

FRAUD DETECTION

Association Rules and Classification

กุหลาบไฟ

นายรัชชานนท์ พันภาพสินธุ์ 653020217-4
นางสาวธมน ศิริพราหมณ์กุล 653020603-9
นางสาวอริสรา ยังอยู่ 653020606-3



DATA DESCRIPTION

23 Columns
585,177 Rows

Name	Description	Data Type
TRANSACTION_ID	รหัสเฉพาะสำหรับแต่ละธุรกรรม (Unique Identifier)	Integer
TX_DATETIME	วันที่และเวลาที่เกิดธุรกรรม	Datetime
CUSTOMER_ID	รหัสประจำตัวลูกค้า	Integer
TERMINAL_ID	รหัสประจำตัวเครื่อง terminal	Integer
TX_AMOUNT	จำนวนเงินในการทำธุรกรรม	Float
TX_TIME_SECONDS	เวลาในการทำธุรกรรม(วินาที)	Integer
TX_TIME_DAYS	จำนวนวันในการทำธุรกรรม	Integer

DATA DESCRIPTION

Name	Description	Data Type
TX_FRAUD	ตัวบ่งชี้ว่าธุรกรรมนั้นเป็นการฉ้อโกงหรือไม่ (Non-Fraud, Fraud)	string
TX_FRAUD_SCENARIO	สถานการณ์การฉ้อโกง (0 = ไม่มีการฉ้อโกง (No Fraud), 1 = การฉ้อโกงจากจำนวนเงินสูงผิดปกติ, 2 = การฉ้อโกงผ่านเครื่องรับชำระเงิน (Terminal Fraud), 3 = การฉ้อโกงผ่านการขโมยข้อมูลบัตร (Card-not-present Fraud))	Integer
TX_DURING_WEEKEND	ตัวบ่งชี้ว่าธุรกรรมเกิดขึ้นในช่วงสุดสัปดาห์หรือไม่ (0 = ไม่ใช่, 1 = ใช่)	Integer
TX_DURING_NIGHT	ตัวบ่งชี้ว่าธุรกรรมเกิดขึ้นในช่วงกลางคืนหรือไม่ (0 = ไม่ใช่, 1 = ใช่)	Integer
CUSTOMER_ID_NB_TX_1DAY_WINDOW	จำนวนธุรกรรมที่ลูกค้าทำในช่วง 1 วันที่ผ่านมา	Float

DATA DESCRIPTION

Name	Description	Data Type
CUSTOMER_ID_AVG_AMOUNT_1DAY_WINDOW	ค่าเฉลี่ยจำนวนเงินของธุรกรรมที่ลูกค้าทำในช่วง 1 วันที่ผ่านมา	Float
CUSTOMER_ID_NB_TX_7DAY_WINDOW	จำนวนธุรกรรมที่ลูกค้าทำในช่วง 7 วันที่ผ่านมา	Float
CUSTOMER_ID_AVG_AMOUNT_7DAY_WINDOW	ค่าเฉลี่ยจำนวนเงินของธุรกรรมที่ลูกค้าทำในช่วง 7 วันที่ผ่านมา	Float
CUSTOMER_ID_NB_TX_30DAY_WINDOW	จำนวนธุรกรรมที่ลูกค้าทำในช่วง 30 วันที่ผ่านมา	Float
CUSTOMER_ID_AVG_AMOUNT_30DAY_WINDOW	ค่าเฉลี่ยจำนวนเงินของธุรกรรมที่ลูกค้าทำในช่วง 30 วันที่ผ่านมา	Float

DATA DESCRIPTION

Name	Description	Data Type
TERMINAL_ID_NB_TX_1DAY_WINDOW	จำนวนธุรกรรมที่เกิดขึ้นบน terminal ในช่วง 1 วันที่ผ่านมา (มี delay 7 วัน)	Float
TERMINAL_ID_RISK_1DAY_WINDOW	ค่าความเสี่ยงของ terminal ในช่วง 1 วันที่ผ่านมา (มี delay 7 วัน)	Float
TERMINAL_ID_NB_TX_7DAY_WINDOW	จำนวนธุรกรรมที่เกิดขึ้นบน terminal ในช่วง 7 วันที่ผ่านมา (มี delay 7 วัน)	Float
TERMINAL_ID_RISK_7DAY_WINDOW	ค่าความเสี่ยง (risk score) ของ terminal ในช่วง 7 วันที่ผ่านมา (มี delay 7 วัน)	Float
TERMINAL_ID_NB_TX_30DAY_WINDOW	จำนวนธุรกรรมที่เกิดขึ้นบน terminal ในช่วง 30 วันที่ผ่านมา (มี delay 7 วัน)	Float
TERMINAL_ID_RISK_30DAY_WINDOW	ค่าความเสี่ยง ของ terminal ในช่วง 30 วันที่ผ่านมา (มี delay 7 วัน)	Float



DATA PREPROCESSING

ตรวจสอบและจัดการค่า missing

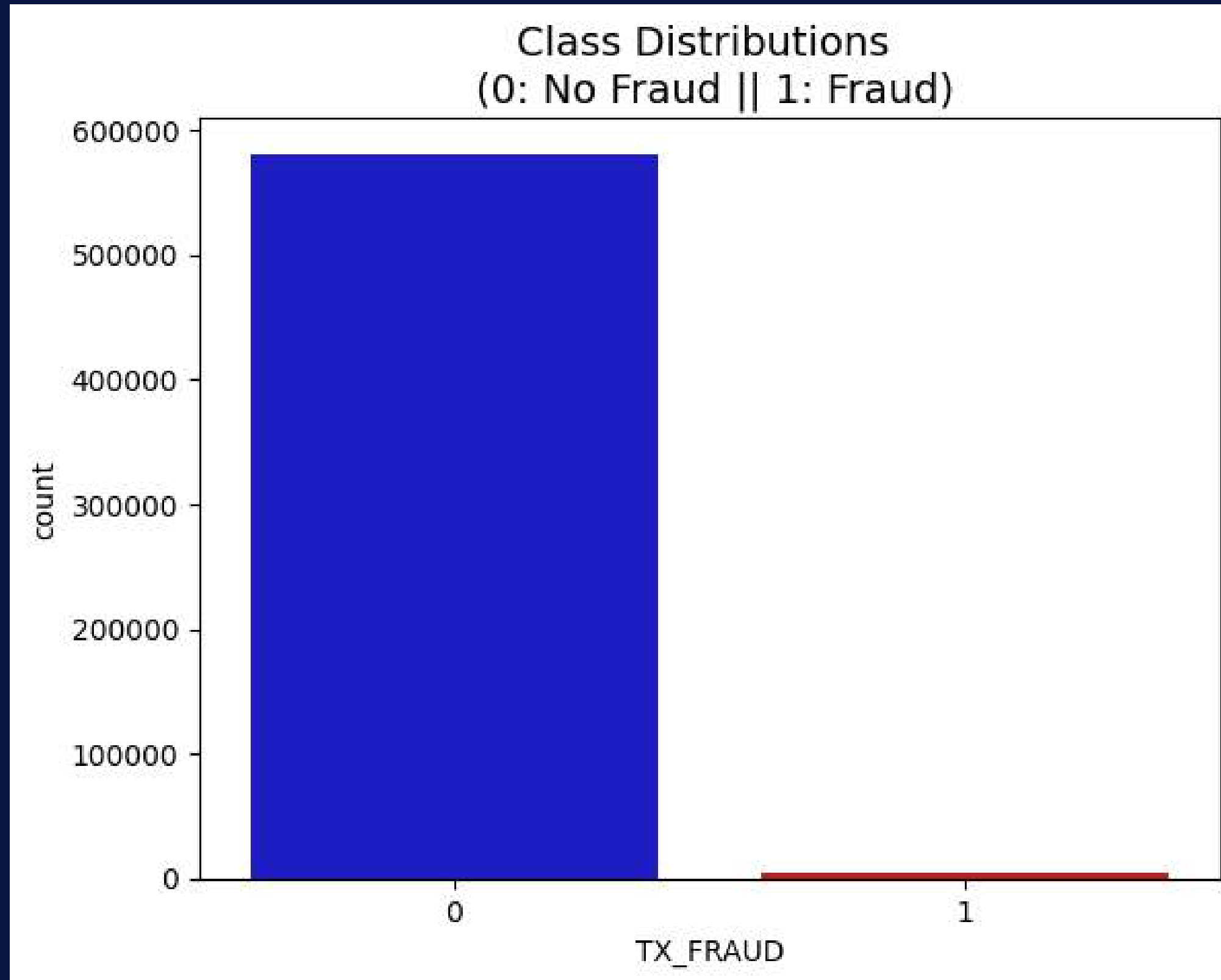
23 Columns
585,177 Rows

TRANSACTION_ID	0
TX_DATETIME	0
CUSTOMER_ID	0
TERMINAL_ID	0
TX_AMOUNT	0
TX_TIME_SECONDS	0
TX_TIME_DAYS	0
TX_FRAUD	0
TX_FRAUD_SCENARIO	0
TX_DURING_WEEKEND	0
TX_DURING_NIGHT	0

CUSTOMER_ID_NB_TX_1DAY_WINDOW	0
CUSTOMER_ID_AVG_AMOUNT_1DAY_WINDOW	0
CUSTOMER_ID_NB_TX_7DAY_WINDOW	0
CUSTOMER_ID_AVG_AMOUNT_7DAY_WINDOW	0
CUSTOMER_ID_NB_TX_30DAY_WINDOW	0
CUSTOMER_ID_AVG_AMOUNT_30DAY_WINDOW	0
TERMINAL_ID_NB_TX_1DAY_WINDOW	0
TERMINAL_ID_RISK_1DAY_WINDOW	0
TERMINAL_ID_NB_TX_7DAY_WINDOW	0
TERMINAL_ID_RISK_7DAY_WINDOW	0
TERMINAL_ID_NB_TX_30DAY_WINDOW	0
TERMINAL_ID_RISK_30DAY_WINDOW	0

dtype: int64

DATA EXPLORATORY



No Frauds

99.26 %

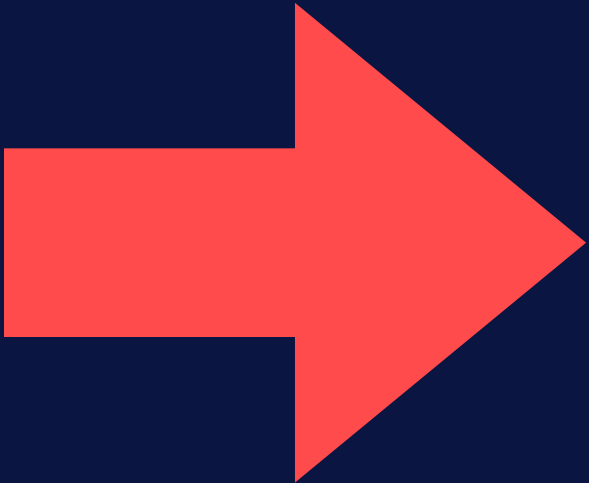
Frauds

0.74 %

FEATURE ENGINEERING



TX_HOUR
0:00 - 5:59
6:00 - 11:59
12:00 - 17:59
18:00 - 23:59



TRANSACTION_TIME_CATEGORY
Night
Morning
Afternoon
Evening





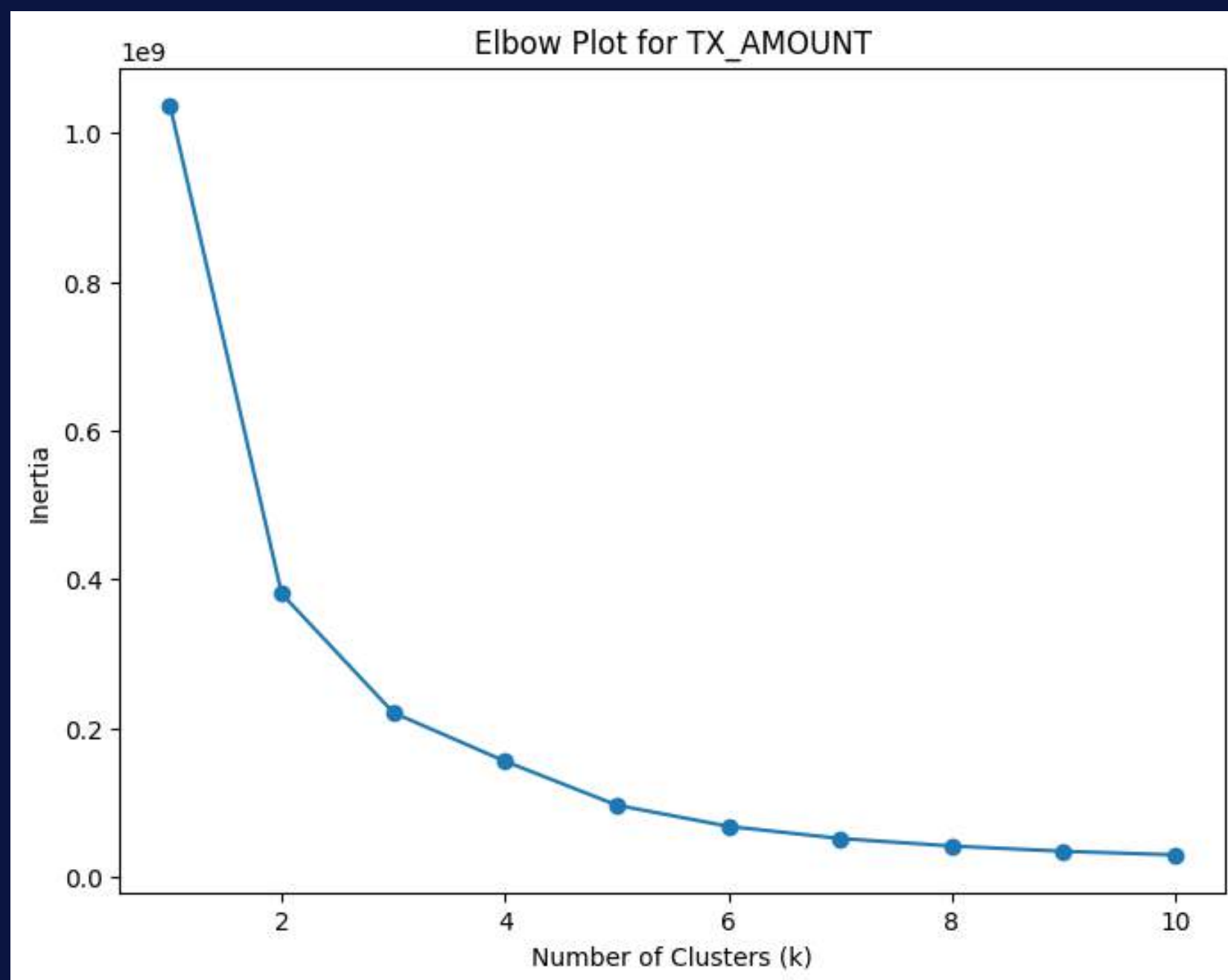
ASSOCIATION RULES



K-MEANS CLUSTERING



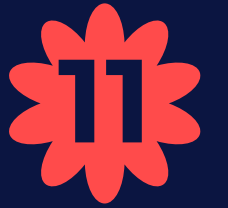
- TX_AMOUNT



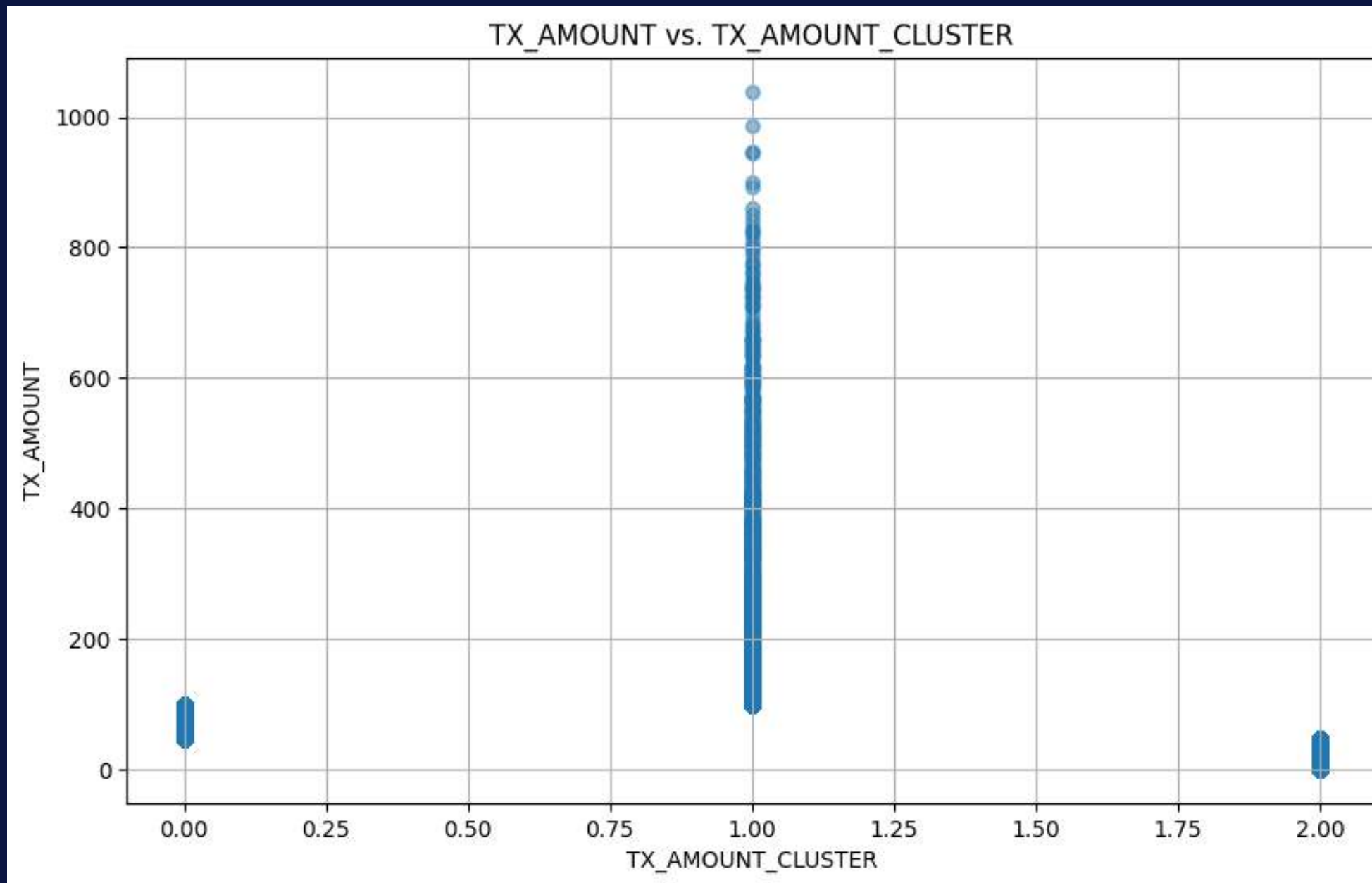
```
# from the elbow plot.  
optimal_k = 3  
  
# Perform KMeans clustering with the optimal k  
kmeans = KMeans(n_clusters=optimal_k, random_state=42)  
df['TX_AMOUNT_CLUSTER'] = kmeans.fit_predict(df[['TX_AMOUNT']])
```

	count	mean	median	min	max
TX_AMOUNT_CLUSTER					
0	208712	68.491887	66.84	45.17	99.24
1	80857	130.381692	121.55	99.25	1039.30
2	295608	22.067495	21.23	0.00	45.16

K-MEANS CLUSTERING



- TX_AMOUNT



TX_AMOUNT_CLUSTER

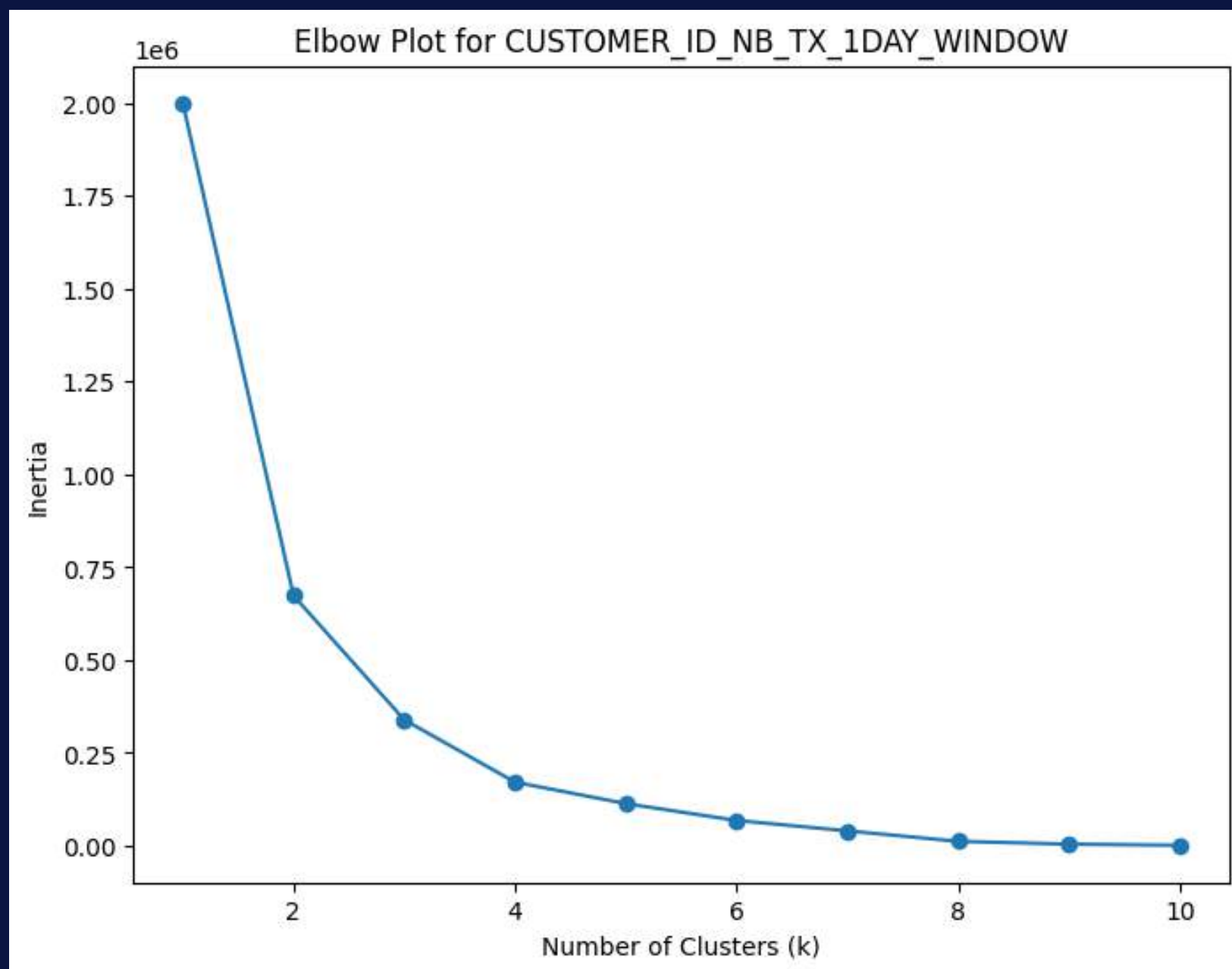
0	medium
1	high
2	low

	count	mean	median	min	max
TX_AMOUNT_CLUSTER					
high	80857	130.381692	121.55	99.25	1039.30
medium	208712	68.491887	66.84	45.17	99.24
low	295608	22.067495	21.23	0.00	45.16

K-MEANS CLUSTERING



- CUSTOMER_ID_NB_TX_1DAY_WINDOW



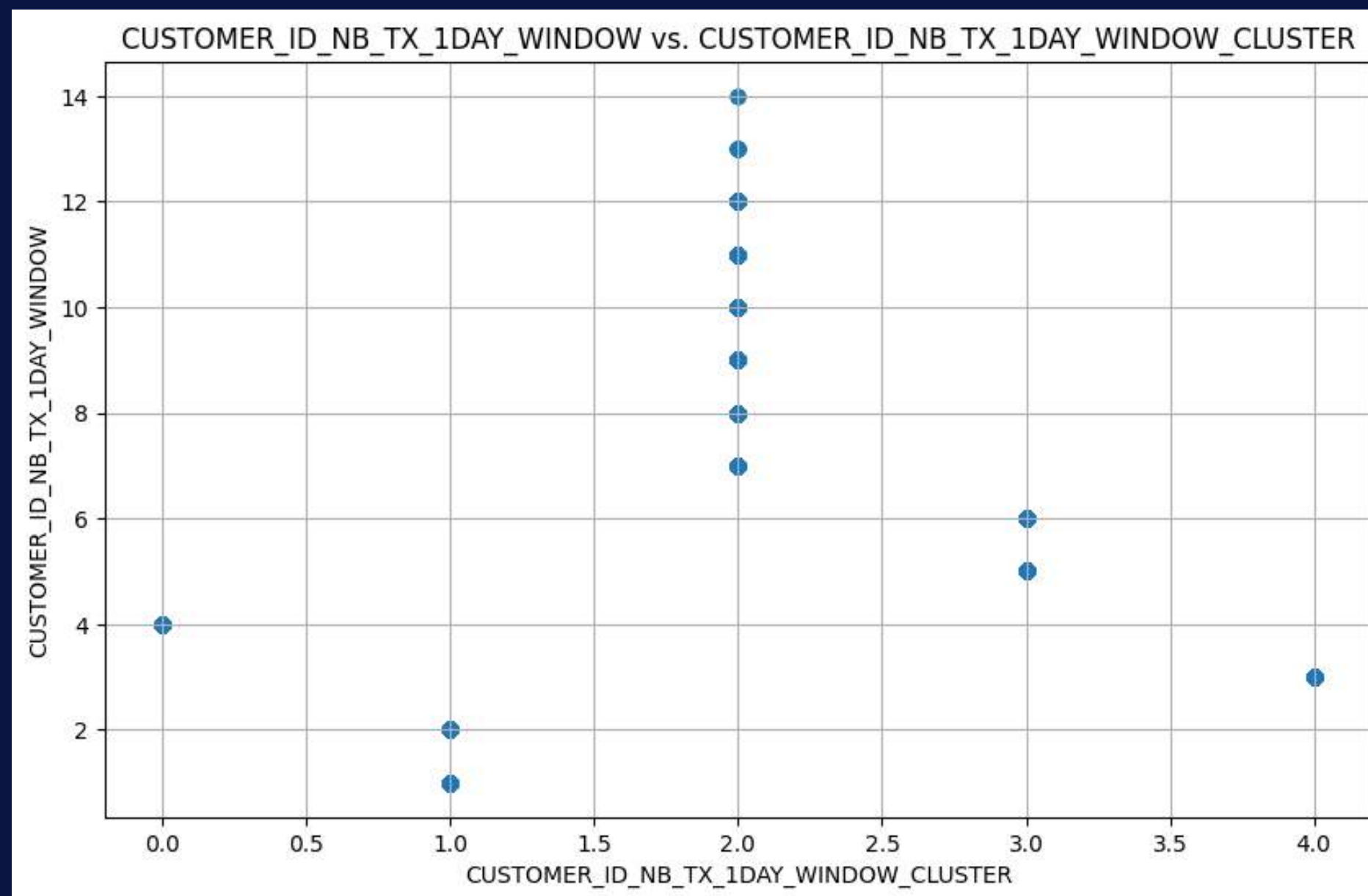
```
# from the elbow plot.  
optimal_k = 5  
  
# Perform KMeans clustering with the optimal k  
kmeans = KMeans(n_clusters=optimal_k, random_state=42)  
df['CUSTOMER_ID_NB_TX_1DAY_WINDOW_CLUSTER'] = kmeans.fit_predict(df[['CUSTOMER_ID_NB_TX_1DAY_WINDOW']])
```

	count	mean	median	min	max
CUSTOMER_ID_NB_TX_1DAY_WINDOW_CLUSTER					
0	106568	4.000000	4.0	4.0	4.0
1	190991	1.613704	2.0	1.0	2.0
2	41618	7.683983	7.0	7.0	14.0
3	119840	5.375976	5.0	5.0	6.0
4	126160	3.000000	3.0	3.0	3.0

K-MEANS CLUSTERING



- CUSTOMER_ID_NB_TX_1DAY_WINDOW



CUSTOMER_ID_NB_TX_1DAY_WINDOW_CLUSTER

0	High-Frequency Users
1	Occasional Users
2	Moderate Users
3	Regular Users
4	One-Time Users

	count	mean	median	min	max
CUSTOMER_ID_NB_TX_1DAY_WINDOW_CLUSTER					
High-Frequency Users	106568	4.000000	4.0	4.0	4.0
Moderate Users	41618	7.683983	7.0	7.0	14.0
Occasional Users	190991	1.613704	2.0	1.0	2.0
One-Time Users	126160	3.000000	3.0	3.0	3.0
Regular Users	119840	5.375976	5.0	5.0	6.0

ENCODE



CUSTOMER_ID_NB_TX_1DAY_WINDOW_CLUSTER

TX_AMOUNT_CLUSTER

TRANSACTION_TIME_CATEGORY

TX_FRAUD

TX_DURING_WEEKEND

- List Items
- Encode One-Hot Format

k1 FREQUENT ITEMSETS

	support	itemsets
0	0.182112	(CUSTOMER_ID_NB_TX_1DAY_WINDOW_CLUSTER=High-Frequency Users)
1	0.071120	(CUSTOMER_ID_NB_TX_1DAY_WINDOW_CLUSTER=Moderate Users)
2	0.326382	(CUSTOMER_ID_NB_TX_1DAY_WINDOW_CLUSTER=Occasional Users)
3	0.215593	(CUSTOMER_ID_NB_TX_1DAY_WINDOW_CLUSTER=One-Time Users)
4	0.204793	(CUSTOMER_ID_NB_TX_1DAY_WINDOW_CLUSTER=Regular Users)
...
1074	0.000027	(TX_FRAUD=1, TRANSACTION_TIME_CATEGORY=night, CUSTOMER_ID_NB_TX_1DAY_WINDOW_CLUSTER=Regular Users, TX_AMOUNT_CLUSTER=low, TX_DURING_WEEKEND=1)
1075	0.007001	(TX_AMOUNT_CLUSTER=medium, TRANSACTION_TIME_CATEGORY=night, CUSTOMER_ID_NB_TX_1DAY_WINDOW_CLUSTER=Regular Users, TX_FRAUD=0, TX_DURING_WEEKEND=0)
1076	0.000034	(TX_AMOUNT_CLUSTER=medium, TX_FRAUD=1, TRANSACTION_TIME_CATEGORY=night, CUSTOMER_ID_NB_TX_1DAY_WINDOW_CLUSTER=Regular Users, TX_DURING_WEEKEND=0)
1077	0.002574	(TX_AMOUNT_CLUSTER=medium, TRANSACTION_TIME_CATEGORY=night, CUSTOMER_ID_NB_TX_1DAY_WINDOW_CLUSTER=Regular Users, TX_FRAUD=0, TX_DURING_WEEKEND=1)
1078	0.000007	(TX_AMOUNT_CLUSTER=medium, TX_FRAUD=1, TRANSACTION_TIME_CATEGORY=night, CUSTOMER_ID_NB_TX_1DAY_WINDOW_CLUSTER=Regular Users, TX_DURING_WEEKEND=1)

ASSOCIATION RULES



Antecedents		
TX_AMOUNT_CLUSTER	High	
TX_DURING_WEEKEND	Yes	
TRANSACTION_TIME_CATEGORY	Afternoon	
CUSTOMER_ID_NB_TX_1DAY_WINDOW_CLUSTER	Regular Users	
Support	Confidence	Lift
0.000099	0.0344	4.64

Consequents	
TX_FRAUD	Fraud



ASSOCIATION RULES



Antecedents		
TX_AMOUNT_CLUSTER	High	
TX_DURING_WEEKEND	No	
TRANSACTION_TIME_CATEGORY	Morning	
CUSTOMER_ID_NB_TX_1DAY_WINDOW_CLUSTER	Occasional Users	
Support	Confidence	Lift
0.000313	0.0257	3.47

Consequents	
TX_FRAUD	Fraud



[illegible]



CONCLUSION

- มูลค่าสูง (TX_AMOUNT_CLUSTER=high) — เป็นปัจจัยเสี่ยงที่สำคัญที่สุด
- การรวมกันของปัจจัย พฤติกรรมลูกค้า + เวลา + มูลค่า — ทำให้เกิดความเสี่ยงในการฉ้อโกง
- วันหยุด (TX_DURING_WEEKEND=Weekend) — เสี่ยงที่จะฉ้อโกงมากกว่าวันธรรมดา
- ความถี่ในการใช้งาน

ลูกค้าที่ใช้งานน้อยมาก (One-Time/Occasional)  มีความเสี่ยงในการฉ้อโกง

ลูกค้าที่ใช้งานถี่มาก (High-Frequency) 

CLASSIFICATION

เพื่อสร้างโมเดล Machine Learning ที่สามารถตรวจจับ
ธุรกรรมฉ้อโกง (Fraudulent Transactions)
ได้อย่างมีประสิทธิภาพ



- แปลง TX_DATETIME เป็นรูปแบบ datetime
- Normalize ข้อมูลด้วย MinMaxScaler

```
1 # แปลงคอลัมน์ datetime และเรียงลำดับ
2 df['TX_DATETIME'] = pd.to_datetime(df['TX_DATETIME'])
3 df = df.sort_values('TX_DATETIME').reset_index(drop=True)
```

```
from sklearn.preprocessing import MinMaxScaler

# Create a MinMaxScaler object
scaler = MinMaxScaler()

# Select the columns to scale
columns_to_scale = ['TX_AMOUNT', 'CUSTOMER_ID_AVG_AMOUNT_1DAY_WINDOW', 'CUSTOMER_ID_AVG_AMOUNT_7DAY_WINDOW',
                    'CUSTOMER_ID_AVG_AMOUNT_30DAY_WINDOW', 'TX_AMOUNT_RATIO_7DAY']

# Fit and transform the selected columns
df[columns_to_scale] = scaler.fit_transform(df[columns_to_scale])
```

- สร้างคอลัมน์ อัตราส่วนของจำนวนเงินที่ทำธุรกรรม**เทียบกับ**ค่าเฉลี่ยของธุรกรรมใน 7 วันก่อนหน้า
- สร้างคอลัมน์ ระดับความเสี่ยงของจุดรับชำระเงินจากธุรกรรม 30 วันที่ผ่านมา
- แปลง TRANSACTION_TIME_CATEGORY ให้เป็นตัวเลข

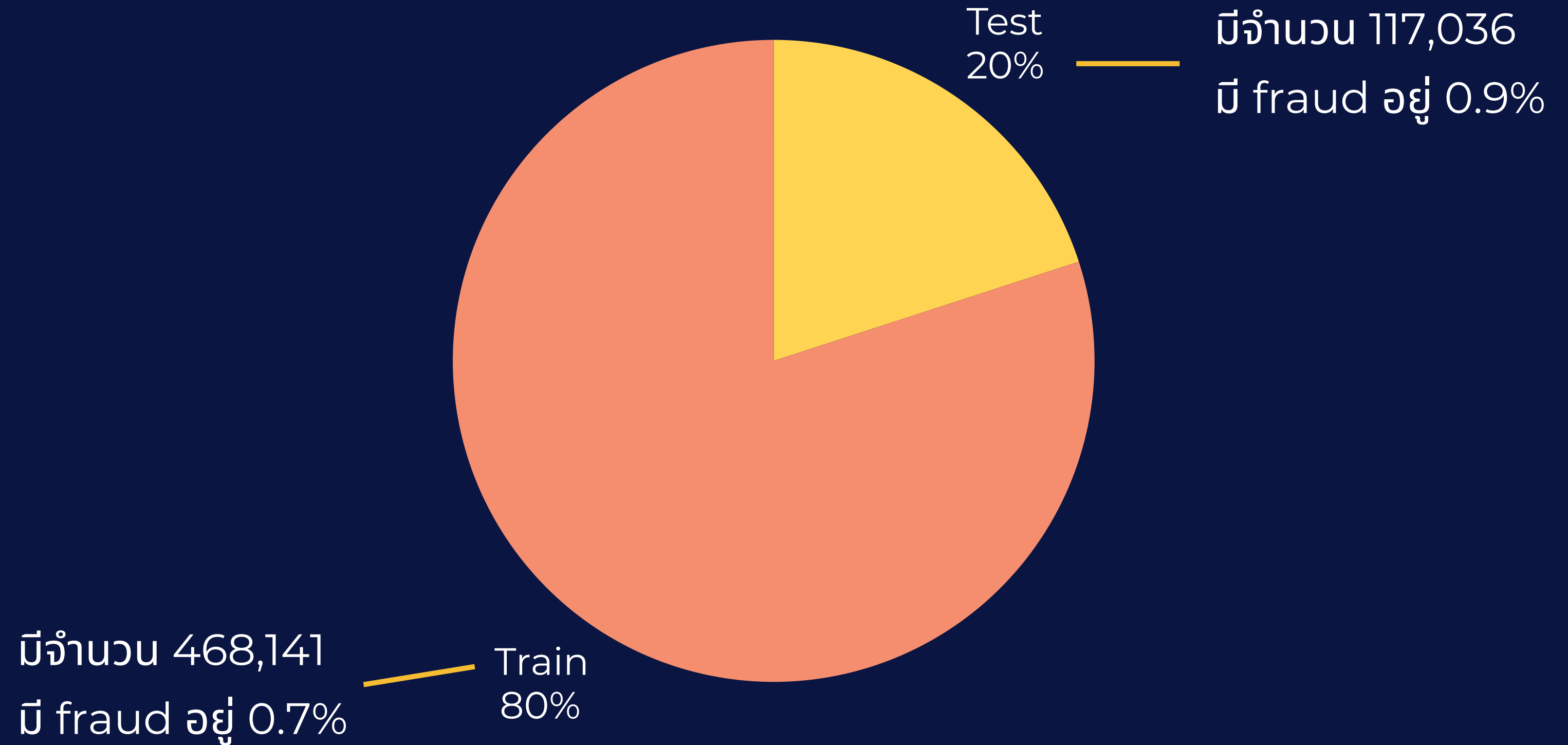
```
df['TX_AMOUNT_RATIO_7DAY'] = df['TX_AMOUNT'] / (df['CUSTOMER_ID_AVG_AMOUNT_7DAY_WINDOW'] + 1e-6)
df['TERMINAL_RISK_AMOUNT'] = df['TERMINAL_ID_RISK_30DAY_WINDOW'] * df['TX_AMOUNT']
df = pd.get_dummies(df, columns=['TRANSACTION_TIME_CATEGORY'], dtype=int)
```


FEATURES

```
features = [  
    'TX_DATETIME', 'TX_AMOUNT', 'TX_DURING_WEEKEND', 'CUSTOMER_ID_NB_TX_1DAY_WINDOW',  
    'CUSTOMER_ID_AVG_AMOUNT_1DAY_WINDOW', 'CUSTOMER_ID_NB_TX_7DAY_WINDOW',  
    'CUSTOMER_ID_AVG_AMOUNT_7DAY_WINDOW', 'CUSTOMER_ID_NB_TX_30DAY_WINDOW',  
    'CUSTOMER_ID_AVG_AMOUNT_30DAY_WINDOW', 'TERMINAL_ID_NB_TX_1DAY_WINDOW',  
    'TERMINAL_ID_RISK_1DAY_WINDOW', 'TERMINAL_ID_NB_TX_7DAY_WINDOW',  
    'TERMINAL_ID_RISK_7DAY_WINDOW', 'TERMINAL_ID_NB_TX_30DAY_WINDOW',  
    'TERMINAL_ID_RISK_30DAY_WINDOW',  
    'TRANSACTION_TIME_CATEGORY_afternoon', 'TRANSACTION_TIME_CATEGORY_evening',  
    'TRANSACTION_TIME_CATEGORY_morning', 'TRANSACTION_TIME_CATEGORY_night',  
    'TX_AMOUNT_RATIO_7DAY', 'TERMINAL_RISK_AMOUNT', 'TX_FRAUD'  
]
```

TRAIN AND TEST

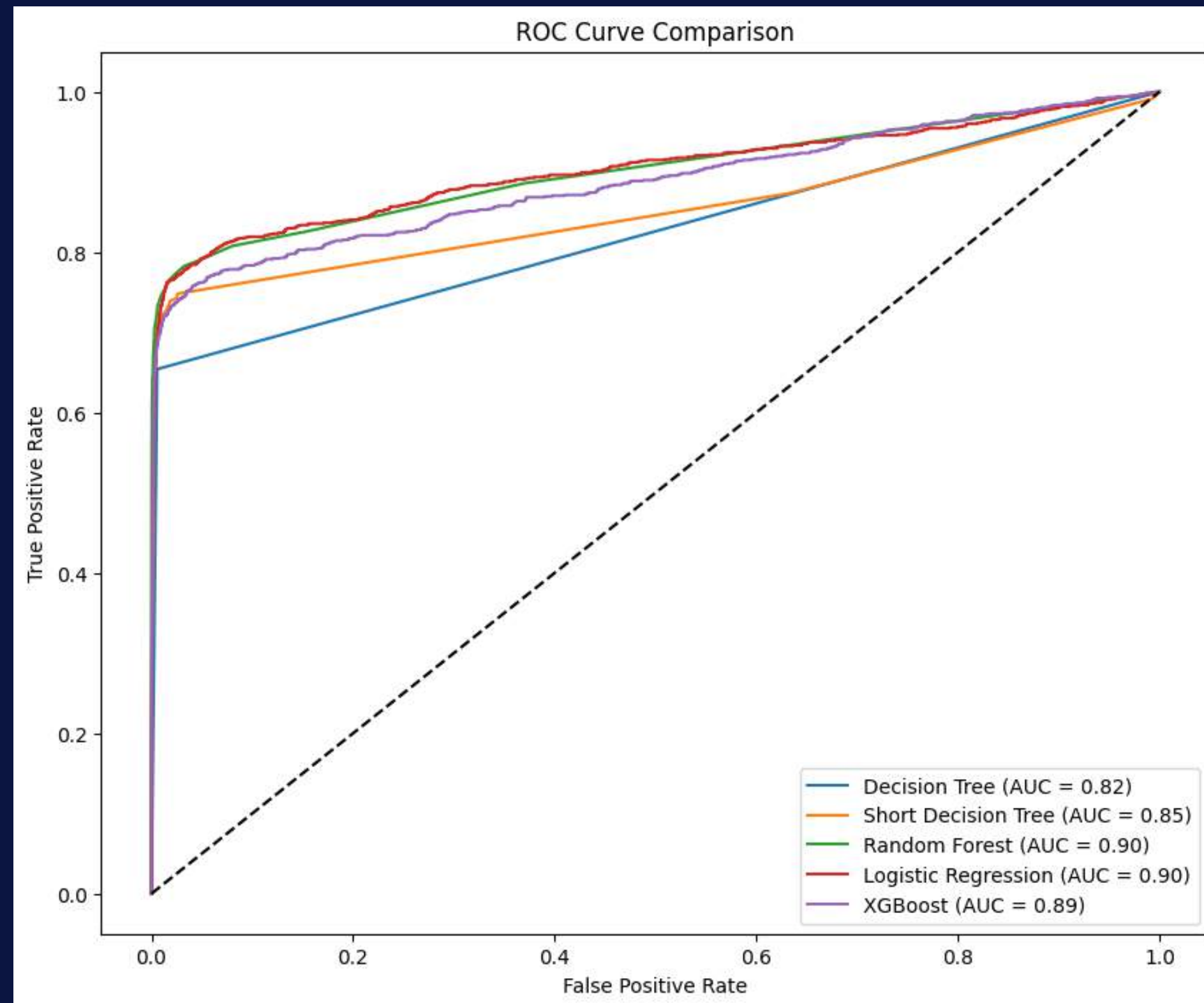
แบ่งข้อมูลตาม TX_DATETIME



MODEL

Model	Precision	Recall	F1	AUC-ROC
Logistic Regression	0.046	0.748	0.086	0.849
Decision Tree	0.335	0.526	0.402	0.758
Short Decision Tree	0.217	0.651	0.309	0.791
Random Forest	0.801	0.511	0.609	0.851
XGBoost	0.011	0.852	0.022	0.834

เปรียบเทียบ ROC CURVE ของแต่ละ MODEL



MODEL ที่ดีที่สุด คือ RANDOM FOREST

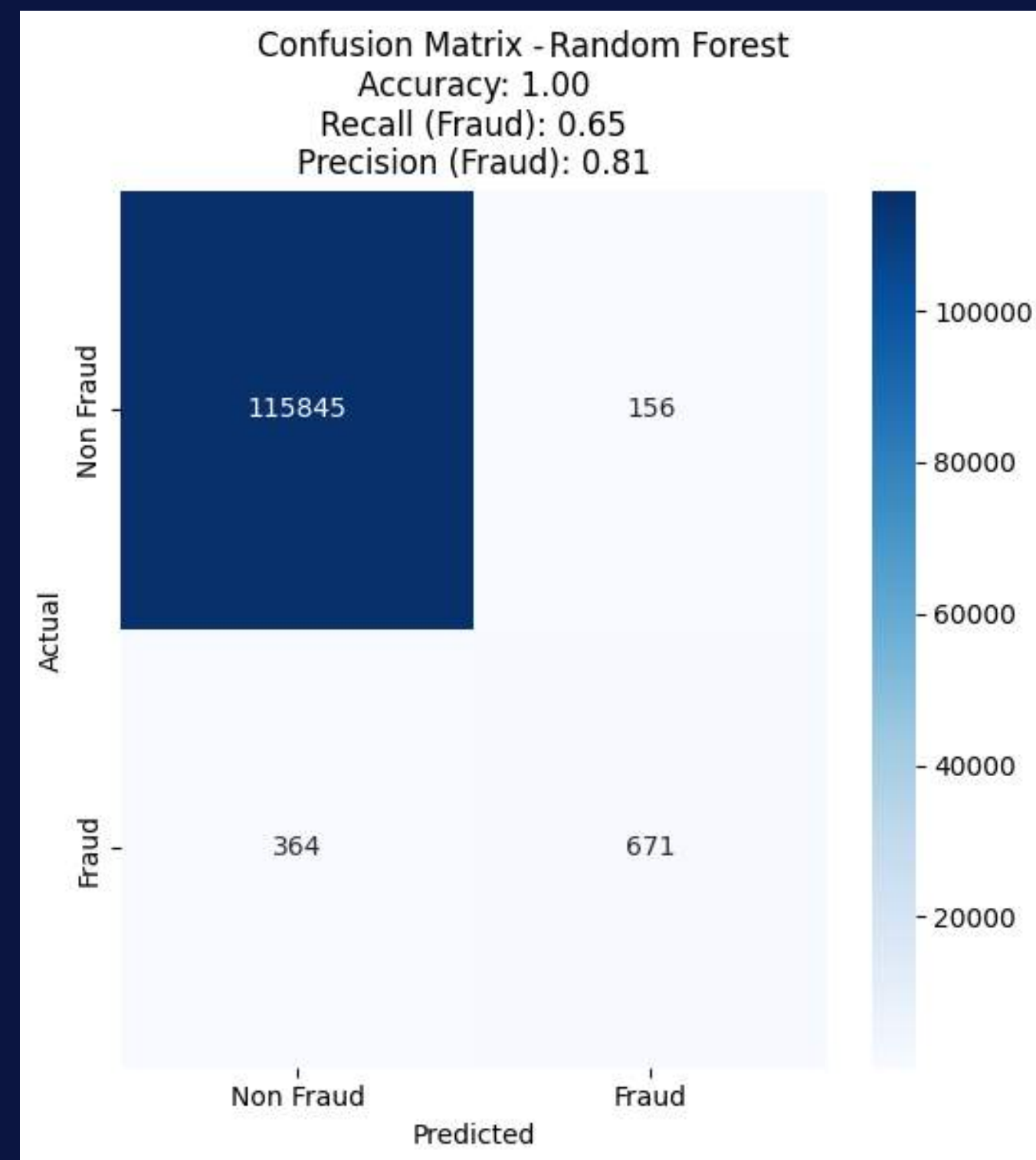
25

```
=====
Best Model: Random Forest
              precision    recall  f1-score   support

   Non Fraud      1.00      1.00      1.00    116001
     Fraud      0.81      0.65      0.72     1035

 accuracy              1.00    117036
 macro avg      0.90      0.82      0.86    117036
weighted avg      1.00      1.00      1.00    117036

AUC-ROC: 0.9014848530999255
```



THANK YOU

