CSE 455/555 Spring 2013 Homework 10: Boosting

Jason J. Corso
Computer Science and Engineering
SUNY at Buffalo
jcorso@buffalo.edu

This assignment does not need to be submitted and will not be graded, but students are advised to work through the problems to ensure they understand the material.

You are both allowed and encouraged to work in groups on this and other homework assignments in this class. These are challenging topics, and working together will both make them easier to decipher and help you ensure that you truly understand them.

Ada-Boost

Ada-Boost 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

$$E = \sum_{n=1}^{N} \exp\{-t_n f_m(x_n)\}$$

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

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

1. By treating the previous $m-1$ base classifier $y_1(x), \ldots, y_{m-1}(x)$ and their coefficient $\alpha_1, \ldots, \alpha_{m-1}$ as fixed, show that the error function E in m-th round can be written as

$$E = \sum_{n=1}^{N} w_n^{(m)} \exp\left\{-\frac{1}{2} t_n \alpha_m y_m(x_n)\right\} \tag{1}$$

2. Show that minimizing the error function E in equation 1 with respect to base classifiers $y_m(x)$ is equivalent to minimzing the following error function

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

   where $\mathbf{1}(\cdot)$ is an indicator function.
   Hint: *separate the correctly and incorrectly classified points will make it much easier.*

3. Show that minimizing the error function E in equation 1 with respect to α_m , we will get

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

   where

$$\epsilon_m = \frac{\sum_{n=1}^{N} w_n^{(m)} \mathbf{1}(y_m(x_n) \neq t_n)}{\sum_{n=1}^{N} w_n^{(m)}}$$

4. Implement the Ada-boost algorithm using any base classifier you like (*e.g.* perceptron, decision tree, *etc.*).