Homework 4 for Math 173A - Fall 2024

1. (a) Find an expression for the orthogonal projection of a point $x \in \mathbb{R}^n$ onto the convex set

$$B = \{z \in \mathbb{R}^n : 0 \leq z_i \leq 1 \text{ for each } i = 1, ..., n\}.$$

You need to show your work, and justify your answer. The expression can be written piecewise, and per dimension if it's easier / more compact.

Hint: It might be helpful to sketch B, when $n = 2$ (i.e., in 2 dimensions), and use the sketch to help you figure out what the projection should be.

(b) Let $f : \mathbb{R}^n \to \mathbb{R}$ be given by

$$f(x) = \|Ax\|_2^2 + a^T x$$

where $A \in \mathbb{R}^{n \times n}$ is a positive definite matrix, and $a \in \mathbb{R}^n$. Write a projected gradient descent algorithm to solve

$$\min_{x \in \Omega} f(x)$$

for $\Omega = B$, with B from part (a). You do not need to specify the step size for this problem.

(c) Repeat part (b) but for $\Omega = B_2^n = \{z \in \mathbb{R}^n : \|z\|_2 \leq 1\}$.

2. Consider the *hollow* sphere S in $\mathbb{R}^n$, i.e., the set $S := \{x \in \mathbb{R}^n : \|x\|_2^2 = 1\}$. Consider the function $f : \mathbb{R}^n \to \mathbb{R}$ given by

$$f(x) = x^T Q x$$

where Q is an $n \times n$ symmetric matrix. For this problem you may use the fact that $\nabla f(x) = 2Qx$.

(a) For an arbitrary point $y \in \mathbb{R}^n$, $\Pi(y)$ be the projection of y onto S. Find an expression for $\Pi(y)$ and give a short argument (i.e., proof) for why this is the correct expression. Make sure to handle the case $y = 0$ (i.e., the zero vector).

Hint: Recall that the projection minimizes $\|x - y\|$ for $y \in S$. One approach would be to consider the reverse triangle inequality $\|x - y\| \geq |\|x\| - \|y\||$ and find a projection formula that achieves the global lower bound (i.e. equality instead of inequality). This would prove you've found the projection.

(b) Is S a convex set?

(c) Write a projected gradient descent algorithm, with constant step size μ, for

$$\min_{x \in \mathbb{R}^n} x^T Q x \qquad \text{subject to} \qquad \|x\|_2^2 = 1.$$

(d) Is the projected gradient descent algorithm guaranteed to converge to the solution for small enough μ? If not, can you give an example of Q and an initialization $x^{(0)}$ where the algorithm won't converge? **Hint:** Consider a diagonal matrix Q where not all entries are equal.

3. **Coding Question:** Consider the function $f : \mathbb{R}^2 \to \mathbb{R}$ with $f(x_1, x_2) = (x_1 - 2)^2 + (x_2 - 2)^2 - x_1 x_2$ and the following constrained optimization problem:

$$\min_{x_1, x_2} f(x_1, x_2) \quad \text{subject to} \quad 0 \leq x_i \leq 1, \ i = 1, 2.$$

Write a projected gradient descent algorithm, with constant step size $\mu = 0.001$ starting at $(0.5, 0.5)$ for 175 iterations, for the above optimization problem. Plot the function value $f(x^{(t)})$ against the iteration t.

4. **Coding Question:** We will redo the MNIST coding question from HW3.5 but using different versions of gradient descent

$$w^{(t+1)} = w^{(t)} - \mu p^{(t)}.$$

This time, we will **differentiate 4's and 9's**. That is, instead of classifying images of 0 and 1, you classify the images of 4 and 9. You can reuse the template from the previous homework for loading / formatting MNIST. But you need to change the learning rate and the number of iterations.

(a) Implement and run L^∞ gradient descent with step size $\mu = 10^{-8}$. Run your algorithm for at least 1000 iterations and initialize with $w^{(0)} = 0$ (i.e. the zero vector). **Recall:** L^∞ gradient descent uses steps

$$p^{(t)} = \text{sign}(\nabla F(w^{(t)})) \|\nabla F(w^{(t)})\|_1.$$

(b) Implement and run L^1 gradient descent (aka. coordinate descent) with step size $\mu = 10^{-4}$. Run your algorithm for at least 1000 iterations and initialize with $w^{(0)} = 0$ (i.e. the zero vector). **Recall:** L^1 gradient descent uses steps

$$p^{(t)} = \text{sign}(\nabla_{j^*} F(w^{(t)})) \|\nabla F(w^{(t)}\|_\infty e_{j^*},$$

where j^* is the location of the largest entry of the gradient, and e_j is the zero vector with a 1 in the j^{th} entry.

(c) Compare these two descent plots of $F(w)$, along with the analogous plot for gradient descent from HW3.5. Which performs best, and do you have an argument for why? Do you think the performance would change with different step sizes?

(d) For the coordinate descent problem, rerun gradient descent but store a running sum of which entry of $p^{(t)}$ is nonzero at each iteration (not the actual value of the direction vector, just e_{j^*}). This will result in a size 784 vector of mostly zeros, and should have integers at various entries whose sum equals the number of iterations. Reshape this vector to be a 28x28 image and display the result. Why do you think these are the pixels that were chosen in the gradient? How can you use this to interpret the algorithm and its results?