Enefit Energy Data Challenge Report

Keenan Wallace

**Overview**

The aim of this project was to create a machine learning model to predict the energy production and consumption levels of prosumer clients of Eesti Energia in Estonia based on a variety of factors. This was a challenge sent out through the website Kaggle by Eesti Energia with the hope that being better able to predict energy production and consumption trends of clients will help reduce costs from energy imbalances.

Eesti Energia is an Estonia based energy company which originally focused on oil shale electricity production, but has shifted its focus towards renewable energy with the promise of stopping oil shale energy production by 2030 and being carbon neutral by 2045. One of the issues that renewable energy sources like solar and wind face is that they produce energy intermittently and must be supported by energy storage systems or another type of energy production like nuclear in order to consistently meet the energy demand. This system can lead to additional costs when production doesn’t match consumption due to the necessity of additional production when intermittent energy sources are insufficient, costs to expand the transmission grid, and losses from overproduction. By being able to better predict the consumption and production of clients, Eesti Energia can determine the most efficient layout of resources and cut down on costs. To this end, they posted a challenge to Kaggle, allowing participants access to data on weather, gas and electricity prices, and characteristics of each client, and offering prizes for the most accurate machine learning model prediction of energy consumption and production.

**Project Process**

I started this project with some research into renewable energy and Enefit Green/Eesti Energia, the host company, in order to understand the goals of the challenge and help interpret the data provided. Following this, I investigated further into the data with some data visualization, descriptive statistics, and reviewing of the datasets and included variables. The next step involved understanding the structure and assignment of the challenge and determine how the provided datasets fit in to the larger picture in order to properly merge them such that models could be trained and predictions made. This merging was done in two separate ways, and then focused in on the merge version that yielded better results with the model. Finally I fit both linear regression and random forest machine learning models, making some adjustments to improve the quality of the model, determined by the mean average error between the predicted and test values.

**Data Exploration**

For the data exploration, I started with some boxplots visualizing what was already in the train dataset, and then gradually moved on to merging and subsetting different datasets to experiment and see what trends I could find. The processes of data exploration and visualization and experimental merging were heavily intertwined, though I’ve cut out much of the less useful/successful attempt in the final Data\_Exploration file.

Some interesting trends discovered include interaction between is\_business and other variables like temperature, eic\_count, and installed\_capacity in explaining consumption, but not production. Time series plots gave a good overview of energy trends throughout the year, making it evident that seasons play a strong role in the level of energy production/consumption, yet temperature isn’t as strong of an indicator as I would have originally expected.

**Data Merging**

Weather

Two different methods were tried for merging the weather data with the rest of the data. Version 1 treated the train and test sets separately. The train set used historical weather data, merging the historical weather directly by date to the train set such that each row in the train set included the actual weather that occurred during that period. The test set, on the other hand, used forecast weather data which matched each row with weather data that had been predicted the day before for that region and time. I theorized that by training the model on actual weather data and then testing it with predicted, there would be less error that could result from faulty forecast data that if both the test and train sets used likely erroneous forecast data. However, consistency in data collection methods between the test and train sets is also important, so I also created a version 2 of the merged data set, which treated test and train the same and only separated them into separate sets once all the data sets were merged. This version used exclusively forecast data, without involving the historical weather.

Regardless of version, all weather data merging involved using a separate “mapping” set to match coordinates to regions of Estonia. Then all weather data was consolidated (via average) across regions such that each region only had one entry of each variable per hour. This allowed the weather data to be merged with the train data set using date, time, region, and id block for forecast weather. Not all variables from the weather dataset were used, only temperature, direct solar radiation, precipitation, and dewpoint, which were deemed more likely to be meaningful.

Client

Clients were matched via characteristic data found in both the clients and train datasets (is business, county, product type), allowing further characteristics like EIC count and installed capacity to be merged into the train dataset. The data did not include multiple clients sharing all the same characteristics, which allowed such a merge.

Gas Prices

Gas prices were simply merged by data block id, as there was only one for each day.

Electricity Prices

So that only the electricity prices available when the predictions would be made(at 11am) were used, the electricity price data was first reduced to only what was available at 10am and then merged by data block id such that each price would be applied to all of the rows corresponding to the following day in the train set.

**Machine Learning Model Fitting**

For fitting a machine learning model to predict energy production and consumption given the available data, I used linear regression and random forests in RStudio and random forests in python (through Kaggle kernels). The linear regression and random forests were fit to both versions of the merged train data. There were some variations tried, such as in the treatment of the is\_consumption variable as either a predictor or a factor to separate and recombine the sets on, as well as transformations applied to the target data in the training model. The most successful of these adjustments being the log transformation, which somewhat helped the inequal variance observed in the residual plot of the regression object and brought down the mean absolute error somewhat. The random forest method proved to be a much more accurate model than linear regression and using the same process to fit a random forest model with both versions 1 and 2, version 2 was found to be more effective.

Because fitting the full dataset through a random forest model exceeded the memory in RStudio and failed to run, I exported the more effective version 2 test and train sets to Kaggle kernels to fit through random forest in python. Using the full dataset in this manner, the MAE (mean absolute error) was reduced, and then even further reduced by applying the log transformation in python, eventually reaching an MAE of 116.91, which is more than 300 lower than my first attempt in RStudio with linear regression.

**What I learned**

Through this project I gained experience in data manipulation and interpretation, further improved my skills and proficiency in RStudio, and figured out how to work with and graph time series data, which I had never worked with before. I am now much more comfortable with data merging and subsetting, and can better manipulate files.

In addition, I took my first look at machine learning model and learned the ideas behind some methods as well as how to apply them practically. While I have only at this point skimmed the surface of machine learning, my new understanding of training and testing sets, decision trees and random forests, and the link between linear regression and machine learning models will definitely help me as I continue to more advanced topics.

Apart from these main topics, by searching other’s participants’ code for hints in the Kaggle Competitions discussion session, I now have more experience in interpreting data related python code which I was previously thoroughly unfamiliar with and through my background research I’ve gained a deeper understanding of the process and issues in switching to renewable energy.

**Areas for improvement**

While I was able to create a machine learning model that wasn’t *too* bad, it is still far from the best it can be. During this ISP I didn’t dig too deep into methods for adjusting and fine tuning models, which could have increased the accuracy of my model. When I was fitting the random forest model, I simply kept the default settings without adjusting the parameters, which could have helped the model. For instance, I could increase the number of trees or nodes, or specify different weights or requirements for the splits, which could be helpful if I were to figure out the optimal way to do this. I could also using boosting methods, such as first fitting weaker models like decision trees , testing them, and then adjusting the weights of each depending on their performance before including them in an ensemble forest model. By adjusting the weight of each tree by performance, the forest can become more accurate. However, I suspect that the more intricately the model is tinkered with, the higher the chance of overfitting, which should be watched out for. This could involve an extra test set to check, which I also didn’t use in this project.

Furthermore, I wasn’t able to figure out the properly layout that Enefit wanted the model to be in, and therefore didn’t submit my model. In the future I hope to improve my python skills such that I will be able to better interpret such things.