Trabajo de Fin de Grado

“Análisis predictivo: técnicas y modelos utilizados y aplicaciones del mismo - herramientas *Open Source* que permiten su uso”

Carlos Espino Timón
Grado en Ingeniería Informática
Business Inteligence

Xavier Martínez Fontes
Atanasi Daradoumis Haralabus

16 de enero de 2017

FICHA DEL TRABAJO FINAL

Título del trabajo	*Análisis predictivo: técnicas y modelos utilizados y aplicaciones del mismo - herramientas Open Source que permiten su uso.*
Nombre del autor	*Carlos Espino Timón*
Nombre del consultor/a	*Xavier Martínez Fontes*
Nombre del PRA	*Atanasi Daradoumis Haralabus*
Fecha de entrega	01/2017
Titulación	*Grado en Ingeniería Informática*
Área del Trabajo Final	*Business Inteligence*
Idioma del trabajo	***Español***
Palabras clave	*Análisis predictivo, open source*

Resumen del Trabajo

El Análisis Predictivo es una de las herramientas que forman parte de un conjunto de técnicas más amplio conocido como *Business Intelligence*.

La demanda de especialistas en Análisis Predictivo crece en EEUU a un 27% anual, cuando la media en dicho país para el resto de demandas es del 11%. Una cifra que ilustra perfectamente la enorme importancia que administraciones, empresas y organizaciones están otorgando al Análisis Predictivo.

Esta tendencia al uso del Análisis Predictivo es consecuencia de la nueva cultura que se ha generalizado con respecto a los datos. La capacidad real de almacenar y procesar grandes conjuntos de datos, ligada a los avances experimentados por las TI, ha permitido generar archivos masivos de datos de todo tipo, susceptibles de ser analizados en busca de tendencias.

El Análisis Predictivo requiere de herramientas informáticas capaces de detectar patrones en los datos analizados que permitan formular a partir de los mismos reglas susceptibles de ser utilizadas para formular predicciones. El presente trabajo ha pretendido determinar si existen herramientas *open source* capaces de cumplir los requerimientos del análisis predictivo.

El trabajo comienza definiendo qué es el Análisis Predictivo, los modelos y técnicas aplicables y cuales son sus principales aplicaciones. Posteriormente se han identificado las herramientas *open source* disponibles y se ha evaluado el funcionamiento de las dos principales en diferentes ámbitos.

R, con la interface gráfica R-Studio, y Weka han sido las herramientas elegidas para el análisis y en ambos casos se ha concluido que cumplen con las expectativas requeridas.

Abstract (in English, 250 words or less):

Predictive Analysis is one of the tools that are part of a broader set of techniques known as Business Intelligence.

The demand for specialists in Predictive Analysis grows in the US to 27% per year, when the average in that country for other claims is 11%. A figure that perfectly illustrates the enormous importance that administrations, companies and organizations are giving to Predictive Analysis.

This tendency to use Predictive Analysis is a consequence of the new culture that has been generalized regarding the data. The real capacity of storing and processing large data sets, linked to the advances experienced by IT, has allowed the generation of massive data files of all kinds, which can be analyzed for trends.

Predictive Analysis requires computer tools capable of detecting patterns in the analyzed data that allow to formulate from the same rules that can be used to formulate predictions. The current work has tried to determine if there are open source tools capable of fulfilling the requirements of the predictive analysis.

The work begins by defining what is the Predictive Analysis, the applicable models and techniques and what are its main applications. Later the available open source tools have been identified and the operation of the two main ones in different areas has been evaluated.

R, with the graphic interface R-Studio, and Weka have been the tools chosen for the analysis and in both cases it has been concluded that they fulfill the required expectations.

ÍNDICE

1. INTRODUCCIÓN

1.1. Justificación del Trabajo Final de Grado

Todas las compañías tienen a su alcance soluciones de análisis de datos tan avanzadas y sencillas de usar que no saber lo que sucederá dentro de seis meses es pecado mortal – Juan F. Cía

El análisis predictivo supone un cambio en el juego de los negocios – Forrest Research

La citas anteriores están recogidas del artículo “El ranking de las mejores soluciones de análisis predictivo para empresas” publicado en BBVAOpen4U. Se trata de dos afirmaciones rotundas que señalan al análisis predictivo como herramienta imprescindible en la gestión empresarial.

El Plan de Transformación Digital de la Administración General del Estado y sus Organismos Públicos (Estrategia TIC 2015-2020) establece como Objetivo Estratégico IV la Gestión corporativa inteligente del conocimiento, la información y los datos. El citado Plan, al referirse a la ingente cantidad de datos manejados por las administraciones públicas, señala que “Toda esta información abre nuevas perspectivas y permite habilitar servicios innovadores basados en las tecnologías emergentes como el tratamiento de grandes volúmenes de información, la minería de datos, el análisis predictivo, etc.”

A medias entre lo público y lo privado, una empresa como Aqualia, que da servicio a una población total de 27 millones de personas en un área tan fundamental como el suministro de agua, utiliza el análisis predictivo para prevenir cortes de suministro. Aqualia es capaz de identificar patrones y tendencias de consumo con el fin de predecir la demanda en un escenario cambiante tanto por la climatología como por los bruscos cambios en la población debidos al turismo.

Durante la campaña electoral que enfrentó a Barak Obama y a Mitt Romney los sondeos mostraron un escenario en el que ambos candidatos se alternaban en el liderato debido a los *swing voters* (electores susceptibles de cambiar el sentido de su voto). La campaña de Obama decidió centrarse en esos votantes identificando los grupos que reaccionaban mejor ante una llamada telefónica, el envío de información por correo o ante una visita a la casa. Llegaron a registrar grupos votantes que pudieran cambiar el voto de forma negativa si se les contactaba. Lo consiguieron gracias al análisis predictivo. Se aplicaron técnicas de minería de datos que permitieron identificar patrones de comportamiento y crear un modelo.

In God we trust. All others must bring data. — William Edwards Deming

Empresas, administraciones públicas y organizaciones encuentran en el análisis predictivo una herramienta imprescindible que permite sustituir las corazonadas y las apreciaciones personales por proyecciones científicas capaces de eliminar o disminuir las incertidumbres a la hora de elaborar sus estrategias.

El Análisis Predictivo es una subdisciplina del análisis de datos que usa técnicas de estadística, como aprendizaje computacional o minería de datos, para desarrollar modelos que predicen eventos futuros o conductas. Estos modelos predictivos permiten aprovechar los patrones de comportamiento encontrados en los datos actuales e históricos para identificar riesgos y oportunidades.

Empresas de la importancia de IBM, SAS, Microsoft u Oracle, han desarrollado potentes herramientas de Análisis Predictivo, sin embargo, el objeto de este proyecto se centra en el estudio de herramientas *Open Source* que ofrecen las funcionalidades necesarias para llevar a cabo proyectos de Análisis Predictivo.

Las aplicaciones elegidas han sido R y Weka, dos aplicaciones maduras que disponen de interfaces gráficas que permiten abstraer la complejidad matemática del análisis predictivo, poniéndolo a disposición de empresas, instituciones y organizaciones

1.2. Objetivos del Trabajo Final de Grado

A lo largo del TFG se ha hecho un estudio en profundidad del Análisis Predictivo, así como cuáles son las técnicas utilizadas, los modelos aplicables, las aplicaciones que se le puede dar y los interrogantes éticos que plantea. Por otro lado, se ha hecho un análisis de las principales herramientas de Análisis Predictivo disponibles, tratando exclusivamente las que dispongan de licencia *Open Source*.

Los objetivos que se plantean en este TFG son:

- Conocer qué es el Análisis Predictivo, así como los modelos y técnicas aplicables.
- Indicar las principales aplicaciones que tiene el Análisis Predictivo.
- Reconocer las herramientas *Open Source* disponibles en el mercado que se utilizan para hacer el Análisis Predictivo.
- Evaluar el funcionamiento de las principales herramientas en diferentes ámbitos.

1.3. Organización del Trabajo Final de Grado

Para la correcta realización del TFG en el tiempo preestablecido para el semestre se ha organizado el trabajo en una serie de tareas temporalizadas, con el seguimiento por parte del profesor colaborador para garantizar la correcta ejecución del proyecto.

Tareas realizadas

Las tareas realizadas a lo largo del TFG se han centrado en primer lugar en investigar, analizar y sintetizar la información disponible sobre del Análisis Predictivo, los modelos y técnicas utilizados y las aplicaciones que se le puede dar; a continuación se ha abordado la relación de estos conceptos con las herramientas *Open Source* disponibles en el mercado, evaluando las principales para finalmente formular unas conclusiones al respecto. Por último se ha realizado la presente Memoria y la Presentación Final.

Las tareas realizadas han sido:

- Investigación sobre el Análisis Predictivo
- Investigación sobre los Modelos usados en el Análisis Predictivo
- Investigación sobre las Técnicas empleadas en el Análisis Predictivo
- Investigación sobre las Principales Aplicaciones de Análisis Predictivo
- Conocimiento de las principales herramientas *Open Source* disponibles en el mercado
- Análisis de la herramienta R
- Análisis de la herramienta Weka
- Conclusiones
- Elaboración de la Presentación

Temporalización de las tareas

El tiempo preestablecido para cada tarea ha sido el siguiente:

Tarea	Tiempo previsto
Investigación sobre el Análisis Predictivo	7 días
Investigación sobre los Modelos usados en el Análisis Predictivo	7 días
Investigación sobre las Técnicas empleadas en el Análisis Predictivo	7 días
Investigación sobre las Principales Aplicaciones de Análisis Predictivo	7 días
Conocimiento de las principales herramientas *Open Source*	8 días
Análisis de la herramienta R	18 días
Análisis de la herramienta Weka	18 días
Conclusiones	10 días
Elaboración de la Presentación	17 días

Seguimiento y control

El seguimiento de este TFG se ha realizado por el profesor colaborador de la UOC para la asignatura, quien se ha encargado de asegurar que el proyecto coincide con los objetivos propuestos. Este seguimiento se ha llevado a cabo a través de las entregas realizadas a lo largo del semestre, así como mediante la resolución de las dudas planteadas por el autor en momentos puntuales.

El sistema de control se ha basado en la elaboración de tres Pruebas de Evaluación Continua y una Entrega Final a la que han precedido varias entregas previas que han permitido realizar los ajustes finales siguiendo las instrucciones del profesor colaborador.

1.4. Descripción de los capítulos de la memoria

En el primer capítulo se introduce el TFG justificando la importancia del Análisis Predictivo, se detallan los objetivos y se describe la metodología utilizada.

El segundo capítulo define el Análisis Predictivo, los diferentes modelos, las técnicas aplicadas y las principales aplicaciones del mismo.

El tercer capítulo aborda las principales herramientas de Análisis Predictivo, centrándose en R y Weka.

El cuarto capítulo aborda las conclusiones del trabajo.

2. ANÁLISIS PREDICTIVO

2.1. Qué es el Análisis Predictivo

El análisis predictivo consiste en la tecnología que aprende de la experiencia para predecir el futuro comportamiento de individuos para tomar mejores decisiones – Eric Siegel

El análisis predictivo es un área de la minería de datos que consiste en la extracción de información existente en los datos y su utilización para predecir tendencias y patrones de comportamiento, pudiendo aplicarse sobre cualquier evento desconocido, ya sea en el pasado, presente o futuro. El análisis predictivo se fundamenta en la identificación de relaciones entre variables en eventos pasados, para luego explotar dichas relaciones y predecir posibles resultados en futuras situaciones. Ahora bien, hay que tener en cuenta que la precisión de los resultados obtenidos depende mucho de cómo se ha realizado el análisis de los datos, así como de la calidad de las suposiciones.

En un principio puede parecer que el análisis predictivo es lo mismo que hacer un pronóstico (que hace predicciones a un nivel macroscópico), pero se trata de algo completamente distinto. Mientras que un pronóstico puede predecir cuántos helados se van a vender el mes que viene, el análisis predictivo puede indicar qué individuos es más probable que se coman un helado. Esta información, si se utiliza de la forma correcta, supone un cambio radical en el juego, ya que permite orientar los esfuerzos para ser más productivos en la consecución de los objetivos.

Para llevar a cabo el análisis predictivo es indispensable disponer de una considerable cantidad de datos, tanto actuales como pasados, para poder establecer patrones de comportamiento y así inducir conocimiento. Por ejemplo, en el caso comentado en el párrafo anterior, acerca de quién es más probable que se coma un helado, si se cruzan datos acerca de la temperatura registrada, la época del año y si es fin de semana o festivo se puede inferir qué perfil de persona comerá helado. Este proceso se realiza gracias al aprendizaje computacional. Los ordenadores pueden “aprender” de manera autónoma y de esta forma desarrollar nuevo conocimiento y capacidades, para ello basta con proporcionarles el más potente y gran recurso natural de la sociedad moderna: los datos.

2.1.1. Datos

Los datos son la fuente de la que se obtienen las variables, las relaciones entre ellas, el conocimiento inducido o los patrones de comportamiento identificados, convirtiéndose en un elemento vital de todo análisis predictivo.

En la actualidad se crean más datos en un día de los que se crearon en toda la humanidad hasta el año 2.000 – Andreas Weingend

Con la generalización de las Tecnologías de la Información ha aparecido una nueva dimensión en la que contemplar a las personas. Si antes podían ser vistas como ciudadanos, contribuyentes o consumidores (entre otras visiones), las TI permiten contemplar a las personas como proveedores de datos.

Actos como conducir o caminar con un dispositivo capaz de geoposicionar a su usuario, pagar con una tarjeta de crédito o ver una serie online, generan información susceptible de ser explotada. Enviar correos electrónicos, interactuar en las redes sociales o, simplemente, utilizar motores de búsqueda, también genera datos.

La evolución del correo electrónico puede ejemplificar en gran medida ese cambio de paradigma en virtud del cual el usuario pasa a ser considerado fuente de datos. Los primeros sistemas de correo electrónico exigían vaciar periódicamente los buzones, Gmail entra en escena ofreciendo al usuario la posibilidad de no borrar nunca sus mensajes precisamente porque considera que almacenar los correos de sus usuarios le permite disponer de una fuente de datos de extraordinario valor.

El concepto que engloba el almacenamiento de grandes cantidades de datos y las técnicas utilizadas para encontrar patrones repetitivos en los mismos es denominado big data, según la denominación acuñada por Viktor Schönberger en su ensayo Big data: La revolución de los datos masivos.

Según define IBM, el Big Data es la tendencia en el avance de la tecnología que ha abierto las puertas hacia un nuevo enfoque de entendimiento y toma de decisiones, la cual es utilizada para describir enormes cantidades de datos (estructurados, no estructurados y semi estructurados) que tomaría demasiado tiempo y sería muy costoso cargarlos a un base de datos relacional para su análisis.

Esta capacidad sin precedentes para generar datos crecerá de manera exponencial en los próximos años debido a la generalización de la IoT–*Internet of Things* o Internet de las cosas–, que permitirán que la pauta de consumo de congelados, el patrón de uso de las luces exteriores, o los horarios en los que trabaja la lavadora, se incorporen al ya extraordinario caudal de información susceptible de ser explotada, que constituye una inestimable colección de experiencias sobre las cuales aprender.

The only source of knowledge is experience. — Albert Einstein

En resumen, puede afirmarse que todo evento que se registra se puede analizar para encontrar patrones de comportamiento que puedan ser útiles para tomar unas mejores decisiones en el futuro. No obstante, hay que tener en cuenta que inducir conocimiento a partir de los datos no es una tarea fácil, más aún si se tiene en cuenta la gran cantidad de datos con los que se trabaja actualmente.

En es sentido, Weigend parte de la consideración de los datos como el petróleo del siglo XXI para concluir que “al igual que con el petróleo, el valor real de los datos se logra al refinar la información”. Por ello, una vez se dispone de los datos, llega el momento de inducir conocimiento. Para ello se emplean técnicas de aprendizaje computacional.

2.1.2. Aprendizaje computacional

El aprendizaje computacional es parte fundamental en un proceso de análisis predictivo. El aprendizaje computacional proporciona las técnicas de análisis de datos mediante las cuales se pueden descubrir relaciones entre variables que en un principio pueden parecer insignificantes, pero que tras la aplicación de estas técnicas puede descubrirse la trascendencia de las mismas.

Por ejemplo, un estudio realizado sobre los clientes de la compañía Canadian Tire descubrió que los hábitos de compra podían influir en la fiabilidad de pago de un deudor. Si el cliente suele pagar con tarjeta de crédito en bares supone un mayor riesgo de impago, mientras que si la utiliza para pagar el dentista supone un menor riesgo. Una posible explicación a este descubrimiento puede ser que la persona que visita el dentista se considera que, probablemente, sea más conservadora y lleve una vida más planificada.

Una vez se han establecido correlaciones entre variables entra en juego la labor del ser humano, que consiste en saber interpretar las mismas y hacer las suposiciones apropiadas.

2.1.3. Suposiciones

Si bien establecer correlaciones entre variables puede proporcionar información muy valiosa, hay que saber interpretar las mismas del modo correcto para no llegar a conclusiones erróneas. La correlación no implica causalidad. El descubrimiento de una relación entre A y B no implica que una cause la otra.

Por ejemplo, se identifica una correlación entre el un incremento en las ventas de los chiringuitos de playa y el incremento en las muertes por ahogamiento. Esta información podría llevar a pensar que el hecho de comer en el chiringuito implica un aumento de la probabilidad de ahogarse, sin embargo, esto no es

cierto. Existe el mismo riesgo de ahogamiento para el que come en el chiringuito que para el que se trae la comida de casa. Este incremento de muertes por ahogamiento se debe al buen tiempo, puesto que hace que la gente vaya más a la playa, por lo que aumenta el número de clientes del chiringuito así como el número de personas en la playa, y en consecuencia, también aumenta el número de muertes por ahogamiento.

Una vez definidas las suposiciones correctas hay que tratar de aprovechar las mismas, que se utilizarán para realizar predicciones.

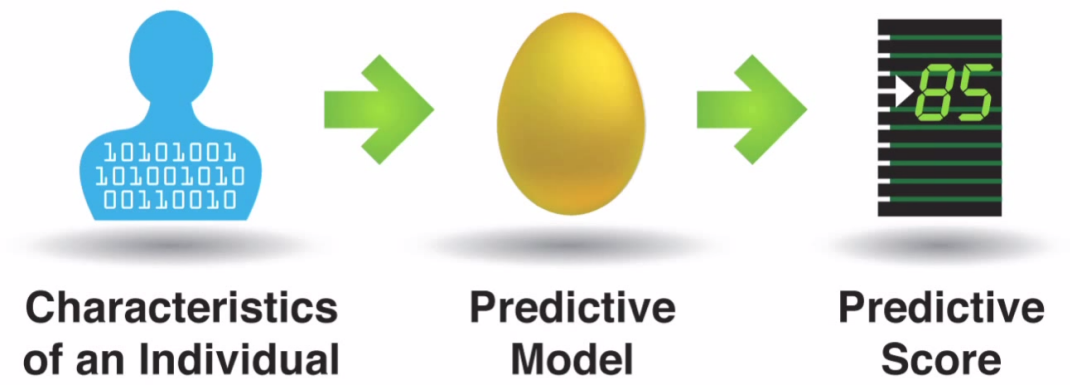
2.1.4. Predicciones

Tras identificar las correlaciones entre variables mediante técnicas de aprendizaje computacional y establecer las suposiciones correctas, se identifican patrones de comportamiento que permiten crear un modelo predictivo[1].

Fuente: Predictive Analytics – The powe to predict who will click, buy lie or die.

Este modelo predictivo se podrá utilizar para predecir qué probabilidades hay de que una persona –en función de los datos que se disponga de la misma– reaccione de una manera determinada (si comprará un producto, si cambiará de voto, si contratará un servicio…). Una vez introducidos los datos de la persona y se aplique el modelo predictivo se obtendrá una calificación que indicará la probabilidad de que se produzca la situación estudiada por el modelo.

1 Los modelos predictivos se desarrollarán en profundidad en el apartado 2.2. Modelos de análisis predictivo.



Fuente: Predictive Analytics – The powe to predict who will click, buy lie or die.

Un ejemplo que puede ayudar a aclarar este concepto sería el de una agencia de seguros de coche. Se puede crear un modelo predictivo que indique el riesgo de que una persona necesite hacer uso del seguro contratado y, en función de los resultados obtenidos, la agencia puede establecer un precio. Obviamente, esta herramienta es de gran ayuda para las aseguradoras, ya que les permite realizar ofertas que se ajustan más a las condiciones del cliente y de este modo no correr riesgos. Porque el hecho de que una persona obtenga una calificación de riesgo por encima de la media por parte del modelo predictivo, no implica que la agencia no deba hacerle un seguro, sino que su seguro debe de ser tener un precio más elevado.

There's no such thing as bad risk, only bad pricing. —Stephen Brobst

Hay que reparar en que un modelo predictivo nunca proporcionará el 100% de acierto, es más, habrá muchas veces en las que se aleje bastante de esos resultados. Esto se debe a que por mucho que se haya repetido un patrón de comportamiento en el pasado, no tiene porque repetirse. Sin embargo, siempre será mejor predecir con ayuda del modelo (aunque el modelo proporcione porcentajes de acierto bajos) que simplemente adivinar.

2.1.5. Influencia sobre las personas

El análisis predictivo proporciona a las empresas las herramientas necesarias para poder orientar sus campañas hacia personas que les ofrezcan una mayor probabilidad de éxito, es decir, hacia personas más influenciables. Se pueden crear modelos que indiquen qué clientes van a comprar si son contactados por parte de la empresa, sin embargo, dentro de esos clientes habrá algunos que hubieran comprado aún sin el contacto por parte de la empresa. Por lo tanto, es necesario crear un modelo que indique qué clientes van a comprar sólo si son contactados, lo que permitirá rentabilizar aún más los esfuerzos. Este tipo de

modelos se denomina el *uplift model* (modelo de elevación) o modelo persuasivo, que son modelos que sirven para predecir la influencia.

Ahora bien, el análisis predictivo no sólo indica a qué personas contactar, sino que además, indica a que personas no se debe contactar porque pueda ser contraproducente. Esto sucede con personas que ya sean clientes de la empresa y que, ante un contacto por parte de la misma, pueda generar pérdidas para la empresa. Por ejemplo, hay clientes que no están del todo satisfechos con su compañía de teléfono y además tienen personas cercanas que acaban de cambiar de compañía y le proporcionan buenas referencias de la competencia, sin embargo tienen contrato de permanencia, por lo que no puede cambiar de compañía. Si este tipo de cliente recibe una llamada cuando está concluyendo su contrato para renovarlo, se puede generar el efecto inverso, recordar al cliente que ahora puede marcharse. A este tipo de clientes se les denomina *sleeping dogs* (clientes dormidos).

2.1.6. Aspectos éticos del análisis predictivo

La discusión ética sobre el uso y los límites de análisis predictivo se incardina en el debate más general del papel que juegan las TI en el nuevo modelo de sociedad que surge como consecuencia de las mismas.

Límites de la transparencia, derecho al olvido o privacidad son conceptos que dominan el debate teórico (y su aplicación práctica) sobre la necesidad de regular esas nuevas tecnologías que han dejado de ser algo posible en el futuro para convertirse en parte de la realidad cotidiana.

El análisis predictivo es capaz de generar conocimiento sobre las personas, pues no trabaja con los datos de un individuo en particular, pues se limita a identificar patrones de comportamiento que se producen en un conjunto de individuos.

Sin embargo, los patrones encontrados a través del análisis predictivo son susceptibles de aplicarse a individuos concretos. Una empresa puede optar por no contratar mujeres con un determinado perfil socioeconómico en una concreta franja de edad por entender que tenderán a quedarse embarazadas en los próximos años. Una compañía de seguros puede negar la cobertura a determinados individuos al deducir que es un candidato potencial a sufrir un infarto por sus patrones de gasto por encima de la media en alcohol y tabaco.

En los dos supuestos anteriores no se ha producido una invasión de la intimidad de las personas afectadas, en el sentido estricto del término, y no se ha recurrido a prácticas ilegales accediendo, por ejemplo, a su historial médico. Sin embargo, el conocimiento inducido a través del análisis predictivo ha provocado una situación discriminatoria.

No es fácil, ni es el objeto de este trabajo, dar respuesta a los dilemas éticos y legales que plantean el análisis predictivo, en particular, y el uso de las TI, en general. Corresponde afirmar, sin embargo, que la resolución del dilema ha de

orientarse en un doble sentido: garantizar la neutralidad de las técnicas y perseguir el uso inadecuado de las mismas.

Privacy is a compromise between the interests of the government and the citizen. —Eric Schmidt

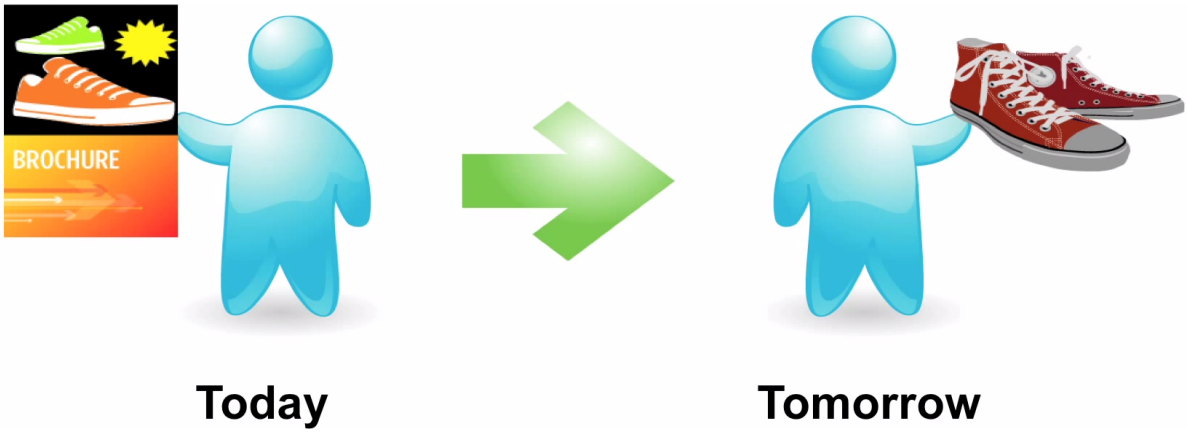
Para finalizar, cabe afirmar que la introducción de las IT han supuesto un auténtico cambio cualitativo en la sociedad contemporánea. Al igual que la generalización del transporte por medios mecánicos supuso una revolución en la movilidad que provocó, a su vez, un cambio radical en la propia concepción del espacio urbano y los hábitos de vida, las IT y, concretamente, la posibilidad real de manejar enormes conjuntos de datos capaces de generar nueva información supone la aparición de nuevas oportunidades que conllevan nuevos riesgos a los que dar respuesta.

Information technology has changed just about everything in our lives. . . . But while we have new ethical problems, we don't have new ethics. —Michael Lotti

2.2. Modelos aplicables en el análisis predictivo

Generalmente, se usa el término análisis predictivo cuando en realidad se está hablando del modelado predictivo, que realiza calificaciones mediante modelos predictivos y pronósticos. Sin embargo, cada vez se está utilizando más el término para referirse a todo lo relacionado con la disciplina analítica, como el modelado descriptivo o el modelado decisivo. Estas disciplinas implican un riguroso análisis de datos y son ampliamente utilizadas en negocios mecanismo de ayuda a la toma de decisiones.

Un modelo predictivo es un mecanismo que predice el comportamiento de un individuo. Utiliza las características del individuo como entrada y proporciona una calificación predictiva como salida. Cuanto más elevada es la calificación, más alta es la probabilidad de que el individuo exhiba el comportamiento predicho.



Fuente: Predictive Analytics – The powe to predict who will click, buy lie or die.

La calificación producida por cualquier modelo predictivo debe de ser tenida en cuenta con especial cuidado y puede requerir que se cruce con otro modelo o que se produzca un análisis adicional a la hora de aplicarla a un individuo concreto. Las calificaciones hablan de tendencias y posibilidades en un grupo lo suficientemente grande, pero no garantiza que la predicción se cumpla en cada caso individual, pues una probabilidad individual por naturaleza simplifica excesivamente la cosa del mundo real que describe.

Puede entenderse mejor con un ejemplo: en una entidad financiera el modelo aplicado al análisis de riesgo predice para un determinado perfil de cliente que el incumplimiento en un recibo mensual de la tarjeta de crédito cuadruplica la posibilidad de impago de al menos otro recibo durante el año. Sin embargo, un examen que incluya nuevas variables puede concluir que el impago fue debido a un gasto extra no previsible, como una avería de importancia en el coche durante el mes que se produjo el impago y que será compensado con la paga extra que llega al mes siguiente, por lo que la previsión no es aplicable.

El tipo de análisis que permiten los modelos predictivos valora la relación existente entre cientos de elementos para aislar los datos que informan sobre un hecho, guiando a la toma de decisiones por un camino seguro. Un paso más allá se encuentra los modelos de decisión, que tienen un modo de trabajar muy similar a la de los modelos predictivos, aunque se emplean en escenarios de mayor complejidad. Se trata de la forma más avanzada de análisis predictivo y consiste en predecir lo que sucedería si se toma una acción determinada. También se conocen como modelos prescriptivos y se basan en la cartografía de las relaciones existentes entre todos los elementos de una decisión.

2.2.1. Modelos predictivos

Los modelos predictivos son modelos de la relación entre el rendimiento específico de una unidad en una muestra y uno o más atributos o características

conocidos de la unidad. El objeto del modelo es evaluar la probabilidad de que una unidad similar en una muestra diferente exhiba un comportamiento específico. Esta categoría abarca modelos que se encuentran en muchas áreas, como el marketing, donde buscan patrones de datos ocultos para responder preguntas sobre el desempeño del cliente, como los modelos de detección de fraude. Los modelos predictivos a menudo ejecutan cálculos durante las transacciones en curso, por ejemplo, para evaluar el riesgo o la oportunidad de un cliente o transacción en particular, de forma que aporte conocimiento a la hora de tomar una decisión. Con los avances en la velocidad de computación, los sistemas de modelado de agentes individuales han sido capaces de simular el comportamiento humano o reacciones ante estímulos o escenarios específicos.

El análisis predictivo construye un modelo estadístico que utiliza los datos existentes para predecir datos de los cuales no se dispone. Como ejemplo del análisis predictivo se incluyen las líneas de tendencia o la puntuación de la influencia[2]. Para la creación del modelo predictivo se utilizan unidades de muestra disponibles con atributos conocidos y un comportamiento conocido, a este conjunto de datos se le denomina conjunto de entrenamiento. Por otro lado, se utilizará una series de unidades de otra muestra con atributos similares, pero de las cuales no se conoce su comportamiento, a este conjunto de datos se le denomina conjunto de prueba.

2.2.2. Modelos descriptivos

Los modelos descriptivos cuantifican las relaciones entre los datos de manera que es utilizada a menudo para clasificar clientes o contactos en grupos. A diferencia de los modelos predictivos que se centran en predecir el comportamiento de un cliente en particular, los modelos descriptivos identifican diferentes relaciones entre los clientes y los productos. La analítica descriptiva proporciona resúmenes simples sobre la audiencia de la muestra y sobre las observaciones que se han hecho. Estos resúmenes pueden constituir la base de la descripción inicial de los datos como parte de un análisis estadístico más amplio, o pueden ser suficientes en si mismos para una investigación en particular.

Los modelos descriptivos no clasifican u ordenan a los clientes por su probabilidad de realizar una acción particular de la misma forma en la que lo hacen los modelos predictivos. Sin embargo, los modelos descriptivos pueden ser utilizados por ejemplo para asignar categorías a los clientes según su preferencia en productos o su franja de edad. Las aplicaciones de los modelos descriptivos pueden ser utilizados para desarrollar nuevos modelos adicionales que pueden imitar un gran volumen de agentes individuales y hacer predicciones. Entre los modelos descriptivos se pueden citar los modelos de simulación, la teoría de colas o las técnicas de previsión.

2 Se pueden ver más aplicaciones del análisis predictivo en el apartado 2.4. Principales aplicaciones del Análisis Predictivo

El análisis descriptivo calcula estadísticas descriptivas para resumir los datos. La mayoría de los análisis sociales (*social analytics*) pertenecen a esta categoría.

2.2.3. Modelos de decisión

Los modelos de decisión describen la relación entre todos los elementos de una decisión –los datos conocidos (incluyendo los resultados de los modelos predictivos), la decisión y el pronóstico de los resultados de una decisión– con la intención de predecir los resultados de una decisión en la que se involucran gran cantidad de variables. Estos modelos pueden ser utilizados en la optimización o maximización de determinados resultados mientras minimizan otros. Los modelos de decisión se utilizan en general para el desarrollo de la decisión lógica o conjunto de reglas de negocio que deberían producir el resultado deseado para cada cliente o circunstancia.

Los modelos de decisión se usan para modelar una decisión que se toma una vez, así como para modelar un enfoque de toma de decisiones repetible que se utilizará una y otra vez. Como ejemplos de este tipo de modelo cabe destacar los árboles de decisión, el análisis Pareto, el análisis SWOT o el análisis de la matriz de decisiones,

2.2.4. Modelos *ensemble*

El modelado *ensemble* –o modelado de conjuntos– consiste en la aplicación del modelado predictivo para combinar dos o más modelos y luego sintetizar los resultados en una sola puntuación o propagación para mejorar la precisión. Al aplicar un solo modelo basado en una muestra de datos puede tener sesgos, una alta variabilidad o inexactitudes absolutas que afectan la confianza de sus hallazgos analíticos. El uso de técnicas de modelado específicas puede presentar inconvenientes similares. Al combinar diferentes modelos o analizar múltiples muestras, se pueden reducir los efectos de esas limitaciones.

Este sistema de modelado *ensemble* considera las predicciones de ambos modelos caso por caso. En ciertos casos, puede dar más credibilidad a un modelo sobre otro, o al revés. Al hacerlo, el modelo *ensemble* se capacita para predecir qué casos son puntos débiles para cada modelo que lo compone. Puede haber muchos casos en los que los dos modelos están de acuerdo, pero cuando hay desacuerdo, el trabajo conjunto de los modelos ofrece la oportunidad de mejorar el rendimiento.

Un ejemplo de modelado de conjuntos es el modelo de *random forest* (bosque aleatorio). Un modelo de bosque aleatorio combina árboles de decisión que

pueden analizar diferentes datos de muestra, evaluar diferentes factores o variables comunes de peso de manera diferente. Los resultados de los diversos árboles de decisión se convierten entonces en un promedio simple o agregados a través de una ponderación adicional.

> *When joined in an ensemble, predictive models compensate for one another's limitations, so the ensemble as a whole is more likely to predict correctly than its component models are – Eric Siegel*

2.2.5. Modelo *uplift*

El modelo *uplift* (o modelo de elevación), también conocido como modelo incremental o de red, es una técnica de modelado predictivo que modela el impacto incremental producido por el tratamiento (como una acción de *marketing*) sobre el comportamiento de un sujeto. Este modelo predictivo predice la influencia de un tratamiento en el comportamiento de un individuo.

El modelo *uplift* utiliza un control científico aleatorio para medir no sólo la eficacia de una acción de marketing, sino también para construir un modelo predictivo que predice la respuesta incremental a la acción de marketing. Se trata de una técnica de minería de datos que se ha aplicado principalmente en las industrias de servicios financieros, telecomunicaciones y comercio minorista para la retención de clientes y ventas cruzadas. Este modelo trata de identificar el *uplift* de una acción, por ejemplo, de una campaña de marketing. El *uplift* se define como la diferencia de la tasa de respuesta entre un grupo, tratado mediante una campaña de *marketing*, y otro grupo aleatorio de control. Ambos grupos tendrán las mismas características salvo que el grupo de control no recibirá el tratamiento.

Un modelo predictivo estándar respondería a la pregunta:

¿El cliente comprará si le contactamos?

Sin embargo, el modelo *uplift* permite hacer otro tipo de preguntas:

¿El cliente comprará sólo si le contactamos?

En un principio puede parecer que se está preguntando lo mismo, pero si se analiza detenidamente se puede ver que implica un paso más allá. La respuesta a esta pregunta permite centrar los esfuerzos en aquellos clientes que sólo comprarán si se les contacta, e ignorar a aquellos clientes que comprarán el producto tanto si se les contacta como si no. Este modelo indicará que clientes son influenciables por los esfuerzos realizados con una campaña publicitaria.

Mediante este modelo se puede hacer una combinación de dos preguntas, lo que ayudará a colocar a cada individuo dentro de uno de los cuatro segmentos

conceptuales que se distinguen a los largo de dos dimensiones. Se puede explicar mediante el ejemplo de la campaña publicitaria:

Compra si recibe una oferta		Compra si no recibe una oferta	
		Si	No
	No	No molestar	Causa perdida
	Si	Comprará igualmente	Influenciable

Este modelo permitirá dividir a los clientes en varios grupos, e identificar al grupo de clientes sobre los que centrar los esfuerzos en aquellos, ya que que son influenciables por las acciones llevadas a cabo por la campaña. Por otro lado, este modelo permitirá identificar a aquellos clientes a los cuales no se les debe molestar, los denominados *sleeping dogs,* dado que un acercamiento puede producir efectos adversos para los intereses de la campaña de marketing.

2.2.6. Validación de los modelos

Una vez se ha creado un modelo es necesario comprobar que el mismo funciona de manera correcta, este es el aspecto más importante de los modelos predictivos, su validación. Puesto que es relativamente fácil crear un modelo se hace muy importante la validación, ya que es la única forma de saber si el modelo funciona.

Una manera muy extendida para comprobar el modelo consiste en dividir el conjunto de datos del que se dispone en dos. Por un lado, se dispone de un conjunto de datos sobre el cual se desarrollará el modelo, este conjunto abarcará dos terceras partes de la muestra y se denomina *training set* (conjunto de entrenamiento). Por otro lado, la tercera parte sobrante se utilizará para validar el modelo y se denomina *test set* (conjunto de test).

2.3. Técnicas aplicables al análisis predictivo

Los enfoques y técnicas utilizados para realizar el análisis predictivo pueden agruparse de una manera muy general en técnicas de regresión y técnicas de aprendizaje computacional.

2.3.1. Técnicas de regresión

Los modelos de regresión son el pilar de la analítica predictiva. El enfoque se basa en el establecimiento de una ecuación matemática como modelo para representar las interacciones entre las diferentes variables en consideración. Dependiendo de la situación, hay una gran variedad de modelos que se pueden aplicar durante la realización del análisis predictivo.

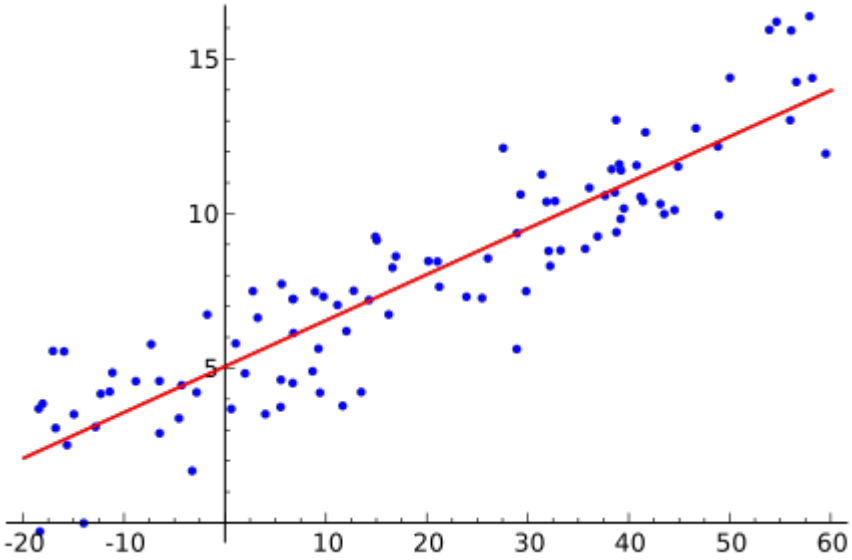
2.3.1.1. Modelo de regresión lineal

El modelo de regresión lineal analiza la relación existente entre la variable dependiente o de respuesta y un conjunto de variables independientes o predictoras. Esta relación se expresa como una ecuación que predice la variable de respuesta como una función lineal de los parámetros. Estos parámetros se ajustan para que la medida de ajuste sea óptima. Gran parte del esfuerzo en la adaptación del modelo se centra en minimizar el error, así como en asegurarse que está distribuido de forma aleatoria respecto a las predicciones del modelo.

El objetivo de la regresión es seleccionar los parámetros del modelo que minimizan la suma de los errores al cuadrado. Esto se conoce como estimación de mínimos cuadrados ordinarios y los resultados en las mejores estimaciones lineales no sesgadas de los parámetros si y sólo si se satisfacen las suposiciones de Gauss-Markov.

Una vez que se ha estimado el modelo, es necesario saber si las variables predictoras pertenecen al mismo. Para ello podemos comprobar la significancia estadística de los coeficientes del modelo que pueden medirse utilizando el estadístico “t”. Esto equivale a probar si el coeficiente es significativamente diferente de cero.

En la siguiente imagen se puede ver un ejemplo de regresión linear simple. Este caso dispone de una variable dependiente y otra independiente, que proporcionan una función lineal que predice los valores de la variable dependiente según la función de la variable independiente. Se puede ver claramente como ciertos valores de la variable dependiente quedan por encima de la función lineal generada y otros por debajo, permitiendo al modelo predecir los valores de la variable dependiente.



Fuente: https://en.wikipedia.org/wiki/Linear_regression

2.3.1.2. Análisis de supervivencia o duración

El análisis de supervivencia es otro nombre para el análisis del tiempo hasta el evento. Estas técnicas se desarrollaron principalmente en las ciencias médicas y biológicas, pero también se usan ampliamente en las ciencias sociales como la economía, así como en la ingeniería (fiabilidad y análisis del tiempo de falla).

La censura y la no-normalidad, que son características de los datos de supervivencia, generan dificultad al intentar analizar los datos usando modelos estadísticos convencionales como la regresión lineal múltiple. La distribución normal, que es una distribución simétrica, toma tanto valores positivos como negativos, pero la duración por su propia naturaleza no puede ser negativa y, por lo tanto, no se puede asumir la normalidad cuando se trata de datos de duración/supervivencia. Por lo tanto, la suposición de normalidad de los modelos de regresión es violada.

El supuesto es que si los datos no fueron censurados sería representativo de la población de interés. En el análisis de supervivencia, las observaciones censuradas surgen cuando la variable dependiente de interés representa el tiempo hasta un evento terminal, y la duración del estudio es limitada en el tiempo.

Un concepto importante en el análisis de supervivencia es la tasa de riesgo, definida como la probabilidad de que el evento ocurra en el tiempo *t* condicional a sobrevivir hasta el tiempo *t*. Otro concepto relacionado con la tasa de riesgo es la función de supervivencia que puede definirse como la probabilidad de sobrevivir al tiempo *t*.

La mayoría de los modelos intentan modelar la tasa de riesgo eligiendo la distribución subyacente dependiendo de la forma de la función de riesgo. Una distribución cuya función de riesgo se inclina hacia arriba se dice que tiene una dependencia de duración positiva, un riesgo decreciente muestra una depen-

dencia de duración negativa mientras que un riesgo constante es un proceso sin memoria usualmente caracterizada por la distribución exponencial.

2.3.1.3. Árboles de clasificación y regresión

El Análisis Discriminante Óptico Jerárquico (*Hierarchial Optimal Discriminat Analysis*, HODA) es una generalización del Análisis Discriminante Óptimo que puede ser utilizado para identificar el modelo estadístico que tiene la máxima precisión para predecir el valor de una variable categórica dependiente para un conjunto de datos que consiste en variables categóricas y variables continuas. La salida de HODA es un árbol que combina variables categóricas y puntos de corte para variables continuas que proporciona máxima precisión predictiva y una evaluación de potencial generalización cruzada del modelo estadístico. El análisis discriminante óptimo es una alternativa al ANOVA (ANalisis Of VAriance o análisis de varianza) y al análisis de regresión, que intentan expresar una variable dependiente como una combinación lineal de otras características o medidas. Sin embargo, ANOVA y el análisis de regresión dan una variable dependiente que es una variable numérica, mientras que el análisis discriminante óptimo jerárquico da una variable dependiente que es una variable de clase.

Los árboles de clasificación y regresión (Classification And Regression Trees, CART) son una técnica de aprendizaje de árboles de decisión no paramétrica que produce árboles de clasificación o regresión, dependiendo de si la variable dependiente es categórica o numérica, respectivamente.

Los árboles de decisión están formados por una colección de reglas basadas en variables en el conjunto de datos de modelado:

- Las reglas basadas en valores de variables se seleccionan para obtener la mejor división para diferenciar observaciones basadas en la variable dependiente.
- Una vez que se selecciona una regla y divide un nodo en dos, se aplica el mismo proceso a cada nodo "secundario", es decir, es un procedimiento recursivo.
- La división se detiene cuando CART detecta que no se pueden realizar más ganancias o se cumplen algunas reglas de parada preestablecidas.

Cada rama del árbol finaliza en un nodo terminal. Cada observación cae en un nodo terminal, y cada nodo terminal es definido de manera única por un conjunto de reglas.

2.3.1.4. Curvas de regresión adaptativa multivariable

Las curvas de regresión adaptativa multivariable (Multivariate Adaptative Regression Splines, MARS) son una técnica no paramétrica que construye modelos flexibles al ajustar regresiones lineales por piezas. Un concepto importante asociado con curvas de regresión es el de un nudo. Un nudo es donde un

modelo de regresión local da paso a otro y por lo tanto es el punto de intersección entre dos curvas.

En las curvas de regresión adaptativa multivariante, las funciones de base son la herramienta utilizada para generalizar la búsqueda de nudos. Las funciones básicas son un conjunto de funciones utilizadas para representar la información contenida en una o más variables. El modelo MARS casi siempre crea las funciones de base en parejas.

La curva de regresión adaptativa multivariable es un modelo que primero realiza un sobreajuste y luego hace una poda para obtener un modelo óptimo. El algoritmo es computacionalmente muy intensivo y en la práctica se requiere especificar un límite superior en el número de funciones de base.

Cabe destacar que estos no son los únicos modelos, existen otros modelos de regresión como los modelos de elección discreta, de regresión logística, de regresión logística multinomial, modelos probit, o los modelos de series temporales. Sin embargo, no es objeto de estudio del presente proyecto el estudio de todos ellos.

2.3.2. Técnicas de aprendizaje computacional

El aprendizaje computacional se empleó originalmente para desarrollar técnicas que permitieran a las computadoras aprender. Hoy en día, ya que incluye una serie de métodos estadísticos avanzados para la regresión y la clasificación, tiene aplicación en una amplia variedad de campos, incluyendo diagnósticos médicos, detección de fraudes de tarjetas de crédito, reconocimiento de la cara y el habla y el análisis del mercado de valores. En ciertas aplicaciones es suficiente predecir directamente la variable dependiente sin tener en cuenta las relaciones subyacentes entre variables. En otros casos, las relaciones subyacentes pueden ser muy complejas y la forma matemática de las dependencias desconocida. Para estos casos, las técnicas de aprendizaje automático emulan la cognición humana y aprenden de los ejemplos de entrenamiento para predecir eventos futuros.

A continuación se proporciona una breve descripción de algunos de estos métodos utilizados comúnmente para el análisis predictivo.

2.3.2.1. Redes neuronales

Las redes neuronales son técnicas de modelado no lineal sofisticadas que son capaces de modelar funciones complejas. Pueden aplicarse a problemas de predicción, clasificación o control en un amplio espectro de campos como las finanzas, la psicología cognitiva/neurociencia, la medicina, la ingeniería y la física.

Las redes neuronales se utilizan cuando no se conoce la naturaleza exacta de la relación entre los valores de entrada y de salida. Una característica clave de las redes neuronales es que aprenden la relación entre los valores de entrada y salida a través del entrenamiento. Existen tres tipos de entrenamiento en redes neuronales utilizadas por diferentes redes, el aprendizaje por refuerzo, el supervisado y no supervisado, siendo el supervisado el más común.

2.3.2.2. Máquinas de vectores de soporte

Las máquinas de vectores de soporte (SVM) se usan para detectar y explotar patrones complejos de datos agrupando, ordenando y clasificando los datos. Son máquinas de aprendizaje que se utilizan para realizar clasificaciones binarias y estimaciones de regresión. Usualmente usan métodos basados en kernel para aplicar técnicas de clasificación lineal a problemas de clasificación no lineal. Hay una serie de tipos de SVM tales como lineal, polinomial, sigmoide, etc.

2.3.2.3. Naïve Bayes

El clasificador bayesiano ingenuo se basa en la regla de probabilidad condicional de Bayes, que se utiliza para la tarea de clasificación. El clasificador bayesiano asume que los predictores son estadísticamente independientes, lo que hace que sea una herramienta de clasificación eficaz que sea fácil de interpretar. Se emplea mejor cuando se enfrenta al problema de la “maldición de la dimensionalidad”, es decir, cuando el número de predicciones es muy alto.

2.3.2.4. K-vecinos más cercanos

El algoritmo vecino más próximo k-NN (*Nearest Neighbor*) pertenece a la clase de métodos estadísticos de reconocimiento de patrones. El método no impone a priori ninguna suposición sobre la distribución de la que se extrae la muestra de modelado. Se trata de un conjunto de entrenamiento con valores positivos y negativos. Una nueva muestra se clasifica calculando la distancia al vecino más cercano del conjunto de entrenamiento. El signo de ese punto determinará la clasificación de la muestra. En el clasificador k-vecino más cercano, se consideran los *k* puntos más cercanos y se utiliza el signo de la mayoría para clasificar la muestra. El rendimiento del algoritmo k-NN está influenciado por tres factores principales:

- la medida de distancia utilizada para localizar a los vecinos más cercanos
- la regla de decisión usada para derivar una clasificación de los k-vecinos más cercanos
- el número de vecinos utilizados para clasificar la nueva muestra.

Se puede demostrar que, a diferencia de otros métodos, este método es universal y asintóticamente convergente, es decir, a medida que el tamaño del conjunto de entrenamiento aumenta, si las observaciones son independientes e idénticamente distribuidas, independientemente de la distribución a partir de la cual se dibuja la muestra, la clase predicha convergerá a la asignación de clase que minimiza el error de clasificación errónea.

Al igual que en apartado anterior, estos nos son los únicas técnicas de aprendizaje computacional, existen otras como la función de base radial, el perceptrón multicapa o modelado predictivo geoespacial. Sin embargo, una vez más, estas técnicas no son objeto de estudio del presente trabajo.

2.3.3. Técnicas aplicables en entornos *Open Source*

Tal y como se ha podido comprobar, hoy en día existe una gran cantidad de técnicas predictivas que sirven para la construcción de modelos. Hay diferentes técnicas que son soportadas por diferentes sistemas y proveedores, pero algunas son exclusivas y no permiten su libre uso, como por ejemplo los algoritmos utilizados por Microsoft, que dispone de algoritmos como: *Microsoft Decision Trees Algorithm, Microsoft Naive Bayes Algorithm* o *Microsoft Clustering Algorithm* entre otros. Sin embargo hay otras técnicas genéricas que son soportadas por la mayoría de los sistemas y entornos.

Los SVM, así como los vecinos más cercanos (NN) y los modelos de regresión logística, son poderosas técnicas genéricas que, aunque matemáticamente diferentes, generan resultados algo comparables. Los árboles de decisión representan otra técnica genérica de modelado predictivo que destaca por su capacidad de explicar la razón detrás de la producción producida. Debido a que son fáciles de usar y entender, los árboles de decisión son la técnica de modelado predictivo más utilizada.

Por otro lado, las técnicas de agrupamiento (*Clustering*) son muy populares cuando el objetivo o la variable de respuesta no es importante, o no está disponible. Como su propio nombre indica, las técnicas de agrupamiento pueden agrupar datos de entrada dependiendo de la similitud.

Cuando una variable objetivo o medida de similitud no es importante, pero las asociaciones entre los elementos de entrada lo son, para encontrarlos se puede utilizar una técnica conocida como reglas de asociación. Por ejemplo, las reglas de asociación se pueden utilizar para descubrir que la gente que compra los pañales y la leche, también compra cerveza.

Aunque todas las técnicas predictivas tienen diferentes fortalezas y debilidades, la exactitud del modelo depende en gran medida de los datos de entrada a procesar y de las características utilizadas para formar un modelo predictivo. La construcción de modelos implica una gran cantidad de análisis de datos. Normalmente, a partir de cientos de campos de datos sin procesar disponibles, se

selecciona un subconjunto y los campos se procesan previamente antes de presentarlos a una técnica de modelado predictivo.

Las técnicas de agrupamiento requieren que el número de grupos se proporcione antes del entrenamiento. En este caso, si el número de grupos es demasiado pequeño, el modelo puede perder importantes diferencias en los datos de entrada, ya que se está obligando a cubrir diferentes datos juntos. Por otra parte, si el número de grupos es demasiado grande, puede faltar similitudes importantes.

2.4. Principales aplicaciones del Análisis Predictivo

En el artículo[3] de la revista Inbound Logistics Latam, Thomas Shimada y el Doctor Fabián López hablan acerca de cuáles son las áreas de aplicación de la analítica predictiva, diferenciando cinco áreas a través de las cuales se hace más rentable la cartera de clientes:

1. Segmentación de clientes. La segmentación permite adecuar las ofertas en función del nivel de ingresos, franja de edad, sexo o estudios realizados, entre otras variables.
2. Personalización de la oferta. Conocer cuál es la siguiente mejor oferta que se le puede hacer a un cliente a partir de su comportamiento histórico. “El cliente ha comprado un nuevo móvil, puede estar interesado en comprar una funda”.
3. Detectar el riesgo de que el cliente abandone la relación comercial en función del ritmo de pedidos o contactos que realiza o de las incidencias que registra.
4. Conocer cuáles son los clientes más propensos a responder a las iniciativas de comunicación publicitaria, para sacar el mayor provecho a la inversión hecha mercadológicamente.
5. Conocer la tasa de deserción; es decir, predecir en forma anticipada y proactiva cuáles son los clientes que están buscando otras ofertas para evitar que estos desvíen su atención hacia la competencia. A través de esta aplicación separo los clientes rentables de los que no lo son.

Este artículo se centra en el sector empresarial y en el beneficio que el análisis predictivo le puede aportar al mismo, sin embargo existen otros sectores que se pueden aprovechar de las ventajas del análisis predictivo.

A continuación se muestran algunos de los sectores que se benefician del análisis predictivo, así como aplicaciones que se utilizan en cada uno de ellos:

2.4.1. Sector de *Marketing*

Algunas de las aplicaciones que tiene el análisis predictivo en este sector son las siguientes:

Marketing directo

3 Se puede encontrar el artículo en http://www.il-latam.com/images/articulos/articulo-revista-109-como-convertir-la-informacion-en-ventaja-competitiva.pdf

Consiste en un modelo capaz de predecir qué clientes responderán ante un contacto de *marketing*. Esto permite a las empresas comunicarse con aquellos clientes que tienen una mayor probabilidad de responder.

Publicidad predictiva

Consiste en un modelo capaz de predecir qué anuncio es más probable que cada cliente haga un clic. Esto permite a las empresas elegir el mejor anuncio basándose en la probabilidad de que el cliente haga clic y en lo que recibe por cada clic.

Predictor de embarazos

Consiste en un modelo capaz de predecir qué clientas van a tener un bebé en los próximos meses. Esta información permite a las empresas realizar ofertas relevantes para los futuros padres, como por ejemplo, oferta de cunas o carritos de bebe.

Persuasión del voto en campañas electorales

Consiste en un modelo capaz de predecir qué votantes se pueden persuadir positivamente durante la campaña mediante contacto (llamada, anuncio de televisión, visita en la casa…). De este modo se pueden centrar los esfuerzos durante la campaña para acceder a aquellos votantes que pueden cambiar de voto.

2.4.2. Sector Actuarial

Una aplicación que tiene el análisis predictivo en este sector es la siguiente:

Detector de fraude

Consiste en un modelo capaz de predecir qué transacciones o solicitudes de crédito o reembolso tienen mayor probabilidad de ser fraudulentas, para que posteriormente sean analizadas con detenimiento.

2.4.3. Sector de Servicios financieros

Algunas de las aplicaciones que tiene el análisis predictivo en este sector son las siguientes:

Compraventa de acciones

Consiste en un modelo capaz de predecir si una acción subirá a bajara. De este modo, al usuario de la aplicación obtendrá información acerca de la proba-

bilidad de que determinadas acciones suban, y así comprarlas, o bajen, y venderlas si es dueño de las mismas.

Estimación del valor hipotecario

Consiste en un modelo capaz de predecir qué clientes van a hacer el pre-pago de una hipoteca en un futuro cercano, de este modo pueden decidir si vender la hipoteca a otro banco o no.

2.4.4. Sector de Administraciones públicas

Una aplicación que tiene el análisis predictivo en este sector es la siguiente:

Reducción de reincidencia

Consiste en un modelo capaz de predecir la probabilidad de que un criminal al que se está enjuiciando pueda delinquir de nuevo. Los jueces y los tribunales pueden consultar las predicciones del modelo para tomar una decisión más correcta sobre el encarcelamiento de un individuo.

2.4.5. Sector Empresarial

Algunas de las aplicaciones que tiene el análisis predictivo en este sector son las siguientes:

Retención de clientes

Consiste en un modelo capaz de predecir qué clientes tienen mayor probabilidad de abandonar a la empresa. De este modo las empresas pueden orientar sus esfuerzos en retener a dichos clientes.

Recomendaciones de películas/canciones

Consiste en un modelo capaz de predecir qué puntuación le dará una persona a una película/canción. Gracias a esta aplicación, las empresas como Netflix o Spotify pueden realizar recomendaciones a clientes sobre películas/canciones que tengan una alta probabilidad de gustarle.

3. PRINCIPALES HERRAMIENTAS DE ANÁLISIS PREDICTIVO

En 1969 nace SPSS de la mano de IBM, SAP AG presenta en 1973 su SAP R/1, antecesor de la actual SAP Business Suite y en 1976 el SAS *Institute* lanza su SAS *Software Package*. Esas primeras herramientas enfocadas al análisis de datos nacieron para ser ejecutadas en costosos *mainframes* y eran utilizadas por grandes corporaciones, universidades y gobiernos.

La generalización de las TI ha permitido la "democratización" del almacenamiento de datos y del procesado de los mismos pues a la bajada del coste de los equipos informáticos y al incremento de sus prestaciones se ha unido el *cloud computing* (la computación en la nube), que pone a disposición de las empresas capacidades de almacenamiento y proceso impensables hace unos pocos años, al tiempo que garantizan la accesibilidad a aplicaciones y datos.

Ante este escenario de nuevas oportunidades, los vendedores están respondiendo creando nuevo software que elimina la complejidad matemática, proporciona interfaces gráficas fáciles de usar y/o incluye atajos que permiten, por ejemplo, reconocer el tipo de datos disponibles y sugerir un modelo predictivo apropiado. Las herramientas de análisis predictivo se han vuelto lo suficientemente sofisticadas como para presentar y diseccionar adecuadamente los problemas de datos, de modo que cualquier informador de datos pueda utilizarlos para analizar datos y obtener resultados significativos y útiles. Por ejemplo, las herramientas modernas presentan hallazgos usando tablas simples, gráficos y puntuaciones que indican la probabilidad de posibles resultados.

Forbes, en su informe *Five Ways Data Analytics Will Shape Business, Sports And Politics In 2016*, señala que la demanda creciente de análisis de datos está provocando una presión sin precedentes sobre el mercado de analistas, lo que ha supuesto que la demanda de profesionales del sector en EEUU crezca a una tasa del 27% anual, muy por encima de la media, que se sitúa en el 11%.

Más allá de la presión sobre los salarios, y siempre según Forbes, la escasez de profesionales y la necesidad de que sean los propios responsables de las áreas que consumen los análisis los que obtengan los datos en tiempo real, lleva a que las empresas apuesten por lo que denominan en el artículo como "autoservicio". Esa tendencia implica la necesidad de herramientas cada vez más sencillas de utilizar

Existen numerosas herramientas disponibles en el mercado que ayudan con la ejecución de análisis predictivo. Estas van desde aquellas que necesitan escasos conocimientos por parte de los usuarios a aquellas que requieren de usuarios con una formación específica. La diferencia entre estas herramientas a menudo se encuentra en el nivel de personalización que ofrecen y la capacidad de trabajar con un número elevado de datos.

Existen varias herramientas comerciales que permiten realizar funciones de análisis predictivo como MATLAB o SAP, sin embargo este sector no es objeto

de estudio de este trabajo, que se centra en dos de las herramientas de código abierto más utilizadas: R y Weka.

No es objeto de este TFG analizar en profundidad los beneficios que genera el uso de herramientas de código abierto, pero resulta obligado señalar algunas de las ventajas que comporta optar por software libre.

En primer lugar garantiza la independencia tecnológica con respecto a los grandes proveedores de software tanto por el almacenamiento de datos en formatos abiertos, que permiten migrar a entornos diferentes, como por la posibilidad de mejorar las herramientas.

En segundo lugar, la utilización de software libre, gratuito en su mayor parte, contribuye a generar un tejido complementario de empresas o autónomos capaces de generar valor añadido mediante la instalación, personalización y desarrollos a medida sobre el mismo, con lo que parte del gasto en TI puede quedar en el entorno local.

Con carácter general son mucho más seguras, un aspecto fundamental a tener en cuenta dada la tendencia imparable a que datos y aplicaciones corran en la nube, o al menos en la red. Este plus de seguridad se debe a la cultura imperante en el software libre tanto a la hora de hacer públicas las vulnerabilidades, lo que contribuye a la inmediatez de su solución, como a la posibilidad de que terceros independientes examinen el código en busca de *bugs* o, en el peor de los casos, puertas traseras mal intencionadas.

En este apartado se analizarán dos herramientas de código abierto: R y Weka. La elección ha estado basada en los siguientes criterios: facilidad para la instalación de las mismas, documentación disponible y buena calidad de la misma.

Un factor que ha contribuido a decantar la elección en favor de R y Weka ha sido la potencia de la comunidad de usuarios de las mismas. En ese sentido, las estadísticas de StackOverflow son concluyentes:

Herramienta	**Consultas**	**Seguidores**
R	165.571	46.000
Weka	2.374	477
KNIME	111	103
Orange	166	40

En los próximos apartados se crearán modelos predictivos con algunas de las técnicas como: árboles de decisión, utilizando el algoritmo C5.0 en R y el J48 en Weka; modelo de agrupamiento, utilizando el algoritmo k-means en R y el SimpleKMeans en Weka; y por último, se creará un modelo de reglas de asociación, mediante el algoritmo apriori en ambos casos. Para cerrar este apartado se hará una comparativa entre las herramientas.

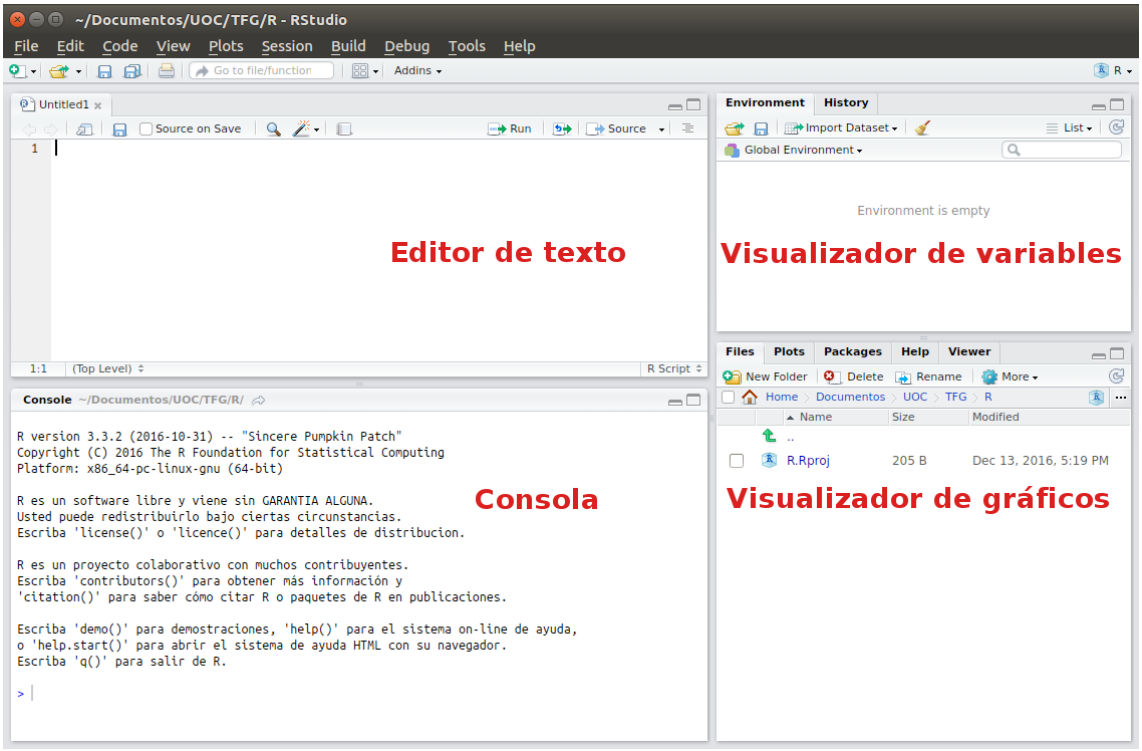
3.1. Estudio de la herramienta R

GNU R es un entorno de software de código abierto para la informática estadística. Utiliza “paquetes”, que se cargan con comandos en una consola, para proporcionar funcionalidades de modelado. R ofrece una amplia variedad de funcionalidades estadísticas como modelado lineal y no lineal, clasificación o agrupamiento. También ofrece la posibilidad de manipular los datos, realizar cálculos sobre ellos y representarlos mediante gráficas.

El lenguaje R es ampliamente utilizado entre los estadísticos y los mineros de datos para el desarrollo de software estadístico y análisis de datos. Este entorno se puede ejecutar en una amplia variedad de plataformas UNIX, en Windows y MacOS.

Para analizar R se utilizará RStudio, que es un entorno de desarrollo integrado (IDE) para R, que incluye una consola, un editor resaltado de sintaxis que soporta la ejecución directa de código, así como herramientas para representar los datos gráficamente, depurar y gestionar el espacio de trabajo.

Tras realizar la instalación de R[4] y RStudio[5] (no se entra en detalle puesto que no es objeto del presente trabajo) el programa presenta el siguiente aspecto:



RStudio permite cargar conjuntos de datos en el entorno desde múltiples formatos (como `.csv`, `.xlsl`, `.txt`, `.json`, `.dbf` o `.xml`) y realizar sobre ellos los pasos previos necesarios para la generación de modelos predictivos. Desde

4 Para más información acerca de la instalación visitar https://www.r-project.org/
5 Para más información acerca de la instalación visitar https://www.rstudio.com/

este entorno se pueden efectuar tareas de procesamiento de datos como: normalizar y/o estandarizar, eliminar o generar nuevos atributos, análisis de componentes principales o aplicar un tratamiento a los valores vacíos.

R es extensible mediante el uso de packages (paquetes) que incorporan funciones adicionales a las proporcionadas por R. El catálogo de paquetes es muy amplio y se pueden destacar, a modo de ejemplo, SQLdf, RODBC, RpostgresSQL y RSQLite, que permiten cargar datos directamente de una base de datos, o ggplot2 y rgi que proporcionan la capacidad de generar gráficos en 2D y 3D. Shiny es capaz de generar gráficos interactivos que pueden ser visualizados en la web mientras que caret permite analizar la calidad de los datos, seleccionar características y construir modelos predictivos.

En los siguientes apartados se muestra la ejecución de tres modelos, se explicará el proceso paso a paso y se analizarán los datos mediante gráficas:

3.1.1. Modelo de clasificación. Árbol de decisión C5.0

Eliminar las incertidumbres, o al menos disminuirlas, a la hora de adoptar decisiones en una empresa o una administración pública es un objetivo fundamental del análisis predictivo. En ningún caso puede pretenderse adivinar lo que va a suceder, pero si puede a aspirarse a encontrar reglas de carácter general que orienten la toma de decisiones.

Como caso tipo puede pensarse en la apertura de nuevos establecimientos por una cadena comercial. Para cada una de las nuevas tiendas a de decidirse la población en la que resulta más adecuado abrirla y en que zona de la ciudad ha de situarse para llegar al público objetivo. También ha de determinarse la superficie óptima y el surtido más adecuado para la zona y clientela a la que se pretende llegar. Habrán de valorarse los efectos de la competencia si existe e, incluso, que incidencia pueden tener otros establecimientos de la propia empresa. En la medida en la que se disponga de un conjunto de datos sólido y significativo el análisis predictivo puede dar respuesta a esos interrogantes.

Para hacer esta tarea se puede usar el modelo de árbol de decisión, para cuya construcción se identifica cuál es la mejor secuencia de preguntas para saber, a partir de las características de cada una de las tiendas, a qué clase corresponde. Evidentemente, “la mejor secuencia” puede entenderse como aquella que, con el mínimo número de preguntas, devuelve una respuesta los suficientemente detallada.

Para crear este modelo es necesario instalar el paquete C5.0, este paquete genera un modelo de minería que realiza una implementación moderna del algoritmo ID3 de Quinlan[6]. Tiene los principios teóricos del ID3 y además dis-

6 Para obtener más información acerca del algoritmo se puede visitar https://es.wikipedia.org/wiki/Algoritmo_ID3

pone de la poda automática. Este modelo permite crear un árbol de decisión o una colección de reglas.

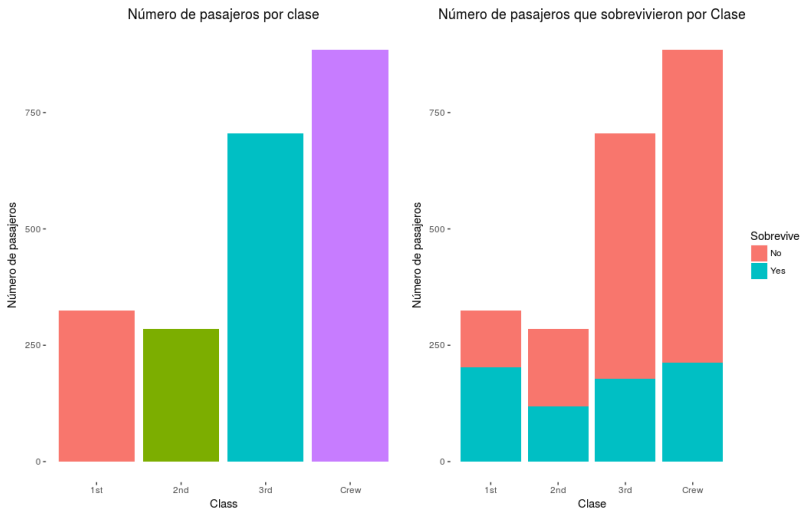
Lo primero que se debe de hacer es cargar los datos en el entorno de trabajo, en este caso se ha decidido trabajar con un conjunto de datos sobre los pasajeros del Titanic[7]. Para cada pasajero se dispone de la clase en la que viajaba (1ª, 2ª, 3ª o *crew*), la edad (*Adult* o *Child*), el sexo (*Male* o *Female*) y si sobrevivió o no. Una vez cargados los datos se creará un modelo que permitirá analizar qué tipo de persona tenía probabilidades de sobrevivir. Se utilizará un árbol de decisión para determinar si un pasajero sobreviviría o no.

R permite crear gráficos que ayudan a la hora de analizar y comprender los datos. En este caso se muestran una serie de gráficos como ejemplo, para los cuales se ha utilizado la librería `ggplot2`, además se muestra el código para generar uno de ellos:

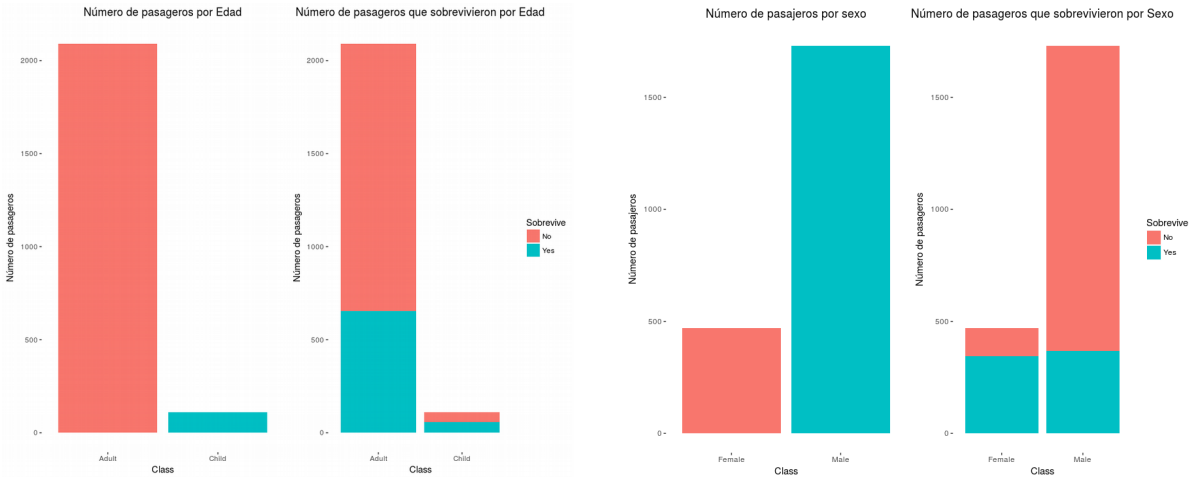
Código de muestra:

```
## Carga de librería

library(ggplot2)

## Gráfico de superviviencia por clase

ggplot(titanic.raw, aes(factor(titanic.raw$Class), fill =
factor(titanic.raw$Survived)))+

  geom_bar() +

  labs(x="Clase", y="Número de pasajeros") +

   ggtitle("Número de pasajeros que sobrevivieron por Clase")
+

  guides(fill=FALSE) +

  theme(plot.title = element_text(hjust = 0.5)) +

  theme(panel.background = element_rect(fill= "white"))
```

Gráficos:



7 Este conjunto de datos se ha descargado de http://www.rdatamining.com/data



El siguiente paso es crear el modelo predictivo. Para ello, será necesario separar los datos en dos conjuntos: uno de entrenamiento sobre el cual se generará el modelo, y otro de prueba, sobre el cual se probará el mismo. Para crear el conjunto de entrenamiento se utilizará el 66% de los datos y el 33% restante se destinará para el conjunto de prueba, pero primero, para obtener unos mejores resultados es conveniente mezclar los datos. Los comandos utilizados para esta tarea son:

```
## Mezclar los datos

s <- runif(nrow(titanic.raw))

titanic <- titanic.raw[order(s),]

## Creamos un conjunto de entrenamiento con el 66% de los
datos y otro                                    de prueba para X e
Y

train <- titanic[1:1914,]

test <- titanic[1915:2201,]
```

Ahora ya se puede generar el modelo, pero primero hay que cargar el paquete C50:

```
## Cargar el paquete

library(C50)

## Crear modelo

model <- C5.0(train[,-4], train[,4])
```

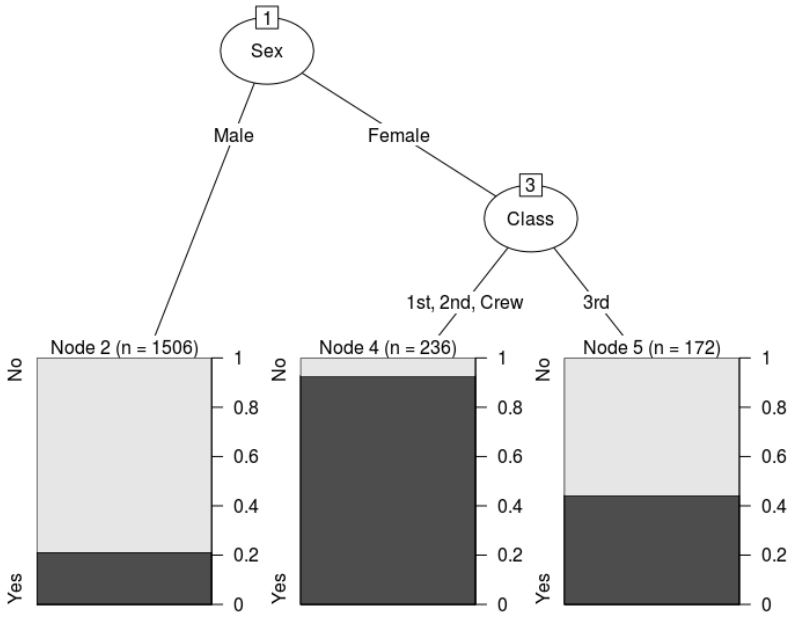
Una vez generado el modelo se puede analizar el mismo:

```
## Analizar modelo

summary(model)
```

```
Console ~/Documentos/UOC/TFG/R/

> ## Analizar modelo
> summary(model)

Call:
C5.0.default(x = train[, -4], y = train[, 4])


C5.0 [Release 2.07 GPL Edition]         Wed Dec 14 16:42:07 2016
-------------------------------

Class specified by attribute `outcome'

Read 1914 cases (4 attributes) from undefined.data

Decision tree:

Sex = Male: No (1506/317)
Sex = Female:
:...Class in {1st,2nd,Crew}: Yes (236/17)
    Class = 3rd: No (172/76)


Evaluation on training data (1914 cases):

           Decision Tree
         ----------------
         Size      Errors

            3  410(21.4%)   <<


          (a)   (b)    <-classified as
         ----  ----
         1285    17    (a): class No
          393   219    (b): class Yes


        Attribute usage:

        100.00% Sex
         21.32% Class


Time: 0.0 secs

> |
```

E imprimir el árbol generado:

```
## Imprimimos

plot(model)
```



A partir del árbol de decisión se pueden extraer las siguientes reglas de decisión:

Sex = "*Male*" → Clase "No"

Sex = "*Female*" & *Class* = "1ª" or "2ª" or "crew" → Clase "Yes"

Sex = "*Female*" & *Class* = "3ª" → Clase "No"

Finalmente se hace la predicción con el conjunto de datos de prueba y se comprueba la precisión del modelo. Es aquí donde se puede contemplar la utilidad del modelo, ya que puede clasificar a instancias (personas) de las cuales no se conocía su clase por los valores de sus atributos.

Para mostrar la matriz con los resultados se utiliza el paquete `gmodels`.

```
## Hacer predicción
prediction <- predict(model, test[,-4])
## Cargamos el paquete
library(gmodels)
## Imprimimos matriz de resultados
CrossTable(test[,4], prediction,
           prop.chisq = FALSE, prop.c = FALSE, prop.r =
FALSE,
           dnn = c('Resultado real', 'Predicción'))
```

```
Console ~/Documentos/UOC/TFG/R/
> CrossTable(test[,4], prediction,
+            prop.chisq = FALSE, prop.c = FALSE, prop.r = FALSE,
+            dnn = c('Resultado real', 'Predicción'))


   Cell Contents
|-------------------------|
|                       N |
|         N / Table Total |
|-------------------------|


Total Observations in Table:  287


               | Predicción
Resultado real |        No |       Yes | Row Total |
---------------|-----------|-----------|-----------|
            No |       185 |         3 |       188 |
               |     0.645 |     0.010 |           |
---------------|-----------|-----------|-----------|
           Yes |        64 |        35 |        99 |
               |     0.223 |     0.122 |           |
---------------|-----------|-----------|-----------|
  Column Total |       249 |        38 |       287 |
---------------|-----------|-----------|-----------|


>
```

En la diagonal desde la esquina superior izquierda (No/No) a la esquina inferior derecha (Yes/Yes) se encuentran las instancias bien clasificadas. El siguiente paso es calcular la precisión del modelo:

```
## Comprobar la precisión del modelo

sum ( prediction==test[,4])/length(prediction)
```

```
Console ~/Documentos/UOC/TFG/R/
> sum ( prediction==test[,4])/length(prediction)
[1] 0.7665505
>
```

El modelo generado ha obtenido un 76,65% de precisión.

Un ejemplo de uso de este tipo de modelos podría ser la clasificación del éxito (número de ventas) de un nuevo plato en una cadena de restaurantes. En función de sus atributos (tipo de comida, composición del plato, ubicación del restaurante, tipo de cliente…), el plato tendrá probabilidad de éxito o no.

3.1.2. Modelo de agrupación. K-means

A las empresas les interesa saber cuáles son sus diferentes grupos de clientes y en qué se parecen entre si.

Se trata de encontrar un método que permita discernir, de acuerdo con el contenido de la base de datos, qué clientes son parecidos, pero sin imponer ningún

criterio a priori. Es importante recalcar que no se indica que los clientes se agrupen porque tengan el mismo atributo (por ejemplo, el horario o la renta), o que se conozca a priori qué etiqueta de clase tiene cada uno, sino que, sin más información que la que aparece en los datos, se debe iniciar un proceso automático cuyo resultado proporcione una división del conjunto original de clientes en subconjuntos formados por clientes que se puedan reconocer como parecidos.

Los métodos de agregación, como K-means, pretenden encontrar precisamente las clases en que puede dividirse el dominio, el conjunto de observaciones.

K-means es un algoritmo de aprendizaje no supervisado que intenta agrupar datos basados en su similitud. El aprendizaje no supervisado significa que no hay resultado que predecir, y el algoritmo sólo trata de encontrar patrones en los datos. En este algoritmo hay que especificar el número de *cluster* (grupos) que en los que se quiere agrupar los datos. El algoritmo asigna de manera aleatoria cada observación a un grupo y encuentra el centroide de cada grupo. A continuación el algoritmo itera a través de dos pasos:

- Reasignar puntos de datos al grupo cuyo centroide es el más cercano.
- Calcular el nuevo centroide de cada grupo.

Estos pasos se repiten hasta que la variación dentro del grupo no se puede reducir más. La variación dentro del grupo se calcula como la suma de la distancia euclidiana entre los puntos da datos y sus respectivos centroides de agrupamiento.

Ahora se pasa a explicar cómo generar un modelo con RStudio, para este caso se ha utilizado un conjunto de datos que es quizás el más conocido en la literatura de reconocimiento de patrones, el conjunto de datos `iris.csv`. Este conjunto consta de 150 instancias con 5 atributos cada una (*Sepal.length, Sepal.width, Petal.length, Petal.width* y la clase), y hay una clase que es separable linearmente de las otras dos. El conjunto fue introducido por Ronal Fisher en 1936[8].

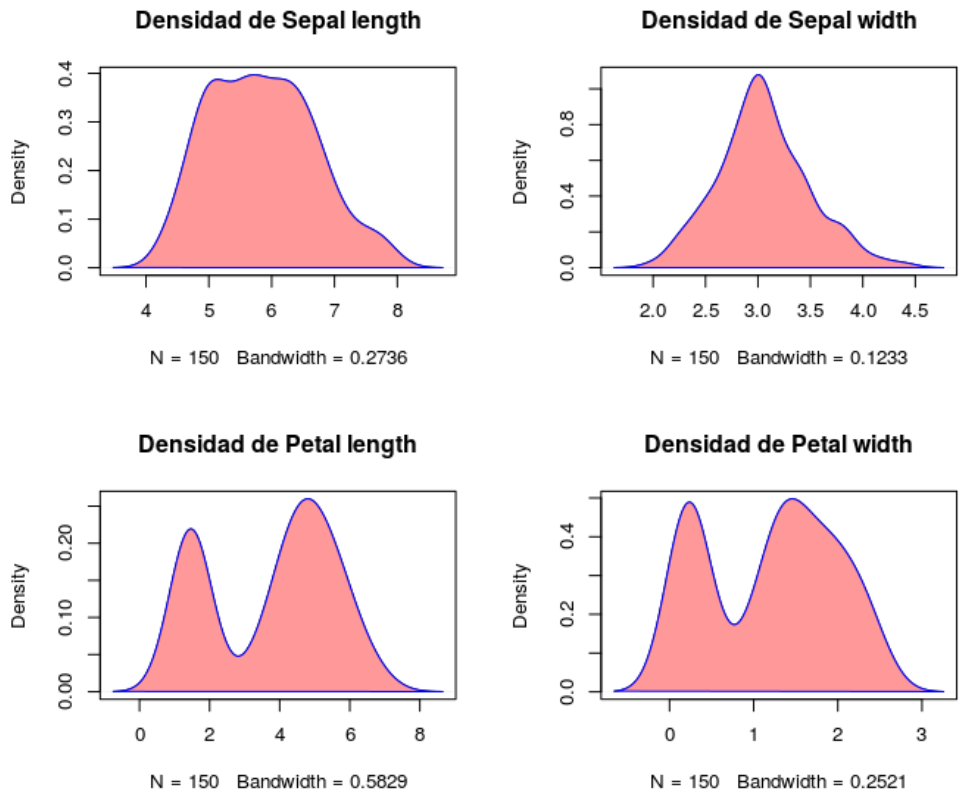
Lo primero será cargar los datos en el programa y comprobar que no dispone de valores vacíos. Una vez comprobado, se puede analizar un poco el conjunto de datos mediante el comando `summary(irisDF)`, que ofrece información acerca de cada atributo como: valores máximos y mínimos, la media, mediana, o cuartiles. R ofrece muchas posibilidades para analizar los datos gráficamente, como por ejemplo la densidad de cada atributo o la correlación entre los mismos. A continuación se muestra el ejemplo del código necesario para generar cada gráfica. Sin embargo, en las imágenes se mostrarán más gráficas de las generadas por el código mostrado.

Generar gráficas de densidades:

```
## Gráfica de Densidad de un atributo

denSL <- density(irisDF$Sepal.length)

plot(denSL, main="Densidad de Sepal length")
```

8 Para más información visitar https://es.wikipedia.org/wiki/Iris_flor_conjunto_de_datos

```
 polygon(denSL, col="#FF9999", border="blue")
```



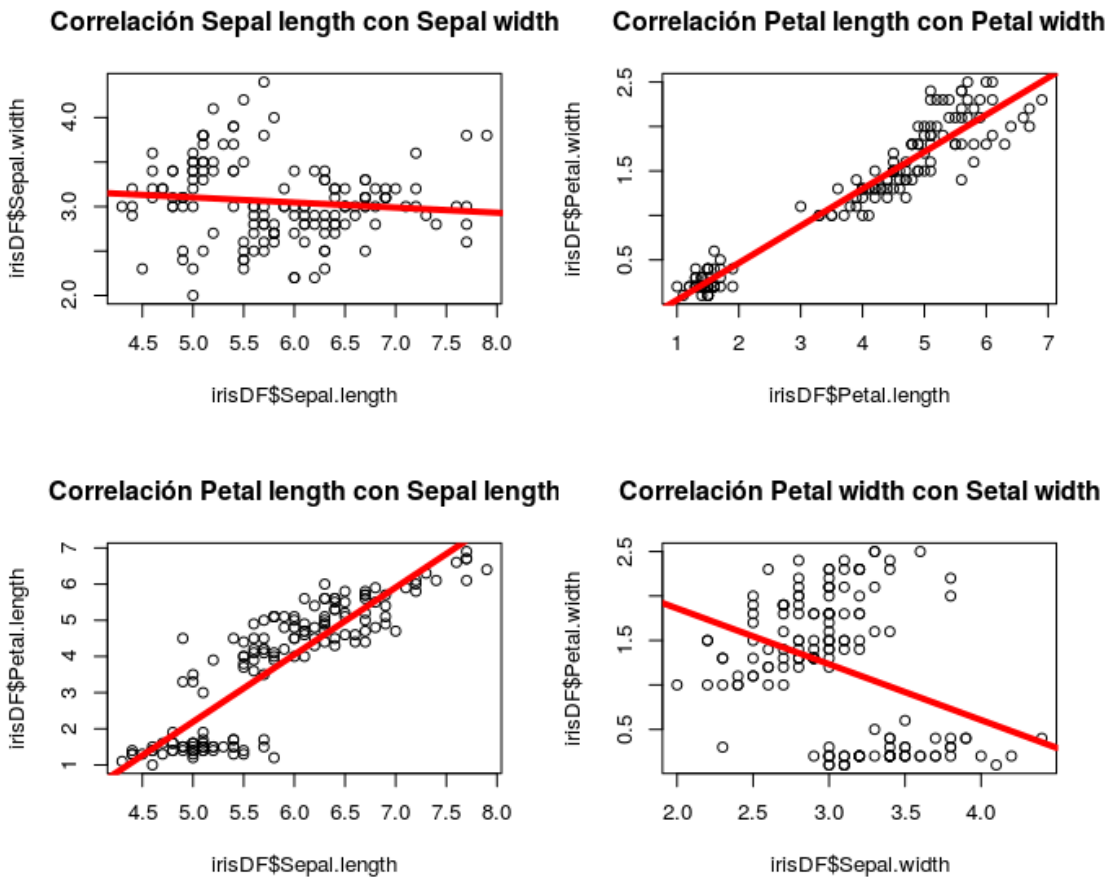
Gráficas obtenidas:

Con estas gráficas se puede analizar la distribución de cada una de las variables, en este caso se puede comprobar como las variables *Sepal length* y *Sepal width* son las que se asemejan más a una distribución normal.

Generar gráficas de correlación entre variables:

```
 ## Obtener gráfica correlación entre Sepal length y Sepal width

 plot(irisDF$Sepal.length, irisDF$Sepal.width, main="Correlación Sepal length con Sepal width")

 abline(lm(irisDF$Sepal.width~irisDF$Sepal.length), col="red", lwd=4)
```

Gráficas obtenidas:



Mediante estas gráficas se puede analizar la correlación entre los atributos. La correlación determina la relación o dependencia que existe entre dos variables que intervienen en una distribución bidimensional. Esta correlación determina si los cambios en una de las variables influyen en los cambios de otra. En caso de que esto suceda, se dice que las variables están correlacionadas o que hay una correlación entre ellas.

A parte del diagrama de dispersión, se ha añadido a la gráfica la recta de mínimos cuadrados[9], mediante la inclinación de esta recta se puede determinar si la correlación es positiva (crece cuando avanza hacia la derecha) o negativa, o si la correlación es fuerte (las instancias del diagrama de dispersión se ajustan a la recta) o débil.

En este caso se puede comprobar como la mayor correlación se encuentra entre los atributos *Petal length* y *Petal width*, que disponen de una fuerte correlación positiva puesto que cuando aumenta un atributo, también aumenta el otro.

Ahora ya se puede pasar a generar el modelo. Lo primero que hay que hacer es eliminar el atributo que indica la clase:

```
## Eliminamos el atributo clase

irisNoClass <- irisDF[-5]
```

9 Para más información visitar https://es.wikipedia.org/wiki/M%C3%ADnimos_cuadrados

Hay que tener en cuenta que a este algoritmo es necesario indicarle el número de agrupamientos que debe generar. En este caso, al tratarse de un conjunto al cual se le conocen las clases, no es necesario hacer este paso, sin embargo, se explicará una forma de estimar el mejor número de grupos posible.

El paquete `NBClust`[10] permite determinar cuál es el mejor número de grupos en un conjunto de datos. Proporciona 30 índices para determinar el número óptimo de grupos en el conjunto de datos y ofrece al usuario el mejor esquema de agrupación desde diferentes resultados.

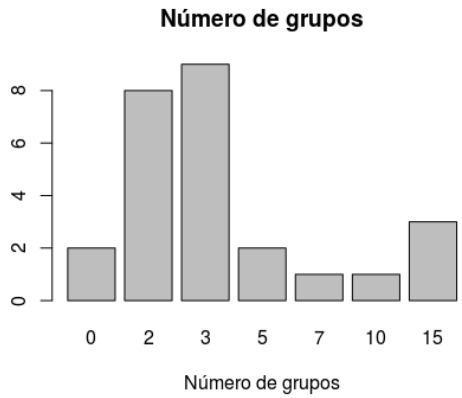
Con este paquete se puede utilizar la siguiente función:

```
## Estimar número de grupos

numberOfCluster <- NbClust(irisNoClass, min.nc=2, max.nc=15,
    method="kmeans")
```

```
Console ~/Documentos/UOC/TFG/R/
> numberOfCluster <- NbClust(irisNoClass, min.nc=2, max.nc=15, method="kmeans")
*** : The Hubert index is a graphical method of determining the number of clusters.
                In the plot of Hubert index, we seek a significant knee that corresponds to a
                significant increase of the value of the measure i.e the significant peak in Hubert
                index second differences plot.

*** : The D index is a graphical method of determining the number of clusters.
                In the plot of D index, we seek a significant knee (the significant peak in Dindex
                second differences plot) that corresponds to a significant increase of the value of
                the measure.

*******************************************************************
* Among all indices:
* 8 proposed 2 as the best number of clusters
* 9 proposed 3 as the best number of clusters
* 2 proposed 5 as the best number of clusters
* 1 proposed 7 as the best number of clusters
* 1 proposed 10 as the best number of clusters
* 3 proposed 15 as the best number of clusters

                   ***** Conclusion *****

* According to the majority rule, the best number of clusters is  3


*******************************************************************
>
```

También se pueden obtener los resultados en una gráfica:

```
barplot(table(numberOfCluster$Best.n[1,]), xlab="Número de
grupos", main="Número de grupos")
```



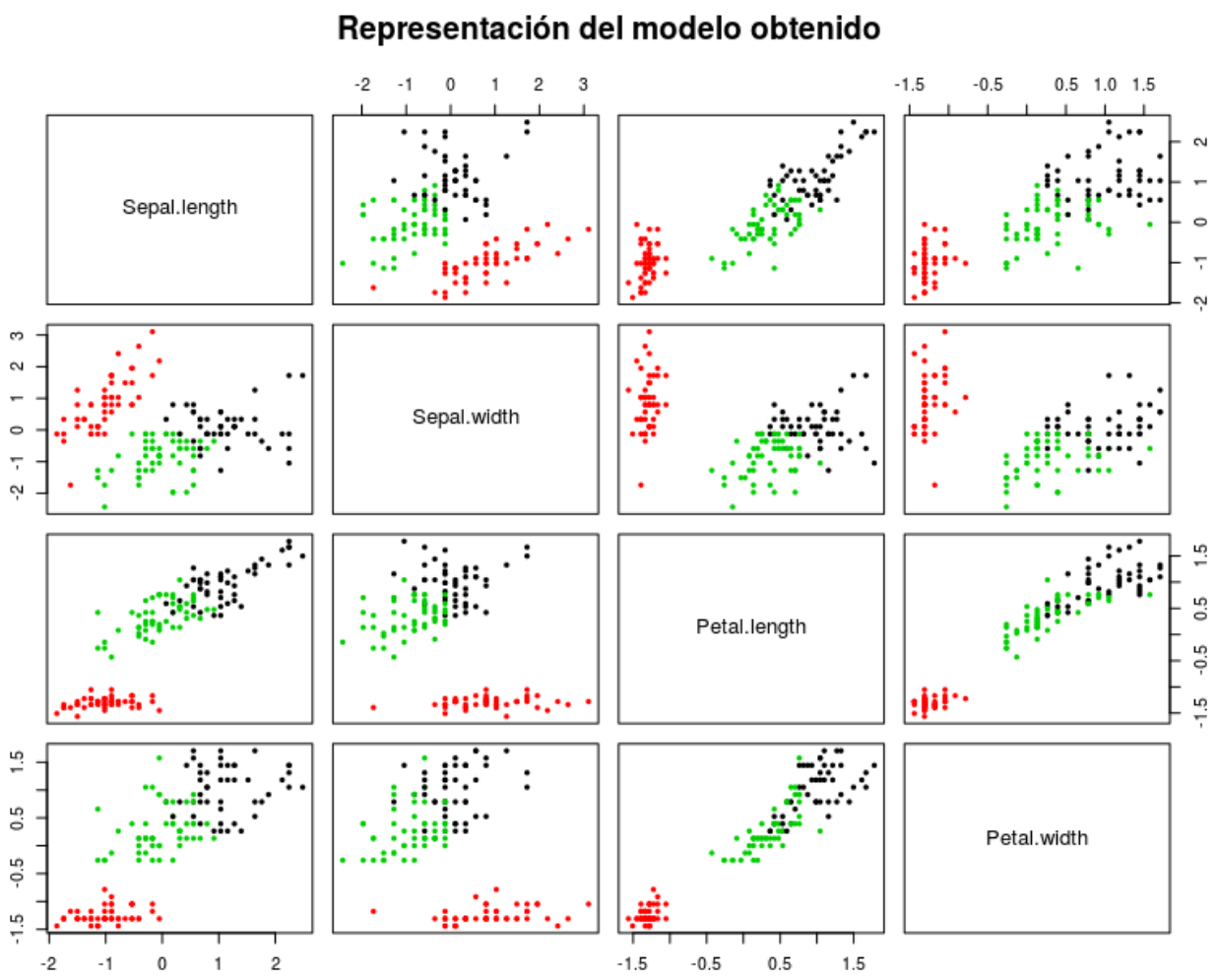
10 Para más información visitar https://cran.r-project.org/web/packages/NbClust/NbClust.pdf

Tal y como se puede comprobar, el número de grupos que ofrece mejores resultados es 3. Ahora se creará el modelo, pero antes se deben estandarizar los datos:

```
## Estandarizamos los datos

irisNoClass <- as.data.frame(scale(irisNoClass))

## Creamos el modelo con 3 cluster

model <- kmeans(irisNoClass, centers = 3, iter.max = 15)
```

Ahora se puede crear una gráfica con los resultados del modelo:

```
plot(irisNoClass, col = model$cluster, main = "Representación
del mode   lo obtenido", pch=20,cex=0.75)
```



Como se puede comprobar en la gráfica, se han marcado las instancias en 3 colores según el grupo.

También se puede comprobar cómo se han clasificado las instancias respecto a su clase real:

```
## Comprobar las instancias clasificadas con su clase real
```

```
 library(gmodels)

 CrossTable(iris$Species, model$cluster,

                 prop.chisq = FALSE, prop.c = FALSE, prop.r =
FALSE,

             dnn = c('Resultado real', 'Predicción'))
```

```
Console ~/Documentos/UOC/TFG/R/
> CrossTable(iris$Species, model$cluster,
+            prop.chisq = FALSE, prop.c = FALSE, prop.r = FALSE,
+            dnn = c('Resultado real', 'Predicción'))


   Cell Contents
|-------------------------|
|                       N |
|         N / Table Total |
|-------------------------|


Total Observations in Table:  150


               | Predicción
Resultado real |         1 |         2 |         3 | Row Total |
---------------|-----------|-----------|-----------|-----------|
        setosa |         0 |        50 |         0 |        50 |
               |     0.000 |     0.333 |     0.000 |           |
---------------|-----------|-----------|-----------|-----------|
    versicolor |        11 |         0 |        39 |        50 |
               |     0.073 |     0.000 |     0.260 |           |
---------------|-----------|-----------|-----------|-----------|
     virginica |        36 |         0 |        14 |        50 |
               |     0.240 |     0.000 |     0.093 |           |
---------------|-----------|-----------|-----------|-----------|
  Column Total |        47 |        50 |        53 |       150 |
---------------|-----------|-----------|-----------|-----------|


>
```

En esta tabla se puede comprobar como se han clasificado, en este caso se han clasificado bien 125 instancias de 150, por lo que el modelo ha obtenido una precisión del 0,83%.

3.1.3. Reglas de asociación

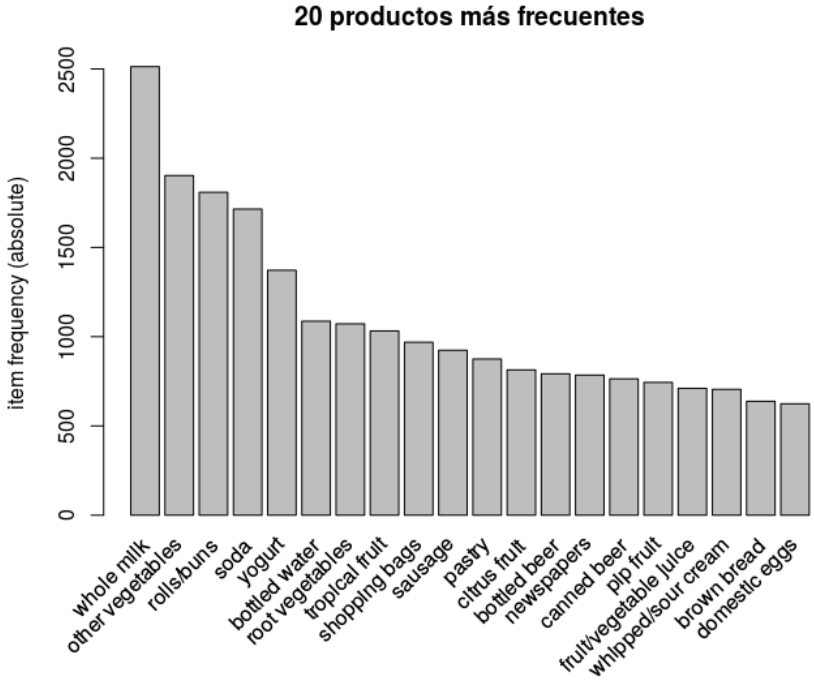
A las grandes superficies les interesa saber qué productos forman la cesta de la compra de sus clientes. El hecho de saber que dos productos tan dispares como la cerveza y los pañales aparecen con frecuencia en los grupos de productos registrados en las transacciones de los puntos de venta puede permitir redistribuir los objetos dentro de las estanterías de manera que su probabilidad de compra aumente. Con la información recogida en las cajas registradoras se construye una base de datos en la que los atributos son los diferentes productos. Cada atributo representa una transacción que corresponde a un único cliente, por ejemplo, leche, café, pan tostado, cerveza, pañales, azúcar o vino.

En este tipo de modelado no se consideran las cantidades y los precios de cada producto, no es relevante. El objetivo no es otro que conocer la composición de la cesta de la compra por tipo de producto.

Para crear un modelo de asociación es necesario utilizar el paquete `arules` y `arulesViz`. Junto con el paquete `arules` viene con un conjunto de datos (`Groceries`) que consiste en una colección de recibos en el que cada línea representa un recibo con los artículos comprados. Cada línea se denomina transacción y cada columna de una fila representa un elemento. Este es el conjunto de datos que se utilizará en este caso.

Tras cargar los paquetes necesarios y el conjunto de datos, se puede generar un diagrama de frecuencia con los 20 elementos más comprados:

```
 ## Cargar las librerias

 library(arules)

 library(arulesViz)

 library(datasets)

 ## Cargar el conjunto de datos

 data(Groceries)

 ## Crear gráfica de frecuencia de productos con los 20 más frecuentes

 itemFrequencyPlot(Groceries,topN=20,type="absolute", main="20 productos  más frecuentes")
```



El siguiente paso es generar las reglas de asociación. En este algoritmo siempre es necesario pasar el soporte mínimo requerido (es una indicación de la

frecuencia con la que el conjunto de elementos aparece en la base de datos) y la confianza (es una indicación de la frecuencia con que se ha encontrado que la regla es verdadera). En este caso se va a establecer un soporte mínimo de 0,001 y una confianza de 0,8. Una regla necesita un soporte de varios cientos de registros (transacciones) antes de que ésta pueda considerarse significativa desde un punto de vista estadístico. A menudo las bases de datos contienen miles o incluso millones de registros.

Tras crear las reglas, se pueden mostrar las 5 primeras:

```
## Obtener las reglas

reglas <- apriori(Groceries, parameter = list(supp = 0.001,
conf =     0.8))

## Mostrar las 5 primeras reglas con 2 dígitos

options(digits=2)

inspect(reglas[1:5])
```

```
Console ~/Documentos/UOC/TFG/R/
> reglas <- apriori(Groceries, parameter = list(supp = 0.001, conf = 0.8))
Apriori

Parameter specification:
 confidence minval smax arem  aval originalSupport maxtime support minlen maxlen target   ext
        0.8    0.1    1 none FALSE            TRUE       5   0.001      1     10  rules FALSE

Algorithmic control:
 filter tree heap memopt load sort verbose
    0.1 TRUE TRUE  FALSE TRUE    2    TRUE

Absolute minimum support count: 9

set item appearances ...[0 item(s)] done [0.00s].
set transactions ...[169 item(s), 9835 transaction(s)] done [0.00s].
sorting and recoding items ... [157 item(s)] done [0.00s].
creating transaction tree ... done [0.00s].
checking subsets of size 1 2 3 4 5 6 done [0.02s].
writing ... [410 rule(s)] done [0.00s].
creating S4 object  ... done [0.00s].
> ## Mostrar las 5 primeras reglas con 2 dígitos
> options(digits=2)
> inspect(reglas[1:5])
    lhs                      rhs           support confidence lift
[1] {liquor,red/blush wine} => {bottled beer} 0.0019  0.90       11.2
[2] {curd,cereals}          => {whole milk}   0.0010  0.91        3.6
[3] {yogurt,cereals}        => {whole milk}   0.0017  0.81        3.2
[4] {butter,jam}            => {whole milk}   0.0010  0.83        3.3
[5] {soups,bottled beer}    => {whole milk}   0.0011  0.92        3.6
>
```

Si se analiza la imagen se puede comprobar como las 5 reglas mostradas están ordenadas por *lift* (medida del desempeño de las regla de asociación), y la regla que ha obtenido unos mejores resultados es:

`{liquor, red/blush wine} → {bottled beer}`

Esta regla indica que existe una relación que dice que los clientes que compran licor y vino, el 90% (*confidence* de 0,9) de las veces también ha comprado cerveza.

Sin embargo, podría ser interesante ordenarlas por algún otro criterio, como puede ser la confianza:

```
## Ordenar reglas por confianza

reglas <- sort(reglas, by="confidence", decreasing=TRUE)

inspect(reglas[1:5])
```

```
Console ~/Documentos/UOC/TFG/R/
> reglas <-sort(reglas, by="confidence", decreasing=TRUE)
> inspect(reglas[1:5])
    lhs                                            rhs          support confidence lift
[1] {rice,sugar}                                => {whole milk} 0.0012  1          3.9
[2] {canned fish,hygiene articles}              => {whole milk} 0.0011  1          3.9
[3] {root vegetables,butter,rice}               => {whole milk} 0.0010  1          3.9
[4] {root vegetables,whipped/sour cream,flour} => {whole milk} 0.0017  1          3.9
[5] {butter,soft cheese,domestic eggs}          => {whole milk} 0.0010  1          3.9
>
```

Hay veces en el que las reglas son muy redundantes, por lo que hay que eliminarlas. Esto se consigue con la siguiente secuencia de acciones:

```
## Eliminar reduncancias

subset.matrix <- is.subset(reglas, reglas)

subset.matrix[lower.tri(subset.matrix, diag=T)] <- NA

redundantes <- colSums(subset.matrix, na.rm=T) >= 1

reglas.podadas <- reglas[!redundantes]

reglas<-reglas.podadas

summary(reglas)
```

Una vez se dispone de las reglas, es interesante poder orientar los resultados hacia productos en concreto, por ejemplo, el supermercado puede necesitar vender una mayor cantidad de un producto concreto, por lo que sería interesante ver junto a qué productos se suele vender.

Para hacer esto se crea un nuevo conjunto de reglas, indicando qué artículo en concreto interesa en la parte izquierda o derecha de la regla. Los siguientes comandos muestran cómo hacerlo, luego ordena las reglas y muestra las 5 que ofrecen una mayor confianza:

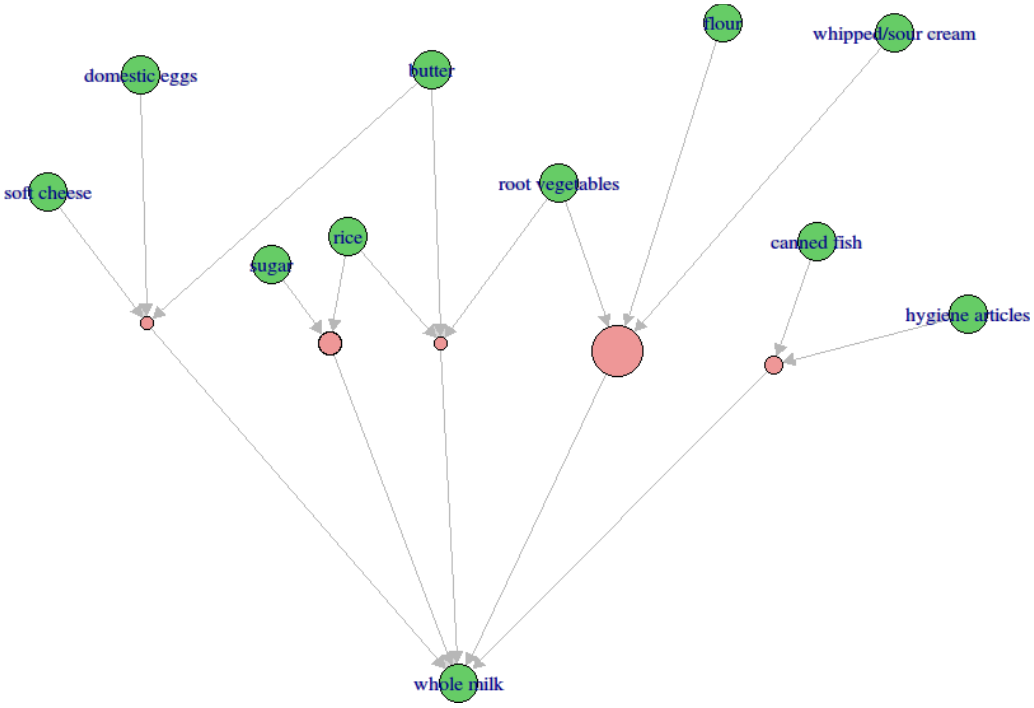
```
## Reglas con un objetivo concreto

reglasLeche <- apriori(data=Groceries,

               parameter=list(supp=0.001,conf = 0.08),

               appearance = list(default="lhs",rhs="whole milk"),

               control = list(verbose=F))

reglasLeche <- sort(reglasLeche, decreasing=TRUE,by="confidence")

inspect(reglasLeche[1:5])
```

```
Console ~/Documentos/UOC/TFG/R/
> reglasLeche <-apriori(data=Groceries, parameter=list(supp=0.001,conf = 0.08),
+                appearance = list(default="lhs",rhs="whole milk"),
+                control = list(verbose=F))
> reglasLeche<-sort(reglasLeche, decreasing=TRUE,by="confidence")
> inspect(reglasLeche[1:5])
    lhs                                            rhs          support confidence lift
[1] {rice,sugar}                                => {whole milk} 0.0012  1          3.9
[2] {canned fish,hygiene articles}              => {whole milk} 0.0011  1          3.9
[3] {root vegetables,butter,rice}               => {whole milk} 0.0010  1          3.9
[4] {root vegetables,whipped/sour cream,flour}  => {whole milk} 0.0017  1          3.9
[5] {butter,soft cheese,domestic eggs}          => {whole milk} 0.0010  1          3.9
>
```

De esta forma se han obtenido las reglas que derivan en la compra de la leche. También se podría hacer a la inversa, reglas que incluyen leche como artículo en la parte izquierda.

Ahora se podría visualizar las 5 reglas con mayor confianza mediante un gráfico, para ello se pueden utilizar el siguiente comando:

```
## Gráfico de las 5 reglas con mayor confianza

plot(reglasLeche[1:5],method="graph",interactive=TRUE,shading=NA)
```



Un ejemplo de uso de este tipo de modelos podría ser la recomendación de películas o música. Estos modelos ya los utilizan las grandes empresas como Netflix o Spotify para recomendar a sus clientes.

3.2. Estudio de la herramienta Weka

Wena (*Wakiato Environment for Knowledge Analysis*) es una colección de algoritmos de aprendizaje automático para tareas de minería de datos. Los algoritmos pueden ser aplicados directamente a un conjunto de datos o llamados desde su propio código Java. Weka contiene herramientas para hacer el procesamiento previo de datos, clasificación, regresión, agrupamiento, reglas de asociación y visualización. También es adecuado para desarrollar nuevos esquemas de aprendizaje de máquinas.

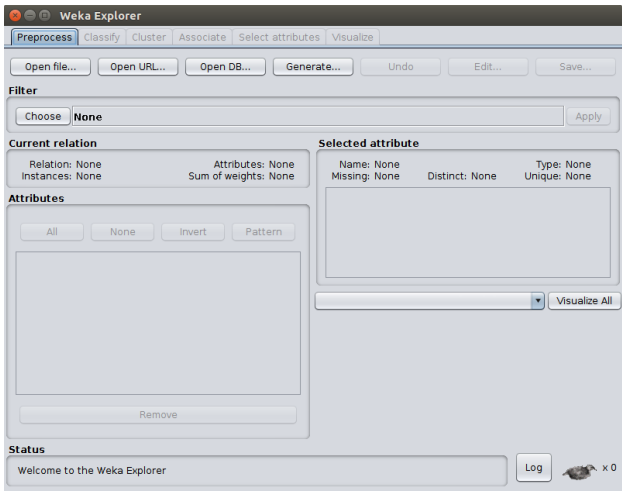
Weka es un software de código abierto emitido bajo la GNU *General Public Licence*.

Tras descargar el Weka[11], se puede abrir el entorno accediendo al directorio donde se ha descargado y ejecutando el siguiente comando:

```
java -jar weka.jar
```

Y aparecerá la siguiente imagen:

Ahora se seleccionará la aplicación `Explorer`, que presentará la siguiente apariencia:

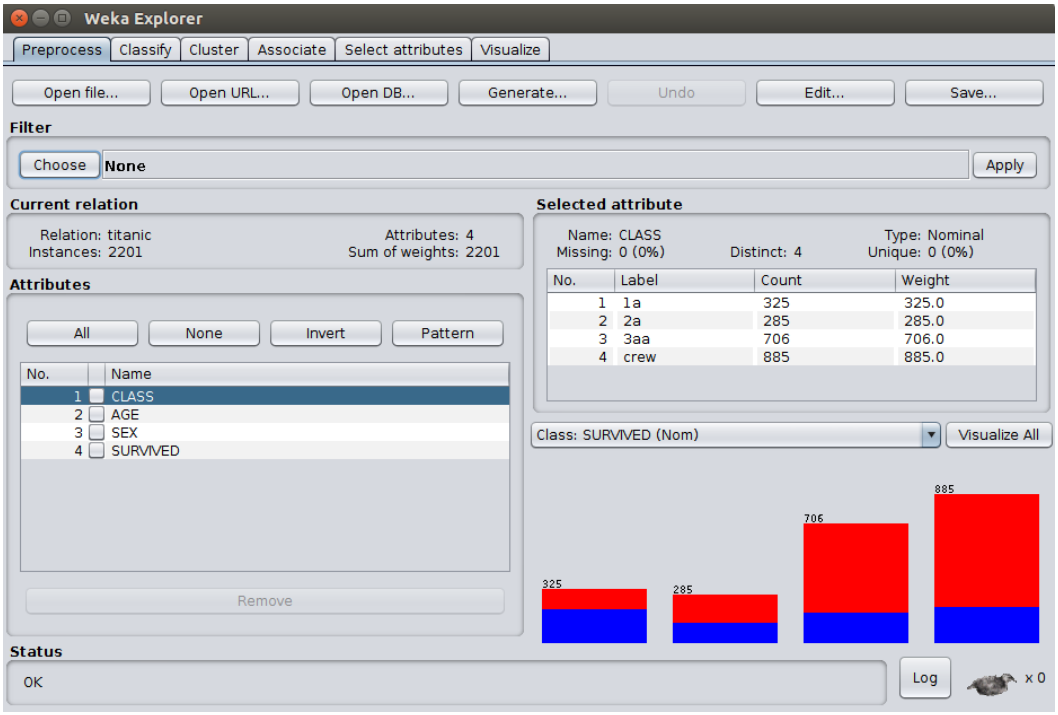


11 Para más información visitar http://www.cs.waikato.ac.nz/ml/weka/

En esta imagen se puede ver como el *Explorer* presenta varias pestañas que ofrecen diferentes funcionalidades como el preprocesamiento de los datos, la clasificación o agrupamiento. Ahora se analizará la primera pestaña de preprocesamiento, el resto se analizará en apartados posteriores.

Desde esta pestaña se cargan los datos en el programa, se puede hacer un análisis inicial de los mismos y aplicar tratamientos como normalizar o discretizar atributos mediante la aplicación de filtros, o eliminar alguno de los mismos. Los filtros de Weka proporcionan una manera sencilla de aplicar un tratamiento al conjunto de datos, ofreciendo métodos supervisados y no supervisados, que se pueden aplicar sobre los atributos o sobre las instancias.

En este caso se ha cargado el conjunto de datos utilizado en el análisis del modelado de árboles de decisión con R, el conjunto de datos `titanic.csv`:



Como se puede contemplar en la imagen, el programa ofrece información acerca del conjunto de datos, como el número de atributos, el número de instancias (totales y por atributo) o el nombre de los atributos. Además, ofrece una gráfica con los atributos, permitiendo elegir qué atributo se quiere representar y sobre qué otro atributo se quiere clasificar. En este caso se representa el atributo *CLASS* sobre el atributo *SURVIVED*.

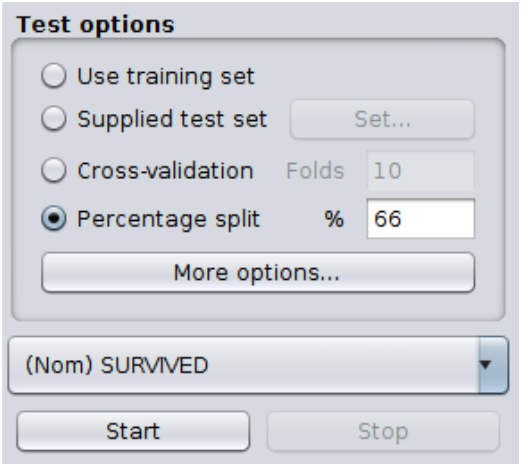
En los siguientes apartados se muestra la ejecución de tres modelos, se explicará el proceso paso a paso y se analizarán los datos mediante gráficas. Weka utiliza algoritmos propios que implementan las funcionalidades necesarias para generar los modelos.

3.2.1. Modelo de clasificación. Árbol de decisión

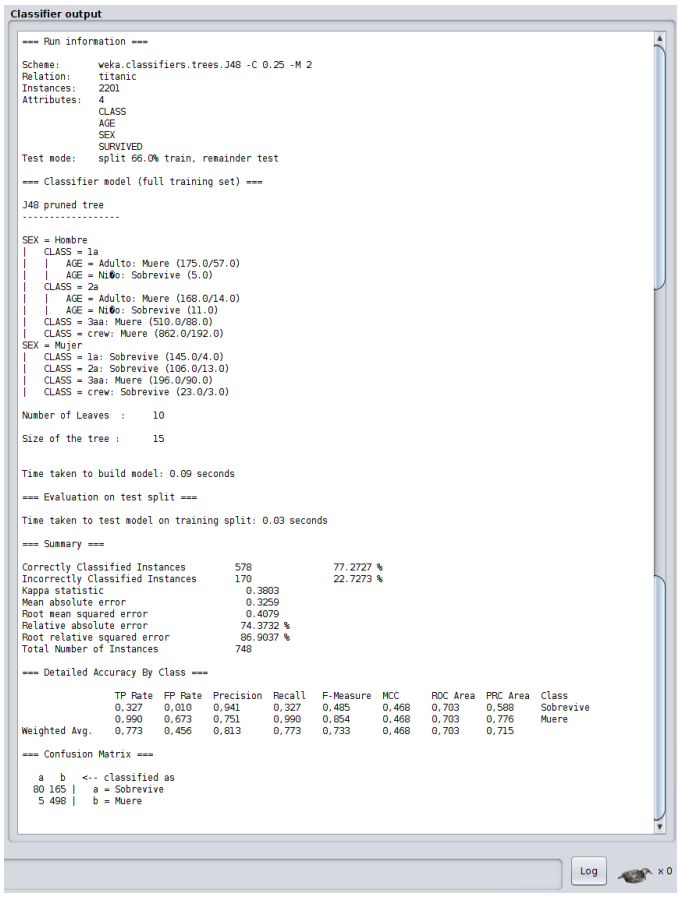
Tras cargar los datos en el programa, se pueden acceder a la pestaña *Classify* para clasificar los mismos. En este caso se ha utilizado el conjunto de datos `titanic.csv`, y se va a utilizar el árbol de clasificación J48, que se trata de una implementación *open source* en el lenguaje de programación Java del algoritmo C4.5 que ofrece Weka. Para ello, será necesario elegir el clasificador en el botón *Choose* y luego *classifiers → trees → J48*.

Este clasificador permite configurar cómo se ejecutará dentro de unos márgenes establecidos, como: proporcionar un conjunto de datos de entrenamiento o un conjunto de datos de prueba distinto de los datos ya cargados; utilizar el método de validación cruzada (garantiza que la partición entre los datos de entrenamiento y prueba son independientes) indicando el número de iteraciones; o dividir el conjunto actual en entrenamiento y prueba, indicando el porcentaje de cada conjunto). Para este caso se ha elegido la opción de dividir el conjunto de datos actual, indicando que el porcentaje será de un 66%, lo que significa que se utilizará el 66% de los datos para el entrenamiento y el 33% para la prueba.

El siguiente paso será elegir sobre qué atributo realizar la clasificación, para este caso el atributo *SURVIVED*, y darle al botón *Start*.

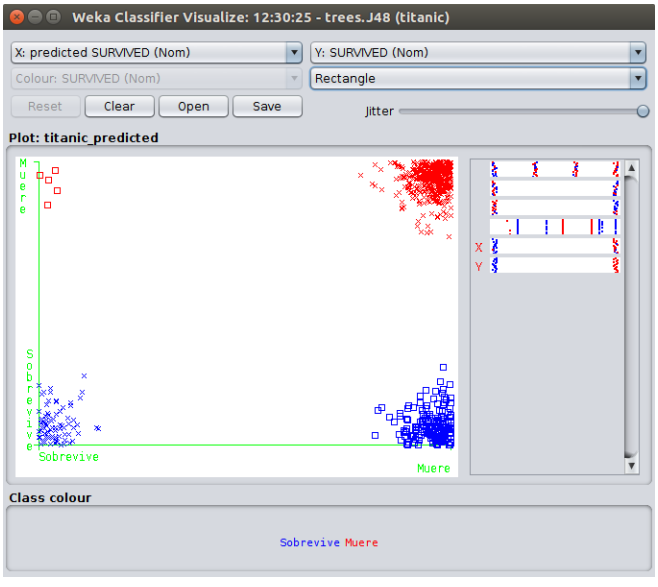


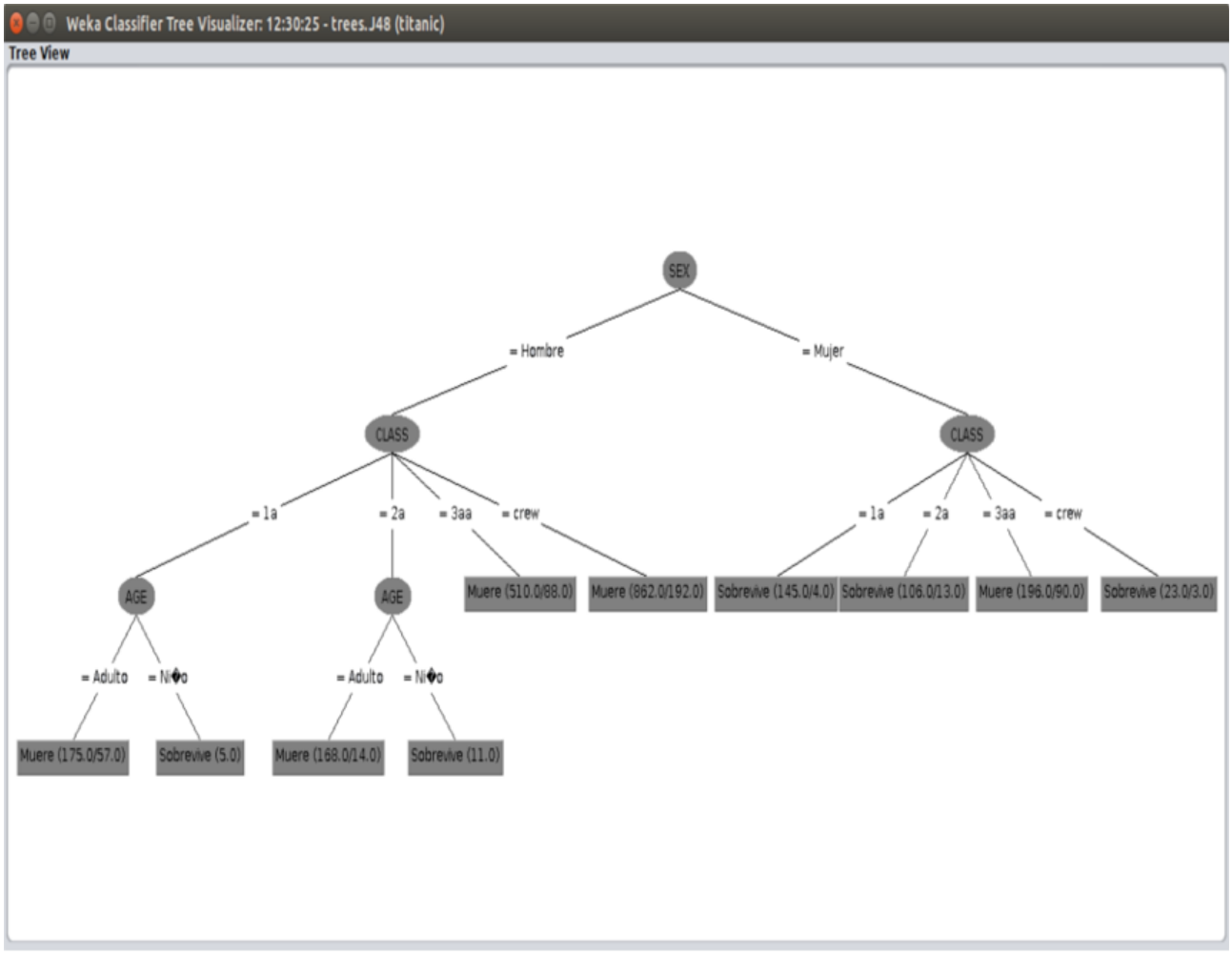
Una vez creado el modelo, el programa proporciona una salida:



De esta salida se obtiene mucha información acerca del modelo, como el algoritmo utilizado, el tamaño del árbol generado, el porcentaje de instancias bien y mal clasificadas o la matriz de confusión, donde se puede comprobar que se han obtenido 578 instancias bien clasificadas de 748.

Una vez generado el modelo se puede visualizar el clasificador de errores o el árbol obtenido.





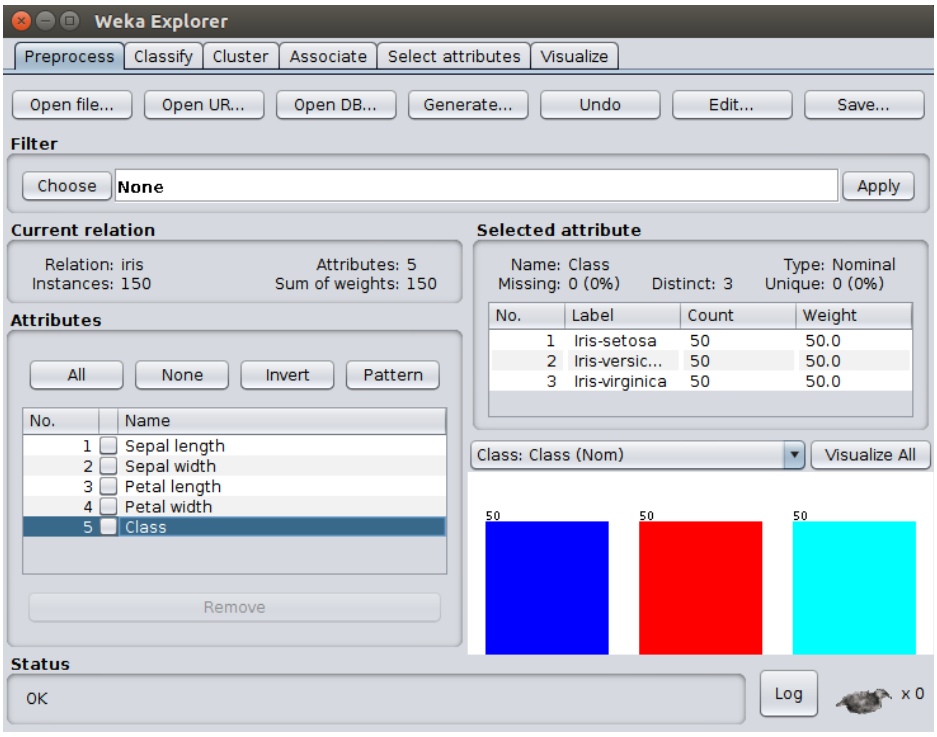
En esta imagen se puede ver el árbol resultante tras la aplicación del algoritmo. Con este árbol se podrían clasificar nuevas instancias, por ejemplo, un `Hombre`, de `2ª clase` y que fuera `Adulto` probablemente muriera, ya que, según el árbol obtenido, de las instancias con esas características murieron 168 y sobrevivieron sólo 14.

3.2.2. Modelo de agrupamiento. SimpleKMeans

El siguiente modelo a analizar es el de agrupamiento. Se utilizará el algoritmo SimpleKMeans implementado en Weka.

Lo primero será cargar un conjunto de datos. Se utilizará el mismo conjunto empleado en el análisis del modelo de agrupamiento generado con R, el conjunto `iris.csv`. Como se puede comprobar en la imagen que se adjunta a continuación, este conjunto de datos dispone de 150 instancias divididas en 3 grupos de 50 instancias cada uno, si se va seleccionando cada uno de los atri-

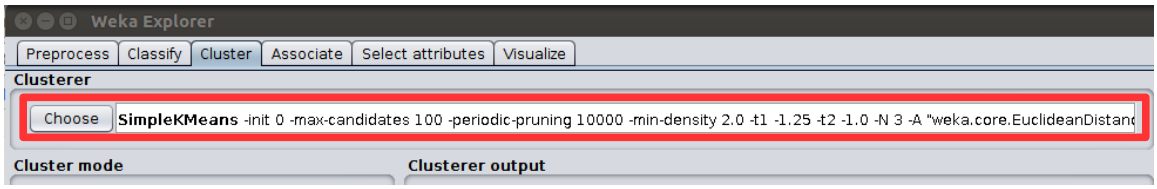
butos se obtiene información acerca del tipo de atributo. En este caso está seleccionado el atributo `Class`, que es nominal.



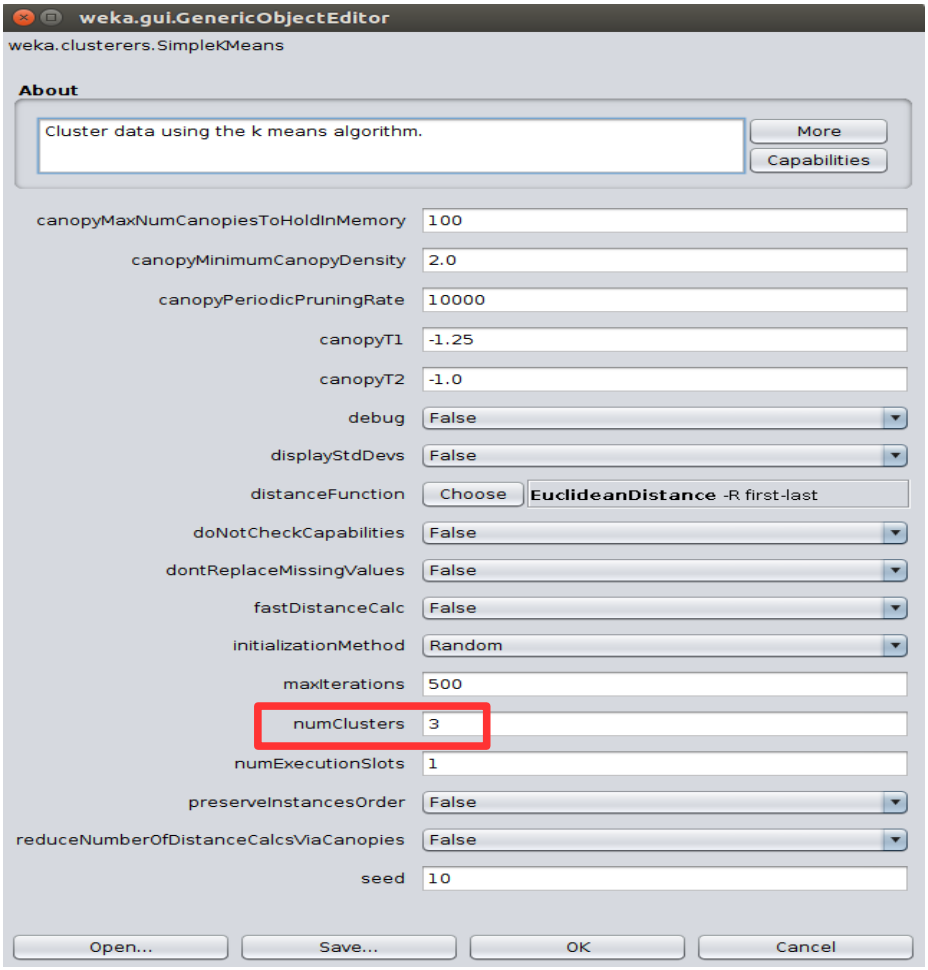
A este conjunto de datos no es necesario realizar ningún tratamiento previo, por lo que se podrá pasar directamente a la pestaña `Cluster` y comenzar a generar el modelo.

El siguiente paso es seleccionar el algoritmo a utilizar para generar el modelo, en este caso se ha decidido utilizar el SimpleKMeans, para ello se selecciona en `Choose` → `clusterers` → `SimpleKMeans`. Luego se configura las condiciones que utilizará el algoritmo, al igual que con el algoritmo utilizado en R, será necesario indicar el número de grupos que se quiere generar. Para ello habrá que analizar los datos y determinar el número de grupos ideal basando la decisión en hipótesis que se puedan extraer de los mismos. Debido a que se trata de un conjunto de datos del que se sabe el grupo al que pertenece cada instancia, se conoce que el número de grupos ideal es 3.

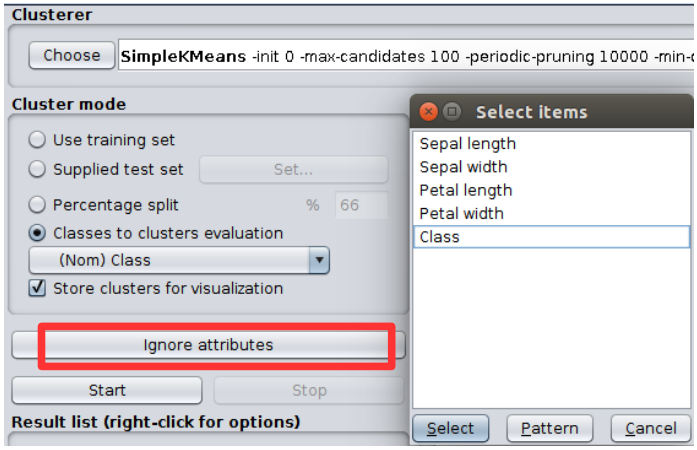
Ahora habrá que indicar el número de grupos, para ello se hará clic sobre el nombre del algoritmo:



Y se configurará el número de grupos, también se pueden configurar aspectos como la función de distancias a utilizar, si se quiere conservar el orden del conjunto de datos o el número máximo de iteraciones. En este caso, el resto de las campos se dejan con los valores por defecto:

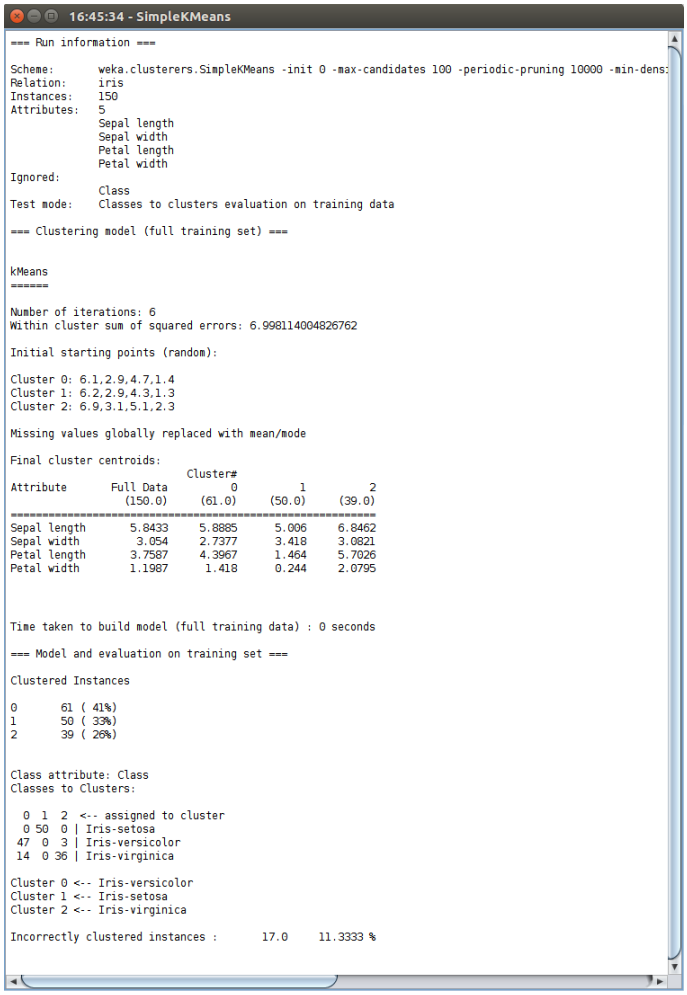


Al igual que cuando se configuró el árbol de decisión, Weka permite usar conjuntos de entrenamiento, proporcionar un conjunto de test, dividir el conjunto de datos o seleccionar un atributo sobre el que evaluar el modelo. En este caso se elige esta última opción y se selecciona el atributo `Class`. Además, dado que se trata de un conjunto de datos en el cual se dispone de la clase real de cada instancia, habrá que ignorar dicho atributo:



Ahora ya se puede generar el modelo haciendo clic sobre el botón `Start`.

A continuación se puede ver el resultado final del modelo creado, con información acerca de el número de iteraciones realizadas, los puntos iniciales de los grupos, los centroides finales obtenidos, el número de instancias de cada grupo, así como la matriz de confusión donde se reflejan como se clasificaron las instancias respecto a la clase original. En este caso se obtuvieron tan sólo 17 instancias mal, lo que supone un 11,33% de error, por lo que el modelo tiene 88,67% de precisión.



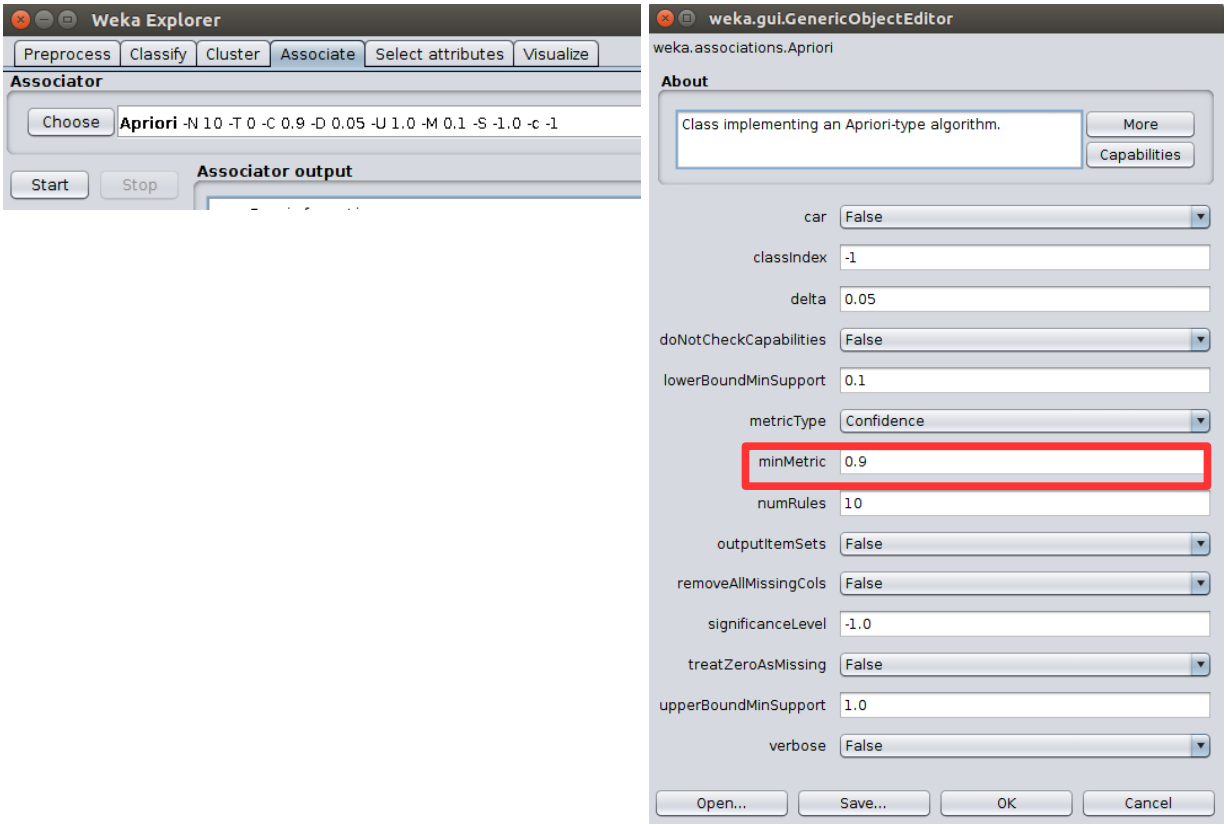
3.2.3. Reglas de asociación

El siguiente modelo a analizar es el de reglas de asociación. Se utilizará el algoritmo `Apriori` implementado en Weka.

Lo primero será cargar un conjunto de datos. Para este caso se utiliza un conjunto de datos que viene con el propio programa `supermarket.arff`, que es un conjunto de datos de información de punto de venta. Los datos son nominales y cada instancia representa una transacción del cliente en un supermer-

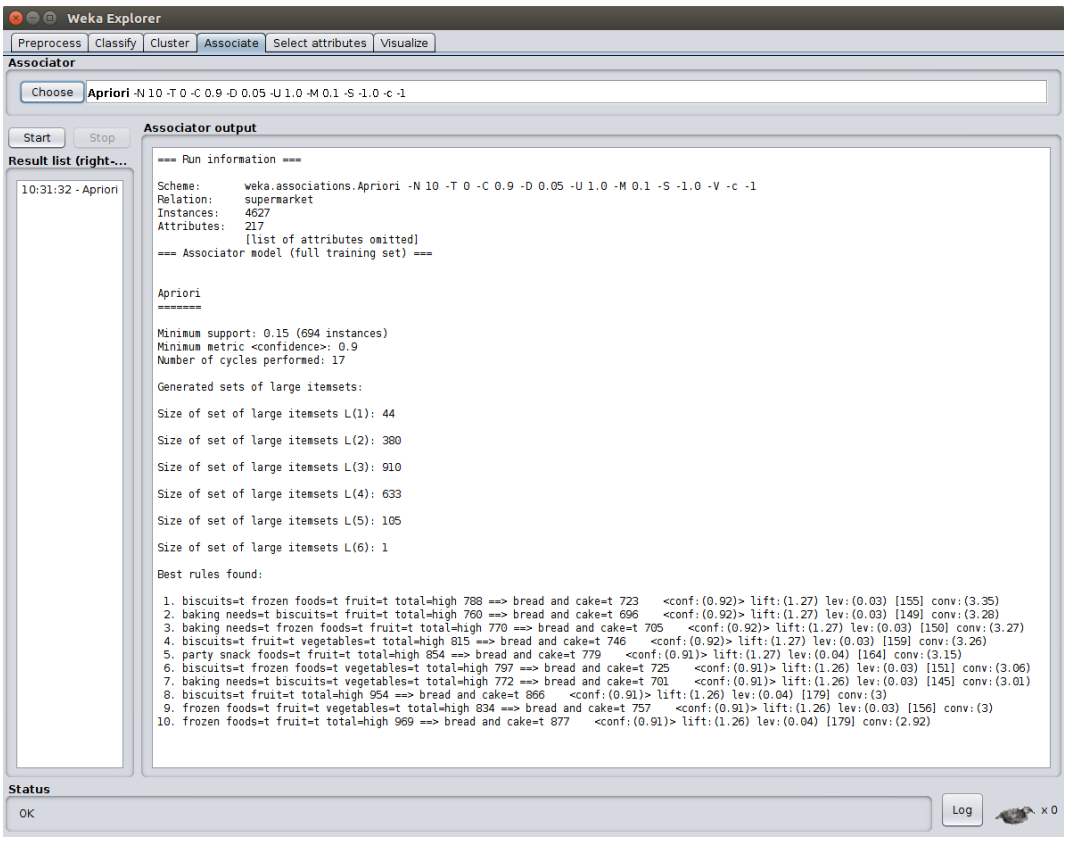
cado, los productos adquiridos y los departamentos involucrados. Los datos contienen 4.627 instancias y 217 atributos. Cada atributo es binario, menos uno, que es nominal llamado `total`, que indica si la transacción fue menor de 100$ (baja) o mayor (alta).

Una vez cargados los datos en el programa, basta con acceder a la pestaña `Associate` y luego seleccionamos el algoritmo a utilizar mediante el botón `Choose → associations → Apriori`. Se pueden configurar los valores que utilizará el algoritmo haciendo clic sobre el nombre del mismo, para que salga la siguiente pantalla:



Se pueden configurar aspectos como el soporte mínimo, el número de reglas a mostrar o la métrica utilizada (en este caso Confianza) y su valor mínimo (en este caso 0.9). Una vez configurado, para generar el modelo se hace clic sobre el botón `Start`.

El modelo generado ha sido el siguiente:



Como se puede comprobar en la imagen, el resumen ofrece información acerca del conjunto de datos utilizado (indicando número de instancias y atributos), el algoritmo utilizado (indicando el soporte y la confianza) y las 10 mejores reglas.

3.3. Comparativa

Una vez hecho el análisis de cada herramienta se puede hacer una comparativa entre ambas, remarcando las ventajas e inconvenientes que ofrecen cada una de ellas. Durante el análisis se ha podido comprobar como las dos herramientas permiten al usuario realizar labores de procesamiento previo de datos, análisis de los mismos, así como la aplicación de una serie de técnicas para generar modelos predictivos, sin embargo existen algunas diferencias entre ellas.

Dado que para crear cualquier modelo siempre se sigue una serie de pasos como: cargar los datos, analizarlos y procesarlos (si es necesario), para luego generar el modelo; es interesante comparar estos pasos con cada una de las herramientas:

Carga del conjunto de datos:

En este aspecto la interfaz gráfica que ofrece Weka facilita esta tarea, ya que permite seleccionar el archivo a cargar directamente. En cambio, en R se debe

de escribir el comando necesario, que debe adaptarse en función de las características del archivo a cargar, indicando si dispone de cabecera o no, o el formato en el que se encuentra el archivo.

En la carga del conjunto de datos Weka ofrece más facilidades a los usuarios, sin embargo es más restrictivo en cuanto a los formatos aceptados. En este punto R ofrece mejores resultados, puesto que es compatible con un rango de tipos de datos más amplio.

Analizar los datos:

Una vez se han cargado los datos se deben analizar los mismos. Es interesante conocer la estructura del conjunto de datos y saber cuántas instancias tiene, cuántos atributos y de qué tipo, así como la distribución de cada atributo, valores máximos y mínimos, medias, medianas… En este apartado sin duda alguna se obtienen mejores resultados con R, ya que Weka muestra unos resúmenes básicos, indicando tan solo el número de instancias y de atributos, y si se selecciona uno, indica el tipo de datos o cuántos valores distintos tiene cada atributo.

R permite analizar los datos con una mayor profundidad, ofreciendo información acerca de medias, medianas, cuartiles… pero no sólo eso, R también permite crear multitud de gráficas de las que se extrae información como la correlación existente entre dos atributos o la distribución que tiene un atributo. En cambio, Weka sólo ofrece poco más que unos gráficos de barras.

Procesamiento de datos previo:

Antes de generar los modelos, los datos deben de tener unas características concretas en función del modelo a ejecutar. Weka dispone de múltiples filtros con funcionalidades ya implementadas para procesar los datos, como filtros para discretizar o normalizar. En cambio, si se quiere llevar a cabo este tipo de tareas con R, es necesario indicarle al programa qué es lo que se quiere hacer de manera exacta, por lo que habrá que escribir líneas de código que implementen esas funciones.

Una vez más, Weka ofrece las funcionalidades de una manera sencilla, mientras que R requiere de unos conocimientos más completos. Sin embargo, cabe destacar que R, si se dispone de los conocimientos necesarios, ofrece una mayor libertad para procesar los datos, ya que no se encuentra condicionado a los filtros existentes.

Generar el modelo:

A la hora de realizar el modelo, una vez más, Weka ofrece mayor facilidad, ya que basta con seleccionar el tipo de modelo que se quiere crear, seleccionar el algoritmo (dentro del conjunto de algoritmos implementados en el programa) que se quiere utilizar para crearlo y configurar los parámetros. Por el contrario, con R es necesario cargar el paquete que dispone del algoritmo que implementa el modelo a crear, escribir el código necesario y ejecutarlo.

Para generar el modelo a veces se divide el conjunto de datos en un conjunto de entrenamiento y otro de prueba. Weka permite hacer esto de una manera muy sencilla, basta con indicar ese aspecto en la configuración previa a la crea-

ción del modelo, mientras que con R es necesario escribir las líneas de código que mezclen los datos y los separen en los dos conjuntos.

A continuación, se muestra una tabla que resume lo expuesto:

	R	Weka
Carga de datos	Dispone de un rango muy amplio de formatos aceptados. Sin embargo, el usuario debe de conocer el comando correcto para cargar cada uno de los datos, así como indicar aspectos como si dispone de cabecera o cómo se separan los datos.	La interfaz gráfica de usuario facilita esta tarea, ya que permite seleccionar el archivo a cargar directamente. En cambio, weka es más restrictivo respecto a los formatos aceptados.
Análisis de datos	Permite realizar un análisis en mayor profundidad obteniendo medias, medianas, cuartiles… además permite crear multitud de gráficas que permiten entender los datos.	La interfaz gráfica de usuario ofrece unos resúmenes básicos de cada uno de los atributos (tipo de datos, valores distintos de cada atributo, así como unos gráficos de barras).
Procesamiento de datos	Ofrece mayor libertad, puesto que permite realizar todo tipo de procesamiento. Sin embargo, el usuario debe de disponer de unos conocimientos más amplios, ya que debe de escribir el código que haga lo que quiere hacer.	La interfaz gráfica de usuario dispone de filtros ya implementados que facilitan esta tarea.
Generación de modelo	Es necesario cargar el paquete y escribir el código que lo ejecute.	La interfaz gráfica de usuario facilita enormemente esta tarea, ya que permite seleccionar el tipo de modelo a crear, el algoritmo a utilizar y configurar los parámetros.

Tras analizar esta serie de pasos que se deben seguir para generar el modelo, es llamativa la diferencia en cuanto a la facilidad de uso que presenta Weka frente a R. Weka dispone de una interfaz gráfica de usuario muy sencilla, que permite ejecutar la mayoría de los análisis más simples y crear modelos. No es necesario disponer de conocimientos de programación para utilizar esta herramienta, ya que su interfaz gráfica facilita este trabajo. Esto lo convierte en una herramienta muy útil para usuarios sin conocimientos de programación, como podrían ser los estadísticos. Debido a las facilidades que ofrece Weka, se puede considerar que la curva de aprendizaje es muy suave.

Por contra, con R, a pesar de que también ofrece una interfaz gráfica de usuario, todo se consigue ejecutando comandos desde una terminal, por lo que el usuario debe de escribir el mismo las instrucciones (al fin y al cabo, R es un lenguaje de programación). Esto hace que R tenga una curva de aprendizaje mucho más pronunciada que Weka, lo que puede frustrar a los usuarios que no dispongan de los conocimientos de programación previos, ya que tendrán que emplear mucho más tiempo en familiarizarse con R antes de poder sacar provecho a la herramienta. Sin embargo, una vez se dispone de los conocimientos necesarios R se convierte en una herramienta mucho más potente que da más libertad al usuario, frente a la poca flexibilidad que ofrece Weka.

Además de lo ya comentado, R tiene otras ventajas respecto a Weka como son: la posibilidad de exportar e importar paquetes que implementen determinadas funcionalidades; una mayor comunidad que usa el programa, lo que se traduce en más gente resolviendo problemas y generando paquetes que implementan funcionalidades de los cuales todo la comunidad se beneficia; la posibilidad de trabajar con un conjunto de datos mucho mayor, ya que Weka en algunos casos puede presentar problemas con grandes conjuntos de datos; o el formidable abanico de posibilidades que ofrece para mostrar los datos gráficamente, existiendo paquetes que permiten personalizar las salidas obtenidas con diferentes formas, colores, en 2D, 3D...

En definitiva, a pesar de que Weka permite usar sus algoritmos en cualquier programa escrito en Java, la herramienta no llega a alcanzar el potencial de R. Sin embargo, debido a las facilidades que ofrece al usuario y su curva de aprendizaje, este programa se puede valorar como una buena forma para familiarizarse con el aprendizaje computacional y la minería de datos. Es por eso por lo que este programa se puede considerar más como un programa orientado a la formación, mientras que R tiene un uso más amplio entre los usuarios en el sector profesional.

R y Weka no son unos recién llegados al campo de juego de los datos. Si bien no logran acreditar una solera comparable con SPSS (1969), SAP (1973) o SAS (1976), no deja de ser cierto que son unos auténticos veteranos con presencia en el mercado desde 1993, en ambos casos.

Ambas herramientas son mantenidas regularmente y han ido lanzando nuevas versiones por lo que las versiones estables han llegado hasta la 3.3.1 en el caso de R y a la 3.6.6 en el caso de Weka.

Un nuevo paralelismo puede encontrarse en la vinculación de ambas herramientas a instituciones universitarias de Nueva Zelanda: R nace en el Departamento de Estadística de la Universidad de Auckland (con antecedentes en los Laboratorios Bell de AT&T), mientras que Weka nace en la Universidad de Waikato.

Puede considerarse que las herramientas analizadas presentan un grado de madurez adecuado y son capaces de dar respuestas satisfactorias a sus usuarios.

4. CONCLUSIONES

El análisis predictivo ha dejado de estar reservado a grandes corporaciones, gobiernos o universidades y se ha generalizado como una herramienta más de la Busines Intelligence (Inteligencia de los negocios) a disposición de todo tipo de empresas y organizaciones.

El requerimiento fundamental para realizar análisis predictivo es la existencia de un conjunto lo suficientemente amplio de datos como para permitir detectar en ellos patrones que permitan formular reglas capaces de anticipar previsiones.

La capacidad de almacenar y gestionar conjuntos de datos masivos ha crecido de manera exponencial en los últimos años al tiempo que ha aparecido una cultura empresarial y gubernamental que apuesta por la recolección de datos de manera sistematizada, en la confianza de que en algún momento podrá extraerse de los mismos información relevante.

Las operadoras telefónicas almacenan el geoposicionamiento de los usuarios, los bancos almacenas millones de tiques abonados con tarjetas de crédito, Google permite almacenar en Gmail gigas y gigas de correos al tiempo que en las redes sociales se invita a los usuarios a compartir opiniones, fotos y vídeos: todos esos elementos constituyen en última instancia datos y del análisis de los datos emergen pautas de comportamiento susceptibles de ser utilizadas en la planificación del transporte o en el marketing personalizado entre un sinnúmero de posibles aplicaciones.

Esa posibilidad real de almacenar y procesar datos, unida a la cultura de conservarlos, requiere de un tercer elemento: las herramientas capaces de encontrar patrones que permitan formular reglas.

Desde la década de los sesenta existen en el mercado potentes herramientas capaces de realizar complejos análisis estadísticos (SPSS, SAP Business Suite o SAS Software Package). Esta oferta de herramientas especializadas se ha visto complementada por herramientas de software libre entre las que destacan las analizadas en este TFG (R y Weka).

El análisis de R y Weka se ha realizado creando con ambas herramientas modelos predictivos. En los dos casos se han realizado árboles de decisión y modelos de agrupamientos con los algoritmos propios de cada una de ellas. En ambos casos se ha finalizado creando un modelo de reglas de asociación utilizando el algoritmo apriori.

El resultado de la comparativa puede resumirse afirmando que ambas herramientas cumplen con las prestaciones exigibles y que la curva de aprendizaje de R, más dura y pronunciada, es compensada por su mayor potencia y flexibilidad.

5. BIBLIOGRAFÍA

- **Strickland, Jeffrey** (2014). *Predictive Analytincs using R*.
- **Mayor, Eric** (2015). *Learning Predictive Analytics with R*.
- **Siegel, Eric** (2013). *Predictive Analytics. The power to predict who will click, buy, lie, or die*.
- **Holtzman, S.** (1989). *Intelligent Decision Systems. Addison-Wesley*
- **Nyce, Charles** (2007). *Predictive Analytics White Paper*.
- **Molina Félix, Luis Carlos; Sangüesa i Solé, Ramón**. Reglas de asociación. *Data mining*.
- **Sangüesa i Solé, Ramón**. Clasificación: árboles de decisión. *Data mining*.
- **Sangüesa i Solé, Ramón**. Agregación (*cluestring*). *Data mining*.
- **Cía, Juan F**. El ranking de las mejores soluciones de análisis predictivo para empresas: https://bbvaopen4u.com/es/actualidad/el-ranking-de-las-mejores-soluciones-de-analisis-predictivo-para-empresas
- **Merino, Pedro Pablo** (2016). Los datos, el nuevo petróleo del siglo XXI: http://ecommerce-news.es/actualidad/los-datos-nuevo-petroleo-del-siglo-xxi-41824.html#
- **Barranco Frangoso, Ricardo** (2012). ¿Qué es Big Data? : https://www.ibm.com/developerworks/ssa/local/im/que-es-big-data/
- **Mathew, George** (2016). Five Ways Data Analytics Will Shape Business, Sports And Politics In 2016: http://www.forbes.com/sites/valleyvoices/2016/01/20/five-ways-data-analytics-will-shape-business-sports-and-politics-in-2016/#463ea2375a1d
- **Shimada, Thomas; López, Fabián**. Analítica Predictiva: cómo convertir la información en ventaja competitiva: http://www.il-latam.com/images/articulos/articulo-revista-109-como-convertir-la-informacion-en-ventaja-competitiva.pdf
- Plan de Transformación Digital de la Administración General del Estado y sus Organismos Públicos (Estrategia TIC 2015-2020), Gobierno de España – Ministerio de Hacienda y Administraciones Públicas.
- Análisis Predictivo para comprender el consumo de agua y mejorar la administración de recursos: http://www.besmart.company/blog/analisis-predictivo-para-comprender-el-consumo-de-agua-y-mejorar-la-administracion-de-recursos/