Churn prediction

Full name: Sintoris Nikolaos
Contact: nikolaossintoris@gmail.com

Contents

1 Goal

The Chief Retail Officer of the bank noticed a steady increase in the number of inactive users over the years which made her realize that the use of a data-driven solution is necessary to overcome the limitations of the current rule-based approach for monitoring user activity. Considering the nature of the data, you proposed to build a classification model that could accurately predict which customers are likely to turn inactive in the next 3 months.

2 Data

The data consists of demographic and behavioral variables. The description of each column in the dataset is provided below:

- CustomerID: unique customer identification number
- Geography: customers' registration region
- Gender: Male/Female
- AgeBand: the range of ages the customer belongs to
- TenureYears: tenure since the first bank account opening
- EstimatedIncome: estimated yearly income
- BalanceEuros: total financial assets (savings/deposits)
- NoProducts: number of total products the customer holds
- CreditCardholder: credit card ownership
- CustomerWithLoan: indicator of whether the customer has taken a loan (consumer/mortgage)
- DigitalTRXratio: digital over physical transactions ratio
- Inactive (target): last 3 months customer's activity (binary)

Our data consist of 10000 rows and 12 columns. We can see an overview of our raw data in Figure 1.

	CustomerID	Geography	Gender	Age_Band	TenureYears	EstimatedIncome	BalanceEuros	NoProducts	CreditCardholder	CustomerWithLoan	Digital_TRX_ratio	Inactive
0	5188208	Rest_GR	Male	18-25	0	40683.96	50086.2120	1	0	0	0.38	0
1	8683784	Thessaloniki	Female	65+	4	2429.51	0.0000	1	1	0	0.33	1
2	3512360	Athens	Male	45-55	4	41694.49	26852.7072	1	1	1	0.72	0
3	7104818	Rest_GR	Male	25-35	5	74523.33	90325.6200	1	0	0	0.08	0
4	6712745	Rest_GR	Female	25-35	9	111050.49	100537.0608	2	0	0	1.38	0
...	...	...	...	...	...	...	...	...	...	...	...	...
9995	5603293	Athens	Female	25-35	5	102771.03	0.0000	2	0	0	0.72	0
9996	6827613	Rest_GR	Male	35-45	8	143715.62	82870.9002	1	0	0	0.69	1
9997	5226960	Rest_GR	Male	35-45	1	35812.84	65007.1752	1	1	0	0.89	1
9998	280950	Athens	Male	35-45	1	99667.13	82639.8672	1	1	1	0.94	0
9999	1831292	Athens	Male	35-45	5	76783.59	84390.5040	1	1	0	0.61	0

Figure 1: Overview of raw data.

3 Exploratory Data analysis (EDA)

Our dataset consists of 3 categorical and 9 numerical features.

Also, we have for each customer only one registration, which makes our work easier as we do not have to handle for the same customers multiple registrations.

Another encouraging thing about our dataset is that we do not have missing values (Figure 2).

	No. Missing Values
CustomerID	0
Geography	0
Gender	0
Age_Band	0
TenureYears	0
EstimatedIncome	0
BalanceEuros	0
NoProducts	0
CreditCardholder	0
CustomerWithLoan	0
Digital_TRX_ratio	0
Inactive	0

Figure 2: Number of missing values per feature.

3.1 Distribution Analysis

Distribution analysis is a fundamental technique in data analysis, crucial for understanding the underlying patterns and characteristics of each feature within a dataset. By employing histograms for every feature, we can visually inspect the frequency distribution and identify essential aspects such as skewness, kurtosis, and the presence

of outliers. This approach not only helps in diagnosing potential issues with data quality and distribution but also provides valuable insights that guide further data preprocessing steps and feature engineering. Understanding the distribution of data is vital for selecting appropriate statistical models, improving the accuracy of predictive algorithms, and ensuring robust and reliable analytical outcomes.

At Figure 3 we can see a plot of histograms for each feature to find out and understand their distribution. After careful examination of the distribution plot, we concluded to the below:

- CustomerID: appears to have a uniform distribution which is expected because we have unique identifiers for each customer.
- TenureYears: distribution indicates a fairly even spread across the years, with slightly more customers in the higher tenure brackets
- EstimatedIncome: appears to have a uniform distribution, that indicates that customer have a wide range of yearly income without any specific concentration in particular ranges
- BalanceEuros: in general, appears to have a normal distribution except a peak around the mid-range values, with a notable number of customers having zero balance. This could indicate that a significant portion of customers might not be actively using their accounts.
- NoProducts: appears to have a right skewed distribution. That means that most of the customers have 1 or 2 products with very few customers having more than 2 products.
- CreditCardholder: most of the customers hold a credit card.
- CustomerWithLoan: the number of customers with loan, are slightly greater than the number of customers without a loan.
- DigitalTRXratio: shows a normal distribution, indicating most customers have a mid-range ratio of digital transactions, with fewer customers having very high or very low ratios.
- Inactive (target): shows a significant class imbalance, with far more active customers than inactive ones.

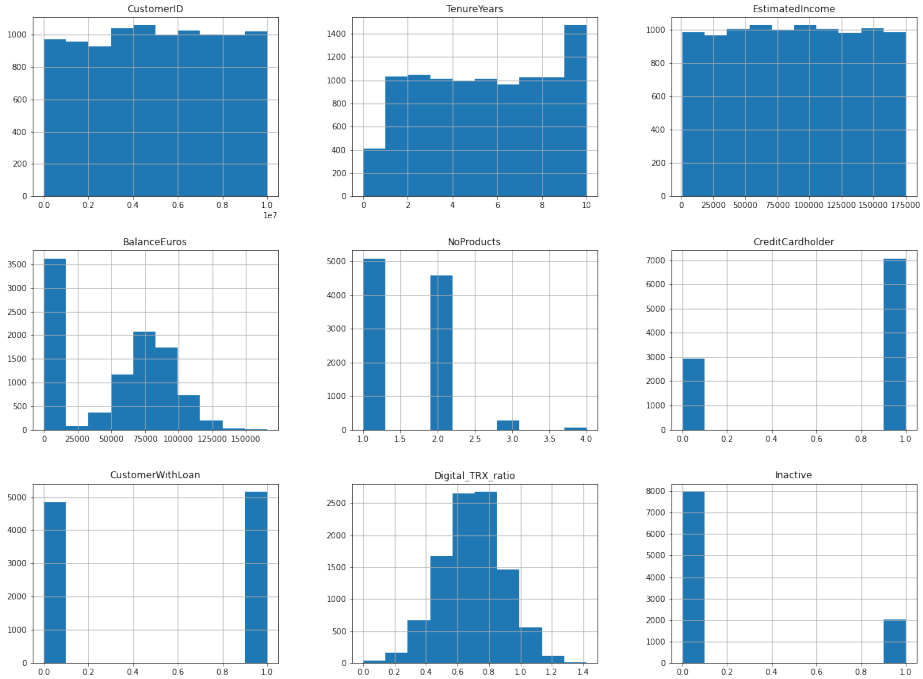


Figure 3: Distribution histograms

3.2 Correlation Analysis

Correlation analysis is a statistical technique used to measure the strength and direction of the relationship between two or more variables. It helps to determine how closely two variables are related and whether changes in one variable are associated with changes in another variable.

The result of a correlation analysis is often represented by a correlation coefficient, which is a numerical value that ranges between -1 and 1. More specifically:

- A correlation coefficient of +1 indicates a perfect positive correlation, meaning that as one variable increases, the other variable increases in a proportional manner.
- A correlation coefficient of -1 indicates a perfect negative correlation, meaning that as one variable increases, the other variable decreases in a proportional manner.

- A correlation coefficient close to 0 suggests no or weak correlation between the variables.

From the correlation heatmap in Figure 4, we can extract several pieces of valuable information regarding the relationships between different features and the target variable (Inactive). More specifically:

- BalanceEuros and Inactive (0.12): A small positive correlation between BalanceEuros and Inactive features, that suggests that higher balances might be associated with likelihood of becoming inactive
- CustomerWithLoan and Inactive (-0.16)**: A small negative correlation between CustomerWithLoan and Inactive features, that suggests that customers with loans are less likely to become inactive, which is logical based on their financial obligations with the bank.

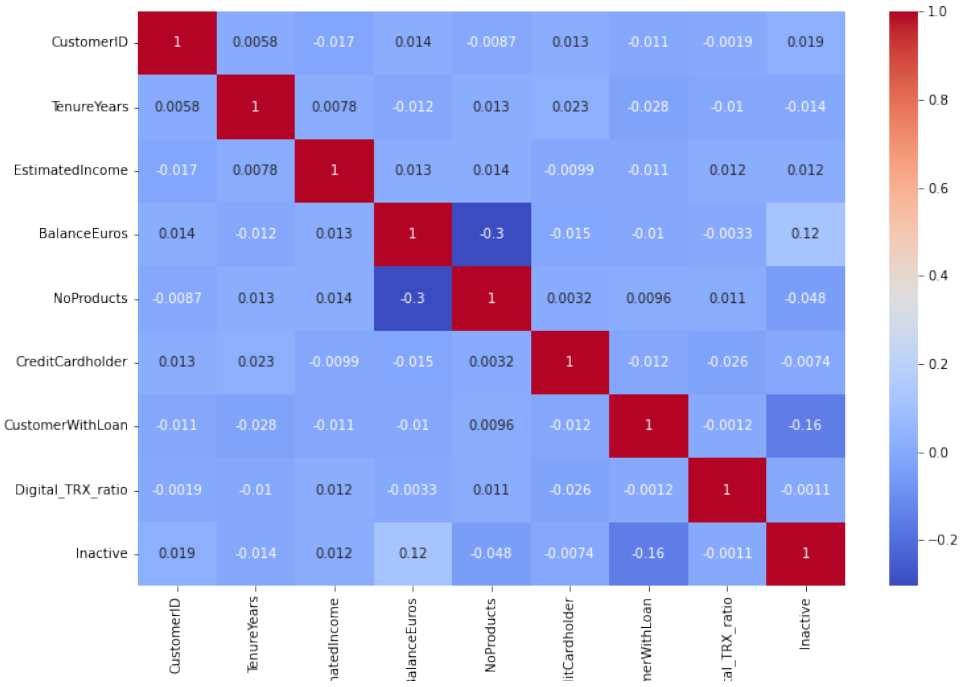


Figure 4: Corellation Heatmap.

3.3 Relationship Between Features

3.3.1 Imbalance Dataset

As we mentioned before in our distribution analysis, we have a significant class imbalance, with far more active customers than inactive ones. More specifically, we have 7958 Active customers and 2042 Inactive customers 5. So, we need to use techniques to deal with class imbalance.

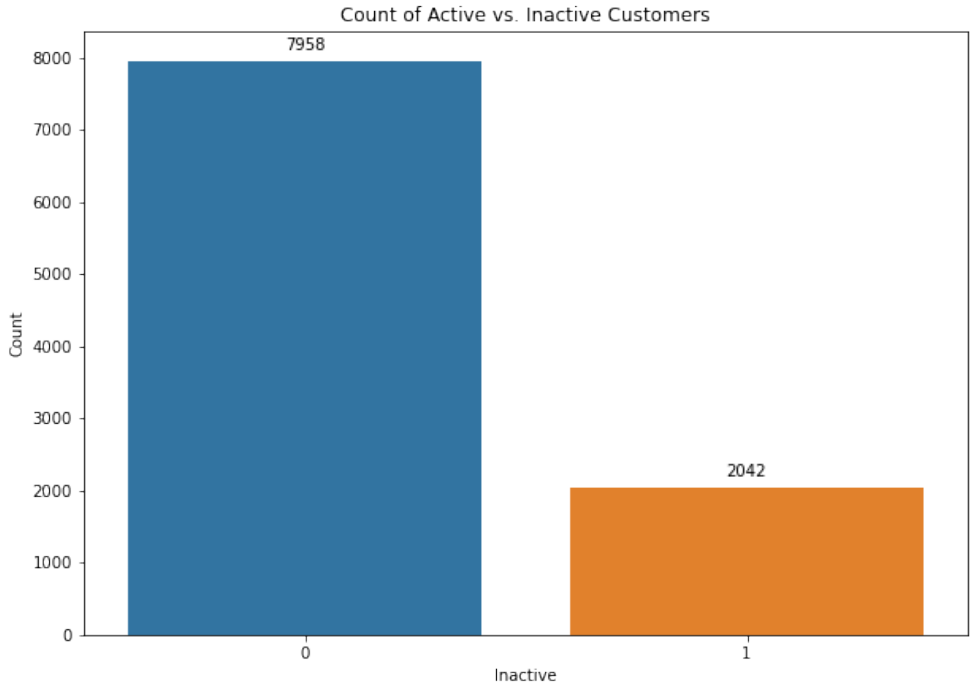


Figure 5: Count of Active vs. Inactive Customers

3.3.2 Geography vs Inactive

From the first plot in Figure 6 we can see that Athens has the highest number of customers among the three regions, while Rest_GR and Thessaloniki have almost the same number. From the second plot it is clear that besides the region, number of active users are greater than the number of inactive users.

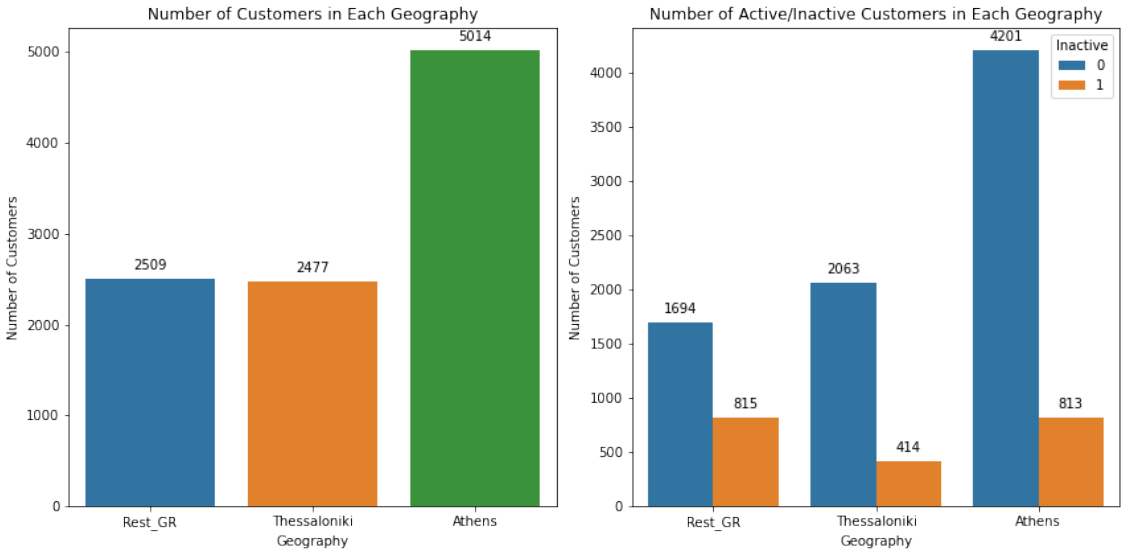


Figure 6: Geography vs Inactive

3.3.3 Gender vs Inactive

From the first plot in Figure 7 we can see that male and female customers are almost the same, with the male customers to be slightly more. From the second plot it is clear that besides the gender of the customers, number of active users are greater than the number of inactive users.

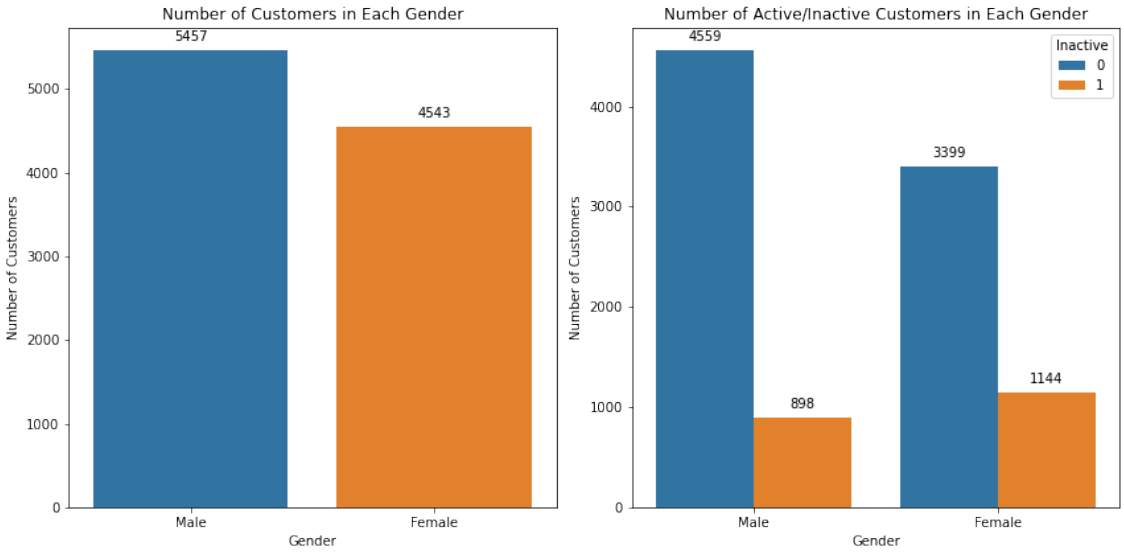


Figure 7: Gender vs Inactive

3.3.4 Age vs Inactive

From the first plot in Figure 8 we can see that the age bands 25-35 and 35-45 have the highest number of customers, indicating a larger customer base in these age groups. From the second plot we can observe that in age bands 18-25, 25-35, 35-45 and 65+ the number of active customers is much greater than the number of inactive customers, while on the other hand in the age bands 45-55 and 55-65 we have equal number of active and inactive customers.

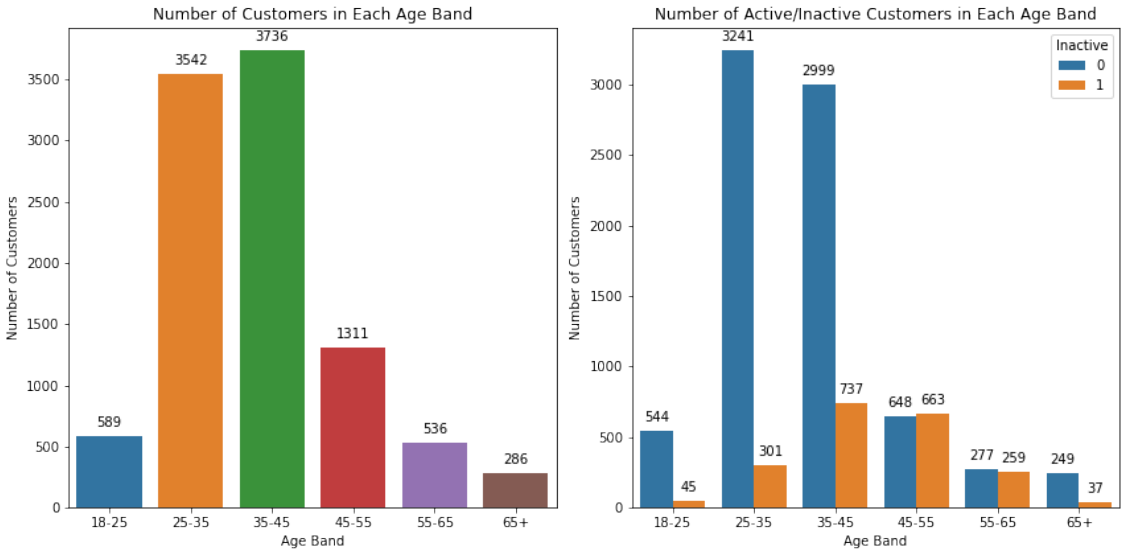


Figure 8: Age vs Inactive

4 Data Preprocessing

4.1 Encode Categorical Features

Categorical data, also known as nominal or ordinal data, is a type of data that consists of values that fall into distinct categories or groups. Unlike numerical data, which represents measurable quantities, categorical data represents qualitative or descriptive characteristics. It is crucial to understand categorical data when working with machine learning models, as most models require numerical inputs, so we need to convert them.

There are two main types of categorical data:

- **Ordinal Data:** refers to categories that have an inherent order or ranking. When encoding ordinal data, it's essential to retain the information about the order
- **Nominal Data:** refers to categories that do not have an inherent order or ranking. When encoding nominal data, the presence or absence of a feature is considered, but the order is not relevant

Our dataset consists of 3 categorical features: Geography, Gender and Age_Band. Geography and Gender features are nominal data so we are going to perform One-Hot Encoding technique, where it creates a new binary column for each category, while for the Age_Band which is an ordinal feature we are going to perform Label Encoding where it assigns a unique integer to each category. Dataset after categorical encoding is depicted in Figure 9.

CustomerID	Geography_Athens	Geography_Rest_GR	Geography_Thessaloniki	Gender_Female	Gender_Male	Age_Band	TenureYears	EstimatedIncome	BalanceEuros	NoProducts	CreditCardholder	CustomerWithLoan	Digital_TRX_ratio	Inactive
5188208	0	1	0	0	1	0	0	40683.96	50086.2120	1	0	0	0.38	0
8683784	0	0	1	1	0	5	4	2429.51	0.0000	1	1	0	0.33	1
3512360	1	0	0	0	1	3	4	41694.49	26852.7072	1	1	1	0.72	0
7104818	0	1	0	0	1	1	5	74523.33	90325.6200	1	0	0	0.08	0
6712745	0	1	0	1	0	1	9	111050.49	100537.0608	2	0	0	1.38	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
5603293	1	0	0	1	0	1	5	102771.03	0.0000	2	0	0	0.72	0
6827613	0	1	0	0	1	2	8	143715.62	82870.9002	1	0	0	0.69	1
5226960	0	1	0	0	1	2	1	35812.84	65007.1752	1	1	0	0.89	1
280950	1	0	0	0	1	2	1	99667.13	82639.8672	1	1	1	0.94	0
1831292	1	0	0	0	1	2	5	76783.59	84390.5040	1	1	0	0.61	0

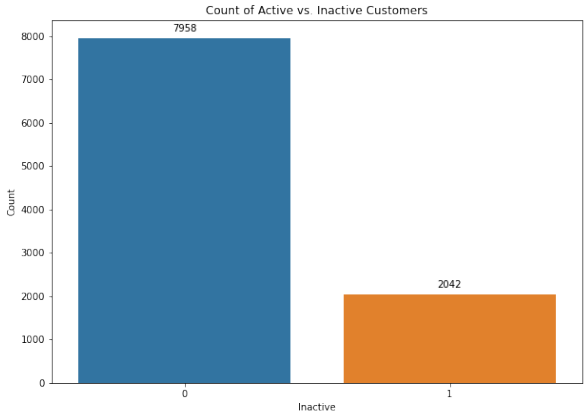
Figure 9: Dataset after Categorical Encoding

4.2 Handle Imbalanced Data

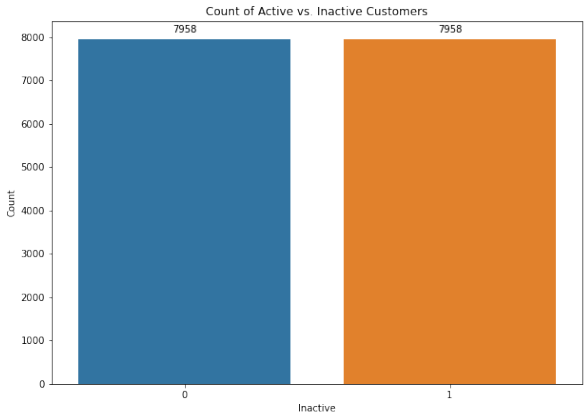
Imbalanced dataset refers to datasets where the target classes do not have equal number of instances. The problem is that the model will not predict accurately minority class, so we will end up with a biased model.

To handle imbalanced dataset in out situation we used a technique called **SMOTE (Synthetic Minority Oversampling Technique)**, which oversamples the minority class. SMOTE looks into minority class instances and use K nearest neighbor to select a random nearest neighbor, and a synthetic instance is created randomly in feature space. For one minority instance find neighbor and compute their distance. Then find a new instance on this distance.

In Figure 10a we can see Active vs Inactive users before applying SMOTE to our dataset, while on the other hand in Figure 10b after applying SMOTE. It is clear that it created 5916 instances of the minority class (inactive).



(a) Target Feature Distribution before SMOTE



(b) Target Feature Distribution after SMOTE

Figure 10: Target Feature Distribution

4.3 Feature Scaling

Feature Scaling is preprocessing technique that transforms feature values to a similar scale, ensuring all features contribute equally to the model. Some common techniques are MIN-MAX Scaling and Standardization.

While Gradient Descent Based Algorithms (Linear Regression, Logistic Regression, Neural Networks, PCA, etc) and Distance Based Algorithms (KNN, K-Means, SVM, etc) require feature scaling, Tree-Based Algorithms are insensitive to the scale of the features, and because our model will be a Random Forest Clasifier, we are not going to use a feature scaling technique.

5 Classification Model

As we mentioned before, we are going to use **Random Forest Classifier** as a machine learning model. A Random Forest Classifier is an ensemble learning method used for classification tasks that operates by constructing multiple decision trees during training and outputting the class that is the mode of the classes (classification) of the individual trees. It leverages the power of bagging, where each tree is trained on a random subset of the data, and feature randomness, where each split considers a random subset of features, to create a diverse set of classifiers. This approach helps to improve the model's accuracy and robustness by reducing overfitting and variance, making it particularly effective for complex datasets with nonlinear relationships.

First, we split our data into train and test set with stratification in order to ensure that the proportion of each class in the original dataset is preserved in both the training and testing sets.

Then we use a hyper-parameter tuning technique called RandomizedSearchCV to find the best parameters of Random Forest Classifier for the specific problem. RandomizedSearchCV is a technique used in hyper-parameter tuning to optimize the performance of machine learning models by randomly sampling a specified number of parameter combinations from a defined parameter grid. Unlike GridSearchCV, which exhaustively searches all possible combinations, RandomizedSearchCV is more efficient as it evaluates a fixed number of random combinations, making it suitable for large datasets or complex models where an exhaustive search would be computationally prohibitive. It helps in finding the best set of hyper-parameters by balancing the trade-off between computational efficiency and thoroughness, ultimately leading to improved model performance.

Finally, we train our model in the training set and we evaluate it on the test set. In Figure 11 we can see the confusion matrix and in Table 1 we can see the evaluation metrics.

Metric	**Value**
Accuracy	89.16%
Precision	90.57%
Recall	87.44%
F1-Score	88.97%

Table 1: Evaluation Metrics for the Classification Model

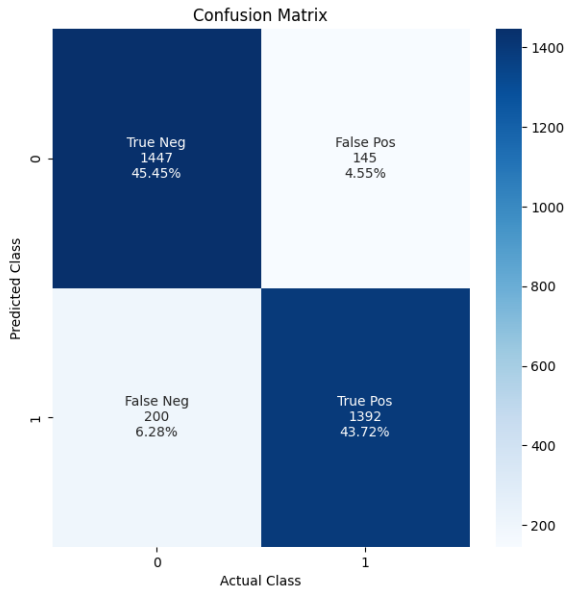


Figure 11: Confusion Matrix

6 Results and Analysis

The confusion matrix shows the following:

- True Negatives (TN): 1447 (45.45
- False Positives (FP): 145 (4.55
- False Negatives (FN): 200 (6.28
- True Positives (TP): 1392 (43.72

The table with the evaluation metrics shows the following:

- An accuracy of 89.16% indicates that the model is correct 89.16% of the time. Is generally good, indicating that the model performs well on the whole dataset.
- A precision of 90.57% means that when the model predicts a user will become inactive, it is correct 90.57% of the time. The model is reliable in predicting inactive users. This is particularly important in this context because false positives (predicting a user will become inactive when they won't) might lead to unnecessary interventions but are less critical than false negatives.

- A recall of 87.44% indicates that the model correctly identifies 87.44% of the users who will become inactive. The model is effective at identifying most users who will become inactive.
- An F1-Score of 88.97% indicates a good balance between precision and recall. Indicating that the model does not favor one over the other excessively.