

# IA et apprentissage automatique

Philippe Preux

`philippe.preux@univ-lille.fr`

<https://philippe-preux.github.io/talks/IA-et-AA.DU-IA-sante.pdf>



# L'IA, c'est quoi ?

- ▶ IA = mécanisme qui donne l'impression de réaliser une tâche que l'on croit que seul l'Homme peut réaliser.


# L'IA, c'est quoi ?

- ▶ IA = mécanisme qui donne l'impression de réaliser une tâche que l'on croit que seul l'Homme peut réaliser.
- ▶ Concept mouvant  
du cognitivisme au comportementalisme.


# L'IA, c'est quoi ?

- ▶ IA = mécanisme qui donne l'impression de réaliser une tâche que l'on croit que seul l'Homme peut réaliser.
- ▶ Concept mouvant  
du cognitivisme au comportementalisme.
- ▶ Pour les scientifiques, la notion recule au fur et à mesure des progrès de la mécanisation.

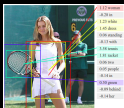
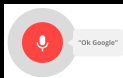
# L'IA, c'est quoi ?

- ▶ IA = mécanisme qui donne l'impression de réaliser une tâche que l'on croit que seul l'Homme peut réaliser.
- ▶ Concept mouvant  
du cognitivisme au comportementalisme.
- ▶ Pour les scientifiques, la notion recule au fur et à mesure des progrès de la mécanisation.
- ▶  IA est actuellement utilisé à tort et à travers.

# L'IA, c'est quoi ?

- ▶ IA = mécanisme qui donne l'impression de réaliser une tâche que l'on croit que seul l'Homme peut réaliser.
- ▶ Concept mouvant  
du cognitivisme au comportementalisme.
- ▶ Pour les scientifiques, la notion recule au fur et à mesure des progrès de la mécanisation.
- ▶  IA est actuellement utilisé à tort et à travers.
- ▶ Aujourd'hui, IA = apprentissage automatique.

## Quelques applications de l'apprentissage automatique



*"man in black shirt  
is playing guitar"*



# IA vs. AA

IA symbolique	Apprentissage automatique



# IA vs. AA

IA symbolique	Apprentissage automatique
Déduction / Inférence	Induction

# IA vs. AA

IA symbolique	Apprentissage automatique
Déduction / Inférence	Induction
Connaissance	Données

# IA vs. AA

IA symbolique	Apprentissage automatique
Déduction / Inférence	Induction
Connaissance	Données
Raisonnement	Généralisation

# IA vs. AA

IA symbolique	Apprentissage automatique
Déduction / Inférence	Induction
Connaissance	Données
Raisonnement	Généralisation
Logique	Statistiques / Optimisation

# IA vs. AA

IA symbolique	Apprentissage automatique
Déduction / Inférence	Induction
Connaissance	Données
Raisonnement	Généralisation
Logique	Statistiques / Optimisation
Si A est vrai et $A \Rightarrow B$ , alors B est vrai.	

# IA vs. AA

IA symbolique	Apprentissage automatique										
Déduction / Inférence	Induction										
Connaissance	Données										
Raisonnement	Généralisation										
Logique	Statistiques / Optimisation										
Si A est vrai et $A \Rightarrow B$ , alors B est vrai.	<table><tr><td>x</td><td>y</td></tr><tr><td>1,03</td><td>2</td></tr><tr><td>-4,97</td><td>-10</td></tr><tr><td>-0,03</td><td>0</td></tr><tr><td><math>\hat{y}</math></td><td><math>= 2x</math></td></tr></table>	x	y	1,03	2	-4,97	-10	-0,03	0	$\hat{y}$	$= 2x$
x	y										
1,03	2										
-4,97	-10										
-0,03	0										
$\hat{y}$	$= 2x$										

# IA vs. AA

IA symbolique	Apprentissage automatique
Déduction / Inférence	Induction
Connaissance	Données
Raisonnement	Généralisation
Logique	Statistiques / Optimisation
Si A est vrai et $A \Rightarrow B$ , alors B est vrai.	$\begin{array}{cc} x & y \\ 1,03 & 2 \\ -4,97 & -10 \\ -0,03 & 0 \end{array}$ $\hat{y} = 2x$
Connaissance exacte Monde déterministe	Connaissance approchée Monde non déterministe

# IA vs. AA

IA symbolique	Apprentissage automatique
Déduction / Inférence	Induction
Connaissance	Données
Raisonnement	Généralisation
Logique	Statistiques / Optimisation
Si A est vrai et $A \Rightarrow B$ , alors B est vrai.	$\begin{array}{cc} x & y \\ 1,03 & 2 \\ -4,97 & -10 \\ -0,03 & 0 \end{array}$ $\hat{y} = 2x$
Connaissance exacte Monde déterministe	Connaissance approchée Monde non déterministe

Méthodes bayésiennes



► connaissance approchée  $\iff$  distance

- ▶ connaissance approchée  $\iff$  distance
- ▶ idéalement :  
donnée = un point dans un espace métrique = un vecteur dans un espace vectoriel

# Différentes tâches résolues par AA

- ▶ Apprentissage supervisé : associer une étiquette à une donnée.  
Étiquette =

# Différentes tâches résolues par AA

- ▶ Apprentissage supervisé : associer une étiquette à une donnée.  
Étiquette =
  - ▶ un symbole (classe)  $\rightsquigarrow$  classification supervisée

# Différentes tâches résolues par AA

- ▶ Apprentissage supervisé : associer une étiquette à une donnée.

Étiquette =

- ▶ un symbole (classe)  $\rightsquigarrow$  classification supervisée
- ▶ un nombre réel  $\rightsquigarrow$  régression

# Différentes tâches résolues par AA

- ▶ Apprentissage supervisé : associer une étiquette à une donnée.

Étiquette =

- ▶ un symbole (classe)  $\rightsquigarrow$  classification supervisée
- ▶ un nombre réel  $\rightsquigarrow$  régression
- ▶ un rang  $\rightsquigarrow$  régression ordinale

# Différentes tâches résolues par AA

- ▶ Apprentissage supervisé : associer une étiquette à une donnée.

Étiquette =

- ▶ un symbole (classe)  $\rightsquigarrow$  classification supervisée
- ▶ un nombre réel  $\rightsquigarrow$  régression
- ▶ un rang  $\rightsquigarrow$  régression ordinale
- ▶ un ensemble de symboles  $\rightsquigarrow$  classification supervisée multi-étiquettes

# Différentes tâches résolues par AA

- ▶ Apprentissage supervisé : associer une étiquette à une donnée.

Étiquette =

- ▶ un symbole (classe)  $\rightsquigarrow$  classification supervisée
- ▶ un nombre réel  $\rightsquigarrow$  régression
- ▶ un rang  $\rightsquigarrow$  régression ordinale
- ▶ un ensemble de symboles  $\rightsquigarrow$  classification supervisée multi-étiquettes
- ▶ ... la légende d'une figure ...



# Différentes tâches résolues par AA

- ▶ Apprentissage supervisé : associer une étiquette à une donnée.

Étiquette =

- ▶ un symbole (classe)  $\rightsquigarrow$  classification supervisée
- ▶ un nombre réel  $\rightsquigarrow$  régression
- ▶ un rang  $\rightsquigarrow$  régression ordinale
- ▶ un ensemble de symboles  $\rightsquigarrow$  classification supervisée multi-étiquettes
- ▶ ... la légende d'une figure ...

On apprend à partir d'exemples = couples (donnée, étiquette).

# Différentes tâches résolues par AA

- ▶ Apprentissage supervisé : associer une étiquette à une donnée.

Étiquette =

- ▶ un symbole (classe)  $\rightsquigarrow$  classification supervisée
- ▶ un nombre réel  $\rightsquigarrow$  régression
- ▶ un rang  $\rightsquigarrow$  régression ordinale
- ▶ un ensemble de symboles  $\rightsquigarrow$  classification supervisée multi-étiquettes
- ▶ ... la légende d'une figure ...

On apprend à partir d'exemples = couples (donnée, étiquette).

On mesure la qualité d'un modèle par sa capacité à prédire la classe de données non utilisées pour l'entraînement du modèle.

# Différentes tâches résolues par AA

- ▶ Apprentissage supervisé : associer une étiquette à une donnée.

Étiquette =

- ▶ un symbole (classe)  $\rightsquigarrow$  classification supervisée
- ▶ un nombre réel  $\rightsquigarrow$  régression
- ▶ un rang  $\rightsquigarrow$  régression ordinale
- ▶ un ensemble de symboles  $\rightsquigarrow$  classification supervisée multi-étiquettes
- ▶ ... la légende d'une figure ...

On apprend à partir d'exemples = couples (donnée, étiquette).

On mesure la qualité d'un modèle par sa capacité à prédire la classe de données non utilisées pour l'entraînement du modèle.

- ▶ Apprentissage non supervisé : regrouper les données en fonction de leur ressemblance.

# Différentes tâches résolues par AA

- ▶ Apprentissage supervisé : associer une étiquette à une donnée.

Étiquette =

- ▶ un symbole (classe)  $\rightsquigarrow$  classification supervisée
- ▶ un nombre réel  $\rightsquigarrow$  régression
- ▶ un rang  $\rightsquigarrow$  régression ordinale
- ▶ un ensemble de symboles  $\rightsquigarrow$  classification supervisée multi-étiquettes
- ▶ ... la légende d'une figure ...

On apprend à partir d'exemples = couples (donnée, étiquette).

On mesure la qualité d'un modèle par sa capacité à prédire la classe de données non utilisées pour l'entraînement du modèle.

- ▶ Apprentissage non supervisé : regrouper les données en fonction de leur ressemblance.
- ▶ Apprentissage par renforcement : apprendre à faire.

# Classification supervisée

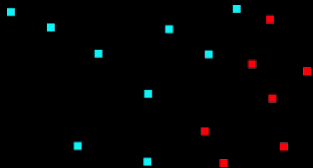
- ▶ rappel : notion centrale de l'AA = distance

# Classification supervisée

- ▶ rappel : notion centrale de l'AA = distance
- ▶  $\rightsquigarrow$  intuition : deux données proches appartiennent à la même classe

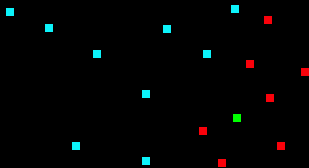
# Classification supervisée

- ▶ rappel : notion centrale de l'AA = distance
- ▶  $\rightsquigarrow$  intuition : deux données proches appartiennent à la même classe



# Classification supervisée

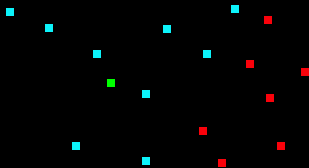
- ▶ rappel : notion centrale de l'AA = distance
- ▶  $\rightsquigarrow$  intuition : deux données proches appartiennent à la même classe





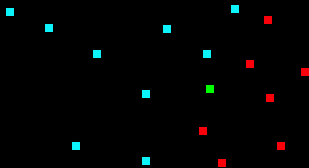
# Classification supervisée

- ▶ rappel : notion centrale de l'AA = distance
- ▶  $\rightsquigarrow$  intuition : deux données proches appartiennent à la même classe



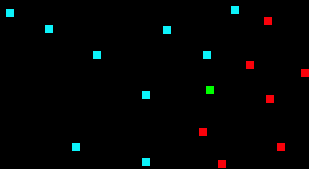
# Classification supervisée

- ▶ rappel : notion centrale de l'AA = distance
- ▶  $\rightsquigarrow$  intuition : deux données proches appartiennent à la même classe



# Classification supervisée

- ▶ rappel : notion centrale de l'AA = distance
- ▶  $\rightsquigarrow$  intuition : deux données proches appartiennent à la même classe



- ▶  $\rightsquigarrow$  algorithme des plus proches voisins k-NN

# Classification supervisée

- ▶ rappel : notion centrale de l'AA = distance

# Classification supervisée

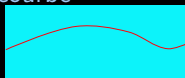
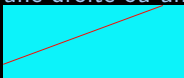
- ▶ rappel : notion centrale de l'AA = distance
- ▶  $\rightsquigarrow$  intuition 2 : les données d'une même classe forment une variété dans l'espace de données

# Classification supervisée

- ▶ rappel : notion centrale de l'AA = distance
- ▶  $\rightsquigarrow$  intuition 2 : les données d'une même classe forment une variété dans l'espace de données
- ▶ variété : dans le plan :

# Classification supervisée

- ▶ rappel : notion centrale de l'AA = distance
- ▶  $\rightsquigarrow$  intuition 2 : les données d'une même classe forment une variété dans l'espace de données
- ▶ variété : dans le plan :
  - ▶ une droite ou une courbe



# Classification supervisée

- ▶ rappel : notion centrale de l'AA = distance
- ▶  $\rightsquigarrow$  intuition 2 : les données d'une même classe forment une variété dans l'espace de données
- ▶ variété : dans le plan :
  - ▶ une droite ou une courbe
  - ▶ une région rectangulaire, elliptique, ...



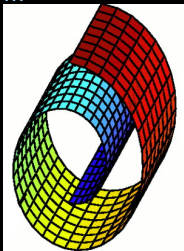
# Classification supervisée

- ▶ rappel : notion centrale de l'AA = distance
- ▶  $\rightsquigarrow$  intuition 2 : les données d'une même classe forment une variété dans l'espace de données
- ▶ variété : dans le plan :
  - ▶ une droite ou une courbe
  - ▶ une région rectangulaire, elliptique, ...
  - ▶ une région bornée



# Classification supervisée

- ▶ rappel : notion centrale de l'AA = distance
- ▶  $\rightsquigarrow$  intuition 2 : les données d'une même classe forment une variété dans l'espace de données
- ▶ variété : dans le plan :
  - ▶ une droite ou une courbe
  - ▶ une région rectangulaire, elliptique, ...
  - ▶ une région bornée
  - ▶ ...



# Classification supervisée

- ▶ rappel : notion centrale de l'AA = distance
- ▶  $\rightsquigarrow$  intuition 2 : les données d'une même classe forment une variété dans l'espace de données
- ▶ variété : dans le plan :
  - ▶ une droite ou une courbe
  - ▶ une région rectangulaire, elliptique, ...
  - ▶ une région bornée
  - ▶ ...
- ▶  $\rightsquigarrow$  rechercher de telles variétés associées à chacune des classes.

# Classification supervisée

- ▶ rappel : notion centrale de l'AA = distance
- ▶  $\rightsquigarrow$  intuition 2 : les données d'une même classe forment une variété dans l'espace de données
- ▶ variété : dans le plan :
  - ▶ une droite ou une courbe
  - ▶ une région rectangulaire, elliptique, ...
  - ▶ une région bornée
  - ▶ ...
- ▶  $\rightsquigarrow$  rechercher de telles variétés associées à chacune des classes.
- ▶ L'espace des données est en dimension  $P \gg 2$  en général.

# Classification supervisée

Arbre de décision : pavage de l'espace des données

- ▶ variétés = parallélépipède (en dimension  $P$ )

# Classification supervisée

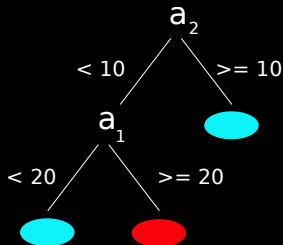
## Arbre de décision : pavage de l'espace des données

- ▶ variétés = parallélépipède (en dimension  $P$ )
- ▶ exemple : données décrites par 2 attributs  $a_1$  et  $a_2$  et leur classe (bleu ou rouge)

# Classification supervisée

Arbre de décision : pavage de l'espace des données

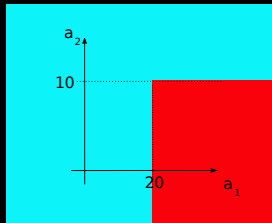
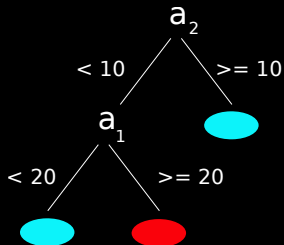
- ▶ variétés = parallélépipède (en dimension  $P$ )
- ▶ exemple : données décrites par 2 attributs  $a_1$  et  $a_2$  et leur classe (bleu ou rouge)



# Classification supervisée

Arbre de décision : pavage de l'espace des données

- ▶ variétés = parallélépipède (en dimension  $P$ )
- ▶ exemple : données décrites par 2 attributs  $a_1$  et  $a_2$  et leur classe (bleu ou rouge)

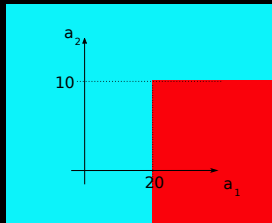
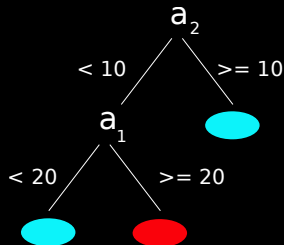




# Classification supervisée

## Arbre de décision : pavage de l'espace des données

- ▶ variétés = parallélépipède (en dimension  $P$ )
- ▶ exemple : données décrites par 2 attributs  $a_1$  et  $a_2$  et leur classe (bleu ou rouge)



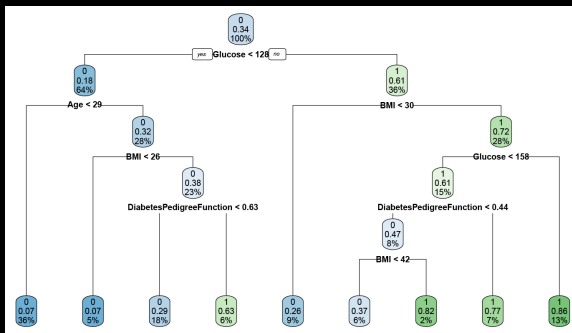
- ▶ Un arbre de décision découpe l'espace des données en hyper-parallélépipèdes.

# Classification supervisée

## Arbre de décision : pavage de l'espace des données

### ► exemple sur jeu de données « diabete » :

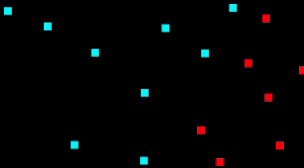
768 exemples décrits par 8 attributs et une classe : *number of times pregnant*, *plasma glucose concentration*, *diastolic blood pressure (mm Hg)*, *triceps skin fold thickness (in mm)*, *2-hr serum insulin measure*, *body mass index*, *a diabetes pedigree function*, and *age* et la classe : patient diabétique ou non.



### ► avantage : modèle interprétable ; associe une importance à chaque attribut pour la décision quant à la classe de la donnée.

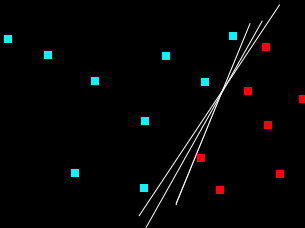
# Classification supervisée

## Séparation linéaire



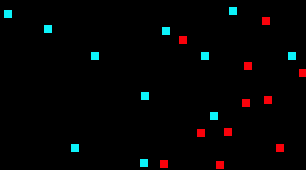
# Classification supervisée

## Séparation linéaire



# Classification supervisée

## Séparation linéaire



- ▶ On cherche une droite qui sépare au mieux.
- ▶ On coupe l'espace des données en 2 par un hyper-plan.
- ▶ Cela revient à chercher des paramètres  $\theta_0 \dots \theta_P$  tels que  $\sum_i \theta_i a_i \geq 0$  pour les bleus,  $< 0$  pour les rouges. On pose  $a_0 = 1$
- ▶ Il y a beaucoup de méthodes pour faire cela.

# Classification supervisée

Perceptron = neurone artificiel

- ▶ On cherche des paramètres  $\theta_0 \dots \theta_P$  tels que  $\sum_i \theta_i a_i \geq 0$  pour les bleus,  $< 0$  pour les rouges.

# Classification supervisée

Perceptron = neurone artificiel

- ▶ On cherche des paramètres  $\theta_0 \dots \theta_P$  tels que  $\sum_i \theta_i a_i \geq 0$  pour les bleus,  $< 0$  pour les rouges.
- ▶ Différents algorithmes permettent de calculer ces  $\theta_i$ .

# Classification supervisée

Perceptron = neurone artificiel

- ▶ On cherche des paramètres  $\theta_0 \dots \theta_P$  tels que  $\sum_i \theta_i a_i \geq 0$  pour les bleus,  $< 0$  pour les rouges.
- ▶ Différents algorithmes permettent de calculer ces  $\theta_i$ .
- ▶ Principe de l'algorithme :
  - ▶ sélectionner au hasard un sous-ensemble d'exemples et calculer pour chacun la classe prédite par les  $\theta_i$  ;
  - ▶ pour chacun de ces exemples, si la classe prédite est différente de la classe de l'exemple, corriger les  $\theta_i$  ;
  - ▶ recommencer ces 2 étapes un certain nombre de fois.



# Classification supervisée

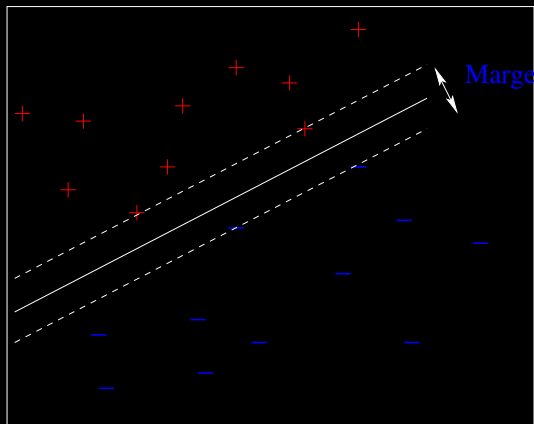
Perceptron = neurone artificiel

- ▶ On cherche des paramètres  $\theta_0 \dots \theta_P$  tels que  $\sum_i \theta_i a_i \geq 0$  pour les bleus,  $< 0$  pour les rouges.
- ▶ Différents algorithmes permettent de calculer ces  $\theta_i$ .
- ▶ Principe de l'algorithme :
  - ▶ sélectionner au hasard un sous-ensemble d'exemples et calculer pour chacun la classe prédite par les  $\theta_i$  ;
  - ▶ pour chacun de ces exemples, si la classe prédite est différente de la classe de l'exemple, corriger les  $\theta_i$  ;
  - ▶ recommencer ces 2 étapes un certain nombre de fois.
- ▶ Quand s'arrêter ?
  - ▶ Découper le jeu d'exemples en un jeu d'entraînement  $\mathcal{E}$  et un jeu de test  $\mathcal{T}$ .
  - ▶ Utiliser  $\mathcal{T}$  uniquement pour estimer la qualité de la prédiction.
  - ▶ Arrêter quand l'erreur de prédiction mesurée sur  $\mathcal{T}$  ne diminue plus.

# Classification supervisée

## Machines à vaste marge

- On cherche les paramètres de l'hyper-plan qui maximise la marge :



# Classification supervisée

## Machines à vaste marge

- ▶ théorème de Cover : si on ajoute des attributs qui sont des combinaisons non linéaires des attributs de base, on augmente la probabilité que les données soient linéairement séparables.
- ▶ Pulvérisation des données dans un espace de grande dimension.
- ▶ Idée naïve : on ajoute au hasard de tels attributs et on espère que ça marche.
- ▶ Idée non naïve : on utilise une méthode pour cela avec des fondements mathématiques. Approche basée sur une notion de distance entre les données.
- ▶ Géométriquement : les données pulvérisées sont linéairement séparables.

# Réseaux de neurones : perceptron multi-couches

- ▶ géométriquement : la séparation entre les 2 classes peut avoir à peu près n'importe quelle forme continue.

# Réseaux de neurones : perceptron multi-couches

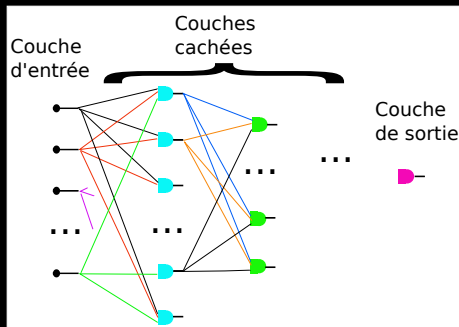
- ▶ géométriquement : la séparation entre les 2 classes peut avoir à peu près n'importe quelle forme continue.
- ▶ un neurone  $\iff$  séparatrice = droite

# Réseaux de neurones : perceptron multi-couches

- ▶ géométriquement : la séparation entre les 2 classes peut avoir à peu près n'importe quelle forme continue.
- ▶ un neurone  $\iff$  séparatrice = droite
- ▶ plusieurs neurones  $\Rightarrow$  séparatrice de forme arbitraire (continue)

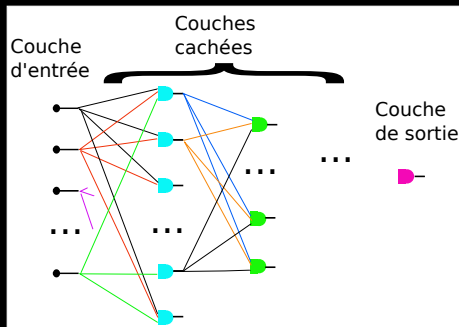
# Réseaux de neurones : perceptron multi-couches

- ▶ géométriquement : la séparation entre les 2 classes peut avoir à peu près n'importe quelle forme continue.
- ▶ un neurone  $\iff$  séparatrice = droite
- ▶ plusieurs neurones  $\Rightarrow$  séparatrice de forme arbitraire (continue)



# Réseaux de neurones : perceptron multi-couches

- ▶ géométriquement : la séparation entre les 2 classes peut avoir à peu près n'importe quelle forme continue.
- ▶ un neurone  $\iff$  séparatrice = droite
- ▶ plusieurs neurones  $\Rightarrow$  séparatrice de forme arbitraire (continue)

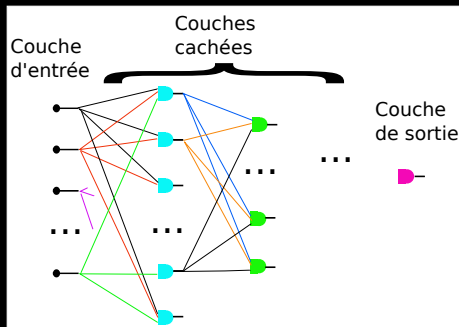


- ▶ un paramètre  $\theta_i$  par connexion



# Réseaux de neurones : perceptron multi-couches

- ▶ géométriquement : la séparation entre les 2 classes peut avoir à peu près n'importe quelle forme continue.
- ▶ un neurone  $\iff$  séparatrice = droite
- ▶ plusieurs neurones  $\Rightarrow$  séparatrice de forme arbitraire (continue)



- ▶ un paramètre  $\theta_i$  par connexion
- ▶ pour calculer ces points, même principe que pour le perceptron

# Réseaux de neurones profonds

- ▶ Même principe mais avec des dizaines, voire des centaines de couches cachées.

# Réseaux de neurones profonds

- ▶ Même principe mais avec des dizaines, voire des centaines de couches cachées.
- ▶ Réseaux récurrents.

# Réseaux de neurones profonds

- ▶ Même principe mais avec des dizaines, voire des centaines de couches cachées.
- ▶ Réseaux récurrents.
- ▶ Réseaux à convolution.

# Réseaux de neurones profonds

- ▶ Même principe mais avec des dizaines, voire des centaines de couches cachées.
- ▶ Réseaux récurrents.
- ▶ Réseaux à convolution.

# Réseaux de neurones profonds

- ▶ Même principe mais avec des dizaines, voire des centaines de couches cachées.
- ▶ Réseaux récurrents.
- ▶ Réseaux à convolution.
- ▶ Typiquement, le réseau est composé (implicitement) de deux parties :

# Réseaux de neurones profonds

- ▶ Même principe mais avec des dizaines, voire des centaines de couches cachées.
- ▶ Réseaux récurrents.
- ▶ Réseaux à convolution.
- ▶ Typiquement, le réseau est composé (implicitement) de deux parties :
  - ▶ les premières couches calculent une représentation des données  
en combinant de manière non linéaire les attributs des données

# Réseaux de neurones profonds

- ▶ Même principe mais avec des dizaines, voire des centaines de couches cachées.
- ▶ Réseaux récurrents.
- ▶ Réseaux à convolution.
- ▶ Typiquement, le réseau est composé (implicitement) de deux parties :
  - ▶ les premières couches calculent une représentation des données  
en combinant de manière non linéaire les attributs des données
  - ▶ les dernières couches déterminent la classe en fonction de cette représentation  
typiquement par combinaison linéaire



# Réseaux de neurones profonds

- ▶ Même principe mais avec des dizaines, voire des centaines de couches cachées.
- ▶ Réseaux récurrents.
- ▶ Réseaux à convolution.
- ▶ Typiquement, le réseau est composé (implicitement) de deux parties :
  - ▶ les premières couches calculent une représentation des données  
en combinant de manière non linéaire les attributs des données
  - ▶ les dernières couches déterminent la classe en fonction de cette représentation  
typiquement par combinaison linéaire
- ▶ thème récurrent en analyse de donnée, apprentissage automatique : utiliser une bonne représentation, adaptée à la tâche à résoudre, est essentiel pour le succès.

# Réseaux de neurones profonds

- ▶ Même principe mais avec des dizaines, voire des centaines de couches cachées.
- ▶ Réseaux récurrents.
- ▶ Réseaux à convolution.
- ▶ Typiquement, le réseau est composé (implicitement) de deux parties :
  - ▶ les premières couches calculent une représentation des données  
en combinant de manière non linéaire les attributs des données
  - ▶ les dernières couches déterminent la classe en fonction de cette représentation  
typiquement par combinaison linéaire
- ▶ thème récurrent en analyse de donnée, apprentissage automatique : utiliser une bonne représentation, adaptée à la tâche à résoudre, est essentiel pour le succès.
- ▶ les réseaux profonds nécessitent beaucoup d'exemples.

# Classification supervisée

## Régression logistique

- ▶ Soit  $p$  la probabilité d'être bleu
- ▶ On cherche des paramètres  $\theta_0 \dots \theta_P$  tels que  $\log \frac{p}{1-p} = \sum_i \theta_i a_i$

# Classification supervisée

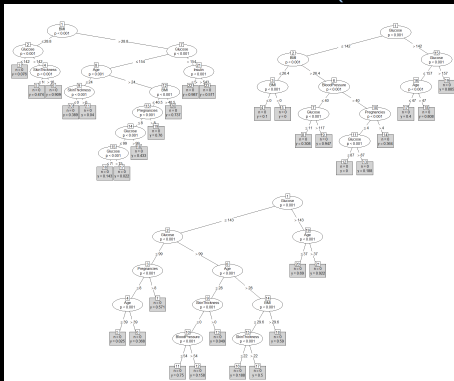
## Forêts

- ▶ Forêt = ensemble d'arbres (de décision)

# Classification supervisée

## Forêts

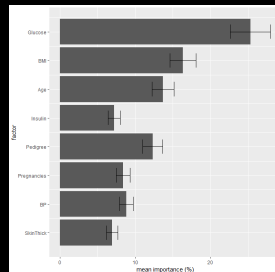
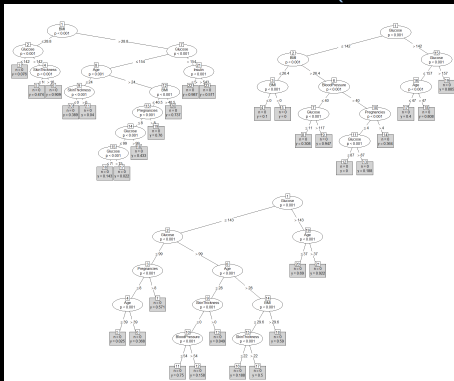
- Forêt = ensemble d'arbres (de décision)



# Classification supervisée

## Forêts

- Forêt = ensemble d'arbres (de décision)

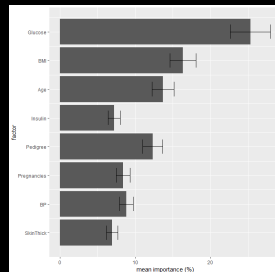
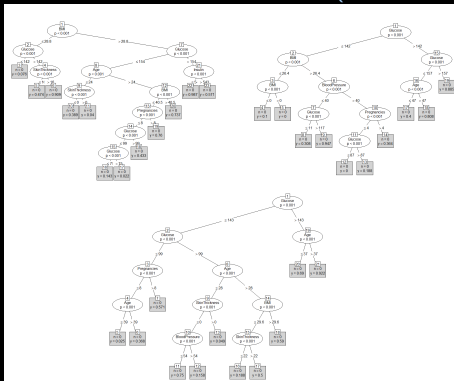


- Différents arbres sélectionnent différents attributs.

# Classification supervisée

## Forêts

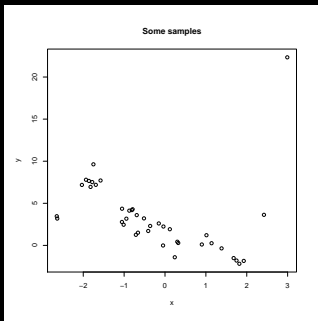
- ▶ Forêt = ensemble d'arbres (de décision)



- ▶ Différents arbres sélectionnent différents attributs.
- ▶ Fonctionnent avec un nombre réduit d'exemples.

# Apprentissage supervisé

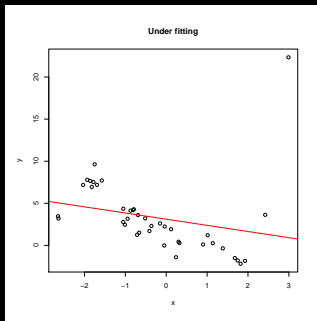
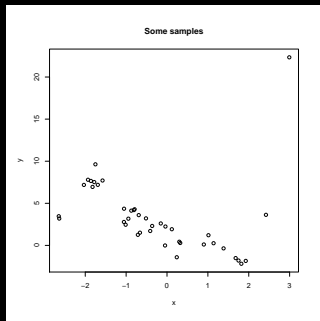
## Sur-apprentissage





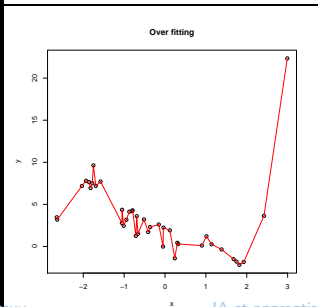
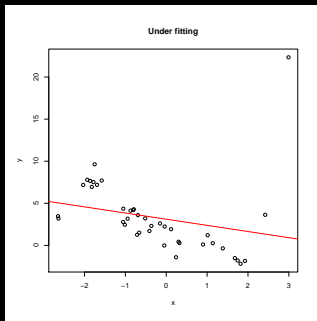
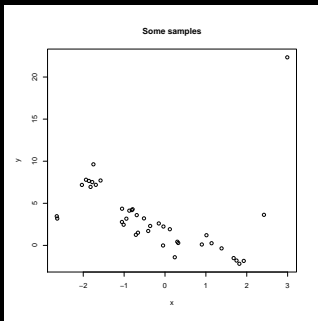
# Apprentissage supervisé

## Sur-apprentissage



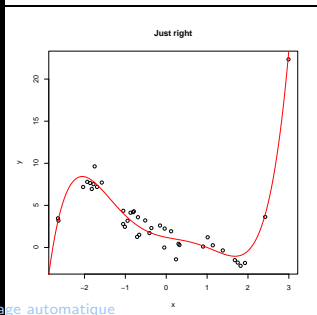
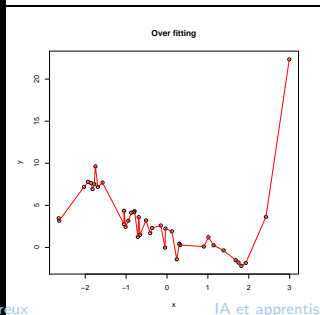
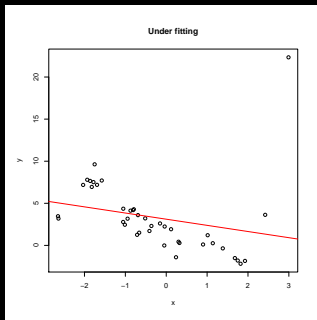
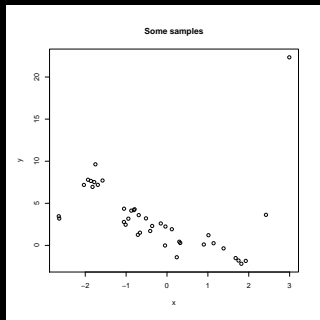
# Apprentissage supervisé

## Sur-apprentissage



# Apprentissage supervisé

## Sur-apprentissage

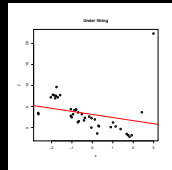


# Apprentissage supervisé

## Sur-apprentissage

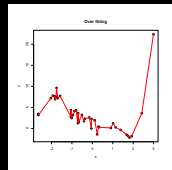
Sous-ajustement/apprentissage (*under-fitting*) : modèle trop simple pour  $y$ .

Le modèle n'a pas assez de paramètres pour s'ajuster à la complexité des données.

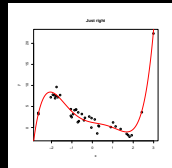


Sur-ajustement/apprentissage (*over-fitting*) : modèle trop complexe pour  $y$ .

Le modèle a trop de paramètres pour la complexité du jeu de données.



Le bon modèle retire juste le bruit.



# Segmentation, catégorisation, *clustering*, *embedding*

## Apprentissage non supervisé

Segmentation de données, catégorisation de données, *Clustering* :

- ▶ Objectif : regrouper les données qui se ressemblent.
- ▶ Une distance est au cœur de ces méthodes.
- ▶ Question : combien y a-t-il de groupes ?

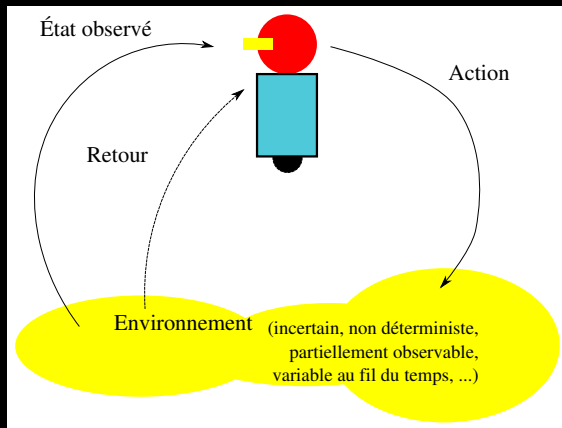
*Embedding* :

- ▶ Objectif : trouver la variété où sont localisées les données.

Cas particulier : ACP.

# Prise de décision séquentielle dans l'incertain

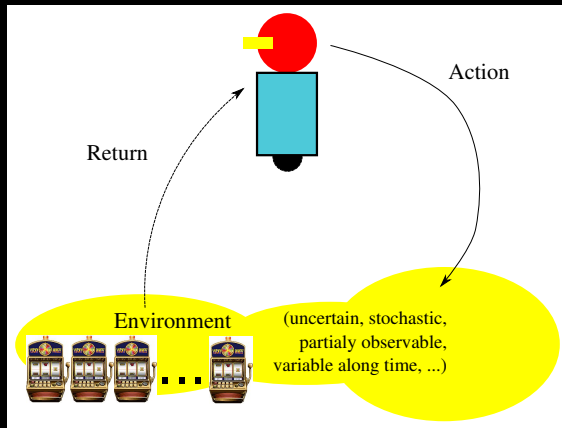
## Apprentissage par renforcement et bandits



- ▶ L'apprentissage supervisé est un outil pour l'AR.
- ▶ Utilisation de réseaux de neurones.

# Prise de décision séquentielle dans l'incertain

## Bandits



- ▶ Formalisme introduit en 1932 dans le cadre des essais cliniques.
- ▶ Nombreuses applications, notamment en médecine.

# Message pour la maison

- ▶ Beaucoup de modèles d'apprentissage supervisé : ils ont chacun leurs qualités et leurs défauts.
- ▶ Face à une étude de cas réelle : essayer plusieurs méthodes et comparer les résultats.
- ▶ Pour un jeu de données, tout est dans la distance (explicite ou implicite) utilisée par le modèle/algorithme.  
Cette distance est fonction des attributs des données.
- ▶ Les bandits fonctionnent.
- ▶ Les meilleurs logiciels d'apprentissage automatique sont tous gratuits et accessibles librement pour les activités publiques, dans les environnements R et python notamment.