Visvesvaraya Technological University

Jnana Sangama, Belagavi - 590018

A Project Work Phase-2 (18CSP83)

Report on

"CREDIT CARD FRAUD DETECTION USING MACINE LEARNING"

Project Report submitted in partial fulfilment of the requirement for the award of the degree of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

Submitted by

BHARGAV M	**1KG19CS009**
GAURAV G	**1KG19CS032**
HARSITH S	**1KG19CS037**
SUHAIL AHMED SAYYED	**1KG19CS100**

Under the guidance of
Mrs. BELJI T
Assistant Professor
Department of Computer Science & Engineering
KSSEM, Bengaluru-560109

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

K. S. School of Engineering and Management

#15, Mallasandra, off. Kanakapura Road, Bengaluru – 560109
2022 - 2023

K. S. School of Engineering and Management

#15, Mallasandra, off. Kanakapura Road, Bengaluru - 560109

Department of Computer Science & Engineering

KSSEM
K S SCHOOL OF ENGINEERING AND MANAGEMENT

CERTIFICATE

Certified that the Project Work Phase-II (18CSP83) entitled **"CREDIT CARD FRAUD DETECTION USING MACHINE LEARNING"** is a bonafide work carried out by:

BHARGAV M	**1KG19CS009**
GAURAV G	**1KG19CS032**
HARSITH S	**1KG19CS037**
SUHAIL AHMED SAYYED	**1KG19CS100**

in partial fulfilment for VIII semester B.E., Project Work in the branch of Computer Science and Engineering prescribed by **Visvesvaraya Technological University, Belagavi** during the period of February 2023 to May 2023. It is certified that all the corrections and suggestions indicated for internal assessment have been incorporated. The Project Work Phase-2 Report has been approved as it satisfies the academic requirements in report of project work prescribed for the Bachelor of Engineering degree.

…………………………… ..…………….……… …..…..……………………

Signature of the Guide **Signature of the HOD** **Signature of the Principal**

[Mrs. Belji T] [Dr. K Venkata Rao] [Dr. K Rama Narasimha]

DECLARATION

We, the undersigned students of 8th semester, Computer Science & Engineering, KSSEM, declare that our Project Work Phase-II entitled "**CREDIT CARD FRAUD DETECTION USING MACHINE LEARNING**", is a bonafide work of ours. Our project is neither a copy nor by means a modification of any other engineering project.

We also declare that this project was not entitled for submission to any other university in the past and shall remain the only submission made and will not be submitted by us to any other university in the future.

Place:
Date:

Name and USN	**Signature**
BHARGAV M (1KG19CS009)	**………………….**
GAURAV G (1KG19CS032)	**………………….**
HARSITH S (1KG19CS037)	**………………….**
SUHAIL AHMED SAYYED (1KG19CS100)	**………………….**

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task will be incomplete without the mention of the individuals, we are greatly indebted to, who through guidance and providing facilities have served as a beacon of light and crowned our efforts with success.

We take this opportunity to express our sincere gratitude to our college **K.S. School of Engineering and Management,** Bengaluru for providing the environment to work on our project.

We would like to express our gratitude to our **MANAGEMENT,** K.S. School of Engineering and Management, Bengaluru, for providing a very good infrastructure and all the kindness forwarded to us in carrying out this project work in college.

We would like to express our gratitude to **Dr. K.V.A Balaji, CEO**, K.S. School of Engineering and Management, Bengaluru, for his valuable guidance.

We would like to express our gratitude to **Dr. K. Rama Narasimha, Principal**, K.S. School of Engineering and Management, Bengaluru, for his valuable guidance.

We like to extend our gratitude to **Dr. K Venkata Rao**, **Professor and Head**, Department of Computer Science & Engineering, for providing a very good facilities and all the support forwarded to us in carrying out this Project Work Phase-II successfully.

We also like to thank our Project Coordinators, **Mrs. Jayashuba J, Asst. Professor, Mrs. Supriya Suresh, Asst. Professor, Department of Computer Science & Engineering** for their help and support provided to carry out the Project Work Phase-II successfully.

Also, we are thankful to **Mrs. Belji T, Assistant Professor**, for being our Project Guide, under whose able guidance this project work has been carried out Project Work Phase-II successfully.

We are also thankful to the teaching and non-teaching staff of Computer Science & Engineering, KSSEM for helping us in completing the Project Work Phase-II work.

BHARGAV M

GAURAV G

HARSITH S

SUHAIL AHMED SAYYYED

ABSTRACT

Every year fraud cost generated in the economy is more than $4 trillion internationally. Financial institutions such as commercial and investment banking operations are increasingly bring targeted. Users can use credit card as it provides an efficient and it is easy to use. Due to the increase of usage of credit cards ,the credit card misuse has been enhanced. Fraud detection means a collection of activities to avoid collecting money by misleading pretensions. The main aim to detect such frauds, including the accessibility of public data, the changes in the fraud nature and high rates of false alarm. A machine learning algorithm was first applied to dataset, which improve the accuracy of the detection of frauds to some extent. A number of sectors are today using fraud detection which includes ecommerce and banking agencies. The mode of payment has moved from cash to digital settlements such as debit/credit card, online wallet payment, online banking etc. As the result financial fraud is increasing at rapid rate for personal gain. The algorithms used are K Nearest Neighbors, Logistic Regression, Decision Tree and Random Forest.

Credit card fraud detection using machine learning involves developing models to identify fraudulent transactions and prevent fraudulent activities in real-time. This paper presents an abstract of such a system, which employs machine learning algorithms to analyze historical transaction data and detect patterns that indicate fraudulent behavior. The system uses a variety of techniques, including supervised and unsupervised learning, to identify fraudulent transactions based on various features such as transaction amount, location, time of day, and user behavior. Once a fraudulent transaction is detected, the system can trigger alerts, freeze accounts, or take other appropriate actions to prevent further losses. The proposed system can improve the accuracy and speed of fraud detection and reduce the impact of fraud on credit card companies and consumers.

TABLE OF CONTENTS

LIST OF FIGURES

Chapter 1

INTRODUCTION

1.1 OVERVIEW

Fraud is defined as a wrongful or criminal deception which is aimed to bring financial or personal gain. Two mechanisms are used to avoid fraud and losses due to fraud. They are Fraud Prevention and Fraud Detection. Fraud Prevention is a proactive method where it stops fraud from being happening. Fraud Detection is used when a fraudulent transaction is attempted by the fraudster.

Users can use credit card as it provides an efficient and it is easy to use. Due to the increase of usage of credit cards ,the credit card misuse has been enhanced. Fraud detection means a collection of activities to avoid collecting money by misleading pretensions. The main aim to detect such frauds, including the accessability of public data, the changes in the fraud nature and high rates of false alarm.

A machine learning algorithm was first applied to dataset, which improve the accuracy of the detection of frauds to some extent. A number of sectors are today using fraud detection which includes ecommerce and banking agencies. The mode of payment has moved from cash to digital settlements such as debit/credit card, online wallet payment, online banking etc. As the result financial fraud is increasing at rapid rate for personal gain. The algorithms used are KNN, Random Forest, Decision Tree and Logistic Regression. Credit card fraud detection is a process of identifying and preventing fraudulent transactions made using credit or debit cards. With the increasing number of credit card transactions, credit card fraud has become a significant problem for financial institutions, merchants, and consumers.

There are several methods and techniques used for credit card fraud detection, including rule-based systems, anomaly detection, and machine learning algorithms. Here's an overview of each method: Rule-based systems: These systems use predefined rules to identify fraudulent transactions. For example, if a transaction exceeds a certain amount or occurs in a foreign country, it may be flagged as suspicious. Anomaly detection: These systems use statistical models to identify transactions that deviate from normal behavior. For example, if a customer suddenly makes a large number of transactions or makes transactions at unusual times of day, it may be flagged as suspicious.Machine learning algorithms: These systems use machine learning algorithms to learn from historical data and identify patterns that indicate fraud. Popular algorithms include logistic regression, decision trees, random forests, and neural networks.

1.2 PURPOSE OF THE PROJECT

With the growing prevalence of electronic payment systems, credit card fraud has become a significant concern for both merchants and consumers. Machine learning techniques can be applied to analyze large volumes of transaction data and identify patterns that indicate fraudulent behavior, such as unusual spending patterns, geographic anomalies, and other factors that can beindicative of fraud. By detecting fraudulent transactions in real-time, credit card companies can prevent financial losses and minimize the impact of fraud on consumers. The use of machine learning algorithms can improve the accuracy and speed of fraud detection.

1.3 SCOPE OF THE PROJECT

The scope of credit card fraud detection using machine learning is quite broad and covers variousaspects of electronic payment systems. Here are some of the key areas where machine learning techniques can be applied to prevent credit card fraud Firstly transaction monitoring where machine learning algorithms can analyze large volumes of transaction data in real-time and detectfraudulent activities. Followed by Risk assessment where machine learning models can be trainedto assess the risk associated with specific transactions, users, or merchants. This can help credit card companies to prioritize their fraud prevention. It is also used in user behavior analysis wheremachine learning algorithms can analyze user behavior patterns and detect suspicious activities such as login attempts from unusual locations, changes in spending patterns, or multiple account registrations with the same device. Another key application is Fraud trend analysis where machinelearning models can be trained to identify emerging fraud patterns and trends. This can help creditcard companies to proactively prevent future fraud attempts and minimize losses.

1.4 DEFINITION

1.4.1 PYTHON

Python is a computer programming language often used to build websites and software, automate tasks, and conduct data analysis. Python is a general-purpose language, meaning it can be used to create a variety of different programs and isn't specialized for any specific problems. Python has a large standard library that provides many useful tools and modules for tasks such as web development, scientific computing, data analysis, and more. It also has a vast ecosystem of third-party libraries and frameworks, such as NumPy, Pandas, Django, and Flask, that extend its capabilities and make it a powerful tool for a wide range of applications. Python's syntax is clean and concise, and it emphasizes code readability and maintainability. It supports object-oriented, functional, and procedural programming paradigms, and its dynamic typing and garbage collection make it easy to use and learn.

1.4.2 JUPYTER NOTEBOOK

The Jupyter Notebook Application is a server-client application that allows editing and running notebook documents via a web browser. The Jupyter Notebook Application can be executed on a local desktop requiring no internet access accessed through the internet. Jupyter Notebook provides an interactive computing environment for coding in many languages.

Jupyter Notebook provides an interactive computing environment where users can write and execute code in different programming languages, including Python, R, and Julia. The code is organized into cells, which can be run independently, allowing users to test and experiment with different code snippets and algorithms. The output of each cell is displayed directly below it, making it easy to visualize and understand the results of the code. In addition to code cells, It alsosupports markdown cells, which allow users to create rich-text documents that include formatted text, images, and equations. This makes it easy to create reports and presentations that combine code, data, and explanatory text. It runs in a web browser and can be accessed locally on a user's computer or remotely on a cloud server. It also supports the creation of interactive dashboards and widgets, which allow users to create dynamic and responsive visualizations and user interfaces.

1.4.3 MACHINE LEARNING

Machine learning (ML) is a type of artificial intelligence (AI) that allows software applications to become more accurate at predicting outcomes without being explicitly programmed to do so. Machine learning algorithms use historical data as input to predict new output values. The primary goal of machine learning is to build models that can generalize from data and make accurate predictions or decisions on new, unseen data. This involves training models on labeled datasets, where the model learns to recognize patterns and relationships in the data and makes predictions based on those patterns.

1.4.4 KNN

K-Nearest Neighbor is one of the simplest Machine Learning algorithms based on Supervised Learning technique. K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories. K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K- NN algorithm. K-NN algorithm can be used for Regression as well as for Classification but mostly it is used for the Classification problems. KNN has several advantages, including its simplicity, ease of implementation, and ability to work well on small datasets. However, it can be computationally expensive and may not work well on high-dimensional datasets with many features.

1.4.5 LOGISTIC REGRESSION

Logistic regression is a statistical method that is used for building machine learning models where the dependent variable is dichotomous: i.e. binary. Logistic regression is used to describe data and the relationship between one dependent variable and one or more independent variables.The independent variables can be nominal, ordinal, or of interval type .The name is derived fromthe concept of the logistic function that it uses. The logistic function is also known as the sigmoid function. The value of this logistic function lies between zero and one . During training,the logistic regression model tries to optimize the weights or coefficients such that the predicted probabilities are as close as possible to the actual binary outcomes in the training set. This is typically done using maximum likelihood estimation or gradient descent optimization.

1.4.6 DECISION TREE

Decision Tree is a Supervised learning technique that can be used for both classification and Regression problems, but mostly it is preferred for solving Classification problems. It is a tree-structured classifier, where internal nodes represent the features of a dataset, branches represent the decision rules and each leaf node represents the outcome.In a Decision tree, there are two nodes, which are the Decision Node and Leaf Node. Decision nodes are used to make any decisionand have multiple branches, whereas Leaf nodes are the output of those decisions and do not contain any further branches.The decisions or the test are performed on the basis of features of thegiven dataset.It is a graphical representation for getting all the possible solutions to a problem/decision based on given conditions.It is called a decision tree because, similar to a tree, itstarts with the root node, which expands on further branches and constructs a tree-like structure

1.4.7 RANDOM FOREST

Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve acomplex problem and to improve the performance of the model. Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output.The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting. Random Forest has several advantages, including its ability to handle high-dimensional data .

Chapter 2

LITERATURE SURVEY

2.1 :Literature Review of Different Machine Learning Algorithms for Credit Card Fraud Detection

Every year fraud cost generated in the economy is more than $4 trillion internationally. This is unsurprising, as the return on investment for fraud can be massive. Cybercrime specialists estimatethat an investment of 1 million dollars into fraud or attack can net up to $100 million. Financial institutions such as commercial and investment banking operations are increasingly being targeted.And we know that the only way to fight fraud effectively is through the use of advanced technology. The answer lies in relying on advanced analytics and enterprise wide data storage capabilities that support the use of artificial intelligence (AI) and machine learning (ML) approaches to stay one step ahead of criminals. AI is best suited to defend against today’s fast-changing and complex bank fraud, where new threats are under development every day. Approaches relying on fragmented and siloed data, rules-based approaches or traditional point-solutions are no longer acceptable. These approaches are not only ineffective, but they are extremely costly to banks and financial services firms because they force legal and compliance teams to spend a lot of time trying to gain access to the data they need. By relying on advanced analytics and AI and ML capabilities, fraud and compliance units can spend their time working onmore-complex fraud issues. Manual investigation can be reduced through the use of complex algorithms powered by ML, often in conjunction with rules, a combination that offers significant advantages over purely based -rules fraud detection. In this paper, we have included different machine learning algorithms used to detect credit card frauds and also provide a comparative studybetween different algorithms.

2.2 :Survey on Credit Card Fraud Detection

Financial Services have serious problems due to credit card fraud. Lot of money is lost due to credit card fraud every year. As the technology is increasing day by day the financial fraud is alsoincreasing. As a result of all this financial loss due to financial fraud is increasing day by day. In order to reduce this problem, fraud detection has become an important tool and probably the best way to stop such frauds. In this study various fraud detection techniques has been employed. Frauddetection using credit card is an extremely difficult task and is very difficult to detect. Many approaches has been proposed to solve this problem. The most commonly used fraud detection methods are rule induction technique, decision tree, Logistic Regression are used.These algorithmscan be used as single model or can be used in combination. These machine learning algorithms aresuccessful in many cases but still cannot generate accurate result. A cost sensitive decision tree approach has been used for fraud detection. A cost called misclassification cost is used which is taken as varying as well as priorities of the fraud also differs according to individual records.

2.3 :Credit Card Fraud detection using Machine Learning Models

Due to increase of fraud which results in loss of money across the globe , several methodologies and techniques developed for detecting frauds Fraud detection involves analyze the activities of users in order to understand the malicious behaviour of users.Malicious behaviour is a broad termincluding delinquency, fraud, intrusion, and account defaulting. This paper presents a survey of current techniques used in credit card fraud detection and evaluates the machine learning approachto identify fraud detection. In the proposed work, we analyze credit card fraud detection using machine learning algorithm namely K-Nearest Neighbor and Ensemble Model of Random Forest. To make the learning process efficient, we used infinite latent feature selection algorithm for feature selection. The performance of the algorithm is evaluated on various measures like Accuracy,Precision and Recall. The Random Forest is studied through the perspective of the Adaptive Nearest Neighbor. They introduce the concept of monotone distance measures and potential nearest neighbors and show that the Random Forest can be viewed as an adaptive learningmechanism of k Potential Nearest.

2.4 :Review of Machine Learning Approach on Credit Card Fraud Detection using Machine Learning

Massive usage of credit cards has caused an escalation of fraud. Usage of credit cards hasresulted in the growth of online business advancement and ease of the e-payment system.The use of machine learning (methods) are adapted on a larger scale to detect and preventfraud. ML algorithms play an essential role in analyzing customer data. In this research article, we have conducted a comparative analysis of the literature review considering theML techniques for credit card fraud detection (CCFD) and data confidentiality. In the end, we have proposed a hybrid solution, using the KNN in a federated learningframework. It has been observed as an effective solution for achieving higher accuracy in CCFD while ensuring privacy. It is imperative for any banking or financial institutionthat issues credit and debit cards to put in place an effective measure to detect any cases of fraudulent transactions. Some of the notable methods identified to help detect fraud incredit card that includes Random Forest, k-nearest neighbors and other techniques. Although the random forest algorithms are quite effective in predicting the class of regression problems, they constitute various limitations when it comes to the CCFD in real-time.

2.5 :Credit Card Fraud Detection Techniques: A Review

The prediction analysis is the approach which can predict future possibilities on the current data. When the physical-card based purchasing technique is applied, the card is given by the cardholder tothe merchant so that a successful payment method can be performed. The fraudulent transactions areconducted by the attacker by stealing the credit card. When the loss of the card is not noticed by the cardholder, a huge loss can be faced by the credit card company. A very little amount of information is required by the attacker for conducting any fraudulent transaction in online transactions. In this research work, various credit card fraud detection techniques are reviewed in terms of certain parameters. The prediction analysis is most useful type of data which is performed today. To performthe prediction analysis the patterns needs to generate from the dataset with the machine learning. Theprediction analysis can be done by gathering historical information to generate future trends. So, theknowledge of what has happened previously is used to provide the best valuation of what will happenin future with predictive analysis.

Chapter 3

PROBLEM IDENTIFICATION

3.1 PROBLEM STATEMENT

The credit card fraud detection problem includes modelling past credit card transactions with the knowledge of the ones turned out to be fraud .This model is then used to identify whether a new transaction is fraudulent or not .Our aim is to detect 100% of the fraudulent transaction while minimizing the incorrect fraud classification. The model needs to be trained on a large dataset of historical credit card transactions that are labeled as either fraudulent or legitimate. The aim is to build a system that can detect fraudulent transactions quickly and accurately, while minimizing the number of false positives (legitimate transactions that are incorrectly flagged as fraudulent). The model needs to be scalable and efficient, able to handle large volumes of transactions in real-time, and adaptable to changing fraud patterns over time. Ultimately, the goal is to prevent financial losses to credit card companies and protect the interests of consumers.

3.2 PROJECT SCOPE

Our proposed strategy focuses on a advanced machine learning procedures for credit card faud detection(CCFD) classification and prediction,thus overcoming existing problem.by utilizing KNN, Random forest ,logistic regession and desicision tree alogorithm.we will make our model to in order to increase the performance and accuracy. Machine learning models can recognise unusual credit card transactions and fraud .The first and foremost step involves collecting and storing raw data,which is then used to train the model to predict the probability of fraud.Utilizing of machine learning algorithms was connected in various credit card transaction datasets.Machine Learning Strategies have diverse power in different credit card transaction datasets. Previously mentioned conventional machine learning techniques gave less exact outcome and results additionally shifts in light of the procedures has been utilized for the prediction. Model training where we train the selected model using the prepared data to develop an accurate and reliable fraud detection system.Model testing and evaluation where we test the model on a separate dataset and evaluate its performance using metrics such as accuracy, precision, recall, and F1-score.Then we finally implement the model in a real-time system that can monitor creditcard transactions and flag potential fraudulent actions.

Chapter 4

GOALS AND OBJECTIVES

4.1 PROJECT GOALS

The goal of this project is to develop an application to Detect The fraudulent Transactions, Minimization of credit card fraud, Better performance and accuracy and Analysis Multiple Machine Learning Algorithm The project's ultimate objective is to minimize financial losses to credit card companies and protect consumers from fraudulent activities. Some of the specific goals of the project may include:Building a robust and scalable fraud detection system that can handle a large volume of credit card transactions.Developing a model that can detect fraudulent transactions with a high degree of accuracy, while minimizing false positives.Creating a system that can adapt to changing fraud patterns and continuously improve its performance over time.Improving the overall security and trustworthiness of credit card transactions for both consumers and credit card companies.Reducing the time and resources required to investigate fraudulent activities and recover losses.Achieving these goals can help reduce the financial impact of credit card fraud and improve the overall reliability and security of credit card transactions.

4.2 PROJECT OBJECTIVES

- To develop an accurate and reliable fraud detection model that can distinguish betweenlegitimate and fraudulent credit card transactions.
- To minimize false positives (legitimate transactions that are incorrectly flagged as fraudulent)while maintaining a high level of detection accuracy.
- To ensure the model's scalability and efficiency to handle a large volume of credit cardtransactions in real-time.
- To incorporate a feedback mechanism to enable the model to learn from new data andimprove its performance over time.
- To deploy the model in a real-time system that can monitor credit card transactions andprovide alerts for potential fraudulent activities.

Chapter 5

SYSTEM REQUIREMENT SPECIFICATION

5.1 Software Requirements Analysis

A software requirements definition is an abstract description of the services, which the system should provide, and the constraints under which the system must operate. It shouldonly specify only the external behavior of the system and is not concerned with system design characteristics. It is a solution, in a natural language plus diagrams, of what services the system is expected to provide and the constraints under which it must operate.

Software Requirements

- OS: Linux, Windows 10
- Preferred browser: Google Chrome
- Client Application : Jupyter Notebook(Python)

5.2 Hardware Requirements Analysis

Hardware Requirements Analysis is to define and analyze a complete set of functional, operational, performance, interface, quality factors, and design, criticality and test requirements.

Hardware Requirements

- Processor: i5 Core Processor
- Ram: 4 GB
- Operating System: Windows 10
- Monitor: 1024*768 Resolution Color

Chapter 6

METHODOLOGY

6.1 Working-Flow of Credit Card Fraud Detection

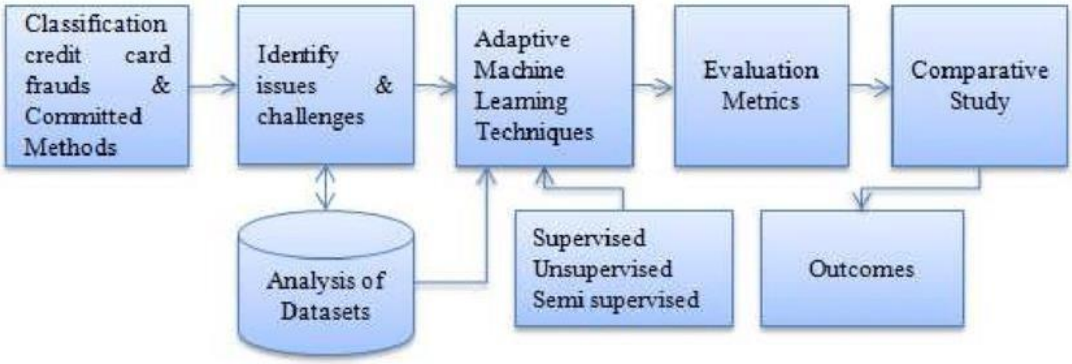

Classification credit card frauds & Committed Methods
Identify issues & challenges
Adaptive Machine Learning Techniques
Evaluation Metrics
Comparative Study
Analysis of Datasets
Supervised Unsupervised Semi supervised
Outcomes

Fig 6.1 Workflow of Credit Card Fraud Detection System

The flowchart above shows the flow of our application. The different classification techniques wehave applied in this study for fraud detection purposes are logistic regression, decision tree, random forest & KNN. Their performances are compared to see which model can better extract the relationship between the features and detect fraudulent transactions. After training all the classifiers, a new ensemble model will be applied as a voting classifier to combine all the other classification techniques. The objective is to reduce the errors of single models, which helps the ensemble model make better predictions compared with the individual classifiers. If all the classifiers are considered as $C1$, $C2$, $C3$, $C4$, and $C5$, then the final classifier will take the votes asthe majority of votes as the final prediction or Ct . $Ct = Majority\{C1, C2, C3, C4 , C5\}$In the nextstep, the data is balanced using the SMOT method to oversample the fraud transactions and under sample the normal transactions. A new ensemble model is also applied as a voting classifier to combine all the other classification techniques. The idea is that the ensemble model is stronger than the single model. The results are based on voting on the predicted classes, aiming to reduce the error. The logistic regression performance is better than the decision tree.

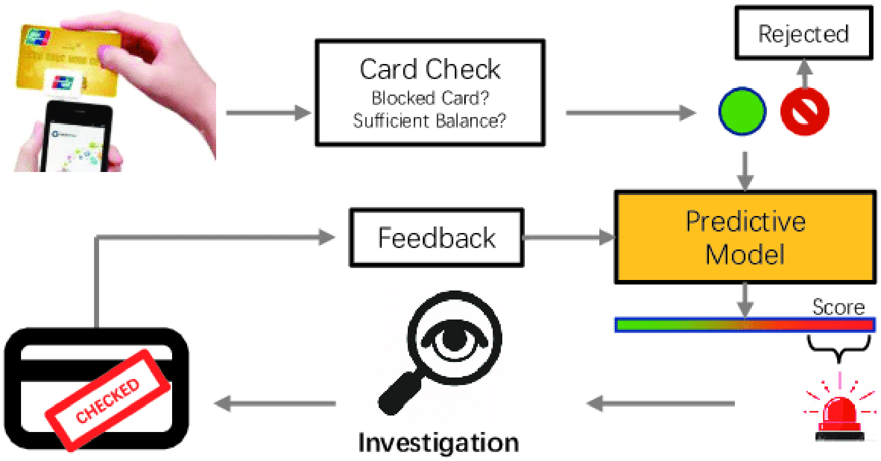


Fig. 6.2: Diagrammatic Representation

In this credit card fraud detection system project. We analyze the given dataset and apply various classification techniques like KNN, Random Forest Model to classify the values as fraudulent or not. We plot a graph and correlation matrix based on this data. We also compare the various machine learning algorithms to find out which is the best classification technique. Data Collection: The first step in credit card fraud detection is to collect data from various sources such as credit card transactions, customer information, and previous fraudulent activities. Data Preprocessing: Once the data is collected, it needs to be preprocessed to remove any inconsistencies or errors. This step involves data cleaning, normalization, and transformation. Model Selection: Once the features are extracted, a suitable machine learning algorithm is selected to build a predictive model.This can include algorithms such as logistic regression, decision trees, random forests, and neuralnetworks. Model Training: The selected model is then trained using historical data, which includes both fraudulent and non-fraudulent transactions. The model is trained to identify patterns and detect anomalies that indicate fraudulent activities. Model Evaluation: The performance of the model is evaluated using various metrics such as accuracy, precision, recall, and F1 score. The model is also tested on a separate set of data to ensure that it can generalize well and detect fraud in new transactions. Model Deployment: Once the model is evaluated and deemed to be effective,it is deployed in a production environment where it can be used to detect fraudulent activities in real-time.

Methodology 1: Loading the dataset

- Download the dataset from a reliable source like Kaggle website.
- Once you have downloaded the dataset, you need to import it into your Python environment using Pandas library.
- After loading the dataset, you can explore it to get a better understanding of its structure and content.
- Finally, you can preprocess the dataset as needed to prepare it for machine learning.

Methodology 2: Fraud identification using Machine Learning models

- Machine learning models can be used to identify fraudulent transactions by analyzing patterns in large amounts of transaction data.
- The first step is to preprocess the data by cleaning and transforming it as necessary. This may include tasks such as handling missing values, removing outliers, and normalizing or scaling the data.
- Next, features must be selected or engineered that will be used as inputs to the machine learning models.
- There are various machine learning models that can be used for credit card fraud detection, including logistic regression, decision trees, random forests, and neural networks.
- Once the model is selected, it must be trained on the preprocessed data. The performance of the model can then be evaluated using various metrics such as precision, recall, and F1 score.
- Finally, the model can be deployed in a production environment to detect fraudulent transactions in real-time.

Methodology 3: Classification and identification of fraudulent transactions

- The first step in credit card fraud detection is to identify whether a given transaction is fraudulent or not.
- Rule-based systems use a set of pre-defined rules to identify fraudulent transactions.
- Anomaly detection is another technique used to identify fraudulent transactions. This technique involves identifying transactions that deviate from the norm or expected behavior
- Machine learning models are also commonly used for identifying fraudulent transactions.

- Once a transaction is identified as potentially fraudulent, the next step is to classify it into different types of fraud.
- Once a transaction is classified, appropriate actions can be taken, such as blocking the card, contacting the cardholder, or notifying law enforcement.

Methodology 4: Enhancing accuracy using Random Forest

- Collect a dataset of credit card transactions and preprocess the data to remove any missing or erroneous values.
- Split the data into a training set and a testing set. The training set is used to train the Random Forest model, while the testing set is used to evaluate the performance of the model.
- The model will learn to identify patterns and relationships in the data that can help it distinguish between legitimate and fraudulent transactions.
- Random Forest has several hyperparameters that can be tuned to optimize the model's performance.
- Once the model is trained, use the testing set to evaluate its performance. You can calculate metrics such as precision, recall, and F1 score to assess the model's accuracy.
- The output of the Random Forest model is a probability score that indicates the likelihood of a transaction being fraudulent.

Chapter 7

IMPLEMENTATION

7.1 FILES USED

- **Jupyter Notebook:**
  - Creditcard.csv
  - CCFD.py

7.2 MODULES AND THEIR ROLES

7.2.1 PYTHON CODE USING MACHINE LEARNING ALGORITHMS

```
# import the necessary
packagesimport numpy as np
import pandas as pd
import matplotlib.pyplot as
pltimport seaborn as sns
from matplotlib import gridspec

 # Load the dataset from the csv file using
pandasdata = pd.read_csv("credit.csv")

 # Grab a peek at the
datadata.head()

 # Print the shape of the data
# data = data.sample(frac = 0.1, random_state =
48)print(data.shape)
print(data.describe())
```

```
# Determine number of fraud cases in dataset
fraud = data[data['Class'] == 1]
valid = data[data['Class'] == 0]
outlierFraction = len(fraud)/float(len(valid))
print(outlierFraction)
print('Fraud Cases: {}'.format(len(data[data['Class'] == 1])))
print('Valid Transactions: {}'.format(len(data[data['Class'] == 0])))

print(“Amount details of the fraudulent transaction”)
fraud.Amount.describe()

# Correlation matrix
corrmat = data.corr()
fig = plt.figure(figsize = (12, 9))
sns.heatmap(corrmat, vmax = .8, square = True)
plt.show()

# dividing the X and the Y from the dataset
X = data.drop(['Class'], axis = 1)
Y = data["Class"]
print(X.shape)
print(Y.shape)
# getting just the values for the sake of processing
# (its a numpy array with no columns)
xData = X.values
yData = Y.values
```

```
# Using Scikit-learn to split data into training and testing sets
from sklearn.model_selection import train_test_split
# Split the data into training and testing sets
xTrain, xTest, yTrain, yTest = train_test_split( xData, yData, test_size = 0.2, random_state = 42)

# Building the Random Forest Classifier (RANDOM FOREST)
from sklearn.ensemble import RandomForestClassifier
# random forest model creation
rfc = RandomForestClassifier()
rfc.fit(xTrain, yTrain)
# predictions
yPred = rfc.predict(xTest)

# Evaluating the classifier
# printing every score of the classifier
# scoring in anything
from sklearn.metrics import classification_report, accuracy_score
from sklearn.metrics import precision_score, recall_score
from sklearn.metrics import f1_score, matthews_corrcoef
from sklearn.metrics import confusion_matrix


n_outliers = len(fraud)
n_errors = (yPred != yTest).sum()
print("The model used is Random Forest classifier")
acc = accuracy_score(yTest, yPred)
print("The accuracy is {}".format(acc))


prec = precision_score(yTest, yPred)
print("The precision is {}".format(prec))
```

```
rec = recall_score(yTest, yPred)
print("The recall is {}".format(rec))

f1 = f1_score(yTest,  yPred)
print("The F1-Score is {}".format(f1))

MCC = matthews_corrcoef(yTest, yPred)
print("The Matthews correlation coefficient is{}".format(MCC))

# printing the confusion matrix
LABELS = ['Normal', 'Fraud']
conf_matrix = confusion_matrix(yTest, yPred)
plt.figure(figsize =(12, 12))
sns.heatmap(conf_matrix, xticklabels = LABELS,
yticklabels = LABELS, annot = True, fmt ="d");
plt.title("Confusion matrix")
plt.ylabel('True class')
plt.xlabel('Predicted class')
plt.show()
```

Credit card fraud detection using machine learning (ML) involves the use of algorithms to analyze credit card transactions and detect fraudulent activities. Here's a basic explanation of the steps involved in building a credit card fraud detection model using ML:

1. Data collection: The first step is to collect credit card transaction data that includes both legitimate and fraudulent transactions.
2. Data preparation: The collected data is then preprocessed and prepared for analysis. This includes tasks such as data cleaning, normalization, and feature engineering.
3. Feature selection: Next, the most relevant features for detecting fraud are selected. This can be done using techniques such as correlation analysis and principal component analysis (PCA).
4. Model training: Once the features are selected, a machine learning algorithm is trained on the prepared data to detect fraud. Common algorithms used for credit card fraud detection include logistic regression, decision trees, and neural networks.
5. Model evaluation: After the model is trained, it is evaluated using performance metrics such as accuracy, precision, recall, and F1 score. The model is then fine-tuned and refined to improve its performance.
6. Deployment: Once the model is deemed effective, it can be deployed to detect fraud in real-time credit card transactions.

The code for credit card fraud detection using machine learning typically involves the following steps:

1. Importing the necessary libraries and loading the data into a data frame.
2. Preprocessing the data by cleaning, normalizing, and transforming the features as needed.
3. Splitting the data into training and testing sets.
4. Selecting the appropriate machine learning algorithm and training the model on the training set.
5. Evaluating the model's performance on the testing set and fine-tuning the model as needed.
6. Deploying the model to detect fraud in real-time credit card transactions.

Overall, credit card fraud detection using machine learning involves a combination of data preparation, feature selection, algorithm selection, and model evaluation. By following these steps and fine-tuning the model, organizations can build effective fraud detection systems to protect themselves and their customers from fraudulent activities.

This code is importing several Python libraries that are commonly used in data analysis and visualization:

1. numpy (imported as np) is a library for numerical computing in Python.
2. pandas (imported as pd) is a library for data manipulation and analysis.
3. matplotlib.pyplot (imported as plt) is a library for creating visualizations, such as graphs and charts.
4. seaborn is a data visualization library built on top of matplotlib that provides additional functionality and styling options.
5. gridspec is a matplotlib module for creating subplots of varying sizes and layouts.

The code is using the from keyword to import the gridspec module directly from the matplotlib library.

The syntax import x as y allows the programmer to give a library a different name within the script. In this code snippet, numpy is being given the name np, pandas is being given the name pd, and matplotlib.pyplot is being given the name plt.

Only 0.17% fraudulent transaction out all the transactions. The data is highly Unbalanced. Lets first apply our models without balancing it and if we don’t get a good accuracy then we can find a way to balance this dataset. But first, let’s implement the model without it and will balance the data only if needed.

After implementing Correlation Matrix, we can observe in the HeatMap we can clearly see that most of the features do not correlate to other features but there are some features that either has a positive or a negative correlation with each other. For example, V2 and V5 are highly negatively correlated with the feature called Amount. We also see some correlation with V20 and Amount. This gives us a deeper understanding of the Data available to us.

This code is building a random forest classifier using the RandomForestClassifier class from the sklearn.ensemble module in scikit-learn. The random forest model is trained on the training set (xTrain and yTrain), and then used to make predictions on the test set (xTest). The predicted labels are stored in the yPred variable.

The code then evaluates the performance of the random forest classifier using various metrics such as accuracy, precision, recall, F1 score, and the Matthews correlation coefficient (MCC). These metrics are calculated using the predicted labels (yPred) and the actual labels (yTest).

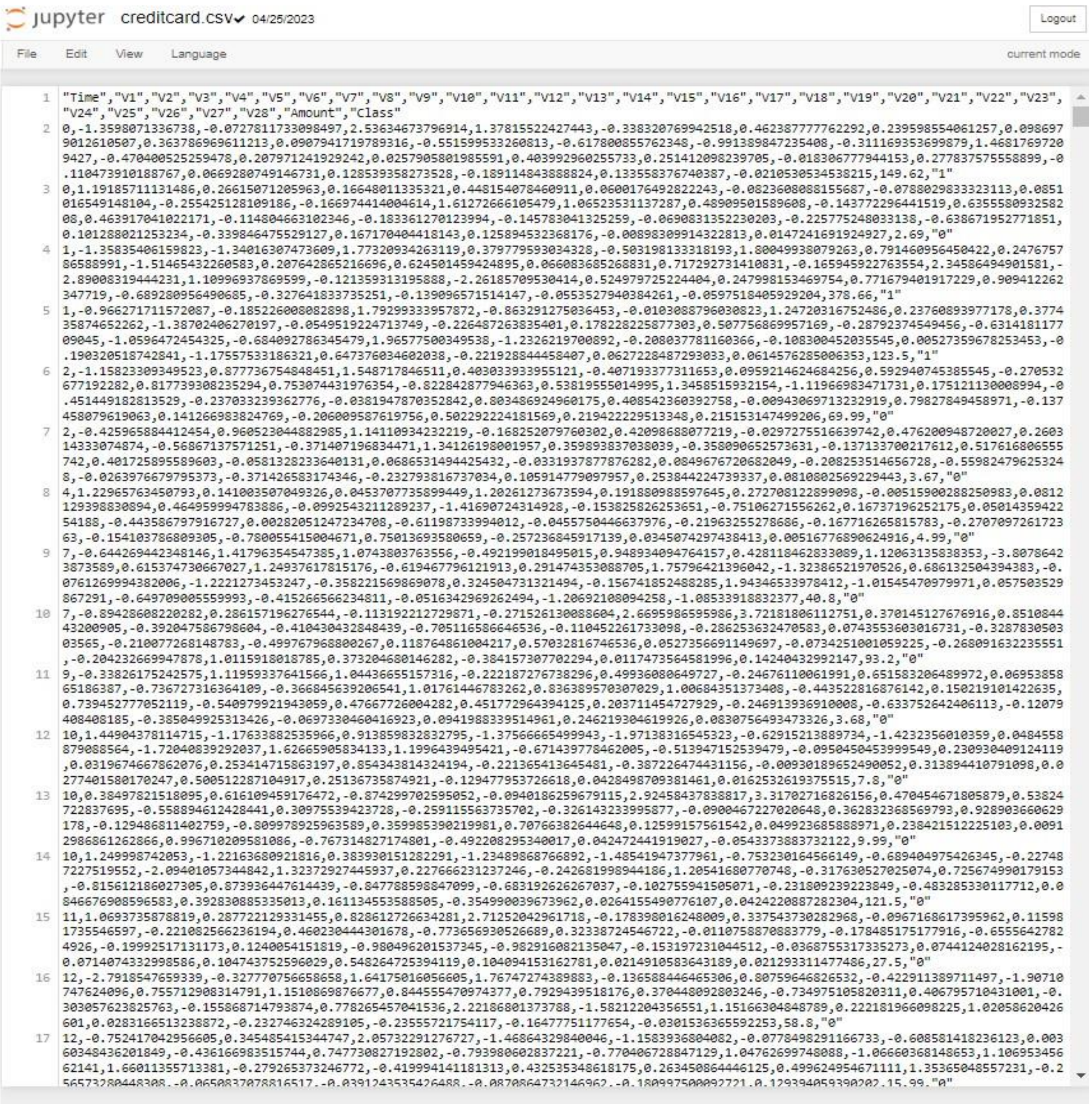
The confusion matrix is also generated using the confusion_matrix function from scikit-learn's metrics module. The matrix is visualized using a heatmap from the **seaborn** library.

Finally, the evaluation metrics and confusion matrix are printed and plotted.

Chapter 8

RESULTS & SNAPSHOTS

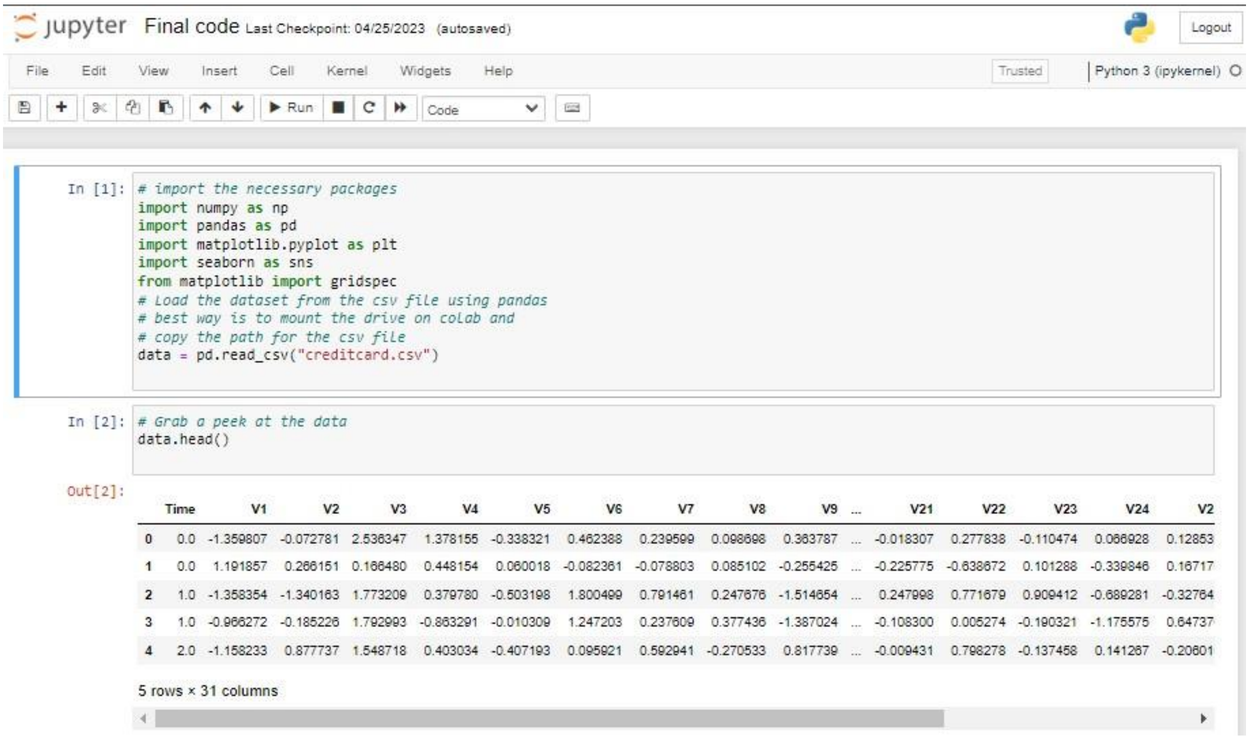
8.1 INPUT FILE



```
"Time","V1","V2","V3","V4","V5","V6","V7","V8","V9","V10","V11","V12","V13","V14","V15","V16","V17","V18","V19","V20","V21","V22","V23",
"V24","V25","V26","V27","V28","Amount","Class"
0,-1.3598071336738,-0.0727811733098497,2.53634673796914,1.37815522427443,-0.338320769942518,0.462387777762292,0.239598554061257,0.098697
9012610507,0.363786969611213,0.0907941719789316,-0.551599533260813,-0.617800855762348,-0.991389847235408,-0.311169353699879,1.4681769720
9427,-0.470400525259478,0.207971241929242,0.0257905801985591,0.403992960255733,0.251412098239705,-0.018306777944153,0.277837575558899,-0
.110473910188767,0.0669280749146731,0.128539358273528,-0.189114843888824,0.133558376740387,-0.0210530534538215,149.62,"1"
0,1.19185711131486,0.26615071205963,0.16648011335321,0.448154078460911,0.0600176492822243,-0.0823608088155687,-0.0788029833323113,0.0851
016549148104,-0.255425128109186,-0.166974414004614,1.61272666105479,1.06523531137287,0.48909501589608,-0.143772296441519,0.6355580932582
08,0.463917041022171,-0.114804663102346,-0.183361270123994,-0.145783041325259,-0.0690831352230203,-0.225775248033138,-0.638671952771851,
0.101288021253234,-0.339846475529127,0.167170404418143,0.125894532368176,-0.00898309914322813,0.0147241691924927,2.69,"0"
1,-1.35835406159823,-1.34016307473609,1.77320934263119,0.379779593034328,-0.503198133318193,1.80049938079263,0.791460956450422,0.2476757
86588991,-1.51465432260583,0.207642865216696,0.624501459424895,0.066083685268831,0.717292731410831,-0.165945922763554,2.34586494901581,-
2.89008319444231,1.10996937869599,-0.121359313195888,-2.26185709530414,0.524979725224404,0.247998153469754,0.771679401917229,0.909412262
347719,-0.689280956490685,-0.327641833735251,-0.139096571514147,-0.0553527940384261,-0.0597518405929204,378.66,"1"
1,-0.966271711572087,-0.185226008082898,1.79299333957872,-0.863291275036453,-0.0103088796030823,1.24720316752486,0.23760893977178,0.3774
35874652262,-1.38702406270197,-0.0549519224713749,-0.226487263835401,0.178228225877303,0.507756869957169,-0.28792374549456,-0.6314181177
09045,-1.0596472454325,-0.684092786345479,1.96577500349538,-1.2326219700892,-0.208037781160366,-0.108300452035545,0.00527359678253453,-0
.190320518742841,-1.17557533186321,0.647376034602038,-0.221928844458407,0.0627228487293033,0.0614576285006353,123.5,"1"
2,-1.15823309349523,0.877736754848451,1.548717846511,0.403033933955121,-0.407193377311653,0.0959214624684256,0.592940745385545,-0.270532
677192282,0.817739308235294,0.753074431976354,-0.822842877946363,0.53819555014995,1.3458515932154,-1.11966983471731,0.175121130008994,-0
.451449182813529,-0.237033239362776,-0.0381947870352842,0.803486924960175,0.408542360392758,-0.00943069713232919,0.79827849458971,-0.137
458079619063,0.141266983824769,-0.206009587619756,0.502292224181569,0.219422229513348,0.215153147499206,69.99,"0"
2,-0.425965884412454,0.960523044882985,1.14110934232219,-0.168252079760302,0.42098688077219,-0.0297275516639742,0.476200948720027,0.2603
14333074874,-0.56867137571251,-0.371407196834471,1.34126198001957,0.359893837038039,-0.358090652573631,-0.137133700217612,0.517616806555
742,0.401725895589603,-0.0581328233640131,0.0686531494425432,-0.0331937877876282,0.0849676720682049,-0.208253514656728,-0.55982479625324
8,-0.0263976679795373,-0.371426583174346,-0.232793816737034,0.105914779097957,0.253844224739337,0.0810802569229443,3.67,"0"
4,1.22965763450793,0.141003507049326,0.0453707735899449,1.20261273673594,0.191880988597645,0.272708122899098,-0.00515900288250983,0.0812
129398830894,0.464959994783886,-0.0992543211289237,-1.41690724314928,-0.153825826253651,-0.75106271556262,0.16737196252175,0.05014359422
54188,-0.443586797916727,0.00282051247234708,-0.61198733994012,-0.0455750446637976,-0.21963255278686,-0.167716265815783,-0.2707097261723
63,-0.154103786809305,-0.780055415004671,0.75013693580659,-0.257236845917139,0.0345074297438413,0.00516776890624916,4.99,"0"
7,-0.644269442348146,1.41796354547385,1.0743803763556,-0.492199018495015,0.948934094764157,0.428118462833089,1.12063135838353,-3.8078642
3873589,0.615374730667027,1.24937617815176,-0.619467796121913,0.291474353088705,1.75796421396042,-1.32386521970526,0.686132504394383,-0.
0761269994382006,-1.2221273453247,-0.358221569869078,0.324504731321494,-0.156741852488285,1.94346533978412,-1.01545470979971,0.057503529
867291,-0.649709005559993,-0.415266566234811,-0.0516342969262494,-1.20692108094258,-1.08533918832377,40.8,"0"
7,-0.89428608220282,0.286157196276544,-0.113192212729871,-0.271526130088604,2.6695986595986,3.72181806112751,0.370145127676916,0.8510844
43200905,-0.392047586798604,-0.410430432848439,-0.705116586646536,-0.110452261733098,-0.286253632470583,0.0743553603016731,-0.3287830503
03565,-0.210077268148783,-0.499767968800267,0.118764861004217,0.57032816746536,0.0527356691149697,-0.0734251001059225,-0.268091632235551
,-0.204232669947878,1.0115918018785,0.373204680146282,-0.384157307702294,0.0117473564581996,0.14240432992147,93.2,"0"
9,-0.33826175242575,1.11959337641566,1.04436655157316,-0.222187276738296,0.49936080649727,-0.24676110061991,0.651583206489972,0.06953858
65186387,-0.736727316364109,-0.366845639206541,1.01761446783262,0.836389570307029,1.00684351373408,-0.443522816876142,0.150219101422635,
0.739452777052119,-0.540979921943059,0.47667726004282,0.451772964394125,0.203711454727929,-0.246913936910008,-0.633752642406113,-0.12079
408408185,-0.385049925313426,-0.0697330460416923,0.0941988339514961,0.246219304619926,0.0830756493473326,3.68,"0"
10,1.44904378114715,-1.17633882535966,0.913859832832795,-1.37566665499943,-1.97138316545323,-0.62915213889734,-1.4232356010359,0.0484558
879088564,-1.72040839292037,1.62665905834133,1.1996439495421,-0.671439778462005,-0.513947152539479,-0.0950450453999549,0.230930409124119
,0.0319674667862076,0.253414715863197,0.854343814324194,-0.221365413645481,-0.387226474431156,-0.00930189652490052,0.313894410791098,0.0
277401580170247,0.500512287104917,0.25136735874921,-0.129477953726618,0.0428498709381461,0.0162532619375515,7.8,"0"
10,0.38497821518095,0.616109459176472,-0.874299702595052,-0.0940186259679115,2.92458437838817,3.31702716826156,0.470454671805879,0.53824
722837695,-0.558894612428441,0.30975539423728,-0.259115563735702,-0.326143233995877,-0.0900467227020648,0.362832368569793,0.928903660629
178,-0.129486811402759,-0.809978925963589,0.359985390219981,0.70766382644648,0.12599157561542,0.049923685888971,0.238421512225103,0.0091
2986861262866,0.996710209581086,-0.767314827174801,-0.492208295340017,0.042472441919027,-0.0543373883732122,9.99,"0"
10,1.249998742053,-1.22163680921816,0.383930151282291,-1.23489868766892,-1.48541947377961,-0.753230164566149,-0.689404975426345,-0.22748
7227519552,-2.09401057344842,1.32372927445937,0.227666231237246,-0.242681998944186,1.20541680770748,-0.317630527025074,0.725674990179153
,-0.815612186027305,0.873936447614439,-0.847788598847099,-0.683192626267037,-0.102755941505071,-0.231809239223849,-0.483285330117712,0.0
846676908596583,0.392830885335013,0.161134553588505,-0.354990039673962,0.0264155490776107,0.0424220887282304,121.5,"0"
11,1.0693735878819,0.287722129331455,0.828612726634281,2.71252042961718,-0.178398016248009,0.337543730282968,-0.0967168617395962,0.11598
1735546597,-0.221082566236194,0.460230444301678,-0.773656930526689,0.32338724546722,-0.0110758870883779,-0.178485175177916,-0.6555642782
4926,-0.19992517131173,0.1240054151819,-0.980496201537345,-0.982916082135047,-0.153197231044512,-0.0368755317335273,0.0744124028162195,-
0.0714074332998586,0.104743752596029,0.548264725394119,0.104094153162781,0.0214910583643189,0.021293311477486,27.5,"0"
12,-2.7918547659339,-0.327770756658658,1.64175016056605,1.76747274389883,-0.136588446465306,0.80759646826532,-0.422911389711497,-1.90710
747624096,0.755712908314791,1.1510869876677,0.844555470974377,0.7929439518176,0.370448092803246,-0.734975105820311,0.406795710431001,-0.
303057623825763,-0.155868714793874,0.778265457041536,2.22186801373788,-1.58212204356551,1.15166304848789,0.222181966098225,1.02058620426
601,0.0283166513238872,-0.232746324289105,-0.23555721754117,-0.16477751177654,-0.0301536365592253,58.8,"0"
12,-0.752417042956605,0.345485415344747,2.05732291276727,-1.46864329840046,-1.1583936804082,-0.0778498291166733,-0.608581418236123,0.003
60348436201849,-0.436166983515744,0.747730827192802,-0.793980602837221,-0.770406728847129,1.04762699748088,-1.06660368148653,1.106953456
62141,1.66011355713381,-0.279265373246772,-0.419994141181313,0.432535348618175,0.263450864446125,0.499624954671111,1.35365048557231,-0.2
56573280448308,-0.0650837078816517,-0.0391243535426488,-0.0870864732146962,-0.180997500092721,0.129394059390202,15.99,"0"
```

Fig. 8.1: Dataset Values

8.2 UPLOADING LIBRARIES AND CHECKING DATA HEAD


jupyter
Final code Last Checkpoint: 04/25/2023 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3 (ipykernel)
Run
Code

In [1]:
```
# import the necessary packages
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from matplotlib import gridspec
# Load the dataset from the csv file using pandas
# best way is to mount the drive on colab and
# copy the path for the csv file
data = pd.read_csv("creditcard.csv")
```

In [2]:
```
# Grab a peek at the data
data.head()
```

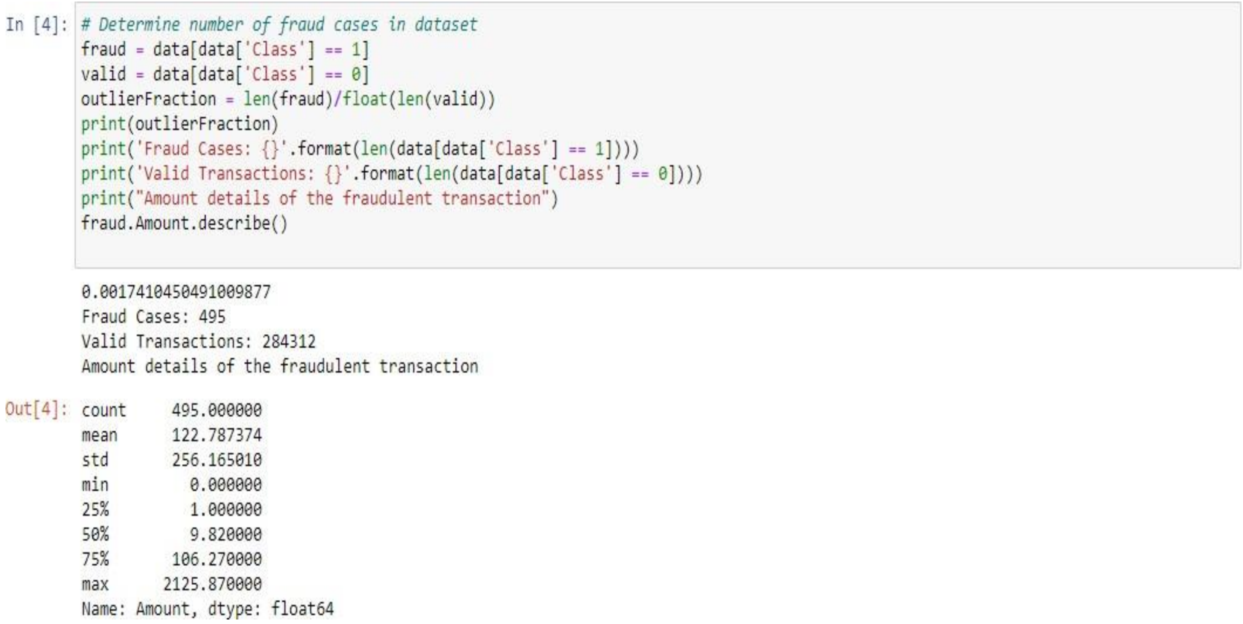
Out[2]:

	Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	...	V21	V22	V23	V24	V2
0	0.0	-1.359807	-0.072781	2.536347	1.378155	-0.338321	0.462388	0.239599	0.098698	0.363787	...	-0.018307	0.277838	-0.110474	0.066928	0.12853
1	0.0	1.191857	0.266151	0.166480	0.448154	0.060018	-0.082361	-0.078803	0.085102	-0.255425	...	-0.225775	-0.638672	0.101288	-0.339846	0.16717
2	1.0	-1.358354	-1.340163	1.773209	0.379780	-0.503198	1.800499	0.791461	0.247676	-1.514654	...	0.247998	0.771679	0.909412	-0.689281	-0.32764
3	1.0	-0.966272	-0.185226	1.792993	-0.863291	-0.010309	1.247203	0.237609	0.377436	-1.387024	...	-0.108300	0.005274	-0.190321	-1.175575	0.64737
4	2.0	-1.158233	0.877737	1.548718	0.403034	-0.407193	0.095921	0.592941	-0.270533	0.817739	...	-0.009431	0.798278	-0.137458	0.141267	-0.20601

5 rows × 31 columns

Fig. 8.2: Libraries and Data Head

8.3 CHECKING TOTAL NUMBER OF FRAUD CASES



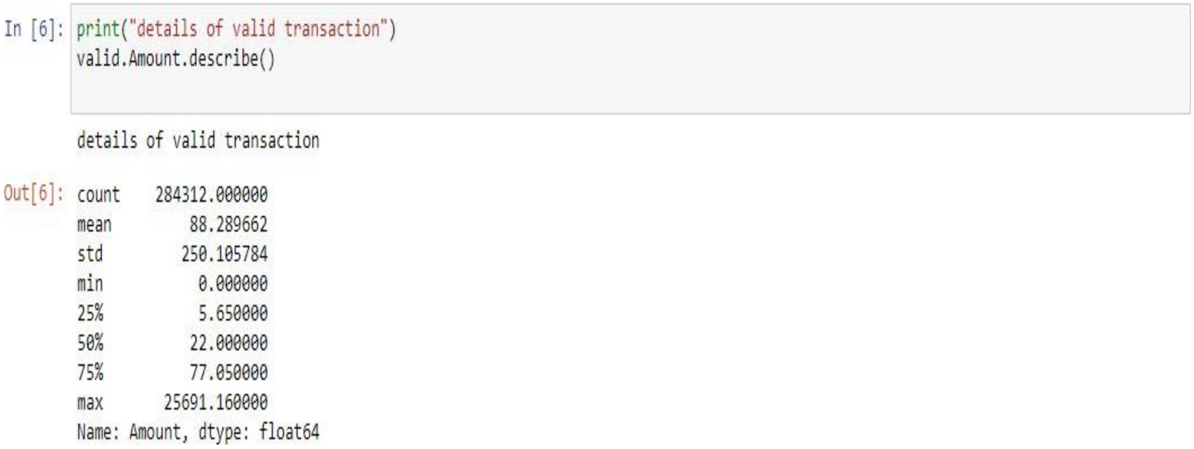
In [4]:
```
# Determine number of fraud cases in dataset
fraud = data[data['Class'] == 1]
valid = data[data['Class'] == 0]
outlierFraction = len(fraud)/float(len(valid))
print(outlierFraction)
print('Fraud Cases: {}'.format(len(data[data['Class'] == 1])))
print('Valid Transactions: {}'.format(len(data[data['Class'] == 0])))
print("Amount details of the fraudulent transaction")
fraud.Amount.describe()
```

```
0.0017410450491009877
Fraud Cases: 495
Valid Transactions: 284312
Amount details of the fraudulent transaction
```

Out[4]:
```
count     495.000000
mean      122.787374
std       256.165010
min         0.000000
25%         1.000000
50%         9.820000
75%       106.270000
max      2125.870000
Name: Amount, dtype: float64
```

Fig. 8.3: Displays Total number of Fraud Cases

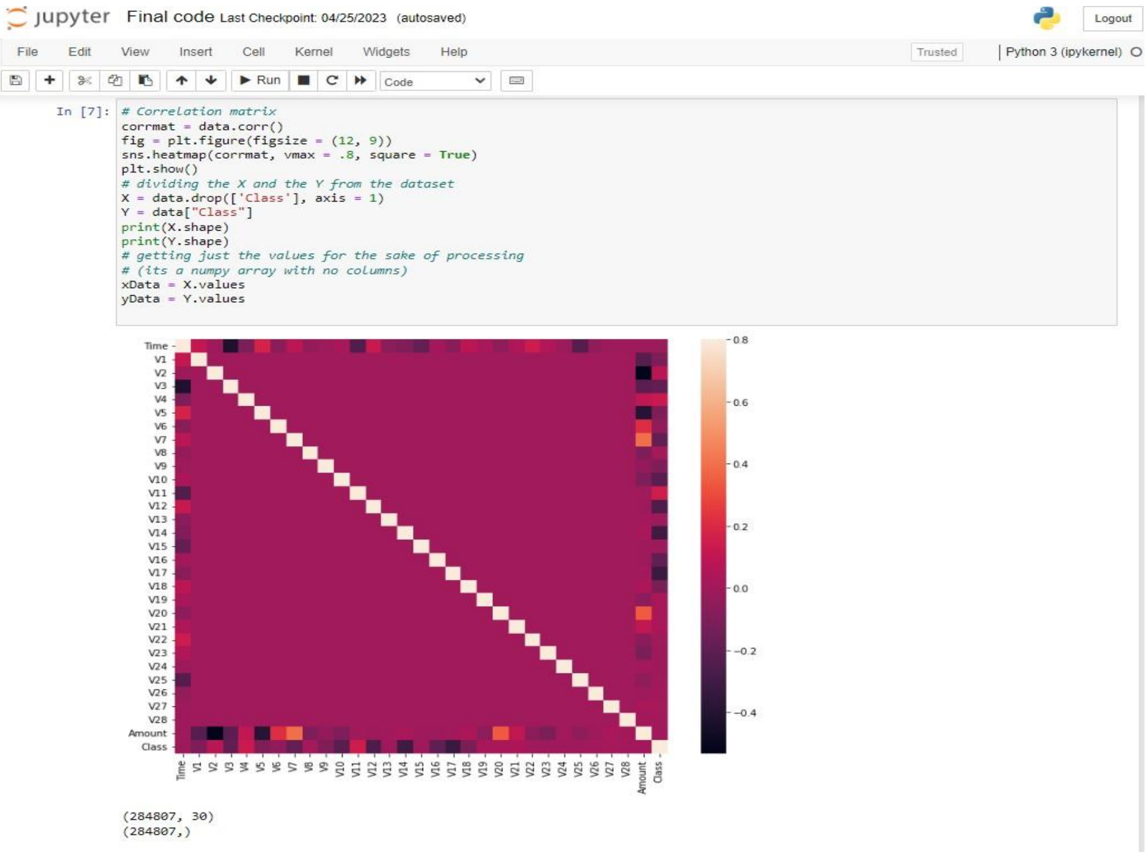
8.4 CHECKING TOTAL NUMBER OF VALID CASES



```
In [6]: print("details of valid transaction")
        valid.Amount.describe()

        details of valid transaction

Out[6]: count    284312.000000
        mean         88.289662
        std         250.105784
        min           0.000000
        25%           5.650000
        50%          22.000000
        75%          77.050000
        max       25691.160000
        Name: Amount, dtype: float64
```

Fig. 8.4: Displays Total number of Valid Cases

8.5 CORRELATION MATRIX

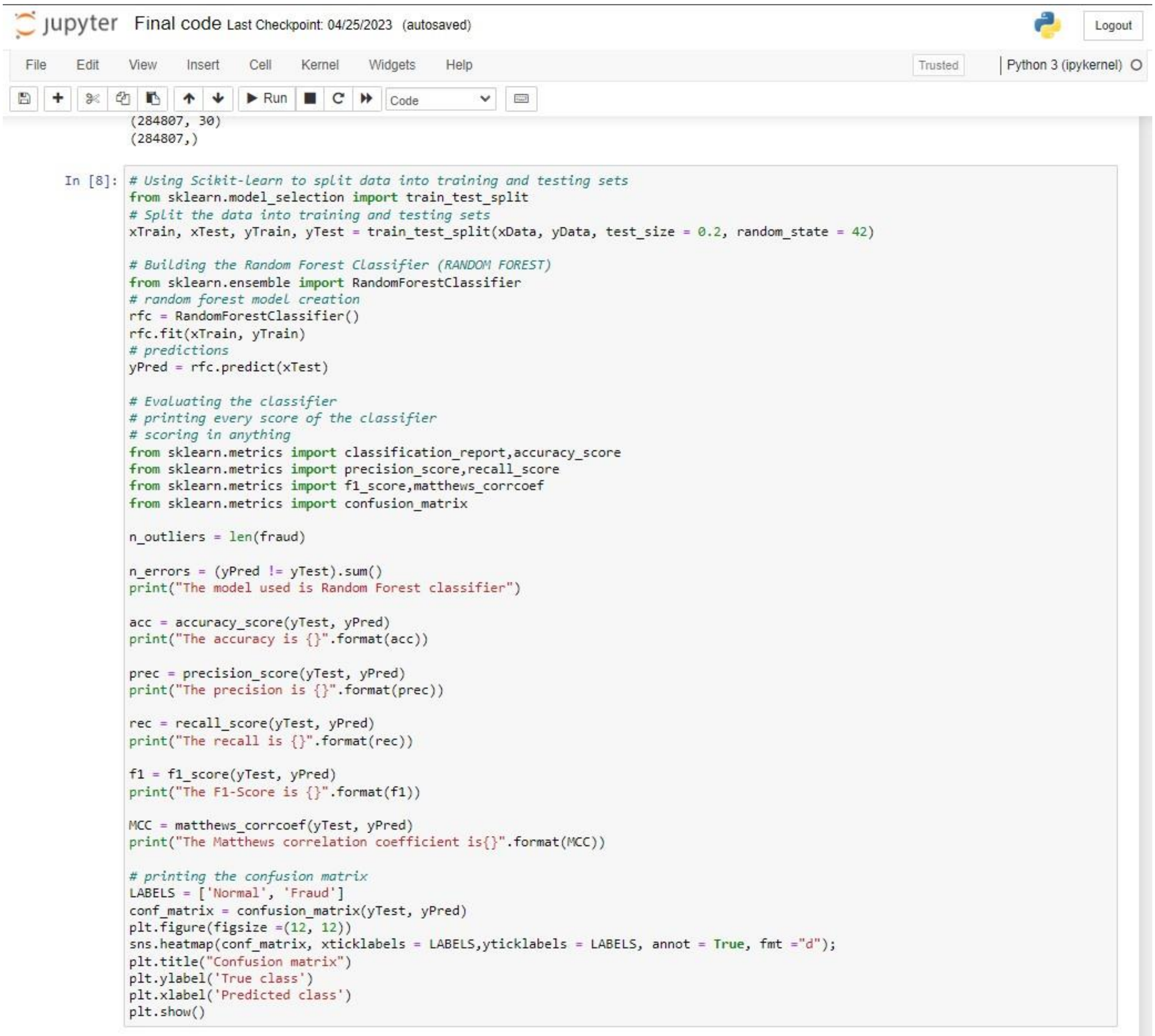


```
In [7]: # Correlation matrix
        corrmat = data.corr()
        fig = plt.figure(figsize = (12, 9))
        sns.heatmap(corrmat, vmax = .8, square = True)
        plt.show()
        # dividing the X and the Y from the dataset
        X = data.drop(['Class'], axis = 1)
        Y = data["Class"]
        print(X.shape)
        print(Y.shape)
        # getting just the values for the sake of processing
        # (its a numpy array with no columns)
        xData = X.values
        yData = Y.values
```

```
(284807, 30)
(284807,)
```

Fig. 8.5: Correlation Matrix

8.6 CODE FOR CONFUSION MATRIX

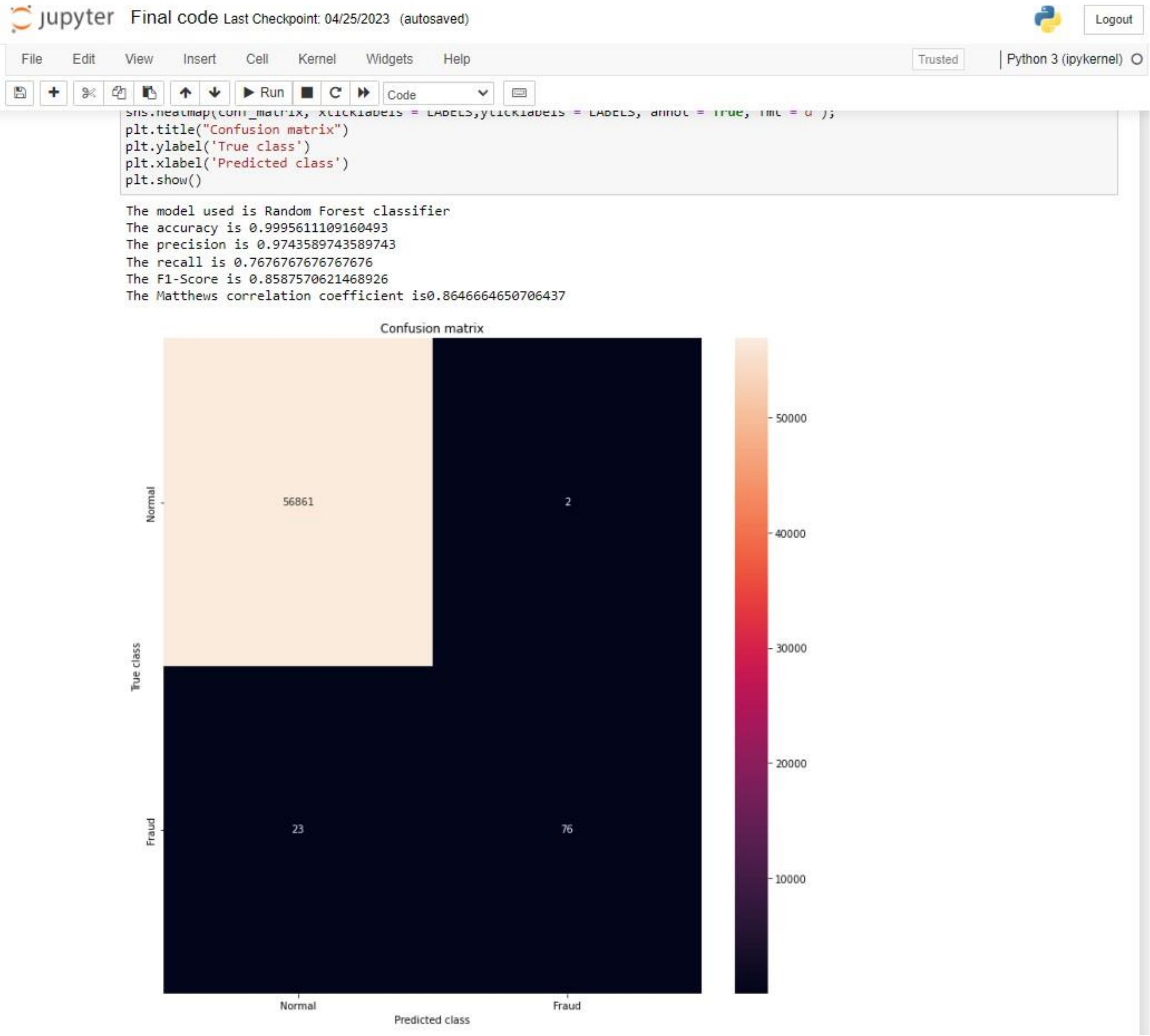

jupyter Final code Last Checkpoint: 04/25/2023 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3 (ipykernel)
Run
Code

```
(284807, 30)
(284807,)
```

```
In [8]: # Using Scikit-learn to split data into training and testing sets
        from sklearn.model_selection import train_test_split
        # Split the data into training and testing sets
        xTrain, xTest, yTrain, yTest = train_test_split(xData, yData, test_size = 0.2, random_state = 42)

        # Building the Random Forest Classifier (RANDOM FOREST)
        from sklearn.ensemble import RandomForestClassifier
        # random forest model creation
        rfc = RandomForestClassifier()
        rfc.fit(xTrain, yTrain)
        # predictions
        yPred = rfc.predict(xTest)

        # Evaluating the classifier
        # printing every score of the classifier
        # scoring in anything
        from sklearn.metrics import classification_report,accuracy_score
        from sklearn.metrics import precision_score,recall_score
        from sklearn.metrics import f1_score,matthews_corrcoef
        from sklearn.metrics import confusion_matrix

        n_outliers = len(fraud)

        n_errors = (yPred != yTest).sum()
        print("The model used is Random Forest classifier")

        acc = accuracy_score(yTest, yPred)
        print("The accuracy is {}".format(acc))

        prec = precision_score(yTest, yPred)
        print("The precision is {}".format(prec))

        rec = recall_score(yTest, yPred)
        print("The recall is {}".format(rec))

        f1 = f1_score(yTest, yPred)
        print("The F1-Score is {}".format(f1))

        MCC = matthews_corrcoef(yTest, yPred)
        print("The Matthews correlation coefficient is{}".format(MCC))

        # printing the confusion matrix
        LABELS = ['Normal', 'Fraud']
        conf_matrix = confusion_matrix(yTest, yPred)
        plt.figure(figsize =(12, 12))
        sns.heatmap(conf_matrix, xticklabels = LABELS,yticklabels = LABELS, annot = True, fmt ="d");
        plt.title("Confusion matrix")
        plt.ylabel('True class')
        plt.xlabel('Predicted class')
        plt.show()
```

Fig. 8.6: Training ,Testing and Splitting Dataset

8.7 CONFUSION MATRIX GRAPH



```
sns.heatmap(conf_matrix, xticklabels = LABELS,yticklabels = LABELS, annot = True, fmt = 'd');
plt.title("Confusion matrix")
plt.ylabel('True class')
plt.xlabel('Predicted class')
plt.show()
```

```
The model used is Random Forest classifier
The accuracy is 0.9995611109160493
The precision is 0.9743589743589743
The recall is 0.7676767676767676
The F1-Score is 0.8587570621468926
The Matthews correlation coefficient is0.8646664650706437
```

Fig. 8.7: Confusion Matrix Graph

Chapter 9 APPLICATIONS

Overall, credit card fraud detection has a wide range of applications in various industries and areas. By implementing effective fraud detection systems, organizations can prevent financial losses, increase customer trust, and protect themselves and their customers from fraudulent cases.Credit card fraud detection has a wide range of applications in different industries and areas. Some of the common applications of credit card fraud detection are:

- **Banking and finance**: The banking and finance sector is the primary industry where credit card fraud detection is used. Banks and financial institutions use fraud detection systems to monitor credit card transactions and prevent fraudulent activities. This helps them to reduce financial losses, increase customer trust, and comply with regulatory requirements.
- **E-commerce:** Credit card fraud is a major concern for e-commerce businesses. Fraud detection systems are used to monitor transactions and identify fraudulent activities such as chargebacks and friendly fraud. This helps e-commerce businesses to prevent financial losses and protect their reputation.
- **Insurance:** Insurance companies use credit card fraud detection systems to prevent fraudulent insurance claims. Fraudulent claims can result in significant financial losses for insurance companies, and fraud detection systems help to prevent such losses.
- **Government:** Credit card fraud detection is also used by government agencies to prevent tax fraud, welfare fraud, and other forms of fraudulent activities that involve the use of credit cards.
- **Healthcare:** Healthcare providers also use credit card fraud detection systems to prevent fraudulent insurance claims and prevent healthcare fraud.

Chapter 10

CONCLUSIONS & FUTURE WORK

10.1 CONCLUSION

This project helps in predicting fraudulent transactions for the given dataset. We are implementing ML algorithms like KNN, DecisionTree, Random Forest, Logistic Regression and choosing the most accurate and time efficient.We are enhancing credit card fraud detection which is a priority for banks and financial organizations.We are identifying suspicious events and reporting them to an analyst while letting normal transactions to be automatically processed.

In conclusion, credit card fraud detection is a crucial area of focus for financial institutions, businesses, and law enforcement agencies. Credit card fraud can result in significant financial losses for individuals and organizations, undermine consumer confidence in e-commerce, and fuel other forms of criminal activity.

By implementing effective credit card fraud detection measures, organizations can reduce the risk of fraudulent transactions and improve the overall security and reliability of payment systems. Credit card fraud detection projects often involve the use of advanced technologies such as machine learning, data analytics, and artificial intelligence. As these technologies continue to evolve and improve, credit card fraud detection is likely to become even more effective and efficient in the future.

10.2 FUTURE ENHANCEMENTS

- Minimizing errors, improving accuracy and keeping the design as simple as possible.
- Making testing and training data equal in the referred dataset.
- Location tracing and blocking of credit cards for better security when fraudulent transaction is detected.
- Improved data sharing between financial institutions and law enforcement agencies can help prevent credit card fraud on a larger scale.
- Biometric authentication, such as fingerprint or facial recognition, can help prevent fraud by providing an additional layer of security.

Chapter 11

CONTRIBUTION TO SOCIETY AND ENVIRONMENT

Credit card fraud detection projects can help prevent financial losses for individuals and organizations by detecting fraudulent transactions and preventing them from going through. By implementing effective credit card fraud detection measures, consumers can feel more confident in using their credit cards for online transactions, which can help drive e-commerce growth. Credit card fraud is often linked to other forms of criminal activity, such as identity theft and money laundering. By detecting and preventing credit card fraud, law enforcement agencies can disrupt criminal networks and prevent further crimes.Credit card fraud detection projects can help identify vulnerabilities in payment systems and inform the development of more secure payment technologies.

Credit card fraud detection projects often involve the use of advanced technologies such as machine learning, data analytics, and artificial intelligence. By developing and refining these technologies, credit card fraud detection projects can contribute to the advancement of the tech industry as a whole. Overall, credit card fraud detection projects can help create a safer, more secure, and more trustworthy financial system for individuals and businesses alike.

REFERENCES

[1] Omar, Kazi Shahrukh, Prodipta Mondal, Nabila Shahnaz Khan, Md Rezaul Karim Rizvi, and Md Nazrul Islam. "A machine learning approach to predict autism spectrum disorder." In 2019 International conference on electrical, computer and communication engineering (ECCE),IEEE, 2019.

[2] Akter,Tania, Md Shahriare Satu, Md Imran Khan, Mohammad Hanif Ali, Shahadat Uddin, Pietro Lio, Julian MW Quinn, and Mohammad Ali Moni. "Machine learning-based models for early stage detection of autism spectrum disorders." IEEE,2019.

[3] Baranwal, Astha, and M. Vanitha. "Autistic spectrum disorder screening: prediction with machine learning models." In 2020 International conference on emerging trends in information technology and engineering (ic-ETITE), pp. 1-7. IEEE,2020.

[4] Eni, M, Dinstein, I, Ilan, M., Menashe, I, Meiri, G, & Zigel, Y. Estimating Autism Severity in Young Children From Speech Signals Using a Deep Neural Network. IEEE 2020.

[5] Raj, S, & Masood, S. (2020). Analysis and Detection of Autism Spectrum Disorder Using Machine Learning Techniques. Procedia Computer Science.

[6] https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud

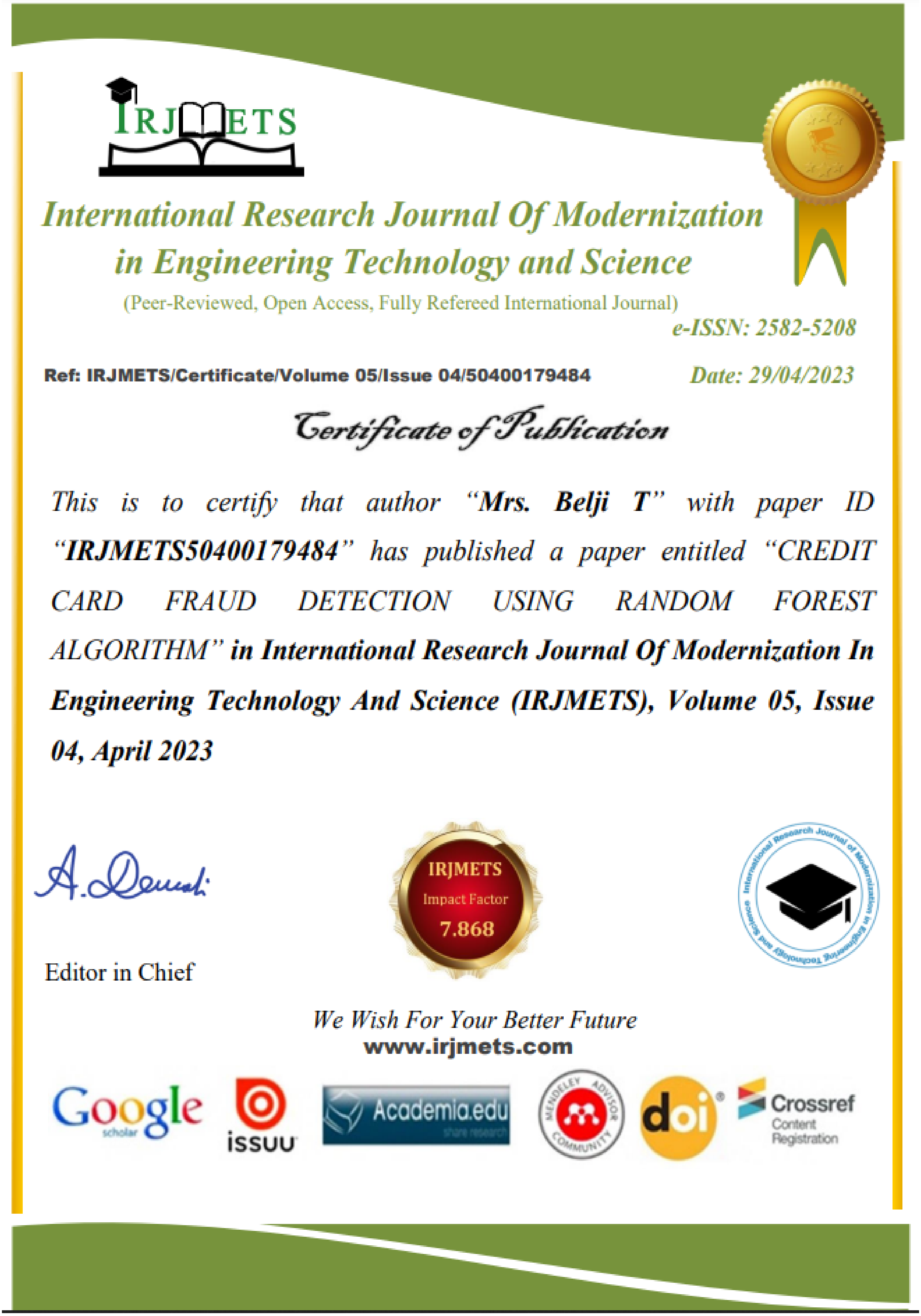
[7] https://www.geeksforgeeks.org/ml-credit-card-fraud-detection/

[8] https://www.youtube.com/watch?v=NCgjcHLFNDg/

[9] Aswathy M S, Liji Sameul “Survey on Credit Card Fraud Detection”.

APPENDIX-I

CERTIFICATES OF PAPER PRESENTED



IRJMETS

International Research Journal Of Modernization in Engineering Technology and Science

(Peer-Reviewed, Open Access, Fully Refereed International Journal)

e-ISSN: 2582-5208

Ref: IRJMETS/Certificate/Volume 05/Issue 04/50400179484 *Date: 29/04/2023*

Certificate of Publication

*This is to certify that author "**Mrs. Belji T**" with paper ID "**IRJMETS50400179484**" has published a paper entitled "CREDIT CARD FRAUD DETECTION USING RANDOM FOREST ALGORITHM" **in International Research Journal Of Modernization In Engineering Technology And Science (IRJMETS), Volume 05, Issue 04, April 2023***

Editor in Chief

IRJMETS
Impact Factor
7.868

International Research Journal of Modernization in Engineering Technology and Science

We Wish For Your Better Future
www.irjmets.com

Google scholar issuu Academia.edu share research MENDELEY ADVISOR COMMUNITY doi Crossref Content Registration

IRJMETS

International Research Journal Of Modernization in Engineering Technology and Science

(Peer-Reviewed, Open Access, Fully Refereed International Journal)

e-ISSN: 2582-5208

Ref: IRJMETS/Certificate/Volume 05/Issue 04/50400179484

Date: 29/04/2023

Certificate of Publication

*This is to certify that author "**Bhargav M**" with paper ID "**IRJMETS50400179484**" has published a paper entitled "CREDIT CARD FRAUD DETECTION USING RANDOM FOREST ALGORITHM" **in International Research Journal Of Modernization In Engineering Technology And Science (IRJMETS), Volume 05, Issue 04, April 2023***

Editor in Chief

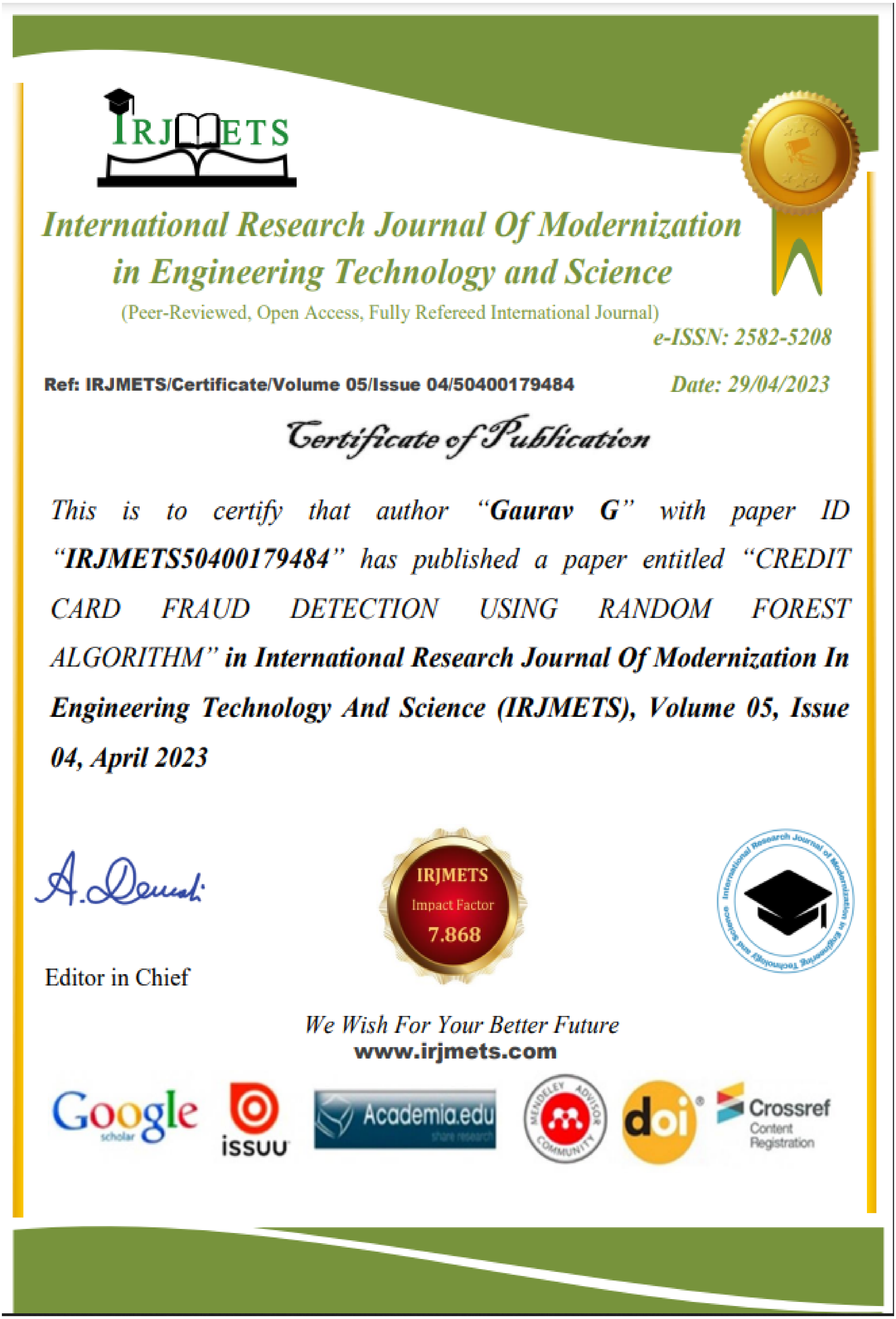
IRJMETS
Impact Factor
7.868

International Research Journal of Modernization in Engineering Technology and Science

We Wish For Your Better Future

www.irjmets.com

Google scholar issuu Academia.edu share research MENDELEY ADVISOR COMMUNITY doi Crossref Content Registration



IRJMETS

International Research Journal Of Modernization in Engineering Technology and Science

(Peer-Reviewed, Open Access, Fully Refereed International Journal)

e-ISSN: 2582-5208

Ref: IRJMETS/Certificate/Volume 05/Issue 04/50400179484 Date: 29/04/2023

Certificate of Publication

*This is to certify that author “**Gaurav G**” with paper ID “**IRJMETS50400179484**” has published a paper entitled “CREDIT CARD FRAUD DETECTION USING RANDOM FOREST ALGORITHM” **in International Research Journal Of Modernization In Engineering Technology And Science (IRJMETS), Volume 05, Issue 04, April 2023***

Editor in Chief

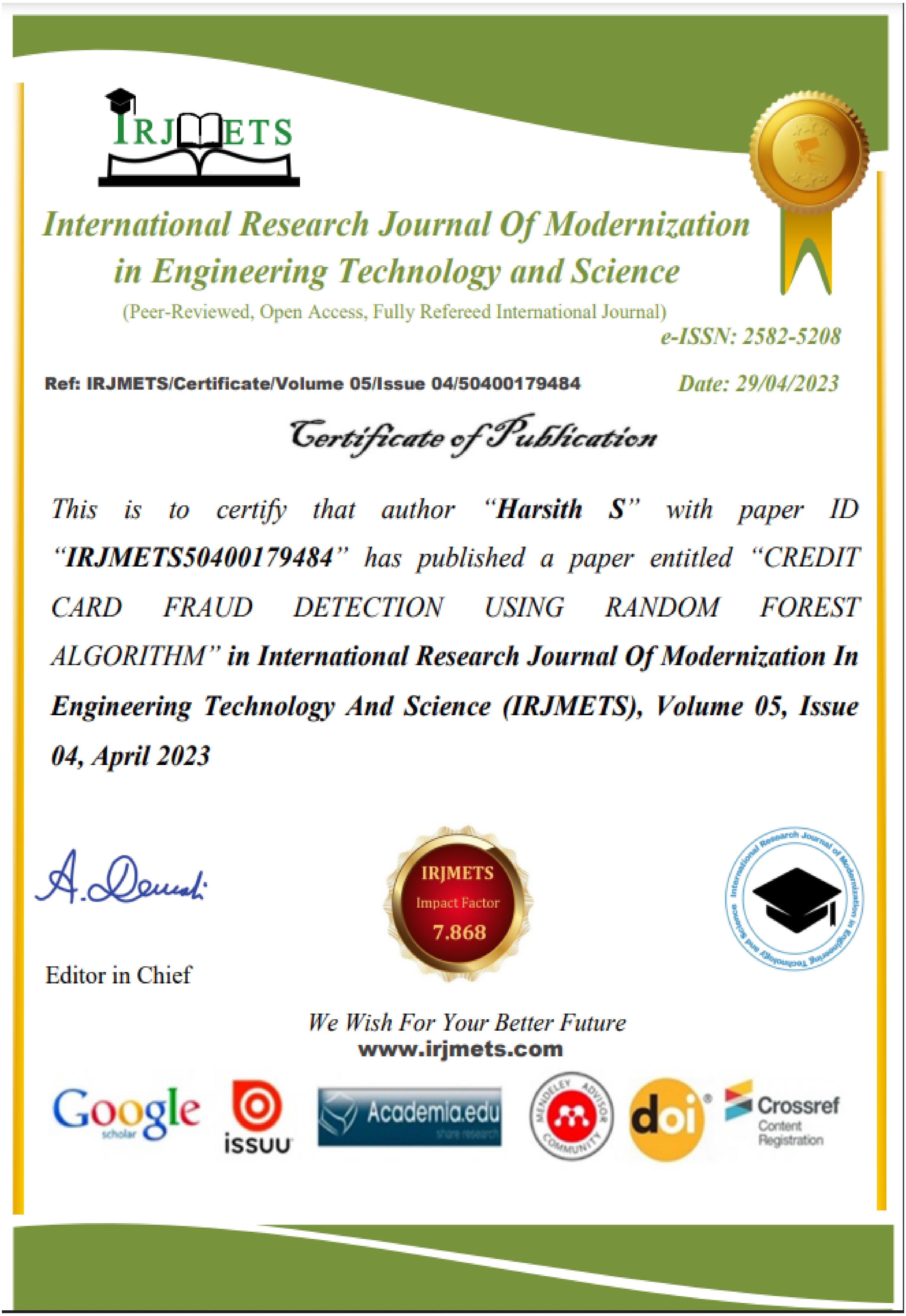
IRJMETS
Impact Factor
7.868

International Research Journal of Modernization in Engineering Technology and Science

We Wish For Your Better Future

www.irjmets.com

Google scholar · issuu · Academia.edu share research · Mendeley Advisor Community · doi · Crossref Content Registration



IRJMETS

International Research Journal Of Modernization in Engineering Technology and Science

(Peer-Reviewed, Open Access, Fully Refereed International Journal)

e-ISSN: 2582-5208

Ref: IRJMETS/Certificate/Volume 05/Issue 04/50400179484

Date: 29/04/2023

Certificate of Publication

*This is to certify that author "**Harsith S**" with paper ID "**IRJMETS50400179484**" has published a paper entitled "CREDIT CARD FRAUD DETECTION USING RANDOM FOREST ALGORITHM" **in International Research Journal Of Modernization In Engineering Technology And Science (IRJMETS), Volume 05, Issue 04, April 2023***

Editor in Chief

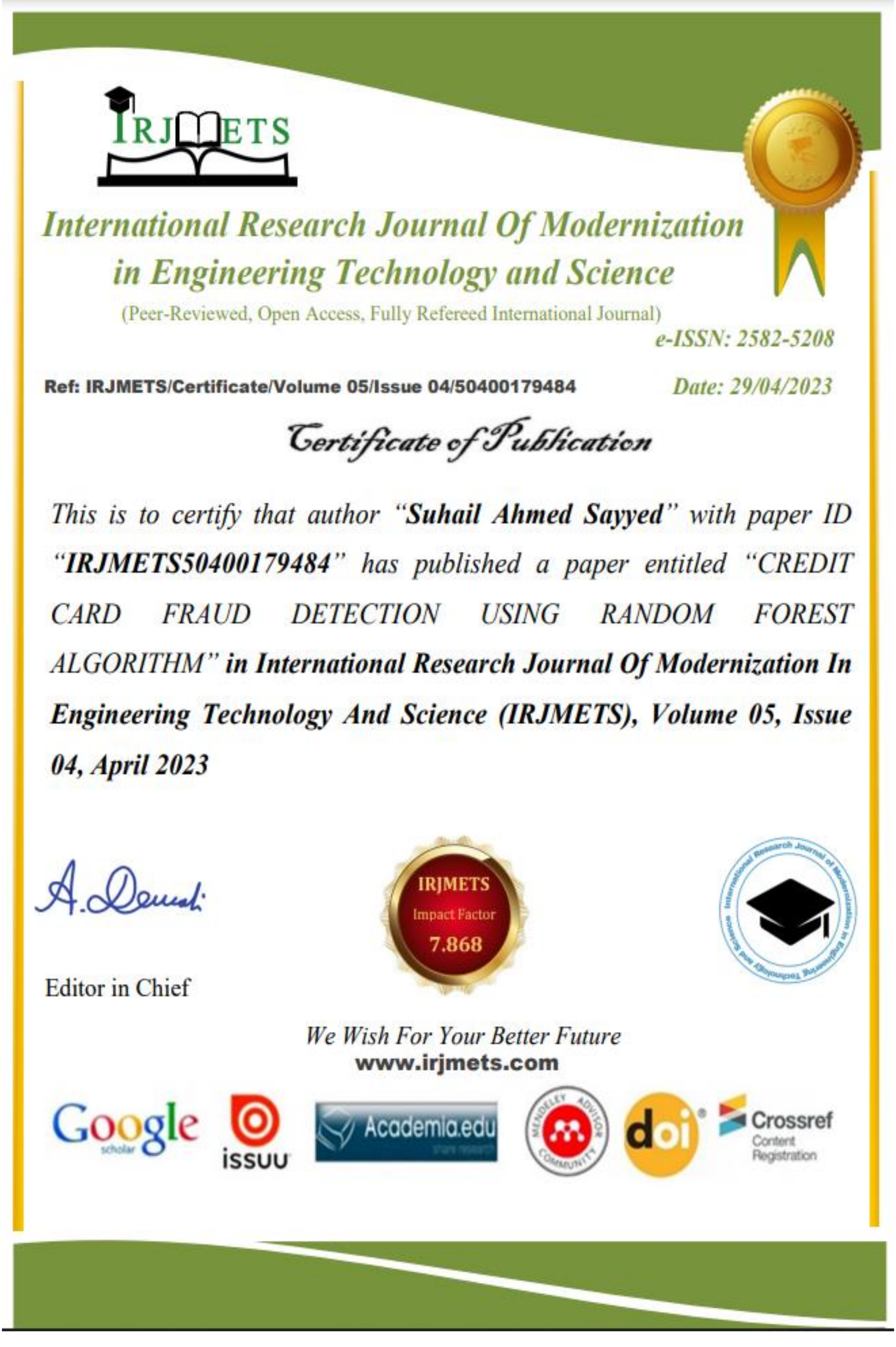
IRJMETS
Impact Factor
7.868

International Research Journal of Modernization in Engineering Technology and Science

We Wish For Your Better Future

www.irjmets.com

Google scholar · issuu · Academia.edu · Mendeley Advisor Community · doi · Crossref Content Registration



IRJMETS

International Research Journal Of Modernization in Engineering Technology and Science

(Peer-Reviewed, Open Access, Fully Refereed International Journal)

e-ISSN: 2582-5208

Ref: IRJMETS/Certificate/Volume 05/Issue 04/50400179484 *Date: 29/04/2023*

Certificate of Publication

*This is to certify that author "**Suhail Ahmed Sayyed**" with paper ID "**IRJMETS50400179484**" has published a paper entitled "CREDIT CARD FRAUD DETECTION USING RANDOM FOREST ALGORITHM" **in International Research Journal Of Modernization In Engineering Technology And Science (IRJMETS), Volume 05, Issue 04, April 2023***

Editor in Chief

IRJMETS Impact Factor 7.868

We Wish For Your Better Future

www.irjmets.com

Google scholar issuu Academia.edu Mendeley Advisor Community doi Crossref Content Registration

APPENDIX-II

JOURNAL PUBLISHED PAPER