# Machine Learning Final Project

# Syed Hamza Jamil : 29454

# Note

**In this notebook there are three different section with 3 different dataset with 3 different Class imbalance technique including subsection like below**

**Section 1 A B C D : Section 2 A B C D : Section 3 A B C D for each individuall section I write interpretaion, it insights and conclusion**

**In last complete project summary**

# Class Imbalance Selected Techniques

- SMOTE (Synthetic Minority Over-sampling Technique): Generates synthetic samples for the minority class to balance the dataset, improving model performance on imbalanced data by providing more instances of the minority class.

- Class Weight Adjustment: Assigns higher weights to minority class samples during model training, allowing the model to pay more attention to these underrepresented classes without altering the dataset.

- Threshold Adjustment: Modifies the decision threshold for class predictions to favor the minority class, helping to balance precision and recall by adjusting the sensitivity of the classifier to different classes

# Instructions

- In this project I create a single master function that have a ability to perfrom complete data science life cycle on different dataset
- There are 3 different section in this projects, in each section,I used different ML technique with different dataset and also compare different ML alogithm accuracy on different dataset, all the selected datasets are imbalance,the technique that we are used are as belows :
- Explanatory Data Analysis
- Data Cleaning
- Data Scaling
- Feature Selection Chi-Square
- Class Imbalance Technique : Smote, Class Weight, Threshold
- Cross Validation with Hyperparameter Tunning
- ML Algorithm : KNN, Logistic, SVM, Random Forest, Naive Bysian
- Model accuracy matrics including classification report and Roc Curve Plot
- Model Accuracy Comparision

# Section 1 A : Simple ML Model

# Dataset 1 : Bank Marketing

- Dataset URL : https://archive.ics.uci.edu/dataset/222/bank+marketing
- Dataset Information : 45210 rows and 17 columns with Two class

# Result and Insight Explanation

- In this dataset the ratio of class imabalance are : 0.13%

- **KNN Model Insight**

The KNN model shows an overall accuracy of 88%, performing well on the majority class (0) with a precision of 0.91 and recall of 0.96. However, it struggles with the minority class (1), achieving a precision of 0.47 and recall of 0.28. The macro average scores indicate a significant disparity between the classes, highlighting the model's difficulty in predicting the minority class..

- **Logistic Regression Model Insight**

The Logistic Regression model has an overall accuracy of 89% and an AUC of 0.85. It performs well on the majority class (precision: 0.90, recall: 0.98) but poorly on the minority class (precision: 0.62, recall: 0.21). This indicates the model is biased towards the majority class, struggling to correctly predict the minority class.

- **Support Vector Machine (SVM)**

The SVM model shows an overall accuracy of 88% and an AUC of 0.72. While it has high precision (0.88) and perfect recall (1.00) for the majority class, it performs poorly for the minority class (precision: 0.54, recall: 0.01). This suggests the model is highly imbalanced, favoring the majority class.

- **Naive Bayes Model Insight**

The Naive Bayes model has an accuracy of 84% and an AUC of 0.82. It achieves good precision (0.93) for the majority class but only moderate recall (0.48) for the minority class. The model's performance indicates some capability in handling class imbalance but still favors the majority class significantly.

- **Random Forest Model Insight**

The Random Forest model achieves an accuracy of 90% and an AUC of 0.90. It performs well for the majority class (precision: 0.92, recall: 0.97) and better than other models for the minority class (precision: 0.62, recall: 0.37). This indicates a more balanced performance, though still favoring the majority class.

# Overall Conclusion for section 1 A

- The model accuracy comparison table shows that Random Forest has the highest testing accuracy (89.69%), indicating its strong performance on unseen data. Logistic Regression and KNN also perform well, with testing accuracies of 88.91% and 87.55%, respectively. SVM has a slightly lower testing accuracy (87.95%), while Naive Bayes performs the least effectively with a testing accuracy of 84.31%. Overall, Random Forest is the most reliable model for predictions, closely followed by Logistic Regression and KNN

# Section 1 B : CI Technique Resampling Smooth

- In this section i trained ML model with data normalization technique minmax, feature selection using Chi Square with hyperparameter tunning grid search and for handling class imbalance issue i use resampling smooth techniques

# Result and Insight Explanation

- **K-Nearest Neighbors (KNN)**

The KNN model, with optimal parameters of n_neighbors=7 and weights='distance', achieved a training and testing accuracy of 85%. It demonstrates balanced performance with a precision of 0.85 for both classes. The ROC curve shows a strong discriminative ability with an AUC of 0.91, indicating the model's good performance.

- **Logistic Regression**

Logistic Regression, with the best parameters C=10 and solver='liblinear', obtained a training and testing accuracy of 81%. It shows better performance on the minority class (precision: 0.83) than KNN but slightly lower recall. The ROC curve with an AUC of 0.89 suggests robust model performance.

- **Support Vector Machine (SVM)**

The SVM model achieved a training and testing accuracy of 81%, with balanced precision and recall for both classes. The model's ROC curve has an AUC of 0.89, indicating strong discriminative power, similar to Logistic Regression.

- **Naive Bayes**

The Naive Bayes model obtained a training and testing accuracy of 70%. It performs well for the minority class with a precision of 0.79 but has lower recall. The ROC curve with an AUC of 0.82 shows moderate performance in distinguishing between classes.

- **Random Forest**

Random Forest, with optimal parameters max_depth=20 and n_estimators=100, achieved a training and testing accuracy of 85%. It has balanced precision and recall for both classes and the highest AUC of 0.92 among the models, indicating excellent discriminative ability and robust performance.

# Overall Conclusion Section 1 B

Among the evaluated models, the Random Forest demonstrated the best overall performance with a testing accuracy of 85% and the highest AUC of 0.92, indicating strong discriminative power and balanced performance across classes. KNN and Logistic Regression also performed well, with testing accuracies of 85% and 81%, respectively, and high AUC values. The SVM model showed similar performance to Logistic Regression, while Naive Bayes, despite having a lower overall accuracy, performed moderately well for the minority class. Overall, Random Forest is the most suitable model for prediction in this scenario.

# Section 1 C : Class Imbalance Technique Class Weight

- In this section i trained ML model with data normalization technique minmax, feature selection using Chi Square with hyperparameter tunning grid search and for handling class imbalance issue i use class weight techniques

# Result and Insight Explanation

- **K-Nearest Neighbors (KNN)**

The KNN model, with optimal parameters n_neighbors=7 and weights='uniform', achieved a training and testing accuracy of 89%. It shows high precision for the majority class (0.92) but lower precision for the minority class (0.54). The ROC curve indicates a moderate discriminative ability with an AUC of 0.82.

- **Logistic Regression**

Logistic Regression, with the best parameters C=1.0, obtained a training and testing accuracy of 83%. It shows high precision for the majority class (0.97) but lower precision and recall for the minority class (0.38 and 0.78, respectively). The ROC curve with an AUC of 0.89 indicates strong discriminative power.

- **Support Vector Machine (SVM)**

The SVM model, with optimal parameters C=0.1 and kernel='linear', achieved a training and testing accuracy of 84%. It shows balanced precision for both classes but lower recall for the minority class. The ROC curve with an AUC of 0.89 suggests robust performance.

- **Naive Bayes**

The Naive Bayes model achieved a training and testing accuracy of 87%. It performs well for the majority class (precision: 0.93) but has lower recall for the minority class. The ROC curve with an AUC of 0.83 indicates moderate performance.

- **Random Forest**

The Random Forest model, with optimal parameters max_depth=None and n_estimators=20, obtained a training and testing accuracy of 86%. It shows balanced precision for both classes but lower recall for the minority class. The ROC curve with an AUC of 0.79 indicates moderate discriminative ability.

# Overall Conclusion SECTION 1 C

Conclusion Among the evaluated models, Logistic Regression and SVM demonstrated strong performance with high AUC values of 0.89, making them reliable choices for prediction. Naive Bayes and Random Forest showed moderate performance, while KNN had lower discriminative ability with an AUC of 0.82. Overall, Logistic Regression and SVM are the most suitable models for prediction in this scenario.

# Section 1 D : Class Imbalance Technique Threshold

- In this section i trained ML model with data normalization technique minmax, feature selection using Chi Square with CV and for handling class imbalance issue i use Threshold techniques

# Result and Insight Explanation

- **K-Nearest Neighbors (KNN)**

The KNN model, with cross-validation and a threshold of 0.5, achieved a training and testing accuracy of 89%. It shows high precision for the majority class (0.92) but lower precision for the minority class (0.50). The ROC curve indicates a moderate discriminative ability with an AUC of 0.80.

- **Logistic Regression**

Logistic Regression, with cross-validation and a threshold of 0.5, obtained a training and testing accuracy of 90%. It demonstrates high precision for the majority class (0.92) and better precision and recall for the minority class (0.62 and 0.30, respectively). The ROC curve with an AUC of 0.88 suggests robust performance.

- **Support Vector Machine (SVM)**

The SVM model, with cross-validation and a threshold of 0.5, achieved a training and testing accuracy of 90%. It shows high precision for the majority class (0.90) but lower recall for the minority class (0.17). The ROC curve with an AUC of 0.83 indicates moderate performance.

- **Naive Bayes**

The Naive Bayes model achieved a training and testing accuracy of 87%. It performs well for the majority class (precision: 0.93) but has lower recall for the minority class. The ROC curve with an AUC of 0.83 indicates moderate performance.

- **Random Forest**

The Random Forest model, with cross-validation and a threshold of 0.5, obtained a training and testing accuracy of 88%. It shows balanced precision for both classes but lower recall for the minority class. The ROC curve with an AUC of 0.83 indicates moderate discriminative ability.

# Overall Conclusion SECTION 1 D

- Logistic Regression and SVM show the highest testing accuracy of 90%, with Logistic Regression having a slightly better balance in performance across classes. Naive Bayes and Random Forest demonstrate moderate performance, while KNN has a lower AUC of 0.80. Overall, Logistic Regression and SVM are the most suitable models for prediction in this scenario.

# Section 2 A : Simple ML Model

# Dataset 2 : Default of Credit Card Clients

- Dataset URL : https://archive.ics.uci.edu/dataset/350/default+of+credit+card+clients
- Dataset Information : 29999 rows and 24 columns with Two class

# Result and Insight Explanation

- In this dataset the ratio of class imabalance are : 0.20%
- K-Nearest Neighbors (KNN)

The KNN model has a training accuracy of 82% and a testing accuracy of 76%. It shows high precision (0.80) and recall (0.92) for the majority class but low performance for the minority class (precision: 0.38, recall: 0.19). The ROC curve has an AUC of 0.61, indicating moderate discriminative ability.

- **Logistic Regression**

Logistic Regression has a training and testing accuracy of 78%. It demonstrates high precision and recall for the majority class (0.78 and 1.00, respectively) but fails to predict the minority class (precision and recall: 0.00). The ROC curve has an AUC of 0.66, indicating moderate performance.

- **Support Vector Machine (SVM)**

The SVM model achieved a training and testing accuracy of 78%. Similar to Logistic Regression, it performs well for the majority class (precision and recall: 0.78 and 1.00) but poorly for the minority class (precision and recall: 0.00). The ROC curve shows an AUC of 0.58, suggesting lower performance.

- **Naive Bayes**

The Naive Bayes model has a training and testing accuracy of 38%. It performs better for the minority class (precision: 0.25, recall: 0.88) compared to other models but has a low overall accuracy. The ROC curve has an AUC of 0.67, indicating moderate performance.

- **Random Forest**

The Random Forest model has a training accuracy of 100% and a testing accuracy of 81%. It shows high precision and recall for the majority class (0.84 and 0.94) and better performance for the minority class (precision: 0.63, recall: 0.35). The ROC curve with an AUC of 0.76 indicates strong performance.

# Overall conclusion SECTION 2 A

Among the models, Random Forest demonstrates the best overall performance with a testing accuracy of 81% and an AUC of 0.76. Logistic Regression and SVM show moderate performance but fail to predict the minority class effectively. Naive Bayes has a lower overall accuracy but performs better on the minority class. KNN has a moderate performance with an AUC of 0.61. Overall, Random Forest is the most suitable model for prediction in this scenario.

# Section 2 B : CI Technique Resampling Smooth

- In this section i trained ML model with data normalization technique minmax, feature selection using Chi Square with hyperparameter tunning grid search and for handling class imbalance issue i use resampling smooth techniques

# Result and Insight Explanation

- K-Nearest Neighbors (KNN)

The KNN model, with optimal parameters n_neighbors=3 and weights='distance', achieved a training accuracy of 78% and a testing accuracy of 80%. It shows high precision (0.85) and recall (0.72) for the majority class, and better performance for the minority class (precision: 0.76, recall: 0.87). The ROC curve has an AUC of 0.86, indicating strong discriminative ability.

- Logistic Regression

Logistic Regression, with optimal parameters C=1.0 and solver='liblinear', achieved a training accuracy of 68% and a testing accuracy of 66%. It demonstrates balanced precision and recall for both classes but overall lower performance compared to other models. The ROC curve with an AUC of 0.72 indicates moderate performance.

- Support Vector Machine (SVM)

The SVM model, with optimal parameters C=10.0 and kernel='rbf', achieved a training accuracy of 71% and a testing accuracy of 70%. It shows good precision and recall for both classes, with an overall balanced performance. The ROC curve has an AUC of 0.75, suggesting good discriminative power.

- Naive Bayes

The Naive Bayes model achieved a training and testing accuracy of 58%. It performs better for the minority class (precision: 0.55, recall: 0.91) compared to the majority class. The ROC curve has an AUC of 0.73, indicating moderate performance.

- Random Forest

The Random Forest model, with optimal parameters max_depth=None and n_estimators=200, achieved a training accuracy of 83% and a testing accuracy of 84%. It shows high precision and recall for both classes, with strong overall performance. The ROC curve with an AUC of 0.91 indicates excellent discriminative ability.

# Over all Conclusion section 2 B

- Among the evaluated models, Random Forest demonstrated the best overall performance with a testing accuracy of 84% and the highest AUC of 0.91. KNN also performed well, with strong discriminative ability and balanced performance. Logistic Regression, SVM, and Naive Bayes showed moderate performance, with SVM slightly better among them. Overall, Random Forest is the most suitable model for prediction in this scenario.

# Section 2 C : CI Technique Class Weight

- In this section i trained ML model with data normalization technique minmax, feature selection using Chi Square with CV and for handling class imbalance issue i use CLASS weight techniques

# Result and Insight Explanation

- K-Nearest Neighbors (KNN)

The KNN model achieved a training accuracy of 80% and a testing accuracy of 79%. It performs well for the majority class (precision: 0.84, recall: 0.91) but shows lower performance for the minority class (precision: 0.53, recall: 0.37). The ROC curve has an AUC of 0.70, indicating moderate discriminative ability.

- Logistic Regression

Logistic Regression achieved a training and testing accuracy of 68%. It demonstrates balanced performance with a precision of 0.88 and recall of 0.68 for the majority class, but lower performance for the minority class (precision: 0.37, recall: 0.66). The ROC curve has an AUC of 0.72, indicating moderate performance.

- Support Vector Machine (SVM)

The SVM model achieved a training accuracy of 77% and a testing accuracy of 78%. It shows good performance for the majority class (precision: 0.88, recall: 0.83) and moderate performance for the minority class (precision: 0.49, recall: 0.58). The ROC curve has an AUC of 0.75, suggesting good discriminative power.

- Naive Bayes

The Naive Bayes model achieved a training and testing accuracy of 72%. It performs well for the majority class (precision: 0.88, recall: 0.75) and shows moderate performance for the minority class (precision: 0.42, recall: 0.64). The ROC curve has an AUC of 0.73, indicating moderate performance.

- Random Forest

The Random Forest model achieved a training accuracy of 81% and a testing accuracy of 81%. It shows strong performance for both classes, with precision and recall of 0.84 and 0.93 for the majority class, and 0.61 and 0.37 for the minority class. The ROC curve has an AUC of 0.75, indicating good performance.

# Overall Conclusion Section 2 C

- Among the models, Random Forest demonstrates the best overall performance with a testing accuracy of 81% and strong discriminative ability. SVM also shows good performance with a balanced accuracy and higher AUC of 0.75. Logistic Regression and KNN show moderate performance, with Naive Bayes slightly behind. Overall, Random Forest and SVM are the most suitable models for prediction in this scenario.

# Section 2 D : Class Imbalance Technique Threshold

- In this section i trained ML model with data normalization technique minmax, feature selection using Chi Square with CV and for handling class imbalance issue i use Threshold techniques

# Result and Insight Explanation

- K-Nearest Neighbors (KNN)

The KNN model achieved an average training accuracy of 80% and a testing accuracy of 79%. It performs well for the majority class with a precision of 0.84 and recall of 0.91 but has lower performance for the minority class (precision: 0.53, recall: 0.37). The model demonstrates an overall balanced performance with a weighted average f1-score of 0.78.

- Logistic Regression

Logistic Regression achieved an average training accuracy of 81% and a testing accuracy of 81%. It shows high precision (0.82) and recall (0.98) for the majority class but lower performance for the minority class (precision: 0.72, recall: 0.22). The model exhibits good overall performance with a weighted average f1-score of 0.77.

- Support Vector Machine (SVM)

The SVM model achieved an average training accuracy of 82% and a testing accuracy of 82%. It shows high precision (0.83) and recall (0.96) for the majority class, but lower performance for the minority class (precision: 0.69, recall: 0.32). The model demonstrates balanced performance with a weighted average f1-score of 0.79.

- Naive Bayes

The Naive Bayes model achieved an average training and testing accuracy of 72%. It performs well for the majority class with a precision of 0.88 and recall of 0.75 but shows moderate performance for the minority class (precision: 0.42, recall: 0.64). The model has a weighted average f1-score of 0.74, indicating moderate overall performance.

- Random Forest

The Random Forest model achieved an average training accuracy of 81% and a testing accuracy of 82%. It shows high precision (0.85) and recall (0.93) for the majority class, and moderate performance for the minority class (precision: 0.63, recall: 0.40). The model exhibits balanced performance with a weighted average f1-score of 0.80.

# Overall Conclusion Section 2 D

- Among the evaluated models, Random Forest and SVM demonstrate the best overall performance with testing accuracies of 82%. Logistic Regression also performs well with a testing accuracy of 81%. KNN and Naive Bayes show moderate performance, with KNN having a slightly higher accuracy. Overall, Random Forest and SVM are the most suitable models for prediction in this scenario due to their balanced performance and high accuracy.

# Section 3 A : Generalize ML Model

# Dataset 3 : Customer Churn

- Dataset URL : https://www.kaggle.com/datasets/hassanamin/customer-churn
- Dataset Information : 900 rows and 10 columns with Two class

# Result and Insight Explanation

class imbalance ratio 0.2

- K-Nearest Neighbors (KNN)

The KNN model shows a training accuracy of 84% and a testing accuracy of 81%. It performs well on the majority class (precision: 0.82, recall: 0.98) but fails completely on the minority class (precision: 0.00, recall: 0.00). The ROC curve indicates poor overall performance with an AUC of 0.50.

- Logistic Regression

Logistic Regression achieves a training accuracy of 85% and a testing accuracy of 82%. It has good performance for the majority class (precision: 0.84, recall: 0.97), but poor performance for the minority class (precision: 0.50, recall: 0.16). The ROC curve indicates moderate performance with an AUC of 0.69.

- Support Vector Machine (SVM)

The SVM model achieves a training accuracy of 84% and a testing accuracy of 82%. Similar to KNN, it performs well for the majority class (precision: 0.82, recall: 1.00) but fails on the minority class (precision: 0.00, recall: 0.00). The ROC curve shows poor overall performance with an AUC of 0.59.

- Naive Bayes

Naive Bayes shows a training accuracy of 89% and a testing accuracy of 89%. It performs well for both classes, especially the minority class (precision: 0.76, recall: 0.59), making it a balanced model. The ROC curve indicates good overall performance with an AUC of 0.89.

- Random Forest

Random Forest achieves a training accuracy of 100% and a testing accuracy of 87%. It performs well for the majority class (precision: 0.88, recall: 0.97) and reasonably well for the minority class (precision: 0.72, recall: 0.41). The ROC curve indicates strong performance with an AUC of 0.86.

# Overall Conclusion section 3 A

Naive Bayes and Random Forest show the best performance among the models, with Naive Bayes having a slightly better AUC of 0.89. Both models are balanced, but Random Forest is more robust for larger datasets. Logistic Regression and SVM provide moderate performance, while KNN struggles significantly with the minority class. For the given dataset, Naive Bayes or Random Forest would be the most reliable choices for prediction.

# Section 3 B : Class Imbalance Technique RESAMPLING Smooth

- In this section i trained ML model with data normalization technique minmax, feature selection using Chi Square with hyper paramenter gridCV and for handling class imbalance issue i use smooth techniques

# Result and Insight Explanation

- K-Nearest Neighbors (KNN)

The K-Nearest Neighbors (KNN) model, tuned with 5 neighbors and distance-based weighting, achieved a training accuracy of 0.90 and a testing accuracy of 0.90. The model shows high precision for class 0 (0.97) and good recall for class 1 (0.97), with an overall ROC AUC of 0.96, indicating strong discriminative power.

- Logistic Regression

The Logistic Regression model, with parameters C=10.0 and solver='lbfgs', resulted in a training accuracy of 0.84 and a testing accuracy of 0.85. It has balanced precision and recall values, with precision of 0.82 and 0.89 for classes 0 and 1, respectively. The ROC curve's AUC of 0.92 reflects solid classification performance.

- Support Vector Machine (SVM)

The SVM model, using a radial basis function (RBF) kernel and C=1.0, achieved a training accuracy of 0.85 and a testing accuracy of 0.86. The model maintains a good balance with precision of 0.82 and 0.89 for classes 0 and 1, respectively. The ROC curve's AUC of 0.94 indicates robust discriminative power.

- Naive Bayes

The Naive Bayes model, with default parameters, achieved a training accuracy of 0.84 and a testing accuracy of 0.84. It shows precision of 0.81 for class 0 and 0.87 for class 1, with an ROC AUC of 0.92, indicating competent performance in classification tasks.

- Random Forest

The Random Forest model, tuned with no maximum depth and 50 estimators, achieved a training accuracy of 0.91 and a testing accuracy of 0.93. The model demonstrates high precision and recall for both classes, with an overall ROC AUC of 0.98, indicating exceptional discriminative ability.

# overall conclusion section 3 B

- Among the evaluated models, the Random Forest classifier stands out with the highest testing accuracy of 0.93 and an AUC of 0.98, making it the most reliable for this classification task. The KNN model also performs robustly with a testing accuracy of 0.90 and an AUC of 0.96. Logistic Regression, SVM, and Naive Bayes show solid performance, with testing accuracies around 0.84-0.86 and AUCs between 0.92 and 0.94. Overall, Random Forest is the most effective model, followed by KNN, while Logistic Regression, SVM, and Naive Bayes provide competent alternatives

# Section 3 C : Class Imbalance Technique Class weight

- In this section i trained ML model with data normalization technique minmax, feature selection using Chi Square with CV and for handling class imbalance issue i use Class weight techniques

# Result and Insight Explanation

-K-Nearest Neighbors (KNN)

The K-Nearest Neighbors (KNN) model, with an average training accuracy of 0.87 and testing accuracy of 0.89, shows good performance with a high precision of 0.92 for class 0 but lower precision of 0.62 for class 1. The ROC curve's AUC of 0.89 reflects decent discriminative ability.

- Logistic Regression

The Logistic Regression model achieved an average training accuracy of 0.81 and testing accuracy of 0.82. It has high precision for class 0 (0.98) but low precision for class 1 (0.42). The ROC curve's AUC of 0.93 indicates strong classification performance.

- Support Vector Machine (SVM)

The SVM model, with an average training accuracy of 0.83 and testing accuracy of 0.86, demonstrates high precision (0.96 for class 0) but moderate recall for class 1 (0.80). The ROC curve's AUC of 0.93 suggests robust discriminative power.

- Naive Bayes

The Naive Bayes model, with an average training accuracy of 0.89 and testing accuracy of 0.90, shows balanced performance with high precision for class 0 (0.93) and reasonable precision for class 1 (0.67). The ROC curve's AUC of 0.93 signifies good classification ability.

- Random Forest

The Random Forest model achieved an average training accuracy of 0.88 and testing accuracy of 0.88. It has high precision for class 0 (0.92) but lower precision for class 1 (0.59). The ROC curve's AUC of 0.90 reflects solid discriminative performance.

# Overall Conclusion SECTION 3 C

- Among the evaluated models, the Naive Bayes and Logistic Regression classifiers show the best overall performance with testing accuracies of 0.90 and 0.82, and AUCs of 0.93. The SVM model also performs well with an accuracy of 0.86 and an AUC of 0.93. The KNN and Random Forest models are competent, with accuracies of 0.89 and 0.88, and AUCs of 0.89 and 0.90, respectively. Overall, Naive Bayes and Logistic Regression are the most reliable models in this classification task.

# Section 3 C : Class Imbalance Technique threshold

In this section i trained ML model with data normalization technique minmax, feature selection using Chi Square with CV and for handling class imbalance issue i use threshold techniques

# Result and Insight Explanation

- K-Nearest Neighbors (KNN)

The K-Nearest Neighbors (KNN) model achieved an average training accuracy of 0.87 and a testing accuracy of 0.89. It shows high precision of 0.92 for class 0 but lower precision of 0.62 for class 1. The ROC curve's AUC of 0.89 indicates decent discriminative ability.

- Logistic Regression

The Logistic Regression model obtained an average training accuracy of 0.84 and a testing accuracy of 0.91. It exhibits high precision for class 0 (0.91) and class 1 (0.90), although recall for class 1 is lower (0.36). The ROC curve's AUC of 0.93 reflects strong classification performance.

- Support Vector Machine (SVM)

The SVM model, with an average training accuracy of 0.88 and testing accuracy of 0.90, demonstrates high precision for both classes (0.93 and 0.67) and a good balance in performance. The ROC curve's AUC of 0.93 suggests robust discriminative power.

- Naive Bayes

The Naive Bayes model, with an average training accuracy of 0.89 and testing accuracy of 0.90, shows balanced performance with high precision for class 0 (0.93) and moderate precision for class 1 (0.67). The ROC curve's AUC of 0.93 indicates good classification ability.

- Random Forest

The Random Forest model achieved an average training accuracy of 0.86 and a testing accuracy of 0.88. It has high precision for class 0 (0.93) but lower precision for class 1 (0.56). The ROC curve's AUC of 0.90 reflects solid discriminative performance.

# Overall Conclusion Section 3 D

- Among the evaluated models, Logistic Regression and Naive Bayes show the best overall performance with testing accuracies of 0.91 and 0.90, and AUCs of 0.93. The SVM model also performs well with an accuracy of 0.90 and an AUC of 0.93. The KNN and Random Forest models are competent, with accuracies of 0.89 and 0.88, and AUCs of 0.89 and 0.90, respectively. Overall, Logistic Regression and Naive Bayes are the most reliable models in this classification task.

# OVERALL PROJECT SUMMARY

**After implementation all different class imbalance technique with different ML algorithm, I analyze that in most of case CI resampling technique work very good, if we compare with different CI technique like class weight, threshold etc and from ML algorithm random forest work very good and KNN also slightly better, both algorithm has also own ability to work or handle class imbalance techniques**