III B.Tech – II Semester
(23E05601L) MACHINE LEARNING LAB

Experiment No: 12

◈ AIM

To implement the **K-Means clustering algorithm** on a dataset, evaluate performance using the sum of Euclidean distances, and test performance for different values of K.

◈ THEORY

📌 Clustering

Clustering is an **unsupervised learning** technique used to group similar data points.

📌 K-Means Algorithm

K-Means forms **K clusters** by:

1. Selecting K random centroids
2. Assigning points to nearest centroid
3. Recomputing centroids
4. Repeating until convergence

Objective

Minimize the **sum of squared distances (SSD)** between points and their cluster centers.

Formula:

$$SSD = \sum_{i=1}^{K} \sum_{x \in C_i} ||x - \mu_i||^2$$

III B.Tech – II Semester
(23E05601L) MACHINE LEARNING LAB

Advantages

✓ Simple & Fast
✓ Works well for large datasets
✓ Easy to interpret clusters

Disadvantages

✗ Requires choosing K manually
✗ Sensitive to outliers
✗ Assumes spherical clusters

◈ USE CASE

Cluster customers based on:

- Annual Income
- Spending Score

(similar to real shopping mall customer segmentation)

◈ CODE

```
import pandas as pd
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score

# Load CSV
df = pd.read_csv("exp12_random_forest.csv")
print(df)

X = df[["Age", "Salary"]]
y = df["Buys_Product"]

# Split
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.3, random_state=0
)
```

III B.Tech – II Semester
(23E05601L) MACHINE LEARNING LAB

```
# Random Forest model
rf = RandomForestClassifier(n_estimators=10, random_state=0)
rf.fit(X_train, y_train)

# Predictions
pred = rf.predict(X_test)
print("Predictions:", pred)

# Accuracy
print("Accuracy:", accuracy_score(y_test, pred))
```

OUTPUT

```
K = 2
Cluster Centers:
 [[82. 78.]
  [30. 40.]]
Sum of Distances (Inertia): 700.4

K = 3
Cluster Centers:
 [[92.5 87.5]
  [27.5 37.5]
  [85.  75. ]]
Sum of Distances (Inertia): 250.3

K = 4
Cluster Centers:
 [[25. 35.]
  [95. 90.]
  [85. 75.]
  [35. 45.]]
Sum of Distances (Inertia): 53.5
```

INTERPRETATION

- SSD (inertia) **decreased** as K increased
- K=4 gives best compact clusters
- But too many clusters can overfit → use **Elbow Method** to choose K

Age	Salary	Buys_Product
18	15000	0
22	18000	0
25	22000	0
30	26000	1
35	30000	1
40	35000	1
45	38000	1
50	42000	1
55	45000	1
60	50000	1