

Introduction à l'apprentissage artificiel -Machine Learning-



Dr. Redouane Ezzahir
Professeur Habilité
Département Informatique
ENSA-Agadir, Université Ibn Zohr
r.ezzahir@uiz.ac.ma

Objectifs

1. Définir le Machine Learning (ML).
2. Identifier si un problème relève ou non du ML.
3. Donner des exemples de cas concrets relevant de grandes classes de problèmes de ML.

Qu'est ce que l'apprentissage Artificiel?

Apprentissage Artificiel

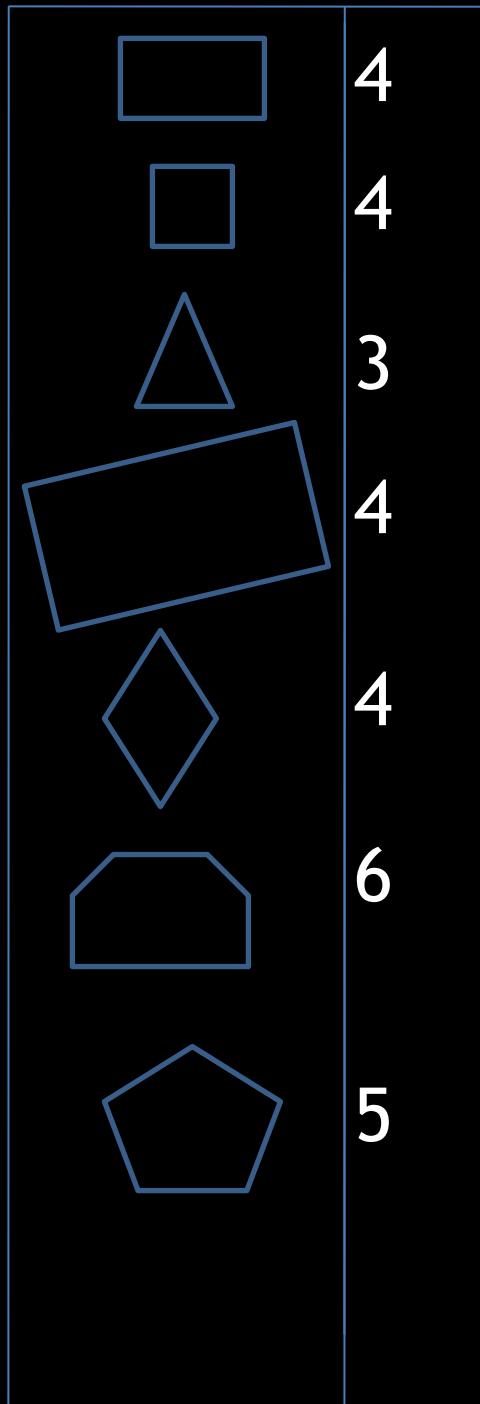
=

Machine Learning (ML)

=

Apprentissage Automatique

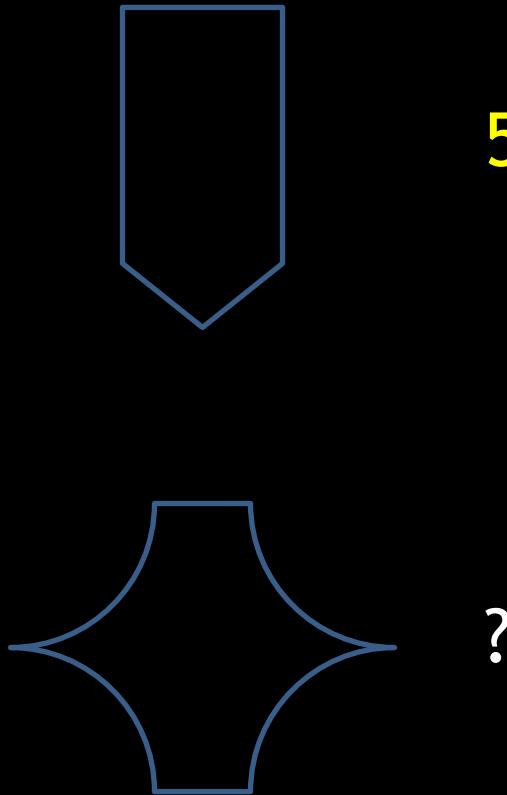
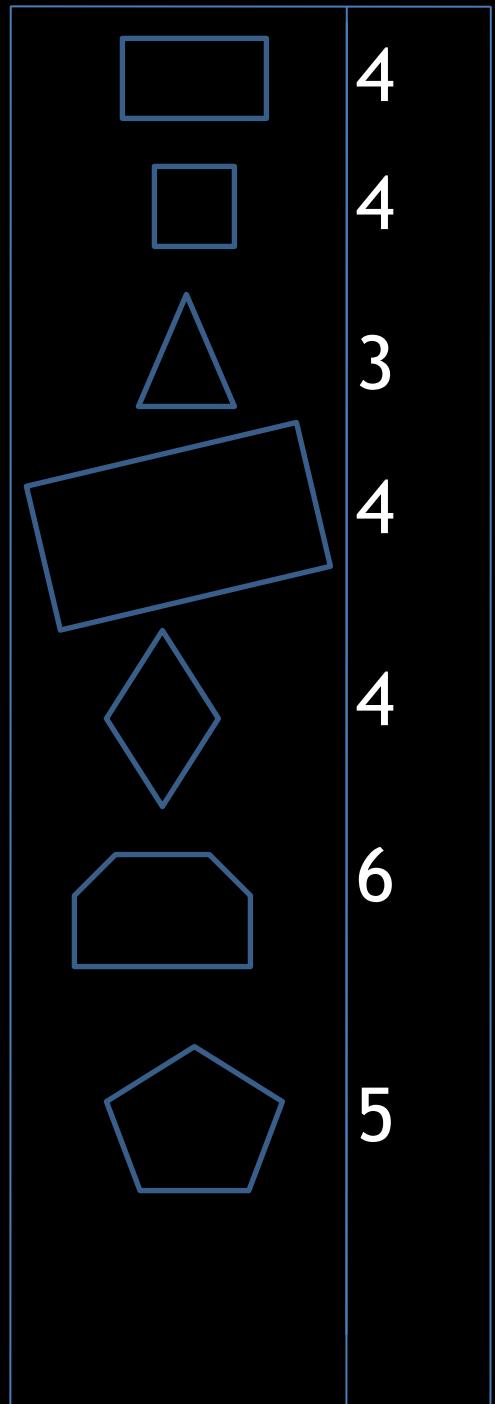
Apprendre par expériences



?



Apprendre par expériences



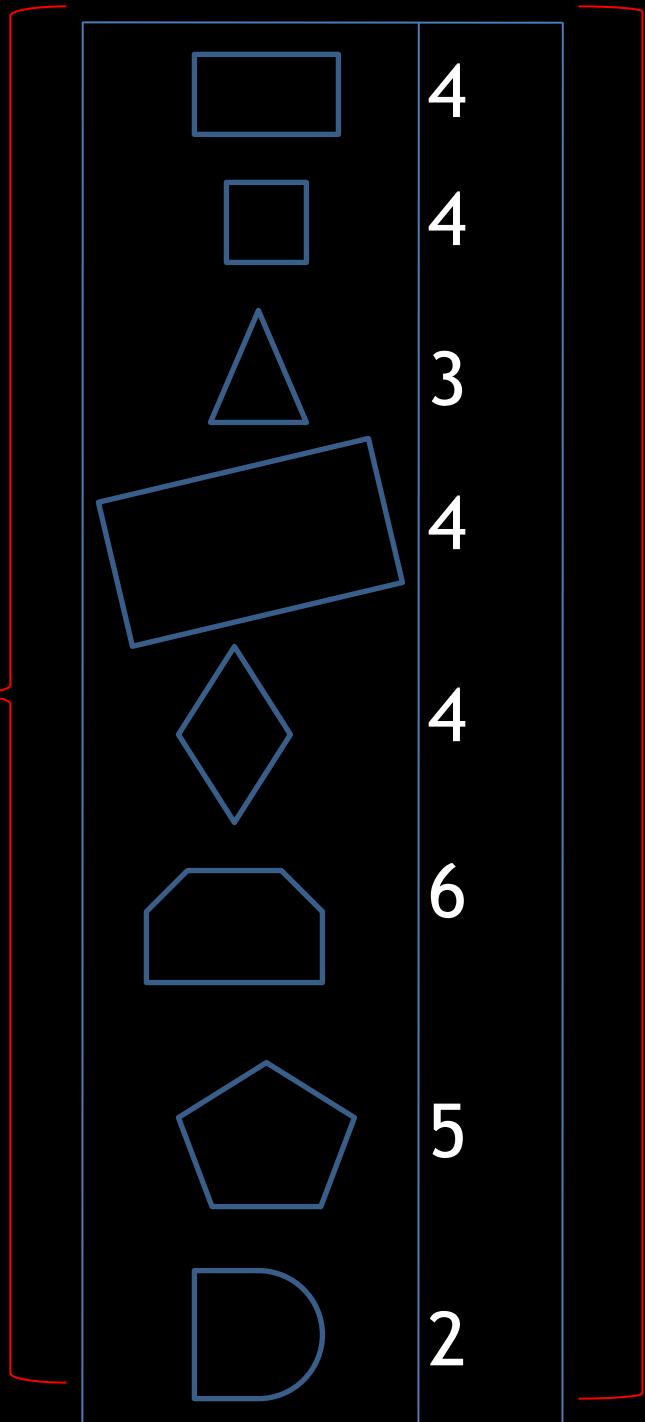
5

?



Apprendre par l'expérience

Ensemble de données d'entraînement (training set)



5

6



Définitions

L'apprentissage est une modification d'un comportement sur la base d'une expérience.

Webster 1984

On parle d'apprentissage artificiel, ou machine Learning, quand le programme a la capacité d'apprendre sans être explicitement programmé en fonction de la tâche.

Arthur Samuel (1959).

On dit qu'un programme informatique «apprend» de l'expérience E pour certaine classes de tâche T et une mesure de performance P, si ses performances en tâche de T, mesurées par P, sont améliorées avec l'expérience E.

Tom Mitchell (1997)

Exemples

T: Classer les e-mails comme spam ou légitime.

P: pourcentage de messages électroniques correctement classés.

E: base de données d'emails, certains avec des étiquettes données par l'homme



T : Recommander des livres

P : #livres recommandés et vendus

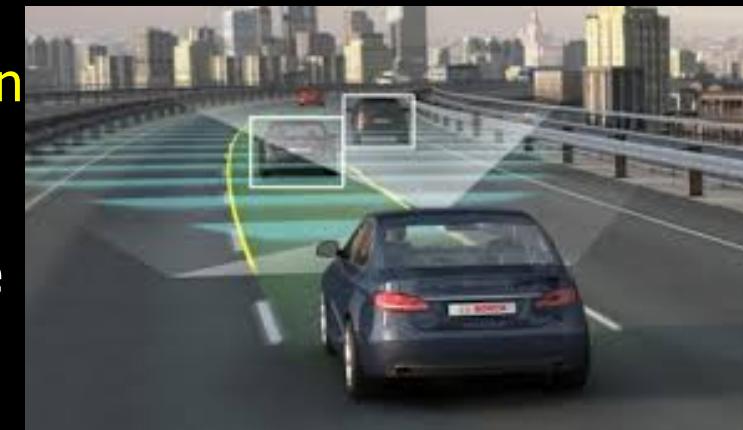
E : base de donnée de livres +
relation recommandation
récompense.



T: Conduite sur autoroute à l'aide de 4 capteurs de vision

P: Distance moyenne parcourue avant une erreur jugée par l'homme

E: une séquence d'images et de commandes de pilotage enregistrées pendant une conduite par un humain.



Quand utiliser du ML ?

Quand utiliser du ML ?

Lorsque l'humain est incapable d'expliquer son expertise.

Exemple : reconnaissance vocale



Quand utiliser du ML ?

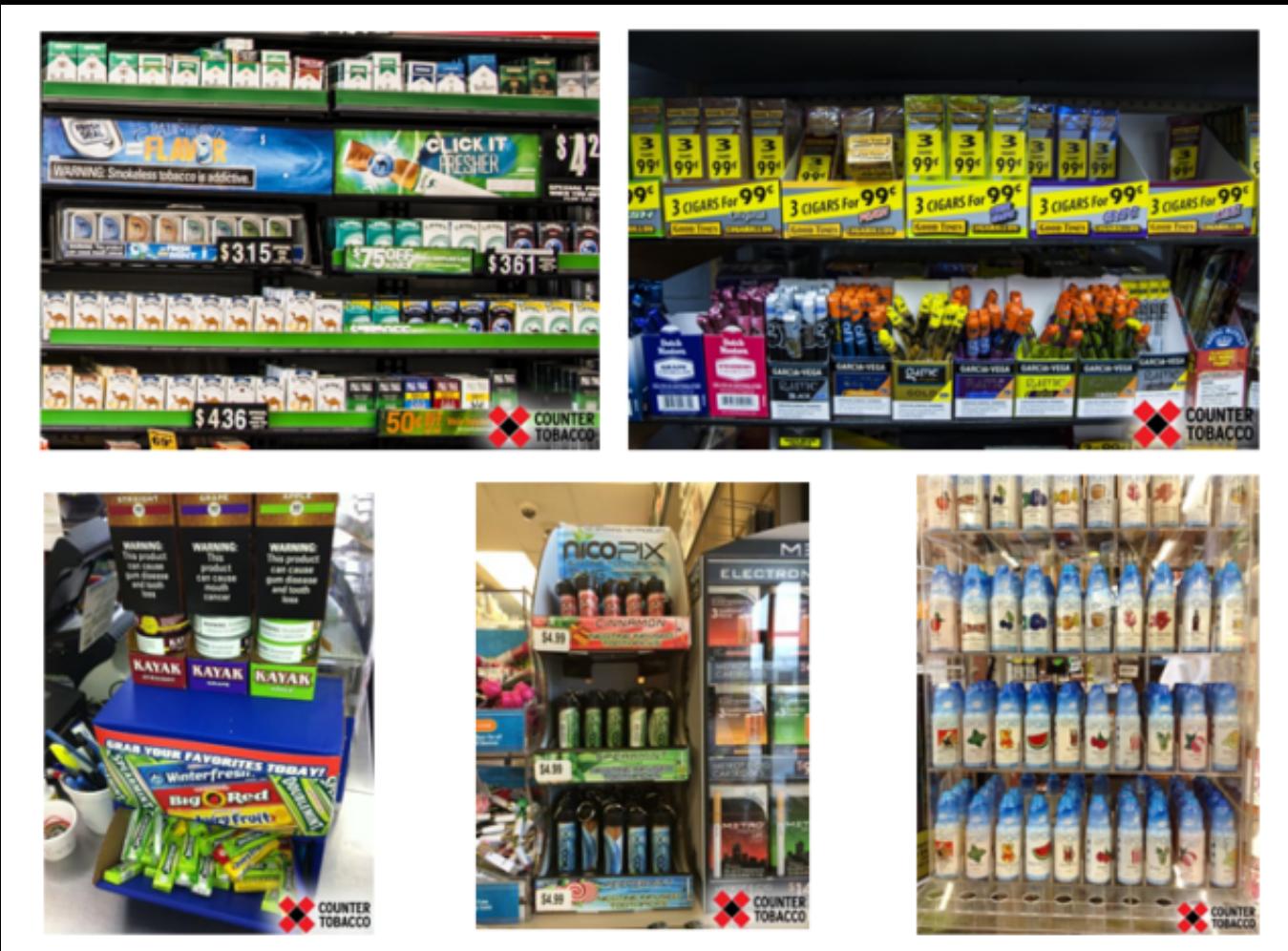
- Lorsque nous avons beaucoup de données
- Envie de les « analysés" pour en tirer profit

Reconnaissance de formes/visages



Quand utiliser le ML

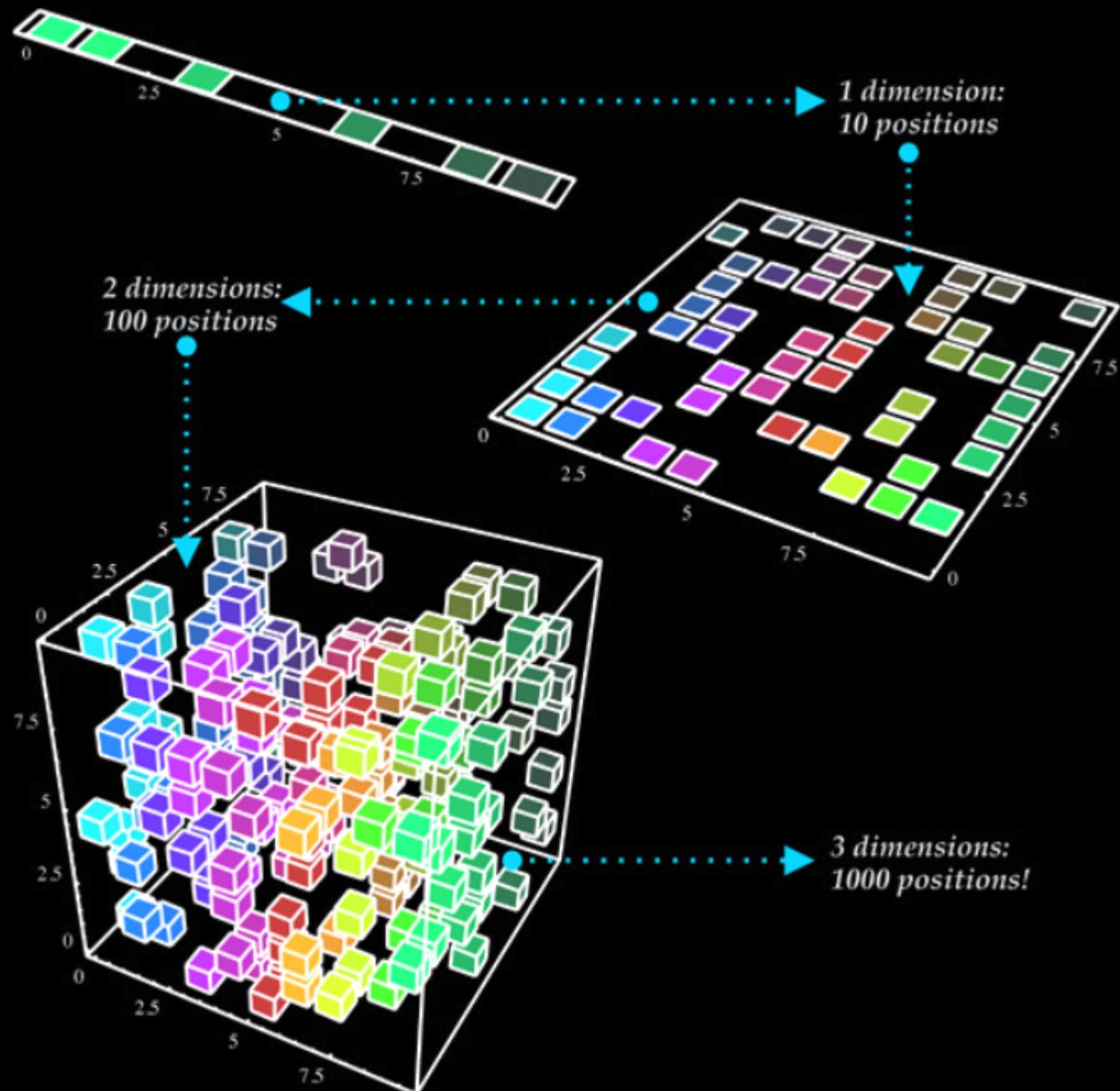
- Quand des modèles existent dans nos données, même si on ne sait pas ce qu'ils sont
- Exemple : La façon dont les produits non liés peuvent être associés les uns aux autres.



Réduction de dimension

□ Les données sont de haute dimension

→ Réduction de la dimension

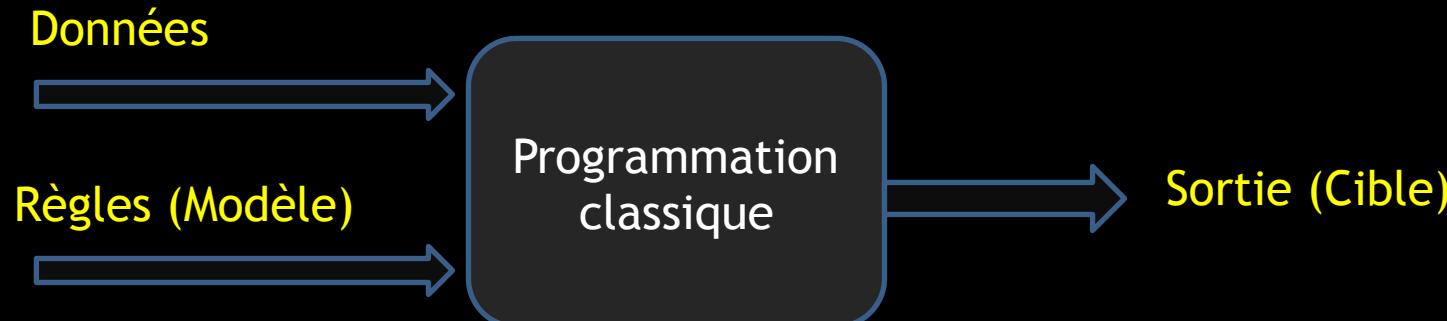


Programmation classique

vs

Machine Learning

Programmation classique vs ML



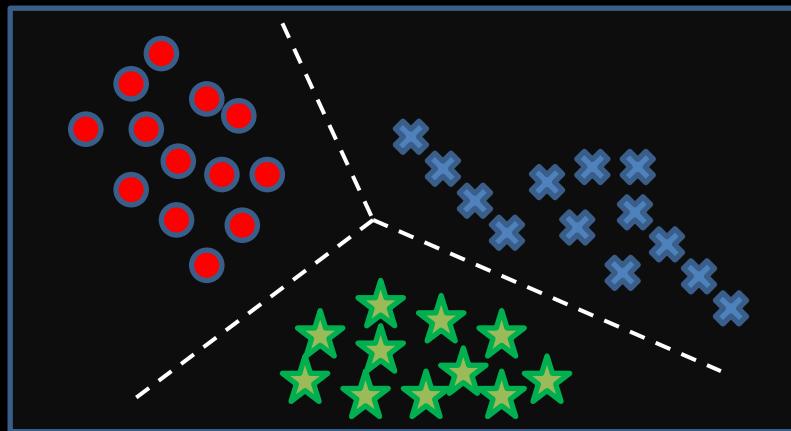
Modèle d'apprentissage artificiel

Un modèle d'apprentissage automatique a pour objectif de **déterminer la structure optimale dans un jeu de données** afin de réaliser la tâche demandée.

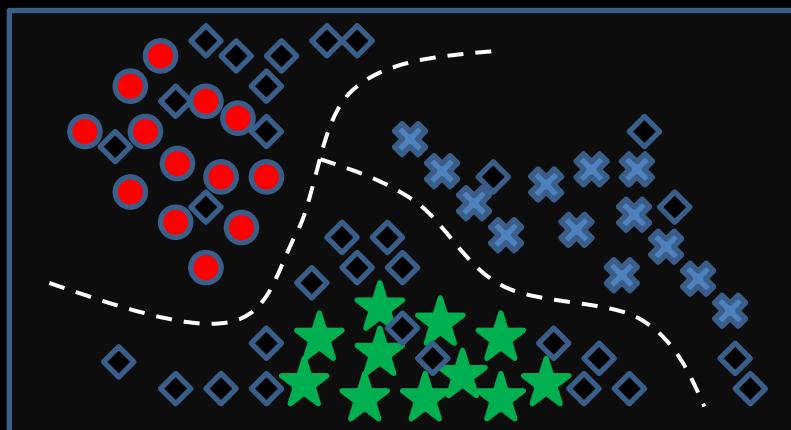
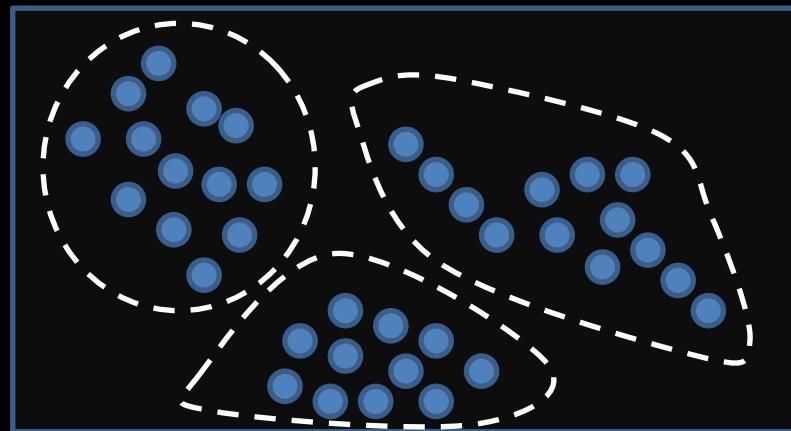
Pour générer un modèle d'apprentissage automatique, vous devez fournir des données de formation à un algorithme d'apprentissage automatique.

Principales classes d'algorithmes de ML

Apprentissage supervisé

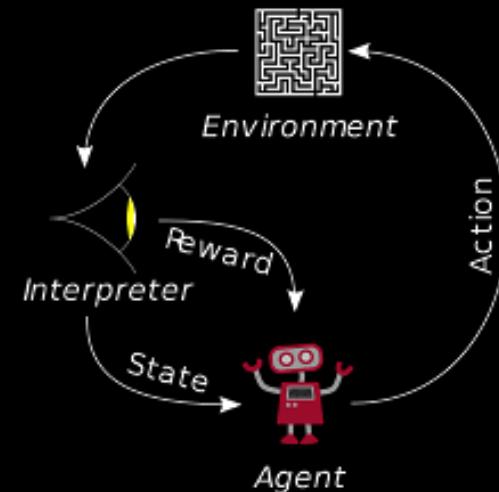


Apprentissage non supervisé



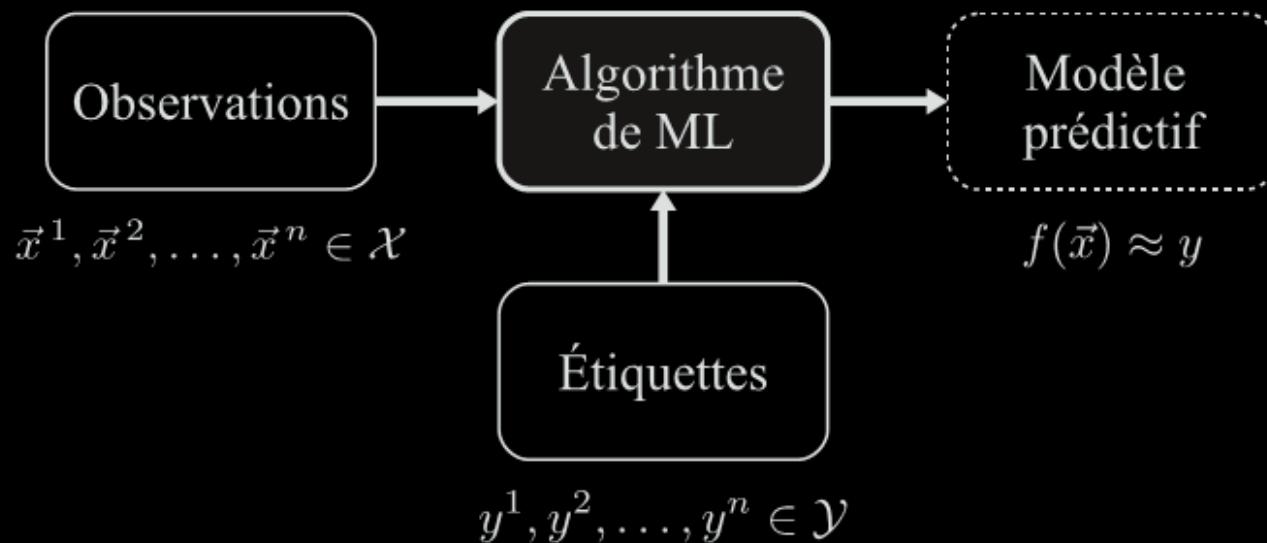
Apprentissage semi-supervisé

Apprentissage par renforcement



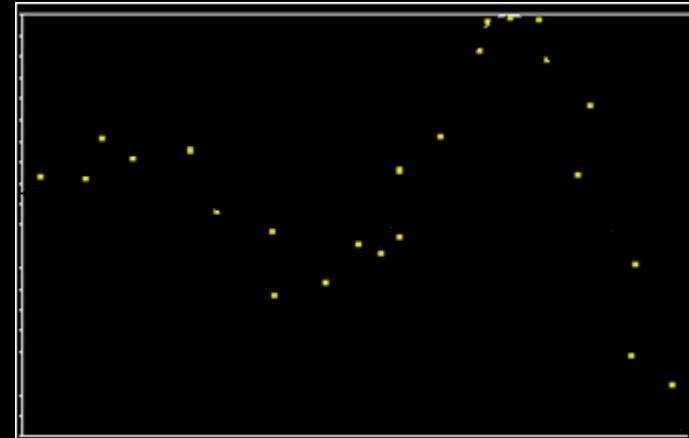
Apprentissage supervisé

- L'ensemble de données (données d'apprentissage) consiste en un ensemble de données d'entrée et de réponses correctes correspondant à chaque donnée.

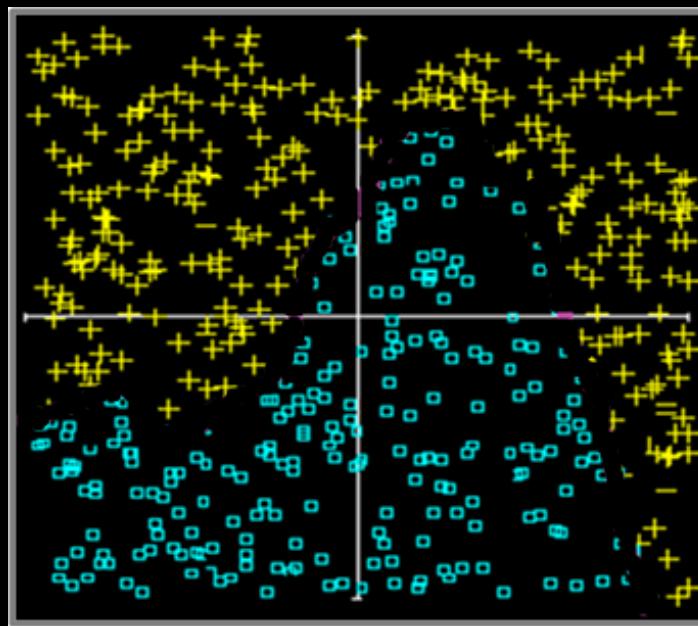


Apprentissage
supervisé

Probleme de Régression



entrée
points = exemples → *courbe = régression*



Problème de Classification

entrée =
position point
sortie désirée
=
classe (\square
 $= -1, + = +1$)
↓
Fonction étiquette = $f(x)$
(et frontière de séparation)

Apprentissage supervisé

Classification et Régression

User ID	Gender	Age	Salary	Purchased
15624510	Male	19	19000	0
15810944	Male	35	20000	1
15668575	Female	26	43000	0
15603246	Female	27	57000	0
15804002	Male	19	76000	1
15728773	Male	27	58000	1
15598044	Female	27	84000	0
15694829	Female	32	150000	1
15600575	Male	25	33000	1
15727311	Female	35	65000	0
15570769	Female	26	80000	1
15606274	Female	26	52000	0
15746139	Male	20	86000	1
15704987	Male	32	18000	0
15628972	Male	18	82000	0
15697686	Male	29	80000	0
15733883	Male	47	25000	1

Figure A: CLASSIFICATION



Valeur discrètes

Temperature	Pressure	Relative Humidity	Wind Direction	Wind Speed
10.69261758	986.882019	54.19337313	195.7150879	3.278597116
13.59184184	987.8729248	48.0648859	189.2951202	2.909167767
17.70494885	988.1119385	39.11965597	192.9273834	2.973036289
20.95430404	987.8500366	30.66273218	202.0752869	2.965289593
22.9278274	987.2833862	26.06723423	210.6589203	2.798230886
24.04233986	986.2907104	23.46918024	221.1188507	2.627005816
24.41475295	985.2338867	22.25082295	233.7911987	2.448749781
23.93361956	984.8914795	22.35178837	244.3504333	2.454271793
22.68800023	984.8461304	23.7538641	253.0864716	2.418341875
20.56425726	984.8380737	27.07867944	264.5071106	2.318677425
17.76400389	985.4262085	33.54900114	280.7827454	2.343950987
11.25680746	988.9386597	53.74139903	68.15406036	1.650191426
14.37810685	989.6819458	40.70884681	72.62069702	1.553469896
18.45114201	990.2960205	30.85038484	71.70604706	1.005017161
22.54895853	989.9562988	22.81738811	44.66042709	0.264133632
24.23155922	988.796875	19.74790765	318.3214111	0.329656571

Figure B: REGRESSION



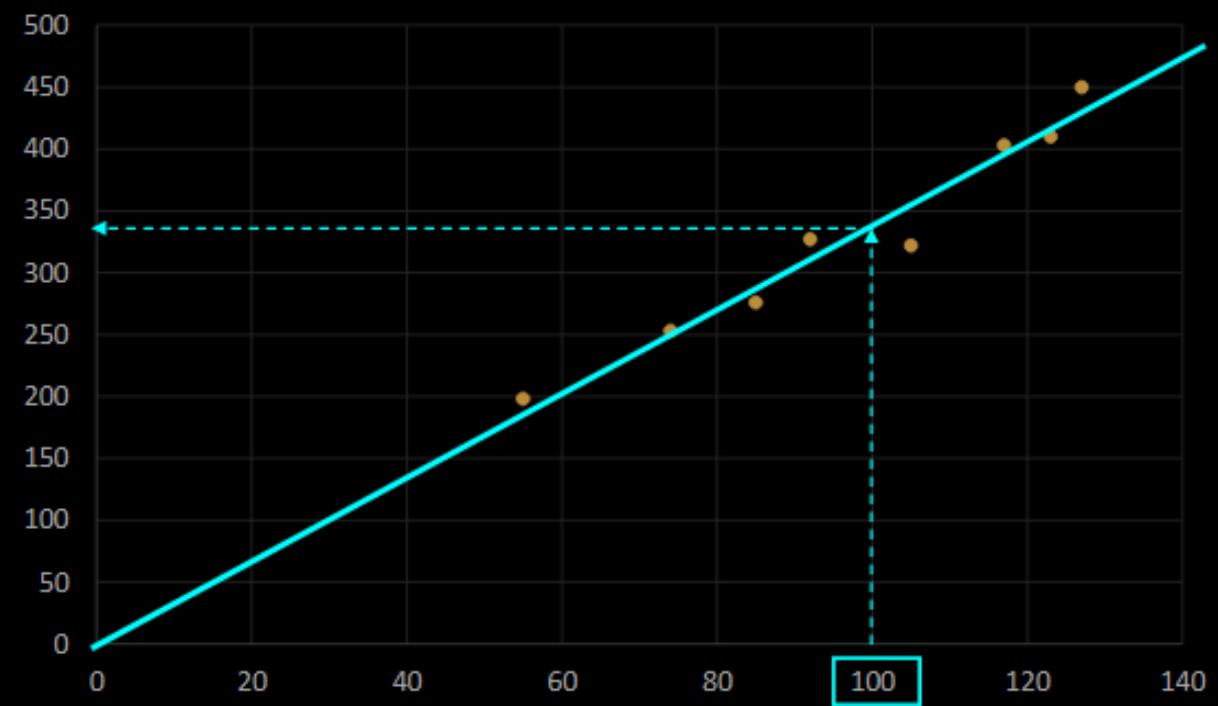
Valeur continues

Apprentissage supervisé Régression

Exemple -Estimation immobilière-

Superficie (m ²)	Prix (k€)
92	298
123	470
74	253
127	450
105	322
85	266
117	403
55	198

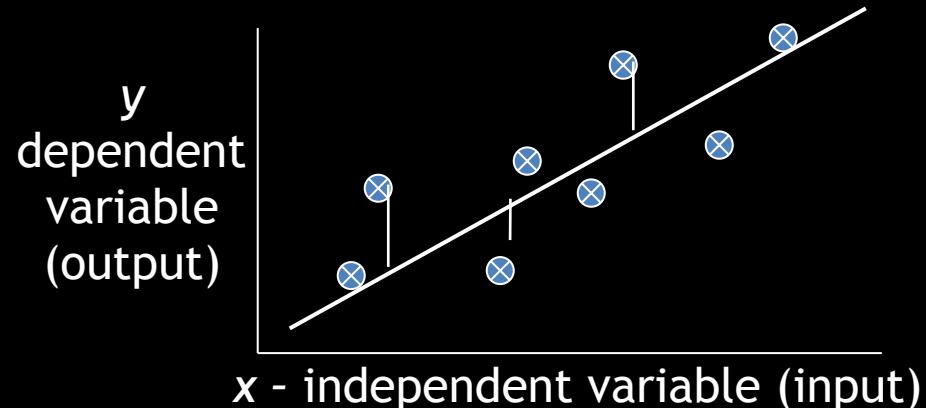
Base d'apprentissage



Modèle de prédiction

Apprentissage supervisé: régression

Régression linéaire



Objectif : apprendre l'hypothèse $h(\Theta)$ (c' à d les coefficients Θ_0 et Θ_1 qui minimisent l'erreur sur les données d'entraînement

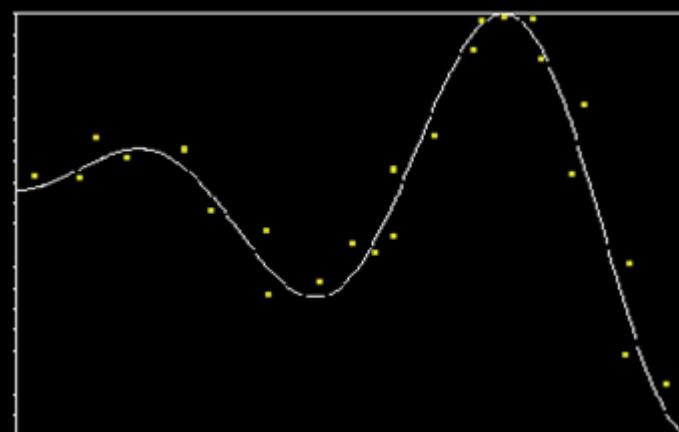
$$h(\Theta) = \Theta_0 + \Theta_1 x_1$$

Cas de plusieurs attributs (de dimension n par exemple)

$$h(\Theta) = \Theta_0 + \Theta_1 x_1 + \Theta_2 x_2 + \dots + \Theta_n x_n$$

$$MAPE = \frac{100}{n} \sum_{k=1}^n \left| \frac{\text{Valeur Observée}_k - \text{Valeur Prédictive}_k}{\text{Valeur Observée}_k} \right|$$

Régression non linéaire



$$h(\Theta) = \Theta_0 + \Theta_1 x + \Theta_2 x^2 + \dots + \Theta_k x^k$$

Apprentissage supervisé: problèmes de classification

Exemple: diagnostique de Cancer

Patient ID	# of Tumors	Avg Area	Avg Density	Diagnosis
1	5	20	118	Malignant
2	3	15	130	Benign
3	7	10	52	Benign
4	2	30	100	Malignant

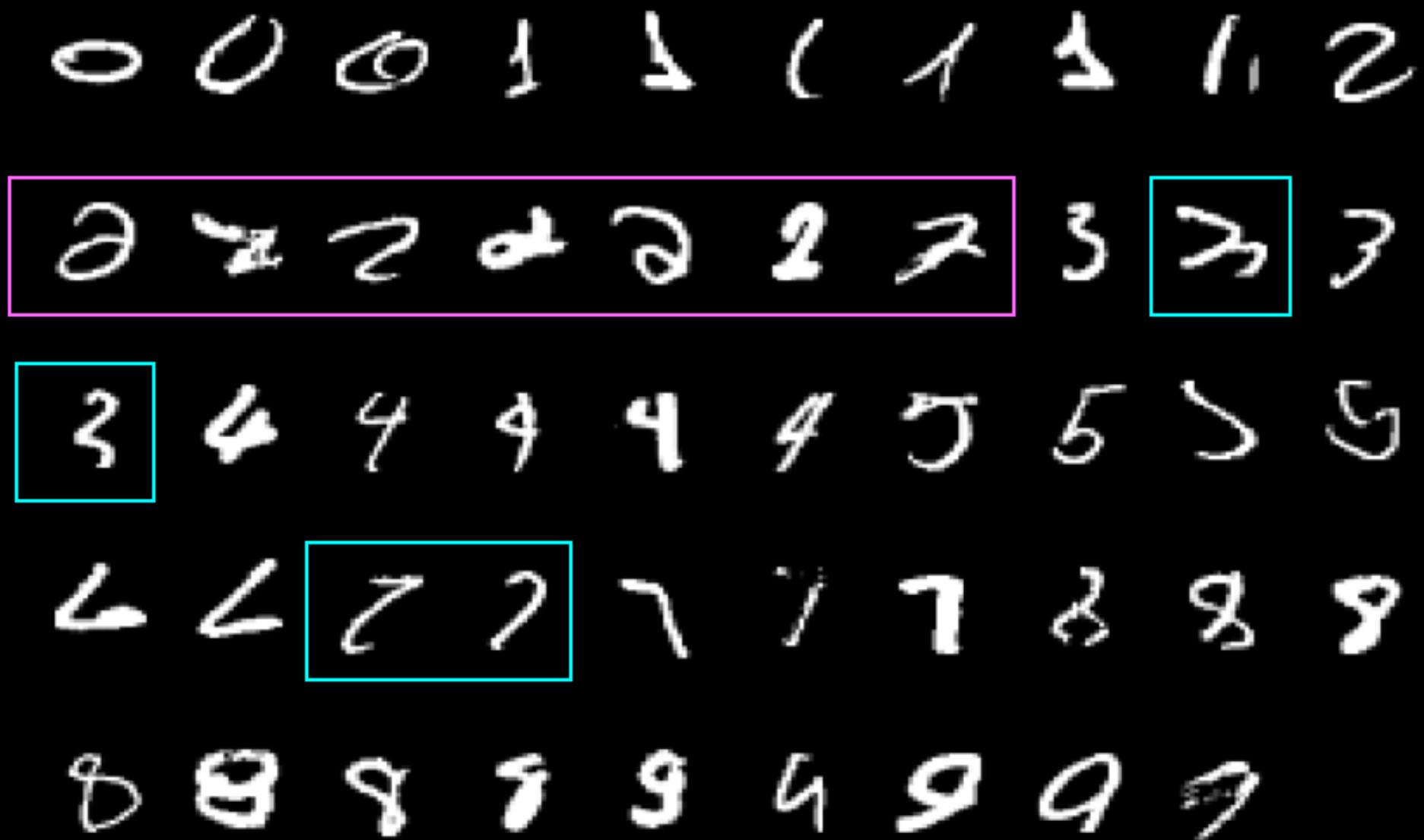
- Utilisez cet ensemble d'entraînement pour apprendre à classer les patients pour lesquels le diagnostic est inconnu:

Patient ID	# of Tumors	Avg Area	Avg Density	Diagnosis
101	4	16	95	?
102	9	22	125	?
103	1	14	80	?

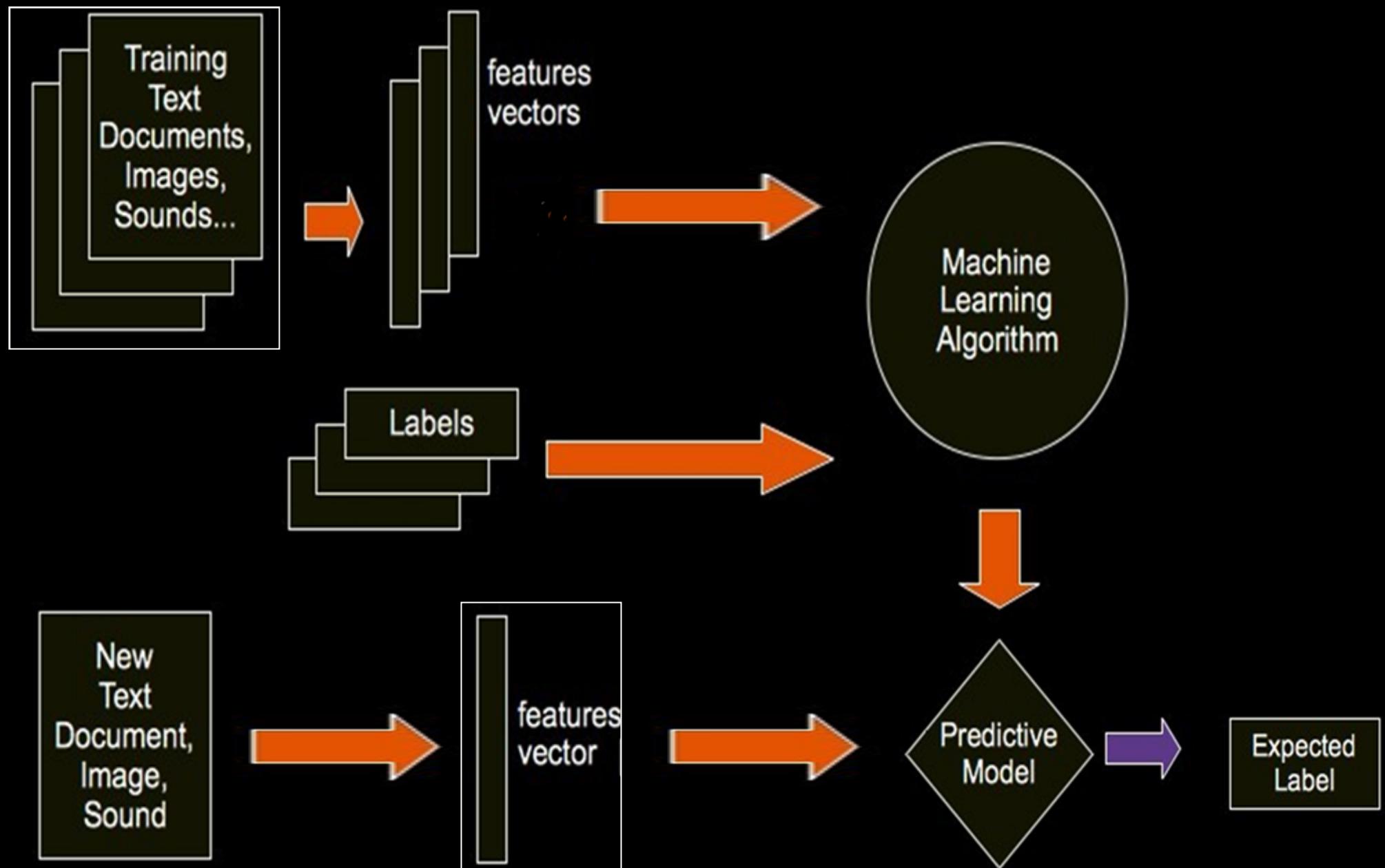
Données d'entrée

Classification

Exemple (Classification) : reconnaissance des chiffres/lettres manuellement écrites



Apprentissage supervisé (Démarche)

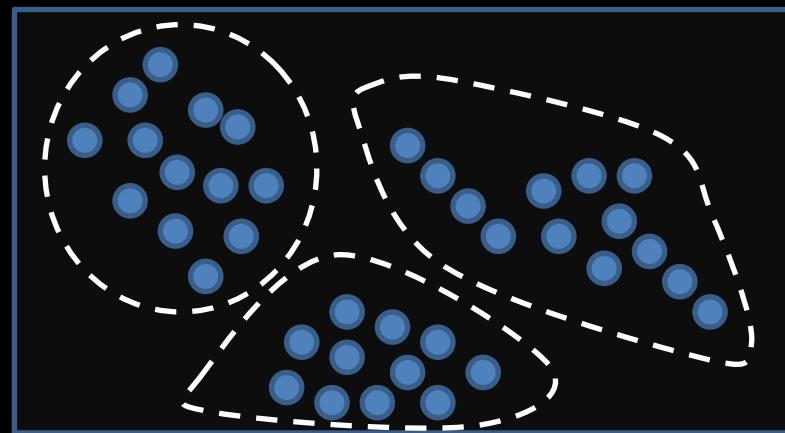


Apprentissage supervisé (Code python)

```
>>> from sklearn import Model  
>>> model = Model(param1=1e-8, param2="auto")  
>>> print(model.param2)  
"auto"  
>>> model.fit(X_train, y_train) # learn from training data  
>>> y_pred = model.predict(X_test) # predict from new data  
>>> model.score(X_test, y_test) # evaluate performance new data  
0.96
```

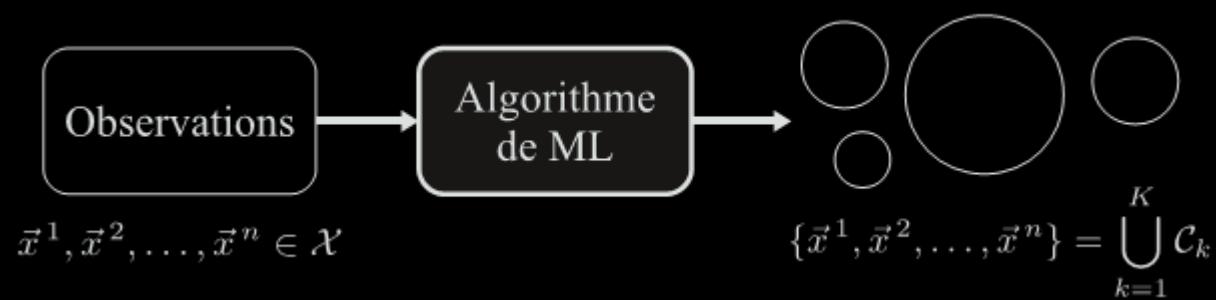
API: fit, predict, score

Apprentissage non supervisé

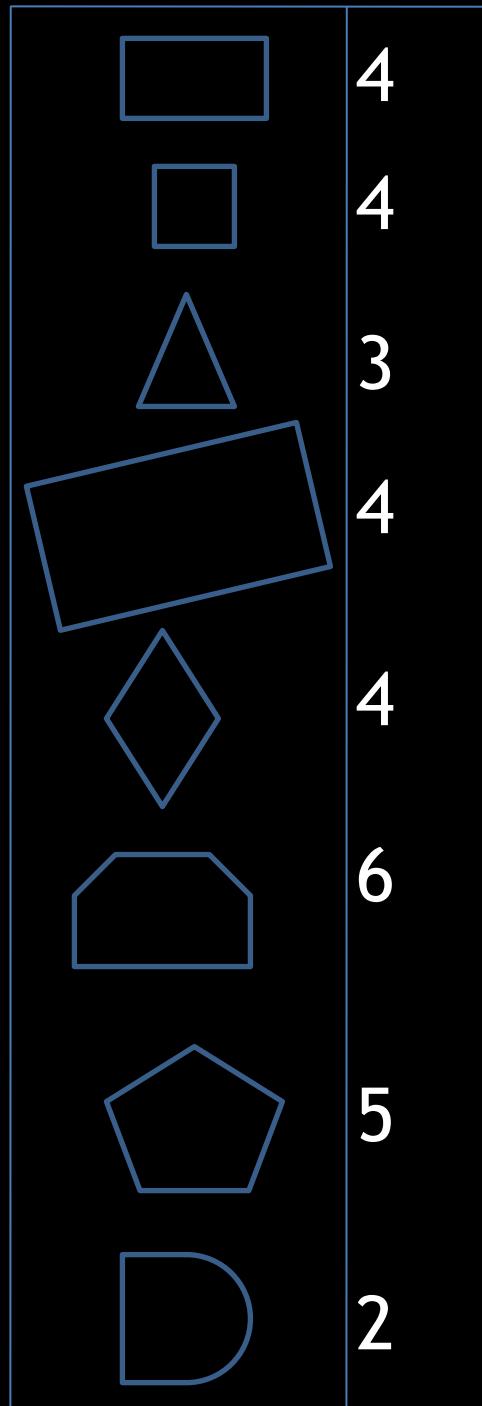


Apprentissage non supervisé

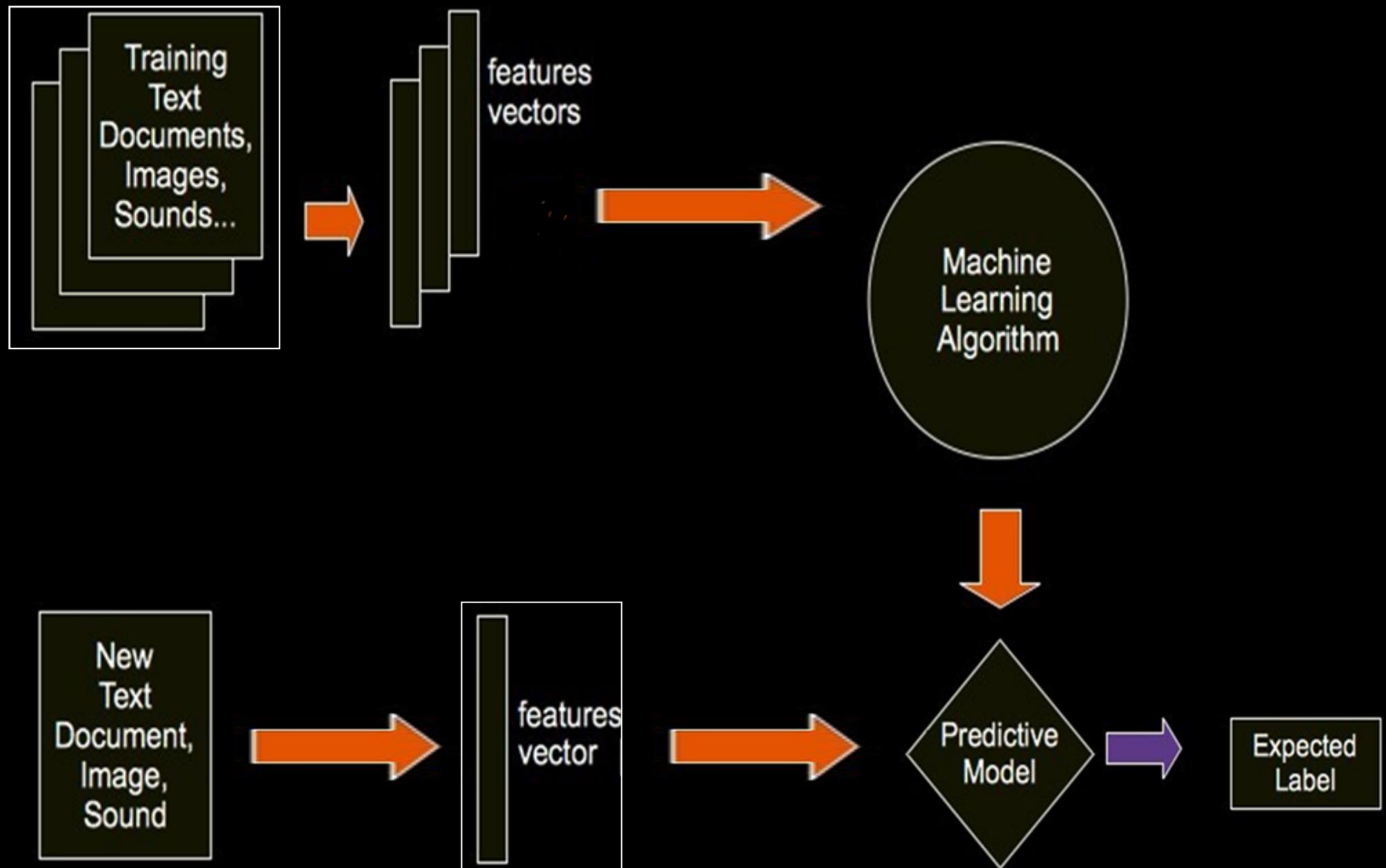
- **Principe:** Le but de l'apprentissage non supervisé est de trouver les sous-ensembles de données qui partagent des caractéristiques similaires **sans indiquer explicitement** que certains points de données appartiennent à une certaine classe.
- L'algorithme doit découvrir cette similitude par lui-même



Apprentissage non supervisé : (Exemple)



Apprentissage non-supervisé (Démarche)



Apprentissage non-supervisé (Code python)

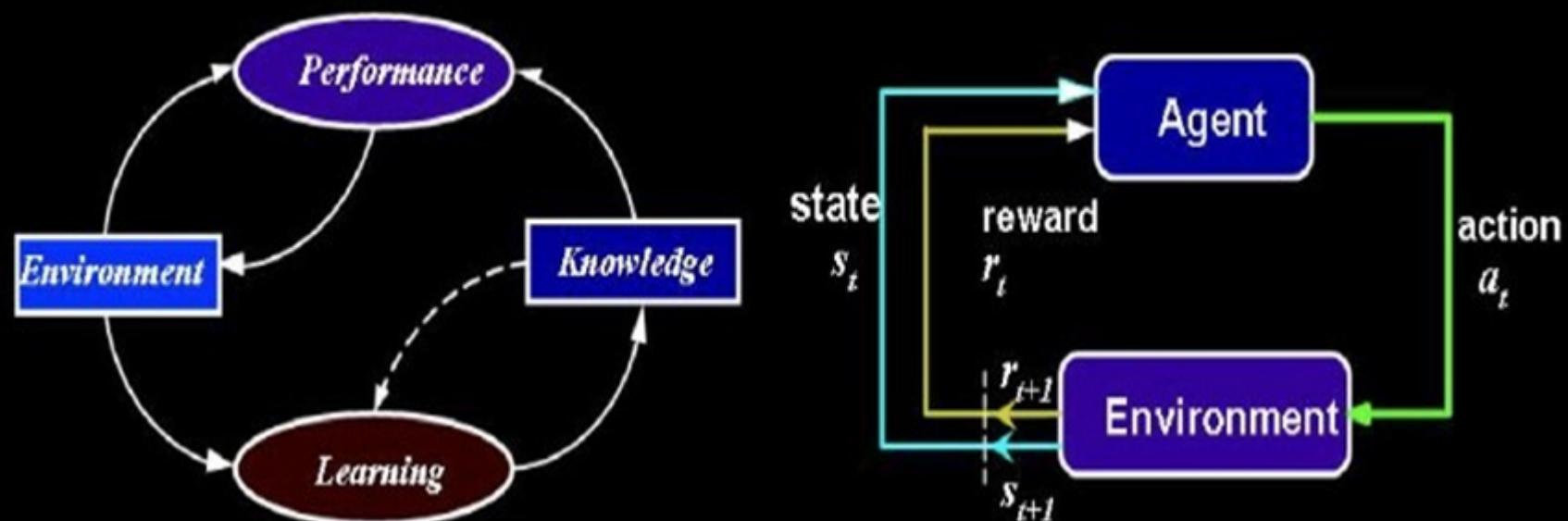
```
>>> from sklearn import Model  
>>> model = Model(param1=1e-8, param2="auto")  
>>> print(model.param2)  
"auto"  
>>> model.fit(X) # learn from training data (no y)  
>>> Xt = model.transform(X) # transform new or same data
```

API: fit, transform

Apprentissage par renforcement

Apprentissage par renforcement

- Le système d'apprentissage peut **interagir avec son environnement** et accomplir des actions.
- En retour de ces actions, le système reçoit une récompense indiquant si la réponse est correcte ou non, mais pas comment l'améliorer.
- L'apprenant par renforcement doit essayer différentes stratégies et apprendre laquelle fonctionne le mieux
- **Principe:** l'algorithme cherche dans l'espace d'état des entrées et des sorties possibles afin de maximiser une récompense.



Quelques considérations générales lors de la conception d'un système d'apprentissage artificiel

Conception d'un système d'apprentissage

- Choisissez l'expérience d'entraînement
- Préparez les données d'entraînement
- Choisissez exactement ce qui doit être appris, c'est-à-dire la **fonction cible**.
- Choisissez comment représenter la fonction cible
- Choisissez un algorithme d'apprentissage pour déduire la fonction cible de l'expérience.

Exhaustivité des données d'entrainements

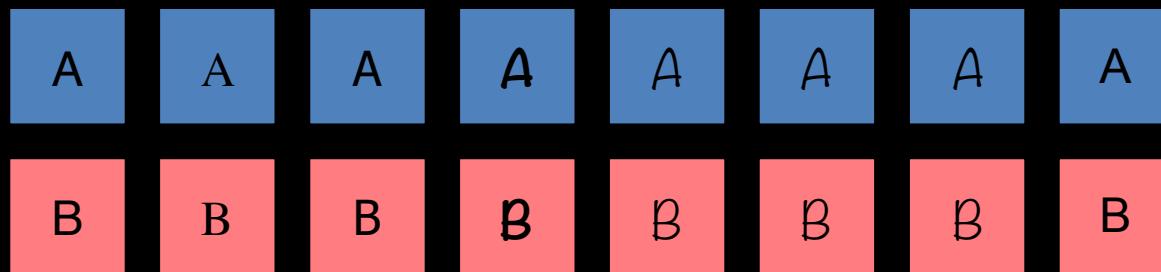
Il n'y a jamais création d'information non présente : il faut des données qui permettent vraiment d'apprendre ou injecter des connaissances a priori.

Ainsi, à partir de la base d'apprentissage de droite, peut-on prédire correctement le pluriel de « caillou » ?

Un fœtus	Des fœtus
Un cheveu	Des cheveux
Un pneu	Des pneus
Un animal	Des animaux
Un rail	Des rails
Une manche	Des manches
Une pelle	Des pelles
Un fenêtre	Des fenêtres
Une porte	Des portes
Un coucou	Des coucous

Parasitage (bruit)

A partir de la base suivante



Que va-t-il être annoncé pour b ?

99% de chance que ce soit A ... à cause du fond, information parasite exceptionnellement corrélée à notre problème et plus simple à analyser

Travail de l'expert

On attend d'un expert du problème :

D'identifier les informations pertinentes des informations parasites

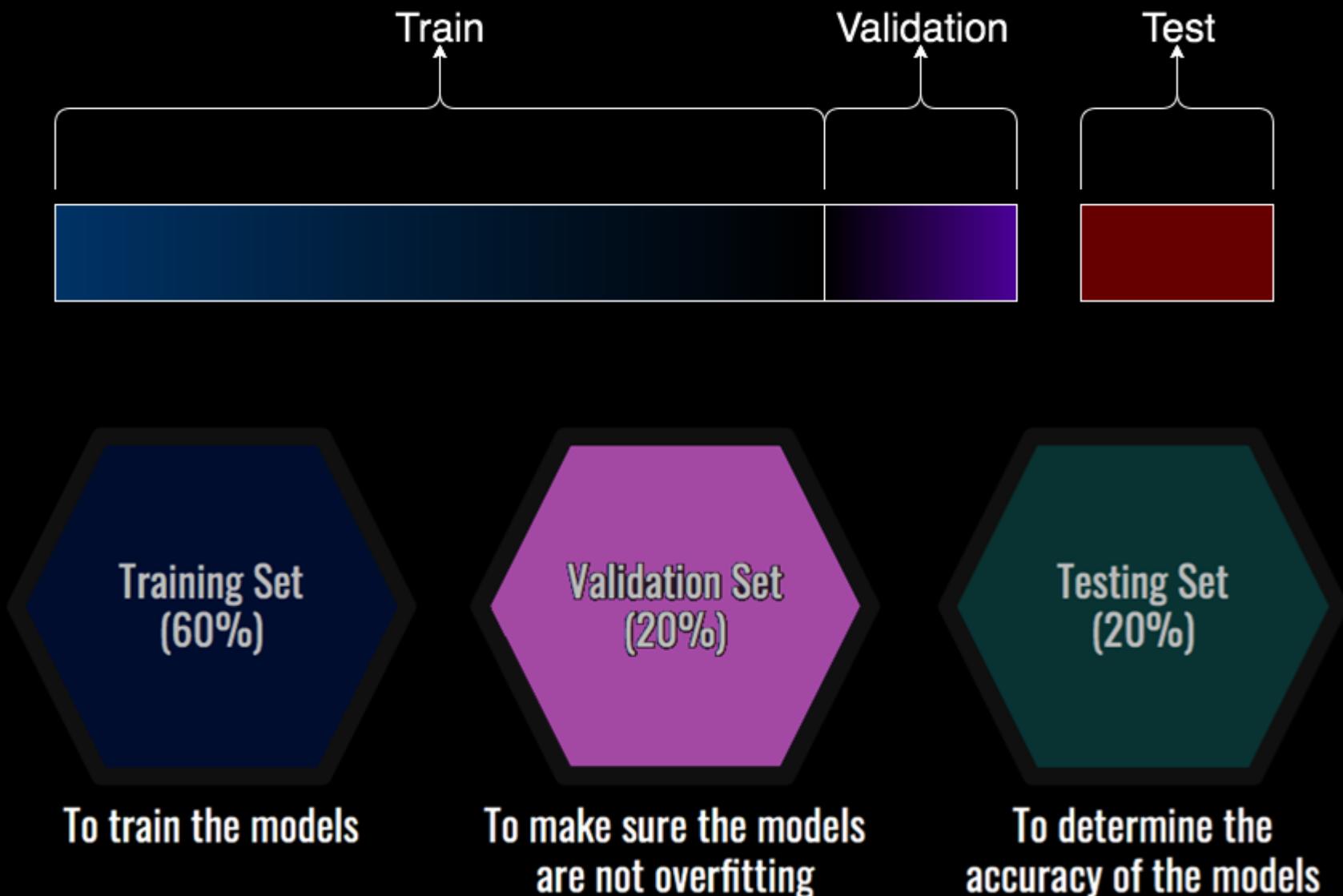
Donner des idées (même non mathématiques) sur le moyens d'éliminer certaines informations parasites (exemple binarisation, normalisation en taille, etc.).

Garantir que la base d'apprentissage est exhaustive.

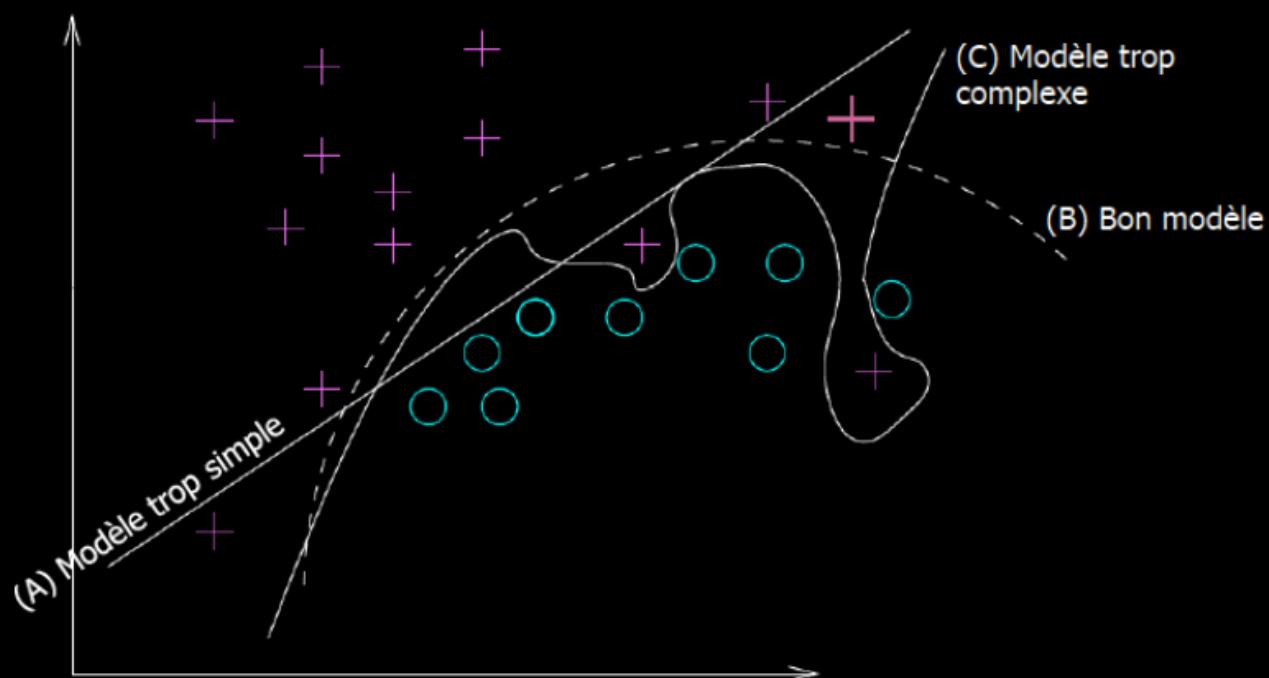
Garantir qu'elle est assez complète pour que les informations parasites non éliminées au prétraitement soient indépendantes du problème traité.

Le modèle appris est-il bon?

Validation du modèle appris



Le modèle appris est-il bon?



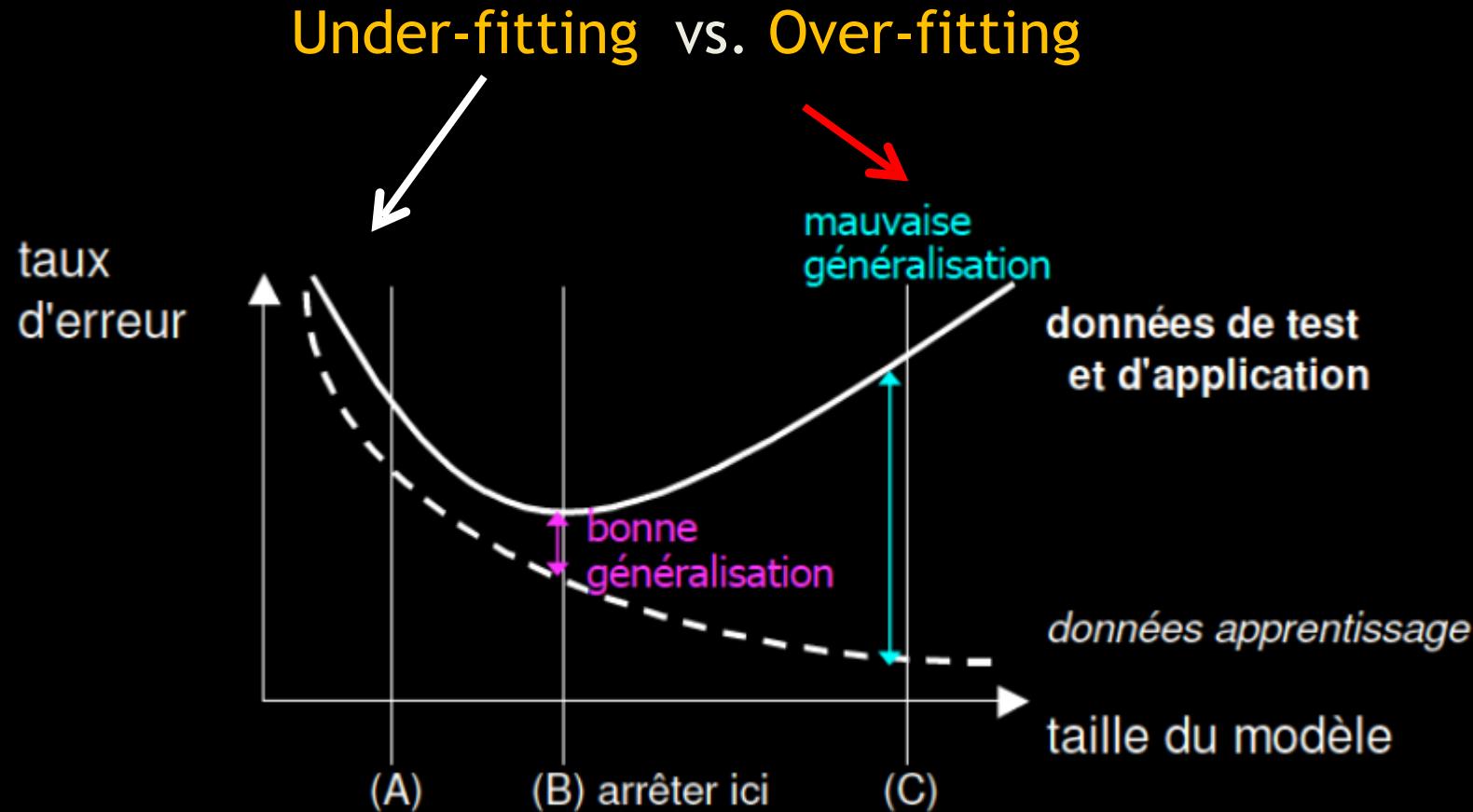
Source : Olivier Bousquet

Overfitting : un modèle trop spécialisé sur les données du Training Set et qui se généralisera mal

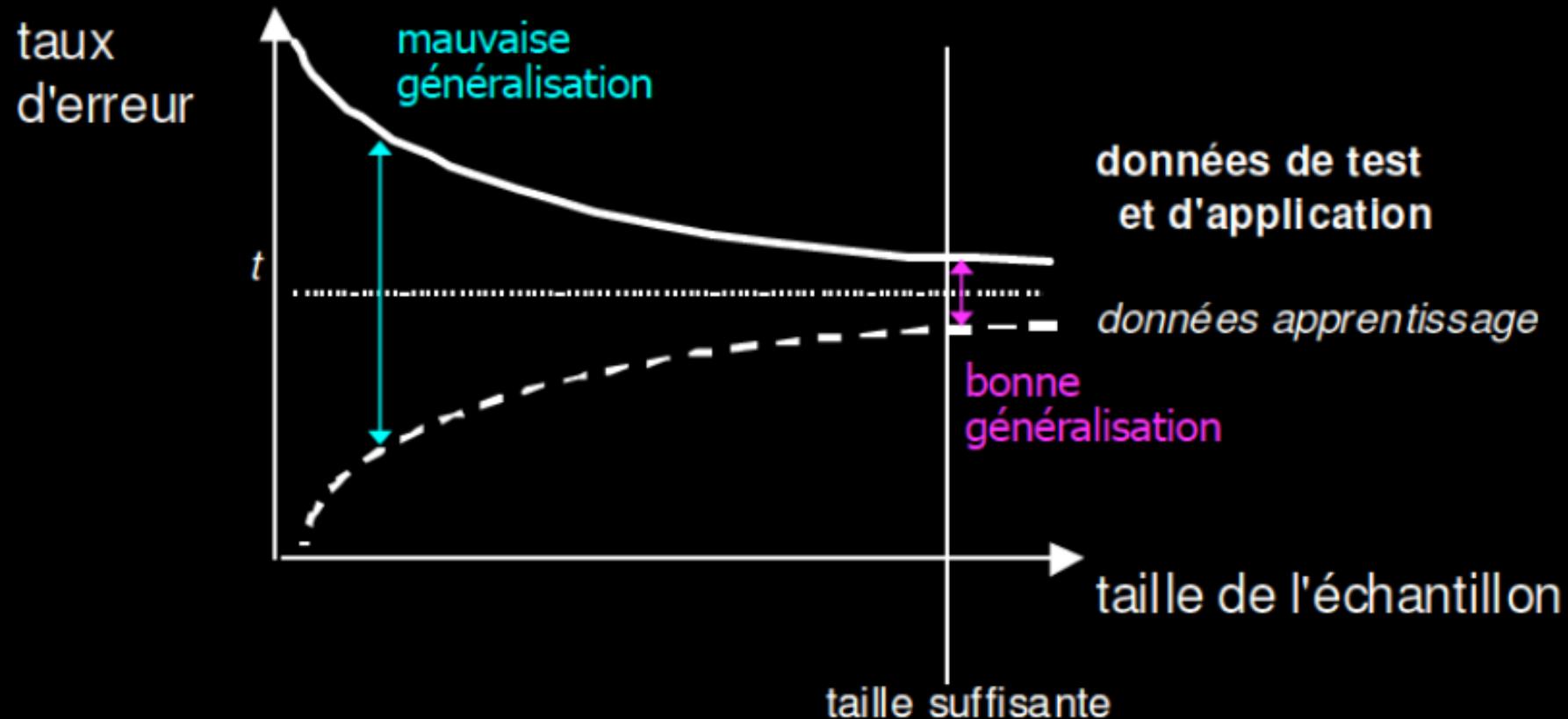
Underfitting : un modèle généraliste incapable de fournir des prédictions précises

L'**Overfitting** (sur-apprentissage), et l'**Underfitting** (sous-apprentissage) sont les causes principales des mauvaises performances des modèles prédictifs générés par les algorithmes de Machine Learning.

Taux d'erreur en fonction de la complexité du modèle.

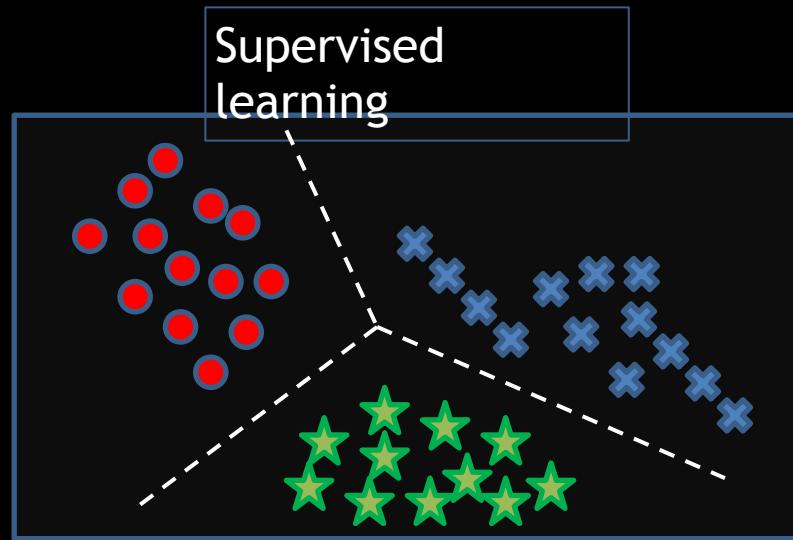


Courbes du taux d'erreur en apprentissage et en test.

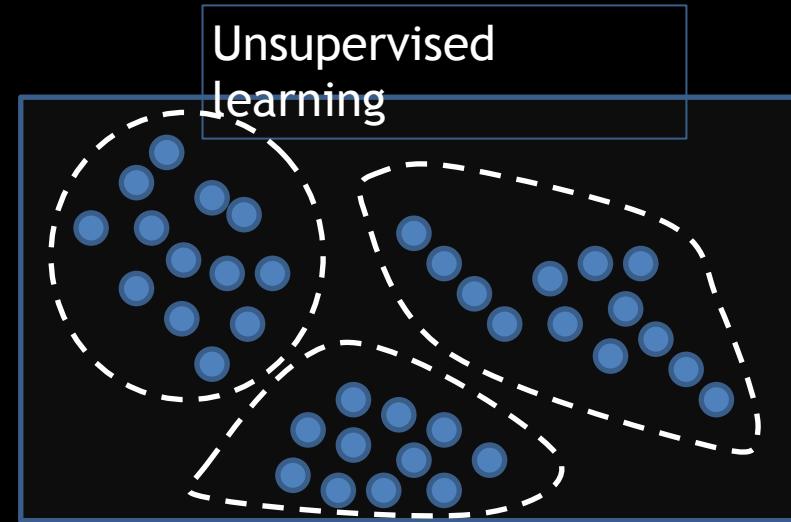


Principales classes d'algorithmes d'apprentissage

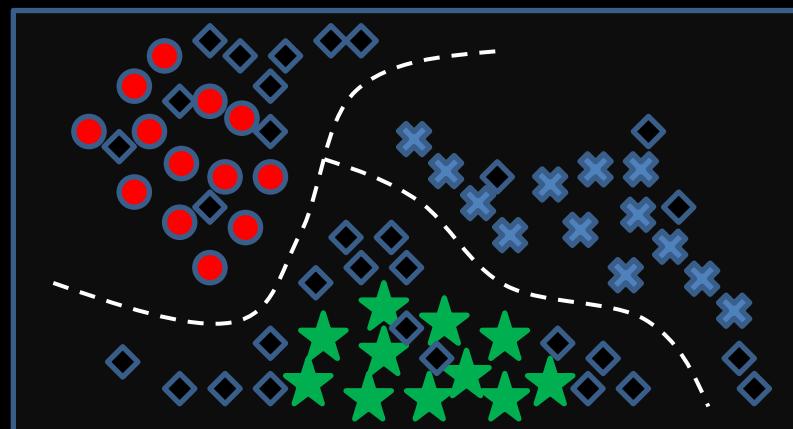
Supervised learning



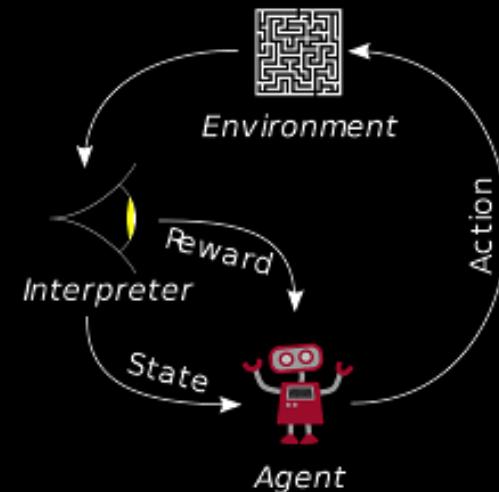
Unsupervised learning



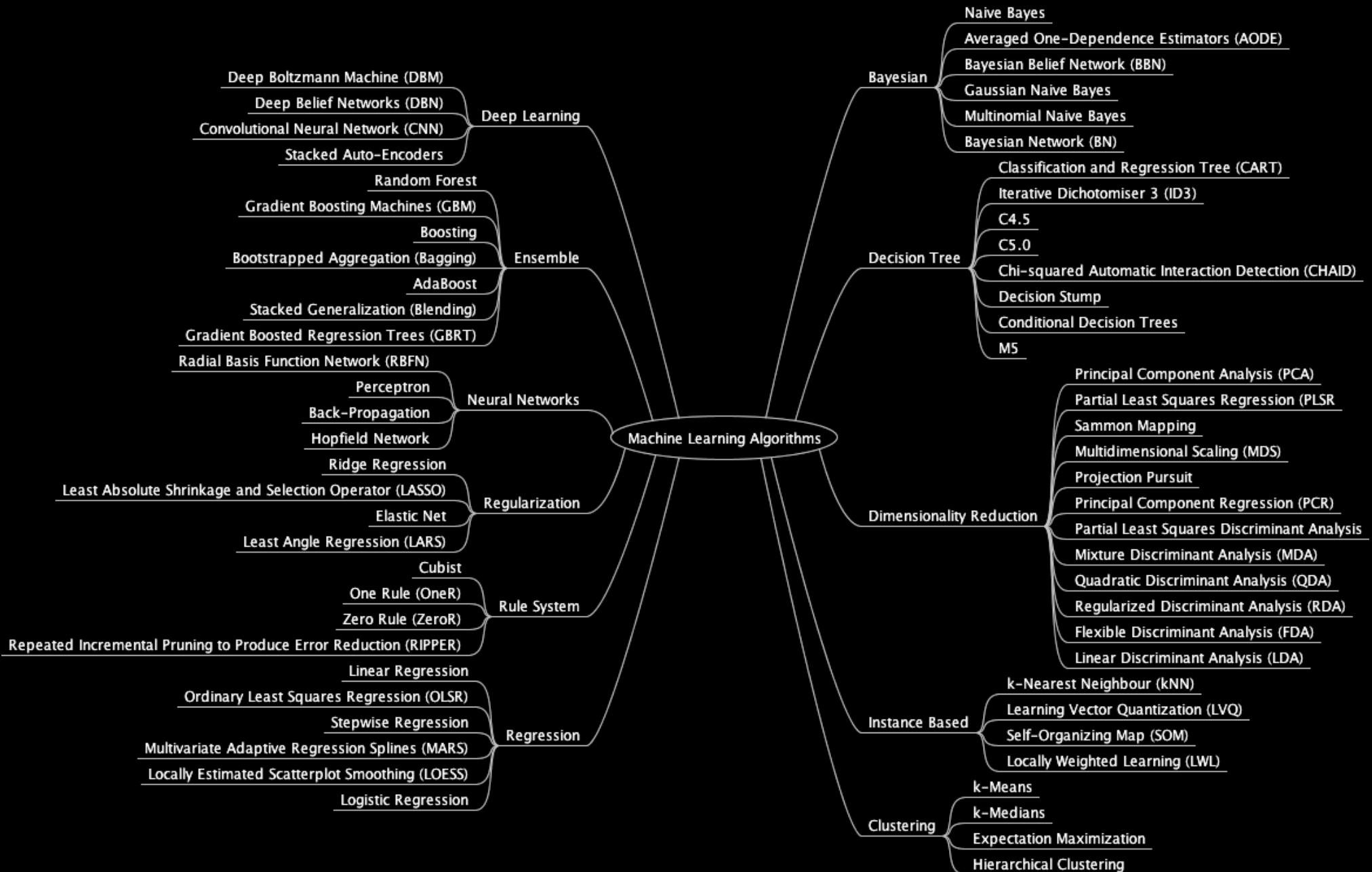
Semi-supervised learning



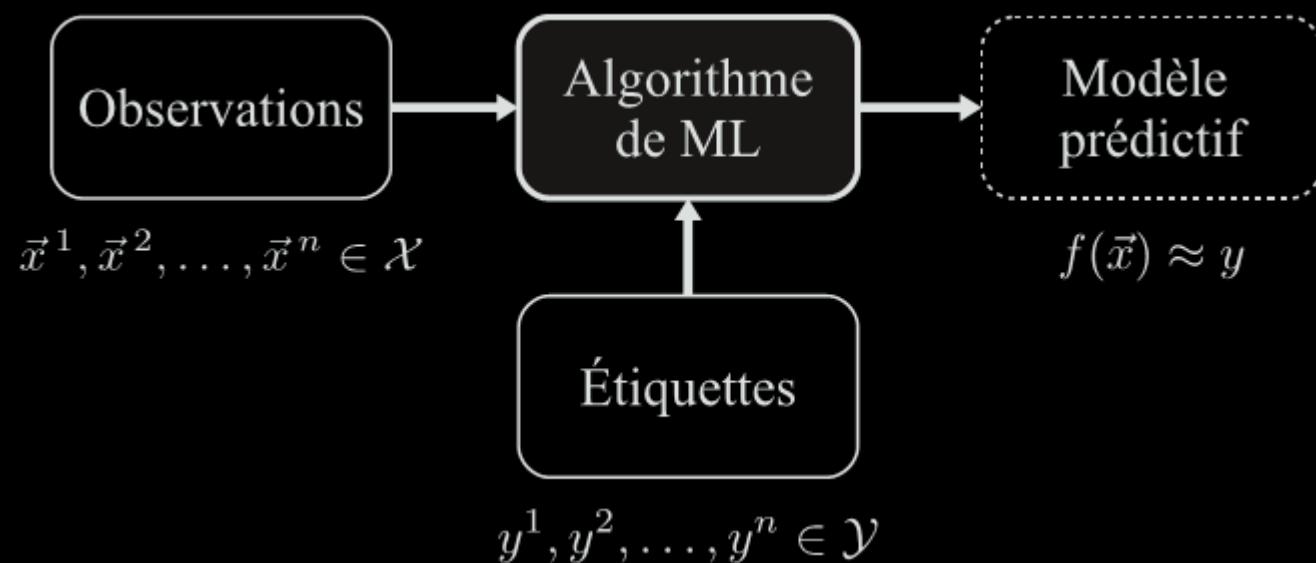
Reinforcement Learning



ML algorithms mindmap



Apprentissage supervisé: Régression linéaire et descente de gradient



Représentation de l'hypothèse (Fonction)

Dans ce cas, nous représentons '**y**' comme une combinaison linéaire des entrées (**x**)

Qui conduit à:

$$h(\Theta) = \Theta_0 + \Theta_1 x_1 + \Theta_2 x_2 + \dots + \Theta_n x_n$$

Où (Θ_i) sont les paramètres (aussi appelé poids)

Pour plus de commodité et de facilité de représentation: $x_0 = 1$

Pour que l'équation ci-dessus devienne:

$$h(x) = \sum_{i=0}^n (\Theta^T x)$$

L'objectif maintenant est d '"apprendre" les paramètres Θ afin que $h(x)$ devienne aussi proche de "y" au moins pour l'ensemble d'entraînement.

Régression linéaire et descente de gradient :

On définit une fonction qui mesure pour chaque valeur de Θ , la distance entre les $h(x^{(i)})$ et les $y^{(i)}$ correspondants

On définit la '**fonction coût**' comme suit:

$$J(\Theta) = \frac{1}{2} \sum_{i=1}^m (h_{(\Theta)}(x^{(i)}) - y^{(i)})^2$$

Nous voulons choisir les paramètres afin de minimiser $J(\Theta)$

L'algorithme Moindres Carrés Moyens LMS (Least Mean Squares) commence par une valeur initiale de Θ et change de façon répétée afin de rendre $J(\Theta)$ plus petit

Régression linéaire et descente de gradient :

Nous arrivons maintenant à l'algorithme Gradient Descent:

La descente de gradient commence par une initiale Θ , et effectue continuellement la mise à jour suivante:

$$\Theta^j := \Theta^j - \alpha \frac{\partial}{\partial \Theta_j} J(\Theta)$$

(Cette mise à jour est effectuée simultanément pour toutes les valeurs de $j=0,1,\dots,n$)

α est appelé taux d'apprentissage

Ceci, en fait, prend la forme suivante:

$$\Theta_j := \Theta_j - \alpha \frac{\partial}{\partial \Theta_j} \left(\frac{1}{2} \sum_{i=1}^m (h_{(\Theta)}(x^{(i)}) - y^{(i)})^2 \right)$$

Régression linéaire et descente de gradient :

Tout ce simplification de l'équation précédente donne:
(pour un seul exemple d'entraînement) :

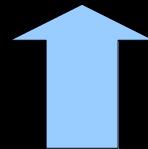
$$\Theta^j := \Theta^j + \alpha (y^{(i)} - h^\Theta(x^{(i)})) x^{(i)}$$

Il y a 2 façons de généraliser l'équation ci-dessus pour plus d'un exemple d'entraînement:

Le premier est:

Répéter jusqu'à la convergence

$$\left\{ \begin{array}{l} \Theta_j := \Theta_j - \alpha \sum_{i=1}^m (y^{(i)} - h_\Theta(x^{(i)})) x_j^{(i)} \quad \text{Pour chaque } j \\ \end{array} \right.$$



Cette méthode est appelée descente par gradient

Régression linéaire et descente de gradient :

La deuxième est appelé descente de gradient stochastique ou descente de gradient incrémentielle

Loop

{

for i=1 to m

{

$$\Theta_j := \Theta_j + \alpha (y^{(i)} - h_{\Theta}(x^{(i)})) x_j^{(i)} \quad (\text{for every } j)$$

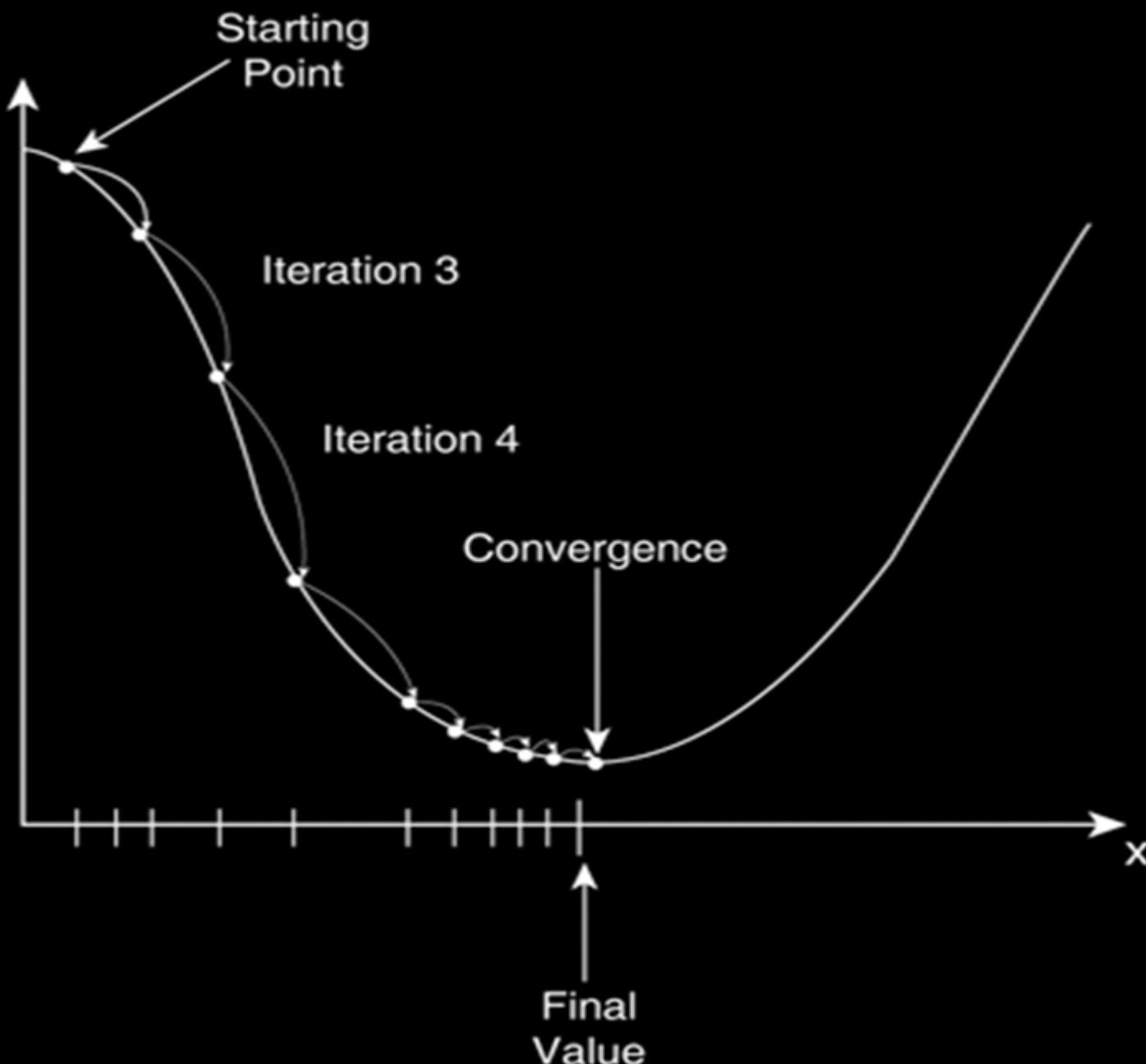
}



Mis à jour les paramètres à partir du (c'est à dire, des dérivées partielles) d'un seul exemple, choisi aléatoirement:

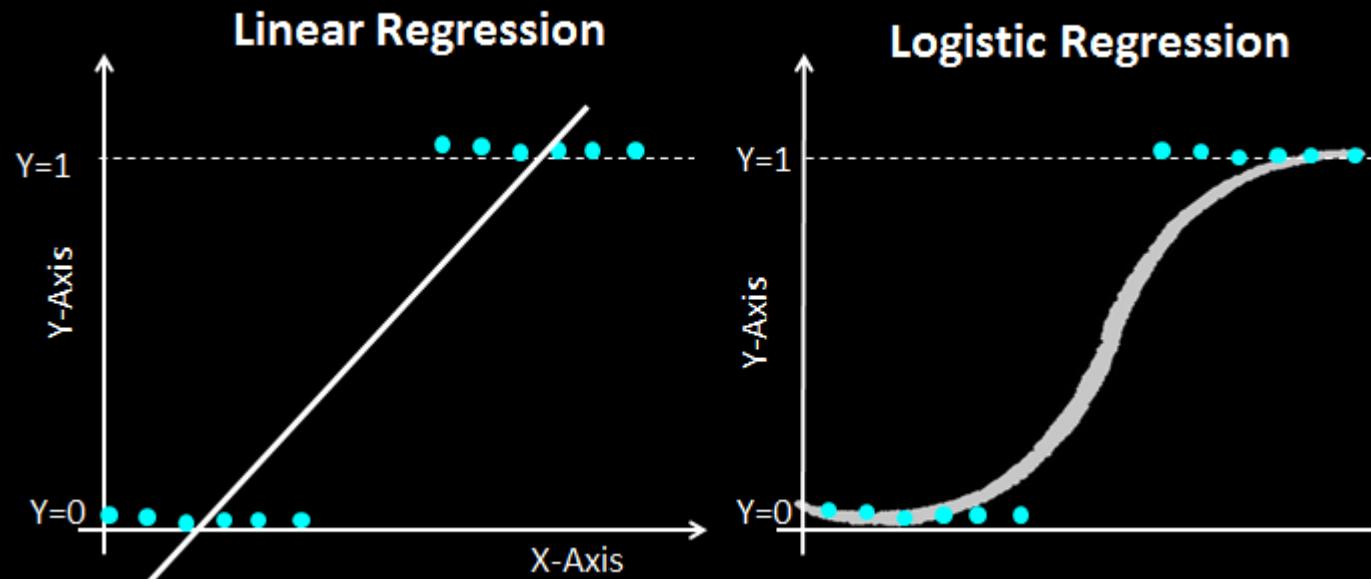
Cette procédure est beaucoup plus efficace lorsque l'ensemble d'entraînement est grand

Régression linéaire et descente de gradient :



Algorithmes de Classification

Classification : Régression Logistique

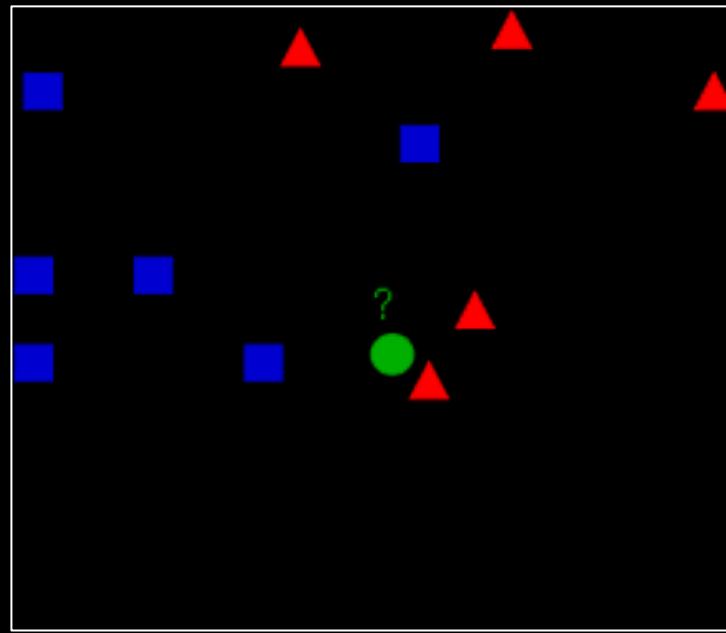


$$h(\Theta) = \Theta_0 + \Theta_1 x$$

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

La régression linéaire vous donne une sortie continue, mais la régression logistique fournit une sortie constante (0 ou 1).

K - plus proches voisins



Exemple de classification k -NN. L'échantillon de test (cercle vert) pourrait être classé soit dans la première classe de carré bleu ou la seconde classe de triangles rouges. Si $k = 3$ (cercle en ligne pleine) il est affecté à la seconde classe car il y a deux triangles et seulement un carré dans le cercle considéré. Si $k = 5$ (cercle en ligne pointillée) il est assigné à la première classe (3 carrés face à deux triangles dans le cercle externe).

K - plus proches voisins

K = 3

Customer	Age	Income	No. credit cards	Loyal
John 	35	35K	3	No
Rachel 	22	50K	2	Yes
Hannah 	63	200K	1	No
Tom 	59	170K	1	No
Nellie 	25	40K	4	Yes
David 	37	50K	2	Yes

Distance from David

$$\sqrt{[(35-37)^2 + (35-50)^2 + (3-2)^2]} = \mathbf{15.16}$$

$$\sqrt{[(22-37)^2 + (50-50)^2 + (2-2)^2]} = \mathbf{15}$$

$$\sqrt{[(63-37)^2 + (200-50)^2 + (1-2)^2]} = \mathbf{152.23}$$

$$\sqrt{[(59-37)^2 + (170-50)^2 + (1-2)^2]} = \mathbf{122}$$

$$\sqrt{[(25-37)^2 + (40-50)^2 + (4-2)^2]} = \mathbf{15.74}$$

Classification par Arbre de décision

Les arbres de décision : exemple

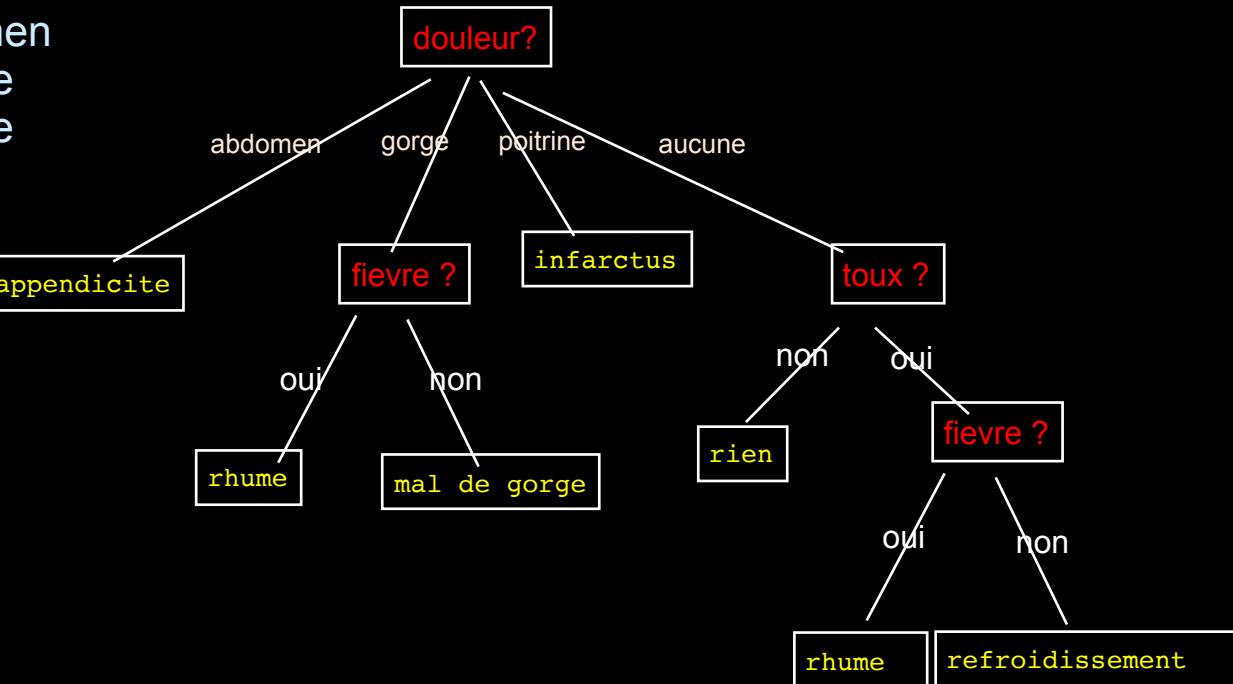
Les arbres de décision sont des classificateurs d'instances représentées dans un formalisme attribut/valeur

- Les nœuds de l'arbre testent les attributs
- Il y a une branche pour chaque valeur de l'attribut testé
- Les *feuilles* spécifient les catégories (deux ou plus)

Exemple

	Toux	Fièvre	Poids	Douleur
Marie	non	oui	normal	gorge
Fred	non	oui	normal	abdomen
Julie	oui	oui	maigre	aucune
Elvis	oui	non	obese	poitrine

Arbres de décision



Les arbres de décision : pouvoir de représentation

- Le choix des attributs est très important !
- Si un attribut crucial n'est pas représenté on ne pourra pas trouver d'arbre de décision qui apprenne les exemples correctement.
- Si deux instances ont la même représentation mais appartiennent à deux classes différentes, le langage des instances (les attributs) est dit inadéquat.

	Toux	Fièvre	Poids	Douleur	Diagnostic
Marie	non	oui	normal	abdomen	rhume
Polo	non	oui	normal	abdomen	appendicite
.....					

Arbres de décision : Algorithme

- Il construit les arbres de décision de haut en bas.
- Il place à la **racine l'attribut le plus important**, c'est-à-dire celui qui sépare le mieux les exemples positifs et négatifs.
- Par la suite, il y a un nouveau nœud pour chacune des valeurs possibles de cet attribut.
- Pour chacun de ces nœuds, on recommence le test avec le sous-ensemble des exemples d'entraînement qui ont été classés dans ce nœud.

Arbres de décision : Exemple

Journée	Ciel	Température	Humidité	Vent	JouerTennis
J1	Ensoleillé	Chaud	Élevée	Faible	Non
J2	Ensoleillé	Chaud	Élevée	Fort	Non
J3	Nuageux	Chaud	Élevée	Faible	Oui
J4	Pluvieux	Tempérée	Élevée	Faible	Oui
J5	Pluvieux	Froide	Normal	Faible	Oui
J6	Pluvieux	Froide	Normal	Fort	Non
J7	Nuageux	Froide	Normal	Fort	Oui
J8	Ensoleillé	Tempérée	Élevée	Faible	Non
J9	Ensoleillé	Froide	Normal	Faible	Oui
J10	Pluvieux	Tempérée	Normal	Faible	Oui
J11	Ensoleillé	Tempérée	Normal	Fort	Oui
J12	Nuageux	Tempérée	Élevée	Fort	Oui
J13	Nuageux	Chaud	Normal	Faible	Oui
J14	Pluvieux	Tempérée	Élevée	Fort	Non

Notions nécessaires (théorie de l'information)

L'entropie de Boltzmann et de Shannon

Shannon en 1949 a proposé une mesure d'entropie valable pour les distributions discrètes de probabilité. Elle exprime la quantité d'information, c'est à dire le nombre de bits nécessaire pour spécifier la distribution

L'entropie d'information est:

$$E = - \sum_{i=1..k} p_i \times \log_2(p_i)$$

où p_i est la probabilité de la classe C_i .

Gain d'information

$$Gain(S, A) = E(S) - \sum_{v \in valeurs(A)} \frac{|S_v|}{|S|} \cdot E(S_v)$$

$|S_v|$: taille de la sous-population dans la branche v de A

Gain: En quoi la connaissance de la valeur de l'attribut A m'apporte une information sur la classe d'un exemple

Calcul de l'entropie des exemples

- ❑ Pour choisir le premier attribut de l'arbre, nous allons choisir l'attribut qui a le plus grand gain d'information.
- ✓ Pour calculer le gain d'information, nous devons d'abord calculer l'entropie des exemples d'entraînement.

Calcul de l'entropie des exemples

Journée	Ciel	Température	Humidité	Vent	JouerTennis
J1	Ensoleillé	Chaud	Élevée	Faible	Non
J2	Ensoleillé	Chaud	Élevée	Fort	Non
J3	Nuageux	Chaud	Élevée	Faible	Oui
J4	Pluvieux	Tempérée	Élevée	Faible	Oui
J5	Pluvieux	Froide	Normal	Faible	Oui
J6	Pluvieux	Froide	Normal	Fort	Non
J7	Nuageux	Froide	Normal	Fort	Oui
J8	Ensoleillé	Tempérée	Élevée	Faible	Non
J9	Ensoleillé	Froide	Normal	Faible	Oui
J10	Pluvieux	Tempérée	Normal	Faible	Oui
J11	Ensoleillé	Tempérée	Normal	Fort	Oui
J12	Nuageux	Tempérée	Élevée	Fort	Oui
J13	Nuageux	Chaud	Normal	Faible	Oui
J14	Pluvieux	Tempérée	Élevée	Fort	Non

Gain d'information

M	Journée	Ciel	Température	Humidité	Vent	JouerTennis	E
P	J1	Ensoleillé	Chaud e	Élevée	Faible	Non	71
D	J2	Ensoleillé	Chaud e	Élevée	Fort	Non	1
S	J3	Nuageux	Chaud e	Élevée	Faible	Oui	
C	J4	Pluvieux	Tempérée	Élevée	Faible	Oui	
E	J5	Pluvieux	Froide	Normal	Faible	Oui	
E	J6	Pluvieux	Froide	Normal	Fort	Non	
E	J7	Nuageux	Froide	Normal	Fort	Oui	
J	J8	Ensoleillé	Tempérée	Élevée	Faible	Non	
E	J9	Ensoleillé	Froide	Normal	Faible	Oui	
E	J10	Pluvieux	Tempérée	Normal	Faible	Oui	
J	J11	Ensoleillé	Tempérée	Normal	Fort	Oui	
E	J12	Nuageux	Tempérée	Élevée	Fort	Oui	
E	J13	Nuageux	Chaud e	Normal	Faible	Oui	
J	J14	Pluvieux	Tempérée	Élevée	Fort	Non	

Gain d'information

Le calcul du gain d'information pour l'attribut **Ciel** va donc

$$\begin{aligned} \text{Gain}(S, \text{Ciel}) &= \text{Entropie}(S) - \sum_{v \in V(\text{Ciel})} \frac{|S_v|}{|S|} \text{Entropie}(S_v) \\ &= 0.94 - ((5/14) \times \text{Entropie}(S_{\text{ensolleillé}}) + \\ &\quad (4/14) \times \text{Entropie}(S_{\text{nuageux}}) + \\ &\quad (5/14) \times \text{Entropie}(S_{\text{pluvieux}})) \\ &= 0.94 - ((5/14) \times 0.971 + (4/14) \times 0 + (5/14) \times 0.971) \\ &= 0.94 - 0.694 \\ &= 0.246 \end{aligned}$$

Gain d'information

On calcule le gain de la même manière pour les trois autres attributs :

$$Gain(S, Ciel) = 0.246$$

$$Gain(S, Humidité) = 0.151$$

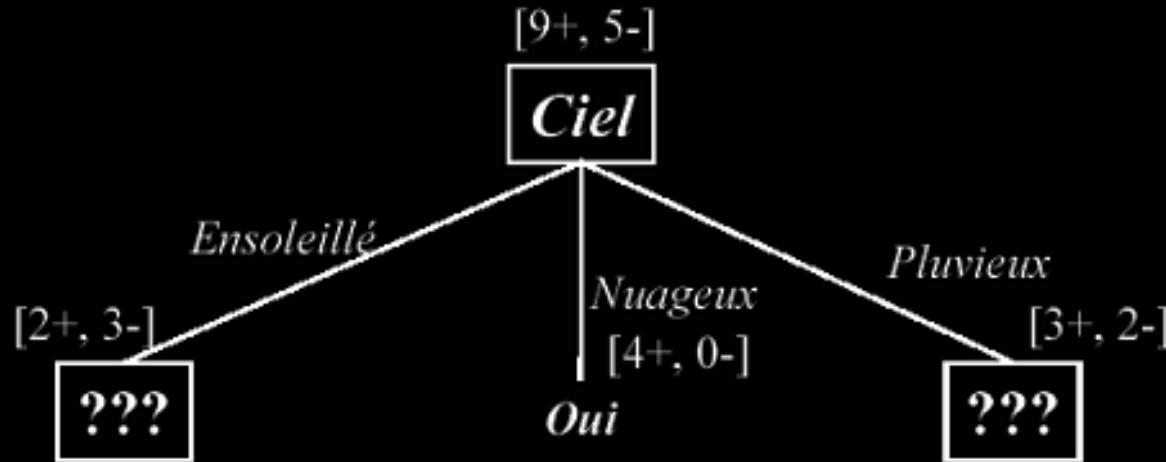
$$Gain(S, Vent) = 0.048$$

$$Gain(S, Température) = 0.029$$

L'attribut qui a la plus grand gain d'information est l'attribut **Ciel**, donc se sera la racine de l'arbre de décision.

Choix du premier attribut

En séparant les exemples selon les valeurs de l'attributs Ciel, on obtient l'arbre partiel:



- On peut voir que lorsque le ciel est nuageux, il reste uniquement des exemples positifs, donc ce nœud devient une feuille avec une valeur de Oui pour la fonction visée.
- Pour les deux autres nœuds, il y a encore des exemples positifs et négatifs, alors il faut recommencer le même calcul du gain d'information, mais avec les sous-ensembles restant.

Arbre de décision final

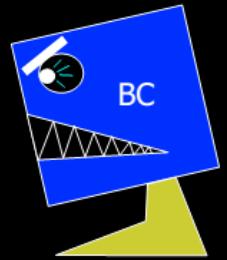
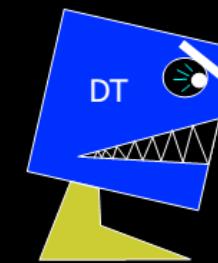
Journée	Ciel	Température	Humidité	Vent	JouerTennis
J1	Ensoleillé	Chaud e	Élevée	Faible	Non
J2	Ensoleillé	Chaud e	Élevée	Fort	Non
J3	Nuageux	Chaud e	Élevée	Faible	Oui
J4	Pluvieux	Tempérée	Élevée	Faible	Oui
J5	Pluvieux	Froide	Normal	Faible	Oui
J6	Pluvieux	Froide	Normal	Fort	Non
J7	Nuageux	Froide	Normal	Fort	Oui
J8	Ensoleillé	Tempérée	Élevée	Faible	Non
J9	Ensoleillé	Froide	Normal	Faible	Oui
J10	Pluvieux	Tempérée	Normal	Faible	Oui
J11	Ensoleillé	Tempérée	Normal	Fort	Oui
J12	Nuageux	Tempérée	Élevée	Fort	Oui
J13	Nuageux	Chaud e	Normal	Faible	Oui
J14	Pluvieux	Tempérée	Élevée	Fort	Non

L'attr
n'app
car il

le
IS

Classificateurs de Bayes

Formidable et ennemi juré
des arbres de décision



$$Y^{\text{predict}} = \underset{v}{\operatorname{argmax}} P(X_1 = u_1 \cdots X_m = u_m \mid Y = v) P(Y = v)$$

Classificateur Naïf Bayes

Dans le cas du classificateur naïf de Bayes (indépendance), cela peut être simplifié:

$$\text{Proba. a posteriori} \quad P(Y|X) = \frac{P(X|Y)P(Y)}{P(X)}$$

$$Y^{\text{predict}} = \underset{v}{\operatorname{argmax}} P(Y = v) \prod_{j=1}^{n_Y} P(X_j = u_j \mid Y = v)$$

Conseil technique:

Si vous avez 10 000 attributs d'entrée, ce produit sera débordement en calcul en virgule flottante. Vous devriez utiliser les journaux:

$$Y^{\text{predict}} = \underset{v}{\operatorname{argmax}} \left(\log P(Y = v) + \sum_{j=1}^{n_Y} \log P(X_j = u_j \mid Y = v) \right)$$

Classificateurs de Bayes Exemple (Problème)

Soit l'ensemble de données d'entraînement sur la météo et la variable cible correspondante «Play».

Objectif: Classer si les joueurs vont jouer ou non en fonction des conditions météorologiques.

□ Étape 1: convertissez le jeu de données en table de fréquences

□ Étape 2: Créez un tableau de probabilité en recherchant les probabilités telles que Probabilité de overcast= 0,29 et probabilité de play de 0,64.

□ Étape 3: Maintenant, utilisez l'équation naïve bayésienne pour calculer la probabilité a posteriori pour chaque classe.

La classe avec la probabilité postérieure la plus élevée est le résultat de la prédiction.

Weather	Play
Sunny	No
Overcast	Yes
Rainy	Yes
Sunny	Yes
Sunny	Yes
Overcast	Yes
Rainy	No
Rainy	No
Sunny	Yes
Rainy	Yes
Sunny	No
Overcast	Yes
Overcast	Yes
Rainy	No

Frequency Table		
Weather	No	Yes
Overcast		4
Rainy	3	2
Sunny	2	3
Grand Total	5	9

Likelihood table				
Weather	No	Yes		
Overcast		4	=4/14	0.29
Rainy	3	2	=5/14	0.36
Sunny	2	3	=5/14	0.36
All	5	9		
	=5/14	=9/14		
	0.36	0.64		

Problème: les joueurs vont jouer si le temps est ensoleillé (Sunny), cette déclaration est-elle correcte?

Classificateurs de Bayes Exemple (solution)

$$P(\text{Oui} \mid \text{Ensoleillé}) = P(\text{Ensoleillé} \mid \text{Oui}) * P(\text{Oui}) / P(\text{Ensoleillé})$$

Nous avons ici

$$P(\text{Ensoleillé} \mid \text{Oui}) = 3/9 = 0,33,$$

$$P(\text{Ensoleillé}) = 5/14 = 0,36,$$

$$P(\text{Oui}) = 9/14 = 0,64$$

Donc , $P(\text{Oui} \mid \text{Ensoleillé}) = 0,33 * 0,64 / 0,36 = 0,60$,
qui a une probabilité plus élevée.

- Naïve Bayes utilise une méthode similaire pour prédire la probabilité de différentes classes en fonction de divers attributs. Cet algorithme est principalement utilisé dans la classification de texte, lorsqu'il est difficile d'avoir plusieurs classes...

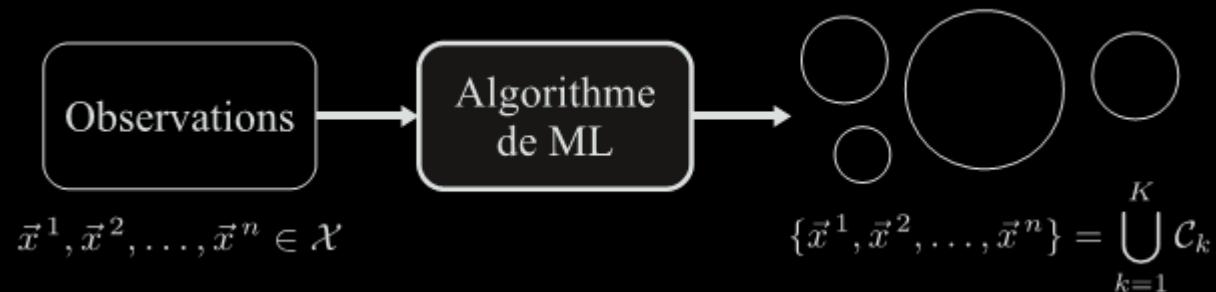
Apprentissage non supervisé : Clustering & K-Means



Clustering

Les objectifs du Clustering

Le Clustering a pour objectif de déterminer le regroupement intrinsèque dans un ensemble de données non étiquetées.



Mais comment décide-t-on de ce qui constitue un bon regroupement?

On peut montrer qu'il n'existe pas de critère de «meilleur» absolu, indépendant du but final du clustering. Par conséquent, c'est l'utilisateur qui doit satisfaire ce critère, de manière à ce que le résultat du regroupement réponde à ses besoins.

Clustering

- Le Clustering a été considéré comme le plus important problème d'apprentissage non supervisé.
- Traite la recherche de structure dans des données non étiquetées c'est-à-dire que, contrairement à l'apprentissage supervisé, les données cibles ne sont pas fournies
- **Principe:**
Le Clustering est «le processus d'organisation des objets dans des groupes dont on se souvient».
- Un concepteur reconnaît que les objets sont «similaires» entre eux et sont «dissemblables» par rapport aux objets appartenant à d'autres groupes.

Les applications les plus courantes:

Marketing : recherche de groupes de clients ayant un comportement similaire à partir d'une base de données volumineuse contenant les propriétés des clients et des enregistrements de leurs achats.

Bio logé : classification des plantes et des animaux en fonction de leurs caractéristiques

Assurance : détection de fraude

Urbanisme: identification des groupes d'habitation en fonction de leur type, de leur valeur et de leur situation géographique;

Études sur les tremblements de terre: regroupement des épicentres

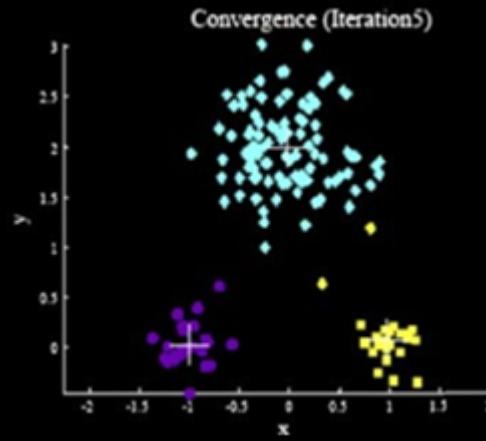
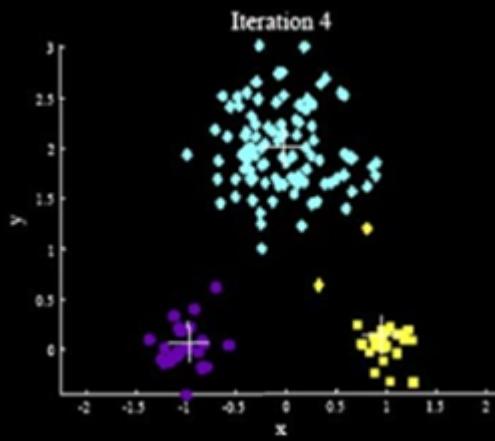
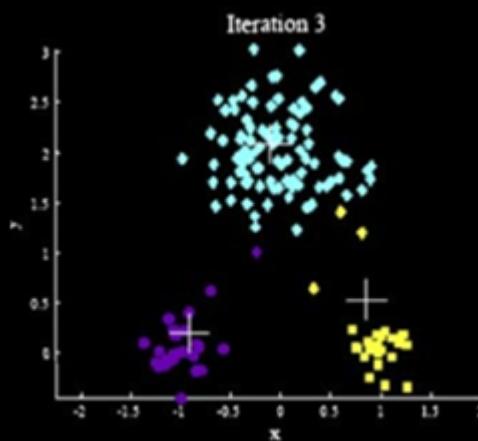
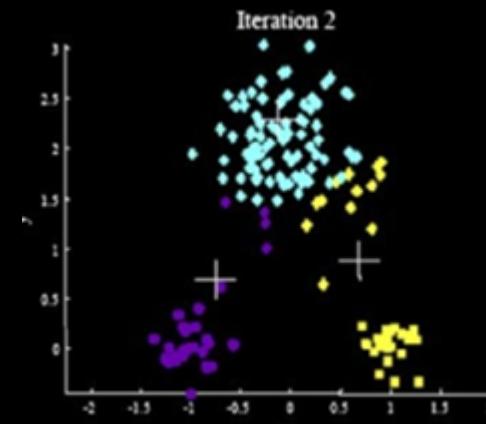
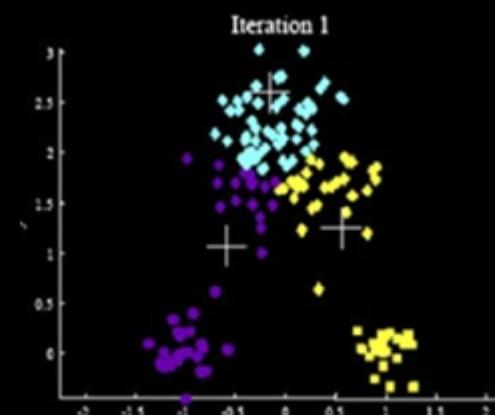
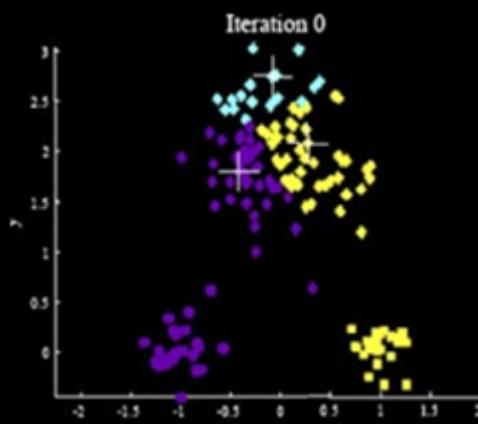
de séisme Identifier les zones

dangereuses;

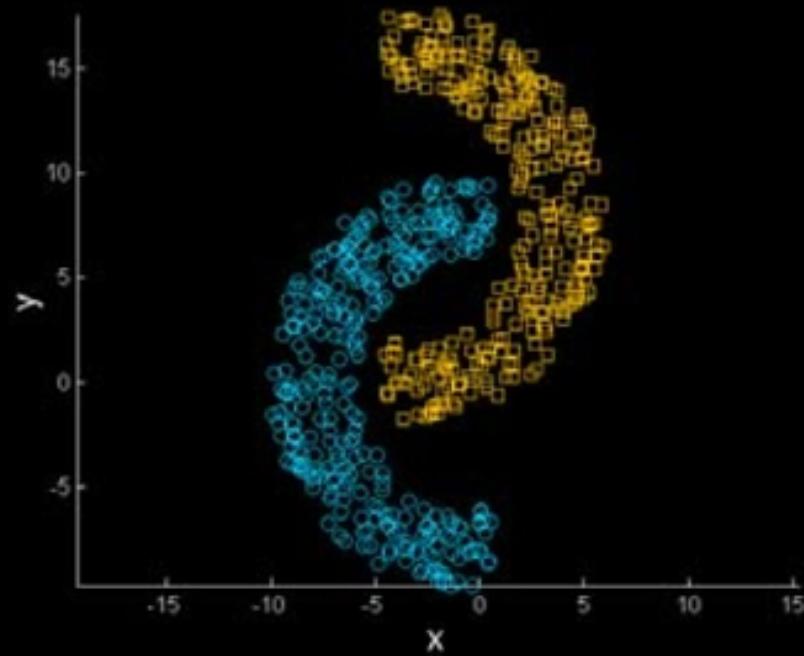
WWW: classification des documents; regrouper les données de flux de clics pour découvrir des groupes de noms d'accès similaires.

Algorithme de K-moyennes (K-means algorithm)

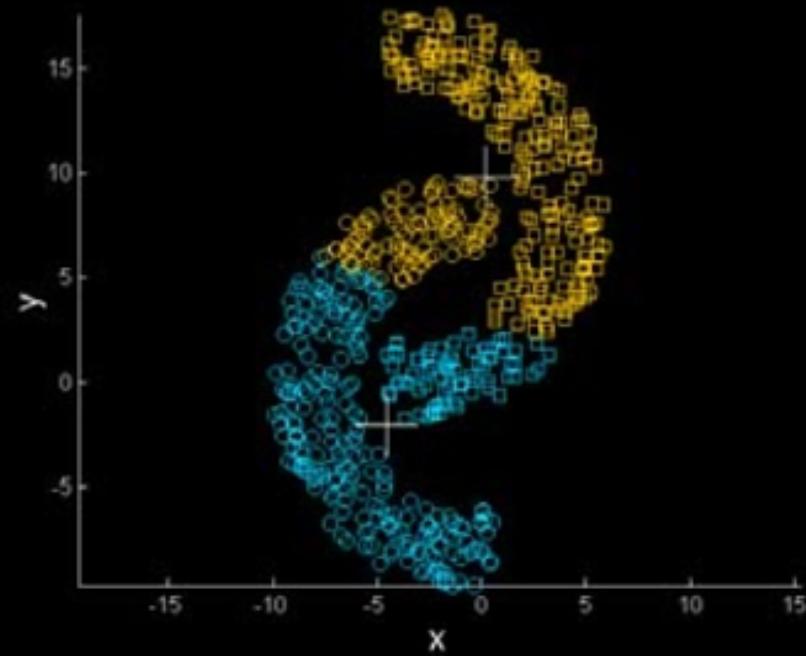
1. Choisissez un nombre (k) de centres de cluster
2. Attribuer chaque point (exemple) au centre de son cluster le plus proche
3. Déplacez le centre de chaque cluster à la moyenne de ses points assignés
4. Répétez les étapes 2 et 3 jusqu'à la fin



Limitation du K-means



Données originales



Clustering $k=2$

Les réseaux de neurones

Motivation:

Les animaux apprennent et apprennent dans le cerveau

Si nous pouvons comprendre le fonctionnement du cerveau, il y a probablement des choses que nous pouvons copier et utiliser pour notre système d'apprentissage automatique.

Le cerveau est extrêmement complexe et incroyablement puissant, mais les blocs de construction atomiques de base sont simples et faciles à comprendre.

Le cerveau fait exactement ce que nous voulons. Il traite les données bruitées et incohérentes et produit des réponses qui sont généralement issues de données très dimensionnelles.

(comme des images) très rapidement



Motivation:

Les unités de traitement de base du cerveau sont des **neurones**

Chaque neurone peut être considéré comme un processeur. Chacun effectue un calcul très simple: décider activé ou non.

Le cerveau est donc un ordinateur massivement parallèle composé de milliards de "processeurs".

Comment l'apprentissage est fait dans le cerveau?

Plasticité (Souplesse) : modifier la force de connexion entre neurones et créer de nouvelles connexions



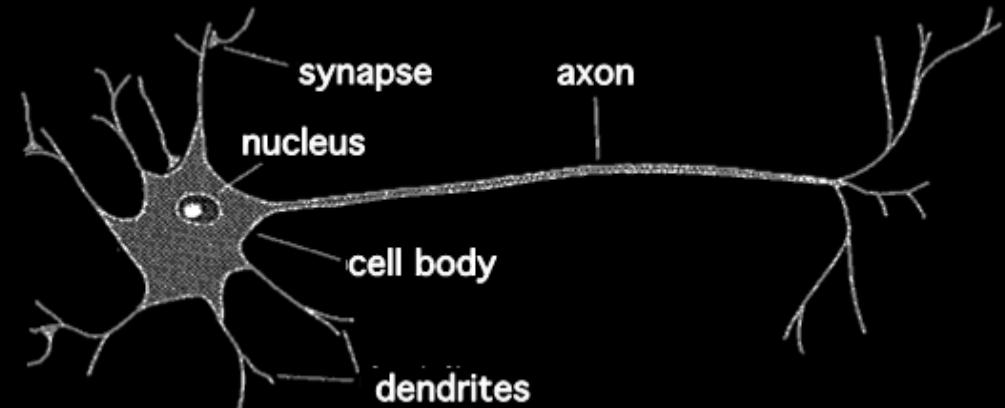
Règle de Hebs:

"Les changements dans la force des connexions inter-neuronales (**synaptiques**) sont proportionnels à la corrélation dans le déclenchement des deux neurones qui se connectent."

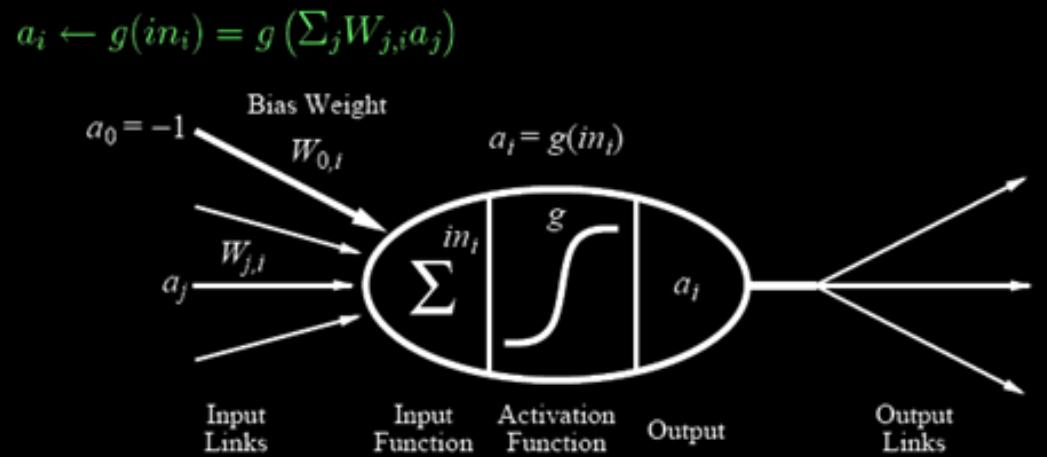
Fondamentalement: «Les neurones qui s'activent ensemble se connectent»

Réseaux de neurones artificiels

Motivation: cerveau humain massivement parallèle (10^{11} neurones, ~ 20 types) petites unités de calcul avec une communication simple à faible bande passante (10^{14} synapses, temps de cycle 1-10ms)



Réalisation: réseau de neurones unités (\approx neurones) connectées par des liaisons pondérées dirigées fonction d'activation des entrées aux sorties *units*

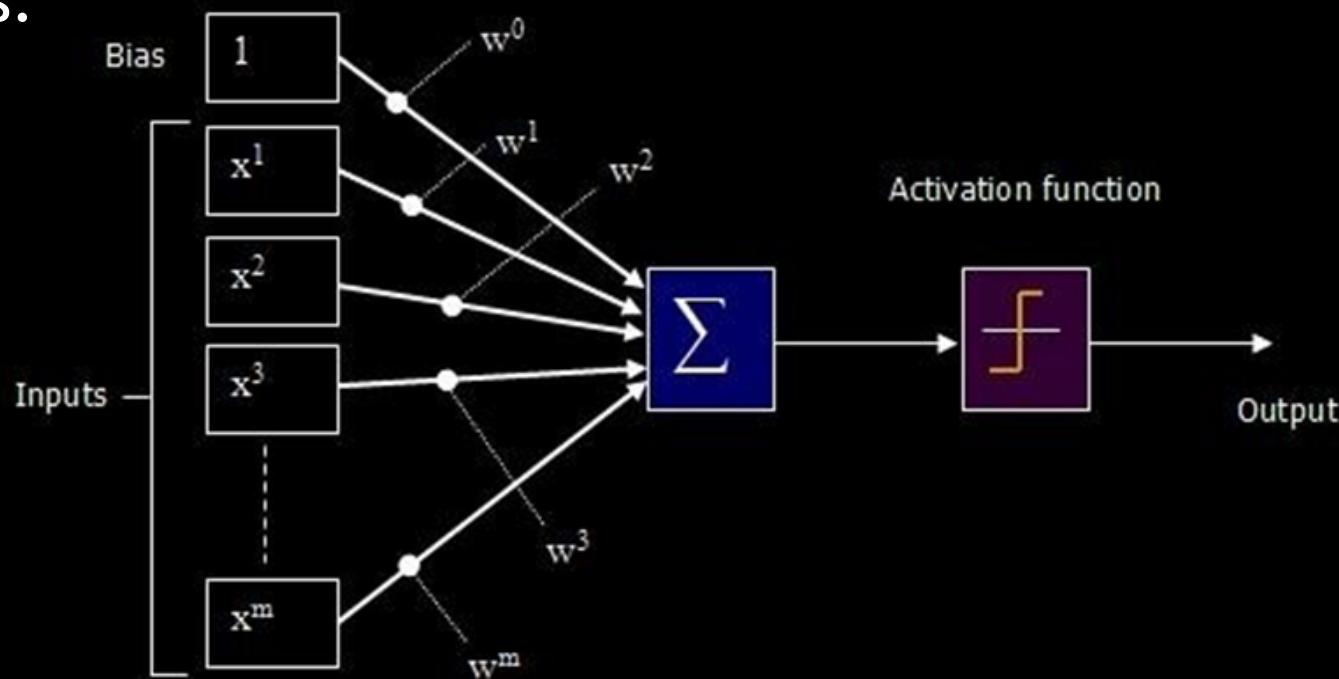


Brief History of Neural Networks

- 1943: neurones machine de Turing
- 1958: perceptrons
- 1969: limites
- 1985: Geoff Hinton et al
- 2006:

Classification (Le Perceptron)

Notre objectif est de trouver le poids w_i capable de classer parfaitement les entrées positives et négatives dans nos données.



$$y = 1 \quad if \sum_{i=0}^n w_i * x_i \geq 0$$

$$= 0 \quad if \sum_{i=0}^n w_i * x_i < 0$$

where, $x_0 = 1$ and $w_0 = -\theta$

Algorithm: Perceptron Learning Algorithm

P \leftarrow inputs with label 1;
N \leftarrow inputs with label 0;
Initialize \mathbf{w} randomly;
while !convergence **do**
 | Pick random $\mathbf{x} \in P \cup N$;
 | **if** $\mathbf{x} \in P$ and $\mathbf{w} \cdot \mathbf{x} < 0$ **then**
 | | $\mathbf{w} = \mathbf{w} + \mathbf{x}$;
 | **end**
 | **if** $\mathbf{x} \in N$ and $\mathbf{w} \cdot \mathbf{x} \geq 0$ **then**
 | | $\mathbf{w} = \mathbf{w} - \mathbf{x}$;
 | **end**
end
//the algorithm converges when all the
inputs are classified correctly

Resources:

Datasets:

UCIRepo:<http://archive.ics.uci.edu/ml/>

Discussions

MachineLearningsubReddit:<http://www.reddit.com/r/MachineLearning/>

Books:

Pattern Clacification (Duda, Hart)

MachineLearning(Tom Mitchell)

IntroductiontoMachineLearning(Ethem Alpaydin)

NeuralNetworks(Simon Haykin)

MachineLearning:analgorithmicapproach(Marsland)

Merci

?

Exercice

Alice veut écrire un programme qui utilise la fréquence des mots « science », « public », « accès », « université », «gouvernement », « financer », « éducation », « budget », « justice »et « loi » pour déterminer si un article traite ou non de politique scientifique. Elle a commencé par annoter un millier d'articles selon leur sujet. Quel genre de problème d'apprentissage automatique doit-elle résoudre ?

Exercice (2)

Parmi les problèmes suivants, lesquels se prêtent bien à être traités par le machine learning ?

1. Déterminer l'horaire optimal pour poster un contenu sur une page web.
2. Déterminer le chemin le plus court entre deux nœuds dans un graphe.
3. Prédire le nombre de vélos à mettre en location à chaque station d'un système de location de vélos citadins.
4. Évaluer le prix qu'un tableau de maître pourra atteindre lors d'une vente aux enchères.
5. Débruiter un signal radio.

Exercice(3)

Benjamin dispose de 10 000 articles de journaux qu'il souhaite classer par leur thématique. Doit-il utiliser un algorithme supervisé ou non supervisé ?

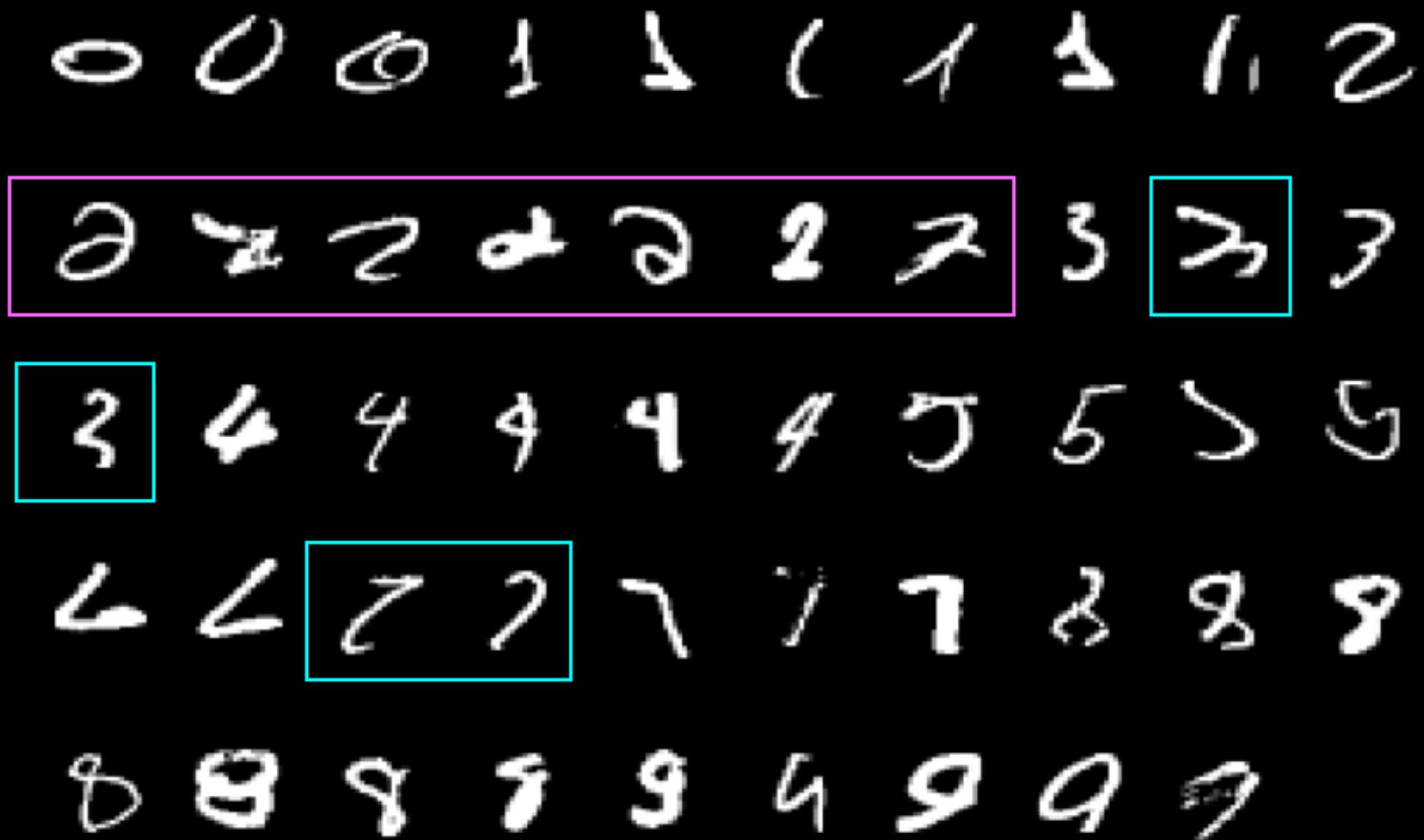
Les données de Cécile sont décrites par 10 variables. Elle aimeraient cependant les représenter sur un graphique en deux dimensions. Quel type d'algorithme d'apprentissage doit-elle utiliser ?

David gère un outil qui permet d'organiser les liens HTML qui ont été sauvegardés. Il souhaite suggérer des catégories auxquelles affecter un nouveau lien, en fonction des catégories déjà définies par l'ensemble des utilisateurs du service. Quel type d'algorithme d'apprentissage doit-il utiliser ?

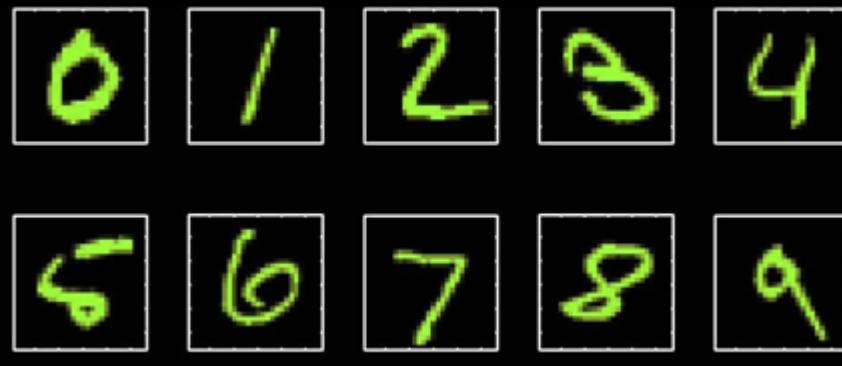
Exercice (4)

- 1.6 Elsa veut examiner ses spams pour déterminer s'il existe des sous-types de spams. Quel type d'algorithme d'apprentissage doit-elle utiliser ?
- 1.7 Tom Mitchell définit le machine Learning comme suit : « *Un programme informatique est dit apprendre de l'expérience E pour la tâche T et une mesure de performance P si sa performance sur T, comme mesurée par P, s'améliore avec l'expérience E* ». Fred écrit un programme qui utilise des données bancaires dans le but de détecter la fraude bancaire. Que sont E, T, et P ?

Exemple (Classification) : reconnaissance des chiffres/lettres manuellement écrites



Exemple (Classification) : reconnaissance des chiffres



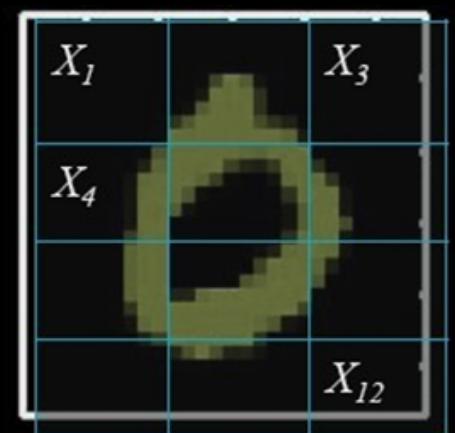
$$\begin{pmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ \vdots \\ 0 \end{pmatrix} = 1$$

Entrée(X_i):
caractéristiques de l'image

Sortie(Y): étiquettes $\{ [y^0, y^1, \dots, y^9]^T / y^i = 0 \text{ ou } 1 \}$

caractéristique(X_i):

Proportion de pixels dans le carré X_i
où $i=1, 2, \dots, 12$



Lors d'un apprentissage par cœur, il faut mémoriser 10^{12} configurations par chiffre!

Définitions

On dit qu'un **programme informatique** «apprend» de l'expérience **E** pour certaine classes de tâche **T** et une mesure de performance **P**, si ses performances en tâche de **T**, mesurées par **P**, sont améliorées avec l'expérience **E**.

Tom Mitchell (1997)

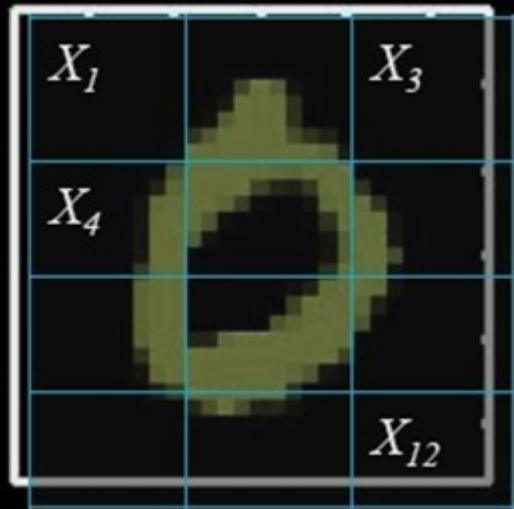
Donc, selon cette définition

Tâche (**T**): reconnaître et classer des mots écrits à la main dans des images

Mesure de performance (**P**): pourcentage de mots correctement classés

Expérience d'entraînement (**E**): une base de données de manuscrits avec des classifications données

10^{12} configurations par chiffre!



Nous avons besoin de ML dans les cas où nous ne pouvons pas écrire directement un programme pour traiter chaque cas.

Donc, il est préférable d'avoir une machine qui apprend d'un grand ensemble d'entraînement