UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS
Facultad 4


# Título: Módulo para la Sumarización lingüística en R

Trabajo de Diploma para optar por el título de
Ingeniero en Ciencias Informáticas

**Autor:** Daliana Ramos García

**Tutor:** MSc. Eric Eduardo Piñera Trinchet

La Habana, junio del 2014

# Pensamiento

*"Todo en el software cambia. Los requisitos cambian. El diseño cambia. El negocio cambia. La tecnología cambia. El equipo cambia. Los miembros del equipo cambian. El problema no es el cambio en sí mismo, puesto que sabemos que el cambio va a suceder; el problema es la incapacidad de adaptarnos a dicho cambio cuando éste tiene lugar."*

*Kent Beck*

# Declaración de autoría

Declaro ser autora de la presente tesis y reconozco a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo la presente a los __ días del mes de _____del año _____.

| | |
|---|---|
| _____ | _____ |
| Firma de la Autora | Firma del Tutor |
| Daliana Ramos García | MSc. Eric Eduardo Piñera Trinchet |

# Agradecimientos

*No han sido pocos los que, de alguna manera, aportaron su granito de arena en estos años y aun pudiendo caer en el error de olvidar a alguien me gustaría expresarles mis más sinceros agradecimientos:*

*A mi mamá por ser la mejor del mundo y por haberme dado la fuerza y apoyo necesario siempre que lo necesite, te quiero mucho.*

*A mi padrastro Eduardo que fue el padre que estuvo conmigo desde casi mis primeros pasos, también te quiero mucho.*

*A mis tíos Rosa y Nelson por haber estado para mí en todo minuto de mi vida en especial de estos años que convivi con ustedes, gracias a los dos, los adoro, ustedes lo saben.*

*A mi novio Froilán por tantos buenos momentos, por tu apoyo en todo este tiempo, sabes que has sido un factor fundamental en este camino que he recorrido a tu lado, te adoro con la vida.*

*A mi tía Claribel que aunque no estuvo presente hoy sé que me desea lo mejor y a mis primos o casi hermanos Yaima, Yarenis, Yunier por ser los mejores del mundo y darme fuerzas y esperanzas siempre, a mis primitos Erika y Alejandro que me dieron la alegría necesaria en momentos difíciles.*

*A mi abuelo Pedro que es el mejor del mundo y siempre me ha deseado lo mejor.*

*A mis hermanos Arianna y Riqui por ser los mejores hermanos del mundo y a mi papá Ricardo, los quiero mucho.*

*A Leiny por ser un hermano para mí en los 4 años que nos conocemos y que seguirán por siempre, a Yuli por ser la mejor amiga que tuve en estos años.*

*A mis compañeros de aula, al igual que los compañeros que ya no compartimos la misma aula pero que en algún momento la compartimos y nos ayudamos mutuamente.*

# Agradecimientos

*A mis compañeros de convivencia por aguantarme todos estos años y ayudarme cuando fue necesario.*

*A mi tutor por ser tan comprensible y ayudarme cuando lo necesite, gracias.*

*A los profes que compartieron conmigo su tiempo y conocimientos.*

*A todos aquellos que de alguna forma influyeron en que hoy esté logrando este sueño, muchas gracias.*

*Daliana*

**Resumen**

Actualmente en las empresas u organizaciones se trabaja diariamente con grandes volúmenes de datos, los cuales son difíciles de administrar para el ser humano en el momento de tomar decisiones. En la presente investigación desarrollada en Universidad de las Ciencias Informáticas se propone un módulo para la sumarización lingüística en R. Este módulo se basa en extraer resúmenes lingüísticos en un leguaje natural fácilmente comprensible para el ser humano en forma de reglas de asociación de un conjunto de datos para usarlos en el proceso de toma de decisiones. El módulo cuenta con cuatro funcionalidades principales las cuales permiten la limpieza y discretización de los datos, la extracción y filtrado de las reglas de asociación para finalmente generar los resúmenes lingüísticos. En el módulo se propone dar un tratamiento difuso a los datos para manejar de forma natural el conocimiento y así obtener resúmenes que muestren el comportamiento de los datos de forma clara y concisa para el ser humano. Para validar el módulo y probar que los resúmenes tuvieran una buena calidad se respetaron las pruebas propuestas por la metodología escogida, además de calcular los indicadores para medir la calidad de los resúmenes.

Palabras claves: Sumarización lingüística, reglas de asociación, toma de decisiones.

**Abstract**

Currently in companies or organizations working daily with large volumes of data, which are difficult to administer to humans when making decisions. In this research developed at the University of Information Sciences proposed the creation of a module for the linguistic summarization in R. This module is based on extracting linguistic summaries in a natural language easily comprehensible to humans in the form of association rules from a dataset for use in the decision making process. The module has four main features which allow cleaning and discretization of data extraction and filtering of association rules and finally generate linguistic summaries. In the proposed module to a fuzzy data processing to handle naturally and get knowledge summaries that show the behavior of the data in a clear and concise to humans. To validate and test the module that summaries had a good quality of evidence proposed by the chosen methodology, in addition to calculating the indicators to measure the quality of the summaries.

**Índice**

# Índice

## Introducción

El desarrollo de todas las ramas de la sociedad moderna se ha acelerado considerablemente con la llegada de la informática. Este avance trae consigo varios sistemas que además de gestionar continuamente información, almacenan una serie de registros los cuales contienen parte de la historia de una empresa u organización, por lo que sería de gran importancia procesarlos.

Los seres humanos usan su cerebro para procesar los datos y extraer de ellos los mejores resultados pero debido al cúmulo de información, que gracias a las nuevas tecnologías existe actualmente, ha surgido una tarea difícil, pues los humanos no están preparados para procesar grandes volúmenes de información para luego encontrar datos de interés.

Gracias a estas tecnologías se cuenta con varias herramientas y técnicas que ayudan a procesar la información, que en muchas ocasiones está guardada en bases de datos de entidades, siendo la minería de datos una de las técnicas más utilizadas para extraer y procesar la información.

La extracción de conocimiento útil, implícito y previamente desconocido, a partir de grandes volúmenes de datos es un proceso denominado descubrimiento de conocimiento en bases de datos (más conocido por su nombre en inglés como Knowledge Discovery in Databases, en adelante KDD) que abarca desde la comprensión y preparación de los datos hasta la interpretación y explotación de los resultados obtenidos a partir de los mismos (1). La minería de datos es el paso particular de este proceso que consiste en la aplicación de algoritmos específicos para la extracción de patrones a partir de los datos (2).

En algunas empresas, instituciones u organizaciones en los momentos en que hace falta la información deseada, esta suele ser en ocasiones, vaga e imprecisa. En estos casos es aconsejable manejar los datos cualitativos y no los cuantitativos por lo que es más conveniente trabajar con conjuntos difusos y variables lingüísticas (3), siendo esta última aquella variable cuyos valores son palabras o frases en un lenguaje natural o artificial, ejemplo alto, medio, bajo, etc. (4).

La presente investigación se desarrolló en la Universidad de las Ciencias Informáticas, en la que los proyectos aumentan cada vez más el volumen de información, lo que hace más difícil obtener la

información deseada, en el momento preciso. Actualmente se cuenta con variadas herramientas destinadas a la minería de datos, las cuales pueden facilitar el trabajo en los proyectos del centro a la hora de tomar alguna decisión, estas herramientas cuentan con diversas funcionalidades propias y librerías actualizadas.

Utilizar una de estas herramientas para permitir la extracción del conocimiento de las bases de datos en un lenguaje natural fácilmente comprensible para el ser humano con el fin de utilizarla en el proceso de toma de decisiones en la gestión de los mismos es de gran utilidad.

En todos los centros de la universidad está desplegado el paquete de herramientas GESPRO v12.05 (3), como solución integral para la gestión de proyectos. En el mismo se registran un conjunto de datos usados para la gestión de los proyectos, entre los que se encuentran:

- Cantidad de recursos humanos (RRHH), distribuidos por tipo de recursos, dígase estudiantes por años, profesores de las facultades o de los centros de desarrollo, especialistas.
- Cantidad de personal externo a la universidad.
- Evaluaciones de los RRHH.
- Cantidad de requisitos, propuestos, aprobados, pospuestos y terminados.
- Cantidad de riesgos, caracterizados por Bajo, Medio y Alto.
- Equipamientos de los proyectos.
- Fondo de tiempo.
- Financiamiento.
- Cantidad de No Conformidades (NC).

La mayoría de la información almacenada en GESPRO corresponde a datos cuantitativos, lo cual dificulta el proceso de toma de decisiones. GESPRO cuenta con un módulo Reportes el cual apoya la toma de decisiones, esto tiene el inconveniente de que la decisión debe ser tomada analizando estos reportes y no brinda una herramienta que auxilie en su interpretación.

Los indicadores que se tienen en cuenta para la evaluación de los proyectos son: Índice de Calidad del Dato (ICD), Índice de Rendimiento de la Logística (IRL), Índice de Rendimiento de los Recursos Humanos (IRRH), Índice de Rendimiento de la Eficacia (IREF), Índice de Rendimiento de Costos (IRC),

Índice de Rendimiento de la Planificación (IRP), Índice de Rendimiento de la Ejecución (IRE) (3).

Sin embargo a pesar de que se tenga la evaluación de estos indicadores aún existen deficiencias en la toma de decisiones pues no se brinda información suficiente sobre el estado de los proyectos, además de no especificarse las causas de las evaluaciones obtenidas de los indicadores, ni su estado, debido a que la gran cantidad de los datos almacenados en las bases de datos históricas son numéricos y no se usan.

Teniendo en cuenta lo descrito anteriormente se puede decir que las principales dificultades que afectan la toma de decisiones en los proyectos del centro son la presencia de grandes cantidades de datos numéricos, la ambigüedad e incertidumbre en los datos y la insuficiencia de información.

Debido a la situación anterior se plantea como **problema** a resolver ¿cómo facilitar la extracción de conocimiento de las bases de datos históricas de los proyectos de la Universidad de las Ciencias Informáticas, de manera que se apoye el proceso de toma de decisiones en la gestión de proyectos?

Por consiguiente **el objeto de estudio** estará enfocado en el proceso de extracción del conocimiento. Siendo el **objetivo general** de la investigación desarrollar un módulo en R para la extracción de resúmenes lingüísticos que apoyen el proceso de toma de decisiones en la gestión de proyectos.

A partir del objetivo general de la investigación se definen los siguientes **objetivos específicos**

1. Establecer los referentes teóricos y metodológicos relacionados con los resúmenes lingüísticos de datos y aprendizaje de reglas de asociación.
2. Definir las funcionalidades a implementar en el módulo.
3. Implementar las funcionalidades definidas.
4. Validar el módulo desarrollado a partir de experimentación con bases de datos de proyectos terminados.

Concretamente se enfoca el **campo de acción** hacia la investigación de los resúmenes lingüísticos de datos a partir de reglas de asociación, siguiendo la **idea a defender,** si se desarrolla un módulo en

R que permita la extracción de resúmenes lingüísticos, se contribuirá a la toma de decisiones en gestión de proyectos en los centros de la universidad.

Como **tareas** que complementen los objetivos específicos para obtener resultados satisfactorios en esta investigación se definen:

1. Revisión documental de investigaciones que trabajan el tema de resúmenes lingüísticos de datos y aprendizaje basado en reglas de asociación.
2. Selección de las herramientas a usar en la investigación.
3. Definición de los componentes del proceso de extracción del conocimiento a desarrollar.
4. Selección del algoritmo a usar para la extracción de las reglas de asociación.
5. Definición de la estructura a usar en los resúmenes.
6. Definición de las funcionalidades del módulo a desarrollar.
7. Implementación de las funcionalidades definidas.
8. Aplicación del módulo desarrollado en los datos de los proyectos de la Universidad.

Teniendo como **posible resultado** un módulo en R para la extracción de resúmenes lingüísticos.

Para dar cumplimiento a las tareas de investigación se utilizaron los siguientes **métodos**:

**Métodos teóricos:**

- **Analítico-sintético:** se utilizó para hacer un análisis y comprensión de la documentación, estudiando las características de los sistemas existentes para la minería de datos.
- **Histórico-lógico:** se utilizó para estudiar los conceptos y la evolución de los sistemas de minería de datos.

**Estructura del trabajo**

# Introducción

El presente trabajo consta de tres capítulos:

**Capítulo I:** contiene la fundamentación teórica de la propuesta, un estudio del estado del arte sobre herramientas de minería de datos y algoritmos para la extracción de reglas de asociación, además de las herramientas y metodologías a utilizar.

**Capítulo II:** presenta la solución propuesta describiendo las fases de planificación, diseño y desarrollo del módulo.

**Capítulo III:** expone los resultados de las pruebas realizadas al sistema.

# Capítulo I
# Fundamentación teórica

## Capítulo I: Fundamentación teórica

En el presente capítulo se abordan varios aspectos teóricos, conceptos y definiciones relacionados con las tendencias actuales los resúmenes lingüísticos de datos a partir de reglas de asociación. Se abordan temas relacionados con el proceso de KDD y la fase de minería de datos, la metodología de software utilizada, así como las herramientas de minería de datos.

### 1.1 Proceso de toma de decisiones

En la actualidad la toma de decisiones es un proceso complejo, los administradores de proyectos triunfadores necesitan buenas capacidades de toma de decisiones para planear, organizar, dirigir y controlar de manera eficiente y eficaz los proyectos de software (5).

El proceso de toma de decisiones es un punto clave en la gestión de los proyectos. Este indica el camino a seguir para solucionar pequeños problemas que pueden parecer no tener fin, además este proceso entra en juego cada vez que en una empresa se realizan actividades de planeación, organización, dirección y control.

En (5) una decisión se define como: "... una elección que se hace a partir de las alternativas disponibles" y el proceso de toma de decisiones como: "... el proceso de identificar problemas y oportunidades y resolverlos."

Este proceso se hace más complicado en una empresa productora de software ya que es un caso distintivo en el que convergen una serie de procesos complejos. En este tipo de empresas, la toma de decisiones suele ser un proceso mal estructurado, en el que los directivos tienes que dirigir bajo condiciones desfavorables. A menudo, se enfocan en el estudio de las posibles alternativas al problema y en el análisis de las consecuencias de la decisión que en el análisis del proceso de toma de decisiones, al que no se le da apenas importancia, no se le dedican recursos, tiempo y no se tienen en cuenta puntos de vista que pueden influir en el éxito final de la decisión (6).

## 1.2 Extracción del conocimiento

El proceso de KDD persigue la extracción de conocimiento no trivial, implícito, previamente desconocido y potencialmente útil a partir de grandes volúmenes de datos (1), como bien se decía anteriormente, este involucra un proceso iterativo e interactivo de búsqueda de modelos, patrones o parámetros. Los patrones descubiertos han de ser válidos, novedosos para el sistema y potencialmente útiles (7).

Es iterativo ya que la salida de alguna de las fases puede hacer volver a pasos anteriores y porque a menudo son necesarias varias iteraciones para extraer conocimiento de alta calidad. Es interactivo porque el usuario, o más generalmente un experto en el dominio del problema, debe ayudar en la preparación de los datos y validación del conocimiento extraído (8).

Este proceso está constituido por una serie de etapas las cuales son (8):

**Integración y recopilación de datos:** se determinan las fuentes de información que pueden ser útiles y dónde conseguirlas. A continuación, se transforman todos los datos a un formato común, frecuentemente mediante un almacén de datos que consiga unificar de manera operativa toda la información recogida, detectando y resolviendo las inconsistencias.

**Selección, limpieza y transformación:** se eliminan o corrigen los datos incorrectos y se decide la estrategia a seguir con los datos incompletos. Además, se proyectan los datos para considerar únicamente aquellas variables o atributos que van a ser relevantes, con el objetivo de hacer más fácil la tarea propia de minería y para que los resultados de la misma sean más útiles.

Suelen englobarse las dos primeras fases bajo el nombre de "preparación de datos".

**Minería de datos:** se decide cuál es la tarea a realizar (clasificar, agrupar, etc.) y se elige el método que se va a utilizar.

**Evaluación e interpretación:** se evalúan los patrones y se analizan por los expertos y si es necesario se vuelve a las fases anteriores para una nueva iteración. Esto incluye resolver posibles conflictos con

el conocimiento que se disponía anteriormente.

**Difusión y uso:** se hace uso del nuevo conocimiento y se hace partícipe de él a todos los posibles usuarios.

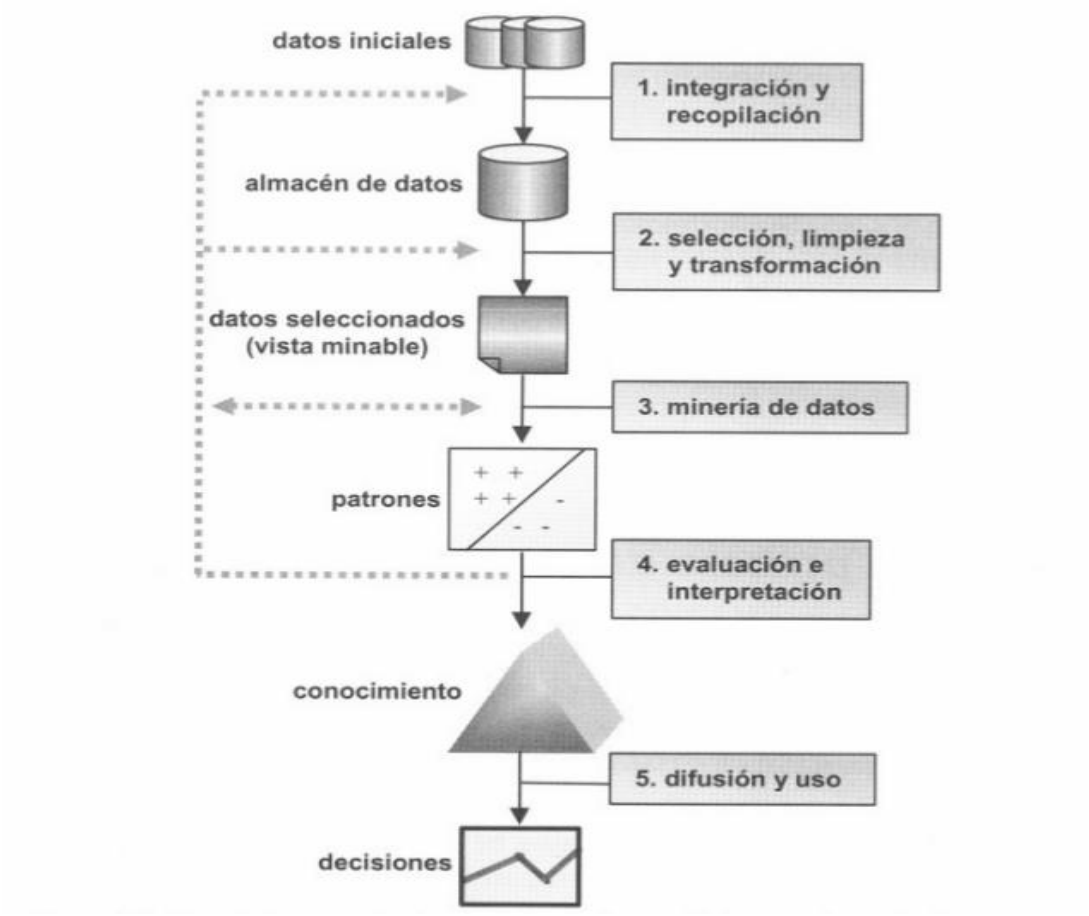


**Ilustración 1: Fases de KDD**

## 1.3 Minería de datos

Es la etapa de descubrimiento dentro del proceso de KDD y consiste en el uso de algoritmos concretos que generan una enumeración de patrones a partir de los datos procesados (2). Así, la minería de datos se centra en el análisis de los datos y en la aplicación de algoritmos que, bajo limitaciones

aceptables de eficiencia computacional, obtengan patrones (o modelos) sobre los datos (2).

El uso de la Minería de Datos está relacionado con los siguientes objetivos (3):

- Predicción: predecir posibles situaciones futuras sobre la base de los acontecimientos anteriores.
- Descripción: explicar las razones por la que ocurren algunos eventos
- Verificación: examinar la existencia de algún tipo de relación entre entidades.
- Detección de excepciones: Detectar situaciones (registros) en las bases de datos que se corresponden a algo inusual.

Los dos retos fundamentales de la minería de datos es trabajar con grandes volúmenes de datos y usar técnicas adecuadas para analizar los mismos y extraer conocimiento útil y novedoso, por lo que se puede decir que el objetivo de esta fase es convertir datos en conocimiento (8).

A continuación se describen los diferentes tipos de problemas que es posible abordar usando técnicas de minería de datos.

### 1.3.1 Clasificación

Se utiliza para clasificar un conjunto de datos basado en los valores de sus atributos. Encuentra las propiedades comunes entre un conjunto de objetos y los clasifica en diferentes clases, de acuerdo a un modelo de clasificación. Para construir este modelo, se utiliza un conjunto de entrenamiento, en el que cada instancia consiste en un conjunto de atributos y el valor de la clase a la cual pertenece. El objetivo de la clasificación es analizar los datos de entrenamiento y mediante un método supervisado, desarrollar una descripción o un modelo para cada clase utilizando las características disponibles en los datos (9). Las técnicas de clasificación más importantes son (1):

- Árboles de decisión (principales algoritmos ID3, C4.5, C5.0)

- Clasificación bayesiana
- Redes neuronales artificiales (principales redes Perceptrón Simple Capa, Perceptrón Multicapa)

### 1.3.2 Regresión

Su objetivo es predecir el valor desconocido de un atributo de un determinado individuo a partir de valores conocidos de otros atributos de dicho individuo. Existen dos tipos de regresión: lineal y no lineal (1).

La regresión lineal es la forma más simple de regresión, en ella se modelan los datos usando una línea recta. En este tipo de regresión se utiliza una variable aleatoria ***y*** (variable respuesta), que es una función lineal de otras variables, ***ai (0 ≤ i ≤ k)***, (variables predictoras), según se muestra en la siguiente ecuación (3): $y = w_0 a_0 + w_1 a_1 + \ldots + w_k a_k$

En algunas ocasiones puede que las variables no presenten dependencia lineal. En tales casos han de emplearse técnicas de regresión no lineal, en las cuales la relación entre las variables puede ser polinómica (1).

### 1.3.3 Clustering

Agrupan datos dentro de un número de clases preestablecidas o no, partiendo de criterios de distancia o similitud, de manera que las clases sean similares entre sí y distintas con las otras clases (10). Este un proceso de aprendizaje no supervisado ya que las clases no están predefinidas sino que deben ser descubiertas dentro de los datos. La técnica de clustering tiene varias aplicaciones entre las que se detallan (1):

- Marketing: descubrimiento de distintos grupos de clientes en la base de datos. Este conocimiento puede ser útil para la política publicitaria y las ofertas.
- Seguros: identificación de grupos de asegurados con características parecidas. Gracias a ello,

es posible calcular los riesgos más fielmente.

- Planificación urbana: identificación de grupos de viviendas de acuerdo a su tipo, valor o situación geográfica.

- World Wide Web (WWW): clasificación de documentos, analizar ficheros log para describir patrones de acceso similares, etc.

### 1.3.4 Reglas de asociación

Las reglas de asociación tienen como objetivo identificar relaciones no explícitas entre variables del conjunto de datos atributos y se suelen expresar mediante reglas del tipo "si el atributo ***X*** toma el valor ***a***, entonces el atributo ***Y*** toma valor ***b***" (2). Una forma más detallada de expresarlas es: el 90% de la gente que compra pan, compra leche, en lenguaje natural se puede expresar como: “la mayoría de la gente que compra pan compra leche”, pudiéndose expresar mediante la notación pan ⇒ leche, en esta notación la parte izquierda se denomina antecedente y la parte derecha consecuente (11).

Se puede describir formalmente como:

Sea $I = \{i_1, i_2, \ldots i_m\}$ un conjunto de literales, llamados ítems. Sea ***D*** un conjunto de transacciones, donde cada transacción ***T*** es el conjunto de ítems, tal que $T \subseteq I$. Cada transacción lleva asociado un identificador, **TID**. Se dice que una transacción ***T*** contiene a ***X*** (un conjunto de ítems de ***I***), si $X \subseteq T$. Una regla de asociación es una implicación de la forma $X \Rightarrow Y$, donde $X \subset I$, $Y \subset I$, y $X \cap Y = \emptyset$. La regla $X \Rightarrow Y$ tiene en el conjunto de transacciones ***D*** una confianza de ***c*** si el ***c%*** de las transacciones de ***D*** que contienes ***X***, también contienen ***Y***. La regla $X \Rightarrow Y$ y tiene un soporte de ***s*** si el ***s%*** de las transacciones de ***D*** contienen $X \cup Y$ (1).

Dado el conjunto de transacciones ***D***, el principal objetivo es generar todas las reglas de asociación que tienen un soporte y una confianza mayores que unos valores concretos establecidos por el usuario (1).

El soporte y la confianza son dos medidas básicas de control, las cuales definen el grado de interés

de la regla. El “soporte” se define como el porcentaje o fracción de registros que contienen a ***X U Y*** del total de registros presentes en la base de datos, mientras que la “confianza” se define como el porcentaje o fracción del número de transacciones que contienen a ***X U Y*** sobre el número total de registros que contienen a ***X*** (12).

### 1.4 Sumarización lingüística

La sumarización lingüística o resúmenes lingüísticos están destinados a capturar características esenciales de los datos originales de acuerdo con las necesidades del usuario. Por lo tanto, proporcionan al usuario una visión más simple de los datos recogidos en una base de datos (13).

Los resúmenes lingüísticos tienen el propósito general de ser una descripción humana coherente de conjuntos de datos, que captan las tendencias principales de los datos. Estos resúmenes no están destinados a ser un sustituto para el análisis estadístico clásico, sino más bien un medio alternativo de representación de los datos que se centra en la comprensibilidad y la interpretación humana rápida, los resúmenes son una breve descripción de las tendencias de los datos expresados en lenguaje natural (14).

La información y el conocimiento lingüístico se pueden obtener de las bases de datos a través de muchos algoritmos diferentes, métodos computacionales y bajo muchas suposiciones (15). Existen distintos enfoques para los resúmenes lingüísticos de datos. El más conocido es el basado en conjuntos difusos, introducido por Yager y desarrollados por muchos otros (3).

Con este enfoque se pueden generar resúmenes como:

- “El 90% de la gente que compra pan compra leche”
- “La mayoría de la gente que compra pan compra leche”.

También se pueden generar resúmenes lingüísticos a partir de reglas de asociación de la forma: Si ***X*** es grande e ***Y*** es medio entonces ***Z*** es pequeño (16).

Yager brindó una aproximación a la terminología usada en los resúmenes lingüísticos, siendo (3):

- ***V*** una cualidad (atributo) de interés, por ejemplo “Fondo de tiempo” de los trabajadores.
- ***Y*** un conjunto de objetos (records) que manifiestan la cualidad, ejemplo: conjunto de trabajadores de los proyectos. Por lo que ***V*** $(y_i)$ representa el valor de la cualidad ***V*** para el trabajador $y_i$, por ejemplo para ***V*** $(y_1)$=56 horas.
- ***D***= $\{V(y_1),\ldots, V(y_n)\}$ un conjunto de datos, es decir la base de datos en cuestión.

Entonces un resumen lingüístico está compuesto por (3):

- Un sumarizador ***S***, ejemplo: trabajadores especialistas,
- Una medida de cantidad, ejemplo: “la mayoría”,
- Medida de calidad, ejemplo: la veracidad del resumen expresada en un número entre 0 y 1.

### 1.4.1 Sumarización lingüística a partir de reglas de asociación

Usar reglas de asociación, como se mencionó anteriormente, es otra forma de generar los resúmenes lingüísticos ejemplo: Si $X_1$ es $S_1$ entonces $Y_1$ es $S_2$, siendo $X_1$ e $Y_1$ atributos de los objetos y $S_1$ y $S_2$ son los sumarizadores, generalmente modelados por términos difusos, usados en el resumen (3). Ejemplo: “Si los trabajadores son especialistas entonces la calidad del proyecto es media”

La siguiente tabla muestra los componentes de un resumen lingüístico (3) (16):

**Tabla 1: Componentes de un resumen lingüístico**

| Componente | Significado | Ejemplo |
|---|---|---|
| D | Base de Datos completa | Base de datos de Gestión de proyectos |
| Y | Conjunto de todos los objetos | Todos los proyectos en la base de datos |
| M | Número de objetos en Y | Total de proyectos (104) |
| $y_m$ | El objeto $m^{th}$ (m-ésimo) | El m-ésimo proyecto de la base de datos (Proyecto 5) |
| $v_n$ | Nombre del atributo $n^{th}$ (n-ésimo) | IE; Índice de Ejecución (quinto atributo) |
| $X_n$ | Dominio de $v_n$ | [0,100] para IE |
| V | Conjunto de todos los nombres de atributos | [Cant. Estudiantes, Cant. Profesores, |

| | | |
|---|---|---|
| | | Fondo de tiempo, Cant. NC, IE] |
| $v_n^m$ | Valor del atributo $n^{th}$ correspondiente a $y_m$ | 80 (IE del décimo proyecto) |
| $d_m$ | Todos los valores de los atributos para $y_m$ | [15, 8, Alto, 12, 85] para el 10mo. Proyecto |
| $S_n$ | Sumarizador | IE alto |
| Q | Cuantificador | Muchos, más que 20 |
| T | Grado de verdad | Valores entre [0,1] |
| C | Grado de generalidad | Valores entre [0,1] |
| U | Grado de utilidad | Valores entre [0,1] |
| O | Grado de novedad | Valores entre [0,1] |
| S | Grado de simplicidad | Valores entre [0,1] |

Batyrshin y Wagenknecht (17) presentan un modelo de descripción lingüística de datos basado en reglas de asociación, así como el manejo de los términos lingüísticos usados en las reglas. El resultado de dicho modelo consiste en reglas de asociación, las cuales pueden ser fácilmente interpretable por los usuarios. Como inconveniente de este trabajo se considera la forma en que son mostrados los resultados, pues no se le incorporan otros componentes de los resúmenes lingüísticos, como son los sumarizadores y los cuantificadores. A su vez se puede mencionar que la extracción de las reglas se hace mediante un algoritmo genético, lo que aplicándose en un marco donde se tenga una gran cantidad de datos elevaría el costo computacional (3).

Chen y colaboradores (18) presentan otro enfoque donde se usan reglas de asociación, el modelo propuesto trabaja principalmente con datos de series temporales. En este enfoque se presenta como negativo la posibilidad de que el número de reglas obtenidas sea inmanejable para los usuarios (3).

Se puede resumir que en los últimos años muchos autores han propuesto enfoques para la extracción de resúmenes lingüísticos. El enfoque basado en conjuntos difusos ha sido el más desarrollado aunque el enfoque basado en reglas de asociación empleado en otras investigaciones se perfila como el más apropiado para ser usado en apoyo a la toma de decisiones. A pesar de la existencia de estos algoritmos, se necesitan nuevos métodos de generar resúmenes lingüísticos, específicamente en

forma de reglas de asociación, que apoyen el proceso de toma de decisiones, ya que el mayor problema que presentan los mencionados anteriormente es la gran cantidad de reglas que se pueden obtener, lo cual dificulta más este proceso (3).

Según Hirota y Pedrycz, las siguientes cinco características son esenciales para medir la calidad de un resumen (16):

1. Validez: Los resúmenes deben ser derivados de datos con alta confianza.
2. Generalidad: Describe cuántos datos apoyan el resumen.
3. Utilidad: Se refiere a los objetivos del usuario con los resúmenes, especialmente en términos del impacto que estos resúmenes pueden tener en la toma de decisiones.
4. Novedad: Esto describe el grado en que los resúmenes se desvían de las expectativas del usuario.
5. Simplicidad: Esta medida se refiere a la complejidad sintáctica de los resúmenes.

## 1.5 Algoritmos para la extracción de reglas de asociación

Actualmente existen en el mundo una diversidad de algoritmos para la extracción de reglas de asociación, a continuación se exponen algunos de ellos.

### 1.5.1 Apriori

En 1993-1994 Agrawal propone el algoritmo “Apriori” para la búsqueda de reglas de asociación en bases de datos. El nombre del algoritmo se basa en el principio general del mismo nombre y denota que un “subconjunto no vacío de un itemset frecuente también resulta ser frecuente” (12), ya que el soporte de un itemset nunca puede ser mayor que el de cualquiera de sus subconjuntos. Formalmente esta propiedad se conoce como anti-monotonía del soporte.

Apriori busca ítemsets frecuentes usando la generación de ítems candidatos y se resume en dos pasos (19):

1. Genera todos los ítemsets que contienen un solo elemento, utilizándolos luego para generar ítemsets que contengan dos elementos, y así sucesivamente. Se toman todos los posibles

pares de ítems que cumplen con las medidas mínimas de soporte inicialmente preestablecidas; esto permite ir eliminando posibles combinaciones, aquellas que no cumplan con los requerimientos de soporte no entrarán en el análisis.

2. Genera las reglas revisando que cumplan con el criterio mínimo de confianza. Es interesante observar que si una conjunción de consecuentes de una regla cumple con los niveles mínimos de soporte y confianza, sus subconjuntos (consecuentes) también los cumplen; en el caso contrario, si algún ítem no los cumple no tiene caso considerar sus subconjuntos.

### 1.5.2 Apriori Difuso

El objetivo principal del algoritmo Apriori difuso, introducido por Hong y otros, es encontrar ítems relevantes así como reglas de asociación difusas en las instancias con valores cuantitativos, descubriendo interesantes patrones (20).

Este método consiste en transformar cada valor cuantitativo en un conjunto difuso de etiquetas lingüísticas asumiendo que las funciones de pertenencia son conocidas de antemano. El algoritmo posteriormente calcula la cardinalidad de cada ítem difuso, a lo que denomina "cuenta". Si el valor de cuenta del ítem difuso es superior o igual que el valor del mínimo soporte este ítem será considerado un ítem difuso frecuente. A continuación combina los ítems frecuentes y vuelve a repetir el proceso. Finalmente, este método obtiene las reglas de asociación difusas mediante el criterio del algoritmo Apriori (20).

### 1.5.3 FP- Growth

Se basa en una estructura de árboles de patrones frecuentes usándolos para almacenar la información principal de la base de datos. El algoritmo hace escaneos de la base de datos sólo dos veces, evitando de esta forma múltiples pasadas y reduciendo los tiempos. No necesita generar candidatos, se reduce así gran cantidad de tiempo consumido en la generación de los candidatos y sus pruebas. Utiliza el principio de “divide y vencerás” por lo que el espacio de búsqueda se reduce de forma significativa. En líneas generales, la magnitud de rapidez de procesado es mayor que en el algoritmo “Apriori”. No obstante, existen algunos tópicos a mejorar de este algoritmo, en especial la reducción del número de

árboles condicionales FP ya que estos absorben memoria y tiempo de cómputo (12).

### 1.5.4 DHP

En el algoritmo de poda y hashing directa (DHP, Direct hashing and Pruning) se emplea una técnica de hash para eliminar todos los conjuntos de ítems innecesarios para la generación del próximo conjunto de ítems. Cada (k+1)-ítemset es añadido a una tabla hash en un valor hash dependiente de las ocurrencias en la BD de los conjuntos candidatos de k elementos que lo formaron, o sea, dependiente del soporte de los conjuntos candidatos de k elementos. Estas ocurrencias son contadas explorando en las transacciones de la BD. Si el soporte asociado a un valor hash es menor que el soporte mínimo entonces todos los conjuntos de ítems de k+1 elementos con este valor hash no serán incluidos entre los candidatos de k+1 elementos en la próxima pasada (21).

De todos los algoritmos existentes en la literatura, Apriori ha sido uno de los de mayor impacto y posiblemente el más referenciado (22), además de ser el algoritmo que sobresale en la minería de datos para la extracción de reglas de asociación, por tal motivo se decide utilizarlo en esta investigación, además la presente investigación se basa en la tesis de maestría de Eric Eduardo Piñera Trinchet, en la cual se propone el algoritmo para resúmenes lingüísticos y utiliza Apriori para extraer las reglas de asociación.

## 1.6 Herramientas para la minería de datos

### Weka

Es de libre distribución (licencia GPL) y destacada por la cantidad de algoritmos que presenta así como por la eficiencia de los mismos, está desarrollada por miembros de la Universidad de Waikato. Proporciona gran cantidad de herramientas para la realización de tareas propias de minería de datos. Soporta varias tareas estándar de minería de datos, especialmente, reprocesamiento de datos, clustering, clasificación, regresión, visualización y selección (10).

Es válido destacar que debido a que Weka es una herramienta bajo la licencia GPL, es posible actualizar su código fuente para incorporar o modificar funcionalidades.

**SPSS Clementine:**

Clementine además de ser multiplataforma se centra en la integración de minería de datos con otros procesos y sistemas de negocio que ayuden a entregar inteligencia predictiva en un tiempo eficiente durante las operaciones de negocio diarias. Es un sistema integrado de minería de datos que permite encontrar patrones en la información para facilitar la toma de decisiones a los usuarios (10).

**R**

Es un lenguaje y entorno de programación, creado en 1993 por Ross Ihaka y Robert Gentleman del Departamento de Estadística de la Universidad de Auckland, cuya característica principal es que forma un entorno de análisis estadístico para la manipulación de datos, su cálculo y la creación de gráficos. R se distribuye gratuitamente bajo los términos de la GNU (General Public Licence) (23).

R ofrece una amplia variedad de estadísticas (modelado lineal y no lineal, pruebas estadísticas clásicas, análisis de series temporales, clasificación, agrupación), además de técnicas gráficas (24), además de ser polimórfico, lo que significa que la misma función se puede aplicar a diferentes tipos de objetos, con resultados adaptados a los diferentes tipos de objetos. Tal función se llama una función genérica (25).

Una característica del lenguaje R es que permite al usuario combinar en un solo programa diferentes funciones estadísticas para realizar análisis más complejos (23). R forma parte de un proyecto colaborativo y abierto. Existe un repositorio oficial de paquetes cuyo número supera los 5583, éstos se han organizado en vistas (o temas), que permiten agruparlos según su naturaleza y función (3).

**RapidMiner / Yale**:

Anteriormente conocido como Yale (por sus siglas en inglés: Yet Another Learning Environment) y desarrollado sobre el lenguaje Java. Trabaja sobre las plataformas Windows y Linux. Además de ser una herramienta flexible para aprender y explorar la minería de datos, la interfaz gráfica de usuario tiene como objetivo simplificar el uso para las tareas complejas de esta área (3). Actualmente se publica bajo los términos de la licencia AGPL (Affero General Public License).

Proporciona más de 500 operadores orientados al análisis de datos, incluyendo los necesarios para realizar operaciones de entrada y salida, pre-procesamiento de datos y visualización. Permite la integración con Weka, a través del uso de los algoritmos incluidos en este último y con otros programas a través de llamadas a las bibliotecas de RapidMiner. Además permite el desarrollo de programas a través de un lenguaje de script. Incluye gráficos y herramientas de visualización de datos y dispone de un módulo de integración con R (3).

**SAS Enterprise Miner**

Su compañía es SAS, es una solución de minería de datos que permite incorporar patrones inteligentes a los procesos de marketing, tanto operativos como estratégicos. El software de SAS, es un sistema de entrega de información que provee acceso transparente a cualquier fuente de datos, incluyendo archivos planos, archivos jerárquicos, y los más importantes manejadores de bases de datos relacionales. También incluye su propia base de datos de información para almacenar y manejar los datos, es decir, un "data warehouse". El sistema soporta un amplio rango de aplicaciones, destacándose el análisis estadístico, análisis gráfico de datos, análisis de datos guiado, mejoramiento de la calidad, diseño experimental, administración de proyectos, programación lineal y no lineal, generación de reportes y gráficas, manipulación y despliegue de imágenes, sistemas de información geográfica, visualización multidimensional de datos, aplicaciones de multimedia, así como los sistemas de información ejecutiva. (10)

**KNIME:**

Fue desarrollado originalmente en el departamento de bioinformática y minería de datos de la Universidad de Constanza, Alemania. KNIME está desarrollado sobre la plataforma Eclipse y programado, esencialmente, en Java. Está concebido como una herramienta gráfica y dispone de una serie de nodos (que encapsulan distintos tipos de algoritmos) y flechas (que representan flujos de datos) que se despliegan y combinan de manera gráfica e interactiva. KNIME integra diversos componentes para aprendizaje automático y minería de datos a través de su concepto de fraccionamiento de datos modular. La interfaz gráfica de usuario permite el montaje fácil y rápido de

nodos para procesamiento de datos (ETL: extracción, transformación, carga), para el análisis de datos, modelado y visualización (3).

Como bien se decía anteriormente la presente investigación forma parte de una maestría en la cual se eligió como lenguaje de programación R, además de eso se puede ver en la Ilustración 2 el comportamiento entre las diferentes herramientas mencionadas anteriormente y otras, en años anteriores (2012-2013). Como se puede observar R entra entre las 3 primeras herramientas más usadas ya sea como entorno o lenguaje de programación, ocupando la posición número 2. El uso de este es bastante aceptado en la actualidad ya que el porciento de diferencia entre el 2012 y el 2013 es relativamente pequeño, por lo que se puede decir que se usa con bastante frecuencia, además de contar con varias librerías que apoyan el trabajo con los datos, por lo que se usará R en su versión 2.14.1.

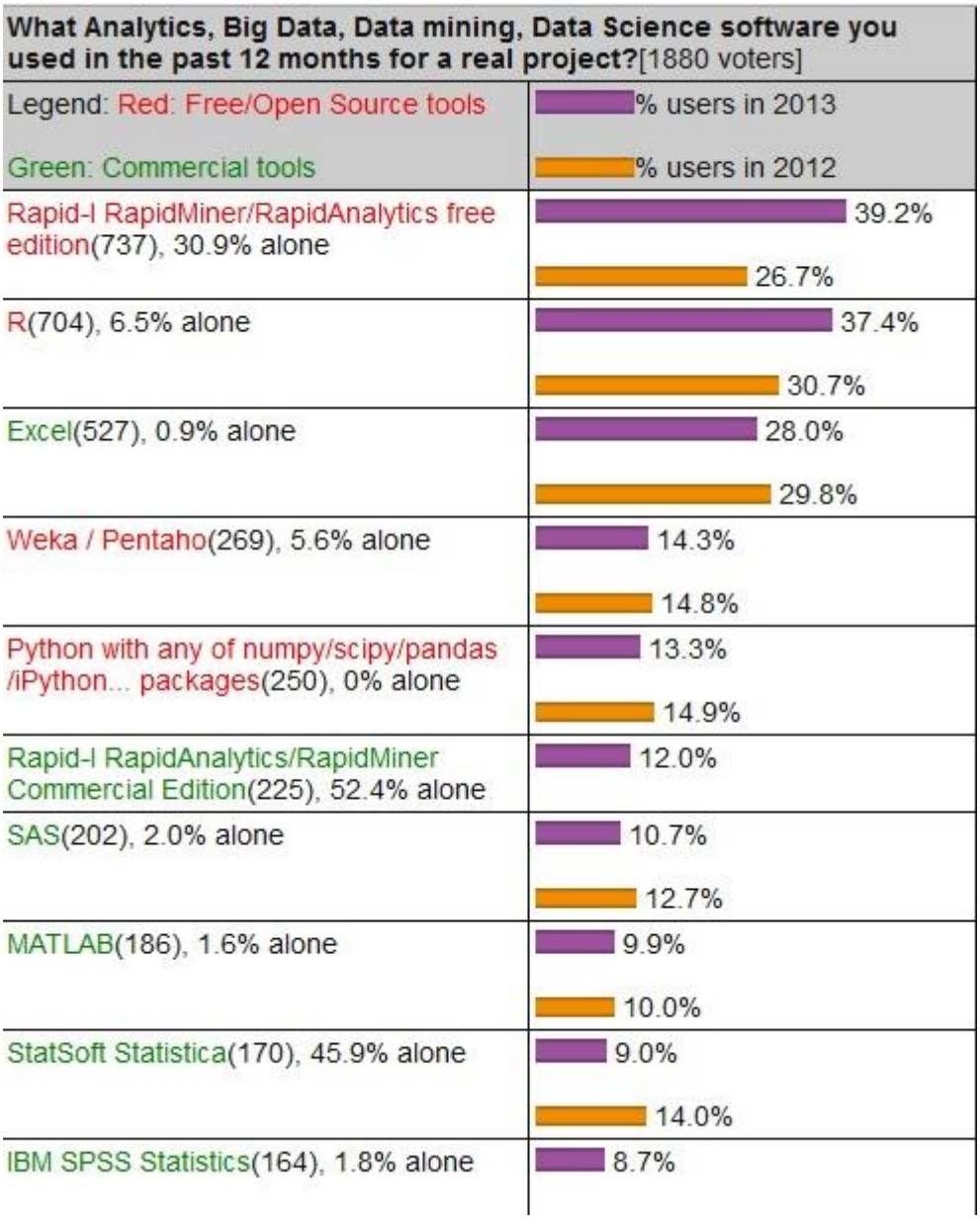


**Ilustración 2: Comparación de las herramientas de minería de datos.** *(26)*

## 1.7 Metodologías de desarrollo de software

Una metodología es un conjunto integrado de técnicas y métodos que permite abordar de forma homogénea y abierta cada una de las actividades del ciclo de vida de un proyecto de desarrollo. Es

un proceso de software detallado y completo (27).

Las metodologías se basan en una combinación de los modelos de proceso genéricos (cascada, incremental…). Definen artefactos, roles y actividades, junto con prácticas y técnicas recomendadas (27).

### 1.7.1 Scrum

Scrum es un proceso ágil y liviano que sirve para administrar y controlar el desarrollo de software. El desarrollo se realiza en forma iterativa e incremental (una iteración es un ciclo corto de construcción repetitivo). Cada ciclo o iteración termina con una pieza de software ejecutable que incorpora nuevas funcionalidades. Las iteraciones en general tienen una duración entre 2 y 4 semanas. Scrum se utiliza como marco para otras prácticas de ingeniería de software como RUP o Extreme Programming (28).

La intención de Scrum es la de maximizar la retroalimentación sobre el desarrollo pudiendo corregir problemas y mitigar riesgos de forma temprana. Cabe mencionar que Scrum no propone el uso de ninguna práctica de desarrollo en particular.

En Scrum, el equipo se focaliza en una única cosa: construir software de calidad. Por el otro lado, la gestión de un proyecto Scrum se focaliza en definir cuáles son las características que debe tener el producto a construir (qué construir, qué no y en qué orden) y en remover cualquier obstáculo que pudiera entorpecer la tarea del equipo de desarrollo. Se busca que los equipos sean lo más efectivos y productivos posible (29).

### 1.7.2 AUP

El AUP es un acercamiento aerodinámico de desarrollo del software basado en el Proceso Unificado Rational de IBM (RUP), basado en disciplinas y entregables incrementales con el tiempo. El ciclo de vida en proyectos grandes es serial mientras que en los pequeños es iterativo. Las disciplinas de AUP son (29):

- Modelado

- Implementación
- Prueba
- Despliegue
- Administración de la configuración
- Administración o gerencia del Proyecto
- Entorno

### 1.7.3 Extreme Programming (XP)

XP es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en el desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores y propiciando un buen clima de trabajo. XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios (30).

Las características fundamentales de XP son (29):

- **Desarrollo iterativo e incremental:** pequeñas mejoras, unas tras otras.
- **Pruebas unitarias continuas:** frecuentemente repetidas y automatizadas, incluyendo pruebas de regresión. Se aconseja escribir el código de la prueba antes de la codificación.
- **Programación por parejas:** se recomienda que las tareas de desarrollo se lleven a cabo por dos personas en un mismo puesto. Se supone que la mayor calidad del código escrito de esta manera -el código es revisado y discutido mientras se escribe- es más importante que la posible pérdida de productividad inmediata.
- **Frecuente interacción del equipo de programación con el cliente o usuario:** se recomienda que un representante del cliente trabaje junto al equipo de desarrollo.

- **Corrección de todos los errores** antes de añadir nuevas funcionalidades. Hacer entregas frecuentes.

- **Refactorización del código:** es decir, reescribir ciertas partes del código para aumentar su legibilidad y mantenibilidad pero sin modificar su comportamiento. Las pruebas han de garantizar que en la refactorización no se ha introducido ningún fallo.

- **Propiedad del código compartida:** en vez de dividir la responsabilidad en el desarrollo de cada módulo en grupos de trabajo distintos, este método promueve el que todo el personal pueda corregir y extender cualquier parte del proyecto. Las frecuentes pruebas de regresión garantizan que los posibles errores serán detectados.

- **Simplicidad en el código:** es la mejor manera de que las cosas funcionen. Cuando todo funcione se podrán añadir funcionalidades si es necesario. La programación extrema apuesta que en más sencillo hacer algo simple y tener un poco de trabajo extra para cambiarlo si se requiere, que realizar algo complicado y quizás nunca utilizarlo.

Además propone varias prácticas con el fin de lograr el éxito final del proyecto, estas son: pruebas, refactorización, programación en pareja, cliente in-situ, estándares de programación, 40 horas por semana, integración continua, propiedad colectiva del código, diseño simple, metáfora, entregas pequeñas y el juego de la planificación.

Después de realizado el estudio de algunas metodologías de desarrollo de software, se selecciona para el desarrollo del módulo XP por ser una metodología ágil, diseñada para equipos de trabajo pequeños y centrada en vincular al cliente en el ciclo de desarrollo. La metodología XP permite de manera eficiente los cambios que se puedan presentar durante todo el desarrollo del módulo proponiendo un ciclo de vida dinámico. El proceso de prueba de XP posibilita probar cada funcionalidad al finalizar cada iteración comprobando si cumple con los requisitos del módulo. De esta metodología se utilizaran algunos artefactos como son: las Historia de Usuario (en los adelante HU), el plan de iteraciones en la fase de planificación, las tarjetas Cargo o Clase, Responsabilidad y Colaborador (en lo adelante CRC), en las fase de diseño, las tareas de programación y los estándares

de codificación en la fase de desarrollo y en la fase de prueba las pruebas unitarias y de aceptación.

## 1.8 Entorno de desarrollo

### Rstudio 0.98.501

Al igual que R, RStudio es un proyecto de código libre y abierto. Es un entorno de desarrollo integrado (IDE) para R. El término IDE proviene de la industria del software y se refiere a una herramienta que facilita el desarrollo de aplicaciones en uno o más lenguajes de programación.

RStudio tiene un editor de texto muy avanzado, un sistema de ayuda de R, control de versiones y mucho más en una sola aplicación. No realiza ninguna operación estadística, sino que sólo hace que sea más fácil para llevar a cabo este tipo de operaciones con R. Lo más importante es que RStudio ofrece muchas instalaciones que hacen que puedas trabajar mucho más fácil.

Se usará como lenguaje de programación R en su versión 2.14.1

## Conclusiones del capítulo

Después de realizado el estudio bibliográfico se puede arribar a las siguientes conclusiones:

- En cuanto a los enfoques y algoritmos estudiados para la sumarización lingüística se puede decir que ninguno de los analizados es explotado al máximo para apoyar la toma de decisiones.
- El empleo de la técnica de reglas de asociación y del algoritmo Apriori para extráelas resulta útil para realizar resúmenes lingüísticos que apoyen el proceso de toma de decisiones.
- Para el desarrollo del módulo se utilizarán varias tecnologías libres, contribuyendo a la independencia tecnológica por las que se abogan en la UCI.

# Capítulo II
# Descripción de la solución propuesta

## Capítulo II: Descripción de la solución propuesta

En el siguiente capítulo se describe de forma general el módulo a desarrollar siguiendo cada una de las fases de la metodología XP (planificación, diseño, desarrollo).

## 2.1 Objeto de informatización

Con el presente trabajo se pretende obtener un módulo en R que permita la extracción de resúmenes lingüísticos para agilizar el proceso de toma de decisiones en la gestión de proyectos en la UCI

Este módulo concibe que a partir de bases de datos de los proyectos del centro se puedan obtener reglas de asociación para luego realizar resúmenes lingüísticos a partir de ellas que tengan como propósito general ser una descripción humana coherente de un conjunto de datos y que cumplan con las principales características de los resúmenes:

- Validez
- Generalidad
- Utilidad
- Novedad
- Simplicidad

## 2.2 Pasos para la sumarización lingüística

En el presente epígrafe se muestran los pasos a seguir para realizar los resúmenes lingüísticos. El algoritmo utilizado para este módulo es el resultado de una investigación de maestría (3).

La estructura adecuada para trabajar con los datos es un arreglo bidimensional ***(mxn)*** (en matemática estos arreglos se llaman matrices), donde ***m*** (las columnas) representan los atributos y ***n*** (las filas) las observaciones, ejemplo:

**Tabla 2: Ejemplo de estructura de un arreglo rectangular**

| | $A_1$ | $A_2$ | … | $A_m$ |
|---|---|---|---|---|

# Capítulo II

# Descripción de la solución propuesta

| | | | | |
|---|---|---|---|---|
| $O_1$ | | | | |
| $O_2$ | | | | |
| … | | | | |
| $O_n$ | | | | |

A continuación se presentan y explican cada uno de los pasos a seguir:

1- Limpiar datos.

2- Discretizar (solo datos numéricos).

3- Extraer reglas de asociación.

   a. Filtrar por consecuentes.

4- Definir sumarizadores.

5- Definir cuantificadores.

6- Transformar reglas en resúmenes lingüísticos.

Para poder cumplir con estos requisitos es necesario primeramente cagar los datos para luego pasar a limpiarlos de manera que no queden datos con valores nulos.

Luego se discretizan los datos lo cual consiste en convertir las variables numéricas a variables lingüísticas. Los términos lingüísticos definidos para la discretización son “alto, medio, bajo”. En los anexos del trabajo se muestra una tabla con los datos discretizados.

Después de obtener los datos discretizados se procede a obtener las reglas de asociación, para luego filtrarlas por consecuente. Para realizar este paso se emplea el algoritmo Apriori en el cual las reglas extraídas están caracterizadas por tener múltiples antecedentes y un solo consecuente.

Existen varias medidas de calidad implementadas para las reglas, en este caso se hará uso del

soporte y la confianza, ambas descritas en el capítulo 1, la medida lift (31) la cual muestra la proporción existente entre el soporte observado de un conjunto de ítems respecto del soporte teórico de ese conjunto dado el supuesto de independencia.

A continuación se filtran las reglas por consecuente para proporcionar una mejor comprensión a los resúmenes.

Un ejemplo de cómo quedarían las reglas es:

| Rules | Support | Confidence | Lift |
|---|---|---|---|
| {IRP=medio} => {Clase=Regular} | 0.353333333333333 | 0.56989247311828 | 1.095947063689 |
| {IRRH=medio} => {Clase=Regular} | 0.393333333333333 | 0.634408602150538 | 1.22001654259719 |
| {ICD=alto} => {Clase=Regular} | 0.52 | 0.549295774647887 | 1.05633802816901 |

Finalmente se presenta el paso de Sumarización Lingüística, que tiene como entrada las reglas agrupadas obtenidas del paso anterior y del que finalmente se obtendrán los resúmenes lingüísticos.

Para este paso se definen los sumarizadores y cuantificadores que no son más que:

Sumarizadores (3): par atributo-valor, definido en el concepto del atributo, ejemplo el atributo "$A_1$ *bajo*", que sería el par formado por el atributo y el valor lingüístico luego de discretizar los datos.

Cuantificador (3): es una indicación de la medida en que los datos satisfacen el resumen, se utilizan para representar la cantidad de elementos que satisfacen el predicado, en este caso los resúmenes obtenidos. Actualmente la lógica clásica se limita al uso de dos cuantificadores: *existe (∃) y para todo (∀).* Sin embargo en el lenguaje natural se usan muchos y diversos cuantificadores, por ejemplo, alrededor de 20, casi todos, pocos, muchos, la mayoría, cerca de la mitad.

Pueden emplearse dos tipos de medida de cantidad:

- Absoluta, como por ejemplo: "cerca de 20", "más o menos 100" o "varios"
- Relativa, como por ejemplo: "algunos", "la mayoría" o "casi todos"

# Capítulo II
# Descripción de la solución propuesta

La confianza obtenida en los resúmenes puede ser usada para definir los cuantificadores, ya que esta indica la probabilidad de que siendo cierto el antecedente lo sea también el consecuente. En la tabla se muestran los cuantificadores usados.

**Tabla 3: Tabla de cuantificadores**

| Valores (entre 0-1) | Cuantificador |
|---|---|
| 0,1 – 0,2 | “Pocas veces” |
| 0,4 – 0,49 | “Cerca de la mitad” |
| 0,5 – 0,79 | “Más de la mitad” |
| 0,8 – 0,9 | “La mayoría de la veces” |
| 0,98 – 1 | “Siempre” |

Ya definidos los sumarizadores y los cuantificadores se procede a transformar las reglas en resúmenes lingüísticos. Como ya se mencionó anteriormente estos resúmenes serán en forma de regla de asociación por los que los sumarizadores coinciden con los consecuentes de las mismas. En los anexos del trabajo se encuentran los resúmenes.

Ejemplo de resumen lingüístico.

**Tabla 4: Ejemplo de reglas y resumen**

| Regla | Resumen |
|---|---|
| “{$A_1$=bajo, $B_1$=medio => $C_1$=alto}” | “La mayoría de las veces, cuando el $\boldsymbol{A_1}$ es bajo y el $\boldsymbol{B_1}$ es medio; entonces el $\boldsymbol{C_1}$ es alto” |

## 2.3 Personas relacionadas con el módulo

Se relacionan con el módulo todos los usuarios que de una forma u otra necesiten realizar resúmenes lingüísticos a partir de reglas de asociación para usarlos en el proceso de toma de decisiones.

# Capítulo II
# Descripción de la solución propuesta

## 2.4 Fase de Planificación

La fase de planificación se plantea como un permanente diálogo entre la parte empresarial y técnica del proyecto, en la que los primeros decidirán el alcance, la prioridad, la composición de las versiones y la fecha de las mismas, mientras que los técnicos son los responsables de estimar la duración requerida para implementar las funcionalidades deseadas por el cliente, de informar sobre las consecuencias de determinadas decisiones, de organizar la cultura de trabajo y finalmente de realizar la planificación detallada dentro de cada versión (32). En esta fase se realizan las historias de usuarios (en lo adelante HU) para facilitar una descripción corta de lo que el sistema debe realizar, además se crea un plan de iteraciones para planificar el tiempo de duración de las tareas.

### 2.4.1 Historias de Usuario

Las HU están incluidas dentro de la fase de planificación, estas son la técnica utilizada en XP para especificar los requisitos de software. Las escriben los propios clientes, tal y como ven ellos las necesidades del sistema. Por tanto serán descripciones cortas y escritas en el lenguaje del usuario sin terminología técnica.

La información de una HU puede variar y ajustarse a las características específicas del proyecto. En este caso quedarán compuestas por:

- **Número:** número asignado a la HU.
- **Nombre de HU:** atributo que contiene el nombre de la HU.
- **Modificación de la HU:** contiene cuántas modificaciones ha sufrido la HU.
- **Programador:** persona encargada de programar la HU.
- **Iteración asignada:** iteración en la que será desarrollada la HU.
- **Prioridad:** evidencia el nivel de prioridad de la HU.
- **Puntos estimados:** contiene la estimación del tiempo de duración de la HU. Cuando el valor es 1 equivale a una semana ideal de trabajo. En la metodología XP una semana ideal de trabajo equivale a 5 días trabajando 40 horas (8 horas diarias), por lo que si el valor es 0.5 equivale a 2 días y medio trabajando; es decir 20 horas.

- **Riesgo de desarrollo:** evidencia el nivel de riesgo en caso de no realizarse la HU.
- **Descripción:** posee una breve descripción de lo que realiza la HU.
- **Observaciones:** brinda información extra que se estime necesaria para una mejor comprensión de la HU.

**Tabla 5: HU Limpiar datos**

| **Historia de usuario** | |
|---|---|
| **Número:** HU #1 | **Nombre HU:** Limpiar datos |
| **Modificación de HU:** Ninguna | |
| **Programador:** Daliana Ramos | **Iteración asignada:** primera |
| **Prioridad:** Alta | **Puntos estimados:** 2 |
| **Riesgo en desarrollo:** Muy alta | |
| **Descripción:** Da la posibilidad al usuario de limpiar los datos, la misma se realiza de varias formas como son: mediante la moda para los valores nominales y para los valores numéricos mediante la media, la mediana, el valor de correlación y omitiendo los datos nulos siempre que estos no sean significativos. | |
| **Observaciones:** | |

**Tabla 6: HU Discretizar**

| **Historia de usuario** | |
|---|---|
| **Número:** HU #2 | **Nombre HU:** Discretizar |
| **Modificación de HU:** Ninguna | |
| **Programador:** Daliana Ramos | **Iteración asignada:** primera |
| **Prioridad:** Muy alta | **Puntos estimados:** 3 |
| **Riesgo en desarrollo:** Alta | |
| **Descripción:** Permite al usuario discretizar los valores numéricos, se puede realizar por varios métodos como son: especificando los límites, mediante rangos equivalentes y a través de conjuntos difusos. | |

| |
|---|
| **Observaciones:** |

**Tabla 7: HU Extraer reglas**

| **Historia de usuario** | |
|---|---|
| **Número:** HU #3 | **Nombre HU:** Extraer reglas |
| **Modificación de HU:** Ninguna | |
| **Programador:** Daliana Ramos | **Iteración asignada:** segunda |
| **Prioridad:** Alta | **Puntos estimados:** 3 |
| **Riesgo en desarrollo:** Alta | |
| **Descripción:** Permite al usuario extraer las reglas de asociación mediante el algoritmo Apriori y luego para una mejor comprensión de los resúmenes lingüísticos se agrupan las reglas extraídas por consecuente, de manera que se tenga organizado para cada atributo presente como consecuente cuales son los atributos que influyen en este como antecedente. | |
| **Observaciones:** | |

**Tabla 8: HU Resumen lingüístico**

| **Historia de usuario** | |
|---|---|
| **Número:** HU #4 | **Nombre HU:** Resumen lingüístico |
| **Modificación de HU:** Ninguna | |
| **Programador:** Daliana Ramos | **Iteración asignada:** segunda |
| **Prioridad:** Alta | **Puntos estimados:** 4 |
| **Riesgo en desarrollo:** Alta | |
| **Descripción:** Permite la transformación de las reglas de asociación a resúmenes lingüísticos, para ello se tendrá en cuenta los sumarizadores y una matriz con los cuantificadores. | |

| Observaciones: |
|---|

### 2.4.2 Plan de iteraciones

Luego de definir las historias de usuario se prosigue a confeccionar el plan de iteraciones, el cual es una guía para la implementación del sistema ya que en él se plasma el tiempo y en qué iteración son implementadas las funcionalidades del sistema.

El módulo será desarrollado en las siguientes iteraciones:

**Primera:** En esta iteración se van a implementar las HU que tienen una prioridad muy alta.

**Segunda:** En esta iteración se van a implementar las HU que tengan prioridad alta.

**Tabla 9: Plan de Iteraciones**

| Iteración | Orden de la HU a implementar | Duración de cada HU(semana ideal) | Duración Total |
|---|---|---|---|
| 1ra | Limpiar datos | 2 | 5 |
| | Discretizar | 3 | |
| 2da | Extraer reglas | 3 | 7 |
| | Resumen lingüístico | 4 | |
| **Total(semana)** | 12 | | |

## 2.5 Fase de diseño

La metodología XP establece una serie de premisas o recomendaciones a la hora de abordar esta etapa, entre estas una es usar las tarjetas CRC (Cargo o clase, Responsabilidad y Colaboración).

### 2.5.1 Tarjetas Cargo o Clase-Responsabilidad-Colaboración (CRC)

La metodología XP no requiere la presentación del sistema mediante diagramas de clases utilizando

notación UML, en su lugar usan otras técnicas como las tarjetas CRC (Cargo o Clase, Responsabilidad y Colaborador).

Las tarjetas CRC trabajan con una metodología basada en objetos, cada tarjeta representa un objeto. El nombre de la clase se coloca a modo de título en la tarjeta, las responsabilidades se colocan a la izquierda y las clases que se implican en cada responsabilidad a la derecha (32).

El lenguaje de programación R, no es basado en la Programación Orientada a Objetos (POO), sino que es un entorno de análisis estadístico para la manipulación de datos, su cálculo y la creación de gráficos. Debido a lo anterior se pretende adaptar las tarjetas CRC de la siguiente forma: cada tarjeta CRC representa una HU, quedando las responsabilidades definidas como las funciones que realiza y los colaboradores serían los métodos que utiliza cada funcionalidad y en caso de pertenecer a algún paquete se especifica el mismo. A continuación se muestran las tarjetas CRC de la propuesta de solución:

**Tabla 10: Tarjeta CRC Limpiar datos**

| **Tarjeta CRC: Limpiar datos** | |
|---|---|
| **Descripción:** Se encarga de la limpieza de los datos. | |
| **Responsabilidades** | **Colaboradores** |
| limpiarModa() | |
| limpiarNulos() | sonSignificativos() |
| limpiarMedia() | mean() del paquete DMwR |
| limpiarMediana() | median() del paquete DMwR |
| limpiarCorrelacion() | cor() del paquete DMwR |

**Tabla 11: Tarjeta CRC Discretizar**

| **Tarjeta CRC: Discretizar** |
|---|
| **Descripción:** Se encarga de discretizar los valores numéricos. |

| Responsabilidades | Colaboradores |
|---|---|
| discretizarLimites() | cut() |
| discretizarRangosEquiv() | bin.var() del paquete Rcmdr |
| discretizarConjuntosDifusos() | fuzzifier() del paquete frbs |

**Tabla 12: Tarjeta CRC Extraer reglas**

| **Tarjeta CRC: Extraer reglas** | |
|---|---|
| **Descripción:** Se encarga de extraer las reglas de asociación y posteriormente filtrarlas por consecuente. | |
| **Responsabilidades** | **Colaboradores** |
| reglasExtraer() | apriori() del paquete arules |

**Tabla 13: Tarjeta CRC Resumen lingüístico**

| **Tarjeta CRC: Resumen lingüístico** | |
|---|---|
| **Descripción:** Se encarga de transformar las reglas de asociación a resúmenes lingüísticos. | |
| **Responsabilidades** | **Colaboradores** |
| resumenLinguistico() | |

## 2.6 Fase de desarrollo

La implementación es un pilar imprescindible en la metodología XP que se realiza durante la fase de desarrollo. A pesar de que en las fases anteriores la metodología propone claramente la generación de artefactos basados en técnicas como HU y tarjetas CRC, para esta fase propone prácticas concretas como lo son: programación en parejas, la utilización de estándares de código, no trabajar más de 40 horas semanales e integrar el código frecuentemente, pero no existe una opción única por parte de los investigadores de cuáles deben ser los artefactos que genera esta fase. Debido a la importancia de esta fase se decide realizar tareas de programación y definir un estándar de codificación.

### 2.6.1 Tareas de programación

Las tareas de programación son actividades sencillas derivadas de las historias de usuario, se plasman en tarjetas de papel donde se describe que se debe realizar. Estas son dinámicas y flexibles ya que pueden ser cambiadas o modificadas e incluso se pueden agregar nuevas tareas.

**Tabla 14: Tareas de programación**

| Historia de usuario | Tareas de programación |
|---|---|
| Limpiar datos | • Limpiar por la moda<br>• Limpiar omitiendo los valores nulos<br>• Limpiar por la media<br>• Limpiar por la mediana<br>• Limpiar por el valor de correlación |
| Discretizar | • Discretizar especificando los límites<br>• Discretizar mediante rangos equivalentes<br>• Discretizar mediante conjuntos difusos |
| Extraer reglas | • Extraer las reglas de asociación y filtrarlas por consecuente |
| Resumen lingüístico | • Transformar las reglas de asociación a resúmenes lingüísticos |

### Tareas para la HU # 1: Limpiar datos

**Tabla 15: HU #1-1 Limpiar por la moda**

| **Tarea** | |
|---|---|
| **Número tarea:** HU #1-1 | **Historia de usuario:** HU #1: Limpiar datos |
| **Nombre de la tarea:** Limpiar por la moda | |
| **Tipo de tarea:** Desarrollo | |
| **Fecha inicio:** 10/02/2014 | **Fecha fin:** 11/02/2014 |
| **Programador responsable:** Daliana Ramos García | |

| |
|---|
| **Descripción:** Se deben limpiar todos los datos nulos mediante el valor que más se repite en los casos de datos nominales. |

**Tabla 16: HU #1-2 Limpiar omitiendo los nulos**

| **Tarea** | |
|---|---|
| **Número tarea:** HU #1-2 | **Historia de usuario:** HU #1: Limpiar datos |
| **Nombre de la tarea:** Limpiar omitiendo los valores nulos | |
| **Tipo de tarea:** Desarrollo | |
| **Fecha inicio:** 12/02/2014 | **Fecha fin:** 13/02/2014 |
| **Programador responsable:** Daliana Ramos García | |
| **Descripción:** Se deben limpiar todos los datos nulos omitiéndolos siempre que estos no sean significativos y sean valores numéricos. | |

**Tabla 17: HU #1-3 Limpiar por la media**

| **Tarea** | |
|---|---|
| **Número tarea:** HU #1-3 | **Historia de usuario:** HU #1: Limpiar datos |
| **Nombre de la tarea:** Limpiar por la media | |
| **Tipo de tarea:** Desarrollo | |
| **Fecha inicio:** 14/02/2014 | **Fecha fin:** 17/02/2014 |
| **Programador responsable:** Daliana Ramos García | |
| **Descripción:** Se deben limpiar todos los datos nulos sustituyéndolos por el promedio de los datos en la columna. | |

**Tabla 18: HU # 1-4 Limpiar por la mediana**

| **Tarea** | |
|---|---|
| **Número tarea:** HU #1-4 | **Historia de usuario:** HU #1: Limpiar datos |
| **Nombre de la tarea:** Limpiar por la mediana | |

| | |
|---|---|
| **Tipo de tarea:** Desarrollo | |
| **Fecha inicio:** 18/02/2014 | **Fecha fin:** 19/02/2014 |
| **Programador responsable:** Daliana Ramos García | |
| **Descripción:** Se deben limpiar todos los datos nulos sustituyéndolos por el valor central de la columna. | |

**Tabla 19: HU # 1-5 Limpiar por el valor de correlación**

| **Tarea** | |
|---|---|
| **Número tarea:** HU #1-5 | **Historia de usuario:** HU #1: Limpiar datos |
| **Nombre de la tarea:** Limpiar por el valor de correlación | |
| **Tipo de tarea:** Desarrollo | |
| **Fecha inicio:** 20/02/2014 | **Fecha fin:** 21/02/2014 |
| **Programador responsable:** Daliana Ramos García | |
| **Descripción:** Se deben limpiar todos los datos nulos sustituyéndolos por el valor de correlación de los datos. | |

### Tareas para la HU # 2: Discretizar

**Tabla 20: HU #2-1 Discretizar especificando los límites**

| **Tarea** | |
|---|---|
| **Número tarea:** HU #2-1 | **Historia de usuario:** HU #2: Discretizar |
| **Nombre de la tarea:** Discretizar especificando los límites. | |
| **Tipo de tarea:** Desarrollo | |
| **Fecha inicio:** 24/02/2014 | **Fecha fin:** 28/02/2014 |
| **Programador responsable:** Daliana Ramos García | |
| **Descripción:** Se dividen los datos de las variables numéricas en intervalos y se codifican los valores según los intervalos en que estén. | |

Tabla 21: HU #2-2 Discretizar mediante rangos equivalentes

| Tarea | |
|---|---|
| **Número tarea:** HU #2-2 | **Historia de usuario:** HU #2: Discretizar |
| **Nombre de la tarea:** Discretizar mediante rangos equivalentes. | |
| **Tipo de tarea:** Desarrollo | |
| **Fecha inicio:** 3/03/2014 | **Fecha fin:** 7/03/2014 |
| **Programador responsable:** Daliana Ramos García | |
| **Descripción:** Factoriza una variable numéricas en intervalos de igual anchura, por la misma frecuencia o en los puntos de corte naturales. | |

Tabla 22: HU #2-3 Discretizar mediante conjuntos difusos

| Tarea | |
|---|---|
| **Número tarea:** HU #2-3 | **Historia de usuario:** HU #2: Discretizar |
| **Nombre de la tarea:** Discretizar mediante conjuntos difusos | |
| **Tipo de tarea:** Desarrollo | |
| **Fecha inicio:** 10/03/2014 | **Fecha fin:** 14/03/2014 |
| **Programador responsable:** Daliana Ramos García | |
| **Descripción:** Discretizar los valores numéricos mediante funciones lingüísticas. | |

## Tareas para la HU # 3: Extraer reglas

Tabla 23: HU #3-1 Extraer reglas de asociación y filtrarlas por consecuente

| Tarea | |
|---|---|
| **Número tarea:** HU #3-1 | **Historia de usuario:** HU #3: Extraer reglas |
| **Nombre de la tarea:** Extraer las reglas de asociación | |
| **Tipo de tarea:** Desarrollo | |

| | |
|---|---|
| **Fecha inicio:** 17/03/2014 | **Fecha fin:** 4/04/2014 |
| **Programador responsable:** Daliana Ramos García | |
| **Descripción:** Se extraen las reglas de asociación mediante el algoritmo apriori y luego se filtran por consecuente. | |

### Tareas para la HU # 4: Resumen lingüístico

**Tabla 24: HU #4-1 Transformar las reglas de asociación a resúmenes lingüísticos**

| **Tarea** | |
|---|---|
| **Número tarea:** HU #4-1 | **Historia de usuario:** HU #4: Resumen lingüístico |
| **Nombre de la Tarea:** Transformar las reglas de asociación a resúmenes lingüísticos | |
| **Tipo de Tarea:** Desarrollo | |
| **Fecha Inicio:** 8/04/2014 | **Fecha Fin:** 25/04/2014 |
| **Programador Responsable:** Daliana Ramos García | |
| **Descripción:** Se deben transformar las reglas de asociación a resúmenes lingüísticos, para esto se definen los sumarizadores y una matriz con los cuantificadores. | |

## 2.6.2 Estándares de codificación

Para conseguir que el código se encuentre en buen estado y que cualquier persona pueda modificar o usar cualquier parte del código es imprescindible que el estilo de codificación sea consistente. A continuación se muestra el estándar utilizado, especificar que es un estilo propio.

### Nombre de las variables

Todas la variables empezarán con minúscula y de ser compuestas son separadas por “.” (punto, Ej. num.fvarinput). Los nombres de las variables no deben comenzar con caracteres como “_”, “-”, “$”, además deben ser significativos.

# Capítulo II
# Descripción de la solución propuesta

**Nombre de los métodos**

Todos los métodos empezarán con minúscula y de ser compuestos la primera inicial mayúscula (Ej. limpiarModa). Los nombres de los métodos no deben comenzar con caracteres como “_”, “-”, “$”, además deben ser significativos.

**Nombres de los scripts**

Los nombres de los script deben empezar con mayúscula y de ser compuestos serán unidos (ejemplo: ResumenLinguistico).

**Comentarios de código**

Los comentarios solo se pueden hacer línea por línea comenzando siempre con el símbolo “#”. Deben contener solo la información que es relevante para la lectura y entendimiento del módulo.

## Conclusiones del capítulo

En este capítulo se arribó a las siguientes conclusiones:

- Se definió la propuesta de solución del problema, detallándola con la ayuda de los artefactos propuestos por la metodología Programación Extrema, lo cual posibilitó organizar todo el proceso de desarrollo.
- Se describió el módulo y se detallaron las funcionalidades mediante historias de usuarios, de las cuales se elaboró el plan de iteraciones para definir el momento en el que serán implementadas y cuánto durará su desarrollo.
- Se realizó una descripción de la fase de diseño donde se elaboraron las tarjetas CRC.
- Finalmente durante la fase de codificación se elaboraron las tareas de programación, derivadas de las historias de usuario y para estandarizar el código generado por el equipo de desarrollo en esta fase, se definieron los estándares de código a utilizar, obteniéndose finalmente la herramienta deseada acorde a los requisitos iniciales.

# Capítulo III
# Validación y prueba de la solución

## Capítulo III: Validación y prueba de la solución

En el presente capítulo se presenta la validación del módulo desarrollado mediante la fase de pruebas de la metodología XP y mediante el cálculo de los cinco indicadores de calidad de los resúmenes.

## 3.1 Fase de Prueba

Según (32) las unidades de test o pruebas son un pilar básico en la metodología XP. El proceso de pruebas se realiza continuamente para asegurar durante todo el proceso de desarrollo el éxito del producto que se está elaborando.

Las pruebas de aceptación o pruebas funcionales de XP están destinadas a evaluar si al final de una iteración se consiguió la funcionalidad que se esperaba y que esta esté en función de los requisitos establecidos inicialmente, estas pruebas usualmente son diseñadas por el usuario o cliente final, mientras que las pruebas unitarias son las encargadas de verificar el código y estas son diseñadas por los programadores.

El objetivo fundamental que tienen las pruebas de software, es verificar los requisitos del sistema, por lo que son los propios requisitos la principal fuente de información a la hora de construir las pruebas del sistema

### 3.1.1 Pruebas de aceptación

Las pruebas de aceptación son creadas a partir de las historias de usuario. Una historia de usuario puede tener más de una prueba de aceptación, tantas como sean necesarias para garantizar su correcto funcionamiento. Una prueba de aceptación es como una caja negra. Cada una de ellas representa una salida esperada del sistema. Es responsabilidad del cliente verificar la corrección de las pruebas de aceptación y tomar decisiones acerca de las mismas. Una historia de usuario no se considera completa hasta que no supera sus pruebas de aceptación (32).

A continuación se presentan algunos casos de pruebas de aceptación, detallando primeramente la información que contienen, estos se aplicaron a cada una de las funcionalidades del módulo.

- **Número**: número que identifica el caso de prueba.
- **Número HU**: número que identifica la HU correspondiente al caso de prueba.
- **Nombre**: nombre de la HU a la que pertenece el caso de prueba.
- **Condiciones de ejecución**: condiciones necesarias para ejecutar la prueba.
- **Entradas**: valores de entrada.
- **Resultado esperado**: salida de la ejecución

**Tabla 25: Caso de prueba #1**

| **Caso de Prueba de Aceptación** | |
|---|---|
| **Número:** 1 | **Número HU:** 1 |
| **Nombre:** Limpiar Datos | |
| **Condiciones de ejecución:** | |
| **Entradas:** Se cargan los datos a limpiar, luego se especifica la posición de la columna a limpiar y para los casos de limpiar omitiendo los nulos se especifica además el porciento de datos nulos significativos y para limpiar sustituyendo por el valor de correlación la columna con la que se va a medir el nivel de correlación. | |
| **Resultados esperados:** Datos limpios, ya sea sustituyendo los valores nulos por la moda en casos nominales así como sustituyéndolos por la media, mediana, el valor de correlación y omitiéndolos en valores numéricos. | |

**Tabla 26: Caso de prueba #2**

| **Caso de Prueba de Aceptación** | |
|---|---|
| **Número:** 2 | **Número HU:** 2 |
| **Nombre:** Discretizar | |
| **Condiciones de ejecución:** Los datos deben estar previamente limpios y deben ser numéricos. | |
| **Entradas:** Se cargan los datos, luego se especifica la posición de la columna a discretizar y para el caso de discretizar mediante conjuntos difusos además se especifica el números de variables de entrada, el nombre de las etiquetas de las variables de entrada y una matriz que contiene los parámetros para formar las funciones de pertenencia. | |

| |
|---|
| **Resultados esperados:** Datos discretizados, ya sea mediante rangos equivalentes, especificando los límites o a través de conjuntos difusos. |

**Tabla 27: Caso de prueba #3**

| **Caso de Prueba de Aceptación** | |
|---|---|
| **Número:** 3 | **Número HU:** 3 |
| **Nombre:** Extraer Reglas | |
| **Condiciones de ejecución:** Los datos deben estar previamente discretizados. | |
| **Entradas:** Se cargan los datos, luego se especifica el soporte y confianza mínimo definido por el usuario y el consecuente por el cual se quieren filtrar las reglas. | |
| **Resultados esperados:** Reglas de asociación filtradas por consecuente con soporte y confianza mayor o igual que las definidas por el usuario | |

**Tabla 28: Caso de prueba #4**

| **Caso de Prueba de Aceptación** | |
|---|---|
| **Número:** 4 | **Número HU:** 4 |
| **Nombre:** Resumen lingüístico | |
| **Condiciones de ejecución:** Reglas filtradas por consecuente. | |
| **Entradas:** Se cargan las reglas, luego se definen los cuantificadores y artículos a usar en los resúmenes. . | |
| **Resultados esperados:** Resúmenes lingüísticos de los datos. | |

### 3.1.1.1 Resultados de las pruebas

Para realizar las pruebas de aceptación se tuvieron en cuenta tres iteraciones en las cuales se detectaron como errores principales:

- Errores en la entrada de datos.
- Errores en la limpieza de los datos.

- Errores en la discretización.
- Errores de validación.

En la primera iteración realizada se encontraron 5 no conformidades y se solucionaron, en la segunda iteración se encontraron 3 no conformidades, las cuales fueron solucionadas, finalmente en la tercera iteración no se encontraron no conformidades viéndose ya erradicados los errores de las iteraciones anteriores.

**Tabla 29: Resultado de las pruebas de aceptación**

| Sistema | HU | Iteración | NC | Cerrada | No procede |
|---|---|---|---|---|---|
| Módulo para la sumarización lingüística en R | 4 | 1ra | 5 | 5 | 0 |
| | | 2da | 3 | 3 | 0 |
| | | 3ra | 0 | 0 | 0 |

### 3.1.2 Pruebas unitarias

Comprueban el comportamiento de cada una de las funcionalidades implementadas de forma independiente y se deben realizar una vez que sea implementada la funcionalidad. Este tipo de prueba permite producir un código de mayor calidad, detectar errores cuando se programan nuevas funcionalidades o se realizan cambios en el código. Sirven como pequeña fuente de documentación sobre qué es lo que se espera que haga el código y obliga a los desarrolladores a escribir el código en pequeñas porciones con el fin de que puedan ser probadas independientemente. En el desarrollo del módulo se utilizaron este tipo de pruebas las cuales se realizaron en el proceso de escritura del código, para ello se usó el paquete RUnit, el cual está destinado a este fin.

Las siguientes imágenes muestran el resultado de las pruebas. En los anexos se pueden encontrar las demás pruebas.

# Capítulo III
# Validación y prueba de la solución

**Imagen de pruebas unitarias de limpiar por la moda**

Resultado de pasar por parámetro los datos nominales.

Function: **limpiarModa** Runs: 1

| line | code | calls | time |
|---|---|---|---|
| 1 | { | 0 | 0 |
| 2 | if (!is.numeric(datos[, pos])) { | 1 | 0 |
| 3 | tabla <- table(datos[, pos]) | 1 | 0 |
| 4 | m = names(tabla)[tabla == max(tabla)] | 1 | 0 |
| 5 | for (j in 1:length(datos[, pos])) { | 1 | 0 |
| 6 | if (datos[j, pos] == "") { | 27 | 0.01 |
| 7 | datos[j, pos] <- m | 3 | 0 |
| 8 | } | 0 | 0 |
| 9 | } | 0 | 0 |
| 10 | } | 0 | 0 |
| 11 | else { | 0 | 0 |
| 12 | print("Los datos deben ser nominales") | 0 | 0 |
| 13 | } | 0 | 0 |
| 14 | return(as.data.frame(datos)) | 1 | 0 |
| 15 | } | 0 | 0 |

Resultado de pasar por parámetro los datos numéricos

Function: **limpiarModa** Runs: 1

| line | code | calls | time |
|---|---|---|---|
| 1 | { | 0 | 0 |
| 2 | if (!is.numeric(datos[, pos])) { | 1 | 0 |
| 3 | tabla <- table(datos[, pos]) | 0 | 0 |
| 4 | m = names(tabla)[tabla == max(tabla)] | 0 | 0 |
| 5 | for (j in 1:length(datos[, pos])) { | 0 | 0 |
| 6 | if (datos[j, pos] == "") { | 0 | 0 |
| 7 | datos[j, pos] <- m | 0 | 0 |
| 8 | } | 0 | 0 |
| 9 | } | 0 | 0 |
| 10 | } | 0 | 0 |
| 11 | else { | 0 | 0 |
| 12 | print("Los datos deben ser nominales") | 1 | 0 |
| 13 | } | 0 | 0 |
| 14 | return(as.data.frame(datos)) | 1 | 0 |
| 15 | } | 0 | 0 |

### 3.1.1 Indicadores de calidad para resúmenes lingüísticos

Para validar los resúmenes se tuvieron en cuenta los indicadores propuestos en el capítulo 1 (validez, generalidad, utilidad, novedad, simplicidad), los cuales se usan como una prueba más para medir la calidad de los resúmenes obtenidos.

En (16) se hace una adaptación a estas características proponiéndose fórmulas matemáticas para su cálculo. Las salidas de este cálculo corresponden a valores entre 0 y 1, donde el valor más cercano a 1 significa un mejor resultado (3).

Un ejemplo de este resultado se muestra a continuación. En los anexos del trabajo se encuentran los demás resultados de los indicadores del archivo utilizado como prueba.

**Tabla de indicadores para la Clase**

| Identificador | Validez | Generalidad | Utilidad | Novedad | Simplicidad |
|---|---|---|---|---|---|
| R1 | 0,569892473 | 0,353333333 | 0,353333333 | 0,569892473 | 1 |
| R2 | 0,634408602 | 0,393333333 | 0,393333333 | 0,606666667 | 1 |
| R3 | 0,549295775 | 0,52 | 0,52 | 0,48 | 1 |
| R4 | 0,523489933 | 0,52 | 0,52 | 0,48 | 1 |
| R5 | 0,52 | 0,52 | 0,52 | 0,48 | 1 |
| R6 | 0,569892473 | 0,353333333 | 0,353333333 | 0,569892473 | 0,5 |
| R7 | 0,569892473 | 0,353333333 | 0,353333333 | 0,569892473 | 0,5 |
| R8 | 0,569892473 | 0,353333333 | 0,353333333 | 0,569892473 | 0,5 |
| R9 | 0,634408602 | 0,393333333 | 0,393333333 | 0,606666667 | 0,5 |
| R10 | 0,634408602 | 0,393333333 | 0,393333333 | 0,606666667 | 0,5 |
| R11 | 0,634408602 | 0,393333333 | 0,393333333 | 0,606666667 | 0,5 |
| R12 | 0,553191489 | 0,52 | 0,52 | 0,48 | 0,5 |
| R13 | 0,549295775 | 0,52 | 0,52 | 0,48 | 0,5 |
| R14 | 0,523489933 | 0,52 | 0,52 | 0,48 | 0,5 |
| R15 | 0,569892473 | 0,353333333 | 0,353333333 | 0,569892473 | 0,25 |
| R16 | 0,569892473 | 0,353333333 | 0,353333333 | 0,569892473 | 0,25 |
| R17 | 0,569892473 | 0,353333333 | 0,353333333 | 0,569892473 | 0,25 |
| R18 | 0,634408602 | 0,393333333 | 0,393333333 | 0,606666667 | 0,25 |
| R19 | 0,634408602 | 0,393333333 | 0,393333333 | 0,606666667 | 0,25 |

| | | | | | |
|---|---|---|---|---|---|
| R20 | 0,634408602 | 0,393333333 | 0,393333333 | 0,606666667 | 0,25 |
| R21 | 0,553191489 | 0,52 | 0,52 | 0,48 | 0,25 |
| R22 | 0,569892473 | 0,353333333 | 0,353333333 | 0,569892473 | 0,125 |
| R23 | 0,634408602 | 0,393333333 | 0,393333333 | 0,606666667 | 0,125 |

La media del indicador de validez para los resúmenes se mantuvo de 0,48 a 1

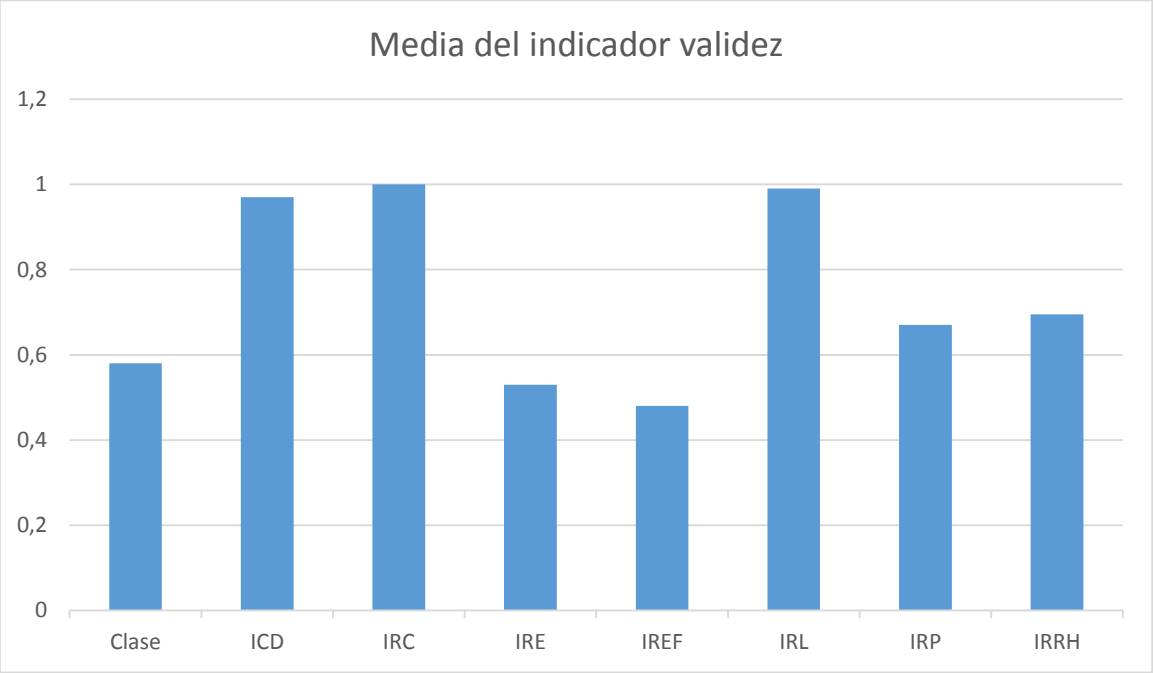


**Ilustración 3: Indicador validez**

La media del indicador de generalidad para los resúmenes se mantuvo por debajo de 0,5

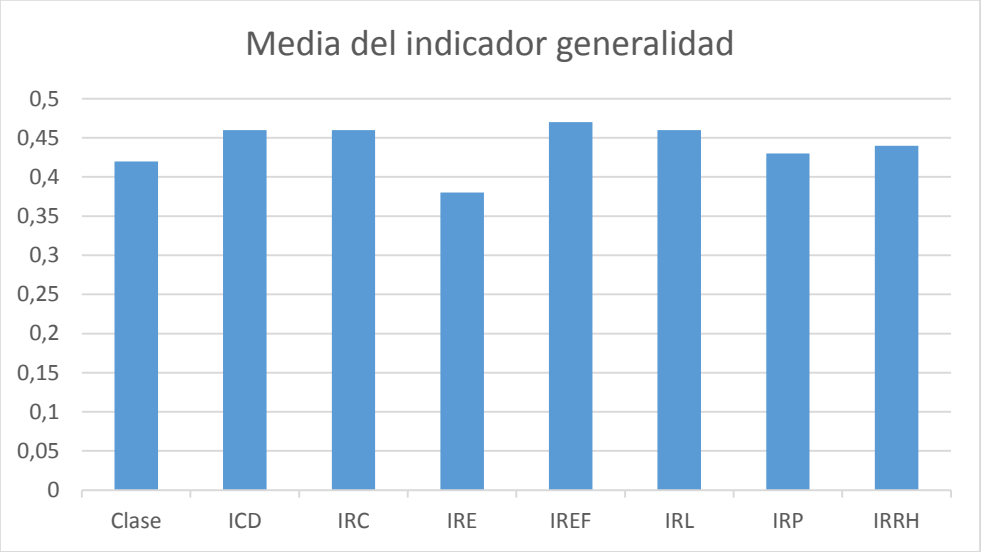


**Ilustración 4: Indicador generalidad**

Equivalentemente a la generalidad, la media del indicador de utilidad para los resúmenes se mantuvo por debajo de 0,5

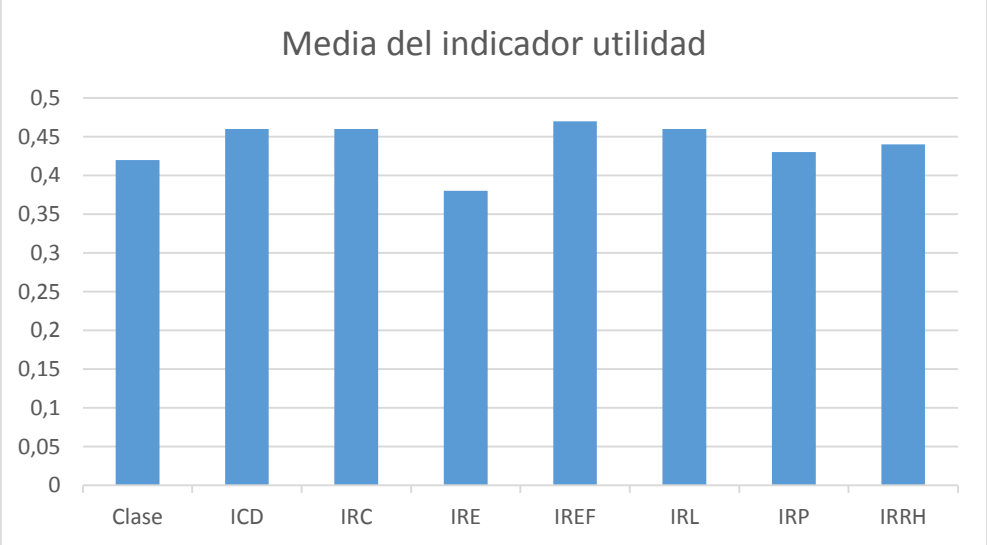


**Ilustración 5: Indicador utilidad**

La media del indicador de novedad para los resúmenes se mantuvo por encima de 0,5.

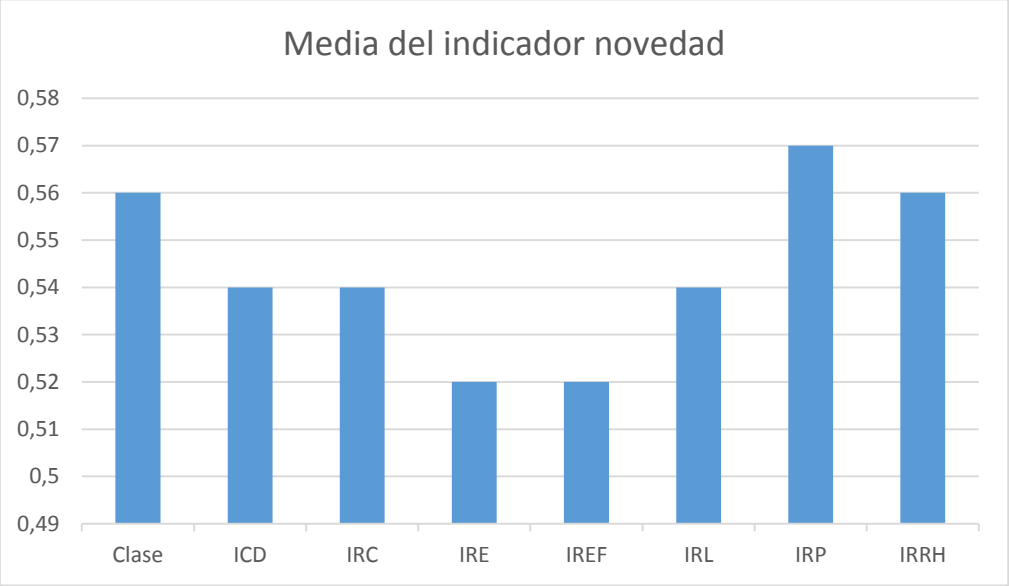


**Ilustración 6: Indicador novedad**

La media del indicador de simplicidad para los resúmenes se mantuvo por entre 0,5 y 0,7

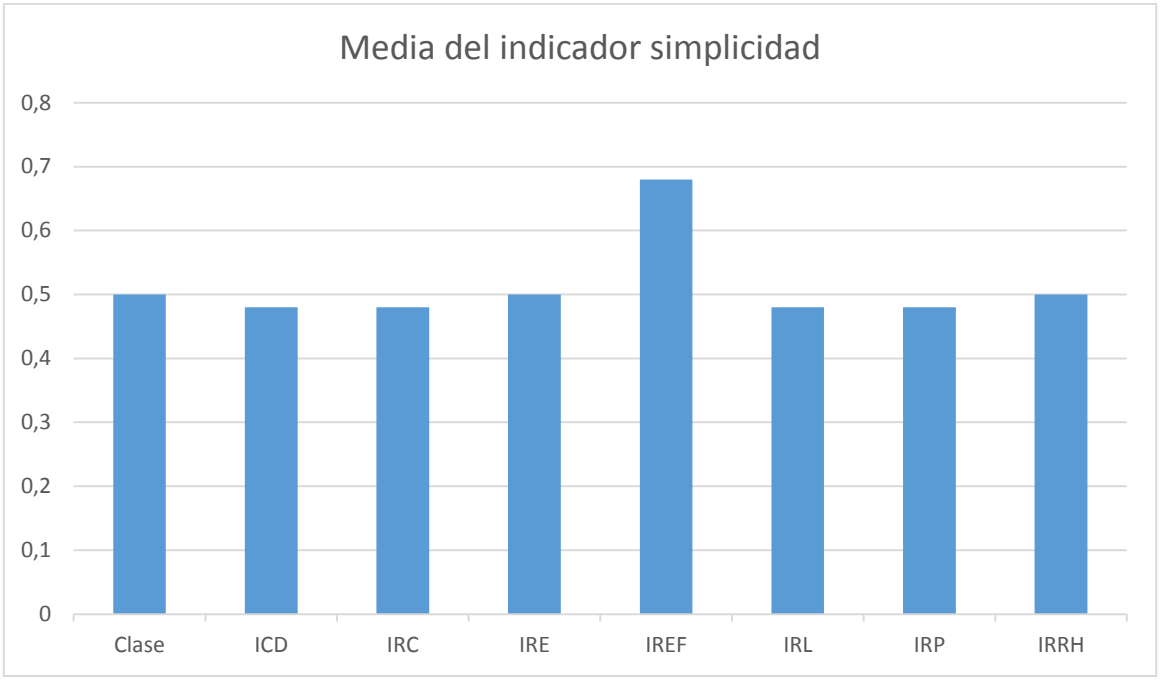


**Ilustración 7: Indicador simplicidad**

Como se muestra en las imágenes se puede observar que los resúmenes cumplieron con el rango propuesto (mantenerse en un rango de 0 a 1).

## Conclusiones del capítulo

Del presente capítulo se concluye que:

- Se ejecutaron las pruebas unitarias y de aceptación que son las que propone la metodología XP, para comprobar que se han alcanzado los objetivos propuestos para este módulo.
- Las pruebas unitarias realizadas al módulo posibilitaron la detección de errores, arrojando finalmente resultados satisfactorios luego de corregirlos.
- Las pruebas de aceptación realizadas arrojaron como resultado en la última iteración correspondencia entre las funcionalidades desarrolladas y los requisitos funcionales antes descritos.

- Finalmente se evaluó la calidad de los resúmenes lingüísticos a través de los indicadores de calidad.

# Conclusiones generales

## Conclusiones Generales

Mediante el desarrollo de este trabajo se cumplieron todos los objetivos de esta investigación de manera satisfactoria y a la vez se llegó a las siguientes conclusiones:

- Los métodos científicos empleados para investigar el objeto de estudio posibilitaron identificar las teorías y los conceptos que sustentan la presente investigación.
- Aplicando la metodología de software, la herramienta seleccionada y el lenguaje de programación se desarrolló un módulo para realizar resúmenes lingüísticos a los datos históricos de los proyectos terminados en los centros de la Universidad de las Ciencias Informáticas, el cual facilita la toma de decisiones en la gestión de proyectos.
- El módulo cuenta con cuatro funcionalidades principales las cuales permiten el tratamiento de datos tanto cualitativos como cuantitativos.
- La validación funcional a través pruebas realizadas al módulo, usando un conjunto de datos de varios proyectos de la universidad, demostró el correcto funcionamiento de la solución.

## Recomendaciones

Luego de concluida la solución propuesta se propone como recomendaciones:

- Extender el paso: Definición de los cuantificadores, de manera que estos se calculen de forma automática.
- Implementar una funcionalidad que permita unificar los resúmenes.

## Bibliografía

1. **L. Torralbo, J. Alfonso.** Marco de Descubrimiento de Conocimiento para Datos Estructuralmente Complejos con énfasis en el Análisis de Eventos en Series. [En línea] 2010. http://oa.upm.es/5729.

2. **González García, Pedro.** *Aprendizaje Evolutivo de Reglas Difusas para Descripción de Subgrupos. Tesis Doctoral.* Granada : Univerdidad de Granada, 2007.

3. **Trinchet, Eric Eduardo Piñera.** *Algoritmo de sumarización lingüística como apoyo a la toma de decisiones en gestión de proyecto.* La Habana : s.n., 2013. Tesis de Maestría.

4. **Zadeh, L.A.** *The Concept of a Linguistic Variable and its Application to Approximate Reasoning-I, Information Sciences.* 1975.

5. **Palenzuela, Ing. Filiberto López.** *Modelo para la toma de decisiones en los proyectos de software basado en lo criterios de expertos.* Ciudad de La Habana : s.n., 2013.

6. **Tamayo, Ing. Karina Sánchez.** Método para evaluar proyectos informáticos y establecer un orden de prioridad que ayude a la toma de decisiones. MEPROI. *Tesis de Maestría.* La Habana : s.n., 2010.

7. **J.M Molina, J. García.** *TÉCNICAS DE ANÁLISIS DE DATOS APLICACIONES PRÁCTICAS UTILIZANDO MICROSOFT EXCEL Y WEKA.* 2006.

8. **J. Hernández, J. Ramírez y C. Ferri.** *Introducción a la Minería de Datos.* Departamento de Sistemas Informáticos y Computación, Universidad Politécnica de Valencia. Madrid : PEARSON EDUCACIÓN, S.A, 2004. ISBN 84-205-4091-9.

9. **Perversi, Ignacio.** *APLICACIÓN DE MINERÍA DE DATOS PARA LA EXPLORACIÓN Y DETECCIÓN DE PATRONES DELICTIVOS EN ARGENTINA.* Instituto Tecnológico de Buenos Aires. 2007.

10. *Herramientas de Minería de Datos.* **Yuniet Rodríguez Suárez, Anolandy Díaz Amador.** No. 3-4, La Habana : s.n., 2009, RCCI, Vol. 3, págs. 73-80.

11. **M.A. Vila, D. Sánchez, L. Cerda Leiva.** REGLAS DE ASOCIACIÓN APLICADAS A LA DETECCIÓN DE FRAUDE CON TARJETAS DE CRÉDITOS.

12. **Dante Conti, Fco. Javier Martínez.** *Reglas de Asociación en Series Temporales: panorama referencial y tendencias.*

13. **J. Kacprzyk, A. Wilbik, S. Zadrozny.** Linguistic Summarization of Trends: A Fuzzy Logic Based Approach. Polonia : s.n.

# Bibliografía


14. **A. van der Heide, G. Triviño.** *Automatically generated linguistic summaries of energy consumption data.* Mires,España : Centro Europeo de Soft Computing.

15. **A. Niewiadomski, J. Ochelska y P. S. Szczepaniak.** Interval-valued linguistic summaries of databases. Polonia : s.n., 2006. Vol. 35, No. 2.

16. **D. Wu, J. M. Mendel, J. Joo.** *Linguistic Summarization Using IF-THEN Rules.* Department of Electrical Engineering, IEEE Signal and Image Processing Institute. Los Angeles : Universidad del Sur de California. CA 90089-2564.

17. **Wagenknecht, Ildar Batyrshin and Michael.** TOWARDS A LINGUISTIC DESCRIPTION OF DEPENDENCIES IN DATA. Int. J. Appl. Math. Comput. Sci. 2002. Vol. 12, no. 3, págs. 391-401.

18. **CHEN, Chun-Hao, HONG, Tzung-Pei and TSENG, Vincent S.** Fuzzy data mining for time-series data. Appl. Soft Comput. 2012. Vol. 12, no. 1.

19. **M.E DE MOYA, J.E RODRÍGUEZ.** *LA CONTRIBUCIÓN DE LAS REGLAS DE ASOCIACIÓN A LA MINERÍA DE DATOS.* 2003.

20. **A.M. Palacios, J. Alcalá-Fdez.** *Extracción de reglas de asociación difusas a partir de datos de baja calidad.* 2012.

21. **Chung, John D. Holt and Soon M.** *Efficient Mining of Association Rules in Text Databases.* Department of Computer Science and Engineering, Wright State University Dayton. Ohio, USA : s.n.

22. **MSc. Raudel Hernández León, Dr. C. José Hernández Palancar, Dr. C. Jesus A. Carracso Ochoa, Dr. C. José Fco. Martínez Trinidad.** *Descubrimiento de conjuntos frecuentes de ítems en datos estáticos y dinámicos.* Ciudad de La Habana : s.n., 2010. ISSN 2072-6260.

23. **J.M. Contreras, E. Molina, P. Arteaga.** *INTRODUCCIÓN A LA PROGRAMACIÓN ESTADÍSTICA CON R PARA PROFESORES.* ISBN: 978-84-693-4859-8.

24. **The R Project for Statistical Computing.** [En línea] 2013. http://www.r-project.org/.

**25. Matloff, Norman.** *The Art of R Programming.* 2009.

**26. KDNUGGETS.COM. *Kdnuggets.*** [En línea] 2013. http://www.kdnuggets.com.

**27. INTECO, Laboratorio Nacional de Calidad del Software.** *INGENIERÍA DEL SOFTWARE: METODOLOGÍAS Y CICLOS DE VIDA.* España : s.n., 2009.

**28. Pressman, R. S.** *Ingeniería del Software. Un enfoque práctico.* 2002. pág. 37. Vol. 1.

**29. Roberth G. Figueroa, Camilo J. Solís, Armando A. Cabrera.** *METODOLOGÍAS TRADICIONALES VS. METODOLOGÍAS ÁGILES.*

# Bibliografía

**30. Patricio Letelier, M. Carmen Penadés.** *Metodologías ágiles para el desarrollo de software: eXtreme Programming(XP).*

**31. Monteserin, Dr. Ariel.** *Reglas de asociación. 2013.*

**32. Escribano, Gerardo Fernández.** *Ingeniería del Software II. Introducción a Extreme Programming.* 2002.

**33. Saeedi, Amirali.** *A Computer-Assisted Qualitative Data Analysis Framework for the Engineering.* Oregon : Oregon State University, 2010.

**34. R. Agarwal, R. Srikant.** *Fast Algorithms for Mining Association Rules.*

**35. Olabe, Xabier Basogain.** *REDES NEURONALES ARTIFICIALES Y SUS APLICACIONES.* Dpto. Ingeniería de Sistemas y Automática, Escuela Superior de Ingeniería de Bilbao, EHU.

**36. Larose, Daniel T.** *Discovering knowledge in data : an introduction to data mining.* EE.UU : A JOHN WILEY & SONS, INC., PUBLICATION, 2005.

**37. García, José Alejandro Lugo.** *MODELO PARA EL CONTROL DE LA EJECUCIÓN DE PROYECTOS BASADO EN INDICADORES Y LÓGICA BORROSA.* Tesis de Maestría.

**38. Fabiola, Ticona Condori Shirley.** ***Metodologías Tradicionales,*** *Metodologías Ágiles, Metodologías para Juegos, Metodologías Educativas y Metodologías para Aplicaciones Móviles.*

**39. Julio Antonio Hernández Pérez, Husseyn Despaigne Reyes.** *Herramienta informática de Minería de Uso de la Web sobre los registros de navegación por Internet. Implementación del módulo para realizar la tarea descriptiva Reglas de Asociación. .* La Habana : s.n., 2011.

## Anexos

## Anexo #1 Indicadores discretizados

| IRE | IRP | IREF | IRRH | ICD | IRL | IRC | Clase |
|---|---|---|---|---|---|---|---|
| bajo | medio | bajo | alto | alto | bajo | bajo | Regular |
| alto | medio | medio | alto | alto | bajo | bajo | Regular |
| bajo | medio | alto | medio | alto | bajo | bajo | Regular |
| alto | medio | alto | medio | alto | bajo | bajo | Bien |
| bajo | bajo | bajo | bajo | alto | bajo | bajo | Mal |
| alto | medio | bajo | medio | alto | bajo | bajo | Regular |
| bajo | bajo | bajo | medio | alto | bajo | bajo | Mal |
| medio | bajo | medio | medio | alto | bajo | bajo | Regular |
| alto | medio | medio | medio | alto | bajo | bajo | Regular |
| alto | medio | medio | medio | alto | bajo | bajo | Regular |
| alto | bajo | bajo | bajo | alto | bajo | bajo | Regular |
| medio | bajo | medio | medio | alto | bajo | bajo | Mal |
| bajo | medio | medio | medio | alto | bajo | bajo | Mal |
| medio | bajo | bajo | medio | alto | bajo | bajo | Mal |
| alto | medio | bajo | alto | alto | bajo | bajo | Regular |
| bajo | medio | medio | medio | alto | bajo | bajo | Mal |
| bajo | medio | alto | medio | alto | bajo | bajo | Regular |
| alto | medio | medio | medio | alto | bajo | bajo | Regular |
| bajo | medio | medio | medio | alto | bajo | bajo | Mal |
| medio | bajo | bajo | medio | alto | bajo | bajo | Regular |
| bajo | medio | bajo | medio | alto | bajo | bajo | Mal |
| bajo | medio | medio | medio | alto | bajo | bajo | Mal |
| alto | medio | bajo | medio | alto | bajo | bajo | Regular |
| alto | bajo | bajo | bajo | alto | bajo | bajo | Regular |
| medio | bajo | medio | medio | alto | bajo | bajo | Regular |
| alto | medio | bajo | bajo | alto | bajo | bajo | Regular |
| medio | medio | medio | medio | alto | bajo | bajo | Regular |
| bajo | bajo | bajo | bajo | bajo | bajo | bajo | Mal |
| bajo | bajo | bajo | medio | alto | bajo | bajo | Mal |
| alto | bajo | medio | medio | alto | bajo | bajo | Regular |
| alto | bajo | bajo | medio | alto | bajo | bajo | Regular |
| alto | medio | medio | alto | alto | bajo | bajo | Bien |

## Anexo #2 Resumen de la clase del proyecto

| Identificador | Resúmenes de la clase |
|---|---|
| R1 | "Más de la mitad de las veces, cuando el IRP es medio; entonces la Clase es Regular." |
| R2 | "Más de la mitad de las veces, cuando el IRRH es medio; entonces la Clase es Regular." |
| R3 | "Más de la mitad de las veces, cuando el ICD es alto; entonces la Clase es Regular." |
| R4 | "Más de la mitad de las veces, cuando el IRL es bajo; entonces la Clase es Regular." |
| R5 | "Más de la mitad de las veces, cuando el IRC es bajo; entonces la Clase es Regular." |
| R6 | "Más de la mitad de las veces, cuando el IRP es medio y el ICD es alto; entonces la Clase es Regular." |
| R7 | "Más de la mitad de las veces, cuando el IRP es medio y el IRL es bajo; entonces la Clase es Regular." |
| R8 | "Más de la mitad de las veces, cuando el IRP es medio y el IRC es bajo; entonces la Clase es Regular." |
| R9 | "Más de la mitad de las veces, cuando el IRRH es medio y el ICD es alto; entonces la Clase es Regular." |
| R10 | "Más de la mitad de las veces, cuando el IRRH es medio y el IRL es bajo; entonces la Clase es Regular." |
| R11 | "Más de la mitad de las veces, cuando el IRRH es medio y el IRC es bajo; entonces la Clase es Regular." |
| R12 | "Más de la mitad de las veces, cuando el ICD es alto y el IRL es bajo; entonces la Clase es Regular." |
| R13 | "Más de la mitad de las veces, cuando el ICD es alto y el IRC es bajo; entonces la Clase es Regular." |
| R14 | "Más de la mitad de las veces, cuando el IRL es bajo y el IRC es bajo; entonces la Clase es Regular." |
| R15 | "Más de la mitad de las veces, cuando el IRP es medio y el ICD es alto y el IRL es bajo; entonces la Clase es Regular." |

## Anexo #3 Resumen del indicador ICD

| Identificador | Resúmenes del indicador ICD |
|---|---|
| R1 | "La mayoría de las veces, cuando el IRP es bajo; entonces el ICD es alto." |
| R2 | "Siempre que, el IREF es medio; entonces el ICD es alto." |
| R3 | "Siempre que, el IRE es alto; entonces el ICD es alto." |
| R4 | "La mayoría de las veces, cuando el IREF es bajo; entonces el ICD es alto." |
| R5 | "La mayoría de las veces, cuando el IRE es bajo; entonces el ICD es alto." |
| R6 | "Siempre que, la Clase es Regular; entonces el ICD es alto." |

| | |
|---|---|
| R7 | "Siempre que, el IRP es medio; entonces el ICD es alto." |
| R8 | "Siempre que, el IRRH es medio; entonces el ICD es alto." |
| R9 | "Siempre que, el IRL es bajo; entonces el ICD es alto." |
| R10 | "Siempre que, el IRC es bajo; entonces el ICD es alto." |
| R11 | "La mayoría de las veces, cuando el IRP es bajo y el IRL es bajo; entonces el ICD es alto." |
| R12 | "La mayoría de las veces, cuando el IRP es bajo y el IRC es bajo; entonces el ICD es alto." |
| R13 | "Siempre que, el IREF es medio y el IRL es bajo; entonces el ICD es alto." |
| R14 | "Siempre que, el IREF es medio y el IRC es bajo; entonces el ICD es alto." |
| R15 | "Siempre que, el IRE es alto y el IRL es bajo; entonces el ICD es alto." |

## Anexo #4 Resumen del indicador IRC

| Identificador | Resúmenes del indicador IRC |
|---|---|
| R1 | "Siempre que, el IRP es bajo; entonces el IRC es bajo." |
| R2 | "Siempre que, el IREF es medio; entonces el IRC es bajo." |
| R3 | "Siempre que, el IRE es alto; entonces el IRC es bajo." |
| R4 | "Siempre que, el IREF es bajo; entonces el IRC es bajo." |
| R5 | "Siempre que, el IRE es bajo; entonces el IRC es bajo." |
| R6 | "Siempre que, la Clase es Regular; entonces el IRC es bajo." |
| R7 | "Siempre que, el IRP es medio; entonces el IRC es bajo." |
| R8 | "Siempre que, el IRRH es medio; entonces el IRC es bajo." |
| R9 | "Siempre que, el ICD es alto; entonces el IRC es bajo." |
| R10 | "Siempre que, el IRL es bajo; entonces el IRC es bajo." |
| R11 | "Siempre que, el IRP es bajo y el ICD es alto; entonces el IRC es bajo." |
| R12 | "Siempre que, el IRP es bajo y el IRL es bajo; entonces el IRC es bajo." |
| R13 | "Siempre que, el IREF es medio y el ICD es alto; entonces el IRC es bajo." |
| R14 | "Siempre que, el IREF es medio y el IRL es bajo; entonces el IRC es bajo." |
| R15 | "Siempre que, el IRE es alto y el ICD es alto; entonces el IRC es bajo." |

## Anexo #5 Resumen del indicador IRE

| Identificador | Resúmenes del indicador IRE |
|---|---|
| R1 | "Más de la mitad de las veces, cuando el IRP es medio; entonces el IRE es bajo." |
| R2 | "Más de la mitad de las veces, cuando el IRRH es medio; entonces el IRE es bajo." |
| R3 | "Más de la mitad de las veces, cuando el ICD es alto; entonces el IRE es bajo." |
| R4 | "Más de la mitad de las veces, cuando el IRL es bajo; entonces el IRE es bajo." |
| R5 | "Más de la mitad de las veces, cuando el IRC es bajo; entonces el IRE es bajo." |
| R6 | "Más de la mitad de las veces, cuando el IRP es medio y el ICD es alto; entonces el IRE es |

| | |
|---|---|
| | bajo." |
| R7 | "Más de la mitad de las veces, cuando el IRP es medio y el IRL es bajo; entonces el IRE es bajo." |
| R8 | "Más de la mitad de las veces, cuando el IRP es medio y el IRC es bajo; entonces el IRE es bajo." |
| R9 | "Más de la mitad de las veces, cuando el IRRH es medio y el ICD es alto; entonces el IRE es bajo." |
| R10 | "Más de la mitad de las veces, cuando el IRRH es medio y el IRL es bajo; entonces el IRE es bajo." |
| R11 | "Más de la mitad de las veces, cuando el IRRH es medio y el IRC es bajo; entonces el IRE es bajo." |
| R12 | "Más de la mitad de las veces, cuando el ICD es alto y el IRL es bajo; entonces el IRE es bajo." |
| R13 | "Más de la mitad de las veces, cuando el ICD es alto y el IRC es bajo; entonces el IRE es bajo." |
| R14 | "Más de la mitad de las veces, cuando el IRL es bajo y el IRC es bajo; entonces el IRE es bajo." |
| R15 | "Más de la mitad de las veces, cuando el IRP es medio y el ICD es alto y el IRL es bajo; entonces el IRE es bajo." |

## Anexo #6 Resumen del indicador IREF

| Identificador | Resúmenes del indicador IREF |
|---|---|
| R1 | "Cerca de la mitad, cuando el ICD es alto; entonces el IREF es bajo." |
| R2 | "Cerca de la mitad, cuando el IRL es bajo; entonces el IREF es bajo." |
| R3 | "Más de la mitad de las veces, cuando el IRC es bajo; entonces el IREF es bajo." |
| R4 | "Cerca de la mitad, cuando el ICD es alto y el IRL es bajo; entonces el IREF es bajo." |
| R5 | "Cerca de la mitad, cuando el ICD es alto y el IRC es bajo; entonces el IREF es bajo." |
| R6 | "Cerca de la mitad, cuando el IRL es bajo y el IRC es bajo; entonces el IREF es bajo." |
| R7 | "Cerca de la mitad, cuando el ICD es alto y el IRL es bajo y el IRC es bajo; entonces el IREF es bajo." |

## Anexo #7 Resumen del indicador IRL

| Identificador | Resúmenes del indicador IRL |
|---|---|
| R1 | "Siempre que, el IRP es bajo; entonces el IRL es bajo." |
| R2 | "Siempre que, el IREF es medio; entonces el IRL es bajo." |
| R3 | "Siempre que, el IRE es alto; entonces el IRL es bajo." |

| | |
|---|---|
| R4 | "Siempre que, el IREF es bajo; entonces el IRL es bajo." |
| R5 | "Siempre que, el IRE es bajo; entonces el IRL es bajo." |
| R6 | "Siempre que, la Clase es Regular; entonces el IRL es bajo." |
| R7 | "Siempre que, el IRP es medio; entonces el IRL es bajo." |
| R8 | "Siempre que, el IRRH es medio; entonces el IRL es bajo." |
| R9 | "Siempre que, el ICD es alto; entonces el IRL es bajo." |
| R10 | "Siempre que, el IRC es bajo; entonces el IRL es bajo." |
| R11 | "Siempre que, el IRP es bajo y el ICD es alto; entonces el IRL es bajo." |
| R12 | "Siempre que, el IRP es bajo y el IRC es bajo; entonces el IRL es bajo." |
| R13 | "Siempre que, el IREF es medio y el ICD es alto; entonces el IRL es bajo." |
| R14 | "Siempre que, el IREF es medio y el IRC es bajo; entonces el IRL es bajo." |
| R15 | "Siempre que, el IRE es alto y el ICD es alto; entonces el IRL es bajo." |

## Anexo #8 Resumen del indicador IRP

| Identificador | Resúmenes del indicador IRP |
|---|---|
| R1 | "Más de la mitad de las veces, cuando el IRE es bajo; entonces el IRP es medio." |
| R2 | "Más de la mitad de las veces, cuando la Clase es Regular; entonces el IRP es medio." |
| R3 | "Más de la mitad de las veces, cuando el IRRH es medio; entonces el IRP es medio." |
| R4 | "Más de la mitad de las veces, cuando el ICD es alto; entonces el IRP es medio." |
| R5 | "Más de la mitad de las veces, cuando el IRL es bajo; entonces el IRP es medio." |
| R6 | "Más de la mitad de las veces, cuando el IRC es bajo; entonces el IRP es medio." |
| R7 | "Más de la mitad de las veces, cuando el IRE es bajo y el ICD es alto; entonces el IRP es medio." |
| R8 | "Más de la mitad de las veces, cuando el IRE es bajo y el IRL es bajo; entonces el IRP es medio." |
| R9 | "Más de la mitad de las veces, cuando el IRE es bajo y el IRC es bajo; entonces el IRP es medio." |
| R10 | "Más de la mitad de las veces, cuando el ICD es alto y la Clase es Regular; entonces el IRP es medio." |
| R11 | "Más de la mitad de las veces, cuando el IRL es bajo y la Clase es Regular; entonces el IRP es medio." |
| R12 | "Más de la mitad de las veces, cuando el IRC es bajo y la Clase es Regular; entonces el IRP es medio." |
| R13 | "Más de la mitad de las veces, cuando el IRRH es medio y el ICD es alto; entonces el IRP es medio." |
| R14 | "Más de la mitad de las veces, cuando el IRRH es medio y el IRL es bajo; entonces el IRP es medio." |
| R15 | "Más de la mitad de las veces, cuando el IRRH es medio y el IRC es bajo; entonces el IRP es medio." |

## Anexo #9 Resumen del indicador IRRH

| Identificador | Resúmenes del indicador IRRH |
|---|---|
| R1 | "Más de la mitad de las veces, cuando el IRE es bajo; entonces el IRRH es medio." |
| R2 | "Más de la mitad de las veces, cuando la Clase es Regular; entonces el IRRH es medio." |
| R3 | "Más de la mitad de las veces, cuando el IRP es medio; entonces el IRRH es medio." |
| R4 | "Más de la mitad de las veces, cuando el ICD es alto; entonces el IRRH es medio." |
| R5 | "Más de la mitad de las veces, cuando el IRL es bajo; entonces el IRRH es medio." |
| R6 | "Más de la mitad de las veces, cuando el IRC es bajo; entonces el IRRH es medio." |
| R7 | "Más de la mitad de las veces, cuando el IRE es bajo y el ICD es alto; entonces el IRRH es medio." |
| R8 | "Más de la mitad de las veces, cuando el IRE es bajo y el IRL es bajo; entonces el IRRH es medio." |
| R9 | "Más de la mitad de las veces, cuando el IRE es bajo y el IRC es bajo; entonces el IRRH es medio." |
| R10 | "Más de la mitad de las veces, cuando el ICD es alto y la Clase es Regular; entonces el IRRH es medio." |
| R11 | "Más de la mitad de las veces, cuando el IRL es bajo y la Clase es Regular; entonces el IRRH es medio." |
| R12 | "Más de la mitad de las veces, cuando el IRC es bajo y la Clase es Regular; entonces el IRRH es medio." |
| R13 | "Más de la mitad de las veces, cuando el IRP es medio y el ICD es alto; entonces el IRRH es medio." |
| R14 | "Más de la mitad de las veces, cuando el IRP es medio y el IRL es bajo; entonces el IRRH es medio." |
| R15 | "Más de la mitad de las veces, cuando el IRP es medio y el IRC es bajo; entonces el IRRH es medio." |

## Anexo #10 Tabla de indicadores para la ICD

| Identificador | Validez | Generalidad | Utilidad | Novedad | Simplicidad |
|---|---|---|---|---|---|
| R1 | 0,851851852 | 0,306666667 | 0,306666667 | 0,693333333 | 1 |
| R2 | 1 | 0,373333333 | 0,373333333 | 0,626666667 | 1 |
| R3 | 1 | 0,4 | 0,4 | 0,6 | 1 |
| R4 | 0,893333333 | 0,446666667 | 0,446666667 | 0,553333333 | 1 |
| R5 | 0,897435897 | 0,466666667 | 0,466666667 | 0,533333333 | 1 |
| R6 | 1 | 0,52 | 0,52 | 0,48 | 1 |
| R7 | 1 | 0,62 | 0,62 | 0,38 | 1 |

| | | | | | |
|---|---|---|---|---|---|
| R8 | 1 | 0,62 | 0,62 | 0,38 | 1 |
| R9 | 0,946308725 | 0,94 | 0,94 | 0,06 | 1 |
| R10 | 0,946666667 | 0,946666667 | 0,946666667 | 0,053333333 | 1 |
| R11 | 0,851851852 | 0,306666667 | 0,306666667 | 0,693333333 | 0,5 |
| R12 | 0,851851852 | 0,306666667 | 0,306666667 | 0,693333333 | 0,5 |
| R13 | 1 | 0,373333333 | 0,373333333 | 0,626666667 | 0,5 |
| R14 | 1 | 0,373333333 | 0,373333333 | 0,626666667 | 0,5 |
| R15 | 1 | 0,393333333 | 0,393333333 | 0,606666667 | 0,5 |
| R16 | 1 | 0,4 | 0,4 | 0,6 | 0,5 |
| R17 | 0,891891892 | 0,44 | 0,44 | 0,56 | 0,5 |
| R18 | 0,893333333 | 0,446666667 | 0,446666667 | 0,553333333 | 0,5 |
| R19 | 1 | 0,326666667 | 0,326666667 | 0,673333333 | 0,5 |
| R20 | 1 | 0,333333333 | 0,333333333 | 0,666666667 | 0,5 |
| R21 | 0,897435897 | 0,466666667 | 0,466666667 | 0,533333333 | 0,5 |
| R22 | 0,897435897 | 0,466666667 | 0,466666667 | 0,533333333 | 0,5 |
| R23 | 1 | 0,353333333 | 0,353333333 | 0,646666667 | 0,5 |
| R24 | 1 | 0,393333333 | 0,393333333 | 0,606666667 | 0,5 |
| R25 | 1 | 0,52 | 0,52 | 0,48 | 0,5 |
| R26 | 1 | 0,52 | 0,52 | 0,48 | 0,5 |
| R27 | 1 | 0,433333333 | 0,433333333 | 0,566666667 | 0,5 |
| R28 | 1 | 0,62 | 0,62 | 0,38 | 0,5 |
| R29 | 1 | 0,62 | 0,62 | 0,38 | 0,5 |
| R30 | 1 | 0,62 | 0,62 | 0,38 | 0,5 |
| R31 | 1 | 0,62 | 0,62 | 0,38 | 0,5 |
| R32 | 0,946308725 | 0,94 | 0,94 | 0,06 | 0,5 |
| R33 | 0,851851852 | 0,306666667 | 0,306666667 | 0,693333333 | 0,25 |
| R34 | 1 | 0,373333333 | 0,373333333 | 0,626666667 | 0,25 |
| R35 | 1 | 0,393333333 | 0,393333333 | 0,606666667 | 0,25 |
| R36 | 0,891891892 | 0,44 | 0,44 | 0,56 | 0,25 |
| R37 | 1 | 0,326666667 | 0,326666667 | 0,673333333 | 0,25 |
| R38 | 1 | 0,326666667 | 0,326666667 | 0,673333333 | 0,25 |
| R39 | 1 | 0,333333333 | 0,333333333 | 0,666666667 | 0,25 |
| R40 | 1 | 0,333333333 | 0,333333333 | 0,666666667 | 0,25 |
| R41 | 0,897435897 | 0,466666667 | 0,466666667 | 0,533333333 | 0,25 |
| R42 | 1 | 0,353333333 | 0,353333333 | 0,646666667 | 0,25 |
| R43 | 1 | 0,353333333 | 0,353333333 | 0,646666667 | 0,25 |
| R44 | 1 | 0,393333333 | 0,393333333 | 0,606666667 | 0,25 |
| R45 | 1 | 0,393333333 | 0,393333333 | 0,606666667 | 0,25 |
| R46 | 1 | 0,52 | 0,52 | 0,48 | 0,25 |
| R47 | 1 | 0,433333333 | 0,433333333 | 0,566666667 | 0,25 |
| R48 | 1 | 0,433333333 | 0,433333333 | 0,566666667 | 0,25 |

| | | | | | |
|---|---|---|---|---|---|
| R49 | 1 | 0,62 | 0,62 | 0,38 | 0,25 |
| R50 | 1 | 0,62 | 0,62 | 0,38 | 0,25 |
| R51 | 1 | 0,326666667 | 0,326666667 | 0,673333333 | 0,125 |
| R52 | 1 | 0,333333333 | 0,333333333 | 0,666666667 | 0,125 |
| R53 | 1 | 0,353333333 | 0,353333333 | 0,646666667 | 0,125 |
| R54 | 1 | 0,393333333 | 0,393333333 | 0,606666667 | 0,125 |
| R55 | 1 | 0,433333333 | 0,433333333 | 0,566666667 | 0,125 |

## Anexo #11 Tabla de indicadores para la ICR

| Identificador | Validez | Generalidad | Utilidad | Novedad | Simplicidad |
|---|---|---|---|---|---|
| R1 | 1 | 0,36 | 0,36 | 0,64 | 1 |
| R2 | 1 | 0,373333333 | 0,373333333 | 0,626666667 | 1 |
| R3 | 1 | 0,4 | 0,4 | 0,6 | 1 |
| R4 | 1 | 0,5 | 0,5 | 0,5 | 1 |
| R5 | 1 | 0,52 | 0,52 | 0,48 | 1 |
| R6 | 1 | 0,52 | 0,52 | 0,48 | 1 |
| R7 | 1 | 0,62 | 0,62 | 0,38 | 1 |
| R8 | 1 | 0,62 | 0,62 | 0,38 | 1 |
| R9 | 1 | 0,946666667 | 0,946666667 | 0,053333333 | 1 |
| R10 | 1 | 0,993333333 | 0,993333333 | 0,006666667 | 1 |
| R11 | 1 | 0,306666667 | 0,306666667 | 0,693333333 | 0,5 |
| R12 | 1 | 0,36 | 0,36 | 0,64 | 0,5 |
| R13 | 1 | 0,373333333 | 0,373333333 | 0,626666667 | 0,5 |
| R14 | 1 | 0,373333333 | 0,373333333 | 0,626666667 | 0,5 |
| R15 | 1 | 0,4 | 0,4 | 0,6 | 0,5 |

## Anexo #12 Tabla de indicadores para la IRE

| Identificador | Validez | Generalidad | Utilidad | Novedad | Simplicidad |
|---|---|---|---|---|---|
| R1 | 0,52688172 | 0,326666667 | 0,326666667 | 0,52688172 | 1 |
| R2 | 0,537634409 | 0,333333333 | 0,333333333 | 0,53763441 | 1 |
| R3 | 0,492957746 | 0,466666667 | 0,466666667 | 0,50704225 | 1 |
| R4 | 0,523489933 | 0,52 | 0,52 | 0,48 | 1 |
| R5 | 0,52 | 0,52 | 0,52 | 0,48 | 1 |
| R6 | 0,52688172 | 0,326666667 | 0,326666667 | 0,52688172 | 0,5 |

| | | | | | |
|---|---|---|---|---|---|
| R7 | 0,52688172 | 0,326666667 | 0,326666667 | 0,52688172 | 0,5 |
| R8 | 0,52688172 | 0,326666667 | 0,326666667 | 0,52688172 | 0,5 |
| R9 | 0,537634409 | 0,333333333 | 0,333333333 | 0,53763441 | 0,5 |
| R10 | 0,537634409 | 0,333333333 | 0,333333333 | 0,53763441 | 0,5 |
| R11 | 0,537634409 | 0,333333333 | 0,333333333 | 0,53763441 | 0,5 |
| R12 | 0,496453901 | 0,466666667 | 0,466666667 | 0,5035461 | 0,5 |
| R13 | 0,492957746 | 0,466666667 | 0,466666667 | 0,50704225 | 0,5 |
| R14 | 0,523489933 | 0,52 | 0,52 | 0,48 | 0,5 |
| R15 | 0,52688172 | 0,326666667 | 0,326666667 | 0,52688172 | 0,25 |

**Anexo #13 Tabla de indicadores para la IREF**

| Identificador | Validez | Generalidad | Utilidad | Novedad | Simplicidad |
|---|---|---|---|---|---|
| R1 | 0,471830986 | 0,446666667 | 0,446666667 | 0,528169014 | 1 |
| R2 | 0,496644295 | 0,493333333 | 0,493333333 | 0,503355705 | 1 |
| R3 | 0,5 | 0,5 | 0,5 | 0,5 | 1 |
| R4 | 0,468085106 | 0,44 | 0,44 | 0,531914894 | 0,5 |
| R5 | 0,471830986 | 0,446666667 | 0,446666667 | 0,528169014 | 0,5 |
| R6 | 0,496644295 | 0,493333333 | 0,493333333 | 0,503355705 | 0,5 |
| R7 | 0,468085106 | 0,44 | 0,44 | 0,531914894 | 0,25 |

**Anexo #14 Tabla de indicadores para la IRL**

| Identificador | Validez | Generalidad | Utilidad | Novedad | Simplicidad |
|---|---|---|---|---|---|
| R1 | 1 | 0,36 | 0,36 | 0,64 | 1 |
| R2 | 1 | 0,373333333 | 0,373333333 | 0,626666667 | 1 |
| R3 | 0,983333333 | 0,393333333 | 0,393333333 | 0,606666667 | 1 |
| R4 | 0,986666667 | 0,493333333 | 0,493333333 | 0,506666667 | 1 |
| R5 | 1 | 0,52 | 0,52 | 0,48 | 1 |
| R6 | 1 | 0,52 | 0,52 | 0,48 | 1 |
| R7 | 1 | 0,62 | 0,62 | 0,38 | 1 |
| R8 | 1 | 0,62 | 0,62 | 0,38 | 1 |
| R9 | 0,992957746 | 0,94 | 0,94 | 0,06 | 1 |
| R10 | 0,993333333 | 0,993333333 | 0,993333333 | 0,006666667 | 1 |
| R11 | 1 | 0,306666667 | 0,306666667 | 0,693333333 | 0,5 |
| R12 | 1 | 0,36 | 0,36 | 0,64 | 0,5 |
| R13 | 1 | 0,373333333 | 0,373333333 | 0,626666667 | 0,5 |
| R14 | 1 | 0,373333333 | 0,373333333 | 0,626666667 | 0,5 |

| | | | | | |
|---|---|---|---|---|---|
| R15 | 0,983333333 | 0,393333333 | 0,393333333 | 0,606666667 | 0,5 |

## Anexo #15 Tabla de indicadores para la IRP

| Identificador | Validez | Generalidad | Utilidad | Novedad | Simplicidad |
|---|---|---|---|---|---|
| R1 | 0,628205128 | 0,326666667 | 0,326666667 | 0,628205128 | 1 |
| R2 | 0,679487179 | 0,353333333 | 0,353333333 | 0,646666667 | 1 |
| R3 | 0,698924731 | 0,433333333 | 0,433333333 | 0,566666667 | 1 |
| R4 | 0,654929577 | 0,62 | 0,62 | 0,38 | 1 |
| R5 | 0,624161074 | 0,62 | 0,62 | 0,38 | 1 |
| R6 | 0,62 | 0,62 | 0,62 | 0,38 | 1 |
| R7 | 0,7 | 0,326666667 | 0,326666667 | 0,673333333 | 0,5 |
| R8 | 0,628205128 | 0,326666667 | 0,326666667 | 0,628205128 | 0,5 |
| R9 | 0,628205128 | 0,326666667 | 0,326666667 | 0,628205128 | 0,5 |
| R10 | 0,679487179 | 0,353333333 | 0,353333333 | 0,646666667 | 0,5 |
| R11 | 0,679487179 | 0,353333333 | 0,353333333 | 0,646666667 | 0,5 |
| R12 | 0,679487179 | 0,353333333 | 0,353333333 | 0,646666667 | 0,5 |
| R13 | 0,698924731 | 0,433333333 | 0,433333333 | 0,566666667 | 0,5 |
| R14 | 0,698924731 | 0,433333333 | 0,433333333 | 0,566666667 | 0,5 |
| R15 | 0,698924731 | 0,433333333 | 0,433333333 | 0,566666667 | 0,5 |

## Anexo #16 Tabla de indicadores para la IRRH

| Identificador | Validez | Generalidad | Utilidad | Novedad | Simplicidad |
|---|---|---|---|---|---|
| R1 | 0,641025641 | 0,333333333 | 0,33333333 | 0,641025641 | 1 |
| R2 | 0,756410256 | 0,393333333 | 0,39333333 | 0,606666667 | 1 |
| R3 | 0,698924731 | 0,433333333 | 0,43333333 | 0,566666667 | 1 |
| R4 | 0,654929577 | 0,62 | 0,62 | 0,38 | 1 |
| R5 | 0,624161074 | 0,62 | 0,62 | 0,38 | 1 |
| R6 | 0,62 | 0,62 | 0,62 | 0,38 | 1 |
| R7 | 0,714285714 | 0,333333333 | 0,33333333 | 0,666666667 | 0,5 |
| R8 | 0,641025641 | 0,333333333 | 0,33333333 | 0,641025641 | 0,5 |
| R9 | 0,641025641 | 0,333333333 | 0,33333333 | 0,641025641 | 0,5 |
| R10 | 0,756410256 | 0,393333333 | 0,39333333 | 0,606666667 | 0,5 |
| R11 | 0,756410256 | 0,393333333 | 0,39333333 | 0,606666667 | 0,5 |
| R12 | 0,756410256 | 0,393333333 | 0,39333333 | 0,606666667 | 0,5 |
| R13 | 0,698924731 | 0,433333333 | 0,43333333 | 0,566666667 | 0,5 |
| R14 | 0,698924731 | 0,433333333 | 0,43333333 | 0,566666667 | 0,5 |

| | | | | | |
|---|---|---|---|---|---|
| R15 | 0,698924731 | 0,433333333 | 0,43333333 | 0,566666667 | 0,5 |

## Anexo #11 Imagen de pruebas unitarias de limpiar por la media

Resultado de pasar por parámetro los datos numéricos

Function: **limpiarMedia** Runs: 1

| line | code | calls | time |
|---|---|---|---|
| 1 | { | 0 | 0 |
| 2 | library(DMwR) | 1 | 0 |
| 3 | if (is.numeric(datos[, pos])) { | 1 | 0 |
| 4 | x <- mean(datos[, pos], na.rm = T) | 1 | 0 |
| 5 | for (j in 1:length(datos[, pos])) { | 1 | 0 |
| 6 | if (is.na(datos[j, pos])) { | 27 | 0.01 |
| 7 | datos[j, pos] <- x | 4 | 0 |
| 8 | } | 0 | 0 |
| 9 | } | 0 | 0 |
| 10 | } | 0 | 0 |
| 11 | else { | 0 | 0 |
| 12 | print("Los datos deben ser numéricos") | 0 | 0 |
| 13 | } | 0 | 0 |
| 14 | return(as.data.frame(datos)) | 1 | 0 |
| 15 | } | 0 | 0 |

Resultado de pasar por parámetro los datos nominales

Function: **limpiarMedia** Runs: 1

| line | code | calls | time |
|---|---|---|---|
| 1 | { | 0 | 0 |
| 2 | library(DMwR) | 1 | 0 |
| 3 | if (is.numeric(datos[, pos])) { | 1 | 0 |
| 4 | x <- mean(datos[, pos], na.rm = T) | 0 | 0 |
| 5 | for (j in 1:length(datos[, pos])) { | 0 | 0 |
| 6 | if (is.na(datos[j, pos])) { | 0 | 0 |
| 7 | datos[j, pos] <- x | 0 | 0 |
| 8 | } | 0 | 0 |
| 9 | } | 0 | 0 |
| 10 | } | 0 | 0 |
| 11 | else { | 0 | 0 |
| 12 | print("Los datos deben ser numéricos") | 1 | 0 |
| 13 | } | 0 | 0 |
| 14 | return(as.data.frame(datos)) | 1 | 0 |
| 15 | } | 0 | 0 |

## Anexo #12 Imagen de pruebas unitarias de limpiar por la mediana

Resultado de pasar por parámetro los datos numéricos

Function: **limpiarMediana** Runs: **1**

| line | code | calls | time |
|---|---|---|---|
| 1 | { | 0 | 0 |
| 2 | library(DMwR) | 1 | 0 |
| 3 | for (i in 1:length(datos)) { | 1 | 0 |
| 4 | no.numeric <- FALSE | 7 | 0 |
| 5 | if (is.numeric(datos[, pos])) { | 7 | 0 |
| 6 | x <- median(datos[, pos], na.rm = T) | 7 | 0 |
| 7 | for (j in 1:length(datos[, pos])) { | 7 | 0 |
| 8 | if (is.na(datos[j, pos])) { | 189 | 0.04 |
| 9 | datos[j, pos] <- x | 2 | 0 |
| 10 | } | 0 | 0 |
| 11 | } | 0 | 0 |
| 12 | } | 0 | 0 |
| 13 | else { | 0 | 0 |
| 14 | no.numeric <- TRUE | 0 | 0 |
| 15 | } | 0 | 0 |
| 16 | } | 0 | 0 |
| 17 | if (no.numeric == TRUE) { | 1 | 0 |
| 18 | print("Los datos deben ser numéricos") | 0 | 0 |
| 19 | } | 0 | 0 |
| 20 | else { | 0 | 0 |
| 21 | return(as.data.frame(datos)) | 1 | 0 |
| 22 | } | 0 | 0 |
| 23 | } | 0 | 0 |

Resultado de pasar por parámetro los datos nominales

Function: **limpiarMediana** Runs: **1**

| line | code | calls | time |
|---|---|---|---|
| 1 | { | 0 | 0 |
| 2 | library(DMwR) | 1 | 0 |
| 3 | for (i in 1:length(datos)) { | 1 | 0 |
| 4 | no.numeric <- FALSE | 7 | 0 |
| 5 | if (is.numeric(datos[, pos])) { | 7 | 0 |
| 6 | x <- median(datos[, pos], na.rm = T) | 0 | 0 |
| 7 | for (j in 1:length(datos[, pos])) { | 0 | 0 |
| 8 | if (is.na(datos[i, pos])) { | 0 | 0 |
| 9 | datos[i, pos] <- x | 0 | 0 |
| 10 | } | 0 | 0 |
| 11 | } | 0 | 0 |
| 12 | } | 0 | 0 |
| 13 | else { | 0 | 0 |
| 14 | no.numeric <- TRUE | 7 | 0 |
| 15 | } | 0 | 0 |
| 16 | } | 0 | 0 |
| 17 | if (no.numeric == TRUE) { | 1 | 0 |
| 18 | print("Los datos deben ser numéricos") | 1 | 0 |
| 19 | } | 0 | 0 |
| 20 | else { | 0 | 0 |
| 21 | return(as.data.frame(datos)) | 0 | 0 |
| 22 | } | 0 | 0 |
| 23 | } | 0 | 0 |

## Anexo #13 Imagen de pruebas unitarias de limpiar por la correlación

Resultado de pasar por parámetro los datos nominales

Function: **limpiarCorrelacion** Runs: **1**

| line | code | calls | time |
|---|---|---|---|
| 1 | { | 0 | 0 |
| 2 | library(DMwR) | 1 | 0 |
| 3 | if (is.numeric(datos[, pos]) && is.numeric(datos[, correlacion])) { | 1 | 0 |
| 4 | x <- cor(datos[pos], datos[correlacion], use = "complete.obs") | 0 | 0 |
| 5 | for (j in 1:length(datos[, pos])) { | 0 | 0 |
| 6 | if (is.na(datos[j, pos])) { | 0 | 0 |
| 7 | datos[j, pos] <- x | 0 | 0 |
| 8 | } | 0 | 0 |
| 9 | } | 0 | 0 |
| 10 | return(as.data.frame(datos)) | 0 | 0 |
| 11 | } | 0 | 0 |
| 12 | else { | 0 | 0 |
| 13 | print("Los datos deben ser numéricos") | 1 | 0 |
| 14 | } | 0 | 0 |
| 15 | } | 0 | 0 |

Resultado de pasar por parámetro los datos numéricos

Function: **limpiarCorrelacion** Runs: **1**

| line | code | calls | time |
|---|---|---|---|
| 1 | { | 0 | 0 |
| 2 | library(DMwR) | 1 | 0 |
| 3 | if (is.numeric(datos[, pos]) && is.numeric(datos[, correlacion])) { | 1 | 0 |
| 4 | x <- cor(datos[pos], datos[correlacion], use = "complete.obs") | 1 | 0 |
| 5 | for (j in 1:length(datos[, pos])) { | 1 | 0 |
| 6 | if (is.na(datos[j, pos])) { | 27 | 0.01 |
| 7 | datos[j, pos] <- x | 3 | 0 |
| 8 | } | 0 | 0 |
| 9 | } | 0 | 0 |
| 10 | return(as.data.frame(datos)) | 1 | 0 |
| 11 | } | 0 | 0 |
| 12 | else { | 0 | 0 |
| 13 | print("Los datos deben ser numéricos") | 0 | 0 |
| 14 | } | 0 | 0 |
| 15 | } | 0 | 0 |

## Anexo #14 Imagen de pruebas unitarias de limpiar omitiendo los valores nulos

Resultado de pasar por parámetro los datos nominales

Function: **limpiarNulos** Runs: **1**

| line | code | calls | time |
|---|---|---|---|
| 1 | { | 0 | 0 |
| 2 | no.numeric <- FALSE | 1 | 0 |
| 3 | for (i in 1:length(datos)) { | 1 | 0 |
| 4 | if (is.numeric(datos[, pos])) { | 7 | 0 |
| 5 | if (!sonSignificativos(datos, porciento)) { | 0 | 0 |
| 6 | datos <- subset(datos, subset = !is.na(datos[, pos])) | 0 | 0 |
| 7 | } | 0 | 0 |
| 8 | else { | 0 | 0 |
| 9 | print("Los datos nulos son significativos") | 0 | 0 |
| 10 | } | 0 | 0 |
| 11 | } | 0 | 0 |
| 12 | else { | 0 | 0 |
| 13 | no.numeric <- TRUE | 7 | 0 |
| 14 | } | 0 | 0 |
| 15 | } | 0 | 0 |
| 16 | if (no.numeric == TRUE) { | 1 | 0 |
| 17 | print("Los datos deben ser numéricos") | 1 | 0 |
| 18 | } | 0 | 0 |
| 19 | else { | 0 | 0 |
| 20 | return(as.data.frame(datos)) | 0 | 0 |
| 21 | } | 0 | 0 |
| 22 | } | 0 | 0 |

Resultado de pasar por parámetro los datos numéricos

Function: **limpiarNulos** Runs: **1**

| line | code | calls | time |
|---|---|---|---|
| 1 | { | 0 | 0 |
| 2 | no.numeric <- FALSE | 1 | 0 |
| 3 | for (i in 1:length(datos)) { | 1 | 0 |
| 4 | if (is.numeric(datos[, pos])) { | 7 | 0 |
| 5 | if (!sonSignificativos(datos, porciento)) { | 7 | 0 |
| 6 | datos <- subset(datos, subset = !is.na(datos[, pos])) | 7 | 0 |
| 7 | } | 0 | 0 |
| 8 | else { | 0 | 0 |
| 9 | print("Los datos nulos son significativos") | 0 | 0 |
| 10 | } | 0 | 0 |
| 11 | } | 0 | 0 |
| 12 | else { | 0 | 0 |
| 13 | no.numeric <- TRUE | 0 | 0 |
| 14 | } | 0 | 0 |
| 15 | } | 0 | 0 |
| 16 | if (no.numeric == TRUE) { | 1 | 0 |
| 17 | print("Los datos deben ser numéricos") | 0 | 0 |
| 18 | } | 0 | 0 |
| 19 | else { | 0 | 0 |
| 20 | return(as.data.frame(datos)) | 1 | 0 |
| 21 | } | 0 | 0 |
| 22 | } | 0 | 0 |

Resultado de pasar por parámetro los datos numéricos pero siendo los datos nulos significativos

Function: **limpiarNulos** Runs: 1

| line | code | calls | time |
|---|---|---|---|
| 1 | { | 0 | 0 |
| 2 | no.numeric <- FALSE | 1 | 0 |
| 3 | for (i in 1:length(datos)) { | 1 | 0 |
| 4 | if (is.numeric(datos[, pos])) { | 7 | 0 |
| 5 | if (!sonSignificativos(datos, porciento)) { | 7 | 0 |
| 6 | datos <- subset(datos, subset = !is.na(datos[, pos])) | 0 | 0 |
| 7 | } | 0 | 0 |
| 8 | else { | 0 | 0 |
| 9 | print("Los datos nulos son significativos") | 7 | 0 |
| 10 | } | 0 | 0 |
| 11 | } | 0 | 0 |
| 12 | else { | 0 | 0 |
| 13 | no.numeric <- TRUE | 0 | 0 |
| 14 | } | 0 | 0 |
| 15 | } | 0 | 0 |
| 16 | if (no.numeric == TRUE) { | 1 | 0 |
| 17 | print("Los datos deben ser numéricos") | 0 | 0 |
| 18 | } | 0 | 0 |
| 19 | else { | 0 | 0 |
| 20 | return(as.data.frame(datos)) | 1 | 0 |
| 21 | } | 0 | 0 |
| 22 | } | 0 | 0 |

## Anexo #15 Imagen de pruebas unitarias de discretizar por los limites

Resultado de pasar por parámetro los datos numéricos y limpios

Function: **discretizarLimites** Runs: 1

| line | code | calls | time |
|---|---|---|---|
| 1 | { | 0 | 0 |
| 2 | if (is.numeric(datos[, pos])) { | 1 | 0 |
| 3 | result <- cut(datos[, pos], 3, include.lowest = TRUE, labels = c("bajo", "medio", "alto"), right = TRUE, ordered_result = TRUE) | 1 | 0 |
| 4 | datos[, pos] <- result | 1 | 0 |
| 5 | } | 0 | 0 |
| 6 | else { | 0 | 0 |
| 7 | print("Los datos deben ser numéricos") | 0 | 0 |
| 8 | } | 0 | 0 |
| 9 | return(as.data.frame(datos)) | 1 | 0 |
| 10 | } | 0 | 0 |

Resultado de pasar por parámetro los datos nominales

Function: **discretizarLimites** Runs: 1

| line | code | calls | time |
|---|---|---|---|
| 1 | { | 0 | 0 |
| 2 | if (is.numeric(datos[, pos])) { | 1 | 0 |
| 3 | result <- cut(datos[, pos], 3, include.lowest = TRUE, labels = c("bajo", "medio", "alto"), right = TRUE, ordered_result = TRUE) | 0 | 0 |
| 4 | datos[, pos] <- result | 0 | 0 |
| 5 | } | 0 | 0 |
| 6 | else { | 0 | 0 |
| 7 | print("Los datos deben ser numéricos") | 1 | 0 |
| 8 | } | 0 | 0 |
| 9 | return(as.data.frame(datos)) | 1 | 0 |
| 10 | } | 0 | 0 |

## Anexo #16 Imagen de pruebas unitarias de discretizar mediante conjuntos difusos

Resultado de pasar por parámetro los datos numéricos y limpios

Function: **discretizarConjuntosDifusos** Runs: 1

| line | code | calls | time |
|---|---|---|---|
| 1 | { | 0 | 0 |
| 2 | library(frbs) | 1 | 0 |
| 3 | if (is.numeric(datos[, pos])) { | 1 | 0 |
| 4 | datos.pos <- as.matrix(datos[, pos], nrow = length(datos)) | 1 | 0 |
| 5 | num.fvarinput <- matrix(num.fvarinput, nrow = 1) | 1 | 0 |
| 6 | num.varinput <- matrix(c(1)) | 1 | 0 |
| 7 | matriz.f <- fuzzifier(datos.pos, num.varinput, num.fvarinput, varinp.mf) | 1 | 0 |
| 8 | data.frame.fuzzy <- as.data.frame(matriz.f, row.names = NULL) | 1 | 0 |
| 9 | names(data.frame.fuzzy)[1:ncol(data.frame.fuzzy)] <- names.fvarinput | 1 | 0 |
| 10 | result <- matrix(nrow = nrow(data.frame.fuzzy), ncol = 1) | 1 | 0 |
| 11 | for (i in 1:nrow(data.frame.fuzzy)) { | 1 | 0 |
| 12 | temp <- data.frame.fuzzy[i, 1] | 27 | 0 |
| 13 | result[i, 1] <- names(data.frame.fuzzy)[1] | 27 | 0 |
| 14 | for (j in 1:ncol(data.frame.fuzzy)) { | 27 | 0 |
| 15 | if (data.frame.fuzzy[i, j] > temp) { | 81 | 0.03 |
| 16 | temp <- data.frame.fuzzy[i, j] | 1 | 0 |
| 17 | result[i, 1] <- names(data.frame.fuzzy)[j] | 1 | 0 |
| 18 | } | 0 | 0 |
| 19 | } | 0 | 0 |
| 20 | } | 0 | 0 |
| 21 | datos[, pos] <- result | 1 | 0 |
| 22 | } | 0 | 0 |
| 23 | else { | 0 | 0 |
| 24 | print("Los datos deben ser numéricos") | 0 | 0 |
| 25 | } | 0 | 0 |
| 26 | return(as.data.frame(datos)) | 1 | 0 |
| 27 | } | 0 | 0 |

Resultado de pasar por parámetro los datos nominales

Function: **discretizarConjuntosDifusos** Runs: **1**

| line | code | calls | time |
|---|---|---|---|
| 1 | { | 0 | 0 |
| 2 | library(frbs) | 1 | 0 |
| 3 | if (is.numeric(datos[, pos])) { | 1 | 0 |
| 4 | datos.pos <- as.matrix(datos[, pos], nrow = length(datos)) | 0 | 0 |
| 5 | num.fvarinput <- matrix(num.fvarinput, nrow = 1) | 0 | 0 |
| 6 | num.varinput <- matrix(c(1)) | 0 | 0 |
| 7 | matriz.f <- fuzzifier(datos.pos, num.varinput, num.fvarinput, varinp.mf) | 0 | 0 |
| 8 | data.frame.fuzzy <- as.data.frame(matriz.f, row.names = NULL) | 0 | 0 |
| 9 | names(data.frame.fuzzy)[1:ncol(data.frame.fuzzy)] <- names.fvarinput | 0 | 0 |
| 10 | result <- matrix(nrow = nrow(data.frame.fuzzy), ncol = 1) | 0 | 0 |
| 11 | for (i in 1:nrow(data.frame.fuzzy)) { | 0 | 0 |
| 12 | temp <- data.frame.fuzzy[i, 1] | 0 | 0 |
| 13 | result[i, 1] <- names(data.frame.fuzzy)[1] | 0 | 0 |
| 14 | for (j in 1:ncol(data.frame.fuzzy)) { | 0 | 0 |
| 15 | if (data.frame.fuzzy[i, j] > temp) { | 0 | 0 |
| 16 | temp <- data.frame.fuzzy[i, j] | 0 | 0 |
| 17 | result[i, 1] <- names(data.frame.fuzzy)[j] | 0 | 0 |
| 18 | } | 0 | 0 |
| 19 | } | 0 | 0 |
| 20 | } | 0 | 0 |
| 21 | datos[, pos] <- result | 0 | 0 |
| 22 | } | 0 | 0 |
| 23 | else { | 0 | 0 |
| 24 | print("Los datos deben ser numéricos") | 1 | 0 |
| 25 | } | 0 | 0 |
| 26 | return(as.data.frame(datos)) | 1 | 0 |
| 27 | } | 0 | 0 |

## Anexo #17 Imagen de pruebas unitarias de discretizar mediante rangos equivalentes

Resultado de pasar por parámetro los datos numéricos y limpios

Function: **discretizarRangosEquiv** Runs: **1**

| line | code | calls | time |
|---|---|---|---|
| 1 | { | 0 | 0 |
| 2 | library(Rcmdr) | 1 | 0 |
| 3 | if (is.numeric(datos[, pos])) { | 1 | 0 |
| 4 | aux <- bin.var(datos[, pos], bins = 3, method = "natural", labels = c("bajo", "medio", "alto")) | 1 | 0 |
| 5 | datos[, pos] <- aux | 1 | 0.01 |
| 6 | } | 0 | 0 |
| 7 | else { | 0 | 0 |
| 8 | print("Los datos deben ser numéricos") | 0 | 0 |
| 9 | } | 0 | 0 |
| 10 | return(as.data.frame(datos)) | 1 | 0 |
| 11 | } | 0 | 0 |

Resultado de pasar por parámetro los datos nominales

Function: **discretizarRangosEquiv** Runs: 1

| line | code | calls | time |
|---|---|---|---|
| 1 | { | 0 | 0 |
| 2 | library(Rcmdr) | 1 | 0 |
| 3 | if (is.numeric(datos[, pos])) { | 1 | 0 |
| 4 | aux <- bin.var(datos[, pos], bins = 3, method = "natural", labels = c("bajo", "medio", "alto")) | 0 | 0 |
| 5 | datos[, pos] <- aux | 0 | 0 |
| 6 | } | 0 | 0 |
| 7 | else { | 0 | 0 |
| 8 | print("Los datos deben ser numéricos") | 1 | 0 |
| 9 | } | 0 | 0 |
| 10 | return(as.data.frame(datos)) | 1 | 0 |
| 11 | } | 0 | 0 |

## Anexo #18 Imagen de pruebas unitarias de resumen lingüístico

Function: **resumenLinguistico** Runs: 1

| line | code | calls | time |
|---|---|---|---|
| 1 | { | 0 | 0 |
| 2 | result <- matrix(nrow = nrow(reglas), ncol = 1) | 1 | 0 |
| 3 | for (i in 1:nrow(reglas)) { | 1 | 0 |
| 4 | a <- reglas[i, 1] | 31 | 0 |
| 5 | b <- sub("} => {", "; entonces ", a, fixed = TRUE) | 31 | 0.02 |
| 6 | b <- gsub("=", " es ", b) | 31 | 0 |
| 7 | b <- gsub(",", " y ", b) | 31 | 0 |
| 8 | for (j in 1:nrow(articulos)) { | 31 | 0 |
| 9 | art <- paste(articulos[j, 1], articulos[j, 2]) | 248 | 0.06 |
| 10 | x <- grepl(art, b, fixed = TRUE) | 248 | 0.03 |
| 11 | if (x == FALSE) { | 248 | 0.04 |
| 12 | b <- sub(articulos[j, 2], art, b) | 248 | 0.03 |
| 13 | } | 0 | 0 |
| 14 | } | 0 | 0 |
| 15 | for (k in 1:nrow(cuantificadores)) { | 31 | 0 |
| 16 | if ((reglas[i, 3] >= cuantificadores[k, 1]) && (reglas[i, 3] <= cuantificadores[k, 2])) { | 155 | 0.04 |
| 17 | d <- cuantificadores[k, 3] | 31 | 0.01 |
| 18 | } | 0 | 0 |
| 19 | } | 0 | 0 |
| 20 | b <- sub("{", d, b, fixed = TRUE) | 31 | 0.01 |
| 21 | b <- sub("}", ".", b, fixed = TRUE) | 31 | 0.01 |
| 22 | result[i, 1] <- b | 31 | 0 |
| 23 | } | 0 | 0 |
| 24 | result <- as.data.frame(result) | 1 | 0 |
| 25 | return(result) | 1 | 0 |
| 26 | } | 0 | 0 |