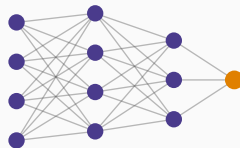


Apprentissage profond

Chapitre 8 : Les réseaux convolutifs



Jaouad Madkour

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

Pourquoi le perceptron ne suffit pas

La convolution

L'architecture typique

Au-delà de l'image

Ce qu'il faut retenir

Pourquoi le perceptron ne suffit
pas

Une image de 100 sur 100

Elle compte 10 000 pixels. Une couche entièrement connectée vers 100 neurones exigerait

$$10\,000 \times 100 + 100 = 1\,000\,100 \text{ paramètres}$$

pour une **seule** couche, sur une image minuscule.

La même chose par convolution

Trente-deux filtres de taille 3×3 :

$$(3 \times 3 \times 1 + 1) \times 32 = 320 \text{ paramètres}$$

Trois mille cent vingt-cinq fois moins, pour une capacité de détection supérieure.

Intuition

Le perceptron traite chaque pixel comme une variable indépendante. Il ignore que deux pixels voisins sont liés, et qu'un motif reste le même motif où qu'il se trouve.

La convolution **inscrit ces deux faits dans l'architecture** : c'est un a priori, au sens du chapitre 2 d'*Apprentissage statistique*. Il ajoute du biais et retranche énormément de variance — le compromis, encore.

La convolution

Le filtre

Un **filtre** est une petite matrice de poids que l'on fait glisser sur l'entrée. En chaque position, on calcule un produit scalaire local.

Le résultat est une **carte d'activation** : elle indique où le motif recherché est présent.

Les trois propriétés

1. **Connectivité locale** — chaque neurone ne voit qu'une petite région.
2. **Partage des poids** — le même filtre s'applique partout, d'où l'économie de paramètres.
3. **Équivariance par translation** — déplacer le motif déplace la réponse, sans la changer.

Définition	
Paramètre	Rôle
Taille du filtre	étendue du motif détecté — 3×3 le plus souvent
Pas	déplacement entre deux positions
Remplissage	bordure ajoutée pour conserver la taille
Nombre de filtres	nombre de motifs différents cherchés

Pooling

Le **pooling** réduit la résolution en ne conservant, sur chaque petite région, que le maximum ou la moyenne.

Deux effets

Il **réduit le calcul**, et il rend la représentation **invariante** à de petits déplacements : peu importe la position exacte du motif dans la région.

L'équivariance de la convolution devient ainsi une invariance — ce qui est exactement ce qu'on veut pour reconnaître un objet où qu'il soit.

L'architecture typique

Un empilement qui monte en abstraction

Le schéma classique

entrée

- [convolution → activation → pooling] × plusieurs
- aplatissement
- couches entièrement connectées
- sortie

La résolution **diminue**, le nombre de filtres **augmente** : on perd en précision spatiale et on gagne en richesse sémantique.

Ce que les couches apprennent

Profondeur	Détecte
Premières couches	contours, coins, variations locales
Couches intermédiaires	textures, motifs composés
Couches profondes	parties d'objets, objets

Au-delà de l'image

La convolution sur des séries

Convolution unidimensionnelle

Le même mécanisme s'applique à une série temporelle : le filtre glisse **le long du temps** et détecte des motifs locaux — une configuration de variations, un décrochage.

Le lien avec la modélisation ARMA

Un filtre convolutif de taille q appliqué à une série calcule une **combinaison linéaire pondérée des q dernières valeurs**.

C'est, structurellement, ce que fait un processus de moyenne mobile $MA(q)$ du chapitre 6 de *Modélisation ARMA*. La convolution en est une version dont les poids sont **appris**, et à laquelle on applique ensuite une non-linéarité.

Ce n'est pas une analogie décorative : c'est la même opération.

La convolution causale

Sur une série, un filtre centré utiliserait des valeurs **futurs** — inacceptable en prévision.

On emploie donc des convolutions **causales**, qui ne regardent que le passé. C'est la même précaution que le découpage temporel du chapitre 3 d'*Apprentissage statistique*, inscrite cette fois dans l'architecture

Ce qu'il faut retenir

Cinq points

1. Une couche dense sur une image 100×100 coûte 1 000 100 paramètres; une convolution 3×3 à 32 filtres en coûte 320.
2. Trois idées : connectivité **locale**, **partage** des poids, **équivariance** par translation.
3. Le **pooling** réduit la résolution et apporte l'invariance aux petits déplacements.
4. Les **connexions résiduelles** donnent au gradient un chemin direct et permettent la très grande profondeur.
5. En dimension une, un filtre convolutif est une **moyenne mobile apprise**; elle doit être **causale** en prévision.

La suite

La convolution capte des motifs **locaux**. Pour une dépendance longue, il faut une mémoire — c'est le chapitre 9.