Machine Learning Assignment

BA-1-1 Wenbin Zhou （周文斌）

August 8, 2021

1 Univariate Gaussian

Question a

Solution: For $X \sim \mathcal{N}(1, 2)$, consider the nromalized random variable:

$$Y = \frac{X-1}{\sqrt{2}} \sim \mathcal{N}(1, 2).$$

The original problem is equivalent to finding the probability of $Y \in [-\frac{\sqrt{2}}{4}, \frac{\sqrt{2}}{2}]$. By looking up the standard Gaussian distribution table, we can obtain that the answer is approximately 0.3979.

Question b

Solution: It is obvious that $\mathcal{N}(x; \mu, \sigma^2)$ reaches its maximum when $x = \mu$, and the value is:

$$\mathcal{N}(\mu; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma}} \exp\left\{-\frac{(\mu-\mu)^2}{2\sigma^2}\right\} = \frac{1}{\sqrt{2\pi\sigma}}.$$

Question c

Solution: By independence, the multivariate jpdf is the product of the pdf of each $x_i, \ i = 1, \dots, n$. We yield that:

$$f(x_1, \dots, x_n) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{(x_i-\mu)^2}{2\sigma^2}\right\} = \frac{1}{(2\pi\sigma^2)^{\frac{n}{2}}} \exp\left\{-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(x_i-\mu)^2\right\}.$$

Question d (bonus)

Solution:We denote $\boldsymbol{X} := (x_1, \dots, x_n)$ and $\boldsymbol{Y} := (y_1, \dots, y_n)$. For arbitrary $i = 1, \dots, n$, the random variable $x_i + y_i$ is still Gaussian, since it is the sum of two independent Gaussians. Therefore $\boldsymbol{X} + \boldsymbol{Y})$ is a vector with its elements made up of Gaussians, so it must be a **multivariate Gaussian**. By simple calculation, we can obtain the parameters:

$$\mathbb{E}(\boldsymbol{X}+\boldsymbol{Y}) = \boldsymbol{0}, \qquad \sum_{\boldsymbol{X}+\boldsymbol{Y}} = \sum_{\boldsymbol{X}} + \sum_{\boldsymbol{Y}},$$

So $\boldsymbol{X}+\boldsymbol{Y} \sim \mathcal{N}(\mathbf{0}, \sum_{\boldsymbol{X}} + \sum_{\boldsymbol{Y}})$. Similarly, we denote $a\boldsymbol{X} := (ax_1, \ldots, ax_n)$, it is a multivariate Gaussain and the parameters are:

$$\mathbb{E}(a\boldsymbol{X}) = \mathbf{0}, \qquad \sum_{a\boldsymbol{X}} = a^2 \sum_{\boldsymbol{X}},$$

So $a\boldsymbol{X} \sim \mathcal{N}(\mathbf{0}, a^2 \sum_{\boldsymbol{X}})$.

2 Perceptron Performance

Question a

Solution: The answer is no. A counterexample can be constructed by consider initializing $\theta_1 = (0, 1)$, we eventually yield $(0, 1)$. However if we initialize $\theta_2 = (0, -1)$, we eventually yield $(0.2, 5)$. The two results are clearly different.

Question b

Solution: Their performance on the test set may vary. Since it is possible for them to obtain different weight vectors, and they are not guaranteed to be optimal. For instance, the errors on the test sets may be different for them.

Question c

Solution: At the end of the procedure, the weight vector is $(-3, 2)$, and the bias (a.k.a. offset) is 2.

3 Logistic-regression, cross-entropy, training

Question a

Solution: With x, w are scalar and $N = 1$ given, we denote the standard sigmoid logistic function by

$$h(x, w) := \sigma(s) = \frac{e^s}{1 + e^s}, \qquad s := wx$$

. The cross entropy can be re-written as

$$E(w) = -y \log h(x, w) + (y - 1)\log(1 - h(x, w))$$

by the chain rule of derivation, we know that

$$\frac{\partial E(w)}{\partial w} = \frac{\partial E}{\partial h}\frac{\partial h}{\partial s}\frac{\partial s}{\partial w}.$$

Proceed the following calculations:

$$\frac{\partial E}{\partial h} = \frac{-y}{h} + \frac{1-y}{1-h},$$

$$\frac{\partial h}{\partial s} = (\frac{e^s}{1+e^s})' = \frac{e^s(1+e^s) - e^{2s}}{(1+e^s)^2} = \frac{e^s}{1+e^s} \cdot \frac{1}{1+e^s} = \sigma(s)\left(1-\sigma(s)\right),$$

$$\frac{\partial s}{\partial w} = x.$$

plug these in and we will get the desired result (which seems quite elegant)

$$\frac{\partial E(w)}{\partial w} = (\sigma(wx) - y)x.$$

Question b

Solution: In Question a we've already found the gradient of the loss function when $N = 1$. We shall use that directly in the algorithm below.

Algorithm 1 Weight updating rule when using SGD

```
Input: The training dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$
Output:
Initialize weights $w_0$, Set t=0
for $t = 0, 1, \ldots$ do
    Pick $i \in \{1, 2, \ldots, N\}$
    Obtain stochastic gradient $g_t$ of $(x_i, y_i) \in \mathcal{D}$. $g_t$ is given by the answer of Question a, which is $g_t = (\sigma(w^t) - y_i)x_i$
    Update $w^{t+1} \leftarrow w^t - \eta_t g_t$
    if $w^{t+1} - w^t < \epsilon$ then
        Break
    end if
end for
return $w^{t+1}$
```

4 ReLU Backpropagation; Single output network

Question a

Solution:

$$z_1 = \frac{1}{10}x = \frac{1}{10} \cdot 2 = \frac{1}{5},$$

$$a_1 = ReLU(\frac{1}{5}) = \frac{1}{5}.$$

Question b

Solution:

$$z_2 = -1 \cdot \frac{1}{5} - 0.2 = -0.4,$$

$$y = \sigma(-0.4) = \frac{e^{-0.4}}{1+e^{-0.4}}$$

Question c

Solution:

$$E = \frac{1}{2}\left(\frac{e^{-0.4}}{1+e^{-0.4}} - 1\right)^2 = \frac{1}{2(1+e^{-0.4})^2}$$

Question d

Solution:

$$\frac{\partial E}{\partial w_1} = \frac{\partial E}{\partial y}\frac{\partial y}{\partial z_2}\frac{\partial z_2}{\partial a_1}\frac{\partial a_1}{\partial z_1}\frac{\partial z_1}{\partial w_1}$$

Question e

Solution:

$$\frac{\partial E}{\partial y} = y - t$$

$$\frac{\partial y}{\partial z_2} = y(1-y)$$

$$\frac{\partial z_2}{\partial a_1} = w_2$$

$$\frac{\partial a_1}{\partial z_1} = \begin{cases} 1 & z_1 > 0 \\ 0 & z_1 < 0 \end{cases}$$

$$\frac{\partial z_1}{\partial w_1} = x$$

Question f

Solution:

$$\frac{\partial E}{\partial w_1} = -\frac{1}{1+e^{-0.4}} \cdot \frac{e^{-0.4}}{1+e^{-0.4}}\left(\frac{1}{1+e^{-0.4}}\right) \cdot -1 \cdot 1 \cdot 2 = \frac{2e^{-0.4}}{(1+e^{-0.4})^3}$$

5 Convolutions and Fully connected Networks

Question a

Solution: There are 3 parameters in the convolutional layer, which are w_1, w_2, w_3.

Question b

Solution: Using simple linear algebra knowledge, we can conclude that it can be written as a 6 × 6 matrix

$$\mathbf{W_1} = \begin{pmatrix} w_2 & w_3 & 0 & 0 & 0 & 0 \\ w_1 & w_2 & w_3 & 0 & 0 & 0 \\ 0 & w_1 & w_2 & w_3 & 0 & 0 \\ 0 & 0 & w_1 & w_2 & w_3 & 0 \\ 0 & 0 & 0 & w_1 & w_2 & w_3 \\ 0 & 0 & 0 & 0 & w_1 & w_2 \end{pmatrix}.$$

Question c

Solution: Since we are provided with a single zero padding on each side, there will only be 3 units in the convolutional layer. They are yielded by filter windows center at the 1st, 3rd and 5th input unit. So the answer is a 3 × 6 matrix

$$\mathbf{W_2} = \begin{pmatrix} w_2 & w_3 & 0 & 0 & 0 & 0 \\ 0 & w_1 & w_2 & w_3 & 0 & 0 \\ 0 & 0 & 0 & w_1 & w_2 & w_3 \end{pmatrix}.$$