In [1]:

```
pip install pygad
```

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

In [2]:

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

In [3]:

```python
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]:

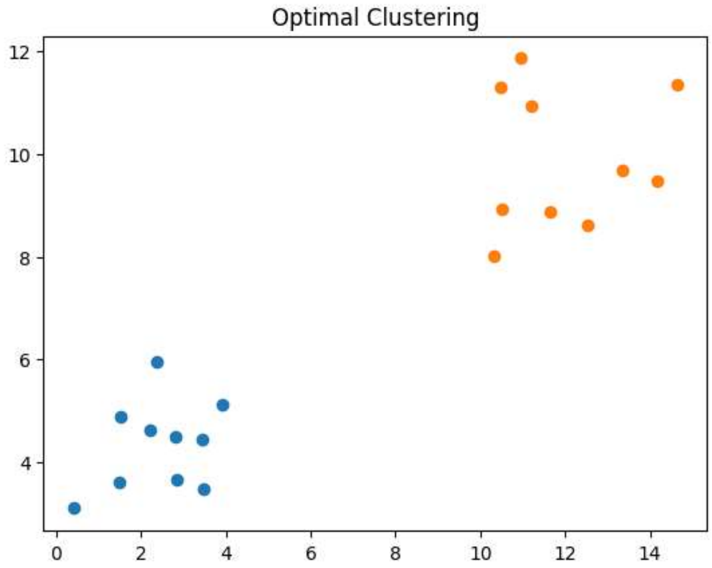
```python
c1 = numpy.array([cluster1_x1, cluster1_x2]).T
c2 = numpy.array([cluster2_x1, cluster2_x2]).T
data = numpy.concatenate((c1, c2), axis=0)
data
```

Out[4]:

```
array([[ 0.4109114 ,  3.09762203],
       [ 2.81530855,  4.49997013],
       [ 2.36209091,  5.94770276],
       [ 3.90556218,  5.12972814],
       [ 1.4777936 ,  3.59769395],
       [ 3.47858903,  3.47236451],
       [ 2.85344877,  3.66783625],
       [ 3.43386639,  4.43342912],
       [ 1.5280868 ,  4.87352422],
       [ 2.20503755,  4.62669532],
       [11.19017463, 10.92232539],
       [14.6334393 , 11.34099479],
       [10.46216783, 11.29992993],
       [10.51456915,  8.93146112],
       [10.33400337,  8.01379586],
       [12.53208493,  8.62347254],
       [13.35201619,  9.69260887],
       [14.16266456,  9.46864143],
       [11.63115869,  8.8810934 ],
       [10.94809856, 11.8612596 ]])
```

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 [6]:

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

In [11]:

```
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 [12]:

```
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
```

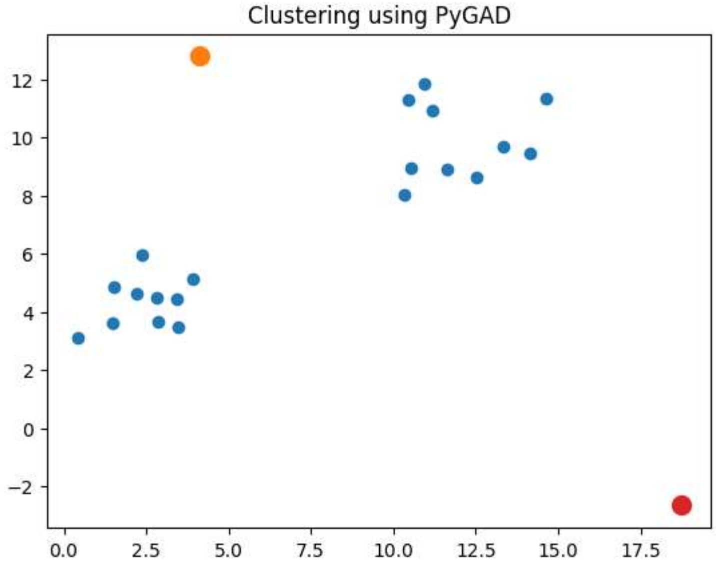
In [13]:
```
num_clusters = 2
num_genes = num_clusters * data.shape[1]

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 [14]:
```
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 [ 4.10834358 12.78586314 18.71256362 -2.66612227]
Fitness of the best solution is 100000000.0
Best solution found after 0 generations
```

In [18]:
```
cluster_centers, all_clusters_dists, cluster_indices, clusters, clusters_sum_dist= cluster_data(best_solution, best_solution_idx)
```

In [19]:
```
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], linewidths=5)
matplotlib.pyplot.title("Clustering using PyGAD")
matplotlib.pyplot.show()
```



In [ ]:
```

```