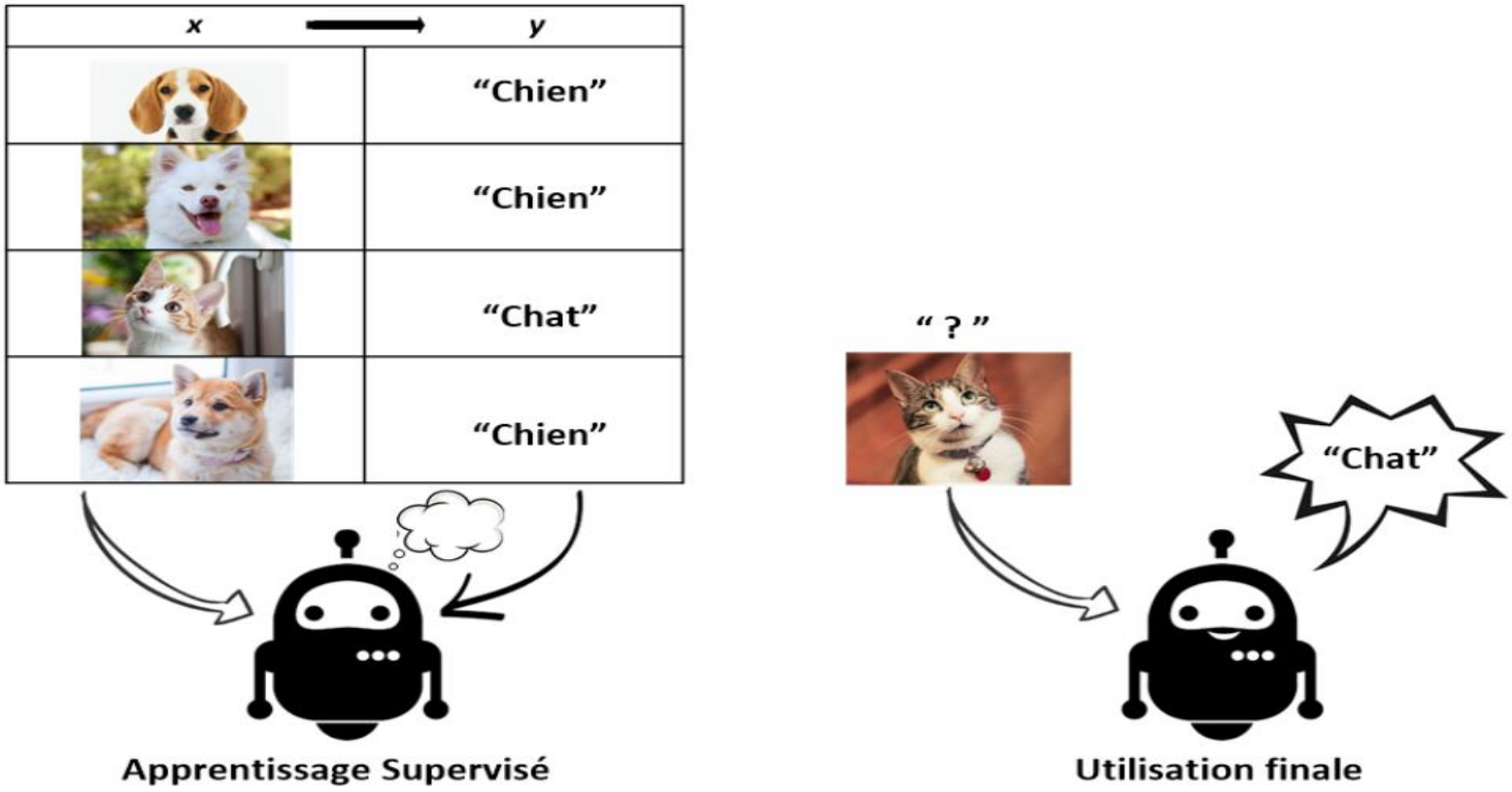
2022/2023

Apprentissage Supervisé

Réalisé par: Naïma Daghfous

Apprentissage Supervisé

- Apprentissage automatique = apprendre un modèle formel à partir de données observées



La machine apprend à partir de milliers d'exemples x, y

Apprentissage Supervisé

- L'apprentissage supervisé fonctionne en 4 étapes :

- Importer un **Dataset** qui contient nos exemples
- Développer un **Modèle** aux paramètres aléatoires
- Développer une **Fonction Coût** qui mesure les erreurs entre le modèle et le Dataset
- Développer un Algorithme d'apprentissage pour trouver les paramètres du modèle qui **minimisent la Fonction Coût**

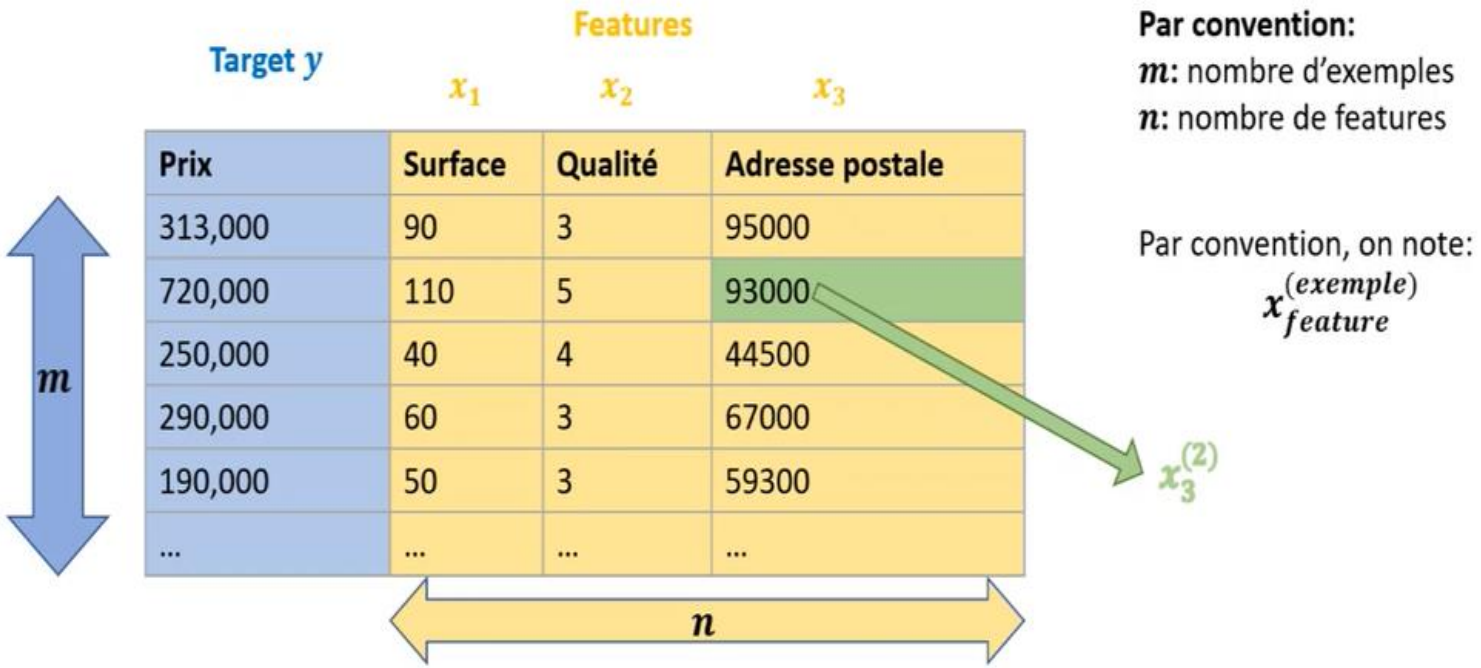
Dataset

- **Collecte des données** et leur labellisation.
- **Nettoyage des données** (Valeurs manquantes, redondance, variables inutiles…).
- **Prétraitement des données** (Identification des variables explicatives et de la sortie cible, séparation en données d'entraînement et données de validation, normalisation des données…)

Exemple deDataset

- Les colonnes sont variables d'entrée, attributs ou caractéristiques.
- les variables résultats ou les cibles. (target)
- Une ligne du tableau est appelée un exemple d'entrainement ou instance.

Exemple de Dataset sur des appartements



Représentation du Dataset

$Dataset\ (x, y)$

y	x_1	x_2	x_3	...	x_n
$y^{(1)}$	$x_1^{(1)}$	$x_2^{(1)}$	$x_3^{(1)}$	...	$x_n^{(1)}$
$y^{(2)}$	$x_1^{(2)}$	$x_2^{(2)}$	$x_3^{(2)}$	...	$x_n^{(2)}$
$y^{(3)}$	$x_1^{(3)}$	$x_2^{(3)}$	$x_3^{(3)}$	...	$x_n^{(3)}$
...	...	...	...	...	...
$y^{(m)}$	$x_1^{(m)}$	$x_2^{(m)}$	$x_3^{(m)}$	...	$x_n^{(m)}$

$vecteur\ target\ y \in \mathbb{R}^{m \times 1}$

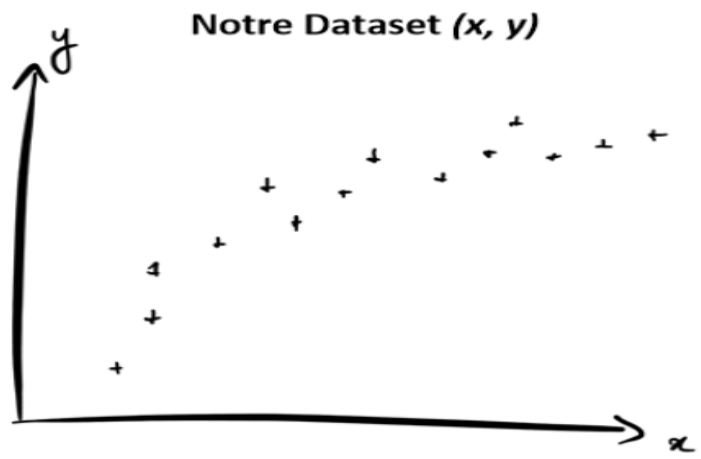
$matrice\ features\ X \in \mathbb{R}^{m \times n}$

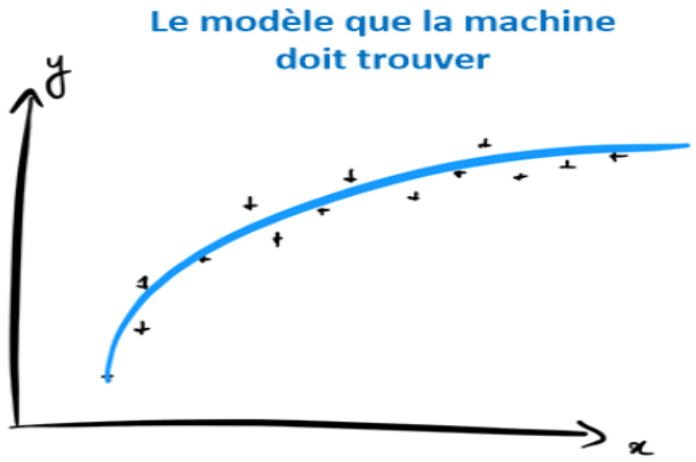
$$y = \begin{pmatrix} y^{(1)} \\ y^{(2)} \\ \dots \\ y^{(m)} \end{pmatrix}$$

$$X = \begin{pmatrix} x_1^{(1)} & \cdots & x_n^{(1)} \\ \vdots & \ddots & \vdots \\ x_1^{(m)} & \cdots & x_n^{(m)} \end{pmatrix}$$

Le Modèle et ses paramètres

- Un modèle est une **représentation** simplifiée de la réalité, que l'on peut utiliser pour **prédire** ce qui se passerait dans certaines conditions. Ça peut être un dessin, une équation physique, une fonction mathématique, une courbe… bref, n'importe quelle représentation.
- Si par exemple le Dataset nous donne le nuage de point suivant, alors la machine devra trouver le modèle qui rentre le mieux dans ce nuage de point.

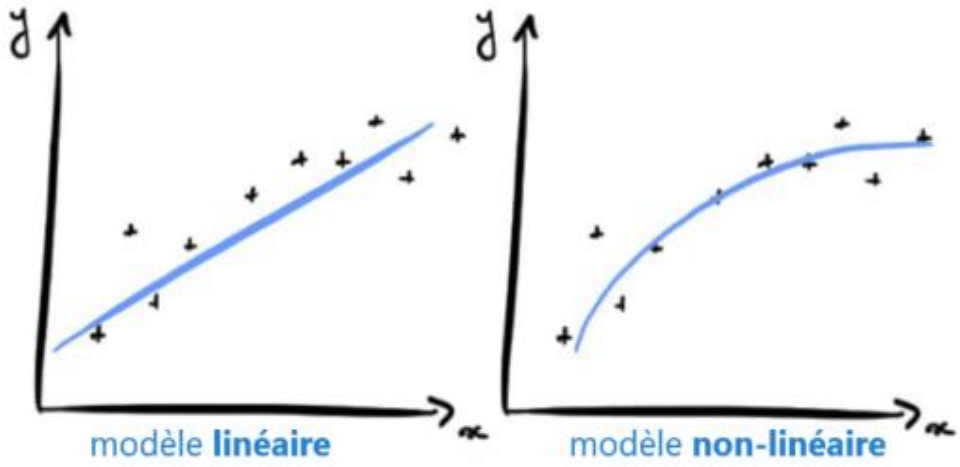




Le Modèle et ses paramètres

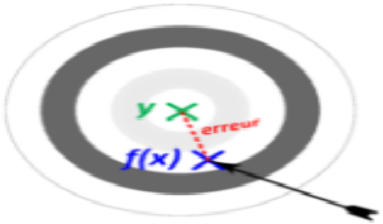
- **Cependant, ce n'est pas à la machine de faire tout le travail !** c'est à nous de choisir le type de modèle (c'est-à-dire la fonction mathématique) et c'est à la machine de trouver les **coefficients** donnant les meilleurs résultats.
- Ces coefficients sont les **paramètres** du modèle.
- Par exemple, on peut choisir de développer :
  - un modèle **linéaire** $f(x) = ax + b$
  - un modèle **non-linéaire** $f(x) = ax^2 + bx + c$

et on laisse la machine trouver la valeur des paramètres (a, b) ou (a, b , c) qui donnent les meilleurs résultats.



La fonction cout

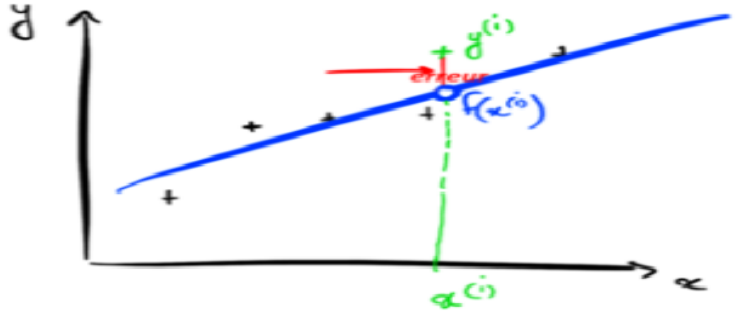
- Pour que la machine trouve le meilleur modèle, il faut qu'elle puisse **mesurer la performance** d'un modèle donné

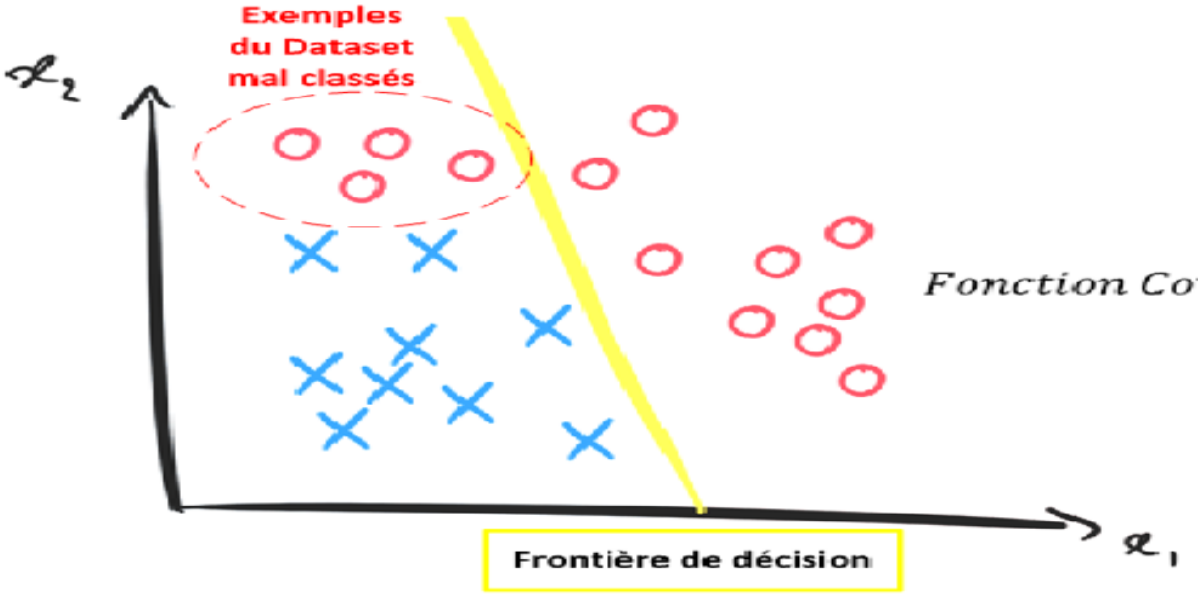


Distance entre **flèche** et **cible**

Équivalent à

Erreur entre **modèle** et **Dataset**





$$Fonction\ Coût = \frac{\#\ Erreurs}{\#\ Exemples} = \frac{4}{22}$$

L'Algorithme d'apprentissage

- La machine cherche les **paramètres** de modèle qui **minimisent** la **Fonction Coût**. C'est ça qu'on appelle l'apprentissage. Cette phrase est **très importante**. C'est l'essentiel de ce qu'il faut comprendre en Machine Learning.

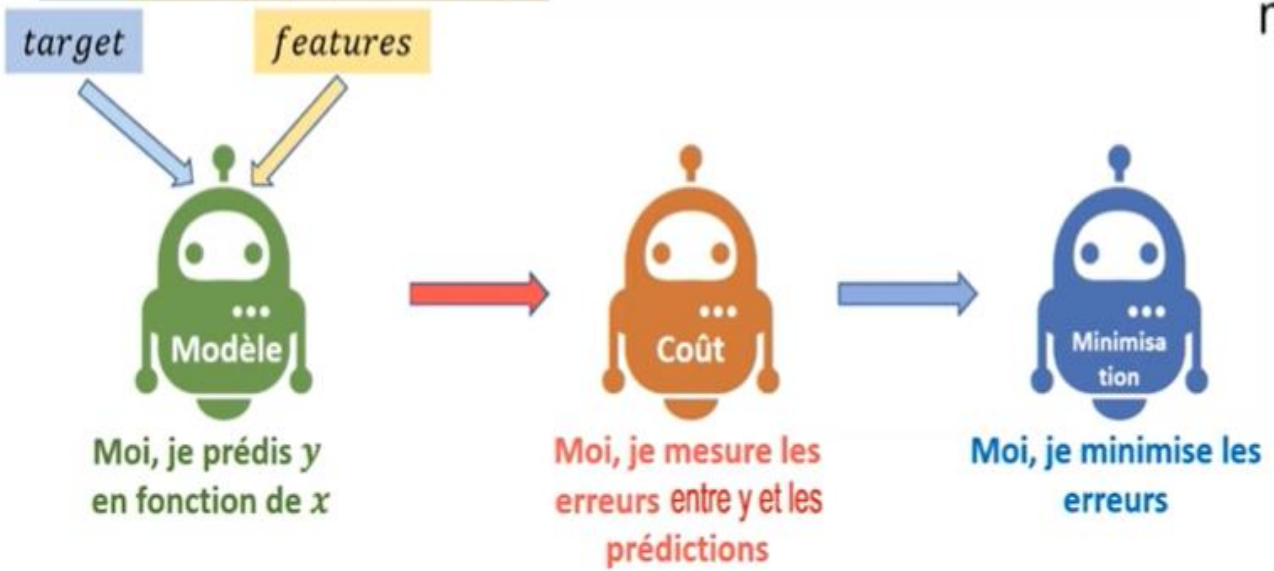
- Pour trouver les paramètres qui minimisent la fonction Coût, il existe un plusieurs stratégies.
- Une stratégie, **très populaire** en Machine Learning, est de considérer la Fonction Coût comme une fonction **convexe**, c'est-à-dire une fonction qui n'a qu'un seul minimum, et de chercher ce minimum avec un algorithme de minimisation appelé **Gradient Descent**.

Schéma de l'Apprentissage Supervisé

$Dataset\ (x, y)$

y	x_1	x_2	x_3	...	x_n
$y^{(1)}$	$x_1^{(1)}$	$x_2^{(1)}$	$x_3^{(1)}$	...	$x_n^{(1)}$
$y^{(2)}$	$x_1^{(2)}$	$x_2^{(2)}$	$x_3^{(2)}$	...	$x_n^{(2)}$
$y^{(3)}$	$x_1^{(3)}$	$x_2^{(3)}$	$x_3^{(3)}$	...	$x_n^{(3)}$
...	...	...	...	...	...
$y^{(m)}$	$x_1^{(m)}$	$x_2^{(m)}$	$x_3^{(m)}$	...	$x_n^{(m)}$

1. Dataset
   y : Target
   x : features
2. Modèle
   paramètres
3. Fonction Coût
4. Algorithme de minimisation



Régression & Classification

L'apprentissage supervisé

Je sais ce que je cherche à prédire et j'ai déjà des données en guise d'exemple à fournir à ma machine

SUPERVISÉ

La régression

Je cherche à prédire une valeur numérique, un chiffre, un nombre ?

Exemples : ventes, marge, stock, température, pression, taille, poids, quantité…

La classification

Je cherche à prédire une catégorie, une dimension, une classe ?

Exemples : oui / non, vrai / faux, homme / femme, segments, espèces d'animaux…

Avantages & Inconvénients

- Quels sont les avantages de l'apprentissage supervisé?

Plusieurs problématiques peuvent être traitées à l'aide de l'apprentissage supervisé.

Entraînement facile et efficace des différents modèles grâce à des données déjà étiquetées.

Les labels permettent de valider le modèle en le testant sur des données étiquetées et en **comparant les résultats prédits et les sorties réelles**.

- Pourquoi l'apprentissage supervisé est de moins en moins populaire?

Difficultés pour étiqueter les données, surtout quand elles sont en grande quantité.

Problème de **sur-apprentissage** si le modèle rencontre des données anormales (problème très fréquent si le jeu de données d'entraînement est de petite taille).