UNIVERSITY OF
MICHIGAN

Department of Mathematics

Math 628

HW 1

September 22, 2025

Question

Consider a binary classification problem with training examples $\{(x_i, y_i)\}_{i=1}^N$, where $y_i \in \{-1, +1\}$. Let the ensemble model be defined as

$$F(x) = \sum_{m=1}^{M} \alpha_m h_m(x),$$

where $h_m(x)$ is the m-th weak learner (e.g., a decision stump) and α_m is its weight.
AdaBoost is designed to minimize the exponential loss:

$$L(F) = \sum_{i=1}^{N} \exp\left(-y_i F(x_i)\right).$$

1. Derive the negative gradient of the loss function with respect to the model's output at an individual data point, i.e., compute

   $$-\frac{\partial L(F)}{\partial F(x_i)}.$$

   Show that this negative gradient is proportional to $y_i \exp\left(-y_i F(x_i)\right)$.

2. Explain how, in each boosting iteration, selecting the weak learner $h_m(x)$ to approximate the negative gradient is equivalent to choosing the direction of steepest descent in function space.

3. Derive the update rule for the weight α_m assigned to the weak learner $h_m(x)$ by performing a line search that minimizes the exponential loss along the direction given by $h_m(x)$. Provide all intermediate steps in your derivation.

4. Combine your results from the previous parts to demonstrate that AdaBoost is equivalent to gradient boosting with the exponential loss function. Discuss any assumptions made in your derivations and comment on the limitations or conditions under which this equivalence holds.

Question

In this question, we will verify the claim from lecture that "most" points in a high-dimensional space are far away from each other, and also approximately the same distance. There is a very neat proof of this fact which uses the properties of expectation and variance.

1. First, consider two independent univariate random variables X and Y sampled uniformly from the unit interval $[0, 1]$. Determine the expectation and variance of the random variable Z, defined as the squared distance $Z = (X - Y)^2$. You are allowed to evaluate integrals numerically (using `scipy.integrate.quad`), but you should explain what integral(s) you are evaluating, and why.

2. Now suppose we sample two points independently from a unit cube in d dimensions. Observe that each coordinate is sampled independently from $[0, 1]$, i.e., we can view this as sampling random variables $X_1, X_2, ..., X_d, Y_1, Y_2, ..., Y_d$ independently from $[0, 1]$. The squared Euclidean distance can be written as $R = Z_1 + ... + Z_d$, where $Z_i = (X_i - Y_i)^2$. Using the properties of expectation and variance, determine $E[R]$ and $Var[R]$. You may give your answer in terms of the dimension d, and $E[Z]$ and $Var[Z]$.

3. Based on your answer to part (b), compare the mean and standard deviation of R to the maximum possible squared Euclidean distance (i.e., the distance between opposite corners of the cube). Why does this support the claim that in high dimensions, "most points are far away from each other"?

Question

Consider m estimators $h_1, h_2, ..., h_m$, each of which accepts an input x and produces an output y, i.e., $y_i = h_i(x)$. These estimators might be generated through a Bagging procedure, but that is not necessary to the result that we want to prove. Consider the squared error loss function

$$L(y,t) = \frac{1}{2}(y-t)^2.$$

Show that the loss of the average estimator

$$\bar{h}(x) = \frac{1}{m}\sum_{i=1}^{m} h_i(x)$$

is smaller than the average loss of the estimators. That is, for any x and t, we have

$$L(\bar{h}(x), t) \leq \frac{1}{m}\sum_{i=1}^{m} L(h_i(x), t).$$