

Evaluación del algoritmo de clasificación basado en reglas JRip para datos standard y desbalanceados

Evaluation of the classification algorithm based on rules JRip for standard and unbalanced data

Laidy De Armas Jacomino^{1*}, Danilo Valdes Ramirez²

Resumen Las técnicas de minería de datos se usan en el aprendizaje automatizado de máquinas desde hace varios años para inferir conocimientos a partir de las informaciones almacenadas en empresas e instituciones. Algunas de estas técnicas más usadas son la clasificación, regresión, agrupamiento, etc. La clasificación supervisada tiene el objetivo de asignar objetos a clases predefinidas a partir de un grupo de objetos previamente almacenados que ya están clasificados. Existen para tareas de clasificación diversos algoritmos, estos se agrupan usualmente en familias atendiendo a la forma de representar el conocimiento que extraen de los datos. Las reglas de decisión son una forma de representación del conocimiento muy usada cuando se quiere que los usuarios entiendan los resultados obtenidos. El presente estudio tiene como objetivo evaluar el comportamiento del algoritmo de clasificación basado en reglas JRip para problemas de clasificación standard y desbalanceados. Este algoritmo es comparado además con otros tres de su familia Ridor, DecisionTable y OneR. Para la evaluación experimental se usa la validación cruzada y para analizar los resultados obtenidos de la misma el test de Friedman y el test post-hoc Finner. Se presentan y discuten los resultados obtenidos de las evaluaciones para los diferentes tipos de problemas de clasificación de minería de datos.

Abstract Data mining techniques have been used in automated machine learning for several years to infer knowledge from information stored in companies and institutions. Some of these techniques more used are the classification, regression, clustering, etc. The supervised classification aims to assign objects to predefined classes from a group of previously stored objects that are already classified. There are several algorithms for classification tasks, these are usually grouped in families according to the way they represent the knowledge they extract from the data. Decision rules are a widely used form of knowledge representation when users are expected to understand the results obtained. The present study aims to evaluate the behavior of the classification algorithm based on rules JRip for standard and unbalanced classification problems. This algorithm is also compared to three other algorithms based on rules Ridor, DecisionTable and OneR. For the experimental evaluation, cross-validation is used and to analyze the results obtained from the same the Friedman test and the post-hoc Finner test. We present and discuss the results obtained from the evaluations for the different types of data mining classification problems.

Palabras Clave

Clasificación — datos desbalanceados — datos con ruido —JRip — reglas —valores perdidos

^{1,2}Facultad de Ciencias Informáticas, Universidad de Ciego de Ávila Máximo Gómez Báez, Ciego de Ávila, Cuba. laidy@unica.cu

*Autor para Correspondencia

Introducción

La minería de datos se ha convertido hoy en día en una herramienta poderosa para analizar la información que tienen las empresas para la toma de decisiones. El aprendizaje automatizado (del inglés Machine Learning) utiliza las técnicas de minería de datos para convertir estas informaciones en conocimientos que aportan valor agregado. Para representar los conocimientos descubiertos por los métodos de aprendizaje automatizado se usan diversas formas o estilos como tablas, árbol de decisión, reglas, etc[14].

Las reglas utilizan un formato Si-Entonces(del inglés If-Then) para representar el conocimiento, la parte If de una regla es una condición (también llamada premisa o antecedente), y la parte Then de la regla (también llamada acción, conclusión o consecuente) permite inferir un conjunto de hechos nuevos si se verifican las condiciones establecidas en la parte If [1]. Igualmente a partir de un conjunto de ejemplos (a partir de ahora datasets) diversos algoritmos de minería de datos extraen conocimiento en forma de reglas, donde los antecedentes son ejemplos (del inglés instances) y los consecuentes pueden

ser clases o conjuntos de clases. Un árbol de decisión se puede expresar como un conjunto de reglas. Para obtener reglas a partir de un árbol cada hoja se convierte en una conclusión y el antecedente de la regla incluye una condición para cada nodo en el camino desde la raíz hasta la hoja.

Los algoritmos de clasificación basados en reglas y en árboles de decisión se consideran técnicas de *caja blanca* pues son altamente interpretables y por ello son muy utilizados en la toma de decisiones. Algunas de las aplicaciones donde se han empleado son la predicción del fracaso escolar[9], diagnóstico de huecos de tensión [12], pronóstico de supervivencia de infarto cerebral aterotrombótico[10], etc.

El presente estudio tiene como objetivo evaluar el comportamiento del algoritmo de clasificación basado en reglas JRip para problemas de clasificación estándar y desbalanceados. JRip es comparado además con otros tres de su familia Ridor, DecisionTable y OneR.

Existen varios trabajos similares al que se propone donde se evalúa el rendimiento de diversos clasificadores basados en reglas. En [4] se comparan 179 clasificadores de diversas familias sobre 121 datasets del repositorio University of California Irvine Machine Learning Repository (UCIMLR)¹. De los clasificadores comparados en [4] 12 son basados en reglas, entre estos se encuentran los evaluados en este trabajo.

En [11] se comparan los clasificadores basados en reglas JRip, RIDOR (Ripple Down Rule learner) y DecisionTable con el datasets Iris del repositorio UCIMLR, usando diversas métricas fundamentalmente calidad de la clasificación y tiempo de ejecución. El clasificador RIDOR dio mejores resultados con este dataset en particular.

Por otro lado en [13] se compara el comportamiento de los algoritmos J48, REPTree, PART, JRip y Ridor en términos de complejidad de la estructura del árbol y precisión de la clasificación usando 7 datasets estándar del UCIMLR. De estos algoritmos J48 y REPTree generan la salida en forma de árbol mientras que PART, Ridor y JRip generan reglas. De los algoritmos basados en reglas, Ridor se comportó mejor que PART y JRip debido al alto porcentaje de precisión de la clasificación en 5 de los 7 datasets.

El presente trabajo se estructura en tres secciones. En la sección 1 *Algoritmos basados en reglas* se describe el funcionamiento del algoritmo JRip y se explica brevemente los clasificadores Ridor, DecisionTable y OneR. En la sección 2 *Materiales y métodos* se presentan los datasets usados para evaluar JRip, las métricas empleadas para medir su comportamiento y la metodología que se sigue en este estudio. En la última sección *Resultados y discusión* se presentan y analizan los resultados experimentales obtenidos a partir del trabajo. Por último, se presentan las conclusiones del estudio.

1. Algoritmos basados en reglas

Los métodos de inducción de reglas permiten a partir de un conjunto de ejemplos de entrenamiento etiquetados

obtener reglas que permitan clasificar nuevos ejemplos. De forma general los sistemas de inducción de reglas evalúan los atributos del conjunto de entrenamiento y deciden cuáles usar para discriminar entre las diferentes clases.

JRip es la implementación del método RIPPER (Repeated Incremental Pruning to Produce Error Reduction) en la herramienta de minería de datos Weka. El método original RIPPER fue propuesto por William W. Cohen en 1995 [2]. Este método consiste en una poda incremental repetida para producir una reducción del error, una versión optimizada del método IREP (Incremental Reduced Error Pruning). De forma general consiste en hacer una lista ordenada de reglas conjuntivas y evaluarlas en orden para encontrar la primera regla que se cumple sobre el ejemplo a clasificar. Una vez encontrada dicha regla se ha encontrado la regla más eficiente para ese ejemplo y es asignado con una etiqueta de valor de salida. RIPPER es un método directo, que extrae reglas directamente desde los datos. Este método está compuesto por tres procedimientos: RIPPER, BUILDRULESET y OPTIMIZERULESET.

JRip cuenta con dos etapas: una etapa de construcción de las reglas y otra de optimización. La etapa de construcción de las reglas a su vez cuenta con una fase de crecimiento y otra en que las reglas son podadas. Este nuevo método soluciona algunos problemas del RIPPER original sobre la longitud del RuleSet o conjunto de reglas y la precisión. A continuación se muestran los pasos de la implementación del algoritmo JRip en Weka:

Inicializar $RS=\{\}$, y para cada clase desde la menos prevalente hasta la más frecuente hacer:

1. **Etapla de construcción:** Repetir los incisos a) y b) hasta que la longitud de descripción (DL) del conjunto de reglas y de los ejemplos sea 64 bits más grande que la DL más pequeña conocida, o que no existan ejemplos positivos.
 - a) **Fase de crecimiento:** Desarrollar una regla añadiendo golosamente antecedentes o condiciones a la regla hasta que sea perfecta, por ejemplo 100 % precisa. El procedimiento prueba cada valor posible para cada atributo y selecciona la condiciones con la mayor ganancia de información: $p(\log(p/t) - \log(P/T))$
 - b) **Fase de poda:** Incrementalmente poda cada regla y permite el podado de cualquier secuencia final de antecedentes. La métrica de podado es $(p-n)/(p+n)$ – pero es realmente $2p/(p+n) - 1$, así que en esta implementación simplemente se usa $p/(p+n)$ (realmente $(p+1)/(p+n+2)$), así que si $p+n$ es 0, esto es 0.5)
2. **Etapla de optimización:** Luego de generar el conjunto de reglas iniciales $\{R_i\}$, genere y pode dos variantes de cada regla R_i de los datos randomizados usando el procedimiento a) y b). Pero una variante es generada desde

¹<http://archive.ics.uci.edu/ml/datasets.html>

una regla vacía mientras que otra es generada añadiendo golosamente antecedentes a la regla original. Además la métrica de poda usada aquí es $(TP+TN)/(P+N)$. Entonces la DL más pequeña posible para cada variante y la regla original son calculadas. La variante con la DL mínima es seleccionada como la representación final de R_i en el conjunto de reglas. Después de que todas las reglas en $\{R_i\}$ hayan sido examinadas y si aun hay positivos residuales, más reglas son generadas basado en los positivos residuales usando el escenario de construcción nuevamente.

3. Eliminar las reglas del conjunto de reglas que puedan incrementar la DL del conjunto de reglas completo si estaba en él y se añade el conjunto de reglas resultante a RS.

La implementación del clasificador JRip en Weka asume tal y como se muestra en el Cuadro 1 clases nominales, binarias y valores perdidos. Además los atributos pueden tomar valores nominales, numéricos, fechas, valores perdidos, etc.

Para evaluar el desempeño de JRip en este trabajo se compara con tres clasificadores de su familia Ridor, DecisionTable y OneR. A continuación se describen brevemente cada uno de ellos. Las propiedades sobre los valores de las clases y atributos que asumen se muestran en el Cuadro 1.

1. **RIDOR**: Genera una regla por defecto (predeterminada) y luego toma las excepciones para la regla predefinida con la mínima tasa de error [5]. Entonces se genera la mejor excepción para cada excepción iterando hasta lograr disminuir el error. Luego genera una expansión similar a un árbol de excepciones. La excepción es un conjunto de reglas que predice clases. Este algoritmo es usado para generar dichas excepciones.
2. **DecisionTable** A partir de los datos de entrenamiento construye una tabla formada por un subconjunto -llamado esquema- de sus atributos y una selección de los ejemplos de entrenamiento. Para clasificar un nuevo ejemplo el algoritmo busca en la tabla todos los ejemplos que concuerdan, teniendo en cuenta sólo los atributos que forman el esquema. Si no se encuentra ningún ejemplo que concuerde, el algoritmo devuelve la clase más cercana de la tabla; en otro caso, se devuelve la clase mayoritaria del conjunto de ejemplos que concordaron [8].
3. **OneR**: El principio de OneR o 1R es hacer reglas que prueban un solo par atributo-valor. Se prueban todos los pares atributo-valor y se selecciona el que ocasione el menor número de errores [7]. Los atributos numéricos los trata como continuos. A continuación se muestra brevemente los pasos que se siguen en este algoritmo:
 - Para cada atributo,
 - Para cada valor del atributo, haga una regla como sigue:

- Cuente cuántas veces aparece cada clase.
- Encuentre la clase más frecuente.
- Haga que la regla asigne esa clase a este atributo-valor.

- Calcule el índice de error de las reglas.
- Elija las reglas con la tasa de error más pequeña.

2. Materiales y métodos

Para evaluar el desempeño del algoritmo basado en reglas JRip se emplearon diferentes métricas según los problemas de clasificación estándar y desbalanceados. Los datasets empleados en la experimentación fueron obtenidos en su mayoría del sitio web de la herramienta de minería de datos Keel² y convertidos al formato .arff del Weka. Solamente los datasets lymph y zoo fueron obtenidos directamente del UCIMLR.

Para los problemas de clasificación estándar se empleó la métrica *precisión de la clasificación* (percent correct en Weka). Esta métrica mide la proporción de casos clasificados correctamente[1] y puede ser empleada cuando hay clases binarias o múltiples. Los problemas de clasificación estándar se dividieron en los que presentan ruido y los que contienen valores perdidos (27 en total).

Se emplearon 15 datasets con 5 % de ruido para probar el comportamiento de JRip con respecto a los otros de su familia antes mencionados. En el Cuadro 2 se muestran los nombres de los datasets, el número de atributos (# atributos), ejemplos (# ejemplos) y clases (# clases) que contienen. Los atributos y las clases contienen valores nominales y numéricos.

En el Cuadro 2 se muestran los 12 datasets con valores perdidos usados en la experimentación. De ellos se muestra una descripción resumen con los nombres, número de atributos (# atributos), ejemplos (# ejemplos) y clases (# clases) que tienen, así como el porcentaje de valores perdidos con respecto al total de ejemplos del datasets (% VP).

Para evaluar el comportamiento de JRip y de los otros algoritmos basados en reglas en problemas desbalanceados se usó como alternativa a la *precisión de la clasificación* la métrica *Kappa*. A diferencia de la precisión de la clasificación Kappa anota los éxitos en la clasificación independientemente para cada clase. En la precisión de la clasificación se tiende a pre-juiciar por la clase mayoritaria, es decir, la que mayor cantidad de ejemplos contiene. Este aspecto es muy importante para la toma de decisiones fundamentalmente en los problemas de diagnósticos médicos, detección de fraudes y otros donde una clasificación errónea tiene gran impacto.

Los datasets desbalanceados usados en este trabajo se dividen en dos grupos: 15 con el ratio o proporción de desbalance (RD) mayor que 9 y clases binarias (Ver Cuadro 3), y 13 datasets con diversos RD pero con múltiples clases (Ver Cuadro 3). En total se emplearon 28 datasets desbalanceados.

²<http://www.keel.es>

Cuadro 1. Propiedades los algoritmos JRip, Ridor, DecisionTable y OneR.

	JRip	Ridor	OneR	DecisionTable
Clase	Nominal, valores perdidos, binaria			Fecha, nominal, valores perdidos, numérica, binaria.
Atributos	Unarios, nominales, numéricos, valores perdidos, fechas, nominales vacíos.			

Cuadro 2. Descripción resumen de los datasets con ruido (1) y valores perdidos (2).

(1)	# Atribs	# Ejems	#Clas	(2)	# Atr	# Ej	#Cla	% VP
contraceptive-5an-nn	9	1473	3	bands	19	539	2	32.28
ecoli-5an-nn	7	336	8	cleveland	13	303	5	1.98
glass-5an-nn	9	214	7	dermatology	34	366	6	2.19
heart-5an-nn	13	270	2	ecoli+MV-10-1tra	7	336	8	48.21
ionosphere-5an-nn	33	351	2	hepatitis	19	155	2	48.39
iris-5an-nn	4	150	3	horse-colic	23	368	2	98.1
page-blocks-5an-nn	10	5472	5	housevotes	16	435	2	46.67
pima-5an-nn	8	768	2	magic+MV-10-1tra	10	1902	2	58.20
satimage-5an-nn	36	6435	7	mammographic	5	961	2	13.63
segment-5an-nn	19	2310	7	marketing	13	8993	9	23.54
sonar-5an-nn	60	208	2	newthyroid+MV-10-1tra	5	215	3	35.35
spambase-5an-nn	57	4597	2	post-operative	8	90	3	3.33
thyroid-5an-nn	21	7200	3					
wdbc-5an-nn	30	569	2					
wine-5an-nn	13	178	3					

Cuadro 3. Descripción resumen de los datasets desbalanceados con $RD > 9$ y cantidad de clases 2 (1) y datasets con múltiples clases (2).

(1)	# Atribs	# Ejems	RD	(2)	# Atribs	# Ejems	RD
abalone9-18	8	731	16.4	balance	4	625	5.88
cleveland-0_vs_4	13	177	12.62	contraceptive	9	1473	1.89
dermatology-6	34	358	16.9	ecoli	7	336	71.25
ecoli-0-1-3-7_vs_2-6	7	281	39.14	glass	9	214	8.44
ecoli-0-1-4-6_vs_5	6	280	13	hayes-roth	4	132	1.7
ecoli4	7	336	15.8	lymph	19	148	40.5
glass2	9	214	11.59	lymphography	18	148	40.5
glass5	9	214	22.78	new-thyroid	5	215	4.84
lymphography-normal-fibrosis	18	148	23.67	pageblocks	10	548	164
poker-9_vs_7	10	244	29.25	shuttle	9	2175	853
shuttle-6_vs_2-3	9	230	22	thyroid	21	720	36.94
shuttle-c2-vs-c4	9	129	20.5	wine	13	178	1.5
yeast-1_vs_7	7	281	39.14	zoo	18	101	10.25
yeast-2_vs_4	8	514	9.08				
zoo-3	16	101	19.2				

Los parámetros del algoritmo JRip así como de los otros algoritmos de su familia empleados son los definidos en la herramienta Weka por defecto. Para JRip se definen los parámetros: Folds 3, minNo 2.0, optimizations 2 y seed 1.

Para realizar el análisis estadístico y la presentación de los resultados se sigue la metodología propuesta por Demsar[3]. En esta metodología se propone en primer lugar aplicar el test de Friedman para conocer si al menos uno de los algoritmos se comporta diferentes a los demás, es decir, tiene diferencias significativas con respecto al resto. El test de Friedman se usa cuando son cinco o menos los métodos comparados, en otro caso se usa una corrección a Friedman, el test Iman y Davemport. Se puede comprobar que existen diferencias significativas si el valor de probabilidad o p-value obtenido es menor que un α , en este caso se usa $\alpha = 0,05$.

Si el resultado del test da que $\alpha < 0,05$ entonces se aplica un test post-hoc para conocer cuál algoritmo en particular se comporta mejor que el resto. En este trabajo se usa el test post-hoc Finner, el cual se usa cuando se quiere comparar un algoritmo contra el resto [6]. En este trabajo se usa además a partir del test post-hoc aplicado el ranking promedio de Friedman para mostrar gráficamente cuán bueno es un método con respecto a los otros. El ranking es obtenido a partir de asignar una posición a cada algoritmo dependiendo de su desempeño para cada dataset. El algoritmo que alcanza el mejor desempeño en un datasets específico obtiene el primer ranking (valor 1), luego el que alcanza el segundo mejor desempeño obtiene el lugar 2, y así sucesivamente. Finalmente los valores obtenidos en el ranking por cada algoritmo para cada dataset se suman y se promedian.

Los tests estadísticos aplicados para evaluar el algoritmo JRip se realizaron con la biblioteca para el análisis estadístico SC-MAMP (Statistical Comparison of Multiple Algorithms in Multiple Problems) del lenguaje de programación R.

3. Resultados y discusión

3.1 Problemas de clasificación estándar

En esta sección se presentan los resultados de la evaluación del desempeño del clasificador JRip con datos estándar, es decir, datos que no presentan desbalance o este es mínimo. Sin embargo, los datasets estándar que se presentan sí tienen problemas de ruido y otros contienen valores perdidos.

Para evaluar la precisión de la clasificación de JRip y de los otros tres clasificadores de su familia Ridor, DecisionTable y OneR se realizó una validación cruzada en la herramienta de minería de datos Weka. Se seleccionaron 10 fold para la validación cruzada para todos los algoritmos evaluados. En el Cuadro 7 se muestran los resultados obtenidos de la evaluación experimental realizada.

A partir de los resultados experimentales obtenidos en la validación cruzada se aplicó el test de Friedman para verificar si existen diferencias significativas entre JRip y al menos otros de los clasificadores. El test de Friedman arrojó como resultado el valor 0.2535. Este resultado valida que no existen

diferencias significativas en el desempeño de los algoritmos basados en reglas comparados para los datasets con valores perdidos usados. Sin embargo en el ranking de Friedman, el clasificador JRip es competitivo con el resto para cuando los datasets tienen valores perdidos como se muestra en el grafo de la Figura 1a. En los nodos del grafo se muestran los resultados obtenidos a partir de aplicar el test post-hoc Finner. El número indica el ranking de Friedman en el que se encuentra cada algoritmo basado en regla comparado. JRip se encuentra sobresaltado en color verde porque para los datos usados obtuvo un mejor desempeño y por lo tanto tiene un ranking menor que el resto.

Cuadro 4. Resultados experimentales sobre datos con valores perdidos.

Dataset	JRip	Ridor	DecisionTable	OneR
bands	67.34	65.50	63.45	66.98
cleveland	54.46	54.76	56.09	52.83
dermatology	92.09	94.82	90.98	49.73 •
ecoli+MV	75.17	76.85	68.85 •	60.95 •
hepatitis	74.75	77.96	74.88	80.00
horse-colic	85.57	85.02	82.60	81.52
housevotes	95.41	94.95	93.35	95.64
magic+MV	78.43	75.86	76.04 •	69.43 •
mammographic	82.62	81.58	82.21	81.89
marketing	22.76	28.20 ○	32.47 ○	28.27 ○
newthyroid+MV	90.11	89.68	90.71	85.95
post-operative	71.11	71.11	68.89	67.78

○, • mejora estadísticamente significativa o empeoramiento

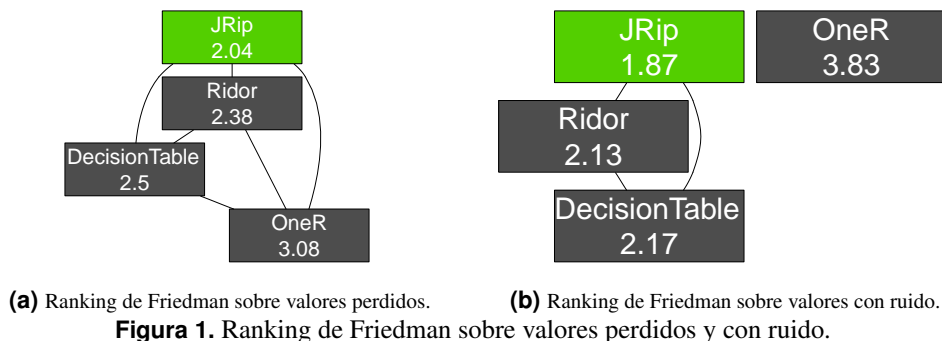
El desempeño del clasificador JRip también fue evaluado con datos ruidosos. Se realizó igualmente una validación cruzada con 10 fold para medir la calidad de la clasificación de este algoritmo con respecto a Ridor, DecisionTable y OneR. Los resultados experimentales sobre datos con ruido se muestran en el Cuadro 8.

Cuadro 5. Resultados experimentales sobre datos con ruido.

Dataset	JRip	Ridor	DecisionTable	OneR
contraceptive	50.78	51.26	51.80	46.77 •
ecoli	74.69	72.33	72.64	57.99 •
glass	58.87	67.77	59.31	52.79
heart	77.41	74.81	81.11	64.81 •
ionosphere	86.31	86.87	87.44	77.51 •
iris	88.00	90.67	92.67	92.67
page-blocks	96.07	95.71	94.88 •	93.18 •
pima	73.44	74.22	74.36	72.66
satimage	85.50	84.26	78.51 •	57.93 •
segment	90.43	91.21	82.12 •	62.25 •
sonar	72.05	70.05	71.60	66.40
spambase	90.21	89.28	85.19 •	77.68 •
thyroid	98.36	98.03	97.44 •	94.43 •
wdbc	92.79	92.09	90.85	84.88 •
wine	93.30	89.35	84.18	78.66 •

○, • mejora estadísticamente significativa o empeoramiento

A partir de los resultados experimentales se aplicó el test de Friedman. En este caso el p-value o valor de probabilidad es 7.111×10^{-5} , valor inferior a 0.05. Se comprobó por tanto que existen diferencias significativas entre JRip y el clasificador



OneR. En el Cuadro 6a se muestran los p-valores obtenidos a partir del test de Friedman.

En la Figura 1b se muestra el grafo con los resultados del ranking de Friedman luego de aplicar el test post-hoc Finner. Se puede observar que JRip obtiene en esta situación resultados significativamente mejores que OneR y es competitivo con el resto. El clasificador OneR se encuentra aislado en el grafo pues tiene un desempeño y por consiguiente un lugar en el ranking significativamente peor para datos con ruido que el resto.

3.2 Problemas de clasificación desbalanceados

En el epígrafe anterior se evaluó el desempeño el algoritmo JRip en problemas de clasificación con datos ruidosos y valores perdidos. En este epígrafe se presentan los resultados experimentales del comportamiento de este clasificador con respecto a los otros de su familia cuando los datasets están desbalanceados. En el Cuadro 10 se muestran los resultados experimentales cuando el RD es superior a 9 y las clases son binarias. En el Cuadro 11 los resultados se basan en datos desbalanceados con múltiples clases y diferentes valores de RD.

La métrica empleada para evaluar los algoritmos para datos desbalanceados es Kappa. En esta métrica los valores cercanos a 1 son los deseados pues indican un desempeño significativamente bueno, mientras que los cercanos a -1 lo contrario.

Los resultados del test de Friedman para el primer tipo de problema desbalanceado que se presenta arrojó 0.1411. A partir de este resultado se comprueba que no existen diferencias significativas entre los algoritmos comparados, todos se comportan de manera similar en este caso en particular. Sin embargo, el algoritmo JRip obtuvo primer lugar en el ranking de Friedman con respecto a los otros basados en reglas comparados como se muestra en el grafo de la figura 2a.

En problemas de desbalance cuando existen múltiples clases, los clasificadores basados en reglas analizados si presentan diferencias significativas ya que el valor de probabilidad a partir del test de Friedman arrojó $9.917e-05$. En el Cuadro 6b se muestra que al menos uno de los algoritmos es significativamente diferente a JRip. El test post-hoc Finner reveló que el comportamiento de JRip es significativamente mejor que el de OneR. Por el contrario, Ridor y DecisionTable presentan com-

portamientos similares a JRip. En el grafo de la Figura 2b se aprecia que Ridor y DecisionTable están relacionados a JRip al tener comportamiento similares en este tipo de desbalance. Por el contrario, el algoritmo OneR se encuentra distante de JRip y solo se conecta con DecisionTable. Los valores del ranking promedio de Friedman muestran que JRip se encuentra en primer lugar con 1,71.

Conclusiones

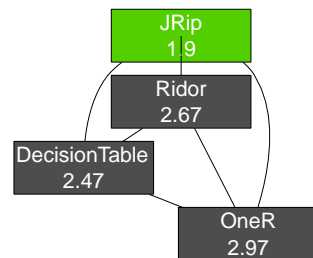
En el presente estudio se evaluó el desempeño del algoritmo basado en reglas JRip para problemas de clasificación con valores ruidosos, valores perdidos y desbalanceados. Se emplearon como métricas la precisión de la clasificación para datos estándar y Kappa para datos desbalanceados. A partir de las evaluaciones experimentales y del test de Friedman y Finner se obtuvieron conclusiones sobre el comportamiento de JRip con respecto a Ridor, DecisionTable y OneR. En problemas de clasificación con datos perdidos el algoritmo JRip se comportó de manera similar al resto de los algoritmos, en el ranking de Friedman obtuvo el primer lugar lo cual lo hace competitivo en estos tipos de problemas. En los problemas de clasificación con datos ruidosos JRip se desempeñó significativamente mejor que OneR y de forma similar al resto.

En situaciones de desbalance cuando solo existen dos clases y el RD es mayor que 9 no se presentaron diferencias entre los algoritmos. Sin embargo, JRip obtuvo el primer lugar en el ranking para este tipo de problema de desbalance. Los test aplicados sobre los resultados experimentales sobre datos desbalanceados con múltiples clases sí mostraron diferencias significativas. El clasificador JRip se comportó significativamente mejor que OneR al igual que en los problemas con ruido. Sin embargo, DecisionTable y OneR tienen un comportamiento similar en este caso en particular.

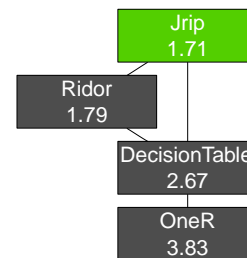
Se concluye por tanto que JRip es significativamente mejor que los otros de su familia comparados cuando existen problemas de clasificación con ruido y desbalance con múltiples clases. En las otras situaciones se recomienda emplear cualquiera de los otros clasificadores comparados menos OneR,

Cuadro 6. Valores p-value en problemas de clasificación con ruido y desbalanceados.

(a) Problemas de clasificación con ruido.				(b) Problemas de clasificación desbalanceados con múltiples clases.			
JRip	Ridor	DecisionTable	OneR	JRip	Ridor	DecisionTable	OneR
n/a	0.672	0.672	0.000	n/a	0.874	0.102	0.000



(a) RD>9 y clases binarias.



(b) Desbalance con múltiples clases.

Figura 2. Ranking de Friedman sobre datos desbalanceados.**Cuadro 7.** Resultados experimentales sobre datos con RD mayor que 9 y clases binarias.

Dataset	JRip	Ridor	DecisionTable	OneR
abalone9-18	0.36	0.16	0.16	
cleveland-0-vs-4	0.38	0.08	0.24	
dermatology-6	0.98	0.98	0.98	
ecoli-0-1-3-7-vs-2-6	0.66	0.47	0.66	
ecoli-0-1-4-6-vs-5	0.56	0.59	0.54	
ecoli4	0.58	0.61	0.58	
glass2	-0.01	-0.02	-0.01	
glass5	0.69	0.44	0.09 •	
lymphography-normal-fibrosis	0.50	0.50	0.40	
poker-9-vs-7	0.39	0.10	0.20	
shuttle-6-vs-2-3	0.90	0.96	1.00	
shuttle-c2-vs-c4	0.90	0.90	0.90	
yeast-1-vs-7	0.31	0.26	0.27	
yeast-2-vs-4	0.65	0.65	0.72	
zoo-3	0.60	0.56	0.50	

○, • mejora estadísticamente significativa o empeoramiento

Cuadro 8. Resultados experimentales sobre datos desbalanceados y múltiples clases

Dataset	JRip	Ridor	DecisionTable	OneR
0.14 balance	0.65	0.62	0.52 •	0.23 •
0.13 contraceptive	0.25	0.28	0.28	0.14 •
0.94 Ecoli-Bal	0.73	0.74	0.67	0.53 •
0.29 Glass-Identification-Database	0.52	0.53	0.52	0.33 •
0.63 Hayes-Roth	0.74	0.66	0.33 •	0.02 •
0.62 lymphography	0.54	0.59	0.55	0.52
-0.02 New-Thyroid-Bal	0.84	0.84	0.81	0.79
0.09 pageblocks	0.77	0.70	0.72	0.73
0.10 shuttle	0.99	0.99	0.98	0.88 •
0.20 thyroid	0.89	0.84	0.85	0.72 •
1.00 wine	0.90	0.84	0.80	0.68 •
1.00 zoo	0.83	0.92	0.81	0.04 •

○, • mejora estadísticamente significativa o empeoramiento

que en todos los experimentos obtuvo resultados inferiores al resto.

Agradecimientos

A la profesora Drc. Maria Matilde García Lorenzo del Centro de Estudios de Informática de la Universidad de Las Villas por sus consejos para el desarrollo de esta investigación.

Referencias

- [1] Rafael Esteban Bello Perez, Zoila Zenaida Garcia Valdivia, Maria Matilde Garcia Lorenzo, and Antonio Reynoso Lobato. *Aplicaciones de la Inteligencia Artificial*. México, segunda edition, 2002. (document), 2
- [2] William W. Cohen. Fast Effective Rule Induction. pages 115–123, 1995. 1
- [3] Janez Demsar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, (7):1–30, 2006. 2
- [4] Manuel Fernández-Delgado, Eva Cernadas, Senén Barro, and Dinami Amorim. Do we Need Hundreds of Classifiers to Solve Real World Classification Problems? *Journal of Machine Learning Research*, 15:3133–3181, 2014. (document)
- [5] Brian R. Gaines and Paul Compton. Induction of Ripple-Down Rules Applied to Modeling Large Databases. *Journal of Intelligent Information System*, 5(3):211–228, 1995. 1
- [6] Salvador García, A. Fernández, J. Luengo, and Francisco Herrera. Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: Experimental analysis of power. *Information Sciences*, 180(10):2044–2064, 2010. 2

- [7] R. C. Holte. Very simple classification rules perform well on most commonly used datasets. *Machine Learning*, 11:63–91, 1993. 3
- [8] Ron Kohavi. The Power of Decision Tables. pages 174–189, 1995. 2
- [9] Carlos Márquez Vera, Cristóbal Romero Morales, and Sebastián Ventura Soto. Predicción del Fracaso Escolar mediante Técnicas de Minería de Datos. *IEEE-RITA*, 7(3):109–117, 2012. (document)
- [10] Dánel Sánchez Tarragó. Pronóstico de supervivencia de infarto cerebral aterotrombótico usando aprendizaje automatizado. pages 1–11. (document)
- [11] V. Veeralakshmi and D. Ramyachitra. Ripple Down Rule learner (RIDOR) Classifier for IRIS Dataset. *International Journal of Computer Science Engineering*, 4(3), 2015. (document)
- [12] Ronald Velandia Ortega and Fredy Hernández Suárez. *Evaluación de algoritmos de extracción de reglas de decisión para el diagnóstico de huecos de tensión*. Ingeniero electricista, Universidad industrial de Santander, Bucaramanga, 2010. (document)
- [13] W. Nor Haizan W. Mohamed, Mohd Najib Mohd Salleh, and Abdul Halim Omar. A Comparative Study of Reduced Error Pruning Method in Decision Tree Algorithms. Penang, Malaysia, 2012. (document)
- [14] Ian H. Witten, Eibe Frank, and Mark A. Hall. *Data Mining. Practical Machine Learning Tools and Techniques*. (document)