Backpropagation and Optimizers

Due: Friday, November 7th at 11:59 pm

Deliverables. Submit a PDF of your write-up to Gradescope *HW3 Written*
Note: we **highly discourage** very long answers, your free response answers should be 1-3 sentences.

Honor Code

Write and sign the following statement:

> "I certify that all solutions in this document are entirely my own and that I have not looked at anyone else's solution. I have given credit to all external sources I consulted."

Backpropagation Practice

Let's practice calculating the gradient using adjoints. We're given the following function and want to find the gradient of the function with respect $\mathbf{x} = [x_1 \;\; x_2 \;\; x_3]$:

$$f(\mathbf{x}) = \frac{x_1}{x_2} e^{x_2 + x_3} - x_2^2$$

Q1. Constructing the Computation Graph Draw the computation graph (check out lecture 12 for reference). We recommend handwriting it instead of using LaTeX.

Q2. Computing Adjoints Now, compute the adjoint for each node in your computation graph with respect to $\mathbf{x} = [x_1 \;\; x_2 \;\; x_3] = [1 \;\; 2 \;\; 3]$. What is the gradient with respect to $\mathbf{x}$? We recommend that you use your computation graph and write it out instead of typing.

Optimizers Overview

Adaptive gradient methods are among the most widely used optimization algorithms in deep learning. In this assignment, we focus on two main algorithms: **Adam** and **AdamW** (briefly AdaGrad, RMSProp). Because the primary sources are dense, please start with the short blog posts below, then (optionally) consult the original papers.

Recommended reading (short):

- Understanding Deep Learning Optimizers: Momentum, AdaGrad, RMSProp, Adam — quick refresher on the progression from SGD to Adam.
- Adam vs. AdamW (A. Bennaceur) — concise comparison highlighting why decoupled weight decay matters.
- **Supplement (Section 3):** CS182 Discussion 2 Solutions (Spring 2021) — see Section 3 for a compact optimizer summary.

Optional primary sources:

- Adam: A Method for Stochastic Optimization (Kingma & Ba, 2015)
- Decoupled Weight Decay Regularization (Loshchilov & Hutter, 2017)

Before answering the questions, also briefly review the original Adam algorithm (Algorithm 1 from Kingma & Ba, 2014) and AdamW, reproduced below for reference.

Algorithm 1 Adam, our proposed algorithm for stochastic optimization (Kingma & Ba, 2014)

Require: α: Stepsize
Require: $\beta_1, \beta_2 \in [0, 1)$: Exponential decay rates for the moment estimates
Require: $f(\theta)$: Stochastic objective function with parameters θ
Require: θ_0: Initial parameter vector

$m_0 \leftarrow 0$ ▷ Initialize 1st moment vector
$v_0 \leftarrow 0$ ▷ Initialize 2nd moment vector
$t \leftarrow 0$ ▷ Initialize timestep
while θ_t not converged **do**
  $t \leftarrow t + 1$
  $g_t \leftarrow \nabla_\theta f_t(\theta_{t-1})$ ▷ Get gradients
  $m_t \leftarrow \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot g_t$ ▷ EMA of gradients
  $v_t \leftarrow \beta_2 \cdot v_{t-1} + (1 - \beta_2) \cdot g_t^2$ ▷ EMA of squared gradients
  $\hat{m}_t \leftarrow m_t / (1 - \beta_1^t)$ ▷ Bias-corrected 1st moment
  $\hat{v}_t \leftarrow v_t / (1 - \beta_2^t)$ ▷ Bias-corrected 2nd moment
  $\theta_t \leftarrow \theta_{t-1} - \alpha \cdot \hat{m}_t / (\sqrt{\hat{v}_t} + \epsilon)$ ▷ Update
end while
return θ_t

Algorithm 2 Adam with L2 regularization and Adam with decoupled weight decay (AdamW)

Require: $\alpha = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$, $\lambda \in \mathbb{R}$

initialize $t \leftarrow 0$, parameter vector $\theta_{t=0} \in \mathbb{R}^n$, first moment $m_{t=0} \leftarrow 0$, second moment $v_{t=0} \leftarrow 0$, schedule multiplier $\eta_{t=0} \in \mathbb{R}$

repeat

  $t \leftarrow t + 1$

  $\nabla f_t(\theta_{t-1}) \leftarrow \textsc{SelectBatch}(\theta_{t-1})$

  $g_t \leftarrow \nabla f_t(\theta_{t-1}) + \lambda \theta_{t-1}$ ▷ (L2-regularized Adam) add penalty inside gradient

  $m_t \leftarrow \beta_1 m_{t-1} + (1 - \beta_1)\, g_t$

  $v_t \leftarrow \beta_2 v_{t-1} + (1 - \beta_2)\, g_t^2$

  $\hat{m}_t \leftarrow m_t / (1 - \beta_1^t)$

  $\hat{v}_t \leftarrow v_t / (1 - \beta_2^t)$

  $\eta_t \leftarrow \textsc{SetScheduleMultiplier}(t)$

  $\theta_t \leftarrow \theta_{t-1} - \eta_t \left(\alpha \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon} + \lambda \theta_{t-1} \right)$ ▷ (AdamW) decoupled decay

until stopping criterion is met

return optimized parameters θ_t

Q3. Reviewing Previous Optimizers. We have learned how variants of gradient descent improve training speed and stability. For each method — **Momentum**, **AdaGrad**, and **RMSProp** — write the update equations, state the key idea (one concise sentence), and describe the problem it solves (one concise sentence). We have provided SGD for you.

SGD

Update: $\theta_t \leftarrow \theta_{t-1} - \alpha\, g_t$, where $g_t = \nabla_\theta f_t(\theta_{t-1})$.

Key idea: Use a random minibatch each step for computational efficiency.

Problem it solves: Full-batch gradient descent is slow and redundant; minibatching makes updates cheaper.

Q4. Handling Large Gradients. What happens when gradients are very large in plain gradient descent, and how does Adam's update rule address this issue?

Q5. Introducing AdamW. How does AdamW decouple weight decay from the gradient update? Why does this help generalization?

Q6. AdamW Weight Decoupling. How does decoupling weight decay affect hyperparameter sensitivity?

Q7. Tracing Adam's Origins (Line Mapping). As seen in lecture, Adam builds on ideas from several earlier optimizers. Now let's explore exactly how these ideas are used in Adam. Below are the key lines (7–11) from the Adam algorithm.

For each line, label which previous algorithm inspired it (or indicate if it was newly introduced) using one of the following categories:

- Momentum
- Adaptive Scaling (RMSProp/AdaGrad)
- Newly Introduced by Adam
- Combination (merges two or more earlier ideas into a single update rule)

Briefly explain your reasoning for each label.

Algorithm 3 Adam: Core Update Lines (Kingma & Ba, 2014)

$m_t \leftarrow \beta_1 m_{t-1} + (1-\beta_1) g_t$	▷ Line 7
$v_t \leftarrow \beta_2 v_{t-1} + (1-\beta_2) g_t^2$	▷ Line 8
$\hat{m}_t \leftarrow m_t / (1-\beta_1^t)$	▷ Line 9
$\hat{v}_t \leftarrow v_t / (1-\beta_2^t)$	▷ Line 10
$\theta_t \leftarrow \theta_{t-1} - \alpha \cdot \hat{m}_t / (\sqrt{\hat{v}_t} + \epsilon)$	▷ Line 11

Q8. Step 1 Gradient: Adam and AdamW updates. Consider the scalar objective

$$f(\theta) = \theta^2, \qquad \nabla_\theta f(\theta) = g_t = 2\theta_t.$$

Use the following hyperparameters unless otherwise specified:

$$\theta_0 = 10, \quad \alpha = 0.1, \quad \lambda = 0.01, \quad \beta_1 = 0.9, \quad \beta_2 = 0.99, \quad \epsilon = 10^{-8}, \quad m_0 = v_0 = 0.$$

Compute all quantities for $t = 1$.

(a) Compute g_1, m_1, v_1.

(b) Compute bias-corrected $\hat{m}_1, \hat{v}_1$ (feel free to refer to Algorithm Block provided in Q1).

(c) Compute $\theta_1^{\text{Adam}} = \theta_0 - \alpha \frac{\hat{m}_1}{\sqrt{\hat{v}_1} + \epsilon}$ (step 1 update of Adam).

(d) Compute $\theta_1^{\text{AdamW}} = \theta_1^{\text{Adam}} - \alpha \lambda \theta_0$ (step 1 update of AdamW).

Q9. Step 2 Gradient: Adam continuation ($\beta_2 = 0.99$). Compute the second update step for Adam (you don't have to compute for AdamW). In other words, compute θ_2^{Adam}.

Q10. Conceptual reflection. Keep the baseline at $\beta_2 = 0.99$. Now imagine we set $\beta_2 = 0.5$ (all else unchanged). Without redoing the full calculation, explain qualitatively what changes at $t = 1$ and $t \geq 2$, and how this would affect stability vs. responsiveness.