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

```
#load the dataset from the csv file using pandas
#best way is to mount the drive on colab and
#copy the path for the csvfile
data=pd.read_csv("/content/creditcard.csv")
```

```
data.head()
```

	Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	...	V21	V22	V23	
0	0	-1.359807	-0.072781	2.536347	1.378155	-0.338321	0.462388	0.239599	0.098698	0.363787	...	-0.018307	0.277838	-0.110474	0
1	0	1.191857	0.266151	0.166480	0.448154	0.060018	-0.082361	-0.078803	0.085102	-0.255425	...	-0.225775	-0.638672	0.101288	-0
2	1	-1.358354	-1.340163	1.773209	0.379780	-0.503198	1.800499	0.791461	0.247676	-1.514654	...	0.247998	0.771679	0.909412	-0
3	1	-0.966272	-0.185226	1.792993	-0.863291	-0.010309	1.247203	0.237609	0.377436	-1.387024	...	-0.108300	0.005274	-0.190321	-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	-0.137458	0

5 rows × 31 columns

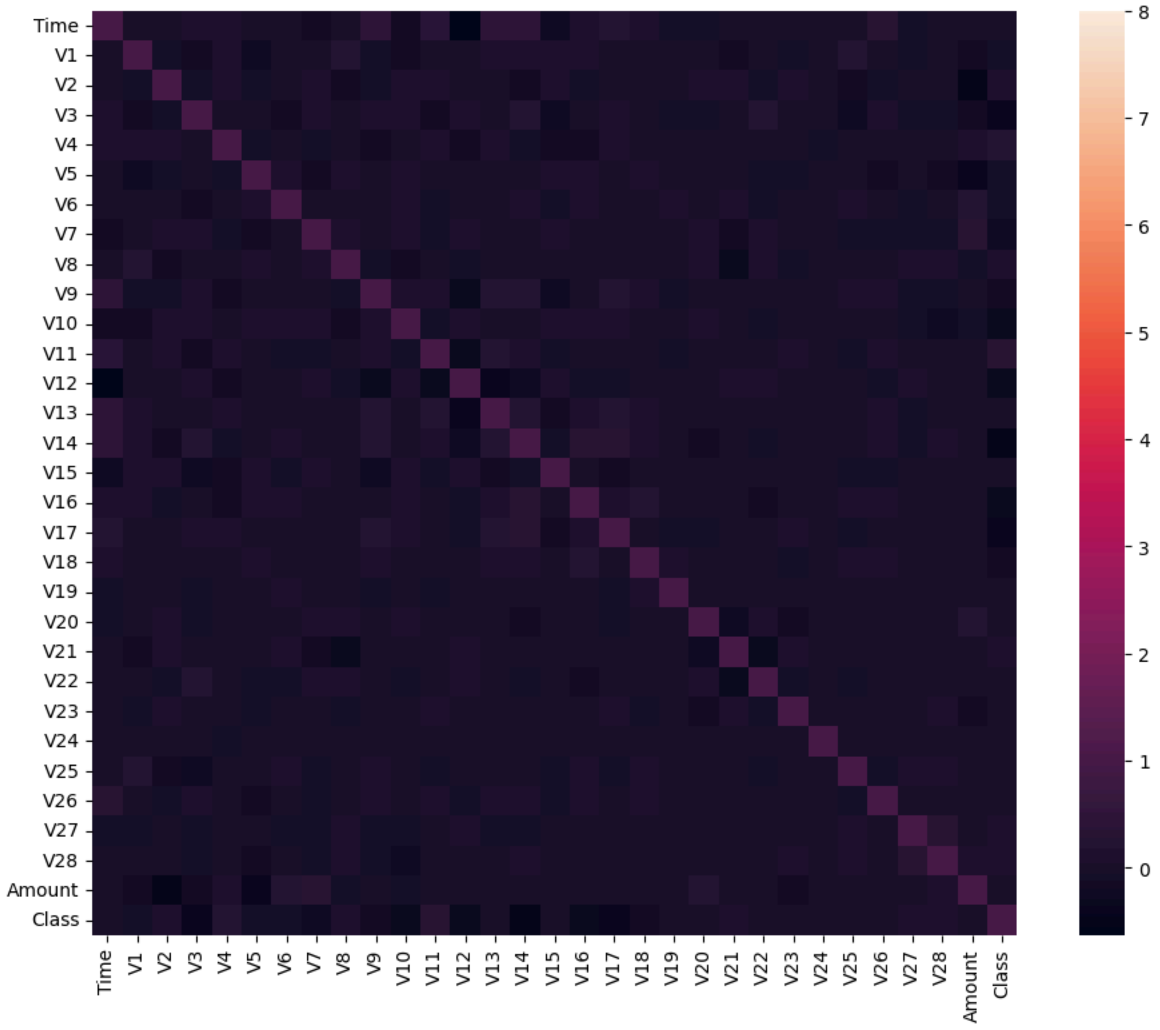
```
#determine number of fraud cases in dataset
fraud = data[data['Class'] == 1]
valid = data[data['Class'] == 0]
outlierfraction =len(fraud)/float(len(valid))
print(outlierfraction)
print('fraud cases:{}'.format(len(data[data['Class'] == 1])))
print('valid transaction: {})'.format(len(data[data['Class'] == 0])))
```

```
0.0038283296393310496
fraud cases:38
valid transaction: 9926)
```

```
print("details of valid transaction ")
valid.Amount.describe()
```

```
details of valid transaction
count    9926.000000
mean       62.919501
std       184.041297
min         0.000000
25%         5.172500
50%        15.950000
75%        51.067500
max      7712.430000
Name: Amount, dtype: float64
```

```
#Correlation materix
Corrmat = data.corr ()
fig = plt.figure (figsize = (12,9))
sns.heatmap(Corrmat, vmax = 8, square = True)
plt.show()
```


Time
V1
V2
V3
V4
V5
V6
V7
V8
V9
V10
V11
V12
V13
V14
V15
V16
V17
V18
V19
V20
V21
V22
V23
V24
V25
V26
V27
V28
Amount
Class
8
7
6
5
4
3
2
1
0