DS-6

電子商務
交易詐欺預測

蘇芷儀[1]、賴威博[2]、藍璟誠[3]、劉育佑[2]、趙駖翰[4]

[1]政大經濟 [2]政大資科 [3]政大資管 [4]政大地政

Introduction

- Background & Objective
  - 隨著電子支付的快速發展，交易詐欺也越來越普遍，不僅給個人帶來經濟損失，也對金融機構和市場造成嚴重影響。
  - 因此，我們希望能夠開發一個交易詐欺預測模型，以提高對交易詐欺行為的識別能力。
- Methodology Overview
  - 我們採用了多種機器學習模型來訓練和評估，包括 Decision Tree、Random Forest、Logistic Regression、Support Vector Machine (SVM)、K-Nearest Neighbor (KNN)和XGBoost 等。通過數據預處理、特徵工程和參數調整，最終選擇出最優模型進行預測。
- Key Steps
  - 數據預處理：包括處理缺失值、特徵縮放和類別變量編碼等。
  - EDA：通過可視化和統計分析，了解數據特徵及其分佈情況。
  - 模型訓練與評估：對多種模型進行訓練，並使用多種評估指標比較其性能。
  - 調整參數：利用交叉驗證和超參數調優技術進一步提升模型性能。

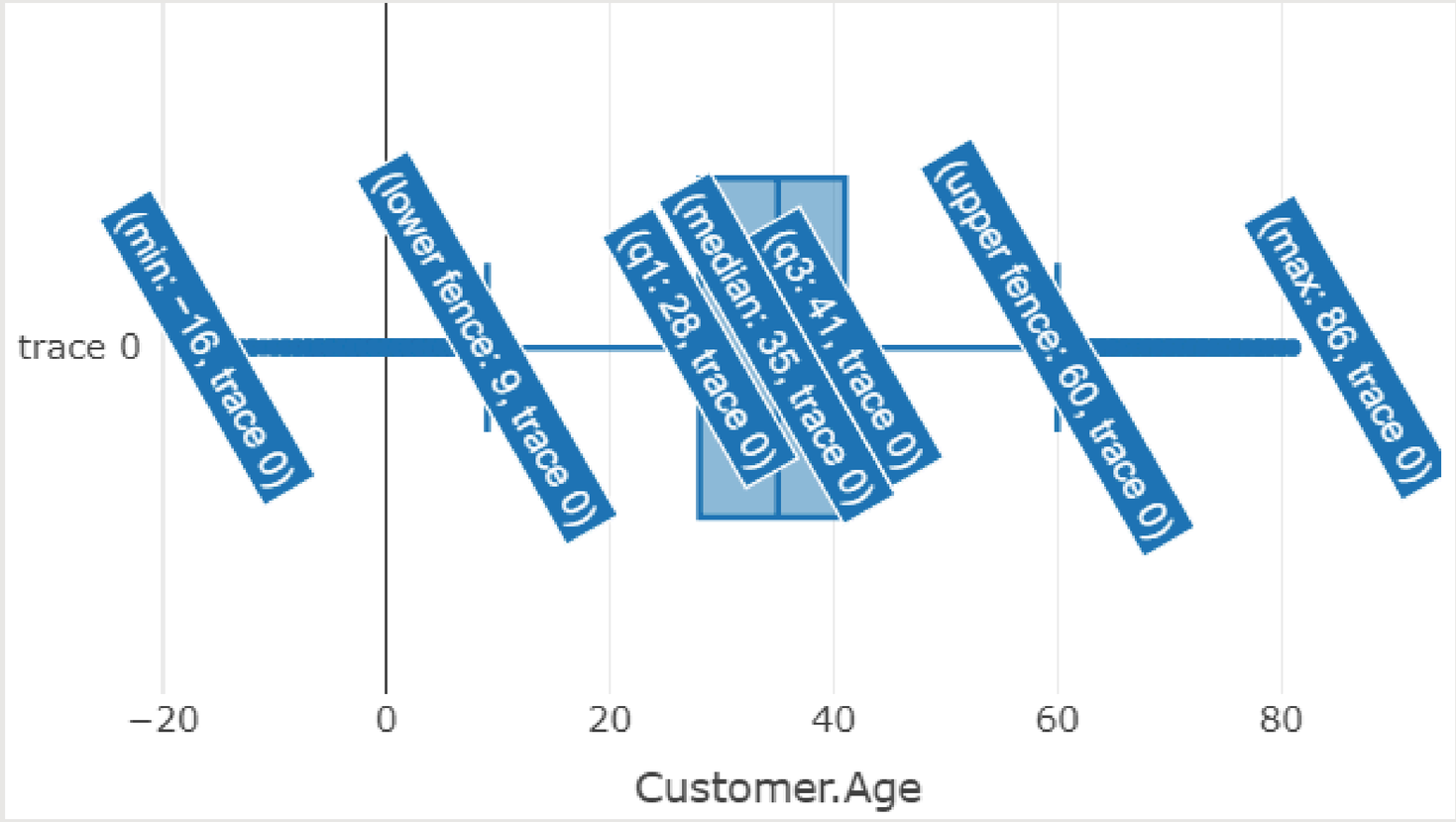
Dataset

- **Data Source**: Kaggle - Fraudulent E-Commerce Transactions
  - 數據是完全合成的，使用 Python Faker 庫生成，以模擬真實的交易模式和欺詐場景
- **Time Period:** 2024 年 1 月 1 日至 4 月 3 日
- **Data Frame:** Data1: (1472952, 16) ; Data2: (23634, 16)
- **Data Features**
  - 基本資訊: Transaction ID, Customer ID, Transaction Amount, Transaction Date, Payment Method, Product Category, Quantity, Transaction Hour
  - 客戶資訊: Customer Age, Customer Location, Account Age Days
  - 交易詳情: Device Used, IP Address, Shipping Address, Billing Address
  - 是否為詐欺: Is Fraudulent

數據預處理 - 缺失值、重複值檢查

```
> # 檢查是否有缺失值
> colSums(is.na(train_df))
    Transaction.ID        Customer.ID Transaction.Amount   Transaction.Date
                 0                  0                  0                  0
    Payment.Method   Product.Category           Quantity       Customer.Age
                 0                  0                  0                  0
 Customer.Location        Device.Used         IP.Address   Shipping.Address
                 0                  0                  0                  0
   Billing.Address      Is.Fraudulent   Account.Age.Days   Transaction.Hour
                 0                  0                  0                  0
> # 檢查每一行是否重複
> sum(duplicated(train_df))
[1] 0
```

數據預處理 - 年齡箱形圖


(min: −16, trace 0)
(lower fence: 9, trace 0)
(q1: 28, trace 0)
(median: 35, trace 0)
(q3: 41, trace 0)
(upper fence: 60, trace 0)
(max: 86, trace 0)
trace 0
−20
0
20
40
60
80
Customer.Age

數據預處理 - 交易日期轉換

```r
# 將Transaction Date轉為datetime
df$Transaction.Date <- as.Date(df$Transaction.Date)

# 每個月幾號、星期幾、月份
df$Transaction.Day <- as.numeric(format(df$Transaction.Date, "%d"))
df$Transaction.DOW <- as.numeric(format(df$Transaction.Date, "%u"))
df$Transaction.Month <- as.numeric(format(df$Transaction.Date, "%m"))
```

Transaction.Date
2024-02-20 05:58:41
2024-02-25 08:09:45
2024-03-18 03:42:55
2024-03-16 20:41:31
2024-01-15 05:08:17

Transaction.Day	Transaction.DOW	Transaction.Month
20	2	2
25	7	2
18	1	3
16	6	3
15	1	1

數據預處理 - 年齡異常值修正

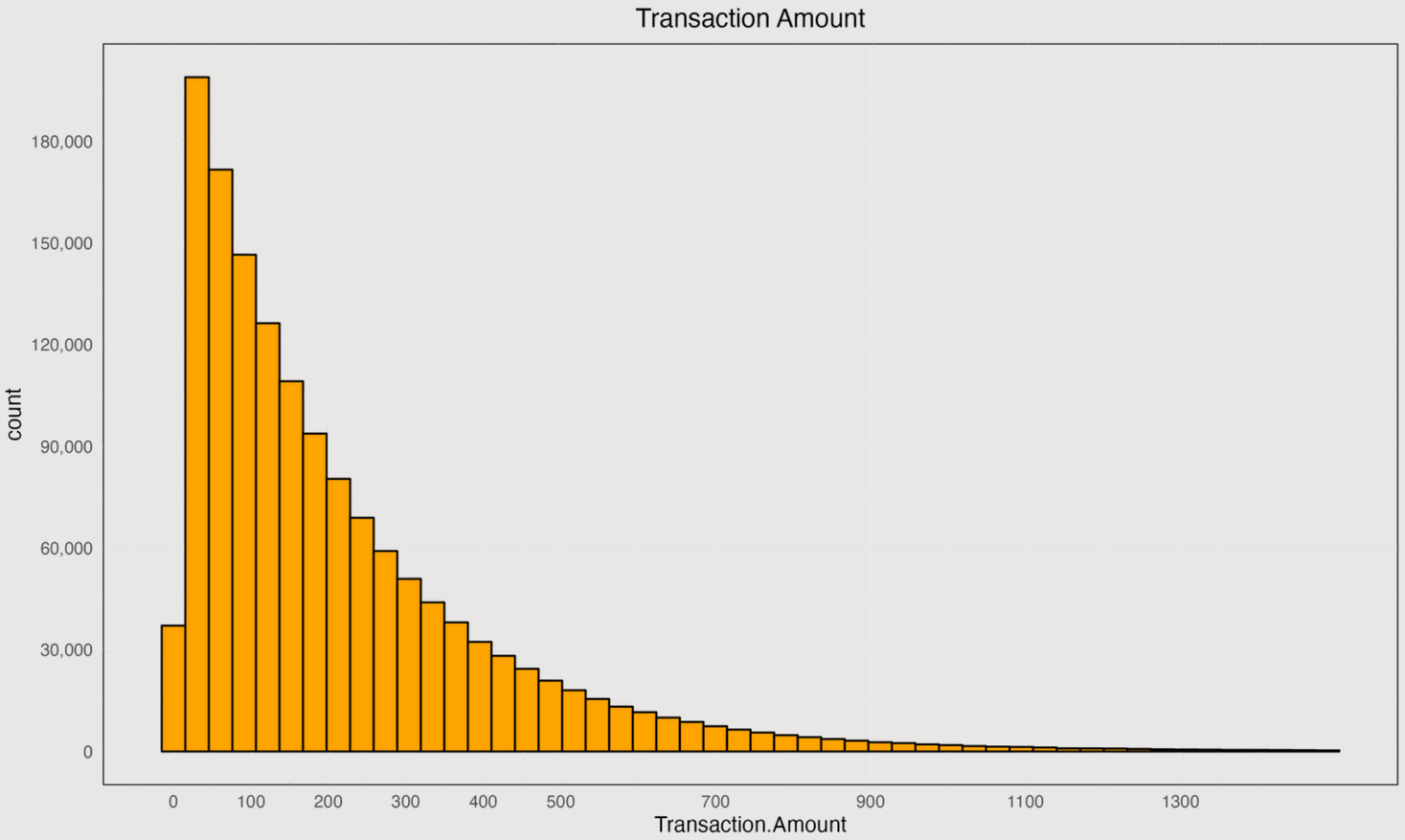
```r
# 修正Customer Age中的異常值
mean_value <- round(mean(df$Customer.Age, na.rm = TRUE), 0)
df$Customer.Age <- ifelse(df$Customer.Age <= -9,
                          abs(df$Customer.Age),
                          df$Customer.Age)
df$Customer.Age <- ifelse(df$Customer.Age < 9,
                          mean_value,
                          df$Customer.Age)
```

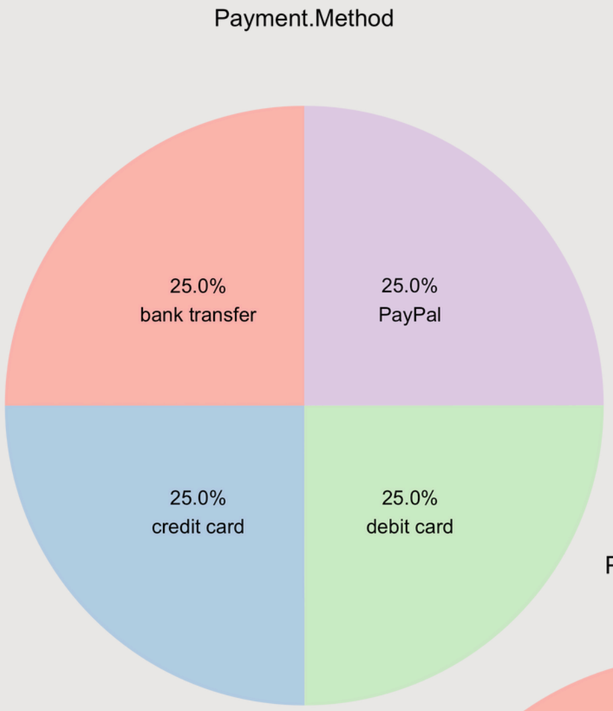
數據預處理 - 地址處理、去除低相關features

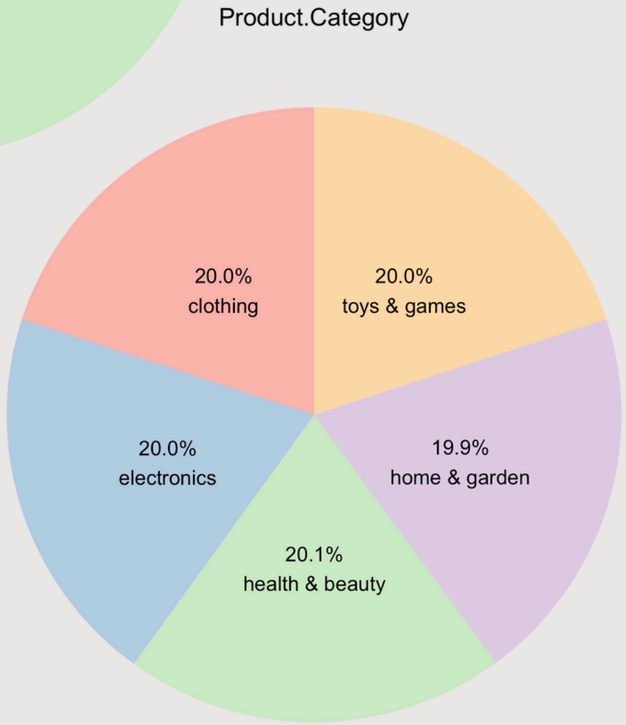
```r
# Shipping Address與Billing Address的異同（異=0，同=1）
df$Is.Address.Match <- as.integer(df$Shipping.Address == df$Billing.Address)

# 除去不相關的features
df <- df[, !names(df) %in% c("Transaction.ID", "Customer.ID",
                             "Customer.Location","IP.Address",
                             "Transaction.Date", "Shipping.Address",
                             "Billing.Address")]
```

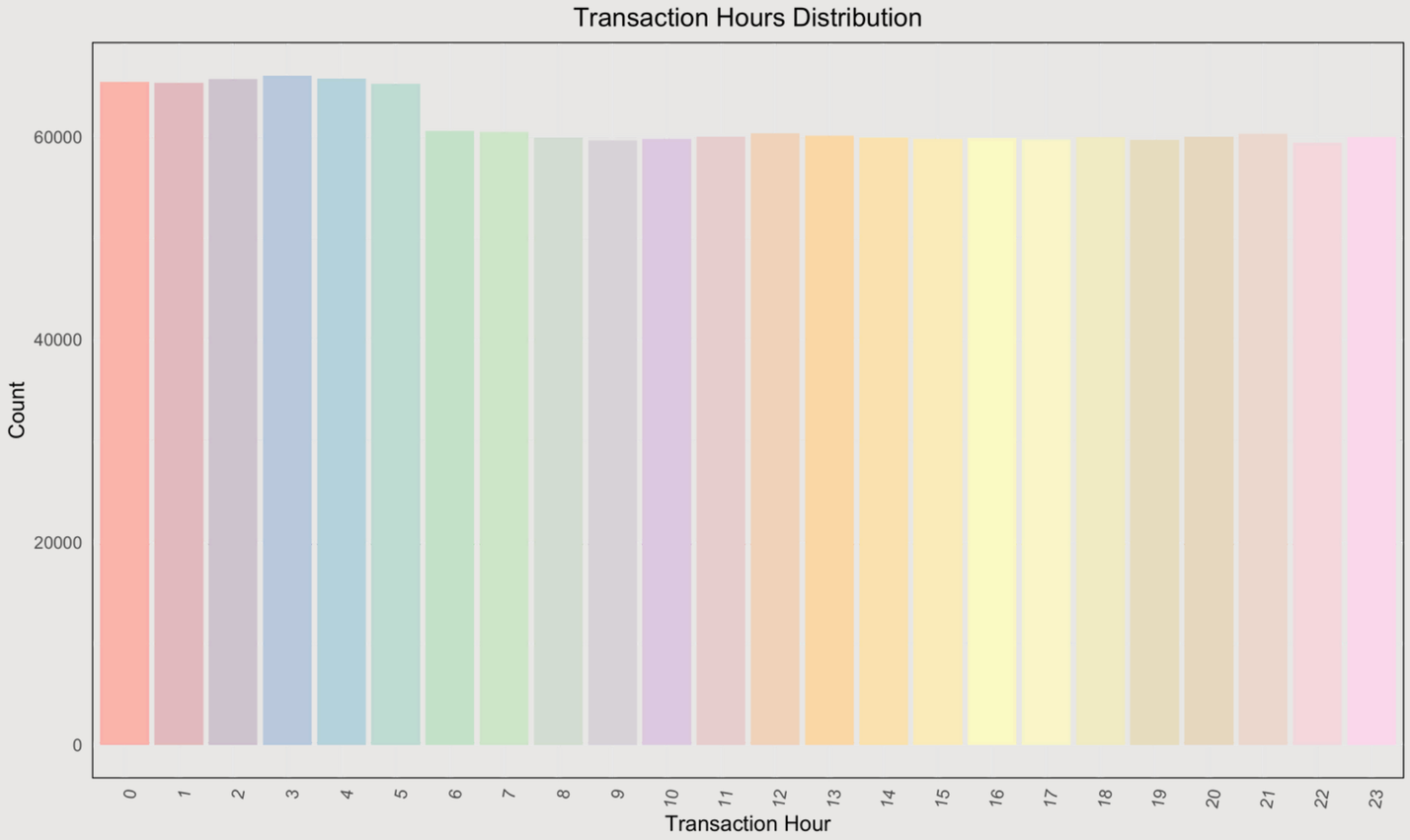
EDA - 基本資訊1

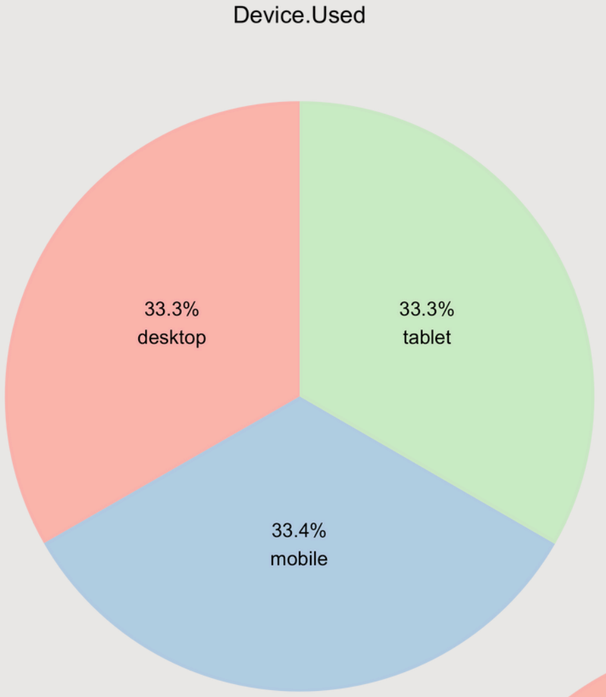

Transaction Amount
count
180,000
150,000
120,000
90,000
60,000
30,000
0
0
100
200
300
400
500
700
900
1100
1300
Transaction.Amount

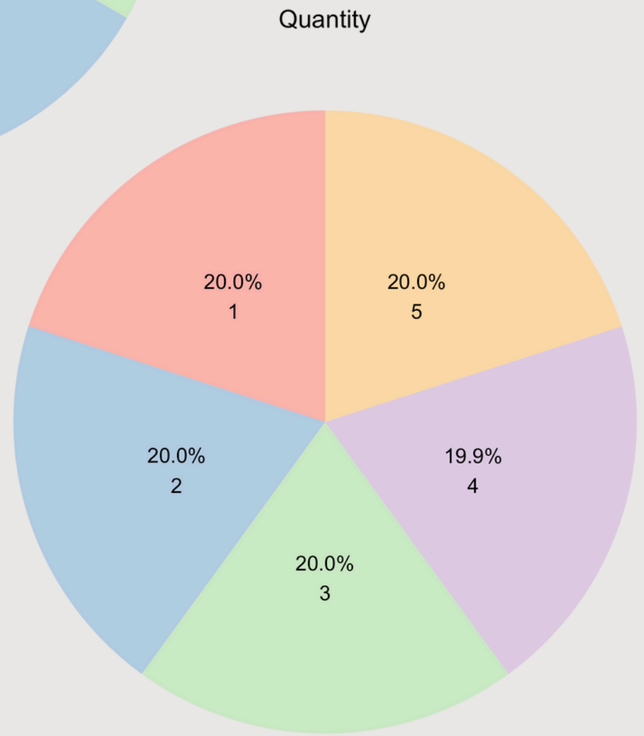

Payment.Method
25.0%
bank transfer
25.0%
PayPal
25.0%
credit card
25.0%
debit card


Product.Category
20.0%
clothing
20.0%
toys & games
20.0%
electronics
19.9%
home & garden
20.1%
health & beauty

EDA - 基本資訊2

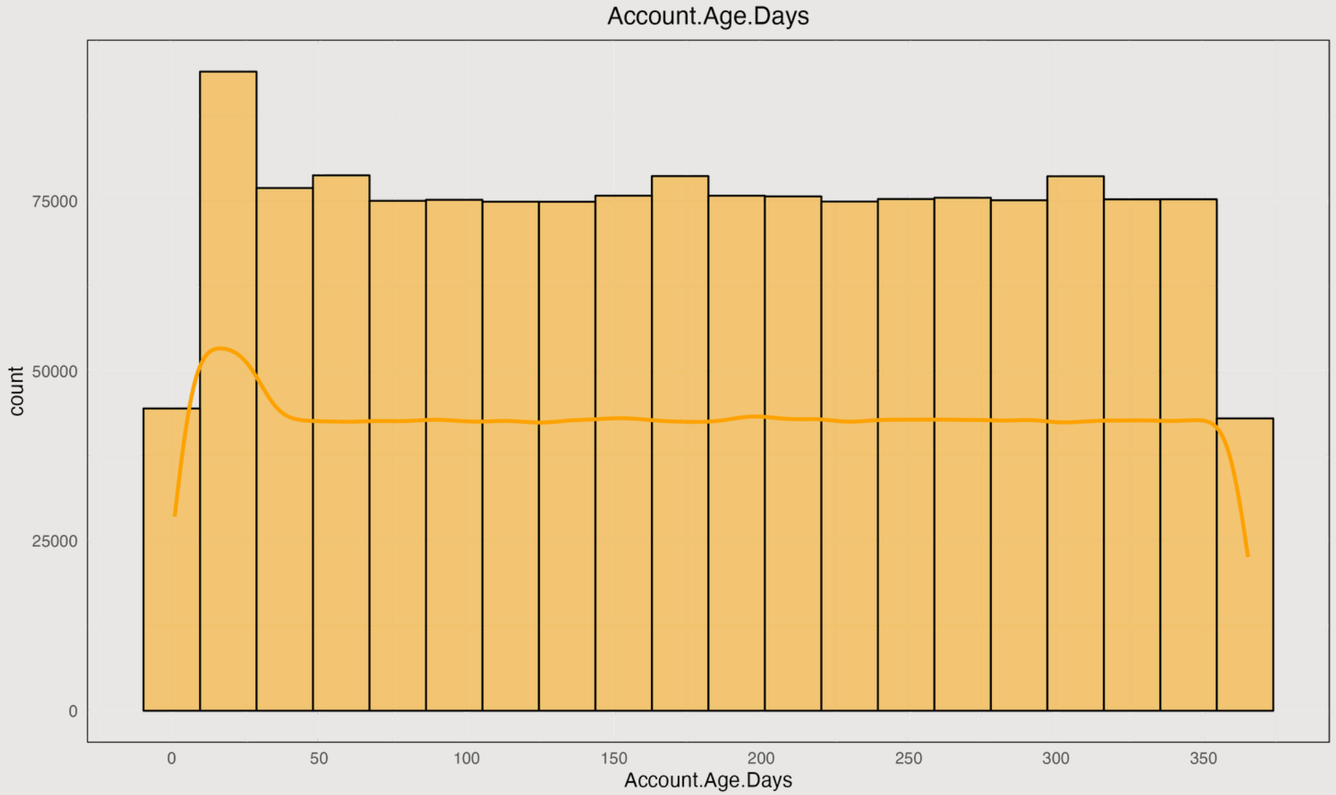

Transaction Hours Distribution
Count
60000
40000
20000
0
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
Transaction Hour


Device.Used
33.3%
desktop
33.3%
tablet
33.4%
mobile

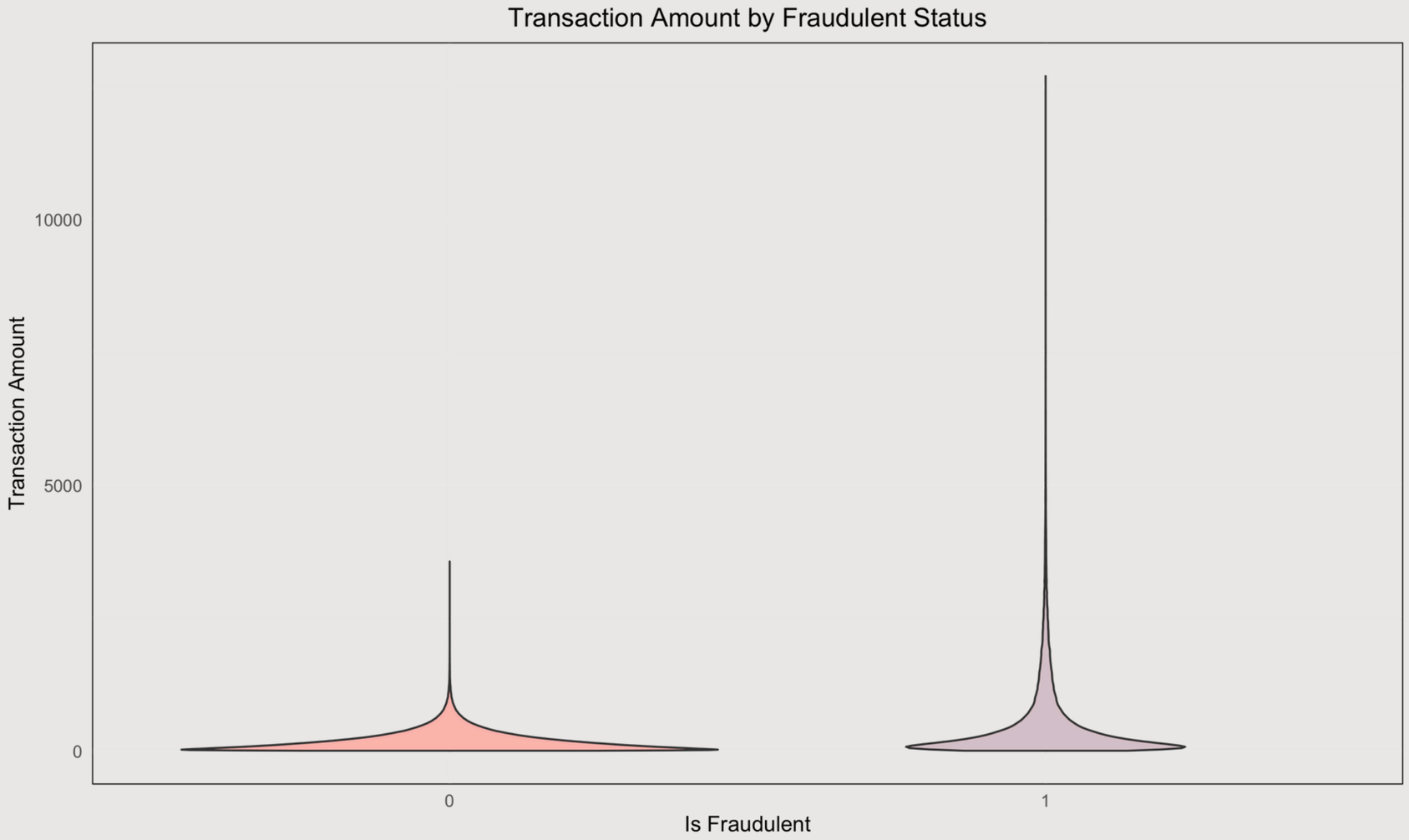

Quantity
20.0%
1
20.0%
5
20.0%
2
19.9%
4
20.0%
3

EDA - 客戶資訊

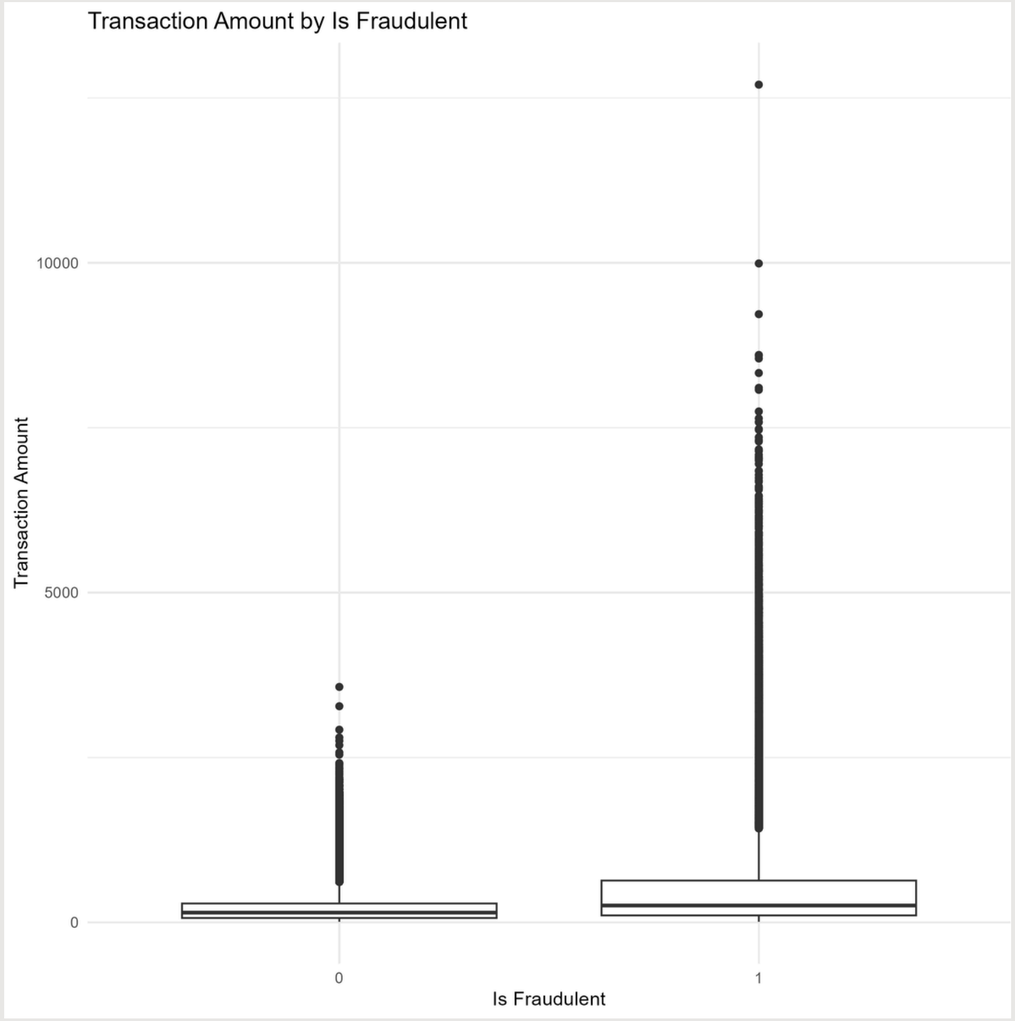

Customer.Age
count
200000
150000
100000
50000
0
10
20
30
40
50
60
70
80
90
Customer.Age


Account.Age.Days
count
75000
50000
25000
0
0
50
100
150
200
250
300
350
Account.Age.Days

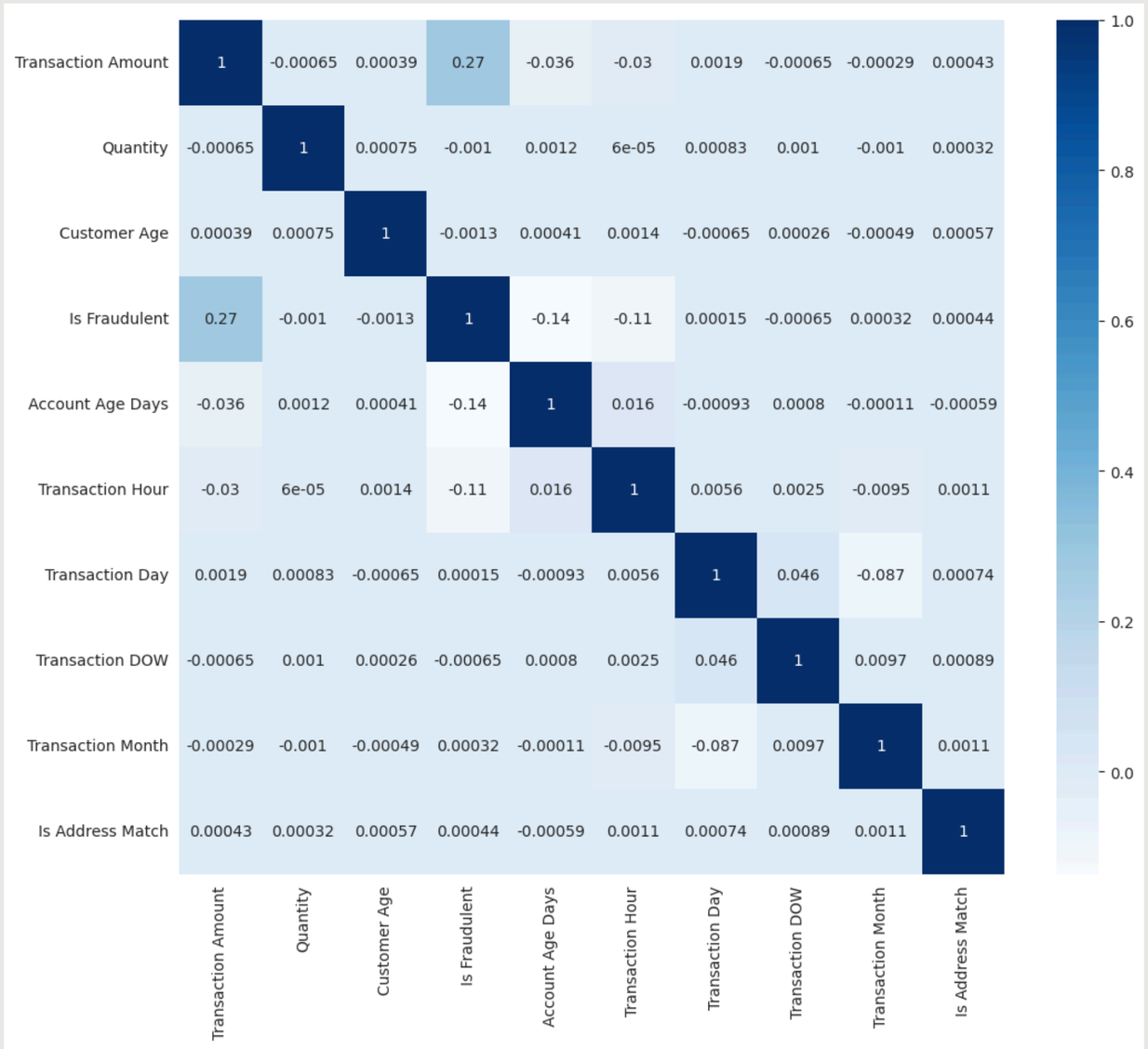
EDA - Is Fraudulent

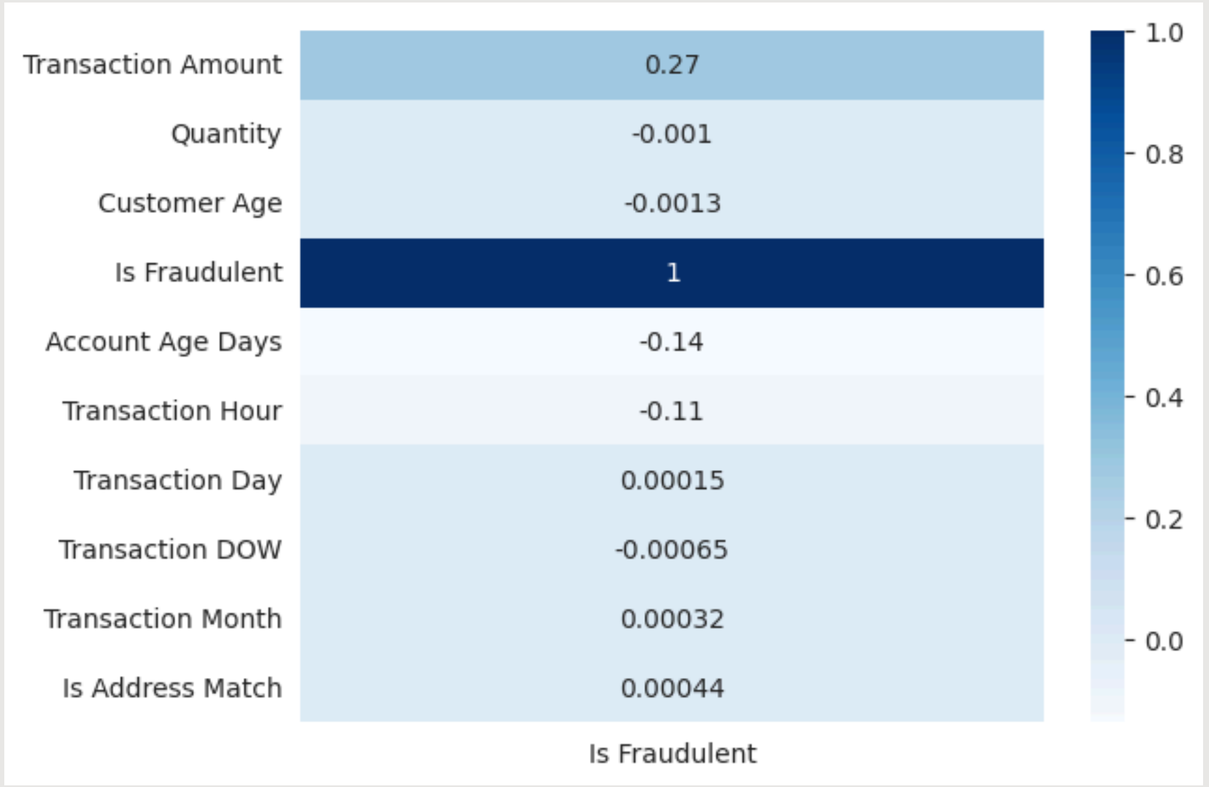


EDA (which is Fraudulent)



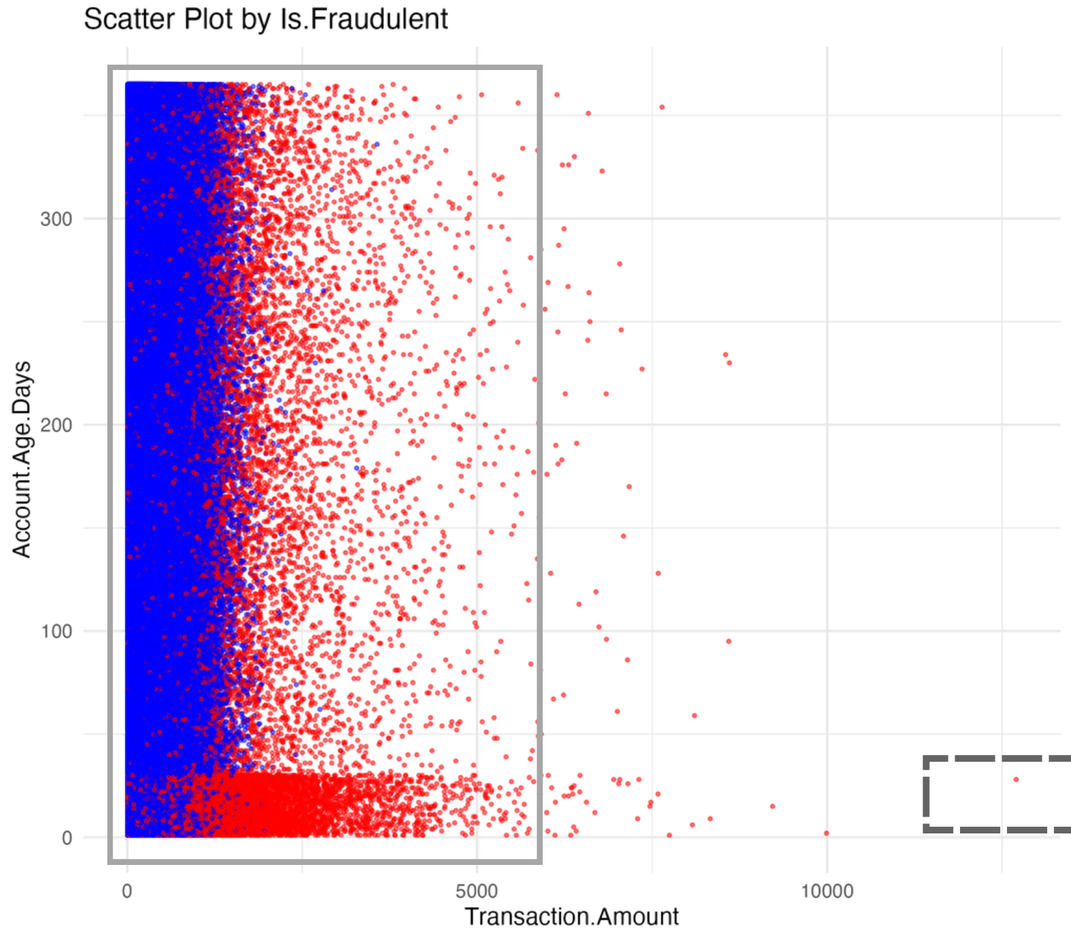
Correlation Heatmap

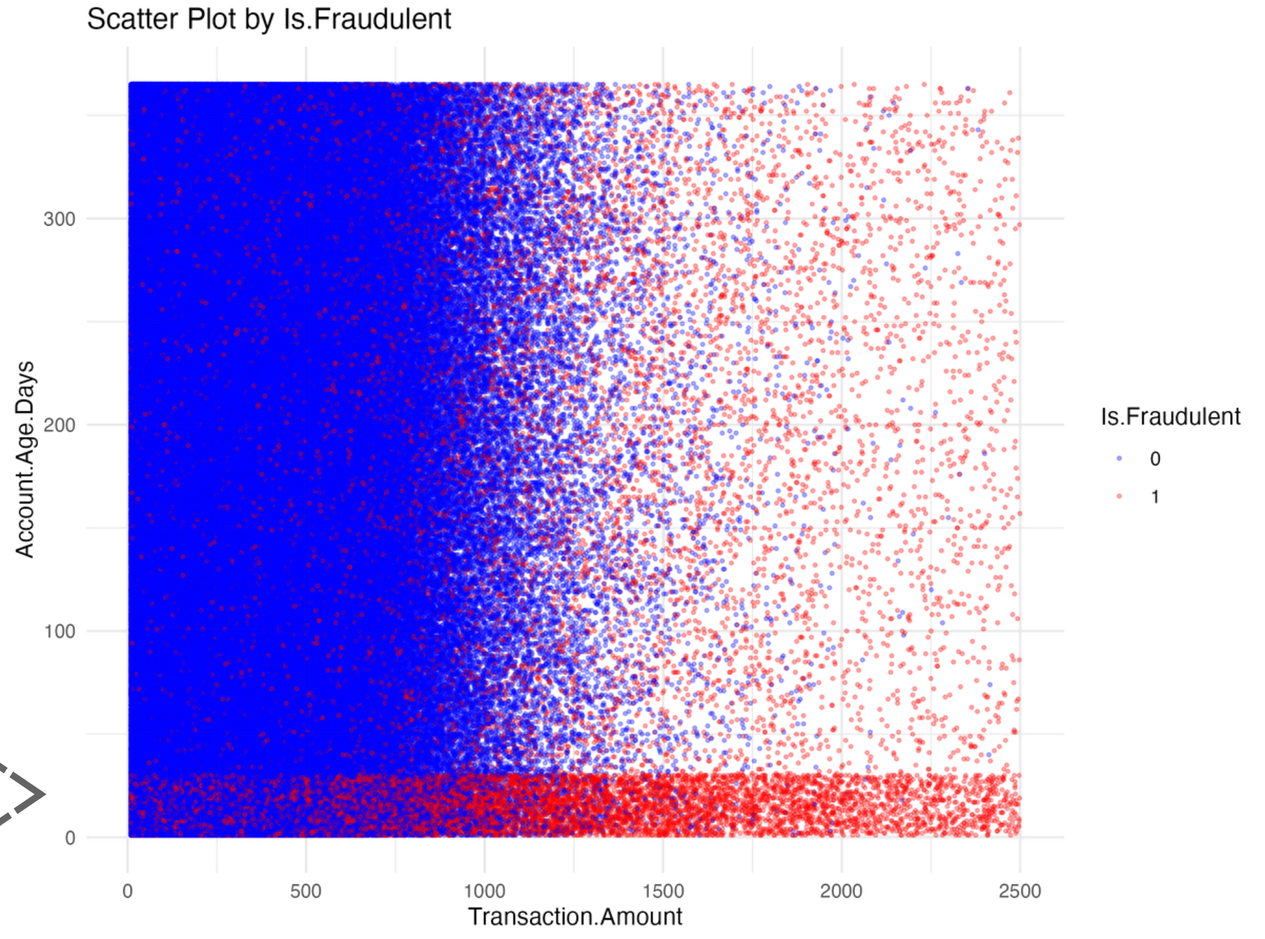




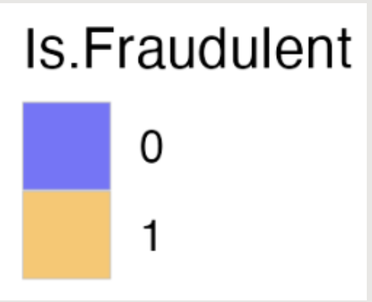
EDA (which is Fraudulent)

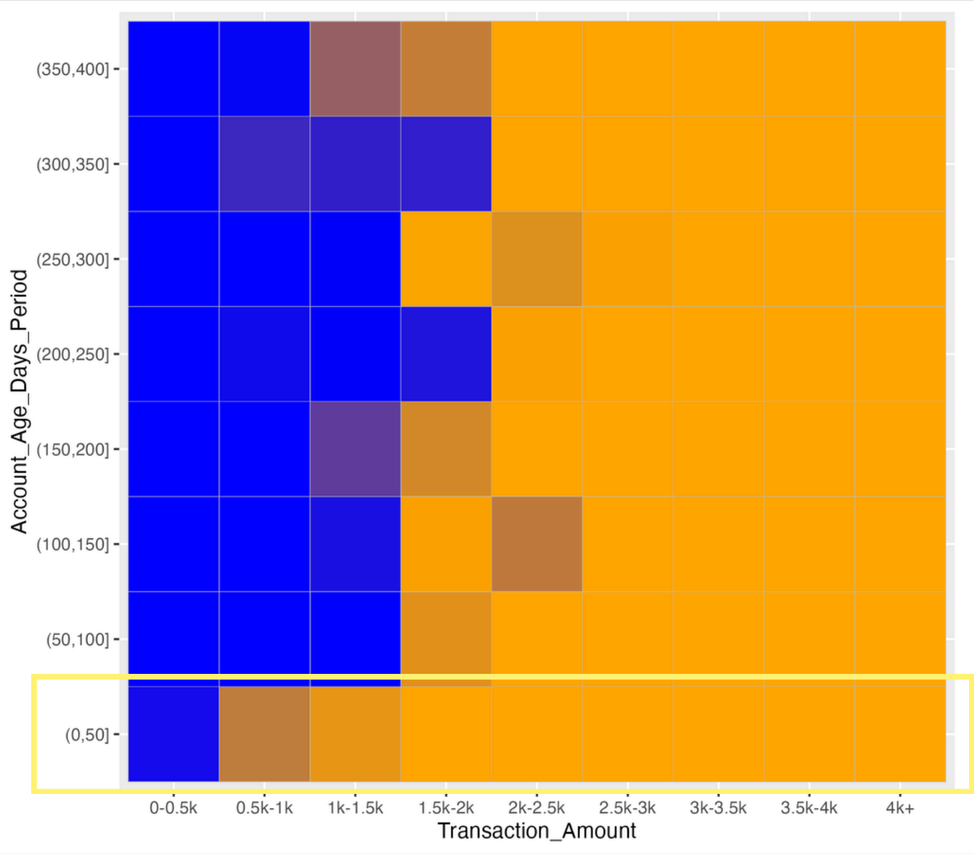
- Transaction Amount vs. Account Age Days



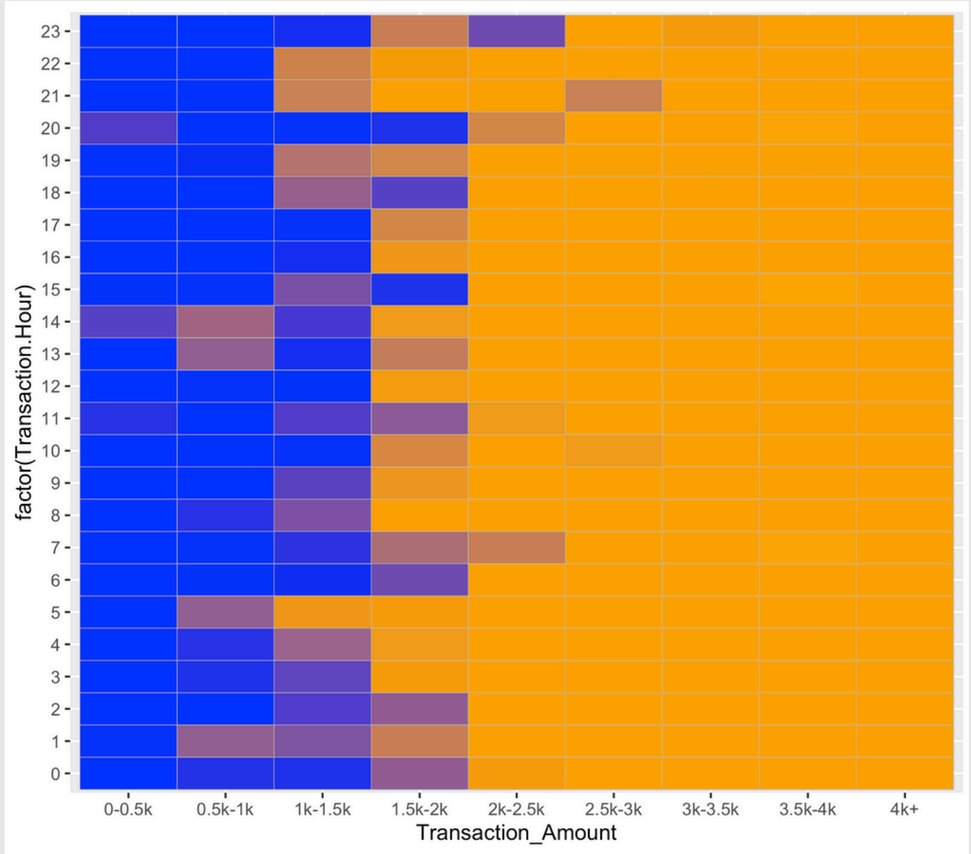


EDA (which is Fraudulent)





(v.s. Account_Age_Days_Period)



(v.s. Transaction_Hour)

Data without over/undersampling

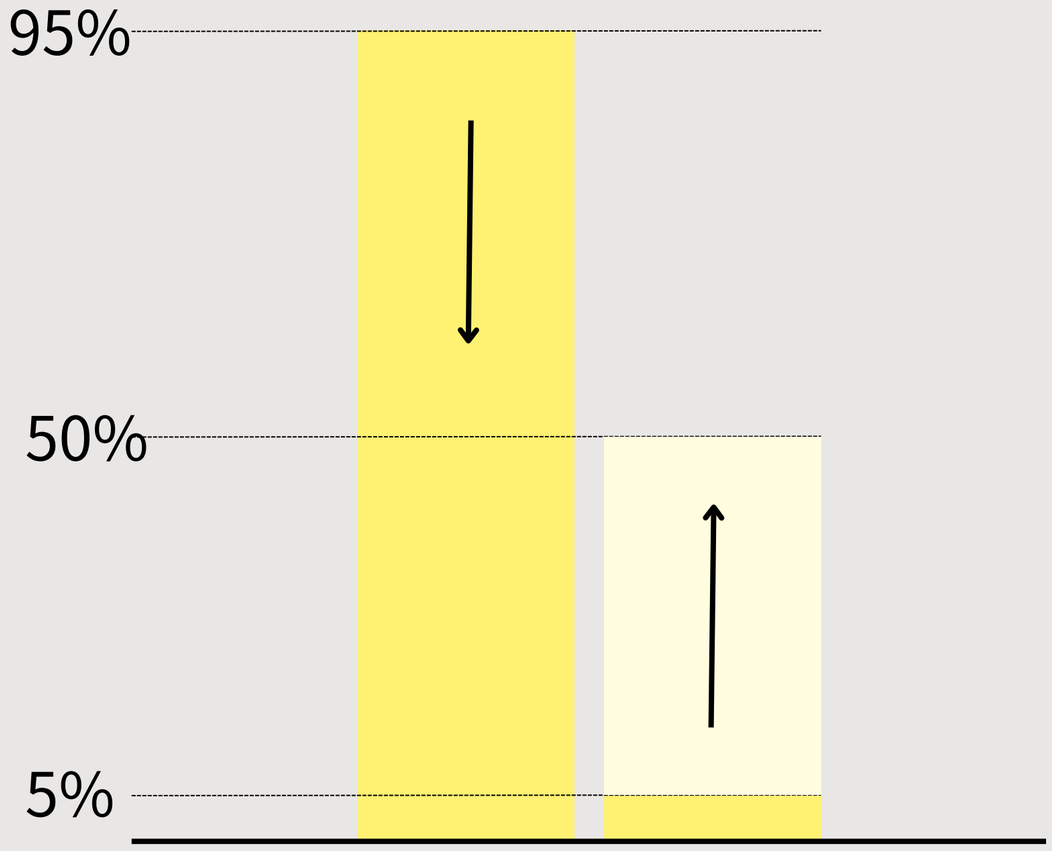
- Train Data : 1472952筆（1399114/73838）
- Test Data ：23634筆（22412/1222）
- Results :
  - Logistic Regression: 79(s)、accuracy : 0.9532
  - Decision Tree: 237 (s)、accuracy : 0.954
  - Random Forest : **out of memory**
  - XGBoost : 8.68 (s)、accuracy : 0.954
  - Sensitivity : 0.99、Specificity : **0.11**

Data with undersampling

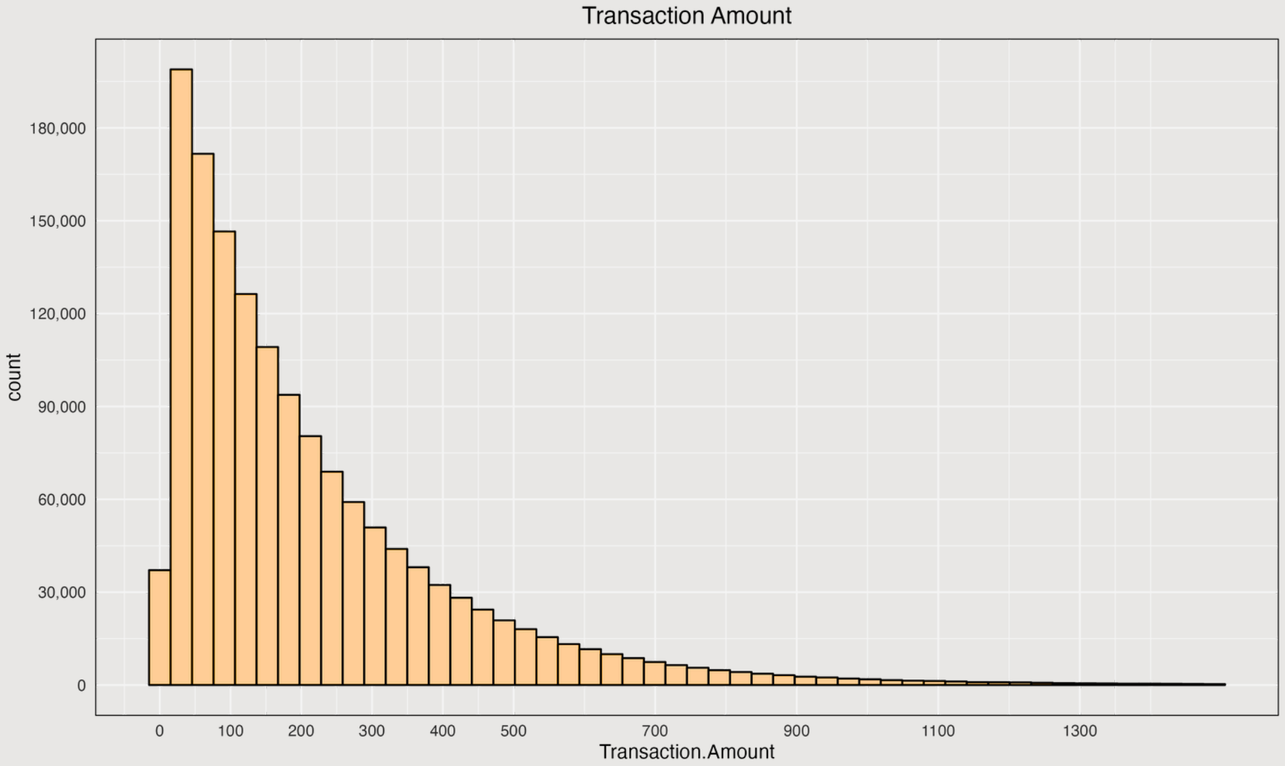
- Train Data : 147676筆（73838/73838）
- Test Data ：23634筆（22412/1222）
- Results :
  - 時間大幅提升
  - accuracy 約0.7
  - specificity 大幅提升（0.5~0.8）

Data with over&undersampling

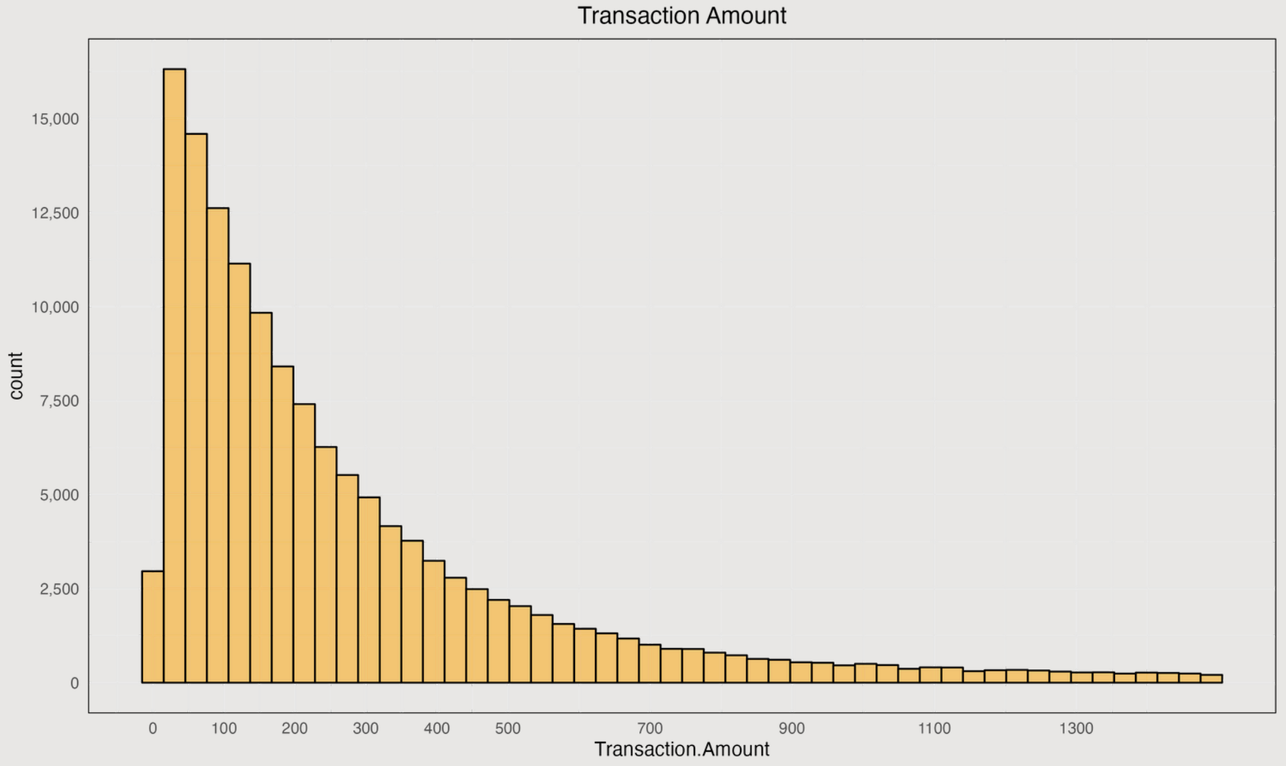
- Train Data : 1472952筆
  (736476/736476)
- Test Data : 23634筆
  (22412/1222)
- Results :
  - 與不做任何處理的原始資料結果相似



EDA (under sampling)



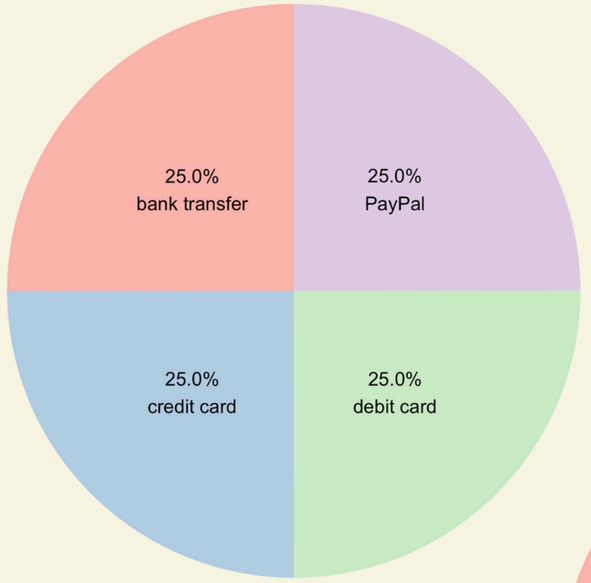
Before



After

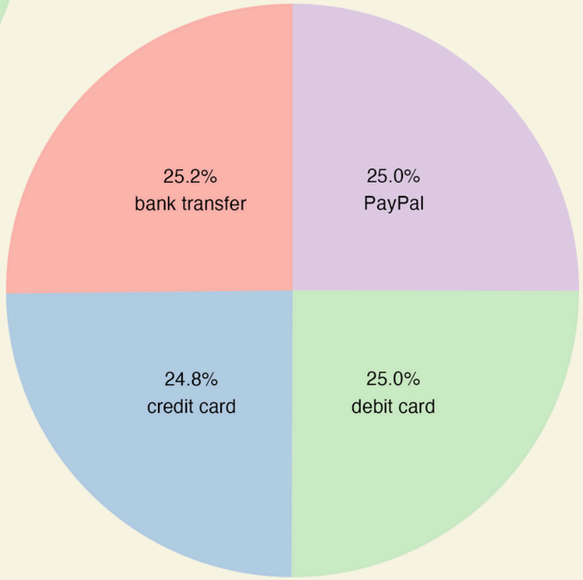
EDA (under sampling)

Payment.Method



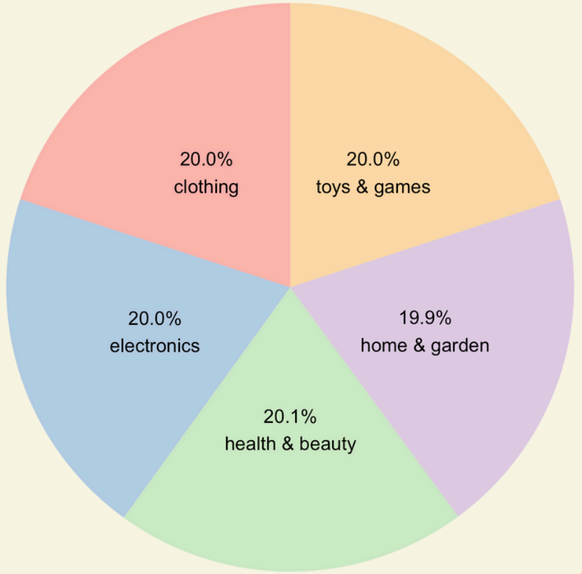
Before

Payment.Method



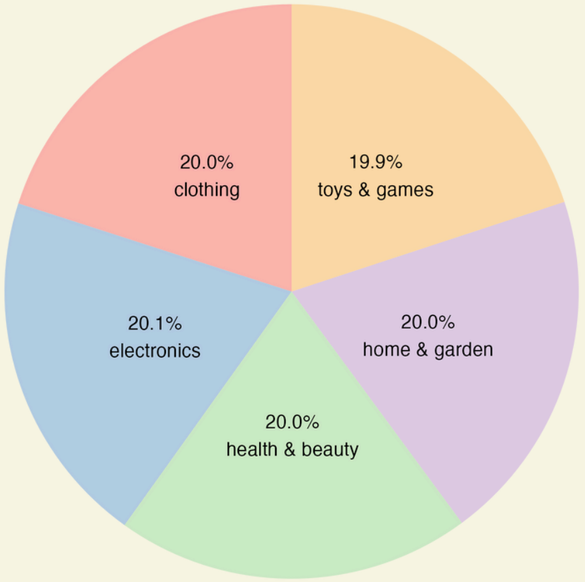
After

Product.Category



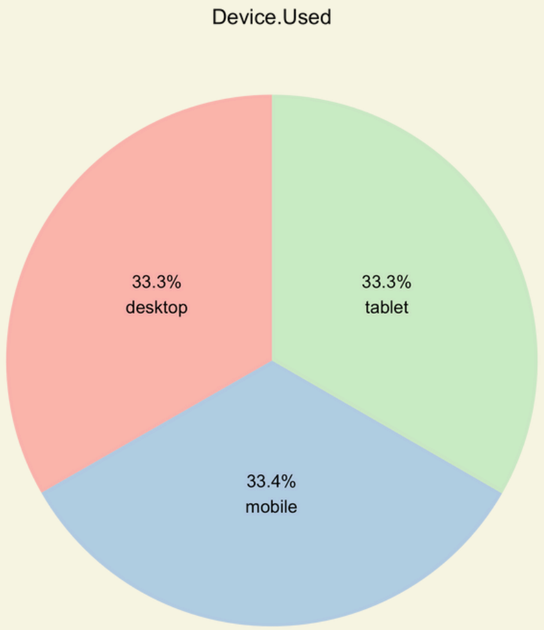
Before

Product.Category

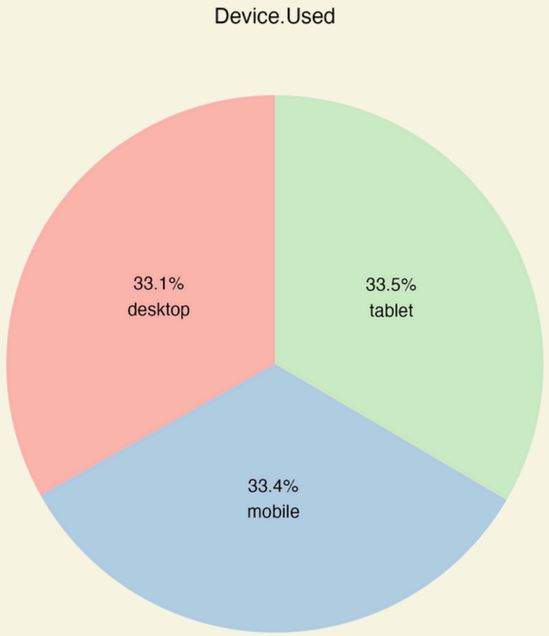


After

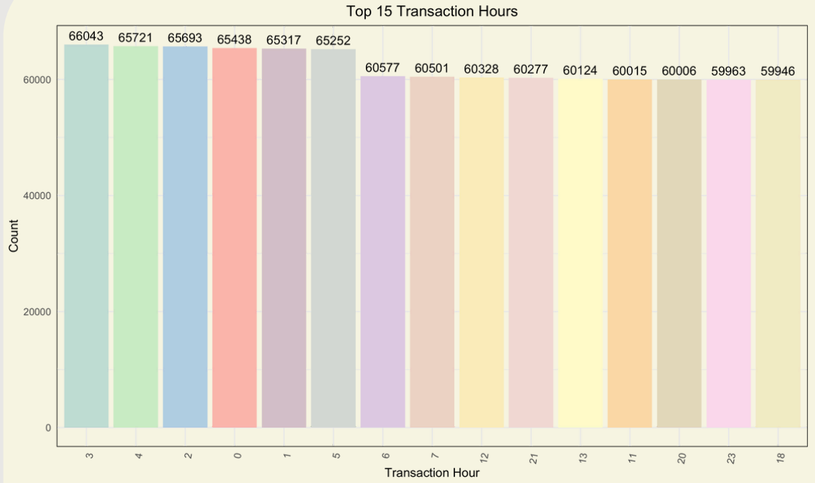
EDA (under sampling)



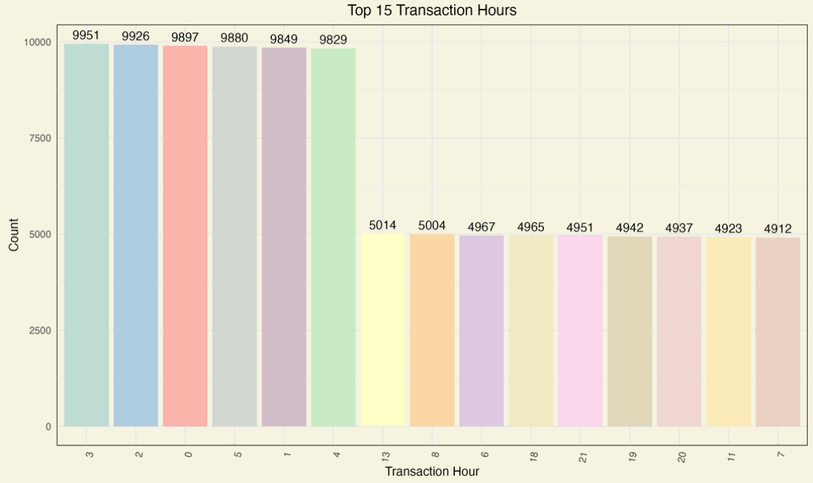
Before



After

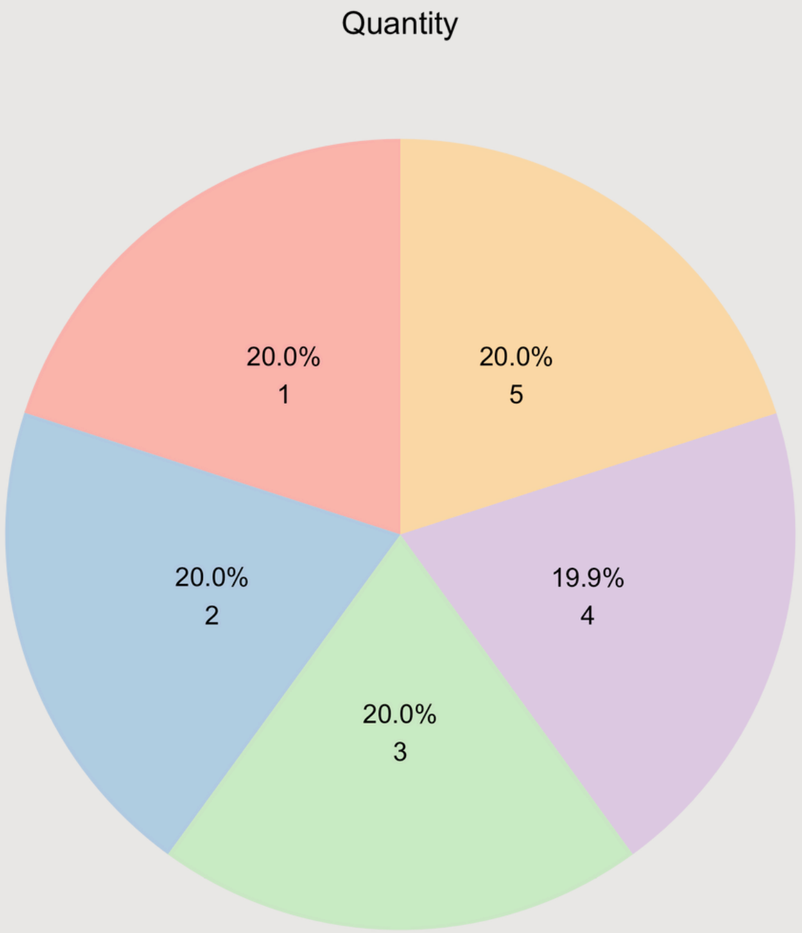


Before

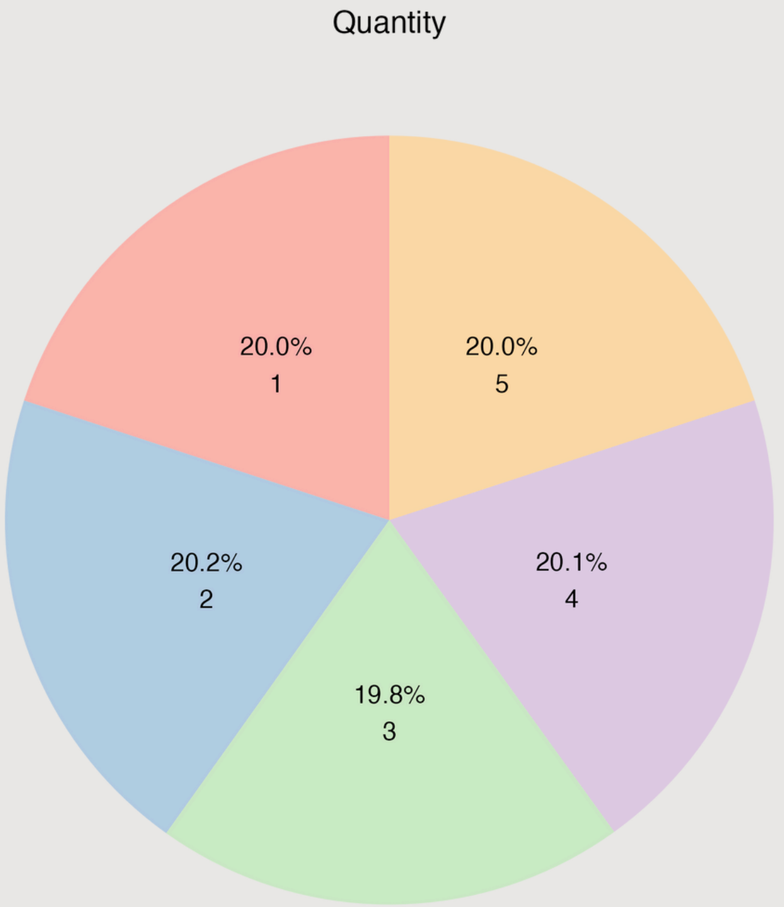


After

EDA (under sampling)

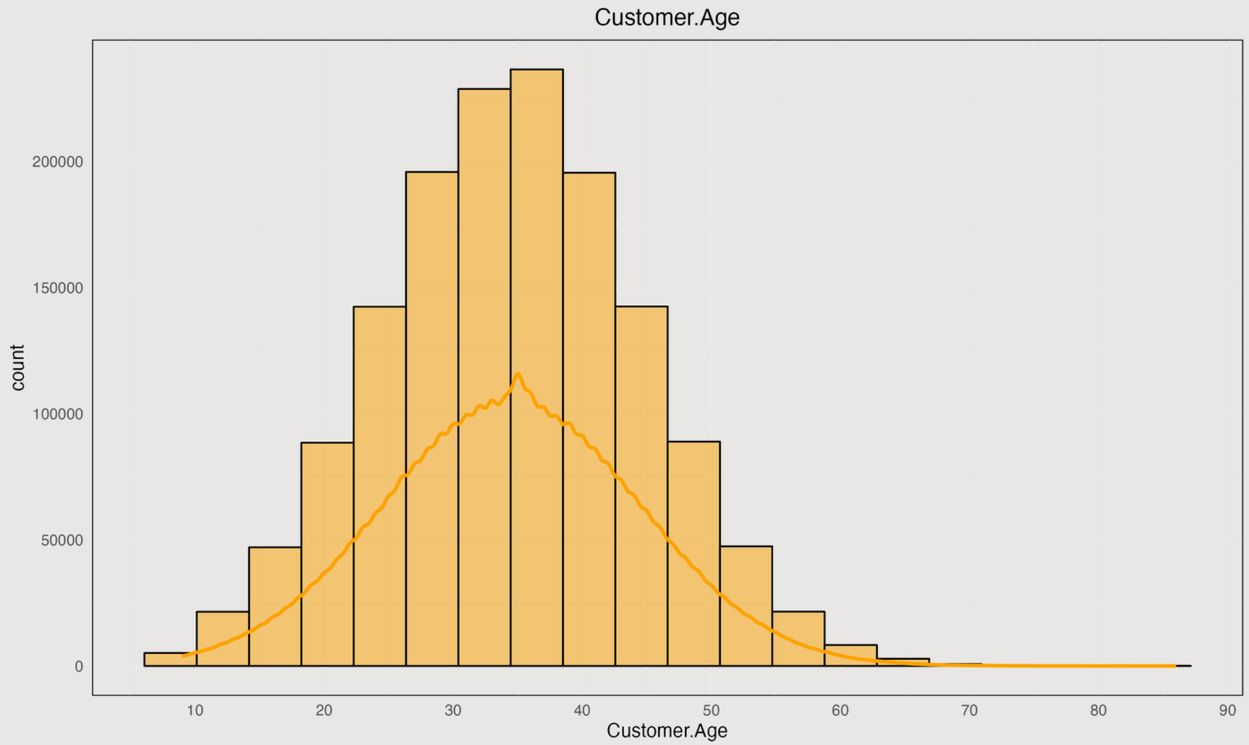


Before

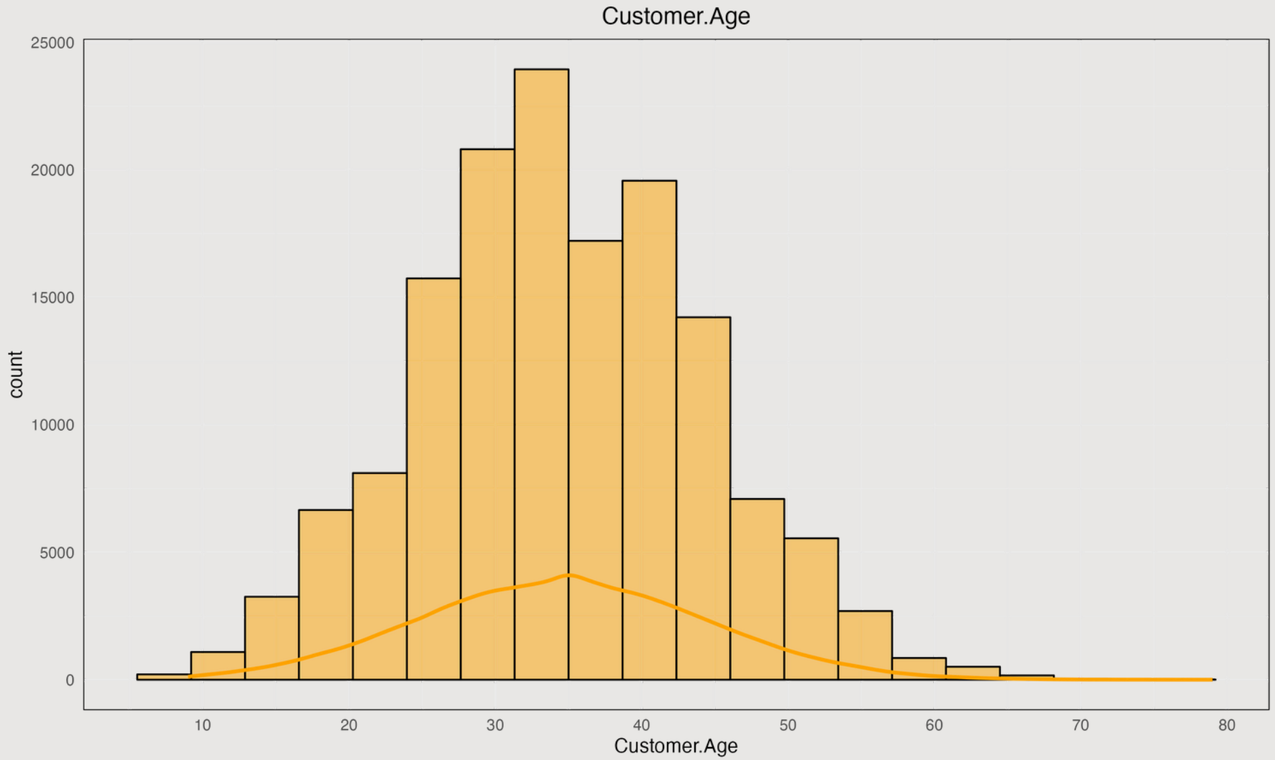


After

EDA (under sampling) - 客戶資訊

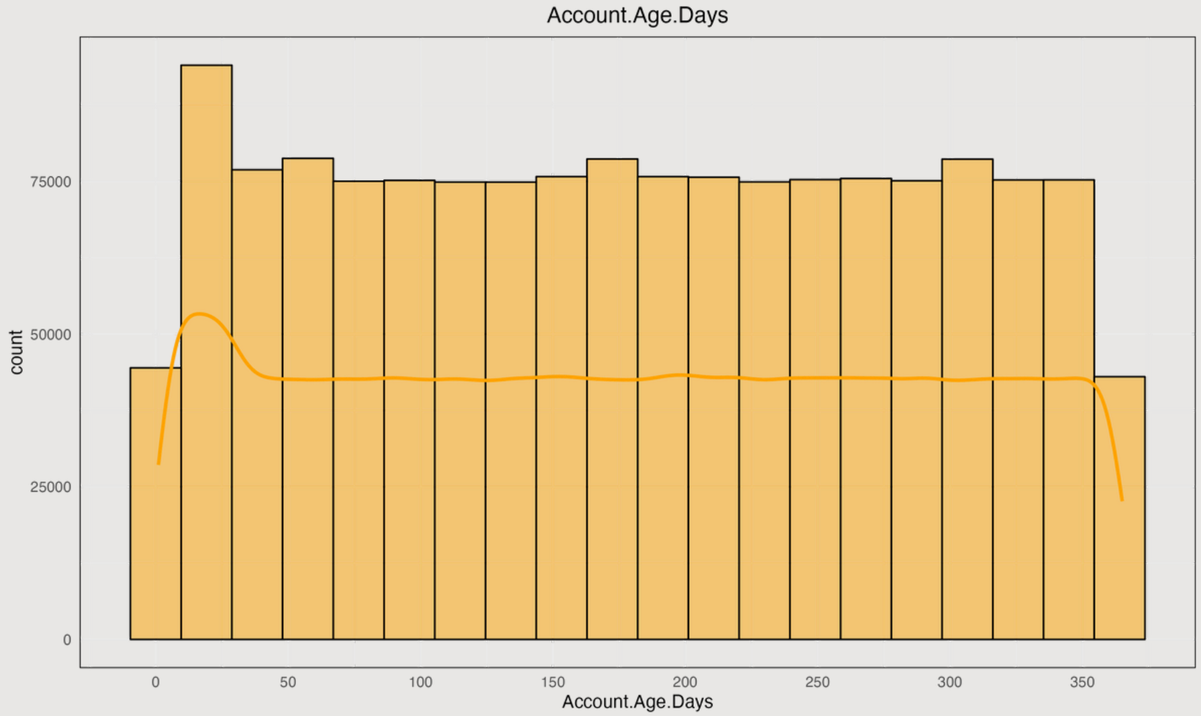


Before

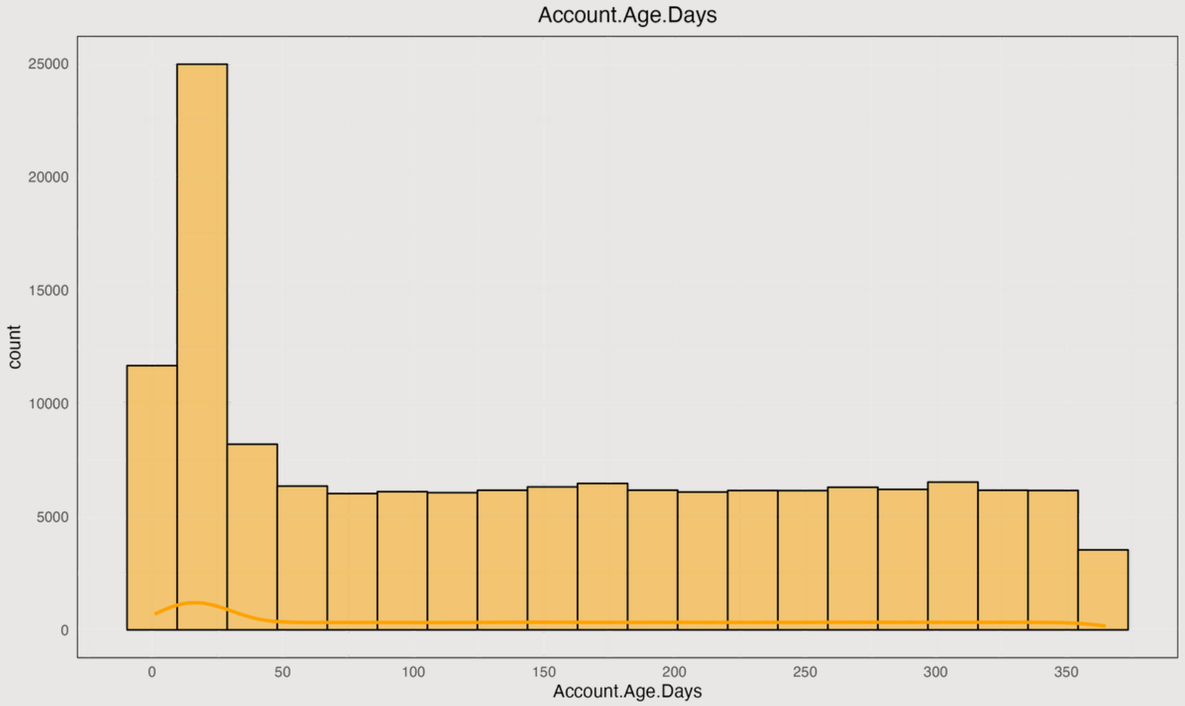


After

EDA (under sampling) - 客戶資訊

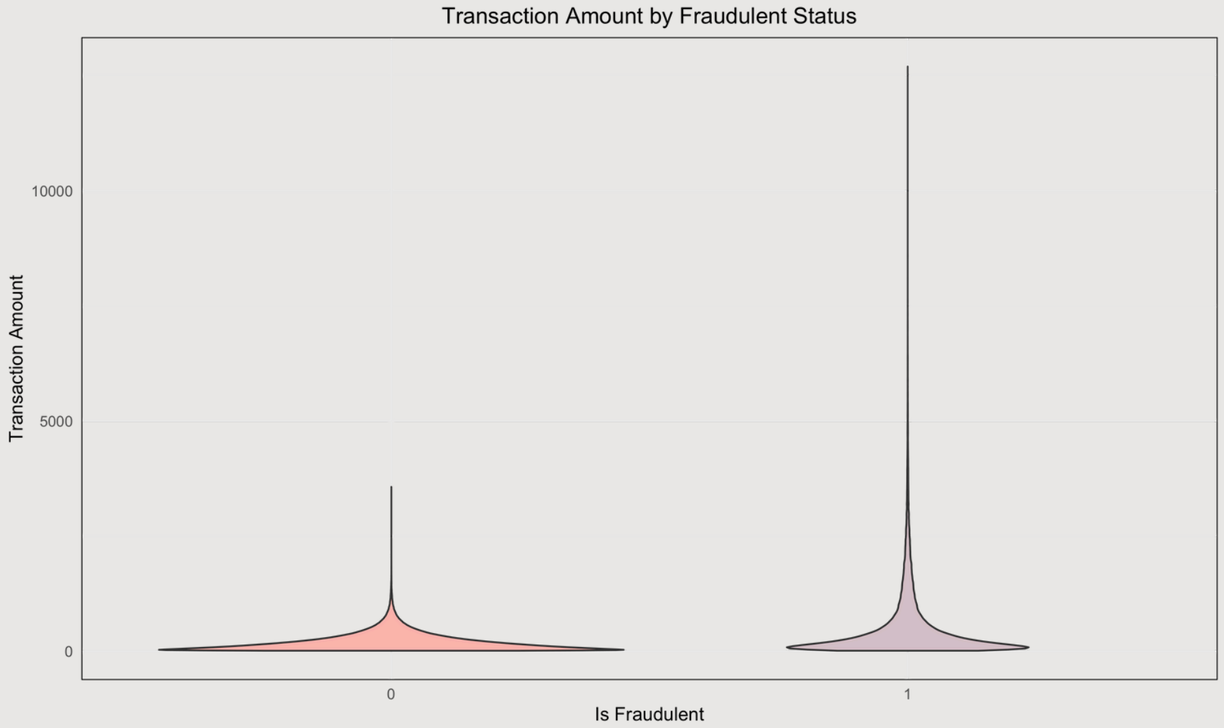


Before

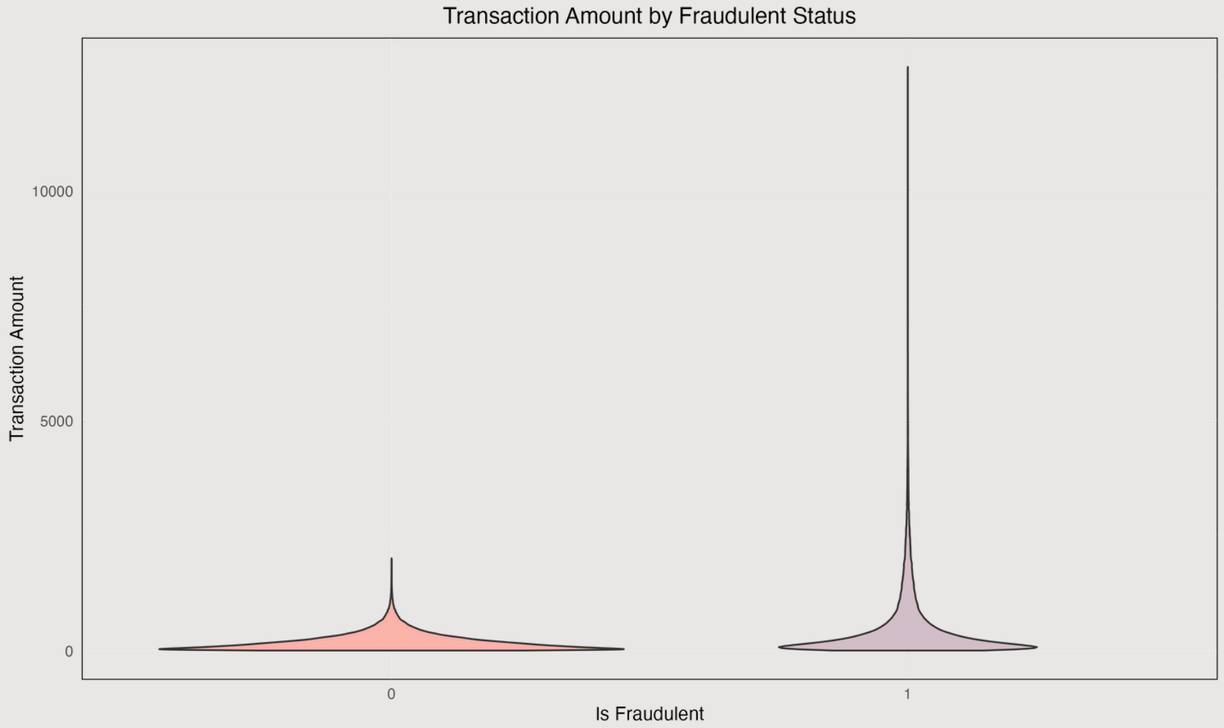


After

EDA(under sampling) - Is Fraudulent

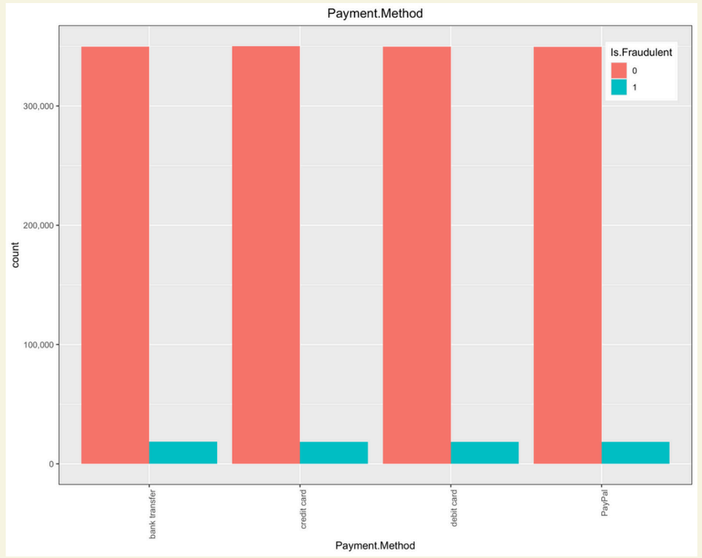


Before

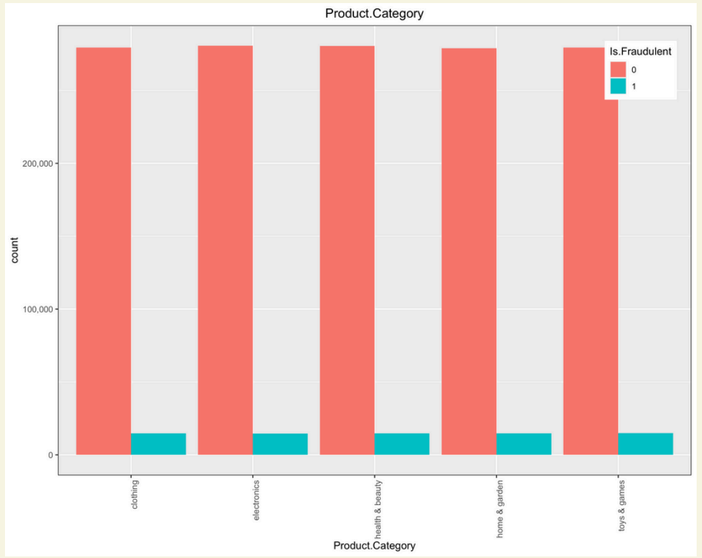


After

EDA (under sampling) - which is Fraudulent

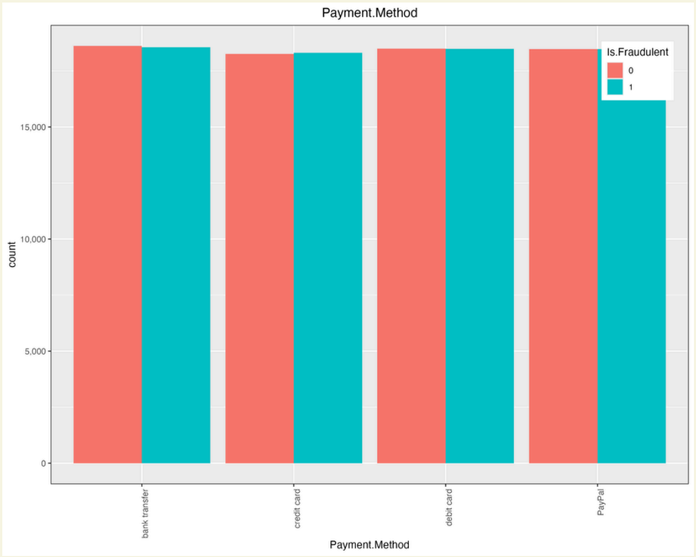


Before

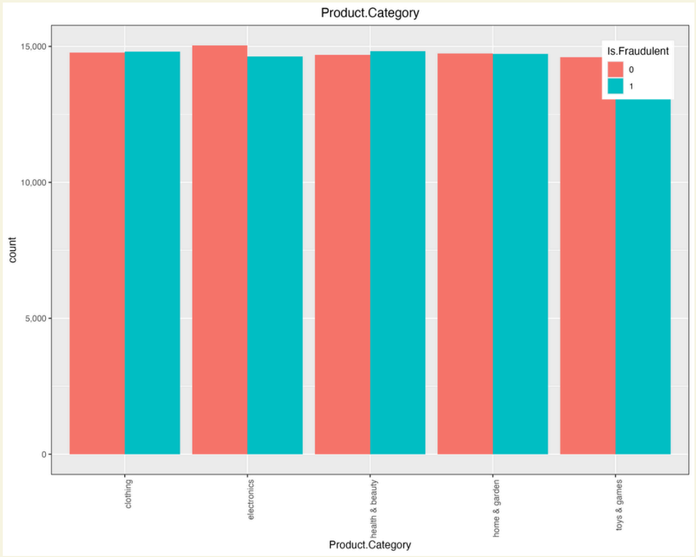


Before

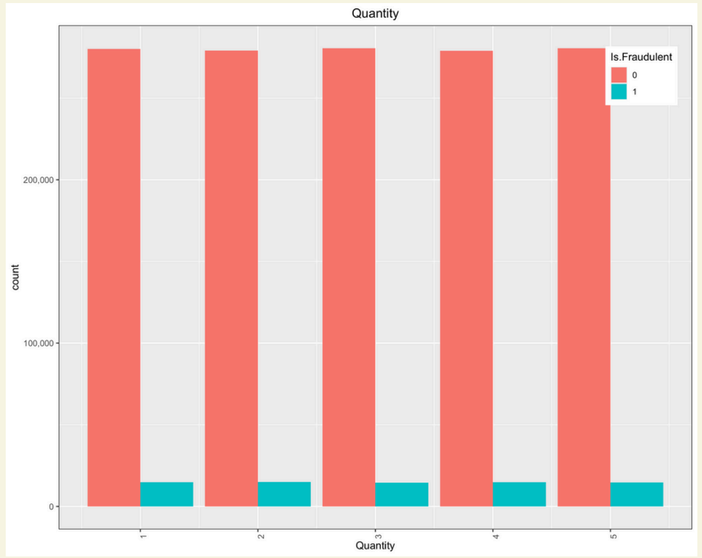
After



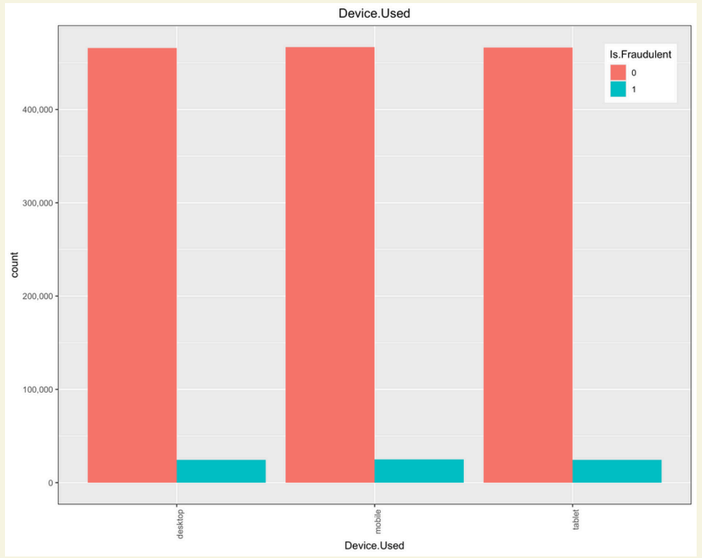
After



EDA (under sampling) - which is Fraudulent

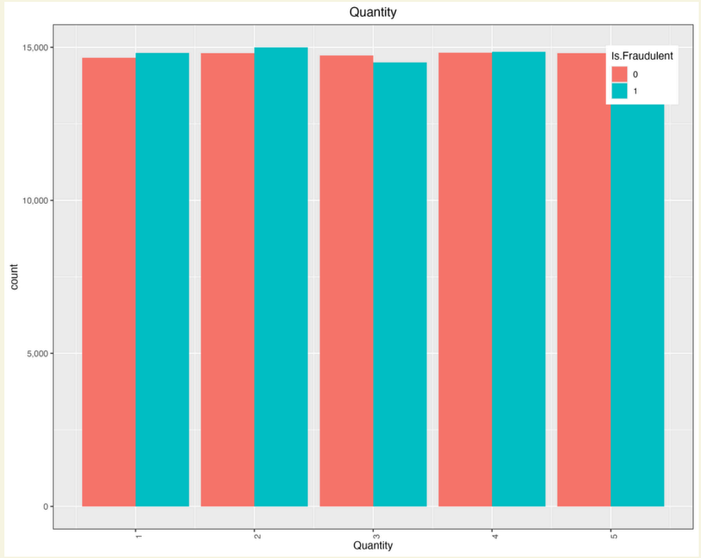


Before

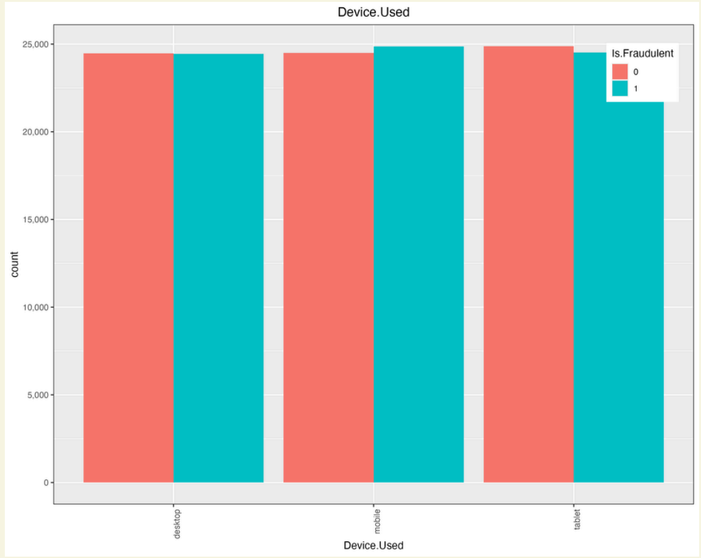


Before

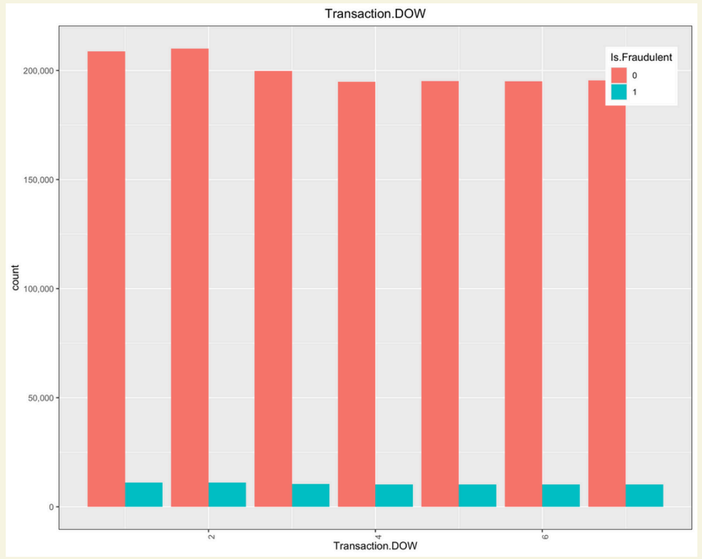
After



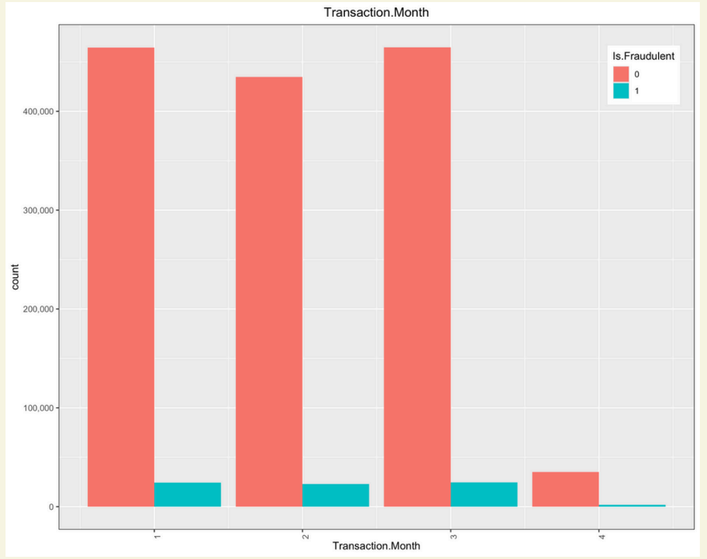
After



EDA (under sampling) - which is Fraudulent

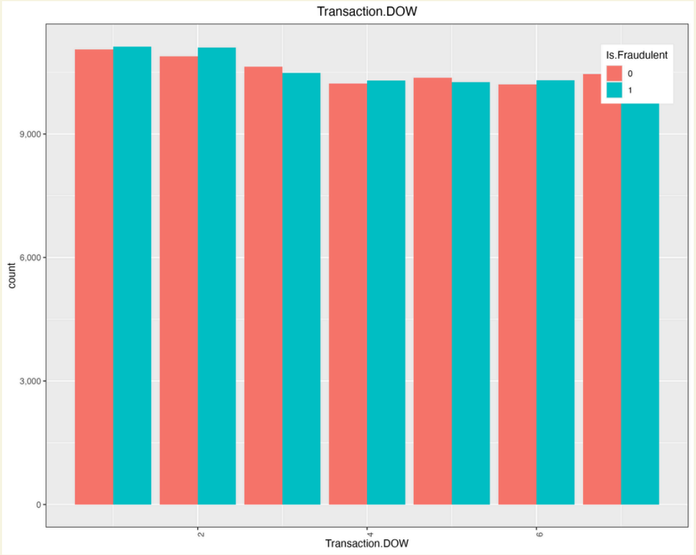


Before

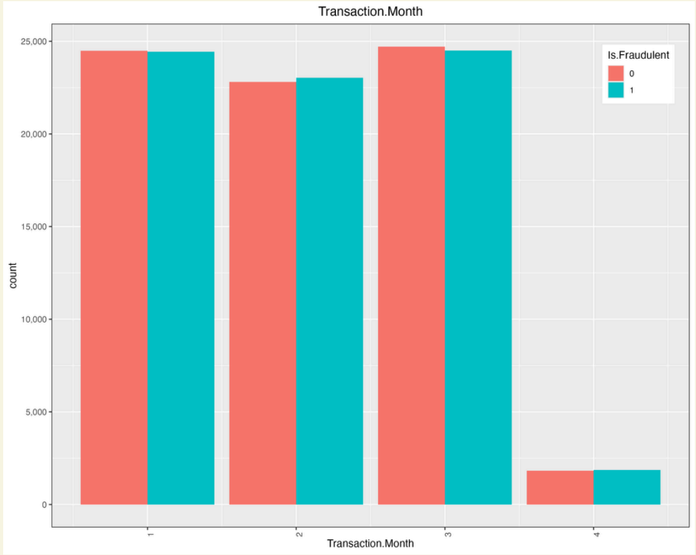


Before

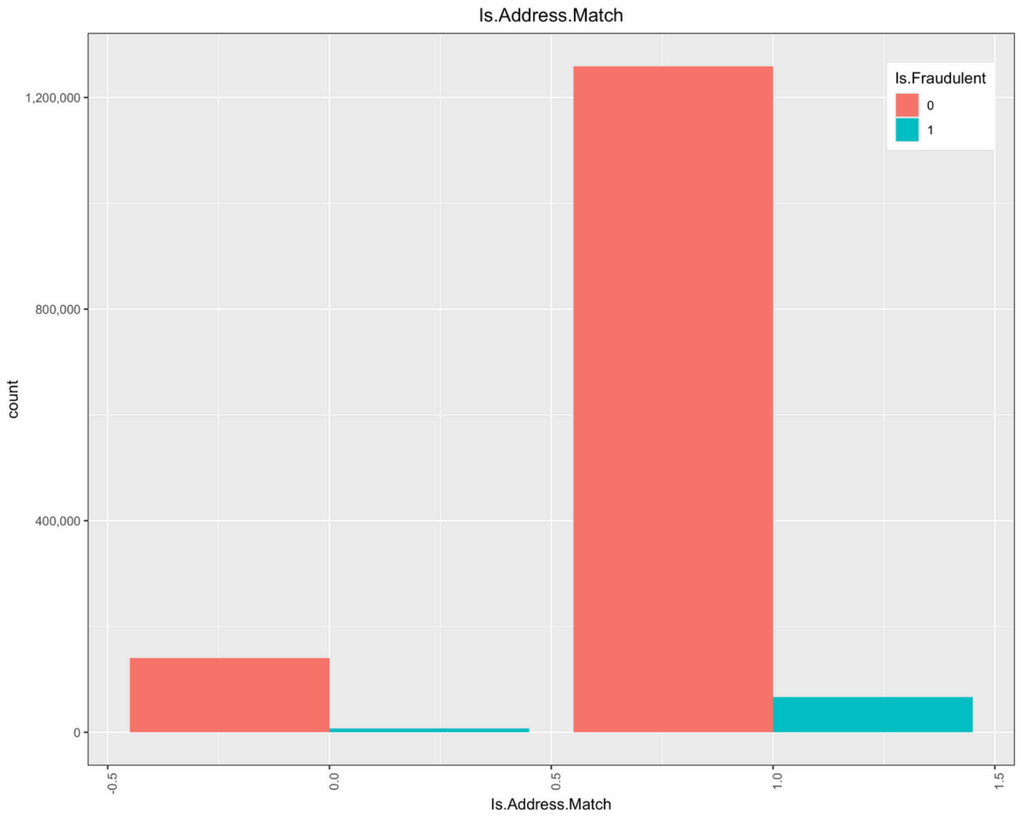
After



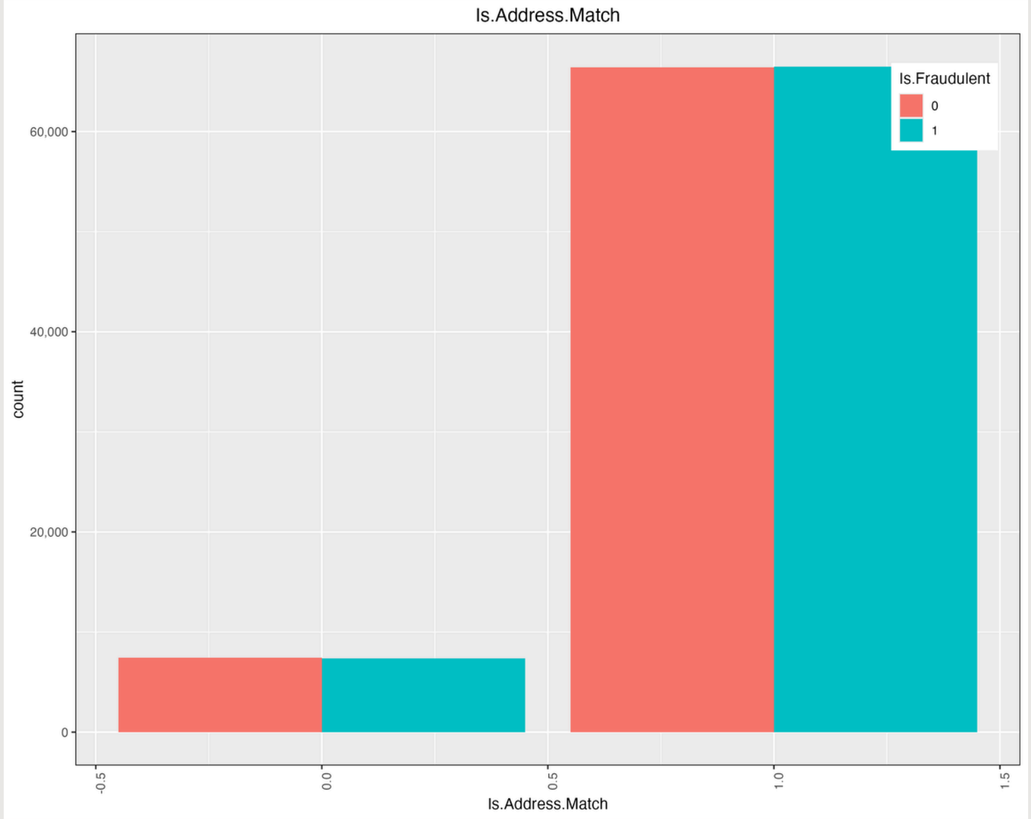
After



EDA (under sampling) - which is Fraudulent



Before



After

Models

Decision Tree

Random Forest

Logistic Regression

SVM

KNN

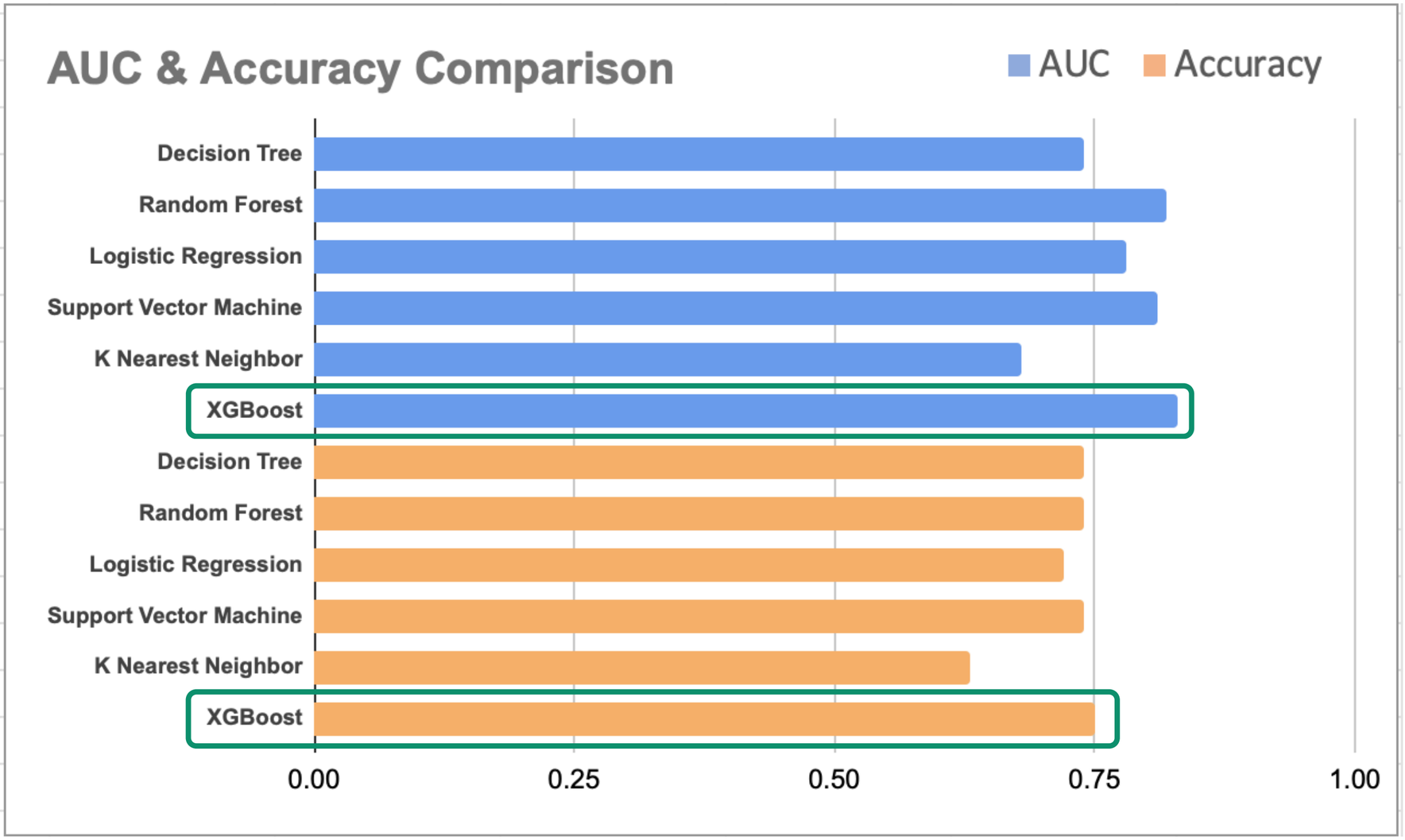
XGBoost

Models Comparison

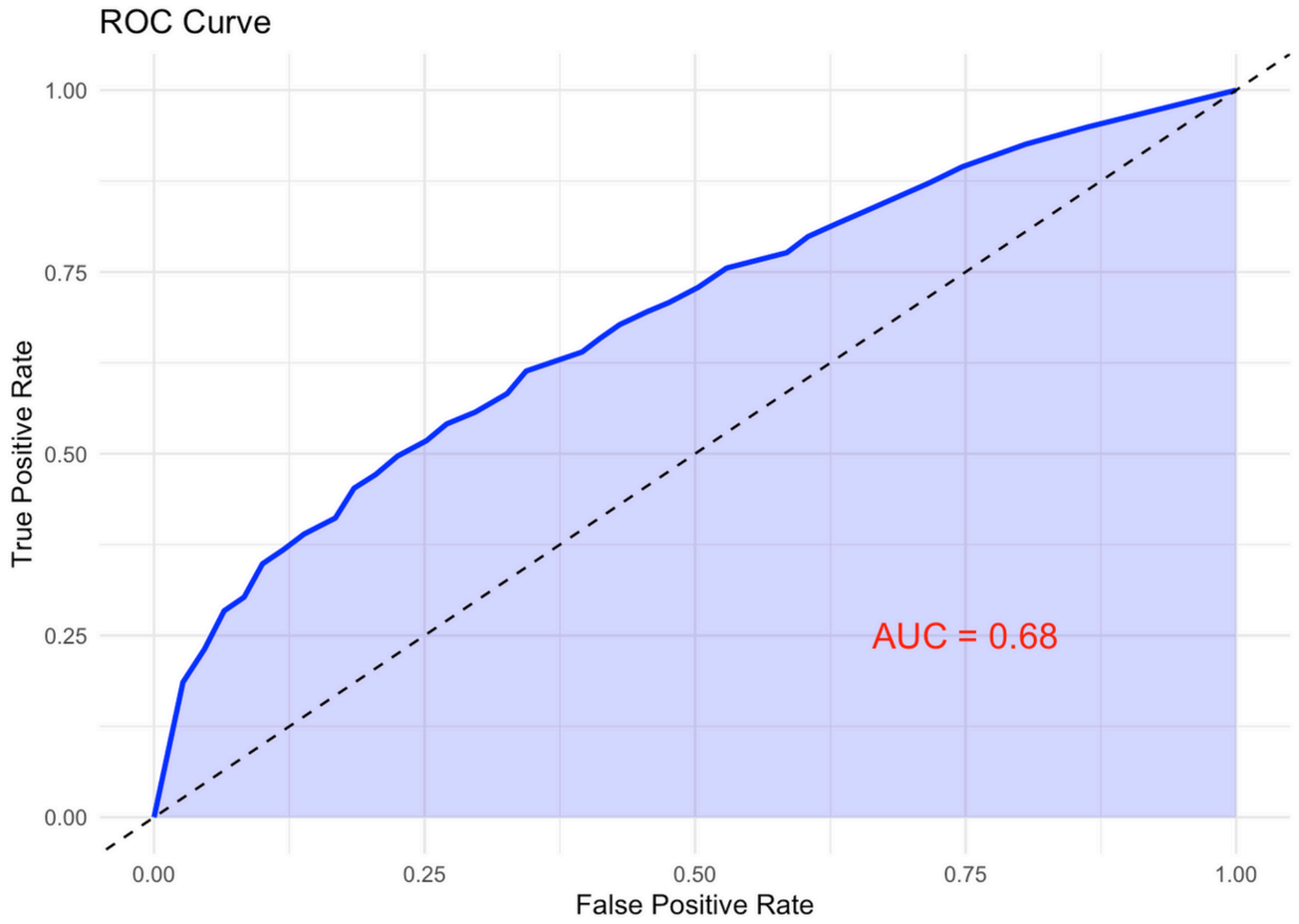
Testing Data

	Decision Tree	Random Forest	Logistic Regression	Support Vector Machine	K Nearest Neighbor	XGBoost
Sensitivity	0.90	0.78	0.72	0.79	0.66	0.78
Specificity	0.57	0.71	0.69	0.68	0.61	0.73
Precision	0.97	0.98	0.98	0.98	0.97	0.98
Recall	0.90	0.78	0.72	0.79	0.66	0.78
F1	0.94	0.87	0.83	0.87	0.78	0.87
Accuracy	0.74	0.74	0.72	0.74	0.63	0.75
AUC	0.74	0.82	0.78	0.81	0.68	0.83

Models Comparison

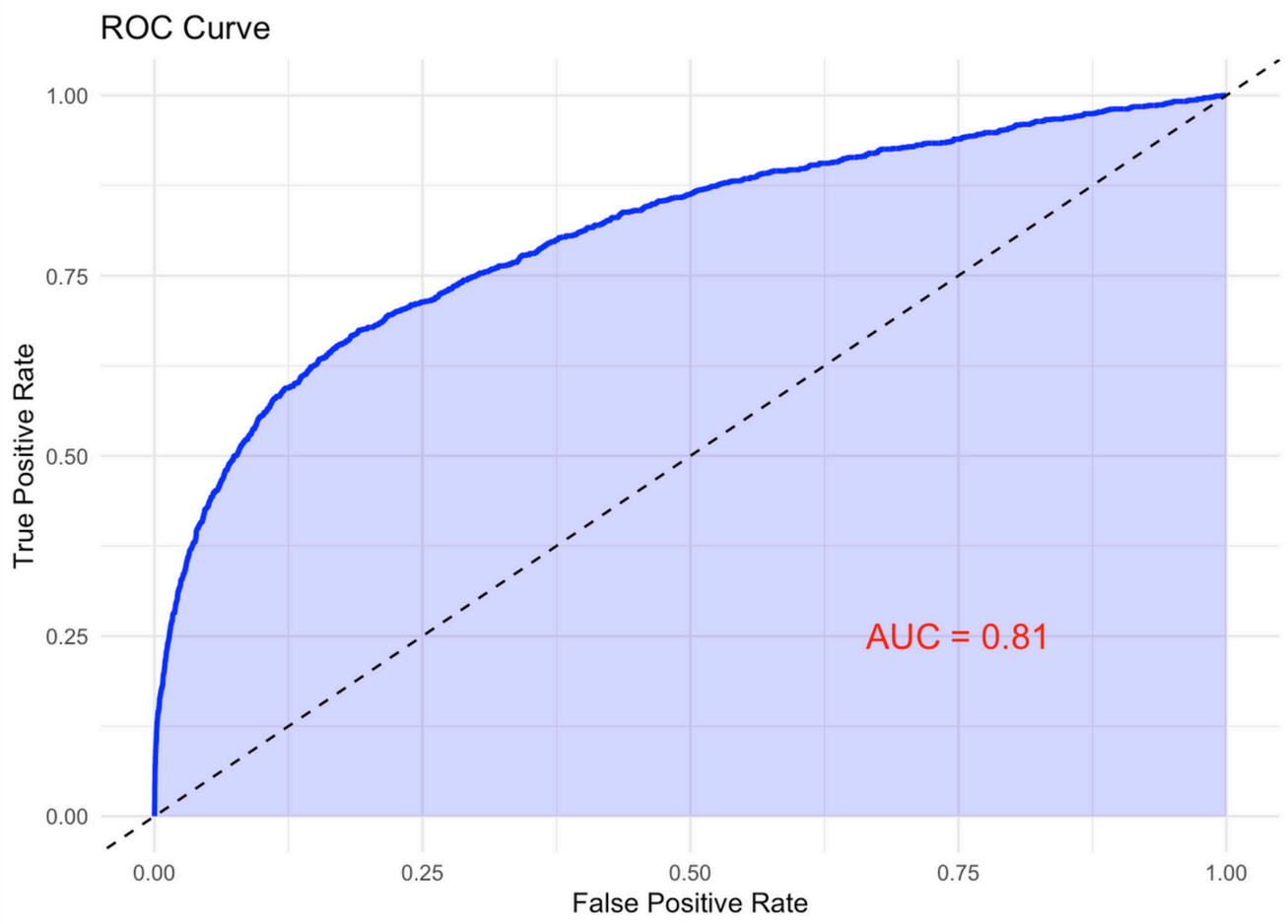


KNN



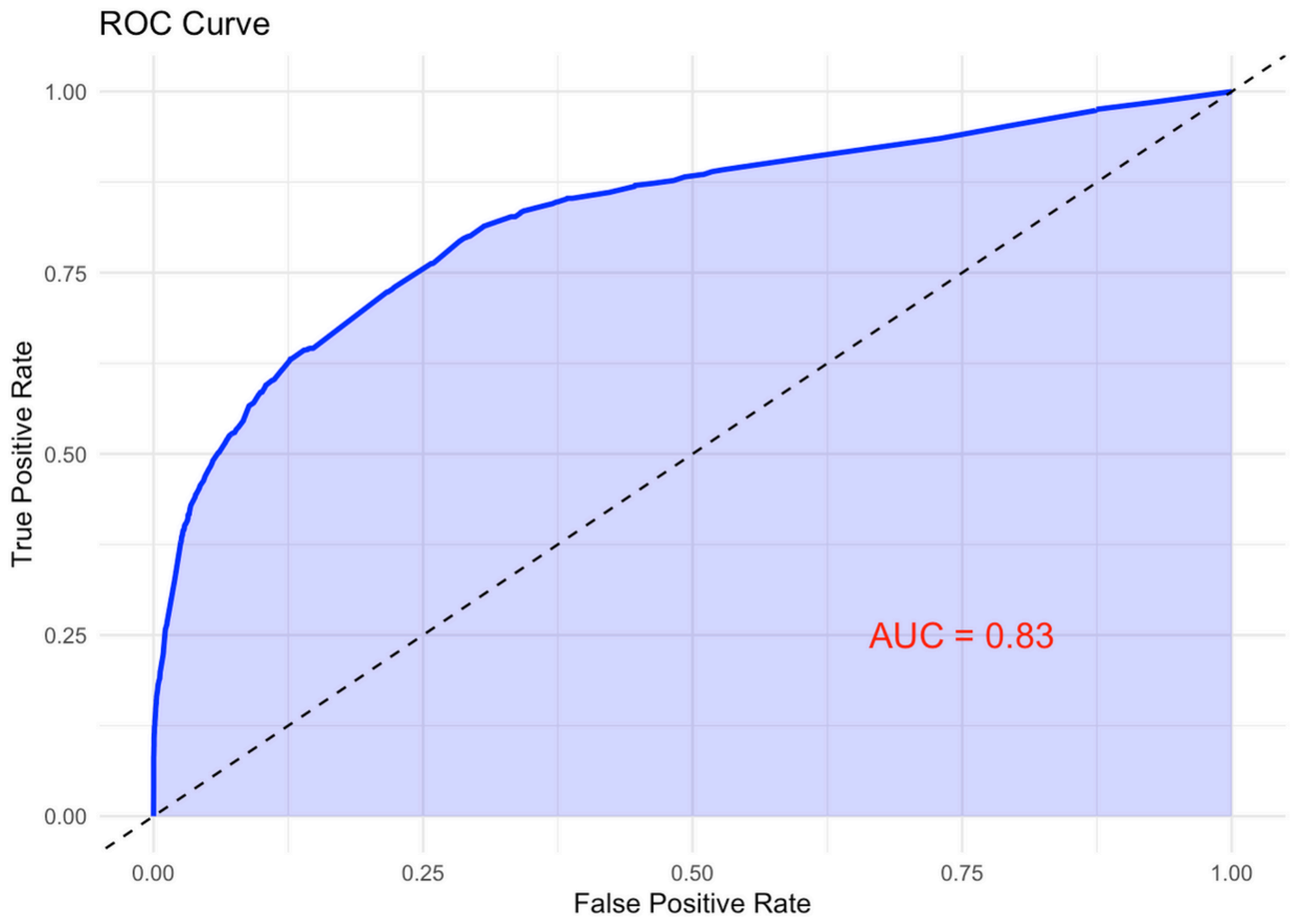
- Accuracy : 0.6349
- Sensitivity : 0.6561
- Specificity : 0.6138
- Precision : 0.9689
- F1 : 0.7824

SVM



- Accuracy : 0.7370
- Sensitivity : 0.7899
- Specificity : 0.6841
- Precision : 0.9787
- F1 : 0.8742

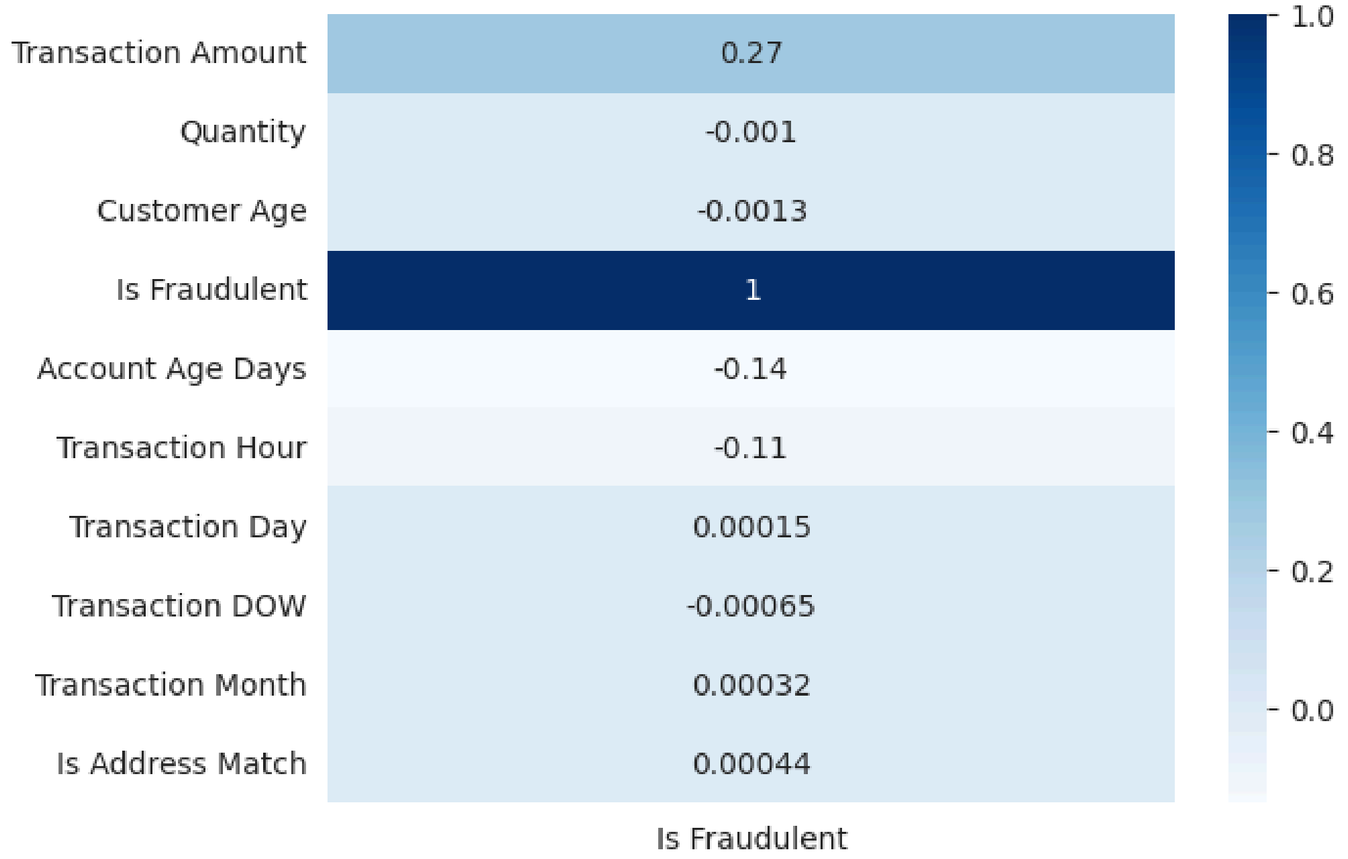
XGBoost



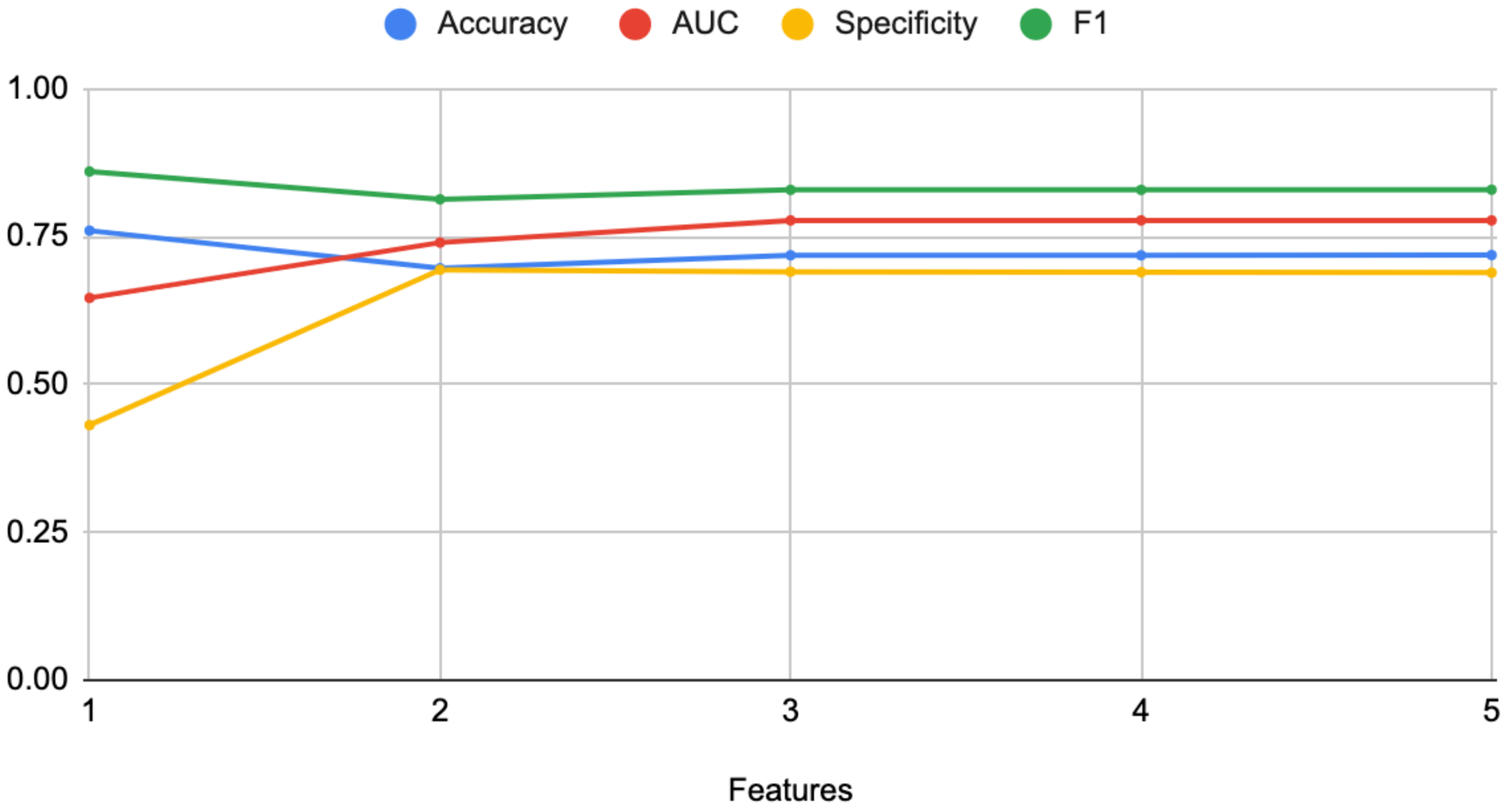
- Accuracy : 0.7533
- Sensitivity : 0.7848
- Specificity : 0.7218
- Precision : 0.9810
- F1 : 0.8720

Feature Selection

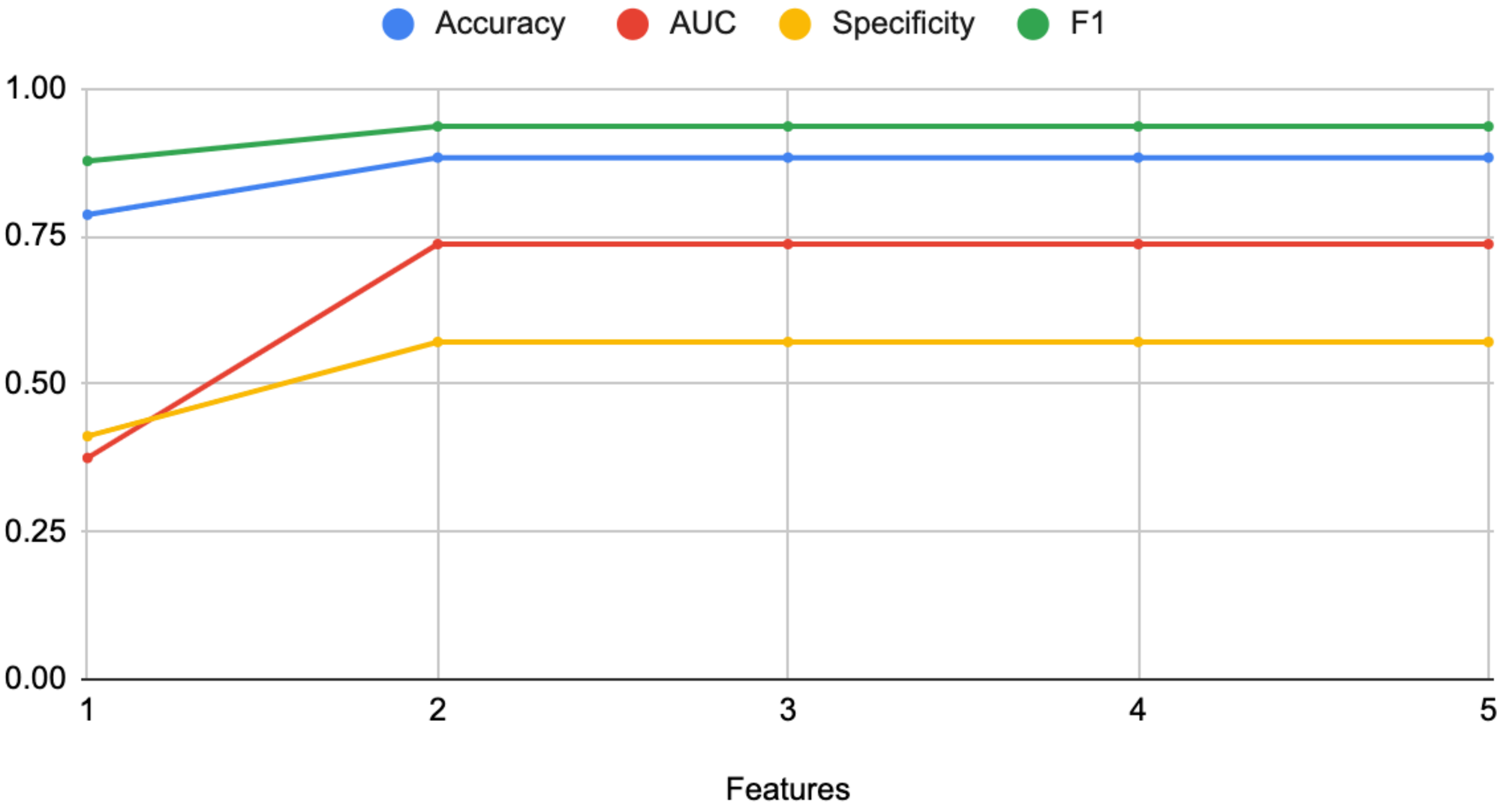
- 希望可以在維持準確度的情況下節省更多時間



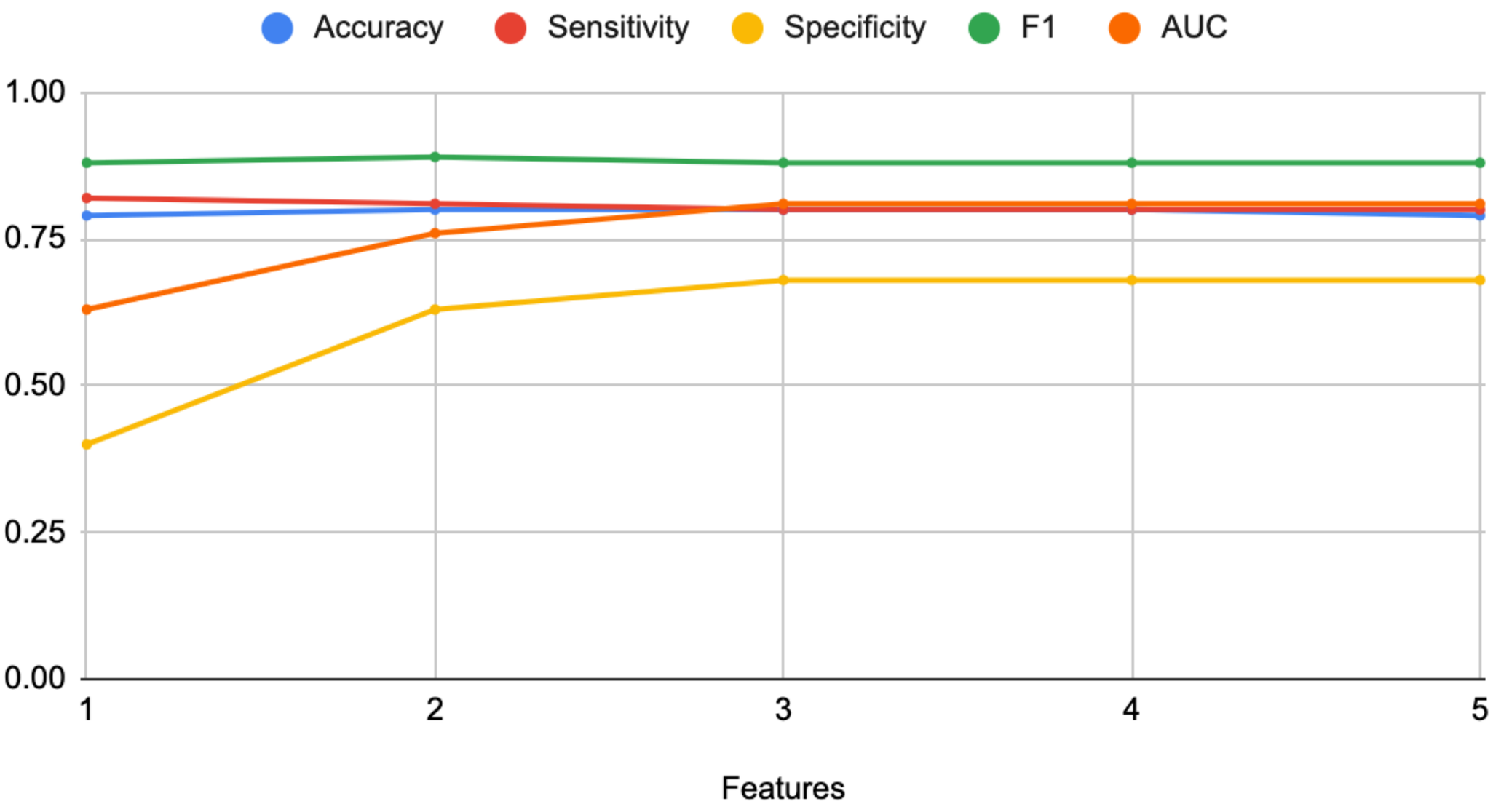
Logistic Regression Result



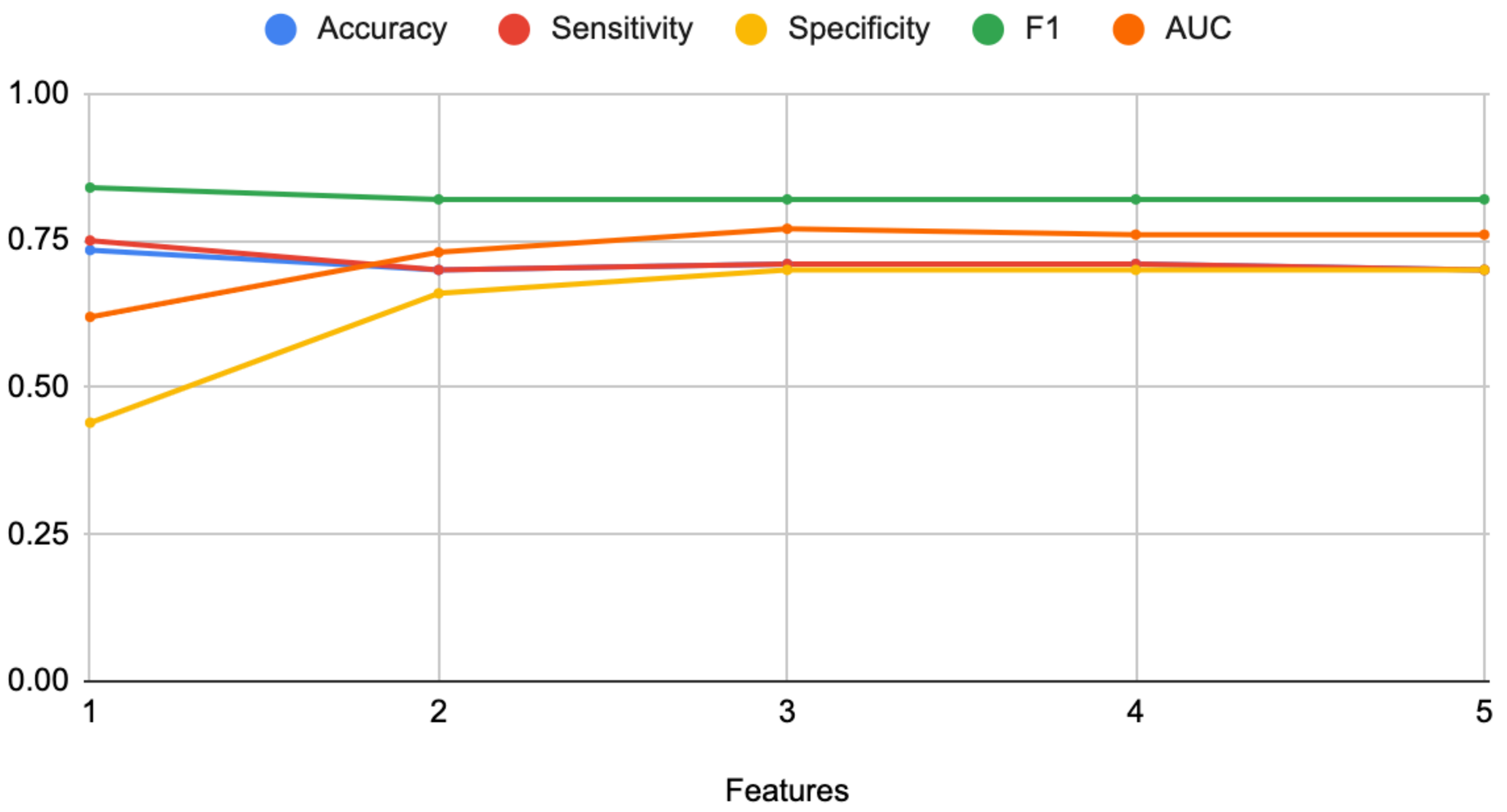
Decision Tree Result



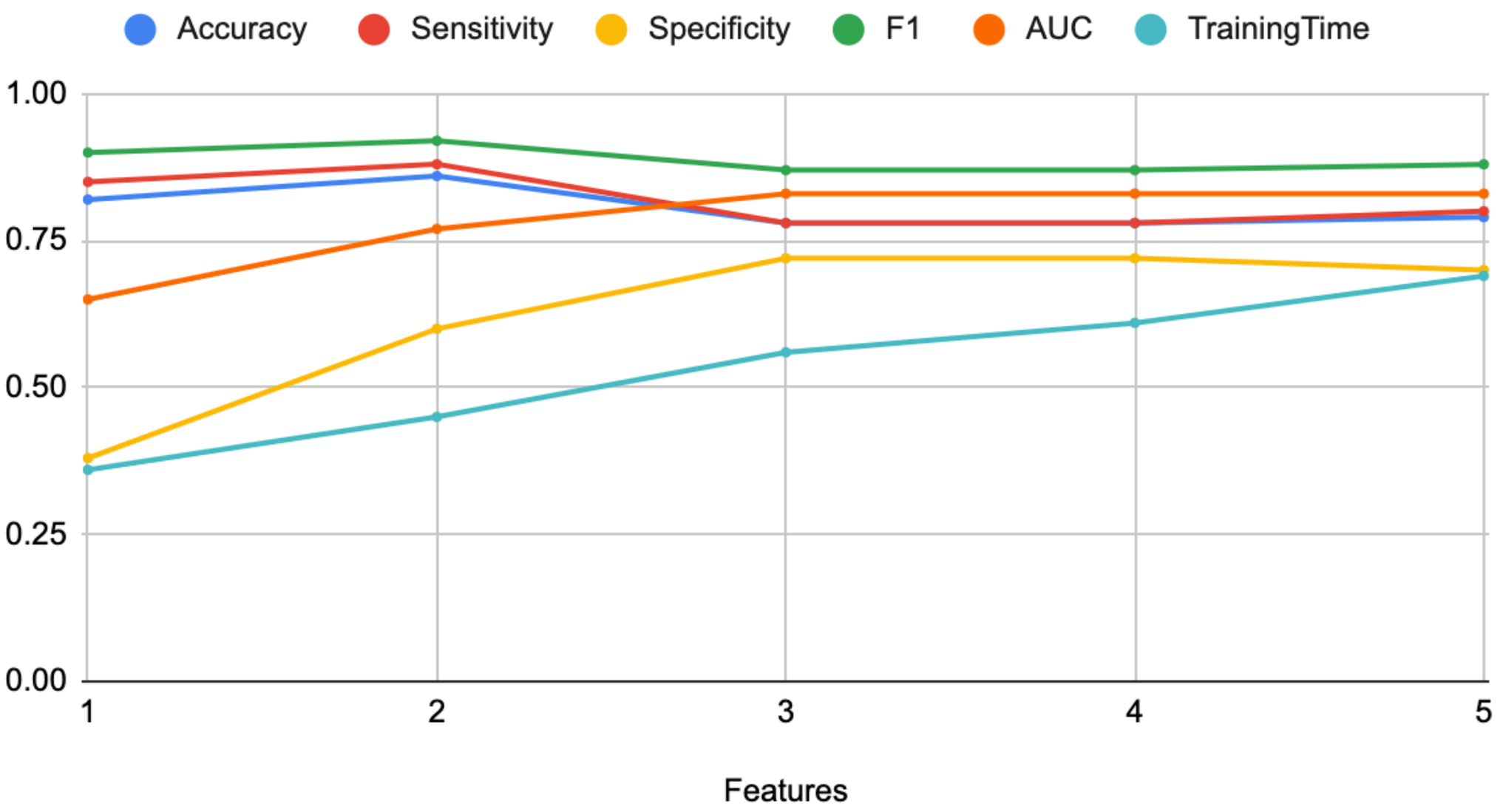
SVM Results


Accuracy
Sensitivity
Specificity
F1
AUC
1.00
0.75
0.50
0.25
0.00
1
2
3
4
5
Features

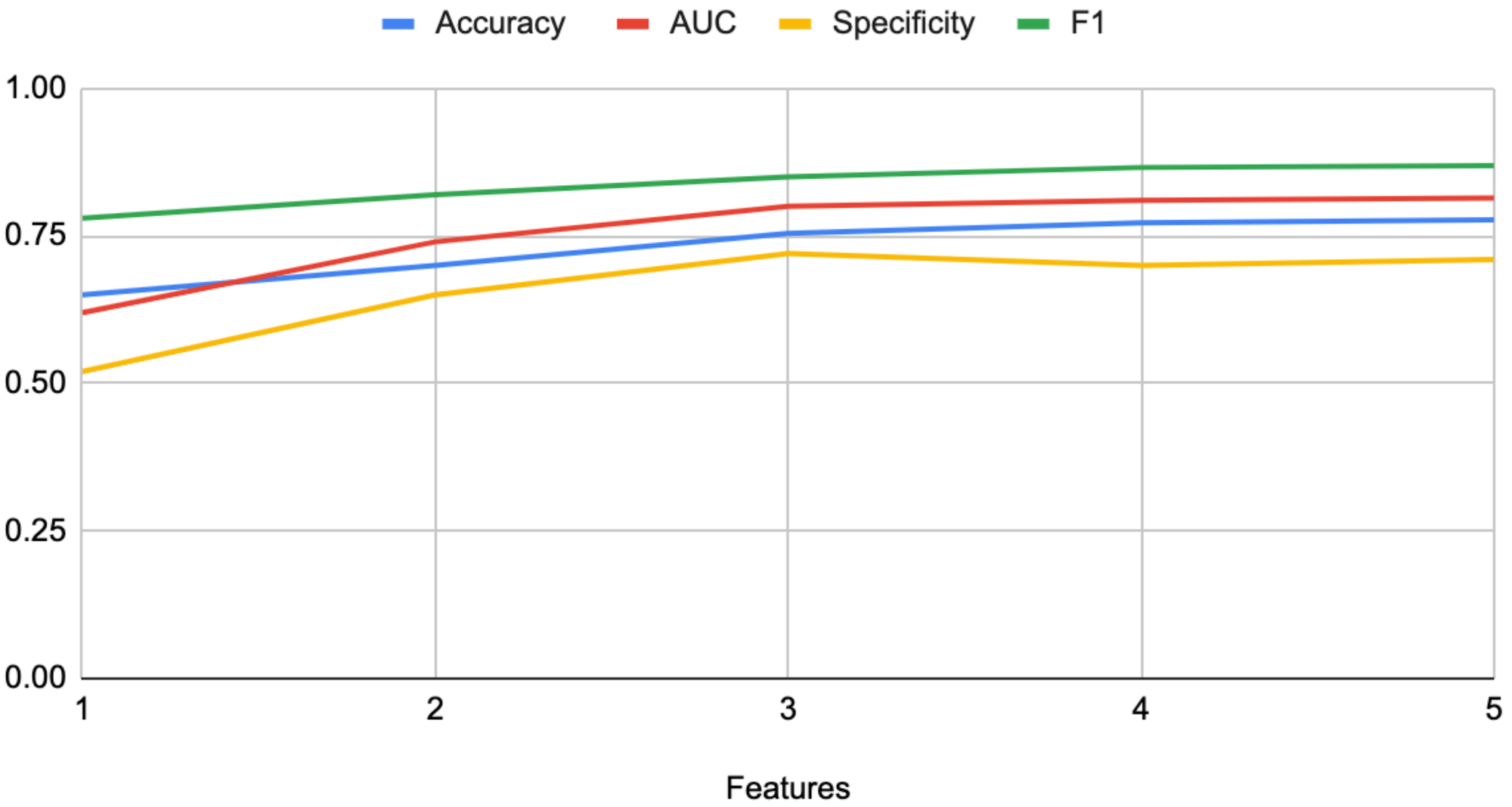
KNN Featue Selection Results



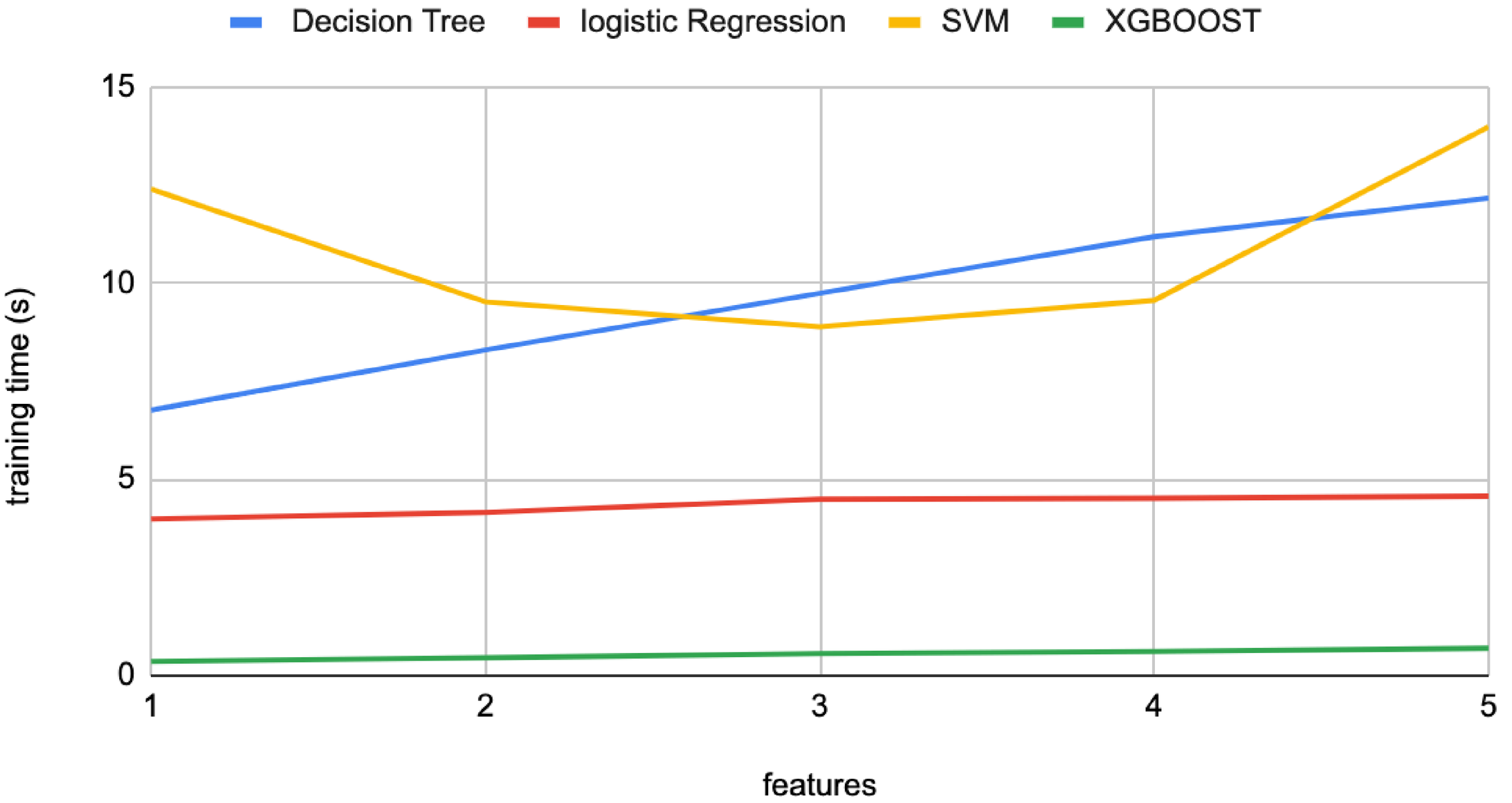
XGBOOST Result



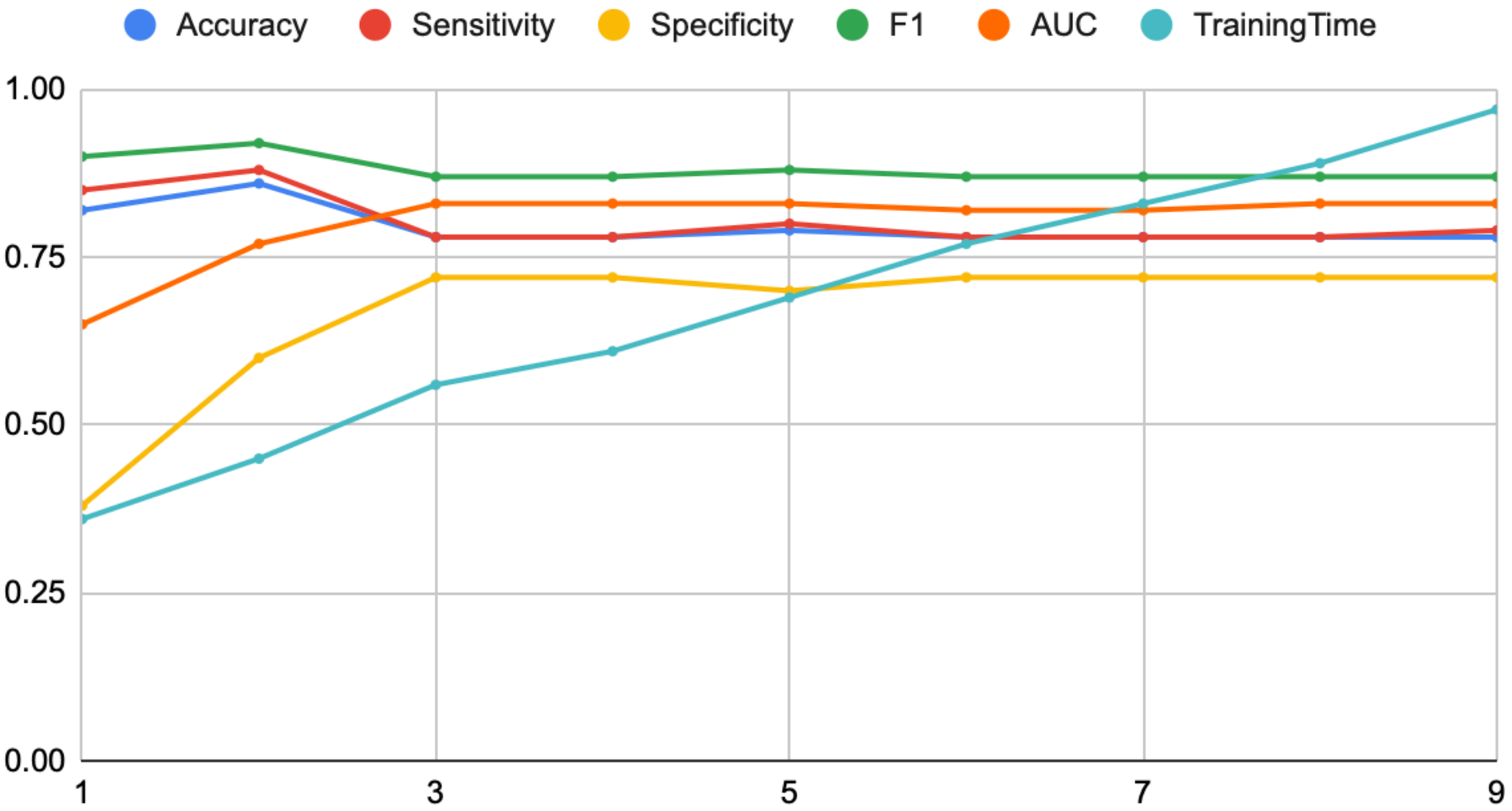
Random Forest Result



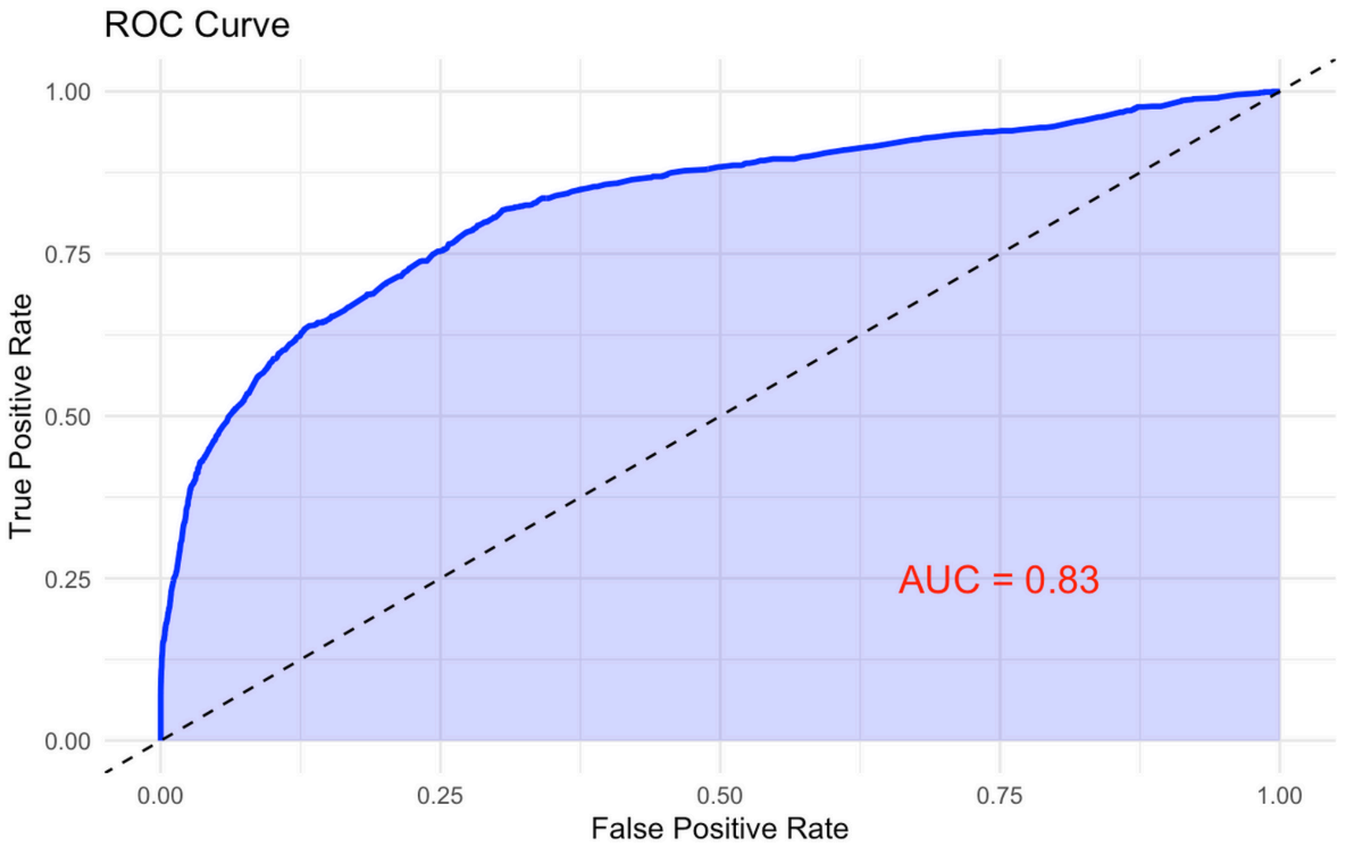
Training times - Features



XGBoost Results(all features)



XGBoost - 3 features



- Accuracy : 0.7513
- Sensitivity : 0.7834
- Specificity : 0.7193
- Precision : 0.9808
- F1 : 0.8711

XGBoost Comparison

Testing Data

	XGBoost	XGBoost grid search	XGBoost 3 features	XGBoost 3 features grid search
Sensitivity	0.7848	0.7784	0.7834	0.7774
Specificity	0.7218	0.7275	0.7193	0.7283
Precision	0.9810	0.9813	0.9808	0.9813
Recall	0.7848	0.7784	0.7834	0.7774
F1	0.8720	0.8681	0.8711	0.8676
Accuracy	0.7533	0.7529	0.7513	0.7529
AUC	0.8265	0.8263	0.8257	0.8263

海報展演 - 人家給予回饋

1. **此類預測問題還可以應用在其他領域？**
    - 回答：
2. **哪些欄位有特別做處理？**
    - 回答：
3. **是否有計算平均單價？**
    - 回答：

海報展演 - 別組有趣事物

- **整體活動回饋**
  - 餐盒吃蠻飽（雖然有點乾）
- **其他組別**
  - 【香蕉甜度分析】
    - 透過香蕉採收數據去推測哪個Feature跟甜度最為相關，並提供現場試吃評比 → 有趣且實用

謝謝聆聽！

祝各位暑假愉快！