Final Report (Team 36)
Molecular Activity Challenge

Yifei Yan, Chen Liang, Tirth Patel, Xinxin Mo, William Xi, Liyuan Xie

Project Overview

It is important to identify molecules that are highly active toward their intended targets but not other targets that might cause side effects when developing the new medicines.

Traditionally, drug discovery is the area of research and development that used the most amount of time and money. The time frame can easily range from 3 to 20 years and costs can range between several billion to tens of billions of dollars for the research teams to find the molecules that highly active toward the target through myriad experiments. Therefore, Merck, the largest pharmaceutical company in the world, proposed a machine learning challenge to help develop safe and effective medicines by predicting molecular activity.

To address this challenge, this project aims to utilize machine learning as a cost-effective tool for predicting biological activities of different molecules, both on- and off-target, given numerical descriptors generated from their chemical structures.

There is a total of 15 datasets included in this challenge, each for a biologically relevant target. Each row of the data corresponds to a molecule and its molecular descriptors that derived from the chemical structure. The training files are of the form -- Column 1: Molecule ID; Column 2: Activity (Note that these are raw activity values and different datasets can have activity measured in different units); Column 3 to end: Molecular descriptors/features

Given, the time and scope of this project, we choose to work on the data set with largest number of molecules and largest number of numerical descriptors (ACT_1). The dataset has 37241 rows and 9493 columns.

Exploratory data analysis (EDA)

A. Missing values

There is no missing value in the data set.

B. Variance analysis/imbalance data

The histogram (Figure 1) shows the imbalanced distribution of the data set, where the number of observations belonging to 4.3 - 4.7 is significantly higher than those belonging to the other classes. There are 68% of values fall into the 4.3 - 4.7 bin. The imbalance is caused by the fact that all molecule activities that are lower than 4.3 are recorded as 4.3 during the original data collecting process.

C. Correlation Heatmap

7 features/fingerprints with the highest correlation to the molecule activity are picked and plotted into a heat map due to display purpose (Figure 2). From the correlation heatmap, deeper color on the legend means the closer the relationship of the two fingerprints. There are 4 deep red

color show 2 different pairs (D_2466&D_4849, D_518&D_517) in the heatmap which means the when one fingerprint exists, the other has a high probability to be present as well.

Dimensionality Reduction

The dataset has 9491 features which could require a vast amount of computation power and time to train the model. Meanwhile, overfitting would occur easily because of too many features. Therefore, dimension reduction has to be performed on the dataset.

I. Near zero variance

Biological datasets come sometimes with predictors that take a unique value across samples. This kind of predictor is not only non-informative, it can break some models you may want to fit to your data. Even more common is the presence of predictors that are almost constant across samples. Therefore, our solution is to remove all predictors that have variance close to zero. However, after using the VarianceThreshold function, only 314 (3.3%) features are removed, which still requires high computation power to complete the downstream processing. Therefore, we make a scree plot to determine the number of factors to retain in the analysis by altering the variance threshold.

The scree plot (Figure 3) has cumulative variance on y-axis and the number of principal components on x-axis. The cumulative variance is calculated by (the sum of variance for x principle components)/(the total variance of all features). From the plot, 450 features could represent 90% of total variance which we believe will be enough for our training data. Thus, we choose to carry those 450 features into the following study. Besides, PCA was also used to reduce the dimensionality. However, it was abandoned because it could only explain 75% variance with 450 features which is much less than near zero variance.

II. Correlation

A correlation matrix (Figure 4) was made on the dataset with 450 features. The dimensionality was further reduced to 50 features by picking the 30 most positively correlated features and 20 most negatively correlated features with respect to the molecular activity.

III. ANOVA

One-way ANOVA is used to determine whether there is any statistical significant difference between the means of two or more independent groups. In this project, ANOVA was calculated for each feature in the 450 dataset relative to the molecular activity. The 50 features with highest ANOVA score were carried on to the following study.

Machine learning models

A. Random Forest Model

A random forest baseline model is built using the RandomForestRegressor function in scikit learn. The model has 500 trees and the maximum features is equal to the square root of the input dimensions. The random forest model is run with 5-folds cross-validation. The number of trees and maximum features was obtained by consultation with colleagues.

After evaluating the performance of the baseline random forest model, we trained the model on 3 datasets created by different pre-processing methods as well as utilized the random search method to tune the parameters. In terms of random search, we first researched on the hyperparameters that are most important to random forest model [3], and we found that the number of trees (n_estimators) and the number of features for splitting each leaf node (max_features) are most important. Apart from these two hyperparameters, we also selected max_depth, min_samples_split, min_samples_leaf, and bootstrap type. Then we defined a domain grid, namely the sets of hyperparameter values for different hyperparameters. Random combinations of the value of the hyperparameters will be created and set to the model. We ran 100 rounds of random search and did cross-validation in each round. The random search results are shown in Figure 5. The best hyperparameters with the best scores were selected to test on the test set data. We also calculated the APE for the predicted activity value (Figure 6). However, we found that though the model trained by 450 features got the best scores under cross-validation, it demonstrated high variance and high bias, which means it would not perform well on the test set. According to the result on the test set (Table 1), the best model, which can reach the lowest mean absolute percentage error (MAPE) and mean absolute error (MAE) as well as highest R^2 score, was trained by the 50 features selected by the correlation. The corresponding MAPE is 5.58%. Best hyperparameters are as follow. The number of maximum features is equal to the log of the input dimensions. The number of trees is set to 360. Other parameters are max_depth = 27, min_samples_split = 5, min_samples_leaf = 2, bootstrap = False.

B. Neural Network Model

The neural network model is built in Keras. It consists of 4 layers: the input layer, two dense hidden layers and the output layer. The hidden layers have nodes of 50/25 for 450 features and 40/20 for 50 features. Rectified linear units are chosen as the activation function as an initial baseline due to work by Cathy Qian[1] and George Dahl [2]. Relu units seemed to perform better than sigmoid for QSAR datasets. The Adam optimizer is chosen for the baseline model as adaptive learning rate optimizers are generally suited for sparse data. The default settings for the Adam optimizer are used. The ANN architecture developed by Cathy Qian serves as the baseline model for this project. The test set MAPE for this baseline is 6.88% (Figure 7). The next step was to tune the neural network hyperparameters. The hyperparameters tuned are learning rate for the adam optimizer, number of epochs and batch size. A 100 iteration random search with 5 fold cross validation is employed to determine the optimal parameters. The random search results are displayed in Figure 8. The metrics used are R^2, mean absolute percentage error and mean absolute error. R^2 was selected as a metric as the competition used R^2 as the metric for deciding winners. From Figure 8 the MAPE and MAE random search points are similar due to the

definition of the two metrics. However, R^2 results contradicted MAPE and MAE results as the 450 feature dataset outperformed the other two datasets while performing similarly for MAPE/MAE. Thus, R^2 was not used as the metric to define optimal tuned parameters and MAPE was used instead. A hypothesis for artificially inflated R^2 may be due to the number of predictors and adjusted R^2 could be used instead. Figure 9 shows the performance of the tuned parameters for the 450 feature, 50 feature and 50 anova feature datasets. The 450 dataset clearly outperformed the 50 feature datasets during training. However, the performance results (Figure 10) indicate that there no significant improvement on the test set. This indicates that overfitting is an issue for application of ANNs on QSAR datasets like Merck. Other work done on QSAR data-sets used minimal preprocessing to preserve as much information as possible. The results here suggest that some dimensionality reduction could potentially lead to an improvement in results. From the model results in Figure 10, there seems to be marginal improvement over the baseline. The issue with the results is the spread of the predictions. The standard deviation of predictions for each data set is high at around 6 to 9%. Certain bins even contain predictions that are off by 30-40%. From a drug discovery standpoint, vastly inaccurate predictions will result in wasted research and development time and funds. Thus, a lower standard deviation is important for predictions. Euclidean K-means clustering is used to identify clusters that may be contributing to the variance observed in predictions for the 450 feature dataset. 10 clusters are chosen using the elbow method. The APE predictions for each cluster, the frequency in each cluster and the MAPE for each cluster is presented in Figure 11. The results identify several clusters (4 and 8) that perform much better than the average MAPE of 6% and some clusters (Cluster 1) performing significantly worse than average. Further work could be to understand and identify the clusters.

User Interface

Since this platform was making not only for research institutes and researcher who have scientific background but also for those who don't but are interested in learning about protein and drug binding, it would be best to make it as straightforward and easy to navigate as possible.(figure 12.) So on the main page, the user is prompted to enter the protein ID in the Merck database or select one from the drop down menu. Similarly, the user will also select the drug compound he or she is interested in. After hitting SUBMIT, the system will show the simulation result as shown in the second picture. For example, the target protein GPCRs and drug compound ACT4_M_5065 have a molecular activity of aboue 5.3. We also want to design a feature where the user can choose to store their own simulation results in a file so they can compare and study.

Bottlenecks / challenges

A. Data Sampling

Machine Learning algorithms tend to produce unsatisfactory classifiers when faced with imbalanced datasets. This happens because Machine Learning Algorithms are usually designed

to improve accuracy by reducing the error. Thus, they do not take into account the class distribution/proportion or balance of classes. Therefore, a new sampling has to be implemented to get a more balanced data.

B. Data Dimensionality

The dataset has extremely high dimensionality. Dimension reduction compromises the data integrity and increases potential error.

C. Data Sparsity

The sparsity of the data after dimension reduction leads to model development problems. Concerns with the model include overfitting and consequently a lack of generalization for the other training sets.

D. Difficult to improve performance of ANN past a bottleneck of 6-7% MAPE

Tuning sees marginal improvement. Overfitting is not an issue at lower dimensionality. Several methods have been tried such as, dropout, batch normalization, regularization; however, it has minimal impact on improve the performance of the ANN.

Next Steps

Due to the challenge that we are facing, maybe minimal or no preprocessing will help to improve the performance of the ANN. And it may require deeper nets and a dedicated GPU. Additionally, try ensemble models and more iterations of the neural network architecture. Since it is difficult to identify the data clusters, reduce overall variance in prediction accuracy may help to predict easier. Also, testing the model on the other Merck datasets to evaluate generalizability and linking the compound name to compound ID to determine biosimilars could be the next steps.

Github Link: https://github.com/xinxinmo/DataXMerckChallenge

References

[1]https://github.com/CathyQian/Data_Science_Projects/tree/master/Predicting_Merck_Molecular_Activity
[2]https://arxiv.org/pdf/1406.1231.pdf
[3]https://towardsdatascience.com/hyperparameter-tuning-the-random-forest-in-python-using-scikit-learn-28d2aa77dd74

Appendix

Figure 1. distribution of the data set

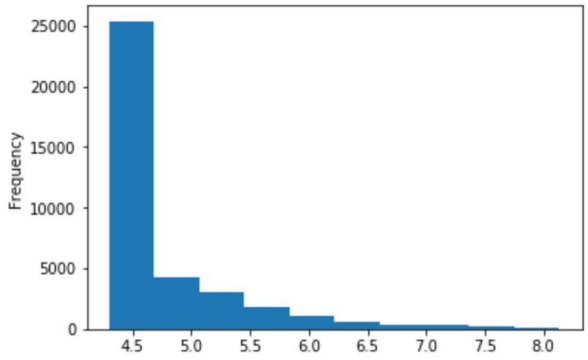


Figure 2. The correlation heatmap with 7 most correlated features

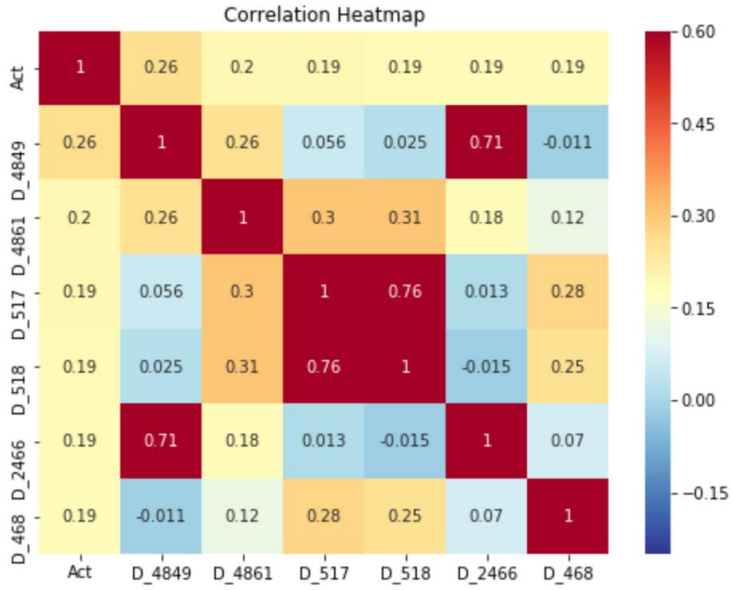


Figure 3. the variance using the Near Zero Variance

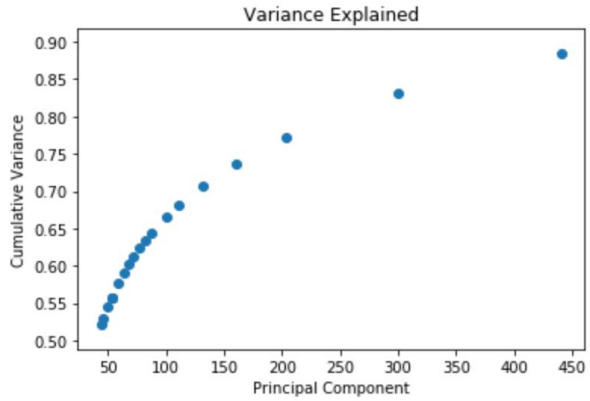


Figure 4. Distribution of correlation scores for 450 features

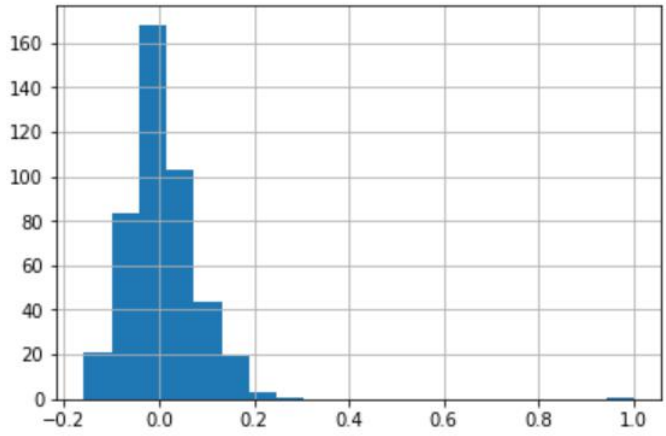


Figure 5. Results of random search: Evaluation metrics are based on R^2, MAPE, and MAE from left to right. Blue, orange, and green dots represent the models trained by the 450-feature dataset, 50-feature dataset selected by correlation, and the 50-feature dataset selected by ANOVA method. The models with the best score are marked with a star sign.

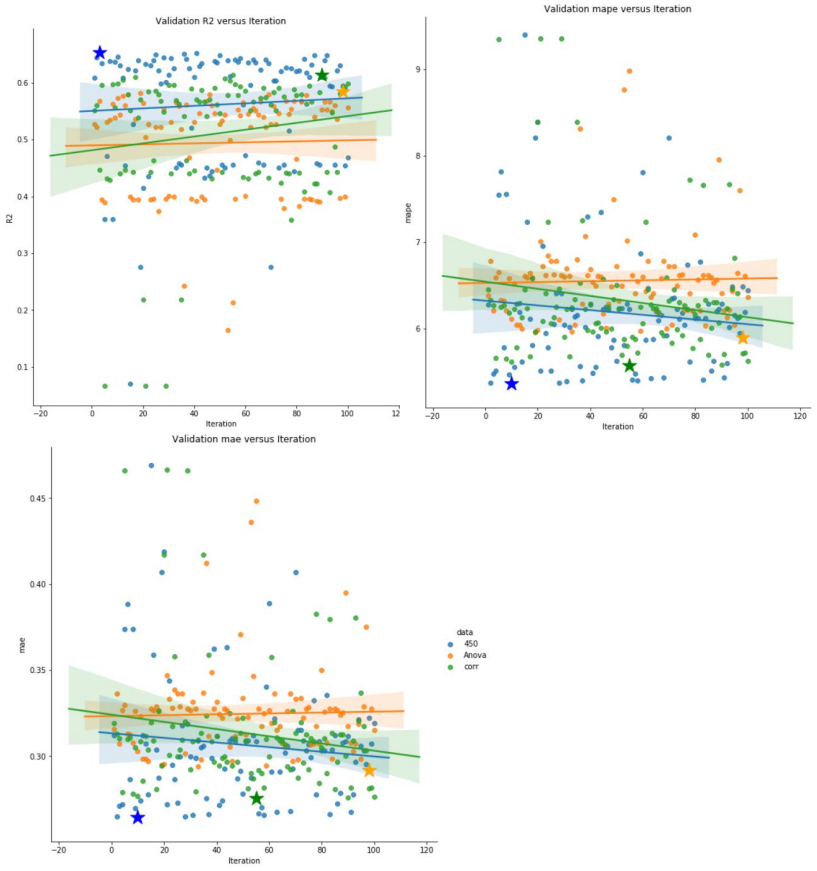


Figure 6. APE histograms of predicted activity value on the test set for best models trained by the 450-feature dataset, 50-feature dataset selected by correlation, and the 50-feature dataset selected by ANOVA method.

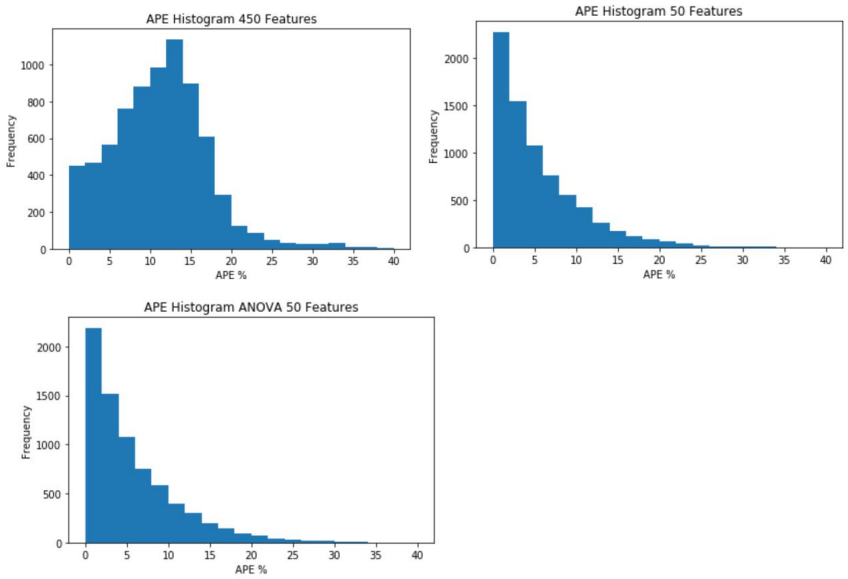
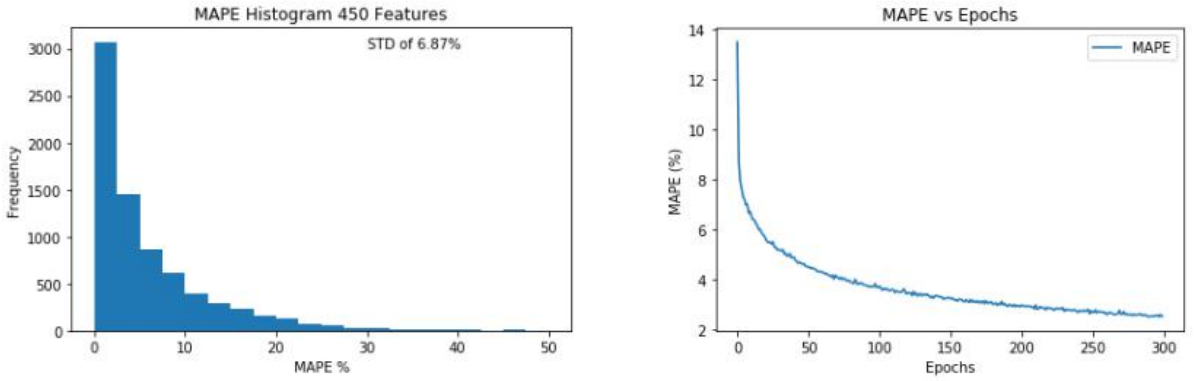


Table 1. Results of the best models on the test set.

Data	R^2	MAPE	MAE	n_estimators	min_samples_leaf	min_samples_split
450 Features	0.101	11.1%	0.530	500	1	2
Anova 50 Features	0.595	5.70%	0.280	500	1	2
50 Features	0.631	5.70%	0.270	360	2	5

Figure 7. APE of predicted MAPE for the baseline model using 450 features, prediction results, hyperparameters and training MAPE vs epochs.



Data	R^2	MAPE	MAE	Batch_size	No_epochs	Learning Rate
450 Features	0.313	6.88%	0.332	300	300	0.001

Figure 8. Random search with cross validation results for the artificial neural network.

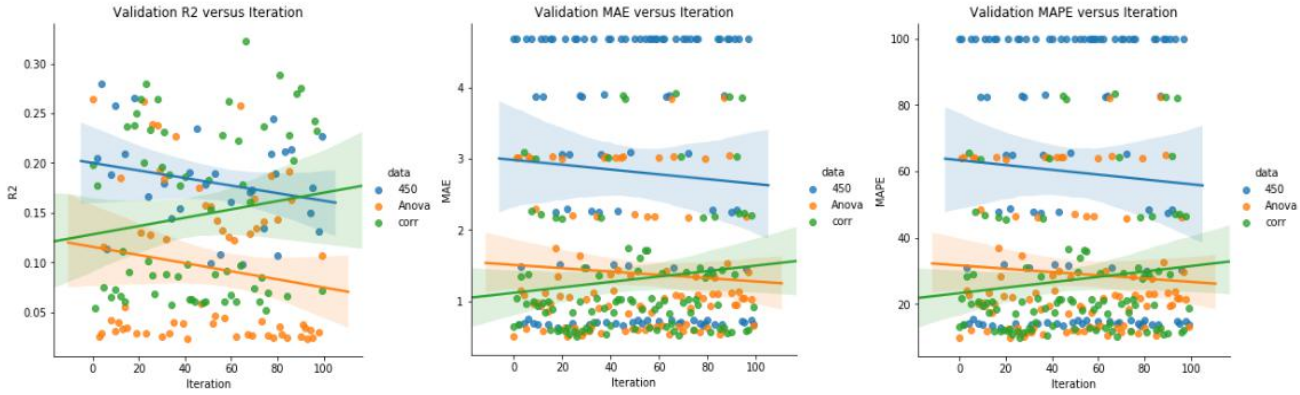


Figure 9. Loss and metrics results as a function of epochs for the three datasets.

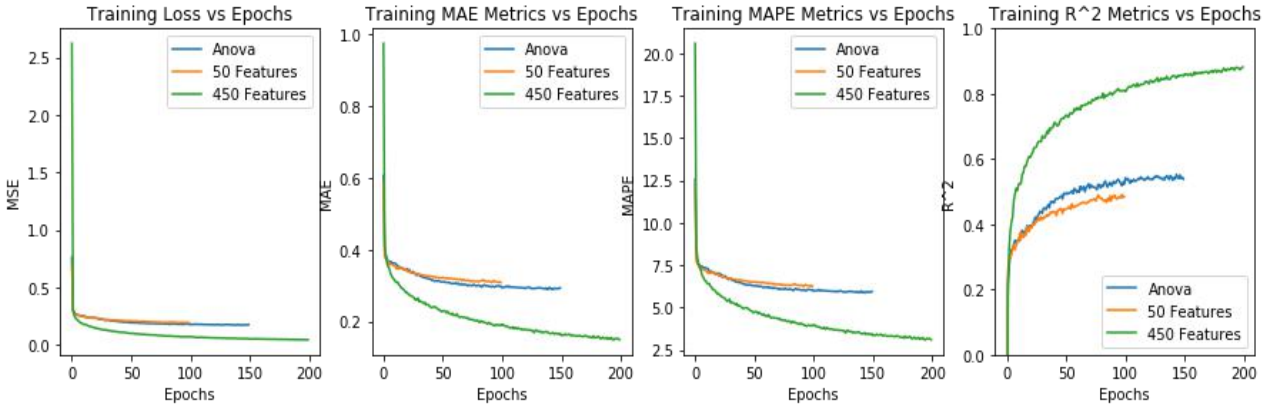
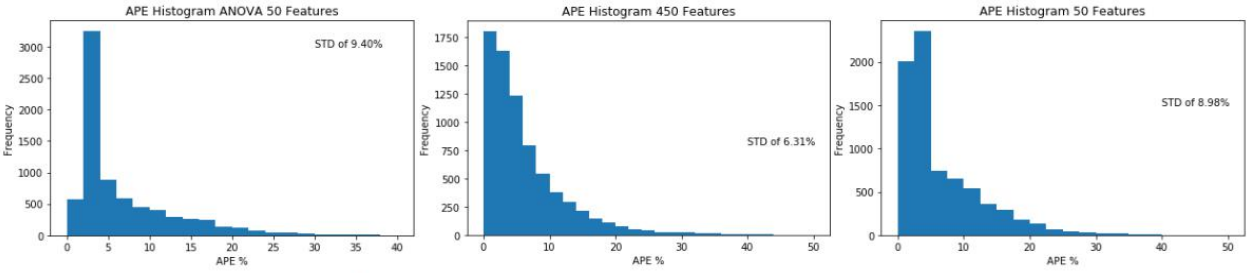


Figure 10. Results of ANN performance for each dataset after tuning.



Data	R^2	MAPE	MAE	Batch_size	No_epochs	Learning Rate
450 Features	0.432	6.31%	0.307	60	200	0.001
Anova 50 Features	0.332	6.82%	0.336	60	150	0.01
50 Features	0.394	6.50%	0.323	50	100	0.01

Figure 11. APE distribution, MAPE and frequency for each cluster.

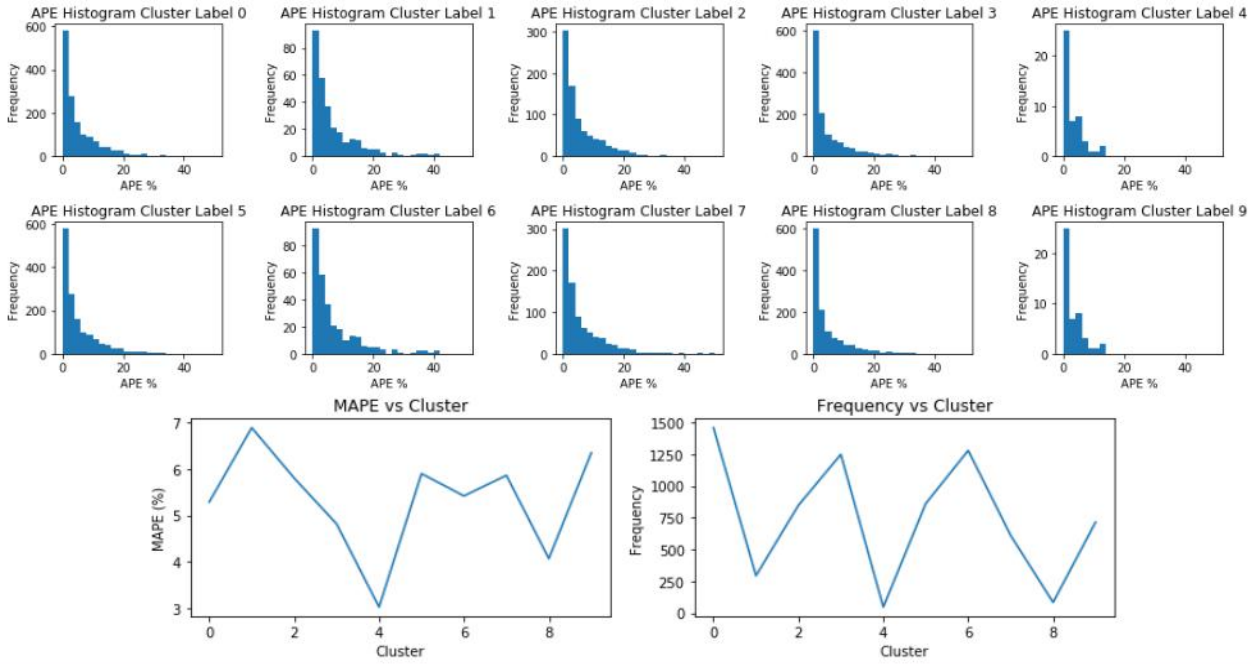


Figure 12. User Interface