# CS 688: Homework 2

Due: Friday Feb 23 by 10:00 PM

**Notes:**

- Homework is due **by 10:00 PM on the due date.** Remember that you may not use more than 2 late days on any one homework, and you only have a budget of 5 in total.
- Please keep in mind the collaboration policy as specified in the course syllabus. If you discuss questions with others you **must** write their names on your submission, and if you use any outside resources you **must** reference them. **Do not look at each others' writeups, including code.**
- There are 3 problems on 2 pages in this homework.
- Remember that you will be graded on the quality of your report and writing. We may or may not even look at your code. So please write carefully and clearly.

**Problems:**

1. (30 points) This problem involves understanding the properties of gradient descent. Start by encoding a gradient descent algorithm for learning a logistic regression model. Your function should take as input a matrix $\mathbf{X}$, where each row corresponds to a training example, and a column vector $\mathbf{y}$ where each row corresponds to a label, and return the learned weight vector $\mathbf{w}$ (which should have one more element than each training example, with the convention that this first element represents the intercept), the cross-entropy error, and the classification error (0-1 loss) it achieved on the training set. Use a learning rate $\eta = 10^{-5}$ and automatically terminate the algorithm if the magnitude of each term in the gradient is below $10^{-3}$ at any step.

   You will train and test your model on the "Cleveland" dataset, which you can learn more about here: `https://archive.ics.uci.edu/ml/datasets/Heart+Disease` Both the training and test sets are available on Piazza.

   Run experiments where you learn logistic regression models on the training set when using three different bounds on the maximum number of iterations: ten thousand, one hundred thousand, and one million. In your writeup, report all four of the following: (1) the cross-entropy error on the training set (2) the classification error on the training data (3) the classification error on the test data (4) the time it took to train your model. In your report, discuss the generalization properties (the **difference** between training and test set classification errors) of the model. How does this relate to the cross-entropy error on the training set?

   Now, scale each of the features in your training data by subtracting the mean and dividing by the standard deviation for each of the features in advance of calling the training function.

Experiment with the learning rate $\eta$ (you may want to start by trying different orders of magnitude), this time using a tolerance (how close to zero you need each element of the gradient to be in order to terminate) of $10^{-6}$. Report the results in terms of number of iterations until the algorithm terminates, and also the final cross-entropy error. Spend some time discussing and analyzing your results in your writeup.

2. (10 points) Consider the loss function $E(\mathbf{w}) = (\max(0, 1 - y_i \mathbf{w}^T \mathbf{x}_i))^2$. What is the gradient $\nabla_i(\mathbf{w})$? Using this, write down what the stochastic gradient descent updates would be for this loss function.

3. (10 points) Consider the two different notions of fairness we discussed in class for the recidivism prediction problem (equal false positive rates and calibration). Which do you think is a more appropriate definition of fairness for the problem of deciding whom to keep in jail? Why? Also discuss some possible drawbacks to prioritizing the measure of fairness that you advocate for.