IFT 615 – Intelligence Artificielle

Été 2022

Formes d'apprentissage

Algorithme des K plus proches voisins

Professeur: Froduald Kabanza

Assistants: D'Jeff Nkashama & Jean-Charles Verdier

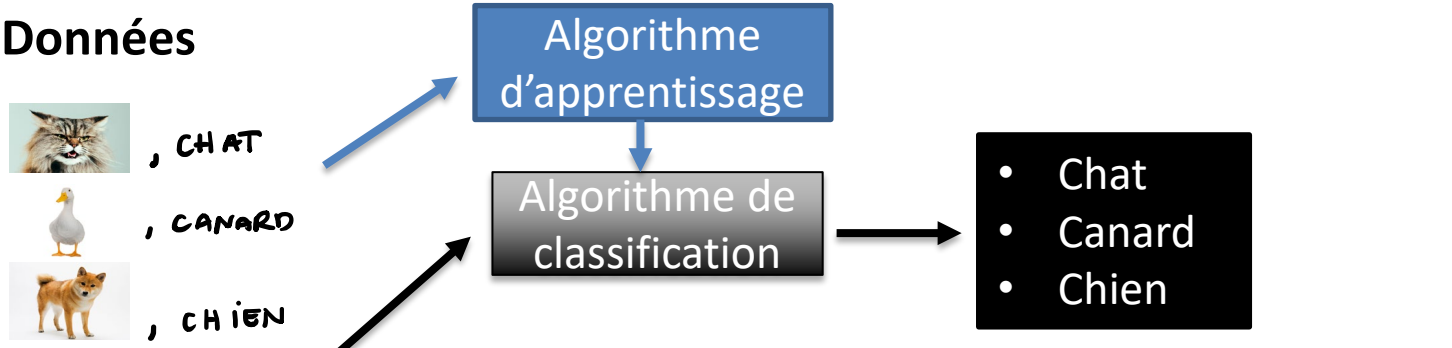
UDS Université de Sherbrooke

Sujets couverts

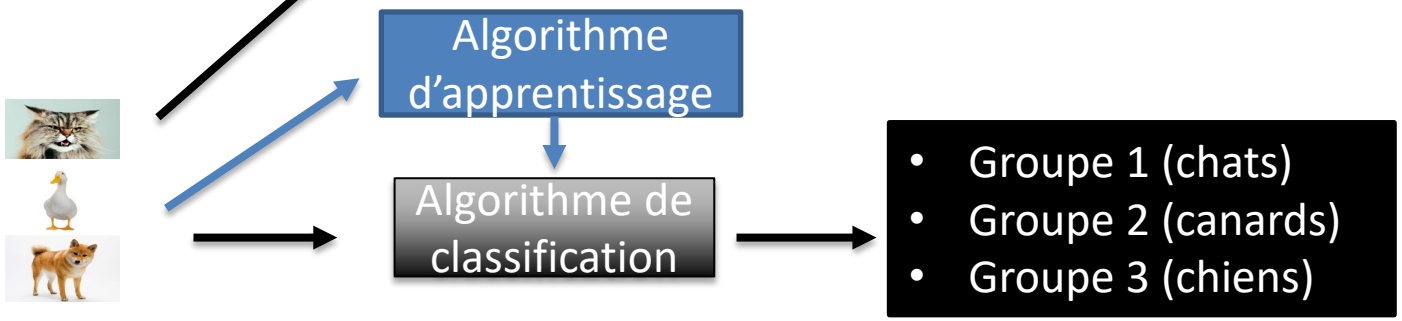
- Formes d'apprentissage
- Classification avec l'algorithme des K plus proches voisins

Types de problèmes d'apprentissage

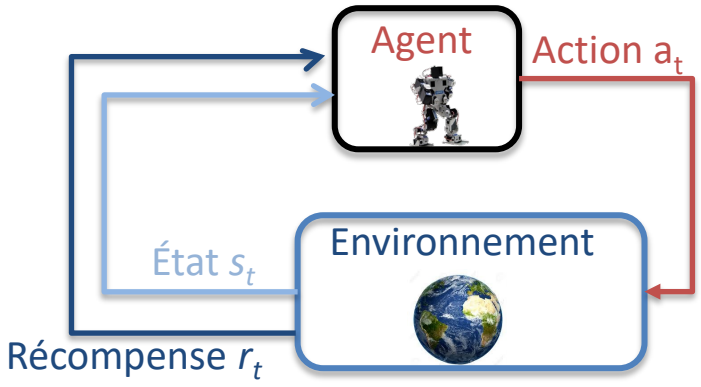
Apprentissage supervisé



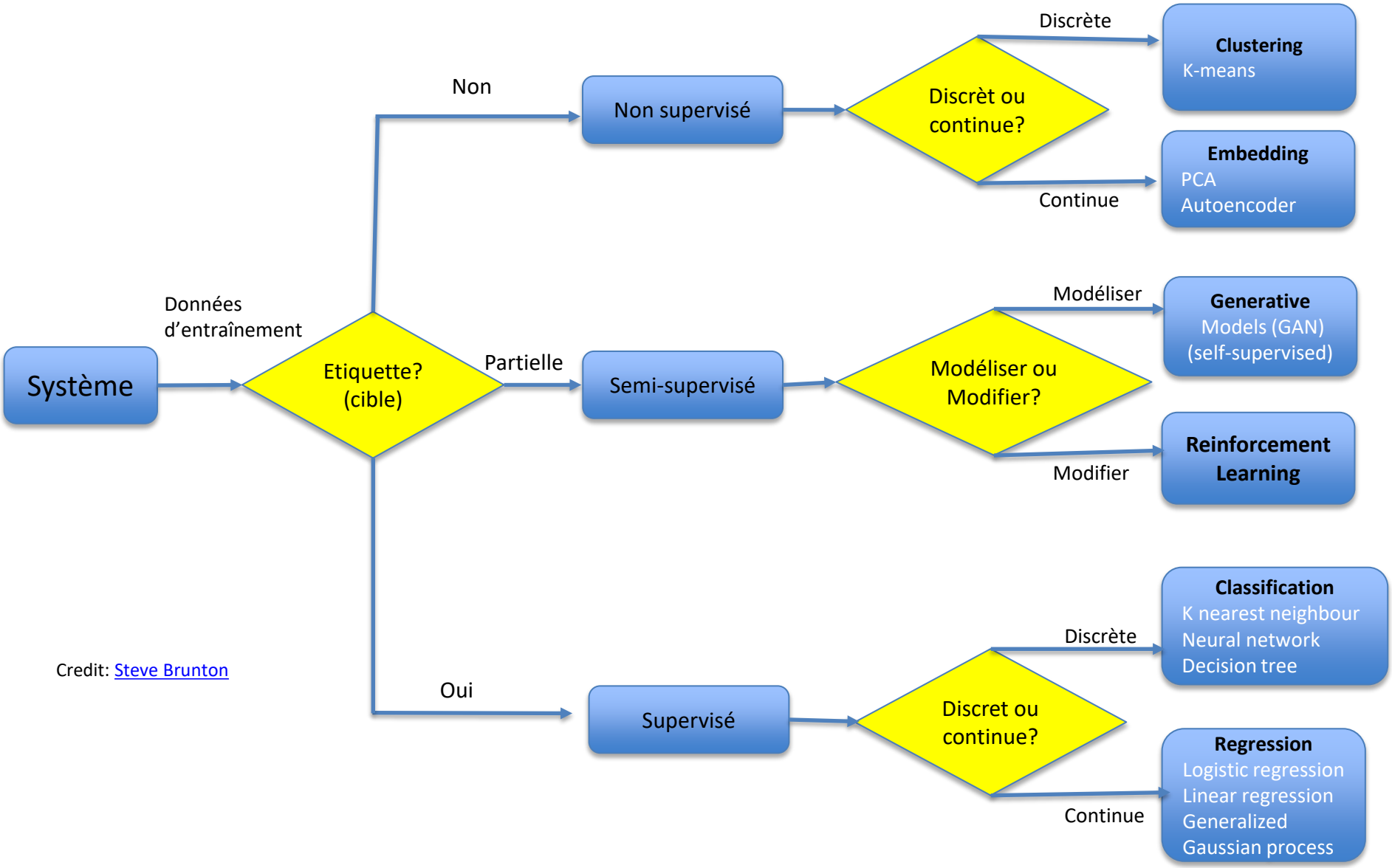
Apprentissage non supervisé



Apprentissage par renforcement



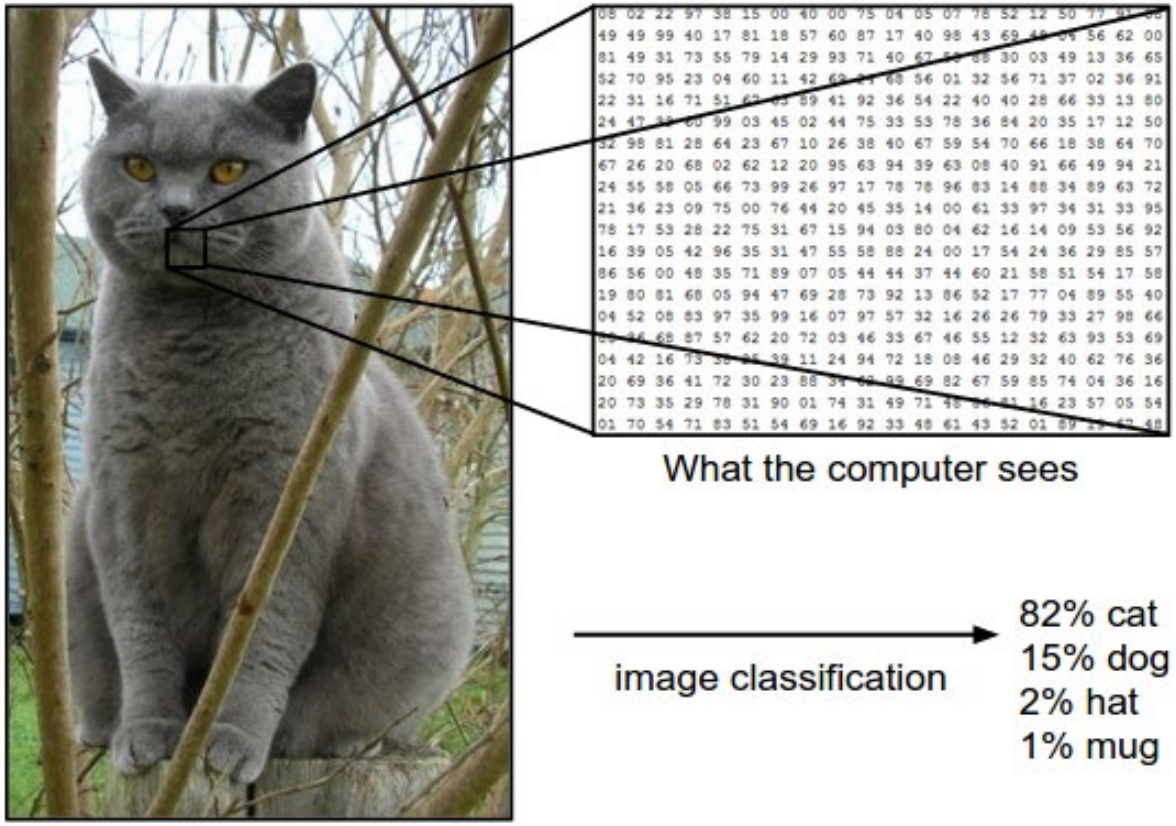
Types de problèmes d'apprentissage



Credit: Steve Brunton

APPRENTISSAGE SUPERVISÉE

Motivation – Classification d'images



Credit : Stanford cs231

Motivation – Classification d'images

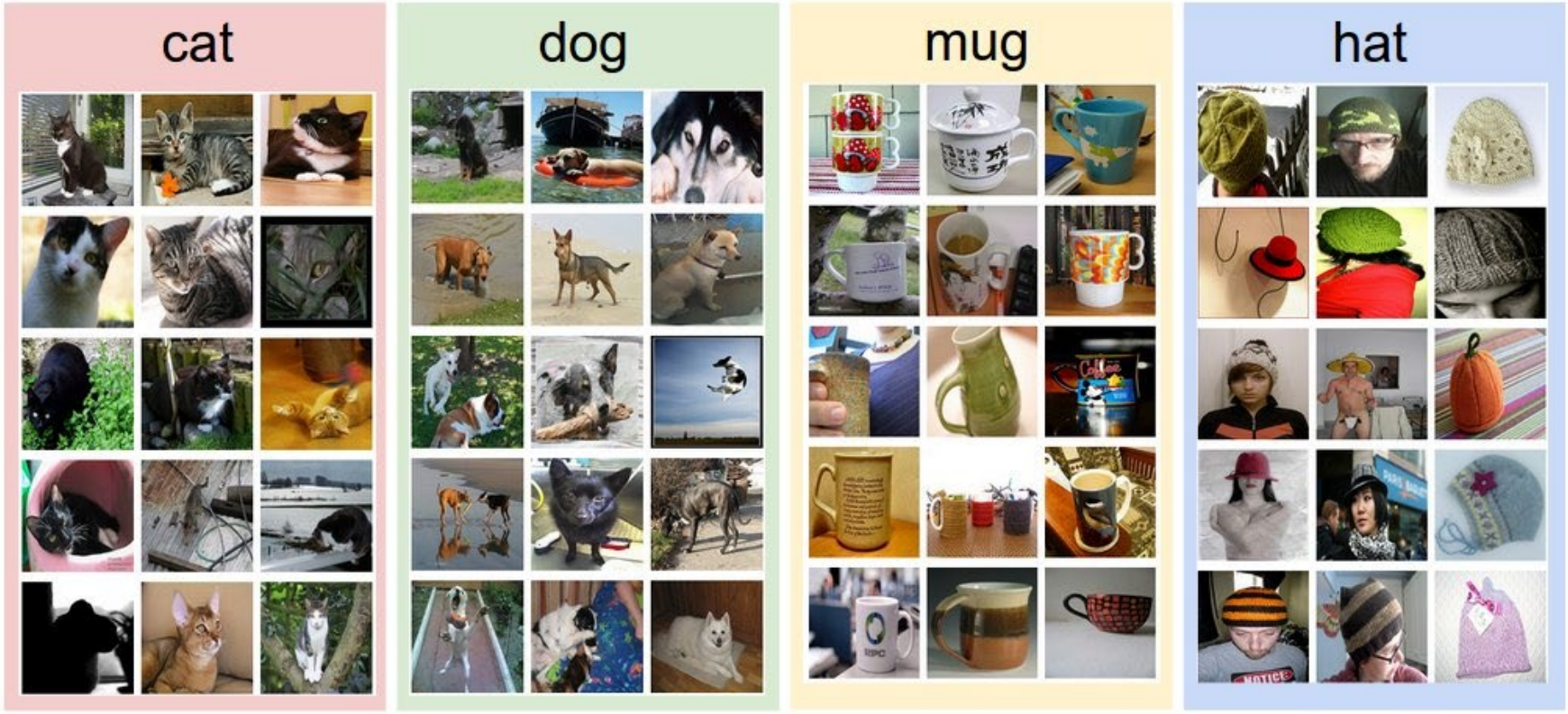
Principales étapes

- **Éntrée** : N images, chacune étiquetée par l'une des K classes. C'est l'ensemble de **données d'entrainement**.

- **Apprentissage** : On utilise les données d'entrainement pour apprendre à quoi ressemble chacune des classe. C.à-d., on entraine un classifieur; autrement dit, on **apprend le modèle**.

- **Évaluation**: Finalement, on évalue la qualité du classifieur en lui demandant de prédire les étiquettes pour un ensemble d'images vues pour la première fois.

Motivation – Classification d'images



Credit: Stanford C231

APPRENTISSAGE SUPERVISÉ : FORMULATION DU PROBLÈME

Apprentissage supervisé

- Un problème d'apprentissage supervisé est formulé comme suit:

  « Étant donné un **ensemble d'entraînement** de N exemples:

$$\underbrace{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_N, y_N)}_{D}$$

  où chaque y_j a été généré par une **fonction inconnue** $y = f(\mathbf{x})$,
  découvrir une nouvelle fonction h (**modèle** ou **hypothèse**)
  qui sera une bonne approximation de f (c'est à dire $f(\mathbf{x}) \approx h(\mathbf{x})$) »

Apprentissage supervisé

- Étant donné un ensemble de données d'entrainement, l'apprentissage est un problème de **recherche** de l'hypothèse *h* **dans un espace d'hypothèses *H***, tel que *h* minimise la **distance** à *f(x)*

- Les données sont souvent bruités et disponibles en quantité limitée. Il y a donc une variation dans les données et dans les modèles (représentations).

- L'erreur dépend de la qualité des données d'entrainements et de la méthode utilisée pour sélectionner/chercher la bonne hypothèse

Apprentissage supervisé

- Données : **ensemble d'entraînement** de N exemples:

$$\underbrace{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_N, y_N)}_{D} \quad \text{avec} \quad y = f(\mathbf{x})$$

- Problème : trouver *h(x)* tel que $f(\mathbf{x}) \approx h(\mathbf{x})$

- Un algorithme d'apprentissage peut donc être vu comme étant une fonction A à laquelle on donne un ensemble d'entraînement et qui donne en retour la fonction *h*

$$A(D) = h$$

Modèles et approche pour chercher la fonction *h*

- Dépendamment des approches d'apprentissage, la fonction *h* peut être représentée de différente manière.

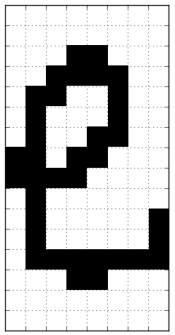
- Dans ce cours, nous voyons:
  - K plus proches voisins
  - Perceptron
  - Régression logistique
  - Réseau de neurones
  - Arbre de décision

- Le livre couvre différentes autres approches

REPRÉSENTATION DES DONNÉES

Représentation des données

- L'**entrée** $\mathbf{x}$ est représentée par un vecteur de valeurs d'attributs réels (représentation factorisée)
  - ex.: une image est représentée par un vecteur contenant la valeur de chacun des pixels

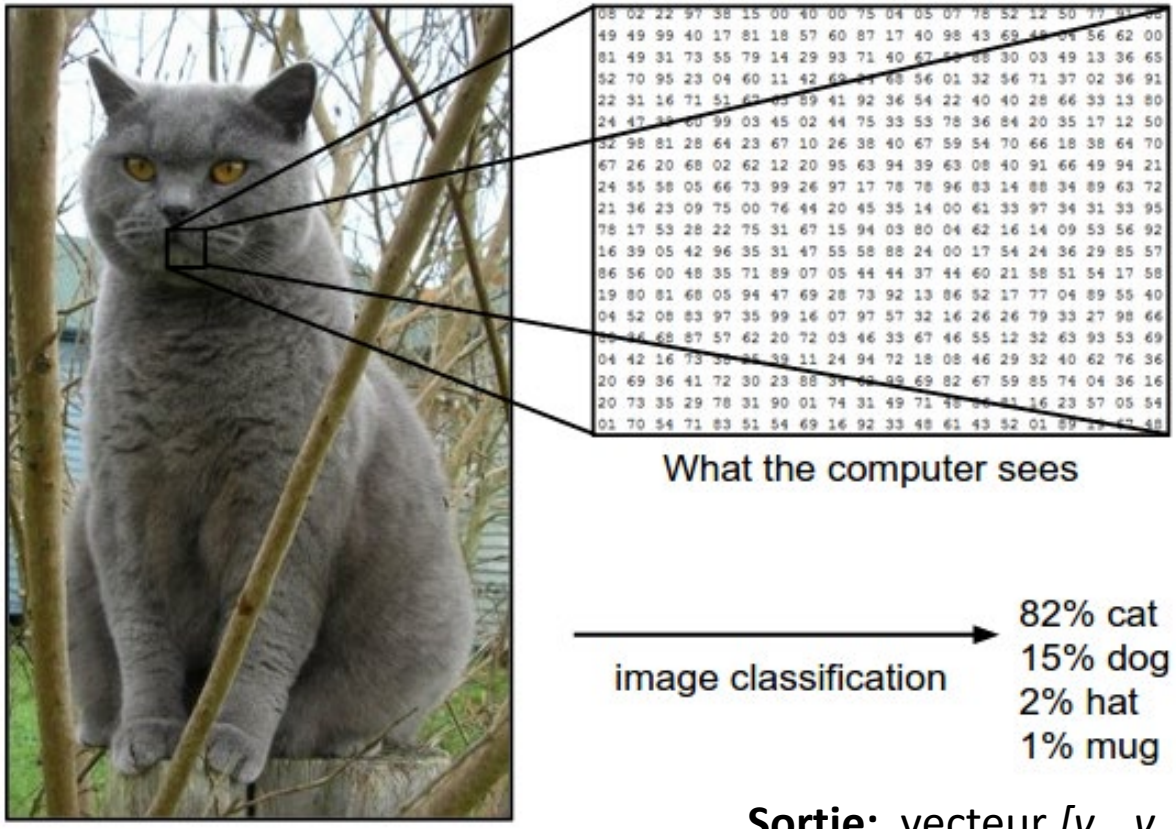


```
array([ 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
        0., 0., 0., 0., 0., 0., 1., 1., 0., 0., 0., 0.,
        1., 1., 1., 1., 0., 0., 0., 1., 1., 0., 0., 1., 0.,
        0., 0., 1., 0., 0., 0., 1., 0., 0., 0., 1., 0., 0.,
        1., 1., 0., 0., 1., 1., 0., 1., 1., 0., 0., 0., 1.,
        1., 1., 1., 0., 0., 0., 0., 0., 1., 0., 0., 0., 0.,
        0., 0., 0., 1., 0., 0., 0., 0., 0., 1., 0., 1., 0.,
        0., 0., 0., 0., 1., 0., 1., 1., 1., 1., 1., 1., 1.,
        0., 0., 0., 1., 1., 0., 0., 0., 0., 0., 0., 0., 0.,
        0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.])
```

- La **sortie désirée** ou **cible** y aura une représentation différente selon le problème à résoudre:
  - problème de classification en *C* classes: valeur discrète (index de 0 à *C*-1)
  - problème de régression: valeur réelle ou continue

Exemple – Classification d'images

Entrée : X = vecteur de pixels



http://cs231n.github.io/assets/classify.png

Sortie: vecteur $[y_1, y_2, y_3, y_4]$

y_i : réel sur l'interval [0,1]

K PLUS PROCHES VOISINS

Exemple: classifieur *k* plus proches voisins

- Possiblement l'algorithme d'apprentissage de classification le plus simple
- **Idée**: étant donnée une entrée $\mathbf{x}$
  1. trouver les *k* entrées $\mathbf{x}_t$ parmi les exemples d'apprentissage qui sont les plus « proches » de $\mathbf{x}$
  2. faire voter chacune de ces entrées pour leur classe associée y_t
  3. retourner la classe majoritaire
- Le succès de cet algorithme va dépendre de deux facteurs
  - la quantité de données d'entraînement (plus il y en a, meilleure sera la performance)
  - la qualité de la mesure de distance (est-ce que deux entrées jugées similaires sont de la même classe?)
    - » en pratique, on utilise souvent la distance Euclidienne:

$$d(\mathbf{x}_1, \mathbf{x}_2) = \sqrt{\sum_k (x_{1,k} - x_{2,k})^2}$$

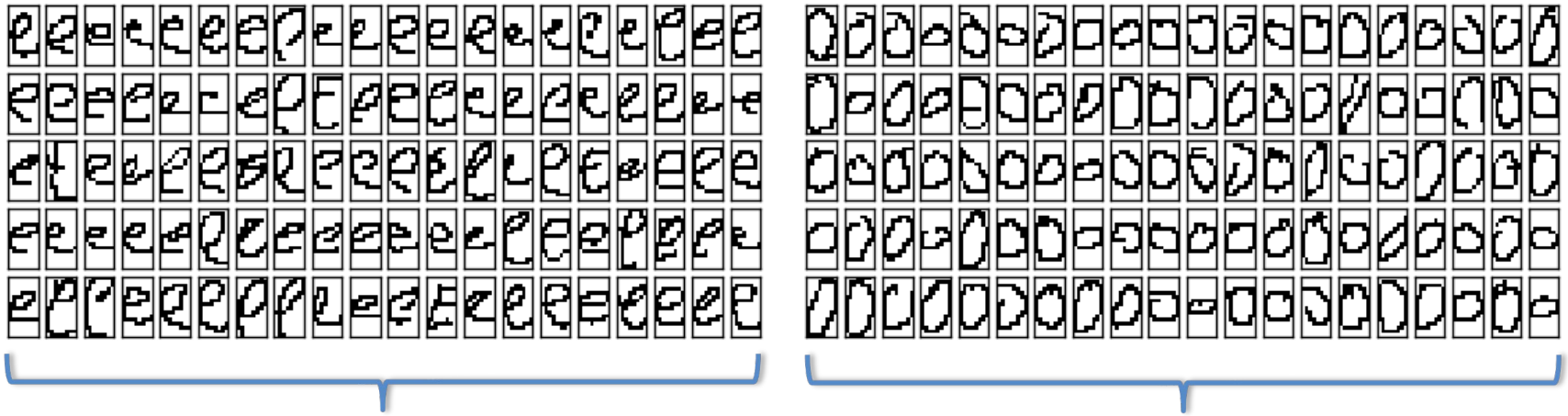
$\mathbf{x}$ = vecteur
x = scalaire

Illustration: 3 plus proches voisins

- Reconnaissance de caractère: est-ce un ‘e’ ou un ‘o’?

Ensemble d’entraînement

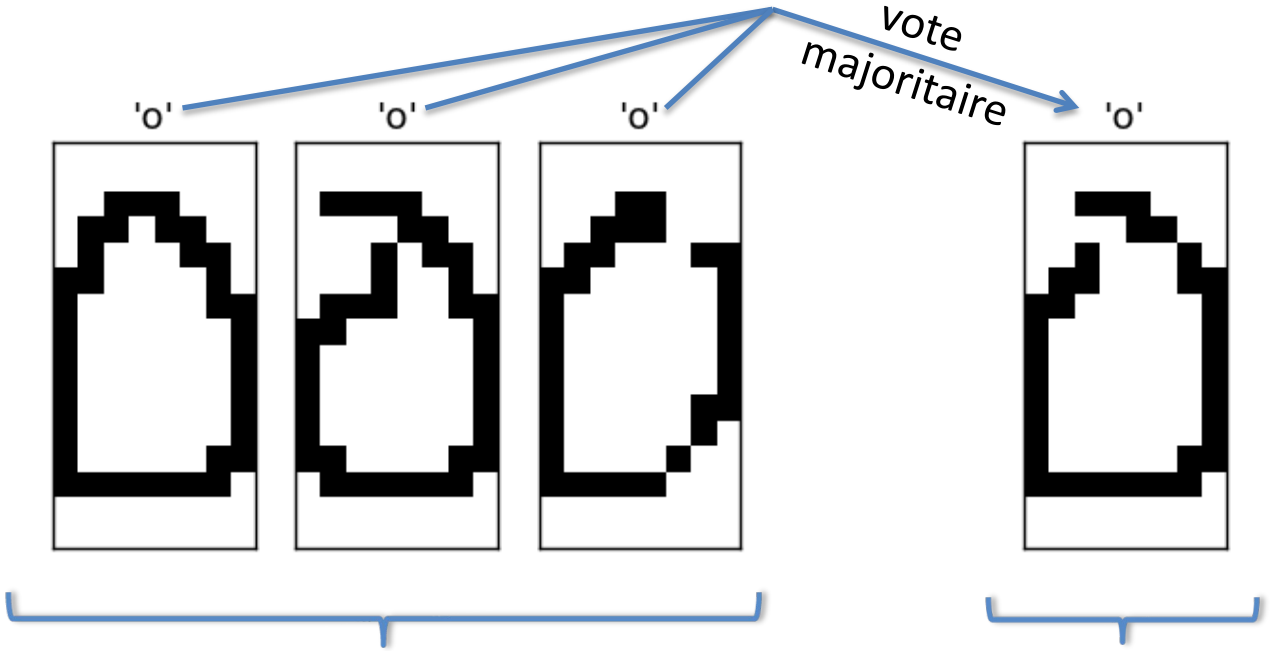
(100 exemples d’apprentissage par classe)



Classe ‘e’ Classe ‘o’

Illustration: 3 plus proches voisins

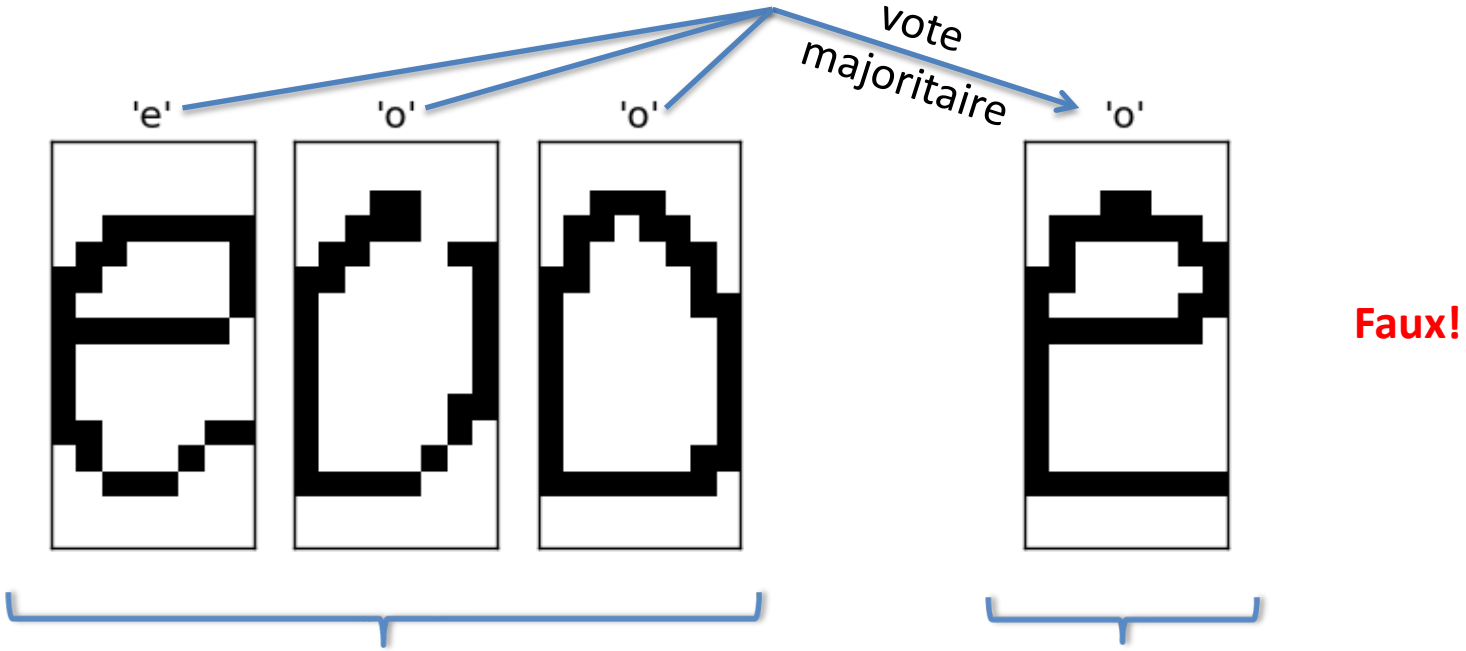
- Reconnaissance de caractère: est-ce un ‘e’ ou un ‘o’?



Vrai!

Illustration: 3 plus proches voisins

- Reconnaissance de caractère: est-ce un ‘e’ ou un ‘o’?



Faux!

Rappel - Apprentissage supervisé

- Un problème d'apprentissage supervisé est formulé comme suit:

  « Étant donné un **ensemble d'entraînement** de N exemples:

  $$\underbrace{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_N, y_N)}_{D}$$

  où chaque y_j a été généré par une **fonction inconnue** $y = f(\mathbf{x})$,
  découvrir une nouvelle fonction h (**modèle** ou **hypothèse**)
  qui sera une bonne approximation de f (c'est à dire $f(\mathbf{x}) \approx h(\mathbf{x})$) »

- Un algorithme d'apprentissage peut donc être vu comme étant une fonction A à laquelle on donne un ensemble d'entraînement et qui donne en retour cette fonction h

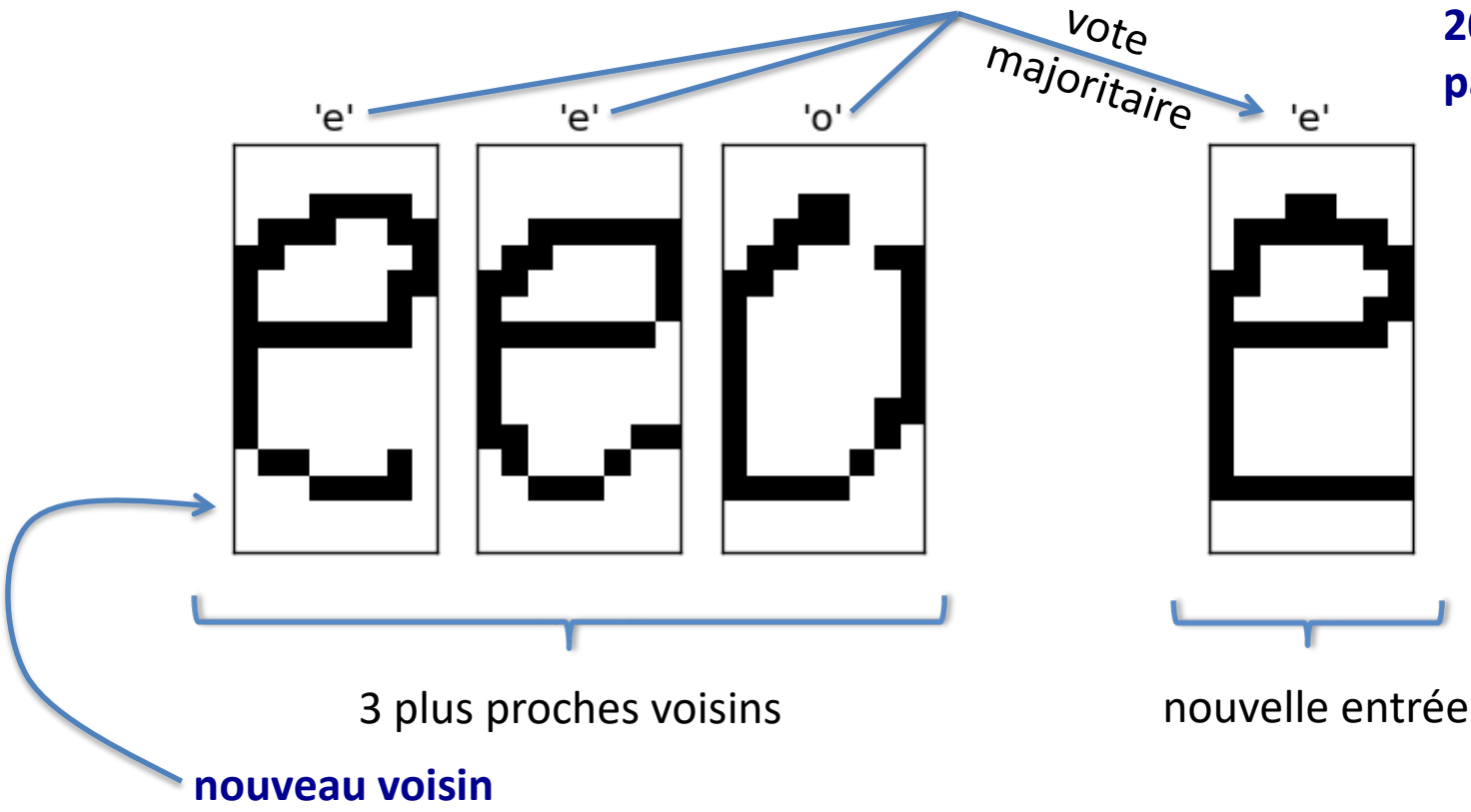
$$A(D) = h$$

Retour sur classifieur *k* plus proches voisins

- Dans le cas de l'algorithme *k* plus proches voisins:
  - A est un programme qui produit lui-même un programme, soit celui qui fait une prédiction à l'aide de la procédure *k* plus proches voisins
  - $h = A(D)$ est le programme qui fait voter les *k* plus proches voisins dans D d'une entrée donnée
  - $h(\mathbf{x})$ est la sortie du programme pour l'entrée $\mathbf{x}$, c'est à dire une prédiction de la classe de $\mathbf{x}$
  - f est la « fonction » qui a généré nos données d'entraînement
    - » ex.: l'être humain qui a étiqueté les images de caractères
- On peut démontrer que plus D est grand, plus h sera une bonne approximation de f
  - intuition: en augmentant la taille de l'ensemble d'entraînement, les *k* plus proches voisins ne peuvent changer qu'en étant encore plus proches (plus similaires) à l'entrée

Illustration: 3 plus proches voisins

- Reconnaissance de caractère: est-ce un ‘e’ ou un ‘o’?



Si on ajoute 200 exemples par classe...

Vrai!

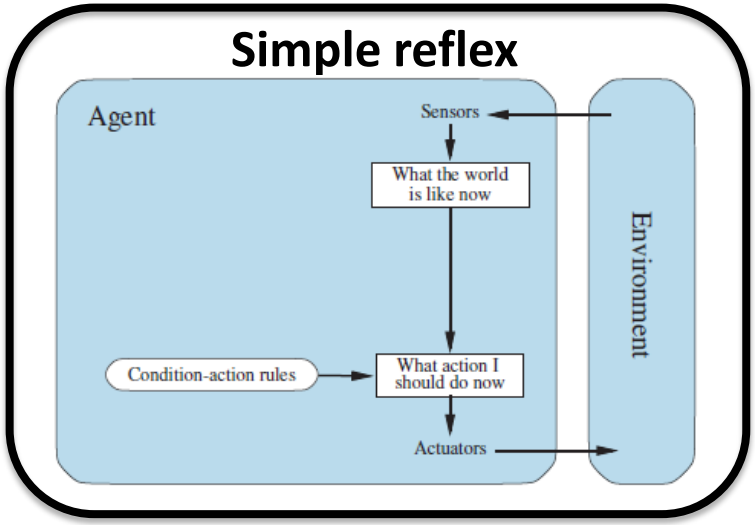
Mesure de la performance d'un algorithme d'apprentissage

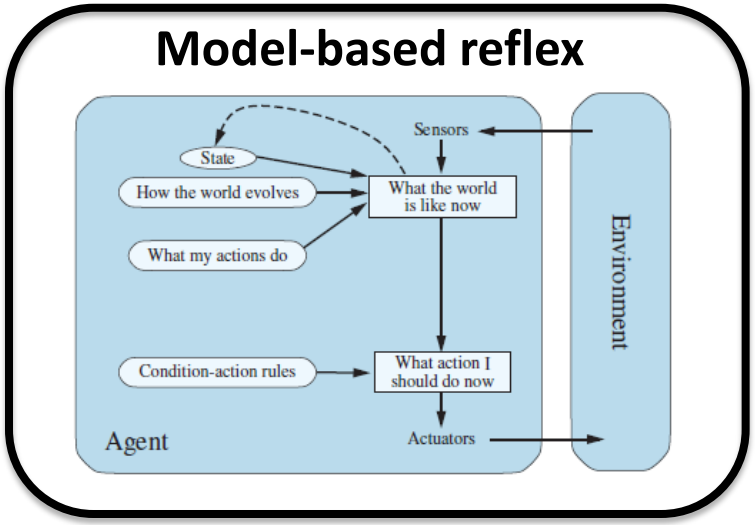
- Comment évaluer le succès d'un algorithme?
  - on pourrait regarder l'erreur moyenne commise sur les exemples d'entraînement, mais cette erreur sera nécessairement optimiste
    - » h a déjà vu la bonne réponse pour ces exemples!
    - » on mesure donc seulement la capacité de l'algorithme à **mémoriser**

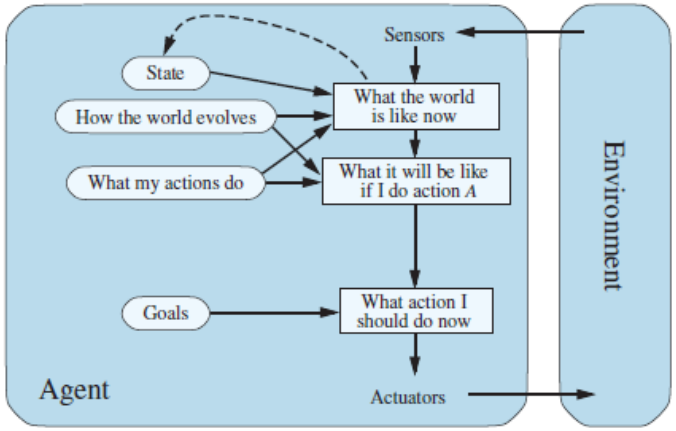
- Ce qui nous intéresse vraiment, c'est la capacité de l'algorithme à **généraliser** sur de **nouveaux exemples**
  - ça reflète mieux le contexte dans lequel on va utiliser *h*

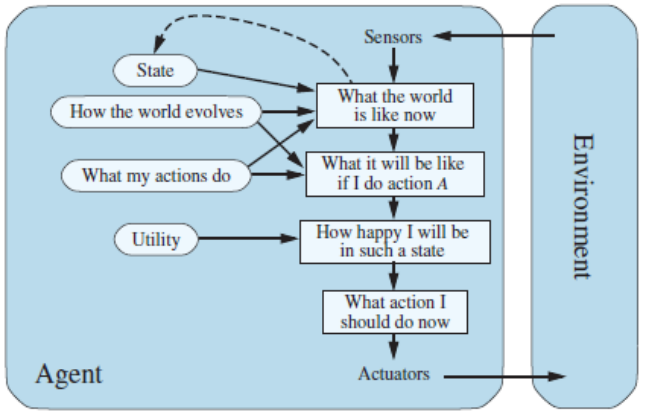
- Pour mesurer la généralisation, on met de côté des exemples étiquetés, qui seront utilisés seulement à la toute fin, pour calculer l'erreur
  - on l'appel l'**ensemble de test**

Algorithme des K plus proches voisins pour quel type d'agent?









Vous devriez être capable de...

- Nommer les trois formes d'apprentissage: supervisé, non supervisé, par renforcement
- Expliquer et simuler l'algorithme des K plus proches voisins