Modèles d'arbres de régression

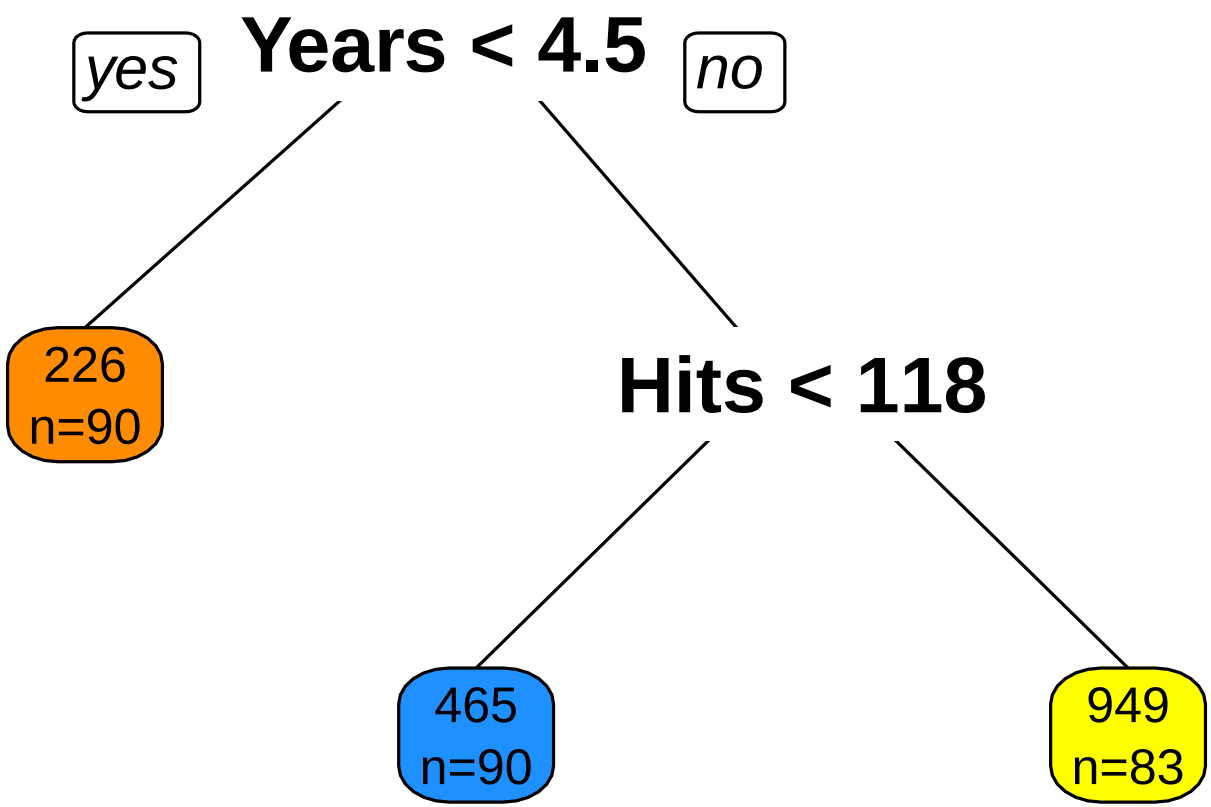
Mini-cours

le 8 février

Introduction

Quoi?

- Déterminer un ensemble de conditions logiques de partition (division) du type **Si...-Alors...** afin de prévoir aussi précisément que possible les valeurs ou classifications prévues des observations.
- Vladimir Guerrero: 7 années, 200 frappes. Prévision de son salaire l'année prochaine?



Contexte de la méthode CART

- Les méthodes dites de partitionnement récursif ou de segmentation datent des années 1960.

- Elles ont été formalisées dans un cadre générique de sélection de modèle par Breiman et col. (1984) [1] sous l'acronyme de **CART: Classification and Regression Tree**.

- L'acronyme CART correspond à deux situations bien distinctes selon que la variable à expliquer, modéliser ou prévoir est
  1. qualitative (classification)
  2. quantitative (régression)

Régression vs. Classification

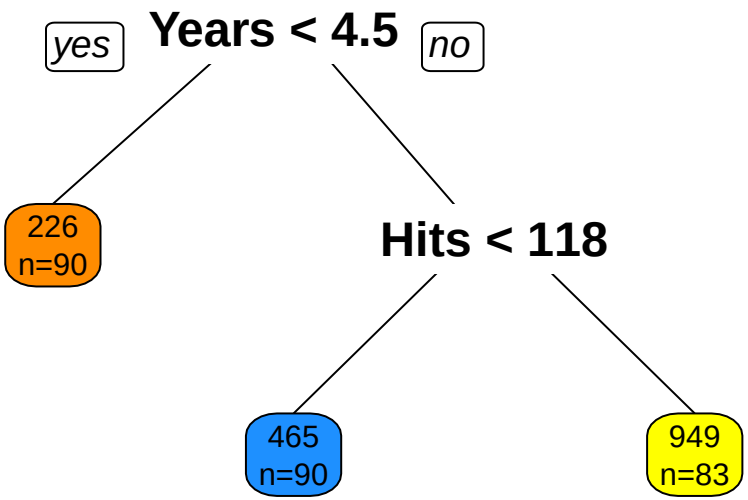


Fig.: Régression

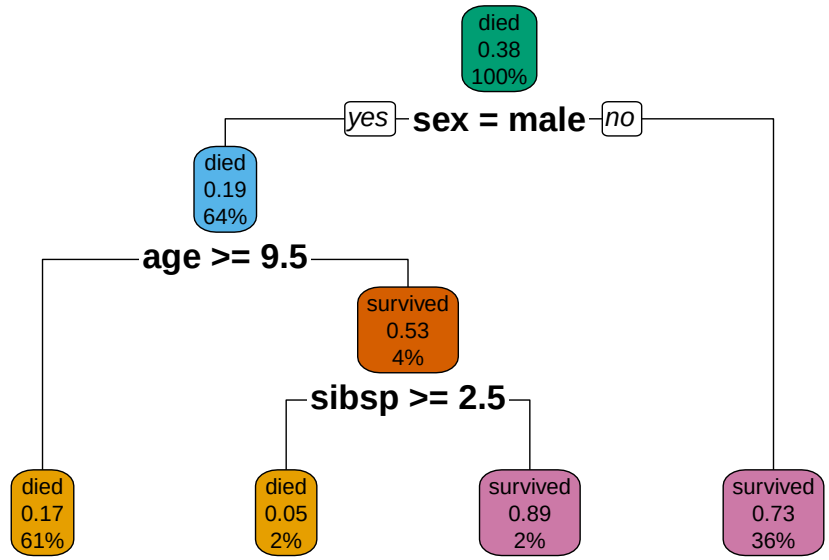


Fig.: Classification

Régression vs. Classification

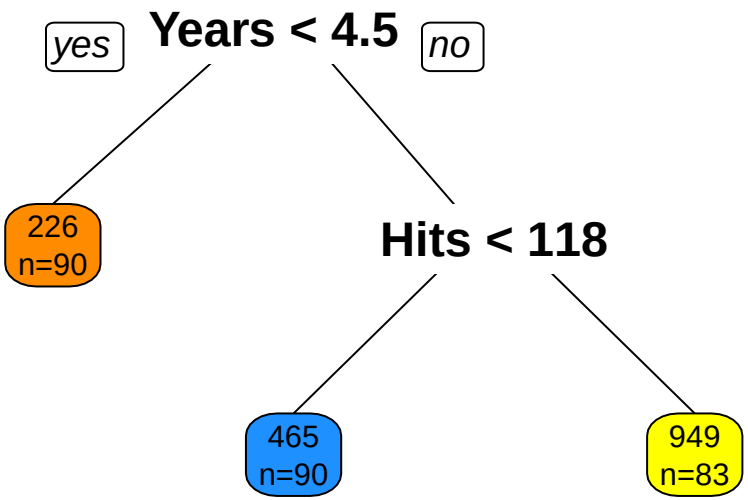


Fig.: Régression

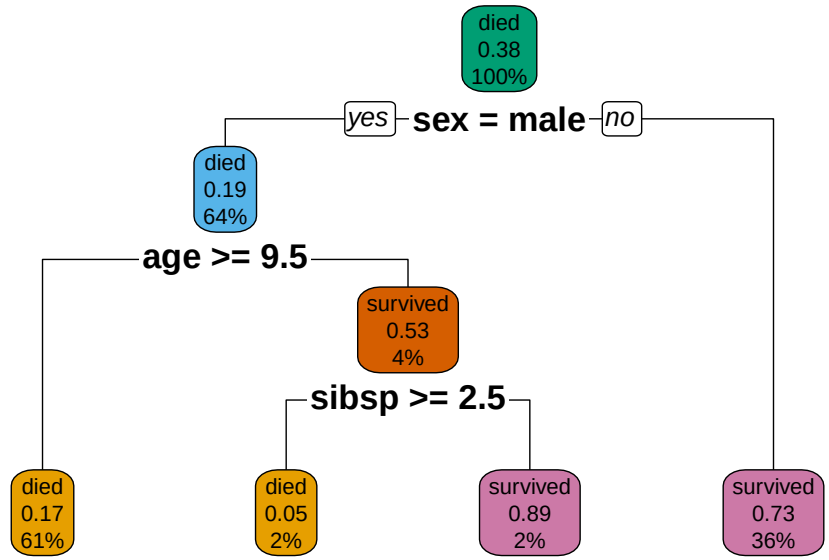
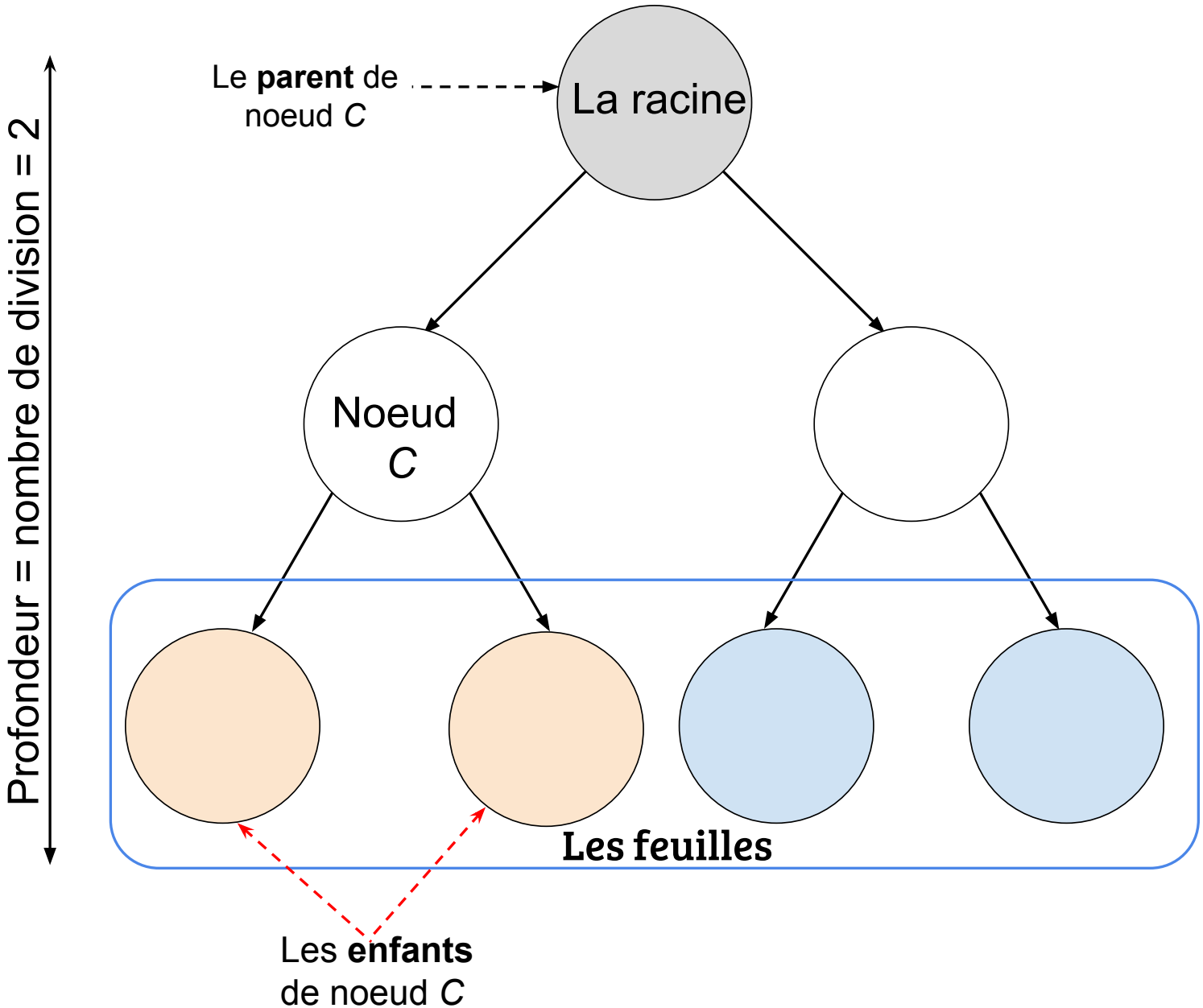


Fig.: Classification

- Aujourd'hui → régression

Vocabulaire



Un exemple justificatif

Ligue majeure de baseball

- Jeux de données de la ligue majeure de baseball en 1986 et 1987, disponible dans le paquet **ISLR** [2, 3] en **R** :

```
library(ISLR)
data(Hitters)
```

Ligue majeure de baseball

- Jeux de données de la ligue majeure de baseball en 1986 et 1987, disponible dans le paquet **ISLR** [2, 3] en **R** :

```
library(ISLR)
data(Hitters)
```

- Variable réponse $y_i, i = 1, \ldots, 263$: salaire annuel (en milliers de dollars) au début de la saison 1987

Ligue majeure de baseball

- Jeux de données de la ligue majeure de baseball en 1986 et 1987, disponible dans le paquet **ISLR** [2, 3] en **R** :

```
library(ISLR)
data(Hitters)
```

- Variable réponse $y_i, i = 1, \ldots, 263$: salaire annuel (en milliers de dollars) au début de la saison 1987

- Variables explicatives:
  1. X_1: nombre d'années dans la ligue majeure

Ligue majeure de baseball

- Jeux de données de la ligue majeure de baseball en 1986 et 1987, disponible dans le paquet **ISLR** [2, 3] en **R** :

```
library(ISLR)
data(Hitters)
```

- Variable réponse $y_i, i = 1, \ldots, 263$: salaire annuel (en milliers de dollars) au début de la saison 1987

- Variables explicatives:
  1. X_1: nombre d'années dans la ligue majeure
  2. X_2: nombre de frappes en 1986

Ligue majeure de baseball

- Jeux de données de la ligue majeure de baseball en 1986 et 1987, disponible dans le paquet **ISLR** [2, 3] en **R** :

```
library(ISLR)
data(Hitters)
```

- Variable réponse $y_i, i = 1, \ldots, 263$: salaire annuel (en milliers de dollars) au début de la saison 1987

- Variables explicatives:
  1. X_1: nombre d'années dans la ligue majeure
  2. X_2: nombre de frappes en 1986

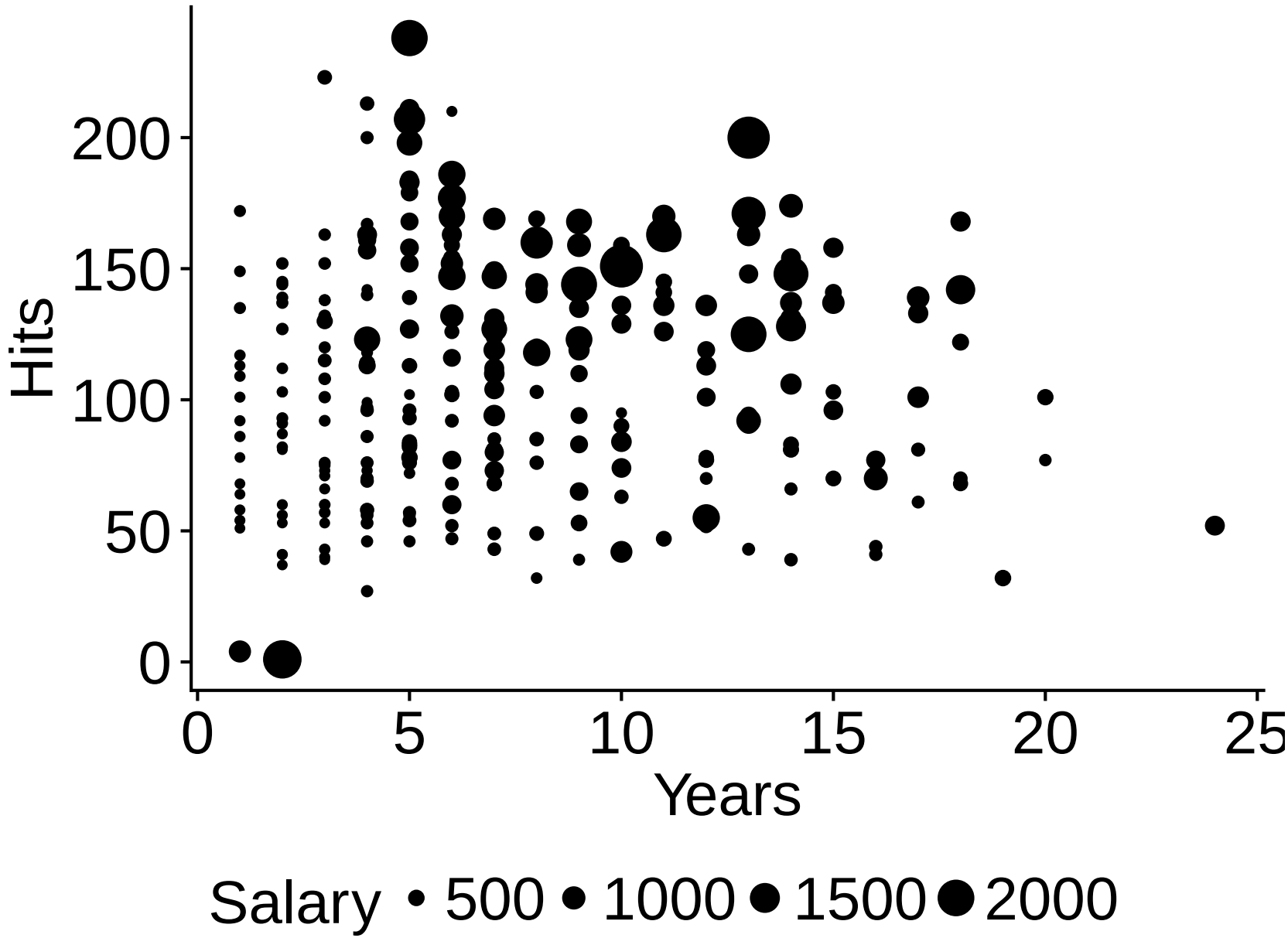
Objectif

Prévoir le salaire annuel (**salary**) au début de la saison 1987 avec les variables explicatives (**years** et **hits**).

Les données

	Years	Hits	Salary
-Andre Dawson	11	141	500
-Andres Galarraga	2	87	92
-Barry Bonds	1	92	100
-Cal Ripken	6	177	1350
-Gary Carter	13	125	1926
-Joe Carter	4	200	250
-Ken Griffey	14	150	1000
-Mike Schmidt	2	1	2127
-Tony Gwynn	5	211	740

Les données


Hits
200
150
100
50
0
0
5
10
15
20
25
Years
Salary • 500 • 1000 • 1500 • 2000

Comment CART fonctionne-t-il?

Il y a deux étapes:

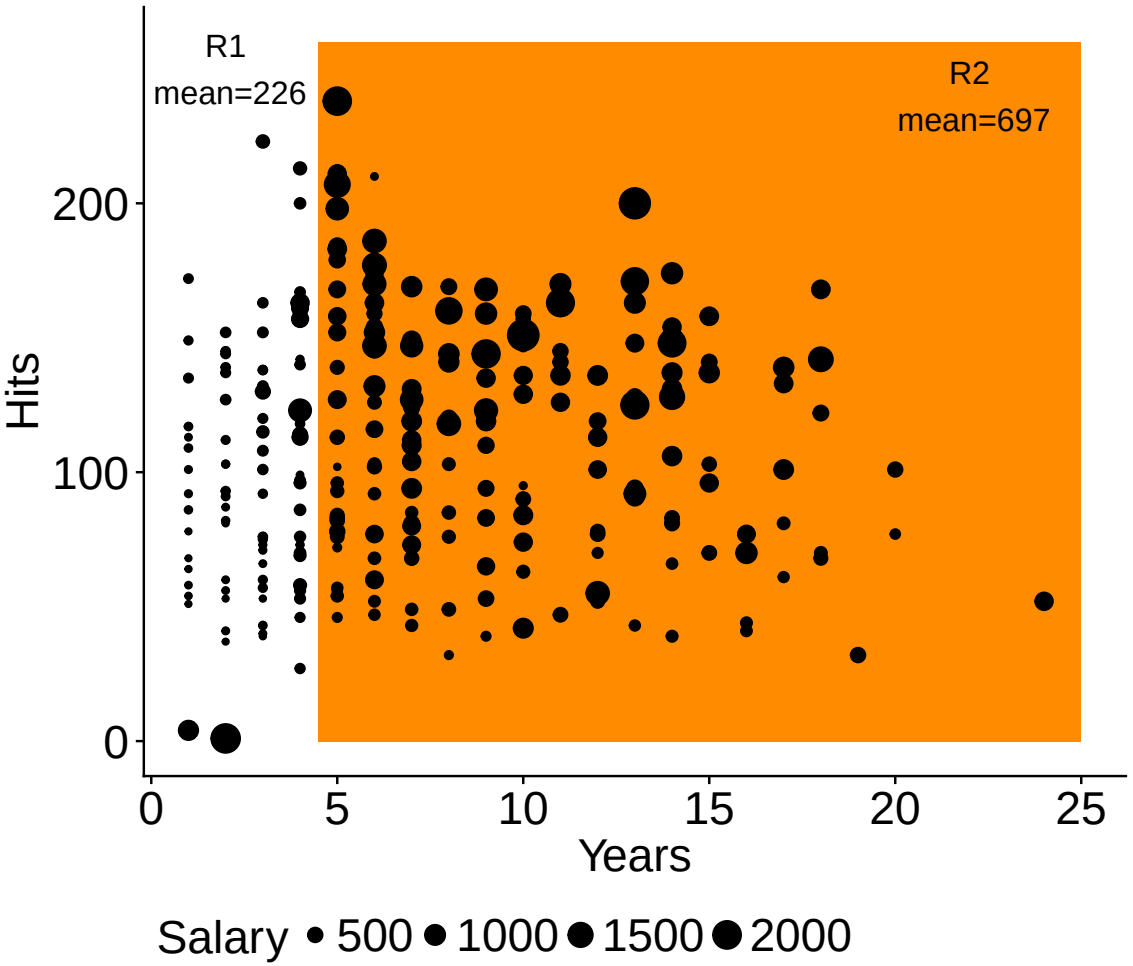
Comment CART fonctionne-t-il?

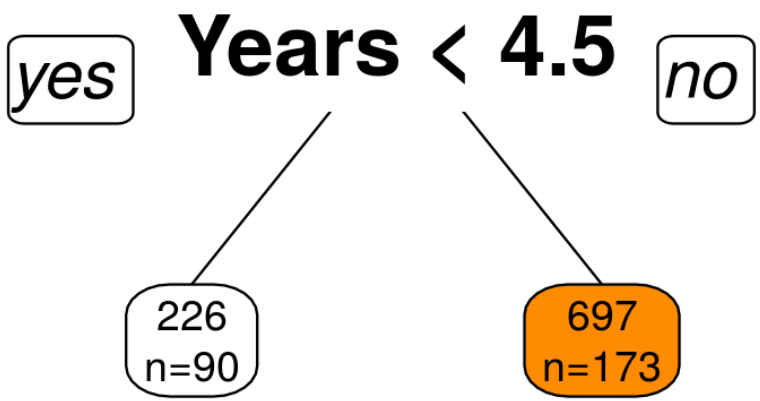
Il y a deux étapes:

1. Nous divisons l'espace prédicteur, c'est-à-dire l'ensemble des valeurs possibles $X_1, X_2, \ldots, X_p$ - en J régions exhaustives et non chevauchantes, $R_1, R_2, \ldots, R_J$.

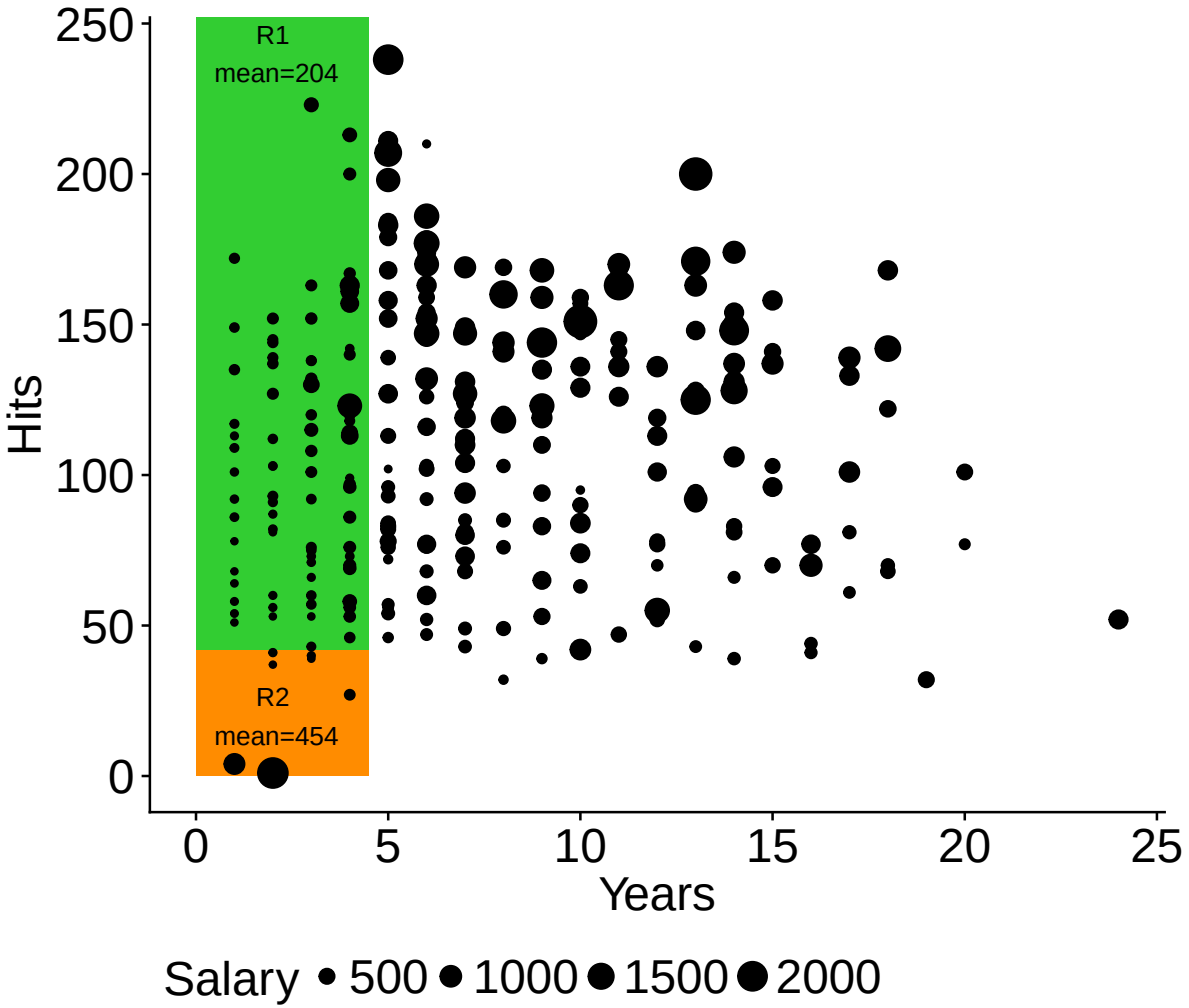
2. Pour chaque observation qui tombe dans la région R_j, nous faisons la même prévision, qui est simplement la moyenne des valeurs de réponse à R_j.

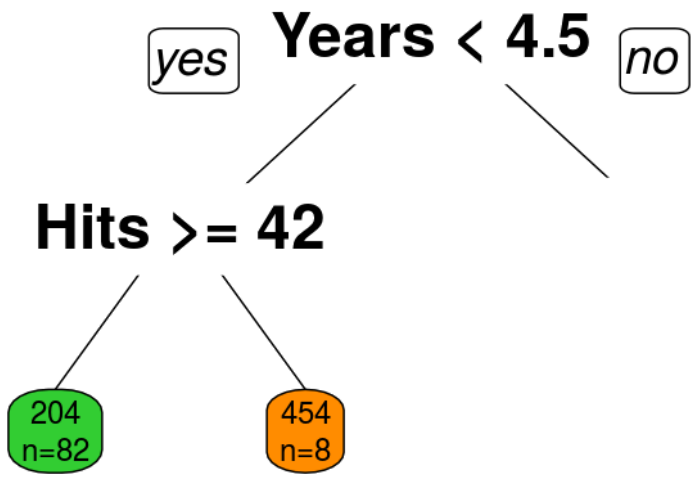
Première division





Deuxième division



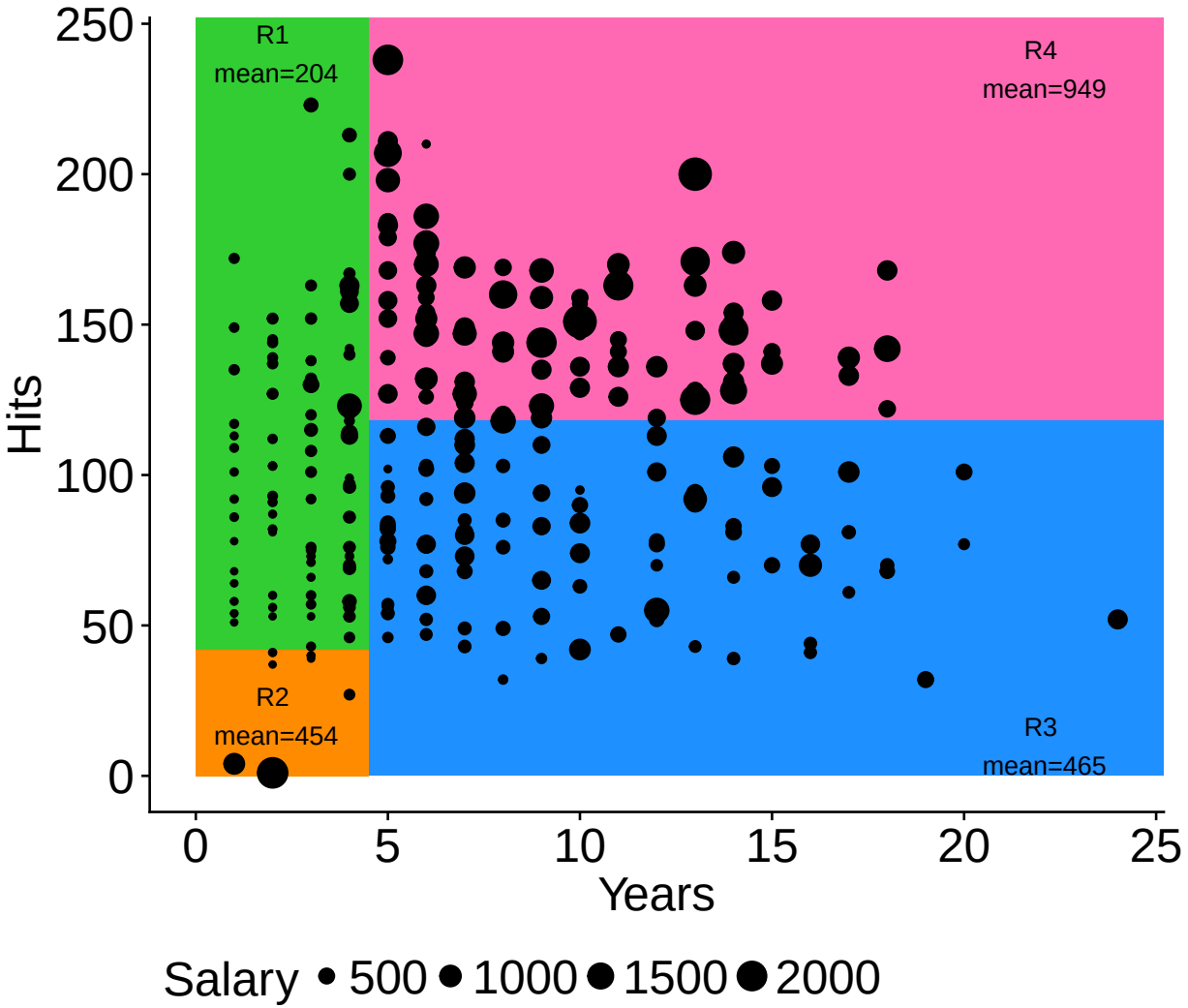


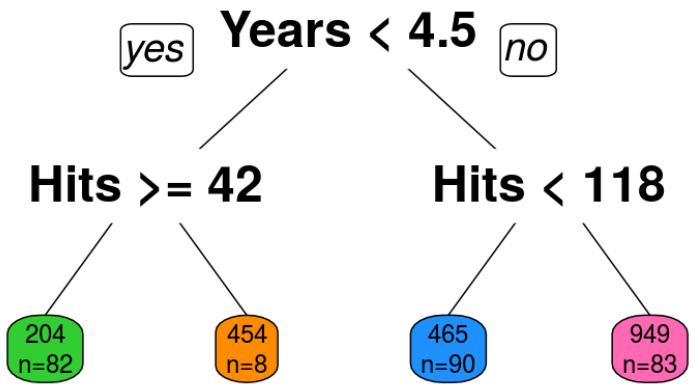
Une erreur dans les données

	Years	Hits	Salary
-Andre Dawson	11	141	500.00
-Andres Galarraga	2	87	91.50
-Barry Bonds	1	92	100.00
-Cal Ripken	6	177	1350.00
-Gary Carter	13	125	1925.57
-Joe Carter	4	200	250.00
-Ken Griffey	14	150	1000.00
-Mike Schmidt	2	1	2127.33
-Tony Gwynn	5	211	740.00

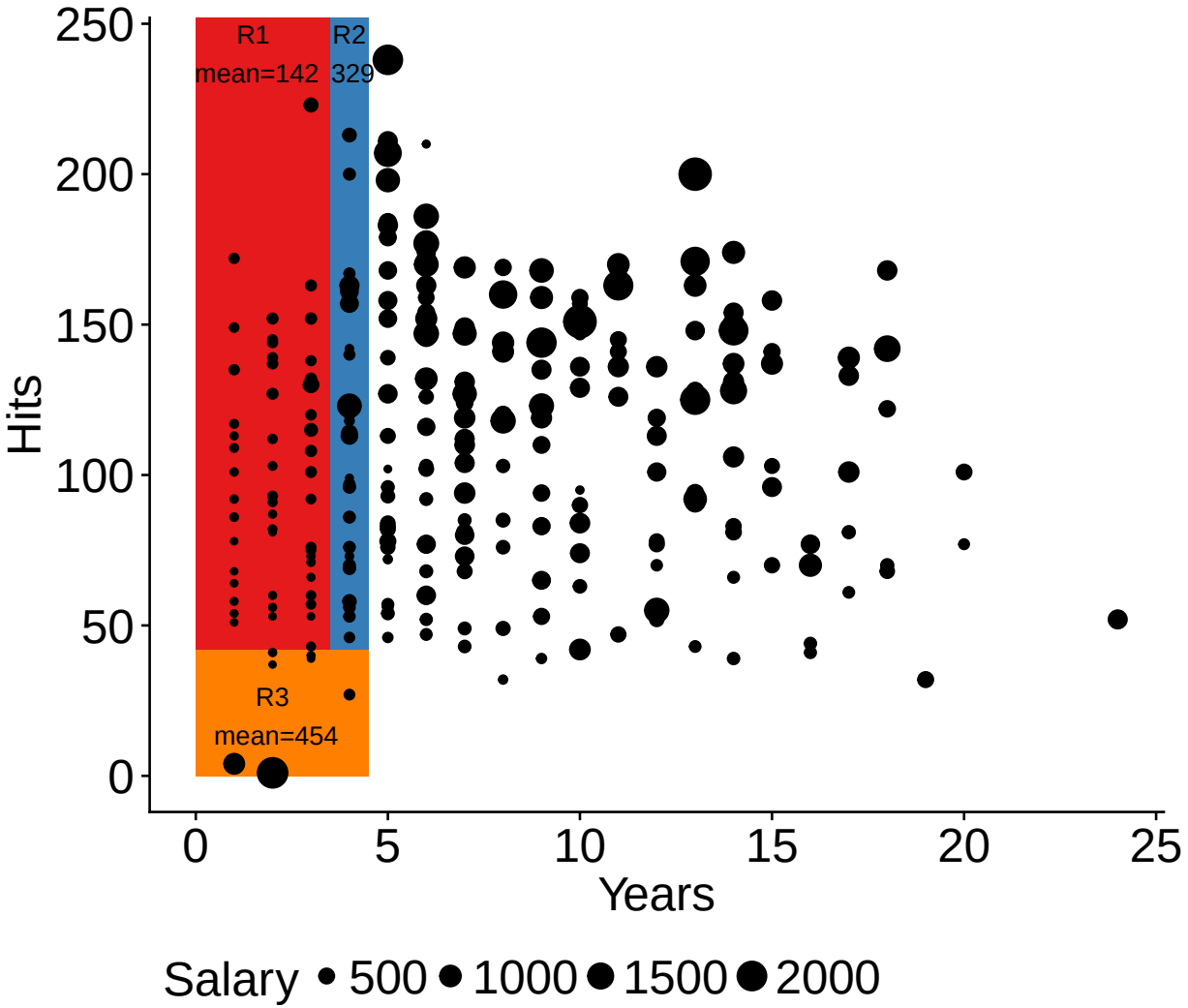
- Mike Schmidt a commencé sa carrière en 1972, et a été introniser au Temple de la renommée du baseball en 1995.

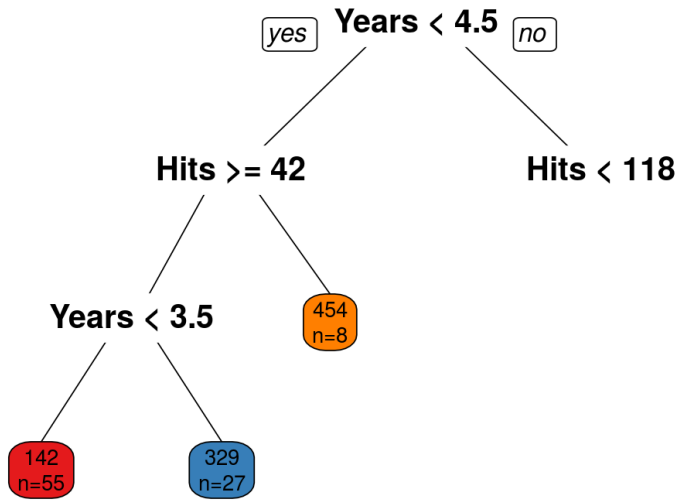
Deuxième division



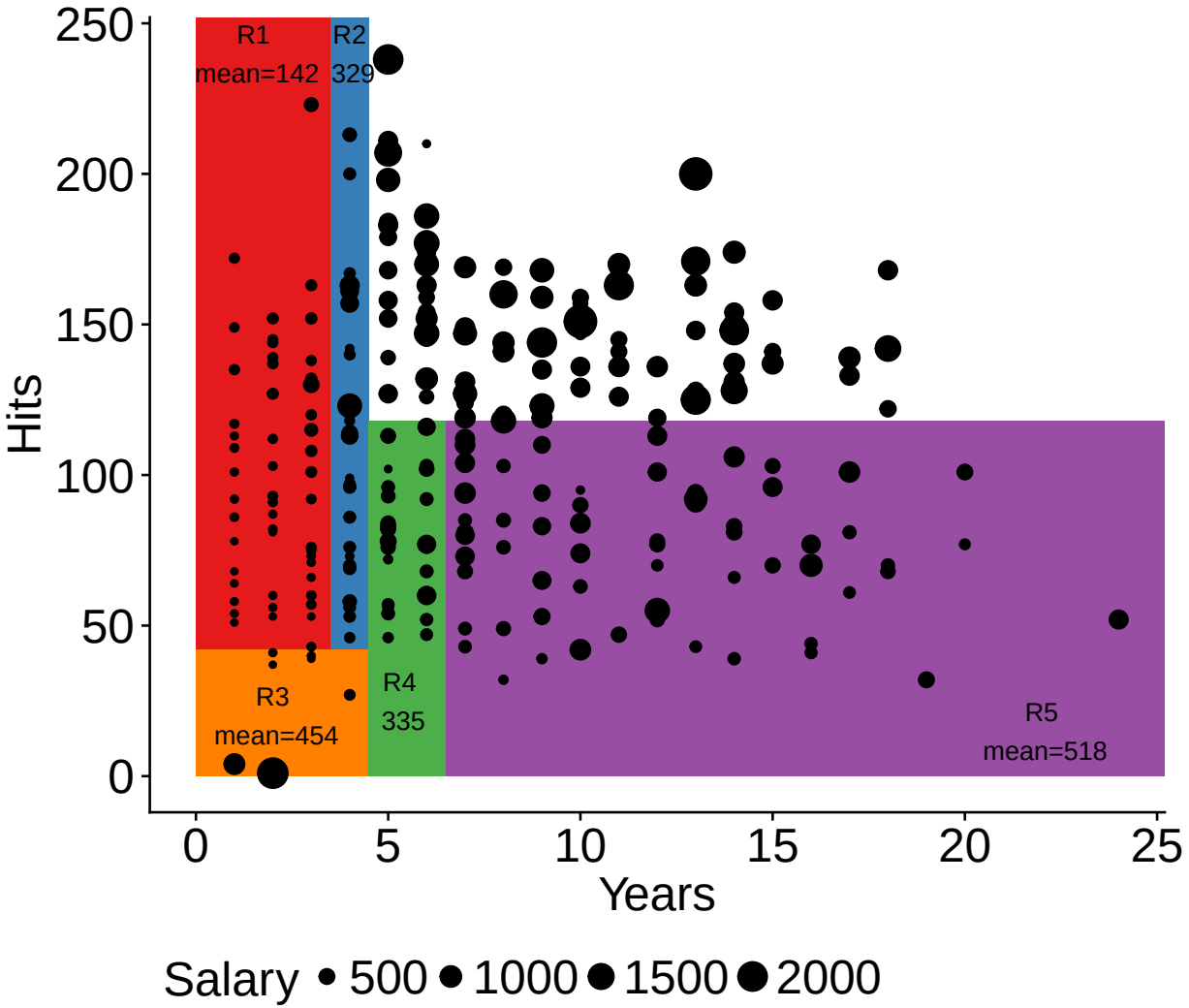


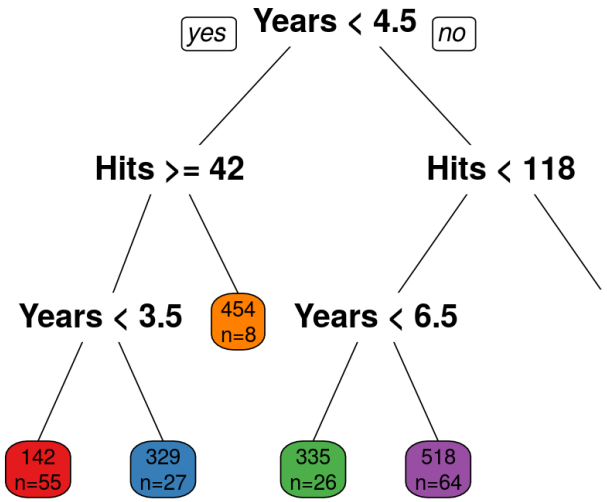
Troisième division



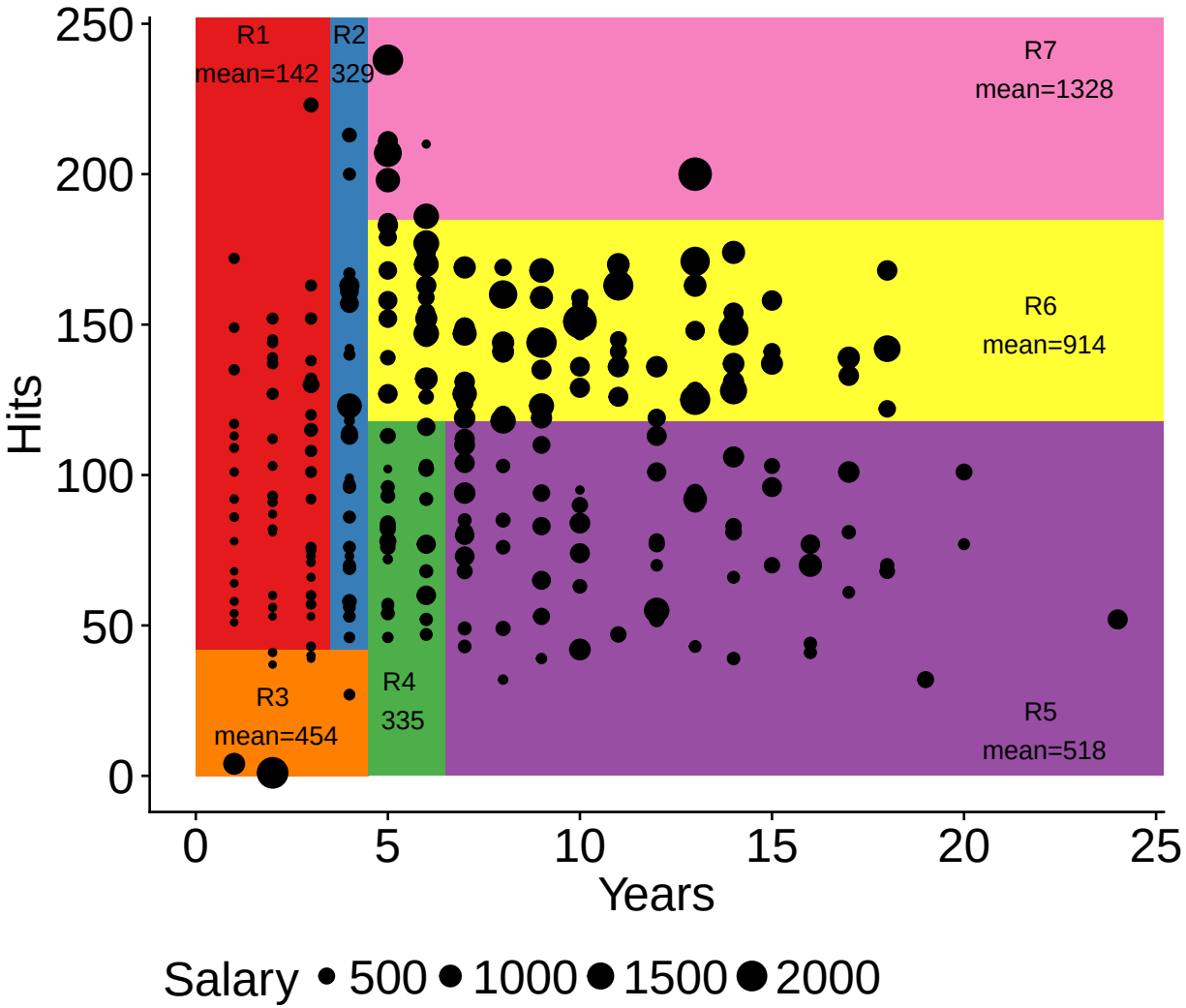


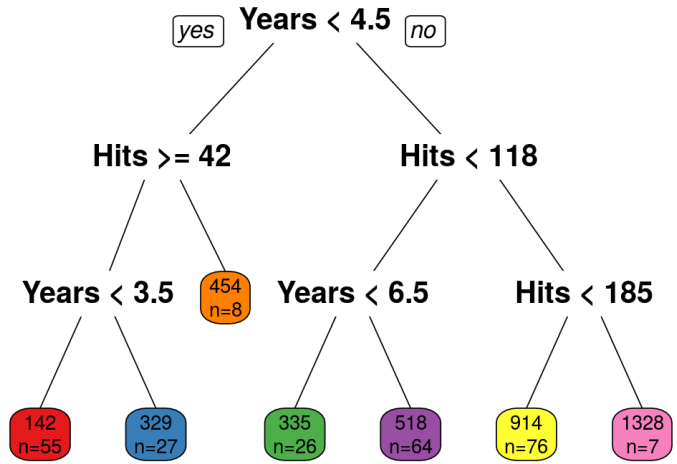
Troisième division



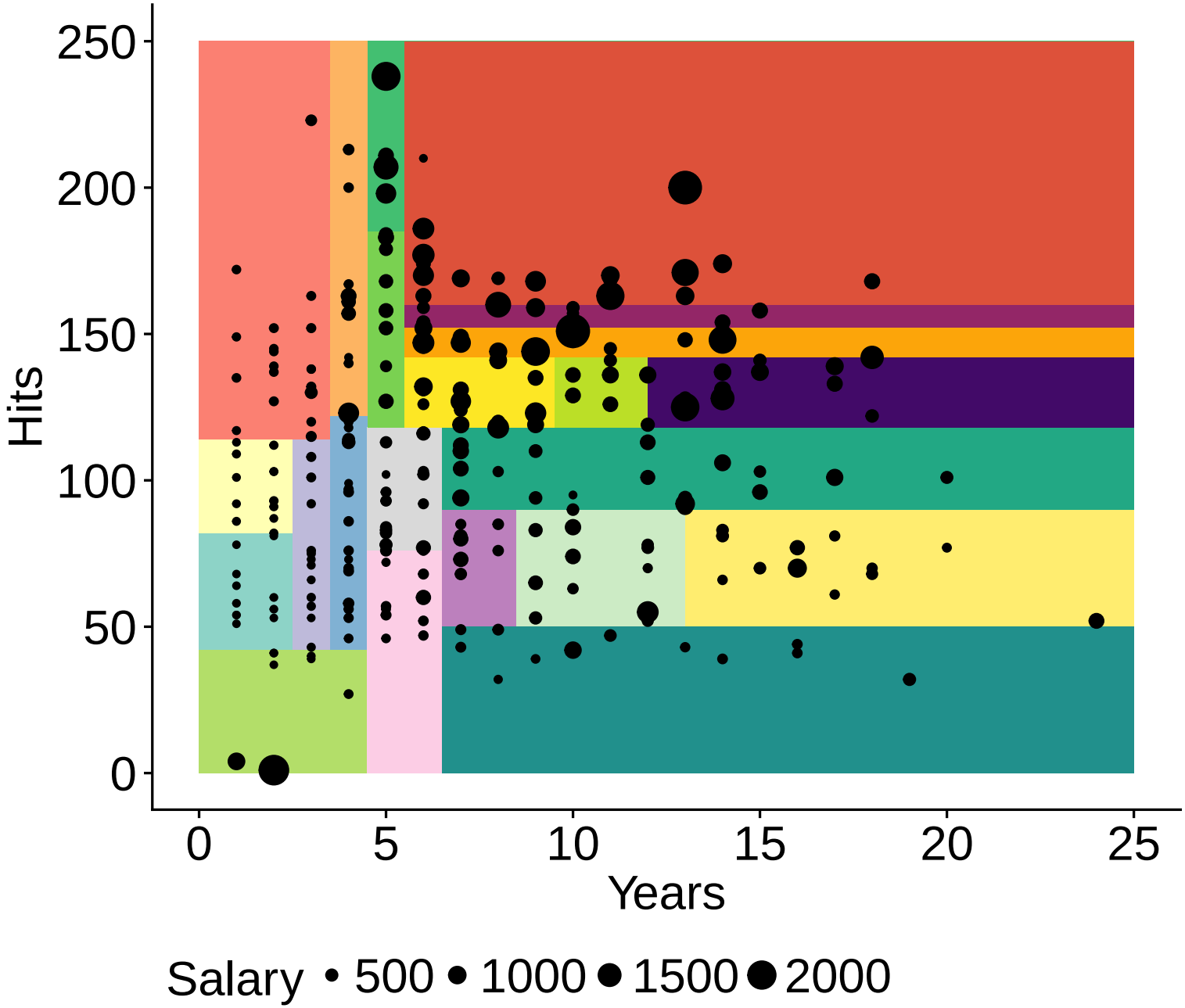


Troisième division

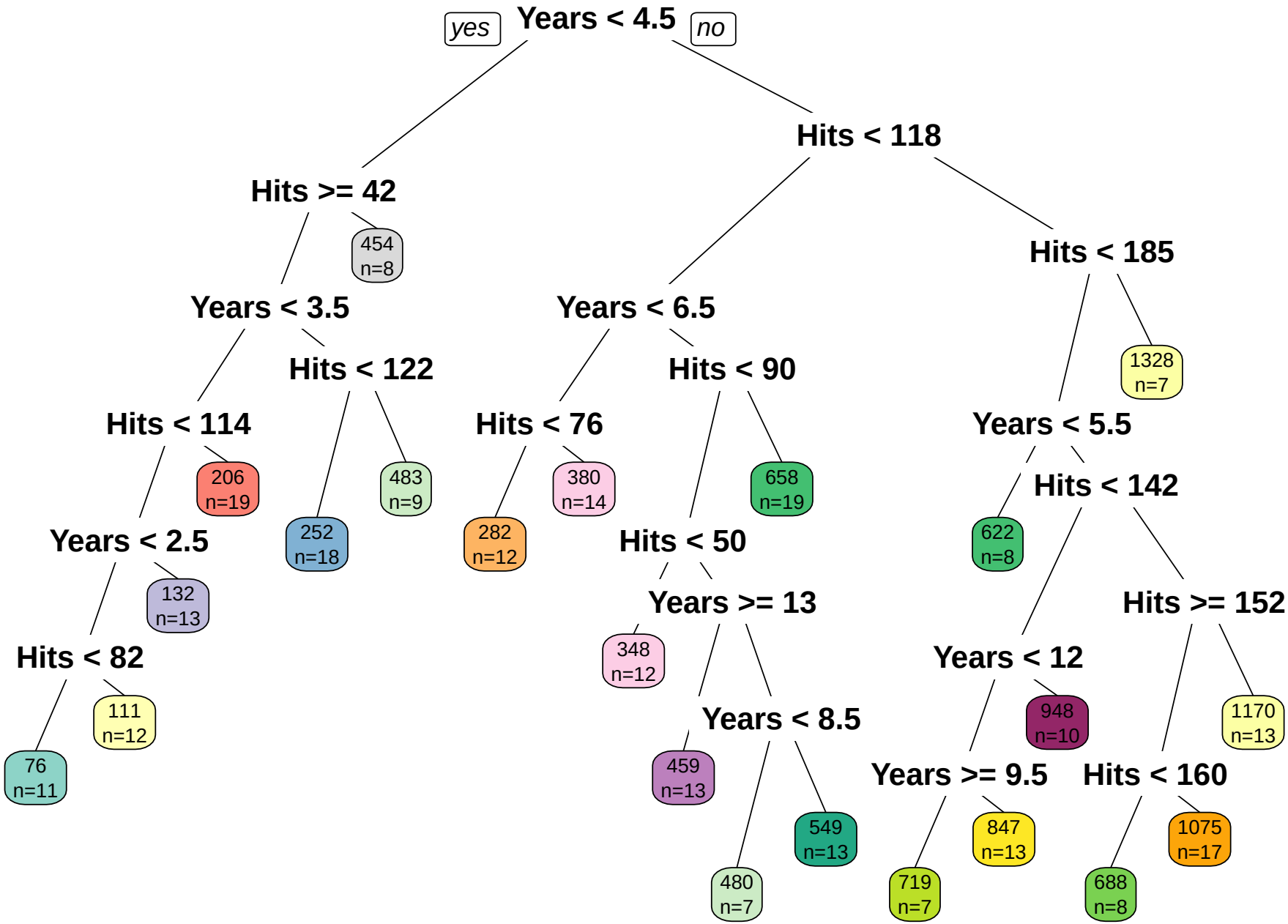




Et si on continue...



Arrêter si le nombre d'observations est inférieur à 20



Les détails

Les détails

L'algorithme considéré nécessite:

1. La définition d'un critère permettant de sélectionner la meilleure division parmi toutes celles admissibles pour les différentes variables.

2. Une règle permettant de décider qu'un noeud est terminal: il devient ainsi une feuille.

3. Élagage (*pruning*) de l'arbre optimal pour éviter le sur-ajustement.

1. Sélectionner la meilleure division

L'objectif est de trouver les divisions $R_1, \ldots, R_J$ qui minimisent la fonction de perte:

$$\sum_{j=1}^{J} \sum_{i \in R_j} (y_i - \hat{y}_{R_j})^2 \tag{1}$$

- $\hat{y}_{R_j}$: la moyenne de la variable réponse dans la région R_j

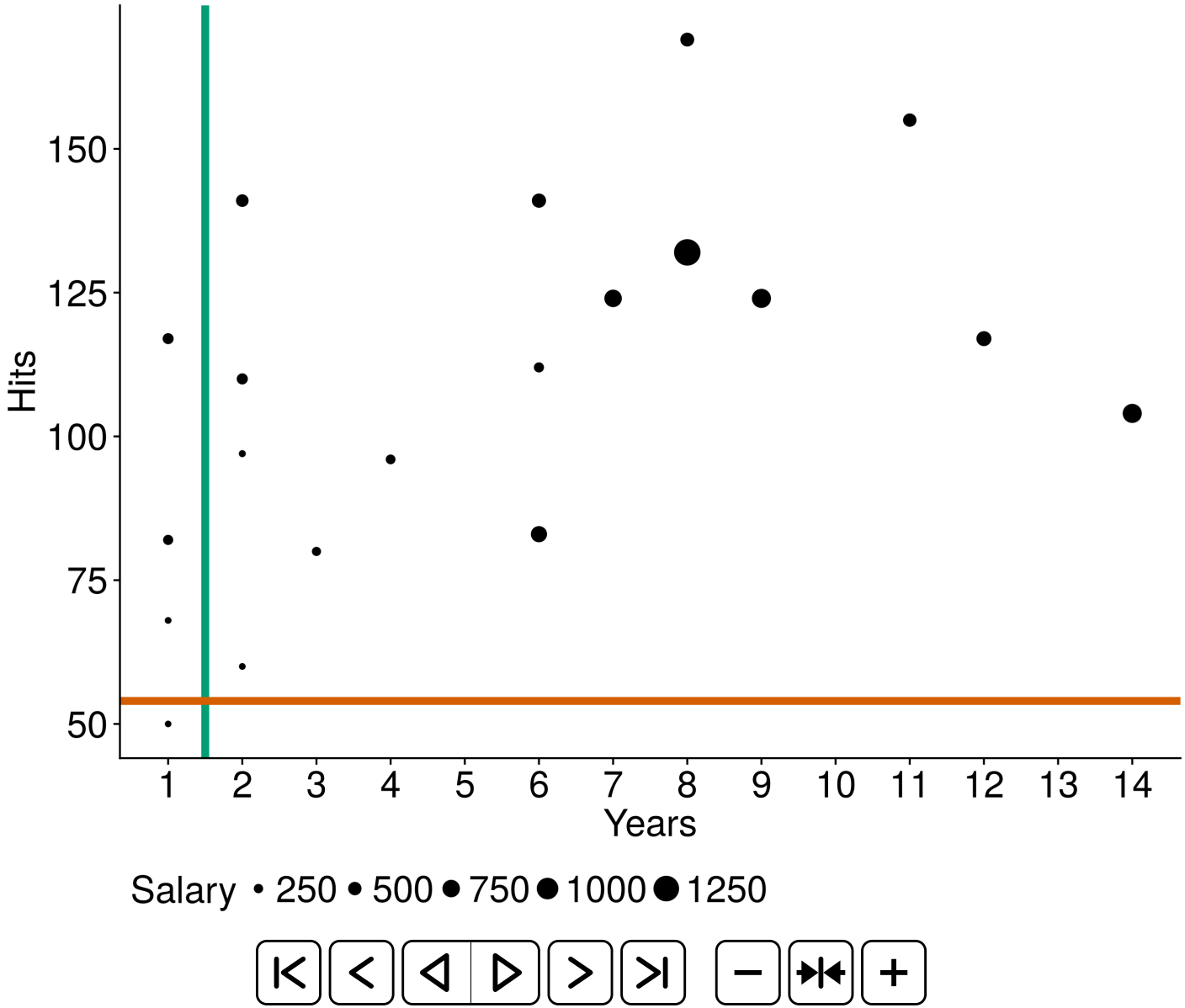
1. Sélectionner la meilleure division

L'objectif est de trouver les divisions $R_1, \ldots, R_J$ qui minimisent la fonction de perte:

$$\sum_{j=1}^{J} \sum_{i \in R_j} (y_i - \hat{y}_{R_j})^2 \tag{1}$$

- $\hat{y}_{R_j}$: la moyenne de la variable réponse dans la région R_j
- Trouver la solution de (1) est quasiment impossible (*NP-hard*). Pourquoi?

Recherche exhaustive pour $J = 4$



1. Sélectionner la meilleure division de façon «*greedy*»

- Au début, toutes les observations appartiennent à la même région.
- La division du noeud crée deux enfants, gauche et droit.
- On cherche pour chaque noeud la division, ou plus précisément la variable (X_j) et la règle de division (s), qui contribuera à la plus forte décroissance de l'hétérogénéité des noeuds enfants à gauche (R_1) et à droite (R_2)

$$R_1(j,s) = \{X|X_j < s\} \qquad \text{et} \qquad R_2(j,s) = \{X|X_j \geq s\}$$

1. Sélectionner la meilleure division de façon «*greedy*»

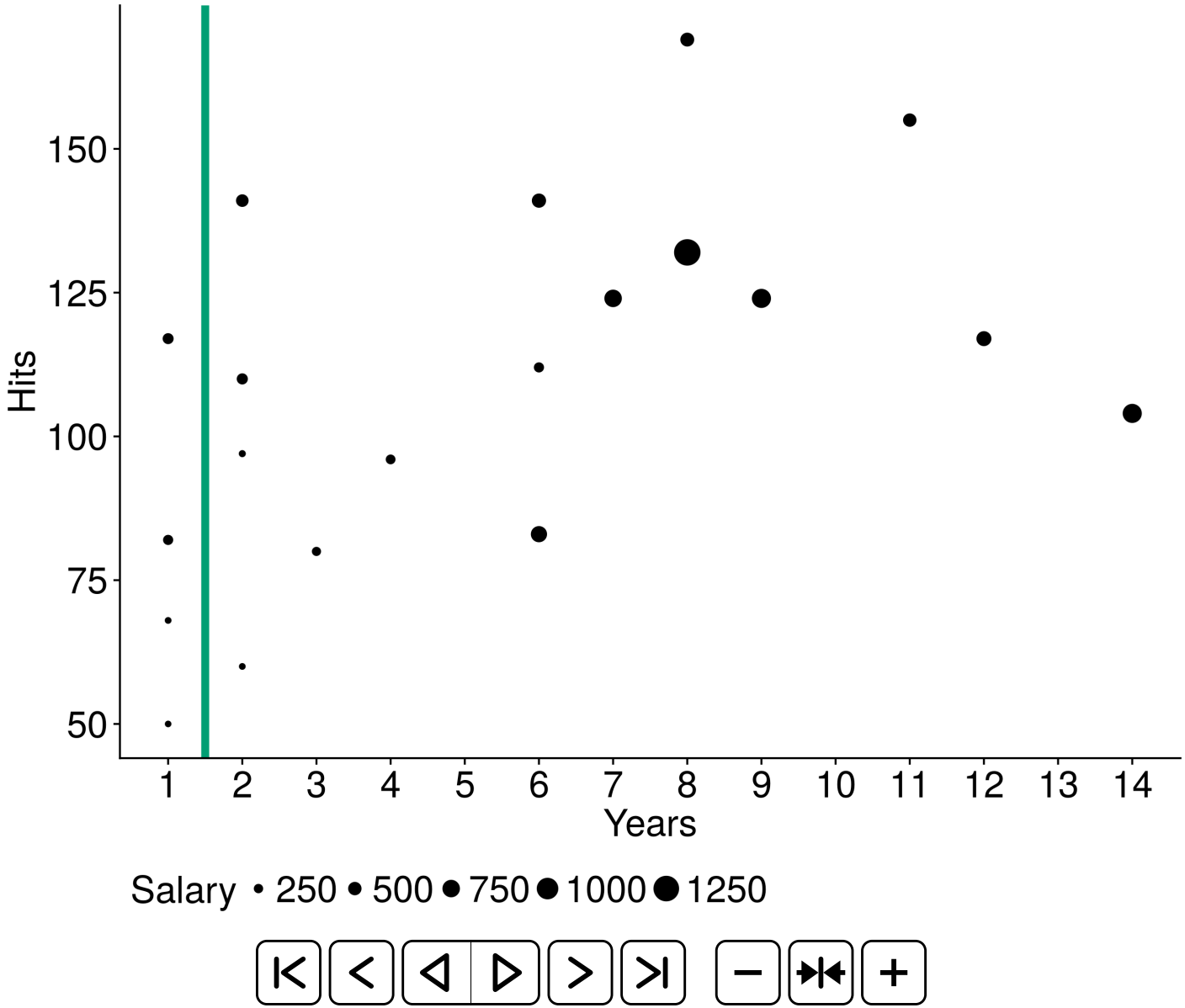
- Au début, toutes les observations appartiennent à la même région.
- La division du noeud crée deux enfants, gauche et droit.
- On cherche pour chaque noeud la division, ou plus précisément la variable (X_j) et la règle de division (s), qui contribuera à la plus forte décroissance de l'hétérogénéité des noeuds enfants à gauche (R_1) et à droite (R_2)

$$R_1(j,s) = \{X|X_j < s\} \qquad \text{et} \qquad R_2(j,s) = \{X|X_j \geq s\}$$

L'objectif est de trouver les valeurs de j et s qui minimisent la fonction de perte:

$$\sum_{i:x_i \in R_1(j,s)} (y_i - \hat{y}_{R_1})^2 + \sum_{i:x_i \in R_2(j,s)} (y_i - \hat{y}_{R_2})^2 \tag{2}$$

La meilleure division de façon «*greedy*»



2. Règle d'arrêt

- Pour éviter un découpage inutilement fin, le nombre d'observations minimal qui doivent exister dans un noeud pour qu'une tentative de division soit effectuée (`minsplit = 20` par défaut en `rpart`).

- La croissance de l'arbre s'arrête à un noeud donné, qui devient donc terminal ou feuille, lorsqu'il contient le nombre d'observations minimum. (`minbucket` = `minsplit/3` par défaut en `rpart`)

3. Élagage de l'arbre optimal

- La démarche de construction précédente fournit l'arbre maximal T_{max} avec $|T_{max}|$ feuilles.

- T_{max} peut être excessivement raffiné et donc conduire à un modèle de prévision très **instable** car fortement dépendant des échantillons qui ont permis son estimation.

- C'est une situation de **sur-ajustement** à éviter au profit de modèles plus parcimonieux donc plus robustes au moment de la prévision.

- Cet objectif est obtenu par une procédure d'élagage (*pruning*) de l'arbre.

3. Élagage de l'arbre optimal

- La construction de la séquence d'arbres emboîtés repose sur une pénalisation de la complexité de l'arbre.

- Pour chaque valuer de α, il existe un arbre $T \subset T_{max}$ qui minimise:

$$\sum_{m=1}^{|T|} \sum_{i:x_i \in R_m} (y_i - \hat{y}_{R_m})^2 + \alpha |T| \tag{3}$$

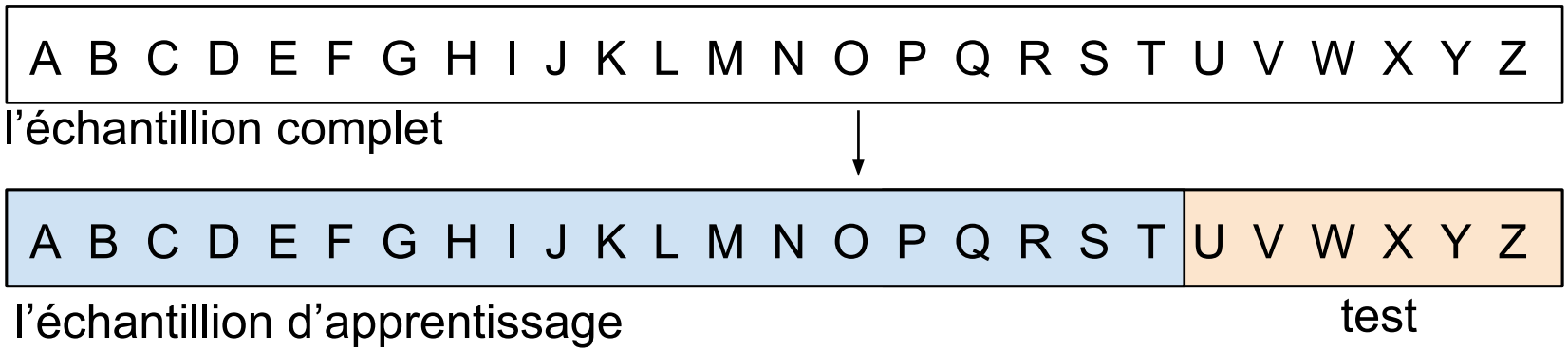
- α est choisi selon la validation croisée par v-ensembles ($v \rightarrow$ `xval=10` en `rpart` par défaut)

Contexte sur la validation croisée

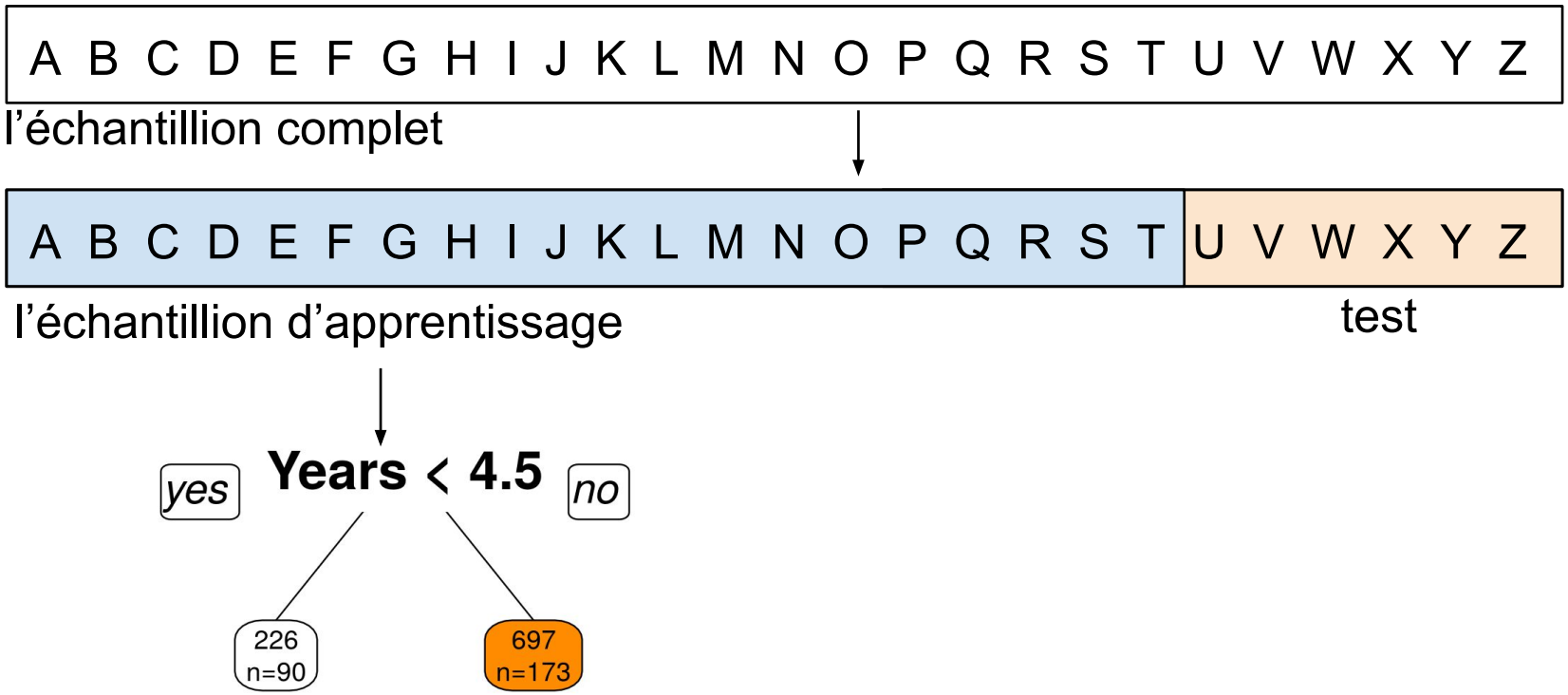
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

l'échantillion complet

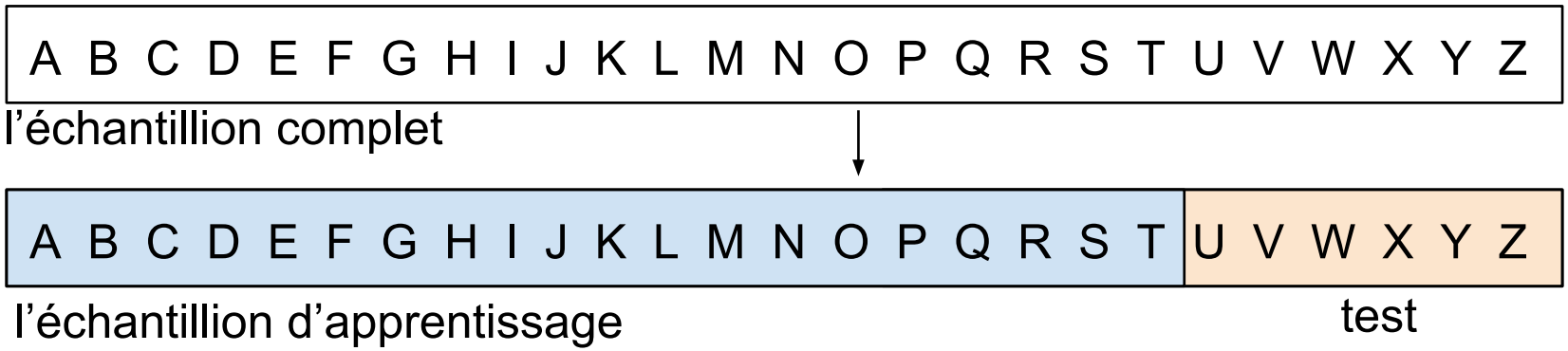
Contexte sur la validation croisée

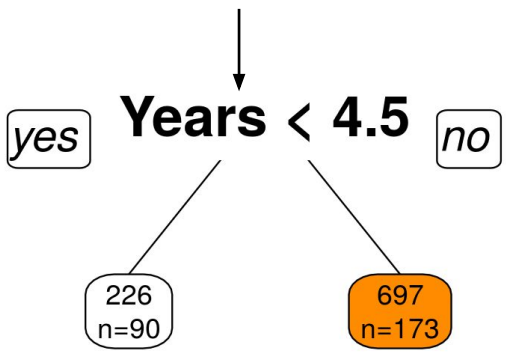


Contexte sur la validation croisée



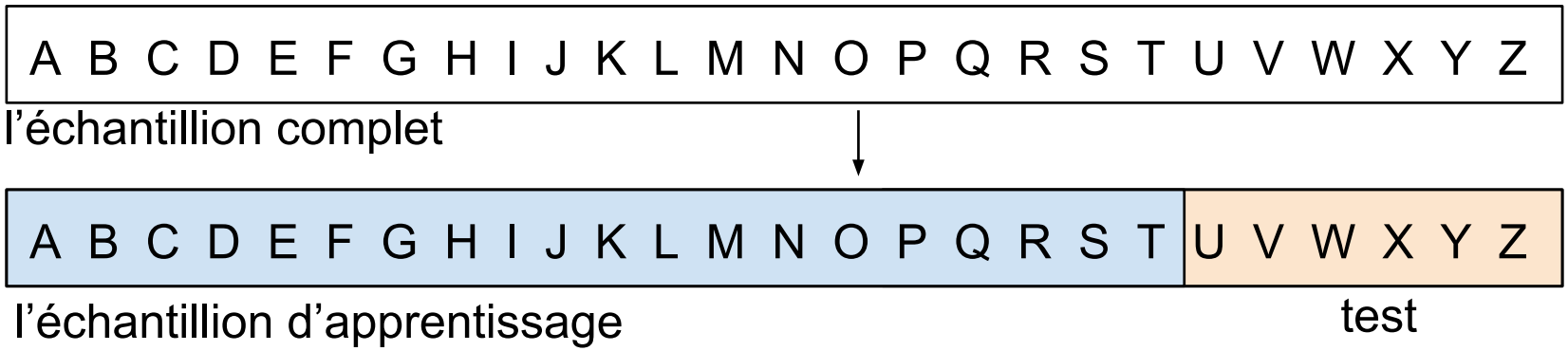
Contexte sur la validation croisée

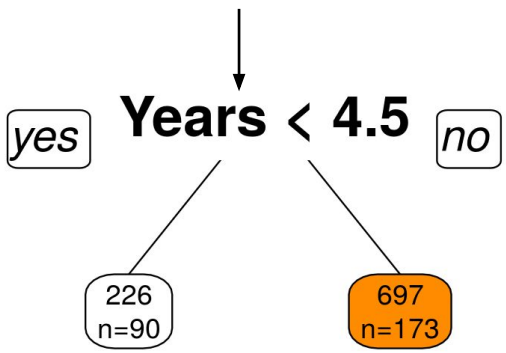




i	years	y_i	$y_i^{(pred)}$
U	5	373	
V	3	277	
W	15	1456	
X	4	455	
Y	1	235	
Z	9	987	

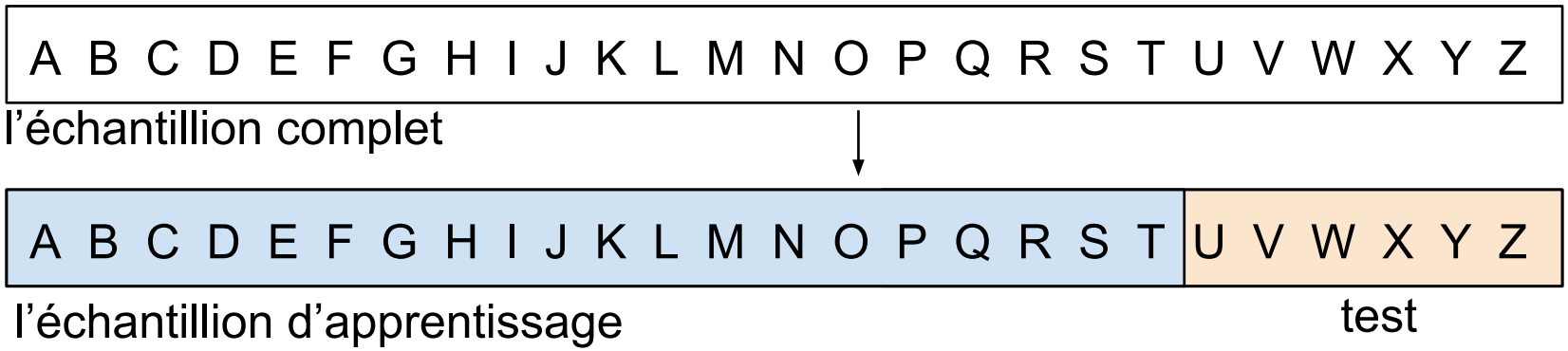
Contexte sur la validation croisée

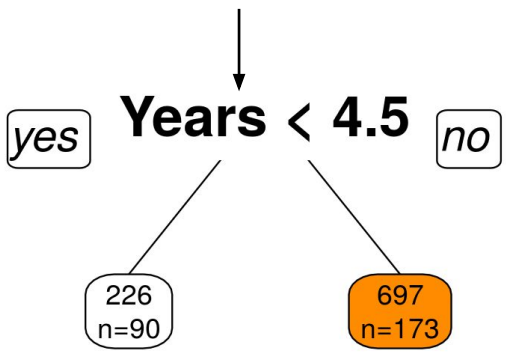




i	years	y_i	$y_i^{(pred)}$
U	5	373	697
V	3	277	226
W	15	1456	697
X	4	455	226
Y	1	235	226
Z	9	987	697

Contexte sur la validation croisée

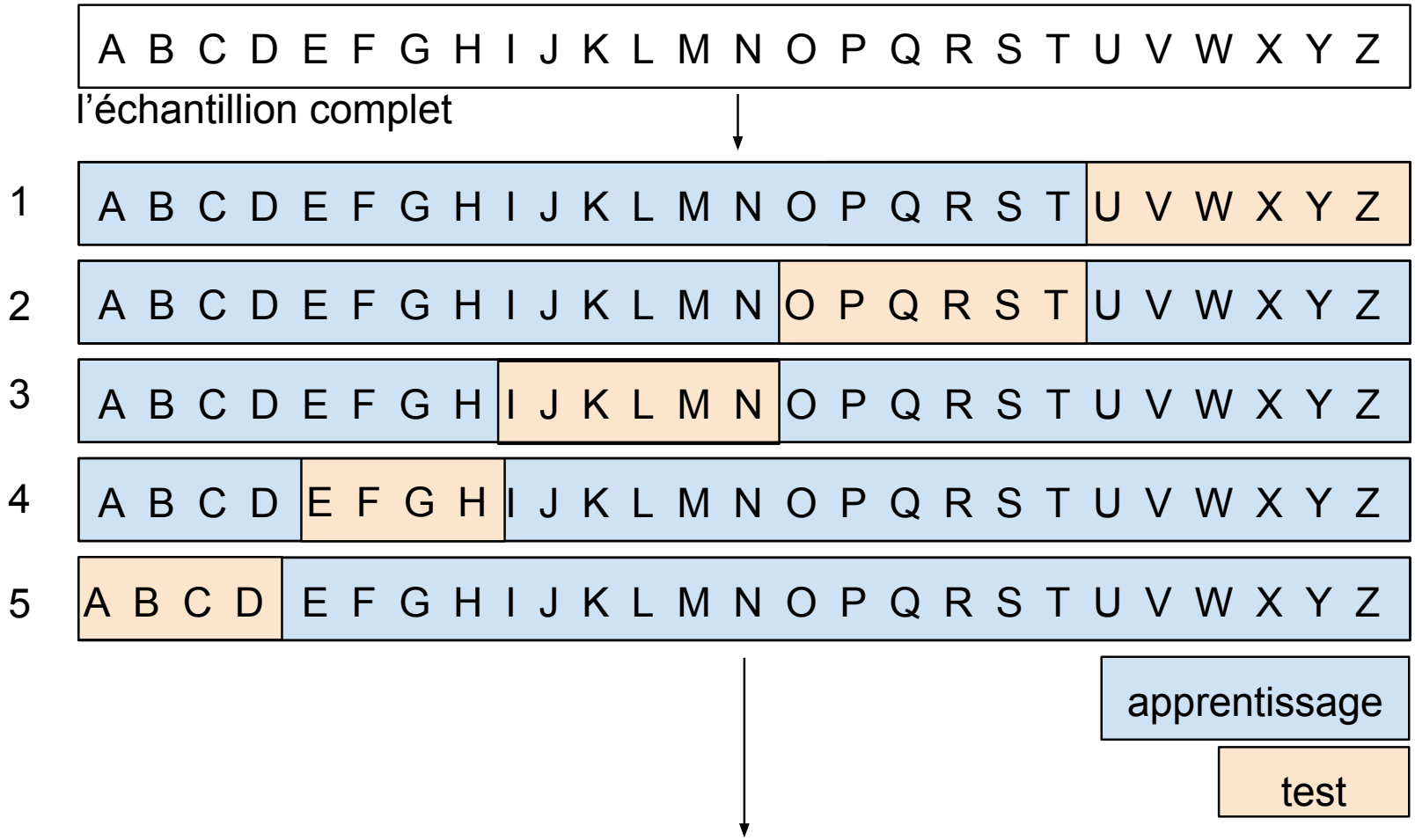




i	years	y_i	$y_i^{(pred)}$
U	5	373	697
V	3	277	226
W	15	1456	697
X	4	455	226
Y	1	235	226
Z	9	987	697

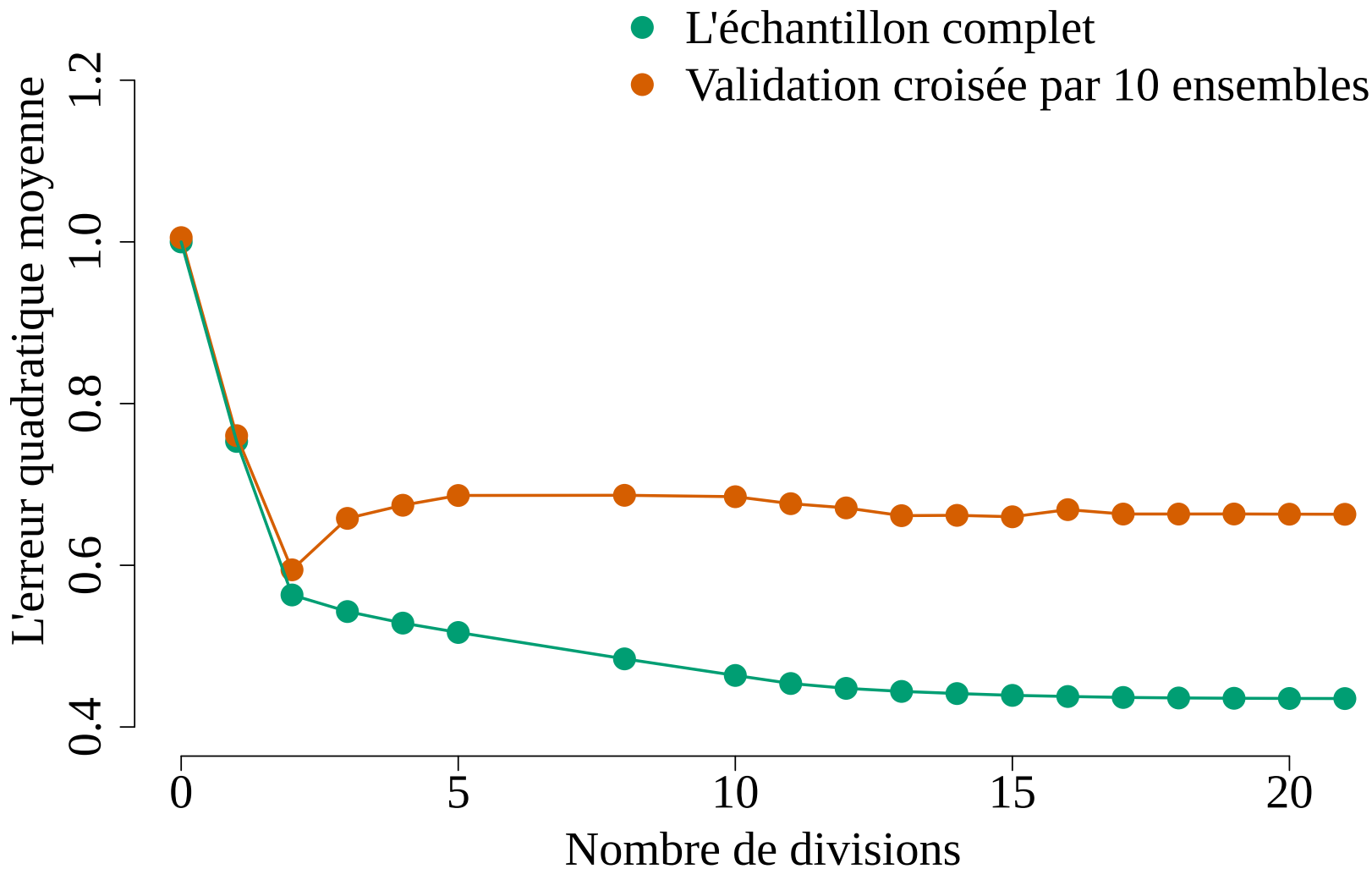
$$MSE^{(test)} = \sum_{i=1}^{6} \left(y_i - y_i^{(pred)} \right)^2 + \alpha |T|$$

Contexte sur la validation croisée



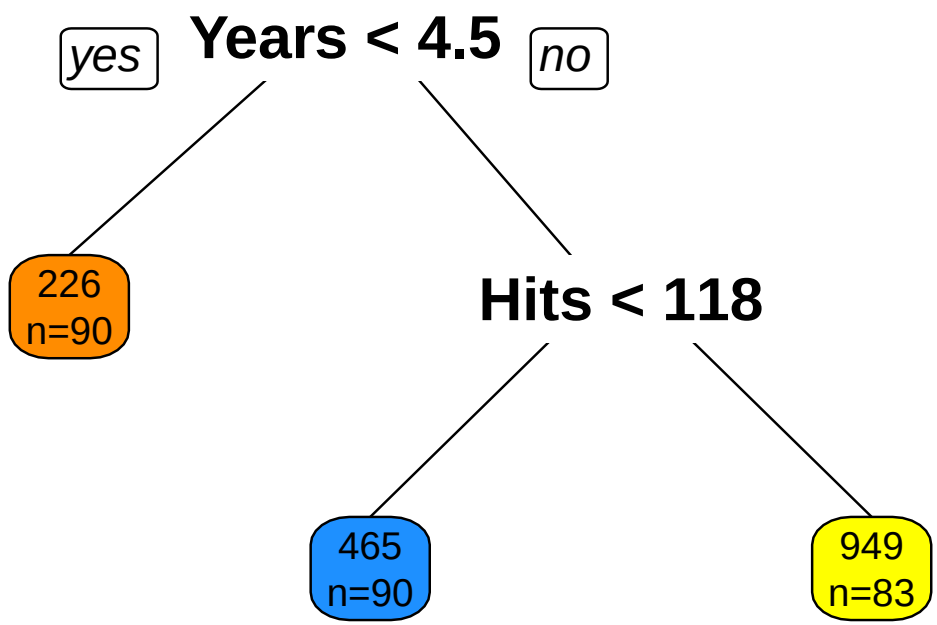
$$CV(\alpha) = \frac{1}{5}\sum_{v=1}^{5} MSE_v^{(test)}$$

Sur-ajustement



Sur-ajustement

```
cart_fit <- rpart::rpart(Salary ~ Years + Hits, data = Hitters)
min_ind <- which.min(cart_fit$cptable[, "xerror"])
min_cp <- cart_fit$cptable[min_ind, "CP"]
prune_fit <- rpart::prune(cart_fit, cp = min_cp)
rpart.plot::rpart.plot(prune_fit)
```



Comparaison avec un modèle linéaire

Comparaison: Modèle linéaire vs. CART

Caractéristique[a]	Modèle linéaire	CART
Hypothèse de linéarité	✓	✗
Hypothèse de distribution	✓	✗
Robuste à multicollinéarité	✗	✓
Données de grande dimension ($n << p$)	✗	✓
Traitement des interactions complexes	✗	✓
Données manquantes	✗	✓
Intervalle de confiance, valeur p	✓	✗

[a] ✓: oui, ✗: non

Modèle linéaire

```
lm(Salary ~ Years * Hits, data = Hitters)
```

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	159.55	95.65	1.67	0.10
Years	-16.08	11.38	-1.41	0.16
Hits	0.60	0.87	0.69	0.49
Years:Hits	0.54	0.11	5.08	0.00

Table: R2 = 0.41

Surface de régression

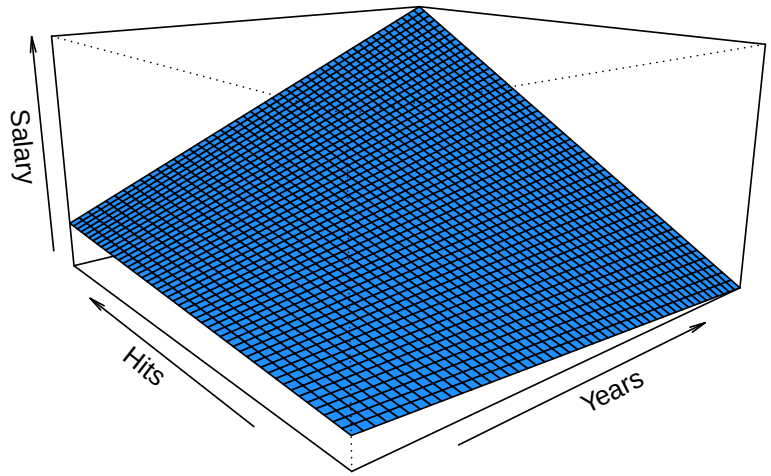


Fig.: Modèle linéaire

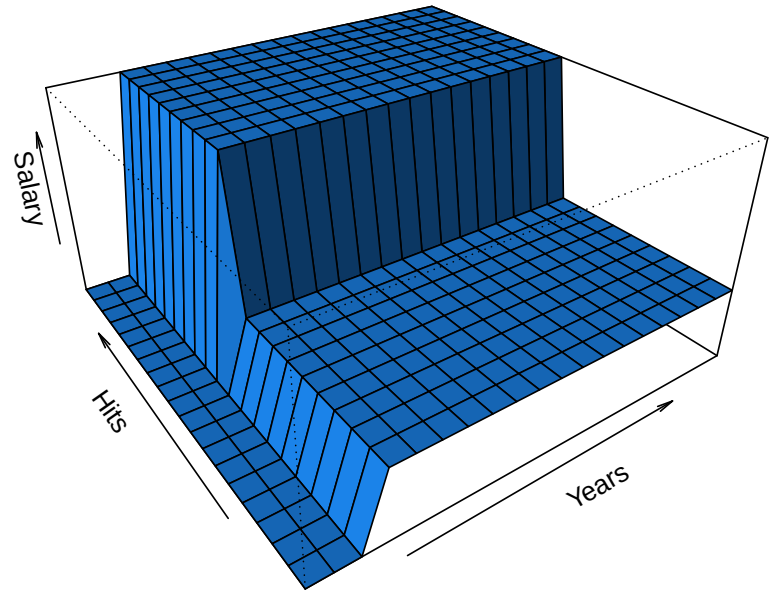
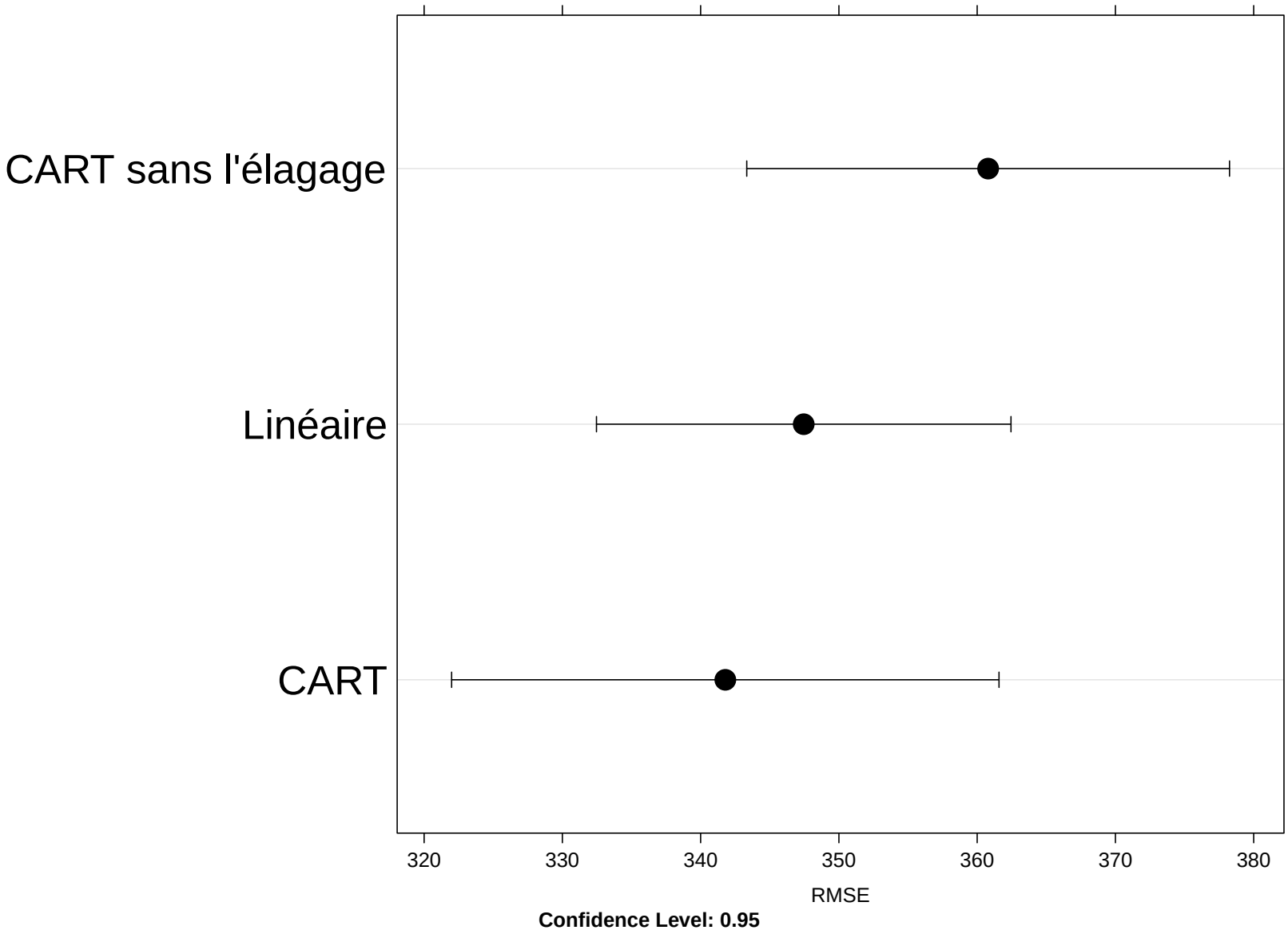
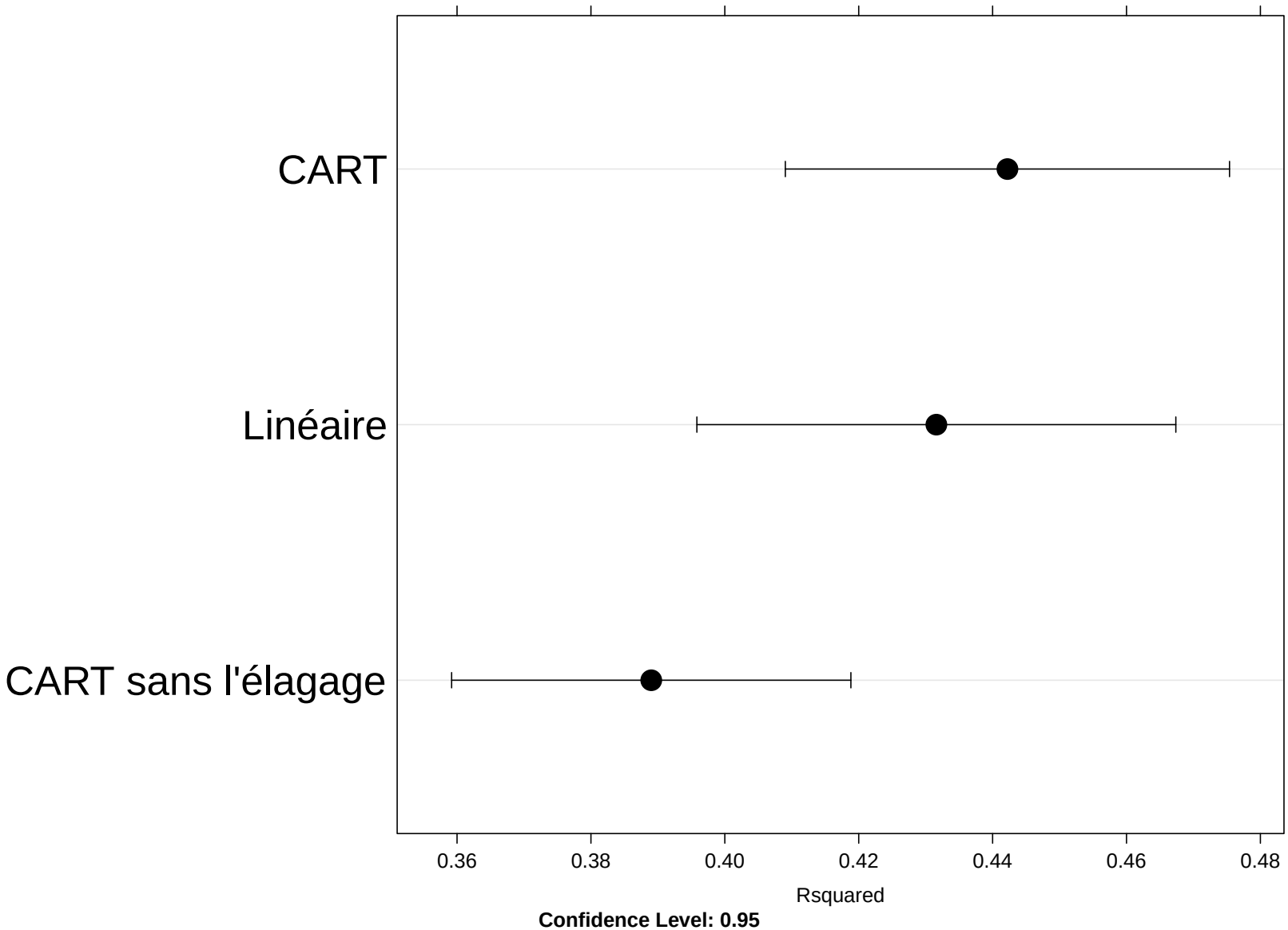


Fig.: CART

Moyenne RMSE: 10 fois validation croisée 10-ensembles



Moyenne R2: 10 fois validation croisée 10-ensembles



Avantages

- Elles produisent des résultats simples à interpréter.
- Permettent de ne pas introduire de structure a priori du lien de dépendance entre la variable à expliquer et covariables.
- Interactions d'ordre supérieur
- Données de grande dimension ($n << p$)

Faiblesses

- Ces modèles sont particulièrement instables, très sensibles à des fluctuations de l'échantillon (prochain cours $\rightarrow$ forêts aléatoires)

Exercise

Construire un arbre à la main = pas de logiciel!

- À partir des données suivantes, construisez un arbre de régression (sans l'élagage)

- Utilisez les paramètres `minsplit = 6` et `minbucket = 2`

	Years	Hits	Salary
-Rey Quinones	1	68	70
-Barry Bonds	1	92	100
-Pete Incaviglia	1	135	172
-Dan Gladden	4	97	210
-Juan Samuel	4	157	640
-Joe Carter	4	200	250
-Tim Wallach	7	112	750
-Rafael Ramirez	7	119	875
-Harold Baines	7	169	950

Références I

[1] Leo Breiman et al. *Classification and regression trees*. CRC press, 1984.

[2] Gareth James et al. *An introduction to statistical learning*. Vol. 112. Springer, 2013.

[3] Gareth James et al. “Package ‘ISLR’”. In: (2017).