DECISION TREES

Santiago Alférez
APRENDIZAJE AUTOMÁTICO DE MÁQUINA
MACC – UR – EICT

- Introducción – ejemplo
- Árboles de regresión
- Árboles de clasificación
- Ventajas y desventajas de los árboles de decisión
- Poda (pruning)
- Conjuntos de árboles
- Bagging
- Random forest

Ejemplo introductorio – Árbol de regresión

Problema: predecir el **salario** de los jugadores de una liga de baseball en función de **dos descriptores** X=(X1,X2):

X1 = número de años jugados

X2 = número de Hits logrados por el jugador en el año anterior

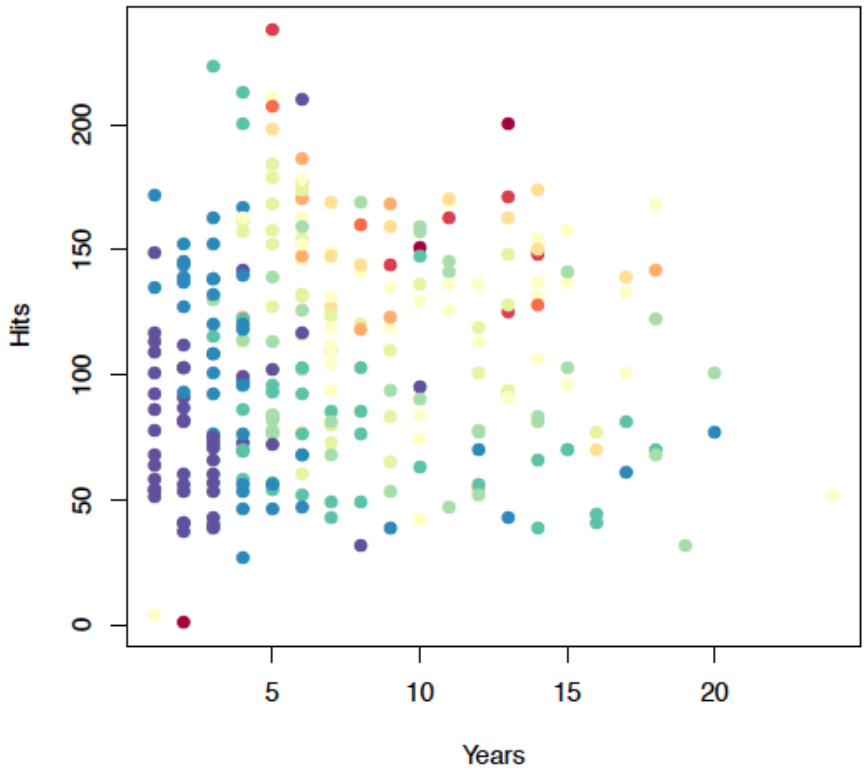
La **respuesta Y** se expresa como el logaritmo del salario en miles de dólares:

$$Y = \log(\text{salario})$$

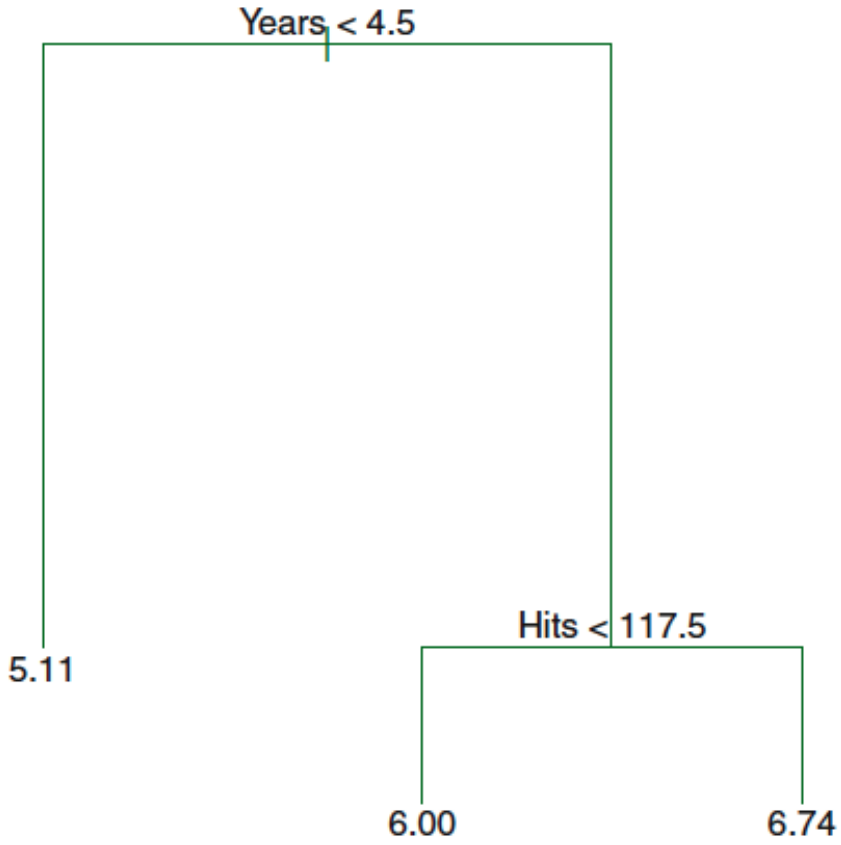
$$\text{Salario} = 1000 \times e^{Y} \text{ dólares}$$

Conjunto de datos de muestras de salarios codificados por colores:

- Bajos (azul, verde)
- Altos (amarillo, rojo)



Ejemplo introductorio - Árbol de regresión



Valores medios de la salida dentro de cada grupo del conjunto de training

Se establecen **reglas de partición** del conjunto de valores de cada descriptor en dos grupos (partición binaria).

La primera partición se define para el descriptor X1=Years (< 4.5).

5.11 es el **valor medio de la salida** dentro del grupo que cumple la regla. El salario medio es

$$1000 \times e^{5.11} = 165174 \text{ dólares}$$

Los jugadores con Years > 4 .5 van a la otra rama.
Se realiza su división en dos grupos usando el descriptor X2=Hits (<117.5).

Ejemplo introductorio - Árbol de regresión

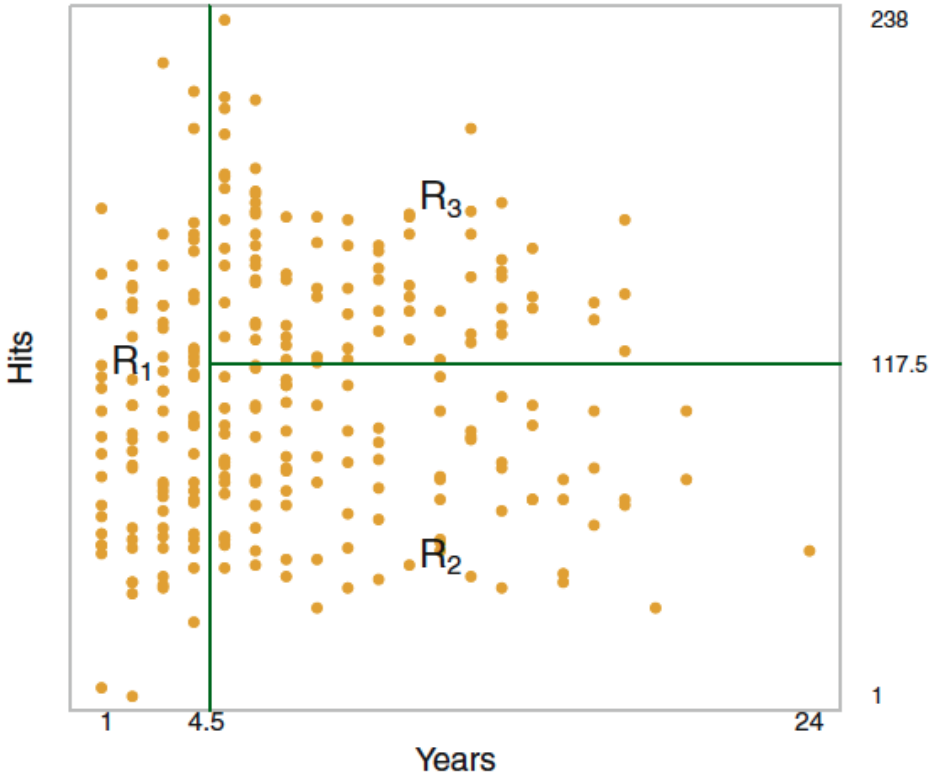
El resultado final es una partición del conjunto de datos en tres regiones.

$$R_1 = \{X \mid \texttt{Years<4.5}\}$$

Salario medio = 165174 $

$$R_2 = \{X \mid \texttt{Years>=4.5}, \texttt{Hits<117.5}\}$$

Salario medio = 402834 $



$$R_3 = \{X \mid \texttt{Years>=4.5}, \texttt{Hits>=117.5}\}$$

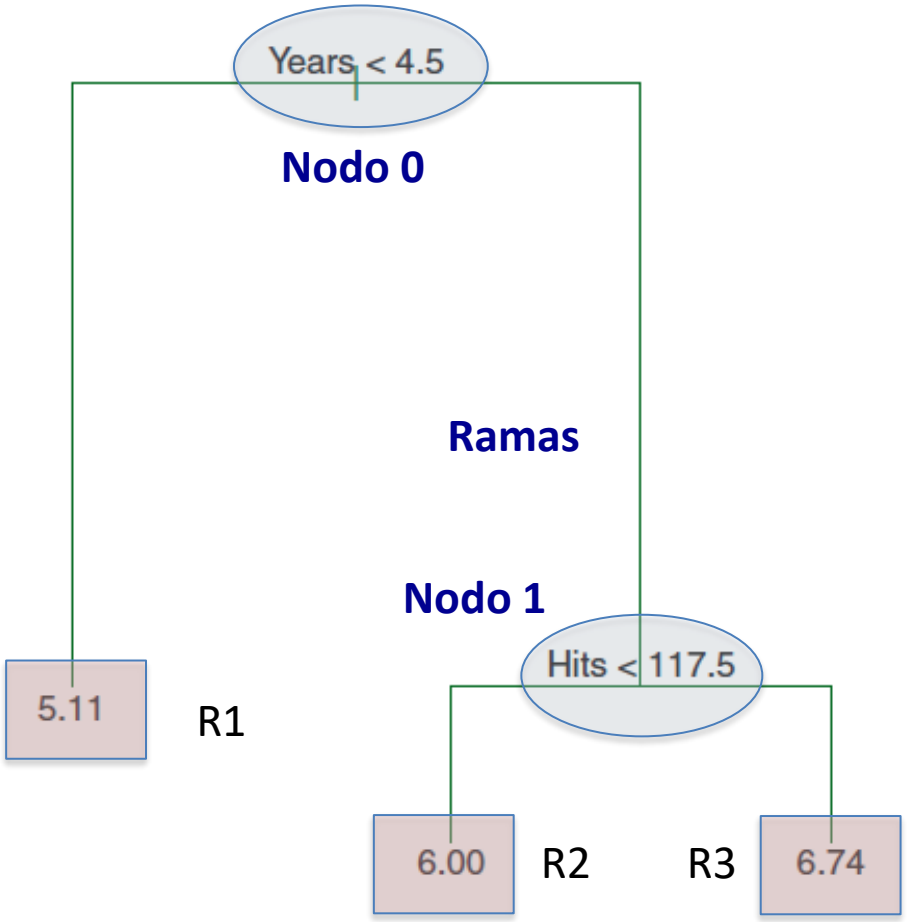
Salario medio = 845346 $

Fronteras de decisión

Ejemplo introductorio - Árbol de regresión

Cómo se construye el árbol

Se construye de arriba hacia abajo usando el conjunto da datos de training (Xi, Y).



Nodos: los puntos donde se toman las decisiones

Hojas: las regiones finales

Interpretación

En este caso, **Years** es el descriptor más significativo, los jugadores con menos experiencia cobran menos, sin importar tanto el número de Hits.

Entre los jugadores con experiencia de más de 4.5 años, ya tiene influencia el número de **Hits** en el salario.

Ejemplo introductorio - Árbol de regresión

Cómo se hacen las predicciones ?

El árbol de decisión es el modelo.

Dada una nueva **observación** con sus valores de los descriptores, se recorre el árbol desde arriba hacia abajo y se obtiene el **valor predicho**, que será el valor obtenido en el training que define la región final de llegada.

Árboles para regresión - Procedimiento general

Dos etapas:

1) **Construcción:** Se busca dividir el espacio de los descriptores (todos sus valores posibles

$$\underbrace{X_1, X_2, \ldots, X_p}_{\mathrm{X}}$$

en un número J de regiones sin solapamiento

$$R_1, R_2, \ldots, R_J$$

2) **Predicción:** Para cada observación que caiga dentro de la región R_j la **predicción** que hacemos de la respuesta es la misma: la **media** de los valores de la respuesta de las observaciones del conjunto de training que están en R_j.

Árboles para regresión - Procedimiento general

CONSTRUCCIÓN DE LAS REGIONES

Se pretende dividir el espacio de los descriptores en rectángulos (cajas). El objetivo es encontrar las cajas

$$R_1, R_2, \ldots, R_J$$

que minimizan el error cuadrático (RSS)

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

donde $\hat{y}_{R_j}$ es la media de las respuestas del conjunto de training dentro de la caja j.

Se utiliza un procedimiento sistemático de división binaria recursiva

Árboles para regresión - Procedimiento general

División binaria recursiva

Inicialmente seleccionamos uno de los descriptores: X_j

y un valor de corte s para separar el espacio de los predictores en **dos regiones**

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

Buscamos los valores de *j* y *s* que minimizan la función

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

donde $\hat{y}_{R_1}$ $\hat{y}_{R_2}$ es la media de las respuestas del conjunto de training en $R_1(j,s)$ $R_2(j,s)$

Árboles para regresión - Procedimiento general

División binaria recursiva

A continuación repetimos el proceso de separación binaria, pero ahora no sobre todo el espacio sino sobre una de las dos regiones obtenidas en el paso anterior. Así se tienen 3 regiones.

De nuevo, se repite el proceso de separación binaria en una de estas tres regiones para minimizar la función.

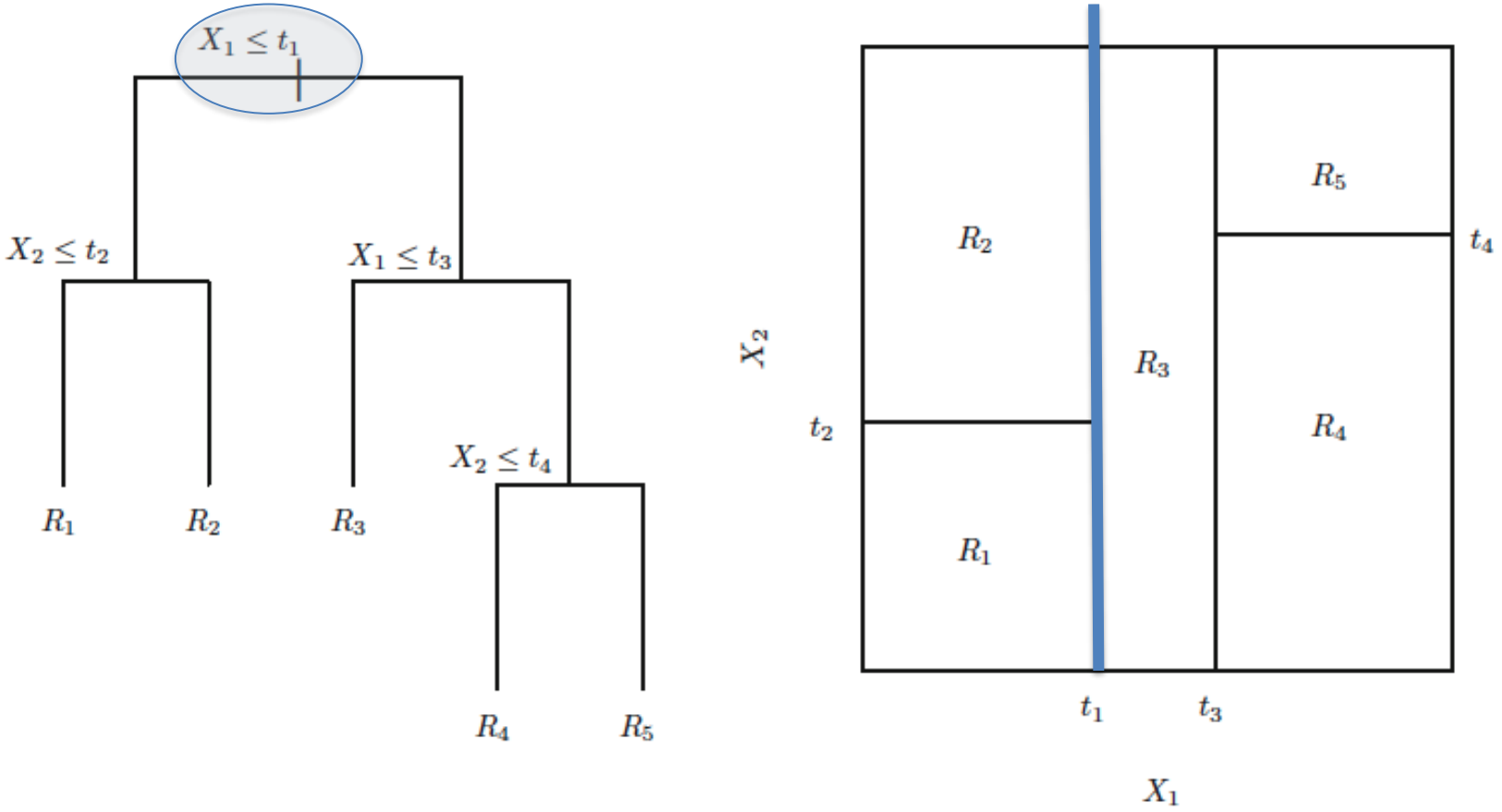
El proceso sigue hasta que se alcanza un criterio de parada y se tienen las regiones

$$R_1, R_2, \ldots, R_J$$

Cada región queda definida por la **media** de los valores de la respuesta de las observaciones del conjunto de training que están en la región.

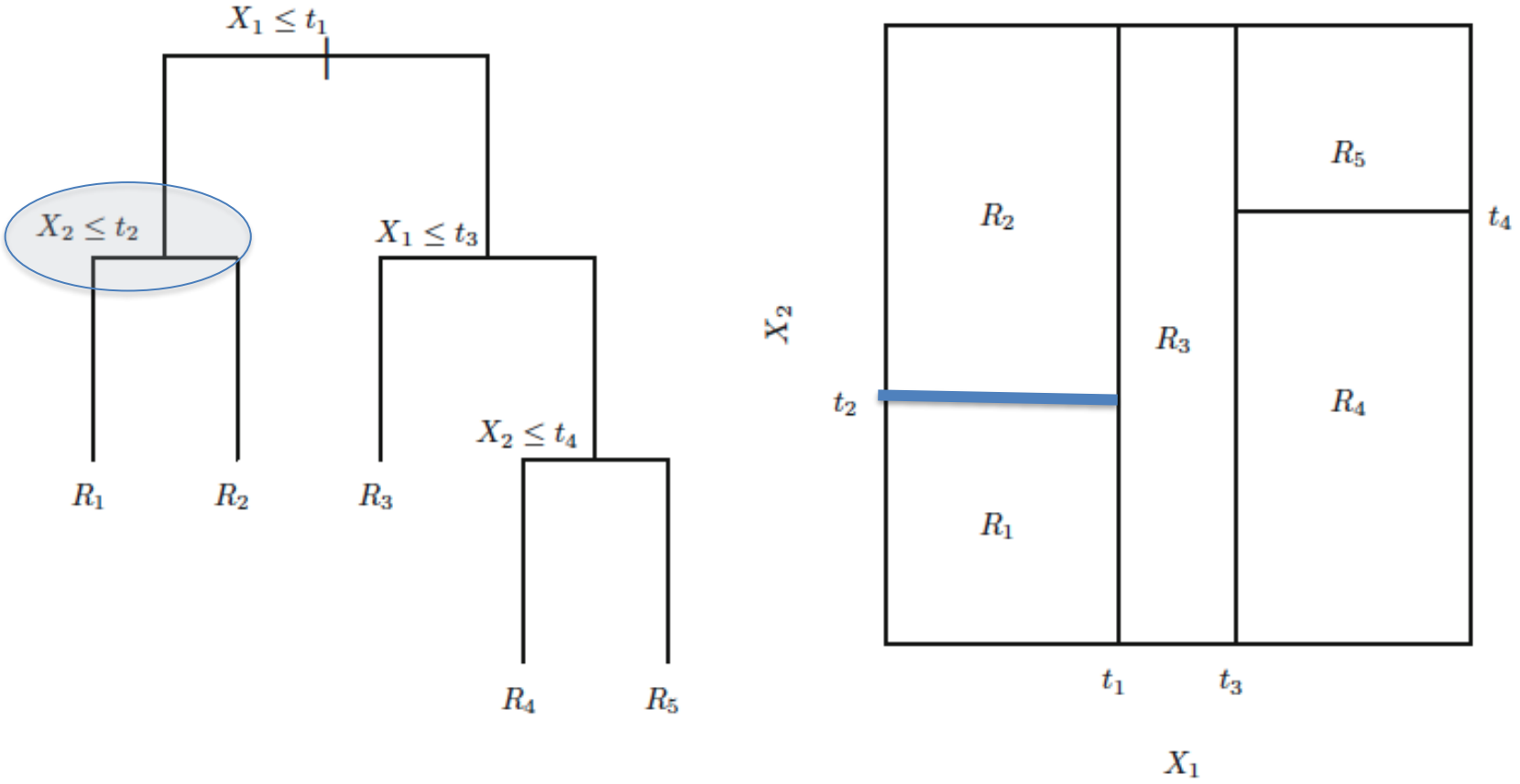
Árboles para regresión - Procedimiento general

Ejemplo



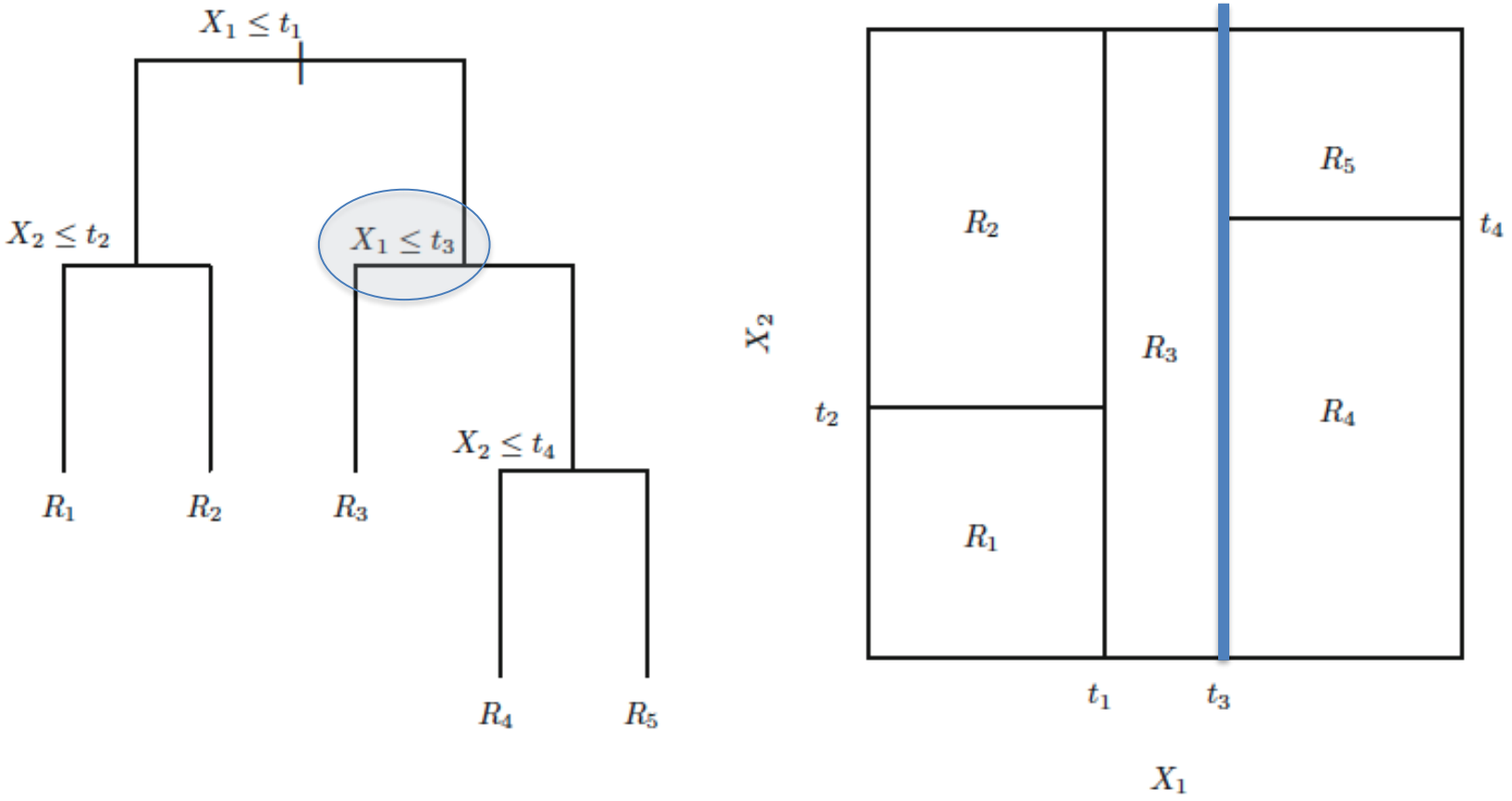
Árboles para regresión - Procedimiento general

Ejemplo



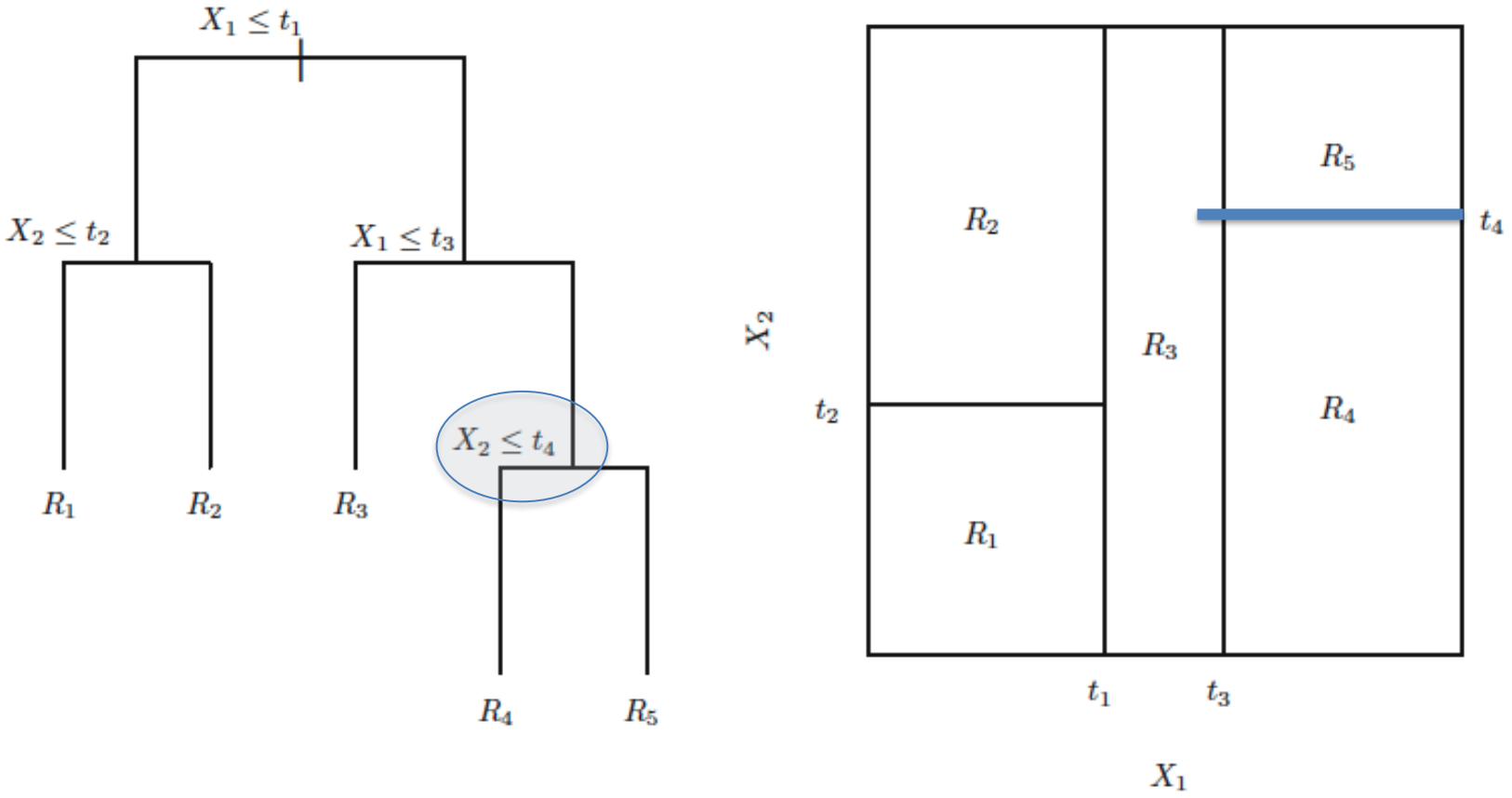
Árboles para regresión - Procedimiento general

Ejemplo



Árboles para regresión - Procedimiento general

Ejemplo



Árboles para Clasificación

El árbol de clasificación es similar al de la regresión pero ahora se trata de predecir una **variable cualitativa**: **la clase a la que pertenece un objeto**.

Ahora el objetivo es que cada región quede definida por **la clase que ocurra con mayor frecuencia** para las observaciones del conjunto de training en la región en cuestión.

Para construir el árbol, se usa también la **división binaria recursiva**, pero no puede usarse el criterio RSS para hacer las particiones.

Una alternativa natural es la **proporción de errores de clasificación**, es decir, la proporción de observaciones del conjunto de training en la región obtenida en el **nodo m** que no pertenecen a la clase más frecuente:

$$E = 1 - \max_k(\hat{p}_{mk})$$

donde $\hat{p}_{mk}$ es la proporción de observaciones en la región del nodo m que son de la clase k.

Árboles para Clasificación

Existen otros criterios mejores que el RMS para la creación de árboles de clasificación. Uno de ellos es el **índice de Gini**:

$$G_m = \sum_{k=1}^{K} \hat{p}_{mk}(1 - \hat{p}_{mk}) = 1 - \sum_{k=1}^{K} \hat{p}_{mk}^2$$

donde *K* es el número total de clases.

Este índice mide la **pureza del nodo:**

Si el valor es pequeño, indica que el nodo contiene sobre todo observaciones de una única clase.

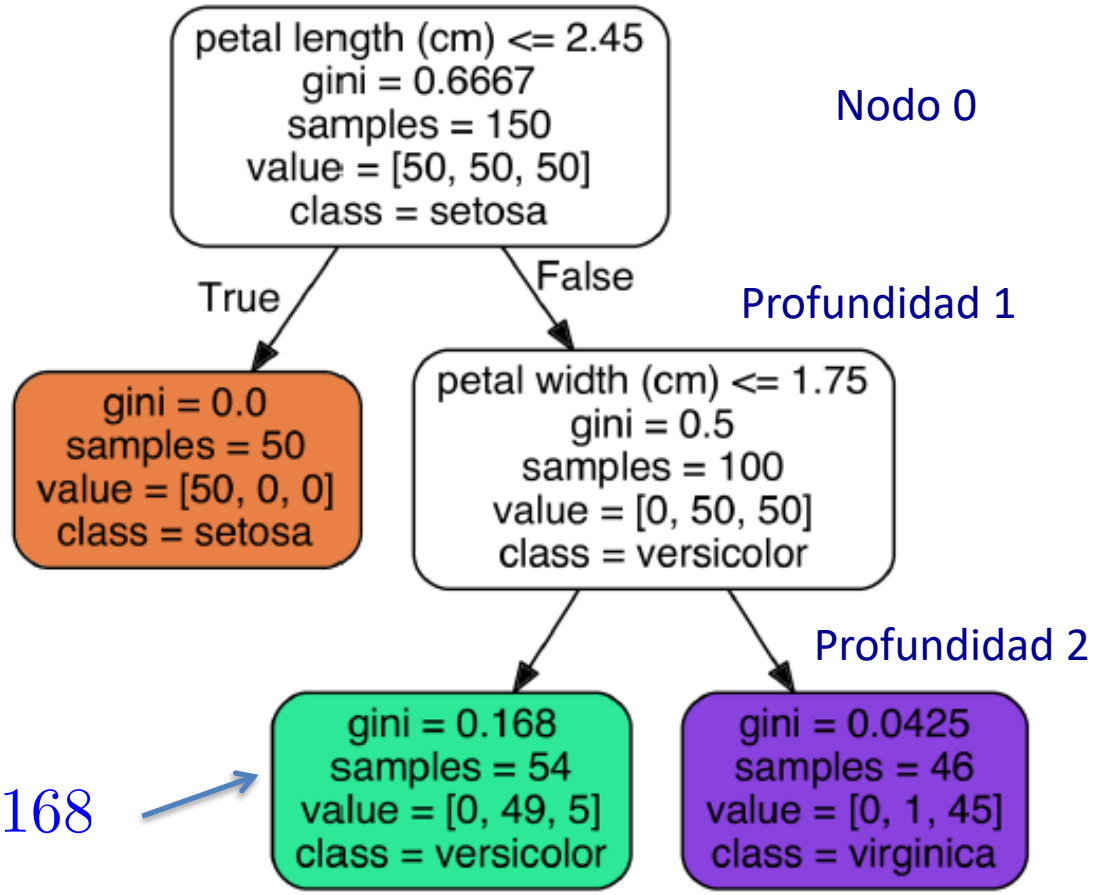
Un índice alternativo (aunque con interpretación similar) es la **entropía**

$$D = -\sum_{k=1}^{K} \hat{p}_{mk} \log \hat{p}_{mk}$$

Árboles para clasificación

Ejemplo: Iris dataset

Longitud y anchura de los pétalos (**descriptores**) de tres **especies** de flores iris: Setosa, Versicolor, Virgínica



Índice Gini

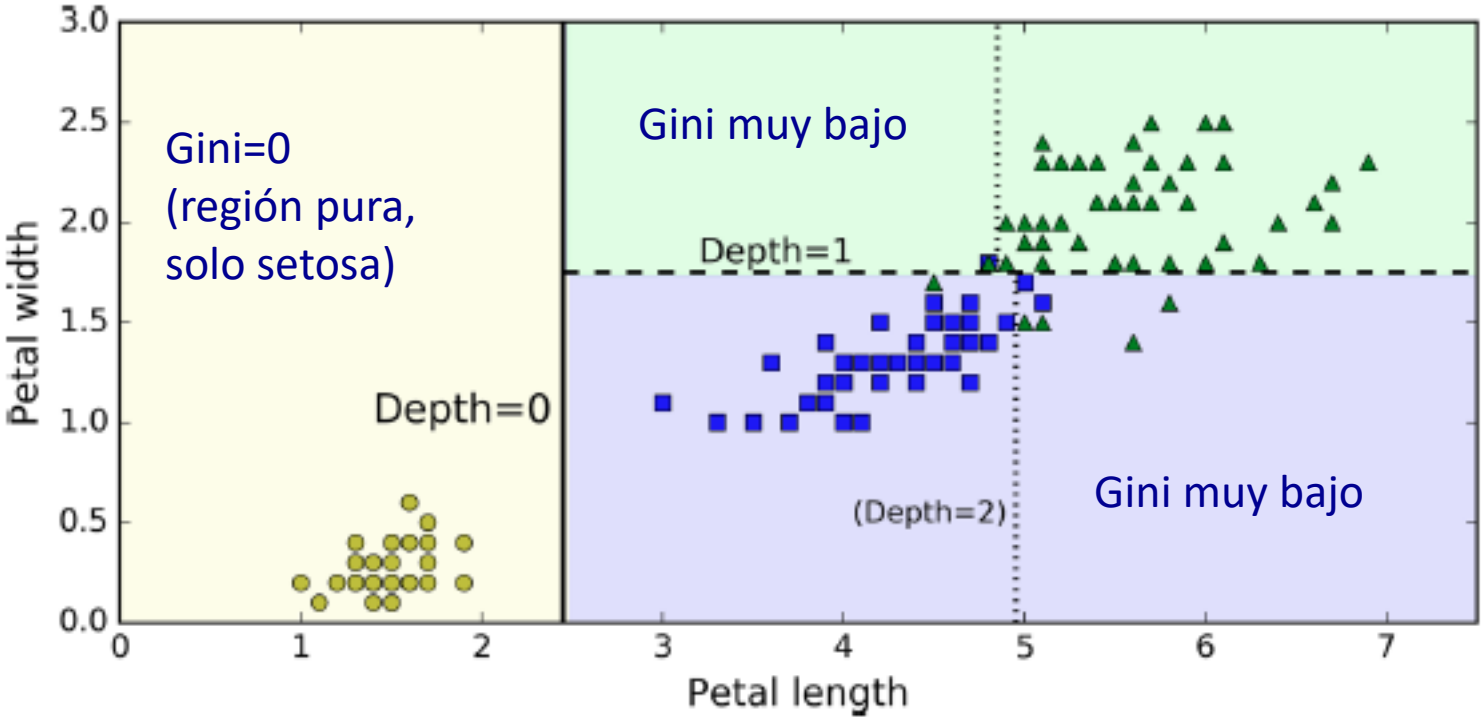
$$G_m = 1 - \sum_{k=1}^{K} \hat{p}_{mk}^2$$

$$= 1 - \frac{0}{54} - \frac{49}{54} - \frac{5}{54} \approx 0.168$$

Árboles para clasificación

Ejemplo: Iris dataset

Longitud y anchura de los pétalos (**descriptores**) de tres **especies** de flores iris: Setosa, Versicolor, Virgínica



Fronteras de decisión

Árboles para clasificación

Construcción de las regiones

Procedimiento de división binaria recursiva

Como en el caso de la regresión, se inicia la creación del árbol dividiendo el conjunto de training en dos partes usando un único descriptor j y un valor umbral (por ejemplo, petal length <= 2.45).

El descriptor y el valor umbral se eligen de manera que se obtengan los subconjuntos más puros según el índice Gini, es decir el mínimo de la función:

$$\frac{n_{izq}}{n} G_{izq} + \frac{n_{der}}{n} G_{der}$$

donde G_{izq}, G_{der} son los índices de impureza en los subconjuntos de izquierda y derecha

n_{izq}, n_{der}, n son los números de elementos en ambos subconjuntos y en el total

Árboles para clasificación

Construcción de las regiones

Procedimiento de división binaria recursiva

El proceso anterior se va repitiendo recursivamente, separando los subconjuntos usando la misma idea, hasta que se alcanza un criterio de parada o no puede encontrarse una división que reduzca el índice de impureza.

Algunos algoritmos permiten fijar una profundidad máxima como un hiperparámetro.

Ventajas y desventajas de los árboles de decisión

Ventajas

Los árboles son muy fáciles de explicar. Siguen una lógica cercana a la forma humana de razonar.

Son fáciles de interpretar especialmente si son pequeños. Admiten una visualización gráfica intuitiva.

Son modelos del tipo "***caja blanca***". Puede seguirse fácilmente la influencia de los distintos descriptores en el proceso de predicción.

Conviven fácilmente predictores cuantitativos y cualitativos sin necesidad de introducir variables ficticias.

Ventajas y desventajas de los árboles de decisión

Desventajas

El **entrenamiento** (creación del árbol) es computacionalmente costoso puesto que, en cada nodo, el proceso de división requiere comparar todos los descriptores. Esto es especialmente notable cuando el número de descriptores y/o el de observaciones son grandes.

No obstante, la realización de **predicciones** es rápida porque atravesar cada nodo implica un solo descriptor.

Los árboles necesitan muy pocas hipótesis sobre los datos y no tienen parámetros. Si no se imponen restricciones, la estructura se adapta a los datos del conjunto de entrenamiento. Esto puede implicar ***"overfitting"***.

Ventajas y desventajas de los árboles de decisión

Desventajas

Para evitar el overfitting con el conjunto de training, suelen introducirse **restricciones** en la construcción del árbol de decisión, es decir alguna forma de regularización asociada a **hiperparámetros**.

Por ejemplo, limitar:

- el número máximo de regiones finales
- el número mínimo de muestras que debe haber en un nodo para que pueda dividirse
- el máximo número de descriptores que se evalúan para dividir los nodos

Otra estrategia para reducir la complejidad del árbol es la **poda (pruning)**

Pruning

Consiste en desarrollar un árbol de gran tamaño, sin restricciones, y posteriormente volver hacia atrás ‘podando ramas’ que cuelgan de ciertos nodos.

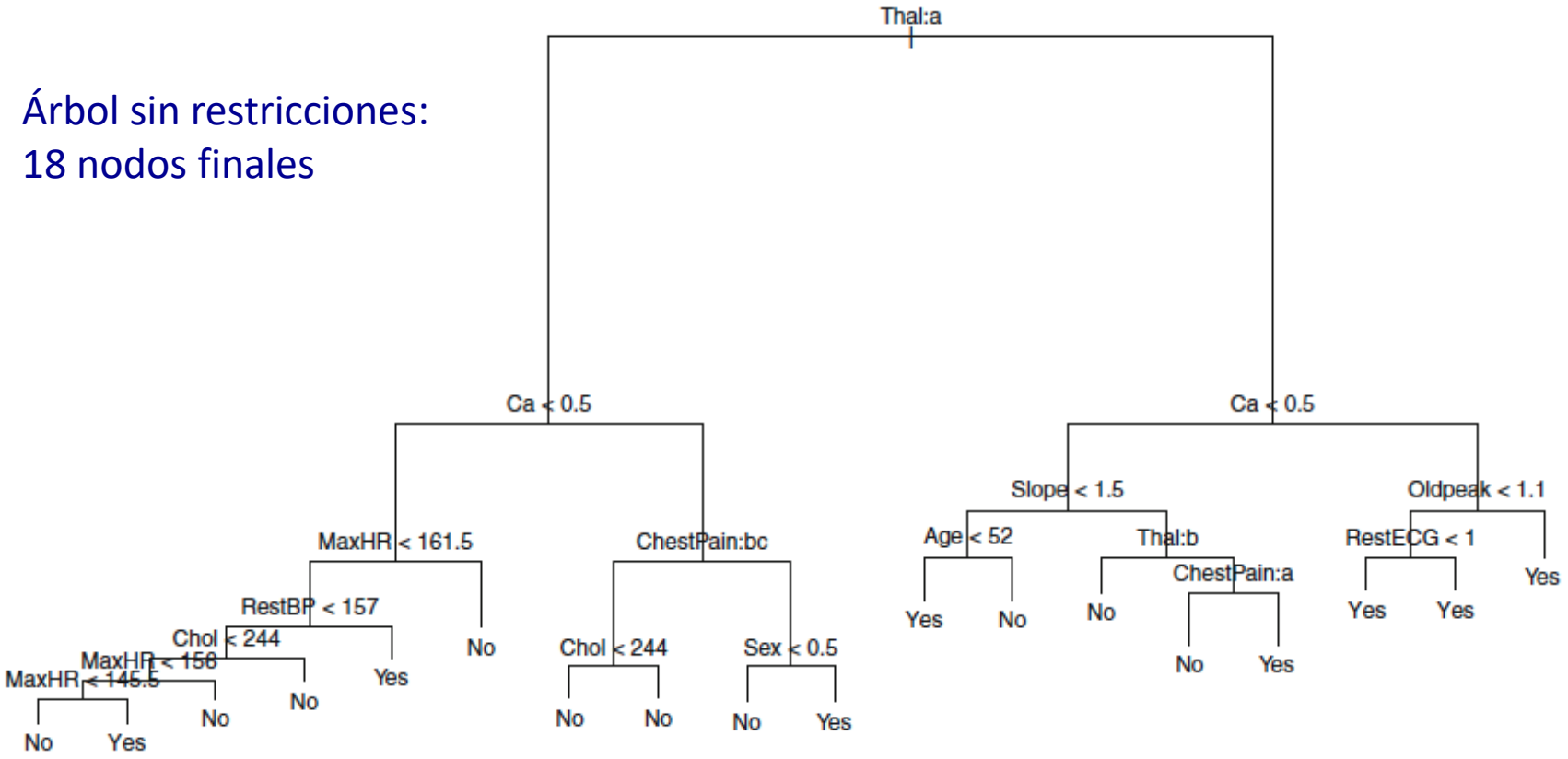
Se trata de eliminar subárboles enteros que cuelgan de dichos nodos y sustituirlos por una hoja final.

Pruning

Ejemplo de poda (pruning)

Heart data set: 303 pacientes
13 descriptores (edad, sexo, colesterol, Thalium test, ….
Salida cualitativa (dolor de pecho): YES NOT

Árbol sin restricciones:
18 nodos finales

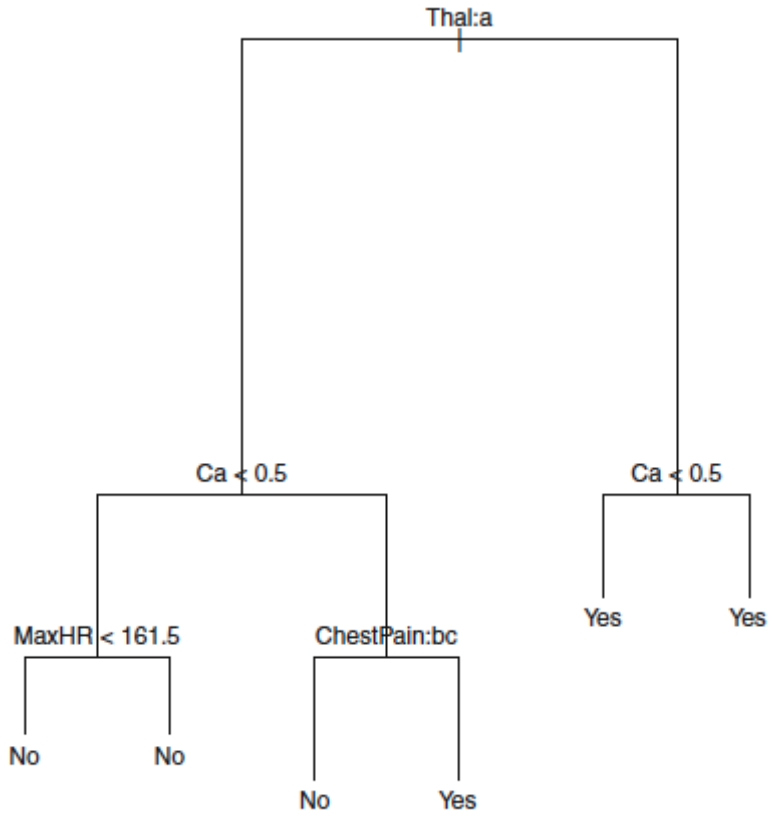


Pruning

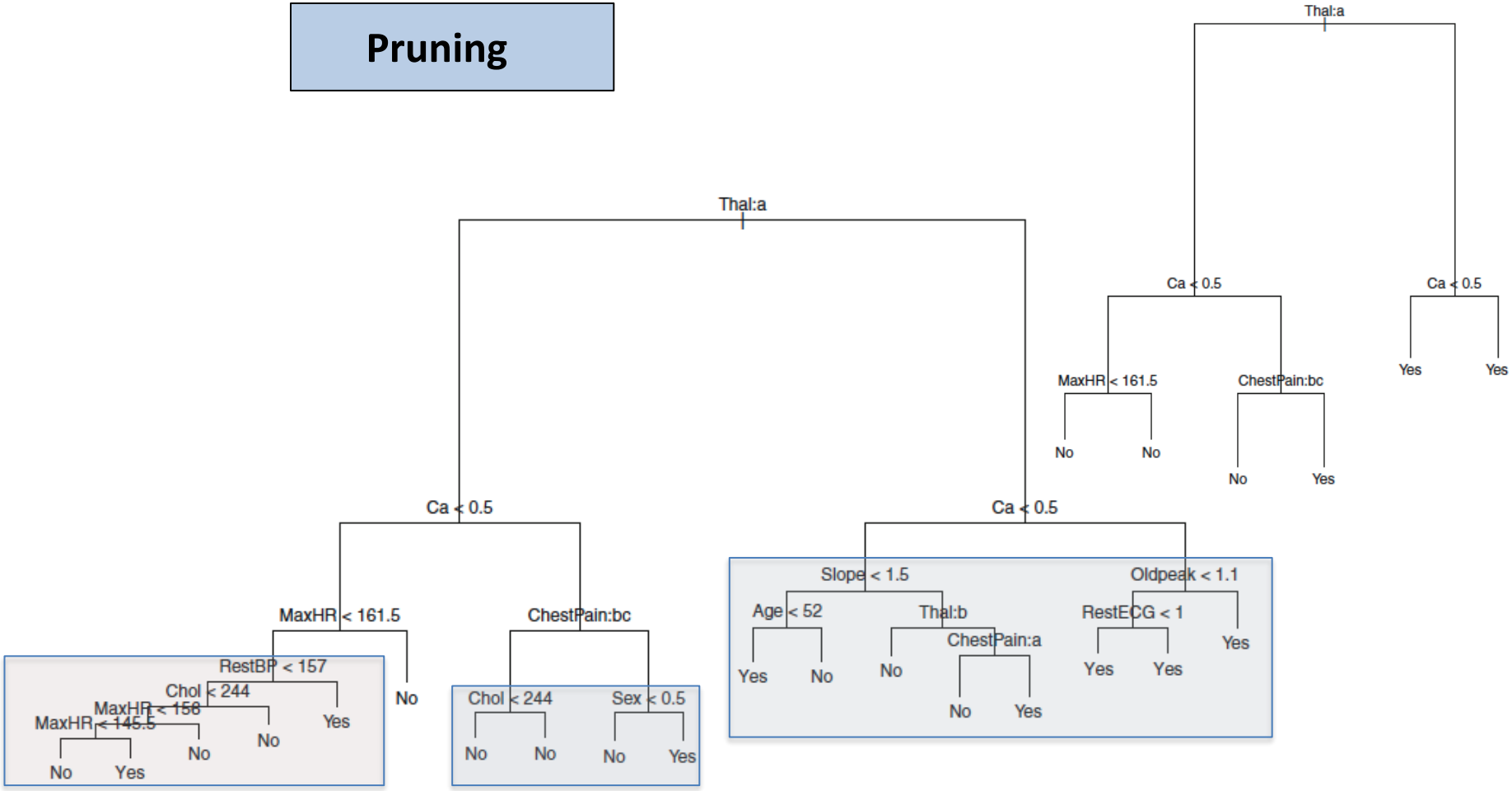
Ejemplo de poda (pruning)

Heart data set: 303 pacientes
13 descriptores (edad, sexo, colesterol, Thalium test, ….
Salida cualitativa (dolor de pecho): YES NOT

Árbol después del pruning:
6 nodos finales



Pruning


Thal:a
Ca < 0.5
MaxHR < 161.5
ChestPain:bc
RestBP < 157
Chol < 244
MaxHR < 156
MaxHR < 145.5
No
Yes
No
No
No
Chol < 244
Sex < 0.5
No
No
No
Yes
Ca < 0.5
Slope < 1.5
Oldpeak < 1.1
Age < 52
Thal:b
RestECG < 1
Yes
ChestPain:a
Yes
No
No
Yes
No
Yes
Yes
Yes
Thal:a
Ca < 0.5
Ca < 0.5
MaxHR < 161.5
ChestPain:bc
Yes
Yes
No
No
No
Yes

Ventajas y desventajas de los árboles de decisión

Desventajas

Los árboles de decisión no tienen el mismo nivel de exactitud en la predicción en comparación con otros métodos de clasificación y regresión.

Existen técnicas para **agrupar** diferentes árboles de decisión y así mejorar la capacidad predictiva, tales como

- Bagging
- Boosting
- Random forest

Conjuntos de clasificadores (Ensemble learning)

Un **único árbol** profundo → bajo sesgo pero **alta varianza**

La idea ahora es:

1) Separar conjunto de entrenamiento aleatoriamente en diferentes subconjuntos
2) Entrenar un árbol para cada subconjunto
3) Combinar los resultados

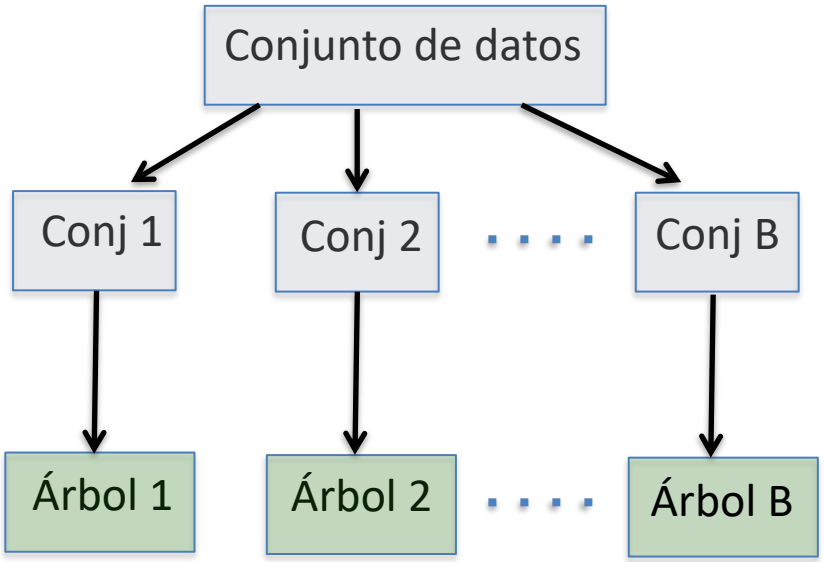
Sabido que

$$\begin{cases} Z_1, Z_2, \cdots, Z_n \text{ con varianza } \sigma^2 \\ \text{La media } \bar{Z} \text{ tiene varianza } \dfrac{\sigma^2}{n} \end{cases}$$

Por tanto, promediar un conjunto de observaciones reduce la varianza

Bagging

Bootstrap aggregating



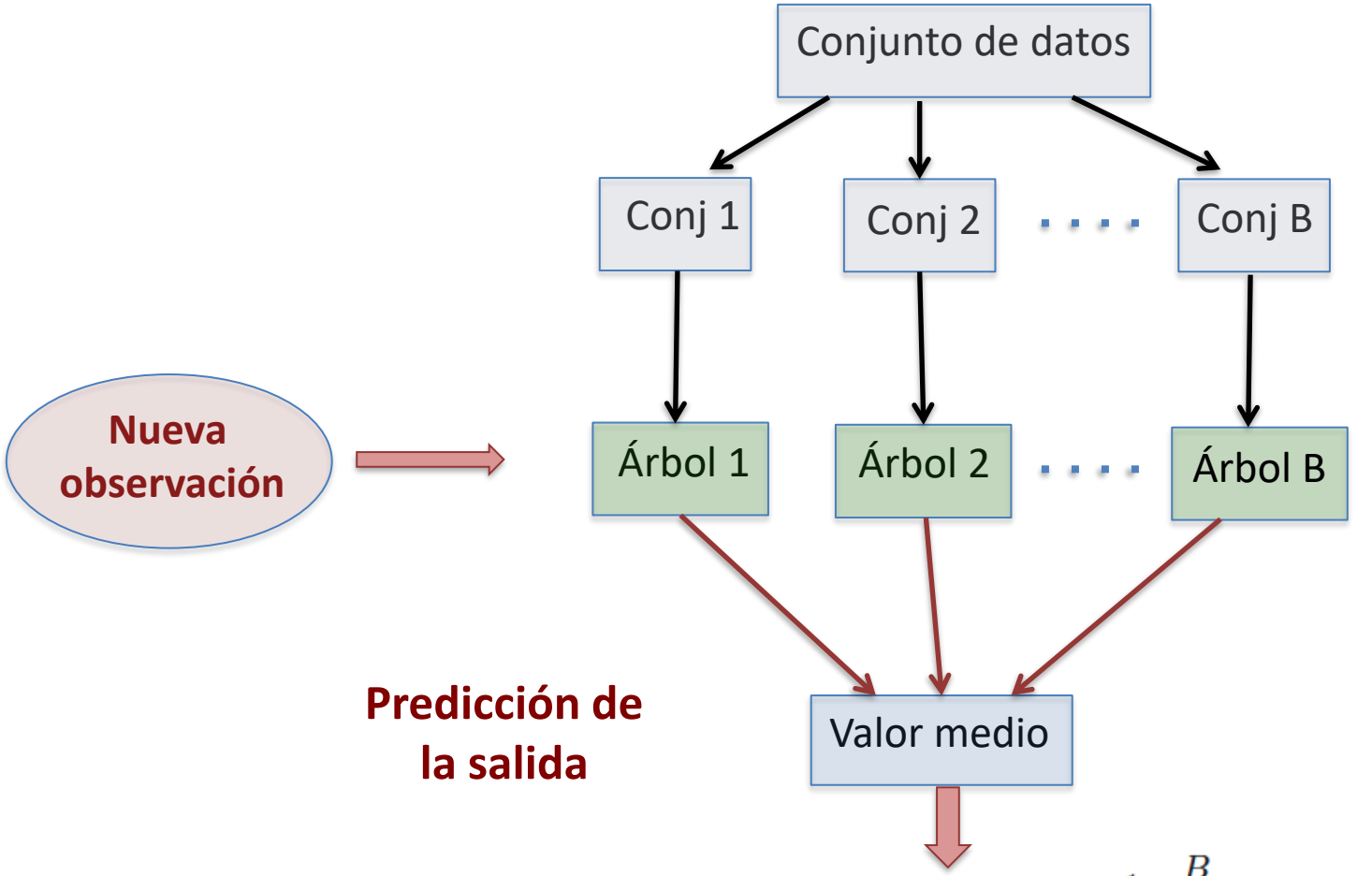
Conjuntos generados con bootstrap

Entrenamiento de B árboles de decisión

Una vez entrenados los árboles, cómo se realizan las predicciones ?

Bagging

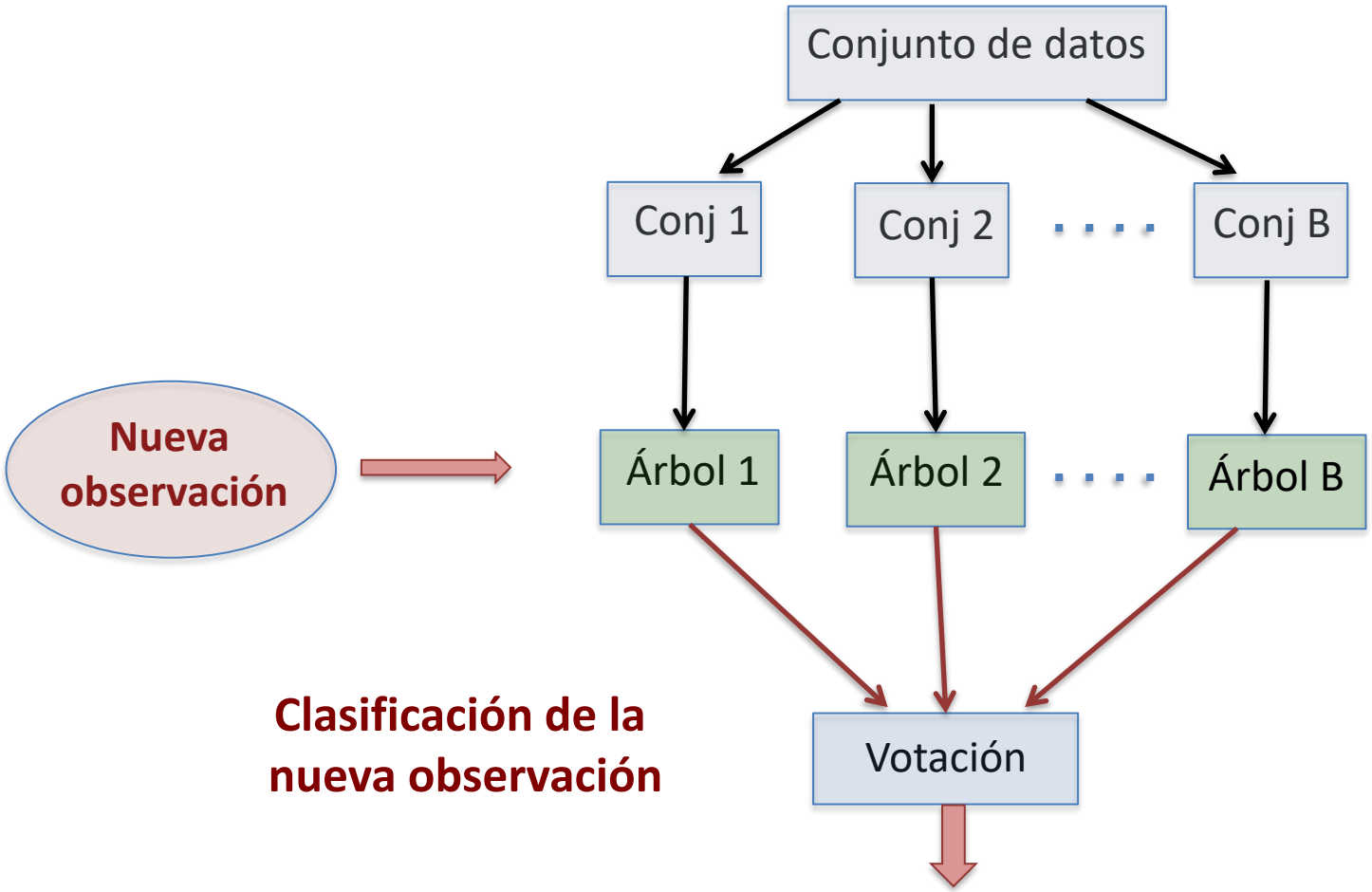
Regresión



$$\hat{f}_{\text{bag}}(x) = \frac{1}{B}\sum_{b=1}^{B} \hat{f}^{*b}(x).$$

Bagging

Clasificación



Clasificación de la nueva observación

La clase mayoritaria entre los B clasificadores

Random forest

Problema del bagging con los árboles de decisión

Supongamos que u**n descriptor (o unos pocos) es muy fuerte en comparación con los otros**. Entonces, en el conjunto de árboles agregados, la mayoría usarán este descriptor en la división del primer nodo.

De esta forma los árboles serán parecidos y sus predicciones estarán muy correlacionadas.

Promediar variables altamente correlacionadas a una reducción de la varianza tan marcada como en el caso de variables altamente no correlacionadas.

Por tanto, **el bagging no reducirá la varianza como sería deseable**.

Random forest

Se sigue utilizando el bagging con un número B de árboles.

Random forest soluciona el problema eligiendo, en cada división, solo un subconjunto con un **número m de los p predictores elegidos al azar**.
Solo uno de estos m predictores se usa en la división.

Habitualmente se escoge $m \approx \sqrt{p}$

lo que hace que se elige un número pequeño de descriptores

Out – of - bag

Cuando se usa el muestreo “bootstrap” sobre un conjunto de datos, se demuestra que se acaba usando alrededor de 2/3 del número de datos.

Esto hace que, usando tanto el bagging o el random forest, cada uno de los árboles usará 2/3 de las observaciones del conjunto de training.

El 1/3 restante (no utilizadas en el training) se denominan **observaciones out-of-bag” (OOB).**

Qué podemos hacer con el 1/3 de las observaciones no utilizadas ?

Usarlas para hacer una evaluación del clasificador (testing)

Test usando Out – of – bag

Cada una de las observaciones del out-of-bag se entran en cada uno de los B árboles, con lo que se obtienen aproximadamente B/3 respuestas predichas.

- En el caso de la **regresión**, obtenemos una respuesta única calculando la media. Se calcula el **error cuadrático** para el conjunto de las observaciones utilizadas.

- En el caso de la **clasificación**, obtenemos la mayoría y el error global de clasificación (proporción de aciertos).

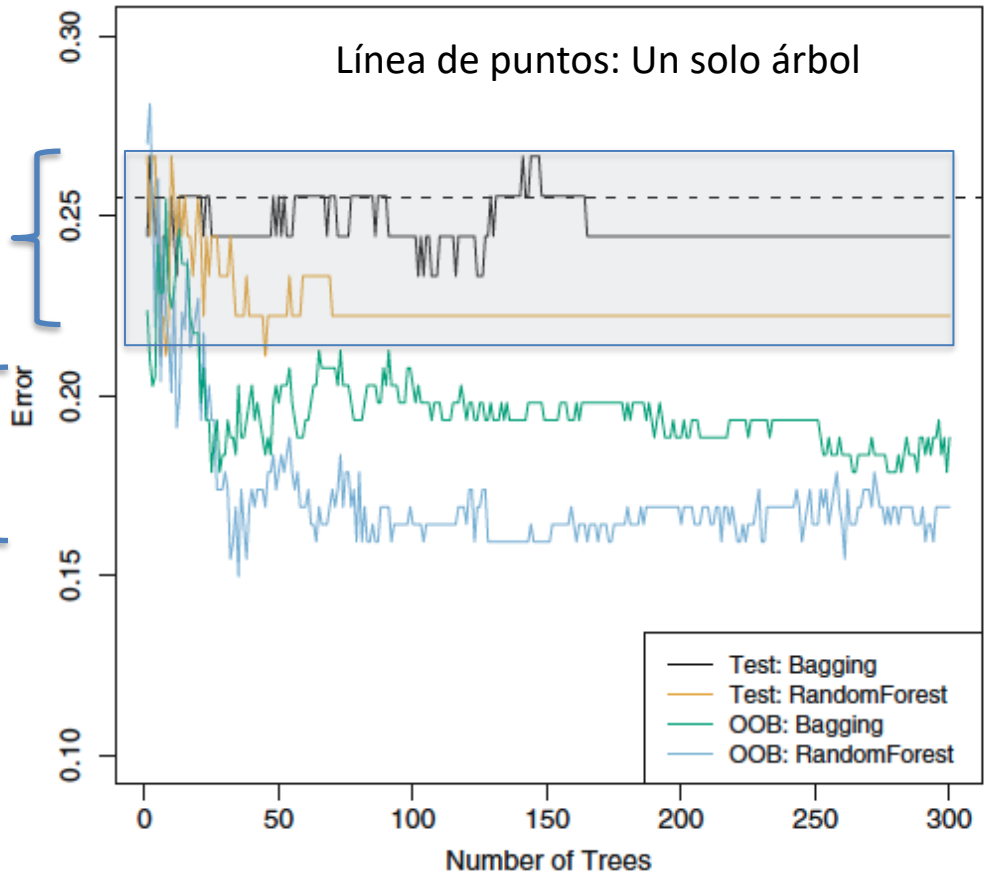
Test usando Out – of – bag

Ejemplo

Heart data set: 303 pacientes
13 descriptores (edad, sexo, colesterol, Thalium test, ….
Salida cualitativa (dolor de pecho): YES NOT

Errores de test

Errores OOB



Random forest con

$m \approx \sqrt{p}$

Resumen

Los **árboles de decisión** son sencillos de explicar y fácilmente interpretables, tanto para clasificación como regresión.

A menudo no son competitivos con otros métodos en cuanto a la exactitud de las predicciones.

Bagging, random forest y boosting (no explicado en esta clase) reducen el error de predicción de los árboles.

Funcionan creciendo muchos árboles con el conjunto de training y combinando sus predicciones para dar una única predicción final.

Random forest y boosting están entre los métodos más utilizados de aprendizaje supervisado, aunque sus resultados pueden ser difíciles de interpretar.