Exercises for the course
Deep Learning 1
Winter Semester 2024/25

Machine Learning Group
Faculty IV – Electrical Engineering and Computer Science
Technische Universität Berlin

Exercise Sheet 7

Exercise 1: Backpropagation for Probabilistic Regression (20 + 20 + 20 + 20 + 20 P)

Let μ and σ be the output of a neural network predicting for a given input $\boldsymbol{x}$ a conditional distribution over targets $t|\boldsymbol{x}$. Specifically, these outputs define the function:

$$f(\mu, \sigma) = -\frac{(t-\mu)^2}{2\sigma^2} - \log\left(\sqrt{2\pi}\sigma\right)$$

which quantifies how likely it is (assuming a Gaussian distribution of parameters μ and σ) to observe the target t.

(a) *Compute* the gradient of the function f with respect to the outputs μ and σ of the neural network.

(b) Gradient descent of the function f can be unstable when σ is small. We would like to stabilize it by letting the network have an output z (instead of σ), and produce σ through the nonlinear activation function $\sigma(z) = \sqrt{1+z^2}$. Note that such an activation function forces the standard deviation to be at least 1. State the chain rule for computing the derivative $\partial f / \partial z$, and develop it to arrive at the final form:

$$\frac{\partial f}{\partial z} = \left(\frac{(t-\mu)^2}{\sigma^4} - \frac{1}{\sigma^2}\right) \cdot z$$

(c) *Show* that the magnitude of the gradient you have derived can be upper-bounded by a quadratic function of the neural network output, specifically:

$$\left|\frac{\partial f}{\partial z}\right| \leq (t-\mu)^2 + 1$$

From this inequality, one can see that there is no major instability to be expected during gradient descent, as long as the mean is set initially and remains reasonably close to the target values. *(Hint: To derive this result, you can observe that $\sigma \geq 1$ and $\sigma \geq |z|$).*

(d) Assume we have a dataset composed of inputs $\boldsymbol{x}_1, \dots, \boldsymbol{x}_N$ and associated targets $t_1, \dots, t_N$. The function to optimize becomes the sum of log-likelihoods and is given by:

$$J(\theta) = \frac{1}{N} \sum_{i=1}^{N} \left(-\frac{|t_i - \mu_i|^2}{2\sigma_i^2} - \log\left(\sqrt{2\pi}\sigma_i\right)\right)$$

where μ_i and σ_i forms the neural network prediction associated to data point $(\boldsymbol{x}_i, t_i)$. Note that each of these predictions can be expressed as a function of the collection of parameters of the neural network, which we denote by θ. *State* the (multivariate) chain rule that allow you to compute $\partial J / \partial \theta$ from the calculations $\partial f / \partial \mu$ and $\partial f / \partial z$ you have already performed above.

(e) We now assume that the neural network is made of a single linear layer with parameters $\theta = (\boldsymbol{w}, \boldsymbol{v})$, and whose outputs are given by the computations $\mu = \boldsymbol{w}^\top \boldsymbol{x}$ and $z = \boldsymbol{v}^\top \boldsymbol{x}$. Develop the objective's gradient, i.e. calculate $\partial J / \partial \boldsymbol{w}$ and $\partial J / \partial \boldsymbol{v}$.

Exercise 1: Backpropagation for Probabilistic Regression (20 + 20 + 20 + 20 + 20 P)

Let μ and σ be the output of a neural network predicting for a given input $\boldsymbol{x}$ a conditional distribution over targets $t|\boldsymbol{x}$. Specifically, these outputs define the function:

$$f(\mu, \sigma) = -\frac{(t-\mu)^2}{2\sigma^2} - \log\left(\sqrt{2\pi}\sigma\right)$$

which quantifies how likely it is (assuming a Gaussian distribution of parameters μ and σ) to observe the target t.

(a) *Compute* the gradient of the function f with respect to the outputs μ and σ of the neural network.

$$\frac{\partial}{\partial \mu} f(\mu, \sigma) = + \frac{2(t-\mu)}{2\sigma^2} - 0 = \frac{t-\mu}{\sigma^2}$$

$$\frac{\partial}{\partial \sigma} f(\mu, \sigma) = -\frac{1}{2}(t-\mu)^2 \cdot (-2)\,\sigma^{-3} - \frac{\sqrt{2\pi}}{\sqrt{2\pi}\cdot\sigma} = \frac{(t-\mu)^2}{\sigma^3} - \frac{1}{\sigma}$$

$$\nabla f = \begin{bmatrix} \frac{\partial f}{\partial \mu} \\ \frac{\partial f}{\partial \sigma} \end{bmatrix} = \begin{bmatrix} \frac{t-\mu}{\sigma^2} \\ \frac{(t-\mu)^2}{\sigma^3} - \frac{1}{\sigma} \end{bmatrix}$$

(b) Gradient descent of the function f can be unstable when σ is small. We would like to stabilize it by letting the network have an output z (instead of σ), and produce σ through the nonlinear activation function $\sigma(z) = \sqrt{1+z^2}$. Note that such an activation function forces the standard deviation to be at least 1. State the chain rule for computing the derivative $\partial f/\partial z$, and develop it to arrive at the final form:

$$\frac{\partial f}{\partial z} = \Big(\frac{(t-\mu)^2}{\sigma^4} - \frac{1}{\sigma^2}\Big) \cdot z$$

$$\frac{\partial f}{\partial z} = \frac{\partial f}{\partial \sigma} \cdot \frac{\partial \sigma}{\partial z} = \left(\frac{(t-\mu)^2}{\sigma^3} - \frac{1}{\sigma}\right) \cdot \frac{1}{\cancel{2}} \cdot \frac{\cancel{2}z}{\sqrt{1+z^2}}$$

$$= \left(\frac{(t-\mu)^2}{\sigma^3} - \frac{1}{\sigma}\right) \cdot \frac{z}{\sigma}$$

$$= \left(\frac{(t-\mu)^2}{\sigma^4} - \frac{1}{\sigma^2}\right) \cdot z$$

(c) *Show* that the magnitude of the gradient you have derived can be upper-bounded by a quadratic function of the neural network output, specifically:

$$\left|\frac{\partial f}{\partial z}\right| \leq (t-\mu)^2 + 1$$

From this inequality, one can see that there is no major instability to be expected during gradient descent, as long as the mean is set initially and remains reasonably close to the target values. *(Hint: To derive this result, you can observe that $\sigma \geq 1$ and $\sigma \geq |z|$).*

$$\sigma \geq 1 \iff \frac{1}{\sigma} \leq 1 \iff \frac{1}{\sigma^3} \leq 1$$

$$\left|\frac{\partial f}{\partial z}\right| = \left|\left(\frac{(t-\mu)^2}{\sigma^4} - \frac{1}{\sigma^2}\right) \cdot z\right|$$

$$= \left|\left(\frac{(t-\mu)^2}{\sigma^3} - \frac{1}{\sigma}\right) \frac{z}{\sigma}\right|$$

$$= \left|\Big(\frac{(t-\mu)^2}{\sigma^3} - \frac{1}{\sigma}\Big] \cdot \frac{\cancel{\sigma}}{\cancel{\sigma}}\right| \qquad \Big|\ \sigma \geq |z|$$

$$= \left| \frac{(t-\mu)^2}{\delta^3} - \frac{1}{\delta} \right|$$

$$\leq \left| \frac{(t-\mu)^2}{\delta^3} \right| + \left| \frac{1}{\delta} \right|$$

$$\leq \left| (t-\mu)^2 \right| + \left| \frac{1}{\delta} \right| \qquad \Big| \ \frac{1}{\delta^3} \leq 1$$

$$\leq (t-\mu)^2 + 1 \qquad \Big| \ \frac{1}{\delta} \leq 1$$

(d) Assume we have a dataset composed of inputs $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$ and associated targets $t_1, \ldots, t_N$. The function to optimize becomes the sum of log-likelihoods and is given by:

$$J(\theta) = \frac{1}{N} \sum_{i=1}^{N} \Big(\underbrace{- \frac{|t_i - \mu_i|^2}{2\sigma_i^2} - \log\left(\sqrt{2\pi}\sigma_i\right)}_{=f} \Big)$$

where μ_i and σ_i forms the neural network prediction associated to data point $(\boldsymbol{x}_i, t_i)$. Note that each of these predictions can be expressed as a function of the collection of parameters of the neural network, which we denote by θ. *State* the (multivariate) chain rule that allow you to compute $\partial J / \partial \theta$ from the calculations $\partial f / \partial \mu$ and $\partial f / \partial z$ you have already performed above.

$$\frac{\partial J}{\partial \theta} = \frac{1}{N} \cdot \sum_{i=1}^{N} \left(\frac{\partial f_i}{\partial \mu_i} \cdot \frac{\partial \mu_i}{\partial \theta} + \frac{\partial f_i}{\partial z_i} \cdot \frac{\partial z_i}{\partial \theta} \right)$$

(e) We now assume that the neural network is made of a single linear layer with parameters $\theta = (\boldsymbol{w}, \boldsymbol{v})$, and whose outputs are given by the computations $\mu = \boldsymbol{w}^\top \boldsymbol{x}$ and $z = \boldsymbol{v}^\top \boldsymbol{x}$. Develop the objective's gradient, i.e. calculate $\partial J / \partial \boldsymbol{w}$ and $\partial J / \partial \boldsymbol{v}$.

$$\frac{\partial J}{\partial w} = \frac{1}{N} \cdot \sum_{i=1}^{N} \left(\frac{\partial f}{\partial \mu_i} \cdot \frac{\partial \mu_i}{\partial w} \right)$$

$$= \frac{1}{N} \cdot \sum_{i=1}^{N} \left(\frac{t - \mu_i}{\delta_i^2} \cdot \frac{\partial \mu_i}{\partial w} \right)$$

$$= \frac{1}{N} \cdot \sum_{i=1}^{N} \left(\frac{t - \mu_i}{\delta_i^2} \cdot x \right)$$

$$\frac{\partial J}{\partial v} = \frac{1}{N} \cdot \sum_{i=1}^{N} \left(\frac{\partial f}{\partial z_i} \cdot \frac{\partial z_i}{\partial v} \right)$$

$$= \frac{1}{N} \cdot \sum_{i=1}^{N} \left(\left(\frac{(t-\mu_i)^2}{\delta_i^4} - \frac{1}{\delta_i^2} \right) \cdot z_i \cdot \frac{\partial \mu_i}{\partial v} \right)$$

$$= \frac{1}{N} \cdot \sum_{i=1}^{N} \left(\left(\frac{(t-\mu_i)^2}{\delta_i^4} - \frac{1}{\delta_i^2} \right) \cdot z_i \cdot x \right)$$