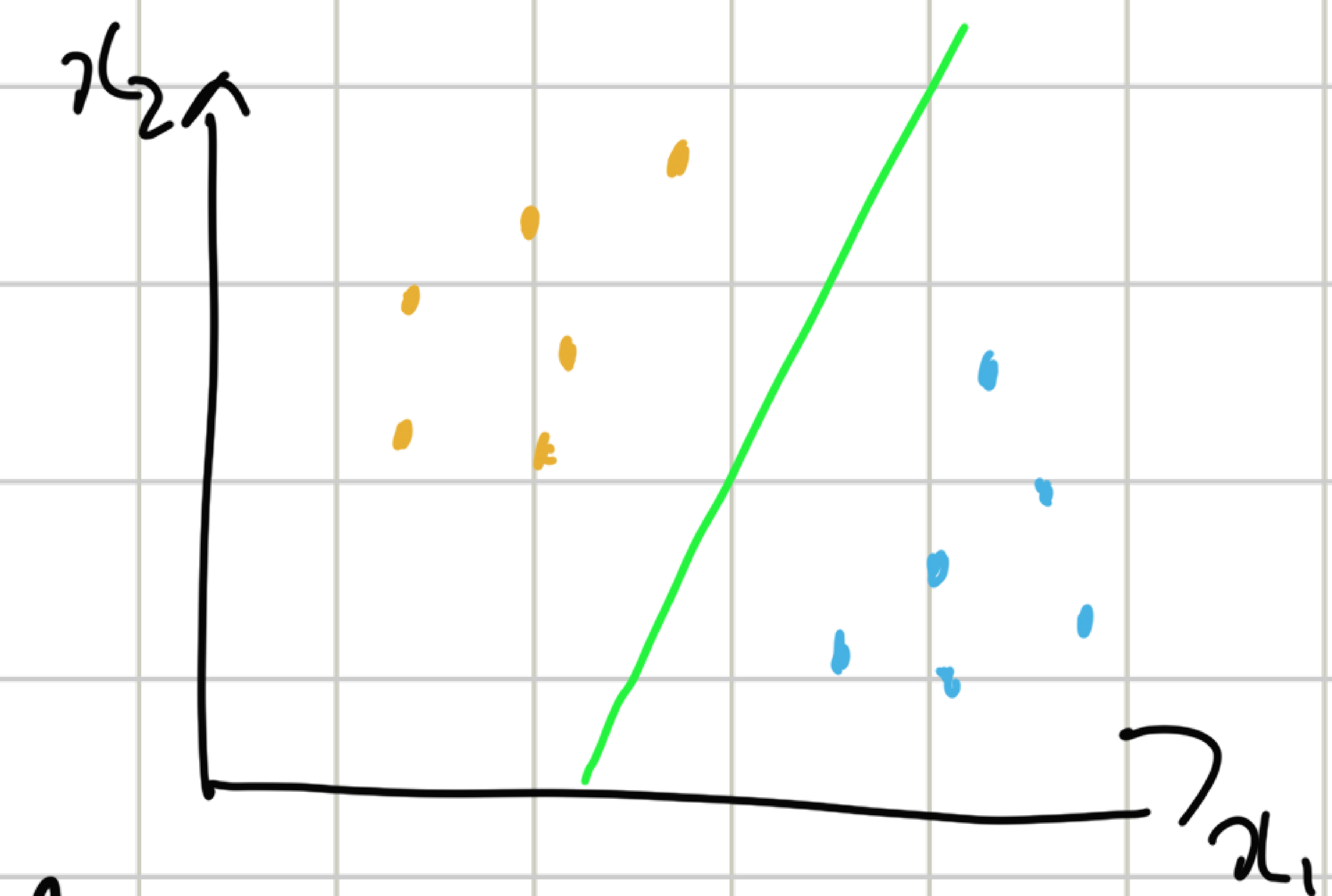
$$z = w_1 x_1 + w_2 x_2 + b$$ fonc° linéaire



$$a = \frac{1}{1+e^{-z}}$$ fonc° sigmoid

$$L = -\frac{1}{m}\sum_{i=1}^{m} y_i \log(a_i) + (1-y_i)\log(1-a_i)$$ fonc° de coût log-loss

$$W = W - \alpha \frac{\partial L}{\partial W}$$ algo descente de gradient.

$$b = b - \alpha \frac{\partial L}{\partial b}$$

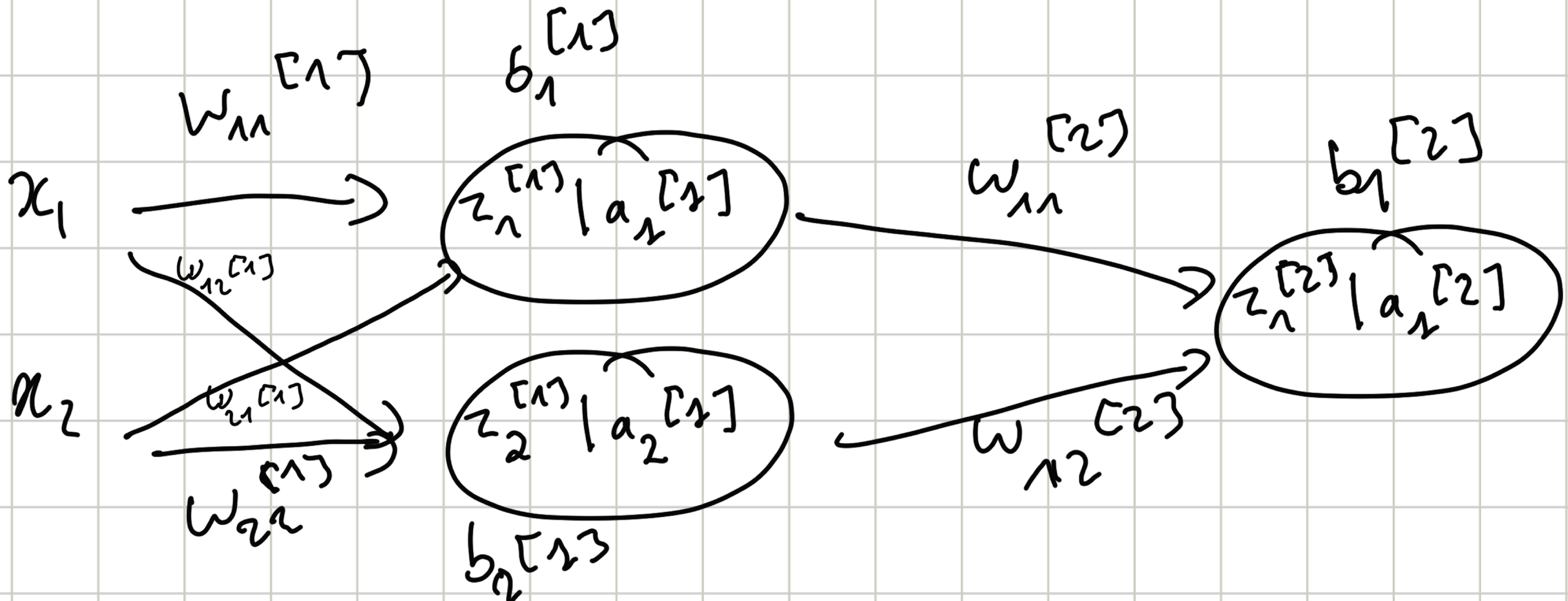
- $$\frac{\partial L}{\partial w_1} = \frac{\partial L}{\partial a} \times \frac{\partial a}{\partial z} \times \frac{\partial z}{\partial w_1} = -\frac{1}{m}\sum\left(\frac{y}{a} - \frac{1-y}{1-a}\right) \times a(1-a)\, x_1$$

$$\frac{\partial L}{\partial w_1} = \frac{1}{m}\sum_{i=1}^{m}(a_i - y_i)\, x_1$$

- $$\frac{\partial L}{\partial w_2} = \frac{\partial L}{\partial a} \times \frac{\partial a}{\partial z} \times \frac{\partial z}{\partial w_2} = \frac{1}{m}\sum_{i=1}^{m}(a_i - y_i)\, x_2$$

- $$\frac{\partial L}{\partial b} = \frac{1}{m}\sum (a - y)$$

Deep neural network (vectorisation)



$$Z^{[1]} = \begin{bmatrix} x_1^{(1)} & x_2^{(1)} \\ \vdots & \vdots \\ x_1^{(m)} & x_2^{(m)} \end{bmatrix} \cdot \begin{bmatrix} w_{11}^{[1]} & w_{21}^{[1]} \\ w_{12}^{[1]} & w_{22}^{[1]} \end{bmatrix} + \begin{bmatrix} b_1^{[1]} & b_2^{[1]} \end{bmatrix} = \begin{bmatrix} z_1^{[1](1)} & z_2^{[1](1)} \\ \vdots & \vdots \\ z_1^{[1](m)} & z_2^{[1](m)} \end{bmatrix}$$

$$W^{[1]} \in \mathbb{R}^{2\times 2} \qquad b^{[1]} \in \mathbb{R}^{1\times 2} \qquad Z^{[1]} \in \mathbb{R}^{m\times 2}$$

$$X \in \mathbb{R}^{m\times 2}$$

Transposons pour une représentation plus claire :

$$Z^{[1]} = \begin{bmatrix} w_{11}^{[1]} & w_{12}^{[1]} \\ w_{21}^{[1]} & w_{22}^{[1]} \end{bmatrix} \cdot \begin{bmatrix} x_1^{(1)} & \dots & x_1^{(m)} \\ x_2^{(1)} & \dots & x_2^{(m)} \end{bmatrix} + \begin{bmatrix} b_1^{[1]} \\ b_2^{[1]} \end{bmatrix} = \begin{bmatrix} z_1^{[1](1)} & \dots & z_1^{[1](m)} \\ z_2^{[1](1)} & \dots & z_2^{[1](m)} \end{bmatrix}$$

$$Z^{[1]} \in \mathbb{R}^{2\times m}$$

Forward propagation :

Soit $\forall i \in \{\text{nombre de couche}\} = \{1, \dots, n\}$:

$$Z^{[i]} = W^{[i]} \cdot A^{[i-1]} + b^{[i]} \in \mathbb{R}^{n \times m} \quad \text{où } A^{[0]} = X.$$

$$A^{[i]} = \frac{1}{1+e^{-Z^{[i]}}} \quad \text{avec } A^{[0]} = X.$$

avec :

$$W^{[i]} \in \mathbb{R}^{n^{[i]} \times n^{[i-1]}} \quad \text{et} \quad b^{[i]} \in \mathbb{R}^{n^{(i)} \times 1}$$

où $n[i]$: nb de neuronne dans la couche i avec $n[0]$ le nombre d'entrée dans le réseau.

Back propagation :

étapes de l'entraînement pour plusieurs couches :

1. $\mathcal{L} = -\frac{1}{m} \sum y \log(A^{[n]}) + (1-y)\log(1-A^{[n]})$, n dernière couche

2. $\frac{\partial \mathcal{L}}{\partial W^{[2]}} =$ $\frac{\partial \mathcal{L}}{\partial W^{[1]}} =$ voir calculs page suivante.

$\frac{\partial \mathcal{L}}{\partial b^{[2]}} =$ $\frac{\partial \mathcal{L}}{\partial b^{[1]}} =$

3. Descente de gradient : minimise la fonc° de coût.

actualisa° de :

$W^{[1]} -= \alpha \frac{\partial \mathcal{L}}{\partial W^{[1]}}$ $b^{[1]} -= \alpha \frac{\partial \mathcal{L}}{\partial b^{[1]}}$

$W^{[2]} -= \alpha \frac{\partial \mathcal{L}}{\partial W^{[1]}}$ $b^{[2]} -= \alpha \frac{\partial \mathcal{L}}{\partial b^{[2]}}$

$$\frac{\partial \mathcal{L}}{\partial W^{[2]}} = \frac{\partial \mathcal{L}}{\partial A^{[2]}} \times \frac{\partial A^{[2]}}{\partial Z^{[2]}} \times \frac{\partial Z^{[2]}}{\partial W^{[2]}}$$, on pose $dZ_2 = \frac{\partial \mathcal{L}}{\partial A^{[2]}} \times \frac{\partial A^{[2]}}{\partial Z^{[2]}}$

$$= dZ_2 \frac{\partial Z^{[2]}}{\partial W^{[2]}}$$ où $dZ_2 = \frac{1}{m} \Sigma\, y \times \frac{1}{A^{[2]}} - (1-y) \times \frac{1}{1-A^{[2]}} \times \frac{e^{-Z^{[2]}} + 1 - 1}{(1+e^{-Z^{[2]}})^2} = -\frac{1}{m} \Sigma (A^{[2]} - y)$

$= A^{[2]} - y$

pour préserver la dimension (n^2, m)

$$\boxed{\frac{\partial \mathcal{L}}{\partial W^{[2]}} = \frac{1}{m} \text{"}\Sigma\text{"} (A^{[2]} - y) \cdot {}^t A^{[1]}} = \frac{1}{m} (A^{[2]} - y) \cdot {}^t A^{[1]}$$

$$\frac{\partial \mathcal{L}}{\partial b^{[2]}} = dZ_2 \times \frac{\partial Z^{[2]}}{\partial b^{[2]}}$$

$$\boxed{\frac{\partial \mathcal{L}}{\partial b^{[2]}} = \frac{1}{m} \Sigma (A^{[2]} - y)}$$

$$\frac{\partial \mathcal{L}}{\partial W^{[1]}} = \frac{\partial \mathcal{L}}{\partial A^{[2]}} \times \frac{\partial A^{[2]}}{\partial Z^{[2]}} \times \frac{\partial Z^{[2]}}{\partial A^{[1]}} \times \frac{\partial A^{[1]}}{\partial Z^{[1]}} \times \frac{\partial Z^{[1]}}{\partial W^{[1]}}$$, $dZ_1 = dZ_2 \times \frac{\partial Z^{[2]}}{\partial A^{[1]}} \times \frac{\partial A^{[1]}}{\partial Z^{[1]}}$

$$= dZ_1 \frac{\partial Z^{[1]}}{\partial W^{[1]}}$$ où $dZ_1 = \frac{1}{m} \Sigma (A^{[2]} - y) \times W^{[2]} \times A^{[1]}(1 - A^{[1]})$

$$\boxed{\frac{\partial \mathcal{L}}{\partial W^{[1]}} = \frac{1}{m} \Sigma\, {}^t W^{[2]} \cdot (A^{[2]} - y) \times A^{[1]} (1 - A^{[1]}) \cdot {}^t X}$$

multiplicat° terme à terme !

$$\frac{\partial \mathcal{L}}{\partial b^{[1]}} = dZ_1 \times \frac{\partial Z^{[1]}}{db^{[1]}}$$

$$\boxed{\frac{\partial \mathcal{L}}{\partial b^{[1]}} = \frac{1}{m} \sum_{axe 1} {}^{t}W^{[2]} \cdot (A^{[2]} - y) \times A^{[1]}(1 - A^{[1]})}$$

Pour généraliser on aura :

Initialisation :

$$W^{[c]} \in \mathbb{R}^{n^{[c]} \times n^{[c-1]}}$$
$$b^{[c]} \in \mathbb{R}^{n^{[c]} \times 1}$$

Forward propagation :

$$Z^{[c]} = W^{[c]} \cdot A^{[c-1]} + b^{[c]}$$

$$A^{[c]} = \frac{1}{1 + e^{-Z^{[c]}}} \text{ avec } A^{[0]} = X.$$

back propagation : (for c in range(cf, 1) :)

$$dZ^{[cf]} = A^{[cf]} - y$$
$$dW^{[c]} = \frac{1}{m} \times dZ^{[c]} \cdot {}^{t}A^{[c-1]}$$
$$db^{[c]} = \frac{1}{m} \sum_{axe 1} dZ^{[c]}$$
$$dZ^{[c-1]} = {}^{t}W^{[c]} \cdot dZ^{[c]} \times A^{[c-1]}(1 - A^{[c-1]})$$