

```
In [1]: pip install pygad
```

```
Collecting pygad
  Downloading pygad-3.0.1-py3-none-any.whl (67 kB)
    0.0/68.0 kB ? eta -:--:--
    10.2/68.0 kB ? eta -:--:--
    10.2/68.0 kB ? eta -:--:--
    ----- 61.4/68.0 kB 409.6 kB/s eta 0:00:01
    ----- 68.0/68.0 kB 412.2 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\sudheer\appdata\local\programs\python\python310\lib\site-packages (from pygad) (3.7.1)
Requirement already satisfied: numpy in c:\users\sudheer\appdata\local\programs\python\python310\lib\site-packages (from pygad) (1.24.3)
Requirement already satisfied: contourpy>=1.0.1 in c:\users\sudheer\appdata\local\programs\python\python310\lib\site-packages (from matplotlib->pygad) (1.0.7)
Requirement already satisfied: cycler>=0.10 in c:\users\sudheer\appdata\local\programs\python\python310\lib\site-packages (from matplotlib->pygad) (0.11.0)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\sudheer\appdata\local\programs\python\python310\lib\site-packages (from matplotlib->pygad) (4.39.4)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\sudheer\appdata\local\programs\python\python310\lib\site-packages (from matplotlib->pygad) (1.4.4)
Requirement already satisfied: packaging>=20.0 in c:\users\sudheer\appdata\local\programs\python\python310\lib\site-packages (from matplotlib->pygad) (23.1)
Requirement already satisfied: pillow>=6.2.0 in c:\users\sudheer\appdata\local\programs\python\python310\lib\site-packages (from matplotlib->pygad) (9.5.0)
Requirement already satisfied: pyparsing>=2.3.1 in c:\users\sudheer\appdata\local\programs\python\python310\lib\site-packages (from matplotlib->pygad) (3.0.9)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\sudheer\appdata\local\programs\python\python310\lib\site-packages (from matplotlib->pygad) (2.8.2)
Requirement already satisfied: six>=1.5 in c:\users\sudheer\appdata\local\programs\python\python310\lib\site-packages (from python-dateutil>=2.7->matplotlib->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
```

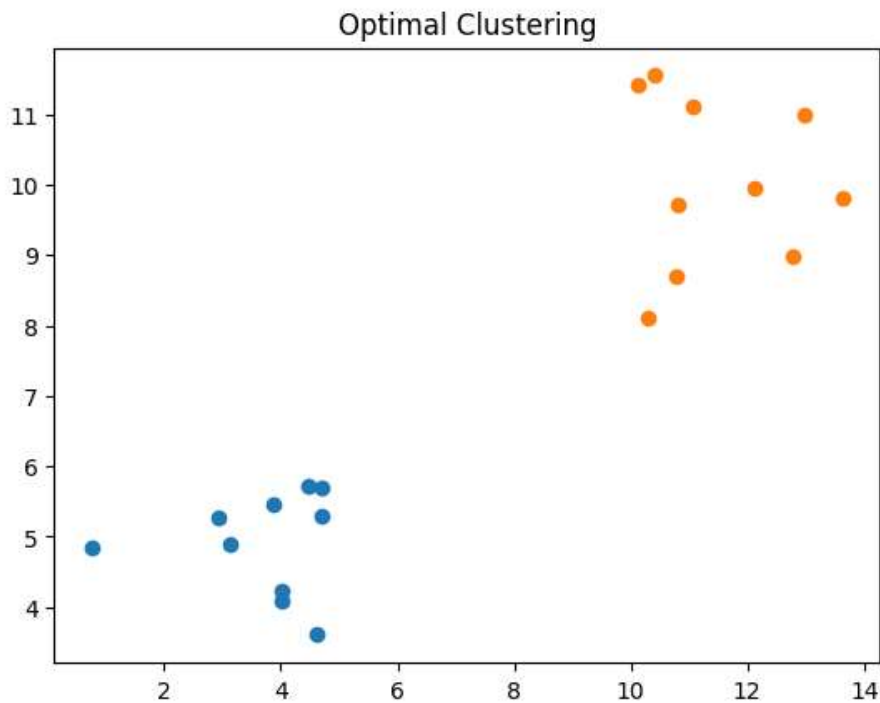
```
In [3]: cluster1_num_samples = 10
cluster1_x1_start = 0
cluster1_x1_end = 5
cluster1_x2_start = 2
cluster1_x2_end = 6
cluster1_x1 = numpy.random.random(size=(cluster1_num_samples))
cluster1_x1 = cluster1_x1 * (cluster1_x1_end - cluster1_x1_start) + cluster1_x1_start
cluster1_x2 = numpy.random.random(size=(cluster1_num_samples))
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([[ 4.69438751,  5.705873  ],
 [ 3.87131639,  5.46526255],
 [ 4.01735929,  4.09274534],
 [ 4.71270115,  5.30175445],
 [ 0.76250531,  4.8385625  ],
 [ 4.01107908,  4.23787743],
 [ 3.12996768,  4.88656676],
 [ 4.62483192,  3.60228952],
 [ 2.94531821,  5.26716143],
 [ 4.47995893,  5.72068074],
 [10.80892592,  9.72238712],
 [12.96991329, 10.98708425],
 [10.30642363,  8.10334115],
 [13.63311567,  9.81764977],
 [12.790838  ,  8.98383032],
 [10.78678092,  8.69031058],
 [11.06753053, 11.11285477],
 [10.12245169, 11.41053333],
 [10.41070042, 11.5486667  ],
 [12.11396885,  9.96121636]])
```

```
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 [9]: 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 [30]: 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 [31]: 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)

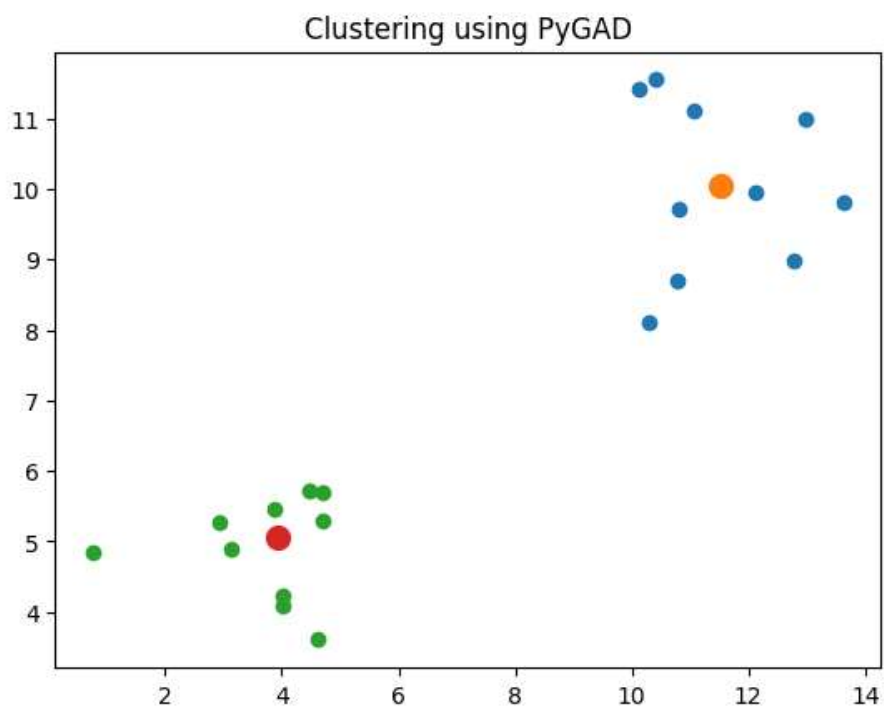
    ga_instance.run()
```

```
In [32]: 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 [11.52753295 10.03584406 3.94123104 5.05784174]
    Fitness of the best solution is 0.03675736812750598
    Best solution found after 93 generations
```

```
In [34]: cluster_centers, all_clusters_dists, cluster_indices, clusters, clusters_sum_dist = cluster_data(best_solution, best_solution_idx)
```

```
In [35]: 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], line
matplotlib.pyplot.title("Clustering using PyGAD")
matplotlib.pyplot.show()
```



In []: