Assignment 2: Machine Learning for Wind Energy Forecasting

Mariusz Albin Szerepko
Universitetet i Oslo
mariussz@ifi.uio.no

Henriette Ekeberg
Universitetet i Oslo
henrieej@ifi.uio.no

StephenSimei Kimogol
Universitetet i Oslo
stephesk@ifi.uio.no

Dersim Waisi
Universitetet i Oslo
dersimwaisi@student.matnat.uio.no

Solution's folder and file structure

The solution folder consists of four main folders:

1) Task1R
    - **Task1.R** The solution as a R code
    - Solution.csv True wind power measurements
    - TrainData.csv Historical data from 01.01.2012 to 31.10.2013
    - WeatherForecastInput.csv Weather forecast for the whole month of 11.2013
    - ForecastTemplate*.csv Results files
2) Task1
    - MachineLearningTechnique.py ML methods
    - **Question1.py** The solution as a python code
    - **Task1.ipynb** The solution as a python notebook file
    - ForecastTemplate*.csv Results files
    - Solution.csv True wind power measurements
    - TrainData.csv Historical data from 01.01.2012 to 31.10.2013
    - WeatherForecastInput.csv Weather forecast for the whole month of 11.2013
    - Task1 / plots*.png Files containing plots
3) Task2
    - **Task2.R** The solution as a R code
    - ForecastTemplate2.csv Results file
    - Solution.csv True wind power measurements
    - TrainData.csv Historical data from 01.01.2012 to 31.10.2013
    - WeatherForecastInput.csv Weather forecast for the whole month of 11.2013
4) Task3
    - **Task3.R** The solution as a R code
    - ForecastTemplate3-LR.csv Results file
    - Solution.csv True wind power measurements
    - TrainData.csv Historical data from 01.01.2012 to 31.10.2013
    - WeatherForecastInput.csv Weather forecast for the whole month of 11.2013
5) Task3 / LSTM-RNN
    - **Task3 LSTM-RNN.ipynb** The solution as a jupyther notebook file
    - **Task3 LSTM-RNN.py** The solution as a python code
    - ForecastTemplate3-RNN.csv Results file
    - Solution.csv True wind power measurements
    - TrainData.csv Historical data from 01.01.2012 to 31.10.2013

Abstract

Over the past couple of decades, there has been more emphasis placed on renewable energy by society. Environmentally friendly and renewable energy is one of the most important measures politicians can introduce to combat climate change. In this paper we will analyse the relationship between wind power generation data, both alone and with two other weather parameters (wind speed and wind direction). Furthermore, based on the historical data, we can build a training model to gain a better insight into the relationship between weather and power. The training model is used to learn about the relationship between the weather and the power by historical data observed and collected on a wind farm in Australia. The historical data contains real wind power data and weather input and machine learning was used to analyse and observe this information. We used different machine learning technique to build our training models and the algorithms were implemented in programming languages Python and R.

Introduction

The main aim of this report has been forecasting wind energy using different machine learning techniques. The mandatory assignment was divided into three parts, the first part focused on the relationship between wind power production and wind speed. As mentioned above, we used machine-learning techniques to find the relationship and further used training data. The machines learning techniques included linear regression, k-nearest neighbour, supported vector regression and artificial neutral networks. The wind forecast was 10m above the ground level. Finally, we evaluated the prediction accuracy and compared the expected wind power and the real wind power measurements. The second part of this mandatory assignment is based on wind power production, depending on more parameters than wind speed, but also temperature and pressure. We focused on the relationship between wind power production and two weather parameters. We focused also on building several regression models, which gave an overview of the parameters for the whole month of November 2013. Finally, we will compare the expected wind power and true wind power measurements using metric RMSE. The last part of the mandatory assignment is based on the same situation as above, but in this case we will make a forecast for wind power production. This forecast is based on the fact that we only have wind power production data and no other data. In the training data file we used time-series data, and applied a linear regression and recurrent neural network to predict the wind power generation. As in the first and second part, we also want in the third part to compare the predicted wind power and the real wind power measurement by using RMSE.

Results and analysis

In this section we will present a brief analysis of the relationship between the wind power and the wind speed based on the training data from the file "TraninData.csv". We used machine-learning techniques to find the relationships and each technique uses different training models.

Task 1

The program can be run in terminal by: *python Question1.py*. The program will first solve linear regression, generate the curves and once you close the generated curve it goes to the next model. The predicted results are saved in files with their respective names. In this task machine learning techniques have been proposed to find the relationship between wind power generation and wind speed. Namely, linear regression, k-nearest neighbour (kNN), supported vector regression (SVR), and artificial neural networks (ANN) have been used. The results are as follows:

Model Name	**RMSE**
Linear Regression	0.216384
K-nearest neighbour (kNN)	0.216294
Supported Vector Regression (SVR)	0.213617
Artificial neural network (ANN)	0.215441

The results are very similar and SVR method gave the best RMSE compared to the other approaches. The small differences between the results are caused by different approaches used by each model for estimation. Also, as shown in illustration 1, in the range between 2 and 8, there is almost a linear relation between WS10 and POWER, and all lines from different techniques are almost the same. There is limited knowledge about the POWER variation in the data we have and using different models does not improve prediction.

Linear regression finds the best-fitting straight line through the points and minimizing the sum of squared errors of the prediction can generate this. The LR method gave a RMSE of 0. 216384. The kNN method finds nodes from training data set closest in distance to the test point, and predicts the power by averaging the values of the k-nearest points. A k of 1640 gave the best results in our trial and error approach. In supported vector regression, we trained the SVR model on the radial basis function kernel and by trying different values for epsilon we found that 0.083 gave the best RMSE. The artificial neural networks technique gave different results on every run with a varied performance. We got RMSE of 0. 215441 with two hidden layers of 4 and 6 nodes respectively.

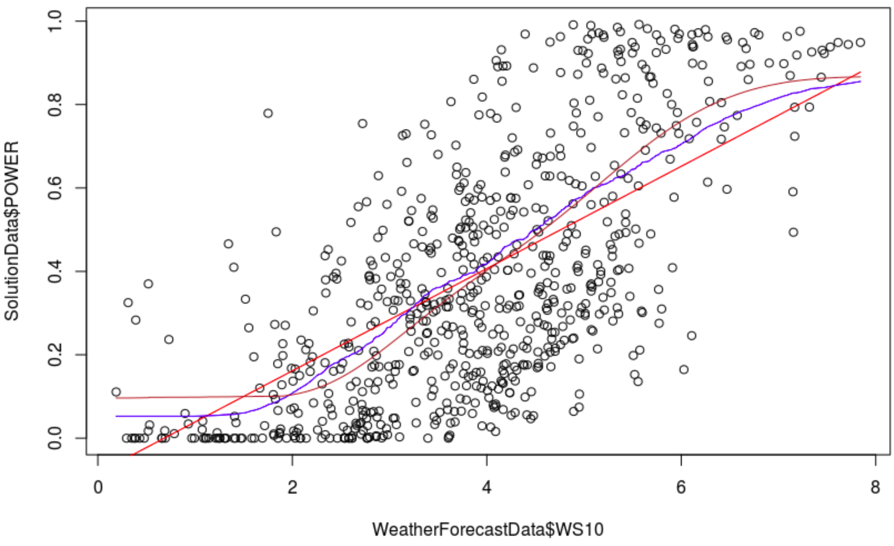

Illustration 1: The diagrams below illustrate the relationship between the room mean square error and True measured wind power for the different approaches.

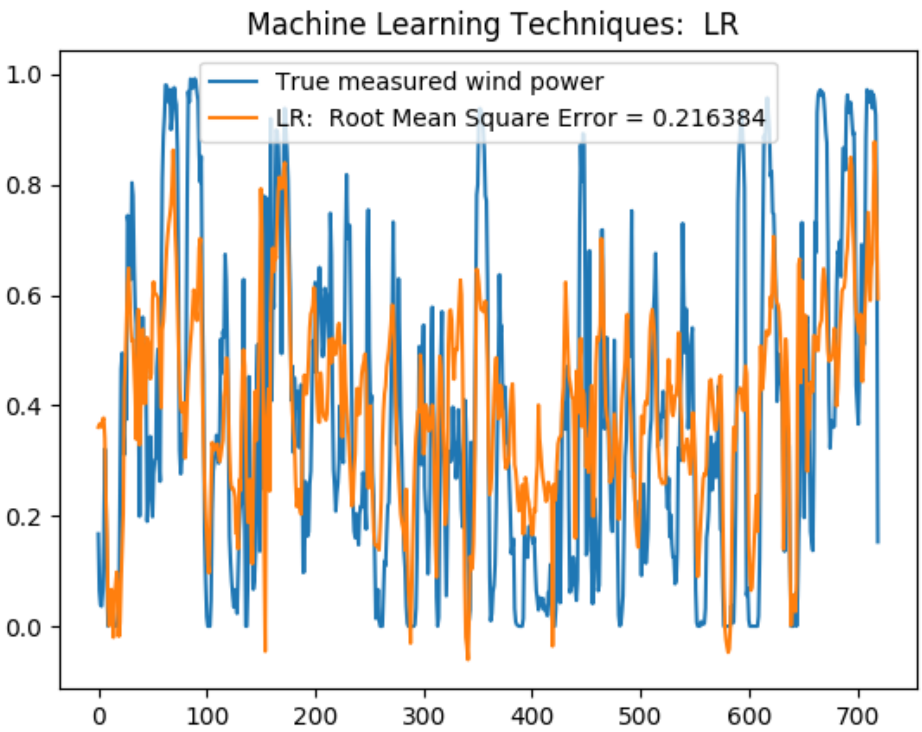

Illustration 2: Training model: Linear Regression RMSE: 0. 216384

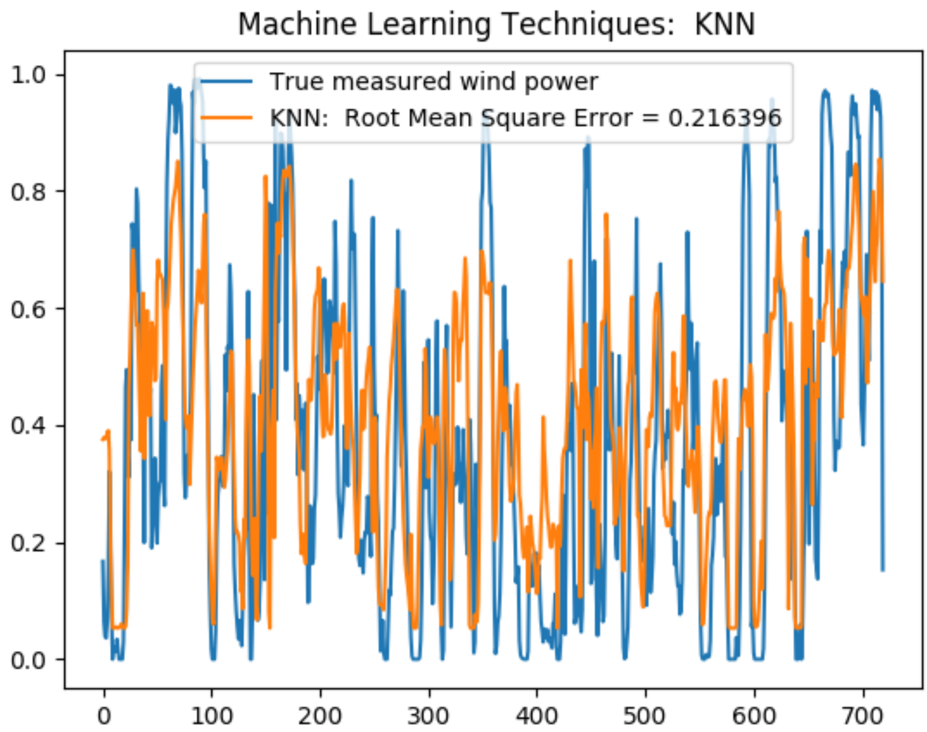

Illustration 3: Training model: k-Nearest Neighbor RMSE: 0. 216294

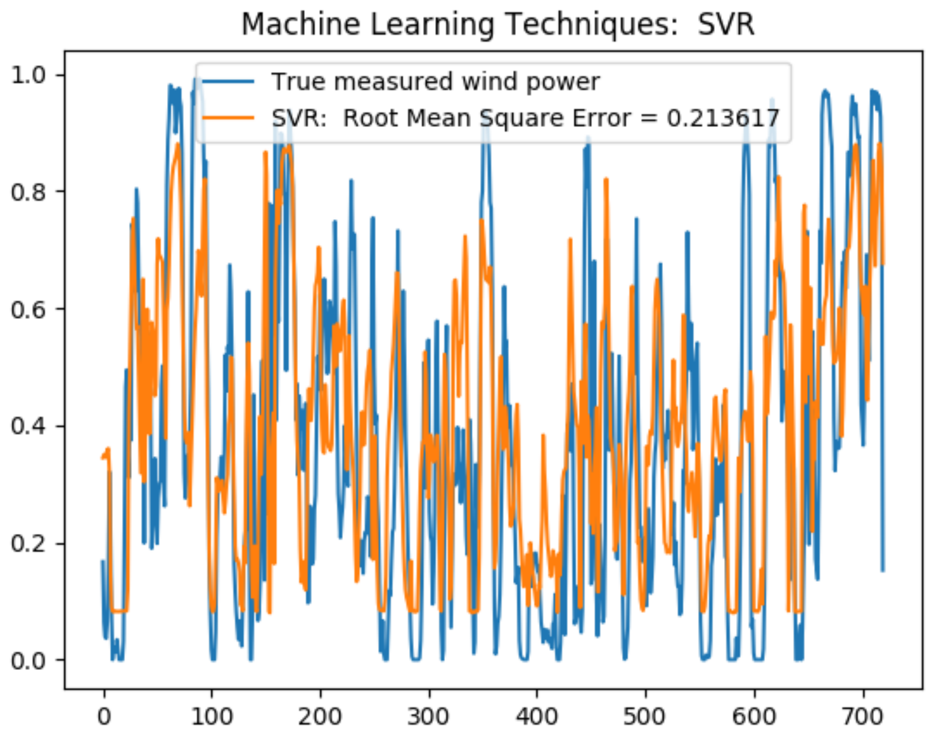

Illustration 4: Training model: Support Vector Regression RMSE: 0. 213617

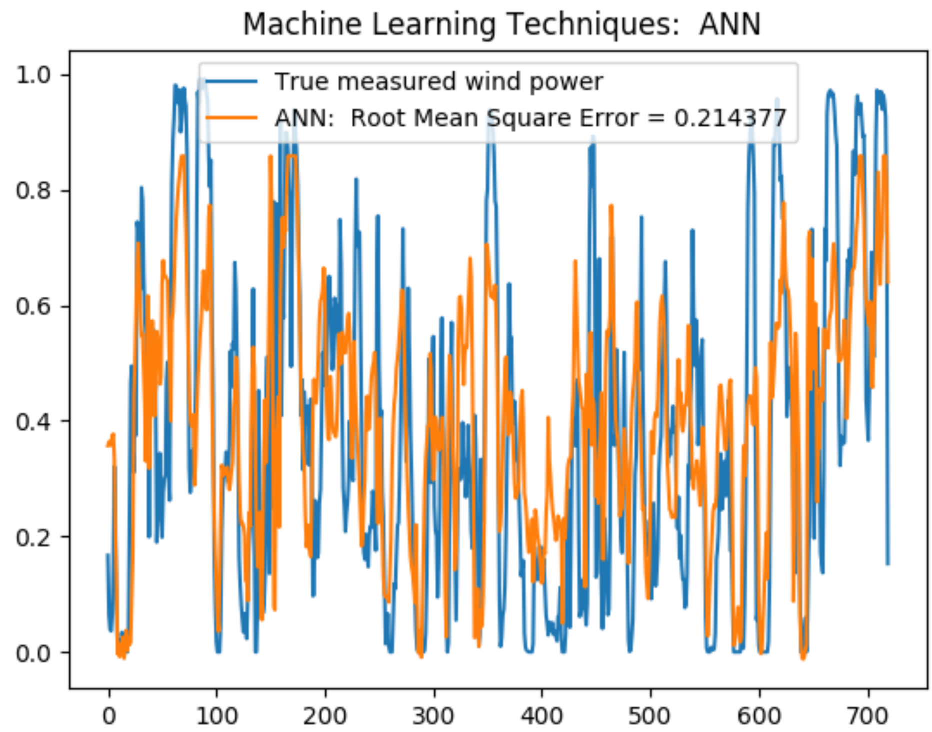

Illustration 5: Training model: Artificial Neural Network RMSE: 0. 215441

Task 2

In this section we present the relationship between the wind speed, wind direction, temperature and pressure. Furthermore, we have built a Multiple Linear Regression (MLR) model to analyze the relationship between the wind power generation and the two others parameters. We will also compare the predicted wind power and the true wind power measurement. Further, three Multiple Linear Regression (MLR) models have been proposed, considering different numbers and kinds of independent variables taken to the model:

1) Model including only the wind speed (**WS10**)- named "***model.MLR.w***"

2) Model including the wind speed (**WS10**), the meridional (**V10**), and the zonal (**U10**) components- named "***model.MLR.wvu***"

3) Model including the wind speed (**WS10**) and the wind direction (**direction**). Wind direction as a transformation of two factors- the zonal (**U10**), and the meridional (**V10**) components- named "***model.MLR.wd***". [http://mst.nerc.ac.uk/wind_vect_convs.html]

In the "**Model.MLR.wvu**" model, the wind direction factor has been calculated from orthogonal velocity components (**V10** and **U10**) and expressed as an azimuth direction (**direction)** [http://mst.nerc.ac.uk/wind_vect_convs.html]. The azimuth direction is an interval from -180 to +180 degrees.

Model name	Independent variables	RMSE
Model.MLR.w	WS10	0.2163841
Model.MLR.wvu	WS10, V10, U10	0.2085747
Model.MLR.wd	WS10, direction (V10 + U10)	0.2155597

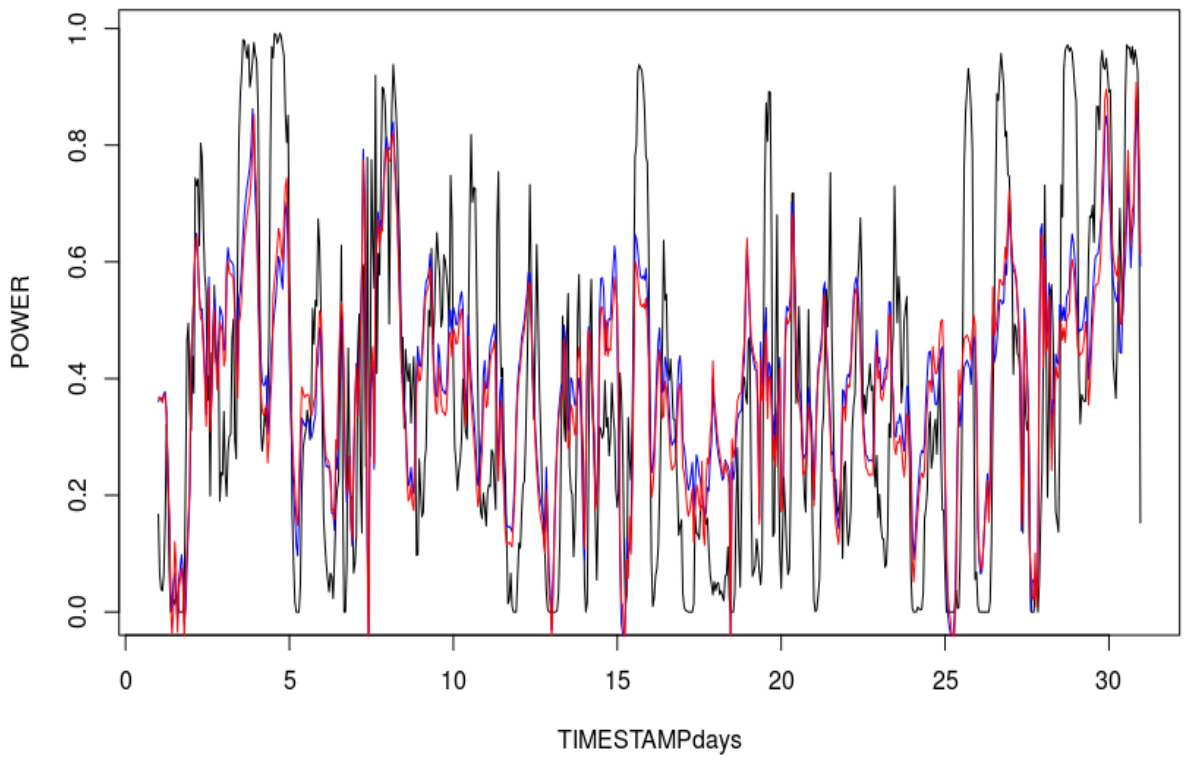

Illustration 6: True wind power measurement (black line), model.MLR.w power prediction (blue line), model.MLR.wd power prediction (red line).

As can be seen, the best predication (smallest RMSE meyric) delivered "MlR.wvu" model, meaning that the direction of the wind plays an important role when wind power is predicted. The correlation between wind directions and generated power can be seen in the illustration 6.

Task 2- further investigation

Surprisingly, model “Model.MLR.wd”, which uses “direction” as an independent variable is not as good as the model which uses “U10” and “V10”, despite the fact that “direction” is a transformation of “U10” and “V10”. The correlation between the POWER and the direction is also not as strong as between the POWER and V10 or the POWER and U10

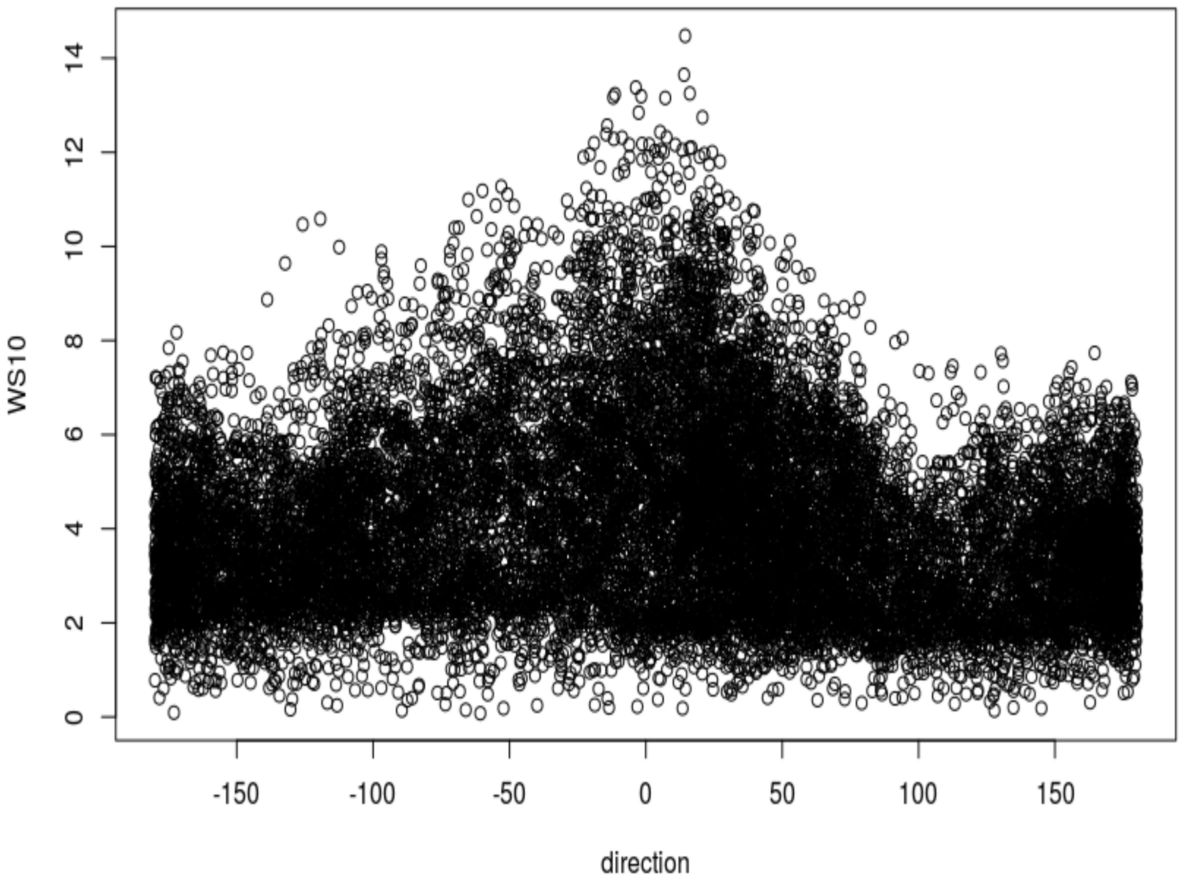

Illustration 7: Dependence between wind speed and wind direction

The above graph shows dependence between wind speed and wind direction. The conclusion from the chart is the fact that WS10 (wind speed) contains some information about the direction of the wind. For example, when the value of wind speed is more than 10 m/s, the wind blew from the East (0 degrees- easterly wind). This fact explains why adding an extra parameter (direction) to the model does not improve prediction significantly.

Task 3

In this section we will present a brief analysis of the wind generation data without the use of other data. This is because we do not always have the information about the weather data or the wind speed data. We will compare the predicted wind power and the real wind power measurements. The power forecasting production has been made. One month forecasting has been estimated based only on historical power data. Three models have been proposed- standard linear regression model (LR), recurrent neural network (RNN), and so-called null model.

Model name	**RMSE**
Linear regression (LR)	0.2932368
Recurent neural network (RNN)	0.125
Null model	0.2978971

The training data for each model is the value of POWER generated by turbine against time. The historical data cover the time period from 01.01.2012 to 31.10.2013, whereas the prediction covers the period of 30 days between 01.11.2013 to 30.11.2013. The LR model has been made by using R language in the R-studio programming environment, but RNN has been written in Python. Those two languages suggest a bit different ways of dealing with input data.

For the RNN the data, first of all, should be scaled, as the LSTM expect data to be in range between -1 and 1. This is a consequence of usage of tanh activation function within the network. In this case, the data (the POWER level) has already been normalized. But, since normalization process transform data to the range between 0 and 1, it had to be scaled anyway. The important thing to do after making the prediction was to invert the scale back to the previous unit.

For the LR model, only the TIMESTAMP data has been encoded. The very first TIMESTAMP value has been changed to 0, and each next observation got its value as a number of days that past from the first data point (for each hour it is a fraction of a day). For example 2012-01-01 01:00:00 got value 0, 2012-01-01 02:00:00 got value 0.04166666667, and so on. The new TIMESTAMP got its new name TIMESTAMP2.

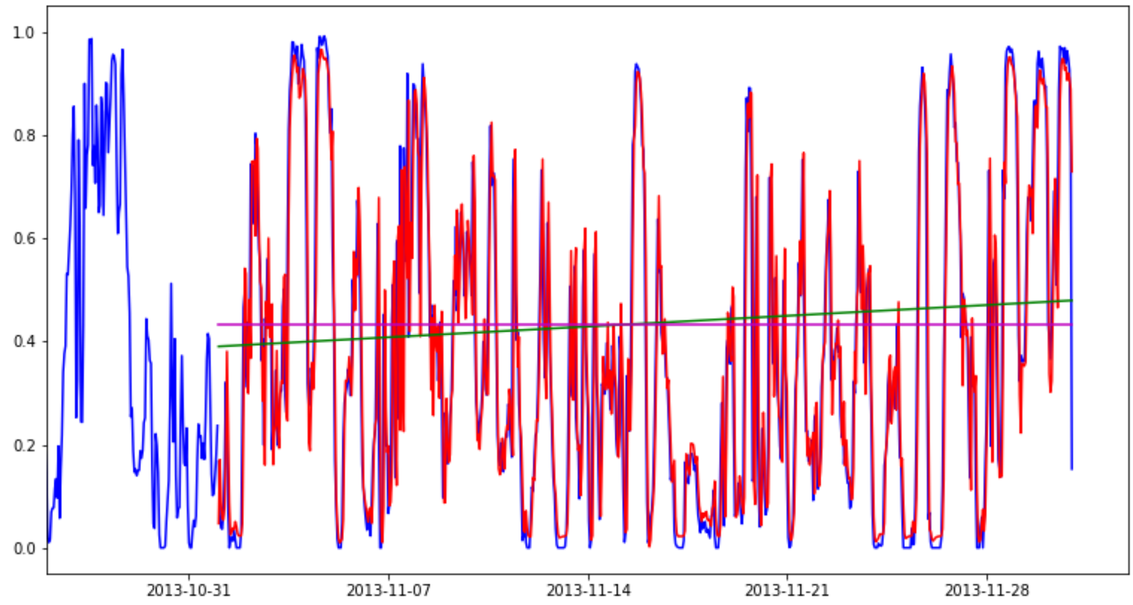

Illustration 8: True wind power measurement (blue curve), LSTM-RNN prediction (red curve), Linear model (green line), Null model (magenta line)

As the LSTM-RNN model has been developed in python and Linear Regression (LR) model in R, the common data plotting LR model had to be migrated to python. To achieve that, the data obtained in R had to be generated in python by using LR model's coefficients- β0 and intercept. In this task, it has also been developed a so-called null model, for the purpose only of showing how poor predictions the linear model delivered in this case.

Conclusion

In this report, we used different machine learning techniques to get more insight into the wind energy forecasting. Based on the historical data we analyzed the wind power and built a training model. In the first task, we concentrated on the relationship between the power generation and wind speed. As mentioned above, the result from the machine learning techniques were very similar, but the SVR method had the best RMSE compared to the others. This can be analyzed as different approaches used by each model for estimation. In task two, we could conclude that wind direction has an important role to predict wind power. This will further explain why adding an extra parameter to the model will not improve the prediction significantly. In the third part, we analyzed only the wind generation data with no other parameters. We developed LSTM-RNN model in Python and the LR model was programmed in R. Furthermore, by using LR model's coefficients- β0 and intercept we could achieve the data obtained in R to generated in Python. We also developed a zero model illustrated in figure 8, to illustrate how bad predication of the linear model delivers.