Forecasting the prices of the day-ahead electricity markets using real data from SEEPEX

Pavle Petkovski
Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University
Skopje, North Macedonia
pavle.petkovski@students.finki.ukim.mk

Aleksandra Dedinec
Faculty of Computer Science and Engineering
Ss. Cyril and Methodius University
Skopje, North Macedonia
aleksandra.kanevche@finki.ukim.mk

***Abstract*—Forecasting the day-ahead electricity prices can be significant for every business involved in the electricity market. In this paper, we compare different machine learning techniques and algorithms using real data from Serbian Power Exchange, weather data from Serbian capital city Belgrade and generation per production type data for Serbian electricity production. Then on this data, we train different machine learning models: Linear Regression, Decision Trees, Support Vector Machines, Random Forest models, Extreme Gradient Boosting models, Deep Learning models. Metric that we used for comparison between models is the coefficient of determination.**

***Index Terms*—electricity, price forecasting, machine learning, deep learning**

I. Introduction

The importance of forecasting prices comes from the volatile nature of electricity production and consumption. Electricity cannot be stored as easily as gas, it is produced at the exact moment of demand. The purpose of this research is to try to predict electricity prices for day-ahead using historical exchange data and other useful data. The electricity price can be affected by many factors, some of which are historical price, volume, day of a week, season, is it a holiday, hour of the day. With the increasing usage of renewable energy sources, the effect of weather conditions such as wind, rain, sunlight on electricity prices also increases.

With the rise of popularity of machine learning there had been rise of interest to research electricity price forecasting in the last few years. The paper [1] compares common statistical method Lasso Estimated Auto Regressive(LEAR) with deep neural networks on various power exchanges historical data. Hybrid methods are gaining popularity one of them is [2] which combines Deep belief networks, Long short-term memory and Convolutional neural networks to make a powerful forecasting model. These papers, use data for markets that exist for many years and are mostly located in developed countries.

The data used for this research is gathered from the Serbian Power Exchange containing hourly prices and volumes from 2016-02-18 to 2021-08-16, historical weather data for Belgrade in the same period and historical electricity production data for Serbia from Enstso-e transparency platform. The Power Exchange works in a way that during the day based on the traders bids the prices for next day (from 00:00 to 23:00) is set. The goal of this research is to predict prices for all 24 hours. SEEPEX data and weather data were gathered using web scrapping tools more precisely Selenium WebDriver and BeautifulSoup Python library and production data was freely available on official site [3]. The weather data for Belgrade contains temperature, air pressure, wind speed and visibility. Electricity production data contain hourly production from biomass, fossil brown coal (lignite), hydro pumped storage, hydro run of the river and poundage, hydro water reservoir and other resources. With all these parameters, machine learning algorithms should be able to spot patterns that affect electricity prices assuming our data has insignificant noise.

The paper is structured as follows. First analysis of the data set is presented, than a description of the methodology used is given, followed by the results and corresponding discussion and conclusion.

II. Data set analysis

Using statistical tools and visualisations of the gathered data it can be seen that most of the assumptions made previously in this paper are to some degree true. First, we have analysed the average prices at monthly level (Figure 1). Knowing that January is the coldest month it is expected that electricity prices are highest and in summer electricity is needed for cooling so a spike in July and August proves that the assumption is true. The second assumption that the day of the week affects the electricity prices can be justified from Figure 2. The graph plot shows that weekend prices decrease. Significant difference between prices can be seen if a day is a holiday or not (Figure 3). Last visualisation and maybe most valuable for this research is the average price in different hours of the day. Figure 4 shows the price pattern during 24 hours of the day from which it can be seen that morning prices are lowest and the peak is at 19:00.

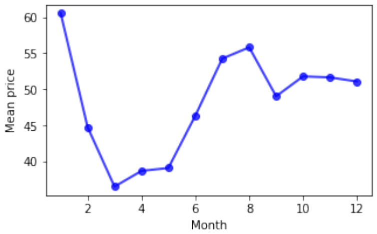

Fig. 1. Visualisation of prices over Months

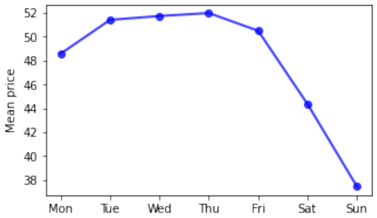

Fig. 2. Visualisation of prices over Days of Week

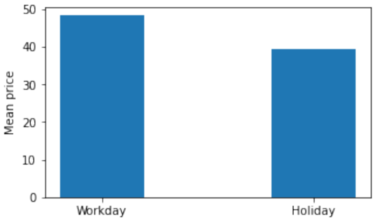

Fig. 3. Electricity prices compared over Workday and Holiday

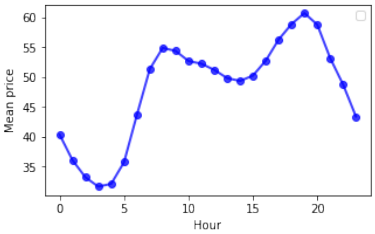

Fig. 4. Visualisation of prices over Hours of the Day

III. Methodology

Methods used for this research are:

- Linear Regression - widely used statistical method which presents response as a linear function of the inputs using the following equation:

$$y(x) = w^T x + \epsilon = \sum_{j=1}^{D} w_j x_j + \epsilon \tag{1}$$

  where the $w^T x$ represents the inner or scalar product between the input vector x and the model's weight vector w^T, ϵ is the residual error between our linear predictions and the true response and y is electricity price. The goal is to minimise ϵ for given data.
- Decision Trees or Classification and regression trees - defined by recursively partitioning the input space, and defining a local model in each resulting region of input space. This can be represented by a tree, with one leaf per region. It may seem that this approach is not suitable for the regression task but if we get mean value for each region this method can perform decently.
- Support Vector Machines - Support Vector Regression used for solving regression problems means solving

$$minimise(1/2||w||^2) \text{ subject to } y_i - \langle w, x_i \rangle - b \le \epsilon \tag{2}$$

  where x_i is a training sample with target value y_i, The inner product plus intercept $\langle w, x_i \rangle - b$ is the prediction for that sample and ϵ is a free parameter that serves as a threshold, all predictions have to be within an ϵ range of the true predictions.
- Random Forest - technique used to reduce variance. The idea is to train M different decision trees on different subsets of data chosen randomly with replacement, and then compute the ensemble

$$f(x) = \sum_{m=1}^{M} \frac{1}{M} f_m(x) \tag{3}$$

  where f_m is the m'th tree.
- Extreme Gradient Boosting (XGBoost) - extremely popular algorithm and was go to algorithm for machine learning competitions until the rise of Deep Learning. It improves the Boosting and Gradient Boosting algorithms. The idea behind boosting is to combine lots of weak learners or classifiers to create strong regressor. At the beginning of training all training samples have same weight but, as training weak learners continues weight of samples that are correctly predicted decreases and weight of incorrect ones increases, in most cases this results lower bias and lower variance. Gradient boosting uses a gradient descent algorithm to minimize error using following equations:

$$w = w + \eta \nabla w$$

$$\nabla w = \frac{\partial L}{\partial w} \tag{4}$$

  where L is loss function, w is weight vector and η is learning rate. Improvements that XGBoost brings to Gradient Boost are Tree Pruning, Sparsity Aware Split Finding, as well as computational improvements like parallelization and cache aware optimisation.
- Deep Neural Networks (DNN) - more precisely Feed Forward Neural Networks which moves information only to next layer not to layer behind or to nodes at same layer,this can be represented as directed acyclic graph which has one input layer, one or several hidden layers and one output layer. Between each node from neighboring layers there is weight that affects the output. Using BackPropagation Algorithm DNN model adjusts the weights to fit the training set. The power of DNN comes from the ability to learn non-linear patterns and for that is used in a wide variety of problems including price forecasting.The relationship between input x and output y is presented by this equation:

$$y = \sum_{j=0}^{h} \left[w_j f\left(\sum_{i=0}^{d} w_{ij} x_i \right) \right] \tag{5}$$

  where w_i and w_{ij} represent the weight and biases that connect the layers.

  For loss function Mean absolute error (MAE) was used which can be calculated by the following equation:

$$MAE = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n} \tag{6}$$

  where y_i is true value and x_i is predicted value.

  During training there was some overfitting present and because of that Lasso Regularisation (L1 Regularization) was used which penalizes large values for weights.

Coefficient of determination (R^2) was used as metric for comparison of machine learning models used in this research. R^2 is a measure of the goodness of fit of a model. R^2 is calculated using the following equation:

$$R^2 = 1 - \frac{\sum_{i=1}^{n} (y_i - f_i)^2}{\sum_{i=1}^{n} (y_i - \overline{y})^2} \tag{7}$$

where f_i is predicted value, y_i is true value. An R^2 of 1 indicates that the regression predictions perfectly fit the data.

A. Inputs

After experimenting with lots of different input variables, the following ones were selected:

- Hour of the day
- Day of the week
- Month
- Holiday flag
- Volume
- Average electricity price for previous day
- Electricity price for previous seven days in exact hour
- Daily average temperature
- Daily maximal temperature
- Daily minimal temperature
- Daily average pressure
- Daily average wind speed

IV. Results and Discussion

The initial data sets were divided into two parts, one for training and one for test. Training data set contained values for the period from 25-02-2016 to 25-12-2019 and test set contained values for the period from 26-12-2019 to 16-08-2021 meaning first 70% of data were used for training and remaining 30% for testing. On this test set all models were tested. The results after training, fine tuning and testing are the following:

- Linear regression model got R^2 of 0.834 on training set and 0.870 on test set. High R^2 value from the test set comes from predicting peak at the end of test period which few of next models did. (Fig. 5)
- Decision tree regressor got R^2 of 0.859 on training set and 0.833 on test set. This model had problems with generalization, it would get high R^2 score on training set but low R^2 score on test. This problem was solved using maximal depth parameter which prevents the tree from overfitting on the training set. (Fig. 6)
- Support vector machine training was time consuming and with that very hard to fine tune the parameters. On training set R^2 was good but, on test set was surprisingly low with maximal R^2 at 0.647. There was no improvement with different kernels and different ”C” values hardly affected anything.
- Random forest regressor got R^2 of 0.902 on training set and 0.863 on test set. For fine tuning there a was trade off between number of estimators and max depth and in the end this model did very well. (Fig. 7)
- Extreme gradient boost had the highest R^2 of 0.938 on training set and decent score of 0.867 on test set. Fine tuning was mainly between number of estimators and maximal tree depth as well as learning rate. (Fig. 8)
- Deep Neural Network model got R^2 of 0.871 on training set and on 0.882 test set with 9 hidden layers and 691 nodes. In fine tuning main challenge was to choose the number of layers and number of nodes. With higher number of layers there was increase in R^2 training score, but decrease in R^2 test score. Improvement was made using L1 regularization, which highly improved generalization. (Fig. 9)

  It is important to note that great increase was made by adding weather information to data set, almost all model got 0.01 to 0.03 R^2 score boost. (Fig. 10)

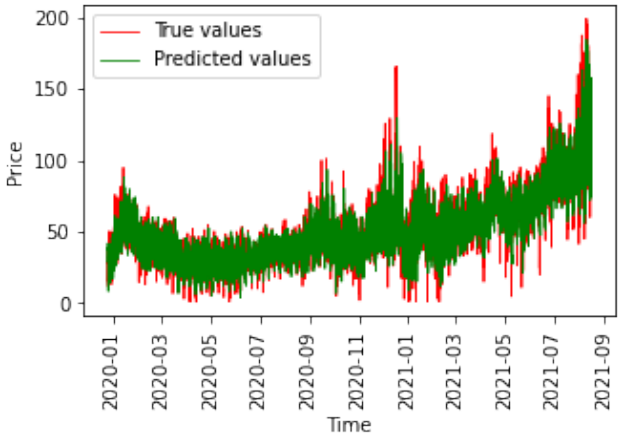

Fig. 5. Comparison between actual and predicted electricity prices using Linear Regression on test data set

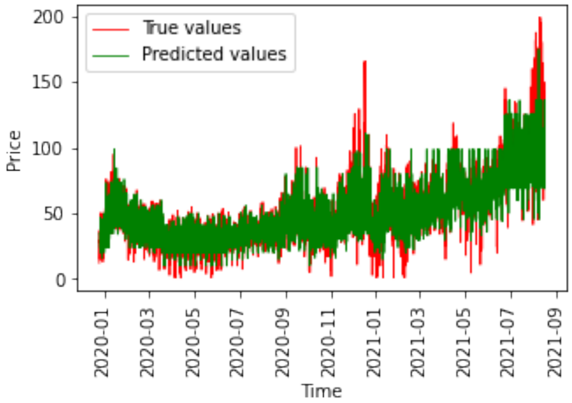

Fig. 6. Comparison between actual and predicted electricity prices using Decision Tree on test data set

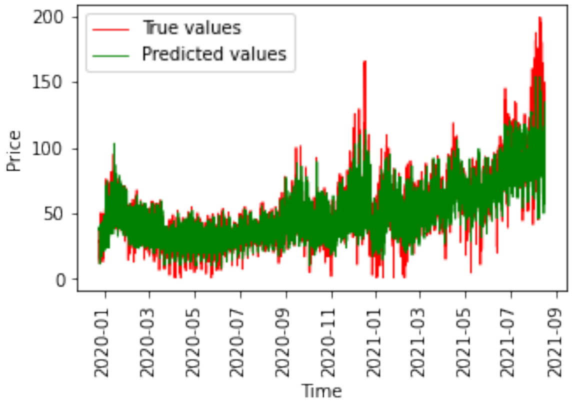

Fig. 7. Comparison between actual and predicted electricity prices using Random Forest on test data set

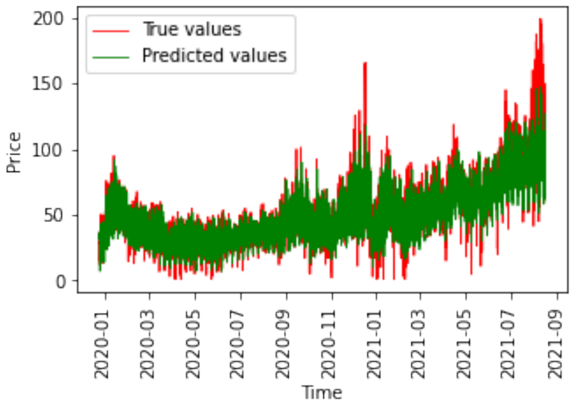

Fig. 8. Comparison between actual and predicted electricity prices using Extreme Gradient Boosting on test data set

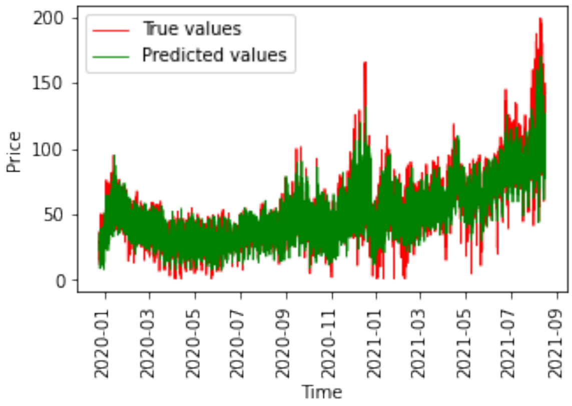

Fig. 9. Comparison between actual and predicted electricity prices using Deep Neural Network on test data set

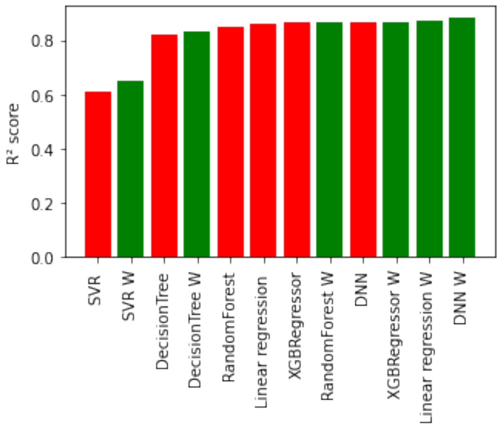

Fig. 10. Comparison between all models made in this research by R^2 score on test set, red bar indicates that model was trained without weather data

V. Conclusion

Using historical price data and weather data all of the mentioned machine learning models were able to predict the electricity prices. Differences between them still exist and from the obtained results, the most adequate are Deep Neural Networks and Extreme Gradient Boosting. Using the electricity production data by generating unit from ENTSO-E did not improve the results as much and in some situations can act as a noise. The reasons for this are still to be analyzed, including the analyzes of the completeness of the data. It would be very interesting in the future to see these methods used on markets where most of the traded electricity is produced from renewable sources.

Acknowledgment

This work was part of my internship in the Macedonian Academy of Sciences and Arts.

References

[1] Jesus Lago, Grzegorz Marcjasz, Bart De Schutter, Rafał Weron, Forecasting day-ahead electricity prices: A review of state-of-the-art algorithms, best practices and an open-access benchmark, Applied Energy, Volume 293, 2021, 116983, ISSN 0306-2619, https://doi.org/10.1016/j.apenergy.2021.116983.

[2] R. Zhang, G. Li and Z. Ma, ”A Deep Learning Based Hybrid Framework for Day-Ahead Electricity Price Forecasting,” in IEEE Access, vol. 8, pp. 143423-143436, 2020, doi: 10.1109/ACCESS.2020.3014241.

[3] https://transparency.entsoe.eu/