Assignment 03

Md Tasin Siddiqi, Hussein Albared

19th November 2021

1 Theoretical Exercises

a)

1. We have to analyse to function.

$$f(x) = 100\left(x_2 - x_1^2\right)^2 + (1 - x_1)^2$$

The first step is to compute the partial derivatives:

$$\frac{\partial f(x)}{\partial x_1} = \frac{\partial}{\partial x_1}100\left(x_2 - x_1^2\right)^2 + \frac{\partial}{\partial x_1}(1 - x_1)^2 = -400x_1\left(x_2 - x_1^2\right) + \frac{\partial}{\partial x_1}(1 - x_1)^2$$

$$= -400x_1\left(x_2 - x_1^2\right) - 2(1 - x_1) = 400\left(x_1^3 - x_1x_2\right) + 2x_1 - 2$$

$$\frac{\partial f(x)}{\partial x_2} = \frac{\partial}{\partial x_2}100\left(x_2 - x_1^2\right)^2 + \frac{\partial}{\partial x_2}(1 - x_1)^2 = 200\left(x_2 - x_1^2\right) + \frac{\partial}{\partial x_2}(1 - x_1)^2$$

$$= 200\left(x_2 - x_1^2\right) + 0 = 200\left(x_2 - x_1^2\right)$$

If we know the partial derivatives we can define the functions Jacobin matrix:

$$\nabla f(x) = \begin{pmatrix} \frac{\partial f(x)}{\partial x_1} \\ \frac{\partial f(x)}{\partial x_2} \end{pmatrix} = \begin{pmatrix} 400\left(x_1^3 - x_1x_2\right) + 2x_1 - 2 \\ 200\left(x_2 - x_1^2\right) \end{pmatrix}$$

2. The second step is to compute the second order partial derivatives:

$$\frac{\partial^2 f}{\partial x_1 \partial x_1} = \frac{\partial f}{\partial x_1}400\left(x_1^3 - x_1x_2\right) + 2x_1 - 22 = -400\left(x_2 - 3x_1^2\right) + 2$$

$$\frac{\partial^2 f}{\partial x_1 \partial x_2} = \frac{\partial f}{\partial x_2}400\left(x_1^3 - x_1x_2\right) + 2x_1 - 2 = -400x_1$$

$$\frac{\partial^2 f}{\partial x_2 \partial x_2} = \frac{\partial f}{\partial x_2}200\left(x_2 - x_1^2\right) = 200$$

With these we can define the function's Hessian matrix:

$$\nabla^2 f(x) = \begin{pmatrix} \frac{\partial^2 f}{\partial x_1 \partial x_1} & \frac{\partial^2 f}{\partial x_1 \partial x_2} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2 \partial x_2} \end{pmatrix} = \begin{pmatrix} -400\left(x_2 - 3x_1^2\right) + 2 & -400x_1 \\ -400x_1 & 200 \end{pmatrix}$$

The Rosenbrock function's Jacobian matrix at $x^* = (1,1)^T$ is

$$\nabla f\left(x^*\right) = \begin{pmatrix} 400\left(1^3 - 1 \cdot 1\right) + 2 \cdot 1 - 2 \\ 200\left(1 - 1^2\right) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

and its Hessian matrix at the same point is

$$\nabla^2 f\left(x^*\right) = \begin{pmatrix} -400\left(1 - 3 \cdot 1^2\right) + 2 & -400 \cdot 1 \\ -400 \cdot 1 & 200 \end{pmatrix} = \begin{pmatrix} 802 & -400 \\ -400 & 200 \end{pmatrix}$$

The Jacobian $\nabla f\left(x^*\right) = 0$ implies that x^* is indeed a local extremum but to verify that it is a minimum we have to check if the Hessian is positive definite.

Note that the Hessian matrix is always symmetrical (Schwarz's theorem) and that a symmetric matrix is positive definite iff. all its principal minors are positive.

$$\nabla^2 f\left(x^*\right) = \begin{pmatrix} 802 & -400 \\ -400 & 200 \end{pmatrix}$$

$M_1 = 802 > 0$ $M_2 = det\nabla^2 f\left(x^*\right) = (802 * 200) - (-400)^2 = 400 > 0$ The point x^* is a minimum of the Rosenbrock function.

3. To convince ourselves that $x^* = (1,1)^T$ is the only minimum, note that the partial derivatives have to following roots:

$$\frac{\partial f}{\partial x_1} : x_2 = \frac{200x_1^3 - x_1 + 1}{200x_1}$$

$$\frac{\partial f}{\partial x_2} : x_2 = x_1^2$$

Therefore any extremum $x = (x_1, x_2)$ has to satisfy

$$x_1^2 = \frac{200x_1^3 - x_1 + 1}{200x_1} \Leftrightarrow x_1 = 1$$

The first steps are identical: Derive the Jacobian matrix and the Hessian Matrix

$$\nabla g(x) = \begin{pmatrix} 2x_1 + 8 \\ -4x_2 + 12 \end{pmatrix}$$

We now have to find the root of $\nabla g(x)$: $\frac{\partial g}{\partial x_1} = 2x_1 + 8 = 0 \Leftrightarrow x_1 = -4$

$\nabla^2 g(x) = \begin{pmatrix} 2 & 0 \\ 0 & -4 \end{pmatrix}$ $\frac{\partial g}{\partial x_2} = -4x_2 + 12 = 0 \Leftrightarrow x_2 = 3$

The only extremum of g is $x^* = (-4, 3)^T$. Note that a symmetric matrix is indefinite iff. its eigenvalues have a different sign. Therefore x^* is a saddle point of g.

b) (a) The area of the error surface is called a flat area or plateau. Changes of the parameter do not result in a significant change of the error value. Conversely, it means that the gradient in these areas has a small magnitude. This problem is often encountered in very deep networks.

(b) Choose a Hessian with mostly negative eigenvalues λ_i and a large magnitude $|\lambda_i| >> 1$. Negative Eigen values correspond to directions in which the function is going down. The bigger the magnitude of the corresponding Eigen value , the steeper the second-order descent direction will be.

c) Given is a two layer neural network with "sigmoid" and "softmax" as activation functions.

Our task is to Compute the forward pass for inputs $x = (0.1, 0.4)$ and the labels $y = (0.1, 0.9)$
Compute the gradient and update the weights based on the error.
Recompute the forward pass with the updated weights.

From the definition of the sigmoid and softmax function

$$\sigma(t) = \frac{1}{1 + e^{-t}} and softmax(x)_i = \frac{e^{x_i}}{\sum_{k=1}^{d} e^{x_k}}$$

The hidden layer returns
$h_1 = \sigma(w_1 x_1 + w_2 x_2 + b_1) = \sigma(0.39) = 0.5963$
$h_2 = \sigma(w_3 x_1 + w_4 x_2 + b_1) = \sigma(0.44) = 0.6083$

Using $h_1 = 0.5963$ and $h_2 = 0.6083$, the output values are
$o_1 = w_5 h_1 + w_6 h_2 + b_2 = 1.1426$
$o_2 = w_7 h_1 + w_8 h_2 + b_2 = 1.2631$

Applying the softmax function results in the final output

$\text{net}\ _1 = \frac{e^{o_1}}{e^{o_1}+e^{o_2}} = 0.4699$

$\text{net}\ _2 = \frac{e^{o_2}}{e^{o_1}+e^{o_2}} = 0.5301$

Using $h_1 = 0.5963$ and $h_2 = 0.6083$, the output values are
$o_1 = w_5 h_1 + w_6 h_2 + b_2 = 1.1426$
$o_2 = w_7 h_1 + w_8 h_2 + b_2 = 1.2631$

Applying the softmax function results in the final output

$\text{net}_1 = \frac{e^{o_1}}{e^{o_1}+e^{o_2}} = 0.4699$
$\text{net}_2 = \frac{e^{o_2}}{e^{o_1}+e^{o_2}} = 0.5301$

We can now calculate the error using the squared error function:

$\text{E}_{total} = \frac{1}{2} \sum_{i=1}^{2} (y_i - net_i)^2$

$\frac{1}{2}\left((0.1 - 0.4699)^2 + (0.9 - 0.5301)^2\right) = 0.1368$

We need the quotient rule $\left(\frac{f}{g}\right) = \frac{f'g - fg'}{g^2}$ to compute the derivative of the "softmax" function:

1. Case $i = j$

$$\frac{\partial}{\partial x_j} softmax(x)_i = \frac{\partial}{\partial x_j} \frac{e^{x_i}}{\sum_{k=1}^{d} e^{x_k}} = \frac{e^{x_i} \sum_{k=1}^{d} e^{x_k} - e^{x_j} e^{x_i}}{\left(\sum_{j=k}^{d} e^{x_k}\right)^2} = \frac{e^{x_i}}{\sum_{k=1}^{d} e^{x_k}} \frac{\sum_{k=1}^{d} e^{x_k} - e^{x_j}}{\sum_{k=1}^{d} e^{x_k}}$$

$=\text{softmax(x)}_i \left(1 - softmax(x)_i\right)$

We need the quotient rule $\left(\frac{f}{g}\right)' = \frac{f'g - fg'}{g^2}$ to compute the derivative of the "softmax" function:

2. Case $i \neq j$

$$\frac{\partial}{\partial x_j} softmax(x)_i = \frac{e^{x_i}}{\sum_{k=1}^{d} e^{x_k}} = \frac{0 - e^{x_i} e^{x_j}}{\left(\sum_{k=1}^{d} e^{x_k}\right)^2} = -\frac{e^{x_i}}{\sum_{k=1}^{d} e^{x_k}} \frac{e^{x_j}}{\sum_{k=1}^{d} e^{x_k}}$$

=-softmax(x)$_i softmax(x)_j$

We have to consider both output o_1 and o_2 for all weight when computing the derivative. We use $\sigma(t)' = \sigma(t)(1 - \sigma(t))$ without proof.

Consider the weight w_5 in the output layer:

$$\frac{\partial E_{total}}{\partial w_5} = \frac{\partial E_{total}}{\partial net_1} \frac{\partial net_1}{\partial o_1} \frac{\partial o_1}{\partial w_5} + \frac{\partial E_{total}}{\partial net_2} \frac{\partial net_2}{\partial o_1} \frac{\partial o_1}{\partial w_5}$$

1. $\frac{\partial E_{total}}{\partial net_1} = \frac{\partial}{\partial net_1} \frac{1}{2} (y_1 - net_1)^2 + \frac{1}{2} (y_2 - net_2)^2 = -(y_1 - \text{ net }_1)$
2. $\frac{\partial E_{total}}{\partial net_2} = \frac{\partial}{\partial net_2} \frac{1}{2} (y_1 - net_1)^2 + \frac{1}{2} (y_2 - net_2)^2 = -(y_2 - \text{ net }_2)$
3. We know from the previous slide: $\frac{\partial net_1}{\partial o_1} = \text{net }_1 (1 - \text{ net }_1)$.
4. We know from the previous slide: $\frac{\partial net_2}{\partial o_1} = - \text{ net }_1 \text{ net }_2$.
5. $\frac{\partial o_1}{\partial w_5} = \frac{\partial}{\partial w_5} w_5 h_1 + w_6 h_2 + b = h_1$

Consider the weight w_5 in the output layer:

$$\frac{\partial E_{total}}{\partial w_5} = \frac{\partial E_{total}}{\partial net_1} \frac{\partial net_1}{\partial o_1} \frac{\partial o_1}{\partial w_5} + \frac{\partial E_{total}}{\partial net_2} \frac{\partial net_2}{\partial o_1} \frac{\partial o_1}{\partial w_5}$$

=-$(y_1 - net_1)(net_1(1 - net_1)) h_1 + (y_2 - net_2)(net_2 net_1) h_1$

The main differences between the partial derivatives are the value of $\frac{\partial o_j}{\partial w_i}$ and the sign of $\frac{\partial net_k}{\partial o_j}$:

$\frac{\partial E_{total}}{\partial w_6} = -(y_1 - net_1)(net_1(1 - net_1)) h_2 + (y_2 - net_2)(net_2 net_1) h_2$

$\frac{\partial E_{total}}{\partial w_7} = (y_1 - \text{ net }_1)(\text{ net }_1 \text{ net }_2) h_1 - (y_2 - \text{ net }_2)(\text{ net }_2 (1 - \text{ net }_2)) h_1$

$\frac{\partial E_{total}}{\partial w_8} = (y_1 - net_1)(net_1 net_2) h_2 - (y_2 - net_2)(net_2(1 - net_2)) h_2$

Consider the following derivative and apply the product rule $(f \cdot g)' = f'g + fg'$:

$$\frac{\partial}{\partial b} \frac{e^{\alpha+b}}{e^{\alpha+b} + e^{\beta+b}} = \frac{\partial}{\partial b} e^{\alpha+b} \left(e^{\alpha+b} + e^{\beta+b}\right)^{-1}$$

$$= e^{\alpha+b} \left(e^{\alpha+b} + e^{\beta+b}\right)^{-1} - e^{\alpha+b} \left(e^{\alpha+b} + e^{\beta+b}\right)^{-2} \left(e^{\alpha+b} + e^{\beta+b}\right)$$

$$=0$$

The partial derivative $\frac{\partial E_{total}}{\partial b_2}$ has to be zero, because all chain rule expansions contain a derivative of the same form.

Consider the weight w_1 in the hidden layer:

$$\frac{\partial E_{total}}{\partial w_1} = \left(\frac{\partial E_{total}}{\partial net_1}\frac{\partial net_1}{\partial o_1} + \frac{\partial E_{total}}{\partial net_2}\frac{\partial net_2}{\partial o_1}\right)\frac{\partial o_1}{\partial w_1} + \left(\frac{\partial E_{total}}{\partial net_1}\frac{\partial net_1}{\partial o_2} + \frac{\partial E_{total}}{\partial net_2}\frac{\partial net_2}{\partial o_2}\right)\frac{\partial o_2}{\partial w_1}$$

$$=(-(y_1 - net_1)(net_1(1 - net_1)) + (y_1 - net_2)(net_2 net_1))\, w_5 (\sigma(h_1)(1 - \sigma(h_1)))\, x_1$$
$$+((y_2 - net_1)(net_1 net_2) - (y_2 - net_2)(net_2(1 - net_2)))\, w_7 (\sigma(h_1)(1 - \sigma(h_1)))\, x_1$$

Consider the weight w_2 in the hidden layer. Only the derivative $\frac{\partial o_1}{\partial w_2}$

$$\frac{\partial E_{total}}{\partial w_2} = \left(\frac{\partial E_{total}}{\partial net_1}\frac{\partial net_1}{\partial o_1} + \frac{\partial E_{total}}{\partial net_2}\frac{\partial net_2}{\partial o_1}\right)\frac{\partial o_1}{\partial w_2} + \left(\frac{\partial E_{total}}{\partial net_1}\frac{\partial net_1}{\partial o_2} + \frac{\partial E_{total}}{\partial net_2}\frac{\partial net_2}{\partial o_2}\right)\frac{\partial o_2}{\partial w_2}$$

$$=(-(y_1 - net_1)(net_1(1 - net_1)) + (y_1 - net_2)(net_2 net_1))\, w_5 (\sigma(h_1)(1 - \sigma(h_1)))\, x_2$$
$$+((y_2 - net_1)(net_1 net_2) - (y_2 - net_2)(net_2(1 - net_2)))\, w_7 (\sigma(h_1)(1 - \sigma(h_1)))\, x_2$$

Consider the weight w_3 in the hidden layer. We have to consider the path trough h_2 :

$$\frac{\partial E_{total}}{\partial w_3} = \left(\frac{\partial E_{total}}{\partial net_1}\frac{\partial net_1}{\partial o_1} + \frac{\partial E_{total}}{\partial net_2}\frac{\partial net_2}{\partial o_1}\right)\frac{\partial o_1}{\partial w_1} + \left(\frac{\partial E_{total}}{\partial net_1}\frac{\partial net_1}{\partial o_2} + \frac{\partial E_{total}}{\partial net_2}\frac{\partial net_2}{\partial o_2}\right)\frac{\partial o_2}{\partial w_1}$$

$$=(-(y_1 - net_1)(net_1(1 - net_1)) + (y_1 - net_2)(net_2 net_1))\, w_6 (\sigma(h_2)(1 - \sigma(h_2)))\, x_1$$
$$+((y_2 - net_1)(net_1 net_2) - (y_2 - net_2)(net_2(1 - net_2)))\, w_8 (\sigma(h_2)(1 - \sigma(h_2)))\, x_1$$

Consider the weight w_4 in the hidden layer. We have to consider the path trough h_2 :

$$\frac{\partial E_{total}}{\partial w_4} = \left(\frac{\partial E_{total}}{\partial net_1}\frac{\partial net_1}{\partial o_1} + \frac{\partial E_{total}}{\partial net_2}\frac{\partial net_2}{\partial o_1}\right)\frac{\partial o_1}{\partial w_2} + \left(\frac{\partial E_{total}}{\partial net_1}\frac{\partial net_1}{\partial o_2} + \frac{\partial E_{total}}{\partial net_2}\frac{\partial net_2}{\partial o_2}\right)\frac{\partial o_2}{\partial w_2}$$

$$=(-(y_1-net_1)(net_1(1-net_1))+(y_1-net_2)(net_2 net_1))w_6(\sigma(h_2)(1-\sigma(h_2)))x_2$$
$$+((y_2-net_1)(net_1 net_2)-(y_2-net_2)(net_2(1-net_2)))w_7(\sigma(h_2)(1-\sigma(h_2)))x_2$$

$$\frac{\partial E_{total}}{\partial b_1}=\sum_{i=1}^{2}\frac{\partial E_{total}}{\partial net_i}\sum_{j=1}^{2}\frac{\partial net_i}{\partial o_j}\sum_{k=1}^{2}\frac{\partial o_j}{\partial \sigma(h_k)}\frac{\partial \sigma(h_k)}{\partial b_1}$$

$\frac{\partial E_{total}}{\partial net_i}=-(y_i-\text{ net }_i)$

$\frac{\partial net_i}{\partial o_j}=net_i(1-net_i)$ if $i=j$ and $\frac{\partial net_i}{\partial o_j}=-\text{ net }_i\text{ net }_j$ otherwise.

$\frac{\partial o_j}{\partial \sigma(h_k)}=w_{2+2i+j}$

$\frac{\partial \sigma(h_k)}{\partial b_1}=\sigma(h_k)(1-\sigma(h_k))$

Apply the update rule $w_i \leftarrow w_i-\eta\frac{\partial E_{total}}{\partial w_i}$ with $\eta=0.5$:

$w_1 \leftarrow w_1-\eta\frac{\partial E_{total}}{\partial w_1}=0.1+\eta(0.00044)=0.1002$

$w_2 \leftarrow w_2-\eta\frac{\partial E_{total}}{\partial w_2}=0.2+\eta(0.00177)=0.2009$

$w_3 \leftarrow w_3-\eta\frac{\partial E_{total}}{\partial w_3}=0.2+\eta(0.00044)=0.2002$

$w_4 \leftarrow w_4-\eta\frac{\partial E_{total}}{\partial w_4}=0.3+\eta(0.00177)=0.3009$

$w_5 \leftarrow w_5-\eta\frac{\partial E_{total}}{\partial w_5}=0.4-\eta(0.10989)=0.3451$

$w_6 \leftarrow w_6-\eta\frac{\partial E_{total}}{\partial w_6}=0.5-\eta(0.11210)=0.4440$

$w_7 \leftarrow w_7-\eta\frac{\partial E_{total}}{\partial w_7}=0.5+\eta(0.10989)=0.5549$

$w_8 \leftarrow w_8-\eta\frac{\partial E_{total}}{\partial w_8}=0.6+\eta(0.11210)=0.6560$

$b_1 \leftarrow b_1-\eta\frac{\partial E_{total}}{\partial b_1}=0.3+\eta(0.00883)=0.3044$

$b_2 \leftarrow b_2-\eta\frac{\partial E_{total}}{\partial b_2}=0.6+\eta(0.00000)=0.6$

The error decreases to $E_{total}=0.11335$ after updating the weights.