EXERCISE	7. FUZZY C-MEANS CLUSTERING
DATE	15.09.2024

AIM:
Implement fuzzy C-means clustering algorithm to cluster the given dataset.

DESCRIPTION:

Fuzzy Clustering is a type of clustering algorithm in machine learning that allows a data point to belong to more than one cluster with different degrees of membership.

Applications in several fields of Fuzzy clustering:

1. **Image segmentation:** Fuzzy clustering can be used to segment images by grouping pixels with similar properties together, such as color or texture.

2. **Pattern recognition:** Fuzzy clustering can be used to identify patterns in large datasets by grouping similar data points together.

3. **Marketing:** Fuzzy clustering can be used to segment customers based on their preferences and purchasing behavior, allowing for more targeted marketing campaigns.

4. **Medical diagnosis:** Fuzzy clustering can be used to diagnose diseases by grouping patients with similar symptoms together.

5. **Environmental monitoring:** Fuzzy clustering can be used to identify areas of environmental concern by grouping together areas with similar pollution levels or other environmental indicators.

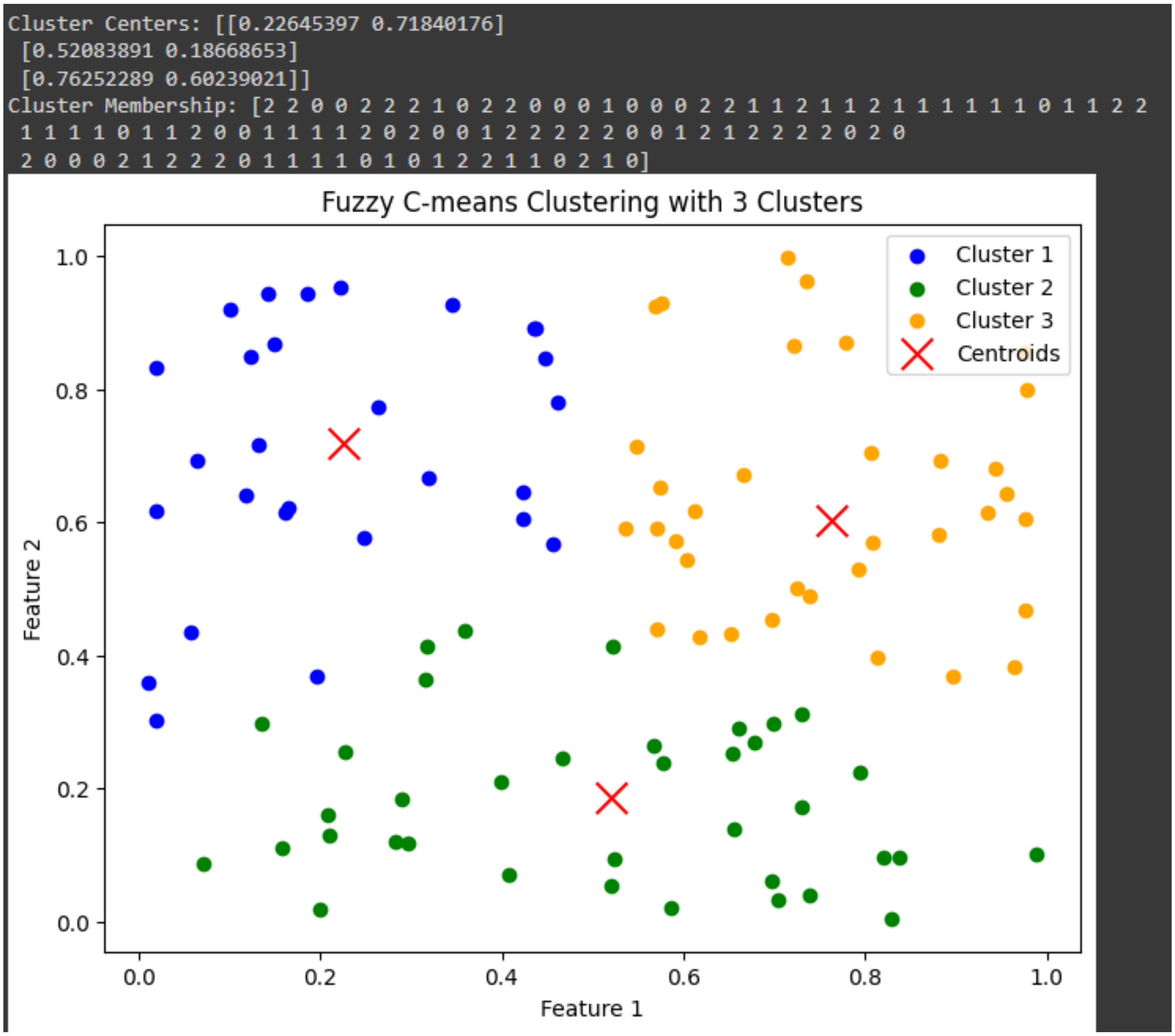
6. **Traffic flow analysis:** Fuzzy clustering can be used to analyze traffic flow patterns by grouping similar traffic patterns together, allowing for better traffic management and planning.

7. **Risk assessment:** Fuzzy clustering can be used to identify and quantify risks in various fields, such as finance, insurance, and engineering.

PROGRAM:

```
import numpy as np
import skfuzzy as fuzz
import matplotlib.pyplot as plt

np.random.seed(0)
data = np.random.rand(100, 2)
n_clusters = 3

# Apply fuzzy c-means clustering
cntr, u, _, _, _, _, _ = fuzz.cluster.cmeans(
    data.T, n_clusters, 2, error=0.005, maxiter=1000, init=None)
cluster_membership = np.argmax(u, axis=0)
print('Cluster Centers:', cntr)
print('Cluster Membership:', cluster_membership)
plt.figure(figsize=(8, 6))

# Scatter plot for each cluster
colors = ['b', 'g', 'orange']
for i in range(n_clusters):
    plt.scatter(data[cluster_membership == i, 0], data[cluster_membership == i, 1],
                c=colors[i], label=f'Cluster {i+1}')
plt.scatter(cntr[:, 0], cntr[:, 1], marker='x', color='red', s=200, label='Centroids')

# Add labels and legend
plt.xlabel('Feature 1')
plt.ylabel('Feature 2')
plt.title(f'Fuzzy C-means Clustering with {n_clusters} Clusters')
plt.legend()
plt.show()
```

OUTPUT:



```
Cluster Centers: [[0.22645397 0.71840176]
 [0.52083891 0.18668653]
 [0.76252289 0.60239021]]
Cluster Membership: [2 2 0 0 2 2 2 1 0 2 2 0 0 0 1 0 0 0 2 2 1 1 2 1 1 2 1 1 1 1 1 1 0 1 1 2 2
 1 1 1 1 0 1 1 2 0 0 1 1 1 1 2 0 2 0 0 1 2 2 2 2 2 0 0 1 2 1 2 2 2 2 0 2 0
 2 0 0 0 2 1 2 2 2 0 1 1 1 1 0 1 0 1 2 2 1 1 0 2 1 0]
```

RESULT:

The above questions are coded and solved using the fuzzy C-means clustering algorithm.