

Qu'est-ce qu'apprendre ?

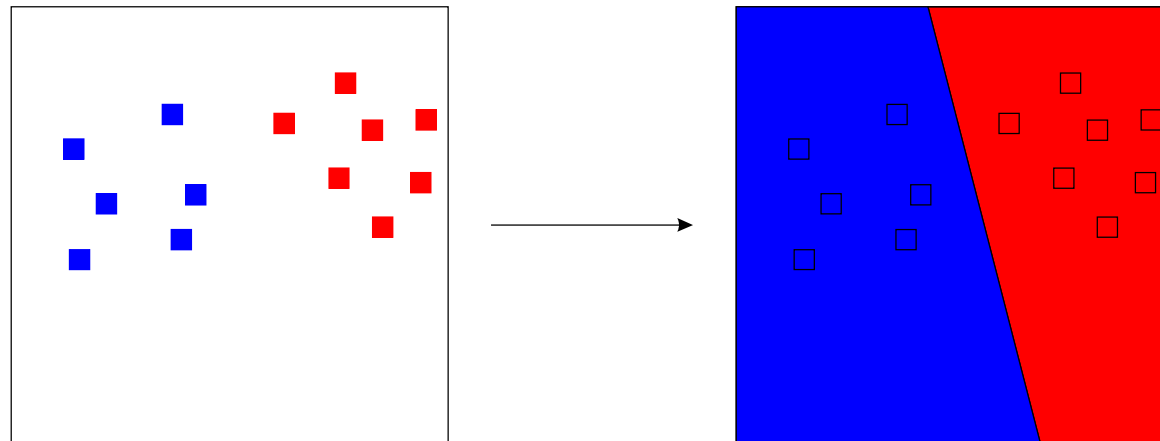
Qu'est-ce qu'apprendre ?

Intuition :

Découvrir (ou estimer) une fonction (ou une distribution) inconnue à partir d'un ensemble d'exemples

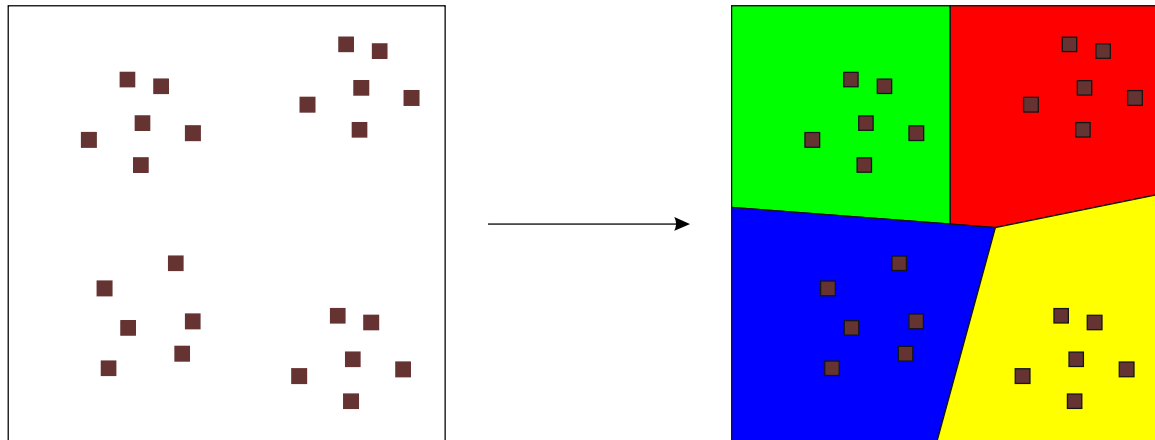
Qu'est-ce qu'apprendre ?

Apprentissage supervisé :



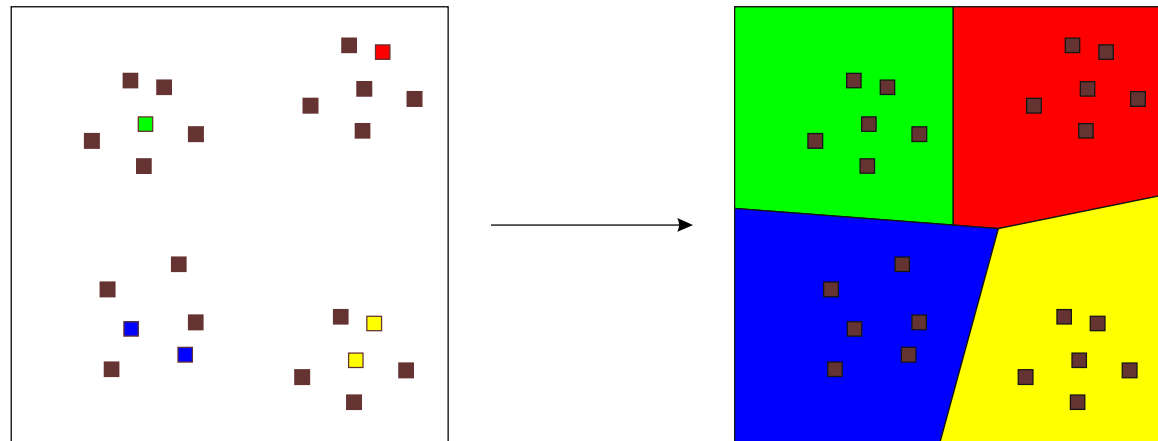
Qu'est-ce qu'apprendre ?

Apprentissage non supervisé :



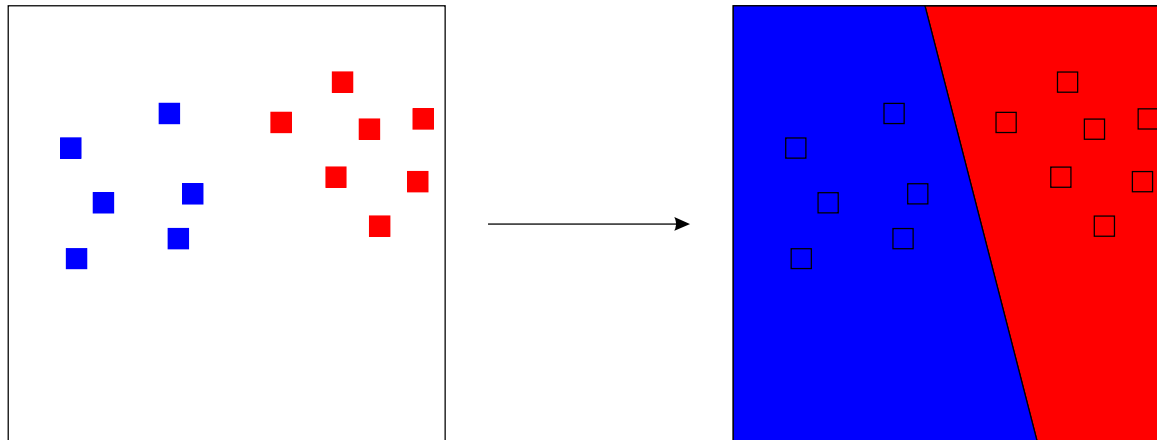
Qu'est-ce qu'apprendre ?

Apprentissage semi supervisé :



Qu'est-ce qu'apprendre ?

Apprentissage supervisé :



Apprentissage supervisé



Apprentissage supervisé

Apprendre ...



Apprentissage supervisé



Apprendre ...

- Apprendre par cœur ?

Apprentissage supervisé



Apprendre ...

- Exemples (Input => Output)
 - $\{1, 2\} \Rightarrow \{3\}$
 - $\{4, 2\} \Rightarrow \{6\}$
 - $\{2, 2\} \Rightarrow \{4\}$
 - $\{8, 13\} \Rightarrow \{21\}$

Apprentissage supervisé



Apprendre ...

- Exemples (Input \Rightarrow Output)
 - $\{1, 2\} \Rightarrow \{3\}$
 - $\{4, 2\} \Rightarrow \{6\}$
 - $\{2, 2\} \Rightarrow \{4\}$
 - $\{8, 13\} \Rightarrow \{21\}$
- Apprendre par cœur ?
 - Dictionnaire ?

Apprentissage supervisé



Apprendre ...

- Exemples (Input => Output)
 - $\{1, 2\} \Rightarrow \{3\}$
 - $\{4, 2\} \Rightarrow \{6\}$
 - $\{2, 2\} \Rightarrow \{4\}$
 - $\{8, 13\} \Rightarrow \{21\}$
- Apprendre par cœur ?
 - Dictionnaire ?
 - Aucune information sur le reste de l'espace d'entrée !

Apprentissage supervisé



Apprendre ...

... n'a d'intérêt que si on généralise !

Apprentissage supervisé

Qu'est-ce que généraliser ?



Apprentissage supervisé



⇒ Généraliser :

⇒ Supposer qu'il existe une **fonction cible** qui a généré les exemples que nous avons à disposition.

⇒ Essayer d'**approximer** les résultats de cette fonction cible à l'aide d'un modèle.

⇒ *Espérer* 😊 que si on approxime « *bien* » les résultats donnés sur les exemples étiquetés, on approximera « *bien* » sur l'ensemble de l'espace d'entrée

Apprentissage supervisé



⇒ Généraliser :

⇒ Supposer qu'il existe une **fonction cible** qui a généré les exemples que nous avons à disposition.

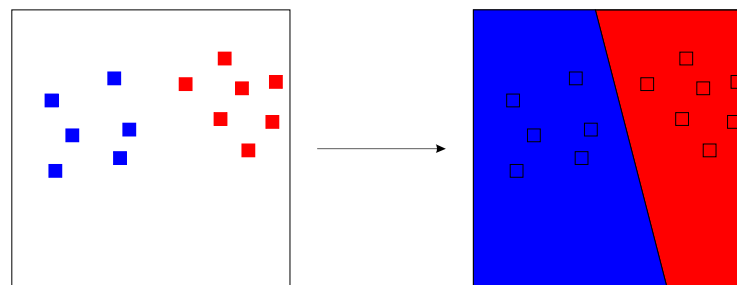
⇒ Essayer d'**approximer** les résultats de cette fonction cible à l'aide d'un modèle.

⇒ *Espérer* ☺ que si on approxime « *bien* » les résultats donnés sur les exemples étiquetés, on approximera « *bien* » sur l'ensemble de l'espace d'entrée

Apprentissage supervisé



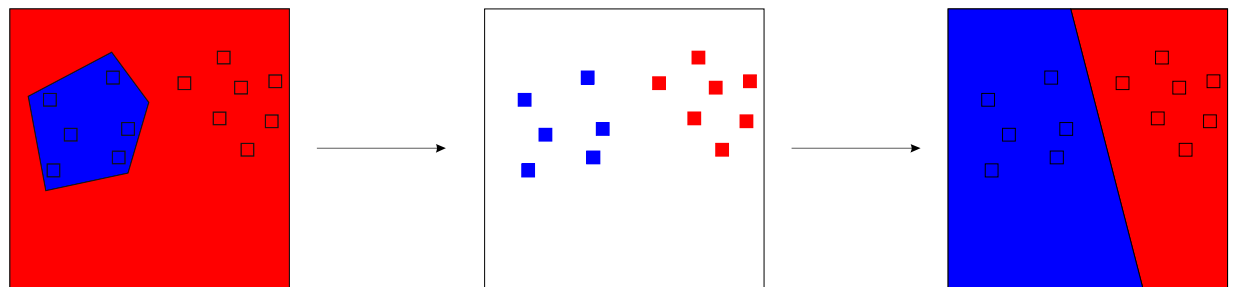
⇒ Contre exemple abstrait:



Apprentissage supervisé



⇒ Contre exemple abstrait:



Apprentissage supervisé

⇒ Contre exemple de l'arnaque à la prédiction



Apprentissage supervisé

Arnaque à la prédiction :



Apprentissage supervisé

Arnaque à la prédiction :



Apprentissage supervisé

Arnaque à la prédiction :



Apprentissage supervisé

Arnaque à la prédiction :



Apprentissage supervisé

Arnaque à la prédiction :

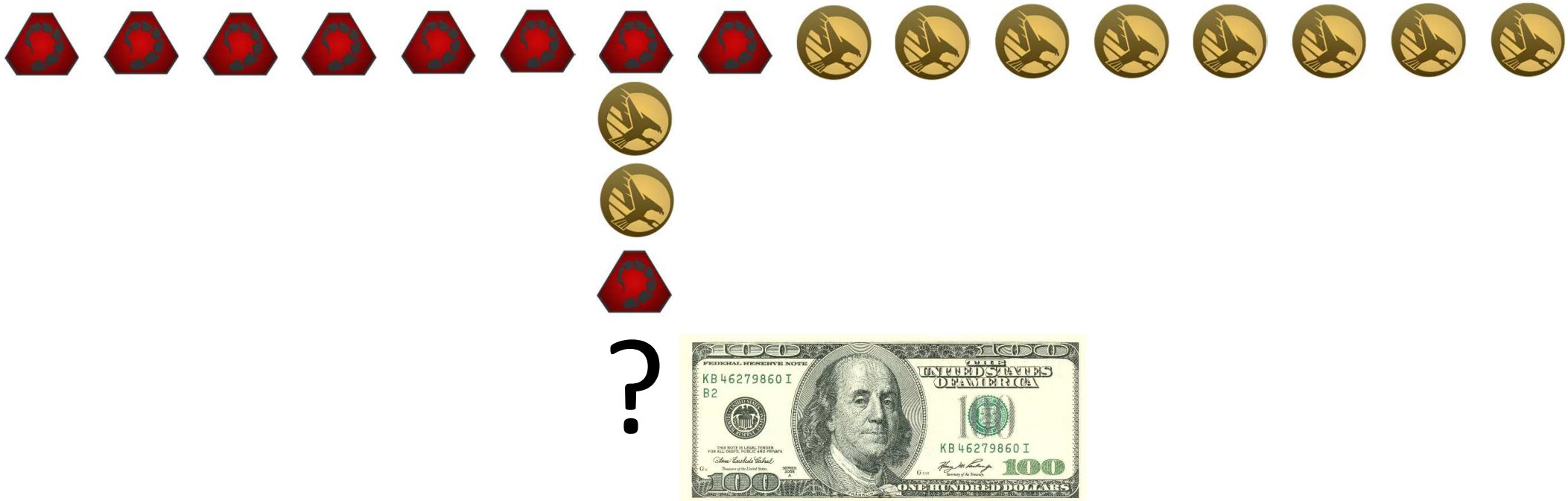


?



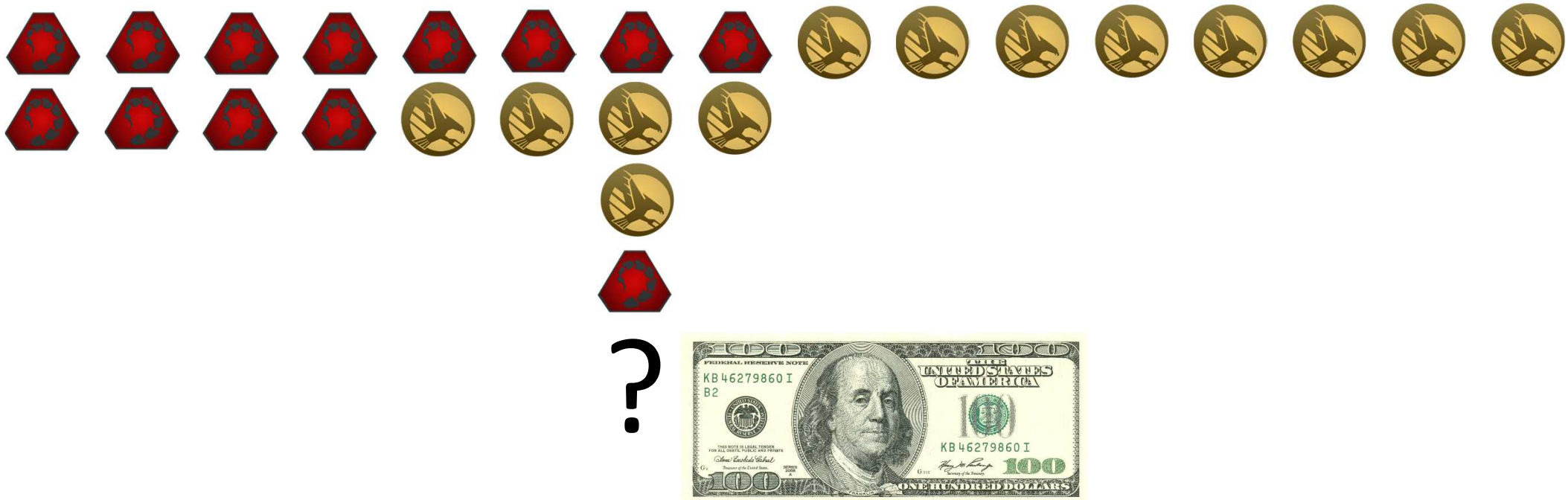
Apprentissage supervisé

Arnaque à la prédiction :



Apprentissage supervisé

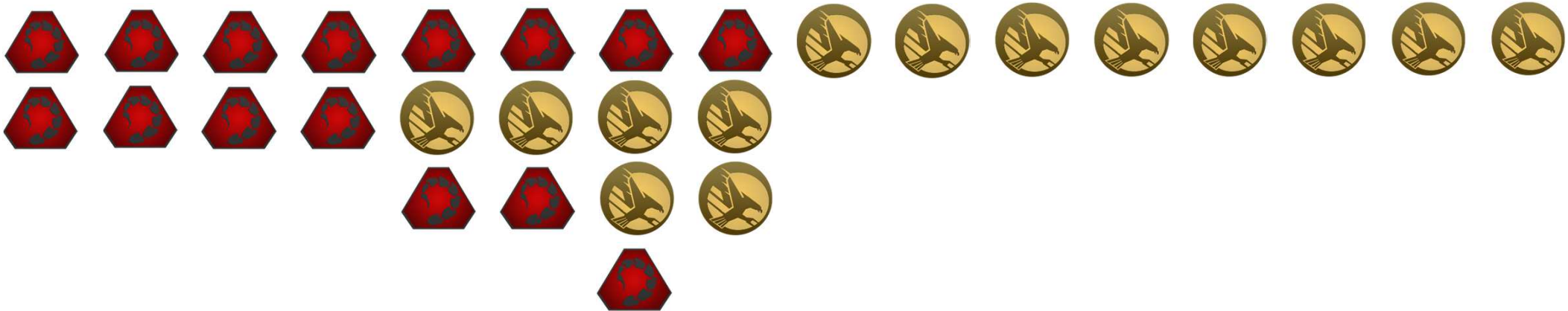
Arnaque à la prédiction :



Apprentissage supervisé



Arnaque à la prédiction :



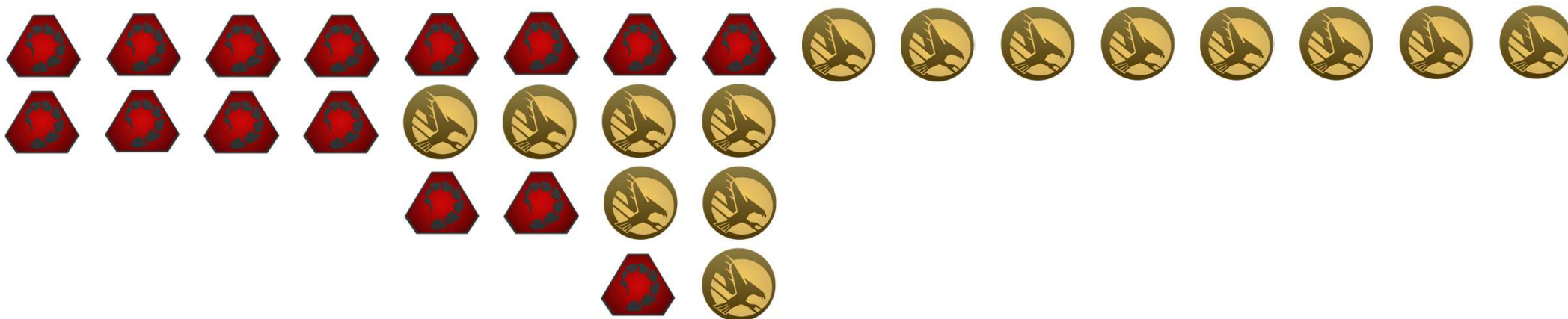
?



Apprentissage supervisé



Arnaque à la prédiction :

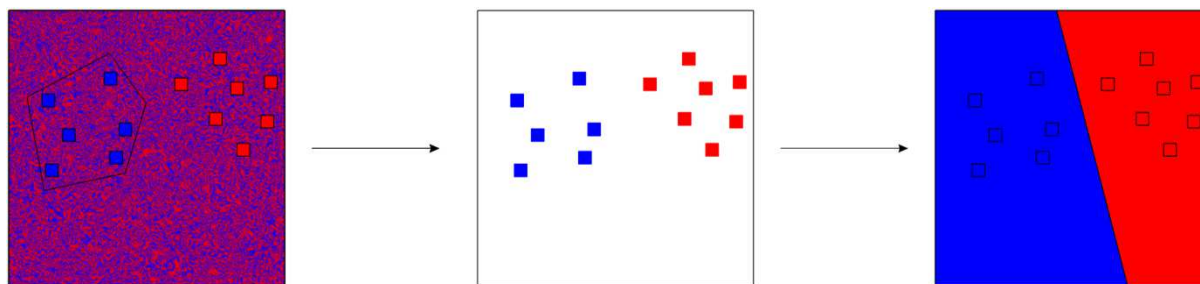


?



Apprentissage supervisé

⇒ Arnaque à la prédiction:



Apprentissage supervisé

Quelles validations théoriques ?

<https://work.caltech.edu/telecourse.html>

Apprentissage supervisé

Inégalité de Hoeffding :

soit μ : probabilité d'obtenir un échantillon bleu dans un ensemble

soit ν : proportion d'échantillons bleus dans un échantillonnage

Si N est mon nombre d'échantillons et ϵ un réel :

$$P[|\nu - \mu| > \epsilon] \leq 2e^{-2\epsilon^2 N}$$

Apprentissage supervisé

Ce qui nous amène à :

soit E_{in} : l'erreur de classement d'une hypothèse sur les échantillons par rapport à la fonction cible.

soit E_{out} : l'erreur de classement d'une hypothèse l'ensemble des entrées possibles par rapport à la fonction cible.

soit g : mon hypothèse

soit M : L'ensemble des hypothèses possibles pour mon modèle

$$P[|E_{in}(g) - E_{out}(g)| > \epsilon] \leq 2Me^{-2\epsilon^2 N}$$

Apprentissage supervisé

Inégalité de Vapnik-Chervonenkis :

soit m_H : le nombre maximum de dichotomies réalisables sur un ensemble d'exemples par une classe d'hypothèse H .

$$P[|E_{in}(g) - E_{out}(g)| > \epsilon] \leq 4m_H(2N)e^{-\frac{1}{8}\epsilon^2 N}$$

Apprentissage supervisé

Conclusion théorique :

- Généraliser est parfois possible
- Cela dépend :
 - Du nombre d'exemples étiquetés à disposition
 - De la qualité de la généralisation que l'on cherche
 - De la complexité du modèle utilisé pour générer nos hypothèses
- Règle générale, approximative mais pratique :
 - Ne pas espérer obtenir une bonne généralisation si le nombre d'exemple à disposition n'est pas supérieur à **10 fois** le nombre de paramètres du modèle utilisé.



Classification VS Régression

Classification :

- Appartenance d'un exemple à un ensemble fini :



Classification VS Régression

Régression :

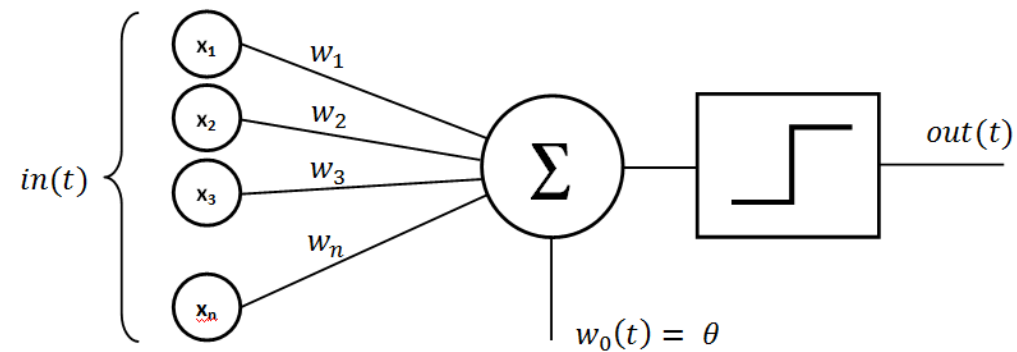
- Prédire une (ou plusieurs) valeurs réelles :



Classification et séparations linéaires

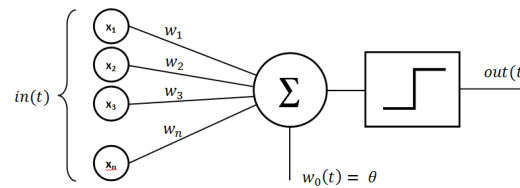
Classification et séparations linéaires

- Retours sur le Perceptron



Classification et séparations linéaires

- Retours sur le Perceptron



- Que l'on peut réécrire :
 - $out = Sign(\sum_{i=0}^n w_i x_i)$
- Ou sous forme matricielle :
 - $out = Sign(W^T X)$ en prenant soin d'ajouter le biais ($x_0 = 1$)

Classification et séparations linéaires

- Algorithmes d'apprentissages du perceptron pour la classification
- But du jeu : déterminer W
- Non supervisée
 - Règle de Hebb
- Supervisée
 - PLA ou Règle de Rosenblatt

Classification et séparations linéaires

- Perceptron Learning Algorithm (pour des sorties à -1 ou 1)
 - Initialiser W (random(-1,1) ou 0)
 - Répéter :
 - Prendre un exemple MAL classé (où $g(X^k) \neq Y^k$) au hasard et, mettre à jour W selon la règle :

$$W \leftarrow W + \alpha Y^k X^k$$

- Règle de Rosenblatt (pour des sorties à 0 ou 1) (marche aussi pour des sorties à -1 ou 1)
 - Initialiser W (random(-1,1) ou 0)
 - Répéter :

$$W \leftarrow W + \alpha (Y^k - g(X^k)) X^k$$

Avec :

- α le pas d'apprentissage
- X^k les paramètres de l'exemple k et le biais $x_0^k = 1$.
- Y^k la sortie attendue pour l'exemple k .
- $g(X^k)$ la sortie obtenue par le perceptron pour l'exemple k .

Régression linéaire

- Minimiser le carré de l'erreur

- Notons $X = \begin{bmatrix} x_0^0 & \cdots & x_n^0 \\ \vdots & \ddots & \vdots \\ x_0^N & \cdots & x_n^N \end{bmatrix}$ et $Y = \begin{bmatrix} y_0^0 & \cdots & y_n^0 \\ \vdots & \ddots & \vdots \\ y_0^N & \cdots & y_n^N \end{bmatrix}$

- Supposons $n \leq N$
- Utilisation de la pseudo inverse pour calculer W en un coup :

$$W = ((X^T X)^{-1} X^T) Y$$