4 - Arbres de décision

Jean Massardi - Été 2024

Plan

1 - Arbres de décision

2 - Élagage d’arbre

3 - Forêt d’arbres

Plan

1 - Arbres de décision

2 - Élagage d’arbre

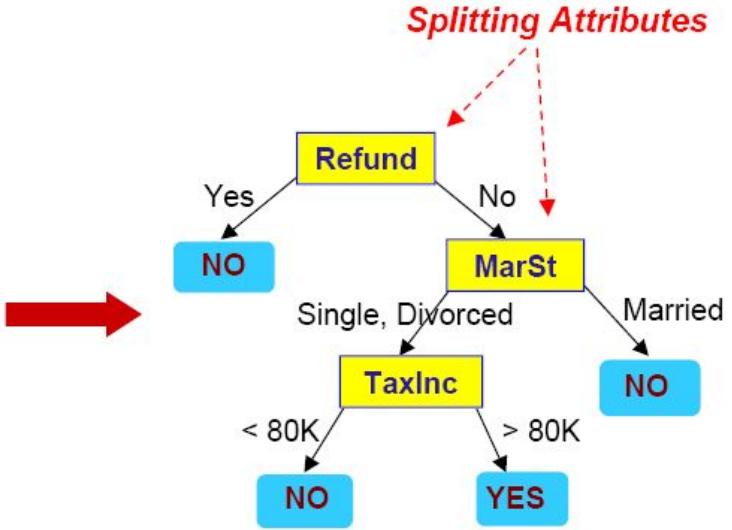
3 - Forêt d’arbres

Définition et Principes

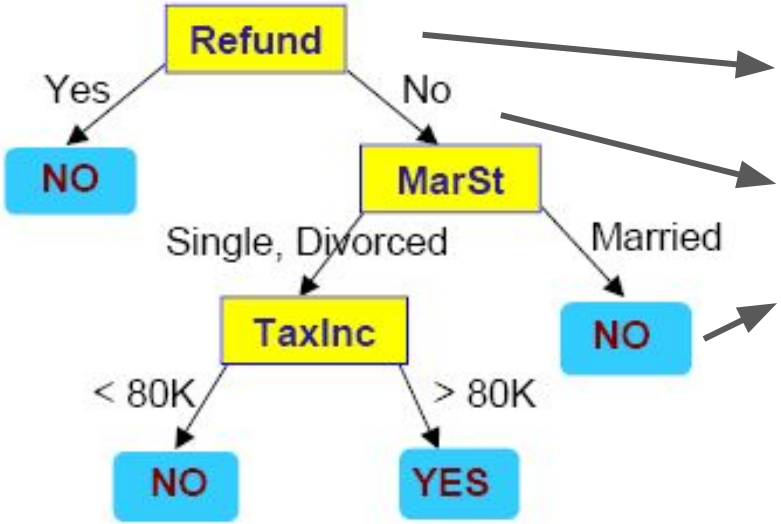
- Méthode de classification plutôt intuitive :
  - Un objet appartient à une catégorie si il répond à plusieurs critères.
  - Une méthode de classification consisterait donc à enchaîner des questions sous la forme de SI - ALORS.
  - L'enchaînement se fait de manière conditionnelle, la réponse à une question précédente indique la question suivante
- Le raisonnement est analogue à

Définition et Principes

Tid	Refund	Marital Status	Taxable Income	Cheat
	categorical	categorical	continuous	class
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

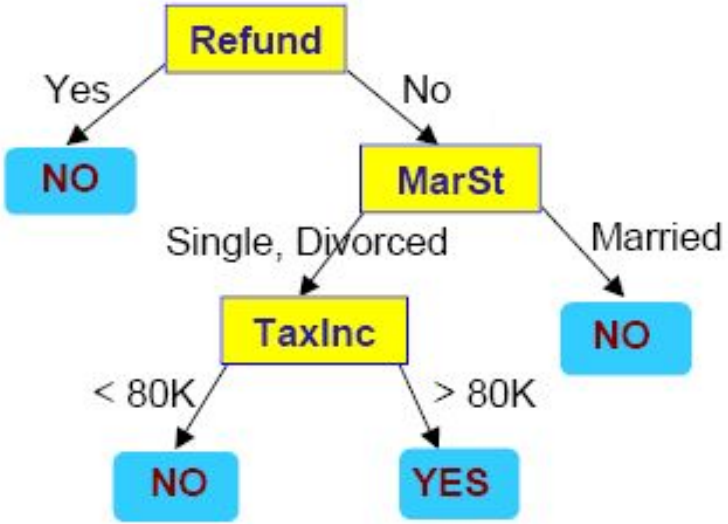


Définition



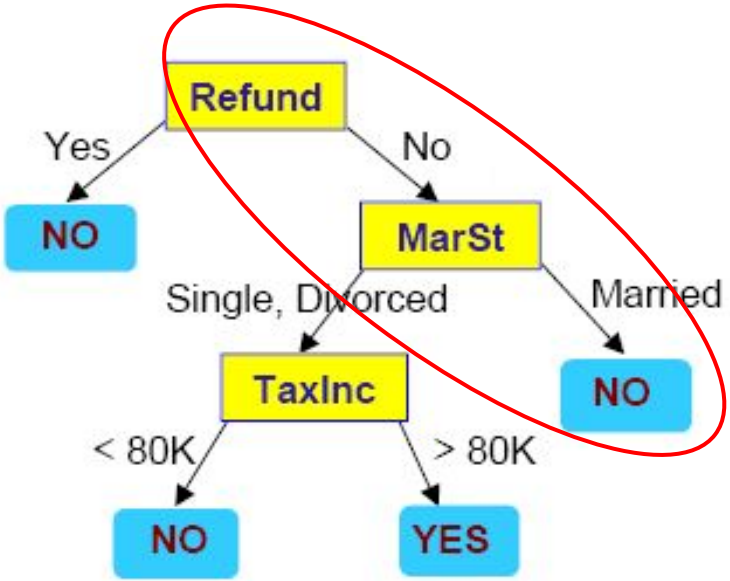
- **Noeuds** : attributs sur lequel un test est réalisé
- **Branches** : Valeur de l'attribut
- **Feuilles** : classe de l'objet

Définition



Refund	MarSt	TaxInc	Class
No	Married	135K	**?**

Définition

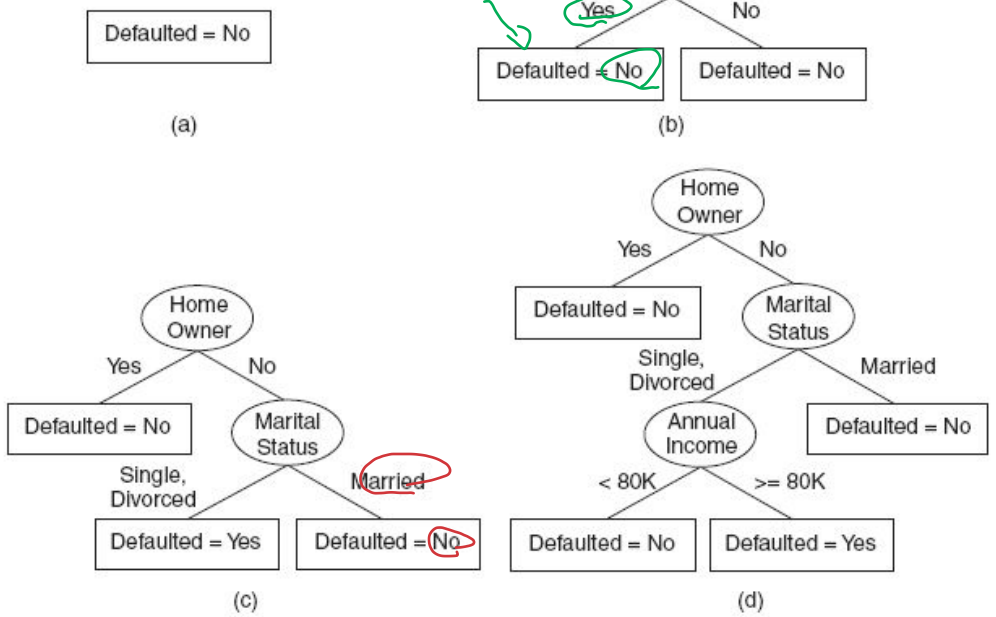


Refund	MarSt	TaxInc	Class
No	Married	135K	**?**

Algorithmes

married no
tous les yes sont no alors

Tid	Home Owner	Marital Status	Annual Income	Defaulted Borrower
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes



Algorithmes (de hunt)

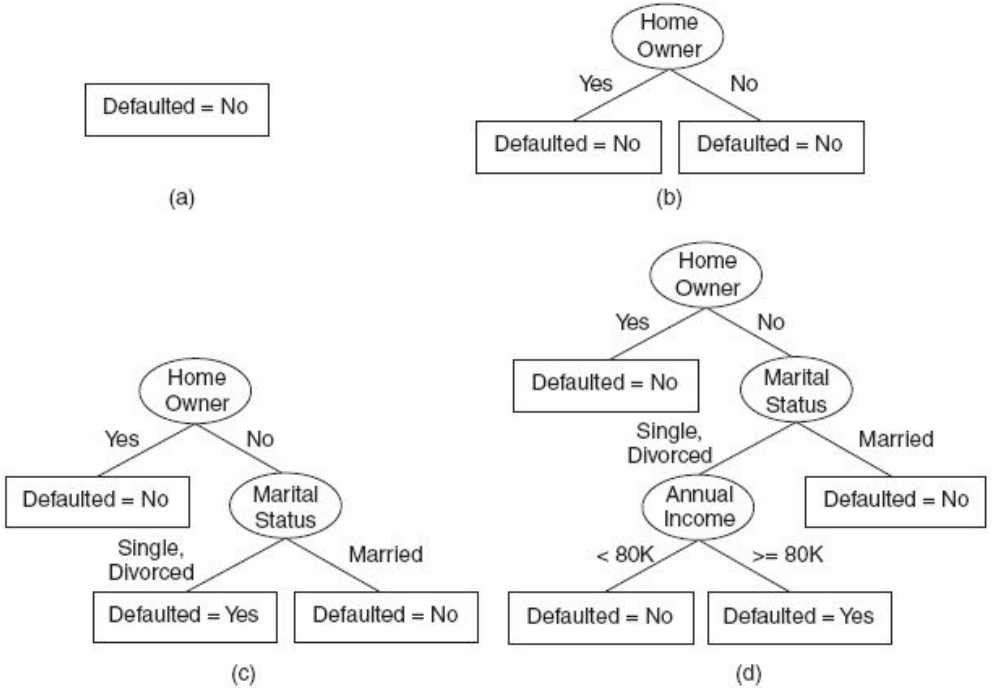
Soit ***D*** un ensemble d'apprentissage

CréerArbre(***D***):

> ***T*** <- noeud vide avec tous les exemples
>
> Tant qu'il y'a des noeuds vide dans l'Arbre :
>
> > On prend un noeud vide
> >
> > Si tous les exemples issus de ce noeud sont de la même classe, le noeud devient une feuille.
> >
> > Sinon on choisit le meilleur attribut, qu'on affecte au noeud. On crée une branche pour chaque valeur de l'attribut et on partitionne les données en fonction de ces valeurs. En bout de branches on ajoute des noeuds vides.

Algorithmes

Tid	Home Owner	Marital Status	Annual Income	Defaulted Borrower
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes



Indice du Gini

- Il existe plusieurs mesure de qu'elle est le meilleur attribut. Nous allons nous concentrer sur l'indice du Gini (Gini Index)
- Le Gini donne une évaluation de l'organisation de l'information. Il se base sur une métrique de l'impureté. Sa valeur est comprise entre 0 et 1
- On choisit comme à chaque étage l'attribut avec le Gini le plus bas

↳ le moins homogène (fachiste) 1 riche tous le reste pauvre

↳ homogène (communisme)

le plus proche de '0'

Indice du Gini

Pour un ensemble de données d'apprentissage ***S*** qui contient ***C*** classes, l'indice de Gini de l'ensemble ***S*** est défini comme suit :

ex: yes
no

$$Gini(S) = 1 - \sum_{j=1}^{C} P_j^2$$

Où Pj est la proportion des objets qui appartiennent à la classe ***j***

P de 1 = à tous les objet
0 = à aucun objet

ex:

C
0
1
0
0
1
1
0
0

$1-\left(\frac{2}{8}\right)^2$

$1-\left(\frac{6}{8}\right)^2$

C
a
b
c
c
b
a
c
c

$1-\left(\frac{3}{9}\right)^2$

$1-\left(\frac{2}{9}\right)^2$

$1-\left(\frac{4}{9}\right)^2$

Indice du Gini

Pour un attribut ***A*** ayant ***v*** valeurs, la formule d'indice du Gini est la suivante :

$$Gini(A) = \sum_{l=1}^{v} \frac{n_{av}}{n_a} Gini(S_v)$$

avec :

- ***nav***, le nombre d'exemple associés à la valeur ***v***
- ***na***, le nombre d'exemple associés à toute les valeurs
- ***Gini(Sv)***, la valeur de Gini de l'ensemble associés à la valeur ***v***

attribut = 4
class à 2 valeurs

Exercice

Construire l'arbre de décision à partir de cet ensemble de données :

$Gini(0) = \sum^{v}$ voir plus

1	2	3	4	
Outlook	Tempreature	Humidity	Windy	Class
sunny	hot	high	false	N
sunny	hot	high	true	N
overcast	hot	high	false	P
rain	mild	high	false	P
rain	cool	normal	false	P
rain	cool	normal	true	N
overcast	cool	normal	true	P
sunny	mild	high	false	N
sunny	cool	normal	false	P
rain	mild	normal	false	P
sunny	mild	normal	true	P
overcast	mild	high	true	P
overcast	hot	normal	false	P
rain	mild	high	true	N

1
2

Exercice

$$Gini(Outlouk) = (\frac{5}{14}Gini(sunny)) + (\frac{4}{14}Gini(overcast)) + (\frac{5}{14}Gini(rain))$$

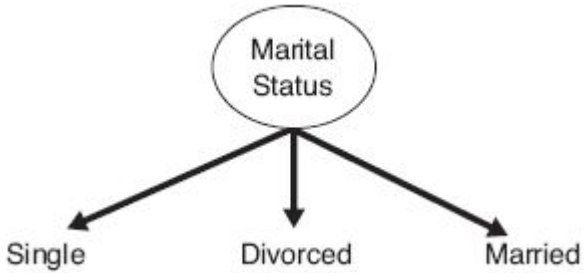
$$Gini(sunny) = 1 - [(\frac{2}{5})^2 + (\frac{3}{5})^2]$$

$0,17(Sunny) \approx 1 - \left(\frac{2}{5}\right)^2 - \left(\frac{3}{5}\right)^2$ ou

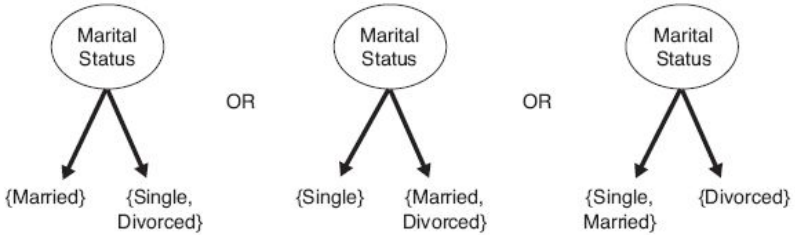
Outlook	Tempreature	Humidity	Windy	Class
sunny	hot	high	false	N
sunny	hot	high	true	N
overcast	hot	high	false	P
rain	mild	high	false	P
rain	cool	normal	false	P
rain	cool	normal	true	N
overcast	cool	normal	true	P
sunny	mild	high	false	N
sunny	cool	normal	false	P
rain	mild	normal	false	P
sunny	mild	normal	true	P
overcast	mild	high	true	P
overcast	hot	normal	false	P
rain	mild	high	true	N

Décomposition des catégories - attributs discrets

Décomposition multibranche



Décomposition binaire



Décomposition des catégories - attributs numérique

La première valeur doit être < à la valeur min

La moyenne entre deux valeurs adjacentes

La dernière valeur doit être > à la valeur min

Cheat	No	No	No	Yes	Yes	Yes	No	No	No	No
Taxable Income										
	60	70	75	85	90	95	100	120	125	220
	55	65	72	80	87	92	97	110	122	172	230
	<= >	<= >	<= >	<= >	<= >	<= >	<= >	<= >	<= >	<= >	<= >	<= >
Yes	0 3	0 3	0 3	0 3	1 2	2 1	3 0	3 0	3 0	3 0	3 0
No	0 7	1 6	2 5	3 4	3 4	3 4	3 4	4 3	5 2	6 1	7 0
Gini	0.420	0.400	0.375	0.343	0.417	0.400	0.300	0.343	0.375	0.400	0.420

$\frac{10}{10}\left(1 - \left(\frac{3}{10}\right)^2 - \frac{7}{10}\right)$

- Choisir la positions qui minimise l'indice de Gini

Overfitting

épouse trop les données.
incapable de suivre sur de la nouvelle information

- Un classificateur avec de bonnes performance sur un ensemble d'apprentissage n'est pas forcément un bon classificateur
- Si un classificateur a une bonne précision sur les données d'apprentissage mais de mauvaise précision sur les données de validation on dit qu'il a un faible pouvoir de généralisation, on parle aussi d'overfitting

Overfitting

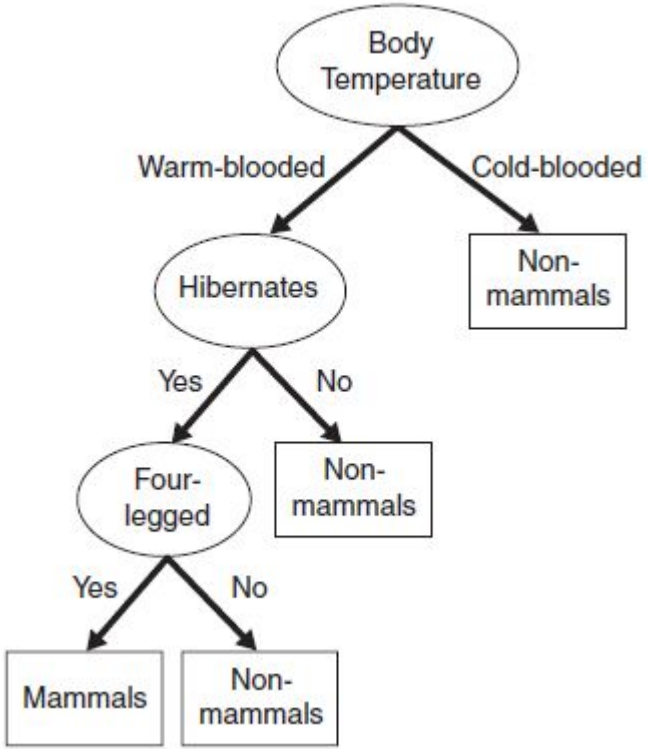
L’overfitting (sur-apprentissage) peut avoir plusieurs causes :

- Un nombre d’exemples trop petit ou pas assez diversifiés
- Du bruit dans les exemples
- Des hyperparamètres mal adapté

Overfitting

Nombre d'exemples trop petit :

Name	Body Temperature	Gives Birth	Four-legged	Hibernates	Class Label
salamander	cold-blooded	no	yes	yes	no
guppy	cold-blooded	yes	no	no	no
eagle	warm-blooded	no	no	no	no
poorwill	warm-blooded	no	no	yes	no
platypus	warm-blooded	no	yes	yes	yes

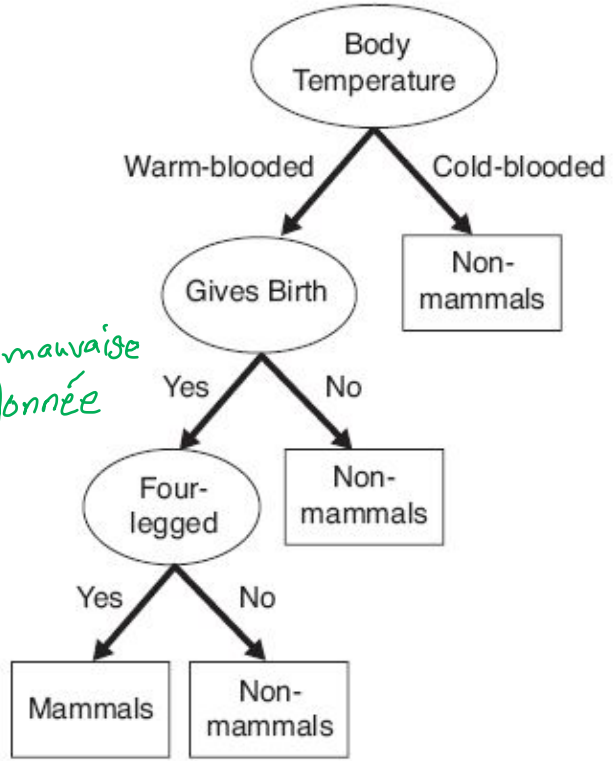


Discussion

Bruit ans les données:

Name	Body Temperature	Gives Birth	Four-legged	Hibernates	Class Label
porcupine	warm-blooded	yes	yes	yes	yes
cat	warm-blooded	yes	yes	no	yes
bat	warm-blooded	yes	no	yes	no*
whale	warm-blooded	yes	no	no	no*
salamander	cold-blooded	no	yes	yes	no
komodo dragon	cold-blooded	no	yes	no	no
python	cold-blooded	no	no	yes	no
salmon	cold-blooded	no	no	no	no
eagle	warm-blooded	no	no	no	no
guppy	cold-blooded	yes	no	no	no

mauvaise donnée



Discussion

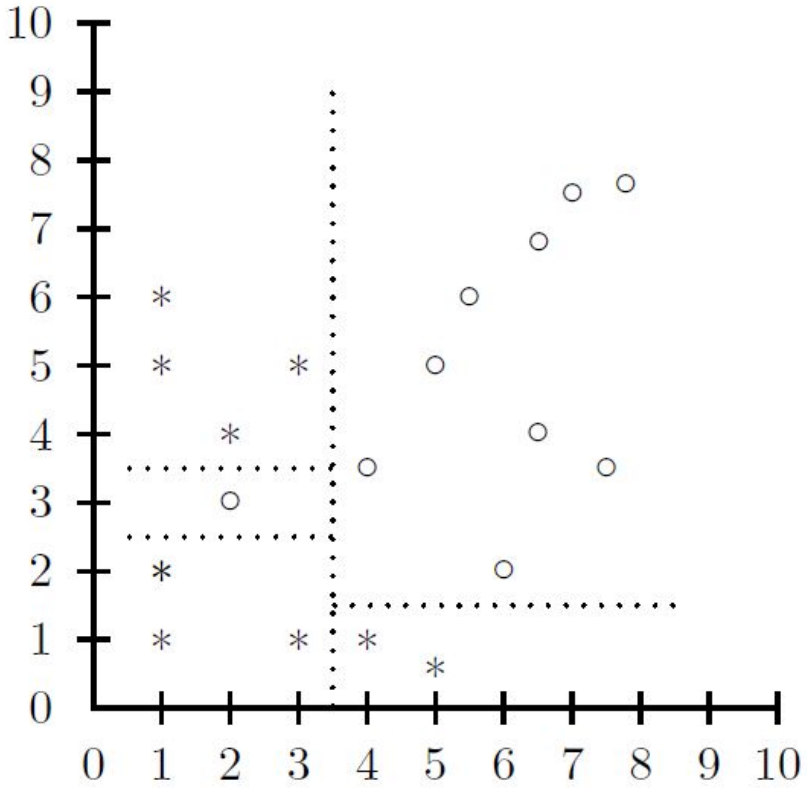
Avantage des arbre de décisions :

- Facile à lire, comprendre et expliquer.
- Pas de paramétrages
- Efficace et plutôt rapide

Limites :

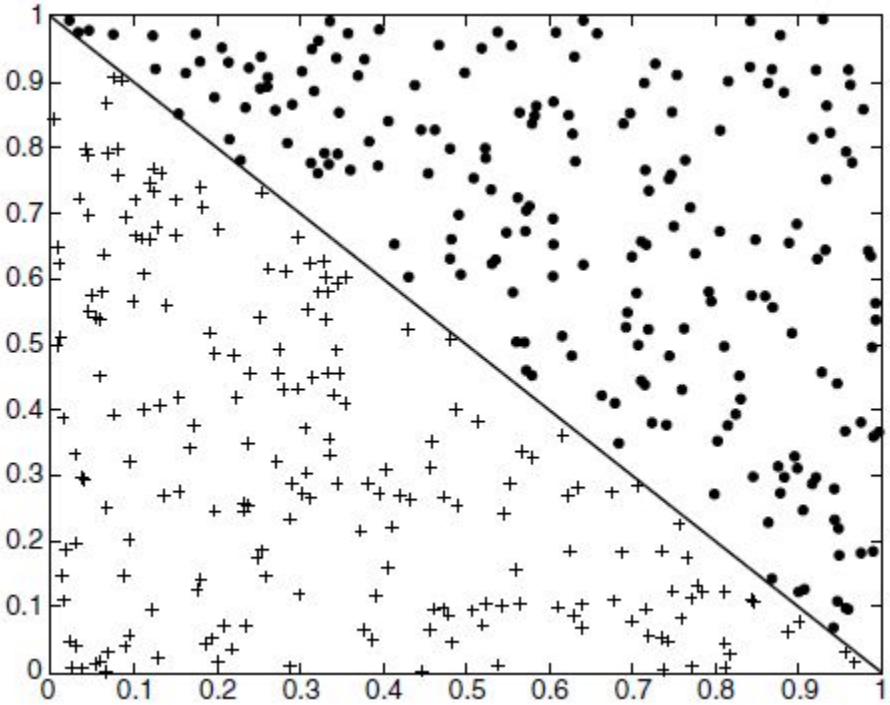
- Pas incrémentale, si on a de nouvelles entrée, il faut recommencer

Discussion



Génère des frontières droites parallèles aux axes des abscisses et des ordonnées.

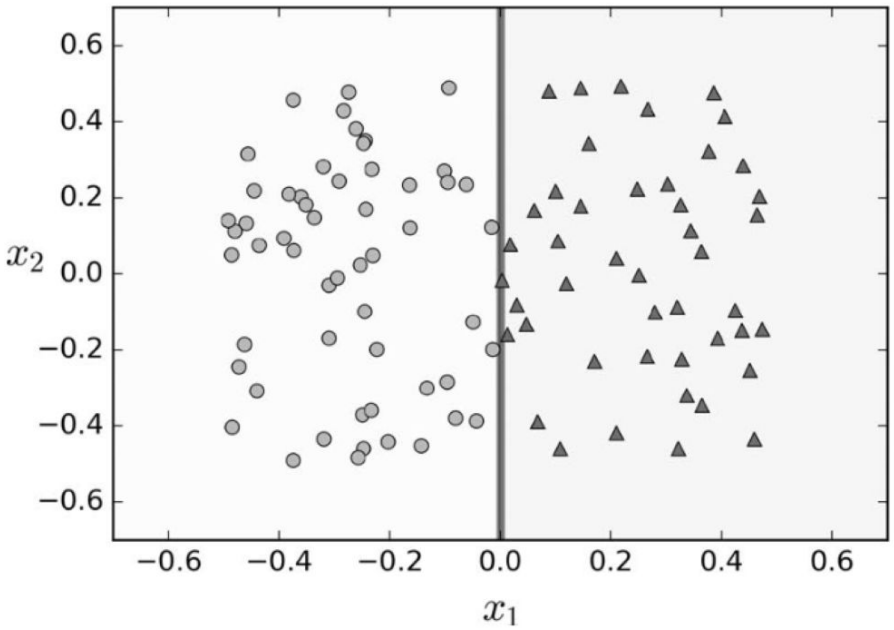
Discussion



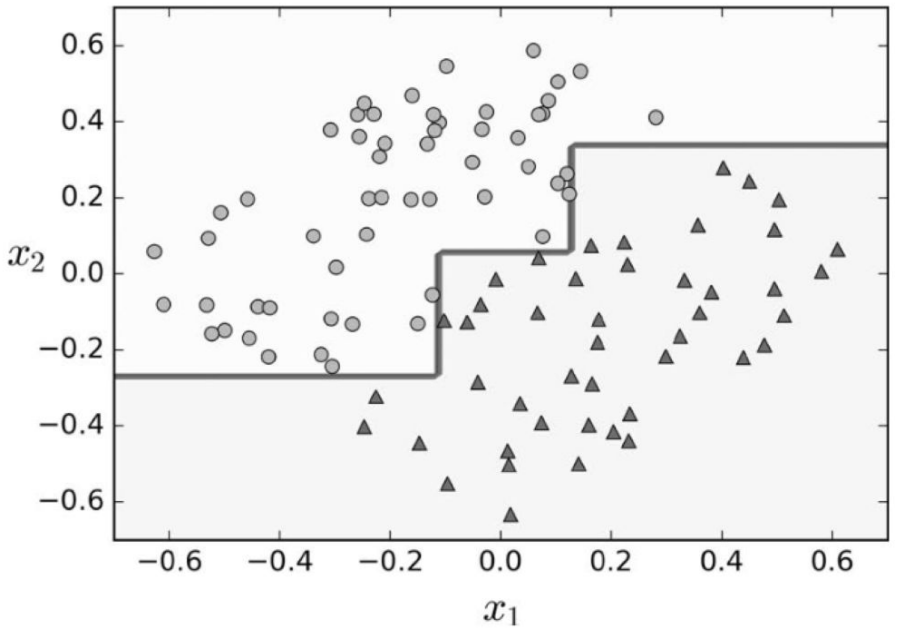
Discussion

même donnée

1 noeud



5 noeuds



Plan

1 - Arbres de décision

2 - Élagage d'arbre

3 - Forêt d'arbres

Prochain cours !!!

Principe

- Lorsqu'on obtient un arbre avec trop de branches ou avec des branches ayant une mauvaise généralisation, plutôt que de reprendre l’apprentissage de 0, on peut être tenté de supprimer les branches concernées. On appelle cette opération de l’élagage.
- Ici on s'intéresse au cas ou l’élagage arrive après la fin du calcul de l’arbre de décision (aussi appelé post-élagage)

Post-élagage critère de taille de l'arbre (nbr de branche)

- Pour faire du post-élagage il faut deux critères :
  - Un critères de qualité d'une branche, qui permet de mettre en évidence deux éléments, son importance dans l'arbre et son taux d'erreur
  - Un critère d'arrêt à l'élagage qui fait un compromis entre le nombre de coupes réalisées et la précision de l'arbre obtenue.
- Pour la suite on écrit ***T0*** l'arbre non élagué, ***T1*** l'arbre avec une coupe, ***T2*** l'arbre avec 2 coupes…

Algorithme

Le critère qualité le plus souvent utilisé pour un noeud est le suivant :

$$w(v,T) = \frac{Err(Coupe(v),T) - Err(v,T)}{n(T)(n(v,T)-1)}$$

Ou :

- Err(Coupe(v),T) = nombre d'erreur dans l'arbre coupé à v
- Err(v,T) = nombre d'erreur dans l'arbre non-élagué
- n(T) = nombre de feuilles dans T
- n(v,T) = nombre de feuilles sous le noeud v

Algorithme

Soit ***T*** un arbre de décision et ***C*** un critère d'arrêt

PostPruning(T,C):

    Te <-T

    Tant que NON C:

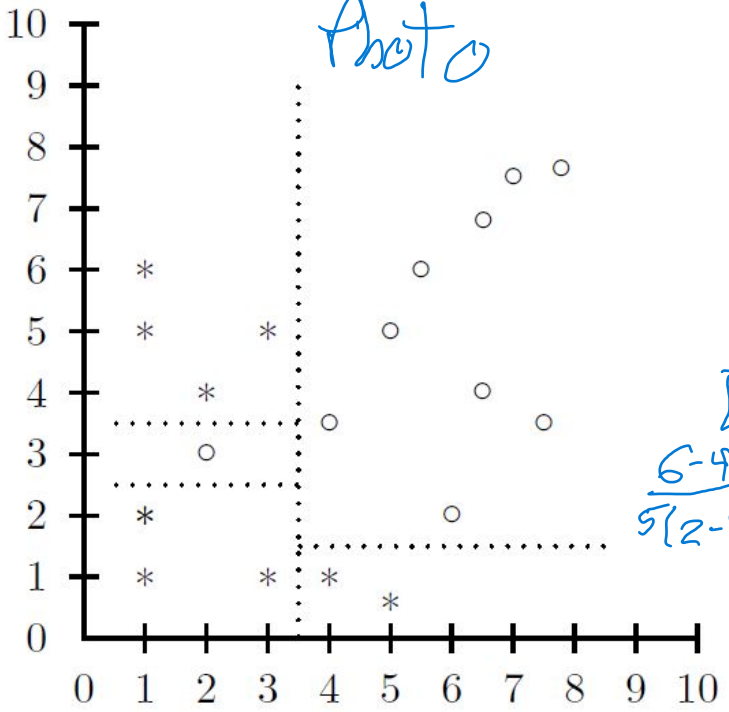
        Pour chaque noeud on calcul le poid w

        Te <- Te dont on remplace le noeud avec le poid le plus faible par une feuille

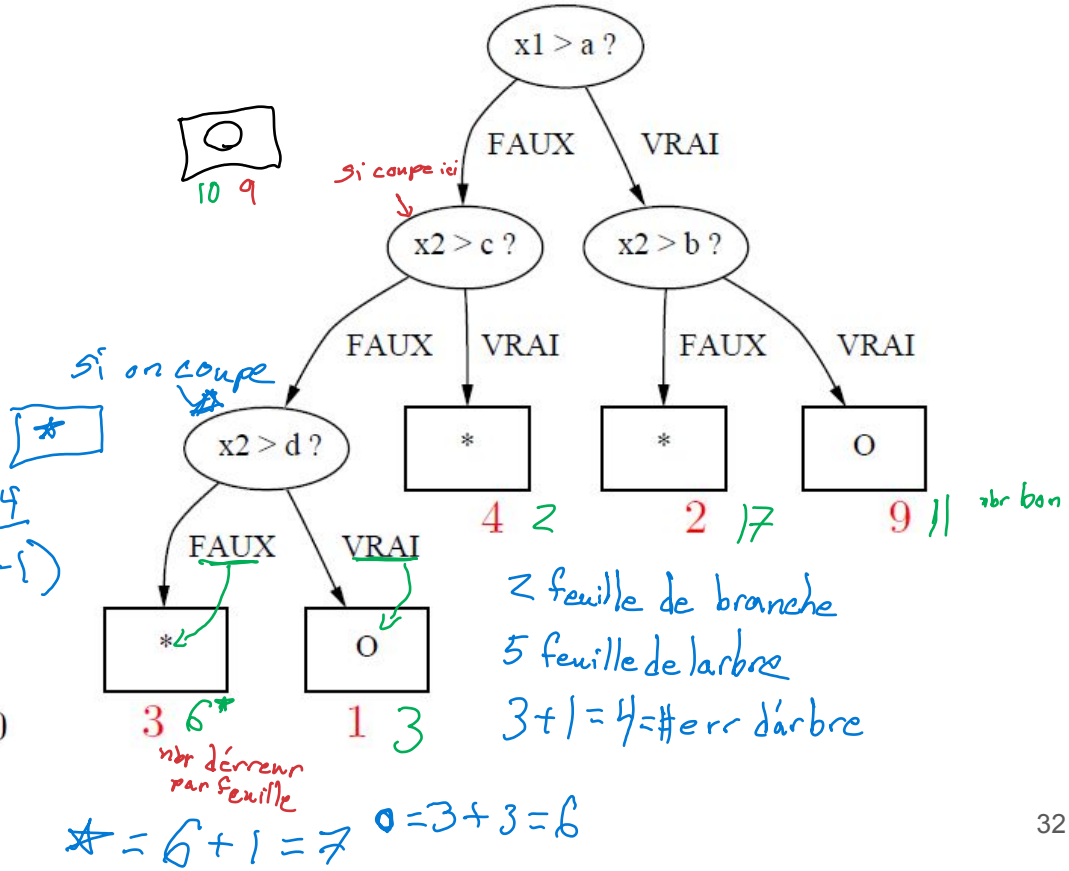
    On retoure Te

on coupe là ou le nombre w = à un nombre le plus petit.

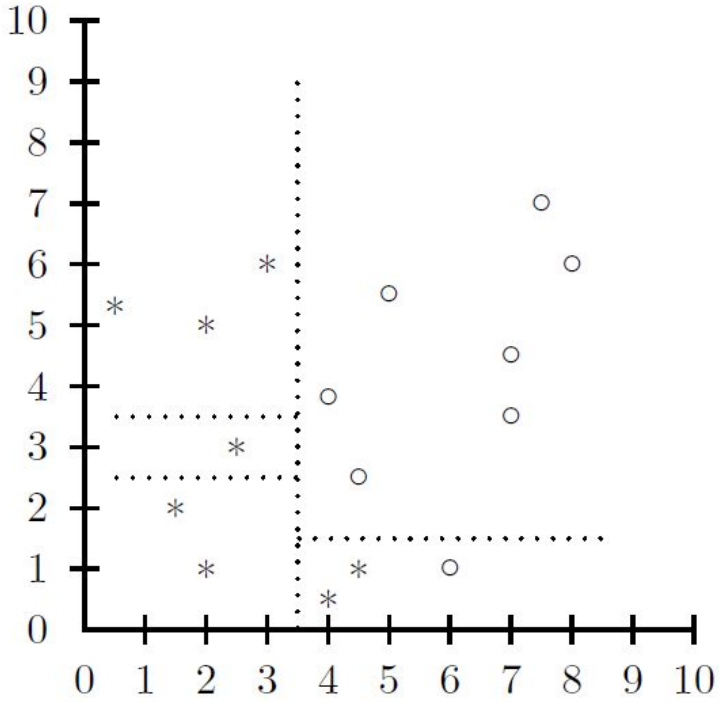
Exemple



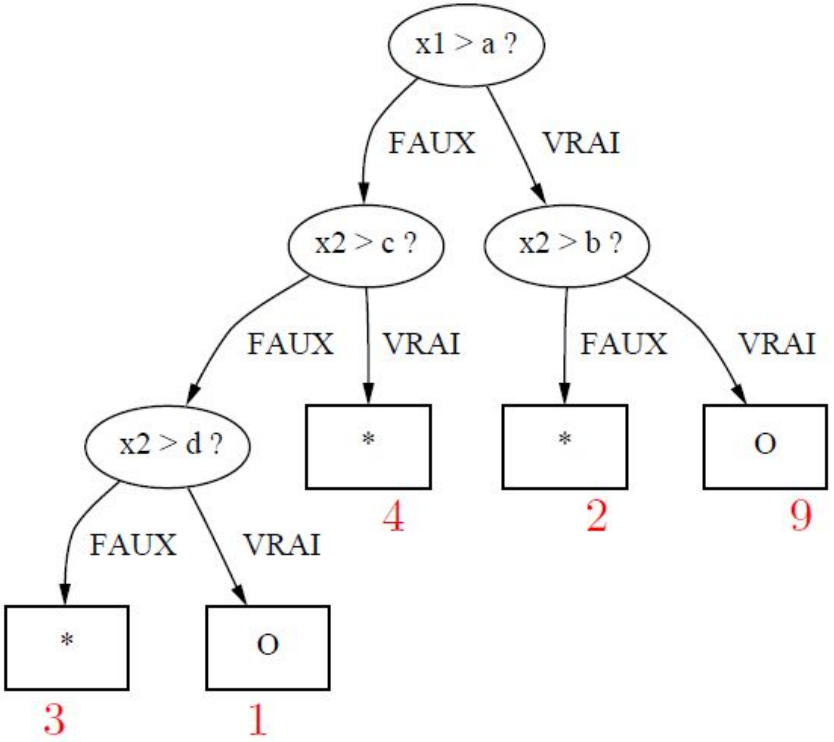
Données d'apprentissages



Exemple



Données de validation



Plan

1 - Arbres de décision

2 - Élagage d’arbre

3 - Forêt d’arbres

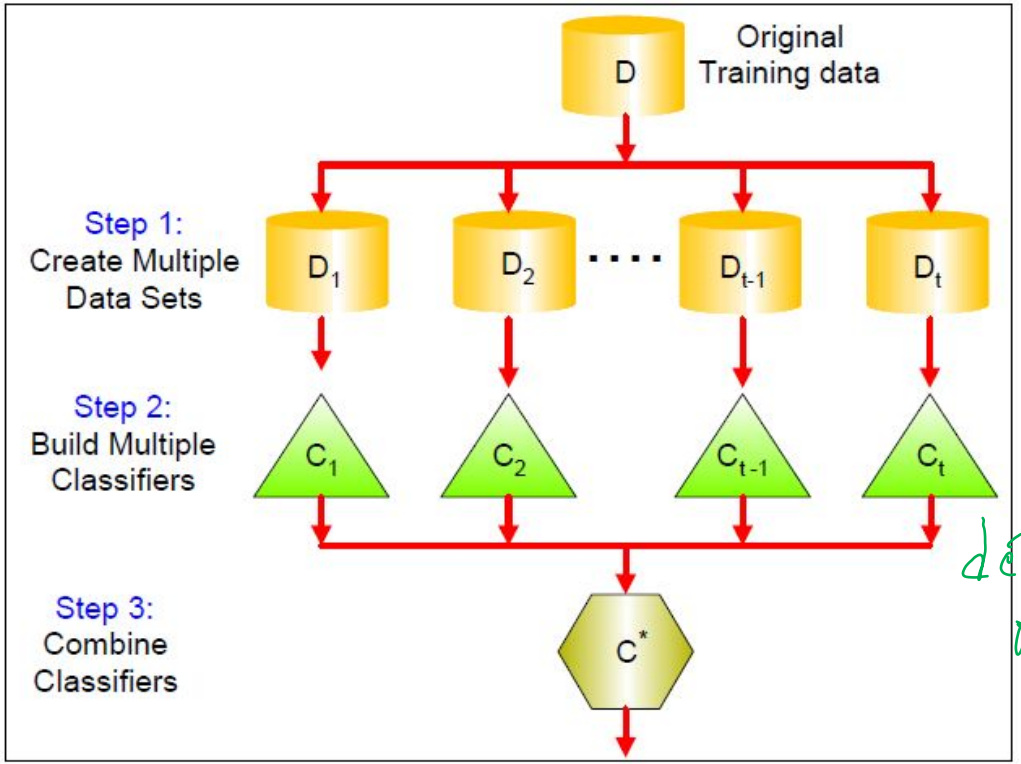
Principes

Idée Générale :

Un classificateur seul peut se tromper, mais plusieurs combinés doivent se tromper moins souvent.

Une façon d’améliorer la précision est de réaliser un vote majoritaire. Chaque classificateur donne son avis sur la classe d'appartenance d’un objet. La classe avec le plus d’avis favorable est considérée comme la bonne.

Fonctionnement



démocratie pour le choix final

Fonctionnement

- La sélection peut se faire selon 2 dimensions :
  - Soit on choisi les attributs au hasard
  - Soit on choisi les enregistrements au hasard
- L'algorithme de classification de base utilisé est l'arbre de décision
- Le nom de forêt d'arbres (random forest) vient du fait qu'on obtient au final une collection d'arbres

Algorithmes

Soit ***S***, un ensemble d'apprentissage comportant ***n*** exemples de ***d*** attributs.

GénérerForêt(***S***,***K,m***):

pour ***i*** dans ***K*** :

Si <- Choisir au hasard ***m*** attributs de ***S***

Ai <- *arbre de décision* obtenue à partir de ***Si***

Retourner la collection des ***Ai***

Caractéristiques

- Fournit généralement une meilleur prédiction
- Permet de supporter des jeux de données ayant beaucoup de dimension et/ou beaucoup d’exemples
- Permet de mettre en évidence certains attributs clefs (utile pour la sélection d‘attributs)
- Nécessite l'entraînement de nombreux arbres de décision
- Fait des choses intéressante dans les cas extrêmes