Fundamentos de Aprendizaje Automático 2015/2016

PRÁCTICA Nº 2

Objetivo

Los objetivos de esta práctica son dobles. Por un lado se introduce el uso de la herramienta **Weka** y, por otro se estudian dos nuevos modelos teóricos mediante su implementación en Java. Los clasificadores a estudiar son los *vecinos próximos* y la *regresión logística*.

Tareas

La planificación temporal sugerida y las tareas a llevar a cabo son las siguientes:

- *1º semana*: Utilizando la herramienta **Weka**, realizar una clasificación *Naive-Bayes* para los conjuntos de datos estudiados en la práctica 1. Analizar los resultados obtenidos con la herramienta y compararlos con los obtenidos en la implementación propia que se realizó en la práctica 1 para este clasificador.

- *2º semana:* Implementar el algoritmo de *vecinos próximos* para realizar una tarea de clasificación de los siguientes conjuntos de datos (probar con diferentes valores de vecindad (K=1, 3, 5, 11, 21 y 51). Ejecutar también estos conjuntos de datos con la implementación de vecinos próximos de Weka.

  - ✓ ***Heart disease*** (http://archive.ics.uci.edu/ml/datasets/Heart+Disease).
  - ✓ **Wine red quality** ((http://archive.ics.uci.edu/ml/datasets/Wine+Quality).
  - ✓ ***Wdbc*** (http://archive.ics.uci.edu/ml/machine-learning-databases/breast-cancer-wisconsin/)

- *3º semana*: Clasificar los patrones del conjunto de datos ***Wdbc*** (http://archive.ics.uci.edu/ml/machine-learning-databases/breast-cancer-wisconsin/) mediante *regresión logística*, aplicando el algoritmo de maximización de la verosimilitud. Comparar los resultados obtenidos para el conjunto de datos ***wdbc*** con el algoritmo de *regresión logística* y con el de *vecinos próximos*. Ejecutar este conjunto de datos con la regresión logística de Weka.

- *4ª semana:* El método de los vecinos próximos tiene algunas desventajas o aspectos que podrían mejorarse. Algunas de los puntos cuestionables son la elección de la función de distancia, el ajuste del valor de vecindad "k" o la estimación de la clase entre los ejemplos vecinos. Precisamente, en la implementación que se ha hecho de este algoritmo se ha considerado por igual a todas las instancias vecinas, cuando lógicamente no lo son porque algunas están más cercanas a la instancia a clasificar que otras. La idea es por tanto

mejorar la selección de la clase estimada con el objetivo de comprobar si es posible afinar las tasas de acierto en los conjuntos de datos propuestos. Para una revisión de algunas alternativas se recomienda consultar el apartado 2.3 del artículo "*Survey of Improving K-Nearest-Neighbor for Classification*" que se incluye como parte del material de la práctica.

Lenguaje

El lenguaje a utilizar para el desarrollo de la práctica será Java. Utilizar la estructura de clases y métodos propuesta en la práctica 1.

Fecha de entrega y entregables

Martes 3 de Noviembre: Fichero comprimido con el código de las clases Java para los clasificadores de vecinos próximos (original y mejorado en clases distintas) y regresión logística. Memoria con los siguientes apartados:

Apartado 1	Resultados de Naive-Bayes obtenidos por Weka en los conjuntos de datos de la práctica 1 y análisis comparativo con los resultados obtenidos por la propia implementación Java.
Apartado 2	Resultados de la clasificación mediante vecinos próximos (implementación original) para diferentes valores de vecindad en los conjuntos de datos propuestos. Comparación con los resultados proporcionados por Weka
Apartado 3	Resultados de la clasificación mediante regresión logística en el conjunto de datos propuesto. Comparación con los resultados proporcionados por Weka
Apartado 4	Resultados de la clasificación mediante vecinos próximos (implementación mejorada) para diferentes valores de vecindad en los conjuntos de datos propuestos. Explicación de la mejora implementada

Breve resumen de Weka para clasificación

Weka es una suite de algoritmos de aprendizaje automático desarrollados por la Universidad de Waikato implementados en Java. Además de los algoritmos de aprendizaje, Weka proporciona herramientas adicionales para realizar transformaciones sobre los datos y visualización de resultados. El diseño de Weka permite añadir nuevas funcionalidades de forma relativamente sencilla. Weka trabaja de forma nativa con ficheros en un formato denominado *arff* cuya estructura es la siguiente:

Nombre de la relación

@relation <nombre-de-relación>

Declaración de atributos

@attribute <nombre> <tipo>

Donde <tipo> puede ser Real, Numeric, Date, String o enumerado (en este caso los posibles valores aparecen entre llaves separados por comas)

Datos

@data

A continuación se especifican los datos, separando entre comas los atributos y las relaciones mediante saltos de línea

No obstante, Weka admite también otros formatos de datos, como CSV, C4.5 o lectura a través de las tablas de una Base de Datos. Al arrancar Weka se muestra una interfaz con las posibles opciones de trabajo:

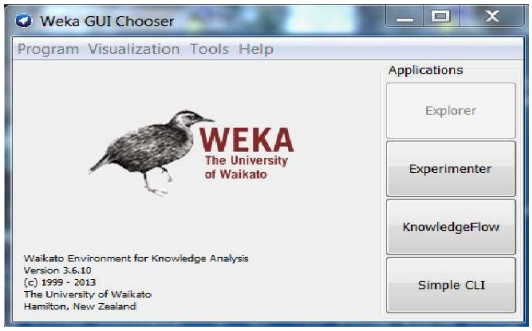

Figura 1: Selector de interfaces

De las cuatro posibles alternativas, en esta práctica vamos a trabajar con la opción "**Explorer**". Esta interfaz es probablemente la más utilizada y permite realizar operaciones sobre un solo archivo de datos. La pantalla inicial de esta interfaz es la siguiente:

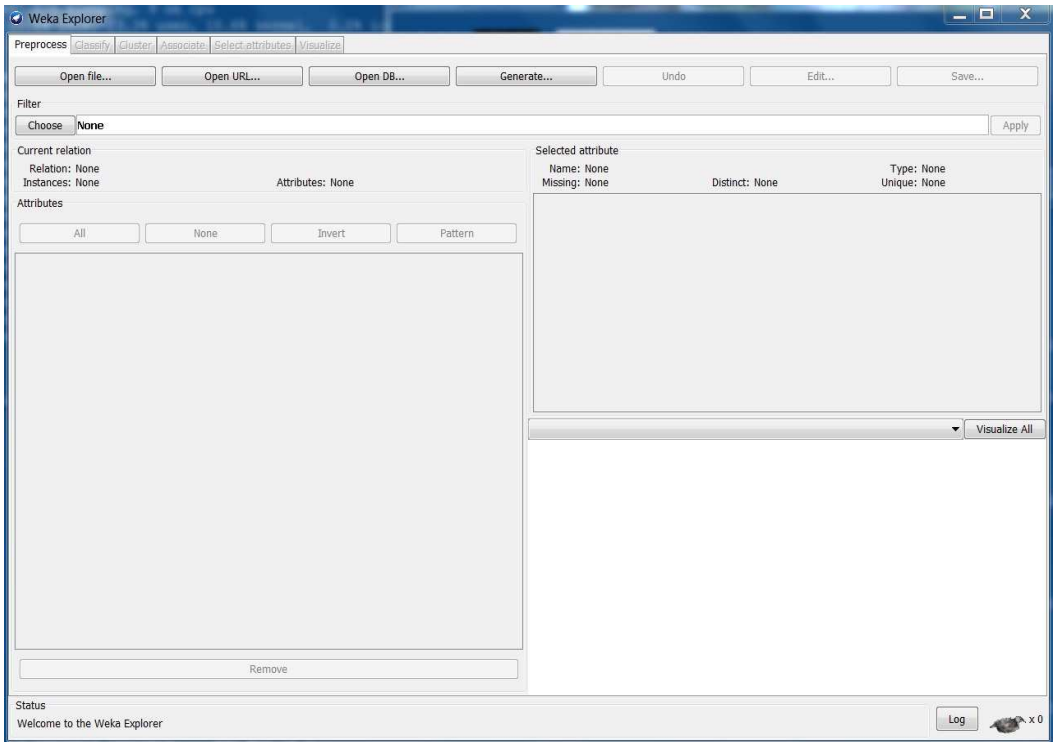

Figura 2: Preprocesado en el Explorer

En la figura aparece la imagen inicial para la pestaña "*Preprocess*". Esta pestaña nos permite realizar varias acciones, aunque la que interesa en nuestro caso es la opción "*Open file*". Desde aquí seleccionaremos el fichero con el conjunto de datos que nos interesa. Una vez cargado el fichero aparecerán en la pantalla diferentes informaciones sobre el contenido del fichero.

Para ejecutar una tarea de clasificación debemos seleccionar en esta ventana la pestaña "*Clasify*". Una vez elegida esta opción, la ventana principal de ***Explorer*** cambia de aspecto y nos muestra, entre otras cosas, el botón "*Choose*", desde el cual podemos seleccionar el algoritmo que nos interesa. Tras seleccionar el algoritmo podemos ejecutarlo mediante el botón "*Start*". Los resultados de la ejecución aparecen en la parte derecha de la pantalla, bajo el título "*Classifier Output*". En la parte izquierda de la pantalla podemos ver y cambiar las opciones de prueba. Por defecto, Weka utiliza validación cruzada con 10 folds, lo que permite comparar los resultados con la implementación que se hizo en la práctica 1 de este clasificador. Bajo el botón "*Start*" aparece la etiqueta "*Result list*" que resumen las distintas ejecuciones realizadas. Pulsando el botón derecho del ratón sobre cualquiera de ellas podemos ver diferentes opciones, especialmente relacionadas con la visualización.

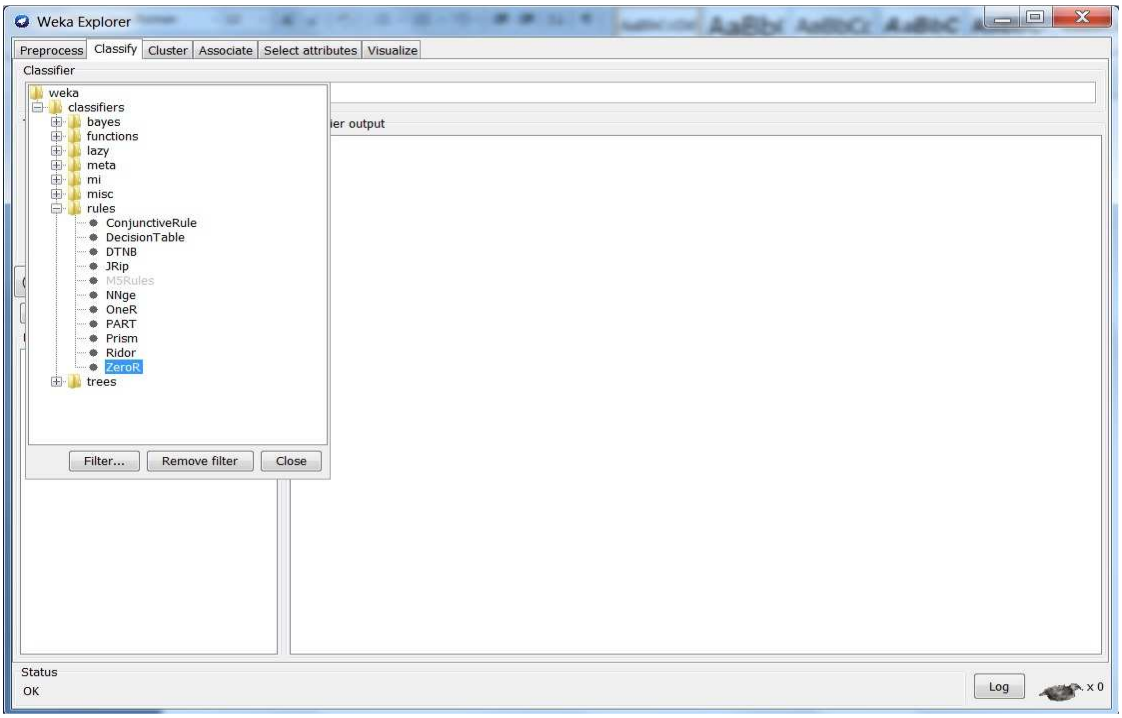

Figura 3: Selección del algoritmo de clasificación

Referencias

1. Diego García Morate. Manual de Weka.
2. Manual de Weka: http://www.cs.waikato.ac.nz/~ml/weka/documentation.html