

In [1]:

```
1 pip install pygad
```

Requirement already satisfied: pygad in c:\users\my home\appdata\local\programs\python\python311\lib\site-packages (3.0.1)
Requirement already satisfied: cloudpickle in c:\users\my home\appdata\local\programs\python\python311\lib\site-packages (from pygad) (2.2.1)
Requirement already satisfied: matplotlib in c:\users\my home\appdata\local\programs\python\python311\lib\site-packages (from pygad) (3.7.1)
Requirement already satisfied: numpy in c:\users\my home\appdata\local\programs\python\python311\lib\site-packages (from pygad) (1.24.3)
Requirement already satisfied: contourpy>=1.0.1 in c:\users\my home\appdata\local\programs\python\python311\lib\site-packages (from matplotlib->pygad) (1.0.7)
Requirement already satisfied: cycler>=0.10 in c:\users\my home\appdata\local\programs\python\python311\lib\site-packages (from matplotlib->pygad) (0.11.0)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\my home\appdata\local\programs\python\python311\lib\site-packages (from matplotlib->pygad) (4.39.4)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\my home\appdata\local\programs\python\python311\lib\site-packages (from matplotlib->pygad) (1.4.4)
Requirement already satisfied: packaging>=20.0 in c:\users\my home\appdata\local\programs\python\python311\lib\site-packages (from matplotlib->pygad) (23.1)
Requirement already satisfied: pillow>=6.2.0 in c:\users\my home\appdata\local\programs\python\python311\lib\site-packages (from matplotlib->pygad) (9.5.0)
Requirement already satisfied: pyparsing>=2.3.1 in c:\users\my home\appdata\local\programs\python\python311\lib\site-packages (from matplotlib->pygad) (3.0.9)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\my home\appdata\local\programs\python\python311\lib\site-packages (from matplotlib->pygad) (2.8.2)
Requirement already satisfied: six>=1.5 in c:\users\my home\appdata\local\programs\python\python311\lib\site-packages (from python-dateutil>=2.7->matplotlib->pygad) (1.16.0)
Note: you may need to restart the kernel to use updated packages.

[notice] A new release of pip available: 22.3.1 -> 23.1.2

[notice] To update, run: python.exe -m pip install --upgrade pip

In [5]:

```
1 import numpy
2 import matplotlib.pyplot as plt
3 import pygad
```

In [9]:

```
1 cluster1_num_samples=10
2 cluster1_x1_start=0
3 cluster1_x1_end=5
4 cluster1_x2_start=2
5 cluster1_x2_end=6
6 cluster1_x1=numpy.random.random(size=(cluster1_num_samples))
7 cluster1_x1=cluster1_x1*(cluster1_x1_end-cluster1_x1_start)+cluster1_x1_start
8 cluster1_x2=numpy.random.random(size=(cluster1_num_samples))
9 cluster1_x2=cluster1_x2*(cluster1_x2_end-cluster1_x2_start)+cluster1_x2_start
10 cluster2_num_samples=10
11 cluster2_x1_start=10
12 cluster2_x1_end=15
13 cluster2_x2_start=8
14 cluster2_x2_end=12
15 cluster2_x1=numpy.random.random(size=(cluster2_num_samples))
16 cluster2_x1=cluster2_x1*(cluster2_x1_end-cluster2_x1_start)+cluster2_x1_start
17 cluster2_x2=numpy.random.random(size=(cluster2_num_samples))
18 cluster2_x2=cluster2_x2*(cluster2_x2_end-cluster2_x2_start)+cluster2_x2_start
```

In [11]:

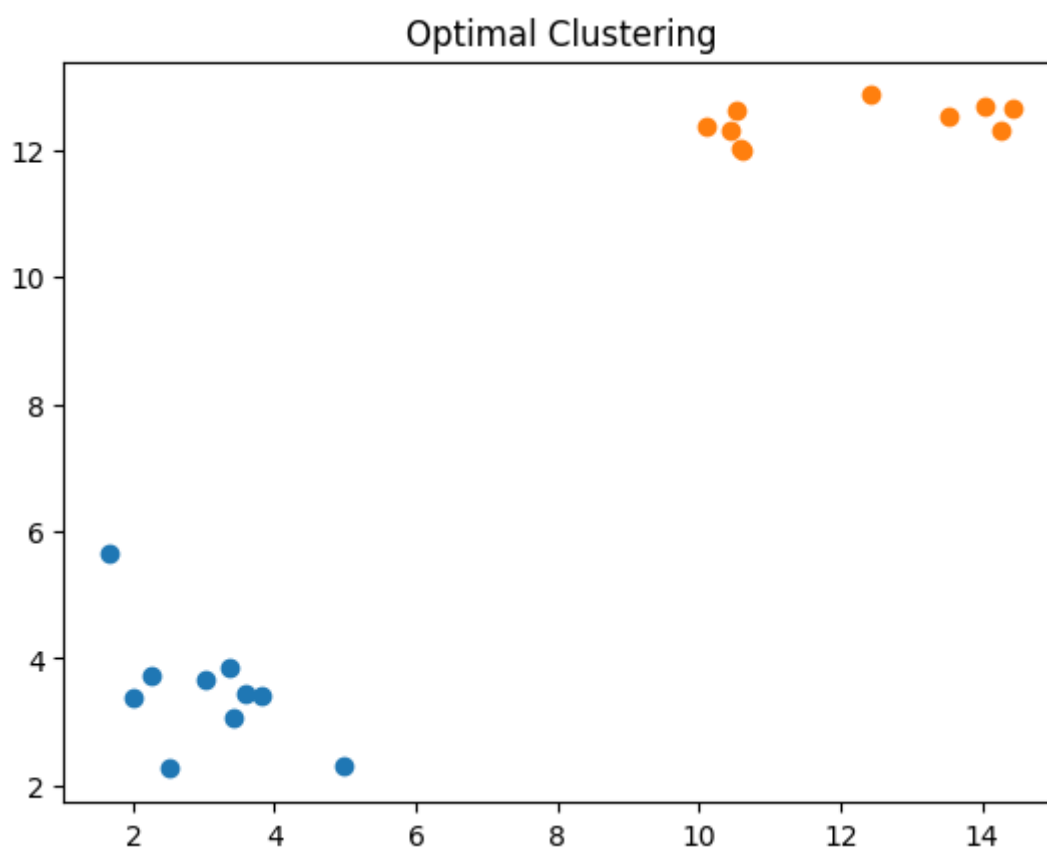
```
1 c1=numpy.array([cluster1_x1,cluster1_x2]).T
2 c2=numpy.array([cluster2_x1,cluster2_x2]).T
3 data=numpy.concatenate((c1,c2),axis=0)
4 data
```

Out[11]:

```
array([[ 3.82485063,  3.43029979],
       [ 3.02723282,  3.65385083],
       [ 4.99044526,  2.32553444],
       [ 2.5243841 ,  2.26495489],
       [ 1.6600979 ,  5.64756281],
       [ 2.00167442,  3.38012241],
       [ 2.25838385,  3.71970117],
       [ 3.42949496,  3.06335032],
       [ 3.59857224,  3.43206905],
       [ 3.35659426,  3.85323178],
       [10.57201042, 12.02157937],
       [10.62281105, 12.01225876],
       [10.4350032 , 12.30942061],
       [10.51702484, 12.62615267],
       [12.43103145, 12.86609733],
       [14.43278725, 12.64651058],
       [14.0363077 , 12.68925019],
       [14.26391689, 12.31451156],
       [13.54356577, 12.52039524],
       [10.09396682, 12.38187009]])
```

In [12]:

```
1 plt.scatter(cluster1_x1,cluster1_x2)
2 plt.scatter(cluster2_x1,cluster2_x2)
3 plt.title("Optimal Clustering")
4 plt.show()
```



In [13]:

```
1 def euclidean_distance(X,Y):
2     return numpy.sqrt(numpy.sum(numpy.power(X-Y,2),axis=1))
```

In [23]:

```

1 def cluster_data(solution,solution_idx):
2     global num_cluster,data
3     feature_vector_length=data.shape[1]
4     cluster_centers=[]
5     all_clusters_dists=[]
6     clusters=[]
7     clusters_sum_dist=[]
8     for clust_idx in range(num_clusters):
9         cluster_centers.append(solution[feature_vector_length*clust_idx:feature_vector_length*(clust_idx+1)])
10        cluster_center_dists=euclidean_distance(data,cluster_centers[clust_idx])
11        all_clusters_dists.append(numpy.array(cluster_center_dists))
12    cluster_centers=numpy.array(cluster_centers)
13    all_clusters_dists=numpy.array(all_clusters_dists)
14    cluster_indices=numpy.argmin(all_clusters_dists,axis=0)
15    for clust_idx in range(num_clusters):
16        clusters.append(numpy.where(cluster_indices == clust_idx)[0])
17        if len(clusters[clust_idx]) == 0:
18            clusters_sum_dist.append(0)
19        else:
20            clusters_sum_dist.append(numpy.sum(all_clusters_dists[clust_idx, clusters[clust_idx]]))
21    clusters_sum_dist = numpy.array(clusters_sum_dist)
22    return cluster_centers, all_clusters_dists, cluster_indices, clusters, clusters_sum_dist
23

```

In [24]:

```

1 def fitness_func(ga_instance,solution, solution_idx):
2     _, _, _, _, clusters_sum_dist = cluster_data(solution, solution_idx)
3     fitness = 1.0 / (numpy.sum(clusters_sum_dist) + 0.00000001)
4     return fitness

```

In [25]:

```

1 num_clusters = 2
2 num_genes = num_clusters * data.shape[1]
3 ga_instance = pygad.GA(num_generations=100,
4                         sol_per_pop=10,
5                         num_parents_mating=5,
6                         init_range_low=-6,
7                         init_range_high=20,
8                         keep_parents=2,
9                         num_genes=num_genes,
10                        fitness_func=fitness_func,
11                        suppress_warnings=True)
12 ga_instance.run()

```

In [26]:

```

1 best_solution, best_solution_fitness, best_solution_idx = ga_instance.best_solution(
2 print("Best solution is {bs}".format(bs=best_solution))
3 print("Fitness of the best solution is {bsf}".format(bsf=best_solution_fitness))
4 print("Best solution found after {gen} generations".format(gen=ga_instance.best_solu

```

Best solution is [11.67524327 12.43313082 3.1741446 3.49773255]

Fitness of the best solution is 0.036638122623876536

Best solution found after 69 generations

In [27]:

```

1 cluster_centers, all_clusters_dists, cluster_indices, clusters, clusters_sum_dist= c

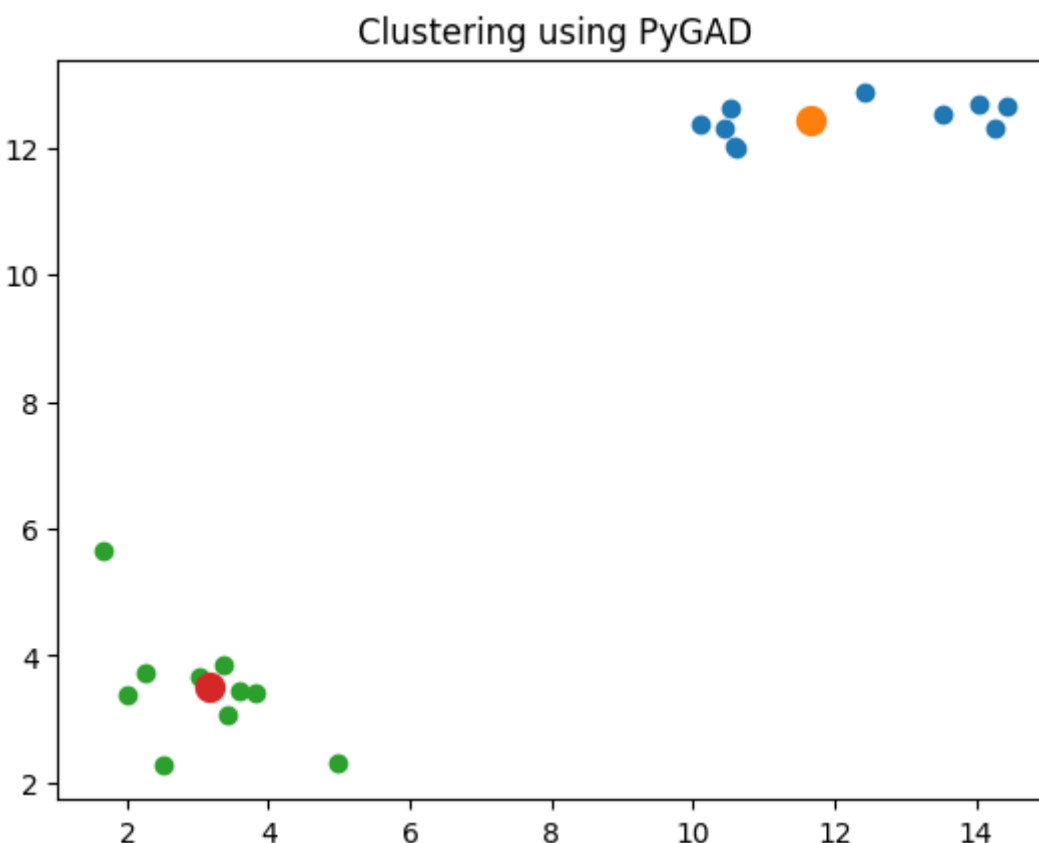
```

In [30]:

```

1 for cluster_idx in range(num_clusters):
2     cluster_x = data[clusters[cluster_idx], 0]
3     cluster_y = data[clusters[cluster_idx], 1]
4     plt.scatter(cluster_x, cluster_y)
5     plt.scatter(cluster_centers[cluster_idx, 0], cluster_centers[cluster_idx, 1], li
6 plt.title("Clustering using PyGAD")
7 plt.show()

```



In []:

```

1

```

