Due date: **Dec. 4, 2020, 11:59 PM** (Arlington time). You have **two** late days — use it at as you wish. Once you run out of this quota, the penalty for late submission will be applied. You can either use your late days quota (or let the penalty be applied). **Clearly indicate** in your submission if you seek to use the quota.

What to turn in:

1. Your submission should include your complete code base in an archive file (`zip, tar.gz`) and `q1/, q2/,` and so on), and a very very clear README describing how to run it.
2. A brief report (typed up, submit as a PDF file, NO handwritten scanned copies) describing what you solved and implemented and known failure cases. The report is **important** since we will be evaluating the grades mostly based on the report.
3. Submit your entire code and report to Canvas.

Notes from instructor:

- Start early!
- You may ask the TA or instructor for suggestions, and discuss the problem with others (minimally). But **all parts of the submitted code must be your own**.
- Use Matlab or Python for your implementation.
- Make sure that the TA can easily run the code by plugging in our test data.

Problem 1

(Non-parameteric density estimation **40pts**)

1. **(20pts)** Write a function `[p, x] = mykde(X,h)` that performs kernel density estimation on data X with bandwidth h. It should return the estimated density $p(x)$ and its domain x where you estimated the $p(x)$ for X in 1-D and 2-D.
   - Generate $N = 1000$ Gaussian random data with $\mu_1 = 5$ and $\sigma_1 = 1$. Test your function `mykde` on this data with $h = \{0.1, 1, 5, 10\}$. In your report, report the histogram of X along with the figures of estimated densities.
   - Generate $N = 1000$ 1-D Gaussian random data with $\mu_1 = 5$ and $\sigma_1 = 1$ and another Gaussian random data with $\mu_2 = 0$ and $\sigma_2 = 0.2$. Test your function `mykde` on this data with $h = \{0.1, 1, 5, 10\}$. In your report, report the histogram of X along with the figures of estimated densities.

2. **(20pts)** Generate 2 sets of 2-D Gaussian random data with $N_1 = 500$ and $N_2 = 500$ using the following parameters:

$$\mu_1 = [1, 0],\ \mu_2 = [0, 2.5],\ \Sigma_1 = \begin{bmatrix} 0.9 & 0.4 \\ 0.4 & 0.9 \end{bmatrix},\ \Sigma_2 = \begin{bmatrix} 0.9 & 0.4 \\ 0.4 & 0.9 \end{bmatrix}. \tag{1}$$

   Test your function `mykde` on this data with $h = \{0.1, 1, 5, 10\}$. In your report, report figures of estimated densities (either 3D or 2D plot with color representing the density).

Problem 2

(Dimension Reduction **60pts**)

1. **(10pts)** Write your own function `[PC] = myPCA(X, k)` that returns k principle components (PCs) on data X.

2. **(10pts)** Run your PCA on MNIST dataset (you used this data for HW2). Taking too long time? You can use the tricks explained in the lecture notes.

3. **(10pts)** Visualize the data using the first 2 PC, i.e., plot each sample as scatter plot by projecting them onto a 2D spaces whose each axis is the PC. Do they look good or not so good? Explain.

4. **(10pts)** Visualize 10 PC as images. That is, reshape each PC as a 28x28 image and show it as a 2D image. Any interesting shapes?

5. **(10pts)** Let's try out classification together with the PCA. For this question, you may use built-in functions or libraries. Use Logistic Regression with 10 output nodes with softmax function as the classifier. First, train a classifier using the raw images for 10 epochs and measure the training time. Then, train a classifier for 10 epochs using transformed data with 30 PCs, i.e., project each image to a 30-D space using the 30 PCs and train the classifier, and measure the training time. How much improvement in the training time? How are their performances on the testing set? Explain your thoughts.

6. **(10pts)** What do Neural Networks learn? Let's take a look at the pre-trained Logistic Regression model using the raw data. Each output node has 784 weights associated with it. Reshape the weights to an 28x28 image; do this for the all output nodes and visualize each of them. Any interesting shapes? What do they represent? How do they differ from the features learned from PCA (from Q4)? Explain your thoughts.