NATIONAL UNIVERSITY OF SINGAPORE,
SCHOOL OF COMPUTING
BT2101: DECISION MAKING METHODS AND TOOLS
AY 2018/2019, SEMESTER 1

INDIVIDUAL HW 2: PREDICT CUSTOMER CHURN

Supervisor:
Assoc. Prof. Keith Barrett Carter

A0192917U

[05/10/2018]

1. Data Analysis and Pre-Processing

For this assignment the data set was provided and minimal clean up was necessary. Mainly converting all features to numerical values and dropping some instances with missing values. Some features were duplicated after creating dummy variables so those had to be dropped too. This was discovered looking at the correlation heatmap of our data as can be seen in Fig. 1. A 80:20 train/test split was used. See the Jupyter notebook for a detailed rundown. Analysing the correlations of the features, one possible classification model was quickly dropped. Naïve Bayes is based on the assumption that features are uncorrelated[1], which as our analysis shows is not really the case.

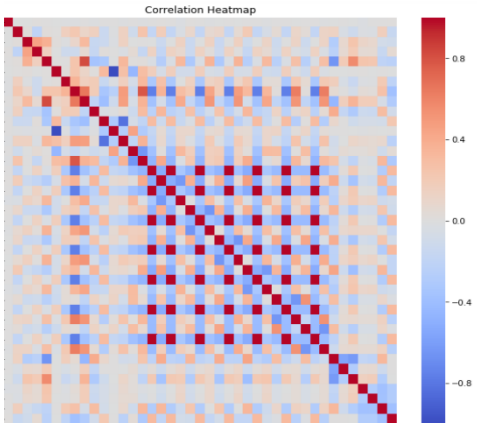

Figure 1: Feature Correlation Heatmap, dark red squares show perfect correlation

2. Model Process

The original approach was to try out some of the most popular classification models, also looked at in class, and then try some additional ones to see what could be achieved. This approach was abandoned due to minimal differences between the models and diminishing returns. Some models not in the notebook were quickly tested, to no avail, in case there could be some astounding results. KNN was a straightforward implementation using cross validation to find the optimal k.[2]

Trying to optimize Logistic Regression proved to be trickier. Different regularization functions and strengths were tested, as well as standardizing the feature matrix.[3] All had very little effect as the model was never overfitted and the regularization never really took effect. We settled on Lasso regularization with next to no strength, as in theory this should be better for correlating features.[4] Standardization was tested multiple times throughout the project with barely any effect, so it was never applied after this.

The decision tree classifier badly overfitted the data, which was corrected by limiting the depth of the tree. Cross validation was again used to find the optimal depth. To optimize the classifier both ‘gini’ and ‘entropy’ criterion were tested with no

[1] https://www.techopedia.com/definition/32335/naive-bayes
[2] https://stats.stackexchange.com/questions/126051/choosing-optimal-k-for-knn
[3] https://sebastianraschka.com/Articles/2014_about_feature_scaling.html
[4] https://stats.stackexchange.com/questions/264016/why-lasso-or-elasticnet-perform-better-than-ridge-when-the-features-are-correlat

considerable difference. To improve the classifier further we ran it through a bagging algorithm. Random forest and Adaboost were also applied as further ensemble methods using the same depth criteria were applicable.

3. Model Results

Table 1: Model Summary

Model	Test Accuracy	10fold CV Accuracy
33 Nearest Neighbours	0.7839	0.7836
Logistic Regression	0.8024	0.8053
Decision Tree	0.7797	0.7959
Bagging	0.7925	0.8020
Random Forest	0.7953	0.7948
Adaboost	0.7974	0.8082

To quickly summarize the model results let's look at Table 1. After all the tried approaches to optimize the models, they all fall within a margin of error of each other regarding their accuracy. 80% accuracy is by no means a bad result, however the advantages of different models seem rather ineffective here. It is for this reason that training additional models was abandoned. The best and most stable model would be Logistic Regression, Figs. 1-3 show it's performance.

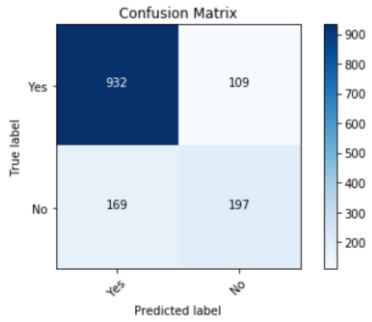

Figure 2: Confusion Matrix

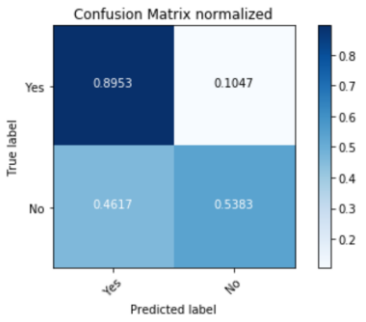

Figure 3: CM normalized

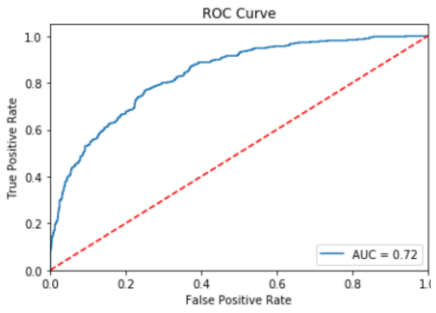

Figure 4: ROC Curve

We can see that the proportion of False Positives is far greater than that of False Negatives. This would mean our model would predict a customer leaving when he really isn't more often than predicting a customer not leaving when he actually is. This seems favourable.

Using the random forest classifier, the individual feature importance's were calculated. The 10 most important features can be seen in Fig. 5 and are not that surprising. No features really stand out.

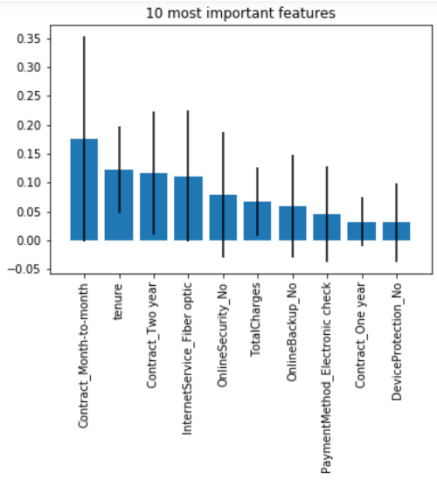

Figure 5: 10 most important features

4. Reflection

Based on our findings we can conclude that the performance of our models is probably limited by the data we have, to around 80%. To further increase our accuracy the best approach would be to get additional data. For instance, getting market data to see the influence competitors have would be interesting. That would allow to model the influence of substitute products, the fluctuations in the economy and the purchasing power too. Overall our accuracy is quite satisfying, but I would have liked more time and data to improve the models further. The performance tweaking done in this assignment sadly had barely any effect.