Pattern Recognition Project

By:

Deena Fathi Mohamed 20201447217

Amgad Mohamed Ahmed 20201446888

George Ayman Botros 20201446826

The required imported libraries:

- import numpy as np -> used for working with arrays
- import pandas as pd -> used for importing multiple file
- from os import listdir -> used to get the list of all files and directories in the specified directory
- from numpy.linalg import eigh -> used for calculating eigen values and vectors of complex data
- import matplotlib.pyplot as plt -> used for data visualization and graphical plotting
- from PIL import Image as PImage -> used for opening, rotating, and displaying an image
- from sklearn.metrics import accuracy_score -> used to calculate the accuracy of either the faction or count of correct prediction
- from sklearn.neighbors import KNeighborsClassifier -> used for implementing learning based on the k nearest neighbors.

Initializing of Data:

We read the folders in the big folder then reading the images and appending them in an array and appending the data matrix and labels according to the read images.

Splitting:

From the Data Matrix we appended the odd rows for training and the even rows for testing. Therefore, there will be 5 instance per person in each, making the data split evenly 50% - 50%.

PCA:

PCA is a method that measures how each variable is associated with one another using a Covariance matrix

It's a technique for feature extraction.

STEPS OF PCA:

1) Centralization (Normalization) of data:

The aim is to Centralization the range of the continuous initial variables so that each one of them contributes equally to the analysis, to overcome the curse of dimensionality problem.

We do it before PCA because if there are large differences between the initial variables, the var with large range will dominate over those with small ranges (For example, a variable that ranges between 0 and 100 will dominate over a variable that ranges between 0 and 1), which will lead to biased results.

So, transforming the data to comparable scales can prevent this problem.

2) Covariance Matrix:

The aim of this step is measure of how much each of the dimensions vary from the mean with respect to each other.

A positive covariance indicates that features increase and decrease together, while a negative covariance indicates that the two features vary in the opposite directions.

3) Compute Eigen Values and Eigen Vectors:

The Eigenvector is the direction of line of best fit (direction of maximum variance in the dataset) drawn between points in a scattered plot, while the Eigenvalue is a number that tells us how the data set is spread out on the line which is an Eigenvector.

They need to be computed in descending order to determine the PCA of the data, also the main part of PCA.

4) Feature Vector:

In this step, what we do is, to choose whether to keep all these components or discard those of lesser significance (of low eigenvalues), and form with the remaining ones a matrix of vectors that we call Feature vector.

So, the feature vector is simply a matrix that has as columns the eigenvectors of the components that we decide to keep, it's first step of dimensionality reduction

5) Recasting data along PCA Axes:

In this step, which is the last one, the aim is to use the feature vector formed using the eigenvectors of the covariance matrix, to reorient the data from the original axes to the ones represented by the principal components.

Classifier Tuning using KNN:

It’s for prediction modeling. Tuning is the process of maximizing a model's performance without overfitting or creating too high of a variance.

To test the accuracy after applying PCA on the data.