Data Science

answer of assignments 4

Professor : Dr. Kherad Pishe

Assistant Professor : Mohammad Reza Khanmohammadi

Hasan Roknabady – 99222042

CREDIT CARD FRAUD ANALYSIS AND MODELING

From traditional to emerging sectors, there is not one single business that is fully immune from fraud. Some studies show that frauds of various kinds could cost businesses 1%-1.75% of their annual sales, this translates to around $200 billion a year!

As one of the most common types of fraudulent activities, credit card transaction fraud impacts around 127 million people, or approximately $8 billion in attempted fraudulent charges on Americans' credit and debit cards. It is therefore imperative for credit card companies to understand the characteristics of a fraudulent transaction and develop predictive models accordingly to flag down potentially risky activities for fraud prevention.

THE DATASET

In this project, we are examining the Credit Card Transactions Fraud Detection Dataset which contains both a training dataset and a testing dataset. We will first perform an exploratory data analysis to the training data to understand which features might be correlated to fraudulent activities and then attempt to create models with those features and test out their predicitve effectiveness.

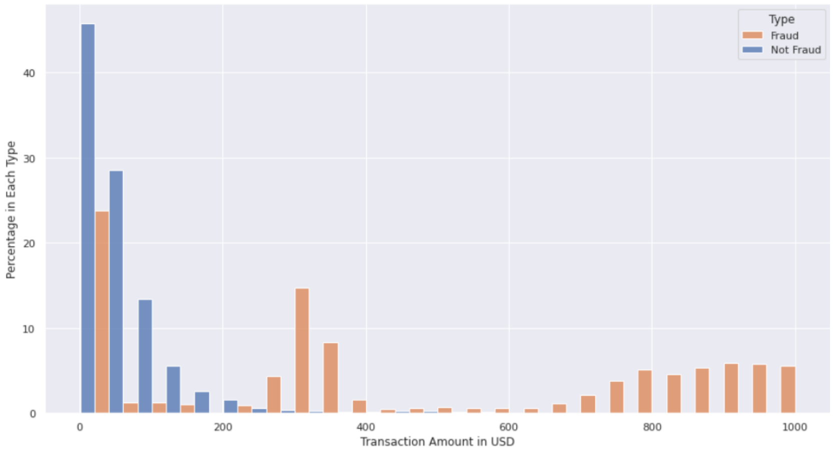
PREPARATION

the training dataset contains 23 columns that detail the time of the credit card transaction, the merchant, the spending category, the transaction amount, and personal infomration about the credit card holders, including their names, genders, locations and birthdays. It also contains a column called "is_fraud" which marks fraudulent transactions as 1 and non-fraudulent as 0. There is no missing data in the dataset and we also remove any duplicated observations in the data set to make it ready for further analysis.

EXPLORATORY DATA ANALYSIS

1. TRANSACTION AMOUNT VS FRAUD

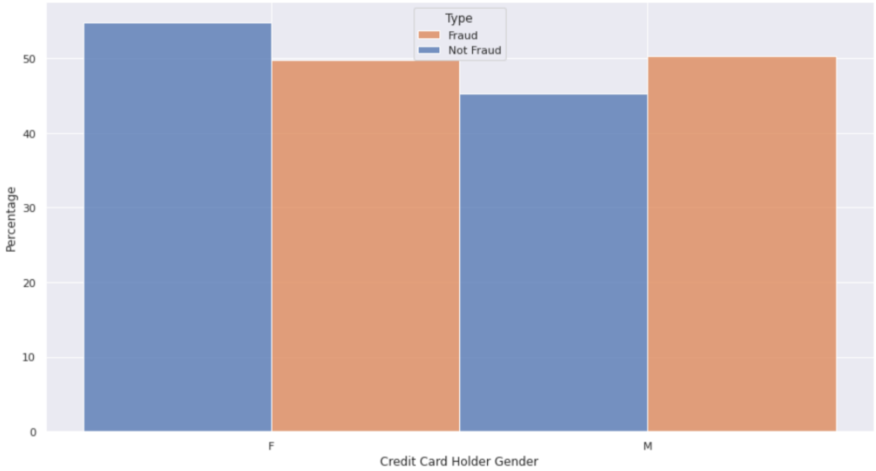
With the dataset cleaned, we can now start to examine how various features relate to fraud. First we will see how the distrition of transaction amount differs between fraudulent and normal activities. As there are extreme outliers in transaction amount, and the 99 percentile is around $546, we subset the data for any transaction amounts below \$1,000 to make the visualizations more readable.



The result is very interesting! While normal transactions tend to be around $200 or less, we see fraudulent transactions peak around \$300 and then at the $800-\$1000 range. There is a very clear pattern here!

2. GENDER VS FRAUD

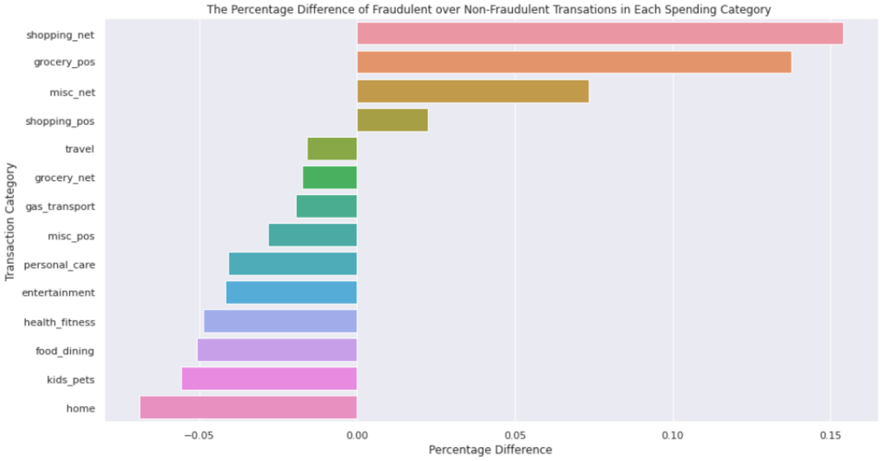
Second, we will examine whether one gender is more susceptible to fraud than the other.



In this case, we do not see a clear difference between both genders. Data seem to suggest that females and males are almost equally susceptible (50%) to transaction fraud. Gender is not very indicative of a fraudulent transaction.

3. SPENDING CATEGORY VS FRAUD

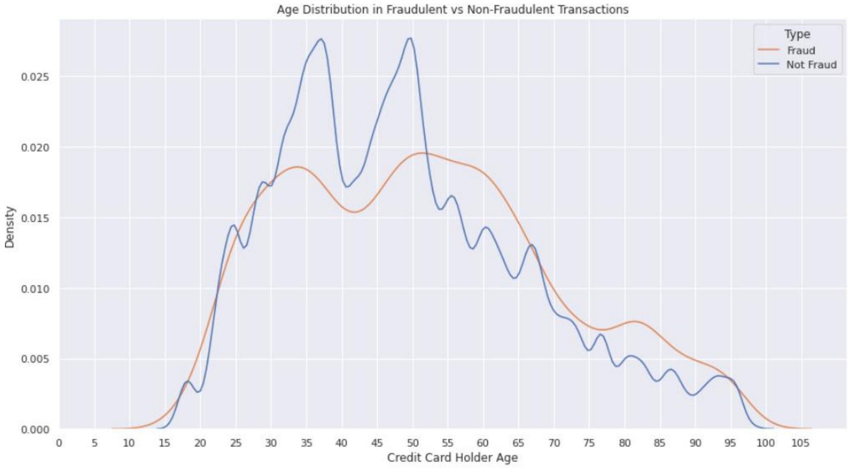
Third, we examine in which spending categories fraud happens most predominantly. To do this, we first calculate the distribution in normal transactions and then the the distribution in fraudulent activities. The difference between the 2 distributions will demonstrate which category is most susceptible to fraud. For example, if 'grocery_pos' accounts for 50% of the total in normal transactions and 50% in fraudulent transactions, this doesn't mean that it is a major category for fraud, it simply means it is just a popular spending category in general. However, if the percentage is 10% in normal but 30% in fraudulent, then we know that there is a pattern.



Some spending categories indeed see more fraud than others! Fraud tends to happen more often in 'Shopping_net', 'Grocery_pos', and 'misc_net' while 'home' and 'kids_pets' among others tend to see more normal transactions than fraudulent ones.

4. AGE VS FRAUD

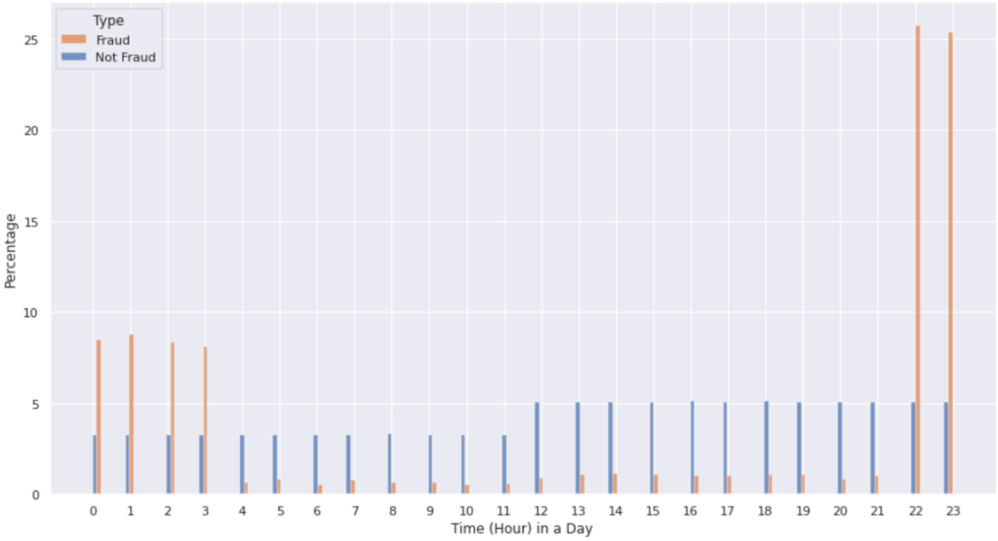
Are older people more prone to credit card fraud? Or is it the other way around? Given the birthday info, we can calculate the age of each card owner (in 2022) and see whether a trend exists.



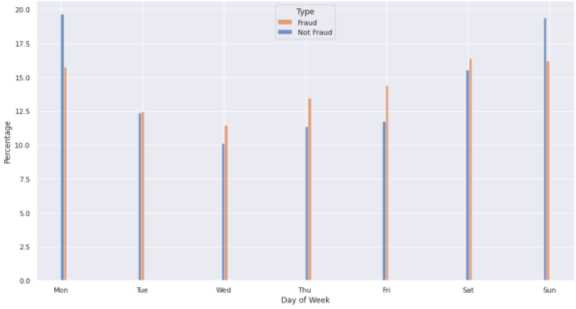
The age distribution is visibly different between 2 transaction types. In normal transactions, there are 2 peaks at the age of 37-38 and 49-50, while in fraudulent transactions, the age distribution is a little smoother and the second peak does include a wider age group from 50-65. This does suggest that older people are potentially more prone to fraud.

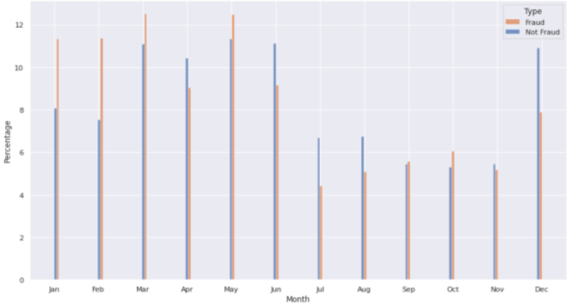
5. CYCLICALITY OF CREDIT CARD FRAUD

How do fraudulent transactions distribute on the temporal spectrum? Is there an hourly, monthly, or seasonal trend? We can use the transaction time column to answer this question.



A very sharp contrast! While normal transactions distribute more or less equally throughout the day, fraudulent payments happen disproportionately around midnight when most people are asleep!

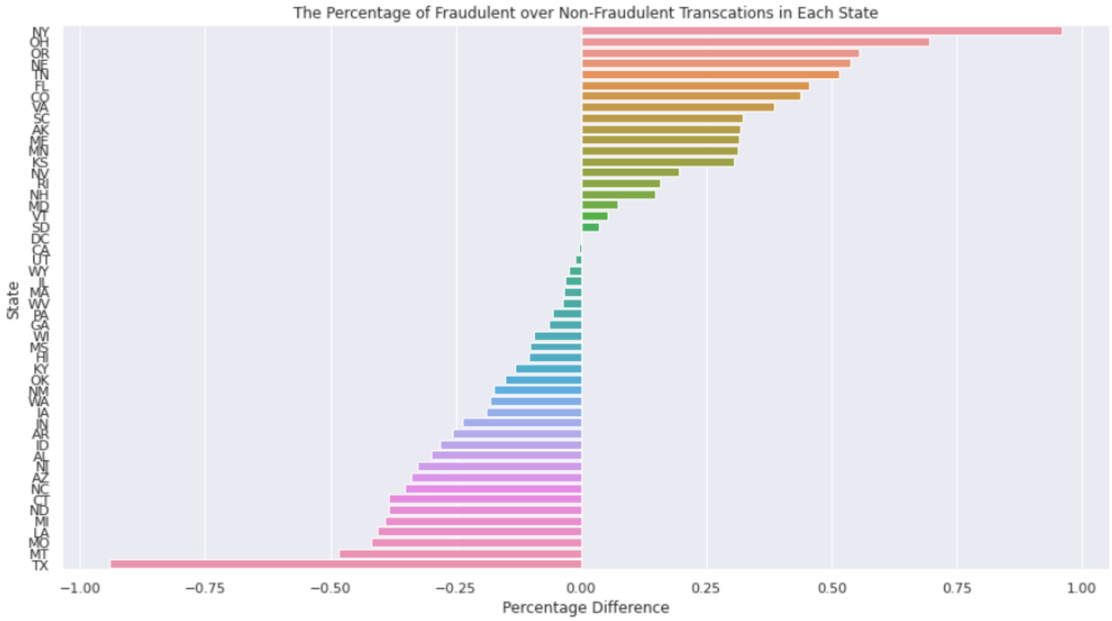




Normal transactions tend to happen more often on Monday and Sunday while fraudulent ones tend to spread out more evenly throughout the week, Very interesting results! While normal payments peak around December (Christmas), and then late spring to early summer, fraudulent transactions are more concentrated in Jan-May. There is a clear seasonal trend.

6. STATE VS FRAUD

Now that we have examined fraud on the temporal level, let's also explore which geographies are more prone to fraud. We will use the same methodology as in Part 3, where we calculate the difference in geographical distribution between the 2 transaction types.



As can be seen, NY and OH among others have a higher percentage of fraudulent transactions than normal ones, while TX and MT are the opposite. However, it should be pointed out that the percentage differences in those states are not very significant but a correlation does exist.

DATA MODELING AND PREDICTION

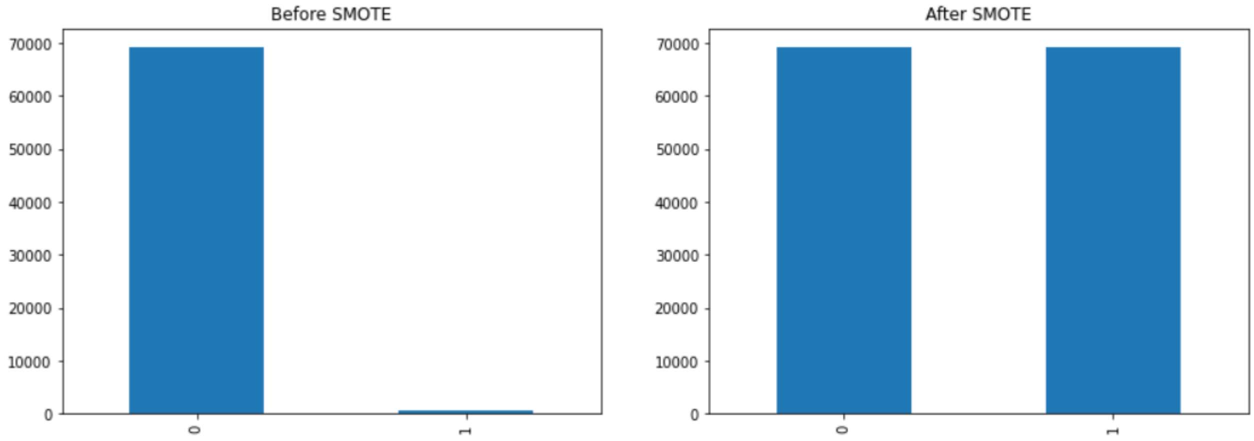
Based on our EDA above, we have found out that the features including transaction amout, credit card holder age, spending category, transaction time and locations all have varying degrees of correlations with credit card fraud. This helps us choose which features we want to include in our data models. The plan is to train the models on the training data set which we have analyzed above and then use the testing dataset to evaluate the model performance.

As data models need numeric input, we need to convert some of our categorical observations into numeric ones. For transaction locations and merchant locations, we already have the longitudinal and latitudinal data. But for shopping categories, we need convert them into dummy variables using pandas.get_dummies.

Now with both datasets cleaned and organized, we can start building models with them. We will first try to use Logistic Regression combined with confusion matrix to evaluate the model. As is very common with fraud data, there is always the issue of class imbalance where actual fraud cases are way fewer than normal cases and constitute only a very small part of the dataset. To counter this imbalance, it's important to use the SMOTE (Synthetic Minority Oversampling Technique) method to resample the training dataset so that the model can be trained on more balanced data for better results.

PRE PROCESSING: RESAMPLING AND SCAILING DATA

We perform data cleaning first after that we perform SMOTE

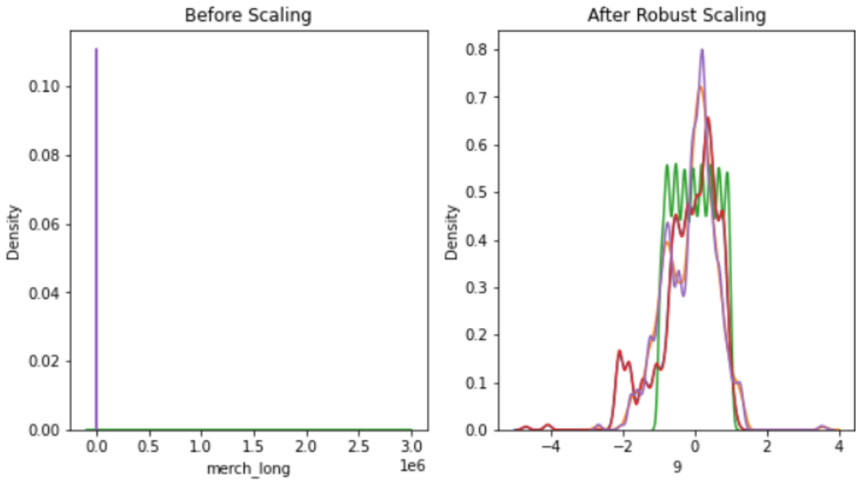


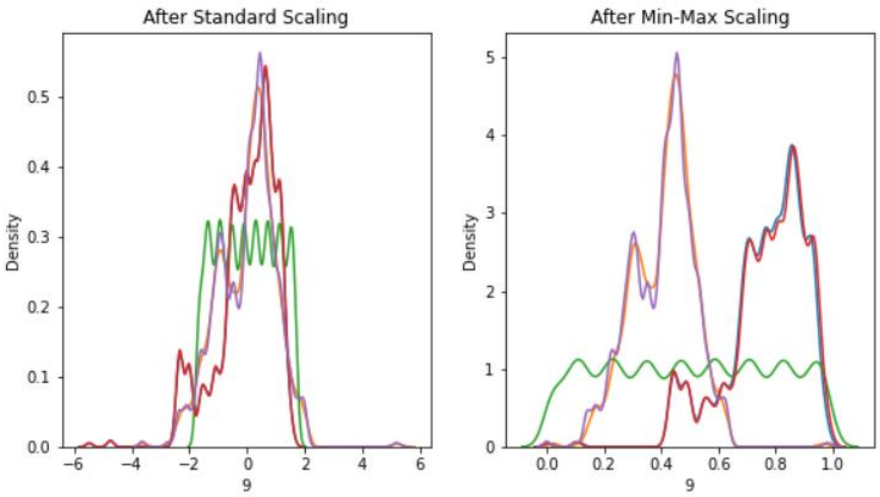
The dataset is heavily imbalanced. Through resampling, fraud transactions (Class = 1) are randomly increased to the same amount as non-fraud transactions (Class = 0) in order to avoid the bias results toward the non-fraudulent class.

SCALING

Robust Scaler VS MinMaxScaler VS Standard Scaler

We perform scailing in continue we have :





Since we have a huge amount of data, its better to normalize the dataset by using RobustScaler which scales the data according to the quantile range.

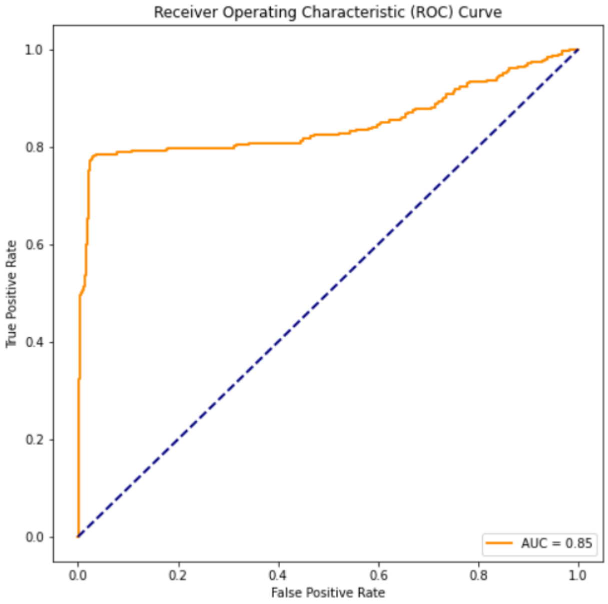
CREDIT CARD FRAUD DETECTION MODEL REPORT : 1. LOGISTIC REGRESSION

1. OVERVIEW:

We employed a Logistic Regression model for credit card fraud detection, aiming to balance precision and recall in identifying fraudulent transactions.

2. Model Evaluation:

- **Accuracy Analysis:**
    - **Overall Accuracy:** The model achieved an accuracy of 95.20%, indicating the proportion of correctly predicted instances among all instances.

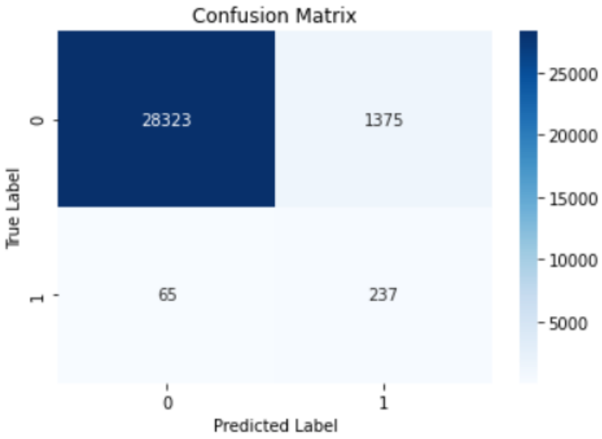


- **Precision (Positive Predictive Value):** Precision is the ratio of correctly predicted positive observations to the total predicted positives. In our case, precision is 14.70%, suggesting that of all transactions predicted as fraudulent, only 14.70% were actually fraudulent.

- **Recall (Sensitivity/True Positive Rate):** Recall is the ratio of correctly predicted positive observations to all observations in the actual class. The model achieved a recall of 78.48%, indicating that it successfully identified 78.48% of all actual fraudulent transactions.

- **F1 Score:** The F1 Score is the weighted average of precision and recall. It is a useful metric when there is an uneven class distribution. The model achieved an F1 Score of 24.76%.

3. Detailed Analysis:



- **False Positives (Type I Error):** The model predicted 1375 instances as fraudulent when they were not. Analyzing these false positives could reveal patterns or characteristics that led to misclassifications.

- **False Negatives (Type II Error):** There were 65 instances of actual fraudulent transactions that the model failed to predict. Understanding these false negatives is crucial for improving the model's sensitivity.

4. Recommendations:

- **Threshold Adjustment:** Depending on business priorities, consider adjusting the classification threshold. Increasing the threshold can enhance precision but might decrease recall, and vice versa.

- **Feature Importance:** Analyze feature importance to identify key variables influencing the model's decisions. This can guide further feature engineering or model refinement.

- **Model Comparison:** Explore other classification algorithms or ensemble methods to compare and potentially improve performance.

- **Continuous Monitoring:** Regularly update the model with new data and assess its performance to ensure relevance and accuracy over time.

5. Next Steps:

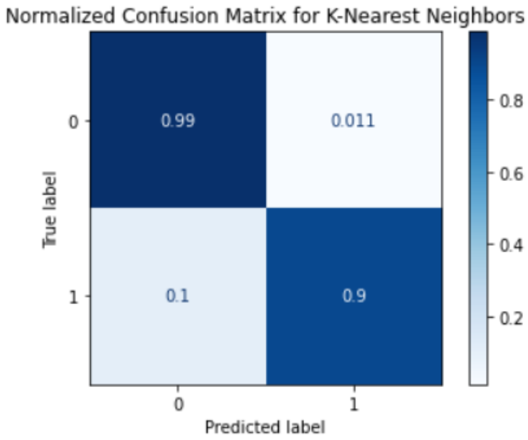
- Conduct a deeper investigation into misclassified instances to uncover underlying patterns.

- Experiment with hyperparameter tuning to optimize model performance.

- Consider incorporating more advanced techniques such as anomaly detection for fraud detection.

2. K-NEAREST NEIGHBORS (KNN) MODEL ANALYSIS

Confusion Matrix:



Interpretation:

- **True Positive (TP):** 271 instances were correctly classified as positive.
- **True Negative (TN):** 29382 instances were correctly classified as negative.
- **False Positive (FP):** 316 instances were incorrectly classified as positive.
- **False Negative (FN):** 31 instances were incorrectly classified as negative.

Classification Report:

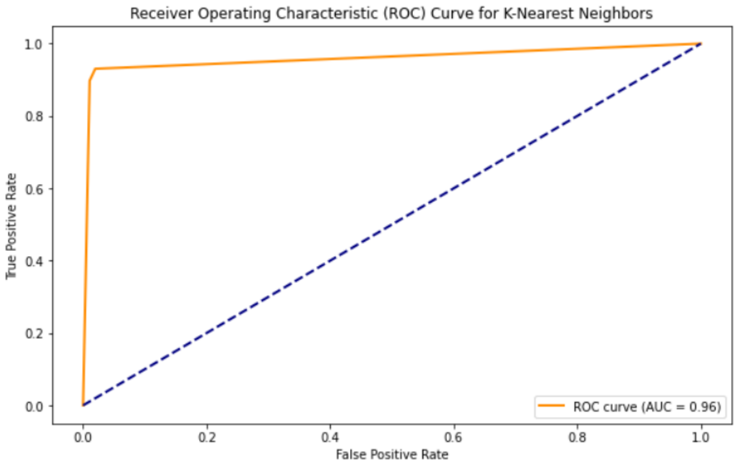
```
[[29382   316]
 [   31   271]]


              precision    recall  f1-score   support

           0       1.00      0.99      0.99     29698
           1       0.46      0.90      0.61       302

    accuracy                           0.99     30000
   macro avg       0.73      0.94      0.80     30000
weighted avg       0.99      0.99      0.99     30000
```

Key Metrics:

- **Precision (Positive Class):** 46% of instances predicted as positive were correct.
- **Recall (Positive Class):** 90% of actual positive instances were correctly predicted.
- **F1-Score (Positive Class):** The harmonic mean of precision and recall is 0.61.

- **Accuracy:** Overall accuracy of the model is 99%.

Overall Analysis:



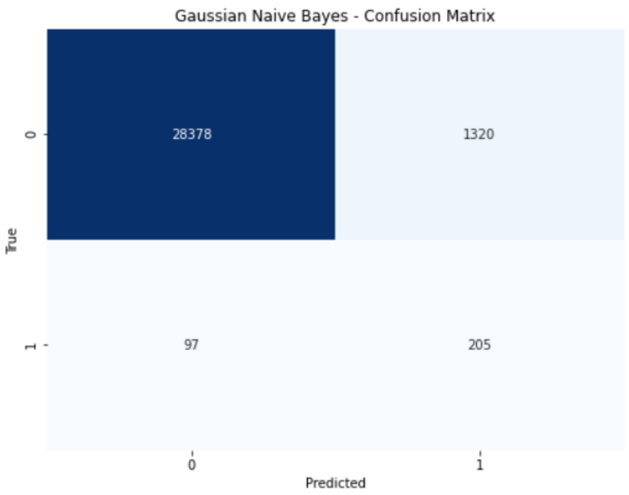
- The model demonstrates high overall accuracy (99%), indicating strong predictive performance.
- Precision for the positive class is relatively low (46%), suggesting caution when interpreting positive predictions.
- The model excels in recall for the positive class (90%), capturing a substantial portion of actual positives.
- The F1-Score balances precision and recall for the positive class.

Recommendations:

- Consider the specific context and consequences of false positives and false negatives to determine the trade-offs that best suit your application.
- Evaluate whether adjusting the model's threshold or exploring additional feature engineering could further optimize performance.
- Monitor and analyze the model's predictions in real-world scenarios to refine and improve its effectiveness.

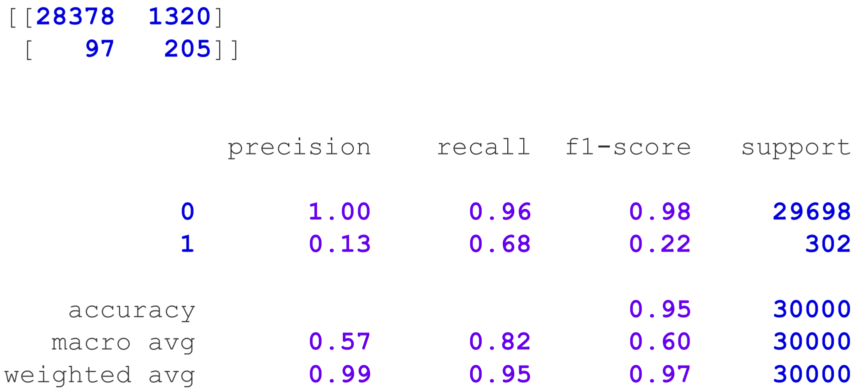
3. GAUSSIAN NAIVE BAYES MODEL ANALYSIS

1. Confusion Matrix:



- **True Positive (TP):** 205
- **True Negative (TN):** 28378
- **False Positive (FP):** 1320
- **False Negative (FN):** 97

2. Classification Report:



```
[[28378  1320]
 [   97   205]]


              precision    recall  f1-score   support

           0       1.00      0.96      0.98     29698
           1       0.13      0.68      0.22       302

    accuracy                           0.95     30000
   macro avg       0.57      0.82      0.60     30000
weighted avg       0.99      0.95      0.97     30000
```

precision recall f1-score support 0 1.00 0.96 0.98 29698 1 0.13 0.68 0.22 302 accuracy 0.95 30000 macro avg 0.57 0.82 0.60 30000 weighted avg 0.99 0.95 0.97 30000

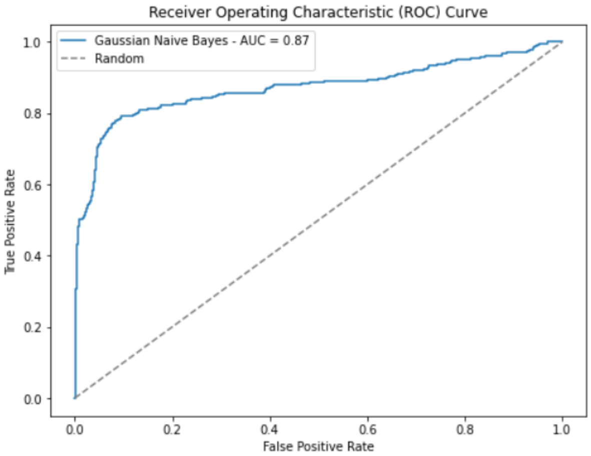
- **Accuracy:** 95%
- **Precision (Positive Predictive Value):** 13%
- **Recall (Sensitivity or True Positive Rate):** 68%

- **F1-Score:** 22%

3. Analysis:

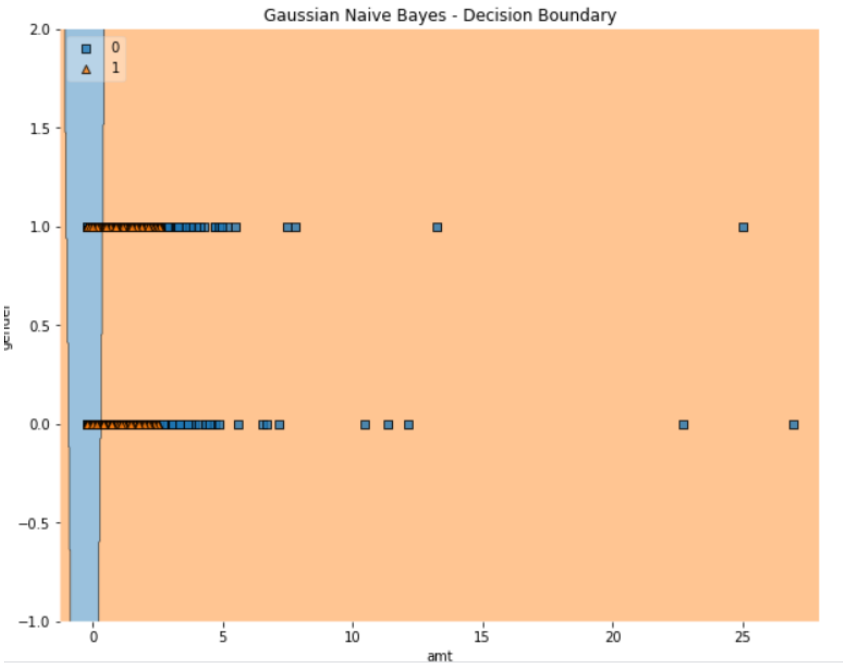
- **Accuracy:** The model achieves a high overall accuracy of 95%, indicating a good proportion of correctly classified instances.
- **Precision:** The precision for the positive class is relatively low at 13%. This suggests that when the model predicts a positive instance, it is often incorrect.
- **Recall:** The recall for the positive class is better at 68%, indicating that the model can identify a significant portion of the actual positive instances.
- **F1-Score:** The F1-Score, which is the harmonic mean of precision and recall, is 22%, providing a balanced measure of the model's performance.
- **False Positives:** The model has a notable number of false positives (1320) compared to true positives (205), contributing to the low precision for the positive class.

4. Receiver Operating Characteristic (ROC) Curve:



- A more detailed analysis can be obtained by examining the ROC curve and the Area Under the Curve (AUC) score. This will provide insights into the model's ability to discriminate between the classes.

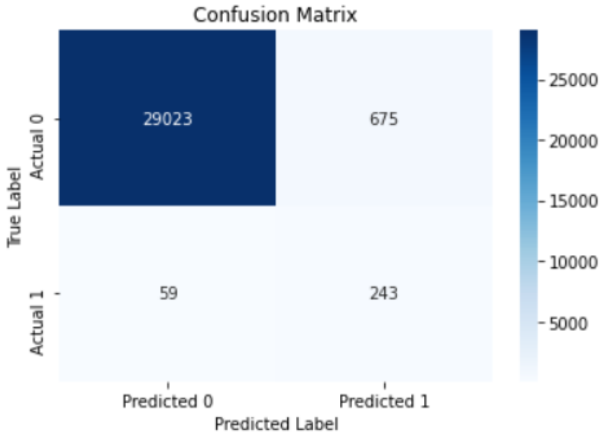
5. Decision Boundary Visualization:



- The Gaussian Naive Bayes model assumes axis-aligned elliptical decision boundaries due to its assumption of conditional independence between features given the class. While visualizing decision boundaries directly is challenging, examining feature pairs' decision regions can provide some insight into the model's behavior.

4. DECISION TREE MODEL ANALYSIS

Confusion Matrix:

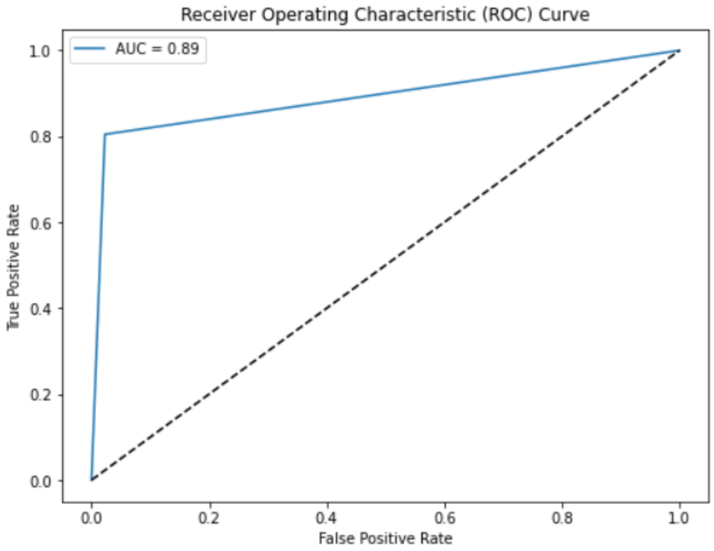


- True Positive (TP): 243
- True Negative (TN): 29023

- False Positive (FP): 675
- False Negative (FN): 59

Classification Report:

```
[[29023   675]
 [   59   243]]


              precision    recall  f1-score   support

           0       1.00      0.98      0.99     29698
           1       0.26      0.80      0.40       302

    accuracy                           0.98     30000
   macro avg       0.63      0.89      0.69     30000
weighted avg       0.99      0.98      0.98     30000
```



precision recall f1-score support 0 1.00 0.98 0.99 29698 1 0.26 0.80 0.40 302 accuracy 0.98 30000 macro avg 0.63 0.89 0.69 30000 weighted avg 0.99 0.98 0.98 30000

Analysis:

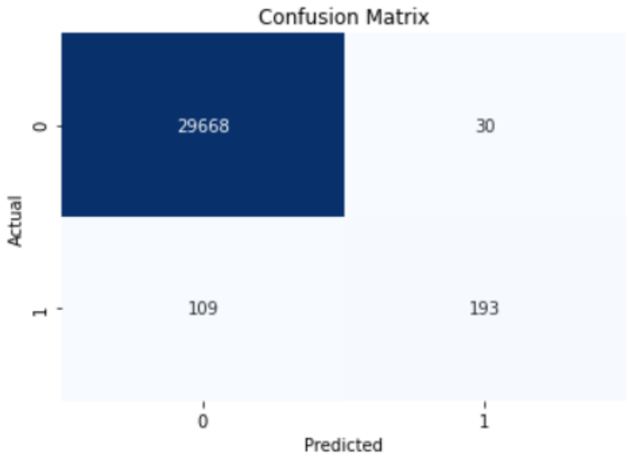
- **Accuracy:** The model has an accuracy of 98%, meaning it correctly predicted the class for 98% of the instances in the test set.
- **Precision:** Precision for class 0 is very high (1.00), indicating that when the model predicts class 0, it is usually correct. Precision for class 1 is lower (0.26), suggesting that there are false positives in the predictions for class 1.
- **Recall (Sensitivity):** Recall for class 0 is high (0.98), indicating that the model captures most of the instances of class 0. Recall for class 1 is even higher (0.80), suggesting that the model is good at identifying instances of class 1.

- **F1-Score:** The F1-score is a balance between precision and recall. The weighted average F1-score is 0.98, which is quite good.

- **Support:** The number of actual instances for each class.

5. RANDOM FOREST

Confusion Matrix:



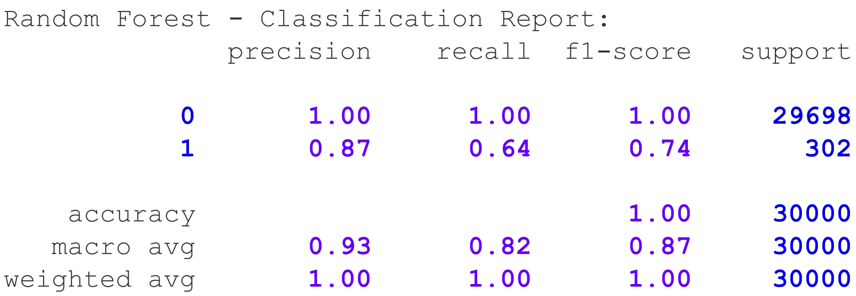
- **True Positives (TP):** 193 - The number of instances where the model correctly predicted class 1.

- **True Negatives (TN):** 29668 - The number of instances where the model correctly predicted class 0.

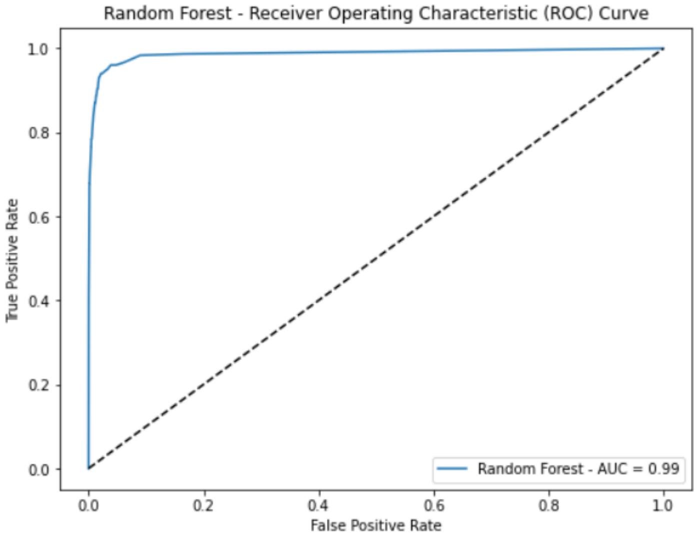
- **False Positives (FP):** 30 - The number of instances where the model incorrectly predicted class 1.

- **False Negatives (FN):** 109 - The number of instances where the model incorrectly predicted class 0.

Random Forest - Classification Report:



```
Random Forest - Classification Report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     29698
           1       0.87      0.64      0.74       302

    accuracy                           1.00     30000
   macro avg       0.93      0.82      0.87     30000
weighted avg       1.00      1.00      1.00     30000
```



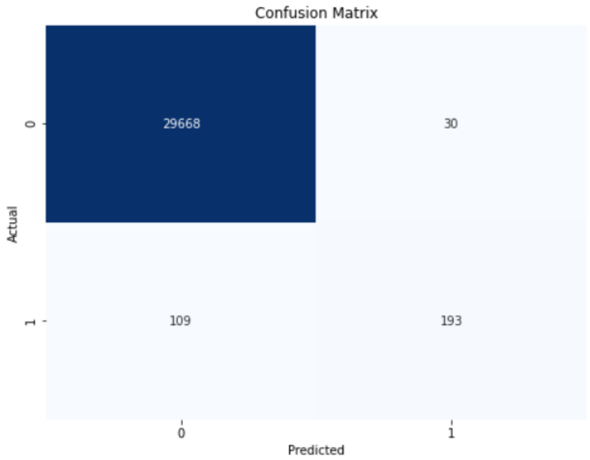
- **Precision (Positive Predictive Value):** 0.87 - Out of all instances predicted as class 1, 87% were actually class 1.
- **Recall (Sensitivity, True Positive Rate):** 0.64 - Out of all actual class 1 instances, the model correctly predicted 64%.
- **F1-score:** 0.74 - The harmonic mean of precision and recall. It balances precision and recall.
- **Accuracy:** 1.00 - Overall correctness of the model on the entire dataset.
- **Macro Avg (Macro Average):** An unweighted average of precision, recall, and f1-score for both classes.
- **Weighted Avg (Weighted Average):** An average where each instance contributes proportionally to its class.

Analysis:

- The model shows high accuracy (1.00) on the given dataset.
- Class 0 has perfect precision, recall, and f1-score, indicating excellent performance on this class.
- Class 1 has lower precision, recall, and f1-score, suggesting that the model is not as accurate in predicting instances of class 1.

6. ADABOOST

Confusion Matrix:

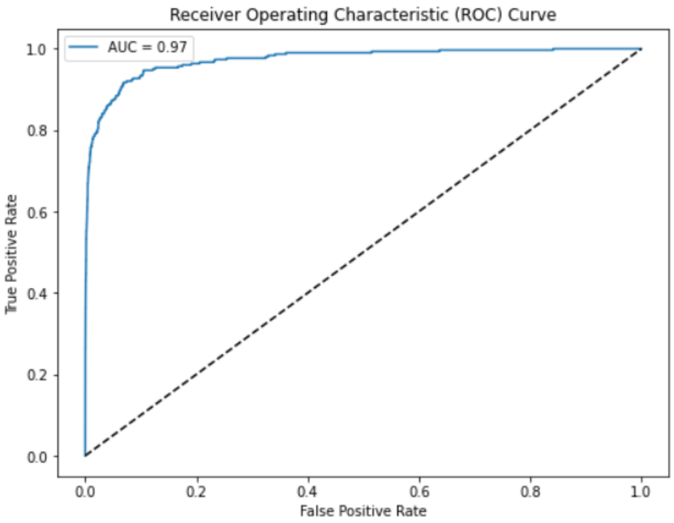


This confusion matrix shows the following:

- True Positive (TP): 145
- True Negative (TN): 29652
- False Positive (FP): 46
- False Negative (FN): 157

Classification Report:

```
     precision    recall  f1-score   support

           0       0.99      1.00      1.00     29698
           1       0.76      0.48      0.59       302

    accuracy                           0.99     30000
   macro avg       0.88      0.74      0.79     30000
weighted avg       0.99      0.99      0.99     30000
```



Analysis:

1. **Accuracy:** The overall accuracy of the model is 99%, which is quite high. However, accuracy alone may not be sufficient to evaluate a model, especially in imbalanced datasets.

2. **Precision-Recall Tradeoff:** The precision for class 1 (positive class) is 0.76, indicating that when the model predicts class 1, it is correct about 76% of the time. The recall (sensitivity) is 0.48, meaning that the model captures only 48% of the actual positive cases.

3. **F1-Score:** The F1-score is the harmonic mean of precision and recall. For class 1, the F1-score is 0.59, which balances precision and recall.

4. **Support:** The support column in the classification report indicates the number of actual instances of each class. Class 0 has a much higher support (29698) than class 1 (302), indicating a class imbalance.

5. **Macro and Weighted Averages:** The macro average and weighted average of precision, recall, and F1-score provide an overall summary. The macro average gives equal weight to each class, while the weighted average considers the number of samples in each class.

Conclusion:

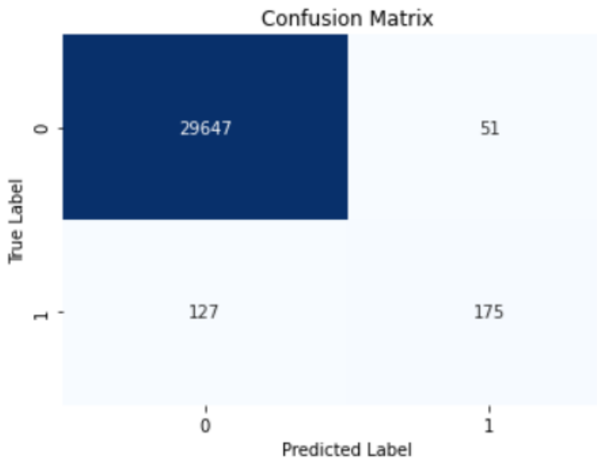
- The model performs exceptionally well in predicting the majority class (class 0) as indicated by high precision, recall, and F1-score.

- However, there is room for improvement in predicting the minority class (class 1), as reflected in lower precision, recall, and F1-score for this class.

- Consider addressing the class imbalance, exploring feature importance, or trying different model architectures to enhance the model's performance on the minority class.

7.BAGGING

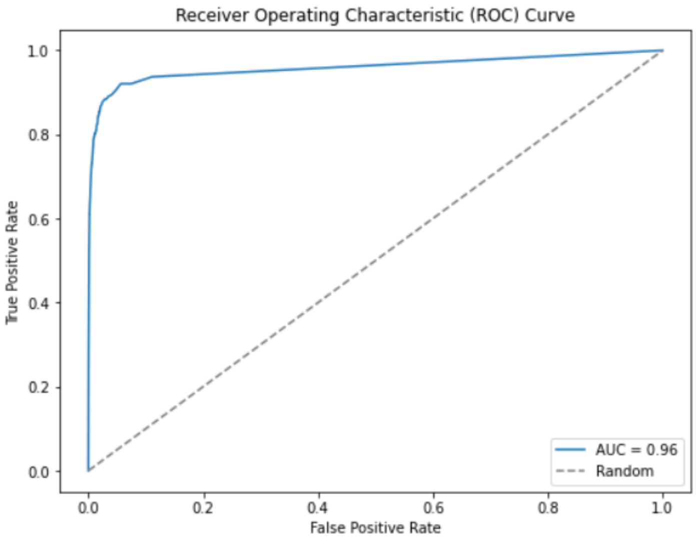
Confusion Matrix:



- **True Positives (TP):** 175 - The actual positive class instances correctly predicted as positive.
- **True Negatives (TN):** 29647 **-** The actual negative class instances correctly predicted as negative.
- **False Positives (FP):** 51 **-** The actual negative class instances incorrectly predicted as positive.
- **False Negatives (FN):** 127 **-** The actual positive class instances incorrectly predicted as negative.

Classification Report:

```
    precision    recall  f1-score   support

           0       1.00      1.00      1.00     29698
           1       0.77      0.58      0.66       302

    accuracy                           0.99     30000
   macro avg       0.89      0.79      0.83     30000
weighted avg       0.99      0.99      0.99     30000
```



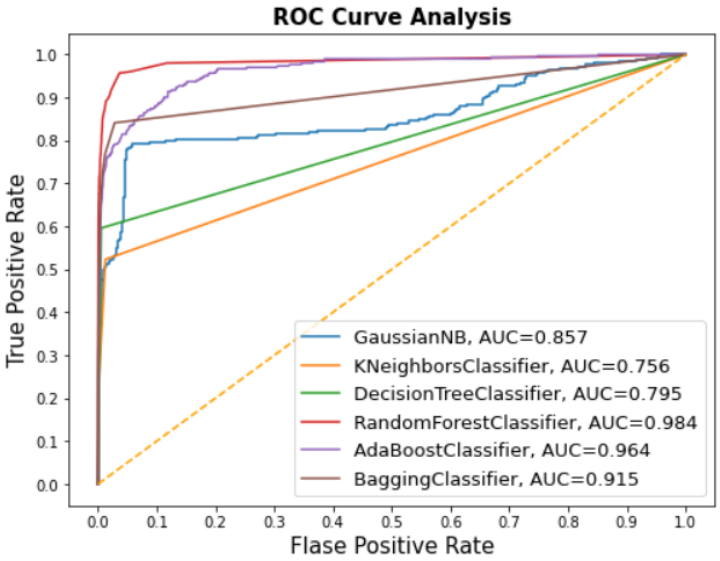
- **Precision:** The ratio of correctly predicted positive observations to the total predicted positives. For class 1, precision is 0.77, indicating that when the model predicts class 1, it is correct 77% of the time.
- **Recall (Sensitivity):** The ratio of correctly predicted positive observations to the total actual positives. For class 1, recall is 0.58, indicating that the model correctly identifies 58% of the actual positive instances.
- **F1-score:** The weighted average of precision and recall. For class 1, the F1-score is 0.66, providing a balance between precision and recall.
- **Support:** The number of actual occurrences of the class in the specified dataset.
- **Accuracy:** The overall accuracy of the model on the entire dataset is 99%.
- **Macro Avg:** The average of precision, recall, and F1-score for both classes, giving equal weight to each class.
- **Weighted Avg:** The weighted average of precision, recall, and F1-score, with weights based on the number of true instances for each class.

Analysis:

- The model performs exceptionally well on the majority class (class 0) with high precision, recall, and F1-score, indicating accurate predictions for the negative class.
- For the minority class (class 1), the model has lower precision and recall, suggesting that there may be room for improvement in identifying positive instances.
- The overall accuracy of 99% might be misleading due to the class imbalance. It's crucial to consider precision, recall, and F1-score, especially for the minority class, to evaluate model performance comprehensively.

CLASSIFICATION EVALUATION VIA AUROC

Based on the AUC values :



1. **Strong Performers:**

   - **RandomForestClassifier and AdaBoostClassifier:** These classifiers demonstrated excellent performance with AUC scores of 0.98 and 0.96, respectively. They are robust ensemble methods that effectively discriminate between positive and negative examples.

2. **Good Performers:**

   - **BaggingClassifier and GaussianNB:** Both classifiers achieved AUC scores around 0.92 and 0.86, respectively. Bagging shows good performance, while GaussianNB, a Naive Bayes classifier, also performed well.

3. **Moderate Performers:**

   - **DecisionTreeClassifier and KNeighborsClassifier:** These classifiers exhibited moderate performance with AUC scores of approximately 0.80 and 0.76, respectively. Further exploration of hyperparameter tuning may enhance their effectiveness.

In conclusion, the Random Forest and AdaBoost classifiers stand out as top performers, showcasing strong discriminatory capabilities. Consider further fine-tuning of hyperparameters for DecisionTreeClassifier and KNeighborsClassifier if needed. Additionally, ensure a comprehensive evaluation using other metrics and explore potential ensemble strategies for improved overall model performance.