Predicting Power Plant CO2 Emissions Executive Summary

Owen Jetton

Using annual energy and temperature data collected from 2016 to 2019, I use four statistical and machine learning techniques to predict how many tons of carbon dioxide a power plant will emit in a given year. I clean up annual government energy data, ensuring I have essential variables such as electricity generated, fuel type used, the sector the energy is consumed by, and annual heating days. The models, linear regression, elasticnet regression, decision tree, and a boosted decision tree ensemble, are tuned using k-fold cross-validation and sequences of hyperparameters. The boosted model ended up performing the best against the cross-validated and testing data.

The question I deal with, is how much carbon dioxide will a power plant emit in a year? The question of how much CO2 is emitted from a power plant is important because of its relation to climate change and carbon dioxide's effect on humans and the environment. This is a prediction problem because it concerns estimating CO2 emissions from a power plant, as opposed to the relation between CO2 emissions and other variables.

I use data from 2016 to 2019, since that is the latest data available, and the data before 2016 was formatted differently. The data regarding CO2 emissions and most other variables I use to solve this problem, came from the U.S. Environmental Information Agency. Data regarding the number of heating days came from the National Centers for Environmental Information. Data cleaning involved binding the datasets from different years, merging the emission data with the temperature data, and assigning multiple qualitative numerical variables to be factors. One challenge with the data is the presence of useless or repetitive variables, such as names with corresponding numerical codes. Additionally, many of the variable names are impairingly long; though this rarely was a problem. One major shortcoming is the fact that the heating days data is generalized by state, and missing for two states. This is a shortcoming because there is a lot of variation in energy demand between different parts of states, and this variation is not depicted in the models used.

I use four methods to solve this problem: linear regression, elasticnet regression, decision tree, and a boosted decision tree. I trained the models on a 5-fold cross-validated training set with over 16 thousand observations, and tuned with regard to the root mean square error (RMSE) for all models, to maintain a consistent benchmark with which to evaluate and compare the models. The penalty and mixture hyperparameters in the elasticnet model were tuned, with an ideal model surfacing with a penalty of 0.01, and a mixture of 0.3, slightly favoring a ridge to a lasso regression. For the decision tree, the cost complexity and tree depth hyperparameters were tuned, and the best tree had a cost complexity of 0 and a tree depth of 10. The boosted model uses 100 trees, and the tuned hyperparameters of tree depth and learning rate yielded the best result with a depth equal to 11 and a learning rate of 0.1.

Much like with tuning, I remain consistent with measuring the success of the models' performances on the cross-validation set and testing data by using the root mean square error. In predicting performance success by using the RMSE from the cross-validation sets, the boosted model performed the best, followed, in order, by the decision tree, linear regression, and elasticnet models. On the test data, the boosted model performed the best, followed by the elasticnet model, then linear regression, with the decision tree coming in last. I believe the decision tree's fall from second to last was induced by overfitting to the training data, which limited its performance on the testing data. During this process, I learned that my computer does not like to compute boosted or decision tree models. I have also learned that a boosted model is the best model for predicting the amount of carbon dioxide emitted from a power plant.