UNIVERSIDAD POLITÉCNICA DE VALENCIA
ESCUELA TÉCNICA SUPERIOR DE INFORMÁTICA APLICADA

MINERÍA DE DATOS CON WEKA PARA LA PREDICCIÓN DEL PRECIO DE AUTOMÓVILES DE SEGUNDA MANO

PROYECTO FIN DE CARRERA

Autor
Agustín José Calleja Gómez

Director
Cèsar Ferri Ramírez

Fecha del proyecto
Diciembre 2010

AGRADECIMIENTOS

Todo viaje llega a su fin, y ciertamente, este ha sido largo…

Quisiera aprovechar este documento para agradecer a todas aquellas personas que me han ayudado a lo largo de este viaje que ha sido la titulación en ingeniería informática técnica. Sin estas personas probablemente no lo habría conseguido.

A mis compañeros de clase, porque sin ellos las penas y las alegrías, no habrían sido las mismas. Porque me explicaban las cosas que no entendía y me han ayudado a entenderme un poco más a mí mismo. Porque han estado ahí cuando les he necesitado y sin pedir nada a cambio. Por la paciencia demostrada, sobre todo por algunos de ellos, para soportarme durante estos años.

A mis profesores, por ser siempre muy profesionales con su trabajo. Por contestar a correos absurdos sobre preguntas aún más absurdas siempre con tacto. Por sus horas de revisión de examen y la paciencia agotada. Por no ponernos nunca las cosas demasiado fáciles.

A mi director de proyecto, por ser mi guía en este tramo final. Por ayudarme en todo lo que ha podido. Por contestar siempre a mis correos con prontitud. Por inspirarme cuando no sabía por dónde ir.

A mi madre, sobre todo a mi madre, por haber hecho con su esfuerzo desde el día en que nací, el que hoy me haya convertido en quién soy. Por sudar sangre para darme todas aquellas oportunidades que ella no tuvo. Por sus consejos y sermones. Por ser siempre la primera en estar ahí cuando he necesitado hablar o cualquier otro tipo de ayuda. Por recordarme cada día lo mucho que me quiere. Por instigarme día sí día también a realizar mis obligaciones, entre ellas, este proyecto.

A mi novia, por servirme de apoyo en la recta final, justo cuando más lo he necesitado. Por instigarme también a realizar esta memoria. Por estar a mi lado durante las largas horas frente al ordenador trabajando en este proyecto. Por ser como es.

Por último, y no por ello menos importante, a mi gata, que falleció este año. Por sus horas de compañía a mi lado mientras estudiaba desde el colegio, hasta la universidad. Por soportar mis travesuras de crio. Por recibirme cada día al llegar a casa. Por todo el cariño que me ha dado.

A todos ell@s, gracias.

Listado de figuras

Tablas

Tabla de contenidos

1. INTRODUCCIÓN

1.1 Alcance del trabajo

La finalidad de este Proyecto Final de Carrera es la de profundizar en un concepto muy interesante relacionado con la obtención y extracción de información relevantes que podemos encontrar en una colección de datos. Estamos hablando de la Minería de Datos. Bajo este nombre se agrupan todas aquellas técnicas que nos ayudan a extraer conocimientos e información relevantes que se encuentran implícitos en las bases de datos.

La información en bruto, puede resultar mucho más provechosa y fácil de trabajar con ella si está ordenada, clasificada y dividida o agrupada en conceptos comunes. Estas dos tareas las aborda la minería de datos, nos provee de herramientas que clasifican y agrupan estos datos "en bruto" y de este modo poder sacar el máximo provecho de ellos.

No obstante, no es lo único que podemos conseguir aplicando diferentes métodos y técnicas de minería de datos. Mediante estos mecanismos de cálculo, asociación y segmentación, a través de la búsqueda de patrones comunes en los datos, situaciones que siempre se repiten o "reglas" implícitas en los propios datos, somos capaces de predecir diferentes situaciones o datos que vamos a recibir en un futuro.

El clásico ejemplo para esto, es el de la cesta de la compra. Mediante sencillos métodos de análisis de las compras hechas en cualquier supermercado, buscando patrones de comportamiento y, como hemos comentado, situaciones que se repiten en varias ocasiones, podemos predecir que, por ejemplo, cuando alguien compra hamburguesas, hay una alta probabilidad de que también compre pan de hamburguesa. A simple vista puede parecer una predicción un tanto evidente, pero no resulta tan evidente cuando se descubren patrones de comportamiento de la gente al comprar y, siempre mediante el análisis de los datos obtenidos, llegamos a la conclusión de que colocar los productos del supermercado en una distribución u otra puede llegar a ser muy relevante a la hora de registrar más compras de unos productos u otros.

En este proyecto me gustaría profundizar mucho más en todos estos conceptos y métodos de minería de datos, y que mejor manera de hacerlo que crear mi propio caso de estudio y poner en práctica todos aquellos conocimientos que vaya adquiriendo a medida que realizo y me sumerjo en estos temas mediante un ejemplo práctico de utilización de técnicas de minería de datos.

1.2 Estructura del proyecto y finalidad

El proyecto constará de varias partes bien marcadas. Primero nos centraremos en el estudio de la minería de datos, analizaremos y marcaremos en detalle los conceptos más importantes de esta metodología de tratamiento de datos. Hablaremos de sus orígenes, de su presente y del futuro que tiene esta "tecnología" a largo plazo.

La segunda parte del proyecto consistirá en crear un caso de estudio real, del cual podamos extraer datos, analizarlos y aplicar diferentes métodos y técnicas de minería de datos.

Para ello, lo imprescindible será crear una base de datos de la cual obtener y clasificar la información que en ella reside. Es por esto que se diseñará una aplicación en Microsoft Access para aprovechar el motor de bases de datos de la misma, así como su fácil e intuitivo lenguaje de programación para aplicaciones Visual Basic.

No obstante, necesitamos una fuente de la cual obtener la información y llenar nuestra base de datos, así como decidir con que información la vamos a nutrir. Optamos por utilizar una de las muchísimas páginas web de artículos de segunda mano para extraer toda esta información y crear nuestro entorno de trabajo. De este modo, los datos serían actuales siempre y corresponderían a la realidad más próxima, esto no puede ser siempre muy importante, pero dado que elegimos como caso de estudio la predicción del precio de vehículos de segunda mano, de nada sirven los datos si no son lo más actuales posible y corresponden con la actualidad socio-económica actual.

Para la realización de la aplicación, se analizará la estructura de la información de la fuente. Como se estructuran los datos en la propia web, que información podemos extraer, que datos son útiles y cuáles no lo son, por lo que tendremos que sumergirnos un poco en el código fuente de la propia página.

Una vez hayamos decidido la información que queremos conseguir, crearemos las tablas y diferentes formularios en la aplicación para la extracción de datos de la web. Utilizando el lenguaje de programación de Microsoft Visual Basic para Aplicaciones, descargaremos toda la información que seleccionemos y se introducirá en nuestras bases de datos para su posterior estudio y tratamiento.

El objetivo consiste en que ante una marca y un modelo de coche dados o elegidos por el usuario de la aplicación, seamos capaces de predecir qué precio tendría ese vehículo en el mercado. Es evidente, que cuanto más específico seamos, la predicción será mejor.

Después de haber obtenido todos los datos referentes a los vehículos de la propia página web, la aplicación será capaz de exportar estos datos a un fichero legible y tratable por la aplicación encargada de modelar y exprimir toda esta información en bruto y construir un modelo de predicción.

Se utilizará para este propósito una herramienta bajo licencia GPL denominada WEKA. Weka es una suite consistente en un conjunto de librerías diseñadas en JAVA por la universidad de Waikato (Nueva Zelanda) con diferentes métodos y técnicas de Minería de Datos.

Weka ocupará la tercera parte del proyecto. Dado que es una herramienta muy completa y muy potente, analizaremos su utilización más básica y en vistas a la comprensión y elaboración del modelado de nuestro caso de estudio. Sin embargo, tampoco dejaremos de lado todos aquellos conceptos y utilidades importantes de la aplicación aunque no se utilicen en el desarrollo y resolución del caso de estudio.

Con los datos exportados de la aplicación en Access, en Weka tendremos que tratarlos y analizarlos en profundidad para que nuestro modelado del caso de estudio sea lo más preciso posible. Cuando hablamos de modelo, nos referiremos a aquella técnica de minería de datos que nos ayude a, en este caso, predecir el precio del vehículo.

En Weka, deberemos decidir que datos son relevantes y cuales no, que atributos influyen realmente en el precio final del vehículo y cuales simplemente resultan un estorbo para el cálculo. Para cumplir con este propósito, después de haber filtrado bien los datos fuente, procederemos a aplicar diferentes métodos y técnicas que Weka nos proporciona.

En sus librerías, Weka tiene implementados los procesos más comunes y útiles de minería de datos. Desde un simple árbol de decisión, hasta métodos más complejos de asociación y agrupamiento, redes neuronales, aprendizaje Bayesiano o regresión.

Tras el análisis de los resultados obtenidos al aplicar varias técnicas, decidiremos cuál es el método más indicado y preciso para resolver nuestro caso de estudio. Al aplicar este método, Weka nos construirá un modelo para la resolución del problema propuesto. Este modelo puede ser un árbol de decisión en el que descendiendo por sus ramas podemos llegar a predecir el precio del coche, o incluso una fórmula matemática que podamos aplicar a unos datos dados por el usuario y nos dé como resultado el precio estimado del vehículo.

La cuarta parte del proyecto, para finalizar, consistirá en implementar este modelo creado en Weka. Aprovechando el aplicativo creado en Access, agregaremos después la funcionalidad de calcular la predicción basándose en los datos obtenidos y el modelo otorgado por Weka.

Así, los propósitos y funcionalidades de la aplicación diseñada serán los siguientes:

- Crear una interfaz de usuario fácil e intuitiva de utilizar
- Facilitar el almacenamiento de una gran cantidad de datos
- Obtener y extraer datos de una página web de artículos de segunda mano
- Realizar un tratado previo de dichos datos
- Exportar estos datos a un formato ideal para ser tratado por Weka
- Implementación del modelo resolutivo del caso de estudio obtenido por Weka

A modo de resumen final, el propósito de este proyecto es el de profundizar y aumentar mis conocimientos en Minería de Datos. Centrando mi experiencia en el mundo de las Tecnologías de la Información, creo que puede resultar de gran utilidad para mi vida laboral adquirir esta clase de conocimientos para el mundo empresarial. Un mundo donde ser capaz de anteponerse a las necesidades de los clientes, o ser capaces de predecir si se van a necesitar más o menos artículos en determinadas épocas del año, por citar algunos ejemplos, puede resultar imprescindible para el progreso y éxito de la empresa. Así como de afianzar mis conocimientos en programación introduciéndome en un lenguaje de programación nuevo y no desarrollado durante la titulación y aplicar los conocimientos adquiridos en asignaturas como Metodología y Tecnología de la Programación, Ingeniería del Software de Sistemas, Programación, Estructura de Datos y Algoritmos, etc.

2. MINERÍA DE DATOS

2.1 Introducción

"Ignorance is the curse of God, knowledge the wing wherewith we fly to heaven"

– William Shakespeare

Siempre se ha dicho y ha sido así desde que el mundo es mundo, que la información es poder. El ser humano siempre ha intentado conocer e investigar a fondo todo aquello que le rodeaba para sacar el máximo partido a sus posibilidades de progreso y éxito, y para ello, disponer de información exclusiva y relevante, siempre ha sido de gran ayuda.

Desde los primeros matemáticos con sus estadísticas y tablas de probabilidad, anticiparse a los hechos que podían acontecerse era clave para el ser humano. De este modo, podían modelar el mundo que les rodeaba, intentar ajustarlo a una serie de patrones que a menudo se repetían con el tiempo, y de este modo, sacar provecho a este "conocimiento".

Hoy en día, vivimos en un mundo saturado de información. Contamos con herramientas tecnológicas que ponen al alcance de nuestra mano vastas e ingentes cantidades de información y datos. La expansión de internet y de los sistemas de información ha revolucionado considerablemente nuestra capacidad de obtener información de una manera fácil y rápida.

No obstante, "con el grado de crecimiento sin precedentes con el que la información es recolectada y almacenada electrónicamente hoy en día en prácticamente todos los campos del comportamiento/desarrollo humanos, la extracción de información útil de todos los datos disponibles se está convirtiendo en un creciente reto científico y una necesidad económica masiva." [Zaki and Ho 2000]. Se estima que la cantidad de información del mundo se dobla cada 20 meses [AI Magazine].

Aquí es donde el desarrollo tecnológico a nivel computacional entra en juego, mejores computadores con los que desarrollar análisis exhaustivos de los datos en busca de información relevante, de relaciones entre los datos, etc. Gracias a este desarrollo y a la creciente necesidad de filtrar y organizar estas cantidades de datos, nació un concepto denominado: *KDD* por sus siglas en inglés: *Knowledge Discovery in Databases.*

En este capítulo haremos una introducción a los conceptos más importantes de uno de los pasos que forman el proceso KDD, la minería de datos. Se desarrollará también el resto de pasos del proceso, no obstante será un análisis muy superficial pues se escapa de los objetivos de este proyecto.

2.1.1 Nacimiento de la minería de datos

Antes de profundizar en materia, analizaremos los hechos que han hecho posible el desarrollo y evolución de la minería de datos y sus técnicas y métodos.

Podemos hablar de tres grandes grupos de actuación, donde la evolución y el desarrollo en estos campos han propiciado el crecimiento y la necesidad de la minería de datos.

Por una parte tenemos un altísimo crecimiento en la recolección de datos. Prácticamente cada acción que realizamos hoy en día en la que interactuamos con un servicio, una empresa, o incluso una persona, queda registrado informáticamente y puede constar como dato almacenado. A continuación, expondremos algunos de los campos donde esta recolección de datos masiva ha crecido más.

- Internet. Fuente inagotable de contenidos de información.
- La banca, especialmente las transacciones con las tarjetas de crédito
- Compra-venta, las transacciones de compras y ventas del mercado.
- Ciencias, biología y química
- Informes gubernamentales
- Sistemas de Gestión de entornos empresariales

"La potencia sin control, no sirve de nada". No habríamos sido capaces de almacenar todas estas ingentes cantidades de información sin unos sistemas informáticos más avanzados y potentes, así como de nuevas tecnologías como el procesado en paralelo o los tremendos avances en almacenamiento de información, con menor coste y menor espacio.

Por último, merece especial mención las matemáticas, madre de la informática, con sus métodos y técnicas en tiempo real y sus aplicaciones a las nuevas tecnologías.

2.1.2 Conceptos básicos

Han existido muchos nombres para la Minería de Datos o disciplinas similares. Entre ellos se encuentra el *"Data fishing", "data discovery",* y, más recientemente, *Knowledge Discovery in Databases (KDD).* A pesar de que para muchos, *KDD* y la minería de datos son sinónimos, *KDD* es un proceso que incluye a la minería de datos como uno de sus pasos. (Piatetsky-Shapiro, G. AI Magazine).

¿Qué es la minería de datos? Podemos encontrar decenas de definiciones a este concepto. La minería de datos consiste en "la aplicación de técnicas en grandes volúmenes de datos para descubrir información útil, aplicable y no trivial". Esta definición, aplicada a un entorno más empresarial podría reconstruirse como "el conjunto de métodos, que junto con un profundo conocimiento del negocio, están orientados a identificar, en grandes volúmenes de datos, relaciones y tendencias ocultas hasta el momento" (Carlos Creus, 2006).

Podemos decir, que la minería de datos es un proceso dentro de un proceso que lo engloba todo, el KDD. En este paso, la minería de datos se encarga de buscar relaciones y patrones entre toda la cantidad de información disponible.

Un patrón, es algo que se repite, una tendencia, como una representación de los datos e información obtenidos de una fuente de información, como puede ser una base de datos. Un patrón, ha de cumplir una serie de características para que nos resulte de utilidad a la hora de trabajar con él y obtener información de utilidad.

Características:

- Ha de ser interesante para la cuestión que estemos analizando, ha de cumplir con nuestras expectativas de búsqueda de información. De nada nos valdría saber que cuando llueve nos mojamos, si lo que buscamos es saber cuándo va a llover.
- Ha de ser aplicable, es decir, debe poder adaptarse a una gran cantidad de los datos de los que disponemos, para poder ser relevante, cuantos más datos cumplan dicho patrón mejor.
- No ha de ser trivial, debe aportarnos alguna clase de conocimiento útil para lo que estamos analizando.
- Ha de ser nuevo y desconocido antes de aplicar los métodos para descubrirlo.
- Debería ser comprensible, patrones retorcidos que relacionan los datos unos con otros a base de "interrelaciones" complejas y "rebuscadas" no nos son de utilidad.

Para obtener estos patrones y poder conseguir información relevante y de utilidad, la minería de datos dispone de varios métodos y algoritmos, que aplicados a grandes cantidades de datos son capaces de descubrir estos nuevos patrones y tendencias ocultas.

Estos métodos se pueden clasificar en dos grandes grupos, según la información que obtenemos al aplicarlos convenientemente. Así, podemos dividirlos en métodos predictivos y métodos descriptivos.

Los métodos predictivos, comprenden el uso de algunas variables o campos de la base de datos para predecir valores futuros o desconocidos, o incluso otras variables de interés.

Los métodos descriptivos, se centran en encontrar patrones comprensibles para el ser humano que describan la información que tenemos.

Aunque los límites entre unos métodos y otros no están claramente definidos, pues algunos métodos predictivos pueden ser descriptivos y viceversa, la distinción es útil para entender el objetivo general del proceso de descubrimiento.

Existen muchos métodos, pasaremos a realizar una breve introducción y clasificación de los métodos de los que hablaremos más en profundidad a lo largo del documento.

Métodos predictivos:

Entre los métodos predictivos más conocidos nos encontramos con los árboles de decisión y los métodos basados en la regresión matemática. Los árboles de decisión pueden utilizarse para conocer si, por ejemplo, un día podremos salir a jugar a tenis utilizando un historial de datos meteorológicos de los días que hemos podido salir a jugar y aquellos en los que el tiempo no lo ha permitido como base.

Los métodos regresivos pueden utilizarse para predecir compras de clientes por grupos de edad, dado un historial de compras por edad para un rango de edades, o incluso, el precio de un vehículo de segunda mano si tenemos como base una relación de datos sobre coches de segunda mano de similares características con sus correspondientes precios, características y atributos.

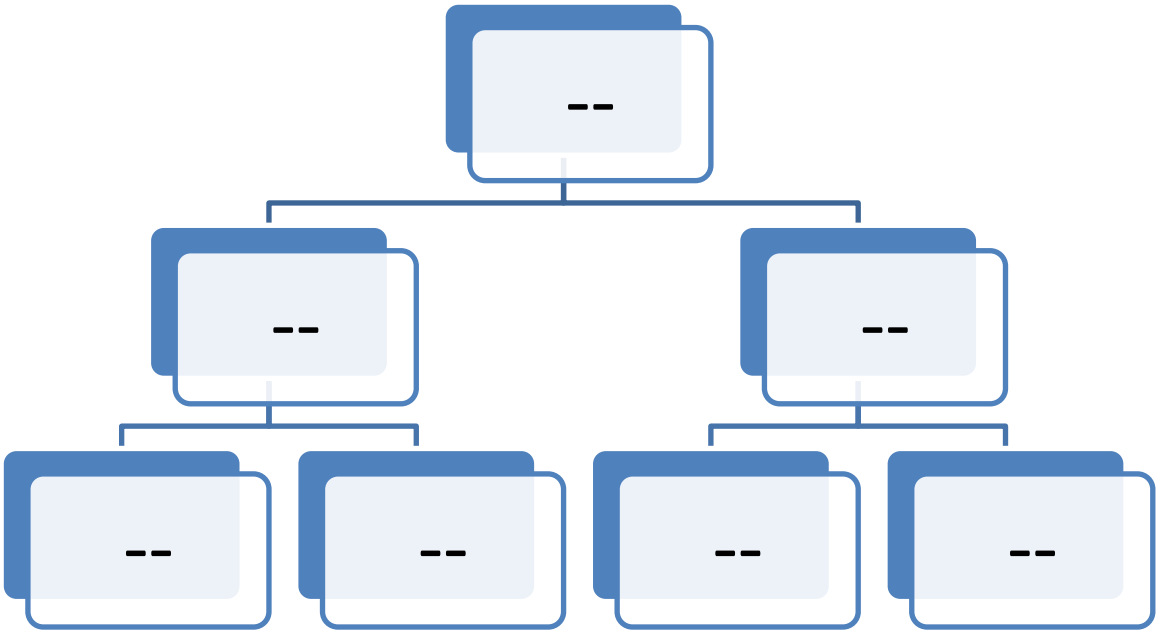

Figura 1. Árbol de decisión

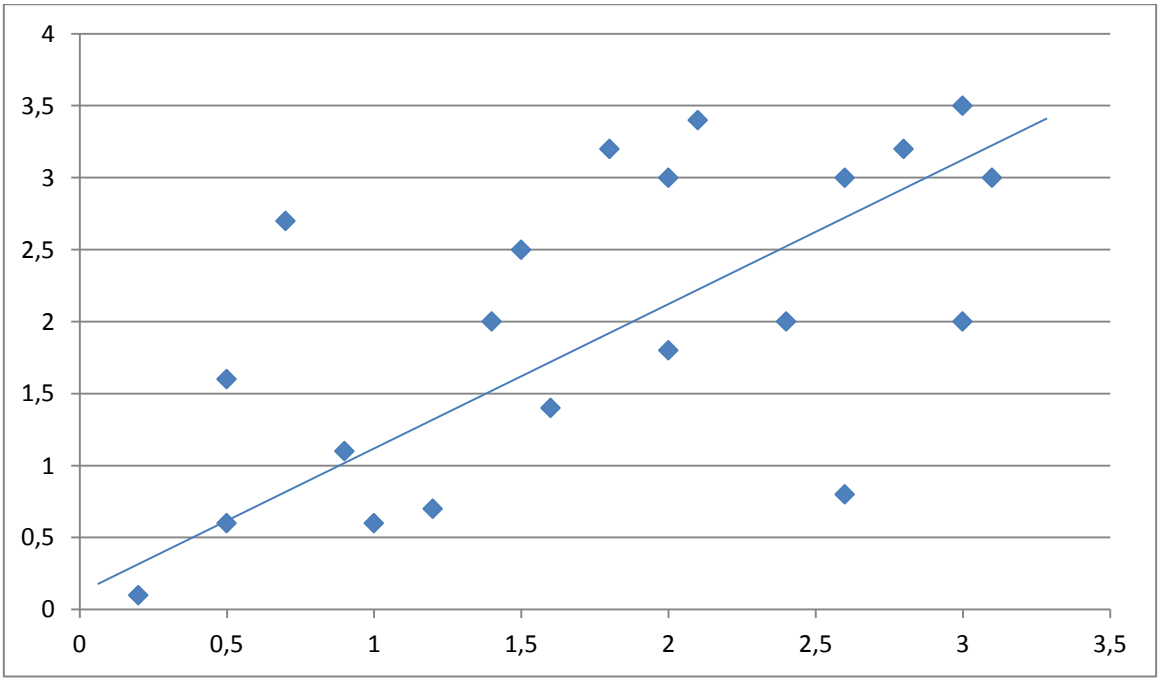

Figura 2. Regresión

Métodos descriptivos:

Los métodos descriptivos no precisan registros de datos o sucesos, no dependen de los patrones obtenidos para detectar reglas, correlaciones y asociaciones. Podemos obtener información prácticamente al momento de la información que tenemos.

El *Clustering* o agrupamiento, es un método mediante el cual descubrimos grupos y estructuras en los datos y que en cierta medida son parecidos o cumplen características similares sin utilizar estructuras conocidas en los datos.

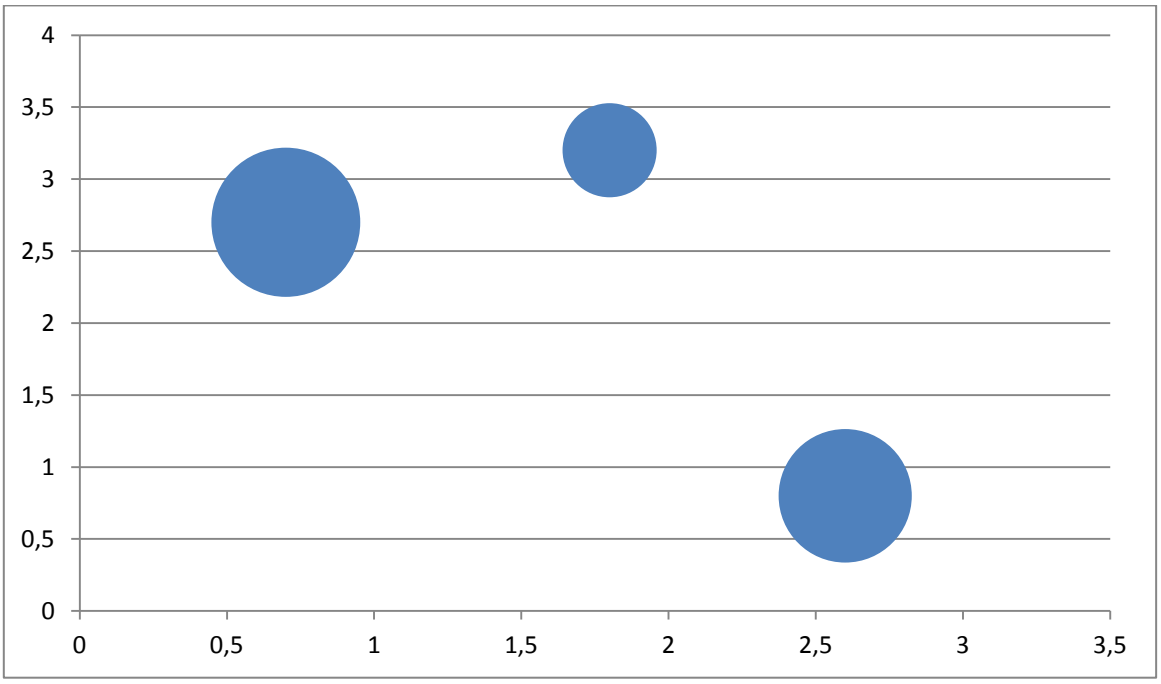

Figura 3. Agrupamiento

La clasificación ABC, nos ayuda a clasificar los ítems en diferentes grupos basándose en valores y criterios cuantitativos. Por ejemplo, para clasificar a los comerciales de la empresa según el número de ventas realizadas, o del importe total de sus ventas.

Análisis asociativo, o comúnmente conocido como "análisis de la cesta de la compra" tiene como objetivo encontrar patrones, particularmente en procesos de negocio, y formular reglas aplicables, como por ejemplo, si un cliente compra hamburguesas, dicho cliente compra también pan de hamburguesa.

El análisis aproximativo incluye tres técnicas diferentes. Encontramos:

- Tablas ponderadas
- Regresión lineal
- Regresión no-lineal

Aunque las técnicas trabajan de forma diferente, el objetivo final de todas ellas es el de aproximar un valor para un atributo específico.

2.1.3 La minería de datos como parte de un proceso mayor

El proceso de minería de datos depende estrechamente del método o técnica que vayamos a utilizar para resolver el problema o el requerimiento de información que se nos ha presentado. Los métodos predictivos suelen requerir "entrenamiento" para, de este modo, ser capaces de modelar las reglas que se deben aplicar a los datos nuevos para la predicción, así como de algún que otro paso de verificación para comprobar la precisión del modelo obtenido. Sin embargo, existen otros métodos que únicamente necesitan ser ejecutados sobre una colección de datos para obtener resultados.

Como ya he mencionado, considero la minería de datos como parte de una tarea mayor de procesamiento de negocio y datos llamado *Knowledge Discovery in Databases (KDD)*. Muchos expertos en la materia coinciden en que la forma de conseguir "conocimientos" de la información en bruto sólo se puede conseguir mediante técnicas modeladas mediante procesos. Colocando los métodos de minería de datos estratégicamente en el centro. No obstante, para que los procesos y métodos de minería nos brinden resultados concluyentes y útiles, los pasos preliminares de preparación de la información y los post-procesos que verifican la información obtenida, son imprescindibles. Estas tareas adicionales conforman el proceso KDD.

2.2 *KNOWLEDGE DISCOVERY IN DATABASES (KDD)*

2.2.1 El proceso KDD

KDD es únicamente el concepto de un proceso de múltiples pasos que identifica patrones en los datos para encontrar nueva información. La minería de datos es únicamente uno de esos pasos del proceso encargado de aplicar técnicas computacionales para encontrar dichos patrones en los datos. Este paso consistente en la utilización de algoritmos que proporcionan patrones en un tiempo aceptable de respuesta, obtenidos siempre, de colecciones de datos como puedan ser las bases de datos. Otros pasos en el proceso KDD son la comprensibilidad y la validación de los patrones descubiertos. KDD es el concepto y la minería de datos es su herramienta. [Witnessminer].

El proceso de descubrimiento de conocimiento en bases de datos es iteractivo, pues consta de varios pasos que pueden llegar a tener que repetirse para extraer la información óptima, e interactivo, pues incluye varios pasos donde la intervención de un usuario experto es imprescindible. En 1996, Brachman y Anand, propusieron una visión práctica del proceso, enfatizando la naturaleza interactiva del mismo. A continuación esbozaremos los pasos básicos del proceso:

1. Desarrollo y comprensión del contexto de trabajo. Identificar el objetivo del proceso KDD desde el punto de vista de la información requerida.
2. Agrupar un conjunto de datos para servir de objetivo del proceso. Seleccionar un grupo de variables, un subconjunto de datos, etc.
3. Limpieza de datos y preprocesado. Eliminar datos inútiles, decidir estrategias para manejar los campos con campos vacios, recolectar la información necesaria.

4. Reducción de los datos y proyección. De esta forma podemos obtener una forma más adecuada de representar nuestra colección de datos. Mejorar la eficiencia de los datos eliminado o combinando variables, o dejando aparte datos invariables.
5. Decidir el método de minería de datos adecuado para los datos que queremos obtener mediante el proceso KDD.
6. Análisis. Aquí se deciden que modelos y parámetros pueden ser adecuados y se decide que método exacto concuerda con el objetivo general del proceso.
7. Minería de Datos. Búsqueda de patrones de interés.
8. Interpretación de los patrones obtenidos, posiblemente volviendo a cualquiera de los pasos anteriores (iteración).
9. Utilización y puesta en práctica de los conocimientos obtenidos. Verificación de los datos obtenidos y otras comprobaciones técnicas.

El proceso puede implicar una iteración significante, es decir, podemos encontrar varios bucles entre cualquiera de los pasos o estados de los que se compone el proceso. En la figura a continuación podemos observar un esquema donde se detallan los pasos básicos de este proceso. La mayor parte de las investigaciones y documentos publicados se centran en el paso 7, la minería de datos, no obstante todos los pasos del proceso son igualmente importantes para la obtención de información y datos útiles y de calidad.

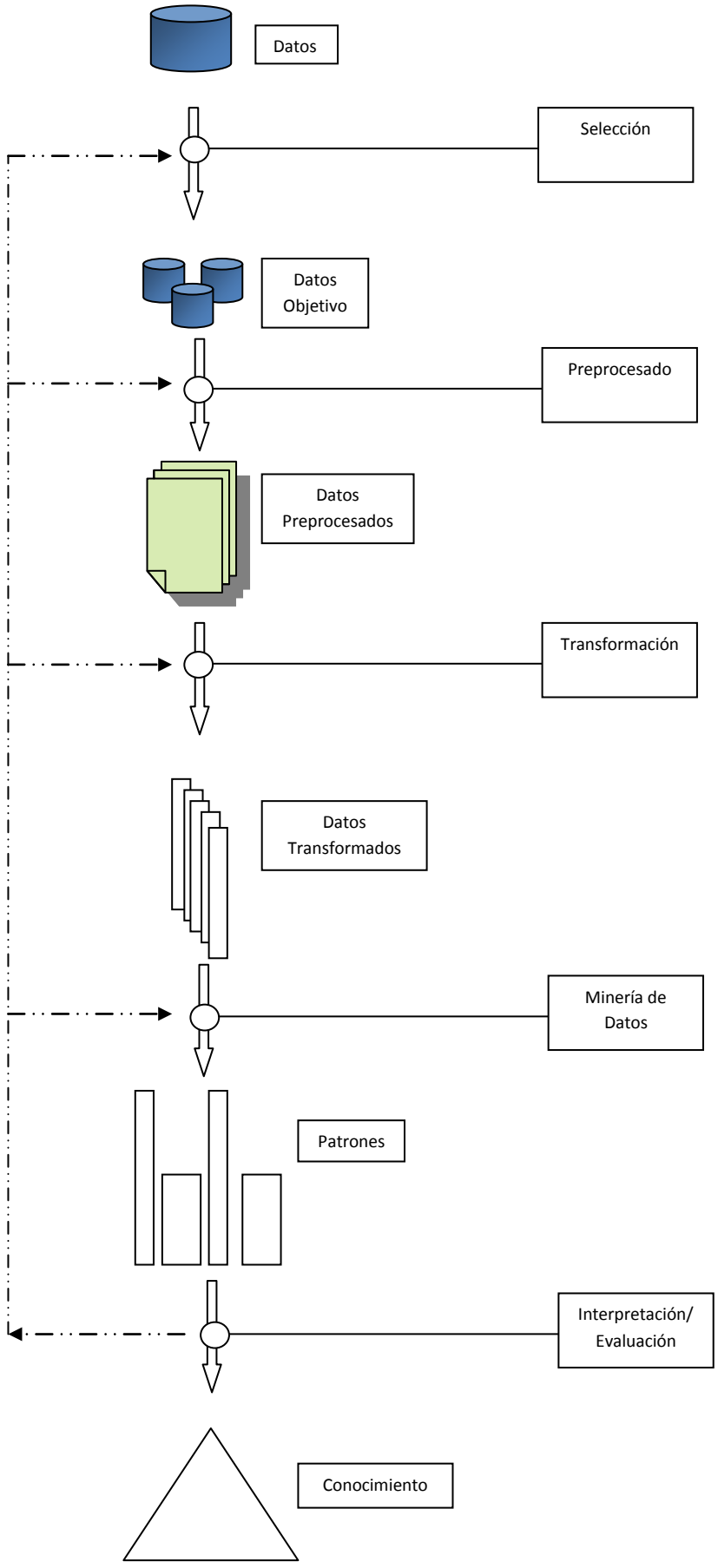

Figura 4. Proceso KDD [Lesley 2004]

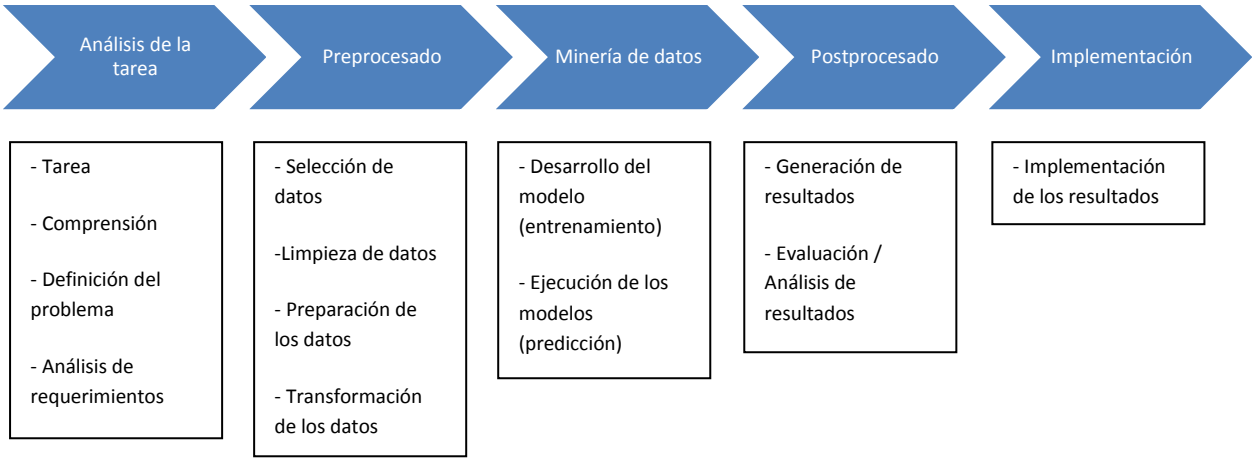

Figura 5. KDD [Lesley 2004]

2.2.1 El proceso de Minería de Datos

Ahora entraremos un poco en detalle en uno de los pasos más importantes del proceso de descubrimiento de conocimiento en bases de datos. Estamos hablando del proceso de minería de datos. Este paso suele ser bastante iterativo, ya que a menudo es necesario aplicar en repetidas ocasiones el método de minería en particular que hayamos seleccionado para trabajar.

Como hemos dicho anteriormente, cuando introducíamos los diferentes tipos de métodos de minería de datos, el proceso KDD consta de dos tipos de objetivos claramente diferenciables. Nuestro objetivo puede ser por un lado el de verificar la información, es decir, verificar que una hipótesis que el usuario ha formulado sobre un conjunto de datos es correcto. El otro tipo de objetivo es el de descubrimiento, donde el sistema busca y encuentra nuevos patrones por sí mismo. No obstante, podemos a la vez hacer una subdivisión dentro de los objetivos de descubrimiento, pues bien pueden tratarse de objetivos con énfasis en la predicción, donde lo que tratamos es de predecir el valor futuro que puede llegar a tener un dato en un momento determinado, o de descripción, donde el sistema busca patrones entre los datos para presentarle la información al usuario de una forma más comprensible y útil.

En este paso del proceso KDD, debemos ajustar los modelos o determinar los patrones adecuados para la colección de datos que estamos analizando. Para conocer si un modelo aplicado u otro es adecuado, debemos recurrir al proceso KDD al completo, pues para ello requiere una interactividad con el usuario que ha de decidir si la información obtenida al aplicar el modelo es lo que se estaba buscando.

Muchos de los métodos con los que cuenta la minería de datos se basan en diferentes técnicas de varios campos, como el aprendizaje de máquinas, reconocimiento de patrones y estadística, en este último encontramos los algoritmos de clasificación y regresión entre otros.

Los métodos y técnicas de minería de datos, la mayoría al menos, los podemos considerar como complementos o híbridos de unos pasos y principios básicos. Así podemos dividir los métodos en tres algoritmos primarios básicos.

- Representación del modelo
- Evaluación del modelo
- Búsqueda

Un esquema más completo del proceso lo podemos encontrar en el propuesto por Fayyad. Este esquema consta de 5 pasos para obtener el conocimiento que queremos extraer de los datos que tenemos.

1. Selección de los datos. Extraemos de una base de datos o cualquier otra colección de datos, aquellos campos y atributos que concuerdan con el objetivo que buscamos
2. Pre-Procesado. En este paso, efectuaremos la limpieza de los datos, como por ejemplo, rellenar campos vacios o atributos inútiles.
3. Transformación. En este paso, la información se convertirá a otros nuevos formatos si es necesario.
4. Minería. El núcleo del proceso, aquí se identifican los patrones y las relaciones entre los datos.
5. Interpretación y evaluación. El usuario debe interactuar con los resultados para tomar las acciones pertinentes en caso de ser válidos y útiles.

En el capítulo siguiente, pasaremos a describir más en profundidad varios métodos y técnicas de minería de datos. Con ello, finalizaremos la introducción a la minería de datos y entraremos en materia de la aplicación realizada y la implementación de uno de estos métodos detallados para un caso de estudio planteado en particular.

3. MÉTODOS DE MINERÍA DE DATOS

3.1 Introducción

Como se ha comentado anteriormente, los usos más comunes de la minería de datos y en sí mismo, el proceso KDD, son la predicción y la descripción. En este capítulo describiremos algunos métodos de minería de datos, diferenciándolos por sus usos más comunes (para la predicción o para la descripción), así como una breve introducción a sus conceptos más importantes y su ejecución. No entraremos no obstante en demasiado detalle, pues se escapa del alcance de este proyecto, pues existen numerosos métodos de minería de datos, algunos de ellos bastante complejos, pero increíblemente eficaces y apurados a la hora, de describir o de predecir información sobre una colección de datos dada.

En la siguiente tabla podemos ver una clasificación de algunos métodos de minería de datos según su propósito. Como podemos observar, algunos de ellos aparecen en varios lugares, esto es, pueden utilizarse con diferentes finalidades.

Tareas	Métodos
Predicción y descripción	Á de árboles de decisión, análisis cesta de la compra, análisis de series temporales, redes neuronales, tecnología de agente de red
Clasificación	Análisis cesta compra, árboles de decisión, redes neuronales, ordenamiento
Regresión	Regresión lineal, regresión logística, regresión multinominal
***Clustering* (Agrupamiento)**	Análisis de grupos, redes neuronales
Summarization	Algorítmos genéticos
Modelado de dependencias	Análisis de la varianza, análisis de enlace
Cambio y detección de desviación	Lógica difusa

Tabla 1. Clasificación métodos minería de datos

3.2 Árboles de decisión

3.2.1 Definición

Un árbol de decisión se utiliza como clasificador para determinar una acción o decisión apropiada (de entre un conjunto predeterminado de acciones) para una situación determinada. Un árbol de decisión nos ayuda a identificar correctamente los factores que se deben considerar y como cada uno de estos factores se ha asociado históricamente a los resultados de la decisión. [SAPDOCS]. La visión esquemática de este método lo hace uno de los métodos más sencillos de interpretar y asimilar la información que contienen. Se denomina árbol de decisión debido a que el resultado del modelo está representado en forma de árbol.

Los árboles de decisión son un método de los clasificados como métodos de aprendizaje supervisados, pues deben ser entrenados con información que contiene un histórico de los propios datos y los resultados que han sido consecuencia de dichos datos para poder utilizarse con el fin de crear predicciones.

Para verificar estas predicciones obtenidas como resultado y comprobar la precisión, podemos ejecutar el modelo entrenado contra otra colección de datos conocida para evaluar dicha precisión del modelo entrenado.

Los pasos serían pues:

1. Entrenamiento. Se modeliza el árbol para representar los patrones detectados en el historial de los datos lo mejor posible.
2. Evaluación. En este paso, totalmente opcional no obstante, podemos probar la validez del modelo entrenado enfrentándolo a otra colección de datos diferente (misma temática y mismo contenido, pero diferentes en sí). Si la precisión alcanzada no es la deseada, deberemos rediseñar el modelo y repetir el proceso.
3. Predicción. Por último, obtenemos el resultado predicho a partir del modelo diseñado, esto es, el valor o valores, o la decisión que buscamos tomar, para un determinado caso dado para nuestro conjunto de datos.

Con esto podemos generar la representación gráfica del árbol. El árbol se construye con los siguientes componentes:

- Nodo raíz: Como nodo único, forma el punto de entrada del árbol. Normalmente el punto más alto.
- Nodos de decisión: Éstos actúan como enrutadores para decidir que rama debemos tomar mientras recorremos el árbol de arriba abajo.
- Nodos hoja: Estos nodos son los que no contienen ningún nodo con “éxito”, es decir, nodos donde se cumple el objetivo, o donde se hace positivo el valor que intentamos predecir.

3.2.2 Ejemplo

En la figura a continuación, podemos observar un sencillo árbol de decisión. En este árbol se pretende predecir si un individuo comprará o no comprará un determinado producto en base a la edad, el salario y la ocupación del mismo. Fuente SAPDOCS.

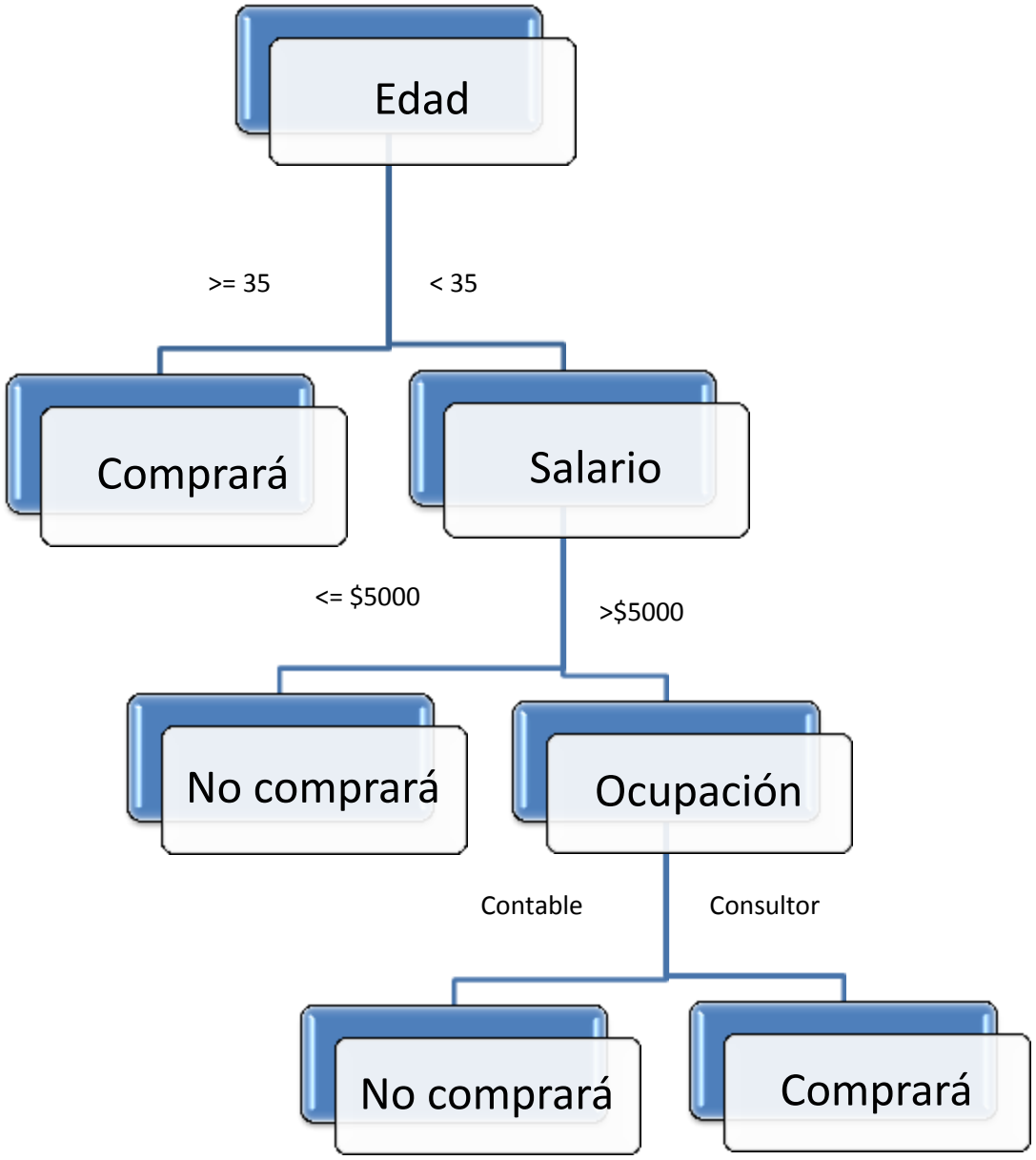

Figura 6. Árbol de decisión

3.3 Agrupación o *Clustering*

3.3.1 Definición

El *Clustering* se utiliza para agrupar los datos en conjuntos bien cohesionados y definidos. Podemos diferenciarlo de los métodos de clasificación normales en el hecho siguiente: las clases en las que se agrupan los datos no están predefinidas como en las clasificaciones normales, si no que se determinan a partir de los datos. Se trata de un método de aprendizaje sin supervisión.

Los resultados que podemos obtener al aplicar este método pueden utilizarse para resumir y analizar los contenidos de una colección de datos dada considerando las características de cada conjunto más que las características de cada registro. Este método puede utilizarse de manera descriptiva como predictiva (a qué grupo pertenecerá un nuevo dato).

3.3.2 Ejemplo

En el siguiente gráfico esquemático, podemos observar como en la gráfica obtenida al representar una serie de datos, según las características por las que estemos representando dichos datos, se puede ver cómo, si elegimos los atributos adecuados, se forman estos agrupamientos de los datos al coincidir características similares unos con otros

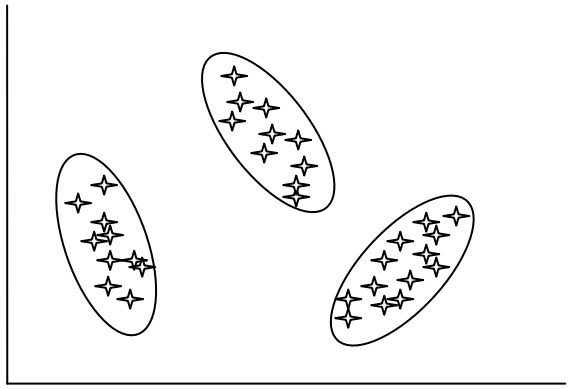

Figura 7. Clustering

3.4 Métodos estadísticos

3.4.1 Introducción

El objetivo de la modelización estadística consiste en explicar el comportamiento de una variable a partir del conocimiento de otras. Subyacente al concepto de modelización está la idea de que una variable tiene una cierta variabilidad y que esta variabilidad está relacionada con el comportamiento de otras variables. [Introducción a la minería de datos, 2004].

Los métodos estadísticos son de los más utilizados en la minería de datos, pues una gran mayoría de los problemas en minería de datos son de predicción de valores que desconocemos en base a valores históricos de los datos.

Podemos utilizar estos métodos tanto si el problema planteado consiste en la predicción de una cierta variable de respuesta, como si se trata de encontrar un modelo casual, en cuyo caso las variables explicativas son causa de la variación de la respuesta, permitiendo por tanto su intervención modificando las variables explicativas (si es posible tal modificación). Basta con que las variables explicativas estén asociadas a la variable de respuesta, para que sabiendo el valor que toman aquellas podamos hacer predicciones sobre el valor que tomará la variable de respuesta. [Introducción a la minería de datos, 2004].

En esta parte del trabajo hablaremos sobre dos de estos métodos, en primer lugar hablaremos de las tablas ponderadas, donde podemos combinar varias dimensiones que caractericen nuestros datos en una sola medida, por ejemplo el "índice de valor de un cliente", este índice numérico nos permitiría juzgar el valor de los clientes de una empresa a primera vista sin tener que analizar los valores y atributos de todos los clientes. Después, entraremos un poco en profundidad a las técnicas de regresión, técnicas que podemos utilizar para predecir valores clave continuos en conjunción con otros valores clave o características, como por ejemplo, los pedidos que se harán este mes teniendo en cuenta los pedidos que se hicieron este mismo mes el año pasado, índice de pérdidas, etc.

3.4.2 Tablas ponderadas

El método de las tablas ponderadas es una técnica de evaluación de alternativas cuando la importancia de cada criterio es diferente. En una tabla ponderada, a cada alternativa se le da una puntuación para cada criterio. Estas puntuaciones se ponderan después en base a la importancia de cada criterio. Todas las puntuaciones ponderadas de una alternativa se suman después para calcular el valor ponderado total de la alternativa en general. La alternativa con la puntuación más alta será probablemente la mejor alternativa para utilizar como base del método de tablas ponderadas para realizar predicciones. [SAPDOCS 2005].

3.4.3 Regresión

El análisis regresivo es una técnica utilizada para inter y extrapolar las observaciones, las cuales pueden clasificarse como regresión lineal o no lineal. Hablamos de modelo de regresión cuando la variable de respuesta y las variables explicativas son todas ellas cuantitativas. Si sólo disponemos de una variable explicativa hablamos de regresión simple, mientras que si disponemos de varias variables explicativas se trata de un problema de regresión múltiple. [Introducción a la minería de datos, 2004].

Para visualizar la relación entre la variable de respuesta y una variable explicativa, obtendremos el diagrama bivariante entre ambas variables. La forma de dicho diagrama aporta información sobre el tipo de relación entre la variable de respuesta y la variable explicativa. [Introducción a la minería de datos, 2004]

Regresión lineal

La regresión lineal es una técnica estadística que intenta construir un modelo para los datos analizados, y a través de éste predecir los datos futuros. Este modelo cuantifica la relación entre dos variables continuas: "la variable dependiente o la variable que intentamos predecir y la variable independiente o la variable predecible". [Rud 2001]. Funciona encontrando una línea a través de los datos que minimiza el valor del error cuadrático de cada punto. La formula de regresión lineal es la siguiente: [Whitehead 2005]

$Y = a + bX + c$
Y: variable dependiente auxiliar, = 1 si el evento sucede, =0 si no sucede
a: el coeficiente del término constante
b: el coeficiente/s en la variable/s dependiente/s
X: la/s variable/s dependiente/s
c: el término de error

Regresión no lineal

La relación entre dos variables puede no ser lineal, para resolver este tipo de problemas surgen las diferentes técnicas que existen de regresión no lineal. La relación puede ser curvilínea o de múltiples líneas. Entre las curvilíneas se encuentra la regresión logarítmica, "este modelo es simplemente una transformación no lineal de la regresión lineal" [Whitehead]. La diferencia fundamental entre la regresión lineal y la logarítmica reside en el hecho de que en la regresión lineal, la variable dependiente es continua, sin embargo, en la logarítmica es discreta o categorica.

La fórmula que describe esta función puede formularse como sigue: [Whitehead]

Ln[p/(1-p)] = a + bX + c
p: probabilidad de que el evento Y ocurra, p(Y=1)
b: el coeficiente/s de la variable/s independiente/s
c: el término de error
p/(1-p): rango de probabilidades
ln[p/(1-p)]: rango de probabilidades logarítmicas

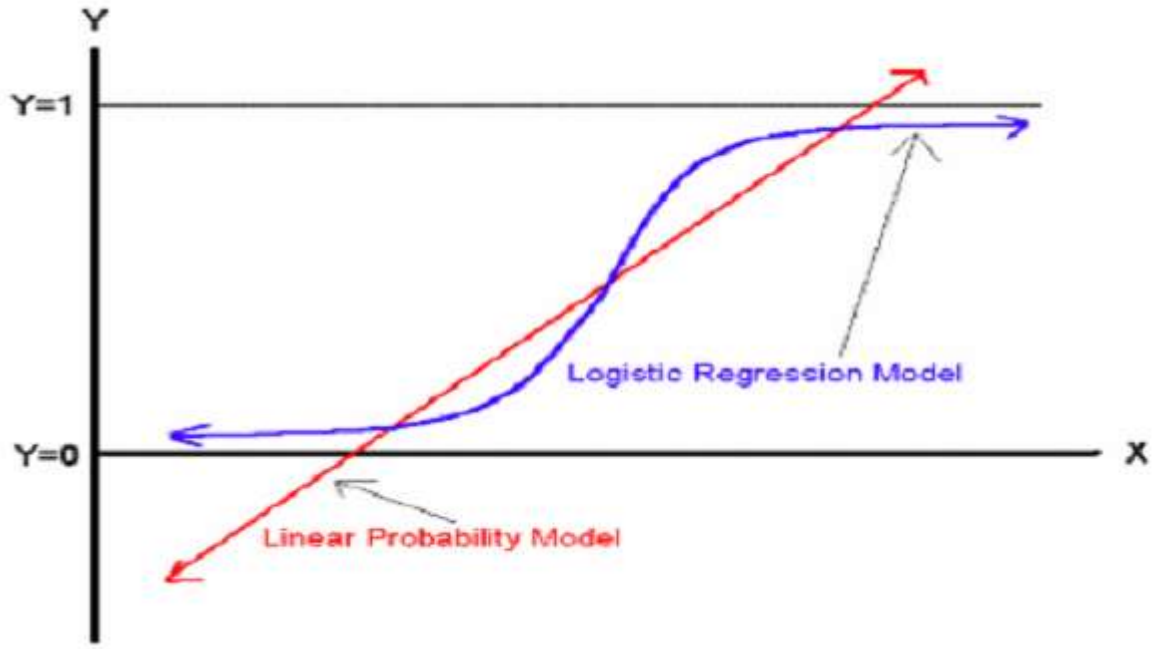

Figura 8: Comparación entre regresión lineal y logarítmica [Whitehead, Introduction to Logistic Regression]

Por otro lado tenemos la regresión logística multinomial, donde la variable dependiente de tipo nominal consta de más de dos categorías (politómica). Este tipo de regresión es una extensión multivariante de la regresión logística binaria clásica. [Hosmer & Lemeshow, 1989]

3.5 Análisis ABC

3.5.1 Definición

Este método se utiliza para clasificar objetos (clientes, productos o empleados) basándose en una medida en particular (ingresos, ganancias o coste. El resultado de la clasificación es una serie de rangos que determinan la importancia relativa de los objetos clasificados representados por una letra, A, B, C, etc. [Dr. Joseph Juran].

De este modo, asignamos un código para identificar los objetos más críticos o importantes de nuestro sistema (códigos A) y los triviales y menos importantes (códigos C).

3.5.2 Ejemplo

En el gráfico siguiente podemos ver una ilustración de una clasificación realizada con este método. En la gráfica, se muestran los ingresos obtenidos por según que clientes. Se puede observar como existen 3 grupos bien diferenciados, el grupo A, correspondiente a un 20% de los clientes que nos generan un 40% de ingresos. Un grupo B, con un 30% de clientes que nos generan otro 40% de ingresos y una última clase C, donde se encuentra el 50% de clientes restantes y que sus ingresos corresponden al 20% del total que hemos obtenido.

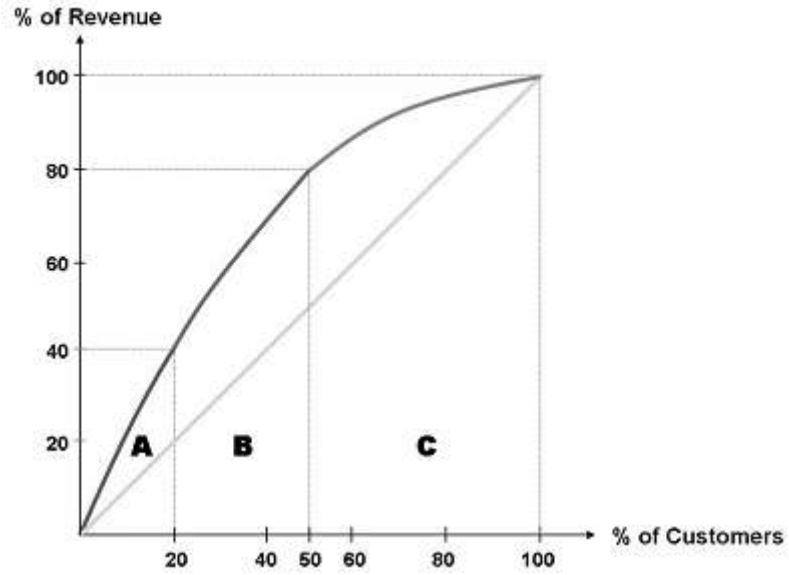

Figura 9: Resultados ABC, ganancias según clientes [SAPDOCS]

3.6 Análisis asociativo

3.6.1 Definición

El análisis asociativo, también denominado análisis de la cesta de la compra por sus usos particulares en este campo, es encontrar patrones y relaciones, especialmente en procesos de negocio, para poder formular reglas del tipo, "Si un cliente compra un producto A, este cliente también compra los productos B y C". [SAPDOCS].

Analizando estos patrones de comportamiento podemos, por ejemplo, predecir las próximas compras de los clientes de una empresa y anticiparnos a ello. Del mismo modo, podemos encontrar relaciones y patrones que desconocíamos en un primer momento y aprovechar este conocimiento para crear ofertas o promociones.

4. CASO DE ESTUDIO. PREDICCIÓN PRECIO DE VEHÍCULOS DE SEGUNDA MANO

4. 1 Introducción

Una vez introducidos algunos conceptos básicos sobre la minería de datos, pasaremos a relatar y detallar la realización de este proyecto.

En un principio, este proyecto iba a ser un análisis de los diferentes métodos que existen de minería de datos en SAP, uno de los mayores ERP a nivel internacional. Así como de tratar temas como el Business Intelligence y sistemas de Data Warehouse, es por ello que la mayoría de mis fuentes provienen de documentos creados por SAP. No obstante, al final no me fue posible la realización de este proyecto y opté por aplicar los conocimientos que había adquirido sobre minería de datos, en mi propio caso de estudio particular.

El caso de estudio en particular elegido fue la predicción del precio de vehículos de segunda mano. Ha sido este, como podría haberlo sido cualquier otro, tan solo necesitábamos un escenario al que poder aplicar los métodos y técnicas de minería de datos. Para ello, necesitábamos una serie de características para poder trabajar sobre este tema:

- Grandes cantidades de datos
- Información actual
- Temática conocida (conocimiento de las circunstancias "de negocio")
- Accesibilidad
- Posibilidad de verificación y creación de nuevos casos

Una vez escogida la temática, había que detallar como íbamos a desarrollar este caso de estudio. Tras barajar varias posibilidades, se optó por descargar los datos fuente de una web de artículos de segunda mano bastante conocida, fiable y de gran éxito en el sector. Para ello, habría que crear alguna aplicación o método mediante el cual pudiésemos extraer dicha información de las bases de datos del servidor de la web.

No fue tarea fácil dar con el código exacto que se ajustase a la estructura de los contenidos de la web, pero con un poco de esfuerzo y dedicación, no hay nada imposible para una mente dispuesta [Proverbio japonés].

Una vez tuviéramos los datos en nuestra aplicación, el siguiente paso sería poder exportar estos datos a algún formato con el que pudiésemos trabajar con ellos (como podemos ver, hasta en un sencillo caso de estudio como este, se están siguiendo los pasos de extracción de conocimiento del que hemos hablado con anterioridad en esta memoria).

En este momento, es donde entra la herramienta de minería de datos Weka. Hablaremos de ella más adelante, por ahora decir que se trata de una aplicación de código libre que incluye varias herramientas para los procesos de minería de datos. Una vez importados los datos a Weka, deberíamos trabajar y procesar los datos, aquí es donde entra la parte más interactiva del usuario clave.

El preprocesado y transformación de los datos es uno de los pasos más importantes de la minería de datos, pues al tratar la información previamente, disminuimos considerablemente el margen de error de nuestra predicción. Deberíamos eliminar campos innecesarios, limpiar o decidir qué hacer con los campos con valores vacios, considerar la utilización de uno u otro atributo, etc.

Con nuestros datos bien procesados y transformados, llega la hora de aplicar algún método o técnica de minería de datos. Se aplicarán varios métodos para observar las diferencias entre ellos y al llegar al mecanismo óptimo, se procederá a implementar en nuestra aplicación.

La implementación debería consistir en lo siguiente, un usuario podría escoger una marca de coche, un modelo, un año de fabricación, etc. Con estos datos introducidos en el programa, nuestra implementación del modelo debería brindarnos una estimación del precio del vehículo formulado. Esta estimación estaría basada en los datos obtenidos de miles de casos particulares de coches de segunda mano en venta reales.

4.2 La obtención de los datos

Antes de comenzar a detallar las características concernientes a la aplicación diseñada, merece especial mención el proceso realizado con anterioridad de análisis y estudio de la fuente de datos elegida.

Como hemos comentado, optamos por recoger los datos de una página web de artículos de segunda mano, la web elegida fue www.coches.net. En esta web podemos encontrar toda clase de vehículos de segunda mano. ¿Por qué esta web? Se trata de una web con unas cantidades bastante importantes de información, tiene una gran popularidad en el sector y cuenta con información bien estructurada, detallada y fiable (en la medida de lo posible en este sector).

De modo que el primer paso sería analizar la estructura de la web, en concreto del apartado concerniente a la venta y adquisición de automóviles de segunda mano. Como podemos observar en la figura siguiente, la web cuenta con un buscador de vehículos bastante completo. En éste, podemos seleccionar y filtrar nuestras búsquedas por varias características. En nuestro caso, queríamos datos en bruto, con lo que sólo filtraríamos por marca de vehículo. Cabe destacar, que para la realización de este trabajo, se necesitaban cantidades considerables de datos, por lo que sólo se barajaron y tuvieron en cuenta las marcas más reconocidas y con mayor número de artículos en venta.

Figura 10. Buscador de www.coches.net

Analizaremos ahora como esta web estructura la información obtenida de las búsquedas realizadas. Para facilitar la búsqueda, la web estructura los vehículos que se ajustan a los patrones de filtrado que hemos seleccionado con una vista preliminar del mismo.

Figura 11. Resultados previos

Como podemos observar en la figura 10, al realizar la búsqueda introduciendo únicamente la marca comercial de vehículos que queremos analizar, la página nos muestra la siguiente información previa. En ella podemos observar que nos dice el número exacto de vehículos que se han encontrado para nuestros criterios, y después mediante una tabla nos muestra por páginas una ficha preliminar de cada vehículo.

En esta ficha por desgracia, no contamos con todos los atributos que nos gustaría obtener de cada vehículo, pues el análisis con minería de datos quedaría bastante paupérrimo, pues en esta vista tan solo tenemos el modelo del vehículo, la provincia, el combustible, los kilómetros, el año de matriculación y el precio.

No obstante, si hacemos click sobre cada coche, nos lleva a una nueva página donde nos ofrece toda la información que el usuario que ha puesto a la venta el vehículo ha introducido en el sistema.

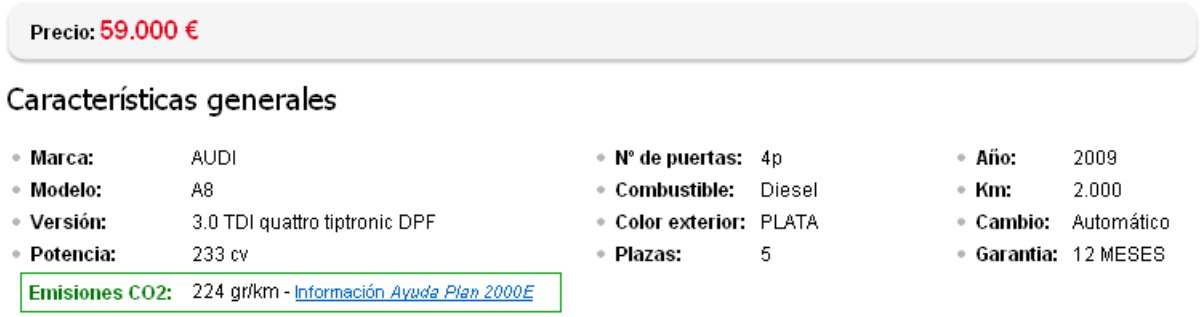

Figura 12. Datos extendidos del vehículo

En esta nueva vista, encontraremos fotos del vehículo, una descripción de las características, del equipamiento, de las condiciones de venta, del usuario vendedor, etc. Lo que a nosotros nos interesa son los datos mostrados en la figura 11, donde se amplían las especificaciones mostradas en la vista preliminar del automóvil. Podemos ver que se incluye la versión del modelo, la potencia, el número de puertas, el color e incluso las emisiones de CO2.

Tras el análisis del portal, llegamos a la conclusión de que es el adecuado. Contiene una gran cantidad de vehículos de segunda mano y ocasión, además de, como ya hemos comentado, una información estructurada y fiable.

No obstante, para extraer la información de la web, necesitaremos realizar un análisis en profundidad del código fuente de la página web. Pues la estrategia que seguiremos para obtener dichos datos será la siguiente:

1. Obtendremos el código fuente de la página web que se muestre con un hipervínculo dado.
2. Volcaremos este código en un fichero de texto auxiliar para poder tratar la información contenida en éste cómodamente.
3. Como hemos comentado, en una primera parte de la búsqueda, solo se muestra una vista preliminar del vehículo. De modo que accediendo a cada hipervínculo que la web proporciona para cada vista preliminar del vehículo, obtendremos el código fuente de la página particular de cada vehículo.
4. Identificaremos dentro de la web particular de cada vehículo dónde y cómo estructura los datos del automóvil la web.
5. Extraeremos dichos datos mediante procesos iterativos de búsqueda y los almacenaremos estructuradamente en nuestra aplicación.

El código fuente de cada página y el método para extraer los datos contenidos en él serán detallados más adelante en la memoria. Éste era un paso preliminar de análisis de nuestra fuente de datos para verificar que seríamos capaces de extraer esta información de una manera relativamente sencilla y sin problemas.

Tras este análisis, lo siguiente sería ponerse manos a la obra con la aplicación a medida que debíamos desarrollar. Lo cual, será detallado en el siguiente capítulo.

4.3 La aplicación

4.3.1 ¿Por qué VBA?

Antes de ponernos a programar, necesitamos realizar un pequeño análisis de los requerimientos de nuestra aplicación para poder decidirnos entre un lenguaje de programación u otro.

La aplicación ocupa un papel de obtención de datos y exportación de los mismos, por lo que tampoco necesitábamos un lenguaje de programación demasiado completo y complicado. No obstante, sí que resultaba de utilidad el contar con un entorno de trabajo que nos facilitara la utilización de bases de datos para agilizar los procesos de almacenamiento y obtención de datos una vez introducidos en nuestra aplicación.

Es por esta razón de necesidad de la parte de bases de datos que nos decantamos por utilizar el entorno de trabajo de Microsoft Access y su lenguaje de programación propio, que no es sino una reducción del exitoso y completo lenguaje de programación Visual Basic. En Access encontramos Visual Basic para Aplicaciones.

Visual Basic para Aplicaciones (VBA) es una implementación del lenguaje de programación Visual Basic 6 de Microsoft basado en el tratamiento de eventos, con un entorno de trabajo integrado en la mayoría de las aplicaciones del paquete de ofimática Microsoft Office. VBA nos facilita desarrollar funciones, automatizar procesos y acceso a Win32 y otras funcionalidades de nivel bajo mediante librerías DLL. Se puede utilizar para controlar muchos de los aspectos de la aplicación sobre la que se esté funcionando, incluyendo la manipulación de las funcionalidades de la interfaz, como menus o barras de herramientas, y trabajar con formularios personalizados y cuadros de diálogo.

Como su nombre sugiera, VBA está relacionada estrechamente con Visual Basic y utiliza Visual Basic Runtime, no obstante, por regla general solo puede ejecutar código desde una aplicación que hace de anfitrión (alguna del paquete Office), en vez de hacerlo como una aplicación por sí misma como ocurre con los programas desarrollados en Visual Basic. No obstante, puede utilizarse para controlar una aplicación desde otra utilizando la automatización OLE. Por ejemplo, podemos crear automáticamente un informe en Word desde datos obtenidos de Excel.

VBA es rico funcionalmente y flexible, no obstante tiene importantes limitaciones, como el soporte restringido para funciones con punteros utilizadas en las Windows API. No obstante, tiene la habilidad de utilizar, no crear, ActiveX/COM y consta de soporte para módulos de con clases.

4.3.2 Desarrollo de la aplicación

A continuación pasaremos a detallar todas y cada una de las características de la aplicación desarrollada. La estructura del detallado de la aplicación, consistirá en los pasos que he seguido para desarrollarla.

- hablaremos de las tablas que he creado y necesitado para la aplicación
- la selección de los atributos para las tablas creadas
- formularios creados para interactuar con el usuario y llevar el flujo de la información
- detallaremos los métodos utilizados

4.3.2.1 Tablas

No se ha necesitado la utilización de muchas tablas para el desarrollo de esta aplicación. No obstante, la funcionalidad de esta aplicación está basada en una única tabla principal y dos tablas auxiliares.

Tabla: coches

Esta tabla es la encargada de ir almacenando y rellenando todos los datos que vamos obteniendo mientras procesamos y filtramos los datos contenidos en los códigos fuente de las páginas web volcados en ficheros de texto. De este modo, los datos obtenidos serían fácilmente manejables tanto desde código como desde la propia aplicación para su tratamiento. Haciendo un alarde de originalidad, denominé a esta tabla "coches". Las características detalladas de la tabla son las siguientes:

Nombre del campo	Tipo de datos
Marca	Texto
Modelo	Texto
Potencia	Número
Puertas	Número
Combustible	Texto
Color	Texto
Plazas	Número
Anyo	Número
Km	Número
Provincia	Texto

Tabla 2. Estructura de la tabla "coches"

Tabla: Marks

Esta tabla se utiliza para servir de origen de datos de los cuadros combinados de 2 de los formularios que componen la aplicación. Aquí se almacenan las marcas comerciales de los vehículos junto a un identificador. Este identificador es el identificador que la web de coches de segunda mano les ha dado. Esto se hizo de este modo debido a que a la hora de obtener datos de la web, resultaba más útil para realizar búsquedas en la web, que tanto la aplicación como la web tuvieran el mismo identificador de marca para no confundirlas. Esta tabla está relacionada con la tabla Modelos mediante el atributo IdMarca. El tipo de relación es "uno a varios". Al relacionar ambas tablas, de nuevo facilitamos el tratamiento de los datos y de las búsquedas para "autorrellenar" cuadros combinados en la aplicación. Se puede observar que en la tabla no están todas las marcas comerciales de vehículos del mercado. Esto es debido a que se escapa del alcance de la aplicación y de este proyecto la inclusión de todas y cada una de las marcas existentes. Del mismo modo ocurrirá por tanto con los modelos de estas marcas, pues se han escogido los más conocidos para poder obtener datos suficientes para realizar la minería de datos.

Marks	
IdMarca	**Marca**
4	AUDI
7	BMW
11	CITROEN
14	FIAT
15	FORD
18	HYUNDAI
28	MERCEDES-BENZ
32	OPEL
33	PEUGEOT
35	RENAULT
39	SEAT
46	TOYOTA
47	VOLKSWAGEN
222	MINI

Tabla 3. Estructura de la tabla "Marks"

Tabla: Modelos

En esta tabla almacenamos los modelos de los vehículos de todas aquellas marcas comerciales que tenemos en la tabla Marks. Del mismo modo que ocurre con la tabla Marks, el identificador del modelo del vehículo es el mismo que se utiliza en la web de coches de donde extraemos los datos.

Modelos		
IdModelo	**idMarca**	**Modelo**
2	39	Ibiza
6	4	80
7	4	A4
8	28	Clase C
11	11	Xantia

13	14	Cinquecento
14	14	Seicento
17	35	Megane
27	4	A6
36	18	Lantra
37	15	Escort
38	15	Focus
39	15	Mondeo
40	15	Scorpio
41	28	Clase E
42	33	306
51	18	Sonata
55	28	Clase SL
56	28	Clase S
67	32	Astra
68	32	Vectra
70	7	Serie 3
71	7	Serie 5
75	14	Punto
76	14	Grande Punto
77	32	Corsa
81	33	106
82	33	205
83	47	Passat
86	15	Fiesta
89	47	Golf
90	35	Clio
94	39	Cordoba
96	4	S6
98	39	Toledo
103	4	S2
104	4	RS2
105	4	A8
109	47	Polo
121	18	Elantra
122	35	Twingo
128	11	ZX
130	11	C8
131	46	Rav4
132	18	Accent
134	46	Celica
150	46	Supra
153	35	Espace
155	33	405
156	33	406
163	14	Panda
171	7	Compact
176	15	Probe
177	7	Serie 7
178	7	Serie 8
188	32	Tigra
216	18	Coupe

225	39	Marbella
250	11	Saxo
251	11	C2
256	46	Land Cruiser
258	28	Clase SLK
269	28	Clase CLK
270	35	Scenic
272	46	Corolla
273	46	Auris
275	28	Clase A
276	14	Bravo
277	14	Stilo
279	33	407
282	7	Z3
289	15	Ka
291	7	Z4
293	33	307
305	46	Avensis
306	35	Grand Espace
322	46	Yaris
331	46	Prius
341	39	Alhambra
344	4	S8
345	4	A3
349	39	Arosa
352	11	Xsara
353	11	C4
356	18	Atos
359	4	S4
360	4	TT
361	47	Lupo
365	15	Puma
380	33	206
382	15	Cougar
385	32	Zafira
392	32	Agila
394	7	Z8
398	47	Bora
399	47	Jetta
400	14	Multipla
405	47	New Beetle
408	4	S3
410	39	Leon
413	11	Xsara Picasso
416	4	RS4
419	4	A2
425	18	Santa Fe
427	11	C5
428	18	Terracan
431	18	Matrix
438	7	X5
447	33	308

449	222	Mini
459	4	RS6
461	47	Touareg
462	47	Touran
474	28	SLR McLaren
476	15	Focus CMAX
477	35	Grand Scenic
478	35	Clio Campus
485	11	C3
490	15	Fusion
491	18	Getz
496	11	C3 Pluriel
497	15	Cmax
506	32	Meriva
514	7	Serie 6
515	7	X3
525	39	Altea
529	18	Tucson
539	7	Serie 1
542	28	Clase CLS
554	33	1007
555	46	Aygo
556	11	C1
558	33	107
566	28	Clase B
570	47	Fox
571	4	Q7
586	11	C6
587	33	207
590	15	SMAX
591	47	Eos
608	4	R8
611	4	A5
612	4	S5
619	11	C4 Picasso
622	18	i30
628	47	Tiguan
635	11	C15
636	47	Transporter
663	11	Jumper
668	28	Vito
670	11	Jumpy
674	35	Kangoo
678	11	Berlingo
681	33	Partner
729	35	R19
745	11	BX
760	15	Orion
761	11	C25
763	35	R5
764	35	R21
773	35	R4

775	32	Kadett
799	47	Scirocco
808	39	Malaga
813	35	R18
814	35	R11
821	33	309
864	7	Z1
868	4	V8
869	4	200
870	4	100
871	4	90
880	7	X6
882	15	Kuga
885	18	i10
890	11	C4 Sedan
891	28	Clase CLC
901	18	i800
904	4	Q5
913	14	Punto Classic
918	32	Insignia
923	39	Exeo
925	4	A4 Allroad Quattro
927	11	C3 Picasso
928	18	i20
935	46	iQ
937	46	Urban Cruiser

Tabla 4. Estructura de la tabla "Modelos"

4.3.2.2 Atributos

En esta sección explicaremos brevemente los atributos que hemos seleccionado como más importantes y adecuados para nuestro caso de estudio.

Marca: Corresponde a la marca corporativa del vehículo. En nuestro proceso de minería de datos trabajaremos con todos los modelos de una marca determinada para realizar nuestro modelo.

Modelo: Se trata del modelo del vehículo, el nombre comercial de una gama de vehículos de una misma marca.

Potencia: La potencia del vehículo expresada en caballos (CV).

Puertas: El número de puertas de las que dispone el vehículo.

Combustible: Especificar el tipo de combustible que utiliza el vehículo, diesel, gasolina…

Color: Pintura aplicada al vehículo.

Plazas: número máximo de plazas autorizadas del vehículo.

Anyo: Año de primera matriculación del vehículo.

Km: Kilómetros que se han hecho con dicho vehículo.

Provincia: Provincia desde la que se oferta la venta de dicho vehículo.

4.3.2.3 Formularios

La aplicación está compuesta por tres formularios básicos que proporcionan el funcionamiento y funcionalidad al programa.

MinnaCar

Este es el formulario de inicio de la aplicación. Se trata de un formulario básico para acceder a las funcionalidades del programa. Como podemos observar en la captura de pantalla del mismo, consta únicamente de 3 botones de comando.

- Iniciar Búsqueda nos llevará al formulario "Inicio".
- Predicción, nos llevará al formulario "Predicción".
- Documentación, enlazará a esta memoria.

Figura 13. Formulario MinnaCar

Inicio

En este formulario se concentra todo el peso de la aplicación, pues bajo el código del formulario se encuentran los métodos de obtención de los datos necesarios de los vehículos anunciados en la página web.

En el formulario, el usuario seleccionará una marca de vehículos para obtener los datos. Estos son los datos necesarios que se deben almacenar en la aplicación y exportar posteriormente a Weka para la realización del modelo de minería de datos, por lo que este es el primer paso que deberemos realizar para cada marca de coche que deseemos analizar.

El funcionamiento es muy simple, seleccionaremos la marca que deseamos analizar y pulsaremos el botón de Buscar. La aplicación buscará en la web todos los vehículos de segunda mano de dicha marca y los introducirá en la tabla Coches uno a uno.

Tras esto, podremos pulsar Exportar, para transformar esta tabla en un archivo compatible con Weka. Se ha optado por el formato de Excel .csv pues es totalmente compatible con Weka y nos permite mantener la estructura de tabla original.

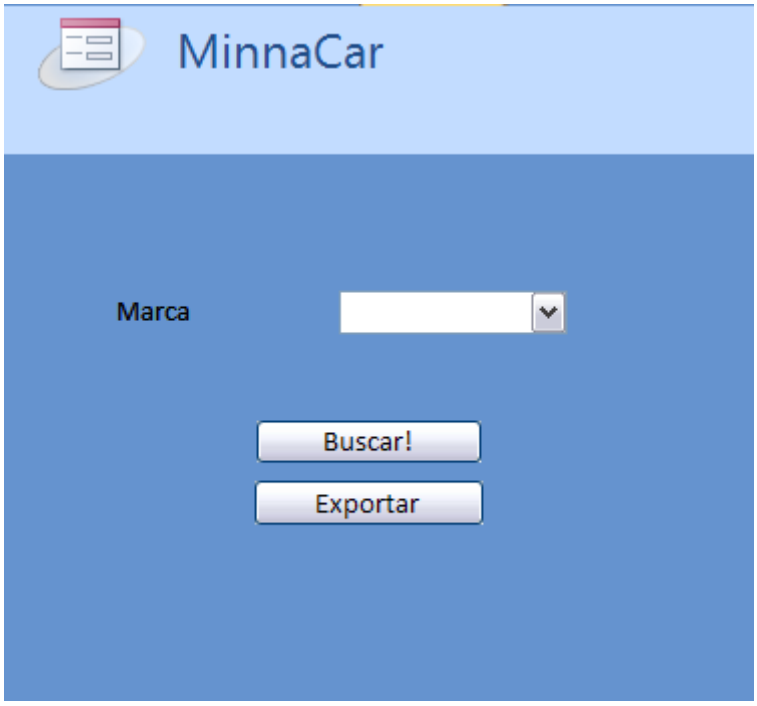

Figura 14. Formulario Inicio

A continuación pasaremos a especificar el código de programación que lleva por debajo este formulario.

El formulario está compuesto por varios métodos que relataremos a continuación. Primeramente describiré las variables globales al formulario que se han utilizado.

`Dim url As String`	Variable que almacenará la url que se irá construyendo. Esta url será la url a la que el programa accederá para descargarse el código fuente y obtener los datos.
`Dim num_pag As String`	Variable contador que almacenará el número de página por el que vamos en la búsqueda en la web.
`Dim Marca As String`	La marca de vehículo seleccionada para el análisis.

`Dim num_coches As Integer`	El número de vehículos encontrados para la marca seleccionada.
`Dim contador As Integer`	Un simple contador.

Tabla 5. Variables globales

Como decíamos antes, la funcionalidad de este formulario comienza al pulsar el botón de Buscar! Tras seleccionar una marca de vehículo se pone en ejecución el siguiente método, con él, extraemos el valor del cuadro combinado que corresponde a la selección de marca del vehículo y lo concatenamos con la url genérica de la página web para construir la url de la que necesitaremos extraer el código fuente. Como se puede observar, la url es bastante sencillo de comprender su funcionamiento. Tenemos por un lado el nombre de la url general, después viene la sección de la página que estamos viendo. PG=1 significa que queremos ver la página 1 de los resultados de la búsqueda y el Id corresponde a la marca del vehículo que queremos buscar. Este Id es el que está plasmado en la tabla Marks, tuve que extraer todos los Id de la web para que pudieran coincidir y facilitar la extracción de los datos.

Como aclaración, los métodos AfterUpdate, como su nombre indica, son métodos que se ejecutan tras actualizar el valor del componente.

```
Private Sub Cuadro_combinado17_AfterUpdate()

Marca = Me.Cuadro_combinado17.Value

url = "http://www.coches.net/coches-de-ocasion.aspx?pg=1&MakeId=" & Marca

End Sub
```

Al pulsar este botón comienza la siguiente ejecución de métodos. Iré relatando la funcionalidad de cada método incluyendo comentarios en el código del programa.

```
Private Sub cmdBuscar_Click()

#Variables privadas del método

      #Número de páginas que almacenan datos de la búsqueda

      Dim num_paginas As Integer

      #Variable auxiliary para crear los archivos que contienen el código
      #fuente de la página web descargada

      Dim MyFile As String

      #Variable para almacenar el directorio de trabajo donde se guardarán
      #las páginas descargadas

      Dim path As String
```

```
#Variables FileSystemObject para la apertura y tratamiento de ficheros de texto

Dim fso,f

#Variable booleana para el bucle de búsqueda

Dim boolStFnd As Boolean


#Constantes

#Esta constant es para utilizarla en el método Dir$, indicando el #valor 16 significa que lo que buscamos es un directorio.

Const ATTR_DIRECTORY = 16

#Especificar en el método OpenTextFile que abriremos el archivo sólo #para lectura

Const ForReading = 1

#Primero vaciaremos la tabla donde almacenamos los datos de los #vehículos, ejecutando un delete sobre la tabla Coches.

DoCmd.SetWarnings False

DoCmd.RunSQL "Delete * from coches"


#Estableceremos el directorio de trabajo del programa, creando un #nuevo subdirectorio. Para ello, comprobaremos si el directorio #existe, si no existe lo crearemos. Aquí almacenaremos los archivos #que contendrán el código fuente de las páginas web.

path = CurrentProject.path & "\paginas"

If Dir$(path, ATTR_DIRECTORY) = "" Then

      MkDir path

End If

#Llamada al método encargado de descargar cada página web, este es #otro método diferente, no obstante incluiremos su funcionamiento #dentro de este para no perder el hilo de ejecución y facilitar la #comprensión del programa.

A = Download(url, path & "\HTML1")
```

```
#Inicio del Método Download

#Llamada al método:

#A= download("http://www.google.com","c:\google.html")

#Recibe como parámetros la web a descargar y la ruta con el nombre
#de fichero donde se almacenará el código fuente de la web dada.

#Devuelve TRUE si tiene éxito.

Public Function download(url, dest)

      On Error Resume Next

      Err.Clear

#Obtenemos la página web haciendo un GET de la url que nos han dado
#como parámetro de llamada al método

      With CreateObject("Microsoft.XMLHTTP")

            .Open "GET", url, False

            .send

            b = .responseBody

#Comprobamos si la llamada ha resultado con éxito, si se ha
#producido un error o tarda demasiado en responder, devuelve false y
#sale del método.

            If Err.Number <> 0 Or .Status <> 200 Then

                  download = False

Exit Function

            End If

      End With

#Si no ha fallado, escribimos el contenido de b, que no es otro que
#el código fuente de la página web en el archivo destino que se ha
#especificado

With CreateObject("ADODB.Stream")

      .Type = 1

      .Open

      .write b
```

```
#Como vemos, guardamos el contenido de b en el fichero especificado
#en la variable dest

      .SaveToFile dest, 2

End With

download = Err.Number = 0

End Function

#Fin del método Download
```

Para comprender el fragmento de código siguiente es necesario que analicemos en primer lugar la estructura del código fuente de la página que nos hemos descargado. Es irrelevante poner el código al completo, por lo que analizaremos aquellos fragmentos de los que deseemos extraer información importante para el caso.

Tras la búsqueda, lo primero que extraeremos de la página será el número de vehículos encontrados para la marca que hemos seleccionado. Cuando leemos un fichero de texto, empezaremos desde el principio hasta el final, por tanto, como este número se muestra el principio de la página, aprovecharemos el recorrido por el código fuente para extraerlo.

```
<!-- Cabezera
de la Grid -->
              <div id="search_info">
              <div
              id="_ctl0_ContentPlaceHolder1_Grid1_info_results">
              <h1>
              Encontrados <strong>6.246</strong> coches AUDI de
              segunda mano
              </h1>
              </div>
```

Encontrados **6.246** coches AUDI de segunda mano

Este es el código que corresponde al "Encontrados….". Como vemos, deberemos buscar en nuestro código la siguiente cadena "Encontrados , para extraer el número de vehículos encontrados.

```
#Primeramente, crearemos un objeto tipo FileSystemObject para poder
#abrir y analizar un fichero de texto como si de un String se
#tratase. El método OpenTextFile tan solo necesita la ruta, la forma
#en la que queremos abrir el archivo y el modo de codificación, True
#para Unicode.

Set fso = CreateObject("Scripting.FileSystemObject")
```

```
Set f = fso.OpenTextFile(path & "\HTML1", ForReading, True)

#Inicializamos la variable a false para iniciar la búsqueda

boolStFnd = False

#Creamos una variable auxiliar para almacenar lo leído del texto
#extraído del código fuente descargado y otra para almacenar el
#número de páginas en formato texto

Dim A, num_pag as string

#Recorremos el bucle mientras no hayamos llegado al final del
#fichero o hayamos encontrado el número de coches.

Do While f.AtEndOfStream <> True And boolStFnd <> True

      A = f.readline

      If InStr(A, "Encontrados <strong>") <> 0 Then

          #Cuando encontramos la línea, ponemos a true la booleana

          boolStFnd = "True"

      #Con la función InStr buscamos la posición del número de
      #vehículos encontrados dentro de la línea. Buscaremos la
      #posición inicial y la final. Utilizaremos la expresión de
      #código HTML "<strong><\strong>" para localizar el principio y
      #el fin. Después extraeremos de la línea esta cadena con la
      #función Mid, que extrae una cadena de una línea dada y solo
      #necesita las posiciones de inicio y fin de la cadena dentro
      #de la línea.

          posIni = InStr(1, A, "<strong>")

          posFin = InStr(1, A, "</strong>")

          num_pag = Mid(A, posIni + 8, ((posFin - (posIni + 8))))

          #Extraemos el "." Para poder convertir el string en int

          num_pag = Replace(num_pag, ".", "")

        #Transformamos num_pag en int y lo almacenamos en num_coches

          num_coches = Val(num_pag)
```

```
            #El número de páginas que contienen resultados es igual a la
            #división entera del número de vehículos entre 30, porque en
            #cada página se muestran 30 anuncios de vehículos

                num_paginas = num_coches / 30

            #Aquí comprobamos la posibilidad de que el número de vehículos
            #encontrados sea menor que 30, con lo que sólo habrá una
            #página de resultados

                If num_paginas <= 0 Then

                    num_paginas = 1

                End If

            #Mostramos una ventana con el número de coches encontrados

                MsgBox "el número de coches es: " & num_coches

            End If

    Loop

    f.Close
```

Una vez obtenido el número de vehículos encontrados y calculado el número de páginas, llega el momento de descargar la página individual de cada vehículo. Para ello, primero descargaremos la página principal con el listado entero de vehículos de dicha página y después cada página de cada vehículo.

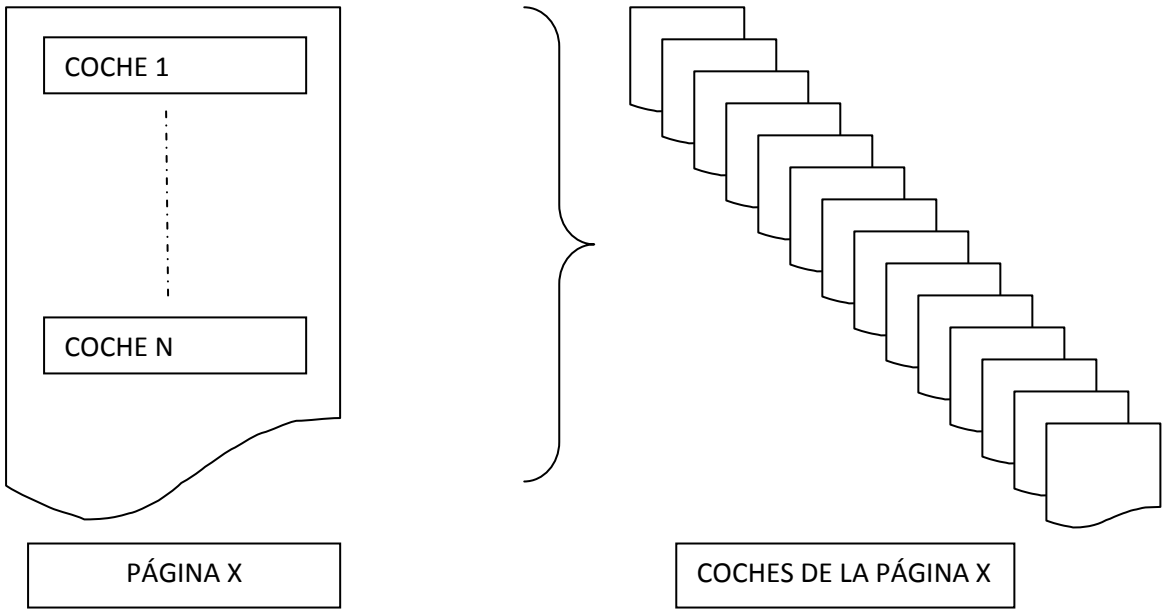

Figura 15. Esquema de páginas web

Primero explicaré el funcionamiento del fragmento de código para después especificar uno de los métodos más importantes del código. Utilizaremos un bucle principal que se ejecutará tantas veces como páginas con resultados se hayan obtenido. Para cada página, construirá la url con el número de página que corresponde según el contador del bucle "i" y descargará su código fuente mediante el método ya explicado "Download". Irá almacenando cada página con nombre "HTMLi"

```
For i = 1 To num_paginas

      #Comprobamos que todavía quedan vehículos (en el método fnFindText
      #vamos restando 1 al número total de vehículos cada vez que se
      #procesa uno

        If num_coches >= 0 Then

            #Descargamos la página principal

            url = "http://www.coches.net/coches-de-ocasion.aspx?pg=" & i &
            "&MakeId=" & Marca

            A = Download(url, path & "\HTML" & i)

            #Hacemos la llamada al método que se encargará de descargar
            #cada página individual de cada vehículo, extraer la
            #información y almacenarla. Se le pasa una ruta de un archivo
            #de texto donde buscar y una cadena a buscar. En este caso, se
            #le pasa la página principal por la que vamos y la cadena <div
            #id = "gridRows"> que es la cadena que indica en el código
            #fuente de la página web principal que a continuación viene la
            #tabla que almacena la referencia de cada vehículo individual.

            A = FindText(path & "\HTML" & i, "<div id=""gridRows"">")

        Else

            #si el número de vehículos es menor que cero, hemos acabado y
            #salimos del bucle haciendo i = num_paginas

            i = num_paginas

        End If

Next i
```

Antes de especificar el código del método FindText, simularemos que el programa ha acabado toda su ejecución. El siguiente fragmento de código se utiliza para eliminar todos los ficheros de texto que se han creado al descargar todas las páginas web. Esto es importante, porque si una búsqueda ha alcanzado los 6000 vehículos, significa que habrán más de 6000 ficheros de texto creados. Se podría haber sobreescrito el mismo fichero una y otra vez, pero me dificultaba las tareas de trazabilidad a la hora de depurar el código.

```
#Primero especificamos una ruta completa donde buscar

MyFile = Dir$(CurrentProject.path & "\paginas\*.*")

#Mientras hayan ficheros en el directorio especificado…

Do While MyFile <> ""

      #hacemos la llamada al método limpiador

    KillProperly CurrentProject.path & "\paginas\" & MyFile

    'need to specify full path again because a file was deleted 1

    MyFile = Dir$(CurrentProject.path & "\paginas\*.*")

Loop

#Función KillProperly, encontré este código por internet y me ha sido de gran ayuda para eliminar todos los archivos encontrados en un directorio dado. Tan solo necesita una cadena que especifique el nombre del fichero y lo elimina.

Public Sub KillProperly(Killfile As String)

    If Len(Dir$(Killfile)) > 0 Then

        SetAttr Killfile, vbNormal

        Kill Killfile

    End If

End Sub

#Al finalizar el método cmdBuscar, se muestra un mensaje de información al usuario para indicar que todo ha salido correctamente

MsgBox "La búsqueda ha finalizado con éxito y los coches se han introducido en la base de datos"

End Sub
```

A continuación, especificaremos el código más complejo e importante de la aplicación. Este código se encargará de descargarse cada página individual de cada vehículo encontrado y extraer toda la información que nos interesa para después introducirla en una tabla de la base de datos. Para ello, tiene que ir buscando línea por línea del código fuente de la página para encontrar los datos relevantes, ir almacenándolos y una vez encontrados todos crear la consulta SQL para introducirlos en la tabla correspondiente de la base de datos.

```
#El método recibe como parámetros la ruta del fichero a buscar y el primer
#string que dará comienzo a la búsqueda de vehículos

Function FindText(strFilePath, strSrTxt)

#Primero comprobaremos que la ruta existe y que el fichero que buscamos
#existe

   If Len(Dir$(strFilePath)) > 0 Then

        strFileRes = "File Exists"

   Else

        strFileRes = "File doesn't exist"

   End If

#Si el fichero existe, lo abrimos en modo lectura para trabajar con él

   If strFileRes = "File Exists" Then

        Const ForReading = 1

        Set fso = CreateObject("Scripting.FileSystemObject")

        Set f = fso.OpenTextFile(strFilePath, ForReading, True)

#Empezamos el bucle que recorrerá la página principal que contiene las
#subpáginas con los vehículos

        Do While f.AtEndOfStream <> True

            A = f.readline

            linea = A

#Buscamos la cadena que recibimos de la llamada al método y reseteamos las
#variables auxiliares que contendrán los datos a extraer de los vehículos

            If InStr(A, strSrTxt) <> 0 Then

                boolStFnd = "True"
```

```
                fnFindText = "Text found" + A
                cochemod = ""
                provincia = ""
                Combustible = ""
                kilometros = ""
                precio = ""
                anio = ""
                sql = ""
                i = 0
#Variables de control de bucle, para salir del bucle y para pasar al
#siguiente vehículo una vez extraídos todos los datos del vehículo en
#cuestión
                textfound = False
                nextcar = False
#Ahora buscaremos la url de cada coche para descargarnos su página
#individual
                While f.AtEndOfStream <> True And textfound <> True
                   linea = f.readline
                   posIni = 0
                    posFin = 0
#Buscamos la linea <script language=" para comprobar que aún quedan
#vehículos por procesar. Esta línea la encontré fijándome en el código
#fuente de la página web, cuando aparece esta línea es el final de la
#tabla de vehículos
                    If InStr(linea, "<script language=") Then
                        textfound = True
                    End If
#Extraemos la url de la página del vehículo como ya hemos comentado con
anterioridad
                    If InStr(linea, "<a href=") Then
                        i = i + 1
```

```
                    posIni = InStr(1, linea, "<a href=")

                    posFin = InStr(1, linea, "title=")

                    cochemod = Mid(linea, posIni + 9, ((posFin -
(posIni + 11))))

                    urlcoche = "http://www.coches.net" & cochemod

#Descargamos la web individual del coche que hemos encontrado

                    A = download(urlcoche, CurrentProject.path &
                "\paginas\Coche" & i)

#Extraemos la provincia del resumen previo del vehículo en la página
#principal

                While f.AtEndOfStream <> True And provinciaenc <> True

                    linea = f.readline

                    If InStr(linea, "<div
                class=""provincia_small""><p><span>") Then

                    posIniP = InStr(1, linea, "<span>")

                    posFinP = InStr(1, linea, "</span>")

                    provincia = Mid(linea, posIniP + 6, ((posFinP -
                (posIniP + 6))))

                    provinciaenc = True

                    ElseIf InStr(linea, "<div
                class=""provincia_small""><p>") Then

                    posIniP = InStr(1, linea, "<p>")

                    posFinP = InStr(1, linea, "</p>")

                    provincia = Mid(linea, posIniP + 3, ((posFinP -
                (posIniP + 3))))

                    provinciaenc = True

                    End If

                    Wend
```

```
#Abrimos el código fuente de la web individual y empezamos la búsqueda en
#este nuevo fichero dejando el otro abierto por donde nos hemos quedado

                    Set fso2 = CreateObject("Scripting.FileSystemObject")

                    Set f2 = fso2.OpenTextFile(CurrentProject.path &
"\paginas\Coche" & i, ForReading, True)

                       Do While f2.AtEndOfStream <> True

                           A2 = f2.readline

                           linea2 = A2

                           posIni2 = 0

                           posFin2 = 0

#Vamos extrayendo del nuevo código fuente todos los datos relevantes del
#vehículo, leyendo cada línea del fichero de texto y buscando las cadenas
#que indican el dato que buscamos. A medida que los encontramos vamos
#almacenándolos en las variables auxiliares

                           If InStr(linea2, "=""txtprecio"">") Then

                            posIni2 = InStr(1, linea2, "=""txtprecio"">")

                           posFin2 = InStr(1, linea2, "&euro")

                            PrecioCoche = Mid(linea2, posIni2 + 13,
                            ((posFin2 - (posIni2 + 13))))

                           ElseIf InStr(linea2, "<ul id=""ftcol2") Then

                              linea2 = f2.readline

                              posIni2 = InStr(1, linea2, "<li><p>")

                              posFin2 = InStr(1, linea2, "</p>")

                              MarcaCoche = Mid(linea2, posIni2 + 7,
((posFin2 - (posIni2 + 7))))

                              linea2 = f2.readline

                              posIni2 = InStr(1, linea2, "<li><p>")

                              posFin2 = InStr(1, linea2, "</p>")

                              ModeloCoche = Mid(linea2, posIni2 + 7,
((posFin2 - (posIni2 + 7))))

                              linea2 = f2.readline
```

```
                                linea2 = f2.readline
                                posIni2 = InStr(1, linea2, "<li><p>")
                                posFin2 = InStr(1, linea2, "</p>")
                                PotenciaCoche = Mid(linea2, posIni2 + 7,
((posFin2 - (posIni2 + 7))))
                            ElseIf InStr(linea2, "<ul id=""ftcol4") Then
                                linea2 = f2.readline
                                posIni2 = InStr(1, linea2, "<li><p>")
                                posFin2 = InStr(1, linea2, "</p>")
                                PuertasCoche = Mid(linea2, posIni2 + 7,
((posFin2 - (posIni2 + 7))))
                                linea2 = f2.readline
                                posIni2 = InStr(1, linea2, "<li><p>")
                                posFin2 = InStr(1, linea2, "</p>")
                                CombustibleCoche = Mid(linea2, posIni2 +
7, ((posFin2 - (posIni2 + 7))))
                                linea2 = f2.readline
                                posIni2 = InStr(1, linea2, "<li><p>")
                                posFin2 = InStr(1, linea2, "</p>")
                                ColorCoche = Mid(linea2, posIni2 + 7,
((posFin2 - (posIni2 + 7))))
                                linea2 = f2.readline
                                posIni2 = InStr(1, linea2, "<li><p>")
                                posFin2 = InStr(1, linea2, "</p>")
                                PlazasCoche = Mid(linea2, posIni2 + 7,
((posFin2 - (posIni2 + 7))))
                            ElseIf InStr(linea2, "<ul id=""ftcol6") Then
                                linea2 = f2.readline
                                posIni2 = InStr(1, linea2, "<li><p>")
                                posFin2 = InStr(1, linea2, "</p>")
```

```
                                AnyoCoche = Mid(linea2, posIni2 + 7,
((posFin2 - (posIni2 + 7))))
                                linea2 = f2.readline
                                posIni2 = InStr(1, linea2, "<li><p>")
                                posFin2 = InStr(1, linea2, "</p>")
                                KmCoche = Mid(linea2, posIni2 + 7,
((posFin2 - (posIni2 + 7))))
                                linea2 = f2.readline
                                posIni2 = InStr(1, linea2, "<li><p>")
                                posFin2 = InStr(1, linea2, "</p>")
                            End If
                        Loop
#Una vez hemos extraído todos los datos del fichero del código fuente de
#la página del vehículo, disminuimos el contador de vehículos y comenzamos
#a dar formato a los datos
                            num_coches = num_coches - 1
#Para introducirlos correctamente en la base de datos, eliminaremos
#comas, comillas y demás caracteres innecesarios y que estorban a la hora
#de introducirlos en la tabla final
                        PlazasCoche = Replace(PlazasCoche, "'", "")
                        ColorCoche = Replace(ColorCoche, "'", "")
                        ColorCoche = Replace(ColorCoche, ",", "")
                        PotenciaCoche = Replace(PotenciaCoche, "'", "")
                        PotenciaCoche = Replace(PotenciaCoche, "cv", "")
                        PuertasCoche = Replace(PuertasCoche, "'", "")
                        PuertasCoche = Replace(PuertasCoche, "p", "")
                        MarcaCoche = Replace(MarcaCoche, "'", "")
                        ModeloCoche = Replace(ModeloCoche, "'", "")
                        ModeloCoche = Replace(ModeloCoche, """", "")
                        provincia = Replace(provincia, "'", "")
                        KmCoche = Replace(KmCoche, "'", "")
```

```
                    KmCoche = Replace(KmCoche, "'", "")
                    PrecioCoche = Replace(PrecioCoche, "'", "")
                    PrecioCoche = Replace(PrecioCoche, " &euro;", "")
                    CombustibleCoche = Replace(CombustibleCoche, "'",
"")
#Debido a la codificación HTML del código fuente de las páginas web,
#acentos y otros caracteres especiales vienen codificados de manera poco
#legible, por lo que tenemos que substituir estos casos por la palabra
#adecuada
                  Select Case provincia
                    Case "A CoruÃ±a"
                        provincia = "A Coruña"
                   Case "AlmerÃ-a"
                        provincia = "Almería"
                    Case "Ã vila"
                        provincia = "Ávila"
                    Case "CÃ¡ceres"
                        provincia = "Cáceres"
                    Case "CÃ¡diz"
                        provincia = "Cádiz"
                    Case "CÃ³rdoba"
                       provincia = "Córdoba"
                    Case "CastellÃ³n"
                        provincia = "Castellón"
                    Case "JaÃ©n"
                        provincia = "Jaén"
                    Case "LeÃ³n"
                        provincia = "León"
                    Case "MÃ¡laga"
                        provincia = "Málaga"
```

```
                        Case "GuipÃºzcoa"
                            provincia = "Guipúzcoa"
                        Case Else    ' Otros valores.
                            provincia = provincia
                    End Select
#Construimos la expresión SQL que introducirá los datos en la tabla #indicada
                    sql = "INSERT INTO coches (Marca, Modelo, Potencia, Puertas, Combustible, Color, Plazas, Anyo, Km, Precio, Provincia) VALUES ('" & MarcaCoche & "','" & ModeloCoche & "','" & PotenciaCoche & "','" & PuertasCoche & "','" & CombustibleCoche & "','" & ColorCoche & "','" & PlazasCoche & "','" & AnyoCoche & "','" & KmCoche & "','" & PrecioCoche & "','" & provincia & "')"
                    DoCmd.SetWarnings False
                    DoCmd.RunSQL sql
#Reseteamos los valores
                   MarcaCoche = "-"
                    ModeloCoche = "-"
                    VersionCoche = "-"
                    PotenciaCoche = ""
                    PuertasCoche = ""
                    CombustibleCoche = "-"
                    ColorCoche = "-"
                    PlazasCoche = "-"
                    AnyoCoche = "-"
                    KmCoche = "-"
                    PrecioCoche = "-"
                    provincia = "-"
                    sql = ""
                  End If
            Wend
```

```
    End If

#Tras llegar aquí, continuaremos con el fichero del código fuente de la
#página principal para procesar el siguiente vehículo, descargar su código
#fuente y extraer sus datos.

    Loop

#Comprobadores por si el fichero no existe o la cadena a buscar no se ha
#encontrado

        If boolStFnd <> "True" Then

            fnFindText = "Text not found"

        End If

            f.Close

        Else

        fnFindText = "File does not exist"

        End If

End Function
```

Llegados a este punto, el programa tras un tiempo de procesado de todas las páginas web (depende del número de vehículos encontrados, puede llegar a tardar unos minutos), tenemos en una tabla en nuestro sistema los datos de todos los vehículos que hemos encontrado. Necesitaremos ahora exportar dichos resultados a un fichero externo para poder tratarlo a continuación con Weka.

Para exportar los datos, tan sólo tendremos que pulsar el botón Exportar! Y se nos generará el fichero exportado en el directorio de trabajo donde tengamos la base de datos. El código es el siguiente:

```
Private Sub Comando21_Click()

      On Error GoTo Err_export

      #Esta es una función de visual basic para exportar el contenido de
      #una tabla de nuestro sistema a un fichero externo, en este caso, un
      #fichero csv. Para ello, tan sólo necesita una "especificación de
      #exportación" llamada cochesA, que hemos creado previamente, donde
      #decimos el formato de las columnas, los separadores de campo, etc.
      #El nombre de la tabla que queremos exportar y el nombre del fichero
      #que se va a crear.
```

```
            DoCmd.TransferText    acExportDelim,    "cochesA",    "coches",
            CurrentProject.path & "\cochesG.csv"


      Exit_export:

Exit Sub

      Err_export:

      MsgBox Err.Description

      Resume Exit_export

End Sub
```

Ahora ya disponemos de un fichero con los datos de los vehículos que hemos extraido de la página web. Este fichero contendrá los datos que los anunciantes han puesto sobre sus vehículos, por lo que necesitará un tratamiento y procesado previo para su trabajo con Weka, no obstante, de esto hablaremos en el siguiente punto de la memoria.

Por ahora, dejaremos de lado la aplicación en Access para hablar del entorno de trabajo en Weka, retomaremos la aplicación para la implementación del modelo de minería de datos descubierto.

4.4 Weka

4.4.1 ¿Qué es Weka?

Técnica y biológicamente hablando, una Weka (Gallirallus australis) es un ave procedente de Nueva Zelanda. Se trata de una especie en peligro de extinción y es famosa por su curiosidad y agresividad. Sería como la "codorniz/perdiz neo zelandesa" por así decirlo, pues su alimentación basada en insectos y pequeños frutos y su aspecto nos recuerdan a estas especies ibéricas.

Figura 16. Weka con polluelo

Dejando de lado la biología, weka se trata de un acrónimo derivado de Waikato Environment for Knowledge Analysis – Entorno para Análisis del Conocimiento de la Universidad de Waikato. Esto es porque fue esta universidad la que inició el desarrollo de Weka en 1993, no obstante, su desarrollo original fue hecho en TCL/TK y C, para en 1997 pasar a reescribirse todo el código fuente del entorno en Java, una plataforma más universal, y añadir las implementaciones de diferentes algoritmos de modelado.

Figura 17. Interfaz principal

Weka está compuesta por una serie de herramientas gráficas de visualización y diferentes algoritmos para el análisis de datos y modelado predictivo. Su interfaz gráfica de usuario nos facilita el acceso a sus múltiples funcionalidades.

Esta potente herramienta de minería de datos se encuentra libremente disponible bajo la licencia pública general de GNU, además, al estar implementada en Java como ya hemos comentado, puede ejecutarse prácticamente bajo cualquier entorno.

La interfaz gráfica de Weka cuenta con 4 formas de acceso a las diferentes funcionalidades de la aplicación.

- Simple CLI (Simple command-line interface), que no es más que el acceso a través de consola de comandos a todas las opciones de Weka.

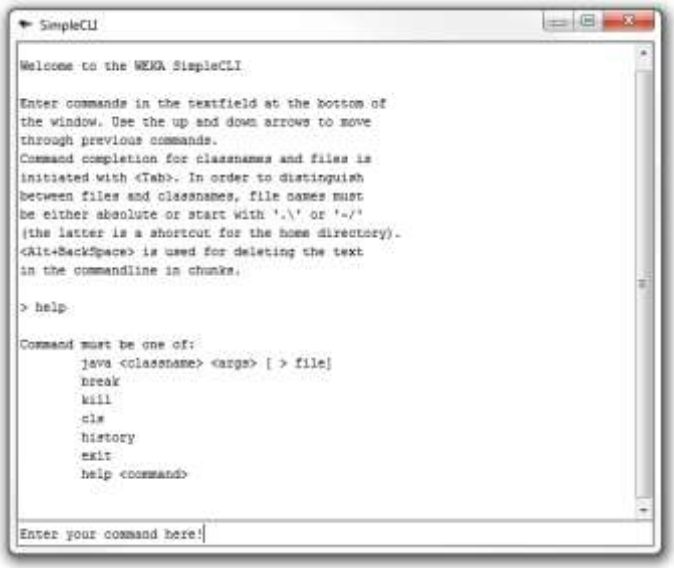

Figura 18. Ventana de Comandos

- Explorer, es la opción más intuitiva para el usuario, pues dispone de varios paneles que dan acceso a las principales características del programa

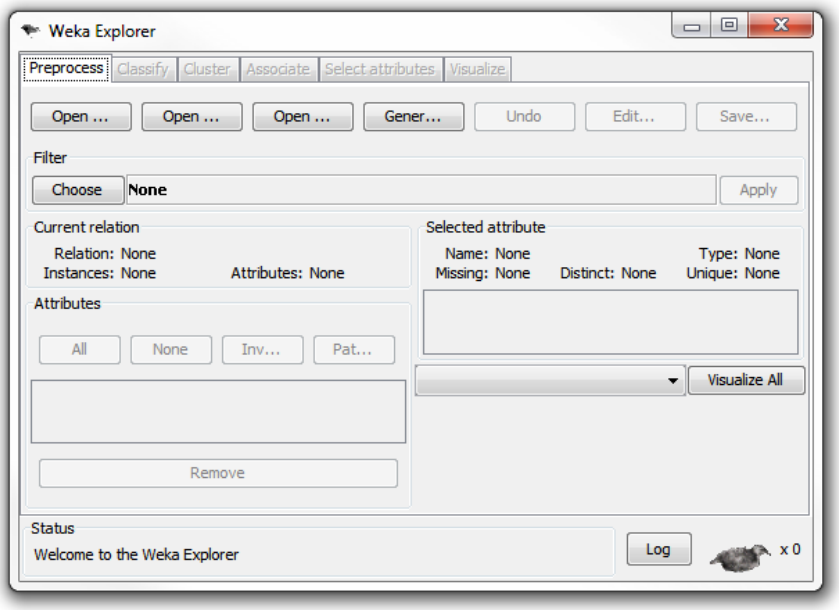

Figura 19. Explorer

- Experimenter, permite la comparación sistemática de una ejecución de los algoritmos predictivos de Weka sobre una colección de conjuntos de datos.

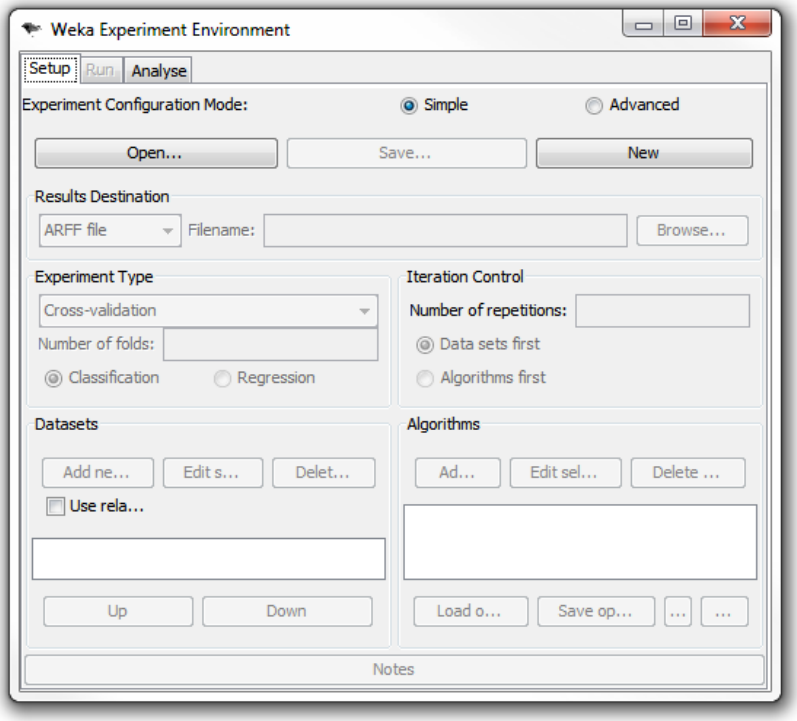

Figura 20. Experimenter

- Knowledge Flow, soporta esencialmente las mismas opciones que la interfaz Explorer, pero esta permite "arrastrar y soltar". Ofrece soporte para el aprendizaje incremental.

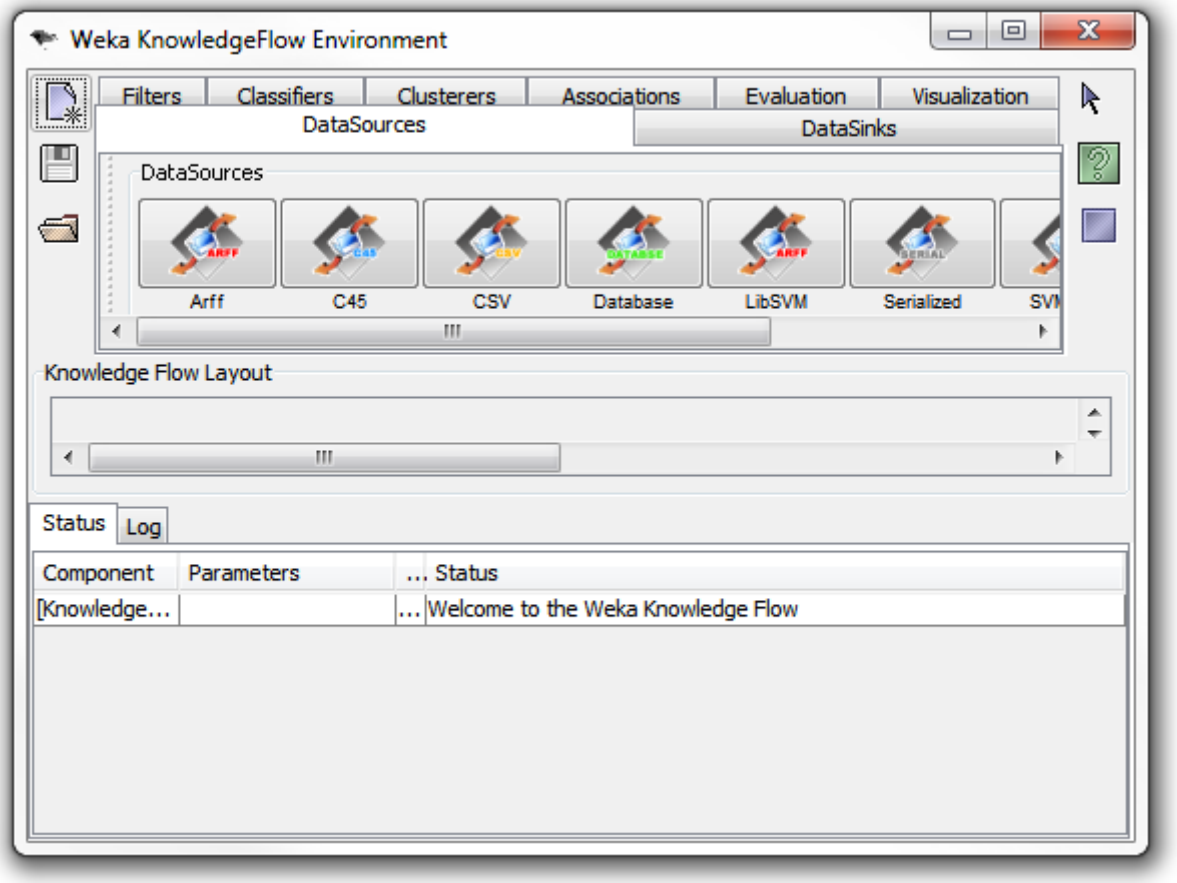

Figura 21. Knowledge flow

4.4.2 Preparación de los datos

Procederemos ahora a relatar el proceso de tratamiento de los datos obtenidos y exportados al fichero csv mediante Weka. Estos datos, habrá que procesarlos detenidamente para que los resultados que obtengamos sean lo más precisos posibles, es decir, con la menor media de error que seamos capaces de conseguir.

Para ello, tendremos que eliminar atributos que no nos proporcionen información relevante para predecir el precio del vehículo. Registros con campos vacíos, que sólo introducen e inducen a errores, valores mal escritos y no asociables, etc.

Al abrir el fichero csv en Weka, el primer problema con el que nos encontramos es que en las descripciones de algunos vehículos, el anunciante ha puesto varios colores separados por comas, por lo que estos registros rompen la estructura del csv, pues es un fichero con los campos separados por comas. Para solventarlo, procedemos a eliminar estos registros.

Ya con el fichero abierto en Weka, para poder trabajar mejor con él, lo guardaremos en formato "arff". Estos archivos con este formato específico, no contienen únicamente los datos en bruto con los que vamos a trabajar, si no que incluyen meta-información de los propios datos, como el nombre y tipo de cada atributo, una descripción del origen de los datos, etc.

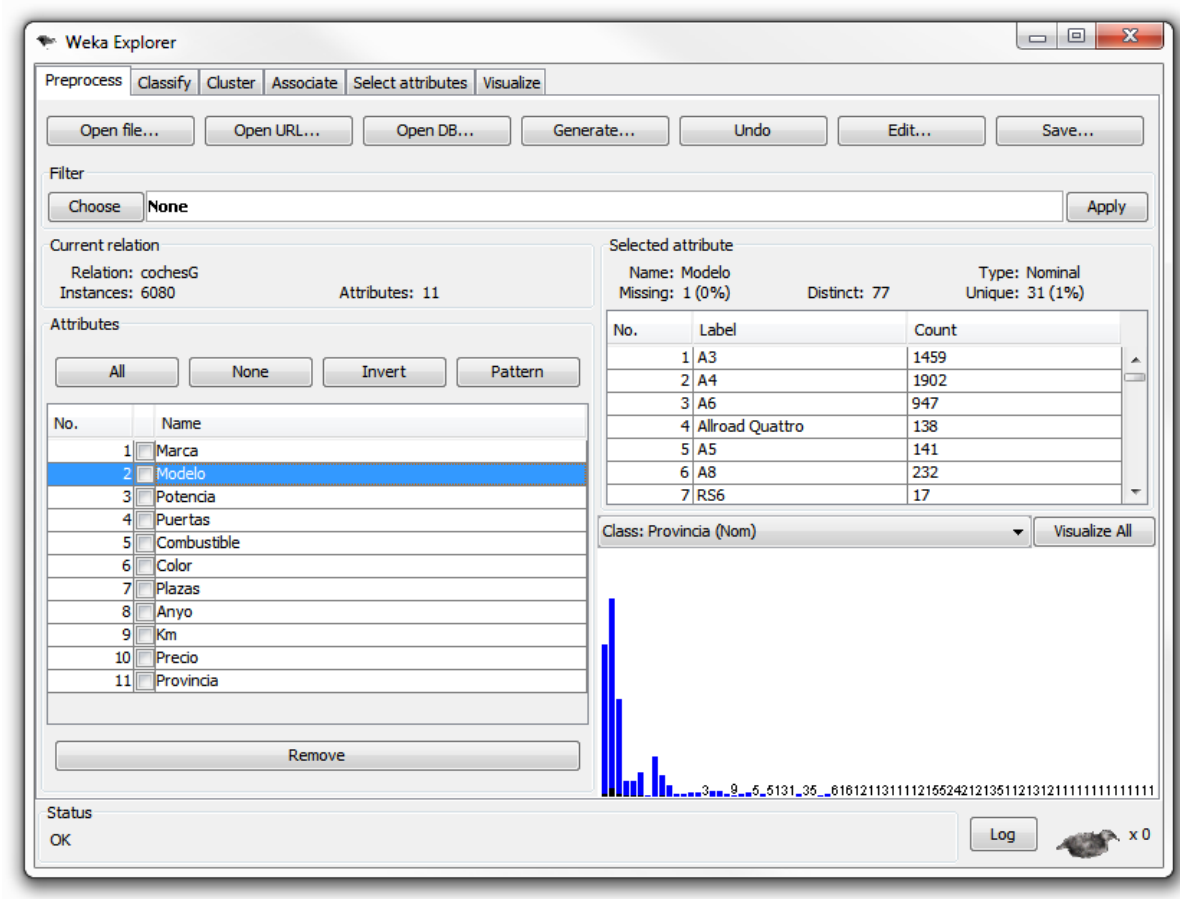

Figura 22. Tras importar los datos

Como podemos observar en la figura 21, Weka reconoce los 11 atributos que forman nuestro origen de datos. Automáticamente, asocia cada atributo de tipo nominal o numérico, según el contenido de los datos.

Adicionalmente, nos muestra información relevante a cada atributo, si vamos seleccionándolos uno a uno, nos muestra en los cuadros de la derecha varios datos:

- Nombre del atributo
- Valores perdidos
- Valores diferentes del atributo
- Tipo de atributo
- Valores que no se repiten
- Una tabla donde podemos ver cada valor las veces que se repite
- Un histograma que muestra una distribución de los valores para este determinado atributo

Antes de comenzar a aplicar ningún método de clasificación, analizamos un poco la información que tenemos preliminarmente. De este modo intentaremos eliminar aquellos atributos que consideremos que no van a ayudar al modelado del método de minería.

La marca del vehículo, es la misma para todos, en nuestro ejemplo, la marca elegida ha sido AUDI. Si todos los vehículos son de esta misma marca, este atributo no tiene sentido.

La provincia del vehículo. Si es cierto que existen variaciones en los precios entre las provincias de España, no obstante, no disponemos de suficientes datos y las diferencias no son tan apreciables como para poder considerar este atributo en el listado de atributos influyentes, por lo que lo eliminaremos también.

El modelo del vehículo, weka nos da información acerca de este atributo y los valores que toma en nuestros datos. Podemos observar que al obtenerse los datos de la web, y estos ser datos introducidos por los anunciantes, muchos de ellos no han sido muy precisos. Otros han introducido demasiada información en el modelo, hay modelos con muy pocas coincidencias, etc. De una muestra de 6080 vehículos, para que nuestro modelo sea lo más preciso posible, nos quedaremos con aquellas instancias de modelos que más registros presenten.

En este caso, nos quedaremos con los siguientes modelos, al lado podemos ver las repeticiones:

Modelo	Ocurrencias
A3	1459
A4	1902
A6	947
Allroad Quattro	138
A5	141
A8	232
TT	393
Q7	200
S3	105

Tabla 6. Modelos con más instancias

Para agilizar el proceso de filtrado de los datos, utilizaremos uno de los filtros de Weka. Seleccionaremos el selector de filtros, dentro de filtros sin supervisión abriremos los filtros de instancia, pues queremos filtrar dentro de un atributo. Seleccionaremos el atributo: RemoveFrequentValues.

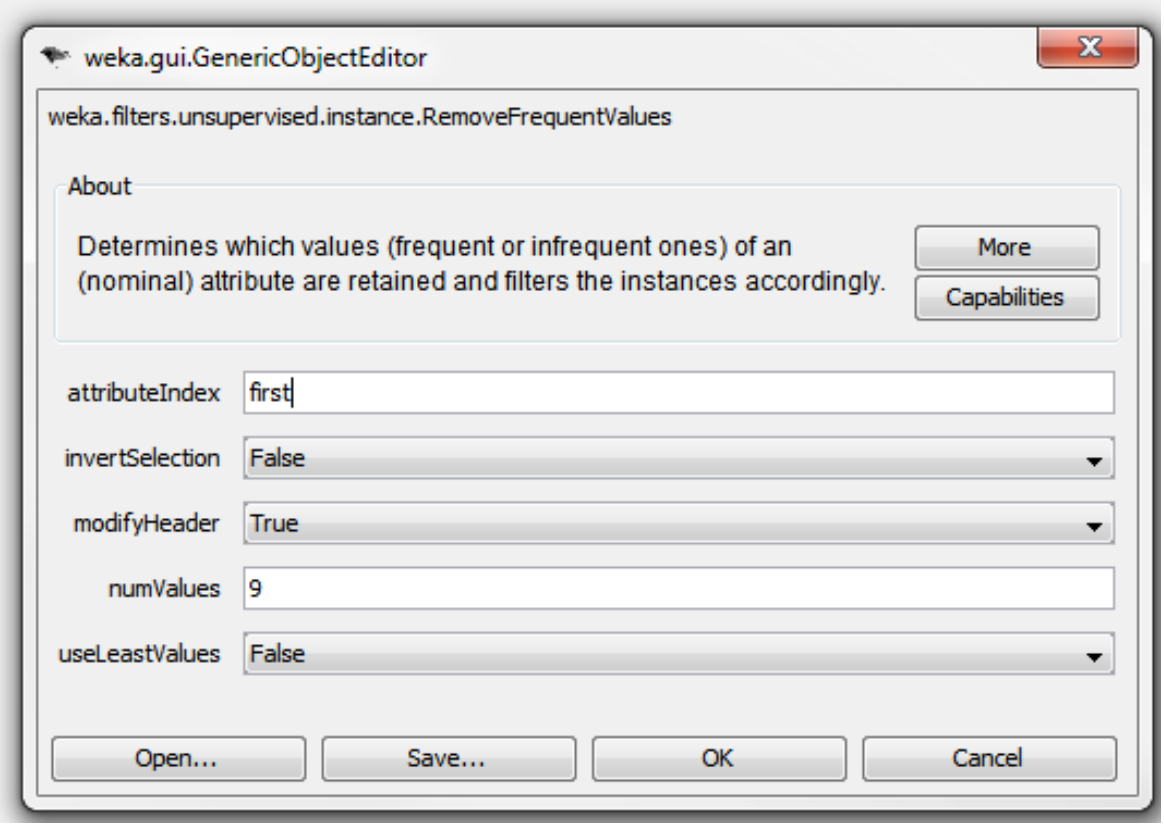

Figura 23. Configuración de filtro.

Este filtro determina con que valores (frecuentes o infrecuentes) de un atributo nominal nos vamos a quedar y filtra las instancias en concordancia.

- AtributeIndex. Indica el número del atributo que vamos a filtrar.
- InvertSelection. Para invertir la selección que realicemos.
- ModifyHeader. Elimina las referencias de las cabeceras para los valores excluidos.
- numValues. El número de valores con los que nos quedamos, tras analizar los datos, vemos que mayores de 100 coincidencias, solo existen 9 valores (los descritos en la tabla 6)
- useLeastValues. Utiliza los valores que menos se repiten, en lugar de los que más ocurrencias tienen.

Tras aplicar el filtro, podemos observar como los gráficos de la interfaz cambian.

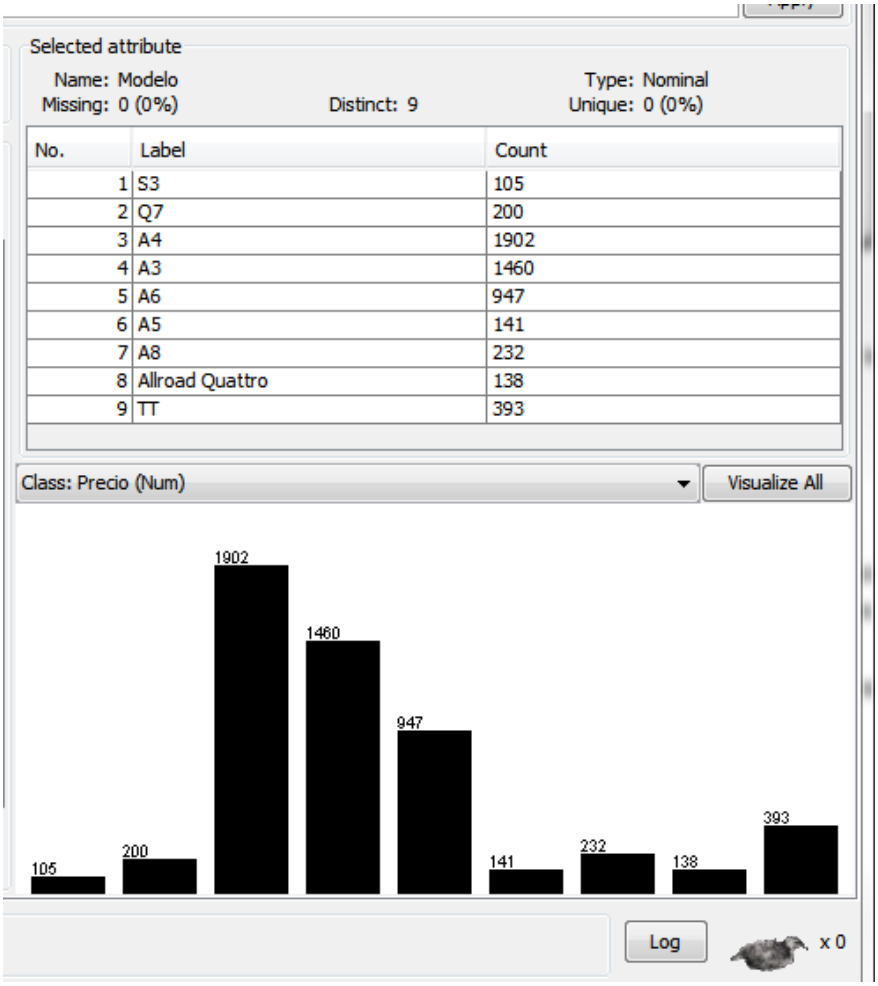

Figura 24. Modelos filtrados

Pasando al siguiente atributo, nos encontramos con que el campo de potencia, tiene muchísimos registros donde este campo no ha sido cumplimentado. Esto es debido, a que los anunciantes, a menudo colocaban la potencia del vehículo en el modelo y no en un campo aparte. Dada la relevancia que tiene este atributo en relación con el precio del vehñiculo, eliminaremos estos registros. Para eliminar todos estos registros con valores perdidos utilizaremos otro filtro de Weka.

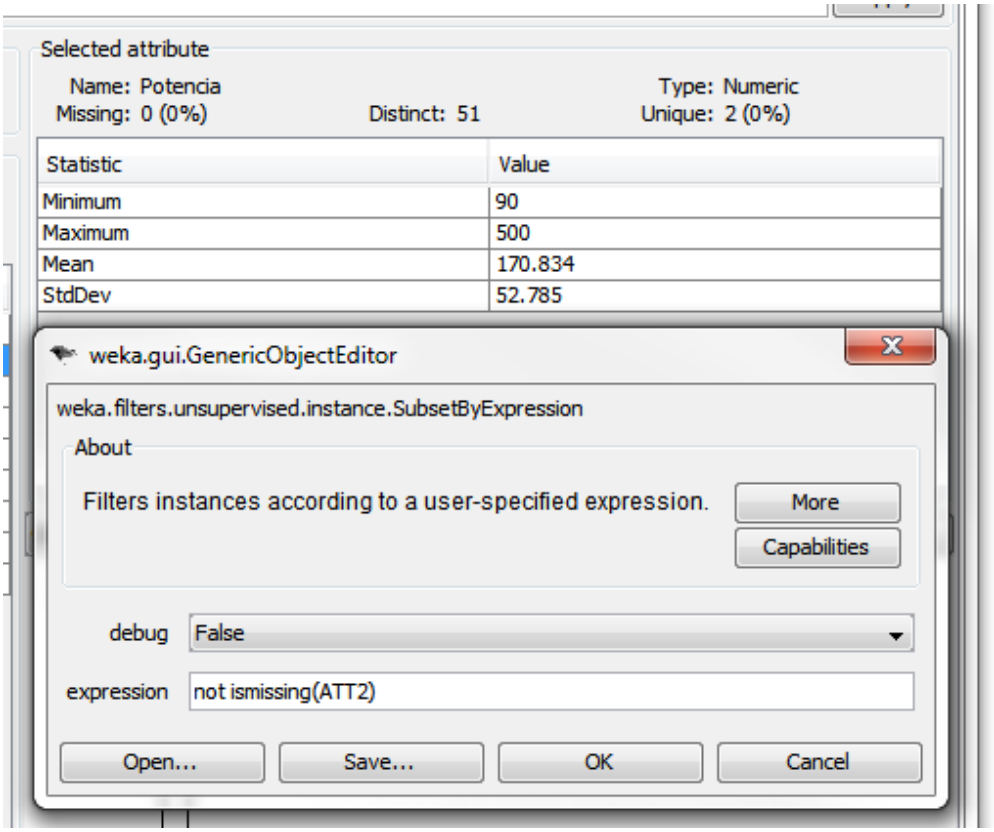

Figura 25. Filtro de valores vacíos

El filtro para instancias SubsetByExpression, nos permite filtrar las instancias según una expresión sencilla que el usuario puede elegir. En este caso, la expresión es: not ismissing(ATT2). Esto hará, que seleccionará aquellas instancias cuyo atributo número dos (la potencia) no esté vacío. Aplicaremos este mismo filtro para los kilómetros del vehículo y eliminar las instancias que no contengan el número de kilómetros recorridos.

El resto de atributos que quedan, tras los filtros aplicados, contienen valores correctos y no presentan problemas, exceptuando el color. Es cierto que el color del vehículo influye en el precio, concretamente, el tipo de pintura. Sin embargo, los anunciantes no han seguido unos patrones demasiado bien definidos a la hora de describir el color del vehículo, por lo que si seleccionamos el atributo COLOR, podemos observar que tiene unos 500 valores diferentes. En primer lugar, limpiaremos un poco el atributo, aunque acabemos desechando este atributo posteriormente.

En primer lugar, filtraremos del mismo modo que el atributo Modelo. Nos quedaremos con aquellas instancias con más coincidencias. Seleccionando 30 valores posibles máximos.

En segundo lugar, uniremos varios valores, pues weka discrimina entre mayúsculas y minúsculas para los valores, por lo que unificaremos todos los valores que coincidan mayúsculas con minúsculas de los valores que más se repitan.

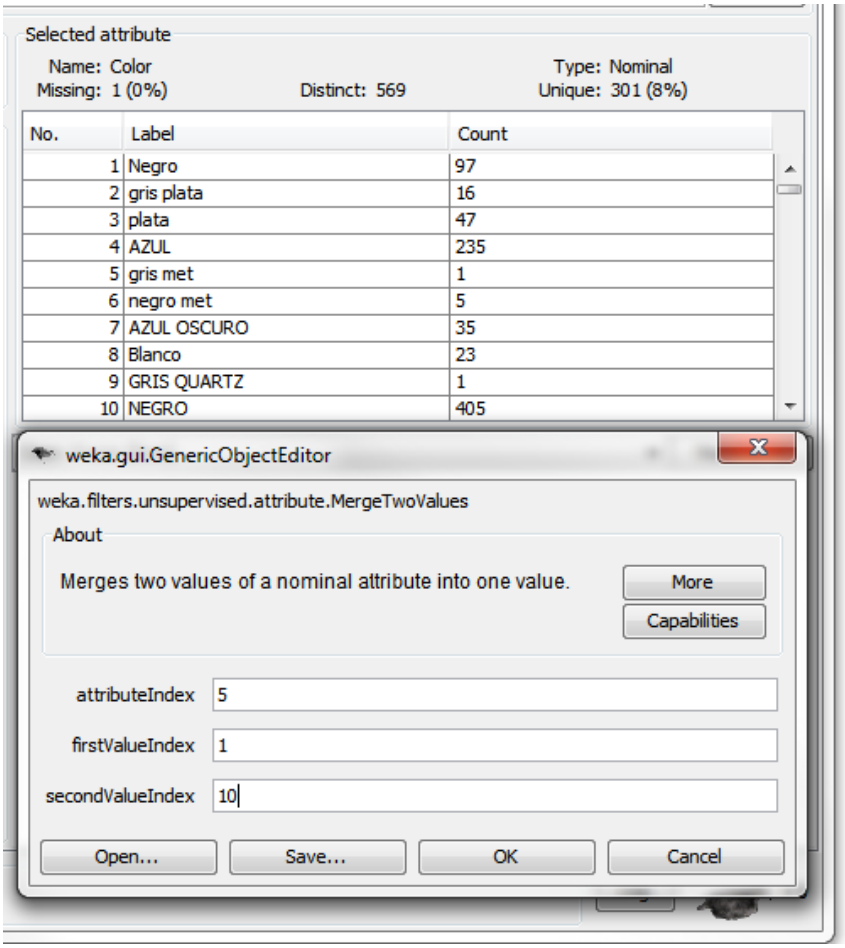

Figura 26. Filtro de color. Unión de instancias

En este filtro, seleccionamos la posición del atributo dentro de la lista de atributos. El color ocupa la posición 5. Después introducimos el valor del índice del valor que queremos combinar con el valor del índice del otro valor con el que queremos combinarlo. En este caso Negro es el número 1 y NEGRO el número 10. Procedemos de esta manera con todos los que sean el mismo color, pero introducido de forma diferente.

Tras aplicar estos filtros, el número de instancias se ve drásticamente reducido, por lo que decido no contemplar el color del vehículo, pues induciría a error en los cálculos la falta de normalización de los valores para este campo.

Con esto, tenemos los atributos bastante limpios y preparados para comenzar a aplicar los métodos de modelado de minería de datos. En el apartado siguiente analizaremos los tres modelos que mejor se adaptan a estos datos y a la finalidad de este proceso de minería, que no es otro que predecir el precio del vehículo según las características del mismo.

4.4.2.1 Importancia de atributos

Antes de comenzar a aplicar métodos de minería de datos, analizaremos las gráficas de relación obtenidas para cada uno de los atributos que hemos seleccionado como válidos para comprobar que efectivamente todos y cada uno de ellos son influyentes para el precio del vehículo.

En la pestaña de Visualize, podemos encontrar gráficas cruzadas de cada atributo con el resto de atributos.

Las gráficas más interesantes y que más claramente podemos analizar como varían claramente el precio final del vehículo según los valores que adopten los atributos son las siguientes:

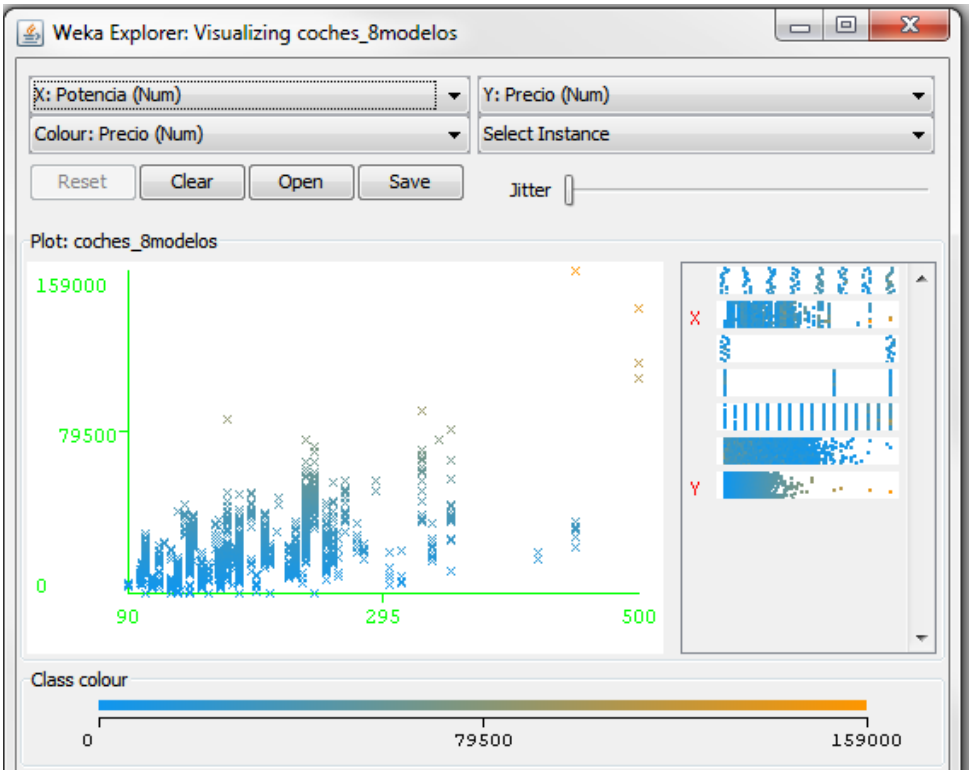

Figura 27. Potencia/Precio

Vemos claramente como a menor potencia, el precio final del vehículo es menor. Esto lo podemos observar en la acumulación de puntos al principio de la gráfica, donde la potencia y el precio son menores.

Del mismo modo ocurre con los kilómetros y el precio. A menor número de kilómetros del vehículo, el precio es mayor.

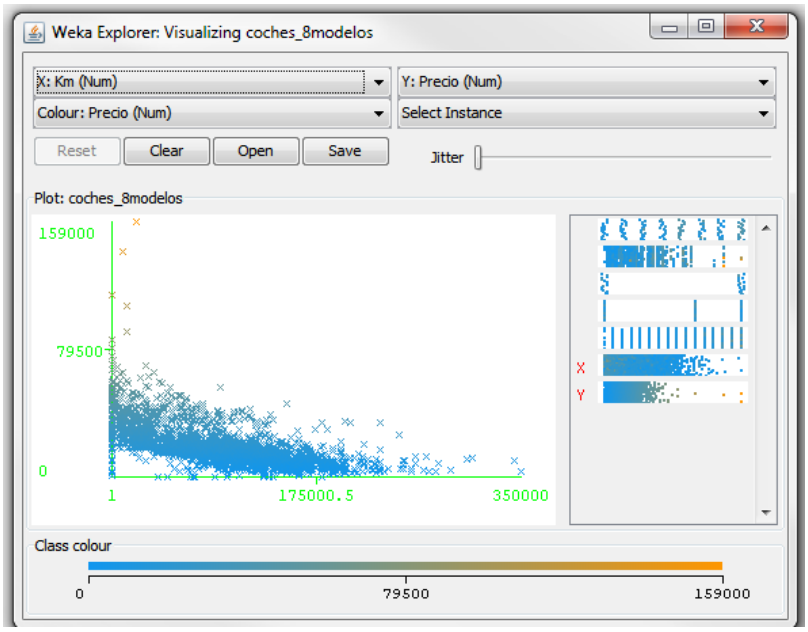

Figura 28. Km/Precio

También podemos observar, como cuanto más nuevo es un vehículo, mayor precio tiene. La novedad se paga.

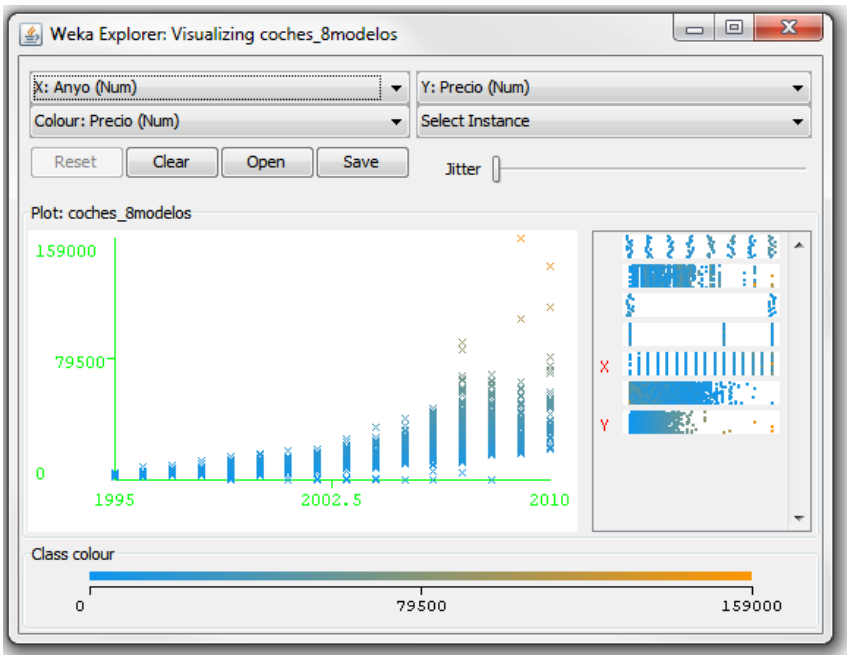

Figura 29. Año/Precio

4.4.3 Análisis de modelos

A continuación, pasaremos a analizar y exponer los resultados de tres algoritmos de los múltiples algoritmos implementados en Weka. La selección de estos métodos ha estado basada en los resultados que deseamos obtener, Weka permite aplicar unos métodos u otros en concordancia con el propósito del estudio. En este caso, el propósito era el de predecir el precio de un vehículo, por lo que, bajo la recomendación del director de este proyecto y varias pruebas con otros algoritmos, expondré los resultados obtenidos con estos tres en concreto únicamente.

4.4.3.1 IBK

A pesar de que este algoritmo no crea ningún tipo de modelo o de reglas de decisión, merece la pena aplicarlo a nuestro conjunto de datos y observar los resultados. Este algoritmo es de la familia de algoritmos incluidos en "lazy learning". Este algoritmo se basa en instancias, por lo que únicamente almacena los datos presentados. Cuando al ejecutarlo se encuentra una nueva instancia, se devuelve desde memoria el conjunto de instancias similares relacionadas y usado para clasificar la instancia en concreto. Cada vez que se encuentra una nueva instancia, el algoritmo calcula su relación con el resto de ejemplos almacenados previamente con el fin de asignar un valor de la función objetivo para esta instancia encontrada.

El concepto principal que fundamenta este algoritmo, es que cada instancia encontrada se va a clasificar en la clase más frecuente a la que pertenezcan sus K vecinos más cercanos. Es por esto que este algoritmo también es conocido como el método K-NN. K Nearest Neighbours.

Ahora, pasaremos a aplicar el método a nuestros datos. Para ello, nos iremos a la pestaña de Classify de Weka.

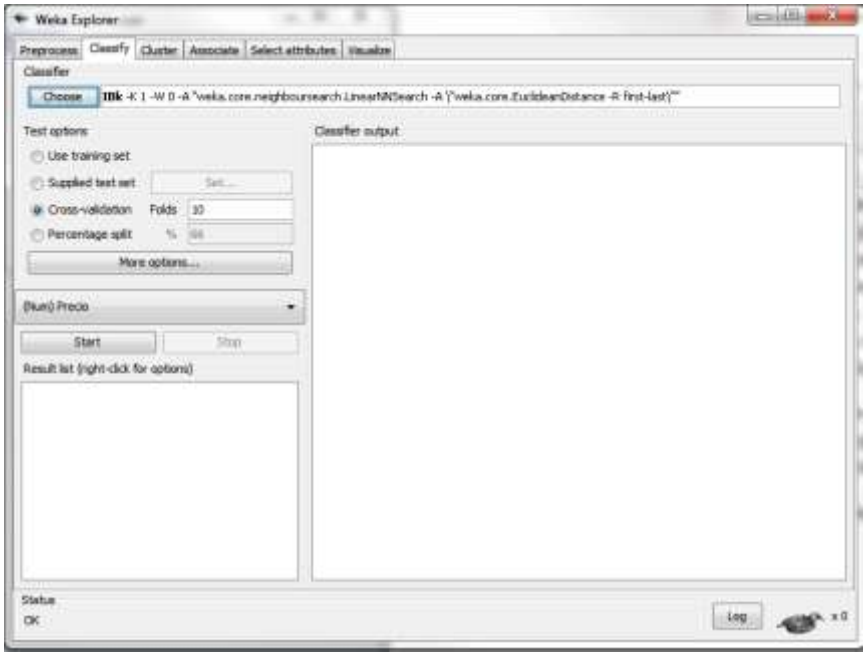

Figura 30. Classify

Aquí, seleccionaremos el clasificador pulsando sobre Choose. Se nos abre una ventana de exploración de los métodos, donde seleccionamos el método IBK, que se encuentra dentro de la carpeta de métodos “Lazy”.

Seleccionando el atributo Precio, como atributo a evaluar, analizamos las opciones del algoritmo.

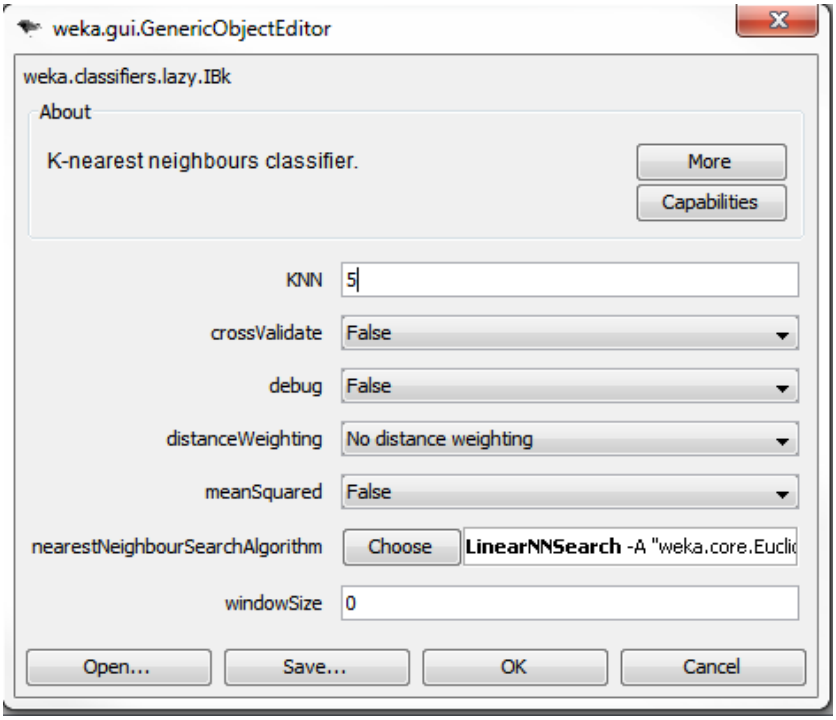

Figura 31. Configuración IBK

En la ventana de configuración del método, podemos seleccionar varias opciones para el algoritmo, para este caso, modificaremos únicamente el número KNN, que es el número de “vecinos”, seleccionando 5. El sistema nos propone 1 por defecto, no obstante, con esta profundidad, el vecino consultado será la propia instancia por lo que el valor decidido será el propio valor de la instancia.

Con esta configuración ejecutamos el método obteniendo los siguientes resultados:

```
=== Summary ===

Correlation coefficient                  0.9268
Mean absolute error                   2275.3392
Root mean squared error               4645.8015
Relative absolute error                 26.4735 %
Root relative squared error             37.6387 %
Total Number of Instances             3543
```

Podemos ver en el resumen de la aplicación de este método, que los resultados que aporta son bastante buenos a simple vista. El coeficiente de correlación, que mide la correlación estadística entre los datos predichos y los datos reales, es bastante bueno (1 es el 100% y es el máximo).

También podemos observar que el error absoluto medio, no es muy elevado, 2275 (que equivaldría a ±2275€ en el precio estimado del vehículo.

No obstante, este método, no crea un modelo para poder implementarlo ni una serie de reglas a aplicar, tan sólo clasifica las instancias.

4.4.3.2 Regresión lineal

Aplicaremos ahora el método de regresión lineal implementado en Weka. Como ya hemos hablado anteriormente en este trabajo, este método intentará construir una función matemática para calcular el valor a predecir. Teniendo en cuenta como afectan en mayor o menor medida el valor de los atributos para el precio del vehículo.

En esta ocasión, ejecutamos el método sin modificar las opciones por defecto y utilizando una validación cruzada de 10 pliegues o "folds".

Tras unos segundos, Weka nos muestra el modelo construido a partir de los datos y el resumen de resultados.

```
Linear Regression Model

Precio =

   1798.9986 * Modelo=A4,A6,TT,Allroad Quattro,A8,A5,Q7 +

   1239.8777 * Modelo=A6,TT,Allroad Quattro,A8,A5,Q7 +

  -2606.1192 * Modelo=TT,Allroad Quattro,A8,A5,Q7 +

   3577.2125 * Modelo=Allroad Quattro,A8,A5,Q7 +

   3370.2118 * Modelo=A8,A5,Q7 +

   1527.7767 * Modelo=A5,Q7 +

   7567.8713 * Modelo=Q7 +

     76.9227 * Potencia +

   3503.4918 * Combustible=Diesel +

  -1825.0759 * Plazas +

   1319.2098 * Anyo +

     -0.0763 * Km +

-2628316.4888
```

Se trata de una función bastante simple de implementar, donde dependiendo del valor de cada atributo, se van aplicando diferentes funciones matemáticas a los datos para predecir el precio final del vehículo.

La fiabilidad de esta función la podemos comprobar con los resultados calculados en Weka.

```
=== Summary ===

Correlation coefficient                  0.8789
Mean absolute error                   3409.7753
Root mean squared error               5886.3599
Relative absolute error                 39.6726 %
Root relative squared error             47.6892 %
Total Number of Instances             3543
```

Observamos que se obtiene un coeficiente de correlación bastante bueno, no obstante la media absoluta de error queda algo elevada, así como la media cuadrática. Esto supondría un error de casi 3500€ en la predicción del precio del vehículo.

4.4.3.3 M5P

Por último aplicaremos el algoritmo M5P a nuestro conjunto de datos. Este algoritmo es una reconstrucción del algoritmo de Quinlan M5. Este algoritmo combina un árbol de decisión normal con funciones de regresión lineal en los nodos.

Primero, utiliza un algoritmo de árbol de decisión para construir un árbol, sin embargo, en vez de maximizar la información ganada en cada nodo interno, se utiliza un criterio de división que minimiza la variación interna de los subconjuntos para los valores de clase de cada rama. EL proceso de división del M5P se detiene si los valores de clase de todas las instancias que alcanzan un nodo varían ligeramente, o tan solo permanecen algunas instancias.

En segundo lugar, el árbol es recortado de nuevo desde cada hoja. Al recortar un nodo interno, se convierte en una hoja con un plano de regresión.

Después, para evitar discontinuidades entre los sub-arboles, se aplica un procedimiento que combina el modelo de predicción de hojas con cada nodo por todo el trayecto de vuelta a la raíz, haciendo más preciso cada uno de estos nodos al combinarlo con el valor predecido por el modelo lineal para dicho nodo.

En resumen, se trata de un algoritmo que combina los árboles de decisión con las funciones de regresión lineal. Para esto, se van creando "reglas", que se deberán ir aplicando consecuentemente según si los datos cumplen una u otra condición para tener que cumplir una regla u otra.

Para aplicar este algoritmo a nuestro conjunto de datos, lo seleccionaremos desde el conjunto de algoritmos almacenados en el apartado de "Rules", que comprende una serie de algoritmos basados en reglas de decisión.

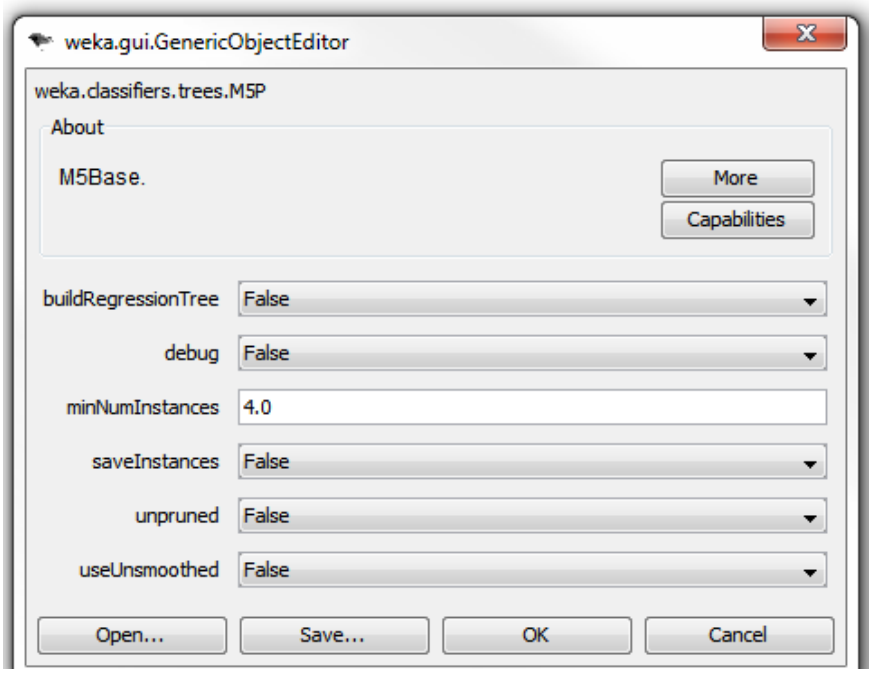

Figura 32. Configuración M5P

Utilizaremos las opciones por defecto del método.

Tras aplicar el algoritmo con una validación cruzada de diez pliegues obtenemos los siguientes resultados.

Se obtienen 25 reglas y los siguientes valores:

```
=== Summary ===

Correlation coefficient                  0.9517
Mean absolute error                   2289.7282
Root mean squared error               3790.9974
Relative absolute error                 26.6409 %
Root relative squared error             30.7133 %
Total Number of Instances             3543
```

Podemos observar que el coeficiente de correlación es muy bueno, alcanzando un 0.95, así como el error medio absoluto, de apenas 2300€.

Se genera el siguiente árbol de decisión:

```
M5 pruned model tree:
(using smoothed linear models)
Anyo <= 2005.5 :
|   Anyo <= 2002.5 :
|   |   Potencia <= 167.5 : LM1 (474/14.839%)
|   |   Potencia >  167.5 : LM2 (263/19.921%)
|   Anyo >  2002.5 :
|   |   Potencia <= 202 : LM3 (814/19.938%)
|   |   Potencia >  202 :
|   |   |   Modelo=A8,A5,Q7 <= 0.5 :
|   |   |   |   Km <= 128500 : LM4 (125/21.606%)
|   |   |   |   Km >  128500 :
|   |   |   |   |   Km <= 171500 :
|   |   |   |   |   |   Combustible=Diesel <= 0.5 :
|   |   |   |   |   |   |   Potencia <= 252.5 : LM5 (5/8.24%)
|   |   |   |   |   |   |   Potencia >  252.5 : LM6 (4/2.244%)
|   |   |   |   |   |   Combustible=Diesel >  0.5 : LM7 (22/8.565%)
|   |   |   |   |   Km >  171500 :
|   |   |   |   |   |   Potencia <= 212.5 : LM8 (5/0%)
|   |   |   |   |   |   Potencia >  212.5 : LM9 (6/12.778%)
|   |   |   Modelo=A8,A5,Q7 >  0.5 : LM10 (48/29.332%)
Anyo >  2005.5 :
|   Potencia <= 166.5 :
|   |   Anyo <= 2007.5 : LM11 (617/17.536%)
|   |   Anyo >  2007.5 :
|   |   |   Potencia <= 130.5 : LM12 (89/12.835%)
|   |   |   Potencia >  130.5 : LM13 (273/28.586%)
|   Potencia >  166.5 :
|   |   Km <= 22950 :
|   |   |   Plazas <= 4.5 :
|   |   |   |   Modelo=Allroad Quattro,A8,A5,Q7 <= 0.5 : LM14 (39/35.719%)
|   |   |   |   Modelo=Allroad Quattro,A8,A5,Q7 >  0.5 :
|   |   |   |   |   Km <= 9400 :
|   |   |   |   |   |   Potencia <= 185 : LM15 (16/25.436%)
|   |   |   |   |   |   Potencia >  185 : LM16 (14/25.455%)
|   |   |   |   |   Km >  9400 : LM17 (21/21.921%)
|   |   |   Plazas >  4.5 :
|   |   |   |   Modelo=TT,Allroad Quattro,A8,A5,Q7 <= 0.5 :
|   |   |   |   |   Km <= 16 :
|   |   |   |   |   |   Potencia <= 175 : LM18 (4/0%)
|   |   |   |   |   |   Potencia >  175 : LM19 (11/25.429%)
|   |   |   |   |   Km >  16 :
|   |   |   |   |   |   Potencia <= 229 : LM20 (23/33.79%)
|   |   |   |   |   |   Potencia >  229 : LM21 (35/35.07%)
|   |   |   |   Modelo=TT,Allroad Quattro,A8,A5,Q7 >  0.5 :
|   |   |   |   |   Potencia <= 283 : LM22 (28/55.515%)
|   |   |   |   |   Potencia >  283 :
|   |   |   |   |   |   Potencia <= 400 : LM23 (10/71.916%)
|   |   |   |   |   |   Potencia >  400 : LM24 (5/92.452%)
|   |   Km >  22950 : LM25 (592/38.383%)
```

Las reglas generadas son las siguientes:

LM num: 1
Precio =
619.8268 * Modelo=A4,A6,TT,Allroad Quattro,A8,A5,Q7
+ 25.2089 * Modelo=A6,TT,Allroad Quattro,A8,A5,Q7
+ 3331.006 * Modelo=TT,Allroad Quattro,A8,A5,Q7
+ 170.4255 * Modelo=Allroad Quattro,A8,A5,Q7
+ 6.937 * Modelo=A8,A5,Q7

+ 12.8673 * Modelo=A5,Q7
+ 63.7384 * Modelo=Q7
+ 21.7723 * Potencia
+ 1712.5738 * Combustible=Diesel
- 96.7428 * Plazas
+ 818.6264 * Anyo
- 0.0115 * Km
- 1633227.9317

LM num: 2
Precio =
68.4267 * Modelo=A4,A6,TT,Allroad Quattro,A8,A5,Q7
+ 25.2089 * Modelo=A6,TT,Allroad Quattro,A8,A5,Q7
- 41.3492 * Modelo=TT,Allroad Quattro,A8,A5,Q7
+ 1976.8033 * Modelo=Allroad Quattro,A8,A5,Q7
- 1386.475 * Modelo=A8,A5,Q7
+ 12.8673 * Modelo=A5,Q7
+ 63.7384 * Modelo=Q7
+ 29.2308 * Potencia
+ 1742.2604 * Combustible=Diesel
- 1155.0164 * Plazas
+ 1061.2004 * Anyo
- 0.0168 * Km
- 2112046.8394

LM num: 3
Precio =
427.1094 * Modelo=A4,A6,TT,Allroad Quattro,A8,A5,Q7
+ 1143.2794 * Modelo=A6,TT,Allroad Quattro,A8,A5,Q7
- 5773.4359 * Modelo=TT,Allroad Quattro,A8,A5,Q7
+ 7281.6358 * Modelo=Allroad Quattro,A8,A5,Q7
+ 104.2165 * Modelo=A8,A5,Q7
+ 12.8673 * Modelo=A5,Q7
+ 63.7384 * Modelo=Q7
+ 42.3595 * Potencia
+ 1713.8141 * Combustible=Diesel
- 6024.4901 * Plazas
+ 2167.1857 * Anyo
- 0.0211 * Km
- 4305739.346

LM num: 4
Precio =
58.7841 * Modelo=A4,A6,TT,Allroad Quattro,A8,A5,Q7
+ 485.5095 * Modelo=A6,TT,Allroad Quattro,A8,A5,Q7
- 444.9914 * Modelo=TT,Allroad Quattro,A8,A5,Q7
+ 556.9712 * Modelo=Allroad Quattro,A8,A5,Q7
+ 655.2148 * Modelo=A8,A5,Q7
+ 12.8673 * Modelo=A5,Q7
+ 63.7384 * Modelo=Q7
+ 73.5797 * Potencia
+ 4938.4653 * Combustible=Diesel
- 5518.9883 * Plazas
+ 1488.0659 * Anyo
- 0.0481 * Km
- 2952540.404

LM num: 5

Precio =

58.7841 * Modelo=A4,A6,TT,Allroad Quattro,A8,A5,Q7
+ 2542.7051 * Modelo=A6,TT,Allroad Quattro,A8,A5,Q7
- 444.9914 * Modelo=TT,Allroad Quattro,A8,A5,Q7
+ 556.9712 * Modelo=Allroad Quattro,A8,A5,Q7
+ 655.2148 * Modelo=A8,A5,Q7
+ 12.8673 * Modelo=A5,Q7
+ 63.7384 * Modelo=Q7
+ 38.7596 * Potencia
+ 4305.9549 * Combustible=Diesel
- 2095.8369 * Plazas
+ 9.4485 * Anyo
- 0.0505 * Km
+ 357.8055

LM num: 6

Precio =

58.7841 * Modelo=A4,A6,TT,Allroad Quattro,A8,A5,Q7
+ 2542.7051 * Modelo=A6,TT,Allroad Quattro,A8,A5,Q7
- 444.9914 * Modelo=TT,Allroad Quattro,A8,A5,Q7
+ 556.9712 * Modelo=Allroad Quattro,A8,A5,Q7
+ 655.2148 * Modelo=A8,A5,Q7
+ 12.8673 * Modelo=A5,Q7
+ 63.7384 * Modelo=Q7
+ 38.7596 * Potencia
+ 4305.9549 * Combustible=Diesel
- 2095.8369 * Plazas
+ 9.4485 * Anyo
- 0.0505 * Km
+ 473.2346

LM num: 7

Precio =

58.7841 * Modelo=A4,A6,TT,Allroad Quattro,A8,A5,Q7
+ 3358.787 * Modelo=A6,TT,Allroad Quattro,A8,A5,Q7
- 444.9914 * Modelo=TT,Allroad Quattro,A8,A5,Q7
+ 556.9712 * Modelo=Allroad Quattro,A8,A5,Q7
+ 655.2148 * Modelo=A8,A5,Q7
+ 12.8673 * Modelo=A5,Q7
+ 63.7384 * Modelo=Q7
+ 38.7596 * Potencia
+ 3825.0595 * Combustible=Diesel
- 2095.8369 * Plazas
+ 9.4485 * Anyo
- 0.0659 * Km
+ 3377.9828

LM num: 8

Precio =

58.7841 * Modelo=A4,A6,TT,Allroad Quattro,A8,A5,Q7
+ 4908.8373 * Modelo=A6,TT,Allroad Quattro,A8,A5,Q7
- 444.9914 * Modelo=TT,Allroad Quattro,A8,A5,Q7
+ 556.9712 * Modelo=Allroad Quattro,A8,A5,Q7
+ 655.2148 * Modelo=A8,A5,Q7
+ 12.8673 * Modelo=A5,Q7
+ 63.7384 * Modelo=Q7
+ 51.4874 * Potencia
+ 3944.3627 * Combustible=Diesel

- 2095.8369 * Plazas
- 311.1037 * Anyo
- 0.0538 * Km
+ 637652.3366

LM num: 9
Precio =
58.7841 * Modelo=A4,A6,TT,Allroad Quattro,A8,A5,Q7
+ 5097.6217 * Modelo=A6,TT,Allroad Quattro,A8,A5,Q7
- 444.9914 * Modelo=TT,Allroad Quattro,A8,A5,Q7
+ 556.9712 * Modelo=Allroad Quattro,A8,A5,Q7
+ 655.2148 * Modelo=A8,A5,Q7
+ 12.8673 * Modelo=A5,Q7
+ 63.7384 * Modelo=Q7
+ 51.4874 * Potencia
+ 3944.3627 * Combustible=Diesel
- 2095.8369 * Plazas
- 311.1037 * Anyo
- 0.0538 * Km
+ 638480.2936

LM num: 10
Precio =
58.7841 * Modelo=A4,A6,TT,Allroad Quattro,A8,A5,Q7
+ 737.6342 * Modelo=A6,TT,Allroad Quattro,A8,A5,Q7
- 444.9914 * Modelo=TT,Allroad Quattro,A8,A5,Q7
+ 556.9712 * Modelo=Allroad Quattro,A8,A5,Q7
+ 1377.7596 * Modelo=A8,A5,Q7
+ 12.8673 * Modelo=A5,Q7
+ 63.7384 * Modelo=Q7
+ 8.4822 * Potencia
+ 915.2108 * Combustible=Diesel
- 1545.9805 * Plazas
+ 2734.0827 * Anyo
- 0.0415 * Km
- 5447947.7179

LM num: 11
Precio =
589.2836 * Modelo=A4,A6,TT,Allroad Quattro,A8,A5,Q7
+ 3232.4344 * Modelo=A6,TT,Allroad Quattro,A8,A5,Q7
- 91.2803 * Modelo=TT,Allroad Quattro,A8,A5,Q7
+ 122.4451 * Modelo=Allroad Quattro,A8,A5,Q7
+ 145.0239 * Modelo=A8,A5,Q7
- 109.0266 * Modelo=A5,Q7
+ 179.4017 * Modelo=Q7
+ 51.3165 * Potencia
+ 1779.2208 * Combustible=Diesel
- 102.2572 * Plazas
+ 1407.6563 * Anyo
- 0.0425 * Km
- 2812784.3645

LM num: 12
Precio =
775.7993 * Modelo=A4,A6,TT,Allroad Quattro,A8,A5,Q7
+ 474.1224 * Modelo=A6,TT,Allroad Quattro,A8,A5,Q7
- 91.2803 * Modelo=TT,Allroad Quattro,A8,A5,Q7

+ 122.4451 * Modelo=Allroad Quattro,A8,A5,Q7
+ 145.0239 * Modelo=A8,A5,Q7
- 109.0266 * Modelo=A5,Q7
+ 179.4017 * Modelo=Q7
+ 75.6836 * Potencia
+ 211.2892 * Combustible=Diesel
- 5484.3679 * Plazas
+ 246.596 * Anyo
- 0.0584 * Km
- 455167.0982

LM num: 13
Precio =
5212.5011 * Modelo=A4,A6,TT,Allroad Quattro,A8,A5,Q7
+ 270.6765 * Modelo=A6,TT,Allroad Quattro,A8,A5,Q7
- 91.2803 * Modelo=TT,Allroad Quattro,A8,A5,Q7
+ 122.4451 * Modelo=Allroad Quattro,A8,A5,Q7
+ 145.0239 * Modelo=A8,A5,Q7
- 109.0266 * Modelo=A5,Q7
+ 179.4017 * Modelo=Q7
- 115.7896 * Potencia
+ 211.2892 * Combustible=Diesel
- 3901.1996 * Plazas
+ 720.7051 * Anyo
- 0.1061 * Km
- 1385577.7525

LM num: 14
Precio =
306.1903 * Modelo=A4,A6,TT,Allroad Quattro,A8,A5,Q7
+ 31.9754 * Modelo=A6,TT,Allroad Quattro,A8,A5,Q7
- 1642.7329 * Modelo=TT,Allroad Quattro,A8,A5,Q7
+ 6113.887 * Modelo=Allroad Quattro,A8,A5,Q7
+ 585.0925 * Modelo=A8,A5,Q7
- 2507.5773 * Modelo=A5,Q7
+ 821.5292 * Modelo=Q7
+ 49.4079 * Potencia
+ 477.4271 * Combustible=Diesel
- 1271.8255 * Plazas
+ 317.1283 * Anyo
- 0.2274 * Km
- 606405.9466

LM num: 15
Precio =
306.1903 * Modelo=A4,A6,TT,Allroad Quattro,A8,A5,Q7
+ 31.9754 * Modelo=A6,TT,Allroad Quattro,A8,A5,Q7
- 1642.7329 * Modelo=TT,Allroad Quattro,A8,A5,Q7
+ 5734.6301 * Modelo=Allroad Quattro,A8,A5,Q7
+ 585.0925 * Modelo=A8,A5,Q7
- 2507.5773 * Modelo=A5,Q7
+ 821.5292 * Modelo=Q7
- 4.3614 * Potencia
+ 477.4271 * Combustible=Diesel
- 429.8984 * Plazas
+ 1409.8011 * Anyo
+ 0.3239 * Km
- 2793858.9706

LM num: 16
Precio =
306.1903 * Modelo=A4,A6,TT,Allroad Quattro,A8,A5,Q7
+ 31.9754 * Modelo=A6,TT,Allroad Quattro,A8,A5,Q7
- 1642.7329 * Modelo=TT,Allroad Quattro,A8,A5,Q7
+ 5734.6301 * Modelo=Allroad Quattro,A8,A5,Q7
+ 585.0925 * Modelo=A8,A5,Q7
- 2507.5773 * Modelo=A5,Q7
+ 821.5292 * Modelo=Q7
+ 91.0233 * Potencia
+ 477.4271 * Combustible=Diesel
- 429.8984 * Plazas
+ 1457.0615 * Anyo
+ 0.1603 * Km
- 2903222.4916

LM num: 17
Precio =
306.1903 * Modelo=A4,A6,TT,Allroad Quattro,A8,A5,Q7
+ 31.9754 * Modelo=A6,TT,Allroad Quattro,A8,A5,Q7
- 1642.7329 * Modelo=TT,Allroad Quattro,A8,A5,Q7
+ 5734.6301 * Modelo=Allroad Quattro,A8,A5,Q7
+ 585.0925 * Modelo=A8,A5,Q7
- 2507.5773 * Modelo=A5,Q7
+ 821.5292 * Modelo=Q7
+ 69.514 * Potencia
+ 477.4271 * Combustible=Diesel
- 429.8984 * Plazas
+ 826.374 * Anyo
- 0.2571 * Km
- 1630983.2738

LM num: 18
Precio =
306.1903 * Modelo=A4,A6,TT,Allroad Quattro,A8,A5,Q7
- 6648.672 * Modelo=A6,TT,Allroad Quattro,A8,A5,Q7
+ 735.8111 * Modelo=TT,Allroad Quattro,A8,A5,Q7
+ 3346.9879 * Modelo=Allroad Quattro,A8,A5,Q7
+ 585.0925 * Modelo=A8,A5,Q7
- 3610.5277 * Modelo=A5,Q7
+ 821.5292 * Modelo=Q7
+ 63.2711 * Potencia
+ 477.4271 * Combustible=Diesel
- 164.9591 * Plazas
- 3190.2296 * Anyo
- 0.3285 * Km
+ 6447373.158

LM num: 19
Precio =
306.1903 * Modelo=A4,A6,TT,Allroad Quattro,A8,A5,Q7
- 3072.0982 * Modelo=A6,TT,Allroad Quattro,A8,A5,Q7
+ 735.8111 * Modelo=TT,Allroad Quattro,A8,A5,Q7
+ 3346.9879 * Modelo=Allroad Quattro,A8,A5,Q7
+ 585.0925 * Modelo=A8,A5,Q7
- 3610.5277 * Modelo=A5,Q7
+ 821.5292 * Modelo=Q7

+ 74.8974 * Potencia
+ 477.4271 * Combustible=Diesel
- 164.9591 * Plazas
- 1022.1724 * Anyo
- 0.3285 * Km
+ 2084928.4375

LM num: 20
Precio =
306.1903 * Modelo=A4,A6,TT,Allroad Quattro,A8,A5,Q7
+ 1420.8985 * Modelo=A6,TT,Allroad Quattro,A8,A5,Q7
+ 735.8111 * Modelo=TT,Allroad Quattro,A8,A5,Q7
+ 3346.9879 * Modelo=Allroad Quattro,A8,A5,Q7
+ 585.0925 * Modelo=A8,A5,Q7
- 3610.5277 * Modelo=A5,Q7
+ 821.5292 * Modelo=Q7
+ 89.195 * Potencia
+ 477.4271 * Combustible=Diesel
- 164.9591 * Plazas
+ 222.831 * Anyo
- 0.3202 * Km
- 424981.8687

LM num: 21
Precio =
306.1903 * Modelo=A4,A6,TT,Allroad Quattro,A8,A5,Q7
+ 4786.0218 * Modelo=A6,TT,Allroad Quattro,A8,A5,Q7
+ 735.8111 * Modelo=TT,Allroad Quattro,A8,A5,Q7
+ 3346.9879 * Modelo=Allroad Quattro,A8,A5,Q7
+ 585.0925 * Modelo=A8,A5,Q7
- 3610.5277 * Modelo=A5,Q7
+ 821.5292 * Modelo=Q7
+ 123.1271 * Potencia
+ 477.4271 * Combustible=Diesel
- 164.9591 * Plazas
+ 158.4115 * Anyo
- 0.3352 * Km
- 303190.1064

LM num: 22
Precio =
306.1903 * Modelo=A4,A6,TT,Allroad Quattro,A8,A5,Q7
+ 31.9754 * Modelo=A6,TT,Allroad Quattro,A8,A5,Q7
+ 1837.7317 * Modelo=TT,Allroad Quattro,A8,A5,Q7
+ 3346.9879 * Modelo=Allroad Quattro,A8,A5,Q7
+ 585.0925 * Modelo=A8,A5,Q7
- 7020.3986 * Modelo=A5,Q7
+ 821.5292 * Modelo=Q7
+ 135.2251 * Potencia
+ 477.4271 * Combustible=Diesel
- 164.9591 * Plazas
+ 1422.8013 * Anyo
- 0.1763 * Km
- 2835472.444

LM num: 23
Precio =
306.1903 * Modelo=A4,A6,TT,Allroad Quattro,A8,A5,Q7

+ 31.9754 * Modelo=A6,TT,Allroad Quattro,A8,A5,Q7
+ 1837.7317 * Modelo=TT,Allroad Quattro,A8,A5,Q7
+ 3346.9879 * Modelo=Allroad Quattro,A8,A5,Q7
+ 585.0925 * Modelo=A8,A5,Q7
- 11085.7117 * Modelo=A5,Q7
+ 821.5292 * Modelo=Q7
+ 224.9095 * Potencia
+ 477.4271 * Combustible=Diesel
- 164.9591 * Plazas
+ 1901.9263 * Anyo
+ 0.251 * Km
- 3821304.7025

LM num: 24
Precio =
306.1903 * Modelo=A4,A6,TT,Allroad Quattro,A8,A5,Q7
+ 31.9754 * Modelo=A6,TT,Allroad Quattro,A8,A5,Q7
+ 1837.7317 * Modelo=TT,Allroad Quattro,A8,A5,Q7
+ 3346.9879 * Modelo=Allroad Quattro,A8,A5,Q7
+ 585.0925 * Modelo=A8,A5,Q7
- 14749.295 * Modelo=A5,Q7
+ 821.5292 * Modelo=Q7
+ 241.2471 * Potencia
+ 477.4271 * Combustible=Diesel
- 164.9591 * Plazas
+ 1901.9263 * Anyo
+ 0.3035 * Km
- 3819950.3915

LM num: 25
Precio =
2760.2112 * Modelo=A4,A6,TT,Allroad Quattro,A8,A5,Q7
+ 2055.9672 * Modelo=A6,TT,Allroad Quattro,A8,A5,Q7
- 210.7363 * Modelo=TT,Allroad Quattro,A8,A5,Q7
+ 3986.2087 * Modelo=Allroad Quattro,A8,A5,Q7
+ 3775.7932 * Modelo=A8,A5,Q7
- 2949.5006 * Modelo=A5,Q7
+ 8047.366 * Modelo=Q7
+ 59.5446 * Potencia
+ 4737.7309 * Combustible=Diesel
- 1095.7087 * Plazas
+ 3283.9484 * Anyo
- 0.0798 * Km
- 6573972.2811

4.4.4 Obteniendo el modelo óptimo

En la siguiente tabla, podemos ver un pequeño resumen con los datos más importantes y relevantes para tomar la decisión sobre que método ha obtenido mejores resultados y, por tanto, qué modelo implementaremos en nuestra aplicación.

Algoritmo	Coeficiente Correlación	Error en la media absoluta	Error absoluto relativo
IBK	0.9268	2275.3392	26.4735%
Regresión Lineal	0.8789	3409.7753	39.6726%
M5P	0.9517	2289.7282	26.6409 %

Tabla 7. Comparativa de resultados

Queda bastante claro a simple vista, que el modelo que deberíamos implementar es el construido por el algoritmo M5P, pues es el que ha alcanzado un coeficiente de correlación mejor, conjuntamente con una media de error absoluto y relativo, aunque mayores que el IBK, aceptables.

Es por tanto, que pasaremos a nuestra aplicación a programar las 25 reglas obtenidas para predecir el precio del vehículo.

4.5 Implementación del modelo

Obtenido el modelo, tan solo nos queda implementarlo en nuestra aplicación. Para esto, crearemos un nuevo formulario donde el usuario del programa podrá seleccionar una serie de datos del vehículo que desee poner en venta. Dado que sólo hemos trabajado con una marca de vehículo y unos modelos en concreto, para experimentar con el modelo creado, tendremos que acortar la selección que el usuario podrá realizar.

Una vez cumplimentados todos los datos requeridos para la aplicación del modelo, el programa calculará un precio estimado para el vehículo basándose en estas quince reglas que el algoritmo aplicado ha creado para el conjunto de datos de muestra.

4.5.1 Volviendo a la aplicación

El formulario encargado de implementar el modelo de minería de datos generado mediante M5P será el formulario llamado Predicción. Para acceder a este formulario, bastará con seleccionarlo desde el formulario de MinnaCar inicial.

En este formulario, el usuario deberá seleccionar (dado que sólo hemos trabajado con la marca AUDI, la marca vendrá por defecto), el modelo de vehículo en el desplegable (en el que podrá seleccionar únicamente los modelos que seleccionamos como más frecuentes), la potencia, el número de plazas, el año de matriculación, los kilómetros y el combustible del vehículo.

Figura 33. Formulario Predicción

Es imprescindible que se hayan rellenado todos y cada uno de los datos para un óptimo resultado de la predicción. Tras cumplimentar todos los campos del formulario, bastará con pulsar el botón de Predecir! Para que comience la ejecución de la predicción.

Pasaremos ahora a analizar el código que corre por detrás de este formulario para ejecutar e implementar el modelo de minería de datos.

El código fuente del formulario se muestra a continuación con los comentarios añadidos explicando el proceso.

```
Private Sub Comando14_Click()
'Variables para albergar los datos introducidos por el usuario
Dim Modelo As String
Dim Potencia As Double
Dim Plazas As Double
Dim Anyo As Double
Dim Km As Double
Dim Combustible As String
Dim Precio As Double
Dim LM As Integer

'Igualamos cada variable con su cuadro de combinación o de texto correspondiente
Modelo = Me.Cuadro_combinado19.Value
Potencia = Val(Me.Texto2.Value)
Plazas = Val(Me.Texto4.Value)
Anyo = Val(Me.Texto6.Value)
Km = Val(Me.Texto8.Value)
Combustible = Me.Cuadro_combinado12.Value
```

```
'Inicializamos la variable Precio que se irá modificando a través del proceso del algoritmo
Precio = 0

'Este bloque de if's anidados corresponde al árbol de decisión que generó weka mediante
'M5P. Como podemos observar
'cuando se cumplen una serie de características, la variable LM se inicializa con un valor
determinado.
'Este valor corresponde al número de regla que se debe aplicar según los datos introducidos
por el usuario.
If Anyo <= 2005.5 Then
    If Anyo <= 2002.5 Then
        If Potencia <= 167.5 Then
            LM = 1
        End If
        If Potencia > 167.5 Then
            LM = 2
        End If
    End If
    If Anyo > 2002.5 Then
        If Potencia <= 202 Then
            LM = 3
        End If
        If Potencia > 202 Then
            If Modelo <> "A8" And Modelo <> "A5" And Modelo <> "Q7" Then
                If Km <= 128500 Then
                    LM = 4
                End If
                If Km > 128500 Then
                    If Km <= 171500 Then
                        If Combustible <> "Diesel" Then
                            If Potencia <= 252.5 Then
                                LM = 5
                            End If
                            If Potencia > 252.5 Then
                                LM = 6
                            End If
                        End If
                        If Combustible = "Diesel" Then
                            LM = 7
                        End If
                    End If
                    If Km > 171500 Then
                        If Potencia <= 212.5 Then
                            LM = 8
                        End If
                        If Potencia > 212.5 Then
                            LM = 9
                        End If
                    End If
                End If
            End If
            If Modelo = "A8" Or Modelo = "A5" Or Modelo = "Q7" Then
                LM = 10
            End If
        End If
    End If
End If
```

```
If Anyo > 2005.5 Then
    If Potencia <= 166.5 Then
        If Anyo <= 2007.5 Then
            LM = 11
        End If
        If Anyo > 2007.5 Then
            If Potencia <= 130.5 Then
                LM = 12
            End If
            If Potencia > 130.5 Then
                LM = 13
            End If
        End If
    End If
    If Potencia > 166.5 Then
        If Km <= 22950 Then
            If Plazas <= 4.5 Then
                If Modelo <> "Allroad Quattro" And Modelo <> "A8" And Modelo <> "A5" And
Modelo <> "Q7" Then
                    LM = 14
                End If
                If Modelo = "Allroad Quattro" Or Modelo = "A8" Or Modelo = "A5" Or Modelo =
"Q7" Then
                    If Km <= 9400 Then
                        If Potencia <= 185 Then
                            LM = 15
                        End If
                        If Potencia > 185 Then
                            LM = 16
                        End If
                    End If
                    If Km > 9400 Then
                        LM = 17
                    End If
                End If
            End If
            If Plazas > 4.5 Then
                If Modelo <> "TT" And Modelo <> "Allroad Quattro" And Modelo <> "A8" And
Modelo <> "A5" And Modelo <> "Q7" Then
                    If Km <= 16 Then
                        If Potencia <= 175 Then
                            LM = 18
                        End If
                        If Potencia > 175 Then
                            LM = 19
                        End If
                    End If
                    If Km > 16 Then
                        If Potencia <= 229 Then
                            LM = 20
                        End If
                        If Potencia > 229 Then
                            LM = 21
                        End If
                    End If
                End If
```

```
                If Modelo = "TT" Or Modelo = "Allroad Quattro" Or Modelo = "A8" Or Modelo =
"A5" Or Modelo = "Q7" Then
                    If Potencia <= 283 Then
                        LM = 22
                    End If
                    If Potencia > 283 Then
                        If Potencia <= 400 Then
                            LM = 23
                        End If
                        If Potencia > 400 Then
                            LM = 24
                        End If
                    End If
                End If
            End If
        End If
        If Km > 22950 Then
            LM = 25
        End If
    End If
End If

'Como se puede observar, el código consiste en transformar literalmente el resultado
obtenido mediante
'Weka a código en visual basic para access
'Con un Select Case, seleccionamos la regla a aplicar según el valor tomado por LM al
finalizar el
'bloque de if's anidados

Select Case LM    ' Evalúa Número.
Case 1

    If Modelo = "A4" Or Modelo = "A6" Or Modelo = "TT" Or Modelo = "Allroad Quattro" Or
Modelo = "A8" Or Modelo = "A5" Or Modelo = "Q7" Then
    Precio = Precio + 619.8268
    End If
    If Modelo = "A6" Or Modelo = "TT" Or Modelo = "Allroad Quattro" Or Modelo = "A8" Or
Modelo = "A5" Or Modelo = "Q7" Then
    Precio = Precio + 25.2089
    End If
    If Modelo = "TT" Or Modelo = "Allroad Quattro" Or Modelo = "A8" Or Modelo = "A5" Or
Modelo = "Q7" Then
    Precio = Precio + 3331.006
    End If
    If Modelo = "Allroad Quattro" Or Modelo = "A8" Or Modelo = "A5" Or Modelo = "Q7" Then
    Precio = Precio + 170.4255
    End If
    If Modelo = "A8" Or Modelo = "A5" Or Modelo = "Q7" Then
    Precio = Precio + 6.937
    End If
    If Modelo = "A5" Or Modelo = "Q7" Then
    Precio = Precio + 12.8673
    End If
    If Modelo = "Q7" Then
    Precio = Precio + 63.7384
    End If
    If Combustible = "Diesel" Then
```

```
    Precio = Precio + 1712.5738
    End If
    Precio = Precio + (21.7723 * Potencia) - (96.7428 * Plazas) + (818.6264 * Anyo) -
(0.0115 * Km) - 1633227.9317

'LM num: 1 en código WEKA
'Precio =
'    619.8268 * Modelo=A4,A6,TT,Allroad Quattro,A8,A5,Q7
'    + 25.2089 * Modelo=A6,TT,Allroad Quattro,A8,A5,Q7
'    + 3331.006 * Modelo=TT,Allroad Quattro,A8,A5,Q7
'    + 170.4255 * Modelo=Allroad Quattro,A8,A5,Q7
'   + 6.937 * Modelo=A8,A5,Q7
'    + 12.8673 * Modelo=A5,Q7
'    + 63.7384 * Modelo=Q7
'    + 21.7723 * Potencia
'    + 1712.5738 * Combustible=Diesel
'    - 96.7428 * Plazas
'    + 818.6264 * Anyo
'    - 0.0115 * Km
'    - 1633227.9317
```

[Trabajamos del mismo modo para las 25 reglas]

```
Case 25
    If Modelo = "A4" Or Modelo = "A6" Or Modelo = "TT" Or Modelo = "Allroad Quattro" Or
Modelo = "A8" Or Modelo = "A5" Or Modelo = "Q7" Then
    Precio = Precio + 2760.2112
    End If
    If Modelo = "A6" Or Modelo = "TT" Or Modelo = "Allroad Quattro" Or Modelo = "A8" Or
Modelo = "A5" Or Modelo = "Q7" Then
    Precio = Precio + 2055.9672
    End If
    If Modelo = "TT" Or Modelo = "Allroad Quattro" Or Modelo = "A8" Or Modelo = "A5" Or
Modelo = "Q7" Then
    Precio = Precio - 210.7363
    End If
    If Modelo = "Allroad Quattro" Or Modelo = "A8" Or Modelo = "A5" Or Modelo = "Q7" Then
    Precio = Precio + 3986.2087
    End If
    If Modelo = "A8" Or Modelo = "A5" Or Modelo = "Q7" Then
    Precio = Precio + 3775.7932
    End If
    If Modelo = "A5" Or Modelo = "Q7" Then
    Precio = Precio - 2949.5006
    End If
    If Modelo = "Q7" Then
    Precio = Precio + 8047.366
    End If
    If Combustible = "Diesel" Then
    Precio = Precio + 4737.7309
    End If
    Precio = Precio + (59.5446 * Potencia) - (1095.7087 * Plazas) + (3283.9484 * Anyo) -
(0.0798 * Km) - 6573972.2811
'LM num: 25
'Precio =
'    2760.2112 * Modelo=A4,A6,TT,Allroad Quattro,A8,A5,Q7
'    + 2055.9672 * Modelo=A6,TT,Allroad Quattro,A8,A5,Q7
'    - 210.7363 * Modelo=TT,Allroad Quattro,A8,A5,Q7
'    + 3986.2087 * Modelo=Allroad Quattro,A8,A5,Q7
```

```
'     + 3775.7932 * Modelo=A8,A5,Q7
'     - 2949.5006 * Modelo=A5,Q7
'     + 8047.366 * Modelo=Q7
'     + 59.5446 * Potencia
'     + 4737.7309 * Combustible=Diesel
'     - 1095.7087 * Plazas
'     + 3283.9484 * Anyo
'     - 0.0798 * Km
'     - 6573972.2811
Case Else     ' Otros valores.
    Debug.Print "Error de asociación en el árbol de decisión"
End Select

'Por último, se muestra el valor en la casilla de precio del vehículo
Me.Texto15.Value = Precio

End Sub
```

Figura 34. Predicción precio

Como podemos ver, para un Audi A3 del 2005 con dichas características, le correspondería aproximadamente un precio de 16958€. Eso si, hay que recordar los `2289.7282` de media de error, por lo que el precio sería una aproximación entre 16958 ±2289.

5 CONCLUSIONES

Como hemos podido ver y hemos explicado durante la memoria de este proyecto, la minería de datos es una herramienta con un potencial increíble y aplicable en un sinfín de proyectos, circunstancias y finalidades.

A través de este proyecto, nos hemos podido introducir un poco en este mundo de la obtención de información relevante de masas de datos compactas. Hemos introducido el concepto de minería de datos, así como analizado varios de sus numerosos tipos y familias de métodos de cálculo y obtención de resultados.

Evidentemente, se puede profundizar muchísimo más en la minería de datos, pero se escapa del objetivo de este proyecto. No obstante, hemos podido aprender a utilizar una herramienta muy potente y con licencia OpenGPL para la minería de datos llamada Weka, hemos analizado sus características y hemos surcado por encima de sus numerosas posibilidades y funcionalidades.

A través de esta potente aplicación, se han podido analizar y aplicar de manera práctica y constructiva varios de los métodos de minería de datos que se encuentran implementados en la misma. Hemos sido capaces de analizar los resultados y discernir entre atributos influyentes y no influyentes, se han filtrado instancias de la gran masa de datos obtenidos y se han creado nuevos modelos de minería de datos para nuestro conjunto de datos de muestra.

Al analizar los resultados otorgados por estos nuevos modelos, hemos tenido que comprender e interpretar dichos resultados, para discernir qué modelo se adaptaba mejor a nuestras necesidades y obteníamos los mejores resultados.

Sin embargo, todo esto no habría sido posible sin la realización en Visual Basic para aplicaciones y Access una pequeña pero potente aplicación que ha sido capaz de descargarse información de más de 6000 vehículos y organizar dicha información para almacenarla correctamente en una base de datos en apenas unos minutos. Además, nos ha proporcionado un entorno idóneo para implementar el modelo de minería de datos obtenido mediante el algoritmo M5P y así poner en práctica la predicción del precio de vehículos de segunda mano con un solo click de ratón.

Siento que me repetiré si sigo con todas las conclusiones que he sacado de la realización de este proyecto. Si puedo decir no obstante, que me ha servido con creces para conseguir mi objetivo personal, que no era otro que el de introducirme en el mundo de la minería de datos.

Dado que no tuve la oportunidad de profundizar en esta serie de conceptos en ninguna de las asignaturas de la titulación, ha sido de gran ayuda poder realizar este proyecto para resolver mis dudas e inquietudes acerca de esta materia. Asimismo, estas nuevas capacidades y conocimientos adquiridos, me resultarán de gran utilidad en mi futura vida profesional, pues como se ha comentado en numerosas ocasiones en esta memoria, la minería de datos es una herramienta de vital importancia en el mundo empresarial. Mundo al que estoy fuertemente ligado, ya que he tenido la suerte de ingresar en una empresa multinacional como Técnico

Informático y donde estoy seguro que todos estos conocimientos adquiridos me serán de gran ayuda para progresar profesionalmente.

No puedo olvidar, que han mejorado mis capacidades y habilidades para la programación y abstracción de procesos algorítmicos, así como mi nivel en el lenguaje de programación Visual Basic.

A nivel personal, puedo decir, que quitando los conocimientos teórico-prácticos que he aprendido, he podido aprender a valorar el esfuerzo y dedicación que supone la elaboración de una aplicación desde cero, ser el encargado de la toma de necesidades, del análisis de la situación, del desarrollo, de la implementación, de las pruebas… He podido comprender de una forma mejor, todo el ciclo que supone la puesta en marcha de una aplicación, es decir, su ciclo de vida, ciclo que tantas veces hemos oído a lo largo de la titulación.

Me gustaría decir que ha mejorado mi trabajo en equipo, no obstante, por incompatibilidades, tuve que optar por realizar el proyecto individualmente, no obstante, mi director de proyecto me ha sido de vital importancia resolviendo mis dudas y guiándome en el camino cuando no sabía que salida tomar exactamente.

6 AMPLIACIONES

Como se suele decir "cuanto más dulce mejor". Este proyecto cuenta con numerosas ampliaciones posibles y mejoras. Pasaré a redactar algunas de ellas:

- El proceso de obtención de datos de las páginas web, resulta demasiado pesado y lento. Con un poco más de tiempo, dedicación y un análisis exhaustivo podría agilizarse un poco este proceso.
- Podría generalizarse la obtención de información, y recoger información no sólo de una página web, si no de varias páginas y así poder contrastar la información.
- La aplicación en Access, podría trasladarse a cualquier otra plataforma o entorno de programación para facilitar su utilización. Realizarla como aplicación web sería idóneo. De este modo, podría ofrecerse como un servicio para todas aquellas personas que quisieran poner su vehículo en venta o comprar uno, poder orientarse a los precios actuales del mercado haciendo una búsqueda y análisis de varias páginas y anuncios de coches.
- En este proyecto, tan sólo se ha implementado el modelo de minería de datos para la marca AUDI, podría implementarse para el resto de marcas para que así fuera más completa. No se ha realizado esto, pues el objetivo era analizar y poner en práctica la minería de datos.
- Mediante Weka, podríamos combinar varios métodos de minería de datos si profundizáramos un poco más en la materia para obtener un modelo óptimo para cada marca de vehículo.
- Sería ideal, combinar de alguna forma la aplicación con Weka, para que los modelos de minería de datos se construyeran "al vuelo" con los datos y la información acabada de obtener al momento. De este modo, nuestros modelos estarían siempre

actualizados, pues como todo, los tiempos cambian, y lo que hoy puede influir bastante en el precio de un vehículo, puede no serlo el día de mañana. Como ejemplo cabría decir, que hace unos cuantos años, poca gente podía permitirse el lujo de disponer de aire acondicionado en el coche y sin embargo hoy en día, es algo casi indispensable y asequible.

Existen numerosas mejoras y ampliaciones que se podrían hacer, y me gustaría haber podido tener la capacidad y tiempo para realizarlas, no obstante, estoy orgulloso del trabajo realizado.

BIBLIOGRAFÍA

J. Hernández, M. J. Ramírez, C. Ferri "Introducción a la Minería de Datos" © Prentice Hall / Addison-Wesley, *ISBN 84 205 4091 9*

BW380 Data Mining "SAP Business Intelligence: Analysis Processes and Data Mining" SAPDOCS

BW310 "Data Warehousing" SAPDOCS

BW360 "SAP BI Performance & Administration" SAPDOCS

Master Thesis "Evaluation of Data Mining Methods to support data warehouse administration and monitoring in SAP business warehouse" Narasimha Raju Alluri (y en consecuencia, toda su bibliografía añadida)

A presentation on data mining with SAP BW 3.5. SAPNET Lesley 2004

RECURSOS DE INTERNET

No son todos los que son, pero si son todos los que están.

http://www.google.es

http://www.witnessminer.com

http://www.appstate.edu/~whiteheadjc/service/logit/

http://en.wikipedia.org/wiki/Main_Page

http://www.cs.waikato.ac.nz/ml/weka/

http://old.nabble.com

http://comments.gmane.org/gmane.comp.ai.weka/20508

http://www.mrexcel.com

http://wekadocs.com/

http://www.opentox.org/dev/documentation/components/m5p

www.canalvisualbasic.net/

www.vb-mundo.com

www.vbtutor.net/vbtutor.html

www.lawebdelprogramador.com

www.microsoft.com

www.wordreference.com