PCA, Density Estimation, and Bayesian Classification

Kanishk Sharma

November 1, 2023

1 Introduction

This project's goal is to use Principal Component Analysis and Bayesian Decision Theory to classify handwritten digits from images that are a subset of the MNIST dataset. The original MNIST dataset contains 70,000 images of handwritten digits, divided into 60,000 training images and 10,000 testing images. In this project we only use images for digits 5 and 6 .The process that is used is quite simple. First, we extract and vectorize the data that is contained in the.mat files. After normalizing the features, we determine the first two principal components. Projecting the normalized features onto the principal components and plotting the scatter plots for the training and testing samples are the next stages. Next, we perform density estimation on the projected samples. Lastly, we use Bayesian Decision Theory to perform classification on the given dataset.

2 Methodology

2.1 Data Extraction and Vectorization

The images are extracted from the provided .mat files in this stage using the loadmat function from the scipy.io library. Then each image is vectorized in order to display it as a 784-d vector because they are all stored as 28x28 arrays. In order to vectorize the data, reshape function was used.

2.2 Feature Normalization

After we have the features of images as 784-d vectors, those features are normalized. First, the training samples for digits 5 and 6 is concatenated using concatenate function from the numpy library and the mean and standard deviation for the combined training data is calculated. After computing the mean and standard deviation, the features are normalized using the following formula:

$$\boxed{y_i = \frac{x_i - m_i}{s_i}} \tag{1}$$

where, x_i represents a specific feature in the 784-dimensional space, s_i and m_i stands for the standard deviation and the mean for any feature x_i respectively, and y_i stands for

the normalized feature. In this step, a slight adjustment had to be made to the standard deviation values. In the case when standard deviation is zero, it is replaced by 1e-5 which is a minimal value. This replacement is made to avoid division by zero while normalizing any feature x_i

2.3 PCA using the training samples

At this step, Principal Component Analysis (PCA) is performed on the normalized training samples. To execute PCA, the normalized training samples for digits 5 and 6 are concatenated. Then, the covariance matrix for the concatenated normalized training samples is computed. After evaluating the covariance matrix using the cov function from the numpy library, the eigenvalues and eigenvectors are calculated using eigenanalysis. The eigenanalysis was performed using linalg.eig function from the numpy library.The eigenvalues and corresponding eigenvectors were then sorted in descending order. Since only 2 principal components are required for the project, the first two eigenvectors are selected which represent the 2 principal components. Then finally the combined normalized training data is projected onto the new feature space.

2.4 Dimension reduction using PCA

Since it was challenging to apply Bayesian Decision Theory in the 784 dimensional space, dimensionality reduction was performed using Principal Component Analysis. The normalized features were then projected on the 2 dimensional feature space using the dot function from the numpy library with the two principal components as the axes. Finally, to visualize the task, scatter plots were created.

2.5 Density Estimation

Now that the features were in 2 dimensional space, Density estimation was performed.To perform density estimation, initially mean and covariance matrix was evaluated for the projected samples of digits 5 and 6. Then using the multivariate_normal function from the scipy.stats library, a two dimensional multivariate normal distribution for digits 5 and 6 was generated. After evaluating the normal distributions, the probability density at each point in the feature space was calculated using the .pdf function. Then, finally to visualize the density estimation, contour plot was generated.

2.6 Bayesian Decision Theory for optimal classification

The final task that was performed was Bayesian decision theory to classify the samples. To accomplish this a function named bayesian_decision_theory was written. The function initially evaluated the likelihood for each digit, in our case digits 5 and 6. After that, the posterior probability was evaluated using the prior probabilities that were given and the likelihood that was evaluated earlier and using the formula

$$\boxed{\text{Posterior Probability} = \text{Likelihood} \times \text{Prior Probability}} \tag{2}$$

Finally, the function declared custom rules for classification which were in accordance to bayesian decision theory for optimal classification. Now, this function was later called to

classify the samples using the multivariate normal distributions evaluated earlier and the number of correct classifications were evaluated. Then to calculate accuracy, first the total number of classifications were calculated. Finally, to compute the accuracy of the process for classifying the training and testing samples, the number of correct classifications were divided by the total number of classifications.

3 Results and Observations

After extracting the data from the provided matlab files, the shape of the training and the testing data was observed to be:

Shape of the training data of 5	(5421, 28, 28)
Shape of training data for digit 6	(5918, 28, 28)
Shape of testing data for digit 5	(892, 28, 28)
Shape of testing data for digit 6	(958, 28, 28)

Table 1: Shapes of the training and testing data

After the vectorization of data, shape of the vectorized training and testing data was:
After the previous step, the features of images which were stored in a 28x28 array were

Shape of vectorized training data for digit 5	(5421, 784)
Shape of vectorized training data for digit 6	(5918, 784)
Shape of vectorized testing data for digit 5	(892, 784)
Shape of vectorized testing data for digit 6	(958, 784)

Table 2: Shapes of the vectorized data

vectorized to be 784 dimensional vectors which was ideal for the project. Then in task 1 these features were normalized using the formula described in section 2.2. The shape of the normalized features was observed to be:

Shape of normalized training sample for digit 5	(5421, 784)
Shape of normalized training sample for digit 6	(5918, 784)
Shape of normalized testing sample for digit 5	(892, 784)
Shape of normalized testing sample for digit 6	(958, 784)

Table 3: Shapes of the normalized features

After normalizing the features, task 2 and task 3 were executed. I concatenated normalized features of digits 5 and 6 and calculated the covariance matrix for the combined normalized training sample. After eigenanalysis and sorting the eigenvalues the first two eigenvalues and the corresponding eigenvectors were selected. The eigenvalues selected were:

$$\begin{bmatrix} 51.41085966 & 42.16025548 \end{bmatrix}$$

.The features were then projected onto the new two dimensional feature space, and the shape of the projected data was:

$$(\ 11339 \quad 2 \ )$$

Now, if we observe the shape of the projected data it has become two dimensional.So it can be understood that the vectors which were in 784 dimensional space have been represented in the two dimensional space .After achieving the dimension reduction, we plot the PCA visualization scatter plots for the samples, with the principal components as the axes.This was a very essential task in the project as can be seen that if the features are represented in two dimensional feature space, the calculations related to density estimation and bayesian decision theory become fairly less complex .The PCA visualizations are shown in figure 1. As is evident from Figure 1 each class looks like a normal distribution because the values become sparsely populated as we move away from the mean, which is the case with a normal distribution.

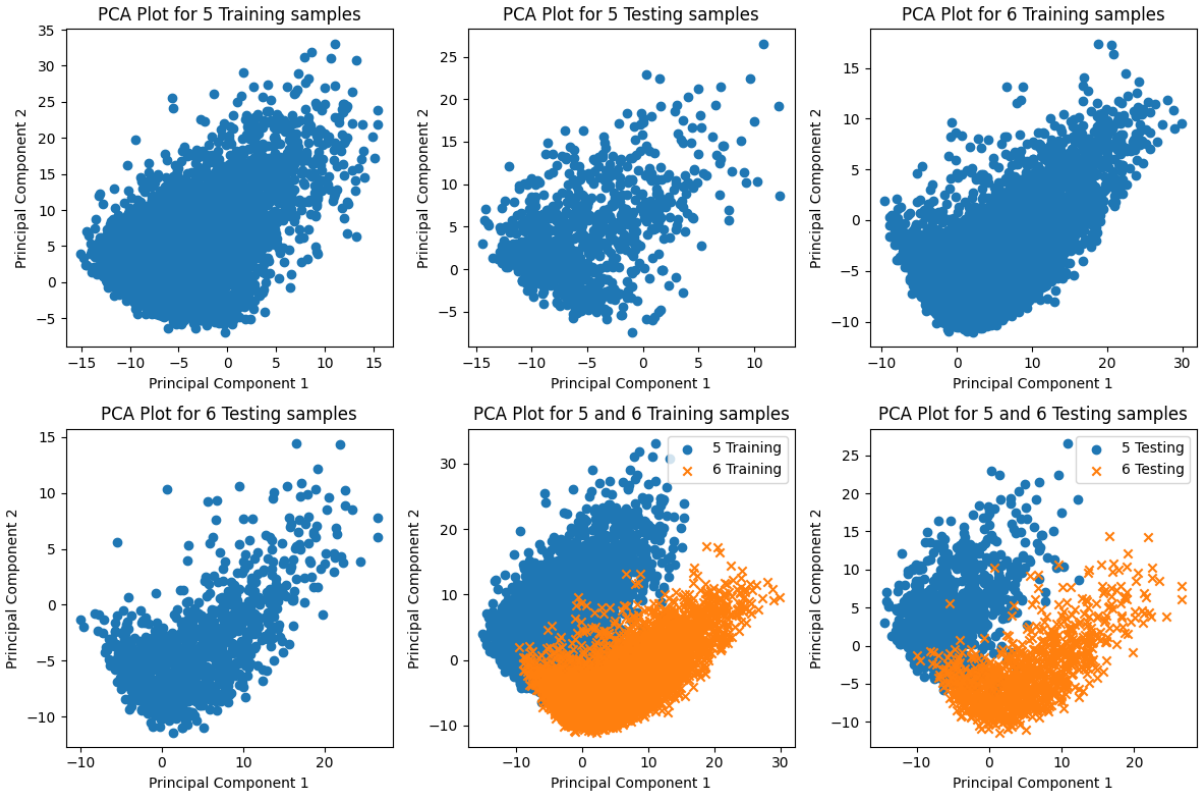

Figure 1: PCA Scatter Plots

After plotting the scatter plots, task 4 (Density Estimation) was carried out. For density estimation, first the parameters for the 2-d normal distribution such as the mean and covariance matrix were evaluated. The results for the evaluations are shown in Table 4.

Mean for projected training samples of digit 5	(-4.45320748, 4.06951377)
Mean for projected training samples of digit	(4.07922233 , 3.72775171)

Table 4: Mean of projected training samples

Covariance matrix for projected training samples of digit 5 was observed to be:

$$\begin{bmatrix} 23.39792743 & 15.13683929 \\ 15.13683929 & 36.44222332 \end{bmatrix}$$

Covariance matrix for projected training samples of digit 6 was observed to be:

$$\begin{bmatrix} 42.26796632 & 17.9467385 \\ 17.9467385 & 18.33394357 \end{bmatrix}$$

Using these values multivariate normal distributions were calculated and plotted with the features as the axes. The 2-D Normal distribution for digits 5 and 6 is shown in figure 2.

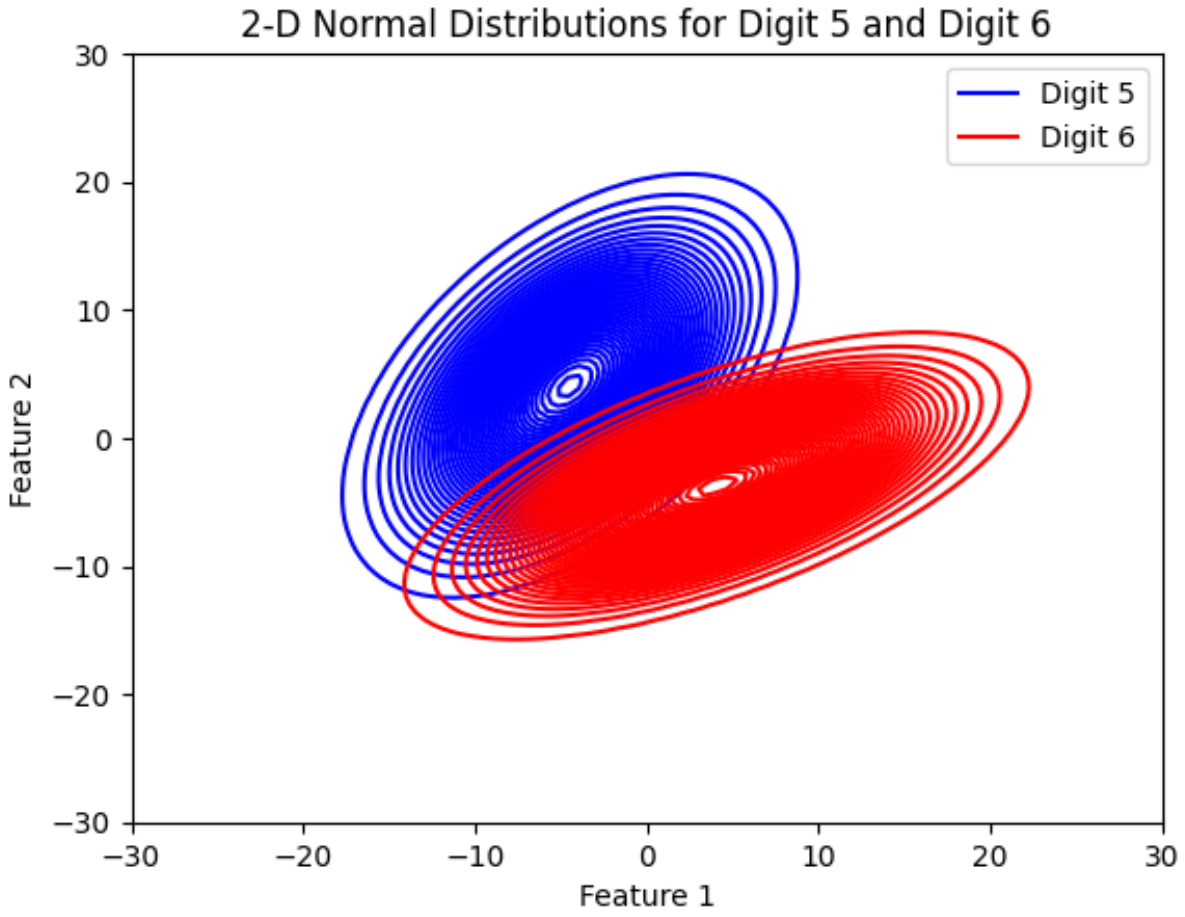

Figure 2: 2-D Normal distributions for digits 5 and 6.

Finally, in the last task bayesian decision theory was implemented and classification was executed. Also, the accuracy was calculated for the classification of testing and training samples. The results are shown in Table 5.

Accuracy for the classification of training samples	94.28 percent
Accuracy for the classification of testing samples	93.95 percent

Table 5: Accuracy of the classification of testing and training samples.

4 Conclusion

To sum up, this project took a thorough approach to image classification, concentrating on differentiating between the numbers five and six. By applying Bayesian Decision Theory and meticulous feature pre-processing, the classifier was able to attain remarkable classification accuracy for both training and testing datasets. The practical value of strong pre-processing methods,dimensionality reduction techniques such as PCA ,the strength of probabilistic modeling, and the application of Bayesian Decision Theory to challenging picture classification problems are all highlighted by this project. It draws attention to the connections that exist between statistics, computer science, and mathematics. The project also provides insights into automated decision-making for a variety of real-world applications.