

PROTOTIPO DE SOFTWARE PARA LA CLASIFICACIÓN DE DATOS MEDIANTE EL MÉTODO BAYESIANO TAN – “udTAN”

*Leslie del Carmen González Díaz

**Jorge Enrique Rodríguez Rodríguez

Resumen

En este artículo se plasma la implementación del método TAN (del inglés *Tree Augmented Naive Bayes*), aplicado a la clasificación de datos. TAN constituye una extensión del clasificador Naïve Bayes (NB), cuya idea es construir una red bayesiana un poco más compleja que el NB, pero donde se da un tratamiento especial a la variable clase; por tanto, se enmarca en la filosofía de aprender redes bayesianas orientadas a clasificación. Con TAN se pretende mantener la simplicidad computacional del clasificador NB pero intentando mejorar la efectividad de la clasificación. Para ello, en lugar de suponer todas las variables independientes, dada la clase, se admiten ciertas dependencias entre los atributos. En concreto, se supone que los atributos constituyen una red bayesiana con forma de árbol. Este algoritmo se basa en el concepto de *información mutua*.

Palabras clave: Clasificar, dependencia, probabilidad, información mutua y método bayesiano..

* Tecnóloga en Sistematización de Datos. Estudiante de Ingeniería en Redes de Computadores de la Universidad Distrital Francisco José de Caldas. les.yeos@gmail.com

** Ingeniero de Sistemas. Especialista en Telemática. Especialista en Ingeniería de Software. Magíster en Ingeniería de Sistemas. Docente Universidad Distrital Francisco José de Caldas. jrodri@udistrital.edu.co

Abstract

This paper exposes the implementation of the method TAN (Tree Augmented Naive Bayes), applied to the data classification. TAN constitutes an extension of the classifier Naïve Bayes (NB). The idea consists on building a bayes network a little more robust than the NB, but where a special treatment is given to the class; therefore, it is oriented in the philosophy of learning bayes networks guided to classification. TAN tries to support the simplicity of compute of the classifier N.B., trying to improve the effectiveness of the classification. For it, instead of supposing all the independent variables given the class, certain dependences between the attributes are admitted. In particular, one assumes that the attributes constitute a bayesiana network with tree form. This algorithm is based on the concept of mutual information.

Keywords: Clasification, dependency, probability, mutual information and bayes method.

Introducción

En el año 1763, dos años después de la muerte de *Thomas Bayes* (1702-1761), se publicó una memoria en la que aparece, por vez primera, la determinación de la probabilidad de las causas a partir de los efectos que han podido ser observados. El cálculo de dichas probabilidades recibe el nombre de teorema de Bayes (Seda, 2005).

En teoría de la probabilidad, el teorema de Bayes es la regla básica para realizar inferencias (Hernández, 2004). Dicho teorema puede verse cómo la búsqueda de la hipótesis más probable dado un

conjunto de datos de entrenamiento y un conocimiento a priori de la probabilidad de cada hipótesis. Conociendo la probabilidad a priori de que ocurra un suceso, el método bayesiano permite modificar su valor cuando se dispone de nueva información, esta es la probabilidad a posteriori (Molinero, 2002).

Las redes bayesianas permiten tratar modelos complejos gracias a la explotación de las independencias presentes en el modelo. De estas consideraciones surge la idea de usar redes bayesianas como clasificadores (Friedman, 1997; Larrañaga, 1998).

Las Redes Bayesianas (RBs) son un formalismo que en los últimos años ha demostrado su potencialidad como modelo de representación del conocimiento con incertidumbre. Este formalismo nació como un aporte de diferentes campos de investigación: teoría de toma de decisiones, estadística e inteligencia artificial. El éxito de numerosas aplicaciones en campos variados como la medicina, la recuperación de información, la visión artificial, la fusión de información, la agricultura, etc., avalan este formalismo (Hernández, 2004).

Los autores orientan el artículo hacia la implementación de este método para realizar una de las tareas más comunes de Minería de Datos, la clasificación. La clasificación de datos es un proceso de dos pasos (figura 1). En el primer paso, se construye un modelo, el cual describa el conjunto preliminar de clases. El modelo es construido analizando los registros ejemplos. Cada registro pertenece a una clase específica conocida, debido a esto, esta técnica de clasificación se conoce como aprendizaje supervisado. En contraste con el aprendizaje no supervisado (también conocido como clustering), en el cual la clase a la que pertenece cada registro es desconocida, y el número de clases por aprender

tampoco puede ser conocido. Generalmente, el modelo aprendido es representado en la forma de reglas de clasificación, árboles de decisión, o fórmulas matemáticas (Han, 2001).

En el segundo paso (Figura 2), el modelo es usado para generar la clasificación de datos desconocidos. Luego se estima la precisión del modelo o clasificador. La precisión de un modelo en un conjunto dado de datos es el porcentaje de ejemplos, del conjunto de entrenamiento, que fueron correctamente clasificados. Si la precisión del modelo es considerada aceptable, el modelo puede ser usado para clasificar futuros conjuntos de datos para los cuales la etiqueta de clase es desconocida.

Figura 1. Fase de aprendizaje

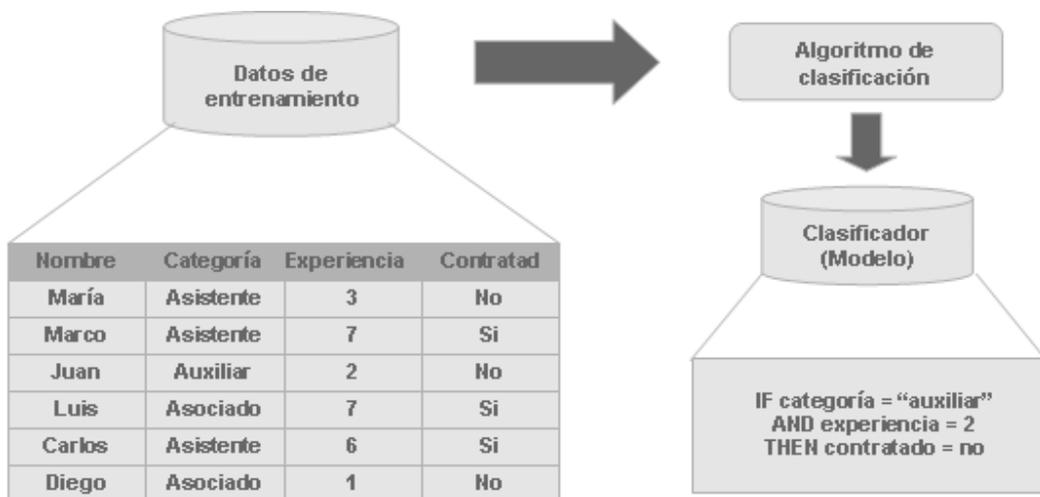
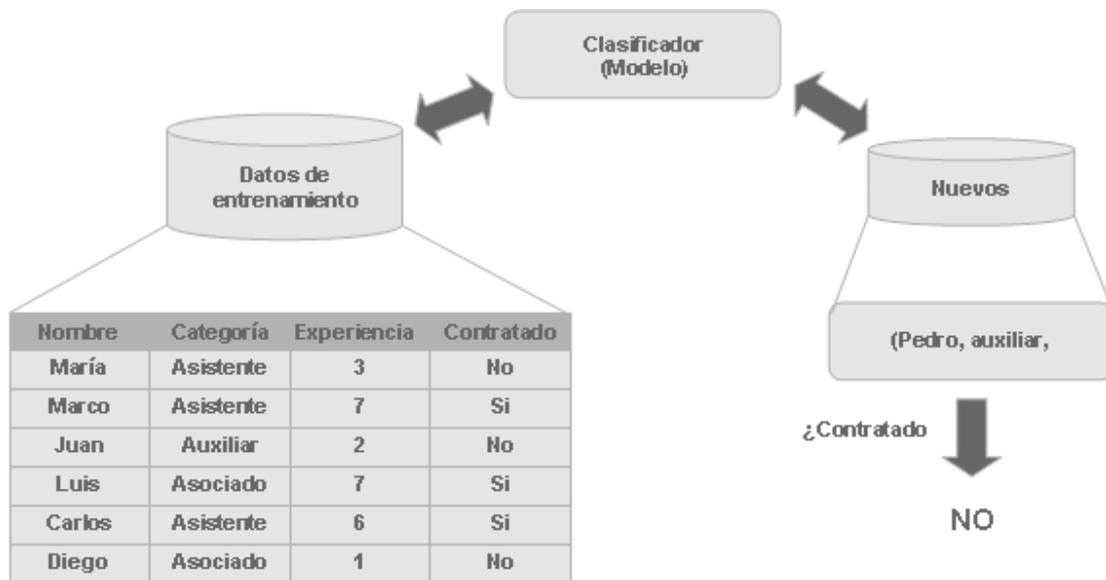


Figura 2. Clasificación / Predicción



Este artículo se encuentra estructurado en: - una descripción del método TAN, - la importancia de este método en la clasificación de datos, - especificación del prototipo de software implementado, - análisis de pruebas y resultados, - conclusiones, y – trabajos futuros.

1. DESCRIPCIÓN DEL MÉTODO TAN

Según explica (Morales, 2005), obtener una red bayesiana a partir de datos es un proceso de aprendizaje, el cual se divide, naturalmente, en dos aspectos:

- a) **Aprendizaje paramétrico:** dada una estructura, obtener las probabilidades a priori y condicionales requeridas.

- b) **Aprendizaje estructural:** obtener la estructura de la red Bayesiana, es decir, las relaciones de dependencia e independencia entre las variables involucradas.

Las técnicas de aprendizaje estructural dependen del tipo del modelo de red: árboles, poliárboles y redes multiconectadas. Otra alternativa, es combinar conocimiento subjetivo del experto con aprendizaje. Para ello se parte de la estructura dada por el experto, la cual se valida y mejora utilizando datos estadísticos.

Aprendizaje Paramétrico

El aprendizaje paramétrico consiste en encontrar los parámetros asociados a una estructura dada de una red bayesiana. Dichos parámetros consisten en las probabilidades *a priori* de los nodos raíz y las probabilidades condicionales de las demás variables, dados sus padres.

Si se conocen todas las variables, es fácil obtener las probabilidades requeridas. Las probabilidades previas corresponden a las marginales de los nodos raíz, y las condicionales se obtienen de las conjuntas de cada nodo con su(s) padre(s).

Aprendizaje Estructural de Árboles

El método de aprendizaje estructural de árboles se basa en el algoritmo desarrollado por Chow y Liu (Chow-Liu, 1968), para aproximar una distribución de probabilidad por un producto de probabilidades de segundo orden, lo que corresponde a un árbol.

Algoritmo TAN

Friedman y col. (Friedman, 1997) presentan un algoritmo denominado Tree Augmented Network (TAN) el cual consiste básicamente en una adaptación del algoritmo de Chow-Liu (Chow-Liu, 1968). Ver figura 3.

Como explica (Larragaña, 2005), 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:

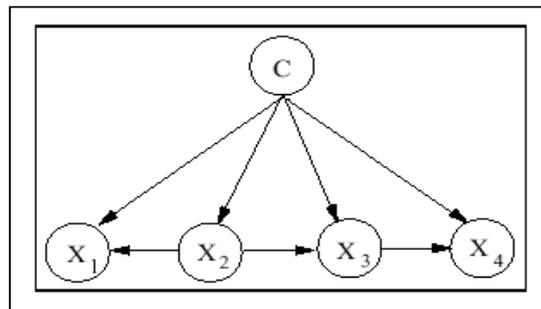
Ecuación 1. Información Mutua

$$I(X, Y | C) = \sum_{i=1}^n \sum_{j=1}^m \sum_{r=1}^w p(\mathbf{x}_i, \mathbf{y}_j, \mathbf{c}_r) \log \left(\frac{p(\mathbf{x}_i, \mathbf{y}_j | \mathbf{c}_r)}{p(\mathbf{x}_i | \mathbf{c}_r) p(\mathbf{y}_j | \mathbf{c}_r)} \right)$$

Como se ve en el pseudocódigo de la Figura 4, 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 a 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.

Figura 3. Ejemplo de estructura Naïve Bayes aumentada a árbol - TAN (Larragaña, 2005)



Luego, se utiliza el algoritmo de Kruskal, que 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. 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 fichero de casos. En la Figura 5 se muestra un ejemplo de aplicación del algoritmo.

(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, manteniendo 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.

Tras aprender la estructura de árbol entre los atributos, el algoritmo TAN añade la variable clase y la hace padre de todos ellos. Este algoritmo tiene dos claras ventajas: un bajo coste computacional $O(n^2 N)$, siendo N el número de instancias en el conjunto de entrenamiento y n el número de variables y, en segundo lugar, asegura que la estructura de red obtenida es la de máxima verosimilitud del conjunto de todas las posibles estructuras TAN (Hernández, 2004).

Figura 4. Algoritmo TAN (Friedman, 1997)

Algoritmo

Paso 1. Calcular $I(X_i, X_j | C)$ con $i < j$, $i, j = 1, \dots, n$.

Paso 2. Construir un grafo no dirigido completo cuyos nodos corresponden 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 $I(X_i, X_j | C)$

Paso 3. A partir del grafo completo anterior y siguiendo el algoritmos de Kruskall, construir un árbol expandido de máximo peso

Paso 4. Transformar el árbol no dirigido resultante en uno dirigido, escogiendo una variable como raíz, para a continuación direccionar el resto de aristas

Paso 5. Construir un modelo TAN añadiendo un nodo etiquetado como C y posteriormente un arco desde C a cada variable predictora X_i

2. IMPORTANCIA DEL MÉTODO TAN EN LA CLASIFICACIÓN DE DATOS

2.1 ¿Por qué Bayes?

La teoría de la probabilidad y los métodos bayesianos son una de las técnicas que más se han utilizado en problemas de inteligencia artificial y por tanto, de aprendizaje automático y minería de datos. Una

de las razones por las que los métodos bayesianos son relevantes al aprendizaje automático y a la minería de datos es que son un método práctico y simple para realizar inferencias a partir de los datos, induciendo modelos probabilísticos que después serán usados para razonar (formular hipótesis) sobre nuevos valores observados. Además, permiten calcular de forma explícita la probabilidad asociada a cada una de las hipótesis posibles, lo que constituye una gran ventaja sobre otras técnicas (cit. Mitchell, 1997).

Adicionalmente, las Redes Bayesianas (RBs) (cit. Pearl, 1988), permiten simplificar el coste computacional del modelo probabilístico, sin pérdida de expresividad por parte del mismo.

¿Por qué el Método TAN?

Ante el buen rendimiento ofrecido por el clasificador Naïve Bayes (NB), a pesar de la fuerte suposición que realiza, respecto a la independencia de los atributos dada la clase, cabe preguntarse si los resultados no serán mejores tras suprimir dicha suposición (Hernández, 2004).

Según (Morales, 2005), las redes bayesianas son una alternativa para minería de datos, la cual tiene varias ventajas:

- Permiten aprender fácilmente sobre relaciones de dependencia y causalidad.
- Permiten combinar conocimiento con datos.
- Evitan el sobre-ajuste de los datos.
- Pueden manejar bases de datos incompletas.

Partiendo de que el clasificador NB constituye la red bayesiana más sencilla que se puede construir orientada a clasificación (Hernández, 2004), TAN constituye una extensión del clasificador NB, con la

cual se pretende mantener la simplicidad computacional del clasificador NB pero intentando mejorar la tasa de acierto durante la clasificación (Friedman, 1997).

3. ESPECIFICACIÓN DEL PROTOTIPO DE SOFTWARE IMPLEMENTADO

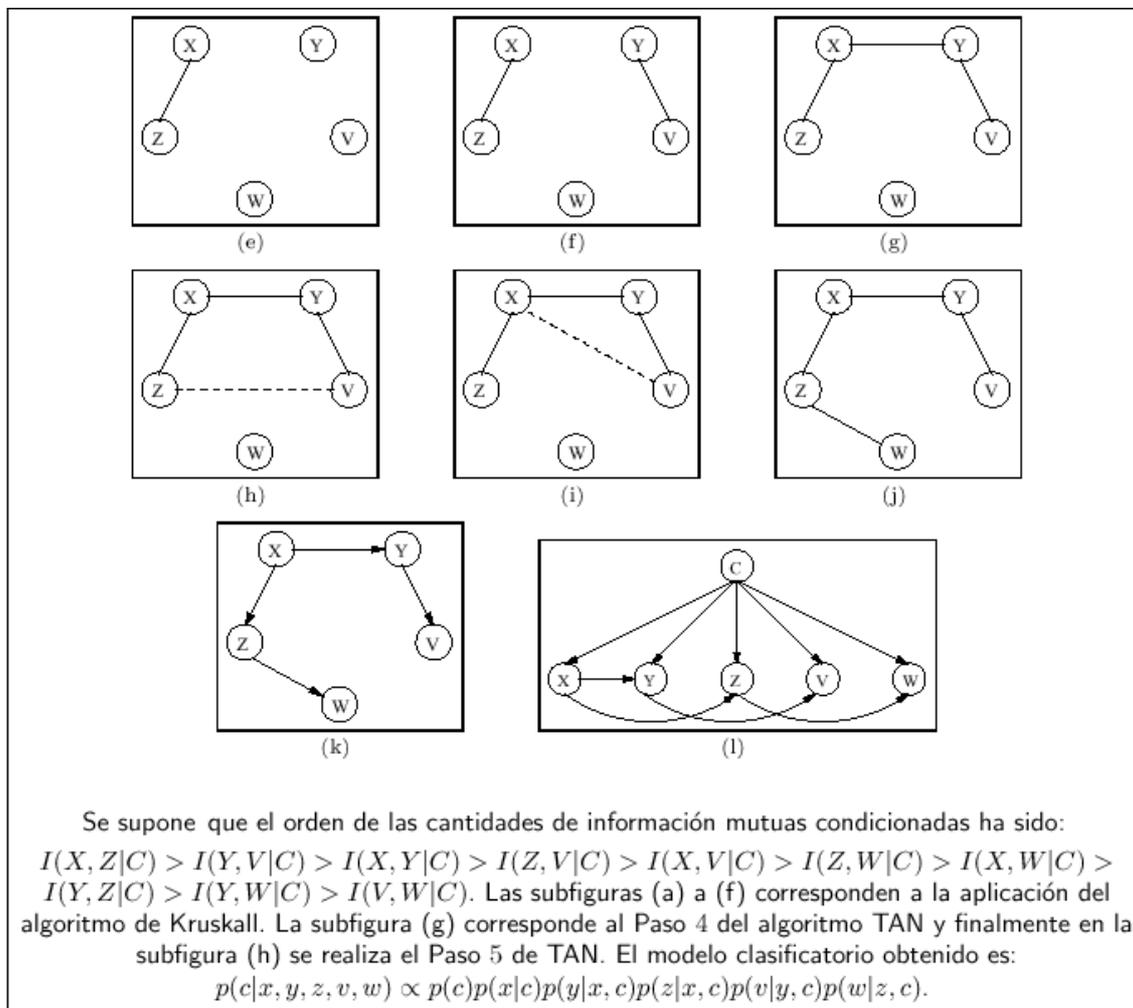
Con el fin de implementar el algoritmo TAN y probar sus resultados, se desarrolló un prototipo de software (en adelante llamado UDTan) en lenguaje Java, cuyo propósito es clasificar un conjunto de instancias, a partir del valor de los atributos que las componen.

UDTan inicia leyendo un archivo de texto, especificado por el usuario, compuesto de N instancias y n variables predictoras, separadas por comas, para cargar los datos del mismo en una matriz.

Después, el programa examina los datos contenidos en cada columna de la matriz, para determinar los distintos valores que puede tomar cada atributo, teniendo en cuenta que la última columna contiene los valores que puede tomar la variable clase. Luego, a partir del producto de las probabilidades de dichos valores halla la cantidad de información mutua entre cada par de atributos.

Figura 5. Ilustración del algoritmo TAN con cinco variables predictoras X, Y, Z, V y W

(Larragaña, 2005)



Una vez obtenidas las cantidades de información mutua de todos los pares de atributos, UDTan las ordena de mayor a menor para ir estableciendo enlaces (aristas) entre dichos pares de atributos (nodos), comenzando por el de mayor peso, sin que se formen ciclos y restringiendo los enlaces convergentes hacia un mismo nodo, construyendo así una red bayesiana en forma de árbol.

Construida la estructura TAN, se utiliza para definir el modelo de clasificación, es decir, la distribución de probabilidad, que representa la posibilidad de que coincidan todos los atributos, dada la clase.

Finalmente, para cada instancia (fila de la matriz), se aplican los cálculos indicados en el modelo obtenido y se clasifica dentro de la clase cuyo valor de probabilidad resulte mayor.

Para determinar la efectividad del algoritmo, se comparan los resultados obtenidos de la clasificación con los valores originales de la última columna (clase) del fichero utilizado.

4. ANÁLISIS DE PRUEBAS Y RESULTADOS

UDTan tiene como datos de salida: - la cantidad de instancias de que consta el fichero, - el número de atributos o variables predictoras y – la matriz de confusión, que muestra el resultado de la clasificación realizada.

La siguiente es la salida obtenida al suministrar el archivo “Soybean” como entrada de UDTan:

Número de Atributos = 35

Número de Instancias = 683

Matriz de Confusión

Tabla 1. Matriz de Confusión UDTan obtenida con el archivo soybean-large

Clasificada como --->	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s
a = 2-4-d-injury	16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
b = alternarialeaf-spot	0	88	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0
c = anthracnose	0	0	44	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
d = bacterial-blight	0	0	0	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
e = bacterial-pustule	0	0	0	0	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0
f = brown-spot	0	1	0	0	0	91	0	0	0	0	0	0	0	0	0	0	0	0	0
g = brown-stem-rot	0	0	0	0	0	0	44	0	0	0	0	0	0	0	0	0	0	0	0
h = charcoal-rot	0	0	0	0	0	0	0	20	0	0	0	0	0	0	0	0	0	0	0
i = cyst-nematode	0	0	0	0	0	0	0	0	14	0	0	0	0	0	0	0	0	0	0
j = diaporthe-pod-&-stem-blight	0	0	0	0	0	0	0	0	0	15	0	0	0	0	0	0	0	0	0
k = diaporthe-stem-canker	0	0	0	0	0	0	0	0	0	0	20	0	0	0	0	0	0	0	0
l = downy-mildew	0	0	0	0	0	0	0	0	0	0	0	20	0	0	0	0	0	0	0
m = frog-eye-leaf-spot	0	9	0	0	0	0	0	0	0	0	0	0	82	0	0	0	0	0	0
n = herbicide-injury	0	0	0	0	0	0	0	0	0	0	0	0	0	8	0	0	0	0	0
o = phyllosticta-leaf-spot	0	0	0	0	0	1	0	0	0	0	0	0	0	0	19	0	0	0	0
p = phytophthora-rot	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	88	0	0	0
q = powdery-mildew	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0
r = purple-seed-stain	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0
s = rhizoctonia-root-rot	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20

Comparación con Otras Técnicas de Aprendizaje

Para realizar las pruebas de funcionamiento de UDTan se usaron como entrada 10 archivos de datos. Con cada uno, se utilizó los siguientes métodos de clasificación para comparar la efectividad del algoritmo TAN: - Naïve Bayes, - Tablas de Decisión, - Algoritmo ZeroR.

Se encontró, que TAN es el algoritmo con mayor efectividad entre los cuatro utilizados, seguido por las técnicas de Naïve Bayes y Tablas de Decisión.

Tabla 2. Cuadro comparativo Técnicas de Aprendizaje

Algoritmo	TAN	NAIVE	TABLAS DE	ZEROR
Archivo		BAIYES	DECISIÓN	
soybean -large ¹	97,95%	93,70%	90,78%	13,47%
iris ²	97,14%	96,00%	97,33%	33,33%
labor-neg ³	99,84%	98,24%	92,98%	64,91%
australian ⁴	96,84%	77,68%	88,11%	55,50%
balance-scale ⁵	89,12%	92,16%	78,72%	46,08%
cars ⁶	100,00%	80,86%	100,00%	62,50%
breast ⁷	100,00%	96,13%	96,70%	65,52%
monk1-full ⁸	96,84%	75,00%	100,00%	50,00%
glass ⁹	98,15%	55,60%	82,24%	35,51%

¹ Este conjunto de datos, determina la posible enfermedad de la soya de frijol, basado en 35 variables predictoras; las enfermedades se agrupan en 19 clases. El número total de instancias es de 683.

² Este ejemplo se refiere al tipo de lirio; el cual se determina con base en el alto y ancho del pétalo y del sépalo. Contiene 150 instancias y tres clases (setosa, virginica, y versicolour). Aquéllas están distribuidas uniformemente de acuerdo a la clase, es decir, existen 50 instancias por cada clase.

³ El conjunto de datos está conformado por 16 atributos, que actúan como variables predictoras y dos clases (contratado, no contratado). El objetivo es determinar si una persona puede o no ser contratada de acuerdo a los 16 atributos. El número de instancias es de 57.

⁴ Esta colección de datos se refiere a la aprobación de solicitudes de tarjetas de crédito australianas. Contiene dos posibles clases (1 ó 0), en las cuales se agrupan 690 instancias, con base en 14 variables predictoras, numéricas y nominales.

⁵ Este ejemplo se generó para modelar resultados de experimentos psicológicos, para determinar el grado de equilibrio. El número total de instancias es de 625, distribuidas en tres clases (49 balanced, 288 left, 288 right).

⁶ Este conjunto de datos se compone de 392 instancias. Se basa en la clasificación de automóviles, según su país de origen (US, Japan, Europe), de acuerdo a 9 atributos, entre otros, año, marca, velocidad, cilindros.

⁷ En este ejemplo se muestran reportes de casos clínicos de pacientes con cáncer, para los cuales, a partir de 10 variables predictoras, se determina si el cáncer es benigno o maligno. El número de instancias es de 699.

⁸ Este ejemplo se refiere a la clasificación binaria de figuras geométricas con base en 6 características, como su color y su forma. Se compone de 432 instancias.

⁹ Se compone de 214 instancias y 10 variables predictoras, que ayudan a determinar la clase en la cual se clasifica un vidrio, de acuerdo al tipo de ventana para la que fue construido.

DNAPromoter ¹⁰	100,00%	99,05%	89,62%	50,00%
---------------------------	---------	--------	--------	--------

5. CONCLUSIONES

- El Teorema de Bayes consiste en el cálculo de la probabilidad de las causas o hipótesis a partir de las observaciones realizadas.
- El conocimiento a priori se puede combinar con datos para determinar la probabilidad de las hipótesis.
- El método bayesiano puede clasificar, combinando las predicciones de varias hipótesis.
- El método TAN de Bayes permite aprender sobre relaciones de dependencia y causalidad.
- En la observación de los resultados obtenidos, se aprecia que la efectividad del método TAN es mayor en todos los casos, superando otros métodos de clasificación.

6. TRABAJOS FUTUROS

Según (Hernández 2004), existen distintas variaciones del algoritmo TAN; entre otras, se puede citar las siguientes:

- Métodos de aprendizaje de redes bayesianas que parten de la estructura del Naive Bayes como estado inicial y, progresivamente añaden arcos hasta llegar a una estructura de TAN (*TAN_{i-m}*).
- Métodos que simultáneamente al proceso de construcción realizan una selección de variables (*sTAN*).

¹⁰ Compuesto por 106 instancias, clasificadas como positivas o negativas, y 59 atributos, este conjunto de datos ayuda a evaluar un algoritmo de aprendizaje híbrido (KBANN) que usa ejemplos para refinar de manera inductiva conocimiento preexistente.

- Métodos que distribuyen los atributos en un conjunto de árboles, un bosque, en lugar de en un único árbol (*FAN*, de *Forest Augmented Naive Bayes*).

Como continuación al trabajo desarrollado, podría realizarse una implementación de las variantes mencionadas, a fin de comparar su efectividad con el método TAN original.

Bibliografía

- [1] CHOW, C.; LIU, C. Approximating discrete probability distributions with dependence trees. IEEE Transactions on Information Theory, 1968.
- [2] FRIEDMAN, N.; GEIGER, D.; GOLDIZMITDT, M. “Bayesian networks classifiers”. Machine Learning, 1997.
- [3] KEOGH, E. J.; PAZZANI, M. Learning augmented Bayesian classifiers: a comparison of distribution-based and non distribution-based approaches. Proceedings of the 7th International Workshop on Artificial Intelligence and Statistics, 1999.
- [4] LARRAGAÑA, P.; LOZANO, J.A. Estimation of Distribution Algorithms. A New Tool for Evolutionary Computation. Kluwer Academic Publishers, 2002.
- [5] MITCHELL, T.M. Machine Learning. Mc Graw Hill, 1997.
- [6] PEARL, J. Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann, 1988.
- [7] HAN, J. and KAMBER, M. Data Mining: Concepts and Techniques. Morgan Kaufmann Publishers. San Francisco (USA), 2001.
- [8] HERNÁNDEZ, J., RAMIREZ, M., FERRI, C. Introducción a la Minería de Datos. Prentice Hall. España, 2004.

Infografía

- [9] LARRAGAÑA, P.; INZA, I. Clasificadores Bayesianos. (Sitio en Internet) Disponible en:
<http://www.sc.ehu.es/ccwbayes/docencia/mmcc/docs/t6bayesianos.pdf>. Acceso el 26 octubre 2005.
- [10] MOLINERO, L. El método bayesiano en la investigación médica. (Sitio en Internet) Disponible en
<http://www.seh-lilha.org/bayes1.htm>. Acceso el 7 septiembre 2005. Publicado en mayo 2002.
- [11] MORALES, E. Redes Bayesianas en Minería de Datos. (Sitio en Internet) Disponible en:
<http://dns1.mor.itesm.mx/~emorales/Cursos/KDD03/node44.html>. Acceso el 26 octubre 2005.
Publicado el 13 septiembre 2005.
- [12] MOREA, L. Curso de Java. (Sitio en Internet) Disponible en:
<http://www.monografias.com/trabajos/java/java.shtml#top>. Acceso el 7 noviembre de 2005.
Publicado 1997.
- [13] SEDA, J. Teorema de Bayes. (Sitio en Internet) Disponible en:
<http://thales.cica.es/rd/Recursos/rd98/Matematicas/28/8.html>. Acceso el 22 agosto 2005.
Publicado el 12 junio 2005.