K Nearest Neighbors Regression on the Wind Turbines Dataset

Andrea Boskovic

12/15/2021

Contents

1 Background

The main goal of this project is to use K-Nearest Neighbors (KNN) Regression to predict turbine capacity using a modified version of the features in the United States Wind Turbine dataset. Our dataset contains information about the location of the wind turbine, the year it became operational, and various statistics about the turbine's operation, such as its height and rotor diameter. We explain how we reach the optimal model that minimizes our objective loss: Mean Square Error (MSE).

2 Preprocessing

To create a best fit model, we first have to clean our dataset. The first step in this process involves dealing with missing values.

We remove the feature representing the retrofit year, a representation of when the turbine was partially retrofit, because over 90% of observations in this column are missing. The retrofit feature, which is an indicator variable showing whether or not the turbine has been partially retrofit, contains information that can be a useful substitute for retrofit year.

When we check for other missing values, we see that one row in our training set at index 10682 contains missing values in three columns, namely in that of the rotor diameter, rotor swept area, and turbine total height from ground to tip. We remove this observation from the dataset. This observation's removal should not alter the dataset significantly because we are only removing one row from a dataest with 50,000 observations.

The next step in our data preprocessing involves verifying that each of our features are of the correct type. State and county are both character types, so we transform these into factor variables so that the model treats them as categorical. Similarly, since the retrofit indicator variable is numeric, we also transform this into a factor. Note that we must perform these transformations of our data, including removing the retrofit year feature, on the test set as well.

Our last preprocessing step involves training our KNN Regression model. When we train our model, we normalize the numeric features by centering and scaling them.

3 Predictors

After testing combinations of features in our KNN Regression model, we find that the predictor that achieves the lowest MSE on the training set is

$$\text{t\_cap} \sim \text{t\_rsa} + \text{t\_hh} + \text{p\_year} + \text{retrofit} + \text{t\_ttlh} + \text{xlong} + \text{ylat},$$

where each of the variables are defined in Table 1.

Intuitively, it makes sense that most of these features would affect a turbine's capacity. If a rotor sweeps more area and is taller, it should generate more energy. Similarly, newer and retrofitted turbines are likely to have a higher capacity because they were more recently built or updated. A turbine's latitude and longitude also might affect its capacity because some areas likely have windier conditions than others, allowing a turbine to generate more energy.

Note that the value of K selected for the KNN Regression on this model is 2, which we found through cross validation, and details on selecting optimal K are given in Section 4 and Section 5.

Table 1: The meanings of variable names in the selected model.

Full Name	Variable Name	Variable Type
Capacity	t_cap	Numeric
Rotor Swept Area	t_rsa	Numeric
Hub Height	t_hh	Numeric
Year Operational	p_year	Numeric
Retrofit	retrofit	Categorical
Total Height	t_ttlh	Numeric
Longitude	xlong	Numeric
Latitude	ylat	Numeric

3.1 Exploratory Data Analysis

To better understand the features in the model, we visualize each of their effects on our target variable, turbine capacity, in Figure 1. We show the turbine capacity against hub height, rotor swept area, and total turbine height, and in each of these visualizations, we fit a linear model to the data, shown in green, to better understand their relationships.

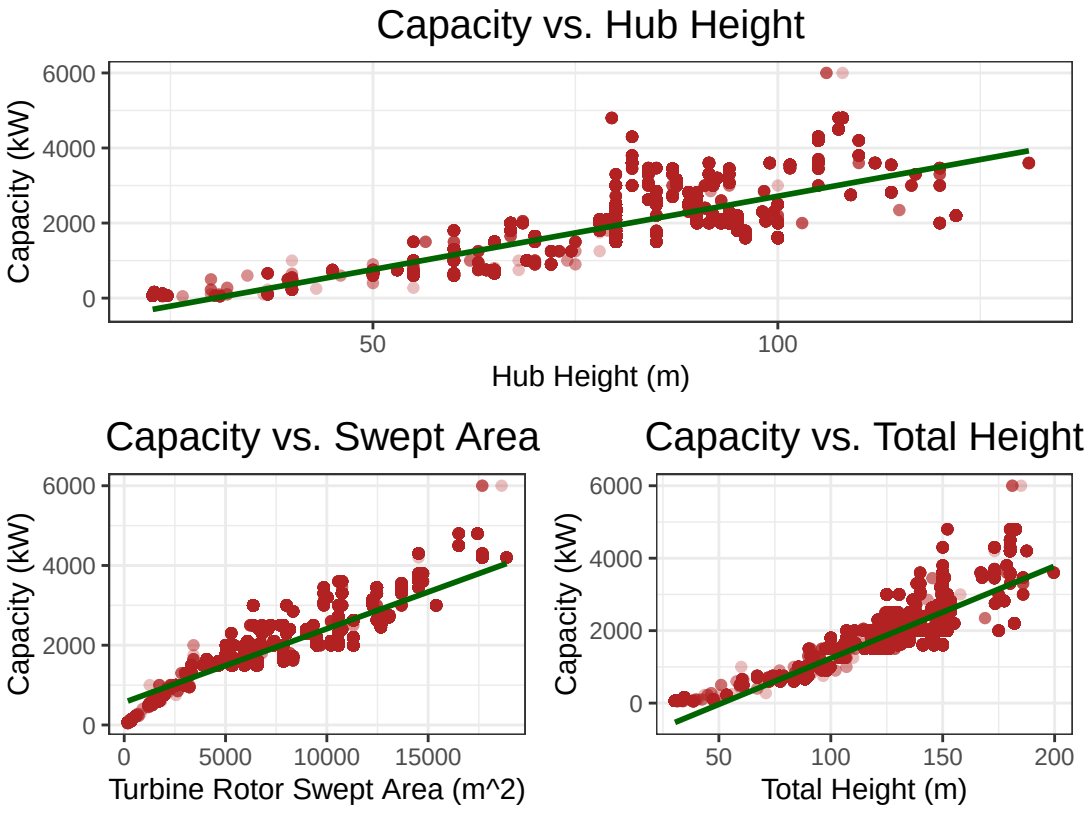

Figure 1: An exploratory data visualization of turbine capacity against some predictors in our KNN Regression model.

In each of these plots, we see that a linear model seems to fit the data decently well overall except in the tails, particularly on the lower end.

4 Training

In training KNN Regression to predict turbine capacity using our set of predictors, as stated in Section 3, we choose $K = 2$ for the number of neighbors. To find this optimal value of K, we performed K-Fold Cross Validation.

Using cross validation, we examine values of K, namely integers, between one and twenty. We use ten folds in this cross validation because this value seems appropriate given that the training dataset contains 50,000 observations. In other words, for each value of K, where K represents the possible number of neighbors, we train a KNN Regression predictor on $50,0000 - \frac{50,000}{10} = 45,000$ observations and test it on the remaining 5,000 observations. Our ten-fold cross validation finds that $K = 2$ successfully minimizes the MSE for the model we choose compared to other values of $K \in \{1, 2, \ldots, 20\}$.

The model we choose, specified in Section 3, is chosen by trial and error. Namely, we test several combinations of predictors and choose the model that minimizes the MSE on the training set.

5 Experimental Results

Here, we discuss how we determine the optimal choice of K for our KNN Regression model, and we show the importance of each of the features in our model.

5.1 Visualizing the Optimal Choice of K

In Figure 2, we see the plot referenced in Section 4 that compares the MSE, our model evaluation metric, against number of neighbors K in the cross validated KNN Regression model. We transform the number of neighbors K with a log in order to better visualize the value of K at which the model reaches a minimum MSE. Now that we have our optimal KNN Regression model and our optimal choice of K for that model, we can use our model with $K = 2$ nearest neighbors to train our predictor.

5.2 Feature Importance

We also may be interested in understanding the importance of each of the features in our model to assess its strengths and weaknesses. To do so, we create a feature importance plot that shows the MSE of each feature after permutations. If shuffling the observations in a feature causes a large degradation in model performance, we know that feature must be important. In other words, if the MSE of a feature after permutations is higher, the feature is more important.

In Figure 3, we see that the rotor swept area is by far the most important feature, followed by the total height of the turbine. The year in which the turbine became operational, the turbine's hub height, the longitude, and the latitude of the turbine all have similar importance, but each is far less important than turbine height and rotor swept area. The retrofit binary indicator feature has the lowest aggregate importance. Note that the dashed line in the plot represents the MSE for the full KNN Regression model.

This plot is quite informative in our understanding of the model. It validates the context for the model, particularly the fact that the amount of area that a turbine's rotor sweeps affects its capacity. Likewise, it makes sense that the turbine's height would affect its capacity and allow it to produce more energy.

5.3 Performance on the Test Set

In Figure 4, we see that our model performs well on the test set against some of the numeric features, specifically turbine hub height, total turbine height, and the rotor swept area. The red points, which represent the predicted turbine capacity plotted against each respective feature in the test set seem to match the training data well.

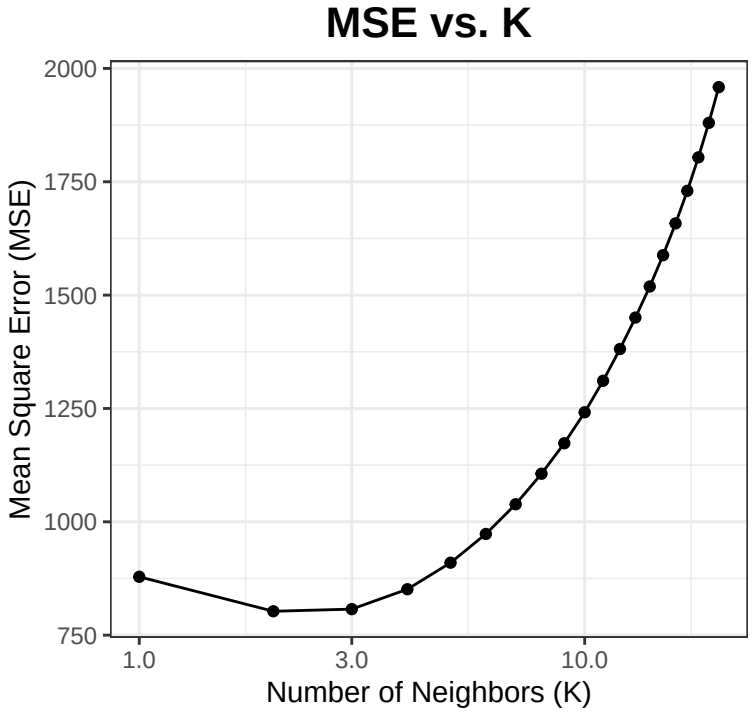

Figure 2: Mean Square Error over Choice of K.

6 Prediction Error Estimation

Finally, to predict the estimated error on the test set, we use the k-fold cross validation from training. Specifically, for each K number of neighbors we test, our k-fold cross validation outputs an MSE. To predict the MSE of the model on the test set, we can then average the MSE's from each K, given by

$$\widehat{L}_{LS} = \frac{\sum_{i=1}^{k} MSE_i}{\max\{i\}},\ i \in \{1, \ldots, 20\}.$$

Although we use $K = 2$ in our KNN model, this seems like a viable estimate because it is unclear whether another value of K will perform better on the test set. Still, this will likely be a pessimistic estimate of our MSE because the value $K = 2$ neighbors is approximately 802, and MSE increases significantly as K increases. We show several choices of K and their corresponding MSE's from cross validation in Table 2.

Clearly, at high values of K, namely $K > 5$, the MSE increases significantly, which is also evident in Figure 2. Using our estimate, we have that

$$\widehat{L}_{LS} = 1303.$$

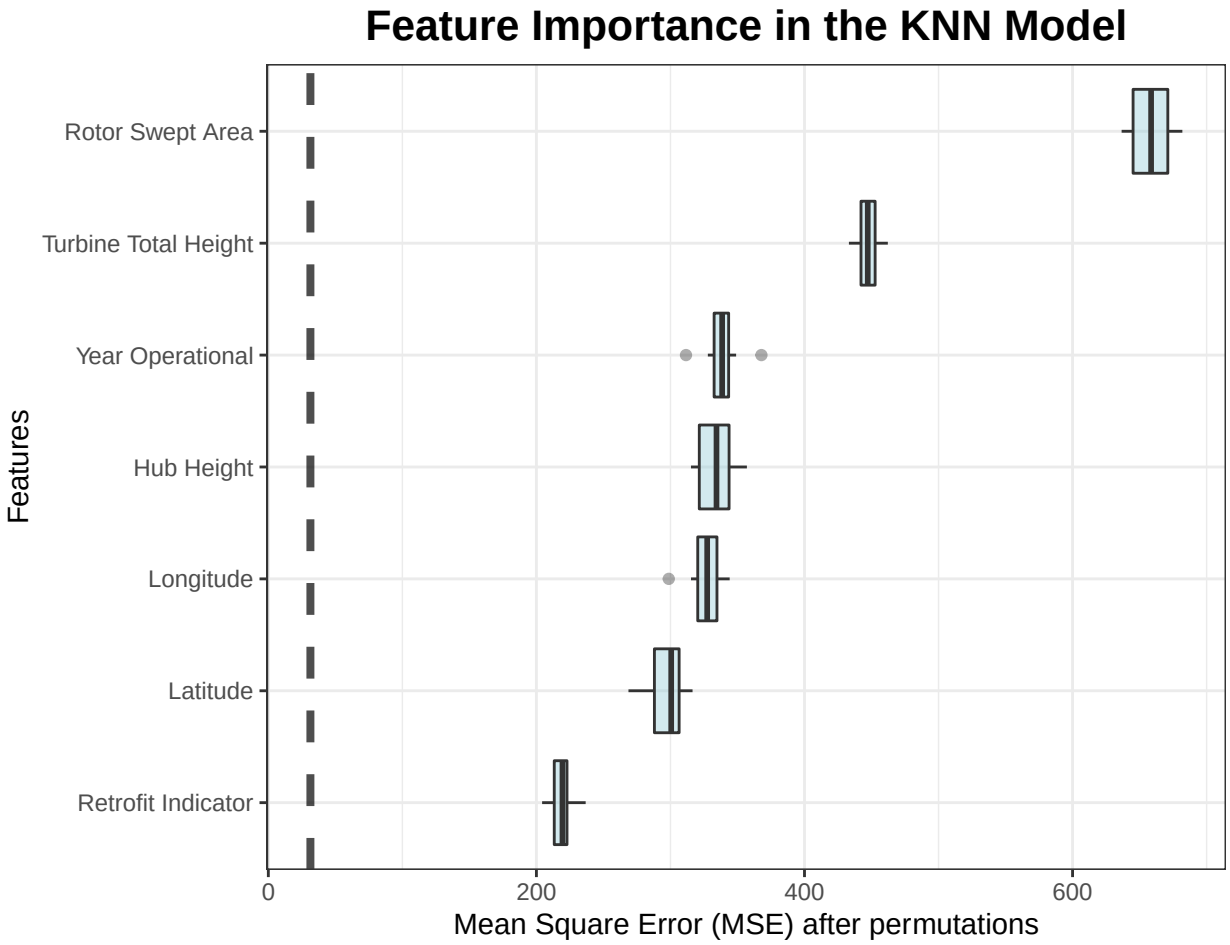

Figure 3: A visualization of feature importance in our KNN Regression model.

Table 2: The values of K against the cross-validated MSE for the KNN Regression model.

K	MSE
1	878.74
2	802.72
3	807.40
5	909.86
10	1241.53
12	1381.13
15	1588.12
18	1803.86
20	1958.82

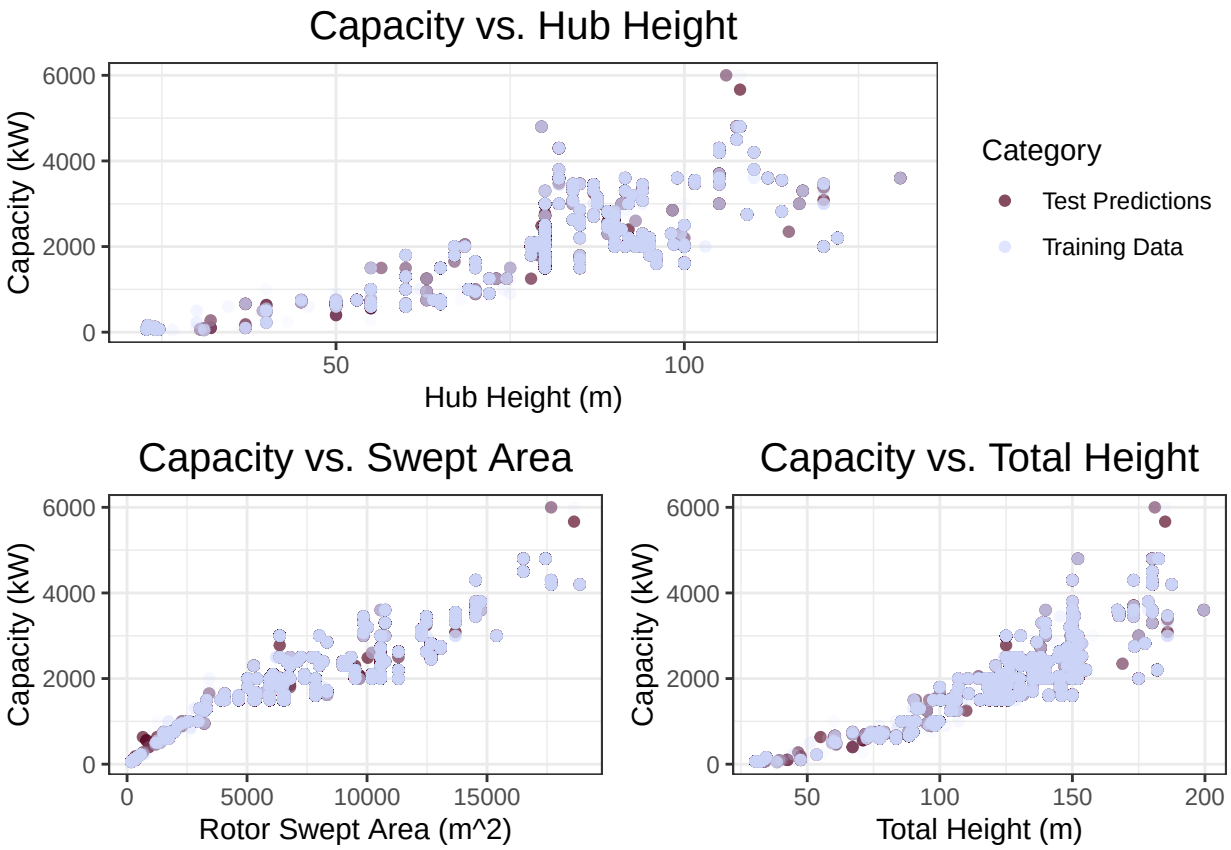

Figure 4: Visualization of the predictions of our model compared to the training set.

7 Conclusion

In this report, we outline the procedure we use to choose and train a KNN Regression model. After trying different models, we find a set of predictors that minimizes MSE, and we then determine that setting the number of neighbors in our regression model to two further minimizes the loss. We then show how we use that model to predict turbine capacity based on the test set.

Without the true test set predictions, we cannot truly evaluate our model's performance, but based on the our predicted loss and our plots of the model's predictions on the test set compared to the training data, we should be fairly confident in our model's ability to predict turbine capacity well.