ECE 046211 - Technion - Deep Learning

HW1 - Optimization and Automatic Differentiation

Question 1 - Convergence of Gradient Descent

Recall from the lecture notes:

- **Definition**: A function f is β-smooth if:

$$\forall w_1, w_2 \in \mathbb{R}^d : ||\nabla f(w_1) - \nabla f(w_2)|| \leq \beta ||w_1 - w_2||$$

- **Lemma**: If f is β-smooth then

$$f(w_1) - f(w_2) - \nabla f(w_2)^T (w_1 - w_2) \leq \frac{\beta}{2} ||w_1 - w_2||^2$$

Prove the lemma.

Hints:

- Represent f as an integral: $f(x) - f(y) = \int_0^1 \nabla f(y + t(x-y))^T (x-y) dt$
- Make use of Cauchy-Schwarz.

Solution 1 - Convergence of Gradient Descent

We represent $f(x) - f(y)$ as an integral, and apply Cauchy-Schwarz and then β-smoothness.

$$||f(x) - f(y) - \nabla f(y)^T (x-y)|$$

$$= |\int_0^1 \nabla f(y + t(x-y))^T (x-y) dt - \nabla f(y)^T (x-y)|$$

$$\leq \int_0^1 ||\nabla f(y + t(x-y)) - \nabla f(y)|| \cdot ||x-y|| dt$$

$$\leq \int_0^1 \beta t ||x-y||^2 dt$$

$$= \frac{\beta}{2} ||x-y||^2$$

Question 2 - Optimization and Gradient Descent

The function $f : \mathbb{R}^d \to \mathbb{R}$ is infinitely continuously differentiable, and satisfies $\min_{w\in\mathbb{R}^d} f(w) = f_* > -\infty$.

We wish to minimize this function using a version of Gradient Descent (GD) with step-size η, where in each iteration the gradients are multiplied by matrix A

$$(*)\ w(t+1) = w(t) - \eta A \nabla f\left(w(t)\right).$$

Matrix A is symmetric and strictly positive (positive definite with strictly positive eigenvalues), i.e., $\lambda_{min} \triangleq \lambda_{min}(A) > 0$, and denote $\lambda_{max} \triangleq \lambda_{max}(A)$.

1. In section only assume that $f(w) = \frac{1}{2} w^T H w$, where H is symmetric and strictly positive (positive definite with strictly positive eigenvalues). Find/choose A and η such that the algorithm $(*)$ converges in minimal number of steps. Why is that choice is infeasible when d is large? What is a common applicable approximation?
2. Prove that Gradient Flow (i.e., GD in the limit $\eta \to 0$):

   $$\dot{w}(t) = -A \nabla f\left(w(t)\right)$$

   converges to a critical point for all f and A that satisfy the conditions in the given question.
   - **Hint**: from the properties of eigenvalues it satifies that $\forall v \in \mathbb{R}^d : \lambda_{min}||v||^2 \leq v^T A v \leq \lambda_{max}||v||^2$.
3. Given that the function f is β-smooth, find a condition on the step-size η such that we get convergence to a critical point in algorithm $(*)$. Prove convergence under this condition.
   - **Hint**: for a β-smooth function, one can write:

   $$f\left(w(t+1)\right) - f\left(w(t)\right) \leq \left(w(t+1) - w(t)\right)^T \nabla f\left(w(t)\right) + \frac{\beta}{2}||w(t+1) - w(t)||^2$$

Solution 2 - Optimization and Gradient Descent

1. We will choose $\eta A = H^{-1}$ to get convergence in a single step:

   $$w(t+1) = w(t) - \eta A \nabla f\left(w(t)\right) = w(t) - \eta A H w(t) = w(t) - H^{-1} H w(t) = 0.$$

   For a large d, inverting a matrix is expensive, an a common approximation is using a diagonal matrix:

   $$(\eta A)_{ij} = \begin{cases} H_{ii}^{-1}, & \text{if } i = j \\ 0, & \text{if } i \neq j \end{cases}.$$

   There are also other approximations as seen in class.

2. It satisfies:

   $$\begin{aligned} f\left(\mathbf{w}(T)\right) - f\left(\mathbf{w}(0)\right) &= \int_0^T \frac{d}{dt} f\left(\mathbf{w}(t)\right) dt = \int_0^T \nabla f\left(\mathbf{w}(t)\right)^T \dot{\mathbf{w}}(t)\, dt \\ &= -\int_0^T \nabla f\left(\mathbf{w}(t)\right)^T \mathbf{A} \nabla f\left(\mathbf{w}(t)\right) dt \leq -\boldsymbol{\lambda}_{\min}(A) \int_0^T \left\|\nabla f\left(\mathbf{w}(t)\right)\right\|^2 dt \leq 0 \end{aligned}$$

   In the last transition:
   - We used that A is symmetrical and strictly positive, hence $\lambda_{min}(A) > 0$.
   - Equivalnce happens if and only if $\forall t \in [0, T] : \nabla f(w(t)) = 0$.

   I.e., unless $\nabla f(w(t)) \to 0$, we get $\lim_{T\to\infty} f(w(T)) = -\infty$. But this contradicts the condition $\min_{w\in\mathbb{R}^d} f(w) = f_* > -\infty$. Hence, $\nabla f(w(t)) \to 0$ and we get convergence to a critical point

of f.

3. Using the hint:

$$f\left(\mathbf{w}\left(t+1\right)\right)-f\left(\mathbf{w}\left(t\right)\right)\leq\left(\mathbf{w}\left(t+1\right)-\mathbf{w}\left(t\right)\right)^{T}\nabla f\left(\mathbf{w}\left(t\right)\right)+\frac{\beta}{2}\left\|\mathbf{w}\left(t+1\right)-\mathbf{w}\left(t\right)\right\|^{2}$$

$$=-\left(\eta\mathbf{A}\nabla f\left(\mathbf{w}\left(t\right)\right)\right)^{T}\nabla f\left(\mathbf{w}\left(t\right)\right)+\frac{\beta}{2}\left\|\eta\mathbf{A}\nabla f\left(\mathbf{w}\left(t\right)\right)\right\|^{2}$$

$$=-\eta\nabla f\left(\mathbf{w}\left(t\right)\right)^{T}\mathbf{A}\nabla f\left(\mathbf{w}\left(t\right)\right)+\frac{\beta\eta^{2}}{2}\nabla f\left(\mathbf{w}\left(t\right)\right)^{T}\mathbf{A}^{2}\nabla f\left(\mathbf{w}\left(t\right)\right)$$

$$\leq-\eta\lambda_{\min}\left\|\nabla f\left(\mathbf{w}\left(t\right)\right)\right\|^{2}+\frac{\beta\eta^{2}\lambda_{\max}^{2}}{2}\left\|\nabla f\left(\mathbf{w}\left(t\right)\right)\right\|^{2}$$

$$=-\eta\left(\lambda_{\min}-\frac{\beta\eta\lambda_{\max}^{2}}{2}\right)\left\|\nabla f\left(\mathbf{w}\left(t\right)\right)\right\|^{2}$$

Hence, if $\eta<\frac{2\lambda_{min}}{\beta\lambda_{max}^{2}}$, then $c\triangleq\eta\left(\lambda_{min}-\frac{\beta\eta\lambda_{max}^{2}}{2}\right)>0$ and it satifies:

$$f(w(t+1))-f(w(t))\leq-c||\nabla f(w(t))||^{2},$$

summing this telescoping series from 0 to $T-1$ we get:

$$f(w(T))\leq-c\sum_{t=0}^{T-1}||\nabla f(w(t))||^{2}+f(w(0)).$$

I.e., as before, unless $\nabla f(w(t))\to0$, we get $\lim_{T\to\infty}f(w(T))=-\infty$. But this contradicts the condition $\min_{w\in\mathbb{R}^{d}}f(w)=f_{*}>-\infty$. Hence, $\nabla f(w(t))\to0$ and we get convergence to a critical point of f as long as $\eta<\frac{2\lambda_{min}}{\beta\lambda_{max}^{2}}$.

Since you were asked to find *some* condition, the above solution is enough. However, there is a tighter solution:

$$f\left(\mathbf{w}\left(t+1\right)\right)-f\left(\mathbf{w}\left(t\right)\right)\leq\left(\mathbf{w}\left(t+1\right)-\mathbf{w}\left(t\right)\right)^{T}\nabla f\left(\mathbf{w}\left(t\right)\right)+\frac{\beta}{2}\left\|\mathbf{w}\left(t+1\right)-\mathbf{w}\left(t\right)\right\|^{2}$$

$$=-\left(\eta\mathbf{A}\nabla f\left(\mathbf{w}\left(t\right)\right)\right)^{T}\nabla f\left(\mathbf{w}\left(t\right)\right)+\frac{\beta}{2}\left\|\eta\mathbf{A}\nabla f\left(\mathbf{w}\left(t\right)\right)\right\|^{2}$$

$$=-\eta\nabla f\left(\mathbf{w}\left(t\right)\right)^{T}\mathbf{A}\nabla f\left(\mathbf{w}\left(t\right)\right)+\frac{\beta\eta^{2}}{2}\nabla f\left(\mathbf{w}\left(t\right)\right)^{T}\mathbf{A}^{2}\nabla f\left(\mathbf{w}\left(t\right)\right)$$

$$=-\nabla f\left(\mathbf{w}\left(t\right)\right)^{T}\left(\eta\mathbf{A}-\frac{\beta\eta^{2}}{2}\mathbf{A}^{2}\right)\nabla f\left(\mathbf{w}\left(t\right)\right)$$

$$\leq-\lambda_{\min}\left(\eta\mathbf{A}-\frac{\beta\eta^{2}}{2}\mathbf{A}^{2}\right)\left\|\nabla f\left(\mathbf{w}\left(t\right)\right)\right\|^{2}$$

To continue the proof as before, it is enough to demand $c\triangleq\lambda_{min}\left(\eta A-\frac{\beta\eta^{2}}{2}A^{2}\right)>0$, that is, $\eta A-\frac{\beta\eta^{2}}{2}A^{2}$ is a strictly positive matrix (all eigenvalues are strictly positive), which yields

$$\forall i:\eta\lambda_{i}-\frac{\beta\eta^{2}}{2}\lambda_{i}^{2}>0\Rightarrow\eta>\frac{2}{\beta\lambda_{max}}.$$

$$\forall i : \eta\lambda_i - \frac{\beta\eta^2}{2}\lambda_i^2 > 0 \Rightarrow \eta > \frac{2}{\beta\lambda_{max}}.$$

Question 3 - Efficient Differentiation

We wish to optimize a loss function $\mathcal{L}(\mathbf{w})$ for $\mathbf{w} \in \mathbb{R}^d$ using Gradient Descent (GD) with some step size schedule η_t

$$(1)\ \ \forall t = 1, 2, \ldots : \mathbf{w}(t) = \mathbf{w}(t-1) - \eta_t \nabla \mathcal{L}(\mathbf{w}(t-1)) \tag{1}$$

initialized from some $\mathbf{w}(0)$. We would like to learn the best step size schedule using GD. **Hint**: throughout this question, you should use the *chain rule*.

1. Suppose we can consider each η_t as a separate parameter for each t. We initialize this parameter with η_0 and update η_{t-1} with a GD step on $\mathcal{L}(\mathbf{w}(t-1))$

$$(2)\ \ \eta_t = \eta_{t-1} - \alpha_t \frac{\partial \mathcal{L}(\mathbf{w}(t-1))}{\partial \eta_{t-1}} \tag{2}$$

for every step of eq. (1), where α_t is the another step size. Calculate $\partial \mathcal{L}(\mathbf{w}(t-1)) / \partial \eta_{t-1}$ as a function of the loss gradients $\nabla \mathcal{L}(\mathbf{w}(t-1))$ and $\nabla \mathcal{L}(\mathbf{w}(t-2))$. 2. Now suppose we want to similarly update α_{t-1} using GD step on $\mathcal{L}(\mathbf{w}(t-1))$ every step of eq. (2) with update step κ_t

$$\alpha_t = \alpha_{t-1} - \kappa_t \frac{\partial \mathcal{L}(\mathbf{w}(t-1))}{\partial \alpha_{t-1}}. \tag{3}$$

Calculate $\partial \mathcal{L}(\mathbf{w}(t-1)) / \partial \alpha_{t-1}$ as a function of $\{\nabla \mathcal{L}(\mathbf{w}(t-k))\}_{k=1}^{3}$. 3. Now we wish to update (η_{t-1}, η_{t-2}) by doing a GD step on $\mathcal{L}(\mathbf{w}(t-1))$

$$(3)\ \ (\eta_{t+1}, \eta_t) = (\eta_{t-1}, \eta_{t-2}) - \alpha_t \left(\frac{\partial \mathcal{L}(\mathbf{w}(t-1))}{\partial \eta_{t-1}}, \frac{\partial \mathcal{L}(\mathbf{w}(t-1))}{\partial \eta_{t-2}} \right) \tag{4}$$

every two steps of eq. (1). Calculate the derivative $\frac{\partial \mathcal{L}(\mathbf{w}(t-1))}{\partial \eta_{t-2}}$ as a function of η_{t-1}, $\{\nabla \mathcal{L}(\mathbf{w}(t-k))\}_{k=1}^{3}$, and $\nabla^2 \mathcal{L}(\mathbf{w}(t-2))$. 4. Now we wish again to update $(\eta_t, \eta_{t+1}, \ldots, \eta_{t+T})$ by doing a GD step on $\mathcal{L}(\mathbf{w}(t-1))$ every T steps of eq. (1)

$$(4)\ \ (\eta_{t+T}, \ldots, \eta_t) = (\eta_{t-1}, \ldots, \eta_{t-1-T}) - \alpha_t \left(\frac{\partial \mathcal{L}(\mathbf{w}(t-1))}{\partial \eta_{t-1}}, \ldots, \frac{\partial \mathcal{L}(\mathbf{w}(t-1))}{\partial \eta_{t-1-T}} \right) \tag{5}$$

Calculate the derivative $\frac{\partial \mathcal{L}(\mathbf{w}(t-1))}{\partial \eta_{t-\tau}}$ as a function of $\left\{\eta_{t-k}, \nabla^2 \mathcal{L}(\mathbf{w}(t-k-1))\right\}_{k=1}^{\tau-1}$, $\nabla \mathcal{L}(\mathbf{w}(t-1))$ and $\nabla \mathcal{L}(\mathbf{w}(t-\tau-1))$. 5. Compare this approach (eq. (4) with $T > 1$) to the first one (eq. (2)). Name one advantage for each approach. Hints: Think of computional complexity, ease of optimization, suitability of the objective.

Solution 3 - Efficient Differentiation

1. Using the chain rule, we have

$$\begin{aligned}\frac{\partial \mathcal{L}\left(\mathbf{w}\left(t-1\right)\right)}{\partial \eta_{t-1}} &= \frac{\partial \mathcal{L}\left(\mathbf{w}\left(t-1\right)\right)}{\partial \mathbf{w}\left(t-1\right)}\frac{\partial \mathbf{w}\left(t-1\right)}{\partial \eta_{t-1}} \\ &= -\nabla \mathcal{L}\left(\mathbf{w}\left(t-1\right)\right)\cdot\frac{\partial}{\partial \eta_{t-1}}\left(\mathbf{w}\left(t-2\right)-\eta_{t-1}\nabla \mathcal{L}\left(\mathbf{w}\left(t-2\right)\right)\right) \\ &= -\nabla \mathcal{L}\left(\mathbf{w}\left(t-1\right)\right)\cdot\nabla \mathcal{L}\left(\mathbf{w}\left(t-2\right)\right)\end{aligned}$$

2. Using the chain rule again,

$$\begin{aligned}\frac{\partial \mathcal{L}\left(\mathbf{w}\left(t-1\right)\right)}{\partial \alpha_{t-1}} &= \frac{\partial \mathcal{L}\left(\mathbf{w}\left(t-1\right)\right)}{\partial \eta_{t-1}}\frac{\partial \eta_{t-1}}{\partial \alpha_{t-1}} \\ &= -\nabla \mathcal{L}\left(\mathbf{w}\left(t-1\right)\right)\cdot\nabla \mathcal{L}\left(\mathbf{w}\left(t-2\right)\right)\frac{\partial}{\partial \alpha_{t-1}}\left(\eta_{t-2}-\alpha_{t-1}\frac{\partial \mathcal{L}\left(\mathbf{w}\left(t-2\right)\right)}{\partial \eta_{t-2}}\right) \\ &= -\nabla \mathcal{L}\left(\mathbf{w}\left(t-1\right)\right)\cdot\nabla \mathcal{L}\left(\mathbf{w}\left(t-2\right)\right)\frac{\partial \mathcal{L}\left(\mathbf{w}\left(t-2\right)\right)}{\partial \eta_{t-2}} \\ &= \nabla \mathcal{L}\left(\mathbf{w}\left(t-1\right)\right)\cdot\nabla \mathcal{L}\left(\mathbf{w}\left(t-2\right)\right)\nabla \mathcal{L}\left(\mathbf{w}\left(t-2\right)\right)\cdot\nabla \mathcal{L}\left(\mathbf{w}\left(t-3\right)\right)\ .\end{aligned}$$

3. In this case, we have

$$\begin{aligned}\frac{\partial \mathcal{L}\left(\mathbf{w}\left(t-1\right)\right)}{\partial \eta_{t-2}} &= \frac{\partial \mathcal{L}\left(\mathbf{w}\left(t-1\right)\right)}{\partial \mathbf{w}\left(t-1\right)}\frac{\partial \mathbf{w}\left(t-1\right)}{\partial \mathbf{w}\left(t-2\right)}\frac{\partial \mathbf{w}\left(t-2\right)}{\partial \eta_{t-2}} \\ &= \nabla \mathcal{L}\left(\mathbf{w}\left(t-1\right)\right)\frac{\partial\left(\mathbf{w}\left(t-2\right)-\eta_{t-1}\nabla \mathcal{L}\left(\mathbf{w}\left(t-2\right)\right)\right)}{\partial \mathbf{w}\left(t-2\right)}\frac{\partial\left(\mathbf{w}\left(t-3\right)-\eta_{t-2}\nabla \mathcal{L}\left(\mathbf{w}\left(\right.\right.\right.\right.}{\partial \eta_{t-2}} \\ &= -\nabla \mathcal{L}\left(\mathbf{w}\left(t-1\right)\right)\left(\mathbf{I}-\eta_{t-1}\nabla^2 \mathcal{L}\left(\mathbf{w}\left(t-2\right)\right)\right)\nabla \mathcal{L}\left(\mathbf{w}\left(t-3\right)\right)\end{aligned}$$

4. In this case, we have

$$\begin{aligned}\frac{\partial \mathcal{L}\left(\mathbf{w}\left(t-1\right)\right)}{\partial \eta_{t-\tau}} &= \frac{\partial \mathcal{L}\left(\mathbf{w}\left(t-1\right)\right)}{\partial \mathbf{w}\left(t-1\right)}\frac{\partial \mathbf{w}\left(t-1\right)}{\partial \mathbf{w}\left(t-2\right)}\frac{\partial \mathbf{w}\left(t-2\right)}{\partial \mathbf{w}\left(t-3\right)}\cdots\frac{\partial \mathbf{w}\left(t-\tau\right)}{\partial \eta_{t-\tau}} \\ &= -\nabla \mathcal{L}\left(\mathbf{w}\left(t-1\right)\right)\cdot\left[\prod_{t'=1}^{\tau-1}\left(\mathbf{I}-\eta_{t-t'}\nabla^2 \mathcal{L}\left(\mathbf{w}\left(t-t'-1\right)\right)\right)\right]\nabla \mathcal{L}\left(\mathbf{w}\left(t-\tau-1\right)\right)\end{aligned}$$

5. Advantages of the first approach (one advantage is enough for getting a full score):
   - It updates the step size more frequently, which is good since delayed updates can harm performance, as we learned in class.
   - It is cheaper, since it does not include the Hessians. As we learned in class, while calculating Hessians is extremely expensive, Hessian-vectors products can be calculated in practice, but are still more expensive then regular gradients.
   - The second approach requires Backpropagation through a long optimization process. This is expensive (since we have to store in memory all the optimization process), and might also lead to instabilities due to vanishing or exploding gradients.

Advantages of the second approach: We update the step size according the loss at a later time which is closer to what we want to minimize (the loss at the end of training), while the first approach is greedy and aims to minimize the loss after a single step.

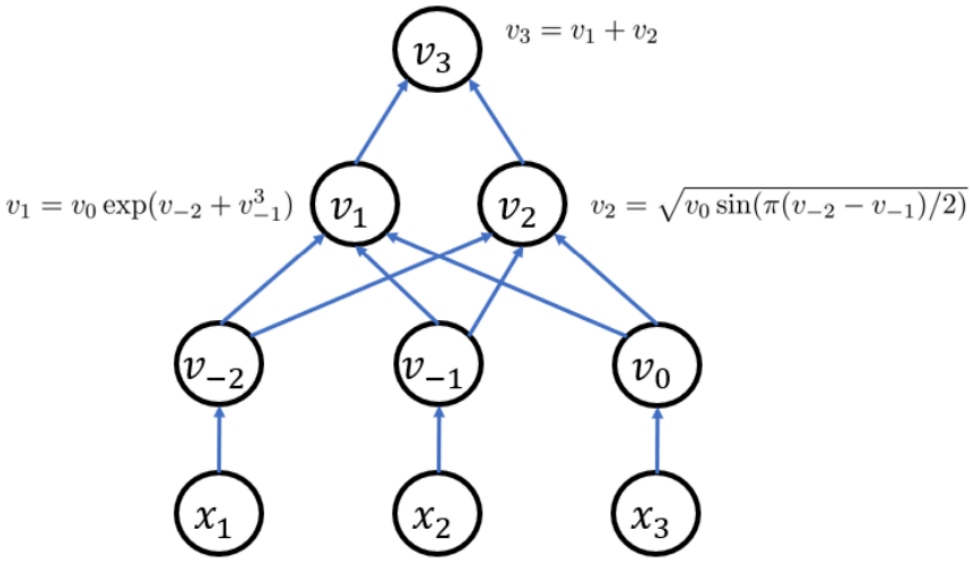
Question 4 - Automatic Differentiation

Consider the following function:

$$y = \exp(x_1 + x_2^3)x_3 + \sqrt{x_3 \sin\left(\frac{\pi}{2}(x_1 - x_2)\right)}$$

1. Write this function as a computational graph with *at least* 2 internal variables (you can draw the graph by hand and attach the drawing as an image file).
2. Use **forward mode autodiff** to calculate $\frac{\partial y}{\partial x_1}$ at $(x_1, x_2, x_3) = (2, 1, 1)$.
3. Use **backward mode autodiff** to calculate $\frac{\partial y}{\partial x_2}$ at $(x_1, x_2, x_3) = (2, 1, 1)$.
4. Use **numerical differentiation** to calculate $\frac{\partial y}{\partial x_3}$ at $(x_1, x_2, x_3) = (1, 1, 1)$. Which method for differentiation will you use? What will be the step size (assume the numerical precision $\epsilon = 0.0001$)?
5. Describe the advantages and disadvantages for each method (forward, backward and numerical) for a general function.

Solution 4 - Automatic Differentiation



1.
2. First, we calcualte the forward propagation:

$$v_1 = \exp(x_1 + x_2^3)x_3 = e^3, v_2 = \sqrt{x_3 \sin(\pi(x_1 - x_2)/2)} = 1$$

$$\rightarrow v_3 = v_1 + v_2 = e^3 + 1$$

Then,

the forward propagation of the derivatives:

$$\dot{v}_1 \triangleq \frac{\partial v_1}{\partial x_1} = \exp(x_1 + x_2^3)x_3 = e^3$$

$$\dot{v}_2 \triangleq \frac{\partial v_2}{\partial x_1} = \sqrt{(x_3 \sin(\pi(x_1 - x_2)/2)} = \frac{\pi \cos(\pi(2-1)/2)}{\sqrt{16 \sin(\pi(2-1)/2)}} = 0$$

$$\rightarrow \dot{v}_3 \triangleq \frac{\partial v_3}{\partial v_{-2}} = \frac{\partial v_3}{\partial x_1} = \dot{v}_1 + \dot{v}_2 = e^3$$

3. Since this is the same point (x_1, x_2, x_3) as the in the previous section, we use the same forward propagation. Backpropagation of the derivaties:

$$\bar{v}_3 \triangleq \frac{\partial v_3}{\partial v_3} = 1, \bar{v}_2 \triangleq \frac{\partial v_3}{\partial v_2} = 1, \bar{v}_1 \triangleq \frac{\partial v_3}{\partial v_1} = 1$$

$$\bar{v}_{-1} \triangleq \frac{\partial v_3}{\partial v_0} = \frac{\partial v_3}{\partial x_2} = \frac{\partial v_3}{\partial v_1}\frac{\partial v_1}{\partial x_2} + \frac{\partial v_3}{\partial v_2}\frac{\partial v_2}{\partial x_2}$$

$$= 1 \cdot 3x_2^2 \exp(x_1 + x_2^3)x_3 + 1 \cdot \frac{-\pi \cos(\pi(2-1)/2)}{\sqrt{16 \sin(\pi(2-1)/2)}} = 3e^3$$

4. We will use the *central difference* as it achieves lower error and is simpler to implement. As recommended in the lecture, we will use the following step size:

$$h = (1 + \max_i |x_i|)\sqrt{\epsilon},$$

to get:

$$\frac{(\exp(2)(1+h) - \exp(2)(1-h))}{2h} = \exp(2)$$

5. In **numerical** and **forward mode** the main problem is that the derivative calculation is performed as the number of input variables (i.e., the number of differentiable variables) - a number which is usually very large (e.g., the weights of a deep neural network). This is not a problem in **backward mode** where this calculation is performed once for all input variables. Moreover, in **numerical** there is also the problem of numerical noise; however, it is very simple to implement. **Forward mode** has an efficiency advantage when the computational graph has a large number of outputs, where the other methods usually need to do the entire computation from scratch for each output. The main disadvantage of **Backward mode** is the need to save the values calculated during the forward propagation in memory.

Question 5 - Automatic Differentiation 2

Write down the chain rule in the dual numbers representation for the following:

$$f(g(h(x+\epsilon x')))$$

What is $\frac{df(x)}{dx}$?

Solution 5 - Automatic Differentiation 2

Solution:

$$\begin{aligned} f(g(h(x+\epsilon x'))) &= f(g(h(x)+\epsilon h'(x)x')) \\ &= f(g(h(x))+\epsilon g'(h(x))h'(x)x') \\ &= f(g(h(x)))+\epsilon f'(g(h(x)))g'(h(x))h'(x)x' \end{aligned}$$

So:

$$\begin{aligned} \left.\frac{df(x)}{dx}\right|_x &= \text{coefficient-of-epsilon}(\text{dual-version}(f)(x+1\cdot\epsilon)) \\ &= f'(g(h(x)))g'(h(x))h'(x) \end{aligned}$$

Credits

- Icons made by Becris from www.flaticon.com
- Icons from Icons8.com - https://icons8.com
- Datasets from Kaggle - https://www.kaggle.com/