MFDNN

Homework 3

24/03/21

3. (a) ① Does $-\log\left(\frac{\exp(f_y)}{\sum_{j=1}^{k}\exp(f_j)}\right) > 0$ hold?

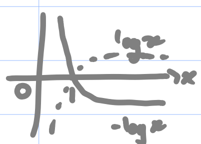
$\Leftrightarrow \quad \frac{\sum_{j=1}^{k}\exp(f_j)}{\exp(f_y)} > 1$ by applying exp to both sides (exp is increasing $\Rightarrow$ preserves inequality)

$\Leftrightarrow \quad \sum_{j=1}^{k}\exp(f_j) > \exp(f_y)$

$\Leftrightarrow \quad \sum_{\underbrace{j\in\{1\dots k\}\setminus\{y\}}_{=:K'}} \exp(f_j) > 0$

exp(x) is strictly positive for all $x \in \mathbb{R}$ so sum is also strictly positive and hence original ineq. holds, i.e. $\ell^{CE}(f,y) > 0$

② We need to show $-\log\left(\frac{\exp(f_y)}{\sum_{j=1}^{k}\exp(f_j)}\right) < \infty$, i.e. is finite/bounded above $\forall \underline{f}, y, k$



$$-\log\left(\frac{\exp(f_y)}{\sum_{j=1}^{k}\exp(f_j)}\right) = \log\left(\frac{\sum_{j=1}^{k}\exp(f_j)}{\exp(f_y)}\right)$$

$$= \log\left(\sum_{j=1}^{k}\exp(f_j)\right) - f_y \leq \log\left(k\exp(f_M)\right) - f_y$$

with $M := \arg\max_{i\in\{1,\dots,k\}}\{f_i\}$

i.e. M s.t. f_M is largest component of $\underline{f}$

$$= \log k + f_M - f_y < \infty$$ i.e. is finite since $k, f_M, f_y \in \mathbb{R}$

So $0 < \ell^{CE}(\underline{f}, y) < \infty$ ∎

(b) $$\ell^{CE}(\lambda \underline{e}_y, y) = -\log\left(\frac{\exp((\lambda \underline{e}_y)_y)}{\sum_{j=1}^{k}\exp((\lambda \underline{e}_y)_j)}\right)$$

$$= -\log\left(\frac{\exp(\lambda)}{\exp(\lambda) + \sum_{j\in K'}\exp(0)}\right) = -\log\left(\frac{\exp(\lambda)}{\exp(\lambda) + (k-1)}\right)$$

$-\log$ is continuous so $\lim_{\lambda\to\infty} -\log(\dots) = -\log\left(\lim_{\lambda\to\infty}\left(\frac{\exp\lambda}{\exp\lambda + k - 1}\right)\right)$

$$= -\log\left(\lim_{\lambda\to\infty}\left(\frac{1}{1+(k-1)e^{-\lambda}}\right)\right)$$

$$= -\log\left(\frac{1}{1+(k-1)(0)}\right)$$

$$= -\log(1) = 0$$

So $\ell^{CE}(\lambda \underline{e}_y, y) \to 0$ as $\lambda \to \infty$ ∎

f differentiable if this limit exists

4. $\frac{d}{dx} f(x) = \lim_{h\to 0}\frac{f(x+h)-f(x)}{h} = \lim_{h\to 0}\frac{f(x+h)-f_I(x)}{h}$ using given assumption on f at given $x\in\mathbb{R}$

If I is unique at $x\in\mathbb{R}$ $\Rightarrow$ there is a small neighbourhood, $(x-\delta, x+\delta)$, $\delta>0$, where I remains unique.
i.e. a neighbourhood where $f(x)=f_I(x)$ exactly.
(If there was no $\delta>0$ for which this was true, then necessarily another function f_i with the value of f_I exists at x and then I wouldn't be unique – contradiction)

So for all h s.t. $0<|h|<\delta$, $f(x+h) = f_I(x+h)$, hence

$$= \lim_{h\to 0}\frac{f(x+h)-f_I(x)}{h} = \lim_{h\to 0}\frac{f_I(x+h)-f_I(x)}{h} = \frac{d}{dx} f_I(x) \quad \blacksquare$$

since f_i are all differentiable, differential exists for $i=I$

5.(a) Given $z\in\mathbb{R}$, $\sigma(z) := \max\{0,z\}$,

If $z>0$, $\sigma(z)=z \Rightarrow \sigma(\sigma(z)) = \sigma(z) = z = \sigma(z)$

If $z\le 0$, $\sigma(z)=0 \Rightarrow \sigma(\sigma(z)) = \sigma(0) = 0 = \sigma(z)$ $\Rightarrow$ $\sigma(\sigma(z)) = \sigma(z)$ $\blacksquare$

(b) $\sigma_s(z) = \log(1+e^z)$ $\qquad$ $\sigma_s'(z) = \frac{e^z}{1+e^z} = e^z(1+e^z)^{-1}$ $\quad \forall z\in\mathbb{R}$ (since $e^z>0\ \forall z$)

$$\sigma_s''(z) = e^z(1+e^z)^{-1} - e^z\cdot e^z(1+e^z)^{-2} = \frac{e^z+e^{2z}}{(1+e^z)^2} - \frac{e^{2z}}{(1+e^z)^2} = \frac{e^z}{(1+e^z)^2}$$

$$|\sigma_s''(z)| = \left|\frac{e^z}{(1+e^z)^2}\right| = \left|\frac{1}{(1+e^{-z})(1+e^z)}\right| = \left|\frac{1}{2+e^z+e^{-z}}\right| < \frac{1}{2}$$

So $\forall z\in\mathbb{R}$, $|\sigma_s''(z)|$ is bounded, which implies that $\sigma_s'(z)$ is Lipschitz continuous.
(for differentiable f, $|f'(x)|\le M$ for all $x\in I$ for some $M>0 \Rightarrow f$ is L. cts.)

$\sigma_R(z) = \max\{0,z\}$ $\qquad$ $\sigma_R'(z) = \begin{cases} 0 & \text{for } z<0 \\ 1 & \text{for } z>0 \\ \text{undefined} & \text{for } z=0 \end{cases}$

Assume $\sigma_R'(z)$ <u>is</u> Lipschitz cts. That is, $\exists M>0$ s.t. $\forall x,y\in\mathbb{R}$

$$|\sigma_R'(x)-\sigma_R'(y)| \le M|x-y|$$

Take $x=\frac{1}{2M}$, $y=-\frac{1}{4M}$, then

$$|\sigma_R'(x)-\sigma_R'(y)| = |1-0| = 1$$

and, $M|x-y| = M\left|\frac{1}{2M}+\frac{1}{4M}\right| = \frac{3}{4}$.

$1 \nleq \frac{3}{4}$ so inequality doesn't hold $\forall x,y\in\mathbb{R}$ i.e. contradiction $\Rightarrow$ $\sigma_R'(z)$ is not Lipschitz cts.

$\sigma(z) = (1+e^{-z})^{-1}$; $\rho(z) = (1-e^{-2z})/(1+e^{-2z})$

5.(c) First note that $2\sigma(2z) - 1 = \frac{2}{1+e^{-2z}} - \frac{1+e^{-2z}}{1+e^{-2z}} = \frac{1-e^{-2z}}{1+e^{-2z}} = \rho(z) \Leftrightarrow \frac{\rho(\frac{1}{2}z)+1}{2} = \sigma(z)$

We continue by induction.

- For $L=2$, $\quad y_{2_s} = A_2 y_{1_s} + \underline{b}_2 \quad$ while $\quad y_{2_t} = C_2 y_{1_t} + \underline{d}_2$
  and $\quad y_{1_s} = \underline{\sigma}(A_1 \underline{x} + \underline{b}_1) \qquad y_{1_t} = \underline{\rho}(C_1 \underline{x} + \underline{d}_1)$

Using the relationship given above, $\quad y_{1_t} = 2\underline{\sigma}(2C_1\underline{x} + 2\underline{d}_1) - \underline{1}$ ← $\underline{1} \in \mathbb{R}^{n_1}$ of just 1s

$$y_{2_t} = 2C_2\underline{\sigma}(2C_1\underline{x} + 2\underline{d}_1) - C_2\underline{1} + \underline{d}_2$$

To represent identical mappings, $y_{2_t} = y_{2_s}$, we would therefore require,

$$2C_2 = A_2 \ ; \ \underline{b}_2 = \underline{d}_2 - C_2\underline{1} \ ; \ A_1\underline{x} + \underline{b}_1 = 2C_1\underline{x} + 2\underline{d}_1$$

$$\Updownarrow$$

$$A_1 = 2C_1 \ ; \ \underline{b}_1 = 2\underline{d}_1$$

So, given $A_1, A_2, \underline{b}_1, \underline{b}_2$, define,

$$C_\ell := \tfrac{1}{2}A_\ell \quad \text{and} \quad \underline{d}_1 := \tfrac{1}{2}\underline{b}_1$$

$$\underline{d}_2 := \underline{b}_2 + C_2\underline{1} = \underline{b}_2 + \tfrac{1}{2}A_2\underline{1}$$

and then the two MLPs represent equivalent mappings with $y_{2_t} = y_{2_s}$ for equal $\underline{x}$ inputs.

- Now assume true for $L = i-1$ and consider $L = i$ (i.e. output of $\ell-1$ layer networks equivalent),

$$y_{i_s} = A_i y_{i-1_s} + \underline{b}_i \quad \text{and} \quad y_{i_t} = C_i y_{i-1_t} + \underline{d}_i$$

By induction step, given $A_1, \ldots, A_{i-1}, \underline{b}_1, \ldots, \underline{b}_{i-1}$, it is possible to find $C_1, \ldots, C_{\ell-1}, \underline{d}_1, \ldots, \underline{d}_{\ell-1}$ s.t.

$$y'_{i-1} := A_{i-1} y_{i-2_s} + \underline{b}_{i-1} = \overbrace{C_{i-1} y_{i-2_t} + \underline{d}_{i-1}}^{y'_{i-1_t}} .$$

Hence, $\quad y_{i-1_s} = \underline{\sigma}(y'_{i-1}) \quad$ and $\quad y_{i-1_t} = \underline{\rho}(y'_{i-1})$

$$\Rightarrow \ y_{i_s} = A_i\underline{\sigma}(y'_{i-1}) + \underline{b}_i \quad \text{and} \quad y_{i_t} = C_i\underline{\rho}(y'_{i-1}) + \underline{d}_i$$

We desire these to be equal, $\quad A_i\left(\tfrac{1}{2}\left(\underline{\rho}\left(\tfrac{1}{2}y'_{i-1}\right) + \underline{1}\right)\right) + \underline{b}_i = C_i\underline{\rho}(y'_{i-1}) + \underline{d}_i$

$$\tfrac{1}{2}A_i\underline{\rho}\left(\tfrac{1}{2}y'_{i-1}\right) + \tfrac{1}{2}A_i\underline{1} + \underline{b}_i = C_i\underline{\rho}(y'_{i-1}) + \underline{d}_i$$

So by setting $C_i := \frac{1}{2}A_i$, $\underline{d}_i := \frac{1}{2}A_i\underline{1} + \underline{b}_i$, keeping the other inductively given $C_\ell, \underline{d}_\ell$ but crucially, adjusting $C_{i-1} \mapsto \frac{1}{2}C_{i-1}$ and $\underline{d}_{i-1} \mapsto \frac{1}{2}\underline{d}_{i-1}$ (so that $y'_{\ell-1_t} = \frac{1}{2}y'_{\ell-1}$), we can make the final ℓ-th layer outputs of the networks too equal ($(\ell-1)$th layers no longer need to be equal).

- Hence, since true for $L=2$, true for all $L \geq 2 \Leftrightarrow$ MLPs are identical mappings. ∎

(note: if given $C_\ell, \underline{d}_\ell$ similar rearrangements can be found identically)

$\underline{\theta} := \{a_1, \dots, a_p, b_1, \dots, b_p, u_1, \dots, u_p\}$

6. For this optimization problem, $\underset{\underline{\theta} \in \mathbb{R}^{3p}}{\text{minimize}} \ \frac{1}{N} \sum_{i=1}^{N} \ell(f_{\underline{\theta}}(x_i), y_i)$,
SGD takes the form:

$$\underline{\theta}^{k+1} = \underline{\theta}^{k} - r \nabla_{\underline{\theta}} \ell_{\underline{\theta}} \left(f_{\underline{\theta}}(X_{i(k)}), Y_{i(k)} \right), \quad r \text{ is learning rate}$$

$Y_{i(k)}$ const. wrt $\underline{\theta}$

$$\nabla_{\underline{\theta}} \ell_{\underline{\theta}} \left(f_{\underline{\theta}}(X_{i(k)}), Y_{i(k)} \right) = \nabla_{\underline{\theta}} f_{\underline{\theta}}(X_{i(k)}) \frac{d\ell_{\underline{\theta}}}{dx} + \underline{0}$$

this differential does exist since $\ell(x,y)$ assumed differentiable in x

$$\nabla_{\underline{\theta}} f_{\underline{\theta}}(X_{i(k)}) = \nabla_{\underline{\theta}} \left(\sum_{j=1}^{p} u_j \sigma(a_j X_{i(k)} + b_j) \right)$$

$$= \left\{ \left(\underline{\sigma}'(\underline{a} X_{i(k)} + \underline{b}) \odot \underline{u} \right) X_{i(k)}, \ \underline{\sigma}'(\underline{a} X_{i(k)} + \underline{b}) \odot \underline{u}, \ \underline{\sigma}(\underline{a} X_{i(k)} + \underline{b}) \right\} \text{ by HW 2.6}$$

At initialization, $k=0$:

$$\left[\nabla_{\underline{\theta}} f_{\underline{\theta}}(X_{i(0)}) \right]_j = \sigma'(a_j^0 X_{i(0)} + b_j^0) u_j^0 X_{i(0)} = 0$$ since $a_j^0 X_{i(0)} + b_j^0 < 0 \ \forall i$ (given) and $\sigma'(z) = 0 \ \forall z < 0$

$$\left[\quad " \quad \right]_{p+j} = \sigma'(a_j^0 X_{i(0)} + b_j^0) u_j^0 = 0 \quad "$$

$$\left[\quad " \quad \right]_{2p+j} = \sigma(a_j^0 X_{i(0)} + b_j^0) = 0$$ by given 'dead condition' directly

So j-th, $(p+j)$th, and $(2p+j)$th components of $\nabla_{\underline{\theta}} f_{\underline{\theta}}(X_{i(0)})$ are 0, so when $\underline{\theta}^{k+1}$ is calculated in SGD these components do not change at all; that is, a_j and b_j (and u_j) do not change, meaning the above gradient component calculations remain the same (equaling 0) for all $k \geq 0$ since a_j, b_j never change from their initialization values which 'killed' the j-th ReLU layer in the first place. Hence, if dead on initialization, the j-th ReLU output remains so throughout training ∎

i.e. unchanging/invariable

7. If σ is the leaky ReLU function, $\sigma'(z) = \begin{cases} 1 & \text{for } z \geq 0 \\ \alpha & \text{for } z < 0 \end{cases}$, i.e. $\sigma'(z) \neq 0$ for $z < 0$.
So the above become:

$$\left[\nabla_{\underline{\theta}} f_{\underline{\theta}}(X_{i(0)}) \right]_j = \sigma'(a_j^0 X_{i(0)} + b_j^0) u_j^0 X_{i(0)} = \alpha u_j^0 X_{i(0)},$$

$$\left[\quad " \quad \right]_{p+j} = \sigma'(a_j^0 X_{i(0)} + b_j^0) u_j^0 = \alpha u_j^0,$$

$$\left[\quad " \quad \right]_{2p+j} = \sigma(a_j^0 X_{i(0)} + b_j^0) = \alpha a_j^0 X_{i(0)} + b_j^0,$$

unexpected factorial :)

none of which are 0! So a_j and b_j (and u_j) are updated for the next iteration of SGD and the value of input/output to the j-th ReLU layer will change and the gradient always be non-zero (if $\theta_i \neq 0 \ \forall i$)