Teoría de la Información

García Prado, Sergio

24 de febrero de 2017

Resumen

En este documento se ha examinado el algoritmo básico de generación de árboles de decisisión a partir de heurísticas basadas en la teoría de información ID3. Para ello se ha realizado una descripción y una simulación paso a paso de su ejecución. Además, se ha realizado un caso práctico de discretización de atributos. Por último, se ha comprobado que el algoritmo J48 tan solo discretiza valores si la ganancia de información obtenida es mayor tras la partición que la del resto de atributos del conjunto de datos.

4. Construir el árbol de decisión según el algoritmo *ID3*, calculando todas las ganancias a la hora de escoger el siguiente atributo

Outlook	Temperature	Humidity	Wind	PlayTennis
Sunny	Hot	High	Weak	NO
Sunny	Hot	High	Strong	NO
Overcast	Hot	High	Weak	YES
Rain	Mild	High	Weak	YES
Rain	Cool	Normal	Weak	YES
Rain	Cool	Normal	Strong	NO
Overcast	Cool	Normal	Strong	YES
Sunny	Mild	High	Weak	NO
Sunny	Cool	Normal	Weak	YES
Rain	Mild	Normal	Weak	YES
Sunny	Mild	Normal	Strong	YES
Overcast	Mild	High	Strong	YES
Overcast	Hot	Normal	Weak	YES
Rain	Mild	High	Strong	NO

Tabla 1: *Datos para el algoritmo* ID3

El algoritmo *ID3* consiste en un generador de árboles de decisión. Los árboles de decisión son estructuras jerárquicas utilizadas para resolver problemas de clasificación mediante técnicas de aprendizaje automático. En este caso, el algoritmo se basa en aprendizaje supervisado, es decir, en el periodo de aprendizaje utiliza un conjunto de datos en el cuál el valor de destino de la clasificación está fijado previamente.

El aprendizaje basado en árboles de decisión utiliza la generación de estructuras en forma de árbol etiquetado. Un árbol etiquetado consiste en un grafo G no dirigido que posee las siguientes propiedades[3]:

- G es conexo y no tiene ciclos.
- G no tiene ciclos y, si se añade alguna arista se forma un ciclo.
- G es conexo y si se le quita alguna arista deja de ser conexo.
- G es conexo y el grafo completo de 3 vértices K_3 no es un menor de G.
- Dos vértices cualquiera de G están conectados por un único camino simple.
- Cada arista posee una etiqueta con la cuál se identifica.

En el contexto del aprendizaje, dichos árboles representan lo siguiente:

- Los vértices internos representan atributos de los datos
- El proceso de clasificación comienza por el vértice padre del arbol (vértice sin padre).
- Las aristas etiquetadas representan el valor que toma el atributo, de tal manera que si el dato toma el valor fijado por la etiqueta de la arista en el atributo fijado en el vértice padre, entonces el próximo atributo a inspeccionar es el que representa el vértice hijo al cual se llega a partir de dicha arista.
- Los vértices hoja (vértices sin hijos) representa los valores que toman el valor de la clase en la cual será clasificado el dato.

Para la generación del árbol de decisión existen varias heurísticas, en el caso del algoritmo *ID3* la intuición utilizada se basa en la disciplina de la teoría de información, que a grandes rasgos consiste en buscar recursivamente el atributo que mayor relación tenga con el valor esperado de la clase para dicho dato. Además se apoya en los conceptos de **Información**, **Entropía** y **Ganancia de Información**, los cuales se describen a continuación: Sea $S = \{S_1, S_2, S_q\}$ el conjunto de valores discretos que puede tomar un determinado atributo y $P(S_i)$ la probabilidad de ocurrencia de dicho valor.

$$I(S_i) = \log \frac{1}{P(S_i)} \tag{1}$$

En la equación (1) se describe la cantidad de información que proporciona un determinado valor S_i del conjunto S de posibles valores que puede tomar el dato.

$$H(S) = \sum_{i=1}^{q} P(S_i) I(S_i) \tag{2}$$

La equación (2) describe la entropia de un atributo, esto puede traducirse como una medida de la cantidad de infomación que se puede obtener de un atributo S. La idea en que se basa esto consiste en la intuición de que si el conjunto de valores está uniformemente distribuido en su rango, la entropía del mismo tomará valores más altos, lo cuál hará que probablemente ese atributo sea más valioso para la clasificación si este está relacionado con la clase como se describe a continuación. En cambio, si la distribución de valores que toma el atributo está muy concentrada, probablemente este atributo suministrará un menor grado de información por lo que su entropía es menor.

$$G(S, A) = H(S) - \sum_{i=1}^{k} \frac{\mid S_{v_i} \mid}{\mid S \mid} H(S_{v_i}) \tag{3}$$

La ganancia de información se define en la equación (3). Esta es la medida que utiliza la heurística en que se basa el algoritmo de generación de árboles *ID3*. Dicha equación proporciona el grado de relación entre un atributo A del conjunto de datos y la clase S en que se desean clasificar los mismos.

Algoritmo 1 ID3 *utiliza la equación* (3) *para generar un ránking con los atributos del conjunto de datos, seleccinando el de mayor ganancia respecto de la clase como vértice padre y creando tantas aristas como valores no vaciós tome dicho atributo. Repite este proceso por cada una de las aristas de forma recursiva eliminando el atributo selecionado y filtrando los datos de entrenamiento a las especificaciones fijadas por el camino seguido, hasta que todos los datos toman un único valor, en cuyo caso crea un vértice hoja con dicho valor, la disyunción de valores de la*

clase si se eliminan todos los atributos o desconocido *si no hay datos suficientes para continuar el proceso*

Para el cálculo de resultados se ha realizado una implementación en el lenguaje *Python* la cual calcula el ranking citado anteriormente. Dicha implementación se puede visualizar a través de la ruta `https://github.com/garciparedes/python-examples/blob/master/machine_learning/utils/gainranking.py`[2]. Además, se ha realizado una implementación básica de la generación del árbol de decisión mediante *ID3* la cual se puede visualizar en `https://github.com/garciparedes/python-examples/blob/master/machine_learning/decision_tree/id3.py`[2]

En este caso, el conjunto de datos se muestra en la tabla 1. Está formado por un conjunto de 4 atributos, además del valor esperado de la clase. Todos ellos toman valores discretos, por lo que no es necesario realizar discretizaciones, lo cual se ajusta perfectamente a la entrada esperada del Algoritmo *ID3*. La clase se corresponde con el atributo **PlayTennis**. A continuación se describe una ejecución paso a paso del algoritmo.

Atributo	Ganancia
Outlook	0.246750
Humidity	0.151836
Wind	0.048127
Temperature	0.029223

Tabla 2: *ID3 sobre el conjunto de datos: Paso 1*

Atributo	Ganancia
Humidity	0.970951
Temperature	0.570951
Wind	0.019973

Tabla 3: *ID3 sobre el conjunto de datos: Paso 2*

Atributo	Ganancia
Wind	0.970951
Humidity	0.019973
Temperature	0.019973

Tabla 4: *ID3 sobre el conjunto de datos: Paso 3*

El en las tablas 2, 3 y 4 se muestran los resultados de los rankings de ganancia en cada paso del algoritmo *ID3* implementado en *Python* tal y como se cita anteriormente. El árbol resultante de la ejecución coincide con el presentado en las diapositivas de la asignatura [1] y se muestra en la figura 1.

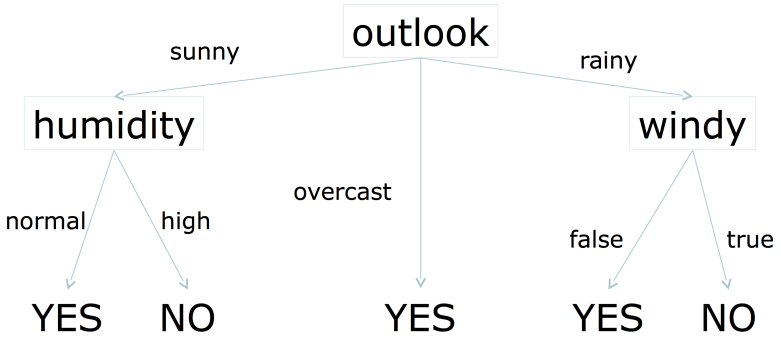

Figura 1: *Árbol resultante tras ejecutar el algoritmo* ID3 *[1]*

5. Calcular las siguientes ganancias en este ejemplo de 6 muestras

Presión	40	48	60	72	80	90
Clase	+	+	-	-	-	+

Tabla 5: *Datos para cálculo de ganancias*

Se pide calcular las ganancias resultantes trás discretizar el atributo *Presión* de la tabla 5 mediante su particionado en los puntos $c_1 = 54$ y $c_2 = 85$. Denotaremos por S al atributo de la clase. Lo siguiente es calcular su entropía tal y como se ha hecho en la equación (4)

$$H(S) = \frac{2}{6}\log(\frac{6}{2}) + \frac{4}{6}\log(\frac{6}{4}) = 0{,}636514... \tag{4}$$

A continuación se muestra el número de instancias asociadas a cada clase para después calcular la ganancia de información obtenida tras dicha discretización a través de las equaciones (5), (6) y (7).

- $A_{c_1} = \{+, -\}$
- $A_{c_1<} \leftarrow \{2+, 0-\}$
- $A_{c_1>} \leftarrow \{1+, 3-\}$

$$H(A_{c_1<}) = \frac{2}{2}\log(\frac{2}{2}) + \frac{0}{2}\log(\frac{2}{0}) = 0{,}0 \tag{5}$$

$$H(A_{c_1>}) = \frac{1}{4}\log(\frac{4}{1}) + \frac{3}{4}\log(\frac{4}{3}) = 0{,}562335... \tag{6}$$

$$G(S, A_{c_1}) = 0{,}636514... - (\frac{2}{6} * 0{,}0 + \frac{4}{6} * 0{,}562335...) = 0{,}261624... \tag{7}$$

A continuación se muestra el número de instancias asociadas a cada clase para después calcular la ganancia de información obtenida tras dicha discretización a través de las equaciones (8), (9) y (10):

- $A_{c_2} = \{+, -\}$

- $A_{c_2<} \leftarrow \{2+, 3-\}$
- $A_{c_2>} \leftarrow \{1+, 0-\}$

$$H(A_{c_2<}) = \frac{2}{5}\log(\frac{5}{2}) + \frac{3}{5}\log(\frac{5}{3}) = 0{,}673011... \tag{8}$$

$$H(A_{c_2>}) = \frac{1}{1}\log(\frac{1}{1}) + \frac{0}{1}\log(\frac{1}{0}) = 0{,}0 \tag{9}$$

$$G(S, A_{c_2}) = 0{,}636514... - (\frac{5}{6} * 0{,}673011... + \frac{1}{6} * 0{,}0) = 0{,}0756715... \tag{10}$$

Puesto que la **Ganancia de Información** obtenida tras la partición por $c_1 = 54$ es significativamente mayor a la obtenida por $c_2 = 85$ debido a que $G(S, A_{c_1}) = 0{,}261624... > G(S, A_{c_2}) = 0{,}0756715...$, el particionamiento se realiza en el punto c_1.

6. Aplicar el algoritmo J48 al ejemplo anterior modificado, en el que se detallan valores numéricos de la temperatura. Explique el resultado del árbol resultante frente a la temperatura

Outlook	Temperature	Humidity	Wind	PlayTennis
Sunny	23	High	Weak	NO
Sunny	24	High	Strong	NO
Overcast	25	High	Weak	YES
Rain	15	High	Weak	YES
Rain	5	Normal	Weak	YES
Rain	4	Normal	Strong	NO
Overcast	3	Normal	Strong	YES
Sunny	12	High	Weak	NO
Sunny	7	Normal	Weak	YES
Rain	14	Normal	Weak	YES
Sunny	16	Normal	Strong	YES
Overcast	17	High	Strong	YES
Overcast	26	Normal	Weak	YES
Rain	13	High	Strong	NO

Tabla 6: *Datos para el algoritmo J48*

El resultado de aplicar el algoritmo J48 al conjunto de datos de la tabla 6 se muestra en la figura 2, que tal y como se puede apreciar, es el mismo que el del ejercicio 4. El motivo por el cual no ha utilizado ninguna posible partición sobre el atributo numérico *temperatura* es debido a que la ganancia de información que este proporciona no supera la del resto de atributos del conjunto de datos. Suponemos que la transformación a valores numéricos de *temperatura* sigue una distribución aproximadamente similar a la indicada mediante valores discretos en la tabla 1, por lo que los resultados de la ganancia de información seria similar a las obtenidas en dicho ejercicio. Puesto que el algoritmo *J48* es una implementación mejorada de *ID3*, en los casos en que no necesite aplicar dichas mejoras como discretización de atributos o podado de ramas, el resultado que obtiene es el mismo, tal y como a sucedido en esta ocasión.

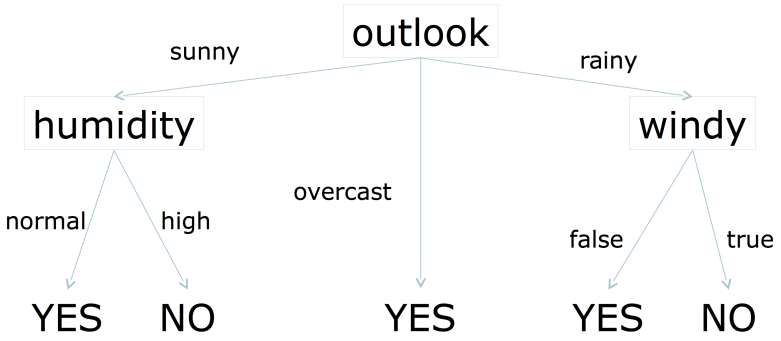

Figura 2: *Árbol resultante tras ejecutar el algoritmo* J48 *[1]*

Referencias

[1] Calonge Cano, T., and Alonso GonzáLez, C. J. Técnicas de Aprendizaje Autómatico, 2016/17.

[2] García Prado, S. Python Examples.

[3] Wikipedia. Tree (graph theory), 2017.