

Universidad de Huelva

Departamento de Tecnologías de la Información



Una propuesta determinista para la obtención de reglas en problemas de minería de datos

Memoria para optar al grado de doctor
presentada por:

Juan Luis Domínguez Olmedo

Fecha de lectura: 15 de enero de 2019

Bajo la dirección del doctor:

Jacinto Mata Vázquez

Huelva, 2019



UNIVERSIDAD DE HUELVA



**Universidad
de Huelva**

***Una Propuesta Determinista para la
Obtención de Reglas en Problemas de Minería de Datos***

Tesis Doctoral

Juan Luis Domínguez Olmedo

2018

UNIVERSIDAD DE HUELVA



Universidad
de Huelva

Departamento de Tecnologías de la Información

***Una Propuesta Determinista para la
Obtención de Reglas en Problemas de Minería de Datos***

Programa de Doctorado: Ciencia y Tecnología Industrial y Ambiental

Tesis Doctoral

Director: **Dr. D. Jacinto Mata Vázquez**

Doctorando: **Juan Luis Domínguez Olmedo**

2018

Tesis Doctoral parcialmente subvencionada por la Consejería de Conocimiento, Investigación y Universidad de la Junta de Andalucía, con el proyecto TIC-7629.



CONSEJERÍA DE CONOCIMIENTO,
INVESTIGACIÓN Y UNIVERSIDAD

TIC-7629

AGRADECIMIENTOS

Agradezco a mi director el haberme dado la oportunidad de investigar con cierta libertad las ideas expuestas en esta tesis.

Dedico esta tesis a quienes no pude prestar más atención por encontrarme absorto analizando datos en una pantalla.

Especial dedicatoria a los que ya no están aquí y no pueden leer estas páginas.

Índice

1. Introducción	1
1.1. Reglas de asociación	3
1.1.1. Discretización de atributos numéricos	6
1.2. Descubrimiento de subgrupos	7
1.3. Clasificación	10
2. Objetivos y metodología	13
2.1. Objetivos	13
2.2. Tests estadísticos para evaluar los resultados	14
2.3. Validación cruzada	15
3. Un método determinista para la búsqueda de reglas	17
3.1. Reglas de asociación numéricas (DEQAR)	17
3.1.1. Descripción del método	17
3.1.2. Reducción del número de reglas	23
3.1.3. Algoritmo DEQAR	25
3.1.4. Resultados experimentales	34
3.1.4.1. Comparativa con Apriori	34
3.1.4.2. Aplicación al conjunto de datos <i>Easterly and Levine</i>	40
3.1.4.3. Aplicación al conjunto de datos <i>STULONG</i>	43
3.2. Descubrimiento de subgrupos (DEQAR-SD)	46
3.2.1. Descripción del método	46
3.2.2. Comparativa con métodos deterministas	48
3.2.3. Comparativa con métodos no deterministas	54
3.3. Clasificación (DEQAR-CLAS)	60
3.3.1. Descripción del método	60
3.3.2. Aplicación a un conjunto de datos médicos	61
4. Conclusiones	69
4.1. Resultados obtenidos	69
4.1.1. Reglas de asociación	69
4.1.2. Descubrimiento de subgrupos	71
4.1.3. Clasificación	72
4.2. Publicaciones asociadas al trabajo desarrollado	73
4.3. Trabajos futuros	74

Apéndice 77

Bibliografía 101

Índice de figuras

Figura 1.1. Esquema de las etapas de un proceso de KDD	2
Figura 1.2. Ejemplos de discretización en igual-anchura e igual-frecuencia	6
Figura 2.1. Proceso de validación cruzada de orden 10	16
Figura 3.1. Ejemplo de creación de las tablas auxiliares	19
Figura 3.2. Ejemplo de búsqueda de intervalos	25
Figura 3.3. Ejemplo de búsqueda de intervalos	25
Figura 3.4. Resultados de ejecución para el conjunto de datos <i>abalone</i> , para varios valores del parámetro <i>delta</i>	28
Figura 3.5. Resultados de ejecución para el conjunto de datos <i>ecoli</i> , para varios valores del parámetro <i>delta</i>	28
Figura 3.6. Resultados de ejecución para el conjunto de datos <i>forestfires</i> , para varios valores del parámetro <i>delta</i>	29
Figura 3.7. Resultados de ejecución para el conjunto de datos <i>housing</i> , para varios valores del parámetro <i>delta</i>	29
Figura 3.8. Resultados de ejecución para el conjunto de datos <i>wine</i> , para varios valores del parámetro <i>delta</i>	30
Figura 3.9. Resultados de ejecución para el conjunto de datos <i>yeast</i> , para varios valores del parámetro <i>delta</i>	30
Figura 3.10. Resultados de ejecución para el conjunto de datos <i>abalone</i> , con distintos valores del parámetro <i>maxAtrA</i>	31
Figura 3.11. Resultados de ejecución para el conjunto de datos <i>ecoli</i> , con distintos valores del parámetro <i>maxAtrA</i>	31
Figura 3.12. Resultados de ejecución para el conjunto de datos <i>housing</i> , con distintos valores del parámetro <i>maxAtrA</i>	32
Figura 3.13. Resultados de ejecución para el conjunto de datos <i>wine</i> , con distintos valores del parámetro <i>maxAtrA</i>	32
Figura 3.14. Valores de confianza obtenidos para el conjunto de datos <i>abalone</i> , variando los parámetros <i>maxAtrA</i> y <i>delta</i>	33
Figura 3.15. Tiempos de ejecución (escala logarítmica) para el conjunto de datos <i>abalone</i> , variando los parámetros <i>maxAtrA</i> y <i>delta</i>	34
Figura 3.16. Resultados para el conjunto de datos <i>abalone</i> , con distintos valores del parámetro <i>delta</i> , empleando condiciones con intervalos cerrados (<i>INT</i>) y sin ellos (<i>NOINT</i>)	46
Figura 3.17. Resultados para el conjunto de datos <i>abalone</i> , con distintos valores del parámetro <i>maxAtrA</i> , empleando condiciones con intervalos cerrados (<i>INT</i>) y sin ellos (<i>NOINT</i>)	47

Figura 3.18. Proceso de entrenamiento en DEQAR-CLAS 61
Figura 3.19. Uno de los árboles resultantes en el clasificador RandomForest ... 67

Índice de tablas

Tabla 3.1. Ejemplo de posibles secuencias de intervalos	20
Tabla 3.2. Reglas con los mismos valores de soporte y confianza pero distinto valor de lift	21
Tabla 3.3. Algunas de las reglas generadas con $minsup = 0.4$ y $minconf = 0.7$ para el conjunto de datos de ejemplo de la Figura 3.1	22
Tabla 3.4. Doce primeras reglas del ranking final correspondiente al conjunto de datos de ejemplo de la Figura 3.1, según una ordenación por confianza-soporte	23
Tabla 3.5. Conjuntos de datos usados para evaluar la influencia de los parámetros	27
Tabla 3.6. Características de los conjuntos de datos usados en los experimentos	35
Tabla 3.7. Valores medios de ranking (mejores 100 reglas)	38
Tabla 3.8. Valores medios de ranking (mejores 25 reglas)	38
Tabla 3.9. p-valores de los tests realizados	39
Tabla 3.10. Subconjuntos de métodos según los tests estadísticos (mejores 100 reglas)	39
Tabla 3.11. Subconjuntos de métodos según los tests estadísticos (mejores 25 reglas)	40
Tabla 3.12. Atributos de un conjunto de datos sobre países del África subsahariana en el período 1960-1980	41
Tabla 3.13. Algunas de las mejores reglas encontradas para el conjunto de datos <i>Easterly and Levine</i>	42
Tabla 3.14. Algunas de las mejores reglas encontradas que involucran la variable <i>WAR</i>	42
Tabla 3.15. Atributos seleccionados del estudio <i>STULONG</i>	43
Tabla 3.16. Una selección de reglas asociando los grupos de atributos "Examen físico" y "Examen bioquímico"	44
Tabla 3.17. Una selección de reglas obtenidas asociando el atributo <i>DEATH?</i>	45
Tabla 3.18. Características de los conjuntos de datos usados en los experimentos	49
Tabla 3.19. Resultados obtenidos para la medida cobertura	50
Tabla 3.20. Resultados obtenidos para la medida relevancia	51
Tabla 3.21. Resultados obtenidos para la medida número de reglas	51
Tabla 3.22. Resultados obtenidos para la medida número de atributos	52

Tabla 3.23. Valores medios de ranking obtenidos por cada método en cada medida de calidad	53
Tabla 3.24. p-valores de los test realizados	53
Tabla 3.25. p-valores ajustados de los tests post-hoc	54
Tabla 3.26. Características de los conjuntos de datos usados en los experimentos	55
Tabla 3.27. Resultados para la medida atipicidad	55
Tabla 3.28. Resultados para la medida sensibilidad	56
Tabla 3.29. Resultados para la medida confianza	56
Tabla 3.30. Valores medios de ranking de los tests de Friedman	57
Tabla 3.31. p-valores de los tests realizados	57
Tabla 3.32. p-valores ajustados del test de Holm para la atipicidad	57
Tabla 3.33. p-valores ajustados del test de Holm para la sensibilidad	58
Tabla 3.34. p-valores ajustados del test de Holm para la confianza	58
Tabla 3.35. Resultados del test de Wilcoxon para DEQAR-SD vs. FuGePSD	58
Tabla 3.36. Atributos del conjunto de datos	63
Tabla 3.37. Resultados en varias medidas de evaluación	64
Tabla 3.38. Resultados para la clase "+"	65
Tabla 3.39. Resultados para la clase "-"	65
Tabla 3.40. Resultados para varios valores del parámetro <i>tamañoRanking</i>	66
Tabla 3.41. Reglas en el clasificador final	66
Tabla A.1. Resultados para el conjunto de datos <i>abalone</i> (mejores 100 reglas)	77
Tabla A.2. Resultados para el conjunto de datos <i>bodyfat</i> (mejores 100 reglas)	77
Tabla A.3. Resultados para el conjunto de datos <i>bolts</i> (mejores 100 reglas)	78
Tabla A.4. Resultados para el conjunto de datos <i>ecoli</i> (mejores 100 reglas)	78
Tabla A.5. Resultados para el conjunto de datos <i>elevators</i> (mejores 100 reglas)	79
Tabla A.6. Resultados para el conjunto de datos <i>forest_fires</i> (mejores 100 reglas)	79
Tabla A.7. Resultados para el conjunto de datos <i>house_16h</i> (mejores 100 reglas)	80
Tabla A.8. Resultados para el conjunto de datos <i>housing</i> (mejores 100 reglas)	80
Tabla A.9. Resultados para el conjunto de datos <i>longley</i> (mejores 100 reglas)	81

Tabla A.10. Resultados para el conjunto de datos <i>mortgage</i> (mejores 100 reglas)	81
Tabla A.11. Resultados para el conjunto de datos <i>pollution</i> (mejores 100 reglas)	82
Tabla A.12. Resultados para el conjunto de datos <i>school</i> (mejores 100 reglas)	82
Tabla A.13. Resultados para el conjunto de datos <i>stock_prices</i> (mejores 100 reglas)	83
Tabla A.14. Resultados para el conjunto de datos <i>veteran</i> (mejores 100 reglas)	83
Tabla A.15. Resultados para el conjunto de datos <i>wine</i> (mejores 100 reglas)	84
Tabla A.16. Resultados para el conjunto de datos <i>yeast</i> (mejores 100 reglas)	84
Tabla A.17. Resultados para el conjunto de datos <i>abalone</i> (mejores 25 reglas)	85
Tabla A.18. Resultados para el conjunto de datos <i>bodyfat</i> (mejores 25 reglas)	85
Tabla A.19. Resultados para el conjunto de datos <i>bolts</i> (mejores 25 reglas)	86
Tabla A.20. Resultados para el conjunto de datos <i>ecoli</i> (mejores 25 reglas)	86
Tabla A.21. Resultados para el conjunto de datos <i>elevators</i> (mejores 25 reglas)	87
Tabla A.22. Resultados para el conjunto de datos <i>forest_fires</i> (mejores 25 reglas)	87
Tabla A.23. Resultados para el conjunto de datos <i>house_16h</i> (mejores 25 reglas)	88
Tabla A.24. Resultados para el conjunto de datos <i>housing</i> (mejores 25 reglas)	88
Tabla A.25. Resultados para el conjunto de datos <i>longley</i> (mejores 25 reglas)	89
Tabla A.26. Resultados para el conjunto de datos <i>mortgage</i> (mejores 25 reglas)	89
Tabla A.27. Resultados para el conjunto de datos <i>pollution</i> (mejores 25 reglas)	90
Tabla A.28. Resultados para el conjunto de datos <i>school</i> (mejores 25 reglas)	90
Tabla A.29. Resultados para el conjunto de datos <i>stock_prices</i> (mejores 25 reglas)	91

Tabla A.30. Resultados para el conjunto de datos <i>veteran</i> (mejores 25 reglas)	91
Tabla A.31. Resultados para el conjunto de datos <i>wine</i> (mejores 25 reglas)	92
Tabla A.32. Resultados para el conjunto de datos <i>yeast</i> (mejores 25 reglas)	92
Tabla A.33. p-valores ajustados del test de Holm (conf-sop-lift, mejores 100 reglas)	93
Tabla A.34. p-valores ajustados del test de Holm (conf-cob-lift, mejores 100 reglas)	94
Tabla A.35. p-valores ajustados del test de Holm (cobertura, mejores 100 reglas)	95
Tabla A.36. p-valores ajustados del test de Holm (atributos, mejores 100 reglas)	96
Tabla A.37. p-valores ajustados del test de Holm (conf-sop-lift, mejores 25 reglas)	97
Tabla A.38. p-valores ajustados del test de Holm (conf-cob-lift, mejores 25 reglas)	98
Tabla A.39. p-valores ajustados del test de Holm (cobertura, mejores 25 reglas)	99
Tabla A.40. p-valores ajustados del test de Holm (atributos, mejores 25 reglas)	100

1. Introducción

Durante el transcurso de las últimas décadas, el volumen de datos almacenados ha crecido considerablemente debido al tratamiento digital de la información obtenida de múltiples fuentes. El uso de estos datos para obtener información que facilite el rendimiento empresarial, o que aporte conocimiento de interés en las organizaciones, ha favorecido la aparición de nuevos métodos automatizados que permiten manejar estos grandes volúmenes de datos. Estos métodos y sus herramientas asociadas pertenecen al ámbito de la extracción de conocimiento a partir de datos (KDD, *Knowledge Discovery from Data*).

Según [Fayyad et al., 1996], "*a nivel abstracto el KDD está relacionado con el desarrollo de métodos y técnicas para dar sentido a los datos. El problema básico abordado por el proceso de KDD es la transformación de datos de bajo nivel (que suelen ser demasiado voluminosos para comprenderlos fácilmente) en otras formas que podrían ser más compactas (por ejemplo, un informe breve), más resumen (por ejemplo, una aproximación descriptiva o modelo del proceso que generó los datos), o más útil (por ejemplo, un modelo predictivo para estimar el valor de casos futuros)*". En el núcleo del proceso estaría la aplicación de métodos específicos de Minería de Datos (DM, *Data Mining*) para el descubrimiento y extracción de patrones.

Según esto, la minería de datos sería una de las etapas del proceso de KDD, que se muestran en forma simplificada en la Figura 1.1. Puede considerarse la minería de datos como "*el proceso de descubrir patrones de interés y conocimiento, a partir de grandes volúmenes de datos*" [Han et al., 2012]. Estos patrones pueden por ejemplo identificar grupos, anomalías o dependencias en los datos, que den lugar a conocimiento de interés para el usuario. Lo interesante que resulten los patrones hallados podría estar en función de alguna medida de interés y un cierto nivel de certeza, fijados de antemano.

Este vertiente de la minería de datos centrada en encontrar patrones de interés, se correspondería con un enfoque *descriptivo*. Por contra, el enfoque *predictivo* estaría orientado a encontrar un modelo (función) que permita con cierto grado de certeza hacer predicciones futuras sobre el valor de un atributo. En ciertos casos, la frontera entre predicción y descripción puede ser algo difusa, ya que algunos de los modelos predictivos pueden ser en parte descriptivos, como se comentará posteriormente.

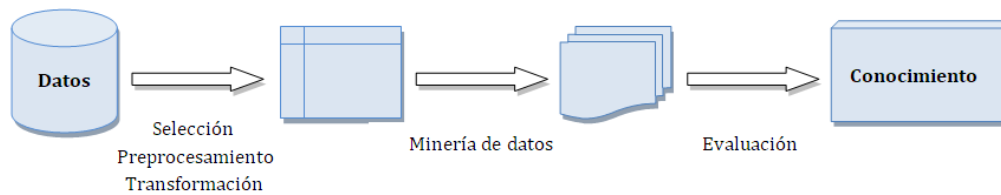


Figura 1.1: Esquema de las etapas de un proceso de KDD.

Los objetivos descriptivos y predictivos de la minería de datos pueden encontrarse en una variedad de tareas. Entre ellas están las siguientes:

- La búsqueda de **reglas de asociación** tiene por objeto encontrar relaciones de interés (asociaciones o dependencias) entre los atributos de un conjunto de datos, con el propósito de obtener información de interés para la toma de decisiones.

Los primeros estudios sobre reglas de asociación se centraron en datos con atributos binarios, principalmente en formato transaccional, para el problema de identificar nuevas oportunidades en la venta de productos [Agrawal et al., 1993] [Agrawal et al., 1994]. Aunque un concepto similar fue introducido y estudiado ya en la década de 1960, dentro del desarrollo del método GUHA [Hájek et al., 1966] [Rauch, 2018].

Las reglas de asociación también se ha aplicado con éxito a otros dominios como la bioinformática, la medicina o el análisis de datos científicos [Carmona-Saez et al., 2006] [Agapito et al., 2015] [Jitkajornwanich et al., 2017] [Rajesh et al., 2018].

- El **agrupamiento** intenta formalizar grupos en los datos, según alguna medida de similitud. En sus orígenes fue empleado en antropología y psicología [Driver et al., 1932] [Zubin, 1938]. Posteriormente ha sido empleado en multitud de ámbitos, entre ellos la medicina [Filipovych et al., 2011] o la bioinformática [Zorita et al., 2015].
- La **clasificación** pretende hallar un modelo partiendo de ciertos datos, que permita predecir el valor de un atributo (la *clase*) en nuevos datos a los que pueda aplicarse dicho modelo. Se trata de un aprendizaje "supervisado" en el que se obtiene un modelo a partir de datos

etiquetados, esto es, en los que se conoce el valor de la clase [Weiss et al., 1991].

- La **regresión** busca un modelo predictivo para asignar valores a un atributo numérico [Dobson, 2001].
- La **detección de anomalías** intenta encontrar observaciones en los datos que no se corresponden con los patrones esperados [Chandola et al., 2009] [Campos et al., 2016].

Se describen a continuación en mayor detalle las tres tareas abordadas en este trabajo: reglas de asociación, descubrimiento de subgrupos y clasificación.

1.1. Reglas de asociación

La obtención de reglas de asociación es una tarea bien estudiada que se centra en buscar asociaciones entre los atributos (variables) de un conjunto de datos. Dichas asociaciones podrían ser de interés al mostrar relaciones desconocidas entre los atributos.

Una regla de asociación toma la forma $A \rightarrow C$, donde A y C , antecedente y consecuente de la regla, son una condición o conjunciones de condiciones sobre atributos del conjunto de datos. Los conjuntos de atributos del antecedente y consecuente deben ser disjuntos. Por ejemplo, la siguiente regla podría representar una posible asociación entre los atributos *edad*, *casado* y *numCoche*s:

$$edad \in [35, 70] \text{ y } casado = 'SI' \rightarrow numCoche = 2$$

En este caso, la condición del atributo numérico *edad* se expresa mediante la pertenencia a un intervalo. También podría emplearse algún operador relacional ("=", ">", "<").

Es importante medir la calidad de las reglas encontradas, por un lado para contar con algún dato objetivo vinculado con cada regla y así poder compararlas; pero no menos importante es emplear dicha medida para poder filtrar y seleccionar las mejores de entre la posible gran cantidad de reglas obtenidas. Se han propuesto un buen número de medidas para evaluar la calidad de las reglas [Lavrač et al., 1999] [An et al., 2001] [Geng et al., 2006] [Guillet et al., 2007] [Lenca et al., 2008]. Por un lado estarían las medidas *objetivas*, derivadas de los propios datos y basadas generalmente en la Estadística o en la Teoría de la

Información; por otra parte, las medidas *subjetivas* tienen en cuenta tanto los datos como el usuario que los examina, y son generalmente usadas en métodos interactivos [Silberschatz et al., 1995] [Silberschatz et al., 1996] [Liu et al., 1999]. Entre las más importantes medidas objetivas figuran las siguientes:

- **soporte (*support*):** mide la frecuencia con que se cumple una condición o conjunto de condiciones. Para una regla dada, indica cuánto se cumple dicha regla en el conjunto de datos, mediante la ratio del número de instancias que cumplen el antecedente y el consecuente sobre el total de instancias (N):

$$\text{soporte}(A \rightarrow C) = \text{soporte}(A \cup C) = n_{AC} / N$$

- **confianza (*confidence*):** mide la exactitud con que se cumple una regla, evaluando la ratio de instancias que cumplen el antecedente y el consecuente sobre las que cumplen el antecedente:

$$\text{confianza}(A \rightarrow C) = \text{soporte}(A \cup C) / \text{soporte}(A) = n_{AC} / n_A$$

- **lift:** mide cuánto más se cumplen a la vez antecedente y consecuente de lo que sería de esperar si fueran estadísticamente independientes [Brin et al., 1997a]. Se calcula mediante el cociente entre la confianza de la regla y el soporte del consecuente:

$$\text{lift}(A \rightarrow C) = \text{confianza}(A \rightarrow C) / \text{soporte}(C) = (n_{AC} \cdot n) / (n_A \cdot n_C)$$

- **leverage:** calcula cuánto difiere la ocurrencia simultánea de antecedente y consecuente de lo esperable si fueran independientes [Piatetsky-Shapiro, 1991]:

$$\begin{aligned} \text{leverage}(A \rightarrow C) &= \text{soporte}(A \rightarrow C) - \text{soporte}(A) \cdot \text{soporte}(C) = \\ &= n_{AC}/N - (n_A/N) \cdot (n_C/N) \end{aligned}$$

- **conviction:** introducido como una alternativa a la confianza, mide el grado de implicación de una regla [Brin et al., 1997b]. Presenta el inconveniente de no estar acotado superiormente:

$$\begin{aligned} \text{conviction}(A \rightarrow C) &= (1 - \text{soporte}(C)) / (1 - \text{confianza}(A \rightarrow C)) = \\ &= (1 - n_C/N) / (1 - n_{AC}/n_A) \end{aligned}$$

En la tarea de la búsqueda de reglas de asociación, es habitual el empleo de los parámetros relativos al soporte mínimo (*minsup*) y confianza mínima (*minconf*) que toda regla debe cumplir para ser considerada de interés. En la regla de ejemplo antes mostrada, sus valores de soporte y confianza podrían indicar que "el 80% de las personas casadas y con una edad entre 40 y 70 años, posee 2 coches; "un 15% de las personas están casadas, con una edad entre 40 y 70 años, y poseen 2 coches":

$$\text{edad} \in [40, 70] \text{ y } \text{casado} = \text{'SI'} \rightarrow \text{numCoches} = 2$$

[soporte = 0.15, confianza = 0.80]

Una variedad de métodos se han desarrollado para tratar el problema de la búsqueda de reglas de asociación, principalmente variaciones del algoritmo *Apriori*. Se describen a continuación algunos de ellos:

- **Apriori.** La búsqueda de reglas se basa en la propiedad de *clausura descendente*, según la cual todo subconjunto de un conjunto frecuente de ítems es también frecuente. Se procede inicialmente generando los conjuntos frecuentes (que satisfagan el nivel de soporte mínimo), y luego se extraen las reglas de asociación relativas a dichos conjuntos frecuentes (y que satisfagan el nivel de confianza mínima). Necesita discretizar previamente los atributos continuos. Puede llevar a obtener una gran cantidad de reglas de asociación a partir de los datos, sobre todo si el nivel de soporte mínimo es bajo, lo que puede resultar inmanejable en términos de tiempo de cómputo y necesidad de memoria [Agrawal et al., 1994] [Govada et al., 2017].
- **Eclat.** Se basa en agrupar los ítems para aproximarse al conjunto de ítems frecuentes maximales, generando luego los ítems frecuentes contenidos en cada grupo. Difiere respecto Apriori, en que si aquel realiza una búsqueda en anchura para determinar conjuntos frecuentes, este la realiza en profundidad. Todo esto acelera generalmente el cálculo del soporte de las reglas [Zaki et al., 1997].
- **FP-growth.** Básicamente, emplea un esquema de eliminación de los ítems que no sean frecuentes, para construir un árbol que ayude a generar los conjuntos frecuentes [Han et al., 2004].

1.1.1. Discretización de atributos numéricos

A la hora de tratar un conjunto de datos que presente algún atributo numérico, algoritmos como Apriori requieren de un procesamiento previo para discretizar dichos atributos, al no poder trabajar directamente con ellos. Dicha discretización consiste generalmente en dividir el rango de cada atributo numérico en ciertos intervalos, y tratar entonces dichos intervalos como valores discretos. Este proceso puede conllevar cierta pérdida de información, ya que las reglas que podrán ser generadas se basan únicamente en las particiones previamente establecidas [Chmielewski et al., 1996]. Y por otro lado, el tamaño de los intervalos puede afectar directamente al soporte y confianza de las reglas; si se emplean intervalos muy pequeños las reglas pueden carecer de suficiente soporte, y si son demasiado grandes, la confianza puede ser muy baja.

Para la tarea de reglas de asociación no pueden aplicarse ciertos métodos de discretización, como los basados en la teoría de la información, puesto que el consecuente puede contener cualquier atributo. En su lugar, se emplean métodos de discretización no supervisada tales como *igual-anchura* (*equal-width*) o *igual-frecuencia* (*equal-frequency*) [Liu et al., 2002]. En estos métodos, un parámetro fija el número de intervalos (*bins*), y cada intervalo se asocia con un valor discreto distinto. En igual-anchura, el rango continuo del atributo se divide en intervalos de igual anchura; en igual-frecuencia, cada valor discretizado contiene el mismo número de valores originales del atributo. En la Figura 1.2 se muestra un ejemplo para un número de intervalos igual a 4.

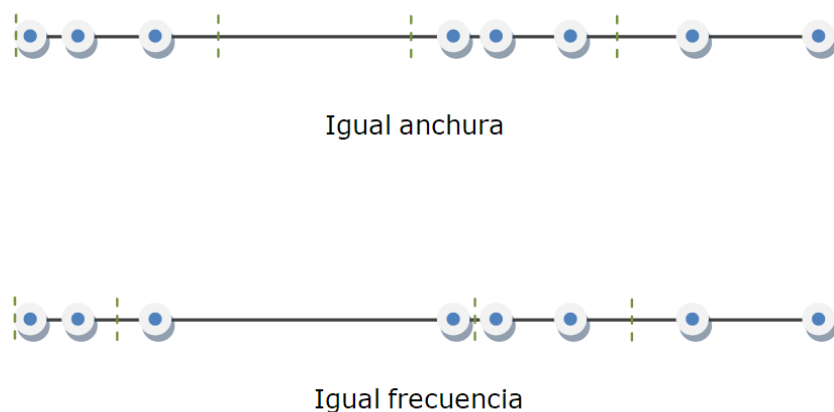


Figura 1.2: Ejemplos de discretización en igual-anchura e igual-frecuencia.

1.2. Descubrimiento de subgrupos

En la confluencia de las técnicas predictivas y descriptivas, el descubrimiento de subgrupos aborda la obtención de reglas descriptivas para un atributo concreto de interés, que pasa a ser el único integrante del consecuente de las reglas [Klößgen, 1996] [Wrobel, 1997]. El resultado deseable sería la obtención de reglas sencillas (comprensibles, con pocos atributos) y altamente significativas.

En esta tarea, sucede también que el tratamiento de los atributos numéricos es relevante porque puede afectar al descubrimiento y calidad de las reglas obtenidas. Generalmente se realiza una fase previa de discretización de dichos atributos, anterior al inicio del proceso de inducción de reglas. Pero el reemplazo de los atributos numéricos por sus correspondientes discretizados puede conducir a resultados sub-óptimos [Grosskreutz et al., 2009].

Se emplean ciertas medidas específicas para evaluar el interés de las reglas y el éxito del descubrimiento de subgrupos. Dichas medidas pueden ser el resultado de calcular un valor medio sobre el conjunto de reglas obtenidas, lo que proporciona una manera de poder comparar diferentes algoritmos. En un conjunto de datos con N instancias, $n(cond)$ representa el número de instancias que satisfacen la condición (o conjunción de condiciones) $cond$. A continuación se detallan algunas medidas de calidad, útiles en la evaluación del descubrimiento de subgrupos:

- **cobertura (*coverage*)**. Mide la frecuencia de instancias cubiertas por el antecedente de la regla:

$$cobertura(A \rightarrow C) = \frac{n(A)}{N}$$

- **relevancia (*significance*)**. Se calcula en función de la razón de verosimilitud de la regla:

$$relevancia(A \rightarrow C) = 2 * \sum_i n(A \wedge C_i) * \log \frac{n(A \wedge C_i)}{n(C_i) * n(A) / N}$$

- **atipicidad (*weighted relative accuracy*)**. Se define como la precisión relativa ponderada de la regla (*WRAcc*):

$$atipicidad(A \rightarrow C) = \frac{n(A)}{N} * \left(\frac{n(A \wedge C)}{n(A)} - \frac{n(C)}{N} \right)$$

Una variedad de métodos se han propuesto para la tarea del descubrimiento de subgrupos. A continuación, se describen algunos de ellos:

- **EXPLORA.** Uno de los primeros métodos desarrollados para esta tarea, emplea árboles de decisión para extraer reglas o patrones a partir de una única tabla de datos. La búsqueda de reglas se basa en espacios de conceptos, empleando un lenguaje proposicional. Utiliza varias medidas de interés de las reglas, como la evidencia, generalidad, redundancia y simplicidad. Puede emplear una búsqueda exhaustiva o heurística [Klösgen, 1996].
- **MIDOS.** Extiende la aplicabilidad de EXPLORA a bases de datos multirelacionales, y también utiliza árboles de decisión. Emplea una poda por soporte mínimo, y evalúa los mejores subgrupos en función del tamaño y la atipicidad [Wrobel, 1997].
- **SD.** Realiza una búsqueda iterativa de reglas en base al conocimiento proporcionado por un experto. Emplea un algoritmo heurístico de búsqueda en haz, y usa como medida de calidad una función dependiente de un *parámetro de generalización* [Gamberger et al., 2002].
- **APRIORI-SD.** Adapta para la tarea de descubrimiento de subgrupos el clasificador APRIORI-C, a su vez una adaptación del algoritmo de reglas de asociación Apriori. Emplea la atipicidad como medida de calidad de las reglas inducidas, así como un esquema de clasificación probabilística para las instancias [Kavšek et al., 2003].
- **CN2-SD.** Es una adaptación del algoritmo de inducción de reglas CN2, modificando el algoritmo de cobertura y la búsqueda heurística. Como medida de calidad utiliza la relación entre la ratio de verdaderos positivos y la ratio de falsos positivos. Permite obtener grupos solapados, lo que permite ampliar el poder descriptivo para el conjunto de subgrupos [Lavrač et al., 2004].
- **SD-MAP.** Es un algoritmo de búsqueda exhaustiva, basado en FP-growth (método de reglas de asociación) y que es adaptado para la tarea de descubrimiento de subgrupos [Atzmueller et al., 2006].
- **SDIGA.** Es un algoritmo evolutivo mono-objetivo que emplea reglas difusas, y que busca reglas para cada uno de los posibles valores del atributo objetivo. Produce reglas de forma que alcancen un valor mínimo de confianza y describiendo subgrupos que tenga instancias no descritas por reglas generadas en iteraciones anteriores. Las reglas son mejoradas en una fase posterior, modificándolas para aumentar su soporte. La

función de ajuste emplea una función de agregación de las medidas cobertura, relevancia, atipicidad, precisión, sensibilidad, soporte y confianza, a elegir por el usuario (entre 1 y 3 de dichas medidas) [del Jesus et al., 2007].

- **NMEEF-SD.** Es un algoritmo evolutivo multi-objetivo, que emplea reglas difusas. Debe ejecutarse tantas veces como valores distintos tenga el atributo de interés, cada una de ellas con el consecuente fijado a uno de dichos valores. Intenta obtener un conjunto de reglas que sea general y preciso, empleando un nivel mínimo de confianza. Permite elegir entre dos y tres medidas de calidad, de entre la cobertura, relevancia, atipicidad, precisión, sensibilidad, soporte y confianza [Carmona et al., 2010].
- **GAR-SD.** Algoritmo evolutivo que busca reglas para cada uno de los posibles valores del atributo de interés. Puede operar con atributos discretos o continuos (mediante el uso de intervalos). Emplea un soporte mínimo en la búsqueda de reglas, y una función de agregación compuesta por el soporte, confianza y relevancia, junto a otras tres medidas relacionadas con los atributos que forman la regla [Pachón et al., 2011].
- **EDER-SD.** Este algoritmo evolutivo mono-objetivo trata de caracterizar la clase minoritaria de un problema. Permite el uso de medidas de calidad tales como la atipicidad, precisión, sensibilidad o relevancia, entre otras [Rodríguez et al., 2012].
- **CGBA-SD.** Este algoritmo evolutivo emplea el paradigma de programación genética para obtener reglas con intervalos en sus atributos continuos. Usa una función de ajuste lineal, calculada respecto al soporte y la confianza. Realiza un filtrado final en base a un valor mínimo de confianza y a la equivalencia entre reglas [Luna et al., 2014].
- **FuGePSD.** Es un algoritmo evolutivo basado en la programación genética y las reglas difusas. Emplea dos funciones de ajuste diferentes, una para optimización local y otra para una optimización global en base a la precisión del conjunto de reglas. La función de ajuste local puede usar una medida de calidad de entre la atipicidad, sensibilidad y confianza difusa [Carmona et al., 2015].

1.3. Clasificación

La tarea de clasificación consiste básicamente en encontrar un modelo predictivo (clasificador) para tratar de predecir con la máxima certeza un atributo de tipo categórico, la *clase*. En el caso más simple, dicho atributo tendría dos posibles categorías, tratándose entonces de un problema de clasificación binaria.

Cuando la distribución de instancias entre las distintas categorías está desequilibrada, se habla de *datos desbalanceados* (*imbalanced data*). En estos casos, la tarea de clasificación resulta generalmente más complicada, puesto que la mayoría de algoritmos no son capaces de obtener buenos resultados para la clase minoritaria, resintiéndose por tanto la precisión global.

Durante las últimas décadas, se han desarrollado una gran variedad de técnicas para resolver problemas de clasificación; por ejemplo, redes neuronales artificiales, árboles de decisión, redes bayesianas o máquinas de soporte vectorial [Cortes et al., 1995] [Hastie et al., 2009]. A continuación, se describen brevemente algunos clasificadores que se han empleado en los estudios comparativos de este trabajo:

- **ZeroR.** Clasifica basándose únicamente en los datos del atributo clase, prediciendo siempre la clase mayoritaria. Puede ser útil para determinar un valor de referencia para la clasificación.
- **Multilayer Perceptron.** Emplea una red neuronal artificial formada por múltiples capas, con retropropagación. Los nodos en esta red son sigmoidales [Rosenblatt, 1958] [Rumelhart et al., 1986] [Zhang et al., 2012].
- **J48.** Utiliza un árbol de decisión C4.5 [Quinlan, 1993]. Un árbol de decisión construye un clasificador mediante una estructura jerárquica, empleando generalmente el concepto de entropía, la cual es una medida de incertidumbre o de desorden, usada para decidir qué atributo debe ser el siguiente en seleccionarse [Breiman et al., 1984] [Quinlan, 1986] [Chen et al., 2016].
- **PART.** Genera una lista de decisión, formada por reglas obtenidas a partir de árboles de decisión, mediante una estrategia "divide y vencerás" [Frank et al., 1998] [Ibarguren et. al, 2016].
- **Random Forest.** Construye un bosque de árboles de decisión aleatorios, mediante un método basado en las técnicas de combinación de modelos [Ho, 1995] [Breiman, 2001].

- **Naive Bayes.** Se basa en el teorema de Bayes, asumiendo que los atributos son independientes. Aunque la independencia generalmente es una suposición pobre, a menudo compite bien con métodos más sofisticados [John et al., 1995] [Rish, 2001].
- **Bayes Net.** Emplea una red de Bayes para representar un conjunto de atributos y sus dependencias condicionales [Pearl, 1985] [Topuz et al., 2018].
- **Logistic.** Construye un modelo de regresión logística multinomial, para predecir las probabilidades de los diferentes resultados posibles de una distribución categórica [Le-Cessie et al., 1992].

En un problema de clasificación binaria, suele hablarse de *casos positivos* y *casos negativos*, en relación con las instancias pertenecientes a las dos categorías de la clase. Se define *TP (True Positive)* como el número de casos positivos correctamente clasificados, *TN (True Negative)* como el número de casos negativos correctamente clasificados, *FN (False Negative)* como el número de casos positivos mal clasificados, y *FP (False Positive)* como el número de casos negativos mal clasificados.

Las medidas empleadas para evaluar la bondad de un clasificador son variadas y, por lo general, se eligen en función del contexto del problema a tratar. En ciertos casos puede ser de interés la tasa de acierto global del clasificador, pero en otros, como en el caso de tratar con datos desbalanceados, puede resultar más significativo el rendimiento en la clasificación de la clase minoritaria. Se exponen a continuación las principales medidas de evaluación empleadas en tareas de clasificación:

- **Tasa de acierto (*accuracy*).** Mide el acierto global en la clasificación, mediante la proporción de aciertos en el total de instancias. Para el caso de una clasificación binaria:

$$\frac{TP + TN}{TP + TN + FN + FP}$$

- **Coeficiente kappa de Cohen.** Mide la concordancia entre dos examinadores en sus correspondientes clasificaciones [Cohen, 1960] [Artstein et al., 2008]:

$$kappa = \frac{tasa\ de\ acierto - acierto\ aleatorio}{1 - acierto\ aleatorio}$$

$$acierto\ aleatorio = \frac{(TN + FP) * (TN + FN) * (TP + FN) * (TP + FP)}{(TP + TN + FN + FP) * (TP + TN + FN + FP)}$$

- **Coefficiente de correlación de Matthews (MCC).** Mide la calidad de una clasificación binaria [Matthews, 1975] [Delgado et al., 2018]:

$$\frac{TP * TN - FP * FN}{\sqrt{(TP + FP) * (TP + FN) * (TN + FP) * (TN + FN)}}$$

- **Sensibilidad (*sensitivity, recall*).** En un problema de clasificación binaria, mide la proporción de casos positivos correctamente clasificados:

$$\frac{TP}{TP + FN}$$

- **Especificidad (*specificity*).** En un problema de clasificación binaria, mide la proporción de casos negativos correctamente clasificados:

$$\frac{TN}{TN + FP}$$

- **Precisión (*precision*).** En un problema de clasificación binaria, mide la proporción de casos clasificados como positivos que efectivamente lo son:

$$\frac{TP}{TP + FP}$$

- **Exhaustividad (*recall*).** En un problema de clasificación binaria, mide la proporción de casos positivos correctamente clasificados:

$$\frac{TP}{TP + FN}$$

- **Medida-F (*F-measure*).** Es la media armónica de la precisión y la exhaustividad:

$$2 * \frac{precisión * exhaustividad}{precisión + exhaustividad}$$

2. Objetivos y metodología

2.1. Objetivos

Como se ha descrito en la sección anterior, la discretización previa de los atributos numéricos, requerida para la aplicación de muchos de los métodos de generación de reglas, conlleva generalmente cierta pérdida de información y puede disminuir la calidad de las reglas obtenidas.

Se ha partido pues de la hipótesis de que, intentando reducir la pérdida de información asociada a la discretización de los atributos numéricos, puede conseguirse aumentar la calidad de las reglas generadas. En este trabajo se ha abordado este problema mediante un enfoque determinista, generando reglas a partir de las diversas combinaciones de intervalos para los atributos numéricos. Sin embargo, este proceso exhaustivo podría ser altamente costoso en tiempo de cómputo, especialmente si los datos analizados tienen una elevada magnitud y complejidad.

Partiendo de este objetivo general, se plantean los siguientes objetivos específicos:

- Definir e implementar estructuras de datos y algoritmos que puedan mejorar el coste computacional vinculado con la generación y evaluación de reglas.
- Conseguir un equilibrio entre el tiempo de cómputo y la calidad de las reglas encontradas, mediante la definición y el uso de parámetros apropiados.
- Aplicar y evaluar el método propuesto en diversos conjuntos de datos de prueba, comparando la bondad de las reglas resultantes con la de otros métodos existentes en la literatura. Con el objetivo de valorar, con más exactitud, la significancia de los resultados obtenidos, se aplicarán diversos tests estadísticos en las evaluaciones propuestas.
- Adaptar y aplicar el método propuesto a distintas tareas de la minería de datos. Concretamente, a la obtención de reglas de asociación, el descubrimiento de subgrupos y la clasificación.

2.2. Tests estadísticos para evaluar los resultados

En minería de datos, se realizan habitualmente análisis experimentales que permiten comparar una técnica con otras en relación a su comportamiento en un determinado problema. Con el propósito de obtener una validación estadística de los resultados, se requiere el empleo de tests apropiados para evaluar dichos estudios comparativos. En el ámbito de la estadística, los tests paramétricos trabajan con valores numéricos y son sensibles a suposiciones de independencia, normalidad y homocedasticidad en la muestra de resultados, mientras que los no paramétricos operan con valores ordinales (rankings) y son más flexibles que los primeros en el tipo de resultados donde pueden ser aplicados [García et al. 2010].

Los tests no paramétricos, permiten por tanto detectar diferencias significativas entre los resultados obtenidos al comparar varios algoritmos, siendo su uso recomendable cuando no se puede asumir que los datos se ajustan a una distribución conocida [Demšar, 2006] [García et al., 2008]. Entre los tests no paramétricos más usados figuran los siguientes:

- *Test de rangos con signo de Wilcoxon.* Esta prueba no paramétrica compara el rango medio de dos muestras para determinar si la posible diferencia entre ellas se debe al azar o no. Se utiliza como alternativa a la prueba *t* de Student cuando no puede suponerse la normalidad de las muestras [Wilcoxon, 1945] [Demšar, 2006].

Básicamente, ordena las diferencias en la ejecución de dos métodos en cada conjunto de datos, en valor absoluto, y compara la suma de los valores de ranking positivos (*R+*, donde el segundo método mejoró al primero) con la suma de los valores de ranking negativos (*R-*, donde el primer método mejoró al segundo). Si el valor resultante es significativo, se rechaza la hipótesis nula (ambos métodos tienen un comportamiento similar).

- *Test de Friedman.* Se usa para la comparación simultánea entre los resultados de más de dos métodos, siendo la alternativa no paramétrica al test de ANOVA de medidas repetidas [Friedman, 1937] [Demšar, 2006]. Procede mediante la ordenación de los métodos separadamente para cada conjunto de datos, obteniendo el mejor método el valor de ranking 1, el segundo mejor el valor de 2, etc., y finalmente calcula los valores medios de ranking para cada método. Permite detectar diferencias considerando el total de

métodos, y una vez rechazada la hipótesis nula (todos los métodos son equivalentes y por tanto los valores medios de ranking son iguales), se pueden realizar tests *post-hoc* para encontrar las comparativas concretas que producen tales diferencias.

- *Test de Iman-Davenport*. Es un test no paramétrico derivado del test de Friedman, y que tiene un comportamiento menos conservativo al emplear un estadístico más ajustado [Iman et. al, 1980] [Demšar, 2006].
- *Test de Holm*. Es un test *post-hoc*, empleado una vez que se ha rechazado la hipótesis de igualdad entre los métodos, y que permite analizar entre cuáles de ellos se produce una diferencia significativa [Holm, 1979] [Demšar, 2006].
- *Test de Bergmann-Hommel*. Test *post-hoc* con mayor poder discriminatorio en ciertos casos, aunque más costoso de calcular [Bergmann et al., 1988] [García et al., 2008].

2.3. Validación cruzada

En muchos problemas de minería de datos, como en el caso de las tareas de clasificación, se requiere disponer de suficientes datos para llevar a cabo los procesos de entrenamiento y validación de un modelo. Pero a veces los datos son limitados, y puede caerse en el error de entrenar y validar el modelo empleando los mismos datos, lo que puede llevar a obtener modelos con mala generalización (el problema del *overfitting*). Para solucionar esto, se puede proceder a validar el modelo con un subconjunto de los datos disponibles, empleando el resto para el entrenamiento. Una técnica frecuentemente usada, y que proporciona buenos resultados, se conoce como validación cruzada (*cross-validation*) [Arlot et al., 2010].

La validación cruzada reduce la varianza y mejora la estimación de generalización de los algoritmos. Por ejemplo, usando validación cruzada de orden 10, los datos de partida se dividen en 10 particiones de igual tamaño y se repite 10 veces el proceso de entrenamiento y validación, empleándose cada una de las 10 particiones una única vez para validación. Finalmente, se hace un compendio de las validaciones obtenidas en el conjunto de las 10 iteraciones. En la Figura 2.1 se muestra un esquema de este proceso.

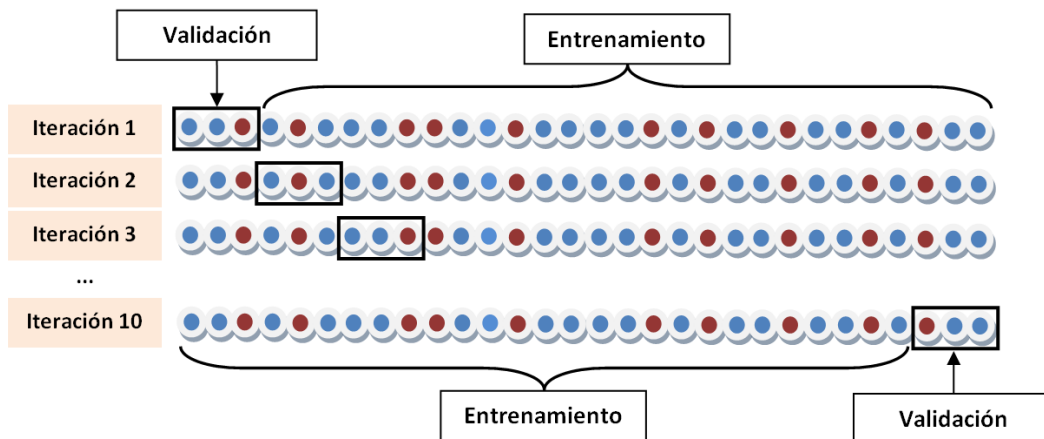


Figura 2.1: Proceso de validación cruzada de orden 10.

En ciertos casos, es conveniente realizar las particiones siguiendo la distribución presente en el conjunto de datos completo (*validación cruzada estratificada*). Por ejemplo, si se tuvieran dos tipos de datos (clases), representados en la figura mediante los colores rojo y azul, y el tipo rojo representase un $1/3$ del total, en cada partición debería respetarse esa misma distribución ($1/3$ para el tipo rojo y $2/3$ para el azul).

3. Un método determinista para la búsqueda de reglas

En este capítulo se describen, en primer lugar, las características del enfoque propuesto en este trabajo. Además, se exponen los principales resultados obtenidos al aplicar los algoritmos a diferentes tareas de minería de datos (reglas de asociación, descubrimiento de subgrupos y clasificación), realizando, para ello, comparativas con diferentes propuestas.

3.1. Reglas de asociación numéricas (DEQAR)

En la tarea de reglas de asociación, es habitual que los datos contengan atributos que puedan tomar valores numéricos, con un rango de valores amplio. Si no se opta por una discretización previa de dichos atributos (esto es, dividir su rango numérico en intervalos y tratar dichos intervalos como valores discretos), puede hablarse de *reglas de asociación numéricas*.

Esta tarea de minería de datos fue presentada inicialmente en [Srikant et al., 1996]. La búsqueda de reglas de asociación numéricas no es una tarea trivial, debido a la complejidad computacional por la explosión combinatoria. Dicha complejidad computacional es, en el caso general, del tipo *NP-completo* [Wijsen et al., 1998].

3.1.1. Descripción del método

El método propuesto (DEQAR, *Deterministic Extraction of Quantitative Association Rules*) no realiza una discretización previa de los atributos numéricos, sino que se basa en una generación "dinámica" de intervalos, buscando reglas válidas que satisfagan los umbrales de mínimo soporte (*minsup*) y mínima confianza (*minconf*).

Este método puede aplicarse a conjuntos de datos que contengan atributos tanto de tipo numérico como categórico (discreto). En el caso de atributos numéricos, sus condiciones pueden tener la forma de un intervalo cerrado de valores, o bien ser un intervalo semiabierto (con $-\infty$ o $+\infty$ en uno de los extremos). Por tanto, para un atributo numérico A , sus condiciones podrían ser: $A \in [a_1, a_2]$, $A \in (-\infty, a]$ o $A \in [a, +\infty)$; o de forma equivalente: $a_1 \leq A \leq a_2$, $A \leq a$ o $A \geq a$. El tratamiento de los atributos categóricos se lleva a cabo asociando previamente sus valores con números naturales, y teniendo en cuenta

únicamente los intervalos cuyos extremos tengan el mismo valor; sería por tanto equivalente a la igualdad con alguna de las categorías del atributo: $A = c$.

Las características principales de DEQAR son las siguientes:

- *Cambio de dominio.* Los extremos de los intervalos para un atributo numérico deben ser valores existentes en el conjunto de datos para dicho atributo. Con esta limitación no se pierden posibles reglas de interés, ya que para cualquier regla, hay siempre una equivalente (en términos de soporte y confianza) que la cumple.

Para buscar eficientemente los extremos de los intervalos, se realiza un cambio de dominio y se trabaja en el conjunto de los números naturales $\{1, 2, \dots, n\}$, siendo n el número de instancias del conjunto de datos. Se realiza para ello la ordenación ascendente de los valores de cada atributo, almacenándose en una tabla auxiliar (T_0). Otras dos tablas auxiliares, T_1 y T_2 , se encargan de almacenar números de fila para facilitar el cálculo del soporte de las condiciones de las reglas. Concretamente, en T_1 se almacena la fila de T_0 en la que se encuentra el correspondiente valor del conjunto de datos de partida; y en T_2 se almacena la fila del conjunto de datos de partida en la que se encuentra el correspondiente valor de T_0 . En la Figura 3.1 se muestra un ejemplo de creación de estas tablas auxiliares. Como se puede apreciar, el primer valor de la primera columna en T_1 almacena un 4, ya que en el atributo A_1 el valor 29 ocupa la cuarta posición en la tabla T_0 ; y en la tabla T_2 el primer valor de su segunda columna almacena un 5, puesto que el valor 10 del atributo A_2 ocupa la quinta fila en los datos de partida.

En la generación final de las reglas, se transformarán, mediante la tabla T_0 , los extremos de los intervalos establecidos en cada regla, para usar los valores originales de los atributos del conjunto de datos de partida. En el ejemplo de la Figura 3.1, el intervalo $[2, 4]$ para el atributo A_1 daría lugar a la condición $A_1 \in [T_0[2, 1], T_0[4, 1]] \equiv A_1 \in [14, 29]$.

Datos de partida			T_0			T_1			T_2		
A_1	A_2	A_3	A_1	A_2	A_3	A_1	A_2	A_3	A_1	A_2	A_3
29	12	31	13	10	31	4	3	1	2	5	1
13	29	55	14	11	52	1	5	4	3	3	3
14	11	52	18	12	52	2	2	2	4	1	4
18	23	52	29	23	55	3	4	3	1	4	2
32	10	77	32	29	77	5	1	5	5	2	5

Figura 3.1: Ejemplo de creación de las tablas auxiliares.

- *Secuencias de intervalos.* Se ha optado por el siguiente esquema de generación de intervalos para las condiciones de las reglas: si se tiene una secuencia de intervalos $S = (I_1, I_2, \dots, I_m)$, donde $I_i = [v_{i1}, v_{i2}]$ es un intervalo para el atributo i -ésimo, la siguiente secuencia a considerar sería una de las tres siguientes:

$$S' = ([v_{11}, v_{12}], [v_{21}, v_{22}], \dots, [v_{m1}, v_{m2}], [1, n])$$

$$S' = ([v_{11}, v_{12}], [v_{21}, v_{22}], \dots, [v_{m1}, v'_{m2}]), \quad v'_{m2} < v_{m2}$$

$$S' = ([v_{11}, v_{12}], [v_{21}, v_{22}], \dots, [v'_{m1}, v_{m2}]), \quad v'_{m1} > v_{m1}$$

Es decir, o bien se añade el intervalo $[1, n]$ para el siguiente atributo, o se reduce el último intervalo por alguno de sus extremos. Además, la reducción de intervalos debe conllevar disminución del soporte, para evitar la generación de reglas redundantes que tengan el mismo soporte. Por tanto, el método no va a generar todas las posibles combinaciones de condiciones, y con ello todas las posibles reglas, sino un subconjunto equivalente en términos de soporte y confianza, con condiciones definidas mediante intervalos "mínimos".

El método procede iterativamente, a partir de la secuencia vacía de intervalos, añadiendo o reduciendo intervalos. En esta enumeración de nuevas secuencias, hay que tener en cuenta el umbral de soporte mínimo y las posibles repeticiones de valores en cada atributo. Un aspecto destacado de este método es que el cálculo del nuevo soporte se realiza progresivamente: si se ha añadido un nuevo intervalo, el soporte no cambia, puesto que el intervalo $[1, n]$ lo cumplen todas las instancias; y si se ha reducido el último intervalo, el nuevo soporte será el resultado de disminuir el soporte de la

secuencia previa con el número de instancias que puedan haberse filtrado con dicha reducción del intervalo.

Con los datos de la Figura 3.1, unas posibles secuencias de intervalos son las mostradas en la Tabla 3.1. Desde la secuencia $([1, 5])$, con un soporte de 5 instancias, podría pasarse a la secuencia $([1, 4])$, que decrementa en 1 su soporte, al haberse filtrado la instancia correspondiente a $A_1 = 5$. El paso de la secuencia $([1, 4], [1, 5])$ a $([1, 4], [1, 4])$ también conlleva la disminución de una instancia en el soporte, al filtrarse la instancia correspondiente a $A_2 = 5$. Esto se comprobaría, en primer lugar, accediendo a la tabla T_2 , que indica que el quinto valor del segundo atributo está en la segunda instancia de los datos de partida ($T_2[5, 2] = 2$); luego, accediendo a T_1 , se vería que en esa segunda instancia está el primer valor del primer atributo ($T_1[2, 1] = 1$); y puesto que 1 pertenece al primer intervalo $([1, 4])$, se comprueba que la instancia no estaba ya filtrada por la primera condición, y hay por tanto reducción del soporte. De forma concisa: $T_1[T_2[5, 2], 1] = 1 \in [1, 4]$, por lo que se filtra una instancia.

secuencia	soporte
<i>vacía</i>	5
$([1, 5])$	5
$([1, 4])$	4
$([1, 4], [1, 5])$	4
$([1, 4], [1, 4])$	3

Tabla 3.1: Ejemplo de posibles secuencias de intervalos.

Sin embargo, no sería considerada la secuencia $([1, 4], [2, 5])$, puesto que no se reduce el soporte, al estar la correspondiente instancia ya filtrada por el primer intervalo: $T_1[T_2[1, 2], 1] = T_1[5, 1] = 5 \notin [1, 4]$.

Como puede verse, para el cálculo del nuevo soporte tras reducir un extremo del intervalo no se necesita un recorrido por todas las instancias del conjunto de datos, sino consultar simplemente las tablas T_1 y T_2 para comprobar si la correspondiente instancia ya estaba filtrada por algún intervalo previo.

Los atributos asociados con intervalos $[1, n]$ se eliminan en la presentación final de las reglas, puesto que no conllevan ningún cambio en el soporte y resultan redundantes.

- *Generación de consecuentes.* Para cada secuencia de intervalos que conforma un antecedente, se procede a buscar intervalos que den lugar a un consecuente, entre los atributos no empleados en dicho antecedente. En este caso hay que tener en cuenta tanto el umbral de soporte mínimo como el de confianza mínima. También se generan secuencias de intervalos para el consecuente, aunque los intervalos $[1, n]$, para cada nuevo atributo de la secuencia, pasan por una "reducción" inicial: se aumenta su extremo inferior y se disminuye su extremo superior mientras no se reduzca el soporte de la regla. Con ello se consigue una regla con los mismos valores de soporte y confianza, pero con un mejor valor de *lift* (un menor soporte del consecuente implica un mayor valor de *lift*). En la Tabla 3.2 se muestra un ejemplo de cómo reduciendo el consecuente sin disminuir el soporte de la regla, puede incrementarse el *lift* sin perder confianza.

antecedente	consecuente	soporte	confianza	soporte del consecuente	lift
$A_1 \in [2, 4]$	$A_2 \in [1, 5]$	3	1.00	5	1.00
$A_1 \in [2, 4]$	$A_2 \in [1, 4]$	3	1.00	4	1.25
$A_1 \in [2, 4]$	$A_2 \in [2, 4]$	3	1.00	3	1.67

Tabla 3.2: Reglas con los mismos valores de soporte y confianza pero distinto valor de *lift*.

- *Proceso de búsqueda de reglas.* El método emplea una búsqueda en profundidad, recorriendo sucesivamente los atributos del primero al último, generando secuencias de intervalos válidos y con vuelta atrás (*backtracking*) cuando ya no puede reducirse más un intervalo por haber alcanzado el umbral de soporte mínimo (o también el de confianza mínima para el caso del consecuente). Se van guardando los valores de soporte de la secuencia de intervalos ya existente al considerar cada nuevo atributo (en la incorporación de un nuevo intervalo $[1, n]$), para no tener que volver a calcular el soporte en el proceso de vuelta atrás.

En la Tabla 3.3 se muestran algunas de las reglas generadas para el conjunto de datos de la Figura 3.1, con un valor de 0.4 para *minsup* (2 instancias) y 0.7 para *minconf*. Se generaron un total de 140 reglas.

regla	soporte	confianza
$A_1 \leq 29 \rightarrow A_2 \geq 11$	0.80	1.00
$A_1 \leq 29 \rightarrow A_2 \geq 11$ y $A_3 \leq 55$	0.80	1.00
$A_1 \leq 29 \rightarrow A_2 \geq 11$ y $A_3 \leq 52$	0.60	0.75
$A_1 \leq 29 \rightarrow A_2 \geq 11$ y $A_3 \geq 52$	0.60	0.75
$A_1 \leq 29 \rightarrow A_2 \in [11, 23]$	0.60	0.75
$A_1 \leq 29 \rightarrow A_2 \in [11, 23]$ y $A_3 \leq 52$	0.60	0.75
$A_1 \leq 29 \rightarrow A_2 \geq 12$	0.60	0.75
$A_1 \leq 29 \rightarrow A_2 \geq 12$ y $A_3 \leq 55$	0.60	0.75
$A_1 \leq 29 \rightarrow A_3 \leq 55$	0.80	1.00
$A_1 \leq 29 \rightarrow A_3 \leq 52$	0.60	0.75
$A_1 \leq 29 \rightarrow A_3 \geq 52$	0.60	0.75
$A_1 \leq 29 \rightarrow A_3 \in [52, 55]$	0.60	0.75
$A_1 \leq 29$ y $A_2 \leq 23 \rightarrow A_3 \leq 52$	0.60	1.00
$A_1 \leq 29$ y $A_2 \leq 12 \rightarrow A_3 \leq 52$	0.40	1.00
$A_1 \leq 29$ y $A_2 \geq 12 \rightarrow A_3 \leq 55$	0.60	1.00
$A_1 \leq 29$ y $A_2 \in [12, 23] \rightarrow A_3 \leq 52$	0.40	1.00
$A_1 \leq 29$ y $A_2 \geq 23 \rightarrow A_3 \in [52, 55]$	0.40	1.00

Tabla 3.3: Algunas de las reglas generadas con *minsup* = 0.4 y *minconf* = 0.7 para el conjunto de datos de ejemplo de la Figura 3.1.

- *Selección final de reglas.* El conjunto de reglas obtenidas se almacena en función de alguna de las medidas de calidad, o de una combinación de ellas, dando lugar a un ranking de reglas según dichos valores de bondad. En la gran mayoría de problemas resultaría inabordable el almacenamiento de todas las reglas obtenidas, por lo que se guarda un subconjunto con las mejores, limitando la longitud de esta lista de reglas mediante el parámetro *tamañoRanking*. Tras la obtención de este ranking, se realiza un filtrado final para eliminar posibles reglas similares. El criterio usado para evaluar la similitud entre reglas es el siguiente: dos reglas son similares si tienen los mismos atributos y para cada atributo existe solapamiento en sus intervalos. Este filtrado final se realiza sucesivamente desde la primera regla, eliminando todas las reglas similares que se encuentren en una posición inferior en el ranking.

En el ejemplo de reglas generadas para el conjunto de datos de la Figura 3.1, si se elige una ordenación según confianza y soporte (a igualdad en la confianza, se elige la regla con mayor soporte), el

ranking final resultante tiene 24 reglas. La Tabla 3.4 muestra las 12 primeras reglas en dicho ranking.

regla	soporte	confianza
$A_1 \leq 29 \rightarrow A_3 \leq 55$	0.80	1.00
$A_1 \leq 29 \rightarrow A_2 \geq 11$	0.80	1.00
$A_1 \leq 29 \rightarrow A_2 \geq 11$ y $A_3 \leq 55$	0.80	1.00
$A_2 \leq 23 \rightarrow A_1 \geq 14$	0.80	1.00
$A_2 \geq 11 \rightarrow A_3 \leq 55$	0.80	1.00
$A_2 \geq 11 \rightarrow A_1 \leq 29$ y $A_3 \leq 55$	0.80	1.00
$A_3 \leq 55 \rightarrow A_2 \geq 11$	0.80	1.00
$A_3 \leq 55 \rightarrow A_1 \leq 29$	0.80	1.00
$A_3 \leq 55 \rightarrow A_1 \leq 29$ y $A_2 \geq 11$	0.80	1.00
$A_2 \leq 23$ y $A_3 \leq 55 \rightarrow A_1 \in [14, 29]$	0.60	1.00
$A_1 \leq 29$ y $A_2 \leq 23 \rightarrow A_3 \leq 52$	0.60	1.00
$A_1 \in [14, 18] \rightarrow A_3 = 52$	0.40	1.00

Tabla 3.4: Doce primeras reglas del ranking final correspondiente al conjunto de datos de ejemplo de la Figura 3.1, según una ordenación por confianza-soporte.

3.1.2. Reducción del número de reglas

Una búsqueda exhaustiva de condiciones para las reglas, como la expuesta anteriormente, puede resultar muy costosa en tiempo de procesamiento incluso para conjuntos de datos no muy grandes. Hay que recordar que la tarea de encontrar reglas de asociación cuantitativas es un problema de tipo *NP*-completo. Aunque DEQAR no genera todas las combinaciones posibles de intervalos para las condiciones, puesto que únicamente considera los subintervalos que reducen el soporte, ya las condiciones para el primer atributo serían del orden de n^2 para el peor caso (cuando todos los valores del atributo son distintos). Por tanto, el árbol resultante de combinar las distintas condiciones para cada atributo podría dar lugar a un proceso muy costoso computacionalmente.

Para controlar la exhaustividad del proceso de búsqueda y reducir el número de reglas generadas, se emplean los parámetros *maxAtrA*, *maxAtrC* y *delta*. Los parámetros *maxAtrA* y *maxAtrC* fijan el máximo número de atributos en el antecedente y el consecuente, respectivamente, limitando las combinaciones de atributos y dando lugar también a reglas más simples y comprensibles.

El parámetro *delta* $\in (0, 1]$ influye en la generación de intervalos para los atributos numéricos, definiendo la disminución mínima del soporte cuando se reducen los intervalos. Esta disminución mínima del soporte es evaluada al

principio de la incorporación de cada nuevo atributo, siguiendo la Ecuación 3.1, donde *sup* es el soporte actual debido a las condiciones ya establecidas.

$$\text{reducción mínima} = (\text{sup} - \text{minsup}) \cdot \text{delta} \quad (3.1)$$

Para conseguir un ajuste del soporte de la regla al umbral del soporte mínimo (*minsup*), también se consideran aquellas reducciones finales con las que se haya alcanzado *minsup*, aun cuando la disminución de soporte sea inferior a la reducción mínima.

En las Figuras 3.2 y 3.3 se han representado gráficamente unos ejemplos de búsqueda de intervalos para las condiciones de dos atributos *A* y *B*. En la Figura 3.2, partiendo del intervalo $[a_1, a_2]$ para el atributo *A* (con un soporte de 8 instancias), y suponiendo un valor de *minsup* = 2 instancias y un valor de *delta* = 1/3 (la reducción mínima sería por tanto 2 instancias), la búsqueda de intervalos para *B* podría dar lugar a los valores b_1, b_2 y b_3 como extremos de sus intervalos. Las condiciones posibles serían las siguientes:

$$\begin{aligned} A \in [a_1, a_2] \text{ y } B \in (-\infty, b_3] \\ A \in [a_1, a_2] \text{ y } B \in (-\infty, b_2] \\ A \in [a_1, a_2] \text{ y } B \in (-\infty, b_1] \\ A \in [a_1, a_2] \text{ y } B \in [b_1, +\infty) \\ A \in [a_1, a_2] \text{ y } B \in [b_1, b_3] \\ A \in [a_1, a_2] \text{ y } B \in [b_1, b_2] \\ A \in [a_1, a_2] \text{ y } B \in [b_2, +\infty) \\ A \in [a_1, a_2] \text{ y } B \in [b_2, b_3] \\ A \in [a_1, a_2] \text{ y } B \in [b_3, +\infty) \end{aligned}$$

En la Figura 3.3 se muestra cómo en el caso de partir del intervalo $[a_2, a_3]$ para el atributo *A*, los intervalos considerados para *B* son distintos, en función de los valores b_4, b_5 y b_6 . Queda patente en este ejemplo, la diferencia entre este método de búsqueda de intervalos respecto a una discretización inicial que considerase los atributos aisladamente.

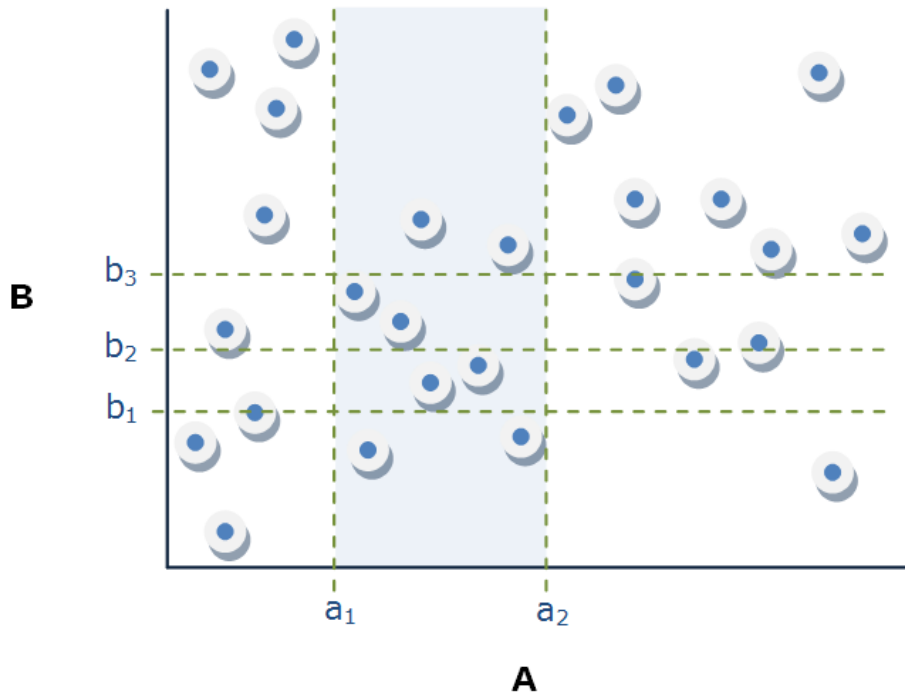


Figura 3.2: Ejemplo de búsqueda de intervalos.

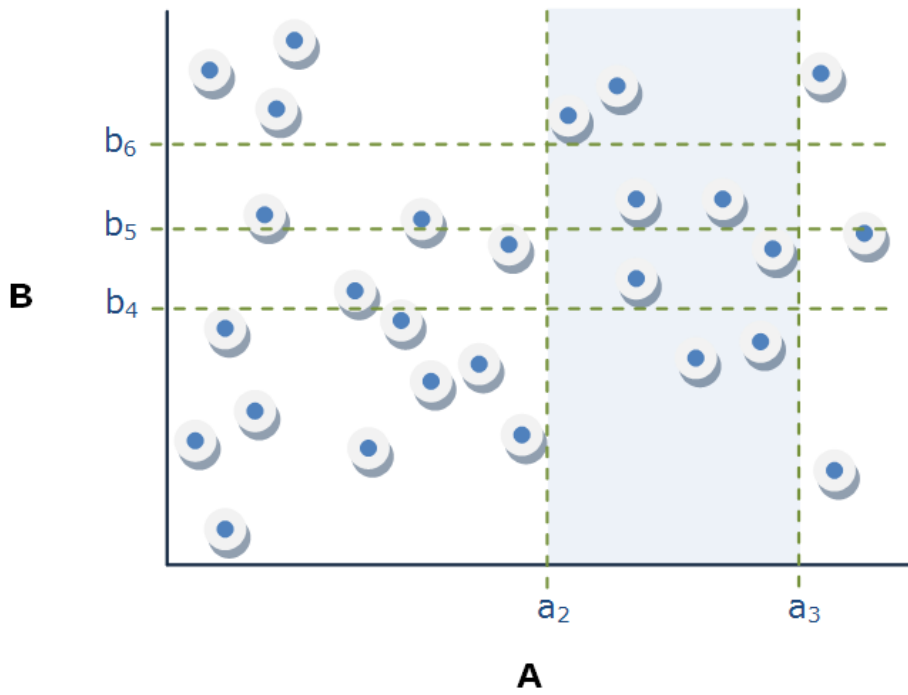


Figura 3.3: Ejemplo de búsqueda de intervalos.

3.1.3. Algoritmo DEQAR

El esquema general del algoritmo DEQAR se muestra en el Algoritmo 1. Tras la creación de las tablas auxiliares, se procede con la generación sucesiva de antecedentes (*obtenerAntecedente*), buscando posibles subintervalos válidos y teniendo en cuenta las restricciones debidas a los parámetros *minsup*, *delta* y *maxAtrA*. Con cada antecedente generado, *obtenerConsecuente* se encarga de buscar un consecuente válido para asociarlo a dicho antecedente, según las restricciones impuestas por los valores de *minsup*, *minconf*, *delta* y *maxAtrC*. Dicha generación de consecuentes opera mediante la búsqueda de posibles subintervalos válidos para los atributos no usados en el antecedente. Al final del proceso se obtiene una lista con las reglas seleccionadas, con un máximo dado por *tamañoRanking* y ordenadas según el mejor valor de la medida *medidaCalidad*, la cual, como se describió anteriormente, podría ser también una combinación de varias medidas.

Algoritmo 1 DEQAR

Entrada: Conjunto de datos D , *minsup*, *minconf*, *delta*, *maxAtrA*, *maxAtrC*, *tamañoRanking*, *medidaCalidad*

Salida: Lista de reglas R

- 1: crear las tablas auxiliares T_0 , T_1 , y T_2
 - 2: **repetir**
 - 3: $A = \text{obtenerAntecedente}()$
 - 4: **repetir**
 - 5: $C = \text{obtenerConsecuente}()$
 - 6: insertar en R la regla $A \rightarrow C$ según su valor de *medidaCalidad*
 - 7: **hasta** (no hay más consecuentes posibles para el antecedente)
 - 8: **hasta** (no hay más antecedentes posibles)
 - 9: filtrar R eliminando las reglas similares
 - 10: transformar las reglas de R con los valores originales de D
 - 11: **devolver** R
-

Para comprobar empíricamente el efecto de la variación del parámetro *delta* en el proceso de generación de reglas, se han medido los tiempos de cómputo y los valores de confianza obtenidos para distintos valores de dicho parámetro. Para ello, se fijó el consecuente de las reglas a un único atributo categórico presente en los conjuntos de datos *abalone*, *ecoli*, *forestfires*, *housing*, *wine* y *yeast*, cuyas características se muestran en la Tabla 3.5. Se ejecutó DEQAR limitando a 3 el número de atributos en el antecedente (*maxAtrA*) y utilizando diferentes valores para el parámetro *delta* (0.40, 0.35, 0.30, 0.25, 0.20, 0.15 y

0.10). Las Figuras 3.4-3.9 muestran el tiempo en segundos y el valor medio de confianza de las 10 mejores reglas obtenidas en cada uno de estos conjuntos de datos.

Puede observarse en todos los casos que, a medida que el valor del parámetro *delta* se aproxima a 0, el tiempo de cómputo se eleva considerablemente, acorde con el incremento del número de reglas generadas. El parámetro *delta* influye en el número de intervalos considerados para cada atributo numérico, lo que afecta al número de posibles combinaciones para las condiciones de las reglas. También se puede comprobar que el valor de la confianza mejora, por lo general, cuando disminuye el valor de *delta*, puesto que se generan más reglas y es más probable que entre ellas se encuentren las de mayor confianza. Sin embargo, las variaciones en el valor de la confianza no son tan pronunciadas como las correspondientes al tiempo de ejecución. Por ejemplo, en el conjunto de datos *yeast*, variando el valor de *delta* de 0.2 a 0.1 se multiplicó por 17 el tiempo de cómputo, y sin embargo, la confianza solo mejoró un 2.4%.

conjunto de datos	# instancias	# atributos	# categóricos
abalone	4,177	9	1
ecoli	336	8	1
forest_fires	517	13	2
housing	506	14	1
wine	178	14	1
yeast	1,484	9	1

Tabla 3.5: Conjuntos de datos usados para evaluar la influencia de los parámetros.

El parámetro *maxAtrA* también afecta de forma importante al número de reglas generadas, puesto que fija el máximo número de atributos que formarán parte del antecedente. Para comprobar cómo afecta este parámetro al tiempo de cómputo y a la confianza de las reglas, se ha procedido a medir los valores obtenidos al variar *maxAtrA*. En las Figuras 3.10-3.13 se muestran el tiempo en segundos y el valor medio de confianza de las 10 mejores reglas obtenidas para los conjuntos de datos *abalone*, *ecoli*, *housing* y *wine*, variando *maxAtrA* desde 1 a 7. Se observa que los valores altos de *maxAtrA* afectaron considerablemente al tiempo de cómputo pero no influyeron de forma tan importante en la confianza. Por ejemplo, en *abalone*, un valor de *maxAtrA* mayor que 5 no aporta una ganancia de confianza tan significativa como el incremento del tiempo de cómputo.

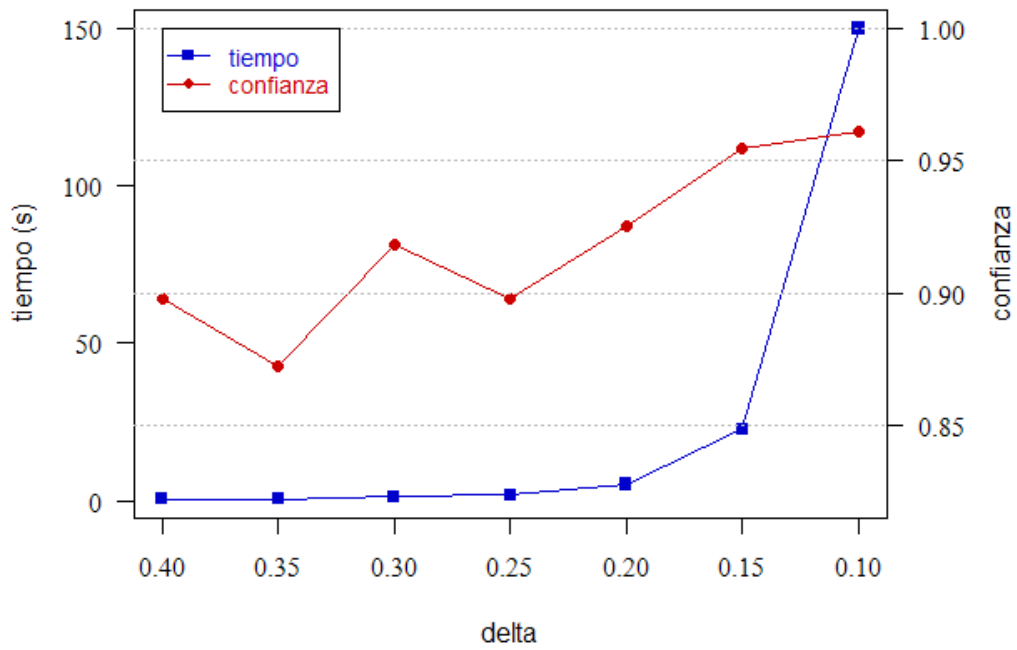


Figura 3.4: Resultados de ejecución para el conjunto de datos *abalone*, para varios valores del parámetro *delta*.

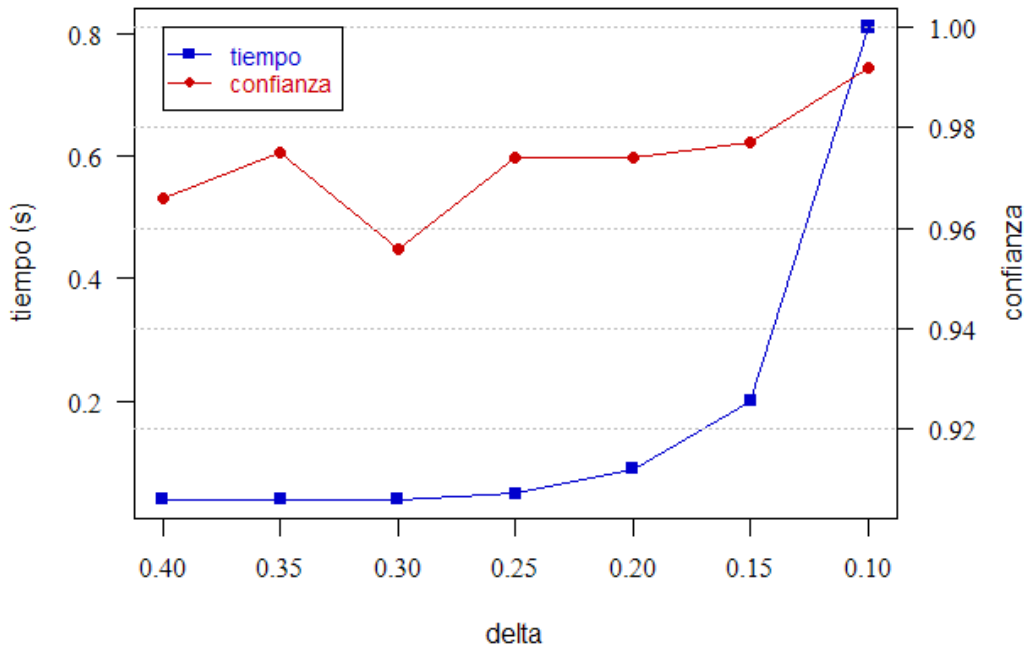


Figura 3.5: Resultados de ejecución para el conjunto de datos *ecoli*, para varios valores del parámetro *delta*.

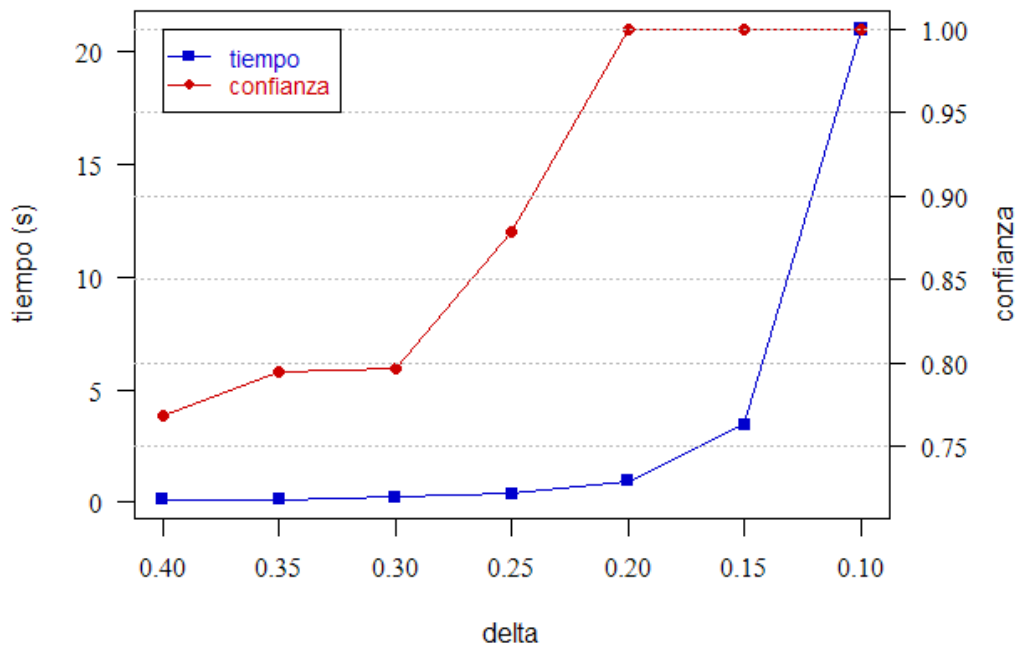


Figura 3.6: Resultados de ejecución para el conjunto de datos *forestfires*, para varios valores del parámetro *delta*.

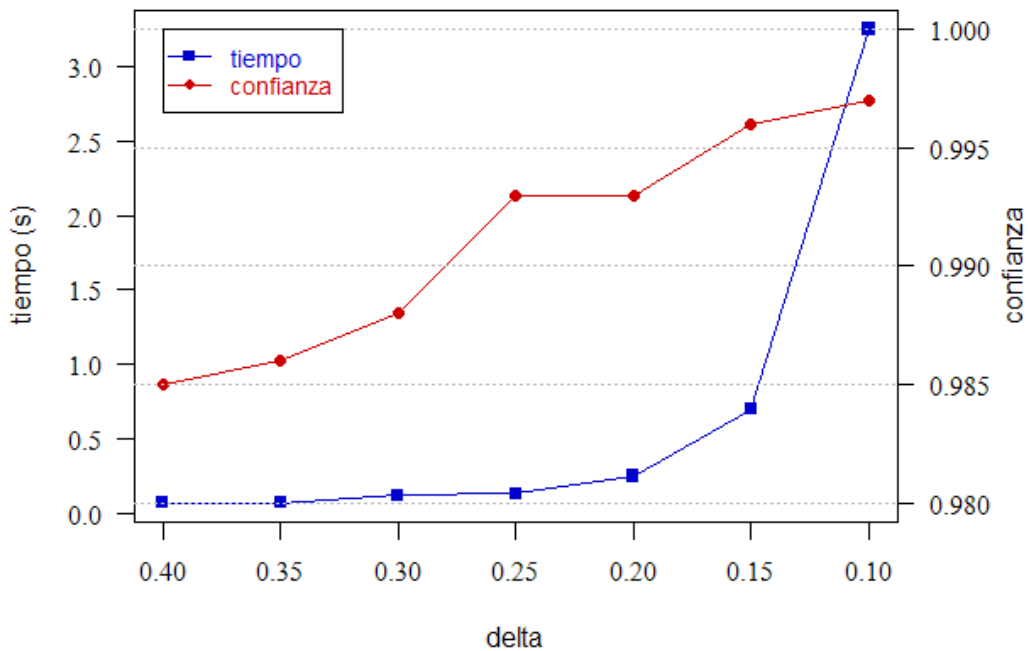


Figura 3.7: Resultados de ejecución para el conjunto de datos *housing*, para varios valores del parámetro *delta*.

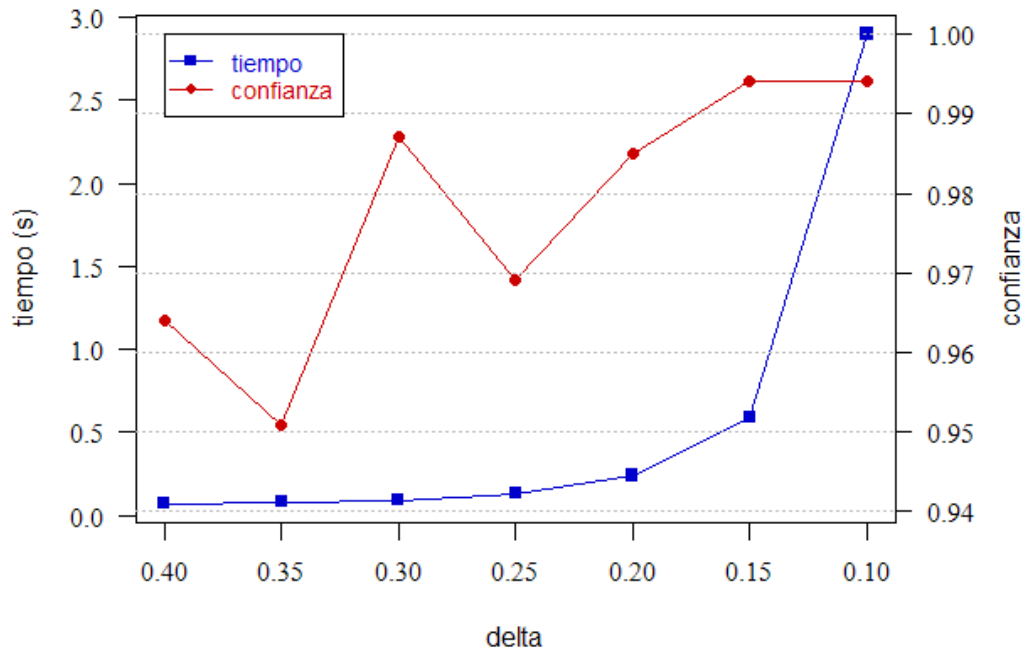


Figura 3.8: Resultados de ejecución para el conjunto de datos *wine*, para varios valores del parámetro *delta*.

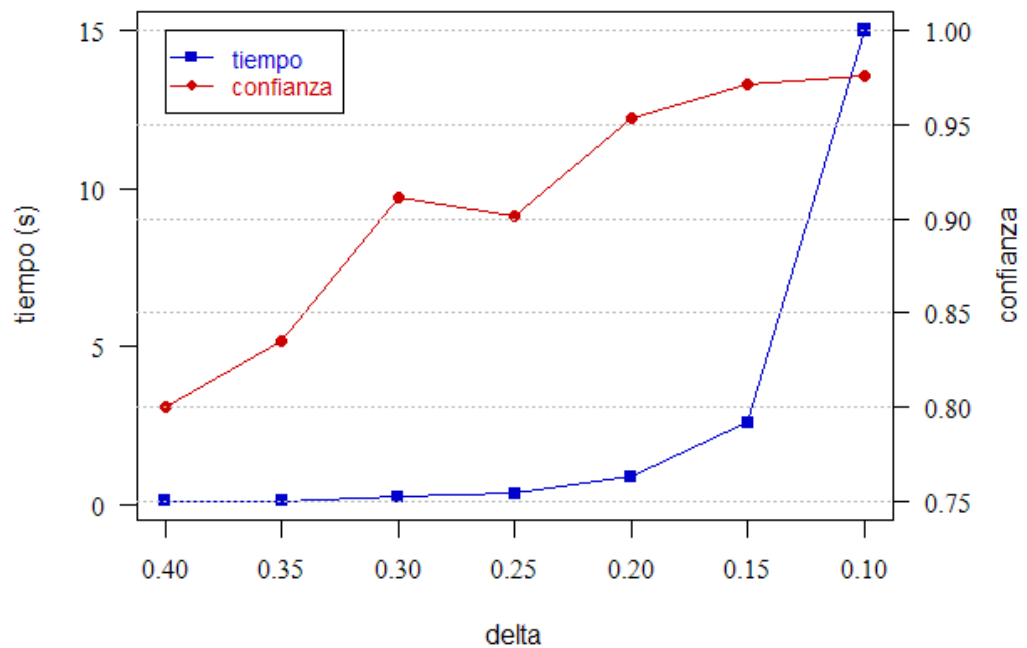


Figura 3.9: Resultados de ejecución para el conjunto de datos *yeast*, para varios valores del parámetro *delta*.

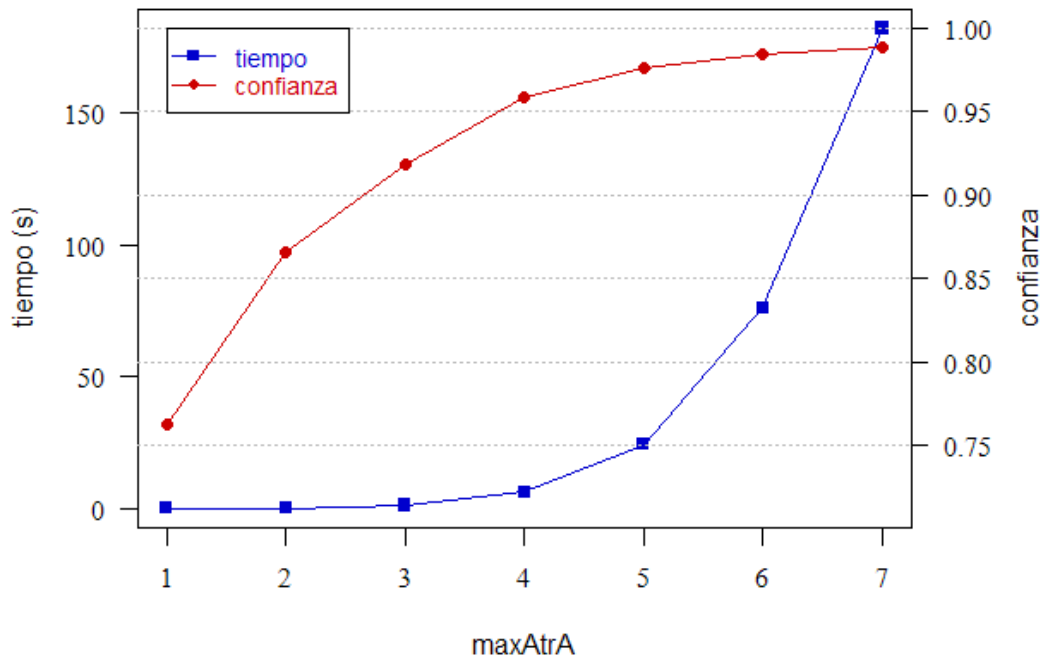


Figura 3.10: Resultados de ejecución para el conjunto de datos *abalone*, con distintos valores del parámetro *maxAtrA*.

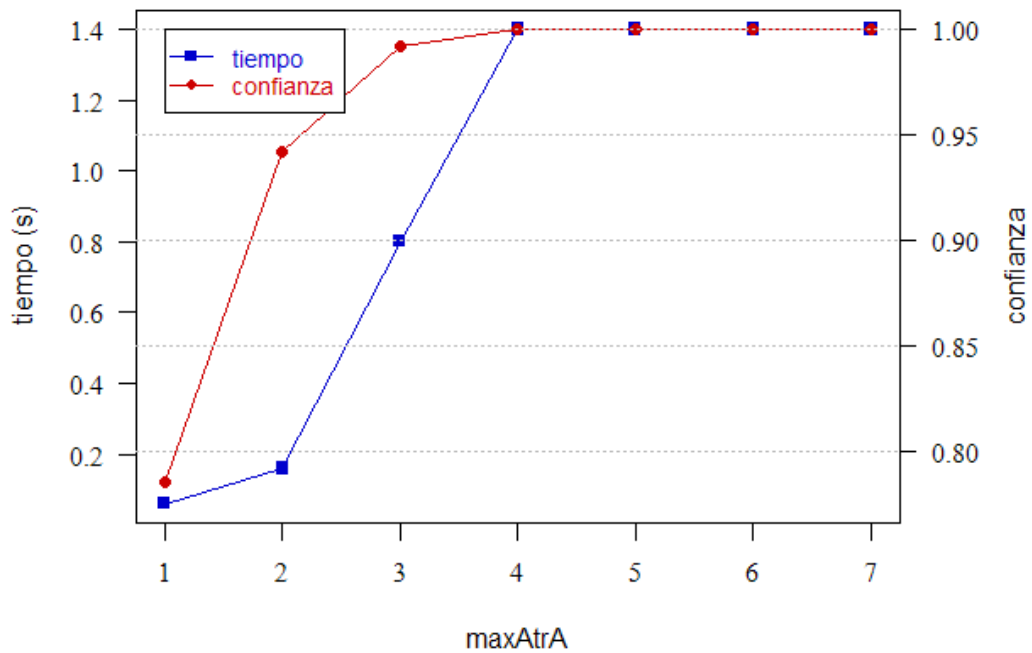


Figura 3.11: Resultados de ejecución para el conjunto de datos *ecoli*, con distintos valores del parámetro *maxAtrA*.

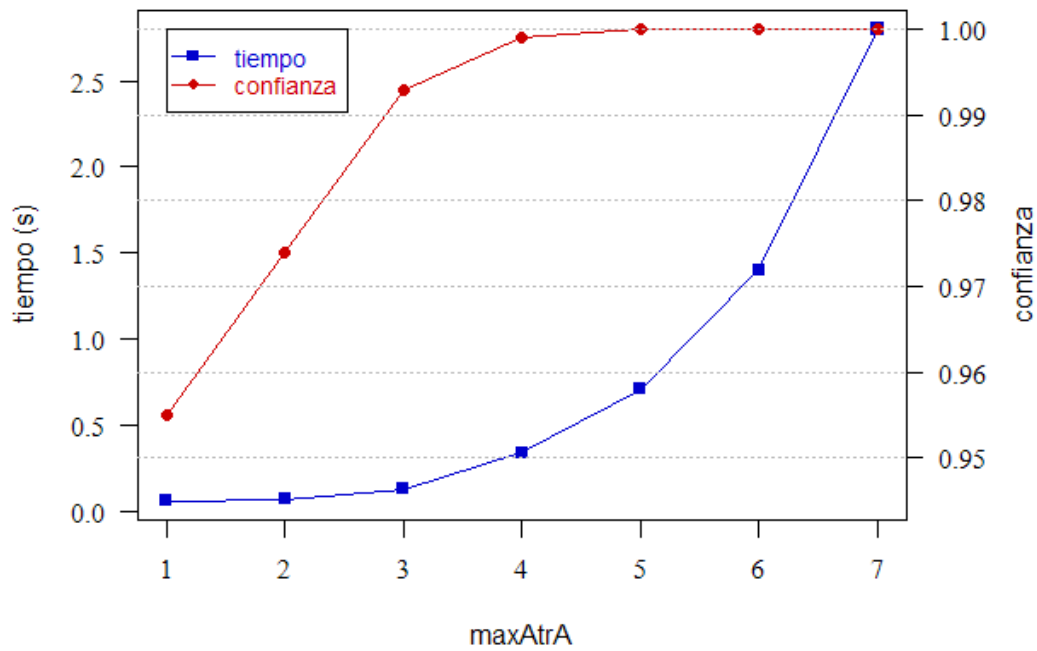


Figura 3.12: Resultados de ejecución para el conjunto de datos *housing*, con distintos valores del parámetro *maxAtrA*.

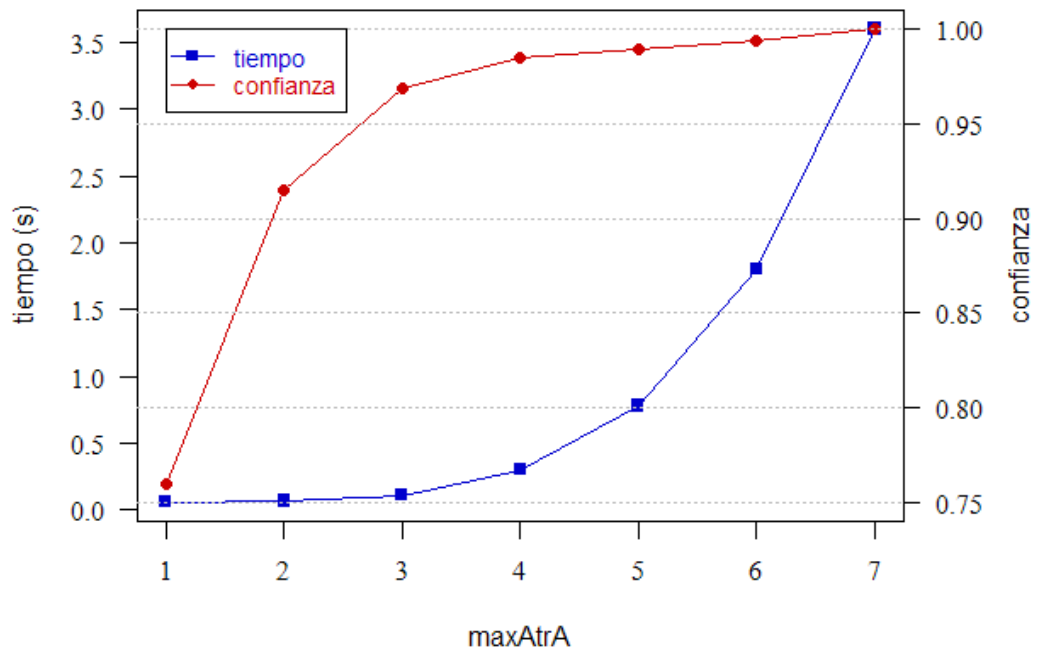


Figura 3.13: Resultados de ejecución para el conjunto de datos *wine*, con distintos valores del parámetro *maxAtrA*.

Se ha realizado un tercer estudio para medir los valores de confianza y tiempo de cómputo en función de los parámetros *maxAtrA* y *delta*. Las Figuras

3.14 y 3.15 representan las gráficas de contorno resultantes para confianza y tiempo (escala logarítmica), en el conjunto de datos *abalone*. Puede observarse cómo el tiempo de cómputo crece considerablemente hacia la esquina inferior derecha, a medida que aumenta *maxAtrA* y disminuye *delta*. Sin embargo, la confianza no presenta una variación tan pronunciada en esa dirección. Por ejemplo, puede observarse en las figuras que los puntos (4, 0.20) y (5, 0.15) tienen unos valores de confianza de 0.961 y 0.998 respectivamente, y sus valores de tiempo son 50 y 4000 segundos, lo que supone multiplicar por 1.04 la confianza y por 80 el tiempo.

A la vista de estos resultados, el uso en DEQAR de estos parámetros para controlar la exhaustividad de la búsqueda de reglas, permite ajustar el tiempo de cómputo a expensas de cierta reducción en la confianza de las reglas obtenidas, consecuencia lógica de haber generado un menor número de reglas. Pero no se observa una correspondencia "lineal" entre el tiempo de procesamiento y la confianza obtenida, sino que con tiempos relativamente bajos pueden conseguirse valores de confianza relativamente altos. Para cada conjunto de datos en particular, habría que encontrar los valores "óptimos" de estos parámetros, aunque parece recomendable no fijar valores muy altos para *maxAtrA* ni muy bajos para *delta*.

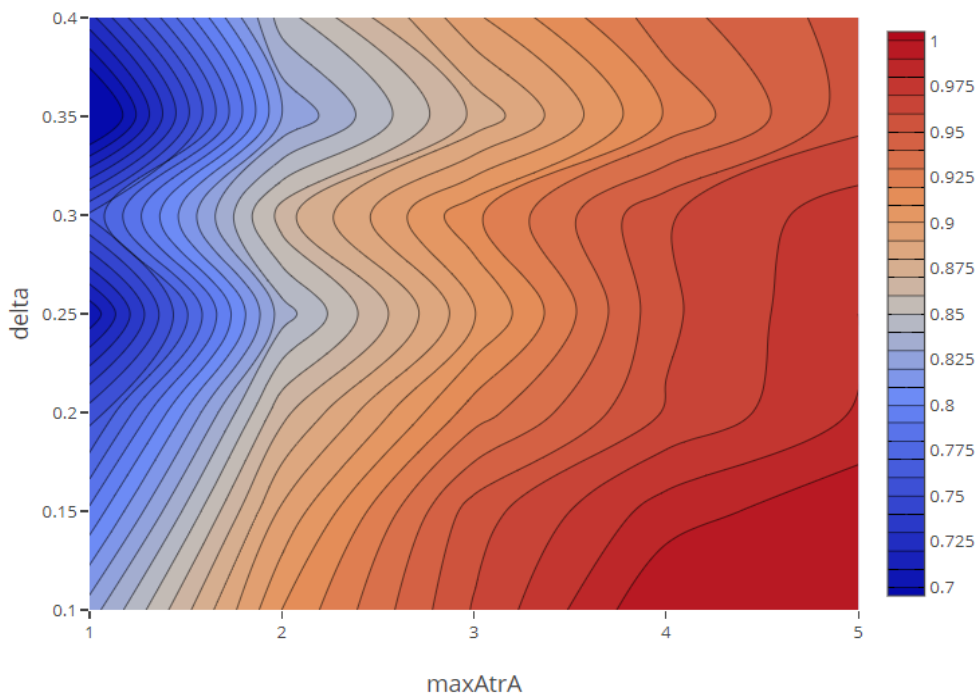


Figura 3.14: Valores de confianza obtenidos para el conjunto de datos *abalone*, variando los parámetros *maxAtrA* y *delta*.

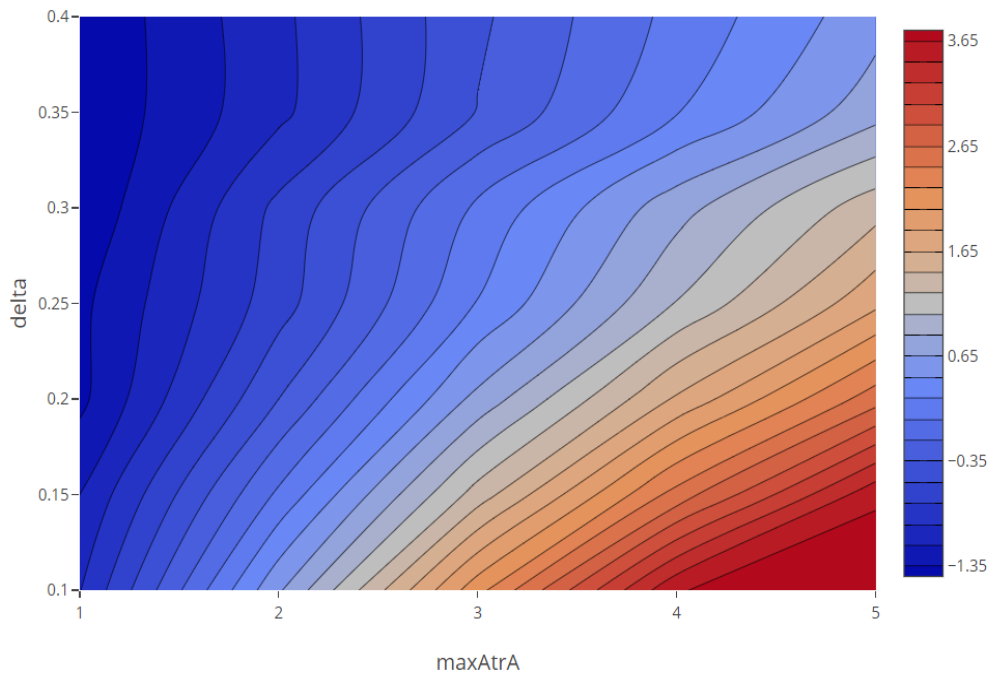


Figura 3.15: Tiempos de ejecución (escala logarítmica) para el conjunto de datos *abalone*, variando los parámetros *maxAtrA* y *delta*.

3.1.4. Resultados experimentales

3.1.4.1. Comparativa con Apriori

Como se ha comentado previamente en la Sección 1.1, el algoritmo Apriori es el método exhaustivo de referencia para la generación de reglas de asociación. Como la mayoría de los algoritmos de búsquedas de reglas de asociación, Apriori necesita una discretización previa de los atributos numéricos, pues precisa que todos los atributos sean de tipo categórico. En la Sección 1.1 se han descrito los métodos de discretización no supervisada igual-anchura e igual-frecuencia, que son los más utilizados en la tarea de reglas de asociación.

Se ha realizado una extensa comparativa entre los algoritmos DEQAR y Apriori, utilizando para ello 16 conjuntos de datos tomados de los repositorios *UCI* [Asuncion et al., 2007] y *Bilkent* [Guvendir et al., 2000], con la mayor parte de sus atributos de tipo numérico. Las características de estos conjuntos de datos se muestran en la Tabla 3.6.

conjunto de datos	# instancias	# atributos	# numéricos	# categóricos
abalone	4,177	9	8	1
bodyfat	252	18	18	0
bolts	40	8	8	0
ecoli	336	8	7	1
elevators	16,599	19	19	0
forest_fires	517	13	11	2
house_16h	22,784	17	17	0
housing	506	14	13	1
longley	16	7	7	0
mortgage	1,049	16	16	0
pollution	60	16	16	0
school	62	20	20	0
stock_prices	950	10	10	0
veteran	137	8	4	4
wine	178	14	13	1
yeast	1,484	9	8	1

Tabla 3.6: Características de los conjuntos de datos usados en los experimentos.

En el algoritmo Apriori, para la discretización de los atributos numéricos mediante igual-anchura (EW) e igual-frecuencia (EF), se emplearon distintos valores para el número de particiones (*bins*): 2, 3, 4, 6, 8 y 9. Por tanto, para cada conjunto de datos se realizaron 12 ejecuciones del algoritmo Apriori, empleando 2 métodos de discretización con 6 valores para el número de particiones. Para llevar a cabo dicha discretización se utilizó el paquete *Weka* [Hall et al., 2009], y se empleó la implementación del algoritmo Apriori obtenida en [www.adrem.ua.ac.be/~goethals/software].

En la experimentación, los valores de los parámetros utilizados en DEQAR fueron 0.5 y 0.4 para el parámetro *delta*, las combinaciones (3, 2) y (2, 1) para el par de parámetros (*maxAtrA*, *maxAtrC*) y un valor de 5000 para el parámetro *tamañoRanking*. Así pues, para cada conjunto de datos se realizaron un total de 4 ejecuciones de DEQAR.

En ambos métodos, los parámetros *minsup* y *minconf* se fijaron a 0.1 y 0.8 respectivamente, y el conjunto de reglas obtenido en cada experimento se ordenó según sus valores de confianza y soporte (en caso de igualdad en la confianza, decide el soporte). Se llevaron a cabo dos comparativas, según las mejores 100 y 25 reglas de cada ranking, para obtener valores de confianza media, soporte medio, *lift* medio, cobertura y número medio de atributos. La cobertura de un conjunto de reglas se calculó como el porcentaje de instancias del conjunto de datos que son cubiertas por el antecedente de alguna de las reglas.

Los resultados de los experimentos para los 16 conjuntos de datos se detallan en las Tablas A.1-A.32 del Apéndice. Como puede verse, en algunos casos Apriori no produjo suficientes reglas para calcular los valores medios de las medidas. Los valores más altos de *lift* estuvieron asociados generalmente con valores más bajos de soporte, debido a la relación existente entre el *lift* y el soporte del consecuente. Con respecto al número de atributos usados en las reglas, DEQAR obtuvo buenos resultados, obviamente por el empleo de los parámetros que limitan su número (*maxAtrA* y *maxAtrC*).

Para evaluar los resultados de la comparativa entre DEQAR y Apriori, se ha llevado a cabo una validación estadística de los mismos, mediante el test de Friedman (tests no paramétrico de comparación múltiple, descrito en la Sección 2.2). Los tests no paramétricos permiten detectar diferencias significativas entre los resultados obtenidos al comparar varios algoritmos. Para usar el test de Friedman se requiere tener una única medida resultado de la ejecución de un método en un conjunto de datos. Con esta medida calculada para cada conjunto de datos, se forma un ranking con los valores obtenidos por los distintos métodos, y al final se obtiene el valor medio de ranking de cada método en todos los conjuntos de datos.

En los experimentos realizados, puesto que se calcularon valores para cinco medidas de calidad en las mejores 100 y 25 reglas, ordenadas según confianza-soporte, se escogió realizar tests usando distintas medidas:

- confianza - soporte - lift
- confianza - cobertura - lift
- cobertura
- número de atributos

Las dos primeras medidas son compuestas, calculándose un único valor según una suma ponderada. Por ejemplo, para la primera de ellas: *confianza* · 1E6 + *soporte* · 1E2 + *lift*. En la medida relativa al número de atributos, una regla se considera mejor que otra cuando presenta un menor número de atributos (reglas más cortas). Con los cuatro tipos de evaluaciones y los dos subconjuntos de reglas (las 100 mejores y las 25 mejores), se realizaron un total de ocho tests de Friedman, empleando el módulo de tests estadísticos del paquete *Keel* [Alcalá-Fdez et al., 2011].

En las Tablas 3.7 y 3.8 se muestran los valores medios de ranking obtenidos por los distintos métodos en los tests de Friedman. Pueden verse los buenos valores de ranking que presentó DEQAR, aunque ninguna de sus cuatro variantes

quedó en el primer lugar en todas las evaluaciones. Con respecto a Apriori, la discretización mediante igual-anchura con 2 *bins* (APRIORI-EW-2) obtuvo buenos resultados, en parte porque fue uno de los métodos que generó más reglas en los experimentos.

La significancia estadística de los resultados obtenidos está relacionada con los p-valores, que se muestran en la Tabla 3.9 junto a los correspondientes para el test de Iman-Davenport. A menor p-valor, mayor evidencia contra la hipótesis nula (no hay diferencias entre los métodos). Si se rechaza la hipótesis nula, lo que ha ocurrido en todos los casos, se pueden realizar tests post-hoc para analizar entre qué pares de métodos ocurren tales diferencias. Se realizaron los correspondientes tests de Holm para cada medida, cuyos resultados se muestran en las Tablas A.33-A.40 del Apéndice.

Empleando los valores proporcionados por el test de Holm, se realizó una agrupación de los mejores subconjuntos de métodos con un nivel de significancia $\alpha = 0.05$, tal que cualquier método del primer subconjunto (*mejor*) tiene que haber obtenido mejores resultados que cualquier algoritmo del segundo subconjunto (*peor*), y no debe haber diferencias significativas entre los métodos pertenecientes a cada subconjunto. Las Tablas 3.10 y 3.11 muestran estos subconjuntos obtenidos en cada tipo de comparación.

Finalmente, analizando estos subconjuntos de métodos, puede verse que DEQAR-0.4-2-1 (DEQAR con $\delta = 0.4$, $\maxAtrA = 2$ y $\maxAtrC = 1$) obtuvo el mejor resultado, puesto que es el único método que pertenece al subconjunto *mejor* en todas las comparaciones. Por el contrario, APRIORI-EF-9 (Apriori con igual-frecuencia y 9 *bins*), APRIORI-EF-8, APRIORI-EF-6 y APRIORI-EW-9 (Apriori con igual-anchura y 9 *bins*) mostraron los peores resultados.

El buen comportamiento mostrado por DEQAR-0.4-2-1 está vinculado con una búsqueda más exhaustiva de intervalos para las reglas ($\delta = 0.4$ frente a $\delta = 0.5$) y al uso de pocos atributos, lo que en general ha dado lugar a la obtención de reglas más simples y con buenos valores de confianza y cobertura. Por el contrario, el empleo de Apriori con mayor número de particiones en la discretización (sobre todo mediante igual-frecuencia) conllevó peores resultados en los experimentos. Podría pensarse que las reglas generadas por DEQAR tienden a centrarse en una parte del espacio de búsqueda, pero observando los excelentes resultados para la medida cobertura, queda patente la diversidad de reglas que se obtienen, empleando distintos atributos y condiciones.

Observando los resultados de esta experimentación, se comprueba que con el algoritmo desarrollado y los adecuados valores de sus parámetros, puede

llevarse a cabo una búsqueda de reglas de asociación lo suficientemente exhaustiva, que permita obtener reglas simples y con buenos valores de bondad.

método	conf-sop-lift	conf-cob-lift	cobertura	# atributos
APRIORI-EF-2	7.53	6.25	7.84	11.38
APRIORI-EF-3	9.75	8.38	9.06	9.81
APRIORI-EF-4	11.78	11.56	10.31	8.50
APRIORI-EF-6	13.91	13.19	11.53	9.53
APRIORI-EF-8	14.72	13.56	13.19	10.59
APRIORI-EF-9	14.97	13.69	12.69	10.22
APRIORI-EW-2	2.81	6.75	7.34	10.75
APRIORI-EW-3	5.50	8.09	9.34	10.59
APRIORI-EW-4	7.41	7.97	8.78	9.97
APRIORI-EW-6	9.84	9.84	10.66	10.53
APRIORI-EW-8	11.34	11.22	11.09	10.91
APRIORI-EW-9	12.28	12.25	11.44	10.19
DEQAR-0.4-3-2	1.72	3.13	3.09	4.78
DEQAR-0.4-2-1	3.25	4.13	3.09	1.84
DEQAR-0.5-3-2	3.88	2.53	3.44	4.72
DEQAR-0.5-2-1	5.31	3.47	3.09	1.69

Tabla 3.7: Valores medios de ranking (mejores 100 reglas).

método	conf-sop-lift	conf-cob-lift	cobertura	# atributos
APRIORI-EF-2	7.66	6.59	7.31	10.47
APRIORI-EF-3	10.13	8.28	9.06	9.03
APRIORI-EF-4	11.69	9.66	9.84	9.53
APRIORI-EF-6	13.69	12.03	11.59	9.38
APRIORI-EF-8	14.59	12.88	12.41	8.03
APRIORI-EF-9	15.34	13.19	12.56	9.44
APRIORI-EW-2	3.03	6.94	7.41	10.28
APRIORI-EW-3	5.22	8.06	8.56	11.06
APRIORI-EW-4	6.97	8.13	8.13	10.78
APRIORI-EW-6	9.56	10.50	10.31	10.75
APRIORI-EW-8	11.16	11.22	10.97	10.75
APRIORI-EW-9	12.09	12.09	11.84	11.56
DEQAR-0.4-3-2	2.13	3.91	3.91	5.75
DEQAR-0.4-2-1	3.28	3.06	2.19	1.84
DEQAR-0.5-3-2	4.16	6.81	7.38	5.78
DEQAR-0.5-2-1	5.31	2.66	2.53	1.56

Tabla 3.8: Valores medios de ranking (mejores 25 reglas).

comparación	Friedman	Iman-Davenport
confianza-soporte-lift (100)	1.12E-10	1.31E-89
confianza-cobertura-lift (100)	1.11E-10	2.22E-16
cobertura (100)	7.00E-11	1.30E-31
# atributos (100)	5.94E-11	1.11E-16
confianza-soporte-lift (25)	9.16E-11	1.96E-80
confianza-cobertura-lift (25)	5.54E-11	2.22E-16
cobertura (25)	8.47E-11	6.18E-25
# atributos (25)	8.88E-11	5.01E-21

Tabla 3.9: *p*-valores de los tests realizados.

comparación	subconjunto <i>mejor</i>	subconjunto <i>peor</i>
confianza-soporte-lift (100)	DEQAR-0.4-3-2	APRIORI-EF-9
	APRIORI-EW-2	APRIORI-EF-8
	DEQAR-0.4-2-1	APRIORI-EF-6
	DEQAR-0.5-3-2	APRIORI-EW-9
		APRIORI-EF-4
		APRIORI-EW-8
confianza-cobertura-lift (100)	DEQAR-0.5-3-2	APRIORI-EF-9
	DEQAR-0.4-3-2	APRIORI-EF-8
	DEQAR-0.5-2-1	APRIORI-EF-6
	DEQAR-0.4-2-1	APRIORI-EW-9
	APRIORI-EF-2	
cobertura (100)	DEQAR-0.4-3-2	APRIORI-EF-9
	DEQAR-0.4-2-1	APRIORI-EF-8
	DEQAR-0.5-2-1	APRIORI-EF-6
	DEQAR-0.5-3-2	APRIORI-EW-9
		APRIORI-EW-8
		APRIORI-EW-6
# atributos (100)		APRIORI-EF-4
	DEQAR-0.5-2-1	APRIORI-EF-2
	DEQAR-0.4-2-1	APRIORI-EW-8
		APRIORI-EW-2
		APRIORI-EF-8
		APRIORI-EW-3
		APRIORI-EW-6
		APRIORI-EF-9
		APRIORI-EW-9
		APRIORI-EW-4
	APRIORI-EF-3	
	APRIORI-EF-6	
	APRIORI-EF-4	

Tabla 3.10: Subconjuntos de métodos según los tests estadísticos (mejores 100 reglas).

comparación	subconjunto <i>mejor</i>	subconjunto <i>peor</i>
confianza-soporte-lift (25)	DEQAR-0.4-3-2	APRIORI-EF-9
	APRIORI-EW-2	APRIORI-EF-8
	DEQAR-0.4-2-1	APRIORI-EF-6
	DEQAR-0.5-3-2	APRIORI-EW-9
	APRIORI-EW-3	APRIORI-EF-4
confianza-cobertura-lift (25)	DEQAR-0.5-2-1	APRIORI-EF-9
	DEQAR-0.4-2-1	APRIORI-EF-8
	DEQAR-0.4-3-2	APRIORI-EW-9
		APRIORI-EF-6
		APRIORI-EW-8
	APRIORI-EW-6	
cobertura (25)	DEQAR-0.4-2-1	APRIORI-EF-9
	DEQAR-0.5-2-1	APRIORI-EF-8
	DEQAR-0.4-3-2	APRIORI-EW-9
		APRIORI-EF-6
		APRIORI-EW-8
	APRIORI-EW-6	
# atributos (25)	DEQAR-0.5-2-1	APRIORI-EW-9
	DEQAR-0.4-2-1	APRIORI-EW-3
		APRIORI-EW-4
		APRIORI-EW-6
		APRIORI-EW-8
		APRIORI-EF-2
		APRIORI-EW-2
		APRIORI-EF-4
		APRIORI-EF-9
		APRIORI-EF-6
	APRIORI-EF-3	
	APRIORI-EF-8	

Tabla 3.11: Subconjuntos de métodos según los tests estadísticos (mejores 25 reglas).

3.1.4.2. Aplicación al conjunto de datos *Easterly and Levine*

Uno de los objetivos principales del estudio y aplicación de las técnicas de minería de datos es el descubrir asociaciones y patrones ocultos en los datos. Este objetivo se ve reforzado cuando la fuente de los datos recoge situaciones reales y aspectos sociales. Por esta razón, se ha aplicado el método DEQAR al conjunto de datos *Easterly and Levine* [Easterly et al., 1997], con objeto de descubrir reglas de asociación que permitan comprender algunas situaciones relevantes.

Dicho conjunto de datos contiene información diversa sobre 160 países durante el período 1960-1980, de la cual se seleccionó la relativa a 47 países subsaharianos, que dan lugar a 141 instancias con 21 atributos. La descripción de dichos atributos se muestra en la Tabla 3.12.

atributo	descripción
YEAR	Década
AGOVDEM	Manifestaciones antigubernamentales
ASSASS	Número de asesinatos por cada mil habitantes, promedio de la década
BLCK	Log de (1+ <i>black market premium</i>), promedio de la década
CABCHG	Cambios importantes del gabinete
COMPOLT	=1 para un país con incidente de genocidio
CONSTCHG	Cambios constitucionales importantes
DEMOC	Medida de democracia
COUPS	Golpes de estado
ELF60	Índice de fraccionamiento etnolingüístico
GOVTCRIS	Crisis de gobierno importantes
GYP	Tasa de crecimiento del PIB per cápita
LLY	Ratio de pasivos líquidos del sistema financiero con el PIB
LRGDP	Log del PIB per cápita, medido al inicio de cada década
LSCHOOL	Log de (1 + años de escolarización), valores quinquenales
LTELPW	Log de (teléfonos por cada 1000 trabajadores)
PURGES	Encarcelamiento o ejecución de la oposición política
REVOLS	Cambio ilegal o forzado en la élite gubernamental
RIOTS	Manifestación violenta de más de 100 ciudadanos
SURP	Ratio entre el superávit del gobierno central y el PIB, promedio de la década
WAR	= 1 si hubo guerra en el territorio nacional durante la década

Tabla 3.12: Atributos de un conjunto de datos sobre países del África subsahariana en el período 1960-1980.

Se realizó un primer estudio de búsqueda de reglas fijando los parámetros *minsup* a 10%, *minconf* a 80%, *delta* a 0.25, *maxAttrAnt* a 2 y *maxAttrCon* a 1, y se ordenaron según los mejores valores de confianza y *lift*. Algunas de las reglas encontradas se muestran en la Tabla 3.13. Puede verse que se hallaron reglas asociando una variedad de atributos, y con altos valores de confianza y *lift*. Por ejemplo, una de las reglas asocia un nivel de escolarización no muy alto con el bajo crecimiento del PIB, expresando: "En la década de 1980, los países con un valor de *LSCHOOL* inferior a 1.138, presentaron un valor de *GYP* entre -0.048 y 0.014".

Un segundo estudio consistió en la búsqueda de reglas que asociaran el atributo *WAR* (guerra en el territorio nacional), fijando los parámetros *minsup* a 10%, *minconf* a 60% y *delta* a 0.25. Algunas de las mejores reglas se muestran en la Tabla 3.14. Puede observarse que en la mayoría de dichas reglas aparece el atributo *BLCK* ("*black market premium*", variación del valor de la moneda en el mercado negro). Por ejemplo, una de ellas relaciona la guerra con valores no bajos de *BLCK* y *ELF60* (índice de fraccionamiento etnolingüístico), expresando que "el 61.1% de los países que sufrieron guerra, tuvieron un valor de *BLCK* superior a 0.206 y un valor de *ELF60* entre 0.08 y 0.9". A su vez, otra regla asocia

valores bajos de *BLCK* (≤ 0.175) y valores no muy altos de *ELF60* (≤ 0.82) con la ausencia de guerra en el 100% de los casos.

antecedente	consecuente	sop	conf	lift
COMPOLT = 0 LLY ≥ 0.291	LSCHOOL $\in [0.837, 1.725]$	10.6%	100%	3.36
DEMOC ≤ 5 LTEL PW ≥ 2.272	LSCHOOL $\in [0.837, 1.725]$	10.6%	100%	3.36
COMPOLT = 1 GYP ≥ -0.038	BLCK $\in [0.180, 0.993]$	10.6%	100%	3.28
YEAR = 70 SURP ≤ -0.028	LLY $\in [0.154, 0.448]$	10.6%	100%	2.82
YEAR = 60 GYP ≤ 0.012	LRGDP $\in [6.287, 7.067]$	10.6%	100%	2.56
YEAR = 80 LSCHOOL ≤ 1.138	GYP $\in [-0.048, 0.014]$	11.3%	100%	1.99
ELF60 $\in [0.62, 0.71]$ WAR = 0	LLY $\in [0.055, 0.230]$	17.0%	100%	1.93

Tabla 3.13: Algunas de las mejores reglas encontradas para el conjunto de datos *Easterly and Levine*.

antecedente	consecuente	sop	conf	lift
WAR = 1	BLCK ≥ 0.206 ELF60 $\in [0.08, 0.9]$	15.6%	61.1%	2.10
WAR = 0	BLCK ≤ 0.463 COMPOLT = 0 GYP ≥ -0.038 LRGDP $\in [5.837, 8.264]$	45.4%	61.0%	1.25
WAR = 0	COMPOLT = 0 GOVTCRIS = 0 RIOTS ≤ 1	46.8%	62.9%	1.21
BLCK ≥ 0.363 ELF60 ≤ 0.89 LRGDP ≤ 8.438	WAR = 1	10.6%	65.2%	2.55
BLCK ≤ 0.175 ELF60 ≤ 0.82	WAR = 0	25.5%	100%	1.34

Tabla 3.14: Algunas de las mejores reglas encontradas que involucran la variable *WAR*.

Como puede observarse, la aplicación del algoritmo DEQAR a este conjunto de datos obtuvo un conjunto de reglas que asocian diversas condiciones sobre los atributos, y con buenos valores para las medidas de calidad. Puesto que la mayor parte de los atributos son de tipo numérico, y dada la característica de DEQAR de no discretizarlos previamente, se consiguió obtener una variedad de condiciones en las reglas, dando como resultado asociaciones de mayor interés.

3.1.4.3. Aplicación al conjunto de datos *STULONG*

En otro trabajo desarrollado durante la realización de esta tesis, también se ha aplicado DEQAR a datos médicos provenientes de un estudio sobre aterosclerosis.

El conjunto de datos utilizado fue *STULONG*, que contiene información sobre un estudio longitudinal de los factores de la aterosclerosis en la población de 1417 hombres de mediana edad [Boudík et al., 2004]. La aterosclerosis es una enfermedad en la que se deposita placa dentro de las arterias. La placa está compuesta por grasas, colesterol, calcio y otras sustancias que se encuentran en la sangre. Con el tiempo, la placa se endurece y estrecha las arterias, con lo cual se limita el flujo de sangre rica en oxígeno a los órganos y a otras partes del cuerpo. Puede causar problemas graves, como ataque cardíaco, accidentes cerebrovasculares e incluso la muerte [www.nhlbi.nih.gov/health-topics/aterosclerosis].

Del conjunto de atributos en los datos de este estudio, se seleccionaron 16 de ellos relativos a la edad, examen físico, examen bioquímico, consumo de alcohol, consumo de tabaco, y si el paciente había fallecido. Dichos atributos se muestran en la Tabla 3.15.

atributo	descripción
AGE	Edad
HEIGHT	Altura
WEIGHT	Peso
BMI	Índice de masa corporal
SYST	Presión arterial sistólica
DIAST	Presión arterial diastólica
TRIC	Grosor de la piel del tríceps
SUBSC	Grosor de la piel subescapular
CHLST	Colesterol
TRIGL	Triglicéridos
MOC_SUC	Azúcar en la orina
MOC_ALB	Albúmina en la orina
ALCO_CONS	Consumo de alcohol
TOBA_CONS	Consumo de tabaco
TOBA_DURA	Duración del consumo de tabaco
DEATH?	Fallecimiento

Tabla 3.15: Atributos seleccionados del estudio *STULONG*.

El primer objetivo del estudio fue buscar reglas que asociaran atributos de los grupos "Examen físico" (*HEIGHT*, *WEIGHT*, *BMI*, *SYST*, *DIAST*, *TRIC* y *SUBSC*) y

"Examen bioquímico" (*CHLST*, *TRIGL*, *MOC_SUC*, *MOC_ALB*), fijando *minsup* a 2%, *minconf* a 0.5 y *delta* a 0.2. En la Tabla 3.16 se muestran algunas las reglas obtenidas, aquellas que presentaron un alto valor de *lift* o una alta confianza.

La última regla indica que "todos los pacientes con un grosor de la piel del tríceps entre 7 y 12, y con un grosor de la piel subescapular entre 32 y 49, no presentaban azúcar en la orina, tenían un nivel de colesterol entre 211 y 300, y un nivel de triglicéridos entre 103 y 350. Dichas condiciones del consecuente, son el doble de frecuentes en el grupo de pacientes con las condiciones del antecedente que en el total de pacientes estudiados".

regla	soporte	confianza	lift
$CHLST \leq 242$ $TRIGL \leq 71$	31	0.79	2.3
\Rightarrow $BMI \in [19.05, 28.41]$ $TRIC \geq 3$ $SUBSC \in [7, 16]$			
$DIAST \geq 82$ $BMI \in [24.11, 27.36]$ $TRIC \geq 11$ $SUBSC \geq 16$	29	0.51	2.6
\Rightarrow $MOC_SUC = no$ $MOC_ALB = no$ $CHLST \in [221, 250]$ $TRIGL \in [105, 274]$			
$TRIC \in [7, 12]$ $SUBSC \in [32, 49]$	29	1.00	2.0
\Rightarrow $MOC_SUC = no$ $CHLST \in [211, 300]$ $TRIGL \in [103, 350]$			

Tabla 3.16: Una selección de reglas asociando los grupos de atributos "Examen físico" y "Examen bioquímico".

Un segundo estudio consistió en buscar reglas que asociaran el atributo *DEATH?* con el resto de atributos. La Tabla 3.17 muestra una selección de las reglas obtenidas. La primera regla asocia los atributos *ALCO_CONS* (consumo de alcohol), *TOBA_DURA* (duración del consumo de tabaco), *CHLST* (colesterol) y *SYST* (presión sanguínea sistólica), con el atributo *DEATH?*. Su valor de *lift* nos indica que cuando se cumple el antecedente, el fallecimiento es 2.7 veces más frecuente que en el caso general (en el total de pacientes estudiados). El atributo

ALCO_CONS mide la ingesta de alcohol teniendo en cuenta tres factores: la cantidad equivalente de alcohol, el tipo de alcohol y el peso (como factor de normalización). La segunda de ella asocia el no fumar, tener el colesterol y la hipertensión no muy elevados, y no presentar obesidad, con la ausencia de fallecimiento.

regla	soporte	confianza	lift
ALCO_CONS \leq 1.2 TOBA_DURA = 20 CHLST \in [197, 273] SYST \geq 149 \Rightarrow DEATH? = yes	30	0.73	2.7
TOBA_CONSO = 0 CHLST \in [190, 261] SYST \leq 133 BMI \in [22.64, 27.41] \Rightarrow DEATH? = no	52	1.00	1.4

Tabla 3.17: Una selección de reglas obtenidas asociando el atributo *DEATH?*.

Analizando las reglas obtenidas, puede verse que también en este conjunto de datos, DEQAR consiguió generar una diversidad de reglas que asociaban distintos atributos y con buenos valores para las medidas de calidad evaluadas. Faltaría conocer la opinión que sobre dichas reglas pudiera tener un experto (médico), que indicara si expresan conocimiento ya constatado o pudieran resultar interesantes por la asociación de sus condiciones. Aunque a la vista de algunas de ellas, podemos afirmar que son consecuentes con los consejos habituales en temas de salud.

3.2. Descubrimiento de subgrupos (DEQAR-SD)

3.2.1. Descripción del método

El método desarrollado, DEQAR-SD, es una adaptación de DEQAR para la tarea de descubrimiento de subgrupos. En DEQAR las condiciones para los atributos numéricos pueden tomar en algunos casos la forma de un intervalo cerrado. En DEQAR-SD se ha optado por no usar este tipo de intervalos, y por tanto una condición para un atributo numérico podrá ser únicamente de las formas $A \leq v$ o $A \geq v$, siendo v un valor existente en los datos para el atributo A .

Se ha hecho así, por un parte, para conseguir reglas con unas condiciones más simples, y por otra, porque se ha comprobado que se consigue una reducción sustancial en el tiempo de cómputo, aunque con la contrapartida de perder algo de confianza en las reglas generadas. Por ejemplo, las Figuras 3.16 y 3.17 muestran los tiempos de cómputo y los valores de confianza con el uso de intervalos cerrados (*INT*) y sin su uso (*NOINT*), para el conjunto de datos *abalone* y variando los parámetros *delta* y *maxAtrA*. Puede verse la gran diferencia en el tiempo de cómputo cuando la búsqueda se hace más exhaustiva (menores valores de *delta* o mayores de *maxAtrA*). Sin embargo, los valores de confianza, aunque en general son algo menores sin el uso de intervalos, no presentan una diferencia tan pronunciada.

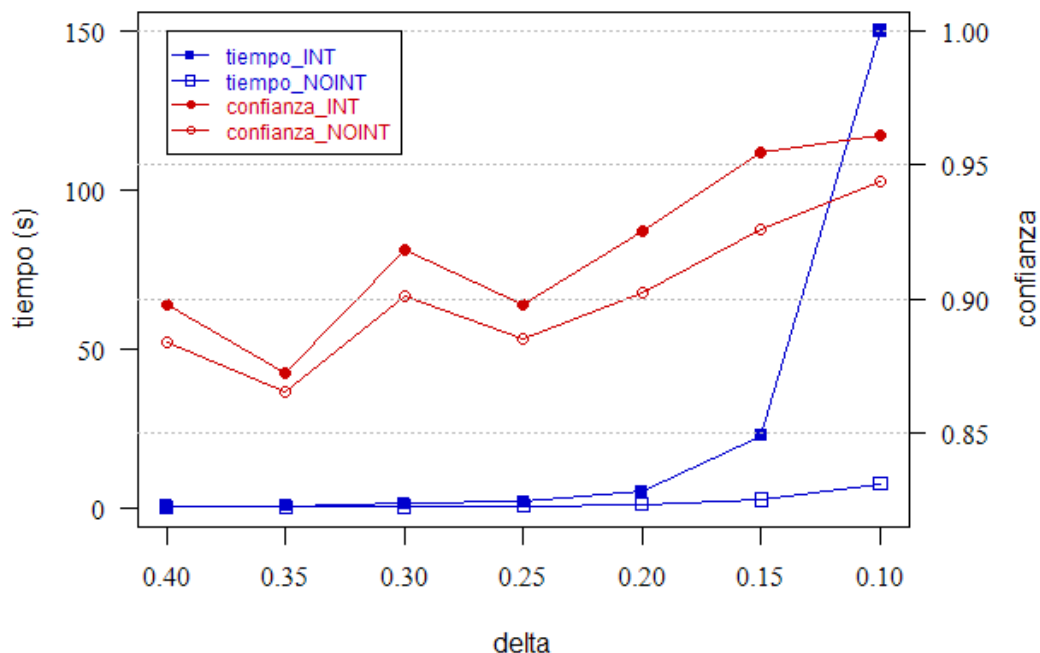


Figura 3.16: Resultados para el conjunto de datos *abalone*, con distintos valores del parámetro *delta*, empleando condiciones con intervalos cerrados (*INT*) y sin ellos (*NOINT*).

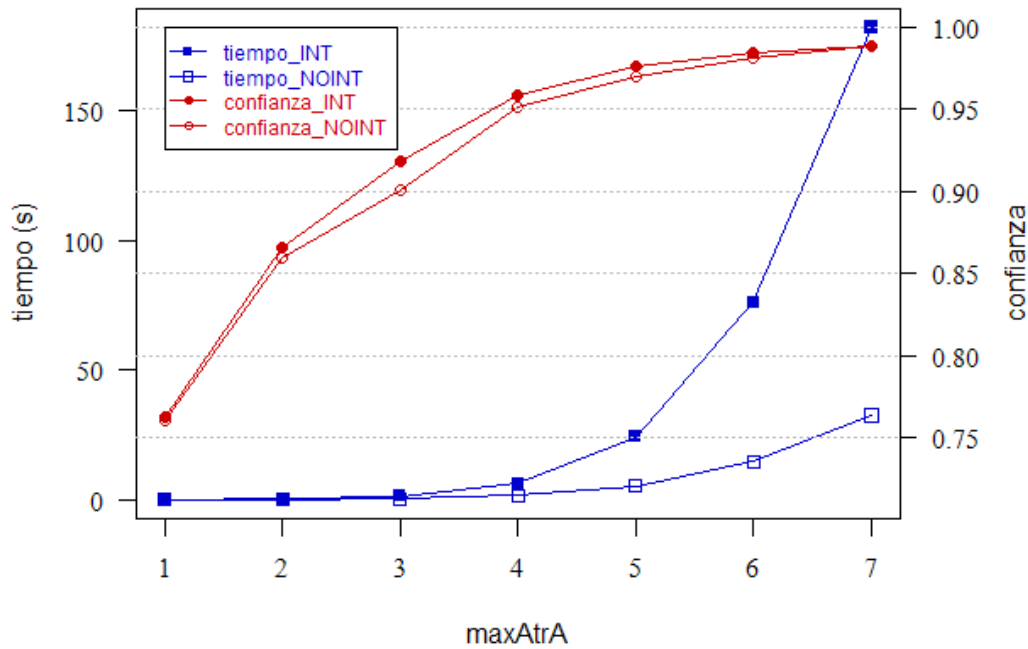


Figura 3.17: Resultados para el conjunto de datos *abalone*, con distintos valores del parámetro *maxAtrA*, empleando condiciones con intervalos cerrados (*INT*) y sin ellos (*NOINT*).

Estas son las principales características diferenciadoras de DEQAR-SD:

- No se emplea directamente un parámetro para el umbral de soporte mínimo (*minsup*), sino que se calcula este soporte mínimo a partir del parámetro *minsens* (sensibilidad mínima). La Ecuación 3.2 muestra dicho cálculo, que es función del número de instancias que componen cada clase.

$$\text{minsup} = \text{minsens} * \min_c (\text{soporte}(\text{clase} = c)) \quad (3.2)$$

- No se utiliza el parámetro relativo a la confianza mínima (*minconf*). En este método, en lugar de su uso, simplemente se descartan las reglas cuyo valor de *lift* sea inferior a 1 (no se alcanza la probabilidad de la clase de dicha regla).
- Con la intención de seleccionar un subconjunto con las mejores reglas, se emplea el parámetro *mincub*, que define un valor mínimo para la ratio de cubrimiento de las instancias de cada clase.

- Durante el proceso se emplea una lista ordenada (ranking) para cada una de las clases. La medida empleada para la ordenación de las reglas es la equivalente a la suma de sensibilidad y especificidad (aunque la clase no tiene porqué ser necesariamente binaria), según expresa la Ecuación 3.3.

$$\begin{aligned} \text{medida}(A \rightarrow C) &= \text{sensibilidad}(A \rightarrow C) + \text{especificidad}(A \rightarrow C) = \\ &= \frac{\text{soporte}(A \wedge C)}{\text{soporte}(C)} + 1 - \frac{\text{soporte}(A) - \text{soporte}(A \wedge C)}{n - \text{soporte}(C)} \end{aligned} \quad (3.3)$$

A mayor valor de esta medida, la regla tiene mayor poder discriminatorio para esa clase, al calcularse tanto la tasa de aciertos en dicha clase como la correspondiente al resto de instancias que no pertenecen a ella.

El esquema del algoritmo DEQAR-SD se muestra en el Algoritmo 2. Se repite sucesivamente la búsqueda de antecedentes válidos, es decir, que cumplan con el soporte mínimo y no superen el máximo número de atributos (línea 2). Dicha búsqueda sigue el mismo esquema que el usado en DEQAR, salvo por las particularidades arriba expuestas, como la ausencia de intervalos cerrados para las condiciones numéricas (línea 3). Para cada antecedente válido generado, se pasa a formar una regla para cada posible valor del atributo clase, asociando dicho antecedente con un consecuente formado por uno de los posibles valores de la clase (línea 4). Si la regla es válida (su valor de *lift* no es inferior a 1 y cumple con el umbral de mínima sensibilidad), pasa a insertarse en el correspondiente ranking para esa clase (líneas 5-7). Tras obtenerse estos rankings de reglas, se pasa a un filtrado en cada uno de ellos, suprimiendo reglas similares y aquellas que no cubran alguna instancia adicional a las ya cubiertas por reglas previas (línea 11). Así mismo, se realiza otro filtrado posterior, para dejar en cada ranking las primeras reglas con las que se alcance el cubrimiento mínimo (*mincub*) para cada clase (línea 12). Finalmente, se devuelve una lista de reglas conteniendo el conjunto de estos rankings.

3.2.2. Comparativa con métodos deterministas

Para comprobar el funcionamiento del método propuesto, se ha llevado a cabo un estudio comparativo entre DEQAR-SD y los métodos clásicos deterministas APRIORI-SD [Kavšek et al., 2003] y CN2-SD [Lavrač et al., 2004].

Para ello, se han utilizado 18 conjuntos de datos provenientes fundamentalmente del repositorio de aprendizaje automático UCI [Asuncion et al., 2007]. En la Tabla 3.18 se muestran las características de estos conjuntos de datos.

Algoritmo 2 DEQAR-SD

 Entrada: Conjunto de datos D , $minsens$, $delta$, $maxAtr$, $tamañoRanking$, $mincub$

 Salida: Lista de reglas R

- 1: crear las tablas auxiliares
 - 2: **repetir**
 - 3: $A = obtenerAntecedente()$
 - 4: **para** cada clase c **hacer**
 - 5: **si** la regla $A \rightarrow c$ es válida **entonces**
 - 6: insertar en el ranking[c] la regla $A \rightarrow c$
 - 7: **fin si**
 - 8: **fin para**
 - 9: **hasta** (no hay más antecedentes válidos)
 - 10: **para** cada clase c **hacer**
 - 11: filtrar el ranking [c] para suprimir reglas similares
 - 12: filtrar el ranking [c] para obtener un cubrimiento mínimo
 - 13: añadir a R las reglas del ranking[c]
 - 14: **fin para**
 - 15: transformar las reglas de R con los valores originales de D
 - 16: **devolver** R
-

conjunto de datos	# instancias	# atributos	# numéricos	# clases
appendicitis	106	7	7	2
australian	690	14	6	2
balance	625	4	4	3
bridges	102	7	3	2
cleveland	303	13	13	5
contraceptive	1473	9	9	3
diabetes	768	8	8	2
echo	131	6	5	2
ecoli	336	7	7	8
haberman	306	3	3	2
hayes-roth	160	4	4	3
heart	270	13	7	2
iris	150	4	4	3
led	500	7	7	10
phoneme	5404	5	5	2
tic-tac-toe	958	9	0	2
wine	178	13	13	3
wisconsin	699	9	9	2

Tabla 3.18: Características de los conjuntos de datos usados en los experimentos.

Las implementaciones empleadas para los algoritmos Apriori-SD y CN2-SD han sido las existentes en el paquete Keel [Alcalá-Fdez et al., 2011]. Puesto que estos métodos requieren de una discretización para el tratamiento de los atributos numéricos, se ha empleado el método de discretización Fayyad como paso previo para su aplicación en los conjuntos de datos [Fayyad et al., 1993]. Este tipo de discretización supervisada está basado en los criterios de mínima entropía y de mínima longitud de la descripción (MDL) [Rissanen, 1978] [Boullé et al., 2018].

En los experimentos, los valores de los parámetros empleados en Apriori-SD y CN2-SD fueron los valores por defecto recomendados. Con respecto a DEQAR-SD, se usaron los valores de 0.5 para *minsens*, 0.33 para *delta*, 2 para *maxAtr*, 5 para *tamañoRanking* y 0.90 para *mincub*. Como medidas de calidad de las reglas, en los tres métodos se calcularon la cobertura media, la relevancia media, el número de reglas y el número medio de atributos del antecedente. Los valores finales de estas medidas en cada conjunto de datos se obtuvieron mediante el empleo de validación cruzada estratificada de orden 10. Los resultados obtenidos por los tres métodos en cada medida evaluada, se muestran en las Tablas 3.19-3.22. Observando los valores particulares en estos conjuntos de datos, puede comprobarse que en la medida cobertura DEQAR-SD consiguió muy buenos resultados, obteniendo el mejor valor de cobertura en 10 de los 18 conjuntos de datos.

conjunto de datos	DEQAR-SD	APRIORI-SD	CN2-SD
appendicitis	0.546	0.469	0.409
australian	0.500	0.499	0.418
balance	0.415	0.457	0.330
bridges	0.403	0.297	0.411
cleveland	0.373	0.542	0.332
contraceptive	0.495	0.090	0.346
diabetes	0.488	0.263	0.468
echo	0.436	0.735	0.430
ecoli	0.246	0.233	0.207
haberman	0.549	0.848	0.500
hayes-roth	0.378	0.073	0.263
heart	0.424	0.484	0.394
iris	0.403	0.295	0.347
led	0.215	0.110	0.197
phoneme	0.454	0.196	0.494
tic-tac-toe	0.417	0.235	0.229
wine	0.327	0.301	0.306
wisconsin	0.502	0.453	0.507

Tabla 3.19: Resultados obtenidos para la medida *cobertura*.

conjunto de datos	DEQAR-SD	APRIORI-SD	CN2-SD
appendicitis	1.449	1.864	1.867
australian	20.505	17.398	16.770
balance	7.641	6.102	7.486
bridges	1.183	0.656	0.653
cleveland	6.902	7.198	7.932
contraceptive	5.924	8.161	5.792
diabetes	6.210	4.876	4.667
echo	0.870	0.249	0.532
ecoli	12.232	9.000	12.193
haberman	1.056	0.550	1.543
hayes-roth	2.409	3.769	4.565
heart	4.799	3.873	5.013
iris	7.652	7.310	9.637
led	21.238	16.132	20.390
phoneme	52.021	44.684	53.188
tic-tac-toe	6.361	3.146	9.672
wine	9.969	8.604	9.896
wisconsin	29.662	24.911	32.278

Tabla 3.20: Resultados obtenidos para la medida *relevancia*.

conjunto de datos	DEQAR-SD	APRIORI-SD	CN2-SD
appendicitis	4.90	10.00	8.80
australian	2.20	10.00	12.20
balance	10.00	7.80	10.10
bridges	5.10	8.10	7.30
cleveland	12.80	5.00	16.10
contraceptive	8.10	8.00	17.00
diabetes	6.10	10.00	11.10
echo	7.20	5.50	9.10
ecoli	12.10	20.50	16.60
haberman	5.00	2.00	2.00
hayes-roth	8.50	5.00	4.00
heart	5.70	10.00	8.80
iris	3.10	15.00	3.00
led	20.10	40.40	25.70
phoneme	9.50	10.00	11.20
tic-tac-toe	2.00	6.00	15.30
wine	4.60	15.00	5.00
wisconsin	2.20	10.00	3.30

Tabla 3.21: Resultados obtenidos para la medida *número de reglas*.

conjunto de datos	DEQAR-SD	APRIORI-SD	CN2-SD
appendicitis	1.38	1.20	1.81
australian	1.07	1.60	2.58
balance	1.75	1.13	1.72
bridges	1.55	1.41	2.18
cleveland	1.95	1.36	2.12
contraceptive	1.71	2.25	2.35
diabetes	1.72	1.46	2.15
echo	1.77	1.12	2.22
ecoli	1.93	1.88	3.43
haberman	1.47	1.00	1.00
hayes-roth	1.20	1.40	1.50
heart	1.61	1.40	1.73
iris	1.83	1.38	1.10
led	1.89	3.33	2.49
phoneme	1.75	1.40	3.28
tic-tac-toe	1.00	1.72	2.74
wine	1.98	1.69	2.46
wisconsin	2.00	1.00	2.97

Tabla 3.22: Resultados obtenidos para la medida *número de atributos*.

Para la medida relevancia, DEQAR-SD obtuvo unos resultados parejos a los de CN2-SD, obteniendo el mejor valor en 8 conjuntos de datos frente a los 9 en los que destacó CN2-SD. En cuanto al tamaño del conjunto de reglas, DEQAR-SD destacó frente a los otros dos algoritmos, obteniendo los conjuntos de reglas más reducidos en 11 de los 18 conjuntos de datos. Y en cuanto al número de atributos, Apriori-SD destacó en la mayoría de los conjuntos de datos, quedando DEQAR-SD en segundo lugar.

Con el fin de obtener una validación estadística de estos resultados, se realizaron test no paramétricos de comparación múltiple. Empleando el test de Friedman para cada una de las medidas de calidad, se calcularon los valores medios de ranking de cada método en cada medida (un menor valor implica una posición más destacada en el ranking). La Tabla 3.23 muestra dichos valores junto a los correspondientes para el test de Iman-Davenport. Pueden verse los buenos valores obtenidos por DEQAR-SD, destacando en las medidas cobertura y número de reglas. Los p -valores obtenidos, que se muestran en la Tabla 3.24, indican que la hipótesis nula (no hay diferencias entre los métodos) puede ser rechazada en todos los casos con un nivel de significación α de 0.05. Los subsiguientes tests post-hoc (Bergmann-Hommel) devolvieron como resultado los p -valores ajustados que se muestran en la Tabla 3.25.

A la vista de la evaluación estadística realizada, puede decirse lo siguiente:

- En la medida cobertura, DEQAR-SD superó a los otros dos métodos, y no se observan diferencias significativas entre APRIORI-SD y CN2-SD.
- En la medida relevancia, APRIORI-SD fue superado por los otros dos métodos, y no se observan diferencias significativas entre DEQAR-SD y CN2-SD.
- En cuanto al número de reglas, DEQAR-SD superó a los otros dos métodos, y no se observan diferencias significativas entre APRIORI-SD y CN2-SD.
- Y respecto al número de atributos, CN2-SD fue superado por los otros dos métodos, y no se observan diferencias significativas entre DEQAR-SD y APRIORI-SD.

Como resumen de todas las pruebas comparativas realizadas, puede concluirse que DEQAR-SD consiguió muy buenos resultados, no habiendo sido superado por ninguno de los otros dos métodos en ninguna comparativa, superando a ambos en las medidas de cobertura y número de reglas, superando a APRIORI-SD en relevancia y a CN2-SD en cuanto a número de variables. El intentar conseguir buenos resultados en varias medidas simultáneamente, empleando una única función de bondad (en este caso la suma de sensibilidad y especificidad), puede no ser siempre lo más adecuado, aunque en el estudio realizado se consiguieron resultados aceptables con el método adoptado.

método	cobertura	relevancia	# reglas	# atrib.
DEQAR-SD	1.44	1.72	1.50	1.89
APRIORI-SD	2.22	2.56	2.19	1.42
CN2-SD	2.33	1.72	2.31	2.69

Tabla 3.23: Valores medios de ranking obtenidos por cada método en cada medida de calidad.

medida	Friedman	Iman-Davenport
cobertura	0.0147	0.0106
relevancia	0.0155	0.0114
# reglas	0.0324	0.0275
# atributos	0.0005	0.0001

Tabla 3.24: p-valores de los test realizados.

pares de métodos	cobertura	relevancia	# reglas	# atrib.
DEQAR-SD vs. APRIORI-SD	0.023	0.037	0.047	0.157
DEQAR-SD vs. CN2-SD	0.023	1	0.047	0.016
APRIORI-SD vs. CN2-SD	0.739	0.037	0.739	0.000

Tabla 3.25: p-valores ajustados de los tests post-hoc.

3.2.3. Comparativa con métodos no deterministas

Con el objetivo de ampliar el alcance de la efectividad del método propuesto en esta tesis, se ha realizado una comparativa con un conjunto de algoritmos que no pertenecen a la categoría de métodos deterministas. Concretamente, la comparativa se ha llevado a cabo con los algoritmos evolutivos *NMEEF-SD* [Carmona et al., 2010], *SDIGA* [del Jesus et al., 2007], *CGBA-SD* [Luna et al., 2014] y *FuGePSD* [Carmona et al., 2015]. Estos algoritmos realizan el proceso de búsqueda en base a los postulados de la evolución biológica, manteniendo posibles soluciones a un problema y haciendo que se mezclen y compitan entre sí, evolucionando hacia mejores soluciones cada vez. Tienen un componente estocástico, al emplear operadores probabilísticos.

Los conjuntos de datos empleados han sido los mismos que en [Carmona et al., 2015], es decir, 17 conjuntos de datos tomados del repositorio UCI. Las características de estos conjuntos de datos se muestran en la Tabla 3.26, pudiéndose observar que todos ellos tienen múltiples atributos numéricos. Asimismo, las medidas de calidad analizadas fueron las mismas que en [Carmona et al., 2015]: atipicidad, sensibilidad y confianza. Los valores de los parámetros empleados para las ejecuciones de DEQAR-SD fueron: $minsens = (0.60, 0.65, 0.70, 0.75)$, $delta = (0.20, 0.25)$, $maxAtr = 2$, $tamañoRanking = 3$ y $mincub = 0.95$. Con el objetivo de encontrar reglas con buenos valores en las medidas de calidad analizadas, se optó por una combinación de atipicidad y confianza (su producto), como medida de valoración de las reglas para su inserción en el ranking.

La evaluación se llevó a cabo siguiendo un proceso de validación cruzada de orden 10. Los resultados obtenidos para cada medida y conjunto de datos se muestran en las Tablas 3.27-3.29. Puede observarse que DEQAR-SD obtuvo buenos resultados, principalmente en la atipicidad.

Para validar estadísticamente los resultados, se empleó el test de Friedman de comparación múltiple para cada una de las medidas de calidad. Los valores medios de ranking de cada método en cada medida se muestran en la Tabla 3.30.

Puede verse que DEQAR-SD obtuvo el mejor resultado en atipicidad y el segundo mejor en cuanto a la confianza. La significancia estadística de estos tests, asociada a sus p-valores, se muestra en la Tabla 3.31. Se observa que, en la medida sensibilidad no puede rechazarse la hipótesis nula (igualdad entre los métodos) con un nivel de significación α de 0.05.

conjunto de datos	# atributos	# numéricos	# clases	# instancias
appendicitis	7	7	2	106
australian	14	6	2	690
balance	4	4	3	625
bridges	7	3	2	102
cleveland	13	13	5	303
diabetes	8	8	2	768
echo	6	5	2	131
german	20	7	2	1000
glass	9	9	6	214
haberman	3	3	2	306
heart	13	7	2	270
hepatitis	19	6	2	155
ionosphere	34	34	2	351
iris	4	4	3	150
led	7	7	10	500
vehicle	18	18	4	846
wine	13	13	3	178

Tabla 3.26: Características de los conjuntos de datos usados en los experimentos.

conjunto de datos	NMEEF-SD	SDIGA	CGBA-SD	FuGePSD	DEQAR-SD
appendicitis	0.0945	0.0966	0.0640	0.0445	0.0922
australian	0.1791	0.0525	0.1770	0.1619	0.1788
balance	0.0710	0.0655	0.0720	0.0815	0.1059
bridges	0.0446	0.0322	0.0290	0.0517	0.0425
cleveland	0.1300	0.0155	0.1080	0.1153	0.1175
diabetes	0.0859	0.0651	0.0720	0.0822	0.0905
echo	0.0390	0.0333	0.0320	0.0677	0.0734
german	0.0669	0.0020	0.0690	0.0323	0.0368
glass	0.0824	0.0176	0.0700	0.0995	0.0947
haberman	0.0499	0.0423	0.0230	0.0394	0.0566
heart	0.1072	0.0586	0.0820	0.1318	0.1340
hepatitis	0.0651	0.0352	0.0560	0.0767	0.0835
ionosphere	0.1413	0.0287	0.0310	0.1888	0.1785
iris	0.2067	0.1693	0.1050	0.2098	0.2178
led	0.0658	0.0593	0.2040	0.0769	0.0646
vehicle	0.0000	0.0240	0.0670	0.0930	0.1475
wine	0.1448	0.0862	0.0640	0.1830	0.2033

Tabla 3.27: Resultados para la medida atipicidad.

conjunto de datos	NMEEF-SD	SDIGA	CGBA-SD	FuGePSD	DEQAR-SD
appendicitis	1.0000	0.9762	0.6670	0.9263	0.8917
australian	0.7989	0.8736	0.8550	0.8004	0.7992
balance	0.5259	0.5839	0.6480	0.6300	0.6053
bridges	0.8787	0.7076	0.6710	0.7231	0.6271
cleveland	0.7763	0.3577	0.7210	0.7701	0.6210
diabetes	0.9180	0.9812	0.5570	0.7038	0.7827
echo	0.8582	0.9834	0.4150	0.8090	0.6310
german	0.4845	0.8408	0.4830	0.7911	0.6862
glass	0.8083	0.9186	0.5890	0.8931	0.8500
haberman	0.9334	0.8370	0.6610	0.9190	0.6627
heart	0.7175	0.9706	0.5410	0.7441	0.7467
hepatitis	0.8059	0.5845	0.6900	0.7425	0.7568
ionosphere	0.9701	0.8108	0.8480	0.9524	0.9336
iris	1.0000	0.9896	0.6720	0.9867	0.9800
led	0.8006	0.8172	0.9530	0.8548	0.8132
vehicle	0.0000	0.5962	0.8400	0.7079	0.8764
wine	0.9190	0.9071	0.4220	0.8819	0.9667

Tabla 3.28: Resultados para la medida sensibilidad.

conjunto de datos	NMEEF-SD	SDIGA	CGBA-SD	FuGePSD	DEQAR-SD
appendicitis	0.9038	0.7104	0.7170	0.9529	0.9234
australian	0.9320	0.6974	0.8580	0.8885	0.9313
balance	0.6981	0.5818	0.6180	0.6728	0.7474
bridges	0.9137	0.7345	0.7230	0.9476	0.9279
cleveland	0.8232	0.7442	0.7540	0.8138	0.8531
diabetes	0.7896	0.6226	0.7930	0.8705	0.7948
echo	0.7614	0.5705	0.6380	0.8366	0.8239
german	0.8767	0.5677	0.8800	0.7434	0.7578
glass	0.8646	0.5501	0.6130	0.9723	0.8383
haberman	0.8030	0.6291	0.6420	0.7974	0.8198
heart	0.7650	0.5778	0.7560	0.8729	0.8302
hepatitis	0.8798	0.8269	0.6590	0.9498	0.9323
ionosphere	0.8663	0.5562	0.9050	0.9679	0.9141
iris	0.9914	0.8990	0.8510	0.9825	1.0000
led	0.6494	0.4948	0.9430	0.8255	0.6258
vehicle	0.0000	0.3070	0.6530	0.7903	0.7484
wine	0.8874	0.8927	0.6390	0.9823	0.9250

Tabla 3.29: Resultados para la medida confianza.

algoritmo	atipicidad	sensibilidad	confianza
NMEEF-SD	2.65	2.65	2.82
SDIGA	4.35	2.65	4.65
CGBA-SD	3.76	3.88	3.71
FuGePSD	2.47	2.65	1.82
DEQAR-SD	1.76	3.18	2.00

Tabla 3.30: Valores medios de ranking de los tests de Friedman.

medida	Friedman	Iman-Davenport
atipicidad	0.0000060	0.0000002
sensibilidad	0.0898695	0.0850551
confianza	0.0000001	0.0000000

Tabla 3.31: p-valores de los tests realizados.

Los correspondientes tests post-hoc de Holm, incluido el de la medida sensibilidad, se muestran en las Tablas 3.32-3.34. Analizando los resultados de estos tests, se ve que DEQAR-SD consiguió significativamente mejores resultados que SDIGA y CGBA en la medida atipicidad, y sin diferencias significativas con el resto de métodos. Con respecto a la sensibilidad, no se encontraron diferencias significativas entre ningún par de métodos. Y en relación con la confianza, también DEQAR-SD consiguió mejores resultados que SDIGA y CGBA, y sin diferencias con los restantes.

comparativa	p
DEQAR-SD vs. SDIGA	0.00002
DEQAR-SD vs. CGBA	0.00204
SDIGA vs. FuGePSD	0.00415
NMEEF vs. SDIGA	0.01161
CGBA vs. FuGePSD	0.10213
NMEEF vs. CGBA	0.19659
DEQAR-SD vs. NMEEF	0.41497
DEQAR-SD vs. FuGePSD	0.57918
SDIGA vs. CGBA	0.57918
NMEEF vs. FuGePSD	0.74488

Tabla 3.32: p-valores ajustados del test de Holm para la atipicidad.

comparativa	p
NMEEF vs. CGBA	0.22740
SDIGA vs. CGBA	0.22740
CGBA vs. FuGePSD	0.22740
DEQAR-SD vs. CGBA	1.35141
DEQAR-SD vs. NMEEF	1.97383
DEQAR-SD vs. SDIGA	1.97383
DEQAR-SD vs. FuGePSD	1.97383
NMEEF vs. SDIGA	3.00000
NMEEF vs. FuGePSD	3.00000
SDIGA vs. FuGePSD	3.00000

Tabla 3.33: p-valores ajustados del test de Holm para la sensibilidad.

comparativa	p
SDIGA vs. FuGePSD	0.00000
DEQAR-SD vs. SDIGA	0.00001
CGBA vs. FuGePSD	0.00415
NMEEF vs. SDIGA	0.00541
DEQAR-SD vs. CGBA	0.00995
NMEEF vs. FuGePSD	0.32598
SDIGA vs. CGBA	0.33065
NMEEF vs. CGBA	0.33065
DEQAR-SD vs. NMEEF	0.33065
DEQAR-SD vs. FuGePSD	0.74488

Tabla 3.34: p-valores ajustados del test de Holm para la confianza.

Puede observarse que el método FuGePSD obtuvo también buenos resultados en las medidas atipicidad y confianza. Para comparar directamente DEQAR-SD y FuGePSD, puede emplearse el test de Wilcoxon en cada una de las medidas. Los resultados de este test se muestran en la Tabla 3.35. Se observa que en la medida atipicidad puede rechazarse la hipótesis nula, por lo que puede afirmarse que DEQAR-SD obtuvo resultados significativamente mejores que FuGePSD en dicha medida.

comparativa	R+	R-	p-valor	hipótesis nula
atipicidad	119	34	0.0448	rechazada por DEQAR-SD
sensibilidad	42	111	0.1089	no rechazada
confianza	44.5	108.5	0.1389	no rechazada

Tabla 3.35: Resultados del test de Wilcoxon para DEQAR-SD vs. FuGePSD.

Como resumen de las comparativas realizadas, puede decirse que el método propuesto (DEQAR-SD) no fue superado significativamente por ninguno de los restantes métodos en ninguna de las medidas analizadas, y presentó mejoras significativas en las medidas atipicidad y confianza, principalmente frente a los métodos SDIGA y CGBA. Para este estudio experimental, la función de selección de reglas empleada en DEQAR-SD (el producto entre atipicidad y confianza) propició obtener unos resultados aceptables, aunque queda patente la dificultad de conseguir simultáneamente buenos resultados en varias medidas mediante el empleo de una función que las combine.

3.3. Clasificación (DEQAR-CLAS)

3.3.1. Descripción del método

Siguiendo con los objetivos de esta tesis, también se ha realizado una adaptación del método DEQAR para llevar a cabo tareas de clasificación. El método propuesto, DEQAR-CLAS, es un clasificador basado en reglas que, al estilo de otros clasificadores basados en reglas de asociación [Liu et al., 1998] [Li et al., 2001], busca reglas que asocien condiciones con el atributo clase y que puedan ser de utilidad para la predicción de su valor. DEQAR-CLAS, tras el proceso de entrenamiento, obtiene una selección de reglas y una clase por defecto. Dicho proceso de entrenamiento se divide en dos fases: la "fase de generación de reglas" y la "fase de selección de reglas", tal como se muestra en la Figura 3.18.

Para generar las reglas en la primera fase, DEQAR-CLAS emplea un esquema similar al desarrollado en DEQAR-SD. De igual forma que en dicho método, se ha optado por no emplear intervalos para las condiciones de los atributos numéricos, y estas serán por tanto de las formas $A \leq v$ o $A \geq v$, siendo v un valor existente en los datos para el atributo A . Tras esta fase de generación de reglas, se obtiene una lista ordenada de reglas (ranking), para cada una de las clases del conjunto de datos de entrenamiento. Estas listas de reglas se mantienen de forma separada y se ordenan en función de los valores de confianza y soporte.

En la segunda fase, se procede a seleccionar las reglas que formarán parte del clasificador. El Algoritmo 3 muestra una descripción de dicho proceso. Con objeto de seleccionar reglas de las distintas clases, se comparan sucesivamente reglas de cada ranking, inicialmente la mejor de cada uno de ellos (líneas 5-7). Para ello, se procede a calcular el número de instancias cubiertas por cada regla (línea 8) y el número de instancias correctamente clasificadas (línea 9). Si la regla clasifica correctamente al menos una instancia (línea 10), se calcula la confianza relativa a dicha regla, comparándose con los correspondientes valores de confianza de las reglas de las otras clases (línea 11). Se guardan la regla y la clase con el mayor valor de confianza obtenido para las reglas comparadas en cada clase (líneas 12-13). El proceso finaliza cuando todas las reglas de los rankings han sido procesadas (líneas 22-24). Finalmente, la clase por defecto se establece como la clase con el mayor número de instancias no cubiertas por ninguna de las reglas seleccionadas, o bien, si se han cubierto todas, la clase mayoritaria en los datos de entrenamiento (líneas 31-35).

Una vez obtenido el clasificador, el proceso de clasificación de una instancia se describe en el Algoritmo 4. Se recorre la lista de reglas, buscando la primera de ellas en la que la instancia cumpla su antecedente, y se le asigna la clase de dicha regla (líneas 2-8). En caso de que ninguna regla pueda aplicarse a la instancia, se le asigna la clase por defecto (líneas 9-11).

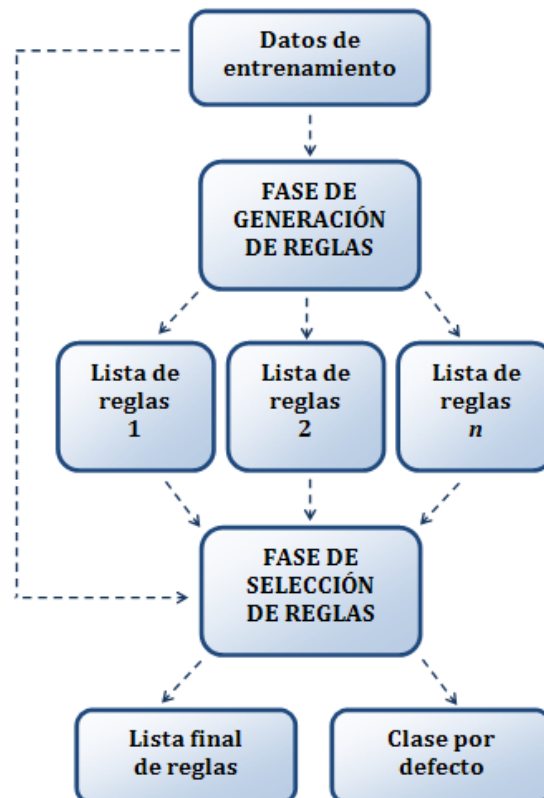


Figura 3.18: Proceso de entrenamiento en DEQAR-CLAS.

3.3.2. Aplicación a un conjunto de datos médicos

El algoritmo DEQAR-CLAS se ha empleado en un conjunto de datos relativos al tratamiento de pacientes con cáncer de próstata, para obtener un modelo predictivo del grado de toxicidad genitourinaria (GU) asociado al tratamiento con radioterapia [Lopez et al., 2015]. Dichos datos incluyen cierta información clínica, patológica y terapéutica, así como la toxicidad resultante en 162 pacientes con cáncer de próstata tratados con terapia de radiación desde junio de 2006 a mayo de 2012 en dos centros de Europa y América Latina.

En la Tabla 3.36 se muestran las características de las 17 variables utilizadas en el conjunto de datos. Puede observarse que hay un total de 10 variables de tipo numérico y que la clase es de tipo binario ('+' para denotar una toxicidad de grado ≥ 2 , '-' para una toxicidad de grado < 2), siendo 38 el número de casos de clase '+' (23.5% del total).

Algoritmo 3 Selección de reglas en DEQAR-CLAS

Entrada: Datos de entrenamiento, rankings de reglas

Salida: Lista final de reglas, clase por defecto

```

1:  $T \leftarrow$  datos de entrenamiento
2:  $parar \leftarrow falso$ 
3: mientras (NO  $parar$ ) hacer
4:    $maxConf = 0$ 
5:   para cada clase  $c$  hacer
6:     si (el ranking  $c$  tiene reglas por procesar) entonces
7:        $R \leftarrow$  siguiente regla del ranking  $c$ 
8:        $nAnt \leftarrow$  número de instancias de  $T$  cubiertas por  $R$ 
9:        $n \leftarrow$  número de instancias de  $T$  bien clasificadas por  $R$ 
10:      si ( $n > 0$ ) entonces
11:        si ( $n/nAnt > maxConf$ ) entonces
12:           $maxR \leftarrow R$ 
13:           $maxC \leftarrow c$ 
14:           $maxConf \leftarrow n/nAnt$ 
15:        fin si
16:      en caso contrario
17:        pasar a la siguiente regla en el ranking  $c$ 
18:      fin si
19:    fin si
20:  fin para
21:  si ( $maxConf = 0$ ) entonces
22:    si (se han procesado todas las reglas) entonces
23:       $parar \leftarrow verdadero$ 
24:    fin si
25:  en caso contrario
26:    añadir la regla  $maxR$  a la lista final de reglas
27:     $T \leftarrow T -$  instancias cubiertas por la regla  $maxR$ 
28:    pasar a la siguiente regla en el ranking  $maxC$ 
29:  fin si
30: fin mientras
31: si ( $T \neq \emptyset$ ) entonces
32:    $claseDefecto \leftarrow$  clase mayoritaria en  $T$ 
33: en caso contrario
34:    $claseDefecto \leftarrow$  clase mayoritaria en los datos de entrenamiento
35: fin si
36: devolver lista final de reglas y  $claseDefecto$ 

```

Algoritmo 4 Proceso de clasificación en DEQAR-CLAS**Entrada:** Lista de reglas, clase por defecto, una instancia**Salida:** Clase predicha

```

1: cubierta ← falso
2: mientras (NO cubierta Y hay reglas para procesar) hacer
3:   R ← siguiente regla de la lista
4:   si (la instancia cumple el antecedente de R) entonces
5:     clasePredicha ← clase de la regla R
6:     cubierta ← verdadero
7:   fin si
8: fin mientras
9: si (NO cubierta) entonces
10:  clasePredicha ← clase por defecto
11: fin si
12: devolver clasePredicha

```

atributo	unidad/valor
Age	Years
Indication treatment	Post-prostatectomy , Primary Prostate Cancer, Recurrence
Radiation technique	Tomotherapy, Rapid Arc
Gleason score	2..10
T stage	T1 , T1b , T1c , T2 , T2a , T2b , T2c , T3 , T3a , T3b , T4
Diagnosis PSA	ng/mL
Risk	Low, Intermediate, High
ADT	No ADT, Short Term, Long Term
Radiation time	Days
Planning tumor volume	Cc
Prostate radiation dose	Gy
Fractionation	Gy
Pelvic treatment	Yes, No
Bladder volume	cc
Bladder mean dose	Gy
Bladder median dose	Gy
GU acute toxicity	+, -

Tabla 3.36: Atributos del conjunto de datos.

En primer lugar, para evaluar los resultados del método propuesto en la predicción de la toxicidad GU, se ha realizado una comparativa con un conjunto de clasificadores. Se han seleccionado clasificadores de distintas familias: métodos basados en reglas (*ZeroR*, *PART*), redes neuronales (*Multilayer Perceptron*), árboles de decisión (*J48*, *Random Forest*), bayesianos (*Bayes Net*, *Naive Bayes*) y regresión logística (*Logistic*). Una descripción de dichos métodos está expuesta en la Sección 1.3.

La implementación de los métodos utilizados en la comparativa ha sido la proporcionada por el paquete *Weka* [Hall et al., 2009]. Se han calculado diferentes medidas para evaluar la bondad de la clasificación, empleando validación cruzada estratificada de 10 iteraciones. Los valores empleados para los parámetros fueron los que mejores resultados proporcionaron en cada clasificador, tras experimentar con diversas combinaciones para cada uno de ellos.

Para evaluar la clasificación en su conjunto, se han calculado diversas medidas: tasa de acierto, el índice kappa, el coeficiente de correlación de Matthews, y el valor medio de sensibilidad y especificidad. La Tabla 3.37 muestra los resultados para estas medidas, y como puede observarse, el clasificador DEQAR-CLAS obtuvo los mejores valores en todas ellas. Aunque igualado con RandomForest en la tasa de acierto, la diferencia en las otras medidas revela una mejor clasificación para el conjunto de las dos clases.

método	tasa de acierto	kappa	MCC	media(sens, espec)
ZeroR	0.765	0.000	0.000	0.500
PART	0.710	0.185	0.185	0.592
MultilayerPerceptron	0.710	0.199	0.199	0.601
J48	0.698	0.117	0.118	0.556
RandomForest	0.772	0.135	0.184	0.550
BayesNet	0.710	0.185	0.185	0.592
NaiveBayes	0.698	0.208	0.210	0.611
Logistic	0.716	0.130	0.132	0.559
DEQAR-CLAS	0.772	0.358	0.358	0.677

Tabla 3.37: Resultados en varias medidas de evaluación.

En cada clase en particular, se evaluaron los resultados según las medidas precisión, exhaustividad y medida-F. En las Tablas 3.38 y 3.39 pueden verse los resultados obtenidos. Para la clase '+', la clase minoritaria, ningún clasificador obtuvo una alta precisión, lo que indica la dificultad de clasificar los casos de alta toxicidad en este conjunto de datos. Aún así, DEQAR-CLAS fue el único método junto con RandomForest que mostró una precisión mayor a 0.5 para esta clase. Sin embargo, RandomForest obtuvo una exhaustividad mucho menor. En cuanto a la medida-F, en ella destacó DEQAR-CLAS, y NaiveBayes fue el segundo mejor método.

método	precisión	exhaustividad	medida-F
ZeroR	0.000	0.000	0.000
PART	0.378	0.368	0.373
MultilayerPerceptron	0.385	0.395	0.390
J48	0.333	0.289	0.310
RandomForest	0.556	0.132	0.213
BayesNet	0.378	0.368	0.373
NaiveBayes	0.378	0.447	0.410
Logistic	0.357	0.263	0.303
DEQAR-CLAS	0.514	0.500	0.507

Tabla 3.38: Resultados para la clase "+".

Para la clase '-', DEQAR-CLAS mostró la mayor precisión, y fue el tercer método con mejor exhaustividad, por detrás de ZeroR y RandomForest. Obsérvese que ZeroR obtuvo una exhaustividad del 100%, puesto que siempre predice dicha clase mayoritaria. Con respecto a la medida-F, DEQAR-CLAS fue el tercer mejor método.

En resumen, DEQAR-CLAS presentó unos muy buenos resultados en este conjunto de datos, obteniendo excelentes valores, por ejemplo, para la tasa de acierto y la medida-F de la clase '+'. El buen resultado global fue debido, principalmente, al buen comportamiento en la clasificación de los casos de toxicidad alta, la clase minoritaria.

método	precisión	exhaustividad	medida-F
ZeroR	0.765	1.000	0.867
PART	0.808	0.815	0.811
MultilayerPerceptron	0.813	0.806	0.810
J48	0.791	0.823	0.806
RandomForest	0.784	0.968	0.866
BayesNet	0.808	0.815	0.811
NaiveBayes	0.821	0.774	0.797
Logistic	0.791	0.855	0.822
DEQAR-CLAS	0.848	0.855	0.851

Tabla 3.39: Resultados para la clase "-".

Los valores de los parámetros empleados en DEQAR-CLAS para obtener dichos resultados fueron los siguientes: $minsens = 0.7$, $tamañoRanking = 1$, $delta = 0.05$ y $maxAtr = 2$. Si $tamañoRanking$ es 1, en la lista de reglas para cada clase se guarda simplemente la mejor regla encontrada para dicha clase, por lo que el clasificador final tendrá una lista formada por dos reglas (en el caso de clases

binarias). Se han calculado los resultados variando dicho parámetro *tamañoRanking*, y se muestran en la Tabla 3.40. Puede verse cómo empeoran las medidas al aumentar el valor de este parámetro. Esto puede deberse, para este conjunto de datos, a que dada la dificultad en la clasificación de la clase minoritaria, la incorporación de más reglas para esta clase da a lugar a unos peores resultados en la clasificación global, por la baja precisión de dichas reglas.

tamañoRanking	tasa de acierto	kappa	MCC	media(sens, espec)
10	0.728	0.295	0.298	0.658
5	0.728	0.307	0.311	0.667
3	0.741	0.315	0.317	0.666
2	0.747	0.326	0.327	0.670
1	0.772	0.358	0.358	0.677

Tabla 3.40: Resultados para varios valores del parámetro *tamañoRanking*.

Una vez realizado el estudio comparativo entre los diversos clasificadores empleando validación cruzada, se utilizó el conjunto completo de los datos para obtener un modelo predictivo para la toxicidad GU. Para ello se ejecutó DEQAR-CLAS con los mismos parámetros con los que se habían obtenido los mejores resultados en validación cruzada. Como resultado, se seleccionaron dos reglas y se estableció la clase '-' como clase por defecto. En la Tabla 3.41 se muestran las reglas obtenidas. Los atributos y valores asociados en dichas reglas pueden considerarse factores de riesgo para la toxicidad GU. Estos factores de riesgo están en su mayoría confirmados por la literatura [Ahmed et al., 2013] [Lopez et al., 2013], lo que puede corroborar el valor del método empleado.

regla	confianza	soporte
Radiation technique = Tomotherapy Y		
Planning tumor volume $\leq 218.62 \rightarrow$ Toxicity = -	0.882	0.556
Prostate radiation dose ≥ 70.02 Y		
Bladder volume $\geq 63.67 \rightarrow$ Toxicity = +	0.329	0.167

Tabla 3.41: Reglas en el clasificador final.

Puede observarse el bajo valor de confianza de la segunda regla, relativa a la predicción de la clase '+', la clase minoritaria. Aunque, dado que en los datos la prevalencia de la alta toxicidad es del 23.5%, su valor de *lift* es de 1.4. Pero puesto que dicha regla se considera cuando no se cumple el antecedente de la

primera (en cuyo caso se predeciría la clase '-'), realmente su antecedente sería "NO (*Radiation technique = Tomotherapy* Y *Planning tumor volume* ≤ 218.62) Y *Prostate radiation dose* ≥ 70.02 Y *Bladder volume* ≥ 63.67 ". Entonces, su precisión, y también la precisión del clasificador para los casos de toxicidad alta, resulta ser 0.512, similar a la precisión obtenida en validación cruzada.

Cabe destacar, aparte de los buenos resultados obtenidos por DEQAR-CLAS en las medidas evaluadas, la simplicidad del modelo generado. Utiliza únicamente dos reglas con dos atributos en su antecedente, más la clase por defecto. Si se compara con otros clasificadores, su simplicidad es evidente. Por ejemplo, el clasificador RandomForest obtuvo los resultados expuestos mediante la generación de 2000 árboles. En la Figura 3.19 se muestra uno de los más simples, y como puede comprobarse, no resulta tan fácil el analizar qué atributos y qué condiciones pueden tener mayor influencia en la clasificación.

```

RandomTree
=====

Radiation technique = ArcTherapy
|  Planning tumor volume < 44.5 : - (13/0)
|  Planning tumor volume >= 44.5
|  |  Radiation time < 63.5
|  |  |  Age < 69.5 : + (15/0)
|  |  |  Age >= 69.5
|  |  |  |  Radiation time < 61
|  |  |  |  |  Age < 77 : + (4/0)
|  |  |  |  |  Age >= 77 : - (1/0)
|  |  |  |  |  Radiation time >= 61 : - (7/0)
|  |  |  Radiation time >= 63.5
|  |  |  |  Indication treatment = Post-prostatectomy : + (1/0)
|  |  |  |  Indication treatment = PrimaryProstateCancer
|  |  |  |  |  Age < 76.5 : - (9/0)
|  |  |  |  |  Age >= 76.5
|  |  |  |  |  |  Gleason score < 7.5 : - (2/0)
|  |  |  |  |  |  Gleason score >= 7.5 : + (1/0)
|  |  |  |  |  Indication treatment = Recurrence : + (0/0)
Radiation technique = Tomotherapy
|  Age < 62.5
|  |  Radiation time < 41.5
|  |  |  Diagnosis PSA < 4.75 : - (2/0)
|  |  |  Diagnosis PSA >= 4.75 : + (4/0)
|  |  Radiation time >= 41.5 : - (15/0)
|  Age >= 62.5
|  |  Radiation time < 54.5 : - (57/0)
|  |  Radiation time >= 54.5
|  |  |  Radiation time < 57.5
|  |  |  |  Gleason score < 6.5 : + (4/0)
|  |  |  |  Gleason score >= 6.5 : - (6/0)
|  |  |  Radiation time >= 57.5 : - (12/0)

```

Figura 3.19: Uno de los árboles resultantes en el clasificador RandomForest.

4. Conclusiones

En esta sección se resumen los resultados obtenidos en los trabajos expuestos en esta memoria. Además, se detallan las publicaciones en revistas y congresos internacionales derivadas de dichos trabajos, así como posibles trabajos futuros que pueden surgir de las ideas expuestas.

4.1. Resultados obtenidos

En esta tesis se han llevado a cabo una serie de trabajos de investigación, fundamentados en la hipótesis inicial de que, en problemas de minería de datos, al obtener reglas sin realizar una discretización previa de los atributos numéricos, puede conseguirse un aumento en la calidad de las reglas generadas, al evitarse la posible pérdida de información asociada al proceso de discretización. Se ha optado por abordar el problema mediante un enfoque determinista exhaustivo, generando reglas directamente según las diversas combinaciones de posibles intervalos para los atributos numéricos.

Se han diseñado e implementado nuevos algoritmos y estructuras de datos, necesarios para una correcta y rápida generación de condiciones para los atributos numéricos que forman las reglas. Dada la inevitable explosión combinatoria ligada al enfoque exhaustivo propuesto, se han definido y empleado los oportunos parámetros en los algoritmos desarrollados, para tratar de conseguir un equilibrio entre el tiempo de cómputo y la calidad de las reglas generadas.

4.1.1. Reglas de asociación

Para la tarea de búsqueda de reglas de asociación, el algoritmo desarrollado, DEQAR, se aplica en conjuntos de datos que pueden contener atributos de tipo tanto categórico como numérico. En la Sección 3.1 se ha descrito con detalle dicho algoritmo. Realiza una búsqueda de reglas válidas, que satisfagan los umbrales mínimos de soporte y confianza, en base a generar los posibles intervalos para los atributos numéricos del conjunto de datos. Los extremos de los intervalos deben ser valores existentes en el conjunto de datos para cada atributo, y emplea un cambio de dominio, operando con números naturales asociados con el orden ascendente de los valores en cada atributo. Para facilitar

esta tarea, se emplean unas tablas auxiliares, que almacenan dicha ordenación de cada atributo y la relación con cada fila en el conjunto de datos inicial. El método emplea una búsqueda en profundidad, recorriendo sucesivamente los atributos del primero al último, generando secuencias de intervalos válidos y con vuelta atrás cuando ya no pueden reducirse más los intervalos por las restricciones debidas a los umbrales mínimos.

Las reglas obtenidas en este proceso se almacenan en función de alguna de las medidas de calidad habituales para reglas de asociación, dando lugar a una lista ordenada de reglas con un tamaño limitado por uno de los parámetros del algoritmo. Otros parámetros controlan la exhaustividad del proceso, limitando el número de atributos empleados en las reglas y el número de intervalos considerados en cada atributo numérico. Finalmente, se realiza un filtrado de la lista de reglas, que elimina posibles reglas similares generadas durante el proceso. Se realizaron distintos experimentos para comprobar empíricamente el efecto de la variación de dichos parámetros en el tiempo de procesamiento y la calidad de las reglas obtenidas. Pudo comprobarse que es posible, mediante el uso de dichos parámetros, controlar la exhaustividad del proceso sin perder una excesiva calidad en las reglas. Se observó que el número de atributos empleados en las reglas influye notablemente en el tiempo de cómputo, debido a su influencia en la combinatoria asociada con la búsqueda exhaustiva de reglas.

Se ha procedido también a comparar los resultados de DEQAR frente a los obtenidos por el algoritmo Apriori, el método exhaustivo de referencia para la generación de reglas de asociación. Dicho método requiere discretizar previamente los atributos numéricos, por lo que en las comparativas se hizo uso de las discretizaciones en igual-anchura y en igual-frecuencia, con distintos valores para el parámetro relativo al número de particiones. Se emplearon 16 conjuntos de datos de distintos repositorios, evaluando las reglas con diferentes medidas de calidad. Finalmente, se llevó a cabo una validación estadística de los resultados obtenidos, mediante el uso de tests no paramétricos.

DEQAR obtuvo unos excelentes resultados, no siendo superado por Apriori en ninguno de los estudios comparativos para distintas medidas de calidad de las reglas, y superando a Apriori en varias de dichas medidas. Tanto en DEQAR como en Apriori se probaron distintos valores para sus parámetros, y solamente DEQAR logró buenos resultados (y estadísticamente significativos) en el conjunto de comparativas. Concretamente, la variante que empleó valores de sus parámetros para realizar una búsqueda más exhaustiva y con pocos atributos, fue la que mejores resultados consiguió.

También se ha aplicado el método propuesto a dos conjuntos de datos reales, uno con información diversa sobre países subsaharianos (*Easterly and Levine*) y el otro relativo a datos médicos provenientes de un estudio sobre aterosclerosis (*STULONG*). En ambos casos, se consiguió generar una diversidad de reglas que asociaban distintos atributos y con buenos valores para las medidas de calidad evaluadas.

A la vista de los resultados de estos estudios experimentales, queda claro que con el algoritmo desarrollado y los adecuados valores de sus parámetros, puede llevarse a cabo una búsqueda de reglas de asociación lo suficientemente exhaustiva que permita obtener reglas simples y con buenos valores de bondad.

4.1.2. Descubrimiento de subgrupos

Se ha realizado una adaptación de DEQAR para la tarea de descubrimiento de subgrupos, desarrollando el algoritmo DEQAR-SD, expuesto en la Sección 3.2. Dicho algoritmo presenta algunas diferencias significativas con respecto al de reglas de asociación, siendo una de ellas el que genera reglas con un único atributo fijado en su consecuente, al ser el objetivo del descubrimiento de subgrupos el encontrar reglas para un atributo de interés de tipo categórico. Otra importante diferencia es que, para seleccionar las reglas se emplea la suma de sensibilidad y especificidad, para intentar obtener reglas con gran poder discriminatorio para el atributo objetivo (clase), al tenerse en cuenta tanto la tasa de aciertos para dicha clase como la correspondiente al resto de instancias que no pertenecen a ella.

Para comprobar el funcionamiento del método propuesto se realizó un estudio comparativo, empleando validación cruzada, entre DEQAR-SD y los métodos clásicos deterministas APRIORI-SD y CN2-SD, utilizando 18 conjuntos de datos provenientes principalmente del repositorio UCI. Se calcularon diversas medidas de calidad para evaluar la calidad de las reglas obtenidas en la comparativa, y se realizaron test no paramétricos para obtener una validación estadística de estos resultados.

Analizando las pruebas comparativas realizadas, puede concluirse que DEQAR-SD consiguió muy buenos resultados, no siendo superado por ninguno de los otros dos métodos en ninguna comparativa, y superando a ambos en algunas de las medidas evaluadas. En el proceso de selección de las mejores reglas, la utilización de una única función de bondad (en este caso la suma de sensibilidad y especificidad), permitió conseguir resultados aceptables en este estudio

comparativo, aunque se comprobó la dificultad de obtener los mejores resultados para varias medidas simultáneamente.

También se ha llevado a cabo una comparativa con un conjunto de algoritmos no deterministas, concretamente con los algoritmos evolutivos NMEEF-SD, SDIGA, CGBA-SD y FuGePSD. Se emplearon para estos experimentos 17 conjuntos de datos tomados del repositorio UCI, evaluando varias medidas de calidad. Como medida para la selección de reglas en DEQAR-SD, se utilizó el producto entre atipicidad y confianza. Tras la oportuna validación estadística de los resultados, se constató que DEQAR-SD consiguió mejores resultados que SDIGA y CGBA en dos de las medidas (atipicidad y confianza), y sin diferencias significativas con en el resto de métodos en el conjunto de medidas, salvo con FuGePSD en la medida atipicidad (donde DEQAR-SD obtuvo resultados ligeramente mejores).

Para este estudio experimental, la función de selección de reglas empleada en DEQAR-SD (el producto entre atipicidad y confianza) obtuvo unos resultados aceptables, aunque queda patente la dificultad de tratar de conseguir buenos resultados en varias medidas simultáneamente.

4.1.3. Clasificación

También se ha realizado una adaptación del método DEQAR para llevar a cabo tareas de clasificación, desarrollándose el algoritmo DEQAR-CLAS, un clasificador basado en reglas. Este algoritmo selecciona reglas que asocien condiciones con el atributo clase y que puedan ser de utilidad para la predicción de su valor, obteniendo, tras el proceso de entrenamiento, una lista de reglas y una clase por defecto.

En una primera fase, se generan reglas mediante un esquema similar al desarrollado en DEQAR-SD, aunque aquí se ordenan en función de sus valores de confianza y soporte, con objeto de tener un conjunto de reglas con alta precisión y que cubran muchos casos. En una segunda fase, se procede a seleccionar las reglas que formarán parte del clasificador, comparando sucesivamente las mejores reglas para cada valor de la clase y seleccionando las que mayor precisión tengan para los casos no cubiertos por reglas ya seleccionadas.

Dicho método clasificador se ha aplicado a datos relativos al tratamiento de pacientes con cáncer de próstata, para obtener un modelo predictivo del grado de toxicidad genitourinaria asociado al tratamiento con radioterapia. Se ha evaluado el método propuesto en su aplicación a este problema mediante una

comparativa, utilizando validación cruzada, con un conjunto de clasificadores de distintas familias: basados en reglas, redes neuronales, árboles de decisión, bayesianos y regresión logística.

Se emplearon como valores para los parámetros, aquellos que mejores resultados proporcionaron en cada clasificador, tras experimentar con diversas combinaciones para cada uno de ellos. Se evaluó el resultado de la clasificación en su conjunto, así como para cada una de las dos clases (alta y baja toxicidad), mediante varias medidas (tasa de acierto, índice kappa, coeficiente de correlación de Matthews, valor medio de sensibilidad y especificidad, precisión, exhaustividad y medida-F). DEQAR-CLAS obtuvo los mejores valores en todas las medidas relativas a la clasificación global. En la clase mayoritaria, obtuvo la mejor precisión, aunque sin llegar a la mejor exhaustividad; y en la clase minoritaria, alta toxicidad, consiguió los mejores valores de exhaustividad y medida-F. Por tanto, aparte del buen resultado en la clasificación global, es destacable el buen comportamiento obtenido en la clasificación de los casos de toxicidad alta, la clase minoritaria.

Por último, se utilizó el conjunto completo de datos para obtener un modelo predictivo para la toxicidad, dando como resultado dos reglas con dos atributos en su antecedente, y la clase mayoritaria como clase por defecto. Cabe destacar la simplicidad del modelo generado, que permite analizar fácilmente qué atributos y condiciones pueden tener gran influencia en la clasificación, y con ello servir de ayuda en un posible control de la toxicidad del tratamiento médico.

4.2. Publicaciones

Las ideas expuestas en esta tesis se han plasmado en las siguientes revistas y congresos internacionales:

- Domínguez-Olmedo, J.L., Mata, J., Pachón, V. y Maña, M.J. (2011). *A Deterministic Approach to Association Rule Mining without Attribute Discretization*, Proceedings of the International Conference on Digital Information Processing and Communications, CCIS 188, Springer. DOI: 10.1007/978-3-642-22389-1_13
- Domínguez-Olmedo, J.L., Mata, J., Pachón, V. y Maña, M.J. (2012). *Rule Extraction from Medical Data without Discretization of Numerical Attributes*, Proceedings of the International Conference on

Health Informatics, vol. 1, HEALTHINF, pp. 397-400. DOI: 10.5220/0003784603970400

- Domínguez-Olmedo, J.L., Mata Vázquez, J. y Pachón, V. (2015). *Deterministic Extraction of Compact Sets of Rules for Subgroup Discovery*, Proceedings of the International Conference on Intelligent Data Engineering and Automated Learning, IDEAL, LNCS vol. 9375, Springer. DOI: 10.1007/978-3-319-24834-9_17
- Domínguez-Olmedo, J.L. y Mata Vázquez, J. (2016). *Comparison of Standard Discretization with a New Method for Quantitative Association Rules*, IEEE Latin America Transactions, vol. 14, no. 4, pp. 1879-1885. DOI: 10.1109/TLA.2016.7483529
- Domínguez-Olmedo, J.L. y Mata Vázquez, J. (2017). *Obtaining Significant and Interpretable Rules for Subgroup Discovery Tasks*, IEEE Latin America Transactions, vol. 15, no. 10, pp. 2012-2016. DOI: 10.1109/TLA.2017.8071249
- Domínguez-Olmedo, J.L., Mata, J., Pachón, V. y Lopez-Guerra, J. L. (2018). *A Rule-based Method Applied to the Imbalanced Classification of Radiation Toxicity*, Proceedings of the 11th International Joint Conference on Biomedical Engineering Systems and Technologies, vol. 5, HEALTHINF, pp. 147-155. DOI: 10.5220/0006586401470155
- Domínguez-Olmedo, J.L., Mata, J., Pachón, V. y Lopez-Guerra, J. L. (aceptado para publicación). *Application of a Rule-based Classifier to Data regarding Radiation Toxicity in Prostate Cancer Treatment*, Communications in Computer and Information Science, Springer, BIOSTEC 2018 Revised Selected Papers.

4.3. Trabajos futuros

Con la experiencia obtenida por la investigación desarrollada en esta tesis, puede pensarse en algunas mejoras y trabajos futuros. Se exponen a continuación algunas ideas:

- Como se ha expuesto anteriormente, generalmente resulta difícil encontrar una función que permita, por sí sola, seleccionar las reglas que puedan tener buenos valores simultáneamente en un conjunto

de medidas. Sería deseable, por tanto, estudiar y adaptar algunos conceptos de optimización multiobjetivo.

- El algoritmo desarrollado para tareas de clasificación, aunque ha mostrado excelentes resultados en su aplicación al problema tratado, debería probarse en su aplicación a más conjuntos de datos y con estudios comparativos con otros clasificadores. Quizás el método de selección de reglas para formar el clasificador pueda mejorarse, o modificarse por un esquema de combinación de reglas que decidan conjuntamente el valor a predecir.
- También podría extenderse el enfoque determinista propuesto a la tarea de regresión, en la cual el atributo del consecuente no es categórico, sino numérico.
- La aplicación de los algoritmos expuestos a conjuntos de datos de gran volumen, aún con las mejoras desarrolladas para el control de la exhaustividad del proceso, resultaría en muchos casos inabordable. Se requiere por tanto el estudio de la posible aplicación de técnicas y heurísticas que, aún no encontrando posiblemente la totalidad de las mejores reglas, permitan dotar al proceso de una mayor escalabilidad.

Apéndice

A continuación, se muestran tablas con el detalle de los resultados experimentales descritos en la Sección 3.1.

método	#reglas	sop	conf	lift	cober	#atrAnt	#atrCon	#atrib
APRIORI-EF-2	22,028	27.4	100.0	2.00	45.1	5.1	1.0	6.1
APRIORI-EF-3	15,928	20.0	100.0	3.00	45.6	5.1	1.0	6.1
APRIORI-EF-4	10,571	15.1	100.0	4.00	38.4	5.0	1.0	6.0
APRIORI-EF-6	5,366	11.6	99.9	6.01	25.9	4.5	1.1	5.5
APRIORI-EF-8	740	10.5	98.2	7.83	12.8	3.0	1.2	4.2
APRIORI-EF-9	10	-	-	-	-	-	-	-
APRIORI-EW-2	24,658	32.2	100.0	1.04	68.7	3.0	1.5	4.5
APRIORI-EW-3	12,783	30.1	100.0	1.24	95.0	3.7	1.3	5.0
APRIORI-EW-4	9,899	27.9	100.0	1.20	98.8	2.7	1.1	3.8
APRIORI-EW-6	6,659	19.5	100.0	1.36	38.7	3.1	1.1	4.1
APRIORI-EW-8	1,487	14.8	99.9	2.81	37.4	3.3	1.2	4.5
APRIORI-EW-9	703	13.2	99.4	3.68	53.5	3.1	1.1	4.1
DEQAR-0.4-3-2	2,035	64.0	100.0	1.16	100.0	1.0	1.7	2.7
DEQAR-0.4-2-1	841	53.9	100.0	1.13	100.0	1.4	1.0	2.4
DEQAR-0.5-3-2	2,275	54.9	100.0	1.12	100.0	1.0	1.8	2.8
DEQAR-0.5-2-1	914	46.0	100.0	1.13	100.0	1.3	1.0	2.3

Tabla A.1: Resultados para el conjunto de datos *abalone* (mejores 100 reglas).

método	#reglas	sop	conf	lift	cober	#atrAnt	#atrCon	#atrib
APRIORI-EF-2	81,423,427	39.6	100.0	2.00	100.0	2.3	1.2	3.5
APRIORI-EF-3	886,017	24.7	100.0	3.01	100.0	2.3	1.2	3.5
APRIORI-EF-4	11,808	16.1	100.0	4.05	98.8	2.7	1.2	3.9
APRIORI-EF-6	151	13.9	98.2	5.94	100.0	1.7	1.2	2.9
APRIORI-EF-8	98	-	-	-	-	-	-	-
APRIORI-EF-9	100	10.7	97.0	8.81	100.0	1.2	1.2	2.5
APRIORI-EW-2	188,161,019	93.0	100.0	1.01	99.2	2.5	1.2	3.7
APRIORI-EW-3	17,535,317	63.1	100.0	1.07	97.6	2.6	1.1	3.7
APRIORI-EW-4	260,976	30.0	100.0	1.34	98.4	3.0	1.0	4.0
APRIORI-EW-6	5,348	12.8	100.0	2.40	86.5	3.5	1.0	4.5
APRIORI-EW-8	750	12.6	97.1	2.28	69.4	3.2	1.0	4.2
APRIORI-EW-9	177	12.6	88.4	2.37	87.7	2.4	1.0	3.4
DEQAR-0.4-3-2	3,232	63.9	100.0	1.38	100.0	1.0	1.9	2.9
DEQAR-0.4-2-1	2,582	63.9	100.0	1.09	100.0	1.0	1.0	2.0
DEQAR-0.5-3-2	3,135	54.8	100.0	1.62	100.0	1.0	2.0	3.0
DEQAR-0.5-2-1	2,710	54.8	100.0	1.14	100.0	1.0	1.0	2.0

Tabla A.2: Resultados para el conjunto de datos *bodyfat* (mejores 100 reglas).

método	#reglas	sop	conf	lift	cober	#atrAnt	#atrCon	#atrib
APRIORI-EF-2	2,310	25.9	100.0	1.98	100.0	3.3	1.2	4.5
APRIORI-EF-3	2,474	20.8	100.0	4.75	50.0	2.1	2.0	4.1
APRIORI-EF-4	1,637	20.1	100.0	4.93	47.5	2.0	2.0	4.0
APRIORI-EF-6	838	20.0	100.0	5.00	20.0	2.0	2.0	4.0
APRIORI-EF-8	467	20.0	100.0	5.00	20.0	2.0	2.0	4.0
APRIORI-EF-9	216	20.0	100.0	5.00	20.0	2.0	2.0	4.0
APRIORI-EW-2	2,903	47.8	100.0	1.29	80.0	2.3	1.4	3.7
APRIORI-EW-3	4,053	22.0	100.0	4.45	100.0	3.1	2.5	5.6
APRIORI-EW-4	3,904	21.4	100.0	4.68	92.5	3.2	2.6	5.8
APRIORI-EW-6	3,713	20.0	100.0	5.00	20.0	3.1	2.7	5.7
APRIORI-EW-8	1,873	20.0	100.0	5.00	20.0	3.1	2.7	5.7
APRIORI-EW-9	1,919	20.0	100.0	5.00	20.0	3.1	2.7	5.7
DEQAR-0.4-3-2	3,438	50.0	100.0	1.26	100.0	1.2	1.5	2.7
DEQAR-0.4-2-1	1,248	45.0	100.0	1.26	100.0	1.5	1.0	2.5
DEQAR-0.5-3-2	3,199	40.6	100.0	1.31	100.0	1.2	1.5	2.7
DEQAR-0.5-2-1	534	33.9	100.0	1.27	100.0	1.6	1.0	2.6

Tabla A.3: Resultados para el conjunto de datos *bolts* (mejores 100 reglas).

método	#reglas	sop	conf	lift	cober	#atrAnt	#atrCon	#atrib
APRIORI-EF-2	2,746	37.2	100.0	1.10	99.4	2.4	1.2	3.6
APRIORI-EF-3	1,018	27.0	100.0	1.07	99.4	2.2	1.1	3.3
APRIORI-EF-4	428	23.2	100.0	1.01	99.7	1.9	1.1	3.0
APRIORI-EF-6	299	18.7	100.0	1.01	99.7	1.5	1.1	2.7
APRIORI-EF-8	308	15.7	100.0	1.14	99.4	1.6	1.2	2.8
APRIORI-EF-9	240	14.4	100.0	1.13	99.7	1.5	1.2	2.7
APRIORI-EW-2	4,705	41.3	100.0	1.17	99.7	2.7	1.3	4.0
APRIORI-EW-3	2,007	31.7	100.0	1.01	98.5	2.4	1.1	3.6
APRIORI-EW-4	995	28.6	100.0	1.02	99.7	2.3	1.2	3.5
APRIORI-EW-6	581	24.3	100.0	1.01	99.4	2.1	1.1	3.2
APRIORI-EW-8	358	21.9	100.0	1.02	99.4	1.9	1.2	3.1
APRIORI-EW-9	280	20.3	100.0	1.02	99.7	1.9	1.1	3.1
DEQAR-0.4-3-2	1,689	64.0	100.0	1.05	100.0	1.4	1.6	3.0
DEQAR-0.4-2-1	515	56.3	100.0	1.09	100.0	1.7	1.0	2.7
DEQAR-0.5-3-2	2,276	54.5	100.0	1.05	100.0	1.4	1.6	3.1
DEQAR-0.5-2-1	476	49.9	100.0	1.09	100.0	1.7	1.0	2.7

Tabla A.4: Resultados para el conjunto de datos *ecoli* (mejores 100 reglas).

método	#reglas	sop	conf	lift	cober	#atrAnt	#atrCon	#atrib
APRIORI-EF-2	9,758,153	50.3	100.0	1.60	100.0	2.2	1.2	3.4
APRIORI-EF-3	647,708	38.6	100.0	2.04	41.1	2.0	1.8	3.8
APRIORI-EF-4	103,100	26.2	100.0	2.05	90.8	1.7	1.4	3.0
APRIORI-EF-6	65,144	24.0	100.0	3.07	25.4	2.0	1.8	3.8
APRIORI-EF-8	52,907	15.7	100.0	3.00	73.1	1.7	1.4	3.0
APRIORI-EF-9	38,812	15.2	100.0	3.09	61.0	1.7	1.4	3.1
APRIORI-EW-2	104,661,028	96.8	100.0	1.02	100.0	2.3	1.5	3.8
APRIORI-EW-3	210,758,018	84.9	100.0	1.11	98.2	2.0	1.8	3.8
APRIORI-EW-4	37,834,094	68.9	100.0	1.22	98.6	2.0	1.5	3.4
APRIORI-EW-6	3,367,403	55.1	100.0	1.20	98.6	2.5	1.3	3.7
APRIORI-EW-8	554,736	47.0	100.0	1.47	96.2	1.9	1.4	3.3
APRIORI-EW-9	877,116	49.7	100.0	1.61	70.2	1.9	1.8	3.7
DEQAR-0.4-3-2	4,762	94.5	100.0	1.00	100.0	1.8	1.8	3.5
DEQAR-0.4-2-1	3,193	76.8	100.0	1.00	100.0	1.6	1.0	2.6
DEQAR-0.5-3-2	4,780	94.5	100.0	1.00	100.0	1.8	1.8	3.5
DEQAR-0.5-2-1	3,670	72.0	100.0	1.00	100.0	1.7	1.0	2.7

Tabla A.5: Resultados para el conjunto de datos *elevators* (mejores 100 reglas).

método	#reglas	sop	conf	lift	cober	#atrAnt	#atrCon	#atrib
APRIORI-EF-2	3,337	24.6	100.0	1.14	98.1	2.3	1.0	3.4
APRIORI-EF-3	592	16.5	100.0	1.17	99.2	2.1	1.0	3.2
APRIORI-EF-4	358	14.9	99.9	1.33	98.7	2.0	1.0	3.0
APRIORI-EF-6	115	15.5	98.8	1.21	98.7	1.3	1.0	2.3
APRIORI-EF-8	95	-	-	-	-	-	-	-
APRIORI-EF-9	86	-	-	-	-	-	-	-
APRIORI-EW-2	173,357	70.1	100.0	1.00	99.8	3.0	1.1	4.1
APRIORI-EW-3	33,214	51.6	100.0	1.00	94.4	2.8	1.2	4.0
APRIORI-EW-4	16,353	41.7	100.0	1.11	99.8	2.5	1.1	3.6
APRIORI-EW-6	2,941	25.2	100.0	1.02	100.0	2.2	1.1	3.3
APRIORI-EW-8	1,823	22.4	100.0	1.05	98.8	2.4	1.1	3.5
APRIORI-EW-9	1,603	21.4	100.0	1.06	97.1	2.4	1.1	3.5
DEQAR-0.4-3-2	3,150	63.6	100.0	1.05	100.0	1.2	1.7	2.9
DEQAR-0.4-2-1	1,710	62.7	100.0	1.03	100.0	1.4	1.0	2.4
DEQAR-0.5-3-2	3,339	54.9	100.0	1.08	100.0	1.0	1.7	2.7
DEQAR-0.5-2-1	2,113	54.3	100.0	1.04	100.0	1.5	1.0	2.5

Tabla A.6: Resultados para el conjunto de datos *forest_fires* (mejores 100 reglas).

método	#reglas	sop	conf	lift	cober	#atrAnt	#atrCon	#atrib
APRIORI-EF-2	35,355	11.8	99.4	1.99	70.5	4.8	1.0	5.8
APRIORI-EF-3	802	13.2	99.7	2.59	70.7	2.8	1.0	3.8
APRIORI-EF-4	241	13.3	97.1	2.44	65.1	2.9	1.1	4.0
APRIORI-EF-6	144	13.7	93.0	2.47	66.9	2.5	1.1	3.6
APRIORI-EF-8	80	-	-	-	-	-	-	-
APRIORI-EF-9	53	-	-	-	-	-	-	-
APRIORI-EW-2	126,296,270	98.1	100.0	1.00	99.0	3.7	1.1	4.8
APRIORI-EW-3	78,544,326	94.0	100.0	1.01	96.9	3.8	1.0	4.8
APRIORI-EW-4	35,368,951	86.6	100.0	1.01	99.4	3.5	1.1	4.6
APRIORI-EW-6	6,497,697	76.2	100.0	1.00	88.3	4.0	1.0	5.0
APRIORI-EW-8	2,361,135	66.8	100.0	1.00	81.5	4.5	1.0	5.5
APRIORI-EW-9	670,261	57.8	100.0	1.00	80.1	4.5	1.0	5.5
DEQAR-0.4-3-2	4,031	64.0	100.0	1.04	100.0	1.0	1.9	2.9
DEQAR-0.4-2-1	2,030	64.0	100.0	1.01	100.0	1.0	1.0	2.0
DEQAR-0.5-3-2	4,143	55.1	100.0	1.09	100.0	1.0	1.9	2.9
DEQAR-0.5-2-1	2,133	55.1	100.0	1.01	100.0	1.0	1.0	2.0

Tabla A.7: Resultados para el conjunto de datos *house_16h* (mejores 100 reglas).

método	#reglas	sop	conf	lift	cober	#atrAnt	#atrCon	#atrib
APRIORI-EF-2	846,785	35.7	100.0	1.56	54.9	3.3	1.1	4.4
APRIORI-EF-3	219,687	27.2	100.0	2.79	38.5	2.6	1.8	4.4
APRIORI-EF-4	55,369	26.0	100.0	2.95	37.4	2.5	1.7	4.2
APRIORI-EF-6	6,629	26.1	100.0	3.74	28.7	2.1	1.8	3.9
APRIORI-EF-8	2,911	26.1	100.0	3.81	27.7	2.0	2.0	3.9
APRIORI-EF-9	1,989	26.1	100.0	3.81	27.7	2.0	2.0	3.9
APRIORI-EW-2	1,434,000	66.2	100.0	1.17	78.7	3.1	1.2	4.2
APRIORI-EW-3	582,954	52.2	100.0	1.13	99.6	2.6	1.1	3.7
APRIORI-EW-4	142,641	38.2	100.0	1.07	100.0	2.5	1.1	3.6
APRIORI-EW-6	21,639	28.0	100.0	2.58	100.0	2.1	1.5	3.6
APRIORI-EW-8	13,065	27.5	100.0	3.13	90.9	2.2	1.6	3.8
APRIORI-EW-9	6,535	27.1	100.0	3.46	75.7	2.0	1.8	3.8
DEQAR-0.4-3-2	3,535	67.9	100.0	1.13	100.0	1.3	1.8	3.1
DEQAR-0.4-2-1	1,895	65.2	100.0	1.04	100.0	1.1	1.0	2.1
DEQAR-0.5-3-2	3,948	63.6	100.0	1.16	100.0	1.3	1.8	3.1
DEQAR-0.5-2-1	2,207	57.7	100.0	1.05	100.0	1.1	1.0	2.1

Tabla A.8: Resultados para el conjunto de datos *housing* (mejores 100 reglas).

método	#reglas	sop	conf	lift	cober	#atrAnt	#atrCon	#atrib
APRIORI-EF-2	3,364	50.0	100.0	2.00	50.0	2.0	2.0	4.0
APRIORI-EF-3	4,302	34.4	100.0	2.93	68.8	1.9	1.8	3.7
APRIORI-EF-4	4,301	25.0	100.0	4.00	25.0	2.7	2.4	5.2
APRIORI-EF-6	2,372	18.8	100.0	5.33	18.8	2.7	2.4	5.2
APRIORI-EF-8	874	12.5	100.0	8.00	37.5	1.7	1.7	3.5
APRIORI-EF-9	1,240	12.5	100.0	8.00	37.5	1.7	1.7	3.5
APRIORI-EW-2	3,255	49.1	100.0	1.96	100.0	1.7	1.5	3.2
APRIORI-EW-3	3,567	28.1	100.0	3.43	100.0	2.2	2.0	4.1
APRIORI-EW-4	3,914	25.0	100.0	4.00	25.0	2.7	2.5	5.2
APRIORI-EW-6	1,917	21.9	100.0	4.67	25.0	2.2	2.2	4.3
APRIORI-EW-8	994	15.4	100.0	6.49	81.3	2.2	2.1	4.3
APRIORI-EW-9	658	15.2	100.0	6.60	43.8	2.3	2.1	4.4
DEQAR-0.4-3-2	2,059	62.5	100.0	1.59	100.0	1.0	1.7	2.7
DEQAR-0.4-2-1	524	48.0	100.0	2.17	100.0	1.6	1.0	2.6
DEQAR-0.5-3-2	1,578	50.0	100.0	2.00	100.0	1.0	1.7	2.7
DEQAR-0.5-2-1	508	44.3	100.0	2.24	100.0	1.2	1.0	2.2

Tabla A.9: Resultados para el conjunto de datos *longley* (mejores 100 reglas).

método	#reglas	sop	conf	lift	cober	#atrAnt	#atrCon	#atrib
APRIORI-EF-2	63,215,203	48.1	100.0	2.00	100.0	2.6	1.2	3.9
APRIORI-EF-3	52,450,272	33.3	100.0	3.00	100.0	2.2	1.5	3.6
APRIORI-EF-4	25,818,869	24.9	100.0	4.01	50.1	2.3	1.5	3.8
APRIORI-EF-6	4,174,239	16.6	100.0	6.00	83.3	1.6	1.4	3.0
APRIORI-EF-8	31,397	12.1	100.0	8.02	96.0	2.3	1.2	3.5
APRIORI-EF-9	972	11.1	100.0	8.99	66.6	1.6	1.4	3.0
APRIORI-EW-2	62,352,696	80.9	100.0	1.17	88.4	2.4	1.7	4.1
APRIORI-EW-3	32,198,193	56.3	100.0	1.62	61.6	3.6	1.2	4.8
APRIORI-EW-4	42,545,680	44.8	100.0	1.91	49.5	2.7	1.2	3.9
APRIORI-EW-6	31,888	22.9	100.0	2.81	96.4	2.6	1.1	3.7
APRIORI-EW-8	16,382	20.5	100.0	2.87	54.1	3.3	1.2	4.5
APRIORI-EW-9	10,539	21.1	100.0	2.74	48.7	3.4	1.2	4.5
DEQAR-0.4-3-2	3,664	64.0	100.0	1.56	100.0	1.0	1.9	2.9
DEQAR-0.4-2-1	2,789	64.0	100.0	1.42	100.0	1.0	1.0	2.0
DEQAR-0.5-3-2	3,250	54.9	100.0	1.82	100.0	1.0	1.9	2.9
DEQAR-0.5-2-1	2,779	54.9	100.0	1.67	100.0	1.0	1.0	2.0

Tabla A.10: Resultados para el conjunto de datos *mortgage* (mejores 100 reglas).

método	#reglas	sop	conf	lift	cober	#atrAnt	#atrCon	#atrib
APRIORI-EF-2	231,524	23.6	100.0	2.05	100.0	2.8	1.1	3.9
APRIORI-EF-3	8,028	16.4	100.0	3.36	83.3	2.4	1.3	3.8
APRIORI-EF-4	479	11.1	100.0	5.82	53.3	2.7	1.8	4.5
APRIORI-EF-6	5	-	-	-	-	-	-	-
APRIORI-EF-8	1	-	-	-	-	-	-	-
APRIORI-EF-9	0	-	-	-	-	-	-	-
APRIORI-EW-2	13,331,566	68.4	100.0	1.09	100.0	2.4	1.3	3.8
APRIORI-EW-3	534,295	61.8	100.0	1.03	96.7	2.1	1.2	3.3
APRIORI-EW-4	124,148	50.2	100.0	1.03	96.7	1.9	1.2	3.1
APRIORI-EW-6	13,044	32.1	100.0	1.07	95.0	2.4	1.1	3.5
APRIORI-EW-8	1,702	24.5	100.0	1.09	93.3	2.0	1.2	3.1
APRIORI-EW-9	865	20.4	100.0	1.11	93.3	1.8	1.2	3.0
DEQAR-0.4-3-2	4,115	63.3	100.0	1.32	100.0	1.0	2.0	3.0
DEQAR-0.4-2-1	1,913	63.3	100.0	1.10	100.0	1.0	1.0	2.0
DEQAR-0.5-3-2	4,206	53.3	100.0	1.42	100.0	1.0	1.9	2.9
DEQAR-0.5-2-1	2,523	53.3	100.0	1.12	100.0	1.0	1.0	2.0

Tabla A.11: Resultados para el conjunto de datos *pollution* (mejores 100 reglas).

método	#reglas	sop	conf	lift	cober	#atrAnt	#atrCon	#atrib
APRIORI-EF-2	174,273,425	43.1	100.0	2.00	100.0	2.9	1.2	4.0
APRIORI-EF-3	33,852,825	30.1	100.0	3.11	85.5	2.4	1.8	4.2
APRIORI-EF-4	345,980	19.7	100.0	4.35	91.9	2.5	1.5	4.0
APRIORI-EF-6	131,761	15.1	100.0	6.38	93.6	2.3	2.2	4.5
APRIORI-EF-8	2,425	12.1	100.0	7.51	74.2	2.4	2.1	4.5
APRIORI-EF-9	20	-	-	-	-	-	-	-
APRIORI-EW-2	119,770,292	84.0	100.0	1.14	87.1	2.2	1.7	3.9
APRIORI-EW-3	32,697,500	62.0	100.0	1.35	75.8	2.1	1.8	3.9
APRIORI-EW-4	3,582,154	46.5	100.0	1.64	83.9	2.5	1.4	3.9
APRIORI-EW-6	106,517	32.2	100.0	2.10	67.7	2.6	1.3	3.9
APRIORI-EW-8	55,399	25.7	100.0	2.58	79.0	2.8	1.4	4.1
APRIORI-EW-9	10,353	22.8	100.0	2.58	62.9	2.6	1.4	4.0
DEQAR-0.4-3-2	3,957	62.9	100.0	1.50	100.0	1.0	2.0	3.0
DEQAR-0.4-2-1	2,477	62.9	100.0	1.30	100.0	1.0	1.0	2.0
DEQAR-0.5-3-2	4,037	54.8	100.0	1.75	100.0	1.0	2.0	3.0
DEQAR-0.5-2-1	2,609	54.8	100.0	1.42	100.0	1.0	1.0	2.0

Tabla A.12: Resultados para el conjunto de datos *school* (mejores 100 reglas).

método	#reglas	sop	conf	lift	cober	#atrAnt	#atrCon	#atrib
APRIORI-EF-2	29,618	31.8	100.0	2.00	99.5	3.6	1.1	4.7
APRIORI-EF-3	2,871	15.5	100.0	3.16	97.8	2.8	1.1	3.9
APRIORI-EF-4	140	12.7	92.6	3.77	91.7	2.2	1.1	3.3
APRIORI-EF-6	6	-	-	-	-	-	-	-
APRIORI-EF-8	3	-	-	-	-	-	-	-
APRIORI-EF-9	0	-	-	-	-	-	-	-
APRIORI-EW-2	39,112	40.6	100.0	1.73	57.8	3.2	1.2	4.3
APRIORI-EW-3	2,698	18.1	100.0	2.67	87.9	2.5	1.2	3.7
APRIORI-EW-4	1,473	11.3	98.6	2.66	87.3	3.2	1.0	4.3
APRIORI-EW-6	16	-	-	-	-	-	-	-
APRIORI-EW-8	2	-	-	-	-	-	-	-
APRIORI-EW-9	3	-	-	-	-	-	-	-
DEQAR-0.4-3-2	2,192	63.9	100.0	1.15	100.0	1.0	1.8	2.8
DEQAR-0.4-2-1	1,026	60.6	100.0	1.12	100.0	1.1	1.0	2.1
DEQAR-0.5-3-2	2,358	54.9	100.0	1.27	100.0	1.0	1.8	2.8
DEQAR-0.5-2-1	955	52.3	100.0	1.12	100.0	1.1	1.0	2.1

Tabla A.13: Resultados para el conjunto de datos *stock_prices* (mejores 100 reglas).

método	#reglas	sop	conf	lift	cober	#atrAnt	#atrCon	#atrib
APRIORI-EF-2	499	16.2	99.7	1.09	92.7	3.1	1.0	4.1
APRIORI-EF-3	216	16.8	97.7	1.11	96.4	2.2	1.0	3.2
APRIORI-EF-4	138	17.1	96.1	1.07	96.4	2.0	1.0	3.0
APRIORI-EF-6	95	-	-	-	-	-	-	-
APRIORI-EF-8	85	-	-	-	-	-	-	-
APRIORI-EF-9	75	-	-	-	-	-	-	-
APRIORI-EW-2	1,849	33.3	100.0	1.03	100.0	2.9	1.1	4.0
APRIORI-EW-3	1,308	20.1	100.0	1.05	96.4	2.7	1.1	3.8
APRIORI-EW-4	1,023	16.3	100.0	1.11	92.0	2.9	1.2	4.0
APRIORI-EW-6	665	14.2	99.5	1.14	85.4	2.8	1.1	3.9
APRIORI-EW-8	370	16.5	96.5	1.12	97.1	2.6	1.1	3.7
APRIORI-EW-9	298	16.9	96.3	1.13	97.1	2.5	1.1	3.5
DEQAR-0.4-3-2	1,832	50.9	100.0	1.04	100.0	1.9	1.4	3.3
DEQAR-0.4-2-1	585	43.5	100.0	1.03	100.0	1.8	1.0	2.8
DEQAR-0.5-3-2	2,006	48.4	100.0	1.04	100.0	1.7	1.5	3.2
DEQAR-0.5-2-1	555	41.0	100.0	1.03	100.0	1.7	1.0	2.7

Tabla A.14: Resultados para el conjunto de datos *veteran* (mejores 100 reglas).

método	#reglas	sop	conf	lift	cober	#atrAnt	#atrCon	#atrib
APRIORI-EF-2	219,125	28.0	100.0	2.23	44.9	3.0	1.2	4.2
APRIORI-EF-3	3,742	15.2	100.0	3.17	92.1	3.0	1.1	4.1
APRIORI-EF-4	312	12.7	98.8	3.34	92.7	2.5	1.1	3.5
APRIORI-EF-6	19	-	-	-	-	-	-	-
APRIORI-EF-8	19	-	-	-	-	-	-	-
APRIORI-EF-9	18	-	-	-	-	-	-	-
APRIORI-EW-2	263,764	36.6	100.0	1.40	77.5	3.0	1.1	4.1
APRIORI-EW-3	23,247	21.0	100.0	2.14	66.9	3.0	1.0	4.0
APRIORI-EW-4	1,499	13.7	100.0	2.86	94.4	2.9	1.0	3.9
APRIORI-EW-6	72	-	-	-	-	-	-	-
APRIORI-EW-8	25	-	-	-	-	-	-	-
APRIORI-EW-9	16	-	-	-	-	-	-	-
DEQAR-0.4-3-2	3,667	63.5	100.0	1.12	100.0	1.0	1.9	2.9
DEQAR-0.4-2-1	1,848	63.4	100.0	1.03	100.0	1.0	1.0	2.0
DEQAR-0.5-3-2	3,653	54.5	100.0	1.23	100.0	1.0	1.9	2.9
DEQAR-0.5-2-1	1,943	54.4	100.0	1.06	100.0	1.0	1.0	2.0

Tabla A.15: Resultados para el conjunto de datos *wine* (mejores 100 reglas).

método	#reglas	sop	conf	lift	cober	#atrAnt	#atrCon	#atrib
APRIORI-EF-2	1,356	13.9	100.0	1.01	93.9	3.0	1.0	4.1
APRIORI-EF-3	613	12.1	100.0	1.01	98.3	2.4	1.0	3.5
APRIORI-EF-4	296	15.9	99.6	1.01	99.8	2.1	1.1	3.1
APRIORI-EF-6	232	16.6	99.5	1.01	100.0	1.6	1.0	2.6
APRIORI-EF-8	227	14.6	99.6	1.01	99.9	1.5	1.1	2.6
APRIORI-EF-9	208	13.0	99.6	1.01	99.9	1.5	1.1	2.6
APRIORI-EW-2	13,858	22.3	100.0	1.01	28.9	4.1	1.0	5.1
APRIORI-EW-3	3,542	17.2	100.0	1.01	88.8	3.7	1.0	4.7
APRIORI-EW-4	2,325	16.5	100.0	1.01	67.3	3.2	1.1	4.3
APRIORI-EW-6	1,109	13.7	100.0	1.01	89.5	3.1	1.1	4.2
APRIORI-EW-8	538	15.1	99.9	1.01	93.5	2.5	1.1	3.6
APRIORI-EW-9	431	16.0	99.8	1.01	97.3	2.4	1.1	3.4
DEQAR-0.4-3-2	2,210	62.5	100.0	1.00	100.0	2.0	1.6	3.5
DEQAR-0.4-2-1	705	58.0	100.0	1.00	100.0	1.7	1.0	2.7
DEQAR-0.5-3-2	2,985	54.2	100.0	1.00	98.4	1.8	1.6	3.4
DEQAR-0.5-2-1	889	53.1	100.0	1.00	100.0	1.6	1.0	2.6

Tabla A.16: Resultados para el conjunto de datos *yeast* (mejores 100 reglas).

método	#reglas	sop	conf	lift	cober	#atrAnt	#atrCon	#atrib
APRIORI-EF-2	22,028	33.9	100.0	2.00	43.6	4.8	1.0	5.8
APRIORI-EF-3	15,928	24.7	100.0	3.00	29.7	4.1	1.0	5.1
APRIORI-EF-4	10,571	17.3	100.0	4.00	21.9	4.6	1.0	5.6
APRIORI-EF-6	5,366	12.5	100.0	6.00	15.1	4.2	1.0	5.2
APRIORI-EF-8	740	10.4	99.6	7.90	12.2	3.5	1.0	4.6
APRIORI-EF-9	10	-	-	-	-	-	-	-
APRIORI-EW-2	24,658	32.8	100.0	1.02	68.7	2.4	1.1	3.6
APRIORI-EW-3	12,783	33.2	100.0	1.17	95.0	3.3	1.2	4.5
APRIORI-EW-4	9,899	34.3	100.0	1.00	88.4	1.9	1.0	2.9
APRIORI-EW-6	6,659	25.6	100.0	1.11	38.4	2.5	1.0	3.5
APRIORI-EW-8	1,487	16.2	100.0	2.68	21.9	2.9	1.1	4.0
APRIORI-EW-9	703	13.3	99.8	4.11	35.7	3.0	1.0	4.0
DEQAR-0.4-3-2	2,035	64.0	100.0	1.27	99.9	1.0	1.9	2.9
DEQAR-0.4-2-1	841	64.0	100.0	1.11	100.0	1.0	1.0	2.0
DEQAR-0.5-3-2	2,275	55.0	100.0	1.31	100.0	1.0	1.8	2.8
DEQAR-0.5-2-1	914	54.9	100.0	1.10	100.0	1.0	1.0	2.0

Tabla A.17: Resultados para el conjunto de datos *abalone* (mejores 25 reglas).

método	#reglas	sop	conf	lift	cober	#atrAnt	#atrCon	#atrib
APRIORI-EF-2	81,423,427	46.4	100.0	2.00	100.0	1.7	1.2	2.9
APRIORI-EF-3	886,017	31.7	100.0	3.00	100.0	1.6	1.1	2.7
APRIORI-EF-4	11,808	21.4	100.0	4.04	97.2	1.8	1.2	3.0
APRIORI-EF-6	151	16.4	100.0	6.00	100.0	1.4	1.1	2.4
APRIORI-EF-8	98	12.7	100.0	7.86	51.2	1.2	1.2	2.5
APRIORI-EF-9	100	11.0	100.0	8.94	99.6	1.2	1.1	2.3
APRIORI-EW-2	188,161,019	95.5	100.0	1.01	99.2	2.1	1.0	3.2
APRIORI-EW-3	17,535,317	70.1	100.0	1.03	87.7	2.2	1.0	3.3
APRIORI-EW-4	260,976	33.9	100.0	1.26	92.9	2.4	1.0	3.4
APRIORI-EW-6	5,348	14.6	100.0	2.68	84.1	2.9	1.0	3.9
APRIORI-EW-8	750	11.9	100.0	2.49	62.7	3.4	1.0	4.4
APRIORI-EW-9	177	12.0	94.0	2.95	64.7	2.6	1.0	3.6
DEQAR-0.4-3-2	3,232	63.9	100.0	1.56	100.0	1.0	1.9	2.9
DEQAR-0.4-2-1	2,582	63.9	100.0	1.21	100.0	1.0	1.0	2.0
DEQAR-0.5-3-2	3,135	54.8	100.0	1.81	54.8	1.0	1.9	2.9
DEQAR-0.5-2-1	2,710	54.8	100.0	1.33	100.0	1.0	1.0	2.0

Tabla A.18: Resultados para el conjunto de datos *bodyfat* (mejores 25 reglas).

método	#reglas	sop	conf	lift	cober	#atrAnt	#atrCon	#atrib
APRIORI-EF-2	2,310	32.8	100.0	1.83	100.0	2.3	1.0	3.3
APRIORI-EF-3	2,474	23.0	100.0	4.01	50.0	2.0	1.3	3.3
APRIORI-EF-4	1,637	20.5	100.0	4.70	47.5	1.7	1.7	3.4
APRIORI-EF-6	838	20.0	100.0	5.00	20.0	2.0	1.7	3.7
APRIORI-EF-8	467	20.0	100.0	5.00	20.0	2.0	1.7	3.7
APRIORI-EF-9	216	20.0	100.0	5.00	20.0	2.0	1.7	3.7
APRIORI-EW-2	2,903	62.8	100.0	1.29	80.0	1.6	1.4	3.0
APRIORI-EW-3	4,053	28.0	100.0	2.80	100.0	2.2	1.4	3.6
APRIORI-EW-4	3,904	25.7	100.0	3.71	92.5	2.2	2.0	4.1
APRIORI-EW-6	3,713	20.0	100.0	5.00	20.0	2.6	2.1	4.7
APRIORI-EW-8	1,873	20.0	100.0	5.00	20.0	2.6	2.1	4.7
APRIORI-EW-9	1,919	20.0	100.0	5.00	20.0	2.6	2.1	4.7
DEQAR-0.4-3-2	3,438	61.4	100.0	1.26	100.0	1.0	1.5	2.5
DEQAR-0.4-2-1	1,248	60.4	100.0	1.14	100.0	1.0	1.0	2.0
DEQAR-0.5-3-2	3,199	51.4	100.0	1.24	100.0	1.0	1.5	2.5
DEQAR-0.5-2-1	534	46.4	100.0	1.23	100.0	1.0	1.0	2.0

Tabla A.19: Resultados para el conjunto de datos *bolts* (mejores 25 reglas).

método	#reglas	sop	conf	lift	cober	#atrAnt	#atrCon	#atrib
APRIORI-EF-2	2,746	49.3	100.0	1.01	99.4	1.6	1.1	2.7
APRIORI-EF-3	1,018	38.1	100.0	1.01	99.1	1.4	1.1	2.5
APRIORI-EF-4	428	32.3	100.0	1.01	99.1	1.4	1.1	2.5
APRIORI-EF-6	299	26.4	100.0	1.01	98.8	1.4	1.1	2.5
APRIORI-EF-8	308	24.0	100.0	1.01	98.5	1.4	1.1	2.6
APRIORI-EF-9	240	23.2	100.0	1.01	98.5	1.4	1.1	2.6
APRIORI-EW-2	4,705	51.9	100.0	1.02	99.4	2.0	1.1	3.1
APRIORI-EW-3	2,007	44.7	100.0	1.01	98.2	1.9	1.1	3.0
APRIORI-EW-4	995	43.4	100.0	1.02	99.1	1.9	1.1	3.0
APRIORI-EW-6	581	36.6	100.0	1.01	99.4	1.6	1.1	2.8
APRIORI-EW-8	358	33.4	100.0	1.01	98.2	1.5	1.1	2.6
APRIORI-EW-9	280	32.4	100.0	1.01	99.4	1.7	1.1	2.8
DEQAR-0.4-3-2	1,689	65.3	100.0	1.02	100.0	1.0	1.8	2.8
DEQAR-0.4-2-1	515	65.2	100.0	1.04	100.0	1.1	1.0	2.1
DEQAR-0.5-3-2	2,276	56.4	100.0	1.11	98.2	1.2	1.6	2.9
DEQAR-0.5-2-1	476	56.1	100.0	1.04	100.0	1.4	1.0	2.4

Tabla A.20: Resultados para el conjunto de datos *ecoli* (mejores 25 reglas).

método	#reglas	sop	conf	lift	cober	#atrAnt	#atrCon	#atrib
APRIORI-EF-2	9,758,153	51.6	100.0	1.16	99.5	1.6	1.2	2.8
APRIORI-EF-3	647,708	39.0	100.0	2.13	40.5	1.6	1.6	3.2
APRIORI-EF-4	103,100	28.0	100.0	1.00	74.1	1.6	1.2	2.8
APRIORI-EF-6	65,144	24.2	100.0	3.27	25.2	1.6	1.6	3.2
APRIORI-EF-8	52,907	16.9	100.0	1.00	60.5	1.7	1.3	3.0
APRIORI-EF-9	38,812	16.6	100.0	1.00	37.8	1.8	1.2	3.0
APRIORI-EW-2	104,661,028	97.3	100.0	1.02	100.0	1.7	1.5	3.2
APRIORI-EW-3	210,758,018	85.2	100.0	1.13	98.2	1.6	1.6	3.2
APRIORI-EW-4	37,834,094	72.7	100.0	1.02	98.6	2.1	1.1	3.2
APRIORI-EW-6	3,367,403	69.5	100.0	1.00	96.4	2.3	1.2	3.5
APRIORI-EW-8	554,736	55.7	100.0	1.00	87.2	2.3	1.2	3.5
APRIORI-EW-9	877,116	50.1	100.0	1.72	52.3	1.6	1.6	3.2
DEQAR-0.4-3-2	4,762	94.7	100.0	1.00	100.0	1.0	1.7	2.8
DEQAR-0.4-2-1	3,193	94.4	100.0	1.00	100.0	1.6	1.0	2.6
DEQAR-0.5-3-2	4,780	94.7	100.0	1.00	100.0	1.0	1.7	2.8
DEQAR-0.5-2-1	3,670	94.4	100.0	1.00	100.0	1.6	1.0	2.6

Tabla A.21: Resultados para el conjunto de datos *elevators* (mejores 25 reglas).

método	#reglas	sop	conf	lift	cober	#atrAnt	#atrCon	#atrib
APRIORI-EF-2	3,337	31.6	100.0	1.14	95.0	1.9	1.0	2.9
APRIORI-EF-3	592	23.7	100.0	1.02	96.3	1.7	1.0	2.7
APRIORI-EF-4	358	21.1	100.0	1.02	98.5	1.4	1.0	2.4
APRIORI-EF-6	115	18.1	100.0	1.23	98.1	1.1	1.0	2.1
APRIORI-EF-8	95	14.2	100.0	1.02	96.5	1.1	1.0	2.1
APRIORI-EF-9	86	13.4	100.0	1.23	94.2	1.1	1.0	2.2
APRIORI-EW-2	173,357	79.3	100.0	1.00	93.0	2.3	1.2	3.5
APRIORI-EW-3	33,214	61.0	100.0	1.00	84.3	2.2	1.2	3.4
APRIORI-EW-4	16,353	56.6	100.0	1.01	98.7	2.2	1.1	3.3
APRIORI-EW-6	2,941	34.7	100.0	1.02	82.0	1.9	1.1	3.0
APRIORI-EW-8	1,823	27.6	100.0	1.03	93.8	1.8	1.0	2.8
APRIORI-EW-9	1,603	26.5	100.0	1.07	64.8	2.3	1.1	3.4
DEQAR-0.4-3-2	3,150	63.8	100.0	1.11	99.4	1.0	1.7	2.7
DEQAR-0.4-2-1	1,710	63.7	100.0	1.04	100.0	1.1	1.0	2.1
DEQAR-0.5-3-2	3,339	54.9	100.0	1.16	91.5	1.0	1.8	2.8
DEQAR-0.5-2-1	2,113	54.9	100.0	1.04	98.5	1.0	1.0	2.0

Tabla A.22: Resultados para el conjunto de datos *forest_fires* (mejores 25 reglas).

método	#reglas	sop	conf	lift	cober	#atrAnt	#atrCon	#atrib
APRIORI-EF-2	35,355	11.8	99.7	1.99	34.0	4.6	1.0	5.6
APRIORI-EF-3	802	14.0	100.0	2.48	26.4	3.6	1.0	4.6
APRIORI-EF-4	241	15.4	100.0	2.48	32.5	2.7	1.0	3.7
APRIORI-EF-6	144	14.0	99.5	2.60	44.4	2.7	1.1	3.8
APRIORI-EF-8	80	13.4	98.1	2.40	34.6	2.4	1.1	3.5
APRIORI-EF-9	53	14.4	95.9	2.37	39.6	2.6	1.1	3.6
APRIORI-EW-2	126,296,270	98.8	100.0	1.00	99.0	3.0	1.2	4.2
APRIORI-EW-3	78,544,326	96.0	100.0	1.01	96.9	3.0	1.0	4.0
APRIORI-EW-4	35,368,951	89.1	100.0	1.01	99.3	3.0	1.0	4.0
APRIORI-EW-6	6,497,697	81.9	100.0	1.00	84.7	3.8	1.0	4.8
APRIORI-EW-8	2,361,135	77.3	100.0	1.00	81.5	3.6	1.0	4.6
APRIORI-EW-9	670,261	74.3	100.0	1.00	80.1	3.8	1.0	4.8
DEQAR-0.4-3-2	4,031	64.0	100.0	1.13	99.1	1.0	1.9	2.9
DEQAR-0.4-2-1	2,030	64.0	100.0	1.02	100.0	1.0	1.0	2.0
DEQAR-0.5-3-2	4,143	55.6	100.0	1.27	99.5	1.0	1.8	2.8
DEQAR-0.5-2-1	2,133	55.2	100.0	1.05	100.0	1.0	1.0	2.0

Tabla A.23: Resultados para el conjunto de datos *house_16h* (mejores 25 reglas).

método	#reglas	sop	conf	lift	cober	#atrAnt	#atrCon	#atrib
APRIORI-EF-2	846,785	38.9	100.0	1.56	49.8	2.8	1.1	3.8
APRIORI-EF-3	219,687	29.9	100.0	1.95	38.5	2.3	1.1	3.4
APRIORI-EF-4	55,369	27.2	100.0	3.05	37.4	2.2	1.6	3.9
APRIORI-EF-6	6,629	26.2	100.0	3.64	28.7	2.4	2.0	4.4
APRIORI-EF-8	2,911	26.2	100.0	3.73	27.7	2.6	2.0	4.6
APRIORI-EF-9	1,989	26.2	100.0	3.73	27.7	2.6	2.0	4.6
APRIORI-EW-2	1,434,000	70.4	100.0	1.13	78.7	2.2	1.1	3.3
APRIORI-EW-3	582,954	59.3	100.0	1.04	99.4	2.4	1.0	3.4
APRIORI-EW-4	142,641	47.8	100.0	1.04	82.6	2.4	1.0	3.5
APRIORI-EW-6	21,639	32.4	100.0	1.09	79.3	1.8	1.1	2.9
APRIORI-EW-8	13,065	31.6	100.0	1.85	90.9	2.0	1.2	3.2
APRIORI-EW-9	6,535	30.2	100.0	2.61	75.7	2.1	1.4	3.5
DEQAR-0.4-3-2	3,535	75.6	100.0	1.06	98.6	1.0	1.7	2.7
DEQAR-0.4-2-1	1,895	69.7	100.0	1.09	100.0	1.3	1.0	2.3
DEQAR-0.5-3-2	3,948	75.6	100.0	1.06	98.6	1.0	1.7	2.7
DEQAR-0.5-2-1	2,207	66.1	100.0	1.10	100.0	1.3	1.0	2.3

Tabla A.24: Resultados para el conjunto de datos *housing* (mejores 25 reglas).

método	#reglas	sop	conf	lift	cober	#atrAnt	#atrCon	#atrib
APRIORI-EF-2	3,364	50.0	100.0	2.00	50.0	2.0	1.7	3.7
APRIORI-EF-3	4,302	37.5	100.0	2.67	37.5	1.8	1.7	3.5
APRIORI-EF-4	4,301	25.0	100.0	4.00	25.0	2.0	1.7	3.7
APRIORI-EF-6	2,372	18.8	100.0	5.33	18.8	2.0	1.7	3.7
APRIORI-EF-8	874	12.5	100.0	8.00	25.0	1.2	1.2	2.5
APRIORI-EF-9	1,240	12.5	100.0	8.00	25.0	1.2	1.2	2.5
APRIORI-EW-2	3,255	50.0	100.0	2.00	50.0	1.7	1.7	3.4
APRIORI-EW-3	3,567	32.0	100.0	3.00	81.3	1.6	1.6	3.2
APRIORI-EW-4	3,914	25.0	100.0	4.00	25.0	2.1	1.7	3.8
APRIORI-EW-6	1,917	25.0	100.0	4.00	25.0	1.8	1.7	3.5
APRIORI-EW-8	994	19.3	100.0	5.17	62.5	1.5	1.5	3.0
APRIORI-EW-9	658	19.3	100.0	4.96	25.0	1.7	1.5	3.2
DEQAR-0.4-3-2	2,059	62.5	100.0	1.60	100.0	1.0	1.7	2.7
DEQAR-0.4-2-1	524	62.5	100.0	1.57	100.0	1.0	1.0	2.0
DEQAR-0.5-3-2	1,578	50.0	100.0	2.00	100.0	1.0	1.7	2.7
DEQAR-0.5-2-1	508	50.0	100.0	2.00	100.0	1.0	1.0	2.0

Tabla A.25: Resultados para el conjunto de datos *longley* (mejores 25 reglas).

método	#reglas	sop	conf	lift	cober	#atrAnt	#atrCon	#atrib
APRIORI-EF-2	63,215,203	49.8	100.0	2.00	100.0	2.0	1.2	3.2
APRIORI-EF-3	52,450,272	33.3	100.0	3.00	66.7	1.7	1.6	3.3
APRIORI-EF-4	25,818,869	24.9	100.0	4.02	50.1	2.1	2.0	4.1
APRIORI-EF-6	4,174,239	16.7	100.0	5.99	16.7	1.8	1.7	3.5
APRIORI-EF-8	31,397	12.5	100.0	8.01	49.6	1.6	1.2	2.7
APRIORI-EF-9	972	11.2	100.0	8.97	11.2	1.8	1.7	3.5
APRIORI-EW-2	62,352,696	84.4	100.0	1.14	88.4	1.9	1.4	3.3
APRIORI-EW-3	32,198,193	57.9	100.0	1.62	59.3	2.8	1.2	4.1
APRIORI-EW-4	42,545,680	46.5	100.0	1.84	49.5	2.2	1.1	3.4
APRIORI-EW-6	31,888	25.4	100.0	2.96	62.9	1.9	1.1	3.0
APRIORI-EW-8	16,382	23.0	100.0	2.67	33.4	3.0	1.0	4.0
APRIORI-EW-9	10,539	24.2	100.0	2.72	28.9	2.7	1.2	3.9
DEQAR-0.4-3-2	3,664	64.0	100.0	1.56	64.0	1.0	1.9	2.9
DEQAR-0.4-2-1	2,789	64.0	100.0	1.55	100.0	1.0	1.0	2.0
DEQAR-0.5-3-2	3,250	54.9	100.0	1.82	54.9	1.0	1.9	2.9
DEQAR-0.5-2-1	2,779	54.9	100.0	1.79	100.0	1.0	1.0	2.0

Tabla A.26: Resultados para el conjunto de datos *mortgage* (mejores 25 reglas).

método	#reglas	sop	conf	lift	cober	#atrAnt	#atrCon	#atrib
APRIORI-EF-2	231,524	27.6	100.0	1.97	100.0	2.5	1.0	3.5
APRIORI-EF-3	8,028	17.7	100.0	4.02	71.7	2.2	1.9	4.1
APRIORI-EF-4	479	13.1	100.0	4.46	45.0	2.4	1.2	3.6
APRIORI-EF-6	5	-	-	-	-	-	-	-
APRIORI-EF-8	1	-	-	-	-	-	-	-
APRIORI-EF-9	0	-	-	-	-	-	-	-
APRIORI-EW-2	13,331,566	76.3	100.0	1.09	96.7	1.9	1.3	3.2
APRIORI-EW-3	534,295	74.8	100.0	1.03	96.7	1.8	1.1	2.9
APRIORI-EW-4	124,148	64.9	100.0	1.03	96.7	1.5	1.2	2.6
APRIORI-EW-6	13,044	43.4	100.0	1.06	95.0	1.8	1.0	2.8
APRIORI-EW-8	1,702	31.1	100.0	1.09	91.7	1.7	1.2	2.9
APRIORI-EW-9	865	27.7	100.0	1.11	93.3	1.6	1.0	2.7
DEQAR-0.4-3-2	4,115	63.3	100.0	1.42	90.0	1.0	2.0	3.0
DEQAR-0.4-2-1	1,913	63.3	100.0	1.19	100.0	1.0	1.0	2.0
DEQAR-0.5-3-2	4,206	53.3	100.0	1.52	100.0	1.0	2.0	3.0
DEQAR-0.5-2-1	2,523	53.3	100.0	1.36	100.0	1.0	1.0	2.0

Tabla A.27: Resultados para el conjunto de datos *pollution* (mejores 25 reglas).

método	#reglas	sop	conf	lift	cober	#atrAnt	#atrCon	#atrib
APRIORI-EF-2	174,273,425	45.3	100.0	2.01	100.0	2.2	1.2	3.4
APRIORI-EF-3	33,852,825	31.5	100.0	2.91	85.5	1.8	1.3	3.0
APRIORI-EF-4	345,980	20.7	100.0	4.32	91.9	2.1	1.6	3.7
APRIORI-EF-6	131,761	16.3	100.0	5.91	45.2	1.3	1.2	2.5
APRIORI-EF-8	2,425	13.1	100.0	7.56	30.7	1.8	1.7	3.5
APRIORI-EF-9	20	-	-	-	-	-	-	-
APRIORI-EW-2	119,770,292	85.7	100.0	1.12	87.1	1.8	1.4	3.2
APRIORI-EW-3	32,697,500	68.2	100.0	1.34	75.8	2.1	1.8	3.9
APRIORI-EW-4	3,582,154	51.2	100.0	1.57	83.9	2.3	1.3	3.6
APRIORI-EW-6	106,517	35.8	100.0	2.04	67.7	2.2	1.2	3.4
APRIORI-EW-8	55,399	29.5	100.0	2.04	78.8	2.3	1.2	3.5
APRIORI-EW-9	10,353	27.1	100.0	1.88	62.9	2.0	1.1	3.1
DEQAR-0.4-3-2	3,957	62.9	100.0	1.56	100.0	1.0	1.9	2.9
DEQAR-0.4-2-1	2,477	62.9	100.0	1.42	100.0	1.0	1.0	2.0
DEQAR-0.5-3-2	4,037	54.8	100.0	1.79	56.5	1.0	2.0	3.0
DEQAR-0.5-2-1	2,609	54.8	100.0	1.64	100.0	1.0	1.0	2.0

Tabla A.28: Resultados para el conjunto de datos *school* (mejores 25 reglas).

método	#reglas	sop	conf	lift	cober	#atrAnt	#atrCon	#atrib
APRIORI-EF-2	29,618	37.0	100.0	2.00	87.0	2.9	1.1	4.0
APRIORI-EF-3	2,871	18.3	100.0	3.07	74.0	2.8	1.2	4.0
APRIORI-EF-4	140	11.4	98.6	3.95	79.8	2.5	1.0	3.5
APRIORI-EF-6	6	-	-	-	-	-	-	-
APRIORI-EF-8	3	-	-	-	-	-	-	-
APRIORI-EF-9	0	-	-	-	-	-	-	-
APRIORI-EW-2	39,112	43.8	100.0	1.68	57.8	2.3	1.1	3.4
APRIORI-EW-3	2,698	22.8	100.0	2.53	71.5	2.1	1.1	3.2
APRIORI-EW-4	1,473	11.3	99.7	3.08	78.7	2.3	1.0	3.3
APRIORI-EW-6	16	-	-	-	-	-	-	-
APRIORI-EW-8	2	-	-	-	-	-	-	-
APRIORI-EW-9	3	-	-	-	-	-	-	-
DEQAR-0.4-3-2	2,192	63.9	100.0	1.30	100.0	1.0	1.9	2.9
DEQAR-0.4-2-1	1,026	63.9	100.0	1.10	100.0	1.0	1.0	2.0
DEQAR-0.5-3-2	2,358	54.9	100.0	1.41	55.0	1.0	1.9	2.9
DEQAR-0.5-2-1	955	54.9	100.0	1.15	100.0	1.0	1.0	2.0

Tabla A.29: Resultados para el conjunto de datos *stock_prices* (mejores 100 reglas).

método	#reglas	sop	conf	lift	cober	#atrAnt	#atrCon	#atrib
APRIORI-EF-2	499	22.7	100.0	1.07	61.3	2.8	1.0	3.8
APRIORI-EF-3	216	15.0	100.0	1.14	84.7	2.4	1.0	3.4
APRIORI-EF-4	138	13.5	100.0	1.14	80.3	2.3	1.0	3.3
APRIORI-EF-6	95	14.4	99.1	1.10	93.4	1.6	1.0	2.6
APRIORI-EF-8	85	14.4	98.7	1.11	93.4	1.4	1.0	2.4
APRIORI-EF-9	75	14.3	98.5	1.12	91.2	1.5	1.0	2.5
APRIORI-EW-2	1,849	47.9	100.0	1.02	100.0	2.4	1.0	3.4
APRIORI-EW-3	1,308	29.0	100.0	1.03	84.7	2.6	1.0	3.6
APRIORI-EW-4	1,023	22.2	100.0	1.09	70.8	2.5	1.1	3.6
APRIORI-EW-6	665	16.6	100.0	1.11	65.7	2.8	1.0	3.9
APRIORI-EW-8	370	12.6	100.0	1.25	48.9	3.1	1.1	4.2
APRIORI-EW-9	298	12.6	100.0	1.27	67.2	2.9	1.0	3.9
DEQAR-0.4-3-2	1,832	64.3	100.0	1.03	100.0	1.4	1.4	2.8
DEQAR-0.4-2-1	585	61.6	100.0	1.02	100.0	1.5	1.0	2.5
DEQAR-0.5-3-2	2,006	60.5	100.0	1.03	100.0	1.3	1.4	2.7
DEQAR-0.5-2-1	555	57.1	100.0	1.02	100.0	1.4	1.0	2.4

Tabla A.30: Resultados para el conjunto de datos *veteran* (mejores 25 reglas).

método	#reglas	sop	conf	lift	cober	#atrAnt	#atrCon	#atrib
APRIORI-EF-2	219,125	31.8	100.0	2.19	42.7	2.5	1.2	3.6
APRIORI-EF-3	3,742	18.3	100.0	3.06	87.1	2.6	1.0	3.6
APRIORI-EF-4	312	15.6	100.0	3.27	79.2	2.2	1.0	3.2
APRIORI-EF-6	19	-	-	-	-	-	-	-
APRIORI-EF-8	19	-	-	-	-	-	-	-
APRIORI-EF-9	18	-	-	-	-	-	-	-
APRIORI-EW-2	263,764	42.6	100.0	1.40	70.2	2.8	1.0	3.8
APRIORI-EW-3	23,247	24.8	100.0	2.05	64.0	2.5	1.0	3.6
APRIORI-EW-4	1,499	16.7	100.0	2.87	88.2	2.6	1.0	3.6
APRIORI-EW-6	72	12.4	99.4	3.01	80.9	1.9	1.0	2.9
APRIORI-EW-8	25	12.6	91.1	2.85	91.0	1.6	1.0	2.6
APRIORI-EW-9	16	-	-	-	-	-	-	-
DEQAR-0.4-3-2	3,667	63.5	100.0	1.25	96.1	1.0	1.9	2.9
DEQAR-0.4-2-1	1,848	63.5	100.0	1.07	100.0	1.0	1.0	2.0
DEQAR-0.5-3-2	3,653	54.5	100.0	1.43	61.2	1.0	1.9	2.9
DEQAR-0.5-2-1	1,943	54.5	100.0	1.13	100.0	1.0	1.0	2.0

Tabla A.31: Resultados para el conjunto de datos *wine* (mejores 25 reglas).

método	#reglas	sop	conf	lift	cober	#atrAnt	#atrCon	#atrib
APRIORI-EF-2	1,356	18.3	100.0	1.01	77.8	2.5	1.0	3.5
APRIORI-EF-3	613	15.0	100.0	1.01	83.6	2.3	1.0	3.3
APRIORI-EF-4	296	13.9	100.0	1.01	80.3	2.2	1.0	3.2
APRIORI-EF-6	232	16.4	100.0	1.01	93.5	1.8	1.0	2.8
APRIORI-EF-8	227	14.1	100.0	1.01	86.9	1.5	1.0	2.5
APRIORI-EF-9	208	13.1	100.0	1.01	87.9	1.6	1.0	2.6
APRIORI-EW-2	13,858	26.0	100.0	1.01	28.9	3.0	1.0	4.0
APRIORI-EW-3	3,542	21.1	100.0	1.01	63.4	3.1	1.0	4.1
APRIORI-EW-4	2,325	21.1	100.0	1.01	28.9	2.6	1.1	3.8
APRIORI-EW-6	1,109	18.2	100.0	1.01	76.6	2.6	1.0	3.6
APRIORI-EW-8	538	16.4	100.0	1.01	74.9	2.4	1.0	3.4
APRIORI-EW-9	431	15.2	100.0	1.01	78.9	2.4	1.0	3.4
DEQAR-0.4-3-2	2,210	63.3	100.0	1.00	87.1	1.6	1.6	3.2
DEQAR-0.4-2-1	705	62.9	100.0	1.00	99.4	1.7	1.0	2.7
DEQAR-0.5-3-2	2,985	55.0	100.0	1.00	82.6	1.4	1.6	3.0
DEQAR-0.5-2-1	889	54.6	100.0	1.00	90.8	1.6	1.0	2.6

Tabla A.32: Resultados para el conjunto de datos *yeast* (mejores 25 reglas).

i	comparación	pHolm
1	APRIORI-EF-9 vs .DEQAR-0.4-3-2	0.000
2	APRIORI-EF-8 vs .DEQAR-0.4-3-2	0.000
3	APRIORI-EF-6 vs .DEQAR-0.4-3-2	0.000
4	APRIORI-EF-9 vs .APRIORI-EW-2	0.000
5	APRIORI-EF-8 vs .APRIORI-EW-2	0.000
6	APRIORI-EF-9 vs .DEQAR-0.4-2-1	0.000
7	APRIORI-EF-8 vs .DEQAR-0.4-2-1	0.000
8	APRIORI-EF-6 vs .APRIORI-EW-2	0.000
9	APRIORI-EF-9 vs .DEQAR-0.5-3-2	0.000
10	APRIORI-EF-8 vs .DEQAR-0.5-3-2	0.000
11	APRIORI-EF-6 vs .DEQAR-0.4-2-1	0.000
12	APRIORI-EW-9 vs .DEQAR-0.4-3-2	0.000
13	APRIORI-EF-4 vs .DEQAR-0.4-3-2	0.000
14	APRIORI-EF-6 vs .DEQAR-0.5-3-2	0.000
15	APRIORI-EF-9 vs .DEQAR-0.5-2-1	0.000
16	APRIORI-EW-8 vs .DEQAR-0.4-3-2	0.000
17	APRIORI-EF-9 vs .APRIORI-EW-3	0.000
18	APRIORI-EW-2 vs .APRIORI-EW-9	0.000
19	APRIORI-EF-8 vs .DEQAR-0.5-2-1	0.000
20	APRIORI-EF-8 vs .APRIORI-EW-3	0.000
21	APRIORI-EW-9 vs .DEQAR-0.4-2-1	0.000
22	APRIORI-EF-4 vs .APRIORI-EW-2	0.000
23	APRIORI-EF-6 vs .DEQAR-0.5-2-1	0.000
24	APRIORI-EF-4 vs .DEQAR-0.4-2-1	0.000
25	APRIORI-EW-2 vs .APRIORI-EW-8	0.000
26	APRIORI-EF-6 vs .APRIORI-EW-3	0.000
27	APRIORI-EW-9 vs .DEQAR-0.5-3-2	0.000
28	APRIORI-EW-6 vs .DEQAR-0.4-3-2	0.000
29	APRIORI-EW-8 vs .DEQAR-0.4-2-1	0.000
30	APRIORI-EF-3 vs .DEQAR-0.4-3-2	0.000
31	APRIORI-EF-4 vs .DEQAR-0.5-3-2	0.000
32	APRIORI-EF-9 vs .APRIORI-EW-4	0.001
33	APRIORI-EW-8 vs .DEQAR-0.5-3-2	0.001
34	APRIORI-EF-2 vs .APRIORI-EF-9	0.001
35	APRIORI-EF-8 vs .APRIORI-EW-4	0.001
36	APRIORI-EF-2 vs .APRIORI-EF-8	0.002
37	APRIORI-EW-2 vs .APRIORI-EW-6	0.002
38	APRIORI-EW-9 vs .DEQAR-0.5-2-1	0.003
39	APRIORI-EF-3 vs .APRIORI-EW-2	0.003
40	APRIORI-EW-3 vs .APRIORI-EW-9	0.005
41	APRIORI-EW-6 vs .DEQAR-0.4-2-1	0.007
42	APRIORI-EF-3 vs .DEQAR-0.4-2-1	0.009
43	APRIORI-EF-6 vs .APRIORI-EW-4	0.009
44	APRIORI-EF-4 vs .DEQAR-0.5-2-1	0.009
45	APRIORI-EF-2 vs .APRIORI-EF-6	0.012
46	APRIORI-EF-4 vs .APRIORI-EW-3	0.014
47	APRIORI-EW-8 vs .DEQAR-0.5-2-1	0.025
48	APRIORI-EW-6 vs .DEQAR-0.5-3-2	0.029
49	APRIORI-EF-3 vs .DEQAR-0.5-3-2	0.035
50	APRIORI-EW-3 vs .APRIORI-EW-8	0.037
51	APRIORI-EF-2 vs .DEQAR-0.4-3-2	0.039
52	APRIORI-EW-4 vs .DEQAR-0.4-3-2	0.050

Tabla A.33: p -valores ajustados del test de Holm (conf-sop-lift, mejores 100 reglas).

i	comparación	pHolm
1	APRIORI-EF-9 vs .DEQAR-0.5-3-2	0
2	APRIORI-EF-8 vs .DEQAR-0.5-3-2	0
3	APRIORI-EF-6 vs .DEQAR-0.5-3-2	0
4	APRIORI-EF-9 vs .DEQAR-0.4-3-2	0
5	APRIORI-EF-8 vs .DEQAR-0.4-3-2	0
6	APRIORI-EF-9 vs .DEQAR-0.5-2-1	0
7	APRIORI-EF-8 vs .DEQAR-0.5-2-1	0
8	APRIORI-EF-6 vs .DEQAR-0.4-3-2	0
9	APRIORI-EF-6 vs .DEQAR-0.5-2-1	0.000001
10	APRIORI-EW-9 vs .DEQAR-0.5-3-2	0.000001
11	APRIORI-EF-9 vs .DEQAR-0.4-2-1	0.000001
12	APRIORI-EF-8 vs .DEQAR-0.4-2-1	0.000002
13	APRIORI-EW-9 vs .DEQAR-0.4-3-2	0.000006
14	APRIORI-EF-6 vs .DEQAR-0.4-2-1	0.000008
15	APRIORI-EF-4 vs .DEQAR-0.5-3-2	0.000009
16	APRIORI-EW-9 vs .DEQAR-0.5-2-1	0.000019
17	APRIORI-EW-8 vs .DEQAR-0.5-3-2	0.000026
18	APRIORI-EF-4 vs .DEQAR-0.4-3-2	0.000055
19	APRIORI-EW-9 vs .DEQAR-0.4-2-1	0.000141
20	APRIORI-EF-4 vs .DEQAR-0.5-2-1	0.000154
21	APRIORI-EW-8 vs .DEQAR-0.4-3-2	0.000154
22	APRIORI-EW-8 vs .DEQAR-0.5-2-1	0.000410
23	APRIORI-EF-2 vs .APRIORI-EF-9	0.000974
24	APRIORI-EF-4 vs .DEQAR-0.4-2-1	0.000974
25	APRIORI-EF-2 vs .APRIORI-EF-8	0.001341
26	APRIORI-EW-6 vs .DEQAR-0.5-3-2	0.001341
27	APRIORI-EW-8 vs .DEQAR-0.4-2-1	0.002355
28	APRIORI-EF-2 vs .APRIORI-EF-6	0.003501
29	APRIORI-EF-9 vs .APRIORI-EW-2	0.003501
30	APRIORI-EF-8 vs .APRIORI-EW-2	0.004716
31	APRIORI-EW-6 vs .DEQAR-0.4-3-2	0.005908
32	APRIORI-EF-6 vs .APRIORI-EW-2	0.011665
33	APRIORI-EW-6 vs .DEQAR-0.5-2-1	0.013401
34	APRIORI-EF-2 vs .APRIORI-EW-9	0.031712
35	APRIORI-EF-3 vs .DEQAR-0.5-3-2	0.044476
36	APRIORI-EF-9 vs .APRIORI-EW-4	0.057816

Tabla A.34: p -valores ajustados del test de Holm (conf-cob-lift, mejores 100 reglas).

i	comparación	pHolm
1	APRIORI-EF-8 vs .DEQAR-0.4-3-2	0
2	APRIORI-EF-8 vs .DEQAR-0.4-2-1	0
3	APRIORI-EF-8 vs .DEQAR-0.5-2-1	0
4	APRIORI-EF-8 vs .DEQAR-0.5-3-2	0.000001
5	APRIORI-EF-9 vs .DEQAR-0.4-3-2	0.000001
6	APRIORI-EF-9 vs .DEQAR-0.4-2-1	0.000001
7	APRIORI-EF-9 vs .DEQAR-0.5-2-1	0.000001
8	APRIORI-EF-9 vs .DEQAR-0.5-3-2	0.000004
9	APRIORI-EF-6 vs .DEQAR-0.4-3-2	0.000060
10	APRIORI-EF-6 vs .DEQAR-0.4-2-1	0.000060
11	APRIORI-EF-6 vs .DEQAR-0.5-2-1	0.000060
12	APRIORI-EW-9 vs .DEQAR-0.4-3-2	0.000078
13	APRIORI-EW-9 vs .DEQAR-0.4-2-1	0.000078
14	APRIORI-EW-9 vs .DEQAR-0.5-2-1	0.000078
15	APRIORI-EF-6 vs .DEQAR-0.5-3-2	0.000161
16	APRIORI-EW-8 vs .DEQAR-0.4-3-2	0.000211
17	APRIORI-EW-8 vs .DEQAR-0.4-2-1	0.000211
18	APRIORI-EW-8 vs .DEQAR-0.5-2-1	0.000211
19	APRIORI-EW-9 vs .DEQAR-0.5-3-2	0.000211
20	APRIORI-EW-8 vs .DEQAR-0.5-3-2	0.000546
21	APRIORI-EW-6 vs .DEQAR-0.4-3-2	0.000703
22	APRIORI-EW-6 vs .DEQAR-0.4-2-1	0.000703
23	APRIORI-EW-6 vs .DEQAR-0.5-2-1	0.000703
24	APRIORI-EF-4 vs .DEQAR-0.4-3-2	0.001744
25	APRIORI-EF-4 vs .DEQAR-0.4-2-1	0.001744
26	APRIORI-EF-4 vs .DEQAR-0.5-2-1	0.001744
27	APRIORI-EW-6 vs .DEQAR-0.5-3-2	0.001744
28	APRIORI-EF-4 vs .DEQAR-0.5-3-2	0.004110
29	APRIORI-EW-3 vs .DEQAR-0.4-3-2	0.018839
30	APRIORI-EW-3 vs .DEQAR-0.4-2-1	0.018839
31	APRIORI-EW-3 vs .DEQAR-0.5-2-1	0.018839
32	APRIORI-EF-3 vs .DEQAR-0.4-3-2	0.034815
33	APRIORI-EF-3 vs .DEQAR-0.4-2-1	0.034815
34	APRIORI-EF-3 vs .DEQAR-0.5-2-1	0.034815
35	APRIORI-EW-3 vs .DEQAR-0.5-3-2	0.038706
36	APRIORI-EF-8 vs .APRIORI-EW-2	0.043959
37	APRIORI-EW-4 vs .DEQAR-0.4-3-2	0.061137

Tabla A.35: p -valores ajustados del test de Holm (cobertura, mejores 100 reglas).

i	comparación	pHolm
1	APRIORI-EF-2 vs .DEQAR-0.5-2-1	0.000001
2	APRIORI-EF-2 vs .DEQAR-0.4-2-1	0.000002
3	APRIORI-EW-8 vs .DEQAR-0.5-2-1	0.000005
4	APRIORI-EW-2 vs .DEQAR-0.5-2-1	0.000009
5	APRIORI-EW-8 vs .DEQAR-0.4-2-1	0.000009
6	APRIORI-EF-8 vs .DEQAR-0.5-2-1	0.000014
7	APRIORI-EW-2 vs .DEQAR-0.4-2-1	0.000014
8	APRIORI-EW-3 vs .DEQAR-0.5-2-1	0.000014
9	APRIORI-EW-6 vs .DEQAR-0.5-2-1	0.000017
10	APRIORI-EF-8 vs .DEQAR-0.4-2-1	0.000022
11	APRIORI-EW-3 vs .DEQAR-0.4-2-1	0.000022
12	APRIORI-EW-6 vs .DEQAR-0.4-2-1	0.000027
13	APRIORI-EF-9 vs .DEQAR-0.5-2-1	0.000043
14	APRIORI-EW-9 vs .DEQAR-0.5-2-1	0.000047
15	APRIORI-EF-9 vs .DEQAR-0.4-2-1	0.000069
16	APRIORI-EW-9 vs .DEQAR-0.4-2-1	0.000075
17	APRIORI-EW-4 vs .DEQAR-0.5-2-1	0.000090
18	APRIORI-EF-3 vs .DEQAR-0.5-2-1	0.000143
19	APRIORI-EW-4 vs .DEQAR-0.4-2-1	0.000143
20	APRIORI-EF-3 vs .DEQAR-0.4-2-1	0.000222
21	APRIORI-EF-6 vs .DEQAR-0.5-2-1	0.000316
22	APRIORI-EF-6 vs .DEQAR-0.4-2-1	0.000490
23	APRIORI-EF-4 vs .DEQAR-0.5-2-1	0.005079
24	APRIORI-EF-2 vs .DEQAR-0.5-3-2	0.007442
25	APRIORI-EF-4 vs .DEQAR-0.4-2-1	0.007442
26	APRIORI-EF-2 vs .DEQAR-0.4-3-2	0.008508
27	APRIORI-EW-8 vs .DEQAR-0.5-3-2	0.022277
28	APRIORI-EW-8 vs .DEQAR-0.4-3-2	0.025475
29	APRIORI-EW-2 vs .DEQAR-0.5-3-2	0.031238
30	APRIORI-EW-2 vs .DEQAR-0.4-3-2	0.035597
31	APRIORI-EF-8 vs .DEQAR-0.5-3-2	0.043428
32	APRIORI-EW-3 vs .DEQAR-0.5-3-2	0.043428
33	APRIORI-EF-8 vs .DEQAR-0.4-3-2	0.048761
34	APRIORI-EW-3 vs .DEQAR-0.4-3-2	0.048761
35	APRIORI-EW-6 vs .DEQAR-0.5-3-2	0.048761
36	APRIORI-EW-6 vs .DEQAR-0.4-3-2	0.054014

Tabla A.36: p -valores ajustados del test de Holm (atributos, mejores 100 reglas).

i	comparación	pHolm
1	APRIORI-EF-9 vs .DEQAR-0.4-3-2	0
2	APRIORI-EF-8 vs .DEQAR-0.4-3-2	0
3	APRIORI-EF-9 vs .APRIORI-EW-2	0
4	APRIORI-EF-9 vs .DEQAR-0.4-2-1	0
5	APRIORI-EF-6 vs .DEQAR-0.4-3-2	0
6	APRIORI-EF-8 vs .APRIORI-EW-2	0
7	APRIORI-EF-8 vs .DEQAR-0.4-2-1	0
8	APRIORI-EF-9 vs .DEQAR-0.5-3-2	0
9	APRIORI-EF-6 vs .APRIORI-EW-2	0
10	APRIORI-EF-8 vs .DEQAR-0.5-3-2	0
11	APRIORI-EF-6 vs .DEQAR-0.4-2-1	0
12	APRIORI-EF-9 vs .APRIORI-EW-3	0
13	APRIORI-EF-9 vs .DEQAR-0.5-2-1	0
14	APRIORI-EW-9 vs .DEQAR-0.4-3-2	0
15	APRIORI-EF-4 vs .DEQAR-0.4-3-2	0.000001
16	APRIORI-EF-6 vs .DEQAR-0.5-3-2	0.000002
17	APRIORI-EF-8 vs .APRIORI-EW-3	0.000003
18	APRIORI-EF-8 vs .DEQAR-0.5-2-1	0.000004
19	APRIORI-EW-2 vs .APRIORI-EW-9	0.000007
20	APRIORI-EW-8 vs .DEQAR-0.4-3-2	0.000008
21	APRIORI-EW-9 vs .DEQAR-0.4-2-1	0.000016
22	APRIORI-EF-4 vs .APRIORI-EW-2	0.000027
23	APRIORI-EF-6 vs .APRIORI-EW-3	0.000048
24	APRIORI-EF-4 vs .DEQAR-0.4-2-1	0.000057
25	APRIORI-EF-6 vs .DEQAR-0.5-2-1	0.000062
26	APRIORI-EF-9 vs .APRIORI-EW-4	0.000062
27	APRIORI-EW-2 vs .APRIORI-EW-8	0.000130
28	APRIORI-EF-3 vs .DEQAR-0.4-3-2	0.000187
29	APRIORI-EW-9 vs .DEQAR-0.5-3-2	0.000222
30	APRIORI-EW-8 vs .DEQAR-0.4-2-1	0.000263
31	APRIORI-EF-2 vs .APRIORI-EF-9	0.000445
32	APRIORI-EF-8 vs .APRIORI-EW-4	0.000525
33	APRIORI-EF-4 vs .DEQAR-0.5-3-2	0.000675
34	APRIORI-EW-6 vs .DEQAR-0.4-3-2	0.000865
35	APRIORI-EF-3 vs .APRIORI-EW-2	0.002155
36	APRIORI-EW-8 vs .DEQAR-0.5-3-2	0.002721
37	APRIORI-EF-2 vs .APRIORI-EF-8	0.003162
38	APRIORI-EW-3 vs .APRIORI-EW-9	0.003668
39	APRIORI-EF-3 vs .DEQAR-0.4-2-1	0.003925
40	APRIORI-EW-9 vs .DEQAR-0.5-2-1	0.004544
41	APRIORI-EF-6 vs .APRIORI-EW-4	0.005252
42	APRIORI-EW-2 vs .APRIORI-EW-6	0.008247
43	APRIORI-EF-4 vs .APRIORI-EW-3	0.009480
44	APRIORI-EF-4 vs .DEQAR-0.5-2-1	0.011726
45	APRIORI-EW-6 vs .DEQAR-0.4-2-1	0.014459
46	APRIORI-EF-2 vs .APRIORI-EF-6	0.025466
47	APRIORI-EF-3 vs .DEQAR-0.5-3-2	0.028947
48	APRIORI-EW-3 vs .APRIORI-EW-8	0.030635
49	APRIORI-EW-8 vs .DEQAR-0.5-2-1	0.037236
50	APRIORI-EF-9 vs .APRIORI-EW-6	0.042137
51	APRIORI-EF-2 vs .DEQAR-0.4-3-2	0.071122

Tabla A.37: p -valores ajustados del test de Holm (conf-sop-lift, mejores 25 reglas).

i	comparación	pHolm
1	APRIORI-EF-9 vs .DEQAR-0.5-2-1	0
2	APRIORI-EF-8 vs .DEQAR-0.5-2-1	0
3	APRIORI-EF-9 vs .DEQAR-0.4-2-1	0
4	APRIORI-EF-8 vs .DEQAR-0.4-2-1	0.000001
5	APRIORI-EW-9 vs .DEQAR-0.5-2-1	0.000002
6	APRIORI-EF-6 vs .DEQAR-0.5-2-1	0.000003
7	APRIORI-EF-9 vs .DEQAR-0.4-3-2	0.000004
8	APRIORI-EW-9 vs .DEQAR-0.4-2-1	0.000009
9	APRIORI-EF-6 vs .DEQAR-0.4-2-1	0.000011
10	APRIORI-EF-8 vs .DEQAR-0.4-3-2	0.000011
11	APRIORI-EW-8 vs .DEQAR-0.5-2-1	0.000040
12	APRIORI-EW-9 vs .DEQAR-0.4-3-2	0.000125
13	APRIORI-EW-8 vs .DEQAR-0.4-2-1	0.000136
14	APRIORI-EF-6 vs .DEQAR-0.4-3-2	0.000148
15	APRIORI-EW-6 vs .DEQAR-0.5-2-1	0.000335
16	APRIORI-EW-6 vs .DEQAR-0.4-2-1	0.001043
17	APRIORI-EW-8 vs .DEQAR-0.4-3-2	0.001453
18	APRIORI-EF-4 vs .DEQAR-0.5-2-1	0.003298
19	APRIORI-EF-2 vs .APRIORI-EF-9	0.009135
20	APRIORI-EF-4 vs .DEQAR-0.4-2-1	0.009135
21	APRIORI-EW-6 vs .DEQAR-0.4-3-2	0.009135
22	APRIORI-EF-9 vs .DEQAR-0.5-3-2	0.015076
23	APRIORI-EF-2 vs .APRIORI-EF-8	0.018645
24	APRIORI-EF-9 vs .APRIORI-EW-2	0.019863
25	APRIORI-EF-8 vs .DEQAR-0.5-3-2	0.030354
26	APRIORI-EF-8 vs .APRIORI-EW-2	0.039868
27	APRIORI-EF-4 vs .DEQAR-0.4-3-2	0.059733

Tabla A.38: p -valores ajustados del test de Holm (conf-cob-lift, mejores 25 reglas).

i	comparación	pHolm
1	APRIORI-EF-9 vs .DEQAR-0.4-2-1	0
2	APRIORI-EF-8 vs .DEQAR-0.4-2-1	0
3	APRIORI-EF-9 vs .DEQAR-0.5-2-1	0
4	APRIORI-EF-8 vs .DEQAR-0.5-2-1	0.000001
5	APRIORI-EW-9 vs .DEQAR-0.4-2-1	0.000001
6	APRIORI-EF-6 vs .DEQAR-0.4-2-1	0.000003
7	APRIORI-EW-9 vs .DEQAR-0.5-2-1	0.000004
8	APRIORI-EF-6 vs .DEQAR-0.5-2-1	0.000008
9	APRIORI-EW-8 vs .DEQAR-0.4-2-1	0.000020
10	APRIORI-EF-9 vs .DEQAR-0.4-3-2	0.000030
11	APRIORI-EF-8 vs .DEQAR-0.4-3-2	0.000049
12	APRIORI-EW-8 vs .DEQAR-0.5-2-1	0.000059
13	APRIORI-EW-6 vs .DEQAR-0.4-2-1	0.000150
14	APRIORI-EW-9 vs .DEQAR-0.4-3-2	0.000258
15	APRIORI-EW-6 vs .DEQAR-0.5-2-1	0.000401
16	APRIORI-EF-6 vs .DEQAR-0.4-3-2	0.000519
17	APRIORI-EF-4 vs .DEQAR-0.4-2-1	0.000562
18	APRIORI-EF-4 vs .DEQAR-0.5-2-1	0.001439
19	APRIORI-EW-8 vs .DEQAR-0.4-3-2	0.002774
20	APRIORI-EF-3 vs .DEQAR-0.4-2-1	0.004464
21	APRIORI-EF-3 vs .DEQAR-0.5-2-1	0.010440
22	APRIORI-EW-6 vs .DEQAR-0.4-3-2	0.013989
23	APRIORI-EW-3 vs .DEQAR-0.4-2-1	0.014924
24	APRIORI-EW-3 vs .DEQAR-0.5-2-1	0.032936
25	APRIORI-EF-4 vs .DEQAR-0.4-3-2	0.040287
26	APRIORI-EW-4 vs .DEQAR-0.4-2-1	0.040287
27	APRIORI-EW-4 vs .DEQAR-0.5-2-1	0.083657

Tabla A.39: p -valores ajustados del test de Holm (cobertura, mejores 25 reglas).

i	comparación	pHolm
1	APRIORI-EW-9 vs .DEQAR-0.5-2-1	0
2	APRIORI-EW-9 vs .DEQAR-0.4-2-1	0.000001
3	APRIORI-EW-3 vs .DEQAR-0.5-2-1	0.000002
4	APRIORI-EW-3 vs .DEQAR-0.4-2-1	0.000005
5	APRIORI-EW-4 vs .DEQAR-0.5-2-1	0.000005
6	APRIORI-EW-6 vs .DEQAR-0.5-2-1	0.000006
7	APRIORI-EW-8 vs .DEQAR-0.5-2-1	0.000006
8	APRIORI-EW-4 vs .DEQAR-0.4-2-1	0.000012
9	APRIORI-EF-2 vs .DEQAR-0.5-2-1	0.000014
10	APRIORI-EW-6 vs .DEQAR-0.4-2-1	0.000014
11	APRIORI-EW-8 vs .DEQAR-0.4-2-1	0.000014
12	APRIORI-EW-2 vs .DEQAR-0.5-2-1	0.000024
13	APRIORI-EF-2 vs .DEQAR-0.4-2-1	0.000032
14	APRIORI-EW-2 vs .DEQAR-0.4-2-1	0.000057
15	APRIORI-EF-4 vs .DEQAR-0.5-2-1	0.000233
16	APRIORI-EF-9 vs .DEQAR-0.5-2-1	0.000304
17	APRIORI-EF-6 vs .DEQAR-0.5-2-1	0.000360
18	APRIORI-EF-4 vs .DEQAR-0.4-2-1	0.000509
19	APRIORI-EF-9 vs .DEQAR-0.4-2-1	0.000657
20	APRIORI-EF-6 vs .DEQAR-0.4-2-1	0.000775
21	APRIORI-EF-3 vs .DEQAR-0.5-2-1	0.000912
22	APRIORI-EF-3 vs .DEQAR-0.4-2-1	0.001935
23	APRIORI-EF-8 vs .DEQAR-0.5-2-1	0.011910
24	APRIORI-EF-8 vs .DEQAR-0.4-2-1	0.022988
25	APRIORI-EW-9 vs .DEQAR-0.4-3-2	0.053194

Tabla A.40: p -valores ajustados del test de Holm (atributos, mejores 25 reglas).

Bibliografía

[Agapito et al., 2015] Agapito, G., Guzzi, P. H. y Cannataro, M. (2015). *DMET-Miner: Efficient discovery of association rules from pharmacogenomic data*, Journal of Biomedical Informatics, vol. 56, pp. 273-283.

[Agrawal et al., 1993] Agrawal, R., Imielinski, T. y Swami, A. (1993). *Mining association rules between sets of items in large databases*, Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, pp. 207-216.

[Agrawal et al., 1994] Agrawal, R. y Srikant, R. (1994). *Fast algorithms for mining association rules in large databases*, Proceedings of the 20th International Conference on Very Large DataBases, San Francisco, pp. 487-499.

[Ahmed et al., 2013] Ahmed, A. A., Egleston, B., Alcantara, P., Li, L., Pollack, A., Horwitz, E.M. y Buyyounouski, M.K. (2013). *A novel method for predicting late genitourinary toxicity after prostate radiation therapy and the need for age-based risk-adapted dose constraints*, International Journal of Radiation Oncology, Biology, Physics, 86(4):709-15.

[Alcalá-Fdez et al., 2011] Alcalá-Fdez, J., Fernandez, A., Luengo, J., Derrac, J., García, S., Sánchez, L. y Herrera, F. (2011). *KEEL Data-Mining Software Tool: Data Set Repository, Integration of Algorithms and Experimental Analysis Framework*, Journal of Multiple-Valued Logic and Soft Computing, 17:2-3, pp. 255-287.

[An et al., 2001] An, A. y Cercone, N. (2001). *Rule Quality Measures for Rule Induction Systems: Description and Evaluation*, Computational Intelligence, 17, pp. 409-424.

[Arlot et al., 2010] Arlot, S. y Celisse, A. (2010). *A survey of cross-validation procedures for model selection*, Statistics Surveys, vol. 4, pp. 40-79.

[Artstein et al., 2008] Artstein, R. y Poesio, M. (2008). *Inter-coder agreement for computational linguistics*, Computational Linguistics 34(4):555-596.

[Asuncion et al., 2007] Asuncion, A. y Newman, D.J. (2007). *UCI Machine Learning Repository*, <http://www.ics.uci.edu/~mlern/MLRepository.html>, Irvine, University of California, School of Information and Computer Science.

- [Atzmueller et al., 2006] Atzmueller, M. y Puppe, F. (2006). *SD-Map – A Fast Algorithm for Exhaustive Subgroup Discovery*. Knowledge Discovery in Databases, Lecture Notes in Computer Science, vol. 4213, Springer.
- [Bergmann et al., 1988] Bergmann, G. y Hommel, G. (1988). *Improvements of general multiple test procedures for redundant systems of hypotheses*, Multiple Hypotheses Testing, pp. 100-115. Springer.
- [Boudík et al., 2004] Boudík, F., Tome, M. y Bultas, J. (2004). *STULONG medical project*, <http://euromise.vse.cz/challenge2004>, Prague.
- [Boullé et al., 2018] Boullé, M. (2018). *Hierarchical two-part MDL code for multinomial distributions*, International Journal of Approximate Reasoning, vol. 103, pp. 71-93.
- [Breiman et al., 1984] Breiman, L., Friedman, J. H., Olshen, R. A. y Stone, C. J. (1984). *Classification and regression trees*, Wadsworth, Belmont, CA.
- [Breiman, 2001] Breiman, L. (2001). *Random Forests*, Machine Learning, 45(1):5-32.
- [Brin et al., 1997a] Brin, S., Motwani, R., y Silverstein, C. (1997). *Beyond market baskets: generalizing association rules to correlations*, Proceedings of the 1997 ACM SIGMOD international conference on Management of data, vol. 26, pp. 265-276.
- [Brin et al., 1997b] Brin, S., Motwani, R., Ullman, J. D., y Tsur, S. (1997). *Dynamic itemset counting and implication rules for market basket data*, Proc. of the ACM SIGMOD, pp. 265-276.
- [Campos et al., 2016] Campos, G.O., Zimek, A., Sander, J. et al. (2016). *On the evaluation of unsupervised outlier detection: measures, datasets, and an empirical study*, Data Mining and Knowledge Discovery, 30(4):891-927.
- [Carmona et al., 2010] Carmona, C.J., González, P., del Jesus, M.J. y Herrera, F. (2010). *NMEEF-SD: non-dominated multi-objective evolutionary algorithm for extracting fuzzy rules in subgroup discovery*, IEEE Trans. on Fuzzy Systems, 18 (5), pp. 958-970.
- [Carmona et al., 2015] Carmona, C.J., Ruiz-Rodado, V., del Jesus, M.J., Weber, A., Grootveld, M., Gonzalez, P. y Elizondo, D. (2015). *A fuzzy genetic programming-based algorithm for subgroup discovery and the application to one problem of pathogenesis of acute sore throat conditions in humans*, Information Sciences, vol. 298, pp. 180-197.

- [Carmona-Saez et al., 2006] Carmona-Saez, P., Chagoyen, M., Rodríguez, A., Trelles, O., Carazo, J. M. y Pascual-Montano, A. (2006). *Integrated analysis of gene expression by association rules discovery*, BMC Bioinformatics, 7:54.
- [Chandola et al., 2009] Chandola, V., Banerjee, A., y Kumar, V. (2009). *Anomaly detection: A survey*, ACM Computing Surveys, 41:1-58.
- [Chen et al., 2016] Chen, Y. L., Wu, C. C. y Tang, K. (2016). *Time-constrained cost-sensitive decision tree induction*, Information Sciences, vol. 354, pp. 140-152.
- [Chmielewski et al., 1996] Chmielewski, M.R. y Grzymala-Busse, J.W. (1996). *Global discretization of continuous attributes as preprocessing for machine learning*, International Journal of Approximate Reasoning, 15(4):319-331.
- [Cohen, 1960] Cohen, J. (1960). *A coefficient of agreement for nominal scales*, Educational and Psychological Measurement, 20, pp. 37-46.
- [Cortes et al., 1995] Cortes, C. y Vapnik, V. N. (1995). *Support-vector networks*, Machine Learning, 20(3):273-297.
- [Delgado et al., 2018] Delgado, R., Núñez-González, J. D. (2018). *Enhancing Confusion Entropy as Measure for Evaluating Classifiers*, Proceedings of the International Joint Conference SOCO'18-CISIS'18-ICEUTE'18, Advances in Intelligent Systems and Computing, vol. 771, pp. 79-89, Springer.
- [del Jesus et al., 2007] del Jesus, M.J., González, P., Herrera, F. y Mesonero, M. (2007). *Evolutionary fuzzy rule induction process for subgroup discovery: a case study in marketing*, IEEE Trans. on Fuzzy Systems, 15(4):578-592.
- [Demšar, 2006] Demšar, J. (2006). *Statistical comparisons of classifiers over multiple data sets*, Journal of Machine Learning Research, vol. 7, pp. 1-30.
- [Dobson, 2001] Dobson, A. J. (2001). *An Introduction to Generalized Linear Models*, Chapman & Hall, New York.
- [Driver et al., 1932] Driver, H. E. y Kroeber, A. L. (1932). *Quantitative expression of cultural relationships*, University of California Press, pp. 211-256.
- [Easterly et al., 1997] Easterly, W. y Levine, R. (1997). *Africa's Growth Tragedy: Policies and Ethnic Divisions*, The Quarterly Journal of Economics, 112(4):1203-1250.

- [Fayyad et al., 1993] Fayyad U. M. y Irani, K. B. (1993). *Multi-interval discretization of continuous-valued attributes for classification learning*, 13th International Joint Conference on Artificial Intelligence, pp. 1022–1029.
- [Fayyad et al., 1996] Fayyad, U., Piatetsky-Shapiro, G. y Smyth, P.J. (1996). *From Data Mining to Knowledge Discovery in Databases*, *AI Magazine* 17(3):37-54.
- [Filipovych et al., 2011] Filipovych, R, Resnick, S. M. y Davatzikos, C. (2011). *Semi-supervised Cluster Analysis of Imaging Data*, *NeuroImage*. 54(3): 2185-2197.
- [Frank et al., 1998] Frank, E. y Witten, I. H. (1998). *Generating Accurate Rule Sets Without Global Optimization*, Proc. Fifteenth International Conference on Machine Learning, pp. 144-151.
- [Friedman, 1937] Friedman, M. (1937). *The use of ranks to avoid the assumption of normality implicit in the analysis of variance*, *Journal of the American Statistical Association*, 32:675-701.
- [Gamberger et al., 2002] Gamberger, D. y Lavrač, N. (2002). *Expert-Guided Subgroup Discovery: Methodology and Application*, *Journal Of Artificial Intelligence Research*, vol. 17, pp. 501-527.
- [García et al., 2008] García, S. y Herrera, F. (2008). *An Extension on "Statistical Comparisons of Classifiers over Multiple Data Sets" for all Pairwise Comparisons*, *Journal of Machine Learning Research*, vol. 9, pp. 2677-2694.
- [García et al. 2010] García, S., Derrac, J. y Herrera, F. (2010). *Un Tutorial Metodológico para hacer Comparaciones Estadísticas con Tests No Paramétricos en Propuestas de Minería de Datos*, V Simposio de Teoría y Aplicaciones de Minería de Datos (TAMIDA).
- [Geng et al., 2006] Geng, L. y Hamilton, H. J. (2006). *Interestingness measures for data mining: A survey*, *ACM Computing Surveys*, 38(3):9.
- [Govada et al., 2017] Govada, A., Patluri, A. y Honnalgere, A. (2017). *Association Rule Mining using Apriori for Large and Growing Datasets under Hadoop*, Proceedings of the VI International Conference on Network, Communication and Computing, pp. 14-17.
- [Grosskreutz et al., 2009] Grosskreutz, H. y Rüping, S. (2009). *On Subgroup Discovery in Numerical Domains*, *Data Mining and Knowledge Discovery*, 19(2), pp. 210-226.

- [Guillet et al., 2007] Guillet, F. y Hamilton, H. J. (2007). *Quality Measures in Data Mining*, Studies in Computational Intelligence, vol. 43.
- [Güvenir et al., 2000] Güvenir, H. A. y Uysal, I. (2000). *Bilkent University Function Approximation Repository*, <http://funapp.cs.bilkent.edu.tr>.
- [Hájek et al., 1966] Hájek, P., Havel, I. y Chytil, M. (1966). *The GUHA method of automatic hypotheses determination*, Computing, pp. 293-308.
- [Hall et al., 2009] Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P. y Witten, I. H. (2009). *The WEKA Data Mining Software: An Update*, SIGKDD Explorations, 11(1).
- [Han et al., 2004] Han, J., Pei, J., Yin, Y. y Mao, R. (2004). *Mining Frequent Patterns without Candidate Generation: A Frequent-Pattern Tree Approach*, Data Mining and Knowledge Discovery, 8(1):53-87.
- [Han et al., 2012] Han, J., Kamber, M. y Pei, J. (2012). *Data Mining: Concepts and Techniques*, Morgan Kaufmann Publishers.
- [Hastie et al., 2009] Hastie, T., Tibshirani, R. y Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, Springer-Verlag.
- [Ho, 1995] Ho, T. K. (1995). *Random Decision Forests*, Proceedings of the 3rd International Conference on Document Analysis and Recognition, pp. 278-282.
- [Holm, 1979] Holm, S. (1979). *A simple sequentially rejective multiple test procedure*. Scandinavian Journal of Statistics, 6:65-70.
- [Ibarguren et al., 2016] Ibarguren, I., Lasarguren, A., Pérez, J. M. et al. (2016). *BFPART: Best-First PART*, Information Sciences, vol. 367–368, pp. 927-952.
- [Iman et al., 1980] Iman, R. L. y Davenport, J. M. (1980). *Approximations of the critical region of the Friedman statistic*, Communications in Statistics, pp. 571-595.
- [Jitkajornwanich et al., 2017] Jitkajornwanich, K. et al. (2017). *Ocean surface current prediction based on HF radar observations using trajectory-oriented association rule mining*, IEEE International Conference on Big Data, Boston, MA, pp. 4293-4300.
- [John et al., 1995] John, G. H. y Langley, P. (1995). *Estimating Continuous Distributions in Bayesian Classifiers*, Proc. Eleventh Conference on Uncertainty in Artificial Intelligence, pp. 338-345.

- [Kavšek et al., 2003] Kavšek B., Lavrač N. y Jovanoski V. (2003). *APRIORI-SD: Adapting Association Rule Learning to Subgroup Discovery*, Advances in Intelligent Data Analysis V, Lecture Notes in Computer Science, vol. 2810, Springer.
- [Klösgen, 1996] Klösgen, W. (1996). *Explora: A multipattern and multistrategy discovery assistant*, Advances in Knowledge Discovery and Data Mining, American Association for Artificial Intelligence, pp. 249-271.
- [Lavrač et al., 1999] Lavrač, N., Flach, P. A. y Zupan, B. (1999). *Rule evaluation measures: A unifying view*, Proceedings of the 9th International Workshop on Inductive Logic Programming, London, Springer-Verlag, pp. 174-185.
- [Lavrač et al., 2004] Lavrač, N., Kavšek, B., Flach, P. A. y Todorovski, L. (2004). *Subgroup discovery with CN2-SD*, Journal of Machine Learning Research, vol. 5, pp. 153-188.
- [Le-Cessie et al., 1992] Le-Cessie, S. y Van Houwelingen, J. C. (1992). *Ridge Estimators in Logistic Regression*, Applied Statistics, 41(1):191-201.
- [Lenca et al., 2008] Lenca, P., Meyer, P., Vaillant, B. y Lallich, S. (2008). *On selecting interestingness measures for association rules: User oriented description and multiple criteria decision aid*, European Journal of Operational Research, 184(2):610-626.
- [Li et al., 2001] Li, W., Han, J. y Pei, J. (2001). *CMAR: accurate and efficient classification based on multiple class-association rules*, Proceedings IEEE International Conference on Data Mining, pp. 369-376.
- [Liu et al., 1998] Liu, B., Hsu, W. y Ma, Y. (1998). *Integrating classification and association rule mining*, Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining, pp. 80-86.
- [Liu et al., 1999] Liu, B., Hsu, W., Mun, L. y Lee, H. (1999). *Finding interesting patterns using user expectations*, IEEE Transactions on Knowledge and Data Engineering, 11(6):817-832.
- [Liu et al., 2002] Liu, H., Hussain, F., Tan, C.L. et al. (2002). *Discretization: An Enabling Technique*, Data Mining and Knowledge Discovery, 6:393.
- [Lopez et al., 2013] Lopez, J.L., Isa, N., Matute, R., Russo, M., Puebla, F. y Miran, K.M. (2013). *Hypofractionated helical tomotherapy using 2.5-2.6 Gy daily fractions for localized prostate cancer*, Clinical and Translational Oncology, 15(4): 271-277.

[Lopez et al., 2015] Lopez Guerra, J.L., Matute, R., Puebla, F., et al. (2015). *Ethnic difference in risk of toxicity in prostate cancer patients treated with dynamic arc radiation therapy*, Tumori, pp. 461-468.

[Luna et al., 2014] Luna, J.M., Romero, J.R., Romero, C. y Ventura, S.. (2014). *On the use of genetic programming for mining comprehensible rules in subgroup discovery*, IEEE Transactions on Cybernetics, 44 (12):2329-2341.

[Matthews, 1975] Matthews, B. W. (1975). *Comparison of the predicted and observed secondary structure of T4 phage lysozyme*, Biochimica et Biophysica Acta, 405(2):442-451.

[Pachón et al., 2011] Pachón, V., Mata, J., Domínguez, J.L., Maña, M.J. (2011). *Multi-objective Evolutionary Approach for Subgroup Discovery*. Hybrid Artificial Intelligent Systems, Lecture Notes in Computer Science, vol. 6679, Springer.

[Pearl, 1985] Pearl, J. (1985). *Bayesian Networks: A Model of Self-Activated Memory for Evidential Reasoning*, UCLA Computer Science Department.

[Piatetsky-Shapiro, 1991] Piatetsky-Shapiro, G. (1991). *Discovery, analysis and presentation of strong rules*, Knowledge Discovery in Databases, pp. 229-248.

[Quinlan, 1986] Quinlan, J. R. (1986). *Induction of Decision Trees*, Machine Learning 1:81-106.

[Quinlan, 1993] Quinlan, J. R. (1993). *C4.5: Programs for Machine Learning*, Morgan Kaufmann Publishers, San Mateo, CA.

[Rajesh et al., 2018] Rajesh, N. y Selvakumar, A.A.L. (2018). *Association rules and deep learning for cryptographic algorithm in privacy preserving data mining*, Cluster Computing, pp. 1-13.

[Rauch, 2018] Rauch, J. (2018). *Expert deduction rules in data mining with association rules: a case study*, Knowledge and Information Systems, pp. 1-29.

[Rish, 2001] Rish, I. (2001). *An empirical study of the naive Bayes classifier*, IJCAI 2001 workshop on empirical methods in artificial intelligence, pp. 41-46.

[Rissanen, 1978] Rissanen, J. (1978). *Modeling by shortest data description*, Automatica, 14 (5), pp. 465-658.

[Rodríguez et al., 2012] Rodríguez, D., Ruiz, R., Riquelme, J.C. y Aguilar-Ruiz, J.S. (2012). *Searching for rules to detect defective modules: a subgroup discovery approach*, Information Sciences, 191:14-30.

[Rosenblatt, 1958] Rosenblatt, F. (1958). *The Perceptron: A Probabilistic Model For Information Storage And Organization In The Brain*, Psychological Review, 65(6):386-408.

[Rumelhart et al., 1986] Rumelhart, D. E., Hinton, G. E. y Williams, R. J. (1986). *Parallel distributed processing: explorations in the microstructure of cognition*, MIT Press Cambridge, pp. 318-362.

[Silberschatz et al., 1995] Silberschatz, A. y Tuzhilin, A. (1995). *On subjective measures of interestingness in knowledge discovery*, First International Conference on Knowledge Discovery and Data Mining, pp. 275-281.

[Silberschatz et al., 1996] Silberschatz, A. y Tuzhilin, A. (1996). *What makes patterns interesting in knowledge discovery systems*, IEEE Transactions on Knowledge and Data Engineering, 8(6):970-974.

[Srikant et al., 1996] Srikant, R. y Agrawal, R. (1996). *Mining Quantitative Association Rules in Large Relational Tables*, Proceedings of the ACM SIGMOD, 1-12.

[Topuz et al., 2018] Topuz, K., Zengul, F. D., Dag, A. et al. (2018). *Predicting graft survival among kidney transplant recipients: A Bayesian decision support model*, Decision Support Systems, vol. 106, pp. 97-109.

[Weiss et al., 1991] Weiss, S. M. y Kulikowski, C. (1991). *Computer Systems that Learn: Classification and Prediction Methods from Statistics, Neural Nets, Machine Learning, and Expert Systems*, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

[Wijsen et al., 1998] Wijsen, J., y Meersman, R. (1998). *On the Complexity of Mining Quantitative Association Rules*, Data Mining and Knowledge Discovery, 2, 263-281.

[Wilcoxon, 1945] Wilcoxon, F. (1945). *Individual comparisons by ranking methods*, Biometrics, vol. 1, pp. 80–83.

[Wrobel, 1997] Wrobel, S. (1997). *An algorithm for multi-relational discovery of subgroups*, Proc. of the 1st European Conference on Principles of Data Mining and Knowledge Discovery, pp. 78–87.

[Zaki et al., 1997] Zaki, M. J., Parthasarathy, S., Ogihara, M. y Li, W. (1997). *New algorithms for fast discovery of association rules*, Proceedings of the Third International Conference on Knowledge Discovery and Data Mining, pp. 283-286.

[Zhang et al., 2012] Zhang, A., Feng, J., Ward, R.D., Wan, P., Gao, Q. et al. (2012). *A New Method for Species Identification via Protein-Coding and Non-Coding DNA Barcodes by Combining Machine Learning with Bioinformatic Methods*, PLOS ONE 7(2):e30986.

[Zorita et al., 2015] Zorita, E., Cuscó, P. y Fillion, G.J. (2015). *Starcode: sequence clustering based on all-pairs search*, Bioinformatics, 31 (12):1913-1919.

[Zubin, 1938] Zubin, J. (1938). *A technique for measuring likemindedness*, Journal of Abnormal Psychology, 33(4):508-516.