Coming into the datathon, I had no intention of trying to win or even complete. This was my first datathon, so I just thought it would be cool to see if I could just make something that worked before the time ran out. I thought using a random forest would be the right model choice since the problem was just asking for prediction and there was nothing special about the data itself (such as image data, where there is a spatial relation between data points).

I took out some unimportant variables (like “id”) and categorical variables that were too large in number of unique values to one-hot-encode, then ran the random forest. Then, I looked back at my data, trying to see if I could incorporate the other categorical variables that I originally omitted into my model.

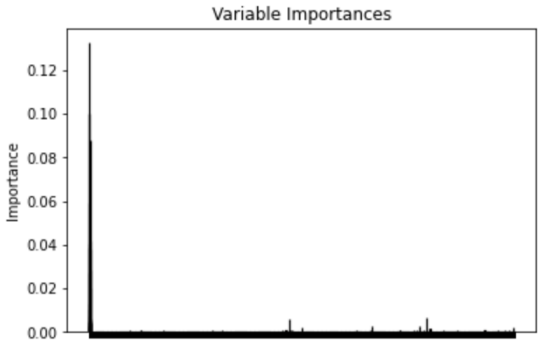
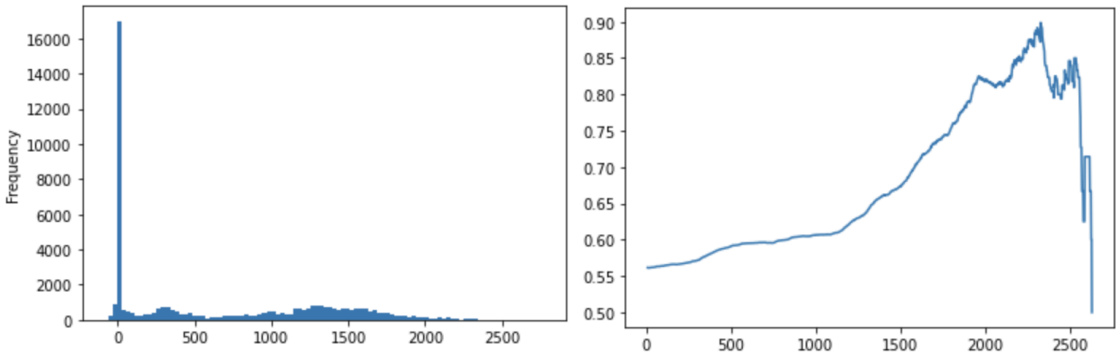


Figure 1: This is a variable importance graph showing the results after initially selecting the most important variables from a previous random forest, then including the one-hot-encoded versions of the “funder,” “installer,” and “scheme_name” (the operator of the waterpoint), which I thought could have some importance in predicting waterpoint quality. The bars on the left were the most important variables previously selected, and the smaller peaks are the other categorical variables that were relatively important. Thus, I could engineer the data to check if a sample had these particular funders, installers, or operators, solving the issue of using these variates that were previously unusable.

Also, I incorporated some feature engineering, like changing the latitude and longitude to x-y-z coordinates and having an indicator variable for certain numerical values being above a certain number.



On the left is a histogram of gps_height. The figure on the right shows the fraction of samples given a gps_height value. I picked a threshold for a boolean variable visually on when this ratio stagnated (in this example, at around 2000).

Lastly, I did some model tuning by increasing the variance of the trees in the random forest and determining the number of samples for splitting a node. Since random forests are

generally robust towards having redundant variables and don't really overfit, I did not attempt to simplify the model by removing variables based on variable importance.

I think my summary makes it seem like I didn't do a lot, but I think a lot of the time was actually spent on figuring out how to put everything together, researching potential pitfalls to implementing random forests, and time spent on ideas not working out whether due to results (like regularizing the model) or inability to implement on time (like oversampling and undersampling). At the same time, there were a lot of things I probably should have done, like using a validation set, having more rigorous validation methods for hyperparameters, and accommodating unbalanced data by oversampling or undersampling for variates. Regardless, I am glad that I was able to compete in the datathon and enjoyed the experience!