

# Práctica 2 SATD

Pedro Allué Tamargo (758267)      Juan José Tambo Tambo (755742)  
Jesús Villacampa Sagaste (755739)

8 de octubre de 2020

## Índice

1. Ejercicio 1	1
2. Ejercicio 2	1
3. Ejercicio 3	1
4. Ejercicio 4	2

## 1. Ejercicio 1

Para diferenciar entre los diferentes nodos de tipo *Row Filter*, se han creado dos tablas tal y como se indica en el gui3n.

- *Row Filter*: Permite filtrar elementos de la tabla a partir de una cadena o una expresi3n regular.
- *Nominal Row Filter*: Se puede filtrar manualmente a partir de los atributos de la tabla o mediante expresiones regulares.
- *Reference Row Filter*: Permite filtrar el contenido de una tabla a partir de una segunda tabla. Para ello, se selecciona la columna deseada de cada tabla para realizar *Join* entre ambas. Tambi3n permite *Join* con la condici3n inversa (*exclude*).

## 2. Ejercicio 2

Para verificar que la columna *R3nking* es de tipo *String*, se selecciona esa columna desde la pre visualizaci3n del fichero y se escoge el tipo de dato correspondiente. Desde ah3 tambi3n se puede modificar el nombre de la columna. Para poder eliminar los comentarios, se deben de modificar las columnas correspondientes a3nadiendo `'//'` al principio de las mismas para que sean de tipo comentario *Java*. Desde la previsualizaci3n de *Knime*, se selecciona la casilla *Java-Style comments* para poder ignorar los mismos.

La columna *class* se puede eliminar a3nadiendo un elemento de *Column Filter* con el cual se introduce esta columna en el apartado de *exclude*.

Si se desea escribir un nuevo fichero CSV, se debe a3nadir un nuevo componente *CSV writer*. En configuraci3n del m3dulo se selecciona el tipo de separador, la ruta donde debe escribir el archivo y que se incluya la cabecera con la opci3n *write column header*.

## 3. Ejercicio 3

En este ejercicio, la modificaci3n del nombre de la columna "marcas" se ha realizado a3nadiendo un nodo *column rename* a continuaci3n de *CSV reader* (mediante el cual se lee el archivo "data1Nuevo.csv").

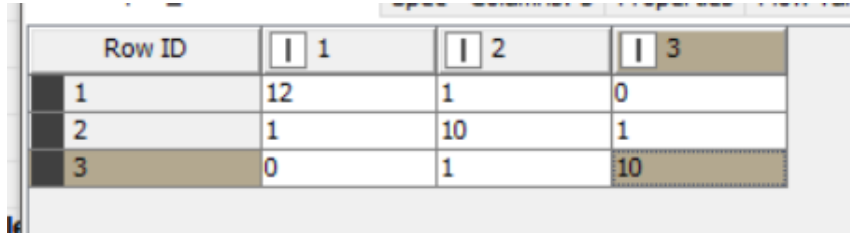
Si se desea filtrar las filas con el campo `comments = .average`, se debe a3nadir un nuevo nodo *Nominal Value Row Filter*. Se selecciona la columna *comments* y en la secci3n de *include* se busca el valor *average*. La columna *ranking* se elimina con el nodo *column filter*.

Para la escritura final, se introduce un nodo *CSV writer* y se configura con el modo *append* (se debe indicar que el archivo ya existe), y tabulador como car3cter delimitador.

## 4. Ejercicio 4

Basándonos en el ejemplo de árbol de decisiones de *Knime*, se ha creado un *Workflow* de árbol de decisión para clasificar las muestras de vinos. Para leer el archivo se utiliza el nodo *File Reader*. Se añade un nodo *Partitioning* con el que se selecciona la cantidad de datos de entrenamiento (el 80%). Seguidamente se añade un nodo *Decision Tree Learner*. Para evitar errores con este nodo, se debe modificar la columna *class* desde el nodo lector de archivos e indicar que es de tipo *String*. Posteriormente se incorpora un nodo *Decision Tree Predictor* el cual "predice" los valores.

Por último, se añade el nodo *Scorer* para poder observar las predicciones realizadas. Los resultados obtenidos son los siguientes:



Row ID	1	2	3
1	12	1	0
2	1	10	1
3	0	1	10

Figura 1: Matriz de confusión del árbol de decisión entrenado con el 80 % de los datos.

El siguiente cuadro (1) muestra los resultados de ejecutar el *workflow* con un volumen de datos al 80 % con 10 iteraciones:

Clase	Positivos confirmados	Falsos negativos
1	0.91221	0.08778
2	0.93633	0.06366
3	0.94682	0.05317

Cuadro 1: Cuadro de datos comparativos para el caso de entrenamiento con el 50 % de los datos. Los resultados están en tanto por 1.

Si se entrena el árbol de decisión con un volumen de datos inferior al 80 % se obtienen los siguientes resultados (cuadros 2 y 3):

Clase	Positivos confirmados	Falsos negativos
1	0.92442	0.07557
2	0.85028	0.14971
3	0.89869	0.10130

Cuadro 2: Cuadro de datos comparativos para el caso de entrenamiento con el 50 % de los datos. Los resultados están en tanto por 1.

Clase	Positivos confirmados	Falsos negativos
1	0.94393	0.05606
2	0.81067	0.18932
3	0.79701	0.20298

Cuadro 3: Cuadro de datos comparativos para el caso de entrenamiento con el 20 % de los datos. Los resultados están en tanto por 1.

Por lo tanto, al compararse estos datos (cuadros 2 y 3) con los obtenidos en una ejecución entrenada con el 80 % de los datos se llega a la conclusión de que con el 80 % de los datos se obtienen mejores resultados, por encima del 90 % en identificación de positivos y con un fallo inferior al 10 % en falsos negativos.