

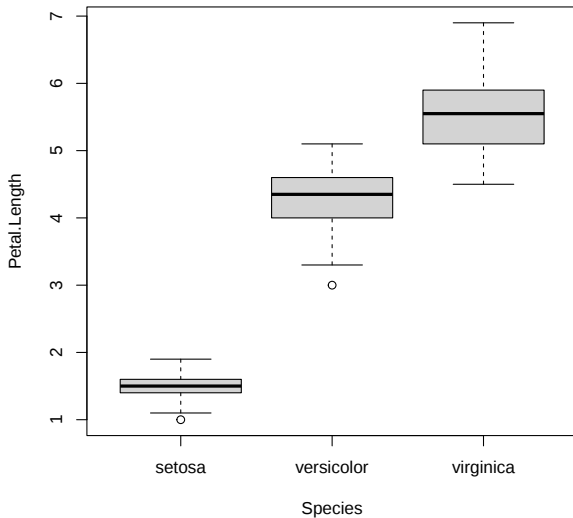
Árboles de decisión

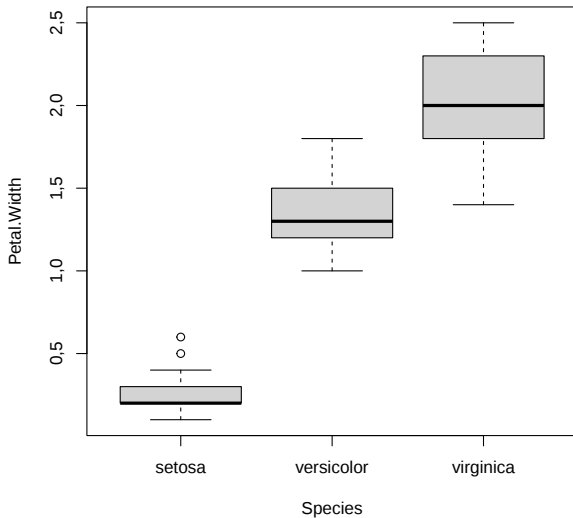
N. Corral,
C. Carleos

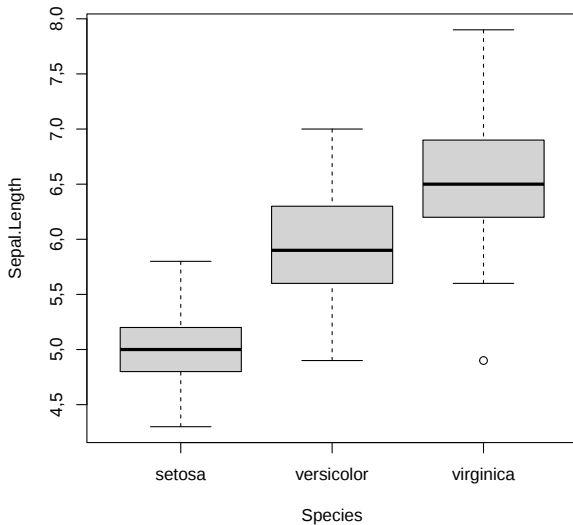
18 de marzo de 2024

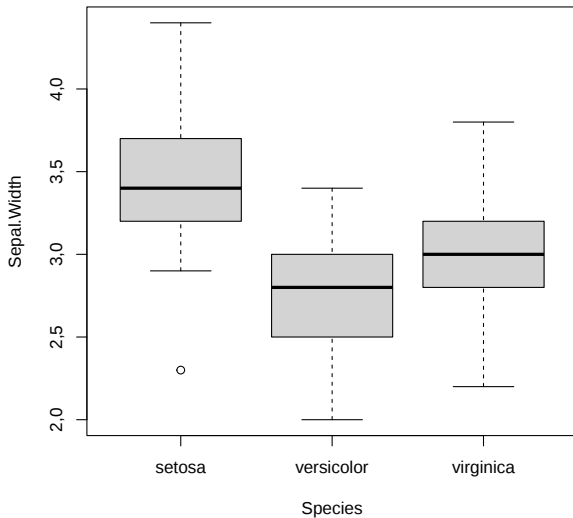
- 1 Ejemplo
- 2 Generalidades
- 3 Árboles de clasificación
- 4 Árboles de regresión
- 5 Resumen
- 6 Ejercicios
- 7 Bibliografía

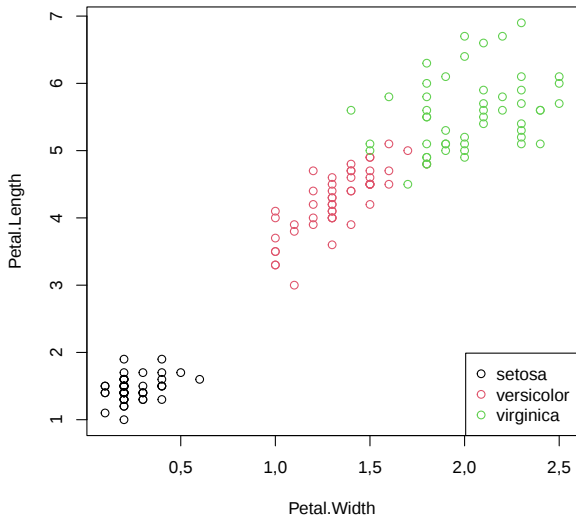
Sepal.Length	Sepal.Width	Petal.Length
Min. :4,30	Min. :2,00	Min. :1,00
1st Qu.:5,10	1st Qu.:2,80	1st Qu.:1,60
Median :5,80	Median :3,00	Median :4,35
Mean :5,84	Mean :3,06	Mean :3,76
3rd Qu.:6,40	3rd Qu.:3,30	3rd Qu.:5,10
Max. :7,90	Max. :4,40	Max. :6,90
Petal.Width	Species	
Min. :0,1	setosa :50	
1st Qu.:0,3	versicolor:50	
Median :1,3	virginica :50	
Mean :1,2		
3rd Qu.:1,8		
Max. :2,5		

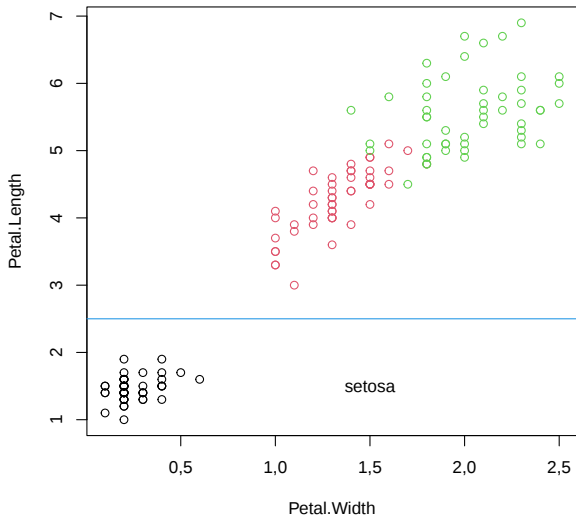


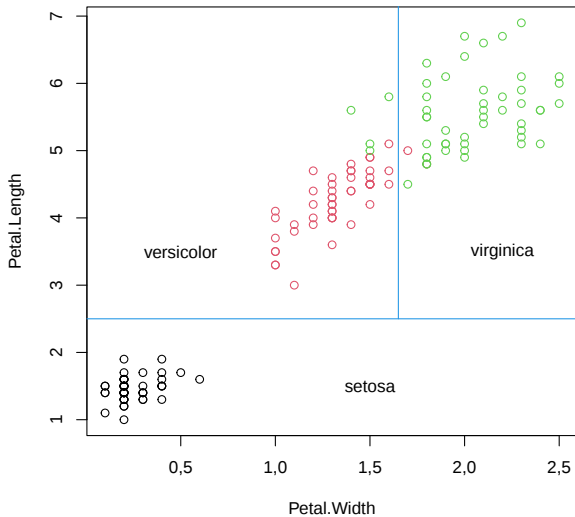


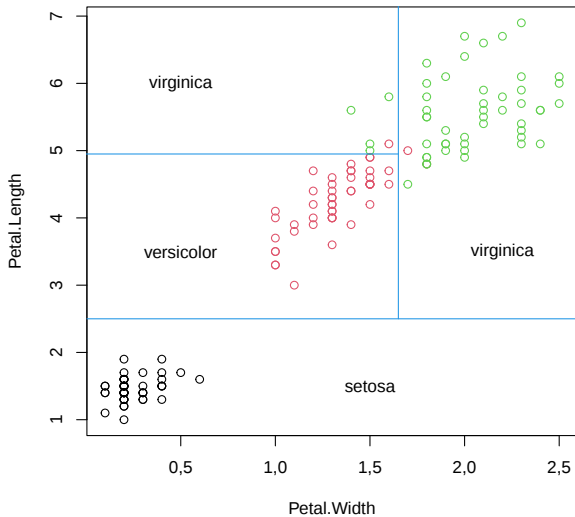












```
> library (rpart.plot) # basta rpart para cálculos
> árbol <- rpart (Species ~
+                 Petal.Length + Petal.Width,
+                 iris)
> árbol
n= 150

node), split, n, loss, yval, (yprob)
      * denotes terminal node
```

```
1) root 150 100 setosa (0,33333 0,33333 0,33333)
  2) Petal.Length< 2.45 50    0 setosa (1,00000 0,00000 0,00000)
  3) Petal.Length>=2.45 100  50 versicolor (0,00000 0,00000 0,00000)
    6) Petal.Width< 1.75 54    5 versicolor (0,00000 0,00000 0,00000)
    7) Petal.Width>=1.75 46    1 virginica (0,00000 0,00000 1,00000)
```



> *prp* (árbol)

Ejemplo

Generalidades

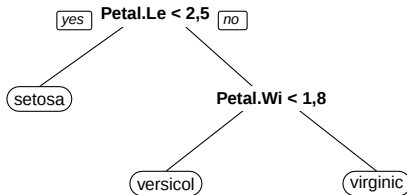
Árboles de clasificación

Árboles de regresión

Resumen

Ejercicios

Bibliografía



```
> prp (árbol, yes.text="verdadero", no.text="falso")
```

Ejemplo

Generalidades

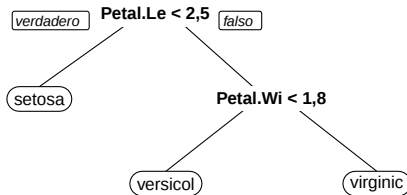
Árboles de clasificación

Árboles de regresión

Resumen

Ejercicios

Bibliografía



`> prp (árbol, type=3)`

Ejemplo

Generalidades

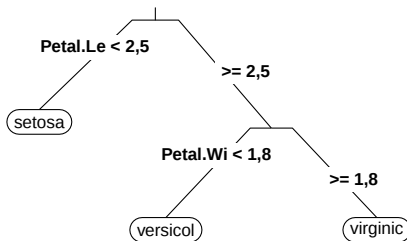
Árboles de clasificación

Árboles de regresión

Resumen

Ejercicios

Bibliografía



`> prp (árbol, type=4)`

Ejemplo

Generalidades

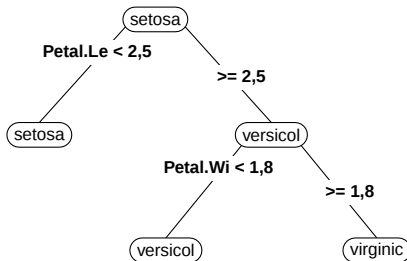
Árboles de clasificación

Árboles de regresión

Resumen

Ejercicios

Bibliografía



Nomenclatura

Ejemplo

Generalidades

Árboles de
clasificación

Árboles de
regresión

Resumen

Ejercicios

Bibliografía

- predictores: $X_1, \dots, X_p$
- respuesta: Y
 - cualitativa: árbol de clasificación
 - cuantitativa: árbol de regresión
- el árbol consta de *ramas*
- cada rama nace de un *nodo* o *nudo*
- cada nodo representa un subconjunto de la muestra
- la muestra completa es el nodo *raíz*
- nodo k^o : regla de decisión asociada a cierta X_i
 - X_i cualitativa: ¿ $X_i \in A_{ik}$?
 - X_i cuantitativa: ¿ $X_i \leq c_k$?
- los nodos finales se llaman *hojas*;
representan una partición de la muestra

CHAID 1980. CHi² Automatic Interaction Detection. Ramificaciones múltiples. Sólo clasificación.

CART 1984. Classification and Regression Trees. Ramificaciones binarias. El más usado en R.

ID3→C4.5 1986. El más popular en Weka (llamado J48). Ramificaciones múltiples. Sólo clasificación.

MARS 1991. Multivariate Adaptive Regression Splines. Sólo regresión (lineal a trozos).

CIT 2006. Conditional Inference Trees. Basados en contrastes múltiples no paramétricos.

Elementos para la construcción

Ejemplo

Generalidades

Árboles de
clasificación

Árboles de
regresión

Resumen

Ejercicios

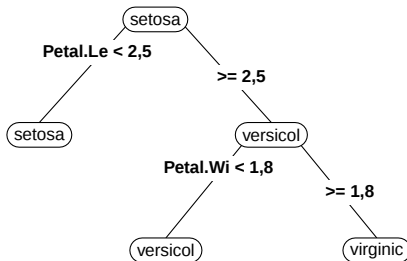
Bibliografía

- Método para elegir hoja candidata a división.
- Criterio de parada.
- Método para asignar a un nodo una predicción.

Notación de los nodos

- t un cierto nodo
- t_L nodo hijo izquierdo
- t_R nodo hijo derecho
- T conjunto de las hojas del árbol

`> prp (árbol, type=4)`



```
> árbol $ frame [, 1:5]           # todos los nodos
```

	var	n	wt	dev	yval
1	Petal.Length	150	150	100	1
2	<leaf>	50	50	0	1
3	Petal.Width	100	100	50	2
6	<leaf>	54	54	5	2
7	<leaf>	46	46	1	3

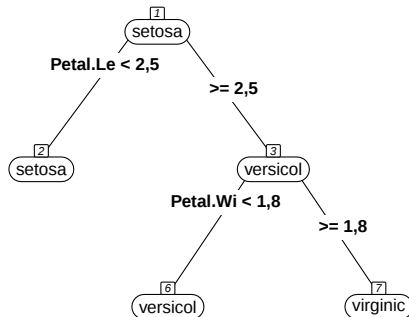
```
> árbol $ where [c(1:5,145:150)] # hojas
```

1	2	3	4	5	145	146	147	148	149	150
2	2	2	2	2	5	5	5	5	5	5

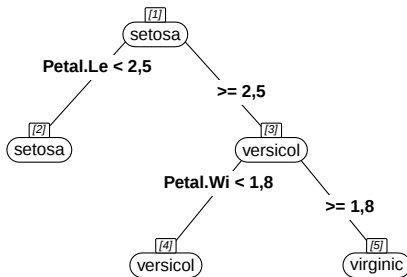
```
> table (árbol$where)
```

2	4	5
50	54	46

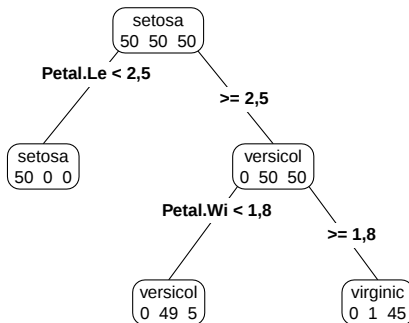
`> prp (árbol, type=4, nn=TRUE)`



```
> prp (árbol, type=4, ni=TRUE)
```



```
> prp (árbol, type=4, extra=1)
```



Elegir siguiente corte: medidas de impureza

$$\begin{aligned} \Phi: \quad \mathbb{P} &\rightarrow \mathbb{R} \\ (p_1, \dots, p_k) &\mapsto \Phi(p_1, \dots, p_k) \end{aligned}$$

donde

$$\mathbb{P} = \{(p_1, \dots, p_k) \in [0; 1]^k \mid p_1 + \dots + p_k = 1\}$$

Requisitos:

- Alcanzar máximo en $(\frac{1}{k}, \dots, \frac{1}{k})$.
- Alcanzar mínimo en conjuntos homogéneos como $(0, \dots, 0, 1, 0, \dots, 0)$.
- Invariante frente a permutaciones de $(p_1, \dots, p_k)$.

- impureza del nodo t :

$$i(t) = \Phi(\Pr[1 \mid t], \dots, \Pr[k \mid t])$$

donde $\Pr[j \mid t]$ es la probabilidad de la clase j en el nodo t

- reducción de la impureza al dividir t en $\{t_L, t_R\}$

$$\Delta i(t \rightarrow t_L, t_R) = i(t) - p_L i(t_L) - p_R i(t_R)$$

donde

- p_L es la proporción de t que se va a t_L
- p_R es la proporción de t que se va a $t_R = 1 - p_L$

- impureza del árbol

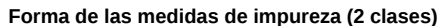
$$I(T) = \sum_{t \in T} \Pr(t) i(t)$$

- medidas habituales
 - entropía o información

$$- \sum_j p_j \log p_j$$

- Gini

$$\sum_j p_j(1 - p_j) = 1 - \sum_j p_j^2$$

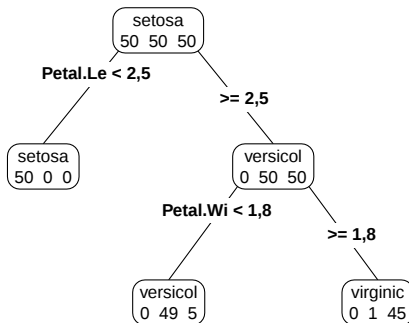


```
> ## por omisión se usa Gini
> a.gini <- rpart (Species ~
+                 Petal.Length + Petal.Width,
+                 iris,
+                 parms = list(split="gini"))
> a.info <- rpart (Species ~
+                 Petal.Length + Petal.Width,
+                 iris,
+                 parms = list(split="information"))
```

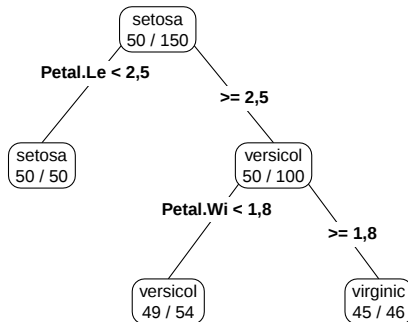
- riesgo de una hoja

$$R(t) = \Pr[\text{error clasif.} \mid t] = 1 - \max_{j \in \text{clases}} \Pr[j \mid t]$$

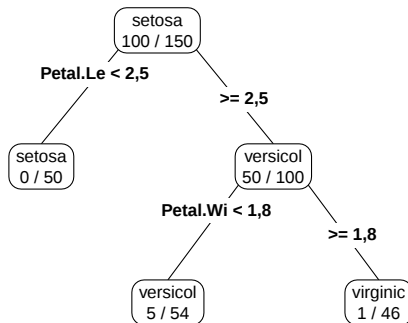
```
> prp (árbol, type=4, extra=1)
```



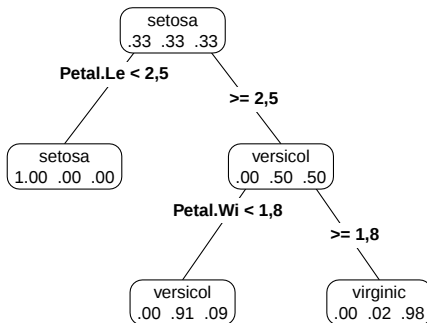
> prp (árbol, type=4, extra=2)



```
> prp (árbol, type=4, extra=3)
```



> prp (árbol, type=4, extra=4)



- riesgo de una hoja

$$R(t) = \Pr[\text{error clasif.} \mid t] = 1 - \max_{j \in \text{clases}} \Pr[j \mid t]$$

- riesgo del árbol

$$R(T) = \Pr[\text{error clasif.}] = \sum_{t \in T} R(t) \Pr[t]$$

- disminución del riesgo si $t \rightarrow \{t_L, t_R\}$

$$\Delta r = R(t) - R(t_L, t_R) > 0$$

- posible criterio de corte: maximizar Δr

¿Riesgo como criterio de corte?

Ejemplo

Generalidades

Árboles de
clasificación

Árboles de
regresión

Resumen

Ejercicios

Bibliografía

- ¿maximizar Δr ?
- supóngase:
 - 80 % de individuos de clase 1 en la raíz
 - cierto corte candidato llevaría a
 - 54 % de clase 1 en $t_L \Rightarrow$ asignar clase 1
 - 100 % de clase 1 en $t_R \Rightarrow$ asignar clase 1pero $\Delta r = 0$ aunque la bifurcación es muy informativa
- supónganse dos cortes candidatos:
 - corte A llevaría a 70 % y 70 % de clase 1
 - corte B llevaría a 85 % y 50 % de clase 1
 - el A tiene menos riesgo
 - el B es preferible en la práctica,
porque establece mejor cómo seguir dividiendo

Criterios de parada

Ejemplo

Generalidades

Árboles de
clasificación

Árboles de
regresión

Resumen

Ejercicios

Bibliografía

- minsplitt** Tamaño mínimo antes de cortar.
Por omisión, 20.
- minbucket** Tamaño mínimo de cada hoja. Por omisión, 7.
- cp** Parámetro de complejidad.
Coeficiente de penalización por número de
hojas.
Por omisión, 1 %.
- maxdepth** Máxima profundidad de las hojas
(0 = profundidad de la raíz).
Por omisión, 30.

```
> árbol$frame[,1:6]
```

	var	n	wt	dev	yval	complexity
1	Petal.Length	150	150	100	1	0,50
2	<leaf>	50	50	0	1	0,01
3	Petal.Width	100	100	50	2	0,44
6	<leaf>	54	54	5	2	0,00
7	<leaf>	46	46	1	3	0,01

```
> table (iris$Species,  
+       predict (árbol, iris, type="class"))
```

	setosa	versicolor	virginica
setosa	50	0	0
versicolor	0	49	1
virginica	0	5	45

```
> rpart (Species~Petal.Length+Petal.Width, iris,
+       control = rpart.control (cp = 0, minsplit = 2))
n= 150
```

```
node), split, n, loss, yval, (yprob)
```

```
* denotes terminal node
```

```
1) root 150 100 setosa (0,33333 0,33333 0,33333)
```

```
2) Petal.Length< 2.45 50 0 setosa (1,00000 0,00000 0,00000)
```

```
3) Petal.Length>=2.45 100 50 versicolor (0,00000 0,00000 0,00000)
```

```
6) Petal.Width< 1.75 54 5 versicolor (0,00000 0,00000 0,00000)
```

```
12) Petal.Length< 4.95 48 1 versicolor (0,00000 0,00000 0,00000)
```

```
24) Petal.Width< 1.65 47 0 versicolor (0,00000 0,00000 0,00000)
```

```
25) Petal.Width>=1.65 1 0 virginica (0,00000 0,00000 0,00000)
```

```
13) Petal.Length>=4.95 6 2 virginica (0,00000 0,00000 0,00000)
```

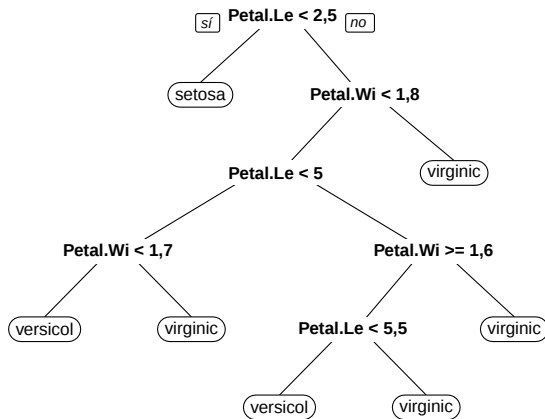
```
26) Petal.Width>=1.55 3 1 versicolor (0,00000 0,00000 0,00000)
```

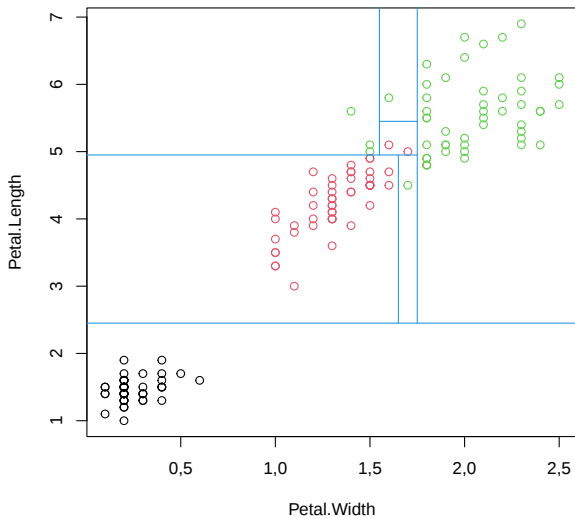
```
52) Petal.Length< 5.45 2 0 versicolor (0,00000 0,00000 0,00000)
```

```
53) Petal.Length>=5.45 1 0 virginica (0,00000 0,00000 0,00000)
```

```
27) Petal.Width< 1.55 3 0 virginica (0,00000 0,00000 0,00000)
```

```
7) Petal.Width>=1.75 46 1 virginica (0,00000 0,00000 0,00000)
```





Criterio de poda

- T_∞ árbol sin ramas, sólo raíz
- $\alpha > 0$ parámetro de complejidad
- $R_\alpha(T) = R(T) + \alpha \cdot |T| \cdot R(T_\infty)$
- T_α único árbol que minimiza R_α
- T_0 árbol completo

```
> printcp (árbol)                                # árbol$cptable
```

Classification tree:

```
rpart(formula = Species ~ Petal.Length + Petal.Width,
```

Variables actually used in tree construction:

```
[1] Petal.Length Petal.Width
```

Root node error: $100/150 = 0,67$

n= 150

	CP	nsplit	rel error	xerror	xstd
1	0,50	0	1,00	1,16	0,051
2	0,44	1	0,50	0,58	0,060
3	0,01	2	0,06	0,07	0,026

Validación cruzada

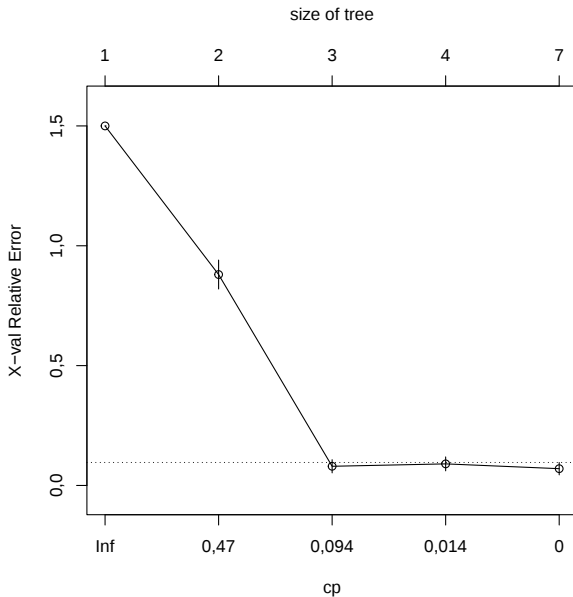
Cambiar número de cruzvalidaciones
para obtener resultados más estables:

```
> a100 <- rpart (Species~Petal.Length+Petal.Width,  
+               iris,  
+               control=rpart.control(cp=0,  
+                                     minsplit=2,  
+                                     xval=100))
```

```
> a100$cptable
```

	CP	nsplit	rel error	xerror	xstd
1	0,50	0	1,00	1,50	0,00000
2	0,44	1	0,50	0,88	0,06031
3	0,02	2	0,06	0,08	0,02752
4	0,01	3	0,04	0,09	0,02909
5	0,00	6	0,01	0,07	0,02583

`> plotcp (a100) # abscisas = medias geométricas`



```
> prune (a100, cp=0.09) # cualq. entre 0,02 y 0,44
n= 150
```

```
node), split, n, loss, yval, (yprob)
      * denotes terminal node
```

```
1) root 150 100 setosa (0,33333 0,33333 0,33333)
  2) Petal.Length< 2.45 50    0 setosa (1,00000 0,00000 0,00000)
  3) Petal.Length>=2.45 100  50 versicolor (0,00000 0,00000 0,00000)
    6) Petal.Width< 1.75 54    5 versicolor (0,00000 0,00000 0,00000)
    7) Petal.Width>=1.75 46    1 virginica (0,00000 0,00000 0,00000)
```

Importancia de las variables

Ejemplo

Generalidades

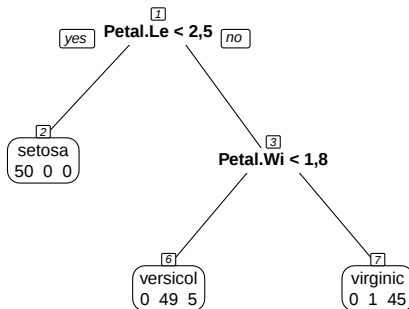
Árboles de
clasificación

Árboles de
regresión

Resumen

Ejercicios

Bibliografía



Importancia de las variables

Ejemplo

Generalidades

Árboles de
clasificación

Árboles de
regresión

Resumen

Ejercicios

Bibliografía

- suma de calidad de la división en nodos cuyas reglas involucran la variable
- si dos variables están muy correladas, su importancia conjunta se dividirá entre las dos y puede que ninguna destaque
- como componente `variable.importance` del objeto `rpart` aparece en forma absoluta

```
> árbol $ variable.importance
      Petal.Width Petal.Length
           88,97          81,34
```
- en la salida de `summary` aparece en porcentajes

Importancia de las variables

```
> summary (árbol)
```

Call:

```
rpart(formula = Species ~ Petal.Length + Petal.Width,  
      n= 150
```

	CP	nsplit	rel error	xerror	xstd
1	0,50	0	1,00	1,16	0,05128
2	0,44	1	0,50	0,58	0,05964
3	0,01	2	0,06	0,07	0,02583

Variable importance

Petal.Width	Petal.Length
52	48

```
Node number 1: 150 observations,      complexity param=
predicted class=setosa      expected loss=0.6667  P
class counts:      50      50      50
probabilities: 0.333 0.333 0.333
```

- Y cuantitativa
- rpart usa el método anova
- predicción $\hat{y}_t$: al nodo t se le asigna $\bar{y} = \frac{1}{n_t} \sum_{i \in t} y_i$
(en clasificación asignábamos la moda)
- criterio: maximizar $SC_t - SC_L - SC_R$ donde
 - SC_t = suma de cuadrados $\sum (y_i - \bar{y})^2$ en el nodo t
 - SC_L = ídem en su nodo hijo L
 - SC_R = ídem en su nodo hijo R
- riesgo en el nodo t : varianza $\frac{1}{n_t} \sum_{i \in t} (y_i - \hat{y}_t)^2$
(en clasificación era la tasa de incorrectos)

Protocolo para construir un árbol

- 1 Determinar el número adecuado de permutaciones para validación cruzada: ¿ $xval = n, 10 \dots$?
- 2 Crear un árbol completo
 - $cp = 0$
 - $minsplit = 2$ ó $minbucket = 1$
 - ¿ $xval$?
- 3 Fijarse en si $xstd$ (errores típicos obtenidos por cruzvalidación) son reducidos. Si no, aumentar $xval$
- 4 Buscar el parámetro de complejidad (cp) adecuado
`plotcp (árbol)                      # o printcp(árbol)`
- 5 Podar el árbol (o recalcularlo con cp)
`arbol <- prune (árbol, cp=...)`
`rpart (... , control=rpart.control(cp=...))`

Ejercicio 1: iris

- Construye un árbol de clasificación para los datos `iris` a partir de todas sus variables (en la presentación usamos sólo las de los pétalos).
- ¿Se gana algo respecto a usar sólo los pétalos?
- Compáralo con el análisis discriminante.

Ejercicio 2: mtcars

- Construir un modelo de regresión lineal para estimar mpg a partir del resto de variables.
- Construir un árbol de regresión con el mismo objetivo.
- Comparar ambos modelos:
 - Con la información que producen las funciones de R.
 - Con el método de retención (entrenamiento y validación).
 - Con validación cruzada.

Ejercicio 3: solder

- Construye un árbol a partir de los datos `solder.balance` para buscar variables que afecten al número de proyecciones (`skips`).

Ejercicio 4: genotipos

- Construye un árbol para los datos `http://bellman.ciencias.uniovi.es/~carleos/master/manadine/curso1/AnalisisDatos1/3-arboles/dat/geno.csv` para predecir genotipo.

Más detalles

- https://es.wikipedia.org/wiki/Aprendizaje_basado_en_%C3%A1rboles_de_decisi%C3%B3n
- <https://cran.r-project.org/web/packages/rpart/vignettes/longintro.pdf>
- https://www.researchgate.net/publication/263671703_Fifty_Years_of_Classification_and_Regression_Trees