ÁRBOLES DE DECISIÓN: EL MODELO C4.5

APRENDIZAJE AUTOMÁTICO SUPERVISADO

Guillermo de Jorge Botana

Dpto. Psicobiología y Metodología en Ciencias del Comportamiento

Facultad de Psicología. Universidad Complutense de Madrid.

Tabla de contenido

NOTA:

El contenido de este texto corresponde a uno de los temas de la asignatura del Máster de Metodología de las Ciencias del Comportamiento y de la Salud y también de la asignatura de Tecnología del Conocimiento del Grado de Psicología. Está elaborado para tener un texto base de lo que es explicado en clase. Aunque el texto es seguido y coherente, puede ser susceptible de algunas mejoras y ampliaciones. No obstante, es lo suficientemente autocontenido para llevar a cabo un estudio independiente sobre él.

He decidido publicar este texto fuera del ámbito de la asignatura por si puede resultar de utilidad para otros estudiantes o por si a otros docentes les puede facilitar la tarea.

Introducción

Intuimos fácilmente lo que es un Árbol de Decisión. Nos imaginamos un conjunto de nodos y ramas por las que vamos decidiendo en base a preguntas o acciones que se nos plantean. Si tenemos que clasificar un evento, un conjunto de preguntas nos saldrán al paso según discurrimos por las ramas y las respuestas a ellas nos harán decantarnos por unas nuevas ramas y desdeñar otras. Además, podemos también intuir que un árbol de este tipo puede valer para predecir eventos futuros en base a las probabilidades de lo que ya ha pasado. Si queremos clasificar a las personas como buenos pagadores y obra en nuestro poder un histórico de las características de otras personas que pagaron en el pasado o bien dejaron a deber, podemos emplear éste para generar un árbol de decisión con preguntas relativas a esas características. Por ejemplo, guiados por el histórico podríamos preguntar: ¿casado?, ¿trabajo fijo?, ¿mayor de 35 años?, ¿deudas anteriores?, etc. Una buena selección y ordenación de estas preguntas harán que ciertas decisiones sean fáciles de tomar pues hechas algunas, la probabilidad de pago o deuda será significativamente diferente. Imagínese el lector la cara del banquero si después de hacer dos de estas preguntas el árbol le identifica que la probabilidad de pago es del 97%. Es justo esto lo que buscan las técnicas que generan árboles de decisión: que las decisiones sean sin incertidumbre.

En los procesos de clasificación basados en árboles se ponen en juego una secuencia de preguntas sobre las características de los eventos o ejemplares y las respuestas a ellas van acotando el espacio de búsqueda. El espacio de búsqueda tiene a su vez estados que se diferencian en cuanto a la utilidad de la información recopilada. Habrá estados en los que la información sea incierta y no permita decantarse por una clasificación clara. Otros estados sí nos permitirán clasificar con un bajo riesgo de equivocarse. Lo deseable es pues llegar a esos estados de poca incertidumbre en el menor número de preguntas.

Si queremos clasificar los animales en reptiles o no reptiles habrá preguntas que nos dejen en una situación incierta para clasificar y otras que nos instigarán a dar una clasificación con un grado de acierto considerablemente alto. La pregunta que indague si un ejemplar tiene patas no nos proporcionará certidumbre. Los habrá con o sin patas, pongamos que en la misma proporción. Tampoco el saber si es vertebrado nos saca de la incertidumbre. Para clasificar de manera rápida (y sin entretenerse en preguntas) un animal en la clase de reptiles es quizás más útil indagar sobre

la distancia entre su vientre y el suelo. Pongamos esa pregunta la primera y llegaremos pronto y bien a la clasificación. Esa primera pregunta nos traslada a un estado en el que en algunas ramas ya estemos “casi” seguros (acaso seguros) de que vamos a dar la respuesta correcta.

Esta es la clave de los sistemas destinados a generar árboles de decisión. No vale cualquier árbol. No todos son iguales. Valdrá el que situé en los primeros niveles del árbol la indagación de las propiedades que nos ayuden a retirar ambigüedad, que nos dejen en estados en los que ya podemos decantarnos sin mucho miedo al fracaso en la clasificación, a decir que muy probablemente pagará o que muy probablemente es un reptil.

¿Con que motivación se crearon estos generadores de árboles de decisión? Básicamente la idea era generar el árbol de decisión menos profundo, con el menor número de preguntas, y lo más generalizable a eventos o ejemplares que no han participado en su generación. Una primera tentativa sería hacer todos los árboles posibles dadas unas variables y sus propiedades para luego elegir el que concilia esas virtudes. El problema es que eso tiene un coste computacional gigante, a veces intratable, debido a la combinatoria. Incluso un ejemplo sencillo de unas pocas variables con unos pocos niveles ya despliega una combinatoria complicada.

Debido a esto, los algoritmos generadores de árboles de decisión suelen pertenecer a la familia de los llamados “algoritmos voraces” (en inglés, “greedy” algorithms). Estos algoritmos destacan por su frugalidad computacional, pues nunca vuelven sobre sus pasos. Una vez tomada una decisión, una vez propuesta la indagación sobre una propiedad, no se arrepiente y sigue para adelante en el proceso. Y seguir para adelante es proponer los siguientes niveles del árbol indagando sobre otras propiedades.

Como se ve, es un algoritmo que tienen que ahorrar balas, que tiene que pensar mucho a la hora de proponer las propiedades a indagar en los sucesivos niveles. Y ahí se justifica el criterio por el cual se guía. Elegirá para la indagación en cada nivel que va construyendo aquellas variables cuyas propiedades particionen el espacio de búsqueda de manera que los estados generados por cada partición contengan menos incertidumbre que los estados anteriores. La cosa es clara. El algoritmo va partiendo el espacio en forma de preguntas de manera que el estado generado por cada partición nos ayuda a decantarnos para clasificar el evento o ejemplar de manera más probable. Las preguntas serán del tipo “¿el día es frío, normal o caluroso?” (tres particiones), “el

día es húmedo o seco" (dos particiones), "el día es ventoso o normal" (dos particiones). La colocación "inteligente" de estas preguntas en un árbol (en una secuencia con ramificaciones) nos ayudarán por ejemplo a clasificar dicho día en "proclive a la pesca". Pero claro, insistimos, todo lo descrito se puede llevar a cabo en la medida de que se tenga un histórico por el que se conocen las propiedades de los eventos o ejemplares y el efecto que tuvieron en otra variable a predecir: cómo eran los días y si hubo buena pesca. Conocemos el pasado y queremos crear un árbol de decisión con él. Pero este árbol ha de ser pequeño y eficiente.

En general, la tarea que llevan a cabo los generadores de árboles de decisión se puede formular con algunas preguntas. Dado un conjunto de variables Independientes o Predictoras y sus propiedades, ¿Cómo podemos hacer un árbol de decisión lo más pequeño posible con la máxima efectividad para predecir la Variable Dependiente?, o de manera más precisa, ¿Qué secuencia ha de llevar el particionado del espacio de búsqueda para generar el árbol más pequeño y generalizable? Vamos a ver en detalle como lo resuelve uno de los algoritmos de generación de árboles de decisión más popularizados: el algoritmo C4.5 de Quinlan.

Conceptos clave

Se ha aludido en la introducción que para la aplicación de los algoritmos se ha de contar con un histórico. Estamos ya acostumbrados a ver la estructura de los históricos en modelos como la regresión lineal. Su formato no es más que una serie de variables predictoras (llamadas independientes en este texto) y una variable criterio a predecir (llamadas dependientes en este texto). Pues bien, todas esas variables suelen representarse en una tabla en forma de columnas. La tabla 1 muestra unos datos que serán utilizados intensivamente a lo largo de ese texto.

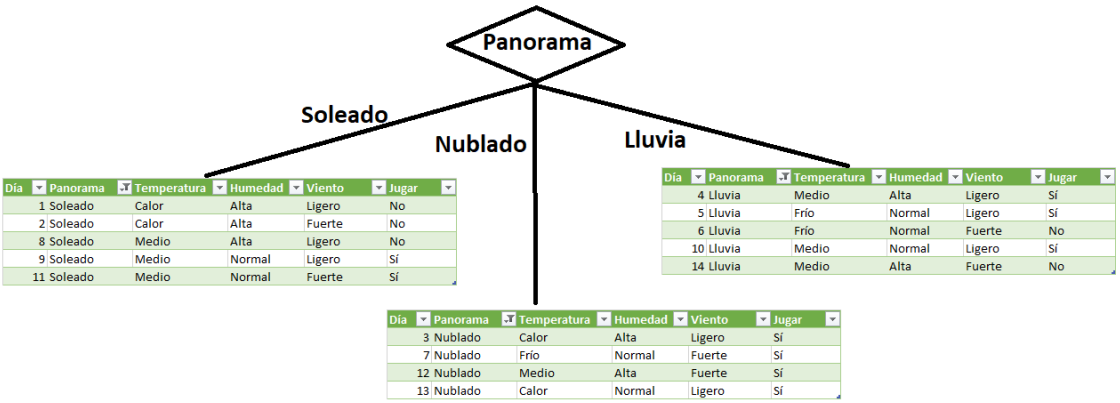
Día	Panorama	Temperatura	Humedad	Viento	Se jugó
1	Soleado	Calor	Alta	Ligero	No
2	Soleado	Calor	Alta	Fuerte	No
3	Nublado	Calor	Alta	Ligero	Sí
4	Lluvia	Medio	Alta	Ligero	Sí
5	Lluvia	Frío	Normal	Ligero	Sí
6	Lluvia	Frío	Normal	Fuerte	No
7	Nublado	Frío	Normal	Fuerte	Sí
8	Soleado	Medio	Alta	Ligero	No
9	Soleado	Medio	Normal	Ligero	Sí
10	Lluvia	Medio	Normal	Ligero	Sí
11	Soleado	Medio	Normal	Fuerte	Sí
12	Nublado	Medio	Alta	Fuerte	Sí
13	Nublado	Calor	Normal	Ligero	Sí
14	Lluvia	Medio	Alta	Fuerte	No

Tabla 1.

Para comprender cómo el algoritmo genera el árbol de esa tabla será importante manejar el argot para interpretar las fórmulas que nos servirán para calcular las incertidumbres en los estados generados por las particiones. Vamos a describir a continuación cada tipo de variable, sus particularidades y el concepto de Ganancia de Información.

Variables Independientes

Cada una de las variables que definen un evento o un ejemplar. Estas variables tienen **propiedades**. Estas propiedades son los que definirán los nodos de decisión y partirán la muestra en submuestras (en subtablas, si se quiere). De ahí el concepto de **partición**. En cada nivel de profundidad del árbol se realizarán tentativas de particiones a partir de estas propiedades y se medirá qué variable nos proporciona una partición que nos lleva a un estado de menor incertidumbre para predecir. Esto identificará la mejor partición para el siguiente nivel. En la tabla 1, Panorama es una Variable Independiente que tiene tres propiedades {Soleado, Nublado, Lluvia}. Si en el primer nivel del árbol esta partición resultase la que más incertidumbre retirase a la distribución de las predicciones sobre Jugar, y generásemos ya el árbol con esta decisión, dicho árbol tendría el aspecto de la figura 1.



Día	Panorama	Temperatura	Humedad	Viento	Jugar
1	Soleado	Calor	Alta	Ligero	No
2	Soleado	Calor	Alta	Fuerte	No
8	Soleado	Medio	Alta	Ligero	No
9	Soleado	Medio	Normal	Ligero	Sí
11	Soleado	Medio	Normal	Fuerte	Sí

Día	Panorama	Temperatura	Humedad	Viento	Jugar
4	Lluvia	Medio	Alta	Ligero	Sí
5	Lluvia	Frío	Normal	Ligero	Sí
6	Lluvia	Frío	Normal	Fuerte	No
10	Lluvia	Medio	Normal	Ligero	Sí
14	Lluvia	Medio	Alta	Fuerte	No

Día	Panorama	Temperatura	Humedad	Viento	Jugar
3	Nublado	Calor	Alta	Ligero	Sí
7	Nublado	Frío	Normal	Fuerte	Sí
12	Nublado	Medio	Alta	Fuerte	Sí
13	Nublado	Calor	Normal	Ligero	Sí

Figura 1

Como se ve en la figura 1, al realizar la partición por las tres propiedades de Panorama, hemos conseguido que el nodo que abre la rama de “Nublado” haya eliminado completamente la ambigüedad. Si nos fijamos, la distribución de la VD Jugar en dicho nodo solo posee la clase “Sí”. En otras palabras, si al comenzar el árbol de decisión sobre un ejemplar futuro se identificase que el Panorama es “nublado”, la predicción sería “Sí” con probabilidad 1. El propio generador del árbol ha detectado sobre el histórico una estructura en la que la propiedad “Nublado” de Panorama nos llevaría a predecir “Sí”. Esta es la clase de hallazgos que el generador quiere detectar. Estructuras con estados poco ambiguos. A partir de aquí, el generador seguiría calculando cuales son las mejores particiones para hacer colgar sobre los otros dos nodos “Soleado” y “Lluvia”. Ahora bien, el nodo abierto por nublado quedaría ya cerrado, pasando de ser un nodo de decisión a una hoja. Lo veremos con más detalle.

Variable Dependiente

Se trata de la variable a predecir, en este caso “Jugar”. Los estados que toma esta variable y que serán predicho se llaman **clases**. En la tabla 1 “Jugar” tiene clases dicotómicas, pues son un conjunto que representa la afirmación o la negación {Sí, no}. Como se ha anticipado, la distribución de la Variable Independiente es sobre la que se calculará el nivel de incertidumbre (mediante un índice de entropía que veremos). Por ejemplo, el nodo formado por la partición de “Nublado” sigue teniendo puntuaciones de la Variable Dependiente, en este caso “Jugar”. Midiendo la Entropía de su distribución obtenemos una Entropía de 0, pues todos los valores son “Sí”. Es decir, una entropía nula nos desvela una incertidumbre Nula, y una muy buena posición

para decidir “Sí” en el futuro. En general, todos los pasos que da el algoritmo para decidir las particiones los hace guiado por el grado de incertidumbre de las tentativas de partición. Para dilucidar que partición tendríamos que colocar debajo del nodo formado por “Nublado” en la figura 1, se tendrían que hacer tentativas de particiones con las Variables Independientes restantes, es decir, con Temperatura, Humedad y Viento, y medir la entropía de la distribución de la Variable Dependiente “Jugar” en cada una de las particiones de cada Variable. Un compendio de la entropía de cada propiedad en cada Variable Dependiente se contrastaría con la entropía que existe sin partición alguna, es decir, sin haber materializado ninguna de las tentativas. Esa es la dinámica que estamos anticipando y que veremos con más detalle en un ejemplo posterior.

Ganancia de Información

Un concepto que ya ha sido anticipado es el de incertidumbre. El algoritmo trata de evitar construir un árbol con estados inciertos. Por ello se colocan con preeminencia las Variables Independientes que retiren esa incertidumbre. Hemos dicho también, acaso sin mucho énfasis aún, que la incertidumbre de un estado se mide calculando la Entropía de la distribución de la Variable Dependiente en cada tentativa de partición. Pues bien, toca ahora definir un criterio con el cuál podamos decir que tal o cual Variable genera particiones menos inciertas, menos entrópicas. Y esto, como también se anticipó, se hace comparando la entropía obtenida después de realizada una tentativa de partición con la entropía obtenida en ausencia de partición. A esto justo nos referimos con el concepto o el índice de Ganancia de Información.

Poner a prueba una tentativa es poner a prueba el efecto de una Variable Independiente sobre la distribución de la Variable Dependiente. Esto significa hacer la partición con esa Variable Independiente y calcular las entropías de cada una de las ramas formadas. La Ganancia de Información o $IG(S)$ (fórmula 1) es una resta entre la Entropía que había en la distribución de la VD sin esa partición $I(S)$ menos la suma de las entropías ponderadas de todas las ramas $IP(S)$. Dicho en román paladino, la entropía sin partición menos la entropía conseguida por las ramas en conjunto.

En concreto, la fórmula es esta:

$$IG(S) = I(S) - IP(S) \qquad \text{(fórmula 1)}$$

Siendo $I(S)$ la entropía en ausencia de partición y $IP(S)$ una integración ponderada de las entropías obtenidas en cada rama por separado. En concreto:

$$I(S) = Entropía(S) \quad \text{(fórmula 2)}$$

Donde S es la distribución entera (sin partición) de la variable Dependiente.

Por otra parte, $IP(S)$ se expresaría:

$$IP(S) = \sum_i P_i \, Entropía(S)_i \quad \text{(fórmula 3)}$$

Donde

S_i la distribución de la Variable Dependiente en la partición i.

P_i es la proporción de ejemplares que permanecen en la partición i frente al total no particionado. Por ejemplificar, la P_i de la participación de "Soleado" de la figura 1 sería 5/14, pues los 5 ejemplares que se obtienen haciendo la partición supondrían un 5/14 del total en ausencia de partición. $P_{Soleado} = \frac{5}{14}$ por tanto.

La lectura en lenguaje natural de la fórmula 3 sería que se trata de una suma ponderada de las entropías de las distribuciones de la VD en cada partición. Es decir, una medida de incertidumbre global si se llevase a cabo esa partición.

A su vez, ambas entropías, la de las distribuciones sin partición y la de las distribuciones de cada partición, se calcularían con la fórmula clásica de la Entropía:

$$Entropía(S) = \sum_j P_j \log_2(P_j)$$ (fórmula 4)

Donde

P_j es la proporción de cada clase j en la Variable Dependiente.

En la figura 1, si nos fijamos en la partición hecha a partir de la propiedad "Soleado", tendremos que hay dos clases {Sí, No} y que $P_{Sí} = \frac{2}{5}$ y $P_{No} = \frac{3}{5}$. Por tanto, si quisiésemos calcular la entropía de la distribución de la VD en dicha partición, desenrollando el sumatorio obtendríamos:

$$Entropía(S_{Soleado}) = \frac{2}{5}\log_2\left(\frac{2}{5}\right) + \frac{3}{5}\log_2\left(\frac{3}{5}\right)$$

Como se intuye, la ganancia de Información es clave. Es el criterio que regirá el algoritmo. Y es de mayor importancia si se piensa en que un algoritmo voraz como éste no volverá atrás. Las decisiones quedarán grabadas en mármol. No hay arrepentimiento. Por tanto, lo que hará es colocar en los siguientes niveles del árbol (incluida la raíz) aquella variable cuya partición proporcione más Ganancia de Información. Y esa Ganancia de Información será calculada con la fórmula 1. Esto se hará recurrentemente en cada nodo hasta conseguir un buen candidato de árbol, es decir, pequeño y generalizable. Vamos a presentar un caso práctico que ayude a comprender la foto total con el uso de las fórmulas.

Caso práctico

Construyamos un árbol de decisión en el ejemplo de la tabla 1. Partimos de un nodo raíz del que aún no cuelgan otros nodos ni hojas. El reto es hacer colgar de ese nodo raíz los nodos de la partición que consiga mayor Ganancia de Información, es decir, que minimice la entropía de las distribuciones de la VD tomando como referencia la Entropía en ausencia de partición. Esto se hace poniendo a prueba cada Variable Independiente y su aporte en la Ganancia de Información. Nos referiremos a estas pruebas con el término "tentativas". En cada tentativa comprobaremos mediante los cálculos de $IG(S)$ cuál es la Ganancia de Información con la partición a partir de cada Variable Independiente. La Variable con mayor ganancia será la candidata para continuar el árbol o pender de la raíz en caso de inicio.

Primer nivel del árbol

Tentativa de partición: Panorama

Empezamos calculando la Ganancia de Información al particionar por la Variable "Panorama". Recordemos de la fórmula 3 que se hacía mediante la suma ponderada de las entropías de las distribuciones de la VD en cada partición. Es decir, calcular las entropías de cada partición por separado y compendiarlas ponderadas mediante el sumatorio. De esa manera la entropía de la partición se calculaba formalmente:

$$IP(S) = \sum_i P_i \, Entropía(S)_i \quad \text{(fórmula 5)}$$

Para mayor comodidad y didactismo, vamos a asignar a los valores numéricos del índice de las propiedades de la partición (a la i) las etiquetas que les corresponderían. La correspondencia sería {1= "soleado", 2 = "nublado", 3 = "lluvia"}. Hecho esto, el sumatorio de la fórmula de $IP(S)$ se desenrollaría como sigue:

$$IP(S) = P_{Soleado} \, Entropía(S)_{Soleado} + P_{Nublado} \, Entropía(S)_{Nublado} + P_{Lluvia} \, Entropía(S)_{lluvia}$$

O expresado de una manera más gráfica:

$$IP(S) = P_{Soleado}\, Entropía(\begin{matrix}\text{No}\\\text{No}\\\text{No}\\\text{Sí}\\\text{Sí}\end{matrix}) + P_{Nublado}\, Entropía(\begin{matrix}\text{Sí}\\\text{Sí}\\\text{Sí}\\\text{Sí}\end{matrix}) + P_{Lluvia}\, Entropía(\begin{matrix}\text{Sí}\\\text{Sí}\\\text{No}\\\text{Sí}\\\text{No}\end{matrix})_{lluvia}$$

Al desenrollar la fórmula nos quedamos más tranquilos. Los términos son más familiares y sobre todo, sustituibles. La tarea sería calcular las entropías de cada propiedad de la partición $Entropía(S)_{Soleado}$, $Entropía(S)_{Nublado}$ y $Entropía(S)_{lluvia}$ además de la proporción de tamaño de la partición de cada partición frente al total $P_{Soleado}$, $P_{Nublado}$ y P_{Lluvia}.

La entropía de “Soleado” se harían mediante:

$$Entropía(S)_{soleado} = \sum_{j} P_j \log_2(P_j)$$

Para hacer esto, haremos con las clases de la VD “Jugar” lo mismo que con las propiedades de las Variables Independientes. Los valores numéricos del índice j que representa las clases se corresponderían en el siguiente conjunto de relaciones j = {1 = “sí”, 2= “no”}. Esto no es más que una forma de naturalizar los índices y acercarlos a nuestro problema. Desenrollando el sumatorio y asignadas ya las etiquetas a j , la fórmula quedaría más naturalizada:

$$Entropía(S)_{soleado} = Entropía(\begin{matrix}\text{No}\\\text{No}\\\text{No}\\\text{Sí}\\\text{Sí}\end{matrix}) = P_{Sí} \log_2(P_{Sí}) + P_{No} \log_2(P_{No})$$

$$Entropía(S)_{soleado} = \frac{2}{5}\log_2\left(\frac{2}{5}\right) + \frac{3}{5}\log_2\left(\frac{3}{5}\right) = 0{,}97$$

Esta sería la entropía de la VD “Jugar” en la partición de Soleado de la variable Panorama. Queda calcular la entropía de la VD “Jugar” en las otras dos particiones de Panorama.

La entropía de la VD en la partición “nublado” sería:

$$Entropía(S)_{Nublado} = Entropía(\boxed{\begin{matrix}\text{Sí}\\ \text{Sí}\\ \text{Sí}\\ \text{Sí}\end{matrix}}) = \frac{5}{5}\log_2\left(\frac{5}{5}\right) + \frac{0}{5}\log_2\left(\frac{0}{5}\right) = 0$$

La entropía es cero porque en la situación de nublado siempre jugar es “sí” y por ello no hay entropía, todo está claro.

La entropía de la VD en la partición “Lluvia” sería:

$$Entropía(S)_{Lluvia} = Entropía(\boxed{\begin{matrix}\text{Sí}\\ \text{Sí}\\ \text{No}\\ \text{Sí}\\ \text{No}\end{matrix}}) = \frac{3}{5}\log_2\left(\frac{3}{5}\right) + \frac{2}{5}\log_2\left(\frac{2}{5}\right) = 0{,}97$$

Calculemos también las proporciones P_i adecuadas de la fórmula 3:

$$P_{Soleado} = \frac{5}{14} = 0{,}36$$

$$P_{Nublado} = 0{,}28$$

$$P_{Lluvia} = \ 0{,}36$$

Y recompongamos entonces la fórmula de $IP(S)$ ya con todos los sumandos y sus términos:

$$IP(S) = (0{,}36\ x\ 0{,}97) + (0{,}28\ x\ 0) + (0{,}36\ x\ 0{,}97) = 0{,}70$$

Respecto a la fórmula $I(S)$ que calcula la entropía total de la distribución de la VD "Jugar" en ausencia de partición la desarrollamos de esta forma (se toman los valores de "Jugar" de la tabla entera):

$$I(S) = Entropía(S) = Entropía($$

Se jugó
No
No
Sí
Sí
Sí
No
Sí
No
Sí
Sí
Sí
Sí
Sí
No

$$)$$

$$I(S) = \sum_j P_j \log_2 P_j$$

$$I(S) = P_{Sí} \log_2(P_{Sí}) + P_{No} \log_2(P_{No})$$

$$I(S) = 0{,}64 \log_2(0{,}64) + 0{,}36 \log_2(0{,}36) = 0{,}94$$

Recuperando la fórmula 1 obtendríamos la ganancia de Información con la partición hecha con "Panorama" y sus tres propiedades. La fórmula es una simple resta entre la Entropía que se obtiene sin partición y la que se obtiene con la partición:

$$IG(S) = I(S) - IP(S)$$

$$IG(S) = 0{,}94 - 0{,}70 = 0{,}24$$

Tentativa de partición: Temperatura

Veamos ahora la variable "temperatura" y la ganancia de Información que se obtendría con su partición. Recordemos que la Entropía en ausencia de partición, la $I(S)$, está ya calculada, por lo que nos centraremos en la $IP(S)$, la que concierne a la partición:

$$IP(S) = \sum_i P_i \, Entropía(S)_i$$

El desarrollo de esta fórmula nos vuelve a ser familiar. Se trata de calcular la entropía de la distribución de la VD pero desglosada por cada partición. Además, la entropía de cada partición será ponderada por

una proporción sensible al tamaño de cada partición frente al total. Desenrollando el sumatorio y asignando los índices a los distintos niveles de Temperatura tendríamos:

$$IP(S) = P_{Calor}\, Entropía(S)_{Calor} + P_{Medio}\, Entropía(S)_{Medio} + P_{Frío}\, Entropía(S)_{Frío}$$

La anterior expresión ya nos guía hacia el cálculo de la entropía de cada partición:

$$Entropía(S)_{calor} = \sum_{j} P_j \log_2 P_j$$

Donde j representa las clases de la Variable Dependiente Jugar. Desenrollando el sumatorio y siendo en nuestro caso las clases el conjunto {"Sí", "No"}, para mayor sencillez podemos expresar la Entropía de la partición "calor" como:

$$Entropía(S)_{Calor} = P_{Sí} \log_2(P_{Sí}) + P_{No} \log_2(P_{No})$$

Y sustituyendo valores:

$$Entropía(S)_{Calor} = 0{,}5 \log_2(0{,}5) + 0{,}5 \log_2(0{,}5) = 1$$

Hacemos lo mismo para los otros dos niveles de "Temperatura", en primer lugar, su propiedad Medio:

$$Entropía(S)_{Medio} = P_{Sí} \log_2(P_{Sí}) + P_{No} \log_2(P_{No})$$

$$Entropía(S)_{Medio} = 0{,}71 \log_2(0{,}71) + 0{,}29 \log_2(0{,}29) = 0{,}87$$

Y luego “Frío”

$$Entropía(S)_{Frío} = P_{Sí}\log_2(P_{Sí}) + P_{No}\log_2(P_{No})$$

$$Entropía(S)_{Frío} = 0{,}66\log_2(0{,}66) + 0{,}34\log_2(0{,}34) = 0{,}92$$

Así, la expresión $IP(S)$ quedaría:

$$IP(S) = 0{,}28\ x\ 1 + 0{,}5\ x\ 0{,}87 + 0{,}22\ x\ 0{,}92 = 0{,}92$$

Ahora ya estamos en disposición de comprobar la Ganancia de Información obtenida con la partición de Temperatura. Para ello volvemos a aplicar:

$$IG(S) = I(S) - IP(S)$$

$$IG(S) = 0{,}94 - 0{,}92 = 0{,}025$$

De momento, ya se comprueba que la Ganancia de Información obtenida con “Panorama” es mayor que la Ganancia de Información de la partición de “Temperatura”. Un $0{,}24$ frente a un $0{,}025$. Por tanto, Temperatura queda descartada para esta primera partición.

Tentativas restantes

Haciendo los mismos cálculos sobre “Humedad” las Ganancia de Información quedaría:

$$IG(S) = 0{,}94 - 0{,}74 = 0{,}20$$

Y para Viento:

$$IG(S) = 0{,}94 - 0{,}90 = 0{,}04$$

Resultado en el primer nivel

Por tanto, se ganará más información con la partición de “Panorama”, y así pues, se procede a colgar del nodo raíz tres ramas que contienen los tres nodos de Panorama {Soleado, Nublado, Lluvia}

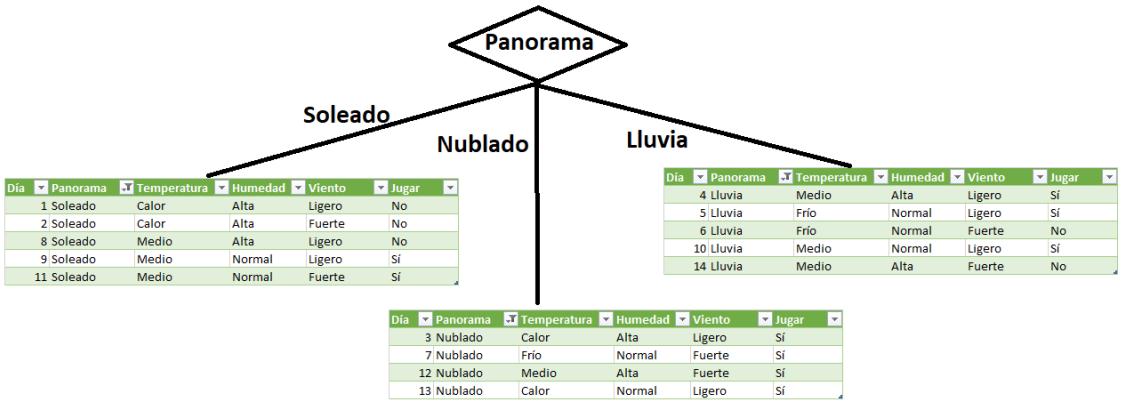


Figura 2

A poco observar nos damos cuenta de que ya hay una rama en la que la decisión es clara. La rama “nublado”, con su nodo, predice siempre jugar. No hay incertidumbre. Esto es justo el objetivo de la técnica. Hacer los nodos cada vez menos inciertos y más claros. De hecho, el nodo de “Nublado” pasará a ser una hoja, en el sentido que no colgarán más ramas de él. No participará más en la decisión pues la decisión, en ese punto, está completamente tomada.

Pero aquí no acaba la generación del árbol. Habrá que seguir desarrollando los nodos de decisión hasta que encontremos situaciones menos inciertas en las demás ramas. En el nodo de “lluvia” y “Soleado” se tiene aún probabilidades parejas de decantarse por el “Sí” o por el “No”, existe todavía encrucijada, existe perplejidad. En concreto, si se emitiese “Sí” en el nodo de “Lluvia” el sistema acertaría en solo el 60%. Es una apuesta muy fuerte comparada con el 100% del nodo “Nublado”. El mismo 60% de aciertos al pronosticar “No” en el nodo de “Soleado”. Esto indica que hay que seguir expandiendo el árbol en busca de una menor incertidumbre. Y en cada nodo de decisión habrá que seguir el mismo procedimiento seguido en la raíz. Estamos ya ante la parte recurrente del algoritmo.

Segundo nivel del árbol

Rama de Soleado

El árbol ya está en construcción. La figura 2 ya lo muestra con sus primeras ramas y nodos. Es más, una de las ramas se ha tornado en hoja al hacerse imposible encontrar mayor certidumbre que la ofrecida en su rama. No obstante, como decíamos, las otras dos ramas aun adolecen de incertidumbre. Es el momento de seguir con el procedimiento y buscar las mejores particiones para ambas ramas. Empecemos por la rama de la propiedad "Soleado". Se presenta su nodo en la figura 3. Lo que se ha de hacer es poner a prueba la Ganancia de Información si se realizase una partición con cada una de las Variables Independientes restantes, a saber, "Temperatura", "Humedad" y "Viento". Si antes llamamos a estas pruebas tentativas, procedamos pues con las tres tentativas y obtengamos la ganadora.

Día	Panorama	Temperatura	Humedad	Viento	Jugar
1	Soleado	Calor	Alta	Ligero	No
2	Soleado	Calor	Alta	Fuerte	No
8	Soleado	Medio	Alta	Ligero	No
9	Soleado	Medio	Normal	Ligero	Sí
11	Soleado	Medio	Normal	Fuerte	Sí

Figura 3

Tomemos "Temperatura" en primer lugar. Calculemos la entropía de la distribución sin partición para más tarde calcular la entropía con las particiones. De su contraste surgirá la ganancia. Respecto a la Entropía sin partición:

$$I(S) = Entropía(S)$$

$$I(S) = \sum_j P_j \log_2 P_j$$

$$I(S) = P_{Sí} \log_2 P_{Sí} + P_{No} \log_2 P_{No}$$

$$I(S) = 0{,}4 \log_2 0{,}4 + 0{,}6 \log_2 0{,}6 = 0{,}97$$

Respecto a la entropía de la partición de “Temperatura” ya sabemos cómo hacerlo. Se trata de una suma ponderada de las entropías de cada partición generada (fórmula 3). Rellenemos dicha fórmula con los valores de las particiones de “Temperatura”. Desenrollada será más fácil la sustitución:

$$IP(S) = P_{Calor}\, Entropía(S)_{Calor} + P_{Medio}\, Entropía(S)_{Medio} + P_{Frío}\, Entropía(S)_{Frío}$$

La entropía de la VD con la propiedad “calor”:

$$Entropía(S)_{Calor} = \sum_j P_j \log_2 P_j$$

$$Entropía(S)_{Calor} = P_{Sí} \log_2 P_{Sí} + P_{No} \log_2 P_{No}$$

$$Entropía(S)_{Calor} = 0 \log_2 0 + 1 \log_2 1 = 0$$

La entropía de la VD con la propiedad “medio”:

$$Entropía(S)_{Medio} = P_{Sí} \log_2 P_{Sí} + P_{No} \log_2 P_{No}$$

$$Entropía(S)_{Medio} = \frac{2}{3}\log_2 \frac{2}{3} + \frac{1}{3}\log_2 \frac{1}{3} = 0{,}92$$

Y la de frío es directamente 0, pues no existen ejemplares con esa propiedad de Temperatura en la partición de “Soleado”.

Por otro lado, las ponderaciones de cada entropía:

$$P_{calor} = \frac{2}{5} = 0{,}4$$

$$P_{Medio} = \frac{3}{5} = 0{,}6$$

$$P_{Frio} = \ 0$$

Sustituyendo todo esto en la fórmula $IP(S)$ tenemos:

$$IP(S) = \ 0{,}4 \ x \ 0 + \ 0{,}6 \ x \ 0{,}92 + \ 0 \ x \ 0 = 0{,}55$$

La Ganancia de Información de la partición a partir de la Variable Temperatura será entonces:

$$IG(S) = \ 0{,}97 - \ 0{,}55 = 0{,}42$$

Respecto a las variables “Humedad”, la Ganancia de Información es:

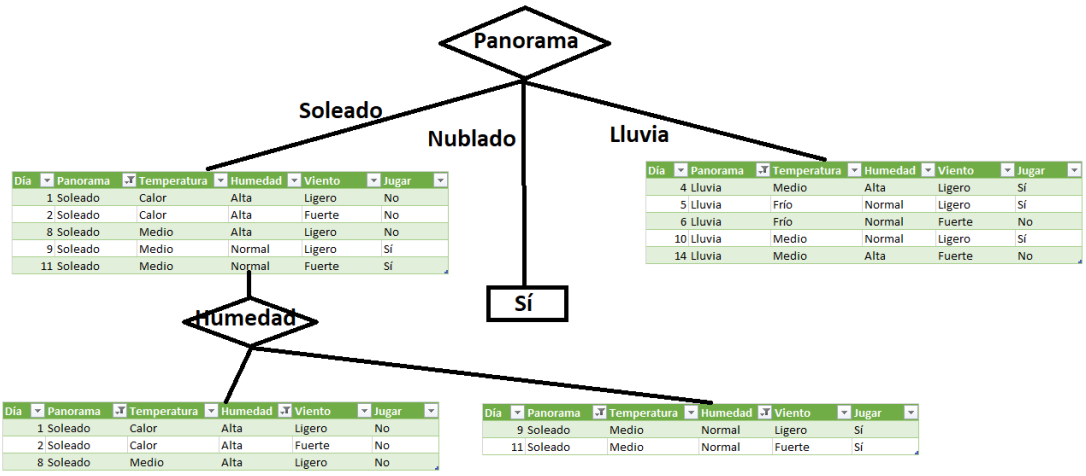
$$IG(S) = \ 0{,}97 - \ 0 = \ 0{,}97$$

Nótese que en “Alta” Humedad nunca se juega, y en “Baja” Humedad siempre. No existe incertidumbre en esta partición.

Respecto a Viento, la Ganancia de Información de su partición es:

$$IG(S) = \ 0{,}97 - \ 0{,}94 = \ 0{,}021$$

La conclusión es que debajo del nodo de “Soleado” hemos de colgar la partición de “Humedad”. Esto nos proporcionará que la rama deje de ser incierta.



Día	Panorama	Temperatura	Humedad	Viento	Jugar
1	Soleado	Calor	Alta	Ligero	No
2	Soleado	Calor	Alta	Fuerte	No
8	Soleado	Medio	Alta	Ligero	No
9	Soleado	Medio	Normal	Ligero	Sí
11	Soleado	Medio	Normal	Fuerte	Sí

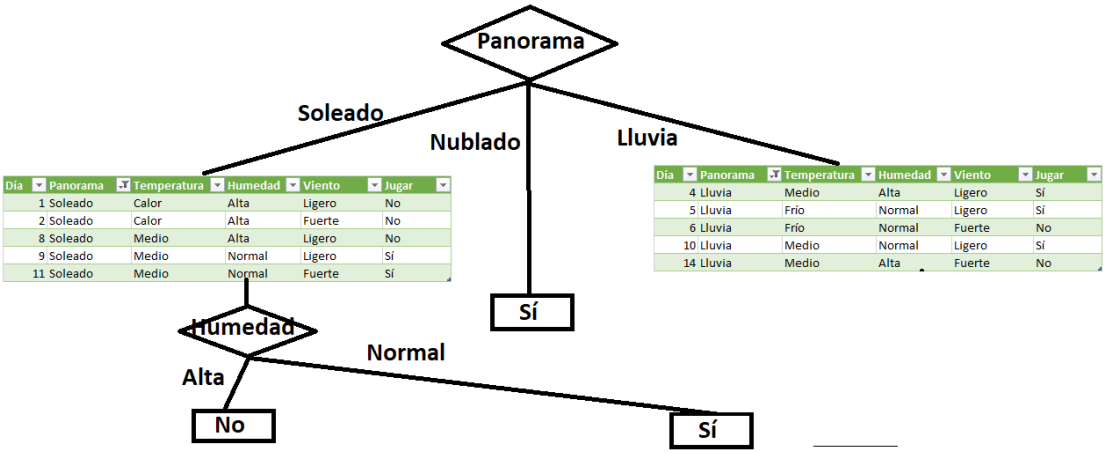
Día	Panorama	Temperatura	Humedad	Viento	Jugar
4	Lluvia	Medio	Alta	Ligero	Sí
5	Lluvia	Frío	Normal	Ligero	Sí
6	Lluvia	Frío	Normal	Fuerte	No
10	Lluvia	Medio	Normal	Ligero	Sí
14	Lluvia	Medio	Alta	Fuerte	No

Día	Panorama	Temperatura	Humedad	Viento	Jugar
1	Soleado	Calor	Alta	Ligero	No
2	Soleado	Calor	Alta	Fuerte	No
8	Soleado	Medio	Alta	Ligero	No

Día	Panorama	Temperatura	Humedad	Viento	Jugar
9	Soleado	Medio	Normal	Ligero	Sí
11	Soleado	Medio	Normal	Fuerte	Sí

Figura 4

Nuevamente, ambas ramas de “Humedad” pasan de ser nodos de decisión a ser hojas. Esto es debido a que no hay más decisión que un “No” rotundo en Humedad Alta y un “Sí” rotundo en Humedad Normal. Podríamos expresar el árbol con sus nodos transformados en hojas (figura 5):



Día	Panorama	Temperatura	Humedad	Viento	Jugar
1	Soleado	Calor	Alta	Ligero	No
2	Soleado	Calor	Alta	Fuerte	No
8	Soleado	Medio	Alta	Ligero	No
9	Soleado	Medio	Normal	Ligero	Sí
11	Soleado	Medio	Normal	Fuerte	Sí

Día	Panorama	Temperatura	Humedad	Viento	Jugar
4	Lluvia	Medio	Alta	Ligero	Sí
5	Lluvia	Frío	Normal	Ligero	Sí
6	Lluvia	Frío	Normal	Fuerte	No
10	Lluvia	Medio	Normal	Ligero	Sí
14	Lluvia	Medio	Alta	Fuerte	No

Figura 5

Rama de Lluvia

Desarrollando la otra rama, la de “Lluvia”, la mayor ganancia de Información en la distribución de la VD se conseguiría con la partición de “Viento”. De nuevo, los nodos generados en dicha partición se transformarían en hojas, pues la incertidumbre es nula en ambos. El árbol quedaría así:

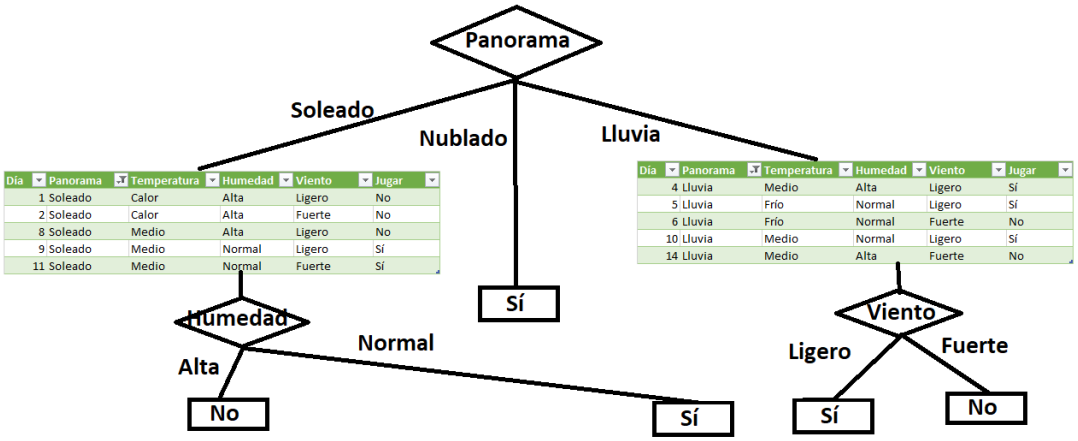


Figura 6

Cuestiones adicionales sobre los árboles de decisión

- No siempre los árboles lograrán un estado en que la incertidumbre sea nula. Más bien lo contrario. La incertidumbre se reducirá en el proceso de generación de nodos, pero siempre se contará con un riesgo de clasificación incorrecta. Eso obliga a conciliar dos parámetros, la profundidad deseada en la generación de nodos y el riesgo asumido (la entropía permitida). Téngase en cuenta que estas técnicas se emplean en conjuntos de datos que sobrepasan al de este ejemplo, donde no se darán situaciones puras de certidumbre. De esa manera, un nodo se podrá transformar en hoja si sobrepasa simplemente un umbral de certidumbre.

- Una Variable Independiente podrá tener variables continuas. Lo que hará el algoritmo es hacer particiones binarias empleando un umbral y tomar en cuenta todas las posibles a la hora de calcular la entropía de la Variable Dependiente. De la misma forma, de ser elegida la variable para la siguiente partición, se empleará la partición con el umbral más exitoso en cuanto a Ganancia de Información. Aunque en el ejemplo presentado son todas variables categóricas o discretas, lo importante en este texto es dejar constancia de que el algoritmo también vale para variables con valores continuos. El segundo de los ejemplos con código R que se presentan a continuación emplea Variables Independientes continuas.

Práctica con código R

Ejercicio del ejemplo desarrollado (VI categóricas)

En primer lugar, veamos el ejercicio desarrollado anteriormente pero ahora usemos las funciones que algunas librerías de R proporcionan. Recordemos que todas las variables, Independientes y Dependiente, son variables categoriales.

El archivo para leer y transformar en una estructura con formato R, en este caso, un "Dataframe", sería el siguiente:

golf.txt:

```
Panorama,Temperatura,Humedad,Viento,Jugar
Soleado,Calor,Alta,Ligero,No
Soleado,Calor,Alta,Fuerte,No
Nublado,Calor,Alta,Ligero,Si
Lluvia,Medio,Alta,Ligero,Si
Lluvia,Frio,Normal,Ligero,Si
Lluvia,Frio,Normal,Fuerte,No
Nublado,Frio,Normal,Fuerte,Si
Soleado,Medio,Alta,Ligero,No
Soleado,Medio,Normal,Ligero,Si
Lluvia,Medio,Normal,Ligero,Si
Soleado,Medio,Normal,Fuerte,Si
Nublado,Medio,Alta,Fuerte,Si
Nublado,Calor,Normal,Ligero,Si
Lluvia,Medio,Alta,Fuerte,No
```

Y el código a utilizar sería el que se muestra a continuación (Puede consultarse el manual de la librería empleada (RWeka.pdf (r-project.org)). El algoritmo “C4.5” es llamado en ella “J48”):

```
library(RWeka)
library(caret)
library(datasets)



table_golf = read.table("golf.txt",
         header = T,
         sep = ",",
        )

data_golf <- as.data.frame(table_golf)


data_golf$Panorama=as.factor(data_golf$Panorama)
data_golf$Temperatura=as.factor(data_golf$Temperatura)
data_golf$Humedad=as.factor(data_golf$Humedad)
data_golf$Viento=as.factor(data_golf$Viento)
data_golf$Jugar=as.factor(data_golf$Jugar)



modelC45 <- J48(`Jugar` ~ ., data = data_golf)
modelC45

plot(modelC45)
```

Corriendo este código con el archivo “golf.txt” cargado obtenemos dos salidas que nos resultan familiares. En primer lugar, obtenemos el árbol en una salida que aún no está graficada (figura 7). No hay más que editar el modelo con el comando “print(modelC45)”. Este gráfico representa el mismo que nosotros hemos desarrollado antes en la figura 6.

```
J48 pruned tree
------------------

Panorama = Lluvia
|   Viento = Fuerte: No (2.0)
|   Viento = Ligero: Si (3.0)
Panorama = Nublado: Si (4.0)
Panorama = Soleado
|   Humedad = Alta: No (3.0)
|   Humedad = Normal: Si (2.0)

Number of Leaves  :     5

Size of the tree :      8
```

Figura 7

No obstante, una versión más amigable del árbol la obtenemos ejecutando una orden de graficación (figura 8). En concreto, el comando “plot(modelC45)”. De esta forma obtenemos un árbol en forma de gráfico similar al de la figura 6.

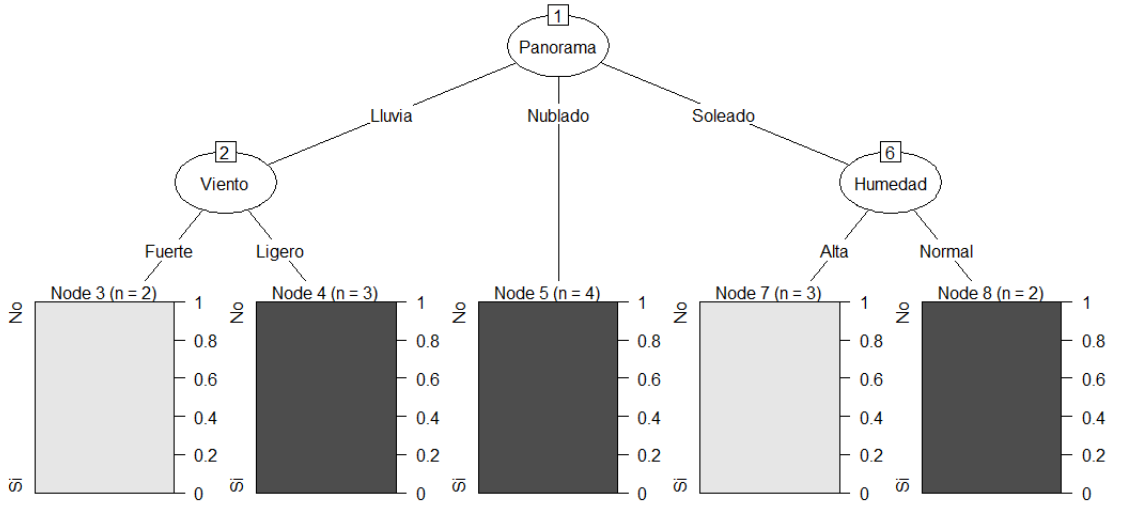


Figura 8

Ejercicio de tres especies de flores Iris (VI continuas)

Veamos otro ejemplo con R:

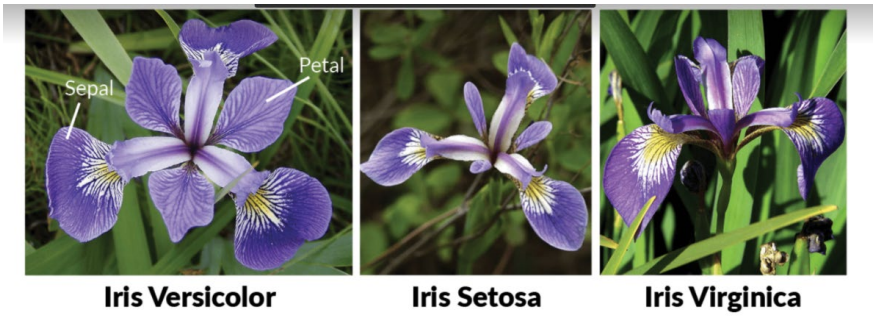


Figura 9

En la tabla 2 se presenta parte de la tabla a analizar. En concreto, el ejemplo clásico de las características de tres especies de flores Iris (ver figura 9 para una imagen). La especie es la VD a clasificar con tres clases {Versicolor, Virginica, Setosa}. Las Variables Independientes son el largo y el ancho del pétalo y el ancho y largo del sépalo (tabla 2). Como se ve, las Variables Independientes son continuas. Lo que hará el algoritmo es hacer particiones binarias empleando un umbral entre dos valores de la distribución.

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
5.0	2.0	3.5	1.0	versicolor
6.0	2.2	4.0	1.0	versicolor
6.0	2.2	5.0	1.5	virginica
6.2	2.2	4.5	1.5	versicolor
4.5	2.3	1.3	0.3	setosa
5.0	2.3	3.3	1.0	versicolor
5.5	2.3	4.0	1.3	versicolor
6.3	2.3	4.4	1.3	versicolor
4.9	2.4	3.3	1.0	versicolor
5.5	2.4	3.8	1.1	versicolor
5.5	2.4	3.7	1.0	versicolor
4.9	2.5	4.5	1.7	virginica
5.1	2.5	3.0	1.1	versicolor
5.5	2.5	4.0	1.3	versicolor
…	…	…	…	…
…	…	…	…	…
…	…	…	…	…

Tabla 2

El código a aplicar será el siguiente:

```
library(RWeka)
library(caret)
library(datasets)

data(iris)
summary(iris)

set.seed(1958)  # fijar la semilla para obtener replicabilidad
train <- createFolds(iris$Species, k=10)

C45Fit <- train(Species ~., method="J48", data=iris,
           tuneLength = 5,
           trControl = trainControl(
            method="cv", indexOut=train))
C45Fit$finalModel

plot(C45Fit$finalModel)
```

De igual manera obtenemos el árbol en formato resumido (figura 10):

```
J48 pruned tree
------------------

Petal.Width <= 0.6: setosa (50.0)
Petal.Width > 0.6
|   Petal.Width <= 1.7
|   |   Petal.Length <= 4.9: versicolor (48.0/1.0)
|   |   Petal.Length > 4.9
|   |   |   Petal.Width <= 1.5: virginica (3.0)
|   |   |   Petal.Width > 1.5
|   |   |   |   Sepal.Length <= 6.9: versicolor (2.0)
|   |   |   |   Sepal.Length > 6.9: virginica (1.0)
|   Petal.Width > 1.7: virginica (46.0/1.0)

Number of Leaves  :     6

Size of the tree :      11
```

Figura 10

Aplicando una graficación más sofisticada mediante el comando “plot(C45Fit$finalModel)” obtenemos el dibujo del árbol (figura 11).

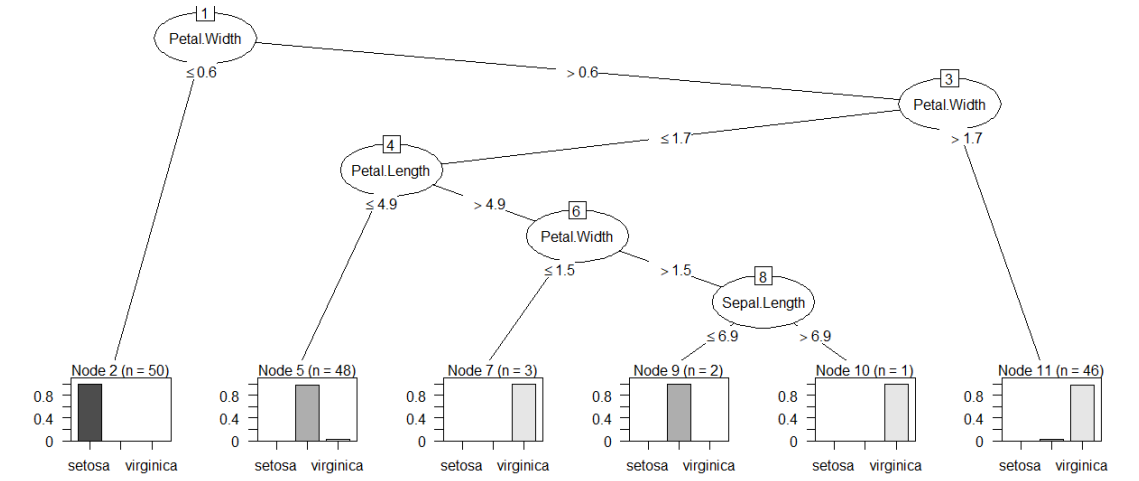


Figura 11