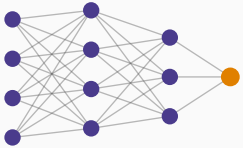
Apprentissage profond

Chapitre 2 : Le perceptron multicouche



Jaouad Madkour

Faculté d'Économie et de Gestion de Tanger · 27 août 2026

Plan du chapitre

L’architecture

Comment les couches s'enchaînent

Perceptron multicouche

Un **perceptron multicouche** est une suite de couches **entièrement connectées** : chaque neurone d'une couche reçoit toutes les sorties de la précédente.

L'écriture matricielle

Pour la couche ℓ :

$$z^{(\ell)} = W^{(\ell)} a^{(\ell-1)} + b^{(\ell)} \qquad a^{(\ell)} = g\big(z^{(\ell)}\big)$$

$W^{(\ell)}$ est une matrice, $b^{(\ell)}$ un vecteur, et g s'applique terme à terme.

C'est l'algèbre linéaire du niveau Licence

Le produit matrice-vecteur du chapitre 10 de *Mathématiques pour la gestion* est exactement l'opération élémentaire d'un réseau.

Un réseau profond n'est qu'une **composition** de ces produits, entrecoupée de non-linéarités. Sans elles, la composition de fonctions linéaires resterait linéaire — et toute la profondeur serait perdue.

Compter les paramètres

Un calcul à savoir faire

La formule par couche

Pour une couche de e entrées vers s sorties :

$$\text{paramètres} = \underbrace{e \times s}_{\text{poids}} + \underbrace{s}_{\text{biais}}$$

Un réseau pour notre série financière

Vingt indicateurs en entrée, deux couches cachées, une sortie :

Couche	Calcul	Paramètres
$20 \to 32$	$20 \times 32 + 32$	672
$32 \to 16$	$32 \times 16 + 16$	528
$16 \to 1$	$16 \times 1 + 1$	17
Total		$\mathbf{1\,217}$

Pourquoi ce comptage est une discipline

Intuition

Mille deux cent dix-sept paramètres pour vingt variables d'entrée. Une régression linéaire en aurait eu **vingt et un**.

Si l'on dispose de mille observations, il y a **plus de paramètres que de données**. Le chapitre 5 d'*Apprentissage statistique* l'a établi : dans ce régime, l'ajustement parfait est automatique et sans valeur.

Compter les paramètres avant d'entraîner est le premier réflexe de prudence. Un réseau surdimensionné pour les données disponibles ne peut que mémoriser.

Les ordres de grandeur usuels

Définition

Contexte	Paramètres
Réseau tabulaire modeste	quelques milliers
Réseau convolutif classique	quelques millions
Grand modèle de langue	des milliards

Chacun exige un volume de données à l'avenant.

Choisir une architecture

Profondeur et largeur

Deux leviers

La **largeur** est le nombre de neurones par couche. La **profondeur** est le nombre de couches.

À nombre de paramètres égal, la profondeur permet des représentations plus abstraites ; la largeur, davantage de descripteurs au même niveau.

Ce qui se pratique

Pour des données tabulaires, deux à quatre couches suffisent presque toujours. Aller au-delà n'améliore rien et complique l'entraînement.

Les architectures très profondes ne se justifient que pour l'image, le texte et le son, où la hiérarchie des représentations a un sens naturel.

La forme en entonnoir — des couches de plus en plus étroites, comme 32 puis 16 ci-dessus — est un point de départ raisonnable, non une règle.

Une architecture ne se devine pas

Le nombre de couches et leur largeur sont des **hyperparamètres**, au même titre que λ dans ridge ou α dans l'élagage.

La sortie

Adapter la dernière couche à la tâche

Trois cas

Tâche	Neurones de sortie	Activation finale
Régression	1	**aucune**
Classification binaire	1	**sigmoïde**
Classification à K classes	K	**softmax**

La fonction softmax

$$\text{softmax}(z)_k = \frac{e^{z_k}}{\sum_j e^{z_j}}$$

Elle transforme des scores quelconques en probabilités : toutes positives, de somme 1.

Trois régimes de marché

Scores en sortie : $2{,}0$; $1{,}0$; $0{,}1$.

Ce qu’il faut retenir

Le résumé du chapitre

Cinq points

1. Un perceptron multicouche enchaîne $z = Wa + b$ puis $a = g(z)$.
2. Une couche de e vers s coûte $e \times s + s$ paramètres.
3. Notre réseau 20–32–16–1 en compte $1\,217$: bien plus qu'une régression, à comparer au nombre d'observations.
4. Deux à quatre couches suffisent sur données tabulaires ; l'architecture est un **hyperparamètre**.
5. Sortie **libre** en régression, **sigmoïde** en binaire, **softmax** à K classes.

La suite

L'architecture est posée. Le chapitre 3 examine les deux choix qui la font fonctionner ou échouer : la fonction de perte et les activations.