MATH3431 Machine Learning and Neural Networks III — Easter term 2025

Revision sheet

Lecturer & Author: Georgios P. Karagiannis — georgios.karagiannis@durham.ac.uk

Exercise 1. Consider a prediction rule $h : \mathbb{R}^d \rightarrow \mathbb{R}^q_+$ with $h(x) = (h_1(x), ..., h_q(x))^\top$ which receives inputs $x = (x_1, ..., x_d)^\top \in \mathbb{R}^d$ and which is modeled as a feedforward neural network (NN) with equation

$$h_k(x) = \sigma_2\left(\sum_{j=1}^{c} w_{2,k,j}\sigma_1\left(\sum_{i=1}^{d} w_{1,j,i}x_i\right)\right)$$

for $k = 1, ..., q$. We consider activation functions $\sigma_1(a) = \frac{1}{1+\exp(-a)}$ and $\sigma_2(a) = \log(1 + \exp(a))$. Parameters $c \in \mathbb{N}_+$, and $d \in \mathbb{N}_+$ are considered as known, while the weights $\{w_{\cdot,\cdot,\cdot}\}$ of the NN are unknown. To learn the unknown weights $\{w_{\cdot,\cdot,\cdot}\}$, we specify the loss function

$$\ell(w, z = (x, y)) = \frac{1}{2}\|h(x) - y\|_2^2 = \frac{1}{2}\sum_{k=1}^{q}(h_k(x) - y_k)^2$$

where $z = (x, y)$ denotes an example, $x \in \mathbb{R}^d$ is the input vector (features), and $y = (y_1, ..., y_q)^\top \in \mathbb{R}^q$ is the output vector (targets).

(1) Perform the forward pass of the back-propagation procedure to compute the activations which may be denoted as $\{a_{t,i}\}$ and outputs which may be denoted as $\{o_{t,i}\}$ at each layer t.

(2) Perform the backward pass of the back-propagation procedure in order to compute the gradient

$$\nabla_w \ell(w, (x, y)) = \left(\left(\frac{\partial}{\partial w_{1,j,i}}\ell(w, (x, y))\right)_{j=1,i=1}^{c,d}, \left(\frac{\partial}{\partial w_{2,k,j}}\ell(w, (x, y))\right)_{k=1,j=1}^{q,c}\right)$$

of the loss function $\ell(w, z)$ with respect to w for any example $z = (x, y)$. Clearly state the steps of the procedure as well as state the quantities

$$\frac{\partial}{\partial w_{1,j,i}}\ell(w, (x, y)), \text{ and } \frac{\partial}{\partial w_{2,k,j}}\ell(w, (x, y))$$

for all $k = 1, ..., q$, $j = 1, ..., c$, and $i = 1, ..., d$.

Solution. I 've got $T = 2$ layers.

(1) Regarding the forward pass. It is

Set: Comp for $i = 1, ..., d$

$$o_{0,i}(x) = x_i$$

t=1: Comp

$$\alpha_{1,j}\left(x\right)=\sum_{i=1}^{d}w_{1,j,i}x_{i}$$

$$o_{1,j}\left(x\right)=\left(1+\exp\left(-\alpha_{1,j}\left(x\right)\right)\right)^{-1}$$

t=2: Comp

$$\alpha_{2,k}\left(x\right)=\sum_{j=1}^{c}w_{2,k,j}o_{1,j}\left(x\right)$$

$$o_{2,k}\left(x\right)=\log\left(1+\exp\left(\alpha_{2,k}\left(x\right)\right)\right)$$

Get: Comp for $k=1,...,q$

$$h_{k}\left(x\right)=o_{2,k}\left(x\right)$$

(2) Regarding the backward pass. It is

$$\frac{\mathrm{d}}{\mathrm{d}\xi}\sigma_{2}\left(\xi\right)=\frac{\exp\left(\xi\right)}{1+\exp\left(\xi\right)}=\sigma_{1}\left(\xi\right)$$

and

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}\xi}\sigma_{1}\left(\xi\right)=&-\frac{\exp\left(-\xi\right)}{1+\exp\left(-\xi\right)}\frac{1}{1+\exp\left(-\xi\right)}\\=&-\sigma_{1}\left(\xi\right)\left(1-\sigma_{1}\left(\xi\right)\right)\end{aligned}$$

t=T=2:

$$\begin{aligned}\tilde{\delta}_{T=2,k}&=\frac{\mathrm{d}}{\mathrm{d}o_{T,k}}\ell_{T}\left(w,z\right)\frac{\mathrm{d}o_{T,k}}{\mathrm{d}\alpha_{T,k}}\\&=\left(o_{T,k}-y_{k}\right)\sigma_{1}\left(\alpha_{T,k}\right)\\&=\left(h_{k}-y_{k}\right)\sigma_{1}\left(\alpha_{T,k}\right)\end{aligned}$$

or for $k=1,...,q$

$$\tilde{\delta}_{2,k}=\left(h_{k}-y_{k}\right)\sigma_{1}\left(\alpha_{T,k}\right)$$

t=1:

$$\begin{aligned}\tilde{\delta}_{t=1,k}=&\sum_{k=1}^{q}w_{1,k,j}\tilde{\delta}_{2,k}\ \frac{\mathrm{d}}{\mathrm{d}\xi}\sigma_{1}\left(\xi\right)\Big|_{\xi=\alpha_{j}}\\=&-\sigma_{1}\left(\alpha_{j}\right)\left(1-\sigma_{1}\left(\alpha_{j}\right)\right)\left[\sum_{k=1}^{q}w_{1,k,j}\tilde{\delta}_{2,k}\right]\end{aligned}$$

Hence

$$\frac{\mathrm{d}}{\mathrm{d}w_{2,k,j}}\ell\left(w,\left(x,y\right)\right)=\left(h_{k}-y_{k}\right)\sigma_{1}\left(\alpha_{2,k}\right)o_{1,k}\left(x\right)$$

$$\frac{\mathrm{d}}{\mathrm{d}w_{1,j,i}}\ell\left(w,\left(x,y\right)\right)=-\sigma_{1}\left(\alpha_{j}\right)\left(1-\sigma_{1}\left(\alpha_{j}\right)\right)\left[\sum_{k=1}^{q}w_{1,k,j}\tilde{\delta}_{2,k}\right]o_{0,i}\left(x\right)$$

Exercise 2. Consider the binary classification learning problem: Let the set of targets be $\mathcal{Y} = \{-1, +1\}$, let the set of inputs be $\mathcal{X} = \left\{x \in \mathbb{R}^d : \|x\|_2 \leq B\right\}$ for some scalar $B > 0$, let the prediction rule be $h_w(x) = x^\top w$, and let the loss function ℓ be

$$\ell(w, z = (x, y)) = \log\left(1 + \exp\left(-yx^\top w\right)\right),$$

for $x \in \mathcal{X}$, $y \in \mathcal{Y}$, and $w \in \mathcal{W}$ where $\mathcal{W} = \left\{w \in \mathbb{R}^d : \|w\|_2 \leq B\right\}$.

(1) Show that the resulting learning problem is convex-Lipschitz-bounded. Specify the parameter of Lipschitzness.

(2) Show that the above loss $\ell(w, z = (x, y))$ is $B^2/4$ smooth.

Hint:: You may use the mean value theorem which states that (under pre-assumed conditions), $f(b) - f(a) = \frac{\mathrm{d}}{\mathrm{d}x} f(x)\big|_{x=c} (b - a)$ for $a \leq c \leq b$.

(3) Consider a risk function $R_g(w) = \mathrm{E}_{z\sim g}(\ell(w, z = (x, y)))$ where g denotes the unknown data generating process. Assume there is a set of available examples $\mathcal{D} = \{z_i = (x_i, y_i); i = 1, ..., n\}$. Now assume that $w \in \mathbb{R}^d$. To learn w, we aim to compute $w^* \in \mathbb{R}^d$ such that

$$w^* = \arg\min_w (f(w))$$

where $f(w) = R_g(w) + \frac{\lambda}{2} \|w\|_2^2$.

(a) Show that the stochastic gradient descent algorithm with batch size one and with learning rate

$$\eta_t = \frac{1}{\lambda t}$$

at iteration $t \in \mathbb{N}_+$ which is used to address the learning problem under consideration has a recursion that can be written in the form

$$w^{(t+1)} = -\frac{1}{\lambda t} \sum_{j=1}^{t} v_j$$

where $\{v_j\}$ is the gradient of the loss function at certain values of w and example. Show your working.

(b) Compute the exact formula of v_j as a function of λ, t, $\mathcal{D}$, and $\left\{w^{(t)}\right\}$. Show your working.

Solution.

(1)

Convexity:: Note that the function $g : \mathbb{R} \to \mathbb{R}$, defined by $g(a) = \log(1 + \exp(a))$ is convex. To see this, note that

$$\frac{\mathrm{d}^2}{\mathrm{d}a^2} g(a) = \frac{\exp(a)}{(1 + \exp(a))^2} \geq 0$$

is non-negative. The convexity of $\ell(\cdot, z)$for all z follows as a composition of g with a linear function.

Lipschitzness:: The function $g(a) = \log(1 + \exp(a))$ is 1-Lipschitz since

$$\left|\frac{\mathrm{d}}{\mathrm{d}a} g(a)\right| = \frac{\exp(a)}{1 + \exp(a)} = \frac{1}{\exp(-a) + 1} \leq 1$$

Hence because $|x|_2 \leq B$, $g(a)$ is B-Lipschitz as a composition.

Boundness:: The norm of each hypothesis w is bounded by B according to the assumptions.

(2)

Smoothness:: It is

$$\begin{aligned}\frac{\mathrm{d}^2}{\mathrm{d}a^2} g(a) &= \frac{\exp(a)}{(1 + \exp(a))^2} \\ &= \frac{1}{\exp(a)(1 + \exp(-a))^2} \\ &= \frac{1}{2 + \exp(a) + \exp(-a)} \leq 1/4\end{aligned}$$

Combine this with the mean value theorem, to conclude that $\frac{\mathrm{d}}{\mathrm{d}a} g(a)$ is 1/4-Lipschitz. Hence, $\ell(\cdot, z)$for all z is $B^2/4$ smooth.

(3)

(a) I need to minimize

$$\begin{aligned}f(w) =& R_g(w) + \frac{\lambda}{2}\|w\|_2^2 \\ =& \mathrm{E}_{z\sim g}\left(\ell(w, z = (x, y)) + \frac{\lambda}{2}\|w\|_2^2\right)\end{aligned}$$

so the online SGD has a recursion

$$w^{(t+1)} = w^{(t)} - \frac{1}{\lambda t}\left(\lambda w^{(t)} + v_t\right)$$

where

$$v_t = \frac{\mathrm{d}}{\mathrm{d}\xi}\ell\left(\xi, z^{(t)} = \left(x^{(t)}, y^{(t)}\right)\right)\bigg|_{\xi = w^{(t)}}$$

and $\left(x^{(t)}, y^{(t)}\right)$ is a randomly drawn example from the dataset. The recursion is

$$\begin{aligned}
w^{(t+1)} =& w^{(t)} - \frac{1}{\lambda t}\left(\lambda w^{(t)} + v_t\right)\\
=& \left(1 - \frac{1}{t}\right) w^{(t)} - \frac{1}{\lambda t} v_t\\
=& \underbrace{\frac{t-1}{t} w^{(t)} - \frac{1}{\lambda t} v_t}_{=\zeta_t}\\
=& \frac{t-1}{t}\left(\frac{t-2}{t-1} w^{(t-1)} - \frac{1}{\lambda(t-1)} v_{t-1}\right) - \frac{1}{\lambda t} v_t\\
=& \frac{t-2}{t-1} w^{(t-1)} - \frac{1}{\lambda t}\left(v_{t-1} + v_t\right)\\
=& -\frac{1}{\lambda t}\sum_{j=1}^{t} v_j
\end{aligned}$$

(b) Regarding the exact formula of v_t

$$v_t = -\frac{\exp\left(-y^{(t)}\left(x^{(t)}\right)^\top w^{(t)}\right)}{1 + \exp\left(-y^{(t)}\left(x^{(t)}\right)^\top w^{(t)}\right)} y^{(t)} x^{(t)}$$

where $\left(x^{(t)}, y^{(t)}\right)$ is a randomly drawn example from the dataset. This is because

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}w}\ell\left(w, z = (x, y)\right) =& \frac{\mathrm{d}}{\mathrm{d}w}\log\left(1 + \exp\left(-y x^\top w\right)\right)\\
=& -\frac{\exp\left(-y x^\top w\right)}{1 + \exp\left(-y x^\top w\right)} y x
\end{aligned}$$