Homework 3

Due on Monday, **6/4/2018**, at **9:00 AM** through E-campus platform.

Be noted that late homework will **NOT** be accepted!!

Grading and Submission Policy

- Please add comments to explain your codes by the end of each meaningful line.
- You can use any programming language to finish this computer assignment.
- Describe how to run your codes in a text file named by YourStudentID_Readme.txt. Make sure that your submitted codes can be readily compiled and produce meaningful results, such as the decision region elaborated in the problem descriptions below. If we cannot run your codes, we cannot grade. This leads to point deduction.
- Write a report to detail your discussions and convert your report to .pdf format. You don't need to turn in the hardcopy of the report.
- Please upload all the files (including readme, codes and discussion report) **without** needing to encapsulate them into a single compressed file. Name all the uploaded files according to the following rules:
  1. Use the following format: YourStudentID_Readme.txt, YourStudentID_Report.pdf, YourStudentID_hw3_code1.xxx, YourStudentID_hw3_code2.xxx, etc.
  2. No Chinese characters in the file name
  3. No space in the file name
- Discussions are encouraged. But plagiarism is strictly prohibited. You need to program your own codes, every line of them. **Again, you will fail the course as penalty for plagiarism!!**

Part I -- Reading Assignment

1. Read Chapter 5, Chapter 6, and Chapter 7. Furthermore, please read Sec. 12.1 and learn Principle Component Analysis (PCA) by yourself.

   Note that PCA will be used in the problem below.

2. Le Cun's famous paper on convolutional neural network (CNN)

   Yann LeCun, "Gradient-Based Learning Applied to Document Recognition," *Proceedings of the IEEE*, 1998.

3. A very good tutorial website for neural network:

   `https://brohrer.mcknote.com/zh-Hant/how_machine_learning_works/how_neural_networks_work.html`

Part II -- Computer Assignment

In this problem, you need to build a neural network in order to classify images into three classes. Particularly, instead of calling on any existing functions in MATLAB or Python that do the tricks for you, you need to code on the error backpropagation algorithm by yourself in order to realize the learning process.

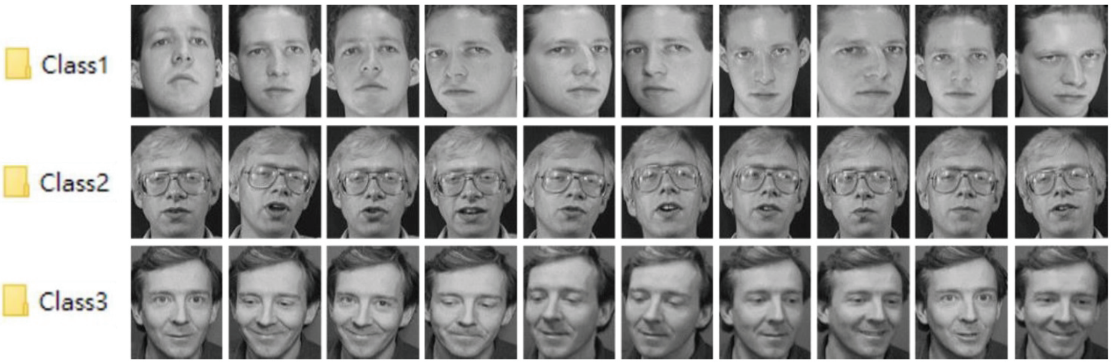

Figure 1: Illustration of the data set containing images of three people. Your task is to establish and train a neural network model capable of performing face recognition.

Data

Data, encapsulated in Data.zip, in this problem are the images of three people as shown in Fig. 1, which are respectively deposited in the three folders Class1, Class2, and Class3. Each class has 1000 30 × 30 images. You need to partition this data set by yourself as your training data and testing data.

Problem Description (100 points)

(a) Please build a two-layer (as in Fig. 2) neural network to perform the face recognition to the images in the given data set, i.e. this is a classification problem with three classes. More detailed steps are as follows:

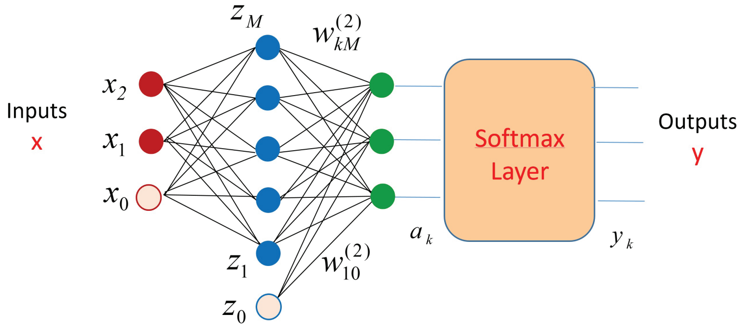

Figure 2: Two layer neural network with three inputs nodes and three output nodes.

⟶ The very first step is to perform dimensionality reduction using principal component analysis (PCA). You need to read Sec. 12.1, as mentioned in the reading assignment, for understanding what PCA is all about.

Then, please employ PCA to map the images in the data down to 2 dimensions, i.e. extract the coefficients of the two principle components for the given data set. You don't need to write PCA by yourself; instead, you can use any functions, in MATLAB or Python, about PCA.

- Build a two-layer neural network. The number of hidden units has to be decided by you. Please adopt sigmoid function as the nonlinear mapping in the hidden neurons, and train the weights using stochastic gradient descent.

  You need to implement the backpropagation algorithm in your code to evaluate the gradient.

  Note that the number of input nodes in the neural network here is three, corresponding to the two principal components and the bias.

- Plot decision regions and discuss with different settings designed by yourself.

(b) In the 2nd part, please build a three-layer neural network to perform the face recognition to the images in the given data set. Discuss the performance difference from that in part (a) by adding one more layer here.

(c) Following part (b), please adopt the rectified linear unit (ReLU) as the nonlinear activation function in the hidden units, where the ReLU is defined as

$$h(a) = \max(0, a).$$

The derivative of ReLU is given by

$$h'(a) = \begin{cases} 1 & \text{if } a > 0, \\ 0 & \text{otherwise,} \end{cases}$$

where we have set the derivative of ReLU at $a = 0$ to 0.
Please redo part (b) and compare the performance difference of the face recognition by neural network, when using ReLU as the nonlinear activation function, from that in part (b)