Reducing Customer Churn in Banking Industry by classifying Attrition using Clustering and other Classification Models.

Abstract: This project will represent the analysis made on a data set which represents the customer churn in a Banking Industry and means to classify customers based on certain info about them (demographic, banking habits, etc) using clustering and multiple classification algorithms and thus expressing which method is superior for the application and the use case along with observing the difference in their performance.

The methodologies used for classification are k-NN (k Nearest Neighbors) and Logistic Regression and for clustering, k-Means.

The data set being used is a .csv file which was procured from kaggle and has 23 features (columns) . Out of which there are 6 categorical data, 16 numeric data and our target variable here will be the Attrition_Flag.

In this report, we shall be applying Exploratory Data Analysis on this data set, trying to understand the data better specifically what each feature signify. We shall process the data by cleaning the data, and then apply feature engineering to find out the best feature that can help us solve our problem and then apply the machine learning algorithms mentioned along with giving a brief explanation on how these algorithms work for our data set and what conclusions can be drawn from them and then finally compare these models to better understand the problem and choose which solution works best.

Introduction: The aim of the project is to analyze a data set of bank customers and classify them based on their attrition flag – whether they are an “Attrited Customer” or an “Existing Customer” .

To understand the problem, we first need to understand “Churn” and “Attrition”, Churn and attrition are similar concepts used to describe customers who have quit using a service. The main distinction between the two terms is that attrition may also include customers who have decreased their engagement or activity with the bank but have not necessarily terminated their relationship, whereas churn typically refers to customers who have terminated their relationship with a bank.

For example, in the banking business, a customer who shuts their account is termed as a churned customer, where as a consumer who reduces their account activity, such as completing fewer transactions, may be labeled an attrited customer.

Attrition can have a significant impact on the effectiveness of retention campaigns for banks. A bank may see a drop in income and profitability as well as a loss of market share to rivals if it is unable to recognize and keep customers who are more likely to leave. A bank's retention effort may be impacted by attrition in a number of ways, including Decreased effectiveness of retention efforts, Reduced customer loyalty, Negative impact on brand image and Increased competition.

Machine learning classification algorithms can be used to solve the problem of classifying bank customers as either "Attrited" or "Existing". Machine learning models can use customer data to analyze patterns and links between various attributes and customer churn, and then use these associations to predict which customers are more likely to leave.

Banks may utilize machine learning algorithms to pinpoint the essential characteristics—like account balance, transaction history, age, and income—that are most indicative of client churn. Banks can create tailored retention strategies that are more likely to be successful by concentrating on these essential characteristics. Banks may also identify which customers are more likely to quit the bank by analyzing past customer data and machine learning models, and they can take proactive steps to keep those customers before they go. In order to improve the entire customer experience and lower customer turnover, banks can use machine learning algorithms to evaluate client interactions and find behavioral patterns linked to increased customer satisfaction and loyalty. Banks can lower the cost of recruiting new customers, which is typically more expensive than keeping existing ones, by utilizing machine learning to predict customer churn and enhance client retention.

Literature Review: In recent times, customer churn has become a significant issue for banks due to the increased number of service providers in the banking sector. To address this issue, this paper by Manas Rahman and V Kumar proposes a machine learning-based method to predict customer churn in a bank by analyzing customer behavior. The study employs KNN, SVM, Decision Tree, and Random Forest classifiers and uses feature selection methods to identify relevant features and verify system performance. The experimentation was conducted on a churn modeling dataset from Kaggle, and the results were compared to find an appropriate model with higher precision and predictability. The Random Forest model after oversampling was found to be better than other models in terms of accuracy.[1]

The paper by Prashant Verma proposes a machine learning-based method to predict customer churn in banks by analyzing customer behavior. The study employs KNN, SVM, Decision Tree, and Random Forest classifiers and uses feature selection methods to identify relevant features and verify system performance. The Random Forest model after oversampling was found to be better than other models in terms of accuracy.[2]

Study by Sahar F. Sabeh focuses on the use of machine-learning models in predicting customer churn and increasing customer retention rate in organizations through CRM systems. Ten different analytical techniques were compared, including Discriminant Analysis, Decision Trees, Support Vector Machines, and ensemble-based learning techniques. The models were applied to a dataset of telecommunication containing 3333 records, and the results showed that Random Forest and ADA Boosting outperformed all other techniques with almost the same accuracy of 96%. Multi-layer perceptron and Support Vector Machine were also recommended with 94% accuracy, while Decision Tree achieved 90%, naïve Bayesian 88%, and Logistic Regression and Linear Discriminant Analysis with 86.7% accuracy.[3]

Saran Kumar A and Saran Kumar A's paper reviews the most popular machine learning algorithms used for predicting churn, not only in the banking sector but also in other sectors that rely on customer participation.[4]

Paper by Ishpreet Kaur and Jasleen Kaur explores the use of various machine learning models such as Logistic Regression, Decision Tree, K-Nearest Neighbor, and Random Forest to predict the probability of customer churn in the banking industry. The paper presents a comparison of these models in terms of performance metrics such as accuracy and recall.[5]

Data Cleaning an EDA : The data is obtained from the zhyli. (2020). Prediction of Churning Credit Card Customers [Data set]. Zenodo.https://doi.org/10.5281/zenodo.4322342. Start by exploring the data to understand what each columns hold. On first glance by printing the head of the data frame, we can see that the data set has a combination of categorical and numerical data. Along with that we can observe that the Attrition_Flag displays the flags for the customers indicating churn. This is our target variable (Y) and we shall drop this column when we will be applying our machine learning models. The data set has features, "Naive_Bayes_Classifier_Attrition_Flag_Card_Category_Contacts_Count_12_mon_Dependent_count_Education_Level_Months_Inactive_12_mon_1" and "Naive_Bayes_Classifier_Attrition_Flag_Card_Category_Contacts_Count_12_mon_Dependent_count_Education_Level_Months_Inactive_12_mon_2". As per the source these columns are classification result for this problem using Naive Bayes Classification. As applying any classification methodology to the data set with these two features as a part of out selected features could lead to model giving is in accurate results, it will be best to drop them. Before beginning any exploratory analysis, it is best to check for missing data, Unknown vales and null or zero values. After applying the isnull().sum() function on our data frame, we can see that there are no null values present in any of our columns.

Feature Name	Description
CLIENTNUM	Unique identifier (Int)
Attrition_Flag	Flag indicating churn (Boolean)
Customer_Age	Age (Int)
Gender	Gender (String)
Dependent_count	Number of dependents (Int)
Education_level	Education level (String)
Marital_Status	Marital Status (String)
Income_Category	Income Category (String)
Card_Category	Type of card (String)
Months_on_book	Months on book (Int)
Total_Relationship_Count	No. of relationships with credit card provider (Int)
Months_Inactive_12_mon	Inactivity in 12 months (Int)
Contacts_Count_12_mon	Contacts in 12 months (Int)
Credit_Limit	Credit limit (Int)
Total_Revolving_Bal	Total Revolving balance (Int)
Avg_Open_To_Buy	Average open to buy ratio (Int)
Total_Amt_Chng_Q4_Q1	Total Amount change fro Q4 to Q1 (Int)
Total_Trans_Amt	Total Transaction Amount (Int)
Total_Trans_Ct	Total Transaction Count (Int)
Total_Ct_Chng_Q4_Q1	Total count change from Q4 to Q1 (Int)
Avg_Utilization_Ratio	Average utilization ratio (Int
NaiveBayes_Attrition_1_Classification	Naive Bayes classifier using one set of parameters (Int)
NaiveBaues_Attrition_2_Classification	Naive Bayes classifier using one set of parameters (Int)

Table 1: Feature Table

In addition to checking for null values, it also necessary to check for any duplicates in the data set by using the CLIENTNUM column in the data.

We can also check for Unknown and 0 values in the data set by using count() and unique() functions as per the requirement. Duplicates and other such discrepancy can cause some issues when we try to classify the churn. Missing values can cause the model to be biased and thus not perform well. Duplicated data will harm the test - train split as the duplicates can be split as well leading to more bias.

Continuing with our EDA, we can check the data split between Existing Customer and Attrited Customer using a pyplot.

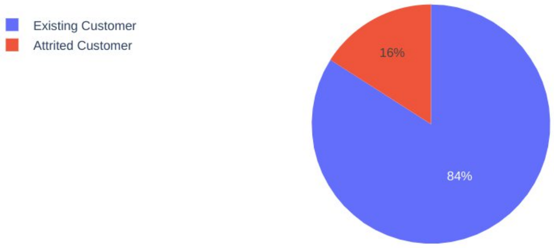


Figure 1: Pie Chart representing percentage split in data w.r.t Existing Customer and Attrited Customer.

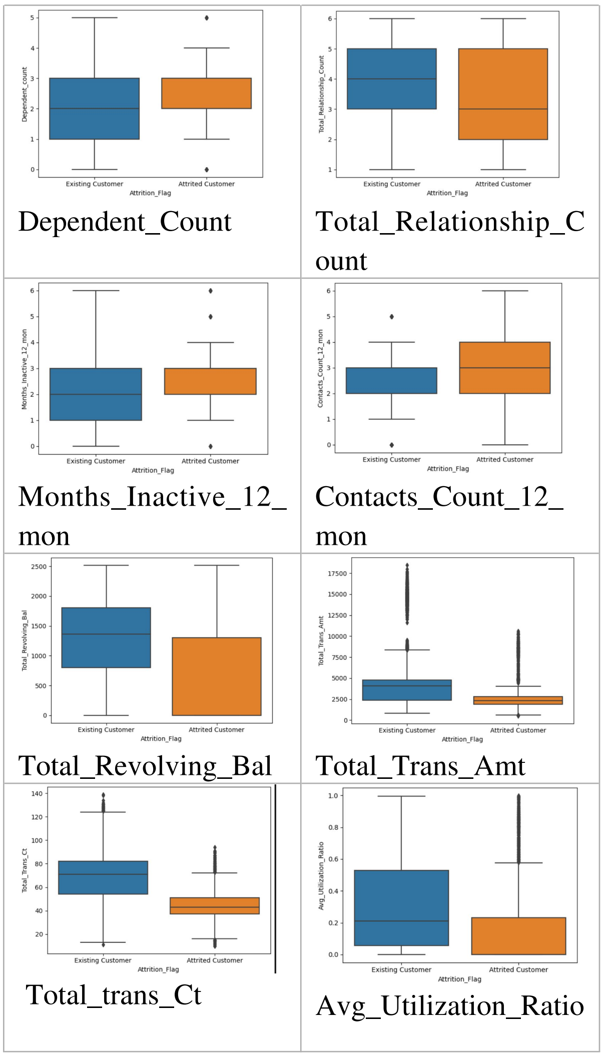
As you can see, the 84% of the customers in the data set are existing customers and 16% are Attrited Customers. We can print the count of these flags based on the client ID. As per the data, 1627 customers are attrited and 8500 are existing customers. We have some categorical data. By applying chi_2_contingency we can see the difference between the categories. This will help us determine if these categories are the features that can be considered for our model. The following table represents the chi2 values of our categorical data. Here we shall focus on the p-values as they can be considered as evidence of a statistically significant result. If p-values > 0,05, we can consider that the null hypothesis for that feature is True.

Feature	Chi_2	P_value	DOF
Gender	13.86	0.00019	1
Income	12.83	0.02500	5
Marital_Status	6.056	0.10891	3
Card_Category	2.234	0.52524	3
Education_Level	12.51	0.051489	6

Table 2: Chi Squared Statistics

After observing the table, we can deduce that only Gender and Income have a relationship with the Attrition Flag since their p-values are 0.00019 and 0.02500 respectively which are < 0.05.

Moving on to numeric features, we can check their significance using box plot.



Dependent_Count

Total_Relationship_Count

Months_Inactive_12_mon

Contacts_Count_12_mon

Total_Revolving_Bal

Total_Trans_Amt

Total_trans_Ct

Avg_Utilization_Ratio

Figure 2: Box Plot for Numeric Features

The above feature are the only ones that have some sort of relation with the Attrition. Here Blue plots represent Existing Customers– and the orange plots resemble Attrited Customers. The other features show no or very small difference in their plots signifying very very slight to no relationship with our target variable.

Feature Engineering: To ensure that all data is accounted for in respect to the attrition flag and to better understand the relationship with the attrition, categorical data should be converted into numerical data. For this purpose we can use the LabelEncoder() function from sklearn.preprocessing. Label Encoder essentially assigns each categorical value to an integer value based on alphabetical order. Thus we shall be left with new data frame, *df_main* which comprises of the numerical features and the converted features. We can verify if the operation was successful by printing the first 20 rows of the data frame. Since now our data is normalized, we can create a heatmap based on the correlation of each feature. We have used the seaborn library for this.

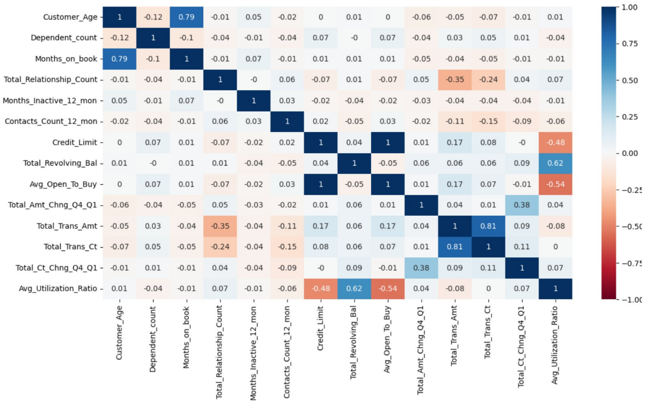


Figure 3: Correlation Matrix

Observing the correlation matrix we can conclude the following: 1) Total_Revolving_bal and Avg_Utilization_Ratio are positively correlated, 2) Avg_Utilization_Ratio and Avg_Open_To_Buy are negatively correlated. 3) Avg_Utilization_Ratio and Credit_Limit are negatively correlated.

Feature Selection: For the machine learning models to perform well, it needs best features to obtain the target variable To find this, we can consider the features as a hyperparameter. To perform hyperparameter tuninig, a combination of Random Forest, GridSearchCV and Stratified k-fold Cross Validation. This process aims to identify the optimal combination of hyperparameters that maximizes the F1 score, which is used as the evaluation metric. The params dictionary specifies the hyperparameters to be tuned, which are max_depth, min_samples_split, and min_samples_leaf. The s_kfold object is employed to partition the data into training and validation sets.

GridSearchCV is used to perform an exhaustive search over the specified hyperparameters, utilizing cross-validation to avoid overfitting. The n_jobs parameter is set to -1 to utilize all available CPU cores for faster processing. Once the best hyperparameters have been identified, the best_params_ and best_score_ attributes of the GridSearchCV object are used to display the best hyperparameters and corresponding F1 score.

Additionally, the feature_importances_ attribute of the best estimator is used to calculate the relative importance of each feature in predicting the target variable. These feature importances are then displayed using a Pandas Series. The following graph represents the feature importance.

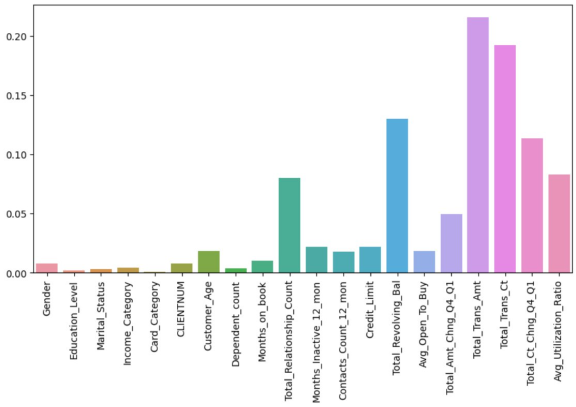


Figure 4: Feature importance based on F1 score.

After this process, we can clearly observe that there are 7 features that we can use to get the desired results. Total_Relationship_Count, Total_Revolving_Bal, Total_Amt_Chng_Q4_Q1, Total_Trans_Amt, Total_Trans_Ct, Total_Ct_Chng_Q4_Q1, Avg_Utilization_Ratio

As we now know which features are important, we can create a new data frame and apply StandardScalar() to apply scaling to get zero mean and unit variance. We then use fit_transform() to compute mean and standard deviation and perform standardization.

Machine Learning: The churn problem is a classic classification problem where just by classifying the customer based on their banking behavior we can classify them. Additionally we can group customers based on the features to check if there are some patterns which can help with the retention campaign.

Logistic Regression can be used as the baseline for this project as it is the most widely used classification model. The logistic regression model outputs a probability score between 0 and 1, which represents the predicted probability of a customer churning. A threshold value can be set to classify customers as churn or non-churn based on their predicted probability score. Logistic regression provides easy interpretability for the relationship between the independent variables (such as demographics, transaction history, etc.) and the dependent variable (churn). This makes it easy for decision-makers to understand the impact of each variable on churn and make targeted interventions to retain customers.

K-Nearest Neighbors is a non-parametric algorithm that makes no assumptions about the underlying distribution of the data. The algorithm simply stores the training data and uses it to classify new observations based on their proximity to the training data. K-NN can be used with any distance metric, such as Euclidean distance or cosine similarity, to measure the distance between observations.

K-NN is a non-parametric algorithm, meaning that it makes no assumptions about the underlying distribution of the data. This is useful when the data does not follow a specific distribution or when there are no clear functional relationships between the independent and dependent variables.

K-Means Clustering is a machine learning algorithm used to group similar data points in a given data set. In this context, it can be used to cluster customers based on their demographic information and banking habits, allowing for targeted marketing and customer retention strategies.

The K-Means algorithm starts by randomly selecting k centroids (or centers) from the data set and assigns each data point to the nearest centroid. It then recalculates the centroid based on the mean of all data points assigned to it, and repeats the process until convergence is achieved. The resulting clusters represent groups of data points that are similar to each other in some way, and can be further analyzed to identify patterns or characteristics of the customers. Using the elbow method we are able to find the right number of clusters for our model.

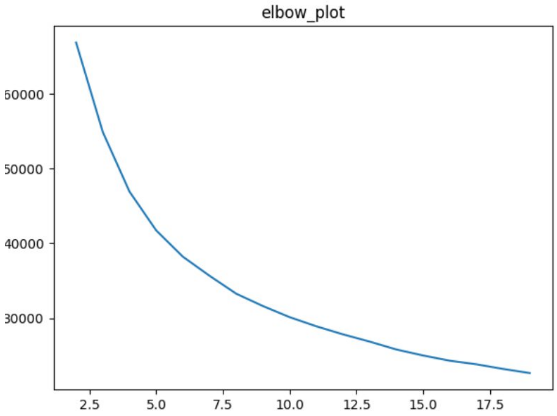


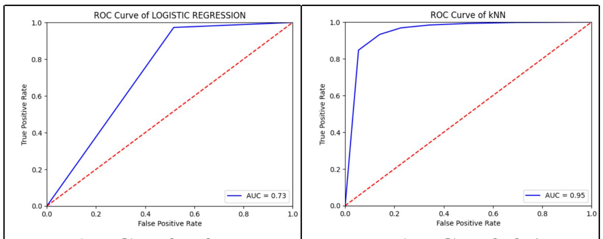
Figure 6: Best no. of clusters is 6.

Results: After applying the mentioned technique, the following results can be obtained.

	KNN	LR
Accuracy on Training set	94.98%	89.07&
Accuracy on Test set	93.7%	89.4%
Recall	82.4%	77.2%
Specificity	77.39%	48.40%

Apart from accuracy in the data set, there is also a need for measuring the Recall and Specificity score for the model. Recall is an important metric to consider in the customer churn prediction problem in the banking industry because it measures the proportion of actual churned customers that are correctly identified as such by the model. Misclassifying a churned customer as not churned (a false negative) can result in a lost opportunity to retain the customer. Specificity score measures the proportion of non-churned customers that are correctly identified as such by the model. Misclassifying a non-churned customer as churned (a false positive) can result in unnecessary and costly retention measures that may not be effective for that customer. As per the score table above, KNN has good recall score and specificity score and does well on both train and test set. Logistic regression tends to overfit on the test data. The Recall score is good but the model lags behind when tested for Specificity.

It is common for the number of customers who do not churn to be significantly higher than the number of customers who do churn. This means that there are fewer samples of the positive class (churned customers) compared to the negative class (non-churned customers), leading to imbalanced data. This could be why the specificity.for Logistic Regression is low. Different performance metrics like AUC-ROC curve can be used to better represent the models performance. The higher the AUC, the better the model performance.



LR: AUC = 0.73

KNN: AUC = 0.95

Figure 7: ROC Curve Graph

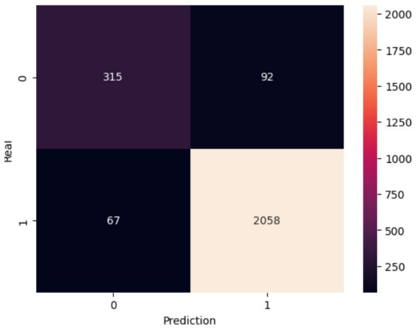


Figure 8: KNN Confusion Matrix

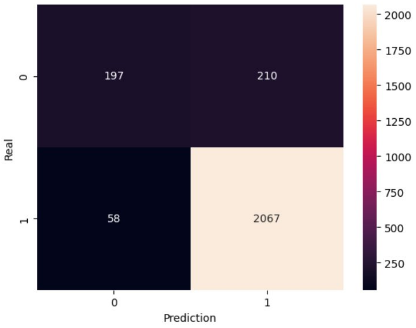


Figure 9: LR Confusion Matrix

For K Means, we us silhouette score to evaluate if the number of clusters is the right amount. The silhouette score obtained is 0,21 .

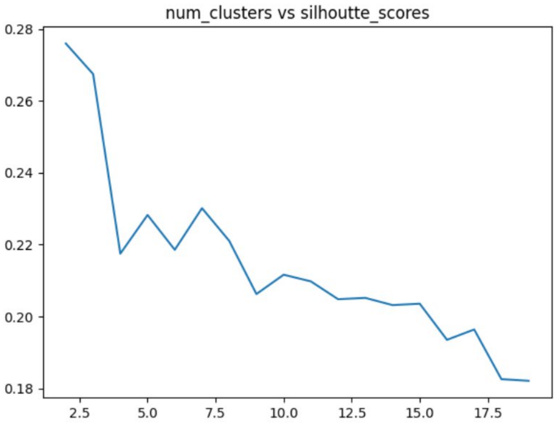


Figure 10: This plot displays how increasing the number of clusters will result in a low silhouette score.

The silhouette score lies between 1 and -1, thus the score of 0.21 depicts some overlap of clusters.

Conclusion: Based on the results obtained, it can be concluded that KNN outperforms logistic regression in predicting customer churn in the banking industry. KNN achieved higher accuracy scores on both the training and test sets, as well as higher recall and specificity scores, indicating that it is better at identifying customers who are likely to churn and customers who are not likely to churn, respectively. However, it is important to note that logistic regression achieved a lower specificity score compared to KNN, indicating that it has a higher false positive rate. This means that logistic regression may incorrectly predict more non-churned customers as churned, resulting in potentially unnecessary retention measures. If the bank wants to prioritize identifying as many customers who are likely to churn as possible while also minimizing the false positive rate, then KNN may be a better choice.

Using the K Means method, the bank can also look for patterns in the behavior of the customers and find additional ways to help with retention.

One possible future improvement for the project is to explore the use of ensemble methods such as Random Forest or Gradient Boosting, which may improve the performance of the models and provide more accurate predictions.

References:

[1]M. Rahman and V. Kumar, "Machine Learning Based Customer Churn Prediction In Banking," 2020 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, India, 2020, pp. 1196-1201, doi: 10.1109/ICECA49313.2020.9297529.

[2]Verma, P., 2020. Churn prediction for savings bank customers: A machine learning approach. Journal of Statistics Applications & Probability, *9*(3), pp.535-547.

[3]Sabbeh, S.F., 2018. Machine-learning techniques for customer retention: A comparative study. International Journal of advanced computer Science and applications, 9(2).

[4]Saran Kumar, A. and Chandrakala, D., 2016. A survey on customer churn prediction using machine learning techniques. International Journal of Computer Applications, 975, p.8887.

[5]I. Kaur and J. Kaur, "Customer Churn Analysis and Prediction in Banking Industry using Machine Learning," 2020 Sixth International Conference on Parallel, Distributed and Grid Computing (PDGC), Waknaghat, India, 2020, pp. 434-437, doi: 10.1109/PDGC50313.2020.9315761.