ISB

Predicting Daily Wind Power Forecast for Optimal Thermal Power Backup

Group – A4

Rahul Gupta (61710574)
Souvik Sen (61710577)
Srikanth Akkapeddi (61710122)
Avinash Dhanuka (61710696)
Mihir Kedia (61710267)
Animesh Biyani (61710155)

Table of Contents

Executive Summary

Business Problem

Wind power has recently started gaining an important role in the energy mix. Several utility companies have switched to wind power to meet most of their demand. However, there is one big problem associated with wind power - intermittency. As a result utility companies have to maintain backup power which is generated using coal. This makes it expensive and dirty. In order to maintain an optimal level of backup power, the utility companies must have accurate wind power forecasts available. This is the problem that we will try to address.

Description of Data

The dataset has been obtained from kaggle.com and contains hourly wind power forecasts (normalized) from July 1, 2009 to July 31, 2012 and contains more than 1,00,000 rows. In addition to wind power, we are also given forecast for wind speed and wind direction by the meteorological department. We expected seasonality in the data but when we plotted the series month-wise and hour-of-day wise, we found that there is no seasonality in the data (Appendix: Fig 1). Next, we plotted the data for different months on day-of-hour basis to check for daily seasonality, and even in this case we did not find any seasonality (Appendix: Fig 2). The last time series component that we checked for was trend. In this case, we found that the trend was not a typical trend like linear or exponential, but a combination of several sinusoids of different frequencies (Appendix: Fig 3).

Conclusion & Recommendations

We found the following model to be optimal: Multiple Linear Regression (with average wind speed and direction as predictors), followed by another MLR run on the residuals with Lag 24 and Lag 48 as predictors. We used RMSE to compare different models (reason provided later). Based on the results, we found that our model captures the trend of the data really well, and if we provide prediction intervals (say 95%), our clients can choose from multiple values and with experience they can get better results using the model.

Data Preparation & Forecasting Methods

Data Preparation

There were several missing values in the dataset in the period from Jan 1, 2010 to Jul 31, 2012. The first step was to remove these rows from the dataset. Owing to the limitation of XL Miner of handling up to 10,000 rows, we reduced our dataset further. The final dataset that we used consisted of hourly wind power forecast from July 1, 2009 to August 20, 2010. Because there is no seasonality in the data, the length of the training period does not make a difference as long as we have sufficient data.

The dataset also contained hourly forecasts for wind speed and wind direction. The meteorological department provides forecasts for the next 48 hours with a frequency of 12 hours i.e. for each hour, we have 4 forecasts. We simply took the average of all the forecasts and used that as the predictor.

The dataset was then partitioned into training set and validation set, with validation set having a length of 48 hours.

Methods Used

We followed a two-step approach:

Step 1: De-trend the series using MLR

Step 2: Extract information from the residuals using AR models and Simple Exponential Smoothing

First step was to generate a naïve forecast so that we have a benchmark to compare our models against. We used a seasonal naïve with a seasonality of 24 hours. With naïve, we got an RMSE of 0.34 (Appendix: Fig 5). We then used the following models:

- MLR to de-trend, followed by Simple Exponential Smoothing
- MLR to de-trend, followed by another MLR with Lag 24 and Lag 48 as predictors
- Neural Networks (1 hidden layer with 25 nodes)

Models Used & Results

De-Trending the series using MLR

To de-trend the time series, we can use one of the two following approaches:

- Centered Moving Average
- Regression

In our case, centered moving average is not a feasible alternative because the data has high frequency and high volatility (risk of over-smoothing and under-smoothing). Therefore we decided to use regression to de-trend the series. To perform regression we needed predictors which are correlated to the output (wind power) and have a similar trend. Luckily, in our case we found that wind speed and wind direction has trends similar to wind power and are highly correlated to wind power as well. (Appendix: Fig 6). We then ran the regression with wind power as the output variable and wind speed & direction as the predictors. This reduced the RMSE to 0.15 (Appendix: Fig 7)

Training Data Scoring - Summary Report

Total sum of squared errors	RMS Error	Average Error
387.0816	0.197377	-1.3805E-16

Validation Data Scoring - Summary Report

Total sum of squared errors	RMS Error	Average Error
1.121939	0.152885	0.029285053

Model 1: MLR to de-trend, followed by simple exponential smoothing

Once we de-trended the data using regression, we obtained residuals which were de-trended and de-seasonalized (as there is no seasonality in the data). We then ran simple exponential smoothing on the residuals with $\alpha = 0.15$ to extract more information from the residuals. The forecasts of residuals were added back to the MLR predicted values (Appendix: Fig 8). However,

we saw only a marginal improvement in the predictive accuracy. In fact, we see similar patterns in the predicted values for both the models.

Training Data Scoring - Summary Report

Total sum of squared errors	RMS Error	Average Error
387.0816	0.197377	-1.3805E-16

Validation Data Scoring - Summary Report

Total sum of squared errors	RMS Error	Average Error
1.121939	0.158056	0.029285053

Model 2: MLR to de-trend, followed by MLR with Lag 24 and Lag 48

We generated an ACF plot for the residuals (Appendix: Fig 10), and found that the initial lags have high correlation. Because we are forecasting for the next 24 hours, we cannot use the initial lags. If we look at the graph, we see that there are significant correlations at lag 24 and lag 48. Therefore, we decided to use these values as the predictors and ran MLR again.

Again, we saw only a marginal improvement in forecast accuracy compared to simple MLR (RMSE reduced to 0.14). Also, we see the same patterns in the predicted values for the validation period (Appendix: Fig 11)

Training Data Scoring - Summary Report

Total sum of squared errors	RMS Error	Average Error
387.0816	0.197377	-1.3805E-16

Validation Data Scoring - Summary Report

Total sum of squared errors	RMS Error	Average Error
1.121939	0.145447	0.029285053

Model 3: Neural Networks

In case of neural networks, we did not use MLR to de-trend. Instead we wanted the neural network to figure out the trend and make predictions. We used a neural network with a single hidden layer and 25 hidden nodes.

Unexpectedly, this model performed the worst among all other models. The forecasts were way-off in peak periods (RMSE increased to 0.18). However, the patterns in the prediction for validation period remain the same (Appendix: Fig 9)

Training Data Scoring - Summary Report

Total sum of squared errors	RMS Error	Average Error
468.8686586	0.21723	0.076667

Validation Data Scoring - Summary Report

Total sum of squared errors	RMS Error	Average Error
1.567001149	0.180682	0.077716

Ensemble Model

We did not use any ensemble models because the patterns in the prediction for validation period were same across all the individual models. In fact, for the neural network we see poorer forecasts. Hence, if we use an ensemble model the RMSE will increase leading to poorer forecasts. Though the forecasting accuracy won't change a lot, it does not make sense from a business perspective to run multiple models without a significant gain in forecasting accuracy.

Appendix

Figure 1

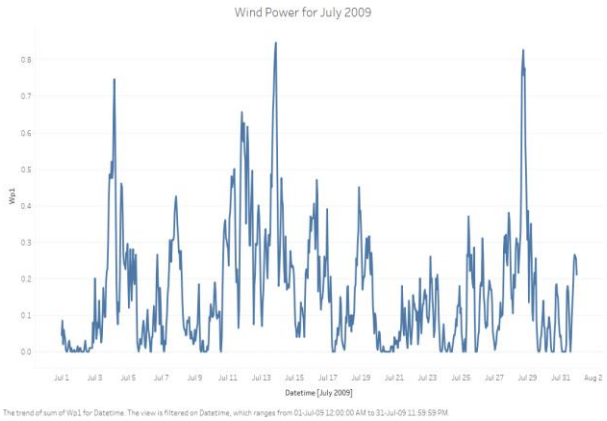

Figure 2

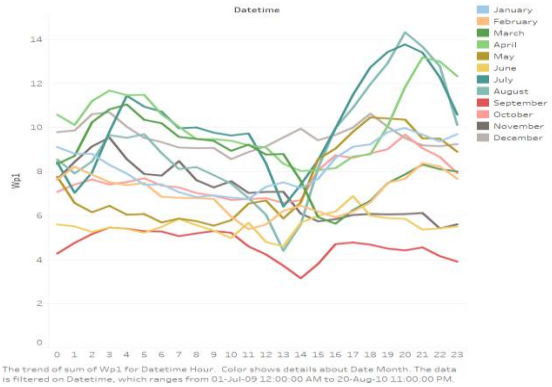

Figure 3

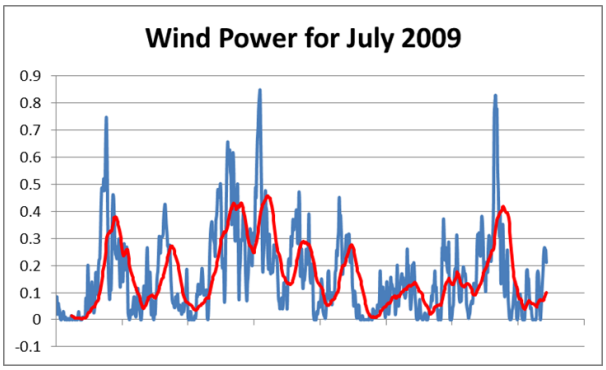

Figure 4

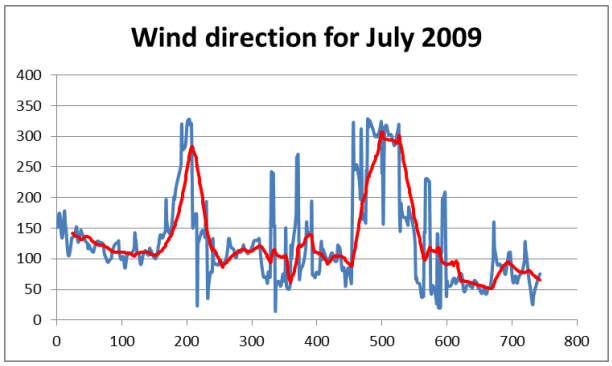

Figure 5

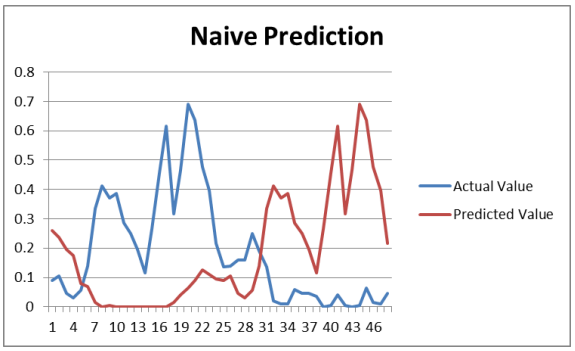

Figure 6

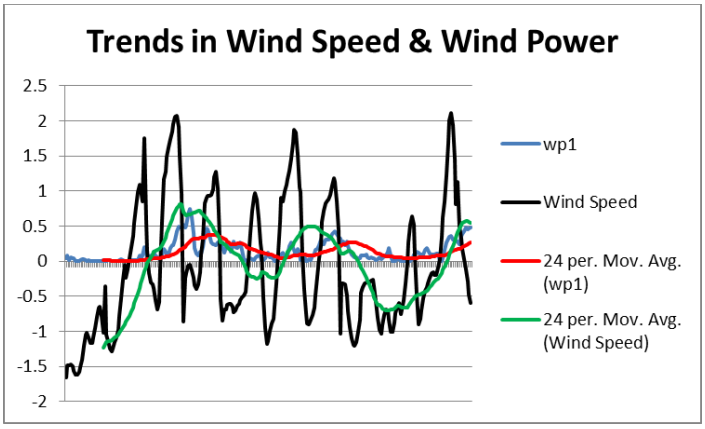

Figure 7

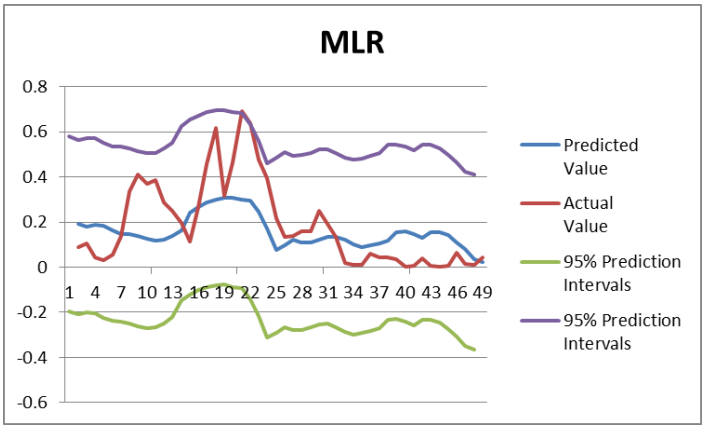

Figure 8

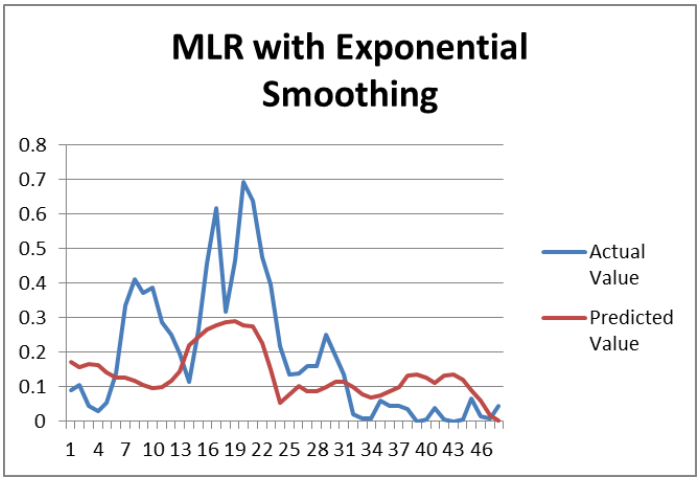

Figure 9

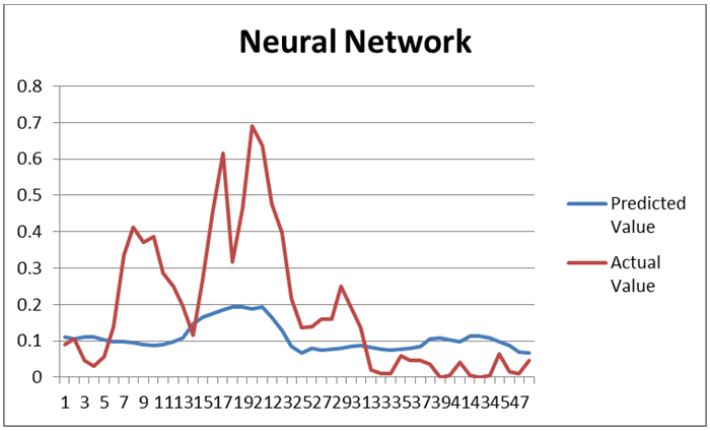

Figure 10

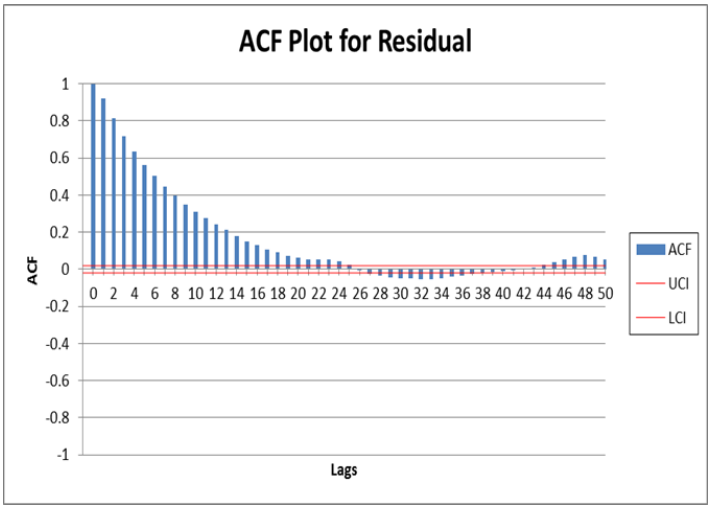

Figure 11