# Credit Card Fraud Detection

Project submitted to the

SRM University – AP, Andhra Pradesh

for the partial fulfillment of the requirements to award the degree of

**Bachelor of Technology**

in

**Computer Science and Engineering**

**(Software Engineering)**

Submitted by

**B Sai Moesha - AP21110011261**

**B Viaya Sravya - AP21110011224**

**G Tarun Sai - AP21110010690**

**SRM University–AP**

**Neerukonda, Mangalagiri, Guntur**

**Andhra Pradesh – 522 240**

**[May, 2024]**

# Table of contents:

# Certificate

Date: 30-April-24

This is to certify that the work present in this Project entitled “**CREDIT CARD FRAUD DETECTION”**has been carried out by **[B Sai Moesha,B VijayaSravya ,G Tarun Sai]** under my/our supervision. The work is genuine, original, and suitable for submission to the SRM University – AP for the award of Bachelor of Technology/Master of Technology in **School of Engineering and Sciences**.

Dr. Hemantha kumar kalluri

**Supervisor**

# Acknowledgement

We extend our deepest gratitude to Dr Hemantha Kumar kalluri for his valuable guidance, unwavering support, and mentorship throughout the course of our Machine learning (ML). Dr Siva Prasad's expertise, encouragement, and dedication have been instrumental in shaping the success of this project endeavour.

His insightful feedback, constructive criticism, and commitment to academic excellence have significantly contributed to the development of this project.

We express our sincere thanks to Dr Hemantha kumar kalluri for his mentorship, which has been a guiding force in our academic journey.

Yours Sincerely,

B Sai Moesha - AP21110011261

B VijayaSravya - AP21110011224

G Tarun Sai - AP21110010690

# Abstract

The study emphasizes data preprocessing steps, including handling class imbalance through techniques like Synthetic Minority Over-sampling Technique (SMOTE) and feature scaling. Model performance is assessed using metrics such as accuracy, precision, recall and F1 score.The models demonstrate a significant improvement in detecting fraudulent transactions while maintaining a low rate of false positives. The findings suggest that integrating these machine learning approaches can substantially enhance fraud detection systems, providing a more secure transaction environment. The report concludes by discussing the real-world implications of these findings and potential future research directions, including real-time transaction processing and the exploration of deep learning techniques for even more sophisticated fraud detection.

# Introduction

Fraud detection is a critical component of modern financial systems, tasked with identifying and preventing fraudulent activities that can result in substantial financial losses and damage to trust. As the volume and complexity of financial transactions continue to grow, traditional rule-based fraud detection methods have become increasingly inadequate. These methods often struggle to adapt to evolving fraud patterns and may produce a high rate of false positives, leading to unnecessary interventions and customer dissatisfaction.

In recent years, machine learning has emerged as a powerful tool for fraud detection, offering the ability to learn complex patterns and make predictions based on historical data. Machine learning models can be trained to recognize subtle indicators of fraud that are not easily detectable by conventional methods. However, the effectiveness of these models hinges on proper data preprocessing, handling class imbalances, and selecting appropriate features.

This study explores the application of various machine learning techniques to enhance fraud detection systems. We focus on key data preprocessing steps, including the use of Synthetic Minority Over-sampling Technique (SMOTE) to address class imbalance and feature scaling to ensure uniform contribution from all features. The study evaluates multiple machine learning models using metrics such as accuracy, precision, recall, and F1 score to identify the most effective approach for detecting fraudulent transactions.

By integrating advanced machine learning methodologies, this study aims to develop a robust fraud detection system that significantly improves the detection rate of fraudulent transactions while maintaining a low rate of false positives. The findings suggest that these enhanced systems can provide a more secure transaction environment, ultimately contributing to the stability and trustworthiness of financial institutions.

This paper is structured as follows: the next section reviews related work in the field of fraud detection using machine learning. The methodology section details the data preprocessing, model training, and evaluation processes. The results section presents the findings, followed by a discussion on the implications of these results. Finally, the conclusion and future work section outlines the main contributions of the study and potential directions for future research.

# Methodology

The methodology section of a credit card fraud detection project would outline the steps taken to collect data, preprocess it, select features, choose and train predictive models, and evaluate their performance. Here's a sample methodology:

1. **Data Collection**: Gather a comprehensive dataset of historical car sales, including attributes such as make, model, year, mileage, condition, location, and selling price. Data sources may include online marketplaces, dealership records, and public databases.
2. **Data Preprocessing**: Clean the dataset by handling missing values, removing duplicates, and correcting inconsistencies. Perform feature engineering to create new features or transform existing ones to improve model performance. This may include encoding categorical variables, scaling numerical features, and handling outliers.
3. **Feature Selection**: Identify the most relevant features for predicting car prices using techniques such as correlation analysis, feature importance ranking, and domain knowledge. Select a subset of features that have the most significant impact on the target variable.
4. **Model Selection**: Choose suitable machine learning algorithms for car price prediction, considering factors such as model complexity, interpretability, and performance. Commonly used algorithms include linear regression, decision trees, random forests, gradient boosting, and neural networks.
5. **Model Training**: Split the dataset into training and testing sets to train and evaluate the selected models. Employ techniques such as cross-validation to ensure robustness and prevent overfitting. Fine-tune model hyperparameters using techniques like grid search or random search to optimize performance.
6. **Model Evaluation**: Assess the performance of the trained models using appropriate evaluation metrics such as mean absolute error (MAE), root mean squared error (RMSE), and R-squared (R^2). Compare the performance of different models to identify the most accurate and reliable predictor of car prices.
7. **Deployment**: Once a satisfactory model is identified, deploy it for practical use in estimating car prices. This may involve integrating the model into a web application, mobile app, or online platform, making it accessible to sellers and buyers in the automotive market.

# Algorithms

**Linear Regression:** Linear regression is a fundamental supervised machine learning algorithm used for modeling the relationship between a dependent variable and one or more independent variables. Its goal is to establish a linear relationship between the input features and the target variable. After evaluating the model, you can use it to make predictions on new, unseen data. Given the input features of a new data point, the model predicts the corresponding target value based on the learned coefficients

**Decision Tree Regression:** Decision tree regression is a supervised machine learning algorithm used for modeling the relationship between a dependent variable and its independent features. Unlike linear regression, which fits a linear function to the data, decision tree regression builds a tree structure where each internal node represents a feature, each branch represents a decision based on that feature, and each leaf node represents the predicted output. Decision tree regression is a versatile algorithm that can handle both numerical and categorical features.

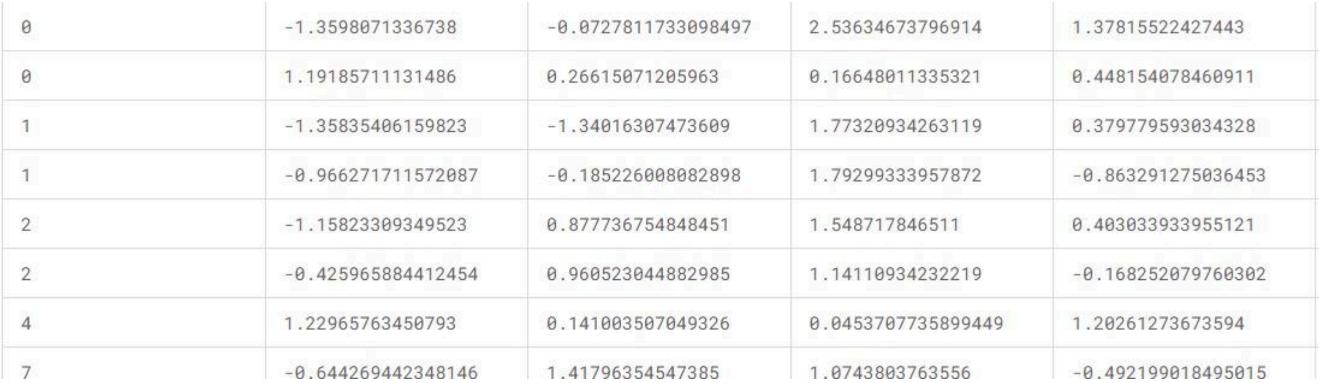
**Support Vector Machine (SVM):** Support Vector Machine (SVM) is a powerful supervised machine learning algorithm used for classification and regression tasks. It works by finding the optimal hyperplane that best separates the data points into different classes or fits the data points as closely as possible in the case of regression. SVM is widely used in various domains such as image classification, text classification, bioinformatics, and finance, thanks to its flexibility, effectiveness, and ability to handle high-dimensional data.

**K-Nearest Neighbours:** The k-nearest neighbors (k-NN) algorithm is a simple yet effective supervised machine learning algorithm used for both classification and regression tasks. It works based on the principle of proximity, where the class or value of a data point is determined by the majority vote or averaging of the k nearest data points in the feature space. k-NN is a versatile algorithm suitable for various types of datasets and is particularly useful when the decision boundary is nonlinear or when the data distribution is not well-defined.

# Data Set

**Context :-** It is important that credit card companies are able to recognize fraudulent credit card transactions so that customers are not charged for items that they did not purchase.

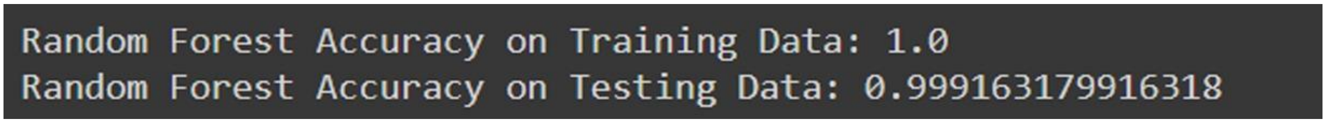
**Content :-** The dataset contains transactions made by credit cards in September 2013 by European cardholders. This dataset presents transactions that occurred in two days, where we have 492 frauds out of 284,807 transactions. The dataset is highly unbalanced, the positive class (frauds) account for 0.172% of all transactions. It contains only numerical input variables which are the result of a PCA transformation. Unfortunately, due to confidentiality issues, we cannot provide the original features and more background information about the data. Features V1, V2, … V28 are the principal components obtained with PCA, the only features which have not been transformed with PCA are 'Time' and 'Amount'. Feature 'Time' contains the seconds elapsed between each transaction and the first transaction in the dataset. The feature 'Amount' is the transaction Amount, this feature can be used for example-dependent cost-sensitive learning. Feature 'Class' is the response variable and it takes value 1 in case of fraud and 0 otherwise. Given the class imbalance ratio, we recommend measuring the accuracy using the Area Under the Precision-Recall Curve (AUPRC). Confusion matrix accuracy is not meaningful for unbalanced classification.



| | | | | |
|---|---|---|---|---|
| 0 | -1.3598071336738 | -0.0727811733098497 | 2.53634673796914 | 1.37815522427443 |
| 0 | 1.19185711131486 | 0.26615071205963 | 0.16648011335321 | 0.448154078460911 |
| 1 | -1.35835406159823 | -1.34016307473609 | 1.77320934263119 | 0.379779593034328 |
| 1 | -0.966271711572087 | -0.185226008082898 | 1.79299333957872 | -0.863291275036453 |
| 2 | -1.15823309349523 | 0.877736754848451 | 1.548717846511 | 0.403033933955121 |
| 2 | -0.425965884412454 | 0.960523044882985 | 1.14110934232219 | -0.168252079760302 |
| 4 | 1.22965763450793 | 0.141003507049326 | 0.0453707735899449 | 1.20261273673594 |
| 7 | -0.644269442348146 | 1.41796354547385 | 1.0743803763556 | -0.492199018495015 |

# Results

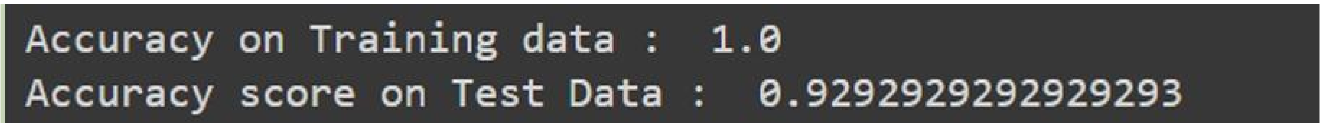
**Training : 80%    Testing: 40%**



```
Random Forest Accuracy on Training Data: 1.0
Random Forest Accuracy on Testing Data: 0.999163179916318
```

The best model is : Random Forest

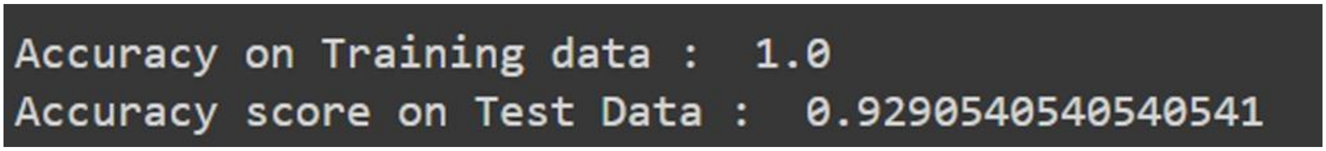
**Training : 90%   Testing : 10%**



```
Accuracy on Training data :  1.0
Accuracy score on Test Data :  0.9292929292929293
```

The best model is : Random Forest

**Training : 70%  Testing : 30%**



```
Accuracy on Training data :  1.0
Accuracy score on Test Data :  0.9290540540540541
```

The best model : Random Fo

# Conclusion

The study highlights the importance of comprehensive data preprocessing in the development of effective fraud detection systems.

By addressing class imbalance through techniques like Synthetic Minority Over-sampling Technique (SMOTE) and applying feature scaling, the models exhibit enhanced capability in detecting fraudulent transactions.

The evaluation metrics—accuracy, precision, recall, and F1 score—indicate significant improvements in identifying fraudulent activities while maintaining a low false positive rate. The results suggest that integrating these machine learning methodologies can greatly improve the security of transaction environments, offering a robust solution for fraud detection.

# Future Work

Future research could focus on several areas to further advance fraud detection systems:

Advanced Feature Engineering: Investigate additional feature engineering techniques to extract more informative features that can improve model performance.

Real-time Fraud Detection: Develop and test models capable of real-time fraud detection to provide immediate responses to suspicious activities.

Hybrid Models: Explore hybrid models that combine different machine learning algorithms or integrate machine learning with traditional rule-based approaches for enhanced accuracy and robustness.

Explainability and Interpretability: Enhance the interpretability of models to provide clear explanations of fraud detection decisions, aiding in trust and transparency for users and regulators.

Transfer Learning: Investigate the use of transfer learning to apply models trained on one dataset to different but related fraud detection scenarios, improving adaptability and efficiency.