**ML - Assignment-5**

Sravya Mannava

700747750

Semester - Spring

Course Id - CS 5710

CRN - 23922

Github Link: https://github.com/Sravya-mannava/Assignment-5

1. Principal Component Analysis

Table

Description automatically generated

Graphical user interface, application

Description automatically generated

1.a. Apply PCA on CC dataset.

A picture containing text

Description automatically generated

Description:

* PCA() performs principal component analysis on dataset x, reducing the dimensionality of the data from the original number of features to 3 principal components.
* fit\_transform() method of the PCA object is called on the data x to obtain a transformed version of the data, where each observation is represented by its three principal components.
* principalDf represents the transformed data x\_pca and three principal components
* finalDf concatenating principalDf with the last column of the original DataFrame df using pd.concat(). This is likely done to include the target variable (the variable being predicted) with the transformed data.

1.b. Apply k-means algorithm on the PCA result and report your observation if the silhouette score has improved or not?

Timeline

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Description:

* X predictor variable contains all rows of finalDf except for the last column, representing the principal components generated by PCA
* Y target variable contains only the last column of finalDf, representing the target variable.

1.c Perform Scaling+PCA+K-Means and report performance.

Scaling

Graphical user interface, text, application

Description automatically generated

PCA

Graphical user interface, application

Description automatically generated

K- means:

Graphical user interface, text, application

Description automatically generated

Description:

* fit\_transform() method of the StandardScaler instance to the feature matrix X to perform feature scaling.
* This method first computes the mean and standard deviation of each feature in X, and then scales the features such that they have zero mean and unit variance
* apply PCA to reduce the dimensionality to 3 components.
* split the data into training and testing sets using the train\_test\_split() function.
* Perform K-means clustering on the training set and test set and predict the cluster for each training data point.
* Finally, evaluate the performance of the clustering on the training & training set using classification\_report(), confusion\_matrix(), accuracy\_score(), and silhouette\_score() functions from sklearn.metrics.

2. Use pd\_speech\_features.csv

a. Perform Scaling

Graphical user interface, table

Description automatically generated with medium confidence

Description:

* StandardScaler ensures that all the features are on the same scale and prevents features with larger magnitude from dominating the distance calculations
* Applies the fit\_transform() method of the StandardScaler instance to the feature matrix X to perform feature scaling

2.b Apply PCA (k=3)

Graphical user interface, application

Description automatically generated

Description:

* PCA() performs principal component analysis on dataset x, reducing the dimensionality of the data from the original number of features to 3 principal components.
* fit\_transform() method of the PCA object is called on the data x to obtain a transformed version of the data, where each observation is represented by its three principal components.
* principalDf represents the transformed data x\_pca and three principal components
* finalDf concatenating principalDf with the last column of the original DataFrame df using pd.concat(). This is likely done to include the target variable (the variable being predicted) with the transformed data

2.c Use SVM to report performance

Graphical user interface, text, application, email

Description automatically generated

Description:

* trains an SVM classifier on the training set, predicts the classes for the test set using the trained classifier, and evaluates the performance using a classification report, confusion matrix, accuracy score, and silhouette score.

3.Apply Linear Discriminant Analysis (LDA) on Iris.csv dataset to reduce dimensionality of data to k=2.

Graphical user interface, application

Description automatically generated

Description:

* fit and transform the scaler object on our training data and only transform our test data.
* LabelEncoder to encode our target variable y into numerical values.
* LDA perform dimensionality reduction on our input features x. Here, we are reducing the number of input features to 2
* transform our training and test data using the fit\_transform and transform methods of the LDA object respectively.

4. Briefly identify the difference between PCA and LDA

In machine learning PCA and LDA are popular for dimensionality reduction. However, they have purpose varies.

Purpose: PCA is used for unsupervised learning and finds the directions of maximum variance in a dataset. It reduces the number of features by transforming the original dataset into a new coordinate system, where the features are uncorrelated and sorted by their variance. PCA is commonly used for data compression, visualization, and noise reduction. LDA, on the other hand, is used for supervised learning and aims to find the linear combinations of features that best separate the classes. It reduces the number of features by projecting the original dataset onto a lower-dimensional space while maximizing the class separability. LDA is commonly used for feature extraction, pattern recognition, and classification.

Method: PCA operates by finding the eigenvectors and eigenvalues of the covariance matrix of the data. The eigenvectors represent the directions of maximum variance, and the eigenvalues represent the amount of variance explained by each eigenvector. PCA selects the top k eigenvectors, where k is the desired dimensionality of the reduced dataset. the between-class scatter matrix and the within-class scatter matrix. LDA selects the top k eigenvectors that correspond to the largest eigenvalues.