Name:- Raj Khatri
Roll Number:- AC-1235
Subject:- Data Mining
Semester:- 6
Practical - 3

In [27]:

```
from sklearn.preprocessing import StandardScaler
from sklearn.datasets import load_wine
from sklearn.utils.validation import check_is_fitted
```

In [3]:

```
import pandas as pd
import numpy as np
import seaborn as sns
```

In [4]:

```
wineQuality = pd.read_csv("winequalityN.csv")
```

In [5]:

```
wineQuality.shape
```

Out[5]:

```
(6497, 13)
```

In [6]:

```
wineQuality
```

Out[6]:

	type	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
0	white	7.0	0.270	0.36	20.7	0.045	45.0	170.0	1.00100	3.00	0.45	8.8	6
1	white	6.3	0.300	0.34	1.6	0.049	14.0	132.0	0.99400	3.30	0.49	9.5	6
2	white	8.1	0.280	0.40	6.9	0.050	30.0	97.0	0.99510	3.26	0.44	10.1	6
3	white	7.2	0.230	0.32	8.5	0.058	47.0	186.0	0.99560	3.19	0.40	9.9	6
4	white	7.2	0.230	0.32	8.5	0.058	47.0	186.0	0.99560	3.19	0.40	9.9	6
...	...	...	...	...	...	...	...	...	...	...	...	...	...
6492	red	6.2	0.600	0.08	2.0	0.090	32.0	44.0	0.99490	3.45	0.58	10.5	5
6493	red	5.9	0.550	0.10	2.2	0.062	39.0	51.0	0.99512	3.52	NaN	11.2	6
6494	red	6.3	0.510	0.13	2.3	0.076	29.0	40.0	0.99574	3.42	0.75	11.0	6
6495	red	5.9	0.645	0.12	2.0	0.075	32.0	44.0	0.99547	3.57	0.71	10.2	5
6496	red	6.0	0.310	0.47	3.6	0.067	18.0	42.0	0.99549	3.39	0.66	11.0	6

6497 rows × 13 columns

In [7]:

```
wineQuality['type'].dtypes
```

Out[7]:

dtype('O')

In [8]:

```
for i in wineQuality.columns:
    if wineQuality[i].dtype != "object":
        if wineQuality[i].mean()!= 0 and wineQuality[i].std()!=1:
            print(i)
```

```
fixed acidity
volatile acidity
citric acid
residual sugar
chlorides
free sulfur dioxide
total sulfur dioxide
density
pH
sulphates
alcohol
quality
```

In [9]:

```
wineQuality
```

Out[9]:

	type	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
0	white	7.0	0.270	0.36	20.7	0.045	45.0	170.0	1.00100	3.00	0.45	8.8	6
1	white	6.3	0.300	0.34	1.6	0.049	14.0	132.0	0.99400	3.30	0.49	9.5	6
2	white	8.1	0.280	0.40	6.9	0.050	30.0	97.0	0.99510	3.26	0.44	10.1	6
3	white	7.2	0.230	0.32	8.5	0.058	47.0	186.0	0.99560	3.19	0.40	9.9	6
4	white	7.2	0.230	0.32	8.5	0.058	47.0	186.0	0.99560	3.19	0.40	9.9	6
...	...	...	...	...	...	...	...	...	...	...	...	...	...
6492	red	6.2	0.600	0.08	2.0	0.090	32.0	44.0	0.99490	3.45	0.58	10.5	5
6493	red	5.9	0.550	0.10	2.2	0.062	39.0	51.0	0.99512	3.52	NaN	11.2	6
6494	red	6.3	0.510	0.13	2.3	0.076	29.0	40.0	0.99574	3.42	0.75	11.0	6
6495	red	5.9	0.645	0.12	2.0	0.075	32.0	44.0	0.99547	3.57	0.71	10.2	5
6496	red	6.0	0.310	0.47	3.6	0.067	18.0	42.0	0.99549	3.39	0.66	11.0	6

6497 rows × 13 columns

In [10]:

```
wineQuality.drop(columns="type",inplace=True)
```

In [11]:

```
df = wineQuality.describe().iloc[1:3]
df
```

Out[11]:

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	(
mean	7.216579	0.339691	0.318722	5.444326	0.056042	30.525319	115.744574	0.994697	3.218395	0.531215	10.491801	5.8
std	1.296750	0.164649	0.145265	4.758125	0.035036	17.749400	56.521855	0.002999	0.160748	0.148814	1.192712	0.8

In [12]:

```
scalar = StandardScaler()
standarized_array = scalar.fit_transform(wineQuality)
```

In [13]:

```
standarized_df = pd.DataFrame(standarized_array,columns=wineQuality.columns)
```

In [21]:

```
wineQuality.describe()
```

Out[21]:

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH
count	6487.000000	6489.000000	6494.000000	6495.000000	6495.000000	6497.000000	6497.000000	6497.000000	6488.000000
mean	7.216579	0.339691	0.318722	5.444326	0.056042	30.525319	115.744574	0.994697	3.218395
std	1.296750	0.164649	0.145265	4.758125	0.035036	17.749400	56.521855	0.002999	0.160748
min	3.800000	0.080000	0.000000	0.600000	0.009000	1.000000	6.000000	0.987110	2.720000
25%	6.400000	0.230000	0.250000	1.800000	0.038000	17.000000	77.000000	0.992340	3.110000
50%	7.000000	0.290000	0.310000	3.000000	0.047000	29.000000	118.000000	0.994890	3.210000
75%	7.700000	0.400000	0.390000	8.100000	0.065000	41.000000	156.000000	0.996990	3.320000
max	15.900000	1.580000	1.660000	65.800000	0.611000	289.000000	440.000000	1.038980	4.010000

In [20]:

```
standarized_df.describe()
```

Out[20]:

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	
count	6.487000e+03	6489.000000	6.494000e+03	6.495000e+03	6.495000e+03	6.497000e+03	6497.000000	6.497000e+03	6.48
mean	4.030827e-16	0.000000	4.726739e-16	-1.400300e-16	-7.001499e-17	-8.749179e-17	0.000000	-3.560916e-15	1.75
std	1.000077e+00	1.000077	1.000077e+00	1.000077e+00	1.000077e+00	1.000077e+00	1.000077	1.000077e+00	1.00
min	-2.634928e+00	-1.577361	-2.194244e+00	-1.018195e+00	-1.342766e+00	-1.663583e+00	-1.941780	-2.530192e+00	3.10
25%	-6.297608e-01	-0.666262	-4.731166e-01	-7.659755e-01	-5.149831e-01	-7.620742e-01	-0.685532	-7.859527e-01	-6.
50%	-1.670299e-01	-0.301823	-6.004599e-02	-5.137559e-01	-2.580848e-01	-8.594301e-02	0.039907	6.448888e-02	-5.
75%	3.728228e-01	0.366316	4.907148e-01	5.581775e-01	2.557116e-01	5.901882e-01	0.712265	7.648525e-01	6.32
max	6.696812e+00	7.533628	9.234043e+00	1.268574e+01	1.584087e+01	1.456357e+01	5.737257	1.476879e+01	4.92

IRIS DATASET

In [15]:

```
iris_df = sns.load_dataset("iris")
```

In [16]:

```
new_iris = iris_df.iloc[:,:-1]
```

In [17]:

```
standarized_array1 = scalar.fit_transform(new_iris)
```

In [18]:

```
standarized_df1 = pd.DataFrame(standarized_array1,columns=new_iris.columns)
```

In [19]:

```
standarized_df1.describe()
```

Out[19]:

	sepal_length	sepal_width	petal_length	petal_width
count	1.500000e+02	1.500000e+02	1.500000e+02	1.500000e+02
mean	-4.736952e-16	-7.815970e-16	-4.263256e-16	-4.736952e-16
std	1.003350e+00	1.003350e+00	1.003350e+00	1.003350e+00
min	-1.870024e+00	-2.433947e+00	-1.567576e+00	-1.447076e+00
25%	-9.006812e-01	-5.923730e-01	-1.226552e+00	-1.183812e+00
50%	-5.250608e-02	-1.319795e-01	3.364776e-01	1.325097e-01
75%	6.745011e-01	5.586108e-01	7.627583e-01	7.906707e-01
max	2.492019e+00	3.090775e+00	1.785832e+00	1.712096e+00

In [ ]:

```

```