ISYE 6740 Homework 4

Total 100 points.

1. **Implementing EM algorithm for MNIST dataset. (50 points)**

   Implement the EM algorithm for fitting a Gaussian mixture model for the MNIST dataset. We reduce the dataset to be only two cases, of digits "2" and "6" only. Thus, you will fit GMM with $C = 2$. Use the data file data.mat or data.dat on Canvas. True label of the data are also provided in label.mat and label.dat

   The matrix images is of size 784-by-1990, i.e., there are totally 1990 images, and each column of the matrix corresponds to one image of size 28-by-28 pixels (the image is vectorized; the original image can be recovered by map the vector into a matrix.)

   Hint: You may find the notes speed-up-GMM.pdf useful, which explains how to evaluate the density of a multi-variate normal distribution. In this homework question, it is recommended you implement the evaluation of the Gaussian density this way, to avoid numerical issues.

   (a) (5 points) Select from data one raw image of "2" and "6" and visualize them, respectively.

   (b) (15 points) Use random Gaussian vector with zero mean as random initial means, and two identity matrices I as initial covariance matrices for the clusters. Plot the log-likelihood function versus the number of iterations to show your algorithm is converging.

   (c) (15 points points) Report, the fitting GMM model when EM has terminated in your algorithms, including the weights for each component and the mean vectors (please reformat the vectors into 28-by-28 images and show these images in your submission). Ideally, you should be able to see these means corresponds to "average" images. No need to report the covariance matrices.

   (d) (15 points) Use the p_{ic} to infer the labels of the images, and compare with the true labels. Report the miss classification rate for digits "2" and "6" respectively. Perform K-means clustering with $K = 2$ (you may call a package or use the code from your previous homework). Find out the miss classification rate for digits "2" and "6" respectively, and compare with GMM. Which one achieves the better performance?

2. **Basic optimization. (50 points.)**

The background of logistic regression will be discussed in the next lecture. Here, we just focus on finding out the property of the optimization problem, related to training a logistic regression.

Consider a simplified logistic regression problem. Given m training samples (x_i, y_i), $i = 1, \ldots, n$. The data $x_i \in \mathbb{R}$, and $y_i \in \{0, 1\}$. To train/fit a logistic regression model for classification, we solve the following optimization problem, where $\theta \in \mathbb{R}$ is a parameter we aim to find:

$$\max_{\theta} \ell(\theta), \tag{1}$$

where the log-likelihood function

$$\ell(\theta) = \sum_{i=1}^{n} \{-\log(1 + \exp\{-\theta x_i\}) + (y_i - 1)\theta x_i\}.$$

(a) (15 points) Derive the gradient of the cost function $\ell(\theta)$ in (1) and write a **pseudo-code** for performing **gradient descent** to find the optimizer θ^*. This is essentially what the training procedure does. pseudo-code means you will write down the procedure in steps for the algorithm, not necessarily any specific programming language.

(b) (15 points) Write down a **stochastic gradient descent** algorithm to solve the training of logistic regression problem (1).

(c) (20 points) **Show that the training problem in basic logistic regression problem is concave.** Derive the Hessian matrix of $\ell(\theta)$ and based on this, show the training problem (1) is concave. Explain why the problem can be solved efficiently and gradient descent will achieve a unique global optimizer, as we discussed in the lecture.