ikayspada

January 28, 2025

[30]:
```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.ensemble import GradientBoostingClassifier
from sklearn.naive_bayes import GaussianNB
from sklearn.metrics import accuracy_score, classification_report,␣
 ↪confusion_matrix
from sklearn.preprocessing import StandardScaler
from sklearn.svm import SVC
```

[9]:
```
# Load data in chunks
chunk_size = 100000  # Adjust based on your system's capacity
chunks = pd.read_csv(r"C:\Users\91703\Downloads\Online fraud detection.csv",␣
 ↪chunksize=chunk_size)

# Example: Process chunks
for chunk in chunks:
    # Perform operations on each chunk
    print(chunk.head())
```

```
   step      type    amount     nameOrig  oldbalanceOrg  newbalanceOrig  \
0     1   PAYMENT   9839.64  C1231006815       170136.0       160296.36
1     1   PAYMENT   1864.28  C1666544295        21249.0        19384.72
2     1  TRANSFER    181.00  C1305486145          181.0            0.00
3     1  CASH_OUT    181.00   C840083671          181.0            0.00
4     1   PAYMENT  11668.14  C2048537720        41554.0        29885.86

      nameDest  oldbalanceDest  newbalanceDest  isFraud  isFlaggedFraud
0  M1979787155             0.0             0.0        0               0
1  M2044282225             0.0             0.0        0               0
2   C553264065             0.0             0.0        1               0
3    C38997010         21182.0             0.0        1               0
4  M1230701703             0.0             0.0        0               0
        step      type     amount     nameOrig  oldbalanceOrg  newbalanceOrig  \
100000    10  CASH_OUT  578570.34  C1842020389           0.00            0.00
100001    10   PAYMENT    3948.26   C924595278           0.00            0.00
100002    10   PAYMENT    1990.85   C164024007       10117.51         8126.67
100003    10   PAYMENT    3233.27  C1783742575        8126.67         4893.40
```

```
100004    10   CASH_IN   43526.60   C508964499       36522.00        80048.60

           nameDest  oldbalanceDest  newbalanceDest  isFraud  isFlaggedFraud
100000  C1750350389       782692.89      1735465.73        0               0
100001  M1742045918            0.00            0.00        0               0
100002  M1018772978            0.00            0.00        0               0
100003   M559512713            0.00            0.00        0               0
100004  C1329428979        32187.91            0.00        0               0
        step      type     amount     nameOrig  oldbalanceOrg  newbalanceOrig  \
200000    13  CASH_OUT  170910.69   C397273470        50508.0            0.00
200001    13  CASH_OUT  411781.30  C1651841579        14847.0            0.00
200002    13  TRANSFER  311671.87  C1987041070       320374.0         8702.13
200003    13  CASH_OUT  406452.75  C1407065300       134694.0            0.00
200004    13  CASH_OUT  283680.73   C725678536          738.0            0.00

           nameDest  oldbalanceDest  newbalanceDest  isFraud  isFlaggedFraud
200000  C2032266039       638862.00       809772.69        0               0
200001  C2052619908       153460.45       565241.76        0               0
200002   C842729010     11400000.00     11700000.00        0               0
200003  C1393541714      2456275.00      2862727.75        0               0
200004  C1857797782            0.00       283680.73        0               0
        step     type     amount     nameOrig  oldbalanceOrg  newbalanceOrig  \
300000    15  CASH_IN   34640.51   C474135841     19300000.0      19300000.0
300001    15  CASH_IN  103462.32  C1401686864     19300000.0      19400000.0
300002    15  CASH_IN  169777.99   C789004993     19400000.0      19600000.0
300003    15  CASH_IN   11414.85   C593058111     19600000.0      19600000.0
300004    15  CASH_IN   65303.45   C712721344     19600000.0      19700000.0

           nameDest  oldbalanceDest  newbalanceDest  isFraud  isFlaggedFraud
300000   C748261222      1460718.26      1426077.75        0               0
300001  C1080314178     15200000.00     15000000.00        0               0
300002  C2028054542       914481.28       744703.28        0               0
300003   C728526866     14100000.00     14100000.00        0               0
300004  C1849844563      1104029.73      1038726.27        0               0
        step     type    amount     nameOrig  oldbalanceOrg  newbalanceOrig  \
400000    18  PAYMENT  16921.87  C1618320367       37556.77        20634.91
400001    18  PAYMENT   7999.89  C1365367196       20634.91        12635.02
400002    18  PAYMENT   5123.27   C859580447       12635.02         7511.76
400003    18  PAYMENT  26718.54  C1915842836        7511.76            0.00
400004    18  PAYMENT  17987.27   C203344781           0.00            0.00

           nameDest  oldbalanceDest  newbalanceDest  isFraud  isFlaggedFraud
400000  M1075568478             0.0             0.0        0               0
400001  M2145025768             0.0             0.0        0               0
400002   M767815689             0.0             0.0        0               0
400003  M1580235908             0.0             0.0        0               0
400004  M1055859135             0.0             0.0        0               0
        step      type     amount     nameOrig  oldbalanceOrg  newbalanceOrig  \
```

```
500000    20  TRANSFER  115755.54  C2082926317           0.0            0.00
500001    20  TRANSFER  164495.57   C643883428           0.0            0.00
500002    20  TRANSFER  224561.64  C1478505186           0.0            0.00
500003    20   PAYMENT   11646.97   C214088711       13423.0         1776.03
500004    20   PAYMENT   10650.89  C1721032314      111476.0       100825.11

          nameDest  oldbalanceDest  newbalanceDest  isFraud  isFlaggedFraud
500000  C1974060990      1919834.28      2035589.82        0               0
500001   C444030714       242061.23       406556.80        0               0
500002  C1874297237       583738.52       808300.16        0               0
500003  M1196976884            0.00            0.00        0               0
500004  M1064978280            0.00            0.00        0               0
        step      type     amount     nameOrig  oldbalanceOrg  newbalanceOrig  \
600000    34   PAYMENT    3430.30  C1640600216         2038.0            0.00
600001    34  CASH_OUT  152778.01  C1080485411          301.0            0.00
600002    34   PAYMENT    5195.79  C1451065354            0.0            0.00
600003    34   PAYMENT   20415.34   C119593188            0.0            0.00
600004    34  CASH_OUT   36609.99  C1254605335       140419.0       103809.01

          nameDest  oldbalanceDest  newbalanceDest  isFraud  isFlaggedFraud
600000   M369970057            0.00            0.00        0               0
600001   C660819982      5137572.10      5290350.12        0               0
600002  M2064487695            0.00            0.00        0               0
600003  M1887461853            0.00            0.00        0               0
600004  C1284671693       229232.73       265842.72        0               0
        step      type     amount     nameOrig  oldbalanceOrg  newbalanceOrig  \
700000    37  CASH_OUT   32564.97  C1795743497            0.0             0.0
700001    37  CASH_OUT  196783.20  C1892197917            0.0             0.0
700002    37  CASH_OUT  251282.06   C947878236            0.0             0.0
700003    37  CASH_OUT  249402.40   C807678130            0.0             0.0
700004    37  CASH_OUT  462517.96   C502878504            0.0             0.0

          nameDest  oldbalanceDest  newbalanceDest  isFraud  isFlaggedFraud
700000  C1336975682       334400.15       366965.13        0               0
700001   C203924698      1476264.93      1891988.23        0               0
700002   C502348426       509223.50       760505.56        0               0
700003   C779978956      9146499.05      9395901.45        0               0
700004  C1875902341       494413.28       956931.23        0               0
        step      type     amount     nameOrig  oldbalanceOrg  newbalanceOrig  \
800000    40  CASH_OUT  355952.68   C290597282            0.0             0.0
800001    40  CASH_OUT   52011.89    C42546182            0.0             0.0
800002    40  CASH_OUT  235401.17  C1330922069            0.0             0.0
800003    40  CASH_OUT  171667.24   C683572274            0.0             0.0
800004    40  CASH_OUT  118855.30   C681660350            0.0             0.0

          nameDest  oldbalanceDest  newbalanceDest  isFraud  isFlaggedFraud
800000  C1140165110       991554.27      1347506.96        0               0
800001   C350610730       191109.02       243120.91        0               0
```

```
800002   C772406271      8453205.24      8688606.41        0               0
800003   C487249044       612491.22       784158.46        0               0
800004  C1584207980      1060154.74      1179010.03        0               0
        step      type     amount     nameOrig  oldbalanceOrg  newbalanceOrig  \
900000    42  CASH_OUT   29974.17  C1093386650            0.0             0.0
900001    42  CASH_OUT   89342.61  C1803662865            0.0             0.0
900002    42  CASH_OUT  208406.41  C1403250352            0.0             0.0
900003    42  CASH_OUT  129956.91   C510282095            0.0             0.0
900004    42  CASH_OUT  404178.28    C22182953            0.0             0.0

           nameDest  oldbalanceDest  newbalanceDest  isFraud  isFlaggedFraud
900000  C1373693818       410728.54       440702.72        0               0
900001  C1753703703      1491766.10      1581108.71        0               0
900002   C270574759       991670.91      1339949.56        0               0
900003  C2132781443       580075.09       710032.00        0               0
900004  C1628692256      2091231.23      2495409.51        0               0
         step      type    amount     nameOrig  oldbalanceOrg  newbalanceOrig  \
1000000    45   PAYMENT   2277.01  C2064115396      501040.13       498763.12
1000001    45   PAYMENT  10381.85  C1886401979      498763.12       488381.27
1000002    45  TRANSFER  76382.84   C977137811       30486.00            0.00
1000003    45   PAYMENT   2570.11  C1436921762           0.00            0.00
1000004    45   PAYMENT  11467.26   C440255297           0.00            0.00

            nameDest  oldbalanceDest  newbalanceDest  isFraud  isFlaggedFraud
1000000    M80574281            0.00            0.00        0               0
1000001  M2118654040            0.00            0.00        0               0
1000002  C2016131934      2301199.88      2377582.73        0               0
1000003   M604607808            0.00            0.00        0               0
1000004  M1339097597            0.00            0.00        0               0
```

[11]:
```
chunk.describe()
```

[11]:
```
                step        amount  oldbalanceOrg  newbalanceOrig  \
count  48575.000000  4.857500e+04   4.857500e+04    4.857500e+04
mean      64.223757  1.260793e+05   7.985808e+05    8.003276e+05
std       22.760428  3.611171e+05   2.741417e+06    2.768839e+06
min       45.000000  9.700000e-01   0.000000e+00    0.000000e+00
25%       46.000000  5.922295e+03   0.000000e+00    0.000000e+00
50%       48.000000  1.862079e+04   1.707000e+04    2.008260e+03
75%       94.000000  1.463953e+05   1.009150e+05    9.411986e+04
max       95.000000  1.000000e+07   3.220000e+07    3.240000e+07

       oldbalanceDest  newbalanceDest       isFraud  isFlaggedFraud
count    4.857500e+04    4.857500e+04  48575.000000         48575.0
mean     8.153901e+05    8.780915e+05      0.012496             0.0
std      2.104390e+06    2.186579e+06      0.111087             0.0
min      0.000000e+00    0.000000e+00      0.000000             0.0
```

```
25%      0.000000e+00    0.000000e+00      0.000000            0.0
50%      0.000000e+00    0.000000e+00      0.000000            0.0
75%      6.801030e+05    7.594912e+05      0.000000            0.0
max      3.480000e+07    3.560000e+07      1.000000            0.0
```

[12]:
```
chunk.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 48575 entries, 1000000 to 1048574
Data columns (total 11 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   step            48575 non-null  int64
 1   type            48575 non-null  object
 2   amount          48575 non-null  float64
 3   nameOrig        48575 non-null  object
 4   oldbalanceOrg   48575 non-null  float64
 5   newbalanceOrig  48575 non-null  float64
 6   nameDest        48575 non-null  object
 7   oldbalanceDest  48575 non-null  float64
 8   newbalanceDest  48575 non-null  float64
 9   isFraud         48575 non-null  int64
 10  isFlaggedFraud  48575 non-null  int64
dtypes: float64(5), int64(3), object(3)
memory usage: 4.1+ MB
```

[13]:
```
#Normalize numerical columns
numeric_cols = ['amount', 'oldbalanceOrg', 'newbalanceOrig', 'oldbalanceDest',␣
 ↪'newbalanceDest']
chunk[numeric_cols] = chunk[numeric_cols].apply(lambda x: (x - x.min()) / (x.
 ↪max() - x.min()))
```

[14]:
```
chunk.head()
```

[14]:

	step	type	amount	nameOrig	oldbalanceOrg	newbalanceOrig	\
1000000	45	PAYMENT	0.000228	C2064115396	0.015560	0.015394	
1000001	45	PAYMENT	0.001038	C1886401979	0.015490	0.015073	
1000002	45	TRANSFER	0.007638	C977137811	0.000947	0.000000	
1000003	45	PAYMENT	0.000257	C1436921762	0.000000	0.000000	
1000004	45	PAYMENT	0.001147	C440255297	0.000000	0.000000	

	nameDest	oldbalanceDest	newbalanceDest	isFraud	isFlaggedFraud
1000000	M80574281	0.000000	0.000000	0	0
1000001	M2118654040	0.000000	0.000000	0	0
1000002	C2016131934	0.066126	0.066786	0	0
1000003	M604607808	0.000000	0.000000	0	0
1000004	M1339097597	0.000000	0.000000	0	0

[15]:
```
#Encode categorical columns
chunk['type'] = chunk['type'].map({'CASH_OUT': 0, 'CASH_IN': 1, 'DEBIT': 2,␣
 ↪'PAYMENT': 3, 'TRANSFER': 4})
chunk['nameOrig'] = chunk['nameOrig'].astype('category').cat.codes
chunk['nameDest'] = chunk['nameDest'].astype('category').cat.codes
```

[16]:
```
chunk.head()
```

[16]:

	step	type	amount	nameOrig	oldbalanceOrg	newbalanceOrig	\
1000000	45	3	0.000228	26623	0.015560	0.015394	
1000001	45	3	0.001038	22339	0.015490	0.015073	
1000002	45	4	0.007638	47990	0.000947	0.000000	
1000003	45	3	0.000257	10981	0.000000	0.000000	
1000004	45	3	0.001147	34348	0.000000	0.000000	

	nameDest	oldbalanceDest	newbalanceDest	isFraud	isFlaggedFraud
1000000	42953	0.000000	0.000000	0	0
1000001	35152	0.000000	0.000000	0	0
1000002	11114	0.066126	0.066786	0	0
1000003	40446	0.000000	0.000000	0	0
1000004	25651	0.000000	0.000000	0	0

[17]:
```
#Split data into features (X) and target variable (y)
X = chunk.drop('isFraud', axis=1)
y = chunk['isFraud']
```

[18]:
```
#Split data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,␣
 ↪random_state=42)
```

[19]:
```
#Scale data using StandardScaler
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)
```

[20]:
```
#Define SVM classifier
svm_model = SVC()

#Train SVM model
svm_model.fit(X_train_scaled, y_train)
```

[20]:
```
SVC()
```

[21]:
```
y_pred_svm = svm_model.predict(X_test_scaled)
print(y_pred_svm)
```

```
[0 0 0 … 0 0 0]
```

[22]:
```
#Evaluate model performance
print("SVM Model Performance:")
print("Accuracy:", accuracy_score(y_test, y_pred_svm))
print("Classification Report:", classification_report(y_test, y_pred_svm))
print("Confusion Matrix:", confusion_matrix(y_test, y_pred_svm))
```

```
SVM Model Performance:
Accuracy: 0.9975295934122491
Classification Report:                precision    recall  f1-score   support

           0       1.00      1.00      1.00      9590
           1       1.00      0.81      0.89       125

    accuracy                           1.00      9715
   macro avg       1.00      0.90      0.95      9715
weighted avg       1.00      1.00      1.00      9715

Confusion Matrix: [[9590    0]
 [  24  101]]
```

[25]:
```
# Gradient Boosting classifier #
gb_model = GradientBoostingClassifier(
n_estimators=100,
learning_rate=0.1,
max_depth=5,
random_state=42
)
#Train Gradient Boosting model
gb_model.fit(X_train, y_train)
```

[25]:
```
GradientBoostingClassifier(max_depth=5, random_state=42)
```

[26]:
```
#Make predictions on test set
y_pred_gb = gb_model.predict(X_test)
print(y_pred_gb)
```

```
[0 0 0 … 0 0 0]
```

[27]:
```
#Evaluate model performance
print("Gradient Boosting Model Performance:")
print("Accuracy:", accuracy_score(y_test, y_pred_gb))
print("Classification Report:", classification_report(y_test, y_pred_gb))
print("Confusion Matrix:", confusion_matrix(y_test, y_pred_gb))
```

```
Gradient Boosting Model Performance:
Accuracy: 0.9995882655687082
Classification Report:                precision    recall  f1-score   support
```

```
           0       1.00      1.00      1.00      9590
           1       0.99      0.98      0.98       125

    accuracy                           1.00      9715
   macro avg       1.00      0.99      0.99      9715
weighted avg       1.00      1.00      1.00      9715

Confusion Matrix: [[9589    1]
 [   3  122]]
```

[31]:
```
#  Naive Bayes classifier #
nb_model = GaussianNB()
#Train Naive Bayes model
nb_model.fit(X_train, y_train)
```

[31]:
```
GaussianNB()
```

[32]:
```
#Make predictions on test set
y_pred_nb = nb_model.predict(X_test)
print(y_pred_nb)
```

```
[0 0 0 … 0 0 0]
```

[35]:
```
#Evaluate model performance
print("Naive Bayes Model Performance:")
print("Accuracy:", accuracy_score(y_test, y_pred_nb))
print("Classification Report:", classification_report(y_test, y_pred_nb))
print("Confusion Matrix:", confusion_matrix(y_test, y_pred_nb))
import warnings
warnings.filterwarnings('ignore')
```

```
Naive Bayes Model Performance:
Accuracy: 0.9871332990221308
Classification Report:               precision    recall  f1-score   support

           0       0.99      1.00      0.99      9590
           1       0.00      0.00      0.00       125

    accuracy                           0.99      9715
   macro avg       0.49      0.50      0.50      9715
weighted avg       0.97      0.99      0.98      9715

Confusion Matrix: [[9590    0]
 [ 125    0]]
```