10009 ENSEÑANZA DE LA MINERÍA DE DATOS Y EL IMPACTO DE LAS HERRAMIENTAS DE SOFTWARE UTILIZADAS

Laura Lanzarini[(1)(2)], Augusto Villa Monte[(1)(3)]

[(1)]Instituto de Investigación en Informática LIDI

Facultad de Informática

Universidad Nacional de La Plata

La Plata, Buenos Aires, Argentina

[(2)]laural@lidi.info.unlp.edu.ar

[(3)]avillamonte@lidi.info.unlp.edu.ar

Resumen: La Minería de Datos reúne a las técnicas que, a partir de datos almacenados en grandes bases de datos, poseen la capacidad de adquirir conocimiento nuevo, novedoso y potencialmente útil. Permite obtener modelos predictivos y/o descriptivos que ayudan a la toma de decisiones.
El mayor obstáculo que enfrentan los alumnos durante todo el proceso de aprendizaje radica en el desconocimiento de la solución del problema planteado. A diferencia del enfoque convencional, se dispone de ejemplos o muestras del problema de las cuales debe extraerse la solución. No se plantean hipótesis a verificar sino que las relaciones entre los datos disponibles deben surgir en forma automática.
Por otro lado, las técnicas Minería de Datos hacen uso de una gran cantidad de conceptos vistos en asignaturas previas que no sólo pertenecen al área de Programación sino también de la matemática ya que se requieren conocimientos de Algebra y Cálculo Vectorial. Los autores de este trabajo son docentes de la asignatura “Minería de Datos utilizando Sistemas Inteligentes” que se dicta en la Facultad de Informática de la UNLP desde 2012 y cuentan con una rica experiencia en el tema que puede ser de interés para la comunidad educativa, tanto en lo referido al enfoque utilizado para dictar la asignatura como a las herramientas de software utilizadas.

Palabras clave: SISTEMAS INTELIGENTES, ESTRATEGIAS ADAPTATIVAS, MINERÍA DE DATOS, SELECCIÓN DE ATRIBUTOS.

1. Introducción

En la actualidad, son numerosas las áreas interesadas en extraer conocimiento útil y novedoso a partir de información almacenada. La tecnología actual permite registrar todo tipo de procesos, en variados formatos y almacenarlo en forma local o subirlo a la nube con suma facilidad. Los datos se encuentran disponibles y contienen el registro de todo lo ocurrido. Esa información es producto de decisiones que fueron tomadas en distintos instantes de tiempo. Analizar los hechos pasados permite comprender los criterios utilizados y asociarlos con los resultados obtenidos ya sea que hayan sido positivos o negativos.

La Minería de Datos, una de las etapas más importantes del proceso de Extracción de Conocimiento o KDD (por su nombre en inglés *Knowledge Discovery in*

Databases), cuenta con un conjunto de técnicas capaces de modelizar y resumir estos datos históricos, facilitando su comprensión y ayudando a la toma de decisiones. Su objetivo es generar una representación alternativa de la información que deje de manifiesto las relaciones existentes en ellos. Luego, a partir de su análisis e interpretación se podrá comprender, por medio de la razón, la naturaleza, cualidades y relaciones de los datos históricos, es decir, se podrá obtener conocimiento.

El proceso de KDD ha sido descripto por varios autores con distinto nivel de detalle [1]. El consenso general reconoce al menos tres etapas: la primera tiene que ver con la manera en que se recolecta y analiza la información con la que se va a trabajar, la segunda se refiere a la construcción del modelo e incluye las técnicas de Minería de Datos y la última consiste del análisis e interpretación del modelo obtenido y eventualmente su comunicación a quienes deben tomar decisiones.

En la mayoría de los casos obtener resultados satisfactorios implica revisar muchas de las acciones realizadas incluso desde el inicio del proceso. Generalmente la información fue recolectada antes de que apareciera en escena el especialista en Minería de Datos. Es decir que ya se tomaron decisiones referidas a qué información debe relevarse y con qué nivel de detalle debe realizarse. Es importante reconocer que no puede representarse o modelizarse lo que no se ha registrado. La información no puede generarse automáticamente; sólo puede transformarse para extraer a partir de ella las relaciones de interés. La calidad de la información recolectada no sólo depende de la precisión empleada en la digitalización sino en su capacidad para describir el problema a resolver.

Este artículo resume los aspectos centrales que son tenidos en cuenta en el dictado de la asignatura “Minería de Datos utilizando Sistemas Inteligentes” en la Facultad de Informática de la UNLP. Se trata de una materia optativa correspondiente al 5to. año de las carreras Licenciatura en Informática, Licenciatura en Sistemas e Ingeniería en Computación.

La organización de este trabajo es la siguiente: la sección 2 describe los problemas habituales que se le presentan a los estudiantes que poseen pocos conocimientos de Cálculo Vectorial, la sección 3 presenta algunos de los casos reales analizados en el curso para motivar a los alumnos a integrar distintos conceptos, la sección 4 describe brevemente las herramientas de software utilizadas en clase y finalmente la sección 5 expone las conclusiones y algunas líneas de trabajo futuras.

2. Representación de la información

Para trabajar en Minería de Datos el primer punto a tener en cuenta es el grado de protagonismo que tiene la información disponible. Es difícil entender, para quien ha realizado una carrera informática, que no se dispone de un algoritmo que resuelva el problema planteado. Esto contradice el enfoque habitual que sugiere que, independientemente del paradigma de programación utilizado, para resolver un problema a través de una aplicación de software, es preciso disponer del algoritmo correspondiente.

Cuando se trabaja con técnicas de Minería de Datos el proceso se invierte y son los datos los que toman el rol principal. Son las distintas *“muestras”* del problema a resolver lo que se busca generalizar y lo que se desconoce es la solución del

problema. Entonces, ¿cómo hacer para escribir una aplicación si se desconoce el algoritmo a seguir?

La respuesta está en que no se trata de manipular la información de una forma predefinida con el objetivo de corroborar una hipótesis previa sino que se buscan patrones o relaciones en los datos que permitan o bien describir una situación o bien predecir el resultado ante nuevos casos.

Es en este punto en el que el concepto de *patrón*, entendido como una regularidad presente en los datos, se convierte en un concepto fundamental [2].

Dado que se trata de un concepto que requiere cierto tiempo de maduración, desde la cátedra se ha desarrollado un Objeto de Aprendizaje (OA) para que el alumno analice patrones utilizando distintas técnicas de visualización.

El OA está formado por cuatro partes:

- Un repaso que enfatiza la relación entre el objeto o elemento del problema que se está caracterizando y la información que queda registrada.

- El contenido central que motiva el OA: el análisis de los datos a través de la identificación de patrones.

- Actividades prácticas utilizando otros juegos de datos que repiten las técnicas utilizadas.

- Una actividad de autoevaluación.

Luego de comprender la importancia que tienen los datos en todo este proceso aparece una nueva pregunta: ¿Cuántos datos son necesarios para poder hacer Minería de Datos? Quienes recién se inician en este tema consideran que es preciso contar con grandes volúmenes de información cuando en realidad sólo es preciso tener una proporción de ejemplos representativos del problema a resolver. Es importante que los ejemplos cubran todas las situaciones posibles con la misma distribución de frecuencia con la que ocurren en la realidad. Este aspecto puede trabajarse en el aula construyendo distintos subconjuntos de ejemplos y analizando en cada caso los cambios que se producen en la distribución de ejemplos correspondientes a cada situación posible del problema planteado.

3. Tipos de problemas a resolver

Como se mencionó previamente, por lo general, cuando el especialista en Minería de Datos se incorpora al grupo de trabajo, la información ya ha sido recolectada. Esto implica que ya se decidió en una instancia previa cuáles características del problema debían ser relevadas y cuáles no. La calidad de la información disponible condicionará la respuesta a obtener. Esta situación en el aula se refleja a través de conjuntos de datos propuestos por la cátedra con diferentes características tanto en lo que se refiere a los tipos de atributos a utilizar como a la calidad de los datos (datos faltantes, inconsistencias, falta de uniformidad en la representación, etc.).

La metodología de procesamiento utilizada en clase y que permite a los alumnos identificar la situación con la que se está trabajando, parte de identificar primero el tipo de problema a resolver.

Hay dos tipos de problemas que pueden ser resueltos: descriptivo y predictivo. El primero tiene que ver con hallar una representación que explique las características relevantes de los distintos grupos presentes en los datos. El resultado obtenido para este tipo de problema serán perfiles generales con capacidad para describir las relaciones más representativas siendo sumamente útiles para comprender el estado de situación y actuar en consecuencia. Sin embargo, cuando se necesita dar una respuesta ante una situación nueva, se está frente a un problema predictivo. El tratamiento de la información cambia radicalmente según el tipo del cual se trate.

3.1 Problemas Descriptivos

Para dar respuesta a un problema descriptivo, las técnicas de agrupamiento resultan de suma utilidad. En esta dirección y con el objetivo de analizar en clase la resolución de problemas reales, se estudian distintos casos pertenecientes al área de la Minería de Datos Educativa en los que han participado los docentes de la asignatura. A continuación se menciona brevemente, a modo de ejemplo, el análisis realizado de la información académica de los alumnos que estudian carreras en Informática en tres Universidades Nacionales distintas: la Universidad Nacional de La Plata, la Universidad Tecnológica Nacional Facultad Regional La Plata y la Universidad Nacional de Río Negro. Más allá de las particularidades de cada uno, el tema común a todos los casos se relaciona con el desgranamiento académico y la deserción universitaria.

- El primer caso analizado fue la información de los alumnos de la carrera Licenciatura en Sistemas de la Sede Atlántida de la UNRN [3]. En esa oportunidad, se utilizó un método basado en proyecciones para seleccionar las características de los estudiantes y se aplicó un método de agrupamiento para determinar el perfil de los alumnos que abandonaban la carrera. Como conclusión relevante se detectó que existía una relación inversa entre el desempeño académico de los alumnos y la cantidad de horas que trabajaban y que la cantidad de alumnos con necesidades económicas era significativa. En este caso la recomendación fue arbitrar los mecanismos para ofrecer a los alumnos, según su perfil, becas de comida, transporte, laborales o de estudio con el objetivo de reducir sus necesidades e incrementar su dedicación a la carrera.

- Los resultados obtenidos para los alumnos de la UNLP fueron totalmente diferentes [4] [5]. Para este caso se representó el avance académico generando 5 atributos nuevos con la cantidad acumulada de asignaturas aprobadas por año. Luego a través de distintas visualizaciones se comprobó que los alumnos con mejores condiciones económicas tenían un desempeño académico entre malo y regular mientras que los que trabajan o manifestaban tener intenciones de hacerlo mostraban un mayor compromiso con sus estudios y aunque demoraban unos años más, tenían mayor chance de finalizarlos. También se observó que los alumnos que lograban mantener su ritmo académico durante el segundo año de estudio terminaban la carrera. Es entre primer y segundo año que se define la continuidad del alumno en la Facultad de Informática. En este caso la recomendación fue reforzar las tutorías a los alumnos entre el segundo

cuatrimestre del primer año y todo el segundo año. Ese es el período de mayor vulnerabilidad para los estudiantes y donde parecen tener la mayor cantidad de dificultades en sus estudios.

- Con respecto los alumnos de la UTN- FRLP los resultados fueron similares pero la población de alumnos tiene mayor edad y la proporción de alumnos que trabajan es mayor por lo que la relación entre trabajo y avance académico se acentúa [6].

Utilizando la experiencia en este tipo de procesamiento se llevan al aula conjuntos de datos que permiten a los estudiantes realizar el mismo recorrido en lo que se refiere a las técnicas a utilizar y obtener resultados que les permitan reconocer patrones. Un ejemplo es la información correspondiente a la encuesta realizada por la Fundación Sadosky a varios alumnos de colegios secundarios acerca de si estudiarían o no una carrera relacionada con informática [7]. En este caso, debe darse a los alumnos una versión preprocesada de la información donde sólo aparezcan las características relevantes. Para más detalles sobre cómo construir esta vista minable puede consultarse [8] donde la selección de atributos a través de filtros permitió establecer que sólo el 10% de la información relevada era relevante a la hora de obtener un perfil general. Este resultado no es un tema menor porque no sólo reduce el tamaño de la información a almacenar sino que facilita la realización de futuras encuestas y agiliza la obtención de los perfiles buscados al reducir el tiempo de cálculo.

Resuelto el problema de la obtención de la información sobre la cual se va a trabajar, los alumnos pueden utilizar las técnicas convencionales de Minería de Datos y obtener respuestas interesantes para un problema real. En el caso particular de la encuesta de la Fundación Sadosky, los resultados obtenidos confirman la importancia de introducir a los jóvenes en las diferentes funcionalidades que puede tener una computadora. Es decir que, si se busca incrementar el interés de los jóvenes por estudiar informática se debería, desde los colegios secundarios propiciar la creación de espacios y actividades tendientes a acercar a los jóvenes a la computadora de maneras no convencionales con la intención de ampliar su rango de aplicación; por ejemplo, a través de la música, el procesamiento de imágenes, la robótica, etc. También, la utilización de juegos y la predisposición a configurar y administrar las aplicaciones de software parece ser un indicio fuerte acerca de la tendencia de los alumnos a estudiar carreras informáticas. En ese sentido, proponer en los colegios la realización de talleres relacionados con estos temas debería incrementar el interés en la temática.

Además de las técnicas de agrupamiento, en la asignatura se enseñan las Reglas de Asociación como herramienta para hallar relaciones entre las características relevadas. Además de trabajar en clase con datos provenientes de repositorios como el UCI [9], se muestra, como aplicación real, el método presentado en [10] por medio del cual es posible identificar los conjuntos de ítems frecuentes a partir de una red neuronal SOM difusa. En dicho trabajo el objetivo fue analizar los e-mails correspondientes a cursos realizados a través de una plataforma de educación a distancia. Con este tipo de análisis se obtuvieron los grupos de palabras más relacionados; es decir, se determinaron los temas de conversación más habituales.

3.2 Problemas Predictivos

Describir y predecir son dos tareas muy diferentes ya que la primera tiene por objetivo principal organizar la información existente mientras que la segunda debe aprender un criterio para dar una respuesta. Esto último implica que los ejemplos disponibles deben llevar registrada la respuesta esperada. Luego, será el criterio con el cual fueron obtenidas esas respuestas lo que la técnica de Minería de Datos buscará imitar.

Es decir que la resolución de problemas predictivos requiere de información "etiquetada" y las técnicas a aplicar utilizan aprendizaje supervisado.

Nuevamente en este caso, la cátedra cuenta con situaciones reales previamente resueltas que se analizan en clase. Las técnicas utilizadas son: árboles de decisión y reglas de clasificación, redes neuronales y técnicas de optimización. A continuación se detallan algunos de los casos analizados en clase luego de haber estudiado la teoría de base. Las soluciones propuestas se basan en la combinación de técnicas con el objetivo de para mejorar su desempeño. Se analizan en clase las siguientes situaciones:

- En [11] se implementa un reconocedor probabilístico basado en un método de votación capaz de identificar al locutor a partir de su señal de voz. Las señales de voz son analizadas luego de ser convertidas en sus correspondientes coeficientes ceptrales y la red neuronal utilizada es una adaptación de la red neuronal competitiva SOM para permitir que reconozca patrones formados por varios ejemplos.

- En [12] se describe la construcción de otro reconocedor biométrico capaz de identificar a una persona por la imagen de su rostro. La identificación de una persona por la imagen de su rostro es un proceso que consiste en comparar una imagen del sujeto de interés con un conjunto de imágenes almacenadas previamente en una base de datos. Para brindar más flexibilidad al reconocedor, la base suele contener varias imágenes de una misma persona buscando modelizar las distintas situaciones que pueden presentarse al momento de capturar una nueva imagen. Esto incluye expresiones faciales, cambios de posición de la cabeza, cambios de escala, etc. La información a buscar en la base de imágenes no es directamente la imagen original sino una caracterización de ella. En [13] se propone utilizar una técnica de optimización para seleccionar los vectores SIFT más representativos logrando no sólo una reducción en los falsos positivos sino también en el tiempo de cómputo requerido para el procesamiento y para el almacenamiento de la base de imágenes.

- También se presentan métodos originales para extraer reglas de clasificación utilizando una red SOM para inicializar una técnica de optimización poblacional. En [14] se combinan dos alternativas de inicialización con dos formas distintas de extraer reglas según si la técnica de optimización población es de tamaño fijo o de tamaño variable. Los resultados de su aplicación a dos bases de datos reales con información de crédito para consumo puede verse en [15].

Como puede observarse, si bien todos son problemas predictivos, las áreas de aplicación son diversas siendo las redes neuronales y las técnicas de optimización

las estrategias de base más utilizadas. La Minería de Datos engloba todas estas técnicas y el estudio de este tipo de soluciones amplia la visión del alumno más allá del procesamiento de datos de repositorio.

4. Software para Minería de Datos

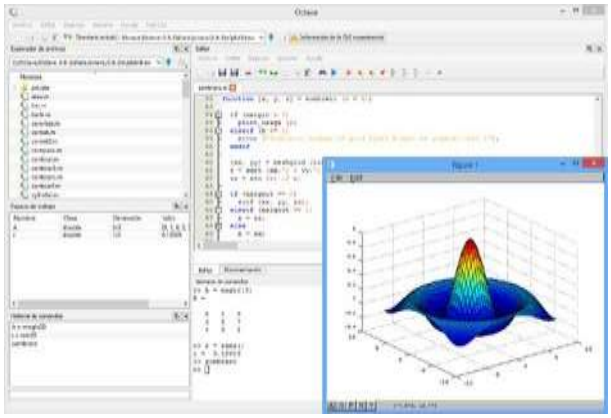
En el aula, para resolver ambos tipos de problemas planteados en la sección anterior, es preciso utilizar alguna herramienta informática que asista al alumno durante el desarrollo y facilite la interpretación de los resultados obtenidos a través de visualizaciones y cálculos de distintas métricas de performance.

En la actualidad, existen varias aplicaciones informáticas para hacer Minería de Datos. Estas aplicaciones de software tienen como objetivo proveer recursos, herramientas y algoritmos para la realización de cada una de las etapas del proceso de KDD. Sin embargo, entran en juego distintos factores al momento de seleccionar una de ellas para utilizarla en el aula. Se debe tener en cuenta: la licencia de distribución, la interfaz gráfica, la capacidad de integración, la velocidad de procesamiento y la biblioteca de funciones.

En este artículo abordaremos brevemente varias aplicaciones de escritorio gratuitas cuyos instaladores pueden descargarse desde Internet.

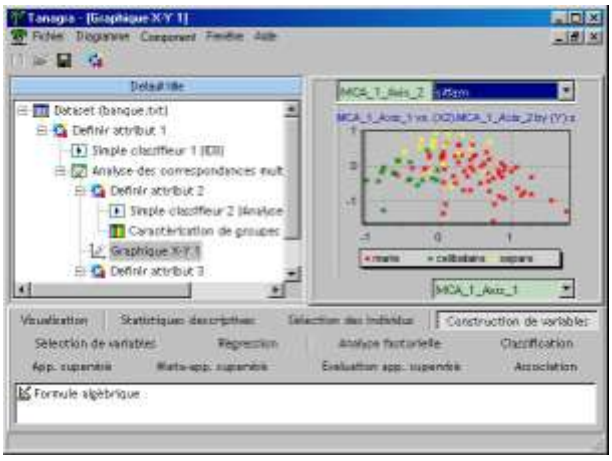
4.1 Octave [16]

GNU Octave es un software gratuito con un lenguaje de programación de alto nivel destinado principalmente a resolver problemas matemáticos a través de cálculos numéricos. Se comenzó a desarrollar a principios de los 90 y desde entonces mejoró versión tras versión. En una de sus últimas versiones incorporó un entorno gráfico y actualmente tiene una gran comunidad que lo promueve. Desarrollado en C++ está disponible para todas las plataformas. Su lenguaje interpretado permite codificar en scripts y prototipar rápidamente soluciones propias a distintos problemas utilizando una sintaxis matricial. Octave es un lenguaje estructurado similar a C que soporta muchas de sus funciones, así como distintas llamadas propias de sistemas UNIX. Además, provee una amplia gama de toolbox con algoritmos ya implementados. Puede ser utilizado a través de la línea de comandos o mediante su interfaz gráfica. Es la principal alternativa a Matlab, un software con similares características pero con licencia propietaria.



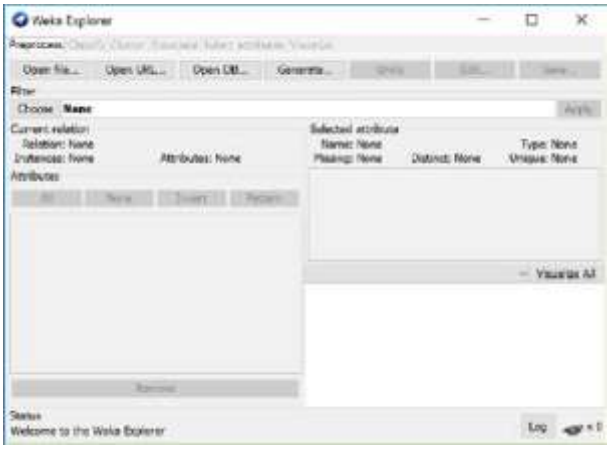
4.2 Tanagra [17]

Esta es una herramienta gratuita desarrollada en 2003 por Ricco Rakotomalala en la Universidad de Lumière en Francia. Nació como proyecto académico con fines de investigación para suceder a Spina, una herramienta previa con menor funcionalidad. Permite realizar varias de las tareas que implica la Minería de Datos. Tiene una interfaz algo arcaica y su funcionalidad no puede extenderse ni utilizarse en otro entorno.



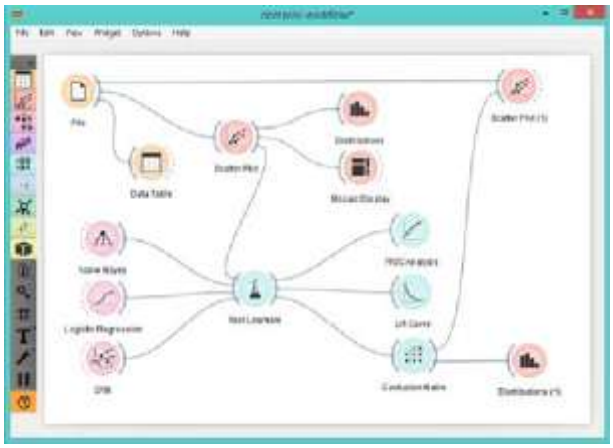
4.3 Weka [18]

Este es un software libre de Minería de Datos multiplataforma desarrollado completamente en Java por la Universidad de Waikato. Provee una amplia colección de algoritmos que pueden invocarse desde proyectos externos o ejecutarse a través de su interfaz. Utiliza principalmente un formato de archivos propio llamado arff (por sus siglas en inglés de Attribute-Relation File Format). Si bien su interfaz gráfica es fácil de utilizar, cuando un proceso aplica varios operadores en cascada no es posible debuguearlo fácilmente.



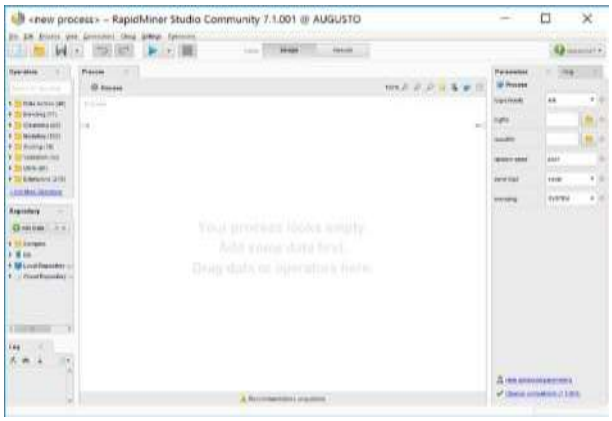
4.4 Orange [19]

Este es un software open source desarrollado en C++ por la Facultad de Informática de la Universidad de Ljubljana. Provee un conjunto de algoritmos que, al igual que Weka, pueden manipularse a través de su entorno gráfico o desde programas pero escritos en Python. Desde su aparición ha mejorado mucho su apariencia visual.



4.5 RapidMiner [20]

RapidMiner es una plataforma para ciencia de datos desarrollada por la compañía con el mismo nombre. Su primera versión fue desarrollada por el Departamento de Inteligencia Artificial de la Universidad de Dortmund en 2001. Llegó a estar en los primeros puestos entre las aplicaciones de Minería de Datos más utilizadas para problemas reales. Provee un ambiente integrado con una gran variedad de operadores y está disponible para todos los sistemas operativos. Tiene buenos resultados tanto en aplicaciones comerciales como en investigación y educación. Abarca una gran cantidad de tareas cubriendo correctamente todos los pasos del proceso de KDD incluyendo la importación de datos, su preparación, la visualización los mismos, la obtención de todo tipo de modelos y la validación de estos. Además tiene muchas extensiones que se le pueden instalar, entre las que se encuentran: la extensión Weka que permite ejecutar sus algoritmos dentro de RapidMiner utilizando los operadores correspondientes y la extensión Octave que permite ejecutar código en dicho lenguaje dentro de aquel. Cabe destacar que posee una interfaz muy bien diseñada que aprovecha la pantalla para permitir buscar los operadores, arrastrarlos al área de trabajo, configurarlos y conectarlos entre sí, pudiendo ejecutar todo el proceso o hacerlo paso por paso visualizando su salida. Existe una versión paga de esta herramienta que permite entre otras cosas procesar una mayor cantidad de registros y leer datos conectándose con distintos motores de base de datos.



4.6 Experiencia en el aula

Aunque existe software como Matlab, SPSS y SAS, reconocidos y aceptados en el ámbito empresarial, no se utilizan en el aula por poseer licencias propietarias. Si bien la institución podría adquirir las licencias correspondientes, no sería posible para el alumno trabajar desde su domicilio. Por esta razón, la cátedra se inclina por alternativas gratuitas que compitan con los productos comerciales.

"Minería de Datos utilizando Sistemas Inteligentes" ya lleva varios años de dictado. En la primera edición se utilizó, por un lado, Weka para la construcción de árboles y reglas, y por otro, Octave para programar los algoritmos de Redes Neuronales.

Para los siguientes dictados Weka fue reemplazado por RapidMiner ya que en él se podían seguir utilizando los algoritmos originales de Weka y además disponía de una amplia gama de operadores. También su interfaz resultaba más visual que Weka y permitía visualizar todo el proceso de KDD que el alumno había desarrollado, pudiendo ejecutarlo operador por operador. Además, los alumnos podían continuar desarrollando sus programas en Octave e incorporarlos en RapidMiner con el operador correspondiente [21].

Para hacerle frente a la dificultad que presentaban los alumnos al momento de programar con un lenguaje de programación matricial, para los siguientes dictados se decidió cambiar Octave por Java. Este cambio tuvo que ver con el nivel de familiarización de los alumnos con el paradigma orientado a objetos y en especial con el lenguaje de programación Java. Se pensó que facilitaría que los alumnos pudieran programar sus propios algoritmos y ejecutarlos de dos maneras: invocando desde un proyecto Java propio los algoritmos de RapidMiner o extendiendo la funcionalidad de RapidMiner con sus propios algoritmos programados en Java.

En todos los casos los docentes prepararon los instructivos necesarios para que el alumno pudiera llevar a cabo la integración de las herramientas. Sin embargo, programar cada operador en Java requería demasiado tiempo para una asignatura cuatrimestral. Por esta razón, actualmente en la asignatura los alumnos utilizan únicamente los operadores de RapidMiner Studio 7.4 (Starter Edition). Esto no permite apreciar muchos de los detalles que se advierten al programar detalladamente cada técnica pero brinda al alumno un panorama más amplio de los distintos tipos de técnicas permitiéndole focalizar en las que resulten de su interés.

5. Conclusiones

La extracción de conocimiento a partir de la información disponible es un tema de sumo interés en la actualidad que está muy lejos de ser resuelto de manera automática. Aún no es posible desarrollar una única aplicación que sin importar cuál sea el origen de los datos pueda obtener el conocimiento deseado sin ningún tipo de intervención.

Las distintas etapas del conocido proceso de KDD no pueden ser recorridas secuencialmente sino que se requiere de una visión integral del problema a resolver para poder seleccionar las técnicas adecuadas que permitirán extraer a partir de los datos esos *"patrones válidos, novedosos, potencialmente útiles y en última instancia comprensibles ..."* tal como lo expresó Fayyad hace 20 años.

Trabajar en Minería de Datos requiere de la conjugación de varios temas. Dado que la mayoría de estos temas provienen del área de las matemáticas, es sólo a través del convencimiento de su utilidad y aplicación en la resolución de problemas reales y concretos que los alumnos estarán dispuestos a revisarlos. Este es el enfoque que se busca incentivar desde la asignatura ejemplificando con soluciones que han sido aplicadas a casos reales por parte de los docentes.

Si bien durante una buena parte del curso se utilizan herramientas de software con operadores predefinidos para realizar las distintas tareas, la cátedra posee conocimientos suficientes para asistir a aquellos alumnos interesados en estudiar con mayor profundidad estas técnicas ya sea para aplicarlas en su Tesina o bien para resolver situaciones concretas.

También asisten al curso, en calidad de oyentes, estudiantes de postgrado y profesionales de distintas áreas que se interesan por esta temática.

Referencias

[1] Fayyad, U., Piatetsky-Shapiro, G., and Smyth, P. (1996). The KDD process for extracting useful knowledge from volumes of data. Commun. ACM, 39(11):27–34.

[2] Witten, I. H., Frank, E., and Hall, M. A. (2011). Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 3rd edition.

[3] Formia, S. and Lanzarini, L. (2013). Caracterización de la deserción universitaria en la UNRN utilizando minería de datos. Revista Iberoamericana de Tecnología en Educación y Educación en Tecnología (TE&ET), (11):92–98.

[4] Lanzarini, L., Charnelli, M. E., Baldino, G., and Díaz, J. (2015). Selección de atributos representativos del avance académico de los alumnos universitarios usando técnicas de visualización. Un caso de estudio. Revista Iberoamericana de Tecnología en Educación y Educación en

Tecnología (TE&ET), (15):42–50.

[5] Lanzarini, L., Charnelli, M. E., and Díaz, J. (2015). Academic performance of university students and its relation with employment. In Proceedings of the XLI Latin American Computing Conference (CLEI) - XXIII Simposio Iberoamericano de Educación Superior en Computación.

[6] Baldino, G. and Lanzarini, L. (2016). Análisis del avance académico de alumnos universitarios. Un estudio comparativo entre la UN-FRLP y la UNLP. In XI Congreso de Tecnología en Educación y Educación en Tecnología (TE&ET 2016), pages 589–596.

[7] Fundación Sadosky (2013). Y las mujeres… ¿Dónde están? Bases de Datos de las encuestas.

http://www.fundacionsadosky.org.ar/

[8] Charnelli, M. E., Lanzarini, L., Baldino, G., and Díaz, J. (2015). Determining the profiles of young people from Buenos Aires with a tendency to pursue computer science studies. In Feierherd, G., Pesado, P., and Spositto, O., editors, Computer Science & Technology Series - Series - XX Argentine Congress of Computer Science, chapter XII Information Technology Applied to Education Workshop, pages 1155–1163. Red UNCI

[9]UC Irvine Machine Learning Repository. http://archive.ics.uci.edu/ml/

[10] Lanzarini, L., Villa Monte, A., and César, E. (2011). E-mail processing with fuzzy SOMs and association rules. Journal of Computer Science and Technology, 11(1):41–46.

[11] Estrebou, C., Lanzarini, L., and Hasperué, W. (2010). Voice recognition based on probabilistic SOM. In Conferencia Latinoamericana de Informática. CLEI 2010.

[12] Lanzarini, L., Ronchetti, F., Estrebou, C., Lens, L., and Bariviera, A. F. (2013). Face recognition based on fuzzy probabilistic SOM. In IFSA World Congress - NAFIPS Annual Meeting. IEEE Catalog Nro.: CFP13750-USB, pages 310–314.

[13] Lanzarini L., La Battaglia J., Maulini J. and Hasperué W. (2010). Face recognition using SIFT and binary PSO descriptors. In Proceedings of the ITI 2010, 32nd International Conference on Information Technology Interfaces, Cavtat/Dubrovnik, pages 557-562.

[14] Lanzarini L., Villa Monte A., Aquino G., De Giusti A. (2015). Obtaining classification rules using lvqPSO. Advances in Swarm and Computational Intelligence. Lecture Notes in Computer Science. Vol 6433, pp. 183-193, doi. 10.1007/978-3-319-20466-6_20, ISSN 0302-9743. Springer-Verlag Berlin Heidelberg.

[15] Lanzarini, L., Villa Monte, A., Bariviera, A., and Jimbo, P. (2017). Simplifying credit scoring rules using lvq+pso. Kybernetes: The International Journal of Systems & Cybernetics,46(1).

[16] Octave *https://www.gnu.org/software/octave/*

[17] Tanagra *https://eric.univ- lyon2.fr/~ricco/tanagra/*

[18] Weka *http://www.cs.waikato.ac.nz/ml/weka/*

[19] Orange *https://orange.biolab.si/*

[20] RapidMiner *https://rapidminer.com/*

[21] Sylvain, M., Schneider E. and Yaoyu Z. (2012). An Octave extension for RapidMiner. In Proceedings of the third RapidMiner Community Meeting and Conference (RCOMM 2012).