Lab 3 Report

We looked on the forum for inspiration when looking for the best method to perform our regression. One that particularly stood out was a post by user MeiChengShih that detailed his stacking of 15 different models to optimize his result, and he showed that it significantly increased his score. Looking forward to our final project and Kaggle competition, we will definitely consider such comprehensive stacking in our methodology. He also detailed his use of outlier detection to improve his model. He did this by finding a set of outliers to remove from his training set. He then retrained his model using this new set and got a significantly better result. Had the scope of this lab been greater, we may have implemented something like this. Another example of this can be found in a post by user Andy Harless in which he exemplifies the importance of taking simple, logarithmic averages in order to improve your score. Finally, we were surprised in general at the wealth of knowledge contained in the posts. Many people openly shared their thinking in regards to the competition, and in reading through them we were given a more realistic look at possible ways that we could take our future tasks.

Our first attempts at a model was using just lasso and just XGBoost. Both these methods on their own proved to give us reasonable results but we knew we could improve our results significantly if we combined these methods in some way. Our first attempt at combining was to implement a form of stacking (implementation is still in the notebook) but the score ended up being worse. We believe this is the result of an an unnecessary normalization of a column in our data but we could not pin down the exact cause. For this reason we believe that this method may still be able to prove our results if we put significant more energy into refining it. However, what we ultimately decided to do was take a linear combination of the results of both lasso and XGBoost. More specifically, our final results took the form of A*lasso + B*xgboost, where $A + B = 1$ and $A > 0$ and $B > 0$. This method was found in a report by Eric Bruin on Kaggle and can be found here: https://www.kaggle.com/erikbruin/house-prices-lasso-xgboost-and-a-detailed-eda. We tried a variety of values for A and B. The somewhat intuitive

try of 0.5 and 0.5 was not the best, but rather A = 0.7 and B = 0.3 provided the best results. If you look at the graph of Lasso vs XGBoost provided in our notebook, you can see that at most points Lasso performs better, but it has a couple distinct values that are out of the ordinary. Weighting Lasso at 0.7 helps to mitigate these discrepancies and ended up giving us the best results.