Tarea 21 abril: 2º parte

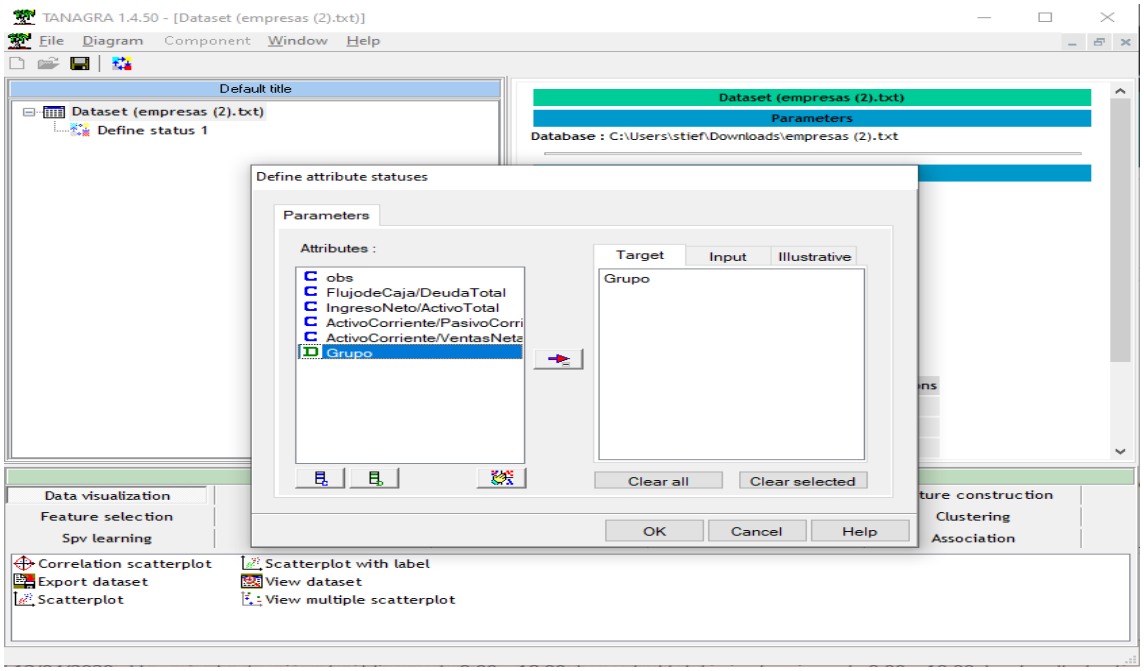
En esta tarea vamos a desarrollar un tutorial para realizar arboles de decisión con la herramienta Tanagra. Las cuestiones a resolver son las siguientes:

1. Considerando la variable grupo como variable explicada y la 4 variables restantes como variables explicativas, aplique el algoritmo ID3· sin cambiar los parámetros. ¿Cuál es el resultado?
2. Modifique los valores de los dos primeros parámetros por 5 y 2 respectivamente. En el árbol de decisión obtenido, ¿cuál es la variable que mejor discrimina a las empresas? ¿Cuántos nodos-hoja tiene el árbol?
3. Modifique los valores de los dos primeros parámetros por 6 y 3 respectivamente. ¿Cambia el árbol de decisión obtenido? Si cambia, indique los cambios.
4. Considerando la variable grupo como variable explicada y la 4 variables restantes como variables explicativas, aplique el algoritmo C4.5 · sin cambiar los parámetros. En el árbol de decisión obtenido, ¿cuál es la variable que mejor discrimina a las empresas? ¿Cuántos nodos-hoja tiene el árbol?

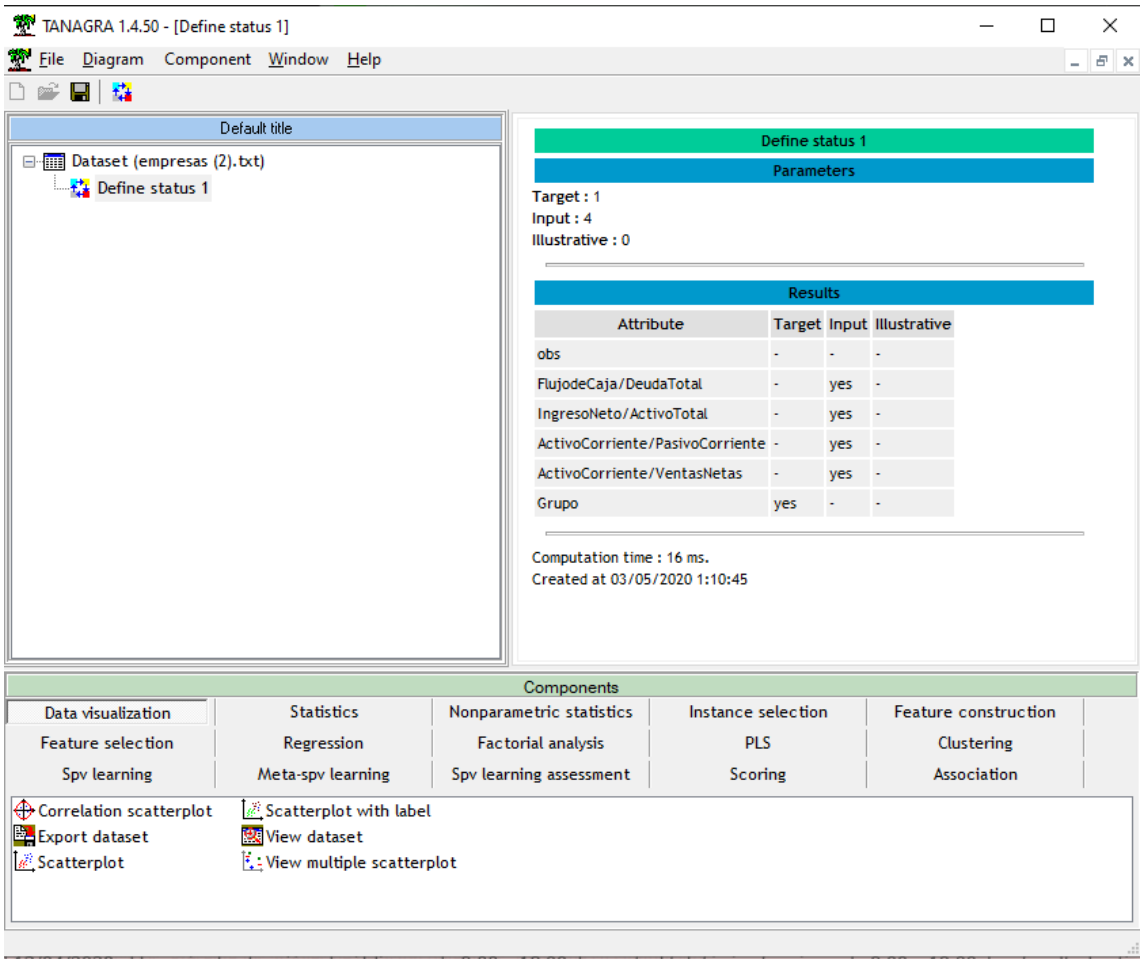
1.

En primer lugar abriremos Tanagra, iremos a File → New file y buscaremos el archivo con el que vamos a trabajar, que se llama empresas.txt.

Una vez tengamos los datos sobre los que vamos a trabajar, iremos a Define Statues y elegiremos con target grupo y como entradas el resto de variables financieras. Nuestra variable objetivo será grupo y las variables explicativas serán los 4 ratios financieros.

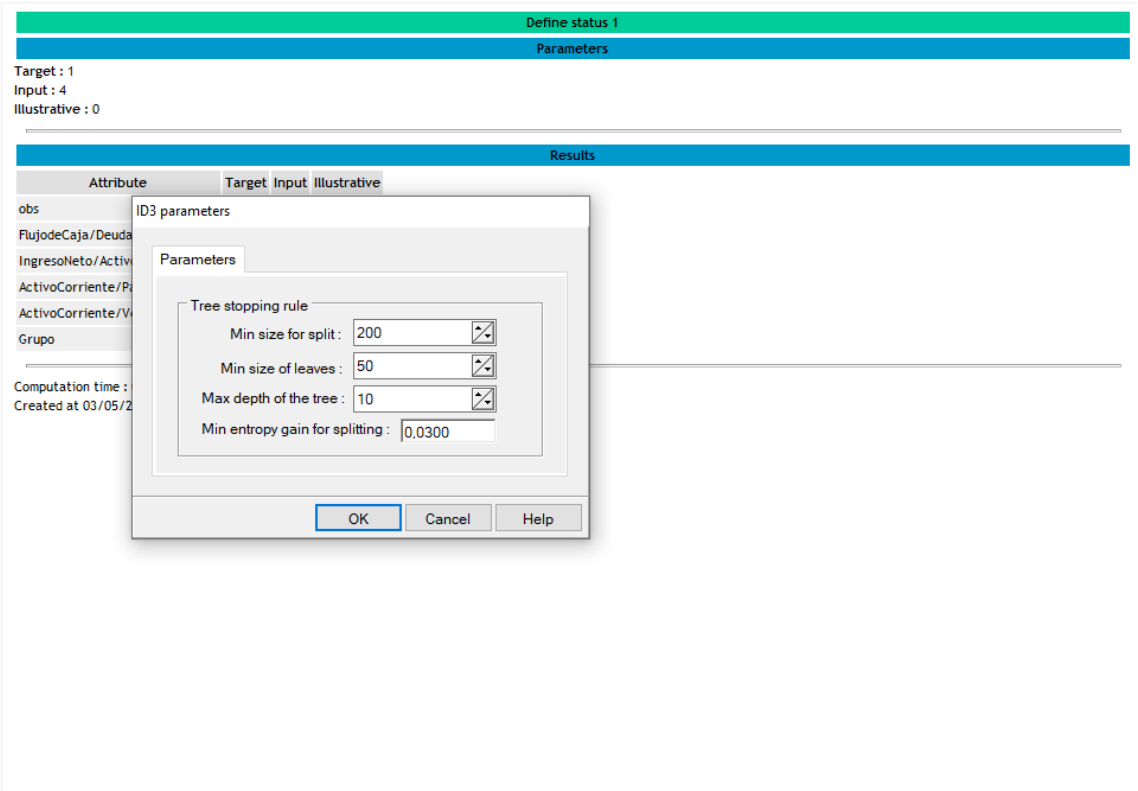


Una vez realizado esto pulsaremos ok. Verificamos que el Define Status queda al cual hemos indicado:



A continuación , usaremos un algoritmo de generación de arboles de descisión llamado ID3. Vamos a la pestaña Spv Learning y seleccionamos y arrastramos ID3 hasta nuestro Define Statues. Dicho algoritmo se encuentra en la pestaña de aprendizaje supervisado porque el árbol de decisión pertenece a la clasificación, y clasificación es aprendizaje supervisado.

Una vez hecho esto lo ejecutamos y observamos que se nos crea un árbol de un nodo y una hoja. Esto es debido a que disponemos de 46 datos, y si miramos la configuración (Supervised Parameter) de ID3 observamos lo siguiente:



¿Qué quieren decir estos datos?

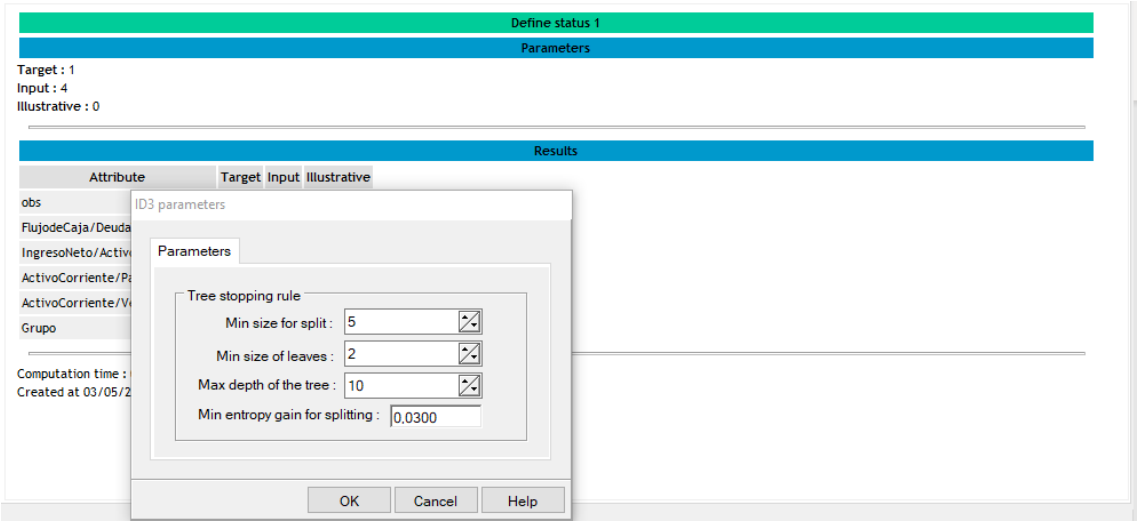
Min size for Split → este número indica cuantos datos tiene que tener el nodo padre para dividirse en más nodos.

Min size of leaves → este número indica que cada hijo tiene que tener mínimo 50 para poder dividirse de nuevo.

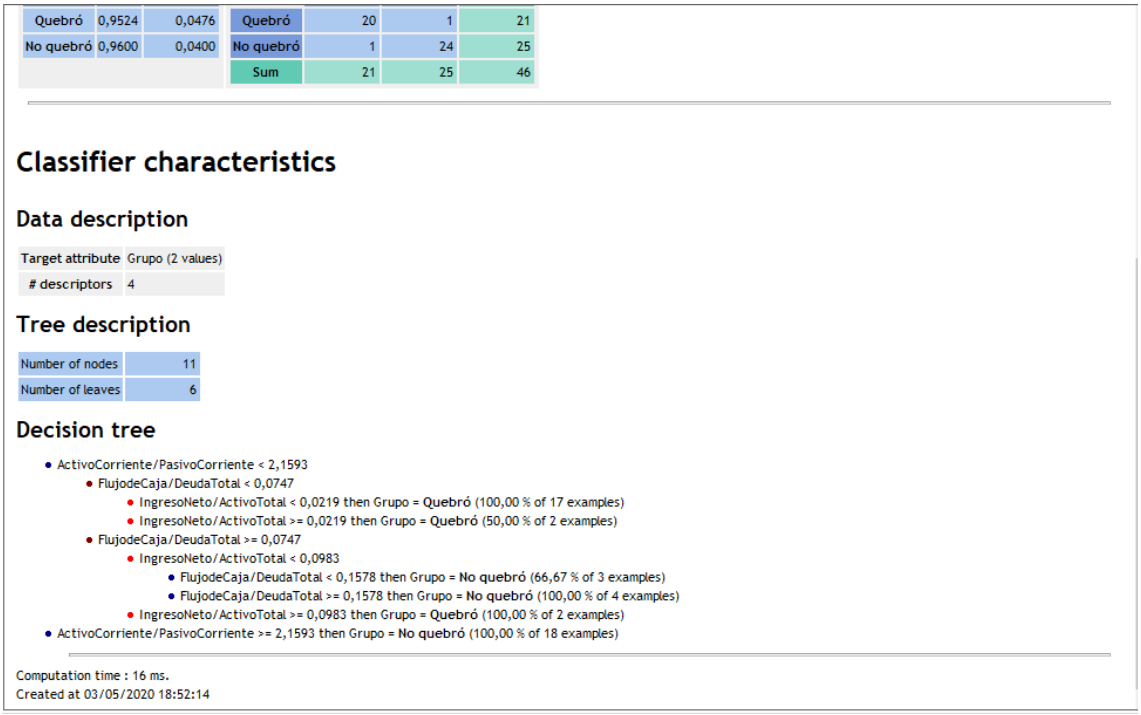
La conclusión que podemos obtener de estos parámetros es que es normal que nos salga un árbol de un solo nodo y una sola hoja, ya que el número mínimo de datos necesarios en la raíz del árbol para que se divida es 200, y nosotros solo contamos con 46 datos.

2.

Procedemos a cambiar los datos Min size for Split y Min size of leaves por 5 y 2 respectivamente y ejecutamos ID3.



De esta manera obtenemos el siguiente árbol de decisiones:



Quebró	0,9524	0,0476	Quebró	20	1	21
No quebró	0,9600	0,0400	No quebró	1	24	25
			Sum	21	25	46

Classifier characteristics

Data description

Target attribute	Grupo (2 values)
# descriptors	4

Tree description

Number of nodes	11
Number of leaves	6

Decision tree

- ActivoCorriente/PasivoCorriente < 2,1593
  - FlujodeCaja/DeudaTotal < 0,0747
    - IngresoNeto/ActivoTotal < 0,0219 then Grupo = **Quebró** (100,00 % of 17 examples)
    - IngresoNeto/ActivoTotal >= 0,0219 then Grupo = **Quebró** (50,00 % of 2 examples)
  - FlujodeCaja/DeudaTotal >= 0,0747
    - IngresoNeto/ActivoTotal < 0,0983
      - FlujodeCaja/DeudaTotal < 0,1578 then Grupo = **No quebró** (66,67 % of 3 examples)
      - FlujodeCaja/DeudaTotal >= 0,1578 then Grupo = **No quebró** (100,00 % of 4 examples)
    - IngresoNeto/ActivoTotal >= 0,0983 then Grupo = **Quebró** (100,00 % of 2 examples)
- ActivoCorriente/PasivoCorriente >= 2,1593 then Grupo = **No quebró** (100,00 % of 18 examples)

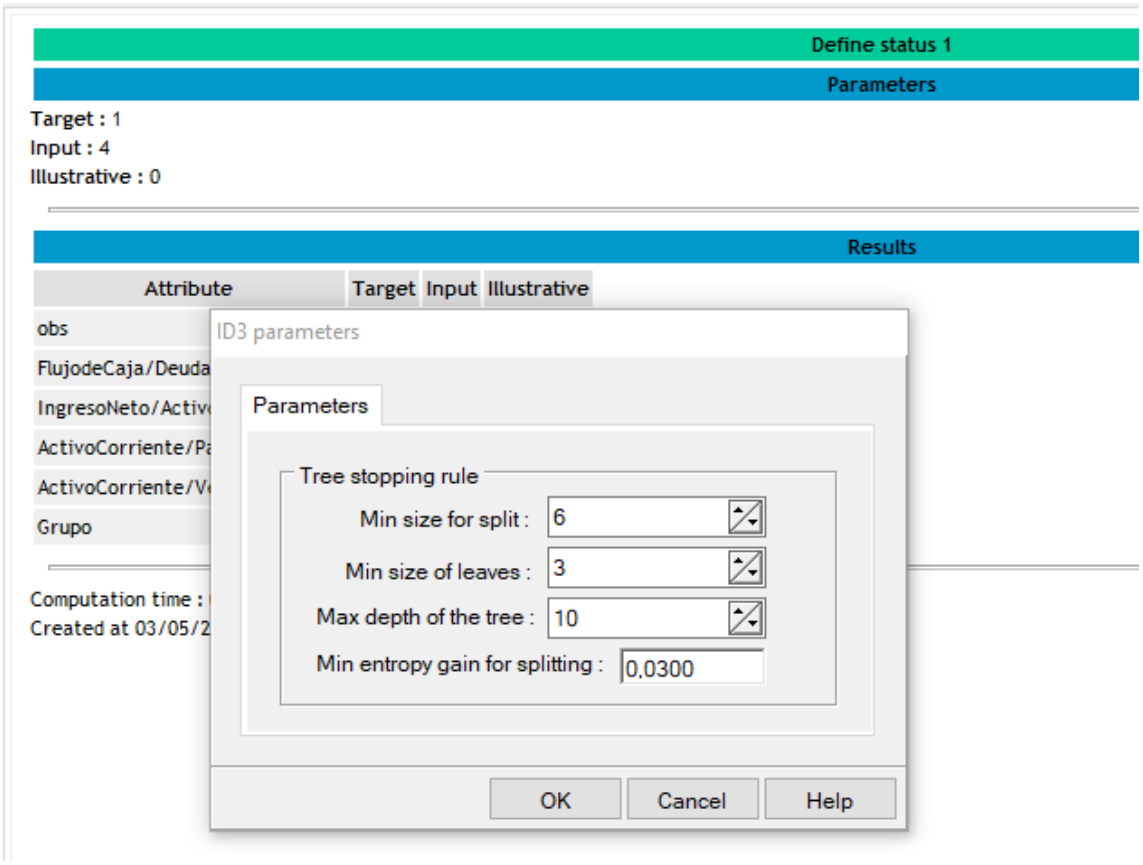
Computation time : 16 ms.
Created at 03/05/2020 18:52:14

En este caso observamos que hay 11 nodos y 6 hojas. A la izquierda del todo del árbol se encontraría la raíz, que no aparece.

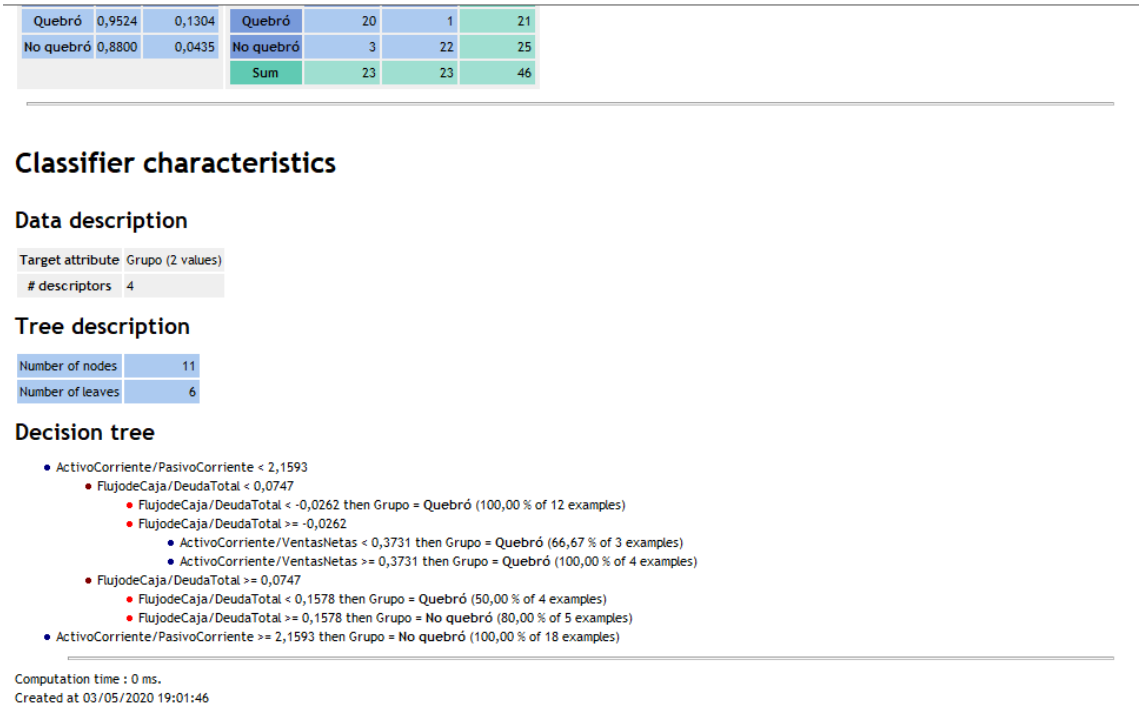
Podemos observar que la variable que mejor discrimina las empresas es ActivoCorriente/PasivoCorriente, ya que es la primera bifurcación desde la raíz.

3.

En esta ocasión modificaremos los parámetros supervisados de la siguiente manera: Min size for Split y Min size of leaves los sustituiremos por 6 y 3 respectivamente.



Una vez realizado este paso, procedemos a ejecutar el ID3 y observar el árbol de decisiones que se ha creado.



Quebró	0,9524	0,1304	Quebró	20	1	21
No quebró	0,8800	0,0435	No quebró	3	22	25
			Sum	23	23	46

Classifier characteristics

Data description

Target attribute	Grupo (2 values)
# descriptors	4

Tree description

Number of nodes	11
Number of leaves	6

Decision tree

- ActivoCorriente/PasivoCorriente < 2,1593
  - FlujodeCaja/DeudaTotal < 0,0747
    - FlujodeCaja/DeudaTotal < -0,0262 then Grupo = **Quebró** (100,00 % of 12 examples)
    - FlujodeCaja/DeudaTotal >= -0,0262
      - ActivoCorriente/VentasNetas < 0,3731 then Grupo = **Quebró** (66,67 % of 3 examples)
      - ActivoCorriente/VentasNetas >= 0,3731 then Grupo = **Quebró** (100,00 % of 4 examples)
  - FlujodeCaja/DeudaTotal >= 0,0747
    - FlujodeCaja/DeudaTotal < 0,1578 then Grupo = **Quebró** (50,00 % of 4 examples)
    - FlujodeCaja/DeudaTotal >= 0,1578 then Grupo = **No quebró** (80,00 % of 5 examples)
- ActivoCorriente/PasivoCorriente >= 2,1593 then Grupo = **No quebró** (100,00 % of 18 examples)

Computation time : 0 ms.
Created at 03/05/2020 19:01:46

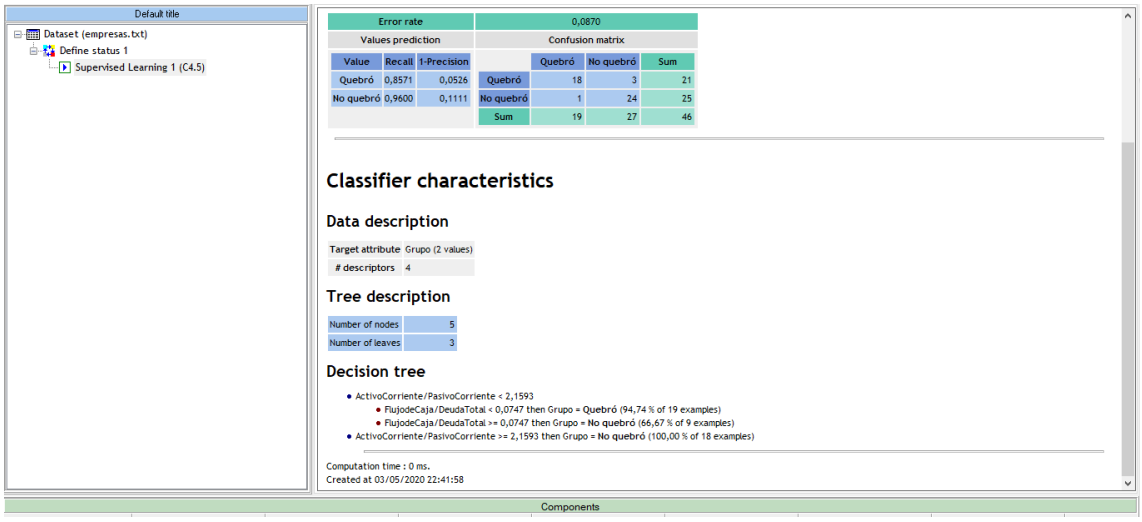
Podemos observar que lo único que cambió respecto al anterior árbol es el porcentaje que sale de empresas que quebraron o no quebraron. Esto se puede observar en las hojas.

Sin embargo, el número de nodos y de hojas sigue siendo exactamente el mismo que en el ejemplo anterior.

4.

En esta ocasión vamos a crear un árbol de decisiones basado en un algoritmo distinto, concretamente el algoritmo C4.5.

Este algoritmo lo encontraremos en la misma pestaña que hemos encontrado el algoritmo ID3, en la pestaña Spv learning. Lo seleccionamos y lo arrastramos hacia el Define Statues, ya que vamos a trabajar con los mismos datos. Una vez lo ejecutemos, observaremos lo siguiente:



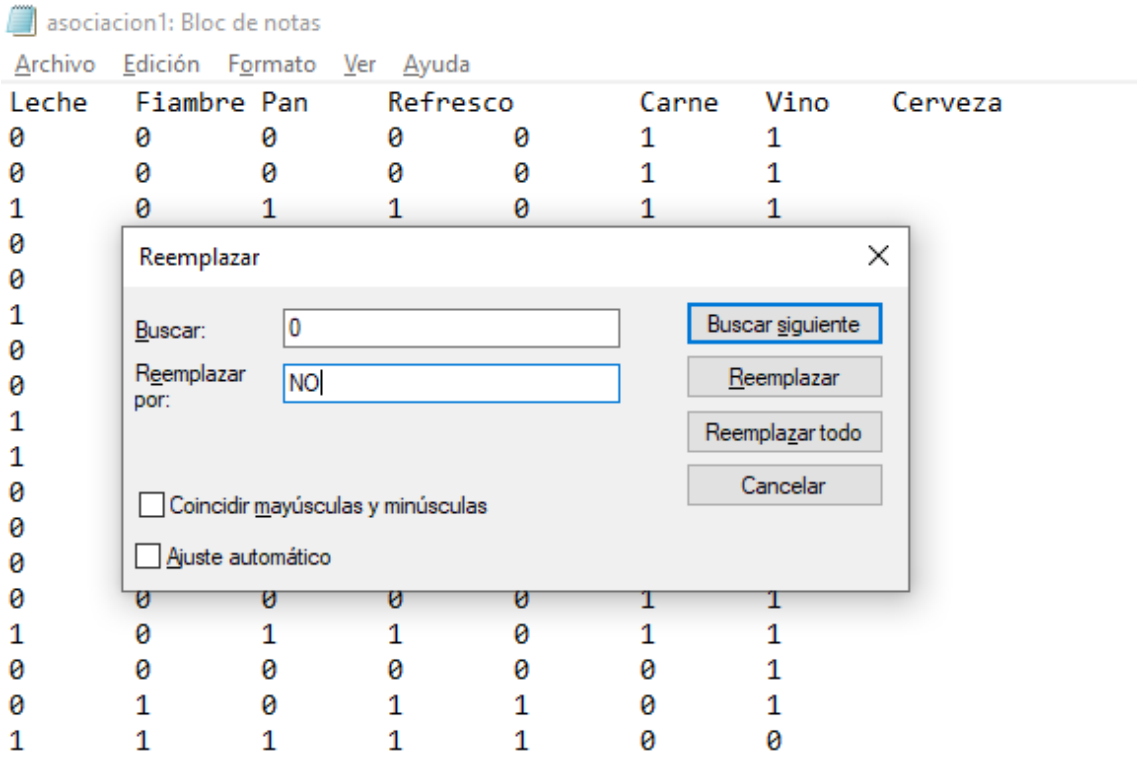
En esta ocasión el árbol consta de 5 nodos y 3 hojas. La variable que más discrimina es la primera ActivoCorriente/PasivoCorriente, ya que es la primera división de la raíz.

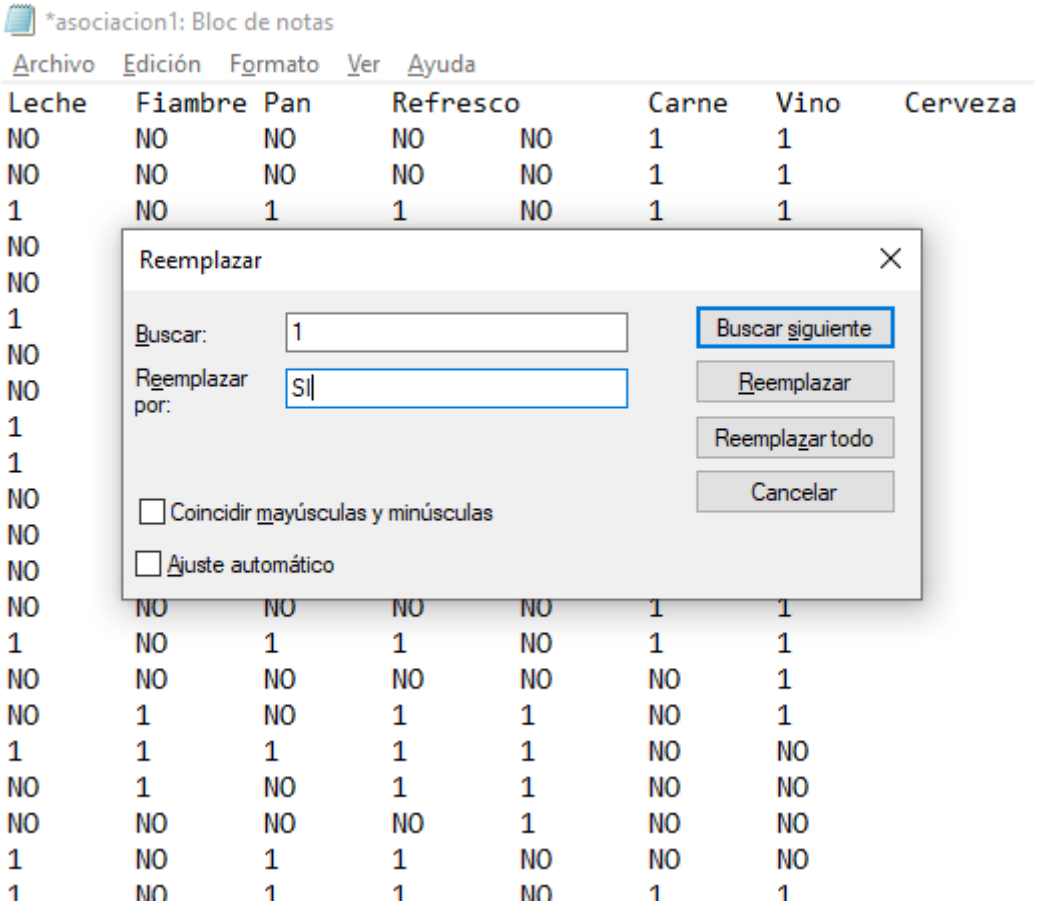
Cuando usamos el algoritmo ID3 para crear un árbol de decisiones, la variable que mas discriminaba fue exactamente la misma. Esto se debe a que es un árbol es fuerte, el algoritmo que apliques no debería de influir en la variable que más discrimina.

Árbol de decisión con los datos recogidos en asociación.txt mediante el algoritmo ID3

Para realizar un árbol de decisión con los datos de asociacion.txt lo primero que haremos será modificar el archivo en cuestión, ya que para realizar arboles de decisión debemos usar datos categóricos, y en el caso del archivo asociación.txt nos encontramos que los datos no son categóricos.

Por tanto, abriremos primeramente el archivo de texto con el bloc de notas. Una vez tengamos el bloc de notas abierto, haremos usaremos el atajo Control +R para reemplazar los 0 en el archivo por NO y los 1 por SI.

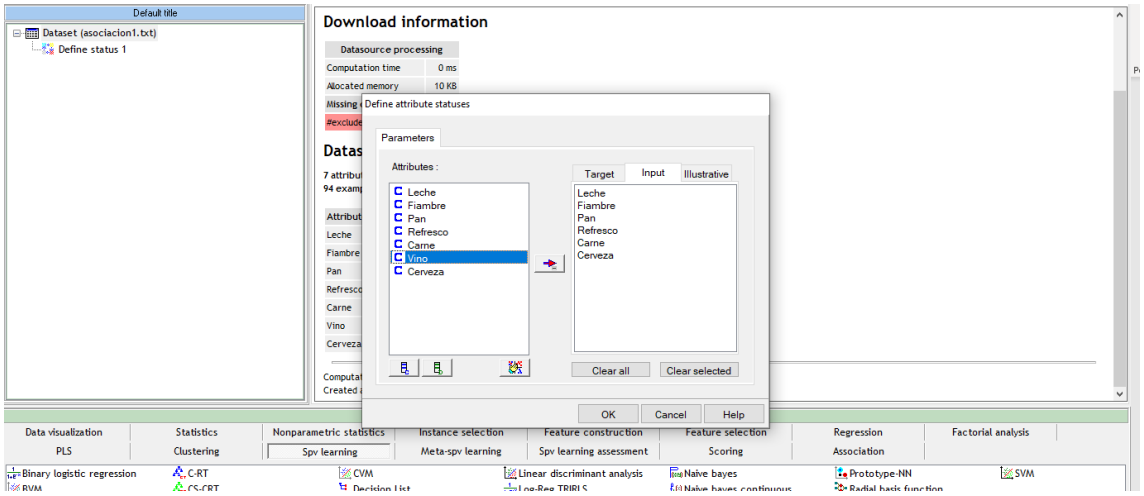




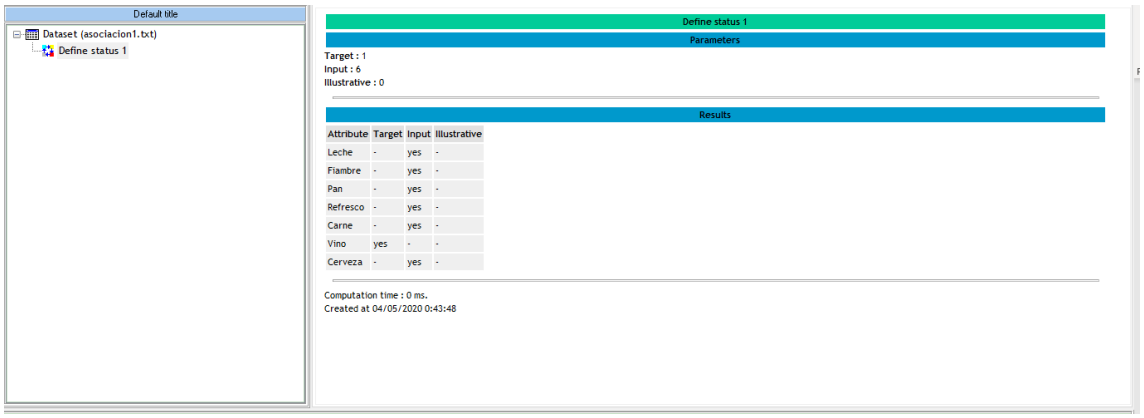
Ya podemos guardar, cerrar el archivo, y podemos abrir Tanagra.

En File→ New File buscaremos el archivo en cuestión y lo abriremos.

Es el momento de crear un Define Statues. En esta ocasión nuestro objetivo será encontrar que productos se compraron junto al vino, por lo que nuestro target será “Vino”, y el resto de alimentos serán Inputs.

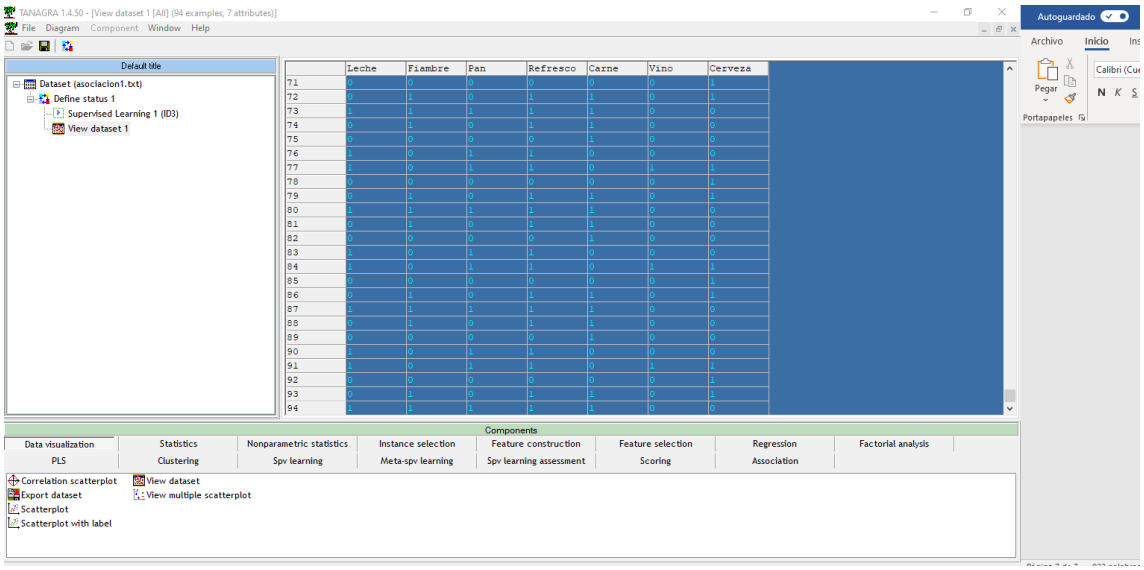


Comprobamos que se ha agregado el Define Statues tal y como queríamos.



Ahora vamos a la pestaña de aprendizaje supervisado (Spv Learning) y seleccionamos el algoritmo ID3, arrastrándolo hacia nuestro Define Statues.

Ahora podríamos intentar ejecutar los parámetros por defecto, pero observaremos que solo se crea un árbol de un nodo y una hoja. Esto se debe a que los parámetros por defecto exigen que la raíz tenga como mínimo 200 datos para poder dividirse. Si nos vamos a la pestaña Data visualization y seleccionamos y arrastramos View dataset hacia Define Statues, podremos observar los datos de los que disponemos.

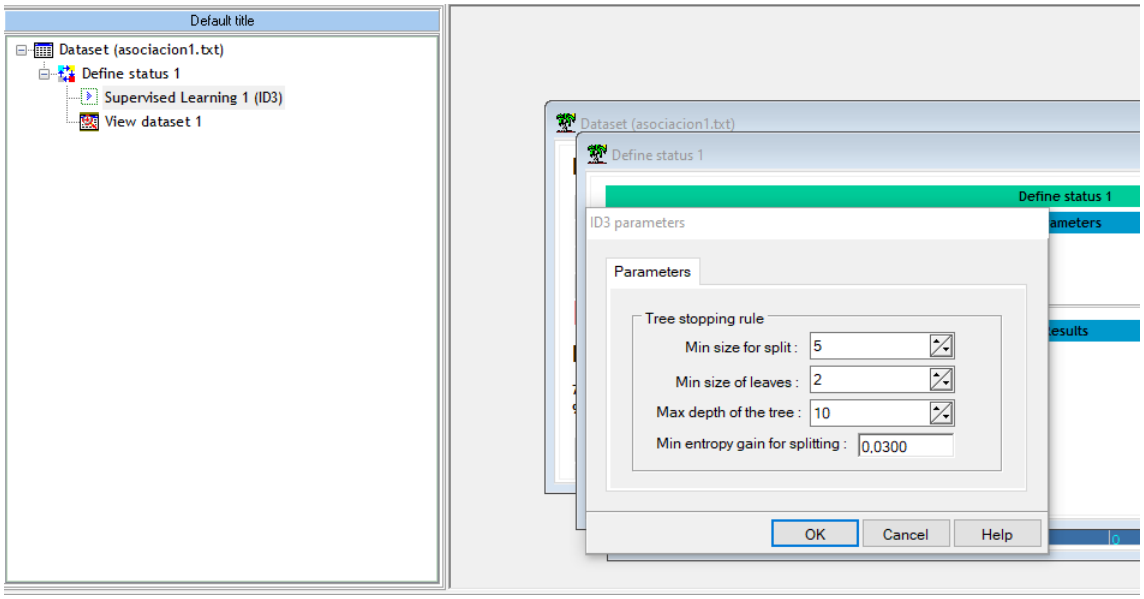


Vemos que los datos que tenemos son 96, por lo que es imposible que con los parámetros por defecto nos cree un árbol de decisión.

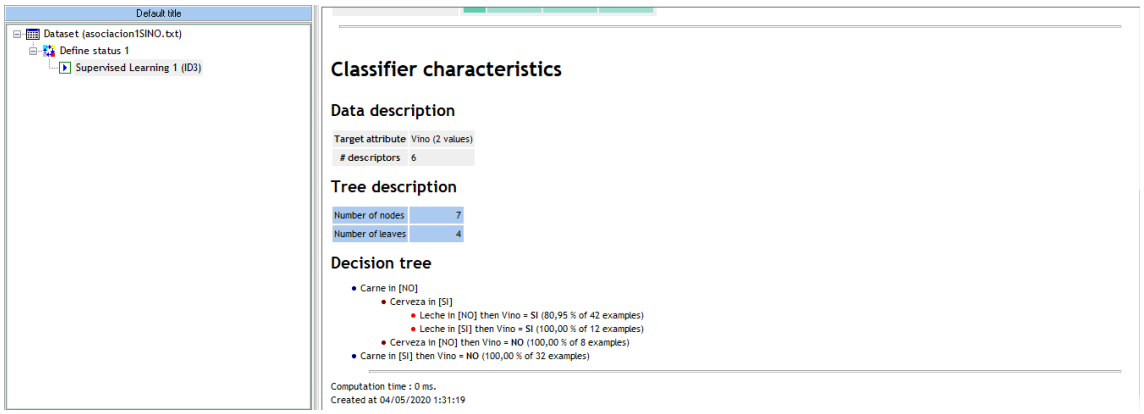
Es por ello que vamos a hacer click en Supervised Learning 1 (ID3) con el botón derecho y seleccionaremos la opción Supervised Parameters.

Es hora de configurar los datos necesarios para que la raíz del árbol se pueda dividir, y los hijos de dicha raíz o “padre”.

Elegiremos 5 y 2 en este caso.



Ahora pulsamos ok y ejecutamos. Podremos observar que se ha generado el siguiente arbol de decisiones:



En nuestro árbol, la raíz no la vemos, pero observamos que la variable que más discrimina es Carne, ya que es la primera bifurcación que nos encontramos desde la raíz.

Podemos observar que el árbol consta de 7 nodos y 4 hojas.

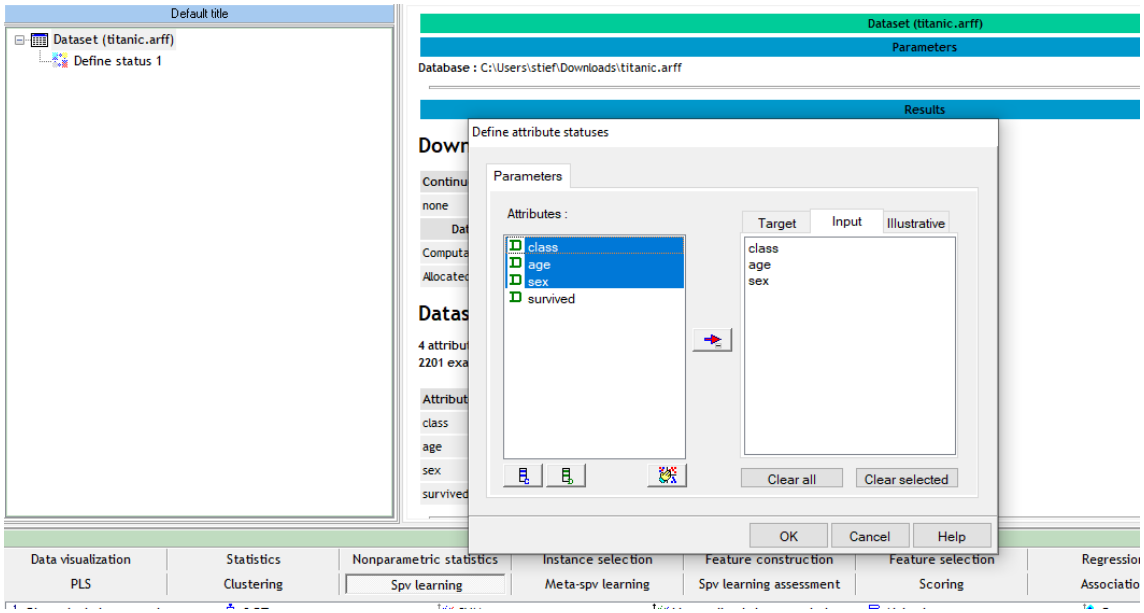
De este árbol podemos aprender por ejemplo, que si un cliente no compró carne, compró también cerveza y no compró leche, entonces en un 80,95% de los casos sí compró vino.

Sin embargo, si sí compró carne, entonces en un 100% de los casos no compró vino.

Árbol de decisión con los datos recogidos en titanic.txt mediante el algoritmo C4-5

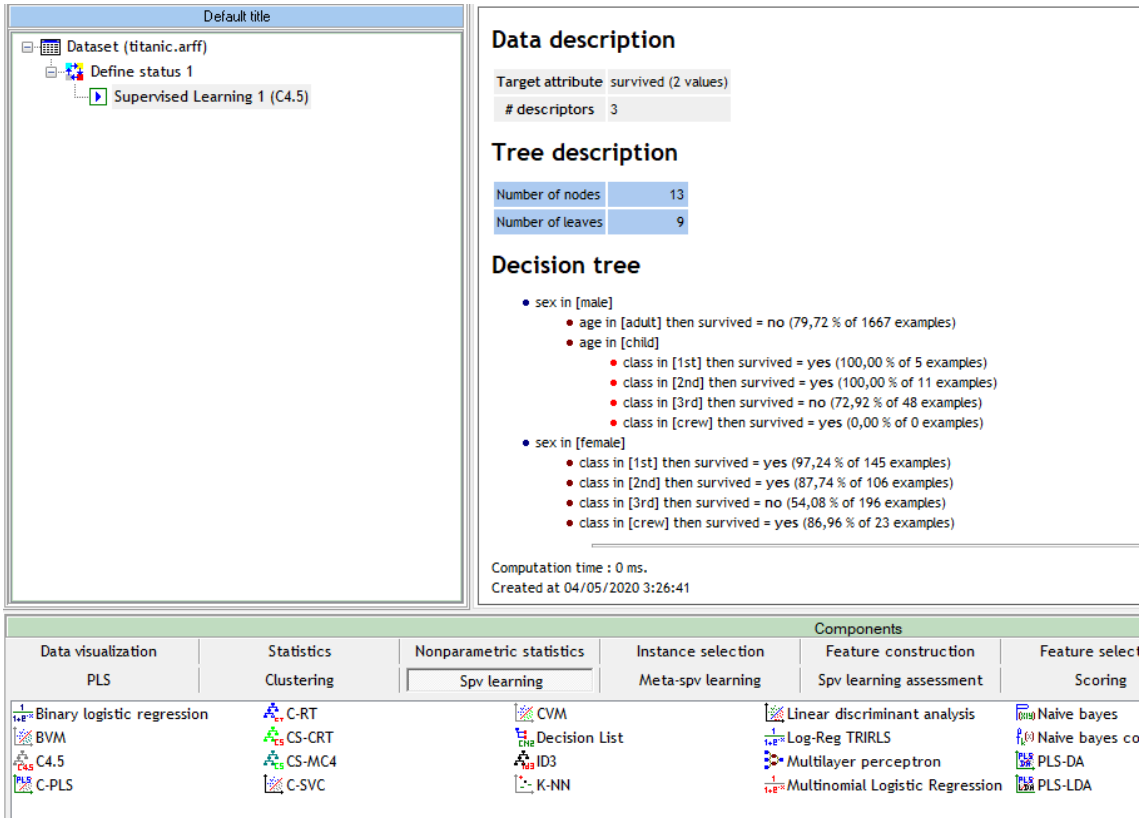
En esta ocasión vamos a trabajar con el archivo titanic.arff.

Procedemos a abrir Tanagra. Vamos a File → New File y buscamos y seleccionamos el archivo en cuestión. Tras esto definiremos las variables que queremos como explicativas y la variable que queremos explicada. En este caso, el target será survived y en input seleccionaremos el resto de variables.



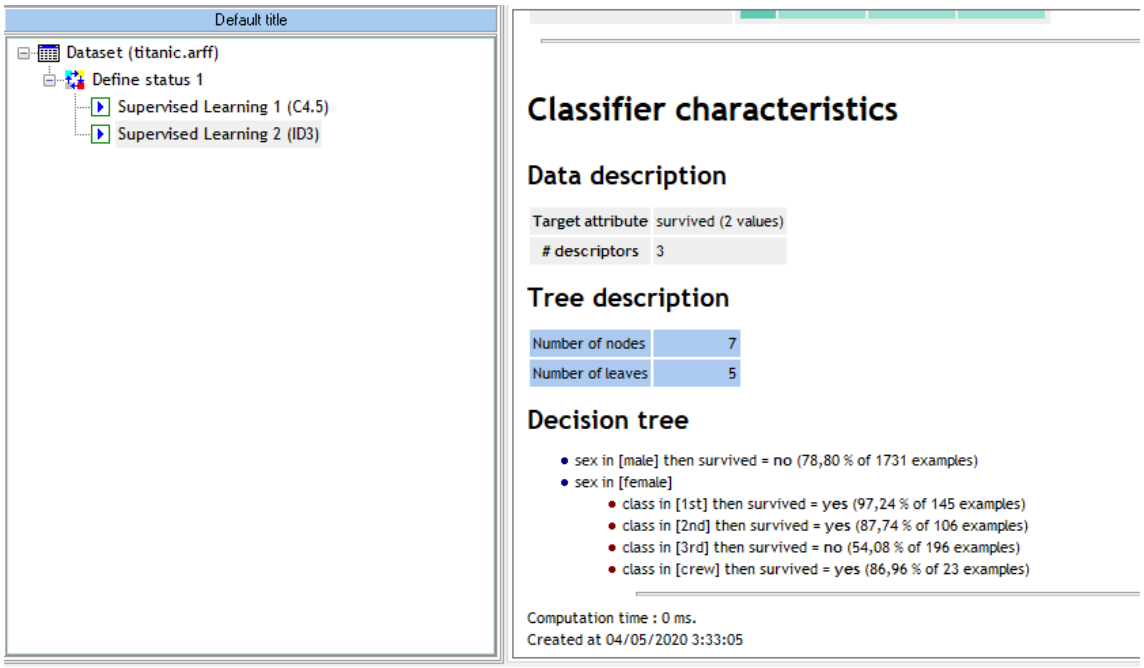
Ejecutamos y observamos que las variables han quedado de la manera que esperamos. Tras esto, nos iremos a la pestaá Spv learning, ya que queremos hacer un árbol de decisión. En esta ocasión vamos a usar el algoritmo C4-5 en vez del algoritmo ID3 para realizar la tarea y así nos familiarizamos más con la herramienta.

Arrastramos C4.5 hasta el Define Statues. Vamos a dejar los parámetros predeterminados y vamos a ejecutar. Deberemos llegar a la siguiente pantalla:



Aquí se nos describe el árbol que hemos creado mediante el algoritmo. Podemos observar que consta de 13 nodos y 9 hojas, y que el atributo que más discrimina es el sexo del tripulante.

Sólo por ver que no tiene que quedar los mismos arboles usando distintos algoritmos, pero el atributo que más discrimina sí tiene que ser el mismo sea cual sea el algoritmo, vamos a ver que árbol quedaría aplicando el ID3.


Default title
Dataset (titanic.arff)
Define status 1
Supervised Learning 1 (C4.5)
Supervised Learning 2 (ID3)
Classifier characteristics
Data description
Target attribute
survived (2 values)
descriptors
3
Tree description
Number of nodes
7
Number of leaves
5
Decision tree
sex in [male] then survived = no (78,80 % of 1731 examples)
sex in [female]
class in [1st] then survived = yes (97,24 % of 145 examples)
class in [2nd] then survived = yes (87,74 % of 106 examples)
class in [3rd] then survived = no (54,08 % of 196 examples)
class in [crew] then survived = yes (86,96 % of 23 examples)
Computation time : 0 ms.
Created at 04/05/2020 3:33:05