CS 391L: Machine Learning **Spring 2021**

Homework 2 - Programming

Lecture: Prof. Adam Klivans

Keywords: SGD, Boosting

1. In this problem, we will compare the performance of three different types of algorithms on a synthetic training set. First, to generate the training set, pick a weight vector $w \in \mathbb{R}^{10}$ of dimension 10 at random (normalize it so that its Euclidean norm $\|w\| = \sqrt{\sum_i w_i^2}$ is equal to 1, i.e., pick w at random (e.g. each entry from $N(0,1)$) and then take $w/\|w\|$). Then generate a training set of size m of the form $\{(x^1, y^1), \ldots, (x^m, y^m)\}$ where each $x^i \in \mathbb{R}^{10}$ is a random vector of dimension 10 and each entry of x^i is chosen from a Gaussian $N(0,1)$ (you may use built-in methods for this). The label y^i is 0 or $+1$ and should be randomly chosen such that $y^i = +1$ with probability exactly $\sigma(w \cdot x^i)$, where σ is the sigmoid function, aka the standard logistic function, and $y^i = 0$ otherwise. The goal is to learn w.

   - Algorithm 1 is logistic regression (you may use built-in methods for this).
   - Algorithm 2 is gradient descent where you train a model of the form $\sigma(w' \cdot x)$ (with parameter w') with respect to square loss, i.e. the loss function is $\frac{1}{2}(\sigma(w' \cdot x) - y)^2$, averaged over the points in the training set (code this up yourself, including calculating the gradient).
   - Algorithm 3 is stochastic gradient descent again with respect to square loss, where during each iteration we use the gradient at one random point from the training set.

   You should measure success as follows: compute $\|w - w'\|$ where w' is the weight vector output by your algorithm. For each value of m, do the following several times (say 10 times) and take the average:

   (a) Generate a fresh w and fresh training data

   (b) Train your algorithm and obtain w'

   (c) Calculate $\|w - w'\|$

   Plot the results for all three algorithms for $m = 50, 100, 150, 200, 250$. For each algorithm, also record the time taken for the entire experiment.

2. NEW: In this problem we will see a toy example of how to use AdaBoost. Please read the documentation on AdaBoostClassifier in sklearn, including how to set the base weak learner (i.e. the `base_estimator`). We will use the Wisconsin breast cancer data set again. For each depth in $1, \ldots, 5$, instantiate an AdaBoost classifier with the base learner set to be a decision tree of that depth, and then record the 10-fold cross-validated error on the entire breast cancer data set. Plot the resulting curve of accuracy against base classifier depth.