FTML session 8: 25/05/2023

1 ONE HIDDEN LAYER NEURAL NETWORK

We derive the gradient computations used in the session, with the same notations as in the instructions, where $\bar{h} \in \mathbb{R}^{m+1}$ and $\bar{x} \in \mathbb{R}^{d+1}$ are defined.

1.1 SGD

The neural network will be trained by SGD. Thus, we need to compute the gradient of the loss l_i for each sample (x_i, y_i).

$$l_i = \frac{1}{2}(f(x_i) - y_i)^2 = \frac{1}{2}(\hat{y}_i - y_i)^2 \tag{1}$$

There will be a gradient with respect to θ, and a gradient with respect to w_h, noted $\nabla_{w_h} l_i$. We drop the i index for simplicity, so the calculation is performed for a given input $x \in \mathbb{R}^d$, that outputs a prediction $\hat{y} \in \mathbb{R}$, with a hidden layer $h \in \mathbb{R}^m$.

In this exercise, the inputs will be stored as line vectors, using the usual convention of the design matrix in $\mathbb{R}^{n,d}$ (n is the number of samples).

1.2 The chain rule

The chain rule is a formal way of writing a product of jacobians in order to compute a gradient or a jacobian of a composition. For instance, if

— $\frac{\partial \hat{y}}{\partial h}$ denotes the jacobian of $h \mapsto \hat{y}$ ($\in \mathbb{R}^{(1,m)}$)

— $\frac{\partial l}{\partial \hat{y}}$ denotes the jacobian of $\hat{y} \mapsto l$ (which is just a derivative $\in \mathbb{R}$)

— $\frac{\partial l}{\partial h}$ denotes the jacobian of $h \mapsto l$ ($\in \mathbb{R}^{(1,m)}$)

The chain rule makes it easier to write and decompose the computation of the jacobian of a composed function. For instance,

$$\frac{\partial l}{\partial h} = \frac{\partial l}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial h} \tag{2}$$

and this product is implicitely a product of **matrices**. Often, we do not write the vectors where the jacobians are computed, although, strictly speaking, we should do it.

Remember that if defined, the gradient is the transpose of the jacobian. Hence, in order to obtain a gradient, we can compute Jacobians with the chain rule and then transpose the results.

1.3 Intermediate variables

Some quantities that are computed during the forward pass are useful in the backpropagation, such as $\text{pre}_y = \langle \theta, \bar{h} \rangle$. Thus, it is a useful, when computing the forward

pass, to return also these intermediate calculations. For each input $x \in \mathbb{R}^{1,d+1}$ (stored as a line vector), we can consider the following intermediate variables :

— $pre_h = \bar{x} w_h \in \mathbb{R}^{1,m}$.

— $pre_y = \langle \theta, \bar{h} \rangle = \bar{h}\theta \in \mathbb{R}$ (we use the convention that $\bar{h}$ is a line vector).

— $\hat{y} = \sigma(pre_y) \in \mathbb{R}$.

1.4 Gradient with respect to θ

We can apply the chain rule. The jacobian of $\theta \mapsto \hat{y}$ is :

$$\frac{\partial l}{\partial \theta} = \frac{\partial l}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial pre_y} \frac{\partial pre_y}{\partial \theta} \tag{3}$$

We have that l is the squared loss :

$$\begin{cases} \mathbb{R} \to \mathbb{R} \\ \hat{y} \mapsto \frac{1}{2}(\hat{y} - y)^2 \end{cases}$$

Hence,

$$\frac{\partial l}{\partial \hat{y}} = \hat{y} - y \tag{4}$$

We also have that, $\hat{y} = \sigma(pre_y)$, which leads to

$$\frac{\partial \hat{y}}{\partial pre_y} = \sigma'(pre_y) \tag{5}$$

(see discussion in class).

Finally, the jacobian of $\theta \mapsto pre_y$ is $\bar{h} \in \mathbb{R}^{1,m+1}$.

To conclude, we get that

$$\nabla_\theta l(\theta, w_h) = (\hat{y} - y)\sigma'(pre_y)\bar{h}^T \in \mathbb{R}^{m+1,1} \tag{6}$$

1.5 Gradient with respect to w_h

In the case of w_h, the situation is slightly different, as we are differentiating with respect to matrix. Strictly speaking, we should not talk about a gradient. However, the concept extends naturally to matrices, and we are still looking for the direction of maximal decrease of the objective function.

Since w_h is a matrix, we can not apply the chain rule directly. However, we can differentiate with respect to each column c of w_h, indexed by n_c, consider the mapping $c \mapsto l$, apply the chain rule and combine the results afterwards.

$$\frac{\partial l}{\partial c} = \frac{\partial l}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial pre_y} \frac{\partial pre_y}{\partial h} \frac{\partial h}{\partial pre_h} \frac{\partial pre_h}{\partial c} \tag{7}$$

We already know $\frac{\partial l}{\partial \hat{y}} \in \mathbb{R}$ and $\frac{\partial \hat{y}}{\partial pre_y} \in \mathbb{R}$.

In order to compute $\frac{\partial pre_y}{\partial h}$, we need to remember that $h \in \mathbb{R}^m$. This mapping does not involve the last component of θ. its jacobien is equal to $\tilde{\theta} = \theta[:-1]^T \in \mathbb{R}^{1,m}$.

We have that $\frac{\partial h}{\partial pre_h}$ is a diagonal matrix $D \in \mathbb{R}^{m,m}$ containing the vector $\sigma'(pre_h)$ on the diagonal.

The column c contains $d+1$ elements. The jacobian of $c \mapsto pre_h$ is the matrix $M_{n_c} \in \mathbb{R}^{m,d+1}$, containing $\bar{x}$ in line n_c, and 0's everywhere else.

Finally, we obtain that

$$\frac{\partial l}{\partial c} = (\hat{y} - y)\sigma'(pre_y)\tilde{\theta}DM_{n_c} \in \mathbb{R}^{1,d+1} \tag{8}$$

We note that $\tilde{\theta}D = u \in \mathbb{R}^{1,m}$ is a vector containing the elementwise products between $\tilde{\theta}$ and $\sigma'(pre_h)$, and we remark that

$$\frac{\partial l}{\partial c} = (\hat{y} - y)\sigma'(pre_y)u_{n_c}\bar{x} \in \mathbb{R}^{1,d+1} \tag{9}$$

Then, by stacking the jacobians with respect to the columns of w_h, we consider the jacobian $\frac{\partial l}{\partial w_h}$ to be :

$$\frac{\partial l}{\partial w_h} = (\hat{y} - y)\sigma'(pre_y)u^T\bar{x} \in \mathbb{R}^{m,d+1} \tag{10}$$

By transposition, the gradient writes

$$\nabla_c l(w_h) = (\hat{y} - y)\sigma'(pre_y)\bar{x}^T u \in \mathbb{R}^{d+1,m} \tag{11}$$