

# INTELIGENCIA ARTIFICIAL

## PRÁCTICA 11



GRUPO 3:  
Enrique Ballesteros Horcajo  
Ignacio Iker Prado Rujas

8 de Mayo de 2015



## Índice

1.	Apartado 1 . . . . .	1
2.	Apartado 2 . . . . .	3



## Introducción

En esta práctica vamos a trabajar con el entorno que nos proporciona WEKA<sup>1</sup>, aplicando herramientas de aprendizaje automático a conjuntos de datos dados por los ficheros `diabetes.arff` y `glass.arff`.

### 1. Apartado 1

Durante todo este apartado los datos han sido tomados de `diabetes.arff` incluido en el subdirectorio `data` de la distribución de WEKA, en el campus virtual. Este dataset contiene los resultados de un estudio relacionado con la diabetes.

- *Analiza el archivo citado y contesta las siguientes preguntas: ¿Cuántas instancias hay? ¿Cuántos atributos se utilizan? ¿Por qué atributo queremos aprender a clasificar?*  
Hay 768 instancias. Se utilizan nueve atributos, ocho como datos (edad, peso, etc.) y uno para clasificar (ser o no diabético). Queremos aprender a clasificar por este último.
- *Ejecuta J48 (versión WEKA de C4.5). Utiliza para la validación el “training set”. Incluye en la memoria los resultados obtenidos y la representación gráfica del árbol correspondiente.*

=== Classifier model (full training set) ===						
(El árbol se muestra en una figura a parte, en la página siguiente)						
Number of Leaves : 20						
Size of the tree : 39						
=== Evaluation on training set ===						
=== Summary ===						
Correctly Classified Instances	646	84.1146 %				
Incorrectly Classified Instances	122	15.8854 %				
Kappa statistic	0.6319					
Mean absolute error	0.2383					
Root mean squared error	0.3452					
Relative absolute error	52.4339 %					
Root relative squared error	72.4207 %					
Total Number of Instances	768					
=== Detailed Accuracy By Class ===						
TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
0.936	0.336	0.839	0.936	0.885	0.888	tested_negative
0.664	0.064	0.848	0.664	0.745	0.888	tested_positive
Weight. Avg.	0.841	0.842	0.841	0.836	0.888	
=== Confusion Matrix ===						
a b <- classified as						
468 32   a = tested_negative						
90 178   b = tested_positive						

<sup>1</sup>Página web de WEKA [aquí](#).

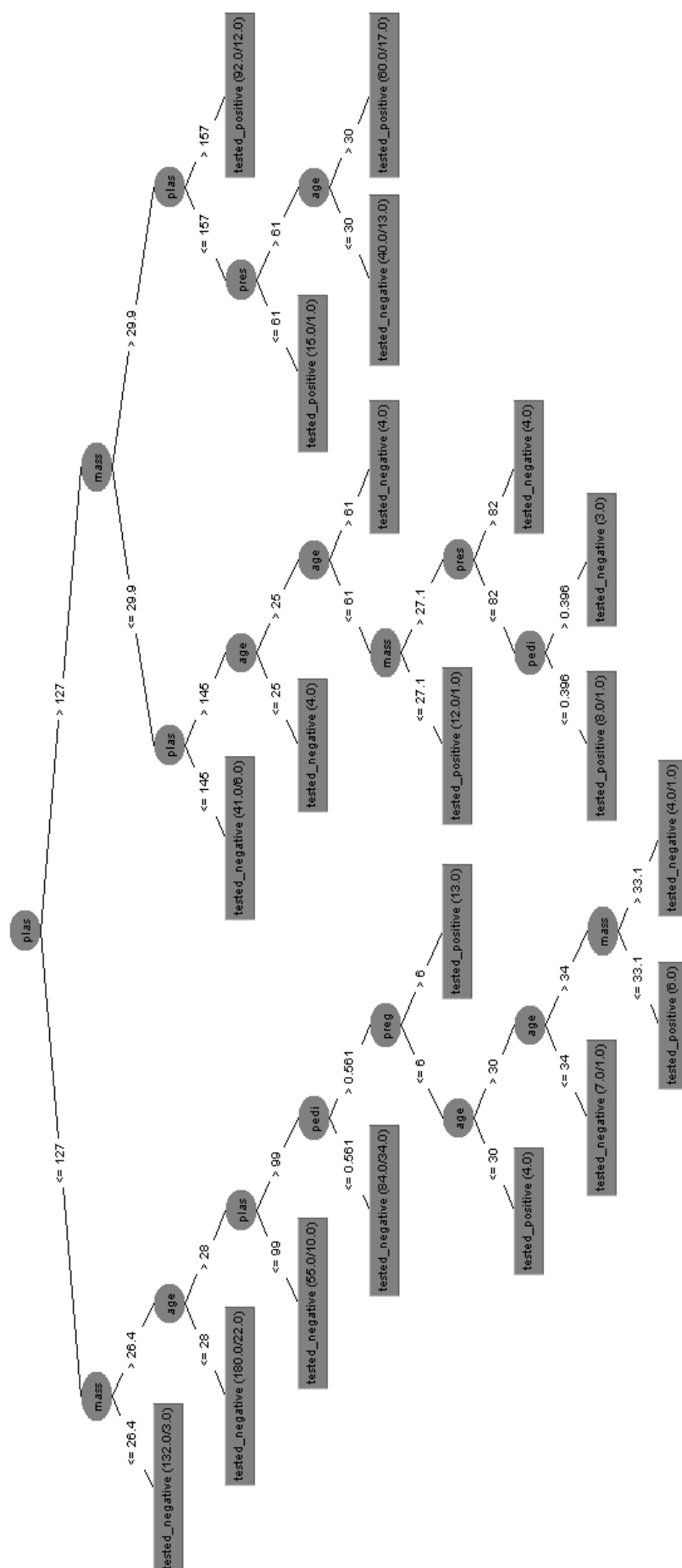


Figura 1: Gráfica del árbol correspondiente a los resultados de la ejecución de J48, con validación “training set”.

- ¿Cuántas instancias han sido mal clasificadas?

Han sido mal clasificadas 122 instancias.

- ¿Cuántos nodos terminales hay en el árbol?

Hay 20 nodos terminales en el árbol.

- ¿Por qué atributo se clasifica en el primer nivel del árbol?

Se clasifica por *plss* (plasma glucose concentration a 2 hours in an oral glucose tolerance test).

- Vuelve a ejecutar *J48* primero con “cross-validation” y después con “percentage split” (con un valor del 66 %) y observa las diferencias. Haz una tabla con el porcentaje de instancias bien clasificadas en cada uno de los tres métodos de validación. Incluye en esa tabla la precisión y el recall para cada uno de los tres métodos.

	Training set	Cross-validation	Percentage split
Porcentaje bien clasificadas	84.1146 %	73.8281 %	76.2452 %
Recall	0.841	0.738	0.762
Precisión	0.842	0.735	0.756
Precisión con negativos	0.839	0.790	0.809
Precisión con positivos	0.848	0.632	0.644

- Comenta los resultados obtenidos.

El porcentaje de bien clasificados es mejor en el primer método, sin embargo hay que explicar por qué es así. El training set crea la clasificación a partir de todos los ejemplos de entrenamiento y luego la prueba sobre los mismos ejemplos, por tanto siempre será el mejor método si se aplica sobre esos ejemplos de entrenamiento. Por ello nos fijaremos en los otros dos. El percentage split obtiene mejor porcentaje de clasificados, y también mejor precisión tanto con negativos como con positivos. Los negativos son más fiables que los positivos, por lo que los positivos deberían ir acompañados de una prueba médica que confirme los resultados.

- ¿Cuál de las tres validaciones te parece más fiable? ¿Por qué?

La cross-validation es más fiable, por lo que un 73 % puede ser un porcentaje de acierto cercano a la realidad. Esto se debe a que la cross-validation obtiene unos resultados más uniformes que no dependen del orden en que se escojan los ejemplos de entrenamiento, sino que presenta una media entre diez percentage splits. Por tanto es más fiable.

- ¿Cuántos falsos negativos (FN) y falsos positivos (FP) se han obtenido con el “percentage split”?

Con percentage split se obtienen 26 falsos negativos y 36 falsos positivos.

## 2. Apartado 2

Para este apartado se utilizará el dataset contenido en el archivo `glass.arff` incluido en el subdirectorio `data` de la distribución de WEKA y en el campus virtual. Este dataset contiene datos sobre tipos de cristal según sus características.

- Analiza el dataset. ¿En cuántas clases se clasifican los ejemplares? ¿Cuántos atributos se utilizan?

En 7 clases distintas: *building windows float processed*, *building windows non float processed*, *vehicle windows float processed*, *vehicle windows non float processed*, *containers*,

tableware y headlamps. Utilizan 9 atributos : RI (refractive index), Na (Sodium), Mg (Magnesium), Al (Aluminum), Si (Silicon), K (Potassium), Ca (Calcium), Ba (Barium) y Fe (Iron). En función de las cantidades para cada cristal se clasificará cada cristal en un tipo de los 7.

- *Aplica el clustering jerárquico a este conjunto de datos para 7 clusters y comenta los resultados obtenidos. ¿Qué clases han quedado mejor clasificadas?*

Ninguna realmente. Puede decirse que el build wind float y el vehicle wind float son las mejor clasificadas, ya que al menos ha metido todos en el mismo cluster. O también, se puede decir que el build wind non-float porque está en el cluster que lleva su nombre. Realmente es una clasificación que creemos no tiene mucha utilidad, pues mete todos en prácticamente la misma clase. Es uno de los problemas que suelen presentarse en el aprendizaje automático.

- *Discretiza los atributos utilizando un filtro supervisado. ¿Qué horquillas se han generado para los valores del Calcio? Vuelve a ejecutar el clustering jerárquico y compara los resultados con los anteriores. ¿Qué clases han quedado mejor clasificadas?*

Cuatro horquillas: De  $-\infty$  a 7.02, de 7.02 a 8.315, de 8.315 a 10.075 y de 10.075 a  $\infty$ .

Los headlamps han quedado mucho mejor clasificados, pues ahora la mayoría están en una clase propia. Los build wind non-float tienen uno más en “su clase”.

En el resto no hay un cambio significativo.

- *Con los atributos en su estado original, utiliza el algoritmo de clustering EM (Expectation Maximization) y compara con los resultados anteriores.*

Ha mejorado la clasificación de los containers, agrupándolos en una sola clase, pero ha empeorado la clasificación del resto dividiéndolos en distintos clusters.

- *¿Qué clases han quedado mejor clasificadas?*

Realmente ninguna, los containers los agrupa juntos, pero mete otros tipos de cristales en esa misma clase por lo que tampoco es una clasificación del todo buena.

- *¿Qué método de los tres anteriores te parece más útil para este ejemplo? ¿Por qué?*

El segundo puede ser mejor porque al menos te separa los headlamps y el resto. Sin embargo, ningún método acaba de resultar útil para el ejemplo porque no clasifica bien los tipos de cristales al no ser capaz de distinguir entre ellos y meterlos en la misma clase.

- *¿Podríamos haber utilizado un algoritmo de clasificación como J48?*

Para clasificar por Type desde luego, por ejemplo utilizando el 66 % de entrenamiento. Aunque clasifica correctamente solo la mitad de los cristales restantes.

- *¿Qué habríamos obtenido con los algoritmos de clustering si no tuviéramos el atributo de clase?*

Agrupar (casi) todos en una misma clase pues no es capaz de encontrar diferencias sutiles entre los distintos cristales que se dan, y de las que se obtienen las diferentes clasificaciones. No es un problema de relajar las exigencias para cada clase, sino que no sabe qué exigir para poder distinguir unas de otras.

- *¿En qué casos sólo podríamos utilizar clustering?*

Si no tuviésemos las distintas clasificaciones por tipo de cristal, sin ejemplos de entrenamiento, solo podríamos utilizar el clustering para agrupar los cristales en clases diferentes.



## Bibliografía

- [1] Russell, S.; Norvig, P,  
*Artificial Intelligence, a modern aproach.*  
New Jersey: Pearson, 2010.
- [2] Apuntes y transparencias de Inteligencia Artificial,  
Doble Grado Matemáticas - Ing. Informática, U.C.M., 2014-2015.