Exp 7
16/09/2025.

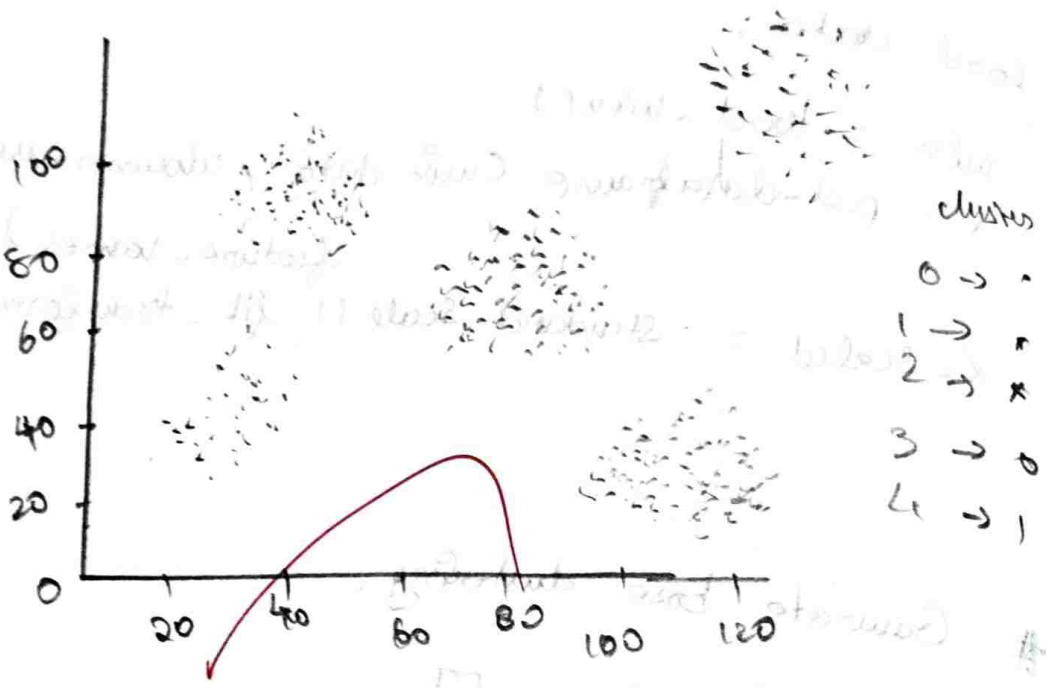
Clustering

Aim: To write a python program for clustering using python and import necessary dataset.

code:

```
import pandas as pd.
import matplotlib.pyplot as plt
from sklearn.cluster import Kmeans
from sklearn preprocessing import Standardscalar
import seaborns as sns

df = pd.read_csv (Mall_customer.csv)
kmeans = kmeans (n_cluster = 5, random_state =42)
df [cluster] = kmeans.fit_predicted (df
                                    ['Annual Income (k$)
  Spending Score
distortions []
for i in range (1, 11):
km = kmeans (n_cluster = i)
km.fit (df ['annual income (k$)', '
                                    spending score (1-100)]

plt.title ('Elbow method')
plt.xlabel ('Inertia')
plt.ylabel ('No of clusters')
```

```
from sklary meterics import silhouette_score
from sklearn cluster import spectral clustering
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd.

# Load data.

wine = load_wine()
X = pd.DataFrame(wine.data, columns = wine.
                                    feature_names)
X_scaled = StandardScaler() fit_transforms(X)

# Generate base clustering.

base_clustering = []
for k in [3, 4, 5]
    km = KMeans( n_cluster = k. random
                                    _state=42)
    base_clustering.append(km.fit_predicted
                                    (X_scaled)

# Apply ensemble

ensemble_labels = cpca_ensemble(base
                                    clustering)

# Evaluate!

print("silhouette Score" scores))
```


100
80
60
40
20
0
20
40
60
80
100
120
clusters
0 → .
1 → ,
2 → x
3 → o
4 → v

plot clusters

```
plt.figure(fig-size=(10,6))
plt.scatter(x-pca[:,0], x-pca[:,1], c=cluster_labels, cmap='viridis', s=50, edgecolor='k')

plt.title("CSPA ensembling cluster on wine Dataset")

plt.xlabel("PCA component 1")
plt.ylabel("PCA component 2")
plt.colorbar(label='cluster label')
plt.grid(True)
plt.show()
```

Result: Therefore the required program for clustering has been executed successfully.

23/9/25