Problem Set 4*

Tasha Pais†

Problem 1

To derive the expression for α_t that makes the empirical error probability $\hat{\epsilon}_{D_t}(h_t) = \frac{1}{2}$, we follow these steps:

1. Express $\hat{\epsilon}_{D_t}(h_t)$ as a function of α_t and $\hat{\epsilon}_{D_{t-1}}(h_t)$:

$$\hat{\epsilon}_{D_t}(h_t) = \sum_{i=1}^{N} D_t(i) \mathbb{1}(h_t(x_i) \neq y_i)$$

   Substituting $D_t(i)$ from AdaBoost:

$$D_t(i) = \frac{D_{t-1}(i) e^{-\alpha_t y_i h_t(x_i)}}{\sum_{j=1}^{N} D_{t-1}(j) e^{-\alpha_t y_j h_t(x_j)}}$$

   Therefore,

$$\hat{\epsilon}_{D_t}(h_t) = \frac{\sum_{i=1}^{N} D_{t-1}(i) e^{-\alpha_t y_i h_t(x_i)} \mathbb{1}(h_t(x_i) \neq y_i)}{\sum_{j=1}^{N} D_{t-1}(j) e^{-\alpha_t y_j h_t(x_j)}}$$

2. Set $\hat{\epsilon}_{D_t}(h_t) = \frac{1}{2}$ and solve for α_t:

$$\frac{1}{2} = \frac{\sum_{i=1}^{N} D_{t-1}(i) e^{-\alpha_t y_i h_t(x_i)} \mathbb{1}(h_t(x_i) \neq y_i)}{\sum_{j=1}^{N} D_{t-1}(j) e^{-\alpha_t y_j h_t(x_j)}}$$

   The solution is:

$$\alpha_t = \frac{1}{2} \log \frac{1 - \hat{\epsilon}_{D_{t-1}}(h_t)}{\hat{\epsilon}_{D_{t-1}}(h_t)}$$

Problem 2

We are asked to consider whether it is possible for the AdaBoost algorithm to select the same classifier in consecutive rounds, i.e., whether $h_{t+1} = h_t$ for some t. To analyze this, we need to understand the mechanism of AdaBoost in updating the distribution D_t and choosing classifiers.

*Due: `November 29, 2023`, Student(s) worked with: `Collaborators`

†NetID: `tdp74`, Email: `tdp74@rutgers.edu`

Recall that AdaBoost updates the distribution D_t for each training example based on the performance of the classifier h_t. Specifically, the weights of the correctly classified examples are decreased, and the weights of the incorrectly classified examples are increased. This ensures that the subsequent classifier h_{t+1} focuses more on the examples that h_t classified incorrectly.

Given the assumption that for all data distributions D with full support, there is always some $h \in H$ such that $\hat{\epsilon}_D(h) < \frac{1}{2}$, but never $h' \in H$ such that $\hat{\epsilon}_D(h') = 0$, we can infer the following:

- Since no classifier can perfectly classify all examples (i.e., $\hat{\epsilon}_D(h') = 0$ is impossible), it implies that there is always room for improvement in the classification.
- The updated distribution D_{t+1} is more focused on examples that h_t failed to classify correctly. Therefore, unless h_t was equally good (or bad) at classifying all examples (which is highly unlikely given the assumption), the distribution D_{t+1} would present a different set of challenges compared to D_t.
- Hence, it is unlikely that the same classifier h_t will be the optimal choice for both D_t and D_{t+1}, as the distribution has been specifically updated to challenge the weaknesses of h_t.

In conclusion, while theoretically not impossible, it is highly improbable under normal circumstances and given the assumptions that $h_{t+1} = h_t$. This is because AdaBoost is designed to adaptively focus on the weaknesses of the previously chosen classifiers, thereby making it unlikely for the same classifier to be optimal in consecutive rounds.

Problem 3

The question at hand is whether in the AdaBoost algorithm, it is possible for a classifier h_{t+n} to be the same as a previous classifier h_t for some t and $n > 1$. To answer this, we must consider the adaptive nature of AdaBoost in updating weights and selecting classifiers over multiple rounds.

AdaBoost updates the weights of the training examples after each round based on the performance of the current classifier. This weight update process is designed to increase the weights of the misclassified examples and decrease the weights of the correctly classified ones. As a result, each classifier in subsequent rounds is chosen to address the weaknesses of the previous classifiers.

Now, considering a future round $t+n$, where $n > 1$, the distribution D_{t+n} would have been influenced by all the intermediate classifiers $h_{t+1}, h_{t+2}, \ldots, h_{t+n-1}$. Each of these classifiers would have altered the distribution based on their respective performances. Thus, the data distribution that h_{t+n} faces would be substantially different from the distribution h_t faced.

However, it is theoretically possible, though highly unlikely, that $h_{t+n} = h_t$ under certain conditions:

1. The set of classifiers H is finite and not very large. In such a case, the algorithm might cycle through all available classifiers and eventually return to h_t.

2. The data and the classifiers exhibit certain symmetries or periodicities, which might lead to similar distributions after a number of rounds.

In practice, given the diversity of real-world data and the typical size and variety of classifiers in H, it is improbable that $h_{t+n} = h_t$ for $n > 1$. AdaBoost's adaptive nature and the diversity of classifiers usually prevent the reselection of a previously chosen classifier, especially after several rounds.

Problem 4

Given the weighted empirical exponential loss function:

$$\hat{l}_{D_{t-1}}(h_t) = \sum_{i=1}^{N} D_{t-1}(i) \exp(-\alpha_t y_i h_t(x_i))$$

To find the optimal α_t, we take the derivative of this loss function with respect to α_t and set it to zero. The derivative is given by:

$$\frac{\partial}{\partial \alpha_t} \hat{l}_{D_{t-1}}(h_t) = \frac{\partial}{\partial \alpha_t} \sum_{i=1}^{N} D_{t-1}(i) \exp(-\alpha_t y_i h_t(x_i))$$

Simplifying this, we get:

$$\sum_{i=1}^{N} -D_{t-1}(i) y_i h_t(x_i) \exp(-\alpha_t y_i h_t(x_i))$$

Setting this derivative to zero for minimization:

$$\sum_{i=1}^{N} -D_{t-1}(i) y_i h_t(x_i) \exp(-\alpha_t y_i h_t(x_i)) = 0$$

To solve for α_t, this equation is typically solved numerically or using approximation methods, since a simple closed-form solution is generally not possible.