**Predicting Customer Churn**

**Using Bank Customer Churn Dataset**

Grant Pennington, [gpennington2@bellarmine.edu](mailto:gpennington2@bellarmine.edu)

**ABSTRACT**

This project goal was to do exploratory data analysis and to use either the Voting Classifier or Random Search CV to decide the best model for the chosen dataset. I used a Bank Customer Churn dataset that was designed to help banks predict which of their customers may leave their bank. Multiple machine learning models are used for this project. Logistic Regression, Support Vector Classifier, Decision Tree Classifier, and Random Forest Classifier. The Voting Classifier is what I decided to choose over the Random Search CV technique.

1. **INTRODUCTION**

For my project, I chose a bank customer churn prediction dataset from Kaggle. This dataset is for ABC Multinational bank and the goal of this dataset is to predict if a customers may leave or not. I used the Voting Classifier model to find out which ML Regression model I should use. I tested Logistic Regression, SVC, Decision Tree, and Random Forest against the dataset.

1. **BACKGROUND**
   1. *Data Set Description*

This dataset was originally created for ABC Multinational Bank to try to predict customer churn. Banks are always wanting to hold onto their customers and increase their revenue to maintain their business. The dataset is a collection of customer data from account holders of ABC Multinational Bank.

This data set can be found at the following link on Kaggle.com. [Bank Customer Churn Dataset | Kaggle](https://www.kaggle.com/datasets/gauravtopre/bank-customer-churn-dataset)

* 1. *Machine Learning Model*

The ML Models used for this project are Logistic Regression, Support Vector Classifier (SVC), Decision Tree Classifier, and Random Forest Classifier. I used the Voting Classifier to detect which model performs best on the dataset.

**Logistic Regression** is a type of regression analysis. It is a predictive modeling technique which that is used to find the relationship between a dependent variable “Y” and an independent variable “X”. Logistic Regression has a binary outcome, meaning the event either happens or it does not happen, often represented as 0 and 1.

**Support Vector Classifier** is part of SVM which is **discriminative classifier** formally defined by a separating hyperplane. An SVM model represents data points as points in space that are mapped, by category, into different levels to make them clearly distinguishable. This allows SVM models to perform linear classification on non-linear data.

**Decision Tree Classifier** is classified as a supervised machine learning model. This means they use data that is already labeled to train a model to make predictions on unlabeled data. Each node of the decision tree represents a certain outcome, and each node often branches into two child nodes. These child nodes will be the resulting outcomes from the parent outcome.

**Random Forest Classifier** is classified as a supervised machine learning model as well. It contains multiple decision trees like a “forest”. It uses randomness to increase accuracy and to help avoid overfitting. The Random Forest Classifier takes random data samples and gets predictions from every decision tree.

According to the Voting Classifier, I found the Random Forest Classifier yields the highest resulting score.

1. **EXPLORATORY ANALYSIS**

This data set includes 10,000 entries with 12 columns of various data types. There were no missing values in this dataset. There were two categorical columns, gender, and country. I removed the country column as I saw it was not important to the target column, churn.

**Table 1: Data Types**

|  |  |
| --- | --- |
| *Variable Name* | *Data Type* |
| customer\_id | int64 |
| credit\_score | int64 |
| country | object |
| gender | object |
| age | int64 |
| tenure | int64 |
| balance | float64 |
| products\_number | int64 |
| credit\_card | int64 |
| active\_member | int64 |
| estimated\_salary | float64 |
| churn | int64 |

1. **METHODS**
   1. *Data Preparation*

I did not have to do much pre-processing for my data set. I dropped one column, the country column, because It was unimportant to predicting the target and it was categorical. I had no missing values, so no dummies had to be created.

* 1. *Experimental Design*

The parameters that performed the best are for my Logistic Regression I used the parameters lbfgs solver, multinomial class, and a max iteration of 10,000. Random Forest Classifier had no parameters. For Decision Tree Classifier I used two different criterions. One with gini and one with entropy. Both had a max depth of 10 and a random state of 0. For SVC I used auto gamma and set probability to true.

Table X: Experiment Parameters

|  |  |
| --- | --- |
| **Experiment Number** | **Parameters** |
| 1 | All four (4) raw features with 80/10/10 split for train, validate, and test |
| 2 | All four (4) normalized features with 80/10/10 split for train, validate, and test |
| 3 | All four (4) raw features with 70/15/15 split for train, validate, and test |
| 4 | All four (4) normalized features with 70/15/15 split for train, validate, and test |

* 1. *Tools Used*

The following tools were used for this analysis: Python running in Google Colab was used for all analysis and implementation. In additio, the following libraries were also used: Pandas 0.18.1, Matplotlib 1.5.3, Seaborn 0.7.1, SKLearn 0.18.1.

1. **RESULTS**
   1. *Mean square Error*

My mean square error result was 0.373.

* 1. *Discussion of Results*

The Voting Classifier depicted the best model for my dataset was the Random Forest Classifier.

My Scores were: Logistic Regression had 0.8145, Random Forest Classifier has 0.857, SVC had 0.853, Decision Tree with gini had 0.8385, and the Decision Tree with entropy was 0.845. Below is my classification report for the Random Forest.

Calendar

Description automatically generated

* 1. *Problems Encountered*

I did not run into many problems. The only problems may have been the dataset I chose, however, it was difficult trying to find and choose a dataset that was good but also somewhat interesting.

* 1. *Improvements/Future Work*

Improvements could have been a better dataset. Some of the data was limited and did not correlate much with the target column we needed to predict.

1. **CONCLUSION**

In conclusion, my model performed okay, however, I think that the dataset I chose was not the best for predicting. The Random Forest Classifier yielded the best results for my dataset. I found that a lot of the columns were not really related and there were not a lot of columns. In terms of preprocessing the dataset was easy to work with.

**REFERENCES**

<https://datagy.io/sklearn-decision-tree-classifier/#:~:text=What%20are%20Decision%20Tree%20Classifiers%3F%20Decision%20tree%20classifiers,that%20can%20be%20used%20to%20make%20a%20prediction>.

[Random Forest Classifier: Overview, How Does it Work, Pros & Cons | upGrad blog](https://www.upgrad.com/blog/random-forest-classifier/)

[Classifying data using Support Vector Machines(SVMs) in Python - GeeksforGeeks](https://www.geeksforgeeks.org/classifying-data-using-support-vector-machinessvms-in-python/#:~:text=A%20Support%20Vector%20Machine%20%28SVM%29%20is%20a%20discriminative,outputs%20an%20optimal%20hyperplane%20which%20categorizes%20new%20examples.)

[What is Logistic Regression? A Beginner's Guide [2022] (careerfoundry.com)](https://careerfoundry.com/en/blog/data-analytics/what-is-logistic-regression/)