ML Experiment 5

DATE:

ML Experiment 5

Kreena Shah
60004210243
C'32

9 Mar'23

Aim : To implement KNN algorithm

Theory :
K-Nearest Neighbours (KNN) algorithm is a supervised machine learning method employed to tackle classification & regression problems

Distance Metrics used in KNN
(1) Euclidean Distance
$$dist(x, X_i) = \sqrt{\sum_{j=1}^{d} (x_j - X_{ij})^2}$$

(2) Manhattan Distance

$$d(x,y) = \sum_{i=1}^{n} |x_i - y_i|$$

(3) Minkowski Distance

$$d(x,y) = \left(\sum_{i=1}^{n} (x_i - y_i)^p\right)^{1/p}$$

Choosing the value of K
The value of K depends on input data
If input data has more outliers, a higher value of K is

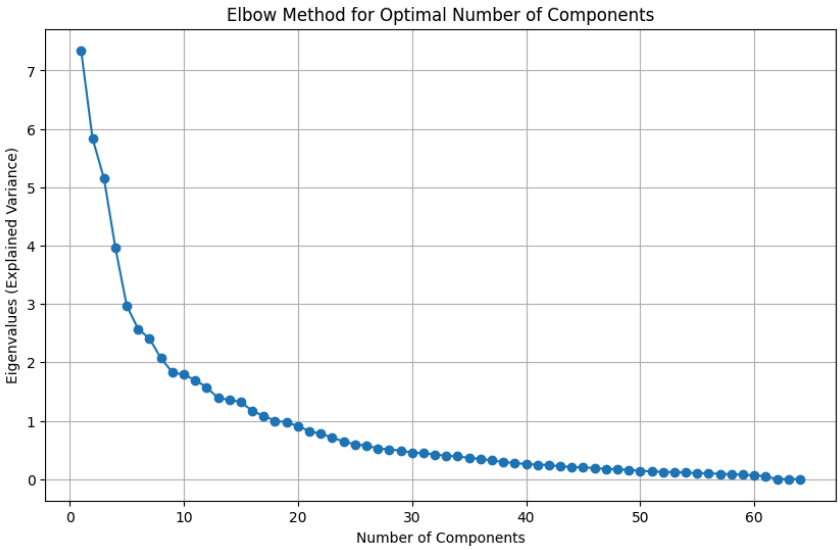
FOR EDUCATIONAL USE

DATE:

Conclusion: Thus, we understood & implement KNN algorithm.

import numpy as np
import matplotlib.pyplot as plt import pandas as pd
from sklearn.datasets import load_digits from sklearn.decomposition import PCA

```
from sklearn.preprocessing import StandardScaler digits_data
= load_digits()
X = digits_data.data scaler =
StandardScaler()
X_scaled = scaler.fit_transform(X) pca = PCA()
X_pca = pca.fit_transform(X_scaled) eigenvalues =
pca.explained_variance_ plt.figure(figsize=(10, 6))
plt.plot(range(1, len(eigenvalues) + 1), eigenvalues, marker='o', linestyle='-')
plt.title('Elbow Method for Optimal Number of Components')
plt.xlabel('Number of Components') plt.ylabel('Eigenvalues (Explained
Variance)') plt.grid(True)
plt.show() optimal_num_components =
10
X_reduced = X_pca[:, :optimal_num_components]
df_reduced = pd.DataFrame(X_reduced, columns=[f'PC{i}' for i in range(1,
optimal_num_components + 1)])
df_reduced['target'] = digits_data.target
df_reduced.to_csv('reduced_digits_dataset.csv', index=False) print("Digits Wine
dataset saved successfully.")
```




```
print(X_pca)
```

```
[[ 1.91421366e+00 -9.54501571e-01 -3.94603482e+00 ... -0.00000000e+00
   0.00000000e+00  8.24385469e-15]
 [ 5.88980330e-01  9.24635800e-01  3.92475494e+00 ...  8.37680823e-16
  -1.88056170e-17  3.08670512e-17]
 [ 1.30203906e+00 -3.17188827e-01  3.02333293e+00 ...  4.30699370e-16
   1.79508547e-17  2.15555254e-17]
```