Artificial Intelligence Solutions & Development

Telco Customer Churn Analysis

Han Yew Yong Matthew ID: 1881707B



# Introduction

This is a project to predict mobile customers churn using algorithms such as logistic regression, random forest and support vector machine.

Customers are the lifeblood of any companies. In a highly competitive mobile services market, retaining customers is as important as finding new ones. It would be ideal if telco companies could detect customers with high probability to churn and do the necessary customer recovery services to retain the customer.

The same logic can be applied to employee churn for HR departments.

# Data Preparation and Feature Selection

Data can be obtain from Kaggle’s repository at <https://www.kaggle.com>

Data from Kaggle comes in an almost clean form. Only a small amount of touch up cleaning is needed in order to prepare our data for our unique problem statement. The cleaning process can be broken down into the following below.

1. ‘TotalCharges’ column contains NaN values. After a browse of our data, I noticed these are rows with ‘Tenure’ = 0. It could be a reason why customers churn hence, I keep the rows and set the NaN values to 0.
2. As most of the columns are of categorical nature, a type conversion into 1 and 0 representation was performed for the machine learning algorithm to process.
3. The customer ID column was removed as we don’t need to feed that into the machine.
4. A plot of the correlation matrix shows that ‘TotalCharges’ is highly correlated with ‘Tenure’ and ‘MonthCharges’. To avoid multicollinearity in our logistic regression model, the ‘TotalCharges’ was removed.
5. A look at the ‘Churn’ column shows that the dataset displays class imbalanced problem. To solves this, the dataset could be upsampled or downsampled. Downsampled was chosen so that every row could describe a unique customer.
6. Lastly, the prepared data is split into training set and testing set.

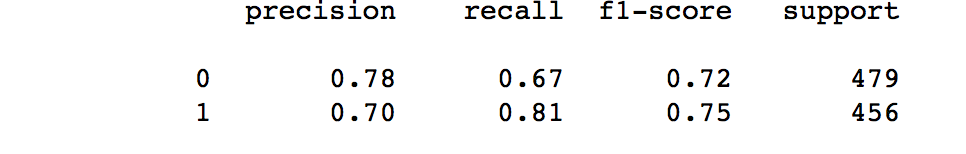
# Methods of Analysis

## Logistic Regression

Logistic Regression can be used not only for regression problem but for binary classification problem, such as this case. In a logistic regression, the logarithmic loss function is used to calculate the cost for misclassifying.

## Results and Analysis

1. Accuracy of logistic regression classifier on test set: 0.74



1. A **classification report** shows the *precision, recall, f1-score* and *support*.

The precision is the ratio tp / (tp + fp) where tp is the number of true positives and fp the number of false positives. The precision is intuitively the ability of the classifier to not label a sample as positive if it is negative. **What percent of your predictions were correct?**

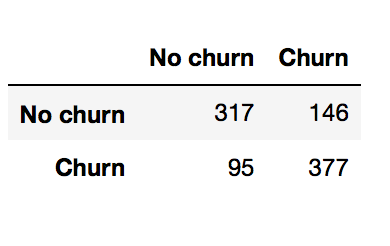
The recall (**sensitivity**) is the ratio tp / (tp + fn) where tp is the number of true positives and fn the number of false negatives. The recall is how many positives were correctly identified. **What percent of the positive cases did you catch?**

**Specificity** – how good a test is at avoiding false alarms. A test can cheat and maximize this by always returning “negative”.

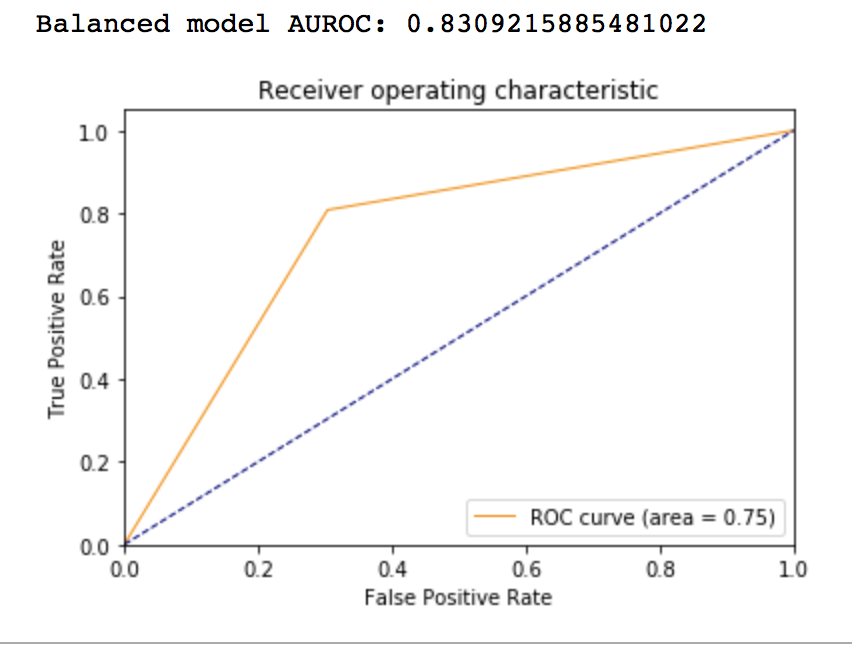
The F-1 score can be interpreted as a weighted harmonic mean of the precision and recall, where an F-1 score reaches its best value at 1 and worst score at 0. F-1 score = 1.0 means recall and precision are equally important. **What percent of positive predictions were correct?**

The support is the number of occurrences of each class in Y Test. Shows whether class is imbalanced or not.

1. A **confusion matrix** shows visually the performance of the classification model.



1. **ROC plot** is a commonly used graph that summarized the performance of a classifier over all possible thresholds. The AUC tells how much the model is capable of distinguishing the classes. When AUC = 0.5, it means the model has no class separation capacity whatsoever.



An ROC curve demonstrates:

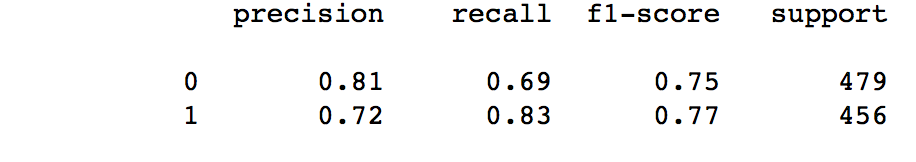
1. The tradeoff between sensitivity and specificity (any increase in sensitivity will be accompanied by a decrease in the specificity.
2. The closer the curve comes to the 45-degree diagonal, the less accurate the test.
3. The closer the curve approaches the top left hand corner, the more accurate the test.

Random Forest Classifier

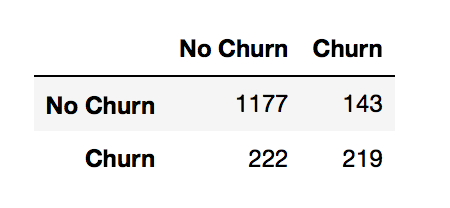
Like Logistic Regression, the Random Forest algorithm can be used for both classification and regression algorithm. It is a very popular classification method. It creates multiple forests of decision trees and merges them together to give a more accurate prediction.

Results and Analysis

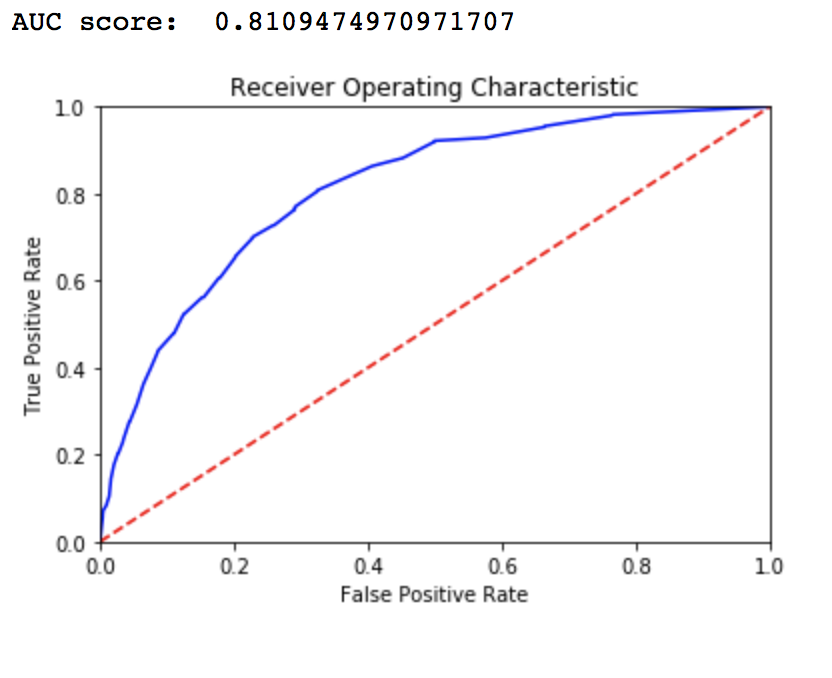
1. Accuracy of random forest classifier on test set: 0.79
2. The predict\_proba method gives the prediction of the probability of churn or no churn for a customer.
3. A classification report shows the overall performance of the classifier.



1. A confusion matrix shows intuitively the performance of the classifier in visual form.



1. AUC of 0.85 shows that the model is good in differentiating the classes.

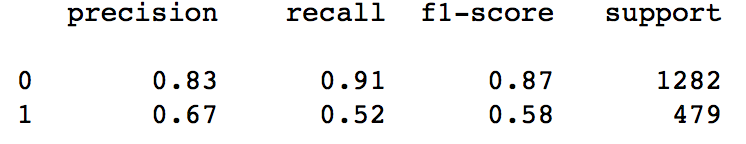


Support Vector Machine

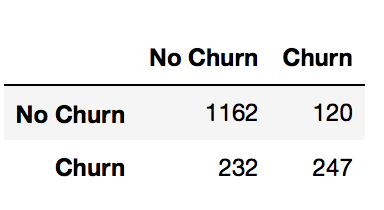
SVM is a supervised ML algorithm which can be used for both classification and regression too. It is very popular in classification problems. It works by finding the hyper-plane that differentiate the two classes very well.

Results and Analysis

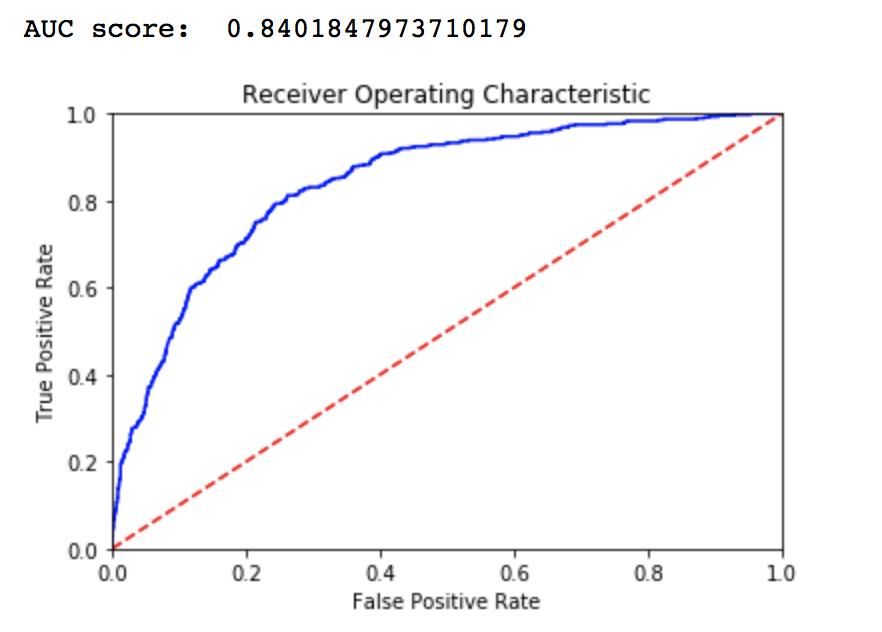
1. Accuracy of Support Vector Classifier on test set: 0.80
2. A classification reports shows the overall performance of the classifier

.

1. A confusion matrix displays a more intuitive feel.



1. AUC score of 0.84 shows SVM is the best ML classifier for the problem statement.



Conclusions

Every single steps build on top of another in the data science workflow. From data preparation, to feature selection, and to feeding the data into the machine learning algorithm, and plotting a visually intuitive graph. Each step is important and if time permit, I would chart the coefficients of the features to illustrate important features affecting the churn.

And there are many more algorithm out there to experiment which would have given a better fit. Alternatively, we could work with the best performing SVM out of the three by tuning its parameter further if time permits.

References

**Websites**

[www.dataschool.io](http://www.dataschool.io)

[www.dataskunksworks.com](http://www.dataskunksworks.com)

[www.kaggle.com](http://www.kaggle.com)

www.elitedatascience.com

**Books**

Python for Data Science Handbook