

## Práctica Tema 2

### Clasificador $k$ -nn por votación

### Curso 2016-2017

El objetivo de esta práctica es familiarizar al alumno con uno de los esquemas de clasificación máquina más sencillos, el clasificador  $k$ -nn por votación (*voting  $k$ -nn*). Para ello se utilizará la herramienta de libre distribución Weka (*Waikato Environment for Knowledge Analysis - Entorno para Análisis del Conocimiento*), un software desarrollado por la Universidad de Waikato para abordar procesos de aprendizaje máquina utilizando Java. Puede descargar el software de la URL <http://www.cs.waikato.ac.nz/ml/weka/index.html>, o utilizarlo en el laboratorio accediendo a través de la ruta 'C:\Archivos de Programa\Weka-3-6\'. Weka implementa varios esquemas de aprendizaje máquina, tanto para resolver procesos de clasificación como de estimación (entre otros). En esta práctica utilizaremos Weka para diseñar el clasificador  $k$ -nn por votación.

Existen muchos manuales disponibles sobre Weka, tanto en la página web de Weka como en páginas personales. Para el alumno interesado, acompañando a este documento se incluye un manual en castellano. Para los propósitos de esta práctica es suficiente con leer el apartado 5.2 del documento.

Al arrancar la aplicación de Weka aparecerá una ventana similar a la mostrada en la Figura 1.

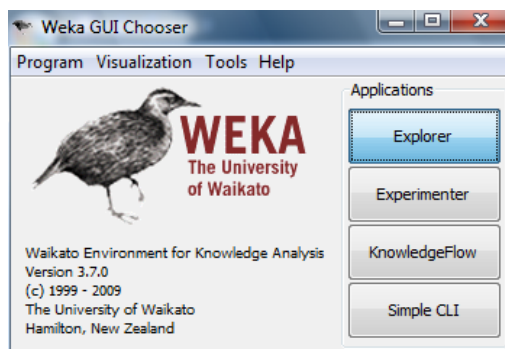


Figura 1. Pantalla de entrada de Weka.

Observe que existen 4 modos de funcionamiento: *Explorer*, *Experimenter*, *Knowledge Flow* y *Simple CLI*:

- *Explorer* es un entorno visual que ofrece una interfaz gráfica con varios paneles para utilizar distintos esquemas de aprendizaje máquina
- *Experimenter* es un entorno centrado en la automatización de tareas para realizar comparaciones sistemáticas de distintos algoritmos sobre un conjunto de datos, facilitando así la realización de experimentos a gran escala
- *Knowledge Flow* soporta las mismas funcionalidades básicas que *Explorer* pero con una interfaz que permite generar proyectos mediante flujos de información

- *Simple CLI (Command-Line Interface)* es un entorno consola para invocar directamente a las opciones de Weka desde línea de comandos

En esta práctica se analizan las prestaciones de un esquema de clasificación  $k$ -NN por votación utilizando el modo *Explorer*. Seleccione por tanto el botón que nos introduce en este modo de funcionamiento.

Observe que tiene activados los botones que permiten abrir ficheros. De forma nativa, Weka trabaja con su propio formato de ficheros, el formato *.arff (Attribute Relation File Format)*, aunque también puede cargar datos en otros formatos (ASCII,...). Los datos proporcionados para este ejercicio siguen este formato. Pulse sobre “Open file” y navegue hasta la carpeta en la que haya almacenado los datos que se proporcionan para este ejercicio: datos sintéticos bidimensionales (característica\_1 y característica\_2, agrupados en el vector de características  $\underline{x}$ ) correspondientes a un problema de clasificación binario (etiqueta “1” para identificar que las observaciones fueron generadas por la Hipótesis H1 y “-1” para la hipótesis complementaria H0). La etiqueta se identifica a través de la variable  $t$ .

- *DatosTrain.arff*: 960 observaciones (400/560), a utilizar como conjunto de entrenamiento de gran tamaño para diseñar el clasificador. Es decir,

$$T = \left\{ \underline{x}^{(i)}, t^{(i)} \right\}_{i=1}^{960}$$

El identificador de la observación se indica a través del superíndice ( $i$ ): vector de características y etiqueta asociados a la misma observación deben tener el mismo superíndice.

- *DatosTrainPeq.arff*: 120 observaciones (50/70), a utilizar como conjunto de entrenamiento reducido para diseñar el clasificador

$$T_{peq} = \left\{ \underline{x}^{(i)}, t^{(i)} \right\}_{i=1}^{120}$$

- *DatosTest.arff*: 1200 observaciones (500/700) independientes de los conjuntos anteriores; a utilizar como conjunto de test para evaluar las prestaciones del clasificador

$$T_{test} = \left\{ \underline{x}^{(i)}, t^{(i)} \right\}_{i=1}^{1200}$$

Seleccione cualquiera de los ficheros “DatosTrain ...”. Una vez abierto, en la parte inferior izquierda de la ventana aparece la lista de variables disponibles, con la primera variable seleccionada (fondo azul). A la derecha aparece un resumen con algunos parámetros de la variable junto con una representación gráfica que corresponde al histograma de la variable (utilizando un color diferente -azul y rojo en este caso- para mostrar cada una de las clases). Presione el botón “Visualize All” para ver los histogramas de todas las variables (tanto de entrada como de salida al clasificador).

A partir del histograma de cada característica, justifique si las funciones densidad de probabilidad marginales pueden considerarse de naturaleza gaussiana.

Para observar el diagrama de puntos (*scatter plot*) asociado a la representación simultánea de las 2 características del vector  $\underline{x}$ , puede acceder a la pestaña *Visualize*.

Observe el *scatter plot* correspondiente a cada uno de los 3 conjuntos (deberá cargar cada conjunto de datos); puede comprobar que en todos los conjuntos se mantiene la misma probabilidad a priori para cada una de las hipótesis. Compare los *scatter plots* de los 3 conjuntos y justifique razonadamente si se puede considerar que los ejemplos de los 3 conjuntos proceden de las mismas distribuciones estadísticas (desconocidas).

A la vista de los resultados, ¿qué puede deducirse sobre el clasificador ideal (lineal/ no lineal)?

Pulsando en la segunda pestaña de la parte superior de la pantalla se accede a la ventana de clasificación. Seleccione en esta ventana el clasificador a utilizar (pinche sobre el botón “*Choose*”). Observe que existen muchos tipos de clasificadores. En concreto, para este ejercicio nos centraremos en el **clasificador *k*-nn por votación**. Este clasificador, ya presentado en las clases teóricas, se implementa en Weka bajo el nombre IBk, y se encuentra en el grupo de los clasificadores denominados “*lazy*” (perezosos) (véase la Figura 2).

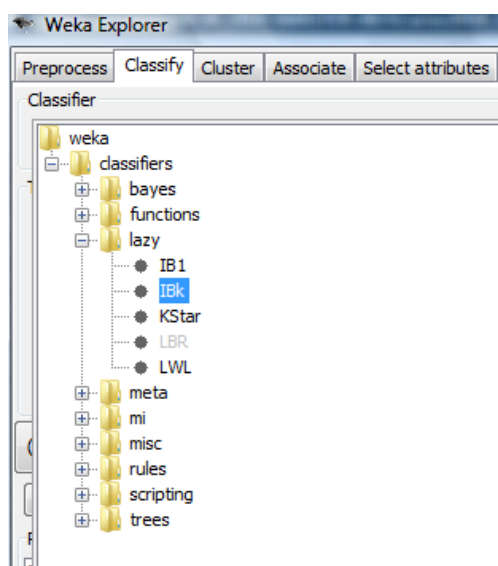


Figura 2. Situación del clasificador *k*-NN por votación en la pestaña “Classify”.

Al seleccionar “IBk” verá que, en la caja de texto superior (contigua al botón “Choose”), aparece el algoritmo seleccionado con varios parámetros. Pinche sobre esta caja de texto para examinar las propiedades del clasificador.

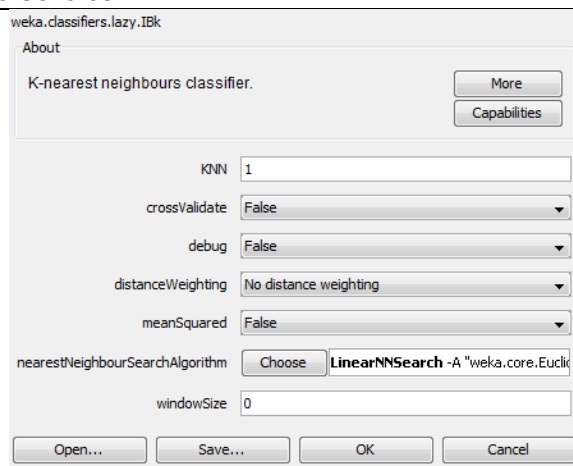


Figura 3. Parámetros de la implementación de clasificador k-NN por votación en Weka.

Si selecciona el botón “More” obtendrá información más extensa sobre el clasificador y su utilización.

Desde la ventana que se muestra en la Figura 3 se pueden modificar los parámetros del clasificador. En concreto, la implementación de Weka permite configurar los siguientes parámetros:

- “KNN”: número de vecinos a considerar. Éste es el **único** parámetro que debe explorar en esta práctica.
- “crossValidate”: utiliza la técnica *leave-one-out* (LOO) para determinar el mejor valor de  $k$  (número de vecinos  $k$ ) a utilizar en el clasificador final. Para un conjunto de  $N$  ejemplos, el esquema LOO realiza  $N$  diseños, cada uno de ellos con  $N-1$  ejemplos, dejando cada vez uno de los ejemplos del conjunto  $T$  fuera del conjunto utilizado para construir el clasificador. El ejemplo no considerado en el diseño se utiliza posteriormente para evaluar las prestaciones del clasificador.

La opción “crossValidate” debe estar desactivada en esta práctica.

- “distanceWeighting”: determina si ponderar la influencia de cada vecino antes de hacer la clasificación. Como se muestra en la Figura 3, esta opción debe estar desactivada para el caso del clasificador k-NN por votación.
- “meanSquared”: si la variable a estimar es numérica, permite activar la opción de minimizar el error cuadrático cuando se activa el parámetro crossvalidate. Para el caso del clasificador k-NN por votación, esta opción debe estar desactivada (véase Figura 3).
- “windowSize”: si es 0, el número de ejemplos de entrenamiento es ilimitado. Si el valor  $p$  indicado en este cuadro de edición es mayor que cero, únicamente se almacenan los  $p$  últimos ejemplos de entrenamiento. En este ejercicio, el valor de este parámetro debe mantenerse a 0 para considerar todos los ejemplos de entrenamiento disponibles.

El clasificador  $k$ -nn se diseña almacenando todos los ejemplos de entrenamiento disponibles (a menos que se restrinja con la opción “windowSize”). Puesto que en este ejercicio se proporciona un conjunto de test (*DatosTest.arff*), las prestaciones del clasificador se evaluarán **SIEMPRE** sobre este conjunto.

Explore qué otros métodos de evaluación se pueden utilizar y justifique razonadamente por qué no conviene utilizar la opción “Use training set” (véase la Figura 4).

En esta práctica, seleccione el *radiobutton* “*Supplied test set*” y busque el fichero *DatosTest.arff*, tal y como se indica en la Figura 4.

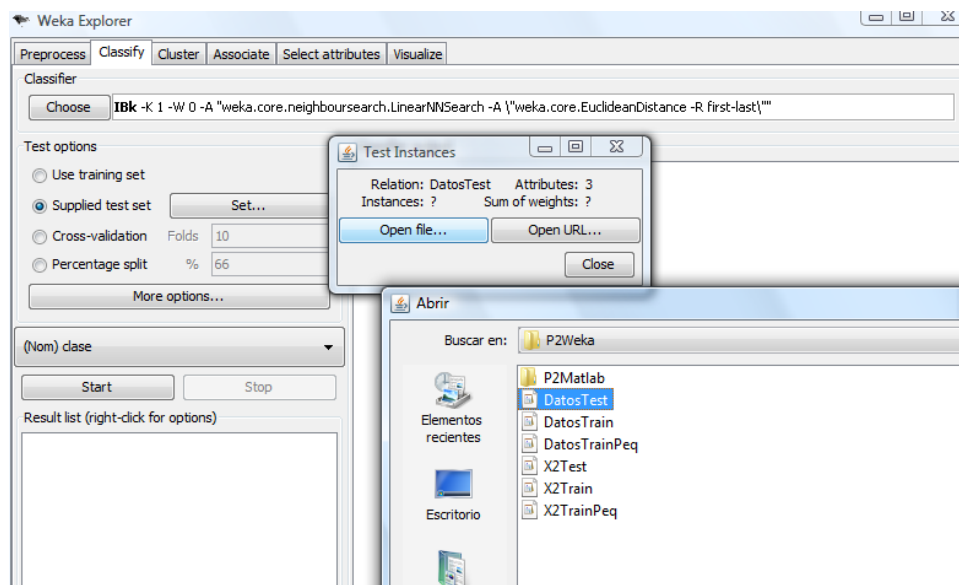



Figura 4. Selección del conjunto de test para evaluar las prestaciones del clasificador.

Seleccione la variable a utilizar como clase (salida del clasificador), que en este caso es la que aparece por defecto, y ejecutar el clasificador presionando el botón “*Start*”.


Los resultados de la clasificación aparecen en la ventana de la derecha. Lea las páginas 17 a 21 el manual para aprender a interpretar estos resultados.

Justifique, de manera intuitiva, por qué es conveniente evaluar las prestaciones del clasificador con un conjunto (*DatosTest.arff*) diferente al utilizado para diseñar el clasificador (conjuntos *DatosTrain.arff* / *DatosTrainPeq.arff*). 

Teniendo en cuenta que las prestaciones de CUALQUIER clasificador se deben evaluar con un conjunto de ejemplos independiente al utilizado en el diseño del clasificador, indique la tasa de acierto (en %) obtenida al evaluar el clasificador  $k$ -nn por votación con distintos valores del parámetro  $k$ . Explore un rango de valores de  $k$  suficientemente amplio e indique las prestaciones obtenidas con cada valor explorado, utilizando para ello una tabla y los dos conjuntos de entrenamiento proporcionados (*DatosTrainPeq* y *DatosTrain*).

Comparando los valores de la tabla, indique cuál es el mejor clasificador para cada conjunto de diseño.

En base a los resultados anteriores, ¿cuál es el mejor valor del parámetro  $k$ ?

¿Depende el valor anterior del tamaño del conjunto de entrenamiento?, ¿con qué conjunto de diseño se obtiene el menor valor de  $k$ ? 

Justifique todas sus respuestas.