1 Boosting (15 points)

In this problem, you will develop an alternative way of forward stagewise boosting. The overall goal is to derive an algorithm for choosing the best weak learner h_t at each step such that it best approximates the gradient of the loss function with respect to the current prediction of labels. In particular, consider a binary classification task of predicting labels $y_i \in \{+1, -1\}$ for instances $\mathbf{x}_i \in \mathbb{R}^d$, for $i = 1, \ldots, n$. We also have access to a set of weak learners denoted by $\mathcal{H} = \{h_j, j = 1, \ldots, M\}$. In this framework, we first choose a loss function $L(y_i, \hat{y}_i)$ in terms of current labels and the true labels, e.g. least squares loss $L(y_i, \hat{y}_i) = (y_i - \hat{y}_i)^2$. Then we consider the gradient g_i of the cost function $L(y_i, \hat{y}_i)$ with respect to the current predictions $\hat{y}_i$ on each instance, i.e. $g_i = \frac{\partial L(y_i, \hat{y}_i)}{\partial \hat{y}_i}$. We take the following steps for boosting:

(a) **Gradient Calculation** (3 points) In this step, we calculate the gradients $g_i = \frac{\partial L(y_i, \hat{y}_i)}{\partial \hat{y}_i}$.

(b) **Weak Learner Selection** (6 points) We then choose the next learner to be the one that can best predict these gradients, i.e. we choose

$$h^* = \arg\min_{h \in \mathcal{H}} \left(\min_{\gamma \in \mathbb{R}} \sum_{i=1}^{n} (-g_i - \gamma h(\mathbf{x}_i))^2 \right)$$

We can show that the optimal value of the step size γ can be computed in the closed form in this step, thus the selection rule for h^* can be derived independent of γ.

(c) **Step Size Selection** (6 points) We then select the step size α^* that minimizes the loss:

$$\alpha^* = \arg\min_{\alpha \in \mathbb{R}} \sum_{i=1}^{n} L(y_i, \hat{y}_i + \alpha h^*(\mathbf{x}_i)).$$

For the squared loss function, α^* should be computed analytically in terms of y_i, $\hat{y}_i$, and h^*. Finally, we perform the following updating step:

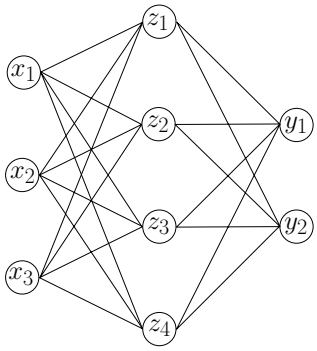
$$\hat{y}_i \leftarrow \hat{y}_i + \alpha^* h^*(\mathbf{x}_i).$$

In this question, you are asked to derive all of the steps for the squared loss function $L(y_i, \hat{y}_i) = (y_i - \hat{y}_i)^2$.

2 Neural Network (15 points)

(a) (5 points) Show that a neural network with a single logistic output and with linear activation functions in the hidden layers (possibly with multiple hidden layers) is equivalent to the logistic regression.

(b) (10 points) Consider the following neural network with one hidden layer. Each hidden layer is defined as $z_k = \tanh(\sum_{i=1}^{3} w_{ki} x_i)$ for $k = 1, \ldots, 4$ and the outputs are defined as $y_j = \sum_{k=1}^{4} v_{jk} z_k$ for $j = 1, 2$. Suppose we choose the squared loss function for every pair, i.e. $L(\mathbf{y}, \widehat{\mathbf{y}}) = \frac{1}{2}\left((y_1 - \widehat{y}_1)^2 + (y_2 - \widehat{y}_2)^2\right)$, where y_j and $\widehat{y}_j$ represent the true outputs and our estimations, respectively. Write down the backpropagation updates for estimation of w_{ki} and v_{jk}.



3 Clustering (15 points)

In the lectures, we discussed k-means. Given a set of data points $\{\mathbf{x}_n\}_{n=1}^N$, the method minimizes the following distortion measure (or objective or clustering cost):

$$D = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|\mathbf{x}_n - \boldsymbol{\mu}_k\|_2^2$$

where $\boldsymbol{\mu}_k$ is the prototype of the k-th cluster. r_{nk} is a binary indicator variable. If $\mathbf{x}_n$ is assigned to the cluster k, r_{nk} is 1 otherwise r_{nk} is 0. For each cluster, $\boldsymbol{\mu}_k$ is the representative for all the data points assigned to that cluster.

(a) (5 points) In the lecture, we showed but did not prove that, $\boldsymbol{\mu}_k$ is the mean of all such data points. That is why the method has MEANS in its name and we keep referring to $\boldsymbol{\mu}_k$ as MEANS, CENTROIDS, etc. You will prove this rigorously next. Assuming all r_{nk} are known (that is, you know the assignments of all N data points), show that if $\boldsymbol{\mu}_k$ is the mean of all data points assigned to the cluster k, for any k, then the objective D is minimized. This justifies the iterative procedure of k-means[1].

(b) (10 points) We now change the distortion measure to

$$D = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} \|\mathbf{x}_n - \boldsymbol{\mu}_k\|_1$$

In other words, the measurement of "closeness" to the cluster prototype is now using L_1 norm ($\|\mathbf{z}\|_1 = \sum_d |z_d|$).

Under this new cost function, show the $\boldsymbol{\mu}_k$ that minimizes D can be interpreted as the elementwise median of all data points assigned to the k-th cluster. (The elementwise median of a set of vectors is defined as a vector whose d-th element is the median of all vectors' d-th elements.)

[1] More rigorously, one would also need to show that if all $\boldsymbol{\mu}_k$ are known, then r_{nk} can be computed by assigning $\mathbf{x}_n$ to the nearest $\boldsymbol{\mu}_k$. You are not required to do so.

4 Mixture Models (15 points)

Suppose $X_1, \ldots, X_n$ are i.i.d distribution random variables with the density function $f_X(x, \lambda) = \lambda e^{-\lambda x}$, for $x \geq 0$ and 0 otherwise. We observe $Y_i = \min\{X_i, c_i\}$ for some fixed known c_i. Assume (for simplicity) that this cut-off happens only for the last $n - r$ variables; i.e. $y_i = c_i$ for $i = r+1, \ldots, n$. The goal is to estimate the value of λ using EM algorithm.

(a) (5 points) Write the log-likelihood in terms of unobserved variables X_i.

(b) (5 points) Write down the E-Step and take the expectation.

(c) (5 points) Write down the M-Step and find the new value of λ.

5 Programming (40 points)

In this problem, you will implement k-means. The problem consists of 2 parts. In part 1, you will implement k-means and evaluate it on a synthetic dataset. In part 2, we will provide you an image. Your task is to use k-means to do vector quantization. **There are 4 questions in total.**

5.1 Data

You are provided with two datasets. The first one is `2DGaussian.csv` which has two dimensions (for part 1), and the second one is `hw4.jpg` (for part 2).

5.2 Implement k-means

As we studied in the class, k-means tries to minimize the following distortion measure (or objective function):

$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} ||\mathbf{x}_n - \boldsymbol{\mu}_k||_2^2$$

where r_{nk} is an indicator variable:

$$r_{nk} = 1 \quad \text{if and only if } \mathbf{x}_n \text{ belongs to cluster } k$$

and $(\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_k)$ are the cluster centers with the same dimension of data points.

(a) (10 points) Implement k-means using random initialization for cluster centers. The algorithm should run until none of the cluster assignments are changed. Run the algorithm for different values of $K \in \{2, 3, 5\}$, and plot the clustering assignments by different colors and markers. (you need to report 3 plots.)

(b) (10 points) Run the algorithm 5 times, each time with different initialization of K cluster centers picked at random, with $K = 4$. Each time, you need to run the algorithm for 50 iterations and record the value of objective function at each iteration. In a single figure, plot all the five curves with different colors and markers. (you need to report 1 plot.)

(c) (5 points) Does k-means always converge after finite number of iterations? Briefly state your reasons.

5.3 Vector Quantization Using k-means

One important application of k-means is vector quantization. This is the process of replacing our data points with the prototypes μ_k from the clusters they are assigned to. This technique can be used for image compression.

(d) (15 points) Download the image `hw4.jpg`, and perform k-means clustering to vector-quantize pixels according their RGB color. Reconstruct the image using the colors in the centers for values of $K \in \{3, 8, 15\}$. Show the reconstructed images. (you need to show three reconstructed images.)

6 Submission Instructions

You need to provide the followings:

- Provide your answers for all of the problems **in hard copy** (if you are printing it as black-white, please make sure that you are using different line styles and colors for each curve in your plot, if there are more than one curves. The papers need to be stapled and submitted to the CS front desk. We suggest printing it as double-sided to save papers.
- Submit ALL the code and report via Blackboard. The only acceptable language is MATLAB.
  - For your program, you MUST include the main function called `CSCI567_hw4.m` in the root of your folder. After running this main file, your program should be able to generate all of the results needed for this programming assignment, either as plots or console outputs. You can have multiple files (i.e your sub-functions), however, the only requirement is that once we unzip your folder and execute your main file, your program should execute correctly. Please double-check your program before submitting. You should only submit one `.zip` file. No other formats are allowed except `.zip` file. Also, please name it as `[lastname]_[firstname]_hw4.zip`

Collaboration You may collaborate. However, you need to write your own solution and submit separately. You also need to list with whom you have discussed.