INF5870 - Assignment 2 Report

Assignment submitted in partial fulfillment of the requirements for the ***Energy Informatics*** *course at the* ***Institute of Informatics, University of Oslo****.*

Authors:
Zahra G. Yndestad
Khalil Abuawad
Marius E. G. Andresen
Christopher A. Trotter

May 18, 2018

Contents

1 Weather Data

Before analyzing the results from applying the machine learning techniques, we will briefly discuss the *type of problem* we are attempting to solve. It will provide us with the necessary intuition for further discussion of *the validity of the models* based on the data provided in the assignment. The type of problem we are concerned with is a *prediction problem*. The goal of a prediction problem is to give the correct label(e.g. prediction or output) to an instance (e.g. context or input). As mentioned in [7], the general prediction paradigm is then

- find a representative set of m instances of the problem, $u_1, ..., u_m$.
- human (the "teacher") provides the correct labels $y_1, ..., y_m$
- each (u_i, y_i) pair is a "labeled example"
- ML algorithm attempts to identify the simple hypothesis which explains the relationship between the inputs and outputs.

This latter approach lends itself well to *generalization*, which is the idea of looking at m examples, *identifying a hypothesis*, and applying it to the next example. Identifying a hypothesis is trivial, but the challenge is that there might be infinitely many hypotheses associated with the same data, and we have to choose something, so choose the simplest one.

When considering the machine learning techniques, mentioned in the assignment, they are concerned with providing examples rather than finding the actual solution, and this is done by *learning* from earlier examples. It is an attempt to understand and go beyond the ability to explain something that has already been seen. Intuitively, understanding is closely related to the capability of predicting what has yet to be observed. In other words, generalization is a form of understanding.

When considering the validity of the model, we have chosen to apply the model validation technique known as *cross-validation*. It is a technique for assessing how the results of a statistical analysis will generalize to an *independent data set*. It is mainly used in the settings where the goal is prediction, and one wants to estimate how accurately a predictive model will perform in practice. If the order of the data is important, then cross-validation might be problematic for *time-series* models. We will then attempt to use an appropriate approach such as *forward chaining* to validate the model.

We will be limiting the scope of the prediction problem to only consider regression problems since the assignment limits the machine learning techniques to supervised statistical learning methods.

2 Question 1

We will provide a formulation of the various models based on the data from the assignment. This includes the training data since we are considering only the model. Further, we hope to make a few statements on the validity of the model based on *cross-validation*. Finally, we will provide the *Root Mean Square Error*, also known as *RMSE*, equation as well as the results from this equation. It will allow us to have a short discussion of the results produced from the different models.

2.1 Linear Regression

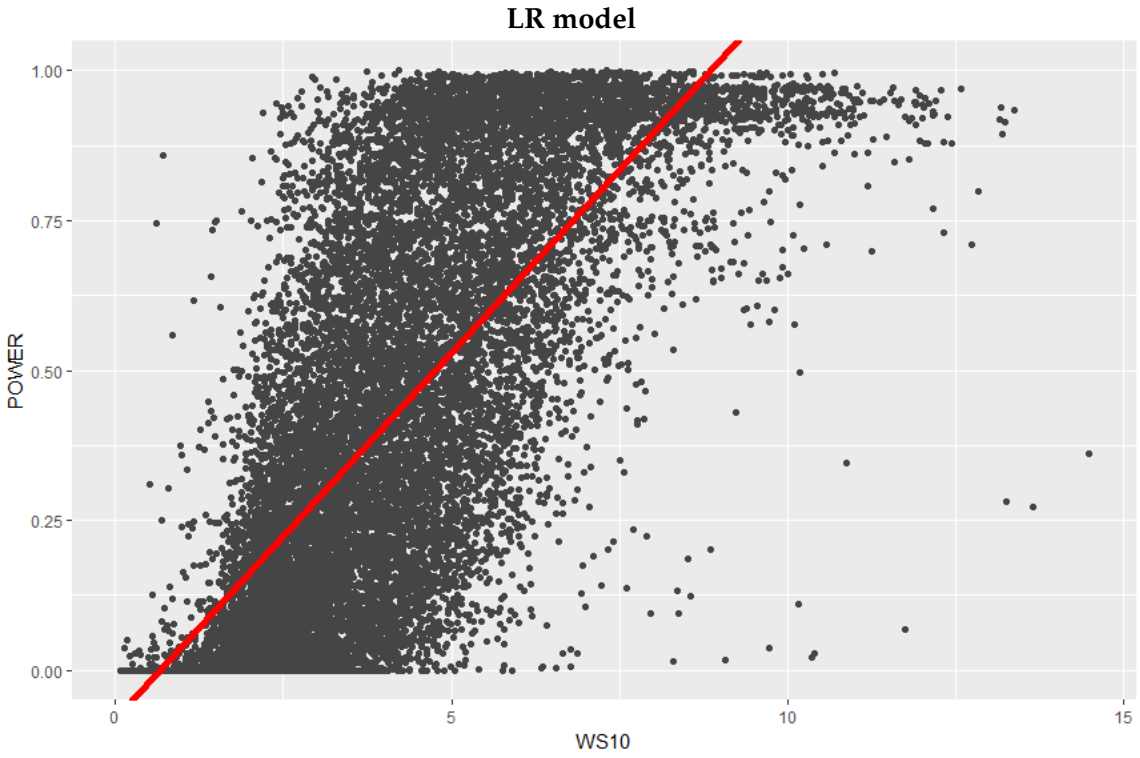
The general formulation of a linear regression problem is

$$y_p = \beta x + C. \tag{1}$$

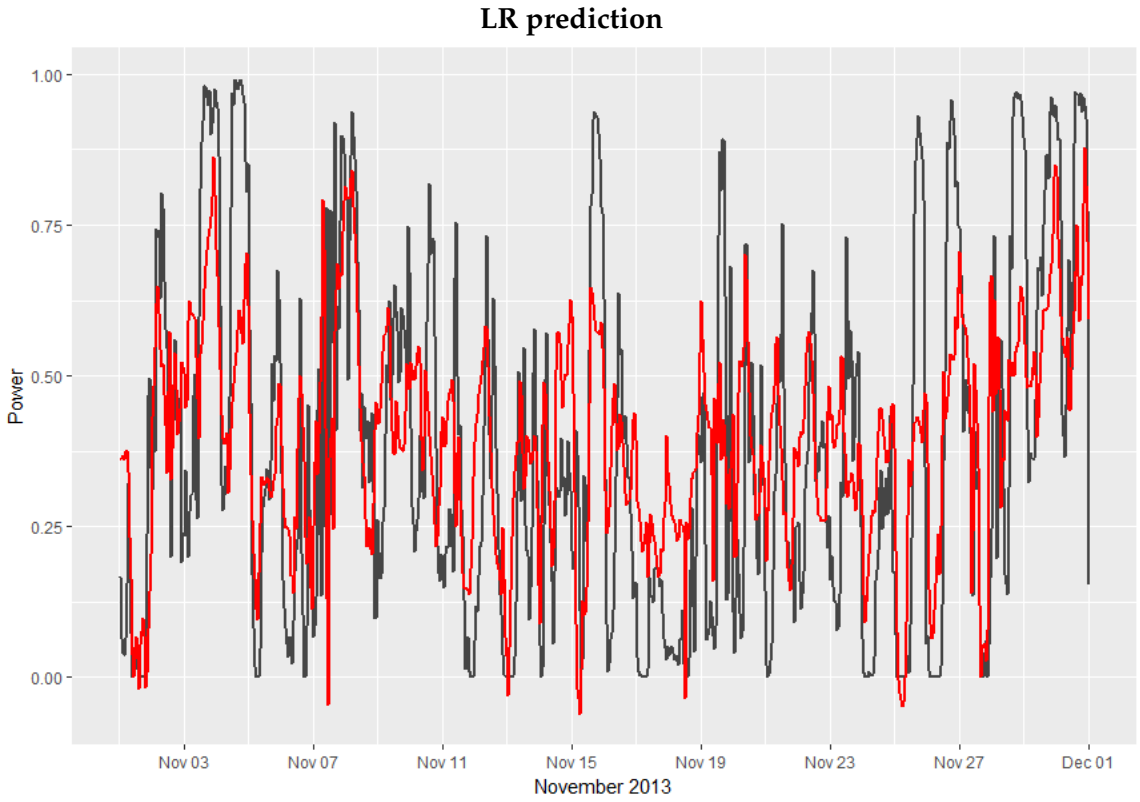
In our case the formula would be

$$Wind\ power_p = \beta \times Wind\ speed + C \tag{2}$$

where $\beta = 0.1225934$ and $C = -0.08367458$, after training the model on 16080 data points from the training data. Below is a figure which represents the linear regression of the above formulation.



Another figure below shows the prediction of wind power from the linear regression model, where the red line is the prediction and the gray is the actual wind power. The coloring of the lines will remain the same for the other models. As for the error value, the RMSE ≈ 0.216. A table of the RMSE value of the different models will be presented after showing the other machine learning techniques.



2.2 K Nearest Neighbors

K nearest neighbors algorithm stores all available cases and predicts the numerical target based on a similarity measure (e.g. distance function). The most common distance function is

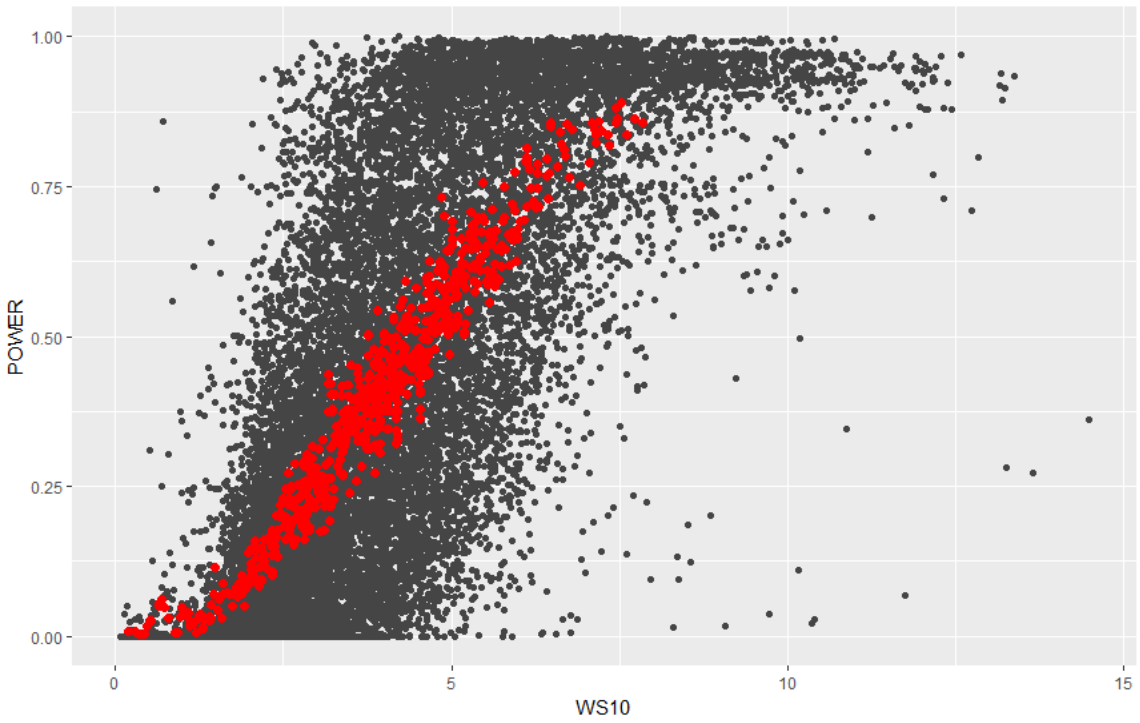
$$Euclidean = \sqrt{\sum_{i=1}^{k}(x_i - y_i)^2}. \tag{3}$$

In our case the formula would be

$$WindPower_p = \sqrt{\sum_{i=1}^{16080}(wind\ speed_i - wind\ power_i)^2}. \tag{4}$$

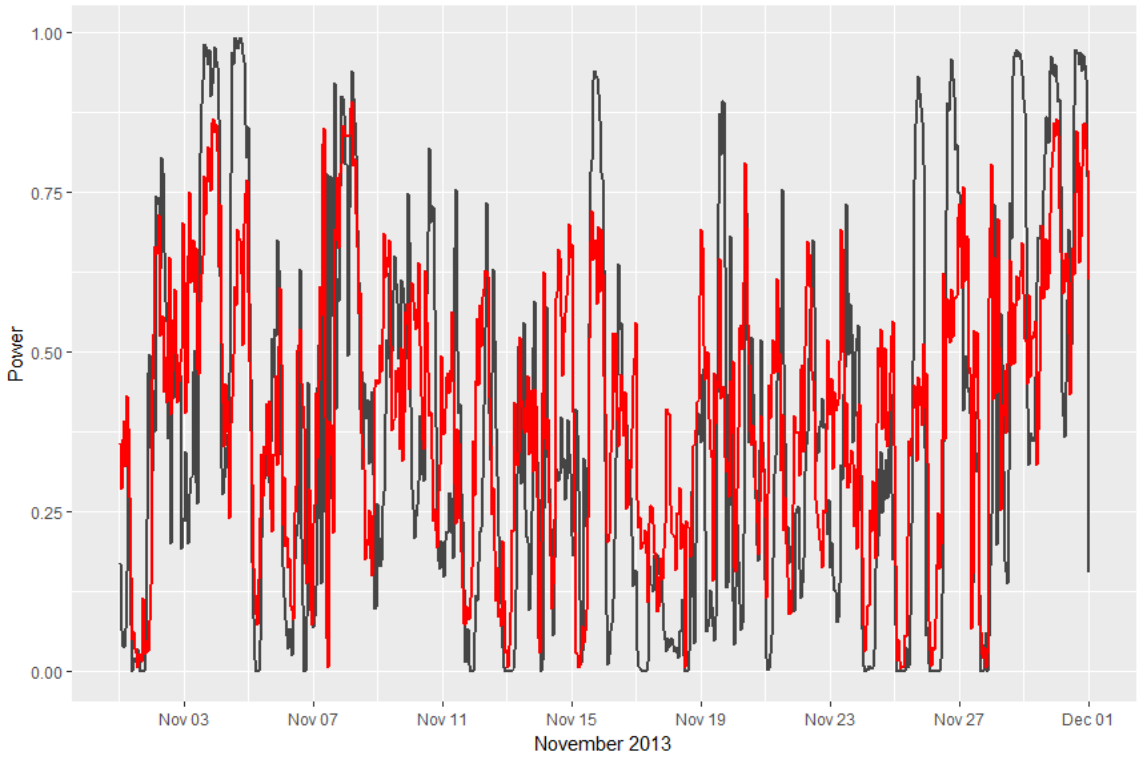
Where $wind\ speed_i$ and $wind\ power_i$ refer to the corresponding columns within the training data, and 16080 is the number of data points in the training data. The optimal number of k neighbors after using *cross-validation* of 10-fold is k = 23. As mentioned earlier, we will give a justification for the validity of the model after introducing the other machine learning techniques. Below is a figure which represents the optimal k nearest neighbor.

KNN model



When KNN is used for regression problems the prediction is based on the mean or the median of the K-most similar instances. Furthermore, we have another figure below which shows the prediction of wind power from the KNN model. Where the coloring scheme is as mentioned before. As for the error value, the RMSE ≈ 0.221.

KNN prediction



2.3 Support Vector Machine

In our assignment, we will only consider the type of support vector machine used in regression problems, known as SVR. The goal is to find a function $f(x)$ that deviates from y_n by a value no greater than ϵ for each training point x, and at the same time is as flat as possible. We are to consider the general linear regression formula as mentioned earlier.

For the general linear regression function, flatness means that we wish to find a small β, which can be described by the following equation

$$min\ \frac{1}{2}\ \beta^2 \tag{5}$$

such that

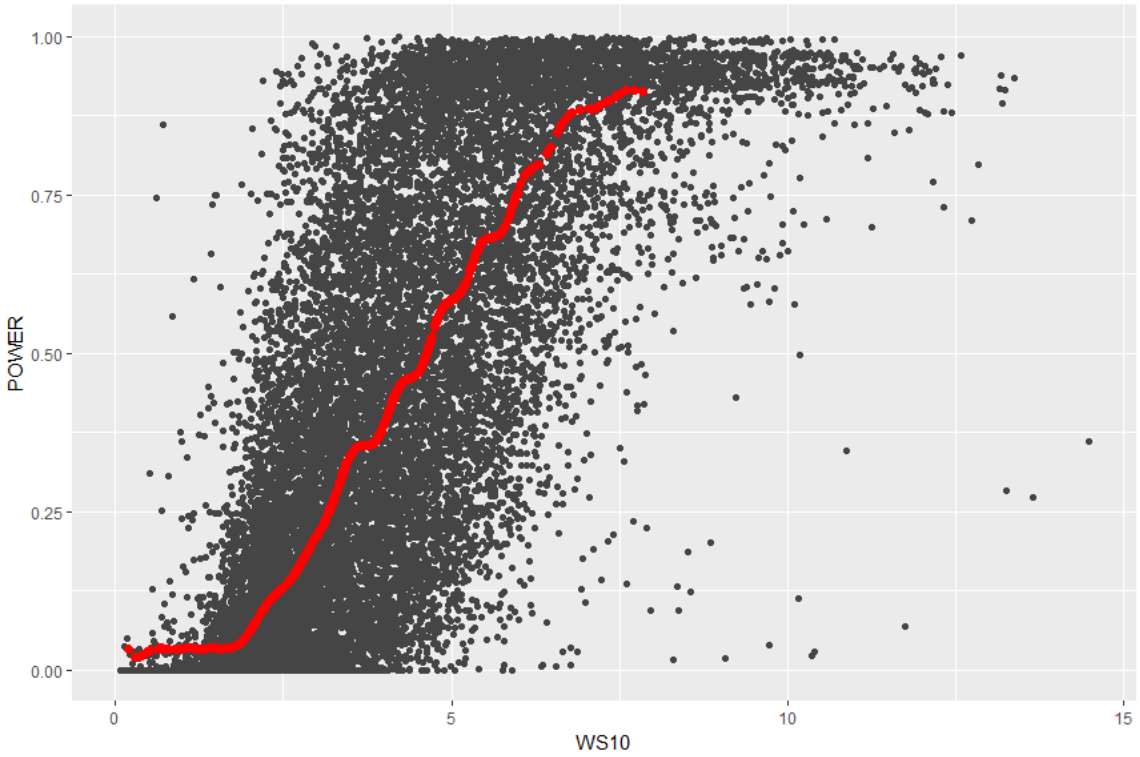
$$y_i - (\beta \times x_i + C) \leq \epsilon$$
$$(\beta \times x_i + C) - y_i \leq \epsilon$$

holds. We wish to have flatness to ensure that we avoid the over-fitting problem. Making it so that function is less sensitive to x and the change of x, as mentioned in [6]. When over-fitting it matches the training model, but will be a poor model for predicting new data. In our case, the epsilon would be determined by the constraints

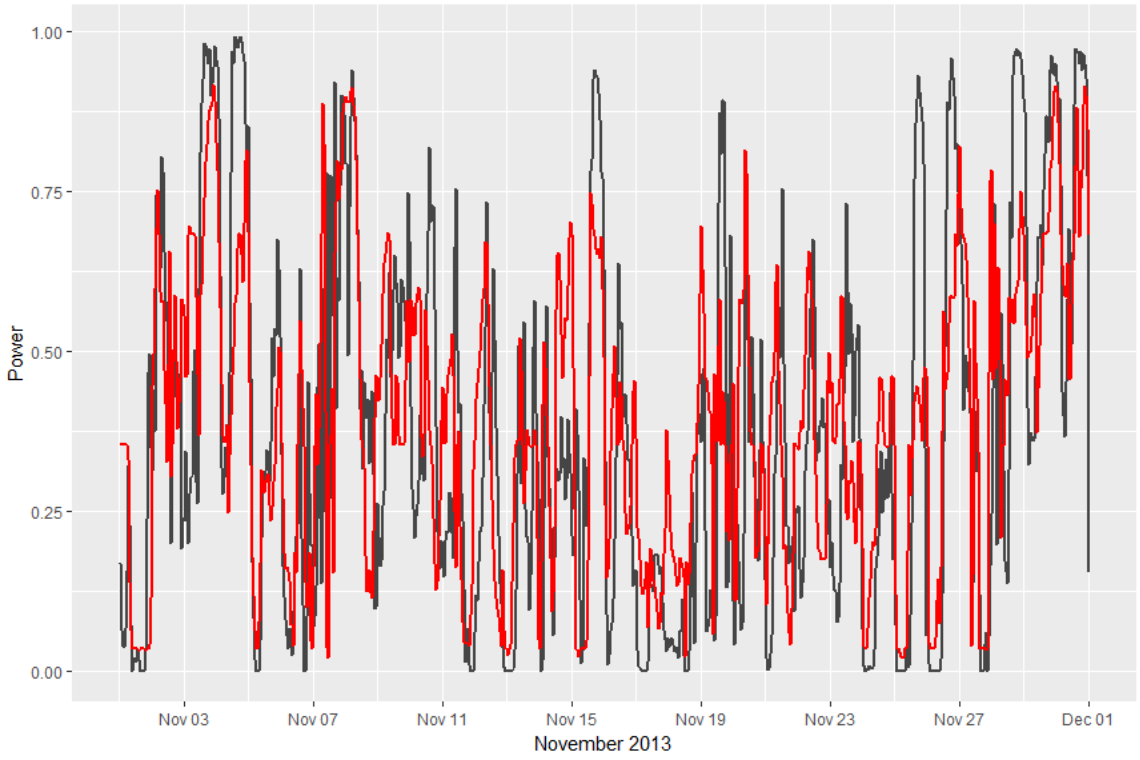
$$wind\ power_i - (\beta \times wind\ speed_i + C) \leq \epsilon$$
$$(\beta \times wind\ speed_i + C) - wind\ power_i \leq \epsilon.$$

Below is first the SVR model and secondly the SVR prediction figure. First, the SVR model represents the linear formula which respects the constraints mentioned and in this case is demonstrated by using the *svmRadial* method, which is available from the popular *R*-library *Caret*. We shall give a brief justification, at the end, for the use of *svmRadial* method. Secondly, we have the figure which shows the prediction of wind power from the SVR model. The coloring scheme remains the same as the other models above. As for the error function, the RMSE ≈ 0.214.

SVR Model



SVR Prediction



2.4 Artificial Neural Network

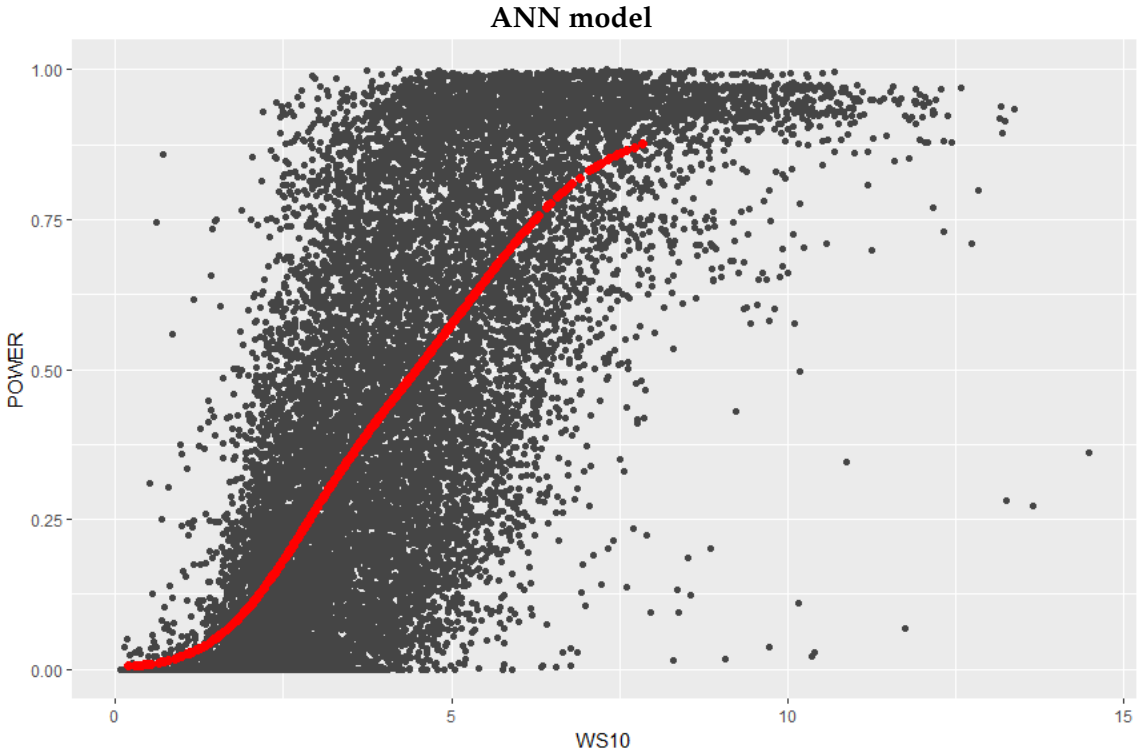
The general formulation of a neural network is

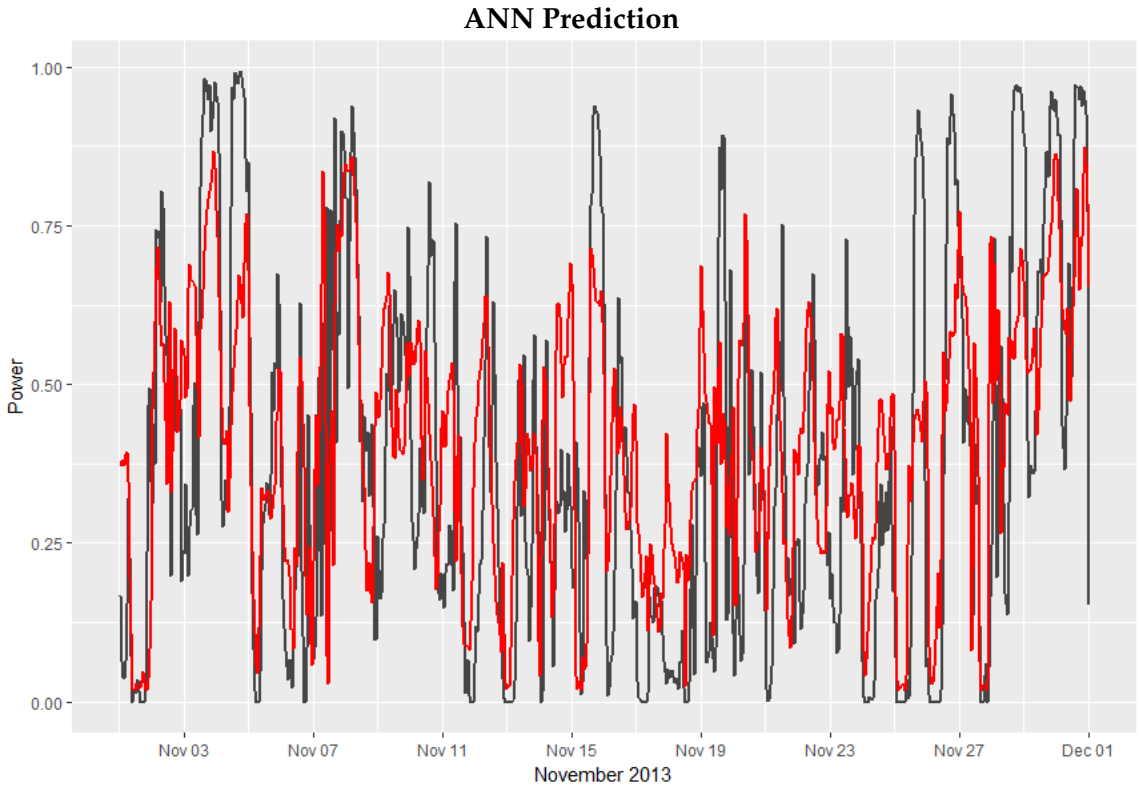
$$y = \sum_{i=1}^{m} (\beta_i x_i) + \epsilon \tag{6}$$

such that

$$f(x) = \begin{cases} 1 & \text{if } \sum \beta x \geq 0 \\ 0 & \text{if } \sum \beta x < 0 \end{cases}$$

holds. This is the formulation provided from the lecture notes [2]. Artificial Neural Network, known as ANN, suffers from the same problem as SVR which is over-fitting when considering large amounts of data. We will give a justification at the end for what may occur when over-fitting the model and discuss the validity of the model. In our case, the variable X is replaced with the *wind speed* in the formulation above. Besides that, everything remains the same. Below is both the ANN model and prediction figure shown in a similar fashion as SVM. As for the error function, the RMSE ≈ 0.216





2.5 Comparing prediction results of machine learning techniques

For our comparative analysis, we are to consider the error function RMSE which is described by the formulation

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i^p - y_i)^2}. \tag{7}$$

Where y^p is the predicted values and y is the actual values [3]. The accuracy of the predictions from the model is determined by this error function. A corollary to this is that we have a method to determine the cost and stability of wind power. When determining the accuracy of the models it is equally important to consider the context and use of the data. To be able to have a fair comparative, the data should be normalized and validated by the same methods. In our case we have chosen to use *cross-validation* and test every model on the whole training set. Hence, the results.

	LR	KNN	SVM (regression)	ANN
RMSE	0.2163841	0.2216553	0.2147279	0.2166363226

TABLE 1: RMSE results from each run of the model

We can see from the table that the model with the lowest RMSE is the SVM model. When considering a SVM model it is important to consider its tendencies to over-fitting the training data. Usually, the model of SVM will be more generalized after being exposed to large training data sets. As mentioned earlier, this also occurs within the ANN model. When

regarding how KNN evaluates the training data set, then it is important to know that finding the optimal k value is what determines a valid model. As the K value increases, the prediction will often lead to a over-fitting problem and the model will be generalized. Lastly, linear regression is a model which is least likely to be prone to an over-fitting problem, since the model considers a general mean between every single data point within the prediction. It is however a poor model for determining extreme data such as antipodal data. A model which is to be considered a valid model must determine the accuracy of the prediction such that it minimizes the over-fitting and generalization of the data. Therefore, the validity of SVM and ANN can be considered poor if it is exposed to large quantities of training data. In our case, we are to consider 16080 data points and to determine if these models are valid based on the data would require even further in depth analysis. However this would be considered outside the scope of this assignment. Although we have given a brief justification in the introduction.

3 Question 2

3.1 Multiple Linear Regression

The general formulation of a multiple linear regression problem is

$$y_p = \beta_1 x_1 + \beta_2 x_2 + ... + C \tag{8}$$

which is similar to the LR model, but may take multiple variables into consideration. In our case the formula would be

$$y_p = \beta_1 \times Wind\ speed + \beta_2 \times Wind\ direction + C \tag{9}$$

where $\beta_1 = 1.235e-01$, $\beta_2 = -6.966e-05$ and $C = -7.463e-02$, after training the model on 16080 data points from the training data. We determine the *wind direction* by

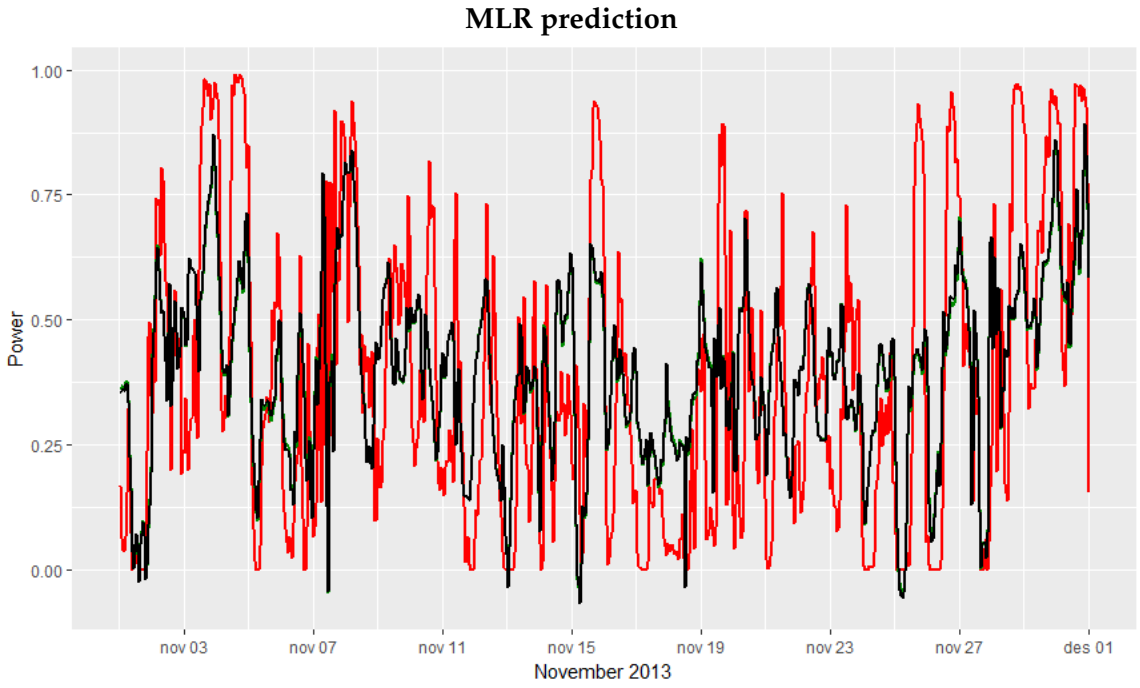
$$Wind\ direction = (270 - atan_2(V10, U10) \times \frac{180}{\pi}) \ \% \ 360 \tag{10}$$

where $V10$ and $U10$ is the data from the training and validation set, known as *training_data* and *weather_forecast_input*, and the formula is given in [8]. The symbol "%" in this case refers to the arithmetic function *modulo*. When considering what may generate wind power, it is important to understand that it may not only depend on wind speed, but multiple variables such as temperature, wind direction and pressure. Below is the provided accuracy results from running our model on the two variables mentioned in the formulation above.

	LR	MLR
RMSE	0.2163841	0.2149417

TABLE 2: RMSE from LR and MLR model

As for the figure below, we have that black is the MLR model, red is the LR model and green is the actual wind power for the time period.



Regarding the figure, it can be obscure which of the models are closest related to the actual wind power, but from the table above we can see that the MLR model outperformed the LR model. The reason for this is that by using multiple variables it may increase the prediction accuracy for wind power, and in our case we can see that wind direction has a positive effect on the accuracy estimation of the wind power. Furthermore, if we also considered temperature and pressure, for example, we could estimate the accuracy of the wind power on a granular level. Providing us with ideal conditions to have a valid model for estimating the accuracy of the wind power.

4 Question 3

When considering the formulation, from lecture [4], it provides the necessary description of the problem, but may require additional intuition to grasp the concept of recurrent neural networks, also known as RNN. In Richard Socher's lecture notes [1] we have found a simple RNN formulation:

$$\begin{aligned} h_t &= Wf(h_{t-1}) + W^{(hx)}x_{[t]} \\ y_t^p &= W^{(S)}f(h_t) \end{aligned} \tag{11}$$

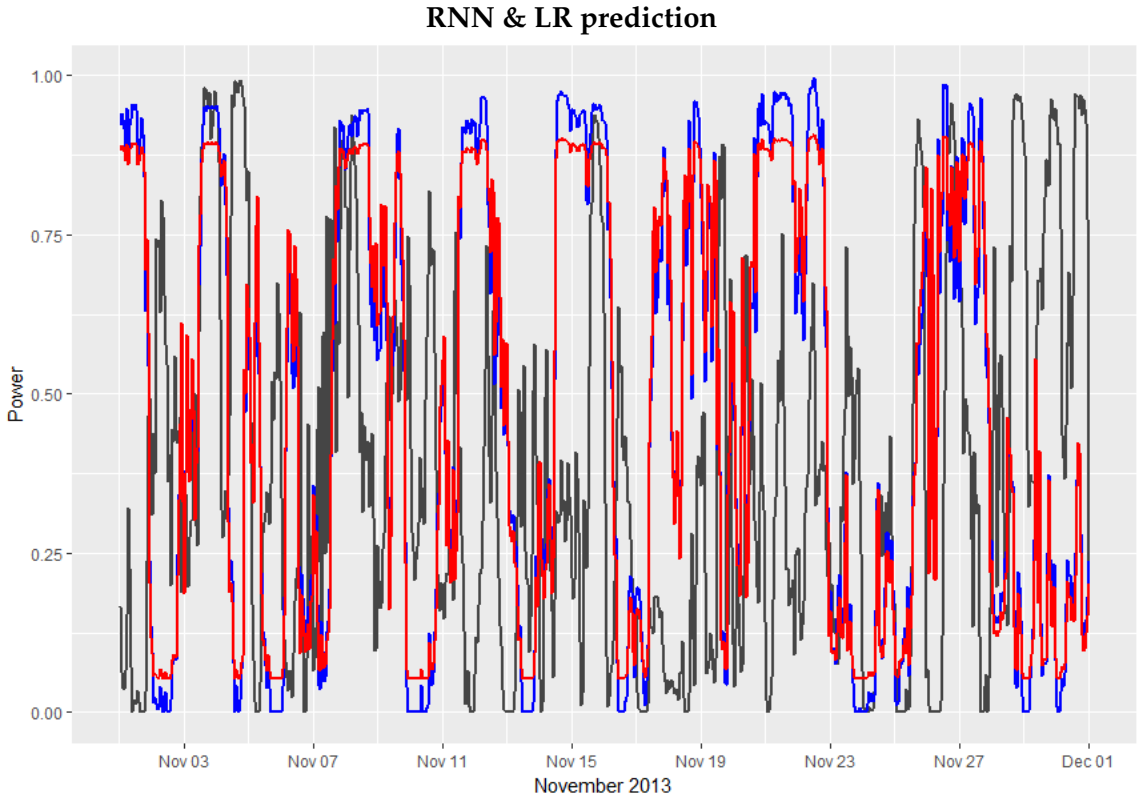
In our formulation, we would replace x with the wind power, which we can denote as $wind^p$. Where we are concerned with relating the hidden layer state h_t with its past layer state h_{t-1}, as described in the lecture notes [4]. Also, the hidden state h_t is calculated based on the previous hidden state h_{t-1} and the input $wind_t^p$ at the current time step. When considering the output y_t^p it is calculated based on the memory h_t at time t.

For our RNN model we were required to partition the training data into two segments, since we were to provide a prediction of power based on power. The first partition was used to train the RNN model with 15000 of the first data points from the training data. As for the remaining data points, they were used to predict the power. The reason for separating the data in such a way is because we may not always have other forms of weather data available such as wind speed, temperature, pressure and wind direction. Our RNN model was heavily inspired by the RNN model provided from the lecture notes in [5]. We were to compare our RNN model with a LR model which partitioned the data in a similar fashion. It provided us with the follow accuracy results:

	LR	RNN
RMSE	0.4792255	0.4725564

TABLE 3: RMSE from LR and RNN model

Finally, we were to plot a figure of the power prediction of the RNN and LR model with the actual power. We have that red is the RNN model, blue is the LR model and grey is the actual wind power for the time period.



5 Few final regards

Caret, the library used to implement the machine learning techniques, from the assignment, made it simple for anyone with prior statistical knowledge to use it without proficient programming abilities. For a computer science student, the black box, or abstraction, over the machine learning functions may cause a lack of understanding of the underlying mathematics. Especially, when wanting to understand the formulations underpinning the

machine learning techniques from the assignment. If one is to fine tune the parameters to their optimal performance, then it requires *a priori* knowledge of the algorithms and an in depth understanding of the context it may be applied.

A File structure

A.1 File structure

Regarding our files the structure is as follows:

1. data - contains both the test data and the output from the weather forecast predictions.
2. documents - contains the report.
3. figures - contain the plots from the assignment.
4. R - contains the code files: aNN.R, kNN.R, LR.R, MLR.R, RNN.R and SVR.R.

The file structure of the assignment where the R project is located is as following.

1. /Assignment 2/Machine learning/Weather Forecast/Weather Forecast.Rproj.

Question 1 is divided into 4 different files, aNN.R, kNN.R, SVR.R, LR.R.

Question 2 only has 1 file called MLR.R. Here we solve multiple linear regression and then just use the same code for linear regression as used in question 1 LR.R. Thus, plotting a figure with three curves.

Question 3 has also only one file called RNN-LR3.R

Predicted forecast data we compile under the run of the code is in the folder predicted forecast which can be found in the folder called -data.

B How to run the program

To be able to run most of the code you will have to install the packages correctly; caret, rnn and neuralnet. We have done it so that it gets installed automatically when ran, but there might be some slight hiccups.

Open the "Weather Forecast.Rproj" file in RStudio. To execute the line of source code where the cursor currently resides you press the Ctrl+Enter key (or use the Run toolbar button). To run the entire document press the Ctrl+Shift+Enter key (or use the Source toolbar button), and press "Run" in R.

References

[1] *CS224d - Deep NLP: Recurrent Neural Networks. Geographic wind direction*. 2016. URL: http://cs224d.stanford.edu/lectures/CS224d-Lecture8.pdf (visited on 05/14/2018).

[2] *INF5870 - Deep Learning for Energy Forecasting. From Biological Model to a Single Artificial Neuron (II)*. 2018. URL: http://folk.uio.no/yanzhang/INF5870-2018/DeepLearningforEnergyForecasting-Lecture10.pdf (visited on 05/13/2018).

[3] *INF5870 - Deep Learning for Renewable Energy Forecasting. Metrics measures the model goodness in prediction*. 2018. URL: http://folk.uio.no/yanzhang/INF5870-2018/DeepLearningforEnergyForecasting-Lecture10.pdf (visited on 05/13/2018).

[4] *INF5870 - Deep Learning for Renewable Energy Forecasting. Recurrent Neural Networks: architecture (I-III)*. 2018. URL: http://folk.uio.no/yanzhang/INF5870-2018/DeepLearningforEnergyForecasting-Lecture10.pdf (visited on 05/14/2018).

[5] *INF5870 - Deep Learning for Renewable Energy Forecasting. R code to build recurrent neural network model*. 2018. URL: http://folk.uio.no/yanzhang/INF5870-2018/DeepLearningforEnergyForecasting-Lecture10.pdf (visited on 05/14/2018).

[6] *INF5870 - Machine Learning for Energy Forecasting. Support Vectors Regression (SVR)*. 2018. URL: http://folk.uio.no/yanzhang/INF5870-2018/MachineLearningforEnergyForecasting-Lecture9.pdf (visited on 05/14/2018).

[7] *Introduction and Models. Prediction Problems*. 2011. URL: https://courses.cs.washington.edu/courses/cse522/11wi/scribes/lecture1.pdf (visited on 05/13/2018).

[8] *Wind Direction Quick Reference. Geographic wind direction*. 2018. URL: https://www.eol.ucar.edu/content/wind-direction-quick-reference (visited on 05/14/2018).