Leveraging Machine Learning Techniques for Detecting Fraudulent Credit Card Transactions

Mumtahena Mim
MASTER OF SCIENCE IN BUSINESS ANALYTICS
University of Limerick
Limerick, Ireland
24214329@studentmail.ul.ie

Anika Tamanna Promi
MASTER OF SCIENCE IN BUSINESS ANALYTICS
University of Limerick
Limerick, Ireland
24214388@studentmail.ul.ie

Athul Polachan
MASTER OF SCIENCE IN BUSINESS ANALYTICS
University of Limerick
Limerick, Ireland
24154741@studentmail.ul.ie

Bhukya Daiva Vara Laxmi Prasad
MASTER OF SCIENCE IN BUSINESS ANALYTICS
University of Limerick
Limerick, Ireland
24188697@studentmail.ul.ie

Saurabh Velukkara Sanjay
MASTER OF SCIENCE IN BUSINESS ANALYTICS
University of Limerick
Limerick, Ireland
24046582@studentmail.ul.ie

***Abstract*— Fraud detection using credit cards is an essential issue in the financial sector, and fraud leads to financial losses and establishes a negative perception of the company among its clients. This project uses state of the art machine learning to design and test models for flagging credit card fraud. Using three publicly available datasets sourced from Kaggle, which include anonymized features, transaction amounts, and fraud indicators, the project aims to address the research question: "Which machine learning model demonstrates the highest effectiveness for detecting fraudulent credit card transactions?" Five machine learning algorithms- Random Forest, Logistic Regression, KNN, Decision Tree, Support Vector Machine with linear, and RBF kernels are incorporated and compared carefully by accuracy, precision, recall, F1-Score, Cohen's Kappa, and ROC-AUC. The project adopts the CRISP-DM process to be able to systematically analyze the data and build the model. Initial analysis provides an understanding of the abilities and inabilities of each model in working with uneven datasets and recognizing dishonest purchase orders. Thus, Random Forest and other ensemble algorithms combined with more complex models such as SVM show better testing precision and recall than simpler model types such as Logistic Regression but provide interpretability and tractability. That way, the findings of this study contribute relatively to the creation of strong fraud detection systems and present practical implications for financial institutions regarding fraud-relevant risk reduction.**

Keywords*— Credit card fraud, applications of machine learning, data science, Random Forest, Decision Tree, Logistic regression, Support Vector machine, K-nearest Neighbor, Accuracy

I. Introduction

Credit card fraud has become rampant in the financial world thus causing great losses and doubted integrity of financial institutions. Since so many more people are engaging in online purchases, a proper system of how to combat fraud has become more pressing. However, fraud detection is a more complex process, especially because of such features as rigorous data imbalance, constantly changing fraud schemes, and the demand for real-time response. Consequently, traditional approaches provide limited results in identifying rich patterns of fraudulent behaviors; therefore, it is crucial to implement advanced machine learning algorithms.

This work explores the ability of the ML techniques to identify credit card cheatings. It seeks to answer the research question: Here we go with the next question: Which machine learning model is most effective to identify Fake Credit Card Transactions? The work done is about comparing these five algorithms: Random Forest, Logistic Regression, K-NN, Decision Tree, and that of SVM in terms of accuracy, precision, recall and F1-score. CRISP-DM outlines a clear process of data preparation, modeling and evaluation of the model with the business.

The subsequent sections deliver a real analysis of the work in question. This paper designed a comparison in terms of the methods that employed machine learning algorithms for the classification of credit card with special emphasis on the imbalanced datasets and the performance of the system in these terms, such as accuracy, precision, and recall. In the current study, Random Forest, Logistic Regression, K-Nearest Neighbors, Decision Tree, Support Vector Machine are applied and the comparative strengths and weaknesses are discussed and it is indicated that Random Forest is the best model for this endeavor. Thus, consistent with the recommendations of the literature, the preprocessing procedure used in this study involved SMOTE and the usage of appropriate evaluation indicators. This research adds knowledge on how to implement and promote the reliability and effectiveness of a large scale fraud detection system, which has important signification for financial organizations seeking to minimize fraud related opportunities and enhance transaction safety.

II. RELATED WORK

Financial fraud has emerged as an increasingly pervasive threat, with far-reaching implications for the finance industry, corporate organizations, and governments [1]. In the field of payments using credit cards as a mode of payment preferred today security has been a subject of worry due to increased cases of fraud. Credit card fraud detection (CCFD) is the activity, which takes an attempt to distinguish between the fraudulent and genuine transactions by taking into account the patterns of the consumption and deviations from the patterns. But the handling of this problem has some difficulties: fraud patterns are not stationary, fraud datasets are inherently imbalanced, it is also challenging to select the proper features and the proper evaluation metrics for this type of data [2].

Machine learning is widely adopted in CCFD where different methods are used to detect fraud and differentiate normal and anomalous activity patterns. For example, Random Forests (RF) is a stable classification algorithm which training uses decision trees in order to increase the probability of correct classification. Compared to other models, RF performs extremely well for large, overly unbalanced datasets, as it combines contribution of multiple trees reducing a disadvantage of individual tree such as over-fitting. However, despite RF being efficient, it is precise in classification problems, though it performs poorly in some regression tasks and is sensitive to variance in large datasets that need more refined adjustments [3]. Nonetheless, because of the aforementioned characteristic of RF it is still the most useful approach to perform the CCFD for the data set in question because of the high robustness of the algorithm.

Another algorithm is the Support Vector Machines (SVM) which is also used in classification model for detecting fraud. SVM specifically searches for a legitimate and fraudulent transactions' feature space, and develops decision boundaries for them. The key factor is that this method demonstrates high efficiency on small and very structured data sets, achieving high accuracy with a small number of features added. Nonetheless, the computations for SVM become expensive for realtime applications specially on large databank containing more than 100,000 records. These restrictions make it less suitable for use in high velocity transactional situations [4].

KNN, under the family of supervised learning algorithms have been seen effective in detecting fraudulent activities during transactions. In other words, KNN analytically identifies and models outliers in transactional data by measuring correlation coefficients or distances between transaction points. Due to the nature of this research, this method proves most effective in low memory and low computation contexts while also outperforming other methods in reducing false alarms while preserving overall detection efficacy [5,6] However, the detection of anomalies very much depends on the quality of the data set and the parameters set, which somewhat reduces the applicability of KNN compared with other approaches based on anomalies.

At its simplest, Logistic Regression (LR) allows an easy and understandable way of detecting fraud which comprises of regression analysis of the connection between predictor variables and a response variable that is binary in nature. That is why this model is most valuable precisely in understanding how different predictors affect the probability of fraudulent transactions. Although the method is not as complex as in other techniques, the usage of linear models may not reveal many features of datasets [7].

Another often used technique in CCFD is Decision Trees (DT) owing to their fast calculations, easy implementation, and the fact that they can deal with noise. Since DT models divide datasets repeatedly, the resulting models are easy to explain to the audience and are also hierarchical in nature, which are always wanted by researchers as well as practitioners. However, DT models can over-fit the data they are trained on and, therefore, may not work well when applied to different data; this is averted by fine tuning such as shrinking the trees or using a bagging model [8].

Another important issue relates to the dependent variable, which is common for credit card fraud datasets – these data are typically private, and the problem is concentrated in the fact that fraudulent controls are relatively rare. Previous research has mitigated these issues by employing oversampling, undersampling strategies and synthesis of datasets. Some algorithms like RF, DT, and KNN are known to help in training skewed data distribution; SVM and LR have been applied on small subset data for fraudlist identification.

Fraud datasets employed for the research purposes are typically considered as private and possess a high separation between the objects: many normal transactions, few fraudulent ones. Previous work in this area has applied techniques like SMOTE, undersampling methods, and cost-sensitive approaches, for adequate training of the models. Both classification and regression models in this project also use similar pre-processing techniques in order to balance classes of a dataset for better prediction of machine learning models. RF and DT approaches have been used earlier with success to handle imbalanced data; whereas, SVM and KNN are used to identify local fraud patterns.

The reuse of the five techniques involves the analysis of the results of the decision-making procedure on new imbalanced data sets and the comparison of the resulting accuracy. The project proposes the use of RF and DT for their applicability to large datasets, SVM due to precision in the small feature subsets KNN for proficiency in anomaly detection, and LR as a base model. Therefore, through comparing of these techniques, this project aims to find out which algorithms or what combined algorithms are the most efficient in detecting frauds in credit card transactions. Furthermore, it will be possible to learn from the outcomes of reusing these methods to veil over the solutions to some of the shortcomings; for instance, Scala SVM and the tendency of DT to over-fit. Finally, this paper's comprehensive evaluation of several methods will help improve the general understanding of fraud detection concepts and their real-world implementation.

III. DATA MINING METHODOLOGY

In the machine learning community, there is considerable interest in making interpretable machine learning models, where individuals can understand that a given classification was made. This is important since, in many domains individuals will not trust an algorithmic result unless that person understands why the prediction was made. Several frameworks exist for auditing machine learning algorithms. Among those, we have chosen CRISP-DM framework in the evaluation stage to assist the auditor in understanding the model.

To address our research question: "*Which machine learning model is the most effective for detecting fraudulent credit card transactions?*" we have implied this method. The process is divided into six key phases, each contributing to a structured approach to data analysis and model evaluation.

A. Business Understanding

The primary objective is to identify the most effective machine learning techniques for detecting fraudulent transactions. The datasets contain highly imbalanced data, with fraudulent transactions representing a small minority. This necessitates models capable of handling class imbalance while minimizing false negatives to ensure accurate fraud detection.

B. Data Understanding

The datasets used in this study are sourced from Kaggle and contain transaction features such as anonymized variables, transaction amount, and class labels (fraud indicator). Along with this, we have found out if any null values are present there or not and if the values are numeric or not.F

Dataset 1: 284,807 rows and 31 columns.

Dataset 2: 568,630 rows and 31 columns.

Dataset 3: 150,000 rows and 32 columns.

Key observations from the data include the significant imbalance between legitimate and fraudulent transactions, necessitating preprocessing steps to balance the dataset.

C. Data Preparation

Preliminary steps were taken to ensure the datasets are suitable for machine learning models:

- Remove Duplicates: Duplicate entries in the datasets were identified and removed to ensure data quality. In Dataset 1, out of 284,807 rows, 1,081 duplicates were found and removed, leaving 283,726 unique rows. Dataset 2 and Dataset 3 did not contain any duplicate entries, so no changes were needed.
- Handling Missing Data: Missing or null values can lead to inaccuracies in model predictions. However, none of the datasets had missing or null values, so this step was straightforward, and no imputation or removal was necessary.
- Remove Outliers: Outliers can distort analysis, especially in numerical features. We used IQR (Interquartile Range) to remove the outliers.
    - Dataset 1: After cleaning, 188,657 rows were classified as legitimate (Class 0), and 473 as fraudulent (Class 1).
    - Dataset 2: After cleaning, 284,315 rows were legitimate (Class 0), and 247,529 were fraudulent (Class 1).
    - Dataset 3: After cleaning, 99,268 rows were legitimate (Class 0), and 269 were fraudulent (Class 1).
- Display Correlation Matrix and Remove Highly Correlated Data: A correlation matrix was used to identify relationships between features. High correlations can introduce redundancy and multicollinearity, affecting model performance. In Dataset 3, the "Time" feature was highly correlated with other variables, so it was removed. No highly correlated features were found in Dataset 1 or Dataset 2, so no further changes were necessary.

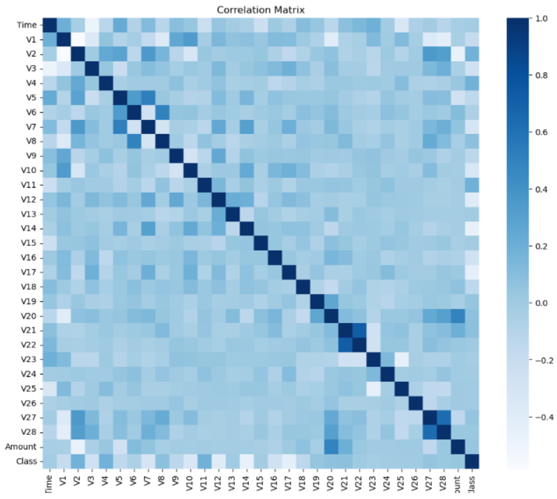

Fig.1 Correlation Matrix in Dataset-1

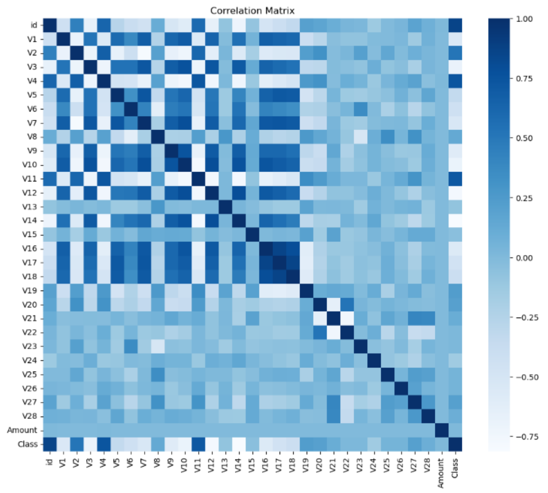

Fig.2 Correlation Matrix in Dataset-2

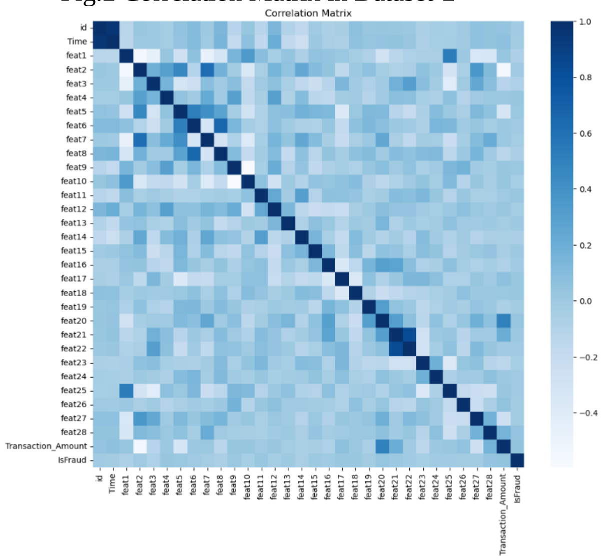

Fig.3 Correlation Matrix in Dataset-3

- Handle Class Imbalance Using SMOTE: Fraudulent transactions (Class 1) were significantly fewer than legitimate transactions (Class 0) in the datasets, leading to imbalanced classes that can bias the model. To address this, SMOTE (Synthetic Minority Oversampling Technique) was applied to Dataset 1 and Dataset 3 to balance the fraud and non-fraud classes by generating synthetic samples for the minority class. Dataset 2 was already balanced, so SMOTE was not applied to it. This step ensures that the model can learn effectively from both classes.

TABLE I.

	Original Class distribution		**balanced class distribution**	
fraud classification	***0 (Not fraud)***	***1 (is fraud)***	***0 (Not fraud)***	***1 (is fraud)***
Dataset 1	188,657	473	188,657	188,657
dataset 2	284,315	247,529	284,315	247,529
dataset 3	99,268	269	99,268	99,268

D. Modeling

Five machine learning techniques were implemented to classify transactions as fraudulent or legitimate:

- Random Forest - After, the data was preprocessed, the data was separated into training and testing data to ensure against overfitting of the model. Subsequently, we used class weights to give more samples from the minority class a higher priority during the model training process. Combining both the SMOTE and class weights, we strengthened the learning process still further. This coupling eliminates possible overfitting to the synthetically created samples that SMOTE entails while enhancing the performance of fraud case identification. The Random Forest approach was subsequently applied on the dataset so that transactions may be easily distinguished from the rest as either fraudulent or genuine.

- Logistic Regression - To perform Logistic Regression, data was also divided into training set and testing set. To do this, the data was scaled to eliminate dominance by large features before training the model which enhanced model performance. The scaled data was further used to train the exact Logistic Regression model to classify the transaction.

- K-Nearest Neighbors (KNN) - For K-NN, we applied two approaches with the hyperparameter n_neighbors = 5 and n_neighbors = 10 for further model assessment. Comparing them, we were able to define the number of neighbor points which provides the best balance between the detection accuracy and the time needed for the calculation.

- Decision Tree - Decision Tree model was applied and developed on the preprocessing data set. Following training, we did a visualization of the decision tree, in order to understand how the model judges the feature data for splitting into legitimate and fraudulent classes.

- Support Vector Machine (SVM) - Since it was computationally intensive to assess the effects of all the features in the full dataset, we first used stepwise forward selection to determine the most influential features in the classification task. Next, we preprocessed the data and decided to take a sample of 20,000 for training and testing. In order to optimize the set of features some dimensionality reduction techniques were used for better performance and to reduce the computational load. The reduced dataset was then split into training and testing sets, and two types of SVM models were applied: The linear SVM for the simple model where the decision hyperplane is a straight line or a plane, and the RBF SVM for the complicated model having curved decision boundaries. It made it possible for SVM to be implemented when there were limited datasets available or when datasets were large.

E. Evaluation

The models were evaluated using multiple metrics, including Accuracy, Precision, Recall, F1-Score, Cohen’s Kappa, MCC, and ROC-AUC. These metrics provide a comprehensive understanding of each model’s ability to handle imbalanced datasets and detect fraud effectively.

F. Deployment

Although deployment is not a core component of this project, the results offer actionable insights for integrating these models into real-time fraud detection systems. Model reproducibility and scalability were considered for potential industrial application.

By following the CRISP-DM framework, the project ensures a systematic and reproducible approach to addressing the research question. This methodology also allows for iterative improvement and thorough evaluation of each model's performance.

IV. EVALUATION

To determine the effectiveness of each machine learning method in detecting fraudulent credit card transactions, a robust evaluation framework was developed. This framework incorporates a combination of performance metrics, parameter tuning, and sampling strategies to ensure comprehensive and reliable results.

A. Evaluation Metrics

The following metrics were chosen to evaluate model performance, each addressing specific challenges posed by imbalanced datasets:

1) Accuracy: Accuracy is probably the simplest measure of accuracy that is widely used for many classification problems. It quantifies the number of instances in the training set that were recognized accurately by the machine out of the entire volume of the dataset. Accuracy works well when used as performance measure wherein the data set has half as many instances of each class. In such cases, it gives an easy to understand take on how the model performs in a general sense. It is selected as the initial measurement since it will establish a foundation for assessing the classification potential of the model. Hence, a high accuracy shows that the model is categorizing major instance types in a good way and consequently shows the capability of the model in learning patterns that it is trained with. But that could be misleading, particularly when dealing with imbalanced datasets, where accuracy could give a somewhat inflated picture by simply focusing more on the dominant class.

TABLE II.

	Dataset-1	Dataset-2	Dataset-3
Random Forest	99.99%	99.99%	99.96%
Decision Tree	99.83%	99.98%	86.88%
KNN (n=5)	95.32%	99.90%	83.64%
KNN (n=10)	93.56%	99.90%	81.55%
Logistic Regression	99.02%	99.86%	84.17%
SVM-Linear	97.20%	99.60%	79.65%
SVM-RBF	98.28%	98.82%	84.78%

Accuracy overall corrects the predictions, making it an effective first indicator of model performance. But alone, does not differentiate between types of errors (false positives vs. false negatives) or evaluate performance in imbalanced datasets. It also does not provide insights into how the model performs across individual classes.

2) Precision: Accuracy is another measure, which is typical for classification problems means the ratio of correctly positive predicted instances to the total number of 'positive' instances. They worked particularly well when the implications of false positives are high. For a given dataset where the positive class is scarce, precision is even more beneficial. Even in such a situation, their accuracy can become overly optimistic about the model's performance while precision speaks directly about the model's ability to classify the positive instances without worrying about the overall negative cases. This makes it a more reliable metric for measuring performance of models in such scenarios. Conversely, a high value of precision means that the model parades a low probability of giving incorrect positives hence advisable for projects where wrong positive prediction can be expensive.

	Dataset-1	Dataset-2	Dataset-3
Random Forest	1%	99.99%	99.94%
Decision Tree	99.97%	99.99%	87.45%
KNN (n=5)	94.21%	99.99%	80.29%
KNN (n=10)	93.41%	99.82%	80.43%
Logistic Regression	99.64%	99.92%	89.03%
SVM-Linear	99.90%	99.81%	90.48%
SVM-RBF	99.80%	99.80%	92.94%

Precision measures the accuracy of positive predictions, making it a critical metric for scenarios where false positives carry significant costs. It highlights the model's reliability in predicting the positive class. Precision does not account for false negatives and may present an incomplete picture when the goal is to ensure high sensitivity to the positive class. This necessitates evaluating recall alongside precision to balance both types of errors.

3) Recall: Sensitivity or recall, in a similar manner to the true positive rate, is the ability of the model to define the actual positive percentage that has been accurately recognized. It is especially useful when false negatives should be avoided to the extent possible. Recall is especially useful in situations of unequal or skewed data or in any application where a false negative is disastrous. For instance, in fraud detection, high recall means that most fraudulent are noticed, although with many alarms (false positives). Review complements accuracy, presenting a view on the manufactured compromise between false positive and false negative.

	Dataset-1	Dataset-2	Dataset-3
Random Forest	99.98%	99.98%	99.97%
Decision Tree	99.86%	99.97%	86.12%
KNN (n=5)	96.58%	99.83%	89.17%
KNN (n=10)	93.73%	98.57%	83.40%
Logistic Regression	98.38%	99.97%	77.95%
SVM-Linear	94.54%	99.83%	78.83%
SVM-RBF	96.77%	96.77%	75.20%

4) F1-Score: F1-score is defined as the average of the measures of precision and recall so it is an instance of a single mutual measurement that reflects the trade-off between the two. It is best applied especially when false positive and / or false negative are acceptable and must be balanced in the results. As for cases where both precision and recall are critical in this study, a measurement model called F1-score was applied to assess the accuracy of models such as Random Forest, KNN, Decision Tree, Logistic Regression, and SVM. While accuracy scores fall into a similar range for two distributions, F1-score considers the compromise between true positive recognition and minimal mining of false positives for sets with different class proportions.

	Dataset-1	Dataset-2	Dataset-3
Random Forest	99.99%	99.99%	99.96%
Decision Tree	99.83%	99.98%	86.78%
KNN (n=5)	95.38%	99.91%	84.49%
KNN (n=10)	93.57%	99.91%	81.89%
Logistic Regression	99.01%	99.81%	83.12%
SVM-Linear	97.14%	99.62%	66.18%
SVM-RBF	98.26%	98.26%	83.13%

5) Cohen's Kappa Score: Cohen's Kappa Score is an assessment of the degree of agreement between two forms of classification, when the agreement that is achieved by chance has been factor in to the calculation. This metric was utilized to measure the degree of reliance that could be placed in the models particularly in datasets which are imbalanced, hence; an accuracy of 95% might be deceptive. It also makes a point to make sure that even performance is rated higher than an optimistic estimate, which is extraordinarily beneficial as soon as utilized in models like Decision Tree and Random Forest.

	Dataset-1	Dataset-2	Dataset-3
Random Forest	99.98%	99.97%	99.92%
Decision Tree	99.65%	99.98%	73.77%
KNN (n=5)	90.64%	99.81%	67.27%
KNN (n=10)	87.11%	99.80%	63.10%
Logistic Regression	98.03%	99.96%	68.34%
SVM-Linear	94.40%	99.20%	59.28%
SVM-RBF	96.55%	99.54%	69.54%

6) Matthews Correlation Coefficient (MCC): MCC does take account of all the measures that makes up the confusion matrix, these include true positives, true negatives, false positives and false negatives of binary classification. Of late, the evaluation of classification algorithms on imbalanced data has been a major concern due to the subpar performance of standard accuracy. They were used on all five techniques in order to get an overall view of the quality of classifications achieved. Averaged evaluation that accomplishes a balanced assessment of all types of errors, especially for the case when errors are uneven in a dataset. While it is clear and easy to interpret compared to other sophisticated indicators such as accuracy.

	Dataset-1	Dataset-2	Dataset-3
Random Forest	99.98%	99.97%	99.92%
Decision Tree	99.65%	99.96%	73.78%
KNN (n=5)	90.67%	99.81%	67.69%
KNN (n=10)	87.11%	99.80%	63.14%

Logistic Regression	98.04%	99.71%	68.88%
SVM-Linear	94.54%	99.20%	61.53%
SVM-RBF	96.60%	96.55%	70.84%

7) ROC-AUC Score: OC-AUC quantifies the capacity of the model at separating the two classes. This curve shows the tradeoff between the true positive rate defined as recall and False Positive Rate. Confusion matrix and ROC-AUC were employed as there was interest in basic measures of discrimination since Random Forest, SVM and Logistic Regression are relied on to segregate the classes. That is why it offers a threshold-independent performance measure, which makes it very useful when dealing with binary classification. The behavior of the model on different thresholds between two classes. How to change threshold and what level of precision and recall we will receive on its basis.

	Dataset-1	**Dataset-2**	**Dataset-3**
Random Forest	1%	1%	1%
Decision Tree	99.95%	99.98%	94.05%
KNN (n=5)	98.66%	99.93%	91.59%
KNN (n=10)	98.42%	99.93%	90.42%
Logistic Regression	99.91%	99.99%	91.85%
SVM-Linear	97.22%	99.61%	79.62%
SVM-RBF	98.29%	98.29%	84.76%

8) Confusion Matrix Visualization: By using the confusion matrix, one can see the amount of true positive and true negative as well as false positive and false negative. It was envisaged that all the models should understand performance specifics in classification arrangement. It makes it possible to understand frequent misclassification patterns, and this is especially important for Decision Tree and KNN, where decision boundaries have an impact on the occurring errors.

9) Root Mean Squared Error: RMSE calculates average amount as well as frequency of the errors and it imposes heavier penalty for larger errors. As for measuring the error magnitude in numerical predictions, especially for Logistic Regression and Decision Tree models, RMSE was used. The prediction errors, with the focus on the larger size of errors. Direction of errors that is positive and negative deviations is not captured here.

	Dataset-1	**Dataset-2**	**Dataset-3**
Random Forest	0.0109	0.0119	0.0201
KNN (n=5)	0.2163	0.0310	0.4045
KNN (n=10)	0.2538	0.0317	0.4295
Logistic Regression	0.0992	0.0379	0.3987
SVM-Linear	0.1673	0.0632	0.4511
SVM-RBF	0.1313	0.1313	0.3902

10) Residual Sum of Squares: RSS measures the total squared differences between observed and predicted values. RSS was used as a complementary metric to RMSE to evaluate the overall error magnitude. The cumulative magnitude of prediction errors can be found from this.

	Dataset-1	**Dataset-2**	**Dataset-3**
Random Forest	9	15	16
Decision Tree	131	19	5208.0000
KNN (n=5)	3532	102	6498
KNN (n=10)	4862	107	7326
Logistic Regression	743	153	6285
SVM-Linear	112	16	814
SVM-RBF	69	69	609

11) Mean Absolute Percentage Error: MAPE measures the percentage error in predictions, making it useful for understanding relative prediction accuracy

	Dataset-1	**Dataset-2**	**Dataset-3**
Random Forest	59679573133.4628	127018199683.2865	1247597358242.0535
KNN (n=5)	133682243818956.4531	254036399366.5736	493027792388927.8125
KNN (n=10)	148960214541122.9062	254036399366.5737	457074304883225.0000
Logistic Regression	7937383226750.5459	1947612395143.7253	216401432684167.2188
SVM-Linear	2251799813685.2754	4503599627370.4990	156500087051124.9375
SVM-RBF	4503599627370.5117	4503599627370.5117	128352589380059.2500

These metrics were selected to provide a holistic evaluation, considering the imbalanced nature of the datasets and the real-world implications of false positives and false negatives.

B. Parameter Tuning

To decrease the chance of suboptimal performance, we checked and compared training and testing accuracy across all datasets by all models. While doing this, we noticed that KNN model had the problem of underfitting in one of the datasets. To address this, we tuned the hyperparameters of the model, more specifically adding more neighbors (n_neighbors = 20). In particular, this kind of adjustment let us determine which of the model's performance enhanced and whether the model was already well fitted without the risk of underfitting. Thus, through conducting systematic tests on this parameter, we were confident that the model could identify the underlying patterns in the data.

C. Key Observations:

a) Dataset-Specific Observations:

Dataset-1: Models provided good accuracy, out of all RF yielded the high test precision of 99.94%, as well as the high test recall of 99.98%. This brought out the main idea that this kind of dataset requires oversampling methodologies such as SMOTE.

Dataset-2: Due to the balanced nature of this dataset there isn't any drastic difference between the performance of various models, further emphasizing how greatly class balance can influence the results of any machine learning algorithm.

Dataset-3: There was lower accuracy and recall for all models (varied between 80.29% for KNN, 92.94% for SVM-RBF) due to a small number of fraud transactions and a higher level of noise, importance of which was revealed in this work.

b) Error Metrics Highlight Challenges in Dataset-3:

Large RSS values for Decision Tree (5208) and KNN (6498) in Dataset-3 suggested that there were large amount of errors made in predicting fraudulent credit card transactions due to problems addressing small size data sets, imbalanced datasets as well as noisy datasets. Mean Absolute Percentage Error (MAPE) values were drastically higher in Dataset-3.

c) Handling Imbalanced Datasets:

Fraudulent transactions are rare in most datasets, making it challenging to achieve high performance without biasing toward the majority class.

d) Consistently High Performance Across Metrics on RF:

Accuracy: RF provided the best mean accuracy rates of 99.99% on Dataset-1 and Dataset-2, and 99.96% for Dataset-3. This suggests that it is powerful in general pattern recognition because its accuracy rate is especially high.

Precision: Conversely, RF maintained very high precision (99.94% in Dataset-3) which implies high ability in minimizing on false positive since the datasets were imbalanced.

Recall: With recall rates ranging between 99.98 percent and 99.97 percent, RF indicated its capacity to avoid missing a significant number of credit card fraudulent transactions hence low false negatives.

F1-Score: From Table 5, it is evident RF gives the best or nearly the best F1-scores of 99.99% (Dataset-1 and Dataset-2), and 99.96% (Dataset-3) which means a good balance between precision and recall.

V. CONCLUSION

In this research, several features of machine learning methods in monitoring and checking fraudulent credit card transaction were investigated using three datasets with different properties. On the basis of the performance measures considered, Random Forest proved to be the most reliable of all the models tested for accuracy, precision, recall, and F1-score. Due to its capacity of limiting class imbalance and being free from false negatives while remaining easily explainable, it is very suitable for application in realworld means of fraud checking. The study also precipitated the need to apply method like SMOTE for handling imbalance in the dataset and also the need to employed metrics like Coherency's kappa and ROC-AUC for determining the level of reliability in the model. Conclusively, the research indicates that the technique with the best outcome is the Random Forest, then SVM and lastly the Logistic Regression for specific activities in that risky business dealing in imbalanced datasets..

VI. LIMITATIONS

- Dataset-3 posed challenges due to its small size, higher noise, and significant imbalance, limiting the generalizability of findings for noisy or sparse datasets.
- Computationally intensive models like SVM with RBF kernel struggled with scalability on large datasets, which limits their practical application in high-velocity transactional systems.
- Minimal exploration of feature importance and interaction effects, which could reveal additional insights or improve model performance.
- Deployment and integration of the models into real-time systems were beyond the scope of this study, limiting practical validation.

VII. FUTURE WORK

- Advanced Feature Engineering: Incorporate domain-specific features such as geographical location, transaction history trends, and customer segmentation to enhance model performance.

- Additional Datasets: Evaluate models on larger and more diverse datasets, including real-world financial institution data, to validate findings and improve generalizability.

- Algorithm Optimization: Explore advanced ensemble techniques, such as XGBoost or LightGBM, and deep learning models like LSTMs for sequential transaction analysis.

VIII. REFERENCES

[1] J. O. Awoyemi, A. O. Adetunmbi and S. A. Oluwadare, "Credit card fraud detection using machine learning techniques: A comparative analysis," 2017 International Conference on Computing Networking and Informatics (ICCNI), Lagos, Nigeria, 2017, pp. 1-9, doi: 10.1109/ICCNI.2017.8123782.

[2] F. N. Ogwueleka, "Data Mining Application in Credit Card Fraud Detection System", Journal of Engineering Science and Technology, vol. 6, no. 3, pp. 311-322, 2011.

[3] Bin Sulaiman, R., Schetinin, V. & Sant, P. Review of Machine Learning Approach on Credit Card Fraud Detection. Hum-Cent Intell Syst 2, 55–68 (2022). https://doi.org/10.1007/s44230-022-00004-0

[4] Sriram Sasank JVV, Sahith GR, Abhinav K, Belwal M. Credit card fraud detection using various classification and sampling techniques: a comparative study. In: IEEE, 2019. p. 1713–1718.

[5] Alam MN, Podder P, Bharati S, Mondal MRH. Effective machine learning approaches for credit card fraud detection. Cham: Springer; 2021.

[6] Itoo F, Meenakshi SS. Comparison and analysis of logistic regression, Naïve Bayes and KNN machine learning algorithms for credit card fraud detection. Int J Inf Technol. 2020;13:1503–11. https://doi.org/10.1007/s41870-020-00430-y.

[7] H. Z. Alenzi and N. O. Aljehane, "Fraud Detection in Credit Cards using Logistic Regression," International Journal of Advanced Computer Science and Applications, vol. 11, (12), 2020.

[8] Gaikwad, J.R., Deshmane, A.B., Somavanshi, H.V., Patil, S.V. and Badgujar, R.A., 2014. Credit card fraud detection using decision tree induction algorithm. International Journal of Innovative Technology and Exploring Engineering (IJITEE), 4(6), pp.2278-3075.

Task Distribution	**Name**	**ID**
Proposal Report	Mumtahena Mim	24214329
Linear Regression on Dataset 1	Athul Polacha	24154741
Linear Regression on Dataset 2	Saurabh Velukkara Sanjay	24046582
Linear Regression on Dataset 3	Bhukya Daiva Vara Laxmi Prasad	24188697
Decision Tree on Dataset 1	Anika Tamanna Promi	24214388
Decision Tree on Dataset 2	Athul Polacha	24154741
Decision Tree on Dataset 3	Saurabh Velukkara Sanjay	24046582
Random Forest on Dataset 1	Mumtahena Mim	24214329
Random Forest on Dataset 2	Anika Tamanna Promi	24214388
Random Forest on Dataset 3	Athul Polacha	24154741
KNN on Dataset 1	Bhukya Daiva Vara Laxmi Prasad	24188697
KNN on Dataset 2	Mumtahena Mim	24214329
KNN on Dataset 3	Anika Tamanna Promi	24214388
SVM on Dataset 1	Saurabh Velukkara Sanjay	24046582
SVM on Dataset 2	Bhukya Daiva Vara Laxmi Prasad	24188697
SVM on Dataset 3	Mumtahena Mim	24214329
Final Report	Mumtahena Mim	24214329