Project Report

Given training and testing datasets consisting of images belonging to two classes digit “0” and digit “1”, this project requires to apply PCA on the datasets, implement Bayesian decision theory and report the accuracy of training and testing. Before implementation of the following tasks, the datasets are converted to 784-dimensional vector.

Tasks Implemented:

1. **Feature Normalization:** The data samples must be normalized to ensure that the mean and standard deviation of each of the 784 features is 0 and 1 respectively. Using function ***normalize(df)***, both training and testing datasets are normalized. The normalized feature y_i is calculates as:

   $$y_i = \frac{(x_i - mean_i)}{standard\ deviation_i}$$

2. **PCA using the training samples:** The covariance matrix is calculated by multiplying the feature matrix with its transpose. The eigen vectors and eigen values are then calculated using np.linalg.eig function. The first principal component is the first column of the resulting matrix and the second column is the second principal component.

   Eigenvector: (2D array of size 784x784)

   [[6.71636171e-04 2.18256179e-03 5.33374668e-04 ... 3.80323705e-04
   -4.39842096e-04 8.56599803e-05]
   [5.47816983e-04 -1.50132206e-03 -3.88916647e-03 ... -6.83721472e-04
   1.08181840e-03 6.36200779e-04]
   [6.55489835e-04 9.36205584e-04 -4.37032530e-04 ... 6.41272463e-04
   -2.04554669e-03 2.94319008e-04]
   ...]

   Eigenvalues: (array of size 784)

   [9.94608219e+01 3.98030418e+01 2.62106449e+01 2.33640930e+01
   1.61768688e+01 1.23765458e+01 1.05256521e+01 9.28483830e+00……..]

   The explained variance on each feature is also calculated to understand how much of variance in the training data is explained by each of the principal components.

   Explained variance: (array of size 784)

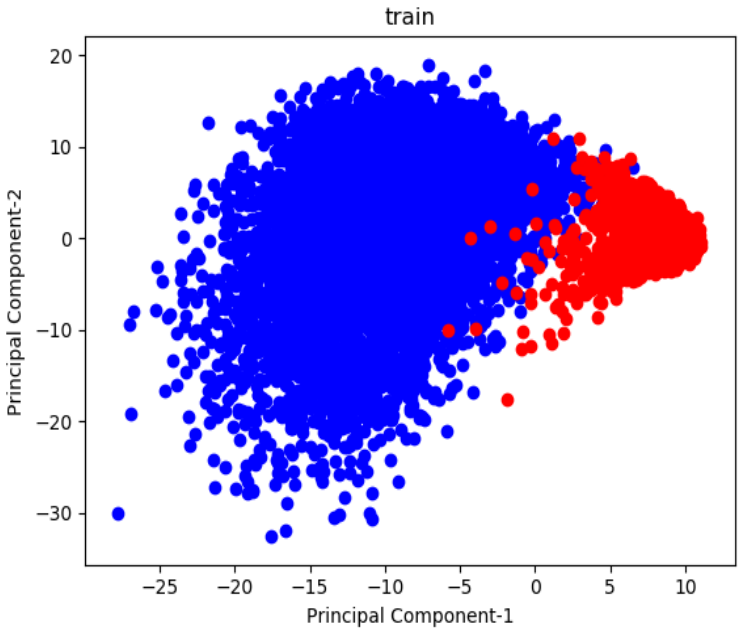
   [12.949114788656136, 4.736561464538868, 3.5096162148983145 ……..]

   The above result shows that the first principal component explains 12.949 % of variance in data and second principal component explains 4.7365 % of variance in data.

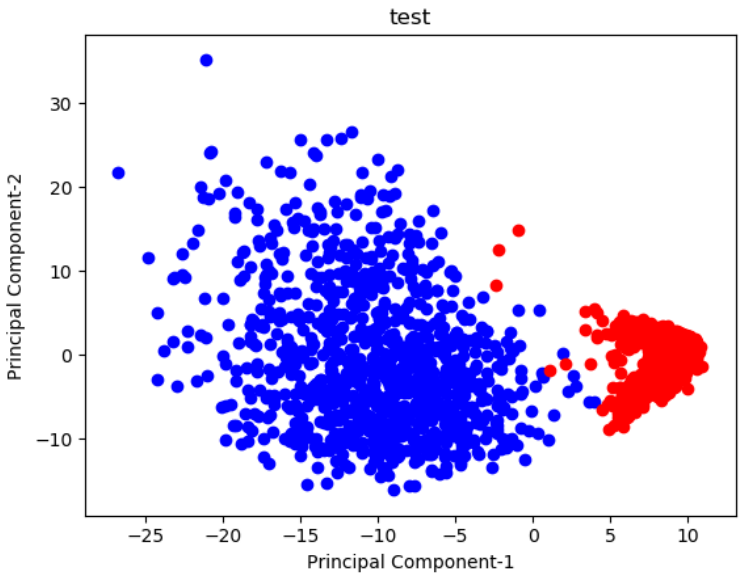
3. **Dimension reduction using PCA:**

   The 2-d projections of the samples on the first and second principal components are plotted. The samples corresponding to digit 0 are labeled blue and digit 1 are labeled red.

Plot for training data samples:



Plot for testing data samples:



4. **Density estimation:** The parameters of the Gaussian distribution are mean and covariance. In this step, the mean and covariance for each class is calculated for training data samples obtained from PCA. As there are two features now, there is a bivariate normal distribution for each digit.
   Below are the Mean and Covariance matrices for each class:

Parameters of Digit 0:

Mean vector:

[[-9.9234539]
[0.85142349]]

Covariance Matrix:

[[25.31833079494246, 15.8988892]
[15.8988892, 79.09351714668102]]

Parameters of Digit 1:

Mean vector:

[[8.71797945]
[-0.74799486]]

Covariance Matrix:

[[2.066297642226159, -0.02148369]
[-0.02148369, 4.083207203282685]]

5. **Bayesian Decision Theory for optimal classification:** In this step, the minimum error rate is calculated using Bayesian decision boundary rule and the training and testing accuracies are reported.
   training data accuracy = 98.78405053296487%
   test data accuracy = 99.29078014184397%