InitialProject

September 21, 2023

[74]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.cluster import KMeans
import warnings
warnings.filterwarnings("ignore", category=FutureWarning)
```

[3]:

```
data = pd.read_csv("Banknote-authentication-dataset- (1).csv")
data
```

[3]:

```
           V1        V2
0     3.62160   8.66610
1     4.54590   8.16740
2     3.86600  -2.63830
3     3.45660   9.52280
4     0.32924  -4.45520
…         …        …
1367  0.40614   1.34920
1368 -1.38870  -4.87730
1369 -3.75030 -13.45860
1370 -3.56370  -8.38270
1371 -2.54190  -0.65804

[1372 rows x 2 columns]
```

[7]:

```
data.describe()
```

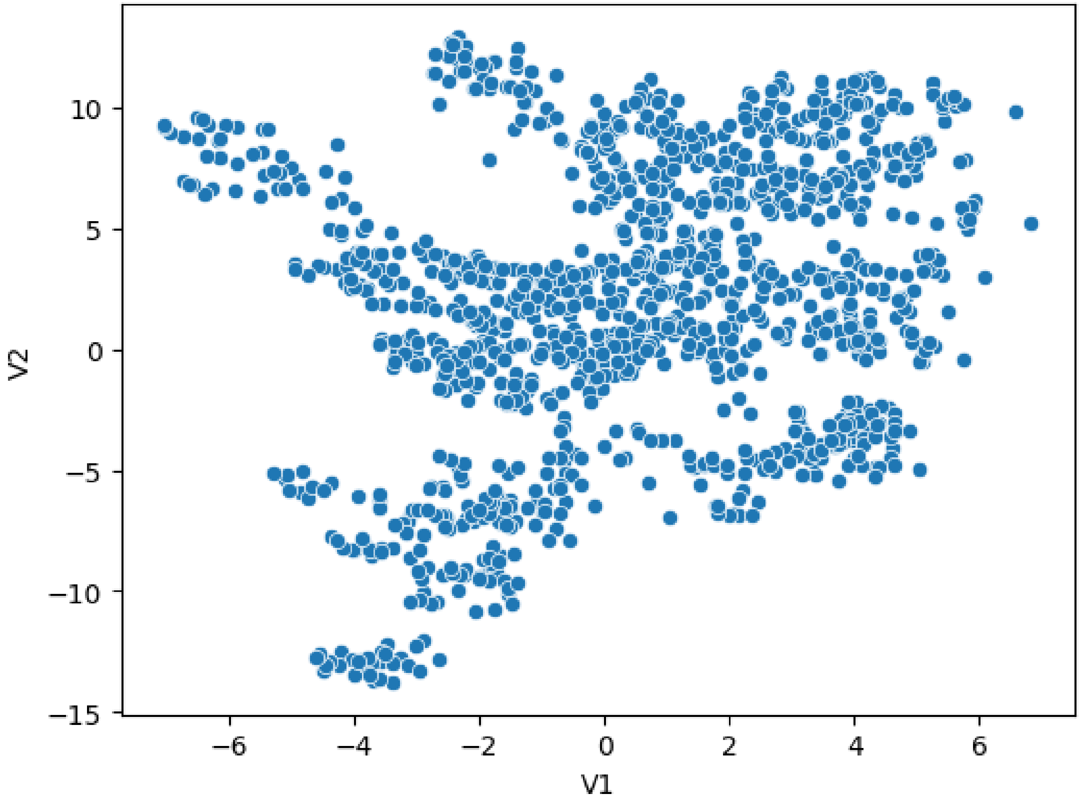
[7]:

```
                V1           V2
count  1372.000000  1372.000000
mean      0.433735     1.922353
std       2.842763     5.869047
min      -7.042100   -13.773100
25%      -1.773000    -1.708200
50%       0.496180     2.319650
75%       2.821475     6.814625
max       6.824800    12.951600
```

Both V1 and V2 present a mean much lower than the median suggesting they are both left skewed.

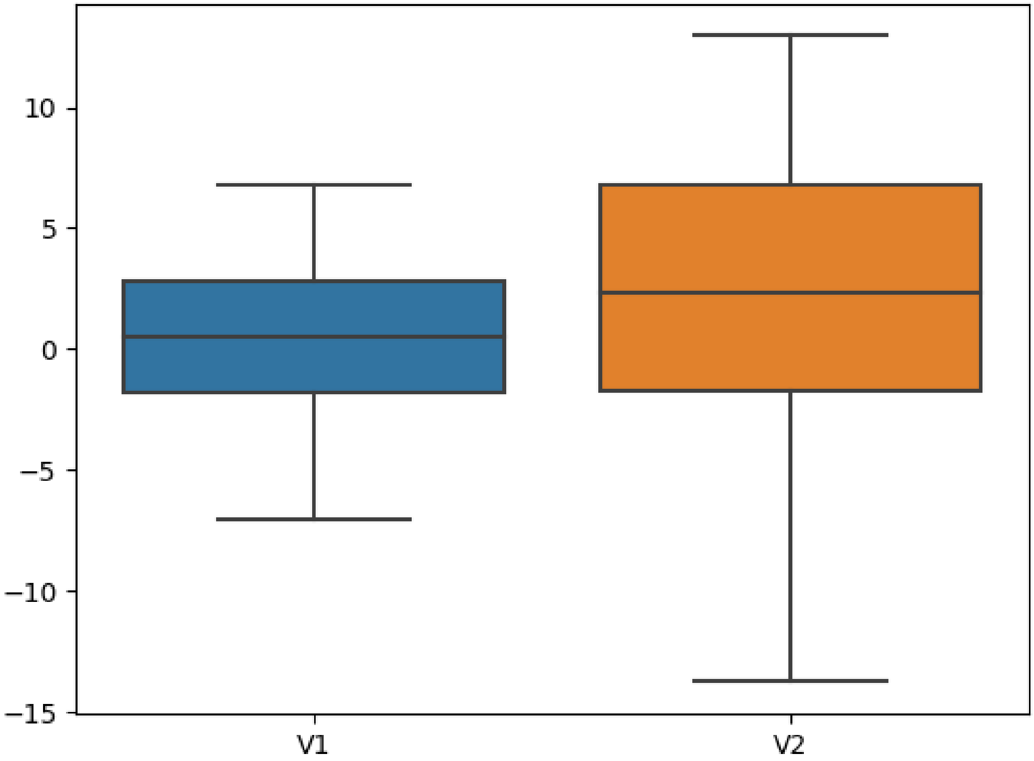
```
[18]: sns.scatterplot(data = data, x = 'V1', y= 'V2')
      plt.show
```

```
[18]: <function matplotlib.pyplot.show(close=None, block=None)>
```



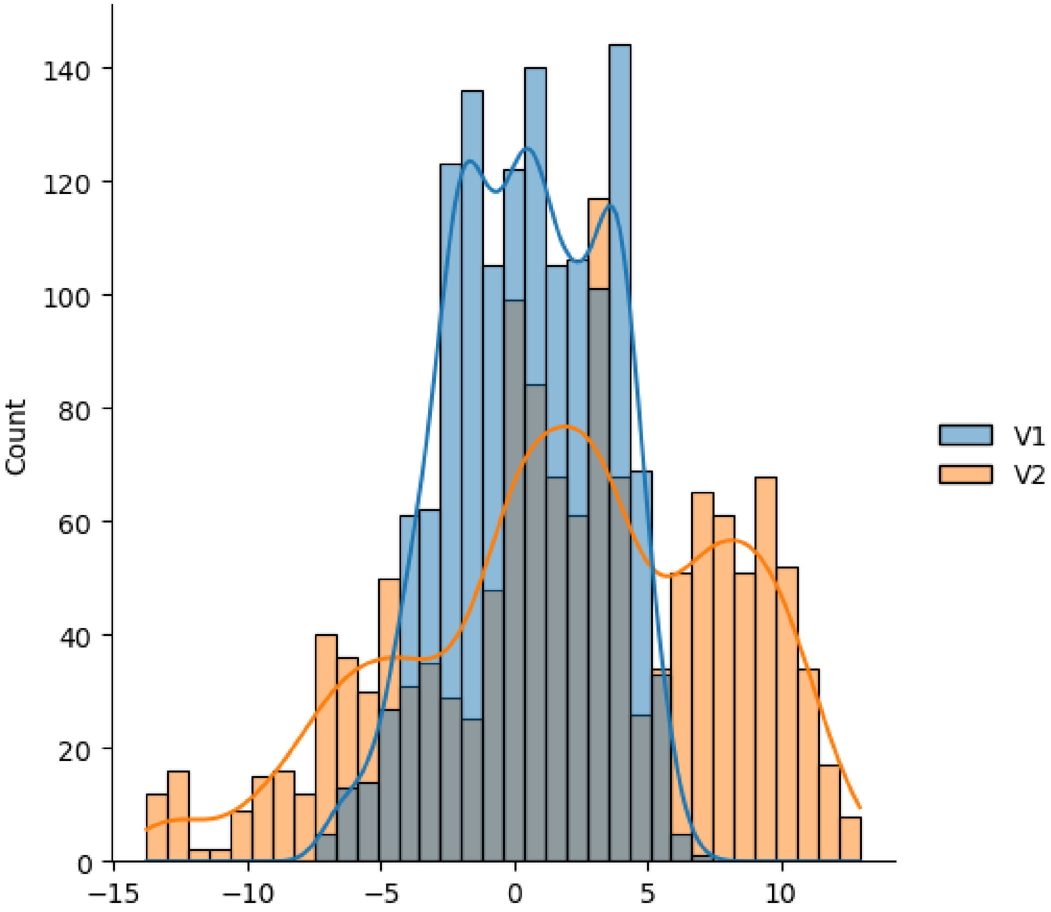
```
[16]: sns.boxplot(data = data[['V1','V2']])

      plt.show()
```



Boxplot does not show the presence of any outliers. V1 is highly centered while V2 has values further from the IQT.

[26]:
```
sns.displot(data[['V1', 'V2']], kde=True, label= ['V1', 'V2'])
plt.show()
```



The variability of V2 is much higher than V1, showing clearly how it is more left skewed. Than can be seen also as it has a higher standard deviation.

[29]:
```
# From here starts the attempt to analyse clusters and see if they are of value␣
  ↪to the data
columns = np.column_stack((data['V1'], data['V2']))
columns
```

[29]:
```
array([[  3.6216 ,   8.6661 ],
       [  4.5459 ,   8.1674 ],
       [  3.866  ,  -2.6383 ],
       …,
       [ -3.7503 , -13.4586 ],
       [ -3.5637 ,  -8.3827 ],
       [ -2.5419 ,  -0.65804]])
```

[162]:
```
# We select 2 as the number of clusters, because we are looking for a␣
  ↪distinction between the original and the forged banknotes
```

```
km = KMeans(n_clusters = 2).fit(columns)
km
```

[162]:

```
KMeans(n_clusters=2)
```

[165]:

```
clusters = km.cluster_centers_
```

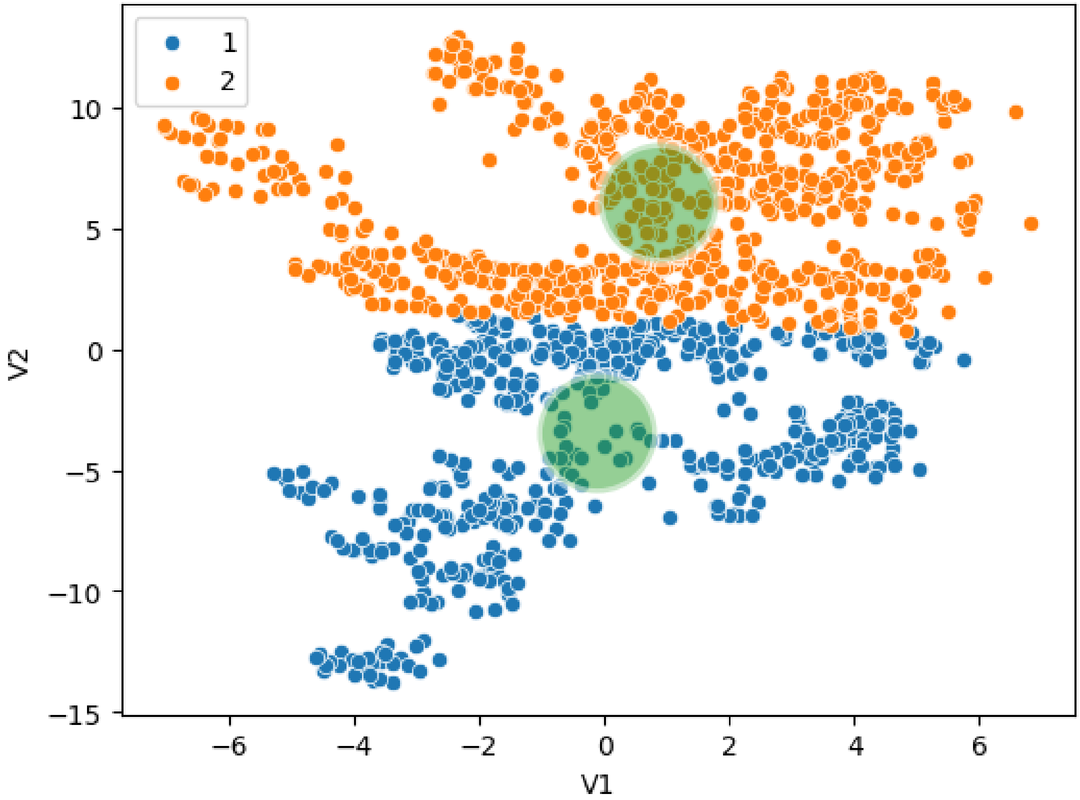
[199]:

```
# Here the data is divided according to the clusters. It creates two new dataframes, but it has been used solely
# for visualization purposes
new = KMeans(n_clusters = 2).fit_predict(columns)
binary1 = data[new ==0]
binary2 = data[new == 1]
```

```
[0 0 1 … 1 1 1]
```

[198]:

```
sns.scatterplot(binary1['V1'], binary1['V2'], label = '1')
sns.scatterplot(binary2['V1'], binary2['V2'], label = '2')
sns.scatterplot(clusters[:,0], clusters[:,1], s= 2000, alpha=0.5)
plt.show()
```



The two clusters present a clear distinction between the results that are supposably genuine (labelled

1) and supposably forged (labelled 2).