

DBSCAN-Clustering

In [1]:

```
# import libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.cluster import DBSCAN
from sklearn.preprocessing import StandardScaler
```

In [2]:

```
# Load dataset
df=pd.read_csv("C:/Users/Ashraf/Documents/Datafiles/Wholesale customers data.csv");
df
```

Out[2]:

	Channel	Region	Fresh	Milk	Grocery	Frozen	Detergents_Paper	Delicassen
0	2	3	12669	9656	7561	214	2674	1338
1	2	3	7057	9810	9568	1762	3293	1776
2	2	3	6353	8808	7684	2405	3516	7844
3	1	3	13265	1196	4221	6404	507	1788
4	2	3	22615	5410	7198	3915	1777	5185
...
435	1	3	29703	12051	16027	13135	182	2204
436	1	3	39228	1431	764	4510	93	2346
437	2	3	14531	15488	30243	437	14841	1867
438	1	3	10290	1981	2232	1038	168	2125
439	1	3	2787	1698	2510	65	477	52

440 rows × 8 columns

In [3]:

```
df.head()
```

Out[3]:

	Channel	Region	Fresh	Milk	Grocery	Frozen	Detergents_Paper	Delicassen
0	2	3	12669	9656	7561	214	2674	1338
1	2	3	7057	9810	9568	1762	3293	1776
2	2	3	6353	8808	7684	2405	3516	7844
3	1	3	13265	1196	4221	6404	507	1788
4	2	3	22615	5410	7198	3915	1777	5185

In [4]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 440 entries, 0 to 439
Data columns (total 8 columns):
 #   Column                Non-Null Count  Dtype
---  -
 0   Channel                440 non-null    int64
 1   Region                 440 non-null    int64
 2   Fresh                  440 non-null    int64
 3   Milk                   440 non-null    int64
 4   Grocery                440 non-null    int64
 5   Frozen                 440 non-null    int64
 6   Detergents_Paper       440 non-null    int64
 7   Delicassen             440 non-null    int64
dtypes: int64(8)
memory usage: 27.6 KB
```

In [5]:

```
df.drop(['Channel', 'Region'], axis=1, inplace=True)
```

In [6]:

```
array=df.values
```

In [7]:

```
array
```

Out[7]:

```
array([[12669, 9656, 7561, 214, 2674, 1338],
       [ 7057, 9810, 9568, 1762, 3293, 1776],
       [ 6353, 8808, 7684, 2405, 3516, 7844],
       ...,
       [14531, 15488, 30243, 437, 14841, 1867],
       [10290, 1981, 2232, 1038, 168, 2125],
       [ 2787, 1698, 2510, 65, 477, 52]], dtype=int64)
```

In [8]:

```
stscaler=StandardScaler().fit(array)
X=stscaler.transform(array)
```

In [9]:

```
X
```

Out[9]:

```
array([[ 0.05293319,  0.52356777, -0.04111489, -0.58936716, -0.04356873,
        -0.06633906],
       [-0.39130197,  0.54445767,  0.17031835, -0.27013618,  0.08640684,
        0.08915105],
       [-0.44702926,  0.40853771, -0.0281571 , -0.13753572,  0.13323164,
        2.24329255],
       ...,
       [ 0.20032554,  1.31467078,  2.34838631, -0.54337975,  2.51121768,
        0.12145607],
       [-0.13538389, -0.51753572, -0.60251388, -0.41944059, -0.56977032,
        0.21304614],
       [-0.72930698, -0.5559243 , -0.57322717, -0.62009417, -0.50488752,
        -0.52286938]])
```

In [10]:

```
dbscan=DBSCAN(eps=0.8, min_samples=6)
dbscan.fit(X)
```

Out[10]:

```
DBSCAN(eps=0.8, min_samples=6)
```

In [11]:

```
#Noisy samples are given the label -1.
dbscan.labels_
```

Out[11]:

```
array([ 0,  0, -1,  0, -1,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
       -1,  0,  0,  0,  0, -1, -1, -1,  0,  0,  0, -1,  0,  0,  0,  0, -1,
        0,  0,  0,  0,  0, -1, -1,  0,  0,  0,  0, -1,  0, -1,  0, -1,  0,
        0,  0,  0,  0,  0, -1,  0,  0,  0,  0, -1,  0,  0,  0, -1,  0,  0,
        0,  0,  0, -1,  0,  0,  0,  0,  0, -1,  0,  0,  0,  0,  0,  0,  0,
       -1, -1, -1,  0,  0,  0,  0, -1, -1,  0,  0,  0,  0,  0,  0, -1,  0,
        0, -1,  0,  0,  0,  0,  0, -1,  0,  0,  0,  0,  0,  0,  0,  0,  0,
        0,  0,  0,  0,  0,  0, -1,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
        0,  0,  0,  0,  0, -1, -1,  0,  0, -1,  0,  0,  0,  0,  0,  0,  0,
        0,  0,  0,  0,  0,  0,  0,  0,  0,  0, -1,  0, -1,  0,  0,  0,  0,
        0, -1,  0,  0,  0,  0, -1,  0,  0,  0,  0,  0, -1,  0, -1,  0,  0,
        0,  0,  0,  0,  0,  0,  0,  0,  0, -1,  0,  0,  0,  0, -1, -1,  0,
        0,  0,  0,  0,  0, -1,  0, -1,  0,  0,  0,  0, -1,  0, -1,  0,  0,
        0,  0,  0,  0,  0,  0,  0,  0, -1,  0,  0,  0,  0,  0,  0,  0,  0,
        0, -1,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0, -1,  0, -1,
        0, -1,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
       -1,  0, -1,  0,  0, -1, -1,  0,  0,  0, -1,  0, -1,  0,  0],
      dtype=int64)
```

In [12]:

```
cl=pd.DataFrame(dbscan.labels_, columns=['cluster'])
```

In [13]:

```
c1
```

Out[13]:

cluster	
0	0
1	0
2	-1
3	0
4	-1
...	...
435	-1
436	0
437	-1
438	0
439	0

440 rows × 1 columns

In [14]:

```
pd.concat([df,c1], axis=1)
```

Out[14]:

	Fresh	Milk	Grocery	Frozen	Detergents_Paper	Delicassen	cluster
0	12669	9656	7561	214	2674	1338	0
1	7057	9810	9568	1762	3293	1776	0
2	6353	8808	7684	2405	3516	7844	-1
3	13265	1196	4221	6404	507	1788	0
4	22615	5410	7198	3915	1777	5185	-1
...
435	29703	12051	16027	13135	182	2204	-1
436	39228	1431	764	4510	93	2346	0
437	14531	15488	30243	437	14841	1867	-1
438	10290	1981	2232	1038	168	2125	0
439	2787	1698	2510	65	477	52	0

440 rows × 7 columns

In []: