In [1]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

from sklearn.cluster import DBSCAN
from sklearn.preprocessing import StandardScaler
from sklearn.preprocessing import normalize
from sklearn.decomposition import PCA
```

In [2]:

```
X=pd.read_csv('covid.csv')
```

In [7]:

```
print(X.head())
```

```
        WHO Region  Cases - cumulative total  \
0           Global                 178503429
1         Americas                  33190195
2  South-East Asia                  29977861
3         Americas                  17927928
4           Europe                   5650315

   Cases - cumulative total per 100000 population  \
0                                       2290.108471
1                                      10027.170000
2                                       2172.300000
3                                       8434.310000
4                                       8687.530000

   Cases - newly reported in last 7 days  \
0                                2504100
1                                  49697
2                                 406980
3                                 515162
4                                  14066

   Cases - newly reported in last 7 days per 100000 population  \
0                                          32.126333
1                                          15.010000
2                                          29.490000
3                                         242.360000
4                                          21.630000

   Cases - newly reported in last 24 hours  Deaths - cumulative total  \
0                                   281610                    3872457
1                                        0                     596003
2                                    42640                     389302
3                                    44178                     501825
4                                      290                     109879

   Deaths - cumulative total per 100000 population  \
0                                        49.681659
1                                       180.060000
2                                        28.210000
3                                       236.090000
4                                       168.940000

   Deaths - newly reported in last 7 days  \
0                                   60068
1                                    1359
2                                   12271
3                                   14424
4                                     307

   Deaths - newly reported in last 7 days per 100000 population  \
0                                           0.770642
1                                           0.410000
2                                           0.890000
3                                           6.790000
4                                           0.470000
```

```
   Deaths - newly reported in last 24 hours Transmission Classification
0                                      6349                          NaN
1                                         0       Community transmission
2                                      1167            Clusters of cases
3                                      1025       Community transmission
4                                        39       Community transmission
```

In [9]:

```
X=X.drop('WHO Region',axis=1)
X=X.drop('Transmission Classification',axis=1)
```

In [19]:

```
X.fillna(method='ffill',inplace=True)
```

In [21]:

```
scaler=StandardScaler()
X_scaled=scaler.fit_transform(X)
X_normalized=normalize(X_scaled)
X_normalized=pd.DataFrame(X_normalized)
```

In [31]:

```
pca=PCA(n_components =2)
X_principal=pca.fit_transform(X_normalized)
X_principal=pd.DataFrame(X_principal)
X_principal.columns=['P1','P2']
print(X_principal.head())
```

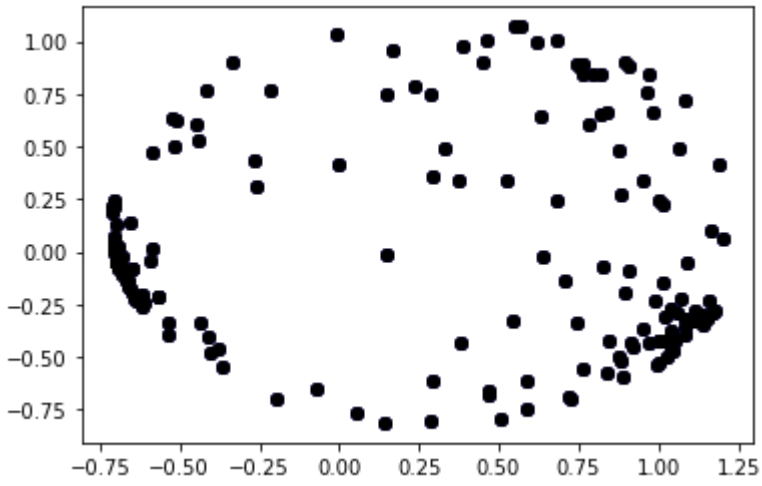
```
         P1        P2
0  0.373862  0.342797
1  0.826963 -0.072608
2  0.292619  0.360169
3  0.819114  0.655301
4  1.136925 -0.311494
```

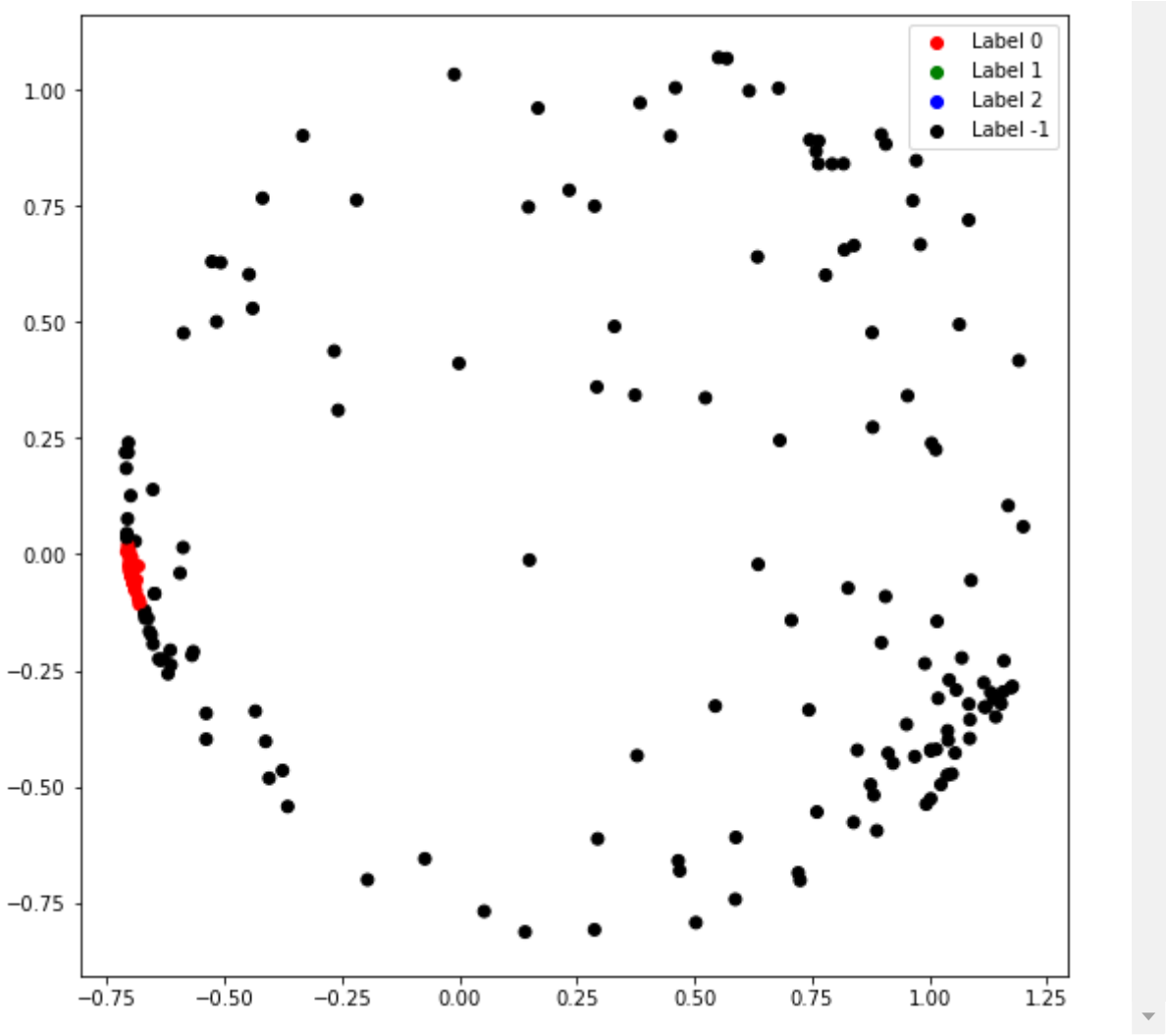
In [38]:

```
db_default = DBSCAN(eps = 0.0375, min_samples = 50).fit(X_principal)
labels = db_default.labels_
```

In [39]:

```
colours = {}
colours[0] = 'r'
colours[1] = 'g'
colours[2] = 'b'
colours[-1] = 'k'

cvec = [colours[label] for label in labels]

r = plt.scatter(X_principal['P1'], X_principal['P2'], color ='r');
g = plt.scatter(X_principal['P1'], X_principal['P2'], color ='g');
b = plt.scatter(X_principal['P1'], X_principal['P2'], color ='b');
k = plt.scatter(X_principal['P1'], X_principal['P2'], color ='k');



plt.figure(figsize =(9, 9))
plt.scatter(X_principal['P1'], X_principal['P2'], c = cvec)


plt.legend((r, g, b, k), ('Label 0', 'Label 1', 'Label 2', 'Label -1'))

plt.show()
```



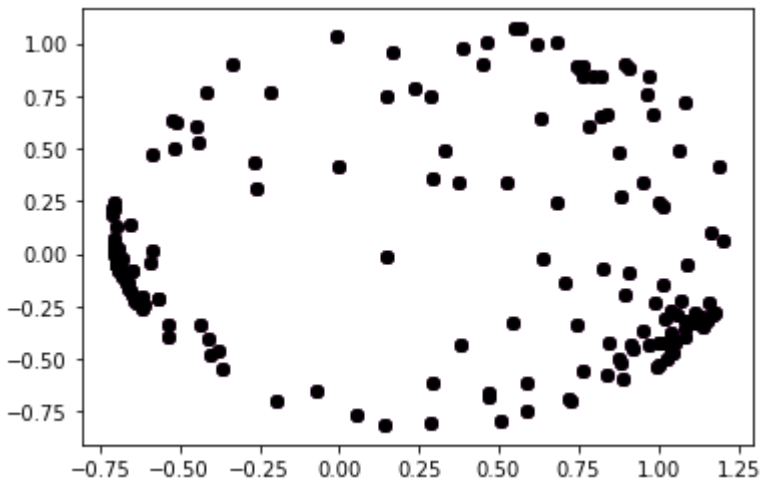


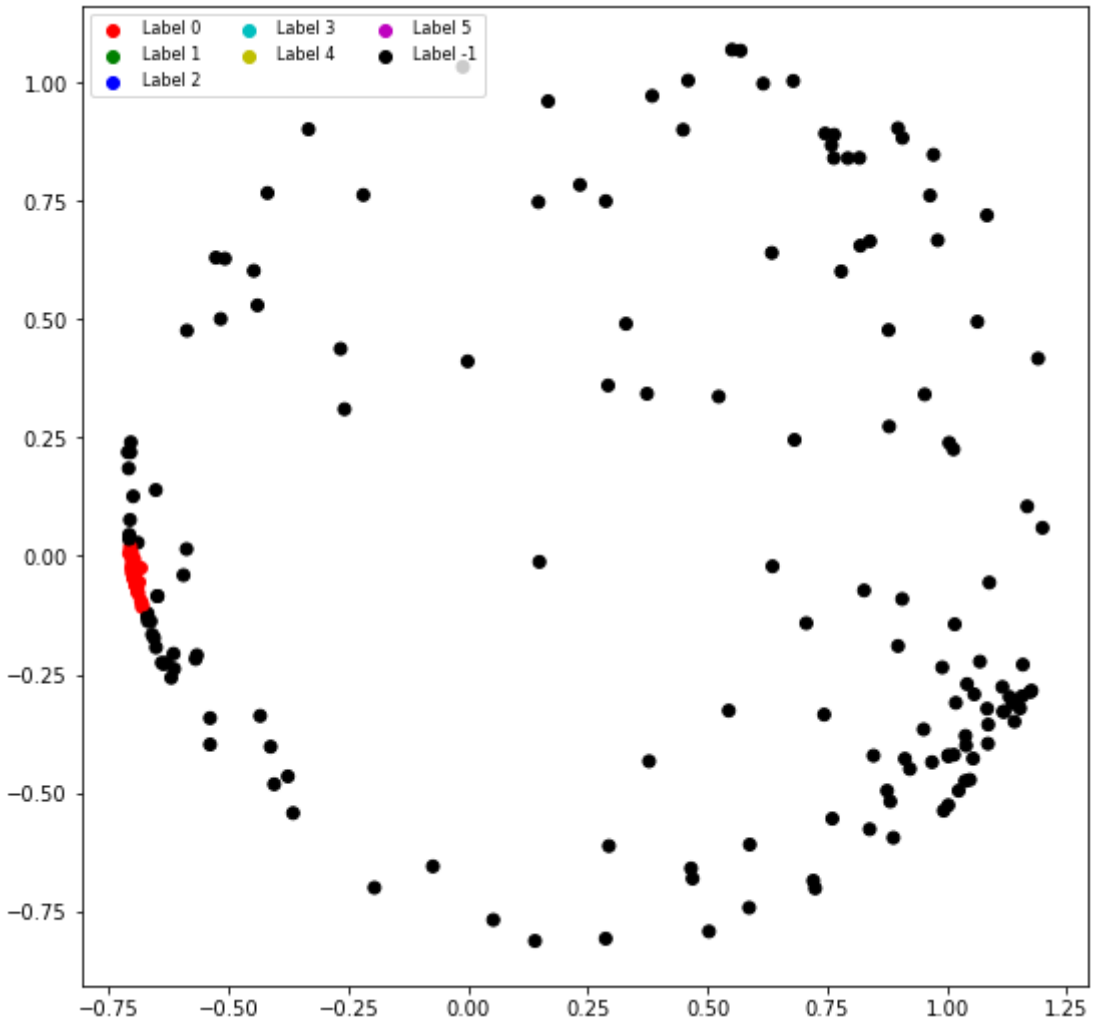
In [40]:

```
db = DBSCAN(eps = 0.0375, min_samples = 50).fit(X_principal)
labels1 = db.labels_
```

In [44]:

```
colours1 = {}
colours1[0] = 'r'
colours1[1] = 'g'
colours1[2] = 'b'
colours1[3] = 'c'
colours1[4] = 'y'
colours1[5] = 'm'
colours1[-1] = 'k'

cvec = [colours1[label] for label in labels]
colors = ['r', 'g', 'b', 'c', 'y', 'm', 'k' ]

r = plt.scatter(X_principal['P1'], X_principal['P2'], marker ='o', color = colors[0])
g = plt.scatter(X_principal['P1'], X_principal['P2'], marker ='o', color = colors[1])
b = plt.scatter(X_principal['P1'], X_principal['P2'], marker ='o', color = colors[2])
c = plt.scatter(X_principal['P1'], X_principal['P2'], marker ='o', color = colors[3])
y = plt.scatter(X_principal['P1'], X_principal['P2'], marker ='o', color = colors[4])
m = plt.scatter(X_principal['P1'], X_principal['P2'], marker ='o', color = colors[5])
k = plt.scatter(X_principal['P1'], X_principal['P2'], marker ='o', color = colors[6])

plt.figure(figsize =(9, 9))
plt.scatter(X_principal['P1'], X_principal['P2'], c = cvec)
plt.legend((r, g, b, c, y, m, k),
           ('Label 0', 'Label 1', 'Label 2', 'Label 3','Label 4','Label 5','Label -1'),scat
plt.show()
```




Label 0
Label 1
Label 2
Label 3
Label 4
Label 5
Label -1
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00
1.25

In [ ]:

In [ ]: