



Introduction au Machine Learning

Massih-Reza Amini, Alexandre Audibert, Aurélien Gauffre

Université Grenoble Alpes
Laboratoire d'Informatique de Grenoble
`Prenom.Nom@univ-grenoble-alpes.fr`



Programme et Équipe

- ❑ Modèles d'apprentissage pour la classification
 - ❑ Perceptron,
 - ❑ Regression Logistique,
 - ❑ Séparateurs à Vaste Marge (SVM),
 - ❑ Perceptron Multi-Couches (PMC)

- ❑ Équipe pédagogique:
Massih-Reza Amini, Alexandre Audibert, Aurélien Gauffre

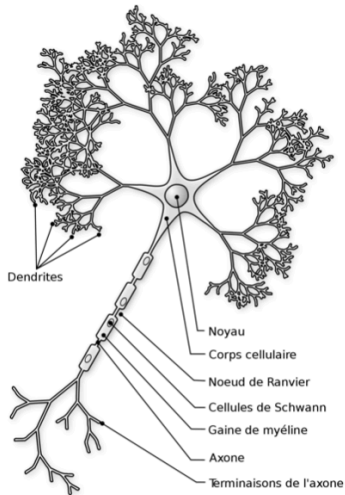
- ❑ Notes: Contrôles continus (TPs notés - 30%) et Examen final (70%)

Premières tentatives pour construire un neurone artificiel

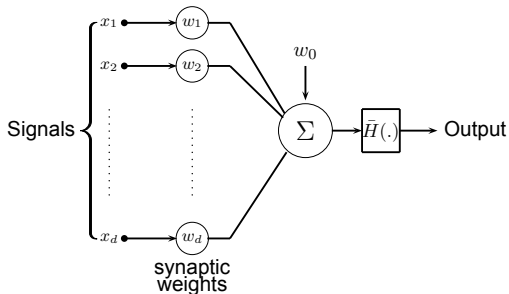
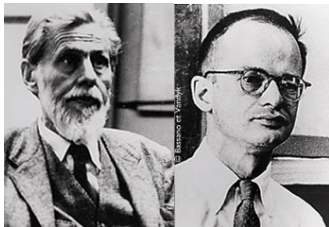
Tout a débuté à la fin du 19eme siècle avec les travaux de Santiago Ramón y Cajal qui a représenté le neurone naturel.



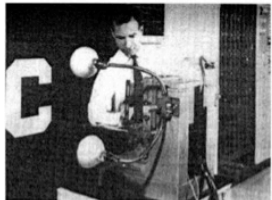
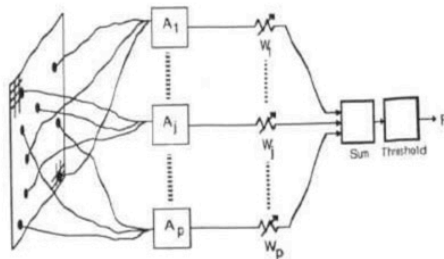
Qu'est ce qu'un neurone?



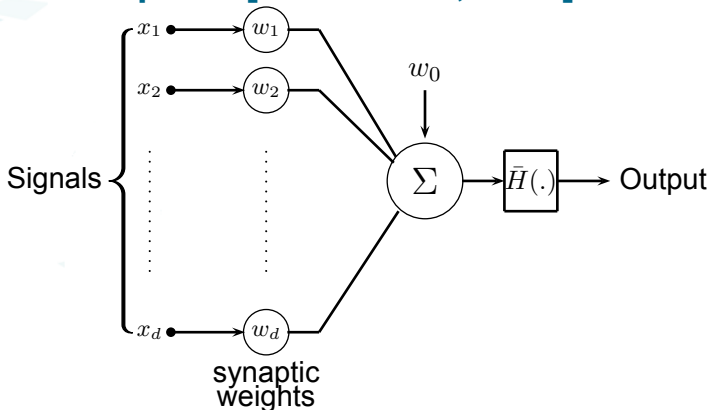
Neurone Formel [McCulloch and Pitts, 1943]



Perceptron [Rosenblatt, 1958]



Perceptron [Rosenblatt, 1958]



- Fonction de prédiction linéaire

$$h_{\mathbf{w}} : \mathbb{R}^d \rightarrow \mathbb{R}$$

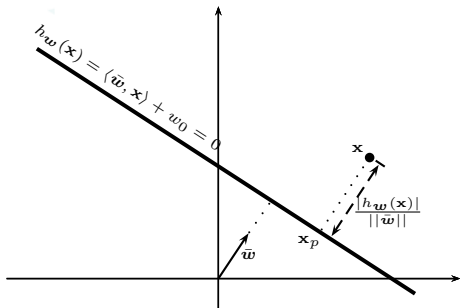
$$\mathbf{x} \mapsto \langle \bar{\mathbf{w}}, \mathbf{x} \rangle + w_0$$

Perceptron [Rosenblatt, 1958]

- Fonction de prédiction linéaire

$$h_{\mathbf{w}} : \mathbb{R}^d \rightarrow \mathbb{R}$$
$$\mathbf{x} \mapsto \langle \bar{\mathbf{w}}, \mathbf{x} \rangle + w_0$$

- Trouve les paramètres $\mathbf{w} = (\bar{\mathbf{w}}, w_0)$ en minimisant la distance entre les exemples mal-classés et la frontière de décision.



Apprendre les paramètres du modèle

- Si un exemple (\mathbf{x}, y) est mal classé

$$\ell(\mathbf{w}, x, y) = -y \times h_{\mathbf{w}}(\mathbf{x})$$

- Les dérivées de $\ell(\mathbf{w}, x, y)$ par rapport aux paramètres

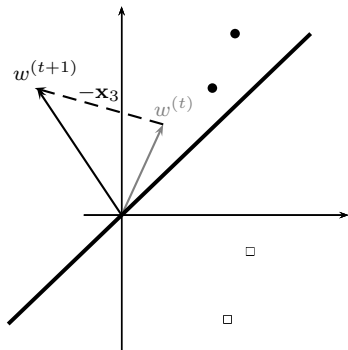
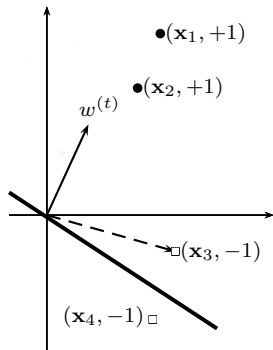
$$\frac{\partial \ell(\mathbf{w}, x, y)}{\partial w_0} = -y,$$

$$\nabla \ell(\mathbf{w}, x, y) = -y\mathbf{x}$$

- Perceptron: Mise à jour en-ligne des paramètres

$$\forall (\mathbf{x}, y), \text{ si } y(\langle \bar{\mathbf{w}}, \mathbf{x} \rangle + w_0) \leq 0 \text{ alors } \begin{pmatrix} w_0 \\ \bar{\mathbf{w}} \end{pmatrix} \leftarrow \begin{pmatrix} w_0 \\ \bar{\mathbf{w}} \end{pmatrix} + \eta \begin{pmatrix} y \\ y\mathbf{x} \end{pmatrix}$$

Mise à jour en-ligne des paramètres



Perceptron (algorithme)

Algorithm 1 L'algorithme de perceptron

- 1: Base d'apprentissage $S = \{(\mathbf{x}_i, y_i) \mid i \in \{1, \dots, m\}\}$
 - 2: Initialiser les poids $w^{(0)} \leftarrow 0$
 - 3: $t \leftarrow 0$
 - 4: Pas d'apprentissage $\eta > 0$
 - 5: **repeat**
 - 6: Choisir aléatoire un exemple $(\mathbf{x}^{(t)}, y^{(t)}) \in S$
 - 7: **if** $y \langle w^{(t)}, \mathbf{x}^{(t)} \rangle < 0$ **then**
 - 8: $w_0^{(t+1)} \leftarrow w_0^{(t)} + \eta \times y^{(t)}$
 - 9: $w^{(t+1)} \leftarrow w^{(t)} + \eta \times y^{(t)} \times \mathbf{x}^{(t)}$
 - 10: **end if**
 - 11: $t \leftarrow t + 1$
 - 12: **until** $t > T$
-

☞ Est-ce que cet algorithme converge?

References



W. Mcculloch, and W. Pitts

A Logical Calculus of Ideas Immanent in Nervous Activity

Bulletin of Mathematical Biophysics

1943



F. Rosenblatt

The perceptron: A probabilistic model for information storage and organization in the brain.

Psychological Review, 65: 386–408.

1958