VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

Winter Semester 2021-2022

A CAL Project Report

on

Bank Chunk Prediction

to be submitted in partial fulfilling of the requirements for the course on

Data Mining and Business Intelligence – ITA5007

(B1+TB1 / B2+TB2)

Submitted By:

Vivek Kumar 21MCA0153
Shani Gupta 21MCA0008

Submitted To:

Prof. PRABADEVI B

TABLE OF CONTENTS

ABSTRACT

1. Introduction
2. Review 1 (Survey, Analysis).
    a. Problem definition
    b. Dataset Description
    c. Review on Existing System
3. Review 2 (Design)
    a. Methodology
        i. Module Description
            1. Data exploration
            2. Pre-processing
            3. Logistic Regression
            4. Random Forest
            5. SVM (Support Vector Machine)
            6. Decision Tree
        ii. Algorithms used
            1. Justification for choosing the models
        iii. Flow diagram of your model
        iv. Dataset after preprocessing
        v. Dataset split(train and test)
4. Review 3 (Code)
    a. Implementation
        i. Software and hardware description
        ii. Output screenshots
    b. Confusion Matrix
    c. Comparison of the models used
    d. Comparison graph
5. Conclusion
6. References

ABSTRACT

In the era of big data, customer churn is a big problem faced by banks in the increasingly competitive market. Therefore, it is very important to establish an efficient early warning system for Customer Churn by mining the information that has an impact on churn from massive customer data. The purpose of this paper is to analyze the quarterly user data of banks, and establish user churn prediction model by using ensemble learning algorithm such as Random Forest, Naïve Bayes, SVM so as to improve the accuracy of prediction, so as to achieve the purpose of helping banks save costs. The experimental results show that the accuracy rate of the model has reached 90%, and the AUC value is more than 80%. The model can be used to predict whether the user may be lost in the future, reserve enough time for the user retention activities, and provide a lot of valuable information to help marketing personnel to formulate feasible user retention scheme, which has a wide range of industry application prospects.

1. INTRODUCTION

New technology, regulation and change in demand has caused a rise of fintech companies challenging banks' dominant position in society. In times of intensified competition, customer turnover can pose a real threat for existing companies. Customer turnover, also referred to as customer churn, is when a customer leaves or ends an engagement with a company during a given time period . As a result of increasing competition, it is important for banks to maintain existing customers, as this is more cost-effective than acquiring new ones, in order to ensure their position in society.

In addition, new technology has increased banks' access to data, and thus data driven customer churn analysis is feasible. Taken together, there is a growing demand for customer churn analysis which studies a set of characteristics in order to predict customer churn. This has intensified the demand for predictive modelling built on for example statistical learning methods. Since, if a bank can predict customer churn, targeted marketing campaigns can be used to persuade

customers to keep their engagement. However, this raises the question of which statistical learning method can predict customer churn the best?

According to Verbeke, popular methods to predict churn probability is logistic regression, Naïve Bayes and decision trees, since they combine both good predictive performance with good interpretability. However, Naïve Bayes and decision trees have practical difficulties resulting in the need for additional methods. This study seeks to evaluate and analyze which additional methods are the best at predicting customer churn and compares logistic regression, random forest and K-nearest neighbor. Which additional method that is to prefer is evaluated through which method provides the most reliable predictions. In order to assess prediction reliability, evaluation measurements based on two cross-validation set approaches are compared. In conclusion, this study aims to find the best statistical learning method for customer churn prediction in addition to concluding which crossvalidation set approach yields the most reliable results.

We aim to accomplish the following for this study:

1. Identify and visualize which factors contribute to customer churn:
2. Build a prediction model that will perform the following:
    - Classify if a customer is going to churn or not
    - Preferably and based on model performance, choose a model that attach a probability to the churn to make it easier for customer service to target ow hanging fruits in their efforts to prevent churn

Data understanding

The bank customer churn data was obtained from Kaggle. It is stored in a csv file, named as "bank customer churn dataset". It has 14 columns, called features, including row number, customer id, surname, credit score, geography, gender, age, tenure, balance, number of products purchased through the bank, whether has a credit card, whether is an active member, estimated salary, and whether exited from the bank.

Prepare data

The categorical data, like the features geography and gender was handled with one-hot encoding. After checking the missing values in this dataset, it showed no missing values.

Data modelling

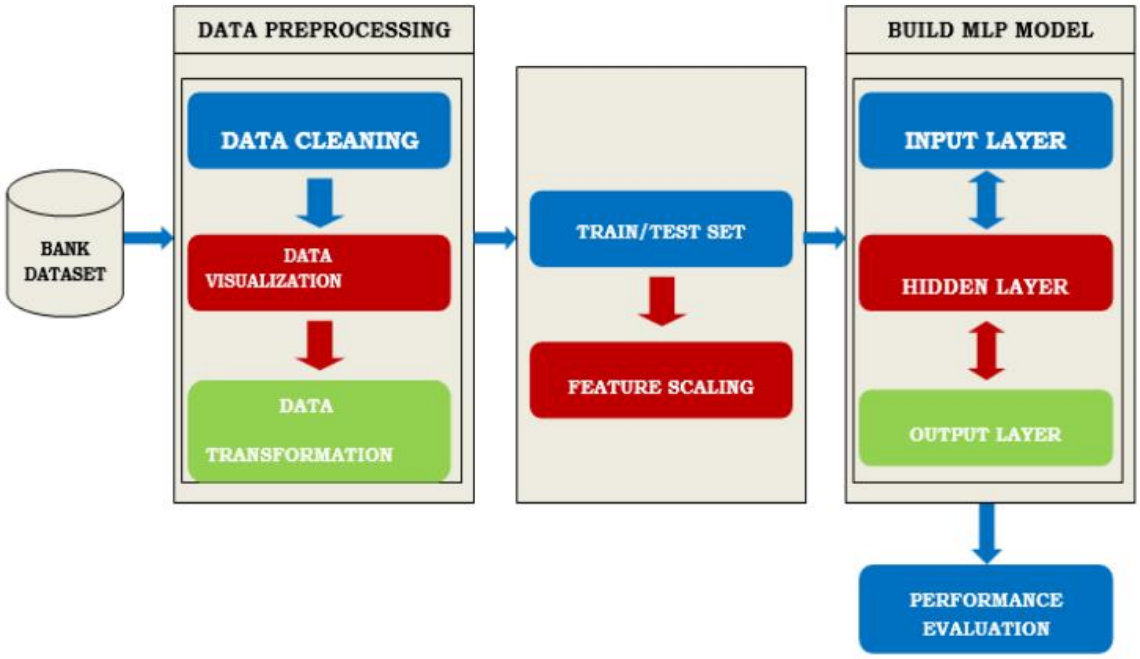
Studied the correlation between every features and churn outcome.

Divided the whole data into training data and test data.

Built several machine learning models (logistic regression, decision tree, support vector machine, and random forest) to predict the customer churn.

Evaluate the results

Evaluated the predictions with accuracy and confusion matrix.



Banks often use the customer churn rate as one of their key business metrics because the cost of retaining existing customers is far less than acquiring new ones, and meanwhile increasing customer retention can greatly increase profits.

Survey & Dataset Collection

a. Problem definition

With the growing competition in banking industry, banks are required to follow customer retention strategies while they are trying to increase their market share by acquiring new customers. It is shown that improving the retention rate by up to 5 % can increase a bank's profit up to 85 % . Additionally, attracting new customers costs more to any company rather than retaining the old ones who are likely to produce more profit.

Thus, banks should maintain their competitive advantage by taking the advantage of machine learning models to predict customer churn

b. Dataset Description

Data source: https://www.kaggle.com/mathchi/churn-for-bank-customers

Dataset selected for this study is publically available in kaggle.com1 . Variables included in the dataset are described in Table 1. Out of 13 variables, CustomerId and Surname need to be removed as they don't have any contribution to the classification purpose. We also replace binary values of the outcome variable (Exited) with "Stayed" and "Left" labels to have a better representation of outputs when visualizing results and discussing the performance. We will use data entirely in the analysis and don't follow any sampling procedure because we need the training sample to be sufficiently large.

Current data doesn't have any missing value in none of its 10000 observations and thus, there won't be any concern in this regard. However, customers who stayed with banks (7963 customers) are around four times the number of those who left (2037 customers). Therefore, data is imbalance with respect to the outcome variable and this concern needs to be addressed in the modeling section. We also need to figure out potential outliers in at each class of the outcome variable for all numeric variables. As depicted in Figures 1 to 4, Balance and Estimated Salary don't include any outliers. Credit Score and Age have only 11 and 13outliers, respectively, in the "Left" class and Age includes 486 outliers in the "Stayed" class. Therefore, in general, there is not a serious concern with regard to outliers as the ratio of outliers-where they were detected- to the size of data is reasonably low. However, we will

analyze data in the absence of these 486 outliers - associated with Age (still consisting only 0.05% of data) - to address any possible noise from these data points in the evaluation of the final model.

RowNumber—corresponds to the record (row) number and has no effect on the output.
CustomerId—contains random values and has no effect on customer leaving the bank.
Surname—the surname of a customer has no impact on their decision to leave the bank.
CreditScore—can have an effect on customer churn, since a customer with a higher credit score is less likely to leave the bank.
Geography—a customer's location can affect their decision to leave the bank.
Gender—it's interesting to explore whether gender plays a role in a customer leaving the bank.
Age—this is certainly relevant, since older customers are less likely to leave their bank than younger ones.
Tenure—refers to the number of years that the customer has been a client of the bank. Normally, older clients are more loyal and less likely to leave a bank.
Balance—also a very good indicator of customer churn, as people with a higher balance in their accounts are less likely to leave the bank compared to those with lower balances.
NumOfProducts—refers to the number of products that a customer has purchased through the bank.
HasCrCard—denotes whether or not a customer has a credit card. This column is also relevant, since people with a credit card are less likely to leave the bank.
IsActiveMember—active customers are less likely to leave the bank.
EstimatedSalary—as with balance, people with lower salaries are more likely to leave the bank compared to those with higher salaries.
Exited—whether or not the customer left the bank.

Variable	Type	Definition	Minimum	Maximum	Mean	Std. Deviation
CustomerId	Nominal	Customer ID				
Surname	Text					
CreditScore	Interval	Customer's credit score	350	850	650.53	96.65
Geography	Nominal	France, Germany, Spain				
Gender	Nominal	Female, Male				
Age	Ratio		18	92	38.92	10.49
Tenure	Interval	Tenure of deposit	0	10	5.01	2.89
Balance	Ratio		0	250898.1	76485.89	62397.41
NumOfProducts	Interval	Number of bank account affiliated products the customer has	1	4	1.53	0.58
HasCrCard	Binary	Does the customer have a credit card through the bank? (Yes=1, No=0)	0	1	0.71	0.46
IsActiveMember	Binary	Is the customer an active member? (Yes=1, No=0)	0	1	0.52	0.50
EstimatedSalary	Ratio	Estimated salary of the customer	11.58	199992.5	100090.2	57510.49
Exited	Binary	Did the customer leave the bank within the last 6 month? (Yes=1, No=0)	0	1	0.2	0.40

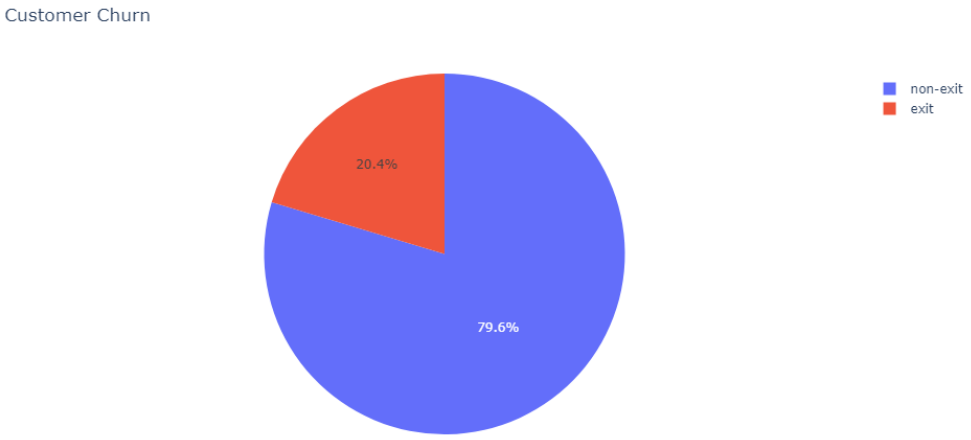
As we know, it is much more expensive to sign in a new client than keeping an existing one. It is advantageous for banks to know what leads a client towards the decision to leave the company. Churn prevention allows companies to develop loyalty programs and retention campaigns to keep as many customers as possible.

the majority of customers with credit score between 600 and 700, the distribution is fairly normal in this respect. Given majority of customers at around 40 year old, age is also fairly normal with a slight skewness to right. However, majority of clients have '0' balance while that of the remaining ones follows a decent normal distribution.

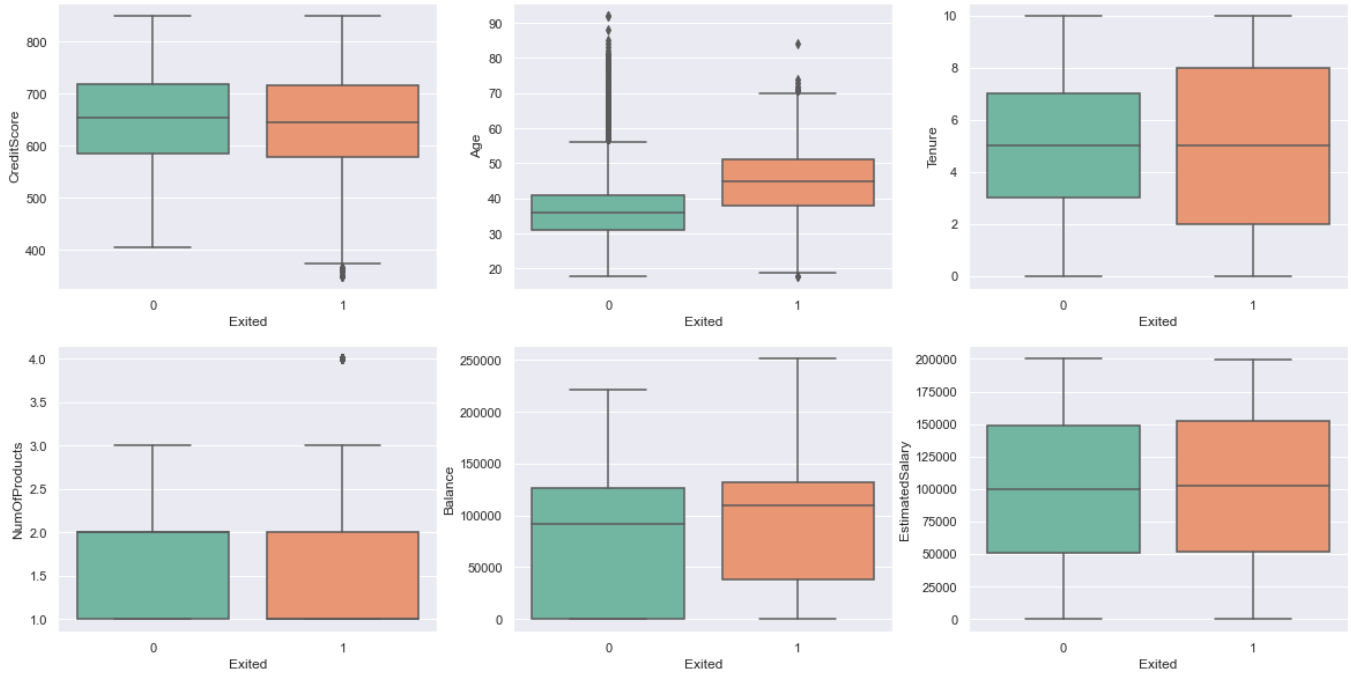
Methodology:

Dataset selected for this study is publically available in kaggle.com1. Variables included in the dataset are described Out of 13 variables, CustomerId and Surname need to be removed as they don't have any contribution to the classification purpose. We also replace binary values of the outcome variable (Exited) with "Stayed" and "Left" labels to have a better representation of outputs when visualizing results and discussing the performance. We will use data entirely in the analysis and don't follow any sampling procedure because we need the training sample to be sufficiently large.

Current data doesn't have any missing value in none of its 10000 observations and thus, there won't be any concern in this regard. However, customers who stayed with banks (7963 customers) are around four times the number of those who left (2037 customers). Therefore, data is imbalance with respect to the outcome variable and this concern needs to be addressed in the modeling section. We also need to figure out potential outliers in at each class of the outcome variable for all numeric variables. As depicted in Figures 1 to 4, Balance and Estimated Salary don't include any outliers. Credit Score and Age have only 11 and 13outliers, respectively, in the "Left" class and Age includes 486 outliers in the "Stayed" class. Therefore, in general, there is not a serious concern with regard to outliers as the ratio of outliers-where they were detected- to the size of data is reasonably low. However, we will analyze data in the absence of these 486 outliers - associated with Age (still consisting only 0.05% of data) - to address any possible noise from these data points in the evaluation of the final model.



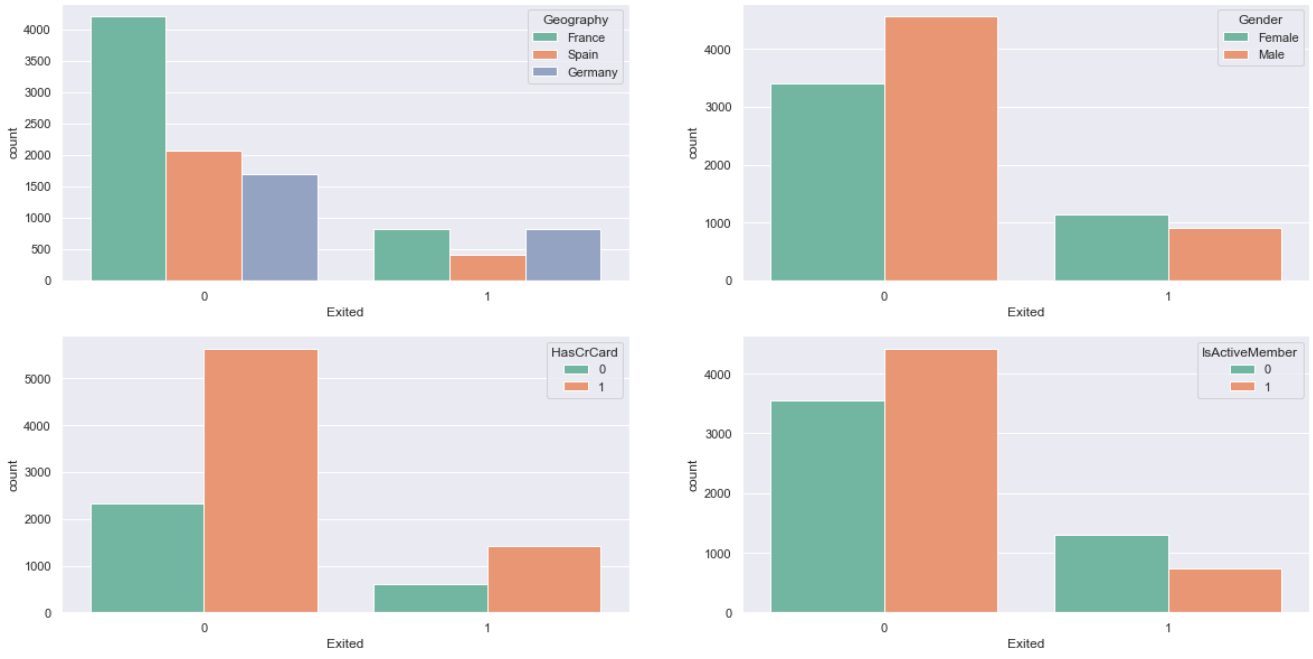
Analyze the relationship between target "Exited" and other Numerical features



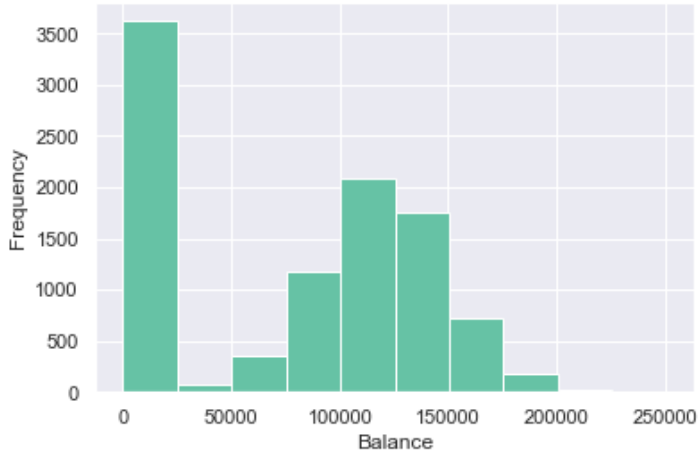
Analyze correlation among "Exited" and other Categorical Features

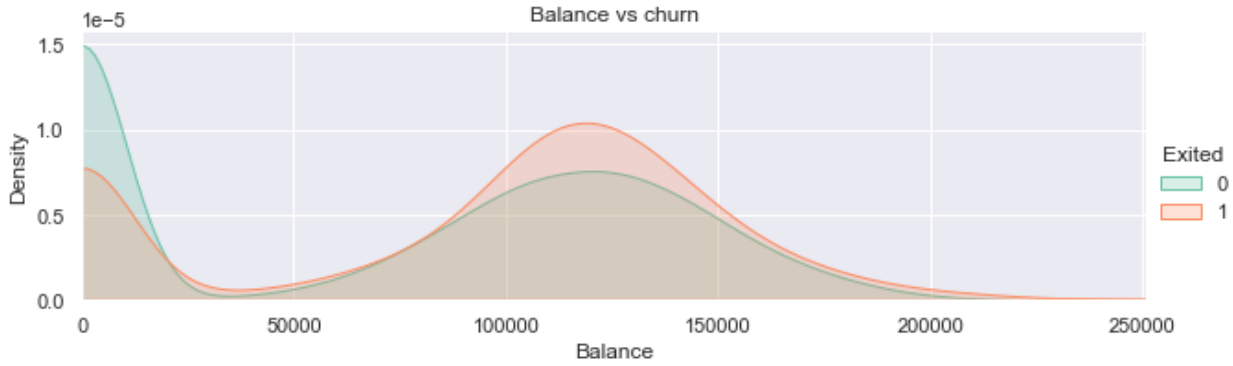
Note that:

In sns.countplot(), y is always the count of each class of x feature and hue is the z-axis data which must be a categorical features. Then the plot show the count of intersection of x and hue. So count plot is usually good to visualize categorical data



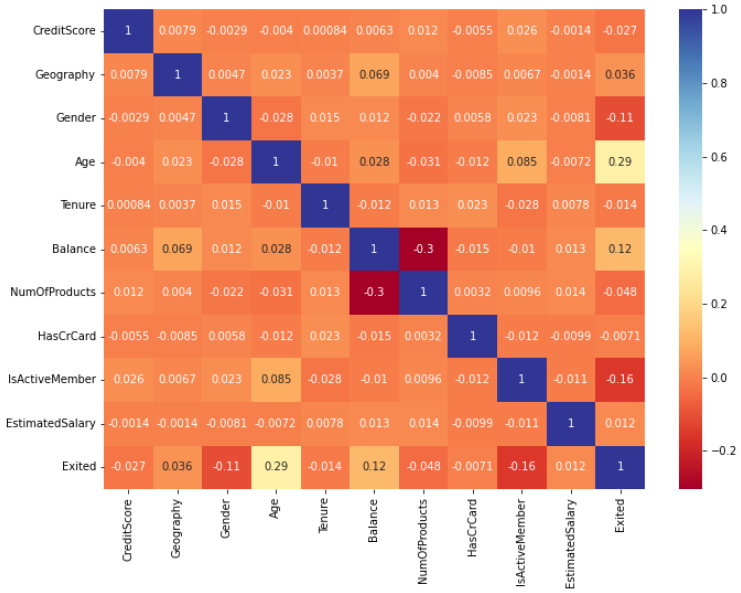
balance vs. churn

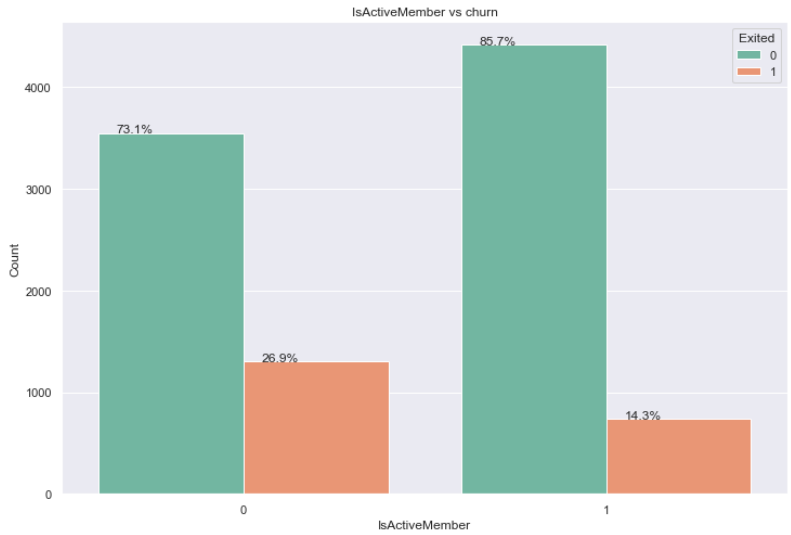




The situation of the customer churn in the studied dataset

Result: In this dataset, there is a rate of 20.4% for the customer churn from bank.





Exploratory Data Analysis

In order to explore the dependency between churning behavior and classes of categorical variable included in the model, we visualize the distribution of two types of customers - as the dependent variable (DV) - at each class and also perform Chi-Square test to detect significant dependencies. Data is pretty balanced across males and females (Figure 7) and females represent significantly lower churning rate than males. (X-squared = 112.92, df = 1, p-value < 2.2e-16). Data is also pretty balanced between banks from Germany and Spain while records from France-based banks are in majority, around twice those of other two banks (Figure 8) and customers of French banks associate with a significantly lower churning rate (X-squared = 301.26, df = 2, p-value < 2.2e-16). Having credit card from a bank (Figure 9) doesn't correspond with a significant difference in its churning rate (X-squared = 0.47134, df = 1, p-value = 0.4924) while active customers associate (Figure 10) with a higher retention rate rather than non-active ones (X-squared = 242.99, df = 1, p-value < 2.2e-16). As indicated in Figure 112 , customers who purchased one or two products

stay with the bank for the most part. On the other hand, most of those who purchased three or four products left the bank (X-squared = 1503.6, df = 3, p-value < 2.2e-16).

Model Selection and Evaluation

Regarding the binary nature of the outcome variable, current problem is well-suited with almost all supervised classification algorithms and tree-based models appear to be the popular ones as suggested by the literature. Reminding that our data lacks missing values and redundant features; and also includes not a large number of IVs which result in irrelevant features, computational costs and overfitting issues are likely to determine why some models might be preferred to others. To this end, we first make a comparison between the performances of different classification techniques in order to candidate two competing models for further analysis.

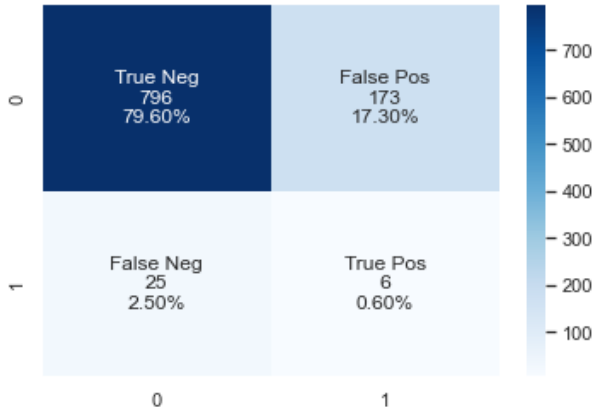
- Models to try:
- Logistic Regression
- Decision Tree
- Random Forest
- SVM

Model Evaluation

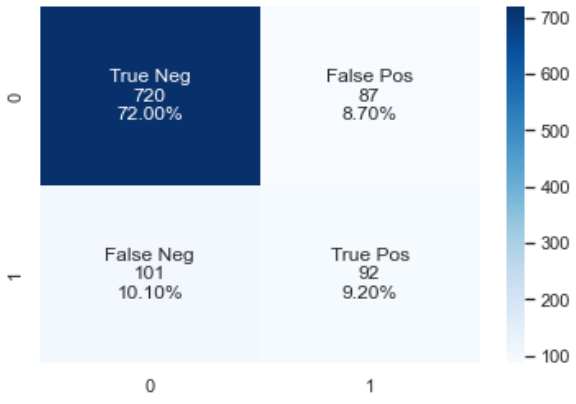
- Cross-Validation to assess models / estimate Model performance
- AUC (Area under curve)

Divide The Whole Dataset Into Training Data And Test Data

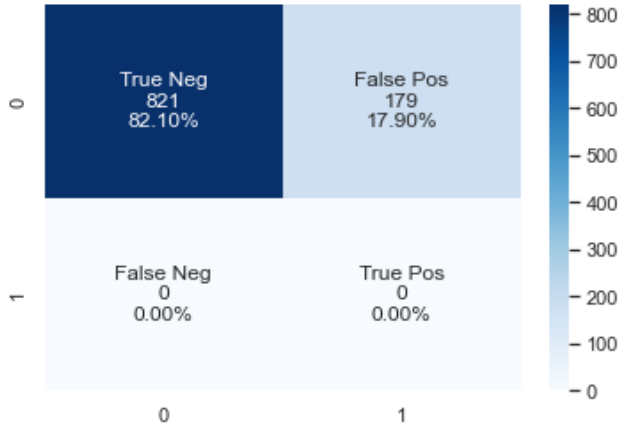
	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary
0	619	0	0	42	2	0.00	1	1	1	101348.88
1	608	2	0	41	1	83807.86	1	0	1	112542.58
2	502	0	0	42	8	159660.80	3	1	0	113931.57
3	699	0	0	39	1	0.00	2	0	0	93826.63
4	850	2	0	43	2	125510.82	1	1	1	79084.10



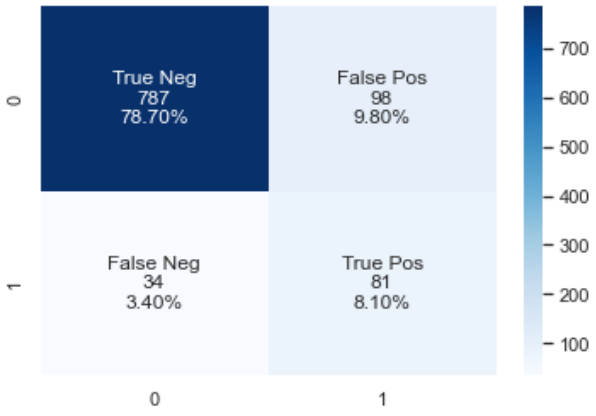
Logistic Regression



Decision Tree



SVM



Random Forest

Logistic Regression :

```
[INFO] Logistic Regression classifier:

              precision    recall  f1-score   support

           0       0.82      0.98      0.89      6382
           1       0.69      0.15      0.25      1618

    accuracy                           0.81      8000
   macro avg       0.75      0.57      0.57      8000
weighted avg       0.79      0.81      0.76      8000
```

Model Precision and re-Call SGD:

```
[INFO] SGD classifier:

              precision    recall  f1-score   support

           0       0.83      0.96      0.89      6382
           1       0.59      0.23      0.33      1618

    accuracy                           0.81      8000
   macro avg       0.71      0.60      0.61      8000
weighted avg       0.78      0.81      0.78      8000
```

Random Forest Model Precision:

```
from sklearn.metrics import classification_report

print('Random Forest:\n')

print(classification_report(y_test,pred))
```

```
[INFO] Random Forest classifier:

              precision    recall  f1-score   support

           0       0.90      0.98      0.94      6382
           1       0.88      0.57      0.69      1618

    accuracy                           0.90      8000
   macro avg       0.89      0.77      0.81      8000
weighted avg       0.90      0.90      0.89      8000
```

Decision Tree Model:

Logistic Regression:

Logistic regression is a classification algorithm often used as baseline model to set a benchmark. It suits well where our label is binary and categorical. LR uses predictive analysis to describe the tradeoff or relationship between a dependent binary variable and a set of independent variables. One drawback is that it doesn't handle collinearity and requires a large sample. It doesn't need the data to be linear in nature, it handles non linear relationships with the use of non linear log loss transformations. Here we get an accuracy of approximately 80.20%.

Code:

```
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import cross_val_score

clf = LogisticRegression()    # L2 regularization is applied by default
clf.fit(X_train, y_train)

pred = clf.predict(X_test)

accuracy = accuracy_score(pred, y_test)
print("Accuracy:", '{:.2%}'.format(accuracy))

cf_matrix = confusion_matrix(pred, y_test)

group_names = ['True Neg','False Pos','False Neg','True Pos']
group_counts = ["{0:0.0f}".format(value) for value in
            cf_matrix.flatten()]
group_percentages = ['{0:.2%}'.format(value) for value in
                cf_matrix.flatten()/np.sum(cf_matrix)]
labels = [f'{v1}\n{v2}\n{v3}' for v1, v2, v3 in
```

```
        zip(group_names,group_counts,group_percentages)]
labels = np.asarray(labels).reshape(2,2)
sns.heatmap(cf_matrix, annot = labels, fmt = '', cmap = 'Blues')
```

Output:

Accuracy: 80.20%

Decision Tree:

The basic structure of the decision tree consists of internal nodes and leaf nodes, where internal nodes checks certain conditions and splitting points and creates branches to reduce entropy. Leaf nodes that represent label values in our case: Exited status as 0 or 1. It supports categorical as well as continuous data.

Result suggests accuracy:81.20%

Code:

```
from sklearn import tree

clf = tree.DecisionTreeClassifier()
clf.fit(X_train, y_train)

pred = clf.predict(X_test)
accuracy = accuracy_score(pred, y_test)
print("Accuracy:", '{:.2%}'.format(accuracy))

cf_matrix = confusion_matrix(pred, y_test)
```

```
group_names = ['True Neg','False Pos','False Neg','True Pos']
group_counts = ["{0:0.0f}".format(value) for value in
                cf_matrix.flatten()]
group_percentages = ['{0:.2%}'.format(value) for value in
                     cf_matrix.flatten()/np.sum(cf_matrix)]
labels = [f'{v1}\n{v2}\n{v3}' for v1, v2, v3 in
          zip(group_names,group_counts,group_percentages)]
labels = np.asarray(labels).reshape(2,2)
sns.heatmap(cf_matrix, annot = labels, fmt = '', cmap = 'Blues')
```

Output:

Accuracy: 81.20%

Support Vector Machine:

Support Vector Machine is a supervised learning technique and it is a discriminative classifier and is formally defined by the hyperplanes. It projects data into higher dimentions to separate them and form boundaries using support vectors and maximizing the distance between them. We used SVM for our bank dataset .

Result suggests Acuuracy:82.10%

Code:

```
from sklearn import svm

# support vector classifier using linear hyper plane

clf = svm.SVC(C=1.0, kernel = 'rbf', probability = True, random_state = 124)

clf.fit(X_train, y_train)

pred = clf.predict(X_test)

accuracy = accuracy_score(pred, y_test)

print("Accuracy:", '{:.2%}'.format(accuracy))

cf_matrix = confusion_matrix(pred, y_test)

group_names = ['True Neg','False Pos','False Neg','True Pos']

group_counts = ["{0:0.0f}".format(value) for value in

            cf_matrix.flatten()]

group_percentages = ['{0:.2%}'.format(value) for value in

                cf_matrix.flatten()/np.sum(cf_matrix)]

labels = [f'{v1}\n{v2}\n{v3}' for v1, v2, v3 in

        zip(group_names,group_counts,group_percentages)]

labels = np.asarray(labels).reshape(2,2)

sns.heatmap(cf_matrix, annot = labels, fmt = '', cmap = 'Blues')
```

Output:

Accuracy: 82.10%

Random Forest:

As the name suggests, Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset**.** Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output.

The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.

Code:

```
from sklearn.ensemble import RandomForestClassifier

clf = RandomForestClassifier(n_estimators = 200, random_state=200)

clf.fit(X_train, y_train)

pred = clf.predict(X_test)

accuracy = accuracy_score(pred, y_test)

print("Accuracy:", '{:.2%}'.format(accuracy))

cf_matrix = confusion_matrix(pred, y_test)

group_names = ['True Neg','False Pos','False Neg','True Pos']

group_counts = ["{0:0.0f}".format(value) for value in

            cf_matrix.flatten()]

group_percentages = ['{0:.2%}'.format(value) for value in

                cf_matrix.flatten()/np.sum(cf_matrix)]

labels = [f'{v1}\n{v2}\n{v3}' for v1, v2, v3 in

        zip(group_names,group_counts,group_percentages)]

labels = np.asarray(labels).reshape(2,2)
```

```
sns.heatmap(cf_matrix, annot = labels, fmt = '', cmap = 'Blues')
```

Output:

Accuracy: 86.80%

CONCLUSION

In conclusion, the overall best performing model is Random Forest and SVM are suggests to solve this bank customer churn prediction problem. This is due to its overall performance for all the evaluation measurements and computational efficiency. For future research in customer churn analysis, methods that yield high interpretability and high predictability would be interesting to study. To be able to combine the knowledge of if a customer ends their engagement, as well as un derstanding why. Taken together, this could not only prevent customer churn but also help outlin e which variables affect customer churn and the bank could use this information to improve their business and customer relations. Moreover, interesting future research would be to analyse at which dataset size leave-one-out cross-validation and k-Fold cross-validation yields the same result.

REFERENCES

https://www.sciencedirect.com/science/article/abs/pii/S0957417408004326

https://edu.medium.com/prediction-of-customer-churn- 5a456c184ed1

https://www.geeksforgeeks.org/python-customer-churn-analysis-prediction/