Predicting national wind power output based on average wind speed using polynomial and decision tree regression

Matias Häggman
Aalto University

1 Introduction

Predicting consumption and production of energy in the national electric grid is essential, because electricity supply has to always meet the demand. Due to the variable nature of wind power, load-following power generation is required to adjust the grid to peaks and lows of wind power generation. Wind power forecasting is therefore important, since it offers crucial information to load-following powerplants, allowing them to increase or decrease power output in order to compensate for the variation in wind power [1]. Goal of this project is to build an algorithm with a capability to predict total national wind power output based on solely national average wind speed. After completion, this algorithm could be utilized to determine future wind power generation based only on given wind forecast at a certain period. In section 2 the problem is formulated in-depth, in section 3 methods of this project are disclosed and the section 4 is reserved for discussing the obtained results. Finally in section 5 the project is concluded.

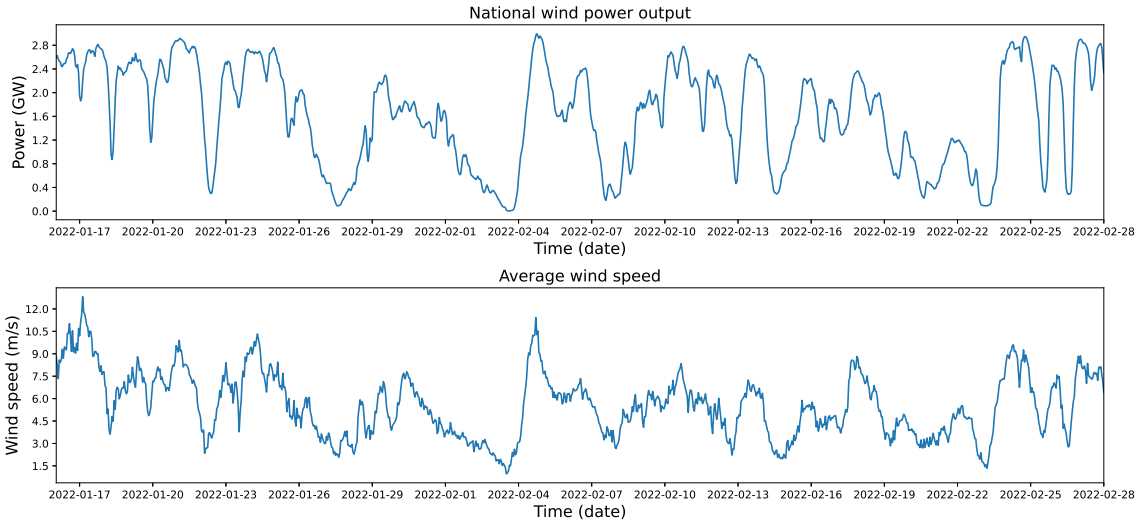

Figure 1: National wind power generation and average wind speed represented as time series.

2 Problem formulation

2.1 Data points, features and labels

Wind speed heavily correlates with the national wind power output, as can be observed from figure 1. We will hence construct our data points such that each **data point consists of wind speed (m/s, decimal value), time (date, integer value) and power output (GW, decimal value)**. Features and labels of this machine learning problem are defined as follows: **wind speed is the feature** and **wind power output is the label** we wish to predict.

2.2 Data collection

Since data on single wind farms is not easily accessible nor very predictable [3], we will use the total national wind power output available at Fingrid's website [4]. Wind speed data on the contrary is locally collected by weather stations and provided by the Finnish Meteorological Institute [5]. Most of wind power in Finland is produced on the western coast and in lapland as can be observed from figure 2, so data from several weather stations near big wind farms on the western coast of Finland and in lapland will be combined to produce a rough estimate of the predominant wind conditions.

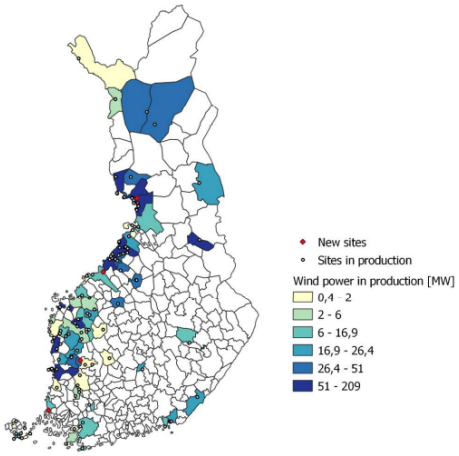

Figure 2: Wind power by municipality in 2019 [2].

3 Methods

3.1 Overview of the dataset and splitting

Pandas package [6] was used to preprocess and parse the data from the chosen sets of data. Scikit-learn package [7] was used to fit the model to data and matplotlib package [8] to visualize the data and results. Wind speed data was collected from five different observation stations near big wind farms, namely: Kalajoki (Ulkokalla), Kemi (Ajos), Pori (Tahkoluoto), Sodankylä (Tähtelä) and Suomussalmi (Pesiö). Numpy package [9] was used to construct a single vector of the mean values of all of the five different wind speed vectors previously mentioned. In addition to wind speed data, wind power output data was retrived from fingrid's database [4]. The time period of 16.1.2022 - 28.2.2022 was selected for the data collection, which gave us 1032 data points in total.

The "wind power generation - hourly data" column from fingrid's wind power output data file will be used as labels and "wind speed (m/s)" as the feature from weather observation station data files. The data set was split into training, validation and test sets using a single split as follows: **Training set 60% of the data, validation and test sets each 20% of the data respectively.** Single split was used, because the data set is relatively large. Ratio-wise the choise was made to spare at least 40% of the data for validation and testing to ensure that every aspect of the set would be sufficiently represented by the validation and test sets, as the data set is somewhat scattered and has outliers.

3.2 First model: polynomial regression with mean squared loss

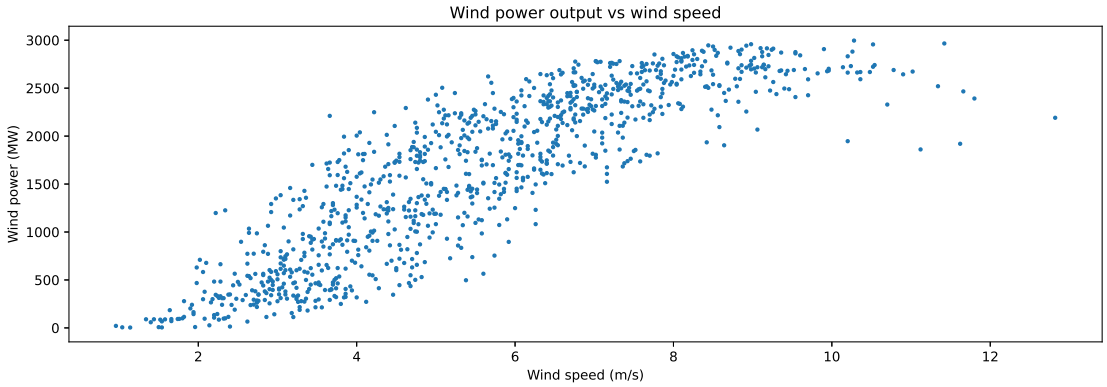

Figure 3: Wind power scatter plotted as a function of wind speed.

Polynomial regression was selected as the first model, since the data in question exhibits polynomial, non-linear characteristics as can be seen from figure 3. Any non-linear relation can be approximated to arbitrary accuracy using a sufficient set of polynomials h generated by the sum $h(x) = \sum_{i=1}^{n}(w_i x^{i-1})$ [10, p.83].

$$L = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{1}$$

The mean squared error (MSE) was chosen as the loss function, since by default polynomial regression learns a hypothesis by minimising the MSE and such they are commonly used together [10, p.83]. MSE is calculated by squaring the difference of points $(y_i, \hat{y_i})$ and dividing by the total number of points N, where y_i is the i:th true value and $\hat{y_i}$ the predicted value. Loss was calculated for both training and validation sets.

3.3 Second model: decision tree with mean squared loss

Albeit having general polynomial features, the data set in question is quite scattered and not evenly distributed along a curve. The image of a polynomial regression fit only consists of points along single continuous curve and as such could be insufficient to represent all of the data. Decision tree on the contrary generates set of piecewise discontinuous functions, i.e a tree, to compute a value for a given label. To learn a hypothesis, a decision tree regressor uses a certain loss function at each node to minimize the loss at each two child nodes. Using tree large enough, any non-linear map can be approximated [10, p.98]. However increasing the depth also results in a more overfitted model and more heavy computations. To determine the optimal tree depth, models with different depths are computed ranging from 2-20 and their train and validation errors plotted. Mean squared loss (1) was selected again as the loss function, as it is commonly used in applications of decision trees for numeric label regression [10, p.97].

4 Results

4.1 Polynomial regression

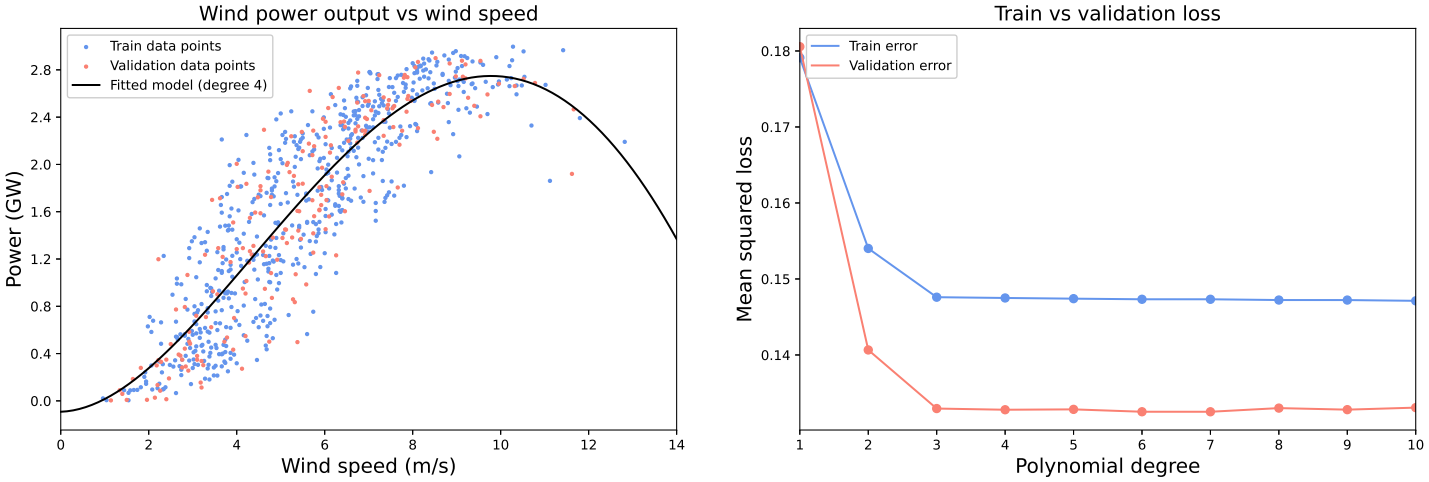

Figure 4: Results from fitting a polynomial of degree one to the data. In the second plot train and validation loss are plotted as functions of polynomial degree.

Fitting polynomials of degree 1 to 10 yielded the following result; training error and validation errors are minimized at degree $= 4$, but slightly increase with the increase of the polynomial degree after. A polynomial of degree 1 was also included to demonstrate the poor performance of a linear fit in context of this problem. Theoretically, increasing the polynomial degree also increases overfitting of the model [10, p.163], but as can be seen from figure 4 the validation and train losses remain almost constant for degrees of 3 to 10.

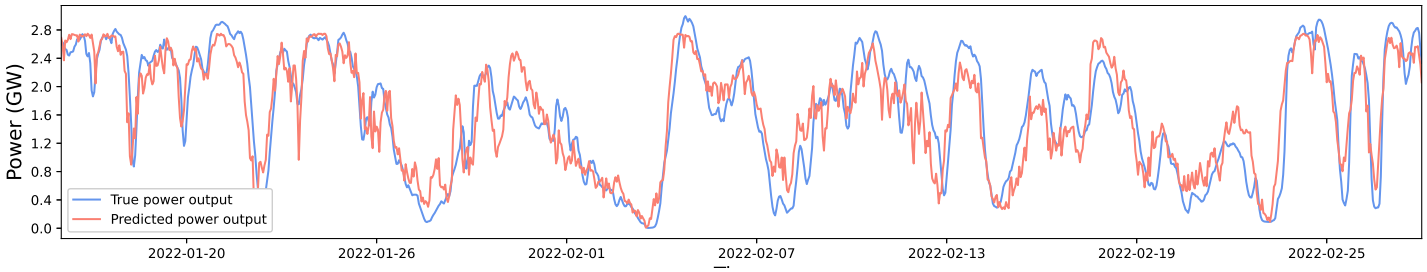

Figure 5: Results from a model of degree four plotted as time series with true wind power output.

4.2 Decision tree

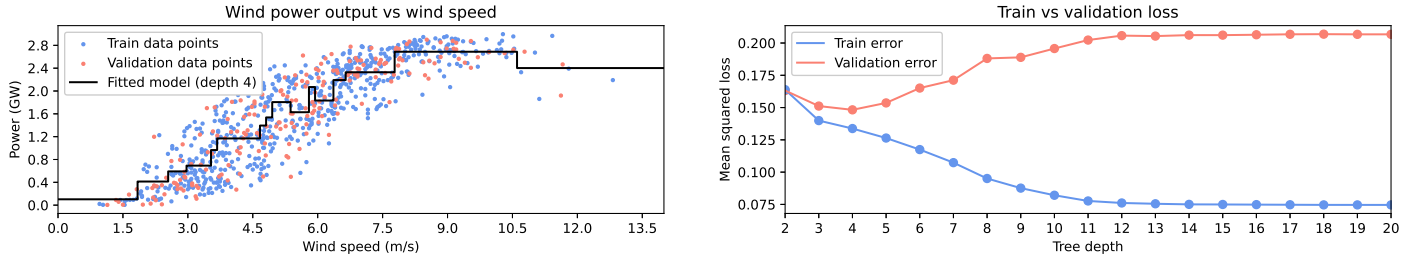

Figure 6: Results from decision tree regression with max. depth of four. In the second plot train and validation loss are plotted as functions of tree depth.

Before rapidly diverging, the validation and training losses minimize at tree depth of four as seen from figure 6. Increasingly accurate predictions could be achieved with a tree of depth n, but this would result in *heavy* overfitting

of the model. An overfitted model could almost perfectly predict labels of the specific set it was trained on, but would deliver poor performance with other sets of data.

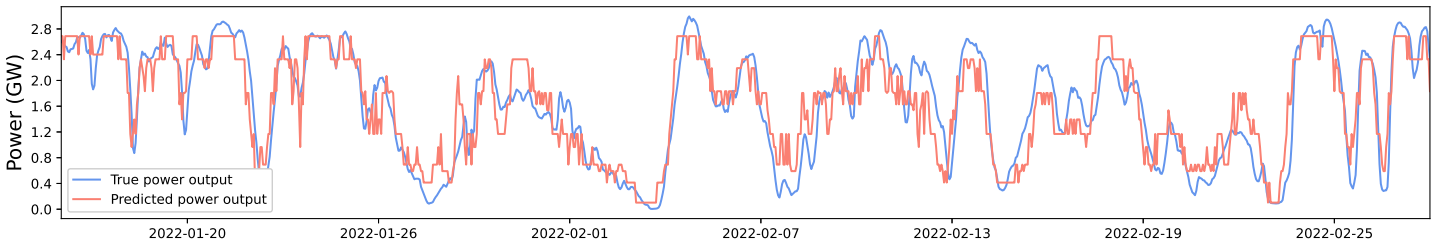

Figure 7: A tree model with depth of four plotted as time series with true wind power output.

4.3 Comparison and model selection

To avoid overfitting, the metric of quality in the context of a machine learning model is measured using the validation error [10, p.161] and thus the model with lowest validation MSE was chosen. As seen from figure 6 for all tree depths the validation error is higher than the train error, suggesting that even at best the model is overfit. For the polynomial model, the opposite is observed from figure 4, as the validation error is always lower than the training error.

Decision tree model with max depth of four yielded a validation MSE of 0.1482 while the polynomial model of degree four resulted in somewhat lower validation MSE of 0.1325, resulting in polynomial regression being chosen as the final model. To measure the quality of the selected model, test error was calculated by using the test set, which consists of data points that the algorithm has not yet seen. The test error for the chosen model was calculated as being 0.1428.

5 Conclusions

As a ML model, polynomial regression is very rudimentary and not able to take into account all qualities of the training data set. Moreover variables which directly affect the theoretical output of national wind power in addition to wind speed are at least; density of air and utilization rate (interruptions caused by malfunctions, repairs, extreme weather, etc) [11]. Air density ρ is furthermore determined by factors such as local temperature and humidity. Not only none of this was taken into account when constructing the training set and the algorithm, but also the way wind speed data was collected could be significantly improved. To be specific, instead of calculating the average wind speed from of all values from the chosen weather observation stations, a *weighted* average should have been calculated instead. With proper weight coefficients such calculation could emphasize each location's impact on national wind power output (i.e observations by wind stations near big wind farms have greater impact on the total wind power output than that of observations near small wind farms). Description of wind power forecast by fingrid: "The forecasting model contains the coordinates and capacity of every wind farm in Finland. The software uses the coordinates to fetch a weather forecast that is as localised as possible to the wind farm. By comparing the weather forecast with the actual generation figures under equivalent conditions, we are usually able to forecast the output very accurately." Considering all the shortcomings mentioned before, the obtained results and accuracy of the model as displayed in figures 4 and 5 are relatively satisfactory.

References

[1] fingrid. *The balance of the electricity system requires substantial forecast data.* Available at: https://www.fingridlehti.fi/en/substantial-forecast-data/#cf136595.

[2] VTT. *Capacity factors of wind power 2019.* Available at: https://www.tuulivoimayhdistys.fi/media/finland-capacity-factors-2019.pdf.

[3] International Energy Agency. *Variability of wind power and other renewables.* Available at: https://web.archive.org/web/20051230204247/http://www.uwig.org/IEA_Report_on_variability.pdf.

[4] Fingrid. *Wind power generation.* Available at: https://www.fingrid.fi/en/electricity-market/electricity-market-information/wind-power-generation/.

[5] Finnish Meteorological Institute. *Download observations.* Available at: https://en.ilmatieteenlaitos.fi/download-observations.

[6] Pandas. *fast, powerful, flexible and easy to use open source data analysis and manipulation tool, built on top of the Python programming language.* Available at: https://pandas.pydata.org/.

[7] scikit learn. Available at: https://scikit-learn.org/stable/.

[8] Matplotlib. *Visualization with Python.* Available at: https://matplotlib.org/.

[9] Numpy. *The fundamental package for scientific computing with Python.* Available at: https://numpy.org/.

[10] Alexander Jung. *Machine Learning: The Basics.* Springer, Singapore, 2022.

[11] Wikipedia. *Wind turbine.* Available at: https://en.wikipedia.org/wiki/Wind_turbine.