00103335: Deep Learning and Reinforcement Learning
Homework 1 *Due*: October 12, 2022

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

1. Let $f(x)$ be a nonincreasing and $g(x)$ a nondecreasing function. Give conditions as general as possible under which each of the following assertions holds. Justify your claim and provide a few examples.
   (a) $f(x) + g(x)$ has a U-shape;
   (b) $f(x) + g(x)$ has a double U-shape.

2. Consider the XOR problem described in Section 6.1.
   (a) For a perceptron with MSE loss and linear output, verify that the solution is $\boldsymbol{w} = \mathbf{0}$ and $b = 1/2$.
   (b) Is the problem solvable by a perceptron with cross-entropy loss and sigmoid output? Find the solution in this case.
   (c) Is the solution for the two-layer feedforward network unique? If not, find a solution different from the one given in the book.

3. Prove that the solutions to optimization problems (6.14) and (6.16) are the conditional mean and median of $\boldsymbol{y}$ given $\boldsymbol{x}$, respectively.

4. Numerical differentiation is an alternative approach to back-propagation for computing the gradient. This can be done, for example, by applying the central difference approximation

$$\frac{\partial J}{\partial \theta} = \frac{J(\theta + \varepsilon) - J(\theta - \varepsilon)}{2\varepsilon} + \text{remainder}$$

   to each parameter of the network.
   (a) Show that the remainder term is $O(\varepsilon^2)$.
   (b) Determine the time complexity of this algorithm and compare it with that of back-propagation.

5. It is mentioned in Section 7.5 that, "For some models, the addition of noise with infinitesimal variance at the input of the model is equivalent to imposing a penalty on the norm of the weights." State this formally for a feedforward network with MSE loss and prove your claim.

6. Consider a feedforward network with one hidden layer $\boldsymbol{h}$ and regularized loss (7.48), where $\Omega(\boldsymbol{h}) = \|\boldsymbol{h}\|_1$. Devise a back-propagation algorithm to solve this problem.

7. Prove that the weight scaling inference rule is exact for
   (a) regression networks with conditionally normal outputs;
   (b) deep networks with softmax outputs and linear hidden layers.