



Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

**DISEÑO DE UNA INTERFAZ GRÁFICA PARA EL ANÁLISIS DE REGLAS DE
ASOCIACIÓN NUMÉRICAS EN LA UNIVERSIDAD DE HUELVA, ESPAÑA**

Nathán Franciné Soto Méndez

Asesorado por el Ing. Marlon Antonio Pérez Türk

Guatemala, junio de 2016

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA



FACULTAD DE INGENIERÍA

DISEÑO DE UNA INTERFAZ GRÁFICA PARA EL ANÁLISIS DE REGLAS DE ASOCIACIÓN NUMÉRICAS EN LA UNIVERSIDAD DE HUELVA, ESPAÑA

TRABAJO DE GRADUACIÓN

PRESENTADO A LA JUNTA DIRECTIVA DE LA
FACULTAD DE INGENIERÍA

POR

NATHÁN FRANCINÉ SOTO MÉNDEZ

ASESORADO POR EL ING. MARLON ANTONIO PÉREZ TÜRK

AL CONFERÍRSELE EL TÍTULO DE

INGENIERO EN CIENCIAS Y SISTEMAS

GUATEMALA, JUNIO DE 2016

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA



NÓMINA DE JUNTA DIRECTIVA

DECANO	Ing. Pedro Antonio Aguilar Polanco
VOCAL I	Ing. Angel Roberto Sic García
VOCAL II	Ing. Pablo Christian de León Rodríguez
VOCAL III	Inga. Elvia Miriam Ruballos Samayoa
VOCAL IV	Br. Raúl Eduardo Ticún Córdova
VOCAL V	Br. Henry Fernando Duarte García
SECRETARIA	Inga. Lesbia Magalí Herrera López

TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO

DECANO	Ing. Pedro Antonio Aguilar Polanco
EXAMINADORA	Inga. Floriza Felipa Ávila Pesquera de Medinilla
EXAMINADOR	Ing. Marlon Antonio Pérez Türk
EXAMINADORA	Inga. Susan Verónica Gudiel Herrera
SECRETARIA	Inga. Lesbia Magalí Herrera López

HONORABLE TRIBUNAL EXAMINADOR

En cumplimiento con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presento a su consideración mi trabajo de graduación titulado:

DISEÑO DE UNA INTERFAZ GRÁFICA PARA EL ANÁLISIS DE REGLAS DE ASOCIACIÓN NUMÉRICAS EN LA UNIVERSIDAD DE HUELVA, ESPAÑA

Tema que me fuera asignado por la Dirección de la Escuela de Ingeniería en Ciencias y Sistemas, con fecha 11 de marzo del 2015.

Nathán Franciné Soto Méndez

Universidad de San Carlos de Guatemala
Facultad de Ingeniería



Guatemala, 28 de abril de 2016

Facultad de Ingeniería
Universidad de San Carlos de Guatemala
Escuela de Ciencias y Sistemas
Presente

Por este medio hago constar como asesor del proyecto de graduación del estudiante **Nathán Franciné Soto Méndez**, quien se identifica con el carnet universitario **200714365**, de la Carrera de Ingeniería en Ciencias y Sistemas, procedí a revisar la tesis, cuyo título es: **"DISEÑO DE UNA INTERFAZ GRÁFICA PARA EL ANÁLISIS DE REGLAS DE ASOCIACIÓN NUMÉRICAS EN LA UNIVERSIDAD DE HUELVA, ESPAÑA"**.

En tal virtud, **LO DOY POR APROBADO**, solicitándole darle el trámite respectivo.

Sin otro particular, me es grato suscribirme.

Atentamente,

*Marlon Antonio Pérez Türk
Ingeniero en Ciencias y Sistemas
Colegiado 4492*

Ing. Marlon Antonio Pérez Türk
Asesor



Universidad San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

Guatemala, 11 de Mayo de 2016

Ingeniero
Marlon Antonio Pérez Türk
Director de la Escuela de Ingeniería
En Ciencias y Sistemas

Respetable Ingeniero Pérez:

Por este medio hago de su conocimiento que he revisado el trabajo de graduación del estudiante **NATHÁN FRANCINÉ SOTO MENDEZ** con carné **200714365**, titulado: **"DISEÑO DE UNA INTERFAZ GRAFICA PARA EL ANALISIS DE REGLAS DE ASOCIACIÓN NUMERICAS EN LA UNIVERSIDAD DE HUELVA, ESPAÑA"**, y a mi criterio el mismo cumple con los objetivos propuestos para su desarrollo, según el protocolo.

Al agradecer su atención a la presente, aprovecho la oportunidad para suscribirme,

Atentamente,


Ing. Carlos Alfredo Azurdia
Coordinador de Privados
y Revisión de Trabajos de Graduación



E
S
C
U
E
L
A

D
E

I
N
G
E
N
I
E
R
Í
A

E
N

C
I
E
N
C
I
A
S

Y

S
I
S
T
E
M
A
S

UNIVERSIDAD DE SAN CARLOS
DE GUATEMALA



FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA EN
CIENCIAS Y SISTEMAS
TEL: 24767644

*El Director de la Escuela de Ingeniería en Ciencias y Sistemas de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer el dictamen del asesor con el visto bueno del revisor y del Licenciado en Letras, del trabajo de graduación **“DISEÑO DE UNA INTERFAZ GRÁFICA PARA EL ANÁLISIS DE REGLAS DE ASOCIACIÓN NUMÉRICAS EN LA UNIVERSIDAD DE HUELVA, ESPAÑA”**, realizado por el estudiante **NATHÁN FRANCINÉ SOTO MÉNDEZ**, aprueba el presente trabajo y solicita la autorización del mismo.*

“ID Y ENSEÑAD A TODOS”

*Ing. **Marlon Antonio Pérez Türk**
Director*

Escuela de Ingeniería en Ciencias y Sistemas



Guatemala, 09 de junio de 2016

Universidad de San Carlos
de Guatemala



Facultad de Ingeniería
Decanato

Ref.DTG.D.291.2016

El Decano de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer la aprobación por parte del Director de la Escuela de Ingeniería en Ciencias y Sistemas, al trabajo de graduación titulado: **DISEÑO DE UNA INTERFAZ GRÁFICA PARA EL ANÁLISIS DE REGLAS DE ASOCIACIÓN NUMÉRICAS EN LA UNIVERSIDAD DE HUELVA, ESPAÑA**, presentado por el estudiante universitario: **Nathán Franciné Soto Méndez**, y después de haber culminado las revisiones previas bajo la responsabilidad de las instancias correspondientes, se autoriza la impresión del mismo.

IMPRÍMASE.

Ing. Pedro Antonio Aguilar Polanco
Decano



Guatemala, junio de 2016

/cc

ACTO QUE DEDICO A:

Dios	Por estar en cada momento.
Mis padres	Aroldo Franciné Soto y Carmen Méndez, por darme la vida y la educación.
Mis hermanas	Caroline y Sheily Soto Méndez, por la compañía y apoyo.
Mi familia	Por ser personas especiales conmigo y apoyarme siempre.
Mis amigos	Por el compañerismo y tiempo compartido durante el proceso de formación académica.

AGRADECIMIENTOS A:

**Universidad de San
Carlos de Guatemala**

Por ser la casa de estudios que me ha formado académicamente.

**Universidad de
Huelva, España**

Por recibirme y brindarme apoyo para realizar intercambio académico durante 6 meses.

**Centro de Estudios
del Mar y Acuicultura
de la Universidad de
San Carlos**

Por apoyarme en realizar mi Ejercicio Profesional Supervisado.

Ing. Marlon Pérez Türk

Por guiar y asesorarme en el informe final de graduación.

2.2.	Estructura.....	15
2.2.1.	Importar datos y reglas.....	16
2.2.1.1.	DataSet	16
2.2.1.1.1.	Primera línea.....	16
2.2.1.1.2.	Segundo bloque	16
2.2.1.1.3.	Tercer bloque	17
2.2.1.2.	Rules	17
2.2.1.2.1.	Primera línea.....	17
2.2.1.2.2.	Segunda línea	18
2.2.2.	Presentar las reglas de forma interactiva y gráfica.....	18
2.2.3.	Exportar nuevas reglas.....	19
2.3.	Modelo de requisitos	20
2.3.1.	Requisitos de información	20
2.3.2.	Requisitos funcionales	21
2.3.2.1.	Objetivos	21
2.3.2.2.	Diagrama de casos de uso.....	23
2.3.2.3.	Casos de uso	24
2.3.2.4.	Actores	26
2.3.3.	Requisitos no funcionales.....	26
2.3.4.	Modelo del diseño	27
2.3.4.1.	Diagrama de clases.....	27
2.3.4.2.	Clases	29
2.3.5.	Despliegue del sistema	33
3.	FASE ENSEÑANZA APRENDIZAJE	35
3.1.	Manual de usuario.....	35
3.1.1.	Inicio y carga de ficheros.....	35
3.1.2.	Presentación de reglas.....	37

3.1.3.	Tipos de reglas visualizadas.....	39
3.1.4.	Modificación de condiciones.....	41
3.1.5.	Eliminación de condiciones.....	42
3.1.6.	Botón favoritas.....	44
3.1.7.	Guardar favoritas	44
4.	FASE DE PRESENTACIÓN DE LA INTERFAZ.....	47
4.1.	Acta de evaluación y presentación del proyecto.....	47
	CONCLUSIONES	49
	RECOMENDACIONES.....	51
	BIBLIOGRAFÍA.....	53

ÍNDICE DE ILUSTRACIONES

FIGURAS

1.	Arquitectura Sistema Minería de Datos.....	5
2.	Representación tabular	12
3.	Representación 2D.....	12
4.	Visualización 3D.....	13
5.	Dos pisos	13
6.	Visualización en grafos	14
7.	Visualización en coordenadas paralelas	14
8.	Manejo reglas.....	19
9.	Modificación condiciones	19
10.	Exportar nuevas reglas	20
11.	Diagrama de casos de uso a.....	24
12.	Diagrama de casos de uso b.....	28
13.	Despliegue del sistema	33
14.	Inicio.....	36
15.	Cargar ficheros.....	36
16.	Ejecutar	37
17.	Presentación reglas.....	37
18.	Gestionar reglas	38
19.	Área de condiciones.....	38
20.	Área estadísticos.....	39
21.	Vista originales.....	39
22.	Vista favoritas.....	40
23.	Botón favoritas	40

24.	Botón originales	40
25.	Condición original	41
26.	Modificar condición con <i>click</i>	41
27.	Modificación con teclado.....	41
28.	Pulsando enter.....	42
29.	Valores iguales	42
30.	Antes de eliminar	43
31.	Después de eliminar	43
32.	Botón de cambios	44
33.	Botón de favoritas	44
34.	Guardar favoritas	45
35.	Exportar fichero.....	45
36.	Acta de evaluación del proyecto	48

TABLAS

I.	Requisitos de información.....	21
II.	Objetivo del sistema: cargar archivos a	22
III.	Objetivo del sistema: cargar archivos b	22
IV.	Objetivo del sistema: gestionar cambios en reglas	22
V.	Objetivo del sistema: listar reglas favoritas	23
VI.	Objetivo del sistema: exportar reglas favoritas	23
VII.	Caso de uso: cargar archivos	24
VIII.	Caso de uso: modificar reglas	25
IX.	Caso de uso: agregar favoritas	25
X.	Caso de uso: exportar reglas	26
XI.	Actor analista	26
XII.	Requisito no funcional: utilización colores adecuados.....	27

LISTA DE SÍMBOLOS

Símbolo	Significado
°	Grados
%	Porcentaje

GLOSARIO

Base de datos	Conjunto de datos que pertenecen al mismo contexto, almacenados sistemáticamente. En una base de datos, la información se organiza en campos y registros. Los datos pueden aparecer en forma de texto, números, gráficos, sonido o vídeo.
Data	Es toda la información que agrega valor a la organización. Son datos que son importantes para la organización.
DER	Diagrama entidad relación.
Diagrama entidad relación	Es visualizar gráficamente la estructura de la base de datos utilizada en la herramienta como tal de la universidad virtual. Todas las relaciones que se tienen en las diferentes tablas utilizadas e implementadas en este proceso.
Hardware	Nombre con el que se designa a los componentes físicos y tangibles de los sistemas informáticos.
Herramienta informática	Subprograma o módulo informático encargado de funciones específicas y afines entre sí para realizar una tarea.

UHU

Universidad de Huelva.

RESUMEN

En la actualidad las organizaciones se enfrentan a un mundo cada vez más competitivo. Por tanto, las estrategias de administración deben ser flexibles para adaptarse a las condiciones cambiantes del entorno.

Lo anterior supone un enorme reto para las organizaciones, en especial en el manejo de grandes volúmenes de información para conocer el entorno y predecir su evolución. En otras palabras es necesario conocer quiénes son los principales clientes, qué tipo de servicios son los que más consumen, cómo se agrupan estos clientes, cómo es su patrón de compra, qué segmento de mercado se desaprovechan, quiénes son los principales competidores, en qué mercado se desempeñan, cuáles son sus puntos débiles y fuertes, y otros.

La principal motivación de la minería de datos son las predicciones. Estas pueden ser triviales. Sin embargo, a mayor cantidad de datos los modelos que se generan pueden descubrir patrones interesantes que no se ven a simple vista.

El objetivo de este proyecto es la creación de una interfaz gráfica desarrollada en lenguaje de programación Java. Este permite visualizar y analizar selectivamente las reglas resultantes de una fase previa de extracción de reglas de asociación numéricas a partir de un conjunto de datos. Por tanto, haría uso de un fichero con las reglas previamente generadas y de otro con el correspondiente conjunto de datos. A partir de ellos, y con la selección interactiva del usuario, se mostraría información de las reglas elegidas. Además

de los cambios en las medidas de calidad de las reglas al variar sus variables e intervalos.

OBJETIVOS

General

Realizar el diseño, codificación y pruebas de una interfaz gráfica desarrollada en lenguaje de programación Java, que permita visualizar y analizar selectivamente las reglas resultantes de una fase previa de extracción de reglas de asociación numéricas a partir de un conjunto de datos. Luego una selección y modificación interactiva de las reglas, por el usuario, para generar un nuevo conjunto de datos en un fichero.

Específicos

1. Desarrollar una interfaz en lenguaje de programación Java, con aplicación de escritorio, de fácil acceso, pensada para que el usuario trabaje de forma intuitiva y pueda hacer uso de forma interactiva.
2. Cargar 2 ficheros con una estructura específica que contienen los datos y las reglas respectivamente, que serán manipuladas y tabuladas por la aplicación.
3. Brinda la utilidad de agregación de reglas favoritas, de forma que se puedan seleccionar las de mayor interés para el analista.
4. Exportar las favoritas a un fichero que posea la misma estructura que el ingresado, de forma que posteriormente se puedan visualizar dichas reglas.

INTRODUCCIÓN

Actualmente, en el siglo XXI encontrar los valores y relaciones de los atributos de los objetos son de gran importancia. Según recientes estudios avalan su actualidad e importancia y su aplicación en áreas como mercadeo, bioinformática, medicina y seguridad de redes, entre otras.

Para esto surgen las Reglas de Asociación, los cuales son parte de la técnica en la Minería de Datos, se ha visto su importancia y consiste en encontrar las asociaciones interesantes en forma de relaciones de implicación entre los valores de los atributos de los objetos de un conjunto de datos.

En la década de 1990 con una aplicación práctica, el análisis de información de ventas para el mercadeo dio origen a esta técnica. A través de ella se descubrían las relaciones a gran escala de los datos recopilados, por los sistemas de terminales de punto de venta de supermercados.

Debido a su utilización para obtener ventajas económicas, la minería de datos es parte del concepto más general de inteligencia de negocios, pero ayuda a encontrar conocimiento escondido en los datos se integró el campo de patrones.

1. FASE DE INVESTIGACIÓN

El proyecto inicia con una fase de investigación, donde se detalla todo el marco teórico donde se desarrolla el proyecto. Para ello es necesario contar con una breve introducción acerca del lugar donde se implementará el proyecto.

1.1. Antecedentes de la empresa

La Universidad de Huelva se fundó en 1993, ubicada en la provincia de Huelva, ciudad situada al suroeste de España perteneciente a la comunidad autónoma de Andalucía. Está ubicada entre mar y montaña en un territorio históricamente multicultural.

En poco más de una década se ha consolidado como una de las universidades más dinámicas y de mayor crecimiento en España. Este aumento de alumnos ha estado motivado por el elevado nivel de su profesorado, por la calidad de su red de infraestructuras y servicios, y por la variada oferta de titulaciones.

Como servicio público de educación superior, la Universidad de Huelva trabaja para formar a sus estudiantes con un profesorado muy involucrado en la docencia y con una amplia trayectoria investigadora. De hecho, la cercanía entre el profesor y el alumno es uno de los rasgos principales del centro. La mayoría de las clases se desarrollan en grupos reducidos que permiten a los alumnos acceder con más facilidad al profesorado.

La vida universitaria fusiona clases, investigación, deporte y ocio. En la Universidad de Huelva la formación en nuevas tecnologías informáticas, el aprendizaje de idiomas, las prácticas de empresa o la movilidad por otros países, constituye a la base sobre la que se asientan su oferta educativa.

1.1.1. Reseña histórica

Las Escuelas Universitarias de Ingeniería Técnica fueron creadas para disponer de un determinado tipo de técnicos que atendieran a una determinada demanda.

Durante mucho tiempo tuvieron carácter subsidiario de las correspondientes Escuelas Técnicas Superiores, pero a partir del decreto 1377/1972 del 10 de mayo, se incorporan tales Escuelas a la Universidad, adquiriendo carácter y rango universitario.

En 1901 se crea en Huelva la Escuela de Minas. Entonces Capataces y Maestros de Minas, pasan más tarde a Facultativos, luego a Peritos y, por último, a Ingenieros Técnicos.

Más tarde se incorporan las Escuelas de Forestales, Industriales y Agrícolas. La intención era atender dos sectores muy importantes en Huelva la riqueza forestal y el desarrollo industrial del Polo. Así se comenzaba con lo que se preveía como un potencial desarrollo agrícola a medio plazo.

Posteriormente, se incorpora a la Escuela Técnica Superior de Ingeniería de La Rábida (aún denominada Escuela Universitaria Politécnica de La Rábida) la Diplomatura en Informática. Esto para subsanar las necesidades de especialistas en dicha materia que tenía la sociedad onubense. El primer curso de Diplomatura en Informática se impartió en el curso 88/89.

Tras la creación, en 1993, de la Universidad de Huelva se producen dos cambios importantes en la Escuela. En primer lugar pasa a llamarse Escuela Técnica Superior de Ingeniería de La Rábida, y en segundo lugar se incorpora la Ingeniería Superior de Química. Estos son primeros estudios superiores de la Escuela que posibilita el cambio de nombre de la misma.

La Escuela Técnica Superior de Ingeniería incorpora, gran variedad de títulos, especialistas y secciones.¹

¹ Historia. *Universidad de Huelva UHU*. http://www.uhu.es/ude/MISION-VISION_UHU_2009.pdf://portal.ingenieria.usac.edu.gt/index.php/trabajador/antecedentes. Consulta: 20 de mayo de 2014.

1.1.2. Misión de la Universidad de Huelva

“Contribuir al desarrollo de la sociedad, en especial de la onubense, a través de la transmisión, generación y transferencia del conocimiento.”²

1.1.3. Visión

“Queremos ser una Universidad innovadora, emprendedora, flexible, con capacidad de dar respuesta a las demandas y cambios de su entorno -en especial el onubense-, con una relación personalizada con su alumnado y una proyección internacional. La Universidad de Huelva aspira a ser reconocida como una institución socialmente responsable, con atención preferente al medio ambiente y al desarrollo sostenible”.³

1.2. Minería de datos

Esta Data Mining es un conjunto de técnicas de aplicación tecnológica. Permite el estudio e interpretación de cantidades de información (base datos) para extraer por lo general conocimiento de interés para el usuario.

Dentro de la minería de datos hay técnicas predictivas que generan un modelo que tiene como función la predicción. Sin embargo, también se pueden usar técnicas descriptivas, que extraigan patrones o reglas que sirvan de descripción de las características generales de los datos.

² Historia. *Universidad de Huelva UHU*. http://www.uhu.es/ude/MISION-VISION_UHU_2009.pdf://portal.ingenieria.usac.edu.gt/index.php/trabajador/antecedentes. Consulta: 20 de mayo de 2014.

³ *Ibíd.*

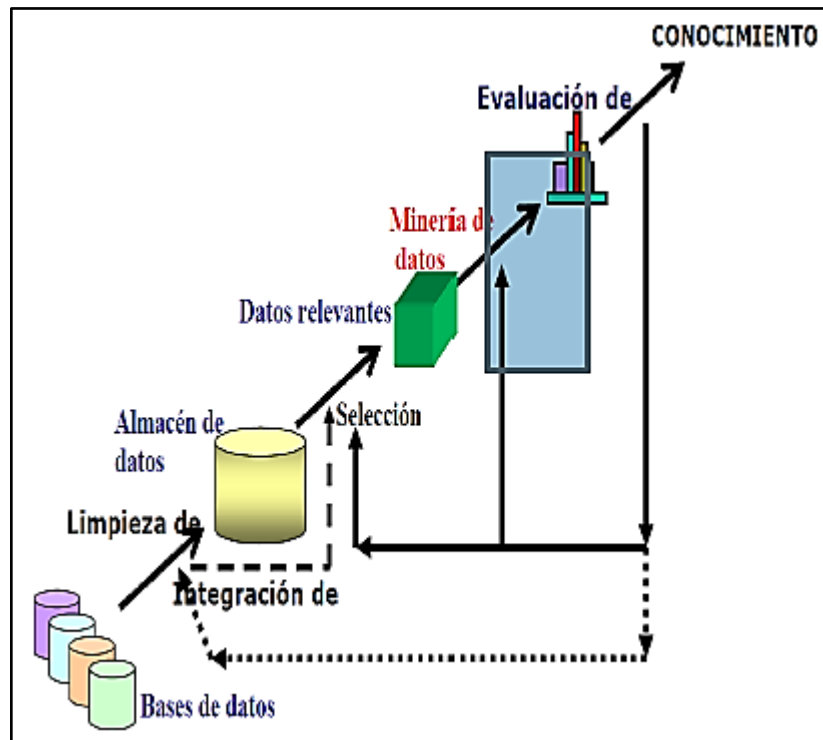
Formalmente la minería de datos se encarga de buscar modelos y patrones en datos. Los datos están almacenados de forma electrónica generalmente en base de datos y las búsquedas se pueden realizar de forma automatizada desde el ordenador.

Los patrones y modelos que se encuentran deben de ser significativos, que sean de interés para el analista de los datos. En otras palabras, la minería de datos es la extracción de patrones interesantes (no triviales, implícitos, previamente desconocidos y potencialmente útiles) o conocimiento a partir de grandes bases de datos.

Debido a su utilización e importancia, la minería de datos puede aportar también ganancias para una empresa. Esto puede pudiendo formar parte del concepto más general de inteligencia de negocios.

Un sistema de minería de datos, generalmente tiene la arquitectura de la figura 1. Donde el motor de minería de datos utiliza y aplica un algoritmo a los datos originales procedentes de la base de datos para encontrar y generar un patrón que luego los almacena en su base de datos de conocimiento, posteriormente evalúa sus resultados y una vez que tiene el modelo lo ofrece y muestra al usuario para que haga uso y pueda hacer las predicciones.

Figura 1. **Arquitectura Sistema Minería de Datos**



Fuente: Curso de Base de Datos. www.cnys.com.mx Sesión 13. Consulta: 24 de mayo de 2014.

Un sistema de minería de datos, generalmente tiene la arquitectura de la figura 1. Donde el motor de minería de datos utiliza y aplica un algoritmo a los datos originales procedentes de la base de datos para encontrar y generar un patrón que luego los almacena en su base de datos de conocimiento, posteriormente evalúa sus resultados y una vez que tiene el modelo lo ofrece y muestra al usuario para que haga uso y pueda hacer las predicciones.

Un sistema de minería de datos, generalmente tiene la arquitectura de la figura 1. Donde el motor de minería de datos utiliza y aplica un algoritmo a los datos originales procedentes de la base de datos para encontrar y generar un

patrón que luego los almacena en su base de datos de conocimiento. Posteriormente, evalúa sus resultados y una vez que tiene el modelo lo ofrece y muestra al usuario para que haga uso y pueda hacer las predicciones.

1.2.1. Conceptos y técnicas

En la minería de datos hay muchos conceptos y técnicas que se utilizan. Por ello, hay que tomar en cuenta muchos aspectos primordiales. Entre estos están.

1.2.1.1. Los datos a ser analizados

Estos datos son relacionales, que se encuentran en el almacén de datos, transaccionales, orientado a objetos, espaciales, series de tiempo, textos, multimedia, heterogéneos y datos que provienen de sistemas heredados o internet.

1.2.1.2. El conocimiento que se desea obtener

El algoritmo que será utilizado: caracterización, discriminación, asociación, clasificación, agrupación *clustering* o cálculo de tendencias/desviaciones. Funciones múltiples/integradas y minería en múltiples niveles.

1.2.1.3. Las técnicas utilizadas

Datos de tiendas departamentales (análisis del carrito de compras), compañías telefónicas, financieras, análisis de fraudes, biominería de datos, análisis de mercado, minería de la web, industria automotriz, entre otros.

1.2.2. Reglas de asociación

En el campo de la minería de datos las reglas de asociación fueron introducidas con el objetivo de encontrar relaciones (dependencias). Entre los atributos de un conjunto de datos, con el fin y objetivo de obtener información cualitativa de interés para la toma de decisiones.

“Los primeros trabajos sobre reglas de asociación se centraron en el ámbito del *marketing*”⁴, y en ellos los datos eran transacciones relativas a compras realizadas por clientes. Por ejemplo, los registros de las compras hechas en un supermercado *market basket analysis*. La siguiente regla podría ser un ejemplo:

cereales → leche
[soporte = 3 %, confianza = 70 %]

Esto representaría el que “el 70 % de los clientes que compran cereales compran también leche, y el 3% de todas las compras incluyen cereales y leche”.⁵

Este tipo de extracción de reglas sobre datos relativos a transacciones que agrupan ítems, pueden denominarse reglas de asociación clásicas. De forma más general, los conjuntos de datos sobre los que extraer reglas pueden ser tablas relacionales, consistentes en tuplas formadas por un número fijo de atributos.

⁴ Historia. *Universidad de Huelva UHU*. http://www.uhu.es/ude/MISION-VISION_UHU_2009.pdf://portal.ingenieria.usac.edu.gt/index.php/trabajador/antecedentes. Consulta: 20 de mayo de 2014.

⁵ *Ibíd.*

Para la mayor parte de los conjuntos de datos pueden generarse un número muy elevado de reglas. Por ello es imprescindible definir algunas medidas que permitan filtrar las más significativas. Las medidas básicas ya descritas en los trabajos iniciales sobre el tema son el soporte *support* y la confianza *confidence*:

- Soporte: mide la frecuencia con que ocurre una regla, evaluando el ratio de tuplas que cumplen el antecedente y el consecuente sobre el total de tuplas (probabilidad de que una tupla cumpla con el antecedente y el consecuente).
- Confianza: mide la exactitud con que se cumple una regla, valuando el ratio de tuplas que cumplen el antecedente y el consecuente sobre las que cumplen el antecedente. De esta manera, una regla de asociación es una relación de la forma $A \Rightarrow C$, donde A (antecedente) y C (consecuente) son condiciones sobre atributos del conjunto de datos.

Generalmente, el antecedente consta de una o más condiciones sobre atributos. Mientras que el consecuente está formado por una condición sobre un atributo. Un ejemplo de una regla de este tipo podría ser:

$edad \in [35, 50]$, casado = 'SI' \Rightarrow numCoches = 2
[soporte = 10 %, confianza = 80 %]

Representaría que “el 80 % de las personas casadas de entre 35 y 50 años tienen 2 coches, y un 10 % de las personas tienen entre 35 y 50 años, están casadas y tienen 2 coches.

1.2.3. Reglas de asociación numéricas

Es habitual que los datos contengan atributos que puedan tomar valores numéricos (enteros o reales), con un rango de valores amplio. "Las condiciones sobre dichos atributos toman entonces la forma de intervalo numérico, y las reglas así generadas se denominan reglas de asociación numéricas, habiendo sido inicialmente presentadas".⁶

La estrategia usada, por diversos investigadores a la hora de tratar el problema de generar reglas de asociación numéricas, ha sido la de dividir el rango de cada atributo numérico en intervalos. Por ello, tratar entonces dichos intervalos como valores discretos.

Se han empleado diversos métodos para particionar el rango de los atributos numéricos. "Estos son dividir en intervalos del mismo tamaño, en intervalos de la misma frecuencia, técnicas de clustering, agrupar intervalos consecutivos, empleo de conjuntos difusos, y otros; pero todos ellos necesitan información a priori del usuario. Algunas de estas técnicas se han presentado".⁷

"También otros investigadores han enfocado el problema mediante el empleo de algoritmos genéticos en la búsqueda de los intervalos de los atributos".⁸

⁶ Historia. *Universidad de Huelva UHU*. http://www.uhu.es/ude/MISION-VISION_UHU_2009.pdf://portal.ingenieria.usac.edu.gt/index.php/trabajador/antecedentes. Consulta: 20 de mayo de 2014.

⁷ *Ibíd.*

⁸ *Ibíd.*

1.2.4. Medidas de calidad de las reglas de asociación

El número de reglas de asociación que puede derivarse de un conjunto de datos puede ser muy elevado. Incluso con los requisitos de que cumplan con unos umbrales de soporte mínimo (*minsup*) y confianza mínima (*minconf*).

Para reducir el conjunto de reglas presentadas se hace necesario el empleo de algunas medidas de calidad que permitan filtrarlas y ordenarlas, según unos criterios de interés para el usuario. Las medidas de calidad para reglas de asociación pueden clasificarse en objetivas o subjetivas.

Una medida objetiva se deriva solamente de los propios datos, estando la mayoría basadas en la estadística o en la teoría de la información. Estas son representan una valoración de la correlación o distribución de los datos.

Una medida subjetiva tiene en cuenta tanto los datos como el usuario que examina los datos. Se requiere para su definición del conocimiento del usuario sobre los datos, representado generalmente como creencias y expectativas. "Los métodos que usan medidas subjetivas son generalmente interactivos, siendo una pieza clave la representación del conocimiento del usuario, que ha sido objeto de varios trabajos en el ámbito de la minería de datos".⁹

"Se centrará en las medidas de calidad objetivas, que han sido objeto de estudio por muchos investigadores".¹⁰ Entre ellas cabe destacar las medidas *lift* (también llamado *interest*) y *leverage*:

⁹ Historia. *Universidad de Huelva UHU*. http://www.uhu.es/ude/MISION-VISION_UHU_2009.pdf://portal.ingenieria.usac.edu.gt/index.php/trabajador/antecedentes. Consulta: 20 de mayo de 2014.

¹⁰ *Ibíd.*

- “*Lift*: introducida en”¹¹, mide el grado de dependencia del antecedente y el consecuente, evaluando cuánto más frecuentemente ocurren conjuntamente el antecedente y el consecuente de lo esperado si fueran estadísticamente independientes.
- *Leverage*: introducida en database, mide la diferencia entre el cumplimiento simultáneo del antecedente y el consecuente con respecto a lo previsto si fueran independientes.

1.2.5. Visualización de reglas

Otros autores han desarrollado técnicas de visualización de reglas de asociación, mayormente para el caso de atributos discretos. “Algunas de estas técnicas son las siguientes”:¹²

- Técnicas basadas en tablas. Por ejemplo: SAS Enterprise Miner, DBMine.
- Técnicas basadas en matrices 2D. Por ejemplo: SGI MineSet, DBMiner.
- Técnicas basadas en grafos. Por ejemplo: DBMiner ball graphs.
- Técnicas basadas en coordenadas paralelas. Por ejemplo: VisAR, Tminer.

¹¹ Historia. *Universidad de Huelva UHU*. http://www.uhu.es/ude/MISION-VISION_UHU_2009.pdf ://portal.ingenieria.usac.edu.gt/index.php/trabajador/antecedentes. Consulta: 20 de mayo de 2014.

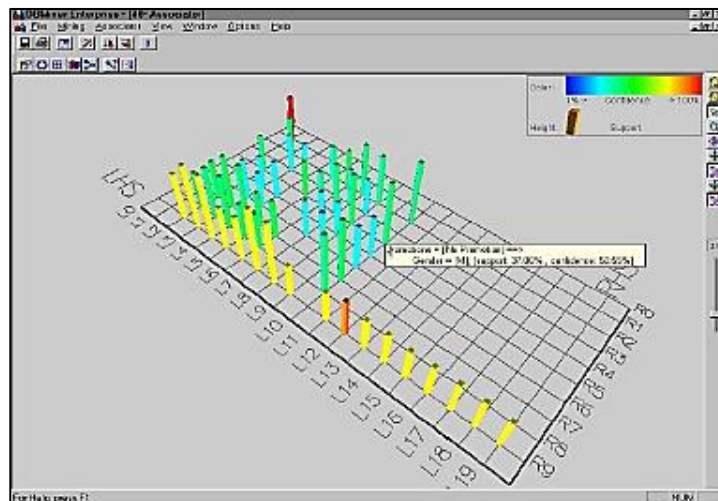
¹² *Ibíd.*

Figura 2. Representación tabular

	Body	Implic	Head	Supp (%)	Conf (%)	L	C	M	
1	cont() = 0.00-100.00'	==>	region() = 0.00-900.00'	20.45	41.4				
2	cont() = 0.00-100.00'	==>	region() = 500.00-1000.00'	20.46	24.05				
3	cont() = 0.00-100.00'	==>	conturb() = 0.00-100.00'	58.17	81.04				
4	cont() = 0.00-100.00'	==>	region() = 1000.00-1500.00'	10.45	14.04				
5	cont() = 0.00-100.00'	==>	region() = United States	22.50	32.04				
6	conturb() = 1000.00-2000.00'	==>	conturb() = 0.00-100.00'	12.91	66.54				
7	conturb() = 0.00-100.00'	==>	region() = 0.00-900.00'	26.45	34.54				
8	conturb() = 0.00-100.00'	==>	conturb() = 1000.00-2000.00'	12.91	16.07				
9	conturb() = 0.00-100.00'	==>	region() = United States	25.9	31.05				
10	conturb() = 0.00-100.00'	==>	conturb() = 0.00-100.00'	58.17	71.06				
11	conturb() = 0.00-100.00'	==>	conturb() = 500.00-1000.00'	12.92	16.42				
12	conturb() = 0.00-100.00'	==>	region() = 1000.00-1500.00'	19.57	22.08				
13	conturb() = 0.00-100.00'	==>	region() = United States	12.92	58.72				
14	conturb() = United States	==>	conturb() = 0.00-100.00'	25.9	81.04				
15	conturb() = United States	==>	conturb() = 0.00-100.00'	22.50	71.06				
16	conturb() = 0.00-100.00'	==>	conturb() = 0.00-100.00'	20.45	100				
17	conturb() = 0.00-100.00'	==>	conturb() = 0.00-100.00'	20.45	100				
18	conturb() = 1000.00-1500.00'	==>	conturb() = 0.00-100.00'	10.45	96.75				
19	conturb() = 500.00-1000.00'	==>	conturb() = 0.00-100.00'	20.46	100				
20	conturb() = 500.00-1000.00'	==>	conturb() = 0.00-100.00'	19.57	96.14				
21									
22									
23	cont() = 0.00-1000.00'	==>	region() = 0.00-900.00' AND conturb() = 0.00-100.00'	20.45	41.4				
24	cont() = 0.00-1000.00'	==>	region() = 500-900.00' AND conturb() = 0.00-100.00'	20.45	41.4				
25	cont() = 0.00-1000.00'	==>	region() = 500.00-1000.00' AND conturb() = 0.00-100.00'	19.57	27.93				
26	cont() = 0.00-1000.00'	==>	region() = 1000.00-1500.00' AND conturb() = 0.00-100.00'	19.57	29.04				
27	cont() = 0.00-1000.00' AND conturb() = 0.00-100.00'	==>	region() = 0.00-1000.00'	19.57	33.23				

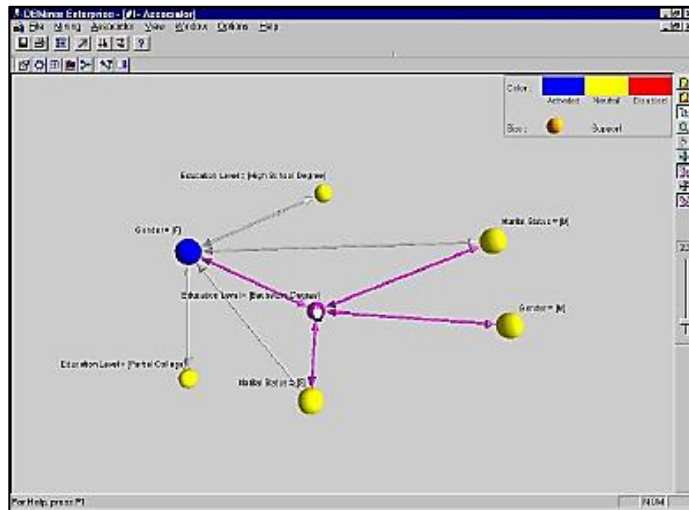
Fuente: Association rules viewer. <http://www2.lifl.fr/~jourdan/download/arv.html>. Consulta: 24 de mayo de 2014.

Figura 3. Representación 2D



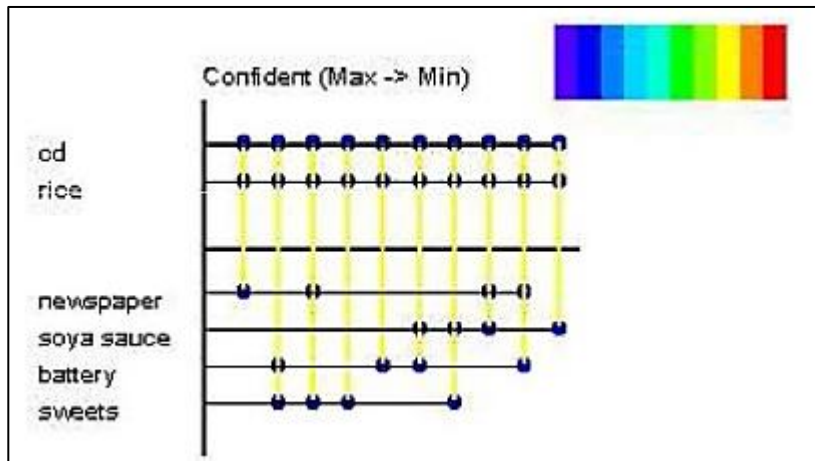
Fuente: Association rules viewer. <http://www2.lifl.fr/~jourdan/download/arv.html>. Consulta: 24 de mayo de 2014.

Figura 6. Visualización en grafos



Fuente: Association rules viewer. <http://www2.lifl.fr/~jourdan/download/arv.html>. Consulta: 24 de mayo de 2014.

Figura 7. Visualización en coordenadas paralelas



Fuente: Association rules viewer. <http://www2.lifl.fr/~jourdan/download/arv.html>. Consulta: 24 de mayo de 2014.

2. FASE TÉCNICO PROFESIONAL

2.1. Descripción general del proyecto

Hasta ahora se ha abordado el proyecto desde el punto de vista teórico y las diferentes áreas en las que se enfocan estas técnicas. Las ventajas que puede traer a un negocio y sus diferentes usos.

A partir de aquí se explicará desde un punto de vista ingenieril. Dado que el objetivo del proyecto es una interfaz gráfica en una aplicación y mostrar que se han alcanzado las capacidades necesarias para llegar a la solución del problema planteado. Para ello hay que definir una serie de parámetros entre estos requisitos (de información, funcionales y no funcionales). La interfaz debe cumplir además de los objetivos y la arquitectura de la aplicación.

Se mostrará en lo sucesivo ejemplos de la aplicación, tras las generación previa de reglas a partir del DataSet Iris. "Este muy empleado a modo de prueba y que puede obtenerse en el repositorio UCI".¹³

2.2. Estructura

La aplicación se divide en 3 fases principales.

¹³ Historia. *Universidad de Huelva UHU*. http://www.uhu.es/ude/MISION-VISION_UHU_2009.pdf ://portal.ingenieria.usac.edu.gt/index.php/trabajador/antecedentes. Consulta: 20 de mayo de 2014.

2.2.1. Importar datos y reglas

El proceso de importar consiste en cargar la información para el manejo de la aplicación. La información para esta interfaz está dividida en dos ficheros, datos y reglas respectivamente.

La estructura para cada fichero debe ser estricta y dispensable para la ejecución exitosa del programa. A continuación, las dos estructuras de cada fichero.

2.2.1.1. DataSet

El primer fichero contiene el tipo de la variable y su respectivo valor para cada dato formando un conjunto de tuplas.

Ejemplo de estructura.

2.2.1.1.1. Primera línea

Son 2 números enteros separados por un espacio. El primero posee el número tuplas de datos en el fichero y el segundo es el número de variables

150 5

2.2.1.1.2. Segundo bloque

En este bloque están las diferentes variables, cada línea es una variable con su nombre y tipo respectivamente. Si el tipo es N significa numérico y C

significa categórico que posteriormente trae el número y valores posibles para esta.

- sepal_length N
- sepal_width N
- petal_length N
- petal_width N
- class C 3 Iris-setosa,Iris-versicolor,Iris-virginica

2.2.1.1.3. Tercer bloque

En este último bloque vienen los datos separados por coma dependiendo como fueron definidos antes: numéricos o categóricos.

- 5.1,3.5,1.4,0.2,Iris-setosa
- 4.9,3.0,1.4,0.2,Iris-setosa
- 4.7,3.2,1.3,0.2,Iris-setosa

2.2.1.2. Rules

Este fichero contiene las reglas que definen el comportamiento de los datos cargados en el fichero anterior.

Ejemplo estructura.

2.2.1.2.1. Primera línea

Número de reglas en el fichero.

335

2.2.1.2.2. Segunda línea

- Resto fichero
- Cada línea del fichero representa una regla

```
ANNAC 4.3 6.1 0.1 0.4 Iris-setosa Iris-setosa 47 0.979167 2.99745 0.2088 48 49
NNNAC 0.1 0.4 Iris-setosa Iris-setosa 47 0.979167 2.99745 0.2088 48 49
ANANC 4.3 6.1 1 1.7 Iris-setosa Iris-setosa 46 0.978723 2.99609 0.204311 47 49
NACNA 3 4.4 1 1.9 Iris-setosa Iris-setosa 46 0.978723 2.99609 0.204311 47 49
```

- Cada línea se interpreta de la siguiente forma
 - El primer bloque: ANNAC

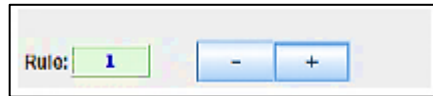
Se interpreta carácter por carácter donde el número de caracteres es igual al número de variables definidas en el primer fichero. Si es A significa antecedente, C consecuente y N no es parte de la regla.

La otra parte son rangos. Esto para cada condición de la regla en orden de derecha a izquierda, según definidos anteriormente y si el valor puede ser categórico o numérico.

2.2.2. Presentar las reglas de forma interactiva y gráfica

Luego que los datos fueron cargados con la selección interactiva del usuario. Se mostraría información de las reglas elegidas, así como los cambios en las medidas de calidad de las reglas al variar sus variables intervalos.

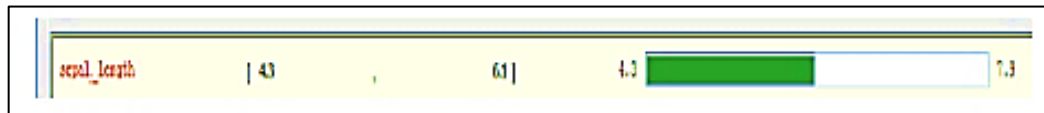
Figura 8. Manejo reglas



Fuente: elaboración propia, empleando VisualQAR.

El usuario podrá modificar, eliminar cada condición de una regla específica de forma interactiva.

Figura 9. Modificación condiciones

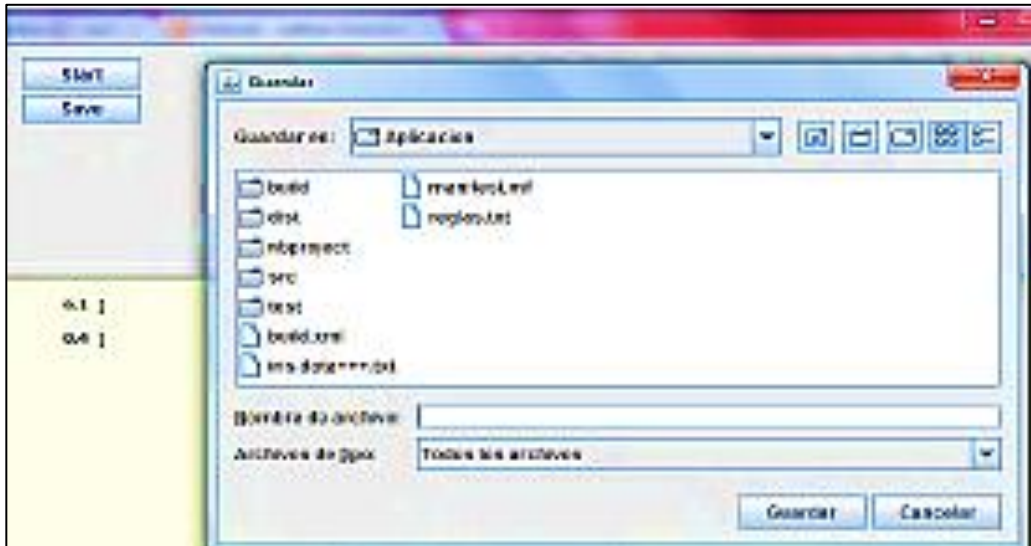


Fuente: elaboración propia, empleando VisualQAR.

2.2.3. Exportar nuevas reglas

Cuando se haya terminado la modificación de condiciones y reglas se procede a exportar las nuevas reglas. La aplicación lo hace por un fichero con la misma estructura al de importación de reglas.

Figura 10. **Exportar nuevas reglas**



Fuente: elaboración propia, empleando VisualQAR.

2.3. **Modelo de requisitos**

A continuación se presenta el modelo de requisitos.

2.3.1. **Requisitos de información**

En la siguiente tabla se muestra los requisitos de información.

Tabla I. **Requisitos de información**

IRQ-01	Interfaz manejo reglas asociación
Objetivos asociados	<ul style="list-style-type: none"> • OBJ-01 Cargar archivos • OBJ-02 Presentar reglas asociadas • OBJ-03 Gestionar cambios para reglas • OBJ-04 Lista reglas favoritas • OBJ-05 Exportar reglas favoritas
Requisitos asociados	<ul style="list-style-type: none"> • FRQ-01 Importación datos • FRQ-02 Ver reglas asociación • FRQ-03 Ver reglas favoritas • FRQ-04 Especificar ruta fichero.
Descripción	La aplicación deberá gestionar las reglas de asociación en memoria y presentarlas al usuario de forma interactiva para ser modificadas, para luego exportar las nuevas reglas.
Datos específicos	<ul style="list-style-type: none"> • Nombres y rutas absolutas de ficheros. • Valores de las variables categóricas y numéricas para cada regla.

Fuente: elaboración propia.

2.3.2. **Requisitos funcionales**

A continuación se presentan los requisitos funcionales.

2.3.2.1. **Objetivos**

En la siguiente tabla se presenta el objetivo del sistema: carga archivos.

Tabla II. **Objetivo del sistema: cargar archivos a**

OBJ-01	Cargar archivos
Descripción	La aplicación al inicio deberá permitir adjuntar 2 ficheros que poseen la información de las reglas y datos.
Estabilidad	Alta.
Comentarios	La Estructura es única y no puede ser modificada.

Fuente: elaboración propia.

Tabla III. **Objetivo del sistema: cargar archivos b**

OBJ-02	Presentar reglas asociadas
Descripcion	Luego de analizar y leer los ficheros la aplicación tiene que presentar de forma gráfica e interactiva para el usuario las respectivas reglas con sus variables de una forma sencilla de comprender.
Estabilidad	Alta.
Comentarios	Ninguno.

Fuente: elaboración propia.

Tabla IV. **Objetivo del sistema: gestionar cambios en reglas**

OBJ-03	Gestionar cambios en reglas.
Descripcio	La interfaz deberá ser útil para modificar los parámetros de las variables una regla seleccionada y capaz de manejar en memoria los cambios para luego en un futuro exportar las nuevas reglas.
Estabilidad	Alta.
Comentarios	Los cambios pueden ser eliminar condiciones o modificar intervalos para las variables numéricas.

Fuente: elaboración propia.

Tabla V. **Objetivo del sistema: listar reglas favoritas**

OBJ-04	Lista reglas favoritas.
Descripción	La aplicación tiene la opción para agregar reglas y categorizarlas como favoritas, este uso es especialmente para las reglas que fueron modificadas o seleccionadas por el usuario.
Estabilidad	Alta.
Comentarios	Estas reglas pueden ser modificadas después que han sido agregadas.

Fuente: elaboración propia.

Tabla VI. **Objetivo del sistema: exportar reglas favoritas**

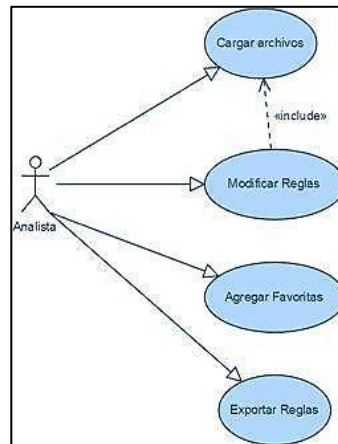
OBJ-05	Exportar reglas favoritas
Descripción	Al finalizar de hacer los cambios la aplicación pedirá información de la ruta para guardar el archivo con las nuevas reglas, utilizando el mismo formato de entrada.
Estabilidad	Alta.
Comentarios	Ninguno.

Fuente: elaboración propia.

2.3.2.2. Diagrama de casos de uso

A continuación se presenta el diagrama de casos de uso.

Figura 11. Diagrama de casos de uso a



Fuente: elaboración propia, empleando Microsoft Visio 2010.

2.3.2.3. Casos de uso

En la siguiente tabla se presenta el caso de uso: cargar archivos.

Tabla VII. Caso de uso: cargar archivos

UC-01	Cargar archivos	
Objetivos asociados	OBJ-01	
Requisitos asociados		
Descripción	Describe como se cargarán los 2 ficheros iniciales a la aplicación.	
Precondición	Ninguna	
Secuencia normal	Paso	Acción
	1	El sistema despliega 2 botones para seleccionar los ficheros.
	2	El usuario selecciona los ficheros.
	3	El sistema despliega el nombre de los ficheros cargados.
	4	El usuario da <i>click</i> en el botón iniciar y los archivos son ejecutados.
Postcondición	Se han cargado los ficheros.	
Excepciones	2	Si la ruta introducida es dañada o el fichero no se carga con éxito, se cancelará la ejecución de la interfaz.

Fuente: elaboración propia.

Tabla VIII. **Caso de uso: modificar reglas**

UC-02	Modificar reglas	
Objetivos asociados	OBJ-02, OBJ-03	
Requisitos asociados		
Descripción	Describe las modificaciones que se pueden realizar a las reglas asociadas desplegadas por la aplicación.	
Precondición	Los archivos deben estar cargados previamente.	
Secuencia normal	Paso	Acción
	1	El sistema despliega de forma interactiva las reglas cargadas anteriormente.
	2	El usuario selecciona la regla a modificar y puede eliminar condiciones con doble <i>click</i> o en el caso de variables numéricas dando <i>click</i> puede modificar el rango.
	3	La aplicación luego de cada modificación irá desplegando los nuevos datos de las medidas de calidad.
Postcondición	Ninguna	
Excepciones	2	Si el usuario no desea guardar la modificación puede regresar a la regla original a través del botón <i>back</i> .

Fuente: elaboración propia.

Tabla IX. **Caso de uso: agregar favoritas**

UC-03	Agregar Favoritas	
Objetivos asociados	OBJ-04	
Requisitos asociados		
Descripción	Describe el procedimiento para agregar reglas a la lista de favoritas.	
Precondición	Reglas modificadas o reglas originalmente cargadas.	
Secuencia normal	Paso	Acción
	1	El usuario selecciona una regla.
	2	El usuario hace las modificaciones convenientes en las condiciones de la regla seleccionada o en variables numéricas.
	3	El usuario dará <i>click</i> en el botón " <i>Add</i> " para agregar la regla actual a la lista de favoritas.
Postcondición	Ninguna.	
Excepciones	2	Si el usuario no desea guardar la modificación puede regresar a la regla original a través del botón <i>back</i> .

Fuente: elaboración propia.

Tabla X. **Caso de uso: exportar reglas**

UC-04	Exportar reglas.	
Objetivos asociados	OBJ-05	
Requisitos asociados		
Descripción	Describe el procedimiento para exportar las reglas de la lista de favoritas.	
Precondición	Reglas agregadas anteriormente a la lista favoritas.	
Secuencia normal	Paso	Acción
	1	El usuario selecciona el botón <i>stop</i> y la aplicación pide la ruta y nombre del fichero a guarda.
	2	El usuario selecciona la ruta y nombre a conveniencia.
	3	La aplicación crea un nuevo archivo con las reglas favoritas en la ruta deseada y el nombre especificado.
Postcondición	Ninguna	
Excepciones	2	Si la ruta introducida por el administrador no es correcta el sistema cancela la exportación del archivo que contiene las nuevas reglas.

Fuente: elaboración propia.

2.3.2.4. Actores

En la siguiente tabla se presenta los actores analistas.

Tabla XI. **Actor analista**

ACT-01	Analista
Descripción	Este actor representa al analista de los datos y manejador único de la aplicación.
Comentarios	Ninguno

Fuente: elaboración propia.

2.3.3. Requisitos no funcionales

En la siguiente tabla se presentan los requisitos no funcionales: utilización colores adecuados.

Tabla XII. **Requisito no funcional: utilización colores adecuados**

NFR-01	Utilización colores adecuados
Objetivos asociados	OBJ-02
Requisitos asociados	-
Descripción	El sistema utilizará en cada componente y enfatizará en las áreas de mayor interés colores llamativos y adecuados para mayor visualización y manejo para el usuario.
Comentarios	Ninguno.

Fuente: elaboración propia.

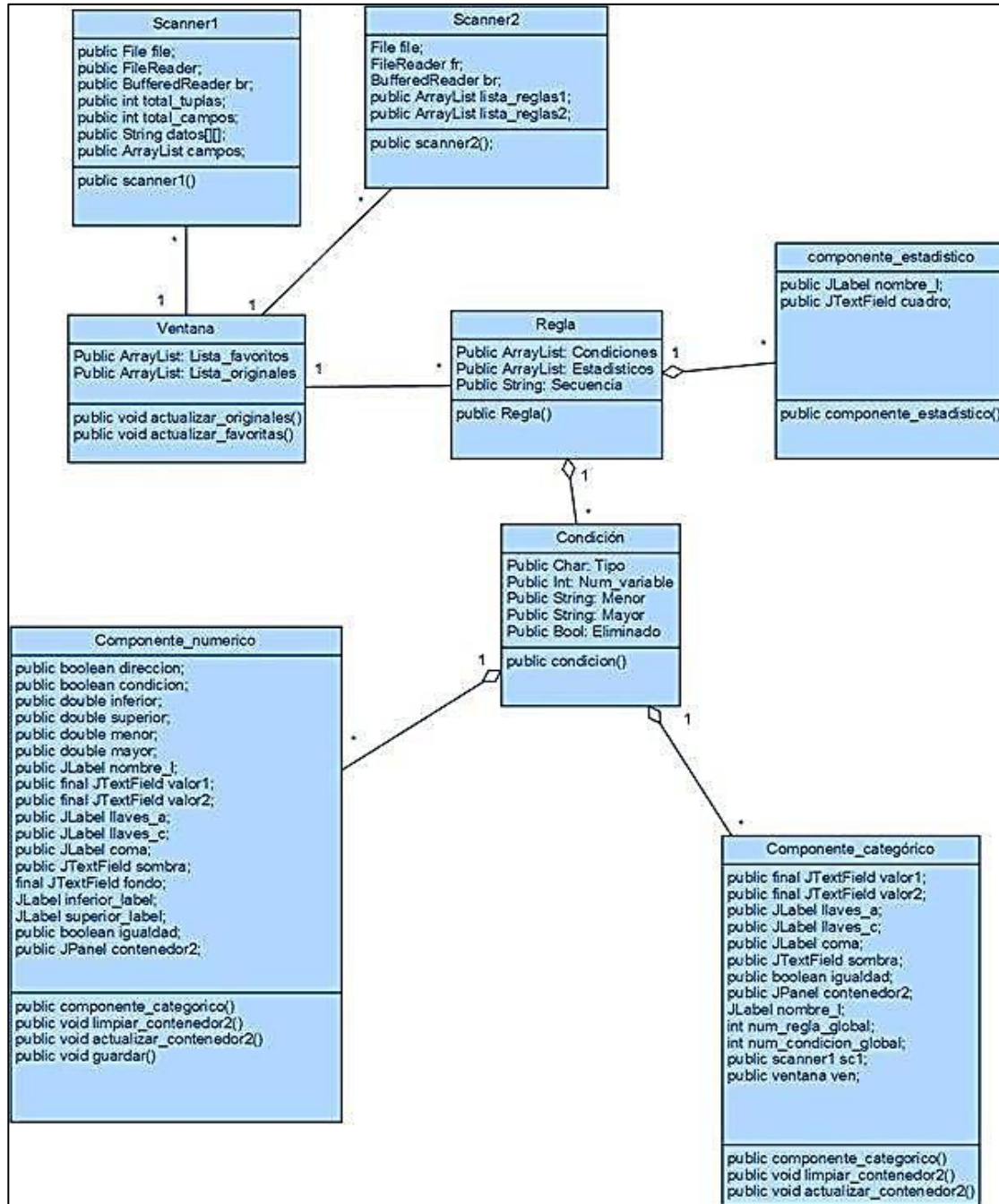
2.3.4. Modelo del diseño

A continuación se presenta el modelo del diseño.

2.3.4.1. Diagrama de clases

En el siguiente diagrama se presentan los casos de uso.

Figura 12. Diagrama de casos de uso b



Fuente: elaboración propia, empleando Microsoft Visio 2010.

2.3.4.2. Clases

A continuación se explicarán las clases modelos de diseño.

- Clase: ventana atributos
Public ArrayList: lista_favoritos: lista para almacenar reglas favoritas
Public ArrayList: lista_originales: lista para almacenar reglas originales
- Métodos
public void actualizar_originales(): método que genera y actualiza reglas originales.
public void actualizar_favoritas(): método que genera y actualiza reglas favoritas.
- Clase: scanner1
Atributos
public File file; variable para el fichero
public FileReader; variable sirve para leer.
public BufferedReader br; *buffer* para leer el fichero
public int total_tuplas; cantidad de tuplas
public int total_campos; cantidad de campos
public String datos[][]; matriz de datos
- Metodos:
public scanner1(): método que ejecuta el *scanner* 1 y lee el fichero número 1.
- Clase: scanner2
Atributos

File file; variable del fichero
FileReader fr; variable para leer
BufferedReader br; buffer para leer el fichero
public ArrayList lista_reglas1; lista de reglas leídas

- Métodos:
public scanner2(): método que ejecuta el *scanner* 2 y lee el fichero número 2.
- Clase: regla
Atributos
Public ArrayList: condiciones: lista que contiene las condiciones
Public ArrayList: estadísticos: lista que contiene los estadísticos.
Public String: secuencia: secuencia de caracteres con las condiciones originales.
- Métodos:
public regla(): crea un nuevo objeto tipo regla
- Clase: condición
Atributos
Public Char: tipo : carácter que define el tipo de condicion puede ser A o C. *Public* Int: Num_variable: número que define el numero de condición.
Public String: menor. Número inferior que puede tener la condición.
Public String: mayor: Número mayor que puede tener la condición.
Public Bool: eliminado: si la condición fue eliminada es definida como *true*.
- Métodos
Public condición(): crea un nuevo objeto tipo condición.

- Clase: componente_numérica

Atributos

public double inferior; Número mínimo que puede tener el componente.

public double superior; Número máximo que puede tener el componente.

public double menor; Valor de la restricción menor.

public double mayor; Valor de la restricción mayor.

public JLabel nombre_; Nombre del componente.

public final JTextField valor1; despliega el valor inferior.

public final JTextField valor2; despliega el valor superior

public JLabel llaves_a;

public JLabel llaves_c;

public JLabel coma;

public JTextField sombra;

final JTextField fondo;

JLabel inferior_label; despliega valor mínimo.

JLabel superior_label; despliega valor máximo.

public boolean igualdad;

public JPanel contenedor2;

- Métodos

public componente_numérico(): crea un nuevo objeto de tipo componente_numérico

public void limpiar_contenedor2(): limpia el panel de los estadísticos y parámetros.

public void actualizar_contenedor2(): actualiza el panel de estadísticos y parámetros.

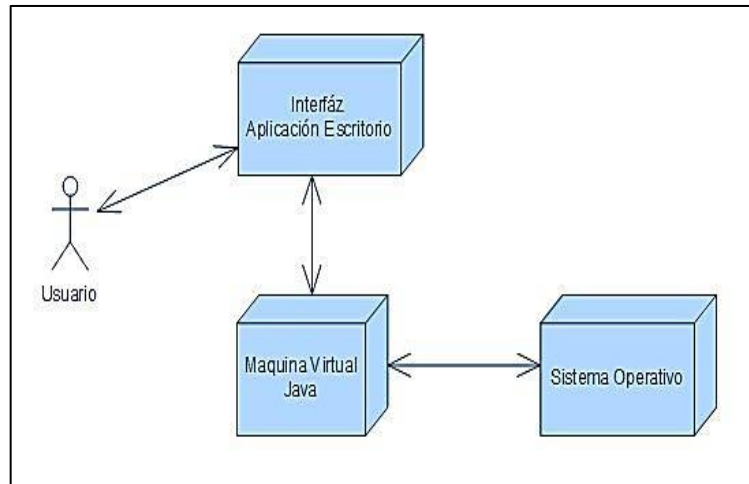
public void guardar(): guarda los nuevos valores en caso de modificación.

- Clase: componente_categorico
 Atributos
 public final JTextField valor1; despliega valor menor numérico
 public final JTextField valor2; despliega valor mayor numérico
 Variables para mostrar en pantalla etiquetado
 public JLabel llaves_a;
 public JLabel llaves_c;
 public JLabel coma;
 public JTextField sombra;
 public boolean igualdad;
 public JPanel contenedor2;
 JLabel nombre_l;
- Métodos:
 public componente_categorico(): crea un nuevo objeto de tipo componente_categorico.
 public void limpiar_contenedor2(): limpia el panel de los estadísticos y parámetros.
 public void actualizar_contenedor2(): actualiza el panel de estadísticos y parámetros.
- Clase: componente_estadístico
 Atributos
 public JLabel nombre_l; despliega el nombre del estadístico o parámetro.
 public JTextField cuadro; valor numérico del estadístico o parámetro.
 Métodos:
 public componente_estadístico(): crea un nuevo objeto de tipo componente_estadístico.

2.3.5. Despliegue del sistema

El despliegue y características del sistema es sencillo. Únicamente es necesario tener instalado en el sistema operativo las librerías de Java. La interfaz estará ejecutándose sobre la máquina virtual de java.

Figura 13. Despliegue del sistema



Fuente: elaboración propia, empleando Microsoft Visio 2010.

3. FASE ENSEÑANZA APRENDIZAJE

En este proceso se entrega todo el material recopilado y la documentación generada a la Escuela Superior Técnica de la Universidad de Huelva, España. Incluye el código fuente, documentación necesaria y una capacitación de 1 hora aproximadamente.

Durante esta fase también se crea material de apoyo como manuales de usuario. Para que puedan consultarlos en cualquier momento que tengan alguna duda.

Esta fase sirve para que los usuarios se familiaricen con el sistema y así puedan aceptar el cambio con la menor resistencia.

La etapa de pruebas, por parte del usuario, es como un ejercicio para que al interactuar con el sistema se dé cuenta de la forma en que se trabaja. Para así familiarizarse con la herramienta sin que este alguien apoyando en la parte técnica.

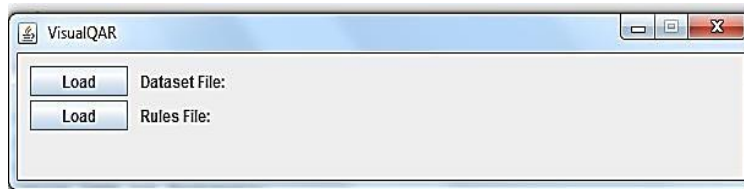
3.1. Manual de usuari006F

La aplicación llamada VisualQAR() es una interfaz para el manejo y control de reglas de asociación.

3.1.1. Inicio y carga de ficheros

Al inicio presenta una pantalla con 2 botones.

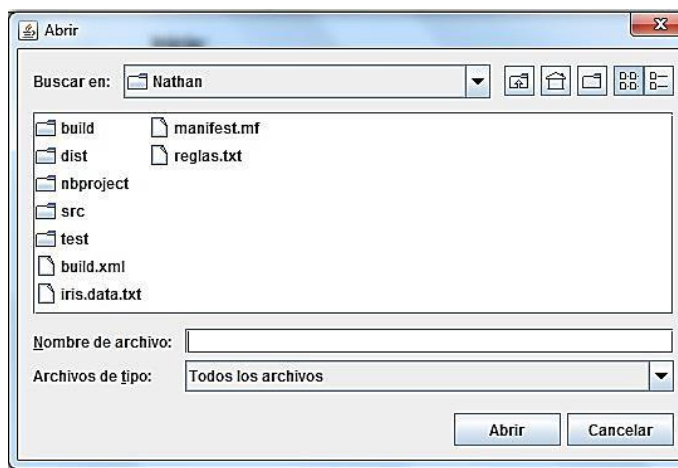
Figura 14. **Inicio**



Fuente: elaboración propia, empleando VisualQAR.

Al pulsar un botón aparece un cuadro para seleccionar el fichero.

Figura 15. **Cargar ficheros**



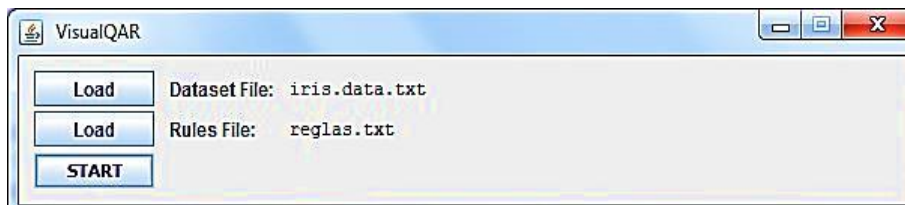
Fuente: elaboración propia, empleando VisualQAR.

El orden de los ficheros es:

- DataSetFile: este contiene todos los datos
- Rules: aquí vienen definidas las reglas y sus condiciones

Luego que son cargados, muestra los nombres respectivos de los ficheros seleccionados y aparece el botón de *start*.

Figura 16. **Ejecutar**

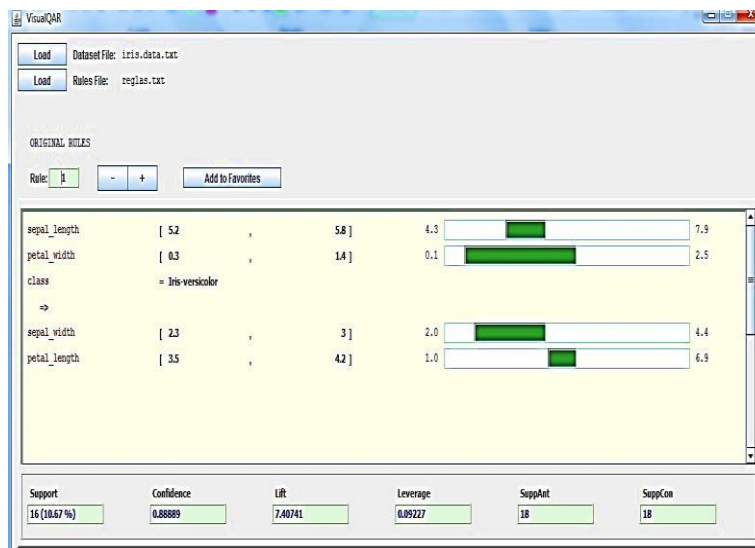


Fuente: elaboración propia, empleando VisualQAR.

3.1.2. Presentación de reglas

Luego de pulsar *start* las reglas son presentadas así.

Figura 17. **Presentación reglas**



Fuente: elaboración propia, empleando VisualQAR.

Para el manejo de reglas y movilización entre cada una, se utiliza estos botones.

Figura 18. **Gestionar reglas**

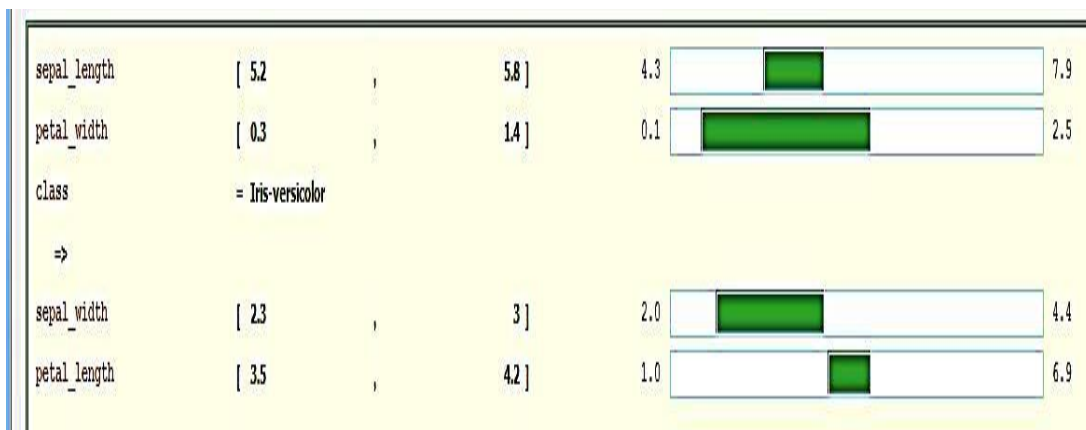


Fuente: elaboración propia, empleando VisualQAR.

Se puede avanzar o retroceder para las reglas, pero depende del tipo de regla que se visualice.

El área más grande muestra las condiciones de las reglas:

Figura 19. **Área de condiciones**

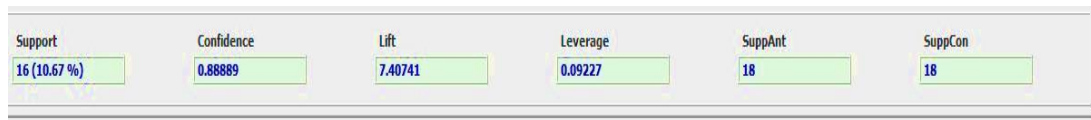


Fuente: elaboración propia, empleando VisualQAR.

Cada línea forma una condición, que puede ser de tipo numérica o categórica.

En la parte de abajo muestra como último bloque el área de las medidas de soporte.

Figura 20. **Área estadísticos**



Fuente: elaboración propia, empleando VisualQAR.

Esta área de datos se va actualizando automáticamente conforme se van modificando las condiciones de la regla mostrada.

3.1.3. Tipos de reglas visualizadas

Pueden ser de 2 tipos las originales o favoritas. Para eso existe un título que muestra en que vista se encuentra actualmente.

- Vista reglas originales

Figura 21. **Vista originales**



Fuente: elaboración propia, empleando VisualQAR.

- Vista reglas favoritas

Figura 22. **Vista favoritas**

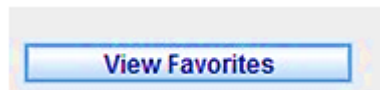


Fuente: elaboración propia, empleando VisualQAR.

Para cambiar estas vistas existe un botón que permite gestionar las dos listas de reglas.

Al inicio se mostrará originales y estará configurado para ver favoritas.

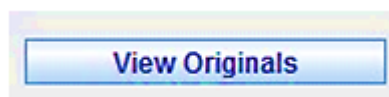
Figura 23. **Botón favoritas**



Fuente: elaboración propia, empleando VisualQAR.

Luego de hacerlo cambia de nombre.

Figura 24. **Botón originales**



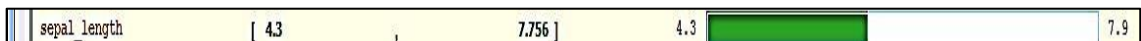
Fuente: elaboración propia, empleando VisualQAR.

3.1.4. Modificación de condiciones

Las únicas condiciones que se pueden modificar son las numéricas, para esto existen 2 formas diferentes:

- Dando *click*: sobre la barra que muestra el límite de la condición sobre el nuevo valor que se quisiera actualizar. Ejemplo:

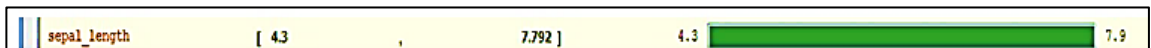
Figura 25. Condición original



Fuente: elaboración propia, empleando VisualQAR.

Al ser modificado la sombra se actualiza, en este caso *click* al final.

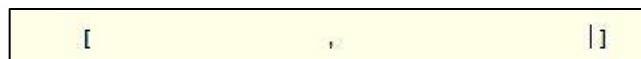
Figura 26. Modificar condición con *click*



Fuente: elaboración propia, empleando VisualQAR.

- Cambiando los valores numéricos con el teclado. Ejemplo límites del 5 al 6.

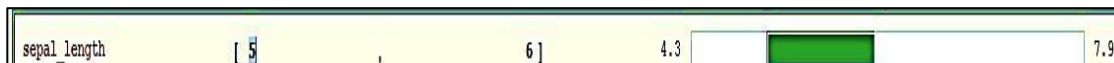
Figura 27. Modificación con teclado



Fuente: elaboración propia, empleando VisualQAR.

Luego que son ingresados con el teclado se pulsa enter.

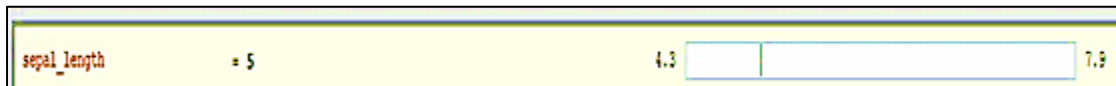
Figura 28. **Pulsando enter**



Fuente: elaboración propia, empleando VisualQAR.

Para un mismo valor se desaparece las llaves y aparece el signo igual, ejemplo:

Figura 29. **Valores iguales**



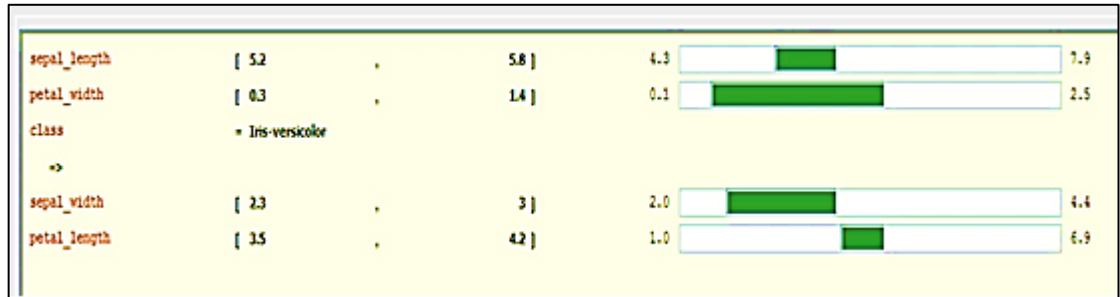
Fuente: elaboración propia, empleando VisualQAR.

3.1.5. **Eliminación de condiciones**

Se pueden eliminar condiciones sin importar que sean categóricas o numéricas. Para eliminar hay que hacer doble *click* para suprimir una condición.

Al inicio

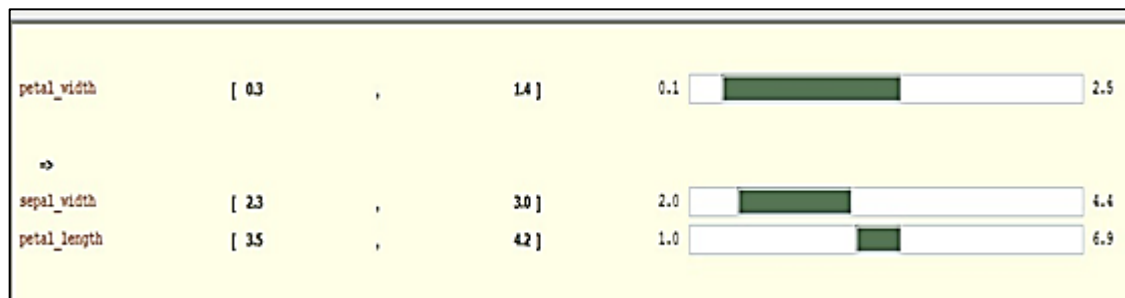
Figura 30. **Antes de eliminar**



Fuente: elaboración propia, empleando VisualQAR.

Luego de eliminar 2 condiciones, se visualiza así.

Figura 31. **Después de eliminar**



Fuente: elaboración propia, empleando VisualQAR.

Deshacer cambios.

Existe el botón *Undo Changes*, este funciona tanto para la visualización de originales como de favoritas.

Figura 32. **Botón de cambios**



Fuente: elaboración propia, empleando VisualQAR.

Si la regla ha sufrido cambios y al pulsar este botón, la regla regresa a su estado inicial.

3.1.6. **Botón favoritas**

Cuando se visualiza una regla original, existe la opción de agregarla como una nueva regla favorita. Esto para realizar esta opción hay que pulsar sobre este botón:

Figura 33. **Botón de favoritas**



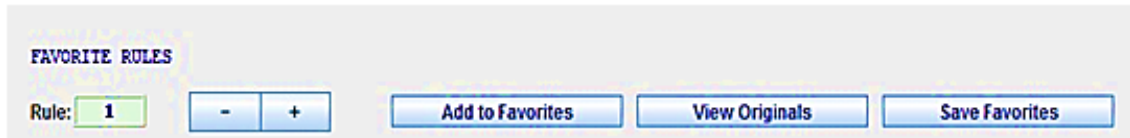
Fuente: elaboración propia, empleando VisualQAR.

Automáticamente se agrega la regla a la posición final de la lista de favoritas.

3.1.7. **Guardar favoritas**

Cuando se visualizan las reglas favoritas aparece una opción *Save Favorites*, al pulsar sobre este botón permite guardar un fichero con la misma estructura que el original que contenía las reglas.

Figura 34. **Guardar favoritas**



Fuente: elaboración propia, empleando VisualQAR.

Pide la ruta del fichero y nombre respectivamente.

Figura 35. **Exportar fichero**



Fuente: elaboración propia, empleando VisualQAR.

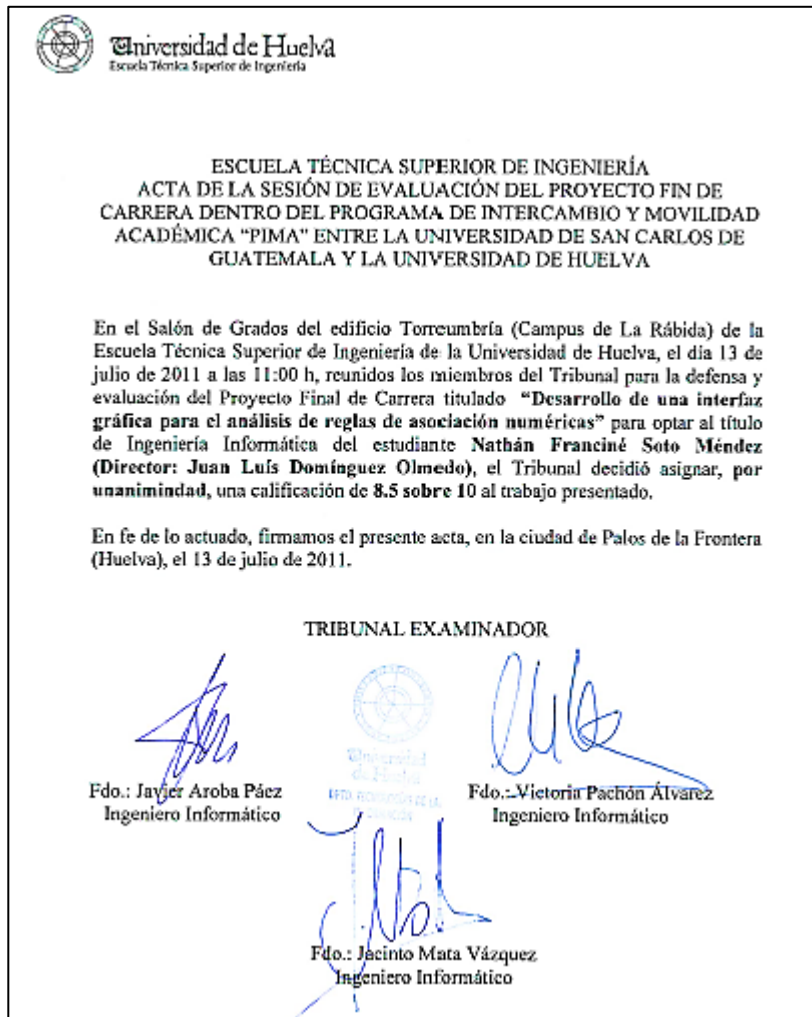
4. FASE DE PRESENTACIÓN DE LA INTERFAZ

En esta fase se hizo una presentación con una terna nombrada por la Escuela Técnica Superior de Ingeniería de la Universidad de Huelva, obteniendo una nota de aprobación por unanimidad de 8,5 sobre 10.

4.1. Acta de evaluación y presentación del proyecto

A continuación se presenta el acta de evaluación del proyecto.

Figura 36. Acta de evaluación del proyecto



Fuente: Universidad de Huelva.

CONCLUSIONES

1. El proceso de extracción de reglas de asociación numéricas da lugar generalmente a un enorme conjunto de reglas. Esto debido principalmente a la existencia de atributos numéricos en el conjunto de datos de partida.
2. Se ha pretendido con este proyecto desarrollar una interfaz visual que permita un mejor análisis de las reglas de asociación numéricas. Esto con la posibilidad de modificar interactivamente las reglas y observar la variación de sus medidas de calidad.
3. La aplicación desarrollada permite una mejor visualización de las reglas, a la vez que ayuda a mejorar la interpretabilidad de las mismas al dar la opción de modificar separadamente cada uno de sus atributos.

RECOMENDACIONES

1. La interfaz posee un flujo que comprende la carga de los ficheros, es muy importante manejar el formato establecido según el manual, porque de esto dependerá el desarrollo y continuación del flujo.
2. Luego de una carga exitosa, es importante que el usuario detecte en las reglas obtenidas el tipo de variable si es numérica o no, porque de esto depende las funcionalidades para cada regla.
3. La interfaz funciona siempre en tiempo real, mostrará información de las reglas elegidas y los cambios en las medidas de calidad de las reglas al cambiar sus variables y/o intervalos.
4. Luego de cumplir el objetivo de manipulación y manejo de las reglas de asociación de una forma interactiva para el usuario se puede considerar exportar dichas reglas y almacenarlas en un lugar seguro para uso futuro.

BIBLIOGRAFÍA

1. AGRAWAL, R. and R. Srikant. *Fast Algorithms for Mining Association Rules in Large Databases. In Proceedings of the 20th International Conference on Very Large Data Bases.* Santiago de Chile, Chile: Morgan Kaufmann, 1994. 280 p.
2. AGRAWAL, R., T. Imieliski, and SWAMI, A. *Mining association rules between sets of items in large databases.* in 1993 ACM SIGMOD International Conference on Management of Data. Washington, D.C., USA: ACM, 1993. 160 p.
3. ALATAS, B., E. Akin, and MODENAR A. Karci, *Multi-objective differential evolution algorithm for mining numeric association rules.* Applied Soft Computing, 2008. 8(1). 656 p.
4. B. LIU, W. Hsu, L. Mun, H. Lee. *Finding interesting patterns using user expectations,* *IEEE Transactions on Knowledge and Data Engineering.* USA, 1999. 832 p.
5. *Curso de Base Datos Sesión 13.* [en línea]. <www.cnys.com.mx>. [Consulta: abril de 2015].
6. G. PIATETSKY-Shapiro. *Discovery, analysis, and presentation of strong rules.* *Knowledge Discovery in Databases.* USA, 1991. 94 p.

7. GYENESEI, Mining. *Weighted Association Rules for Fuzzy Quantitative Items, PKDD 2000*. USA. 2000. 423 p.
8. HAN, J., et al., *Frequent pattern mining: current status and future directions. Data Mining and Knowledge Discovery*, 2007. 15(1). 86 p.
9. HOLT, J.D. and S.M. CHUNG. *Multipass algorithms for mining association rules in text databases*. Knowledge and Information Systems, 2001. 3(2): 183 p.
10. _____. *Efficient mining of association rules in text databases. in eighth International Conference on Information and Knowledge Management*. Kansas City, Missouri, USA: ACM. 1999. 140 p.
11. HU, T., et al. *Discovery of maximum length frequent itemsets*. Information Sciences, 2008. 178 p.
12. K. WANG, S.H. Tay, B. Liu. *Interestingness-Based Interval Merger for Numeric Association Rules, Proceedings 4th Int. Conf. KDD, USA*. 1998. 128 p.
13. KALPANA, B. and R. NADARAJAN. *Incorporating heuristics for efficient search space pruning in frequent itemset mining strategies*. Current science, 2008. 94(1). 101 p.
14. J. MATA, J.L. ÁLVAREZ, J.C. RIQUELME. *Discovering Numeric Association Rules via Evolutionary Algorithm*. PAKDD USA. 2002:40-51, 2002.

15. La Rosa, C., L. XIONG, and K. MANDELBERG. *Frequent pattern mining for kernel trace data. in 2008 ACM Symposium on Applied Computing*. Fortaleza, Ceara, Brazil: ACM, 2008. 160 p.
16. LOPEZ, F.J., et al. *Fuzzy association rules for biological data analysis: A case study on yeast*. BMC Bioinformatics, 2008. 107 p.
17. PARK, J.S., M.-S. CHEN, and P.S. YU. *Using a Hash-Based Method with Transaction Trimming for Mining Association Rules*. IEEE Transaction on Knowledge and Data Engineer. 1997. 825 p.
18. R. AGRAWAL, R. SRIKANT. *Fast Algorithms for Mining Association Rules, Proceedings of the VLDB Conference*. 1994. 489 p.
19. R. AGRAWAL, T. IMIELINSKI, A. SWAMI. *Mining association rules between sets of items in large databases*. Proceedings ACM SIGMOD, 1993. 216 p.
20. R. MILLER, Y. YANG. *Association Rules over Interval Data, Proceedings of the ACM SIGMOD*, 1997. 105 p. ISBN: 0-89791-995-5.
21. R. SRIKANT, R. Agrawal. *Mining Quantitative Association Rules in Large Relational Tables, Proceedings of the ACM SIGMOD*. 1996. 240 p.
22. S. BRIN, R. Motwani, J.D. Ullman, S. Tsur. *Dynamic itemset counting and implication rules for market basket data. Proceedings of the ACM SIGMOD Int'l Conf. on Management of Data*. USA, 1997. 110 p.

23. SAVASERE, A., E. Omiecinski, and S.B. NAVATHE. *An Efficient Algorithm for Mining Association Rules in Large Databases. in 21th International Conference on Very Large Data Bases (VLDB'95)*. Zurich, Switzerland: Morgan Kaufmann Publishers Inc, 1995. 180 p.
24. SRIKANT, R. and R. AGRAWAL. *Mining quantitative association rules in large relational tables*. SIGMOD.
25. SILBERSCHATZ, A. Tuzhilin. *On subjective measures of interestingness in knowledge discovery, First International Conference on Knowledge Discovery and Data Mining*. USA, 1995. 281 p.
26. _____. *What makes patterns interesting in knowledge discovery systems, IEEE Transactions on Knowledge and Data Engineering*. USA, 1996. 974 p.
27. Y. AUMANN, Y. Lindell. *A Statistical Theory for Quantitative Association Rules, Proceedings KDD99*, 1999. 270 p.
28. ZAKI, M.J., et al. *New Algorithms for Fast Discovery of Association Rules*. University of Rochester, 1997. 148 p.
29. ZHANG, M., et al. *Mining periodic patterns with gap requirement from sequences*. ACM Transactions on Knowledge Discovery from Data, 2007. 1(2): 146 p.