DEEP LEARNING: ASSIGNMENTS WITH SOLUTIONS

Assignment 1 (Gradient Verification in Lab 2)**.** Let $\mathcal{L}$ be the loss function, depending on the parameter w and let $J = \frac{\mathrm{d}\mathcal{L}}{\mathrm{d}w}$ be the derivative of $\mathcal{L}$ in w.

a) Let Δw be a (random) vector of length ε and $\Delta\mathcal{L} = \mathcal{L}(w + \Delta w) - \mathcal{L}(w)$. Show that the (correctly computed) derivative must satisfy

$$\big|\Delta\mathcal{L} - \langle J, \Delta w\rangle\big| \ll \varepsilon. \tag{1}$$

b) Assume that $\mathcal{L}$ is twice differentiable and let $\Delta\mathcal{L} = \frac{1}{2}(\mathcal{L}(w + \Delta w) - \mathcal{L}(w - \Delta w))$. Show that the derivative in this case must satisfy even a stronger condition

$$\big|\Delta\mathcal{L} - \langle J, \Delta w\rangle\big| \ll \varepsilon^2. \tag{2}$$

Conclude that this condition is easier to check with limited numerical accuracy.

Solution.

a) By definition of derivative, there must hold

$$\mathcal{L}(w + \Delta w) = \mathcal{L}(w) + J\Delta w + o(\|\Delta w\|). \tag{3}$$

Since $\mathcal{L}$ is a scalar-valued function J is a row vector and $J\Delta w = \langle J, \Delta w\rangle$. We can express

$$\langle J, \Delta w\rangle = \mathcal{L}(w + \Delta w) - \mathcal{L}(w) + o(\|\Delta w\|). \tag{4}$$

Denoting $\Delta\mathcal{L} = \mathcal{L}(w + \Delta w) - \mathcal{L}(w)$ (as in the assignment), there must hold

$$|\langle J, \Delta w\rangle - \Delta\mathcal{L}| = o(\|\Delta w\|) = o(\varepsilon), \tag{5}$$

which is equivalent to

$$|\langle J, \Delta w\rangle - \Delta\mathcal{L}| \ll \varepsilon. \tag{6}$$

b) Since $\mathcal{L}$ is twice differentiable, we can write its second order Taylor expansion about w:

$$\mathcal{L}(w + \Delta w) = \mathcal{L}(w) + \langle J, \Delta w\rangle + \frac{1}{2}\langle \Delta w, H\Delta w\rangle + o(\|\Delta w\|^2), \tag{7}$$

where H is the Hessian matrix. Consider now the displacement $-\Delta w$, the second order expansion for it reads:

$$\mathcal{L}(w - \Delta w) = \mathcal{L}(w) - \langle J, \Delta w\rangle + \frac{1}{2}\langle \Delta w, H\Delta w\rangle + o(\|\Delta w\|^2). \tag{8}$$

Note that the sign of quadratic form $\langle \Delta w, H\Delta w\rangle$ has not changed. Subtracting these two expansions we obtain:

$$\mathcal{L}(w + \Delta w) - \mathcal{L}(w - \Delta w) = 2\langle J, \Delta w\rangle + o(\|\Delta w\|^2). \tag{9}$$

Rearranging and denoting $\Delta\mathcal{L} = \frac{1}{2}(\mathcal{L}(w + \Delta w) - \mathcal{L}(w - \Delta w))$, we obtain

$$(\langle J, \Delta w\rangle - \Delta\mathcal{L}) = o(\|\Delta w\|^2), \tag{10}$$

which is equivalent to

$$\big|\langle J, \Delta w\rangle - \Delta\mathcal{L}\big| \ll \varepsilon^2. \tag{11}$$

□

Assignment 2 (Backprop normalized linear)**.**
Let $x \in \mathbb{R}^n$. Consider the following normalized linear layer (known as "weight normalization"):

$$y_i = \frac{w_i^{\mathsf{T}} x + b_i}{\|w_i\|},$$

where $w_i \in \mathbb{R}^n$ for $i = 1 \dots m$, $b_i \in \mathbb{R}$ and $\|w_i\|$ is the Euclidean norm of vector w_i. Given the gradient of the loss function in y, $g := \nabla_y \mathcal{L} \in \mathbb{R}^m$, compute gradients of the loss in w, b, x.

Solution. We will use the total derivative rule

$$\frac{\mathrm{d}\mathcal{L}}{\mathrm{d}\theta} = \sum_i \frac{\mathrm{d}\mathcal{L}}{\mathrm{d}y_i}\frac{\partial y_i}{\partial \theta} = \sum_i g_i \frac{\partial y_i}{\partial \theta}. \tag{12}$$

Since y_i depends only on b_i and not on b_j for $j \neq i$ for $\nabla_b \mathcal{L}$ we have

$$\frac{\mathrm{d}\mathcal{L}}{\mathrm{d}b_i} = g_i \frac{\partial y_i}{\partial b_i} = \frac{g_i}{\|w_i\|}. \tag{13}$$

For $\nabla_x \mathcal{L}$ we have

$$\frac{\mathrm{d}\mathcal{L}}{\mathrm{d}x_j} = \sum_i g_i \frac{\partial y_i}{\partial x_j} = \sum_i g_i \frac{w_{ij}}{\|w_i\|}. \tag{14}$$

Since y_i depends only on w_i and not on w_j for $j \neq i$ for $\nabla_w \mathcal{L}$ we have

$$\frac{\mathrm{d}L}{\mathrm{d}w_i} = \sum_i g_i \frac{\partial y_i}{\partial w_i} = \sum_i g_i \Big(\frac{x}{\|w_i\|} + (w_i^{\mathsf{T}} x + b_i)\frac{-w_i}{\|w_i\|^3}\Big). \tag{15}$$

□

Assignment 3 (Backprop recurrent sequence)**.**
Let $x \in \mathbb{R}^N$ be a vector with components x_i for $i = 1, \dots N$ and consider a layer performing the following computation:

$$y_i = a(x_i + x_{i+2}) + b \quad \text{for } i = 1 \dots N - 2. \tag{16}$$

Given the gradient of the loss function in y, $g := \nabla_y \mathcal{L} \in \mathbb{R}^{N-2}$, compute the gradient of the loss in a, b and x.

Solution.

$$\frac{\mathrm{d}\mathcal{L}}{\mathrm{d}b} = \sum_{i=1}^{N-2} \frac{\mathrm{d}\mathcal{L}}{\mathrm{d}y_i}\frac{\partial y_i}{\partial b} = \sum_{i=1}^{N-2} \frac{\partial \mathcal{L}}{\partial y_i} = \sum_{i=1}^{N-2} g_i. \tag{17}$$

$$\frac{\mathrm{d}\mathcal{L}}{\mathrm{d}a} = \sum_{i=1}^{N-2} \frac{\mathrm{d}\mathcal{L}}{\mathrm{d}y_i}\frac{\partial y_i}{\partial a} = \sum_{i=1}^{N-2} g_i(x_i + x_{i+2}). \tag{18}$$

$$\frac{\mathrm{d}\mathcal{L}}{\mathrm{d}x_j} = \sum_{i=1}^{N-2} g_i \frac{\partial y_i}{\partial x_j} = \sum_{i=1}^{N-2} g_i a\big([\![j{=}i]\!] + [\![j{=}i{+}2]\!]\big) = \begin{cases} a g_j & \text{if } j \leq 2, \\ a(g_j + g_{j-2}) & \text{if } j = 2, \dots N-2, \\ a g_{j-2} & \text{if } j \geq N-2. \end{cases} \tag{19}$$

□

Assignment 4 (Stochastic Gradient Quantization). Sometimes randomized procedures are used to quantize the gradients for a faster communication in a distributed system (if we want to parallelize training).
Let the gradient $g \in \mathbb{R}^n$ be computed at the worker. The worker can sends a *quantized* gradient $\tilde{g} \in \{0,1\}^n$ to the server, using only 1 bit per coordinate. The worker additionally sends two real numbers to the server a, b and the server reconstructs the gradient as $a\tilde{g}+b$. How to chose the quantization procedure in a randomized way so that $\mathbb{E}[a\tilde{g} + b] = g$ and hence we preserve the guarantee of an unbiased (but more noisy) gradient estimate? Is the choice of a and b satisfying this assumption unique? How to choose a and b such that $\mathbb{E}[a\tilde{g} + b] = g$ and the variance of $a\tilde{g} + b$ is minimal?

Solution. Clearly, given g_i, with a deterministic choice of $\tilde{g}_i \in \{0,1\}$ we cannot achieve the property $\mathbb{E}[a\tilde{g} + b] = g$ for all coordinates and would have a systematic error. Let us choose $\tilde{g}_i \in \{0,1\}$ at random, with probability $\mathbb{P}(\tilde{g}_i{=}1) = \beta_i$. We then have $\mathbb{E}[a\tilde{g}_i + b] = a\beta_i + b$ and can make all coordinates unbiased by setting

$$\beta_i = \frac{g_i - b}{a}, \tag{20}$$

however the probabilities β_i need to be in the range $[0,1]$ and therefore a and b must satisfy

$$0 \leq \frac{g_i - b}{a} \leq 1 \ \ \forall i. \tag{21}$$

Assuming that $a > 0$, it is equivalent to

$$b \leq g_i \leq a + b \ \ \forall i. \tag{22}$$

The choice of a and $b$is clearly non-unique: as long as $b \leq \min_i g_i =: m$ and $a + b \geq \max_i g_i =: M$, we can satisfy the expectation requirement.

Let us determine a and b that give the least variance to the estimate $a\tilde{g}_i + b$ for some fixed i. The variance of a Bernoulli variable with probability β_i is given by $\beta_i(1-\beta_i)$. The variance of $a\tilde{g}_i + b$ is respectively

$$a^2\big(\frac{g_i - b}{a}\big)\big(1 - \frac{g_i - b}{a}\big) = (g_i - b)(a + b - g_i). \tag{23}$$

To minimize this variance subject to the constraints on a and b we need to solve the problem

$$\min_{a,b}(g_i - b)(a + b - g_i) \quad \text{s.t.} \quad b \leq m;\ a + b \geq M. \tag{24}$$

Notice that in the objective both $(g_i - b)$ and $(a+b-g_i)$ are non-negative when constraints are satisfied. The first factor is minimized by choosing $b = m$. The second factor is minimized by choosing $a = M - b = M - m$. Notice that this solution does not depend on the particular coordinate i. Therefore variances of all components of the gradient are simultaneously minimized by this choice of a and b. □

Assignment 5 (SGD + L2)**.** Consider a regularized loss function $\tilde{L}(\theta) = L(\theta) + \frac{\lambda}{2}\|\theta\|^2$. Let $\tilde{g}$ be a stochastic gradient estimate of L at θ. Note that the regularization part of the objective, $\frac{\lambda}{2}\|\theta\|^2$, is known in a closed form and so its gradient g_r is non-stochastic.

a) Design an SGD algorithm that applies momentum (exponentially weighted averaging) to g only but not to g_r.

b) Is it equivalent to an SGD with the momentum applied to both g and g_r but possibly with a different settings of λ, momentum and learning rate?

Solution.

a) The gradient of the regularizer at θ^t is given by $g_r = \lambda\theta^t$. Let $\tilde{g}^t$ be stochastic gradient of $L(\theta)$ at θ^t: $\tilde{g}^t = \hat{\nabla}_\theta L(\theta^t)$. We will use the momentum form of SGD with EWA (lecture 4):

$$v^t = \mu v^{t-1} + \tilde{g}^t; \tag{25a}$$
$$\theta^{t+1} = \theta^t - \alpha(v^t + \lambda\theta^t), \tag{25b}$$

where α is the learning rate and μ is momentum.

b) If we apply the momentum to both $\tilde{g}$ and g_r, we obtain a seemingly different algorithm:

$$v'^t = \mu' v'^{t-1} + \tilde{g}^t + \lambda'\theta^t; \tag{26a}$$
$$\theta^{t+1} = \theta^t - \alpha' v'^t. \tag{26b}$$

The question is whether the first algorithm can be converted into the second one by choosing λ', α', μ' appropriately. To verify this, we will reduce each algorithm to a recurrent relation in main sequence θ^t only. In the algorithm (25) we have for two time steps:

$$\theta^{t+1} = \theta^t - \alpha(v^t + \lambda\theta^t); \tag{27a}$$
$$\theta^t = \theta^{t-1} - \alpha(v^{t-1} + \lambda\theta^{t-1}). \tag{27b}$$

Multiplying the second equation by μ and subtracting from the first we obtain

$$\theta^{t+1} - \mu\theta^t = \theta^t - \mu\theta^{t-1} - \alpha(\tilde{g}^t + \lambda\theta^t - \mu\lambda\theta^{t-1}). \tag{28}$$

Rearranging we get the recurrence:

$$\theta^{t+1} = (1 + \mu - \alpha\lambda)\theta^t - \mu(1 - \alpha\lambda)\theta^{t-1} - \alpha\tilde{g}^t \tag{29}$$

Similarly, in algorithm (26) two time steps express as:

$$\theta^{t+1} = \theta^t - \alpha' v'^t; \tag{30a}$$
$$\theta^t = \theta^{t-1} - \alpha' v'^{t-1}. \tag{30b}$$

Multiplying the second equation by μ' and subtracting from the first we obtain

$$\theta^{t+1} - \mu'\theta^t = \theta^t - \mu'\theta^{t-1} - \alpha'(\tilde{g}^t + \lambda'\theta^t). \tag{31}$$

Rearranging we get the recurrence:

$$\theta^{t+1} = (1 + \mu' - \alpha'\lambda')\theta^t - \mu'\theta^{t-1} - \alpha'\tilde{g}^t. \tag{32}$$

The two recurrent sequences θ^t can be made equal by equating the coefficients at θ^t, θ^{t-1} and $\tilde{g}^t$. We get three equations in three unknowns λ', μ', α':

$$1 + \mu' - \alpha'\lambda' = 1 + \mu - \alpha\lambda, \tag{33a}$$
$$\mu' = \mu(1 - \alpha\lambda), \tag{33b}$$
$$\alpha' = \alpha. \tag{33c}$$

We trivially find α' and μ', and solve for λ' from the first equation:

$$\lambda' = (\mu' - \mu + \alpha\lambda)/\alpha' = (\mu + \mu\alpha\lambda - \mu + \alpha\lambda)/\alpha = \mu\lambda + \lambda = (\mu + 1)\lambda. \tag{34}$$

We obtained that the two algorithms are equivalent up to changing the regularization strength only. If we used EWA form (with q and $1 - q$), the equivalence can be shown by the same method. □