Power Load Forecast for North Macedonia Using Machine Learning

Pande Popovski, Mitko Kostov, Metodija Atanasovski and Goran Veljanovski

Abstract – **Estimating power load has always been of a great importance for making power purchase agreements on the electricity market or maintaining power system stability and reliability. This paper analyses power load forecasts for North Macedonia when different combinations of independent variables are used – air temperature, humidity and wind. Several types of machine learning models were trained in order to evaluate their performances and impact on the power load forecast.**

Keywords – **Machine learning, Regression, Power load forecast, Forecast accuracy, Model performance.**

I. Introduction

Estimating power load has always been a practice of key importance whether it comes to making power purchase agreements on the electricity market, or maintaining power system stability and reliability. With the increasing trend of renewable energy sources over the decades, new questions arise, one of them being inertia in the power systems. Having prior knowledge on electrical energy demand by means of power load forecasting can help to deal with the negative effects that come with the decrease of power system inertia, by minimizing the mismatch that can appear between supply and demand. This can be done through better production planning, which is greatly determined by the estimated increase in power demand. Furthermore, power load forecast can be used when it comes to minimizing power loss and coping with growing power demand: forecasting would determine which power units should increase their production and which generators should be dispatched [1]. Thereby the accuracy and reliability of an optimal calculated generation curve is determined by the reliability of the power load data which is used.

There is a number of factors that play an essential role when conducting power load forecasting - air temperature, day of the week, holidays, geographical differences, demographic information, people's standards of living, etc. [1].

A very close correlation between air temperature and power system load is studied for the Republic of North Macedonia, along with forecast accuracy using sinuses and polynomial regressions and discrete wavelet transformation [2]. In this paper, an analysis is made on the effect of air temperature, humidity and wind speed on the power load forecast accuracy: how a few machine learning models perform when different combinations of meteorological parameters are taken into consideration. Therefore, a contribution of the paper is that several types of machine learning models for the first time were trained on real data of North Macedonia power system in order to evaluate and compare their performances.

The paper is organized as follows. After the introduction, a brief explanation on the used machine learning algorithms is given. The third section shows the numerical results obtained from the trained models, while the fourth section concludes the paper.

Pande Popovski, Mitko Kostov, Metodija Atanasovski and Goran Veljanovski are with the Faculty of Technical Sciences-Bitola, St. Kliment Ohridski University, Republic of N. Macedonia, E-mail: pande.popovski@uklo.edu.mk

II. Algorithms

A. Decision Trees

This type of algorithms obtains its name from the decision tree (DT) diagram. Decision tree diagram is used to graphically present a process of making a decision. It shows the factors that play a role in the decision making, and conditions that have to be fulfilled, as well as all the possible solutions. DT algorithms apply the same concept to machine learning, however a different approach than simple decision making is used in order to construct a model that behaves like a DT. These algorithms partition the feature space into a set of rectangles, and then fit a simple model in each one [3]. The problem results in constructing the algorithm in a way so it could decide itself under what criteria to make the partitions.

A number of DT algorithms are common when dealing with classification and regression problems: ID3 (Iterative Dichotomiser 3), CART (Classification And Regression Tree), MARS (Multivariate Adaptive Regression Splines) and others [3].

There are also a number of methods used to deal with the drawbacks from using TDs. One of these methods is Bootstrap Aggregating (commonly called Bagging) [4], more specifically developed to reduce the negative impact that variance has on the model performance. In other words, it is implemented to reduce the sensitivity of a model to different starting parameters. Bagging method is composed of two parts: bootstrapping and aggregating [5]. Bootstrapping is a sampling method, where a sample is chosen out of a set, using the replacement method. Using sampling, multiple smaller datasets are formed from the original dataset, and a model is trained using each one. Then, the predictions made from these "weaker" models are aggregated to obtain a more accurate prediction:

978-1-7281-7142-5/20/$31.00 ©2020 IEEE

$$\hat{m}^{bag}(x) = \frac{1}{B}\sum_{b=1}^{B} \hat{m}_{S_b}(x), \quad (1)$$

where $\hat{m}_{S_b}$ is estimator of a model m: $X \rightarrow Y$, for a bootstrap sample S_b, which is obtained from the sample $S = \{y_i, x_i\}$ with $i = \{1, \ldots, n\}$.

Decision trees are sensitive to the specific data on which they are trained. Different subsets of the training data may result in different resulting decisions, and the predictions may be completely different. When bagging with decision trees, one is less concerned about individual trees overfitting the training data. The only parameter is the number of samples and consequently the number of trees to include. It can be selected by continuously increasing the number of trees until the accuracy starts to stop improving.

B. Support Vector Machines

Support-Vector Machines (SVMs, also called Support-Vector Networks) are supervised learning models with associated learning algorithms that analyse data used for classification and regression analysis [6]. The support-vector network works in such a way that an input training data (support vectors) is mapped into a high dimensional feature space, and in this space a decision surface is constructed with special properties that ensure high generalization ability of the network [7]. In other words, the algorithm constructs a hyperplane, a subspace whose dimension is one less than that of its ambient space, by using the mapped data from the training dataset [8]. This hyperplane (or set of hyperplanes) acts as a sort of a border that divides data points, which is why SVMs were primarily used for classification problems. Once a model has been trained, further instances added in the created space can be classified using the hyperplane. However, modified SVMs can be used for regression analysis as well.

In their most general form, SVMs [9] find a hyperplane in a space different from that of the input data. It is a hyperplane in a feature space induced by a kernel K. In other words, because most of the time a problem does not contain linearly separable data, it is presented in a higher dimensional space so its data may become linearly separable. This is often called the kernel trick [10]. This can be seen as a set of functions in a Reproducing Kernel Hilbert Space (RKHS) [9].

For regression purposes SVMs solve the following minimization problem:

$$\min_f \|f\|_K^2 + C\sum_{i=1}^{l} |y_i - f(x_i)|_\varepsilon \quad (2)$$

where C is a "regularization parameter" that controls the trade-off between empirical error and complexity of the hypothesis space used. The variables x and y belong to a set of training data containing l instances. The function f represents a hyperplane, expressed via the input data x. K represents the kernel that defines the RKHS, and $\|f\|_K^2$ is the RKHS norm of the function f.

C. Gaussian Process Regression

Gaussian processes (GPs) can be thought of as an extension of multivariate Gaussian distribution. GPs fall in the category commonly known as Bayesian methods, because of the fact that they originate from the Bayes' theorem [11].

The Gaussian distribution is very useful in machine learning because it can conveniently model noise, which is a very common occurrence in data. Also, they are convenient for many analytical manipulations. Because of this, Gaussian distributions are useful for modelling finite collections of real-valued variables. GPs are the extension of multivariate Gaussians to infinite-sized collections of real-valued variables. This extension defines GPs as distributions not just over random vectors, but also over random functions [11].

Following the assumption that data can be represented as a sample from a multivariate Gaussian distribution, the following expression is given [12]:

$$\begin{bmatrix} \mathrm{y} \\ y_* \end{bmatrix} \sim N\left(0, \begin{bmatrix} K & K_*^T \\ K_* & K_{**} \end{bmatrix}\right), \quad (3)$$

where tilde (~) denotes "has the probability distribution of", T indicates matrix transposition. Matrices K, K_* and K_{**} contain the values of a covariance function for all possible combinations of data points, due to the variance of the responses differing:

$$K = \begin{bmatrix} k(x_1,x_1) & k(x_1,x_2) & \cdots & k(x_1,x_n) \\ k(x_2,x_1) & k(x_2,x_2) & \cdots & k(x_2,x_n) \\ \vdots & \vdots & \ddots & \vdots \\ k(x_n,x_1) & k(x_n,x_2) & \cdots & k(x_n,x_n) \end{bmatrix},$$

$$K_* = \begin{bmatrix} k(x_*,x_1) & k(x_*,x_2) & \cdots & k(x_*,x_n) \end{bmatrix},$$

$$K_{**} = \begin{bmatrix} k(x_*,x_*) \end{bmatrix},$$

$$k(x,x') = \sigma_f^2 e^{\left[\frac{-(x-x')^2}{2l^2}\right]},$$

with l and δ - a length parameter and the Kronecker delta function, respectively. The goal is, given n observations y, to predict y_*. The conditional probability $p(y_*|\mathrm{y})$ follows a Gaussian distribution y:

$$y_* \mid \mathrm{y} \sim N\left(K_* K^{-1}\mathrm{y},\ \ K_{**} - K_* K^{-1} K_*^T\right). \quad (4)$$

The best estimate for y_* is the mean of this distribution:

$$\bar{y}_* = K_* K^{-1}\mathrm{y}, \quad (5)$$

and the uncertainty in the estimate is captured in its variance:

$$\mathrm{var}(y_*) = K_{**} - K_* K^{-1} K_*^T. \quad (6)$$

TABLE I
PERFORMANCE RESULTS FROM TRAINING DATASET

Attributes:	Measures	Bagged trees	Radial SVM	Exponential GPR
Temperature	R	0,93	0,91	0,93
	RMSE [MW]	59,29	67,48	59,34
	MSE [MW]	3515,54	4553,07	3521,28
	MAE [MW]	45,78	54,69	46,11
Temperature, Wind	R	0,92	0,91	0,91
	RMSE [MW]	63,78	66,43	66,21
	MSE [MW]	4068,10	4412,28	4384,16
	MAE [MW]	50,73	53,37	53,55
Temperature, Humidity	R	0,92	0,91	0,91
	RMSE [MW]	63,28	66,72	66,30
	MSE [MW]	4004,26	4451,58	4396,12
	MAE [MW]	50,37	53,95	53,82
Temperature, Humidity, Wind	R	0,91	0,91	0,91
	RMSE [MW]	66,75	65,67	65,54
	MSE [MW]	4456,20	4311,93	4295,65
	MAE [MW]	53,16	52,70	52,72

III. CASE STUDY

The entirety of the collected data is made up of meteorological documentation and power load data. More specifically the meteorological data consists of the daily averages of the air temperature (expressed in degrees Celsius), air humidity (expressed in *%*), and wind speed (expressed in mph-miles per hour) for the city of Skopje, North Macedonia throughout the years 2014-2019 [13]. For the same time period, the daily averages of the power load in the Republic of North Macedonia are provided [14].

The data is divided into a training dataset, which consists of the statistical information for the period 2014-2018 (1826 instances), and the test dataset with the data for the year 2019 (365 instances). The performance of the trained models for the training dataset is evaluated using 10-fold cross-validation.

Multiple models were trained using Linear Regression, Regression Trees, Support Vector Machines, Gaussian Process Regression and Ensembles of Trees. Different combinations of the meteorological factors were used as attributes. In each of the models, the average daily power load remained as the dependant variable. More specifically, four cases were taken into consideration:

- Attributes: *Temperature*; Dependant variable: *Power load*
- Attributes: *Temperature and wind*; Dependant variable: *Power load*
- Attributes: *Temperature and humidity*; Dependant variable: *Power load*
- Attributes: *Temperature, humidity and wind*; Dependant variable: *Power load.*

The following three machine learning algorithms generated the most accurate models: Ensemble of trees (Bootstrap Aggregation, or *Bagged trees*), Support Vector Machine (*Radial kernel SVM*) and Gaussian Process Regression (*Exponential kernel GPR*). The measures used to evaluate model performance are the following [15]:

Mean-squared error (MSE):

$$MSE = \frac{(y_{1*} - a_1)^2 + \cdots + (y_{n*} - a_n)^2}{n}. \quad (7)$$

Mean-absolute error (MAE):

$$MAE = \frac{|y_{1*} - a_1| + \cdots + |y_{n*} - a_n|}{n}. \quad (8)$$

Root mean-squared error (RMSE):

$$RMSE = \sqrt{\frac{(y_{1*} - a_1)^2 + \cdots + (y_{n*} - a_n)^2}{n}}. \quad (9)$$

Correlation coefficient (*R*):

$$R = \frac{S_{YA}}{\sqrt{S_Y S_A}}. \quad (10)$$

where:

$$S_{YA} = \frac{\sum_{i=1}^{n}(y_i - \bar{y})(a_i - \bar{a})}{n-1},\ S_Y = \frac{\sum_{i=1}^{n}(y_{i*} - \bar{y})^2}{n-1},\ S_A = \frac{\sum_{i=1}^{n}(a_i - \bar{a})^2}{n-1}.$$

In Eqs. (6)-(9), n represents the total number of instances, $y_{1*},..,y_{n*}$ are the predicted response values, $a_1,..,a_n$ are the actual response values, and $\bar{y}_*$ and $\bar{a}$ are their average values.

Table I shows the performance results of the trained models when the training dataset (years 2014-2018) is tested by 10-fold cross-validation. By observing the acquired numerical results from the cross-validation, it can be concluded that the performance of the trained models does not change noticeably by adding humidity and wind speed as attributes alongside air temperature. In fact, slightly worse results are obtained for the models trained with the *Bagged trees* and *Exponential kernel GPR* algorithms. A minor improvement in forecast accuracy is noted for the model trained with *Radial kernel SVM*. However, this applies for the training dataset, as different results might be acquired from the test dataset.

TABLE II
PERFORMANCE RESULTS FROM TEST DATASET

Attributes	Measures	Bagged trees	Radial SVM	Exponen t ial GPR
Temperature	R	0,93	0,95	0,94
	RMSE [MW]	67,41	56,23	64,49
	MSE [MW]	4543,66	3162,17	4158,66
	MAE [MW]	51,30	41,54	47,80
Temperature, Wind	R	0,94	0,94	0,95
	RMSE [MW]	59,32	58,97	57,24
	MSE [MW]	3518,44	3478,35	3276,04
	MAE [MW]	45,14	43,64	41,83
Temperature, humidity	R	0,94	0,95	0,95
	RMSE [MW]	58,71	55,36	54,77
	MSE [MW]	3447,72	3064,59	3000,08
	MAE [MW]	43,63	41,44	40,83
Temperature, Humidity, Wind	R	0,94	0,95	0,95
	RMSE [MW]	61,69	55,57	55,77
	MSE [MW]	3805,37	3088,48	3109,75
	MAE [MW]	46,84	41,74	41,66

Performance results obtained for the test dataset (year 2019) are given in Table II. Results from the test dataset show that when only the air temperature has been chosen as an independent variable, prediction accuracy for the models trained with *Bagged trees* and *Exponential GPR* is worse compared to the training dataset, whereas there is an improvement for the model trained with *Radial kernel SVM*. When multiple independent variables are considered, an improvement in forecast accuracy is noted in each model.

Examining the test results further, it can be noticed that trained models perform overall better when humidity is taken as an attribute along with temperature, compared to when wind and temperature are taken as attributes. This leads to the consideration that humidity may be more closely correlated to power load than wind speed is. When all three attributes are used to train a model, very slight changes in accuracy can be noticed compared to the case when two attributes are used.

IV. CONCLUSION

From several machine learning algorithms used for power load forecast, the following three are presented in the paper: *Bagged trees*, *Radial kernel SVM* and *Exponential kernel GPR*. Using each of the three algorithms, a number of models are trained where different combinations of meteorological factors are used.

From the results it can be concluded that combining the attributes improves its accuracy. From the four cases, the most accurate results are achieved when *air temperature* and *humidity* are used as independent variables. The models trained with *Exponential kernel GPR* gain the most accuracy from using multiple attributes, while for the models trained with *Radial kernel SVM* the least improvement is noted. However, in general the models trained with *Radial kernel SVM* give the most accurate results.

ACKNOWLEDGEMENT

This research is supported by the EU H2020 project TRINITY (Grant Agreement no. 863874) This paper reflects only the author's views and neither the Agency nor the Commission are responsible for any use that may be made of the information contained therein.

REFERENCES

[1] E. A. Feinberg, D. Genethliou, "Chapter 12 Load Forecasting", *Applied Mathematics for Power Systems*, pp.269-282.
[2] M. Kostov, M. Atanasovski, G. Janevska, and B. Arapinovski, "Power System Load Forecasting by using Sinuses Approximation and Wavelet Transform", ICEST - Ohrid, North Macedonia, 27-29 June 2019.
[3] T. Hastie, R. Tibshirani, and J. Friedman, "The Elements of Statistical Learning: Data Mining, Inference, and Prediction" - Second EDITION, pp. 305-306, *Springer*, May 2001.
[4] "Bootstrap Aggregation, Random Forests and Boosted Trees", https://www.quantstart.com/articles/bootstrap-aggregation-random-forests-and-boosted-trees.
[5] L. Breiman, "Bagging Predictors", *Machine Learning*, vol. 24, no. 2, pp. 123-140, 1996.
[6] https://en.wikipedia.org/wiki/Support_vector_machine.
[7] C. Cortes, V. Vapnik, "Support-Vector Networks", Machine Learning 20, pp. 273-297, 1995, https://doi.org/10.1007/BF00994018.
[8] https://en.wikipedia.org/wiki/Hyperplane.
[9] T. Evgeniou, M. Pontil, "Support Vector Machines: Theory and Applications", chapter from book *Machine Learning and Its Applications: Advanced Lectures*, pp 249-257, 2001.
[10] https://en.wikipedia.org/wiki/Bayes%27_theorem
[11] C. B. Do, H. Lee, "Gaussian Processes", lecture notes, November 22, 2008.
[12] M. Ebden, "Gaussian Processes for Regression: A Quick Introduction", August 2008.
[13] https://www.wunderground.com/history.
[14] *Dispatching and Operational Reports for Power System Load Data* obtained from MEPSO.
[15] I. H. Witten, E. Frank, M. A. Hall, C. J. Pal, *Data Mining: Practical Machine Learning Tools and Techniques*, Burlington, USA, Morgan Kaufmann, 2017.