Homework 1
600.482/682 Deep Learning
Spring 2020

Mou Zhang

February 23, 2020

Due Sunday 2/23 11:59pm.
Please type your answers inline of the LaTeX file
Submit PDF to Gradescope with entry code 9G83Y7

1. In this exercise you are going to derive the well-known sigmoid expression for a Bernoulli distributed (binary) problem. The probability of the "positive" event occurring is p. The probability of the "negative" event occurring is $q = 1 - p$.

   (a) What are the odds o of the "positive" event occurring? Please express the result using p only.

   In statistics, the logit of the probability is the logarithm of the corresponding odds, i.e. $\text{logit}(p) = \log(o)$.

   Solution: The odds o of the "positive" event occurring is $\frac{p}{1-p}$.

   (b) Given $\text{logit}(p) = x$, please derive the inverse function $\text{logit}^{-1}(x)$. Please express the result using x only.

   The inverse function of the logit in (b) is actually the sigmoid function $S(x)$. You may already have noticed that the probability $p = \text{logit}^{-1}(x) = S(x)$. This means that the range of the sigmoid function is the same as the range of a probability, i.e. $(0, 1)$. The domain of the sigmoid function is $(-\infty, \infty)$. Therefore, the sigmoid function maps all real numbers to the interval $(0, 1)$.

   Solution: Since $logit(p) = x$, we have $logit^{-1}(x) = logit^{-1}(logit(p)) = p$. Then

$$\begin{aligned} x &= logit(p) = \log\frac{p}{1-p} \\ &\Rightarrow \frac{p}{1-p} = e^x \\ &\Rightarrow p = e^x - p \cdot e^x \\ &\Rightarrow p \cdot (1 + e^x) = e^x \\ &\Rightarrow p = \frac{e^x}{1+e^x} \\ &\Rightarrow logit^{-1}(x) = \frac{e^x}{1+e^x} \end{aligned}$$

   (c) Now we look into the saturation of the sigmoid function. Calculate the value of the sigmoid function $S(x)$ for $x = \pm 100, \pm 10$, and 0. Round the results to two decimal places.

   Solution: Since Sigmoid function $S(x) = \frac{e^x}{1+e^x} = \frac{1}{1+e^{-x}}$, we have

$$\begin{aligned} S(100) &= \frac{1}{1+e^{-100}} = 1.00 \\ S(-100) &= \frac{1}{1+e^{100}} = 0.00 \end{aligned}$$

$$
\begin{aligned}
S(10) &= \frac{1}{1+e^{-10}} = 0.9999546 = 1.00 \\
S(-10) &= \frac{1}{1+e^{-10}} = 0.0000453978 = 4.53^{-5} = 0.00 \\
S(0) &= \frac{1}{1+e^{0}} = 0.50.
\end{aligned}
$$

(d) Calculate the derivatives of the sigmoid function $S'(x)$ and the value of $S'(x)$ for $x = \pm 100, \pm 10$, and 0. Round the results to two decimal places.

You may have noticed that $S(\pm 100)$ is very close to $S(\pm 10)$; the derivatives at $x = \pm 100$ and $x = \pm 10$ are very close to zero. This is the saturation of the sigmoid function when $|x|$ is large. The saturation brings great difficulty in training deep neural networks. This will reappear in later lectures.

Solution:

$$
\begin{aligned}
S'(x) &= (\frac{1}{1+e^{-x}})' \\
&= \frac{(1)' \cdot (1+e^{-x}) - 1 \cdot (1+e^{-x})'}{(1+e^{-x})^2} \\
&= \frac{e^{-x}}{(1+e^{-x})^2}
\end{aligned}
$$

So

$$
\begin{aligned}
S'(100) &= 3.72 \times 10^{-44} = 0.00 \\
S'(-100) &= 3.72 \times 10^{-44} = 0.00 \\
S'(10) &= 4.54 \times 10^{-5} = 0.00 \\
S'(-10) &= 4.54 \times 10^{-5} = 0.00 \\
S'(0) &= 0.25
\end{aligned}
$$

2. Recall in class, we learned the form of a linear classifier as $f(\boldsymbol{x};\boldsymbol{W}) = \boldsymbol{W}\boldsymbol{x} + \boldsymbol{b}$. We will soon learn, that iteratively updating the weights in negative gradient direction will allow us to slowly move towards an optimal solution. We will call this technique backpropagation. Obviously, computing gradients is an important component of this technique. We will investigate the first derivative of a commonly used loss function: the softmax loss. Here, we consider a multinomial (multiple classes) problem.
Let's first define the notations:

$$\begin{aligned}
\text{input features}: &\quad \boldsymbol{x} \in \mathbb{R}^D. \\
\text{target labels (one-hot encoded)}: &\quad \boldsymbol{y} \in \{0,1\}^K. \\
\text{multinomial linear classifier}: &\quad \boldsymbol{f} = \boldsymbol{W}\boldsymbol{x} + \boldsymbol{b}, \quad \boldsymbol{W} \in \mathbb{R}^{K\times D} \text{ and } \boldsymbol{f}, \boldsymbol{b} \in \mathbb{R}^K \\
\text{e.g., for the k-th classification}: &\quad f_k = \boldsymbol{w}_k^T \boldsymbol{x} + b_k, \text{ corresponding to } y_k, \\
&\quad \text{where } \boldsymbol{w}_k^T \text{ is the k-th row of } \boldsymbol{W}, k \in \{1...K\}
\end{aligned}$$

(a) Please express the softmax loss of logistic regression, $L(\boldsymbol{x},\boldsymbol{W},\boldsymbol{b},\boldsymbol{y})$ using the above notations.
Solution: Assume $y_a = 1$

$$\begin{aligned}
L(\boldsymbol{x},\boldsymbol{W},\boldsymbol{b},\boldsymbol{y}) &= -\log\left(\frac{e^{f_a}}{\sum_j e^{f_j}}\right) \\
&= -\log\left(\frac{e^{f}\cdot y}{\sum_j e^{f_j}}\right) \\
&= -\log\left(\frac{e^{Wx+b}\cdot y}{\sum e^{Wx+b}}\right)
\end{aligned}$$

(b) Please calculate its gradient derivative $\frac{\partial L}{\partial \boldsymbol{w}_k}$.
Solution: Assume $y_a = 1$

$$\begin{aligned}
L(\boldsymbol{x},\boldsymbol{W},\boldsymbol{b},\boldsymbol{y})' &= \left(-\log\left(\frac{e^{Wx+b}\cdot y}{\sum e^{Wx+b}}\right)\right)' \\
&= -\frac{1}{\left(\frac{e^{Wx+b}\cdot y}{\sum e^{Wx+b}}\right)}\cdot\left(\frac{e^{Wx+b}\cdot y}{\sum e^{Wx+b}}\right)' \\
&= -\frac{1}{\left(\frac{e^{Wx+b}\cdot y}{\sum e^{Wx+b}}\right)}\cdot\left(\frac{e^{W_a x+b}}{\sum e^{Wx+b}}\right)' \\
&= -\frac{\sum e^{Wx+b}}{e^{Wx+b}\cdot y}\cdot\frac{(e^{W_a x+b})'\cdot(\sum e^{Wx+b}) - (\sum e^{Wx+b})'\cdot(e^{W_a x+b})}{(\sum e^{Wx+b})^2}
\end{aligned}$$

(1)If $k \neq a$, then

$$\begin{aligned}
L(\boldsymbol{x},\boldsymbol{W},\boldsymbol{b},\boldsymbol{y})' &= -\frac{\sum e^{Wx+b}}{e^{W_a x+b}}\cdot\frac{(e^{W_a x+b})'\cdot(\sum e^{Wx+b}) - (\sum e^{Wx+b})'\cdot(e^{W_a x+b})}{(\sum e^{Wx+b})^2} \\
&= -\frac{\sum e^{Wx+b}}{e^{W_a x+b}}\cdot\frac{0 - (e^{W_k x+b})'\cdot e^{W_a x+b}}{(\sum e^{Wx+b})^2} \\
&= \frac{(e^{W_k x+b})'}{\sum e^{Wx+b}} \\
&= \frac{x\cdot e^{W_k x+b}}{\sum e^{Wx+b}} \\
&= x\cdot\frac{e^{W_k x+b}}{\sum e^{Wx+b}}
\end{aligned}$$

(2)If $k = a$, then

$$L(\boldsymbol{x},\boldsymbol{W},\boldsymbol{b},\boldsymbol{y})' = -\frac{\sum e^{Wx+b}}{e^{W_k x+b}}\cdot\frac{(e^{W_k x+b})'\cdot(\sum e^{Wx+b}) - (\sum e^{Wx+b})'\cdot(e^{W_k x+b})}{(\sum e^{Wx+b})^2}$$

$$
\begin{aligned}
&= -\frac{\sum e^{Wx+b}}{e^{W_k x+b}} \cdot \frac{(x \cdot e^{W_k x+b}) \cdot (\sum e^{Wx+b}) - (x \cdot e^{W_k x+b}) \cdot (e^{W_k x+b})}{(\sum e^{Wx+b})^2} \\
&= -\frac{x \cdot (\sum e^{Wx+b} - e^{W_k x+b})}{\sum e^{Wx+b}} \\
&= -x + x \cdot \frac{e^{W_k x+b}}{\sum e^{Wx+b}}
\end{aligned}
$$

3. In class, we briefly touch upon the Kullback-Leibler (KL) divergence as another loss function to quantify agreement between two distributions p and q. In machine learning scenarios, one of these two distributions will be determine by our training data, while the other is being generated as output of our model. The goal of training our model is to match these two distributions as well as possible. KL divergence is asymmetric, so that assigning these distributions to p and q will matter. Here, you will investigate this difference by calculating the gradient. The KL divergence is defined as

$$\mathrm{KL}(p||q) = \sum_d p(d) \log \left(\frac{p(d)}{q(d)} \right)$$

(a) Show that KL divergence is asymmetric using the following example. We define a discrete random variable X. Now consider the case that we have two sampling distributions $P(x)$ and $Q(x)$, which we present as two vectors that express the frequency of event x:

$$\begin{aligned} P(x) &= [1,\ 6,\ 12,\ 5,\ 2,\ 8,\ 12,\ 4] \\ Q(x) &= [1,\ 3,\ 6,\ 8,\ 15,\ 10,\ 5,\ 2] \end{aligned}$$

Please compute 1) the probability distribution, $p(x)$ and $q(x)$ (hint: calculate the normalization); and 2) both directions of KL divergence, $\mathbf{KL}(p||q)$ and $\mathbf{KL}(q||p)$.

Solution:

1)

$$\begin{aligned} p(x) &= [0.02,\ 0.12,\ 0.24,\ 0.1,\ 0.04,\ 0.16,\ 0.24,\ 0.08] \\ q(x) &= [0.02,\ 0.06,\ 0.12,\ 0.16,\ 0.3,\ 0.2,\ 0.1,\ 0.04] \end{aligned}$$

2)

$$\begin{aligned} \mathbf{KL}(p||q) &= \sum_d p(d) \log \left(\frac{p(d)}{q(d)} \right) = 0.351797804 = 0.35 \\ \mathbf{KL}(q||p) &= \sum_d q(d) \log \left(\frac{q(d)}{p(d)} \right) = 0.484260943 = 0.48 \end{aligned}$$

(b) Next, we try to optimize the weights $\boldsymbol{W}$ of a model in an attempt to minimize KL divergence. As a consequence, $q = q_{\boldsymbol{W}}$ now depends on the weights. Please express $\mathbf{KL}(q_{\boldsymbol{W}}||p)$ and $\mathbf{KL}(p||q_{\boldsymbol{W}})$ as optimization objective functions. Can you tell which direction is easier for computation? To find out, please look back at the original expression of $\mathbf{KL}(q_{\boldsymbol{W}}||p)$ and $\mathbf{KL}(p||q_{\boldsymbol{W}})$ and see which terms can be grouped to be a constant. This constant can be thus cancelled out when calculating the gradient. Then, please also calculate the gradient of $\mathbf{KL}(q_{\boldsymbol{W}}||p)$ and $\mathbf{KL}(p||q_{\boldsymbol{W}})$ w.r.t. $q_{\boldsymbol{W}}(d)$, the d-th element of $q_{\boldsymbol{W}}$.

Solution:

We know that $p, q_{\boldsymbol{W}}$ are probabilty distributions. $p = \overline{p}/Z_p$, where Z_p is the normalization constant. Similarly for q.

1)The easy direction is $\mathbf{KL}(q_{\boldsymbol{W}}||p)$

$$\begin{aligned} \mathbf{KL}(q_{\boldsymbol{W}}||p) &= \sum_d q_{\boldsymbol{W}}(d) \log \left(\frac{q_{\boldsymbol{W}}(d)}{p(d)} \right) \\ &= \sum_d q_{\boldsymbol{W}}(d)(\log q_{\boldsymbol{W}}(d) - \log p(d)) \\ &= \sum_d q_{\boldsymbol{W}}(d) \log q_{\boldsymbol{W}}(d) - \sum_d q_{\boldsymbol{W}}(d) \log p(d) \end{aligned}$$

The left part is entropy part and the right part is the cross-entropy part.

Let's look at the right part:

$$\sum_d q_{\boldsymbol{W}}(d) \log p(d) = \sum_d q_{\boldsymbol{W}}(d) \log(\frac{\overline{p}(d)}{Z_p})$$

$$
\begin{aligned}
&= \sum_d q_{\boldsymbol{W}}(d) \log \overline{p}(d) - \sum_d q_{\boldsymbol{W}}(d) \log Z_p \\
&= \sum_d q_{\boldsymbol{W}}(d) \log \overline{p}(d) - \log Z_p
\end{aligned}
$$

Since Z_p is a constant, we can drop it when optimizing. This leaves the optimize problem as:

$$
\arg\min_{\boldsymbol{W}} \mathbf{KL}(q_{\boldsymbol{W}}||p) = \arg\min_{\boldsymbol{W}} \sum_d q_{\boldsymbol{W}}(d) \log q_{\boldsymbol{W}}(d) - \sum_d q_{\boldsymbol{W}}(d) \log \overline{p}(d)
$$

Let's calculate the gradient.

$$
\begin{aligned}
&\nabla[\sum_d q_{\boldsymbol{W}}(d) \log q_{\boldsymbol{W}}(d) - \sum_d q_{\boldsymbol{W}}(d) \log \overline{p}(d)] \\
&= \sum_d \nabla[q_{\boldsymbol{W}}(d) \log q_{\boldsymbol{W}}(d)] - \sum_d \nabla[q_{\boldsymbol{W}}(d)] \log \overline{p}(d) \\
&= \sum_d \nabla[q_{\boldsymbol{W}}(d)](1 + \log q_{\boldsymbol{W}}(d)) - \sum_d \nabla[q_{\boldsymbol{W}}(d)] \log \overline{p}(d) \\
&= \sum_d \nabla[q_{\boldsymbol{W}}(d)](1 + \log q_{\boldsymbol{W}}(d) - \log \overline{p}(d)) \\
&= \sum_d \nabla[q_{\boldsymbol{W}}(d)](\log q_{\boldsymbol{W}}(d) - \log \overline{p}(d))
\end{aligned}
$$

2)The harder direction is $\mathbf{KL}(p||q_{\boldsymbol{W}})$

$$
\begin{aligned}
\mathbf{KL}(p||q_{\boldsymbol{W}}) &= \sum_d p(d) \log \left(\frac{p(d)}{q_{\boldsymbol{W}}(d)}\right) \\
&= \sum_d p(d)(\log p(d) - \log q_{\boldsymbol{W}}(d)) \\
&= \sum_d p(d) \log p(d) - \sum_d p(d) \log q_{\boldsymbol{W}}(d))
\end{aligned}
$$

Since it's obvious that the left part does not contribute to the optimization, we only need to calculate the right part.

$$
\begin{aligned}
\sum_d p(d) \log q_{\boldsymbol{W}}(d)) &= \frac{1}{Z_p} \sum_d \overline{p}(d) \log \frac{\overline{q}_{\boldsymbol{W}}}{Z_q} \\
&= \frac{1}{Z_p} \sum_d \overline{p}(d)(\log \overline{q}_{\boldsymbol{W}} - \log Z_q) \\
&= (\frac{1}{Z_p} \sum_d \overline{p}(d) \log \overline{q}_{\boldsymbol{W}}) - (\frac{1}{Z_p} \sum_d \overline{p}(d) \log Z_q) \\
&= (\frac{1}{Z_p} \sum_d \overline{p}(d) \log \overline{q}_{\boldsymbol{W}}) - (\log Z_q)(\frac{1}{Z_p} \sum_d \overline{p}(d)) \\
&= (\frac{1}{Z_p} \sum_d \overline{p}(d) \log \overline{q}_{\boldsymbol{W}}) - \log Z_q
\end{aligned}
$$

This leaves the optimize problem as:

$$
\arg\min_{\boldsymbol{W}} \mathbf{KL}(p||q_{\boldsymbol{W}}) = \arg\min_{\boldsymbol{W}} [(\frac{1}{Z_p} \sum_d \overline{p}(d) \log \overline{q}_{\boldsymbol{W}}) - \log Z_q]
$$

Let's calculate the gradient.

$$
\begin{aligned}
&\nabla[(\frac{1}{Z_p} \sum_d \overline{p}(d) \log \overline{q}_{\boldsymbol{W}}) - \log Z_q] \\
&= \frac{1}{Z_p} \sum_d \overline{p}(d) \nabla[\log \overline{q}_{\boldsymbol{W}}] - \nabla \log Z_q
\end{aligned}
$$

3)The first direction $\mathbf{KL}(q_{\boldsymbol{W}}||p)$ is more convenient because we don't need to normalize p. For most models, we can't compute Z_p because p is presumed to be a complex model. If we can not even calculate Z_p, it is not likely to compute more complex thing, for example, $\sum_d \overline{p}(d) \log \overline{q}_{\boldsymbol{W}}$.

4. In this problem, you are provided an opportunity to perform hands-on calculation of the SVM loss and softmax loss we learned in class.
   We define a linear classifier:
$$f(\boldsymbol{x}, \boldsymbol{W}) = \boldsymbol{W}\boldsymbol{x} + \boldsymbol{b}$$
   and are given a data sample:
$$\boldsymbol{x}_i = \begin{bmatrix} -15 \\ 22 \\ -44 \\ 56 \end{bmatrix}, \; y_i = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}.$$
   Assume that the weights of our model are given by
$$\boldsymbol{W} = \begin{bmatrix} 0.01, \; -0.05, \; 0.1, \; 0.05 \\ 0.7, \; 0.2, \; 0.05, \; 0.16 \\ 0.0, \; -0.45, \; -0.2, \; 0.03 \end{bmatrix}, \boldsymbol{b} = \begin{bmatrix} 0.0 \\ 0.2 \\ -0.3 \end{bmatrix}.$$
   Please calculate 1) SVM loss (hinge loss) and 2) softmax loss (cross-entropy loss) of this sample. Use the natural log.

   Solution:
$$f = Wx + b = \begin{bmatrix} 0.01, \; -0.05, \; 0.1, \; 0.05 \\ 0.7, \; 0.2, \; 0.05, \; 0.16 \\ 0.0, \; -0.45, \; -0.2, \; 0.03 \end{bmatrix} \cdot \begin{bmatrix} -15 \\ 22 \\ -44 \\ 56 \end{bmatrix} + \begin{bmatrix} 0.0 \\ 0.2 \\ -0.3 \end{bmatrix} = \begin{bmatrix} -2.85 \\ 0.86 \\ 0.28 \end{bmatrix}$$
   1)SVM loss(hinge loss) is
$$\begin{aligned} L_{SVM} &= \sum_{j \neq y_i} max(0, f_j - f_{y_i} + 1) \\ &= max(0, -2.85 - 0.28 + 1) + max(0, 0.86 - 0.28 + 1) \\ &= 1.58 \end{aligned}$$
   2)softmax loss is
$$\begin{aligned} L_{softmaxloss} &= -\log\left(\frac{e^{f_{y_i}}}{\sum_j e^{f_j}}\right) \\ &= -\log\left(\frac{1.32313}{0.05784 + 2.36316 + 1.32313}\right) \\ &= 1.0401905 = 1.04 \end{aligned}$$