UNIVERSITÀ
di VERONA

Machine Learning Project Report

- **Project Title:** Predicting Bank Term Deposit Subscription
- **Master Degree:** Artificial Intelligence
- **Course:** Machine Learning and Deep Learning
- **Academic Year:** 2024/2025
- **Matricula Number:** VR528190
- **Author:** Shah Rukh Aleem

Table of Contents

1. **Motivation and Rationale**
2. **State of the Art**
3. **Objectives**
4. **Methodology**
    - Dataset Description
    - Data Preprocessing
    - Model Selection and Training
5. **Experiments and Results**
    - Evaluation Metrics
    - Results
    - Discussion
6. **Conclusions**
7. **Bibliography**
8. **Appendix**

1. Motivation and Rationale

Predicting customer behavior has become a critical component of financial marketing plans. The project will define the prediction of whether a customer will choose a bank's term deposit. Such predictions give power to the banks to design specific marketing campaigns, allocate resources appropriately, and increase conversion rates. The motivation for this project is to use data-driven insights to improve decision-making and reduce costs linked to ineffective outreach activities. By identifying those customers who have a high probability of subscribing, it will further enhance campaign effectiveness while maintaining ethical and responsible marketing principles.

2. State of the Art

Machine learning became a basic tool in the financial services industry for predictive analytics. Ensemble methods like Random Forest and Gradient Boosting are favored for their resistance to overfitting and effectiveness with complex data, especially when compared to simpler methods like logistic regression in binary classification. Though the effectiveness of these algorithms is well documented in the literature, issues such as imbalanced data, feature interpretability, and threshold optimization are yet to be properly explored. This initiative updates the state of affairs by merging Random Forest with advanced techniques that focus on imbalanced classes and evaluate model performance in real-world situations.

3. Objectives

- **General Objective**: Create a robust and reliable machine learning model to predict customer subscriptions to bank term deposits.
- **Specific Objectives**:
    - Comprehend and preprocess the dataset to guarantee quality inputs for the modeling process.
    - Effectively handle missing data and encode categorical variables.
    - Use advanced methods to balance the dataset and address class imbalance.
    - Train a Random Forest classifier with optimized hyperparameters.
    - Assess the model's performance through various metrics and analyze feature importance for enhanced interpretability.

4. Methodology

Dataset Description

The dataset is sourced from the Bank Marketing Dataset available in the UCI Machine Learning Repository. It comprises 45,211 entries and 17 variables, with the target variable denoted as y, which indicates whether a customer has subscribed to a term deposit ('yes' or 'no'). The features encompass demographic data (such as age, occupation, and marital

status), financial information (including account balance), and campaign-specific variables (like contact duration and the number of previous contacts).

Data Preprocessing

1. **Handling Missing Values**:
    - Columns containing ”unknown” values, such as ‘job’ and ‘education’, were filtered to exclude rows with these entries.
    - Non-essential columns, including ‘contact’ and ‘poutcome’, were removed to because they had a large proportion of unknown values (28.79% and 81.74% respectively), which could introduce noise into the model.
2. **Encoding Categorical Data**:
    - Categorical variables (e.g., `job`, `marital`) were label encoded using sklearn’s LabelEncoder.
3. **Feature Scaling**:
    - Numerical features were standardized using StandardScaler to enhance model convergence.
4. **Handling Class Imbalance**:
    - Tomek Links, one of the under-sampling techniques, was used to remove the redundant instances of the majority class (‘No’ class), thus creating a more balanced training dataset.

Model Selection and Training

After trying out many algorithms such as Logistic Regression, K-Nearest Neighbors (KNN), and Support Vector Machine (SVM) with different distance measures, the Random Forest Classifier was found to be the best algorithm for this job. It is strong, can grow bigger easily, and is good at handling complicated data sets, making it the right help for this problem.

The hyperparameters listed below for Random Forest were fine-tuned using Grid Search and showed the top performance.

random_state=42, class_weight=’balanced’, max_depth=None, min_samples_split=2, n_estimators=300

These parameters effectively balanced the complexity of the model and its ability to generalize, resulting in enhanced performance metrics.

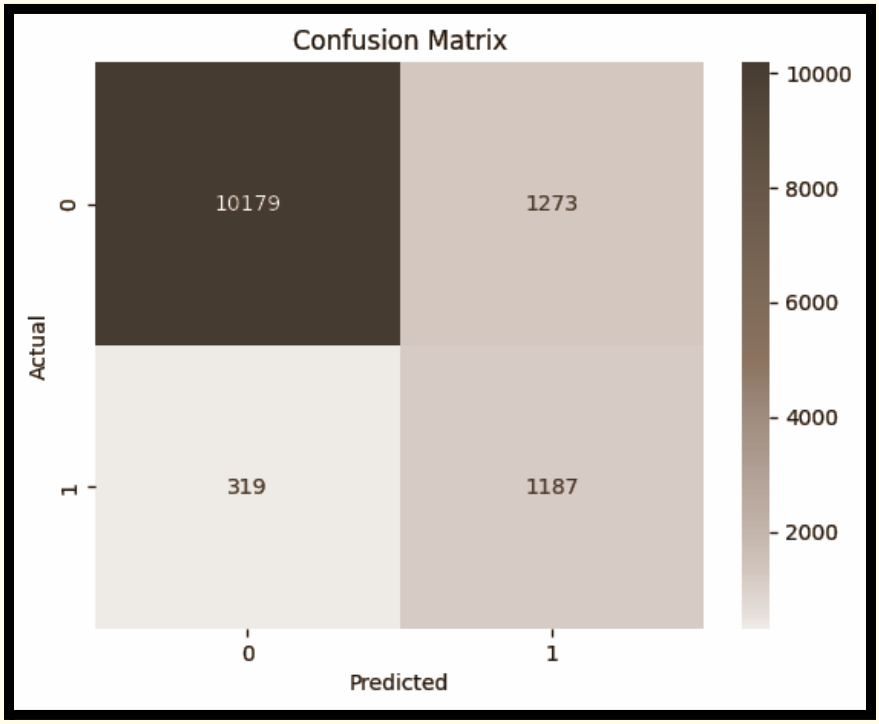
The choice of the Random Forest Classifier was based on its robustness, scalability, and capacity to offer insights into feature importance. The following hyperparameters were carefully adjusted:

- max_depth and min_samples_split: The default settings were retained to prioritize simplicity and generalization, while a threshold was adjusted from 0.5 to 0.25 to improve recall for the minority class. This allows the model to capture more potential 'Yes' cases, even at the cost of some precision.

5. Experiments and Results

Confusion Matrix

The confusion matrix provides the following insights into the model's classification performance:



As observed in the confusion matrix, it shows that the model correctly identifies the majority of instances for both categories.

Evaluation Metrics

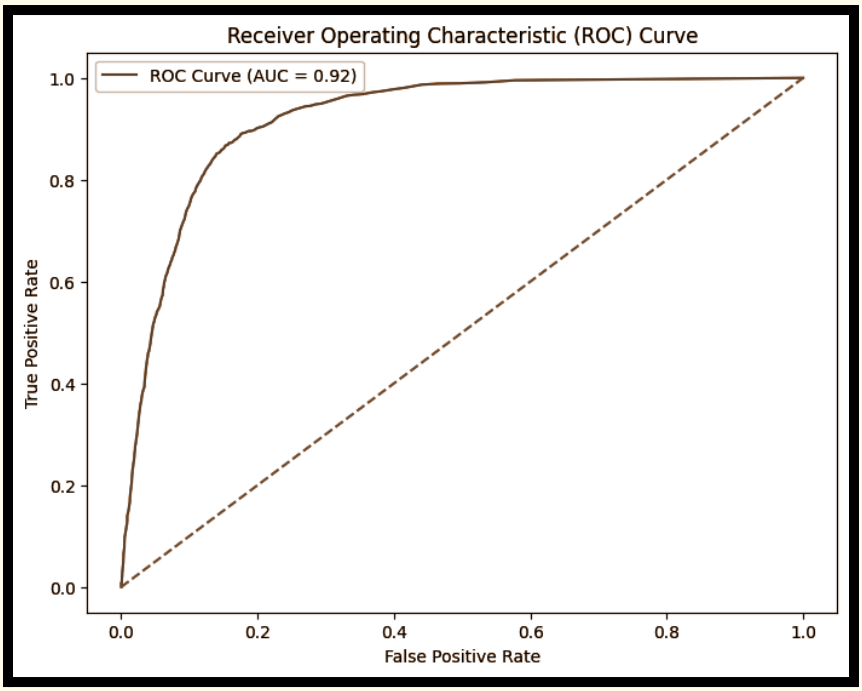
The performance of the model was measured using metrics like accuracy, precision, recall, F1-score, and ROC-AUC. These measures give a notion of the model's quality and the balance involved in its classification.

Results

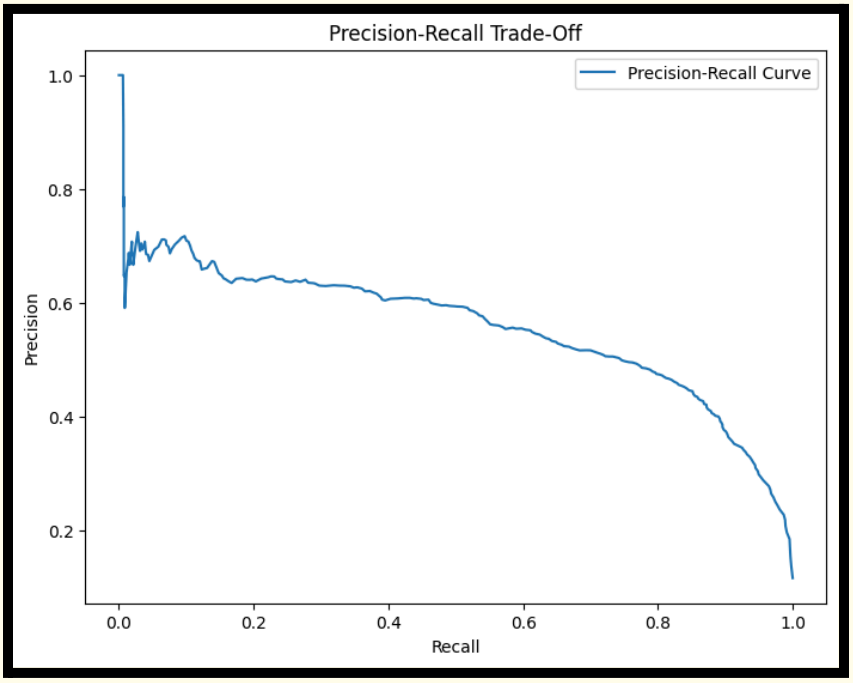
1. **Classification Report:**

	Precision	Recall	F1-score	Support
No	0.97	0.89	0.93	11452
Yes	0.48	0.79	0.60	1506
Accuracy			0.88	12958
Macro Avg	0.73	0.84	0.76	12958
Weighted Avg	0.91	0.88	0.89	12958

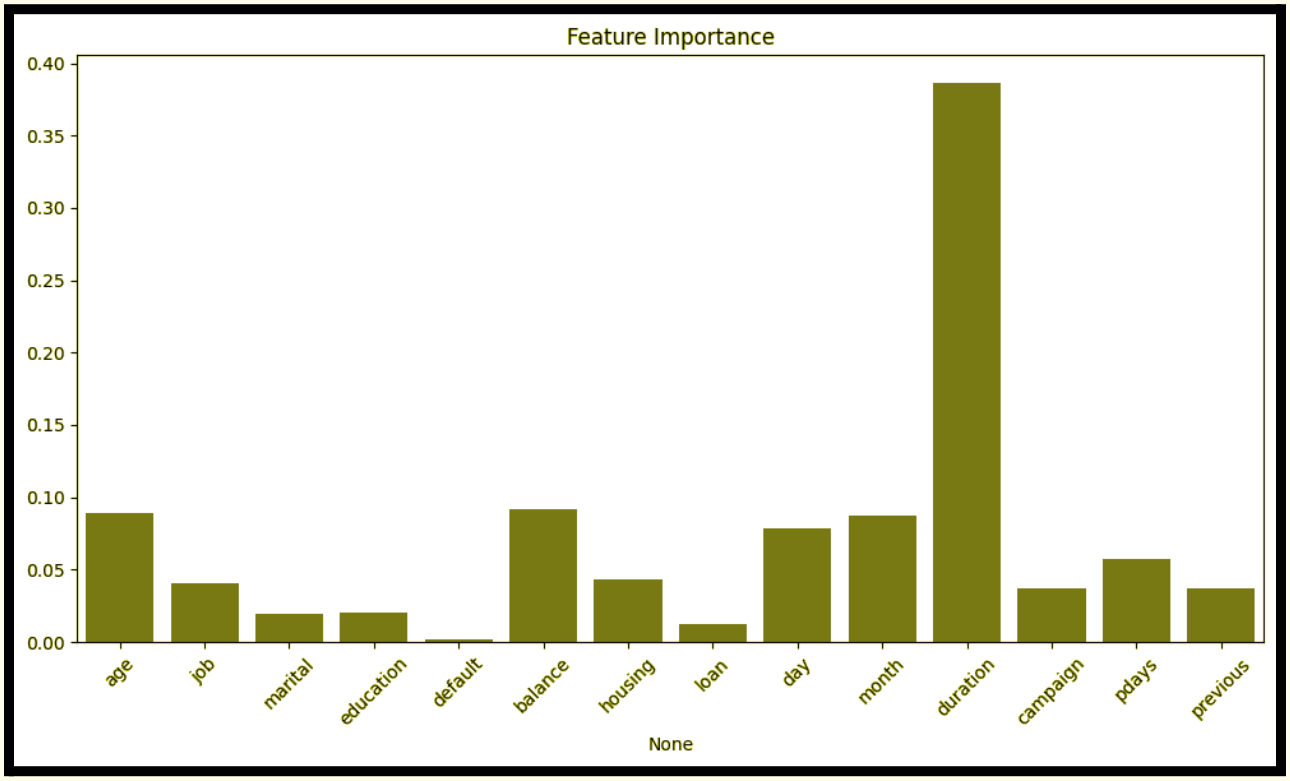
2. **Feature Importance:**
    - Key features: `duration` (38.6%), `balance`, `age`, and `month`.
    - The analysis of feature importance indicates that duration is the most critical predictor.
3. **Visualizations:**
    - **ROC Curve:** Demonstrated an AUC of 0.92, reflecting excellent discrimination capability.



- **Precision-Recall Curve**: Highlighted the trade-offs between precision and recall for different thresholds.



- **Feature Importance**: The most important feature, 'duration,' is a strong predictor but is only available after a customer interaction. For real-time prediction (before calling the customer), other features like balance, age, and previous campaign performance must be relied upon.



Discussion

The high ROC-AUC score demonstrates the model's proficiency in distinguishing between different classes. Modifying the decision threshold has enhanced the recall rate for the minority class, which is critical in addressing imbalanced datasets. Nevertheless, this adjustment has resulted in a decline in precision. Besides, the confusion matrix demonstrates that although the model successfully detects true positives, it still generates false negatives as a result of class imbalance. Future research could investigate advanced methodologies such as SMOTE for synthetic oversampling and evaluate alternative algorithms (e.g., XGBoost).

6. Conclusions

This project effectively applied machine learning techniques to predict customer subscription to term deposits. Key contributions include: Significance of preprocessing procedures such as addressing missing values and implementing data scaling. Effectiveness of Tomek Links in achieving dataset balance. Enhanced interpretability of Random Forest through feature importance assessment. Future endeavors may focus on refining feature engineering and investigating hybrid models for improved generalization. Future work may explore SMOTE oversampling and hybrid models like XGBoost to improve recall.

7. Bibliography

1. Lichman, M. (2013). UCI Machine Learning Repository.
2. Breiman, L. (2001). Random Forests. Machine Learning, 45(1), 5-32.
3. Pedregosa et al. (2011). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12, 2825-2830.
4. Tomek, I. (1976). Two Modifications of CNN. IEEE Transactions on Systems, Man, and Cybernetics.

8. Appendix

- Plots: ROC Curve, Precision-Recall Curve, Feature Importance.
- Code: Full implementation of data preprocessing, model training, and evaluation.