CCPS 844 Data Mining (Lab 6) Submit your solution as a pdf file

Q-1 Select a dataset of your choice. Apply PCA on it like we have applied in “Simple Example.ipynb”. If you want, you can add further details like “plot_pca_iris.ipynb”.

In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsClassifier
from sklearn import metrics

% matplotlib inline

from IPython.core.display import display, HTML
display(HTML("<style>.container { width:90% !important; }</style
>"))
pd.set_option('display.max_columns', 100)

df = pd.read_csv('winequality-red.csv')
```

In [3]:
```
df.sample()
```

Out[3]:

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH
1260	8.6	0.635	0.68	1.8	0.403	19.0	56.0	0.99632	3.02

In [31]:
```
X = df.iloc[:,0:3]
```

In [32]:
```
X.sample()
```

Out[32]:

	fixed acidity	volatile acidity	citric acid
867	6.9	0.51	0.23

In [34]:
```
from sklearn import decomposition
pca = decomposition.PCA(n_components=2)
pca.fit(X)
```

Out[34]:
```
PCA(copy=True, iterated_power='auto', n_components=2, random_s
tate=None,
  svd_solver='auto', tol=0.0, whiten=False)
```

In [35]:
```
X1 = pca.transform(X)
```

In [36]:
```
X1
```

Out[36]:
```
array([[-0.9417702 ,  0.23605474],
       [-0.54789059,  0.408953  ],
       [-0.54164699,  0.28813908],
       ...,
       [-2.02330735, -0.06580543],
       [-2.42639976,  0.0237052 ],
       [-2.29128709, -0.44430566]])
```