CodSoft_intern_Project1. Credit Card Fraud Detection.ipynb

Importing the important modules

```
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score
```

0s

```
[4] # loading the dataset to a Pandas DataFrame
credit_card_data = pd.read_csv('/content/creditcard.csv')
```

1s

```
[5] # first 5 rows of the dataset
credit_card_data.head()
```

0s

	Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	...	V21	V22	
0	0	-1.359807	-0.072781	2.536347	1.378155	-0.338321	0.462388	0.239599	0.098698	0.363787	...	-0.018307	0.277838	-0.1
1	0	1.191857	0.266151	0.166480	0.448154	0.060018	-0.082361	-0.078803	0.085102	-0.255425	...	-0.225775	-0.638672	0.1
2	1	-1.358354	-1.340163	1.773209	0.379780	-0.503198	1.800499	0.791461	0.247676	-1.514654	...	0.247998	0.771679	0.9
3	1	-0.966272	-0.185226	1.792993	-0.863291	-0.010309	1.247203	0.237609	0.377436	-1.387024	...	-0.108300	0.005274	-0.1
4	2	-1.158233	0.877737	1.548718	0.403034	-0.407193	0.095921	0.592941	-0.270533	0.817739	...	-0.009431	0.798278	-0.1

5 rows × 31 columns

0s completed at 4:53 PM

CodSoft_intern_Project1. Credit Card Fraud Detection.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

RAM Disk

Files

..
sample_data
creditcard.csv

Disk 83.30 GB available

```
Class     1
dtype: int64
```

[9]
```
# distribution of legit transactions & fraudulent transactions
credit_card_data['Class'].value_counts()
```

```
0.0    29704
1.0       94
Name: Class, dtype: int64
```

This Dataset is highly unblanced

0 --> Normal Transaction

1 --> fraudulent transaction

```
# separating the data for analysis
legit = credit_card_data[credit_card_data.Class == 0]
fraud = credit_card_data[credit_card_data.Class == 1]
```

[11]
```
print(legit.shape)
print(fraud.shape)
```

```
(29704, 31)
(94, 31)
```

0s completed at 4:53 PM

CodSoft_intern_Project1. Credit Card Fraud Detection.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

+ Code + Text

RAM Disk

Files

..
sample_data
creditcard.csv

Disk 83.30 GB available

```
Name: Amount, dtype: float64
```

[13]
```
fraud.Amount.describe()
```

```
count      94.000000
mean       95.590000
std       257.920621
min         0.000000
25%         1.000000
50%         1.050000
75%        99.990000
max      1809.680000
Name: Amount, dtype: float64
```

```
# compare the values for both transactions
credit_card_data.groupby('Class').mean()
```

Class	Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	...	V20	
0.0	21422.566422	-0.184395	0.106639	0.759413	0.194690	-0.186422	0.096790	-0.096841	0.018203	0.361273	...	0.044000	-0.
1.0	19007.702128	-8.099702	6.084984	-11.565958	6.014185	-5.681925	-2.370349	-7.912202	4.043743	-2.891421	...	0.679513	0.

2 rows × 30 columns

0s completed at 4:53 PM

CodSoft_intern_Project1. Credit Card Fraud Detection.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

RAM Disk

+ Code + Text

Files

..
sample_data
creditcard.csv

Disk 83.30 GB available

```
[15] legit_sample = legit.sample(n=492)
```

Concatenating two DataFrames

```
[16] new_dataset = pd.concat([legit_sample, fraud], axis=0)
```

```
new_dataset.head()
```

	Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	...	V21	V2
6797	8563	-1.779024	-0.121922	3.389072	-0.303840	-1.187161	0.662469	-0.450036	0.307505	3.149560	...	-0.204773	0.52729
24446	33238	1.038128	-0.915699	0.819137	0.583963	-1.417737	-0.077895	-0.781279	0.153917	-0.517252	...	-0.139073	-0.10783
4403	3767	1.401566	-0.683480	-0.914975	-1.616425	1.458382	3.274707	-1.170528	0.710384	0.360121	...	-0.278178	-0.89819
22679	32378	1.098897	0.000231	-0.157568	0.407444	-0.066279	-0.719611	0.418325	-0.171220	-0.414952	...	-0.303750	-1.14495
24970	33463	0.976263	-0.118509	0.153144	1.493143	-0.389002	-0.585357	0.269744	-0.053734	0.262421	...	0.056877	0.04945

5 rows × 31 columns

```
[18] new_dataset.tail()
```

0s completed at 4:53 PM

CodSoft_intern_Project1. Credit Card Fraud Detection.ipynb

Splitting the data into Features & Targets

```
[21] X = new_dataset.drop(columns='Class', axis=1)
     Y = new_dataset['Class']
```

```
[22] print(X)
```

```
        Time        V1        V2        V3        V4        V5        V6  \
6797    8563 -1.779024 -0.121922  3.389072 -0.303840 -1.187161  0.662469
24446  33238  1.038128 -0.915699  0.819137  0.583963 -1.417737 -0.077895
4403    3767  1.401566 -0.683480 -0.914975 -1.616425  1.458382  3.274707
22679  32378  1.098897  0.000231 -0.157568  0.407444 -0.066279 -0.719611
24970  33463  0.976263 -0.118509  0.153144  1.493143 -0.389002 -0.585357
...      ...       ...       ...       ...       ...       ...       ...
27362  34521  1.081234  0.416414  0.862919  2.520863 -0.005021  0.563341
27627  34634  0.333499  1.699873 -2.596561  3.643945 -0.585068 -0.654659
27738  34684 -2.439237  2.591458 -2.840126  1.286244 -1.777016 -1.436139
27749  34687 -0.860827  3.131790 -5.052968  5.420941 -2.494141 -1.811287
29687  35585 -2.019001  1.491270  0.005222  0.817253  0.973252 -0.639268

             V7        V8        V9  ...       V20       V21       V22  \
6797  -0.450036  0.307505  3.149560  ...  0.279180 -0.204773  0.527291
24446 -0.781279  0.153917 -0.517252  ... -0.391134 -0.139073 -0.107833
4403  -1.170528  0.710384  0.360121  ...  0.221827 -0.278178 -0.898199
22679  0.418325 -0.171220 -0.414952  ...  0.084010 -0.303750 -1.144951
24970  0.269744 -0.053734  0.262421  ... -0.107481  0.056877  0.049455
...         ...       ...       ...  ...       ...       ...       ...
27362 -0.123372  0.223122 -0.673598  ... -0.165249 -0.159387 -0.305154
```

0s completed at 4:53 PM

CodSoft_intern_Project1. Credit Card Fraud Detection.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

Files

.. sample_data creditcard.csv

Disk 83.30 GB available

+ Code + Text

RAM Disk

Split the data into Training data & Testing Data

```
[24] X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.2, stratify=Y, random_state=2)
```

```
[25] print(X.shape, X_train.shape, X_test.shape)
```

```
(586, 30) (468, 30) (118, 30)
```

Model Training

Logistic Regression

```
[26] model = LogisticRegression()
```

```
# training the Logistic Regression Model with Training Data
model.fit(X_train, Y_train)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/linear_model/_logistic.py:458: ConvergenceWarning: lbfgs failed to converge (st
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
```

0s completed at 4:53 PM

(33) Project 10. Credit Card Fraud | CodSoft_intern_Project1. Credit C | Credit Card Fraud Detection | Kag | Dhar9292/CODSOFT_PROJECT

colab.research.google.com/drive/1jCRZ7AC0vXqeQ67OEyKP78Bav9B2nlgh#scrollTo=hlNfrSC1PGfZ

CodSoft_intern_Project1. Credit Card Fraud Detection.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

Files

..
sample_data
creditcard.csv

Disk 83.30 GB available

+ Code + Text

RAM Disk

[28] 0s
```
# accuracy on training data
X_train_prediction = model.predict(X_train)
training_data_accuracy = accuracy_score(X_train_prediction, Y_train)
```

[29] 0s
```
print('Accuracy on Training data : ', training_data_accuracy)
```

```
Accuracy on Training data :  0.9829059829059829
```

[30] 0s
```
# accuracy on test data
X_test_prediction = model.predict(X_test)
test_data_accuracy = accuracy_score(X_test_prediction, Y_test)
```

[31] 0s
```
print('Accuracy score on Test Data : ', test_data_accuracy)
```

```
Accuracy score on Test Data :  0.9830508474576272
```

0s completed at 4:53 PM

Type here to search

35°C Haze

4:57 PM 8/10/2023