Python Project:
Evaluating Wind Turbine Performance Improvement Using Regression

Andreo II Lozada

INTRODUCTION

Background

A renewable energy company (the "Owner") operating a fleet of wind farms has engaged with a company (the "Seller") that develops and installs wind turbine blade hardware claimed to improve aerodynamics, resulting to higher energy generation performance of wind turbines. As a pilot project, these blade upgrades were done to a few of the Owner's wind turbines. The Owner wants to evaluate and quantify the improvement in the pilot wind turbine's energy generation performance to decide if they should contract the Seller to do the upgrades to the rest of the fleet. However, the energy generation before and after improvements are not directly comparable due to differences in resources (wind speed).

Objective

This project aims to model the wind turbine power generation before and after execution of blade improvement activities using regression, excluding any other factors that may affect the turbine generation such as difference in wind speed and other weather conditions, wind turbine stoppages, and power output curtailment due to grid limitations by outlier detection and removal, and compare the predictions of the models to determine whether there is an improvement in generation. Any improvement should be quantified in terms of annual energy production by generating synthetic data to represent annual variables and use the model to predict the generation. Although several variables will be tested, a hypothesis is that wind speed will be the most significant variable to model power based on wind power curves.

DATA PREPARATION

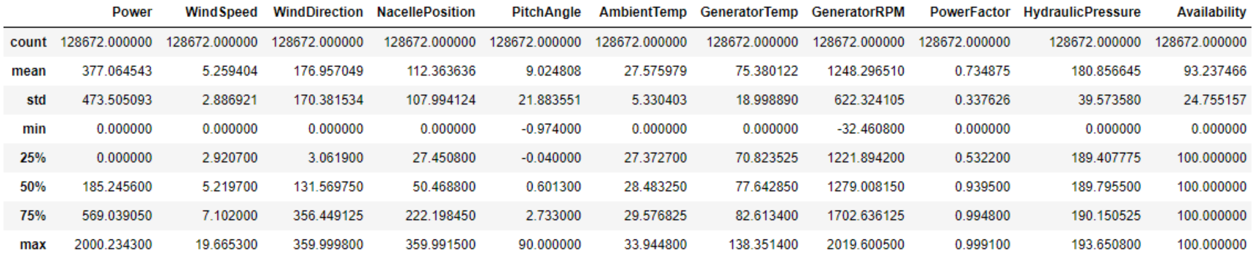
Raw data from the wind farm's Supervisory Control and Data Acquisition (SCADA) system was obtained from the Owner with permission (see *Appendix 1* in the Report folder), with brief descriptions of each column from the turbine manufacturer. To model the turbine performance before and after improvements separately, 30 months of the most recent available 10-minute data (at the time of this project) was used to cover 1 full year before improvements, several months of available data after the improvements were implemented, and the remaining unused data for Out-of-Sample predictions for testing the model.

The SCADA data includes several columns, some of which can already be determined as not useful to the project, such as turbine ID and turbine number. Manually chosen columns include:

- *DateTime* – date and time of recording
- *Power* – power output of wind turbine, in kilowatts (kW)
- *WindSpeed* – wind speed recorded by the anemometer at the nacelle, in meters per second (m/s)
- *WindDirection* – wind angle from north measured by the wind vane at the nacelle, in degrees
- *NacellePosition* – angle of the nacelle from north, in degrees
- *PitchAngle* – blade angle where zero is parallel to the wind, in degrees
- *AmbientTemp* – air temperature measured at the nacelle, in degrees Celsius (°C)
- *GeneratorTemp* – temperature of the electric power generator, in degrees Celsius (°C)
- *GeneratorRPM* – rotational speed of the electric power generator, in revolutions per minute (rpm)
- *PowerFactor* – the ratio of the electric power generator's real power (W) and apparent power (VA)
- *HydraulicPressure* – hydraulic pressure of the control system of the turbine, in bars
- *Availability* – the percentage of time the turbine is able to generate full capacity

Data Preprocessing and Summary Statistics

As the SCADA data was being imported, the DateTime column was also parsed to date time datatype and was set as the index of the data frame. Rows with missing data were dropped. Upon initial analysis of the summary statistics, it was found that some Power values were negative. These could have been consumption readings when the turbine is down. All negative Power values were replaced with zero. The updated summary statistics is as shown in *Figure 1*.



	Power	WindSpeed	WindDirection	NacellePosition	PitchAngle	AmbientTemp	GeneratorTemp	GeneratorRPM	PowerFactor	HydraulicPressure	Availability
count	128672.000000	128672.000000	128672.000000	128672.000000	128672.000000	128672.000000	128672.000000	128672.000000	128672.000000	128672.000000	128672.000000
mean	377.064543	5.259404	176.957049	112.363636	9.024808	27.575979	75.380122	1248.296510	0.734875	180.856645	93.237466
std	473.505093	2.886921	170.381534	107.994124	21.883551	5.330403	18.998890	622.324105	0.337626	39.573580	24.755157
min	0.000000	0.000000	0.000000	0.000000	-0.974000	0.000000	0.000000	-32.460800	0.000000	0.000000	0.000000
25%	0.000000	2.920700	3.061900	27.450800	-0.040000	27.372700	70.823525	1221.894200	0.532200	189.407775	100.000000
50%	185.245600	5.219700	131.569750	50.468800	0.601300	28.483250	77.642850	1279.008150	0.939500	189.795500	100.000000
75%	569.039050	7.102000	356.449125	222.198450	2.733000	29.576825	82.613400	1702.636125	0.994800	190.150525	100.000000
max	2000.234300	19.665300	359.999800	359.991500	90.000000	33.944800	138.351400	2019.600500	0.999100	193.650800	100.000000

Figure 1: Dataset Summary Statistics

Exploratory Data Analysis

Univariate and multivariate analysis were done on Power, WindSpeed, and Availability. High count of zero Power and WindSpeed values were found, which is normal due to several turbine stoppages every day, and the *low wind season* in the location of the turbine where wind speeds are generally low for 6 months. These could also be missing data. Availability was also observed, but there seem to be no pattern, with 100% Availability values across the entire span of Power values, and zero power values across the entire span of Availability values. Availability does not seem to significantly relate to the Power value.

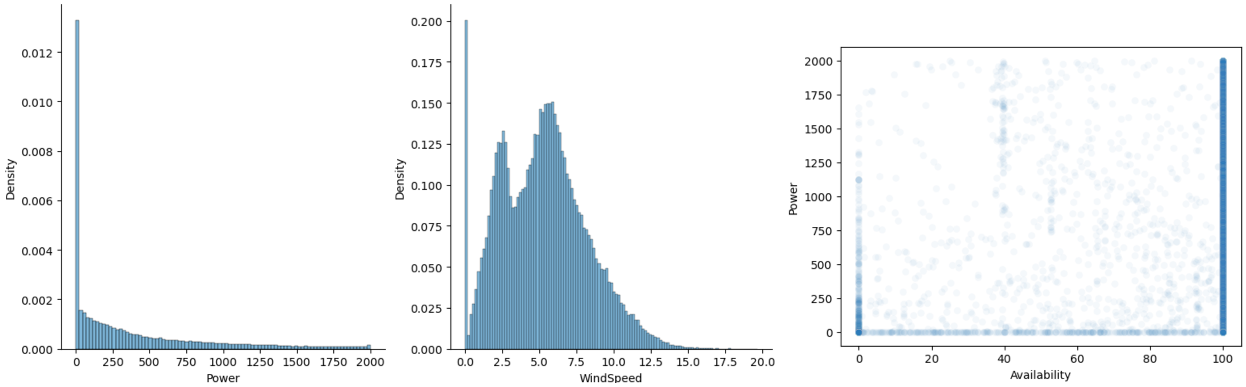


Figure 2: Exploratory Data Analysis

With these findings, rows with zero values of Power and WindSpeed as well as the entire Availability column were dropped altogether. This might not seem to make technical sense because zero power output due to zero wind speed and/or unavailability are normal occurrences in wind turbine operations. However, with the purpose of this project, removing these may exclude or reduce influences of no-wind and unavailability to the model, making it better for its intended purpose.

Outlier Detection and Removal

Focusing on evaluating the energy generation performance of the wind turbine excluding external factors that may affect generation such as grid curtailment, stoppages, and the lower power output during ramp up or down after or before stoppages, respectively, outliers must be identified and removed before modelling. However, too much filtering may result to overfitting, which is not good especially for other purposes such as forecasting using unseen data.

One method in handling outliers is removing data outside the range of *Q1 – 1.5*IQR* and Q3 + 1.5*IQR, where *Q1* and *Q3* are the *1st* and *3rd Quartiles*, respectively, and *IQR* is the *Interquartile Range* equal to *Q3 – Q1*. The factor *1.5* was calculated based on the *Central Limit Theorem* stating that in a normal distribution, data beyond 3 standard deviations above and below the mean are outliers.

The data was split into the *Before* and *After* improvement datasets based on dates given by the Owner. The *Before* improvement dataset covers 1 full year until the improvement activities started. Although there are more available data, using just the most recent year would capture the entire general recent state of the wind turbine performance before any improvement activities were done. The *After* improvement dataset starts after the activities were completed until the most recent available data. This was done to separately detect outliers and avoid misclassification of performance differences or data leakage.

Since Power is expected to range from zero to the turbine's maximum capacity, quartiles of Power alone cannot detect outliers. Quantiles with conditions based on another variable can be used. *Wind Power Curves* show the relationship between the power output and wind speed, commonly used in the industry by wind turbine manufacturers, service providers, and generation companies for performance evaluation and planning due to the simplicity of being able to get a reasonable estimate of the energy generation of a wind turbine using only 1 parameter. These are non-linear, following a *sigmoid function*.

There are several variants of the sigmoid function, such as the logistic function commonly used in classification, the hyperbolic tangent, etc. After plotting a variety of modified sigmoid functions over actual wind turbine scatter plots, using hyperbolic tangent (tanh) with the wind speed expressed in a 4th degree polynomial was chosen to fit best based on visual approximation. The function is described in *Equation 1*.

$$po\widehat{w}er = maxpower \times tanh((a \times windspeed + b)^4) \tag{1}$$

After deciding on a function, *Quantile Regression* was used to fit conditional quantile lines that would serve as ranges for Power values to be kept. Quantile regression fits the wind power curve with the minimum total absolute errors with weights based on quantiles, unlike linear regression which is just based on the ordinary least square error. With this, extreme values have less impact on the fitted line. This will also be used later as one of the regression models. To fit a line, an error function shown in *Equation 2* must be minimized. Equations 1 and 2 were written as functions and was fitted to an arbitrary list of quantile values by minimizing Equation 2 using the *minimize()* function from *scipy.optimize*.

$$\epsilon = \begin{cases} quantile \times (power - po\widehat{w}er), & (power - po\widehat{w}er) \geq 0 \\ (quantile - 1) \times (power - po\widehat{w}er), & (power - po\widehat{w}er) < 0 \end{cases} \tag{2}$$

The 25th and 75th quantile regression lines substituted Q1 and Q3, and the upper and lower bounds of the range for outlier detection was calculated. The fitted quantile regression lines and the resulting bounds were visualized on top of the Power vs WindSpeed scatter plots in *Figure 3*. Datapoints outside the upper and lower bounds were assumed outliers and were filtered out. An arbitrary value of ±1.5% for tolerance was also applied to the bounds to not lose any datapoints that had minimal deviation especially towards the right side of the curve where the power plateaus to the turbine's maximum capacity.

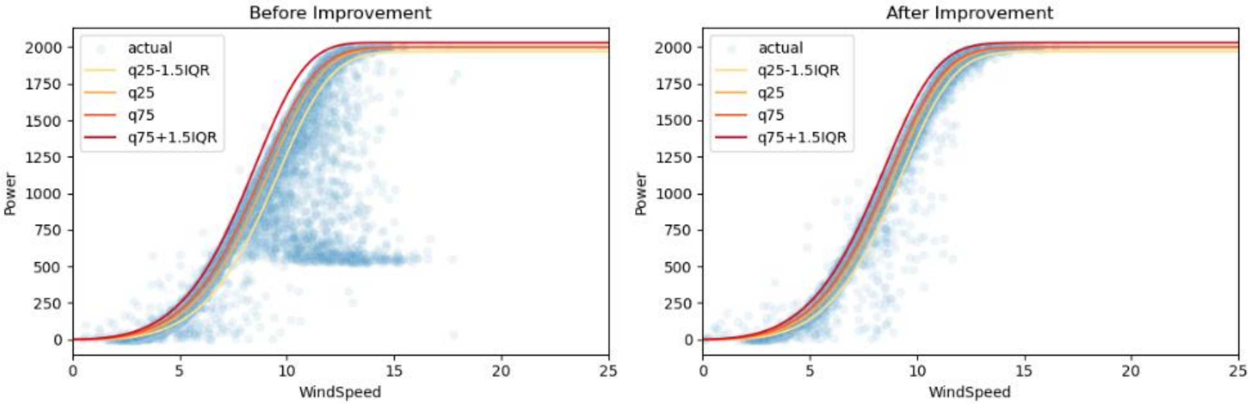


Figure 3: Power vs WindSpeed Scatter Plots and Quantile Regression Lines

MODEL IMPLEMENTATION

After filtering out the outliers, both *Before* and *After* improvement datasets were split into target and features data frames and were further split into 80% training and 20% test sets for modelling.

Regression With Only Wind Speed as the Feature

Following the concept of wind power curves, with wind speed theoretically being the most significant parameter driving the power output of a wind turbine, the first set of regression models were done with only WindSpeed as the sole feature. There was no need for feature selection and scaling.

First, regression based on the hyperbolic tangent sigmoid function defined in Equation 1 were fitted using the 50th quantile (median) and ordinary least squares, followed by regressors including *Decision Tree, Random Forest, GradientBoost, AdaBoost, XGBoost, CatBoost, LightGBM, SGD,* and *Artificial Neural Network*. All these models work well on non-linear regression such as a wind power curve. With this, linear regression was not included in the models. All models were separately fitted with the training sets of the *Before* and *After* improvement datasets. Below are some of the resulting power curves in *Figure 4*.

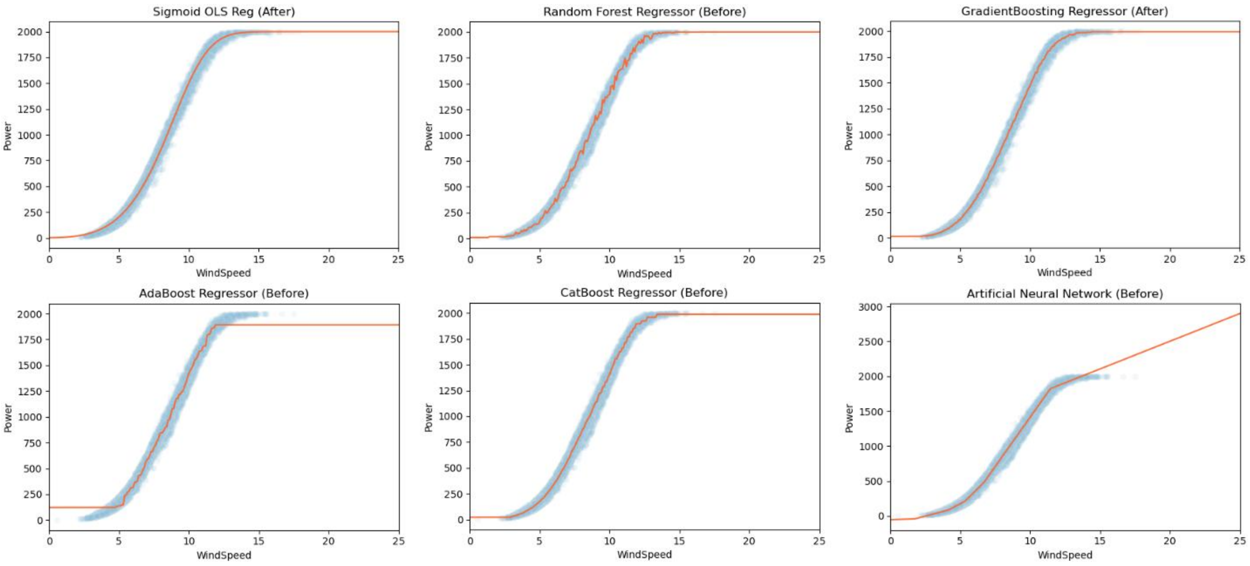


Figure 4: Sample Wind Power Curve Results

Regression With Multiple Features

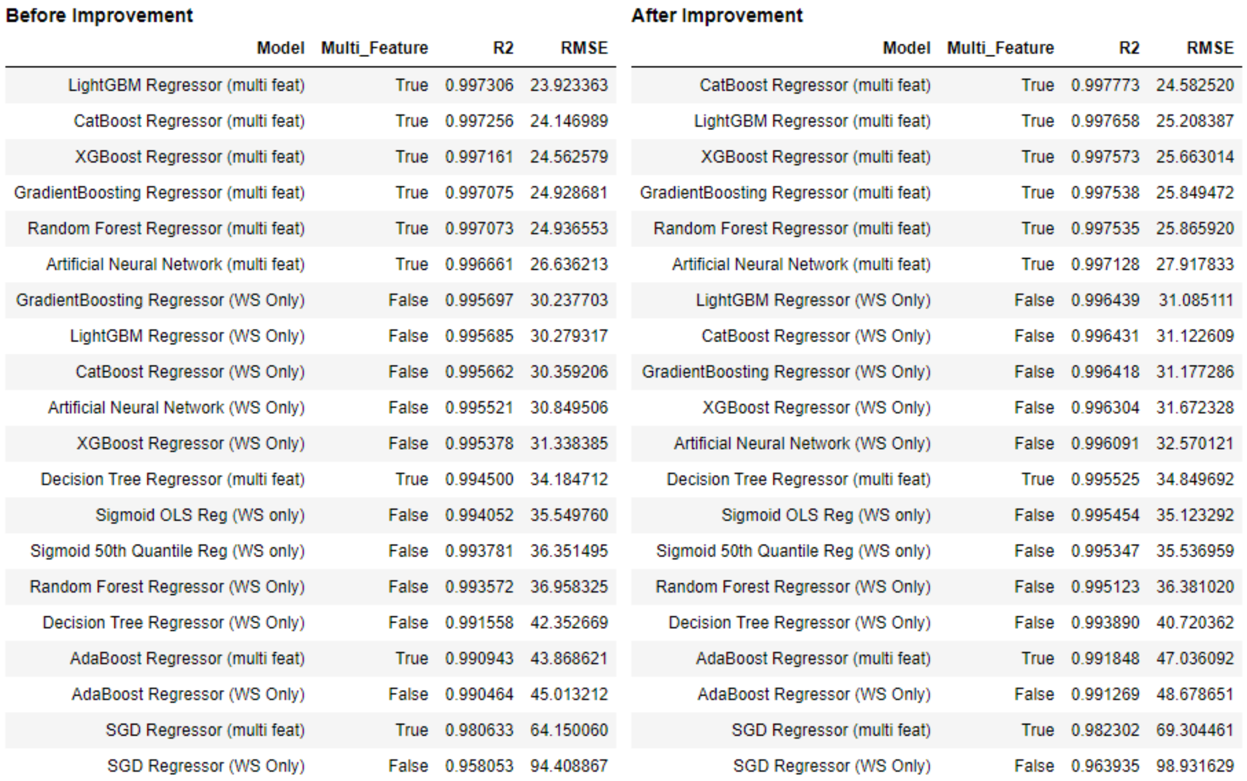
There is still the possibility that other features may influence the power. First, the features were scaled using *Standard Scaler*. Then, *SelectFromModel* with a *Random Forest Regressor* estimator was implemented for feature selection. However, the feature selection results included only WindSpeed, which supports the concept of wind power curve suggesting that wind speed alone can reasonably predict power. To still try multiple features, 4 features were forcibly selected using SelectKBest instead.

These could have been done in a *Pipeline* including modelling. However, running the code becomes slow. Instead of repeating feature scaling and selection for every model, these were done before looping through the models for faster processing. All the models in the previous section except for the sigmoid function (because these were designed for only 1 feature) were refitted.

RESULTS INTERPRETATION AND IMPLICATIONS

Regression Results

All models, both using single (WindSpeed only) and multiple features, for both the *Before* and *After* improvement datasets were evaluated using their corresponding test sets. The ranked accuracy scores are tabulated in *Figure 5*.



Before Improvement

Model	Multi_Feature	R2	RMSE
LightGBM Regressor (multi feat)	True	0.997306	23.923363
CatBoost Regressor (multi feat)	True	0.997256	24.146989
XGBoost Regressor (multi feat)	True	0.997161	24.562579
GradientBoosting Regressor (multi feat)	True	0.997075	24.928681
Random Forest Regressor (multi feat)	True	0.997073	24.936553
Artificial Neural Network (multi feat)	True	0.996661	26.636213
GradientBoosting Regressor (WS Only)	False	0.995697	30.237703
LightGBM Regressor (WS Only)	False	0.995685	30.279317
CatBoost Regressor (WS Only)	False	0.995662	30.359206
Artificial Neural Network (WS Only)	False	0.995521	30.849506
XGBoost Regressor (WS Only)	False	0.995378	31.338385
Decision Tree Regressor (multi feat)	True	0.994500	34.184712
Sigmoid OLS Reg (WS only)	False	0.994052	35.549760
Sigmoid 50th Quantile Reg (WS only)	False	0.993781	36.351495
Random Forest Regressor (WS Only)	False	0.993572	36.958325
Decision Tree Regressor (WS Only)	False	0.991558	42.352669
AdaBoost Regressor (multi feat)	True	0.990943	43.868621
AdaBoost Regressor (WS Only)	False	0.990464	45.013212
SGD Regressor (multi feat)	True	0.980633	64.150060
SGD Regressor (WS Only)	False	0.958053	94.408867

After Improvement

Model	Multi_Feature	R2	RMSE
CatBoost Regressor (multi feat)	True	0.997773	24.582520
LightGBM Regressor (multi feat)	True	0.997658	25.208387
XGBoost Regressor (multi feat)	True	0.997573	25.663014
GradientBoosting Regressor (multi feat)	True	0.997538	25.849472
Random Forest Regressor (multi feat)	True	0.997535	25.865920
Artificial Neural Network (multi feat)	True	0.997128	27.917833
LightGBM Regressor (WS Only)	False	0.996439	31.085111
CatBoost Regressor (WS Only)	False	0.996431	31.122609
GradientBoosting Regressor (WS Only)	False	0.996418	31.177286
XGBoost Regressor (WS Only)	False	0.996304	31.672328
Artificial Neural Network (WS Only)	False	0.996091	32.570121
Decision Tree Regressor (multi feat)	True	0.995525	34.849692
Sigmoid OLS Reg (WS only)	False	0.995454	35.123292
Sigmoid 50th Quantile Reg (WS only)	False	0.995347	35.536959
Random Forest Regressor (WS Only)	False	0.995123	36.381020
Decision Tree Regressor (WS Only)	False	0.993890	40.720362
AdaBoost Regressor (multi feat)	True	0.991848	47.036092
AdaBoost Regressor (WS Only)	False	0.991269	48.678651
SGD Regressor (multi feat)	True	0.982302	69.304461
SGD Regressor (WS Only)	False	0.963935	98.931629

Figure 5: Regression Accuracy Scores

Observations and Interpretation

All models have high accuracy scores, with high *R-Squared (R^2)* and low *Root Mean Squared Error (RMSE)* values. This may be partly due to the removal of outliers, which was deliberately done with the aim of to isolate only the energy generation performance of the turbine. It can also be observed that there are no significant differences in accuracy, except for the few bottom models, suggesting that there is minimal benefit in using multi-feature models over single feature in this case.

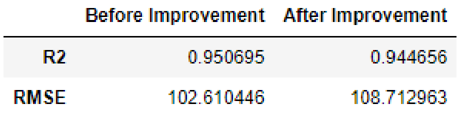
Based on these observations, as well as to simplify the model and enable the usability of wind power curves, the best model will be chosen from the single-feature models (with only WindSpeed). This will also make the quantification of the turbine performance improvement simpler and more straightforward by requiring only wind speed. With the low significance of other features to power compared to wind speed as well as the minimal accuracy gains versus the added complexity of having more features, the multi-feature models will not be selected.

Aside from accuracy, another consideration for choosing the best models was having minimal difference between the RMSE values of the *Before* and *After* improvement models to minimize error in the estimated improvement value in the end due to difference in accuracy. For this dataset, the best models for the *Before* and *After* improvement datasets using only WindSpeed as the feature were the *LightGBM Regressor* and the *CatBoost Regressor* models, respectively. The models have RMSE values of around 30 and 31, respectively, which are relatively low and of negligible difference compared to the highest possible actual value of 2,000. With an almost equal RMSE, they are acceptable for comparison of predictions.

<u>OUT-OF-SAMPLE PREDICTIONS</u>

The unused imported data older than the *Before* improvement period was used for *Out-of-Sample (OoS)* predictions. Unfortunately, having only a few months' worth of available data in the *After* improvement state, there is no available *OoS* data for that state to maximize the training data used in modelling. An *OoS* test can be done with new data once the *After* improvement state has elapsed close to a year. Since the only *OoS* data available is just prior to the *Before* improvement dataset, it is expected that the *After* improvement model will have lower accuracy, accounting for potential improvement over the *Before* improvement performance.

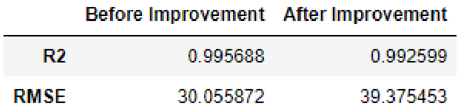
The *OoS* data has observable outliers, simulating how new unseen data will be like. First, the best models were tested on the unfiltered out-of-sample data to simulate the model will perform on new unseen data.



	Before Improvement	After Improvement
R2	0.950695	0.944656
RMSE	102.610446	108.712963

Figure 6. Out-of-Sample Accuracy Using Unfiltered Data

Although these accuracy scores are not low, it is still worse than the model accuracy scores on the test data due to the outliers. It is normal to test the model on the unfiltered out-of-sample data to simulate new unseen data. However, the purpose of this project is not to accurately forecast the net generation, but to evaluate the generation performance of the turbine isolating all other factors that may affect the generation such as stoppages and grid curtailment. Therefore, an *OoS* prediction will be done with cleaned *OoS* data. Since the *OoS* period is just before the *Before* improvement period, it can be assumed that the outlier removal method used for the *Before* improvement dataset is also applicable to the *OoS* data.



	Before Improvement	After Improvement
R2	0.995688	0.992599
RMSE	30.055872	39.375453

Figure 7. Out-of-Sample Results Using Cleaned Data

The accuracy of the best model on Out-of-Sample data was still high, with the expected relatively lower accuracy on the *After* improvement model.

EVALUATING THE PERFORMANCE IMPROVEMENT OF THE WIND TURBINE

Wind Distribution Profile

To quantify the turbine performance improvement in terms of *Annual Energy Production (AEP)*, a representation of an annual wind speed profile is needed to predict the energy generation using the models. The wind speed distribution of the most recent *full years* before outlier removal was plotted. Using partial-year data will skew the distribution. For the data used, the wind speed distribution is observed to resemble a *Weibull Distribution.*

A synthetic annual wind speed data can be used to approximate a realistic annual wind speed profile. Using these with the models will quantify the improvement in terms of AEP. To generate the synthetic wind speed data, a Weibull distribution curve was fitted to the wind speed distribution using the *Weibull_min()* function from *scipy.stats*. Random numbers with size worth 1 year were then generated based on this Weibull distribution to represent a synthetic annual wind speed distribution. The actual and the synthetic average wind speeds for a year were *5.3529* and *5.3443*, respectively, which are close. Another method that could have been done is time series forecasting, if only more historical data is available to consider trends and annual variations.

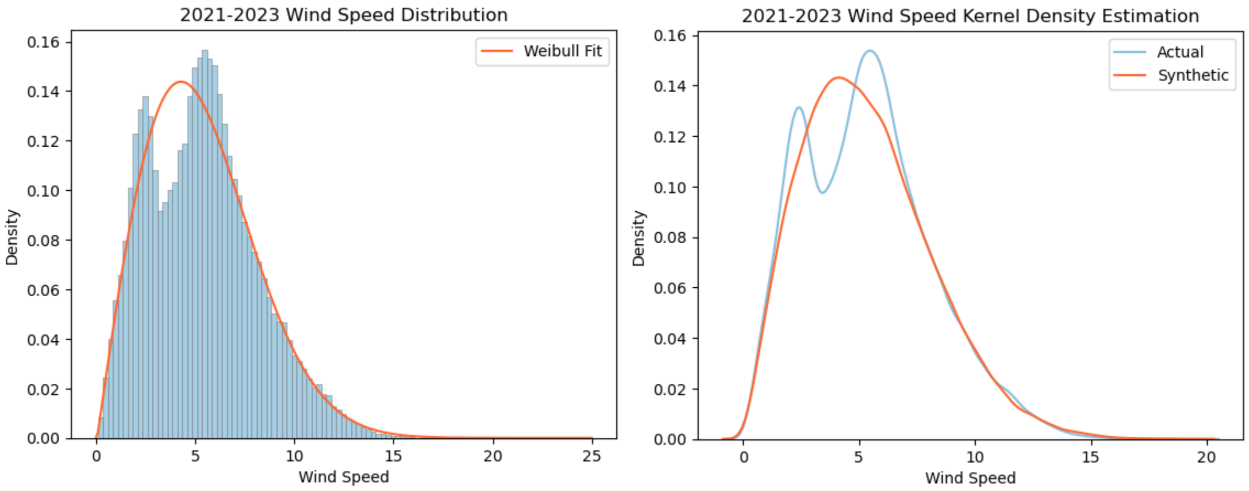


Figure 8. Wind Speed Distribution Vs Weibull Distribution

Using the synthetic wind speed data on both the best *Before* and *After* improvement models, the representative AEP of the turbine for each state were predicted. Although the difference of these predictions may already approximate the percentage improvement, the AEP needs further adjustments to have a closer approximate of the energy (kWh) improvement. A simplified was this was done was getting the average *Availability* for the same recent full years of data and randomly replacing a number of rows equivalent to the unavailability percentage with zeros. This simplified approach however only includes stoppages and does not consider grid curtailment, ramping up/down after/before stoppages, or any other irregular scenarios. For the turbine used in this run, the resulting predictions are summarized in *Figure 9*.

Annual Energy Production (kWh)		**Difference**	
Before	**After**	**kWh**	**Percentage**
3,256,971.57	3,377,171.28	120,199.71	3.69%

Figure 9. Wind Turbine AEP Improvement

Note that this result only applies to one particular wind turbine with less than a full year of data in the *After* improvement state. Rerunning the code using 4 other turbines with data from a similar period yielded different figures, summarized in *Figure 10.* There were different best models with close RMSE values. These alternate runs were not detailed in this report anymore as all steps were the same. The additional turbine data were also not included in the project file to maintain a reasonable file size.

Turbine	**Best Regression Model**		**RMSE**		**Annual Energy Production (kWh)**		**Difference**	
No.	**Before**	**After**	**Before**	**After**	**Before**	**After**	**kWh**	**Percentage**
1	GradientBoost	LightGBM	30.24	31.09	3,256,971.57	3,377,171.28	120,199.71	3.69%
2	GradientBoost	CatBoost	48.34	43.75	3,100,344.14	3,223,366.84	123,022.70	3.97%
3	CatBoost	CatBoost	34.10	30.71	3,198,325.48	3,260,040.60	61,715.13	1.93%
12	GradientBoost	LightGBM	49.65	50.45	2,830,334.54	2,937,317.17	106,982.63	3.78%
16	GradientBoost	CatBoost	46.84	39.76	3,132,718.86	3,151,761.48	19,042.62	0.61%

Figure 10. Alternate Wind Turbine Run Results

CONCLUDING REMARKS

The energy generation performance improvement of a wind turbine after upgrading its blades was evaluated by splitting the data into *Before* and *After* improvement datasets, minimizing effects of other factors aside from wind such as stoppages and grid curtailment to the turbine output by statistically detecting outliers using quantile regression of a sigmoid function, then modelled using different regressions with single and multi-feature models. The best model was then selected based on accuracy, but also considering simplicity after testing different models and doing Out-of-Sample predictions. Finally, to quantify the improvement in terms of energy, a synthetic wind profile was generated following a Weibull distribution fitted to the actual wind speed distribution using recent full years of available historical data.

Various model configurations that were tested in this project proved that wind speed is the most significant feature for predicting the energy generation of a wind turbine. Models using only wind speed as the feature, although simpler and more straightforward, are not inferior to models with more features.

Only testing 2 wind turbines both with limited *After* improvement state data does not yet give conclusive evidence whether the Owner should also do the upgrades on the rest of their fleet of not. But this project was coded in such a way that it is very easy to rerun using different turbines and/or periods of data, assuming the format of the raw SCADA data is the same. It would be best to obtain more data and do more test on more wind turbines before reaching a final decision.

Video Presentation Link: https://www.youtube.com/watch?v=YYVaXZYbsXs

References

Donnelly, R. A., Jr. (2020). Business Statistics: A First Course (3rd ed.). Pearson.

Cole, S. (n.d.). Wind Turbine Power Curve. Retrieved from https://theroundup.org/wind-turbine-power-curve/

Wood, T. (n.d.). Sigmoid Function. DeepAI: Machine Learning Glossary and Terms. Retrieved from https://deepai.org/machine-learning-glossary-and-terms/sigmoid-function

Koenker, R., & Hallock, K. F. (2001). Quantile Regression. Journal of Economic Perspectives, 15(4), 143-156.

Brownlee, J. (2021, February 8). Function Optimization with SciPy. Machine Learning Mastery. Retrieved from https://machinelearningmastery.com/function-optimization-with-scipy/

Onnen, H. (2021, October 22). Probability Distributions with Python's Scipy. Towards Data Science. Retrieved from https://towardsdatascience.com/probability-distributions-with-pythons-scipy-3da89bf60565

Python Documentation:

https://numpy.org/doc/
https://pandas.pydata.org/docs/
https://seaborn.pydata.org/
https://scikit-learn.org/stable/
https://catboost.ai/en/docs/
https://docs.scipy.org/doc/scipy/