

ENSAMBLE DE ALGORITMOS BAYESIANOS CON ÁRBOLES DE DECISIÓN, UNA ALTERNATIVA DE CLASIFICACIÓN.

Roberto Portugal & Miguel A. Carrasco

*Departamento de Ciencia de la Computación
Pontificia Universidad Católica de Chile
Av. Vicuña Mackenna 4860(143), Santiago de Chile
rportug@ing.puc.cl & mlcarras@puc.cl*

Abstract: Las redes bayesianas y los árboles de decisión han demostrado ser métodos eficientes en aplicaciones relacionadas con la clasificación. Hoy en día, aún siguen habiendo intentos por combinar estos métodos con el propósito de obtener un algoritmo híbrido que implemente las fortalezas de ambos. Esta investigación se enfoca en establecer la mejora que se obtiene en la exactitud de predicción, cuando algoritmos bayesianos son combinados con árboles de decisión a través de los métodos de ensamble Stacked Generalization y Vote. Los resultados obtenidos muestran que el algoritmo bayesianos mejora significativamente la exactitud de predicción al ser ensamblados con el árbol de decisión. *Copyright © 2006*

Key Words: Stacked Generalization, Vote, Bayes Network, NBtree, Naive Bayes.

1. INTRODUCCIÓN

Eventos raros son aquellos que ocurren muy infrecuentemente y que son difíciles de detectar. Dependiendo del contexto en el que estos se presenten, pueden tener diferentes significados y traer diferentes consecuencias como fraude financiero, anomalías cardíacas, accidentes de tránsito aéreo, intrusión en la red de datos, violación de seguridad informática, etc.

La detección de eventos raros ha sido abordada por el área de la computación como un problema de clasificación. Muchos algoritmos han sido creados y empleados en esta tarea. Los modelos de algoritmos más conocidos para resolver este tipo de problemas son redes neuronales, árboles de decisión, reglas de decisión y redes bayesianas. Dentro de estos modelos, los árboles de decisión son los más utilizados en esta tarea (Ding & Perrizo, 2002), siendo los algoritmos ID3 y sus variantes los más conocidos (como C4.5, J4.8). En general, este tipo de algoritmos se caracterizan porque son entrenados en una fase de aprendizaje en la que aprenden a modelar el comportamiento de los datos/eventos. Para realizar dicho propósito, el conjunto de datos se divide en training set y test set. i) El *training set* es el conjunto

de datos utilizado para modelar el comportamiento del clasificador. ii) El *test set* es el conjunto de datos no conocidos en el momento del entrenamiento, para medir la exactitud de predicción del algoritmo clasificador.

Encontrar un clasificador eficiente no es una tarea fácil. Cada clasificador se caracteriza porque emplea una representación diferente de los datos. Encontrar una buena representación de estos, que calce con el problema a resolver, requiere de tiempo y de varios ensayos previos. En realidad, aún no existe una regla general acerca de qué métodos de clasificación son los más apropiados para cuales tipos de problemas. Estudios previos han demostrado que ciertas partes del espacio de datos es mejor modelado por un método de clasificación en comparación a otros. El empleo de diferentes clasificadores puede proporcionar información complementaria importante sobre la representación de los datos. Esto ha originado que utilizar una combinación o ensamble de clasificadores sea una buena alternativa. Varios métodos para creación de ensamble de clasificadores ya han sido propuestos, no obstante, tampoco existe una clara figura sobre cual método de ensamble es mejor que otro, o cuando emplear un determinado método de ensamble.

Actualmente uno de los métodos de ensamble más utilizados es el método *Stacked Generalization*, el cual ha sido utilizado en varios trabajos experimentales, y es el cual emplearemos en nuestra investigación, junto con el método *Vote*.

Este trabajo experimental fue motivado en parte por la existencia del algoritmo NBtree. NBTree es un algoritmo creado como una solución híbrida, debido a que la eficiencia del algoritmo Naive-Bayes no alcanza el mismo rendimiento de los árboles de decisión cuando el set de datos es muy grande (Kohavi, 1996). Adicionalmente, en este último tiempo varias investigaciones han sido realizadas sobre métodos bayesianos. Los métodos de inducción bayesiana han demostrado ser una clase de algoritmos tan competitivos como los métodos árboles de decisión y redes neuronales (Singh & Provan, 1995), y se están usando exitosamente en muchas aplicaciones relacionadas con la clasificación (Bala *et al.*, 2003).

El objetivo de esta investigación es determinar si la exactitud de clasificación de los algoritmos bayesianos (Bayes Net, Naives Bayes) mejora en alguna medida cuando son combinados con métodos basados en árboles de decisión (J4.8). Adicionalmente, esta investigación presentan los resultados obtenidos de la ejecución individual y combinada de estos algoritmos sobre distintos set de datos.

Este documento está organizado de la siguiente manera: en la sección 2 se amplían los conceptos utilizados en esta investigación y se hace referencia a trabajos previos relacionados; la sección 3 describe el método establecido para el procesamiento y evaluación de los clasificadores en forma individual y combinada; en la sección 4 se discuten los resultados obtenidos; y finalmente, en la sección 5 se presentan las conclusiones.

2. ANTECEDENTES

Esta sección describe con más detalle el método de ensamble y los algoritmos utilizados, así como también hace referencia a algunos trabajos de investigación relacionados.

2.1 Material Complementario

Un clasificador provee una función que mapea/clasifica un registro de datos dentro de una de las distintas clases predefinidas (Kohavi, 1996). Se requiere de varios ensayos previos para encontrar un método de clasificación que sea el mejor en las tareas o problemas a solucionar, y que se comporte también como un buen ensamble de clasificadores.

Un ensamble de clasificadores es un conjunto de clasificadores que combinan de alguna forma las decisiones individuales de cada uno de ellos para clasificar nuevas instancias (Dzeroski & Zenki, 2000). Existen varias razones que justifican el ensamble de clasificadores. Algunas de éstas, son: i) los datos para training pueden no proveer suficiente información para elegir un único mejor clasificador debido a que el tamaño disponible en estos datos es pequeño en comparación al espacio de hipótesis (Dietterich, 2000); ii) la combinación redundante y complementaria de clasificadores mejora la robustez, exactitud y generalidad de toda la clasificación (Kotsiantis & Pintelas, 2004b); iii) diferentes clasificadores utilizan diferentes técnicas y métodos de representación de los datos, lo que permite obtener resultados de clasificación con diferentes patrones de generalización; iv) los ensambles son frecuentemente mucho más exactos que los clasificadores individuales (Dzeroski & Zenki, 2000).

Unos de los métodos existentes para la construcción de ensambles es el método *Stacked Generalization*, conocido también como *Stacking*. Este método combina múltiples modelos que han sido entrenados para una tarea de clasificación, es decir, combina varios clasificadores para inducir un clasificador de nivel más alto con un mejor rendimiento (Ting & Witten, 1997). Un algoritmo de aprendizaje es empleado para determinar como las salidas individuales de los clasificadores deben ser combinadas. En este método, el conjunto de datos de entrada para la fase de training forman los datos del *nivel 0*, y todos los clasificadores son ejecutados sobre este nivel. Los datos del *nivel 1* son las salidas o resultados individuales obtenidos de los clasificadores, y a su vez, la entrada para el algoritmo de aprendizaje. Este algoritmo de aprendizaje trata estos datos como otro problema, y la salida que genera es la predicción final del ensamble. A continuación, describimos cada uno de los métodos de clasificación que emplea esta investigación.

i) Las Redes Bayesianas (BN) representan las dependencias que existen entre los atributos a través de una distribución de probabilidad condicional en un grafo dirigido acíclico (DAG) (Sahami, 1996). La dependencia entre dos atributos es descrita por la presencia de un arco entre ellos, y su influencia causal, por la dirección del arco. La independencia entre atributos se representa por la ausencia de un arco que conecte atributos particulares.

ii) El clasificador Naive Bayes (NB) es un caso especial de una red bayesiana, en el que se asume que los atributos son condicionalmente independientes dado un atributo clase (Singh & Provan, 1995). Este clasificador es muy eficiente en ciertos dominios, debido a que es muy robusto ante atributos irrelevantes.

iii) El árbol de decisión es una estructura en la cual los datos son divididos de acuerdo a un criterio (*test*). Cada nodo del árbol denota un test sobre un atributo. Cada rama representa un resultado del test, y las hojas del nodo representan clases o distribución de clases (Ding & Perrizo, 2002). Cada instancia de datos tiene varios atributos, uno de los cuales (el objetivo o atributo clase), indica la clase a la cual pertenece cada instancia. Los algoritmos ID3, C4.5 y J4.8 son algunos ejemplos de árboles de decisión usados con bastante frecuencia.

iv) El algoritmo NBTree es un clasificador híbrido que intenta utilizar las ventajas de los clasificadores Decisión Tree (segmentación) y Naive Bayes (acumulación de la evidencia). En este algoritmo, el árbol de decisión segmenta los datos y cada segmento de datos, representado por una hoja, es descrito por un clasificador Naive Bayes (Kohavi, 1996). NBTree es un clasificador híbrido nativo, es decir, no está construido a través de un método de ensamble.

2.2 Trabajos Relacionados

Estudios comparativos han sido realizados para analizar y determinar el comportamiento de redes bayesianas respecto a otros algoritmos (redes neuronales y GMM) (Taniguchi *et al.*, 1998). Asimismo, los algoritmos Bayesian networks y Naive Bayes son tomados como puntos de referencia en investigaciones que tratan de mejorar la creación y aprendizaje de redes bayesianas, tales como algoritmos KBD e Info-AS (Singh & Provan, 1995; Sahami, 1996). Otros estudios comparativos sobre ensambles de diferentes algoritmos como Naive Bayes, C4.5, J4.8 y BP, han sido realizados para determinar la eficiencia de nuevos métodos de ensambles (Ting & Witten, 1997; Kotsiantis & Pintelas, 2004a).

El método stacking ha sido bastante evaluado y se ha demostrado que empleando *multi-response model trees* (M5'), como algoritmo de aprendizaje en el nivel 1, se comporta mejor que stacking con *multi-response lineal regresion* (MLR) al momento de determinar la participación de los clasificadores en la decisión final del ensamble (Kotsiantis & Pintelas, 2004b). Por otro lado, también han habido varios intentos por combinar las fortalezas de los clasificadores redes bayesianas y árboles de decisión, para crear un clasificador híbrido nativo (Kohavi, 1996; Bala *et al.*, 2003).

En la mayoría de los trabajos revisados, se realiza un estudio comparativo de nuevos algoritmos bayesianos con otros métodos de clasificación con el fin de evaluar la predicción de éstos. También, estudios comparativos son realizados con el propósito de medir la exactitud de predicción entre

ensambles de clasificadores, o para medir el desempeño de nuevos métodos de ensambles. El aporte de esta investigación radica en determinar si la exactitud de predicción sobre un set de datos, al combinar métodos bayesianos con árboles de decisión, tiene un mejor rendimiento a través del método de ensamble stacking o vote. También se analiza el impacto de contar con más evidencia en el comportamiento de los clasificadores.

3. MÉTODO PROPUESTO

Esta investigación emplea los clasificadores bayesianos Bayes Net, Naive Bayes, NBTree y el árbol de decisión J4.8. Estos clasificadores son evaluados en forma independiente sobre el set de datos experimental y de control, todos con diferente número de instancias. Los set de datos experimentales presentan la misma distribución con el fin de determinar el impacto en el desempeño de los clasificadores cuando éstos procesan más evidencia.

Los algoritmos Bayes Net y J4.8; y Naive Bayes y J4.8, son ensamblados a través de los métodos stacking o vote, y evaluados sobre los mismos set de datos. El algoritmo NBTree es considerado como un punto de referencia entre el desempeño de algoritmos híbridos nativos y algoritmo híbridos construidos con métodos de ensamble.

- Los set de datos para el experimento están compuestos por instancias formadas por 55 atributos, siendo 10 de ellos atributos con valores continuos y el resto con valores binarios. No se aplica ningún filtro ni se realiza ningún pre-procesamiento sobre los datos que reduzca su dimensión con el fin de aliviar el procesamiento.
- Los set de datos experimentales son contaminados con instancias extrañas para alterar ligeramente la distribución uniforme en los datos. Si bien, esta decisión influirá en la exactitud de predicción de los clasificadores, se considera que ayudará en la evaluación de su comportamiento, pues todos los clasificadores son enfrentados a las mismas instancias.

Con el objetivo de evaluar el comportamiento de los clasificadores ante la presencia de mayor evidencia, se emplean ocho set de datos de diferentes tamaños (ver Tabla 1). Los set de datos con mayor número de instancias incluyen los set de datos más pequeños. Así, cada uno tiene la misma evidencia que el set anterior, más nueva evidencia. Respecto a los datos de control, empleamos tres set con una distribución de datos diferente en cada uno de ellos utilizando la base de datos disponible en (Newman *et al.*, 1998).

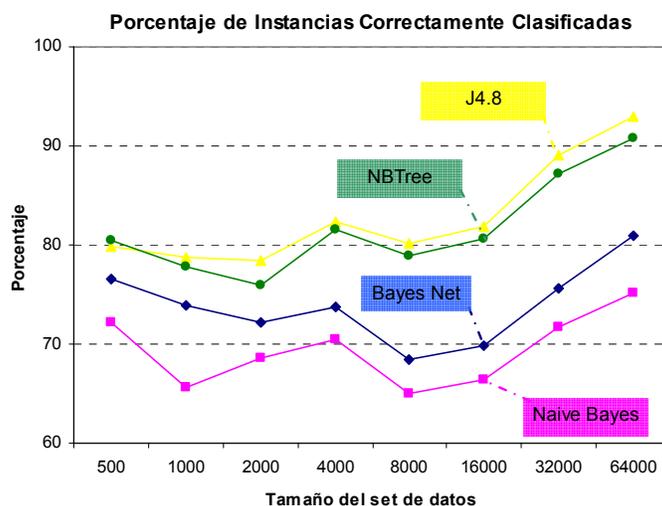


Fig. 1. Resultados de la ejecución individual de los algoritmos BN, NB, NBTree y J4.8, sobre los set de datos experimentales.

Tabla 1 Descripción y tamaño de los set de datos empleados.

	Dataset	Instancias	Atributos	Atributos Continuos	Atributos Discretos
Experimental	dataset-1	500	55	10	45
	dataset-2	1000	55	10	45
	dataset-3	2000	55	10	45
	dataset-4	4000	55	10	45
	dataset-5	8000	55	10	45
	dataset-6	16000	55	10	45
	dataset-7	32000	55	10	45
	dataset-8	64000	55	10	45
Control	balance	625	5	0	5
	monks-2	601	8	1	7
	zoo	101	17	0	17

Para evaluar el rendimiento del clasificador, empleamos el método de selección de datos k -cross validation. Con este método los datasets son aleatoriamente reordenados y divididos en k -fold de igual tamaño. En cada iteración, 1-fold es empleado como dataset de testing y el resto como dataset de training. Se obtiene como resultado final el promedio de los resultados parciales de cada iteración. En particular, utilizamos este algoritmo con $k=10$.

4. RESULTADOS

Todos los clasificadores fueron ejecutados con la configuración por defecto entregada por la herramienta de apoyo para realizar las pruebas (Witten & Frank, 2000). La única excepción fue que utilizamos el algoritmo M5' en el nivel 1 del método stacking, por ser más eficiente (Dzeroski & Zenki, 2000; Kotsiantis & Pintelas, 2004a). La tabla 2 muestra los resultados obtenidos de la ejecución individual y ensamblada de los algoritmos sobre cada uno de los set de datos.

Los resultados muestran que el clasificador BN presenta siempre un mejor desempeño que el clasificador NB (Tabla 2). El árbol de decisión J4.8 presenta mejores resultados que los algoritmos bayesianos BN y NB en todos los set de evidencia.

Tabla 2 Porcentaje de instancias correctamente clasificados por los algoritmos de clasificación en los set de datos experimentales.

Data Set	CLASIFICADORES			HIBRIDO	ENSAMBLE DE CLASIFICADORES			
	BN	NB	J4.8	NBTree	Stacking BN+J48	Stacking NB+J48	Vote BN+J48	Vote NB+J48
ds-1	76.60	72.20	79.80	80.40	80.80	81.20	80.00	79.80
ds-2	73.90	65.70	78.70	77.80	78.70	79.10	79.10	78.30
ds-3	72.25	68.65	78.45	75.95	76.20	75.25	78.70	78.00
ds-4	73.80	70.40	82.38	81.50	82.83	82.63	82.50	81.80
ds-5	68.40	64.97	80.09	78.90	80.50	80.30	79.99	78.69
ds-6	69.85	66.47	81.82	80.67	82.11	82.06	81.54	80.46
ds-7	75.68	71.68	89.05	87.11	87.60	87.49	87.93	87.27
ds-8	80.93	75.20	93.04	90.83	87.13	87.30	92.08	91.52

El algoritmo NBTree presenta un comportamiento cercano a los ensambles con stacking y vote, pero nunca superior a ellos. Además, el algoritmo es ligeramente mejor que el algoritmo Vote en la medida que va procesando set de datos con más instancias. Sin embargo, en los set de datos con mayor número de instancias, el método vote mejora su rendimiento. En la figura 1 se grafican los resultados de los clasificadores individuales de la tabla 2.

En general los algoritmos BN, NB, J4.8 y NBTree presentan un comportamiento similar en cada set de datos. La exactitud de clasificación es mejor en estos algoritmos cuando procesan los set de evidencias con un mayor número de instancias. De estos resultados, se puede concluir que la distribución de los datos es mejor modelada por los algoritmos J4.8 y NBTree.

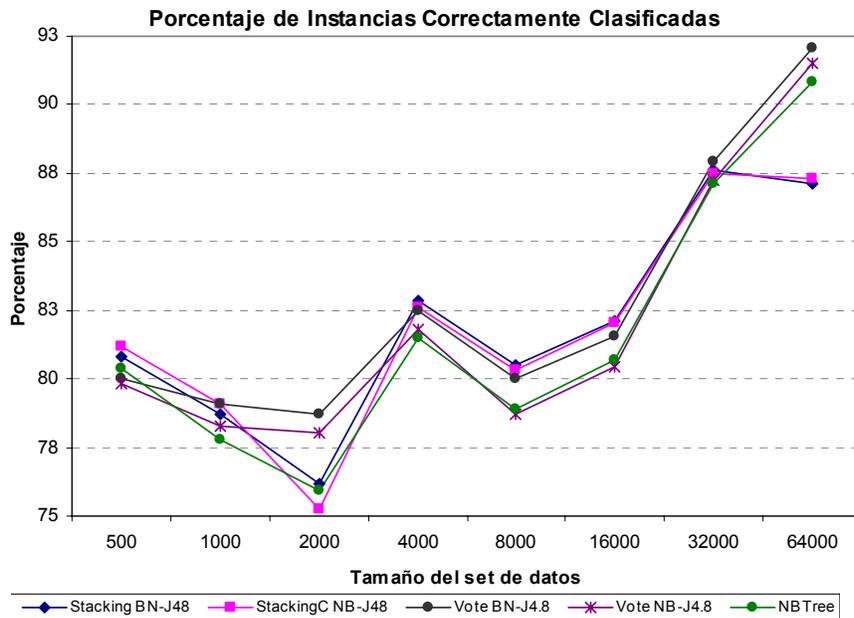


Fig. 2. Resultados obtenidos cuando los algoritmos bayesianos *BN* y *NB* son combinados con el árbol de decisión *J4.8* a través de los métodos *stacking* y *vote*.

La figura 2 muestra que los algoritmos bayesianos mejoran hasta en un 15% el porcentaje de clasificación al ser ensamblados con el árbol de decisión *J4.8*. Sin embargo, la exactitud de clasificación del ensamble no refleja ser mejor a la obtenida por los clasificadores individuales que los componen, en particular al desempeño logrado por el árbol de decisión *J4.8* (ver fig.1).

Todos los ensambles presentan un buen comportamiento en los dos datasets con mayor evidencia. El ensamble con el método *Vote* presenta el mejor comportamiento en el set de evidencias con mayor número de instancias (dataset-8). Además, el comportamiento del algoritmo *NBTree* es muy similar al presentado por el ensamble *NB* y *J4.8*, realizado con el método *Vote*.

El ensamble de algoritmos logra que la exactitud de predicción del nuevo clasificador híbrido sea homogénea; la diferencia en la exactitud de predicción presentada por el algoritmo *Naive Bayes* en la figura 1 desaparece al producirse el ensamble.

Todos los algoritmos ensamblados presentan un comportamiento similar antes el procesamiento del mismo set de datos. En cada set de datos, la exactitud aumenta o disminuye en todos los ensambles. De la figura 2 también se desprende que en la medida que los algoritmos ensamblados procesan set de datos con mayor número de instancias, tienden a mejorar su exactitud de clasificación, así mismo, el tiempo de procesamiento también se incrementa.

En la tabla 3 se muestran los resultados obtenidos al aplicar los mismos algoritmos de clasificación sobre los set de datos de control. Para el proceso de aprendizaje de los clasificadores sobre estos set de datos, se configura el número de iteraciones para el método *cross validation* en 10.

Tabla 3 Porcentaje de instancias correctamente clasificadas en los set de datos de control.

DATAS ET	CLASIFICADORES			HIBRIDO	ENSAMBLE DE CLASIFICADORES		
	BN	NB	J4.8	NBT	Stack BN+J4.8	Stack NB+J4.8	Vote BN+J4.8
balance	23.94	24.10	32.39	24.12	43.68	43.38	30.12
monks-2	42.96	42.91	66.07	69.35	68.14	67.61	65.39
zoo	94.57	94.08	92.61	94.45	95.96	94.96	93.10
dataset-1	75.36	71.68	80.36	77.92	80.94	80.62	80.64

De estos resultados se desprende que para algunas distribuciones (set de datos *zoo*), los algoritmos bayesianos modelan mejor los datos en comparación al algoritmo *J4.8*. Asimismo, se observa que el rendimiento de un ensamble puede ser mayor al desempeño de los clasificadores individuales que lo forman.

5. CONCLUSIONES

Combinar clasificadores puede ser una buena alternativa cuando se pretende obtener un mejor desempeño que el reportado en la ejecución individual de cada clasificador. En este último tiempo, las redes bayesianas han logrado captar la

atención de los investigadores, pero a pesar de ello, aún no queda claro si estos algoritmos bayesianos logran mejorar significativamente su predicción al ser combinados con otros algoritmos de clasificación.

En esta investigación, se realiza en una primera etapa un estudio comparativo entre el desempeño individual de los algoritmos Bayes Net, Naive Bayes, J4.8 y NBTree; en una segunda etapa, los métodos Stacked Generalization y Vote son empleados para crear ensambles de los clasificadores Bayes Net + J4.8; Naive Bayes + J4.8, y sus desempeños también son comparados con los obtenidos en la primera etapa. Todos estos clasificadores, individuales y combinados, fueron entrenados con el método cross validation y evaluados basándose en la exactitud de predicción.

Los resultados obtenidos muestran que los algoritmos bayesianos mejoran significativamente la exactitud de predicción al ser ensamblados con el árbol de decisión J4.8. Si bien, la exactitud de predicción de los algoritmos bayesianos reportó ser menor en comparación al algoritmo J4.8, en los set de datos experimentales ésta puede ser mejorada si los parámetros de configuración de ejecución son afinados; tales como, el estimador a emplear, o algoritmo de búsqueda a utilizar. También se puede concluir que la exactitud de predicción de un clasificador individual o de un ensamble puede ser mejorada con un set de evidencia bastante representativo del problema. La exactitud de un ensamble definitivamente puede ser mayor a la exactitud de predicción obtenida por los algoritmos individuales base.

Como trabajo futuro, quedan pendientes algunas actividades como la ejecución de estos clasificadores con una configuración de más iteraciones del método cross validation; si bien este método demanda bastante recurso computacional, incrementar el número de iteraciones provee mayor precisión sobre los resultados obtenidos. También resta continuar con validación de estos clasificadores en otros set de datos que contengan una distribución diferente. Si bien, en las pruebas experimentales el algoritmo J4.8 reportó mejores resultados que los algoritmos bayesianos, las pruebas de control indicaron que no siempre puede ser así. Finalmente, la reducción de la dimensión de los set de datos es un tema importante por realizar, así como el ensamble de algoritmos bayesianos con redes neuronales, lo cual es también soportado por lo métodos de ensamble descritos aquí.

6. REFERENCIAS

- Bala, J., K. C. Chang, A. Williams and Y. Weng (2003). *A Hybrid Bayesian Decision Tree for Classification*. Workshop on Probabilistic Graphical Models for Classification, Cavtat-Dubrovnik, Croatia.
- Dietterich, T. G. (2000). Ensemble Methods in Machine Learning. *First International Workshop on Multiple Classifier Systems*. I. J. K. a. F. Roli, New York: Springer Verlag. **1857**: 1-15.
- Ding, Q. and W. Perrizo (2002). *Decision Tree Classification of Spatial Data Streams Using Peano Count Trees*. Proc. of the ACM 124 Symposium on Applied Computing, Madrid, Spain.
- Dzeroski, S. and B. Zenki (2000). "Is Combining Classifiers Better than Selecting the Best One." *International Conference on Machine Learning (ICML)*: 123-130.
- Kohavi, R. (1996). *Scaling up the accuracy of naive-Bayes classifiers: a decision tree hybrid*. Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD96), Portland, OR., AAAI Press.
- Kotsiantis, S. and P. Pintelas (2004a). *A Hybrid Decision Support Tool*. Proceedings of 6th International Conference on Enterprise Information Systems, Porto, Portugal.
- Kotsiantis, S. and P. Pintelas (2004b). "Selective Voting." *IEEE 4th International Conference on Intelligent Systems Design and Applications (ISDA)*: 397-402.
- Newman, D. J., S. Hettich, C. L. Blake and C. J. Merz (1998). UCI Repository of machine learning databases [<http://www.ics.uci.edu/mllearn/MLRepository.htm>], University of California, Irvine, Dept. of Information and Computer Sciences.
- Sahami, M. (1996). *Learning Limited Dependence Bayesian Classifiers*. 2th International Conference on Knowledge Discovery in Databases (KDD96), Menlo Park, CA, AAAI Press.
- Singh, M. and G. M. Provan (1995). Efficient Learning of Selective Bayesian Network Classifier. *International Conference on Machine Learning*. Philadelphia, PA., Computer and Information Science Department, University of Pennsylvania.
- Taniguchi, M., M. Haft, J. Hollmén and V. Tresp (1998). *Fraud detection in communications networks using neural and probabilistic methods*. Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'98), Washington, USA.
- Ting, K. M. and I. H. Witten (1997). *Stacked Generalizations: When Does It Work?* Proceedings of the 15th International Joint Conference on Artificial Intelligence, Nagoya, Japan, Morgan Kaufmann.
- Witten, I. H. and E. Frank (2000). *Data Mining: Practical machine learning tools with Java implementations*, Morgan Kaufmann, San Francisco, CA.