Hyper Parameter Tuning

Hyper parameter tuning is the process by which the parameter in the algorithm are tuned to get the best possible model keeping the dataset constant.

Once the model in developed with the best features and with best split, we performed parameter tuning on all the 3 models viz Linear Regression, Random Forest and Neural Network.

HYPER PARAMETER TUNING IN LINEAR REGRESSION

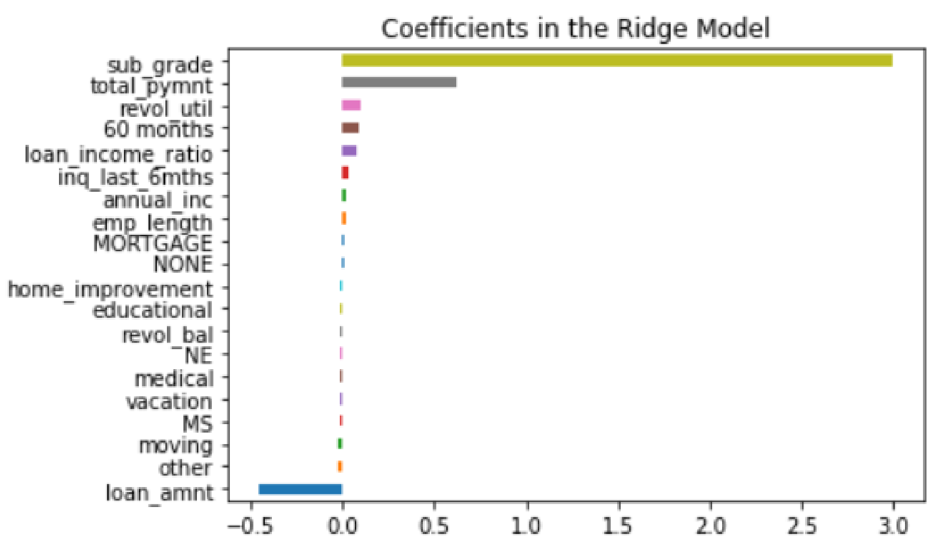
We used cross validation method for the regularizations. The model was trained with Lasso, Ridge and ElasticNet regularization and we were able to calculate the error metrics after each regularization.

Ridge Regularization also known as L2 penalty: - *Ridge regression is an L2 penalized model where we simply add the squared sum of the weights to our cost function*.

We designed our model with a series of alpha values and calculated the best alpha, then again for precision we modulated the alpha centered around the best alpha.
With the best alpha, we trained our model and calculated MAPE score to be 4.6997569430429325

We plotted graph to check the feature importance

```
Ridge picked 83 features and eliminated the other 1 features
```



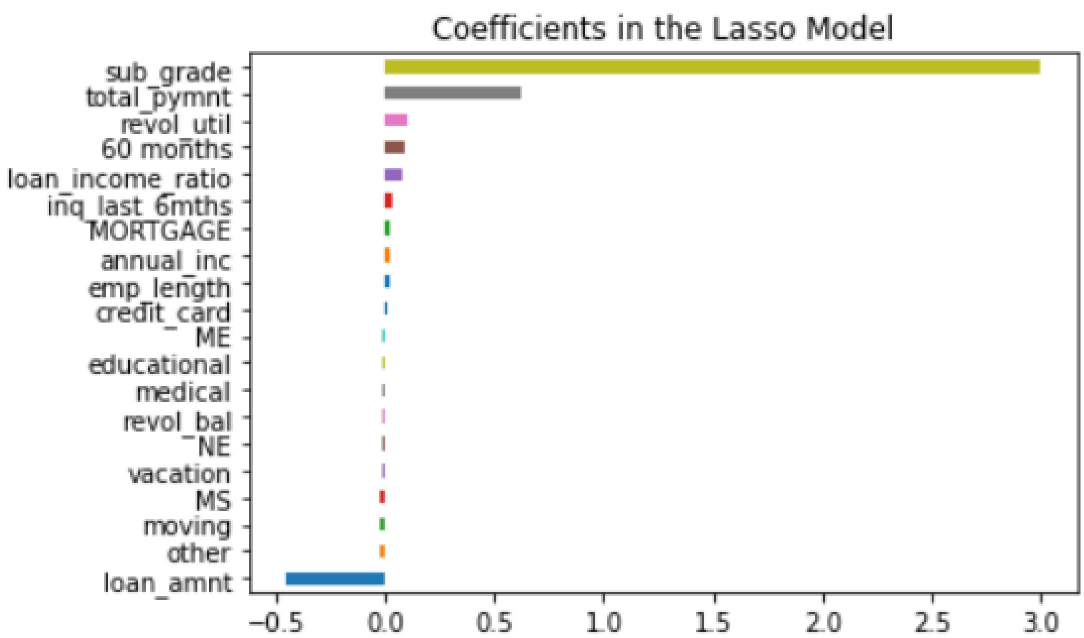
Sub-grade as the most importance is predicting the interest rate.

Lasso Regularization known as L1 Penalty: -*L1 regularization yields sparse feature vectors : most feature weights will be zero. Sparsity can be useful in practice if we have a high dimensional dataset with many features that are irrelevant.*

We designed our model with a series of alpha values and calculated the best alpha, then again for precision we modulated the alpha centered around the best alpha.
With the best alpha, we trained our model and calculated MAPE score to be 4.6997569430429325

We plotted graph to check the feature importance: -

```
Lasso picked 79 features and eliminated the other 5 features
```

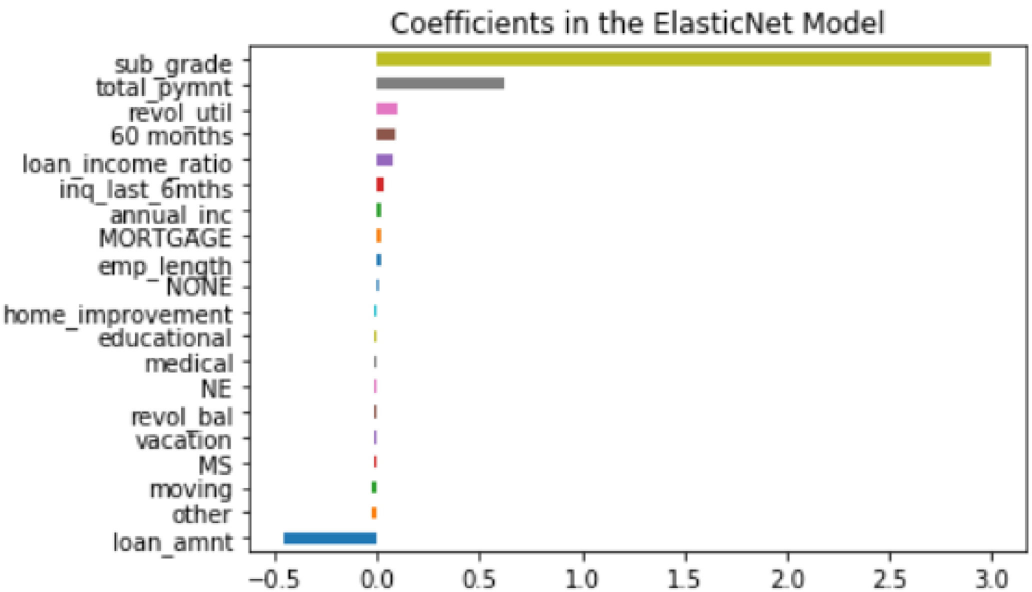


ElasticNet Regularization (L1 & L2 Penalty) : - *ElasticNet is a compromise between Ridge and Lasso regression. It has a L1 penalty to generate sparsity and a L2 penalty to overcome some of the limitations of Lasso*

In ElasticNet regularization, we designed our model with a series of L1 ratios and trained our model to find out the best L1 ratio and then for precision we modulated the L1 ratio centered around the best L1 ratio.
Keeping that ratio constant, we applied a series of alpha to get the best alpha for our model.

We trained our model with the best L1 ratio and the best alpha and calculated MAPE score to be 4.699797676032727
We plotted the graph for the best features

```
ElasticNet picked 80 features and eliminated the other 4 features
```



CONCLUSION

The model performed slightly better with L1 regularization. It do not need L2 regularization to overcome any shortcoming which is reflected in ElasticNet. Thus without any regularization also the model performs good, but for slightly better performance we can go with L1 regularization.

Process	TEST MAPE	Best Alpha
Linear Regression with Lasso regularization (L1 Penalty)	4.699522969272883	6e-05
Linear Regression with Lasso regularization (L2 Penalty)	4.699756943042921	3.1500000000000004
Linear Regression with ElasticNet regularization (L1 and L2 Penalty)	4.699797676032727	6e-05

HYPER PARAMETER TUNING IN RANDOM FOREST

For Random Forest we performed hyper parameter tuning by using Grid Search. We defined a parameter grid which was feed to the GridSearchCV. The model was trained with the best possible hyper parameters.

The best hyper parameters: No of estimator: 30, no of depth: 8

Hyper Parameter tuning in Neural Network

For Neural network we performed hyper parameter tuning by using Grid Search. We defined a parameter grid which was feed to the GridSearchCV. The model was trained with the best possible hyper parameters.

The best hyper parameters {'activation': 'identity','learning_rate': 'constant','max_iter': 300, 'solver': 'lbfgs'}