ECSE-6610: PR Homework Set 4

Chengjiang Long

April 16, 2018

Assigned Date: April 17, 2018.
Due Date: April 27, 2018.
Collaboration Policy. Homeworks will be done individually: each student must hand in their own answers. Use of partial or entire solutions obtained from others or online is strictly prohibited.
Late Policy. No late submissions will be allowed without consent from the instructor. If urgent or unusual circumstances prohibit you from submitting a homework assignment in time, please e-mail me explaining the situation.
Submission Format. Electronic submission of a zip file is mandatory. Include code in your pdf file as needed to make your answers clear. Submit all code separately.

Problem 1 (30 points) AdaBoost is a powerful method combining 'base' classifiers so that the performance of the ensemble would be significantly better than any of the base classifiers. Consider the exponential error function

$$E = \sum_{n}^{N} \exp\{-t_n f_m(\mathbf{x}_n)\} \tag{1}$$

where $f_m(\mathbf{x}) = \frac{1}{2}\sum_{l=1}^{m} \alpha_l y_l(\mathbf{x})$.

In AdaBoost we are actually minimizing the exponential error with respect to both the base classifiers $y_1(\mathbf{x}), y_2(\mathbf{x}), \ldots, y_m(\mathbf{x})$ and the weighting coefficient $\alpha_1, \alpha_2, \ldots, \alpha_m$.

(a) **[10 points]** By treating the previous $m-1$ base classifier $y_1(\mathbf{x}), y_2(\mathbf{x}), \ldots, y_{m-1}(\mathbf{x})$ and their coefficient $\alpha_1, \alpha_2, \ldots, \alpha_{m-1}$ as fixed, show that the error function E in m-th round can be written as

$$E = \sum_{n}^{N} w_n^{(m)} \exp\{-\frac{1}{2} t_n \alpha_m y_m(\mathbf{x}_n)\}$$

(b) **[10 points]** Show that minimizing the error function E in Equation (1) with respect to base classifiers $y_m(\mathbf{x})$ is equivalent to minimzing the following error function

$$J_m = \sum_{n}^{N} w_n^{(m)} \mathbf{1}(y_m(\mathbf{x}_n) \neq t_n)$$

where $\mathbf{1}(\cdot)$ is an indicator function.

Hint: *separate the correctly and incorrectly classified points will make it much easier.*

(c) **[10 points]** Show that minimizing the error function E in Equation (1) with respect to α_m , we will get

$$\alpha_m = \ln\{\frac{1-\epsilon_m}{\epsilon_m}\}$$

where

$$\epsilon_m = \frac{\sum_n^N w_n^{(m)} \mathbf{1}(y_m(\mathbf{x}_n) \neq t_n)}{\sum_n^N w_n^{(m)}}$$

Problem 2 (20 points) HMM

(a) **[5 points]** Assume we have an HMM called M and a sequence of n observations called O that were generated from M. Does the sequence of observations O or $O + o_{n+1}$ (*i.e.*, the same sequence with one additional observation o_{n+1}) have a higher probability under M? If not enough information is given, explain what extra information is required.

(b) **[15 points]** Answer the following questions using the transition matrix T and emission probabilities E below. Below, $\odot$ and $\otimes$ are two output variables, A and B are two hidden states; s_n refers to the n-th hidden state in the sequence and on refers to the n-th observation.

Table 1: Transition matrix T

	A	B	END
START	0.5	0.5	0.0
A	0.2	0.3	0.5
B	0.4	0.4	0.2

Table 2: Emission matrix E

	$\odot$	$\otimes$
A	0.5	0.5
B	0.3	0.7

(1) Is $P(o_2 = \odot | s_1 = B) = P(o_2 = \odot | o_1 = \otimes)$?

(2) Is $P(s_2 = B | s_1 = A) = P(s_2 = B | s_1 = A, o_1 = \odot)$?

(3) Is $P(o_2 = \odot | s_1 = A) = P(o_2 = \otimes | s_1 = A, s_3 = A)$?

(4) Compute the probability of observing $\otimes$ as the first emission of a sequence generated by an HMM with transition matrix T and emission probabilities E.

(5) Compute the probability of the first state being A given that the last token in an observed sequence of length 2 was the token $\odot$.

Problem 3 (20 points) Consider a three-layer network for classification with output units employing softmax activation function, trained with 0-1 signals.

(a) **[10 points]** Derive the learning rule if the criterion function (per pattern) is sum squared error, *i.e.*,

$$J(w) = \frac{1}{2} \sum_k^c (y_k - t_k)^2$$

(b) **[10 points]** Repeat for the criterion function is cross-entropy, *i.e.*,

$$J_{ce}(w) = \sum_k^c t_k \ln \frac{t_k}{z_k}$$

Hint: *derive your solution based on back-propagation.*

Problem 4 (30 points + Extra 20 points) Download the dataset from my Google Drive:
train: https://drive.google.com/open?id=1QHpu5xfbKxHIWYVH7BqCFWNArQ5Fgxs7
test: https://drive.google.com/open?id=18Y4aLI2VIZ2eH6FQhSJqyej-8viTeCVZ

Note that this is a subset of the LeCun's MNIST dataset containing just the digits 0, 1, and 2. The full dataset is available at http://yann.lecun.com/exdb/mnist. The dataset is split into training and testing pictures. For convenience, I named each image as "img_[*image id number*]_lb_[*image label*].png".

(a) **[15 points]** Design and implement a 3-layer perceptron network with SGD. Plot the training error and testing error vs iterations.

(b) **[15 points]** Modify and implement the LeNet network with SGD. Plot the training error and testing error vs iterations.

(e) **[Extra 20 points]** Modify and implement the AlexNet network with SGD, as well as with RMSProp and with Adam optimizer. Plot the training error and testing error vs iterations, and discuss what observe.