Laboratoire d'Informatique Parallélisme Réseaux Algorithmique Distribuée

UNIVERSITÉ DE VERSAILLES ST-QUENTIN-EN-YVELINES

université PARIS-SACLAY

MAISON DE LA SIMULATION

APPRENTISSAGE AUTOMATIQUE (ML)

Nahid.Emad@uvsq.fr

Cours d'introduction à l'apprentissage automatique

PLAN

1. ENVIRONNEMENT : LANGAGES, LOGICIELS, RÉFÉRENCES, ETC.
2. INTRODUCTION : GÉNÉRALITÉ SUR L'APPRENTISSAGE AUTOMATIQUE
3. DIFFÉRENTS TYPES DE MÉTHODES D'APPRENTISSAGE AUTOMATIQUE
4. APPRENTISSAGE SUPERVISÉ
   a) RAPPEL DE CALCUL MATRICIEL
   b) RÉGRESSION (NON) LINÉAIRE: ARBRES DE DÉCISION, SVM, (RÉSEAUX DE NEURONES), …
5. APPRENTISSAGE NON-SUPERVISÉ
   a) ANALYSER LA STRUCTURE DES DONNÉES ET CHERCHER LES RESSEMBLANCES (K-MEAN CLUSTERING)
   b) DÉTECTER DES ÉCHANTILLONS AVEC DES FEUTURES TRÈS DIFFÉRENTS (DES ANOMALIES) DES AUTRES (PAR EX: ISOLATION FOREST).
   c) RÉDUCTION DE DIMENSION (ACP)
6. APPRENTISSAGE PAR RENFORCEMENT
7. APPRENTISSAGE (NON)SUPERVISÉ AVEC EXEMPLES (TD/TP)

Quatre étapes principales :

1. **Définition de Dataset** : Soit $\{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \ldots, (x^{(m)}, y^{(m)})\}$ notre *dataset*

2. **Choix du Modèle (et ses paramètres):** Régression linéaire ou polynomiale. La fonction f est définie par : $f(x) = a.x + b$ ou $f(\mathrm{x}) = \mathrm{a}.x^2 + b.x + c$

3. **Optimisation du modèle** : Trouver les paramètres minimisant la fonction d'erreur ou de coût (a, b ou a, b, c pour la régression avec un algorithme d'apprentissage comme la Décente de Gradient).

4. **Évaluation du modèle :** Tester le modèle obtenu.

Modèles de classification :

Avec **régression linéaire** avec des labels dénombrables

a) **Binaire :** $y = \{a, b\}$ (par exemple $a = -1, b = +1$)

On sépare un nuage de points en 2 classes. Par ex. une plante est classée toxique ou non-toxique en fonction de ses caractéristiques, Pour cela, on définit une droite f appelé **frontière de décision** (f est zéro sur la frontière, > ou < au-dessus et en-dessous).

b) **Multi-classe :** $y = \{1,2, \ldots, c\}$ (avec $c > 2$).

On détermine des étiquettes pouvant avoir plus de deux valeurs. Par exemple les mentions très bien, bien, assez bien, passable qu'on peut avoir après un test.

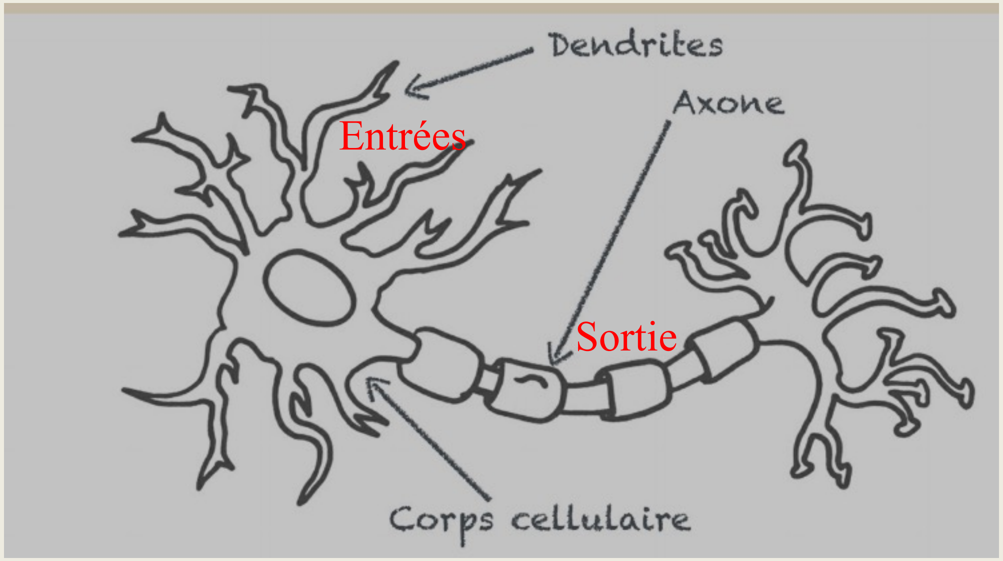
Avec **régression non-linéaire** avec des labels réels (non-dénombrables)

…

Modèles de réseaux de neurones artificiels (RN) :

Il s'inspire du neurone (la cellule organique qui compose le cerveau de l'être humain) et ses communications avec les autres neurones.

La structure générale d'un neurone est constituée d'un corps cellulaire (contenant son ADN), d'un réseau de dendrites (filaments permettant le transit des signaux électriques) et un autre filament plus long appelé **Axone** qui véhicule l'influx nerveux en sortie.



Modèles de réseaux de neurones artificiels (exemple avec un seul neurone) :

Dendrites correspondent aux entrées du neurone.

Axone correspond à sa sortie.

Cerveau est un réseau de neurones connectés les uns aux autres à l'aide de leurs dendrites et axones.

Neurone artificiel simple : le neurone calcule la somme pondérée de ses entrées, alors il s'active (en donnant 1 comme code d'activation) si le résultat est supérieur à un seuil sinon il produira zéro (signifiant non-activation). Une entrée peut être 0 ou 1. La somme pondérée des entrées est donc la somme des poids des entrées 1.

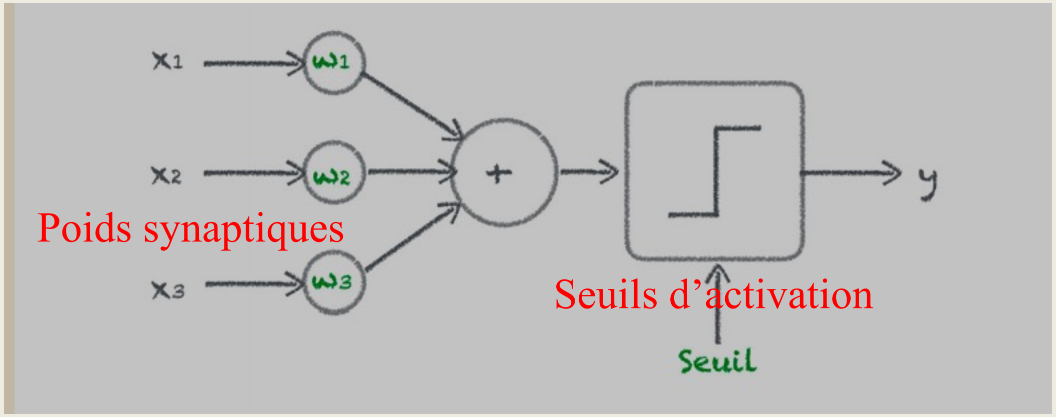
Soit *f* la fonction de transfert.

1- Agrégation :

$z = w_1 x_1 + w_2 x_2 + w_3 x_3$

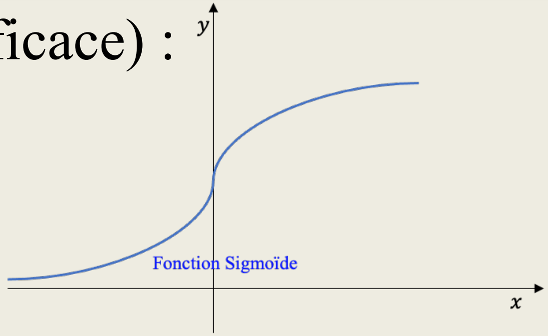
2- Activation :

$y = 1 \; si \; z \geq seuil, y = 0$ sinon.

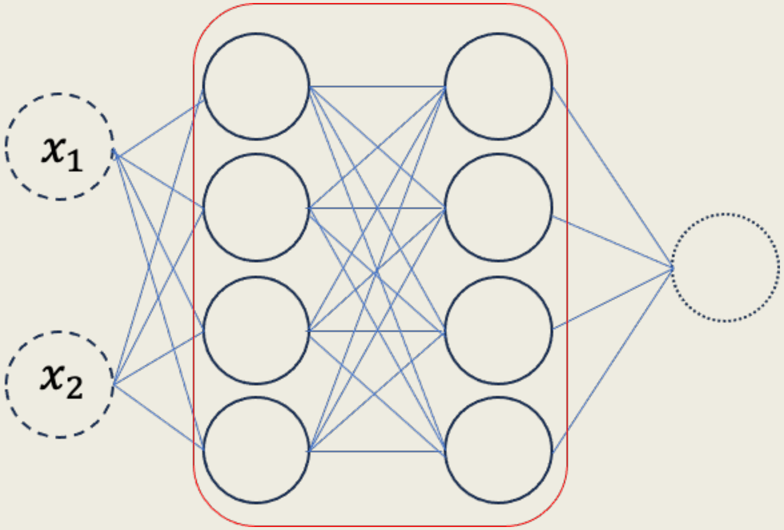


Fonction d'activation non-binaire (plus efficace) :

$$a(z) = \frac{1}{(1 + e^{-z})}$$



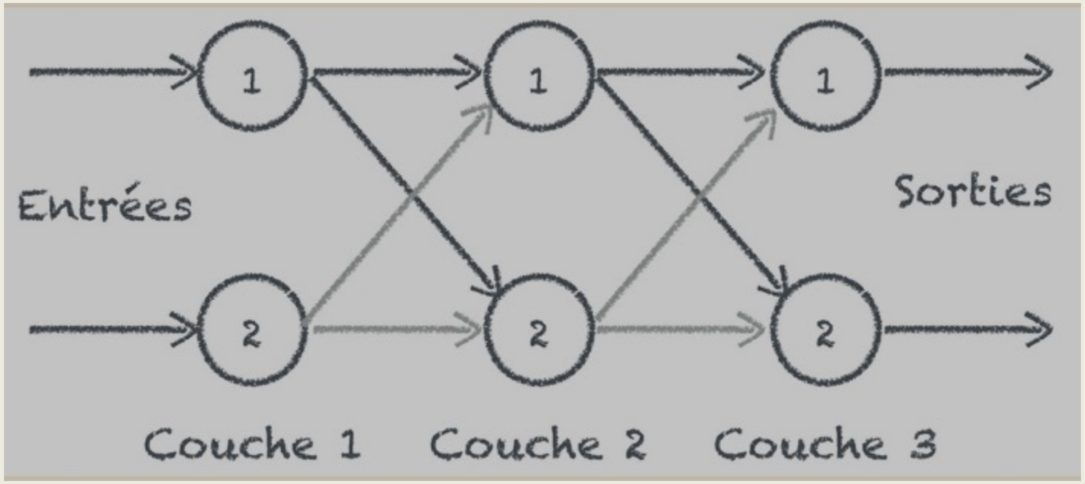
Comme pour le réseau de neurones du cerveau qui est organisé en couches, il faut définir un RN artificiel en connectant plusieurs neurones les uns aux autres en couches.



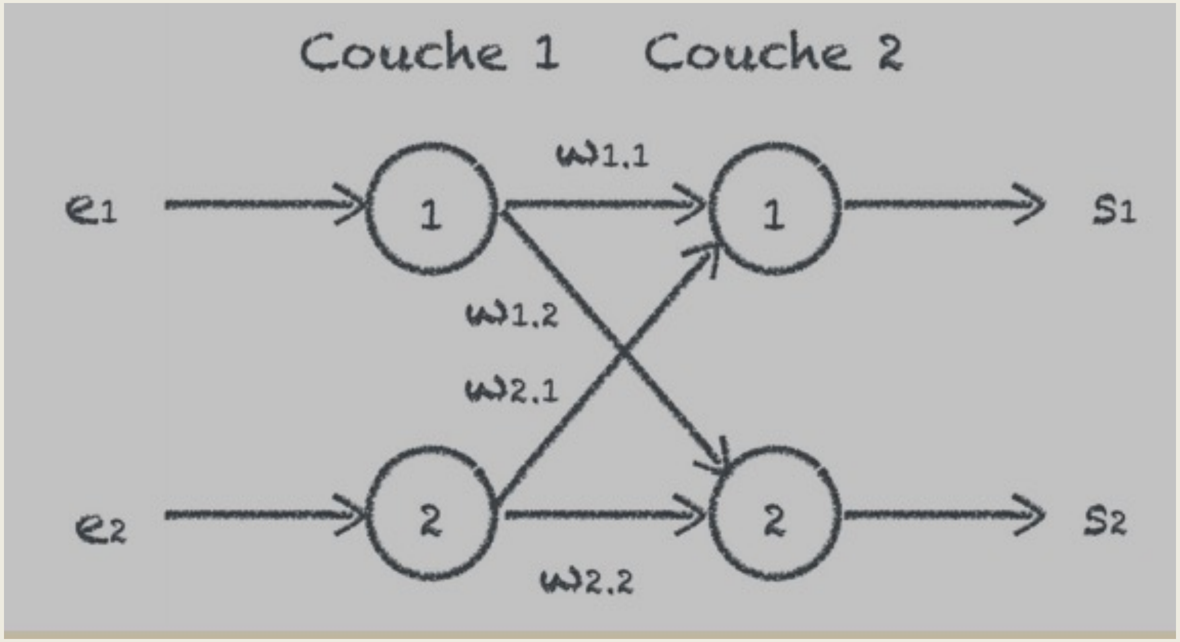
Perceptron

Exemple de réseau de 2+2+2 neurones en 3 couches

- Couche d'entrée : propagation des données à la couche suivante.
- Couche interne: connexion des sorties de neurones de la couche1 à tous les neurones de la couche3.
- Couche de sortie



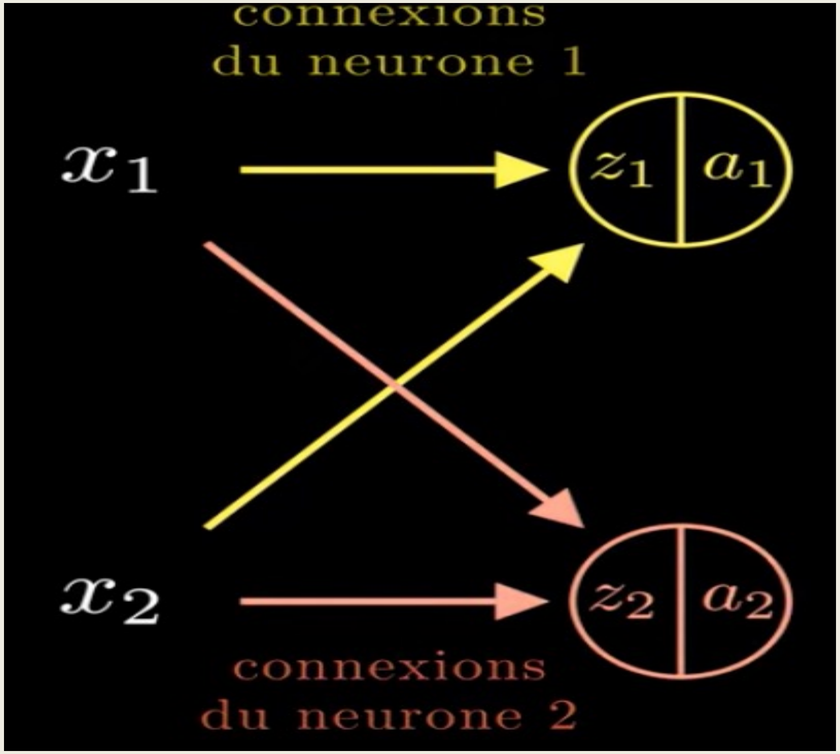
Perceptron : Exemple de réseau de 2+2 neurones en 2 couches



1. Choix initial des poids
2. Données d'entrées e_1 et e_2.
3. Calcul $x_1 = w_{11}e_1 + w_{21}e_2$ et $x_2 = w_{12}e_1 + w_{22}e_2$.
4. $s_1 = \frac{1}{(1+e^{-x_1})}$ et $s_2 = \frac{1}{(1+e^{-x_2})}$

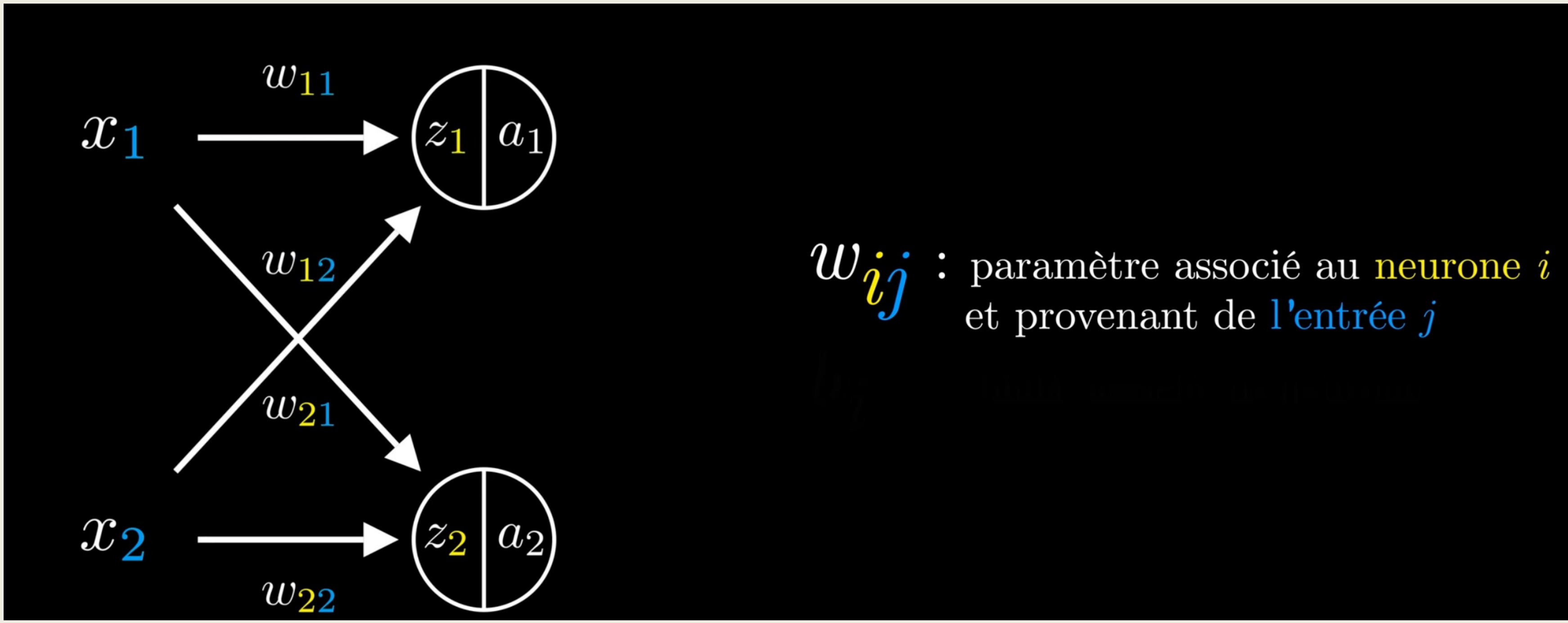
Modèles de RN artificiels (exemple avec deux neurones) :

1ère couche : x_1 et x_2 sont les entrées, avec des poids $w_{i,j}$ différents, pour chacun des 2 neurones.



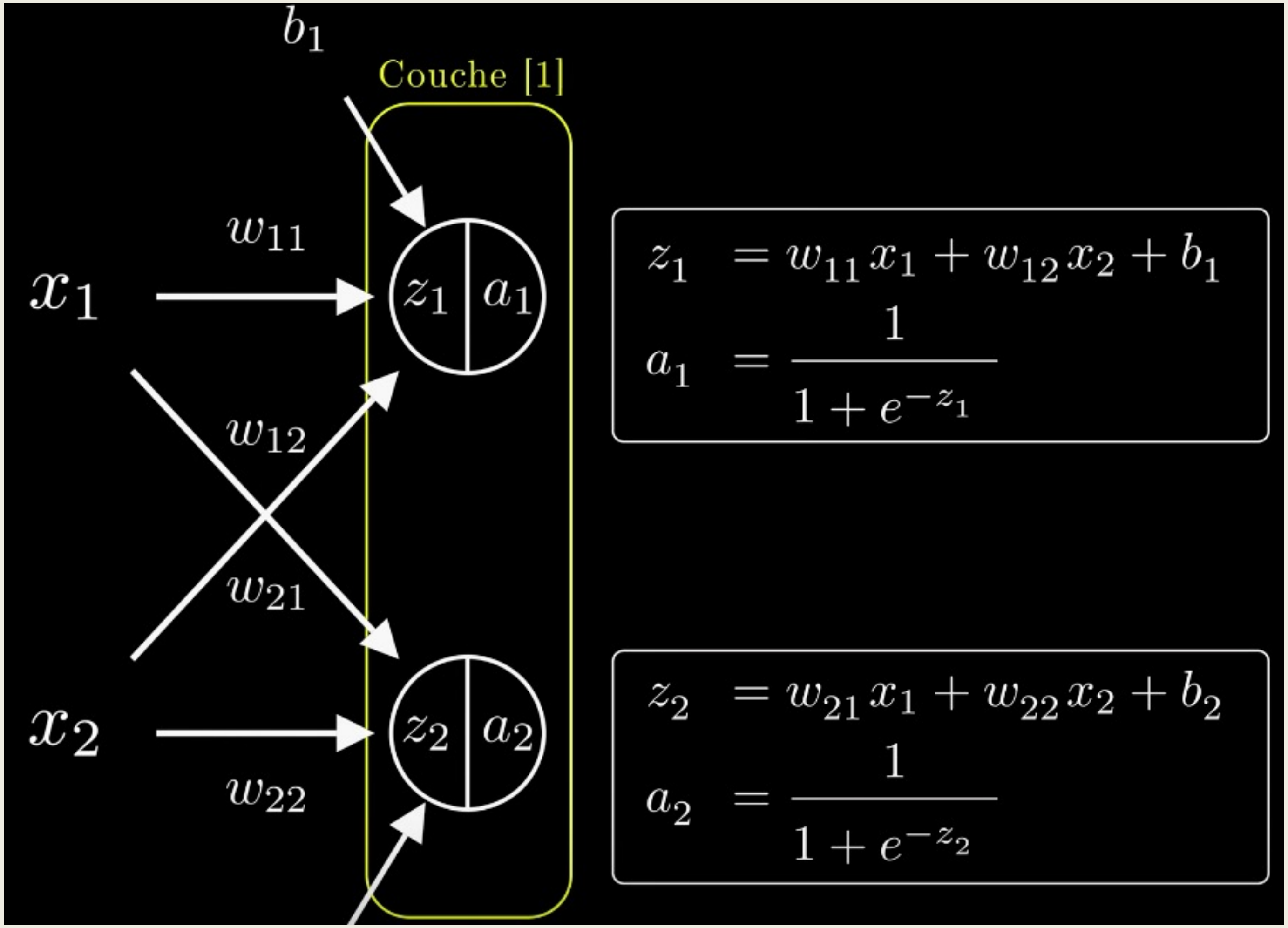
Modèles de RN artificiels (exemple avec deux neurones) :

1ère couche : x_1 et x_2 sont les entrées, avec des poids $w_{i,j}$ différents, pour chacun des 2 neurones.



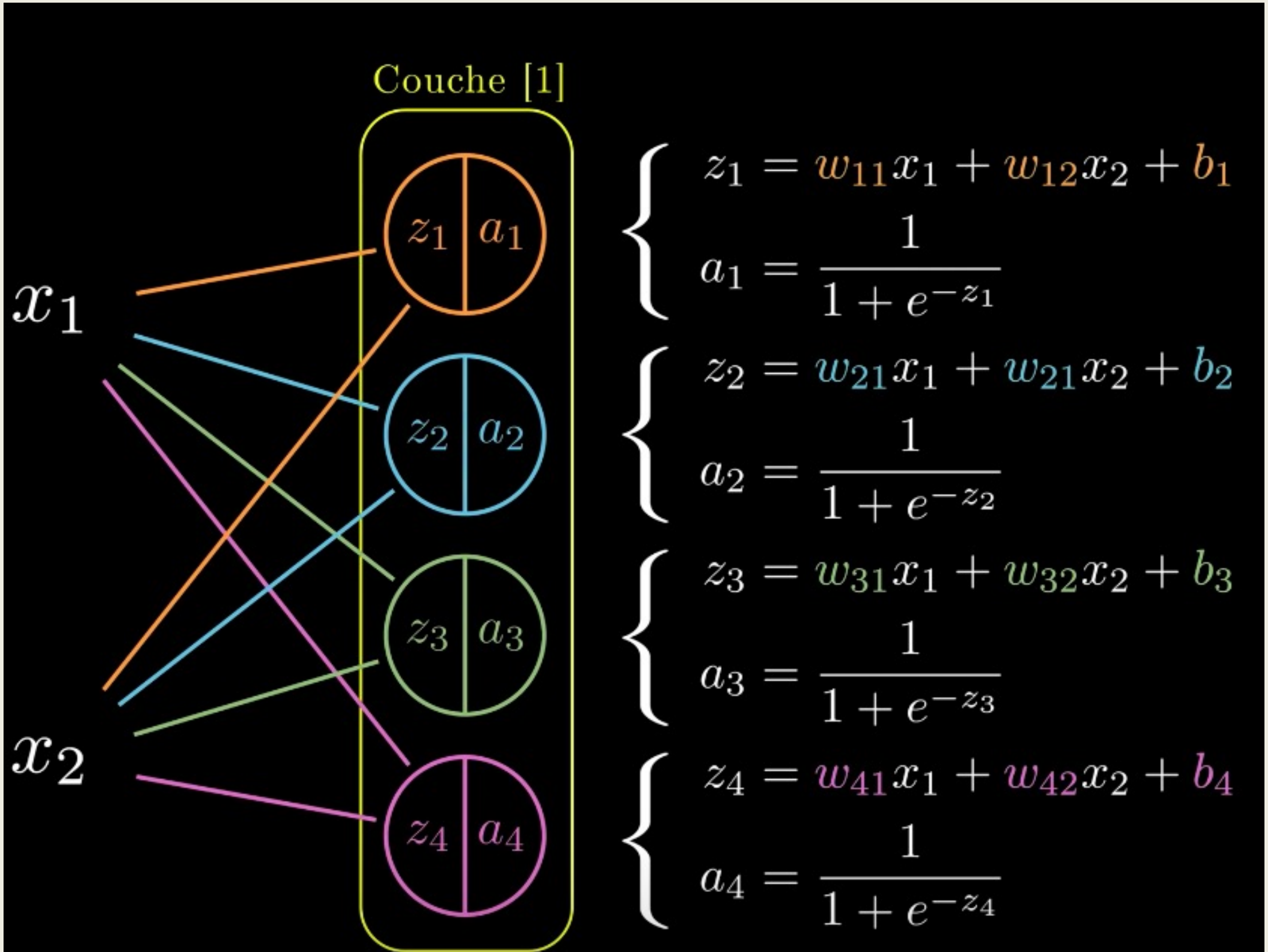
Attention:
Notation différente de celles de la page 9 (des indices de variables représentant les poids)

Réseau de 2 neurones d'une couche

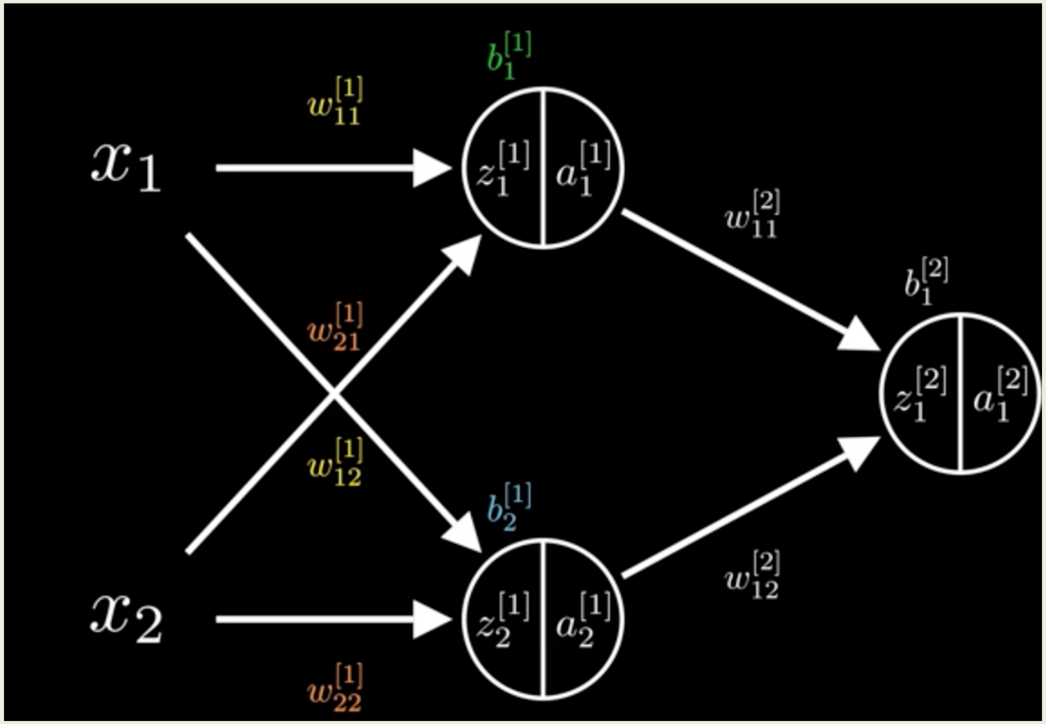


Premier Réseau de neurones

Réseau de 4 neurones d'une couche



Calcul matriciel : Réseau de (2+1) neurones et deux couches



INDICATION DU NOMBRE DE COUCHES

1. $w_{ij}^{[C]}$ le poids lié au neurone i et provenant de l'entrée j associé à la couche C.
2. Les valeurs z d'une couche C utilise les activations a de la couche (C-1).

$$z_1^{[2]} = w_{11}^{[2]} a_1^{[1]} + w_{12}^{[2]} a_2^{[1]} + b^{[2]}$$

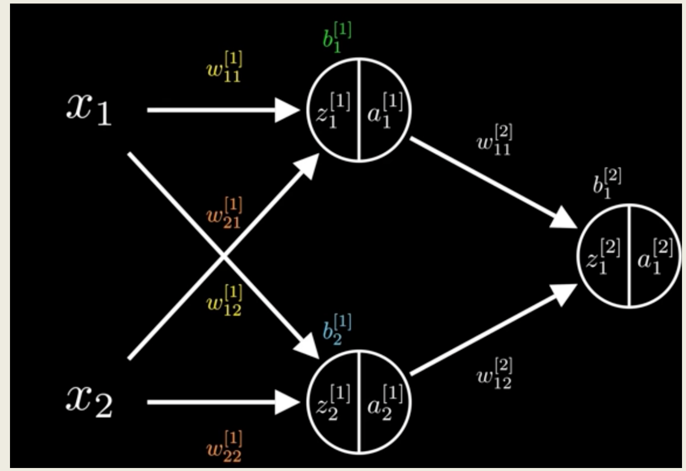
Et des échantillons avec plusieurs caractéristiques ???

Premier Réseau de neurones

Calcul matriciel : Réseau de (2+1) neurones en deux couches

1. $w_{ij}^{[C]}$ le poids lié au neurone i et provenant de l'entrée j associé à la couche C.

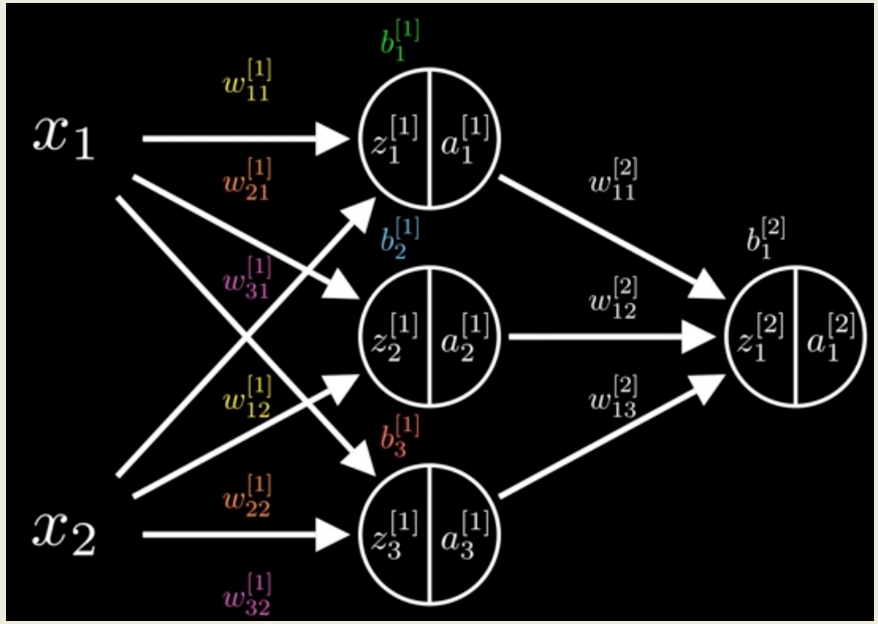
2. Les valeurs z d'une couche C utilise les activations a de la couche (C-1).



$$z_1^{[2]} = w_{11}^{[2]} a_1^{[1]} + w_{12}^{[2]} a_2^{[1]} + b^{[2]}$$

$$Z^{[1]} = \underbrace{\begin{bmatrix} x_1^{(1)} & x_2^{(1)} \\ x_1^{(2)} & x_2^{(2)} \\ \vdots & \vdots \\ x_1^{(m)} & x_2^{(m)} \end{bmatrix}}_{X \in \mathbb{R}^{m\times 2}} \cdot \underbrace{\begin{bmatrix} w_{11}^{[1]} & w_{21}^{[1]} \\ w_{12}^{[1]} & w_{22}^{[1]} \end{bmatrix}}_{W^{[1]} \in \mathbb{R}^{2\times 2}} + \underbrace{\begin{bmatrix} b_1^{[1]} & b_2^{[1]} \end{bmatrix}}_{b^{[1]} \in \mathbb{R}^{1\times 2}} = \underbrace{\begin{bmatrix} z_1^{[1](1)} & z_2^{[1](1)} \\ z_1^{[1](2)} & z_2^{[1](2)} \\ \vdots & \vdots \\ z_1^{[1](m)} & z_2^{[1](m)} \end{bmatrix}}_{Z^{[1]} \in \mathbb{R}^{m\times 2}}$$

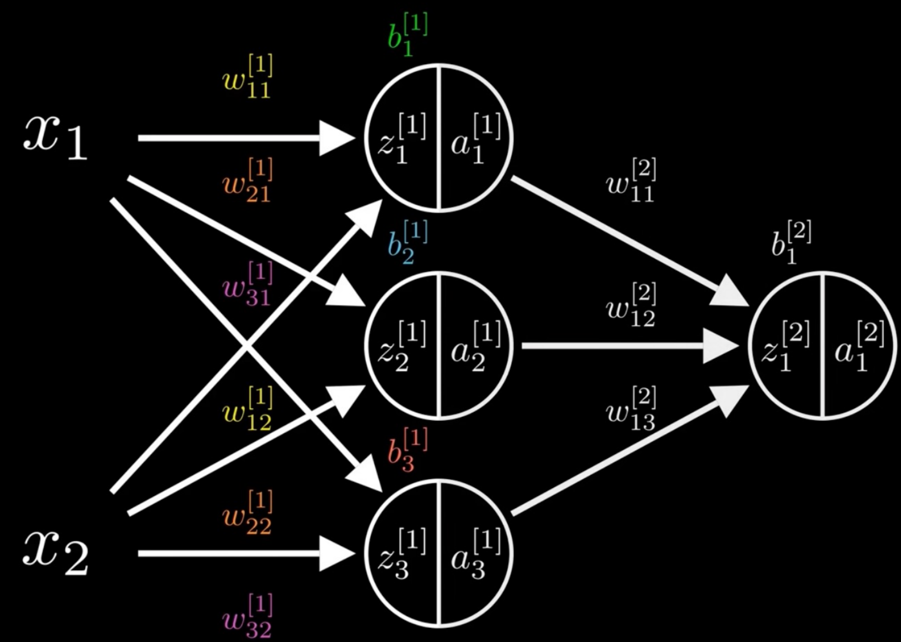
Calcul matriciel : Réseau de 3+1 neurones en deux couches



$$Z^{[1]} = \underbrace{\begin{bmatrix} x_1^{(1)} & x_2^{(1)} \\ x_1^{(2)} & x_2^{(2)} \\ \vdots & \vdots \\ x_1^{(m)} & x_2^{(m)} \end{bmatrix}}_{X \in \mathbb{R}^{m\times 2}} \cdot \underbrace{\begin{bmatrix} w_{11}^{[1]} & w_{21}^{[1]} & w_{31}^{[1]} \\ w_{12}^{[1]} & w_{22}^{[1]} & w_{32}^{[1]} \end{bmatrix}}_{W^{[1]} \in \mathbb{R}^{2\times 3}} + \underbrace{\begin{bmatrix} b_1^{[1]} & b_2^{[1]} & b_3^{[1]} \end{bmatrix}}_{b^{[1]} \in \mathbb{R}^{1\times 3}} = \underbrace{\begin{bmatrix} z_1^{[1](1)} & z_2^{[1](1)} & z_3^{[1](1)} \\ z_1^{[1](2)} & z_2^{[1](2)} & z_3^{[1](2)} \\ \vdots & \vdots & \vdots \\ z_1^{[1](m)} & z_2^{[1](m)} & z^{[1](m)} \end{bmatrix}}_{Z^{[1]} \in \mathbb{R}^{m\times 3}}$$

Réorganisation de calcul : Réseau de 3+1 neurones en deux couches

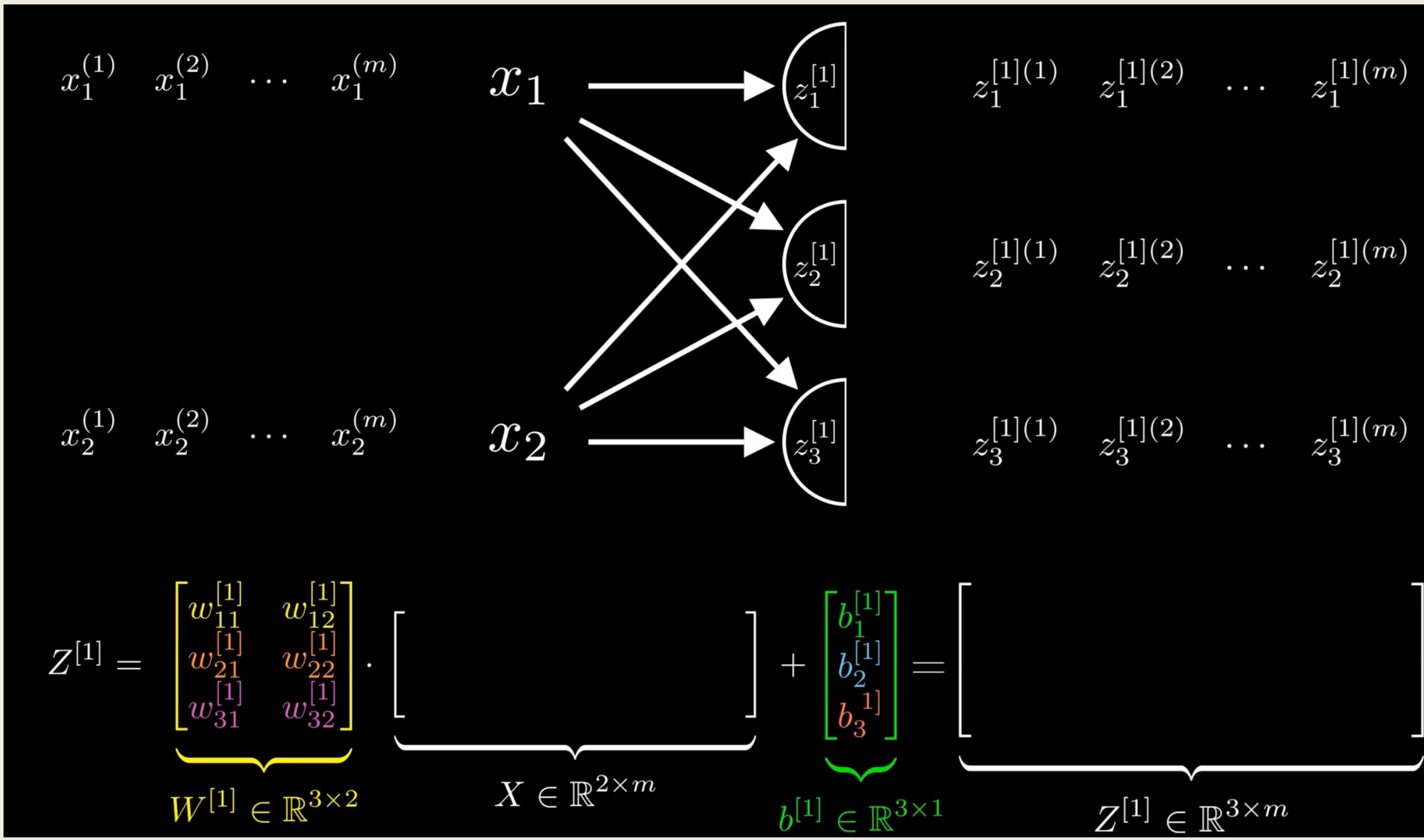
Simplification des calculs : Inversion des lignes et colonnes



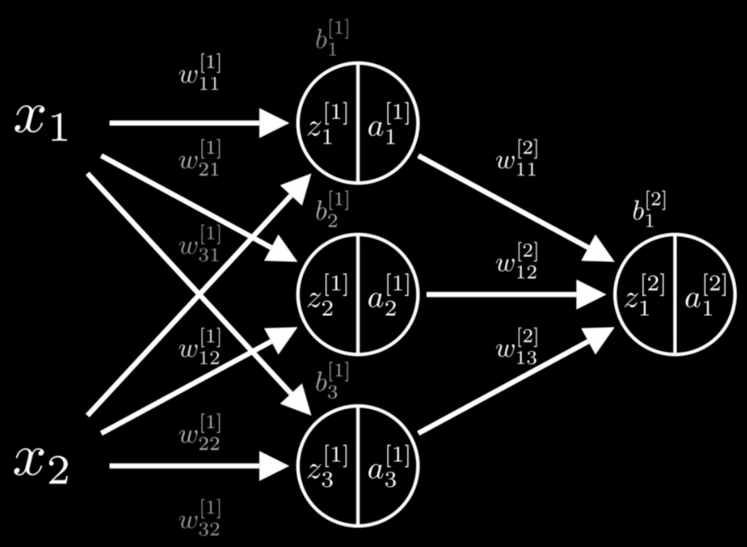
$$Z^{[1]} = \underbrace{\begin{bmatrix} w_{11}^{[1]} & w_{12}^{[1]} \\ w_{21}^{[1]} & w_{22}^{[1]} \\ w_{31}^{[1]} & w_{32}^{[1]} \end{bmatrix}}_{W^{[1]} \in \mathbb{R}^{3\times 2}} \cdot \underbrace{\begin{bmatrix} x_1^{(1)} & x_1^{(2)} & \cdots & x_1^{(m)} \\ x_2^{(1)} & x_2^{(2)} & \cdots & x_2^{(m)} \end{bmatrix}}_{X \in \mathbb{R}^{2\times m}} + \underbrace{\begin{bmatrix} b_1^{[1]} \\ b_2^{[1]} \\ b_3^{1]} \end{bmatrix}}_{b^{[1]} \in \mathbb{R}^{3\times 1}} = \underbrace{\begin{bmatrix} z_1^{[1](1)} & z_1^{[1](2)} & \cdots & z_1^{[1](m)} \\ z_2^{[1](1)} & z_2^{[1](2)} & \cdots & z_2^{[1](m)} \\ z_3^{[1](1)} & z_3^{[1](2)} & \cdots & z_3^{[1](m)} \end{bmatrix}}_{Z^{[1]} \in \mathbb{R}^{3\times m}}$$

Premier Réseau de neurones

Réseau de 3 neurones en une couche : les $x_i^{(j)}$ entrent en continu dans le (les $z_i^{[\,](\,)}$ sortent du) ième neurone.



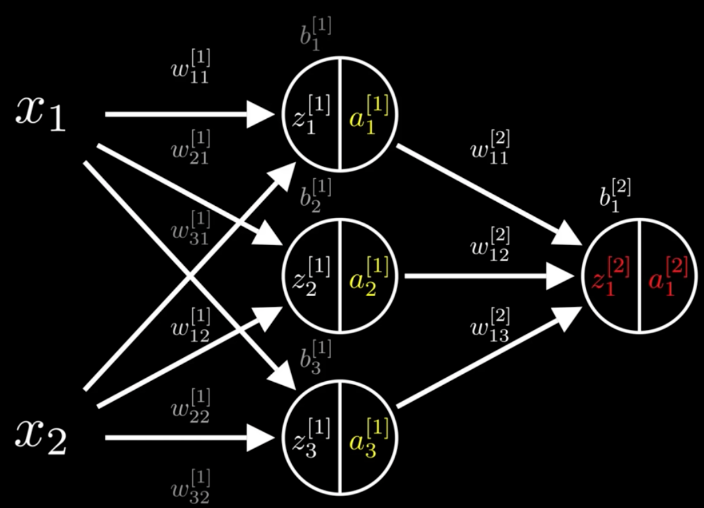
Calcul matriciel : Réseau de 3+1 neurones en deux couches



$$Z^{[1]} = W^{[1]} \cdot X + b^{[1]} = \underbrace{\begin{bmatrix} z_1^{[1](1)} & z_1^{[1](2)} & \cdots & z_1^{[1](m)} \\ z_2^{[1](1)} & z_2^{[1](2)} & \cdots & z_2^{[1](m)} \\ z_3^{[1](1)} & z_3^{[1](2)} & \cdots & z_3^{[1](m)} \end{bmatrix}}_{Z^{[1]} \in \mathbb{R}^{3\times m}}$$

$$A^{[1]} = \frac{1}{1+e^{-Z^{[1]}}} = \underbrace{\begin{bmatrix} a_1^{[1](1)} & a_1^{[1](2)} & \cdots & a_1^{[1](m)} \\ a_2^{[1](1)} & a_2^{[1](2)} & \cdots & a_2^{[1](m)} \\ a_3^{[1](1)} & a_3^{[1](2)} & \cdots & a_3^{[1](m)} \end{bmatrix}}_{A^{[1]} \in \mathbb{R}^{3\times m}}$$

Calcul matriciel : Réseau de 3+1 neurones en deux couches



$$Z^{[1]} = W^{[1]} \cdot X + b^{[1]} = \underbrace{\begin{bmatrix} z_1^{[1](1)} & z_1^{[1](2)} & \cdots & z_1^{[1](m)} \\ z_2^{[1](1)} & z_2^{[1](2)} & \cdots & z_2^{[1](m)} \\ z_3^{[1](1)} & z_3^{[1](2)} & \cdots & z_3^{[1](m)} \end{bmatrix}}_{Z^{[1]} \in \mathbb{R}^{3\times m}}$$

$$A^{[1]} = \frac{1}{1 + e^{-Z^{[1]}}} = \underbrace{\begin{bmatrix} a_1^{[1](1)} & a_1^{[1](2)} & \cdots & a_1^{[1](m)} \\ a_2^{[1](1)} & a_2^{[1](2)} & \cdots & a_2^{[1](m)} \\ a_3^{[1](1)} & a_3^{[1](2)} & \cdots & a_3^{[1](m)} \end{bmatrix}}_{A^{[1]} \in \mathbb{R}^{3\times m}}$$

$$Z^{[2]} = W^{[2]} \cdot A^{[1]} + b^{[2]}$$

$W^{[2]} \in \mathbb{R}^{(n^{[2]} \times n^{[1]})}$

$b^{[2]} \in \mathbb{R}^{(n^{[2]} \times 1)}$

$n^{[1]}$: nombre de neurones dans la couche [1]

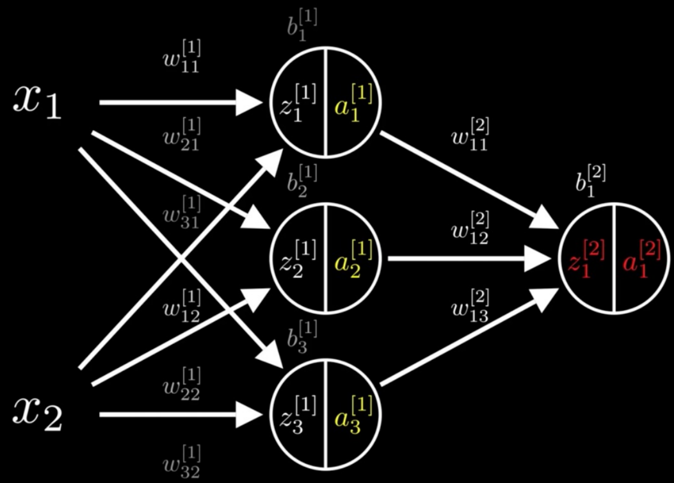
$n^{[2]}$: nombre de neurones dans la couche [2]

Premier Réseau de neurones

Calcul matriciel : Réseau de 3+1 neurones en deux couches

Propagation vers l'avant

Forward Propagation



$$Z^{[1]} = W^{[1]} \cdot X + b^{[1]} = \underbrace{\begin{bmatrix} z_1^{[1](1)} & z_1^{[1](2)} & \cdots & z_1^{[1](m)} \\ z_2^{[1](1)} & z_2^{[1](2)} & \cdots & z_2^{[1](m)} \\ z_3^{[1](1)} & z_3^{[1](2)} & \cdots & z_3^{[1](m)} \end{bmatrix}}_{Z^{[1]} \in \mathbb{R}^{3\times m}}$$

$$A^{[1]} = \frac{1}{1+e^{-Z^{[1]}}} = \underbrace{\begin{bmatrix} a_1^{[1](1)} & a_1^{[1](2)} & \cdots & a_1^{[1](m)} \\ a_2^{[1](1)} & a_2^{[1](2)} & \cdots & a_2^{[1](m)} \\ a_3^{[1](1)} & a_3^{[1](2)} & \cdots & a_3^{[1](m)} \end{bmatrix}}_{A^{[1]} \in \mathbb{R}^{3\times m}}$$

$$Z^{[2]} = W^{[2]} \cdot A^{[1]} + b^{[2]}$$

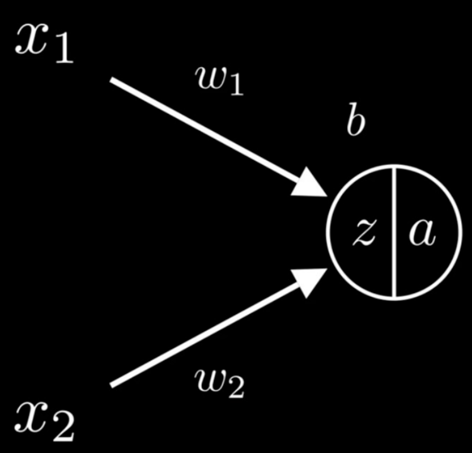
$$A^{[2]} = \frac{1}{1+e^{-Z^{[2]}}}$$

$$Z^{[3]} = W^{[3]} \cdot A^{[2]} + b^{[3]}$$ (dans le cas d'une 3ieme couche)

$\cdots$

Calcul matriciel : Un neurone, une couche

Propagation vers arrière



$$W = \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}$$

1. Définir une Fonction Coût

$$\mathcal{L} = -\frac{1}{m}\sum y \times log(A) + (1-y) \times log(1-A)$$

Permet d'évaluer les erreurs du modèle.

2. Calculer les dérivées partielles

$$\frac{\partial \mathcal{L}}{\partial W} = \frac{1}{m} X^T \cdot (A-y) \qquad \frac{\partial \mathcal{L}}{\partial b} = \frac{1}{m}\sum(A-y)$$

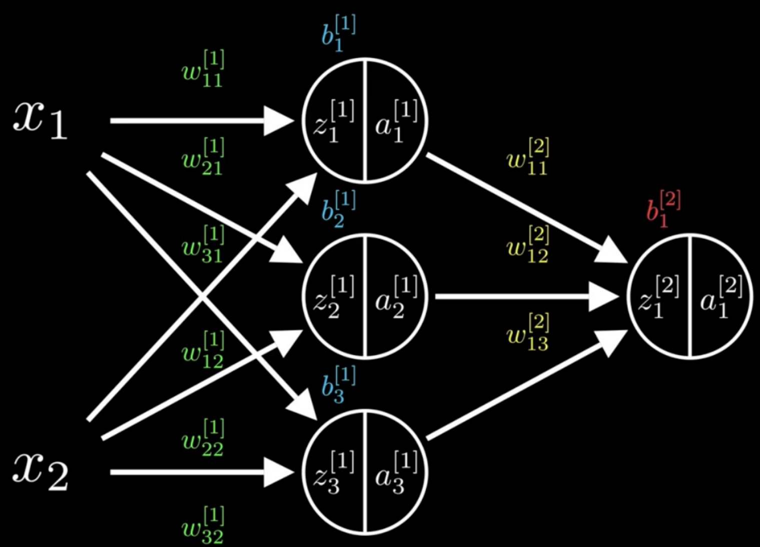
Permet de comprendre comment la fonction coût évolue par rapport aux différents paramètres W et b

3. Mettre à jour les paramètres W et b

$$W = W - \alpha \frac{\partial \mathcal{L}}{\partial W} \qquad b = b - \alpha \frac{\partial \mathcal{L}}{\partial b}$$

Permet de minimiser la fonction coût grâce à la descente de gradients

Calcul matriciel : Réseau de 3+1 neurones en deux couches



$$W^{[1]} = \begin{bmatrix} w_{11}^{[1]} & w_{12}^{[1]} \\ w_{21}^{[1]} & w_{22}^{[1]} \\ w_{31}^{[1]} & w_{32}^{[1]} \end{bmatrix} \qquad W^{[2]} = \begin{bmatrix} w_{11}^{[2]} \\ w_{12}^{[2]} \\ w_{13}^{[2]} \end{bmatrix}$$

$$b^{[1]} = \begin{bmatrix} b_1^{[1]} \\ b_2^{[1]} \\ b_3^{[1]} \end{bmatrix} \qquad b^{[2]}$$

1. Définir une Fonction Coût

$$\mathcal{L} = -\frac{1}{m}\sum y \times log(A^{[2]}) + (1-y) \times log(1-A^{[2]})$$

2. Calculer les dérivées partielles

$$\boxed{\begin{matrix} \dfrac{\partial \mathcal{L}}{\partial W^{[2]}} = & \qquad & \dfrac{\partial \mathcal{L}}{\partial W^{[1]}} = \\ \dfrac{\partial \mathcal{L}}{\partial b^{[2]}} = & \qquad & \dfrac{\partial \mathcal{L}}{\partial b^{[1]}} = \end{matrix}}$$

3. Mettre à jour les paramètres W et b

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

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