# Problem Set 1

CS 4375

Due: 9/19/2025 by 11:59pm

Note: all answers should be accompanied by explanations for full credit. All code used as part of your solutions must be included to receive credit. **NO machine learning libraries may be used in your solutions, e.g., scikitlearn.** Late homeworks will not be accepted.

## Warm-Up: Properties of Convex Functions (20 pts)

Recall that a function $f : \mathbb{R}^n \to \mathbb{R}$ is convex if for all $x, y \in \mathbb{R}^n$ and $\lambda \in [0, 1]$, $\lambda f(x) + (1-\lambda)f(y) \geq f(\lambda x + (1-\lambda)y)$. Using this definition, you can show that $g(x) = f(Ax + b)$ is a convex function for $x \in \mathbb{R}^m$ whenever $f : \mathbb{R}^n \to \mathbb{R}$ is a convex function and $A \in \mathbb{R}^{n \times m}$ and $b \in \mathbb{R}^n$:

$$\begin{aligned} g(\lambda x + (1-\lambda)y) &= f(A(\lambda x + (1-\lambda)y) + b) \\ &= f(\lambda(Ax + b) + (1-\lambda)(Ay + b)) \\ &\leq \lambda f(Ax + b) + (1-\lambda) f(Ay + b) \\ &= \lambda g(x) + (1-\lambda) g(y), \end{aligned}$$

where the first equality is by definition, the inequality is by convexity of $f$, and the final equality is also by definition. Using the same strategy as above, prove that convex functions have the following properties.

1. $g(x) = w \cdot f(x)$ is a convex function for $x \in \mathbb{R}^n$ whenever $f : \mathbb{R}^n \to \mathbb{R}$ is a convex function and $w \in \mathbb{R}, w \geq 0$.
2. $g(x) = f_1(x) + f_2(x)$ is a convex function for $x \in \mathbb{R}^n$ whenever $f_1 : \mathbb{R}^n \to \mathbb{R}$ and $f_2 : \mathbb{R}^n \to \mathbb{R}$ are convex functions
3. $g(x) = \max(f_1(x), f_2(x))$ is a convex function for $x \in \mathbb{R}^n$ whenever $f_1 : \mathbb{R}^n \to \mathbb{R}$ is a convex function and $f_2 : \mathbb{R}^n \to \mathbb{R}$ is a convex function.
4. $g(x) = \exp(f(x))$ is a convex function for $x \in \mathbb{R}^n$ whenever $f : \mathbb{R}^n \to \mathbb{R}$ is a convex function. Hint: use the fact that log is a concave function.

## Problem 1: Linear Regression (30 pts)

Consider the general linear regression problem from class with data observations $x^{(1)}, \ldots, x^{(M)} \in \mathbb{R}^n$ and corresponding labels $y^{(1)}, \ldots, y^{(M)} \in \mathbb{R}$ where the goal is to minimize the squared error,

$$\frac{1}{M}\left[\sum_{m=1}^{M} (a^T x^{(m)} + b - y^{(m)})^2\right],$$

over all possible choices of $a \in \mathbb{R}^n$ and $b \in \mathbb{R}$.

Consider a modified loss function of the form

$$\sum_{m \in \{1,\ldots,M\}} \left| a^T x^{(m)} + b - y^{(m)} \right|$$

1. Using the same strategy as the warm-up, argue that the modified loss is a convex function of $a \in \mathbb{R}^n$ and $b \in \mathbb{R}$.

2. Implement the subgradient descent method to minimize this loss function in Python or MATLAB. Your function should take as input an $m \times n$ matrix $X$ whose rows are the data observations and a vector $y$ of corresponding labels and return the $n+1$ dimensional vector $\begin{bmatrix} a \\ b \end{bmatrix}$.

   (a) How should you pick the step size?

3. More generally, what are the pros and cons of the modified loss versus the original least squares loss for (polynomial) regression problems?

## Problem 2: Perceptron Learning (50 pts)

Consider the data set (perceptron.data) attached to this homework. This data file consists of $M$ data elements of the form $(x_1^{(m)}, x_2^{(m)}, y^{(m)})$ where $x_1^{(m)}, x_2^{(m)} \in \mathbb{R}$ define a data point in $\mathbb{R}^2$ and $y^{(m)} \in \{-1, 1\}$ is the corresponding class label.

1. Consider using the feature map $\phi(x_1, x_2) = \begin{bmatrix} x_1 \\ x_2 \\ x_1^2 + x_2^2 \end{bmatrix}$

   (a) Prove that any circle in $\mathbb{R}^2$ can be represented in the feature space given by $\phi$.

   (b) Do all lines in the feature space correspond to circles?

2. In class, we saw how to use the perceptron algorithm to minimize the following loss function.

$$\frac{1}{M} \sum_{m=1}^{M} \max\{0, -y^{(m)} \cdot (w^T x^{(m)} + b)\}$$

   What is the smallest, in terms of number of data points, data set containing both class labels on which the perceptron algorithm under feature map $\phi$, i.e., $x^{(m)} \to \phi(x^{(m)})$, with step size one, fails to converge?

3. Implement the perceptron algorithm using the feature map $\phi$.

   (a) For each optimization strategy below, report the values of $w$, $b$, and the loss function for iterations $1, 10, 10^2, 10^3, 10^4$, and $10^5$. Each descent procedure should start from the initial point

$$w^0 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \qquad b^0 = 0$$

   and use the data set provided above and the feature map $\phi$.

i. Standard subgradient descent with the step size $\gamma_t = 1$ for each iteration.

ii. Stochastic subgradient descent where exactly one component of the sum is chosen to approximate the gradient at each iteration. Instead of picking a random component at each iteration, you should iterate through the data set starting with the first element, then the second, and so on until the $M^{th}$ element, at which point you should start back at the beginning again. Again, use the step size $\gamma_t = 1$.

iii. When using a fixed step size, does the rate of convergence change as you change the step size? Provide some example step sizes to back up your statements.