Contents

[Introduction 1](#_Toc149570026)

[Dataset summary 1](#_Toc149570027)

[Business Description 2](#_Toc149570028)

[Hypothesis 2](#_Toc149570029)

[General Goal 2](#_Toc149570030)

[Success Criteria/Indicators 2](#_Toc149570031)

[Technology Used 3](#_Toc149570032)

[Machine Learning algorithms 3](#_Toc149570033)

[Libraries 3](#_Toc149570034)

[Business Understanding 4](#_Toc149570035)

[Data understanding 4](#_Toc149570036)

[Descriptive Statistics 5](#_Toc149570037)

[Data Preparation 6](#_Toc149570038)

[Data Cleaning 6](#_Toc149570039)

[Exploratory Data Analysis 7](#_Toc149570040)

[Introduction to Machine Learning Models 18](#_Toc149570041)

[Modeling 19](#_Toc149570042)

[Machine Learning models to be analysed. 19](#_Toc149570043)

[Hyperparameters Results for SVM and Random Forest 20](#_Toc149570044)

[Grid Search to Find Optimal Hyperparameters (KERNEL='RBF') 20](#_Toc149570045)

[Grid Search to Find Optimal Hyperparameters (KERNEL='Linear') 21](#_Toc149570046)

[Performance Grid Search using RBF Kernel Vs Linear Kernel. 22](#_Toc149570047)

[Grid Search to Find Optimal Hyperparameters 23](#_Toc149570048)

[Classification Report 23](#_Toc149570049)

[Features Importances 24](#_Toc149570050)

[Feature Importance Conclusion 25](#_Toc149570051)

[Checking Models and Algorithms 26](#_Toc149570052)

[Model Tuning and grid search 28](#_Toc149570053)

[Variable intuition/feature importance 29](#_Toc149570054)

[Model Comparisons 30](#_Toc149570055)

[Test Score Comparison 30](#_Toc149570056)

[Cross Validation Score Comparison 31](#_Toc149570057)

[AUC-ROC Curve Score Comparison 32](#_Toc149570058)

[Challenges encountered. 33](#_Toc149570059)

[References 34](#_Toc149570060)

# Introduction

The tendency of customers to switch to another competitor or stop using a service is the main and major challenge faced by businesses across the industries, including the banking sector.  
The ability to predict whether a customer will exit or not the bank can help banking institutions to take measures to retain them by reducing the costs of acquiring new clients and improving the customer service and satisfaction.

For this purpose, we have decided to work with a dataset called Bank Churners that was taken from Kaggle, which contains private and sensitive information about customers of an anonymous and unknown a bank and their transaction history. The dataset includes the behaviour of the clients in relation to bank transaction history, as well as, demographic features, such as age, gender, marital status, income level, education level and so on.

Our objective is to build a predictive machine learning model that can accurately classify customers either churn or not churn based on these features, among other techniques we will apply.

In summary, the idea of this project is to provide insights and recommendations to help the banks to reduce customer churn, increase customer retention, and improve their overall performance.

## 

## Dataset summary

This dataset contains more than 10,000 observations and 23 attributes, with the attrition column indication the Existing Costumers and Attrited Customers.  
Specifically, there are 8,500 customers who will stay while 1,627 customers have decided to leave the bank.

There are key features that are relevant to the problem of employee attrition:

Attrition\_Flag: Distinguishes between Existing Customers and Attrited Customers. This key feature will be the target variable for attrition prediction.  
  
Customer\_Age: Provides the age of the customer, offering insights into the age demographics of employees.

Dependent\_count: Represents the number of dependants the employee has. This information can be relevant to understand the family size of employees. For example, if an employee is married and has children, or other family members who rely on their income.

Education\_Level: Reflects the educational background of the employee, a factor linked to career decisions.

Marital\_Status: Highlights the marital status of the employee, a personal aspect that may impact attrition.

Income\_Category: Categorizes the income level of the employee, a crucial factor for bank institutions.

Months\_on\_book: Specifies the duration of the employee's association with the bank, contributing to attrition analysis.

# Business Description

Our project is focused on the banking industry, with the main goal of improving and retaining the customers to reduce churn rates. To achieve this objective, we will train, test, and develop Machine Learning models that predict whether a customer is likely to churn or not, based on his/her demographic features and transaction history.

## Hypothesis

As a starting point, we would like to investigate and analyse why the customers, who hold bank accounts, will churn or not from the banks based on his/her demographic features and transaction history. This will enable banking institutions to take proactive measures to retain their customers, improve customer satisfaction, and reduce the costs of new clients. By applying Machine Learning models on the Bank Churners dataset, we will try to accurately predict whether a customer will exit or not based on their banking data.

## General Goal

Our general goal is to develop a predictive Machine Learning model which can accurately classify customers as either churn or not based on their demographic features and transaction history. By achieving this goal, we aim to provide banking institutions with useful findings and recommendations to reduce customer churn rates and improve overall performance.

## Success Criteria/Indicators

The success of our project will be measured by the accuracy of our Machine Learning Models in predicting customer churn.  
We will use metrics such as accuracy, precision, recall and confusion matrix to evaluate the performance across the seven Machine Learning Models. Additionally, GridSearchCV will be applied to find the Hyperparameters and Cross Validation will ensure the authenticity of the modelling results.

# Technology Used

## Machine Learning algorithms

For our project we have implemented seven Machine Learning for classification models to experiment with multiple algorithms to find the most suitable and accurate model:

1. Decision Tree
2. Support Vector Machine
3. Logistic Regression
4. AdaBoost
5. Random Forest
6. Gaussian Naïve Bayes
7. KNeighbours

We will compare them and analyse the outcomes for all of them. Moreover, we used hyperparameters such as Kernel RBF and Linear applied to Support Vector Machine and applied cross-validation and confusion matrices to evaluate the performance of the models. Additionally, we have also performed the Hyperparameter Tuning applied for Random Forest model by specifying the number of folds for ‘k-fold’ and adjusting the parameters to plot the accuracies of ‘max\_depth’, then we used the classification report to assess the model’s performance.

## Libraries

Different libraries have been used for the purpose of performing the analysis of the dataset which is being implemented in Jupyter Notebook.

The following libraries are crucial for data analysis:

* Warnings will supress the errors that would normally be displayed.
* Scikit-learn for machine learning tasks.
* Seaborn and Matplotlib for visualizations.
* Pandas
* NumPy
* Encoders to encode categorical variables.
* data\_profiling which provides a way to quickly generates an overview of the dataset.
* Imbelear.over\_sampling.SMOTE for oversampling imbalanced datasets
* Missigno for analysing missing data.

## Business Understanding

Customer churn or attrition is a serious problem for banking industry and many other businesses such as telecommunications industry and hospital industry. Interestingly, according to Swetha Amaresan’s research (2021) on the matter has described that it costs more to acquire new customers than it does to retain existing customers. In fact, an increase in customer retention of just 5% can create at least a 25% increase in profit.

Understanding the reasons why customers churn is critical for businesses as it allows bank or financial institutions to take proactive measures to retain their customers.

With that being said, we will implement Machine Learning models to help the industry reduce customer churn, identify patterns, and predict which customer is likely to churn.

## Data understanding

In this stage, we will develop a deep understanding of the data we have using various techniques.

By Importing ProfileReport, it generates a detailed report that summarised the statistical measures and visualizations of the data and it allows a quick overview of the data we are dealing with.

Graphical user interface, application

Description automatically generated

After performing .head() and .tail() we can see that the dataset contains 23 columns and 10,127 observations (0 to 10,126). Moreover, it can be observed that all null values from the dataset are marked as ‘Unknown’, so we have decided to keep them for our future analysis.

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

## 

# Descriptive Statistics

# Data Preparation

We will clean the data to prepare for modelling.

Importing the library ‘Missigno’ it is more visual to analyse whether there are missing values or not. In this case, there are no missing or null values in the dataset.

A graph with text on it

Description automatically generated with medium confidence

## Data Cleaning

The 'clientum' column and both columns containing 'Naive\_Bayes\_Classifier' will be removed, as these columns are not useful for modeling and are dropped from the dataset.

# Exploratory Data Analysis

In this phase we need to prepare the data to perform a deep analysis through Exploratory Data Analysis (EDA).

It is clear that the number of Existing Customers is much higher than the number of Attrited Customers in the target variable, which means that we are dealing with class imbalance.

With that being said, Tara Boyle (2019) has stated that most machine learning algorithms work best when the number of samples in each class are about equal. This is because most algorithms are designed to maximize accuracy and reduce error.

We will address imbalanced data before applying our Machine Learning models chosen.

A screenshot of a graph

Description automatically generatedA pie chart with a number of customers

Description automatically generated

*Distribution of the Categorical Variables:*

Analysing the distribution of categorical variables can help the company identify predominant categories, which can inform the management decisions related to marketing strategies and client segmentation.

A graph with blue bars

Description automatically generated  
A graph with blue bars

Description automatically generated

Even though there is not a significantly difference, it seems that older customers are more likely to churn from the bank.

A graph showing different colored squares

Description automatically generated

*Marital Status*

Given the Marital Status, the customers married, and single ones are the majority in this analysis. Therefore, we could recommend offering special promotions and incentives to these groups.

A graph of different colored squares

Description automatically generated with medium confidence

*Gender Distribution*

The gender distribution between males and females are quite similar For Existing Customers and those who have left the bank.

A graph of a bar chart

Description automatically generated with medium confidence

*Income level*

The majority of the customers have an income less than $40k thus, it is clear that customers who earn less are likely to qualify for basic credit card (Blue). But on the other hand, the more a customer earns, the more likely they are to qualify for premium credit cards with higher credit limits.



*Education Level*

Graduates and High School students are the main customers; therefore, it would be a good strategy to offer them special discounts or credit cards without interest to incentive them to keep operating with the bank in order to retain them.

A graph of a graph showing different levels of education

Description automatically generated with medium confidence

*Months on Book*

The bank is successfully retaining the majority of its customer over a period of 35 months, either for Existing Customers or for Churned Customers on their books. Thus, the bank may offer them competitive interest rates and fees to keep them for a longer period of time.

A comparison of a graph

Description automatically generated

*Comparing Gender Vs Income Level*

On one hand, it is clear that there is a gender gap between Males and Females, where there are approximately 700 males who earn +$120K, around 1500 males who earn between 80K – 120K and roughly 1400 males earning about 60K – 80K. On the other hand, the majority of females earn less than $40K.

A graph of different colored bars

Description automatically generated

*Services Customers use.*

The variable Total Relationship Count is related to the number of products held by the customers. The majority of the customers have three services with the bank, followed by the same number of customers between four and six services. Customers with a higher relationship count are more likely to stay, this is a positive indicator for the bank.

A graph of a bar graph

Description automatically generated

*Months Inactive*

Churned Customers and Existing Customers tend to have periods of longer inactivity, most of them lasting three months, suggesting that prolonged inactivity could be a good indicator of customer attrition.

A graph of a bar chart

Description automatically generated with medium confidence

*Credit Offered to Customers*

The majority of the customers either existing or churned have a credit limit ranging from $2000 to $4000.

A graph of credit limit

Description automatically generated

The scatterplot clearly shows that for Existing Customers, there are three distinct clusters when comparing the total transaction counts and total transaction amount. The largest cluster has total transaction amount between 2500 and 5000 and is the most densely populated. The second cluster has total transaction amount around 8000. The third indicates the highest transactions amount, with total counts between 100 and 200. Conversely, Churned Customers clearly indicate that there is one cluster that has total transaction amount around 3000 and total transaction counts between 40 and 50 suggesting that the bank activity of the customers who want to leave may not be enough to keep them engaged with the bank.

A close-up of a graph

Description automatically generated

*Customers are using a percentage of their credit limit.*

There are a significant number of customers who are using their credit limit raging from 0% to 5%, it suggests that customers are using a very small portion of their credit limit.

A graph of a graph

Description automatically generated with medium confidence

*Correlations among the Features*

Given the fact that the Attrited Customers are only 16% of the sample, from our point of view is a good practice to analyse the customers as a whole and not only the ones who have left the bank, as we believe the customers who remain in the bank are likely to churn in the future as well. Therefore, we have decided to perform a correlated heatmap to analyse the features accurately and identify potential patterns or trends in the data.

The correlated heatmap depicts features high positive correlated (when two variables tend to increase or decrease together) between total ‘Customer\_ Age’ and ‘months\_on\_book’. This suggests that as a customer’s age increases, the number of months in the bank also tends to increase. Other features to be analysed are ‘Total\_Trans\_Amt and ‘Total\_Trans\_Ct. as this suggests that as the number of transactions a customer makes increases, the total transaction amount also tends to increase.

On the other hand, there are features negatively correlated (when one variable increases the other decreases). By analysing these correlations among ‘Avg\_Utilization\_Ratio’, ‘Avg\_Open\_To\_Buy’ and ‘Credit\_Limit’ indicate that as the ‘Avg\_Utilization\_Ratio’ increases, both ‘Avg\_Open\_To\_Buy’ and ‘Credit\_Limit’ decrease. In other words, customers who use their credit regularly usually tend to have lower credit limits.

A graph with red and blue squares

Description automatically generated

*Visualisation of the Positive correlations*

A close-up of a map

Description automatically generated

*Visualisation of the Negative correlation*

A graph showing the amount of attrition

Description automatically generated

# Introduction to Machine Learning Models

The initial question we need to address is how we can improve, reduce errors, and avoid bias when applying Machine Learning Model?

There are steps that we have implemented to improve the accuracy of our models:

1. Encoding Categorical variables into numerical variables.

We have selected the categorical features to transform them into numerical variables using the Label Encoder. Specifically, we have inverted the Labels for the ‘Attrition Flag’ feature to make it readable. 0 for Existing customers and 1 for Churned Customers

1. Data normalization.

As we do not have negative values in our dataset, we rescaled the continuous variables applying MinMaxScaler in our data to a range between 0 and 1. Particularly, this method is useful to see all the variables from the same lens (same scale), in this way we will bring all values into the range [0,1].

When we Encode Categorical Data, we turned string variables into numerical variables, when we did that, we do not have to scale or normalized that data, it is not recommended to normalize or scale them because they are no longer continuous variables.

1. We need to separate and define the dataset into X (input features) and y (target variable) and then split the data into independent and dependent variable.

The main variable for predicting a customer churned or not is the target variable (dependant variable) ‘Attrition\_Flag’, which is a binary classification. Then the model evaluates ‘y’ depending on the banking data from the customers, such as age, marital status, income, education level and bank transaction history.

## Modeling

In this phase, we have built and trained seven different Machine Learning models splitting the data into the training set (70%) and the testing set (30%).

In other words, we have ensured that we scaled the data after splitting. Therefore, we could scale the train and test set separately to prevent data leakage (Jason Brownlee, 2020)

# Machine Learning models to be analysed.

We have chosen seven Machine Learning models to analyse:

1. Decision Tree
2. Support Vector Machine
3. Logistic Regression
4. AdaBoost
5. Random Forest
6. Gaussian Naïve Bayes
7. KNeighbours

The decision for choosing those models was because they can be used for binary classification.  
In this phase, we have built and trained seven different Machine Learning models splitting the data into the training set (70%) and the testing set (30%).

Feature importance technique is determined by using the Decision Tree, Random Forest and Adaboost to choose the importance features. It basically uses a trained supervised classifier to select features. The higher the score, higher is the importance of that attribute.

We also applied Yellow Brick Classifier Metrics to understand the performance of our machine learning models:

* Confusion Matrix
* Classification Report
* Class Predictor Error
* ROC-AUC is a metric will tell us about the capability of model in distinguishing the classes. Higher the AUC, better the model
* Class Balance Visual

GridSearchCV is applied to find the optimal hyperparameters in this study. Cross validation is used to provide the authenticity of the modelling results by using Confusion Matrix, Cross validation and Hyperparameters.

# Hyperparameters Results for SVM and Random Forest

## Grid Search to Find Optimal Hyperparameters (KERNEL='RBF')

A screenshot of a computer

Description automatically generated

The best test score is 91,65% corresponding to hyperparameters {'C': 1000, 'gamma': 0.01}

A graph of a number of data

Description automatically generated with medium confidence

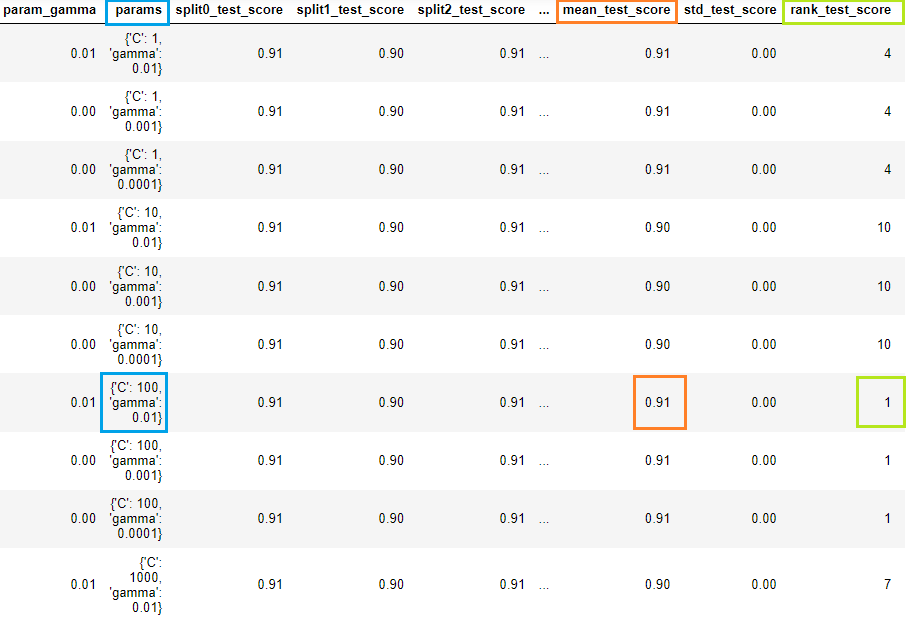
Building and Evaluating the Final Model

* Confusion Matrix  
  [[2543 0]

[487 9]]

* Accuracy 83.97%
* Precision 83.93%
* Sensitivity/Recall 100%

## Grid Search to Find Optimal Hyperparameters (KERNEL='Linear')



The best test score is 90.55% corresponding to hyperparameters {'C': 100, 'gamma': 0.01}

A graph of a graph

Description automatically generated with medium confidence

Building and Evaluating the Final Model

* Confusion Matrix  
  [[2474 69]  
   [ 229 267]]
* Accuracy 90.19%
* Precision 91.53%
* Recall 97.29%

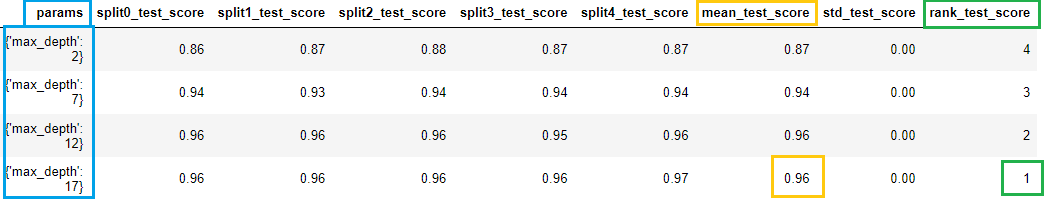
## Performance Grid Search using RBF Kernel Vs Linear Kernel.

Based on the results provided by RBF and Linear Kernel Model, it appears that has a relatively good fit, indicating that the model is able to generalize reasonably well to new, unseen data, without overfitting the training data.

In terms of evaluating the final model, the Linear Kernel Model has a higher accuracy and precision compared to the RBF Kernel Model. However, The RFB kernel model has a perfect sensitive/recall, meaning it correctly identified all positive cases.

Hyperparameter Tuning Max Depth

We have performed the Hyperparameter Tuning applied for Random Forest model by specifying the number of folds (5), which means that the dataset will be split into 5 and also, we adjusted the parameters to plot the accuracy of ‘max\_depth’ in a range of (2, 20, 5), then we used the classification report to assess the model’s performance.

Chart, line chart

Description automatically generated

After fitting the model with the values of ‘max\_depth’, the above plot indicates that the test accuracy 96% is high when ‘max\_depth’ is set in a range of 2 to 20 while splitting the model into 5.

## Grid Search to Find Optimal Hyperparameters

We can now find the optimal hyperparameters using GridSearchCV.

We can get accuracy of 90.32% using {'max\_depth': 10, 'max\_features': 5, 'min\_samples\_leaf': 100, 'min\_samples\_split': 200, 'n\_estimators': 300}

## Classification Report

The outcomes based on the model that was trained using optimal hyperparameters found that through the GridSearchCV, we achieved accuracy of 90% which is relatively high. Additionally, the precision and recall for Existing Customers were both high, indicating that the model is able to correctly identify positive cases with high accuracy. However, the recall for Churned Customers was lower than the recall for Existing Customers, indicating that the model may have more difficulty correctly identifying negative cases.

A screenshot of a computer screen

Description automatically generated

# Features Importances

Feature importance technique is determined by using the Decision Tree, Random Forest and Adaboost to choose the importance features. It basically uses a trained supervised classifier to select features. The higher the score, higher is the importance of that attribute.   
  
Decision Tree Model: Total Transaction Count seems to be the most important feature; it means that this feature contributes 41% to the overall model, followed by Total Transaction Amount (13%) and Total Revolving Balance (11%).

A graph with a bar chart

Description automatically generated

Random Forest Model: Total Transaction Count seems to be the most important feature; it means that this feature contributes 30% to the overall model, followed by Total Transaction Amount (21%) and Total amount change from the fourth quarter (Q4) to the first quarter (Q1) (14.5%).  
  
A graph with a bar graph

Description automatically generated

AdaBoost Model: Total Transaction Count seems to be the most important feature; it means that this feature contributes 30% to the overall model, followed by Total Revolving Balance (19%) and 12 Months Inactive (15.5%).

A graph with green bars

Description automatically generated

## Feature Importance Conclusion

Total Transaction Count stands out across all the models analysed, suggesting a strong correlation between transaction activity and the probability of a customer to churn. The Transaction Count might include deposits and withdrawals, purchases, transfers, and bill payments. In the context of churn analysis, it seems customers with lower Transaction Count are more likely to churn, based on the future importance in our models.

# Checking Models and Algorithms

Evaluation Metric **Accuracy**, which evaluates the correct predictions of the model.

After performing the *k-*fold cross validation, the comparison of the models is as follows:

A graph showing a comparison of a number of data

Description automatically generated with medium confidence

The accuracy is very high for all the models, considering that accuracy focuses on the overall Existing Customers case. However, we need to evaluate how well it predicts the Churned Customers case. Since the Random Forest model has the best accuracy (96%) among all the models, it is used to evaluate the test set.

The Confusion Matrix results are good, but still, 102 out of 496 Churned Customers are not caught. Therefore, we should focus on recall, a metric which minimises false negatives.

A blue squares with numbers

Description automatically generated

Evaluation Metric **recall** is selected, a metric which minimises false negative.

After performing the *k-*fold cross validation, the comparison of the models is as follows:

A graph showing a comparison of a model

Description automatically generated with medium confidence

AdaBoost has the best recall out of all the models (84.5%), it is used to evaluate the test set.

AdaBoost performs much better with 84 cases out of 496 cases of Churned Customers not caught. False positives are lower from 102 cases to 84 cases. However, there are still 84 Churned Customers cases in the test set which are not caught. This will be further taken care in the following section.

A screenshot of a computer

Description automatically generated

## Model Tuning and grid search

Since the Random Forest is the best accurate model out of all the models, a Grid Search is performed (on Resampled data training) for Random Forest model by varying number of estimators, criterion, and maximum depth.

Outcomes:

Best: 0.962229 using {'criterion': 'gini', 'max\_depth': 8, 'n\_estimators': 100}

The Recall results on the test set for Churned Customers are better (47 cases) comparing to our previous test sets and the overall model performance is reasonable. Therefore, the model is in line with the training set results.

A screenshot of a computer

Description automatically generated

## Variable intuition/feature importance

Total Transaction Count seems to be the most important feature; it means that this feature contributes 26% to the overall model, followed by Total Transaction Amount (23%) and Total Revolving Balance (12%).

A graph with red bars

Description automatically generated

# Model Comparisons

## Test Score Comparison

It is evident that Decision Tree shows the highest test score among all the models with a test score of 92.66%, followed closely by Support Vector Machine with a score of 90.58%. Logistic Regression also performs well with a test score of 85.55%. However, KNeighbors has the lowest test score among the models, standing at 74%.

Based on the test results a high test score indicates that these models are effective in making accurate predictions on new, unseen data.

A graph of blue bars with white text

Description automatically generated

## Cross Validation Score Comparison

The main purpose of cross validation is to prevent overfitting, which occurs when a model is trained too well on the training data and performs poorly on new, unseen data. By evaluating the model on multiple validation sets, cross validation provides a more realistic estimate of the model’s generalization performance, i.e., its ability to perform well on new, unseen data. (Sharma, 2017)

The following machine learning models were cross-validated using 10 folds, except for Support Vector Machine and AdaBoost, which of both were cross-validated using 5 folds due to computational expensive.

The results show that the Decision Tree has performed with a high cross-validation score of 92%, suggesting that the model is effective in preventing overfitting and performs well across different subsets of the data. Both Logistic Regression and Gaussian Naïve Bayes also demonstrate high cross-validation score of 87%. However, Support Vector Machine show the lowest cross-validation score among all the models, standing at 77%. This model could be sensitive to overfitting due to computational cost.

A graph with green bars

Description automatically generated

## AUC-ROC Curve Score Comparison

AUC - ROC curve is a performance measurement for the classification problems at various threshold settings. ROC is a probability curve and AUC represents the degree or measure of separability. It tells how much the model is capable of distinguishing between classes. Higher the AUC, the better the model is at predicting 0 classes (Existing Customers) as 0 and 1 classes (Churned Customers) as 1 (Narkhede, 2018).

It is clear that Support Vector Machine and Logistic Regression have an excellent performance in distinguishing between positive and negative classes. The AUC-ROC score of 92% for both models suggest a strong ability to rank instances correctly. This is followed closely by Adaboost, Random Forest and the others. However, Kneighbors has the lowest AUC-ROC score of 71% among the models.

A graph of different sizes of bars

Description automatically generated with medium confidence

# Challenges encountered.

In the data analysis process, we encountered a challenge related to the use of Ordinal Encoder, which encodes the features with ordinal numbers starting from 1 onwards. We came across this problem when we were working with Yellowbrick, a visualisation library, where the error message stated, ‘could not decode 1, 2’. 1 for Existing Customers and 2 for Churned customers. This error came up due to a mismatch between desired class labels and those contained in the target variable. Label Encoder ensured that classes are labelled correctly for Yellowbrick visualisations. For our project 0 is for Existing Customers and 1 is for Churned Customers.

To overcome this challenge, we had to change the Encoder from *Ordinal* to *Label*. Label encoding resolved the ambiguity in class labeling. This adjustment was crucial for the accurate interpretation and visualisation of the data.

# References

“Predicting Credit Card Customer Segmentation.” *Www.kaggle.com*, [www.kaggle.com/datasets/thedevastator/predicting-credit-card-customer-attrition-with-m](http://www.kaggle.com/datasets/thedevastator/predicting-credit-card-customer-attrition-with-m).

Swetha Amaresan. “What Is Customer Churn? [Definition].” *Hubspot.com*, Swetha Amaresan, 16 Feb. 2018, blog.hubspot.com/service/what-is-customer-churn.

Boyle, Tara. “Methods for Dealing with Imbalanced Data.” *Medium*, Towards Data Science, 3 Feb. 2019, towardsdatascience.com/methods-for-dealing-with-imbalanced-data-5b761be45a18.

Brownlee, Jason. “Data Leakage in Machine Learning.” *Machine Learning Mastery*, 15 Aug. 202AD, machinelearningmastery.com/data-leakage-machine-learning/.

Ingole, Mayur . “Data Science Process Framework (CRISP-DM).” *Www.linkedin.com*, 5 July 2022, www.linkedin.com/pulse/data-science-process-framework-crisp-dm-mayur-ingole/. Accessed 2 May 2023.

“Seaborn.boxplot — Seaborn 0.11.1 Documentation.” Seaborn.pydata.org,

seaborn.pydata.org/generated/seaborn.boxplot.html.

“Bar Charts and Pie Charts - Problem Solving with Python.” Problemsolvingwithpython.com, problemsolvingwithpython.com/06-Plotting-with-Matplotlib/06.06-Bar-Charts-and-Pie-Charts/. Accessed 30 Oct. 2023.

“Pandas.Series.value\_counts — Pandas 0.25.1 Documentation.” Pandas.pydata.org, pandas.pydata.org/pandas-docs/version/0.25.1/reference/api/pandas.Series.value\_counts.html#:~:text=With%20normalize%20set%20to%20True. Accessed 30 Oct. 2023.

“3.3. Metrics and Scoring: Quantifying the Quality of Predictions — Scikit-Learn 0.22.1 Documentation.” Scikit-Learn.org, scikit-learn.org/stable/modules/model\_evaluation.html.

scikit learn. “Sklearn.tree.DecisionTreeClassifier — Scikit-Learn 0.22.1 Documentation.” *Scikit-Learn.org*, 2019, scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html.

scikit-learn developers. “Sklearn.svm.SVC — Scikit-Learn 0.22 Documentation.” *Scikit-Learn.org*, 2019, scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html.

scikit-learn developers. “Sklearn.linear\_model.LogisticRegression — Scikit-Learn 0.21.2 Documentation.” *Scikit-Learn.org*, 2014, scikit-learn.org/stable/modules/generated/sklearn.linear\_model.LogisticRegression.html.

“Sklearn.ensemble.AdaBoostClassifier — Scikit-Learn 0.22.1 Documentation.” *Scikit-Learn.org*, scikit-learn.org/stable/modules/generated/sklearn.ensemble.AdaBoostClassifier.html.

Scikit-learn. “Sklearn.ensemble.RandomForestClassifier — Scikit-Learn 0.20.3 Documentation.” *Scikit-Learn.org*, 2018, scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html.

“Sklearn.naive\_bayes.GaussianNB — Scikit-Learn 0.22.1 Documentation.” *Scikit-Learn.org*, scikit-learn.org/stable/modules/generated/sklearn.naive\_bayes.GaussianNB.html.

---. “Sklearn.neighbors.KNeighborsClassifier — Scikit-Learn 0.22.1 Documentation.” *Scikit-Learn.org*, 2019, scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html.

“Sklearn.metrics.roc\_curve — Scikit-Learn 0.23.0 Documentation.” Scikit-Learn.org, scikit-learn.org/stable/modules/generated/sklearn.metrics.roc\_curve.html.

“ROCAUC — Yellowbrick V1.5 Documentation.” Www.scikit-Yb.org, www.scikit-yb.org/en/latest/api/classifier/rocauc.html.

“ML | Label Encoding of Datasets in Python.” GeeksforGeeks, 15 Oct. 2018, www.geeksforgeeks.org/ml-label-encoding-of-datasets-in-python/.

“Subplotting with Matplotlib and Seaborn.” DEV Community, dev.to/thalesbruno/subplotting-with-matplotlib-and-seaborn-5ei8.

Holtz, Yan. “Control Color in Seaborn Heatmaps.” The Python Graph Gallery, python-graph-gallery.com/92-control-color-in-seaborn-heatmaps/.

Narkhede, Sarang. “Understanding AUC - ROC Curve.” Medium, Towards Data Science, 26 June 2018, towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5.

Sharma, Abhishek. “Cross Validation in Machine Learning.” GeeksforGeeks, 21 Nov. 2017, www.geeksforgeeks.org/cross-validation-machine-learning/.