Arbre et fôrets aléatoires

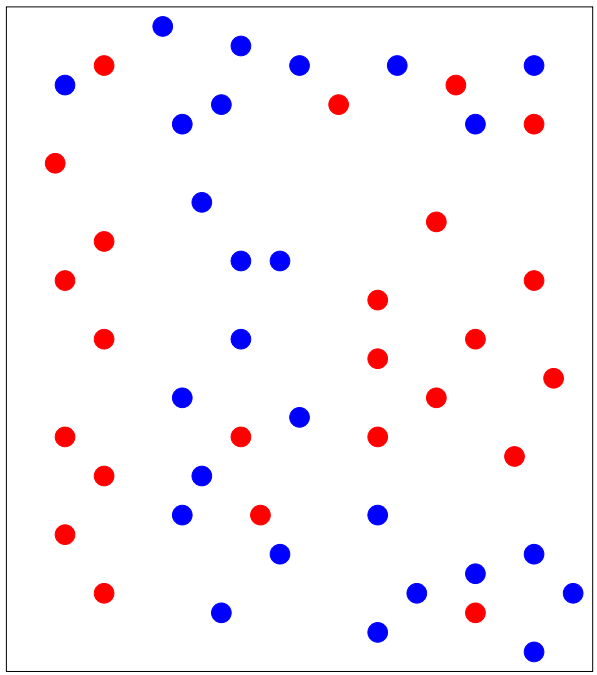
CEPE, Eric Matzner-Løber

Notations

- On se place toujours dans le cas où on cherche à expliquer une variable qualitative Y par p variables explicatives $X_1, \ldots, X_p$.
- Y peut admettre un nombre quelconque de modalités et les variables $X_1, \ldots, X_p$ peuvent être qualitatives et/ou quantitatives.
- Néanmoins, pour présenter la méthode on supposera que Y admet 2 modalités (0 ou 1) et que l'on a simplement 2 variables explicatives quantitatives.

Représentation des données

On dispose de n obervations $(X_1, Y_1), \ldots, (X_n, Y_n)$ où $X_i \in \mathbb{R}^2$ et $Y_i \in \{0, 1\}$.



Objectif : trouver une partition qui sépare "au mieux" les points.

Arbre de décision CART

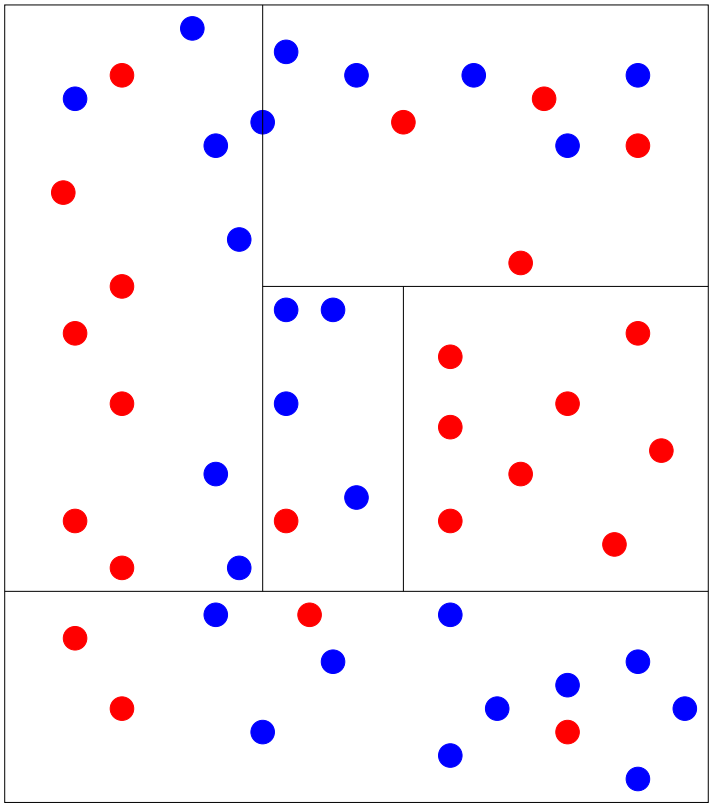
Un arbre binaire de décision CART - Classification And Regression Tree - est un algorithme de moyennage local par partition (moyenne ou vote à la majorité sur les éléments de la partition), dont la partition est construite par divisions successives au moyen d'hyperplans orthogonaux aux axes dépendant des données (X_i, Y_i).

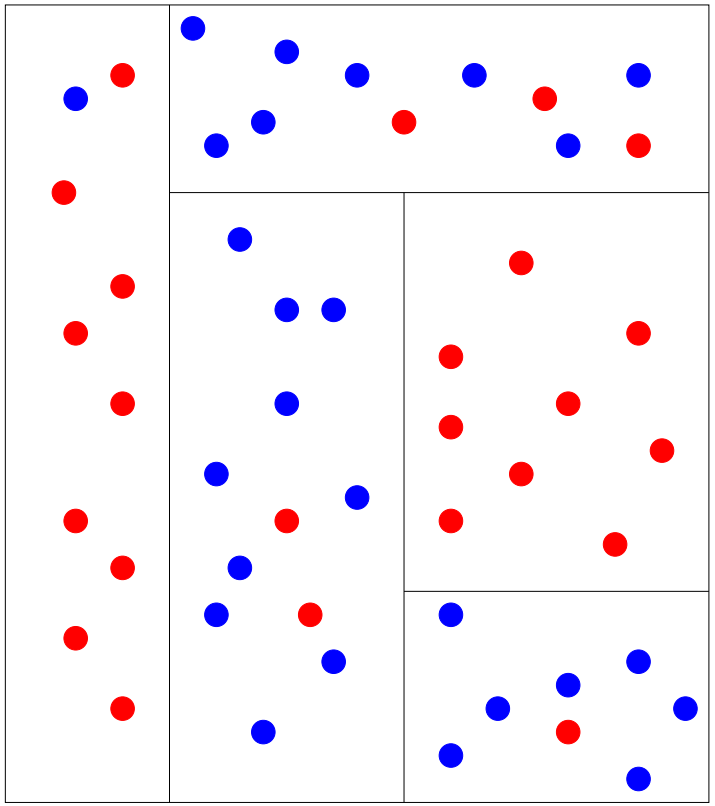
Les éléments de la partition d'un arbre sont appelés les nœuds terminaux ou les feuilles de l'arbre.

L'ensemble constitue le nœud racine. Chaque division définit 2 nœuds, les nœuds fils à gauche et à droite, soit terminal, soit interne, par le choix conjoint : d'une variable X_j et d'une valeur.

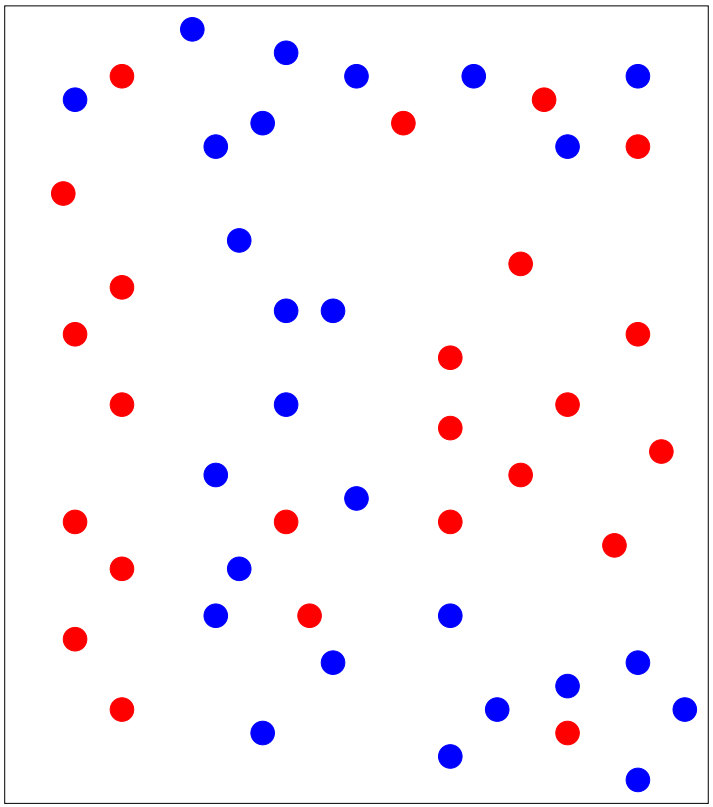
La méthode CART

La méthode CART propose de construire une partition basée sur des divisions successives parallèles aux axes.

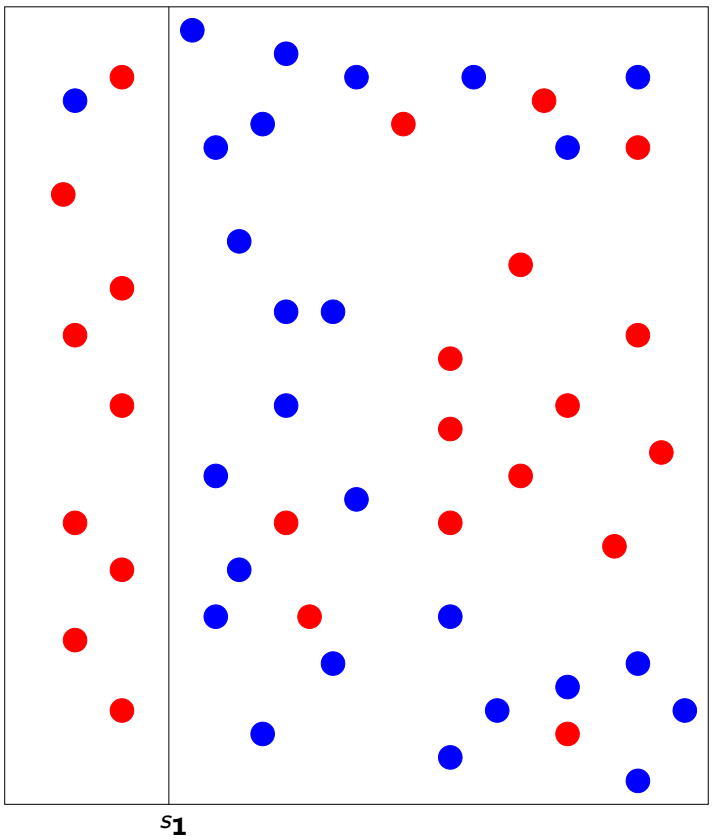




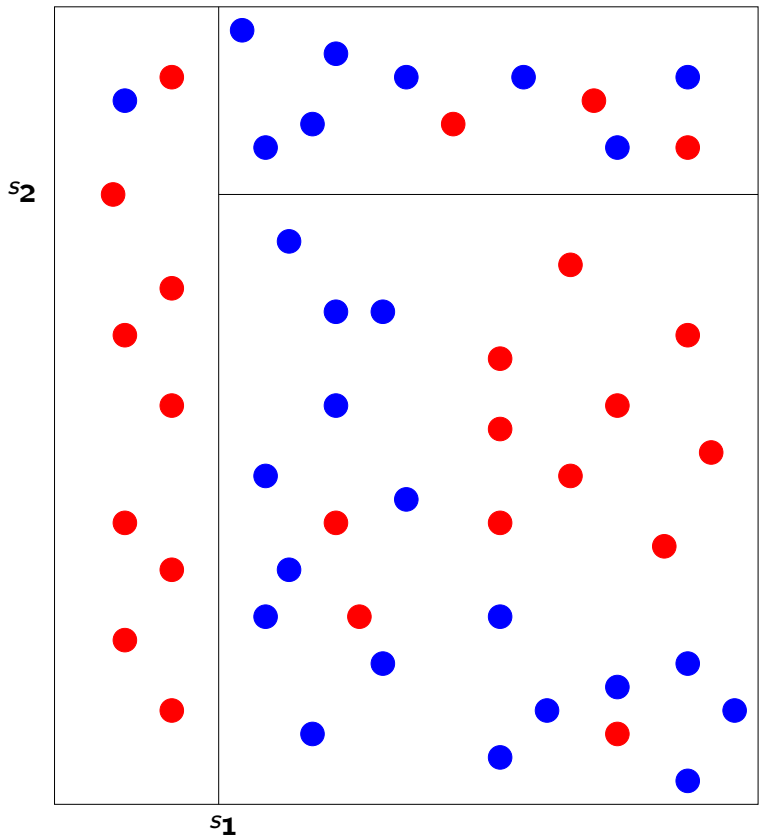
- A chaque étape, la méthode cherche une nouvelle division (une variable et un seuil de coupure) qui optimise une fonction de cout (Entropie de Shannon, indice de Gini...)



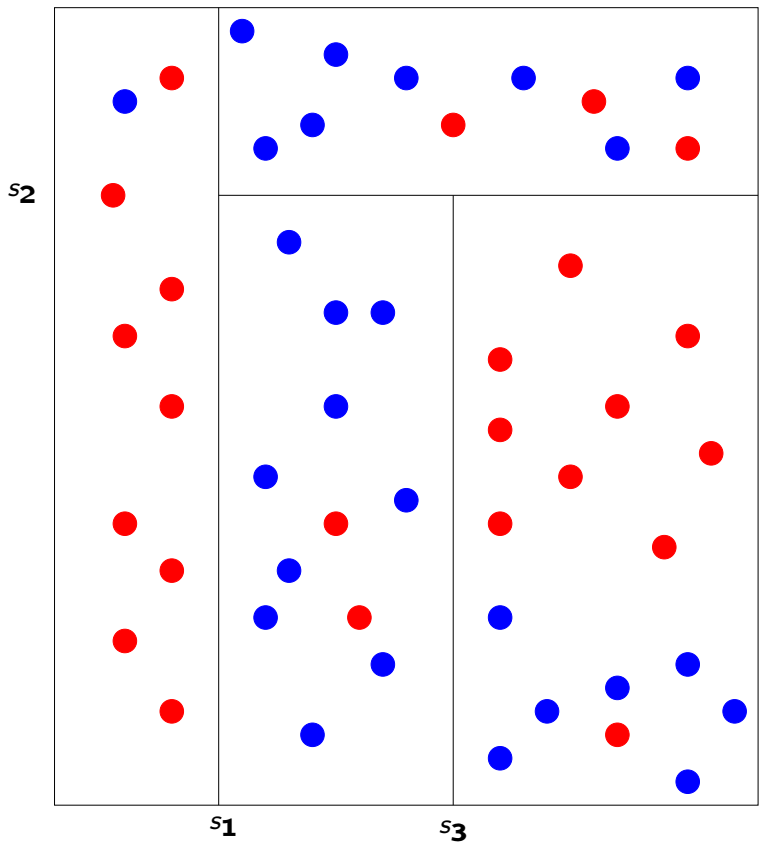
- A chaque étape, la méthode cherche une nouvelle division (une variable et un seuil de coupure) qui optimise une fonction de cout (Entropie de Shannon, indice de Gini...)



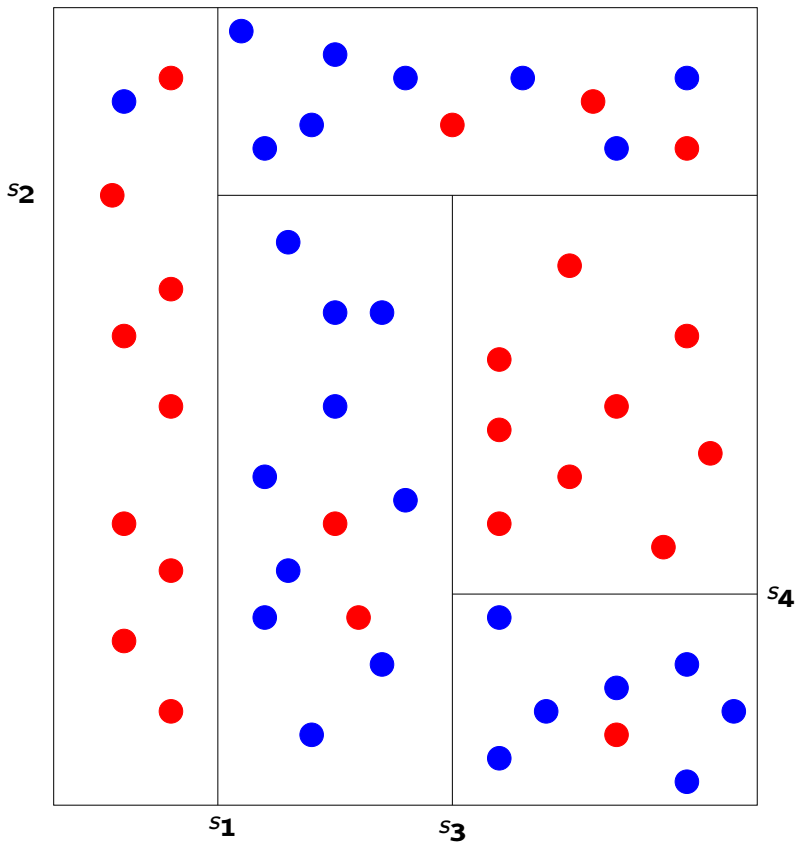
- A chaque étape, la méthode cherche une nouvelle division (une variable et un seuil de coupure) qui optimise une fonction de cout (Entropie de Shannon, indice de Gini...)



- A chaque étape, la méthode cherche une nouvelle division (une variable et un seuil de coupure) qui optimise une fonction de cout (Entropie de Shannon, indice de Gini...)



- A chaque étape, la méthode cherche une nouvelle division (une variable et un seuil de coupure) qui optimise une fonction de cout (Entropie de Shannon, indice de Gini...)

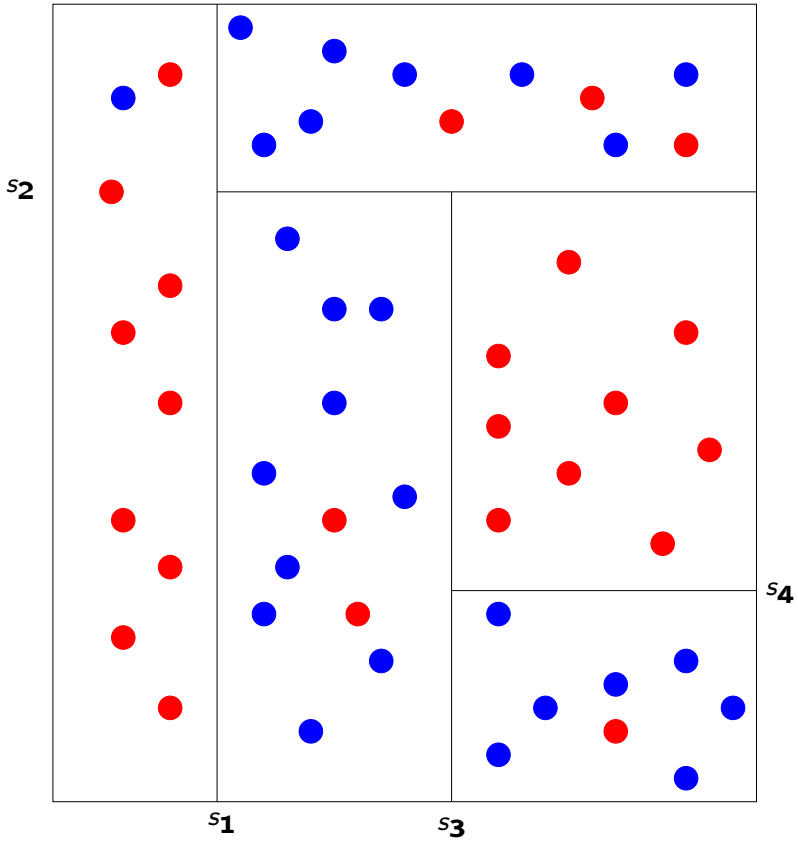


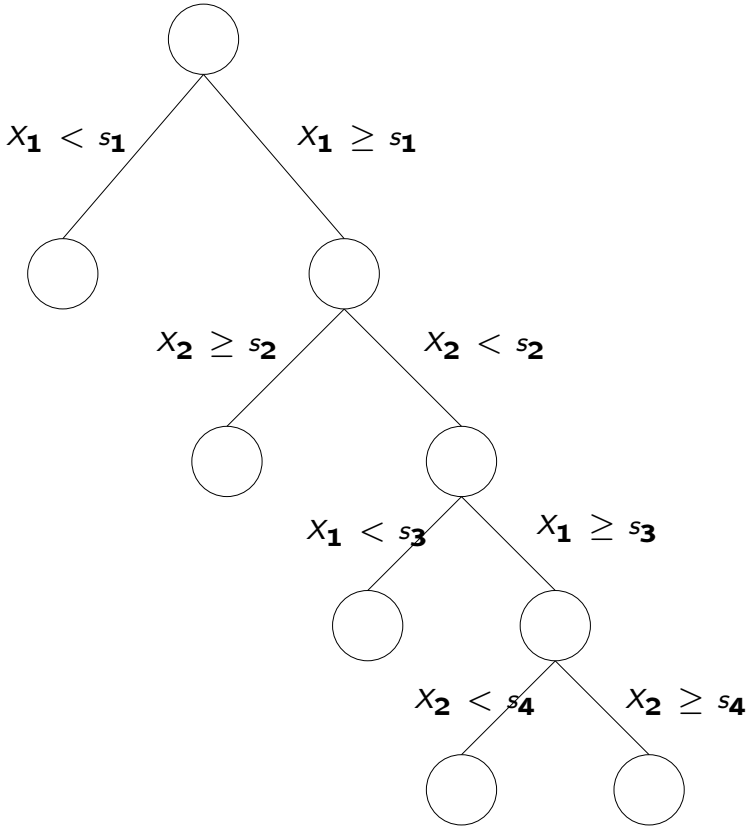
Arbre de décision CART

Ce choix se fait par maximisation du gain d'homogénéité, défini à l'aide d'une fonction d'hétérogénéité H, sur les observations de la variable à expliquer.
Pour un nœud k, si k_g et k_d désignent les nœuds fils à gauche et à droite issus de la division de ce nœud, on choisit la variable explicative et le seuil maximisant :

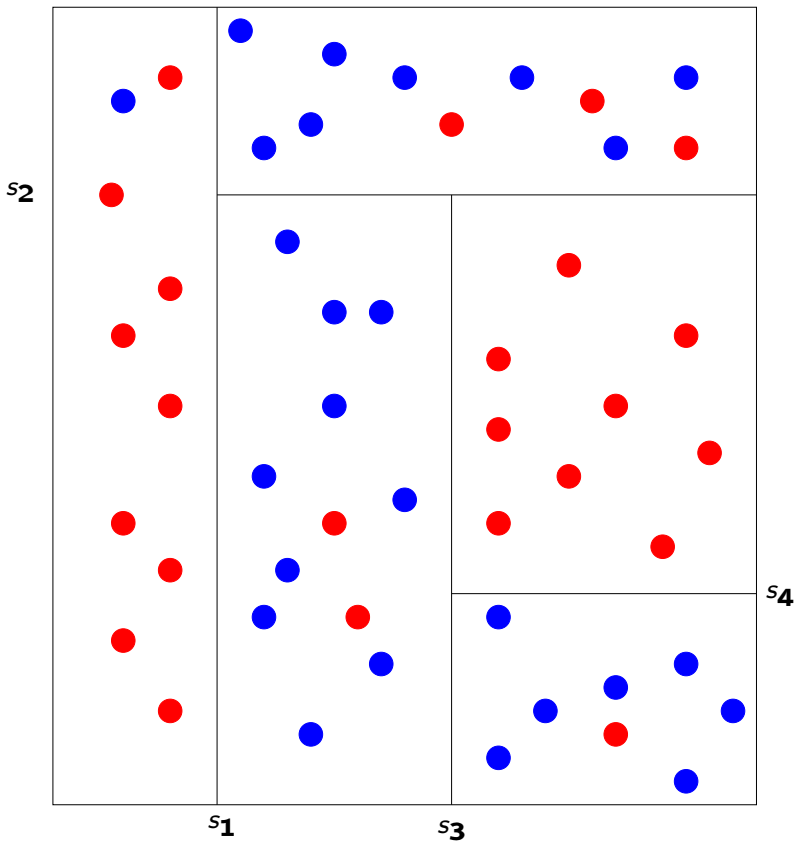
- en régression $H_k - (H_{k_g} + H_{k_d})$ où $H_k =$ la variance empirique des y_i du nœud k,
- discrimination binaire : $H_k - (p_{k_g} H_{k_g} + p_{k_d} H_{k_d})$, avec p_k la proportion d'observations dans le nœud k et $H_k = p_k^1(1 - p_k^1) + p_k^{-1}(1 - p_k^{-1} = 1 - (p_k^1)^2 - p_k^{-1})^2$ où p_k^δ est la proportion de y_i du nœud k égaux à δ : Indice de Gini

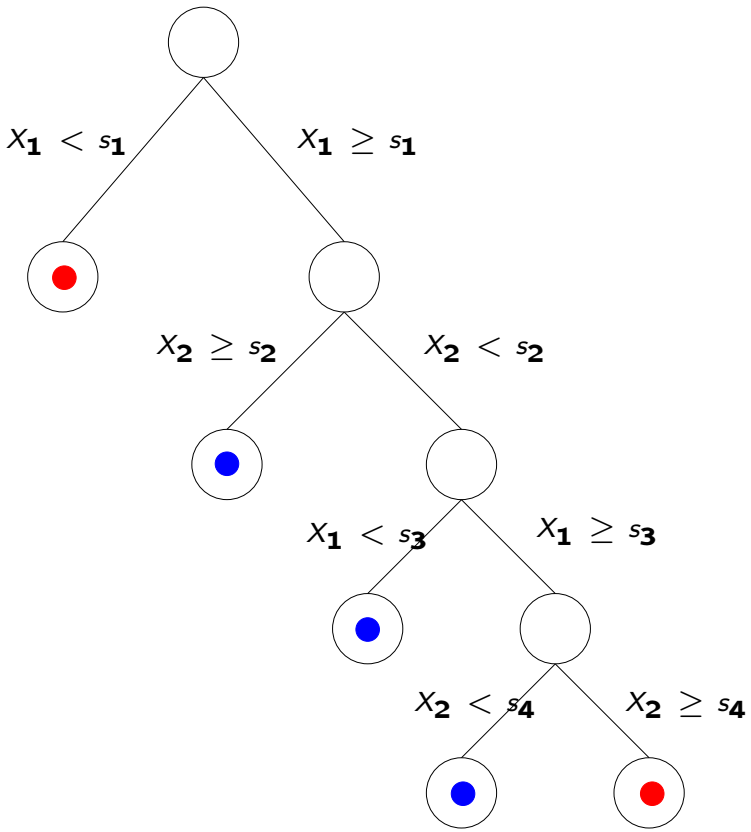
Représentation de l'arbre





Règle d'affectation





Consiste à effectuer un vote à la majorité dans les nœuds terminaux ou à la moyenne dans le cas de la régression.

Critère d'arrêt

Question

A quel moment stopper les divisions ?

- Il existe plusieurs crières d'arrêts.
- Par défaut, la fonction `rpart` de R utilise un critère de type validation croisée.
- Ce critère peut être changé en modifiant la valeur `minsplit` dans la fonction `rpart`.

Attention

- L'arbre souffre d'une grande instabilité (fléau de la dimension, sensibilité à l'échantillon)
- La qualité de prédiction d'un arbre est souvent médiocre comparée à celle d'autres algorithmes.

  Donc agrégation d'arbres ! ! ! cela donne une forêt.

Rmq : l'arbre sélectionné ne dépend que de quelques variables explicatives, et est souvent interprété (à tort) comme une procédure de sélection de variables.

Agrégation d'algorithmes

Les méthodes d'agrégation d'algorithmes de prédiction se décrivent de la façon suivante

- Construction d'un grand nombre d'algorithmes de prédiction simples $\hat{\phi}_b$, $b = 1, \cdots, B$
- Agrégation ou combinaison de ces algorithmes sous la forme : $\hat{\Phi} = \sum_b \alpha_b \hat{\Phi}_b$ ou $signe(\sum_b \alpha_b \hat{\Phi}_b)$

Ici agrégation par bagging ou boosting.
Le **bagging** s'applique à des algorithmes instables, de variance forte,
le **boosting** à des algorithmes fortement biaisés, mais de faible variance.

Définition

- Comme son nom l'indique, une forêt aléatoire est définie à partir d'un ensemble d'arbres.

Définition

Soit $\hat{h}_k(x), k = 1, \ldots, B$ des prédicteurs par arbre $(\hat{h}_k : \mathbb{R}^d \to \{0, 1\})$. Le prédicteur des forêts aléatoires est obtenu par agrégation de cette collection d'arbres :

$$\hat{h}(x) = \frac{1}{B} \sum_{k=1}^{B} \hat{h}_k(x).$$

Pourquoi agréger ?

- On se place dans le modèle de régression

$$Y = m(X) + \varepsilon.$$

- On note

$$\hat{m}(x) = \frac{1}{B} \sum_{k=1}^{B} \hat{m}_k(x)$$

un estimateur de m obtenu en agrégeant $\hat{m}_1, \ldots, \hat{m}_k$.

- $\hat{m}(x)$ et $\hat{m}_k(x)$ sont des variables aléatoires.
- On peut mesurer l'intérêt d'agréger en comparant les performances de $\hat{m}(x)$ à celles de $\hat{m}_k(x)$ (en comparant, par exemple, le biais et la variance de ces estimateurs).

Biais et variance

- Hypothèse : les variables aléatoires $\hat{m}_1, \ldots, \hat{m}_B$ sont i.i.d.
  - **Biais :**

$$\mathbb{E}[\hat{m}(x)] = \mathbb{E}(\hat{m}_k(x)).$$

Conclusion

Agréger ne modifie pas le biais.

  - **Variance :**

$$\mathrm{Var}[\hat{m}(x)] = \frac{1}{B}\mathrm{Var}[\hat{m}_k(x)].$$

Conclusion

Agréger tue la variance.

- Les conclusions précédentes sont vraies sous l'hypothèse que les variables aléatoires $\hat{m}_1, \ldots, \hat{m}_B$ sont i.i.d.
- Les estimateurs $\hat{m}_1, \ldots, \hat{m}_B$ étant construits sur le même échantillon, l'hypothèse d'indépendance n'est clairement pas raisonnable !

Idée

Atténuer la dépendance entre les estimateurs $\hat{m}_k$ en introduisant une nouvelle source d'aléa.

Bagging

Les $\hat{m}_k$ ne vont pas être construits sur l'échantillon $\mathcal{D}_n$, mais sur des échantillons bootstrap $\theta_k(\mathcal{D}_n)$ obtenus en tirant n observations avec remise dans $\mathcal{D}_n$.

Bagging

Entrées :

- $x \in \mathbb{R}^d$ l'observation à prévoir
- un régresseur (arbre CART, 1 plus proche voisin...)
- $\mathcal{D}_n$ l'échantillon
- B le nombre d'estimateurs que l'on agrège.

Pour $k = 1, \ldots, B$:

1. Tirer un échantillon bootstrap $\theta_k(\mathcal{D}_n)$ dans $\mathcal{D}_n$
2. Ajuster le régresseur sur cet échantillon : $\hat{m}_k(x, \theta_k(\mathcal{D}_n))$

Sortie : L'estimateur $\hat{m}_B(x) = \frac{1}{B}\sum_{k=1}^{B} \hat{m}_k(x)$.

Choix du nombre d'itérations

- Deux paramètres sont à choisir : le nombre d'itérations B et le régresseur.
- On montre en utilisant la loi des grands nombres que, lorsque B est grand, $\hat{m}$ se "stabilise".
- Le nombre d'itérations B n'est pas un paramètre à calibrer, il est préconisé de le prendre le plus grand possible en fonction du temps de calcul.

Propriétés

$$\begin{aligned}\mathbb{E}[\hat{m}_B(x)] &= \mathbb{E}[\hat{m}_k(x, \theta_k(\mathcal{D}_n)] \\ \text{Var}[\hat{m}_B(x)] &= \rho(x)\,\text{Var}[\hat{m}_k(x, \theta_k(\mathcal{D}_n)] + \frac{1-\rho(x)}{B}\,\text{Var}[\hat{m}_k(x, \theta_k(\mathcal{D}_n)]\end{aligned}$$

où $\rho(x) = corr(\hat{m}_k(x, \theta_k(\mathcal{D}_n)), \hat{m}_{k'}(x, \theta_{k'}(\mathcal{D}_n)))$ pour $k \neq k'$.

- Bagger ne modifie pas le biais.
- Lorsque B est grand, $\text{Var}[\hat{m}_B(x)] \approx \rho(x)\,\text{Var}[\hat{m}_k(x, \theta_k(\mathcal{D}_n)]$ $\Longrightarrow$ la variance diminue d'autant plus que la corrélation entre les prédicteurs diminue.
- Il est donc nécessaire d'utiliser des estimateurs $\hat{m}_k$ sensibles à de légères perturbations de l'échantillon.
- Les arbres sont connus pour posséder de telles propriétés.

Calculs

Avec même variance pour chaque estimateur

$$
\begin{aligned}
\mathrm{Var}[\hat{m}_B(x)] &= \frac{1}{B^2}\sum_i\sum_j cov(\hat{m}_i,\hat{m}_j) \\
&= \frac{1}{B^2}\sum_i(\sum_{j\neq i} cov(\hat{m}_i,\hat{m}_j)+\mathrm{Var}(\hat{m}_i)) \\
&= \frac{1}{B^2}\sum_i((B-1)\,\mathrm{Var}(\hat{m}_i)\rho+\mathrm{Var}(\hat{m}_i)) \\
&= \frac{1}{B}((B-1)\rho\,\mathrm{Var}(\hat{m}_k)+\mathrm{Var}(\hat{m}_k)) \\
&= \rho(x)\,\mathrm{Var}(\hat{m}_k)+\frac{1-\rho(x)}{B}\,\mathrm{Var}(\hat{m}_k)
\end{aligned}
$$

Retour aux forêts aléatoires

- **Rappels :** Forêts aléatoires = collection d'abres.
- Les forêts aléatoires les plus utilisées sont (de loin) celles proposées par Léo Breiman (au début des années 2000).
- Elles consistent à agréger des arbres construits sur des échantillons bootstrap.
- On pourra trouver de la doc à l'url

  http ://www.stat.berkeley.edu/ breiman/RandomForests/

Algorithme : randomforest

- $x \in \mathbb{R}^d$ l'observation à prévoir ;
- $\mathcal{D}_n$ l'échantillon ;
- B le nombre d'arbres ;
- $m \in \mathbb{N}^\star$ le nombre de variables candidates pour découper un nœud.

Pour $k = 1, \ldots, B$:

1. Tirer un échantillon bootstrap dans $\theta_k(\mathcal{D}_n)$
2. Construire un arbre CART sur cet échantillon bootstrap, chaque coupure est sélectionnée en minimisant la fonction de coût de CART sur un ensemble de m variables choisies au hasard parmi les d. On note $h(., \theta_k(\mathcal{D}_n))$ l'arbre construit.

Sortie : L'estimateur $h(x) = \frac{1}{B} \sum_{k=1}^{B} h(x, \theta_k(\mathcal{D}_n))$.

Commentaires

- Méthode simple à mettre en oeuvre et déjà implémentée sur la plupart des logiciels statistiques (sur R, il suffit de lancer la fonction randomForest du package **randomForest**).

- Avantages :
  1. Permet de fournir des estimations précises sur des données complexes (beaucoup de variables, donnée manquantes...).
  2. Estimateur peu sensible au choix de ses paramètres (B, p...)
- Incovénient : aspect boite noire pour l'estimateur final.

Ajustement des paramètres

On remarque que si m diminue, la variance diminue (la corrélation entre les arbres diminue), mais le biais augmente (les arbres ont une moins bonne qualité d'ajustement).

Compromis biais/variance, le choix optimal de m lié aussi au nombre d'observations dans les nœuds terminaux.

Analyser l'erreur out of bag (what is it ?) **err.rate**

Importance des variables

- Calculer l'erreur out of bag d'un arbre.
- Créer un échantillon out of bag permuté (en permutant aléatoirement les valeurs de la variable explicative X_j dans l'échantillon Out Of Bag) et calculer l'erreur Out Of Bag de l'arbre sur cet échantillon permuté. (**importance**

On recommence pour tous les arbres de la forêt, est ce possible d'étendre ce critère à d'autres méthodes ?

Paramétrage

Par défaut, **randomForest** prend

- un nombre d'observations dans les nœuds terminaux égal à 5 en régression, 1 en discrimination
- un nombre de variables explicatives m égal à $d/3$ en régression et $\sqrt{d}$ en discrimination.