

ANÁLISIS DE CLASIFICADORES BAYESIANOS

por M.Ing. Enrique Fernández

1. Introducción

Una red bayesiana es un grafo acíclico dirigido en el que cada nodo representa una variable y cada arco una dependencia probabilística, son utilizadas para proveer: una forma compacta de representar el conocimiento, y métodos flexibles de razonamiento. El obtener una red, bayesiana a partir de datos, es un proceso de aprendizaje que se divide en dos etapas: el aprendizaje estructural y el aprendizaje paramétrico.

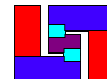
En este trabajo se describirá el funcionamiento de tres algoritmos de Clasificadores, Naïve Bayes, TAN y KDB. Se mostrará además como, a través del programa Elvira, se puede llegar a obtener una red Bayesiana con estos clasificadores. Dicha red variará dependiendo del algoritmo clasificador aplicado, y de la combinación de este con algún algoritmo de inducción de árboles de decisión. Por último se mostrará una comparación que permita analizar las diferencias entre los distintos clasificadores y la influencia que en ellos genera los algoritmos generadores de árboles de decisión.

1.1. Definición de conceptos

1.1.1. Aprendizaje automático

El aprendizaje puede ser definido como "cualquier proceso a través del cual un sistema mejora su eficiencia" [Felgaer, P. et al, 2003]. La habilidad de aprender es considerada como una característica central de los "sistemas inteligentes" [García-Martínez, 1997; García Martínez *et al.*, 2003; Fritz *et al.*, 1989; García-Martínez, 1993; García-Martínez, 1995; García-Martínez & Borrajo, 2000], y es por esto que se ha invertido esfuerzo y dedicación en la investigación y el desarrollo de esta área. El desarrollo de los sistemas basados en conocimientos motivó la investigación en el área del aprendizaje con el fin de automatizar el proceso de adquisición de conocimientos el cual se considera uno de los problemas principales en la construcción de estos sistemas.

Un aspecto importante en el aprendizaje inductivo es el de obtener un modelo que represente el dominio de conocimiento y que sea accesible para el usuario, en particular, resulta importante obtener la información de dependencia entre las variables involucradas en el fenómeno, en los sistemas donde se desea predecir el comportamiento de algunas variables desconocidas basados en otras conocidas; una representación del conocimiento que es capaz de capturar esta información sobre las dependencias entre las variables son las redes bayesianas [Ramoni & Sebastiani, 1999; Felgaer, P. et al, 2003].



1.1.2. Minería de datos

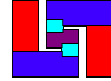
Se denomina Minería de Datos [Servente, & García-Martínez, 2002; Perichinsky & García-Martínez, 2000; Perichinsky *et al.*, 2000; Perichinsky *et al.*, 2001; Perichinsky *et al.*, 2003; Felgaer, P. et al, 2003] al conjunto de técnicas y herramientas aplicadas al proceso no trivial de extraer y presentar conocimiento implícito, previamente desconocido, potencialmente útil y humanamente comprensible, a partir de grandes conjuntos de datos, con objeto de predecir de forma automatizada tendencias y comportamientos; y describir de forma automatizada modelos previamente desconocidos [Piatetski-Shapiro *et al.*, 1991; Chen *et al.*, 1996; Mannila, 1997; Felgaer, P. et al, 2003]. El término Minería de Datos Inteligente [Evangelos & Han, 1996; Michalski *et al.*, 1998; Felgaer, P. et al, 2003] refiere específicamente a la aplicación de métodos de aprendizaje automático [Michalski *et al.*, 1983; Holsheimer & Siebes, 1991; Felgaer, P. et al, 2003], para descubrir y enumerar patrones presentes en los datos, para estos, se desarrollaron un gran número de métodos de análisis de datos basados en la estadística [Michalski *et al.*, 1982; Felgaer, P. et al, 2003]. En la medida en que se incrementaba la cantidad de información almacenada en las bases de datos, estos métodos empezaron a enfrentar problemas de eficiencia y escalabilidad y es aquí donde aparece el concepto de minería de datos. Una de las diferencias entre al análisis de datos tradicional y la minería de datos es que el primero supone que las hipótesis ya están construidas y validadas contra los datos, mientras que el segundo supone que los patrones e hipótesis son automáticamente extraídos de los datos [Hernández Orallo, 2000; Felgaer, P. et al, 2003]. Las tareas de la minería de datos se pueden clasificar en dos categorías: minería de datos descriptiva y minería de datos predicativa [Piatetski-Shapiro *et al.*, 1996; Han, 1999; Felgaer, P. et al, 2003].

1.2. Redes Bayesianas

Una red bayesiana es un grafo acíclico dirigido en el que cada nodo representa una variable y cada arco una dependencia probabilística, en la cual se especifica la probabilidad condicional de cada variable dados sus padres, la variable a la que apunta el arco es dependiente (causa-efecto) de la que está en el origen de éste. La topología o estructura de la red nos da información sobre las dependencias probabilísticas entre las variables pero también sobre las independencias condicionales de una variable (o conjunto de variables) dada otra u otras variables, dichas independencias, simplifican la representación del conocimiento (menos parámetros) y el razonamiento (propagación de las probabilidades).

El obtener una red Bayesiana a partir de datos, es un proceso de aprendizaje que se divide en dos etapas: el aprendizaje estructural y el aprendizaje paramétrico [Pearl, 1988; Hernández O.J. et al, 2004]. La primera de ellas, consiste en obtener la estructura de la red bayesiana, es decir, las relaciones de dependencia e independencia entre las variables involucradas. La segunda etapa, tiene como finalidad obtener las probabilidades a priori y condicionales requeridas a partir de una estructura dada.

Estas redes [Pearl, 1988; Hernández O.J. et al, 2004] son utilizadas en diversas áreas de aplicación como por ejemplo en medicina [Beinlinch *et al.*, 1989; Hernández O.J. et al, 2004], ciencia [Breese & Blake, 1995; Hernández O.J. et al, 2004], y economía



[Hernández O.J. et al, 2004]. Las mismas proveen una forma compacta de representar el conocimiento y métodos flexibles de razonamiento - basados en las teorías probabilísticas - capaces de predecir el valor de variables no observadas y explicar las observadas. Entre las características que poseen las redes bayesianas, se puede destacar que permiten aprender sobre relaciones de dependencia y causalidad, permiten combinar conocimiento con datos [Heckerman *et al.*, 1995; Díaz & Corchado, 1999; Hernández O.J. et al, 2004] y pueden manejar bases de datos incompletas [Heckerman, 1995; Heckerman & Chickering, 1996; Ramoni & Sebastiani, 1996; Hernández O.J. et al, 2004].

A continuación se describirá el funcionamiento de los clasificadores naïve Bayes, TAN y KDB.

1.2.1 Descripción de los algoritmos de aprendizaje

1.2.1.1. Introducción al Clasificador naïve Bayes

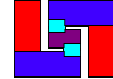
El paradigma clasificatorio en el que se utiliza el teorema de Bayes en conjunción con la hipótesis de independencia condicional de las variables predictoras dada la clase que se conoce bajo diversos nombres que incluye los de idiota Bayes [Ohmann y Col, 1988; Larrañaga. P. et al, 2004], naïve Bayes [Kononenko, 1990; Larrañaga. P. et al, 2004], simple Bayes [Gammerman y Thatcher, 1991; Larrañaga. P. et al, 2004] y Bayes independiente [Todd y Stamper, 1994; Larrañaga. P. et al, 2004].

A pesar una larga tradición en la comunidad de reconocimiento de patrones [Duda y Hart, 1973; Larrañaga. P. et al, 2004] el clasificador Naïve Bayes aparece por primera vez en la literatura del aprendizaje automático a finales de los ochenta [Cestnik y Col, 1987; Larrañaga. P. et al, 2004] con el objetivo de comparar su capacidad predictiva con la de métodos mas sofisticados. De manera gradual los investigadores de esta comunidad de aprendizaje automático se han dado cuenta de su potencialidad y robustez en problemas de clasificación supervisada.

En esta sección se va a afectar una revisión del paradigma naïve *Bayes*, el cual debe su nombre a las hipótesis tan simplificadoras – independencia condicional de las variables predictoras dada la variable clase- sobre las que se construye dicho clasificador. Partiremos del paradigma clásico de diagnóstico para, una vez comprobado que necesita de la estimación de un número de parámetros ingente, ir simplificando paulatinamente las hipótesis sobre las que se construye hasta llegar al modelo naïve Bayes. Veremos a continuación un resultado teórico que nos servirá para entender mejor las características del clasificador naïve Bayes.

1.2.1.2. Del Paradigma Clásico de Diagnóstico al Clasificador Naïve Bayes

Vamos y comenzar recordando el teorema de Bayes con una formulación de sucesos, para posteriormente formularlo en términos de variables aleatorias. Una vez visto el teorema de Bayes, se presenta el paradigma clásico de diagnóstico. Viéndose la necesidad de ir simplificando las premisas sobre las que se construye en aras de obtener paradigmas que puedan ser de aplicación para la resolución de



problemas reales. El contenido de este apartado resulta ser una adaptación del material que Díez y Nell (1998) dedican al mismo.

Teorema (Bayes, 1764) Sean A y B dos sucesos aleatorios cuyas probabilidades se denotan por $p(A)$ y $p(B)$ respectivamente, verificándose que $p(B) > 0$. Supongamos conocidas las probabilidades a priori de los sucesos A y B, es decir, $p(A)$ y $p(B)$, así como la probabilidad condicionada del suceso B dado el suceso A, es decir $p(B|A)$. La probabilidad a posteriori del suceso A conocido que se verifica el suceso B, es decir $p(A|B)$, puede calcularse a partir de la siguiente fórmula:

$$p(A|B) = \frac{p(A, B)}{p(B)} = \frac{p(A) p(B|A)}{p(B)} = \frac{p(A) p(B|A)}{\sum_{A'} p(A') p(B|A')}$$

La formulación del teorema de Bayes puede efectuarse también para variables aleatorias, tanto unidimensionales como multidimensionales.

Comenzando por la formulación para dos variables aleatorias unidimensionales que denotamos por X e Y, tenemos que:

$$P(Y = y | X = x) = \frac{P(Y = y) p(X = x | Y = y)}{\sum_{Y'} p(Y = y') p(X = x | Y = y')}$$

El teorema de Bayes también puede ser expresado por medio de una notación que usa el número de componentes de cada una de las variables multidimensionales anteriores X e Y, de la siguiente manera:

$$P(Y = y | X = x) = p(Y_1 = y_1, \dots, Y_m = y_m | X_1 = x_1, \dots, X_m = x_m) \\ = \frac{p(Y_1 = y_1, \dots, Y_m = y_m) p(X_1 = x_1, \dots, X_m = x_m | Y_1 = y_1, \dots, Y_m = y_m)}{\sum_{Y'_1, \dots, Y'_m} p(X_1 = x_1, \dots, X_m = x_m | Y_1 = y'_1, \dots, Y_m = y'_m) p(Y_1 = y'_1, \dots, Y_m = y'_m)}$$

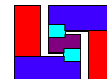
En el problema de clasificación supervisada reflejado en la tabla 1, tenemos que Y = C es una variable unidimensional, mientras que $X = (X_1, \dots, X_n)$ es una variable n-dimensional.

	X_1	X_n	Y
$(x^{(1)}, y^{(1)})$	$x_1^{(1)}$	$x_n^{(1)}$	$y^{(1)}$
$(x^{(2)}, y^{(2)})$	$x_1^{(2)}$	$x_n^{(2)}$	$y^{(2)}$
.....			
$(x^{(n)}, y^{(n)})$	$x_1^{(n)}$	$x_n^{(n)}$	$y^{(n)}$

Tabla 1: Problema de clasificación supervisada.

Vamos a plantear la formulación clásica de un problema de diagnóstico utilizando una terminología habitual en medicina. Es evidente que la terminología puede trasladarse a otras ramas de la ciencia y de la técnica, en particular a la ingeniería. La terminología a usar incluye términos como:

Hallazgo, con el cual nos referiremos a la determinación del valor de una variable predictora X_r . Así por ejemplo x_r (Valor de la variable X_r) puede estar representando la existencia de vómitos en un determinado enfermo.



Evidencia, denota el conjunto de todos los hallazgos para un determinado individuo. Es decir $x = (x_1, \dots, x_n)$ puede estar denotando (si $n = 4$) que el individuo en cuestión es joven, hombre, presenta vómitos y además no tiene antecedentes familiares.

Diagnóstico, denota el valor que toman las m variables aleatorias Y_1, \dots, Y_m , cada una de las cuales se refiere a una enfermedad.

Probabilidad a priori del diagnóstico, $p(y)$ o $p(Y_1 = y_1, \dots, Y_m = y_m)$, se refiere a la probabilidad de un diagnóstico concreto, cuando no se conoce nada acerca de los hallazgos, es decir, cuando se carece de evidencia.

Probabilidad a posteriori de un diagnóstico, $p(y|x)$ o $p(Y_1 = y_1, \dots, Y_m = y_m | X_1 = x_1, \dots, X_m = x_m)$, es decir, la probabilidad de un diagnóstico concreto cuando se conocen n hallazgos (evidencia).

En el planteamiento clásico del diagnóstico (véase Tabla 2) se supone que los m diagnósticos posibles son no excluyentes, es decir, pueden ocurrir a la vez, siendo cada uno de ellos dicotómico. Para fijar ideas en relación con el ámbito médico, podemos pensar que cada uno de los m posibles diagnósticos no excluyentes se relaciona con una enfermedad, pudiendo tomar dos valores: 0 (no existencia) y 1 (existencia). Por lo que se refiere a los n hallazgos o síntomas, se representarán por medio de las n variables aleatorias X_1, \dots, X_n y también asumiremos que cada variable predictora es dicotómica, con valores 0 y 1. El valor 0 en la variable X_i indica la ausencia del i -ésimo hallazgo o síntoma mientras que el valor 1 indica la presencia del hallazgo o síntoma correspondiente.

	X_1	X_n	Y_1	Y_m
$(x^{(1)}, y^{(1)})$	$x_1^{(1)}$	$x_n^{(1)}$	$y_1^{(1)}$	$y_m^{(1)}$
$(x^{(2)}, y^{(2)})$	$x_1^{(2)}$	$x_n^{(2)}$	$y_1^{(2)}$	$y_m^{(2)}$
.....
$(x^{(N)}, y^{(N)})$	$x_1^{(N)}$	$x_n^{(N)}$	$y_1^{(N)}$	$y_m^{(N)}$

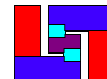
Tabla 2: Problema clásico de diagnóstico.

El problema del diagnóstico consiste en encontrar el diagnóstico más probable a posteriori, una vez conocido el valor de la evidencia. En notación matemática el diagnóstico óptimo, (y_1^*, \dots, y_m^*) será aquel que verifique:

$$(y_1^*, \dots, y_m^*) = \arg \max_{(y_1, \dots, y_m)} p(Y_1 = y_1, \dots, Y_m = y_m | X_1 = x_1, \dots, X_m = x_m)$$

Aplicando el teorema de Bayes para calcular $p(Y_1 = y_1, \dots, Y_m = y_m | X_1 = x_1, \dots, X_m = x_m)$ obtenemos:

$$p(Y_1 = y_1, \dots, Y_m = y_m | X_1 = x_1, \dots, X_m = x_m) = \frac{p(Y_1 = y_1, \dots, Y_m = y_m) p(X_1 = x_1, \dots, X_m = x_m | Y_1 = y_1, \dots, Y_m = y_m)}{\sum_{y'_1, \dots, y'_m} p(Y_1 = y'_1, \dots, Y_m = y'_m) p(X_1 = x_1, \dots, X_m = x_m | Y_1 = y'_1, \dots, Y_m = y'_m)}$$



Veamos a Continuación el número de parámetros que se deben estimar para poder especificar el paradigma anterior y de esa forma obtener el valor de (y_1^*, \dots, y_m^*) . Es importante tener en cuenta que la estimación de cada uno de los parámetros anteriores se deberá efectuar a partir del archivo de N casos, reflejado en la Tabla 4.

Para estimar $p(Y_1 = y_1, \dots, Y_m = y_m)$, y teniendo en cuenta que cada variable Y_i es dicotómica, necesitaremos un total de $2^m - 1$ parámetros. De igual forma, por cada una de las distribuciones de probabilidad condicionada, $p(X_1 = x_1, \dots, X_m = x_m | Y_1 = y_1, \dots, Y_m = y_m)$, se necesitan estimar $2^n - 1$ parámetros. Al tener un total de 2^m de tales distribuciones de probabilidad condicionadas, debemos estimar $(2^n - 1) 2^m$ parámetros. Es decir, que el número total de parámetros necesarios para determinar un modelo concreto del paradigma clásico de diagnóstico es: $2^n - 1 + 2^m (2^n - 1)$. Para hacernos una idea del número de parámetros a estimar podemos consultar la Tabla 3, en la cual vemos de manera aproximada el número de parámetros a estimar para distintos valores de m (número de enfermedades) y n (número de hallazgos).

M	n		Parámetros
3	10	≈	$8 \cdot 10^3$
5	20	≈	$33 \cdot 10^6$
10	50	≈	$11 \cdot 10^{17}$

Tabla 3: Número de parámetros a estimar, en función de m (número de enfermedades) y n (número de síntomas), en el paradigma clásico de diagnóstico.

Ante la imposibilidad de poder estimar el elevado número de parámetros que se necesitan en el paradigma clásico de diagnóstico, en lo que sigue se simplificaran las premisas sobre las que se ha construido dicho paradigma.

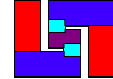
En primer Lugar vamos a considerar que los diagnósticos son excluyentes, es decir, que dos diagnósticos no pueden darse al unísono. Esto trae como consecuencia que en lugar de considerar el diagnóstico como una variable aleatoria m-dimensional, este caso puede verse como una única variable aleatoria unidimensional siguiendo una distribución polinomial con m valores posibles.

Vamos a denotar por X_1, \dots, X_n a las n variables predictoras. Supongamos que todas ellas sean binarias. Denotamos por C la variable de diagnóstico, que suponemos puede tomar m posibles valores. La tabla 4 refleja la situación anterior. La búsqueda del diagnostico mas probable a posteriori, c^* , una vez conocidos los síntomas de un determinado paciente, $x = (x_1, \dots, x_n)$, puede plantearse como la búsqueda del estado de la variable C con mayor probabilidad a posteriori. Es decir

$$c^* = \arg \max_c p(C = c | X_1 = x_1, \dots, X_m = x_m)$$

El cálculo de $p(C = c | X_1 = x_1, \dots, X_m = x_m)$ puede llevarse a cabo utilizando el teorema de Bayes, y ya que el objetivo es calcular el estado de C, c^* , con mayor probabilidad a posteriori, no es necesario calcular el denominador del teorema de Bayes. Es decir,

$$p(C = c | X_1 = x_1, \dots, X_m = x_m) \propto p(C=c) p(X_1 = x_1, \dots, X_m = x_m | C = c)$$



Por tanto, en el paradigma en el que los distintos diagnósticos son excluyentes, y considerando que el número de posibles diagnósticos es m , y que cada variable predictora X_i es dicotómica, tenemos que el número de parámetros a estimar es $(m - 1) + m (2^n - 1)$, de los cuales:

- ❖ $m - 1$ se refiere a las probabilidades a priori de las variable C ;
- ❖ $m (2^n - 1)$ se relacionan con las probabilidades Condicionadas de cada posible combinación de las variables predictoras dado cada posible valor de la variable.

La Tabla 4 nos da una idea del número de parámetros a estimar para distintos valores de m y n .

M	n		Parámetros
3	10	≈	$3 \cdot 10^3$
5	20	≈	$5 \cdot 10^6$
10	50	≈	$11 \cdot 10^{15}$

Tabla 4: Número de parámetros a estimar, en función de m (número de enfermedades) y n (número de síntomas), en el paradigma clásico de diagnóstico con diagnósticos excluyentes.

Vemos de nuevo que el número de parámetros a estimar sigue siendo elevado, de hay que necesitamos imponer suposiciones más restrictivas para que los paradigmas puedan convertirse en modelos implementables.

Vamos finalmente a introducir el paradigma naïve Bayes: diagnósticos excluyentes y hallazgos condicionalmente independientes dado el diagnóstico. El paradigma naïve Bayes se basa en dos premisas establecidas sobre las variables predictoras (hallazgos, síntomas) y la variaba a predecir (diagnósticos). Dichas premisas son:

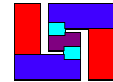
Los diagnósticos son excluyentes, es decir, la variable C a predecir toma uno de sus m posibles valores: c_1, \dots, c_m ;

Los hallazgos son coincidentalmente independientes dado el diagnóstico, es decir, que si una conoce, el valor de la variable diagnóstico, el conocimiento del valor de cualquiera de los hallazgos es irrelevante para el resto de los hallazgos. Esta condición se expresa matemáticamente por medio de la fórmula:

$$p(X_1 = x_1, \dots, X_m = x_m | C = c) = \prod_{i=1}^n p(X_i = x_i | C = c)$$

ya que por medio de la regla de la cadena se obtiene:

$$p(X_1 = x_1, \dots, X_m = x_m | C = c) = p(X_1 = x_1 | X_2 = x_2, \dots, X_n = x_n, C = c) \\ p(X_2 = x_2 | X_3 = x_3, \dots, X_n = x_n, C = c) \\ \dots \dots p(X_n = x_n | C = c)$$



Por otra parte teniendo en cuenta la independencia condicional entre las variables predictoras dada la variable clase, se tiene que:

$$p(X_i = x_i | X_{i+1} = x_{i+1}, \dots, X_n = x_n, C = c) = p(X_i = x_i | C = c)$$

Por todo $i = 1, \dots, n$. De ahí que se verifique la ecuación.

Por tanto, en el paradigma naïve Bayes, la búsqueda del diagnóstico más probable, c^* , una vez conocidos los síntomas (x_1, \dots, x_n) determinado paciente, se reduce a:

$$\begin{aligned} c^* &= \arg \max_c p(C = c | X_1 = x_1, \dots, X_n = x_n) \\ &= c^* = \arg \max_c p(C = c) \prod_{i=1}^n p(X_i = x_i | C = c) \end{aligned}$$

Suponiendo que todas las variables predictoras son dicotómicas, el número de parámetros necesarios para especificar un modelo naïve Bayes resulta ser $(m - 1) + m n$, ya que: Se necesitan $(m - 1)$ parámetros para especificar la probabilidad a priori de la variable C . Para cada variable predictoras X_i se necesitan m parámetros para determinar las distribuciones de probabilidad condicionadas.

Con los números reflejados en la tabla 5 nos podemos hacer una idea del número de parámetros necesarios en función del número de posibles diagnósticos y del número de síntomas necesarios para especificar el paradigma naïve Bayes.

M	n	parámetros
3	10	32
5	20	104
10	50	509

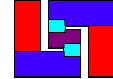
Tabla 5: Número de parámetros a estimar en el paradigma naïve Bayes en función del número de diagnósticos posibles (m) y del número de síntomas (n).

En el caso de que las n variables predictoras X_1, \dots, X_n sean continuas, se tiene que el paradigma naïve Bayes se convierte en buscar el valor de la variable C , que denotamos por c^* , que maximiza la probabilidad a posteriori de la variable C , dada la evidencia expresada como una instanciación de las variables X_1, \dots, X_n , esto es, $x = (x_1, \dots, x_n)$.

Es decir, el paradigma naïve Bayes con variables continuas trata de encontrar c^* verificando:

$$\begin{aligned} c^* &= \arg \max_c p(C = c | X_1 = x_1, \dots, X_n = x_n) \\ &= c^* = \arg \max_c p(C = c) \prod_{i=1}^n f_{X_i | C = c}(x_i | c) \end{aligned}$$

donde $f_{X_i | C = c}(x_i | c)$ denota, para todo $i = 1, \dots, n$, la función de densidad de la variable X_i condicionada a que el valor del diagnóstico sea c .



Suele ser habitual utilizar una variable aleatoria normal (para cada valor de C) para modelar el comportamiento de la variable X_i . Es decir, para todo c, y para todo $i \in \{1, \dots, n\}$, asumimos

$$f_{X_i | C=c}(x_i | c) \sim N(x_i; \mu_i^c; (\sigma_i^c)^2)$$

En tal caso el paradigma naïve Bayes obtiene c^* como:

$$c^* = \arg \max_c p(C=c) \prod_{i=1}^n \left| \frac{1}{\sqrt{2\pi}\sigma_i^c} e^{-1/2((x_i - \mu_i^c)/\sigma_i^c)^2} \right|$$

En este caso el número de parámetros a estimar es $(m - 1) + 2mn$:

- ❖ $m - 1$ en relación con las probabilidades a priori $p(C=c)$
- ❖ $2nm$ en relación con las funciones de densidad condicionadas

Finalmente puede ocurrir que algunos de los hallazgos se recojan en variables discretas mientras que otros hallazgos sean continuos. En tal caso hablaremos del paradigma naïve Bayes con predictoras continuas y discretas.

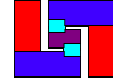
Supongamos que de las n variables predictoras, n_1 de ellas, X_1, \dots, X_{n_1} , sean discretas, mientras que el resto $n - n_1 = n_2$, Y_1, \dots, Y_{n_2} , sean continuas. En principio al aplicar directamente la formula del paradigma naïve Bayes correspondiente a esta situación se obtiene:

$$p(c | x_1, \dots, x_{n_1}, y_1, \dots, y_{n_2}) \propto p(c) \prod_{i=1}^{n_1} p(x_i | c) \prod_{j=1}^{n_2} f(y_j | c)$$

Esta expresión puede propiciar el conceder una mayor importancia a las variables continuas, ya que mientras que $p(x_i | c)$ verifica $0 \leq p(x_i | c) \leq 1$, puede ocurrir que $f(y_j | c) > 1$. Con objeto de evitar esta situación, proponemos la normalización de la aportación de las variables continuas dividiendo cada uno de los factores correspondientes por el $\max_y f(y_j | c)$. Obtenemos por tanto:

$$p(c | x_1, \dots, x_{n_1}, y_1, \dots, y_{n_2}) \propto p(c) \prod_{i=1}^{n_1} p(x_i | c) \prod_{j=1}^{n_2} \frac{f(y_j | c)}{\max_y f(y_j | c)}$$

En el caso en que las funciones de densidad de las variables continuas condicionadas a cada posible valor de la variable clase sigan distribuciones normales, es decir si $Y_j | C=c \sim N(y_j; \mu_j^c, (\sigma_j^c)^2)$, se tiene que la formula se expresa de la siguiente manera:



$$p(c | x_1, \dots, x_{n1}, y_1, \dots, y_{n2}) \propto p(c) \prod_{i=1}^{n_1} p(x_i | c) \prod_{j=1}^{n_2} e^{-1/2 (y_j - \mu_{cj} / \sigma_{cj})^2}$$

La Figura 1: refleja la estructura gráfica de un modelo naïve Bayes.

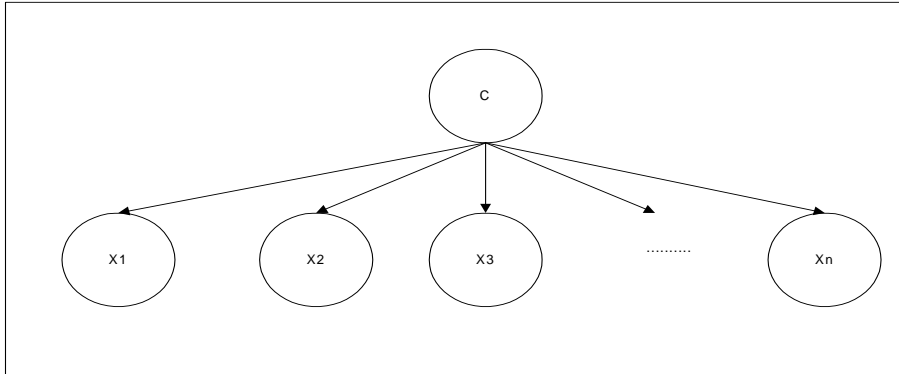


Figura 1: Estructura gráfica de un modelo naïve Bayes.

1.2.1.3. Naïve Bayes Aumentado a Árbol (TAN)

En esta sección vamos a presentar algunos trabajos que construyen clasificadores con estructura naïve Bayes aumentada a árbol (Tree Augmented Network (TAN)). Para obtener este tipo de estructura se comienza por una estructura de árbol con las variables predictoras, para posteriormente conectar la variable clase con cada una de las variables predictoras. La Figura 1 ilustra un ejemplo de estructura naïve Bayes aumentada a árbol.

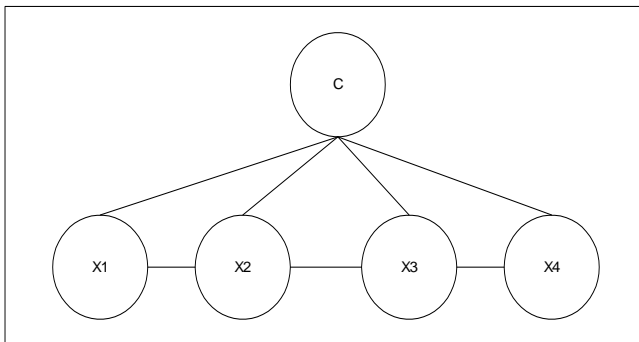
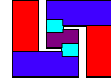


Figura 2: Ejemplo de estructura naïve Bayes aumentada a árbol (Tree Augmented Network (TAN))

Fridman y Col. (1997) presentan un algoritmo denominado Tree Augmented Network (TAN) el cual consiste básicamente en una adaptación del algoritmo de Chow-Liu (1968). En dicho algoritmo se tiene en cuenta la cantidad de información mutua condicionada a la variable clase, en lugar de la cantidad de información mutua en la que se basa el algoritmo de Chow-Liu. La cantidad de información mutua entre las variables discretas X e Y condicionada a la variable C se define como:

$$I(X, Y | C) = \sum_{i=1} \sum_{j=1} \sum_{r=1} p(x_i, y_j, c_r) \log \frac{p(x_i, y_j, c_r)}{p(x_i | c_r) p(y_j | c_r)}$$



Tal y como puede verse en el pseudocódigo de la Figura 5, TAN consta de cinco pasos. En el primer paso se calculan las cantidades de información mutua para cada par de variables (X_i, X_j) condicionadas por la variable C . A continuación se debe construir un grafo no dirigido completo con n nodos, uno por cada una de las variables predictoras, en el cual el peso de cada arista viene dado por la cantidad de información mutua entre las dos variables unidas por la arista condicionada a la variable clase. El algoritmo de Kruskal parte de los $n(n-1)/2$ pesos obtenidos en el paso anterior para construir el árbol expandido de máximo peso de la siguiente manera:

1. Asignar las dos aristas de mayor peso al árbol a construir.
2. Examinar la siguiente arista de mayor peso, y añadirla al árbol a no ser que forme un ciclo, en cuyo caso se descarta y se examina la siguiente arista de mayor peso.
3. Repetir el paso 2 hasta que se hayan seleccionado $n-1$ aristas.

Las propiedades teóricas de este algoritmo de construcción de TAN son análogas a las del algoritmo de Chow-Liu (1968). Es decir, si los datos han sido generados por una estructura Tree Augmented Network, el algoritmo TAN es asintóticamente correcto, en el sentido de que si la muestra de casos es suficientemente grande, recuperará la estructura que generó el archivo de casos. En la Figura 3a se muestra un ejemplo de aplicación del algoritmo.

<p>Paso 1. Calcular $I(X_i, X_j C)$ con $i < j, i, j = 1, \dots, n$</p> <p>Paso 2. Construir un grafo no dirigido completo cuyos nodos correspondan a las variables predictoras X_1, \dots, X_n. Asignar a cada arista conectando las variables X_i y X_j un peso dado por $I(X_i, X_j C)$</p> <p>Paso 3. A partir del grafo completo anterior y siguiendo el algoritmo de Kruskal construir un árbol expandido de máximo peso</p> <p>Paso 4. transformar el árbol no dirigido resultante en uno dirigido, escogiendo una variable como raíz, para a continuación direccionar el resto de las aristas</p> <p>Paso 5. Construir un modelo TAN añadiendo un nodo etiquetado como C y posteriormente un arco desde C a cada variable predictora X_i</p>
--

Figura 3: Pseudocódigo del algoritmo TAN (Friedman y Col, 1997).

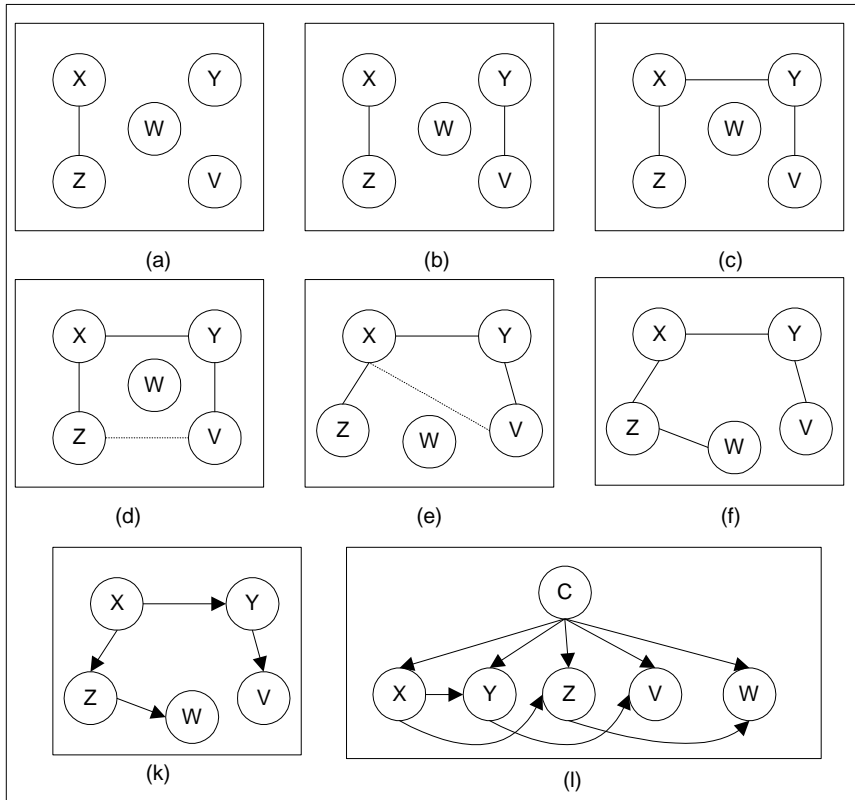
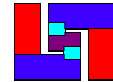
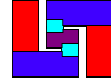


Figura 3a: Ilustración del algoritmo TAN con cinco variables predictoras X, Y, Z, V y W. Se supone que el orden de las cantidades de información mutuas condicionadas ha sido: $I(X, Z | C) > I(Y, V | C) > I(X, Y | C) > I(Z, v | C) > I(X, V | C) > I(Z, W | C) > I(X, Y | C) > I(X, W | C) > I(Y, Z | C) > I(Y, W | C) > I(V, W | C)$. Las subfiguras (a) a (f) corresponden a la aplicación del algoritmo de Kruskall. La subfigura (g) corresponde al Paso 4 del algoritmo TAN y finalmente en la subfigura (h) se realiza el Paso 5 de TAN. El modelo clasificatorio obtenido es: $P(C | x, y, z, v, w) \propto p(c) p(x | c) p(y | x, c) p(z | x, c) p(v | y, c) p(w | z, c)$

Keogh y Pazzani (1999) proponen un algoritmo voraz que va añadiendo arcos a una estructura naïve Bayes. En cada paso se añade el arco que, mantiene la condición de que en la estructura final cada variable no tenga más de un padre, mejore en mayor medida el porcentaje de bien clasificados obtenido mediante el mismo.

1.2.1.4. Clasificadores Bayesianos k Dependientes (KDB)

Sahami (1996) presenta un algoritmo denominado k Dependence Bayesian classifier (KDB) el cual posibilita atravesar el amplio espectro de dependencias disponibles entre el modelo naïve Bayes y el modelo correspondiente a una red Bayesiana completa. El algoritmo se fundamenta en el concepto de clasificador Bayesiano K-dependiente, el cual contiene la estructura del clasificador naïve Bayes y permite a cada variable predictorora tener un máximo de K variables padres sin contar a la variable clase. De esta manera, el modelo naïve Bayes se corresponde con un clasificador Bayesiano 0-dependiente, el modelo TAN sería un clasificador Bayesiano 1-dependiente y el clasificador Bayesiano completo (en la estructura no se refleja ninguna independencia) correspondería a un clasificador Bayesiano (n - 1)- dependiente. El pseudocódigo del algoritmo KDB puede consultarse en la Fig. 4.



Paso 1. Para cada variable predictora X_i $i = 1, \dots, n$ calcular la cantidad de información mutua con respecto a la clase C , $I(X_i, C)$

Paso 2. Para cada par de variables predictoras calcular la cantidad de información mutua condicionada a la clase, $I(X_i, X_j | C)$, con $i \neq j$, $i, j = 1, \dots, n$

Paso 3. Inicializar a vacío la lista de variables usando N

Paso 4. Inicializar la red Bayesiana a construir, BN , con un único nodo, el correspondiente a la variable C

Paso 5. Repetir hasta que N incluya a todas las variables del dominio:

Paso 5.1. Seleccionar de entre las variables que no están en N , aquella X_{\max} con mayor cantidad de información mutua respecto a C , $I(X_{\max}, C) = \max_{X \notin N} I(X, C)$

Paso 5.2. Añadir un nodo a BN , $X \notin N$ representando X_{\max}

Paso 5.3. Añadir un arco C a X_{\max} en BN

Paso 5.4. Añadir $m = \min(|N|, k)$ arcos de las m variables distintas X_j en que tengan los mayores valores $I(X_{\max}, X_j | C)$

Paso 5.5. Añadir X_{\max} a N

Paso 6. Computar las probabilidades condicionadas necesarias para especificar la red Bayesiana BN

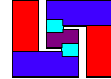
Figura 4: Pseudocódigo del algoritmo KDB (Sahami, 1996).

La idea básica del algoritmo consiste en generalizar el algoritmo propuesto por Fridman y Col (1997) permitiendo que cada variable tenga un número de padres, sin contar la variable clase C , acotado por k . El autor comenta una posible mejora del algoritmo flexibilizando la determinación de k por medio de la obtención de un umbral de cantidad de información mutua, el cual debería de ser sobrepasado para que el correspondiente arco fuese incluido.

2. Obtención de redes bayesianas a través de Elvira

El programa Elvira es fruto de un proyecto de investigación financiado por la CICYT y el Ministerio de Ciencia y Tecnología Español, en el que participan investigadores de varias universidades españolas y de otros centros.

El programa Elvira está destinado a la edición y evaluación de modelos gráficos probabilistas, concretamente redes bayesianas y diagramas de influencia. Elvira cuenta con un formato propio para la codificación de los modelos, un lector-intérprete para los modelos codificados, una interfaz gráfica para la construcción de redes, con opciones específicas para modelos canónicos (puertas OR, AND, MAX, etc.), algoritmos exactos y aproximados (estocásticos) de razonamiento tanto para variables discretas como continuas, métodos de explicación del razonamiento, algoritmos de toma de decisiones, aprendizaje de modelos a partir de bases de datos, fusión de redes, etc. Elvira está escrito y compilado en el lenguaje Java, lo cual permite que funcione en diferentes plataformas y sistemas operativos (MS-DOS/Windows, linux, Solaris, etc.).



2.1. Uso de Elvira

Este programa permite el ingreso de las redes Bayesianas de dos formas: (a) por un lado el ingreso manual, donde el usuario dibuja la red bayeasiana en la pantalla y carga los valores de probabilidad asociados a cada nodo, (b) mediante la importación de archivos de casos.

El presente trabajo se centrará en la generación de redes Bayesianas a partir de un archivo de datos, ya que esta operativa es la que se vincula con la minería de datos.

2.1.1. Primeros Pasos

El programa Elvira puede obtenerse gratis desde Internet a través de: <http://leo.ugr.es/elvira>.

Para poder utilizarlos se debe instalar previamente “La Máquina Virtual” de Java, que será el encargado de interpretar el programa. Este Software puede obtenerse desde la página “www.java.com”, una vez instalada la “Máquina Virtual”, lo que resta es descomprimir el archivo Elvira.zip que automáticamente generará una carpeta Elvira donde se le indique (por ejemplo C:); dentro de la carpeta Elvira aparecerá un archivo *Elvira.jar* el cual se ejecutará con solo hacerle doble clic .

Una vez iniciado Elvira aparecerá una pantalla como la que se muestra en la figura 5.

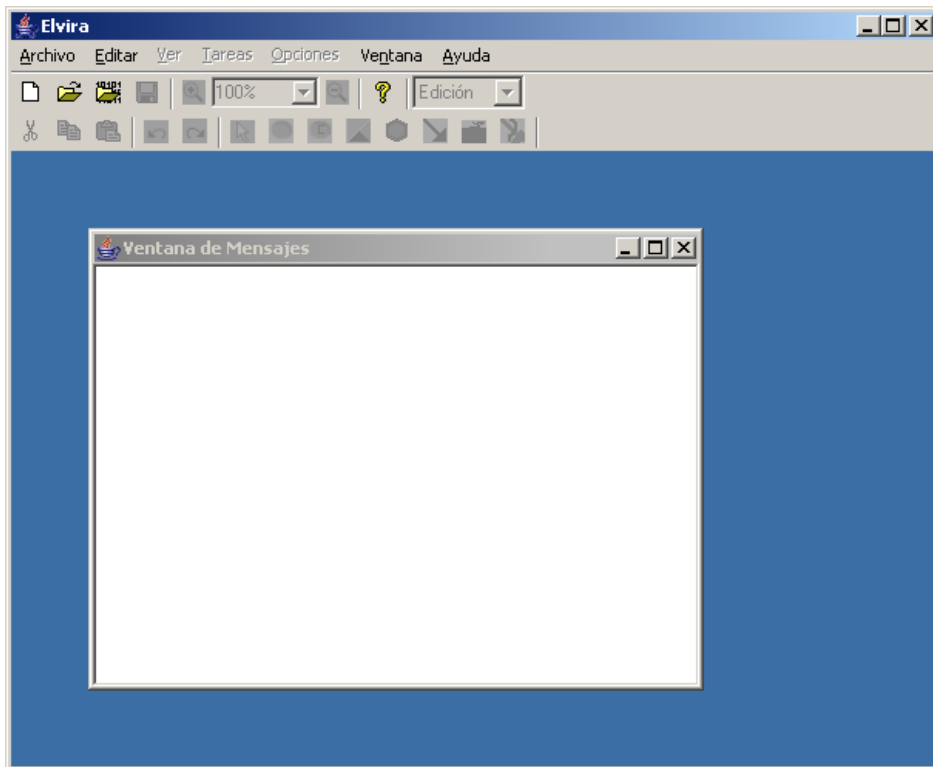
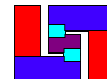


Figura 5: pantalla principal

Aquí la ventana de mensajes que aparece activa será utilizada para que el programa detalle mensaje a indicar al operador por inconvenientes con el uso del mismo.



A continuación vamos a importar un archivo de datos. Los archivos a importar deberán estar en formato .csv (texto plano con los campos dentro del mismo separados por “;” o “,”). El sistema asume que el primer registro del archivo contiene el nombre de cada nodo, este punto es muy importante ya que si se omite el sistema asumirá como nombre de los nodos el valor de las variables indicadas en la primera fila.

A continuación, en la figura 6, se muestra un fragmento de un archivo a importar con Elvira, el mismo lleva en el primer registro el nombre de los campos que se detallan en los subsiguientes registros:

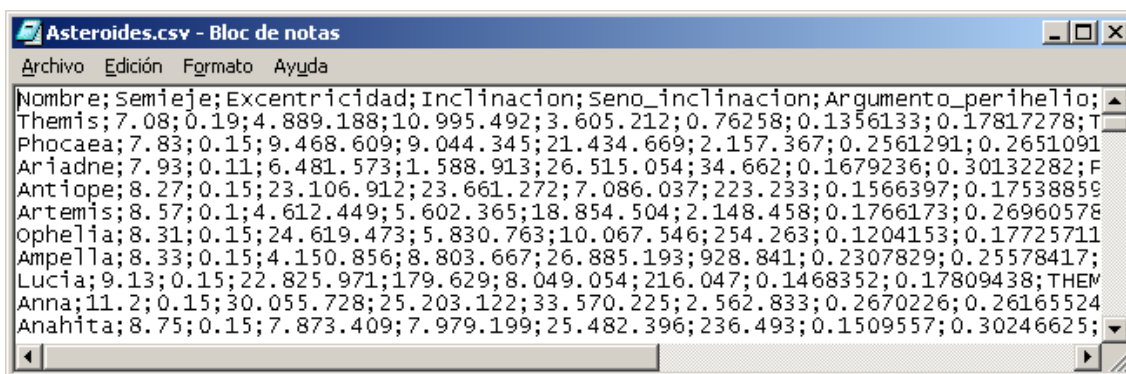


Figura 6: archivo de datos a importar

Para ingresar esta información al sistema se debe importar el archivo al formato .csv. para ello desde el menú “Archivo” se debe seleccionar la opción “Importar Fichero de Casos”, como se muestra a continuación en la figura 7:

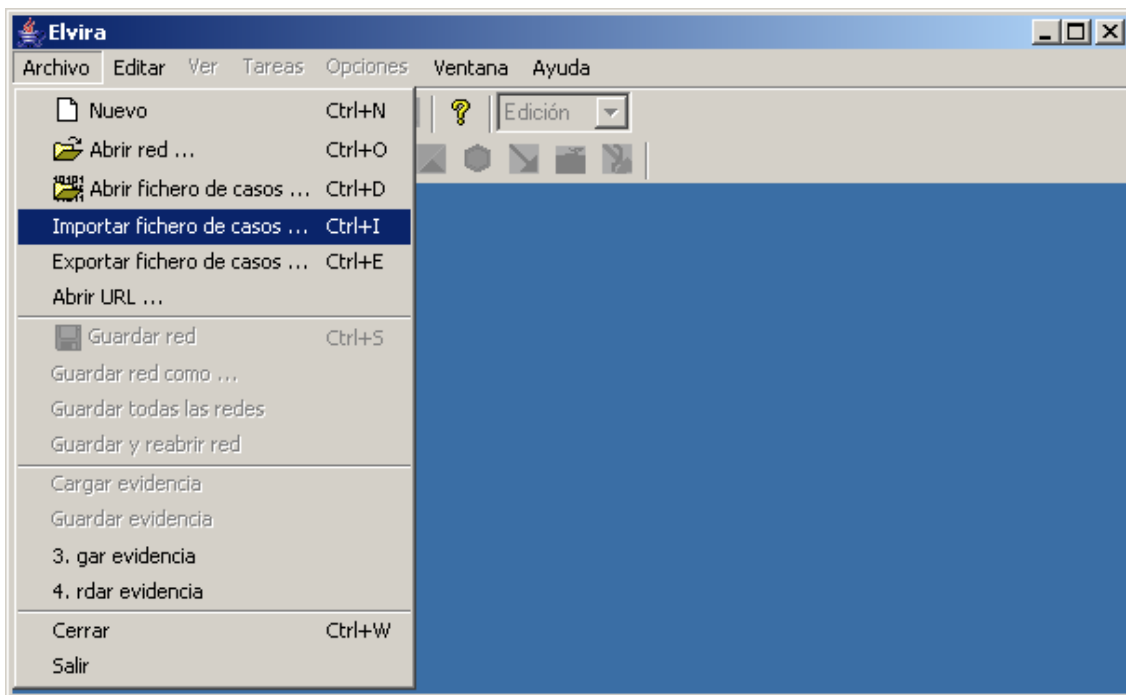
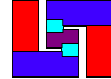


Figura 7: Ingreso a la opción de importación de fichero



Esta opción desplegará una ventana con tres solapas, a continuación, en la figura 8, se describe la primera de ellas:

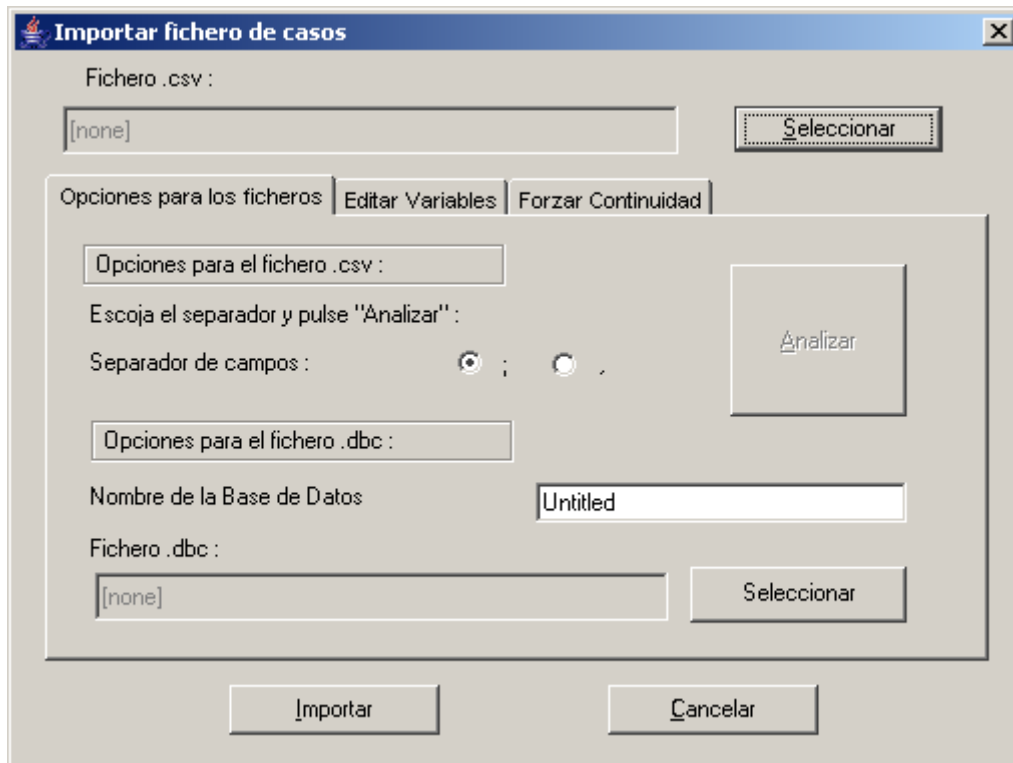


Figura 8: menú de importación de fichero, opciones de importación

El botón seleccionar permite indicar la ubicación del archivo a procesar. Opcionalmente desde esta solapa se podrá indicar el carácter de separación entre campos (“;” o “;”) y analizar el estado del archivo (mediante el botón analizar) para ver si el mismo podrá ser importado o no. Por último se debe seleccionar la ubicación y el nombre del nuevo archivo a crear como resultado de la importación de los datos.

Si el análisis previo del archivo da un resultado favorable, se está en condiciones de seleccionar el botón “Importar” para obtener el archivo .dbc, en caso contrario se podrá analizar la información a importar con mas precisión desde las otras dos solapas que se describen a continuación:

Solapa “editar variables”: Aquí en función del nombre que cada variable tenga en el primer registro del archivo se podrá modificar su nombre, editar sus posibles estados o ingresar algún comentario. A continuación, en la figura 9, se detalla el contenido de esta solapa:

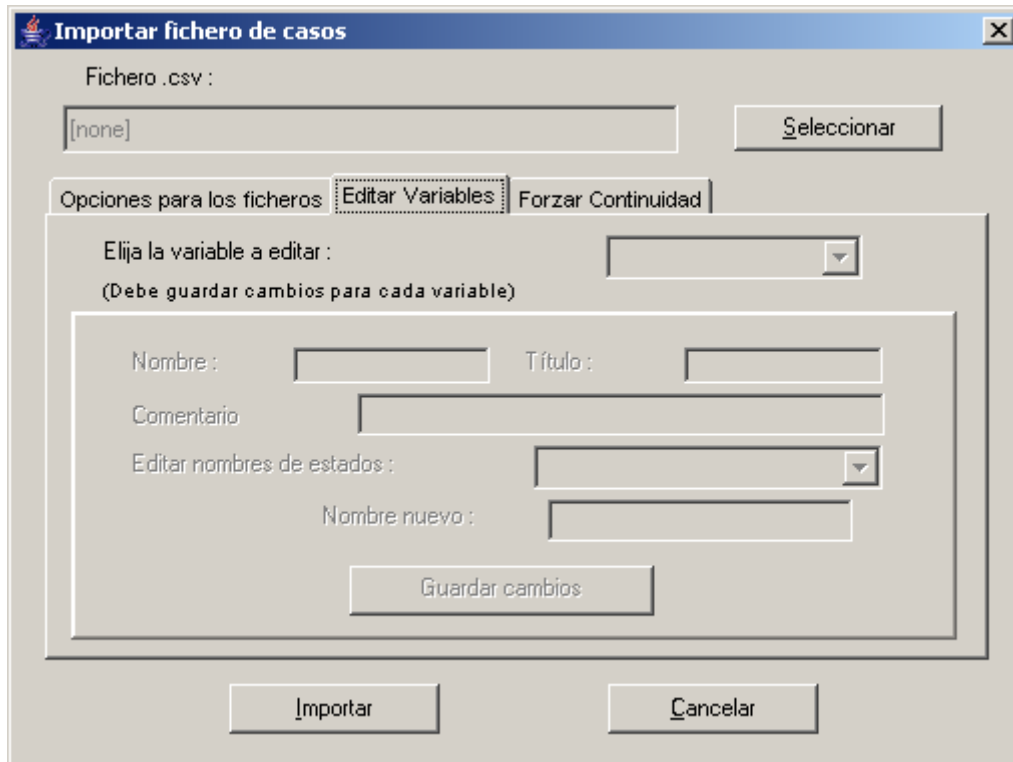
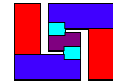


Figura 9: Menú de importación de fichero, edición de variables

Solapa “Forzar Continuidad”: Aquí se analizan las variables de forma similar a la solapa anterior y en caso de que se necesite forzar la continuidad de alguna o todas la variables a importar. A continuación, en la figura 10, se detalla el contenido de esta solapa:

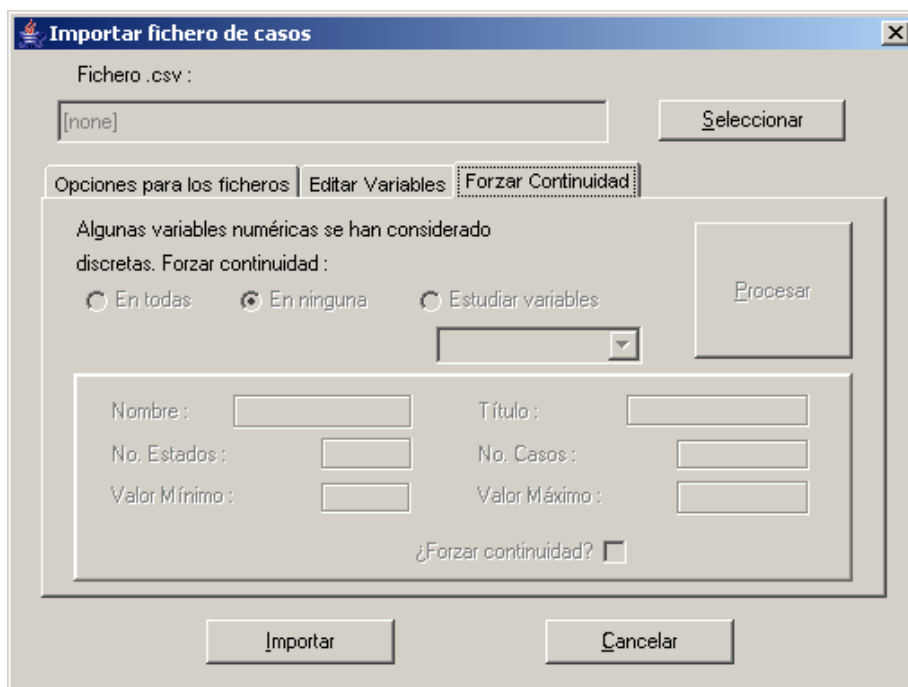
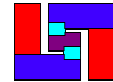


Figura 10: Menú de importación de fichero, forzar continuidad de los datos



A continuación, en la figura 11, se muestra el resultado de haber importado el archivo `creditos.csv`

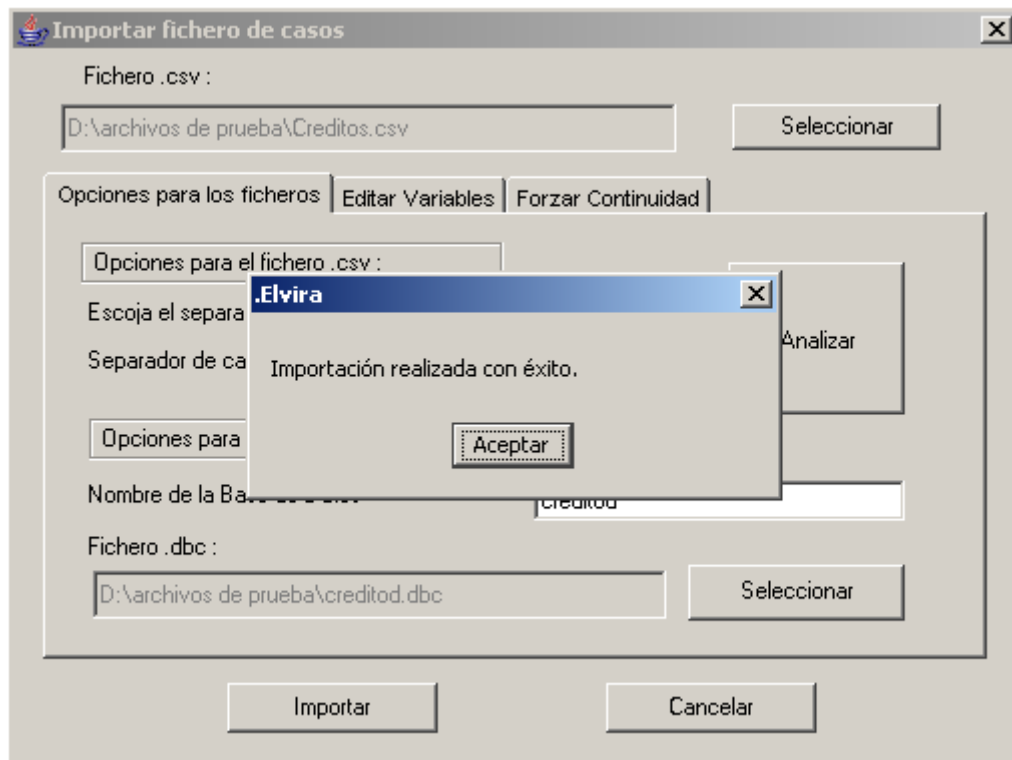


Figura 11: Confirmación de importación

Una vez importado el archivo a Formato `.dbc` podremos comenzar a trabajar con el. Para ello debemos ir nuevamente a el menú “Archivo” y aquí seleccionar la opción “Abrir fichero de Casos”, como se muestra a continuación en la figura 12

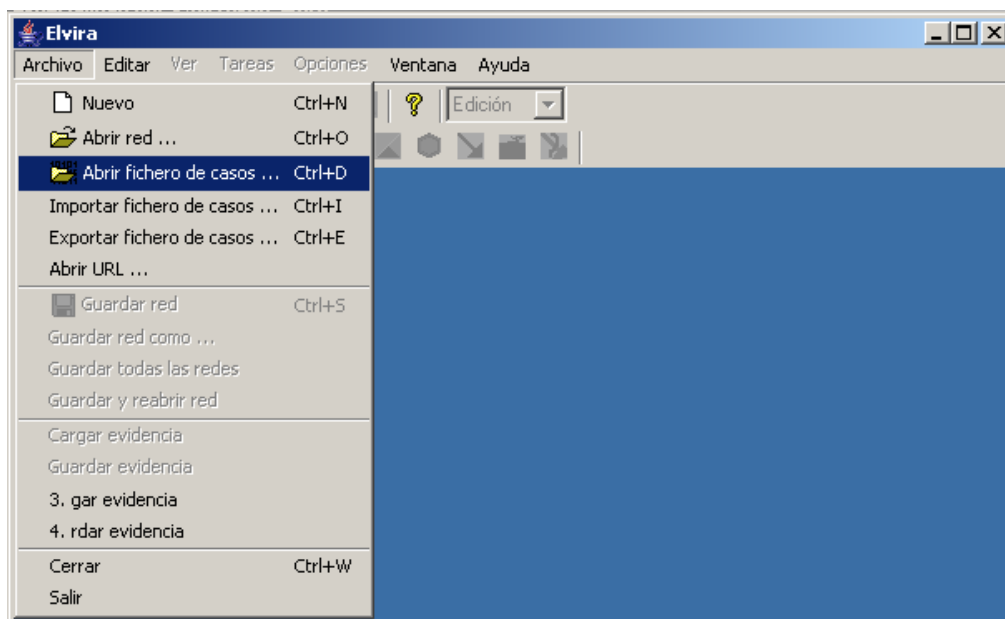
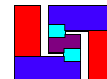


Figura 12: Ingreso a la opción abrir fichero de casos



En esta opción se deberá seleccionar en primer lugar el archivo .dbc a procesar, el cual debe ser ingresado, mediante su selección, en el campo “Fichero de Casos”. Esta pantalla además posee tres solapas, las cuales se describen a continuación:

- ❖ Opciones de preproceso: Aquí se encuentran los filtros que se pueden aplicar al archivo de casos para que luego pueda ser interpretado de forma exitosa por el clasificador.
- ❖ Aprendizaje automático: Aquí se encuentran los clasificadores a aplicar para realizar el aprendizaje (por ejemplo: Naïve Bayes, TAN o KDB)
- ❖ Opciones de Post aprendizaje: Aquí se pueden aplicar opciones de testeo de clasificación, como puede ser por ejemplo aplicar validaciones cruzadas.

A continuación, en la figura 13, se muestra la pantalla del programa:

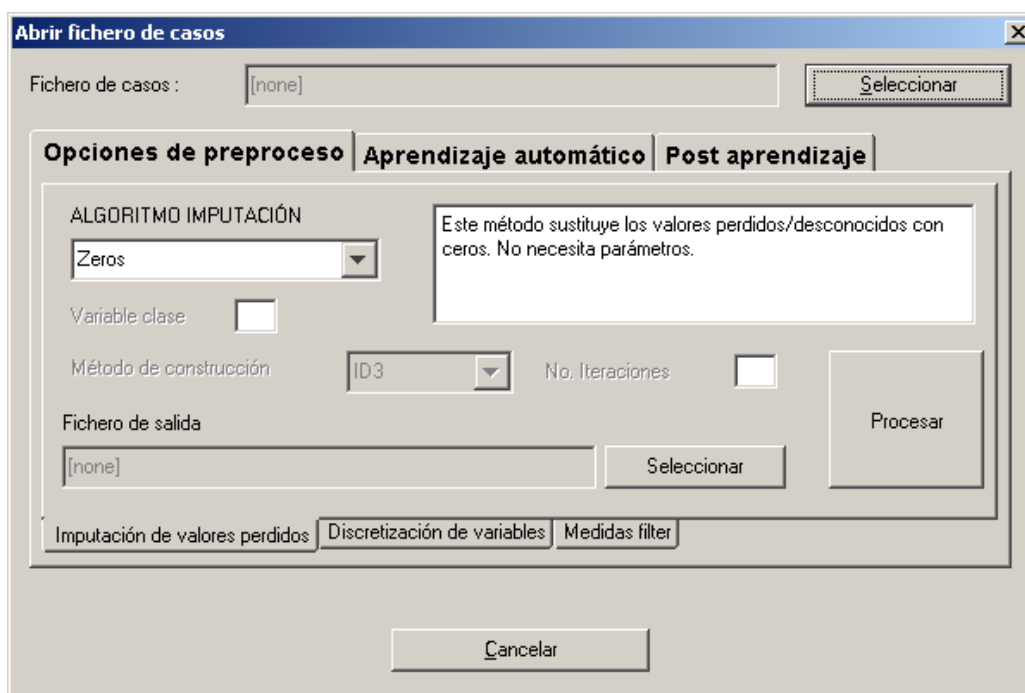


Figura 13: Menú apertura de fichero, opciones de preproceso

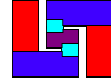
A partir de ahora centraremos la explicación en función del clasificador y el algoritmo de imputación a aplicar al archivo importado para generar la Red Bayesiana.

2.1.2. Ejecución de clasificadores.

Los clasificadores son algoritmos dedicados a interpretar los datos provistos en el archivo de entrenamiento para generar la red Bayesiana. Actualmente existen varios algoritmos de este tipo, el presente trabajo solo analizará los soportados por el programa Elvira, estos fueron descriptos en el punto 1.2 y son Naïve Bayes, TAN y KDB.

2.1.2.1. Caso 1 obtención de una red mediante el clasificador Naïve Bayes

Para este caso en la primer solapa no se hará selección alguna, solo se reemplazara por ceros la falta de valor en alguna de la variables a importar. Esto se hace con el



fin de mostrar como opera el clasificador sin influencia de otros algoritmos que lo pueden complementar en la generación de la red Bayesiana. Por lo tanto pasaremos directamente a la solapa “Aprendizaje automático” (ver figura 14).

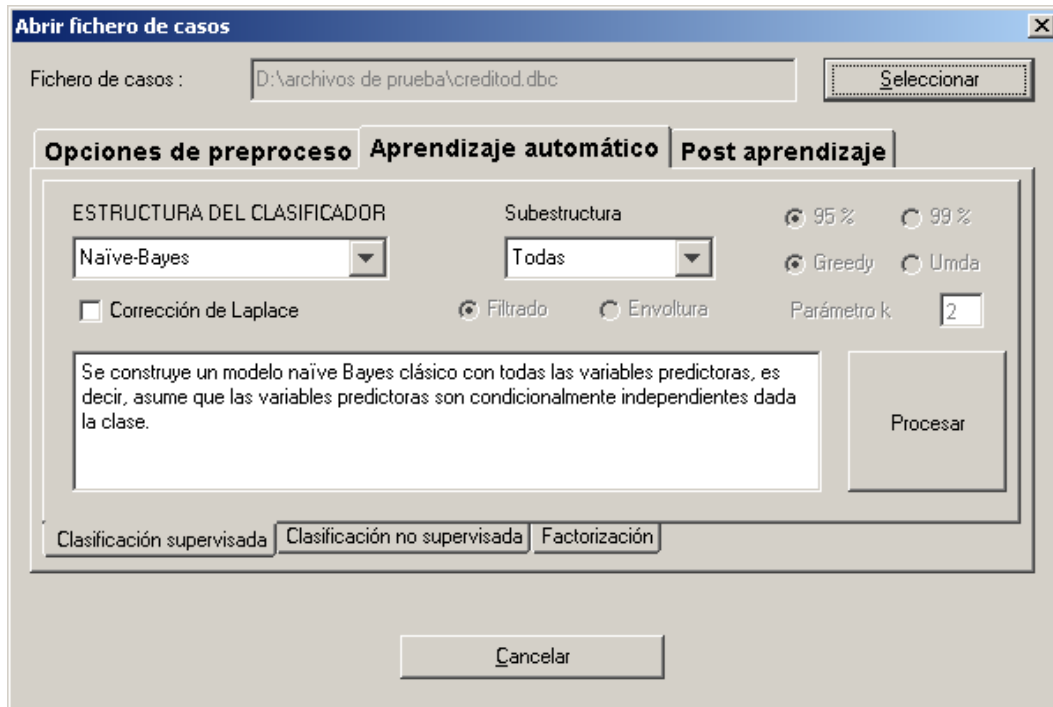


Figura 14: Menú apertura de fichero, opciones de aprendizaje automático

Aquí, continuando con el ejemplo anterior, se ha seleccionado “creditod.dbc” y como por defecto el sistema sugiere utilizar el clasificador Naïve Bayes no hay mas que ejecutar el botón “Procesar”, una vez ejecutado se debe cerrar la ventana de incorporación de archivo y en la ventana principal aparecerá la gráfica de la red generada. A continuación, en la figura 15, se muestra la red obtenida:

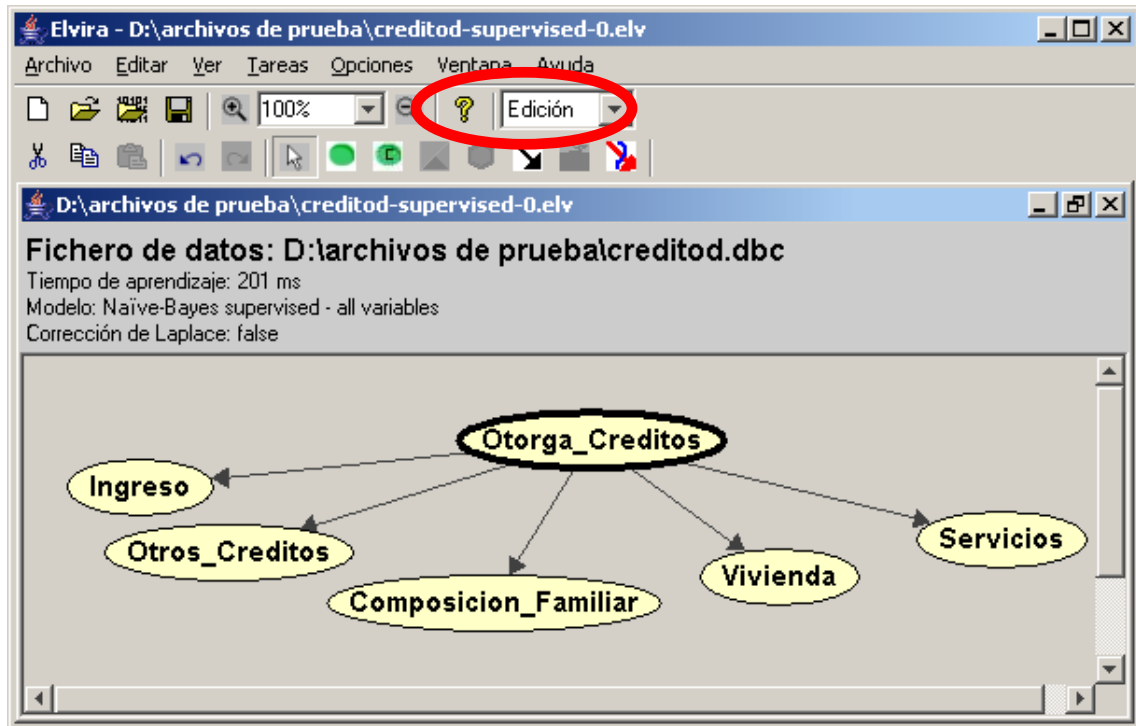
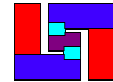


Figura 15: Red Bayesiana en modo edición

Para obtener mayores detalle sobre la red obtenida se debe cambiar, en la barra de tareas, la opción “Edición” por la opción “Inferencia”, a continuación, en la figura 16, se detalla esta última pantalla.

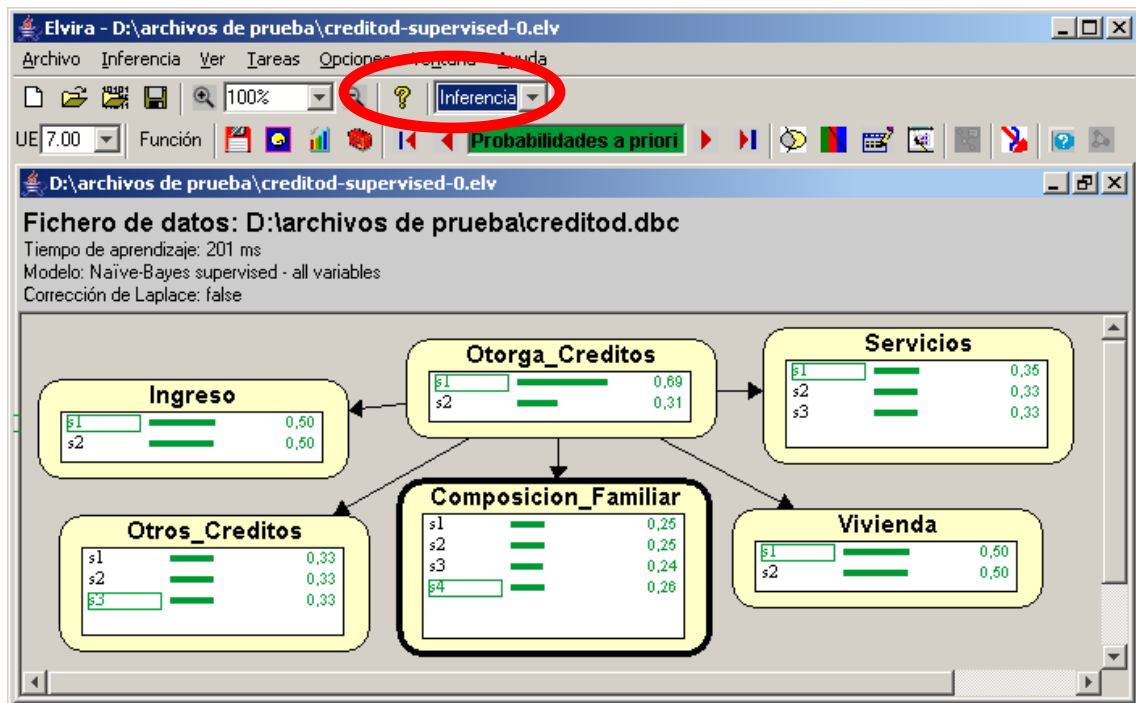
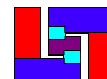


Figura 16: Red Bayesiana en modo inferencia

Aquí pueden verse las probabilidades de cada una de las distintas variable. Una vez analizados todos los tipos de Clasificadores analizaremos con mayor detalle las opciones del menú de Inferencia.



A continuación se detallan los valores de probabilidad asociados a cada nodo en particular

Valor	Probabilidad
S1	0,6944
S2	0,3056

Tabla 6: Nodo: Otorga_Créditos

Nodo Padre	Valores	
Otorga crédito	S1	S2
Valor	Probabilidad	
S1	0,47	0.5682
S2	0,53	0.4318

Tabla 7: Nodo: Ingreso

Nodo Padre	Valores	
Otorga crédito	S1	S2
Valor	Probabilidad	
S1	0.45	0.0682
S2	0.41	0.1591
S3	0.14	0.7727

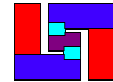
Tabla 8: Nodo: Otros_Créditos

Nodo Padre	Valores	
Otorga crédito	S1	S2
Valor	Probabilidad	
S1	0.25	0.25
S2	0.3	0.1364
S3	0.27	0.1818
S4	0.18	0.4318

Tabla 9: Nodo: Composición_familiar

Nodo Padre	Valores	
Otorga crédito	S1	S2
Valor	Probabilidad	
S1	0.39	0.75
S2	0,61	0.25

Tabla 10: Nodo: Vivienda



Nodo Padre	Valores	
Otorga crédito	S1	S2
Valor	Probabilidad	
S1	0.36	0.3182
S2	0.33	0.3182
S3	0.31	0.3636

Tabla 11: Nodo: Servicios

2.1.2.2. Caso 2 obtención de una red mediante el clasificador TAN

Para generar una red mediante el Clasificador TAN se deberá cambiar la opción “Estructura del Clasificador”, que por defecto sugiere Naïve Bayes, a TAN. Como se indica a continuación, en la figura 17:

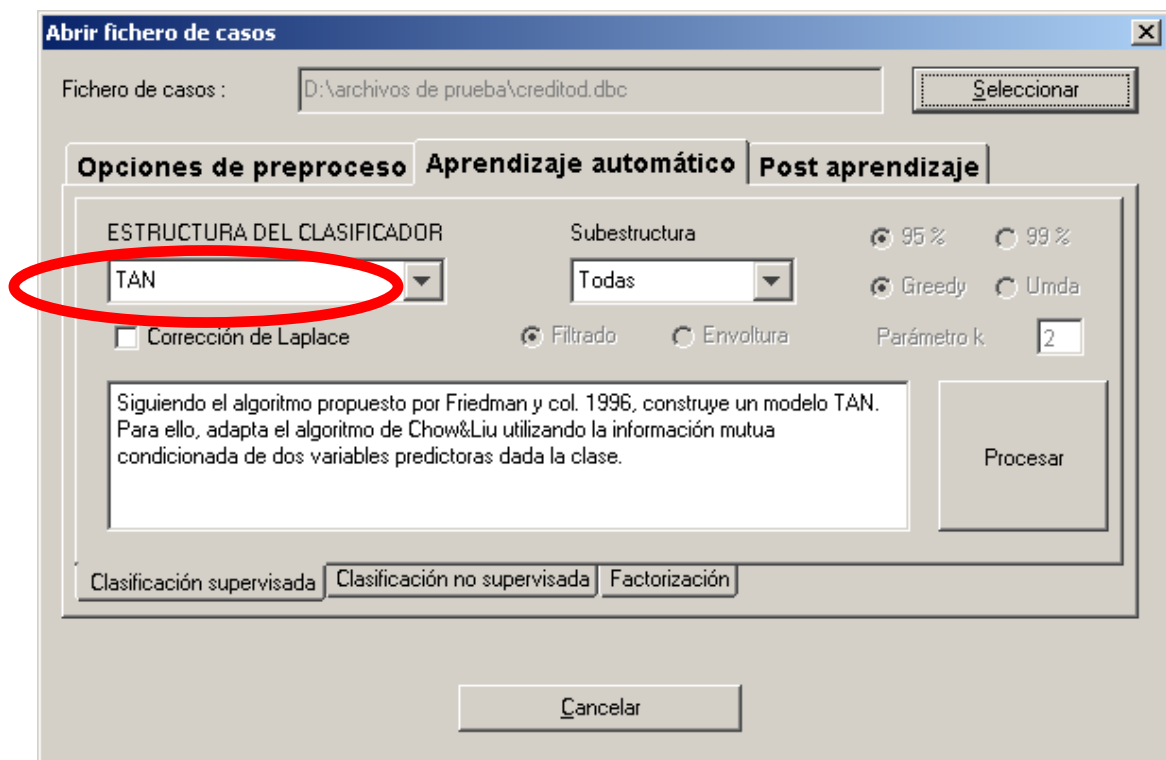


Figura 17: Menú apertura de fichero, opciones de aprendizaje automático

A continuación, en la figura 18, se detalla la red obtenida en modo Edición:

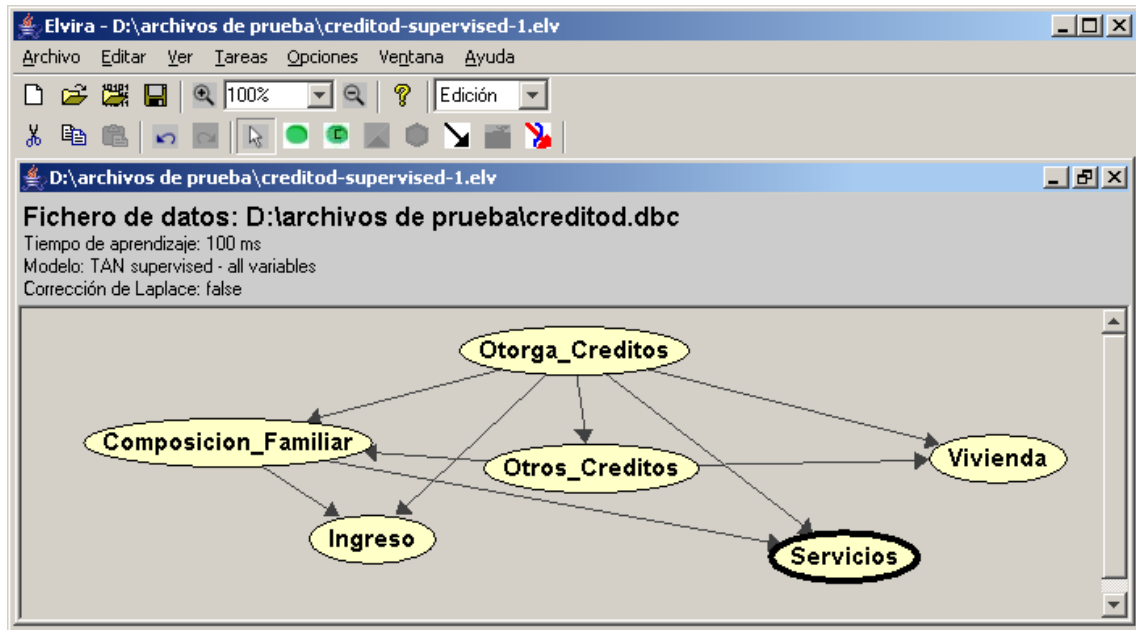
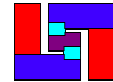


Figura 18: Red Bayesiana en modo edición

Por último, en la figura 19, se indican los valores de probabilidad obtenidos desde el modo Inferencia:

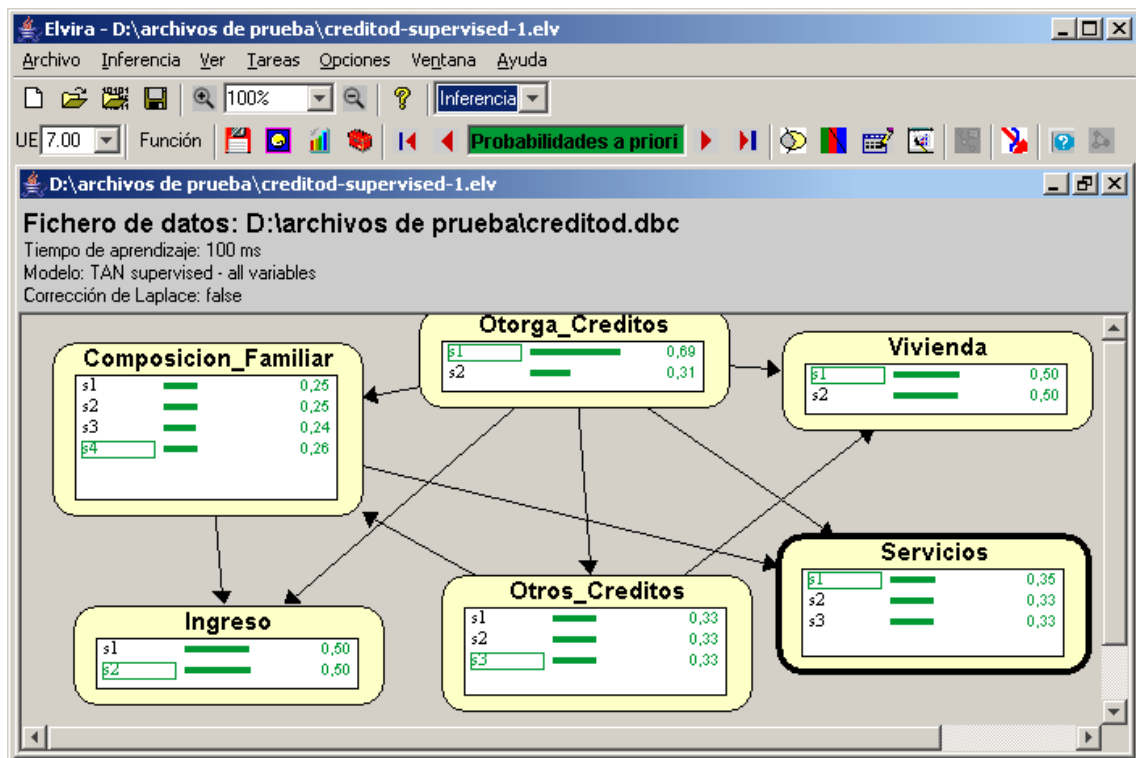
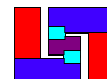


Figura 19: Red Bayesiana en modo inferencia

A continuación se detallan los valores de probabilidad asociados a cada nodo en particular



Valor	Probabilidad
S1	0,6944
S2	0,3056

Tabla 12: Nodo: Otorga_Créditos

Nodo Padre	Valores							
Composición_Familiar	S1	S1	S2	S2	S3	S3	S4	S4
Otorga_Crédito	S1	S2	S1	S2	S1	S2	S1	S2
Valor	Probabilidad							
S1	0.6	0.2727	0.5	0.5	0.4444	0.625	0.2778	0.7368
S2	0.4	0.7273	0.5	0.5	0.5556	0.375	0.7222	0.2632

Tabla 13: Nodo: Ingreso

Nodo Padre	Valores	
Otorga crédito	S1	S2
Valor	Probabilidad	
S1	0.45	0.0682
S2	0.41	0.1591
S3	0.14	0.7727

Tabla 14: Nodo: Otros_Créditos

Nodo Padre	Valores					
Otros_Creditos	S1	S1	S2	S2	S3	S3
Otorga_Crédito	S1	S2	S1	S2	S1	S2
Valor	Probabilidad					
S1	0.2667	0	0.2195	0.4286	0.2857	0.2353
S2	0.2667	0	0.2927	0	0.4286	0.1765
S3	0.2667	0	0.2927	0	0.2143	0.2353
S4	0.1999	1	0.1951	0.5714	0.0714	0.3529

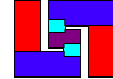
Tabla 15: Nodo: Composición_familiar

Nodo Padre	Valores					
Otorga_Crédito	S1	S1	S2	S2	S3	S3
Otros_Creditos	S1	S2	S1	S2	S1	S2
Valor	Probabilidad					
S1	0.4667	1	0.439	0.8571	0	0.7059
S2	0.5333	0	0.561	0.1429	1	0.2941

Tabla 16: Nodo: Vivienda

Nodo Padre	Valores							
Composición_Familiar	S1	S1	S2	S2	S3	S3	S4	S4
Otorga_Crédito	S1	S2	S1	S2	S1	S2	S1	S2
Valor	Probabilidad							
S1	0.4	0.2727	0.3333	0.3333	0.3333	0.5	0.3889	0.2632
S2	0.28	0.3636	0.3333	0.3333	0.3704	0.25	0.3333	0.3158
S3	0.32	0.3637	0.3334	0.3334	0.2963	0.25	0.2778	0.421

Tabla 17: Nodo: Servicios:



2.1.2.3. Caso 3 obtención de una red mediante el clasificador KDB

Para generar una red mediante el Clasificador KDB se deberá cambiar la opción “Estructura del Clasificador”, que por defecto sugiera Naïve Bayes, a KDB. Como se indica a continuación, en la figura 20:

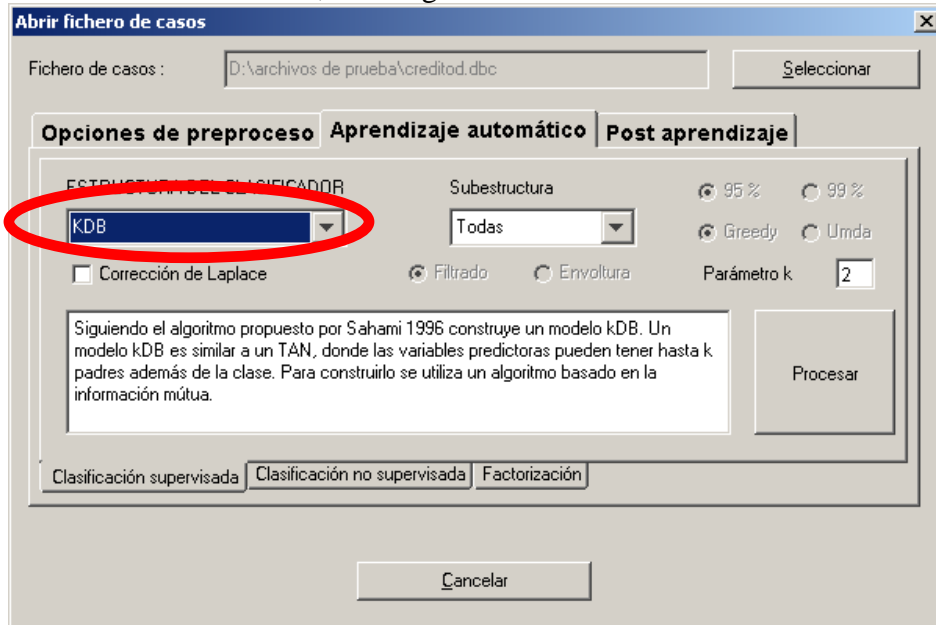


Figura 20: Menú apertura de fichero, opciones de aprendizaje automático

A continuación, en la figura 21, se detalla la red obtenida en modo Edición:

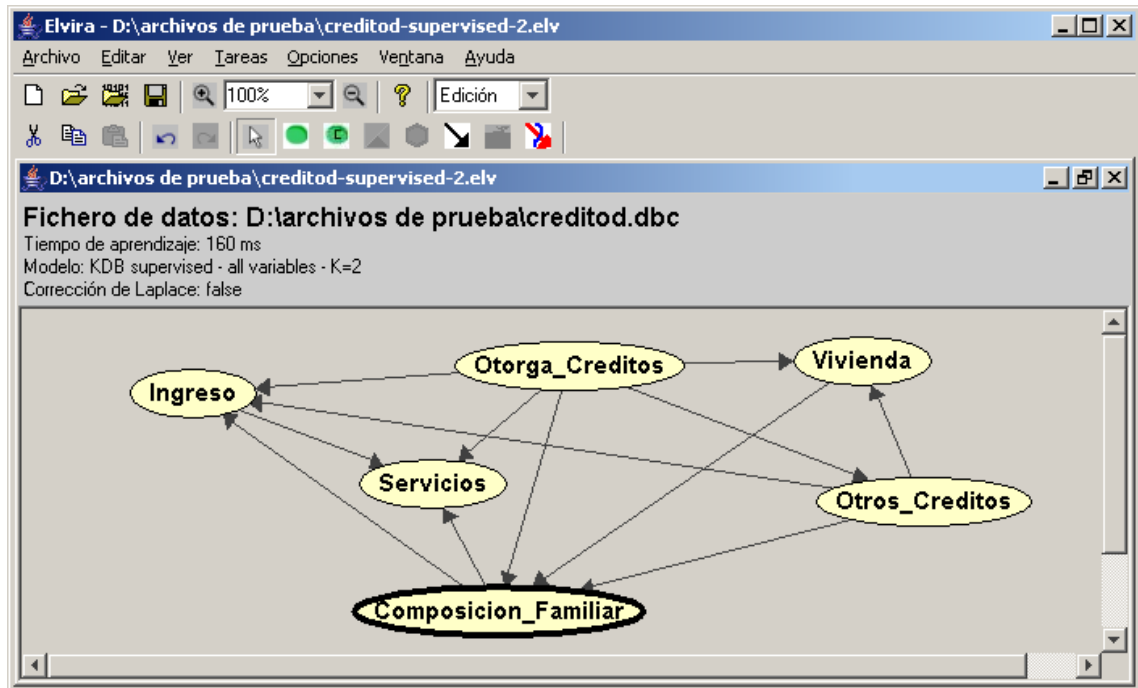


Figura 21: Red Bayesiana en modo edición

Por último, en la figura 22, se indican los valores de probabilidad obtenidos desde el modo Inferencia:

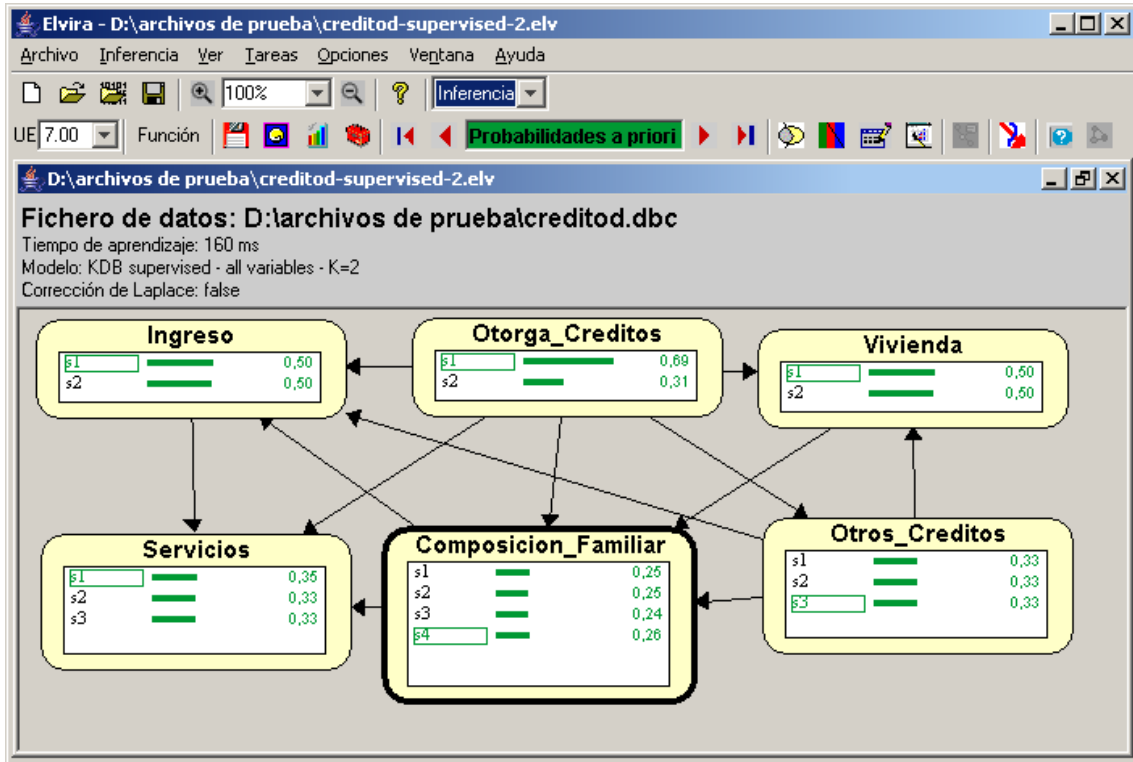
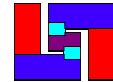


Figura 22: Red Bayesiana en modo inferencia

A continuación se detallan los valores de probabilidad asociados a cada nodo en particular

Valor	Probabilidad
S1	0,6944
S2	0,3056

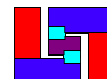
Tabla 18: Nodo: Otorga_Créditos

Nodo Padre	Valores											
Otorga_Crédito	S1	S1	S1	S1	S1	S1	S2	S2	S2	S2	S2	S2
Otros_Créditos	S1	S1	S2	S2	S3	S3	S1	S1	S2	S2	S3	S3
Vivienda	S1	S2	S1	S2	S1	S2	S1	S2	S1	S2	S1	S2
Valor	Probabilidad											
S1	0.2857	0.25	0.1667	0.2609	0	0.2857	0	0	0.5	0	0.25	0.2
S2	0.2857	0.25	0.3333	0.2609	0	0.4286	0	0	0	0	0.25	0
S3	0.2857	0.25	0.3333	0.2609	0	0.2143	0	0	0	0	0.25	0.2
S4	0.1429	0.25	0.1667	0.2173	1	0.0714	1	1	0.5	0	0.25	0.6

Tabla 19: Nodo: Composición_familiar

Nodo Padre	Valores	
Otorga crédito	S1	S2
Valor	Probabilidad	
S1	0.45	0.0682
S2	0.41	0.1591
S3	0.14	0.7727

Tabla 20: Nodo: Otros_Créditos



Nodo Padre	Valores																							
Otorga_Crédito	S1	S1	S1	S1	S1	S1	S1	S1	S1	S1	S1	S1	S2	S2	S2	S2	S2	S2	S2	S2	S2	S2	S2	S2
Composición_Familiar	S1	S1	S1	S2	S2	S2	S3	S3	S3	S4	S4	S4	S1	S1	S1	S2	S2	S2	S3	S3	S3	S4	S4	S4
Otros_Creditos	S1	S2	S3	S1	S2	S3	S1	S2	S3	S1	S2	S3	S1	S2	S3	S1	S2	S3	S1	S2	S3	S1	S2	S3
Valor	Probabilidad																							
S1	0.5	0.6667	0.5	0.5	0.5	0.5	0.5	0.5	0	0.3333	0.25	0	0	0	0.375	0	0	0.5	0	0	0.625	1	1	0.5833
S2	0.5	0.3333	0.5	0.5	0.5	0.5	0.5	0.5	1	0.6667	0.75	1	1	1	0.625	1	1	0.5	1	1	0.375	0	0	0.4167

Tabla 21: Nodo: Ingreso

Nodo Padre	Valores					
Otros_Creditos	S1	S1	S1	S2	S2	S2
Otorga_Crédito	S1	S2	S3	S1	S2	S3
Valor	Probabilidad					
S1	0.4667	0.439	0	1	0.8571	0.7059
S2	0.5333	0.561	1	0	0.1429	0.2941

Tabla 22: Nodo: Vivienda

Nodo Padre	Valores																	
Otorga_Crédito	S1	S1	S1	S1	S1	S1	S1	S1	S1	S2	S2	S2	S2	S2	S2	S2	S2	S2
Composición_Familiar	S1	S1	S2	S2	S3	S3	S4	S4	S1	S1	S2	S2	S3	S3	S4	S4	S4	
Ingreso	S1	S2	S1	S2	S1	S2	S1	S2	S1	S2	S1	S2	S1	S2	S1	S2	S1	S2
Valor	Probabilidad																	
S1	0.4	0.4	0.333	0.333	0.333	0.333	0.4	0.333	0.333	0.2	0.333	0.333	0.333	0.333	0.285	0.285	0.285	0.285
Sz2	0.2667	0.333	0.333	0.333	0.4167	0.333	0.333	0.333	0.333	0.333	0.333	0.333	0.333	0.333	0.333	0.333	0.333	0.333
S3	0.333	0.333	0.334	0.334	0.25	0.334	0.334	0.334	0.334	0.334	0.334	0.334	0.334	0.334	0.428	0.428	0.428	0.428

Tabla 23: Nodo: Servicios:

2.1.3. Instanciación de nodos

Para poder ingresar evidencia a la red el software Elvira debe estar en modo de trabajo “Inferencia” (como se muestra en la figura 23).

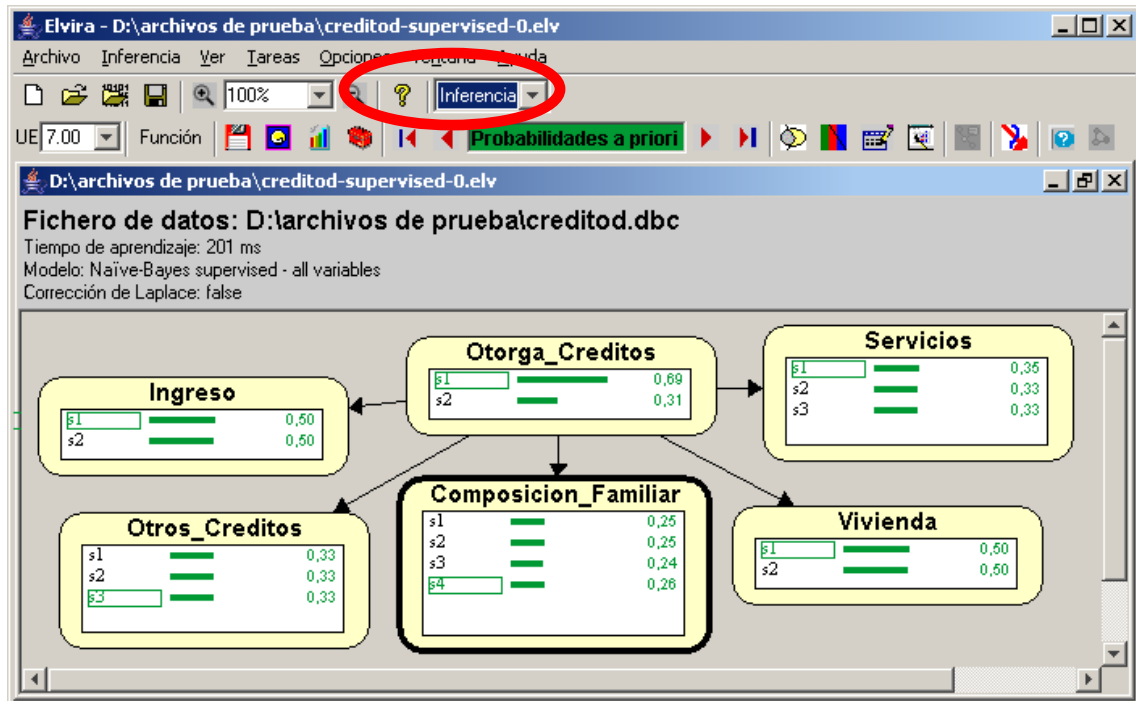
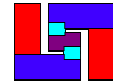


Figura 23: Red Bayesiana en modo inferencia

Desde esta pantalla y con solo hacer doble click sobre el valor del nodo que deseamos instanciar, el programa inmediatamente recalculara los valores de probabilidad de toda la red. A continuación en la figura 24 se muestra la respuesta del software ante la instanciación del valor “S1” del nodo “Otorga_Creditos”:

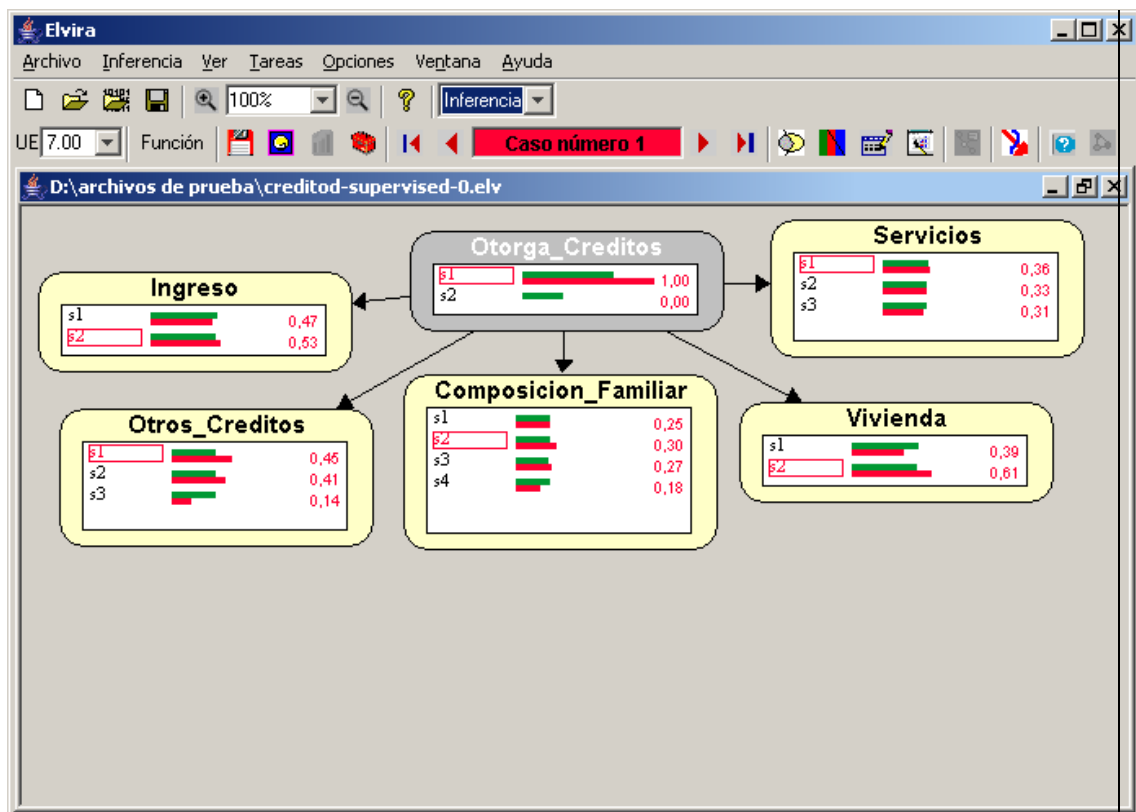
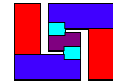


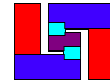
Figura 24: Red Bayesiana en modo inferencia



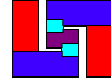
La figura 24 muestra la respuesta del sistema ante una instanciación, mediante el color verde se indican los valores de probabilidad a priori (sin evidencia) y con color rojo se detallan los valores de probabilidad fruto del ingreso de la evidencia.

3. Bibliografía

- [Beinlich, I. et al, 1989] Beinlich, I.; Suermondt, H.; Chavez, R.; Cooper, G.; 1989; *The Alarm Monitoring System: A case study with two probabilistic inference techniques for belief networks*. In proceedings of the 2° European Conference on Artificial Intelligence in Medicine.
- [Blurock, E., 1996] Blurock, E.; 1996; *The ID3 Algorithm*, Research Institute Institute for Symbolic Computation; www.risc.unilinz.ac.at/people/blurock/analysis/manual/document ; Australia.
- [Breese & Blake, 1995] Breese, J.; Blake, R.; 1995; *Automating computer bottleneck detection with belief nets*. Proceeding of the conference on Uncertainty in artificial Intelligence. Morgan Kaufmann, San Francisco, CA.
- [Chen, M. et al, 1996] Chen, M.; Han, j.; Yu, P.; 1996; *Data Mining: An overview from Database perspective*. IEEE Transaction on Knowledge and Data Eng.
- [Diaz, F. et al, 1999] Diaz, F.; Corchado, J.; 1999; *Rough sets bases learning for Bayesian networks*. International workshop on objective bayesian methodology; Valencia, Spain.
- [Evangelos, S. et al, 1996] Evangelos, S.; Han J.; 1996; *Proceeding of the Second International Conference on Knowledge Discovery and Data Mining*. Portland, EE.UU.
- [Felgaer, P. et al, 2003] Felgaer, P.; Britos, P.; Sicre, J.; Servetto, A.; García-Martínez, R. y Perichinsky, G.; 2003; *Optimización de Redes Bayesianas Basada en Técnicas de Aprendizaje por Instrucción*.; Proceedings del VIII Congreso Argentino de Ciencias de la Computación. Pág. 1687.
- [Fritz, W. et al, 1989] Fritz, W.; García Martínez, R.; Blanque, J.; Rama, A.; Adobbati, R.; Samo, M.; 1989; *The Autonomous Intelligence System. Robotics and Autonomous System*. Vol. 5 Nber. 2. Elsevier.

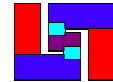


- [Gallion, R. et al, 1993] Gallion, R.; St Clair, D.; Sabharwal, C.; Bond, W.; 1993; *Dynamic ID3: A Symbolic Learning Algorithm for Many-Valued Attribute Domains*. Engineering Education Center, University of Missouri-Rolla, St. Luis, EE.UU.
- [García Martínez, R., 1993] García Martínez, R.; 1993; *Heuristic Theory Formation as a machine learning method*. Proceeding VI International Symposium on Artificial Inteligence. Páginas 294 a 298. Editorial Limusa. Mexico.
- [García Martínez, R., 1995] García Martínez, R.; 1995; *Theory Formation by Heuristics*. Proceeding of the II International Congress on Information Engineering. PP 200-205. School of Engineering Pess. University of Buenos Aires.
- [García Martínez, R., 1997] García Martínez, R.; 1997; *Sistemas Autónomos: Aprendizaje Automático*; Nueva Librería, Buenos Aires, Argentina.
- [García Martínez, R. et al, 2003] García Martínez R.; Pasqín D.;2003; *Sistemas Inteligentes*; Capítulo 1: “Aprendizaje Automático”, Capítulo 2: “Redes Neuronales Artificiales”; Nueva Librería, Buenos Aires, Argentina.
- [García Martínez & Borrajo, 2000] García Martínez, R.; Borrajo D.; 2000; *An integrated approach of learning, planning and executing*; Journal of Intelligent and Robotic Systems. Volumen 29, Número 1, páginas 47 a 78. Kluwer Academic Press.
- [Heckerman, D., 1995] Heckerman, D.; 1995; *A Tutorial on learning bayesian network*. Technical report MSR-TR-95-06, Microsoft research , Redmond, WA.
- [Heckerman D. et al, 1996] Heckerman, D.; Chickering, M.; 1996; *Efficient approximation for the marginal likelihood on incomplete data given a bayesian network*. Technical report MSR-TR-96-8, Microsoft Research, , Microsoft Corporation.
- [Hernández O.J., 2000] Hernández Orallo, J.; 2000; *Extracción Automática de Conocimiento de base de datos e ingeniería del software*. Programación declarativa e ingeniería de la programación.

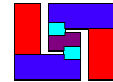


[Hernández O.J. et al, 2004]

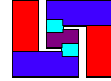
Hernández Orallo, J.; Ferri Ramírez, C.; Ramírez Quintana J.;2004;*Introducción a la minería de datos*; Capítulo 10: “Métodos Bayesianos”; PEARSON EDUCACION.



- [Holsheimer & Siebes, 1994] Holsheimer, M. & Siebes A.; 1994 ;*Data Mining: the search for knowledge in databases*. Computer Science/Departament of Algorithmics and Architecture, Centrum voor Wiskunde en inform tica , CS-R9406;  msterdam, Holanda.
- [Larra naga, P. et al, 2004] Larra naga, P. e Inza, I.; *Clasificadores Bayesianos*; Departamento de Ciencias de la Computaci n e Inteligencia Artificial; Universidad del Pa s Vasco-Euskal Herreiko Unibertsitatea; 2004; <http://www.sc.ehu.es/ccwbayes/docencia/mmcc/docs/t6bayesianos.pdf>
- [Mannilla, H., 1997] Mannilla, H.; 1997; *Methods and problems in data mining*. Inc. Proc. of International Conference on Database Theory, Delphi, Greece.
- [Michalski, R. et al, 1982] Michalski R.; Baskin, A.; Spackman, K.; 1982; *A Logic-Based Approach to Conceptual Database Analysis*, Sixth Annual Symposium on Computer Applications on Medical Care, George Washington University, Medical Center, Washington, DC, EE.UU.
- [Michalski, R. et al, 1983] Michalski R.; 1983; *A Theory and Methodology of Inductive Learning*; Morgan-Kauffman; EE.UU.
- [Michalski, R. et al, 1998] Michalski R.; Bratko, I.; Kubat, M.; 1998; *Machine Learning and data mining . Methods and Applications*; Wiley & Sons Ltd.; EE.UU:
- [Ochoa, M. et al, 2004] Ochoa, M.; Garc a Mart nez, R.; Britos P.; 2004; *Trabajo final de Especialidad en Ingenier a de Sistemas Expertos Herramientas Inteligentes para Explotaci n de Informaci n*; Instituto Tecnol gico de Buenos Aires.
- [Pearl, J., 1988] Pearl, J.; 1988; *Probabilistic reasoning in intelligent systems*. Morgan Kaufmann, San Mateo, CA.
- [Perichinsky, G. et al, 2000] Perichinsky,G.; Garc a-Mart nez, R.; Proto, A.; 2000; *Knowledge Discovery Based on Computational Taxonomy And Intelligent Data Mining*; CD del VI Congreso Argentino de Ciencias de la Computaci n. ([\cacic2k\cacic\sp\039\IS-039.htm](http://cacic2k.cacic.sp/039/IS-039.htm)). Usuhaia. Octubre 2 al 6.



- [Perichinsky, G. et al, 2001] Perichinsky, G.; Garcia Martinez, R.; Proto, A.; Sevetto, A.; Grossi, D.; 2001; *Integrated Environment of Systems Automated Engineering*; Proceedings del II Workshop de Investigadores en Ciencias de la Computación. Mayo. Editado por Universidad Nacional de San Luis en el CD Wicc2001:\Wiccflash\Areas\IngSoft\Integrated_Environment.pdf .
- [Perichinsky, G. et al, 2003] Perichinsky, G.; Servente, M.; Servetto, A.; García-Martínez, R.; Orellana, R.; Plastino, A.; 2003; *Taxonomic Evidence and Robustness of the Classification Applying Intelligent Data Mining*; Proceedings del VIII Congreso Argentino de Ciencias de la Computación. Pág. 1797-1808.
- [Perichinsky, & García M., 2000] Perichinsky, G. y Garcia Martinez, R.; 2000; *A Data Mining Approach to Computational Taxonomy*; Proceedings del Workshop de Investigadores en Ciencias de la Computación. Páginas 107-110. Editado por Departamento de Publicaciones de la Facultad de Informática. Universidad Nacional de La Plata. Mayo.
- [Piatetski-Shapiro, G. et al, 1991] Piatetski-Shapiro, G.; Frawley, W.; Matheus C.; 1991; *Knowledge discovery in databases: an overview*; AAAI-MIP Press; Menlo Park; California.
- [Piatetski-Shapiro, G. et al, 1996] Piatetski-Shapiro, G.; Fayyad, U.; Smith P.; 1996; *From data mining to Knowledge discovery*; AAAI Press/MIT Press; CA.
- [Quinlan, J, 1993a] Quinlan J.; 1993; *The effect of Noise on Concept Learning*, en R. Michalski, J. Carbonell & T. Mitchells (Eds.) Machine Learning, The Artificial Intelligence Approach. Morgan Kaufmann, Vol. I, Capítulo 6, páginas 149 a 167. San Mateo, CA: Morgan Kaufmann, EE.UU.
- [Quinlan, J.,1993b] Quinlan J.; 1993; *Learning Efficient Clasifications Procedures and Their Application to Chess Games*, en R. Michalski, J. Carbonell & T. Mitchells (Eds.) Machine Learning, The Artificial Intelligence Approach. Morgan Kaufmann, Vol. II, Capítulo 15, páginas 463 a 482. EE.UU.



[Servente, M. et al, 2002]

Servente M. ; García Martínez R.;2002; *Tesis Doctoral Algoritmos TDIDT aplicados a la minería de datos inteligente*; Universidad de Buenos Aires.

[Ramoni & Sebastiani et al, 1996]

Ramoni, M; Sebastián P.; 1996; *Learning Bayesian networks from incomplete databases*. Technical report KMI-TR-43, knowledge Media Institute, The open University.