Investigation on the performance of the EPF toolbox models with NGBoost Model for Electricity Price Forecasting

Maria Antony

***Abstract*— As electricity is an important part of everyone's life, it is very important to understand electricity price forecasting which has an impact on living expenses. In this perspective, Electricity price forecasting is an important task to be performed which may bring some impact from tiny to gigantic things. Finding the right model for this forecast is always challenging and exciting.**

In this work, the two models, LEAR and DNN of an electricity price forecasting toolbox(EPF) for point forecast are compared with a probabilistic model called Natural Gradient Boost(NG Boost). The two models of the EPF toolbox are observed and then analyzed with the NGboost model using evaluation matrices such as Mean squared error(MSE) Mean absolute error(MAE and Root mean squared error(RMSE).

Keywords—EPF toolboxNGBoost, Evaluation Metrics

I. INTRODUCTION

Electricity price forecasting is an important field that has to be discussed concerning sustainability and for a smooth going of one's life. Choosing the best algorithm for a forecast is always challenging. There exist many pros and cons for the existing works related to the research in this area. Electricity price forecasting (EPF) advances are continually introducing new techniques intending to close the gap between projections and real prices. However, development in this sector is neither consistent or straightforward to track[1]. Existing papers are not clear about the efficiency of methods. There exist checking of these issues by comparing the state-of-the-art statistical and deep learning methods across multiple years and markets and by putting forward a set of best practices.

This paper deals with an investigation of the performance of the models in the EPFtoolbox library and a probabilistic model, NGBOOST. EPFtoolbox is the first-ever made library for research in electricity price forecasting. There exist five different types of benchmark datasets to carry out the research and it also provides two different models LEAR and DNN models. This paper deals with training and testing on three different datasets among the five benchmark datasets. The datasets that have been used are PJM, NP, and FR datasets. Upon these, all the two models provided by the EPF toolbox have been used and along with that we have used NGBOOST. Upon the completion of testing the dataset for 31 days, we have used different evaluation matrices such as MAE and sMAPE. Using these matrices we have compared the accuracy and performance of all the three different models and recorded them. Also, we have carried out find the reliability scores for NGBOOST and plotted the graph to show its accuracy. RMSE of the NGBOOST is also found. Thus this paper describes all these processes in detail.

II. EXISTING WORK

This particular part of the paper gives an idea about the existing works on electricity price forecasting using different models. Since this paper is particularly focused on the investigation of the Electricity Price Forecasting and NG boost, there don't exist any papers discussing both. In this regard, It is very much accurate to have a look at the literature review papers discussing on EPF toolbox and some on the NGboost.

The work in the paper [1] suggests the best methods to be used for electricity price forecasting. Mainly it describes the state-of-the-art as EPF. It has been done by analyzing various factors affecting the accuracy between the actual data and the predicted data. As the comparisons on EPF are conducted on various types of unique data which might not be accessible, this paper provides around five benchmark datasets to make the research much more understandable and comparable. Using this we can further investigate as the framework remains the same and thus meaningful comparisons can be obtained. For building up smooth research, an open python library has been built which is called as EPFtoolbox, which provides five datasets with two-point forecast models known as LEAR and DNN. This paper claims that DNN is more likely to perform better than the LEAR model after conducting the experiments on the datasets provided. By conducting the performance analysis results are obtained in terms of statistical testing and accuracy matrix which supports the above statement. This paper also provides a set of guidelines to be followed for further research. So in short, this paper provides a detailed view of the state of the art in epf, then the information on the publicly available data sets and model for future research, and a glance at the open-source toolbox. This paper also reviews various statistical, ML, and hybrid methods as well. [1] also provides a statement on state-of-art stating that, according to the various studies that have been conducted, LEAR is an accurate model and further be optimized though by employing variance stabilizing transformations, aggregating forecasts from multiple calibration windows, and/or using long-term seasonal decomposition to alter the prices [1].

In the paper [2] the author addresses the existing flaws in present investigations and the field that may be studied in the future, in order to give a certain reference for future price forecasting research, after summarizing the application features of various price forecasting models. Because of the non-storable and real-time nature of electricity commodities, price formation is more difficult than for common commodities and is influenced by a variety of market factors such as load, weather, season, and market rivalry. This research distinguishes between several types of price

XXX-X-XXXX-XXXX-X/XX/$XX.00 ©20XX IEEE

forecasting models and examines their application properties. Finally, flaws in existing research are examined, as well as fields that can be researched in the future, to offer a solid foundation for future study. The paper states that electricity price forecasting models may be classified into five groups based on modeling principles: Market Equilibrium, Structural Model, Statistical Model, Intelligent Model, and Combination Model. It may be classified into short-term, mid-term, and long-term price forecasting based on the forecasting time. The former receives the greatest attention, and statistical models, intelligence models, and combination models are the most often utilised short-term models. However, compared to the short-term, there is less study on price predicting in the mid- and long-term. The correlation coefficient (R), mean absolute error (MAE), mean absolute percentage error (MAPE), and root-mean-square error(RMSE) are the four types of indicators used to evaluate predicting effectiveness.

Another paper[3] uses the electricity price in the Australian electricity market as an example, solving the actual operation with the genetic algorithm and the **back propagation(**BP**)** neural network model, identifying the cause of the final error, and demonstrating the superiority of the BP neural network based on genetic algorithm over the traditional BP neural network. To estimate power prices, this research employs a BP neural network method based on GA. A prediction model incorporating the effect of price fluctuation trend and load fluctuation on electricity price is created using the Neural Network Toolbox and the Sheffield Genetic Algorithm Toolbox in MATLAB. The paper has concluded that **t**he BP neural network technique based on GA is useful for predicting power prices. The daily average percentage error can be kept below 15% to meet the forecast accuracy standards[3]. This technique, when compared to the classic BP neural network algorithm, may overcome the limitations of local optimization and greatly enhance forecast accuracy, particularly during periods of high price swings. There are still flaws in this article. The BP neural network approach enhanced by GA is more accurate than BP neural network prediction when the price volatility of electricity is considerable, but the accuracy is still not perfect. One of the most crucial issues is the scarcity of data. Furthermore, this article solely evaluates the influence of past price and load changes on the outcomes of electricity price forecasting, neglecting market supply and demand as well as other subjective elements such as power producers, and therefore ignoring numerous unknown aspects in real life

III. EPF TOOLBOX

The epftoolbox is the world's first open-access library for energy price forecasting research. Its major purpose is to create a collection of tools available for energy price forecasting research that ensures repeatability and establishes research standards[1][4]. The library consists of three main components which are nothing but data management for data processing and dataset extraction. The second one is the model subpackage[1][5]. It includes two forecasting models for projecting power prices. There is a module for the LEAR model and another one for the DNN model in this package[1][5]. And the last package is called an evaluation subpackage. The evaluation subpackage comprises a module for evaluating model performance in terms of accuracy measures, as well as a module for statistically comparing model forecasts.

A. Data Management

1. Dataset Extraction

This module provides an easy-to-use interface for downloading data from several day-ahead power markets. The module is based on the read data function, which may be used to get five market data[1][5]. Along with this data, it also facilitates reading the data from other sources as well. Another feature is that it automatically splits the data into train data and test data by specifying the path and the other parameters.

The library gives simple access to a set of tools and standards for evaluating and comparing novel strategies for predicting electricity prices[5].

a. Open-access benchmark dataset: The library provides free access to five datasets. For an EPF benchmark dataset to be fair, it must meet three criteria:

- consists of numerous power markets to test the capability of new models in a variety of scenarios.
- be lengthy enough for out-of-sample datasets spanning 1–2 years to be used to assess algorithms, and
- be recent enough to account for the implications of integrating renewable energy sources on wholesale energy costs

They have proposed five datasets reflecting five alternative day-ahead power markets, each of which has six years of data, based on these requirements. Each market's pricing have its own set of dynamics[1][5].

The first dataset covers the Nord Pool (NP), which is the Nordic nations' European electricity market, from January 1, 2013, to December 24, 2018. The dataset includes hourly day-ahead pricing observations, day-ahead load forecasts, and day-ahead wind generation forecasts[1][5]. The second dataset comes from the United States Pennsylvania–New Jersey–Maryland (PJM) market. It covers the same period as Nord Pool, that is, from January 1, 2013, to December 24, 2018. The three-time series are Commonwealth Edison (COMED) zonal pricing (a zone in the state of Illinois) and two day-ahead load projection series, one for the system load and the other for the COMED zonal load. The third dataset reflects the EPEX-BE market, which is run by EPEX SPOT and is Belgium's day-ahead electricity market. The data is from January 9, 2011, to December 31, 2016. The day-ahead load prediction and day-ahead generation forecast in France are represented by the two exogenous data series. While this choice may appear odd, research has shown that these two are the strongest predictors of Belgian pricing. The fourth dataset reflects the EPEX-FR market, which is likewise run by EPEX SPOT and is a day-ahead power market in France. The dataset covers the same period as the EPEX-BE dataset, namely from January 9, 2011, to December 31, 2016. The information also includes a day-ahead load prediction and a day-ahead generation projection, in addition to energy pricing. The last dataset is for the EPEX-DE market, which is likewise run by EPEX

SPOT and is the German power market. The data is from January 9, 2012, to December 31, 2017. The dataset also includes day-ahead zonal load projections in the TSO Amprion zone, as well as aggregated day-ahead wind and solar generation forecasts in the zones of the three largest9 TSOs (Amprion, TenneT, and 50Hertz)[1][5].

2. Dataset Wrangling

This module is for converting data into a format that can be read and processed by the EPFtoolbox library's prediction models. The module's capabilities are currently confined to scaling activities. This module contains two components, the Data scaler class, and the scaling function. The data scaler is for scaling operations and the Scaling function is for scaling a list of datasets[1][5].

B. Forecasting Models

Two state-of-art forecasting models are included in the library, which may be used automatically in any day-ahead market without the requirement for specialist expertise. The library currently includes two major models, one of which is based on a deep neural network. A second model, based on an autoregressive model with LASSO regulation, was developed (LEAR)[1][5].

1. LEAR Model

   The Lasso Calculated AutoRegressive (LEAR) model is a parameter-rich ARX structure estimated using L1-regularization [6]. It was first released under the name LassoX in [7]. The LEAR is based on the so-called full ARX or fARX model, which is a parameter-rich autoregressive specification with exogenous variables that are inspired by the general autoregressive model specified by Equation (1) in [8], but with several key changes.

$$\operatorname{asinh}(x) = \log\left(x + \sqrt{x^2 + 1}\right) \quad (1)$$

   where x is the price adjusted by a factor for asymptotically normal consistency to the standard deviation after removing the in-sample median and dividing by the median absolute deviation.

2. DNN model

   The DNN is a four-layer deep feedforward neural network that uses a multivariate framework (one model with 24 outputs), is estimated with Adam [9], and its hyperparameters and input characteristics are optimized with the tree Parzen estimator[1],[5],[10], a Bayesian optimization technique. Figure 1 depicts the structure of the system.

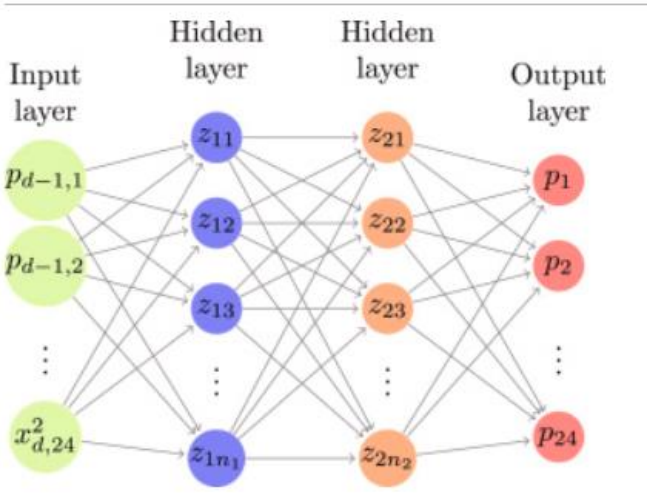

Fig. 1. Visualization of a sample DNN model.

C. Model Evaluation

This subpackage contains tools for assessing forecasts and forecasting models using accuracy measures and statistical tests. The accuracy metrics module provides the initial collection of tools, which analyze the mistakes of predictions based on a single number.

1. Accuracy Metrics

Some of the key accuracy matrices are MAE, MAPE, sMAPE, RMSE, MASE, Rmae, Naïve Forecast. The most commonly used accuracy matrices are mean absolute error(MAE) and the root mean square error(RMSE). Since the RMSE has certain disadvantages, the focus goes on MAE and sMAPE[1][5].

$$\mathrm{MAE} = \frac{1}{N}\sum_{k=1}^{N}|p_k - \hat{p}_k|,$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{k=1}^{N}(p_k - \hat{p}_k)^2},$$

The downside of RMSE is that it does not adequately capture the underlying problem.

The mean absolute percentage error is another common measure (MAPE).

$$\mathrm{MAPE} = \frac{1}{N}\sum_{k=1}^{N}\frac{|p_k - \hat{p}_k|}{|p_k|}.$$

While it gives a relative error metric for comparing datasets, its values grow quite huge when prices approach 0 (independent of the real absolute errors), and it is therefore not particularly instructive[1][5]. The symmetric mean absolute percentage error (sMAPE) is another popular metric:

$$\mathrm{sMAPE} = \frac{1}{N}\sum_{k=1}^{N}2\frac{|p_k - \hat{p}_k|}{|p_k| + |\hat{p}_k|},$$

Although the sMAPE addresses some of these concerns, it has an undefined mean and infinite variance statistical distribution.

2. Statistical Tests
While it is critical to use appropriate criteria to assess forecast accuracy, it is also crucial to determine if any difference in accuracy is statistically significant. This is critical in determining if the variation in accuracy is real and not just due to random variances across forecasts.

   a. Diebold-Mariano test
   The Diebold–Mariano (DM) test [131] is arguably the most widely used method for determining the importance of predicting accuracy discrepancies. It's an asymptotic z-test of the hypothesis[1][5].
   b. Giacomini-white test
   For conditional predicting ability, the DM test has been substituted with the Giacomini–White (GW) test [137] in some of the more recent EPF research [11], [12], [13]. The latter is preferable since it is a generalization of the DM test for unconditional predictive ability[1][5].

II. PROBABILISTIC MODELS

Estimating the uncertainty in a machine learning model's predictions is critical for real-world production deployments. We want our models to not only produce accurate forecasts but also to provide a correct assessment of uncertainty for each prediction[14]. Predictive uncertainty estimations are crucial for identifying manual fallback options or for human inspection and intervention when model predictions are part of an automated decision-making workflow or manufacturing line[14]. Probabilistic prediction (or probabilistic forecasting) is a logical technique to measure such uncertainties since the model generates a complete probability distribution over the whole result space[14].

A. NGBoost

Typical regression models provide a point estimate based on covariates, but probabilistic regression methods provide a complete probability distribution across the resulting space based on the covariates. The basic learner, parametric probability distribution, ad scoring rule make up this algorithm. Any base learner, any family of continuous parameter distributions, and any scoring system may be employed with NGBoost[15]. NGBoost matches or outperforms existing probabilistic prediction algorithms while providing extra benefits in terms of flexibility, scalability, and usability[15]. Gradient Boosting approaches have consistently outperformed structured or tabular input data in terms of predicting accuracy[15]. With Gradient Boosting and probabilistic predictions, NGBoost offers predictive uncertainty estimation (including real-valued outputs). NGBoost addresses technological hurdles that make general probabilistic prediction difficult with gradient boosting by utilizing Natural Gradients[15].

1. Base Learner

With the Base parameter, NGBoost may be used with any sklearn regressor as the base learner. A depth-3 regression tree is a default[16].

2. Distributions
NGBoost may be used with several distributions, divided into regression (support on an infinite set) and classification (support on a finite set) (support on a finite set)[16].

3. Scores
Although each score may not be implemented for each distribution, NGBoost offers the log score (LogScore, also known as negative log-likelihood) and CRPS (CRPScore). (“Usage”) The Score parameter in the function Object() { [native code] } specifies the score[16].

4. Other Arguments
The learning rate, number of estimators, minibatch fraction, and column subsampling are also easily adjusted[16].

```
ngb = NGBRegressor(n_estimators=100, learning_rate=0.01,
        minibatch_frac=0.5, col_sample=0.5)
ngb.fit(X_reg_train, Y_reg_train)
```

The NgBoost method may be built by combining gradient boosting with the natural gradient[15].

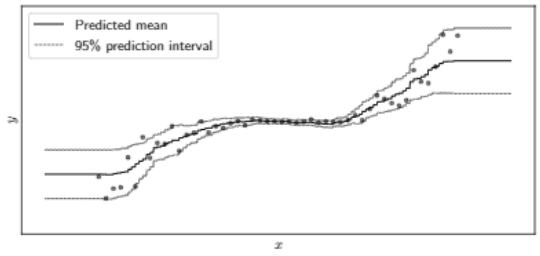

Fig 2. Prediction intervals fit with NGBoost for a toy 1-dimensional probabilistic regression issue. Datapoints are represented by the dots. After fitting the model, the thick black line represents the expected mean. The top and lower quantiles, which encompass 95 percent of the forecast distribution, are represented by thin grey lines[16].

The natural gradient and a multiparameter boosting technique are used in NGBoost to efficiently predict how parameters of the anticipated outcome distribution fluctuate with observable data[15]. NGBoost outperforms prior algorithms for probabilistic regression while retaining some key advantages: NGBoost is adaptable, scalable, and simple to use[15].

III. METHODS

This paper discusses the comparison of a point prediction using EPF toolbox models and of a probabilistic prediction using natural Gradient Boost(NGBoost). The methods have been carried out separately on EPFtoolbox models and NGBoost model and are as follows:

1. EPFtoolbox

As the epf toolbox is the first library for electricity price forecasting, we have gone through the entire documentation to investigate more on this library and to compare this with other models. Epf toolbox provides two different models called LEAR and DNN. And also provides five open benchmark datasets.

a. Data Preparation

As there are five open data sets of different power markets, we have fetched three different datasets, to conduct experiments. The datasets are Nord Pool dataset for a period of 01.01.2013 to 24.12.2018, then the PJM dataset for the same period as Nord Pool and EPEX-France for a period of 09.01.2011 to 31.12.2016. Each dataset has 5 years of data and for each day, there are 24 predictions that represent each hour. Dataset was already cleaned and all set to train.

b. Splitting the dataset into train and test data

The splitting of data is automatically performed in such a way that, all the data except the last 365 days data has been taken for training and the excluded 365 days data have been used for testing the data. Similarly, splitting has been carried out for all three datasets. Upon the completion of splitting, the path to save the forecast has been set and defined empty forecast array and the real values are to be predicted in a more friendly format[1].

c. Models usage

Once the data is all set to get trained, the next step we have carried out is using the model. As there are two models in the library, we have carried out predictions using both LEAR and DNN models. And predicted the price of 24 hours for 365 days. Along with prediction, the performance using the MAE and SMape for all 365 days is calculated and recorded.

This set of operations are carried out for all three datasets and got the results.

2. NGBoost

LEAR and DNN models have to be compared with NGBoost model to compare their performances.

a. Data Preparation

The experiments have been carried out on the three different datasets provided in the EPFtoolbox. The datasets that we have chosen are PJM dataset, NordPool dataset, and France dataset. We have taken the whole dataset for training and thirty-one days of data for testing.

b. Splitting the dataset into train and test data

We have obtained the train and test data from the EPF toolbox. The datasets that have been provided as benchmark datasets have been converted into a CSV file and saved for all three datasets. These CSV files are then used for training and testing. There exist Forty-Three Thousand Six Hundred Eighty rows and four columns for each training set of all three datasets, which contains twenty-four-hour data per day. Similarly, we have seven hundred forty-four rows and four columns for the test set which depicts thirty-one days of data and represents twenty-four hours of data per day. Once the splitting of the data into train and test data set is done, we have defined an empty array for the forecast to be saved. Along with this forecast array we have defined the real values to be predicted in a more friendly format. Since the ngboost has certain limitations of taking the input as date, we have just converted the data into ngboost acceptable format using the lambda function. This has been carried out for all the three datasets and after that, we have set the index as the date for both the test and train data. The setting index as the date has been carried out for real values data and forecast data as well. Soon after this, x_train, x_test, y_train, and y_test have been defined. X_train takes all the columns as input except price and y_train takes this price. Similarly, x_test and y_test have been set.

c. Models usage

Once the data has been all set to get trained, we have set a loop over each date and trained. Once the training is done, it returns 'ngboost trained' statement and it also returns the forecasted values. Those predicted values have been saved in the forecast array. We are also trying to find the mean and standard deviation for further results. All these results have been recorded and saved.

d. Hyperparameter Tuning

It has been found that The MAE and sMAPE values are a bit higher for NGBoost when compared to DNN and LEAR models. So a hyperparameter tuning has been carried out by setting different parameters. And each change for around five experiments has been done. The base_max_depth, n_estimators, and learning rate from the substring were tested for different values and recorded.

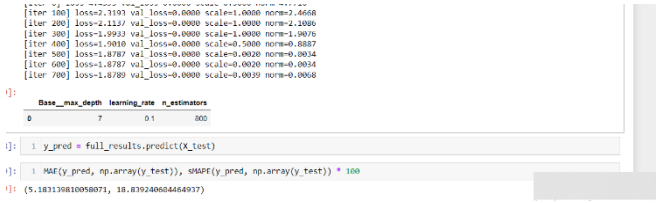

Fig 3. Hyperparameter tuning performed for NGBOOST

Base_max_depth	Learning_rate	n-estimators	MAE	sMAPE
7	0.1	800	5.1831	18.839
9	0.09	800	5.1454	18.618
11	0.1	620	5.1728	18.73

Table 1. Results obtained after hyperparameter tuning for NGBOOST

e. Calculation of reliability scores.

The reliability diagram is a diagnostic graph that is commonly used to describe and assess probabilistic forecasts. Its advantages include the simplicity with which it can be created and the clarity with which it can be defined[17]. To find the reliability diagram, along with the forecast, we have also found the mean and standard deviation and combined the forecasted array and the real values arrays to obtain a combined data frame. And from this combined data frame, we have calculated the reliability scores and plotted and also found the sharpness score.

3. Comparison on the performance matrices
   After all theses processes, We have calculated the MAE and Smape of all LEAR, DNN and NGBOOST models on all the three different datasets and recorded for comparison. RMSE of the NGBOOST model on all the three data sets has been calculated and plotted a few.

IV. Results

In this section, we sum up all the results and findings obtained throughout the way of our project. The main objective of this paper is to investigate different models for electricity price forecasting and compare them to understand the state-of-the-art. We have carried out the processes on different datasets provided as benchmark datasets from EPFtoolbox. The forecasted values with and without ensemble have been provided. Thus we have calculated the performance using performance matrices such as mean absolute error(MAE) and symmetric mean absolute percentage error of all the datasets using the three models. Since it is a comparison and emphasizes more on NGBOOST as well, we have then calculated the RMSE for all the three datasets using NGBOOSt. All these results are depicted in table 1.

	PJM	NP	FR
LEAR_MAE	2.2324	1.7682	3.6409
LEAR_sMAPE	7.5971	5.3357	10.1360
DNN_MAE	2.1765	1.5872	3.4928
DNN_sMAPE	7.5091	4.7148	9.1979
NGBOOST MAE	10.5821	2.9515	14.8901
NGBOOST sMAPE	28.3385	8.4511	28.6185
NGBOOST RMSE	0.2152	0.2085	0.3711

Table 2:
MAE, sMAPE, and RMSE were calculated for the models. PJM stands for Pennsylvania-New Jersey-Maryland[18], NP represents NordPool and FR represents French dataset. All the sMAPE are represented in terms of percentage(%).

The above table shows the different matrices calculated values for comparison. The findings in the table say that, when it goes to PJM, we can observe that, the DNN model has the least MAE among all the three models whereas, NGBOOST has the highest MAE. Similarly, sMAPE is also the same lowest for DNN and highest for NGBOOST as the lower values are always better. When we look into the NordPool dataset, we will come to know the same thing as PJM. DNN has the lowest MAE and MAPE when compared to LEAR and NGBOOST. It repeats the same for the French dataset as well. Another finding that could be useful is nothing but since all the datasets are models tested for thirty-one days, we can see a big deviation in NGBOOST which reduce the scope of using NGBOOST for the electricity price forecasting.

The reliability diagram is a common diagnostic aid for quickly evaluating the reliability of probabilistic forecast systems[17]. Reliability diagrams are used to determine if the forecast value Xi is accurate. A probability forecast is considered reliable1 if the event occurs with a relative frequency that is consistent with the forecast value[17]. A forecast is said to be reliable if A reliable forecast should have a reliability diagram close to the diagonal.

The average accuracy for the cases in that bin is indicated by the blue lines in the plot:

The model is overconfident for the samples in that bin if the blue line is near the bottom of a red bar. If a blue line appears on top of a red bar, the model's predictions aren't accurate enough. Overconfidence (conf > acc) leads to a higher number of false positives[19]. Underconfidence (acc > conf) leads to a higher number of false negatives[19].

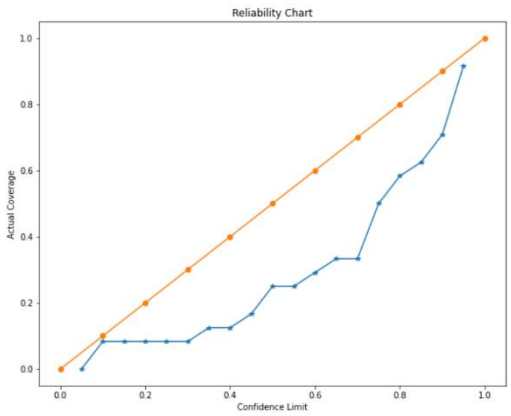

Fig 4. Reliability diagram of PJM_forecast using NGBOOST
This picture depicts overconfident

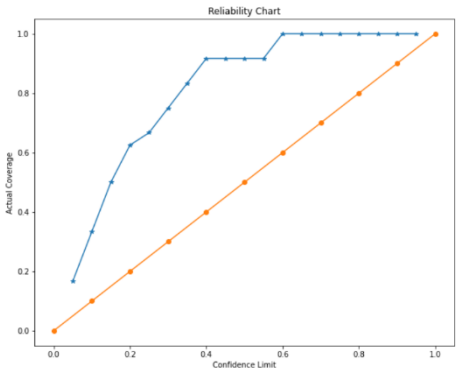

Fig 5. Reliability diagram of NP_forecast using NGBOOST
This picture depicts under-confident

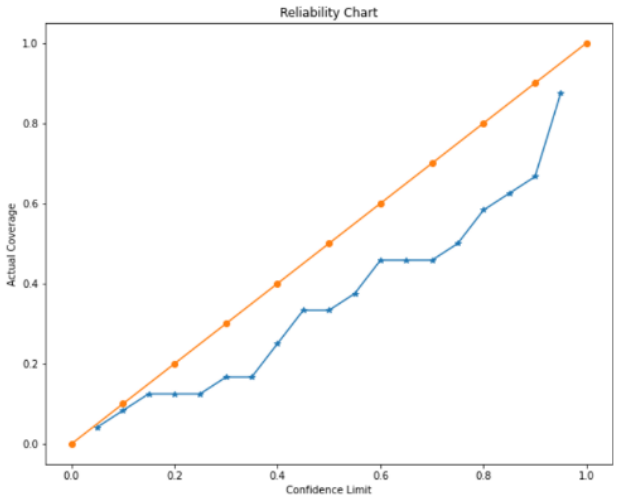

Fig 6. Reliability diagram of FR_forecast using NGBOOST
The picture represents overconfident.

The average accuracy for the cases in that bin is indicated by the blue lines in the plot:

V. Discussion

Based on the findings and researchers, it founds that DNN is the most appropriate model that can be used for electricity price forecasting since it has a low value of MAE and SMAPE when compared to the other two models. for electricity price forecasting. Further investigation on calibration has a good scope for research. Experiments on EPFtoolbox models were conducted with an easy-to-use interface for testing the model in the provided test dataset[1]. The other option available for further investigation on LEAR model is using flexible recalibration[1]. This also provides a flexible interface. For the DNN model, there is a scope for hyperparameter optimization even though DNN is state-of-the-art. Along with this, there is a chance for easy recalibration and flexible calibration as well. When it comes to NGBOOST model, even though we have tried hyperparameter tuning, there is a way for improving reliability using calibration. It may help to make the forecast much more reliable.

III. Conclusion

An investigation of the performance of different models has been conducted. We have carried out training and testing on the different open benchmark datasets available in EPFtoolbox for electricity price forecasting. Among the five datasets, three datasets of PJM, NP, and FR have been chosen and trained using LEAR, DNN, and NGBOOST models where LEAR and DNN are the models from the EPFtoolbox library. Each dataset has been checked for a month of data and found MAE and sMAPE for each model. Table 1 depicts the complete results. And it is very clear from the table that, the forecasting using the DNN model results in better and more trustworthy output. So, DNN can be accepted as the best model and can also be termed as state-of-the-art for electricity price forecasting. NGBOOST doesn't perform well even though the hyper-parameter tuning is done. Thereafter RMSE for ngboost has been found and the Reliability diagram for each dataset using NGBOOST model has been plotted. A forecast is said to be reliable if the real value and the forecast value match.
To conclude, out of all the three models, that is the two from EPFtoolbox and the NGBOOST, the DNN models perform the best with the least MAE and sMAPE

References

[1]. Lago, J., Marcjasz, G., de Schutter, B., & Weron, R. (2021). Forecasting day-ahead electricity
prices: A review of state-of-the-art algorithms, best practices and an open-access benchmark. Applied Energy, 293, 116983. https://doi.org/10.1016/J.APENERGY.2021.116983
[2]. Zelikman, E., Zhou, S., Irvin, J., Raterink, C., Sheng, H., Avati, A., Kelly, J., Rajagopal, R., Ng, A.
Y., & Gagne, D. (2020). Short-Term Solar Irradiance Forecasting Using Calibrated Probabilistic Models. https://arxiv.org/abs/2010.04715
[3]. Mitrentsis, G., & Lens, H. (2022). An interpretable probabilistic model for short-term
solar power forecasting using natural gradient boosting. Applied Energy, 309,118473. https://doi.org/10.1016/J.APENERGY.2021.118473
[4]. https://github.com/jeslago/epftoolbox
[5]. https://epftoolbox.readthedocs.io/en/latest/
[6]. Tibshirani R. Regression shrinkage and selection via the lasso J R Stat Soc Ser B Stat Methodol (1996), pp. 267-288, 10.1111/j.2517-6161.1996.tb02080.x
[7]. Uniejewski B., Nowotarski J., Weron R.Automated variable selection and shrinkage for day- ahead electricity price forecasting Energies, 9 (8) (2016), p. 621, 10.3390/en9080621
[8]. Ziel F.Forecasting electricity spot prices using lasso: On capturing the autoregressive intraday structure IEEE Trans Power Syst, 31 (6) (2016), pp. 4977-4987, 10.1109/tpwrs.2016.2521545
[9]. Kingma D.P., Ba J.Adam: A method for stochastic optimization 3rd International Conference
on Learning Representations, ICLR (2015) http://arxiv.org/abs/1412.6980Google Scholar
[10]. Bergstra J., Bardenet R., Bengio Y., Kégl B. Algorithms for hyper-parameter optimization
Advances in Neural Information Processing Systems (2011), pp. 2546-2554 Google Scholar
[11]. Marcjasz G., Serafin T., Weron R. Selection of calibration windows for day-ahead electricity price forecasting Energies, 11 (9) (2018), p. 2364, 10.3390/en11092364
[12]. Serafin T., Uniejewski B., Weron R. Averaging predictive distributions across calibration windows for day-ahead electricity price forecasting Energies, 12 (13) (2019), p. 2561, 10.3390/en12132561
[13]. Marcjasz G, Lago J, Weron R, Schutter BD. 2020. Neural networks in day-ahead electricity price forecasting: single vs. multiple outputs. Google Scholar
[14]. Duan, T., Avati, A., Ding, D. Y., Thai, K. K., Basu, S., Ng, A., & Schuler, A. (2020). NGBoost: Natural Gradient Boosting for Probabilistic Prediction.
[15]. T. Duan, A. Anand, D. Y. Ding, K. K. Thai, S. Basu, A. Ng, and A. Schuler, "Ngboost: Natural
gradient boosting for probabilistic prediction," in International Conference on Machine Learning. PMLR, 2020, pp. 2690–2700
[16]. https://stanfordmlgroup.github.io/projects/ngboost/
[17]. Bröcker, J., & Smith, L. A. (2007). Increasing the Reliability of Reliability Diagrams, Weather
and Forecasting, 22(3), 651-661. Retrieved May 4, 2022, from https://journals.ametsoc.org/view/journals/wefo/22/3/waf993_1.xml
[18]. https://sandbox.zenodo.org/record/632147#.YnMW7trMJPY
[19]. https://github.com/hollance/reliability-diagrams