Project Part 1 [10 points] Density estimation and classification
(Due July 28)

In this part, you need to first perform parameter estimation from a given dataset (which is a subset from the MNIST dataset). The MNIST dataset contains 70,000 images of handwritten digits, divided into 60,000 training images and 10,000 testing images. We use only images for digit “0” and digit “1” in this question.

Therefore, we have the following statistics for the given dataset:

Number of samples in the training set: "0": 5923 ;"1": 6742.

Number of samples in the testing set : "0": 980; "1": 1135

You are required to extract the following two features for each image:

(1) the average brightness of the image (simply averaging all the pixel values); and

(2) the average of the variances for each of the rows of the image (compute the variance for the pixels in each row, and then average all the variances from all the rows).

We assume that these two features are independent, and that each image (represented by a 2-D features vector) is drawn from a 2-D normal distribution.

We also further assume that the prior probabilities are the same (P(Y=0) = P(Y=1) =0.5), although you may have noticed that these two digits have different numbers of samples in both the training and the test sets.

You may go to the original MNIST dataset (available here http://yann.lecun.com/exdb/mnist/) to extract the images for digit 0 and digit 1, to form the dataset for this project. To ease your effort, we have also extract the necessary images, and store them in “.mat” files. You may use the following piece of code to read the dataset in Python (or you may use Matlab, since these are .mat files):

```
import scipy.io
Numpyfile= scipy.io.loadmat('matlabfile.mat')
```

The key algorithmic tasks in this part of the project include:

(1) Extracting the features and then estimating the parameters for the 2-D normal distribution for each class/digit, using the training data. Note: You will have two distributions, one for each digit; Also, as mentioned above, we *assume that the features are independent*.

(2) Use the estimated distributions for doing minimum-error classification.

Detailed Tasks (required):

1. Write code to extract features for both training set and testing set.
2. Write code to estimate/compute the parameters of the relevant distributions, using only the training set.
3. Write code to implement the classifier and use it produce a predicted label for each testing sample; Use the same classifier to classify the training samples as well.
4. Write code to compute the classification accuracy (this should be done separately for the training set and the test set respectively).
5. Submit a short report summarizing the results, including the estimated parameters of the distributions and the final classification accuracy.

Detailed Tasks (optional):

1. Repeat the experiments without assuming independence of the features.
2. Consider doing multi-class classification (e.g., considering three or four classes), and visualize the samples in the 2-D feature space to get a sense of if they samples are well separated.

Optional tasks are to be explored on your own if you are interested and have extra time for them. No submission is required on the optional tasks. No grading will be done even if you submit any work on the optional tasks. No credit will be assigned to them even if you submit them. (So, please do not submit any work on optional tasks.)

Evaluation criteria: Working code; Correct final results (for both estimated parameters and the classification results).