

INF8215 - Intelligence artificielle

Méthodes et algorithmes

Module 7: Apprentissage supervisé



POLYTECHNIQUE
MONTRÉAL

Quentin Cappart

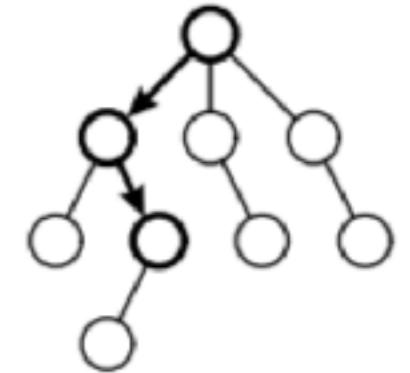
Contenu du cours

Raisonnement par recherche (essais-erreurs avec de l'intuition)

Module 1: Stratégies de recherche

Module 2: Recherche en présence d'adversaires

Module 3: Recherche locale



Raisonnement logique

Module 4: Programmation par contraintes

Module 5: Agents logiques

Module 6: Logique du premier ordre et inférence

SS SSSS Breeze		Breeze	PIT
WIND	Breeze	SOLID	Breeze
SS SSSS Solid		Breeze	
WIND	Breeze	PIT	Breeze

Raisonnement par apprentissage

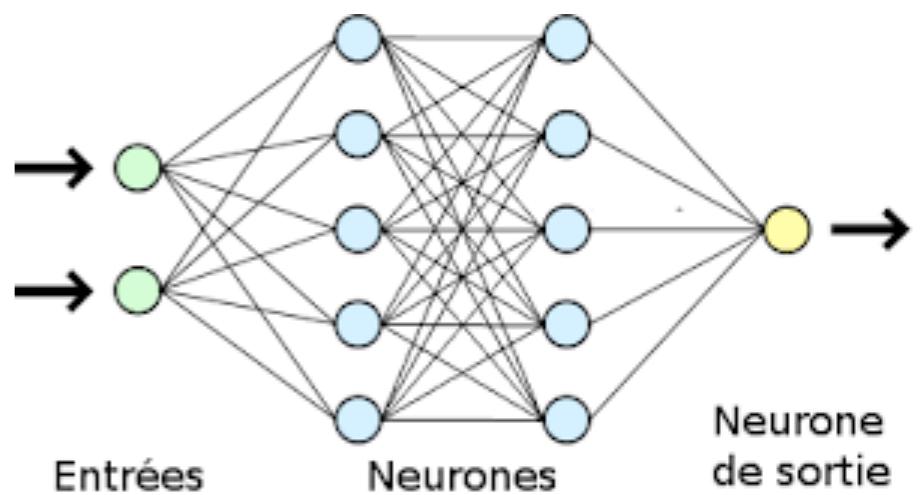
Module 7: Apprentissage supervisé



Module 8: Réseaux de neurones et apprentissage profond

Module 9: Apprentissage non-supervisé

Module 10: Apprentissage par renforcement



Applications industrielles

Présentation d'une entreprise utilisant des techniques d'IA

Table des matières

Apprentissage supervisé

1. Motivation et intérêt de l'apprentissage automatique
2. Classification des principaux types d'apprentissage
3. Définition de l'apprentissage supervisé
4. Méthode de la régression linéaire simple
5. Apprentissage par descente de gradient
6. Méthode de la régression linéaire multiple
7. Méthode de la régression logistique
8. Graphe de dépendance, *forward pass*, et *backward pass*

Problèmes abordés

1. Prédiction de la note d'un étudiant à l'examen final
2. Prédiction d'une réussite ou non à l'examen

Apprentissage automatique

Dans notre vie quotidienne

Nous utilisons régulièrement des outils basés sur de l'apprentissage automatique...

...sans forcément nous en rendre compte



Pouvez-vous donner quelques exemples d'applications usuelles ?

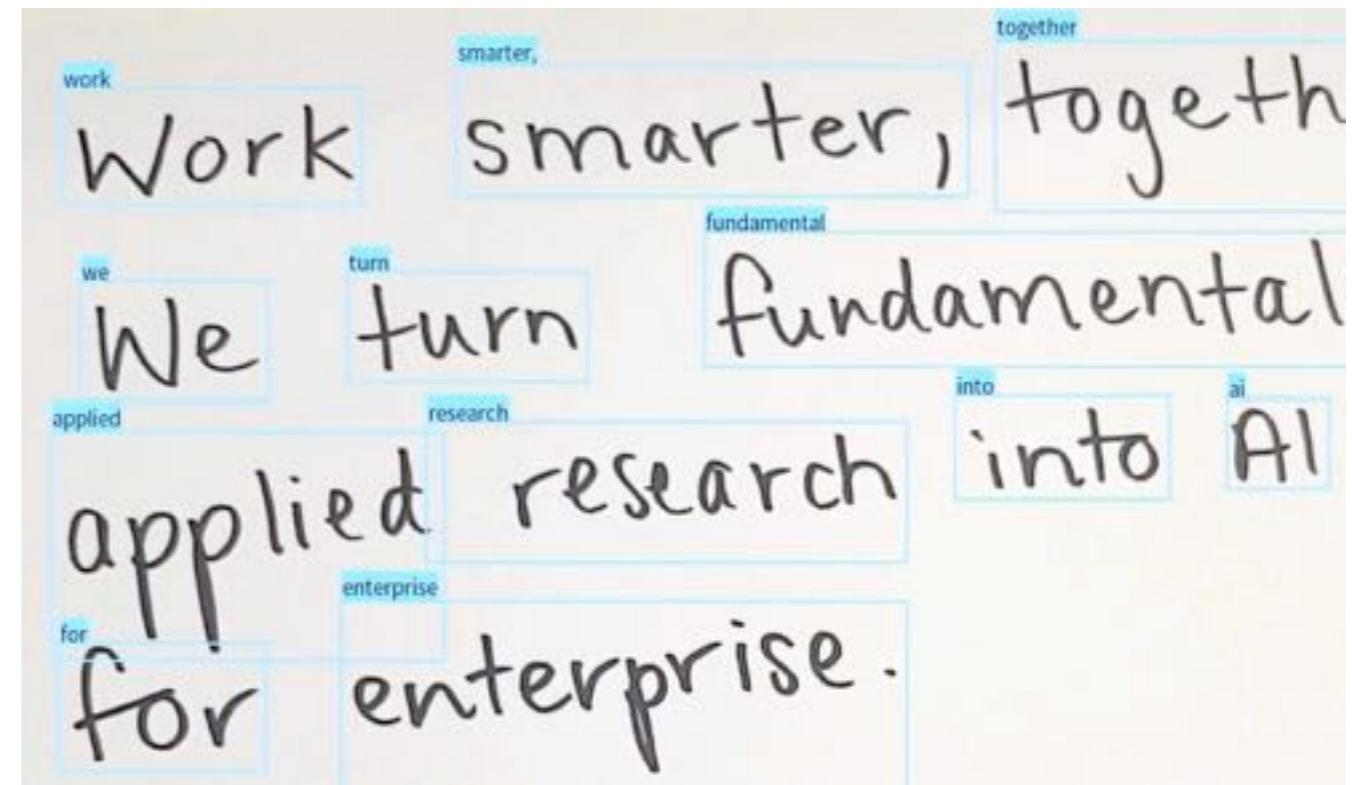
Reconnaissance faciale



<https://github.com/justadudewhohacks/face-recognition.js>

Reconnaissance de caractères (OCR - optical character recognition)

Reconnaissance de caractères



ELEMENT AI
<https://www.elementai.com/api/ocr> servicenow®

Dispatching automatique de lettres postales

Reconnaissance de réponses dans un formulaire administratif

Objectif simple, mais il est extrêmement difficile de construire un algorithme pour cette tâche

Conduite autonome



<https://www.tesla.com/AI>

Moteur de recherche

A screenshot of a Google search interface showing search suggestions for the query "apprentissage auto". The suggestions include:

- Apprentissage auto
- apprentissage automatique
- apprentissage autonome
- apprentissage autorégulé
- apprentissage autonomie adolescent
- apprentissage autorégulé définition
- apprentissage autonome ulaval
- apprentissage autodidacte
- apprentissage autonome robot
- apprentissage automobile
- apprentissage automatique mémoire

Below the suggestions are two buttons: "Recherche Google" and "J'ai de la chance". A small link "Signaler des prédictions inappropriées" is also visible.

A screenshot of a Google search results page for the query "apprentissage automatique". The results include:

- A snippet from Wikipedia: "« apprentissage machine », apprentissage artificiel ou apprentissage statistique est un champ d'étude de l'intelligence artificielle qui se fonde sur des approches mathématiques et statistiques pour donner aux ordinateurs la capacité d'« apprendre » à partir de données, c'est-à-dire d'améliorer leurs performances à ...". Below it is a link to "fr.wikipedia.org › wiki › Apprentissage_automatique".
- A snippet from Oracle: "Apprentissage automatique — Wikipédia". Below it is a link to "www.oracle.com › ... › Solutions › Intelligence artificielle".
- A snippet from Université de Montréal: "admission.umontreal.ca › programmes › dess-en-appre...". Below it is a link to "D.E.S.S. en apprentissage automatique - Université de Montréal".
- A snippet from Ivado: "ivado.ca › evenements › ecole-en-apprentissage-autome...". Below it is a link to "École en apprentissage automatique (2e édition) | Ivado".

Deux types d'apprentissage

L'auto-complétion de ma recherche

Les pages Web les plus probables

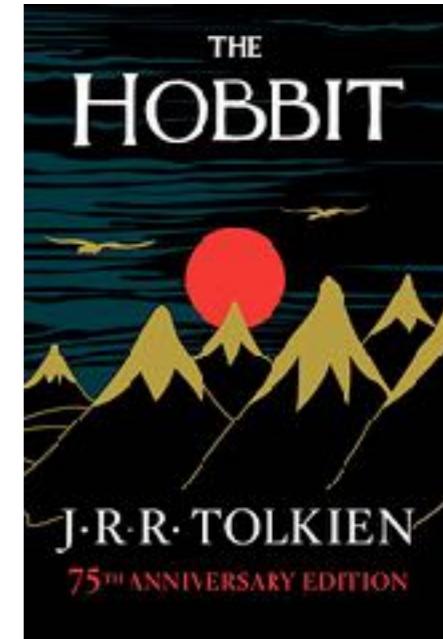
Et cela, étant donné le profil d'une personne

Traduction automatique du langage

DÉTECTOR LA LANGUE ANGLAIS FRANÇAIS ARABE

In a hole in the ground there lived a hobbit. Not a nasty, dirty, wet hole, filled with the ends of worms and an oozy smell, nor yet a dry, bare, sandy hole with nothing in it to sit down on or to eat: it was a hobbit-hole, and that means comfort.

247/5000



FRANÇAIS ANGLAIS ARABE

2020

Dans un trou dans le sol vivait un hobbit. Pas un trou désagréable, sale et humide, rempli d'extrémités de vers et d'une odeur suintante, ni encore un trou sec, nu et sablonneux sans rien pour s'asseoir ou pour manger: c'était un hobbit-trou, et que signifie confort.

FRANÇAIS ANGLAIS ARABE

2021

Dans un trou dans le sol vivait un hobbit. Pas un trou sale, sale, humide, rempli de bouts de vers et d'une odeur de suin, ni encore un trou sec, nu, sablonneux, sans rien pour s'asseoir ou manger : c'était un trou de hobbit, et ça veut dire confort.

FRANÇAIS ANGLAIS ARABE

La traduction n'est pas parfaite

Donne néanmoins l'idée générale

Peut servir de base à de meilleures traductions

Traduction améliorée !

Encore sujette à amélioration

Chatbots

Chatbots vocaux



amazon Alexa



Apple - Siri

Utilisation multiple de l'apprentissage automatique

Reconstruction d'une phrase à partir d'une onde acoustique

Analyse de la requête

Réponse à donner ou prise de décision

Chatbots textuels

Support clientèle sur un site web

Générateur d'histoires...



<https://play.aidungeon.io/>

Contraction de textes



Input: Article 1st sentence	Model-written headline
metro-goldwyn-mayer reported a third-quarter net loss of dlr\$ 16 million due mainly to the effect of accounting rules adopted this year	mgm reports 16 million net loss on higher revenue
starting from july 1, the island province of hainan in southern china will implement strict market access control on all incoming livestock and animal products to prevent the possible spread of epidemic diseases	hainan to curb spread of diseases
australian wine exports hit a record 52.1 million liters worth 260 million dollars (143 million us) in september, the government statistics office reported on monday	australian wine exports hit record high in september

<https://ai.googleblog.com/2016/08/text-summarization-with-tensorflow.html> [Google AI, 2016]

Utile pour saisir les idées principales d'un long texte

Alice et Bob ont pris le train pour **aller au zoo**.

Ils ont **vu** un bébé **girafe**, un **lion**, et tout plein **d'oiseaux** tropicaux.

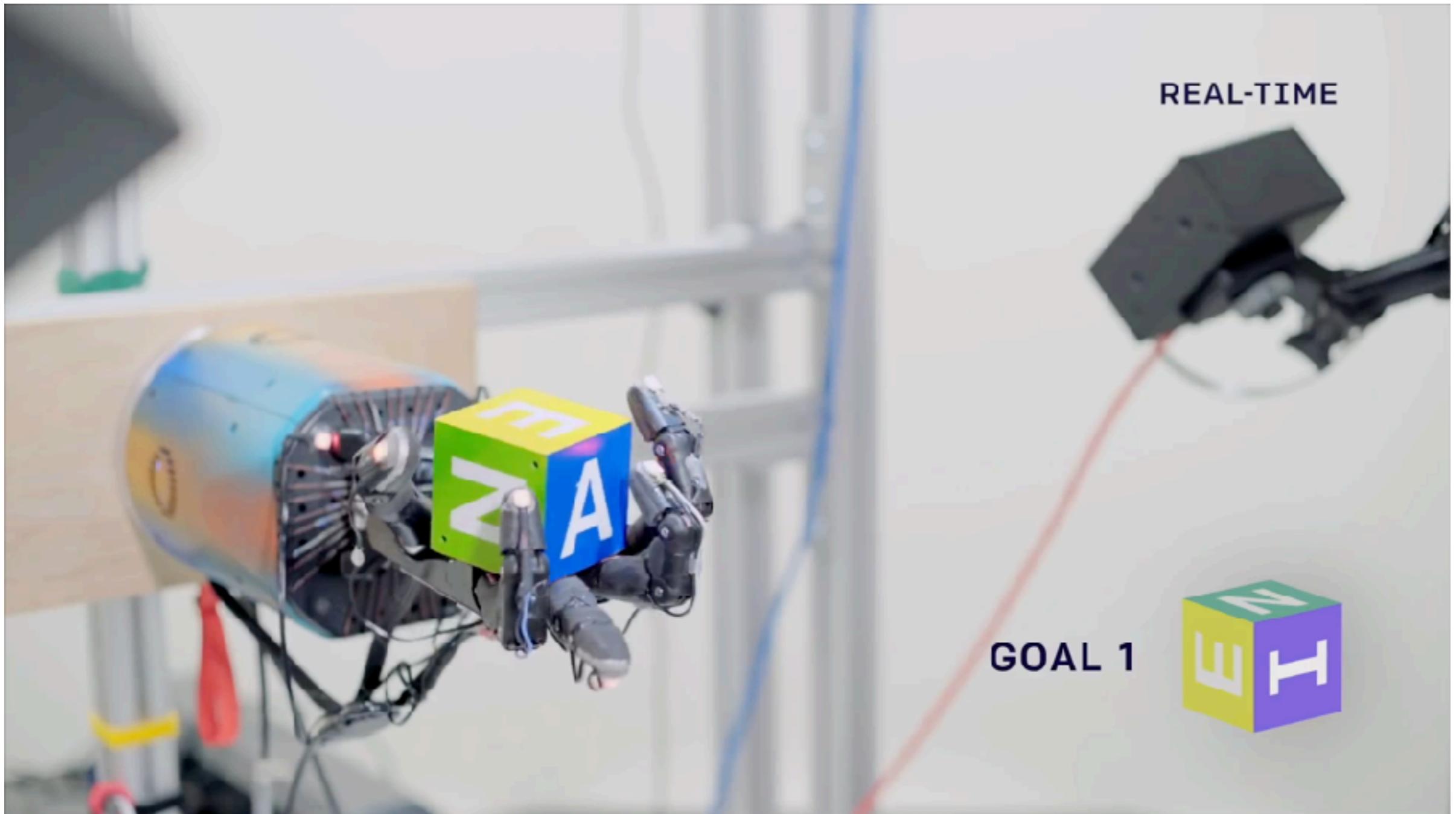


Alice et Bob sont partis au zoo et ont vu beaucoup d'animaux.

Extraction des éléments importants

Respect des règles de la langue

Application en robotique

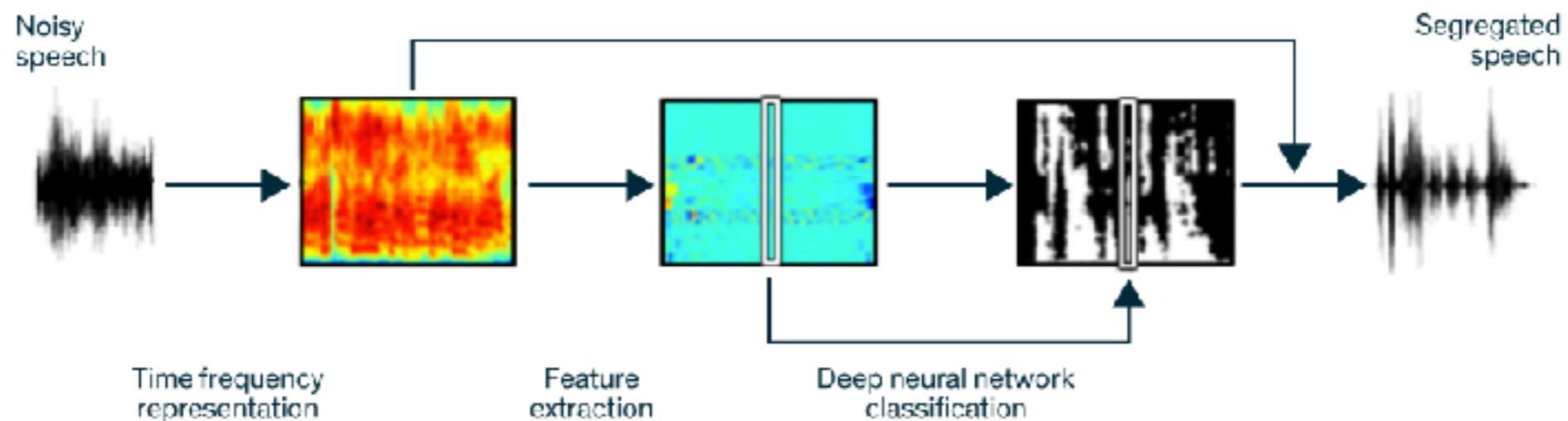


Learning dexterous in-hand manipulation [OpenAI, 2020]

<https://openai.com/blog/learning-dexterity/>

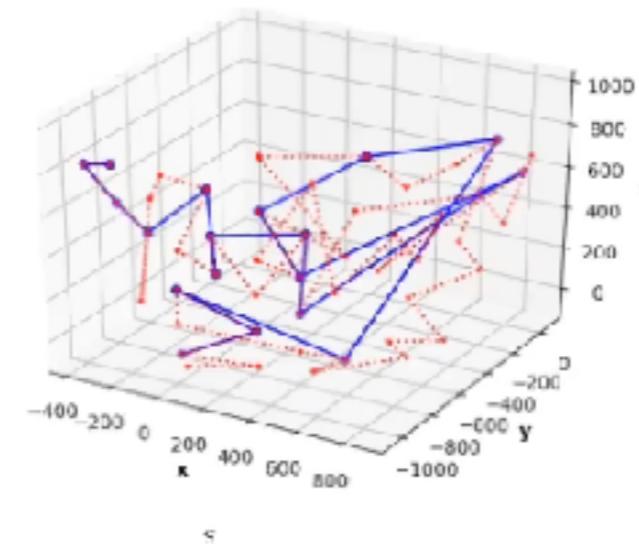
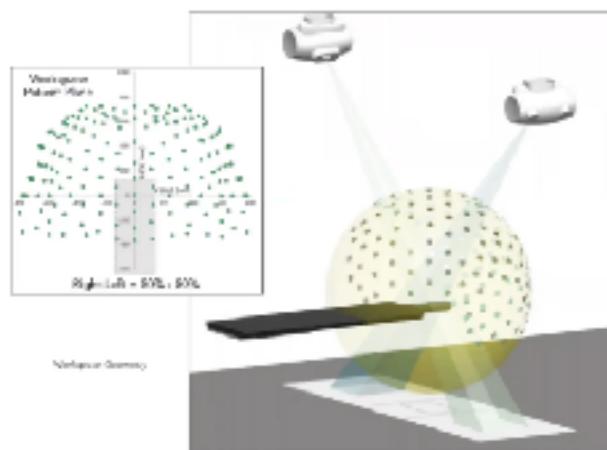
Applications dans le domaine médical

Prothèse auditive pour mal-entendants



<https://spectrum.ieee.org/consumer-electronics/audiovideo/deep-learning-reinvents-the-hearing-aid>

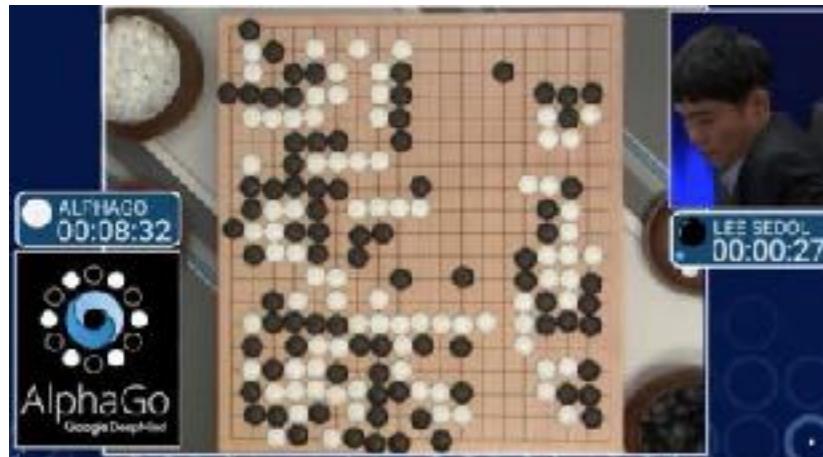
Amélioration du processus de traitement pour une radiothérapie



Deep Q-learning for simultaneous beam orientation and trajectory optimization for Cyberknife [Peyman et al., 2021]
(Projet de recherche mené dans notre laboratoire)

Apprentissage pour les jeux

Jeu de Go (2016)



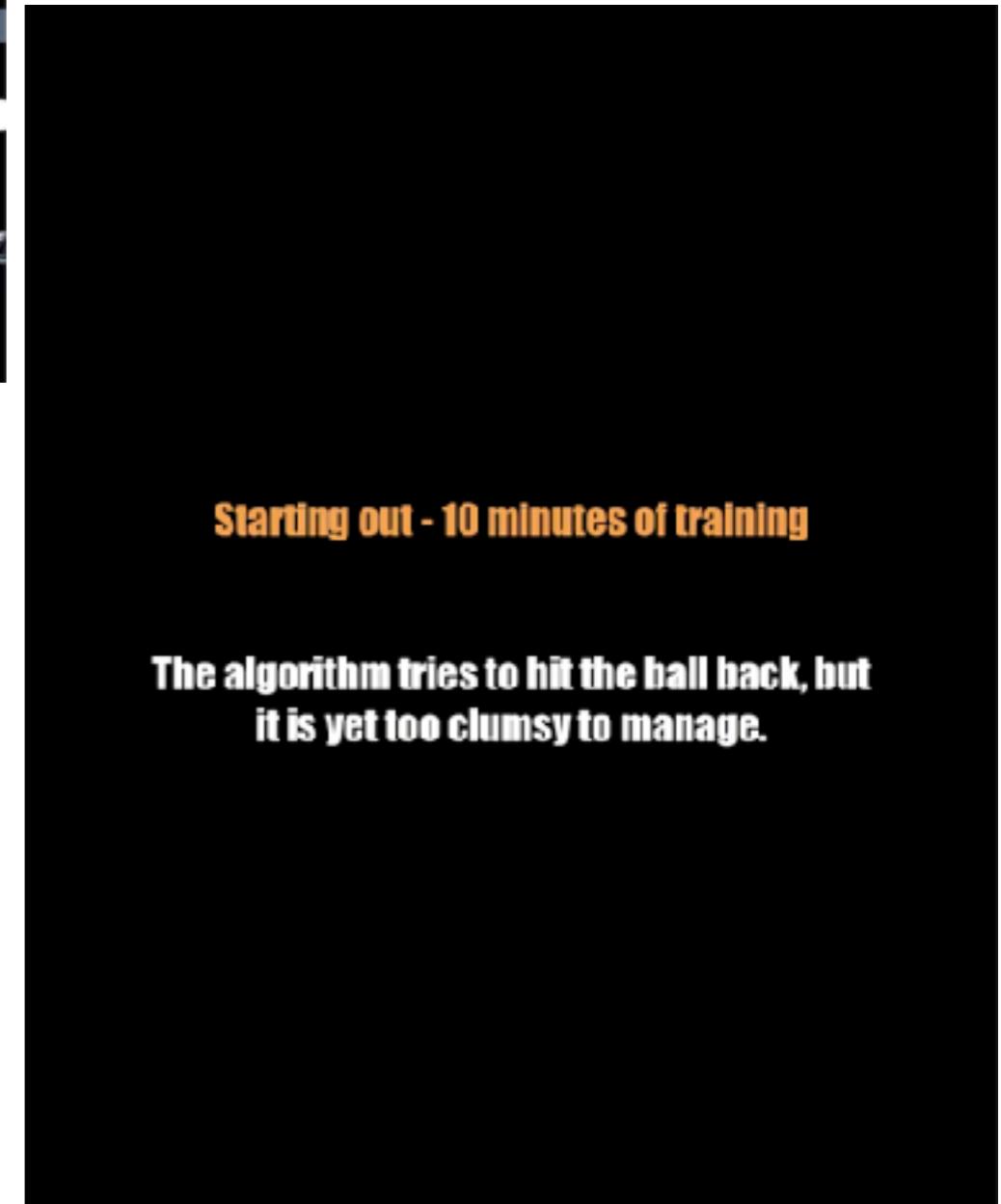
Mastering the game of Go with deep neural networks and tree search [DeepMind, 2016]

Défaite du champion du monde de Go (Lee Sedol) par une IA Starcraft 2 (2019)



AlphaStar: Mastering the real-time strategy game StarCraft II [DeepMind, 2019]

Atari (2013)



Playing Atari with Deep Reinforcement Learning [DeepMind, 2013]

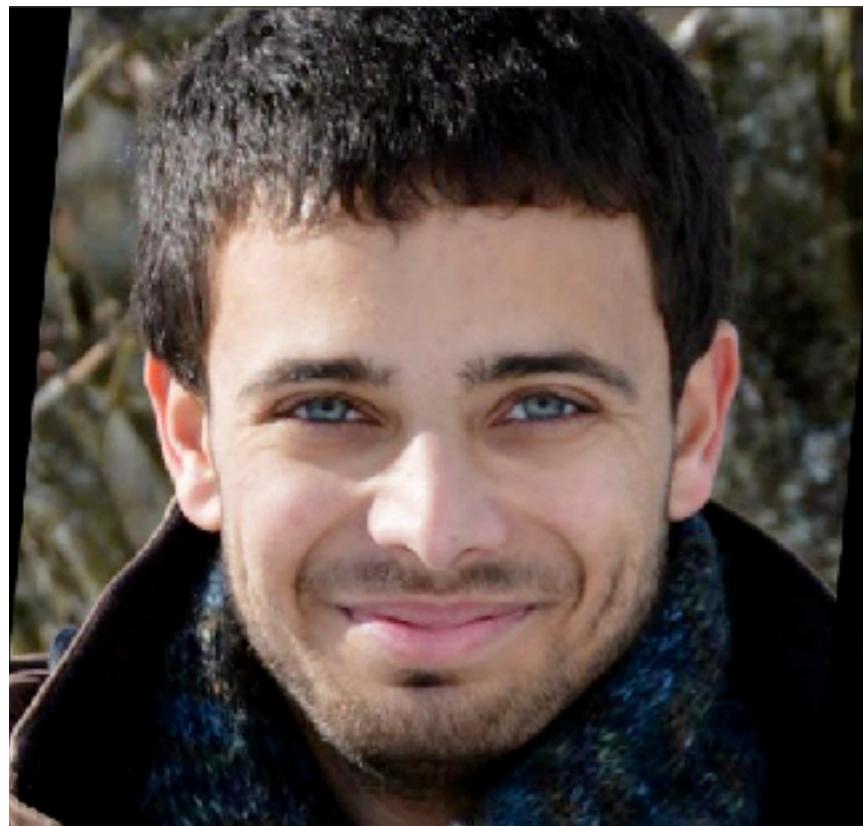
Créations artistiques

Transfert du style d'une image

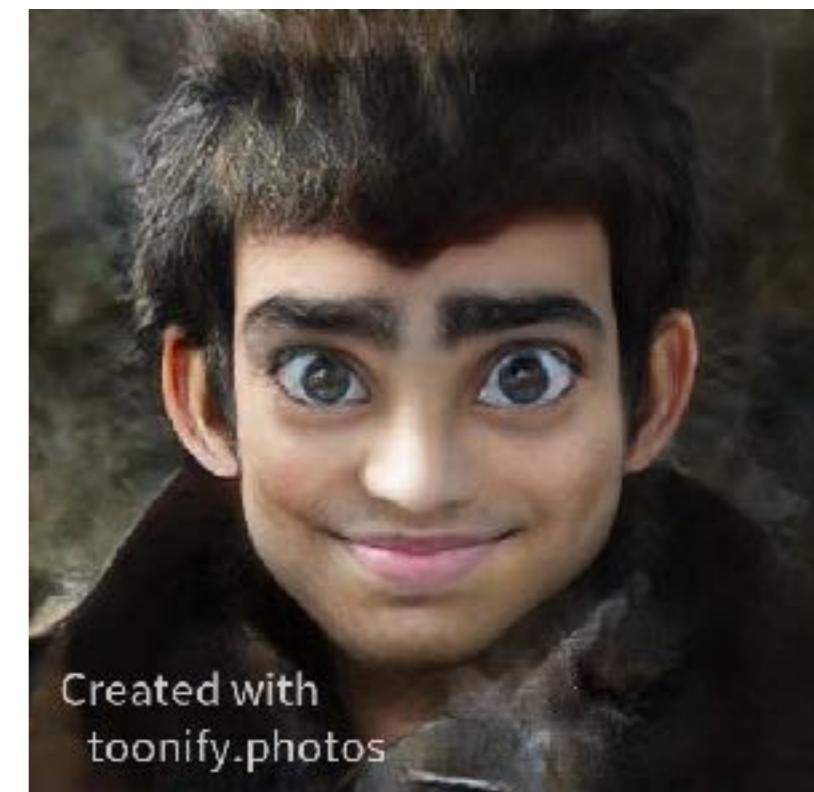


Image style transfer using convolutional neural networks [Gatys et al. 2016]

Créations artistiques



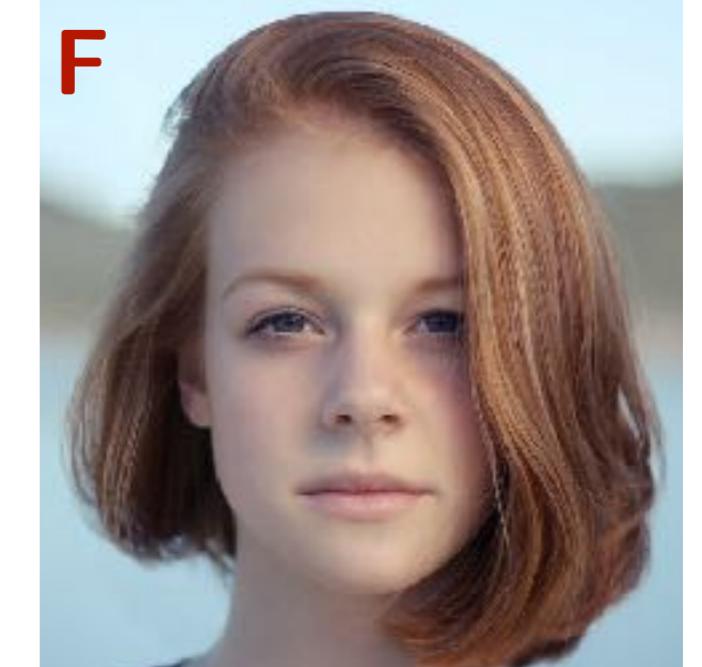
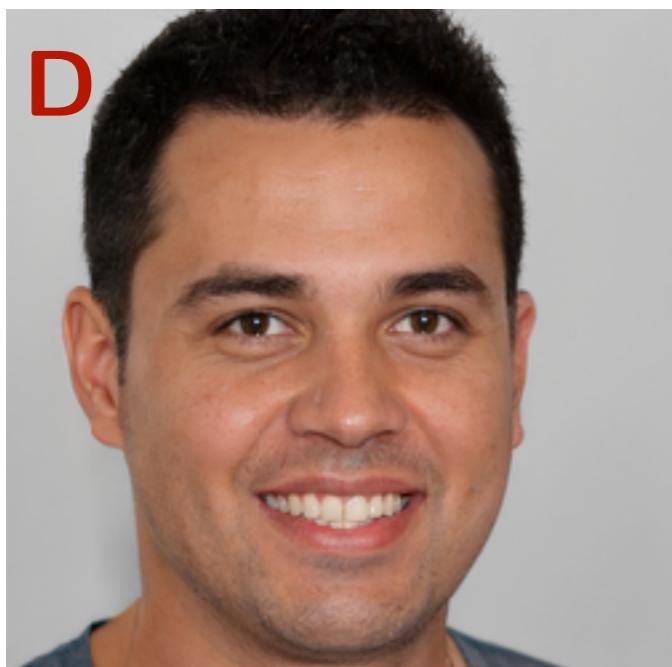
<https://makemeazombie.com/about.html>



Created with
toonify.photos

<https://toonify.photos/>

Génération automatique



Une seule de ces photos provient d'une vraie personne, laquelle ?

<https://thispersondoesnotexist.com/>

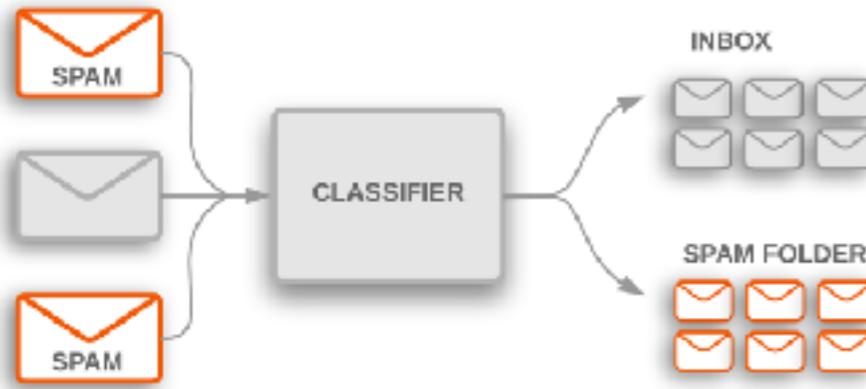
Image par Free-Photos de Pixabay
Quentin Cappart

Cela étant dit, je n'en ai pas la confirmation !

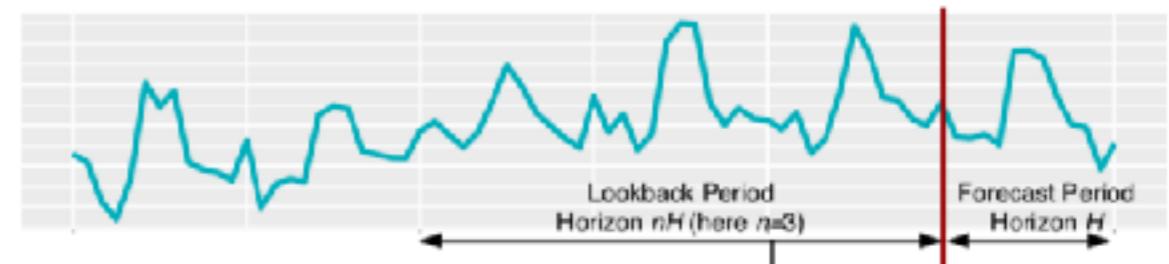
Ouvre directement la porte à de nombreuses dérives éthiques

Autres applications

Détection de spam



Prédiction de ventes



[N-BEATS, Oreshkin et al. 2020]

Enormément de ressources sur le Web



<https://www.youtube.com/watch?v=Bui3DWs02h4>



Open AI joue à cache-cache.. et casse le jeu!

3 780 379 vues • il y a 1 an

Jeux un océan de Weights & Biases here et inscrivez-vous gratuitement pour une démo: <https://www.wandb.com/papers>

Leur billet de blog est disponible ici: <https://www.wandb.com/articles/better...>

Le papier "Emergent Tool Use from Multi-Agent Interaction"

[LIRE LA SUITE](#)



Beaucoup d'applications et de données
Présenté sous la forme de compétitions

De plus en plus d'entreprises passent par Kaggle pour la recherche de solutions à leurs problèmes

Définition de l'apprentissage automatique

Définition moderne (parmi d'autres)



"A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E"

<http://www.cs.cmu.edu/afs/cs.cmu.edu/user/mitchell/ftp/mlbook.html>

Tom Mitchell (1997)

Fait apparaître trois éléments fondamentaux

- (1) L'apprentissage est dédié pour une tâche précise (T)
- (2) L'apprentissage a besoin d'expériences passées (E - données)
- (3) Besoin d'avoir un moyen d'évaluer les performances d'un modèle (P)

Confusion avec l'intelligence artificielle



"Most of what is labeled AI today, particularly in the public sphere, is actually machine learning (ML), a term in use for the past several decades"

<https://hdsr.mitpress.mit.edu/pub/wot7mkc1/release/8>

Michael Jordan (2019)

L'apprentissage automatique est un sous-domaine de l'intelligence artificielle

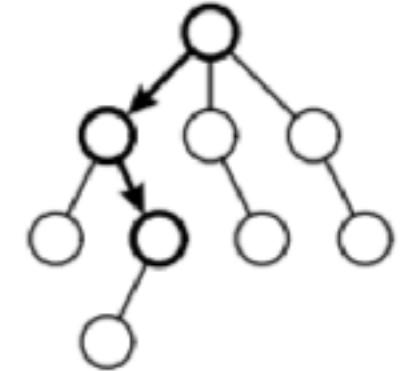
Contenu du cours

Raisonnement par recherche (essais-erreurs avec de l'intuition)

Module 1: Stratégies de recherche

Module 2: Recherche en présence d'adversaires

Module 3: Recherche locale



Raisonnement logique

Module 4: Programmation par contraintes

Module 5: Agents logiques

Module 6: Logique du premier ordre et inférence

SS SSSS Breeze		Breeze	PIT
Breeze	SS SSSS Breeze	Breeze	PIT
SS SSSS Breeze		Breeze	Breeze
Breeze	Breeze	PIT	Breeze

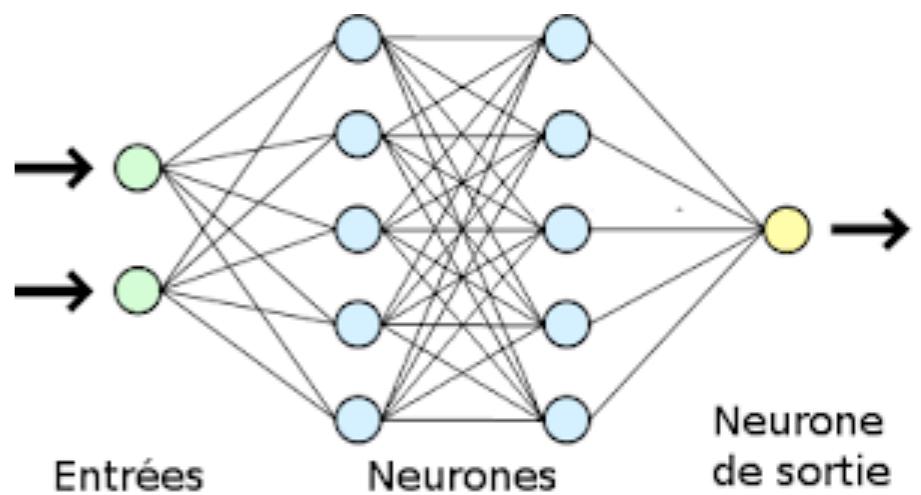
Raisonnement par apprentissage

Module 7: Apprentissage supervisé

Module 8: Réseaux de neurones et apprentissage profond

Module 9: Apprentissage non-supervisé

Module 10: Apprentissage par renforcement



Applications industrielles

Présentation d'une entreprise utilisant des techniques d'IA

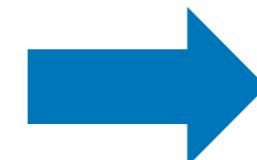
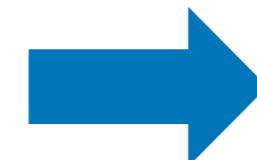
Objectif de l'apprentissage automatique

Objectif

Construire une fonction capable d'effectuer une prédiction, pour une situation spécifique



Dashcam



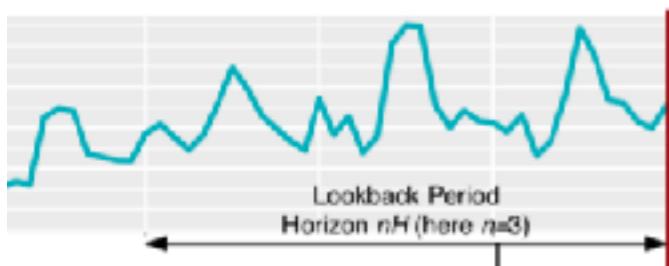
Freiner

"This movie was far better than the trailer made it look."

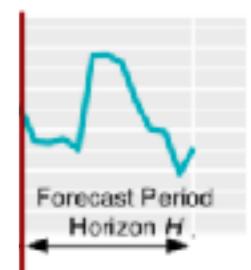
Review



Score



Ventes précédentes



Ventes futures

Notez la grande variété des situations pouvant être considérées (outil utile dans énormément de contextes)

Intérêt de l'apprentissage automatique



Pourquoi ne pas résoudre ces problèmes avec les techniques déjà vues ?

Difficulté de la tâche

Certaines tâches sont très difficiles à résoudre avec les techniques précédentes



Reconnaissance d'images

- (1) Déceler manuellement des caractéristiques (approche traditionnelle)
- (2) Exploiter les similarités avec d'autres photos déjà vues (apprentissage)

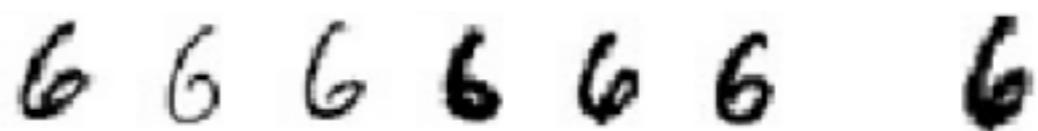
Cette deuxième option est plus facile (et efficace) à mettre en oeuvre

Besoin de généraliser

Il est trivial d'implémenter un algorithme mémorisant et reproduisant les expériences déjà vues



Quelle est la limitation majeure de cet algorithme ?



Il est incapable de généraliser à des nouvelles expériences, similaires, mais non-vues

Un intérêt majeur de l'apprentissage automatique est de pouvoir généraliser

Utilisation des techniques déjà vues

Beaucoup de méthodes d'apprentissage automatique sont basées sur des algorithmes déjà vus

Exemple: un clustering peut se faire avec de la recherche locale

Ainsi, la frontière entre les différents types de raisonnement n'est pas si stricte...

Principaux types d'apprentissage

Apprentissage supervisé

Apprentissage qui utilise des données labellisées (on connaît leur vraie valeur - *ground truth*)

Similaire à un professeur qui va vous enseigner des notions

Il est important d'avoir ces données à disposition



Apprentissage par renforcement

Apprentissage par l'expérience et par essais-erreurs

Il n'y a pas de données labellisées, simplement un signal de récompense

Signal de récompense: positif en cas de bonne action, négatif sinon

L'agent va effectuer des actions, et recevoir une récompense

Son objectif est d'obtenir le plus de récompenses lors de ces actions

Il est important de savoir comment récompenser les actions faites

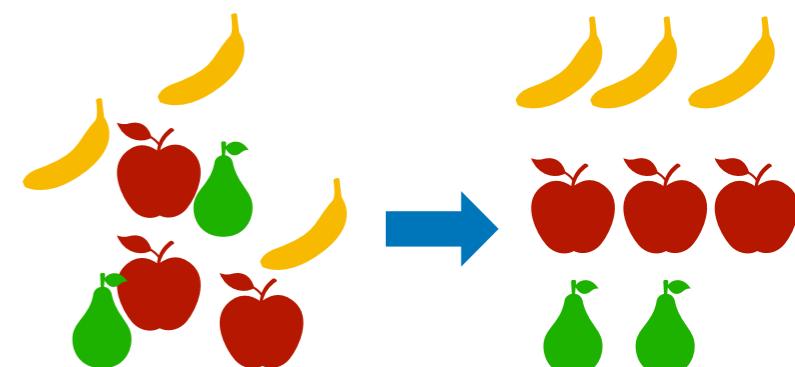


Apprentissage non-supervisé

Apprentissage qui utilise des données non labellisées, et sans récompense

L'objectif est d'apprendre des associations de similarité entre les données

Exemple: détecter une anomalie (malware parmi des programmes sains)



Limitations de l'apprentissage automatique

Limitations de l'apprentissage automatique

Certaines tâches ne sont pas propices à être résolues avec de l'apprentissage automatique

Différents critères rentrent en jeu (efficacité, éthique, type d'application, etc.)



(1) Tâches très bien résolues avec des algorithmes traditionnels

Trier une liste

Trouver le plus court chemin entre deux noeuds dans un graphe (algorithme de Dijkstra)

(2) Tâches trop coûteuse pour entraîner un modèle

Systèmes temps-réels

Systèmes avec peu de ressources

(3) Peu ou aucune donnée à utiliser

Prédiction d'une maladie rare

Protection de la vie privée

(4) Situations éthiques délicates

Prédiction si un étudiant doit être admis à une université ou non

Prédiction si une personne a droit à un prêt hypothécaire

(5) Tâches nécessitant des garanties

Systèmes critiques où une erreur de prédiction n'est pas permise, ou très coûteuse à réparer

Table des matières

Apprentissage supervisé

-  1. Motivation et intérêt de l'apprentissage automatique
-  2. Classification des principaux types d'apprentissage
- 3. Définition de l'apprentissage supervisé
- 4. Méthode de la régression linéaire simple
- 5. Apprentissage par descente de gradient
- 6. Méthode de la régression linéaire multiple
- 7. Méthode de la régression logistique
- 8. Graphe de dépendance, *forward pass*, et *backward pass*

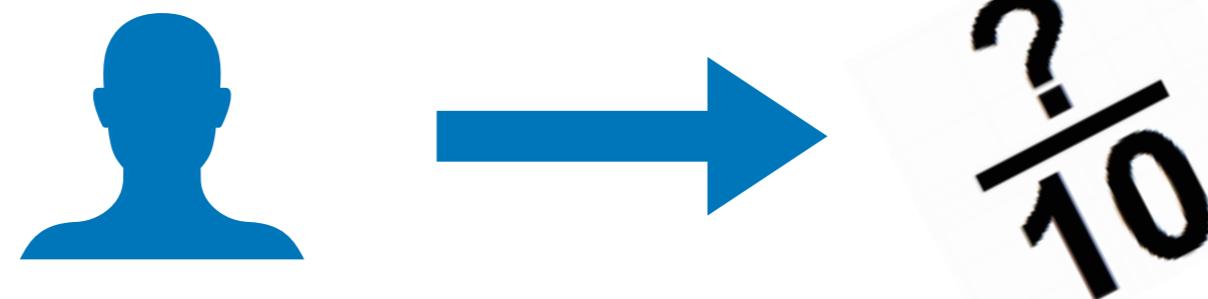
Problèmes abordés

- 1. Prédiction de la note d'un étudiant à l'examen final
- 2. Prédiction d'une réussite ou non à l'examen

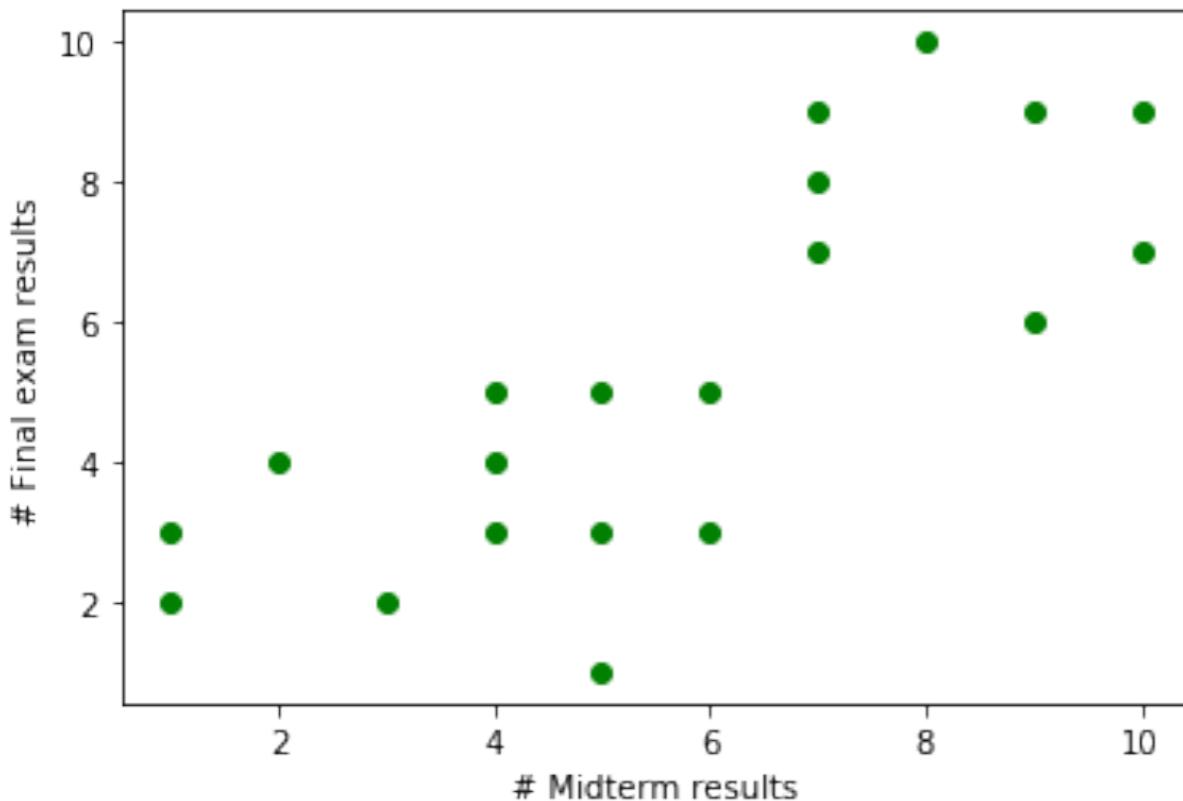
Cas d'étude: le problème de l'étudiant

Problème de l'étudiant

Suite aux notes obtenus à l'intra, un étudiant se demande combien il obtiendra à son examen



Pour cela, il se base sur les résultats des étudiants de l'année précédente



Formalisation du problème

Un étudiant est caractérisé par sa note à l'intra

$$x \in [0..10]$$

Le résultat est la note à l'examen final

$$y \in [0..10]$$

On souhaite construire une fonction de prédiction

$$f: [0..10] \rightarrow [0..10]$$

Formalisation du problème

Principes de l'apprentissage supervisé

Présence de données historiques, dont on connaît la vraie valeur

Les données sont utilisées pour construire notre fonction de prédiction

La fonction va être utilisée pour prédire la valeur de nouvelles situations

Notations et vocabulaire général

Training set: les données utilisées pour construire la fonction

m : taille de l'ensemble

$$D : \left\{ (x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \dots, (x^{(m)}, y^{(m)}) \right\}$$

Data: une donnée bien précise

$$(x^{(i)}, y^{(i)}), \text{ avec } x \in [0..10] \text{ et } y \in [0..10]$$

$x^{(i)}$: feature de la donnée i (par la suite, nous verrons qu'on peut avoir plusieurs features)

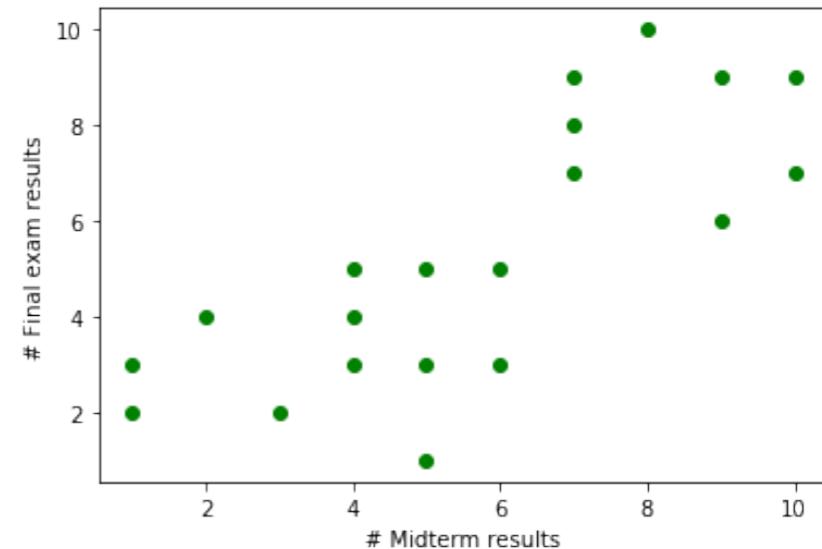
$y^{(i)}$: label de la donnée i (true value)

Fonction de prédiction:

$$f : [0..10] \rightarrow [0..10]$$

Valeur prédite: output de la fonction pour de nouvelles données

$\hat{y} = f(x)$: le chapeau indique qu'on a une valeur estimée



Fonction de prédiction



Notre objectif est de construire une fonction de prédiction.

Quelle forme de fonction utiliser ? (linéaire, polynomiale, exponentielle, etc.)

Hypothèse

Concept fondamental en apprentissage automatique

Préconception que l'on fait sur notre fonction

Approximation par une fonction linéaire

Hypothèse très simple

Peut approximer nos données de manière assez précise

Cette notion de précision doit encore être définie

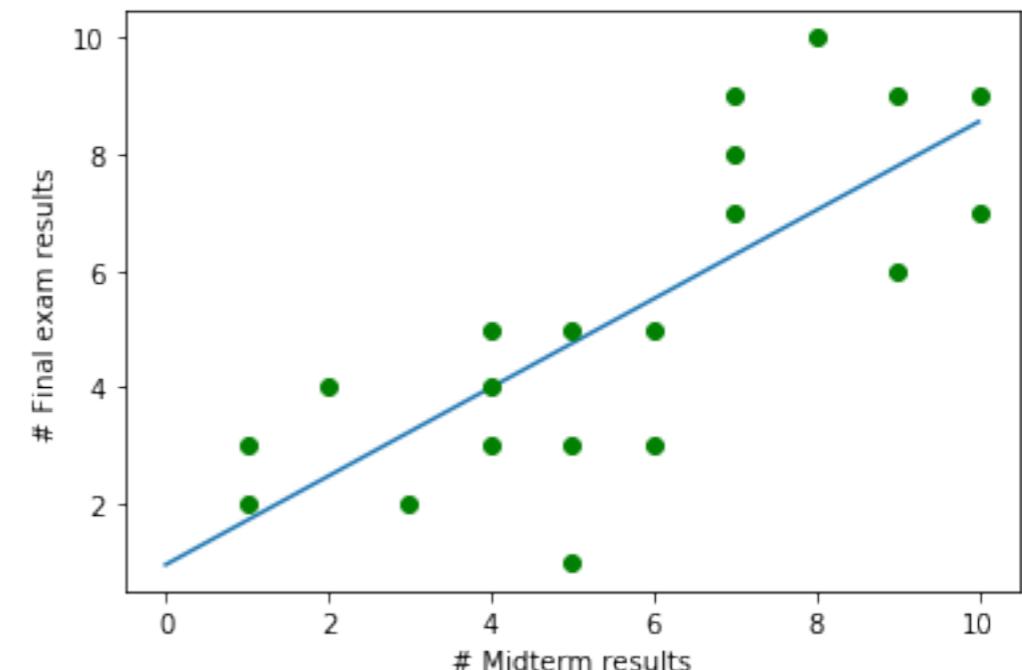
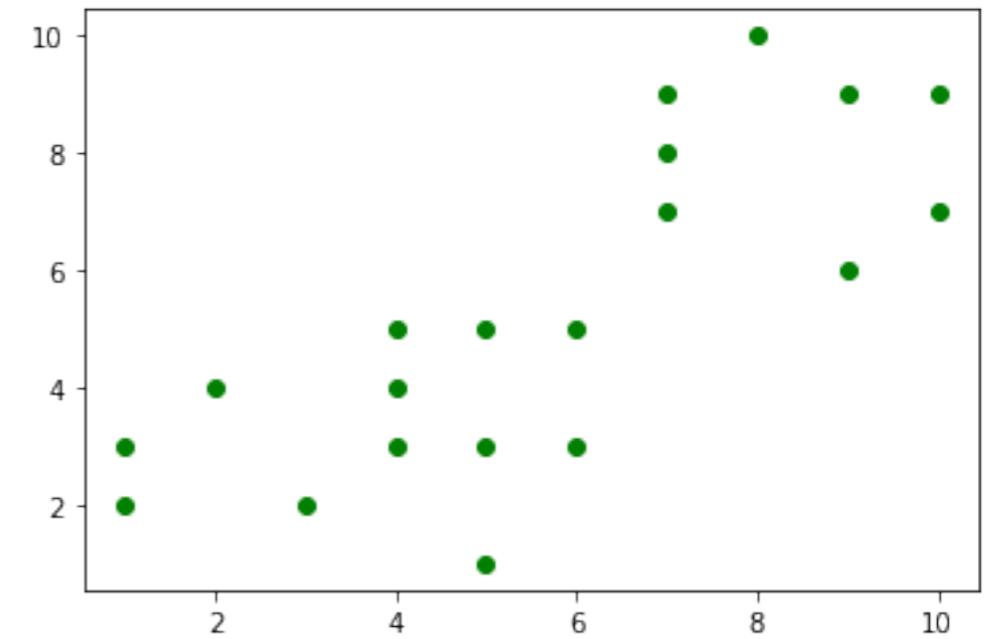
$$\hat{y} = f(x) = w \cdot x + b$$

x : caractéristique (feature)

b, w : paramètres de la fonction (parameters)

w : poids de la caractéristique x (weight)

b : biais (bias or intercept)



Régression linéaire simple

Régression linéaire simple

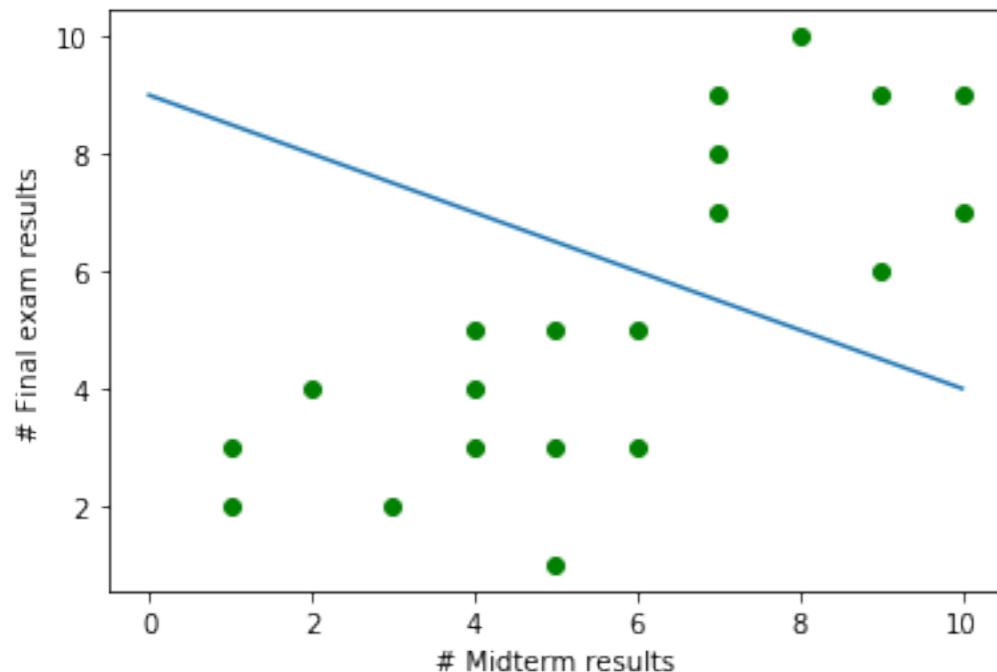
L'objectif est de construire la fonction de prédiction suivante

$$\hat{y} = f(x) = w \cdot x + b$$

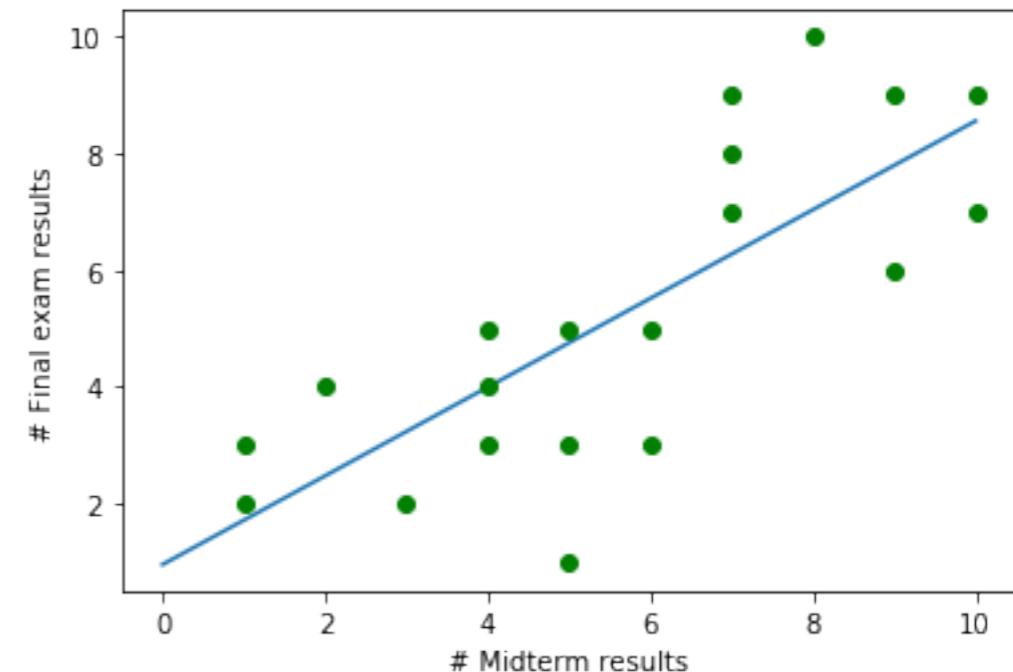
C'est-à-dire, déterminer les valeurs des paramètres w et b

Intuitivement, on souhaite obtenir la droite qui donne une bonne approximation de nos données

$$f(x) = -0.5x + 9$$



$$f(x) = 0.76x + 0.96$$



La fonction de gauche semble être de meilleure qualité

Fonction de coût

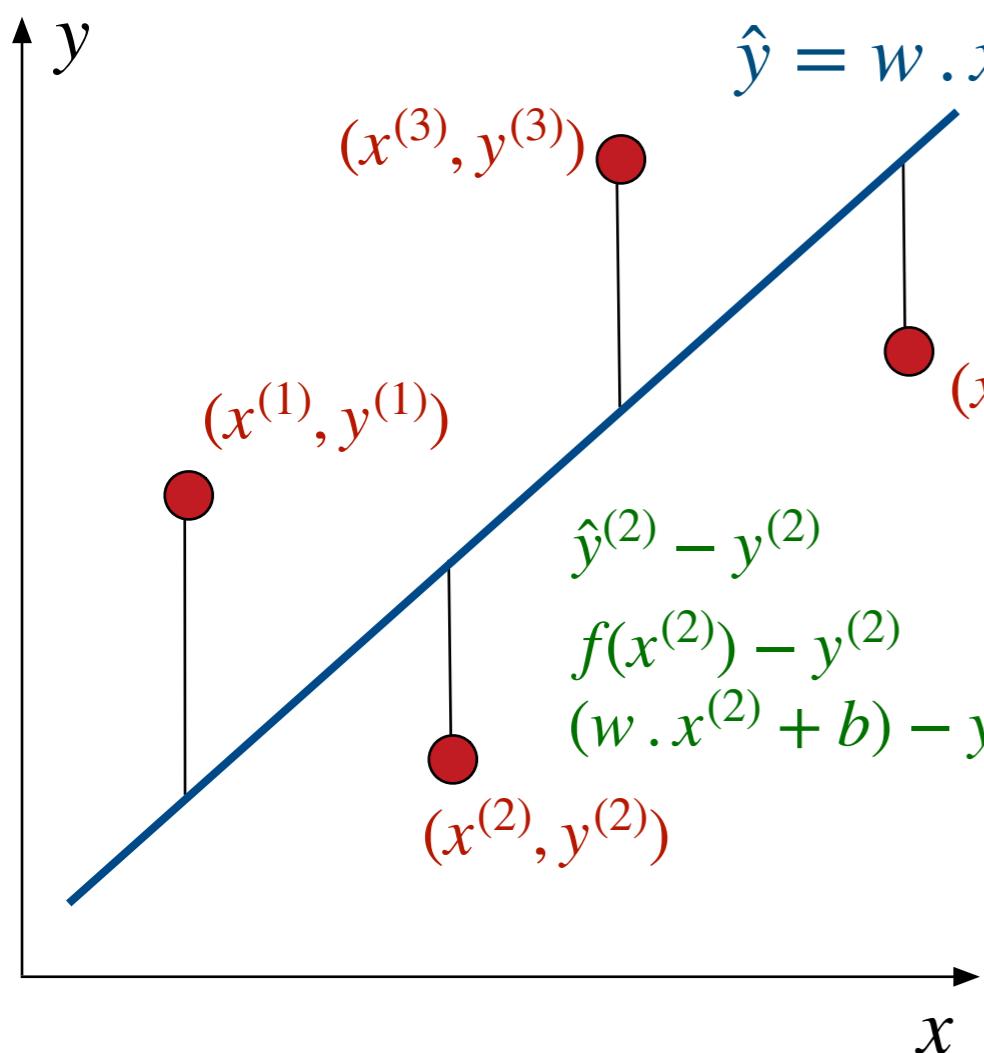


Comment formaliser la qualité d'une fonction par rapport à nos données ?

Formalisation de notre objectif

Intuitivement, on veut construire la fonction (linéaire) qui approxime le mieux nos données

Une métrique possible est de minimiser l'écart avec tous les points



$\hat{y} = w \cdot x + b$ Fonction d'écart (*loss function*)

Fonction choisie: erreur quadratique

$$L(\hat{y}^{(i)}, y^{(i)}) = (\hat{y}^{(i)} - y^{(i)})^2$$

Evite que les écarts positifs et négatifs se compensent

Forte pénalité pour les grands écarts

Fonction de coût

Moyenne des écarts entre nos données et notre fonction

$$J(w, b) = \frac{1}{2m} \sum_{i=1}^m L(\hat{y}^{(i)}, y^{(i)})$$

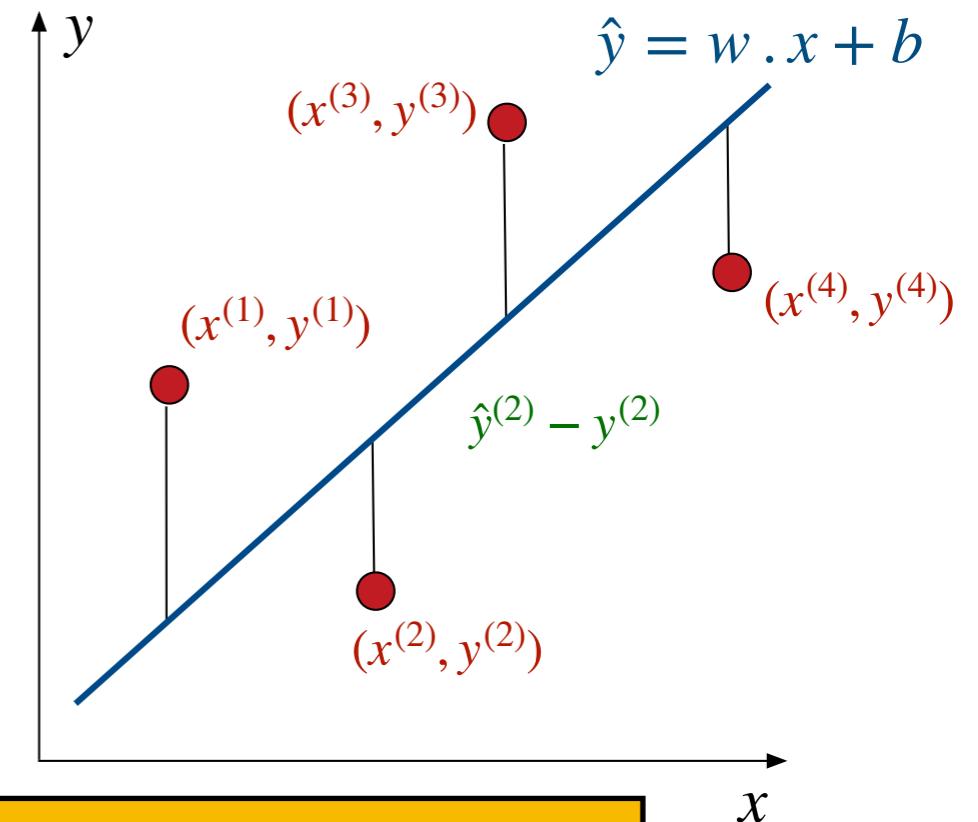
Note: le facteur $1/2$ ne change rien aux écarts, mais va simplifier les futurs calculs

Formalisation de l'apprentissage supervisé

Fonction de coût

Mesure d'à quel point une prédiction est loin des données

$$\begin{aligned} J(w, b) &= \frac{1}{2m} \sum_{i=1}^m L(\hat{y}^{(i)}, y^{(i)}) \\ &= \frac{1}{2m} \sum_{i=1}^m (\hat{y}^{(i)} - y^{(i)})^2 \text{ avec } \hat{y}^{(i)} = w \cdot x^{(i)} + b \end{aligned}$$



Autrement dit, l'apprentissage supervisé consiste à trouver la fonction qui approxime le mieux nos données

Entraîner un modèle équivaut à trouver les paramètres qui minimisent la fonction de coût

$$\min_{w,b} J(w, b)$$



$$\min_{w,b} \frac{1}{2m} \sum_{i=1}^m L(\hat{y}^{(i)}, y^{(i)})$$

Descente de gradient

Fonction à minimiser

$$J(w, b) = \frac{1}{2m} \sum_{i=1}^m (\hat{y}^{(i)} - y^{(i)})^2 \text{ avec } \hat{y}^{(i)} = w \cdot x^{(i)} + b$$

?

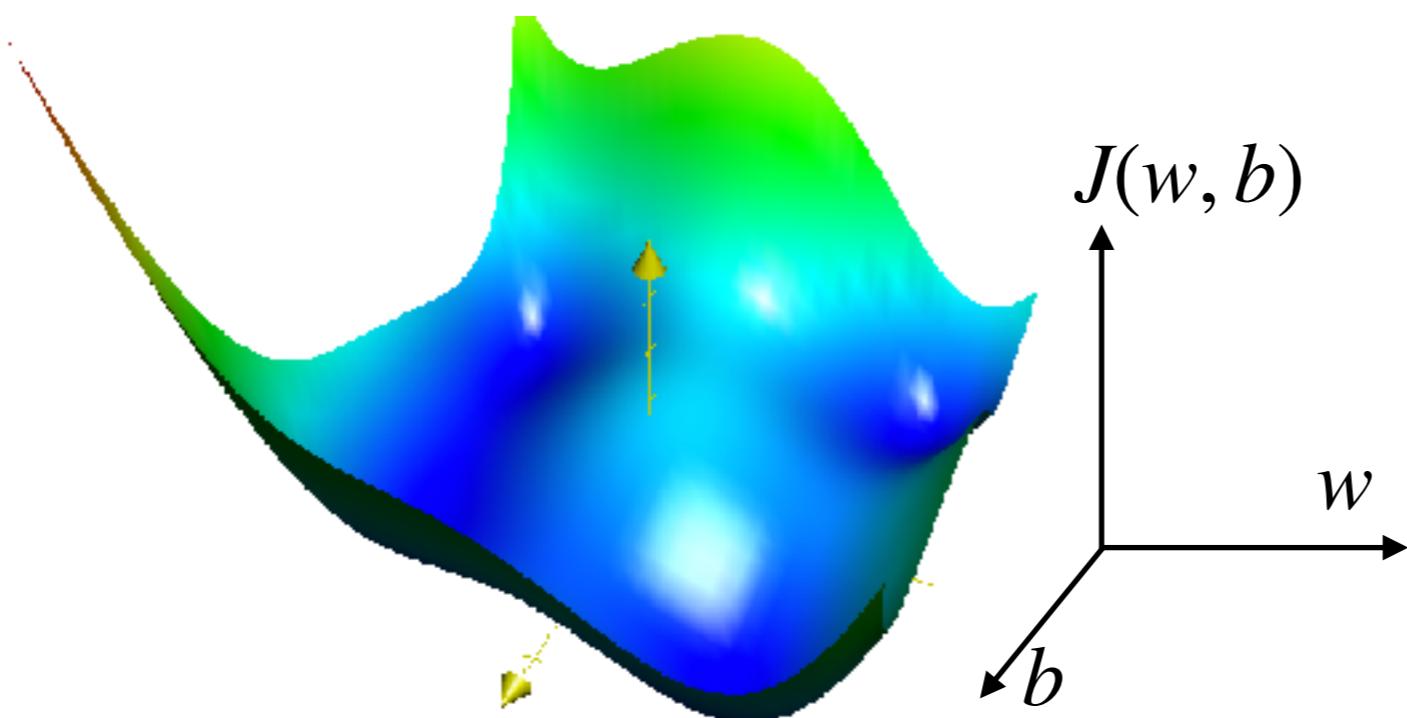
Comment minimiser cette fonction ?

Descente de gradient

Une méthode (parmi d'autres) pour minimiser une fonction

Nécessite d'avoir une fonction de coût principalement différentiable

Modifie/Améliore les paramètres étape par étape en suivant la direction donnée par le gradient



Basé sur trois ingrédients essentiels

Choix d'une position initiale

Direction du mouvement à prendre

Intensité du mouvement à prendre

Peut également être vue comme un *hill climbing* (module 3) dans un espace continu

Descente de gradient: cas à 1 paramètre

Application d'une descente de gradient

$$J(w) = w^2, w \in \mathbb{R}$$

Initialisation: valeur aléatoire pour w

Direction: obtenue avec la dérivée de $J(w)$ (direction de la pente la plus forte)

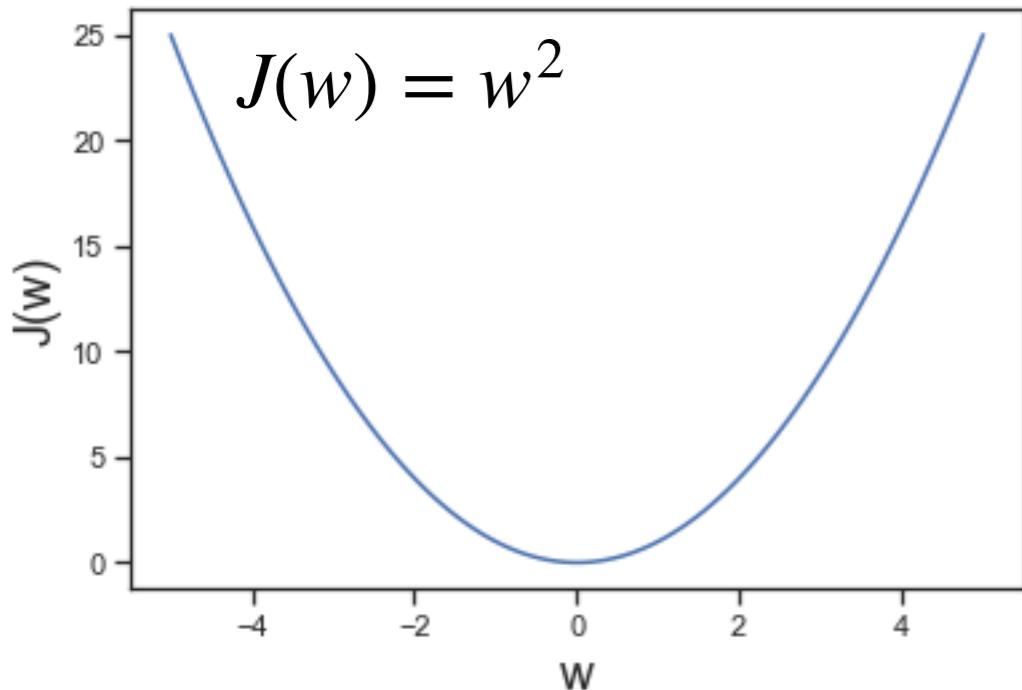
$$\frac{dJ(w)}{dw} = 2w$$

Intensité: obtenue par une valeur positive (*learning rate*)

$$\alpha \in]0,1]$$

Etape de descente de gradient: mise à jour de w

$$w := w - \alpha \frac{dJ(w)}{dw}$$



Dérivée de $J(w)$: $\frac{dJ(w)}{dw} = 2w$

Cas 1: w est positif

$$w > 0 \rightarrow \frac{dJ(w)}{dw} > 0$$

L'étape de mise à jour va diminuer w

Cas 2: w est négatif

$$w < 0 \rightarrow \frac{dJ(w)}{dw} < 0$$

L'étape de mise à jour va augmenter w

On répète ce processus jusqu'à une situation où le coût ne diminue plus (convergence)

Descente de gradient: cas général

Descente de gradient dans le cas général

Fonction à une seule variable: la direction de la pente la plus forte est donnée par la **dérivée**

Fonction à plusieurs variables: la direction de la pente la plus forte est donnée par le **gradient**

Gradient d'une fonction: vecteur constitué des dérivées partielles de chaque variable

$$\nabla f(x_1, \dots, x_n) = \left(\frac{\partial f}{\partial x_1}, \dots, \frac{\partial f}{\partial x_n} \right)$$

Notre fonction de coût

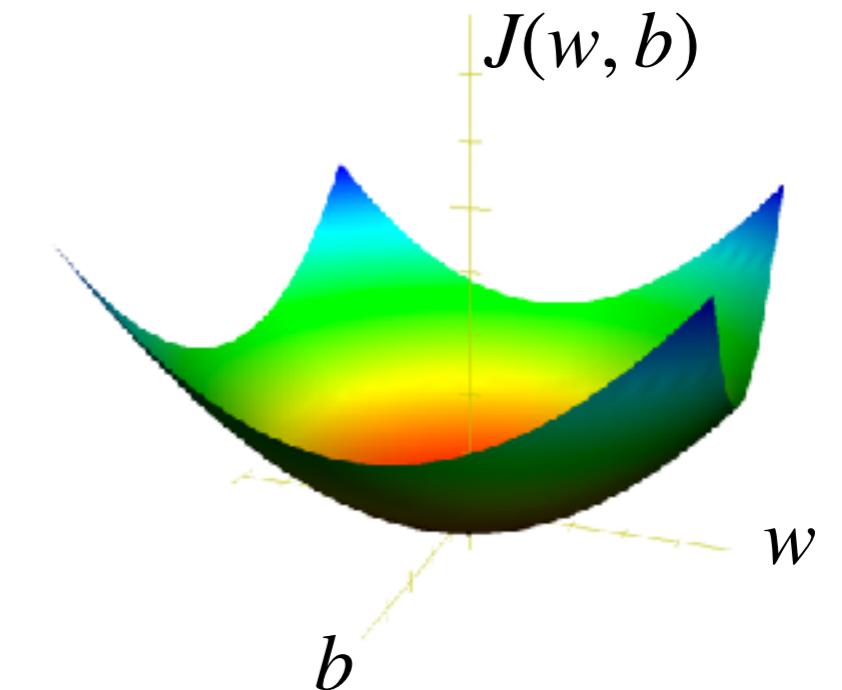
$$J(w, b) = \frac{1}{2m} \sum_{i=1}^m (\hat{y}^{(i)} - y^{(i)})^2 \text{ avec } \hat{y}^{(i)} = w \cdot x^{(i)} + b$$

$$\nabla J(w, b) = \left(\frac{\partial J}{\partial w}, \frac{\partial J}{\partial b} \right)$$

Mise à jour des paramètres

Chaque paramètre est mis à jour suivant sa dérivée partielle

$$w_i = w_i - \alpha \frac{\partial J(w, b)}{\partial w_i} \quad b = b - \alpha \frac{\partial J(w, b)}{\partial b}$$



Convexité et gradient de la fonction de coût

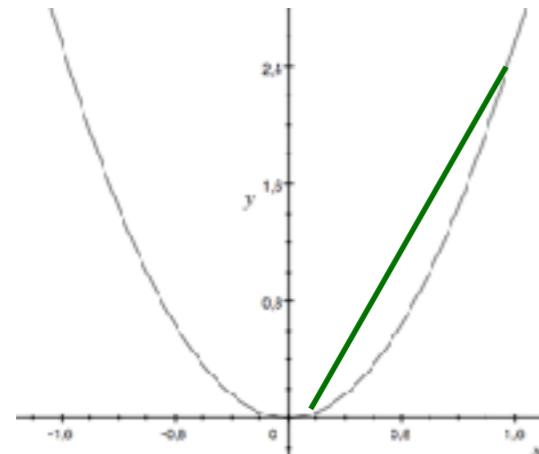


Est-ce que la descente de gradient va converger vers le minimum global de notre fonction de coût ?

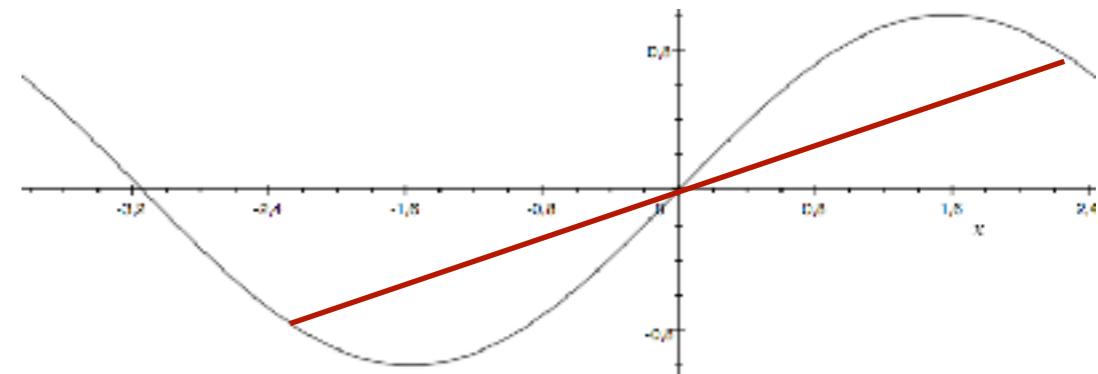


Convexité

Une fonction est convexe si, pour deux positions quelconques de la fonction, la droite liant les deux points est toujours au dessus de la fonction



Fonction convexe



Fonction non convexe

Pour une fonction convexe, on a la garantie que la descente de gradient converge vers le minimum global

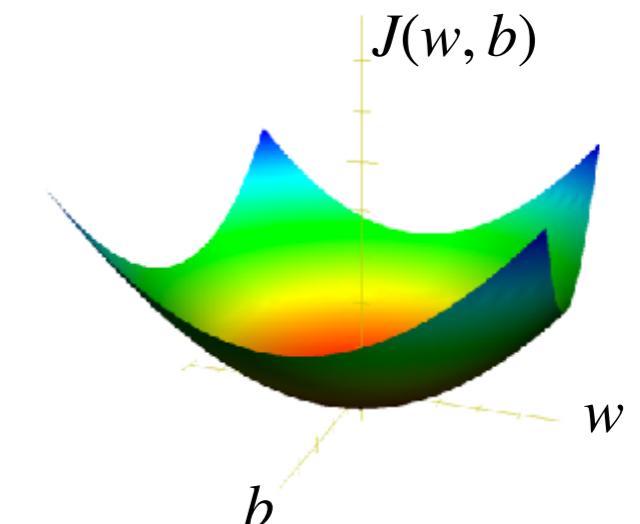
Sous la condition que le learning rate ne soit pas trop grand

Bonne nouvelle

Notre fonction de coût est convexe (parabole)

$$J(w, b) = \frac{1}{2m} \sum_{i=1}^m (\hat{y}^{(i)} - y^{(i)})^2 \text{ avec } \hat{y}^{(i)} = w \cdot x^{(i)} + b$$

Va converger vers le minimum global



Gradient de la fonction de coût



Quel est le gradient de notre fonction de coût ?

Calcul direct sur base de simples règles d'analyse

Calcul du gradient

$$J(w, b) = \frac{1}{2m} \sum_{i=1}^m (w \cdot x^{(i)} + b - y^{(i)})^2 \quad \nabla J(w, b) = \left(\frac{\partial J}{\partial w}, \frac{\partial J}{\partial b} \right)$$

$$\frac{\partial J(w, b)}{\partial w} = \frac{\partial \left(\frac{1}{2m} \sum_{i=1}^m (w \cdot x^{(i)} + b - y^{(i)})^2 \right)}{\partial w}$$

$$\frac{\partial J(w, b)}{\partial w} = \frac{1}{2m} \left(\frac{\partial ((w \cdot x^{(1)} + b - y^{(1)})^2)}{\partial w} + \dots + \frac{\partial ((w \cdot x^{(m)} + b - y^{(m)})^2)}{\partial w} \right)$$



Règles d'analyse

$$\frac{d(ax + bx)}{dx} = \frac{d(ax)}{dx} + \frac{d(bx)}{dx}$$

$$\frac{d(ax)}{dx} = a \frac{d(x)}{dx}$$

Dérivée partielle pour le paramètre w

Application des règles de l'analyse

Dérivée partielle pour une donnée spécifique

$$\begin{aligned} \frac{\partial ((w \cdot x^{(i)} + b - y^{(i)})^2)}{\partial w} &= 2x^{(i)}(w \cdot x^{(i)} + b - y^{(i)}) \\ &= 2x^{(i)}(\hat{y}^{(i)} - y^{(i)}) \end{aligned}$$

Contribution de la donnée i pour le paramètre w

Terme final

$$\begin{aligned} \frac{\partial J(w, b)}{\partial w} &= \frac{1}{2m} \left(2x^{(1)}(\hat{y}^{(1)} - y^{(1)}) + \dots + 2x^{(m)}(\hat{y}^{(m)} - y^{(m)}) \right) \\ &= \frac{1}{m} \sum_{i=1}^m x^{(i)}(\hat{y}^{(i)} - y^{(i)}) \end{aligned}$$

Moyenne de la contribution de chaque donnée

Descente de gradient: récapitulatif

Fonction de coût

$$J(w, b) = \frac{1}{2m} \sum_{i=1}^m (\hat{y}^{(i)} - y^{(i)})^2$$

avec $\hat{y}^{(i)} = w \cdot x^{(i)} + b$

Expression du gradient

$$\frac{\partial J(w, b)}{\partial w} = \frac{1}{m} \sum_{i=1}^m x^{(i)} (\hat{y}^{(i)} - y^{(i)})$$

$$\frac{\partial J(w, b)}{\partial b} = \frac{1}{m} \sum_{i=1}^m (\hat{y}^{(i)} - y^{(i)})$$

Dérivation en exercice

Update de la descente de gradient

$$w = w - \alpha \frac{\partial J(w, b)}{\partial w}$$

$$b = b - \alpha \frac{\partial J(w, b)}{\partial b}$$

Algorithme

```
gradientDescent(D) :  
    initialize w, b  
    repeat until convergence :  
        compute  $\hat{y}^{(i)}$   $\forall i \in \{1, \dots, m\}$   
        compute  $J(w, b)$   
        compute  $\frac{\partial J}{\partial w}, \frac{\partial J}{\partial b}$   
         $w = w - \alpha \frac{\partial J}{\partial w}$   
         $b = b - \alpha \frac{\partial J}{\partial b}$   
    return w, b
```

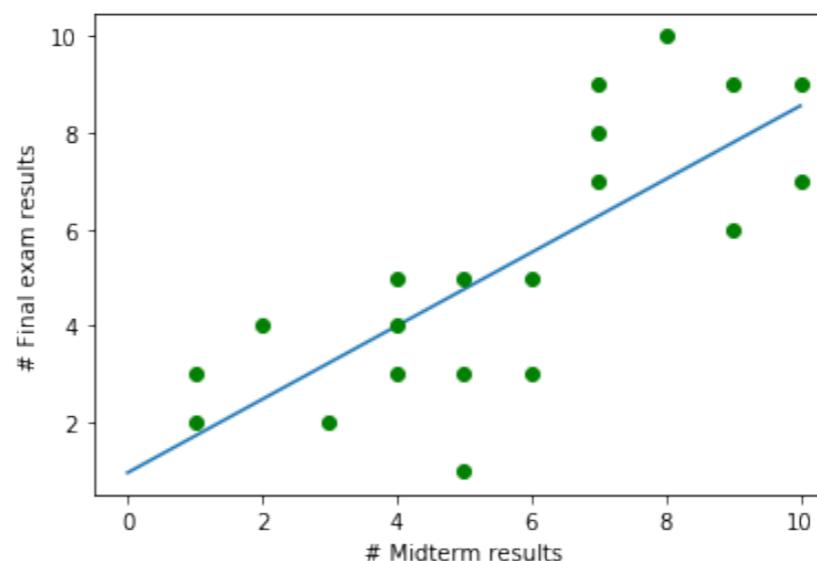
Prédiction du modèle

Calcul de la fonction de coût

Calcul du gradient

Itération de descente de gradient

On retourne les paramètres trouvés



$$\hat{y} = 0.76x + 0.96$$

Table des matières

Apprentissage supervisé

-  1. Motivation et intérêt de l'apprentissage automatique
-  2. Classification des principaux types d'apprentissage
-  3. Définition de l'apprentissage supervisé
-  4. Méthode de la régression linéaire simple
-  5. Apprentissage par descente de gradient
- 6. Méthode de la régression linéaire multiple
- 7. Méthode de la régression logistique
- 8. Graphe de dépendance, *forward pass*, et *backward pass*

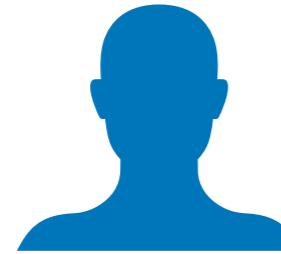
Problèmes abordés

-  1. Prédiction de la note d'un étudiant à l'examen final
- 2. Prédiction d'une réussite ou non à l'examen

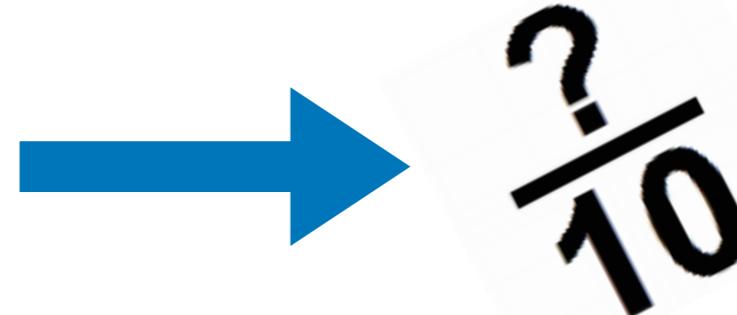
Régression linéaire multiple

Problème de l'étudiant

Au lieu de n'avoir qu'une seule feature (x), on peut en utiliser plusieurs



Résultats de l'intra
heures d'étude
heures de repos



L'idée est d'avoir une représentation plus fidèle de la réalité

Même principe que la régression linéaire, mais est plus difficile à visualiser (espace à n dimensions)

Formalisation étendue

$(x^{(i)}, y^{(i)})$: Donnée d'entraînement

$x^{(i)}$: vecteur des features de la donnée i

$y^{(i)}$: label de la donnée i (true value)

$x_j^{(i)}$: feature j de la donnée i

Approximation par une fonction linéaire

$$\hat{y} = w_1 x_1 + w_2 x_2 + \dots + w_n x_n + b$$

x_1, \dots, x_n : features d'une donnée du problème

b, w_1, \dots, w_n : paramètres devant être appris

?

Combien de paramètres doit-on calculer avec n features ?

Pour n features, on a $n+1$ paramètres

x et w sont représentés par des vecteurs

$$x \in \mathbb{R}^n \quad w \in \mathbb{R}^n \quad b \in \mathbb{R}$$

$$\hat{y} = (w_1 \quad \dots \quad w_n) \times \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} + b = w^T x + b$$

Régression linéaire multiple

Fonction de coût

$$J(w_1, \dots, w_n, b) = \frac{1}{2m} \sum_{i=1}^m (\hat{y}^{(i)} - y^{(i)})^2$$

avec $\hat{y}^{(i)} = \sum_{j=1}^n w_j x_j^{(i)} + b = w^T x + b$

Expression du gradient

$$\frac{\partial J(w_1, \dots, w_n, b)}{\partial w_j} = \frac{1}{m} \sum_{i=1}^m x_j^{(i)} (\hat{y}^{(i)} - y^{(i)}) \quad \forall j \in \{1, \dots, n\}$$

le gradient propre à un paramètre considère la feature qui y est liée

$$\frac{\partial J(w_1, \dots, w_n, b)}{\partial b} = \frac{1}{m} \sum_{i=1}^m (\hat{y}^{(i)} - y^{(i)})$$

Update de la descente de gradient

$$w_j = w_j - \alpha \frac{\partial J(w, b)}{\partial w_j} \quad \forall j \in \{1, \dots, n\}$$

$$b = b - \alpha \frac{\partial J(w, b)}{\partial b}$$

Algorithme d'apprentissage

```
gradientDescent(D) :  
    initialize w, b  
    repeat until convergence :  
        compute  $\hat{y}^{(i)}$   $\forall i \in \{1, \dots, m\}$   
        compute  $J(w_1, \dots, w_n, b)$   
        compute  $\frac{\partial J}{\partial w_1}, \dots, \frac{\partial J}{\partial w_n}, \frac{\partial J}{\partial b}$   
         $w_1 = w_1 - \alpha \frac{\partial J}{\partial w_1}$   
        ...  
         $w_n = w_n - \alpha \frac{\partial J}{\partial w_n}$   
         $b = b - \alpha \frac{\partial J}{\partial b}$   
    return  $w_1, \dots, w_n, b$ 
```

Exactement le même principe que précédemment !

Table des matières

Apprentissage supervisé

-  1. Motivation et intérêt de l'apprentissage automatique
-  2. Classification des principaux types d'apprentissage
-  3. Définition de l'apprentissage supervisé
-  4. Méthode de la régression linéaire simple
-  5. Apprentissage par descente de gradient
-  6. Méthode de la régression linéaire multiple
- 7. Méthode de la régression logistique
- 8. Graphe de dépendance, *forward pass*, et *backward pass*

Problèmes abordés

- 1. Prédiction de la note d'un étudiant à l'examen final
- 2. Prédiction d'une réussite ou non à l'examen

Cas d'étude: prédiction d'une réussite

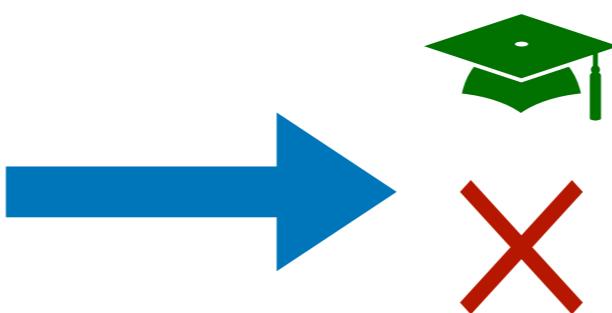
Prédiction d'une réussite

L'étudiant souhaite maintenant savoir s'il va réussir ou non son examen

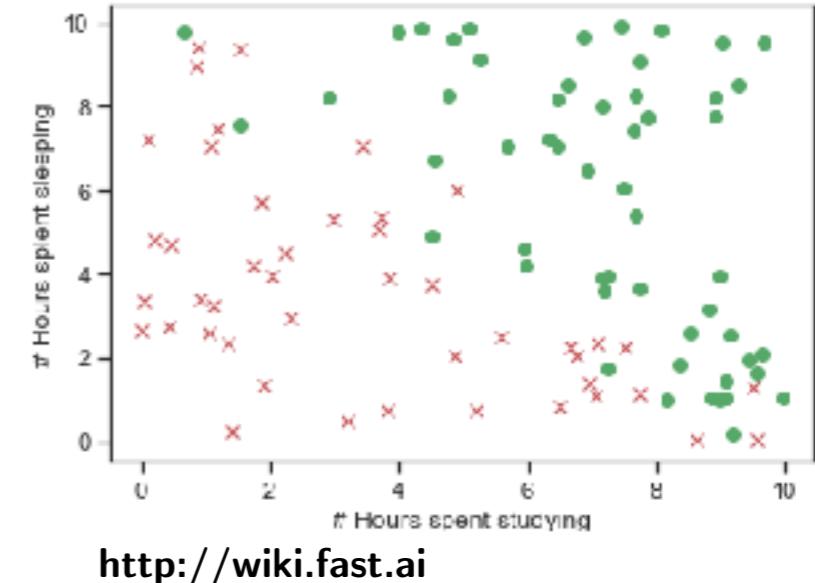
Il se base sur deux critères, le nombre d'heures d'étude, et de repos



heures d'étude
heures de repos



Réussite
Echec



Formalisation du problème

L'étudiant est ainsi représenté par deux features

Le nombre d'heures passées à étudier

$$x_1 \in \mathbb{R}$$

Le nombre d'heures passées à dormir la veille de l'examen

$$x_2 \in \mathbb{R}$$

Le résultat est une valeur binaire

L'étudiant a t-il réussi ou non ?

$$y \in \mathbb{R}$$

On souhaite construire une fonction de prédiction nous donnant une probabilité de succès

$$f: \mathbb{R}^2 \rightarrow [0,1]$$

La prédiction peut ensuite se faire suivant un seuil

$$\begin{aligned} f(x) = \hat{y} \geq 0.5 &\rightarrow \text{Réussite} \\ f(x) = \hat{y} < 0.5 &\rightarrow \text{Echec} \end{aligned}$$

Introduction à la régression logistique



Que pensez-vous de l'utilisation d'une régression linéaire ?

Insuffisance de la régression linéaire

Des valeurs très grandes peuvent être prédites (négatives ou positives)

Dans notre cas, on ne souhaite obtenir qu'une probabilité de succès (comprise entre 0 et 1)

La régression linéaire, employée seule n'est pas suffisante

Régression logistique

Etant donné un input x , on veut construire une fonction

$$\hat{y} = \mathbb{P}(y = 1 | x)$$

La prédiction \hat{y} représente la probabilité de réussite à l'examen

$\hat{y} \approx 0$: probable de rater

$\hat{y} \approx 1$: probable de réussir

Il est ainsi crucial d'obtenir une valeur entre 0 et 1



Fonction sigmoïde

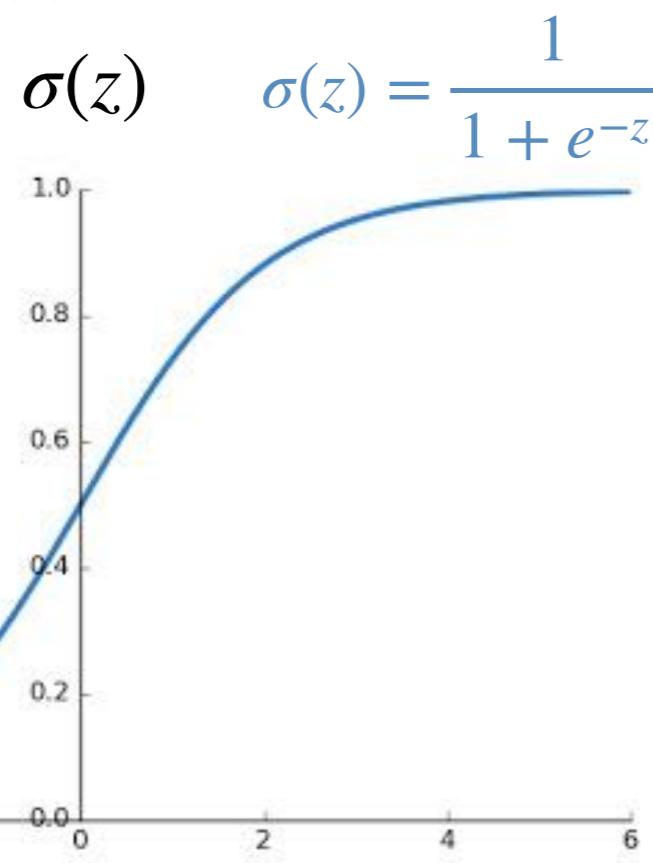
Clef de voute de la régression logistique

Comprime n'importe quelle valeur réelle entre 0 et 1

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

Fonction sigmoïde

Fonction sigmoïde



Comprime une valeur réelle entre 0 et 1

$$\sigma : \mathbb{R} \rightarrow [0,1]$$

Lorsque z est positivement grand, $\sigma(z) = \frac{1}{1+0} = 1$

Confiant sur la réussite

Lorsque z est négativement grand, $\sigma(z) = \frac{1}{1+\infty} = 0$

Confiant sur l'échec

Lorsque z est nul, $\sigma(z) = \frac{1}{1+1} = 0.5$

Uncertain sur le choix

Fonction est différentiable (important pour l'apprentissage)

Régression logistique

- (1) Combinaison linéaire des features
- (2) Sigmoïde appliquée sur cette combinaison

Frontière de décision

$$f(x) = \hat{y} \geq 0.5 \rightarrow \text{Réussite}$$

$$f(x) = \hat{y} < 0.5 \rightarrow \text{Echec}$$

Ou n'importe quel autre seuil

Régression logistique

$$\hat{y} = \sigma(w^T x + b)$$
$$\hat{y} \in [0,1]$$

Fonction d'écart pour la régression logistique



Comment caractériser la qualité d'une fonction de regression logistique ?

Encore une fois, il s'agit de mesurer l'écart entre valeur prédictive, et vraie valeur

Un écart nul indique une prédiction parfaite sur nos données

Fonction d'écart (loss function)

Fonction choisie: *binary cross-entropy loss*

$$L(\hat{y}, y) = - (y \log(\hat{y}) + (1 - y) \log(1 - \hat{y}))$$

Situation où $y = 0$ (valeur d'échec)

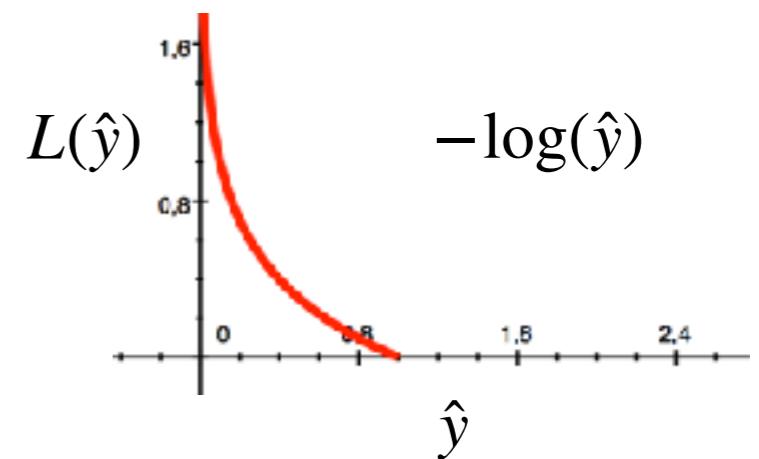
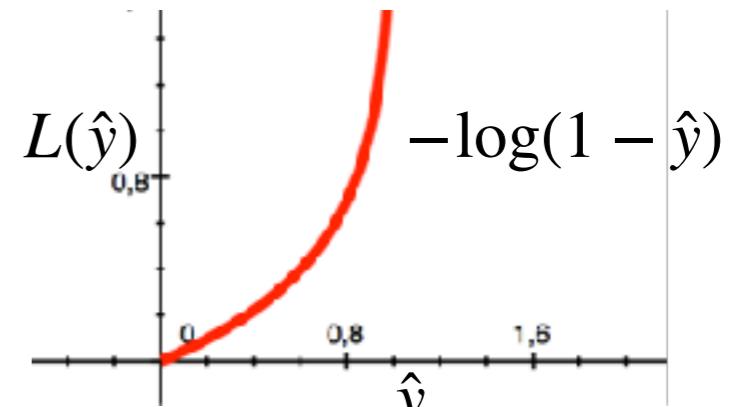
$$L(\hat{y}, y) = - \log(1 - \hat{y})$$

Pousse \hat{y} à être petit ($\hat{y} \rightarrow 0$)

Situation où $y = 1$ (valeur de succès)

$$L(\hat{y}, y) = - \log(\hat{y})$$

Pousse \hat{y} à être grand ($\hat{y} \rightarrow 1$)



Minimiser cette fonction fait en sorte que la valeur de prédiction soit proche de la vraie valeur

Fonction de coût pour la régression logistique

Fonction de coût

Prédiction pour chaque donnée d'entraînement

$$\left\{ (x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \dots, (x^{(m)}, y^{(m)}) \right\} \rightarrow \hat{y}^{(i)} \approx y^{(i)} \quad \forall i \in \{1, \dots, m\}$$
$$\rightarrow \sigma(w^T x^{(i)} + b) \approx y^{(i)}$$

La fonction de coût reprend l'écart moyen des prédictions de toutes les données d'entraînement

$$J(w, b) = \frac{1}{m} \sum_{i=1}^m L(\hat{y}^{(i)}, y^{(i)})$$
$$= -\frac{1}{m} \sum_{i=1}^m (y^{(i)} \log(\hat{y}^{(i)}) + (1 - y^{(i)}) \log(1 - \hat{y}^{(i)}))$$

Avec $\hat{y}^{(i)} = \sigma(w^T x^{(i)} + b)$ et $\sigma(z) = \frac{1}{1 + e^{-z}}$

Cette fonction est différentiable et peut être minimisée par une descente de gradient

Elle est également convexe, on a ainsi une convergence vers un minimum global

Preuve: <https://math.stackexchange.com/questions/1582452/logistic-regression-prove-that-the-cost-function-is-convex>

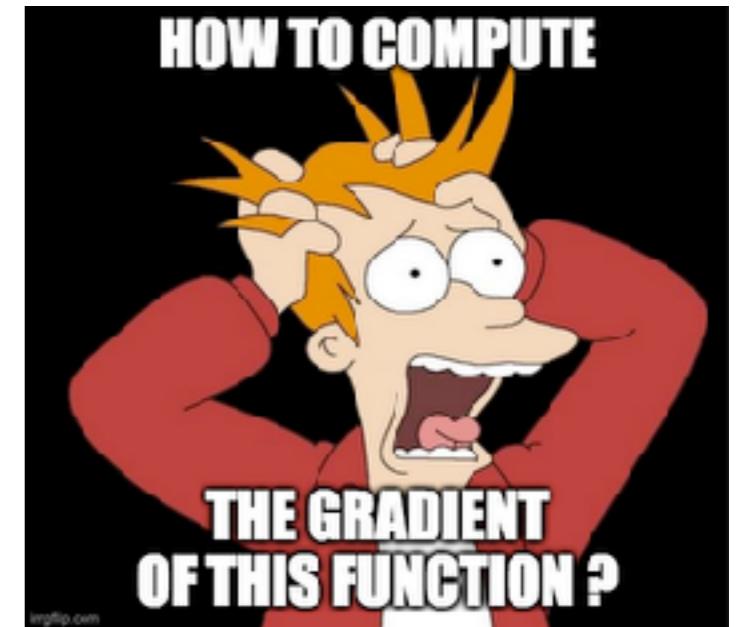
Pour cela, on doit calculer son gradient...

Calcul du gradient

$$\begin{aligned} J(w, b) &= \frac{1}{m} \sum_{i=1}^m L(\hat{y}^{(i)}, y^{(i)}) \\ &= -\frac{1}{m} \sum_{i=1}^m \left(y^{(i)} \log(\hat{y}^{(i)}) + (1 - y^{(i)}) \log(1 - \hat{y}^{(i)}) \right) \end{aligned}$$

Avec $\hat{y}^{(i)} = \sigma(w^T x^{(i)} + b)$ et $\sigma(z) = \frac{1}{1 + e^{-z}}$

Fonction de coût



Malgré l'apparence complexe, cette tâche n'est pas si difficile !

L'astuce est de procéder par décomposition

Méthode de calcul de la dérivée

- (1) Exprimer la fonction en un graphe de dépendances (computation graph)
- (2) Evaluer la fonction avec la valeur actuelle des paramètres (forward pass)
- (3) Utiliser la règle du chaînage (chain rule) pour calculer les dérivées partielles (backward pass)

Exemple

On veut faire une descente de gradient sur cette fonction

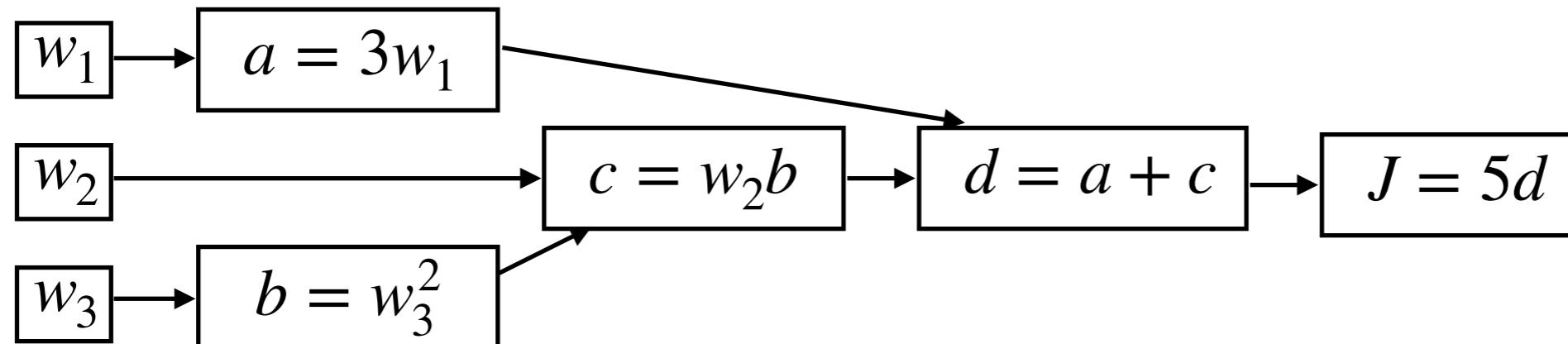
$$J(w_1, w_2, w_3) = 5(3w_1 + w_2 w_3^2)$$

Graphe de dépendances et forward pass

Graphe de dépendances

Division du calcul de la fonction initial en une séquence de sous-calculs

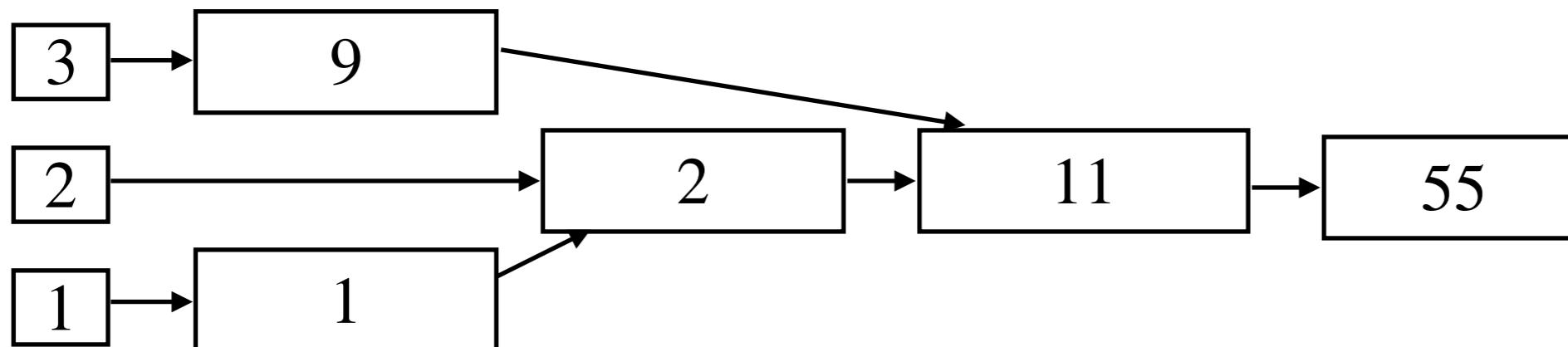
$$J(w_1, w_2, w_3) = 5(3w_1 + w_2 w_3^2)$$



Forward pass

Calculer la valeur de la fonction, étant donné la valeur actuelle de nos paramètres

Supposons: $w_1 = 3, w_2 = 2, w_3 = 1$

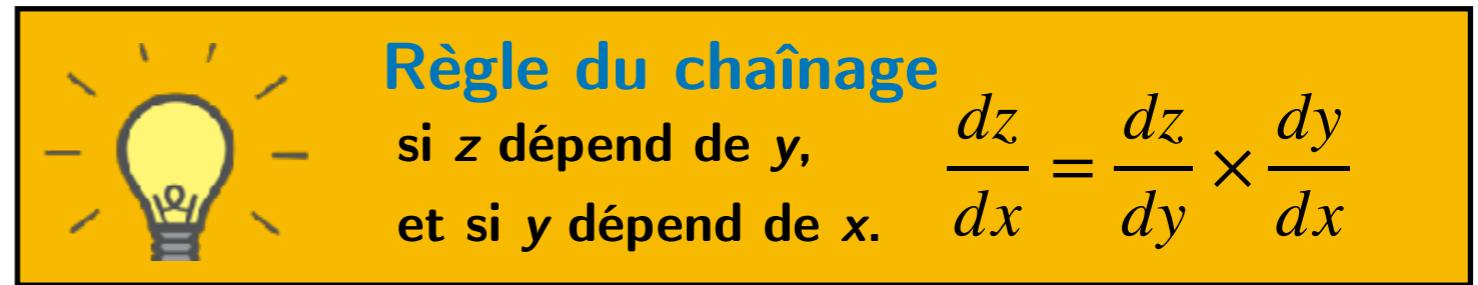
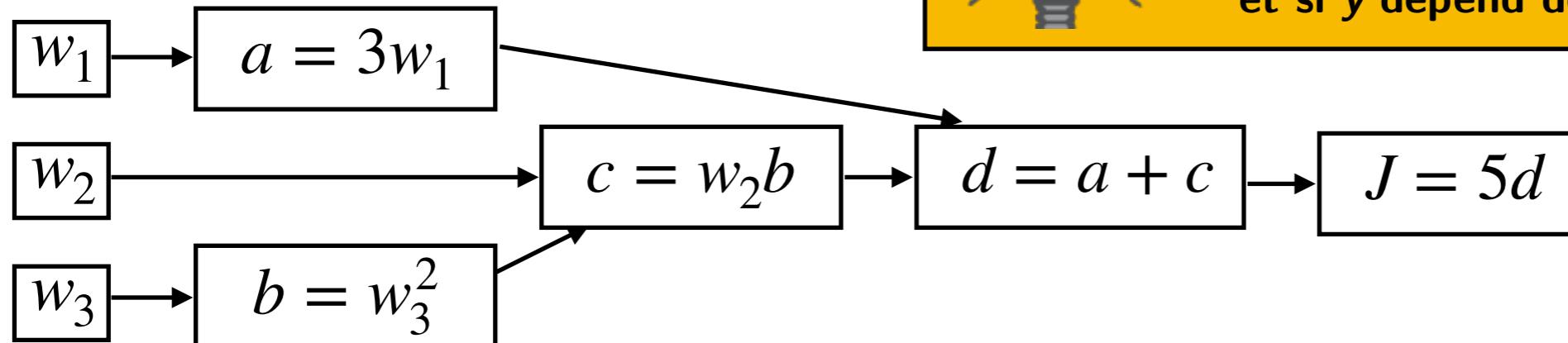


Cette information nous donne le coût actuel de notre fonction de prédiction

Backward pass

Graphe de dépendances

$$J(w_1, w_2, w_3) = 5(3w_1 + w_2 w_3^2)$$



Backward pass

Nous permet de calculer la valeur des dérivées partielles de chaque paramètre

L'idée est de calculer et d'évaluer les dérivées partielles en utilisant la règle d'analyse du chaînage

$$\frac{\partial J}{\partial w_1} = \frac{\partial J}{\partial d} \times \frac{\partial d}{\partial a} \times \frac{\partial a}{\partial w_1} = 5 \times 1 \times 3 = 15$$

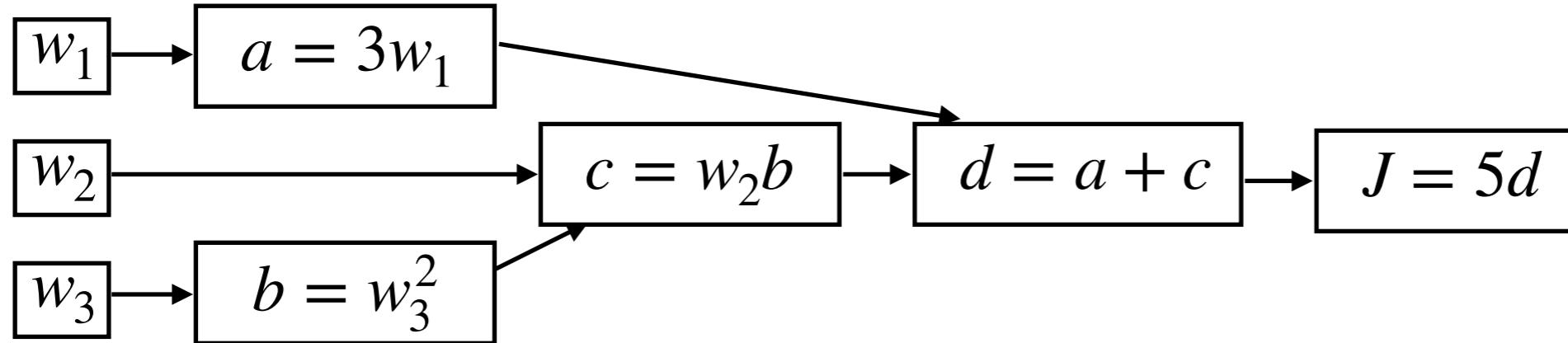
$$\frac{\partial J}{\partial w_2} = \frac{\partial J}{\partial d} \times \frac{\partial d}{\partial c} \times \frac{\partial c}{\partial w_2} = 5 \times 1 \times b = 5b = 5w_3^2 = 5$$

$$\frac{\partial J}{\partial w_3} = \frac{\partial J}{\partial d} \times \frac{\partial d}{\partial c} \times \frac{\partial c}{\partial b} \times \frac{\partial b}{\partial w_3} = 5 \times 1 \times w_2 \times 2w_3 = 10w_2w_3 = 20$$

Descente de gradient

Graphe de dépendances

$$J(w_1, w_2, w_3) = 5(3w_1 + w_2 w_3^2)$$



Valeurs de nos dérivées partielles

Avec $w_1 = 3, w_2 = 2, w_3 = 1$

$$\frac{\partial J}{\partial w_1} = 15 \quad \frac{\partial J}{\partial w_2} = 5 \quad \frac{\partial J}{\partial w_3} = 20$$

Descente de gradient

Supposons un learning rate de $\alpha = 0.1$

$$w_1 = w_1 - \alpha \times 15 = 3 - 0.1 \times 15 = 1.5$$

$$w_2 = w_2 - \alpha \times 5 = 2 - 0.1 \times 5 = 1.5$$

$$w_3 = w_3 - \alpha \times 20 = 1 - 0.1 \times 20 = 0.8$$

Nouvelles valeurs: $w_1 = 1.5, w_2 = 1.5, w_3 = 0.8$

Et on peut refaire une nouvelle évaluation avec ces valeurs

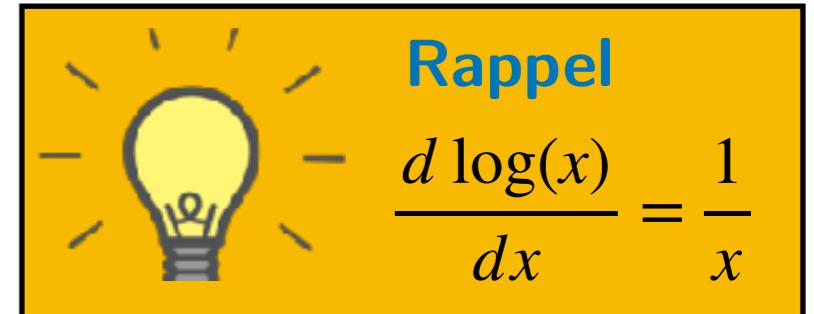


Graphe de dépendances sur notre fonction d'écart

Fonction d'écart (loss function)

$$L(\hat{y}, y) = - (y \log(\hat{y}) + (1 - y) \log(1 - \hat{y}))$$

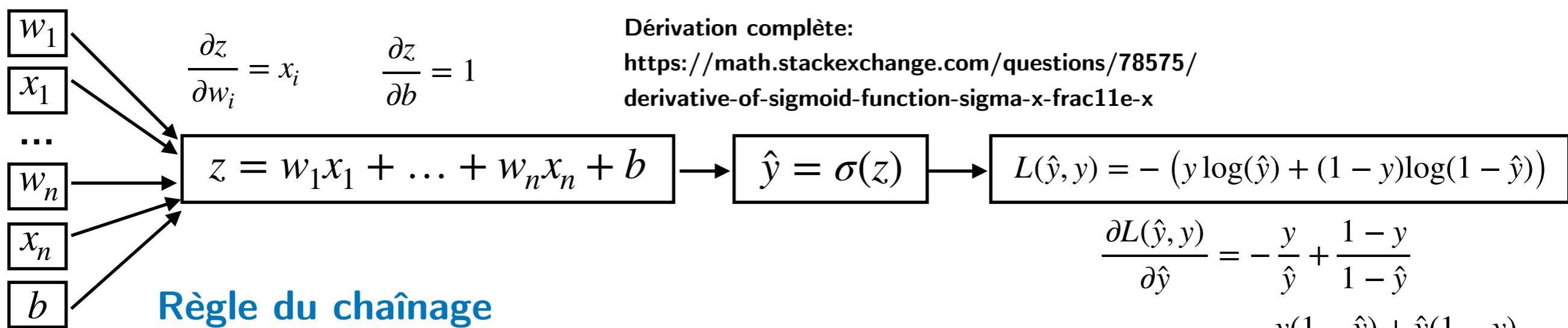
Avec $\hat{y} = \sigma(w^T x + b)$ et $\sigma(z) = \frac{1}{1 + e^{-z}}$



Dérivées partielles à calculer

$$\frac{\partial L}{\partial w_1}, \dots, \frac{\partial L}{\partial w_n}, \frac{\partial L}{\partial b}$$

Graphe de dépendances



$$\frac{\partial L}{\partial w_i} = \frac{\partial L}{\partial \hat{y}} \times \frac{\partial \hat{y}}{\partial z} \times \frac{\partial z}{\partial w_i} = \frac{\hat{y} - y}{\hat{y}(1 - \hat{y})} \times \hat{y}(1 - \hat{y}) \times x_i = x_i(\hat{y} - y)$$

$$\frac{\partial L}{\partial b} = \frac{\partial L}{\partial \hat{y}} \times \frac{\partial \hat{y}}{\partial z} \times \frac{\partial z}{\partial b} = \frac{\hat{y} - y}{\hat{y}(1 - \hat{y})} \times \hat{y}(1 - \hat{y}) \times 1 = \hat{y} - y$$

$$\begin{aligned}\frac{\partial L(\hat{y}, y)}{\partial \hat{y}} &= -\frac{y}{\hat{y}} + \frac{1 - y}{1 - \hat{y}} \\ &= \frac{-y(1 - \hat{y}) + \hat{y}(1 - y)}{\hat{y}(1 - \hat{y})} \\ &= \frac{-y + \hat{y}y + \hat{y} - \hat{y}y}{\hat{y}(1 - \hat{y})} \\ &= \frac{\hat{y} - y}{\hat{y}(1 - \hat{y})}\end{aligned}$$

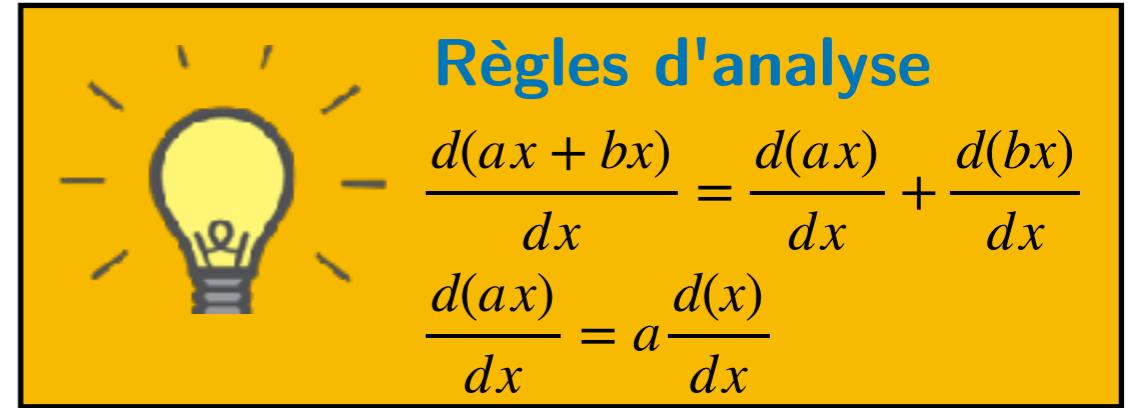
On a nos dérivées !

Minimisation de la fonction de coût

Fonction de coût

$$J(w, b) = \frac{1}{m} \sum_{i=1}^m L(\hat{y}^{(i)}, y^{(i)})$$

$$J(w, b) = \frac{1}{m} \left(L(\hat{y}^{(1)}, y^{(1)}) + \dots + L(\hat{y}^{(m)}, y^{(m)}) \right)$$



Même expression que pour la régression linéaire, sauf que la fonction d'écart est différentes

Gradient de la fonction

$$\frac{\partial J(w, b)}{\partial w_j} = \frac{1}{m} \left(\frac{\partial L(\hat{y}^{(1)}, y^{(1)})}{\partial w_j} + \dots + \frac{\partial L(\hat{y}^{(m)}, y^{(m)})}{\partial w_j} \right) \quad \forall j \in \{1, \dots, n\}$$

Avec

$$\frac{\partial L(\hat{y}^{(i)}, y^{(i)})}{\partial w_j} = x_j^{(i)} (\hat{y}^{(i)} - y^{(i)})$$

$$\frac{\partial L(\hat{y}^{(i)}, y^{(i)})}{\partial b} = (\hat{y}^{(i)} - y^{(i)})$$

Intuitivement, on additionne et on prend la contribution moyenne de chaque donnée d'entraînement

Toutes les briques assemblées

Pseudo-code: problème de l'étudiant

```
1 n_step = 1000
2 alpha = 0.1
3 w1, w2, b = 0, 0, 0
4
5 for j in range(n_step):
6     dw1, dw2, db, J = 0, 0, 0, 0
7     # Considering all the training set
8     for i in range(m):
9
10         # Forward pass
11         z_i = w1 * x1[i] + w2 * x2[i] + b
12         a_i = sigmoid(z_i)
13         J = J + loss(a_i,y[i])
14
15         # Backward pass
16         dz_i = a_i - y[i]
17         dw1 = dw1 + x1[i] * dz_i
18         dw2 = dw2 + x2[i] * dz_i
19         db = db + dz_i
20
21     J = J / m
22     dw1 = dw1 / m
23     dw2 = dw2 / m
24     db = db / m
25     # Gradient descent step
26     w1 = w1 - alpha * dw1
27     w2 = w2 - alpha * dw2
28     b = b - alpha * db
```

Forward pass

$$J(w, b) = \frac{1}{m} \sum_{i=1}^m L(\hat{y}^{(i)}, y^{(i)}) \quad \text{With } \hat{y}^{(i)} = \sigma(w^T x^{(i)} + b)$$

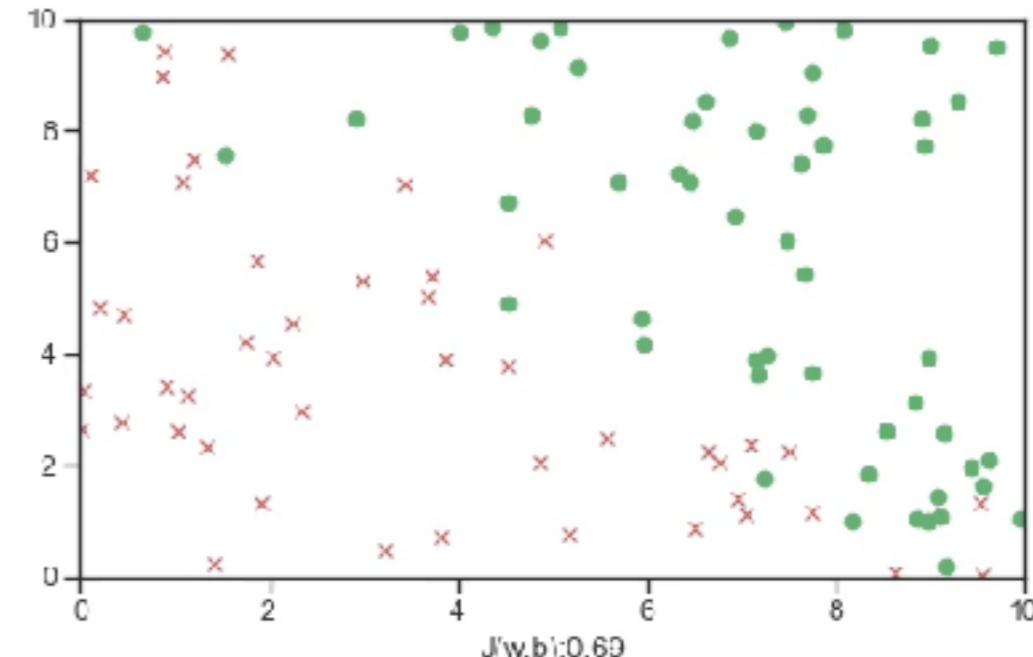
Backward pass

$$\frac{\partial J(w, b)}{\partial b} = \frac{1}{m} \left(\frac{\partial L(\hat{y}^{(1)}, y^{(1)})}{\partial b} + \dots + \frac{\partial L(\hat{y}^{(m)}, y^{(m)})}{\partial b} \right)$$

$$\frac{\partial J(w, b)}{\partial w_1} = \frac{1}{m} \left(\frac{\partial L(\hat{y}^{(1)}, y^{(1)})}{\partial w_1} + \dots + \frac{\partial L(\hat{y}^{(m)}, y^{(m)})}{\partial w_1} \right)$$

$$\frac{\partial J(w, b)}{\partial w_2} = \frac{1}{m} \left(\frac{\partial L(\hat{y}^{(1)}, y^{(1)})}{\partial w_2} + \dots + \frac{\partial L(\hat{y}^{(m)}, y^{(m)})}{\partial w_2} \right)$$

Avec $\frac{\partial L(\hat{y}^{(i)}, y^{(i)})}{\partial w_i} = x_i^{(i)} (\hat{y}^{(i)} - y^{(i)})$ et $\frac{\partial L(\hat{y}^{(i)}, y^{(i)})}{\partial b} = (\hat{y}^{(i)} - y^{(i)})$



Synthèse des notions vues

Apprentissage automatique



"A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E"

<http://www.cs.cmu.edu/afs/cs.cmu.edu/user/mitchell/ftp/mlbook.html>

Tom Mitchell (1997)

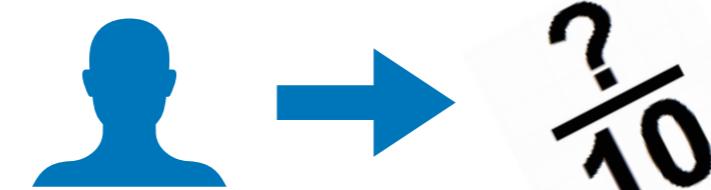
Construire une fonction capable d'effectuer une prédiction, de qualité évaluable, sur base de données historiques, pour réaliser une certaine tâche

Apprentissage supervisé

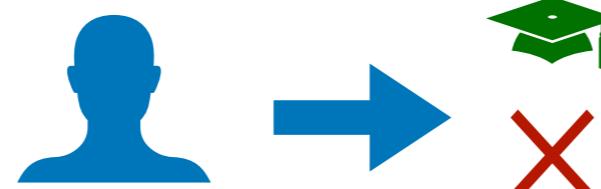
Minimisation d'un coût entre une prédiction et les données connues

Plusieurs hypothèses possibles

Régression linéaire simple



Régression linéaire multiple

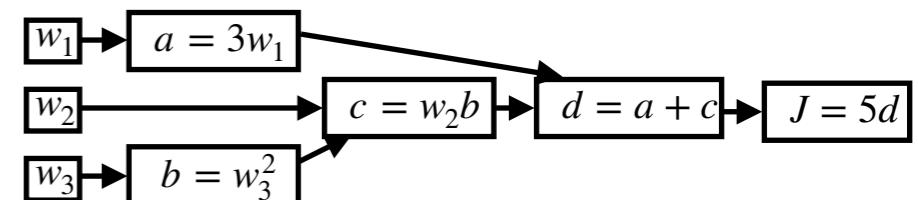
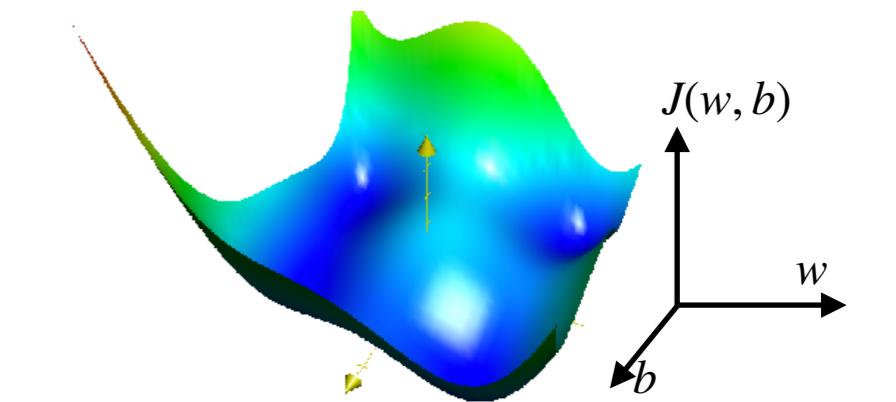


Réussite

Régression logistique



Echec

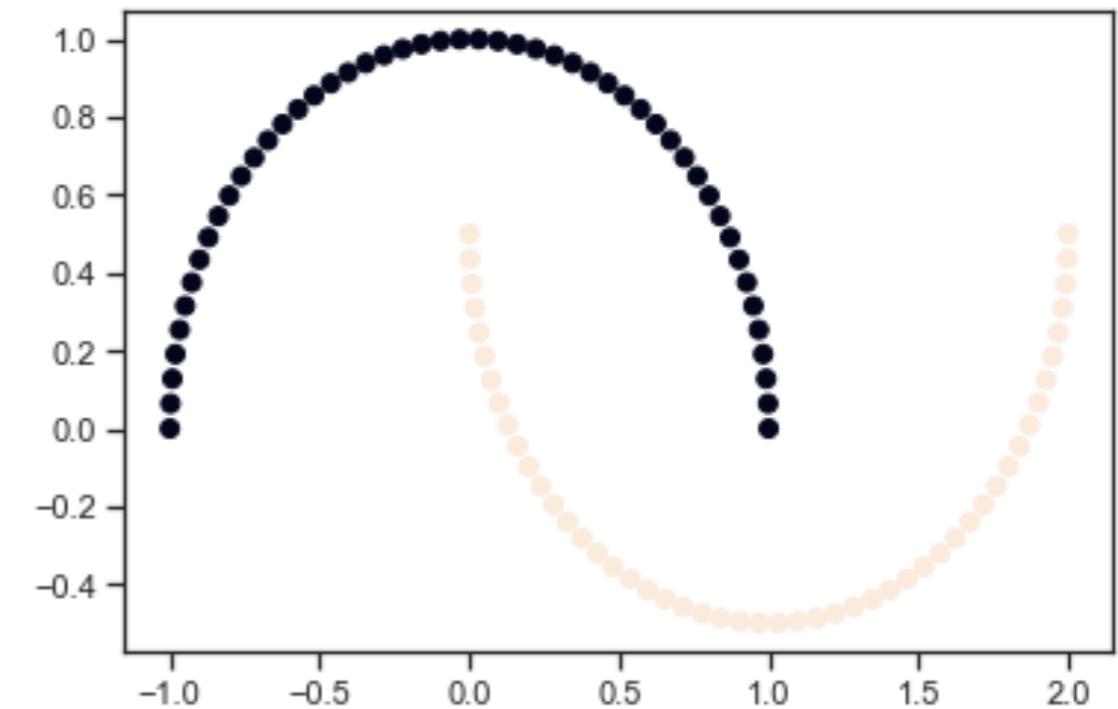
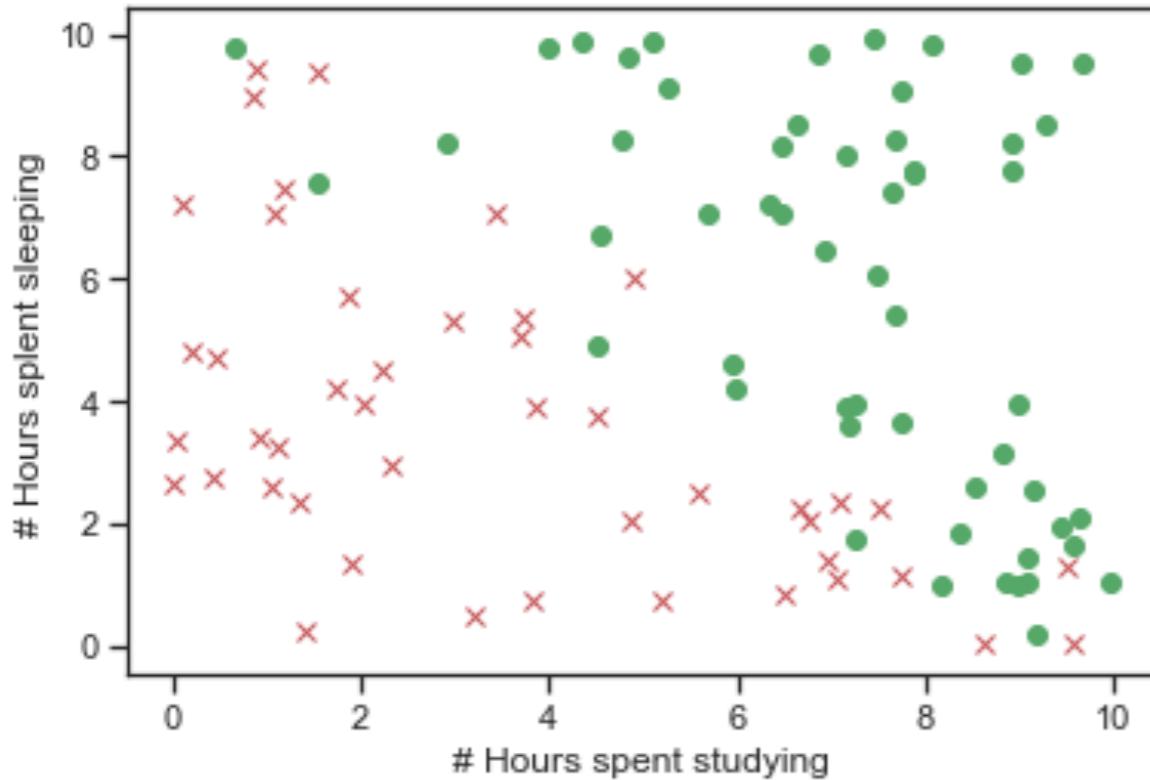


Apprentissage par descente de gradient via un graphe de dépendance

Limitation de nos méthodes de regression



Quelles sont les limitations de nos méthodes de régression actuelles ?



Hypothèse de séparation linéaire

Notre prédition consiste en une combinaison linéaire de nos variables (éventuellement sigmoidisée)

N'est efficace que si une tendance grossièrement linéaire entre les données peut-être établie

Figure de droite: séparation linéaire acceptable

Figure de gauche: données non linéairement séparables

Il nous faut une fonction de prédition plus expressive !

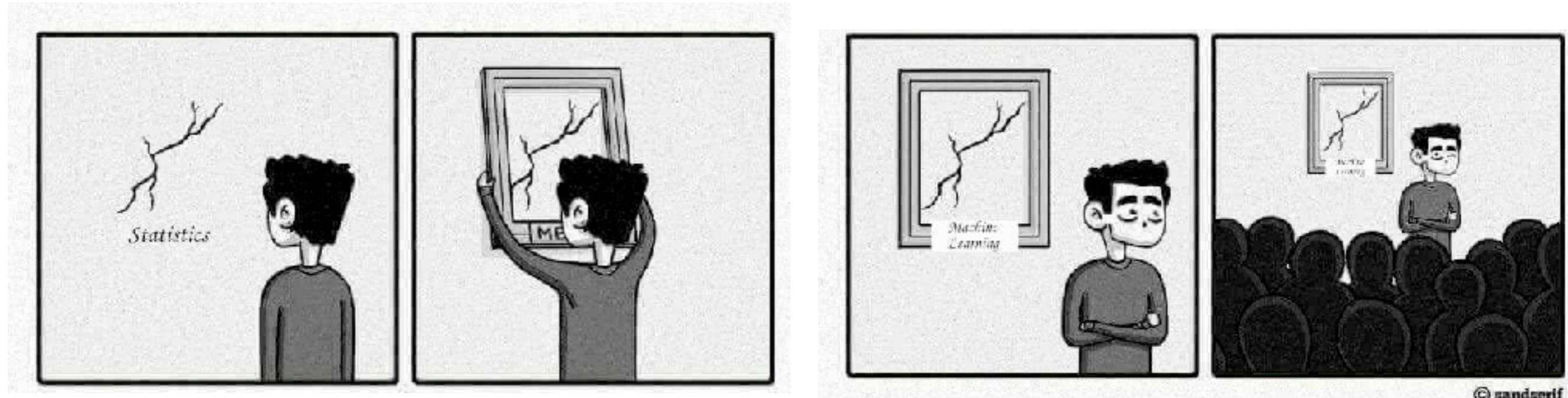
Exemples de questions d'examen

Théorie

1. Donner et expliquer une fonction d'écart ou de coût pour une régression donnée
2. Décrire la fonction sigmoide, ses intérêts, et donner un cas d'utilisation
3. Expliquer les différences entre features et paramètres
4. Expliquer le principe de la descente de gradient

Pratique

1. Effectuer une étape de descente de gradient pour une fonction simple tout en détaillant les différentes étapes (forward et backward pass)



© sandserif

INF8215 - Intelligence artificielle

Méthodes et algorithmes

Apprentissage supervisé: FIN



**POLYTECHNIQUE
MONTRÉAL**

Quentin Cappart