00103335: Deep Learning and Reinforcement Learning
Homework 2 *Due*: November 4, 2022

Note: Unless otherwise noted, equation and figure numbers refer to those in the DL book.

1. State and prove a convergence theorem for stochastic gradient descent under conditions (8.12) and (8.13). *Hint*: See Robbins and Monro (1951, *Ann. Math. Statist.*, 22, 400–407).

2. In this exercise, we establish a convergence result for gradient descent with Polyak averaging.
   (a) Let $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_T$ be an arbitrary sequence of vectors. For the algorithm with initialization $\boldsymbol{w}^{(1)} = \mathbf{0}$ and update rule
   $$\boldsymbol{w}^{(t+1)} = \boldsymbol{w}^{(t)} - \eta \boldsymbol{v}_t,$$
   show that
   $$\sum_{t=1}^{T} \langle \boldsymbol{w}^{(t)} - \boldsymbol{w}^*, \boldsymbol{v}_t \rangle \le \frac{\|\boldsymbol{w}^*\|^2}{2\eta} + \frac{\eta}{2} \sum_{t=1}^{T} \|\boldsymbol{v}_t\|^2.$$
   (b) Let f be a convex, ρ-Lipschitz function, $\boldsymbol{w}^* = \arg\min_{\|\boldsymbol{w}\| \le B} f(\boldsymbol{w})$, and $\bar{\boldsymbol{w}} = \sum_{t=1}^{T} \boldsymbol{w}^{(t)}/T$. Use part (a) to show that the gradient descent algorithm for minimizing f with $\eta = B/(\rho\sqrt{T})$ satisfies
   $$f(\bar{\boldsymbol{w}}) - f(\boldsymbol{w}^*) \le \frac{B\rho}{\sqrt{T}}.$$

3. Represent the convolution example in Figure 9.1 (3×4 input, 2×2 kernel, "valid" convolution) as matrix multiplication with a doubly block circulant matrix.

4. Consider the pooling example in Figure 9.9. Design a set of filters such that the max pooling unit can learn to be invariant to (a) rotation, and (b) scaling.

5. The Hopfield network is a type of recurrent network consisting of n units with states s_i and update rule
   $$s_i \leftarrow \sigma\left(\sum_{j \ne i} w_{ij} s_j - \theta_i\right),$$
   where $\sigma(x) = 2I(x \ge 0) - 1$, $w_{ij} = w_{ji}$, and $w_{ii} = 0$. The network is updated in an asynchronous manner, so that one unit is randomly selected and updated at each time step. Prove that the network will eventually reach a stable state at a local minimum of the energy function
   $$E(\boldsymbol{s}) = -\frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij} s_i s_j + \sum_{i=1}^{n} \theta_i s_i.$$

6. Design a recurrent neural network to approximate the dynamics of the Lorenz 96 model
   $$\frac{dx_i}{dt} = (x_{i+1} - x_{i-2}) x_{i-1} - x_i + F, \quad i = 1, \ldots, n,$$
   where F is a forcing constant and the indices are cyclic so that $x_{-1} = x_{n-1}$, $x_0 = x_n$, and $x_{n+1} = x_1$.