Introducción a Weka

García Prado, Sergio

27 de febrero de 2017

Resumen

Abstract

1. Tómese la siguiente función lógica y obténgase el árbol de decisión correspondiente usando WEKA

$$\neg(A \wedge B) \vee \neg(C \wedge D) \oplus E \tag{1}$$

En esta práctica se ha generado la tabla de verdad de la función descrita en la ecuación (1). Por lo tanto, el conjunto de datos que se ha utilizado en este caso sigue la siguiente estructura: *a*) los atributos de cada dato se corresponden con una determinada combinación de las variables de entrada de (1), *b*) mientras que el valor de la clase será el valor de verdad de dicha combinación.

El conjunto de datos está compuesto por tanto, por 5 atributos de tipo booleano (discretos) al igual que la clase, que también es de tipo booleano. Puesto que la función está formada por 5 argumentos y cada uno de ellos puede tomar 2 valores, nuestro conjunto de datos está formado por $2^5 = 32$ datos. Todos ellos han sido recogidos en la tabla 1. Dicha tabla de verdad ha sido generada a partir de la clase *LogicDataSet* codificada en el lenguaje *Python*, a la cual se puede acceder a partir de `https://github.com/garciparedes/python-examples/blob/master/data_sets/logic_data_set.py`[2].

El propósito de esta práctica es analizar el comportamiento de las estrategias de aprendizaje automático básadas en estructuras jerárquicas (árboles). Para ello se ha utilizado la herramienta *Weka*, que permite la realización de diversas tareas relacionadas con el aprendizaje automático de manera simple y de manera gráfica sobre conjuntos de datos. En este caso se ha realizado una comparativa entre los algoritmos de clasificación basada en árboles mediante aprendizaje supervisado.

Debido a la naturaleza intrínseca de un estructura en forma de árbol, esto permiten la representación de funciones lógicas con un alto grado de acierto. Para comprobar dicha cualidad, se ha realizado un ánalisis de resultados de los algoritmos *ID3* y *J48*, cuyas diferencias son las siguientes:

- **ID3**: Es el algoritmo básico de generación de árboles de decisión a partir de un conjunto de datos formado tanto por atributos como valores de clase de carácter discreto. Utiliza heurísticas basadas en la Teoría de Información. Debido a sus simplicidad, está muy condicionado al conjunto de datos de entrenamiento.
- **J48**: Es la versión implementada en el lenguaje *Java* del algoritmo **C4.5**, una versión con numerosas mejoras respecto de *ID3*, tales como: *a*) la capacidad de procesar atributos continuos mediante la generación, *b*) manejo de valores desconocidos, *c*) mismos atributos de entrada para distintos valores para la clase de destino y *d*) poda para tratar de evitar el sobreajuste.

Para los casos de prueba, en los dos casos se han utilizado los parámetros por defecto de la herramienta *Weka*. Tras utilizar todo el conjunto de datos tanto para entrenamiento como para test, los árboles generados por el algoritmo **ID3** y **J48** se muestran en las figuras 1 y 2. Dichas figuras han sido generadas a partir de la herramienta *treetograph*[3].

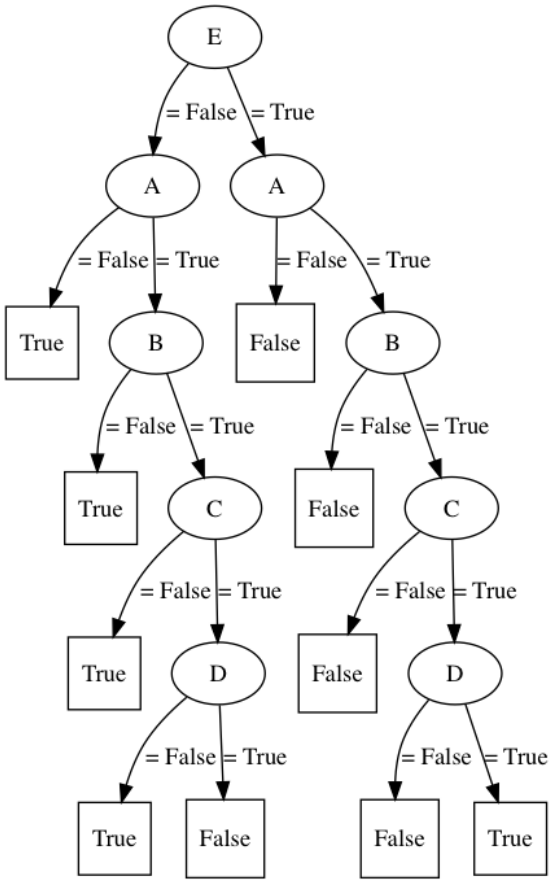

Figura 1: *Árbol de decisión generado a partir del algoritmo* ID3

A	B	C	D	E	Result
False	False	False	False	False	True
True	False	False	False	False	True
False	True	False	False	False	True
True	True	False	False	False	True
False	False	True	False	False	True
True	False	True	False	False	True
False	True	True	False	False	True
True	True	True	False	False	True
False	False	False	True	False	True
True	False	False	True	False	True
False	True	False	True	False	True
True	True	False	True	False	True
False	False	True	True	False	True
True	False	True	True	False	True
False	True	True	True	False	True
True	True	True	True	False	False
False	False	False	False	True	False
True	False	False	False	True	False
False	True	False	False	True	False
True	True	False	False	True	False
False	False	True	False	True	False
True	False	True	False	True	False
False	True	True	False	True	False
True	True	True	False	True	False
False	False	False	True	True	False
True	False	False	True	True	False
False	True	False	True	True	False
True	True	False	True	True	False
False	False	True	True	True	False
True	False	True	True	True	False
False	True	True	True	True	False
True	True	True	True	True	True

Tabla 1: *Tabla de verdad de la ecuación 1*

h

		Valor Real		
		Positivo	Negativo	p_j
Valor Predicho	Positivo	15	1	0,9375
	Negativo	1	15	0,9375
	π_j	0,5	0,5	$N = 32$

Tabla 2: *Resultados J48*

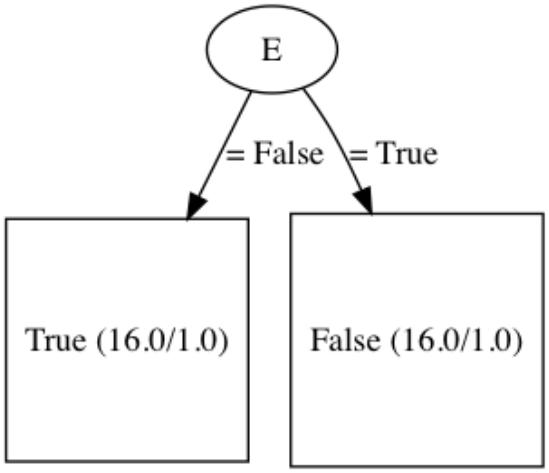

Figura 2: *Árbol de decisión generado a partir del algoritmo* J48

h

		Valor Real		
		Positivo	Negativo	p_j
Valor Predicho	Positivo	16	0	1
	Negativo	0	16	1
	π_j	0,5	0,5	$N = 32$

Tabla 3: *Resultados ID3*

Referencias

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

[2] García Prado, S. Python Examples. `https://github.com/garciparedes/python-examples`.

[3] Taboada Rodero, I. J. treetograph. `https://github.com/ismtabo/treetograph`.