Credit Card fraud detection

Importing libraries

In [3]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns



from sklearn.model_selection import train_test_split, KFold, StratifiedKFold, cross
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix
from sklearn.neighbors import KNeighborsClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.preprocessing import StandardScaler
from sklearn.feature_selection import SelectFromModel
from imblearn.pipeline import Pipeline
from imblearn.over_sampling import SMOTE
from sklearn.feature_selection import SelectKBest, mutual_info_classif
from category_encoders import WOEEncoder
from sklearn.feature_selection import RFE
from sklearn.linear_model import LogisticRegression
from imblearn.pipeline import Pipeline as ImbPipeline
```

Reading dataset

In [5]:

```
df=pd.read_csv("fraud test.csv")
df.head()
```

Out[5]:

	Unnamed: 0	trans_date_trans_time	cc_num	merchant	category	amt
0	0	21/06/2020 12:14	2.291160e+15	fraud_Kirlin and Sons	personal_care	2.86
1	1	21/06/2020 12:14	3.573030e+15	fraud_Sporer-Keebler	personal_care	29.84
2	2	21/06/2020 12:14	3.598220e+15	fraud_Swaniawski, Nitzsche and Welch	health_fitness	41.28
3	3	21/06/2020 12:15	3.591920e+15	fraud_Haley Group	misc_pos	60.05
4	4	21/06/2020 12:15	3.526830e+15	fraud_Johnston-Casper	travel	3.19

5 rows × 23 columns

In [6]:

```
print(df.columns)
```

```
Index(['Unnamed: 0', 'trans_date_trans_time', 'cc_num', 'merchant', 'category',
       'amt', 'first', 'last', 'gender', 'street', 'city', 'state', 'zip',
       'lat', 'long', 'city_pop', 'job', 'dob', 'trans_num', 'unix_time',
       'merch_lat', 'merch_long', 'is_fraud'],
      dtype='object')
```

Checking null values in data

In [8]:

```
df.isnull().sum()
```

Out[8]:

```
Unnamed: 0               0
trans_date_trans_time    0
cc_num                   0
merchant                 0
category                 0
amt                      0
first                    0
last                     0
gender                   0
street                   0
city                     0
state                    0
zip                      0
lat                      0
long                     0
city_pop                 0
job                      0
dob                      0
trans_num                0
unix_time                0
merch_lat                0
merch_long               0
is_fraud                 0
dtype: int64
```

In [9]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 555719 entries, 0 to 555718
Data columns (total 23 columns):
 #   Column                 Non-Null Count   Dtype  
---  ------                 --------------   -----  
 0   Unnamed: 0             555719 non-null  int64  
 1   trans_date_trans_time  555719 non-null  object 
 2   cc_num                 555719 non-null  float64
 3   merchant               555719 non-null  object 
 4   category               555719 non-null  object 
 5   amt                    555719 non-null  float64
 6   first                  555719 non-null  object 
 7   last                   555719 non-null  object 
 8   gender                 555719 non-null  object 
 9   street                 555719 non-null  object 
 10  city                   555719 non-null  object 
 11  state                  555719 non-null  object 
 12  zip                    555719 non-null  int64  
 13  lat                    555719 non-null  float64
 14  long                   555719 non-null  float64
 15  city_pop               555719 non-null  int64  
 16  job                    555719 non-null  object 
 17  dob                    555719 non-null  object 
 18  trans_num              555719 non-null  object 
 19  unix_time              555719 non-null  int64  
 20  merch_lat              555719 non-null  float64
 21  merch_long             555719 non-null  float64
 22  is_fraud               555719 non-null  int64  
dtypes: float64(6), int64(5), object(12)
memory usage: 97.5+ MB
```

In [10]:
```
df.describe()
```

Out[10]:

	Unnamed: 0	cc_num	amt	zip	lat	
count	555719.000000	5.557190e+05	555719.000000	555719.000000	555719.000000	555719.0
mean	277859.000000	4.178387e+17	69.392810	48842.628015	38.543253	-90.2
std	160422.401459	1.309837e+18	156.745941	26855.283328	5.061336	13.7
min	0.000000	6.041621e+10	1.000000	1257.000000	20.027100	-165.6
25%	138929.500000	1.800430e+14	9.630000	26292.000000	34.668900	-96.7
50%	277859.000000	3.521420e+15	47.290000	48174.000000	39.371600	-87.4
75%	416788.500000	4.635330e+15	83.010000	72011.000000	41.894800	-80.1
max	555718.000000	4.992350e+18	22768.110000	99921.000000	65.689900	-67.9

Distribution of fraudulent transaction

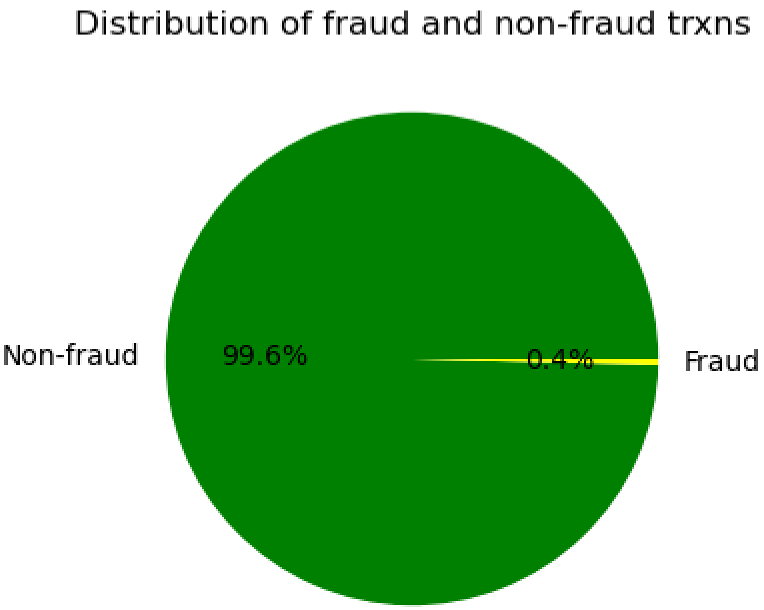
In [12]:
```
fraud = df[df['is_fraud'] == 1]
non_fraud = df[df['is_fraud'] == 0]
outlierFraction = len(fraud)/float(len(non_fraud))
```

```
print(outlierFraction)
print('Fraud Cases: {}'.format(len(df[df['is_fraud'] == 1])))
print('Non fraud Transactions: {}'.format(len(df[df['is_fraud'] == 0])))
```

```
0.003874820710510248
Fraud Cases: 2145
Non fraud Transactions: 553574
```

In [13]:

```
fraud_counts = df['is_fraud'].value_counts()

# Plot the pie chart
plt.figure(figsize=(5, 4))
plt.pie(fraud_counts, labels=['Non-fraud', 'Fraud'], autopct='%1.1f%%', colors=['gr
plt.title('Distribution of fraud and non-fraud trxns')
plt.show()
```

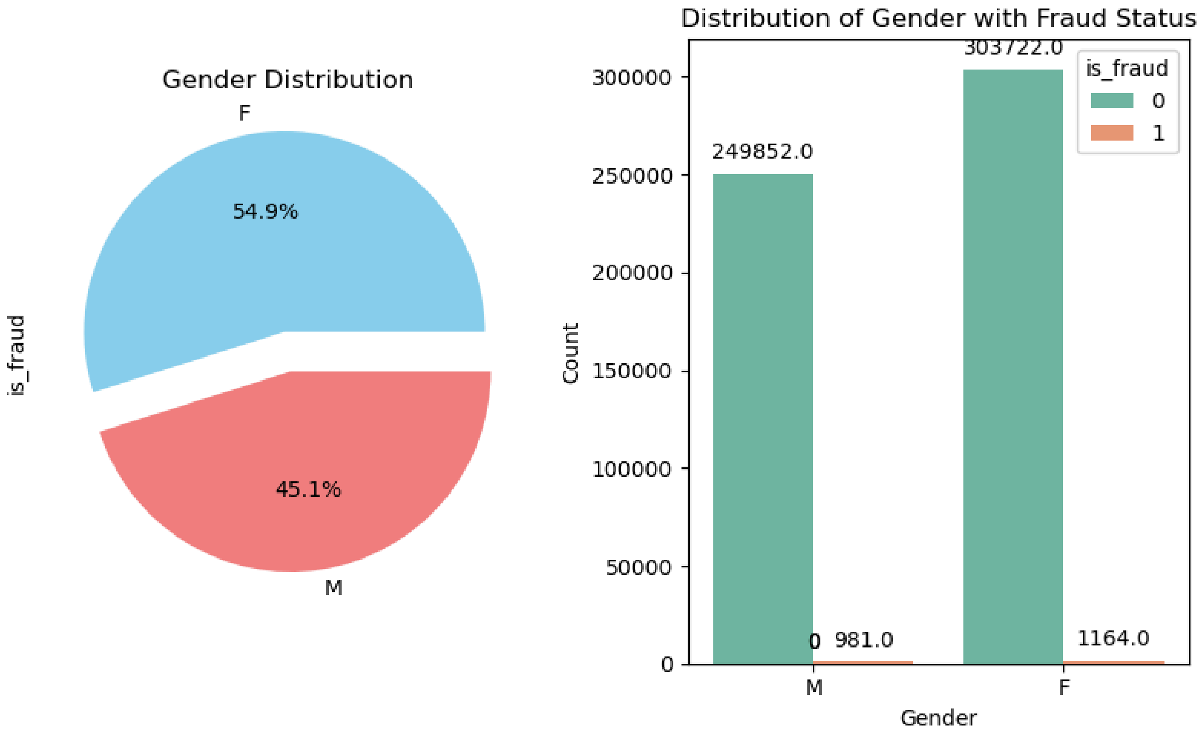


Distribution of gender verses fraudulent transactions

In [15]:

```
fig, axs = plt.subplots(ncols=2, nrows=1, figsize=(8, 5))

# Gender distribution Pie chart
explode = [0.1, 0.1]
df.groupby('gender')['is_fraud'].count().plot.pie(explode=explode, autopct="%1.1f%%
axs[0].set_title("Gender Distribution")

# Fraud Status by gender Count Plot
ax = sns.countplot(x="gender", hue="is_fraud", data=df, ax=axs[1], palette='Set2')

# Add values on top of each bar
for p in ax.patches:
    ax.annotate(f'{p.get_height()}', (p.get_x() + p.get_width() / 2., p.get_height(
                ha='center', va='center', xytext=(0, 10), textcoords='offset points

# Set labels and title
axs[1].set_title("Distribution of Gender with Fraud Status")
```

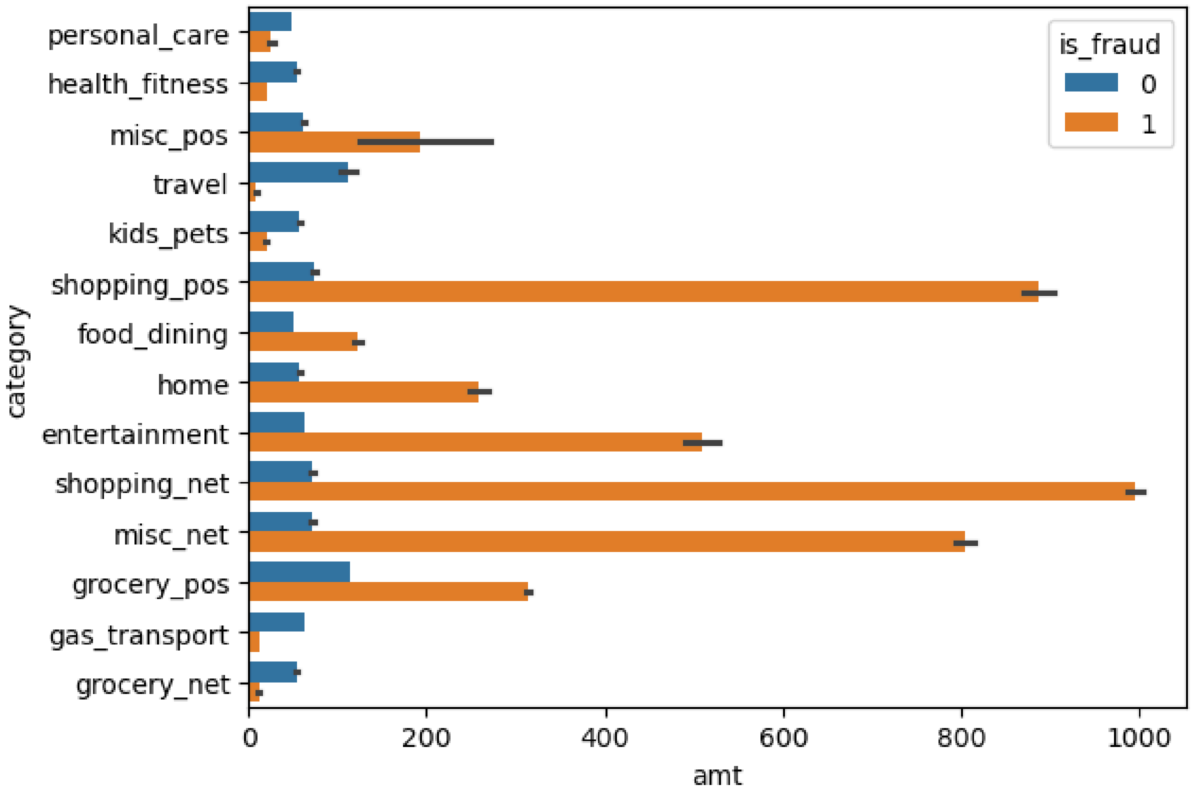
```
axs[1].set_xlabel("Gender")
axs[1].set_ylabel("Count")

# Show the plot
plt.tight_layout()
plt.show()
```



Category verses fraudulent and non-fraudulent activities

In [17]:

```
sns.barplot(x = "amt", y = "category", data = df, hue = "is_fraud")
plt.show()
```



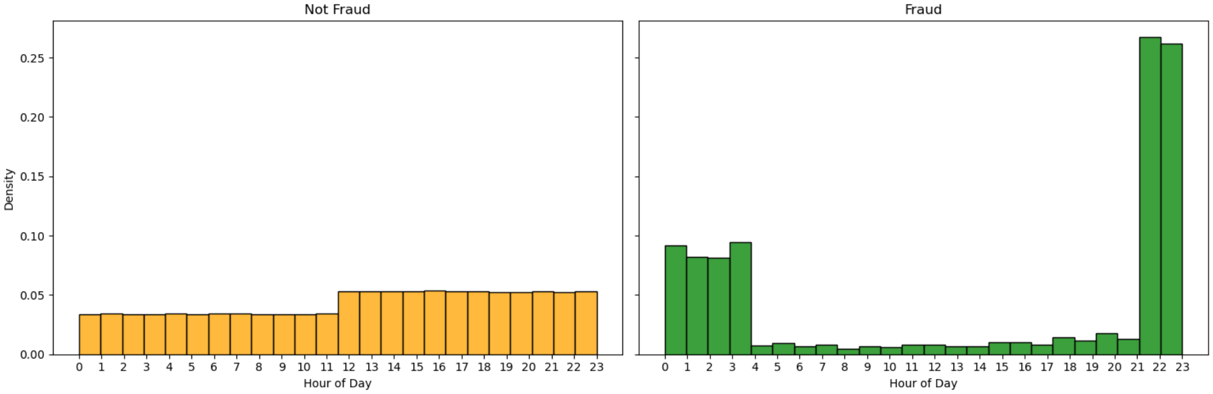
Distribution of fraudulent activities by hour

In [19]:
```
# Convert 'trans_date_trans_time' column to datetime format
df['trans_date_trans_time'] = pd.to_datetime(df['trans_date_trans_time'])
```

```
C:\Users\Ifean\AppData\Local\Temp\ipykernel_20284\243000987.py:2: UserWarning: Parsi
ng dates in %d/%m/%Y %H:%M format when dayfirst=False (the default) was specified. P
ass `dayfirst=True` or specify a format to silence this warning.
  df['trans_date_trans_time'] = pd.to_datetime(df['trans_date_trans_time'])
```

In [20]:
```
# Convert 'trans_date_trans_time' column to datetime and extract the hour
df['hour'] = pd.to_datetime(df['trans_date_trans_time']).dt.hour

# Create a figure with two subplots
fig, axes = plt.subplots(1, 2, figsize=(15, 5), sharey=True)

# Plot histogram for non-fraudulent transactions
sns.histplot(x='hour', data=df[df["is_fraud"] == 0],
             stat="density", bins=24, ax=axes[0], color="orange")
axes[0].set_title("Not Fraud")
axes[0].set_xlabel("Hour of Day")
axes[0].set_ylabel("Density")
axes[0].set_xticks(range(24))

# Plot histogram for fraudulent transactions
sns.histplot(x='hour', data=df[df["is_fraud"] == 1],
             stat="density", bins=24, ax=axes[1], color="green")
axes[1].set_title("Fraud")
axes[1].set_xlabel("Hour of Day")
axes[1].set_ylabel("Density")
```

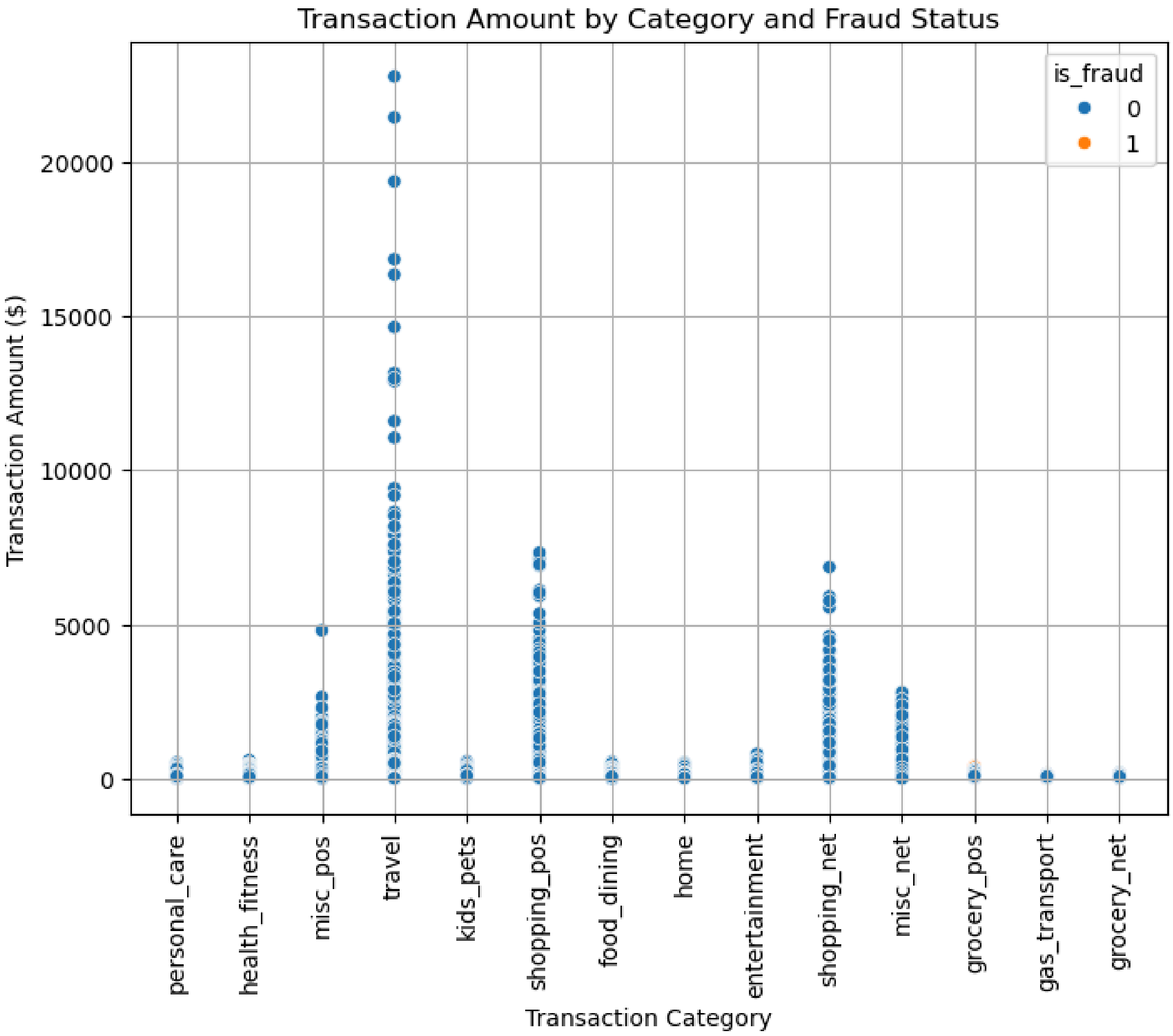
```
axes[1].set_xticks(range(24))

# Adjust layout and display the plot
plt.tight_layout()
plt.show()
```



Transaction Amount by category and Fraud Status

In [22]:

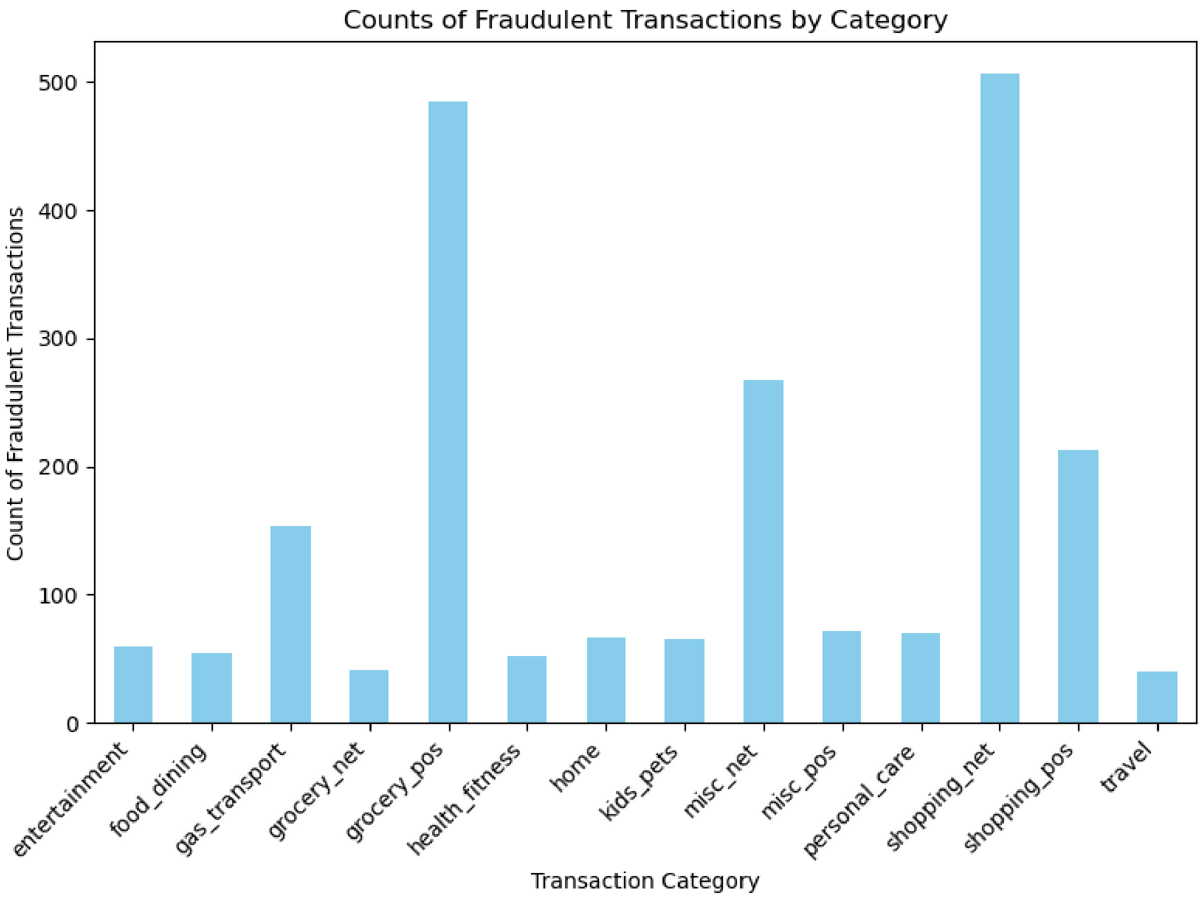
```
# Txn amt by cat and fraud
plt.figure(figsize=(8,6))
sns.scatterplot(x = "category", y = "amt", data = df, hue = "is_fraud")
plt.grid(True)
plt.xticks(rotation=90)
plt.title('Transaction Amount by Category and Fraud Status')
plt.xlabel('Transaction Category')
plt.ylabel('Transaction Amount ($)')

plt.show()
```



Counts of Fraudulent Transactions by Category

In [24]:

```
fraud_counts = df.groupby('category')['is_fraud'].sum()

plt.figure(figsize=(8, 6))
fraud_counts.plot(kind='bar', color='skyblue')

plt.xlabel('Transaction Category')
plt.ylabel('Count of Fraudulent Transactions')
plt.title('Counts of Fraudulent Transactions by Category')
plt.xticks(rotation=45, ha='right')
plt.tight_layout()
plt.show()
```

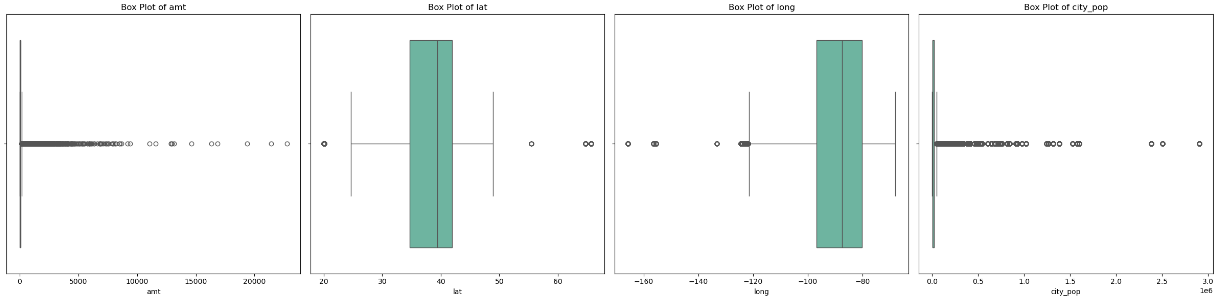


Checking outliers

In [26]:

```python
columns = ['amt', 'lat', 'long', 'city_pop']
fig, axes = plt.subplots(nrows=1, ncols=len(columns), figsize=(6*len(columns), 6))
for i, col in enumerate(columns):
    sns.boxplot(x=df[col], ax=axes[i], orient='h', palette='Set2')
    axes[i].set_title(f'Box Plot of {col}')
    axes[i].set_xlabel(col)

plt.tight_layout()
plt.show()
```

```
C:\Users\Ifean\AppData\Local\Temp\ipykernel_20284\409219191.py:4: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.1
4.0. Assign the `y` variable to `hue` and set `legend=False` for the same effect.

  sns.boxplot(x=df[col], ax=axes[i], orient='h', palette='Set2')
C:\Users\Ifean\AppData\Local\Temp\ipykernel_20284\409219191.py:4: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.1
4.0. Assign the `y` variable to `hue` and set `legend=False` for the same effect.

  sns.boxplot(x=df[col], ax=axes[i], orient='h', palette='Set2')
C:\Users\Ifean\AppData\Local\Temp\ipykernel_20284\409219191.py:4: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.1
4.0. Assign the `y` variable to `hue` and set `legend=False` for the same effect.

  sns.boxplot(x=df[col], ax=axes[i], orient='h', palette='Set2')
C:\Users\Ifean\AppData\Local\Temp\ipykernel_20284\409219191.py:4: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.1
4.0. Assign the `y` variable to `hue` and set `legend=False` for the same effect.

  sns.boxplot(x=df[col], ax=axes[i], orient='h', palette='Set2')
```



Feature Engineering

In [28]:

```
# 1. Age at Transactions
df['dob'] = pd.to_datetime(df['dob'])
df['trans_date_trans_time'] = pd.to_datetime(df['trans_date_trans_time'])
df['age_at_txns'] = (df['trans_date_trans_time'] - df['dob']).dt.days // 365

# 2. Time Since Last Transactions
df.sort_values(['cc_num', 'trans_date_trans_time'], inplace=True)
df['time_since_last_txn'] = df.groupby('cc_num')['trans_date_trans_time'].diff().dt

# 3. Transaction Amount Relative to Average
df['avg_txn_amount'] = df.groupby('cc_num')['amt'].transform('mean')
df['txn_amount_relative_to_avg'] = df['amt'] / df['avg_txn_amount']

# 4. Cumulative Transactions Amount
df['cumulative_txn_amount'] = df.groupby('cc_num')['amt'].cumsum()


print(df[['age_at_txns', 'time_since_last_txn', 'txn_amount_relative_to_avg', 'cumu
C:\Users\Ifean\AppData\Local\Temp\ipykernel_20284\3040901476.py:2: UserWarning: Pars
ing dates in %d/%m/%Y format when dayfirst=False (the default) was specified. Pass `
dayfirst=True` or specify a format to silence this warning.
  df['dob'] = pd.to_datetime(df['dob'])
        age_at_txns  time_since_last_txn  txn_amount_relative_to_avg  \
157              34                  NaN                    1.874601   
741              34                  0.0                    1.180761   
3047             34                  0.0                    0.981211   
4351             34                  0.0                    1.319409   
7695             34                  0.0                    2.225882   
...             ...                  ...                         ...   
552584           65                  0.0                    1.046956   
552892           65                  0.0                    0.072532   
553559           65                  0.0                    1.519678   
553560           65                  0.0                    2.367252   
553883           65                  0.0                    0.325442   

        cumulative_txn_amount  
157                    124.66  
741                    203.18  
3047                   268.43  
4351                   356.17  
7695                   504.19  
...                       ...  
552584               54223.53  
552892               54228.11  
553559               54324.07  
553560               54473.55  
553883               54494.10  

[555719 rows x 4 columns]
```

Dropping some columns and data transformation

```
In [30]: # Drop the 'Unnamed' and cc_num column
         df.drop(columns=['Unnamed: 0','cc_num'], inplace=True)
```

```
In [31]: df=df.drop(columns=['first', 'unix_time', 'dob', 'zip', 'city','street', 'state', '
```

```
In [32]: df['merchant'] = df['merchant'].apply(lambda x : x.replace('fraud_',''))
         df['gender'] = df['gender'].map({'F': 0, 'M': 1})
```

Converting categorical columns into numeric

```
In [34]: encode_column = ['job', 'merchant', 'category', 'lat']
         woe_encoder = WOEEncoder()

         df_trasform = woe_encoder.fit_transform(df[encode_column], df['is_fraud'])

         df[encode_column] = df_trasform
```

```
In [35]: df_trasform.head()
```

Out[35]:

	job	merchant	category	lat
157	-1.264408	-0.291217	-1.092515	43.0048
741	-1.264408	0.369140	-0.606015	43.0048
3047	-1.264408	0.735637	-0.341222	43.0048
4351	-1.264408	-1.280704	-1.049973	43.0048
7695	-1.264408	-0.407388	-0.762903	43.0048

In [36]:

```
df.head()
```

Out[36]:

	merchant	category	amt	gender	lat	long	city_pop	job	merch
157	-0.291217	-1.092515	124.66	0	43.0048	-108.8964	1645	-1.264408	42.945
741	0.369140	-0.606015	78.52	0	43.0048	-108.8964	1645	-1.264408	42.193
3047	0.735637	-0.341222	65.25	0	43.0048	-108.8964	1645	-1.264408	43.932
4351	-1.280704	-1.049973	87.74	0	43.0048	-108.8964	1645	-1.264408	43.546
7695	-0.407388	-0.762903	148.02	0	43.0048	-108.8964	1645	-1.264408	42.876

In [37]:

```
# Drop irrelevant columns
df.drop(columns=['hour','age_at_txns', 'long', 'city_pop', 'merchant','lat','time_s
```

In [38]:

```
df.describe()
```

Out[38]:

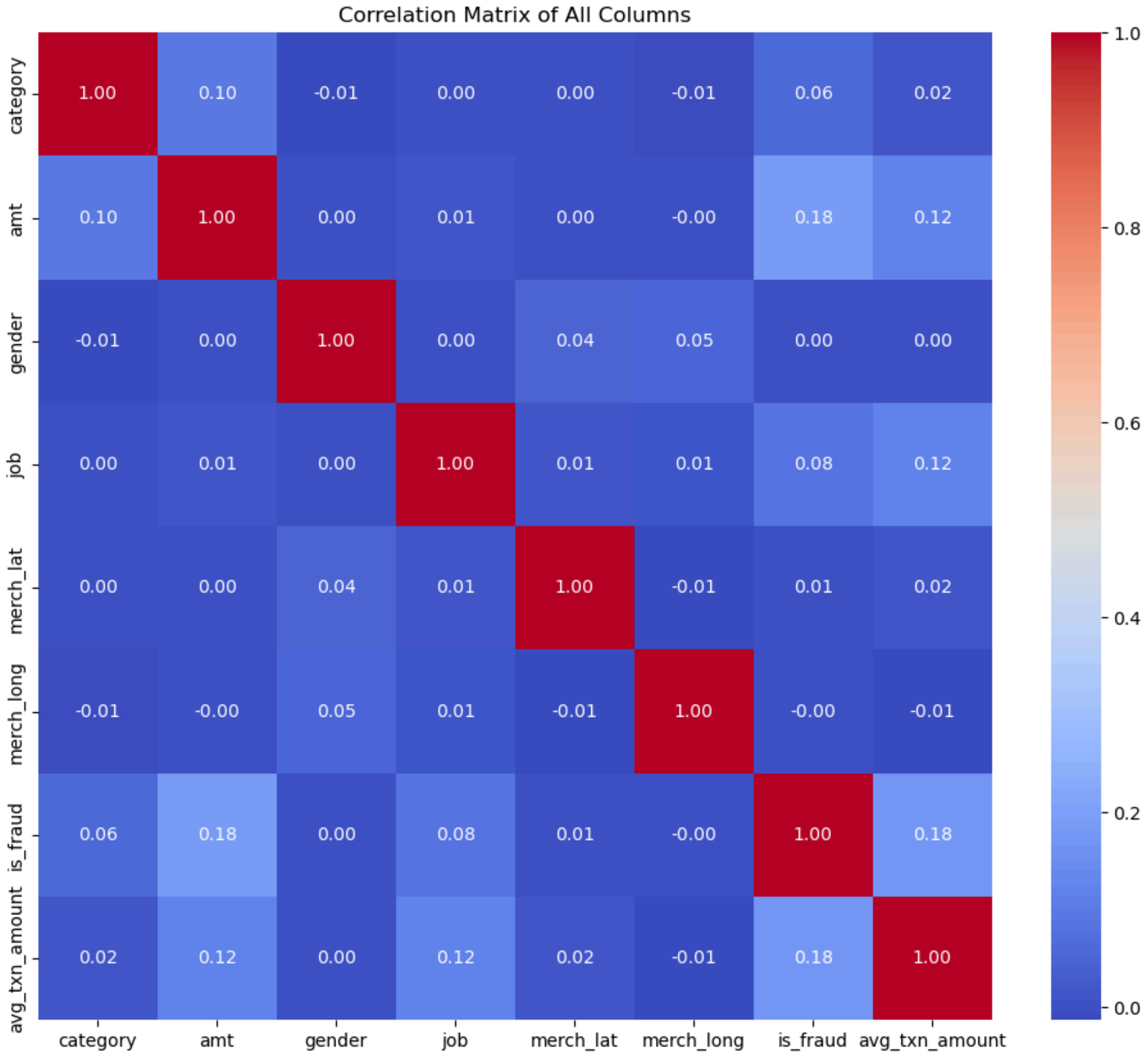
	category	amt	gender	job	merch_lat	merc
count	555719.000000	555719.000000	555719.000000	555719.000000	555719.000000	555719.
mean	-0.337399	69.392810	0.451367	-0.566606	38.542798	-90.
std	0.781185	156.745941	0.497630	1.285966	5.095829	13.
min	-1.092515	1.000000	0.000000	-2.733942	19.027422	-166.
25%	-1.017154	9.630000	0.000000	-1.711002	34.755302	-96.
50%	-0.606015	47.290000	0.000000	-0.864405	39.376593	-87.
75%	0.106981	83.010000	1.000000	0.588900	41.954163	-80.
max	1.152850	22768.110000	1.000000	8.037234	66.679297	-66.

In [39]:

```
scaler = StandardScaler()

# Fit and transform the data
df_scaled = pd.DataFrame(scaler.fit_transform(df), columns=df.columns)

# Check the scaled data
print(df_scaled.head())
```

```
   category       amt    gender      job  merch_lat  merch_long  is_fraud  \
0 -0.966630  0.352591 -0.907034 -0.54263   0.863987   -1.332516 -0.062248
1 -0.343857  0.058229 -0.907034 -0.54263   0.716338   -1.343523 -0.062248
2 -0.004893 -0.026430 -0.907034 -0.54263   1.057714   -1.417631 -0.062248
3 -0.912171  0.117051 -0.907034 -0.54263   0.981836   -1.382180 -0.062248
4 -0.544691  0.501622 -0.907034 -0.54263   0.850449   -1.390938 -0.062248

   avg_txn_amount
0       -0.156213
1       -0.156213
2       -0.156213
3       -0.156213
4       -0.156213
```

In [40]:
```
# Calculate the correlation matrix for the entire DataFrame
corr_matrix = df.corr()

# Plot the correlation matrix
plt.figure(figsize=(12, 10))
sns.heatmap(corr_matrix, annot=True, fmt=".2f", cmap='coolwarm', cbar=True)
plt.title("Correlation Matrix of All Columns")
plt.show()
```



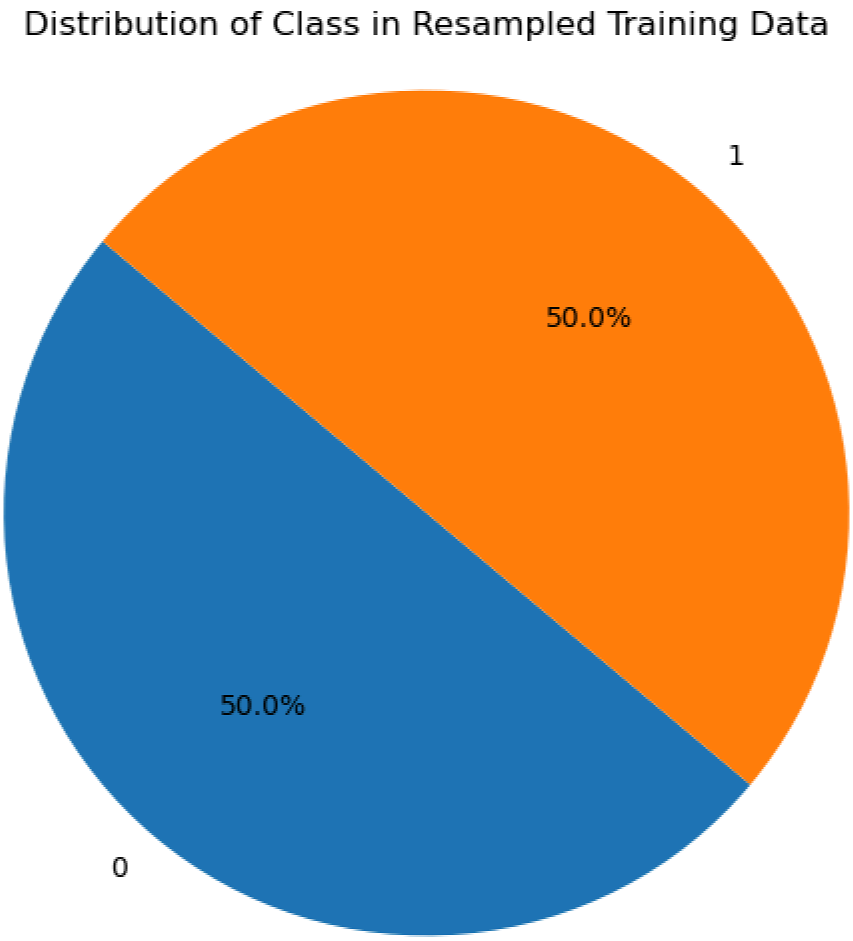
In [41]:
```
X = df.drop(labels='is_fraud', axis=1)
y = df.loc[:,'is_fraud']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_sta
```

Balancing dataset using SMOTE

In [43]:
```
smote = SMOTE(random_state=1)
X_train_resampled, y_train_resampled = smote.fit_resample(X_train, y_train)
```

```
C:\Users\Ifean\anaconda3\Lib\site-packages\sklearn\base.py:474: FutureWarning: `Base
Estimator._validate_data` is deprecated in 1.6 and will be removed in 1.7. Use `skle
arn.utils.validation.validate_data` instead. This function becomes public and is par
t of the scikit-learn developer API.
  warnings.warn(
```

In [44]:
```
class_counts = y_train_resampled.value_counts()

# Create a pie chart
plt.figure(figsize=(6, 6))
plt.pie(class_counts, labels=class_counts.index, autopct='%1.1f%%', startangle=140)
plt.title('Distribution of Class in Resampled Training Data')
plt.axis('equal')
plt.show()
```



Imbalanced classification modelling with synthetic oversampling and stratified K-fold cross-validation

In [46]:

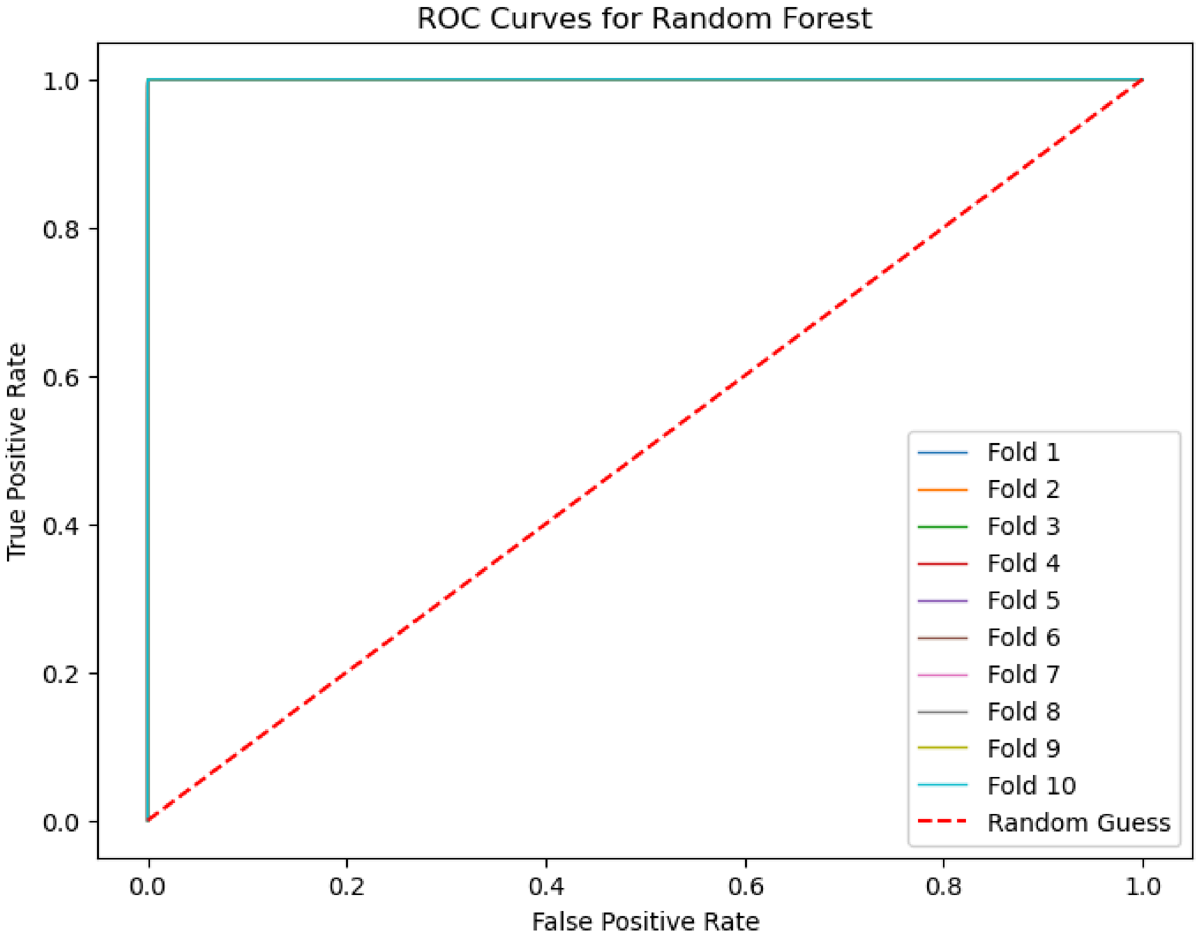
```
skf = StratifiedKFold(n_splits=10, shuffle=True, random_state=42)
```

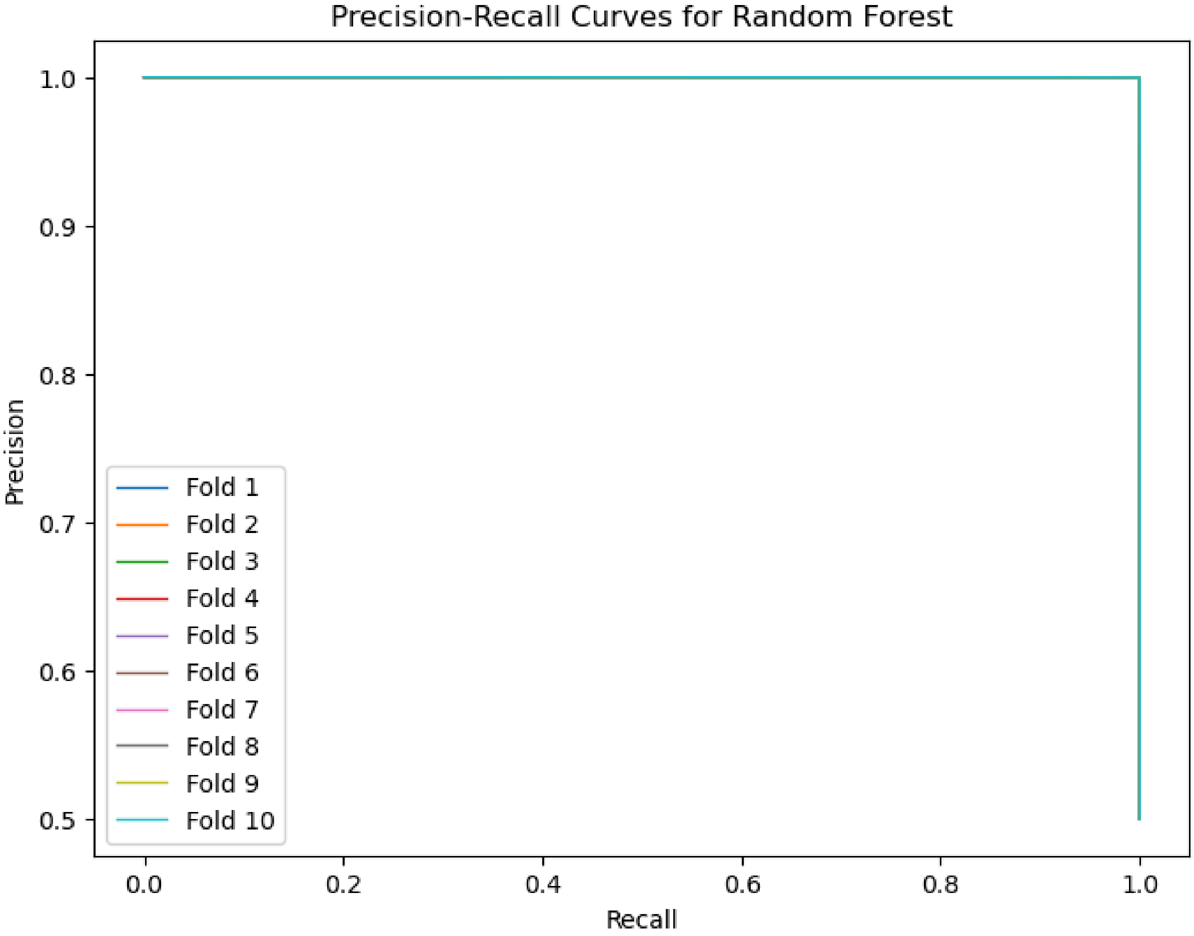
In [47]:

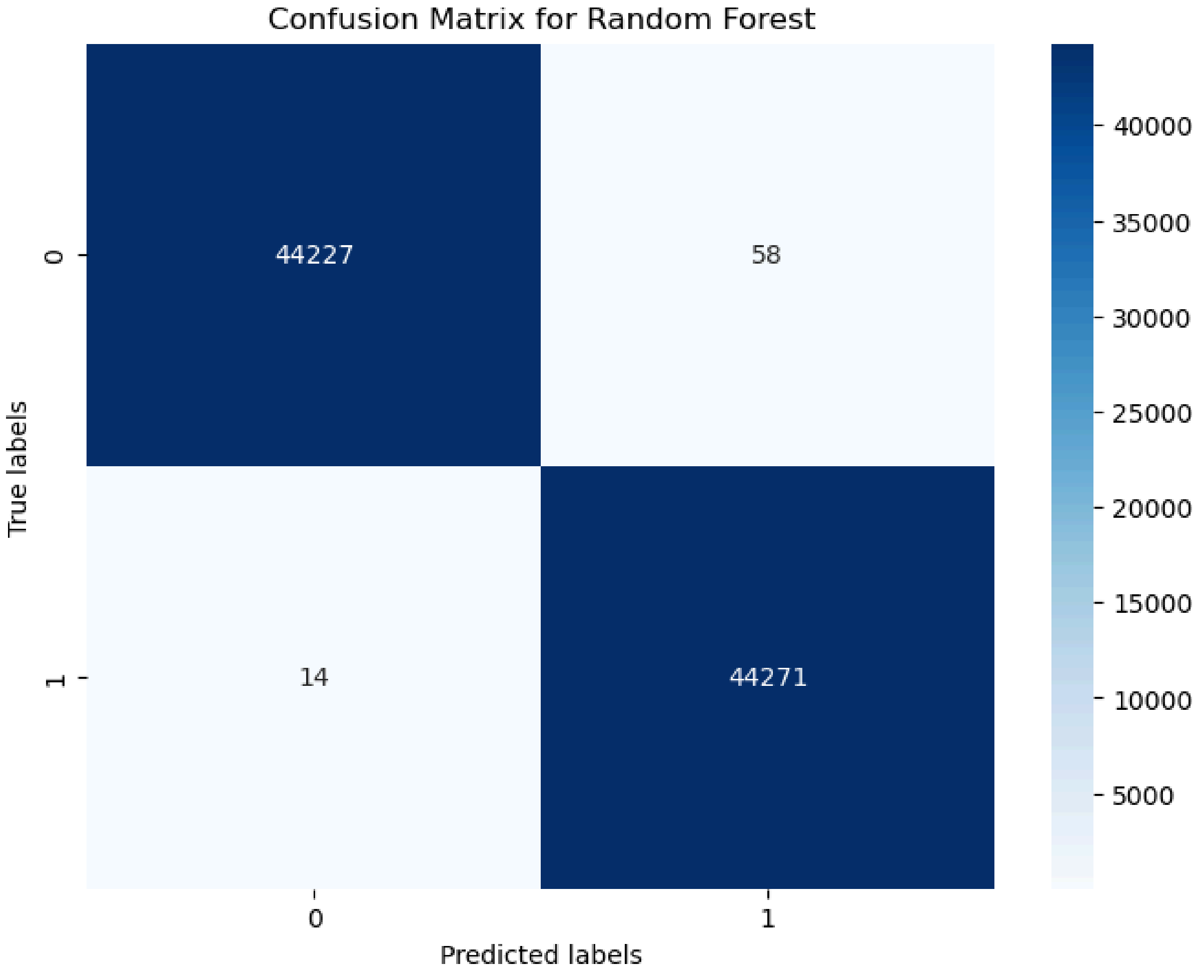
```
# Define the classifier
classifier = RandomForestClassifier(n_estimators=100, random_state=0, n_jobs=-1)

# List to store metrics
roc_auc_scores, fprs, tprs, precisions, recalls, f1_scores = [], [], [], [], [], []
confusion_matrices, accuracies = [], []

# Stratified cross-validation
skf = StratifiedKFold(n_splits=10, shuffle=True, random_state=42)

for train_index, test_index in skf.split(X_train_resampled, y_train_resampled):
    X_train_fold, X_test_fold = X_train_resampled.iloc[train_index], X_train_resamp
    y_train_fold, y_test_fold = y_train_resampled.iloc[train_index], y_train_resamp

    # Fit and predict
    classifier.fit(X_train_fold, y_train_fold)
    y_scores = classifier.predict_proba(X_test_fold)[:, 1]
    y_pred = (y_scores > 0.5)

    # Calculate metrics
    fpr, tpr, _ = roc_curve(y_test_fold, y_scores)
    fprs.append(fpr)
    tprs.append(tpr)
    roc_auc_scores.append(auc(fpr, tpr))
    precision, recall, _ = precision_recall_curve(y_test_fold, y_scores)
    precisions.append(precision)
    recalls.append(recall)
    conf_mat = confusion_matrix(y_test_fold, y_pred)
    confusion_matrices.append(conf_mat)
    accuracies.append(accuracy_score(y_test_fold, y_pred))
    f1_scores.append(f1_score(y_test_fold, y_pred))

# Results
print(f'Average Accuracy for Random Forest: {np.mean(accuracies):.2f}')
print(f'Average F1 Score for Random Forest: {np.mean(f1_scores):.2f}')
print(f'Average ROC AUC Score for Random Forest: {np.mean(roc_auc_scores):.2f}')
print(f'Average Precision for Random Forest: {np.mean([np.mean(precision) for preci
print(f'Average Recall for Random Forest: {np.mean([np.mean(recall) for recall in r

def plot_roc_curves(fprs, tprs, model_name):
    plt.figure(figsize=(8, 6))
    for i in range(len(fprs)):
        plt.plot(fprs[i], tprs[i], lw=1, label=f'Fold {i+1}')
    plt.plot([0, 1], [0, 1], linestyle='--', color='r', label='Random Guess')
    plt.xlabel('False Positive Rate')
    plt.ylabel('True Positive Rate')
    plt.title(f'ROC Curves for {model_name}')
    plt.legend(loc='lower right')
    plt.show()
```

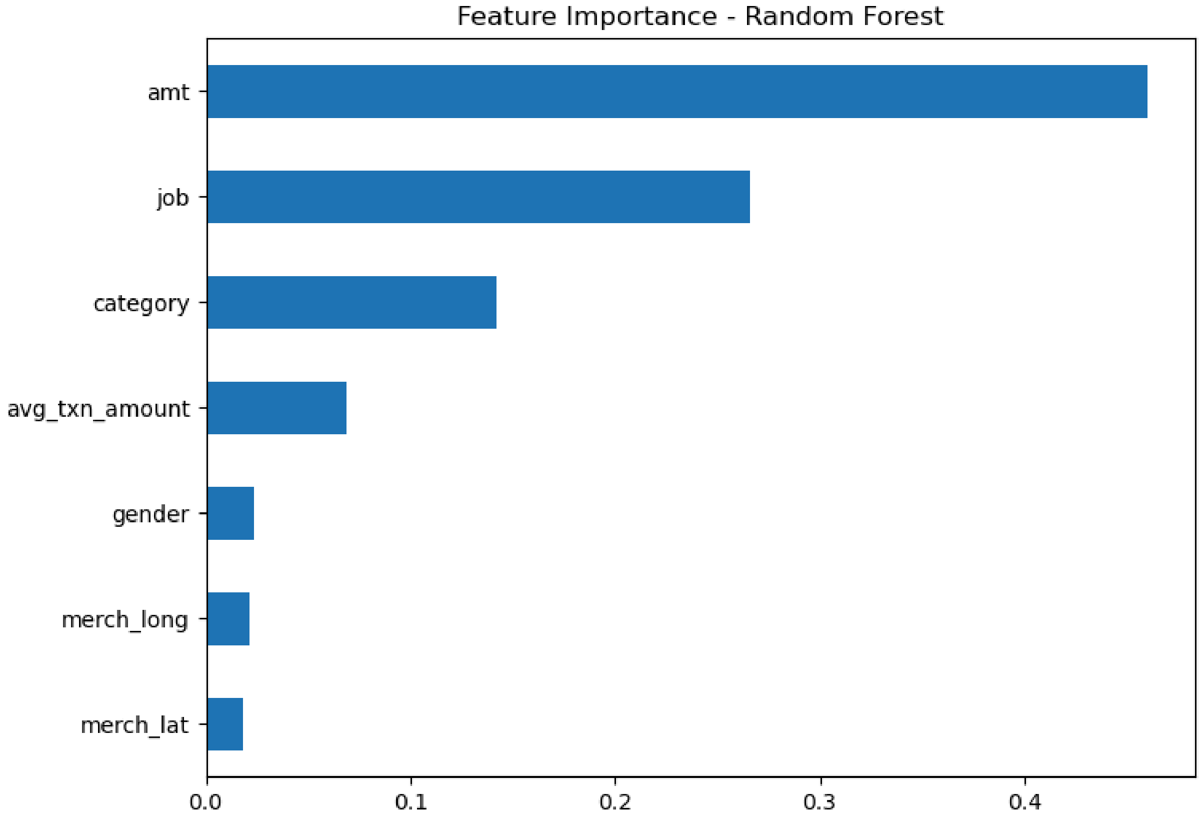
```python

def plot_precision_recall_curves(precisions, recalls, model_name):
    plt.figure(figsize=(8, 6))
    for i in range(len(precisions)):
        plt.plot(recalls[i], precisions[i], lw=1, label=f'Fold {i+1}')
    plt.xlabel('Recall')
    plt.ylabel('Precision')
    plt.title(f'Precision-Recall Curves for {model_name}')
    plt.legend(loc='lower left')
    plt.show()

def plot_confusion_matrix(conf_matrix, model_name):
    plt.figure(figsize=(8, 6))
    sns.heatmap(conf_matrix.astype(int), annot=True, fmt="d", cmap="Blues")
    plt.title(f'Confusion Matrix for {model_name}')
    plt.xlabel('Predicted labels')
    plt.ylabel('True labels')
    plt.show()

# Visualisation
plot_roc_curves(fprs, tprs, "Random Forest")
plot_precision_recall_curves(precisions, recalls, "Random Forest")
plot_confusion_matrix(np.mean(confusion_matrices, axis=0), "Random Forest")

# Feature Importance
feature_importances = pd.Series(classifier.feature_importances_, index=X_train_resa
plt.figure(figsize=(8, 6))
feature_importances.sort_values().plot(kind='barh')
plt.title('Feature Importance - Random Forest')
plt.show()
```

```
Average Accuracy for Random Forest: 1.00
Average F1 Score for Random Forest: 1.00
Average ROC AUC Score for Random Forest: 1.00
Average Precision for Random Forest: 0.99
Average Recall for Random Forest: 0.99
```


ROC Curves for Random Forest
True Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
Fold 1
Fold 2
Fold 3
Fold 4
Fold 5
Fold 6
Fold 7
Fold 8
Fold 9
Fold 10
Random Guess


Precision-Recall Curves for Random Forest
Precision
Recall
1.0
0.9
0.8
0.7
0.6
0.5
0.0
0.2
0.4
0.6
0.8
1.0
Fold 1
Fold 2
Fold 3
Fold 4
Fold 5
Fold 6
Fold 7
Fold 8
Fold 9
Fold 10


Confusion Matrix for Random Forest
True labels
Predicted labels
0
1
0
1
44227
58
14
44271
40000
35000
30000
25000
20000
15000
10000
5000

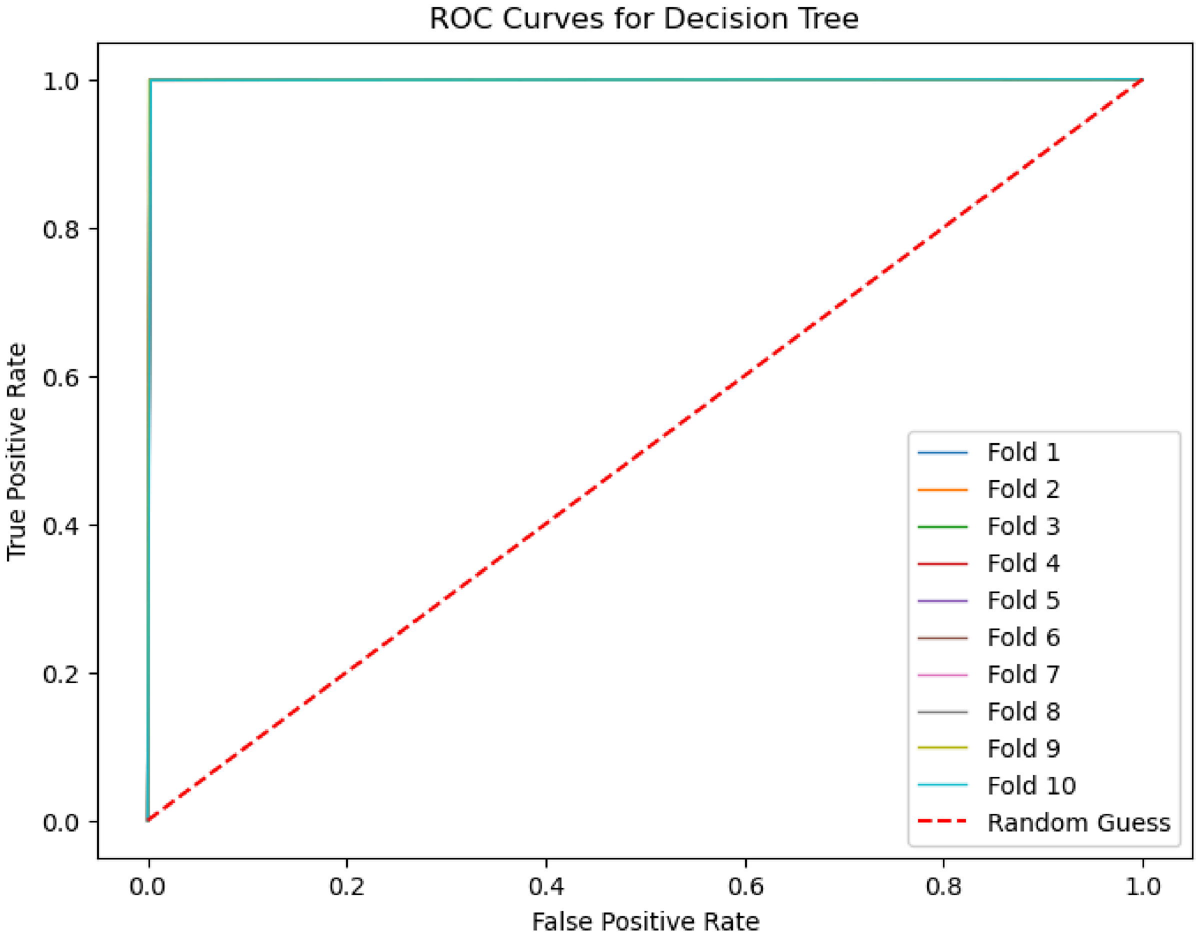

Feature Importance - Random Forest
amt
job
category
avg_txn_amount
gender
merch_long
merch_lat
0.0
0.1
0.2
0.3
0.4

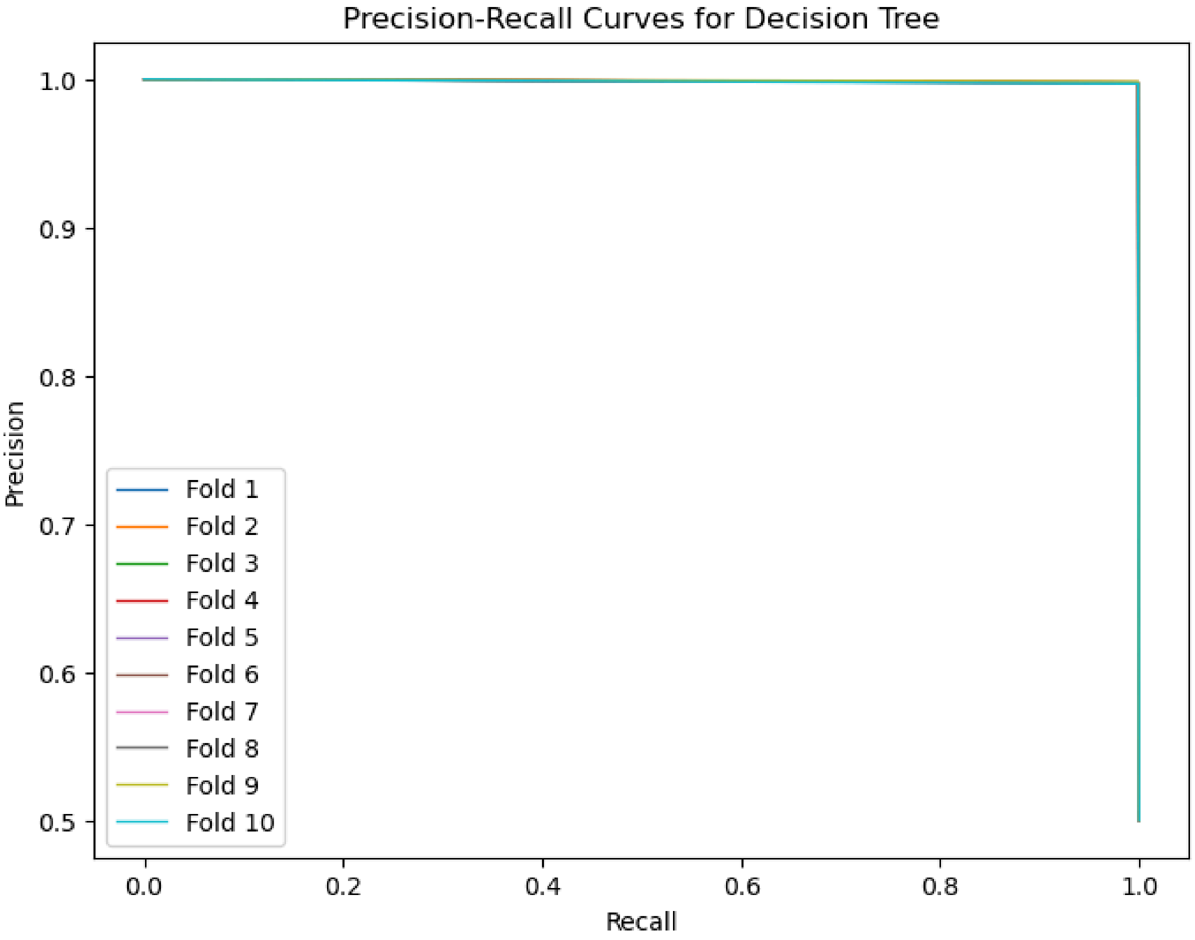
Decision Tree Modelling

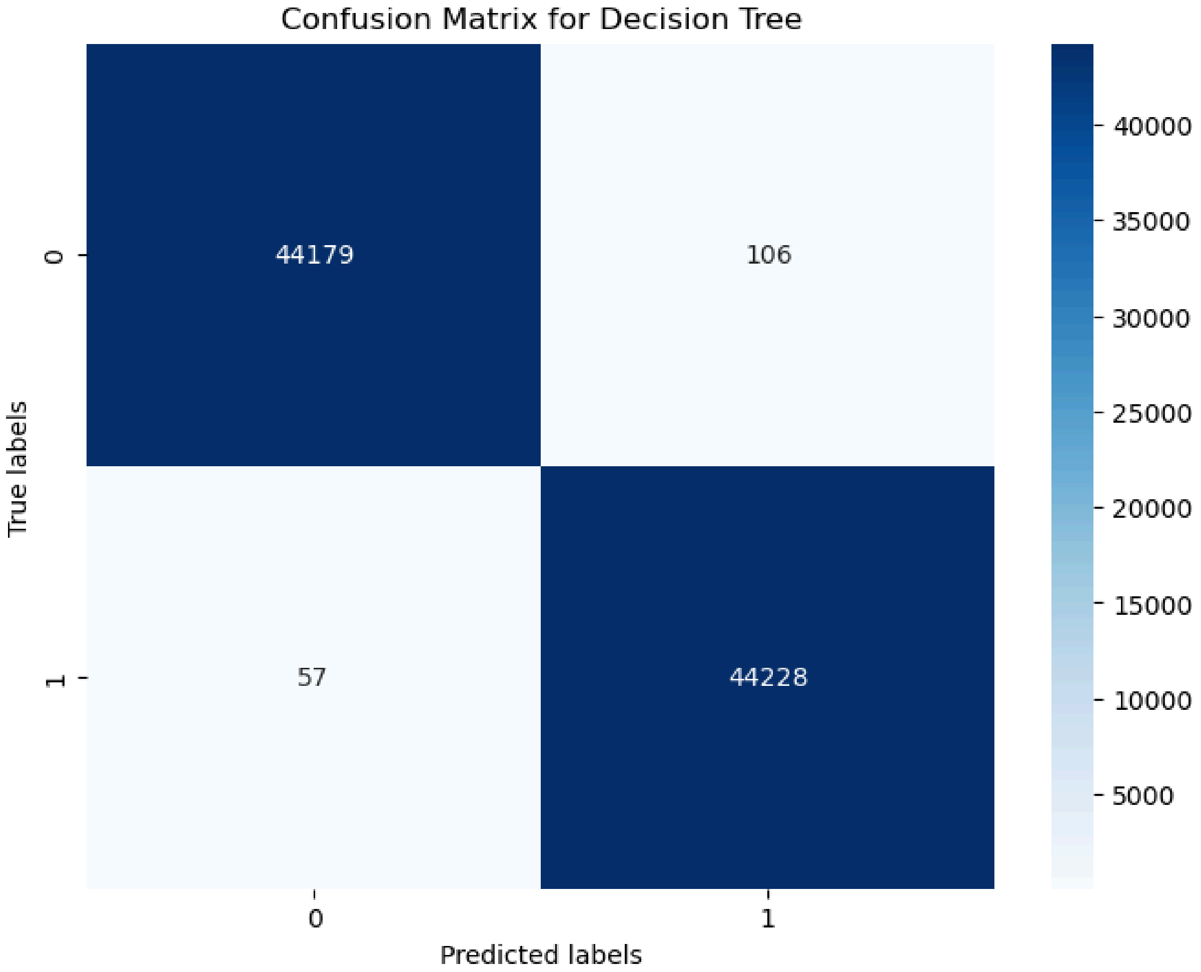
In [49]:

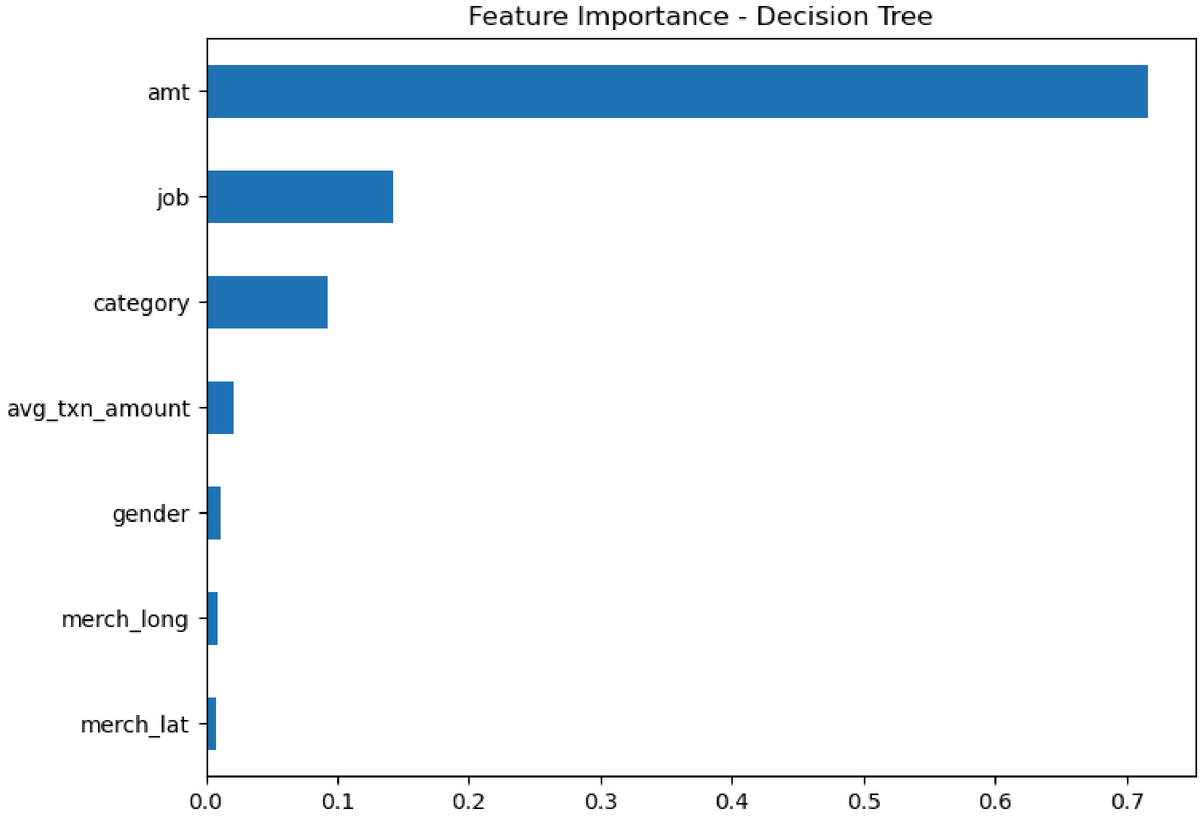
```
# Define classifier
classifier = DecisionTreeClassifier(random_state=42)

# List to store metrics
roc_auc_scores, fprs, tprs, precisions, recalls, f1_scores = [], [], [], [], [], []
confusion_matrices, accuracies = [], []

# Cross-validation
skf = StratifiedKFold(n_splits=10, shuffle=True, random_state=42)

for train_index, test_index in skf.split(X_train_resampled, y_train_resampled):
    X_train_fold, X_test_fold = X_train_resampled.iloc[train_index], X_train_resamp
    y_train_fold, y_test_fold = y_train_resampled.iloc[train_index], y_train_resamp

    # Fit and predict
    classifier.fit(X_train_fold, y_train_fold)
    y_scores = classifier.predict_proba(X_test_fold)[:, 1]
    y_pred = (y_scores > 0.5)

    # Calculate metrics
    fpr, tpr, _ = roc_curve(y_test_fold, y_scores)
    fprs.append(fpr)
    tprs.append(tpr)
    roc_auc_scores.append(auc(fpr, tpr))
    precision, recall, _ = precision_recall_curve(y_test_fold, y_scores)
    precisions.append(precision)
    recalls.append(recall)
    conf_mat = confusion_matrix(y_test_fold, y_pred)
    confusion_matrices.append(conf_mat)
    accuracies.append(accuracy_score(y_test_fold, y_pred))
    f1_scores.append(f1_score(y_test_fold, y_pred))

#results
print(f'Average Accuracy for Decision Tree: {np.mean(accuracies):.2f}')
print(f'Average F1 Score for Decision Tree: {np.mean(f1_scores):.2f}')
print(f'Average ROC AUC Score for Decision Tree: {np.mean(roc_auc_scores):.2f}')
print(f'Average Precision for Decision Tree: {np.mean([np.mean(precision) for preci
print(f'Average Recall for Decision Tree: {np.mean([np.mean(recall) for recall in r

def plot_roc_curves(fprs, tprs, model_name):
    plt.figure(figsize=(8, 6))
    for i in range(len(fprs)):
        plt.plot(fprs[i], tprs[i], lw=1, label=f'Fold {i+1}')
    plt.plot([0, 1], [0, 1], linestyle='--', color='r', label='Random Guess')
    plt.xlabel('False Positive Rate')
    plt.ylabel('True Positive Rate')
    plt.title(f'ROC Curves for {model_name}')
    plt.legend(loc='lower right')
    plt.show()

def plot_precision_recall_curves(precisions, recalls, model_name):
    plt.figure(figsize=(8, 6))
    for i in range(len(precisions)):
        plt.plot(recalls[i], precisions[i], lw=1, label=f'Fold {i+1}')
    plt.xlabel('Recall')
    plt.ylabel('Precision')
    plt.title(f'Precision-Recall Curves for {model_name}')
    plt.legend(loc='lower left')
    plt.show()

def plot_confusion_matrix(conf_matrix, model_name):
    plt.figure(figsize=(8, 6))
    sns.heatmap(conf_matrix.astype(int), annot=True, fmt="d", cmap="Blues")
    plt.title(f'Confusion Matrix for {model_name}')
    plt.xlabel('Predicted labels')
    plt.ylabel('True labels')
    plt.show()

# Visualisations
plot_roc_curves(fprs, tprs, "Decision Tree")
plot_precision_recall_curves(precisions, recalls, "Decision Tree")
plot_confusion_matrix(np.mean(confusion_matrices, axis=0), "Decision Tree")

# Feature importance visualization
feature_importances = pd.Series(classifier.feature_importances_, index=X_train_resa
plt.figure(figsize=(8, 6))
feature_importances.sort_values().plot(kind='barh')
plt.title('Feature Importance - Decision Tree')
plt.show()
```

```
Average Accuracy for Decision Tree: 1.00
Average F1 Score for Decision Tree: 1.00
Average ROC AUC Score for Decision Tree: 1.00
Average Precision for Decision Tree: 0.83
Average Recall for Decision Tree: 0.67
```


ROC Curves for Decision Tree
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.2
0.4
0.6
0.8
1.0
Fold 1
Fold 2
Fold 3
Fold 4
Fold 5
Fold 6
Fold 7
Fold 8
Fold 9
Fold 10
Random Guess


Precision-Recall Curves for Decision Tree
Precision
Recall
1.0
0.9
0.8
0.7
0.6
0.5
0.0
0.2
0.4
0.6
0.8
1.0
Fold 1
Fold 2
Fold 3
Fold 4
Fold 5
Fold 6
Fold 7
Fold 8
Fold 9
Fold 10


Confusion Matrix for Decision Tree
True labels
0
1
44179
106
57
44228
Predicted labels
0
1
40000
35000
30000
25000
20000
15000
10000
5000


Feature Importance - Decision Tree
amt
job
category
avg_txn_amount
gender
merch_long
merch_lat
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7

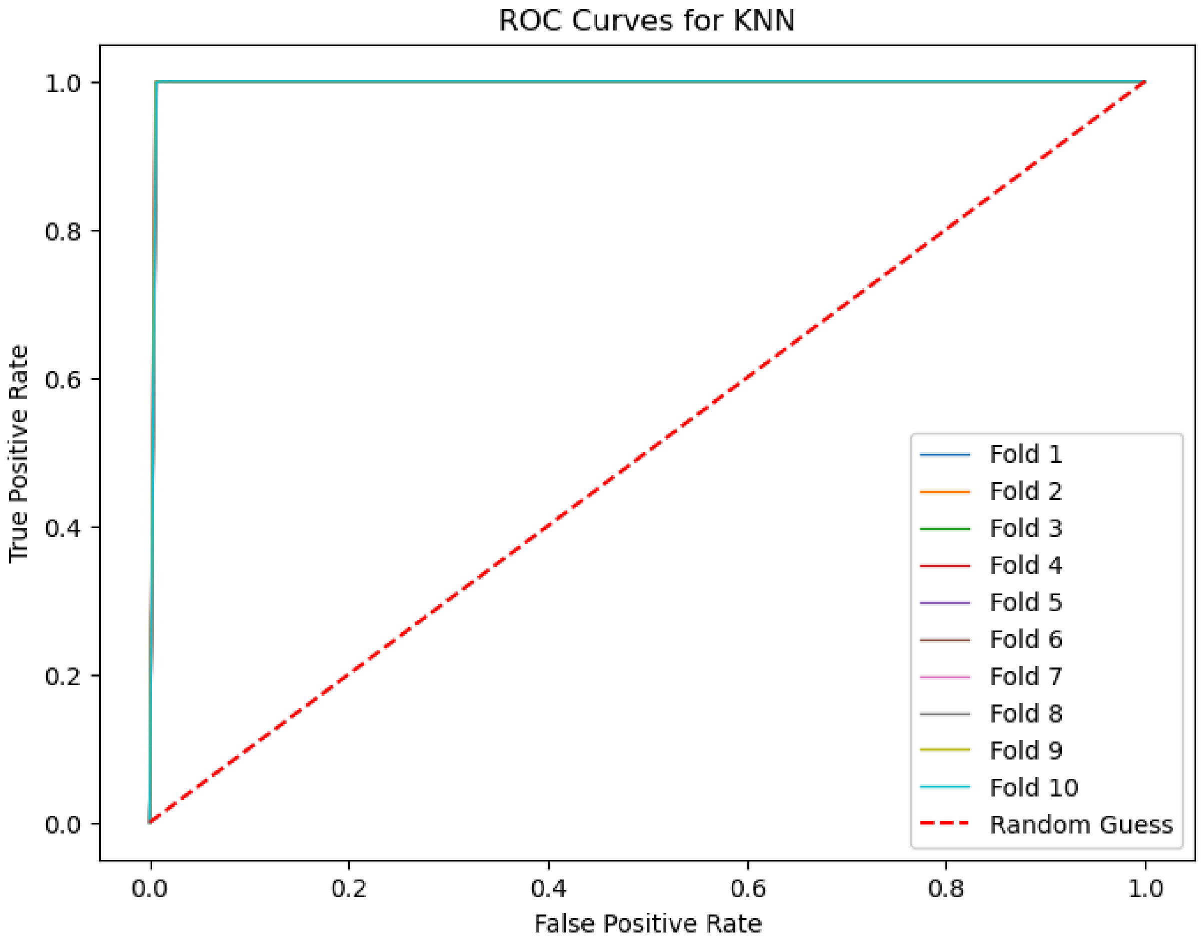
KNN Modelling

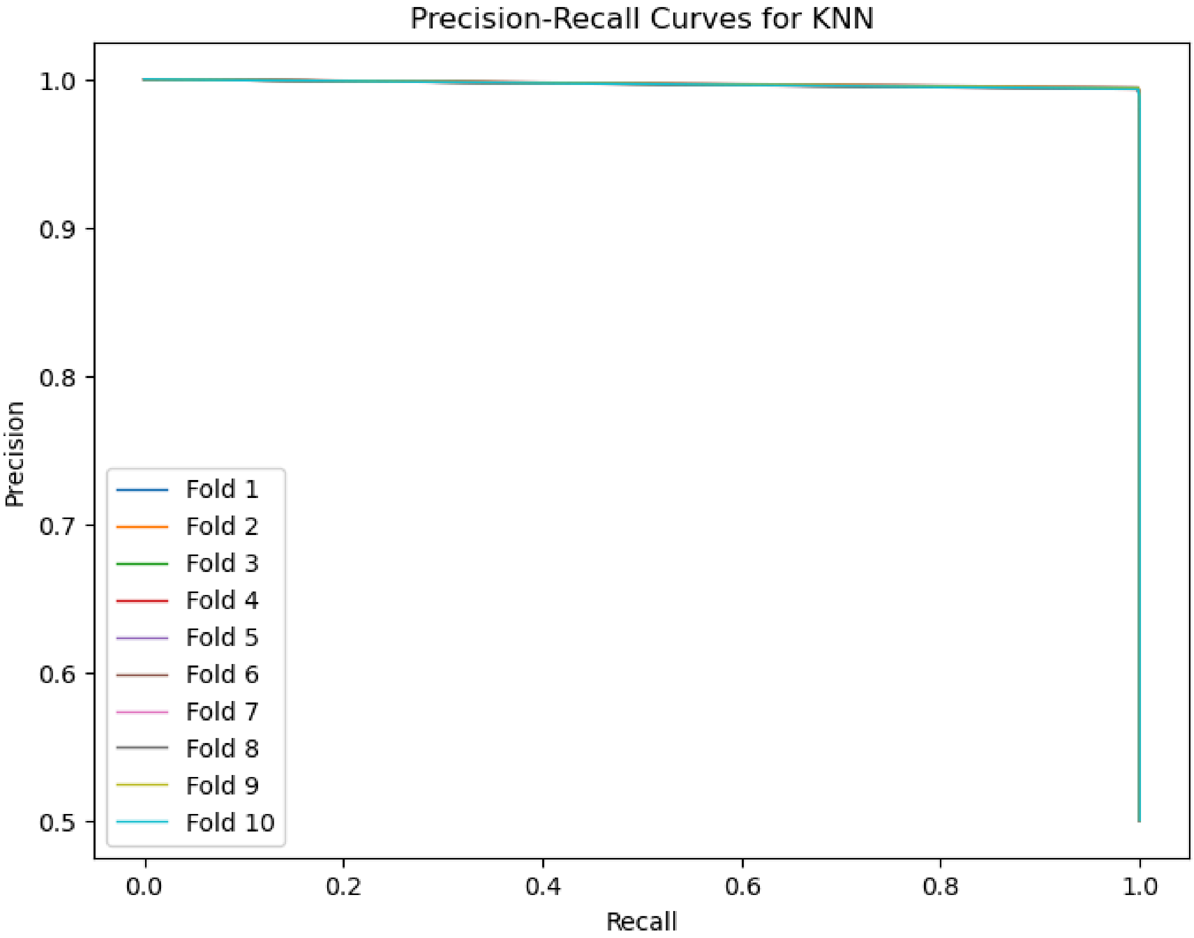
In [51]:

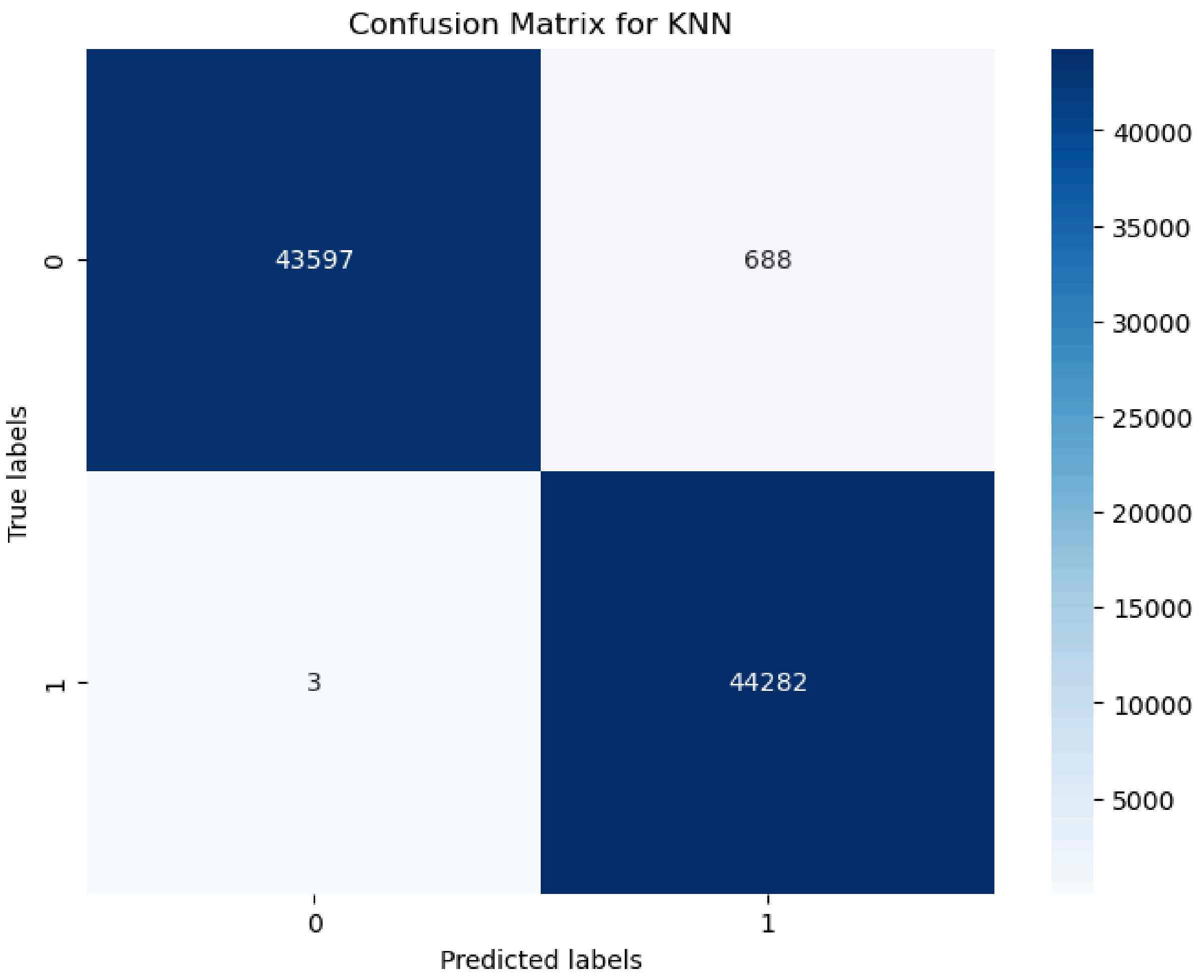
```
# Define classifier
classifier = KNeighborsClassifier()

# Store metrics
roc_auc_scores, fprs, tprs, precisions, recalls, f1_scores = [], [], [], [], [], []
confusion_matrices, accuracies = [], []

# Cross-validation
skf = StratifiedKFold(n_splits=10, shuffle=True, random_state=42)

for train_index, test_index in skf.split(X_train_resampled, y_train_resampled):
    X_train_fold, X_test_fold = X_train_resampled.iloc[train_index], X_train_resamp
    y_train_fold, y_test_fold = y_train_resampled.iloc[train_index], y_train_resamp

    # Fit and predict
    classifier.fit(X_train_fold, y_train_fold)
    y_scores = classifier.predict_proba(X_test_fold)[:, 1]
    y_pred = (y_scores > 0.5)

    # Collect and calculate metrics
    fpr, tpr, _ = roc_curve(y_test_fold, y_scores)
    fprs.append(fpr)
    tprs.append(tpr)
    roc_auc_scores.append(auc(fpr, tpr))
    precision, recall, _ = precision_recall_curve(y_test_fold, y_scores)
    precisions.append(precision)
    recalls.append(recall)
    conf_mat = confusion_matrix(y_test_fold, y_pred)
    confusion_matrices.append(conf_mat)
    accuracies.append(accuracy_score(y_test_fold, y_pred))
    f1_scores.append(f1_score(y_test_fold, y_pred))

# Results
print(f'Average Accuracy for KNN: {np.mean(accuracies):.2f}')
print(f'Average F1 Score for KNN: {np.mean(f1_scores):.2f}')
print(f'Average ROC AUC Score for KNN: {np.mean(roc_auc_scores):.2f}')
print(f'Average Precision for KNN: {np.mean([np.mean(precision) for precision in pr
print(f'Average Recall for KNN: {np.mean([np.mean(recall) for recall in recalls]):.

def plot_roc_curves(fprs, tprs, model_name):
    plt.figure(figsize=(8, 6))
    for i in range(len(fprs)):
        plt.plot(fprs[i], tprs[i], lw=1, label=f'Fold {i+1}')
    plt.plot([0, 1], [0, 1], linestyle='--', color='r', label='Random Guess')
    plt.xlabel('False Positive Rate')
    plt.ylabel('True Positive Rate')
    plt.title(f'ROC Curves for {model_name}')
    plt.legend(loc='lower right')
    plt.show()

def plot_precision_recall_curves(precisions, recalls, model_name):
    plt.figure(figsize=(8, 6))
```

```python
    for i in range(len(precisions)):
        plt.plot(recalls[i], precisions[i], lw=1, label=f'Fold {i+1}')
    plt.xlabel('Recall')
    plt.ylabel('Precision')
    plt.title(f'Precision-Recall Curves for {model_name}')
    plt.legend(loc='lower left')
    plt.show()

def plot_confusion_matrix(conf_matrix, model_name):
    plt.figure(figsize=(8, 6))
    sns.heatmap(conf_matrix.astype(int), annot=True, fmt="d", cmap="Blues")
    plt.title(f'Confusion Matrix for {model_name}')
    plt.xlabel('Predicted labels')
    plt.ylabel('True labels')
    plt.show()

# Visualisations
plot_roc_curves(fprs, tprs, "KNN")
plot_precision_recall_curves(precisions, recalls, "KNN")
plot_confusion_matrix(np.mean(confusion_matrices, axis=0), "KNN")
```

```
Average Accuracy for KNN: 0.99
Average F1 Score for KNN: 0.99
Average ROC AUC Score for KNN: 1.00
Average Precision for KNN: 0.92
Average Recall for KNN: 0.86
```



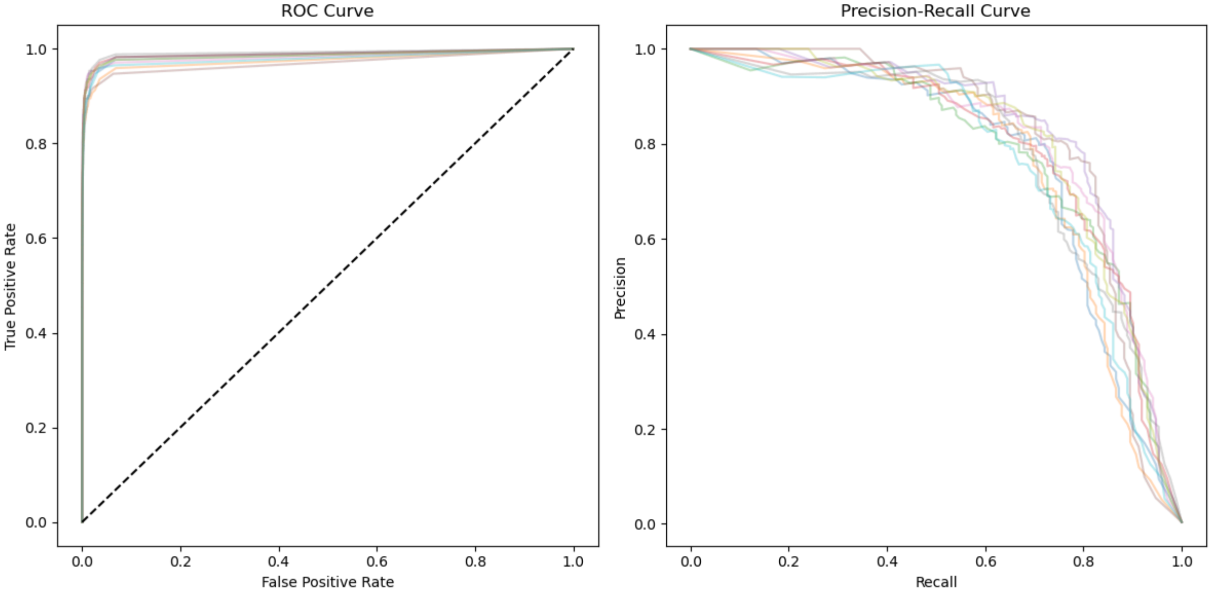

Precision-Recall Curves for KNN
Precision
Recall
1.0
0.9
0.8
0.7
0.6
0.5
0.0
0.2
0.4
0.6
0.8
1.0
Fold 1
Fold 2
Fold 3
Fold 4
Fold 5
Fold 6
Fold 7
Fold 8
Fold 9
Fold 10



Dynamic Synthetic Oversampling with Stratified K-Fold Cross-Validation within a Pipeline Framework

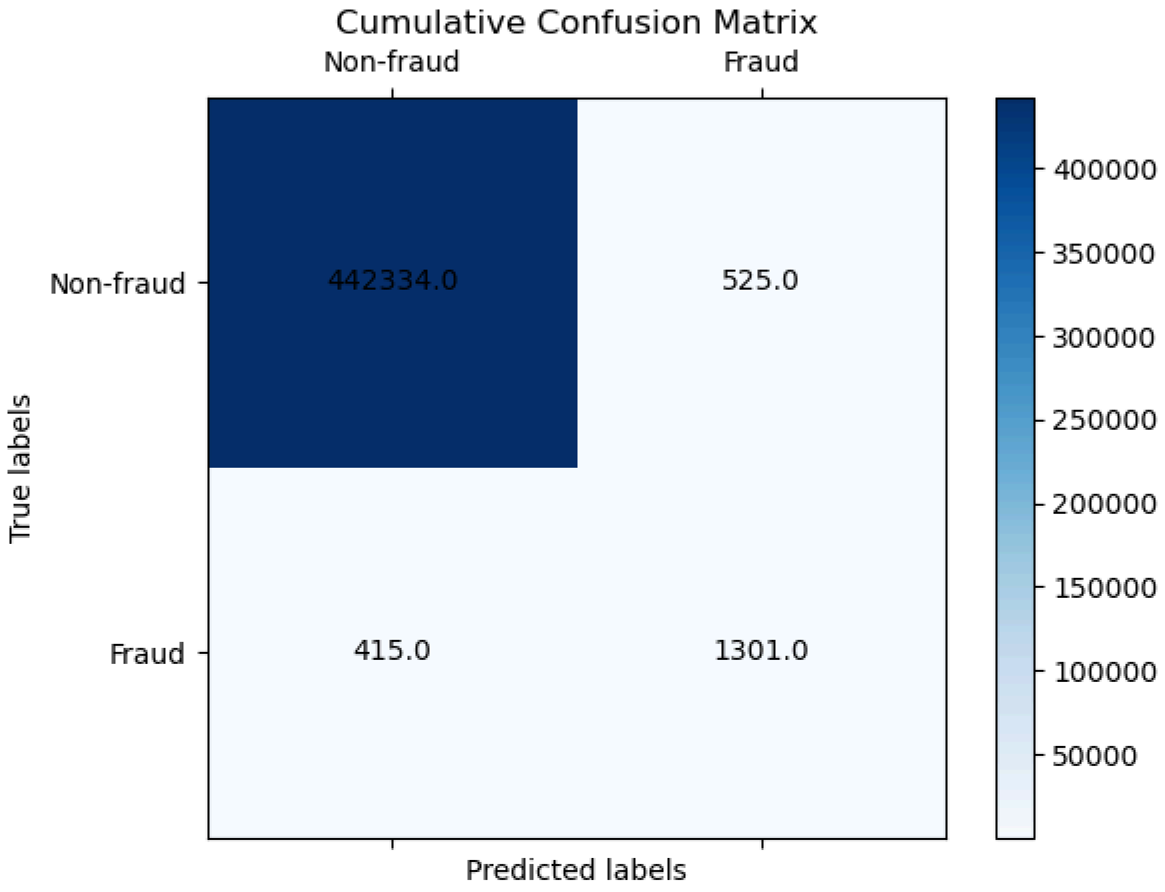
In [53]:
```
# Define the pipeline
pipeline = Pipeline([
    ('smote', SMOTE(random_state=1)),
    ('classifier', RandomForestClassifier(n_estimators=100, random_state=0, n_jobs=
])

# Prepare cross-validation
skf = StratifiedKFold(n_splits=10, shuffle=True, random_state=42)
scores = []
cumulative_conf_matrix = np.zeros((2, 2))  # Assuming binary classification

fig, (ax_roc, ax_pr) = plt.subplots(1, 2, figsize=(12, 6))

# Cross-validation loop
for train_index, test_index in skf.split(X_train, y_train):
    X_train_fold, X_test_fold = X_train.iloc[train_index], X_train.iloc[test_index]
    y_train_fold, y_test_fold = y_train.iloc[train_index], y_train.iloc[test_index]

    # Fiting the model
    pipeline.fit(X_train_fold, y_train_fold)
    y_pred = pipeline.predict(X_test_fold)
    y_probs = pipeline.predict_proba(X_test_fold)[:, 1]

    # Metrics collection
    scores.append({
        'roc_auc': roc_auc_score(y_test_fold, y_probs),
        'f1': f1_score(y_test_fold, y_pred),
        'accuracy': accuracy_score(y_test_fold, y_pred),
        'recall': recall_score(y_test_fold, y_pred),
        'precision': precision_score(y_test_fold, y_pred)
    })

    # confusion matrix
    cumulative_conf_matrix += confusion_matrix(y_test_fold, y_pred)

    # ROC curve
    fpr, tpr, _ = roc_curve(y_test_fold, y_probs)
    ax_roc.plot(fpr, tpr, alpha=0.3)

    # Precision-Recall curve
    precision, recall, _ = precision_recall_curve(y_test_fold, y_probs)
    ax_pr.plot(recall, precision, alpha=0.3)

# Average metrics and ploting ROC & Precision-Recall
ax_roc.plot([0, 1], [0, 1], 'k--')
ax_roc.set_title('ROC Curve')
ax_roc.set_xlabel('False Positive Rate')
ax_roc.set_ylabel('True Positive Rate')

ax_pr.set_title('Precision-Recall Curve')
ax_pr.set_xlabel('Recall')
ax_pr.set_ylabel('Precision')

plt.tight_layout()
plt.show()

# Calculate average metrics
average_scores = {metric: np.mean([score[metric] for score in scores]) for metric i
print("Average Metrics:", average_scores)

# display the cumulative confusion matrix
print("Cumulative Confusion Matrix:\n", cumulative_conf_matrix.astype(int))

# Defining the plot function for cumulative confusion matrix
def plot_cumulative_confusion_matrix(matrix):
    fig, ax = plt.subplots()
    cax = ax.matshow(matrix, cmap='Blues')
    plt.title('Cumulative Confusion Matrix')
    fig.colorbar(cax)
    ax.set_xlabel('Predicted labels')
    ax.set_ylabel('True labels')
    ax.set_xticklabels([''] + ['Non-fraud', 'Fraud'])
    ax.set_yticklabels([''] + ['Non-fraud', 'Fraud'])

    # Looping over data dimensions and create text annotations.
    for i in range(matrix.shape[0]):
        for j in range(matrix.shape[1]):
            ax.text(j, i, str(matrix[i, j]), ha="center", va="center", color="black

    plt.show()

# Call the plot function
plot_cumulative_confusion_matrix(cumulative_conf_matrix)

# Feature Importance Visualisation
feature_importances = pd.Series(pipeline.named_steps['classifier'].feature_importan
plt.figure(figsize=(10, 8))
feature_importances.sort_values().plot(kind='barh')
plt.title('Feature Importance - RandomForest')
plt.show()

# DataFrame for the metrics
metrics_df = pd.DataFrame(scores)
print(metrics_df.describe().transpose()[['mean', 'std']])
```

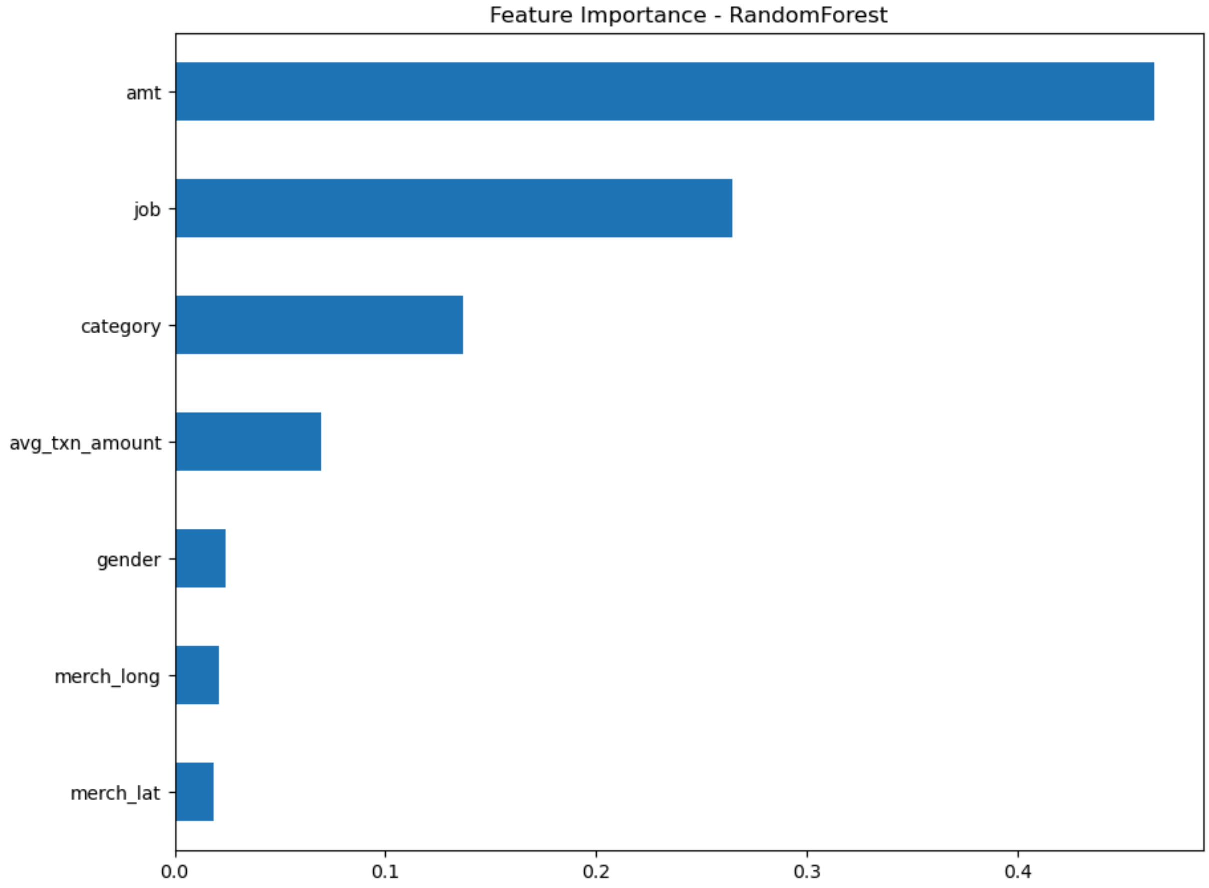
```
C:\Users\Ifean\anaconda3\Lib\site-packages\sklearn\base.py:474: FutureWarning: `Base
Estimator._validate_data` is deprecated in 1.6 and will be removed in 1.7. Use `skle
arn.utils.validation.validate_data` instead. This function becomes public and is par
t of the scikit-learn developer API.
  warnings.warn(
C:\Users\Ifean\anaconda3\Lib\site-packages\sklearn\base.py:474: FutureWarning: `Base
Estimator._validate_data` is deprecated in 1.6 and will be removed in 1.7. Use `skle
arn.utils.validation.validate_data` instead. This function becomes public and is par
t of the scikit-learn developer API.
  warnings.warn(
C:\Users\Ifean\anaconda3\Lib\site-packages\sklearn\base.py:474: FutureWarning: `Base
Estimator._validate_data` is deprecated in 1.6 and will be removed in 1.7. Use `skle
arn.utils.validation.validate_data` instead. This function becomes public and is par
t of the scikit-learn developer API.
  warnings.warn(
C:\Users\Ifean\anaconda3\Lib\site-packages\sklearn\base.py:474: FutureWarning: `Base
Estimator._validate_data` is deprecated in 1.6 and will be removed in 1.7. Use `skle
arn.utils.validation.validate_data` instead. This function becomes public and is par
t of the scikit-learn developer API.
  warnings.warn(
C:\Users\Ifean\anaconda3\Lib\site-packages\sklearn\base.py:474: FutureWarning: `Base
Estimator._validate_data` is deprecated in 1.6 and will be removed in 1.7. Use `skle
arn.utils.validation.validate_data` instead. This function becomes public and is par
t of the scikit-learn developer API.
  warnings.warn(
C:\Users\Ifean\anaconda3\Lib\site-packages\sklearn\base.py:474: FutureWarning: `Base
Estimator._validate_data` is deprecated in 1.6 and will be removed in 1.7. Use `skle
arn.utils.validation.validate_data` instead. This function becomes public and is par
t of the scikit-learn developer API.
  warnings.warn(
C:\Users\Ifean\anaconda3\Lib\site-packages\sklearn\base.py:474: FutureWarning: `Base
Estimator._validate_data` is deprecated in 1.6 and will be removed in 1.7. Use `skle
arn.utils.validation.validate_data` instead. This function becomes public and is par
t of the scikit-learn developer API.
  warnings.warn(
C:\Users\Ifean\anaconda3\Lib\site-packages\sklearn\base.py:474: FutureWarning: `Base
Estimator._validate_data` is deprecated in 1.6 and will be removed in 1.7. Use `skle
arn.utils.validation.validate_data` instead. This function becomes public and is par
t of the scikit-learn developer API.
  warnings.warn(
C:\Users\Ifean\anaconda3\Lib\site-packages\sklearn\base.py:474: FutureWarning: `Base
Estimator._validate_data` is deprecated in 1.6 and will be removed in 1.7. Use `skle
arn.utils.validation.validate_data` instead. This function becomes public and is par
t of the scikit-learn developer API.
  warnings.warn(
C:\Users\Ifean\anaconda3\Lib\site-packages\sklearn\base.py:474: FutureWarning: `Base
Estimator._validate_data` is deprecated in 1.6 and will be removed in 1.7. Use `skle
arn.utils.validation.validate_data` instead. This function becomes public and is par
t of the scikit-learn developer API.
  warnings.warn(
```



```
Average Metrics: {'roc_auc': 0.9832867517538381, 'f1': 0.7346512270440831, 'accurac
y': 0.9978856208961536, 'recall': 0.758187134502924, 'precision': 0.713606678537408
4}
Cumulative Confusion Matrix:
 [[442334    525]
 [   415   1301]]
```

```
C:\Users\Ifean\AppData\Local\Temp\ipykernel_20284\1974960730.py:72: UserWarning: set
_ticklabels() should only be used with a fixed number of ticks, i.e. after set_ticks
() or using a FixedLocator.
  ax.set_xticklabels([''] + ['Non-fraud', 'Fraud'])
C:\Users\Ifean\AppData\Local\Temp\ipykernel_20284\1974960730.py:73: UserWarning: set
_ticklabels() should only be used with a fixed number of ticks, i.e. after set_ticks
() or using a FixedLocator.
  ax.set_yticklabels([''] + ['Non-fraud', 'Fraud'])
```


Cumulative Confusion Matrix
Non-fraud
Fraud
Non-fraud
Fraud
442334.0
525.0
415.0
1301.0
True labels
Predicted labels
400000
350000
300000
250000
200000
150000
100000
50000


Feature Importance - RandomForest
amt
job
category
avg_txn_amount
gender
merch_long
merch_lat
0.0
0.1
0.2
0.3
0.4

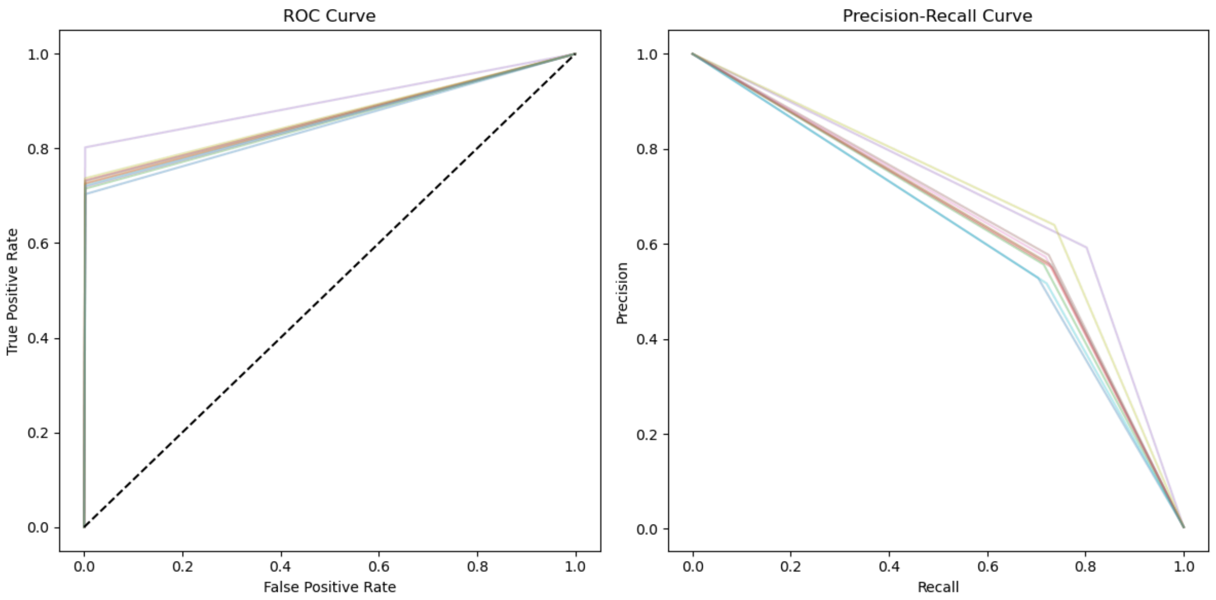
```
               mean       std
roc_auc    0.983287  0.006776
f1         0.734651  0.026385
accuracy   0.997886  0.000226
recall     0.758187  0.035515
precision  0.713607  0.032880
```

Decision Tree

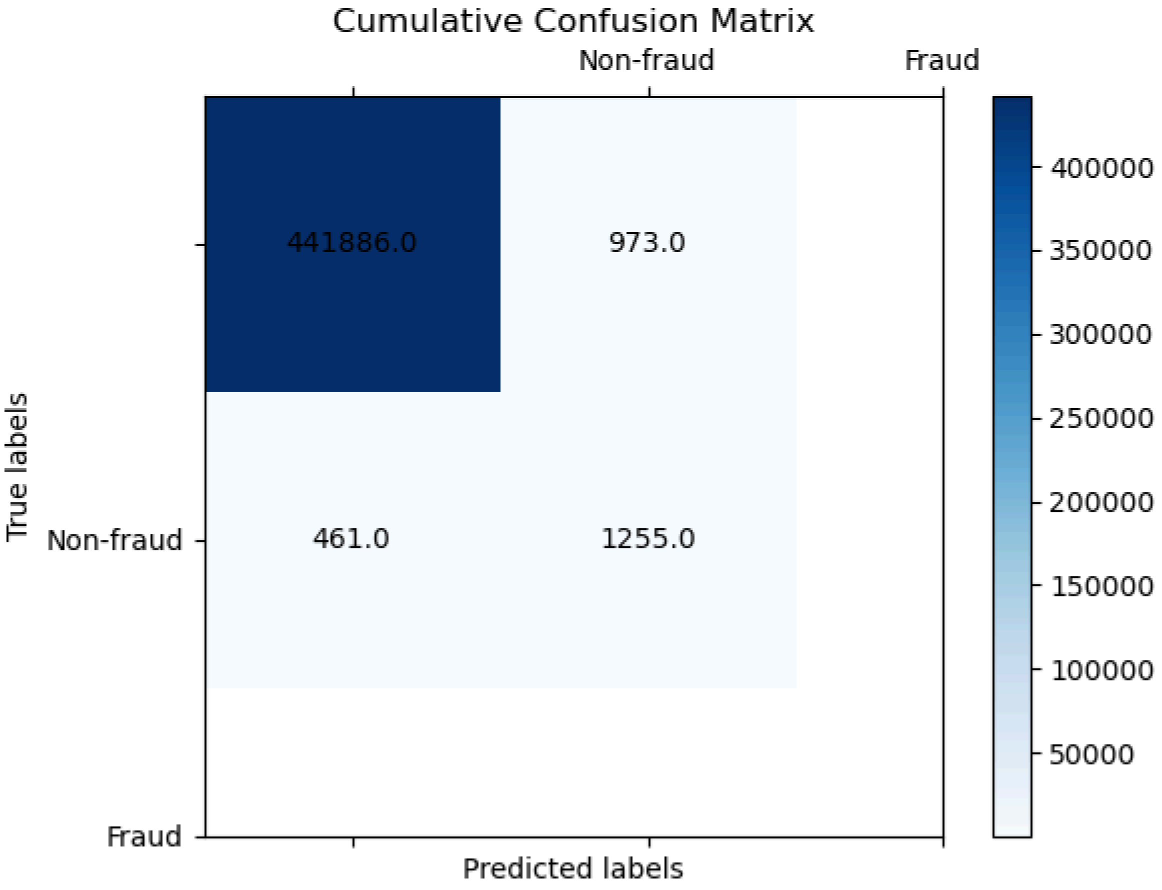
In [55]:

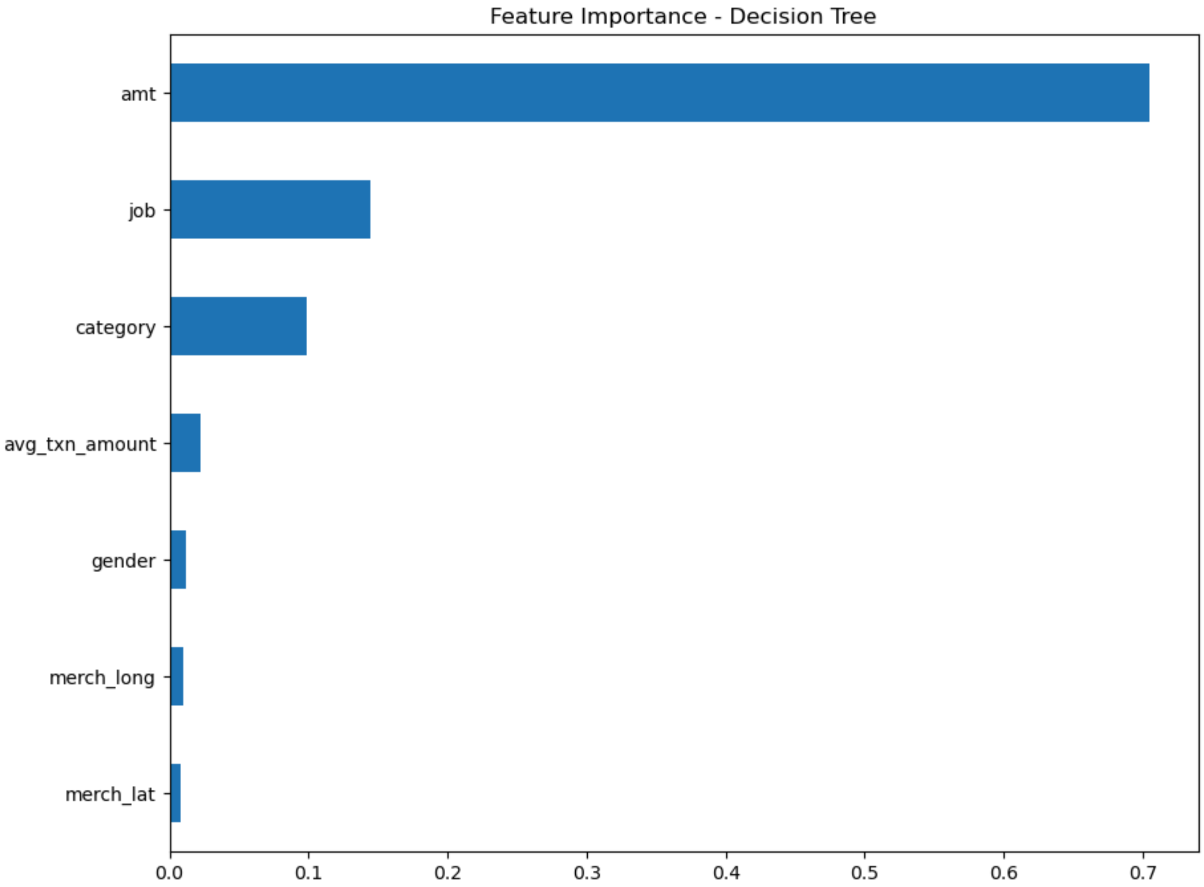
```
# Define the pipeline
pipeline = Pipeline([
    ('smote', SMOTE(random_state=1)),
    ('classifier', DecisionTreeClassifier(random_state=42))
])

# Cross-validation
skf = StratifiedKFold(n_splits=10, shuffle=True, random_state=42)
scores = []
cumulative_conf_matrix = np.zeros((2, 2))

fig, (ax_roc, ax_pr) = plt.subplots(1, 2, figsize=(12, 6))

# Cross-validation loop
for train_index, test_index in skf.split(X_train, y_train):
    X_train_fold, X_test_fold = X_train.iloc[train_index], X_train.iloc[test_index]
    y_train_fold, y_test_fold = y_train.iloc[train_index], y_train.iloc[test_index]

    # Fitting the model
    pipeline.fit(X_train_fold, y_train_fold)
    y_pred = pipeline.predict(X_test_fold)
    y_probs = pipeline.predict_proba(X_test_fold)[:, 1]

    # Metrics collection
    scores.append({
        'roc_auc': roc_auc_score(y_test_fold, y_probs),
        'f1': f1_score(y_test_fold, y_pred),
        'accuracy': accuracy_score(y_test_fold, y_pred),
        'recall': recall_score(y_test_fold, y_pred),
        'precision': precision_score(y_test_fold, y_pred)
    })

    # Confusion matrix
    cumulative_conf_matrix += confusion_matrix(y_test_fold, y_pred).astype(int)

    # ROC curve
    fpr, tpr, _ = roc_curve(y_test_fold, y_probs)
    ax_roc.plot(fpr, tpr, alpha=0.3)

    # Precision-Recall curve
    precision, recall, _ = precision_recall_curve(y_test_fold, y_probs)
    ax_pr.plot(recall, precision, alpha=0.3)

# Finalize ROC & Precision-Recall plots
ax_roc.plot([0, 1], [0, 1], 'k--')
ax_roc.set_title('ROC Curve')
ax_roc.set_xlabel('False Positive Rate')
ax_roc.set_ylabel('True Positive Rate')

ax_pr.set_title('Precision-Recall Curve')
ax_pr.set_xlabel('Recall')
ax_pr.set_ylabel('Precision')

plt.tight_layout()
plt.show()

# Calculate average metrics
average_scores = {metric: np.mean([score[metric] for score in scores]) for metric i
print("Average Metrics for Decision Tree:", average_scores)


def plot_cumulative_confusion_matrix(matrix):
    fig, ax = plt.subplots()
    cax = ax.matshow(matrix, cmap='Blues')
    plt.title('Cumulative Confusion Matrix')
    fig.colorbar(cax)
    ax.set_xlabel('Predicted labels')
    ax.set_ylabel('True labels')
    ax.set_xticks([0, 1, 2])  # Set the ticks first
    ax.set_xticklabels(['', 'Non-fraud', 'Fraud'])
    ax.set_yticks([0, 1, 2])  # Set the ticks first
    ax.set_yticklabels(['', 'Non-fraud', 'Fraud'])

    # Looping over data dimensions and create text annotations.
    for i in range(matrix.shape[0]):  # Corrected indentation
        for j in range(matrix.shape[1]):  # Corrected indentation
            ax.text(j, i, str(matrix[i, j]), ha="center", va="center", color="black

    plt.show()

plot_cumulative_confusion_matrix(cumulative_conf_matrix)

# Feature importance visualization
feature_importances = pd.Series(pipeline.named_steps['classifier'].feature_importan
plt.figure(figsize=(10, 8))
feature_importances.sort_values().plot(kind='barh')
plt.title('Feature Importance - Decision Tree')
plt.show()
```

```
C:\Users\Ifean\anaconda3\Lib\site-packages\sklearn\base.py:474: FutureWarning: `Base
Estimator._validate_data` is deprecated in 1.6 and will be removed in 1.7. Use `skle
arn.utils.validation.validate_data` instead. This function becomes public and is par
t of the scikit-learn developer API.
  warnings.warn(
C:\Users\Ifean\anaconda3\Lib\site-packages\sklearn\base.py:474: FutureWarning: `Base
Estimator._validate_data` is deprecated in 1.6 and will be removed in 1.7. Use `skle
arn.utils.validation.validate_data` instead. This function becomes public and is par
t of the scikit-learn developer API.
  warnings.warn(
C:\Users\Ifean\anaconda3\Lib\site-packages\sklearn\base.py:474: FutureWarning: `Base
Estimator._validate_data` is deprecated in 1.6 and will be removed in 1.7. Use `skle
arn.utils.validation.validate_data` instead. This function becomes public and is par
t of the scikit-learn developer API.
  warnings.warn(
C:\Users\Ifean\anaconda3\Lib\site-packages\sklearn\base.py:474: FutureWarning: `Base
Estimator._validate_data` is deprecated in 1.6 and will be removed in 1.7. Use `skle
arn.utils.validation.validate_data` instead. This function becomes public and is par
t of the scikit-learn developer API.
  warnings.warn(
C:\Users\Ifean\anaconda3\Lib\site-packages\sklearn\base.py:474: FutureWarning: `Base
Estimator._validate_data` is deprecated in 1.6 and will be removed in 1.7. Use `skle
arn.utils.validation.validate_data` instead. This function becomes public and is par
t of the scikit-learn developer API.
  warnings.warn(
C:\Users\Ifean\anaconda3\Lib\site-packages\sklearn\base.py:474: FutureWarning: `Base
Estimator._validate_data` is deprecated in 1.6 and will be removed in 1.7. Use `skle
arn.utils.validation.validate_data` instead. This function becomes public and is par
t of the scikit-learn developer API.
  warnings.warn(
C:\Users\Ifean\anaconda3\Lib\site-packages\sklearn\base.py:474: FutureWarning: `Base
Estimator._validate_data` is deprecated in 1.6 and will be removed in 1.7. Use `skle
arn.utils.validation.validate_data` instead. This function becomes public and is par
t of the scikit-learn developer API.
  warnings.warn(
C:\Users\Ifean\anaconda3\Lib\site-packages\sklearn\base.py:474: FutureWarning: `Base
Estimator._validate_data` is deprecated in 1.6 and will be removed in 1.7. Use `skle
arn.utils.validation.validate_data` instead. This function becomes public and is par
t of the scikit-learn developer API.
  warnings.warn(
C:\Users\Ifean\anaconda3\Lib\site-packages\sklearn\base.py:474: FutureWarning: `Base
Estimator._validate_data` is deprecated in 1.6 and will be removed in 1.7. Use `skle
arn.utils.validation.validate_data` instead. This function becomes public and is par
t of the scikit-learn developer API.
  warnings.warn(
C:\Users\Ifean\anaconda3\Lib\site-packages\sklearn\base.py:474: FutureWarning: `Base
Estimator._validate_data` is deprecated in 1.6 and will be removed in 1.7. Use `skle
arn.utils.validation.validate_data` instead. This function becomes public and is par
t of the scikit-learn developer API.
  warnings.warn(
```



```
Average Metrics for Decision Tree: {'roc_auc': 0.8645736303655047, 'f1': 0.636849924
4464731, 'accuracy': 0.9967744481633757, 'recall': 0.7313443492452059, 'precision':
0.5646127438611479}
```



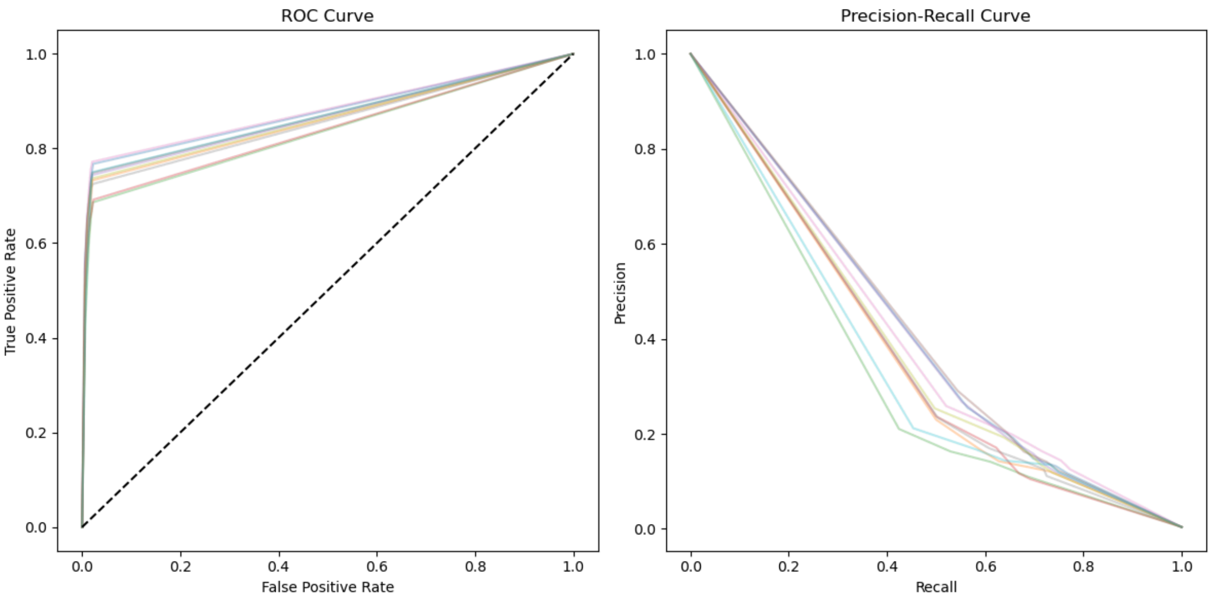


KNN

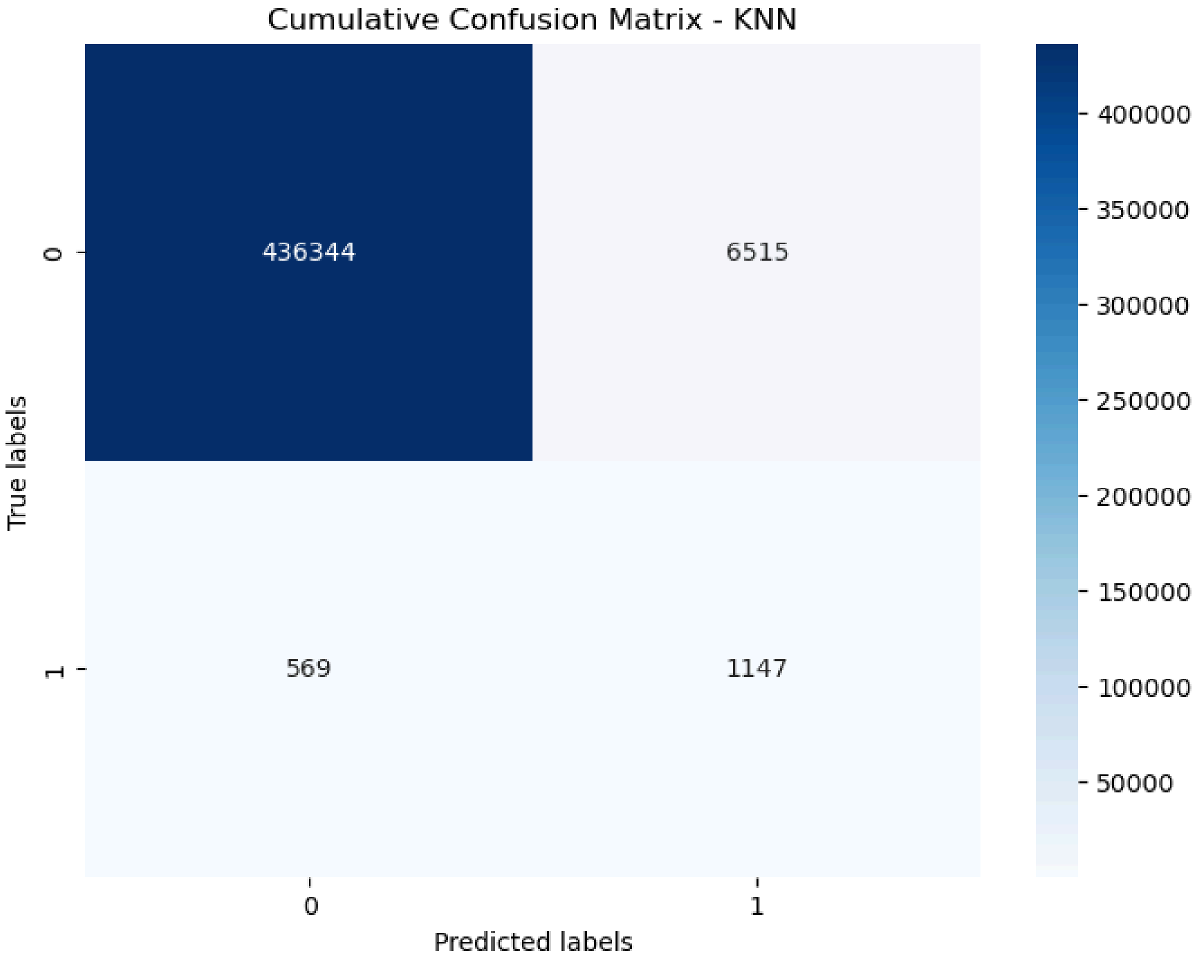
In [57]:

```
# Define the pipeline with imblearn's Pipeline
pipeline = ImbPipeline([
    ('smote', SMOTE(random_state=1)),
    ('classifier', KNeighborsClassifier())
])

# Cross-validation setup
skf = StratifiedKFold(n_splits=10, shuffle=True, random_state=42)
scores = []
cumulative_conf_matrix = np.zeros((2, 2), dtype=int)

fig, (ax_roc, ax_pr) = plt.subplots(1, 2, figsize=(12, 6))

# Cross-validation loop
for train_index, test_index in skf.split(X_train, y_train):
    X_train_fold, X_test_fold = X_train.iloc[train_index], X_train.iloc[test_index]
    y_train_fold, y_test_fold = y_train.iloc[train_index], y_train.iloc[test_index]

    # Fit the model
    pipeline.fit(X_train_fold, y_train_fold)
    y_pred = pipeline.predict(X_test_fold)
    y_probs = pipeline.predict_proba(X_test_fold)[:, 1]

    # Metrics collection
    scores.append({
        'roc_auc': roc_auc_score(y_test_fold, y_probs),
        'f1': f1_score(y_test_fold, y_pred),
        'accuracy': accuracy_score(y_test_fold, y_pred),
        'recall': recall_score(y_test_fold, y_pred),
        'precision': precision_score(y_test_fold, y_pred)
    })

    # Confusion matrix update
    cumulative_conf_matrix += confusion_matrix(y_test_fold, y_pred)

    # ROC curve
    fpr, tpr, _ = roc_curve(y_test_fold, y_probs)
    ax_roc.plot(fpr, tpr, alpha=0.3)

    # Precision-Recall curve
    precision, recall, _ = precision_recall_curve(y_test_fold, y_probs)
    ax_pr.plot(recall, precision, alpha=0.3)

# Finalizing ROC & Precision-Recall plots
ax_roc.plot([0, 1], [0, 1], 'k--')
ax_roc.set_title('ROC Curve')
ax_roc.set_xlabel('False Positive Rate')
ax_roc.set_ylabel('True Positive Rate')

ax_pr.set_title('Precision-Recall Curve')
ax_pr.set_xlabel('Recall')
ax_pr.set_ylabel('Precision')

plt.tight_layout()
plt.show()

# Calculating average metrics
average_scores = {metric: np.mean([score[metric] for score in scores]) for metric i
print("Average Metrics for KNN:", average_scores)

# Display cumulative confusion matrix
plt.figure(figsize=(8, 6))
sns.heatmap(cumulative_conf_matrix, annot=True, fmt="d", cmap='Blues')
plt.title("Cumulative Confusion Matrix - KNN")
plt.xlabel("Predicted labels")
plt.ylabel("True labels")
plt.show()
```

```
C:\Users\Ifean\anaconda3\Lib\site-packages\sklearn\base.py:474: FutureWarning: `Base
Estimator._validate_data` is deprecated in 1.6 and will be removed in 1.7. Use `skle
arn.utils.validation.validate_data` instead. This function becomes public and is par
t of the scikit-learn developer API.
  warnings.warn(
C:\Users\Ifean\anaconda3\Lib\site-packages\sklearn\base.py:474: FutureWarning: `Base
Estimator._validate_data` is deprecated in 1.6 and will be removed in 1.7. Use `skle
arn.utils.validation.validate_data` instead. This function becomes public and is par
t of the scikit-learn developer API.
  warnings.warn(
C:\Users\Ifean\anaconda3\Lib\site-packages\sklearn\base.py:474: FutureWarning: `Base
Estimator._validate_data` is deprecated in 1.6 and will be removed in 1.7. Use `skle
arn.utils.validation.validate_data` instead. This function becomes public and is par
t of the scikit-learn developer API.
  warnings.warn(
C:\Users\Ifean\anaconda3\Lib\site-packages\sklearn\base.py:474: FutureWarning: `Base
Estimator._validate_data` is deprecated in 1.6 and will be removed in 1.7. Use `skle
arn.utils.validation.validate_data` instead. This function becomes public and is par
t of the scikit-learn developer API.
  warnings.warn(
C:\Users\Ifean\anaconda3\Lib\site-packages\sklearn\base.py:474: FutureWarning: `Base
Estimator._validate_data` is deprecated in 1.6 and will be removed in 1.7. Use `skle
arn.utils.validation.validate_data` instead. This function becomes public and is par
t of the scikit-learn developer API.
  warnings.warn(
C:\Users\Ifean\anaconda3\Lib\site-packages\sklearn\base.py:474: FutureWarning: `Base
Estimator._validate_data` is deprecated in 1.6 and will be removed in 1.7. Use `skle
arn.utils.validation.validate_data` instead. This function becomes public and is par
t of the scikit-learn developer API.
  warnings.warn(
C:\Users\Ifean\anaconda3\Lib\site-packages\sklearn\base.py:474: FutureWarning: `Base
Estimator._validate_data` is deprecated in 1.6 and will be removed in 1.7. Use `skle
arn.utils.validation.validate_data` instead. This function becomes public and is par
t of the scikit-learn developer API.
  warnings.warn(
C:\Users\Ifean\anaconda3\Lib\site-packages\sklearn\base.py:474: FutureWarning: `Base
Estimator._validate_data` is deprecated in 1.6 and will be removed in 1.7. Use `skle
arn.utils.validation.validate_data` instead. This function becomes public and is par
t of the scikit-learn developer API.
  warnings.warn(
C:\Users\Ifean\anaconda3\Lib\site-packages\sklearn\base.py:474: FutureWarning: `Base
Estimator._validate_data` is deprecated in 1.6 and will be removed in 1.7. Use `skle
arn.utils.validation.validate_data` instead. This function becomes public and is par
t of the scikit-learn developer API.
  warnings.warn(
C:\Users\Ifean\anaconda3\Lib\site-packages\sklearn\base.py:474: FutureWarning: `Base
Estimator._validate_data` is deprecated in 1.6 and will be removed in 1.7. Use `skle
arn.utils.validation.validate_data` instead. This function becomes public and is par
t of the scikit-learn developer API.
  warnings.warn(
```



```
Average Metrics for KNN: {'roc_auc': 0.8605726174483929, 'f1': 0.244921399009033, 'a
ccuracy': 0.9840656845495177, 'recall': 0.6684652522779818, 'precision': 0.149976740
356965}
```

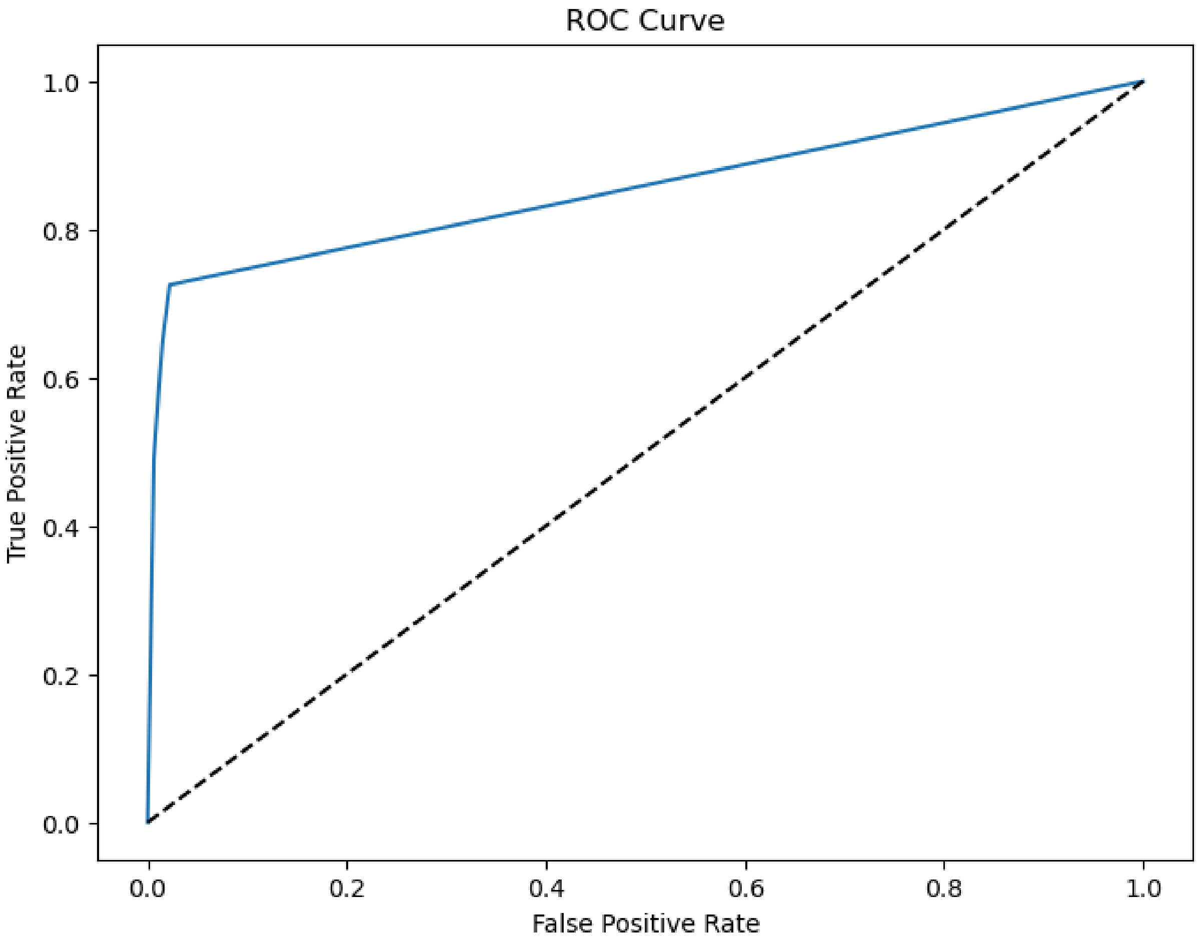


In [58]:

```
# predictions/probabilities) from model
y_probs = pipeline.predict_proba(X_test)[:, 1]  # Probabilities for the positive cl

# Calculate ROC curve
fpr, tpr, thresholds = roc_curve(y_test, y_probs)
```

```
# Plot ROC curve
plt.figure(figsize=(8, 6))
plt.plot(fpr, tpr, label='ROC Curve')
plt.plot([0, 1], [0, 1], 'k--')
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('ROC Curve')
```

Out[58]:
```
Text(0.5, 1.0, 'ROC Curve')
```



In [1]:
```
!dir
```

```
 Volume in drive C has no label.
 Volume Serial Number is FA70-33D1

 Directory of C:\Users\Ifean

03/26/2025  11:51 AM    <DIR>          .
10/29/2024  03:42 AM    <DIR>          ..
10/06/2024  01:13 AM    <DIR>          .anaconda
02/09/2025  11:42 PM    <DIR>          .azuredatastudio
03/24/2025  06:41 PM    <DIR>          .conda
10/06/2024  12:53 AM                25 .condarc
10/06/2024  12:49 AM    <DIR>          .continuum
03/21/2025  11:24 AM                53 .git-for-windows-updater
03/20/2025  06:38 PM               208 .gitconfig
03/26/2025  11:47 AM    <DIR>          .ipynb_checkpoints
10/06/2024  01:13 AM    <DIR>          .ipython
03/20/2025  01:12 PM    <DIR>          .jupyter
10/06/2024  01:23 AM    <DIR>          .matplotlib
03/25/2025  01:03 AM    <DIR>          .virtual_documents
12/15/2024  11:19 PM    <DIR>          .vscode
10/05/2024  10:48 PM    <DIR>          3D Objects
12/15/2024  11:08 PM    <DIR>          anaconda3
10/06/2024  03:57 PM    <DIR>          Contacts
03/20/2025  07:37 PM    <DIR>          Credit-Card-Fraud-Detection
03/25/2025  01:03 AM         1,547,779 Credit-Card-Fraud-Detection.ipynb
03/26/2025  11:51 AM           529,510 Credit_Card_Fraud_detection .ipynb
02/28/2025  06:12 PM    <DIR>          Documents
03/26/2025  11:08 AM    <DIR>          Downloads
03/20/2025  05:22 PM            41,560 Evaluating Machine Learning Methods in Credit
Card Fraud Detection.ipynb
10/06/2024  03:57 PM    <DIR>          Favorites
03/20/2025  01:27 PM       143,639,688 fraud test.csv
03/20/2025  06:36 PM       172,927,448 GitHubDesktopSetup-x64.exe
10/06/2024  03:57 PM    <DIR>          Links
02/07/2025  02:00 AM    <DIR>          Microsoft
10/06/2024  03:57 PM    <DIR>          Music
03/24/2025  05:30 PM           815,733 NNAMDI OKEKE PROJECT CODE.ipynb
03/20/2025  01:14 PM           310,275 NWwaterdata.csv
03/22/2025  12:24 AM    <DIR>          Octoparse
03/22/2025  12:23 AM        85,506,776 Octoparse Setup 8.7.4.exe
03/26/2025  08:56 AM    <DIR>          OneDrive
03/22/2025  12:40 AM       516,228,608 Orange3-3.38.1-Miniforge-x86_64.exe
10/05/2024  11:46 PM    <DIR>          Roaming
10/06/2024  03:57 PM    <DIR>          Saved Games
10/06/2024  03:57 PM    <DIR>          Searches
10/06/2024  01:14 AM            34,660 simulated_water_distribution_dataset.csv
02/09/2025  09:27 PM         4,290,992 SQL2022-SSEI-Dev.exe
02/09/2025  09:43 PM       496,190,328 SSMS-Setup-ENU.exe
10/07/2024  02:41 AM           318,838 synthetic_water_data.csv
02/23/2025  03:14 AM       536,136,512 TableauPublicDesktop-64bit-2024-3-0.exe
12/27/2024  01:44 AM             5,915 Training.ipynb
10/07/2024  03:47 AM           584,692 Untitled.ipynb
10/06/2024  01:25 AM           388,915 Untitled1.ipynb
10/07/2024  12:14 AM             2,691 Untitled2.ipynb
10/07/2024  12:28 AM           596,577 Untitled3.ipynb
10/07/2024  02:40 AM           738,292 Untitled4.ipynb
```

```
03/20/2025  01:14 PM         1,386,579 Untitled5.ipynb
10/10/2024  10:29 PM                72 Untitled6.ipynb
01/26/2025  04:52 AM             3,083 Untitled7.ipynb
10/11/2024  08:30 AM           880,187 UU Interview Script.ipynb
10/11/2024  08:30 AM           880,187 UU Interview.ipynb
01/20/2025  09:31 PM    <DIR>          Videos
01/26/2025  04:50 AM            41,516 water_treatment_data.csv
              29 File(s)  1,964,027,699 bytes
              28 Dir(s)  373,852,708,864 bytes free
```

In [3]:

```
!jupyter nbconvert "Credit-Card-Fraud-Detection.ipynb" --to webpdf
```

```
[NbConvertApp] Converting notebook Credit-Card-Fraud-Detection.ipynb to webpdf
[NbConvertApp] WARNING | Alternative text is missing on 29 image(s).
[NbConvertApp] Building PDF
Traceback (most recent call last):
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\exporters\webpdf.py", l
ine 78, in main
    from playwright.async_api import async_playwright  # type: ignore[import-not-fou
nd]
    ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
ModuleNotFoundError: No module named 'playwright'

The above exception was the direct cause of the following exception:

Traceback (most recent call last):
  File "C:\Users\Ifean\anaconda3\Scripts\jupyter-nbconvert-script.py", line 10, in <
module>
    sys.exit(main())
             ^^^^^^
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\jupyter_core\application.py", lin
e 283, in launch_instance
    super().launch_instance(argv=argv, **kwargs)
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\traitlets\config\application.py",
line 1075, in launch_instance
    app.start()
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\nbconvertapp.py", line
420, in start
    self.convert_notebooks()
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\nbconvertapp.py", line
597, in convert_notebooks
    self.convert_single_notebook(notebook_filename)
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\nbconvertapp.py", line
563, in convert_single_notebook
    output, resources = self.export_single_notebook(
                        ^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\nbconvertapp.py", line
487, in export_single_notebook
    output, resources = self.exporter.from_filename(
                        ^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\exporters\templateexpor
ter.py", line 386, in from_filename
    return super().from_filename(filename, resources, **kw)  # type:ignore[return-va
lue]
           ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\exporters\exporter.py",
line 201, in from_filename
    return self.from_file(f, resources=resources, **kw)
           ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\exporters\templateexpor
ter.py", line 392, in from_file
    return super().from_file(file_stream, resources, **kw)  # type:ignore[return-val
ue]
           ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\exporters\exporter.py",
line 220, in from_file
    return self.from_notebook_node(
           ^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\exporters\webpdf.py", l
ine 174, in from_notebook_node
    pdf_data = self.run_playwright(html)
               ^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\exporters\webpdf.py", l
ine 163, in run_playwright
    pdf_data = pool.submit(run_coroutine, main(temp_file)).result()
               ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\Lib\concurrent\futures\_base.py", line 456, in resu
lt
    return self.__get_result()
           ^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\Lib\concurrent\futures\_base.py", line 401, in __ge
t_result
    raise self._exception
  File "C:\Users\Ifean\anaconda3\Lib\concurrent\futures\thread.py", line 58, in run
    result = self.fn(*self.args, **self.kwargs)
             ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\exporters\webpdf.py", l
ine 161, in run_coroutine
    return loop.run_until_complete(coro)
           ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\Lib\asyncio\base_events.py", line 687, in run_until
_complete
    return future.result()
           ^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\exporters\webpdf.py", l
ine 84, in main
    raise RuntimeError(msg) from e
RuntimeError: Playwright is not installed to support Web PDF conversion. Please inst
all `nbconvert[webpdf]` to enable.
```

In [5]:

```
!jupyter nbconvert "Credit-Card-Fraud-Detection.ipynb" --to pdf
```

```
[NbConvertApp] Converting notebook Credit-Card-Fraud-Detection.ipynb to pdf
[NbConvertApp] ERROR | Error while converting 'Credit-Card-Fraud-Detection.ipynb'
Traceback (most recent call last):
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\nbconvertapp.py", line
487, in export_single_notebook
    output, resources = self.exporter.from_filename(
                        ^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\exporters\templateexpor
ter.py", line 386, in from_filename
    return super().from_filename(filename, resources, **kw)  # type:ignore[return-va
lue]
           ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\exporters\exporter.py",
line 201, in from_filename
    return self.from_file(f, resources=resources, **kw)
           ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\exporters\templateexpor
ter.py", line 392, in from_file
    return super().from_file(file_stream, resources, **kw)  # type:ignore[return-val
ue]
           ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\exporters\exporter.py",
line 220, in from_file
    return self.from_notebook_node(
           ^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\exporters\pdf.py", line
184, in from_notebook_node
    latex, resources = super().from_notebook_node(nb, resources=resources, **kw)
                       ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\exporters\latex.py", li
ne 92, in from_notebook_node
    return super().from_notebook_node(nb, resources, **kw)
           ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\exporters\templateexpor
ter.py", line 424, in from_notebook_node
    output = self.template.render(nb=nb_copy, resources=resources)
             ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\jinja2\environment.py", line 130
4, in render
    self.environment.handle_exception()
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\jinja2\environment.py", line 939,
in handle_exception
    raise rewrite_traceback_stack(source=source)
  File "C:\Users\Ifean\anaconda3\share\jupyter\nbconvert\templates\latex\index.tex.j
2", line 8, in top-level template code
    ((* extends cell_style *))
    ^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\share\jupyter\nbconvert\templates\latex\style_jupyt
er.tex.j2", line 176, in top-level template code
    \prompt{(((prompt)))}{(((prompt_color)))}{(((execution_count)))}{(((extra_spac
e)))}
    ^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\share\jupyter\nbconvert\templates\latex\base.tex.j
2", line 7, in top-level template code
    ((*- extends 'document_contents.tex.j2' -*))
    ^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\share\jupyter\nbconvert\templates\latex\document_co
ntents.tex.j2", line 51, in top-level template code
    ((*- block figure scoped -*))
    ^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\share\jupyter\nbconvert\templates\latex\display_pri
ority.j2", line 5, in top-level template code
    ((*- extends 'null.j2' -*))
    ^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\share\jupyter\nbconvert\templates\latex\null.j2", l
ine 30, in top-level template code
    ((*- block body -*))
  File "C:\Users\Ifean\anaconda3\share\jupyter\nbconvert\templates\latex\base.tex.j
2", line 222, in block 'body'
    ((( super() )))
  File "C:\Users\Ifean\anaconda3\share\jupyter\nbconvert\templates\latex\null.j2", l
ine 32, in block 'body'
    ((*- block any_cell scoped -*))
    ^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\share\jupyter\nbconvert\templates\latex\null.j2", l
ine 85, in block 'any_cell'
    ((*- block markdowncell scoped-*)) ((*- endblock markdowncell -*))
    ^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\share\jupyter\nbconvert\templates\latex\document_co
ntents.tex.j2", line 68, in block 'markdowncell'
    ((( cell.source | citation2latex | strip_files_prefix | convert_pandoc('markdown
+tex_math_double_backslash', 'json',extra_args=[]) | resolve_references | convert_ex
plicitly_relative_paths | convert_pandoc('json','latex'))))
    ^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\filters\pandoc.py", lin
e 36, in convert_pandoc
    return pandoc(source, from_format, to_format, extra_args=extra_args)
           ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\utils\pandoc.py", line
50, in pandoc
    check_pandoc_version()
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\utils\pandoc.py", line
98, in check_pandoc_version
    v = get_pandoc_version()
        ^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\utils\pandoc.py", line
75, in get_pandoc_version
    raise PandocMissing()
nbconvert.utils.pandoc.PandocMissing: Pandoc wasn't found.
Please check that pandoc is installed:
https://pandoc.org/installing.html
```

In [7]:

```
!jupyter nbconvert "Credit-Card-Fraud-Detection.ipynb" --to pdf
```

```
[NbConvertApp] Converting notebook Credit-Card-Fraud-Detection.ipynb to pdf
[NbConvertApp] ERROR | Error while converting 'Credit-Card-Fraud-Detection.ipynb'
Traceback (most recent call last):
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\nbconvertapp.py", line
487, in export_single_notebook
    output, resources = self.exporter.from_filename(
                        ^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\exporters\templateexpor
ter.py", line 386, in from_filename
    return super().from_filename(filename, resources, **kw)  # type:ignore[return-va
lue]
           ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\exporters\exporter.py",
line 201, in from_filename
    return self.from_file(f, resources=resources, **kw)
           ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\exporters\templateexpor
ter.py", line 392, in from_file
    return super().from_file(file_stream, resources, **kw)  # type:ignore[return-val
ue]
           ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\exporters\exporter.py",
line 220, in from_file
    return self.from_notebook_node(
           ^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\exporters\pdf.py", line
184, in from_notebook_node
    latex, resources = super().from_notebook_node(nb, resources=resources, **kw)
                       ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\exporters\latex.py", li
ne 92, in from_notebook_node
    return super().from_notebook_node(nb, resources, **kw)
           ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\exporters\templateexpor
ter.py", line 424, in from_notebook_node
    output = self.template.render(nb=nb_copy, resources=resources)
             ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\jinja2\environment.py", line 130
4, in render
    self.environment.handle_exception()
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\jinja2\environment.py", line 939,
in handle_exception
    raise rewrite_traceback_stack(source=source)
  File "C:\Users\Ifean\anaconda3\share\jupyter\nbconvert\templates\latex\index.tex.j
2", line 8, in top-level template code
    ((* extends cell_style *))
    ^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\share\jupyter\nbconvert\templates\latex\style_jupyt
er.tex.j2", line 176, in top-level template code
    \prompt{(((prompt)))}{(((prompt_color)))}{(((execution_count)))}{(((extra_spac
e)))}
    ^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\share\jupyter\nbconvert\templates\latex\base.tex.j
2", line 7, in top-level template code
    ((*- extends 'document_contents.tex.j2' -*))
    ^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\share\jupyter\nbconvert\templates\latex\document_co
ntents.tex.j2", line 51, in top-level template code
    ((*- block figure scoped -*))
    ^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\share\jupyter\nbconvert\templates\latex\display_pri
ority.j2", line 5, in top-level template code
    ((*- extends 'null.j2' -*))
    ^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\share\jupyter\nbconvert\templates\latex\null.j2", l
ine 30, in top-level template code
    ((*- block body -*))
  File "C:\Users\Ifean\anaconda3\share\jupyter\nbconvert\templates\latex\base.tex.j
2", line 222, in block 'body'
    ((( super() )))
  File "C:\Users\Ifean\anaconda3\share\jupyter\nbconvert\templates\latex\null.j2", l
ine 32, in block 'body'
    ((*- block any_cell scoped -*))
    ^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\share\jupyter\nbconvert\templates\latex\null.j2", l
ine 85, in block 'any_cell'
    ((*- block markdowncell scoped-*)) ((*- endblock markdowncell -*))
    ^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\share\jupyter\nbconvert\templates\latex\document_co
ntents.tex.j2", line 68, in block 'markdowncell'
    ((( cell.source | citation2latex | strip_files_prefix | convert_pandoc('markdown
+tex_math_double_backslash', 'json',extra_args=[]) | resolve_references | convert_ex
plicitly_relative_paths | convert_pandoc('json','latex'))))
    ^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\filters\pandoc.py", lin
e 36, in convert_pandoc
    return pandoc(source, from_format, to_format, extra_args=extra_args)
           ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\utils\pandoc.py", line
50, in pandoc
    check_pandoc_version()
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\utils\pandoc.py", line
98, in check_pandoc_version
    v = get_pandoc_version()
        ^^^^^^^^^^^^^^^^^^^^
  File "C:\Users\Ifean\anaconda3\Lib\site-packages\nbconvert\utils\pandoc.py", line
75, in get_pandoc_version
    raise PandocMissing()
nbconvert.utils.pandoc.PandocMissing: Pandoc wasn't found.
Please check that pandoc is installed:
https://pandoc.org/installing.html
```

```
In [9]: !pip install nbconvert[webpdf]
```

```
Requirement already satisfied: nbconvert[webpdf] in c:\users\ifean\anaconda3\lib\sit
e-packages (7.16.4)
Requirement already satisfied: beautifulsoup4 in c:\users\ifean\anaconda3\lib\site-p
ackages (from nbconvert[webpdf]) (4.12.3)
Requirement already satisfied: bleach!=5.0.0 in c:\users\ifean\anaconda3\lib\site-pa
ckages (from nbconvert[webpdf]) (4.1.0)
Requirement already satisfied: defusedxml in c:\users\ifean\anaconda3\lib\site-packa
ges (from nbconvert[webpdf]) (0.7.1)
Requirement already satisfied: jinja2>=3.0 in c:\users\ifean\anaconda3\lib\site-pack
ages (from nbconvert[webpdf]) (3.1.4)
Requirement already satisfied: jupyter-core>=4.7 in c:\users\ifean\anaconda3\lib\sit
e-packages (from nbconvert[webpdf]) (5.7.2)
Requirement already satisfied: jupyterlab-pygments in c:\users\ifean\anaconda3\lib\s
ite-packages (from nbconvert[webpdf]) (0.1.2)
Requirement already satisfied: markupsafe>=2.0 in c:\users\ifean\anaconda3\lib\site-
packages (from nbconvert[webpdf]) (2.1.3)
Requirement already satisfied: mistune<4,>=2.0.3 in c:\users\ifean\anaconda3\lib\sit
e-packages (from nbconvert[webpdf]) (2.0.4)
Requirement already satisfied: nbclient>=0.5.0 in c:\users\ifean\anaconda3\lib\site-
packages (from nbconvert[webpdf]) (0.8.0)
Requirement already satisfied: nbformat>=5.7 in c:\users\ifean\anaconda3\lib\site-pa
ckages (from nbconvert[webpdf]) (5.10.4)
Requirement already satisfied: packaging in c:\users\ifean\anaconda3\lib\site-packag
es (from nbconvert[webpdf]) (24.1)
Requirement already satisfied: pandocfilters>=1.4.1 in c:\users\ifean\anaconda3\lib
\site-packages (from nbconvert[webpdf]) (1.5.0)
Requirement already satisfied: pygments>=2.4.1 in c:\users\ifean\anaconda3\lib\site-
packages (from nbconvert[webpdf]) (2.15.1)
Requirement already satisfied: tinycss2 in c:\users\ifean\anaconda3\lib\site-package
s (from nbconvert[webpdf]) (1.2.1)
Requirement already satisfied: traitlets>=5.1 in c:\users\ifean\anaconda3\lib\site-p
ackages (from nbconvert[webpdf]) (5.14.3)
Collecting playwright (from nbconvert[webpdf])
  Downloading playwright-1.51.0-py3-none-win_amd64.whl.metadata (3.5 kB)
Requirement already satisfied: six>=1.9.0 in c:\users\ifean\anaconda3\lib\site-packa
ges (from bleach!=5.0.0->nbconvert[webpdf]) (1.16.0)
Requirement already satisfied: webencodings in c:\users\ifean\anaconda3\lib\site-pac
kages (from bleach!=5.0.0->nbconvert[webpdf]) (0.5.1)
Requirement already satisfied: platformdirs>=2.5 in c:\users\ifean\anaconda3\lib\sit
e-packages (from jupyter-core>=4.7->nbconvert[webpdf]) (3.10.0)
Requirement already satisfied: pywin32>=300 in c:\users\ifean\anaconda3\lib\site-pac
kages (from jupyter-core>=4.7->nbconvert[webpdf]) (305.1)
Requirement already satisfied: jupyter-client>=6.1.12 in c:\users\ifean\anaconda3\li
b\site-packages (from nbclient>=0.5.0->nbconvert[webpdf]) (8.6.0)
Requirement already satisfied: fastjsonschema>=2.15 in c:\users\ifean\anaconda3\lib
\site-packages (from nbformat>=5.7->nbconvert[webpdf]) (2.16.2)
Requirement already satisfied: jsonschema>=2.6 in c:\users\ifean\anaconda3\lib\site-
packages (from nbformat>=5.7->nbconvert[webpdf]) (4.19.2)
Requirement already satisfied: soupsieve>1.2 in c:\users\ifean\anaconda3\lib\site-pa
ckages (from beautifulsoup4->nbconvert[webpdf]) (2.5)
Collecting pyee<13,>=12 (from playwright->nbconvert[webpdf])
  Downloading pyee-12.1.1-py3-none-any.whl.metadata (2.9 kB)
Collecting greenlet<4.0.0,>=3.1.1 (from playwright->nbconvert[webpdf])
  Downloading greenlet-3.1.1-cp312-cp312-win_amd64.whl.metadata (3.9 kB)
Requirement already satisfied: attrs>=22.2.0 in c:\users\ifean\anaconda3\lib\site-pa
ckages (from jsonschema>=2.6->nbformat>=5.7->nbconvert[webpdf]) (23.1.0)
```

```
Requirement already satisfied: jsonschema-specifications>=2023.03.6 in c:\users\ifea
n\anaconda3\lib\site-packages (from jsonschema>=2.6->nbformat>=5.7->nbconvert[webpd
f]) (2023.7.1)
Requirement already satisfied: referencing>=0.28.4 in c:\users\ifean\anaconda3\lib\s
ite-packages (from jsonschema>=2.6->nbformat>=5.7->nbconvert[webpdf]) (0.30.2)
Requirement already satisfied: rpds-py>=0.7.1 in c:\users\ifean\anaconda3\lib\site-p
ackages (from jsonschema>=2.6->nbformat>=5.7->nbconvert[webpdf]) (0.10.6)
Requirement already satisfied: python-dateutil>=2.8.2 in c:\users\ifean\anaconda3\li
b\site-packages (from jupyter-client>=6.1.12->nbclient>=0.5.0->nbconvert[webpdf])
(2.9.0.post0)
Requirement already satisfied: pyzmq>=23.0 in c:\users\ifean\anaconda3\lib\site-pack
ages (from jupyter-client>=6.1.12->nbclient>=0.5.0->nbconvert[webpdf]) (25.1.2)
Requirement already satisfied: tornado>=6.2 in c:\users\ifean\anaconda3\lib\site-pac
kages (from jupyter-client>=6.1.12->nbclient>=0.5.0->nbconvert[webpdf]) (6.4.1)
Requirement already satisfied: typing-extensions in c:\users\ifean\anaconda3\lib\sit
e-packages (from pyee<13,>=12->playwright->nbconvert[webpdf]) (4.11.0)
Downloading playwright-1.51.0-py3-none-win_amd64.whl (34.9 MB)
   ---------------------------------------- 0.0/34.9 MB ? eta -:--:--
   ---------------------------------------- 0.3/34.9 MB ? eta -:--:--
    --------------------------------------- 0.8/34.9 MB 2.2 MB/s eta 0:00:16
   - -------------------------------------- 1.3/34.9 MB 2.3 MB/s eta 0:00:15
   -- ------------------------------------- 1.8/34.9 MB 2.2 MB/s eta 0:00:15
   -- ------------------------------------- 2.4/34.9 MB 2.3 MB/s eta 0:00:15
   --- ------------------------------------ 2.9/34.9 MB 2.3 MB/s eta 0:00:15
   --- ------------------------------------ 3.4/34.9 MB 2.3 MB/s eta 0:00:14
   ---- ----------------------------------- 3.9/34.9 MB 2.3 MB/s eta 0:00:14
   ---- ----------------------------------- 4.2/34.9 MB 2.3 MB/s eta 0:00:14
   ----- ---------------------------------- 4.7/34.9 MB 2.3 MB/s eta 0:00:14
   ------ --------------------------------- 5.2/34.9 MB 2.3 MB/s eta 0:00:14
   ------ --------------------------------- 5.8/34.9 MB 2.3 MB/s eta 0:00:13
   ------- -------------------------------- 6.3/34.9 MB 2.3 MB/s eta 0:00:13
   ------- -------------------------------- 6.8/34.9 MB 2.3 MB/s eta 0:00:13
   -------- ------------------------------- 7.3/34.9 MB 2.3 MB/s eta 0:00:13
   --------- ------------------------------ 7.9/34.9 MB 2.3 MB/s eta 0:00:12
   --------- ------------------------------ 8.4/34.9 MB 2.3 MB/s eta 0:00:12
   --------- ------------------------------ 8.7/34.9 MB 2.3 MB/s eta 0:00:12
   ---------- ----------------------------- 9.2/34.9 MB 2.3 MB/s eta 0:00:12
   ----------- ---------------------------- 9.7/34.9 MB 2.3 MB/s eta 0:00:12
   ----------- ---------------------------- 10.2/34.9 MB 2.3 MB/s eta 0:00:11
   ------------ --------------------------- 10.7/34.9 MB 2.3 MB/s eta 0:00:11
   ------------ --------------------------- 11.3/34.9 MB 2.3 MB/s eta 0:00:11
   ------------- -------------------------- 11.8/34.9 MB 2.3 MB/s eta 0:00:11
   -------------- ------------------------- 12.3/34.9 MB 2.3 MB/s eta 0:00:10
   -------------- ------------------------- 12.8/34.9 MB 2.3 MB/s eta 0:00:10
   --------------- ------------------------ 13.4/34.9 MB 2.3 MB/s eta 0:00:10
   --------------- ------------------------ 13.9/34.9 MB 2.3 MB/s eta 0:00:10
   ---------------- ----------------------- 14.4/34.9 MB 2.3 MB/s eta 0:00:09
   ----------------- ---------------------- 14.9/34.9 MB 2.3 MB/s eta 0:00:09
   ----------------- ---------------------- 15.5/34.9 MB 2.3 MB/s eta 0:00:09
   ------------------ --------------------- 16.0/34.9 MB 2.3 MB/s eta 0:00:09
   ------------------ --------------------- 16.3/34.9 MB 2.3 MB/s eta 0:00:09
   ------------------- -------------------- 16.8/34.9 MB 2.3 MB/s eta 0:00:08
   ------------------- -------------------- 17.3/34.9 MB 2.3 MB/s eta 0:00:08
   -------------------- ------------------- 17.8/34.9 MB 2.3 MB/s eta 0:00:08
   --------------------- ------------------ 18.4/34.9 MB 2.3 MB/s eta 0:00:08
   --------------------- ------------------ 18.9/34.9 MB 2.3 MB/s eta 0:00:08
   ---------------------- ----------------- 19.4/34.9 MB 2.3 MB/s eta 0:00:07
   ---------------------- ----------------- 19.9/34.9 MB 2.3 MB/s eta 0:00:07
   ----------------------- ---------------- 20.4/34.9 MB 2.3 MB/s eta 0:00:07
   ----------------------- ---------------- 20.7/34.9 MB 2.3 MB/s eta 0:00:07
   ------------------------ --------------- 21.2/34.9 MB 2.3 MB/s eta 0:00:06
   ------------------------ --------------- 21.8/34.9 MB 2.3 MB/s eta 0:00:06
   ------------------------- -------------- 22.3/34.9 MB 2.3 MB/s eta 0:00:06
   -------------------------- ------------- 22.8/34.9 MB 2.3 MB/s eta 0:00:06
   -------------------------- ------------- 23.3/34.9 MB 2.3 MB/s eta 0:00:06
   --------------------------- ------------ 23.9/34.9 MB 2.3 MB/s eta 0:00:05
   --------------------------- ------------ 24.4/34.9 MB 2.3 MB/s eta 0:00:05
   ---------------------------- ----------- 24.9/34.9 MB 2.3 MB/s eta 0:00:05
   ----------------------------- ---------- 25.4/34.9 MB 2.3 MB/s eta 0:00:05
   ----------------------------- ---------- 26.0/34.9 MB 2.3 MB/s eta 0:00:04
   ------------------------------ --------- 26.5/34.9 MB 2.3 MB/s eta 0:00:04
   ------------------------------ --------- 27.0/34.9 MB 2.3 MB/s eta 0:00:04
   ------------------------------- -------- 27.5/34.9 MB 2.3 MB/s eta 0:00:04
   ------------------------------- -------- 27.8/34.9 MB 2.3 MB/s eta 0:00:04
   -------------------------------- ------- 28.3/34.9 MB 2.3 MB/s eta 0:00:03
   --------------------------------- ------ 28.8/34.9 MB 2.3 MB/s eta 0:00:03
   --------------------------------- ------ 29.4/34.9 MB 2.3 MB/s eta 0:00:03
   ---------------------------------- ----- 29.9/34.9 MB 2.3 MB/s eta 0:00:03
   ---------------------------------- ----- 30.4/34.9 MB 2.3 MB/s eta 0:00:02
   ----------------------------------- ---- 30.9/34.9 MB 2.3 MB/s eta 0:00:02
   ------------------------------------ --- 31.5/34.9 MB 2.3 MB/s eta 0:00:02
   ------------------------------------ --- 32.0/34.9 MB 2.3 MB/s eta 0:00:02
   ------------------------------------ --- 32.2/34.9 MB 2.3 MB/s eta 0:00:02
   ------------------------------------- -- 32.8/34.9 MB 2.3 MB/s eta 0:00:01
   -------------------------------------- - 33.3/34.9 MB 2.3 MB/s eta 0:00:01
   -------------------------------------- - 33.8/34.9 MB 2.3 MB/s eta 0:00:01
   ---------------------------------------  34.3/34.9 MB 2.3 MB/s eta 0:00:01
   ---------------------------------------  34.6/34.9 MB 2.3 MB/s eta 0:00:01
   ---------------------------------------- 34.9/34.9 MB 2.3 MB/s eta 0:00:00
Downloading greenlet-3.1.1-cp312-cp312-win_amd64.whl (299 kB)
Downloading pyee-12.1.1-py3-none-any.whl (15 kB)
Installing collected packages: pyee, greenlet, playwright
  Attempting uninstall: greenlet
    Found existing installation: greenlet 3.0.1
    Uninstalling greenlet-3.0.1:
      Successfully uninstalled greenlet-3.0.1
Successfully installed greenlet-3.1.1 playwright-1.51.0 pyee-12.1.1
```

In [11]:

```
!playwright install chromium
```

```
Downloading Chromium 134.0.6998.35 (playwright build v1161) from https://cdn.playwri
ght.dev/dbazure/download/playwright/builds/chromium/1161/chromium-win64.zip
|                                                                                |
0% of 141.8 MiB
|â- â- â- â- â- â- â- â-
|  10% of 141.8 MiB
|â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â-
|  20% of 141.8 MiB
|â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â-
|  30% of 141.8 MiB
|â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â-
 â- â- â- â-                                                 |  40% of 141.8 MiB
Downloading Chromium 134.0.6998.35 (playwright build v1161) from https://playwright.
download.prss.microsoft.com/dbazure/download/playwright/builds/chromium/1161/chromiu
m-win64.zip
|                                                                                |
0% of 141.8 MiB
|â- â- â- â- â- â- â- â-
|  10% of 141.8 MiB
|â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â-
|  20% of 141.8 MiB
|â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â-
|  30% of 141.8 MiB
|â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â-
 â- â- â- â-                                                 |  40% of 141.8 MiB
|â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â-
 â- â- â- â- â- â- â- â- â- â- â- â-                                         |  50%
of 141.8 MiB
|â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â-
 â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â-
|  60% of 141.8 MiB
|â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â-
 â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â-
                         |  70% of 141.8 MiB
|â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â-
 â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â-
 â- â- â- â- â- â- â- â-                 |  80% of 141.8 MiB
|â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â-
 â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â-
 â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â-         |  90% of 141.8 MiB
|â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â-
 â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â-
 â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- | 100% of 1
41.8 MiB
Chromium 134.0.6998.35 (playwright build v1161) downloaded to C:\Users\Ifean\AppData
\Local\ms-playwright\chromium-1161
Downloading FFMPEG playwright build v1011 from https://cdn.playwright.dev/dbazure/do
wnload/playwright/builds/ffmpeg/1011/ffmpeg-win64.zip
|                                                                                |
0% of 1.3 MiB
|â- â- â- â- â- â- â- â-
|  10% of 1.3 MiB
|â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â-
|  20% of 1.3 MiB
|â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â- â-
|  30% of 1.3 MiB
```

```
|â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
 â– â– â– â–                                                 |  40% of 1.3 MiB
|â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
 â– â– â– â– â– â– â– â– â– â– â– â–                                         |  51%
of 1.3 MiB
|â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
 â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
|  60% of 1.3 MiB
|â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
 â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
                         |  70% of 1.3 MiB
|â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
 â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
 â– â– â– â– â– â– â– â–                 |  80% of 1.3 MiB
|â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
 â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
 â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–         |  90% of 1.3 MiB
|â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
 â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
 â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– | 100% of
1.3 MiB
FFMPEG playwright build v1011 downloaded to C:\Users\Ifean\AppData\Local\ms-playwrig
ht\ffmpeg-1011
Downloading Chromium Headless Shell 134.0.6998.35 (playwright build v1161) from http
s://cdn.playwright.dev/dbazure/download/playwright/builds/chromium/1161/chromium-hea
dless-shell-win64.zip
|                                                                                |
0% of 87.8 MiB
|â– â– â– â– â– â– â– â–
|  10% of 87.8 MiB
|â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
|  20% of 87.8 MiB
|â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
|  30% of 87.8 MiB
|â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
 â– â– â– â–                                                 |  40% of 87.8 MiB
|â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
 â– â– â– â– â– â– â– â– â– â– â– â–                                         |  50%
of 87.8 MiB
|â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
 â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
|  60% of 87.8 MiB
|â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
 â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
                         |  70% of 87.8 MiB
Downloading Chromium Headless Shell 134.0.6998.35 (playwright build v1161) from http
s://playwright.download.prss.microsoft.com/dbazure/download/playwright/builds/chromi
um/1161/chromium-headless-shell-win64.zip
|                                                                                |
0% of 87.8 MiB
|â– â– â– â– â– â– â– â–
|  10% of 87.8 MiB
|â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
|  20% of 87.8 MiB
|â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
|  30% of 87.8 MiB
```

```
|â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
 â– â– â– â–                                                 |  40% of 87.8 MiB
|â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
 â– â– â– â– â– â– â– â– â– â– â– â–                                         |  50%
of 87.8 MiB
|â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
 â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
|  60% of 87.8 MiB
|â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
 â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
                         |  70% of 87.8 MiB
|â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
 â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
 â– â– â– â– â– â– â– â–                 |  80% of 87.8 MiB
|â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
 â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
 â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–         |  90% of 87.8 MiB
|â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
 â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
 â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– | 100% of 8
7.8 MiB
Chromium Headless Shell 134.0.6998.35 (playwright build v1161) downloaded to C:\User
s\Ifean\AppData\Local\ms-playwright\chromium_headless_shell-1161
Downloading Winldd playwright build v1007 from https://cdn.playwright.dev/dbazure/do
wnload/playwright/builds/winldd/1007/winldd-win64.zip
|â– â– â– â– â– â– â– â–
|  12% of 0.1 MiB
|â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
|  25% of 0.1 MiB
|â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
 â– â– â– â–                                                 |  40% of 0.1 MiB
|â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
 â– â– â– â– â– â– â– â– â– â– â– â–                                         |  52%
of 0.1 MiB
|â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
 â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
                         |  71% of 0.1 MiB
|â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
 â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
 â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–         |  90% of 0.1 MiB
|â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
 â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â–
 â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– â– | 100% of
0.1 MiB
Winldd playwright build v1007 downloaded to C:\Users\Ifean\AppData\Local\ms-playwrig
ht\winldd-1007
```

```
Error: Request to https://cdn.playwright.dev/dbazure/download/playwright/builds/chro
mium/1161/chromium-win64.zip timed out after 30000ms
    at ClientRequest.rejectOnTimeout (C:\Users\Ifean\anaconda3\Lib\site-packages\pla
ywright\driver\package\lib\server\utils\network.js:76:15)
    at Object.onceWrapper (node:events:632:28)
    at ClientRequest.emit (node:events:518:28)
    at TLSSocket.emitRequestTimeout (node:_http_client:863:9)
    at Object.onceWrapper (node:events:632:28)
    at TLSSocket.emit (node:events:530:35)
    at Socket._onTimeout (node:net:609:8)
    at listOnTimeout (node:internal/timers:594:17)
    at process.processTimers (node:internal/timers:529:7)
Error: Request to https://cdn.playwright.dev/dbazure/download/playwright/builds/chro
mium/1161/chromium-headless-shell-win64.zip timed out after 30000ms
    at ClientRequest.rejectOnTimeout (C:\Users\Ifean\anaconda3\Lib\site-packages\pla
ywright\driver\package\lib\server\utils\network.js:76:15)
    at Object.onceWrapper (node:events:632:28)
    at ClientRequest.emit (node:events:518:28)
    at TLSSocket.emitRequestTimeout (node:_http_client:863:9)
    at Object.onceWrapper (node:events:632:28)
    at TLSSocket.emit (node:events:530:35)
    at Socket._onTimeout (node:net:609:8)
    at listOnTimeout (node:internal/timers:594:17)
    at process.processTimers (node:internal/timers:529:7)
```

In [ ]:

```
!jupyter nbconvert "Credit-Card-Fraud-Detection.ipynb" --to webpdf
```

In [ ]:

```

```