Homework #2: Deep and Reinforcement Learning

Module 1, Lectures 4–5: Optimisation and Regularisation

Srikanth Pai, Asst. Professor, MSE

Instructions

Answers should be concise and mathematically precise. State any result you invoke before using it.

Problem Set

1. Gradient descent and momentum on a quadratic.

Consider the scalar loss $L(\theta) = \frac{1}{2}a\theta^2$ with $a > 0$.

(a) For gradient descent $\theta_{t+1} = \theta_t - \eta\, L'(\theta_t)$, show that $\theta_t \to 0$ if and only if $\eta < 2/a$.
(b) Add momentum ($\beta \in [0, 1)$): $v_{t+1} = \beta v_t + L'(\theta_t)$, $\theta_{t+1} = \theta_t - \eta v_{t+1}$. Eliminate v_t to write the update as a second-order recurrence in θ alone.
(c) Let $\alpha = \eta a$ with $0 < \alpha < 1$. From the characteristic equation of that recurrence, find the value β_c at which the roots become complex and the convergence factor there. Compare with the no-momentum factor $1 - \alpha$.
(d) For $a = 4$, $\eta = 0.1$, $\beta \in \{0, 0.5, 0.9\}$: compute the roots and state whether the iterates converge.

2. Why Adam needs bias correction.

Adam forms $m_t = \beta_1 m_{t-1} + (1-\beta_1)\mathbf{g}_t$ and $v_t = \beta_2 v_{t-1} + (1-\beta_2)\mathbf{g}_t^2$ (componentwise square), with $\hat{m}_t = m_t/(1-\beta_1^t)$ and $\hat{v}_t = v_t/(1-\beta_2^t)$. Take $m_0 = v_0 = 0$ and a constant gradient $\mathbf{g}_t = \mathbf{g}$.

(a) Show that $m_t = (1-\beta_1^t)\mathbf{g}$ and $v_t = (1-\beta_2^t)\mathbf{g}^2$, hence $\hat{m}_t = \mathbf{g}$ and $\hat{v}_t = \mathbf{g}^2$.
(b) The Adam step is $-\eta\, \hat{m}_t/(\sqrt{\hat{v}_t} + \varepsilon)$. For $\beta_1 = 0.9$, $\beta_2 = 0.999$ (ignore ε), compare the uncorrected first step $m_1/\sqrt{v_1}$ with the corrected $\hat{m}_1/\sqrt{\hat{v}_1}$. Is the uncorrected first step larger or smaller than intended? One line: why?
(c) For $\beta_1 = 0.9$, find the smallest t with $1 - \beta_1^t > 0.99$. Is bias correction still material at $t = 100$?

3. L2 regularisation as MAP estimation.

Let $\mathcal{D} = \{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^n$ be a training set and let $L(\mathbf{w}) = -\sum_i \log p(y^{(i)} \mid \mathbf{x}^{(i)}; \mathbf{w})$ be the negative log-likelihood.

(a) Suppose we place a Gaussian prior on the weights: $\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \tau^2 I)$. Write down $\log p(\mathbf{w} \mid \mathcal{D})$ up to a constant using Bayes' theorem. Show that maximising this posterior is equivalent to minimising

$$\tilde{L}(\mathbf{w}) = L(\mathbf{w}) + \tfrac{1}{2\tau^2}\|\mathbf{w}\|^2.$$

Identify λ in the standard formulation $L + \frac{\lambda}{2}\|\mathbf{w}\|^2$ in terms of τ.

(b) The regularised gradient descent update is $\mathbf{w} \leftarrow \mathbf{w} - \eta \nabla_{\mathbf{w}} \tilde{L}$. Expand this to show it equals $\mathbf{w} \leftarrow (1 - \eta\lambda)\mathbf{w} - \eta \nabla_{\mathbf{w}} L$. Why is the factor $(1 - \eta\lambda)$ called "weight decay"?

(c) Let $L(\mathbf{w}) = \frac{1}{2}\|X\mathbf{w} - \mathbf{y}\|^2$ and $\tilde{L}(\mathbf{w}) = L(\mathbf{w}) + \frac{\lambda}{2}\|\mathbf{w}\|^2$. Derive the minimiser $\hat{\mathbf{w}}_\lambda$. With $X^\top X = Q\Lambda Q^\top$, express $\hat{\mathbf{w}}_\lambda$ in the eigenbasis, give the shrinkage factor applied to the direction with eigenvalue λ_j, and state which directions are shrunk most.

4. Dropout: expected output and inverted dropout.

Let $\mathbf{h} \in \mathbb{R}^d$ be the output of a hidden layer at test time. At training time, each coordinate h_j is independently zeroed with probability $1 - p$ (i.e. retained with probability p): $\tilde{h}_j = m_j h_j$, where $m_j \sim \text{Bernoulli}(p)$ independently.

(a) Compute $\mathbb{E}[\tilde{h}_j]$ and $\text{Var}(\tilde{h}_j)$ in terms of p and h_j.

(b) Suppose the next layer computes $a = \mathbf{w}^\top \tilde{\mathbf{h}}$. Compute $\mathbb{E}[a]$. Show that $\mathbb{E}[a] \neq \mathbf{w}^\top \mathbf{h}$ unless $p = 1$, i.e. naive dropout causes a systematic mismatch between training and test outputs.

(c) *Inverted dropout* multiplies the retained activations by $1/p$ at training time: $\tilde{h}_j = m_j h_j / p$. Show that $\mathbb{E}[\tilde{h}_j] = h_j$. Explain why this fixes the train-test mismatch without requiring any rescaling at test time.

(d) Implement inverted dropout for a fixed hidden vector $\mathbf{h}$. Over 10,000 random masks, compare the sample mean of $\tilde{\mathbf{h}}$ with $\mathbf{h}$, and report how $\text{Var}(\tilde{h}_j)$ changes as p decreases from 1.

5. (DS students.) *Optimisers and regularisation on high-dimensional, small-data regression.*

Generate a synthetic dataset with $d = 200$ features, $x_j \sim \mathcal{N}(0, 1)$, using 20 training examples and 100 validation examples, with

$$y = 0.05 + 0.01 \sum_{j=1}^{d} x_j + \epsilon, \qquad \epsilon \sim \mathcal{N}(0, 0.01^2).$$

Fit linear regression with squared loss.

(a) **Optimisers.** Implement SGD, SGD with momentum ($\beta = 0.9$), and Adam from scratch. With no regularisation, train each and plot training and validation loss versus epoch. Which converges fastest, and which shows the widest train–validation gap?

(b) **Weight decay.** Using Adam, for $\lambda \in \{0, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1\}$, train to convergence and plot training and validation loss versus epoch. Report the λ with the lowest validation loss.

(c) **Early stopping.** Using Adam with $\lambda = 0$, train for many epochs, recording training and validation loss every 10 epochs, and return the parameters from the epoch of minimum validation loss. Report that validation loss and compare it with the best λ from part (b).