Universidad Central “Marta Abreu” de Las Villas

Facultad de Matemática, Física y Computación

Licenciatura en Ciencia de la Computación

TRABAJO DE DIPLOMA

Extensión del módulo de visualización científica de Weka.

Autor:*Shamith Wimukthi Yatagama Lokuge*

Tutores : *Dr. Carlos Pérez Risquet*

Santa Clara

2015

Hago constar que el presente trabajo fue realizado en la Universidad Central Marta Abreu de Las Villas como parte de la culminación de los estudios de la especialidad de Ciencias de la Computación, autorizando a que el mismo sea utilizado por la institución, para los fines que estime conveniente, tanto de forma parcial como total y que además no podrá ser presentado en eventos ni publicado sin la autorización de la Universidad.

_____

Firma del autor

Los abajo firmantes, certificamos que el presente trabajo ha sido realizado según acuerdos de la dirección de nuestro centro y el mismo cumple con los requisitos que debe tener un trabajo de esta envergadura referido a la temática señalada.

_____ _____

Firma del tutor Firma del jefe del Seminario

Dedicatoria

Quisiera dedicar mi Trabajo de diploma a mis padres, que son lo más grande que tengo en la vida.

Agradecimientos

Largo y colmado de esfuerzos ha sido el camino transitado para llegar a la materialización de este añorado sueño, en el trayecto muchos han sido los inconvenientes pero verdaderamente aún más numerosas han sido las personas que han brindado su aporte de una forma u otra en toda esta carrera.

A todos ellos mi más Sincero Agradecimiento.

Hoy con la realización de este trabajo culmino esta decisiva etapa en mi vida, su materialización no habría sido posible sin la ayuda incondicional y oportuna de un grupo de personas que no quiero dejar de mencionar:

A mi tutor Carlos Pérez Risquet por su valiosa ayuda.

A Romel que desde lejos continuaba pendiente de todo.

A mi madre por sus múltiples consejos metodológicos y su disposición.

A Kasuni y Lakshan por su apoyo y alegrando mis días con su presencia.

A mi amigo Issak por formar parte de mi vida en los 6 años de la carrera, por su constancia, entrega y ayuda incondicional.

A todas las personas, compañeros de estudio y profesores, que con pequeños aportes contribuyeron con este trabajo.

En fin a todos aquellos que de una forma u otra me brindaron su apoyo y ayuda incondicional, porque:

“No hay deber más necesario que el de dar las gracias”.

"Sabios son aquellos que dominan el cuerpo, la palabra y la mente. Ellos son los verdaderos Maestros."

Dhammapada 17:14

Resumen

La visualización científica es una herramienta útil durante el proceso de extracción de conocimientos. Weka es un poderoso sistema de aprendizaje automático, que en cambio ofrece pocas alternativas en el área de la visualización.

En este proyecto se adiciona un módulo de visualización científica a Weka. Al módulo se han incorporado las siguientes técnicas de visualización para datos multiparamétricos: Patrón Recursivo, Theme River, Table Lens.

El módulo tiene gran flexibilidad, por lo que en futuros trabajos se podrán adicionar otras técnicas o modificar las existentes. Fue creado como un paquete independiente de Weka, que permite incorporarlo a otros sistemas.

Abstract

The scientific visualization is a useful tool during the process of discovery of knowledge. Weka is a powerful machine learning system, which in return offers very few alternatives in the visualization area.

In this project, a module of scientific visualization is added to Weka. The following visualization techniques have been added to the module for multiparametric data: Recursive Pattern, Theme River, Table Lens. A very valuable technique has been added to the cluster from gravity affected particles.

The module has great flexibility, so that in future papers, other techniques can be added or the present ones modification. It was developed as an independent package, which permits its involvement in other systems.

Contenido

Introducción

Inicialmente en el año 1993 la universidad de Waikato ubicada en Nueva Zelanda se dio a la tarea de desarrollar la primera implementación para analizar datos procedentes de la agricultura y con esto se dio lo que fue la primera versión de WEKA, ésta fue desarrollada en TCL/TK y lenguaje C, cuatro años más tarde en 1997 se decide escribir todo el código original en java, además se le incluyeron implementaciones de algoritmos de modelado.

En el año 2005 esta herramienta muy flexible y fácil de utilizar, recibe el galardón (*Data Mining and Knowledge Discovery Service*), por parte de la ACM (*Asociación for Computing Machinery*) que es la Sociedad Científica Para el Desarrollo de la Computación Educacional (Martias and Araugo, 2006).

Weka es un conjunto de librerías java para la extracción de conocimientos desde bases de datos. El paquete Weka contiene una colección de herramientas de visualización y técnicas de aprendizaje automático, supervisado, no supervisado, esta herramienta por su nombre en inglés (*Waika to Environment for Knowledge Analysis*) además es una herramienta de distribución de licencia GNU-GLP o software libre (Martias and Araugo, 2006).

La minería de datos encierra un grupo de técnicas para la solución de múltiples problemas en distintos campos, en la actualidad existen muchas herramientas enfocadas a la minería de datos que utilizan este sin número de técnicas con el objetivo de obtener conocimientos nuevos desde las bases de datos. Con el paso del tiempo han surgido más y más herramientas para la explotación de datos con el fin de abarcar un mercado que ha tenido un enorme auge en los últimos años, brindando soluciones sobre las fuertes demandas de las empresas, han surgido herramientas de índole comercial como Cart, SPSS Clementine, Kxen, SAS Enterprise Miner, Tiberius, por otro lado han surgido herramientas para la aplicación de minería de acceso gratuito, o licencia libre dentro de este otro grupo se pueden mencionar MiningMart, Orange, TariyKDD, ARMiner y

WEKA, que es la herramienta de minería que se seleccionó para desarrollar esta investigación.

El incremento constante de los volúmenes de datos generados en muchos campos de aplicación crea la necesidad de elaborar herramientas que permitan extraer información de estos datos del contenido de la potencia de las interfaces gráficas modernas. Junto al desarrollo de nuevas técnicas de visualización se han creado numerosas utilidades que emplean estas técnicas, tanto en forma de bibliotecas como de programas (Grandy, 2014).

La ciencia siempre ha tratado de entender los fenómenos de la naturaleza. Sin embargo, estos fenómenos son a veces muy grandes, o muy pequeños, muy rápidos o muy lentos para ser estudiados con los métodos tradicionales. La visualización científica es una herramienta que permite a los científicos computacionales analizar, entender y comunicar los datos numéricos generados durante una investigación.

Es una representación visual apropiada para un conjunto de datos que permita mayor efectividad en el análisis y evaluación de los mismos. Eso se permite la transformación de los datos numéricos o simbólicos y la información en imágenes geométricas generadas por computadora. Es una metodología para interpretar, a través de una imagen en la computadora, tanto datos de mediciones como los generados por modelos computacionales. La investigación y el desarrollo de la visualización científica se han centrado en cuestiones relacionadas con el renderizado de gráficos en tres dimensiones, animaciones de series temporales y visualización interactiva en tiempo real (Morell, C.Pérez, 2007).

Una clase especial de datos son los datos multiparamétricos (datos multidimensionales o datos multivariados). Los datos multiparamétricos son los que poseen m variables o dimensiones de datos escalares distribuidos sobre puntos en el espacio de observación. Estas variables pueden ser cuantitativas o cualitativas y a su vez ordinales o nominales (Hansen and Johnson, 2005).

El objetivo fundamental de los métodos de visualización para datos multiparamétricos es lograr que las representaciones revelen correlaciones o patrones entre los atributos (Eick, 2000; Theisel, 2000). Con este fin existe actualmente una amplia gama de técnicas de visualización, para las cuales se han creado además diversas mejoras. Las técnicas pueden ser clasificadas en geométricas, basadas en icono, basadas en píxel y proyecciones (Salgado, 2003).

Por las ventajas señaladas anteriormente de Weka se concreta con los algoritmos para análisis de datos y modelado predictivo, unidos a una interfaz gráfica de usuario para acceder fácilmente a sus funcionalidades. El sistema Weka, que es utilizado en áreas como la minería de datos, presenta un módulo de visualización científica muy pobre, con unas pocas opciones. Adicionarle a este sistema un módulo de visualización científica incrementaría las potencialidades del mismo permitiendo a los usuarios ser parte del proceso de extracción de conocimiento, mediante un método tan interactivo y natural como la visualización.

Objetivo General.

Extender el módulo de visualización científica del sistema Weka, para que facilite la exploración y análisis de los datos iniciales y apoye la visualización de resultados de métodos de aprendizaje automático.

Objetivos Específicos

1. Estudiar técnicas de visualización de grandes volúmenes de datos y seleccionar las más adecuadas para añadirlas al Weka.
2. Implementar un paquete para la exploración y análisis visual de los datos iniciales.
3. Adicionar el módulo de visualización a Weka .

Preguntas de Investigación:

1. ¿Cuáles técnicas de visualización sería apropiado añadir a la extensión?
2. ¿Qué estructura específica se adoptaría para la adición a Weka de cada una de ellas?
3. ¿Cómo enfrentar el reto de agregar técnicas de visualización científica a un sistema que no está específicamente diseñado para ello?

Justificación de la investigación

- La adición de un módulo de visualización científica a Weka aportaría valiosas herramientas a los usuarios del sistema.
- La representación de los datos iniciales permitiría a los usuarios conocer con facilidad características de los mismos, que podrían ser desde informaciones estadísticas básicas hasta importantes conclusiones sobre un problema.

- Los resultados producidos por los métodos de aprendizaje automático del sistema serían más fáciles de estudiar permitiendo investigaciones más profundas y análisis más detallados.
- Los usuarios del sistema con menos experiencia asimilarían más fácilmente los resultados

Viabilidad de la investigación.

El estado actual de las técnicas de visualización científica ofrece una amplia gama de ideas a desarrollar en este trabajo. Por otro lado se cuenta con el apoyo del grupo de Computación Gráfica del Centro de Estudios Informáticos que tiene suficiente experiencia en el área.

Existe más de una biblioteca gráfica que ya ha sido estudiada y por tanto puede ser empleada en la elaboración de este proyecto. La herramienta WEKA ha sido utilizada a lo largo de la carrera, así como el lenguaje Java.

Estructura general de la tesis.

Este trabajo está dividido en tres capítulos.

Primer capítulo: Se realiza una introducción sobre la visualización científica junto con la herramienta Weka. En la visualización científica se describen los tipos de datos que utilizan las técnicas de visualización y sobre la base de este criterio se efectúa una clasificación de las mismas. Posteriormente se realiza una explicación de las técnicas fundamentales Profundizando en las usadas para representar datos multiparamétricos. Se abordan también las técnicas de visualización como apoyo a la búsqueda de conglomerados. Finalmente se realiza un resumen de las principales características y funcionalidades de Weka, especialmente las opciones de visualización.

Segundo capítulo: Se describen los datos usados por el sistema Weka y se eligen las técnicas que se adicionarán al módulo de visualización. Para cada una de estas técnicas se seleccionan los detalles específicos de la implementación. A continuación se realiza el proceso de diseño del software y se muestra la metodología para extender el módulo.

Tercer capítulo: Se presenta el manual de usuario del módulo y se dan algunos consejos para su utilización.

Capítulo 1. Técnicas de Visualización Científica.

Debido al gran avance que existe día con día en las tecnologías de información, las organizaciones se han tenido que enfrentar a nuevos desafíos que les permitan analizar, descubrir y entender más allá de lo que sus herramientas tradicionales reportan sobre su información, al mismo tiempo que durante los últimos años el gran crecimiento de las aplicaciones disponibles en internet (*geo-referenciamiento, redes sociales, etc.)* han sido parte importante en las decisiones de negocio de las empresas. Una característica fundamental de las modernas tecnologías es la producción de grandes volúmenes de datos.

En los últimos años, los avances en la tecnología han facilitado la obtención de grandes cantidades de información. Mediante imágenes de satélite, estaciones de medición de alta precisión, supercomputadoras o cualquier otra fuente de este tipo, se generan a diario volúmenes de datos muy grandes y complejos. Para el estudio de estos datos es necesario el uso de técnicas avanzadas debido a que no pueden ser analizados suficientemente bien en forma numérica. De todos los datos generados por entes especializados, solo una cuarta parte se almacena, y de éstos solo una cuarta parte realmente se analiza. Como se puede ver se pierden datos valiosos, de muchas informaciones importantes solamente se utiliza un pequeño por ciento.

La recolección de datos se realiza con motivos bien definidos, pues evidentemente los datos son susceptibles a contener información relevante para un determinado problema. La extracción de información a partir de los datos puede convertirse en un reto para los investigadores, producto del volumen de los datos primarios o de la complejidad de los mismos (Hansen, Johnson, 2005).

La ciencia ha desarrollado diversos métodos para la obtención de información, y uno de ellos se basa en la creación de imágenes a partir de los datos. Este método, conocido como visualización, ha sido utilizado como vía natural para mostrar información (Hansen, Johnson, 2005). Recientes investigaciones han impulsado en gran medida este campo mediante el uso de la computación.

La visualización de datos puede categorizar a diferentes metas, entre ellos categorización por datos iniciales es más importante. La naturaleza del objetivo que se persigue está en relación directa al conocimiento que se tenga sobre los datos iniciales. Los objetivos pueden ser los siguientes (Grandy, 2014):

- Análisis exploratorio.
- Análisis confirmativo.
- Presentación de información.

El **Análisis exploratorio** Se tiene un conjunto de datos sin una hipótesis específica. Estos datos se someten a un proceso de búsqueda interactiva de información que va a arrojar como resultado una visualización que soporte una hipótesis sobre el conjunto de datos.

El **Análisis confirmativo** Se tiene un conjunto de datos sobre los que se plantea una hipótesis. Se realiza un procesamiento de dichos datos que genera una visualización mediante la cual se pueda validar o refutar la hipótesis que se tenía de ellos.

La **Presentación de información** parte de hechos que son fijos a priori y que se desean enfatizar y mostrar con extrema calidad.

Para llevar acabo y tener un buen resultado es importante realizar un breve estudio sobre las posibilidades que ofrecen las técnicas de visualización y las facilidades de weka en este aspecto.

1.1 Introducción a las técnicas de visualización científica.

Diversos enfoques se han empleado para agrupar y clasificar las diferentes técnicas de Visualización Científica existentes. Un enfoque establecido para clasificar las técnicas es a través del tipo de dato sobre el que opera. Por tipo de dato, se entiende como al tipo que pertenecen los atributos o variables. Atendiendo a este criterio se encuentran las siguientes categorías (Hansen and Johnson, 2005; Grandy, 2014).
Técnicas de visualización para datos volumétricos.

Técnicas de visualización para datos fluidos.

Técnicas de visualización para datos multiparamétricos.

Técnicas de visualización de la información.

Existen diversos enfoques para especificar los datos (Pérez, 2004). Estos enfoques permiten definir una amplia variedad de característica de los datos como son la dimensionalidad, el nivel de medición y la estructura. En este trabajo utilizaremos un enfoque simple para definir los datos (Pérez, 2004; Hansen and Johnson, 2005; Grandy, 2014).

Los datos volumétricos: Se representan una malla de tres dimensiones donde cada punto tiene asociado un valor. En general los datos se definen como un conjunto S de muestras, donde cada elemento s∈ S es un vector de la forma (x, y, z, v) que contiene las coordenadas espaciales y un elemento que es un escalar (Gallagher, 1994; Pérez, 2004; Hansen and Johnson, 2005 ; Grandy, 2014).

Los datos fluidos: Los campos vectoriales representan una malla de dimensión menor o igual que tres donde cada punto está relacionado con un vector. Una de las áreas de mayor uso de los campos vectoriales es para representar datos de fluidos (Gallagher, 1994; Hansen, Johnson, 2005).

Los datos multiparamétricos: son aquellos en que el número de variables relacionadas con cada observación es mayor o igual que dos. Estas variables pueden ser cuantitativas o cualitativas y a su vez ordinales o nominales (Pérez, 2004; Hansen and Johnson, 2005).

La información: En algunas aplicaciones los datos presentan una estructura que no concuerda con ninguna de las anteriores o que sencillamente no puede ser definida con exactitud. A estos datos se le suele llamar información y entre las principales se identifican estructuras como árboles, grafos e hipertexto (Keim, 2002; Hansen and Johnson, 2005; Grandy, 2014).

A continuación se relacionan las principales técnicas de visualización de acuerdo al tipo de dato con que operan. Este trabajo se centró en la implementación de técnicas de visualización para datos multiparamétricos. Es por ello que se hace mayor énfasis en este tipo de técnicas.

1.2 Técnicas de visualización para volúmenes.

La visualización de volúmenes permite la obtención de información a partir de datos espaciales. A pesar de que la estructura de los datos volumétricos ya se definió anteriormente en algunas aplicaciones específicas pueden representarse mediante funciones, imágenes o mediante observaciones individuales (Gallagher, 1994; Hansen, Johnson, 2005).

Dentro del campo de la visualización de datos volumétricos pueden encontrarse técnicas que se clasifican como técnicas de representación de superficies o de representación directa de volúmenes (Hansen, Johnson, 2005), estas técnicas se pueden observar en mayor detalle a continuación.

Representación de superficies

Cuando se representan volúmenes a través de superficies, solo se muestra la parte exterior de los datos del volumen, de forma tal que las superficies creadas se colorean según el valor v. Se conoce como contorneado a la técnica fundamental de extracción de superficies a partir de datos volumétricos, que en dos dimensiones se llama isolíneas y en tres isosuperficies (Gallagher, 1994) (Hansen, Johnson, 2005)(Grandy, 2014). En la figura 1-1 se observa la técnica de isosuperficie.

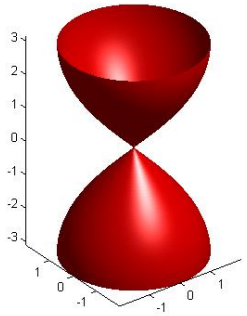

Figura 1-1: Isosuperficies.

Representación Directa de Volúmenes

Cuando se visualizan los volúmenes de forma directa se muestra mayor cantidad de información que cuando se utiliza la técnica de superficie para datos volumétricos, esta efectividad se logra con una mayor complejidad computacional. Para dibujar volúmenes existen dos técnicas básicas, estas son Raycasting y la Proyección de Elementos del Volumen (Hansen, Johnson, 2005) (Grandy, 2014).

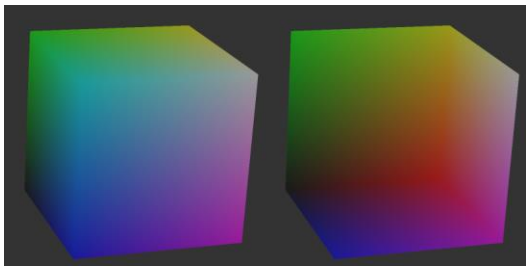

Figura 1-2: Raycasting

1.3 Técnicas de visualización para datos de fluidos.

Para utilizar algoritmos se requiere que los atributos sean de naturaleza vectorial. Con los vectores se pueden representar datos como fuerza y velocidad. Estos conjuntos de datos generalmente resultan del estudio de fluidos (Hansen and Johnson, 2005; Grandy, 2014).

Un método sencillo para representar campos vectoriales es conocido con el nombre de *erizo*. (Hansen, Johnson, 2005) Cada línea comenzará en el punto asociado al vector. El

mayor inconveniente de esta técnica se hace palpable cuando el conjunto de datos crece, pues en estas circunstancias se pierden los detalles.

La técnica de Alteración (en Inglés *Warping*), ver en figura 1-3, está basada en que los campos vectoriales generalmente se asocian con acción. En esta técnica se procederá a deformar la geometría representada por los datos, la deformación estará basada en los vectores asociados. La imagen creada consistirá en la geometría sin deformar y la deformada, lo que permite observar a través de la comparación el efecto provocado por los vectores. En caso de que la magnitud de algunos vectores sea muy pequeña el efecto de los mismos no se notará por lo que la técnica perderá efectividad (Hansen, Johnson, 2005).

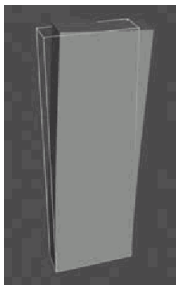

Figura 1-3: Warping. La geometría original puede verse de forma alambrada. El cuerpo sólido esta deformado por la acción del campo vectorial.

La visualización de fluidos es un caso particular de la visualización de atributos vectoriales. Este campo ha sido estudiado en profundidad y existen un número considerable de técnicas que se pueden usar. De estas técnicas se pueden destacar algunas como son las flechas (en Inglés *arrows*) y las líneas de flujo (en Inglés *StreamLines*) (Grandy, 2014).

Cuando se quiere mostrar el movimiento de partículas producto de la acción de un campo vectorial se puede usar la técnica de las líneas de flujo. Los vectores son usados para construir una línea que representa el recorrido de una partícula en un espacio de tiempo(Grandy, 2014). Para ver un ejemplo se puede observar la figura 1-4.

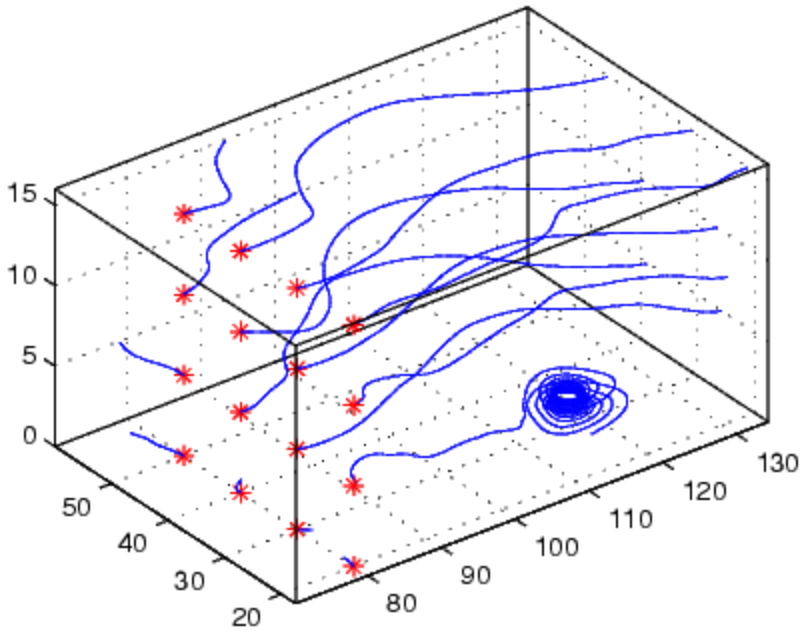

Figura 1-4: StreamfLines.

1.4 Visualización de información.

La visualización de la información contempla la presentación de datos jerárquicos, de textos y grafos. Estos datos suelen tener características estructurales especiales, que los diferencian de otros tipos y que pueden y deben ser utilizadas en el proceso de visualización (Andrews, 2005; Keim, 2002; Hansen and Johnson, 2005; Grandy, 2014). A continuación se relacionan algunas de las técnicas de visualización fundamentales en esta área.

Visualización de datos jerárquicos

La jerarquía es una forma común de representar ciertos datos, en muchos casos ellos son intrínsecamente jerárquicos. La jerarquía describe la topología de los datos, no el tipo de las dimensiones (Martig, 2000; Theisel, 2000). Los ejemplos de uso de esta clase de información son muy disímiles y uno clásico lo constituye la estructura de directorios de los sistemas de archivo.

Todos los objetos se pueden describir jerárquicamente. Para ello basta con observar que los objetos suelen estar formados por múltiples componentes que a su vez están formados por subcomponentes. Cada uno de estos componentes representa un elemento en la jerarquía y está descrito por diferentes rasgos. Otro ejemplo común de jerarquía es la relación que existe entre clases de la programación orientada a objetos.

La representación tradicional o universal, de los datos jerárquicos es el árbol (Martig, 2000; Gallagher, 1994; Hansen and Johnson, 2005). Un árbol es una estructura jerárquica por naturaleza. La raíz del árbol representa la mayor jerarquía y cada rama representa un nuevo nivel de jerarquía. Esta técnica permite que niveles de la misma jerarquía estén a la misma "distancia" de la raíz. Los árboles se han utilizado en diversas esferas que van desde la creación de genealogías hasta la representación del curso evolutivo.

Otra variante muy popular de imagen que muestra jerarquías es la pirámide (Theisel, 2000). En la pirámide el triángulo superior representa la máxima jerarquía y el descenso de nivel significa descender en la jerarquía.

Cuando se implementa la técnica de la jerarquía se persiguen dos metas fundamentalmente(Grandy, 2014):

- Mostrar las relaciones jerárquicas entre los objetos.
- Mostrar los rasgos que describen los objetos.

Tomando como base la representación jerárquica en forma de árbol, se han desarrollado varias técnicas de visualización. Entre las más importantes están (Salgado, 2003; Theisel, 2000; Grandy, 2014):

ConeTree.

Hiperbolic browser.

Tree View.

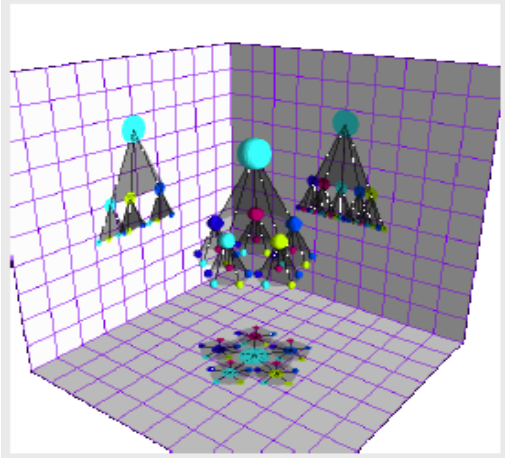

Figura 1-5: ConeTree.

Una de las técnicas antes mencionada es la ConeTree que se puede observar en la figura 1-5. Algunas representaciones se han creado basándose en la proyección de una jerarquía en un espacio dado. El espacio a utilizar es por lo general 2D pero no está excluida la utilización de 1D o 3D. En este tipo de visualización el espacio es dividido en partes según los niveles de jerarquía. Como ejemplo tenemos las técnicas de TreeMap, en la figura 1-5 se muestra un ejemplo.

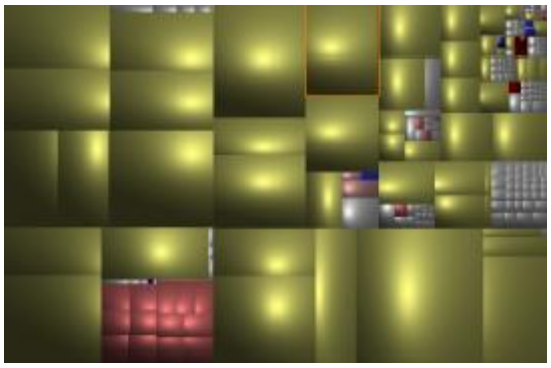

Figura 1-6: TreeMap. Muestra el uso de un disco.

Visualización de Redes y Grafos

Gran cantidad de problemas se pueden representar como redes y por tanto pueden ser analizados usando este tipo de visualización. Entre las problemáticas que por su

naturaleza se expresan como redes, se encuentran los mapas conceptuales y los problemas de tráfico.

Las redes son en esencia nodos y enlaces entre nodos. Tanto los nodos como los enlaces pueden contener información variada (Hansen and Johnson, 2005). La representación de redes puede utilizarse en áreas como la comprensión de estructuras y el flujo de tráfico, haciendo posible la identificación de nodos importantes y de enlaces claves (Hansen and Johnson, 2005). La visualización de una red persigue en un amplio sentido los siguientes objetivos (Hansen and Johnson, 2005; Grandy, 2014):

Mostrar la estructura de la red y las conexiones entre nodos.
Mostrar características de los enlaces.
Mostrar datos de los nodos.
Encontrar una representación de una red es fácil pues el nombre de la estructura de datos sugiere la misma, pero esta representación puede mostrar deficiencias en muchas aplicaciones. El motivo es que al representar una red puede producirse desorden en la imagen. La probabilidad de surgimiento del desorden se incrementa con el crecimiento de la red. El origen del desorden es la gran cantidad de líneas que representan los enlaces y que suelen cruzar el grafo en todas direcciones. Otra de las deficiencias del método tradicional es que las variaciones en la posición de los nodos pueden tener un fuerte impacto en la percepción de la representación (Gallagher, 1994; Hansen and Johnson, 2005).

Con el objetivo de combatir estas deficiencias se crearon un número de técnicas de visualización de los grafos. La característica fundamental de estas es la obtención de una matriz como resultado de la visualización, dicha matriz representa el grafo. Este diseño minimiza el desorden pero en cambio impacta la percepción de la estructura y limita el tamaño de la red (Hansen, Johnson, 2005).

Existen metodologías para minimizar el desorden que se basa en el filtrado de los datos. Al filtrar los datos se reduce el volumen tanto de nodos como de enlaces. Esta variante es aceptable cuando no es grande la pérdida de información o cuando el desorden es intratable por otras vías (Martig, 2000; Gallagher, 1994).

1.5 Técnicas de visualización para datos multiparamétricos.

Existen una serie de problemas en que cada punto de dato contiene más de un atributo, estos atributos pueden ser fechas, precios o valores descriptivos. A este tipo de datos se les llama multiparamétricos y se encuentran generalmente en aplicaciones de minería de datos, estadísticas e inteligencia artificial (Keim, 2002). Los datos multiparamétricos, también llamados multidimensionales o datos n-dimensionales, consisten en un número de n registros donde cada uno está definido por un vector de d valores. Estos datos pueden ser vistos como una matriz de nxd, donde cada fila representa un registro y cada columna representa una observación, variable o dimensión (Keim, 2002).

El objetivo fundamental de los métodos de visualización para datos multiparamétricos es lograr que las representaciones revelen correlaciones o patrones entre los atributos (Eick, 2000; Keim, 2002; Grandy, 2014). Con este fin existe actualmente una amplia gama de técnicas de visualización, para las cuales se han creado además diversas mejoras. Las técnicas pueden ser clasificadas en geométricas, basadas en iconos, basadas en píxel y proyecciones (Keim, 2002; Grandy, 2014) entre otras.

1.5.1 Técnicas geométricas.

Las técnicas geométricas se basan en el establecimiento de una relación entre los datos correspondientes a los atributos y un espacio geométrico (Yang, 2010). Dicho de otra forma son las técnicas que utilizan elementos como puntos, líneas o curvas como propiedades visuales para representar los datos (Keim, 2002; Grandy, 2014). Existe un gran número de ellas, las visualizaciones geométricamente transformadas pretenden encontrar patrones interesantes en conjuntos de datos multidimensionales.

La técnica de Table Lens.

Una idea similar para representar datos multiparamétricos es el TableLens. En esta técnica para visualización se soporta una forma ligera para análisis exploratorio de datos (EDA) por integración de una organización familiar. El Table Lens es esencialmente el equivalente gráfico de una tabla de relaciones o una hoja contable en la cual las filas representan casos y las columnas representa variables. Además cada filas y columnas pueden ser asignadas para cantidades variables que se asignadas de espacio, que le permite soportar incorporación directa de representaciones textuales de valores. Para las

variables cuantitativas, una barra gráfica se usa para representar los valores, las barras dentro de la columna de cada variable se alinean con su borde izquierdo que puede indicar un valor mínimo.

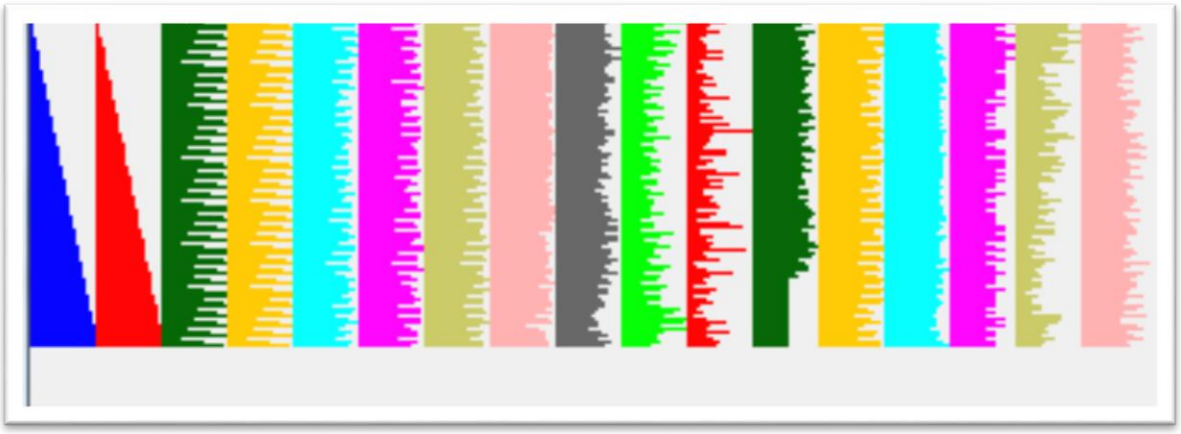

Figura 1-7: Gráfico Table Lens.

Esta técnica, puede verse en la figura 1-7, permite identificar con facilidad diferencias entre grupos de observación ya que por lo general observaciones pertenecientes a un mismo grupo presentan una forma similar (MatLab, 2014).

Diagramas de dispersión (en inglés*ScatterPlot*).

El Diagrama de dispersión es una técnica simple muy utilizada. Su forma más sencilla se manifiesta cuando los datos tienen solo dos dimensiones. Con dos dimensiones la técnica consiste en trazar un eje de coordenadas y utilizar los valores de las dimensiones como punto (x,y) de R^2 resultando un gráfico donde se observan dispersos los puntos de datos. Por otro lado, visualizar datos de más de dos dimensiones no es obvio, para lograrlo pueden utilizarse proyecciones, que provocan pérdida de información debido a la reducción de la dimensión (Keim, 2002; Hansen and Johnson, 2005; Grandy, 2014).

Para datos multiparamétricos es muy frecuente utilizar matrices de diagramas de dispersión. Las matrices resultantes son cuadradas y el elemento (i,j) de la matriz es un diagrama de dispersión de la dimensión i y la j. El diseño evita la pérdida de información pero en cambio son engorrosos los análisis complejos. Una deficiencia adicional es que la

diagonal principal de la matriz es subutilizada. Algunos trabajos actuales están encaminados a aprovechar mejor esta región de la representación (Keim, 2002). Ver figura 1-8.

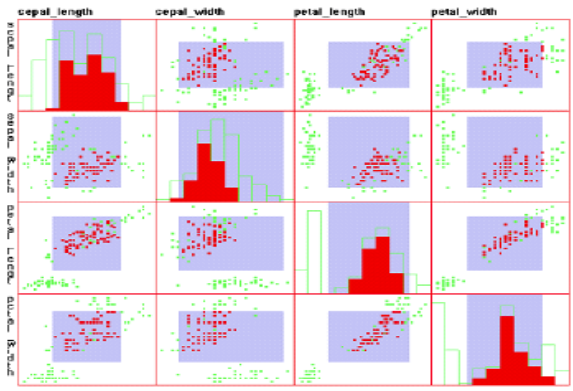

Figura 1-8: Matrices de diagramas de dispersión con la diagonal principal con histogramas.

Coordenadas Paralelas.

Las coordenadas paralelas son un método de visualización diseñado para crear una representación en 2D de datos multidimensionales sin perdida de información. La técnica fue introducidapor**Inselberg y Dimsdale(1990)**. En ella se visualiza una tupla de datos (x1, x2,…, xn) comouna línea poligonal, conectando los puntos x1, x2, xn en n ejes y paralelos (obsérvese la figura2.1-E). Para volúmenes de datos suficientemente grandes la visualización de la técnica puede llegar a ser confusa. Una posible solución consiste en una extensión en3D, donde el plano xy representa la versión en 2D de las coordenadas paralelas, mientras la dimensión z representa la densidad de los eventos [33]**.**

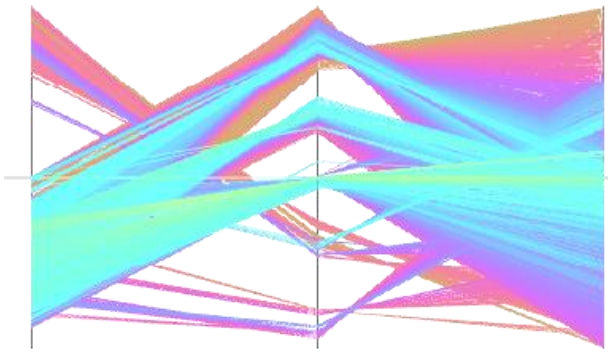

Figura 1-9: Coordenadas paralelas. Este ejemplo muestra un conjunto de datos con tres atributos.

Río Temático (Theme River)

La visualización de Río Temático es de gran utilidad para ver la variación de datos en una gran colección de datos. Los cambios son mostrados en el contexto de una línea de tiempo. Los cambios en la imagen permiten al usuario discernir patrones más fácilmente y analizar la relación entre los datos.

Tales patrones son más difíciles de analizar en otro tipo de visualizaciones. El flujo de izquierda a derecha es interpretado como el movimiento de los datos en el tiempo. En cualquier punto del gráfico el grosor de la línea indica el valor o fuerza de la variable (Pérez , 2004).

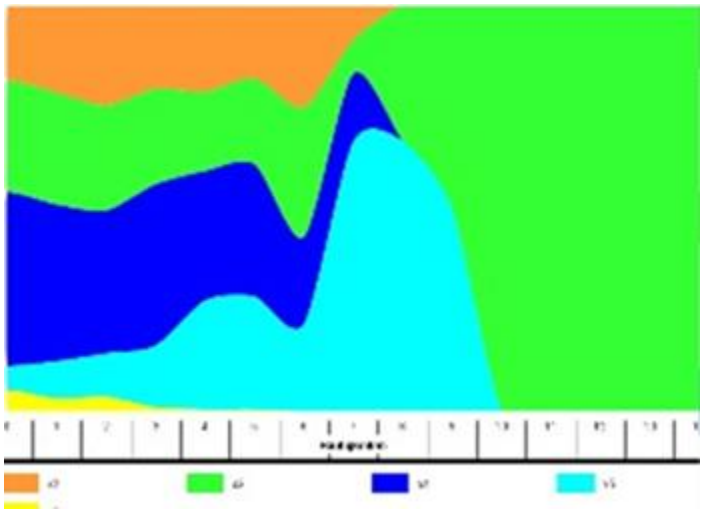

Figura 1-10: **Río Temático (ThemeRiver)**

Un efecto secundario de esta visualización es que al eliminar los espacios en blanco para no afectar la continuidad de la visualización. Cada variable tiene un efecto sobre la adyacente a ella que no es necesariamente consecuente con los valores de las variables en ese momento (Pérez, 2004).

1.5.2 Técnicas basadas en iconos.

Las técnicas basadas en iconos visualizan datos multidimensionales mediante la asignación de cada objeto de datos sobre valores de los parámetros en pequeñas gráficas primitivas. Normalmente, los valores de los atributos están representados por la x e y posición del icono así como la longitud, el ángulo o forma de algún componente cónico. Para lograr un buen resultado, los componentes dentro de un icono deben ser distinguibles, iconos separados deben ser claramente identificables y los iconos deben ser percibidos como distintos si difieren en algunos de los componentes.Las técnicas basadas en iconos tienen dos parámetros que la caracterizan. El primero es el tipo defigura que representará cada observación, o sea, la forma del icono; el segundo parámetro es la forma en que se definirá la posición de cada icono en la imagen [14] (Grandy, 2014). Estas técnicas no sufren de pérdida de información. Se logra evitar la pérdida de información al realizar una proyección de las dimensiones a los diferentes rasgos del icono (Grandy, 2014). Las técnicas basadas en iconos son recomendadas cuando el número de dimensiones oscila entre diez y quince y el número de mediciones de las mismas es alto. Estas técnicas se pueden utilizar con una referencia espacial.

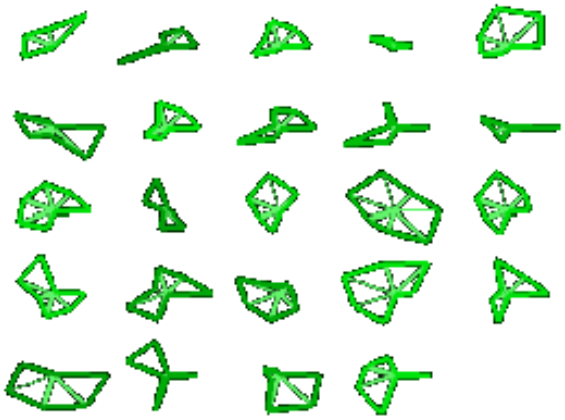

Figura 1-11: StarField.

1.5.3 Técnicas orientadas a píxel.

La visualización de un conjunto de datos de gran tamaño resulta un reto para técnicas geométricas y basadas en iconos. Al graficarlos suele surgir des orden en la imagen, que

está originado por el tamaño de la simple. Partiendo de esta idea resulta lógico concluir que minimizando el espacio que ocupa un solo punto de dato en la imagen se mejoraría la percepción visual (Theisel, 2000; Hansen and Johnson, 2005). Para lograrla maximización del número de elementos a representar, algunas técnicas utilizan los pixeles de la pantalla como unidades básicas de representación (Yang, 2010). El procedimiento consiste en relacionar cada valor de una dimensión a un color y agrupar los pixeles de cada dimensión en áreas adyacentes (Keim, 2002).

De esta manera una computadora con una resolución de pantalla de 1024 * 800 pixeles podría potencialmente representar 819200 elementos de un conjunto de datos univariados. Este tipo de técnicas se utiliza en diferentes modos de posicionamiento de los pixeles para lograr diferentes objetivos. Colocar los pixeles en la forma apropiada ofrece la posibilidad de observar información sobre correlaciones, de pendencias y regiones trascendentales. Dos de los modos de posicionamiento de los pixeles son los patrones recursivos (Ankerst, 1996) (obsérvese Figura 1-10) y los segmentos de círculo (obsérvese Figura 1-11) (Ankerst, 1996; Keim, 2002). Otros ejemplos de técnicas orientadas a pixel son la de espiral (Ankerst, 1996).

Patrones recursivos

La técnica orientada a pixel patrones recursivos tiene una manera especial de interactuar con los datos, permitiendo la definición de diferentes niveles de recursividad. Está particularmente dirigida a representar un conjunto de datos con un orden natural de acuerdo a un atributo, propiedad que la convierte en una opción para problemas de series de tiempo. Una posibilidad simple que provee la técnica es la de organizar los puntos de datos de izquierda a derecha, línea por línea o columna por columna. Una vía posible de mejorarla visualización es la organización delos pixeles en pequeños grupos y organizarlos grupos para formar un patrón global. Esta estrategia corresponde a un planteamiento en dos fases con un patrón de primer orden

Formado por la agrupación de los pixeles y un patrón de segundo orden formado por el orden global. Al tomar los resultados de la estructura de segundo orden como elelemento básico de construcción de una estructura de tercer nivel, puede realizarse la introducción de un tercer patrón. Este proceso puede ser repetido hasta un nivel arbitrario formando un esquema general recursivo. Este esquema general recursivo puede ser distribuido de dos formas para cada nivel de recursividad línea a línea (*line by line*), donde las estructuras de cada orden se posicionan en la imagen de izquierda a derecha o intercalando el sentido (*back and fort*), las estructuras de cada orden se posicionan intercalando el sentido, es decir para una línea se posicionan de izquierda a derecha y en la siguiente línea de derecha a izquierda (Keim, 2002).

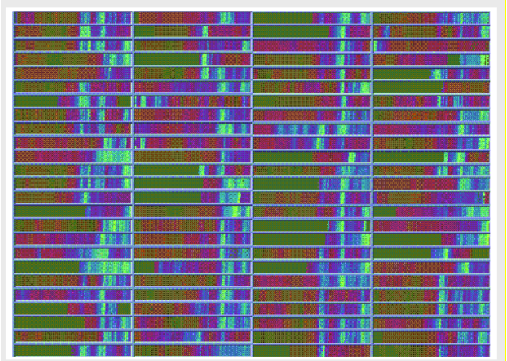

Figura 1-12: La técnica de patrones recursivos.

La misma secuencia básica se hace en todos los niveles de recursión con la única diferencia que los elementos básicos situados en el nivel i son los patrones resultantes de las ubicaciones del nivel i -1. Si w_i es el número de elementos ubicados de izquierda a derecha en el nivel i y h_i es el número de filas en el nivel i, entoncesel patrón en el nivel i consiste de $w_i * h_i$ nivel (i-1)-patrones, y el máximo número de pixeles que pueden ser representados en el nivel k está dado por la productoria desde i = 1 hasta k de $w_i * h_i$ (Keim, 2002). El algoritmo Draw (obsérvese el listado de la figura 1.12 permite realizar la visualización de patrones recursivos. Inicialmente el algoritmo se llama con Draw (0, 0, MAX-LEVEL) con el ancho y alto de todos los niveles de recursividad almacenados en un arreglo previamente definido. La condición de parada es la llegada al nivel de recursión 0.Para los niveles i (i = 1), el algoritmo dibuja w_i nivel (i-1)-patrones h_i veces, en dependencia de la opción line by line o back and fort, lo hace alternando o no elsentido.

```
DRAW(x,y,level)

1 ifx = 0

2then SET-PIXEL(x,y,color)

3       else for h ← 1 toheight[level]

4 do if h mod 2 = 0

5 then for w ←1 towidth[level]

6 do DRAW(x,y,level- 1)

7    x ←x+∏_{i=1}^{level−1} w i

8          else for w ← 1 towidth[level]

9       do x←x- ∏_{i=1}^{level−1} w i
```

10 DRAW(x,y,level -1)

11 y ←$\prod_{i=1}^{level-1} h\, i$

Algoritmo de patrones recursivos

Segmentos de círculo

Como su nombre lo indica la idea fundamental de esta técnica es mostrar las dimensiones de los datos como segmentos de un círculo. Si el conjunto de datos consiste en n variable, el círculo es consecuentemente particionado en n segmentos. Los elementos de los datos dentro de cada segmento son organizados de un lado hacia el otro a través de la llamada línea de dibujo o drawline, ortogonal a la línea que divide las dos líneas del borde, los segmentos (obsérvese Figura 1-13).Cada vez que la línea de dibujo toca uno de las líneas del borde, la primera se mueve en paralelo junto a la línea divisoria del segmento hacia el exterior del círculo y la dirección de la línea de dibujo cambia. Este proceso se repite luego para cada una de las variables restantes (Keim, 2002; Hansen and Johnson, 2005).

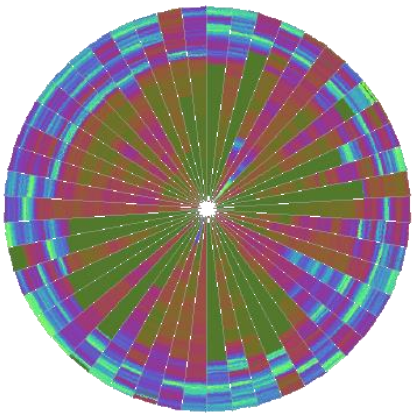

Figura 1-13: Píxel compacto usando posicionamiento basado en segmentos de círculo.

El algoritmo Fill- Segment se llama con las dos líneas del borde del segmento y se introduce en la subrutina Initial-Pixels. La función Initial-Pixels dibuja los primeros pixeles de un segmento hasta que la siguiente línea de dibujo tenga al menos un pixel entre las dos líneas del borde. El valor de retorno de Initial-Pixels es el número de pixel es dibuja dos hasta ahora. La función Compute-Next-Point se mueve hacia delante por la

línea de dibujo. La función Point-Betw-Lines comprueba si un punto está aún en el segmento. Si el punto no está en el segmento, la línea de dibujo se mueve un pixel hacia el exterior en paralelo con la línea divisoria del segmento. La nueva línea de dibujo dibuja el pixel en la dirección opuesta a la anterior.

1.6 Visualización como apoyo a los métodos automáticos de extracción de conocimientos.

Los métodos de aprendizaje automático tratan de extraer conocimientos de un conjunto de datos. Los procesos de extracción de conocimientos pueden tener variados objetivos, entre ellos (Nauck, 1997):

- Extracción de conocimientos para construir un modelo de clasificación.
- Selección de rasgos significativos de un conjunto de datos.
- Extracción de dependencias entre atributos.
- Agrupamiento de observaciones del conjunto de datos.

Para cada uno de estos objetivos los investigadores han desarrollado diversos métodos. Los procedimientos ideados tienen como base, esencialmente, la estadística y la inteligencia artificial. Los métodos de aprendizaje automático se dividen en dos categorías esenciales (Nauck, 1997): el aprendizaje supervisado y el aprendizaje no supervisado.

Se han desarrollado técnicas para apoyar diferentes procesos de extracción de conocimientos, o sencillamente algunas se han adaptado para mostrar información. Entre estas técnicas destacan aquellas que permiten mostrar reglas, conglomerados y árboles de clasificación (Pérez, 2004; Hansen, Johnson, 2005).

1.7 Descripción general de WEKA (Waikato Environment for Knowledge Analysis)

La herramienta Weka es un ambiente de trabajo para la prueba y validación de algoritmos de la IA. Tiene implementada una colección de algoritmos conocidos, varias maneras para preprocesar los archivos de datos a utilizar por dichos algoritmos; así como facilidades para validar los mismos. Posee interfaces gráficas de usuario (GUI: Graphical

User Interface) y cuenta con herramientas para realizar tareas de regresión, clasificación, agrupamiento, asociación y visualización (Martias and Araugo, 2006).

Entre los métodos de aprendizaje automático que contiene se encuentran los de clasificación, búsqueda de conglomerados y selección de rasgos. Además, brinda gran potencialidad para la transformación de los datos a través de numerosos filtros.

Técnicas de Visualización implementadas en Weka

Weka es una herramienta realmente pobre en el área de las visualizaciones, donde brinda muy pocas opciones a los usuarios. De esta forma se hace prácticamente imposible realizar una verdadera exploración de los datos, constituyendo una real limitante para los usuarios de este sistema. Sin embargo, hemos visto en el desarrollo del capítulo la forma en que algunas técnicas sencillas, tales como las coordenadas paralelas o los segmentos de círculo, pueden mejorar la exploración y análisis de los datos.

1.8 Conclusiones parciales.

Al concluir la revisión bibliográfica se arriban a las conclusiones siguientes:

- Dependiendo del tipo de dato que se utilice será la clasificación que tendrán las técnicas de visualización. Este criterio es particularmente útil al desarrollar aplicaciones específicas pues permite seleccionar las técnicas a emplear.
- Existen diversas técnicas de visualización para datos multiparamétricos por lo que estas suelen ser útiles para diferentes conjuntos de datos. Por ello es conveniente utilizar varias simultáneamente.
- Weka presenta serias deficiencias en el área de la visualización científica. Existe un número considerable de técnicas que pueden incluírsele para aumentar su potencial.

Capítulo 2. Diseño e Implementación de la extensión.

En el segundo capítulo se definen las técnicas que se incluyen en el módulo de visualización para el sistema Weka, además de realizar el proceso de diseño del nuevo módulo definiendo las características específicas que se utilizarán en la implementación. Se muestra una metodología para la realización de futuras extensiones al módulo de visualización.

2.1 Análisis del problema. Selección de las técnicas a incluir en la extensión.

El proceso de desarrollo de técnicas de visualización para algún sistema tiene como punto de partida natural el tipo de datos que utiliza el sistema. Esto se obtiene producto del modelo de clasificación de las técnicas que se describió anteriormente. La correcta descripción de los datos usados por Weka permite identificar las técnicas que pueden aportar más al software.

Descripción de los datos.

Weka utiliza una fuente de datos multiparamétricos que consiste en un conjunto C de cardinalidad m en que cada elemento $Oi=<V_1,\ldots V_n>$, $i=1\ldots m$ es una n-upla. A cada elemento de C se le conoce como instancia u observación y cada componente de una observación es una dimensión o variable (Pérez, 2004; Grandy, 2014). En estos datos, cada variable puede tener nivel de medición continua o nominal.

Adicionalmente se ha podido constatar que los conjuntos de datos suelen tener un número relativamente mediano de dimensiones y el número de observación en los datos varía considerablemente.

Elección de las técnicas

Al estudiar los objetivos del proyecto, y las funcionalidades actuales del sistema Weka se decidió desarrollar un paquete con el objetivo de visualización de datos multiparamétricos.

Las técnicas que se decidieron incluir en el módulo de visualización de datos multiparamétricos son:

- Patrón Recursivo.
- Table Lens.
- Theme River.

Las técnicas seleccionadas ofrecen un amplio número de beneficios y generalidad ya que se incluyen las más tradicionales que permiten mostrar conjuntos de datos con disímiles características en cuanto a la cantidad de observación, dimensiones y el tipo de las mismas.

2.2 Diseño de la extensión realizada al sistema Weka.

Para el desarrollo y extensión de cualquier sistema computacional se tiene como necesidad inicial lograr un diseño apropiado del problema a tratar. Con este objetivo se debe hacer un adecuado modelado del negocio, identificando los actores y caso de uso que intervienen en el mismo. Para ello se tendrá en cuenta el diseño ya establecido del sistema que se está extendiendo y las características de los módulos que se pretenden añadir. Se analizarán las clases que surgen en la solución del problema así como otras con las cuales se interactúa y que ya estaban presentes. Logrando de esta forma una mejor comprensión del problema y de la solución que se propone.

Análisis de actores y casos de uso

Los modelos de casos de uso muestran el conjunto de casos de uso, actores y sus relaciones. Cubren la vista de casos de uso estática de un sistema. Estos diagramas son especialmente importantes en el modelado y organización del comportamiento de un sistema. La extensión implementada está destinada a un solo tipo de actor que es el especialista o investigador de cualquier rama que desee explorar sus datos de manera visual y que sean capaces de interpretar de forma correcta los resultados que brinda la herramienta

A partir del planteamiento del problema podemos extraer al único actor que tiene el sistema al cual llamaremos especialista y es el encargado de interactuar con el mismo con múltiples objetivos. A su vez los casos de uso del sistema son muchos por los que analizaremos los que surgen como consecuencia de la inclusión de los nuevos módulos. El diagrama que analizaremos a continuación ilustra lo anterior.

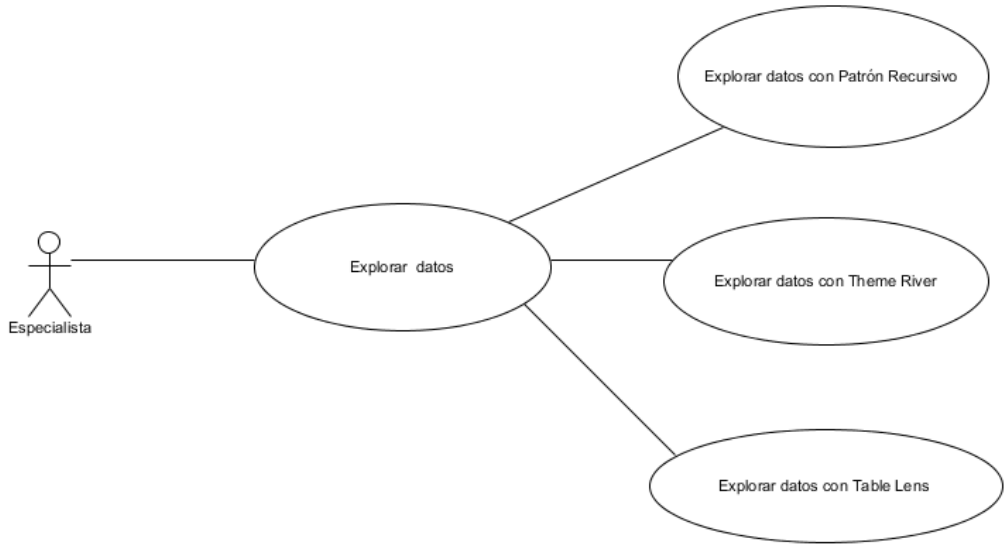

Figura 2-1: Actores y casos de uso.

Como casos de uso principales se consideraron: la exploración de un conjunto de datos y la visualización del resultado de un algoritmo previamente obtenido. La instancia de caso de uso tenemos la exploración usando Patrón Recursivo, Theme River, Table Lens.

Diagrama de estado

En el proceso de obtención de una visualización exploratoria de los datos el sistema pasa por una serie de estados los cuales se muestran en el siguiente diagrama. Consideramos un estado inicial en el que ya se captaron los datos y el usuario está listo para elegir una visualización.

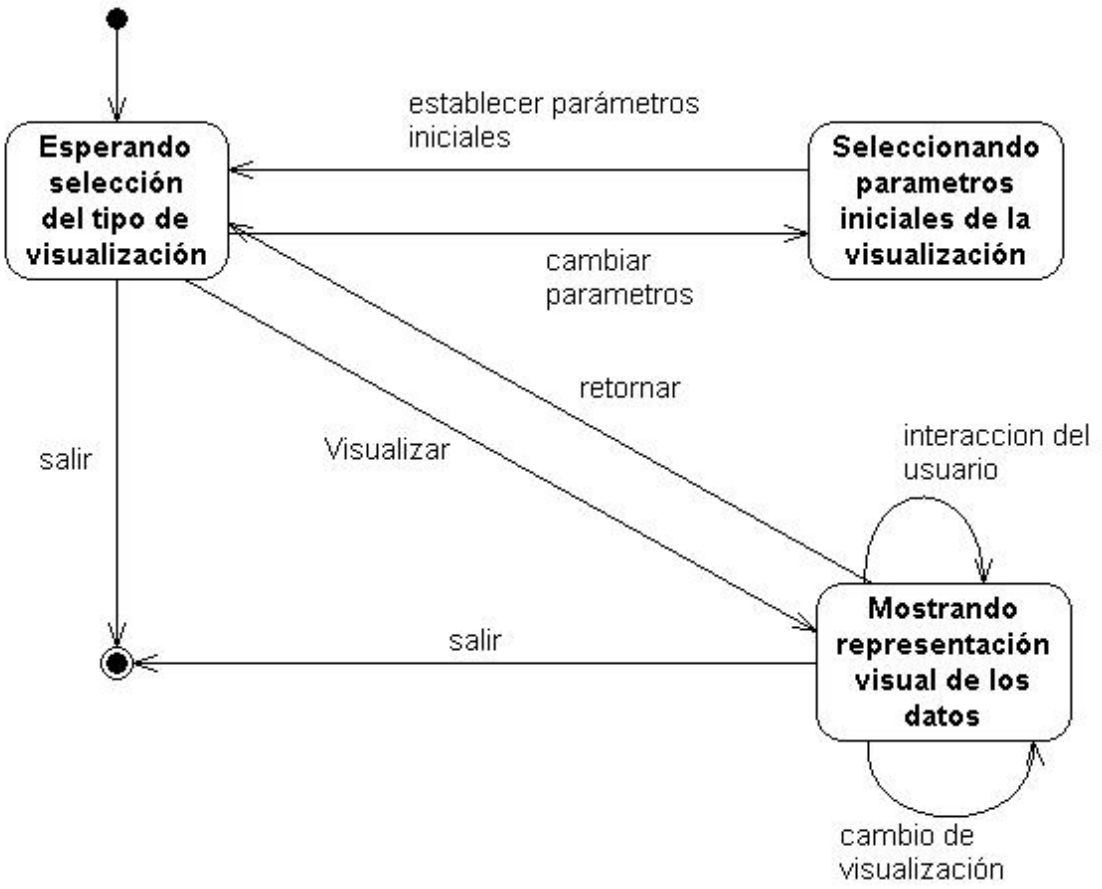

Figura 2-2: Diagrama de transición de estado para el proceso de exploración visual de datos.

Componentes del sistema

El sistema cuenta con dos componentes principales: una fuente de datos y un paquete Weka que contiene todas las implementaciones de este sistema. En este paquete se puede identificar como un tercer componente el módulo de visualización (*Visualization)* donde se encuentran los nuevos algoritmos implementados. Se puede considerar una componente más, ya que su implementación no lo hace dependiente del sistema Weka. Ver Figura 2.4.

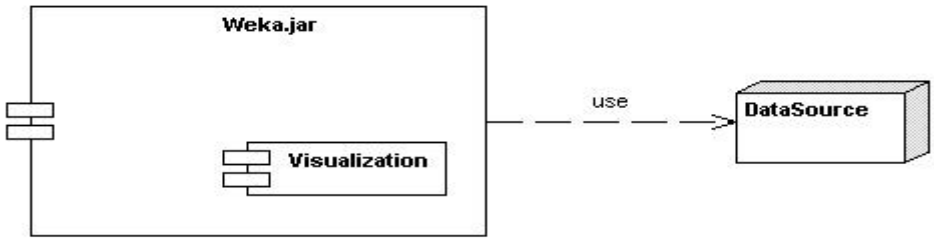

Figura 2-3: Diagrama de componentes del sistema

Diagrama de clases

El módulo de visualización está dividido en dos paquetes: el de visualización de conglomerados y el de visualización de datos multiparamétricos. A continuación se muestran las clases que conforman el paquete de visualización de datos multiparamétricos (figura 2-5).

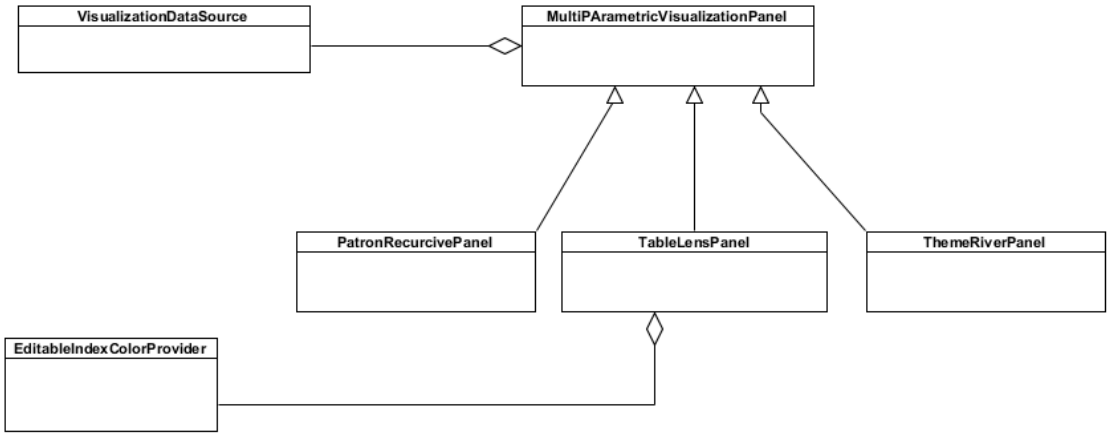

Figura 2-4: Diagrama de clases del paquete de visualización de datos multiparamétricos

Para lograr la integración del Weka con el nuevo paquete de visualización de datos multiparamétricos se procede a la creación de un paquete que cumpla con esta responsabilidad. El diagrama de clases que aparece a continuación describe dicho paquete

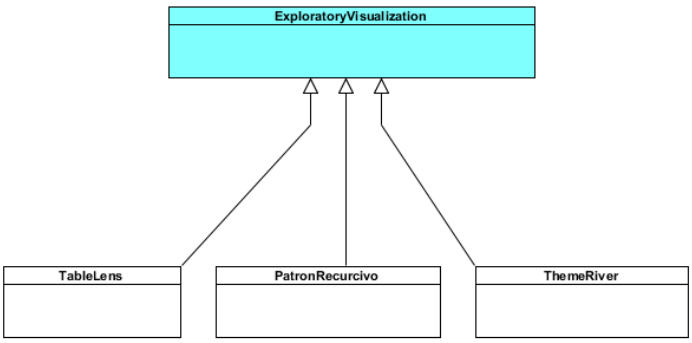

Figura 2-5: Diagrama de clases del paquete de enlace

El siguiente diagrama de clases muestra las relaciones entre los diferentes paneles de configuración para las técnicas (figura 2-8).

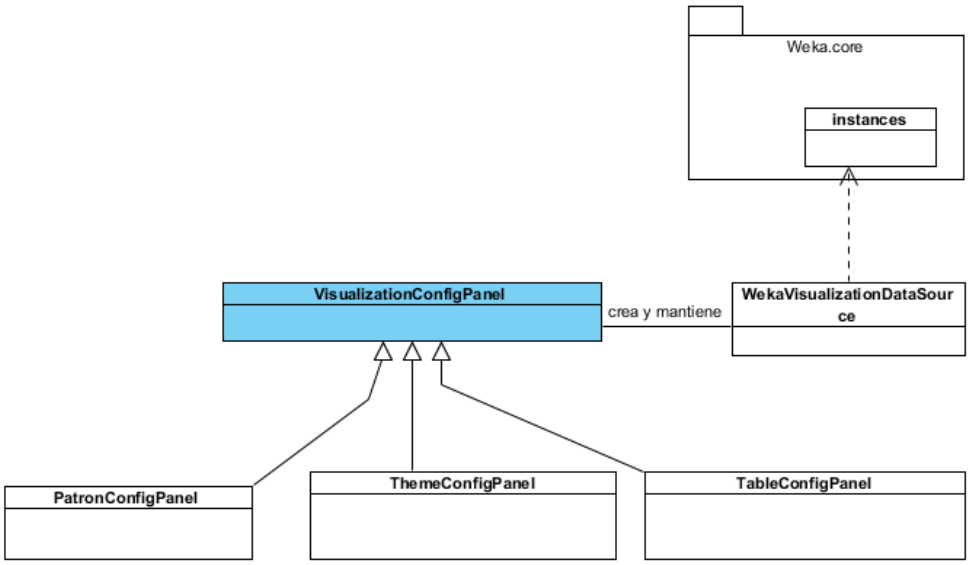

Figura 2-6: Diagrama de clases de configuración.

Los dos diagramas de clases que aparecen en figura 2-9 y 2-10 muestran las relaciones entre las clases del paquete de visualización de datos multiparamétricos y Weka.

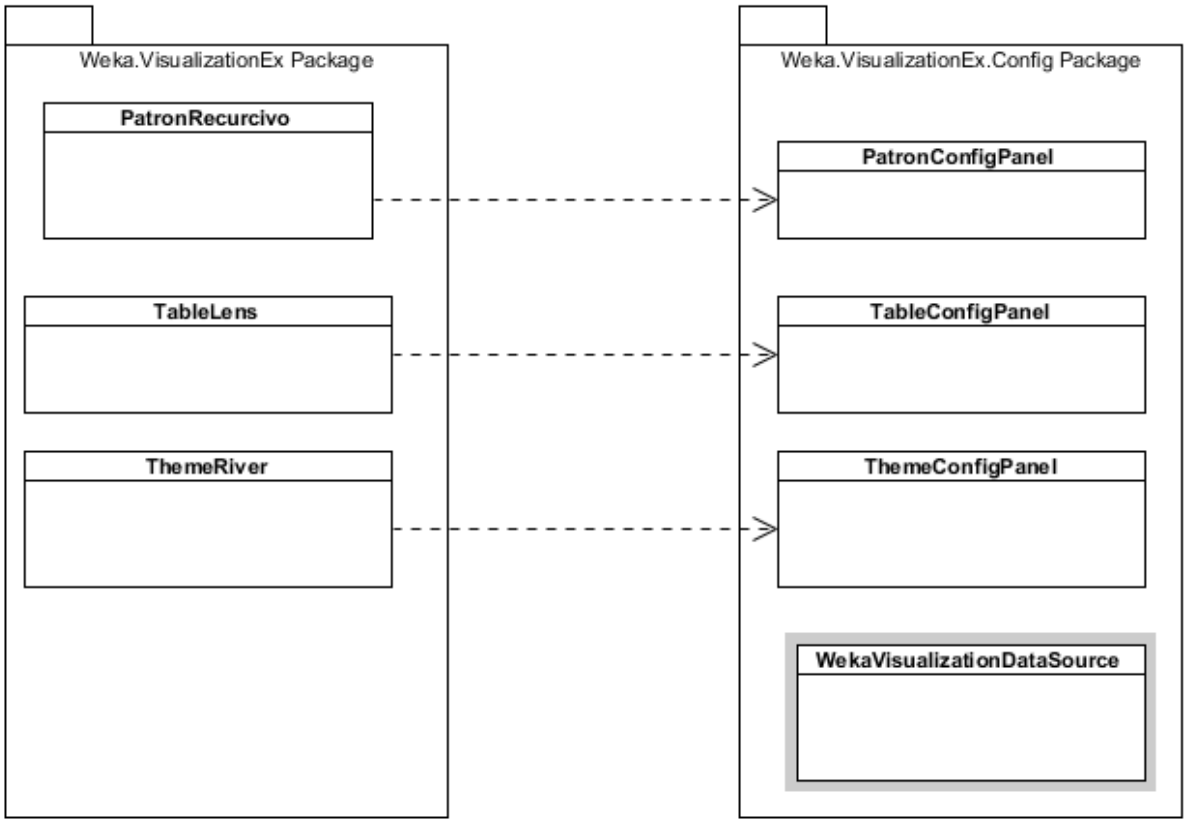

Figura 2-7: Diagrama de clases que muestra las relaciones entre clases de dos paquetes.

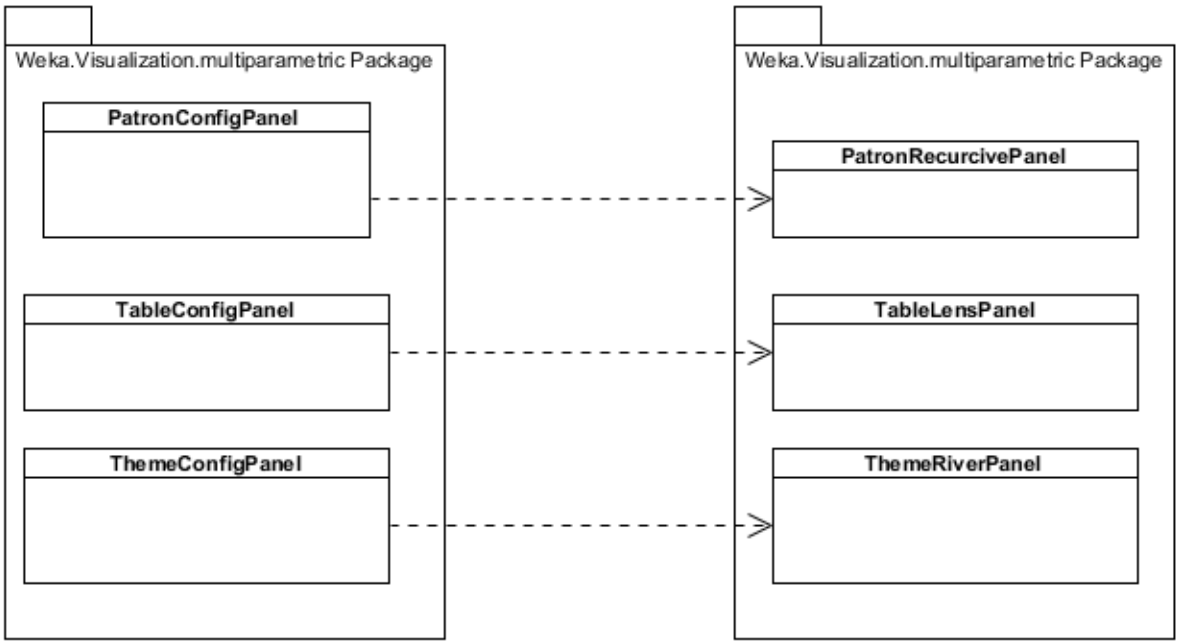

Figura 2-8: Diagrama de clases de interacción del paquete weka.VisualizationEx y weka.VisualizationEx.config

Diagramas de Colaboración

Uno de los objetivos es que el paquete de visualización de datos multiparamétricos sea independiente de Weka y que además sea fácilmente extensible. El siguiente diagrama de colaboración muestra cómo debe usarse la jerarquía de clases para conseguir estos objetivos (figura 2-11).

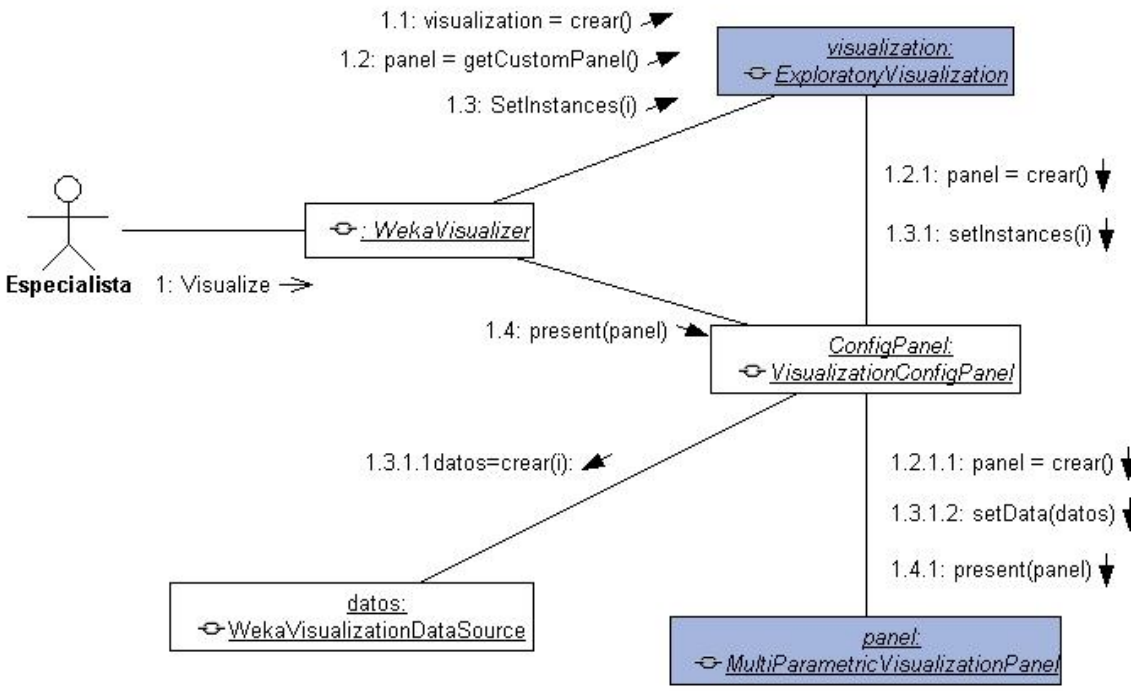

Figura 2-9: Diagrama de colaboración de las clases del paquete de Visualización de datos Multiparamétricos y del paquete de Integración con Weka.

Como puede observarse en este diagrama se utilizan las clases abstractas *ExploratoryVisualization* y *MultiParametricVisualizationPanel*. El diagrama muestra la integración entre el paquete de visualización el paquete de integración que se basa en la utilización de subclases de *ExploratoryVisualization* y *MultiParametricVisualizationPanel.*

2.3 Consideraciones acerca de las técnicas implementadas.

Las técnicas seleccionadas presentan una estructura general, la cual contiene parámetros que pueden variar de una implementación a otra. A continuación se establecen las consideraciones realizadas en cada método para esta versión en particular.

2.3.1 Método de Patrón Recursivo.

La técnica de Patrón Recursivo fue implementada en la clase *PatrónRecursivoPanel*. La implementación difiere en cierta medida de la idea expuesta anteriormente.

La mayor diferencia con la idea original, es que se ha utilizado un esquema, en que cada par observación-dimensión ya no es representada por un simple píxel, sino por un punto o patrón cuyo tamaño puede ser variado interactivamente. Esta pequeña modificación resulta particularmente útil cuando el conjunto de datos es más bien pequeño. Puede notarse que la idea original de utilizar un solo píxel es un caso particular de la variante aquí utilizada.

2.3.2 Método de Theme River

Para esta técnica se desarrolló dos clases, una como *ThemeRiverPanel,* El otro como ,*ThemeConfigPanel.* Esta técnica para su visualización crea un conjunto de puntos basado en los valores de las variables. Con cada conjunto de puntos se crea una línea del río. La unión de estas líneas permite analizar fácilmente el comportamiento de las variables en el tiempo. Gracias a esta forma de representación es posible analizar grandes conjuntos de datos y de elevadas dimensiones.

2.3.3 Método de Table Lens

Para la visualización de manera coordinada se crearon dos clases. La primera *TableLensPanel*está orientada a la visualización sobre el weka. La otra *TableConfigPanel* permite la configuración de Panel .

2.4 Metodología para la adición de una técnica para visualización de datos multiparamétricos.

Uno de las premisas más importantes del proyecto es ofrecer un mecanismo para extender el módulo de visualización, adicionándole nuevas técnicas. Se diseñó una jerarquía de clases orientada a este fin.

La adición de una técnica de visualización al sistema requiere de tres pasos fundamentales. El primero es la creación de una subclase de *MultiParametricVisualizationPanel*,que debe localizarse en el paquete *visualization* y es la encargada de implementar la técnica de visualización científica. El siguiente paso es la creación de una subclase de *VisualizationConfigPanel* que debe localizarse en el paquete *weka.VisualizationEx* y tiene la responsabilidad de manejar la interfaz de usuario que permite la interacción con los parámetros de configuración de la técnica de visualización. El último paso es la creación de una subclase de *ExploratoryVisualization* que debe manejar el proceso de configuración de los parámetros iniciales de la técnica y asignar la fuente de datos que se utilizará, esta clase se colocará en el paquete *weka.VisualizationEx.*

2.4.1 Creación de la subclase de *MultiParametricVisualizationPanel.*

La clase *MultiParametricVisualizationPanel* que hereda de la clase javax.swing.JPanel tiene las operaciones necesarias para acceder a la fuente de datos. Las responsabilidades de cualquier subclase de ella son:

- Realizar la representación visual deseada.
- Manejar la interacción con el usuario a través de los dispositivos de entrada.
- Ofrecer un conjunto de operaciones que permita cambiar los parámetros de la visualización. Estos parámetros pueden ser el color, la geometría y cualquier otra propiedad que afecte la representación.

Si se desea utilizar el módulo de visualización en otros sistemas, las subclases de *MultiParametricVisualizationPanel* deben utilizar solo elementos del paquete de visualización.

2.4.2 Creación de la subclase de *VisualizationConfigPanel.*

La clase *VisualizationConfigPanel*hereda de javax.swing.JPanel. Cada subclase de *VisualizationConfigPanel* está asociada con una subclase de *MultiParametricVisualizationPanel*.

La responsabilidad fundamental de todo *VisualizationConfigPanel* es crear y mantener una técnica de visualización ofreciendo una interfaz para acceder a todos los parámetros de la misma. Otras responsabilidades son:

- Ofrecer operaciones que permitan establecer los parámetros iniciales de la técnica de visualización.
- Transformar la fuente de datos de Weka a la fuente de datos del paquete de visualización.

2.4.3 Creación de la subclase de *ExploratoryVisualization*.

Toda subclase de *ExploratoryVisualization* que se encuentre en el paquete *weka.VisualizationEx* será reconocida por el sistema como una técnica de visualización que puede ser utilizada. Las subclases de *ExploratoryVisualization* pueden opcionalmente implementar la interfaz weka.core.OptionHandler. Es importante aclarar que cada subclase de *ExploratoryVisualization* tiene asociada una única subclase de *VisualizationConfigPanel* y por tanto una sola subclase de *MultiParametricVisualizationPanel.*

Las responsabilidades de las subclases de *ExploratoryVisualization* son:

- Crear y mantener una instancia de *VisualizationConfigPanel.*
- Ofrecer las operaciones necesarias para configurar los parámetros de la técnica de visualización.

Operaciones a implementar.

Toda subclase de *ExploratoryVisualization* debe implementar las siguientes operaciones:

Operación	**Descripción de la operación**
Void set Instances (Instantesi)	Asignar la fuente de datos de la técnica de visualización.
JPanelgetCustomPanel()	Obtener el panel donde se realizará la representación.
InstancesgetInstances()	Obtener la fuente de datos de la técnica de visualización.
StringtoString()	Obtener una descripción de la técnica.

Los tipos de cada uno de los parámetros configurables deben implementar también la interfaz *java.io.Serializable*.

2.5 Implementación de la extensión.

Durante el desarrollo o extensión de un sistema deben tomarse decisiones de implementación como la plataforma objetivo y el lenguaje de desarrollo. Deben definirse también las fases del proceso de desarrollo. En cada proyecto hay que puntualizar en determinadas áreas específicas de la aplicación. En el caso particular de este proyecto debe fijarse la API gráfica a utilizar.

Elección del lenguaje de desarrollo

En este proyecto el objetivo es adicionar técnicas de visualización científica al sistema Weka. Dicha herramienta está construida con el lenguaje Java. Durante el proceso de integración de las técnicas de visualización con el sistema actual inevitablemente tendrán

que realizarse modificaciones en el código fuente de Weka. Esto hace al lenguaje Java la elección obvia para mantener la mayor compatibilidad posible con el sistema actual.

Además de las razones descritas, este proyecto necesita de un lenguaje potente en la implementación de interfaz gráfica lo que se puede conseguir con la plataforma Java 2. Por otra parte la implementación del módulo de visualización científica en este lenguaje, permitiría la fácil integración de este con otros sistemas producto de la flexibilidad y potencialidad de Java.

Elección de la versión de Weka a utilizar como sistema base

En este proyecto se eligió Weka 3-5-2 con las modificaciones implementadas en la UCLV, como la versión a extender.

Elección de la API gráfica.

Las técnicas que se decidió implementar durante el proyecto no requieren de gran complejidad en la API gráfica. En particular no necesitan de una API 3D. Por esto se decidió utilizar la tecnología Java2D que brinda un excelente desempeño y potencialidad para la programación gráfica en 2D. Esta decisión facilita a los usuarios la utilización del sistema ya que no necesitan paquetes adicionales a los de la plataforma Java.

Etapas de desarrollo.

Se definieron las siguientes etapas de desarrollo:

1. Implementar y probar la técnica de Patrón Recursivo.
2. Implementar y probar la técnica de Theme River.
3. Implementar y probar la técnica de Table Lense.

Se utilizó un modelo de prototipos en que las fases se completaron con rapidez de forma que se fueron creando prototipos con una calidad incremental. Con el objetivo de evitar los daños estructurales y las ineficiencias que puede producir el modelo se realizó un riguroso proceso inicial de diseño.

Estrategia de prueba

En él proyecto se utilizó una estrategia de prueba incremental en que cada componente era probado individualmente después de cada ciclo de desarrollo. La estrategia básica utilizada fue probar los elementos de bajo nivel primeramente y luego pasar a los elementos de nivel medio y alto confiando en que los de niveles inferiores ya no contenían errores.

Durante las pruebas se utilizó un amplio rango de datos de entrada que comprendían conjuntos de datos de dominio público. Estos datos cubrían todas las opciones posibles para el proyecto (Anexo 1).

Algunos elementos de la extensión fueron probados con especial rigor pues dependían de teorías matemáticas y eran la base para parte de los componentes. Estos elementos fueron los métodos de optimización que se usaron en el Escalado Multidimensional, que incluyeron el de optimización combinatoria y el de optimización basada en el implementado por Weka.

2.6 Conclusiones parciales.

Las técnicas incluidas en el módulo de visualización ofrecen un amplio número de beneficios ya que se incluyen las tradicionales, con ciertas mejoras, lo que permite mostrar conjuntos de datos con disímiles características.

- El proceso de diseño del nuevo módulo admite la realización de futuras extensiones y modificaciones.

Capítulo 3. Manual de Usuario.

En este capítulo se realiza una presentación al usuario de las nuevas facilidades añadidas al sistema de aprendizaje automático Weka. Se efectúa un análisis detallado de las opciones y modo de uso de cada una de estas nuevas visualizaciones, permitiendo de esta forma al usuario explotar al máximo los beneficios de la extensión realizada.

3.1 Requerimientos del sistema y facilidades añadidas.

La extensión realizada a Weka no trajo como consecuencia nuevos requerimientos para su correcta ejecución, por tanto para la ejecución de la misma, solo se necesita la instalación de una virtual de Java.

Nuevas facilidades del sistema

El sistema Weka está diseñado para usuarios expertos en el dominio. En especial las nuevas técnicas de visualización con que cuenta el sistema, aunque pueden resultar intuitivas es necesario contar con conocimiento previo para aplicarlas de forma adecuada y lograr una correcta comprensión de los resultados que brinda.

La extensión realizada al sistema permite realizar exploración y análisis visual de datos multiparamétricos a través de las técnicas de: Patrón recursivo, Table Lens y Theme River.

3.2 Manual de usuario

Después de elegido la fuente de datos de interés se procede a seleccionar la opción *Visualize*, en la figura 3.1 se observan las opciones que brinda la vista de visualización, la cual está dividida en tres áreas fundamentales.

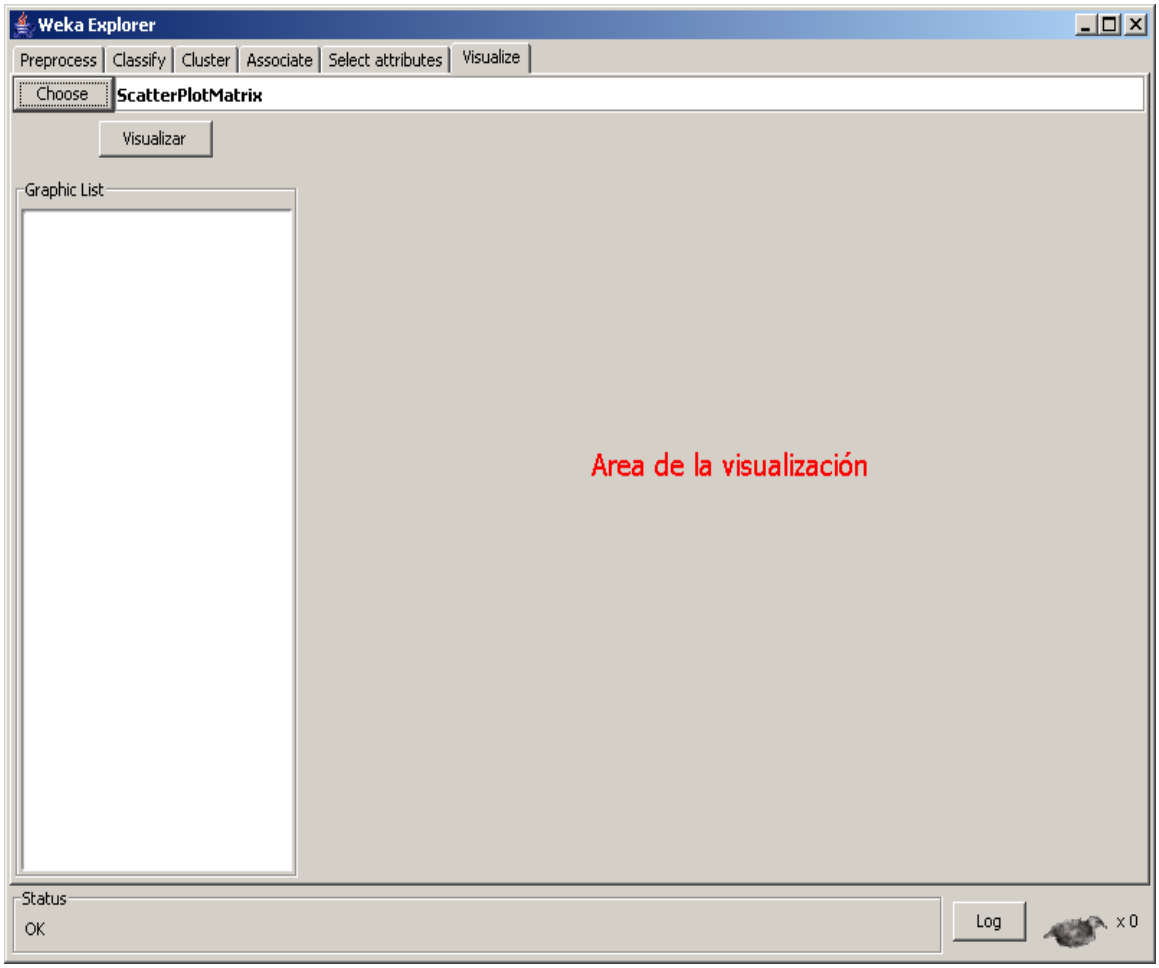

Figura 3-1: Vista de visualización en Weka

El área superior se utiliza para la selección de la técnica de visualización deseada y para configurar los parámetros de la misma. En la opción *Choose* se muestran todas las técnicas de visualización disponibles para la exploración de datos. A la derecha de esta opción y luego de realizarse la elección de la técnica, se brinda la posibilidad de modificar sus parámetros iniciales.

El área inferior izquierda está ocupada por la lista de gráficos (*GraphicList*), donde se añade el nombre de la visualización elegida luego de seleccionar la opción *Visualize*. En esta área se almacenan todas las visualizaciones que se realizan y sus resultados, los cuales pueden ser visualizados eligiendo la técnica deseada.

La última área constituye la mayor de todas y está reservada para mostrar la figura que surge como resultado de la visualización realizada y eventualmente también muestra parámetros propios de la visualización que permiten interactuar con la misma.

Patrón Recursivo.

La técnica de las Patrón Recursivo se encuentra disponible en la opción *Choose* y no presenta parámetros iniciales que se puedan editar. En la siguiente figura se muestra un ejemplo de la aplicación de esta técnica y en el cual nos basaremos para explicar los parámetros de interacción que ofrece.

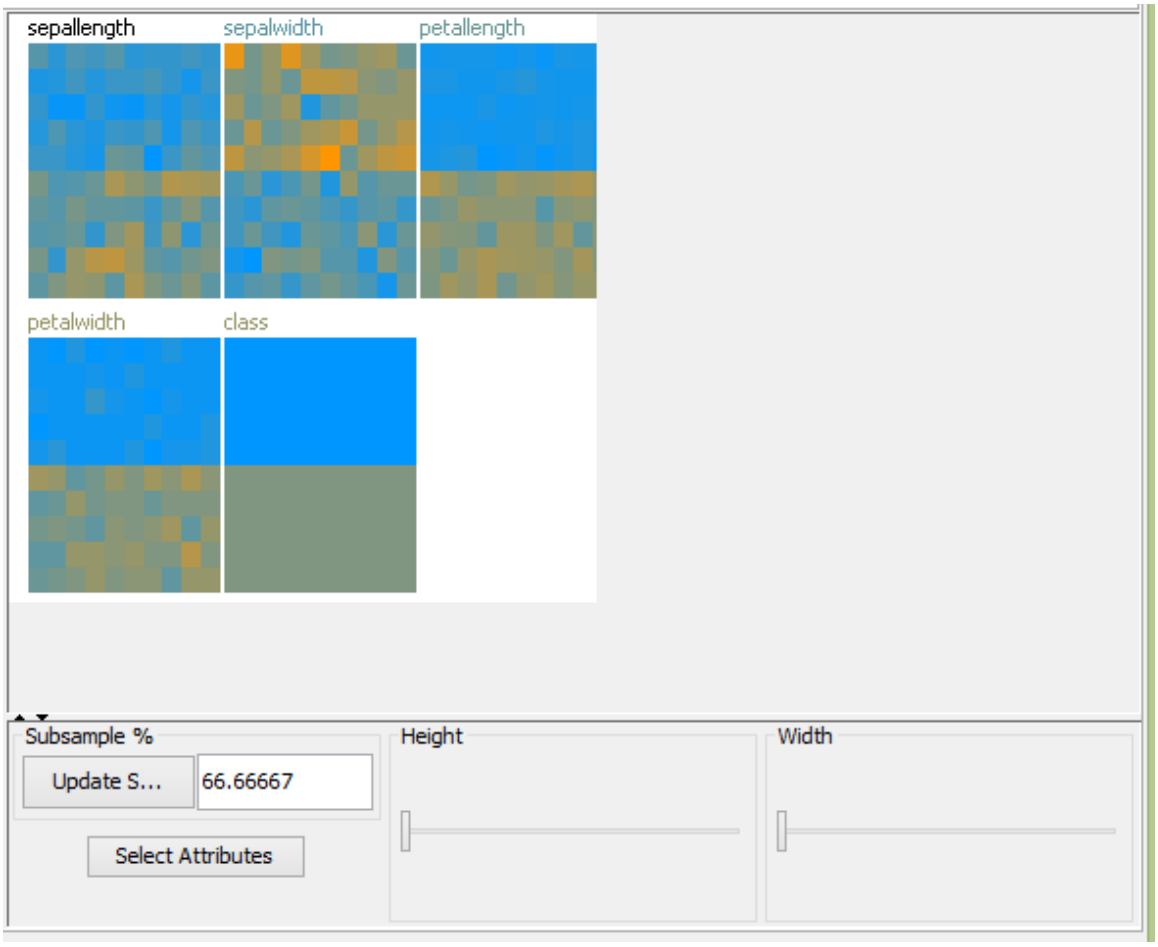

Figura 3-2: Técnica de Patrón Recursivo.

Esta visualización se brinda las opciones de *UpdateSubsample* % y *SelectAttributes*.

La primera opción se refiere a la cantidad de observaciones que se van a mostrar en la imagen. Esta aparece con el nombre de *UpdateSubsample* % y realiza un filtrado de los datos, disminuyendo los mismos a la proporción que se especifica a la derecha de esta elección. En la figura 3-3 puede observarse la opción.

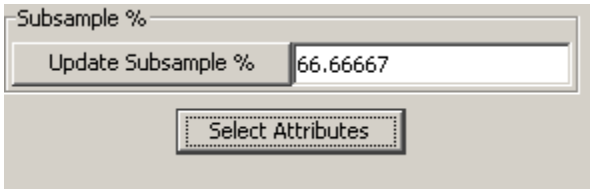

Figura 3-3: Las opciones para seleccionar un subconjunto de trabajo

Otra posible elección para el usuario se encuentra en *SelectAttributes*(figura 3-4); al elegir esta opción aparece la vista que se ofrece a continuación, la cual permite elegir los atributos que se desean tomar en cuenta para la construcción de la imagen. De esta manera el usuario puede concentrarse en atributos de interés.

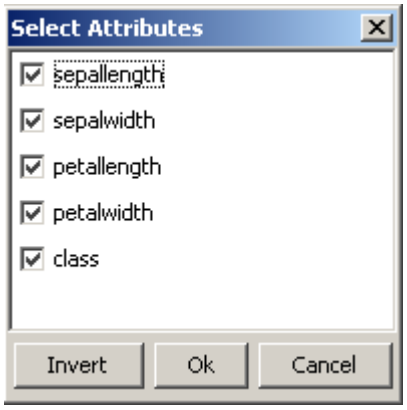

Figura 3-4: Vista que permite seleccionar los atributos de interés.

Aunque esta técnica está especialmente diseñada para mostrar un gran volumen de datos, en caso de un conjunto extremadamente grande de los mismos sí se debe considerar la reducción de las observaciones. La cantidad de atributos en cambio sí suele influir mucho más en la expresividad de la representación, pues un número elevado de estos disminuye

en gran medida el espacio de representación de cada atributo, en cuyo caso lo mejor es realizar una selección de los mismos.

Otra posible interacción que se brinda es la elección del atributo mediante el cual se realiza el ordenamiento de las observaciones. Por defecto las mismas aparecen ordenadas por el atributo de la clase, pero el usuario puede elegir cualquier otro con solo seleccionar el atributo deseado en la imagen. En caso de que el atributo seleccionado sea nominal se utilizará para ordenar la codificación que utiliza Weka donde se establece un número para cada valor posible.

Entre los parámetros de interacción que se permiten se encuentra además una escala situada en la esquina inferior derecha la cual permite al usuario controlar el tamaño de los puntos que conforman la imagen.

Theme River

El orden de las variables puede ser cambiado pero inicialmente comienzan en el siguiente orden temperatura, temperatura mínima, temperatura máxima, nubosidad, presión, humedad, vapores, lluvias y escarcha. Cuenta con una leyenda de colores en la parte inferior que permite al usuario saber que color corresponde a cada variable y entre otros tiene la opción de cambiar el tamaño de rio.

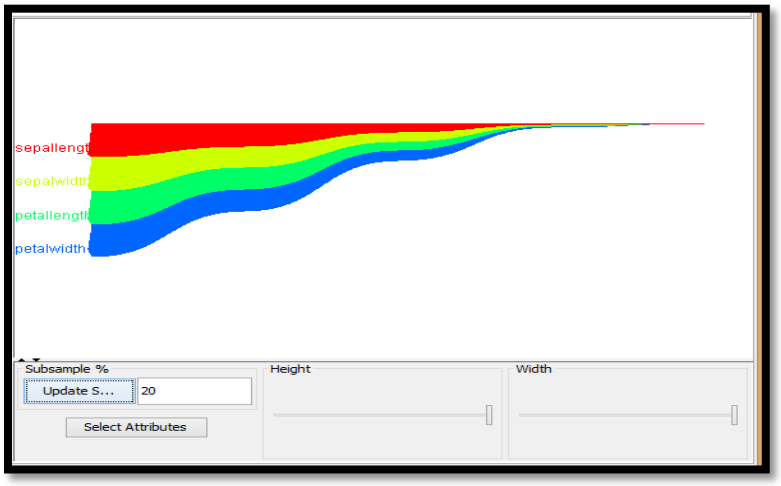

Figura 3-5: Técnica de Theme River.

Table Lens

Los nombres de las variables son mostrados en la parte inferior de cada columna de forma tal que constituye una guía para el usuario y se permite ordenar las observaciones respecto a cada atributo.

Figura 3-6: Técnica de Patrón Recursivo.

3.3 Conclusiones Parciales.

- La utilización de la extensión de Weka requiere de conocimiento previo de los usuarios en el área de visualización científica.
- La interacción con las técnicas de visualización es un proceso sencillo para el usuario.
- El sistema tiene pocos requerimientos de hardware y software.

Conclusions finales.

- Con este módulo de visualización científica se incrementan las potencialidades de Weka para la exploración y análisis de datos iniciales, al adicionarle las técnicas de visualización Patrón recursivo, Theme River y Table Lens.
- El módulo implementado es además independiente de Weka por lo que puede ser utilizado por otros sistemas.

Recomendaciones

Luego de concluida la investigación se recomienda:

- Adicionar nuevas técnicas de visualización para la exploración y el análisis de los conjuntos de datos que conviertan a Weka en una herramienta más poderosa.
- Extender y modificar las técnicas implementadas agregándole nuevas funcionalidades, entre ellas:
    - Adicionar nuevas formas de proyectar los datos en la técnica de visualización basada en proyecciones, usando PCA, SOM y GTM.
    - Adicionar nuevos tipos de iconos como los Rostros de Chernoff a la técnica de visualización de iconos.
    - Adicionar otros métodos de posicionamiento de los iconos.
- Realizar un estudio para determinar las formas en que las técnicas de visualización podrían sugerir a los especialistas qué métodos de aprendizaje automático utilizar.

Referencias Bibliográficas.

Aarts, E. and Korst J. 1989. *Simulated Annealing and Boltzmann machines: a Stochastic Approach to Combinatorial Optimization and Neural Computing.*Anchor Press.

Andrews, K. 2007. *Information Visualization*. [cited; Available from: http://courses.iicm.edu/ivis/].

Ankerst, M., D.A. Keim, and H.-P. Kriegel,. 1996. *'Circle Segments': A Technique for Visually Exploring Large Multidimensional DataSet*, in *Visualization '96*.: San Francisco, CA.

Cui, Q., M.O. Ward, and E.A. Rundensteiner,. 2007. *Enhancing Scatterplot Matrices for Data with Ordering or Spatial Attributes*.

Davidson, I ., 2000. *Visualizing Clustering Results*.

Eick, S.G ,. 2000. *Visualizing Multi-Dimensional Data.* ACM SIGGRAPH. **34 No. 1**.

Fernández, E., M. Giorgi, and T. Laurenzo,. 2003. *VIEG, una herramienta para la Visualización de Información Estructurada mediante Grafos*.

Gallagher, R.S. 1998. *Computer Visualization: Graphics Techniques for Engineering and Scientific Analysis*.

Grande, S. 2014. *e-interactive*. Obtenido de http://www.e-interactive.es/blog/visualizacion-de-datos-10-potentes-herramientas-que-debes-conocer/#axzz3VM9gAmVt

Hansen, C.D. and C.R. Johnson ,. 2005. *The visualization handbook*. Elsevier.

Han, J. and N. Cercone, *RuleViz.* 1999*: A Model for Visualizing Knowledge Discovery Process*.

Keim, D.A., 2002 *Information Visualization and Visual Data Mining.* IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER GRAPHICS, **7, NO. 1**.

Lorenzo, M.M.G., 2000 et al., *RedesNeuronalesArtificiales*. , Santa Clara.

MatLab , 2014.

Maniyar, D.M. and I.T. Nabney, 2008 , *Visual Data Mining using Principled Projection Algorithms and Information Visualization Techniques*.

Martig, S.R. and S.M. Castro, 2000 *Visualización de Grafos*.

Matías, H. and L.I. Araugo, 2006, *Extensiones al Ambiente de Aprendizaje Automatizado Weka*. , UCLV: Santa Clara.

Mazza, Riccardo. 2011. Introduction to Information Visualization, Springer publishingCompany

Morell, A. and C. Perez. 2006, Biblioteca de módulos de visualización de fluidos para Open DX, Ciego de Ávila, .

Nauck, D., F. Klawonn, and R. Kruse. 1997 ,*Foundations of Neuro-Fuzzy Systems*.

O.Ward, M., 2002, *A Taxonomy of Glyp Placement Strategies for Multidimensional Data Visualization*, in *Computer Science Department*. Worcester Polytechnic Institute.

Peng, W., 2005, *Clutter-Based Dimension Reordering in Multi-Dimensional Data Visualization*, in *Computer Science*, WORCESTER POLYTECHNIC INSTITUTE.

Pérez Risquet, C. and J.C. Ortega Camacho. 2004, *Modelación de datos para la visualización científica.*.

Petoer pirolli, R. r. 2013, *Table Lens as a tool for Making sens for data.*Obtenido de parc.xerox.com: parc.xerox.com.

Ramana, S. C. 2013 , *parc.xerox.com*

Salgado Milán, 2003 E. *Visualization Techniques*. [cited.

StatSoft, 2003 *STATISTICA*.

Streit, Marc, Rupert C Ecker, KatjaOsterreicher, Georg E Steiner, Horst Bischof, Christine Bangert,Tamara Kopp, y RaduRogojanu. 2006. 3d parallel coordinate systems—a new data visualization method in the context of microscopy-based multicolor tissue cytometry, Cytometry Part A, 69(7),601–611.

Theisel, 2000, H. *Scientific Visualization*. [cited.

Ware, C., *Information Visualization. Perception for Design*. 2007: Morgan Kaufmann.

Ward, M. O. (2008). Multivariate Data Glyphs: Principles and Practice. Handbook of Data Visualization. C.-h. Chen, W. Härdle and A. Unwin, Springer-Verlag Berlin Heidelberg**:** 179-199.

Ramana, S. C. 2013. parc.xerox.com

Xie, Z., et al., *Exploratory Visualization of Multivariate Data with Variable Quality*. 2010.

Yang, J., et al., *Value and Relation Display for Interactive Exploration of High Dimensional Datasets.* 2010.

Yin, H., *Nonlinear Multidimensional Data Projection and Visualisation.* 2002.

Yin, K. and I. Davidson, *Further Applications of a Particle Visualization Framework.* 2008.