

Due Date : February 4th (11pm), 2020Instructions

- For all questions, show your work!
- Use LaTeX and the template we provide when writing your answers. You may reuse most of the notation shorthands, equations and/or tables. See the assignment policy on the course website for more details.
- Submit your answers electronically via Gradescope.

Question 1 (4-4-4). Using the following definition of the derivative and the definition of the Heaviside step function :

$$\frac{d}{dx}f(x) = \lim_{\epsilon \rightarrow 0} \frac{f(x+\epsilon) - f(x)}{\epsilon} \quad H(x) = \begin{cases} 1 & \text{if } x > 0 \\ \frac{1}{2} & \text{if } x = 0 \\ 0 & \text{if } x < 0 \end{cases}$$

1. Show that the derivative of the rectified linear unit $g(x) = \max\{0, x\}$, **wherever it exists**, is equal to the Heaviside step function.
2. Give two alternative definitions of $g(x)$ using $H(x)$.
3. Show that $H(x)$ can be well approximated by the sigmoid function $\sigma(x) = \frac{1}{1+e^{-kx}}$ asymptotically (i.e for large k), where k is a parameter.

Answer 1.

1. a) For the first case, if $x > 0$, when $|\epsilon| < x$, $g(x) = g(x+\epsilon) = x+\epsilon$ and $g(x) = x$, we get ;

$$\lim_{\epsilon \rightarrow 0} \frac{g(x+\epsilon) - g(x)}{\epsilon} = \frac{x+\epsilon - x}{\epsilon} = \frac{\epsilon}{\epsilon} = 1$$

- b) For the second case, if $x = 0$, since the rectified linear unit is only defined to the right or to the left of $x = 0$, we need to check if the left and right-hand limits are equal ;

$$\begin{aligned} \lim_{\epsilon \rightarrow 0^-} \frac{g(x+\epsilon) - g(x)}{\epsilon} &= \frac{0 - 0}{\epsilon} = 0 \\ \lim_{\epsilon \rightarrow 0^+} \frac{g(x+\epsilon) - g(x)}{\epsilon} &= \frac{\epsilon - 0}{\epsilon} = 1 \end{aligned}$$

Since the limits to the left and to the right of $x = 0$ are not equal, $g'(0)$ is undefined and the function g is not differentiable at $x = 0$.

- c) For the last case, if $x < 0$, when $|\epsilon| < -x$, $g(x) = g(x+\epsilon) = 0$, we get ;

$$\lim_{\epsilon \rightarrow 0} \frac{g(x+\epsilon) - g(x)}{\epsilon} = \frac{0 - 0}{\epsilon} = 0$$

2. $g(x) = \max\{0, x\} = xH(x)$ and $\int_{-\infty}^x H(z)dz$

The second alternative definition can be explained by the fact that the integral is the area under the curve, so the integral from $-\infty$ to any point less than 0 is 0 and on the right side of the y axis, the integral to a point x is the area of the rectangle of length x and height 1, which is x.

3. By taking the limit when $k \rightarrow \infty$, we get ;

$$\lim_{k \rightarrow \infty} \sigma(x) = \frac{1}{1 + e^{-kx}} = \begin{cases} 1 & \text{if } x > 0 \text{ (because } e^{-\infty} = 0) \\ \frac{1}{2} & \text{if } x = 0 \text{ (because } e^0 = 1) \\ 0 & \text{if } x < 0 \text{ (because } e^{\infty} = \infty) \end{cases}$$

Question 2 (3-3-3-3). Recall the definition of the softmax function : $S(\mathbf{x})_i = e^{x_i} / \sum_j e^{x_j}$.

1. Show that softmax is translation-invariant, that is : $S(\mathbf{x} + c) = S(\mathbf{x})$, where c is a scalar constant.
2. Show that softmax is not invariant under scalar multiplication. Let $S_c(\mathbf{x}) = S(c\mathbf{x})$ where $c \geq 0$. What are the effects of taking c to be 0 and arbitrarily large ?
3. Let \mathbf{x} be a 2-dimensional vector. One can represent a 2-class categorical probability using softmax $S(\mathbf{x})$. Show that $S(\mathbf{x})$ can be reparameterized using sigmoid function, i.e. $S(\mathbf{x}) = [\sigma(z), 1 - \sigma(z)]^\top$ where z is a scalar function of \mathbf{x} .
4. Let \mathbf{x} be a K -dimensional vector ($K \geq 2$). Show that $S(\mathbf{x})$ can be represented using $K - 1$ parameters, i.e. $S(\mathbf{x}) = S([0, y_1, y_2, \dots, y_{K-1}]^\top)$ where y_i is a scalar function of \mathbf{x} for $i \in \{1, \dots, K - 1\}$.

Answer 2.

1.

For $i \in \{1, \dots, K\}$,

$$\begin{aligned} \text{softmax}(\mathbf{x} + c)_i &= \frac{e^{x_i + c}}{\sum_{j=1}^K e^{x_j + c}} \\ &= \frac{e^{x_i} e^c}{\sum_{j=1}^K e^{x_j} e^c} \\ &= \frac{e^{x_i} e^c}{e^c \sum_{j=1}^K e^{x_j}} \\ &= \frac{e^{x_i}}{\sum_{j=1}^K e^{x_j}} \\ &= \text{softmax}(\mathbf{x})_i \end{aligned}$$

2. To prove that $S(\mathbf{x})$ is not invariant under scalar multiplication we need to provide an example where $S(c\mathbf{x}) \neq S(\mathbf{x})$, with $c \geq 0$.

Consider a 2-dimensional vector $\mathbf{x} = [0, 1]^T$ and $c = 2$. We get $S(c\mathbf{x}) = S([0, 2]^T) = [\frac{1}{1+e^2}, \frac{e^2}{1+e^2}]^T$ and $S(\mathbf{x}) = S([0, 1]^T) = [\frac{1}{1+e}, \frac{e}{1+e}]^T$. Therefore,

$$S(c\mathbf{x}) \neq S(\mathbf{x})$$

When $c = 0$, $e^{cx_i} = e^0 = 1$;

$$S(c\mathbf{x})_i = \frac{1}{\sum_1^n 1} = \frac{1}{n}$$

Meaning that if the element value of $S(c\mathbf{x})$ reflects the probability of several events, all events will have equal probability.

When c is arbitrarily large, $c \rightarrow \infty \Rightarrow S(c\mathbf{x})_i = \lim_{c \rightarrow \infty^+} \frac{(e^{x_i})^c}{\sum_j (e^{x_j})^c} = \lim_{c \rightarrow \infty^+} \frac{1}{\sum_j (\frac{e^{x_j}}{e^{x_i}})^c}$

If $x_i = x_j$, $\lim_{c \rightarrow \infty^+} (\frac{e^{x_j}}{e^{x_i}})^c = 1$.

If $x_i > x_j$, $\lim_{c \rightarrow \infty^+} (\frac{e^{x_j}}{e^{x_i}})^c = 0$.

If $x_i < x_j$, $\lim_{c \rightarrow \infty^+} (\frac{e^{x_j}}{e^{x_i}})^c = \infty^+$.

To resume, if x_i is the maximum of all x_j s, $\lim_{c \rightarrow \infty^+} \frac{1}{\sum_j (\frac{e^{x_j}}{e^{x_i}})^c} = \frac{1}{0+\dots+1+\dots+0} = 1$.

For any other x_i , $\lim_{c \rightarrow \infty^+} \frac{1}{\sum_j (\frac{e^{x_j}}{e^{x_i}})^c} = \frac{1}{\infty+\infty+\dots+1+\dots+\infty} = 0$.

3. With the sigmoid function, $\sigma(z) = \frac{1}{(1+e^{-z})}$ and the general form of the k-class softmax; $\frac{e^{x_i}}{\sum_{i=1}^k e^{x_i}}$, we can split up the softmax is a two-class case.

$$\text{softmax}(x) = \left(\frac{e^{x_1}}{e^{x_1} + e^{x_2}} + \frac{e^{x_2}}{e^{x_1} + e^{x_2}} \right)$$

For the x_1 case;

$$\text{softmax}(x_1) = \frac{e^{x_1}}{e^{x_1} + e^{x_2}} = \frac{1}{1 + e^{x_2-x_1}} = \frac{1}{1 + e^{-(x_1-x_2)}} = \frac{1}{(1 + e^{-z})}$$

Thus, if $z = x_1 - x_2$, $\sigma(z) = \text{softmax}(x_1)$. We can compute a similar equality for $\text{softmax}(x_2)$;

$$\begin{aligned} \text{softmax}(x_2) &= \frac{e^{x_2}}{e^{x_1} + e^{x_2}} \\ &= \frac{1}{e^{x_1-x_2} + 1} \\ &= \frac{1 + e^{x_1-x_2} - e^{x_1-x_2}}{e^{x_1-x_2} + 1} \\ &= 1 - \frac{e^{x_1-x_2}}{e^{x_1-x_2} + 1} \\ &= 1 - \frac{1}{1 + e^{-(x_1-x_2)}} \\ &= 1 - \frac{1}{1 + e^{-z}} \end{aligned}$$

Therefore, $1 - \sigma(z) = \text{softmax}(x_2)$ and $(\sigma(z), 1 - \sigma(z)) = (\text{softmax}(x_1), \text{softmax}(x_2))$

4. For $i \in \{1, \dots, K-1\}$, let constant $c = -\mathbf{x}_1$, and $y_i = x_{i+1} - x_1$

As we showed earlier, $S(\mathbf{x})$ is translation invariant, i.e, $S(\mathbf{x} + c) = S(\mathbf{x})$. Thus, we have ;

$$\begin{aligned} S(x) &= S(x + c) \\ &= S(x - x_1) \\ &= S(x_1 - x_1, x_2 - x_1, x_3 - x_1, \dots, x_k - x_1)^T) \\ &= S([0, y_1, \dots, y_{k-1}]^T) \end{aligned}$$

Question 3 (16). Consider a 2-layer neural network $y : \mathbb{R}^D \rightarrow \mathbb{R}^K$ of the form :

$$y(x, \Theta, \sigma)_k = \sum_{j=1}^M \omega_{kj}^{(2)} \sigma \left(\sum_{i=1}^D \omega_{ji}^{(1)} x_i + \omega_{j0}^{(1)} \right) + \omega_{k0}^{(2)}$$

for $1 \leq k \leq K$, with parameters $\Theta = (\omega^{(1)}, \omega^{(2)})$ and logistic sigmoid activation function σ . Show that there exists an equivalent network of the same form, with parameters $\Theta' = (\tilde{\omega}^{(1)}, \tilde{\omega}^{(2)})$ and tanh activation function, such that $y(x, \Theta', \tanh) = y(x, \Theta, \sigma)$ for all $x \in \mathbb{R}^D$, and express Θ' as a function of Θ .

Answer 3.

Recall that $\sigma(x) = \frac{1}{1+e^{-x}}$ and that $\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$. Let's begin by showing that $\sigma(x)$ is a function of $\tanh(x)$;

$$\begin{aligned} \sigma(x) &= \frac{1}{1 + e^{-x}} \\ &= \frac{1}{1 + e^{\frac{-1}{2}x - \frac{1}{2}x}} \\ &= \frac{1}{1 + \frac{e^{\frac{-1}{2}x}}{e^{\frac{1}{2}x}}} \\ &= \frac{e^{\frac{1}{2}x}}{e^{\frac{1}{2}x} + e^{\frac{-1}{2}x}} \\ &= \frac{1}{2} \left(\frac{2e^{\frac{1}{2}x}}{e^{\frac{1}{2}x} + e^{\frac{-1}{2}x}} \right) \\ &= \frac{1}{2} \left(\frac{e^{\frac{1}{2}x} - e^{\frac{-1}{2}x} + e^{\frac{1}{2}x} + e^{\frac{-1}{2}x}}{e^{\frac{1}{2}x} + e^{\frac{-1}{2}x}} \right) \\ &= \frac{1}{2} \left(\frac{e^{\frac{1}{2}x} - e^{\frac{-1}{2}x}}{e^{\frac{1}{2}x} + e^{\frac{-1}{2}x}} + 1 \right) \\ &= \frac{1}{2} \left(\tanh\left(\frac{1}{2}x\right) + 1 \right) \end{aligned}$$

Which, for the neural network, gives us ;

$$\begin{aligned}
y(x, \Theta, \sigma)_k &= \sum_{j=1}^M \omega_{kj}^{(2)} \frac{1}{2} (\tanh(\frac{1}{2} (\sum_{i=1}^D \omega_{ji}^{(1)} x_i + \omega_{j0}^{(1)})) + 1) + \omega_{k0}^{(2)} \\
&= \sum_{j=1}^M \frac{\omega_{kj}^{(2)}}{2} (\tanh(\sum_{i=1}^D \frac{\omega_{ji}^{(1)}}{2} x_i + \frac{\omega_{j0}^{(1)}}{2}) + 1) + \omega_{k0}^{(2)} \\
&= \sum_{j=1}^M \frac{\omega_{kj}^{(2)}}{2} \tanh(\sum_{i=1}^D \frac{\omega_{ji}^{(1)}}{2} x_i + \frac{\omega_{j0}^{(1)}}{2}) + \sum_{j=1}^M \frac{\omega_{kj}^{(2)}}{2} + \omega_{k0}^{(2)} \\
&= \sum_{j=1}^M \tilde{\omega}_{kj}^{(2)} \tanh(\sum_{i=1}^D \tilde{\omega}_{ji}^{(1)} x_i + \tilde{\omega}_{j0}^{(1)}) + \tilde{\omega}_{k0}^{(2)} \\
&= y(x, \Theta', \tanh)_k
\end{aligned}$$

Thus, there exists an equivalent network such that $y = (x, \Theta, \tanh) = y(x, \Theta', \sigma)$ for all $x \in \mathbb{R}^D$.
 $\Theta' = (\tilde{\omega}^{(1)}, (\tilde{\omega}_{k0}^{(2)}, \tilde{\omega}_{k1}^{(2)}, \dots, \tilde{\omega}_{kM}^{(2)})) = (\frac{\omega^{(1)}}{2}, (\sum_{j=1}^M \frac{\omega_{kj}^{(2)}}{2} + \omega_{k0}^{(2)}, \frac{\omega_{k1}}{2}, \dots, \frac{\omega_{kM}}{2}))$

Question 4 (5-5). Fundamentally, back-propagation is just a special case of reverse-mode Automatic Differentiation (AD), applied to a neural network. Based on the “three-part” notation shown in Table and , represent the evaluation trace and derivative (adjoint) trace of the following examples. In the last columns of your solution, numerically evaluate the value up to 4 decimal places.

1. Forward AD, with $y = f(x_1, x_2) = 1/(x_1 + x_2) + x_2^2 + \cos(x_1)$ at $(x_1, x_2) = (3, 6)$ and setting $\dot{x}_1 = 1$ to compute $\partial y / \partial x_1$.
2. Reverse AD, with $y = f(x_1, x_2) = 1/(x_1 + x_2) + x_2^2 + \cos(x_1)$ at $(x_1, x_2) = (3, 6)$. Setting $\bar{y} = 1$, $\partial y / \partial x_1$ and $\partial y / \partial x_2$ can be computed together.

Answer 4. Reuse the tables to prepare your answer.

TABLE 1 – Forward AD example, with $y = f(x_1, x_2) = \frac{1}{(x_1 + x_2)} + x_2^2 + \cos(x_1)$ at $(x_1, x_2) = (3, 6)$ and setting $\dot{x}_1 = 1$ to compute $\partial y / \partial x_1$.

Forward evaluation trace			Forward derivative trace		
v_{-1}	$= x_1$	$= 3$	$= \dot{v}_{-1}$	\dot{x}_1	$= 1$
v_0	$= x_2$	$= 6$	$= \dot{v}_0$	\dot{x}_2	$= 0$
v_1	$= v_{-1} + v_0$	$= 3 + 6$	\dot{v}_1	$= \dot{v}_{-1} + 0$	$= 1$
v_2	$= \frac{1}{v_1}$	$= 2\frac{1}{9}$	\dot{v}_2	$= -v_1^{-2}$	$= \frac{-1}{9^2}$
\Downarrow v_3	$= v_0^2$	$= 6^2$	\Downarrow \dot{v}_3	$= 2v_0x_0$	$= 2 \times 6 \times 0$
v_4	$= \cos(v_{-1})$	$= \cos(3)$	\dot{v}_4	$= -\sin(v_{-1})x_{-1}$	$= -\sin(3) \times 1$
v_5	$= v_2 + v_3$	$= 0.1111 + 36$	\dot{v}_5	$= \dot{v}_2 + \dot{v}_3$	$= -0.0123$
v_6	$= v_5 + v_4$	$= 36.1111 - 0.9900$	\dot{v}_6	$= \dot{v}_5 + \dot{v}_4$	$= -0.0123 - 0.1411$
y	$= v_6$	$= 35.1211$	$= \dot{y}$	\dot{v}_6	$= -0.1534$

TABLE 2 – Reverse AD example, with $y = f(x_1, x_2) = \frac{1}{(x_1+x_2)} + x_2^2 + \cos(x_1)$ at $(x_1, x_2) = (3, 6)$. Setting $\bar{y} = 1$, $\partial y / \partial x_1$ and $\partial y / \partial x_2$ are computed in one reverse sweep.

Forward evaluation trace			Reverse adjoint trace		
v_{-1}	$= x_1$	$= 3$	\bar{x}_1	$= \bar{v}_{-1}$	$= -0.1534$
v_0	$= x_2$	$= 6$	\bar{x}_2	$= \bar{v}_0$	$= -0.0123$
v_1	$= v_{-1} + v_0$	$= 3 + 6$	\bar{v}_0	$= \bar{v}_0 + \bar{v}_1 \frac{\partial v_1}{\partial v_0}$	$= \frac{-1}{9^2}$
v_2	$= \frac{1}{v_1^2}$	$= 2\frac{1}{9}$	\bar{v}_{-1}	$= \bar{v}_{-1} + \bar{v}_1 \frac{\partial v_1}{\partial v_{-1}}$	$= -\sin(3) - \frac{1}{9^2}$
v_3	$= v_0^2$	$= 6^2$	\bar{v}_1	$= \bar{v}_2 \frac{\partial v_2}{\partial v_1}$	$= \frac{-1}{9^2}$
v_4	$= \cos(v_{-1})$	$= \cos(3)$	\bar{v}_0	$= \bar{v}_3 \frac{\partial v_3}{\partial v_0}$	$= 1 \times 2 \times 6 \times 0$
v_5	$= v_2 + v_3$	$= 0.1111 + 36$	\bar{v}_{-1}	$= \bar{v}_4 \frac{\partial v_4}{\partial v_{-1}}$	$= 1 \times -\sin(3) \times 1$
v_6	$= v_5 + v_4$	$= 36.1111 - 0.9900$	\bar{v}_2	$= \bar{v}_5 \frac{\partial v_5}{\partial v_2}$	$= 1$
y	$= v_6$	$= 35.1211$	\bar{v}_3	$= \bar{v}_5 \frac{\partial v_5}{\partial v_3}$	$= 1$
			\bar{v}_4	$= \bar{v}_6 \frac{\partial v_6}{\partial v_4}$	$= 1$
			\bar{v}_5	$= \bar{v}_6 \frac{\partial v_6}{\partial v_5}$	$= 1$
			\bar{v}_6	$= \bar{y}$	$= 1$

Question 5 (6). Compute the *full*, *valid*, and *same* convolution (with kernel flipping) for the following 1D matrices : $[1, 2, 3, 4] * [1, 0, 2]$

Answer 5. Full : $[,]$; Valid : $[,]$; Same : $[,]$.

Full convolution : padding the matrix with zeros in such a way that on convoluting with the kernel, the elements of the original matrix get visited k times (k being the size of the kernel)

\mathbf{X} being the input, $\mathbf{X} = [0, 0, 1, 2, 3, 4, 0, 0]$.

Thus, $[1, 2, 3, 4] * [1, 0, 2] = [1, 2, 5, 8, 6, 8]$

Valid convolution : convolution of the matrix with the kernel without any padding. $[1, 2, 3, 4] * [1, 0, 2] = [5, 8]$

Same convolution : zero padding the matrix such that the result of the convolution with the given kernel results in a matrix of the same shape as the input. \mathbf{X} being the input, $\mathbf{X} = [0, 1, 2, 3, 4, 0]$

Thus, $[1, 2, 3, 4] * [1, 0, 2] = [2, 5, 8, 6]$

Question 6 (5-5). Consider a convolutional neural network. Assume the input is a colorful image of size 256×256 in the RGB representation. The first layer convolves 64 8×8 kernels with the input, using a stride of 2 and no padding. The second layer downsamples the output of the first layer with a 5×5 non-overlapping max pooling. The third layer convolves 128 4×4 kernels with a stride of 1 and a zero-padding of size 1 on each border.

1. What is the dimensionality (scalar) of the output of the last layer ?
2. Not including the biases, how many parameters are needed for the last layer ?

Answer 6.

1. We can calculate the output shape of a convolutional layer with this specific formula ;

$$o = \frac{(W - F + 2P)}{S} + 1$$

where W , F , P , and S are the input size, kernel size, padding size, and stride size. After the first convolutional layer, the output size is $64 \times 125 \times 125$ ($\frac{256-8}{2} + 1$), then the second layer downsamples the output to $64 \times 25 \times 25$ and finally the output size is $128 \times 24 \times 24$ ($25 - 4 + 2 + 1$).

2. For the last layer, there are $128 \times 64 \times 4 \times 4 = 131072$ needed parameters.

Question 7 (4-4-6). Assume we are given data of size $3 \times 64 \times 64$. In what follows, provide a correct configuration of a convolutional neural network layer that satisfies the specified assumption. Answer with the window size of kernel (k), stride (s), padding (p), and dilation (d , with convention $d = 1$ for no dilation). Use square windows only (e.g. same k for both width and height).

1. The output shape (o) of the first layer is $(64, 32, 32)$.
 - (a) Assume $k = 8$ without dilation.
 - (b) Assume $d = 7$, and $s = 2$.
2. The output shape of the second layer is $(64, 8, 8)$. Assume $p = 0$ and $d = 1$.
 - (a) Specify k and s for pooling with non-overlapping window.
 - (b) What is output shape if $k = 8$ and $s = 4$ instead?
3. The output shape of the last layer is $(128, 4, 4)$.
 - (a) Assume we are not using padding or dilation.
 - (b) Assume $d = 2$, $p = 2$.
 - (c) Assume $p = 1$, $d = 1$.

Answer 7. Fill up the following table,

		i	p	d	k	s	o
1.	(a)	64	3	1	8	2	32
	(b)	64	3	7	2	2	32
2.	(a)	32	0	1	4	4	8
	(b)	32	0	1	8	4	7
3.	(a)	8	0	1	2	2	4
	(b)	8	2	2	3	2	4
	(c)	8	1	1	4	2	4

For this question, the table above was completed using the following equations ;

$$o = \frac{i + 2p - k'}{s} + 1$$

$$k' = k + (k - 1)(d - 1)$$

Where o (output size), i (input size), k' (effective kernel size), k (kernel size), s (stride), d (dilatation) $\in \mathbb{N}$, and p (zero-padding) $\in \mathbb{Z}^{\geq 0}$.

Since all the subquestions were solved with the same approach, I will only demonstrate the procedure for the first subquestion a) ;

$$k' = 8 + (8 - 1)(1 - 1) = 8$$

$$32 = \left\lfloor \frac{64 + 2p - 8}{s} \right\rfloor + 1$$

$$\left\lfloor \frac{56 + 2p}{s} \right\rfloor = 31$$

Thus, one possible solution for s could be 2 and 3 for p.