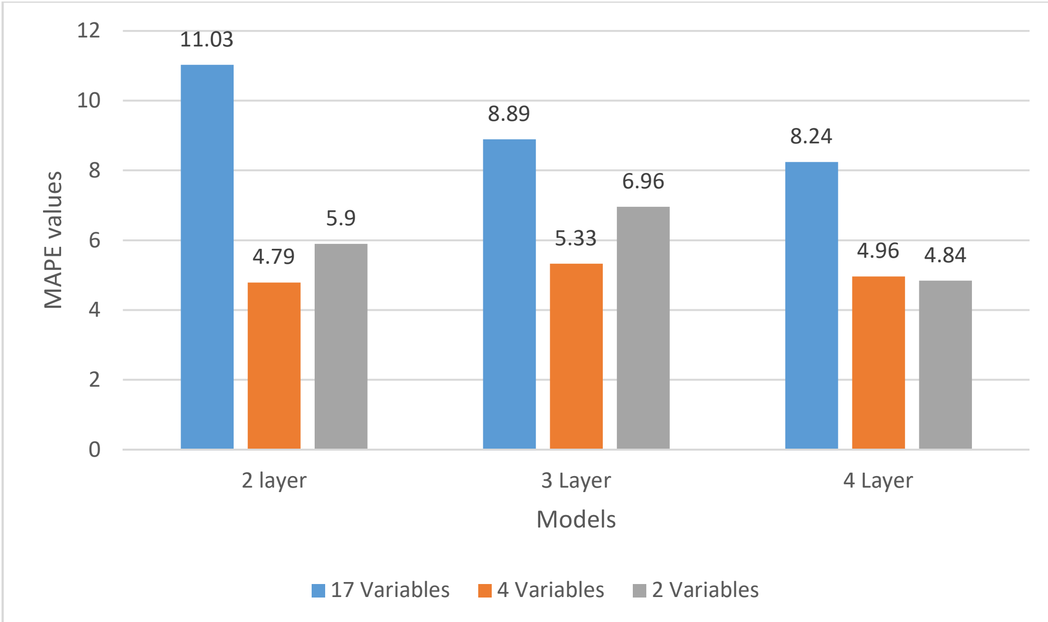

Figure 35: Comparison of MAPE for different layers of deep neural networks using Random forest variable selection approach for three months duration dataset to predict day-ahead price (July 31, 2015)

Case-4: (Activation function= Relu, epochs=500, Relative importance method)

In this case, six months dataset with different sets of variables were used to predict the day-ahead price of the electricity market. These variables are considered based on the Relative importance method explained in the next chapter. 17 variables used in this case study were clearly mentioned in the chapter- 2. Three different layers (2, 3, and 4) were used to understand the behavior of the model. Clearly, there was a reduction in the MAPE value as we reduce the number of variables. This model performed better when compared with the three months dataset mention in the previous cases. Also, clearly it can be seen that there is inconsistency in the MAPE value as we increase the number of layers for all sets of the variable.

Table 9: MAPE evaluation for six months dataset using Feed forward neural network

MAPE	**17 variables**	**4 variables**	**2 variables**
Relative importance variable selection approach			
2 layers	8.4	3.78	2.36
3 Layers	5.4	3.12	2.37
4 Layers	10.8	4.78	2.35
Multivariate Adaptive Regression Splines variable selection approach			
2 layers	8.4	3.78	2.36
3 Layers	5.4	3.12	2.37
4 Layers	10.8	4.78	2.35
Random forest variable selection approach			
2 layers	8.4	4.08	2.36
3 Layers	5.4	4.10	2.37
4 Layers	10.8	4.08	2.35

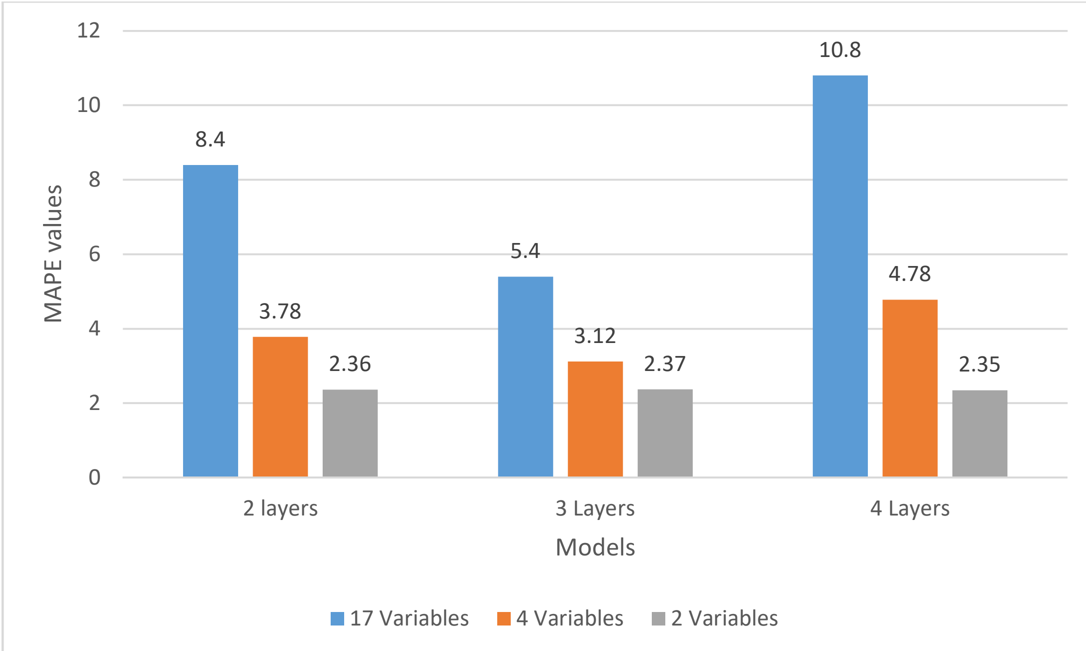

Figure 36: Comparison of MAPE for different layers of deep neural networks using Relative importance variable selection approach for six months duration dataset to predict day-ahead price (July 31, 2015)

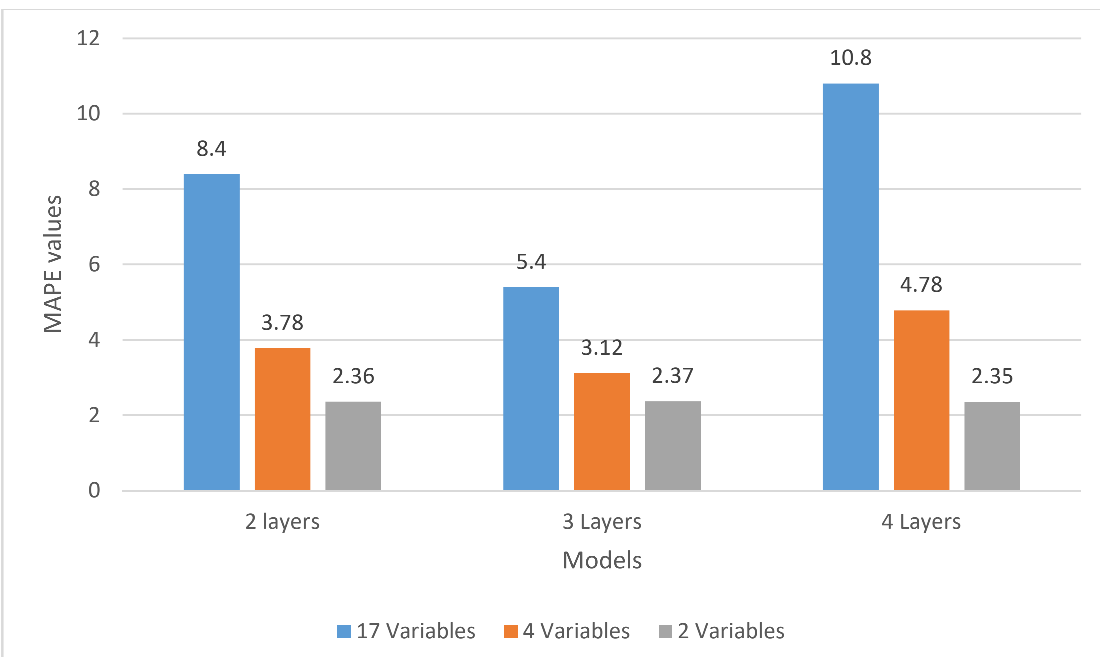

Figure 37: Comparison of MAPE for different layers of deep neural networks using MARS variable selection approach for six months duration dataset to predict day-ahead price (July 31, 2015)

Case-5: (Activation function= Relu, epochs=500, Multivariate Adaptive Regression Splines method)

In this case, six months dataset with different sets of variables were used to predict the day-ahead price of the electricity market. These variables are considered based on the Multivariate Adaptive Regression Splines method explained in the next chapter. 17 variables used in this case study were clearly mentioned in the chapter- 2. The Same set of 4 variables were used. Three different layers (2, 3, and 4) were used to understand the behavior of the model. Clearly, there was a reduction in the MAPE value as we reduce the number of variables. This model performed better when compared with the three months dataset mention in the previous cases. Also, clearly it can be seen that there is inconsistency in the MAPE value as we increase the number of layers for all sets of the variable.

Case-6: (Activation function= Relu, epochs=500, Random forest method)

In this case, six months dataset with different sets of variables were used to predict the day-ahead price of the electricity market. These variables are considered based on the Random forest method explained in the next chapter. 17 variables used in this case study were clearly mentioned in the chapter- 2. Three different layers (2, 3, and 4) were used to understand the behavior of the model. Clearly, there was a reduction in the MAPE value as we reduce the number of variables. This model performed better when compared with the three months dataset mention in the previous cases. Also, clearly it can be seen that there is inconsistency in the MAPE value as we increase the number of layers for all sets of the variable.

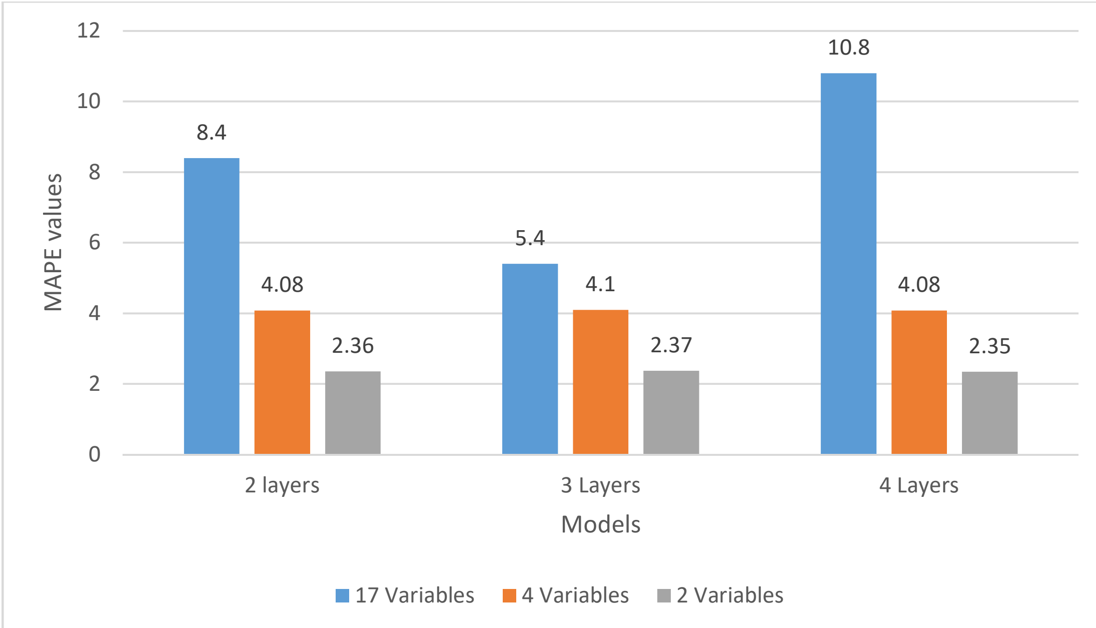

Figure 38: Comparison of MAPE for different layers of deep neural networks using Random forest variable selection approach for six months duration dataset to predict day-ahead price (July 31, 2015)

Case-7: (Activation function= Relu, epochs=500, Three common variables from variable selection approach, Dataset duration= Three months)

In this case, three months dataset with three common variables selected from the variable selection approach were used to predict the day-ahead price of the electricity market. Three different layers (2, 3, and 4) were used to understand the behavior of the model. For these variables, as we increase the layers, inconsistencies were observed. This can be reduced by increasing the no of epochs. MAPE results for three common variables were compared with all the previous combination (17 variables, 4 variables using different approaches, 2 variables using different approaches) shown in table 10.

Table 10: MAPE Evaluation for Three months dataset using Feed forward neural network using three common variables

MAPE	17 var-	4 var- using LR	4 var- using GCV	4 var- using RF	3 var- using all methods	2 var- using LR & GCV	2 var- using RF
three common variables from three different variable selection approaches							
2 layer	11.03	5.43	6.92	4.79	5.03	2.35	5.9
3 Layer	8.89	3.70	6.01	5.33	5.55	4.61	6.96
4 Layer	8.24	4.52	4.64	4.96	5.15	3.81	4.84

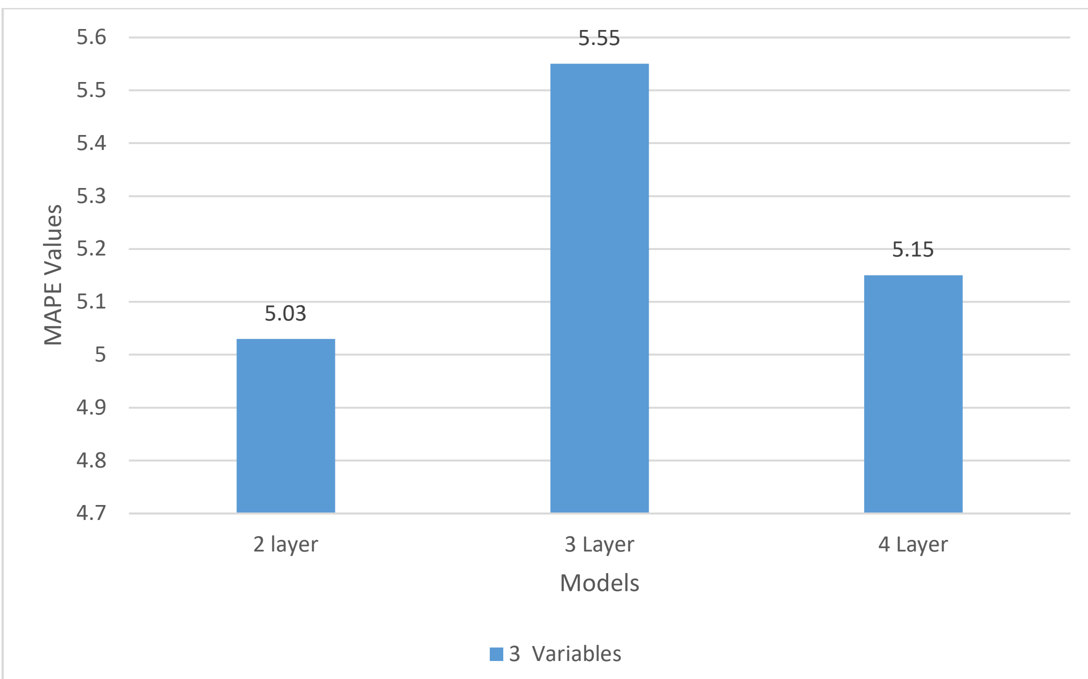

Figure 39: Comparison of MAPE for different layers of deep neural networks using three common variables for three months duration dataset to predict day-ahead price (July 31, 2015)

Case-8: (Activation function= Relu, epochs=500, Three common variables from variable selection approach, Dataset duration= Six months)

In this case, six months dataset with three common variables selected from the variable selection approach were used to predict the day-ahead price of the electricity market. Three different layers (2, 3, and 4) were used to understand the behavior of the model. For these variables, as we increase the layers, inconsistencies were observed. This can be reduced by increasing the no of epochs. MAPE results for three common variables were compared with all the previous combination (17 variables, 4 variables using different approaches, 2 variables using different approaches) shown in table 11.

Table 11: MAPE Evaluation for Six months dataset using Feed forward neural network using three common variables

MAPE	17 var-	4 var- using LR & MARS	4 var- using RF	3 var- using all methods	2 var- using LR & GCV
three common variables from three different variable selection approaches					
2 layer	8.4	3.78	4.08	3.23	2.36
3 Layer	5.4	3.12	4.10	4.19	2.37
4 Layer	10.8	4.78	4.08	3.42	2.35

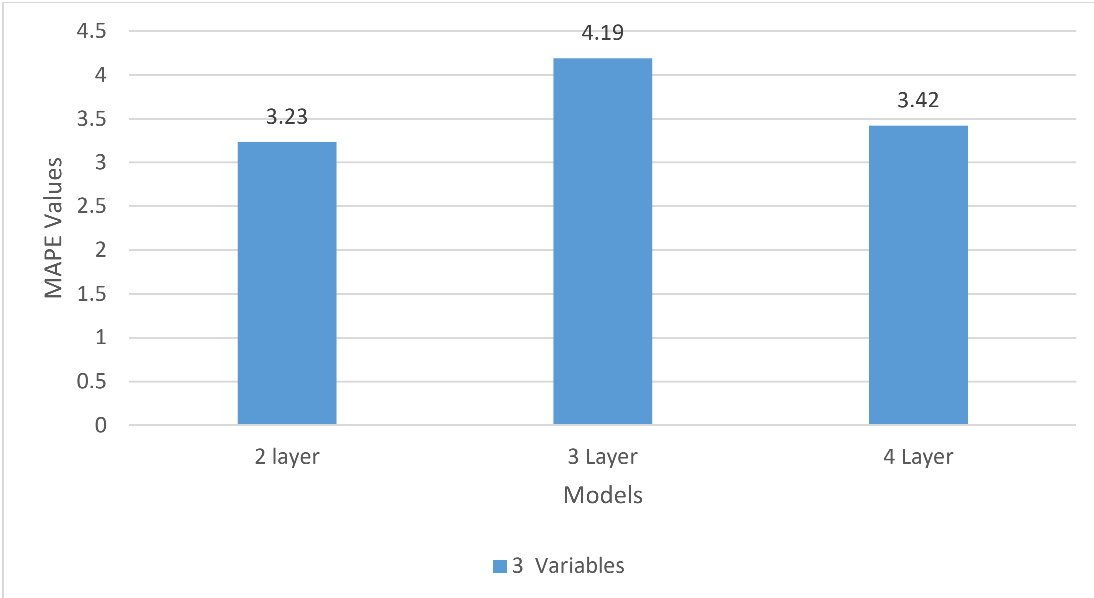

Figure 40: Comparison of MAPE for different layers of deep neural networks using three common variables for six months duration dataset to predict day-ahead price (July 31, 2015)

Case-8: (Activation function= Relu, epochs=500, Three common variables from variable selection approach, Weekend & weekday dataset for six months)

In this case, weekend & weekday datasets with three common variables selected from the variable selection approach were used to predict the day-ahead price of the electricity market. Since the sample size is small when compared to the previous three and six months dataset, 2 layers were used to understand the behavior of the model. Hybrid ARIMA-Multi-layered deep neural network was used in this case. This hybrid model performed better than the other hybrid models used in this thesis work. The comparison results are shown in the next chapter 4 which talks about the features selection approaches applied to the different hybrid models.

Table 12: MAPE Evaluation for weekday/weekend using Feed forward neural network using three common variables

MAPE	3 Variables - Weekday	2 Variable - Weekend
2 layer	3.14	7.34

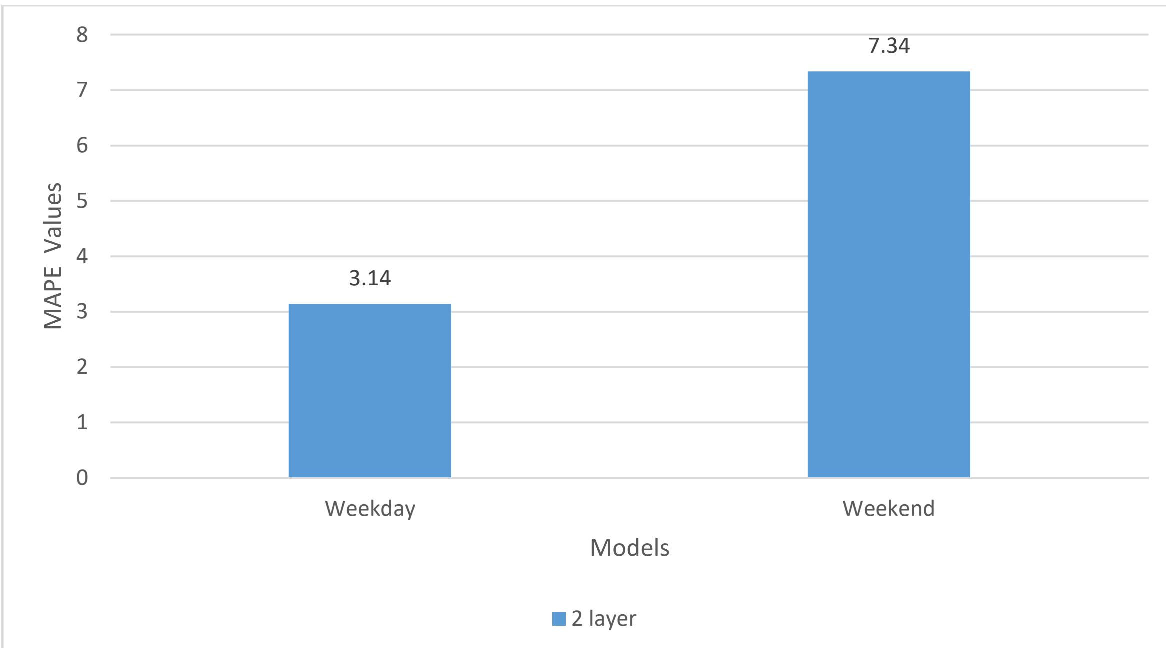

Figure 41: Comparison of MAPE for different layers of deep neural networks using three common variables for weekday/weekend duration dataset to predict day-ahead price (July 31, 2015)

5 FEATURE SELECTION

Feature selection [45] is an important step to improve the accuracy of the predictive model. It is greatly used to reduce the computation time and discard the features that have a high correlation among them. Feature Selection is also known as variable selection or attribute selection. It is the automatic selection of the features that are very relevant to our problem.

Feature selection is different from dimensionality reduction. While feature selection includes or exclude the variables, dimensionality reduction creates a new combination of features. Some of the examples of dimensionality reduction methods are Principal Component Analysis (PCA), Singular Value Decomposition (SVD) and Sammon's mapping.

Feature selection methods act a filter muting out important features that are highly irrelevant to the data. They help us to create an accurate predictive model by choosing features that give better accuracy whilst requiring fewer data. Feature selection helps us to find the unwanted, redundant features that do not contribute to the predictive model or in fact which greatly reduces the performance of the model. Fewer variables are desired in the predictive model since it reduces the complexity by reducing the computation time. Model with fewer variables is easy to comprehend and explain. In short variable selection can be summarized as follows: improving the prediction performance of the predictors, selecting predictors that are faster and efficient, and

"The objective of variable selection is three-fold: improving the prediction performance of the predictors, providing faster and more cost-effective predictors, and providing a better understanding of the underlying process that generated the data".

— Guyon and Elisseeff in "An Introduction to Variable and Feature Selection" (PDF)

Feature selection algorithms:

There are three general classes of feature selection algorithms. They are Filter Methods Wrapper Methods, Embedded Methods.

Filter Methods:

Filter methods use a statistical measure to compute the score of the features. These features are ranked by their score and kept or removed from the dataset. These methods are often univariate and consider the features with regard to the predictor variables ore with regard to the response variable.

Example of Filter methods are Chi squared test, information gain and correlation coefficient scores.

Wrapper Methods:

Wrapper methods select the set of features as a search problem, compares it with the other set of features and evaluates it. A predictive model is used to evaluate the different set of features by assigning a score based on the accuracy of the model. The search process may be methodical, stochastic or it may use heuristic.

Example for wrapper method is Recursive feature elimination algorithm.

Embedded Methods.

Embedded methods select which feature best contribute to the accuracy of the model. The best example of embedded methods is regularization methods.

The regularization method is also called as penalization methods since it introduces additional constraints into the predictive algorithm. This is done by introducing bias into the model towards fewer coefficients, thereby reducing the complexity.

Examples of regularization algorithms are the LASSO, Elastic Net and Ridge Regression.

5.1 Feature selection approaches

Finding the best feature selection approaches that best explains the variance in the dependent variable or response variable is the key to build a high-performance predictive model.

5.1.1 Relative Importance by Linear regression (LR)

A linear regression can be used to identify the key variables in the model. This method identifies the important variables as a relative percentage. The first step involves fitting the linear regression model using the given set of predictor variables. Then, by using ‘calc.relimp’ function in ‘R’ software, the relative percentage can be computed.

Relative importance refers to the measure of the contribution of the individual regressors in the multiple regression model. The assessment of the relative importance in the model is simple as long as the regressors are uncorrelated.

It gives the total proportion of the variance explained by the model with all the variables.

It also gives the individual contribution of each predictor variable to the overall R^2.

R^2 Measures the proportion of the variance in the dependent variable that is explained by the regressors in the model.

5.1.2 Multivariate Adaptive Regression Splines (MARS)

Multivariate Adaptive Regression Splines can be used for variable selection. The earth package in 'R' software estimates the variable importance based on the Generalized cross-validation (GCV), a number of subset models the variable occurs (nsubsets) and residual sum of squares (RSS).

There are three statistics that can be used to measure the importance of the variable in the MARS model. They are Generalized cross-validation (GCV), a number of subset models the variable occurs (nsubsets) and residual sum of squares (RSS).

MARS model includes a backward elimination features selection which estimates the variable Importance based on the reduction in the error of the generalized cross-validation.

It tracks the changes in the model statistics for each predictor variable and adds the reduction in the statistics such as GCV when new features are added to the model. This total reduction gives the measure of the variable importance.

Generalized cross-validation (GCV):

Generalized cross-validation is a model validation technique for assessing the statistical results. It tells how well the predictive model performs in practice. The main goal of cross-validation is to limit problems like overfitting by testing the dataset in the training phase. It also gives an insight on how well the predictive model performs to a particular dataset (unknown dataset).

Residual sum of squares (RSS):

In statistics, the residual sum of squares (RSS), also known as the sum of squared residuals (SSR) or the sum of squared errors of prediction (SSE) is the sum of the square of the residuals. Residuals are deviations from the actual values of data. It helps to estimate the discrepancy between the observed and the actual data. Smaller the RSS, better the predictive performance of the model. It indicates how well the model generalizes to future results. It provides the measure to aid in model selection and parameter selection.

5.1.3 Random forest for variable selection (RF)

Random forest can be very effective in determining the best set of predictors by best explaining the variance in the response or independent variable.

How variable Importance works in random forest:

1. For each tree in the model, it calculates the no of votes
2. Then, performs a random permutation of the predictor's value (let's say variable-k) in the dataset and check the number of votes for the correct class.
3. Subtract the number of votes for the correct class in the permuted data from the number of votes for the correct class in the original dataset.
4. The average of the value in all the trees is the variable importance score. This score is normalized by computing the standard deviation.
5. Variable having the large values are ranked more important than the other variables.

5.2 Feature Selection Results:

Three different feature selection approaches were used in this thesis work as mentioned earlier. Variables selected from these approaches were applied to the different hybrid models such as ARIMA, ARIMA-GLM, ARIMA- RF, ARIMA-SVM, ARIMA- Deep neural network and multi-layered deep neural network with different layers. All these combinations were tested with the same three months, six months and weekend/weekday dataset.

5.2.1 Variable Importance Percentage Computation

Case-1: Feature / Variable Selection using Relative importance method for three months dataset

Table 13: Feature selection using Relative Importance for three months dataset

S.NO	FEATURES/VARIABLES	RELATIVE PERCENTAGE
1	PRICED.6	0.21385445
2	PRICED	0.15704328
3	P.DD.6	0.14576855
4	C.GD.6	0.08898134
5	C.PD.6	0.05958672
6	P.DD.1	0.04435511
7	WIND SPEED	0.04430523
8	H.GD.6	0.03662183
9	C.PD.1	0.03564394
10	W.PD.1	0.03521148
11	TEMP	0.03469284
12	S.PD.6	0.02087770
13	C.GD.1	0.01930565
14	H.GD.1	0.01888319
15	H.GD.1	0.01871824
16	S.PD.1	0.01546665
17	IRRADIANCE	0.01068381

Case-2: Feature / Variable Selection using Multivariate Adaptive Regression Splines method for three months dataset

Table 14: Feature selection using generalized cross-validation (GCV), number of subset models the variable occurs (nsubsets) and residual sum of squares (RSS) for three months dataset

S.NO	VARIABLES	NSUBSETS	GCV	RSS
1	PriceD	20	100	100
2	PriceD.6	19	63.7	64.2
3	P.DD.6	17	43.7	44.6
4	wind speed	17	43.7	44.6
5	C.PD.1	16	37	38.1
6	P.DD.1	14	28	29.3
7	Temp	13	24.7	26.2
8	H.GD.1	12	22.9	24.3
9	S.PD.1	7	11.1	12.9
10	W.PD.1	5	8	9.8

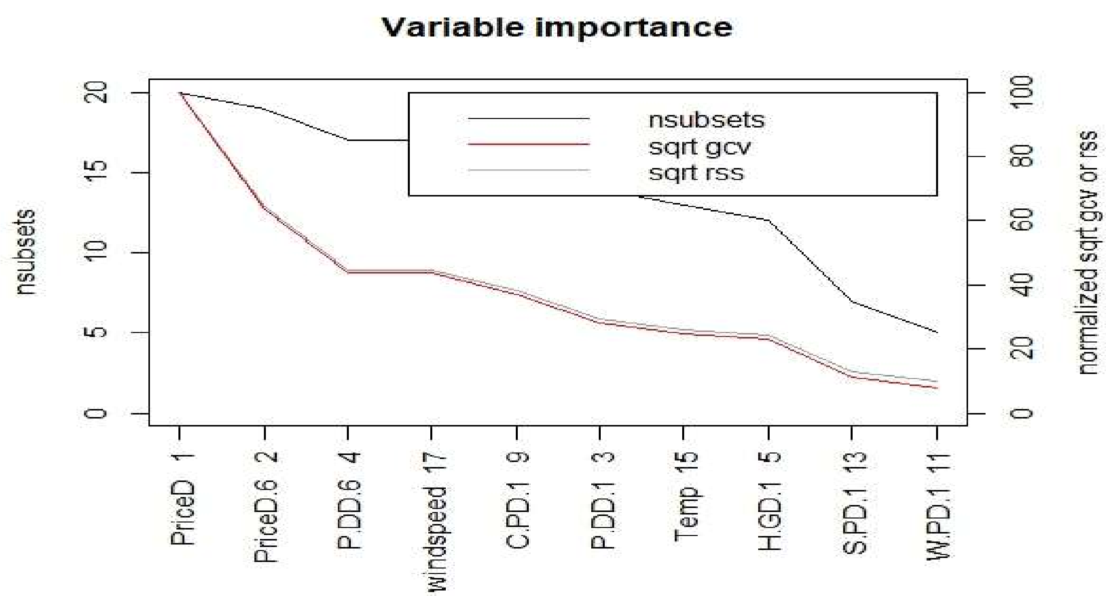

Figure 42: Variable importance plot for three months dataset

Case-3: Feature / Variable Selection using Random forest method for three months dataset

Table 15: Feature selection using Random forest for three months dataset

S.NO	FEATURES/VARIABLES	MEAN DECREASE IN ACCURACY
1	PriceD	13.5009
2	PriceD.6	18.18507
3	P.DD.1	4.914126
4	P.DD.6	16.60764
5	H.GD.1	5.346575
6	H.GD.6	8.859291
7	C.GD.1	2.648366
8	C.GD.6	10.3902
9	C.PD.1	8.71356
10	C.PD.6	11.73852
11	W.PD.1	3.183474
12	W.PD.6	2.334897
13	S.PD.1	1.796189
14	S.PD.6	2.431934
15	Temp	2.80884
16	irradiance	2.545041
17	windspeed	5.796735

Case-4: Feature / Variable Selection using Relative importance method for six months dataset

Table 16: Feature selection using Relative importance for six months dataset

S.NO	FEATURES/VARIABLES	MEAN DECREASE IN ACCURACY
1	PriceD	0.23318
2	PriceD.6	0.21085
3	P.DD.6	0.093681
4	C.PD.1	0.07304
5	W.PD.1	0.05857
6	C.PD.6	0.05188
7	C.GD.6	0.051605
8	Temp	0.04573
9	P.DD.1	0.038579
10	H.GD.1	0.026128
11	W.PD.6	0.025694
12	windspeed	0.024878
13	H.GD.6	0.02375
14	C.GD.1	0.017722
15	irradiance	0.01022
16	S.PD.6	0.007337
17	S.PD.1	0.00715

Case-5: Feature / Variable Selection using Multivariate Adaptive Regression Splines method for six months dataset

Table 17: Feature selection using generalized cross-validation (GCV), number of subset models the variable occurs (nsubsets) and residual sum of squares (RSS) for six months dataset

S.NO	VARIABLES	NSUBSETS	GCV	RSS
1	PriceD	24	100	100
2	PriceD.6	23	59.1	59.7
3	P.DD.6	21	38	39
4	C.PD.1	21	38	39
5	P.DD.1	20	35.9	37
6	windspeed	19	31	32.2
7	Temp	18	28.1	29.4
8	H.GD.6	17	25.4	26.8
9	C.GD.6	16	23.1	24.5
10	C.PD.6	11	14.5	16.1
11	W.PD.6	6	7.7	9.3
12	W.PD.1	5	6.2	7.8

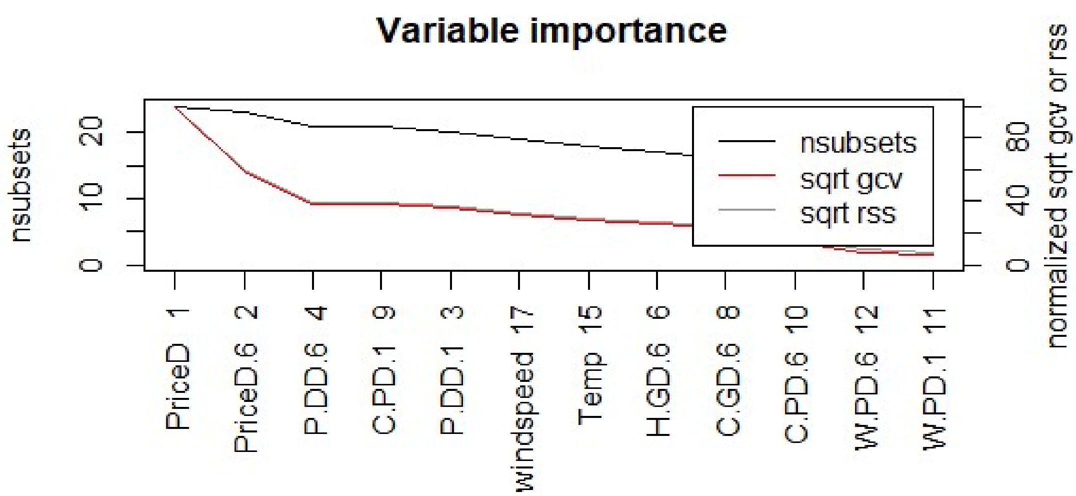

Figure 43: variable importance plot for six months dataset

Case-6: Feature / Variable Selection using Random forest method for six months dataset

Table 18: Feature selection using Random forest for six months dataset

S.NO	FEATURES/VARIABLES	MEAN DECREASE IN ACCURACY
1	PriceD	30.270782
2	PriceD.6	27.72341
3	P.DD.1	6.823675
4	P.DD.6	10.94817
5	H.GD.1	10.89454
6	H.GD.6	10.03618
7	C.GD.1	6.621256
8	C.GD.6	19.20725
9	C.PD.1	14.2654
10	C.PD.6	7.790629
11	W.PD.1	8.915463
12	W.PD.6	4.221552
13	S.PD.1	4.768152
14	S.PD.6	2.89059
15	Temp	8.641536
16	irradiance	4.705143
17	WINDSPEED	6.63406

5.2.2 Applying Feature Selection to the Different Hybrid Models

Different hybrid models were used to predict the day-ahead price of the Iberian Energy Market Operator (MIBEL). Four datasets (Three months, Six months, weekday, and weekend) were used to validate the performance of the model. Single day (July-31, 2015) was selected to forecast and validate the performance of the different hybrid models. For weekend alone, July 26, 2015 was selected to forecast and validate the performance of the different hybrid models. Three different set of variables (17, 4, 2) were used in this problem. At last, three common variables selected from these feature selection approaches were tested with all these datasets.

Case-1: Applying Hybrid models to three months dataset using all 17 variables

In this case, three months dataset with all 17 variables were used to predict the day-ahead price of the electricity market. 17 variables used in this case study were clearly mentioned in the chapter- 2. For 17 variables, we can see that all hybrid models performed better. Clearly, ARIMA-RF outperforms other hybrid method in this case.

Case-2: Applying Hybrid models to three months dataset using Relative importance approach for 4 variables

In this case, three months dataset with four variables taken from Relative importance approach were used to predict the day-ahead price of the electricity market. Four variables taken from this approach as follows: PriceD.6, Price D, P.DD.6, C.GD.6. For 4variables, we can see that all hybrid models performed better. Clearly, ARIMA-GLM outperforms other hybrid method in this case. Also, there was a significant reduction in MAPE in almost all the hybrid models in this case than using all the 17 variables.

Table 19: Comparison of day-ahead forecasting performance of several hybrid models using feature selection approaches for three months dataset

Parameter	ARIMA	ARIMA-GLM	ARIMA-SVM	ARIMA- RF
MAPE-17 Variables	3.29	3.22	3.29	3.10
MAPE- 4 Variables (Relative Importance)	3.05	2.83	3.00	3.027
MAPE- 4 Variables (Multivariate Adaptive Regression Splines (MARS))	3.10	3.10	3.02	3.53
MAPE- 4 Variables (Random forest)	3.10	2.90	3.04	3.06
MAPE- 3 Variables	3.07	2.85	3.008	3.23
MAPE- 2 Variables (Relative Importance)	2.27	2.27	2.28	2.78
MAPE- 2 Variables (Multivariate Adaptive Regression Splines (MARS))	2.27	2.27	2.28	2.78
MAPE- 2 Variables (Random forest)	2.27	2.27	2.28	2.78

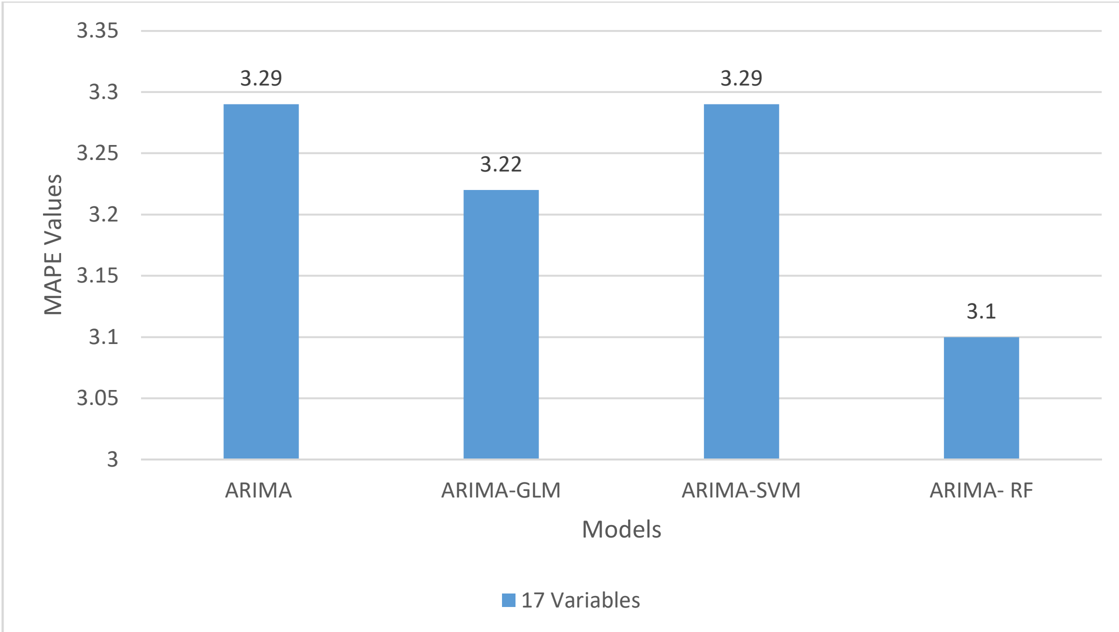

Figure 44: Comparison of MAPE for 90 days (May 01, 2015 to July 30, 2015) for 17 variables to predict day-ahead price (July 31, 2015)

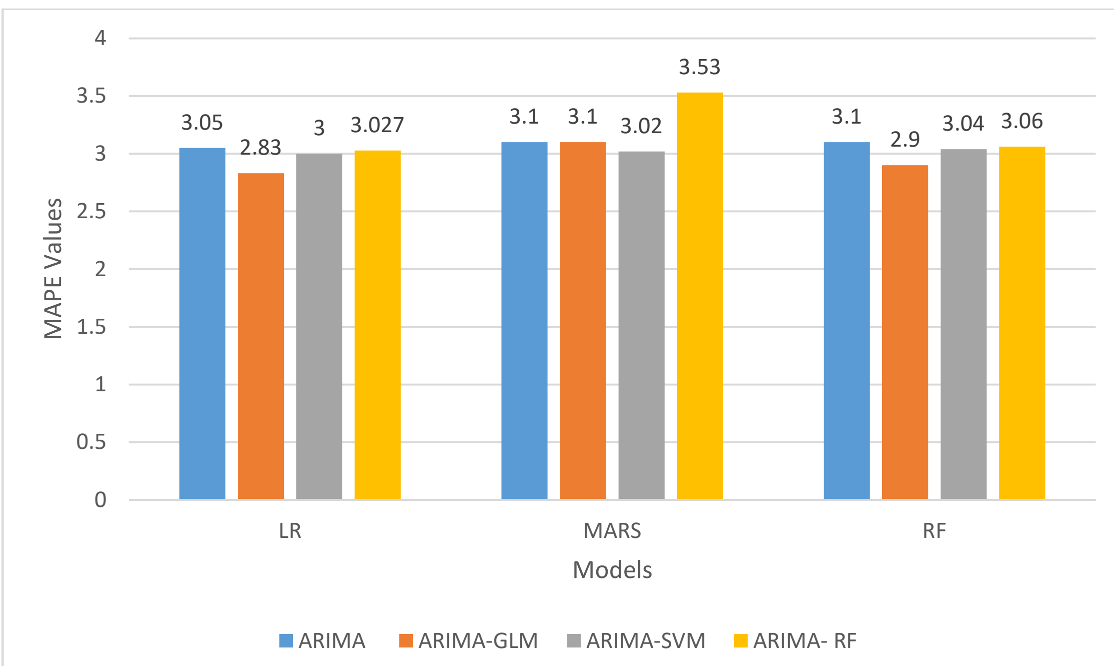

Figure 45: Comparison of MAPE for 90 days (May 01, 2015 to July 30, 2015) for 4 variables selected from variable selection approaches to predict day-ahead price (July 31, 2015)

Case-3: Applying Hybrid models to three months dataset using Multivariate Adaptive Regression Splines (MARS) for 4 variables

In this case, three months dataset with four variables taken from MARS approach were used to predict the day-ahead price of the electricity market. Four variables taken from this approach as follows: PriceD, PriceD.6, P.DD.6, Wind speed. From using 4 variables in this approach, we can see that there is an increase in MAPE in almost all the hybrid models. Clearly, ARIMA-SVM outperforms other hybrid method in this case. It can be inferred that that this predictor variable from this variable selection approach does not help in reducing the MAPE.

Case-4: Applying Hybrid models to three months dataset using Random forest for 4 variables

In this case, three months dataset with four variables taken from Random forest approach were used to predict the day-ahead price of the electricity market. Four variables taken from this approach as follows: PriceD.6, P.DD.6, PriceD, C.PD.6. From using 4 variables in this approach, we can see that there is a reduction in MAPE in almost all the hybrid models than the previous case using MARS model. Clearly, ARIMA-GLM outperforms other hybrid method in this case. It can be inferred that that this predictor variables from this variable selection approach are better than MARS model but comes second in performance when compared to the relative importance approach.

Case-5: Applying Hybrid models to three months dataset using Relative importance approach for 2 variables

In this case, three months dataset with two variables taken from Relative importance approach were used to predict the day-ahead price of the electricity market. Two variables taken from this approach as follows: PriceD.6, Price D. For 2 variables, we can see that all hybrid models performed better than 17 & 4 variables taken from different approaches. Clearly, ARIMA & ARIMA-GLM outperforms other hybrid method in this case. Also, there was a significant reduction in MAPE in almost all the hybrid models in this case than using all the 17 & 4 variables.

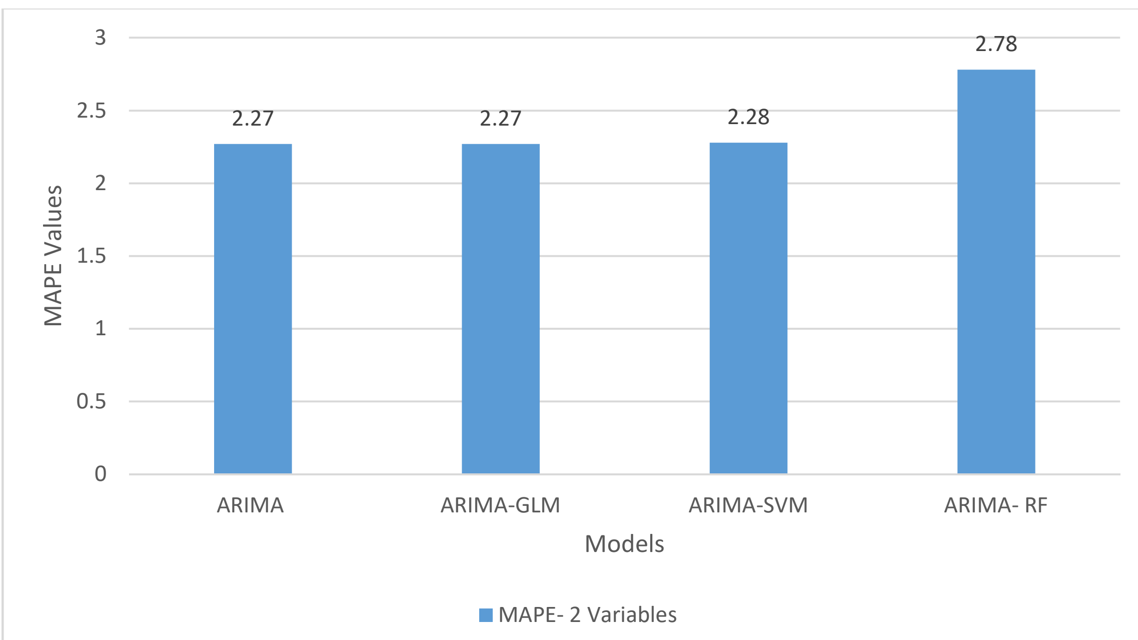

Figure 46: Comparison of MAPE for 90 days (May 01, 2015 to July 30 2015) for 2 variables selected from variable selection approaches to predict day-ahead price (July 31, 2015)

Case-6: Applying Hybrid models to three months dataset using Multivariate Adaptive Regression Splines (MARS) for 2 variables

In this case, three months dataset with two variables taken from Multivariate Adaptive Regression Splines (MARS) were used to predict the day-ahead price of the electricity market. Two variables taken from this approach as follows: PriceD.6, Price D. For 2 variables, we can see that all hybrid models performed better than 17 & 4 variables taken from different approaches. Clearly, ARIMA & ARIMA-GLM outperforms other hybrid method in this case. Also, there was a significant reduction in MAPE in almost all the hybrid models in this case than using all the 17 & 4 variables.

Case-7: Applying Hybrid models to three months dataset using Random forest for 2 variables

In this case, three months dataset with two variables taken from Random forest were used to predict the day-ahead price of the electricity market. Two variables taken from this approach as follows: PriceD.6, Price D. For 2 variables, we can see that all hybrid models performed better than 17 & 4 variables taken from different approaches. Clearly, ARIMA & ARIMA-GLM outperforms other hybrid method in this case. Also, there was a significant reduction in MAPE in almost all the hybrid models in this case than using all the 17 & 4 variables.

Table 20: Comparison of day-ahead forecasting performance of several hybrid models using feature selection approaches for six months dataset

Parameter	ARIMA	ARIMA-GLM	ARIMA-SVM	ARIMA- RF
MAPE-17 Variables	3.50	3.19	3.36	3.55
MAPE- 4 Variables (Relative Importance)	3.11	2.94	3.05	2.97
MAPE- 4 Variables (Multivariate Adaptive Regression Splines (MARS))	3.11	2.94	3.05	2.97
MAPE- 4 Variables (Random forest)	2.20	2.22	2.22	1.99
MAPE- 3 Variables	3.10	2.89	3.02	3.36
MAPE- 2 Variables (Relative Importance)	2.38	2.39	2.40	2.83
MAPE- 2 Variables (Multivariate Adaptive Regression Splines (MARS))	2.38	2.39	2.40	2.83
MAPE- 2 Variables (Random forest)	2.38	2.39	2.40	2.83

Case-8: Applying Hybrid models to six months dataset using all 17 variables

In this case, six months dataset with all 17 variables were used to predict the day-ahead price of the electricity market. 17 variables used in this case study were clearly mentioned in the chapter- 2. For 17 variables, we can see that all hybrid models performed better. Clearly, ARIMA-GLM outperforms other hybrid method in this case.

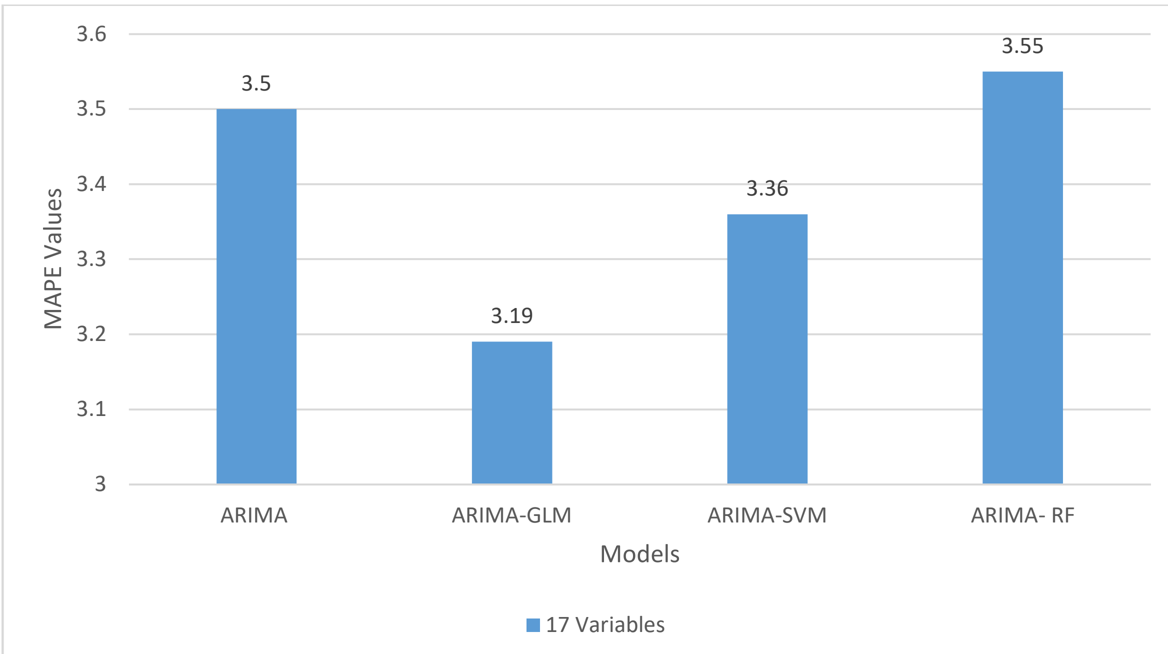

Figure 47: Comparison of MAPE for six months (Feb 01, 2015 to July 30 2015) for 17 variables to predict day-ahead price (July 31, 2015).

Case-9: Applying Hybrid models to six months dataset using Relative importance approach for 4 variable

In this case, six months dataset with four variables taken from Relative importance approach were used to predict the day-ahead price of the electricity market. Four variables taken from this approach as follows: PriceD.6, Price D, P.DD.6, C.GD.6. For 4 variables, we can see that all hybrid models performed better than using 17 variables. Clearly, ARIMA-GLM outperforms other hybrid method in this case. Also, there was a significant reduction in MAPE in almost all the hybrid models in this case than using all the 17 variables.

Case-10: Applying Hybrid models to six months dataset using Multivariate Adaptive Regression Splines (MARS) for 4 variables

In this case, six months dataset with four variables taken from MARS approach were used to predict the day-ahead price of the electricity market. Four variables taken from this approach as follows: PriceD.6, Price D, P.DD.6, C.GD.6. These variables are same as the previous set of variables taken from the relative importance method. For 4 variables, we can see that all hybrid models performed better than using 17 variables. Clearly, ARIMA-GLM outperforms other hybrid method in this case. Also, there was a significant reduction in MAPE in almost all the hybrid models in this case than using all the 17 variables.

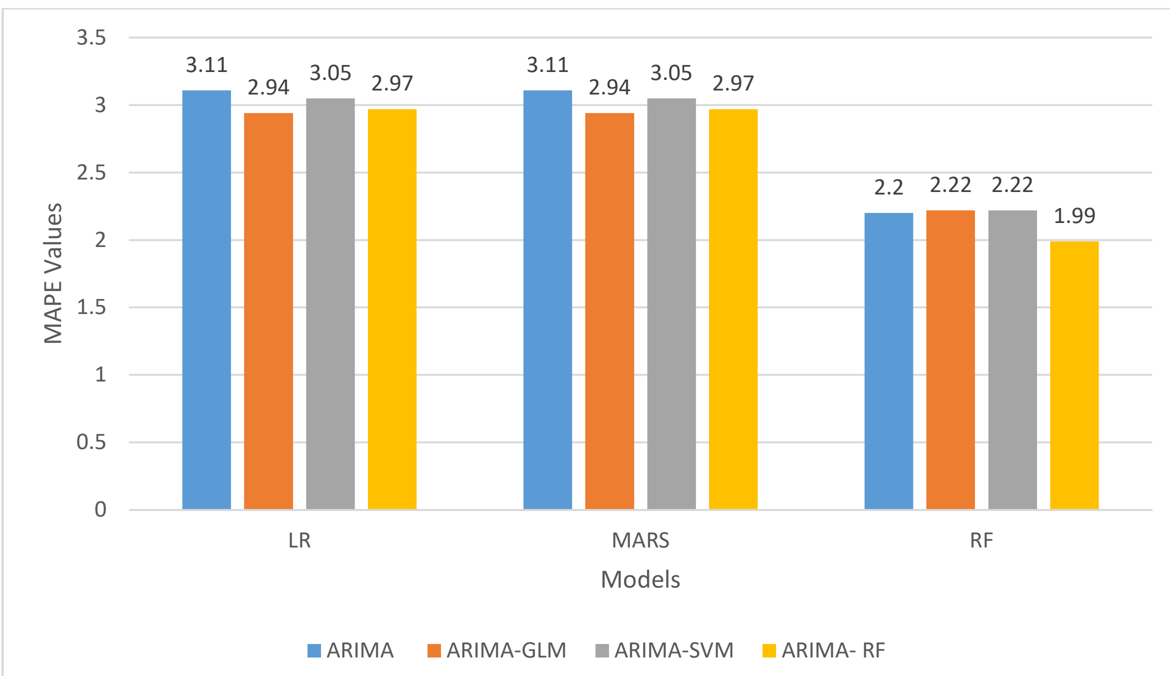

Figure 48: Comparison of MAPE for six months (Feb 01, 2015 to July 30 2015) for 4 variables selected from variable selection approaches to predict day-ahead price (July 31, 2015).

Case-11: Applying Hybrid models to six months dataset using Random forest for 4 variables

In this case, six months dataset with four variables taken from Random forest approach were used to predict the day-ahead price of the electricity market. Four variables taken from this approach are as follows: PriceD.6, PriceD, C.GD.6, and C.PD.1. For 4 variables, we can see that all hybrid models performed better than the previous set of 4 variables using Relative importance and MARS approach. Clearly, ARIMA-RF outperforms other hybrid method in this case. Also, there was a significant reduction in MAPE in almost all the hybrid models in this case than using all the 17 variables and other 4 variables.

Case-12: Applying Hybrid models to six months dataset using Relative importance approach for 2 variables

In this case, six months dataset with two variables taken from Relative importance approach were used to predict the day-ahead price of the electricity market. Two variables taken from this approach as follows: PriceD.6, Price D. For 2 variables, we can see that all hybrid models performed better than 17 & 4 variables taken from the same approach. Clearly, ARIMA & ARIMA-GLM outperforms other hybrid method in this case. Also, there was a significant reduction in MAPE in almost all the hybrid models in this case than using all the 17 & 4 variables from the same approach.

Case-13: Applying Hybrid models to six months dataset using Multivariate Adaptive Regression Splines (MARS) for 2 variables

In this case, six months dataset with two variables taken from Multivariate Adaptive Regression Splines (MARS) were used to predict the day-ahead price of the electricity market. Two variables taken from this approach as follows: PriceD.6, Price D. For 2 variables, we can see that all hybrid models performed better than 17 & 4 variables taken from the same approach. Clearly, ARIMA & ARIMA-GLM outperforms other hybrid method in this case. Also, there was a significant reduction in MAPE in almost all the hybrid models in this case than using all the 17 & 4 variables from the same approach.

Case-14: Applying Hybrid models to six months dataset using Random forest for 2 variables

In this case, six months dataset with two variables taken from Random forest were used to predict the day-ahead price of the electricity market. Two variables taken from this approach as follows: PriceD.6, Price D. For 2 variables, we can see that all hybrid models performed better than 17 variables but worse than 4 variables taken from the same approach. Clearly, ARIMA & ARIMA-GLM outperforms other hybrid method in this case.

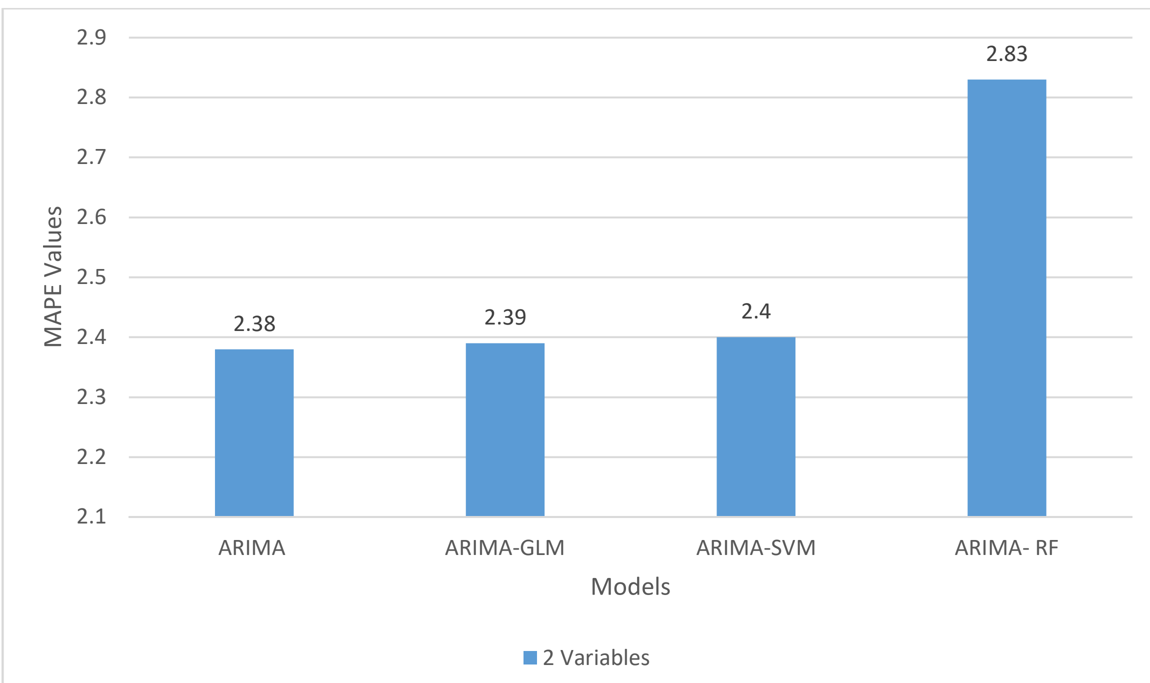

Figure 49: Comparison of MAPE for six months (Feb 01, 2015 to July 30 2015) for 2 variables selected from variable selection approaches to predict day-ahead price (July 31, 2015)

Case-15: Applying Hybrid models to three/six month's dataset using three common variables from variable selection approaches

In this case, three & six months dataset with three common variables selected from the variable selection approach were used to predict the day-ahead price of the electricity market. Clearly, ARIMA-GLM outperforms other hybrid method in this case.

Table 21: Comparison of MAPE for three/six month's dataset using three common variables from variable selection approaches

Parameter	ARIMA	ARIMA-GLM	ARIMA-SVM	ARIMA-RF	ARIMA-FFNN
MAPE- Three Months	3.07	2.85	3.008	3.23	5.03
MAPE- Six Months	3.10	2.89	3.02	3.36	3.23

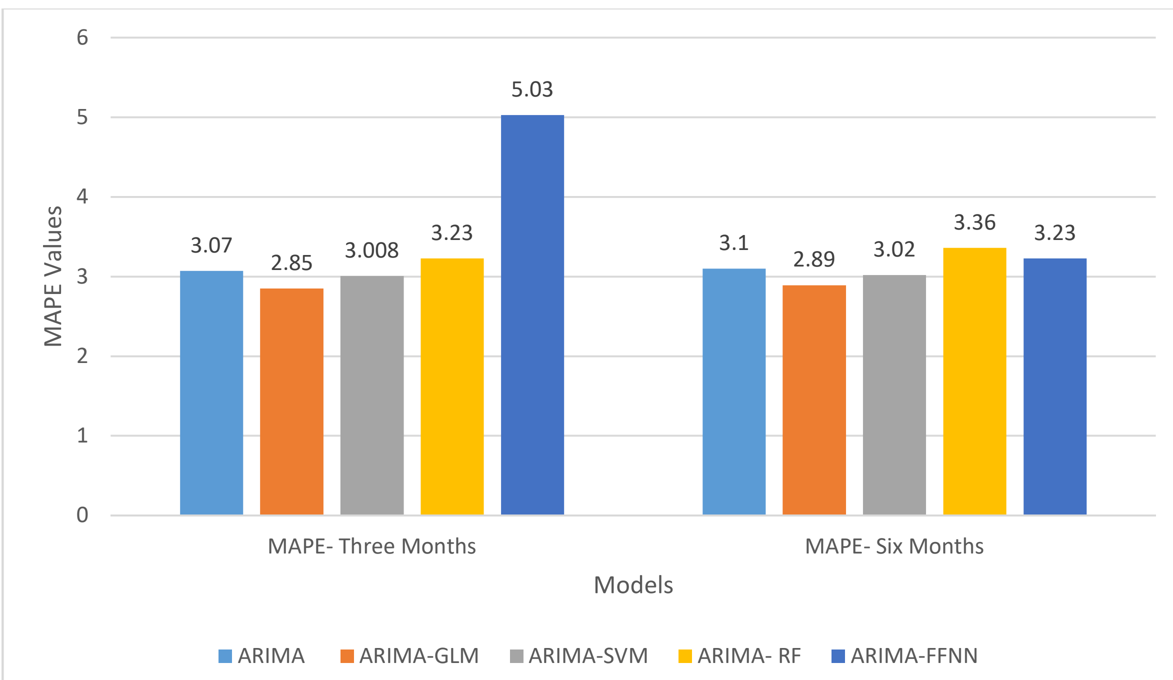

Figure 50: Comparison of MAPE for three & six months for 3 common variables selected from variable selection approaches to predict day-ahead price (July 31, 2015).

Case-16: Applying Hybrid models to Weekday/Weekend dataset using three common variables from variable selection approaches

In this case, three & six months dataset with three common variables selected from the variable selection approach were used to predict the day-ahead price of the electricity market. Clearly, ARIMA-FFNN outperforms other hybrid method in this case for weekday dataset while, ARIMA-SVM performs better for the weekend dataset.

Table 22: Comparison of MAPE for weekday/weekend dataset using three common variables from variable selection approaches

Parameter	ARIMA	ARIMA-GLM	ARIMA-SVM	ARIMA-RF	ARIMA-FFNN
MAPE- Weekday-	3.35	3.17	3.32	3.65	3.14
MAPE- Weekend – two var	7.2	7.15	7.11	7.52	7.34

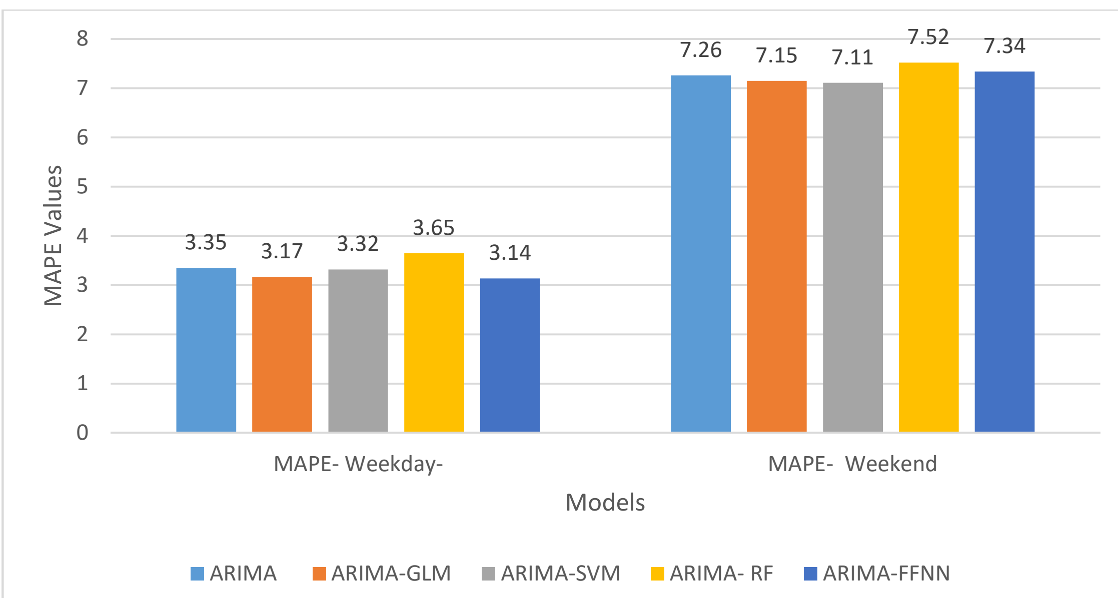

Figure 51: Comparison of MAPE for three & six months for 3 common variables selected from variable selection approaches to predict day-ahead price (July 31, 2015).

6 CONCLUSION AND FUTURE WORK

6.1 Conclusion

This work investigated a novel two-stage approach that combined the ARIMA model in Stage-1 and the resulting residuals as input to another forecasting method in Stage-2. The datasets used were drawn from the Iberian electricity markets. The results indicated a promising insight into the need for a focus on the residual improvement and training for forecasting the price markets.

In the first part of the research, existing forecasting algorithms were tested and a novel hybrid forecasting technique, namely ARIMA-GLM, was developed that performed well for the different duration of the dataset. In the second part of the research, state of the art tool named TensorFlow was explored to test the day-ahead price forecast. In the third part of the research, feature selection algorithms were explored to reduce the error considerably than using all the variables. The key findings of this research are as follows:

1. Developed a hybrid forecasting method named ARIMA-GLM that improved accuracies in the prediction of day-ahead price data for Iberian electricity market datasets. A novel, hybrid method of forecasting was developed which utilized the advantages of both ARIMA and GLM methods. Such hybrid method of forecasting showed significant improvements in the accuracy of predicting day-ahead price data for longer duration of the dataset. This ARIMA-GLM method was compared

with other existing forecasting methods like RF, SVM, ARIMA and other hybrid methods such as ARIMA-RF, ARIMA-SVM & ARIMA-LOWESS. For longer duration of the dataset such as for 60 days, 75 days, 90 days, 180 days, ARIMA-GLM performed better than the other hybrid methods.

2. The second part of this research investigated the performance of deep neural network models for the day-ahead price forecast of the Iberian electricity market. Rectified linear unit is used as an activation function in all these cases. In this case, three-months, six-month, weekday/weekend dataset with different sets of variables were used to predict the day-ahead price of the electricity market. Three different layers (2, 3, and 4) were used to understand the behavior of the model. For 17 variables, as the no of layers were increased in the model, it performed well for the complex relationship between the 17 independent and dependent variables. For 4 and 2 variables, as we increased the layers, inconsistencies were observed.
3. The third part of this research investigated the performance of feature selection models for the day-ahead price forecast of the Iberian electricity market. Feature selected using LR, MARS and random forest were applied to different hybrid models to predict the day-ahead price of the Iberian Energy Market Operator (MIBEL). Four datasets (Three months, Six months, weekday, and weekend) were used to validate the performance of the model. Three different set of variables (17, 4, 2) were used in this problem. At last, three common variables selected from these feature selection approaches were tested with all these datasets. MAPE was reduced considerably using 4, 3 and 2 variables selected from these feature selection approaches.

6.2 Future work

Future work would include testing and validating the results with larger data sets and investigation of the impacts on the MAPE values. For deep neural networks, different activation functions such as Exponential linear unit (ELU), sigmoid function and Leaky RELU can be applied to the same dataset to study the behavior of these models. Also, different feature selection algorithms to filter the best features that improves the predictive model.

REFERENCES

[1] J. Contreras, R. Espínola, S. Member, and F. J. Nogales, “ARIMA Models to Predict Next-Day Electricity Prices,” vol. 18, no. 3, pp. 1014–1020, 2003.

[2] R. Weron, “Electricity price forecasting: A review of the state-of-the-art with a look into the future,” *Int. J. Forecast.*, vol. 30, no. 4, pp. 1030–1081, 2014.

[3] “Electricity Market representation.” [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0169207014001083.

[4] N. V Karakatsani and D. W. Bunn, “Forecasting electricity prices : The impact of fundamentals and time-varying coefficients,” *Int. J. Forecast.*, vol. 24, no. 4, pp. 764–785, 2008.

[5] T. Jónsson, P. Pinson, H. A. Nielsen, H. Madsen, and T. S. Nielsen, “Forecasting Electricity Spot Prices Accounting for Wind Power Predictions,” vol. 4, no. 1, pp. 210–218, 2013.

[6] J. Luis, R. Espínola, A. Cruz, and A. Mu, “The effect of wind generation and weekday on Spanish electricity spot price forecasting,” vol. 81, pp. 1924–1935, 2011.

[7] Y. Dong, J. Wang, H. Jiang, and J. Wu, “Short-term electricity price forecast based on the improved hybrid model,” *Energy Convers. Manag.*, vol. 52, no. 8–9, pp. 2987–2995, 2011.

[8] C. R. Knittel and M. R. Roberts, “An empirical examination of restructured electricity prices,” vol. 27, pp. 791–817, 2005.

[9] A. Misiorek, “Forecasting spot electricity prices : A comparison of parametric and semiparametric time series models,” vol. 24, pp. 744–763, 2008.

[10] F. L. C. Mun, “Short-term forecasting of electricity prices in the Colombian electricity market,” vol. 3, no. April, pp. 980–986, 2009.

[11] J. Mei, D. He, R. Harley, T. Habetler, and G. Qu, “A random forest method for real-time price forecasting in New York electricity market,” in *2014 IEEE PES General Meeting | Conference & Exposition*, 2014, pp. 1–5.

[12] D. Sansom, T. Downs, and T. Saha, “Evaluation of support vector machine based forecasting tool in electricity price forecasting for Australian national electricity market participants,” *J. Electr. …*, vol. 22, no. 3, pp. 227–233, 2003.

[13] J. H. Zhao, Z. Y. Dong, Z. Xu, and K. P. Wong, “A statistical approach for interval

forecasting of the electricity price," *IEEE Trans. Power Syst.*, vol. 23, no. 2, pp. 267–276, 2008.

[14] J. Che and J. Wang, "Short-term electricity prices forecasting based on support vector regression and Auto-regressive integrated moving average modeling," *Energy Convers. Manag.*, vol. 51, no. 10, pp. 1911–1917, 2010.

[15] X. Yan and N. A. Chowdhury, "Mid-term electricity market clearing price forecasting: A hybrid LSSVM and ARMAX approach," *Int. J. Electr. Power Energy Syst.*, vol. 53, no. 1, pp. 20–26, 2013.

[16] L. Wu and M. Shahidehpour, "A Hybrid Model for Day-Ahead Price Forecasting Constants :," vol. 25, no. 3, pp. 1519–1530, 2010.

[17] T. Hansen, E. Chong, S. Suryanarayanan, A. Maciejewski, and H. Siegel, "A Partially Observable Markov Decision Process Approach to Residential Home Energy Management," *IEEE Trans. Smart Grid*, vol. 3053, no. JUNE, pp. 1–1, 2016.

[18] L. Wu and M. Shahidehpour, "A hybrid model for integrated day-ahead electricity price and load forecasting in smart grid," no. April, pp. 1937–1950, 2014.

[19] J. Mei, D. He, R. Harley, T. Habetler, and G. Qu, "A random forest method for real-time price forecasting in New York electricity market," 2014, pp. 1–5.

[20] L. Wang, Z. Zhang, and J. Chen, "Short-Term Electricity Price Forecasting with Stacked Denoising Autoencoders," *IEEE Trans. Power Syst.*, vol. 32, no. 4, pp. 2673–2681, 2017.

[21] Z. Qiu, "Mutivariable mutual information based feature selection for electricity price forecasting," *2012 Int. Conf. Mach. Learn. Cybern.*, pp. 168–173, 2012.

[22] A. Mohamed and M. E. El-Hawary, "Effective input features selection for electricity price forecasting," *Can. Conf. Electr. Comput. Eng.*, vol. 2016–Octob, 2016.

[23] O. Abedinia, N. Amjady, and H. Zareipour, "A New Feature Selection Technique for Load and Price Forecast of Electrical Power Systems," *IEEE Trans. Power Syst.*, vol. 32, no. 1, pp. 62–74, 2017.

[24] "ARIMA flowchart." [Online]. Available: https://www.analyticsvidhya.com/blog/2015/12/complete-tutorial-time-series-modeling/.

[25] "Support Vector Machine." [Online]. Available: http://dni-institute.in/blogs/building-predictive-model-using-svm-and-r/.

[26] "Random forest." [Online]. Available: https://medium.com/@williamkoehrsen/random-forest-simple-explanation-377895a60d2d.

[27] R. A. Chinnathambi, "Investigation of forecasting methods for the hourly spot price of the Day-Ahead Electric Power Markets," pp. 3079–3086, 2016.

[28] C. Monteiro, L. A. Fernandez-Jimenez, and I. J. Ramirez-Rosado, "Explanatory

information analysis for day-ahead price forecasting in the Iberian electricity market," *Energies*, vol. 8, no. 9, pp. 10464–10486, 2015.

[29] C. García-Martos, J. Rodríguez, and M. J. Sánchez, "Mixed models for short-run forecasting of electricity prices: Application for the Spanish market," *IEEE Trans. Power Syst.*, vol. 22, no. 2, pp. 544–552, 2007.

[30] J. P. S. Catal??o, S. J. P. S. Mariano, V. M. F. Mendes, and L. A. F. M. Ferreira, "Short-term electricity prices forecasting in a competitive market: A neural network approach," *Electr. Power Syst. Res.*, vol. 77, no. 10, pp. 1297–1304, 2007.

[31] A. T. Lora, J. M. R. Santos, A. G. Exposito, J. L. M. Ramos, and J. C. R. Santos, "Electricity Market Price Forecasting Based on Weighted Nearest Neighbors Techniques," *IEEE Trans. Power Syst.*, vol. 22, no. 3, pp. 1294–1301, 2007.

[32] A. J. Conejo, M. A. Plazas, R. Espinola, 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.

[33] N. Amjady, "Day-Ahead Price Forecasting of Electricity Markets by a New Fuzzy Neural Network," *IEEE Trans. Power Syst.*, vol. 21, no. 2, pp. 887–896, 2006.

[34] N. M. Pindoriya, S. N. Singh, and S. K. Singh, "An Adaptive Wavelet Neural Network-Based Energy Price Forecasting in Electricity Markets," *IEEE Trans. Power Syst.*, vol. 23, no. 3, pp. 1423–1432, 2008.

[35] J. P. S. Catal??o, H. M. I. Pousinho, and V. M. F. Mendes, "Neural networks and wavelet transform for short-term electricity prices forecasting," *2009 15th Int. Conf. Intell. Syst. Appl. to Power Syst. ISAP '09*, 2009.

[36] H. M. I. Pousinho, V. M. F. Mendes, and J. P. S. Catal??o, "Short-term electricity prices forecasting in a competitive market by a hybrid intelligent approach," *Energy Convers. Manag. J.*, vol. 39, no. 1, pp. 29–35, 2011.

[37] M. Shafie-Khah, M. P. Moghaddam, and M. K. Sheikh-El-Eslami, "Price forecasting of day-ahead electricity markets using a hybrid forecast method," *Energy Convers. Manag.*, vol. 52, no. 5, pp. 2165–2169, 2011.

[38] J. P. S. Catal??o, 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.

[39] N. Amjady and F. Keynia, "Day-Ahead Price Forecasting of Electricity Markets by Mutual Information Technique and Cascaded Neuro-Evolutionary Algorithm," vol. 24, no. 1, pp. 306–318, 2009.

[40] "Introduction to deep learning." [Online]. Available: https://machinelearningmastery.com/what-is-deep-learning/.

[41] "Comparison of deep learning with simple neural network." [Online]. Available: https://www.xenonstack.com/blog/data-science/log-analytics-with-deep-learning-and-machine-learning.

[42] "Model of a simple neuron." [Online]. Available: https://machinelearningmastery.com/neural-networks-crash-course/.

[43] "Model of a simple network." [Online]. Available: https://machinelearningmastery.com/neural-networks-crash-course/.

[44] T. Hossen, S. J. Plathottam, R. K. Angamuthu, P. Ranganathan, and H. Salehfar, "Short-Term Load Forecasting Using Deep Neural Networks (DNN)."

[45] "Introdcution to Feature Selection." [Online]. Available: https://machinelearningmastery.com/an-introduction-to-feature-selection/.

APPENDIX A

FORECASTING OF DAY-AHEAD PRICE

```
library(stats)
library("mgcv")
# Convert some factor variables to numeric (train and test sets)
# 90 days data july 2015 including temp variables
dataset <- read.csv("C:\\Users\\radhakrishnan.angamu\\Google Drive\\Flash Drive\\Forecasting
Competition\\COMPLATT DATA\\Historical Data\\XREG - 3 Months data (Temp).csv")
head(dataset)
names(dataset)
str(dataset)
#90d july 2015 inc temp variables
pastset  <- dataset[1:2184, ]
#NEW Training data
Yts=ts(pastset$Actual,start=c(2015,1),frequency=8760)
#     Yts=ts(Actual$value[1:2184],start=as.Date("5/1/2015",format="%m/%d/%Y"),frequency=24)
#8760
# Step 2.) Itialize model (arima,etc...)
require(forecast)
fit <- auto.arima(Yts)
fit
auto.arima(Yts, approximation=FALSE,trace=FALSE)
diagnostic=auto.arima(Yts)
diagnostic
# 90d  july 2015 inc temp variables
InitializedModel=arima(Yts,order=c(4,1,3),xreg=pastset[1:2184,c(3,4,5,6,7,8,9,10,11,12,13,14,15
,16,17,18,19)]) #add command xreg for multivariate
# 90d  july 2015 -four variables
InitializedModel=arima(Yts,order=c(4,1,3),xreg=pastset[1:2184,c(3,4,5,6)])  #add  command  xreg
for multivariate
# Make the prediction

#90d july 2015 inc temp variables
Prediction=forecast(InitializedModel,h=24,xreg=dataset[2185:2208,c(3,4,5,6,7,8,9,10,11,12,13,1
4,15,16,17,18,19)],method="CSS") #add xreg for multivariate
#90d  july 2015 -four variables
Prediction=forecast(InitializedModel,h=24,xreg=dataset[2185:2208,c(3,4,5,6)],method="CSS")
#add xreg for multivariate
#Create a data frame with the Residuals
residuals_Arima <- InitializedModel$residuals
#plot the graph
plot(Prediction)
Prediction$mean
plot(Prediction$mean)
```

```
f=Prediction$mean
plot(f)
# 90d data-july
arima.predjuly90daysinctemp =as.numeric(Prediction$mean)
# 90d data-july -four var
arima.predjuly90daysfourvar =as.numeric(Prediction$mean)
#MAPE
# 90djuly 2015 inc temp variables
mape=mean(abs(dataset$Actual[2185:2208] - f)/dataset$Actual[2185:2208])
mape
#Feeding Residuals to the GLM
dat_ts <- ts(residuals_Arima)
library(kernlab)
library(stats)
dat=as.numeric(dat_ts)
newpastset <-cbind(dat,pastset)
View(newpastset)
ArimaGLM.model <- glm(dat ~ PriceD + PriceD.6, family=gaussian,data = newpastset)

#90d july 2015 inc temp variables
ArimaGLM.model <- glm(dat ~ PriceD + PriceD.6 + P.DD.1 + P.DD.6 + H.GD.1
 + H.GD.6 + C.GD.1 + C.GD.6 + C.PD.1 + C.PD.6 + W.PD.1 + W.PD.6+ S.PD.1+ S.PD.6+ Temp+
irradiance+ windspeed, family=gaussian,data = newpastset)
#90d july 2015 -four var
ArimaGLM.model <- glm(dat ~ PriceD + PriceD.6 + P.DD.1 + P.DD.6, family=gaussian,data =
newpastset)
#forecast
# 90d july 2015 inc temp variables
forecastset  <- dataset[2185:2208,]
attach(forecastset)
#90d july 2015 incl temp variables
ARIMAGLM.pred<- predict(ArimaGLM.model, newdata=data.frame(PriceD , PriceD.6 , P.DD.1,
P.DD.6 , H.GD.1 , H.GD.6 , C.GD.1 , C.GD.6 , C.PD.1 , C.PD.6  , W.PD.1,W.PD.6, S.PD.1, S.PD.6,Temp,
irradiance, windspeed,data=forecastset))
#90d july 2015 -four var
ARIMAGLM.pred<- predict(ArimaGLM.model, newdata=data.frame(PriceD , PriceD.6 , P.DD.1,
P.DD.6,data=forecastset))
# 2 variables
svm.pred<- predict(ArimaGLM.model, newdata=data.frame(PriceD , PriceD.6 ,data=forecastset))
#final Adjusted Price data from two models
Hybridmodelprice = f + ARIMAGLM.pred
Hybridmodelprice
# 90d july 2015 inc temp variables
ARIMAGLMjuly90dtemp =as.numeric(Hybridmodelprice)
ARIMAGLMjuly90dtemp
# 90d july 2015 -four var
ARIMAGLMjuly90dfourvar =as.numeric(Hybridmodelprice)
ARIMAGLMjuly90dtemp
```

```
#MAPE
#90 d july 2015 inc  temp variables
mape=mean(abs(dataset$Actual[2185:2208] - Hybridmodelprice)/dataset$Actual[2185:2208])
mape
#ARIMA SVM
library(kernlab)
ARIMAsvm.model <- ksvm(dat ~ PriceD + PriceD.6 + P.DD.1 + P.DD.6 + H.GD.1
           + H.GD.6 + C.GD.1 + C.GD.6 + C.PD.1 + C.PD.6 + W.PD.1
           + W.PD.6+ S.PD.1+ S.PD.6+ Temp+ irradiance+ windspeed, data = newpastset, kernel =
"vanilladot")
# Four Var
ARIMAsvm.model <- ksvm(dat ~ PriceD + PriceD.6 + P.DD.1 + P.DD.6, data = newpastset, kernel =
"vanilladot")
forecastset  <- dataset[2185:2208,]
attach(forecastset)
ARIMASVM.pred<-  predict(ARIMAsvm.model,  newdata=data.frame(PriceD  ,  PriceD.6  ,  P.DD.1,
P.DD.6 , H.GD.1 , H.GD.6 , C.GD.1 , C.GD.6 , C.PD.1 , C.PD.6   , W.PD.1,W.PD.6, S.PD.1, S.PD.6,Temp,
irradiance, windspeed,data=forecastset))
# Four Var
ARIMASVM.pred<-  predict(ARIMAsvm.model,  newdata=data.frame(PriceD  ,  PriceD.6  ,  P.DD.1,
P.DD.6 ,data=forecastset))
Hybridmodelprice = f + ARIMASVM.pred
Hybridmodelprice
# 90 d july 2015 inc temp variables
ARIMASVMjuly90dtemp =as.numeric(Hybridmodelprice)
ARIMASVMjuly90dtemp
# 90 d july 2015 fourvar
ARIMASVMjuly90dfourvar =as.numeric(Hybridmodelprice)
ARIMASVMjuly90dfourvar
mape=mean(abs(dataset$Actual[2185:2208] - Hybridmodelprice)/dataset$Actual[2185:2208])
mape
#ARIMA RF
library(randomForest)
ARIMARF.model <- randomForest(dat ~ PriceD + PriceD.6 + P.DD.1 + P.DD.6 + H.GD.1
           + H.GD.6 + C.GD.1 + C.GD.6 + C.PD.1 + C.PD.6 + W.PD.1
           + W.PD.6+ S.PD.1+ S.PD.6+ Temp+ irradiance+ windspeed, data = newpastset,
           importance = TRUE, ntree=1000,mtry = 2)
# Four Var
ARIMARF.model <- randomForest(dat ~ PriceD + PriceD.6 + P.DD.1 + P.DD.6, data = newpastset,
                importance = TRUE, ntree=1000,mtry = 2)
forecastset  <- dataset[2185:2208,]
attach(forecastset)
ARIMARF.pred<- predict(ARIMARF.model, newdata=data.frame(PriceD , PriceD.6 , P.DD.1, P.DD.6
, H.GD.1 , H.GD.6 , C.GD.1 , C.GD.6 , C.PD.1 , C.PD.6    , W.PD.1,W.PD.6, S.PD.1, S.PD.6,Temp,
irradiance,   windspeed,data=forecastset))
# Four Var
ARIMARF.pred<- predict(ARIMARF.model, newdata=data.frame(PriceD , PriceD.6 , P.DD.1, P.DD.6
, H.GD.1,data=forecastset))
```

```
ARIMARF.pred
Hybridmodelprice = f + ARIMARF.pred
Hybridmodelprice
# 90d july 2015 inc temp variables
ARIMARFjuly90dtemp =as.numeric(Hybridmodelprice)
ARIMARFjuly90dtemp
# 90d july 2015 four var
ARIMARFjuly90dfourvar =as.numeric(Hybridmodelprice)
ARIMARFjuly90dfourvar

mape=mean(abs(dataset$Actual[2185:2208] - Hybridmodelprice)/dataset$Actual[2185:2208])
mape
# ARIMA LOWESS
ARIMAlowess.model <- loess(dat ~ PriceD + PriceD.6 + P.DD.1 + P.DD.6, data = newpastset)
ARIMAlowess.pred<-  predict(ARIMAlowess.model,  newdata=data.frame(PriceD  ,  PriceD.6  ,
P.DD.1, P.DD.6
                                        ,data=forecastset))
Hybridmodelprice = f + ARIMAlowess.pred
Hybridmodelprice
# 90d july 2015 inc temp variables
ARIMAlowessjuly90dfourvar =as.numeric(Hybridmodelprice)
mape=mean(abs(dataset$Actual[2185:2208] - Hybridmodelprice)/dataset$Actual[2185:2208])
mape
# RF SVM
library(randomForest)
pastset  <- dataset[1:168, ]
rf <- randomForest(Actual ~ PriceD + PriceD.6 + P.DD.1 + P.DD.6 + H.GD.1
              + H.GD.6 + C.GD.1 + C.GD.6 + C.PD.1 + C.PD.6 + W.PD.1
              + W.PD.6 + S.PD.1+ S.PD.6+ Temp+ irradiance+ windspeed,
              data = pastset, importance = TRUE, ntree=1000,mtry = 2)
RF.pred<- predict(rf, newdata=data.frame(PriceD , PriceD.6 , P.DD.1, P.DD.6 , H.GD.1
                              , H.GD.6 , C.GD.1 , C.GD.6 , C.PD.1 , C.PD.6
                              , W.PD.1 ,W.PD.6,S.PD.1,S.PD.6,Temp, irradiance,windspeed,
                              data=forecastset))
RF.pred
residuals_RF <- rf$residuals
dat_ts <- ts(residuals_RF)
dat=as.numeric(dat_ts)
newpastset<-cbind(dat,pastset)
View(newpastset)


RFSVM.model <- randomForest(dat ~ PriceD + PriceD.6 + P.DD.1 + P.DD.6 + H.GD.1
                  + H.GD.6 + C.GD.1 + C.GD.6 + C.PD.1 + C.PD.6 + W.PD.1
                  + W.PD.6+ S.PD.1+ S.PD.6+ Temp+ irradiance+ windspeed, data = newpastset,
                  importance = TRUE, ntree=1000,mtry = 2)
forecastset  <- dataset[169:192,]
attach(forecastset)
```

```
ARIMARF.pred<- predict(ARIMARF.model, newdata=data.frame(PriceD , PriceD.6 , P.DD.1, P.DD.6
, H.GD.1 , H.GD.6 , C.GD.1 , C.GD.6 , C.PD.1 , C.PD.6   , W.PD.1,W.PD.6, S.PD.1, S.PD.6,Temp,
irradiance, windspeed,data=forecastset))
ARIMARF.pred
Hybridmodelprice = f + ARIMARF.pred
Hybridmodelprice
# 1 week july 2015 inc temp variables
ARIMARFHybridmodelpricejulyoneweektemp =as.numeric(Hybridmodelprice)
ARIMARFHybridmodelpricejulyoneweektemp
mape=mean(abs(dataset$Actual[169:192] - Hybridmodelprice)/dataset$Actual[169:192])
mape
#Comparison of MAPE for 90d Dataset  with four var
plot(dataset$Actual[2185:2208],type="l",xlim=c(0,24),ylim=c(40,70),xlab="Hour",ylab
="Price(Euros)",col="blue")
#lines(hmodfile:///C:/Users/radhakrishnan.angamu/Google Drive/Flash Drive/Forecasting
Competition/Hybrid ARIMA -GLM.R,col="red", lwd=2)
lines(ARIMAGLMjuly90dfourvar, col="red", lwd=2)
lines(arima.predjuly90daysfourvar, col="green", lwd=2)
lines(ARIMASVMjuly90dfourvar, col="orange", lwd=2)
lines(ARIMARFjuly90dfourvar, col="black", lwd=2)
lines(ARIMAlowessjuly90dfourvar, col="brown", lwd=2)

legend('bottomright',  c("Actual","ARIMA=2.80","ARIMA-GLM=2.59","ARIMA-RF=3.12","ARIMA-
SVM=2.73","ARIMA-Lowess=3.11"),
    lty=1,title="MAPE", col=c('blue','green','red','black','orange','brown'), bty='n', cex=.75)
#one week Dataset
y = c(5.36,5.00,3.73,5.24)
barplot( y,main="Comparison of MAPE for One Week Dataset", xlab="Methods",
     ylab="MAPE", names.arg=c("ARIMA","AR-GLM","AR-SVM","AR-RF"),
     border="black
```