Práctica 3: Integración de *KNIME* y *WEKA*

Sistemas de Ayuda a la Toma de Decisiones

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

3 de noviembre de 2020

Índice

1. Ejercicio 1

Se ha creado el *workflow* ilustrado en la Figura 1 para trabajar con los datos del conjunto de datos *yellow-small.data*.
Se ha utilizado un nodo *Rule Engine* para crear una nueva columna *"class"*. El contenido de este nodo son las siguientes reglas:

```
$Color$ MATCHES "YELLOW" AND $Size$ MATCHES "SMALL" => "inflated"
TRUE => "not inflated"
```

Tras este nodo se ha utilizado un nodo *String manipulation* para concatenar los valores de las columnas *class* e *inflated (true/false)* utilizando la expresión:

```
string($class$ + " is " + $Inflated (True/False)$)
```

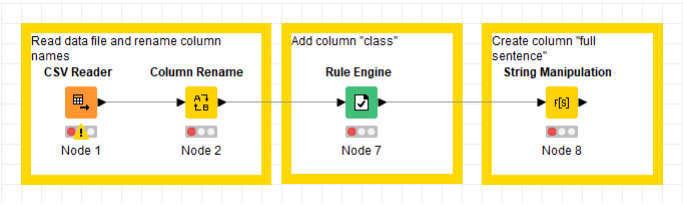

Figura 1: Workflow del ejercicio 1

2. Ejercicio 2

Se va a proceder a analizar un conjunto de datos que describe el número de visitantes de un sitio web en los meses de junio/julio de 2010 (fichero *website1.txt*).

Para calcular los parámetros de media, desviación típica, Kurtosis se ha utilizado el *workflow* mostrado en la Figura 2.
La *Kurtosis* es una medida estadística que muestra la forma de una distribución de probabilidad. Una *Kurtosis* grande implica una mayor concentración de valores de la variables o muy cerca de la media de la distribución (pico) o muy lejos de ella (colas de la distribución), al mismo tiempo que existe una menor frecuencia de valores intermedios.
¿Cómo ilustramos esto en este conjunto de datos?

Para entrenar la red Bayesiana (Figura 2) se deben preparar los datos. Para ello se debe crear una nueva columna *isWeekend* para ilustrar si es fin de semana o no. Utilizando el nodo *Rule engine* se utilizarán las siguientes reglas:

```
$weekday$ MATCHES "Sat" => "Yes"
$weekday$ MATCHES "Sun" => "Yes"
TRUE => "No"
```

Se ha utilizado un nodo *Column filter* para eliminar la columna *weekday* ya que la red Bayesiana presenta un mejor rendimiento si conoce este valor ya que si se entrena con esta variable reconoce la regla de creación de la columna *isWeekend*.
Para llegar a esta conclusión se han probado las distintas combinaciones de columnas utilizando el *Column Filter*.
¿Explica más pruebas realizadas?

Para dibujar la curva *ROC* se ha utilizado un nodo *ROC Cuve (local)* a la salida del nodo *Naive Bayes Predictor*. Se puede observar en la Figura **numFigura** que la forma de la gráfica... **(?)**.

Meter gráfica aqui

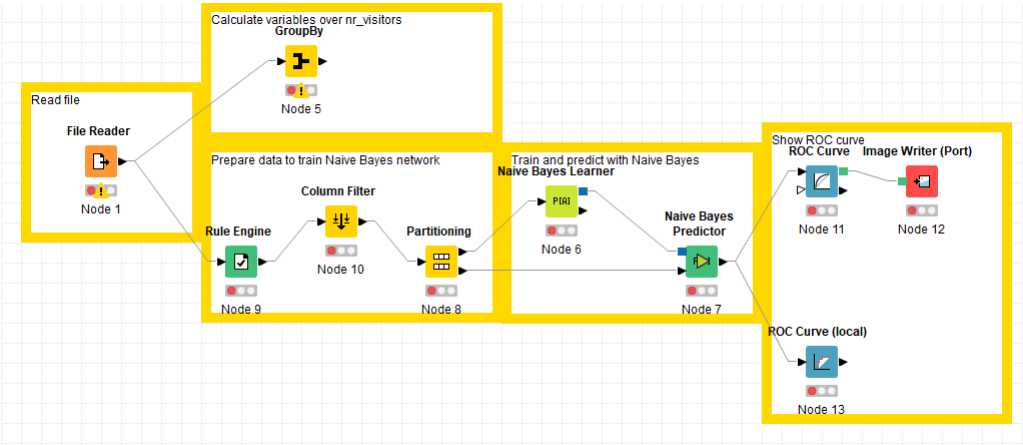

Figura 2: Workflow de ejercicio 2

3. Ejercicio 3

Para la evaluación de los distintos conjuntos de datos se han creado 2 *workflows*. Uno de ellos (Figura 4) utiliza las herramientas de *KNIME* para evaluar los datos. El otro (Figura 3) utiliza las herramientas de *WEKA* para evaluar los datos.
En el anexo se pueden observar las tablas comparativas de las distintas ejecuciones de los *workflows* tanto con *KNIME* (K) como con *WEKA* (W). Para el conjunto de datos *wine* se han entrenado las distintas herramientas con el 80 %, 50 % y 30 %.
Para el conjunto de datos *iris* se han entrenado las distintas herramientas con el 80 %, 50 % y 30 % de los datos. Se puede observar que en todos los modelos se obtienen los resultados más precisos cuando se usa un mayor número de datos de entrenamiento (80 %), llegando en algunos casos a clasificar correctamente el 100 % de los elementos pertenecientes a la clase *Iris-Setosa*, ya que la tasa de acierto (*True Positives*) es 1.

Para el conjunto de datos *adult* se han entrenado las distintas herramientas con el 80 %, 50 % y 30 %. Se puede apreciar que conforme se reduce el número de datos de entrenamiento se pierde precisión, identificando datos pertenecientes a la otra clase de datos (>50K) como datos de clase <=50K. El método que más ha errado en la identificación de las clases ha sido el perceptrón multicapa.
El método que menos ha variado la precisión conforme se entrenaba con menos datos ha sido el *J48* y en especial utilizando los nodos de *WEKA* siempre ha mantenido el porcentaje de *True Positives* por encima del 93 % en la clase <=50K y del 60 % en la clase >50K.

En cuanto a los problemas que han surgido, con el *dataset adult* se necesitaba una columna *Class* y por lo tanto, se ha renombrado la última columna a *"Class"*. Para ello se ha utilizado un nodo *Column Filter* para eliminar todas las variables no numéricas.

Para simplificar y agilizar el entrenamiento y ejecución de cada uno de los clasificadores, se han creado dos *workflows*, uno que agrupa todos los clasificadores que provee el propio *Knime*4 y otro que agrupa los nodos que proporciona el *plugin* de *Weka*3.

De esta manera en cada *Workflow* se indica el *dataset* deseado con el nodo *File Reader* y se selecciona el % de datos a entrenar con el nodo *partitioning*. Los resultados se escriben en archivos *csv* ditintos, indicados con los nodos *csv wirter*. Así se centralizan los distintos modelos de entrenamiento con una sola ejecución.

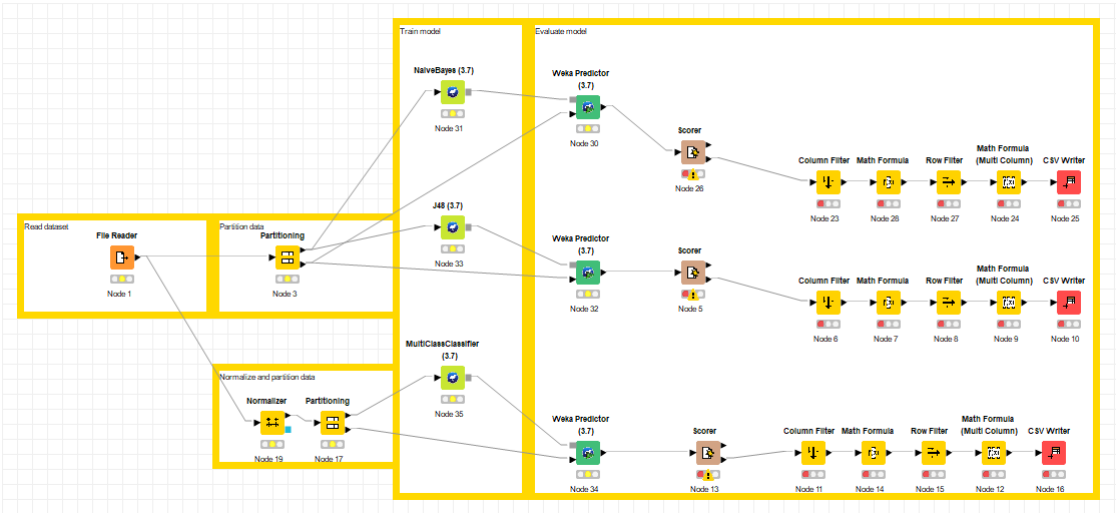

Figura 3: Workflow con modelos de entrenamiento de *Weka*

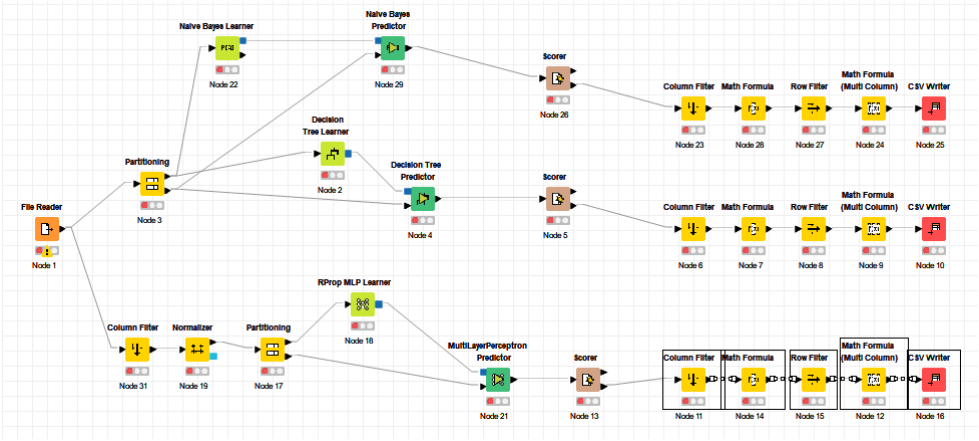

Figura 4: Workflow con modelos de entrenamiento de *Knime*

4. Anexo 1: Cuadros de datos ejercicio 3

4.1. Dataset Adult

4.1.1. Naive Bayes

Clase	True Positives (K)	False Positives (K)	True Positives (W)	False Positives (W)
<=50K	0.942	0.058	0.936	0.064
>50K	0.511	0.489	0.515	0.485

Cuadro 1: Datos de entrenamiento del dataset *Adult* con el 80 % usando *Naive Bayes*

Clase	True Positives (K)	False Positives (K)	True Positives (W)	False Positives (W)
<=50K	0.941	0.059	0.933	0.067
>50K	0.513	0.487	0.513	0.487

Cuadro 2: Datos de entrenamiento del dataset *Adult* con el 50 % usando *Naive Bayes*

Clase	True Positives (K)	False Positives (K)	True Positives (W)	False Positives (W)
<=50K	0.939	0.061	0.934	0.066
>50K	0.508	0.492	0.522	0.478

Cuadro 3: Datos de entrenamiento del dataset *Adult* con el 30 % usando *Naive Bayes*

4.1.2. J48

Clase	True Positives (K)	False Positives (K)	True Positives (W)	False Positives (W)
<=50K	0.900	0.100	0.936	0.064
>50K	0.603	0.397	0.621	0.379

Cuadro 4: Datos de entrenamiento del dataset *Adult* con el 80 % usando *J48*

Clase	True Positives (K)	False Positives (K)	True Positives (W)	False Positives (W)
<=50K	0.897	0.103	0.937	0.063
>50K	0.598	0.402	0.607	0.392

Cuadro 5: Datos de entrenamiento del dataset *Adult* con el 50 % usando *J48*

Clase	True Positives (K)	False Positives (K)	True Positives (W)	False Positives (W)
<=50K	0.890	0.110	0.933	0.067
>50K	0.599	0.401	0.600	0.400

Cuadro 6: Datos de entrenamiento del dataset *Adult* con el 30 % usando *J48*

4.1.3. Perceptrón multicapa

Clase	True Positives (K)	False Positives (K)	True Positives (W)	False Positives (W)
<=50K	0.901	0.099	0.927	0.073
>50K	0.377	0.623	0.602	0.398

Cuadro 7: Datos de entrenamiento del dataset *Adult* con el 80 % usando *Perceptrón multicapa*

Clase	True Positives (K)	False Positives (K)	True Positives (W)	False Positives (W)
<=50K	1.000	0.000	0.930	0.070
>50K	0.000	1.000	0.605	0.395

Cuadro 8: Datos de entrenamiento del dataset *Adult* con el 50 % usando *Perceptrón multicapa*

Clase	True Positives (K)	False Positives (K)	True Positives (W)	False Positives (W)
<=50K	0.890	0.110	0.932	0.068
>50K	0.389	0.611	0.587	0.413

Cuadro 9: Datos de entrenamiento del dataset *Adult* con el 30 % usando *Perceptrón multicapa*

4.2. Dataset Iris

4.2.1. Naive Bayes

Clase	True Positives (K)	False Positives (K)	True Positives (W)	False Positives (W)
Iris-Setosa	1	0	1	0
Iris-versicolor	0,947	0,053	0,928	0,072
Iris-virginica	0,983	0,017	0,913	0,087

Cuadro 10: Datos de entrenamiento del dataset *Iris* con el 80 % usando *Naive Bayes*

Clase	True Positives (K)	False Positives (K)	True Positives (W)	False Positives (W)
Iris-Setosa	1	0	1	0
Iris-versicolor	0,938	0,062	0,895	0,105
Iris-virginica	0,874	0,126	0,912	0,088

Cuadro 11: Datos de entrenamiento del dataset *Iris* con el 50 % usando *Naive Bayes*

Clase	True Positives (K)	False Positives (K)	True Positives (W)	False Positives (W)
Iris-Setosa	1	0	1	0
Iris-versicolor	0,904	0,096	0,814	0,186
Iris-virginica	0,883	0,117	0,932	0,068

Cuadro 12: Datos de entrenamiento del dataset *Iris* con el 20 % usando *Naive Bayes*

4.2.2. J48

Clase	True Positives (K)	False Positives (K)	True Positives (W)	False Positives (W)
Iris-Setosa	1	0	1	0
Iris-versicolor	0,947	0,053	0,903	0,097
Iris-virginica	0,983	0,017	0,910	0,090

Cuadro 13: Datos de entrenamiento del dataset *Iris* con el 80 % usando *J48*

Clase	True Positives (K)	False Positives (K)	True Positives (W)	False Positives (W)
Iris-Setosa	1	0	0,968	0,032
Iris-versicolor	0,952	0,048	0,909	0,091
Iris-virginica	0,886	0,114	0,885	0,115

Cuadro 14: Datos de entrenamiento del dataset *Iris* con el 50 % usando *J48*

Clase	True Positives (K)	False Positives (K)	True Positives (W)	False Positives (W)
Iris-Setosa	1	0	0,922	0,078
Iris-versicolor	0,935	0,065	0,910	0,090
Iris-virginica	0,907	0,093	0,942	0,058

Cuadro 15: Datos de entrenamiento del dataset *Iris* con el 20 % usando *J48*

4.2.3. Perceptrón multicapa

Clase	True Positives (K)	False Positives (K)	True Positives (W)	False Positives (W)
Iris-Setosa	0,8	0,2	1	0
Iris-versicolor	0,865	0,135	0,883	0,117
Iris-virginica	0,860	0,140	0,822	0,178

Cuadro 16: Datos de entrenamiento del dataset *Iris* con el 80 % usando *Perceptrón multicapa*

Clase	True Positives (K)	False Positives (K)	True Positives (W)	False Positives (W)
Iris-Setosa	0,984	0,016	1	0
Iris-versicolor	0,898	0,102	0,933	0,066
Iris-virginica	0,915	0,085	0,947	0,053

Cuadro 17: Datos de entrenamiento del dataset *Iris* con el 50 % usando *Perceptrón multicapa*

Clase	True Positives (K)	False Positives (K)	True Positives (W)	False Positives (W)
Iris-Setosa	0,995	0,005	0,995	0,005
Iris-versicolor	0,797	0,203	0,885	0,115
Iris-virginica	0,853	0,147	0,832	0,168

Cuadro 18: Datos de entrenamiento del dataset *Iris* con el 20 % usando *Perceptrón multicapa*