Q3.

a) $\frac{(\text{input\_size} + 2 \times \text{padding\_size} - \text{Kernel\_size})}{\text{stride}}$

First layer:

$$\frac{(128 + 6 - 2)}{2} = \frac{128}{2} = 64$$

$$\Rightarrow \boxed{64 \times 64 \times 32}$$

Second layer: (max pooling & non-overlapping)

$$\frac{64}{2} = 32 \xrightarrow[\text{max pooling}]{\text{after}} 32 \times 32 \times 32$$

Third_layer: (64 3x3 , stride = 1 zero_padding = 1)

$$\Rightarrow \frac{32 + 2 - 3 + 1}{1} = 32$$

$$\Rightarrow \boxed{64 \times 32 \times 32}$$

b)

Number of parameters needed in the Conv layer without considering bias:

$(m \times n \times d) \times K$ ($m \times n$: filter height & width; d: num of filters in previous layer; K: num of filters in this layer)

$$\Rightarrow 3 \times 3 \times 32 \times 64 = \boxed{18432}$$

C)

Same Convolution:

Putting Center of filter on the first element.

[0][1, 1, 4, 4, 4, 1, 1][0]

[1/4, 1/4, 1/4] → Slide

$= [2/4, 6/4, 9/4, 3, 9/4, 6/4, 2/4]$

Full Convolution:

Bottom Right element of filter on Top Left element of matrix.

[0,0] [1, 1, 4, 4, 4, 1, 1] [0,0]

[1/4, 1/4, 1/4] → Slide

$= [1/4, 2/4, 6/4, 9/4, 3, 9/4, 6/4, 2/4, 1/4]$

Valid Convolution:

normal Convolution

[1, 1, 4, 4, 4, 1, 1]

[1/4, 1/4, 1/4] → slide

$= [6/4, 9/4, 3, 9/4, 6/4]$

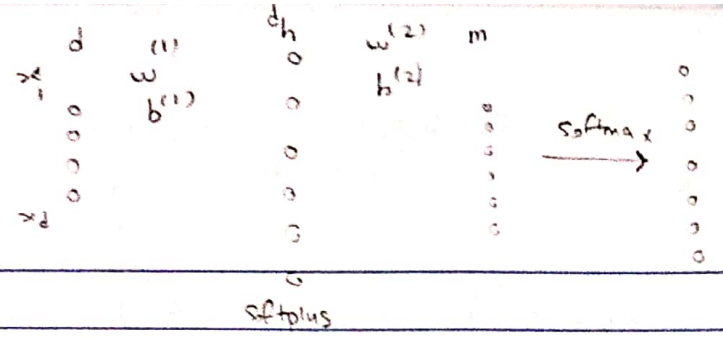
d)

- Valid & Full Convolution tend to keep data from the Corner of the image while Same Convolution uses no padding and might result in loss of data.

- Valid & Full Convolution makes the image bigger while Same Convolution reduces the size.

- Same Convolution is practical because at the end we want lower dimention representation of our data.

2.



4.

a) - $b^{(1)}$ contains a bias term for every d_h neuron in the hidden layer so it is a $d_h \times 1$ vector

- h^a in matrix form is: $h^a = W^{(1)} \cdot X + b^{(1)} \in R^{d_h}$

one element of $h^a \rightarrow h^a_j$ $\quad$ j^{th} row of $W^{(1)} \rightarrow W^{(1)}_{j,:}$

j^{th} element of $b^{(1)} \rightarrow b^{(1)}_j$

$$h^a_j = W^{(1)}_{j,:} \cdot X + b^{(1)}_j = \left(\sum_{i=1}^{d} W^{(1)}_{ji} \cdot X_i\right) + b^{(1)}_j$$

- output vector of the hidden layer h^s with respect to h^a is:

$h^s = \text{Softplus}(h^a)$ $\rightarrow$ softplus is applied on each d_h elements of h^a $\rightarrow h_s \in R^{d_h}$

b)

$W^{(2)}$ $\rightarrow$ just like what we had in $W^{(1)}$ each row of $W^{(2)}$ represents the weights for every neuron in the output layer (m) and each column represent weights for every neuron in the hidden layer (d_h)

So $W^{(2)} \in R^{m \times d_h}$

and likewise $b^{(2)}$ is R^m $\rightarrow$ m bias term per m output neuron

$$o^a = W^{(2)} \cdot h^{(s)} + b^{(2)} \in R^m$$

one element of $o^a \rightarrow o^a_k \Rightarrow$ k^{th} element of $W^{(2)} \rightarrow W^{(2)}_{k,:}$

k^{th} element of $b^{(2)} \rightarrow b^{(2)}_k$

$$o^a_k = W^{(2)}_{k,:} \cdot h^{(s)} + b^{(2)}_k = \left(\sum_{i=1}^{d_h} W^{(2)}_{ki} \, h^{(s)}_i\right) + b^{(2)}_k$$

(1)

c)

$$o^s = \text{Softmax}(o^a) = \frac{\exp(o^a)}{\sum_{k=1}^{m} \exp(o_k^a)}$$

$$o_k^s = \text{Softmax}(o_k^a) = \frac{\exp(o_k^a)}{\sum_{i=1}^{m} \exp(o_i^a)}$$

Since $\exp(x)$ is always positive o_k^s is just division of some $\exp(x)$ function so o_k^s are always positive.

$$\sum_{k=1}^{m} o_k^s = \sum_{k=1}^{m} \frac{\exp(o_k^a)}{\sum_{i=1}^{m} \exp(o_i^a)} = \frac{\sum_{k=1}^{m} \exp(o_k^a)}{\sum_{i=1}^{m} \exp(o_i^a)} = 1$$

why it is important? because the task is a classification problem and using a softmax as last layer has the interpretation of probability of the test input belong to each of the m class so according to the laws of probability it should sum to one to be a valid probability distribution.

d)

$$\text{Softmax} \Rightarrow p(Y=k \mid X) \quad k=1,\dots,m = \frac{e^{o_{ik}^a}}{\sum_{i=1}^{m} e^{o_i^a}} = \frac{1}{\sum_{i=1}^{m} e^{(o_i^a - o_{ik})}}$$

$$= \frac{1}{\sum_{i=1}^{m} e^{-(o_{ik}^a - o_i^a)}}$$

if $m=2 \longrightarrow$ binary classification assume labels are $\{0,1\}$

$$P(Y=1|X) = \frac{1}{e^{-(o_1^q - o_2^q)} + 1} = \frac{1}{e^{-(o_1^q - o_2^q)} + 1}$$

tion

$$= \frac{1}{1+e^{-o'^q}} \quad (\text{where } o' = o_1^q - o_2^q)$$

$$= \delta(o'^q) = \text{Sigmoid}$$

e)

$$L_{MSE}(\delta(o^q), Y) \qquad \delta(o^q) = o^{(s)}$$

$$\frac{\partial L_{MSE}(o^{(s)}, Y)}{\partial o^q} = \frac{\partial L(o^{(s)}, Y)}{\partial o^{(s)}} \times \frac{\partial o^{(s)}}{\partial o^{(q)}}$$

$$= \boxed{(\delta(o^q) - Y) \times \delta(o^q)(1 - \delta(o^q))}$$

f) $L_{CE}(\delta(o^q), y) = -(y \log(\delta(o^q)) + (1-y)\log(1-\delta(o^q)))$

n = # of training example

y = true value

$\delta(o^q)$ = predicted value

$$\frac{\partial L_{CE}(\delta(o^q), y)}{\partial o^q} = \frac{\partial L_{CE}(\delta(o^q), y)}{\partial(\delta(o^q))} \times \frac{\partial(\delta(o^q))}{\partial(o^q)}$$

$$\frac{\partial L_{CE}(\delta(o^q), y)}{\partial(\delta(o^q))} = \boxed{-\left(\frac{y}{\delta(o^q)} - \frac{(1-y)}{1-\delta(o^q)}\right) \times \delta(o^q)(1-\delta(o^q))}$$

g) first using MSE means that we assume that the underlying data has been generated from a normal distribution in bayesian term this means that we assume a guassion prior while in reality a dataset that can be classified into 2 categories is not from a normal distribution but a bernouil distribution

Secondly → MSE function is non Convex for binary classification (why?) in otherway it is not guaranteed to minimize cost function - this because MSE expects real valued input in range $(-\infty, +\infty)$ while binary classification model output probabilities in range $(0,1)$ through the Sigmoid function.

3) So when a bounded value from a sigmoid function is passed to the MSE the result is not convex. on one side the function is concave while on the otherside the function is convex and no clear minimum point

In addition binary cross entropy loss is equivalent to fitting the model using maximum likelihood estimation and the decision boundry in a classification task is large (in comparison with regression) but MSE doesn't punish missclassification enough.

h)

$$L(x, y) = -\log o_y^S(x)$$

$$= -\log \text{softmax}\left(o_y^a(x)\right)$$

$$= -\log \frac{\exp\left(o_y^a(x)\right)}{\sum_{i=1}^{m} \exp\left(o_i^a(x)\right)}$$

i)

$$\hat{R}(f, D) = \frac{1}{n} \sum_{i=1}^{n} L\left(f_\theta(x^{(i)}), y^{(i)}\right)$$

which is the average over a finite dataset D

θ is the parameter that the model learns during training, include all the weight and biased in all the layers:

$$\theta = \left\{ W^{(1)}, b^{(1)}, W^{(2)}, b^{(2)} \right\}$$

$W^{(1)} \in R^{d_h \times d}$ $\quad b^{(1)} \in R^{d_h}$ $\quad W^{(2)} \in R^{m \times d_h}$ $\quad b^{(2)} \in R^{m}$

$$n_\theta = d_h \times (d+1) + m \times (d_h + 1)$$ scalar parameters

the optimization problem of training the model to find the optimal values of parameter is:

$$\theta^* = \underset{\theta}{\text{argmin}}\ \hat{R}(f_\theta, D) = \underset{\theta}{\text{argmin}}\ \frac{1}{n} \sum_{i=1}^{n} L\left(f_\theta(x^{(i)}), y^{(i)}\right)$$

j)

learning rate

$$\theta^{(t+1)} = \theta^{(t)} - \eta \nabla_\theta \hat{R} = \theta^{(t)} - \eta \frac{\partial}{\partial \theta} \left(\frac{1}{n} \sum_{i=1}^{n} L\left(f_\theta(x^{(i)}), y^{(i)}\right) \right)$$

$$\theta = \left[W^{(1)}, b^{(1)}, W^{(2)}, b^{(2)} \right]$$

$$W^{(1)(t+1)} = W^{(1)(t)} - \eta \frac{\partial}{\partial W^{(1)}} \hat{R}$$

$$W^{(2)(t+1)} = W^{(2)(t)} - \eta \frac{\partial}{\partial W^{(2)}} \hat{R}$$

$$b^{(1)(t+1)} = b^{(1)(t)} - \eta \frac{\partial}{\partial b^{(1)}} \hat{R}$$

$$b^{(2)(t+1)} = b^{(2)(t+1)} - \eta \frac{\partial}{\partial b^{(2)}} \hat{R}$$

k)

$$\theta^{(t+1)} = \theta^{(t)} - \eta \frac{\partial}{\partial \theta} \left(\frac{1}{n} \sum_{i=1}^{n} L\left(f_\theta^{(t)}(x^{(i)}), y^{(i)}\right) \right) - \eta \frac{\partial}{\partial \theta} \left(\lambda \|\theta\|_2^2 \right)$$

l)

$$L(x, y) = -\log o_y^s(x) = -\log \text{Soft max}\left(o_y^a(x)\right) = -\log \frac{\exp\left(o_y^a(x)\right)}{\sum_{i=1}^{m} \exp\left(o_i^a(x)\right)}$$

$$\frac{\partial L(x,y)}{\partial o_k^a} = \frac{\sum_{i=1}^{m} \exp\left(o_i^a(x)\right)}{\exp\left(o_y^a(x)\right)} \times \frac{\partial}{\partial o_k^a} \left(\frac{\exp\left(o_y^a(x)\right)}{\sum_{i=1}^{m} \exp\left(o_i^{(a)}(x)\right)} \right)$$

for $k \neq y$

$$+ \frac{\exp\left(o_k^{(a)}(x)\right) \exp\left(o_y^a(x)\right)}{\left(\sum_{i=1}^{m} \exp\left(o_i^{(a)}(x)\right)\right)^2} \times \frac{\exp\left(o_i^{(a)}(x)\right)}{\exp\left(o_y^a(x)\right)}$$

$$= \frac{\exp\left(o_k^a(x)\right)}{\sum_{i=1}^{m} \exp\left(o_i^{(a)}(x)\right)} = o_k^s$$

(5)

$$\frac{\partial L(x,y)}{\partial o_y^a} = -\frac{\sum_{i=1}^{m} \exp(o_i^a(x))}{\exp(o_y^a(x))} \times \frac{\partial}{\partial o_y^a}\left(\frac{\exp(o_y^a(x))}{\sum_{i=1}^{m} \exp(o_i^a(x))}\right)$$

$$= -\frac{\exp(o_y^a(x))\sum_{i=1}^{m}\exp(o_i^a(x)) - \exp(o_y^a(x))^2}{\left(\sum_{i=1}^{m}\exp(o_i^a(x))\right)^2} \times \frac{\sum_{i=1}^{m}\exp(o_i^a(x))}{\exp(o_y^{(a)}(x))}$$

$$= -\frac{\exp(o_y^a(x))\left(\sum_{i=1}^{m}\exp(o_i^a(x)) - \exp(o_y^a(x))\right)}{\sum_{i=1}^{m}\exp(o_i^{(a)}(x))} \times \frac{1}{\exp(o_y^{(a)}(x))}$$

$$\Rightarrow -\left[1 - \frac{\exp(o_y^a(x))}{\sum_{i=1}^{m}\exp(o_i^{(a)}(x))}\right] = -1 + o_y^s(x)$$

$$\frac{\partial L(x,y)}{\partial o_i^a} = \begin{bmatrix} o_1^s - 0 \\ \vdots \\ o_y^s(x) - 1 \\ \vdots \\ o_m^s - 0 \end{bmatrix} = \begin{bmatrix} o_1^s(x) \\ \vdots \\ o_y^s(x) \\ \vdots \\ o_m^s(x) \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix} = o^{(s)} - \text{onehot}_m(y)$$

$\left(\frac{\partial L}{\partial o^a}\right)$

(6)

m)

$$\frac{\partial L}{\partial w^{(2)}_{kj}} = \frac{\partial L}{\partial o^a_k} \times \frac{\partial o^a_k}{\partial w^{(2)}_{kj}}$$

$$\frac{\partial L}{\partial b^{(2)}_k} = \frac{\partial L}{\partial o^a_k} \times \frac{\partial o^a_k}{\partial b^{(2)}_k}$$

$$o^a_k = w^{(2)}_{k,:} \cdot h^{(s)} + b^{(2)}_k \Rightarrow \frac{\partial o^a_k}{\partial w^{(2)}_{kj}} = \frac{\partial w^{(2)}_{k,:} \cdot h^{(s)}}{\partial w^{(2)}_{kj}} + \frac{\partial b^{(2)}_k}{\partial w^{(2)}_{kj}} = \boxed{h^{(s)}_j}$$

$$\Rightarrow \boxed{\frac{\partial o^a_k}{\partial b^{(2)}_k} = 1}$$

$$\frac{\partial L(x,y)}{\partial o^a_k} = o^s_k(x) - \text{onehot}(y)_k \quad \text{from part (L)}$$

$$\Rightarrow \boxed{\frac{\partial L(x,y)}{\partial w^{(2)}_{kj}} = \frac{\partial L(x,y)}{\partial o^a_k} \times h^{(s)}_j = \left(o^s_k(x) - \underbrace{\text{onehot}_m(y)}_{k\text{th entry of the one-hot representation of label}}\right) \times h^{(s)}_j}$$

$$\boxed{\frac{\partial L(x,y)}{\partial b^{(2)}_k} = \frac{\partial L(x,y)}{\partial o^a_k} \times 1 = \frac{\partial L(x,y)}{\partial o^a_k} = \left(o^s_k(x) - \text{onehot}_m(y)_k\right)}$$

n)

$$\boxed{\frac{\partial L}{\partial w^{(2)}} = \frac{\partial L}{\partial o^a} \times \frac{\partial o^a}{\partial w^{(2)}} = \frac{\left(o^{(s)} - \text{onehot}(y)\right)}{m} h^{s\top}}$$

$$\boxed{\frac{\partial L}{\partial b^{(2)}} = \frac{\partial L}{\partial o^a} \times \frac{\partial o^a}{\partial b^{(2)}} = \frac{\partial L}{\partial o^a} = \left(o^{(s)} - \text{onehot}_m(y)\right)}$$

$$\left(\frac{\partial L}{\partial o^a}\right) = \underbrace{\left(o^{(s)} - \text{onehot}_m(y)\right)}_{m \times 1} \underbrace{h^{s\top}}_{1 \times d_h} \Rightarrow \boxed{\frac{\partial L}{\partial w^{(2)}} \text{ had dimension } m \times d_h \text{ like } w^{(2)}}$$

$$\boxed{\frac{\partial L}{\partial b^{(2)}} = \frac{\partial L}{\partial o^a} \Rightarrow \text{has dimension } m \times 1 \text{ like } b^{(2)}}$$

(7)

o)

$$\frac{\partial L}{\partial h_j^s} = \sum_{k=1}^{m} \frac{\partial L}{\partial o_k^a} \frac{\partial o_k^a}{\partial h_j^s}$$

$$o_k^a = W_{k,:}^{(2)} \cdot h^{(s)} + b_k^{(2)} = \left(\sum_{j=1}^{d_h} W_{kj}^{(2)} \cdot h_j^{(s)} \right) + b_k^{(2)}$$

$$\boxed{\frac{\partial o_k^a}{\partial h_j^{(s)}} = W_{kj}^{(2)}}$$

$$\boxed{\frac{\partial L}{\partial o_k^a} = \left(o_k^s(x) - \text{onehot}_m(y)_k \right)}$$

$$\boxed{\frac{\partial L}{\partial h_j^s} = \sum_{k=1}^{m} \left(o_k^{(s)}(x) - \text{onehot}_m(y)_k \right) W_{kj}^{(2)}}$$

p)

$$\frac{\partial L(x,y)}{\partial h^{(s)}} = \begin{bmatrix} \sum_{k=1}^{m} \left(o_k^{(s)} x - I(k=y) \right) W_{k1}^{(2)} \\ \vdots \\ \sum_{k=1}^{m} \left(o_k^s - I(k=y) \right) W_{kd_h}^{(2)} \end{bmatrix}_{d_h \times 1}$$

$$= \begin{bmatrix} \sum_{k=1}^{m} o_k^{(s)}(x) W_{k1}^{(2)} \\ \vdots \\ \sum_{k=1}^{m} o_k^{(s)}(x) W_{kd_h}^{(2)} \end{bmatrix} - \begin{bmatrix} \sum_{k=1}^{m} I_{(k=y)} W_{k1}^{(2)} \\ \vdots \\ \sum_{k=1}^{m} I(k=y) W_{kd_h}^{(2)} \end{bmatrix}$$

$$= W^{(2)T} o^{(s)} - W^{(2)T} \text{onehot}_m(y) = \boxed{W^{(2)T} \left(o^s - \text{onehot}_m(y) \right)}$$

$W^{(2)} \longrightarrow m \times d_h$

$\text{onehot}_m(y) \longrightarrow m \times 1$ $\Rightarrow \dfrac{\partial L(x,y)}{\partial h^{(s)}} = W^{(2)^T}\left(o^{(s)} - \text{onehot}_m(y)\right)$

$o^{(s)} \longrightarrow m \times 1$

$d_h \times m$, $m \times 1$ $\rightarrow$ $d_h \times 1$

dimension of previous vector $\left(\dfrac{\partial L}{\partial h^{(s)}}\right)$

q) $\dfrac{\partial L}{\partial h_j^a} = \dfrac{\partial L}{\partial h_j^{(s)}} \times \dfrac{\partial h_j^{(s)}}{\partial h_j^a}$ $\qquad h_j^{(s)} = \text{Softplus}(h_j^a)$

$$\frac{\partial h_j^{(s)}}{\partial h_j^a} = \frac{\partial \ln(1+\exp(h_j^a))}{\partial h_j^a} = \frac{1}{1+\exp(-h_j^a)}$$

$$\boxed{\frac{\partial L}{\partial h_j^a} = \left(\sum_{k=1}^{m} \left(o_k^{(s)}(x) - \text{onehot}_m(y)_k\right) W^{(2)}_{kj}\right) \frac{1}{1+\exp(-h_j^a)}}$$

r)

$$\frac{\partial L}{\partial h^a} = \begin{bmatrix} \left(\sum_{k=1}^{m} \left(o_k^{s}(x) - I(k=y)\right) W^{(2)}_{k1}\right) \frac{1}{1+\exp(-h_1^a)} \\ \vdots \\ \left(\sum_{k=1}^{m} \left(o_k^{s}(x) - I(k=y)\right) W^{(2)}_{kd_h}\right) \frac{1}{1+\exp(-h_{d_h}^a)} \end{bmatrix}$$

(9)

$$w^{(2)T}\left(o^{S} - \text{onehot}_{m}(y)\right) \cdot \frac{1}{1+\exp(-h^{q})} = \frac{\partial L}{\partial h^{q}}$$

$\left(o^{S} - \text{onehot}_{m}(y)\right) \rightarrow m \times 1$

(o^S: $m \times 1$, $\text{onehot}_m(y)$: $m \times 1$)

$w^{(2)T} \rightarrow d_h \times m$

$\frac{1}{1+\exp(-h^q)} \rightarrow d_h$

$\frac{\partial L}{\partial h^q} = (d_h \times m) \times (m \times 1) \times (d_h \times 1)$

$= d_h \times 1 \rightarrow$ dimension of $\frac{\partial L}{\partial h^q}$

HW 3

1- a) $\text{Sign}(x) = \mathbb{1}_{\{x \geq 0\}}(x) - \mathbb{1}_{\{x \leq 0\}}(x)$

b) $g(x) = \begin{cases} x & x>0 \\ 0 & x<0 \\ 0 & x=0 \end{cases} \Rightarrow g'(x) = \begin{cases} 1 & x>0 \\ 0 & x<0 \\ \text{not defined} & x=0 \end{cases}$

at $x=0$ $g'(x)$ doesn't exists because of the discontinuity.

$g'(0^-)$ and $g(0^+)$ are not equal. So $g'(x) = \begin{cases} 1 & x>0 \\ 0 & x<0 \end{cases}$

$H(x) = \begin{cases} 1 & x>0 \\ 0 & x<0 \\ \frac{1}{2} & x=0 \end{cases}$. So $g'(x)$ is equal to $H(x)$ where

$g'(x)$ exists (everywhere except $x=0$)

c) ① $g(x) = x H(x)$

② $g(x) = H(|x|) = H(-x)$

d) $\sigma(x) = \frac{1}{1+e^{-kx}}$ $\rightarrow$ when k is large $\xrightarrow{(\infty)}$ we have 3 possible values for x:

① $x>0 \Rightarrow m = kx$ will be ∞ x (positive) $(x) = +\infty \Rightarrow \lim_{m \to \infty} \frac{1}{1+e^{-m}} = 1$ ✓

because e^{-m} when $m \to \infty$ will be $e^{-\infty} \simeq 0$

② $x<0 \Rightarrow m = kx$ will be ∞ x (negative) $(x) = -\infty \Rightarrow \lim_{m \to -\infty} \frac{1}{1+e^{-m}} = 0$ ✓

because e^{-m} when $m \to -\infty$ will be $e^{-(-\infty)} = e^{+\infty} = \infty$ and $\frac{1}{1+\infty} \simeq 0$

③ $x=0 \Leftrightarrow n = kx = k \times 0 = 0 \Rightarrow \frac{1}{1+e^0} = \frac{1}{2}$ ✓

so it is like $H(x)$.

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

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

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

$J = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_m}{\partial x_1} & \cdots & \frac{\partial f_m}{\partial x_n} \end{bmatrix}$ where $\frac{\partial f_i}{\partial x_j} = \delta_{i=j}\sigma'(x_j) = \delta_{i=j}(\sigma(x_j) - \sigma^2(x_j))$

f) $\sigma(x) = \frac{1}{1+e^{-x}} \Rightarrow \ln(\sigma(x)) = \ln\left((1+e^{-x})^{-1}\right) =$

$-\ln(1+e^{-x})$

$\text{softplus}(x) = \ln(1+e^{x}) \Rightarrow -\text{softplus}(-x) = -\ln(1+e^{-x})$

$\text{so } \ln(\sigma(x)) = -\text{softplus}(-x)$ ✓

4.

g) $\text{SoftPlus}(x) - \text{SoftPlus}(-x) = \ln(1+e^x) - \ln(1+e^{-x})$

$= \ln(\delta(-x)) + \ln(-\delta(x))$ ... based on previous part

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

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

h) $S(x+c)_i = \frac{e^{(x_i+c)}}{\sum_j e^{(x_j+c)}} = \frac{e^c e^{x_i}}{e^c \sum_j e^{x_j}} = \frac{e^{x_i}}{\sum_j e^{x_j}}$

$= S(x)_i$ ✓

i) $S(cx)_i = \frac{e^{(cx)_i}}{\sum_j e^{(cx)_j}} \xrightarrow{cx=t} \Rightarrow S(t)_i =$

$\frac{e^{t_i}}{\sum_j e^{t_j}}$

$S(x)_i = \frac{e^{x_i}}{\sum_j e^{x_j}}$

if $S(t)_i = S(x)_i \Rightarrow \frac{e^{t_i}}{\sum_j e^{t_j}} = \frac{e^{x_i}}{\sum_j e^{x_j}}$

in this case, we should have $t = x$ in order to have the above equation correct $\Rightarrow t = cx \Rightarrow cx \overset{?}{=} x$ it is only true if $c=1$. So in general we can say that softmax is not invariant under multiplication

i cont.) $S(cx)_i = \frac{e^{cx_i}}{\sum_j e^{cx_j}} = \frac{(e^{x_i})^c}{\sum_j (e^{x_j})^c} = \frac{1}{\sum_j \left(\frac{e^{x_j}}{e^{x_i}}\right)^c}$

$= \left(\sum_j \left(\frac{e^{x_j}}{e^{x_i}}\right)^c\right)^{-1}$

↳ considering this part, when $c>1$ then this part will be greater if $x_i < x_j$ and will be smaller i.e $x_i > x_i$

Finally The result of the sum will be defendant to how many x_j are smaller than x_i and how much is the difference. Because forexample $\frac{1}{e} + 1.25 < (\frac{1}{e})^2 + (1.25)^2$ but $\frac{1}{e} + 1.1 > (\frac{1}{e})^2 + (1.1)^2$

The same things happens when $c<1$ but in the opposite way

Finally the result will be to the power of -1 which makes the result opposite if it was larger. The multiplications make the results more spiky and we it gets smaller and vice versa.

j) $\frac{\partial S(x)}{\partial x_j} = \left[\frac{\partial S(x)_1}{\partial x_j}, \frac{\partial S(x)_2}{\partial x_j} \dots \frac{\partial S(x)_n}{\partial x_j}\right]$ will be more confident

if $i \neq j$ → $\frac{\partial S(x)_i}{\partial x_j} = \frac{\partial}{\partial x_j}\left(\frac{e^{x_i}}{\sum_t e^{x_t}}\right) = \frac{-e^{x_i} e^{x_j}}{(\sum_t e^{x_t})^2} = \frac{-e^{x_i}}{\sum_t e^{x_t}} \frac{e^{x_j}}{\sum_t e^{x_t}}$

$= -S(x)_i S(x)_j$

if $i = j$ → $\frac{\partial S(x)_i}{\partial x_j} = \frac{e^{x_i} \sum_t e^{x_t} - e^{x_i} e^{x_j}}{(\sum_t e^{x_t})^2} = S(x)_i - S(x)_i S(x)_j$

⇒ writing the above results together we get:

$$\frac{\partial S(x)_i}{\partial x_j} = \delta_{i=j} \cdot S(x)_i - S(x)_i S(x)_j = S(x)_i (\delta_{i=j} - S(x)_j)$$

6

k) $\frac{\partial S(x)_i}{\partial x_j} = S(x)_i (\delta_{i=j} - S(x)_j)$ ⇝ from previous part

⇒ the diagonal elements have $i=j$ and will be:

$S(x)_i - S(x)_i S(x)_j$

and the others will be $(i \neq j) = -S(x)_i S(x)_j$

$$\Rightarrow \frac{\partial S(x)_i}{\partial x_j} = \text{diag}(S(x)) - \begin{bmatrix} S(x)_1 \\ S(x)_2 \\ \vdots \\ S(x)_n \end{bmatrix} \begin{bmatrix} S(x)_1 & S(x)_2 & \cdots & S(x)_n \end{bmatrix}$$

$$= \text{diag}(S(x)) - S(x)\,S(x)^T$$

m)

$$\nabla_u L(x, c) = \sum_{i=1}^{K} -c_i \nabla_u \log(S(x(u))_i) \quad (1)$$

$$\frac{\partial \log(S(x(u))_k)}{\partial u_K} \xrightarrow{K=j} \frac{S(x(u))_K - S(x(u))_k S(x(u))_j}{\log(S(x(u))_k)}$$

$K \neq j$

$$= \frac{-S(x(u))_k S(x(u))_j}{\log(S(x(u))_k)}$$

using the result of previous question and equation (1) we get:

$$\nabla_u L(x, c) = -\sum_{i=1}^{K} c_i \left(\nabla_u x(u)_i - \nabla_u x(u)_i S(x(u))_k\right)$$

1)

4)

$$\nabla_u \log S(x(u))$$

$$= \frac{\partial \log S(x(u))}{\partial S(x(u))} \times \frac{\partial S(x(u))}{\partial x(u)} \times \frac{\partial x(u)}{\partial u}$$

$$= \frac{1}{S(x(u))} \times \left(\text{diag}\left(S(x(u)) - S(x(u)) S(x(u))^T\right) \nabla_u x(u) \right)$$

$$= \left(I - S(x(u))^T \right) \nabla_u x(u)$$

$$= \nabla_{j,u} x(u) - \nabla_u x(u) \, S(x(u))^T$$

$$\nabla_{i,u} x(u) - \underbrace{\sum_j \nabla_u x(u)_j \, S(x(u))_j^T}$$

according to the question $S(x(u))_j$ is the probability of j

$$= \nabla_{i,u} x(u) - E_j \left[\nabla_u x(u)_j \right]$$