EP2420 - Project 1: Task II - 2.2, Task III

André Silva

November 7, 2020

1 Task II - 2.2

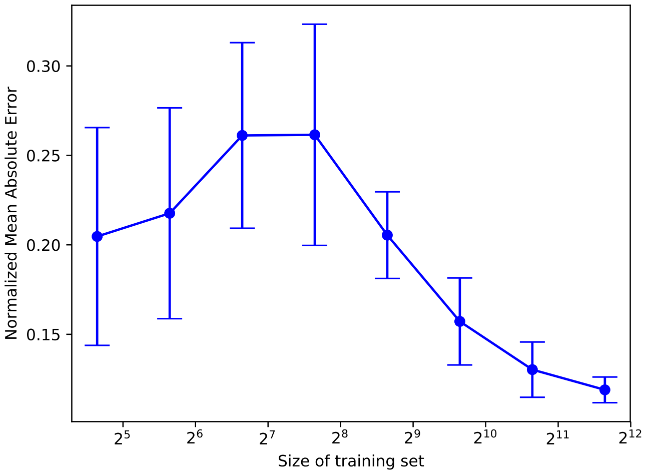

Figure 1: Measured *Normalized Mean Absolute Error* for a given training set size using *Lasso Regression* [2]. Error bars represent 2σ over 50 measurements. Data points represent means.

Intuition tells us that the more information we have access to, the better conclusion we can extract from them.

By looking at (1), we notice that the curve begins by unexpectedly increasing, followed by an expected decrease.

We can also notice that the variance of the measurements starts off very high, and shows a decreasing tendency as we increase the size of the training set. This partly explains why the curve doesn't follow our initial expectations.

Another reason that can explain this is the fact that outliers in small training sets have a higher influence than in large training sets, and the probability of including outliers in randomly picked training sets decreases as we decrease its size.

Overall, we can conclude that the larger a training set is the better a model can predict the system, as the training set becomes more general and so more information can be inferred from it.

2 Task III

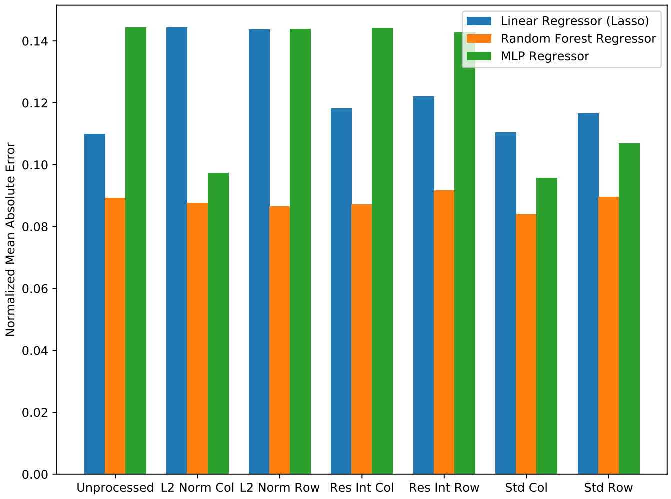

Figure 2: Measured *Normalized Mean Absolute Error* for a given pre-processed dataset.

In (2), we can see that pre-processing the data impacts some models more than others.

In the case of *Random Forest Regression* [1], the gains or losses of accuracy are marginal for all pre-processing methods. This happens by definition, since tree partition algorithms are not affected by the scaling or centralization of datapoints.

MLP Regressor [3] seems to be the model which suffers the biggest impact from data pre-processing, showing either no difference or a considerable gain.

Lasso [2] consistently shows worse performance on pre-processed data when compared to the original one.

By analyzing (2) again, we can see that standardizing along each column is the method that results in the best results for all 3 regression techniques.

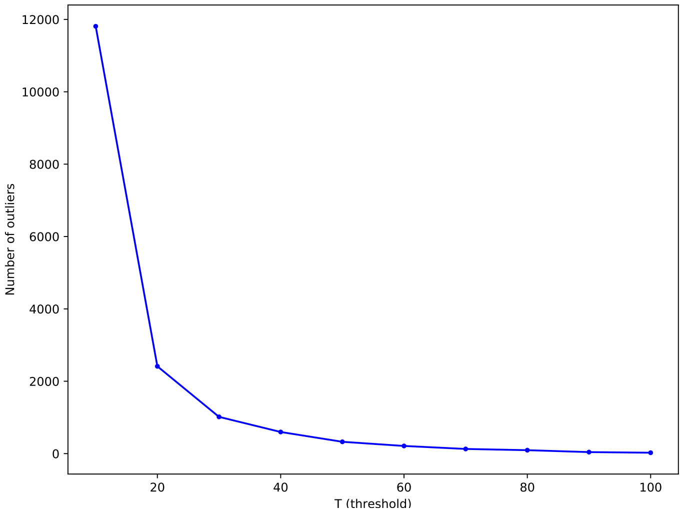

Figure 3: Number of outlier points for a given threshold on the standardized design matrix along each column.

(3) shows, as expected, a function that decreases while T increases, eventually converging to 0. This happens because we have standardized the data along each column, making each feature space look like a Gaussian distribution with 0 mean and unit variance.

As we filter rows based on absolute values, we will find less and less examples of outliers as T increases, by the definition of a Gaussian distribution.

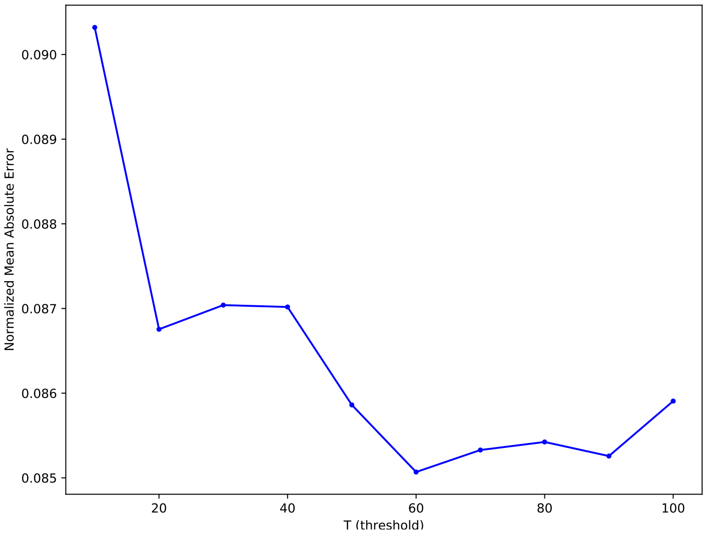

Figure 4: *Normalized Mean Absolute Error* for a given threshold T (outlier removal) using *Random Forest Regression* [1].

(4) shows us that the *NMAE* starts off by decreasing, as we increase the threshold for outlier removal, suggesting low threshold values may lead to the removal of important parts of the dataset. As we incorporate more datapoints in our dataset, we can better predict outcomes. This goes in hand with the results of section (1).

As we reach a very high treshold we notice a slight increase, which might represent the impact more extreme outliers have on the outcome of the tested model, *Random Forest Regression* [1]. This approximation to a parabola was the expected result for this experiment.

References

[1] *sklearn.ensemble.randomforestregressor*. `https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.randomforestregressor.html#sklearn.ensemble.randomforestregressor`. accessed: 2020-10-31.

[2] *sklearn.linear_model.Lasso*. `https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.Lasso.html`. accessed: 2020-11-07.

[3] *sklearn.neural_network.MLPRegressor*. `https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPRegressor.html#sklearn.neural_network.MLPRegressor`. accessed: 2020-10-31.