EE5434 Homework 4

Out: Nov. 13
Due: 11:59PM, ~~Dec. 9th (Monday)~~ Nov. 30th (Saturday), 2019, at Canvas.
Full mark: 100 pts
You are allowed to form groups of at most two members. If you choose to do it yourself, 10 bonus points will be added. However, the total points are still bounded by 100.

You can use APIs to implement ~~SVM and~~ ANN. There are two choices: Sklearn or Keras.

Submission format: your source codes, a readme file, and a report. All the files should be in pdf format. Zip all into one file. The submission site is still Canvas.
Naming rules: If you have two members in the group, such as Yaser (first name) Mostafa (family name) and Xuantian (first name) Chan (family name), name it as 2-MostafaYaser-ChanXuantian.zip. If you only have one member such as Bruce (first name) Lee (family name), name it as 1-LeeBruce.zip. Pay attention to the capitalized letters and the order. Based on the lessons I learned from last semester, adding bonus points for a large number of groups with either 1 or 2 members can be error-prone if you don't follow these rules. Thus, we must enforce this. 5 pts will be deducted if the naming rule is not followed.

In this homework, you will apply what we learned to a classic handwritten digit classification problem. The inputs are images containing handwritten digits from 0 to 9. The output of the classification should the correct label (0-9).

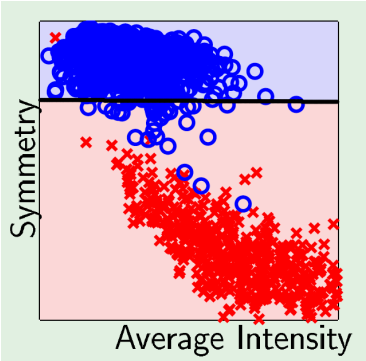
Data availability:
http://amlbook.com/support.html
Please scroll down to the bottom of the above page to download the data. You can get both the extracted features (symmetry and intensity) and also the raw features. When we use "two features", we refer to the extracted features. If we use "raw features", we refer to the 256 grayscale values.

This homework contains the following components starting from linear classification models to more flexible classification. The features are from extracted features to raw features. For each task, you should have two programs. One is the training program. The other is the test program. Name them as ProblemX-train.py and ProblemX-test.py. If a problem needs two different training/testing programs (such as with and without regularization), name them as ProblemX-1-train.py and ProblemX-2-train.py. For each program, add comments for the learning algorithm and testing algorithm.

Tasks:

1. (10 pts) **Plot** a scatter plot (see an example below this problem) of two features for 10 labels (i.e. 0-9), **analyze, and conclude** whether these two features might do a good job

in separating the 10 digits. Feature 1 is the intensity and feature 2 is the symmetry feature. The purpose of this basic data exploration technique is to get an insight about the effectiveness of these two features. If it is too crowded to use one plot, you should use multiple ones. Your conclusion should be supported by your plots and analysis. If not, you won't get full mark for this question.



1 and 5 means two labels. You can assume the inputs are only concerned with 1 or 5.

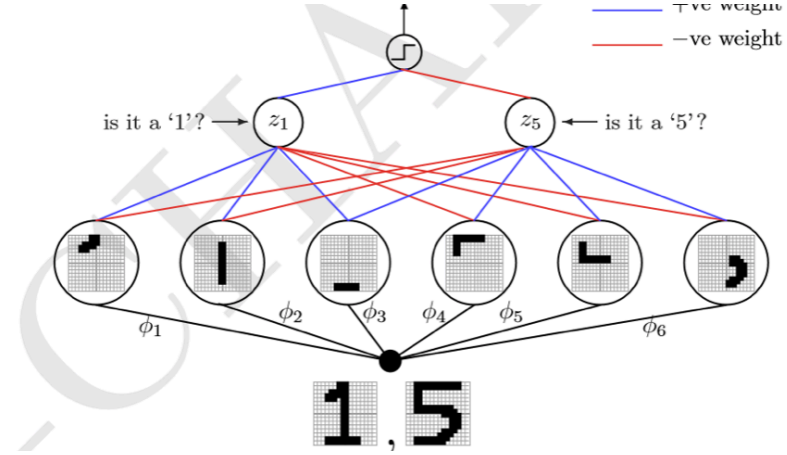
2. (15 pts) Apply neural network of 1 hidden layer to classify 1 and 5. The features are: symmetry and average intensity. Use 3-fold cross-validation. **Clearly describe and plot your network structure, such as the number of units in each layer**. Train/test at least 3 sets of different number of hidden units. **Compare their performance**: **in-sample error and test set error**.

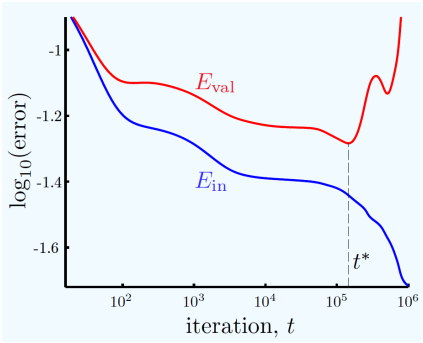
30 pts

3. 3.1 (~~20~~ pts) Apply two-layer neural network for classification of 1 and 5, using the raw features as input. A sample structure is provided below. The output layer has one unit with sign function. Apply 3-fold cross-validation. **Train and test** the following structures [256, 6, 2, 1], [256, 3, 2, 1]. **Report** the final in-sample error and test-set performance for each structure. For the structure [256, 6, 2, 1], **plot** the change of the in-sample error and test-set error for each iteration. A sample training dynamic figure is shown below (for format demo only)

This is the number of units in each layer. e.g. there are 256 units in the input layer

~~3.2 (10 pts) Then add regularization and repeat the above experiments. Describe your regularization methods clearly. Report all the results similar to the above model in Problem 4.~~





4. (40 pts) Apply neural network ~~and SVM~~ for classification for all 10 digits, using the raw features as input. Describe your methods. **Report and compare** their classification accuracy using 3-fold cross-validation. Report the accuracy for each fold. Report the variance of the accuracy for five folds. **Analyze** the causal of the difference.
   The points are given based on your efforts of improving the performance.

(5 pts) Based on the above experiments, **draw a conclusion** about what is the best method for handwritten digit classification. Note that this conclusion must be supported by your analysis and experimental results. Otherwise, you won't get full credit.

3 fold cross validation: suppose you have 99 data points, divide it into three parts of roughly equal size. Call them set1, set2, set3. Each has 33 data points. Now, train your model using two sets and test your trained model on the third set. For example, train on set 2 and set 3, test on set 1. You will have three different test sets and thus three accuracy values. You can report their variance.