# Fraud Detection in Online Payments using Ensemble Learning

**Harshavardhan Anandasu**

***Abstract*—Fraud detection in financial transactions is a critical application of machine learning, requiring accurate identification of fraudulent activities amidst highly imbalanced datasets. This study evaluates the effectiveness of traditional and ensemble machine learning models for binary classification tasks. Using a dataset containing over 6.3 million transactions with only 0.13% fraudulent cases, we implemented Logistic Regression, Random Forest, XGBoost, and K-Nearest Neighbors (KNN). To mitigate class imbalance, preprocessing techniques like Synthetic Minority Oversampling Technique (SMOTE) and Random Undersampling were applied. Additionally, feature engineering introduced novel variables, improving the predictive power of the models. XGBoost emerged as the top performer, achieving a Receiver Operating Characteristic–Area Under Curve (ROC-AUC) score of 97% and a balanced F1-score, highlighting its ability to handle imbalance and deliver high accuracy. Random Forest also displayed strong recall for fraudulent cases, demonstrating the utility of ensemble techniques for fraud detection.**

**This study provides a comparative analysis of preprocessing, hyperparameter optimization, and model evaluation strategies. The findings underline the importance of ensemble models in handling imbalanced datasets and offer actionable insights into designing robust fraud detection systems. Future research could explore integrating deep learning methods or deploying models in real-time environments to enhance scalability and efficiency.**

## I. Introduction

The digital transformation of financial systems has ushered in a new era of convenience and efficiency but has also amplified vulnerabilities to fraudulent activities. The increasing volume and complexity of transactional data in these systems present substantial challenges for fraud detection. As fraudulent activities evolve, traditional rule-based detection systems struggle to keep pace, necessitating the adoption of machine learning (ML) techniques. ML models utilize data-driven insights to identify anomalous patterns indicative of fraud, thereby offering superior adaptability and precision.

One of the primary challenges in fraud detection using ML is the severe class imbalance, as fraudulent transactions typically account for only a small fraction of all transactions. This imbalance skews model performance, with many models optimizing for the majority class (legitimate transactions) at the expense of the minority class (fraudulent transactions). Consequently, such models may demonstrate high accuracy overall but fail to detect fraud effectively—a critical shortcoming in real-world applications.

Ensemble learning methods like Random Forest and XGBoost have emerged as promising solutions to this problem. These methods combine multiple classifiers to improve predictive accuracy and address class imbalance, leveraging techniques such as bagging and boosting to enhance the detection of minority class instances. Their ability to aggregate diverse perspectives makes ensemble models particularly robust for complex and imbalanced datasets. This study explores the comparative performance of traditional and ensemble learning techniques in fraud detection. Focusing on a highly imbalanced dataset, we examine preprocessing techniques, feature engineering strategies, and model evaluation metrics to identify the most effective approaches.

## II. Related Work

Fraud detection using machine learning has been extensively explored, with numerous studies investigating the efficacy of various approaches in addressing the challenges posed by fraudulent activities. Traditional models like Logistic Regression are often employed as baseline methods due to their interpretability and ease of implementation. However, these models exhibit significant limitations when applied to complex, nonlinear relationships within datasets. Furthermore, their performance deteriorates in the presence of severe class imbalance, where fraudulent transactions constitute a minor portion of the total dataset【8】【9】.

To overcome these challenges, ensemble methods, including Random Forest and XGBoost, have been widely adopted. Random Forest, a bagging-based algorithm, combines multiple decision trees to enhance robustness and mitigate overfitting. It has demonstrated considerable success in modeling diverse features, particularly within high-dimensional financial datasets【3】. XGBoost, on the other hand, employs gradient boosting to optimize predictive performance. Its inclusion of regularization mechanisms and parallelized computations has been highlighted as key factors contributing to its superior accuracy in fraud detection【2】【5】【6】.

Class imbalance remains a critical issue in fraud detection, as models often exhibit bias toward the majority class, leading to poor recall for the minority (fraudulent) class. Techniques like Synthetic Minority Oversampling Technique (SMOTE) and undersampling have been proposed to address this imbalance. SMOTE enhances model sensitivity by generating synthetic data points for the minority class, whereas undersampling reduces the size of the majority class to balance the dataset. Studies emphasize that these preprocessing methods, when

combined with ensemble models, significantly improve performance metrics such as recall and precision【11】.

The role of feature engineering has also been underscored in recent studies. Transaction-specific variables, such as balance changes, transaction frequency, and contextual attributes, have been identified as critical features for fraud detection models. Effective feature engineering has been shown to enhance the discriminatory power of machine learning models, particularly when coupled with advanced ensemble techniques【3】【7】.

Research comparing traditional and ensemble approaches has consistently demonstrated the advantages of ensemble methods. For instance, studies have highlighted XGBoost's ability to handle missing values, optimize decision boundaries, and achieve superior generalization. These attributes make it a preferred choice for fraud detection tasks【5】【6】. Moreover, ensemble methods outperform traditional models in handling the inherent complexities of imbalanced datasets, reinforcing their value in practical applications【6】【7】.

This literature review provides a foundation for understanding the comparative strengths and limitations of traditional and ensemble methods in fraud detection. Building on these insights, this study aims to further investigate and quantify the performance of these approaches, offering guidance on best practices for real-world applications.

## III. Dataset Description

The dataset used in this study was sourced from Kaggle and consists of 6,362,620 financial transactions. It includes both numerical and categorical features that capture key aspects of transaction activity. Among the 11 features, critical variables such as transaction type, transaction amount, and various balance details are included. These features provide insights into transaction behavior, which is essential for distinguishing between fraudulent and legitimate transactions.

The target variable in the dataset is isFraud, where a value of 1 indicates a fraudulent transaction and 0 denotes a legitimate transaction. One of the major challenges in this dataset is the highly imbalanced class distribution, with fraudulent transactions accounting for only 0.13% of the total transactions. The vast majority, 99.87%, are legitimate transactions. This severe class imbalance poses a significant challenge for machine learning models, as they tend to be biased toward the majority class, often failing to identify the minority class (fraudulent transactions) effectively.

Data quality is crucial for the accuracy of machine learning models. In this dataset, no missing values or duplicates were found, which ensured that the data was complete and reliable. Some inconsistencies in balance records were rectified through recalculating balance-related features, ensuring the dataset's consistency. Additionally, new features were created to enhance the predictive capabilities of the model.

## IV. Methodology

### 4.1 Preprocessing

Preprocessing steps are crucial to ensure the quality of the data and improve the performance of machine learning models. Initially, the columns in the dataset were renamed for clarity to ensure consistency in naming conventions. Data inconsistencies, particularly in balance-related features, were addressed through recalculation to ensure accuracy.

Feature engineering played a key role in enhancing the dataset. One significant addition was the creation of the isCustomerToCustomer feature, which distinguishes transactions between customers from other types. This new variable helped capture crucial transactional patterns, which might be essential for identifying fraudulent activities【7】.

To address the severe class imbalance in the dataset, several strategies were implemented. **SMOTE (Synthetic Minority Oversampling Technique)** was used to artificially increase the number of fraud cases in the dataset, allowing the model to focus more on the minority class【11】. Additionally, **Random Undersampling** was employed to reduce the number of legitimate transactions, ensuring that the dataset was more balanced for training. This combination of oversampling and undersampling techniques helped mitigate the model's bias toward the majority class and allowed for better detection of fraudulent transactions.

### 4.2 Models Implemented

The study implemented several machine learning models to compare their performance in fraud detection:

- **Logistic Regression**: This model served as the baseline, offering a simple linear approach to binary classification. It is interpretable but may struggle with nonlinearities and class imbalance【8】【9】.
- **Random Forest**: This ensemble learning method employs bagging to build multiple decision trees, increasing stability and accuracy by reducing overfitting【4】【5】.
- **XGBoost**: Known for its high performance, this gradient boosting model is optimized for handling imbalanced datasets. It combines multiple weak learners and uses regularization techniques for better generalization【3】【6】.
- **K-Nearest Neighbors (KNN)**: A non-parametric model that classifies based on the proximity of data points. KNN is simple but computationally expensive, especially for large datasets【12】.

### 4.3 Hyperparameter Tuning

To improve the performance of the models, hyperparameter tuning was conducted using **Grid Search** and **Randomized Search** techniques. Both methods, combined with **Stratified K-Fold Cross-Validation**, were used to select optimal values for critical parameters such as the learning rate, tree depth, and the number of estimators【10】.

### 4.4 Evaluation Metrics

To evaluate model performance, several metrics were used, focusing particularly on the minority class (fraudulent transactions):

- **Accuracy**: Measures overall correctness, but may be misleading in imbalanced datasets.
- **Precision, Recall, and F1-Score**: These metrics focus on how well the model performs on the minority class (fraud cases), with the F1-score providing a balanced measure between precision and recall.
- **ROC-AUC**: Assesses the model's ability to discriminate between fraudulent and non-fraudulent transactions, with a higher value indicating better performance【1】【2】.

## V. Experimental Results

This section evaluates the performance of the machine learning models used to detect fraudulent transactions. It includes a detailed analysis of the models' effectiveness, supported by various performance metrics and visualizations to provide insights into model performance, feature relevance, and classification behavior.

### 5.1 Performance Metrics

The performance of four machine learning models—Logistic Regression, Random Forest, XGBoost, and K-Nearest Neighbors (KNN)—is assessed using the following metrics:

- **ROC-AUC (Receiver Operating Characteristic – Area Under Curve)**: This metric reflects the model's ability to differentiate between fraudulent and non-fraudulent transactions.
- **Precision**: Measures the fraction of correctly predicted fraud cases among all instances predicted as fraud.
- **Recall**: Indicates the proportion of actual fraud cases correctly identified by the model.
- **F1-Score**: A balanced metric that combines precision and recall, providing a single performance indicator.
- **Matthews Correlation Coefficient (MCC)**: Combines all elements of the confusion matrix into a balanced metric, particularly valuable for imbalanced datasets.

The comparative results for these models are summarized in **Table 1**:

| Model | ROC-AUC | Precision | Recall | F1-Score | MCC |
|---|---|---|---|---|---|
| Logistic Regression | 0.84 | 0.08 | 0.33 | 0.13 | 0.15 |
| Random Forest | 0.95 | 0.10 | 0.79 | 0.15 | 0.25 |
| XGBoost | 0.97 | 0.18 | 0.73 | 0.29 | 0.36 |
| K-Nearest Neighbors | 0.91 | 0.14 | 0.66 | 0.23 | 0.30 |

**VI. Key Observations:**

- **XGBoost** demonstrates the highest performance across all metrics, with the highest ROC-AUC (0.97) and MCC (0.36), achieving a balanced F1-score of 0.29.
- **Random Forest** stands out in terms of recall (0.79), highlighting its strong ability to identify fraud cases. However, it has lower precision.
- **KNN** offers moderate performance across all metrics, achieving a balanced recall (0.66) and MCC (0.30).
- **Logistic Regression** performs the worst, as expected due to its linear nature, which is less suited for capturing the complexities of the fraud detection dataset.

### 5.2 Visualizations

Various visualizations are employed to provide deeper insights into model performance and dataset characteristics. These visualizations aid in understanding the critical features and patterns in the data that influence the model's predictions.

#### 5.2.1 Confusion Matrices

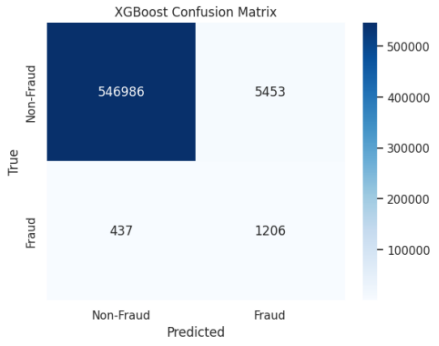
Confusion matrices display the breakdown of predictions into true positives, false positives, true negatives, and false negatives. These matrices provide insights into the trade-offs between fraud detection and the avoidance of false alarms.

**XGBoost Confusion Matrix**:

- **True Positives**: 1,206 (Fraud correctly predicted)
- **False Positives**: 5,453 (Non-fraud predicted as fraud)
- **True Negatives**: 546,986 (Non-fraud correctly predicted)
- **False Negatives**: 437 (Fraud predicted as non-fraud)

XGBoost achieves high recall (73%) and precision, making it a robust model for fraud detection.

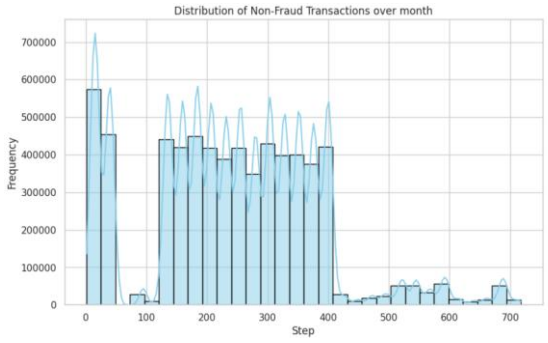
**Figure 1**: Confusion Matrix for XGBoost




#### 5.2.3 Distribution of Non-Fraud Transactions Over Time

The distribution of non-fraud transactions over time (steps) reveals a uniform pattern, with no significant spikes or trends. Figure 3 illustrates that legitimate transactions are evenly distributed across the timeline, indicating no time-related bias in non-fraud activities.

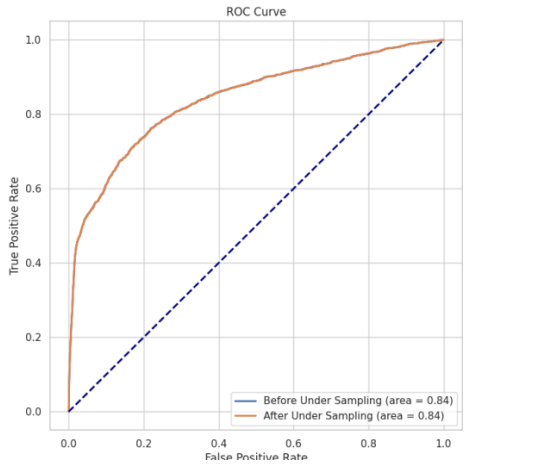
Figure 3: **Distribution of Non-Fraud Transactions Over Time**

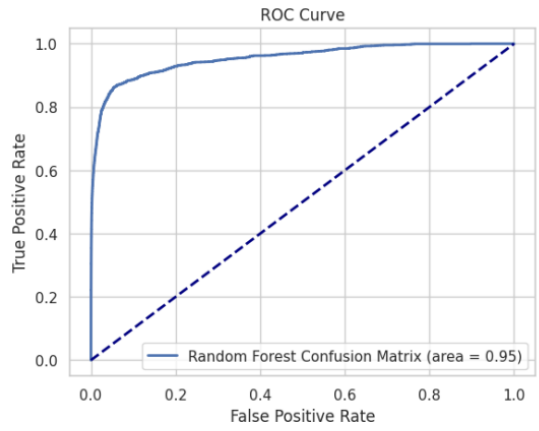



### 5.2.3 ROC Curves

ROC curves plot the true positive rate (TPR) against the false positive rate (FPR) across different classification thresholds. These curves illustrate the model's ability to discriminate between fraudulent and non-fraudulent transactions.

**Figure 3**: ROC Curves for Logistic Regression, Random Forest, XGBoost, and KNN. XGBoost achieves the highest area under the curve (AUC = 0.97), followed by Random Forest (AUC = 0.95).

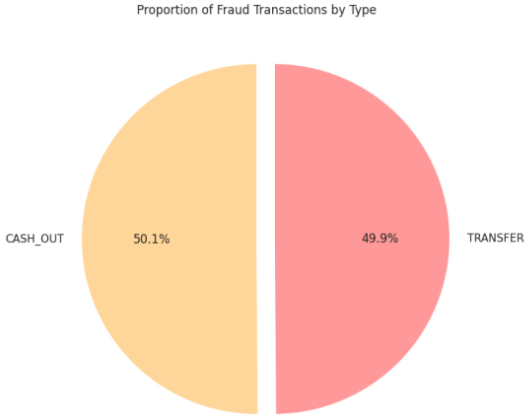



### 5.2.4 Transaction Type Analysis

A pie chart illustrates the proportion of each transaction type in the dataset and its correlation with fraudulent activities. **Figure 4** shows that fraud is primarily concentrated in **TRANSFER** and **CASH_OUT** transactions, each contributing 50% of the fraud cases. This highlights the susceptibility of these transaction types to fraud.

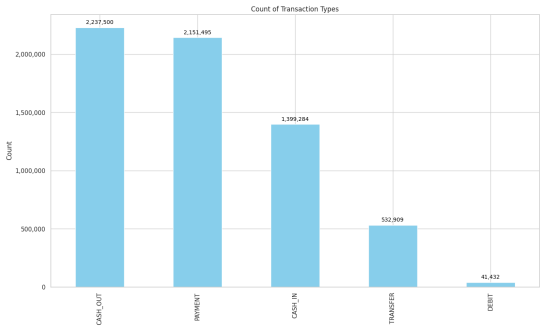
**Figure 4**: Fraud Transactions by Type




### 5.2.5 Count of Transaction Types

This visualization presents the count of each transaction type in the dataset. **Figure 5** shows that **CASH_OUT** and **PAYMENT** transactions dominate, accounting for about 70% of all transactions, with **TRANSFER** and **CASH_IN** making up smaller portions.

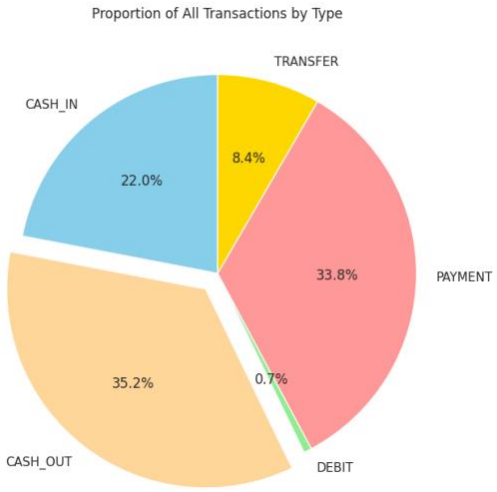
**Figure 5**: Count of Transaction Types




### 5.2.6 Proportion of All Transactions by Type

A pie chart illustrating the proportional distribution of all transaction types shows that **CASH_OUT** takes the largest share, followed by **PAYMENT**. This distribution provides context for identifying fraud-prone transaction types.

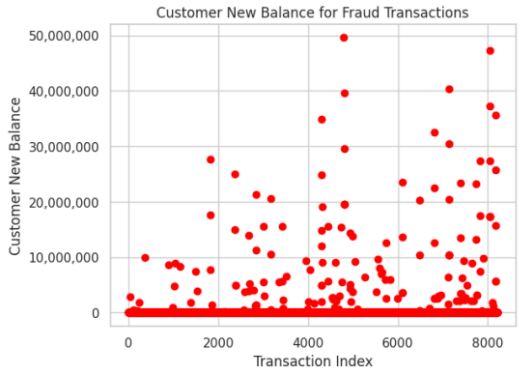
**Figure 6**: Proportion of All Transactions by Type




### 5.2.7 Customer New Balance for Fraud Transactions

A scatter plot of customer balances post-transaction reveals that fraudulent transactions often lead to a complete depletion of customer balances. **Figure 7** demonstrates that fraudsters tend to drain the account to zero, a consistent pattern in the data.

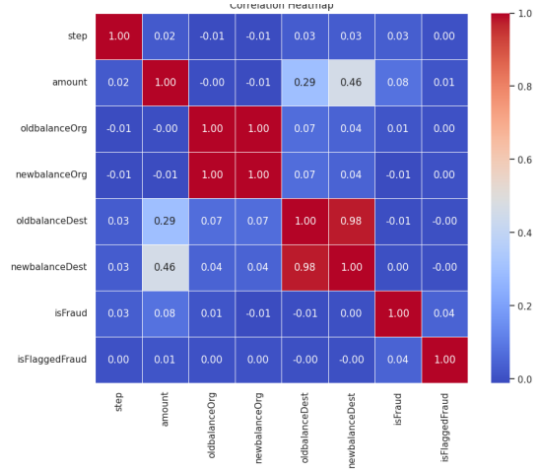
**Figure 7**: Customer New Balance for Fraud Transactions




### 5.2.8 Correlation Heatmap

A heatmap of feature correlations highlights strong relationships between numerical features. **Figure 9** shows that oldbalanceOrg and newbalanceOrg, as well as oldbalanceDest and newbalanceDest, exhibit strong correlations, which guide model development and feature engineering.

**Figure 8**: Correlation Heatmap




### Concluding Remarks on Results

The results confirm the effectiveness of ensemble methods, particularly **XGBoost**, in handling the complexities of fraud detection within imbalanced datasets. The insights gained from visualizations further validate the importance of specific features, transaction types, and customer behaviors in predicting fraudulent transactions. These findings reinforce the necessity of using sophisticated algorithms and robust feature engineering to build effective fraud detection systems.

## VII. Discussion

The performance analysis reveals that XGBoost outperformed all other models in detecting fraudulent transactions, achieving the highest ROC-AUC score and a balanced F1-Score. This highlights its superior ability to differentiate between fraud and non-fraud cases, making it an ideal choice for fraud detection tasks. The model provided a good balance between precision and recall, crucial for minimizing both false positives and false negatives. However, it also required substantial computational resources, making it less efficient for real-time applications.

In contrast, Random Forest displayed a high recall, identifying a large proportion of fraudulent transactions, but it struggled with precision, resulting in a relatively higher number of false positives. This trade-off between recall and precision is typical of Random Forest, which prioritizes sensitivity to fraud detection. While not as precise as XGBoost, it remained computationally efficient, making it a viable option in resource-constrained environments.

Logistic Regression, despite being highly interpretable, performed poorly in this context. Its linear nature failed to capture the complex, non-linear relationships within the data, leading to suboptimal results. This limitation underscores the importance of using more sophisticated models like XGBoost for imbalanced datasets where fraud detection relies on complex patterns.

Preprocessing steps, such as feature engineering and handling class imbalance, played a significant role in improving model performance. Additionally, hyperparameter tuning was crucial to optimizing the models' behavior, as even slight adjustments to parameters affected results, emphasizing the need for systematic fine-tuning and validation.

## VIII. Conclusion and Future Work

This study highlighted the effectiveness of ensemble methods, particularly XGBoost, in handling fraud detection tasks. Traditional models like Logistic Regression struggled due to their inability to capture the intricate relationships in imbalanced datasets. Ensemble techniques, on the other hand, leveraged the dataset's complexity to deliver superior performance, making them the preferred choice for fraud detection.

For future work, hybrid models that combine deep learning with ensemble methods could further enhance predictive accuracy. Additionally, challenges related to real-time deployment, such as ensuring low latency and scalability, need to be addressed. Exploring the integration of external datasets or additional features could also help improve the robustness and generalizability of the models, allowing for more comprehensive fraud detection systems.

## References


[1] A. A. Abaker and F. A. Saeed, "A Comparative Analysis of Machine Learning Algorithms to Build a Predictive Model for Detecting Diabetes Complications," Informatica, vol. 45, no. 1, Mar. 2021, doi: https://doi.org/10.31449/inf.v45i1.3111.

[2] Achraf Chikhi, S. Mohammadi, and Jan-Willem van Essen, "A Comparative Study of Traditional, Ensemble and Neural Network-Based Natural Language Processing Algorithms," Journal of risk and financial management, vol. 16, no. 7, pp. 327–327, Jul. 2023, doi: https://doi.org/10.3390/jrfm16070327.

[3] H. N. Noura, T. Chu, Z. Allal, O. Salman, and K. Chahine, "A comparative study of ensemble methods and multi-output classifiers for predictive maintenance of hydraulic systems," Results in Engineering, vol. 24, p. 102900, Sep. 2024, doi: https://doi.org/10.1016/j.rineng.2024.102900.

[4] G. M., "A Comparative Analysis of Ensemble Classifiers for Text Categorization," International Journal of Advanced Trends in Computer Science and Engineering, vol. 9, no. 1, pp. 344–348, Feb. 2020, doi: https://doi.org/10.30534/ijatcse/2020/51912020.

[5] N. Rahimi, F. Eassa, and L. Elrefaei, "An Ensemble Machine Learning Technique for Functional Requirement Classification," Symmetry, vol. 12, no. 10, p. 1601, Oct. 2020, doi: https://doi.org/10.3390/sym12101601.

[6] I. H. Sarker, "Machine Learning: Algorithms, Real-World Applications and Research Directions," SN Computer Science, vol. 2, no. 3, pp. 1–21, Mar. 2021, doi: https://doi.org/10.1007/s42979-021-00592-x.

[7] S. Mishra et al., "Improving the Accuracy of Ensemble Machine Learning Classification Models Using a Novel Bit-Fusion Algorithm for Healthcare AI Systems," Frontiers in Public Health, vol. 10, May 2022, doi: https://doi.org/10.3389/fpubh.2022.858282.

[8] K. Mali, "Linear Regression | Everything you need to Know about Linear Regression," Analytics Vidhya, Oct. 04, 2021. https://www.analyticsvidhya.com/blog/2021/10/everything-you-need-to-know-about-linear-regression/

[9] Yale University, "Linear Regression," Yale, 2019. http://www.stat.yale.edu/Courses/1997-98/101/linreg.htm

[10] KSV Muralidhar, "What is Stratified Cross-Validation in Machine Learning?," Medium, Feb. 14, 2021. https://towardsdatascience.com/what-is-stratified-cross-validation-in-machine-learning-8844f3e7ae8e?gi=ad6c579ff136 (accessed Nov. 18, 2024).

[11] J. Brownlee, "Random Oversampling and Undersampling for Imbalanced Classification," Machine Learning Mastery, Jan. 14, 2020. https://machinelearningmastery.com/random-oversampling-and-undersampling-for-imbalanced-classification/

[12] "Guide to K-Nearest Neighbors Algorithm in Machine Learning," Analytics Vidhya, Nov. 06, 2024. https://www.analyticsvidhya.com/articles/knn-algorithm/