

Introduction à l'Intelligence Artificielle (L2 Portail Sciences et Technologies)

Andrea G. B. Tettamanzi
Laboratoire I3S – Équipe SPARKS
`andrea.tettamanzi@univ-cotedazur.fr`



univ-cotedazur.fr

Séance 9

Réseaux de neurones artificiels

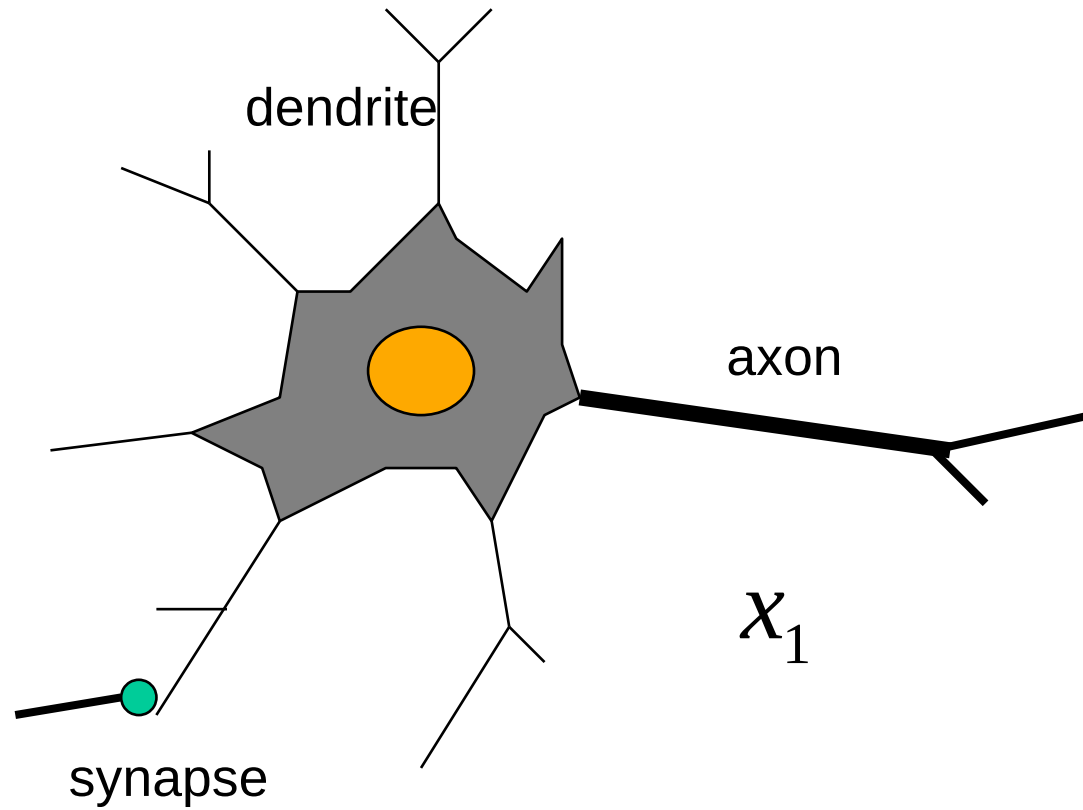
Plan pour cette séance

- Réseaux de neurones
- Rétropropagation
- Réseaux de Kohonen
- Apprentissage profond

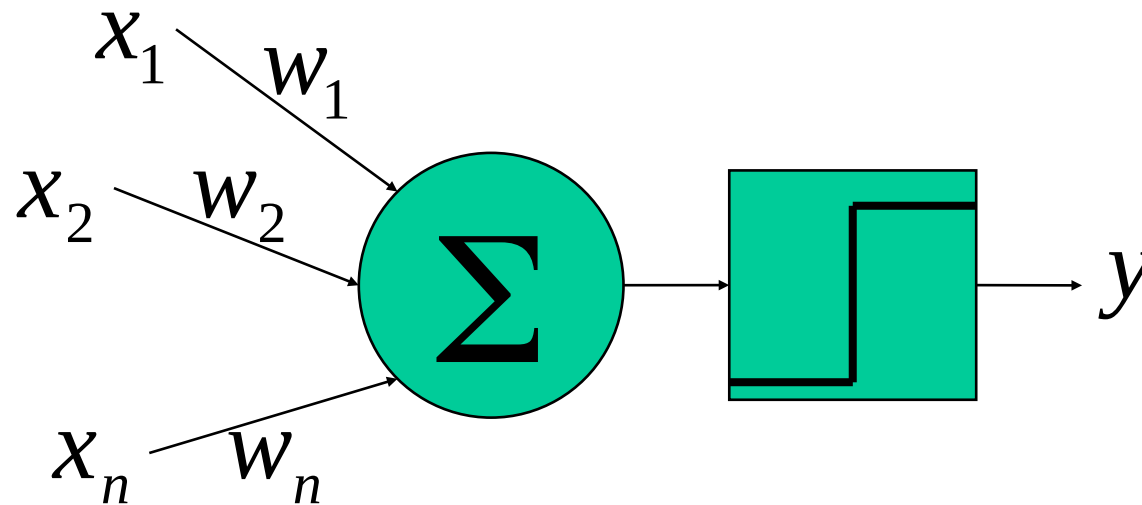
Réseaux de neurones artificiels

- Des modèles de cerveau très simplifiés.
- Combinaison de neurones et de connexions synaptiques, capables de faire passer des données à travers plusieurs couches.
- Modèles de reconnaissance et de classification des patrons.
- Amélioration des performances par « apprentissage ».
- Système adaptatif qui s'auto-organise afin de se rapprocher de la solution.

Neurone



Neurone artificiel



Fonctions d'activation

Sigmoïdale

Logistique

$$f(x) = \frac{1}{1 + e^{-x}}$$

Tangente hyperbolique

$$f(x) = \tanh(x)$$

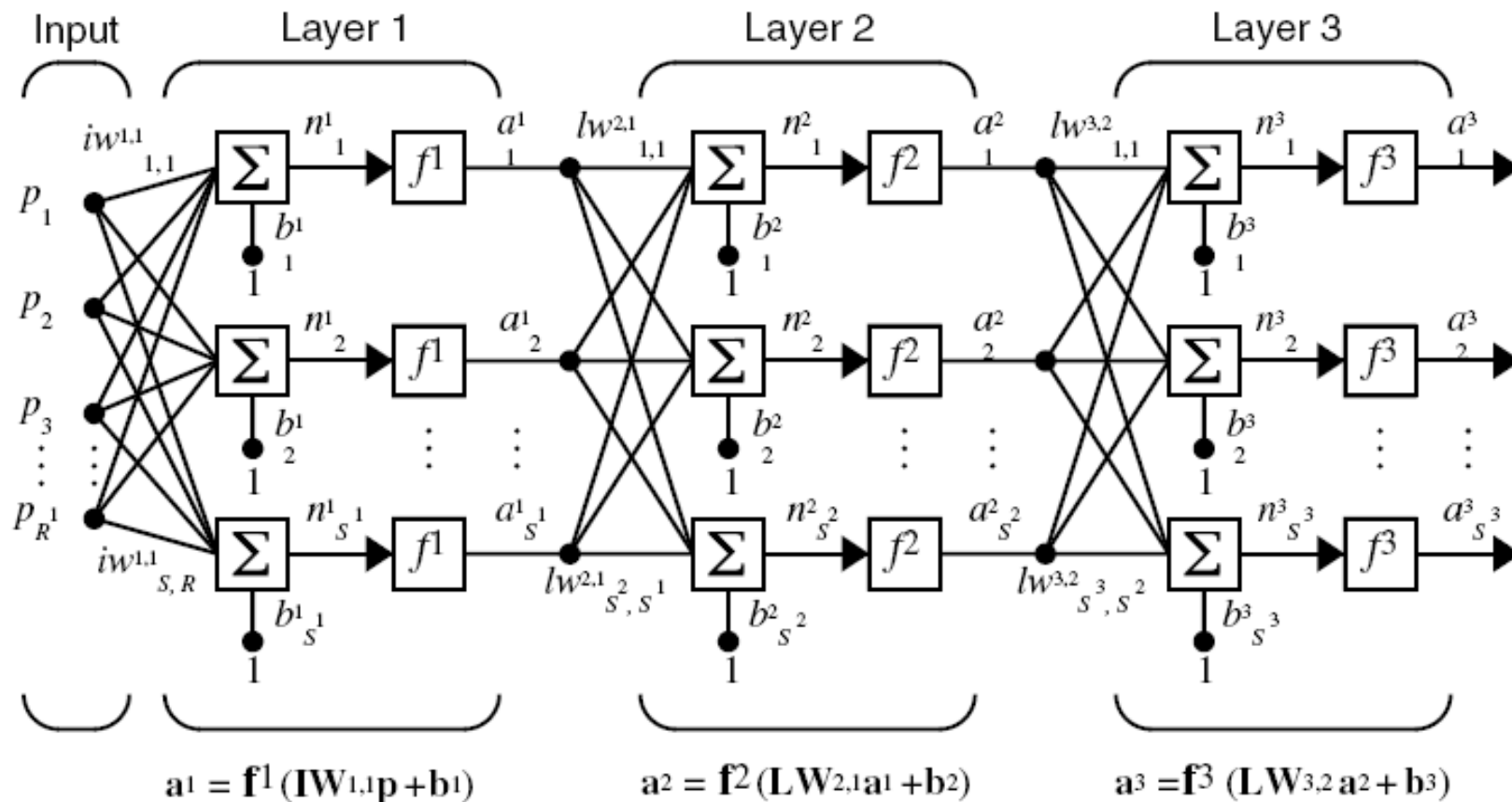
Rectificateur

$$f(x) = \max\{0, x\}$$

SoftMax

$$f(\mathbf{x})_i = \frac{e^{x_i}}{\sum_j e^{x_j}}$$

RNs multicouches acycliques



$$a^3 = f^3(LW_{3,2} f^2(LW_{2,1} f^1(IW_{1,1}p + b^1) + b^2) + b^3)$$

Rétropropagation de l'erreur

$$L(\mathbf{y}, f(\mathbf{x})) = \|\mathbf{y} - f(\mathbf{x})\|^2 \quad \text{Erreur}$$

$$\mathbf{a}^i = f^i(\mathbf{W}^i \cdot \mathbf{a}^{i-1}) \quad \text{Propagation de l'activation}$$

$$L(\mathbf{y}, f^N(\mathbf{W}^N f^{N-1}(\mathbf{W}^{N-1} \dots f^2(\mathbf{W}^2 f^1(\mathbf{W}^1 \mathbf{x}))))$$

$$\nabla_{\mathbf{W}^i} L = \delta^i (\mathbf{a}^{i-1})^T \quad \text{Gradient des poids de la } i\text{-ème couche}$$

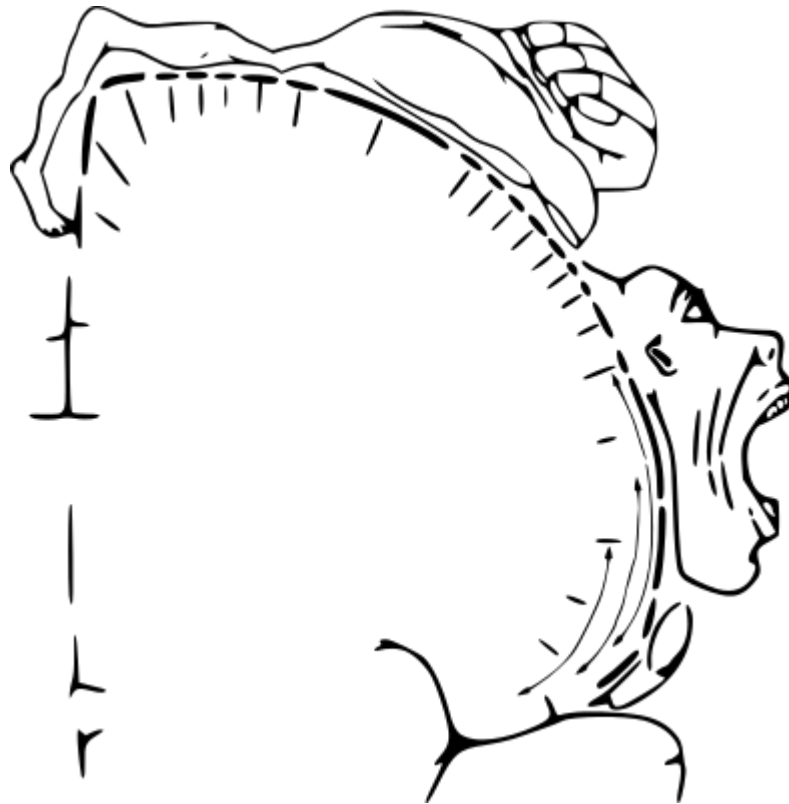
$$\delta^{i-1} = (f^{i-1})' \cdot (\mathbf{W}^i)^T \cdot \delta^i \quad \text{Rétropropagation du gradient}$$

$$\Delta \mathbf{W}^i = -\eta \nabla_{\mathbf{W}^i} L \quad \text{Mise à jour des poids}$$

Réseaux de Kohonen

- Les cartes auto-organisées (*self-organizing maps*, SOM) sont des méthodes d'analyse automatique des données basées sur une architecture de réseau neuronal, inventée par Teuvo Kohonen (1980).
- Les SOMs sont liées à la quantification vectorielle (QV) classique, utilisée dans le traitement et la transmission de signaux numériques.
- Comme dans la QV, les SOMs représentent une distribution d'éléments de données d'entrée utilisant un ensemble fini de modèles.
- Dans les SOMs, cependant, ces modèles sont automatiquement associés aux nœuds d'une grille régulière (généralement bidimensionnelle) de manière ordonnée : la carte de similarité.
- Cette organisation permet d'obtenir un aperçu des relations topographiques des données, en particulier des éléments de données à haute dimension

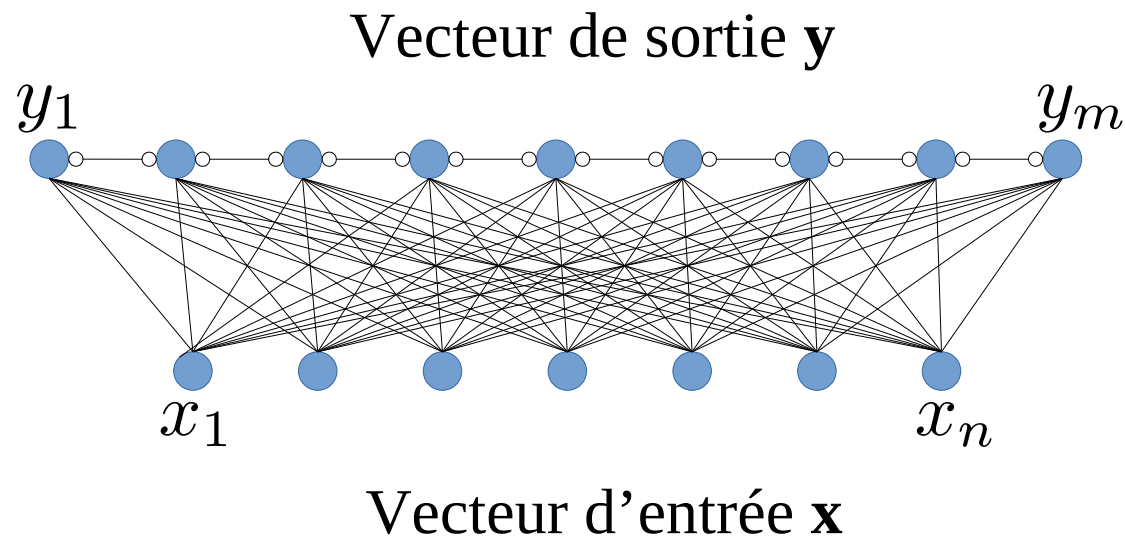
SOMs: Motivation biologique



L'homuncule cortical sensorimoteur 2D - un exemple célèbre de carte du cerveau

Architecture d'une SOM

- Deux couches : couche d'entrée et couche de sortie (carte)
- Les couches d'entrée et sortie sont complètement connectées
- Les neurones de sortie sont interconnectés dans un voisinage défini
- Une topologie (fonction de voisinage h) est définie sur la couche de sortie, le plus souvent 1- ou 2-dimensionnelle.

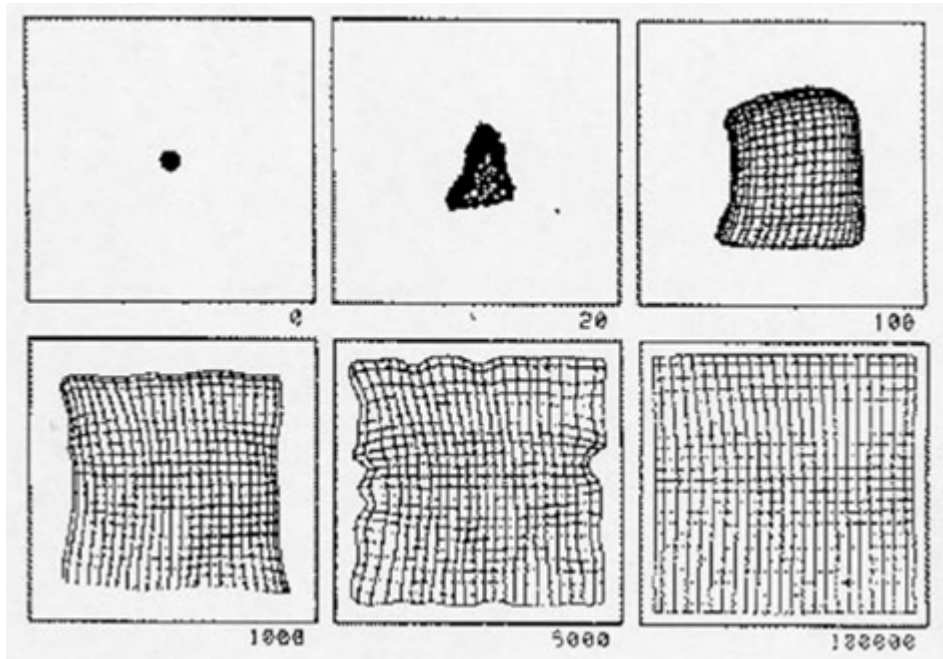


Algo d'apprentissage des SOM

- 1) Initialisation aléatoire de tous les poids
- 2) On choisit un vecteur d'entrée $\mathbf{x} = (x_1, \dots, x_n)$
- 3) On détermine le neurone de sortie $k = \arg \min_i \|\mathbf{w}_i - \mathbf{x}\|$
- 4) Mise à jour de tous les poids pour tous les neurones dans le voisinage du neurone k :
$$\mathbf{w}_i \leftarrow \mathbf{w}_i + \delta \cdot h(i, k) \cdot (\mathbf{x} - \mathbf{w}_i)$$

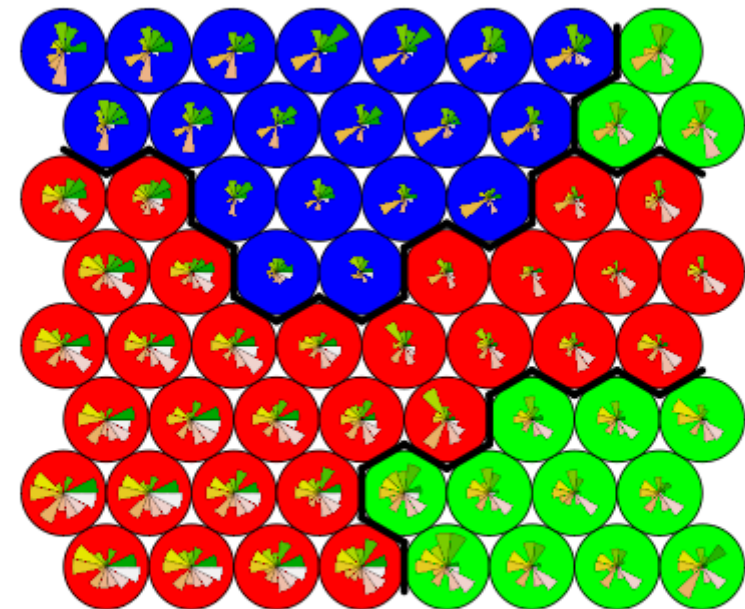
(c-à-d, \mathbf{w}_i glisse vers \mathbf{x})
Fonction de voisinage typique : $h(i, j) = e^{-\lambda d(i, j)^2}$
- 5) S'il n'y a pas de convergence, on restreint la fn de voisinage (en augmentant λ), on diminue le pas δ et on retourne à 2.

Exemples de SOMs



Apprentissage d'une représentation 2D
d'un espace d'entrée (carré) 2D

Codes plot

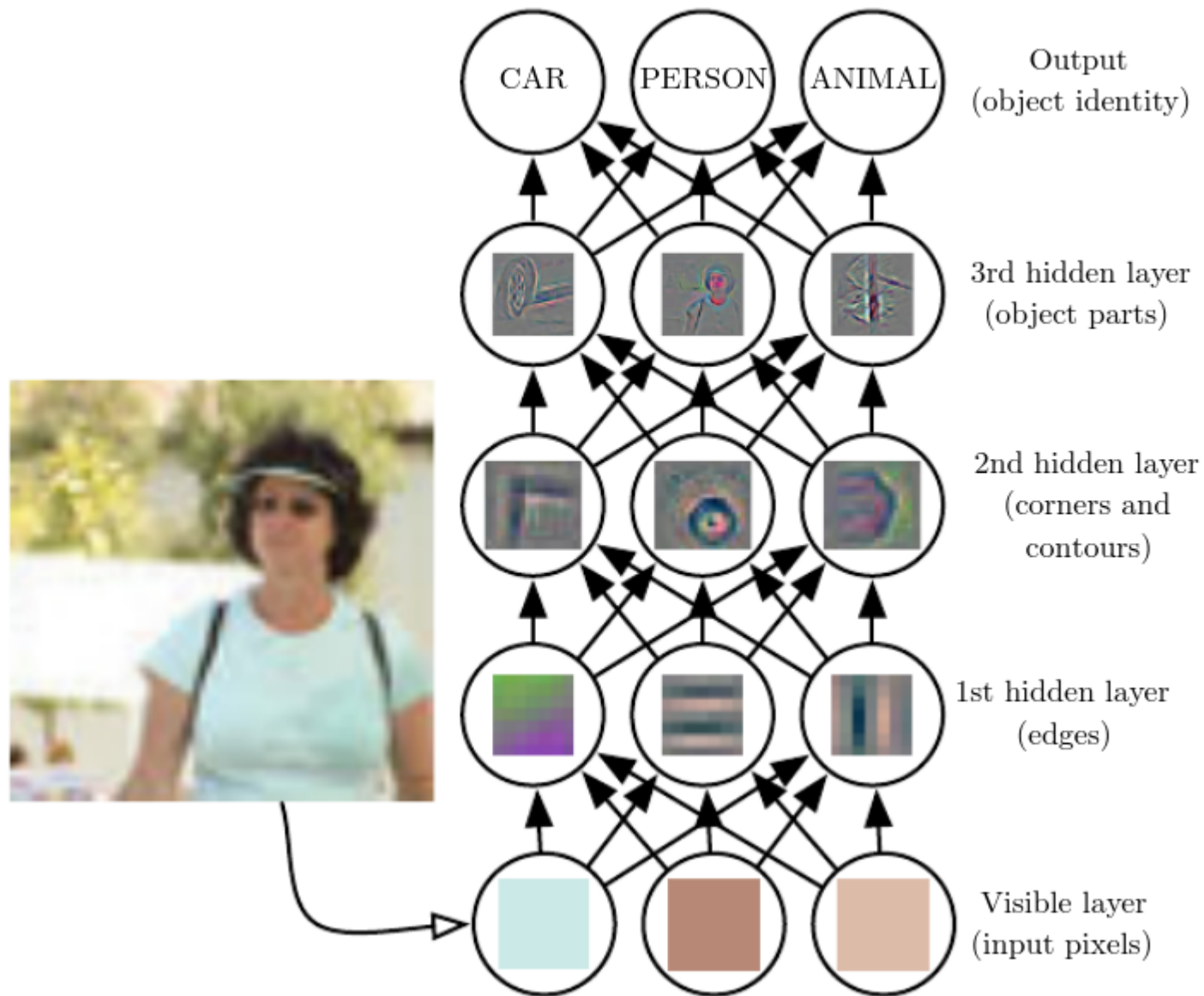


| | | |
|------------------|---------------------|------------|
| ■ alcohol | ■ tot. phenols | ■ col. hue |
| ■ malic acid | ■ flavonoids | ■ OD ratio |
| ■ ash | ■ non-flav. phenols | ■ proline |
| ■ ash alkalinity | ■ proanth | |
| ■ magnesium | ■ col. int. | |

Clustering d'un jeu de données de vins

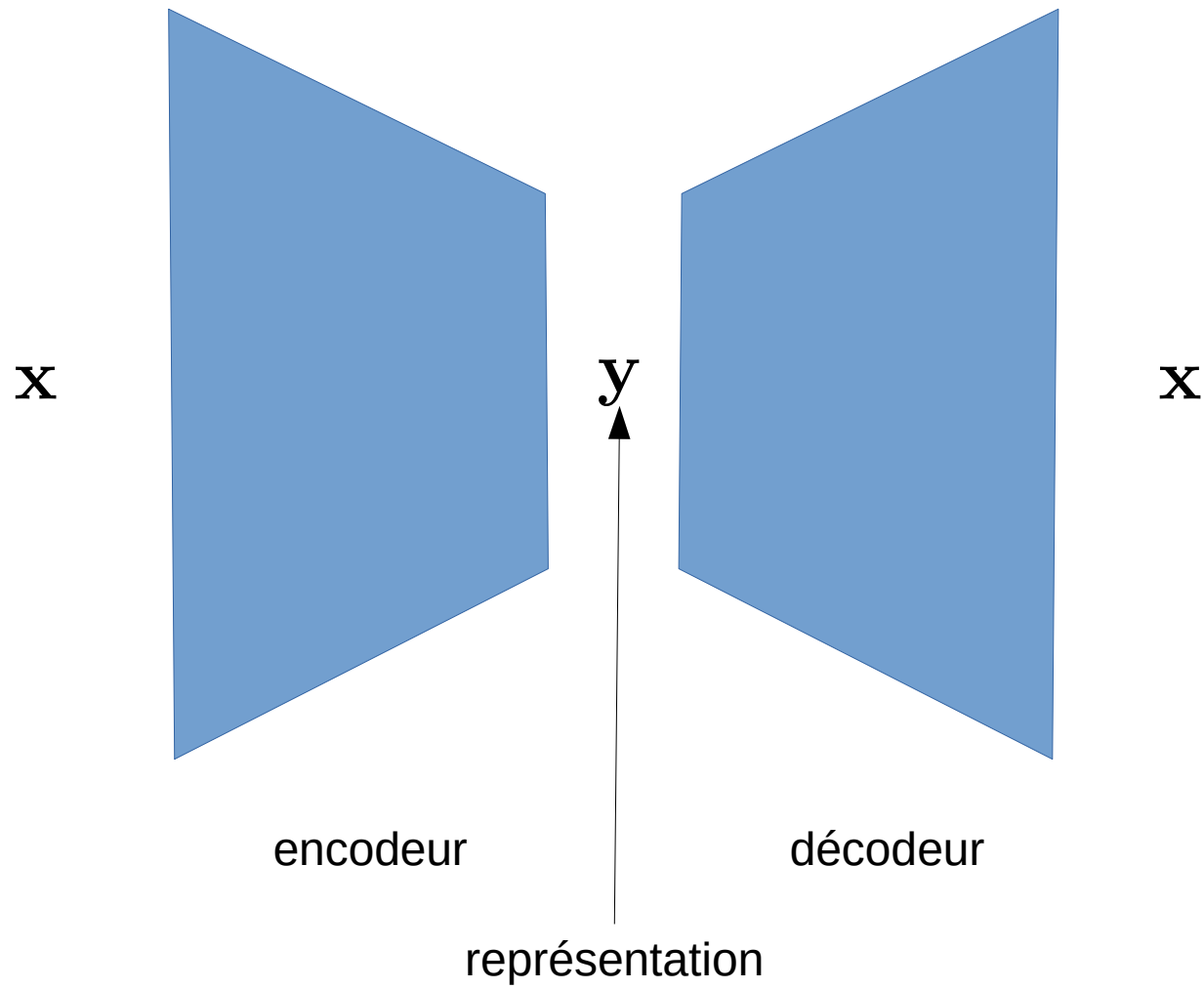
Apprentissage profond

- La difficulté d'un problème d'apprentissage dépend de la représentation des entrées
- L'idée de fond est celle d'apprendre des représentations
- La quintessence de l'apprentissage de représentation est l'auto-encodeur
- Représentations basées sur des représentations plus simples
- On utilise des réseaux de neurones structurées
- Extension de l'idée des Rns multicouches acycliques

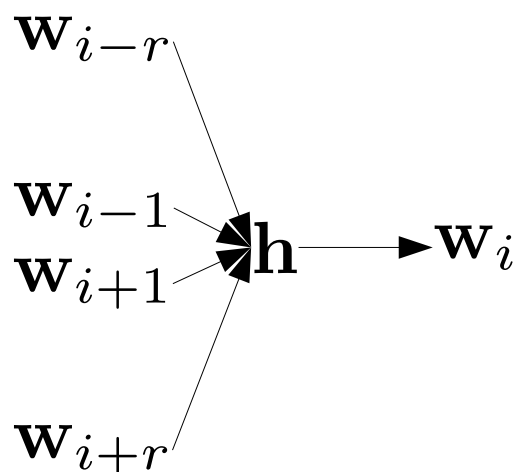
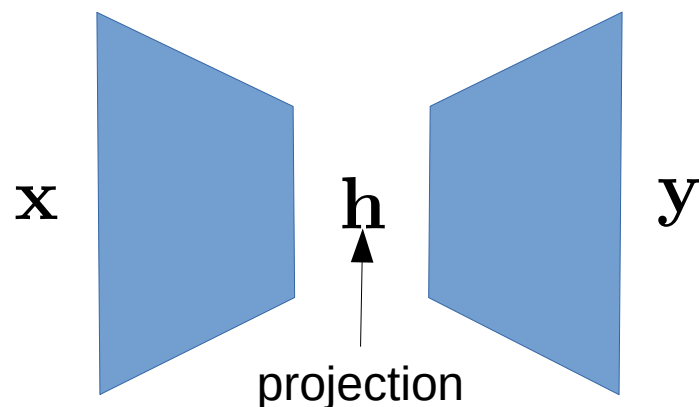


[Ian Goodfellow and Yoshua Bengio and Aaron Courville. *Deep Learning*, MIT Press, 2016]

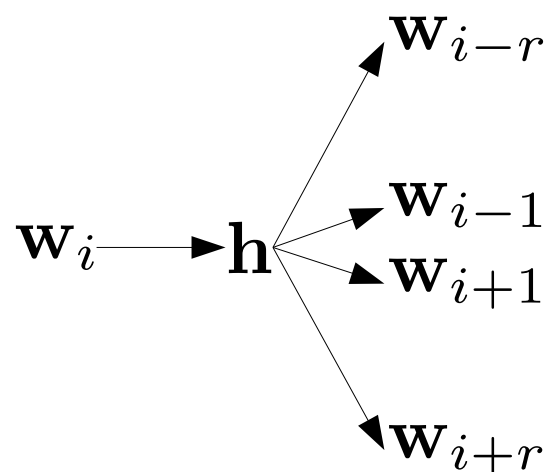
Auto-encodeur



Plongement lexical : word2vec



CBOW



skipgram

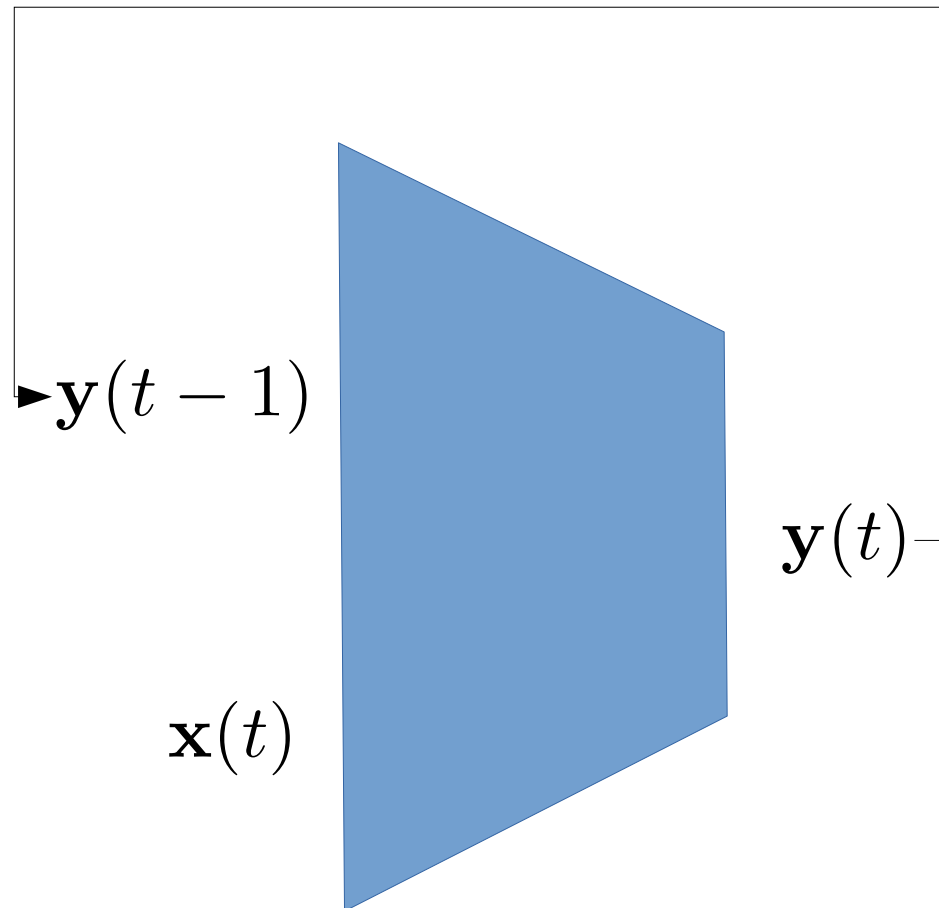
RNs convolutionnels

- Les réseaux convolutionnels sont simplement des réseaux de neurones qui utilisent la convolution au lieu de la multiplication matricielle générale dans au moins une de leurs couches
- Très adaptés pour traiter des données qui ont une topologie à grille
 - Séries historiques (grille 1D)
 - Images (grille 2D)

RNs récurrents

- Particulièrement adaptés pour traiter des données séquentielles
 - Signaux audio
 - Vidéo (images en mouvement)
 - Textes (y inclus parlé)
- La sortie du réseau (pour l'entrée de la période t) est réinjectée dans le réseaux avec l'entrée de la période $t + 1$

RN récurrent



Déroulement d'un RN récurrent

