Modelos supervisados

PID_00284578

Raúl Montoliu Colás

Tiempo mínimo de dedicación recomendado: 1 hora

Universitat Oberta de Catalunya

Raúl Montoliu Colás

Ingeniero en Informática por la Universidad Jaume I (UJI) de Castellón. Doctor en métodos avanzados informáticos por la misma universidad. Actualmente trabaja como docente en el departamento de Ingeniería y Ciencia de los Computadores de la UJI y como investigador en el grupo de investigación Machine Learning for Smart Environments del Instituto de Nuevas Tecnologías de la Imagen (INIT). Desde el 2017 colabora como docente en la Universitat Oberta de Catalunya (UOC).

El encargo y la creación de este recurso de aprendizaje UOC han sido coordinados por el profesor: Julià Minguillón Alfonso

Primera edición: septiembre 2021

Av. Tibidabo, 39-43, 08035 Barcelona
Autoría: Raúl Montoliu Colás
Producción: FUOC

Índice

Introducción

Este módulo está dedicado a los modelos supervisados. En los modelos supervisados, las muestras que se usarán para construir el modelo han sido previamente etiquetadas por un experto. Existen dos tipos principales de problemas supervisados: los problemas supervisados de regresión y los supervisados de clasificación. En el primer caso, la variable objetivo que hay que predecir es una variable continua. Por lo que respecta a los problemas supervisados de clasificación, la variable objetivo es discreta, y por lo tanto el problema consiste en predecir, dada una muestra nueva, a qué conjunto, de los posibles, pertenece. A cada uno de los posibles conjuntos se le conoce como *clase*.

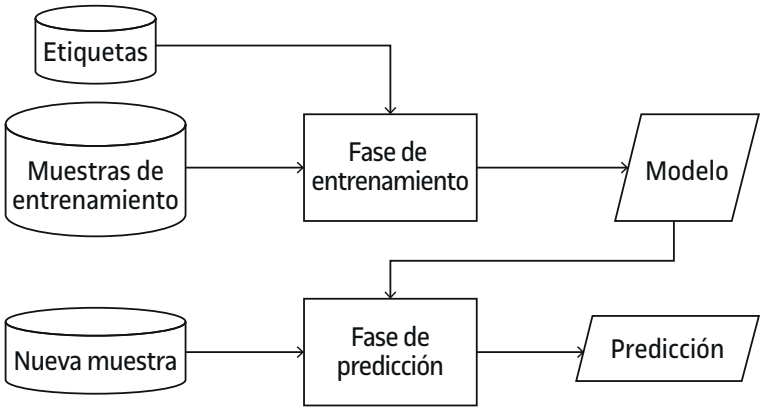
Un ejemplo clásico de problema supervisado de regresión es predecir el precio de venta de una vivienda, a partir de los datos de esta. Es supervisado, puesto que el aprendizaje se desarrolla a partir de un conjunto de muestras de viviendas de las que ya se conoce el precio de venta. Es de regresión, puesto que el precio de venta es una variable continua.

Un ejemplo clásico de problema supervisado de clasificación es, dada una mamografía, decidir si hay cáncer o no. Es supervisado, puesto que, para poder crear el modelo de aprendizaje, se ha tenido que usar un conjunto de mamografías previamente etiquetadas por un experto. Es de clasificación, puesto que la variable objetivo es discreta. En este ejemplo existirán dos clases: tener cáncer o no tenerlo.

Para crear un modelo supervisado, se necesita un conjunto de muestras de entrenamiento y el valor de la variable objetivo. Al valor de la clase objetivo también se le conoce como *etiqueta* de la muestra. A la fase de creación del modelo también se suele llamarla *fase de entrenamiento del modelo*. Esta fase tendrá como entrada las muestras de entrenamiento y las etiquetas de cada muestra, y producirá como salida el modelo.

Posteriormente, cuando tengamos una nueva muestra de la que deseamos predecir o conocer el valor de la variable objetivo, se aplicará la fase de predicción. Esta fase toma como entrada la nueva muestra y el modelo previamente obtenido. A la salida se obtendrá el valor predicho de la variable objetivo. La figura 1 muestra un esquema del funcionamiento de los problemas supervisados.

Figura 1. Esquema del funcionamiento de los problemas supervisados



1. Modelos basados en vecindad

El algoritmo *k*-NN o *k - vecinos más cercanos* (en inglés, *k nearest neighbours*) es uno de los algoritmos de clasificación supervisada más simples que existen. A pesar de ello, en muchos problemas reales suele obtener excelentes resultados comparándolo con otros algoritmos más complejos.

Una de sus principales características es que no incluye fase de entrenamiento. Por lo tanto, no se genera un modelo que luego se usará para clasificar las muestras nuevas. A este tipo de algoritmos se les llama *métodos de aprendizaje vagos*, o *lazy learning methods*, en inglés.

El funcionamiento del método es muy sencillo. Para cada muestra nueva por clasificar, se calcula la distancia con todas las muestras de entrenamiento y se seleccionan las *k* muestras más cercanas. La etiqueta de la muestra nueva se determina como la etiqueta mayoritaria entre sus *k* muestras vecinas más cercanas.

Su principal desventaja es la lentitud de la fase de predicción, puesto que es necesario calcular la distancia de la nueva muestra con respecto a todas las muestras de entrenamiento. En conjuntos de entrenamiento muy grandes, este proceso puede requerir mucho tiempo.

La figura 2 presenta un conjunto de muestras que nos servirá para ilustrar el funcionamiento de este algoritmo. Las muestras de entrenamiento son las aspas azules, los círculos rojos y los cuadrados verdes. La muestra nueva es el triángulo negro situado en las coordenadas [1.25,3]. El objetivo es averiguar a qué clase de las tres existentes pertenece la muestra nueva. Para ello, en primer lugar se calcula la distancia a todas las muestras de entrenamiento. Por ejemplo, podemos usar la distancia euclidiana. Las distancias que se obtienen en este ejemplo se muestran en la tabla 1.

Si $k = 1$, la muestra más cercana es la número 6 ([1,3]). Como es de color azul, se concluiría que la muestra es azul. Sin embargo, si $k = 3$, la segunda y la tercera muestras más cercanas son de la clase roja: muestras número 8 ([1.5,3.5]) y 7 ([2,3]), respectivamente. Por lo tanto, con $k = 3$, se concluiría que la muestra es roja, puesto que, de las tres muestras vecinas, dos son rojas y una azul. Si ampliamos el valor *k*, el resultado puede cambiar. Por ejemplo, con $k = 5$, tres de las muestras más cercanas son azules (las muestras 6, 4 y 5) y dos rojas (las muestras 7 y 8), por lo que, en este caso, la conclusión sería clasificar la nueva muestra como azul.

Este sencillo ejemplo demuestra que la elección del valor k puede hacer que varíe el resultado obtenido.

Figura 2. Conjunto de datos con tres clases. La muestra dibujada con un triángulo negro situada en las coordenadas [1.25,3] es la muestra nueva.

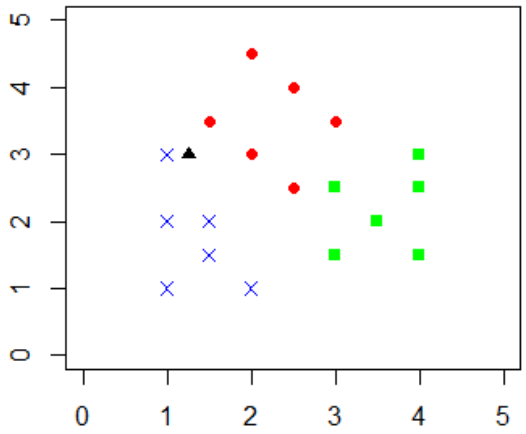


Tabla 1. Distancias de la nueva muestra (situada en las coordenadas [1.25,3]) con respecto a todas las muestras de entrenamiento. La distancia se ha redondeado con dos decimales.

Número muestra	Clase	Coordenadas	Distancia
1	Azul	[1.0,1.0]	2.02
2	Azul	[2.0,1.0]	2.14
3	Azul	[1.5,1.5]	1.52
4	Azul	[1.0,2.0]	1.03
5	Azul	[1.5,2.0]	1.03
6	Azul	[1.0,3.0]	0.25
7	Roja	[2.0,3.0]	0.75
8	Roja	[1.5,3.5]	0.56
9	Roja	[2.5,2.5]	1.35
10	Roja	[2.0,4.5]	1.68
11	Roja	[2.5,4.0]	1.60
12	Roja	[3.0,3.5]	1.82
13	Verde	[3.0,1.5]	2.30
14	Verde	[3.0,2.5]	1.82
15	Verde	[3.5,2.0]	2.46
16	Verde	[4.0,1.5]	3.13
17	Verde	[4.0,2.5]	2.79
18	Verde	[4.0,3.0]	2.75

1.1. Detalles importantes del método

Para que el método funcione correctamente, es recomendable que los valores de los atributos sean números continuos y que estén normalizados. En este caso, puede usarse la distancia euclidiana, que se define como se muestra a continuación:

$$d(x_i,x_j) = \sqrt{\sum_{a=1}^{M}(x_i(a) - x_j(a))^2} \qquad (1)$$

donde x_i y x_j son dos muestras que tienen M atributos, y $x_i(a)$, $x_j(a)$ son el valor del atributo a-ésimo de las muestras x_i y x_j, respectivamente.

Existen otras funciones de distancia que pueden usarse. Por ejemplo, en el caso de tratar con atributos nominales o binarios, una de las más utilizadas es la distancia de Hamming, que se define como se muestra a continuación:

$$d(x_i,x_j) = \sum_{a=1}^{M} \delta(x_i(a),x_j(a)) \quad (2)$$

donde $\delta(x_i(a),x_j(a))$ toma el valor cero si $x_i(a) = x_j(a)$, y 1 en caso contrario.

En el caso de valores discretos o no numéricos, deberán crearse funciones de distancia específicas que tengan en cuenta el tipo de datos del atributo. Por ejemplo, imaginemos que tenemos una base de datos de personas y de cada una se almacena la edad, el salario y los estudios universitarios completados. Los atributos Edad y Salario no presentan problemas, al ser variables numéricas continuas. El cálculo de la distancia en el atributo Estudios universitarios completados es más complicado. Por ejemplo, podríamos hacernos las siguientes preguntas: ¿cuál es la distancia entre *Ingeniería Informática* y *Arquitectura Técnica*? y ¿esa distancia es superior o inferior a la distancia entre *Ingeniería Industrial* y *Derecho*? Para poder aplicar el algoritmo *k*-NN, deberíamos poder resolver estas preguntas. A este fin, debería crearse una función de distancia que fuese capaz de tratar este tipo de datos. Por ejemplo, a la distancia entre *Ingeniería Informática* y *Arquitectura Técnica* se le podría dar valor 0.5, y a la distancia entre *Ingeniería Industrial* y *Derecho* se le podría dar el valor 0.1. El rango de valores que debería obtenerse debería ser similar al que se consigue al calcular la distancia entre los valores numéricos normalizados.

La fórmula de distancia para aplicar en este ejemplo podría ser la siguiente:

$$d(x_i,x_j) = \sqrt{d_{edad}(x_i,x_j) + d_{salario}(x_i,x_j) + d_{estudios}(x_i,x_j)}$$

donde

$$d_{edad}(x_i,x_j) = (x_i(edad) - x_j(edad))^2$$

$$d_{salario}(x_i,x_j) = (x_i(salario) - x_j(salario))^2$$

y $d_{estudios}$ devolvería el valor de la diferencia entre dos valores del atributo Estudios universitarios completados, que tendría que diseñarse tal como se ha comentado anteriormente.

1.2. Variantes del *k*-NN

Existen muchas variantes del método *k*-NN. Un método muy utilizado es el *weighted k-NN*, donde los vecinos más cercanos contribuyen más a la elección de la clase que los más lejanos.

Volviendo al ejemplo mostrado en la figura 2, con $k = 3$, hemos concluido que la muestra nueva pertenece a la clase roja, puesto que dos de los tres vecinos más cercanos son rojos y solo uno es azul. Por lo tanto, la clase roja tiene 2 votos por 1 de la azul.

Podríamos usar como peso la inversa de la distancia. Así, el peso del primer vecino (muestra 6 azul) sería 1/0.25 = 4.0 (véase tabla 1); del segundo (muestra 8 roja), 1/0.56 = 1.78, y del tercero (muestra 7 roja), 1/0.75 = 1.33. Por lo tanto, el primer vecino contribuye con 4.0 a la votación para la clase azul, y los dos siguientes vecinos contribuyen con 1.78 y 1.33 a la votación para la clase roja. Puesto que los votos para la clase azul son superiores a la suma de los votos para la clase roja ($4.0 > 1.78 + 1.33$), se concluiría que la muestra es azul.

Existen otras variantes que asignan pesos para que determinados atributos tengan más importancia que otros a la hora de calcular la distancia.

2. Árboles de decisión

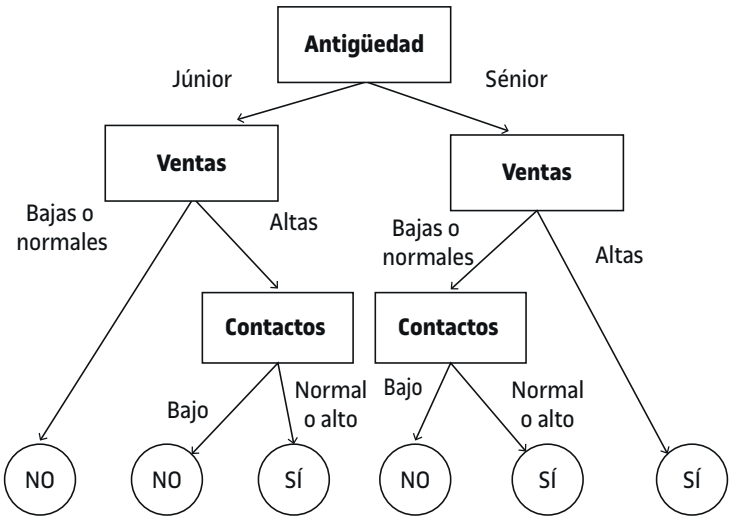
Los árboles de decisión son uno de los modelos supervisados de clasificación que se usan más en problemas de minería de datos. La razón principal es porque tienen una alta capacidad explicativa y porque es muy fácil interpretar el modelo que se obtiene. Los árboles de decisión pueden usarse tanto en problemas supervisados de clasificación como en problemas supervisados de regresión. En este módulo solo trataremos los primeros.

La idea principal de los árboles de decisión es subdividir el espacio de datos de entrada para generar regiones disjuntas, de forma que todas las muestras que pertenezcan a la misma región sean de la misma clase. Si una región tiene muestras de diferentes clases, será subdividida en regiones más pequeñas siguiendo el mismo criterio. El proceso finaliza cuando se han partido las muestras de entrada en regiones de forma que para todas se cumple que solo tienen muestras de una única clase. Un árbol de decisión se llama *completo* o *puro* si es posible construir un árbol donde se cumpla la condición anterior.

Un árbol de decisión consta de nodos hoja o terminales, que representan regiones etiquetadas de acuerdo a una clase, y nodos internos o *splits*, que representan condiciones que permiten decidir a qué subregión va cada elemento que llega a dicho nodo. Cuando, en la fase de predicción, se presenta una muestra nueva a un árbol de decisión, se empieza por el nodo raíz que contiene una condición, la cual determinará por qué rama del árbol debe ir la muestra. Una vez seleccionada la rama, la muestra llegará o bien a un nodo con otra condición o a un nodo terminal, en cuyo caso se determinará la etiqueta predicha para la muestra como la indicada en el nodo terminal.

La figura 3 muestra un ejemplo de un posible árbol de decisión para un hipotético problema supervisado. Las muestras se refieren a los datos de una serie de comerciales de una empresa y tienen tres atributos. El primer atributo es la antigüedad y sus posibles valores son *Senior* o *Junior*. El segundo es el número de ventas que han llevado a cabo y sus posibles valores son *Bajas*, *Normales* o *Altas*. Por último, el tercer atributo es el número de contactos que han realizado y sus posibles valores son *Bajo*, *Normal* o *Alto*. La variable objetivo es si obtendrán o no un bonus al final del año. Para crear el árbol anterior, se han usado los datos de los últimos años de la empresa. Por lo tanto, las muestras estarán etiquetadas, puesto que se sabe si obtuvieron o no el bonus.

Figura 3. Ejemplo de un árbol de decisión



Supongamos que tenemos una muestra nueva de la que quiere predecirse si obtendrá el bonus o no. La nueva muestra tiene los valores *[Senior, Bajas, Normal]*. Para predecir la variable objetivo, empezaremos en el nodo raíz, donde, dependiendo del valor del atributo Antigüedad, irá por la izquierda o por la derecha. Puesto que el valor es *Senior*, irá por la rama de la derecha. En el siguiente nodo se nos preguntará por Número de ventas. Puesto que el valor para ese atributo es *Bajas*, iremos por la rama de la izquierda. En el siguiente nodo se comprobará el atributo Contactos. Al tener el valor *Alto*, iremos por la derecha para llegar al nodo terminal, que nos indica que obtendrá el bonus.

Imaginamos ahora que tenemos otra muestra nueva de la que quiere predecirse si obtendrá el bonus o no. La muestra nueva tiene los valores *[Junior, Normal, Alto]*. De igual forma al caso anterior, empezaremos en el nodo raíz, donde, dependiendo del valor del atributo Antigüedad, irá por la izquierda o por la derecha. Puesto que el valor es *Junior*, irá por la rama de la izquierda. En el siguiente nodo nos preguntará por Número de ventas. Puesto que el valor para ese atributo es *Normal*, iremos por la rama de la izquierda y llegaremos directamente a un nodo terminal, que nos indica que no obtendrá el bonus. En este caso, no ha sido necesario consultar el valor del atributo Contactos.

A partir del árbol creado, resulta sencillo extraer un conjunto de reglas que ayudan a entender el problema de clasificación. Las reglas podrían resumirse en que se obtendrá el bonus si se cumplen algunas de las siguientes condiciones:

- La antigüedad es sénior y las ventas son altas.
- La antigüedad es sénior, las ventas son bajas o normales y los contactos son normales o altos.
- La antigüedad es júnior, las ventas son altas y los contactos normales o altos.

En cualquier otro caso, no se obtendrá el bonus.

De las reglas anteriores, pueden extraerse conclusiones importantes. Por ejemplo, una de ellas es que un comercial sénior tiene más facilidades para obtener el bonus que un júnior.

2.1. Construcción del árbol de decisión

Para explicar cómo se construyen los árboles de decisión, se usarán los datos de ejemplo mostrados en la tabla 2, donde se presenta un conjunto de muestras sobre si podrá jugarse o no a golf según una serie de parámetros. Para cada muestra, se dispone del tiempo (*Lluvioso*, *Nublado* o *Soleado*), la temperatura (*Calor*, *Normal* o *Frío*), la humedad (*Alta* o *Normal*) y el viento (*Sí* o *No*). La etiqueta de cada muestra es *Sí*, si puede jugarse con esas condiciones, o *No*, si no es posible.

Tabla 2. Datos de ejemplo sobre si podrá o no jugarse a golf según las condiciones atmosféricas.

Tiempo	Temperatura	Humedad	Viento	¿Se jugará?
Lluvioso	Calor	Alta	No	No
Lluvioso	Calor	Alta	Sí	No
Nublado	Calor	Alta	No	Sí
Soleado	Normal	Alta	No	Sí
Soleado	Frío	Normal	No	Sí
Soleado	Frío	Normal	Sí	No
Nublado	Frío	Normal	Sí	Sí
Lluvioso	Normal	Alta	No	No
Lluvioso	Frío	Normal	No	Sí
Soleado	Normal	Normal	No	Sí
Lluvioso	Normal	Normal	Sí	Sí
Nublado	Normal	Alta	Sí	Sí
Nublado	Calor	Normal	No	Sí
Soleado	Normal	Alta	Sí	No

El problema que se plantean los métodos de construcción de árboles de decisión es el siguiente: ¿cuál es la mejor secuencia de preguntas para saber, a partir de la descripción de una muestra en términos de sus atributos, a qué clase corresponde? Evidentemente, «la mejor secuencia» puede entenderse como aquella que, con el mínimo número de preguntas, devuelve una respuesta lo suficientemente detallada.

En otras palabras, la «mejor secuencia» es la que efectúa las preguntas más discriminantes, es decir, aquellas cuyas respuestas permiten descartar un número más amplio de objetos diferentes del que estamos considerando y, por tanto, llegar con mayor rapidez a decidir a qué clase pertenece.

Uno de los algoritmos más famosos para construir árboles de decisión se llama ID3 y hace uso de la teoría de la información. El algoritmo ID3 construye un árbol de decisión de arriba abajo basándose únicamente en los ejemplos

iniciales proporcionados. Para ello, usa el concepto de *ganancia de información* para seleccionar el atributo más útil en cada paso. El atributo más útil es aquel que permite separar mejor los ejemplos respecto a la clasificación final.

Para poder aplicar el algoritmo, hemos de comenzar sabiendo cómo se mide la ganancia de información, y para ello hay que introducir el concepto de *entropía de Shannon*, como se muestra a continuación:

$$E(S) = \sum_{i=1}^{C} -p_i \log_2(p_i) \tag{3}$$

donde S es un conjunto de muestras, C es el número de clases diferentes que tenemos y cada p_i es la proporción de muestras que hay de la etiqueta i-ésima en el conjunto. Por ejemplo, $S = 14$ y $C = 2$ para los datos mostrados en la tabla 2, puesto que hay 14 muestras y 2 clases.

En el caso particular de un problema binario (como el presentado en la tabla 2), la fórmula anterior puede expresarse como:

$$E(S) = -P_+ \log_2(P_+) - P_- \log_2(P_-) \tag{4}$$

donde P_+ y P_- son las proporciones de muestras positivas y negativas, respectivamente.

Sabiendo que, en el ejemplo mostrado en la tabla 2, hay 9 muestras positivas (sí que se jugará al golf) y 5 negativas (no se jugará al golf), obtendremos $P_+ = 9/14$ y $P_- = 5/14$. Por lo tanto, podemos calcular $E(S)$ como:

$$\begin{aligned} E(S) &= -\frac{9}{14}\log_2\left(\frac{9}{14}\right) - \frac{5}{14}\log_2\left(\frac{5}{14}\right) = \\ &= 0.64\log_2(0.64) - 0.36\log_2(0.36) = \\ &= 0.94 \end{aligned}$$

Para decidir por qué atributo tenemos que dividir primero, tenemos que calcular la ganancia de información que se obtiene para cada atributo. Para ello, debemos calcular la entropía de cada posible división por un atributo como:

$$E(S,X) = \sum_{q \in X} P(S_q)E(S_q) \tag{5}$$

donde X representa el conjunto de valores de un determinado atributo, q es cada uno de esos valores, $P(S_q)$ es la proporción de muestras que tienen el valor q en ese atributo y $E(S_q)$ es la entropía de las muestras con valor q para

el atributo estudiado. $E(S_q)$ se calcula con la ecuación 3 o con su equivalente 4, si el problema es binario.

Por ejemplo, para el atributo Tiempo, que tiene tres posibles valores $X = \{Lluvioso, Nublado, Soleado\}$, $E(S, Tiempo)$ se calculará como se muestra a continuación:

$$\begin{aligned} E(S,Tiempo) &= P(S_{Lluvioso})E(S_{Lluvioso}) + \\ &+ P(S_{Nublado})E(S_{Nublado}) + \\ &+ P(S_{Soleado})E(S_{Soleado}) \end{aligned}$$

Cada uno de esos valores se calcula como sigue, teniendo en cuenta que $S_{Lluvioso}$, $S_{Nublado}$ y $S_{Soleado}$ son el conjunto de muestras donde el atributo Tiempo tiene valor *Lluvioso* (5 muestras), *Nublado* (4 muestras) y *Soleado* (5 muestras), respectivamente:

$$\begin{aligned} P(S_{Lluvioso}) &= 5/14 \\ P(S_{Nublado}) &= 4/14 \\ P(S_{Soleado}) &= 5/14 \\ E(S_{Lluvioso}) &= -\frac{2}{5}\log_2\left(\frac{2}{5}\right) - \frac{3}{5}\log_2\left(\frac{3}{5}\right) \\ E(S_{Nublado}) &= -\frac{4}{4}\log_2\left(\frac{4}{4}\right) - \frac{0}{4}\log_2\left(\frac{0}{4}\right) \\ E(S_{Soleado}) &= -\frac{3}{5}\log_2\left(\frac{3}{5}\right) - \frac{2}{5}\log_2\left(\frac{2}{5}\right) \end{aligned}$$

$P(S_{Lluvioso})$ es la proporción de muestras que tienen el valor *Lluvioso* para el atributo Tiempo, respecto al total. Puesto que hay 5 muestras que cumplen esta condición y 14 en total, $P(S_{Lluvioso}) = 5/14$. El resto de proporciones se calculan de la misma forma. Para calcular $E(S_{Lluvioso})$, debemos obtener el número de muestras positivas y negativas pero únicamente mirando las 5 muestras que tienen el valor *Lluvioso* para el atributo Tiempo. En este caso, hay 2 positivas y 3 negativas. Para calcular el resto de entropías, se procede de la misma forma. Es importante destacar que tenemos que considerar $0\log_2(0) = 0$ para poder calcular la entropía $E(S_{Nublado})$.

Por lo tanto, $E(S, Tiempo) = (5/14) * 0.971 + (4/14) * 0.0 + (5/14) * 0.971 = 0.693$.

La ganancia de información para un atributo concreto se define como:

$$G(S,X) = E(S) - E(S,X) \tag{6}$$

Para el atributo Tiempo, la ganancia de información será $G(S,Tiempo) = 0.94 - 0.693 = 0.247$.

Para el atributo Temperatura, que tiene tres posibles valores $X = \{Calor, Normal, Frío\}$, $E(S,Temperatura)$ se calculará como se muestra a continuación:

$$\begin{aligned} E(S,Temperatura) &= P(S_{Calor})E(S_{Calor}) + \\ &+ P(S_{Normal})E(S_{Normal}) + \\ &+ P(S_{Frío})E(S_{Frío}) \end{aligned}$$

Cada uno de esos valores se calcula como sigue, considerando que S_{Calor}, S_{Normal} y $S_{Frío}$ tienen 4, 6 y 4 muestras, respectivamente:

$$\begin{aligned} P(S_{Calor}) &= 4/14 \\ P(S_{Normal}) &= 6/14 \\ P(S_{Frío}) &= 4/14 \\ E(S_{Calor}) &= -\frac{2}{4}\log_2\left(\frac{2}{4}\right) - \frac{2}{4}\log_2\left(\frac{2}{4}\right) \\ E(S_{Normal}) &= -\frac{4}{6}\log_2\left(\frac{4}{6}\right) - \frac{2}{6}\log_2\left(\frac{2}{6}\right) \\ E(S_{Frío}) &= -\frac{3}{4}\log_2\left(\frac{3}{4}\right) - \frac{1}{4}\log_2\left(\frac{1}{4}\right) \end{aligned}$$

Por lo tanto, $E(S,Temperatura) = (4/14) * 1.0 + (6/14) * 0.918 + (4/14) * 0.811 = 0.911$.

Para el atributo Temperatura, la ganancia de información será $G(S,Temperatura) = 0.94 - 0.911 = 0.029$.

Para el atributo Humedad, que tiene dos posibles valores $X = \{Alta, Normal\}$, $E(S,Humedad)$ se calculará como se muestra a continuación:

$$\begin{aligned} E(S,Humedad) &= P(S_{Alta})E(S_{Alta}) + \\ &+ P(S_{Normal})E(S_{Normal}) \end{aligned}$$

Cada uno de esos valores se calcula como sigue, considerando que S_{Alta} y S_{Normal} tienen 7 muestras cada uno:

$$\begin{aligned} P(S_{Alta}) &= 7/14 \\ P(S_{Normal}) &= 7/14 \\ E(S_{Alta}) &= -\frac{3}{7}\log_2\left(\frac{3}{7}\right) - \frac{4}{7}\log_2\left(\frac{4}{7}\right) \\ E(S_{Normal}) &= -\frac{6}{7}\log_2\left(\frac{6}{7}\right) - \frac{1}{7}\log_2\left(\frac{1}{7}\right) \end{aligned}$$

Por lo tanto, $E(S,Humedad) = (7/14) * 0.985 + (7/14) * 0.592 = 0.788$.

Para el atributo Humedad, la ganancia de información será $G(S,Humedad) = 0.94 - 0.788 = 0.152$.

Por último, para el atributo Viento, que tiene dos posibles valores $X = \{Si,No\}$, $E(S,Viento)$, se calculará como se muestra a continuación:

$$\begin{aligned} E(S,Viento) &= P(S_{Si})E(S_{Si}) + \\ &+ P(S_{No})E(S_{No}) \end{aligned}$$

Cada uno de esos valores se calcula como sigue, considerando que S_{Si} y S_{No} tienen 6 y 8 muestras, respectivamente:

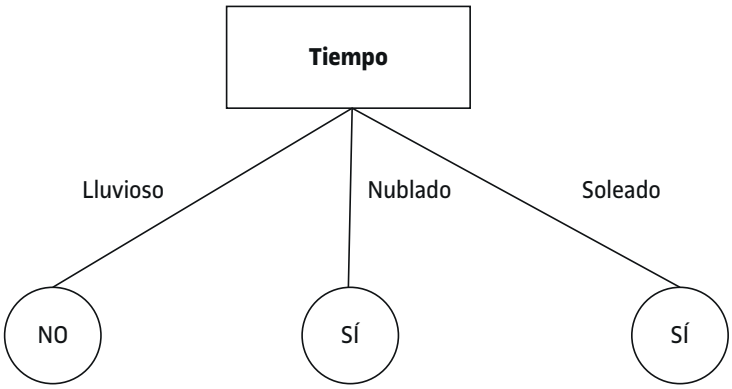
$$\begin{aligned} P(S_{Si}) &= 6/14 \\ P(S_{No}) &= 8/14 \\ E(S_{Si}) &= -\frac{3}{6}\log_2\left(\frac{3}{6}\right) - \frac{3}{6}\log_2\left(\frac{3}{6}\right) \\ E(S_{No}) &= -\frac{6}{8}\log_2\left(\frac{6}{8}\right) - \frac{2}{8}\log_2\left(\frac{2}{8}\right) \end{aligned}$$

Por lo tanto, $E(S,Viento) = (6/14) * 1 + (8/14) * 0.811 = 0.892$.

Para el atributo Viento, la ganancia de información será $G(S,Viento) = 0.94 - 0.892 = 0.048$.

Como el objetivo es quedarse con el atributo que proporciona mayor ganancia de información, la primera división del árbol la realizaríamos con el atributo Tiempo. El árbol resultante se muestra en la figura 4.

Figura 4. Árbol de decisión tras la primera iteración del algoritmo ID3 para los datos mostrados en la tabla 2



Si el proceso de creación del árbol finalizara tras este primer paso, concluiríamos que, si el tiempo es soleado, se jugaría al golf con una certeza 3/5. Si el tiempo es lluvioso, no se jugaría con una certeza de 3/5 (1 – 2/5). Por último, si el tiempo es *Nublado*, hay una certeza del 100 % de que se jugaría al golf.

Para continuar el proceso, hay que seleccionar una rama no terminal y estudiar qué atributo podría usarse para continuar subdividiendo. En el caso de la rama de la izquierda (Tiempo = *Lluvioso*), tendría que repetirse el proceso usando únicamente las muestras que cumplen con ese requisito, que son las que figuran en la tabla 3. Ahora, tendría que estudiarse cuál es el siguiente atributo que seleccionar para dividir entre Temperatura, Humedad y Viento. Este caso es más fácil, ya que puede comprobarse que el mejor atributo es Humedad, puesto que se obtienen nodos terminales.

Tabla 3. Datos de ejemplo sobre si podrá o no jugarse a golf cuando el atributo Tiempo es *Lluvioso*.

Tiempo	Temperatura	Humedad	Viento	¿Se jugará?
Lluvioso	Calor	Alta	No	No
Lluvioso	Calor	Alta	Sí	No
Lluvioso	Normal	Alta	No	No
Lluvioso	Frío	Normal	No	Sí
Lluvioso	Normal	Normal	Sí	Sí

En el caso de la rama de la derecha, se usarán únicamente las muestras con el atributo Tiempo igual a *Soleado* que aparecen en la tabla 4. En este caso, el mejor atributo será Viento, pues se obtienen nodos terminales.

Tabla 4. Datos de ejemplo sobre si podrá o no jugarse a golf cuando el atributo Tiempo es *Soleado*.

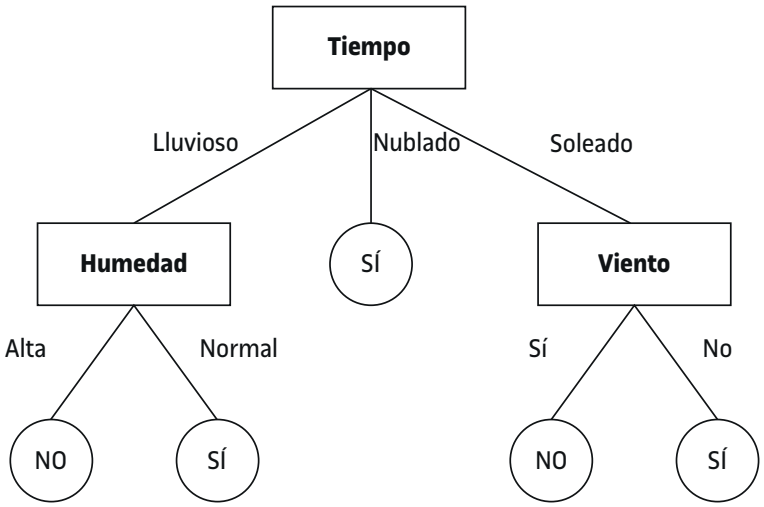
Tiempo	Temperatura	Humedad	Viento	¿Se jugará?
Soleado	Normal	Alta	No	Sí
Soleado	Frío	Normal	No	Sí
Soleado	Frío	Normal	Sí	No
Soleado	Normal	Normal	No	Sí
Soleado	Normal	Alta	Sí	No

El árbol resultante final será el que se muestra en la figura 5. Tal como puede comprobarse, no ha sido necesario usar el atributo Temperatura para crear el árbol. A partir del árbol resultante, es muy sencillo obtener un conjunto de reglas que definen el problema. Dichas reglas nos dicen que se jugará al golf si se cumple alguna de las siguientes condiciones:

- El tiempo es normal.
- El tiempo es lluvioso y la humedad es normal.
- El tiempo es soleado y no hay viento.

En cualquier otro caso, no se jugará al golf.

Figura 5. Árbol de decisión resultante de aplicar el algoritmo ID3 a los datos mostrados en la tabla 2.



2.2. Poda de árboles

Si el número de atributos es muy grande, el árbol resultante puede tener muchísimas ramas, de forma que no sea tan fácil extraer un conjunto de reglas que proporcionen una información muy fácil de interpretar. Además, llegar a tener un árbol con todos los nodos terminales puede comportar que el modelo resultante esté sobreentrenado (*overfitting*, en inglés). El problema es que el árbol creado será muy específico para el conjunto de entrenamiento y seguramente cometerá muchos errores para datos nuevos.

Para resolver este problema, existe una segunda fase en el proceso de creación del árbol llamada *poda*. La poda consiste en eliminar las hojas del árbol que causan sobreentrenamiento. Para ello, se calcula cuál es la partición que aporta una menor ratio entre el incremento de profundidad media del árbol y el decremento del error global de clasificación. Es decir, se eliminan las hojas que menos ayudan a mejorar el árbol durante el proceso de creación. Esta fase se lleva a cabo una vez que se ha creado el árbol.

Otros modelos de clasificación supervisada

Existen muchos métodos de clasificación supervisada. Algunos de los más importantes son la regresión logística (que, a pesar del nombre, es un método de clasificación y no de regresión), las máquinas de vectores de soporte (*support vector machines* o SVM, en inglés) y las redes neuronales. Las redes neuronales están en auge gracias a las técnicas de aprendizaje profundo (*deep learning*, en inglés), que están obteniendo resultados excelentes en problemas con un número enorme de muestras. El estudio de estas técnicas está fuera de los objetivos de este módulo.

3. Modelos de regresión

Prácticamente todos los métodos de clasificación tienen variantes para ser aplicados en problemas de regresión tanto lineales como no lineales. Por ejemplo, existen algoritmos que construyen árboles de decisión para problemas de regresión.

En este apartado, se explicará cómo se resuelve un problema de regresión lineal sencillo usando la técnica de los mínimos cuadrados generalizados. Para problemas más complejos, incluidos los no lineales, es conveniente probar diferentes métodos para comprobar cuál es el que obtiene los mejores resultados.

Supongamos que queremos predecir el valor de una vivienda a partir de la superficie de esta. La variable superficie es la variable independiente o x. La variable precio es la variable dependiente o y. Se asume que existe una relación lineal entre ambas variables. La relación lineal se explica con esta ecuación: $y = mx + n$. Por lo tanto, el problema consiste en, dados un conjunto de muestras de entrenamiento x y el valor de la clase objetivo para cada muestra y, encontrar los parámetros de la recta m y n que mejor se ajusten a dichos datos de entrada. Es un problema supervisado, puesto que para obtener los parámetros de la recta, es necesario conocer *a priori* el valor de la variable objetivo (precio de venta). Es un problema de regresión y no de clasificación, puesto que la variable objetivo es continua.

Una vez obtenidos los parámetros de la recta, dada una observación nueva de la superficie, solo tenemos que aplicar la ecuación $y = mx + n$ para predecir el precio de la vivienda.

Este problema es un problema sobredeterminado, pues se tienen muchas más ecuaciones $y_i = mx_i + n$ (una por cada muestra de entrada) que incógnitas (m y n). Este tipo de problemas se resuelven con el método de mínimos cuadrados generalizados, y se obtiene:

$$m = \frac{S_{xy}}{S_x^2} \tag{7}$$

$$n = \bar{y} - m\bar{x} \tag{8}$$

donde $\bar{x}$ y $\bar{y}$ son las medias muestrales de x e y, respectivamente; S_x^2 es la varianza muestral de x, y S_{xy} es la covarianza muestral entre x e y. Estas cantidades pueden calcularse como se muestra a continuación:

$$\bar{x} = \frac{\sum_{i=1}^{N} x_i}{N} \quad (9)$$

$$\bar{y} = \frac{\sum_{i=1}^{N} y_i}{N} \quad (10)$$

$$S_x^2 = \frac{\sum_{i=1}^{N} (x_i - \bar{x})^2}{N} \quad (11)$$

$$S_y^2 = \frac{\sum_{i=1}^{N} (y_i - \bar{y})^2}{N} \quad (12)$$

$$S_{xy} = \frac{\sum_{i=1}^{N} (x_i - \bar{x})(y_i - \bar{y})}{N} \quad (13)$$

donde N es el número de muestras del conjunto de entrenamiento.

Supongamos que tenemos los datos mostrados en la tabla 5, donde $N = 12$, x es la superficie en metros cuadrados e y es el precio en miles de euros. Las cantidades anteriores serán, para este ejemplo:

$$\bar{x} = 149.33 \quad (14)$$

$$\bar{y} = 130.08 \quad (15)$$

$$S_x^2 = 1143.52 \quad (16)$$

$$S_y^2 = 1203.36 \quad (17)$$

$$S_{xy} = 1107.88 \quad (18)$$

Tabla 5. Datos de ejemplo para realizar una regresión lineal.

Superficie	Precio	Superficie	Precio
100	90	160	127
105	95	165	140
110	96	172	138
120	102	180	155
140	115	185	180
155	123	200	200

Por lo tanto, los parámetros de la recta serán $m = 0.97$ y $n = -14.6$. Ahora, si queremos predecir el valor de una nueva vivienda sabiendo que la superficie es 150, aplicaremos el modelo de regresión, y obtendremos $y = mx + n = 0.97 * 150 - 14.6 = 130.73$.

La figura 6 presenta las muestras de la tabla 5 con círculos negros y la recta resultante. El triángulo azul es la muestra nueva.

Figura 6. Recta de regresión resultante teniendo como entrada los datos de la tabla 5.