In [1]:

```
import pandas as pd
from sklearn.cluster import DBSCAN
import matplotlib.pyplot as plt
%matplotlib inline
```

In [2]:

```
df = pd.read_csv("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]:

```
#scaling
from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
```

In [4]:

```
x_scaled = scaler.fit_transform(df)
```

In [5]:

```
x_scaled
```

Out[5]:

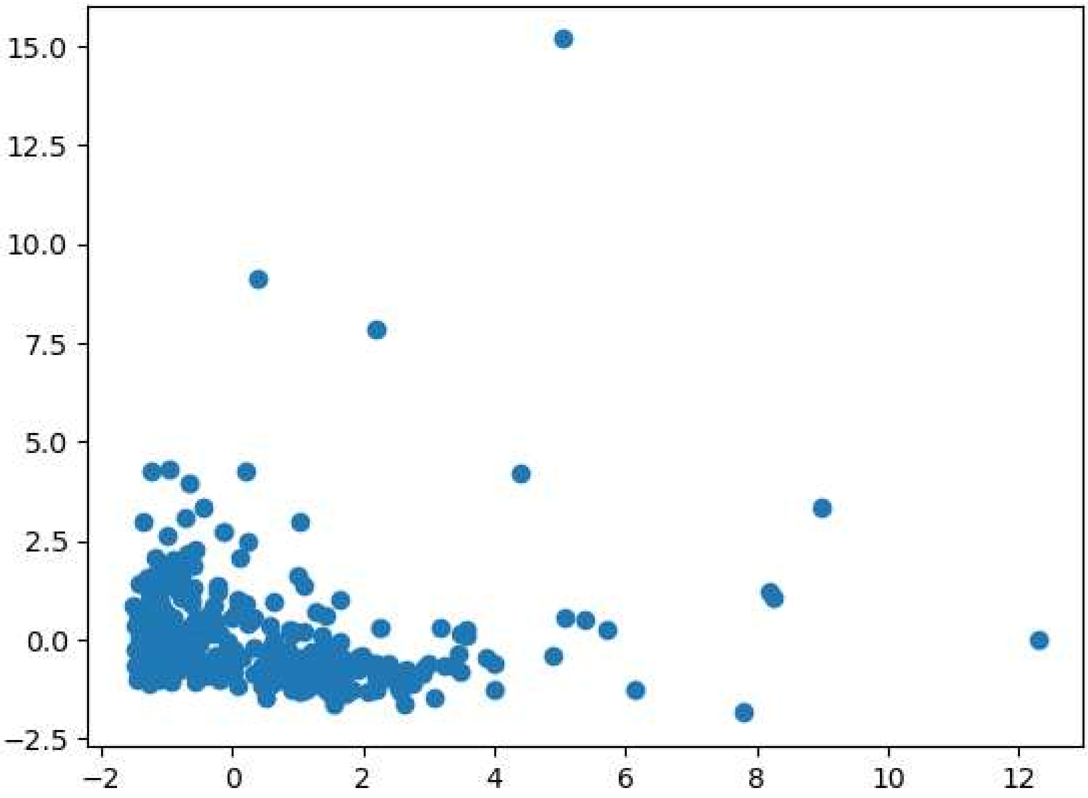
```
array([[ 1.44865163,  0.59066829,  0.05293319, ..., -0.58936716,
        -0.04356873, -0.06633906],
       [ 1.44865163,  0.59066829, -0.39130197, ..., -0.27013618,
         0.08640684,  0.08915105],
       [ 1.44865163,  0.59066829, -0.44702926, ..., -0.13753572,
         0.13323164,  2.24329255],
       ...,
       [ 1.44865163,  0.59066829,  0.20032554, ..., -0.54337975,
         2.51121768,  0.12145607],
       [-0.69029709,  0.59066829, -0.13538389, ..., -0.41944059,
        -0.56977032,  0.21304614],
       [-0.69029709,  0.59066829, -0.72930698, ..., -0.62009417,
        -0.50488752, -0.52286938]])
```

In [6]:

```
#pca for reduce number of dimensional

from sklearn.decomposition import PCA
pca=PCA(n_components=3)
pca_scaled=pca.fit_transform(x_scaled)
plt.scatter(pca_scaled[:,0],pca_scaled[:,1])
```

Out[6]:

```
<matplotlib.collections.PathCollection at 0x186c6e70a10>
```



In [7]:

```
#apply DBscan Algorithm
dbcan=DBSCAN(eps=0.5)
dbcan.fit(pca_scaled)
```

Out[7]:

```
▾ DBSCAN
DBSCAN()
```

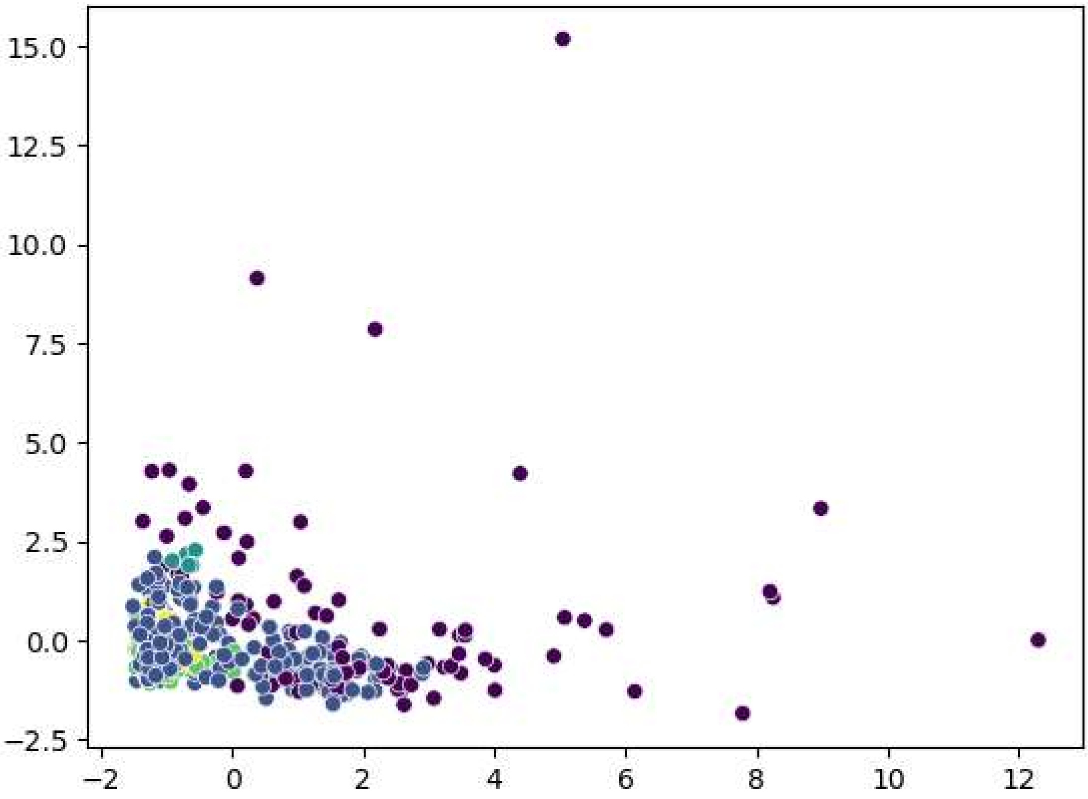
```
In [8]:  dbcan.labels_

Out[8]: array([ 0,  0, -1,  0, -1,  0,  0,  0,  0,  0,  0,  0, -1,  0,  0,  0,
0,
                0,  0,  0,  0,  0,  1, -1, -1,  0,  0,  0, -1,  0,  0,  0,  0,
0,
                0,  0,  0,  0,  0, -1,  1,  0,  0,  0,  0, -1,  0, -1,  0, -1,
0,
                0, -1,  0,  0,  0, -1,  0,  0,  0,  0, -1,  0,  0,  0, -1,  0,
0,
                0,  0,  0, -1,  0, -1,  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,  0,
0,
                0, -1,  0,  0,  0,  0,  0,  0,  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,  0,  0,  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, -1,  0, -1,  0,  0,
0,
                0,  0,  0,  0,  0,  0,  0,  0,  0, -1, -1,  2,  2, -1, -1, -1,
2,
                2, -1,  2, -1,  2, -1,  2, -1,  2,  2, -1, -1, -1,  2, -1,  2,
2,
                2,  2,  2,  2,  2, -1,  2,  2,  2, -1,  2,  2,  2,  2,  2,  2,
2,
                2, -1, -1,  2,  2,  2,  2, -1,  2,  2,  2,  2,  2, -1,  2, -1, -
1,
                2,  2,  2, -1, -1,  2,  2,  2,  2, -1, -1, -1,  2, -1,  2,  2,
2,
                2,  0,  0,  0,  0,  0,  0,  0,  0,  0,  1,  0, -1,  0,  0,  0,
0,
                0,  0,  0,  0, -1,  3, -1,  3, -1, -1,  3, -1, -1, -1, -1, -1, -
1,
               -1,  3,  3, -1,  3,  3, -1,  3,  3, -1,  3,  3,  3, -1,  3,  3,
3,
                3,  3, -1,  3,  3,  3,  3,  3, -1,  3, -1, -1, -1,  3,  3,  3,
3,
                0,  0,  0, -1,  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,
                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,  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, -1,  0,  0,  0,  0,  0,  0,  0, -1,  0, -1,  0,  0],
              dtype=int64)
```

In [9]:

```
import seaborn as sns
sns.scatterplot(x = pca_scaled[:,0],y= pca_scaled[:,1],c=dbcan.labels_)
```

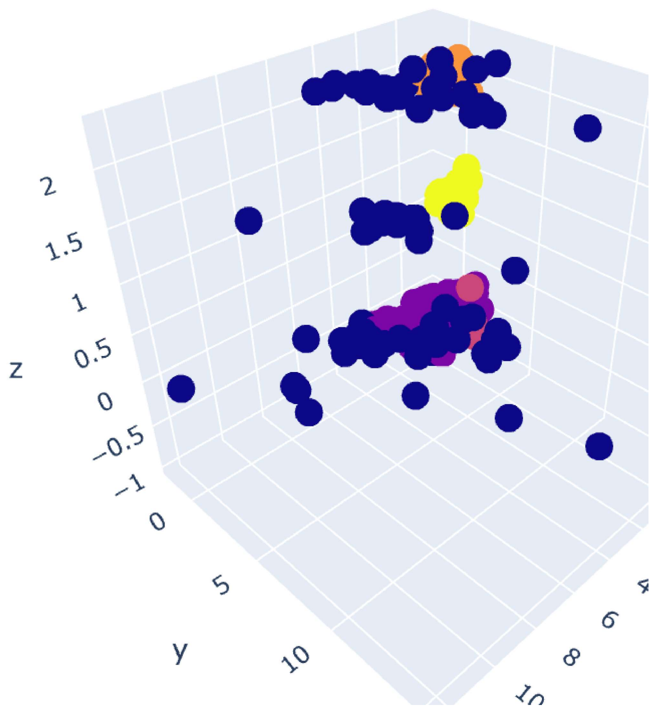
Out[9]:

```
<Axes: >
```



In [10]:

```
#3D plot
import plotly.express as px

fig = px.scatter_3d(pca_scaled, x=pca_scaled[:,0], y=pca_scaled[:,1], z=pca
              color=dbcan.labels_)
fig.show()
```



In [ ]:

In [ ]: