CIS3333: Mathematics of Machine Learning Fall 2025

Homework 9

Due: November 28, 2025, 11:59 PM ET

Submission Instructions: Submit a single PDF to Gradescope. Show key steps and justify your answers conceptually.

Collaboration & AI Policy: You may discuss approaches with classmates, but write up your own solutions and list collaborators. If you use computational tools (including LLMs) for checking, cite them and ensure the reasoning is your own.

Note: Throughout this homework, you may assume all functions are differentiable whenever gradients or Hessians are used.

Problem 1: SGD vs. GD (10 points)

In this problem, we will compare the convergence rates and computational costs of Stochastic Gradient Descent (SGD) and Gradient Descent (GD).

Recall the convergence bound for SGD with a constant step size η:

$$\mathbb{E}[f(\bar{x}_T)] - f(x^*) \leq \frac{R^2}{2\eta T} + \frac{\eta G^2}{2},$$

where $R = \|x_0 - x^*\|_2$ and G^2 is a bound on the second moment of the stochastic gradients.

(a) (3 points) **Optimal Step Size for SGD.** Find the constant step size η^* (in terms of T, R, G) that minimizes the upper bound on the error. Plug this η^* back into the bound to show that the error rate of the average iterate $\bar{x}_T = \frac{1}{T}\sum_{t=0}^{T-1} x_t$ is:

$$\mathbb{E}[f(\bar{x}_T)] - f(x^*) \leq \frac{RG}{\sqrt{T}}.$$

(b) (2 points) **Optimal Step Size for GD.** For L-smooth convex functions, the final iterate x_T of Gradient Descent (GD) with constant step size $\eta \leq \frac{1}{L}$ satisfies:

$$f(x_T) - f(x^*) \leq \frac{R^2}{2\eta T}.$$

What is the optimal step size η we should use for GD? What is the resulting convergence rate in terms of L, R, T?

Now let us compare the two methods. For a dataset of size N,

- One step of GD computes the full gradient, costing N gradient evaluations.
- One step of SGD computes a single gradient, costing 1 gradient evaluation.

Suppose we have a fixed total budget of K gradient evaluations. Assume for simplicity that $L = R = G = 1$.

(c) (3 points) Express the number of iterations T_{GD} (for GD) and T_{SGD} (for SGD) possible with budget K. Then, write the error bounds for both methods strictly in terms of the budget K and N using the optimal rates from (a) and (b).

(d) (2 points) Find the critical budget K^* (in terms of N) where the two methods have approximately the same error bound. Does SGD win for $K < K^*$ or $K > K^*$?

Problem 2: Learning Rate Schedules for SGD (8 points)

In this problem, you will see how different learning-rate schedules change the convergence guarantee from lecture. For SGD with step sizes $(\eta_t)_{t=0}^{T-1}$ let

$$\sum_{t=0}^{T-1} \eta_t = S_T \quad \text{and} \quad \sum_{t=0}^{T-1} \eta_t^2 = Q_T.$$

Then, the convergence guarantee from lecture can be simplified to:

$$\mathbb{E}[f(\tilde{x}_T)] - f(x^*) \;\leq\; \frac{R^2}{2S_T} + \frac{G^2 Q_T}{2S_T},$$

where $\tilde{x}_T = \frac{1}{S_T} \sum_{t=0}^{T-1} \eta_t x_t$ is the weighted average of the iterates, $R := \|x_0 - x^*\|_2$ and G^2 bounds the second moment of the stochastic gradients.

We will compare three step-size schedules:

$$\text{(i) } \eta_t = \eta, \qquad \text{(ii) } \eta_t = \frac{\eta}{t+1}, \qquad \text{(iii) } \eta_t = \frac{\eta}{\sqrt{t+1}}$$

for a fixed $\eta > 0$. You may use the following summation upper bounds:

$$\sum_{t=0}^{T-1} \frac{1}{t+1} \leq 1 + \log T, \quad \sum_{t=0}^{T-1} \frac{1}{(t+1)^2} \leq 2, \quad \sum_{t=0}^{T-1} \frac{1}{\sqrt{t+1}} \leq 2\sqrt{T}.$$

(a) (3 points) Compute the error bound for each of the three step-size schedules above by computing S_T and Q_T for each schedule in terms of T, η, R, and G and plugging them into the convergence bound.

(b) (3 points) Does the error bound go to 0 as $T \to \infty$ when η is fixed for each schedule? Why or why not?

(c) (2 points) Which schedule converges fastest for fixed η? Why?

Problem 3: Adaptive Methods and Sign GD (8 points)

Recall the 2D diagonal quadratic from class:

$$f(x_1, x_2) = \frac{1}{2}(\lambda_1 x_1^2 + \lambda_2 x_2^2)$$

with $0 < \lambda_1 < \lambda_2$. This problem is ill-conditioned when $\kappa = \lambda_2/\lambda_1$ is large.

We saw that adaptive methods like RMSProp attempt to fix ill-conditioning by adapting the step size for each coordinate. The key idea is to normalize the gradient by an estimate of the curvature. In particular, consider the update rule for RMSProp: for $j = 1, 2$,

$$s_{t+1,j} = \beta s_{t,j} + (1-\beta) g_{t,j}^2 \quad \text{and} \quad x_{t+1,j} = x_{t,j} - \frac{\eta}{\sqrt{s_{t+1,j}} + \epsilon} g_{t,j},$$

where $g_t = \nabla f(x_t) \in \mathbb{R}^2$.

(a) (3 points) Suppose we set $\beta = 0$ and $\epsilon = 0$. Show that the update rule simplifies to

$$x_{t+1,j} = x_{t,j} - \eta \operatorname{sign}(g_{t,j}), \quad \text{for} \quad j = 1, 2$$

where $\operatorname{sign}(x) = 1$ if $x > 0$, -1 if $x < 0$, and 0 if $x = 0$.

This algorithm is known as *Sign GD*. Unlike standard GD which takes steps proportional to the gradient magnitude, Sign GD takes steps of fixed size η along each coordinate, moving purely based on the direction of the gradient.

Let us see how Sign GD behaves on the 2D diagonal quadratic. Suppose we start at $x_0 = (1, 1)$ and run the update derived in (a) with step size $\eta = 0.01$.

(b) (3 points) Determine the number of steps T required for each coordinate to reach 0.

(c) (2 points) Are the convergence times the same for both coordinates? Do they depend on the condition number $\kappa = \lambda_2/\lambda_1$? Why or why not?