

## TP 2

### Réseau de neurones artificiels à deux couches

#### Objectifs pédagogiques

- Comprendre la structure d'un réseau de neurones à deux couches.
- Implémenter la propagation avant (forward propagation) de manière vectorisée.
- Approfondir les concepts de couches, fonctions d'activation, et calculs matriciels.
- Renforcer les compétences en programmation Python pour des tâches pratiques.

#### Étape 1 : Introduction aux couches d'un réseau

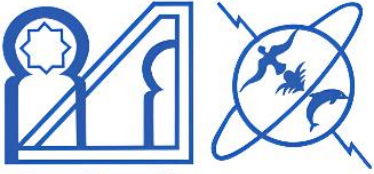

##### Question conceptuelle

- Qu'est-ce qu'une couche dans un réseau de neurones artificiels ?
- Décrivez brièvement le rôle d'une couche cachée et d'une couche de sortie.

##### Exercice manuel

Pour un réseau à deux couches :

- Couche 1 :  $Z^{[1]} = W^{[1]}X + b^{[1]}$  et  $A^{[1]} = \frac{1}{1+e^{-Z^{[1]}}}$
- Couche 2 :  $Z^{[2]} = W^{[2]}A^{[1]} + b^{[2]}$  et  $A^{[2]} = \frac{1}{1+e^{-Z^{[2]}}}$

 <p>Faculté des Sciences Université Mohammed V de Rabat</p>	<p>Deep Learning MSID 2024 / 2025</p>	 <p>جامعة محمد الخامس كلية العلوم الرباط</p>
--	---	---

Calculez  $Z^{[1]}$ ,  $A^{[1]}$ ,  $Z^{[2]}$  et  $A^{[2]}$  à la main pour les données suivantes :

- $X = \begin{bmatrix} 0.5 \\ 1.0 \end{bmatrix}$
- $W^{[1]} = \begin{bmatrix} 0.2 & 0.4 \\ 0.3 & 0.1 \end{bmatrix}$
- $b^{[1]} = \begin{bmatrix} 0.1 \\ -0.2 \end{bmatrix}$
- $W^{[2]} = \begin{bmatrix} 0.5 & -0.3 \end{bmatrix}$
- $b^{[2]} = 0.2$

Utilisez la fonction sigmoïde pour la fonction d'activation.

## Étape 2 : Implémentation de la couche [1]

Implémentez la propagation avant pour la couche 1

Créez une fonction Python qui calcule  $Z^{[1]}$  et  $A^{[1]}$  :

Testez votre implémentation :

- Donnez  $X = \begin{bmatrix} 0.5 \\ 1.0 \end{bmatrix}$ ,  $W^{[1]}$ , et  $b^{[1]}$  définis ci-dessus.
- Affichez les résultats de  $Z^{[1]}$  et  $A^{[1]}$

### Questions

- Pourquoi applique-t-on une fonction d'activation comme la sigmoïde après  $Z^{[1]}$  ?
- Que se passe-t-il si  $W^{[1]}$  ou  $b^{[1]}$  sont très grands ?

### Étape 3 : Implémentation de la couche [2]

Implémentez la propagation avant pour la couche 2

Ajoutez une fonction Python pour calculer  $Z^{[2]}$  et  $A^{[2]}$  :

Testez votre implémentation :

- Utilisez les sorties de la couche 1 comme entrée pour la couche 2.
- Affichez  $Z^{[2]}$  et  $A^{[2]}$ .

## Étape 4 : Vectorisation d'un réseau de neurones

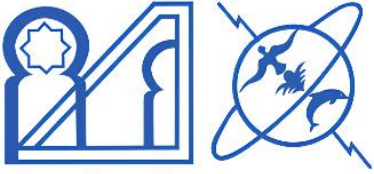

### Qu'est-ce que la vectorisation ?

La vectorisation consiste à utiliser des opérations matricielles (comme des multiplications de matrices) pour effectuer des calculs dans un réseau de neurones. Cela permet de traiter plusieurs exemples d'entrée simultanément en une seule étape.

### Avantages de la vectorisation

- **Efficacité computationnelle** : Les opérations matricielles sont beaucoup plus rapides, car elles sont optimisées par les bibliothèques numériques (comme NumPy).
- **Lisibilité du code** : Les calculs vectorisés sont souvent plus compacts et faciles à lire.
- **Parallélisation** : Les calculs sont plus facilement parallélisables sur des processeurs modernes ou des GPU.

Implémentez une fonction Python qui effectue les calculs vectorisés pour les deux couches :

 <p><b>Faculté des Sciences</b> Université Mohammed V de Rabat</p>	<p><b>Deep Learning</b> <b>MSID</b> <b>2024 / 2025</b></p>	 <p>جامعة محمد الخامس كلية العلوم الرباط</p>
---	--	---

Testez cette fonction avec un lot de données  $X$  (par exemple, une matrice de dimension  $2 \times 3$ , où chaque colonne représente un exemple) :