Anna Krasilnikova 2562668
Mirza Misbah Mubeen Baig 2571567
Shahzain Mehboob 2571564

Exercise 6.1 – *Maximum Likelihood Estimation (MLE) and Cross-Entropy*

Given a set of m i.i.d. samples $X = \{x^{(1)}, \dots, x^{(m)}\}$ drawn from a data-generating distribution $\hat{p}_{data}(x)$ and a parametric family of probability distributions over the same space $p_{model}(x;\theta)$.

a) *Write down the maximum likelihood estimator for θ.*

The maximum likelihood estimator for θ can be defined as:

$$\theta_{ML} = arg\max_{\theta} p_{model}(x;\theta) = arg\max_{\theta} \prod_{i=1}^{m} p_{model}(x^{(i)};\theta)$$

b) *Explain the difference between empirical distribution $\hat{p}_{data}(x)$ and the data-generating distribution $p_{data}(x)$.*

The data generating distribution is the underlying distribution of the training dataset. If the given samples are generated by a normal distribution. In the reality, however, it is the distribution we train th model using the samples in hand to get.[1]

On the other hand, the empirical distribution is the distribution associated with the empirical measure of a sample, strictly speaking, we don't know anything at the start – we have just a collection of observations, and we want to derive some knowledge from that collection, we are just taking empirical measure of a sample (random measure arising from a particular realization of a sequence of random variables)[2]

c) *Rewrite the expression derived in a) as an expectation using empirical distribution $\hat{p}_{data}(x)$. Give an argument why it is possible.*

First thing we can do – to take the log MLE, since it won't change its likelihood. Secondly, we know that argmax does not change over rescaling. Hence, we can divide the formula defined in a) by n to get the criterion as an expectation. At the end we can get following:

$$MLE_{\theta} = arg\max_{\theta} \frac{1}{m} \sum^{m} \log(p_{model}(\theta) = arg\max_{\theta} \mathbb{E}_{x\sim p}[\log(p_{model})\theta]$$

d) *Show that minimizing the cross-entropy between $\hat{p}_{data}(x)$ and $p_{model}(x;\theta)$ is exactly the same as computing the maximum likelihood estimator in a).*
Definition of KL-divergence is:

$$D_{KL}(P||Q) = \mathbb{E}_{x\sim P}\left[log\frac{P(x)}{Q(x)}\right]$$

for Maximum likelihood for some $N(\mu, \sigma^2)$ given $\{x\}_i$, for $i \in 1..n$ we can make following observations:

Let's start by converting the given definition of KL divergence, using p_{model} and $\hat{p}_{data}(x)$:

$$D_{KL}(\hat{p}_{data}||p_{model}) = \mathbb{E}_{x\sim\hat{p}_{data}}[\log(\hat{p}_{data}) - \log(p_{model})] = \mathbb{E}(\log(\hat{p})) - \mathbb{E}(\log(p))$$

Since we know that $\mathbb{E}(\log(\hat{p}))$ – is data-generation process function – we can minimize KL divergence by minimizing $-\mathbb{E}_{x\sim\hat{p}}[\log(p_{model})]$.

Hence at $\mathbb{E}_{x\sim\hat{p}}[\log(p_{model}(x;\ \theta))]$ it's actually equal to minimization. This concludes the proof.

Exercise 6.2 – *Validation and Cross-Validation*

---

[1] https://www.quora.com
[2] https://en.wikipedia.org/wiki/Empirical_distribution_function

Anna Krasilnikova 2562668
Mirza Misbah Mubeen Baig 2571567
Shahzain Mehboob 2571564

The commonly used activation function in hidden layers of a Neural Network is a Sigmoid function which is defined as:

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

a) *Prove that the derivative of sigmoid function is $\sigma(x) - \sigma^2(x)$*

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

Taking the derivative:

$$\frac{d\sigma(x)}{dx} = \frac{d}{dx}\left[\frac{1}{1+e^{-x}}\right]$$

$$= \frac{d}{dx}(1+e^{-x})^{-1} = -(1+e^{-x})^{-2}(-e^{-x}) = \frac{e^{-x}}{1+e^{-x}} = \frac{1}{1+e^{-x}} \cdot \frac{e^{-x}}{1+e^{-x}}$$

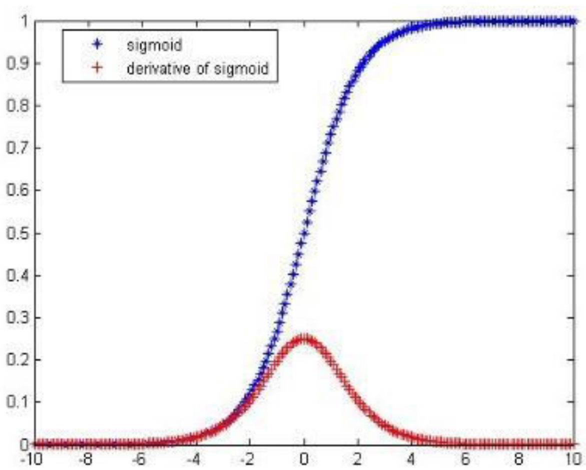
$$= \frac{1}{1+e^{-x}} \cdot \frac{(1+e^{-x})-1}{1+e^{-x}}$$

$$= \frac{1}{1+e^{-x}} \cdot \left(\frac{(1+e^{-x})}{1+e^{-x}} - \frac{1}{1+e^{-x}}\right) = \frac{1}{1+e^{-x}}\left(1 - \frac{1}{1+e^{-x}}\right)$$

$$= \sigma(x) \cdot \big(1 \cdot \sigma(x)\big) = \sigma(x) - \sigma^2(x),$$

which concludes the proof.

b) *Sketch the gradient of the sigmoid function and also explain what are the inherent properties you observe from the computed gradient?*



Here we observe that $\sigma'(x)$ evaluated at x is simply $\sigma(x)$ weighted by 1-minus $\sigma(x)$. This turns out to be a convenient form for efficiently calculating gradients used in neural networks, if one keeps in memory the Feed-forward activations of the sigmoid function for a given layer, the gradients for that layer can be evaluated using simple multiplication and subtraction rather than performing only re-evaluation, which requires extra exponentiation.

c) *Prove that the sigmoid function is point symmetric.*

The sigmoid function σ has the property that its graph $y = \sigma(x)$ has symmetry about point $(0, \frac{1}{2})$. It satisfies the equation $\sigma(x) + \sigma(-x) = 1$.

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

By multiplying numerator and denominator by e^{-x}, we get:

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

Anna Krasilnikova 2562668
Mirza Misbah Mubeen Baig 2571567
Shahzain Mehboob 2571564

Therefore,
since $\sigma(x) + 1 - \sigma(x) = 1 \implies 1 = 1$,
which satisfies that the sigmoid function is point symmetric with initial condition where $y = \frac{1}{2}$ at $x = 0$

d) *We know from Newton's method the importance of Taylor series in optimization, additionally, Taylor expansion could be beneficial in providing a cheaper computational alternative for activation functions. So find the first 3 terms in the Taylor series for the sigmoid function centered at 0.*

$$f(x) = \frac{1}{1+e^{-x}} \implies f(0) = \frac{1}{1+e^{0}} = \frac{1}{2}$$

$$f'(x) = \frac{e^{-x}}{(1+e^{-x})^2} \implies f'(0) = \frac{e^{0}}{(1+e^{0})^2} = \frac{1}{2^2} = \frac{1}{4}$$

$$f''(x) = \frac{(1+e^{-x})^2(-e^{-x}) - (e^{-x} \cdot 2(1+e^{-x})(-e^{-x})}{(1+e^{-x})^4}$$

$$= \frac{-e^{-x} - e^{-2x} + 2e^{-2x}}{(1+e^{-x})^3}$$

$$= \frac{-e^{-x} + e^{-2x}}{(1+e^{-x})^3} = \frac{e^{-x}(-1+e^{-x})}{(1+e^{-x})^3}$$

Hence.

$$f''(0) = \frac{e^{0}(-1+e^{0})}{(1+e^{0})^3} = \frac{-1+1}{(1+1)^3} = 0$$

$$f''(x) = \frac{(1+e^{-x})^3(-e^{-x}(-1+e^{-x}) + e^{-x} \cdot (-e^{-x}) - e^{-x}(-1+e^{-x}) \cdot 3(1+e^{-x})^2 \cdot (-e^{-x})}{(1+e^{-x})^5}$$

$$= \frac{(-1+e^{-x})(e^{-x} - e^{-2x} - e^{-2x}) - e^{-x}(-1+e^{-x})3(-e^{-x})}{(1+e^{-x})^4}$$

$$= \frac{(1+e^{-x})(e^{-x} - 2e^{-2x}) + 3e^{-2x}(-1+e^{-x})}{(1+e^{-x})^4}$$

$$= \frac{e^{-x} + e^{-2x} - 2e^{-2x} - 2e^{-3x} - 3e^{-2x} + 3e^{-3x}}{(1+e^{-x})^4}$$

$$= \frac{e^{-x} - 4e^{-2x} + e^{-3x}}{(1+e^{-x})^4} = \frac{e^{-x}(1 - 4e^{-x} + e^{-2x})}{(1+e^{-x})^4}$$

$$\implies f'''(0) = \frac{e^{0}(1 - 4e^{0} + e^{0})}{(1+e^{-x})^4} = \frac{1-4+1}{2^4} = -\frac{2}{16} = -\frac{1}{8}$$

$$T^3_{|0|}f(x) = \frac{f(0)}{0!}(x-0)^0 + \frac{f'(0)}{1!}(x-0)^1 + \frac{f''(0)}{2!}(x-0)^2 + \frac{f'''(0)}{3!}(x-0)^3$$

$$= \frac{\frac{1}{2}}{1}x^0 + \frac{\frac{1}{4}}{1}x^1 + \frac{0}{2}x^2 - \frac{\frac{1}{8}}{6}x^3 = \frac{1}{2} + \frac{1}{4}x^1 - \frac{1}{48}x^3$$

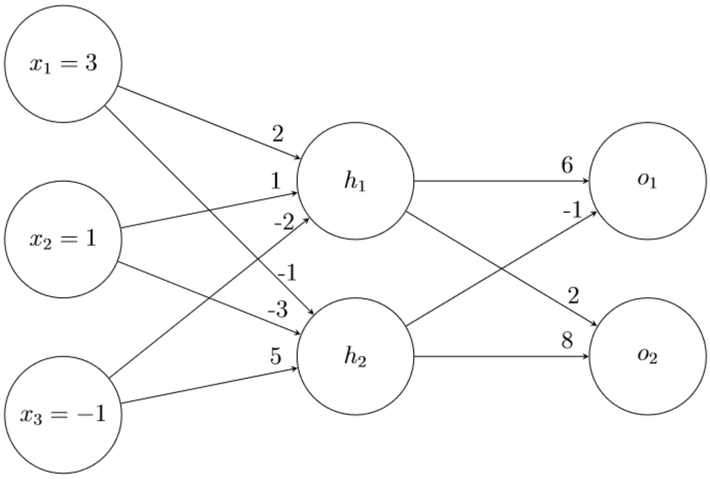
Exercise 6.3. – *Basics of Forward and Backward passes in computational graphs*

a) *The simple one-layer Neural Network takes an input of three features, and produces a vector output. Apply a forward pass with the given inputs and weights in the circles and above the arrows respectively, use ReLU function (ReLU function is defined as:*

$$ReLU(x) = max(0, x))$$

for the hidden nodes and softmax function for the output nodes. If this is a binary classification problem, what would be the predicted class label for this given input.

Anna Krasilnikova 2562668
Mirza Misbah Mubeen Baig 2571567
Shahzain Mehboob 2571564



Sizing Neural network:

$(3 \times 2) + (2 \times 2) = 10$ weights

$2 + 2 = 4$ biases

$\Longrightarrow$ 14 learnable parameters total

$$h_1 = \max[0, \{(2 \times 3) + (1 \times 1) + (-1 \times (-2))\}]$$
$$h_1 = \max[0, \{6 + 1 + 2\}]$$
$$h_1 = \max[0,9]$$
$$h_1 = 9$$
$$h_2 = \max[0, \{(3 \times (-1)) + (1 \times (-3)) + (-1 \times 5)\}]$$
$$h_2 = \max[0, \{-3 - 3 - 5\}]$$
$$h_2 = \max[0, -11]$$
$$h_2 = 0$$

For softmax function $\hat{Y}_i = sofmax(z)_i = \frac{\exp(z)_i}{\sum_j \exp(z)_i}$

$$O_{11} = \frac{\exp(9 \times 6)}{\exp(9 \times 6) + \exp(0 \times (-1))} = 1$$
$$O_{12} = \frac{\exp(0 \times (-1))}{\exp(9 \times 6) + \exp(0 \times (-1))} = 0$$
$$O_1 = [1,0]$$

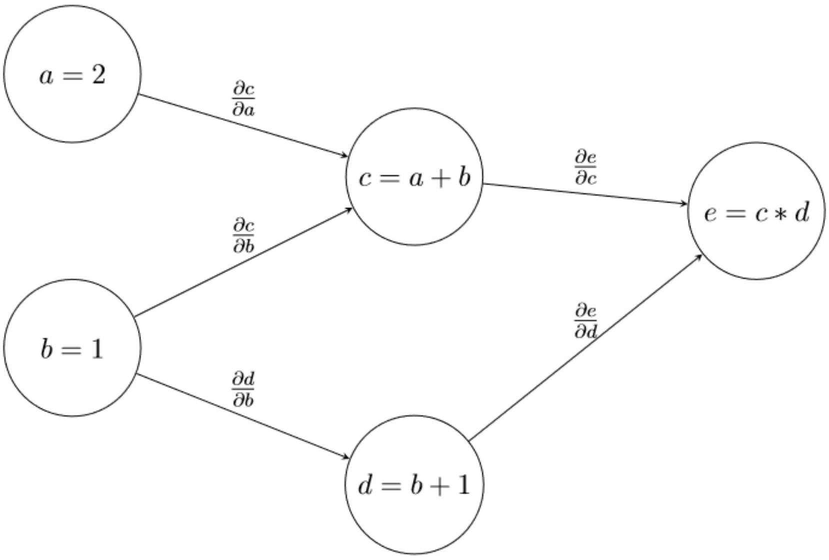
$$O_{21} = \frac{\exp(9 \times 2)}{\exp(9 \times 2) + \exp(0 \times 8)} = 1$$
$$O_{21} = \frac{\exp(0 \times 8)}{\exp(9 \times 2) + \exp(0 \times 8)} = 0$$
$$O_2 = [1,0]$$

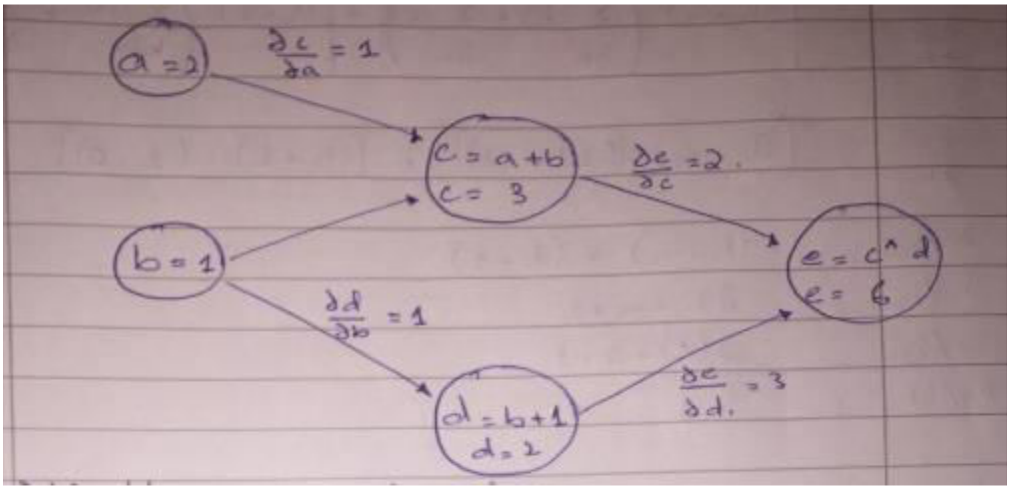
The predicted class label for the given input would be x_1 and $(h_1 | x_1, x_2, x_3)$

b) *For the computation graph, write down the expressions using chain rule and compute the final values for*

- $\frac{\partial e}{\partial b}$
- $\frac{\partial e}{\partial a}$

Anna Krasilnikova 2562668
Mirza Misbah Mubeen Baig 2571567
Shahzain Mehboob 2571564





Directly connected nodes:

$$\frac{\partial c}{\partial a} = \frac{\partial(a+b)}{\partial a} = \frac{\partial a}{\partial a} + \frac{\partial b}{\partial a} = 1$$

$$\frac{\partial d}{\partial b} = \frac{\partial(b+1)}{\partial b} = 1$$

$$\frac{\partial e}{\partial c} = \frac{\partial(c*d)}{\partial c} = d \cdot \frac{\partial(c)}{\partial c} = d = 2$$

$$\frac{\partial c}{\partial d} = \frac{\partial(c*d)}{\partial d} = c \cdot \frac{\partial(d)}{\partial d} = c = 3$$

Indirectly connected nodes:

$$1 - \frac{\partial e}{\partial b} = \frac{\partial(c*d)}{\partial b} = \frac{\partial(a+b)(b+1)}{\partial b}$$

$$\frac{\partial e}{\partial b} = \left[(b+a) \cdot \frac{\partial(b+1)}{\partial b}\right] + \left[(b+1) \cdot \left(\frac{\partial(b)}{b} + \frac{\partial(a)}{\partial b}\right)\right]$$

$$\frac{\partial e}{\partial b} = \left[(b+a) \cdot \left(\frac{\partial(b)}{\partial b} + \frac{\partial(1)}{\partial b}\right)\right] + \left[(b+1) \cdot \left(\frac{\partial(b)}{b} + \frac{\partial(a)}{\partial b}\right)\right]$$

$$\frac{\partial e}{\partial b} = [(b+a) \cdot ((1) + (a))] + [(b+1) \cdot (1+a)]$$

$$\frac{\partial e}{\partial b} = (b+a) + (b+1)$$

Anna Krasilnikova 2562668
Mirza Misbah Mubeen Baig 2571567
Shahzain Mehboob 2571564

$$\frac{\partial e}{\partial b} = 2b + a + 1$$

$$\frac{\partial e}{\partial b} = 2 \cdot (1) + 2 + 1$$

$$\frac{\partial e}{\partial b} = 5$$

$$2 - \frac{\partial e}{\partial a} = \frac{\partial (c * d)}{\partial a} = \frac{\partial (a + b)(b + 1)}{\partial a}$$

$$\frac{\partial e}{\partial a} = (b + 1)\left[\frac{\partial a}{\partial a} + \frac{\partial b}{\partial a}\right]$$

$$\frac{\partial e}{\partial a} = (b + 1)(1 + 0)$$

$$\frac{\partial e}{\partial a} = b + 1$$

$$\frac{\partial e}{\partial a} = 1 + 1 \Longrightarrow \frac{\partial e}{\partial a} = 2$$