

Une introduction à l'apprentissage automatique

Ralph Sarkis

SUMM - 2018

14 janvier 2018

Introduction

De quoi parle-t-on ?

Introduction

De quoi parle-t-on ?

- ▶ Champ d'étude de l'intelligence artificielle.
- ▶ Algorithmes qui s'adaptent en fonction des données
- ▶ Permet de résoudre des problèmes très complexes

Introduction

De quoi parle-t-on ?

- ▶ Champ d'étude de l'intelligence artificielle.
- ▶ Algorithmes qui s'adaptent en fonction des données
- ▶ Permet de résoudre des problèmes très complexes

Exemples

Reconnaissance d'images, prédiction sur la bourse ou des événements sportifs, jeux de stratégie, etc...

Plan de la présentation

Plan de la présentation

- ▶ Jeu de Nim

Définition du jeu, explication et test de l'algorithme.

Plan de la présentation

- ▶ Jeu de Nim

Définition du jeu, explication et test de l'algorithme.

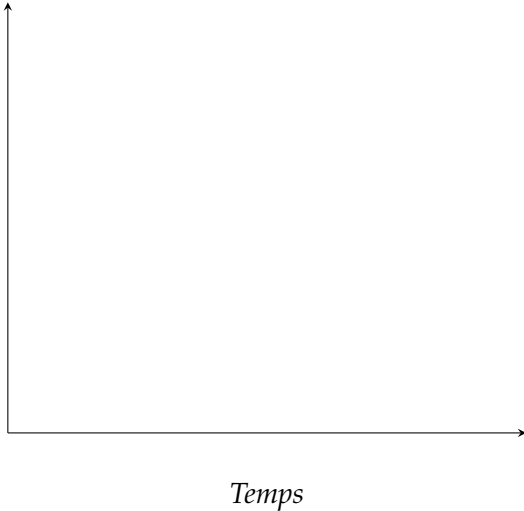
- ▶ Classifieurs linéaires

Étude complète d'un exemple et généralisation.

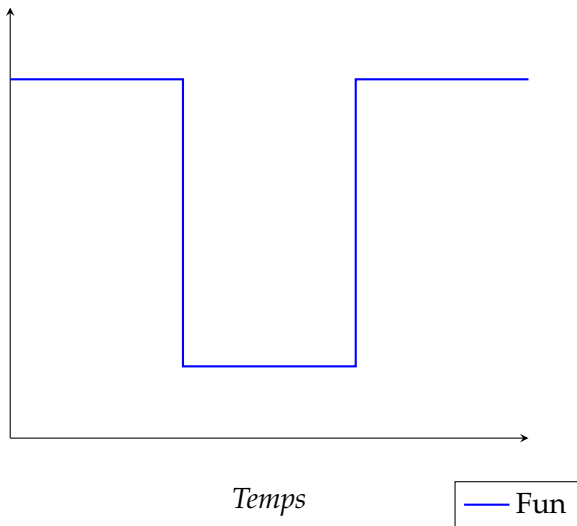
Plan de la présentation

- ▶ Jeu de Nim
Définition du jeu, explication et test de l'algorithme.
- ▶ Classifieurs linéaires
Étude complète d'un exemple et généralisation.
- ▶ Réseaux de neurones génératifs (GANs)
Description du modèle et illustration de son efficacité.

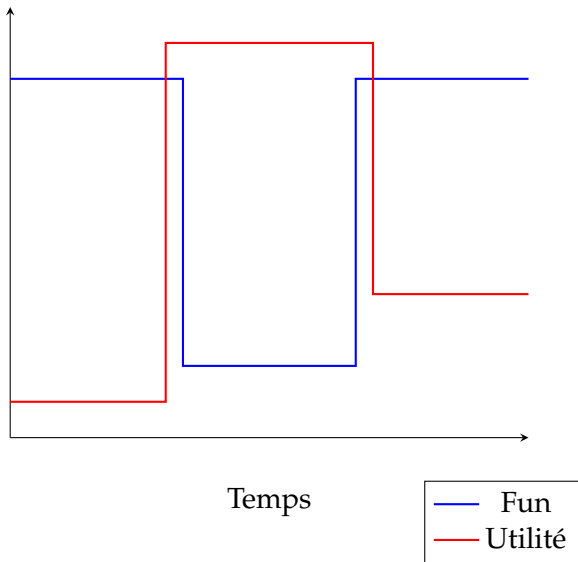
Plan de la présentation



Plan de la présentation



Plan de la présentation



Introduction

Jeu de Nim

Classifieurs linéaires

GANs

Les règles du jeu de Nim

- ▶ 2 joueurs s'affrontent.
- ▶ Il y a 21 bâtons.
- ▶ À son tour, un joueur choisit de retirer 1, 2 ou 3 bâtons.
- ▶ Le joueur qui retire le dernier bâton perd la partie.

C'est l'heure du duel!



Il reste 21 bâtons.

C'est l'heure du duel!



Il reste 20 bâtons.

C'est l'heure du duel!



Il reste 19 bâtons.

C'est l'heure du duel!



Il reste 18 bâtons.

C'est l'heure du duel!



Il reste 17 bâtons.

C'est l'heure du duel!



Il reste 16 bâtons.

C'est l'heure du duel!



Il reste 15 bâtons.

C'est l'heure du duel!



Il reste 14 bâtons.

C'est l'heure du duel!



Il reste 13 bâtons.

C'est l'heure du duel!



Il reste 12 bâtons.

C'est l'heure du duel!



Il reste 11 bâtons.

C'est l'heure du duel!



Il reste 10 bâtons.

C'est l'heure du duel!



Il reste 9 bâtons.

C'est l'heure du duel!



Il reste 8 bâtons.

C'est l'heure du duel!



Il reste 7 bâtons.

C'est l'heure du duel!



Il reste 6 bâtons.

C'est l'heure du duel!



Il reste 5 bâtons.

C'est l'heure du duel!



Il reste 4 bâtons.

C'est l'heure du duel!



Il reste 3 bâtons.

C'est l'heure du duel!



Il reste 2 bâtons.

J'ai gagné!



Il reste 1 bâton.

La stratégie de l'IA

Résultats

# boîte	# de billes dans la boîte		
	Blanc (1)	Vert (2)	Bleu (3)
14	13	17	27
13	36	7	11
12	8	23	2
11	5	43	5
10	49	3	0
9	0	2	27
8	1	12	59
7	5	38	0
6	42	0	2
5	0	0	0
4	1	0	108
3	2	70	1
2	42	0	1

Conclusion

Avantages

Conclusion

Avantages

- ▶ Très simple à implémenter.

Conclusion

Avantages

- ▶ Très simple à implémenter.
- ▶ Bonne intuition sur la stratégie et les résultats.

Conclusion

Avantages

- ▶ Très simple à implémenter.
- ▶ Bonne intuition sur la stratégie et les résultats.

Inconvénients

Conclusion

Avantages

- ▶ Très simple à implémenter.
- ▶ Bonne intuition sur la stratégie et les résultats.

Inconvénients

- ▶ L'algorithme ne converge pas tout le temps vers une stratégie optimale.

Conclusion

Avantages

- ▶ Très simple à implémenter.
- ▶ Bonne intuition sur la stratégie et les résultats.

Inconvénients

- ▶ L'algorithme ne converge pas tout le temps vers une stratégie optimale.
- ▶ Elle demande beaucoup de ressources.

Introduction

Jeu de Nim

Classifieurs linéaires

GANs

Le problème

Le problème

Formulation du problème

En analysant les propriétés quantitatives d'objets divers, comment répartir ses objets dans des groupes avec des propriétés similaires.

Le problème

Formulation du problème

En analysant les propriétés quantitatives d'objets divers, comment répartir ses objets dans des groupes avec des propriétés similaires.

Et concrètement ?

Une banque de données contient plusieurs milliers d'images et nous voulons les trier en deux catégories : les images de chats et les autres images.

La solution

L'apprentissage supervisé, une technique d'apprentissage automatique.

La solution

L'apprentissage supervisé, une technique d'apprentissage automatique.

En bref

L'algorithme commence avec une liste d'exemple d'objets déjà classés, et il va l'utiliser pour apprendre à classer des nouveaux objets.

L'implémentation

L'implémentation

Un réseau artificiel de neurones

L'implémentation

Un réseau artificiel de neurones

- ▶ Inspiré du fonctionnement du cerveau.

L'implémentation

Un réseau artificiel de neurones

- ▶ Inspiré du fonctionnement du cerveau.
- ▶ Technique très populaire car démontrée très efficace.

L'implémentation

Un réseau artificiel de neurones

- ▶ Inspiré du fonctionnement du cerveau.
- ▶ Technique très populaire car démontrée très efficace.
- ▶ Fonctionnement relativement simple.

L'implémentation

Un réseau artificiel de neurones

- ▶ Inspiré du fonctionnement du cerveau.
- ▶ Technique très populaire car démontrée très efficace.
- ▶ Fonctionnement relativement simple.

Allons voir comment ça marche !

Réseau de neurones

On commence par les observations x_1 , x_2 et x_3 .



Réseau de neurones

Ensuite, les résultats y_1 and y_2 .

$$x_1$$

$$x_2$$

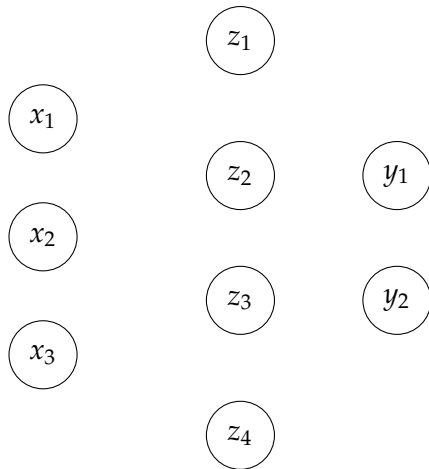
$$x_3$$

$$y_1$$

$$y_2$$

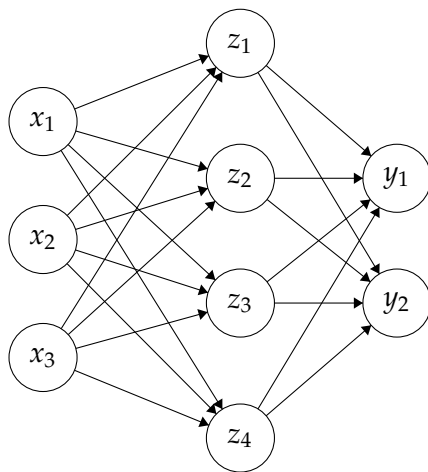
Réseau de neurones

La couche cachée avec les neurones z_1, z_2, z_3 and z_4 .



Réseau de neurones

Finalement, les liens entre les neurones.



Réseau de neurones

Comment calcule-t-on les activations dans les autres couches ?

Réseau de neurones

Comment calcule-t-on les activations dans les autres couches ?

Propagation de l'information

Chaque flèche dans le graphique représente un coefficient que l'on dénotera par w .

Réseau de neurones

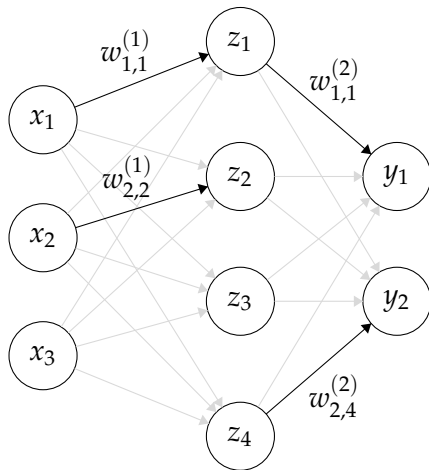
Comment calcule-t-on les activations dans les autres couches ?

Propagation de l'information

Chaque flèche dans le graphique représente un coefficient que l'on dénotera par w .

$$w_{i,j}^{(\ell)} \quad \begin{cases} \ell & \text{Numéro de la couche} \\ i & \text{Indice du neurone de la couche actuel} \\ j & \text{Indice du neurone de la couche précédente} \end{cases}$$

Exemples de coefficients



Calcul des activations

On simplifie les formules avec la notation vectorielle.

$$\begin{aligned} \vec{z} &= \sigma \left(W^{(1)} \cdot \vec{x} \right) \\ \vec{y} &= \sigma \left(W^{(2)} \cdot \vec{z} \right) \end{aligned} \quad \text{avec } W^{(\ell)} = (w^{(\ell)})_{i,j} \text{ et } \sigma(s) = \frac{1}{1+e^{-s}}$$

Calcul des activations

On simplifie les formules avec la notation vectorielle.

$$\begin{aligned} \vec{z} &= \sigma \left(W^{(1)} \cdot \vec{x} \right) \\ \vec{y} &= \sigma \left(W^{(2)} \cdot \vec{z} \right) \end{aligned} \quad \text{avec } W^{(\ell)} = (w^{(\ell)})_{i,j} \text{ et } \sigma(s) = \frac{1}{1+e^{-s}}$$

Exemple

$$z_1 = \sigma \left(w_1^{(1)} \cdot \vec{x} \right) = \sigma \left(w_{1,1}^{(1)} x_1 + w_{1,2}^{(1)} x_2 + w_{1,3}^{(1)} x_3 \right)$$

Entraînement du réseau

Au début, les coefficients sont arbitraires. Ensuite, le réseau classifie des observations déjà cataloguées et s'ajuste pour donner des meilleurs réponses.

Entraînement du réseau

Au début, les coefficients sont arbitraires. Ensuite, le réseau classifie des observations déjà cataloguées et s'ajuste pour donner des meilleures réponses.

Fonction objectif

$$J(W, \vec{x}) = \sum_{i=1}^2 \frac{1}{2} (y_i - y_i^*)^2$$

$$J(W) = \sum_{\vec{x}} J(W, \vec{x})$$

Entraînement du réseau

Au début, les coefficients sont arbitraires. Ensuite, le réseau classifie des observations déjà cataloguées et s'ajuste pour donner des meilleures réponses.

Fonction objectif

$$J(W, \vec{x}) = \sum_{i=1}^2 \frac{1}{2} (y_i - y_i^*)^2$$

$$J(W) = \sum_{\vec{x}} J(W, \vec{x})$$

Quand l'erreur J est petite, on se rapproche de la solution.

Algorithme du gradient

On cherche donc à minimiser l'erreur en utilisant une technique d'optimisation simple.

Algorithme du gradient

On cherche donc à minimiser l'erreur en utilisant une technique d'optimisation simple.

$$W \leftarrow W - \frac{\partial J}{\partial W}$$

Algorithme du gradient

On cherche donc à minimiser l'erreur en utilisant une technique d'optimisation simple.

$$W \leftarrow W - \frac{\partial J}{\partial W}$$

Rincer et répéter.

Conclusion

Avantages

Conclusion

Avantages

- ▶ Extrêmement efficace.

Conclusion

Avantages

- ▶ Extrêmement efficace.
- ▶ Applicable à de nombreux problèmes.

Conclusion

Avantages

- ▶ Extrêmement efficace.
- ▶ Applicable à de nombreux problèmes.

Inconvénients

Conclusion

Avantages

- ▶ Extrêmement efficace.
- ▶ Applicable à de nombreux problèmes.

Inconvénients

- ▶ Besoin de beaucoup de données.

Conclusion

Avantages

- ▶ Extrêmement efficace.
- ▶ Applicable à de nombreux problèmes.

Inconvénients

- ▶ Besoin de beaucoup de données.
- ▶ Aucune intuition sur le résultat.

Introduction

Jeu de Nim

Classifieurs linéaires

GANs

L'inverse des classifieurs

Les GANs (*Generative Adversarial Networks*) sont des réseaux génératifs.

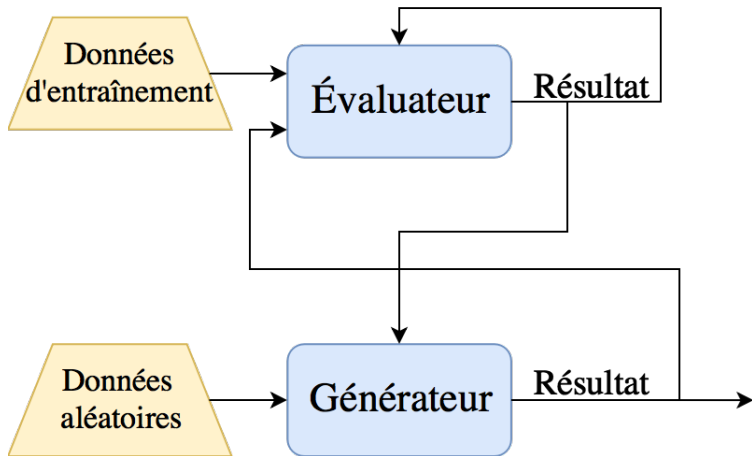
On veut, à partir de données aléatoires, générer une observation qui ressemble aux éléments d'une classe précise.

Exemple

Générer une image de chat, appliquer une texture réaliste à un modèle abstrait, etc.

Comment ça marche ?

Deux modèles se battent pour devenir meilleur.



Exemple 1 : des chats



2. Images générée par Alexia Jolicoeur-Martineau, plus d'infos sur <https://ajolicoeur.wordpress.com/cats/>

Exemple 2 : des célébrités

Merci bien !