Exercise 3

Malik Mardan

1. a)

A stochastic gradient descent is a greedy optimization algorithm used to minimize a loss function by iteratively updating weights in the opposite direction of the gradient of the loss. Formula: $w := w - \eta \frac{\partial L}{\partial w}$.
Instead of using the entire dataset we use randomly chosen samples.

b).

Full-batch GD updates weights after computing the gradient over the entire dataset, while mini-batch GD updates after smaller subsets, making it faster and more efficient.

2.

Forward:

$s_1 = \omega_1 x \quad , s_2 = \omega_2 x$

$z_1 = \max(0, s_1) \quad , z_2 = \max(0, s_2)$

$a_1 = z_1 \cdot u_{1,1} + z_2 \cdot u_{2,1} \quad , a_2 = z_1 \cdot u_{1,2} + z_2 \cdot u_{2,2}$

$f_j = \dfrac{e^{a_j}}{e^{a_1} + e^{a_2}}$ for $j = 1, 2$

$L = \frac{1}{2}\left[(f_1 - y_1)^2 + (f_2 - y_2)^2\right]$

Let $e_j = f_j - y_j$

Backward:

1)

$$\frac{\partial L}{\partial f_j} = e_j$$

2)

softmax Jacobian: $\dfrac{\partial f_i}{\partial a_j} = f_i\left(\delta_{ij} - f_j\right)$

So vector form with $f = [f_1, f_2]^T$, $e = [e_1, e_2]^T$:

$$g_a = \frac{\partial L}{\partial a} = \left(\operatorname{diag}(f) - f f^T\right) e$$

For 2 classes:

$$g_{a_1} = e_1 f_1 (1 - f_1) - e_2 f_1 f_2$$
$$g_{a_2} = e_2 f_2 (1 - f_2) - e_1 f_1 f_2$$

3)

$$\boxed{\frac{\partial L}{\partial u_{k,j}} = z_k g_{a_j}}$$

$$\frac{\partial L}{\partial u_{1,1}} = z_1 g_{a_1} \qquad \frac{\partial L}{\partial u_{1,2}} = z_1 g_{a_2}$$

$$\frac{\partial L}{\partial u_{2,1}} = z_2 g_{a_1} \qquad \frac{\partial L}{\partial u_{2,2}} = z_2 g_{a_2}$$

4)

$$\frac{\partial L}{\partial z_k} = u_{k,1} g_{a_1} + u_{k,2} g_{a_2}$$

5)

ReLU: $1[s_k > 0]$. $\frac{\partial s_k}{\partial w_k} = x$

$$\frac{\partial L}{\partial w_k} = \left(u_{k,1} g_{a_1} + u_{k,2} g_{a_2}\right) 1[w_k x > 0] x$$

So: $$\frac{\partial L}{\partial w_1} = \left(u_{1,1} g_{a_1} + u_{1,2} g_{a_2}\right) 1[w_1 x > 0] x$$

$$\frac{\partial L}{\partial w_2} = \left(u_{2,1} g_{a_1} + u_{2,2} g_{a_2}\right) 1[w_2 x > 0] x$$

IPYNP - Pen and Paper Task

$$x^{k+1} = x^k - \alpha_k \nabla f(x^k)$$

gradient formula: $\nabla f(x_1, x_2) = (2\cdot(x_1-6)-x_2, 2x_2-x_1)$

$$\nabla f(6,6) = (2\cdot(6-6)-6, 2\cdot 6-6)$$
$$= (-6, 6)$$

$$x^{(1)} = (6,6) - \frac{1}{2}(-6,6) = (6,6) - (-3,3)$$
$$= (9,3)$$

$$\nabla f(9,3) = (2\cdot(9-6)-3, 2\cdot 3-9))$$
$$= (3, -3)$$
$$x^{(2)} = (9,3) - \frac{1}{3}(3,-3) = (9,3) - (1,-1)$$
$$= (8,4)$$
$$\nabla f(8,4) = (2(8-6)-4, 2\cdot 4-8))$$
$$= (0,0)$$
$$x^{(3)} = (8,4) - \frac{1}{4}\cdot(0,0) = (8,4)$$

If we keep going nothing will happen, we have already reached the minimum, since the gradient descent update is $(0,0)$.