

SII, Simposio Argentino de Informática Industrial

Sistema de Soporte de Decisión para la Gestión de fallos en equipos industriales, basado en métodos de ensamble.

Cynthia L. Corso, M. Florencia Pereyra, Gimena Martínez, Fabián Gibellini, Leonardo Ciceri

Centro de Investigación, Desarrollo y Transferencia de Sistemas de Información. Universidad Tecnológica Nacional. Facultad Regional Córdoba. Maestro M. López esquina Cruz Roja Argentina. Ciudad Universitaria – Córdoba.

`cynthia@bbs.frc.utn.edu.ar`, `gimemartinez@bbs.frc.utn.edu.ar`,
`florenciapereyra30@gmail.com`, `speaker@bbs.frc.utn.edu.ar`,
`leocic@bbs.frc.utn.edu.ar`

Resumen. Los fallos en equipos industriales representan eventos críticos en el ámbito de cualquier organización. Su clasificación y caracterización representa un factor importante que apoya el proceso de toma de decisiones en las actividades de mantenimiento. La Minería de Datos ha desempeñado un rol significativo en la evaluación y clasificación de los fallos presentados. Los algoritmos basados en redes bayesianas y árboles de decisión han sido utilizados, de manera individual y en conjunto, para la construcción de modelos de clasificación híbridos, con el propósito de la evaluación y caracterización de fallos. Este trabajo propone el desarrollo de modelos híbridos usando los métodos de ensamble Grading y Vote, combinando las técnicas de redes bayesianas (BayesNet y Naive BayesUpdateable) y árboles de decisión (RandomTree). Se determina la precisión de los métodos de ensamble con los distintos algoritmos, mediante experimentos con el mismo set de datos particionado.

1 Introducción

En el contexto de cualquier organización, la presentación de fallos resiente el normal funcionamiento de las actividades programadas, debido a que cuando estos ocurren pueden desviar o detener por completo el proceso productivo.

Analizando diferentes áreas de la organización, se observa una tendencia en que si los fallos no se encuentran clasificados a partir de un patrón, pueden incurrir en un problema de clasificación, lo que imposibilita actuar de manera preventiva ante los mismos.

La clasificación toma como entrada un conjunto de datos, y utiliza un clasificador o método para obtener la clase a la que pertenece cada uno de los datos del conjunto. El esquema general de los algoritmos de clasificación supone la división del conjunto de datos, en datos de entrenamiento y de prueba. Los primeros se utilizan para modelar el

comportamiento del clasificador, mientras que los segundos permiten validar la precisión del clasificador en el proceso de predicción.

Obtener un clasificador eficiente no es una tarea simple. Cada clasificador se caracteriza porque emplea una representación diferente de los datos. Encontrar una representación de éstos, que mejor se adapte con el problema a resolver, requiere de tiempo y de varios experimentos previos. El uso de distintos algoritmos clasificadores en conjunto, puede proporcionar información complementaria importante sobre la representación de los datos, como así también aumentar la precisión de los modelos resultantes. Esto ha motivado la necesidad de utilizar una combinación o ensamble de clasificadores como un enfoque apropiado para el tratamiento de problemas de clasificación.

Diferentes métodos de ensamble han sido propuestos para la fusión de clasificadores, sin embargo no se encuentran definidos ni criterios ni reglas sobre cuál de éstos es mejor en comparación con otro, o cuando es conveniente utilizar cada uno de ellos.

Para llevar a cabo los experimentos del presente trabajo, se seleccionan dos métodos de ensamble: *Grading* y *Vote*. *Grading* es utilizado en diferentes trabajos experimentales, y siendo comparado con otros métodos ha demostrado su capacidad de rendimiento y performance [1], [2], [3]. Asimismo, a partir de los métodos de ensamble seleccionados, se va a realizar la fusión de diferentes algoritmos pertenecientes a las siguientes técnicas: árboles de decisión y redes bayesianas; experimentando con diferentes combinaciones entre algunos de ellos.

Los árboles de decisión son uno de los más utilizados por su sencillez y su bajo costo computacional. Otras investigaciones han considerado la utilización de las técnicas bayesianas, que demostraron ser tan competitivas como los árboles de decisión y las redes neuronales, para el tratamiento de problemas de clasificación [4].

El objetivo de este trabajo es determinar qué método de ensamble, de los seleccionados para los experimentos, aplicado a la fusión de un conjunto de algoritmos brinda mayor precisión, de manera que la información provista para el proceso de toma de decisiones sea lo más certera posible.

El resto del trabajo está organizado de la siguiente manera: en la sección II, se amplían los conceptos teóricos utilizados en este trabajo y se hace referencia a trabajos previos relacionados; la sección III describe la metodología propuesta para el procesamiento y evaluación de los clasificadores en forma individual y combinada; en la sección IV se discuten los resultados obtenidos; y finalmente, en la sección V se presentan las conclusiones y futuras líneas de investigación.

2 Marco teórico

2.1 Toma de decisiones en la empresa

Una de las principales funciones de los directivos en una empresa es el proceso de toma de decisiones. Una decisión es la elección de la alternativa más adecuada de entre varias posibilidades con el fin de alcanzar un estado deseado, considerando la limitación de recursos. Un aspecto importante en la toma de decisiones es la información de la que dispone el decisor. En la toma de cualquier decisión se necesita algún tipo de información, aunque sea muy escasa. Con la obtención de información se elaboran, sintetizan y almacenan datos sobre un determinado hecho [5]. Esta información es útil antes de la toma de decisiones, pero también enriquece la solución final si se incorpora paulatinamente durante todo el proceso. Por supuesto, a más información, más garantía de éxito en la toma de decisiones, pero hay que tener en cuenta la relación directa entre la información, su coste y el tiempo de recopilación, resumen, etc. Actualmente existe tal cantidad de información sobre cualquier hecho que llega a sobrepasar la capacidad humana de búsqueda y síntesis, por lo que son útiles las bases de datos u otros sistemas de información computarizados [6]. Es aquí donde toman importancia los algoritmos automatizados de Minería de Datos. Se puede considerar a las herramientas de Minería de Datos como herramientas de soporte a la toma de decisiones.

2.2 Minería de Datos

Los datos tal cual se almacenan en las bases de datos no suelen proporcionar beneficios directos; su valor real reside en la información que se puede extraer de ellos, es decir, información que ayude a tomar decisiones o a mejorar la comprensión de los fenómenos que rodean a la empresa. [7]. La minería de datos, asociándola directamente a las actividades de negocios, es el conjunto de metodologías, aplicaciones y tecnologías que permiten reunir, depurar y transformar datos de los sistemas transaccionales e información no estructurada (interna y externa a la compañía) en información estructurada, para su explotación directa o para su análisis y conversión en conocimiento y así dar soporte a la toma de decisiones sobre el negocio. Es el proceso completo de extracción de información, que se encarga además de la preparación de los datos y de la interpretación de los resultados obtenidos, a través de grandes cantidades de datos, posibilitando de esta manera el encuentro de relaciones o patrones entre los datos procesados.

2.3 Técnicas de Minería de Datos

Las técnicas de Minería de Datos intentan obtener patrones o modelos a partir de los datos recopilados. Una técnica constituye el enfoque conceptual para extraer la información de los datos, y, en general es implementada por varios algoritmos. Cada

algoritmo representa, en la práctica, la manera de desarrollar una determinada técnica paso a paso, de forma que es preciso un entendimiento de alto nivel de los algoritmos para saber cuál es la técnica más apropiada para cada problema. Asimismo es preciso entender los parámetros y las características de los algoritmos para preparar los datos a analizar [7]. A continuación se describen algunas de las técnicas de minería de datos.

Aprendizaje Bayesiano. La clasificación bayesiana representa un método basado en estadísticos. Su funcionamiento usa el cálculo de probabilidades a partir del teorema de Bayes, presentado en la ecuación 1.

$$P(h|D) = \frac{P(h|D)P(h)}{P(D)} \quad (1)$$

Dónde:

$P(h)$ es la probabilidad a priori de la hipótesis h . En este caso sería la probabilidad de que se presentara una determinada clasificación del conjunto de datos de entrenamiento.

$P(D)$ es la probabilidad de observar el conjunto de entrenamiento D , cuando es usado para clasificar.

$P(D|h)$ es la probabilidad de observar el conjunto de entrenamiento D en un universo donde se verifica la hipótesis h .

$P(h|D)$ es la probabilidad a posteriori de h , cuando se ha observado el conjunto de entrenamiento D .

El aprendizaje bayesiano puede entenderse como el proceso de encontrar la hipótesis más probable, dado un conjunto de datos de entrenamiento D y un conjunto a priori sobre la probabilidad de cada hipótesis [8]. El modo de aplicación del teorema de Bayes para la clasificación es el cálculo de la hipótesis con mayor probabilidad a posteriori como se muestra en la ecuación 2.

$$h_{MAP} \equiv \underset{h \in H}{\arg \max} \frac{P(D|h)P(h)}{P(D)} \quad (2)$$

Siendo D los datos de entrenamiento y h cada una de las hipótesis, en nuestro caso modos de fallo, que tenemos para clasificar. El subíndice *MAP* se corresponde con *Máximo a posteriori*. De este modo, se clasifica la instancia como aquella que tiene una mayor probabilidad a posteriori.

Dentro de las características más importantes de este tipo de aprendizaje está que cada ejemplo de entrenamiento afecta a la probabilidad de las hipótesis. A pesar de ser este un método en el que se tiene una restricción tan fuerte como la independencia de atributos, se pueden obtener clasificadores precisos aún cuando no se cumple ésta [9].

Redes Bayesianas. Una red bayesiana representa la distribución de probabilidades conjunta para un conjunto de variables [10]. En ellas podemos establecer dependencias entre los atributos mediante un grafo dirigido acíclico, en el que los nodos representan las variables y los arcos las dependencias entre ellas, siendo asignada a cada uno de los nodos del grafo una tabla de probabilidades condicionadas.

Todo método de aprendizaje de este tipo consta de dos elementos. Uno de ellos es la medida de calidad que se usa para determinar cuál es el mejor conjunto de redes bayesianas, y el otro es un algoritmo de búsqueda que se usa para seleccionar un conjunto de redes bayesianas de alta calidad, para elegir la mejor.

Árboles de decisión. De todos los métodos de aprendizaje, los sistemas basados en árboles de decisión son quizás los métodos más fáciles de utilizar y de entender. Un árbol de decisión es un conjunto de condiciones organizadas en una estructura jerárquica, de tal manera que la decisión final a tomar se puede determinar siguiendo las condiciones que se cumplen desde la raíz del árbol hasta alguna de sus hojas [11].

Una de las ventajas de los árboles de decisión es que, en su forma más general, las opciones posibles a partir de una determinada condición son excluyentes. La tarea de aprendizaje, para la cual los árboles de decisión se adecuan mejor es la clasificación. Es por ello la elección del algoritmo perteneciente a esta categoría.

2.4 Métodos de ensamble o fusión

Un método de ensamble es una agrupación de clasificadores base, que combinan sus predicciones siguiendo un determinado esquema, con el objetivo de obtener una predicción más fiable que la que normalmente serían capaces de obtenerlo de forma individual [12].

Grading representa un método de ensamble que se basa en la implementación del algoritmo *grading* y que permite la creación de un clasificador de nivel-0. Este método utiliza un concepto denominado meta-clasificador o también llamado clasificador de nivel-1. Este clasificador intenta resolver como combinar los resultados obtenidos en los algoritmos de nivel-0. La tarea de aprendizaje para cada clasificador de nivel-1 es predecir si la predicción del clasificador de nivel-0 es incorrecta. Un mecanismo de votos con pesos sobre las predicciones de los clasificadores base, produce como resultado la predicción final de la clase. El peso asignado al voto de cada clasificador base es la confianza en que su predicción sea correcta. Este peso es estimado por el meta-clasificador asociado al clasificador base [13].

Vote es uno de los métodos de ensamble más simple y sencillo, que en la gran mayoría de los casos es altamente efectivo. Puede ser aplicado tanto para el tratamiento de problemas de clasificación como regresión.

El mecanismo de funcionamiento de este método de ensamble se basa generando dos o más sub-modelos. Sobre cada sub-modelo se efectúan predicciones que se com-

binan utilizando la media, permitiendo que cada sub-modelo vote para determinar cuál sería el resultado [14].

En este trabajo se consideran los métodos de ensamble *Grading* y *Vote* como mecanismo de fusión de los algoritmos base seleccionados.

3 Metodología

Este trabajo utiliza los clasificadores bayesianos Naive BayesUpdateable, Bayes Net y el árbol de decisión RandomTree. Los algoritmos citados son implementaciones provistas por la herramienta de aprendizaje automático Weka en su versión 3.6. El set de datos original cuenta con 6400 instancias, 36 atributos continuos y 1 atributo discreto. El mismo fue sometido al proceso de minería de datos, el cual cuenta con seis fases. En la fase tres *Preparación de datos* se eliminaron aquellos datos que se consideraron innecesarios para el proceso de extracción del conocimiento; además, se convirtieron algunos atributos continuos en discretos. De esta manera se logró limpiar el set de datos con el cual se llevaron a cabo los experimentos, logrando que los clasificadores mencionados al comienzo de esta sección sean evaluados de manera individual sobre un set de datos de 6400 instancias, 6 atributos continuos y 5 atributos discretos. Este set de datos se obtuvo del caso de estudio que se presenta a continuación.

Caso de estudio

En una empresa de productos alimenticios, la cual cuenta con cinco sucursales y en cada una de ellas funcionan 3 máquinas productoras, una máquina llega a bombear 100000 litros de aceite en 1 hora. Con el paso del tiempo, la misma comienza a perder su función productiva bombeando menor cantidad de lo que debe, o directamente no expulsando el mismo. Esto influye directamente en el proceso de producción, debido a que la calidad y/o cantidad lograda no es igual a la esperada. Entre las causas más comunes se encuentran la obstrucción parcial o total de los conductos y el desgaste mecánico de los componentes, debiéndose esto a la falta de mantenimiento. En la siguiente tabla se presentan las variables consideradas en el set de datos.

Tabla 1. Variables consideradas en el set de datos.

Variabes	Descripción
Fecha	Fecha en la que se reportó el incidente
Número de sucursal	Sucursal de la fábrica en la cual se encuentra la máquina afectada
Número de máquina	Número de identificación de la máquina afectada
Componente afectado	Elemento perteneciente a la máquina, que se encuentra implicado en la causa
Causa	Motivo por el cual la máquina deja de bombear o bombea menor cantidad de aceite
Acción correctiva	Operación que se realiza para volver la máquina a

Tiempo ocioso	su estado ideal de producción Cantidad de tiempo en que la máquina no se encuentra productiva
Porcentaje de disminución de calidad	Porcentaje en que la calidad del producto se ve disminuida
Porcentaje de disminución de cantidad	Porcentaje en que la cantidad producida del producto se reduce
Gravedad de la causa	Impacto causado en la producción
Rol que reportó	Nombre del rol de la persona que reportó el defecto

Luego de haber limpiado el set de datos, el mismo fue particionado en 8, y cada partición posee 800 instancias más que la anterior. Por ejemplo, partición 1: 800 instancias, partición 2: 1600 instancias, etcétera. El propósito por el cual se considera la utilización de particiones con distintas instancias, es demostrar el comportamiento de los clasificadores ante la presentación de un mayor volumen de datos.

Los algoritmos Naive BayesUpdateable, Bayes Net y RandomTree, son fusionados por medio de los métodos de ensamble Grading y Vote, y evaluados sobre las particiones del set de datos seleccionado. En los experimentos ejecutados no se aplica ningún método de filtro, ni se realiza ninguna tarea de pre-procesado sobre los datos para reducir el nivel de dimensionalidad de los mismos.

Para la ejecución de los experimentos sobre los algoritmos seleccionados, se utilizó la configuración de parámetros por defecto que define la herramienta Weka, para cada uno de ellos.

La técnica utilizada para la evaluación de resultados en los experimentos es K-CrossValidation, este método consiste en dividir el conjunto de entrenamiento en "K" subconjuntos. Uno de los subconjuntos se utiliza como datos de prueba y los (K-1) como datos para efectuar el proceso de entrenamiento. Este proceso se repite durante "K" iteraciones, con cada uno de los posibles subconjuntos de datos de prueba. La elección del número de "K" iteraciones en todos los experimentos será de un valor diez, que representa el valor por defecto.

4 Resultados

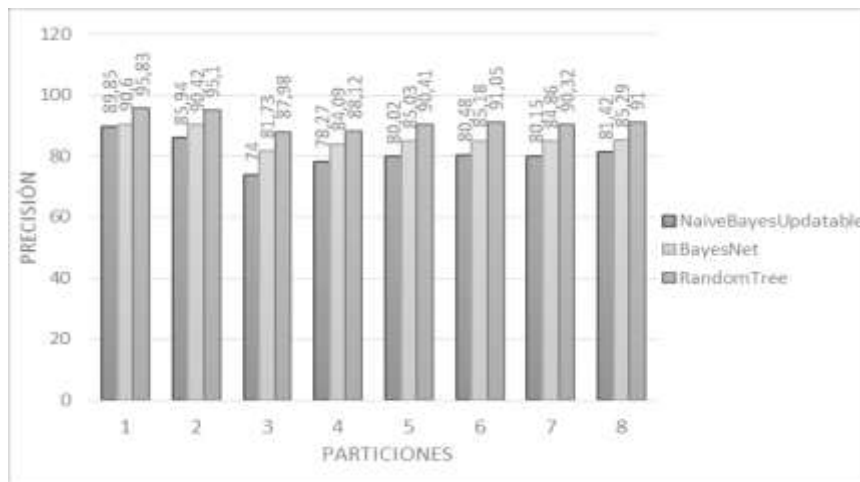
En la tabla 2 se visualizan los resultados obtenidos en la ejecución de los experimentos con cada uno de los algoritmos de manera individual. En cambio, la tabla 3 refleja los resultados de la fusión de los algoritmos, mediante los dos métodos de ensamble seleccionados, sobre las particiones del set de datos considerado.

Tabla 2. Precisión de los algoritmos en cuanto a la clasificación de instancias.

Particiones	Instancias	NaiveBayesUpdatable	BayesNet	RandomTree
Partición 1	800	90,75	91	96,5
Partición 2	1600	86,75	91,06	95,38
Partición 3	2400	74,08	82,13	88,25
Partición 4	3200	78,97	84,22	88,56
Partición 5	4000	80,45	85,48	90,65
Partición 6	4800	80,81	85,69	91,35
Partición 7	5600	80,79	85	90,95
Partición 8	6400	81,97	85,8	91,06

Antes de comenzar con la ejecución de los experimentos, se planteó la hipótesis de que al aumentar la cantidad de instancias en las distintas particiones del set de datos, la precisión de los algoritmos aumentaría. Sin embargo, luego de realizar los experimentos, y como se puede observar en la tabla 2, dicha hipótesis fue rechazada.

Analizando los resultados obtenidos en la tabla 2, se puede evidenciar que de los tres algoritmos seleccionados, RandomTree es el que presenta, en todos los casos, un mejor comportamiento en las predicciones. Esto también se puede visualizar en la Fig. 1.

**Fig. 1.** Precisión de clasificación de los algoritmos.**Tabla 3.** Precisión en la clasificación usando métodos de ensamble.

Particiones	Instancias	Grading	Grading	Vote	Vote
		NBU+RT	BN+RT	NBU+RT	BN+RT
Partición 1	800	95,88	95,88	96,38	96,5
Partición 2	1600	94,25	94,25	95,56	95,5
Partición 3	2400	87,83	87,83	88,54	88,54

Partición 4	3200	89,91	89,91	88,78	88,88
Partición 5	4000	90,78	90,78	90,75	90,9
Partición 6	4800	90,9	90,9	91,52	91,52
Partición 7	5600	90,63	90,63	91	91,04
Partición 8	6400	91,11	91,11	91,25	91,28

Del análisis de la tabla 3 se concluye en primera instancia que, al combinar los algoritmos mediante un método de fusión, la precisión aumenta con respecto a los experimentos de los mismos ejecutados de forma individual. Se puede visualizar que los algoritmos bayesianos obtienen una mayor precisión al ser combinados con un algoritmo basado en la construcción de un árbol de decisión como es **RandomTree**.

Además, se puede observar que el método de fusión **Grading** no presenta variaciones en sus resultados con ninguno de los pares de algoritmos seleccionados.

Por otro lado, en la mayoría de los experimentos se puede evidenciar que, el método **Vote** ensamblado con los algoritmos **NaiveBayesUpdatable** y **RandomTree** presenta mayor exactitud que el método **Grading** ensamblado con el mismo par de algoritmos. Lo mismo sucede al fusionar el método **Vote** con el otro par de algoritmos, **BayesNet** y **RandomTree**, obteniendo nuevamente mayor precisión que el método **Grading** combinado a los algoritmos antes mencionados. En la Fig. 2 se puede apreciar lo mencionado anteriormente. Sin embargo, se presenta en una única partición que **Grading** obtiene mayor precisión que **Vote**, y esto se ve reflejado en la Fig. 3.

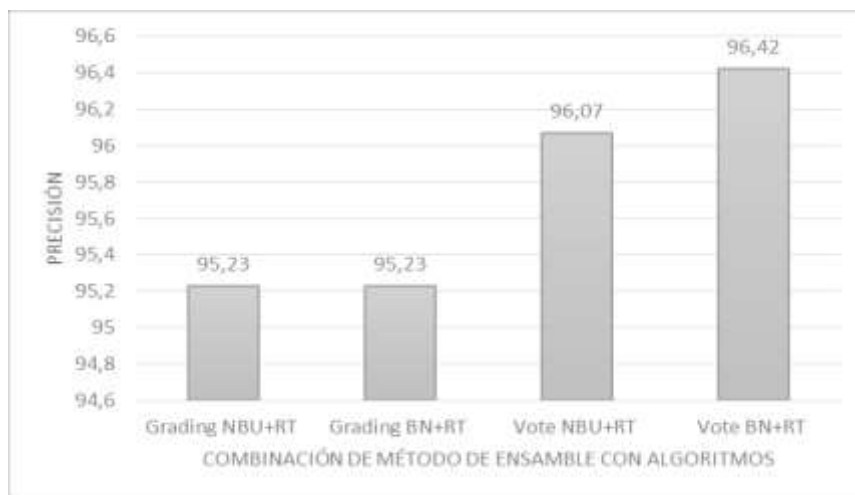


Fig. 2. Resultados de la precisión de los métodos de ensamble fusionando los algoritmos, en la partición 1.

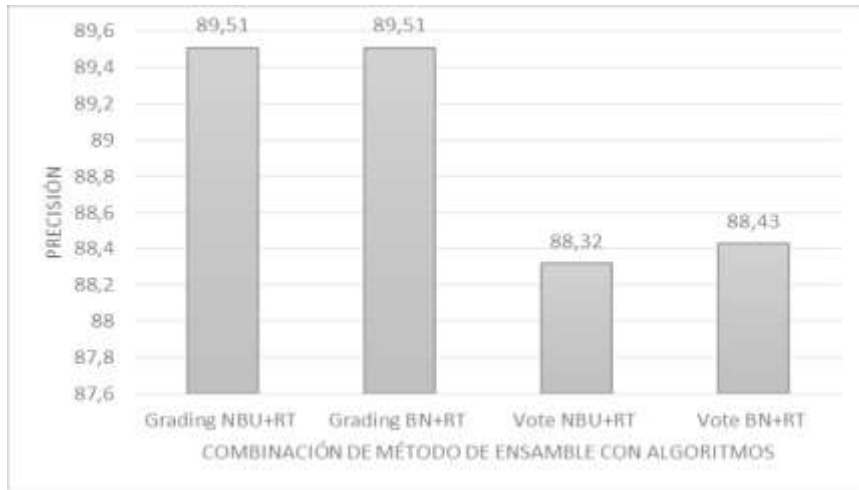


Fig. 3. Resultados de la precisión de los métodos de ensemble fusionando los algoritmos, en la partición 4.

Finalmente luego de realizar un análisis más exhaustivo, se pudo concluir que, si bien la diferencia entre las precisiones obtenidas por el método Vote con cada uno de los dos pares de algoritmos es mínima y en algunos casos nula, la fusión de los algoritmos **BayesNet** y **RandomTree** obtiene una mínima diferencia positiva. Por lo tanto es el método sugerido para apoyar al proceso de toma de decisiones. No obstante, en la Fig. 4 se puede apreciar que Vote presenta la misma precisión para las dos fusiones.

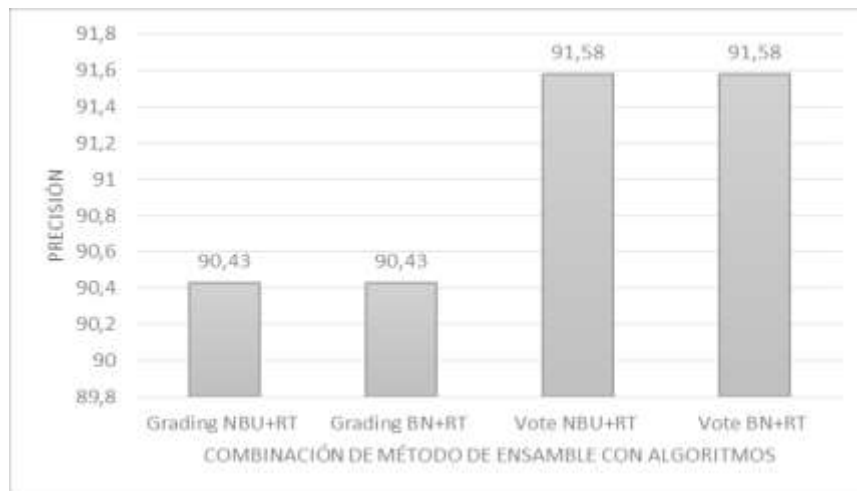


Fig. 4. Resultados de la precisión de los métodos de ensemble fusionando los algoritmos, en la partición 6.

5 Conclusiones

La utilización de métodos de ensamble representa una adecuada alternativa para el tratamiento de problemas de clasificación supervisada, en comparación con el desempeño obtenido en la ejecución de los algoritmos seleccionados de manera individual.

Este trabajo realiza en primera instancia un análisis comparativo acerca del desempeño individual de los algoritmos Naive BayesUpdateable, BayesNet y RandomTree; en una segunda instancia se considera la fusión de los mismos utilizando los métodos de ensamble Grading y Vote. Los resultados son comparados con los obtenidos de forma individual, utilizando el método K-CrossValidation, y evaluados tomando como medida de valoración la exactitud de la clasificación.

Luego del análisis de las experimentaciones, se puede concluir que en situaciones donde se requiera llevar adelante un proceso de toma de decisiones, el método Vote fusionado con los algoritmos BayesNet y RandomTree es el que obtiene mayor precisión en la clasificación de instancias, por lo cual esta combinación sería la recomendada para utilizarse en un sistema que brinde soporte al proceso de toma de decisiones.

Como futuras líneas de investigación, quedan pendientes realizar los experimentos con otros aspectos que no fueron considerados en este trabajo. Uno de los más importantes es realizar los ensayos con más iteraciones del método K-CrossValidation. Además, realizar un análisis del comportamiento de las particiones 4 y 6, ya que son casos particulares en los cuales los resultados difieren de las conclusiones arribadas.

Referencias

1. A. Seewald y J. Furnkranz: Grading Classifiers. Recuperado de <http://www.ofai.at/cgi-bin/get-tr?paper=ofai-tr-2001-01.pdf> (2001)
2. I. Ledesma Espino: Aprendizaje Automático en Conjunto de Clasificadores heterogéneo y Modelado de Agentes (Tesis de Doctorado). Universidad Carlos III de Madrid (2004)
3. A. Seewald y J. Furnkranz: An Evaluation of Grading Classifiers. International Symposium on Intelligent Data Analysis (2001) 115-124
4. Q. Ding y W. Perrizo: Decision Tree Classification of Spatial Data Streams Using Peano Count Trees. Proc. of the ACM 124 Symposium on Applied Computing (2000)
5. González Ramírez, M.R.: Sistemas de información para la empresa. Recuperado de <https://riunet.upv.es/bitstream/handle/10251/16502/TomaDecisiones.pdf>. Publicaciones de la Universidad de Alicante (2001)
6. Gil Pechuán, I.: Sistemas y tecnologías de la información para la gestión. Recuperado de <https://riunet.upv.es/bitstream/handle/10251/16502/TomaDecisiones.pdf>. Universidad Politécnica de Valencia. (1996)
7. Jose M. Molina y Jesús García: Técnicas de Análisis de Datos (2004)
8. T. M. Mitchell: Machine Learning. McGraw-Hill International Edition, New York (1990)
9. B. Sierra Araujo: Aprendizaje Automático: Conceptos básicos y avanzados. Aspectos prácticos utilizando software WEKA. Prentice-Hall Madrid (2006)
10. D. Heckerman, D. Geiger y D. Chickering: Learning Bayesian-Networks: The Combination Knowledge and Statistical Data. Machine Learning, Vol. 20 (1999) 197-243

11. C. Pérez López y D. González: Minería de Datos: Técnicas y Herramientas. Thomson Ediciones Paraninfo Madrid (2007)
12. J. Orallo Fernández y M. J. Quintana Ramírez: Extracción Automática de Base de Datos e Ingeniería de Software. Prentice-Hall Madrid (2005)
13. K. Ting y I. Witten: Stacked generalization: when does it work?. In Proceedings of the International Joint Conference on Artificial Intelligence (1997)
14. E. Bauer y R. Kohavi: An Empirical comparison of voting classification algorithms: Bagging, boosting, and variants. Machine Learning, Vol. 36. (1999) 105-169
15. S. Dzeroski y B. Zenki: Is Combining Classifiers Better than Selecting the Best One. International Conference on Machine Learning (ICML) (2000) 123-130
16. S. Kotsiantis y P. Pintelas: A Hybrid Decision Support Tool. Proceedings of 6th International Conference on Enterprise Information Systems (2004)
17. S. Kotsiantis y P. Pintelas: Selective Voting. IEEE 4th International Conference on Intelligent Systems Design and Applications (ISDA) (2004) 397-402
18. G. Weiss: UCI Repository of machine learning databases. <http://strom.cis.fordhman.edu/~gweiss/data-mining/datasets.html>. University of Fordhman, Bronx N. Y., Dept. of Information and Information Sciences.
19. J. Bala, K. C. Chang y Y. Weng: A Hybrid Bayesian Decision Tree for Classification. Workshop on Probabilistic Graphical Models for Classification (2003)
20. Piatesky – Shapiro, G y Frawley, W.: Knowledge Discovery in Data Bases. http://www.scielo.org.ve/scielo.php?script=sci_arttext&pid=S1012-15872007000100008. (1991)
21. Q. Ding y W. Perrizo: Decision Tree Classification of Spatial Data Streams Using Peano Count Trees, Proc. of the ACM 124 Symposium on Applied Computing, 2000
22. Dietterich, T. G: Ensemble Methods in Machine Learning. First International Workshop on Multiple Classifier Systems. I. J. K. a. F. Roli, New York: Springer Verlag. 1857: 1-15,2000.
23. Kotsiantis, S. and P. Pintelas: Selective Voting. IEEE 4th International Conference on Intelligent Systems Design and Applications (ISDA): 397-402, 2004.
24. Dzeroski, S. and B. Zenki: Is Combining Classifiers Better than Selecting the Best One. International Conference on Machine Learning (ICML): 123-130, 2000.
25. J. Orallo Fernández y M. J. Quintana Ramírez: Extracción Automática de Base de Datos e Ingeniería de Software, Madrid: Prentice-Hall, 2005.
26. Tong Tong, Qinquan Gao, Ricardo Guerrero, Christian Ledig, Liang Chen, Daniel Rueckert, and the Alzheimer's Disease Neuroimaging Initiative: A Novel Grading Biomarker for the Prediction of Conversion From Mild Cognitive Impairment to Alzheimer's Disease. Institute of Electronical and Electronics Engineers. 2016.
27. Mostafa Abdollahpur, Shadi Ghiasi, Mohammad Javad Mollakazemi, Ali Ghaffari: Cycle selection and neuro-voting system for classifying heart sound recordings. Institute of Electronical and Electronics Engineers. 2016.
28. Vidhya Natarajan, Tzu-Yi Hung, Sriram Vaikundam Liang-Tien Chia: Convolutional Networks for Voting-based Anomaly Classification in Metal Surface Inspection. Institute of Electronical and Electronics Engineers. 2017.
29. Lovedeep Gondara. RPC: An efficient classifier ensemble using random projections. 14th International Conference on Machine Learning and Applications. 2015.