Deep Learning Assignment 1

Kevin de Vries (10579869)
University of Amsterdam
kevin.devries@student.uva.nl

Question 1

Question 1.1 a)

Gradient of loss w.r.t $x^{(N)}$:

$$\frac{\partial L}{\partial x_i^{(N)}} = \frac{\partial}{\partial x_i^{(N)}} \left(-\sum_j t_j \log(x_i^{(N)}) \right) = -\sum_j t_j \frac{\delta_{ij}}{x_i^{(N)}} = -\frac{t_i}{x_i} \tag{1}$$

Gradient of $x^{(N)}$ w.r.t $\tilde{x}^{(N)}$:

$$\begin{aligned}
\frac{\partial x_i^{(N)}}{\partial \tilde{x}_j^{(N)}} &= \frac{\partial}{\partial \tilde{x}_j^{(N)}} \left(\text{softmax}(\tilde{x}_i^{(N)}) \right) = \frac{\partial}{\partial \tilde{x}_j^{(N)}} \left(\frac{\exp(\tilde{x}_i^{(N)})}{\sum_k \exp(\tilde{x}_k^{(N)})} \right) \\
&= \frac{\partial}{\partial \tilde{x}_j^{(N)}} \left(\exp(\tilde{x}_i^{(N)}) \right) \frac{1}{\sum_k \exp(\tilde{x}_k^{(N)})} + \frac{\partial}{\partial \tilde{x}_j^{(N)}} \left(\frac{1}{\sum_k \exp(\tilde{x}_k^{(N)})} \right) \exp(\tilde{x}_i^{(N)}) \\
&= \frac{\delta_{ij} \exp(\tilde{x}_i^{(N)})}{\sum_k \exp(\tilde{x}_k^{(N)})} - \frac{\exp(\tilde{x}_i^{(N)}) \exp(\tilde{x}_j^{(N)})}{\left(\sum_k \exp(\tilde{x}_k^{(N)}) \right)^2} = \text{softmax}(\tilde{x}_i^{(N)}) \left(\delta_{ij} - \text{softmax}(\tilde{x}_j^{(N)}) \right)
\end{aligned} \tag{2}$$

Gradient of $x^{(l<N)}$ w.r.t $\tilde{x}^{(l<N)}$:

$$\frac{\partial x_i^{(l<N)}}{\partial \tilde{x}_j^{(l<N)}} = \frac{\partial}{\partial \tilde{x}_j^{(l<N)}} \left(\max(0, \tilde{x}_i^{(l<N)}) \right) = \delta_{ij} \mathbb{I}(\tilde{x}_i^{(l<N)} > 0) = \begin{cases} \delta_{ij} & \text{if } \tilde{x}_i^{(l<N)} > 0 \\ 0 & \text{if } x_i^{(l<N)} \leq 0 \end{cases} \tag{3}$$

Gradient of $\tilde{x}^{(l)}$ w.r.t $x^{(l-1)}$:

$$\frac{\partial \tilde{x}_i^{(l)}}{\partial x_j^{(l-1)}} = \frac{\partial}{\partial x_j^{(l-1)}} \left(\sum_n W_{in}^{(l)} x_n^{(l-1)} + b_i^{(l)} \right) = W_{ij}^{(l)} \tag{4}$$

Gradient of $\tilde{x}^{(l)}$ w.r.t $W^{(l)}$:

32nd Conference on Neural Information Processing Systems (NIPS 2018), Montréal, Canada.

$$\frac{\partial \tilde{x}_i^{(l)}}{\partial W_{jk}^{(l)}} = \frac{\partial}{\partial W_{jk}^{(l)}} \left(\sum_n W_{in}^{(l)} x_n^{(l-1)} + b_i^{(l)} \right) = \delta_{ij} x_k^{(l-1)} \tag{5}$$

Gradient of $\tilde{x}^{(l)}$ w.r.t $b^{(l)}$:

$$\frac{\partial \tilde{x}_i^{(l)}}{\partial b_j^{(l)}} = \frac{\partial}{\partial b_j^{(l)}} \left(\sum_n W_{in}^{(l)} x_n^{(l-1)} + b_i^{(l)} \right) = \delta_{ij} \tag{6}$$

Question 1.1 b)

Gradient of loss w.r.t $\tilde{x}^{(N)}$:

$$\begin{aligned} \frac{\partial L}{\partial \tilde{x}_j^{(N)}} &= \sum_i \frac{\partial L}{\partial x_i^{(N)}} \frac{\partial x_i^{(N)}}{\partial \tilde{x}_j^{(N)}} = \sum_i \frac{\partial L}{\partial x_i^{(N)}} \text{softmax}(\tilde{x}_i^{(N)}) \left(\delta_{ij} - \text{softmax}(\tilde{x}_j^{(N)}) \right) \\ \Rightarrow \frac{\partial L}{\partial \tilde{x}^{(N)}} &= \frac{\partial L}{\partial x^{(N)}} \odot \text{softmax}(\tilde{x}^{(N)}) - \text{softmax}(\tilde{x}^{(N)}) \text{softmax}(\tilde{x}^{(N)})^T \frac{\partial L}{\partial x^{(N)}} \end{aligned} \tag{7}$$

Gradient of loss w.r.t $\tilde{x}^{(l<N)}$:

$$\begin{aligned} \frac{\partial L}{\partial \tilde{x}_j^{(l<N)}} &= \sum_i \frac{\partial L}{\partial x_i^{(l)}} \frac{\partial x_i^{(l)}}{\partial \tilde{x}_j^{(l)}} = \sum_i \frac{\partial L}{\partial x_i^{(l)}} \delta_{ij} \mathbb{I}(\tilde{x}_i^{(l)} > 0) = \frac{\partial L}{\partial x_j^{(l)}} \mathbb{I}(\tilde{x}_j^{(l)} > 0) \\ \Rightarrow \frac{\partial L}{\partial \tilde{x}^{(l<N)}} &= \frac{\partial L}{\partial x^{(l)}} \odot \mathbb{I}(\tilde{x}^{(l)} > 0) \end{aligned} \tag{8}$$

Gradient of loss w.r.t $x^{(l<N)}$:

$$\begin{aligned} \frac{\partial L}{\partial x_j^{(l<N)}} &= \sum_i \frac{\partial L}{\partial \tilde{x}_i^{(l+1)}} \frac{\partial \tilde{x}_i^{(l+1)}}{\partial x_j^{(l)}} = \sum_i \frac{\partial L}{\partial \tilde{x}_i^{(l+1)}} W_{ij}^{(l+1)} \\ \Rightarrow \frac{\partial L}{\partial x^{(l<N)}} &= (W^{(l+1)})^T \frac{\partial L}{\partial \tilde{x}^{(l+1)}} \end{aligned} \tag{9}$$

Gradient of loss w.r.t $W^{(l)}$:

$$\begin{aligned} \frac{\partial L}{\partial W_{jk}^{(l)}} &= \sum_i \frac{\partial L}{\partial \tilde{x}_i^{(l)}} \frac{\partial \tilde{x}_i^{(l)}}{\partial W_{jk}^{(l)}} = \sum_i \frac{\partial L}{\partial \tilde{x}_i^{(l)}} \delta_{ij} x_k^{(l-1)} = \frac{\partial L}{\partial \tilde{x}_j^{(l)}} x_k^{(l-1)} \\ \Rightarrow \frac{\partial L}{\partial W^{(l)}} &= \frac{\partial L}{\partial \tilde{x}^{(l)}} (x^{(l-1)})^T \end{aligned} \tag{10}$$

Gradient of loss w.r.t $b^{(l)}$:

$$\begin{aligned} \frac{\partial L}{\partial b_j^{(l)}} &= \sum_i \frac{\partial L}{\partial \tilde{x}_i^{(l)}} \frac{\partial \tilde{x}_i^{(l)}}{\partial b_j^{(l)}} = \sum_i \frac{\partial L}{\partial \tilde{x}_i^{(l)}} \delta_{ij} = \frac{\partial L}{\partial \tilde{x}_j^{(l)}} \\ \Rightarrow \frac{\partial L}{\partial b^{(l)}} &= \frac{\partial L}{\partial \tilde{x}^{(l)}} \end{aligned} \tag{11}$$

Question 1.1 c)

Since with $B \neq 1$ we have a total loss of

$$L_{total}(\{x^{(0),s}, t^s\}_{s=1}^B) = \frac{1}{B}\sum_{s=1}^{B} L(x^{(0),s}, t^s),$$

the gradient of the total loss w.r.t any input or activation $a^{(l),r}$ is

$$\frac{\partial}{\partial a^{(l),r}}\left(L_{total}(\{x^{(0),s}, t^s\}_{s=1}^B)\right) = \frac{1}{B}\frac{\partial}{\partial a^{(l),r}}\left(\sum_{s=1}^{B} L(x^{(0),s}, t^s)\right) = \frac{1}{B}\frac{\partial}{\partial a^{(l),r}}\left(L(x^{(0),r}, t^r)\right)$$

and the gradient of the total loss w.r.t a sample independent model parameter $\theta^{(l)}$ is

$$\frac{\partial}{\partial \theta^{(l)}}\left(L_{total}(\{x^{(0),s}, t^s\}_{s=1}^B)\right) = \frac{1}{B}\sum_{s=1}^{B}\frac{\partial}{\partial \theta^{(l)}}\left(L(x^{(0),s}, t^s)\right),$$

Which is equal to the sample mean of the gradients in the batch w.r.t $\theta^{(l)}$. In practice, due to the chain rule, the $\frac{1}{B}$ term only has to be applied explicitly at the backward pass of the loss module during back-propagation.

Question 1.2

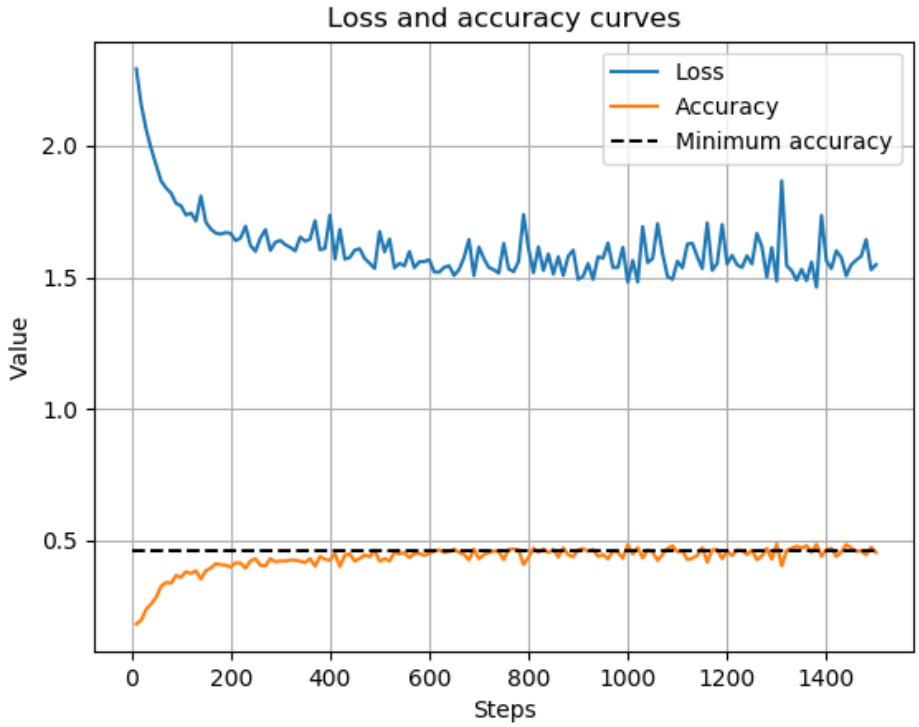

Figure 1: The loss and accuracy curves of the numpy implementation of the MLP for Question 1.2.

In Figure 1 we find the loss and accuracy curves of the numpy implementation of the MultiLayer Perceptron (MLP) using the default parameters. The MLP reaches the desired accuracy of the question.

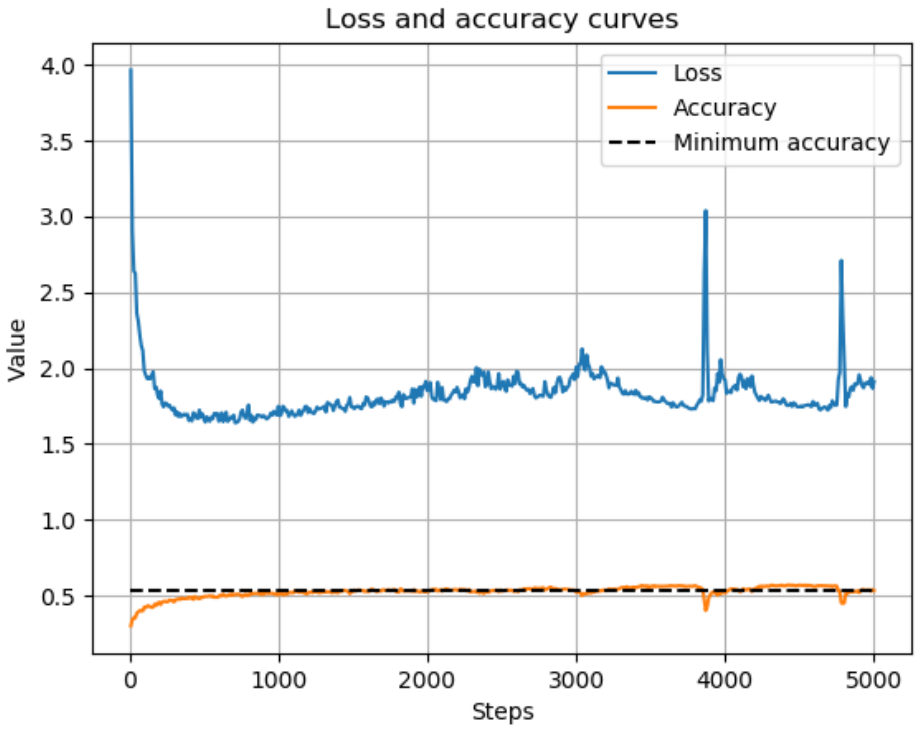

Figure 2: The loss and accuracy curves of the pytorch implementation of the MLP for Question 2.

Question 2

In Figure 2 we find the loss and accuracy curves of the pytorch implementation of the MultiLayer Perceptron (MLP). The MLP reaches the desired accuracy of the question using a relatively wide network with two hidden layers with a 1000 hidden units each. The optimizer used to reach the desired accuracy is the Adam optimizer with a weight decay of 0.01, betas of 0.75 and 0.999 and a learning rate of 0.0001. The maximum accuracy reached with these parameters was 0.5703. The training was done for 5000 steps with a batch size of 500. It was observed that the most important parameter to gain higher accuracies is the number of hidden layers and hidden units per layer. Wider networks with two hidden layers seemed to gain the best accuracies.

Question 3

From all subquestions in Question 3 only Question 3.2a is relevant for the report. The rest of Question 3 is present in the code folder.

Question 3.2 a)

Gradient of loss w.r.t γ:

$$\frac{\partial L}{\partial \gamma_j} = \sum_s \sum_i \frac{\partial L}{\partial y_i^s} \frac{\partial y_i^s}{\partial \gamma_j} = \sum_s \sum_i \frac{\partial L}{\partial y_i^s} \delta_{ij} \hat{x}_i^s = \sum_s \frac{\partial L}{\partial y_j^s} \hat{x}_j^s \tag{12}$$

Gradient of loss w.r.t β:

$$\frac{\partial L}{\partial \beta_j} = \sum_s \sum_i \frac{\partial L}{\partial y_i^s} \frac{\partial y_i^s}{\partial \beta_j} = \sum_s \sum_i \frac{\partial L}{\partial y_i^s} \delta_{ij} = \sum_s \frac{\partial L}{\partial y_j^s} \tag{13}$$

Gradient of loss w.r.t x:

$$\frac{\partial \mu_i}{\partial x_j^r} = \frac{\partial}{\partial x_j^r} \left(\frac{1}{B} \sum_s x_i^s \right) = \frac{1}{B} \sum_s \delta_{ij} \delta_{rs} = \frac{\delta_{ij}}{B} \tag{14}$$

$$\Rightarrow \frac{\partial \sigma_i^2}{\partial x_j^r} = \frac{\partial}{\partial x_j^r} \left(\frac{1}{B} \sum_s (x_i^s - \mu_i)^2 \right) \tag{15}$$

$$= \frac{2}{B} \sum_s (x_i^s - \mu_i) \left(\delta_{ij} \delta_{rs} - \frac{\partial \mu_i}{\partial x_j^r} \right) = \frac{2}{B} \sum_s (x_i^s - \mu_i) \left(\delta_{ij} \delta_{rs} - \frac{\delta_{ij}}{B} \right) \tag{16}$$

$$\Rightarrow \frac{\partial}{\partial x_j^r} \left(\frac{1}{\sqrt{\sigma_i^2 + \epsilon}} \right) = -\frac{1}{2} (\sigma_i^2 + \epsilon)^{-\frac{2}{3}} \frac{2}{B} \sum_s (x_i^s - \mu_i) \left(\delta_{ij} \delta_{rs} - \frac{\delta_{ij}}{B} \right) \tag{17}$$

$$= -\frac{(\sigma_i^2 + \epsilon)^{-\frac{2}{3}}}{B} (x_i^r - \mu_i) \delta_{ij} \tag{18}$$

$$\Rightarrow \frac{\partial \hat{x}_i^s}{\partial x_j^r} = \frac{\partial}{\partial x_j^r} \left(\frac{x_i^s}{\sqrt{\sigma_i^2 + \epsilon}} \right) - \frac{\partial}{\partial x_j^r} \left(\frac{\mu_i}{\sqrt{\sigma_i^2 + \epsilon}} \right) \tag{19}$$

$$= (x_i^s - \mu_i) \frac{\partial}{\partial x_j^r} \left(\frac{1}{\sqrt{\sigma_i^2 + \epsilon}} \right) + \frac{1}{\sqrt{\sigma_i^2 + \epsilon}} \left(\delta_{ij} \delta_{rs} - \frac{\partial \mu_i}{\partial x_j^r} \right) \tag{20}$$

$$= -(x_i^s - \mu_i) \frac{(\sigma_i^2 + \epsilon)^{-\frac{2}{3}}}{B} (x_i^r - \mu_i) \delta_{ij} + \frac{1}{\sqrt{\sigma_i^2 + \epsilon}} \left(\delta_{ij} \delta_{rs} - \frac{\delta_{ij}}{B} \right) \tag{21}$$

$$= \frac{\delta_{ij}}{\sqrt{\sigma_i^2 + \epsilon}} \left(\delta_{rs} - \frac{1}{B} \right) - \frac{1}{B} (x_i^s - \mu_i) (x_i^r - \mu_i) \delta_{ij} (\sigma_i^2 + \epsilon)^{-\frac{2}{3}} \tag{22}$$

$$\Rightarrow \frac{\partial L}{\partial x_j^r} = \sum_s \sum_i \frac{\partial L}{\partial y_i^s} \frac{\partial y_i^s}{\partial \hat{x}_i^s} \frac{\partial \hat{x}_i^s}{\partial x_j^r} = \sum_s \sum_i \frac{\partial L}{\partial y_i^s} \gamma_i \frac{\partial \hat{x}_i^s}{\partial x_j^r} \tag{23}$$

$$= \sum_s \sum_i \frac{\partial L}{\partial y_i^s} \gamma_i \left(\frac{\delta_{rs} - \frac{1}{B}}{\sqrt{\sigma_i^2 + \epsilon}} - \frac{(x_i^s - \mu_i) (x_i^r - \mu_i)}{B (\sigma_i^2 + \epsilon)^{\frac{2}{3}}} \right) \delta_{ij} \tag{24}$$

$$= \frac{\gamma_j}{\sqrt{\sigma_j^2 + \epsilon}} \left(\frac{\partial L}{\partial y_j^r} - \frac{1}{B} \sum_s \frac{\partial L}{\partial y_j^s} - \frac{\hat{x}_j^r}{B} \sum_s \hat{x}_j^s \frac{\partial L}{\partial y_j^s} \right) \tag{25}$$

Question 4

Although I have implemented the ConvNet and got an accuracy of around 0.75 using the default parameters, I did not manage to get a plot of the loss and accuracy curves in time.