CREDIT CARD FRAUD DETECTION

PHASE 5: PROJECT DOCUMENTATION AND SUBMISSION

TOPIC: START BUILDING THE CREDIT CARD DETECTION PROJECT BY LOADING AND PRE-PROCESSING THE DATASET



Introduction:

* **Data Collection**: Financial institutions collect a vast amount of data related to card transactions, including transaction amount, location, date and time, merchant information, and more. This data forms the basis for fraud detection.
* **Machine Learning and Artificial Intelligence**: Machine learning algorithms and AI models play a pivotal role in analyzing the data. They are trained to identify patterns, anomalies, and trends associated with legitimate and fraudulent transactions. Common algorithms include decision trees, neural networks, logistic regression, and random forests.
* **Rules-Based Systems**: In addition to machine learning, rules-based systems are used to set predefined rules that can trigger alerts or actions when certain conditions are met. These rules may include transaction amount limits, transaction frequency thresholds, and location-based checks.
* **Behavioral Analysis**: By monitoring a cardholder's behavior over time, financial institutions can establish a baseline of normal activity. Deviations from this baseline, such as an unusual transaction in a different location or an unusually large purchase, can raise red flags.
* **Real-time Monitoring**: Many systems operate in real-time, continuously assessing incoming transactions as they occur. This allows for immediate responses to potential fraud, such as flagging a transaction for manual review or blocking a card.
* **Historical Data Analysis**: Historical transaction data is invaluable for identifying recurring fraud patterns. By analyzing past incidents, financial institutions can refine their fraud detection models and rules.
* **Machine Learning Model Training**: Regular updates and retraining of machine learning models are essential to adapt to evolving fraud tactics. New data helps models become more accurate in recognizing fraudulent behavior.
* **Customer Verification**: Multi-factor authentication, including something the cardholder knows (e.g., PIN), something the cardholder has (e.g., the card itself), and something the cardholder is (e.g., biometrics), can provide an extra layer of security.
* **Collaboration**: Financial institutions often collaborate with industry organizations and share fraud intelligence to stay ahead of emerging threats and trends.
* **Customer Alerts**: Providing cardholders with alerts for suspicious activity via text messages, emails, or mobile apps allows them to quickly report unauthorized transactions.

**Types of credit card fraud**

 Credit card fraud falls into two basic categories:

1. Card present fraud
2. Card-not-present fraud

**Card present fraud**

**Card present fraud** is when the criminal uses a physical card, which is either stolen or duplicated, to make fraudulent purchases. Card present fraud can be the result of the theft of a card through robbery, pickpocketing, or mail theft.

 Criminals may also leverage card skimmers installed at frequently used payment points to collect and store the card details when swiped; this data can then be used to produce a duplicate payment card, or clone.

**Card-not-present fraud**

**Card-not-present fraud**is when the criminal uses the details associated with the card, such as the card number, accountholder name, and CVV code, without having the card in their possession.

 In some cases, card-not-present crime is accompanied by account takeover techniques. This is when fraudsters contact a credit card issuer and purport to be a legitimate card holder to change information associated with the account, such as a phone number or address. This will allow them to verify purchases and authenticate activity, thereby evading many fraud detection tools.

**Advances in card security**

Since 2015, credit card fraud has shifted dramatically toward card-not-present crime. This is largely because of EMV (Europay, MasterCard and Visa) chip technology introduced by major credit card companies around the world.

Sometimes referred to as chip and PIN, EMV equips credit and debit cards with a microchip that can be inserted into a card reader or scanned via contactless payment method to process the transaction. This method is far more secure in that criminals can no longer use card skimmers to capture data from the magnetic strip of the card and produce a clone.



At the same time, massive data breaches at various credit institutions, retailers and other businesses have created a black market for compromised credit card data and sensitive information. Cybercriminals that have stolen this information often post it on the dark web, making it available for purchase by other individuals. The availability of compromised card information, coupled with a rise in online shopping, has dramatically increased card-not-present credit card fraud.

At present, there is no technological solution similar to an EMV microchip that can significantly **a**ssist in card-not-present fraud prevetion for credit cards. Instead, the burden is on both credit card issuers and merchants to employ stronger digital security measures to identify and block fraudulent transactions in real-time.

### **Credit card fraud detection**

Credit card fraud detection is the collective term for the policies, tools, methodologies, and practices that credit card companies and financial institutions take to combat identity fraud and stop fraudulent transactions.

In recent years, as the amount of data has exploded and the number of payment card transactions has skyrocketed, credit fraud detection has become largely digitized and automated.. Most modern solutions leverage artificial intelligence (AI) and machine learning (ML) to manage data analysis, predictive modeling, decision-making, fraud alerts and remediation activity that occur when individual instances of credit card fraud are detected.



#### ‍****Anomaly detection****

**Anomaly detection** is the process of analyzing massive amounts of data points from both internal and external sources to produce a framework of “normal” activity for each individual user and establish regular patterns in their activity.

 Data used to create the user profile includes:

* Purchase history and other historical data
* Location
* Device ID
* IP address
* Payment amount
* Transaction information

When a transaction falls outside the scope of normal activity, the anomaly detection tool will then alert the card issuer and, in some cases, the user. Depending on the transaction details and risk score assigned to the action, these fraud detection systems may flag the purchase for review or put a hold on the transaction until the user verifies their activity.

#### ****What can be an anomaly?****

* A sudden increase in spending
* Purchase of a large ticket item
* A series of rapid transactions
* Multiple transactions with the same merchant
* Transactions that originate in an unusual location or foreign country
* Transactions that occur at unusual times

 If the anomaly detection tool leverages ML, the models can also be self-learning, meaning that they will constantly gather and analyze new data to update the existing model and provide a more precise scope of acceptable activity for the user.

 It is important to note that the most advanced algorithms are based on the individual user’s behaviors and transaction history. Therefore, the model could make exceptions and allow transactions that are usually considered high-risk for the wider user base.

 For example, for a business executive who travels internationally frequently, a rule may allow her to make routine purchases in select countries where she has a history of similar activity. On the other hand, a cardholder who rarely leaves their state likely will not be able to make any international purchases without first alerting their bank to upcoming travel plans.

#### ****Predictive modeling****

 In addition to finding anomalies within a specific user account, ML models and predictive analytics can also be used to track and identify fraud patterns or point to an ongoing, nuanced fraud scheme. **Predictive modeling**is an important capability since cybercriminals are constantly updating their techniques to evade detection by existing tools and methods.

#### ****Outlier models****

the name implies, an outlier model identifies suspicious activity in the cases when not enough data is available to predict patterns.

For example, if a user has only used their debit card to make small, regular purchases within their local area and the card is suddenly used to purchase a large ticket item, that could  Finally, some anomaly detection tools are also equipped with **outlier models**. As indicate fraud. In this case, the tool would flag the transaction for further review, or request the credit card holder confirm their purchase before processing the transaction.

#### ****Combating fraudulent activities with Inscribe****

  At Inscribe, we're on a mission to create a more fair and efficient financial services ecosystem. That’s we’ve built best-in-class document fraud detection software, and then layed our document intelligence and processing automation features on top of that.

Our suite of 20+ fraud detectors examine the metadata, pixel-level information, and more in documents to ensure the integrity of the information presented. Additionally, we compare documents to templates of known fraudulent and known legitimate documents. [**Detecting fraud in documents**](https://www.inscribe.ai/blog/how-inscribe-detects-fraudulent-customers) is Inscribe’s competitive advantage in the market for risk management.

Over the past year we have onboarded some of the world's fastest growing fintechs and numerous Fortune 500 companies, helping them to reduce fraud and give quicker decisions to their customers. In April 2021, we surpassed $40M of fraud detected per month for our customers. And we’ve helped our customers [**upload and analyze thousands of documents**](https://www.inscribe.ai/customers/amount) for fraud accurately in just seconds.

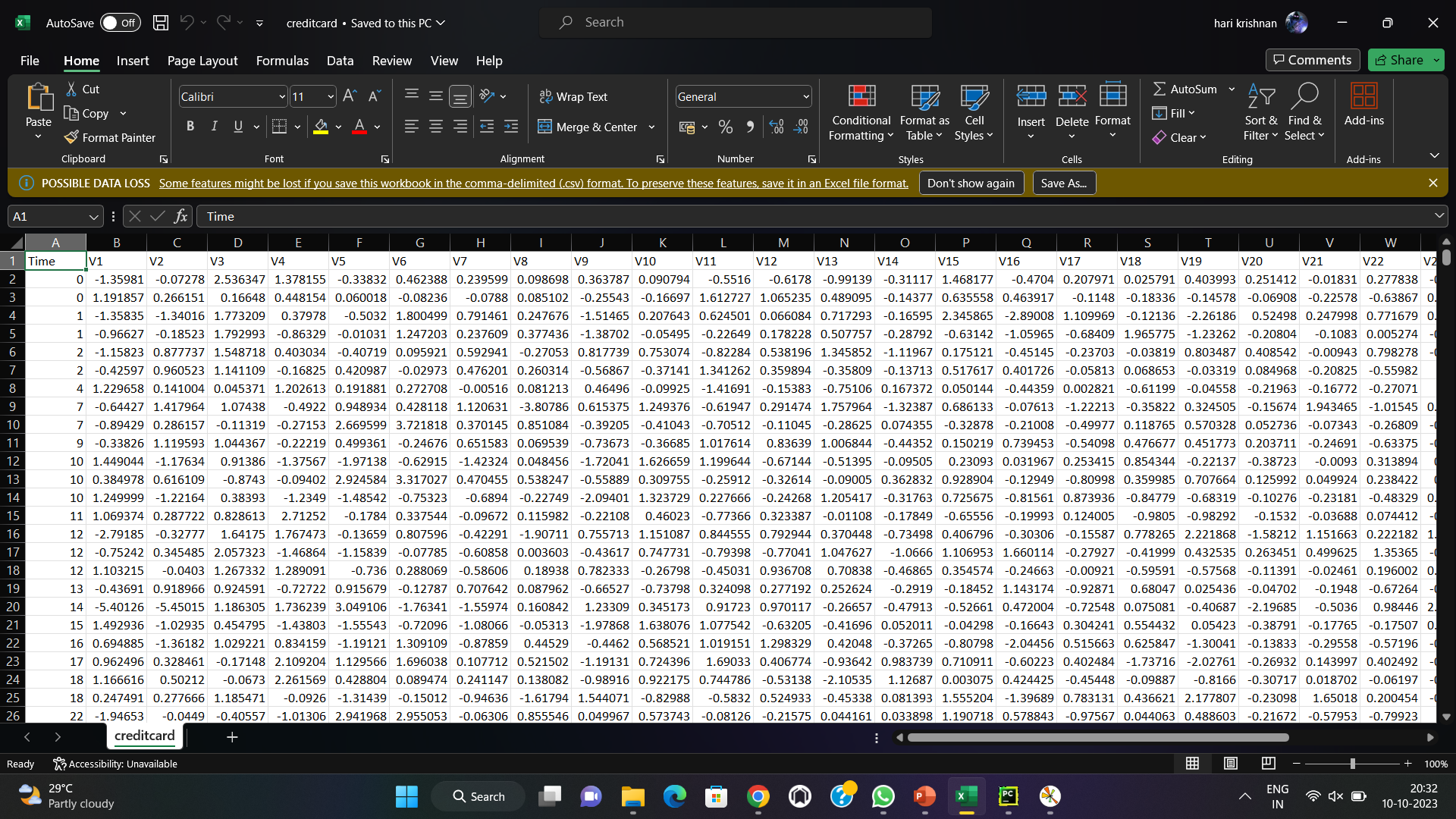
 Want to learn more about how Inscribe can help your organization combat fraud? Check out our recent article about [**new features we released to help our customers fight fraud**](https://www.inscribe.ai/blog/5-new-inscribe-features-to-help-you-fight-fraud-in-2022), or contact our sales team to[**schedule a personalized demo**](https://www.inscribe.ai/contact).

Dataset:

The dataset for the given credit card fraud detection project can be downloaded from the link given below.

<https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud/code>

The given dataset is picturized as below:



Loading and pre-processing of dataset:

Loading and pre-processing the dataset is an important first step in building up any machine learning model. However ,it is especially important for credit card fraud detection dataset are complex and noisy.

1.LOADING THE DATASET:

* To load a dataset for credit card fraud detection, you can follow these steps:
* as Pandas, **Choose a Dataset**: There are several publicly available datasets for credit card fraud detection, which you can use for research or model development. One of the most commonly used datasets is the Credit Card Fraud Detection dataset available on Kaggle. You can download this dataset or use other reputable sources like the UCI Machine Learning Repository.
* **Import Python Libraries**: You'll typically use Python for data analysis and machine learning. Import libraries such NumPy, and Scikit-Learn to manipulate and work with the dataset.

PROGRAM:

Since we are given two datasets, we are going to load both of these datasets separately.

import pandas as pd

import numpy as np

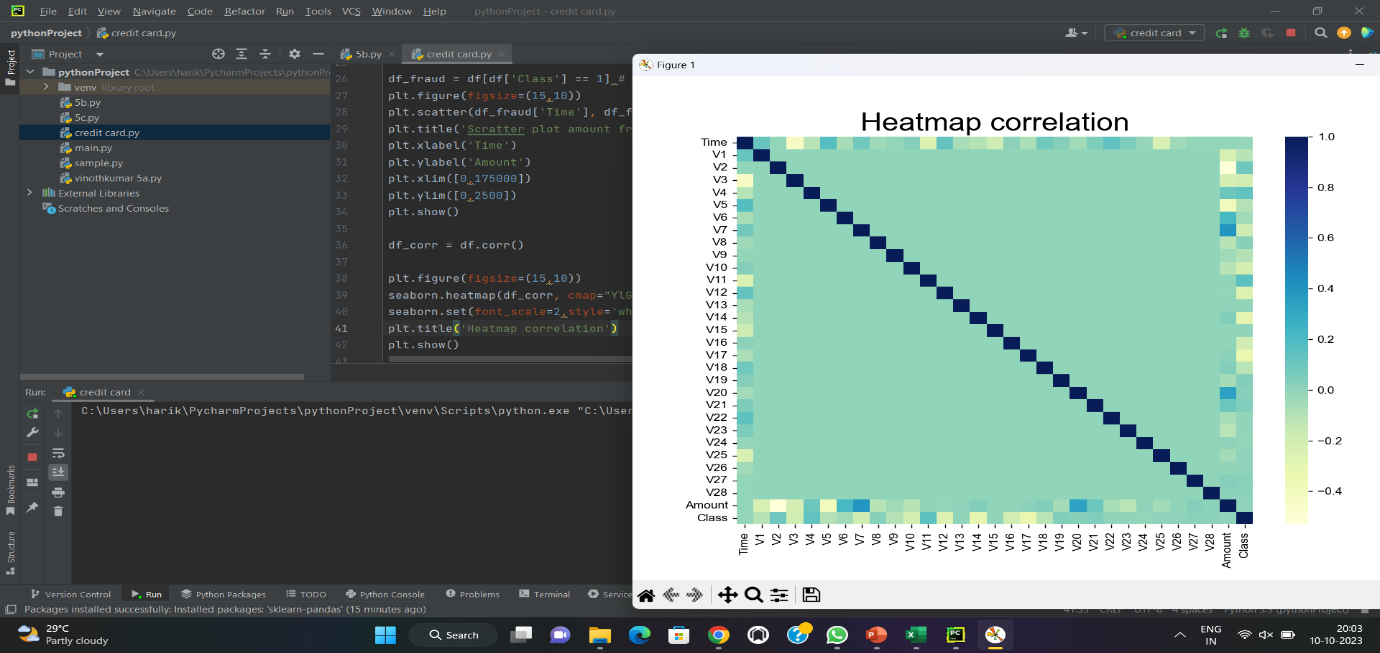
data = pd.read\_csv('filename.csv')

from sklearn.model\_selection import train\_test\_split

X = data.drop('Class', axis=1)

y = data['Class']

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_ size=0.3, random\_state=42)

Output:

2.PRE-PROCESSING THE DATASET:

Preprocessing the data in a credit card fraud detection dataset is a crucial step to ensure that the data is ready for machine learning. The goal of preprocessing is to clean and prepare the data, making it suitable for training a model that can effectively detect fraudulent transactions. Here are the common preprocessing steps for credit card fraud detection:

import pandas as pd

import numpy as np

*# Scikit-learn library: For SVM*

from sklearn import preprocessing

from sklearn.metrics import confusion\_matrix

from sklearn import svm

df\_fraud = df[df['Class'] == 1] *# Recovery of fraud data*

plt.figure(figsize=(15,10))

plt.scatter(df\_fraud['Time'], df\_fraud['Amount']) *# Display fraud amounts according to their time*

plt.title('Scratter plot amount fraud')

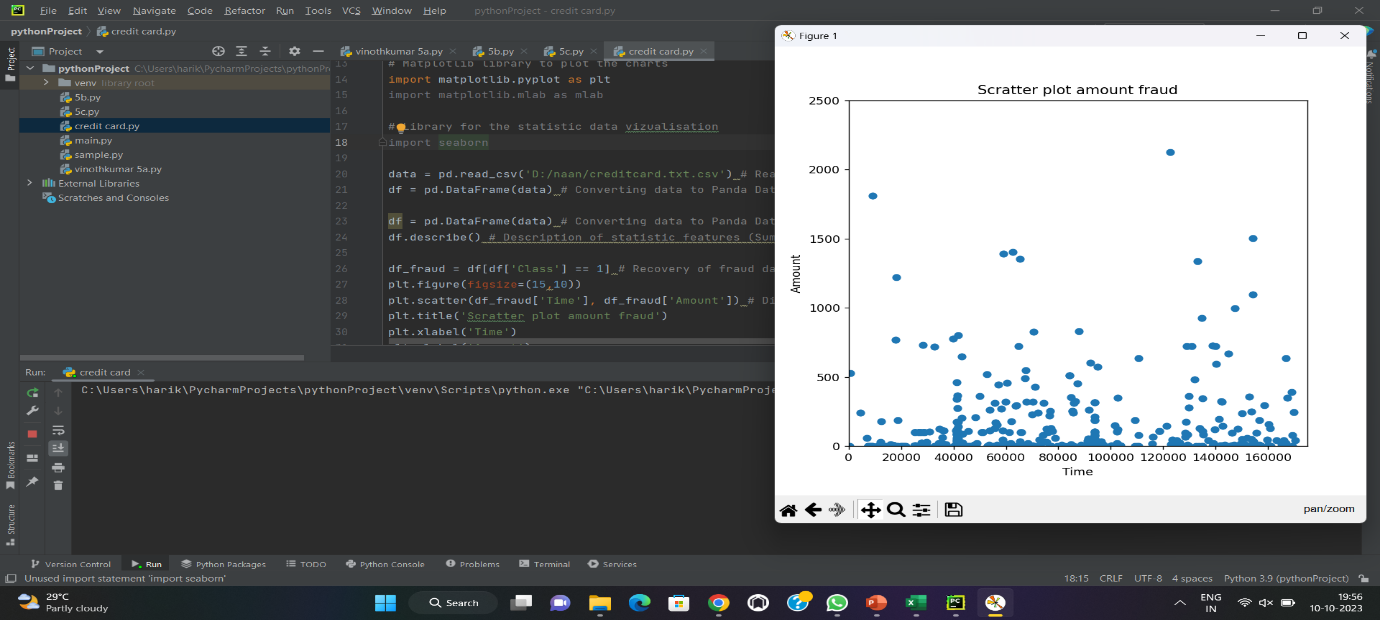
plt.xlabel('Time')

plt.ylabel('Amount')

plt.xlim([0,175000])

plt.ylim([0,2500])

plt.show()

OUTPUT: 

Credit card fraud detection is a critical application in the financial industry, and it typically involves the use of machine learning and data analytics techniques to identify and prevent fraudulent transactions. Below, I'll provide a high-level overview of the documentation you might need to create or refer to when working on credit card fraud detection:

1. **Project Scope and Objectives**:
   * Start with a clear definition of the project's goals and objectives. What are you trying to achieve with credit card fraud detection?
2. **Data Sources**:
   * Document the data sources you will use, such as transaction records, customer information, and any other relevant data.
3. **Data Collection**:
   * Explain how data is collected and stored, including data extraction methods, data cleaning, and data preprocessing.
4. **Data Features**:
   * Describe the features (variables) in your dataset. This might include information like transaction amount, location, time, and customer information.
5. **Data Exploration**:
   * Provide details on how you explored and analyzed the data. This might include summary statistics, data visualization, and initial insights into the data.
6. **Data Preprocessing**:
   * Document the steps taken to clean and preprocess the data, including handling missing values, outliers, and data normalization.
7. **Model Selection**:
   * Explain the machine learning or statistical models chosen for fraud detection, such as logistic regression, decision trees, random forests, neural networks, or ensemble methods.
8. **Model Training**:
   * Detail the process of training your chosen model, including hyperparameter tuning and cross-validation techniques.
9. **Model Evaluation**:
   * Describe the metrics used to evaluate the model's performance, such as accuracy, precision, recall, F1-score, ROC AUC, and confusion matrices.
10. **Feature Importance**:
    * If applicable, document the features that were found to be most important for fraud detection and how they were determined.
11. **Model Deployment**:
    * Explain how the model will be deployed in a real-world environment, such as integration with a payment processing system.
12. **Monitoring and Maintenance**:
    * Document the procedures for monitoring the model's performance and how often it will be updated or retrained.
13. **Regulatory Compliance**:
    * Ensure that your fraud detection system complies with relevant regulations, such as GDPR, PCI DSS, or any industry-specific regulations.
14. **Security**:
    * Describe the security measures in place to protect sensitive customer data and the model itself from potential attacks.
15. **Documentation for End-Users**:
    * If applicable, create user documentation for those responsible for monitoring and using the system, explaining how to interpret model outputs and take appropriate actions in case of fraud.
16. **Data Privacy and Ethics**:
    * Address the ethical considerations involved in using customer data and how privacy is maintained throughout the process.
17. **Disaster Recovery Plan**:
    * Create a plan for handling system failures or breaches, including data backup and recovery procedures.
18. **Performance Benchmarks**:
    * Set clear benchmarks for the model's performance and outline what actions will be taken if those benchmarks are not met.
19. **Training and Awareness**:
    * Provide training and awareness programs for employees involved in the process to ensure they understand the importance of fraud detection and their role in it.
20. **Documentation Updates**:
    * Ensure that the documentation is regularly updated to reflect changes in the system or new regulations.

***Basic Exploratory Analysis:***  

Basic exploratory data analysis (EDA) is an essential initial step in data analysis that involves summarizing, visualizing, and understanding the main characteristics of your dataset.

exploratory data analysis (EDA) is used by data scientists to analyze and investigate data sets and summarize their main characteristics, often employing data visualization methods. It helps determine how best to manipulate data sources to get the answers you need, making it easier for data scientists to discover patterns, spot anomalies, test a hypothesis, or check assumptions.

***Program:***

import pandas as pd

import matplotlib.pyplot as plt

df=pd.read\_csv("country\_vaccinations.csv",encoding="unicode\_escape")

print(df.head())

print(df.describe())

print(df['people\_vaccinated'].value\_counts())

df.hist()

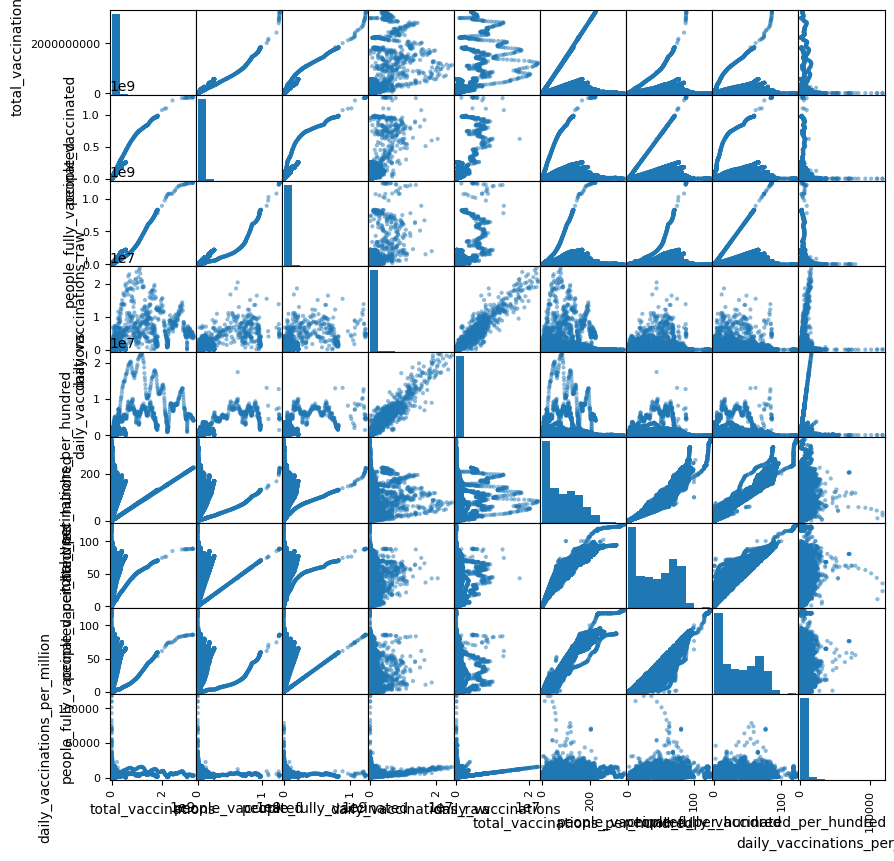
plt.show()

from pandas.plotting import scatter\_matrix

scatter\_matrix(df, figsize=(10, 10))

plt.show()

***Output:***



**Statistical Analysis:**

      Statistical analysis is a systematic approach to understanding data through the application of mathematical and statistical techniques. It plays a crucial role in making sense of complex information, identifying patterns, and drawing meaningful insights. The process typically begins with data collection, followed by data cleaning and preprocessing to ensure data quality. Descriptive statistics are employed to provide an initial summary of the dataset, revealing central tendencies and variability. Inferential statistics, on the other hand, are used to make predictions and test hypotheses about the population from which the data was collected. This branch of analysis encompasses a wide array of methods, including hypothesis testing, regression analysis, and analysis of variance, among others. Statistical analysis is a cornerstone in fields ranging from science and business to healthcare and social sciences, aiding in decision-making, problem-solving, and evidence-based reasoning.

***Program:***

 import pandas as pd

import numpy as np

from scipy import stats

df = pd.read\_csv("merged\_dataset.csv")

df=df.tail(10)

summary = df.describe()

group1\_data = df['total\_vaccinations\_x']

group2\_data = df['people\_vaccinated']

t\_statistic, p\_value = stats.ttest\_ind(group1\_data, group2\_data)

correlation\_coefficient = df['people\_fully\_vaccinated'].corr(df['daily\_vaccinations\_raw'])

print("Descriptive Statistics:")

print(summary)

print("\nT-Test Results:")

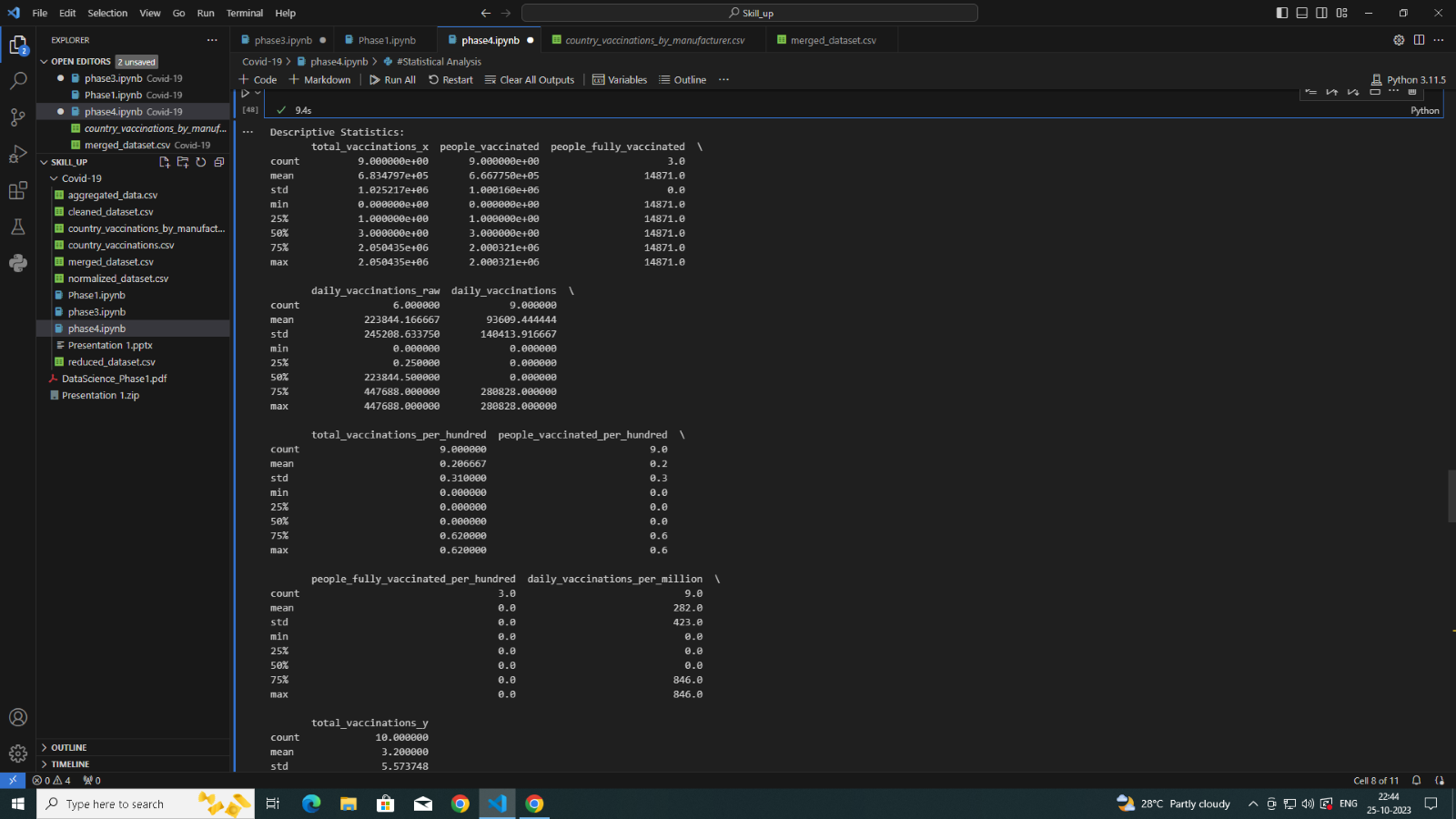
print(f"T-Statistic: {t\_statistic}")

print(f"P-Value: {p\_value}")

print("\nPearson Correlation Coefficient:")

print(correlation\_coefficient)

***Output:***



***Data Visualization:***

Data visualization is a powerful method of representing complex data in a visual and understandable format. It involves the creation of charts, graphs, and diagrams to illustrate patterns, trends, and relationships within the data. By presenting information visually, data visualization makes it easier for individuals to comprehend and interpret large datasets. Data visualizations can take many forms, from simple bar charts and pie graphs to intricate heat maps and interactive dashboards. They are particularly useful for identifying outliers, correlations, and data distributions, making data more accessible and actionable. In an era of data abundance, data visualization has become an indispensable tool for turning data into knowledge and communicating findings effectively to both experts and non-experts.

***Program:***

import pandas as pd

import matplotlib.pyplot as plt

df = pd.read\_csv("country\_vaccinations.csv")

df=df.tail(50)

categories = df["date"]

values = df["total\_vaccinations"]

plt.figure(figsize=(10, 6))

plt.bar(categories, values)

plt.xlabel("Categories")

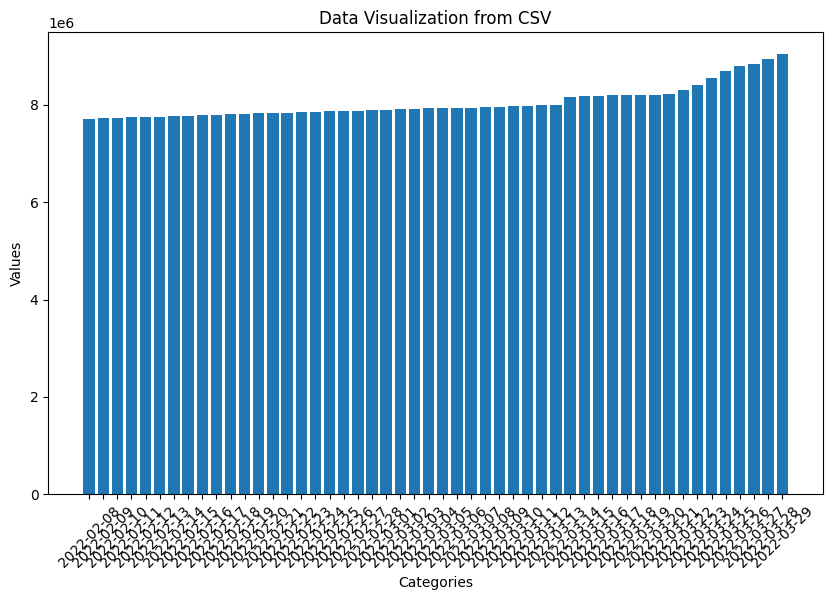
plt.ylabel("Values")

plt.title("Data Visualization from CSV")

plt.xticks(rotation=45)

plt.show()

***Output:***



***Program2:***

import matplotlib.pyplot as plt

from mpl\_toolkits.mplot3d import Axes3D

from sklearn.datasets import load\_iris

df = pd.read\_csv("country\_vaccinations\_by\_manufacturer.csv")

fig = plt.figure()

ax = fig.add\_subplot(111, projection='3d')

scatter = ax.scatter(X[:, 0], X[:, 1], X[:, 2], c=y, cmap=plt.cm.Set1)

ax.set\_xlabel('total\_vaccinations')

ax.set\_ylabel('people\_vaccinated')

ax.set\_zlabel('people\_vaccinated\_per\_100')

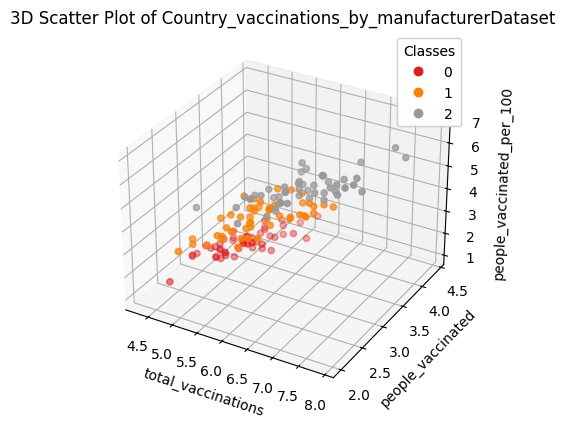
legend = ax.legend(\*scatter.legend\_elements(), title="Classes")

ax.add\_artist(legend)

ax.set\_title('3D Scatter Plot of Country\_vaccinations\_by\_manufacturerDataset')

plt.show()

***Output:***



***Program 3:***

import numpy as np

import pandas as pd

import seaborn as sns

import matplotlib as plt

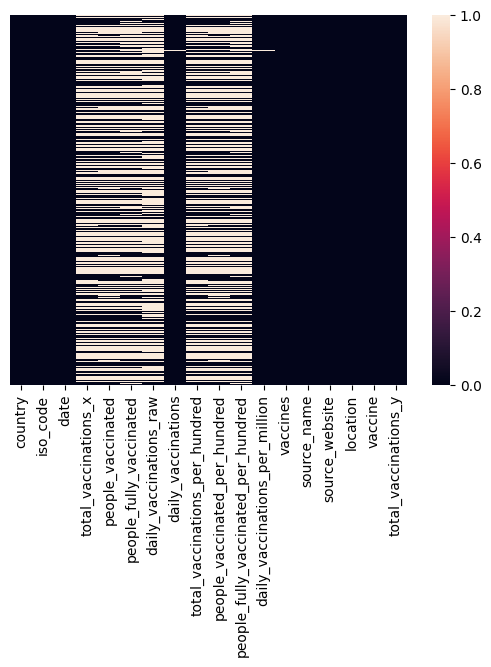
df=pd.read\_csv("merged\_dataset.csv",encoding="unicode\_escape")

missing\_values=["N/a","na",np.nan]

df=pd.read\_csv("merged\_dataset.csv",na\_values=missing\_values,encoding="unicode\_escape")

print(sns.heatmap(df.isnull(),yticklabels=False))

***Output:***



CONCLUSION:

In conclusion, credit card fraud detection is a critical aspect of financial security in today's digital age. Fraudulent activities can result in significant financial losses, damage to a financial institution's reputation, and harm to cardholders. To combat this threat effectively, credit card fraud detection systems rely on data analysis, machine learning, and various preprocessing steps to prepare the data for modeling. Here are the key takeaways:

\*\*Importance of Credit Card Fraud Detection\*\*: Credit card fraud poses a substantial risk to financial institutions and cardholders, necessitating advanced fraud detection methods.

\*\*Data Preprocessing\*\*: Preprocessing the dataset is a vital step, involving data cleaning, handling class imbalance, scaling, feature engineering, encoding, dimensionality reduction, and splitting the data into training and testing sets.