

## Unsupervised Machine Learning Demo using K-Means (Python)

### OBJECTIVE:

Demonstrate a simple Unsupervised Machine Learning example using K-Means clustering in Python.

---

### PROBLEM STATEMENT:

We have customer data with:

1. Annual Income
2. Spending Score

We do NOT have labels.

We want to group customers automatically using clustering.

---

### STEP 1: Install Required Libraries

```
pip install numpy pandas matplotlib scikit-learn
```

---

### STEP 2: Import Libraries

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
```

---

### STEP 3: Create Sample Dataset

```
data = {
    "Income": [15, 16, 17, 18, 19, 60, 62, 63, 64, 65],
    "Spending": [39, 41, 40, 42, 38, 80, 82, 81, 83, 79]
}
```

```
df = pd.DataFrame(data)
print(df)
```

---

### STEP 4: Visualize Data (Before Clustering)

```
plt.scatter(df["Income"], df["Spending"])
plt.xlabel("Income")
plt.ylabel("Spending Score")
plt.title("Customer Data")
plt.show()
```

---

#### STEP 5: Apply K-Means

```
kmeans = KMeans(n_clusters=2)
kmeans.fit(df)
```

---

#### STEP 6: Get Cluster Labels

```
labels = kmeans.labels_
print(labels)
```

---

#### STEP 7: Add Cluster Column

```
df["Cluster"] = labels
print(df)
```

---

#### STEP 8: Visualize Clusters

```
plt.scatter(df["Income"], df["Spending"], c=df["Cluster"])
plt.xlabel("Income")
plt.ylabel("Spending Score")
plt.title("Customer Segments")
plt.show()
```

---

#### INTERPRETATION:

Cluster 0 → Low Income, Low Spending  
Cluster 1 → High Income, High Spending

---

#### WHY THIS IS UNSUPERVISED LEARNING:

- No target column
- No labels provided
- Algorithm discovers patterns automatically

-----  
BONUS: Change Number of Clusters

```
kmeans = KMeans(n_clusters=3)
kmeans.fit(df)
df["Cluster"] = kmeans.labels_
```