

In [1]:

```
pip install pygad
```

Collecting pygad

Downloading pygad-3.0.1-py3-none-any.whl (67 kB)

```

0.0/68.0 kB ? eta -:--:--
-----
30.7/68.0 kB ? eta -:--:--
-----
30.7/68.0 kB ? eta -:--:--
-----
30.7/68.0 kB ? eta -:--:--
-----
30.7/68.0 kB ? eta -:--:--
-----
61.4/68.0 kB 233.8 kB/s eta 0:00:01
-----
68.0/68.0 kB 245.6 kB/s eta 0:00:00

```

Collecting cloudpickle (from pygad)

Downloading cloudpickle-2.2.1-py3-none-any.whl (25 kB)

Requirement already satisfied: matplotlib in c:\users\monim\appdata\local\programs\python\python310\lib\site-packages (from pygad) (3.7.1)

Requirement already satisfied: numpy in c:\users\monim\appdata\local\programs\python\python310\lib\site-packages (from pygad) (1.24.3)

Requirement already satisfied: contourpy&gt;=1.0.1 in c:\users\monim\appdata\local\programs\python\python310\lib\site-packages (from matplotlib-&gt;pygad) (1.0.7)

Requirement already satisfied: cyclers&gt;=0.10 in c:\users\monim\appdata\local\programs\python\python310\lib\site-packages (from matplotlib-&gt;pygad) (0.11.0)

Requirement already satisfied: fonttools&gt;=4.22.0 in c:\users\monim\appdata\local\programs\python\python310\lib\site-packages (from matplotlib-&gt;pygad) (4.39.4)

Requirement already satisfied: kiwisolver&gt;=1.0.1 in c:\users\monim\appdata\local\programs\python\python310\lib\site-packages (from matplotlib-&gt;pygad) (1.4.4)

Requirement already satisfied: packaging&gt;=20.0 in c:\users\monim\appdata\local\programs\python\python310\lib\site-packages (from matplotlib-&gt;pygad) (23.1)

Requirement already satisfied: pillow&gt;=6.2.0 in c:\users\monim\appdata\local\programs\python\python310\lib\site-packages (from matplotlib-&gt;pygad) (9.5.0)

Requirement already satisfied: pyparsing&gt;=2.3.1 in c:\users\monim\appdata\local\programs\python\python310\lib\site-packages (from matplotlib-&gt;pygad) (3.0.9)

Requirement already satisfied: python-dateutil&gt;=2.7 in c:\users\monim\appdata\local\programs\python\python310\lib\site-packages (from matplotlib-&gt;pygad) (2.8.2)

Requirement already satisfied: six&gt;=1.5 in c:\users\monim\appdata\local\programs\python\python310\lib\site-packages (from python-dateutil&gt;=2.7-&gt;matplotlib-&gt;pygad) (1.16.0)

Installing collected packages: cloudpickle, pygad

Successfully installed cloudpickle-2.2.1 pygad-3.0.1

Note: you may need to restart the kernel to use updated packages.

In [2]:

```
import numpy
import matplotlib.pyplot
import pygad
```

In [3]:

```

cluster1_num_samples = 10
cluster1_x1_start = 0
cluster1_x1_end = 5
cluster1_x2_start = 2
cluster1_x2_end = 6
cluster1_x1 = numpy.random.random(size=(cluster1_num_samples))
cluster1_x1 = cluster1_x1 * (cluster1_x1_end - cluster1_x1_start) + cluster1_x1_start
cluster1_x2 = numpy.random.random(size=(cluster1_num_samples))
cluster1_x2 = cluster1_x2 * (cluster1_x2_end - cluster1_x2_start) + cluster1_x2_start
cluster2_num_samples = 10
cluster2_x1_start = 10
cluster2_x1_end = 15
cluster2_x2_start = 8
cluster2_x2_end = 12
cluster2_x1 = numpy.random.random(size=(cluster2_num_samples))
cluster2_x1 = cluster2_x1 * (cluster2_x1_end - cluster2_x1_start) + cluster2_x1_start
cluster2_x2 = numpy.random.random(size=(cluster2_num_samples))
cluster2_x2 = cluster2_x2 * (cluster2_x2_end - cluster2_x2_start) + cluster2_x2_start

```

In [4]:

```

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

```

Out[4]:

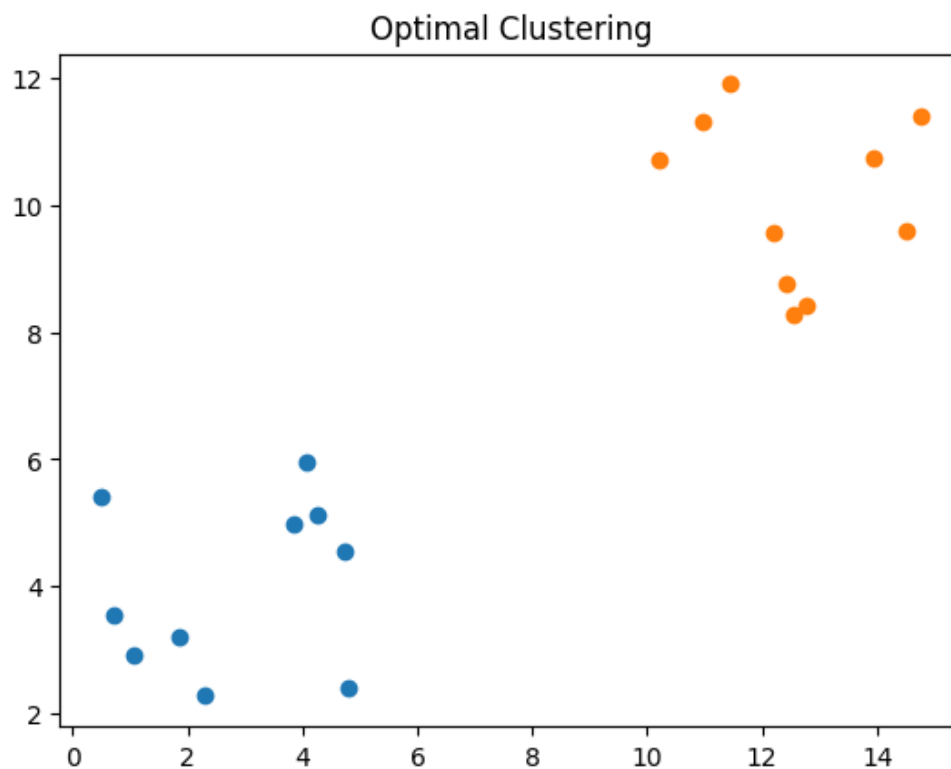
```

array([[ 0.71941003,  3.53653324],
       [ 4.26191525,  5.12498592],
       [ 4.08148235,  5.9503819 ],
       [ 0.49312755,  5.40829267],
       [ 1.84880109,  3.20500963],
       [ 1.05101371,  2.93058502],
       [ 4.72736921,  4.55212511],
       [ 3.84367792,  4.99038507],
       [ 2.29618804,  2.27212718],
       [ 4.80447106,  2.38829288],
       [12.55668469,  8.28758265],
       [12.76231819,  8.41746915],
       [10.96550718, 11.32385557],
       [12.19414733,  9.5598373 ],
       [14.75601296, 11.40398415],
       [13.93254823, 10.75638683],
       [12.42213152,  8.76929899],
       [11.45008114, 11.9058583 ],
       [14.49917346,  9.59797736],
       [10.19189558, 10.72382352]])

```

In [5]:

```
matplotlib.pyplot.scatter(cluster1_x1, cluster1_x2)
matplotlib.pyplot.scatter(cluster2_x1, cluster2_x2)
matplotlib.pyplot.title("Optimal Clustering")
matplotlib.pyplot.show()
```



In [12]:

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

In [16]:

```
def cluster_data(solution, solution_idx):
    global num_cluster, data
    feature_vector_length = data.shape[1]
    cluster_centers = []
    all_clusters_dists = []
    clusters = []
    clusters_sum_dist = []
    for clust_idx in range(num_clusters):
        cluster_centers.append(solution[feature_vector_length*clust_idx:feature_vector_length*(clust_idx+1)])
        cluster_center_dists = euclidean_distance(data, cluster_centers[clust_idx])
        all_clusters_dists.append(numpy.array(cluster_center_dists))
        cluster_centers = numpy.array(cluster_centers)
        all_clusters_dists = numpy.array(all_clusters_dists)
        cluster_indices = numpy.argmin(all_clusters_dists, axis=0)
        for clust_idx in range(num_clusters):
            clusters.append(numpy.where(cluster_indices == clust_idx)[0])

    if len(clusters[clust_idx]) == 0:
        clusters_sum_dist.append(0)
    else:
        clusters_sum_dist.append(numpy.sum(all_clusters_dists[clust_idx, clusters[clust_idx]]))
    clusters_sum_dist = numpy.array(clusters_sum_dist)
    return cluster_centers, all_clusters_dists, cluster_indices, clusters, clusters_sum_dist
```

In [17]:

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

In [19]:

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

In [21]:

```
best_solution, best_solution_fitness, best_solution_idx = ga_instance.best_solution()
print("Best solution is {bs}".format(bs=best_solution))
print("Fitness of the best solution is {bsf}".format(bsf=best_solution_fitness))
print("Best solution found after {gen} generations".format(gen=ga_instance.best_solution_generation))
```

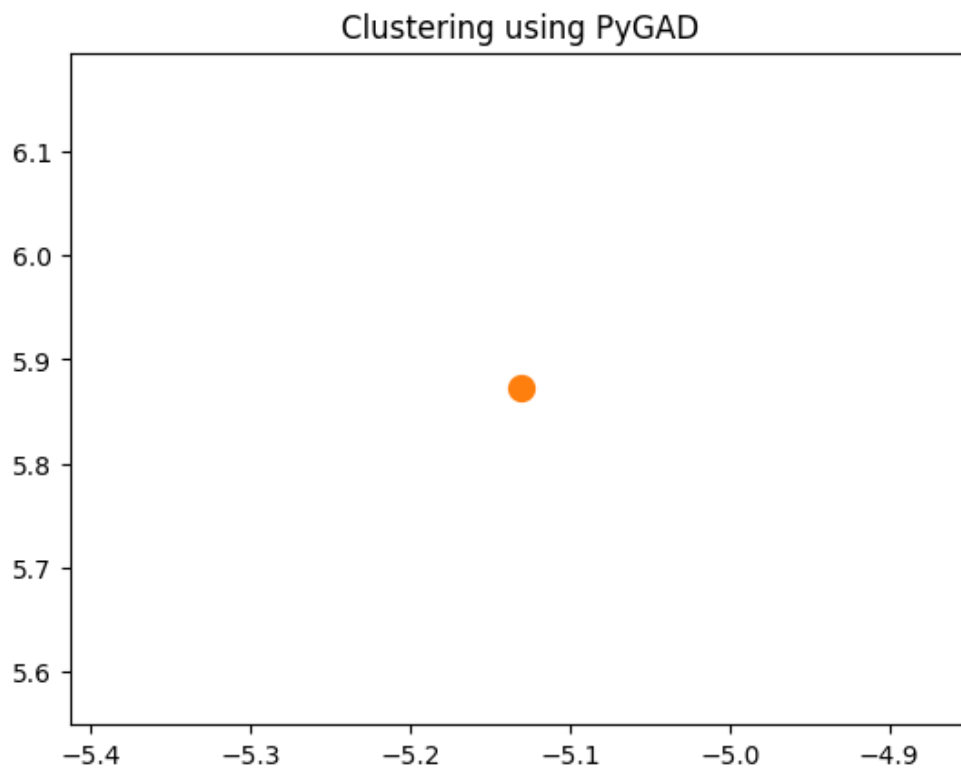
```
Best solution is [ 8.28558978  0.8396686  -5.13091634  5.87184623]
Fitness of the best solution is 100000000.0
Best solution found after 0 generations
```

In [31]:

```
cluster_centers, all_clusters_dists, cluster_indices, clusters, clusters_sum_dist = cluster_data(best_sol
```

In [34]:

```
for cluster_idx in range(num_clusters):  
    cluster_x = data[clusters[cluster_idx], 0]  
    cluster_y = data[clusters[cluster_idx], 1]  
matplotlib.pyplot.scatter(cluster_x, cluster_y)  
matplotlib.pyplot.scatter(cluster_centers[cluster_idx, 0], cluster_centers[cluster_idx, 1], linewidth=1)  
matplotlib.pyplot.title("Clustering using PyGAD")  
matplotlib.pyplot.show()
```



In [ ]: