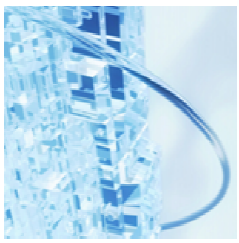


Prácticas de Minería de Datos

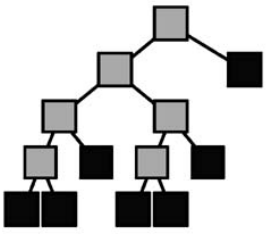
Grado en Ingeniería Informática

Curso 2013-14

PRÁCTICA 5

Clasificación

Uso del *Experimenter* de WEKA y Clasificación Sensible al Coste



OBJETIVOS

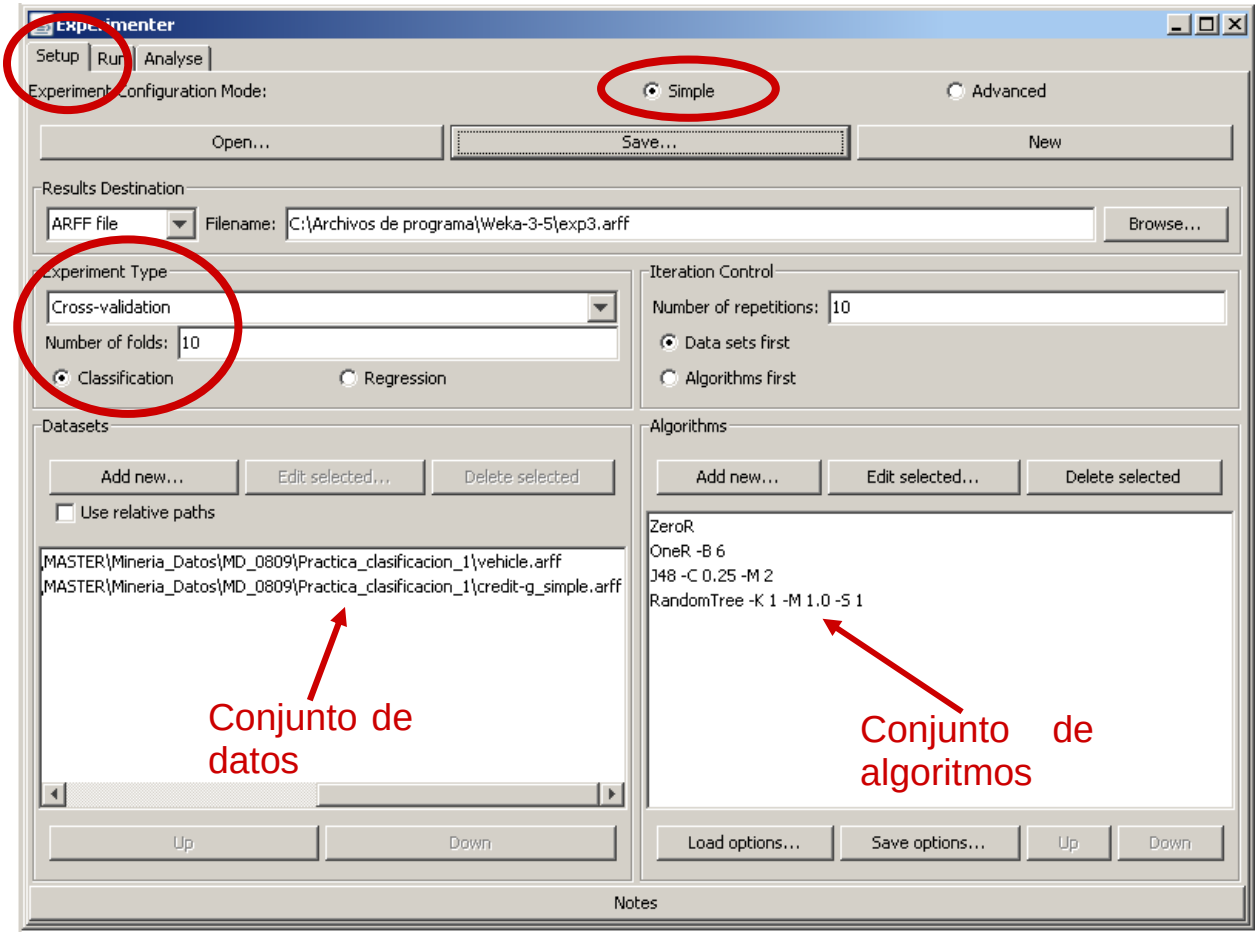
- Familiarizarse con el *Experimenter* de WEKA para realizar comparativas.
- Aprender a realizar aprendizaje sensible al coste mediante la utilización de la matriz de costes.

1. Primera parte: Comparación de métodos mediante el *Experimenter*

El modo experimentador (*Experimenter*) permite aplicar uno o varios métodos de clasificación sobre uno o varios conjuntos de datos y, posteriormente realizar contrastes estadísticos entre ellos para determinar de manera estadística cual se comporta mejor (dirá si las diferencias aparentes en porcentajes de aciertos de distintos algoritmos son estadísticamente significativas, o son debidas al azar).

Las fases de uso de Experimenter son **Setup** (configurar), **Run** (ejecutar) y **Analyse** (análisis estadístico), las cuales se pueden ver en las pestañas superiores.

Una vez abierto el modo experimentador se observa una ventana como la de la figura, que corresponde a la sección *Setup*. Por defecto el modo experimentador está configurado en modo simple, no obstante, esto puede variarse a modo avanzado pulsando el botón *Advanced*. Nos centraremos en el modo simple.



Los botones ***open***, ***save*** y ***new*** sirven, respectivamente, para cargar, salvar o generar un fichero con los datos del experimento.

Lo primero a realizar es **abrir** (botón ***Open***) o **crear un fichero configuración** (botón ***New***) que contendrá todos los ajustes, ficheros involucrados, etc, pertenecientes a un experimento. Una vez creado podemos guardarlo (Weka por defecto no guarda nada), para lo que usaremos el botón ***Save***.

El siguiente paso es **decidir dónde queremos almacenar los resultados**, si es que queremos hacerlo. Los **resultados** del experimento pueden ser almacenados con tres formatos distintos: fichero arff, fichero CSV y en una base de datos. Para ello, simplemente se debe seleccionar la opción que corresponda dentro del cuadro *Results Destination.* Dicho fichero será usado posteriormente para analizar el experimento.

Tras indicar si el problema es de clasificación **(classification)** o de regresión **(regression)** (en esta práctica, se trata del primero), lo siguiente es **definir el tipo de validación que tendrá el experimento.** Al igual que en el modo *explorer*, tenemos tres tipos de validación:

- **Cross-validation:** validación-cruzada estratificada (por defecto de 10 hojas, aunque es modificable)
- **Train-Test Percentage Split (data randomized):** entrenamiento con un porcentaje de la población tomando ese porcentaje de forma aleatoria (desordena aleatoriamente los datos y después coge el primer 66% para construir el clasificador y el 34% restante para hacer el test)
- **Train-Test Percentage Split (order preserved):** entrenamiento con un porcentaje de la población tomando el porcentaje de forma ordenada (igual que antes pero no desordena el conjunto de datos antes de dividirlos en entrenamiento y test)

En el panel **Datasets** añadiremos (botón ***Add new…*)** los conjuntos de datos sobre los que realizaremos el experimento.

En el panel **Iteration Control** definiremos el **número de repeticiones** (veces que queremos que se ejecute cada uno de los algoritmos) **de nuestro experimento**, especificando si queremos que se realicen **primero los archivos de datos** (ejecutar para cada archivo de datos todos los algoritmos) o **primero los algoritmos** (ejecutar cada algoritmo sobre todos los archivos de datos).

El algoritmo que sigue el Experimenter es (si seleccionamos **algorithms first**, el bucle externo será el de los algoritmos, y el interno el de los datasets):

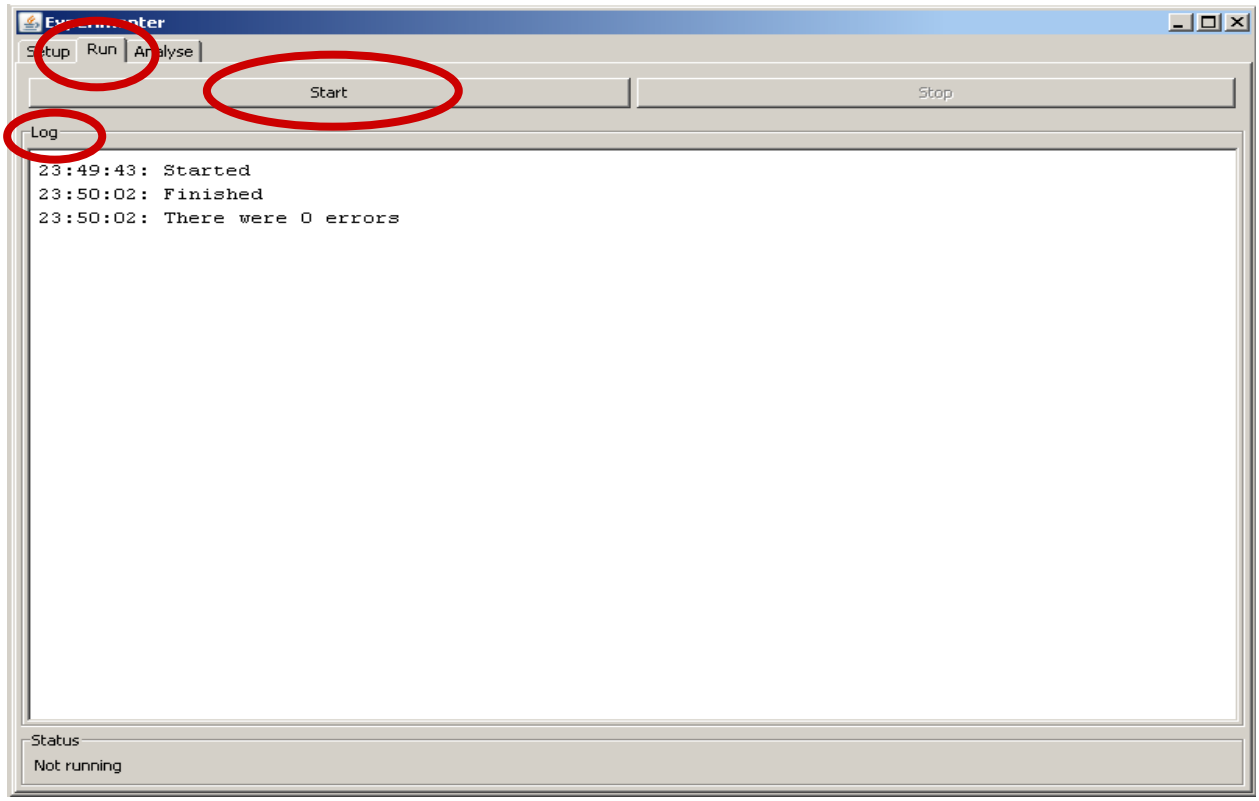
```
Repite para cada dataset
  Repite para cada algoritmo
    Repite “Number of iterations”
        Haz validación cruzada
```

A modo de ejemplo, si indicamos que el experimento se repita 5 veces y hemos seleccionado una validación cruzada de 10 hojas, cada uno de los algoritmos de aprendizaje serán ejecutados un total de 5 x 10 = 50 veces.

En el panel **Algorithms** se añaden los algoritmos a aplicar sobre el conjunto de datos del panel *Datasets*, **configurados convenientemente** según nos interese.

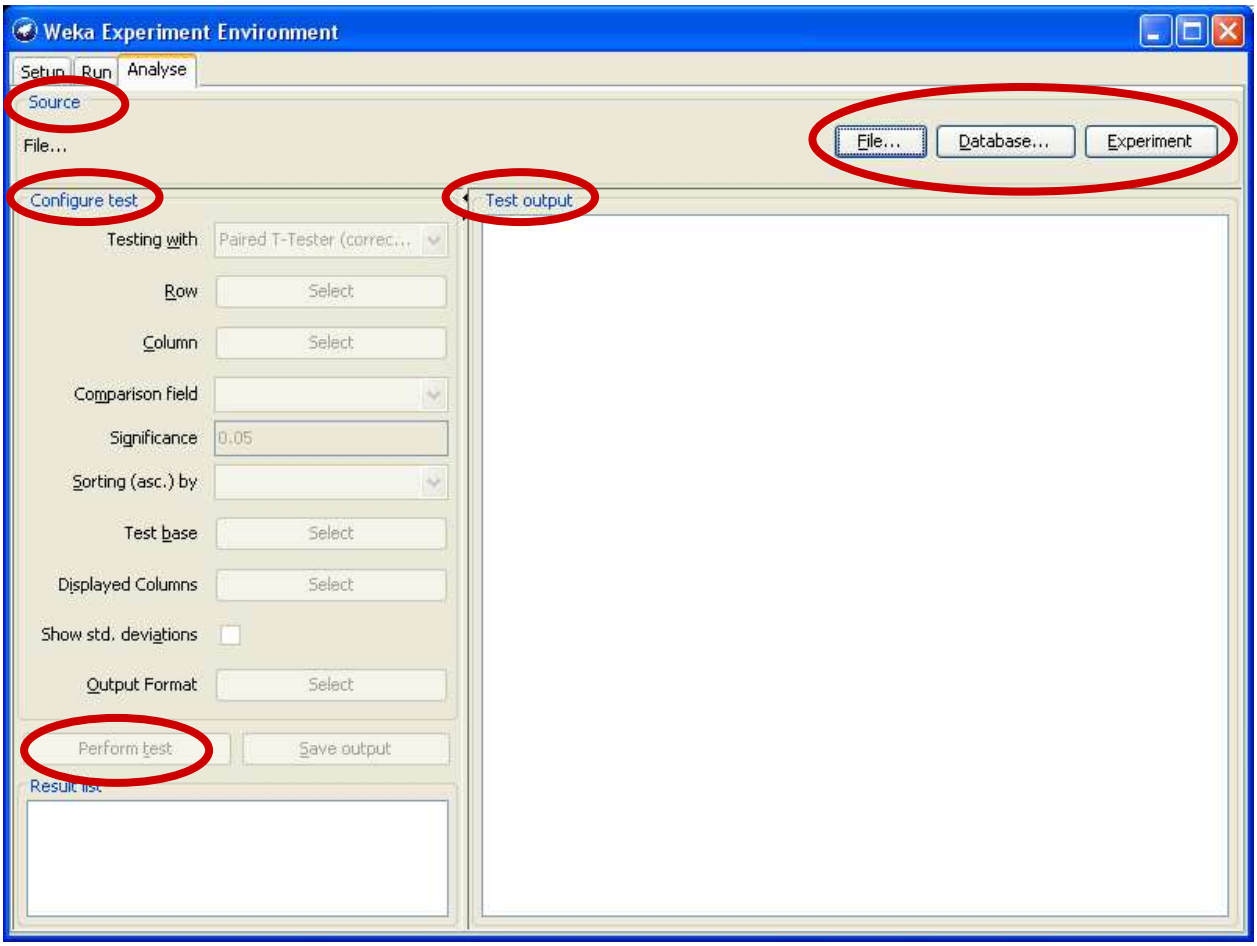
Si se desea se pueden añadir comentarios y notas al experimento; para ello, Weka dispone de un cuadro de texto (botón ***Notes***) en la parte inferior de la ventana.

Una vez configurado el experimento, para realizarlo se selecciona la pestaña ***Run***, donde se deberá pulsar el botón ***Start*** para realizar el experimento. Si todo va bien, se debe obtener un *Log* sin errores, tal y como aparece en la figura.



Los resultados del experimento se almacenarán en el fichero seleccionado en la pestaña ***Setup***, bajo el formato elegido.

Finalmente, para analizar los datos se selecciona la pestaña ***Analyse***, que permite, entre otras cosas, ver los resultados de los experimentos, realizar contrastes estadísticos, etc.



El primer paso será cargar los datos del experimento que se quiere analizar, con los botones ***File, Database o Experiment*** (este último procesa los resultados del experimento que está actualmente cargado). Una vez cargados, se visualiza en el panel ***Source*** la cantidad de resultados en dicho fichero.

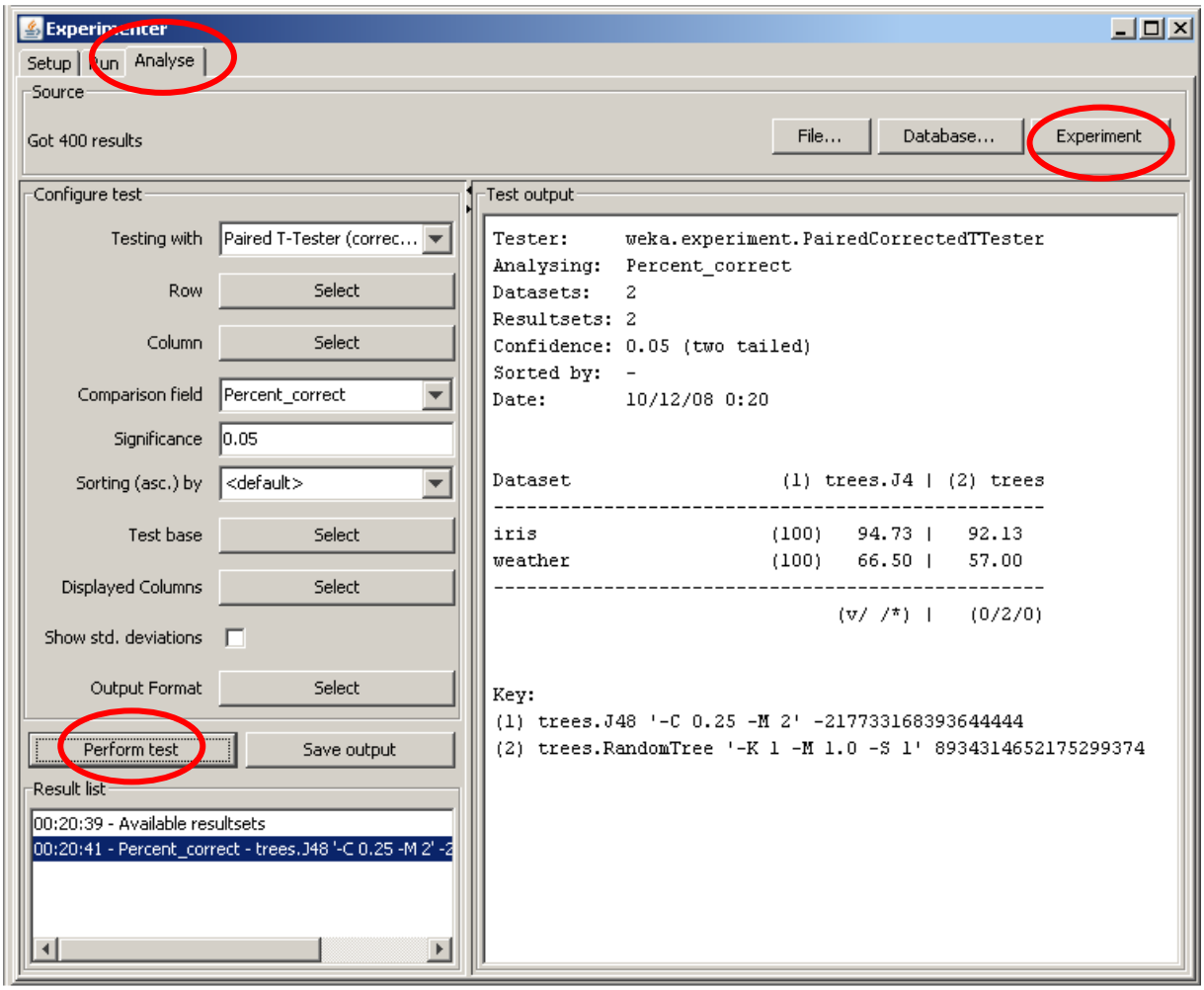
Para definir el test que se quiere realizar se seleccionan las opciones que ofrece el panel ***Configure test***. A continuación se realiza una breve descripción de las mismas:

- **Row key fields.** Selecciona el atributo o atributos que harán de filas en la matriz de resultados.
- **Run fields** Define el atributo que identifica cada una de las iteraciones.
- **Column key fields**. Selecciona los atributos que actuarán de columnas en la matriz de resultados.
- **Comparison fields**. El atributo que va a ser comparado en el contraste.
- **Significance.** Nivel de significación para realizar el contraste estadístico. Por defecto es el 0,05 (esto más o menos quiere decir que la probabilidad de que el Experimenter os diga que una diferencia es significativa cuando realmente es debida al azar, es del 5%)
- **Test base**. Seleccionamos qué algoritmo de los utilizados (o el resumen, o el ranking) utilizaremos de base para realizar el test.
- **Show std. Deviations**. Muestra las desviaciones estándar.

Una vez configurado el test, el botón ***Perform test*** realiza el test (T-Student) y muestra los resultados en el panel ***Test output***.

Por ejemplo, queremos realizar (***New***) sobre las bases de datos (***Datasets-Add new***) iris.arff y weather.numeric.arff (que incluye Weka en el directorio data) una comparación entre dos clasificadores (***Algorithms-Add new***), ambos mediante árboles de decisión, uno con C4.5 (J48 en Weka) y otro con RandomTreee para comprobar si hay diferencias significativas entre ambos métodos.

Para ello una vez realizado (***Run-Start***) el experimento, en el modo análisis de resultados (***Analyse-Experiment***), realizamos un test (***Perform test***) al 0,05 de confianza, comparando como atributo el porcentaje de acierto (***percent correct***) de ambos clasificadores.



Como se puede observar en las filas aparecen los Datasets (iris y weather), y en las columnas los algoritmos utilizados (J48 y RandomTree). El algoritmo con respecto al cual se hace la comparación (eso se puede cambiar), es el de más a la izquierda (J48).

En este caso, para el dominio ***iris***, J48 (función 1) obtiene un 94,73% de acierto frente a los 92,13% que se obtienen con RandomTree mientras que en ***weather*** los porcentajes son respectivamente 66,50% y 57,00%

Al lado de los resultados de cada algoritmo (excepto el que está más a la izquierda que es contra el que se comparan), puede aparecer una **v**, un *****, o **nada**:

- La **v** significa que la mejora es estadísticamente significativa.
- El ***** significa que el empeoramiento es estadísticamente significativo.
- Y si no aparece **nada**, que la diferencia no es estadísticamente significativa, es decir, que la diferencia podría ser debida al azar, y por tanto no podemos afirmar que un algoritmo sea mejor que el otro, o al contrario.

En este caso al no aparecer nada a la derecha de RandomTree significa que, aunque obtiene peores resultados que J48 en ambos problemas, ese empeoramiento no es significativo, por lo que desde el punto de vista estadístico no podemos decir que uno sea mejor que el otro (y por tanto ambos algoritmos tienen una precisión similar).

En la última línea de la tabla aparece un indicador del tipo:
(v/ /*) | (0/2/0)
(mejor/igual/peor)
siendo (xx,yy,zz), xx el número de veces que (en cuantos dominios o problemas) el algoritmo a comparar es significativamente mejor que el base, yy el número de veces que son iguales, y zz el número de veces que es peor (en este caso, RandomTree es igual en los dos problemas, y por tanto mejor, peor en 0 dominios).

Si queremos, podemos cambiar el algoritmo con el que se comparan todos los demás. Esto se puede hacer pulsando el botón **Test Base** y cambiando el algoritmo

Ejercicio para la primera parte

1. Usar el *Experimenter* para comparar los resultados de los algoritmos J48, 1R y el árbol seleccionado en la práctica anterior sobre la base de datos ***credit-g.arff, contact-lenses.arff, iris.arff*** y ***vehicle.arff***. Haz que cada algoritmo se ejecute 20 veces sobre cada conjunto de datos usando una validación cruzada de 5 hojas. Usar como base para el test, el algoritmo ZeroR, que selecciona la clase mayoritaria para realizar la clasificación (si el 80% de los datos son de la clase A, el 15% de la B y el 5% de la C, clasifica a todos como clase A). Es conveniente comparar contra este clasificador porque el % de aciertos que obtengamos con él es el que tendrá que superar el resto de clasificadores. Haz que en el test aparezca las desviaciones típicas con idea de ver las desviaciones con respecto a la media.

2. ¿Cómo Justifica los resultados obtenidos sobre la base de datos ***credit-g.arff***?

3. Repite el test usando como base Ranking ¿Cuáles son los dos mejores algoritmos?

4. Repite el experimento ejecutando cada algoritmo 20 veces sobre cada conjunto de datos usando para entrenamiento un 70% de los datos seleccionados al azar (el 30% restante para test). Usa como base para el test el algoritmo que mejor Ranking de en los experimentos.

5. Realizar un pequeño informe con las conclusiones obtenidas en los dos experimentos realizados

2. Segunda parte: Aprendizaje sensible al coste mediante el *Explorer*

Hay muchas ocasiones donde el coste de errar en la clasificación será diferente según el tipo de error cometido. Por ejemplo, en los entornos médicos, no es igual de costoso, detectar una enfermedad cuando ésta no existe, que enviar a casa a un paciente enfermo diagnosticándolo como sano. De la misma manera, en el entorno bancario, si tenemos un modelo que nos recomienda si conceder o no un crédito a un determinado cliente, es más costoso que el sistema se equivoque dando un crédito a una persona que no lo devuelve, que la situación contraria, denegárselo a uno que sí lo devolvería. Este tipo de costes están presentes en otros muchos problemas de clasificación

Para este tipo de problemas, la información sobre los costes de los errores viene expresada a través de una matriz de coste, en la que se recoge el coste de cada una de las posibles combinaciones entre la clase predicha por el modelo y la clase real.

Weka nos ofrece mecanismos para evaluar modelos con respecto al coste de clasificación, así como de métodos de aprendizaje basados en reducir el coste en vez de incrementar la precisión.

2.1 Evaluación sensible al coste

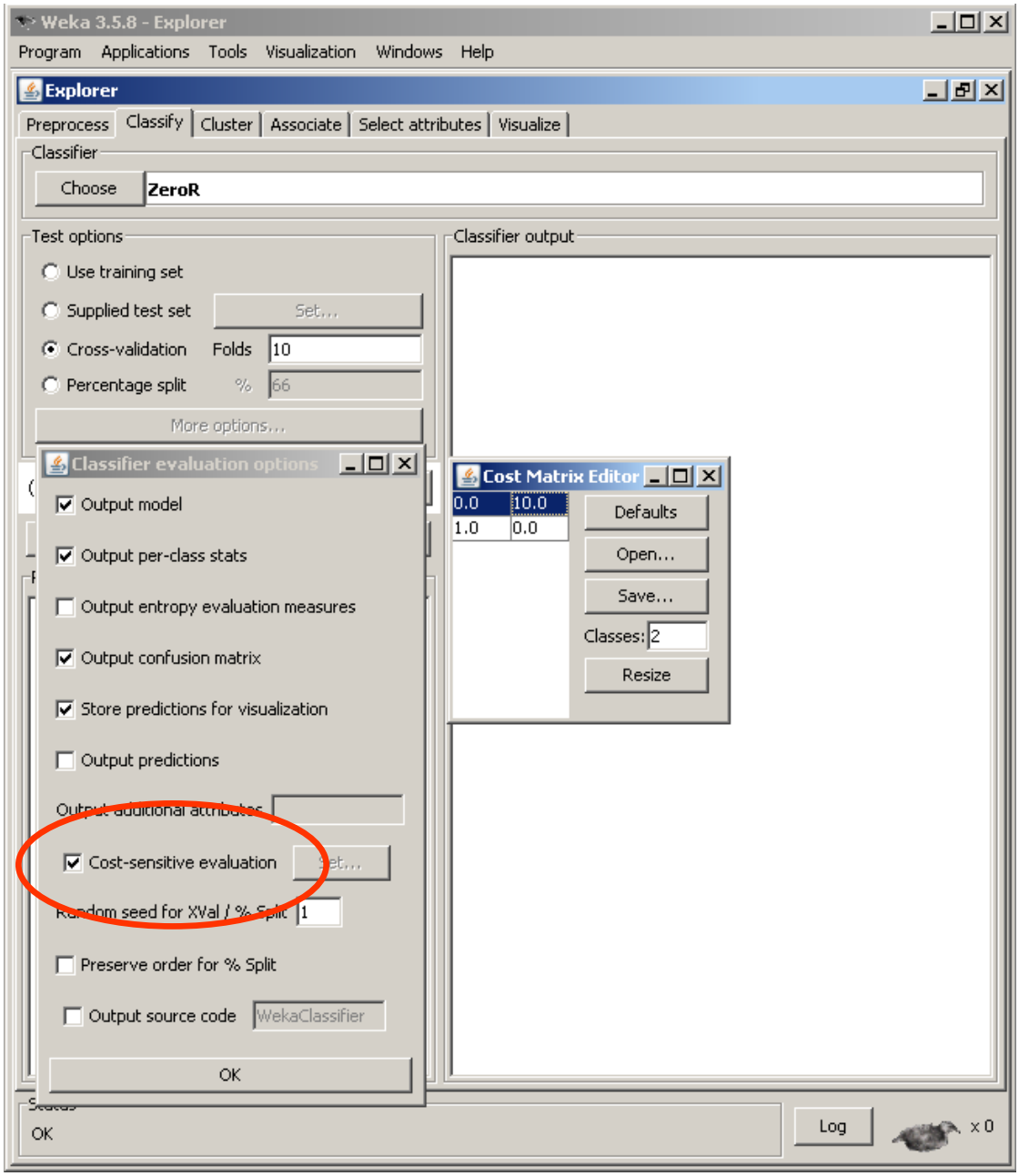
La evaluación sensible al coste nos permite, como su nombre indica, determinar el coste en el que incurrimos al realizar una determinada clasificación.

Para realizar la evaluación sensible al coste debemos primero seleccionar el conjunto de datos (***Preprocess-Open file***) y el clasificador (***Classify-Choose***) con el que deseemos trabajar. Una vez hecho esto, para evaluar los modelos con respecto a la matriz de coste pulsamos en la ventana ***Classify***, en ***Test options***, el botón ***More options***.

Habilitamos la opción ***Cost-sensitive evaluation***.

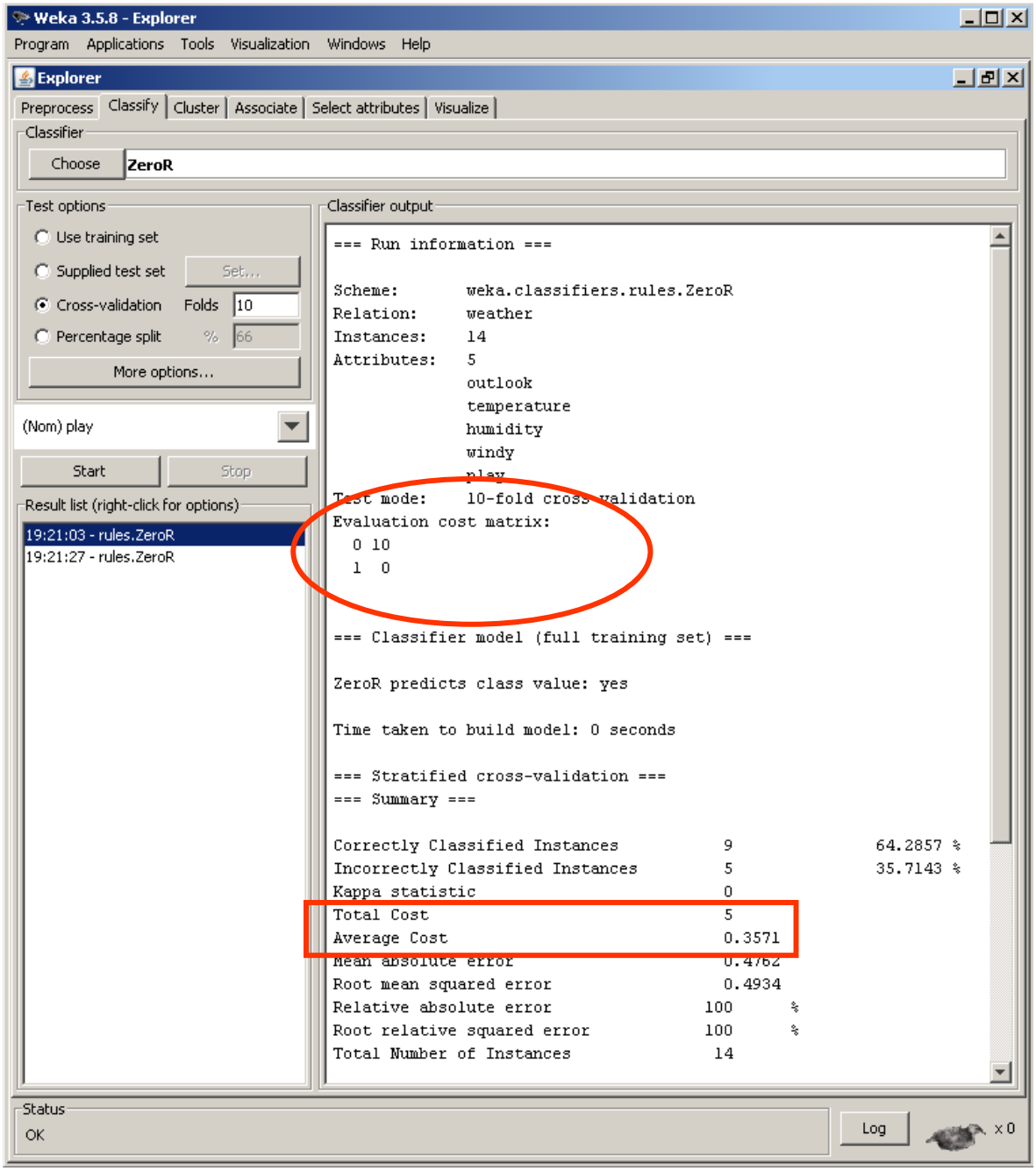
Pulsamos ***Set*** para introducir la matriz. Indicamos el número de clases e introducimos la matriz de costes. El número de clases vendrá dado por el conjunto de datos que se pretende clasificar.

Es importante pulsar '*intro'* tras modificar cada coste de la matriz.



Una vez definida la matriz de costes y una vez realizada la clasificación (***Start***), podremos observar en los resultados, la matriz de coste que se ha empleado y en las estadísticas aparecerá el coste total y el coste medio junto con el resto de medidas.

A modo de ejemplo, en la siguiente ventana se muestra los resultados que se obtienen y el coste en el que se incurre en el problema ***weather.numeric*** (días en los que se ha podido jugar al tenis dependiendo de ciertos aspectos metereológicos) al asignar siempre la clase mayoritaria (ZeroR) utilizando validación cruzada de 10 hojas considerando que es 10 veces más costoso anular un partido de tenis cuando en realidad si se podría haber jugado, que jugar un partido de tenis y tener que interrumpirse por circunstancias meteorológicas.



Se obtiene una precisión del 64.28% y un coste de 5 unidades, ya que de las 5 instancias del problema mal clasificadas éstas se corresponden al hecho de predecir que el partido se puede celebrar y luego no es posible celebrarlo por las circunstancias meteorológicas como se puede apreciar en la matriz de confusión resultante:

=== Confusion Matrix ===

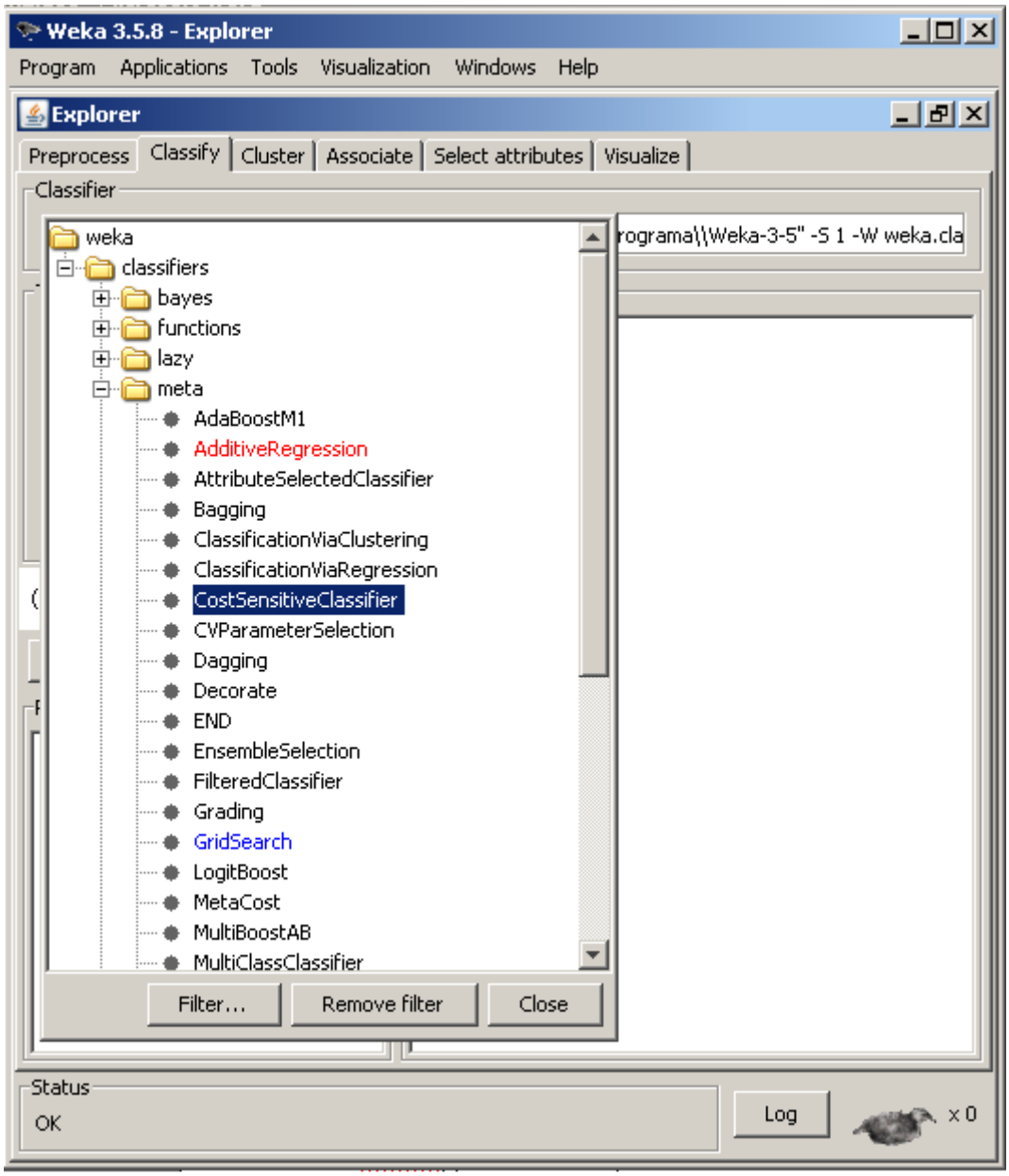
a b <-- classified as
9 0 | a = yes
5 0 | b = no

2.2 Aprendizaje sensible al coste

En la construcción del modelo anterior no hemos tenido en cuenta los costes introducidos en la matriz de costes (sólo lo hemos utilizado para calcular cuál es el coste en el que hemos incurrido al realizar la clasificación con el método elegido).

Por tanto el modelo obtenido no es dependiente del coste ni se ha tenido en cuenta para la construcción del mismo, por lo que se ha construido intentado reducir el número de errores, no el coste de los mismos.

Weka, ofrece varias técnicas que permiten adaptar la clasificación a un contexto con costes. Para ello, seleccionamos (***Choose***) la opción *CostSensitiveClassifier* en *Meta*, en lugar de un algoritmo de clasificación. Este paquete ofrece dos técnicas bastante sencillas para que un método de aprendizaje sea sensible al contexto.



La primera versión consiste en que los modelos aprendidos asignen las clases de manera que se minimicen los costes de clasificación errónea. Normalmente, esta asignación se realiza de manera que se reduzcan los errores en la clasificación.

La segunda técnica consiste en modificar los pesos asignados a cada clase de manera que se da más importancia a los ejemplos que son susceptibles de cometer los errores más costosos.

Para optar entre estos métodos utilizamos la opción *MinimizeExpectedCost*. Si fijamos esa opción como cierta, se utiliza la primera técnica (asignación de clase de coste mínimo), si la dejamos como falsa, se utiliza la segunda (compensar los costes mediante los pesos de los ejemplos).

Así pues, para realizar aprendizaje sensible al coste, en primer lugar seleccionaremos el conjunto de datos a clasificar. A continuación seleccionamos de la carpeta *Meta* la opción *CostSensitiveClassifier*.

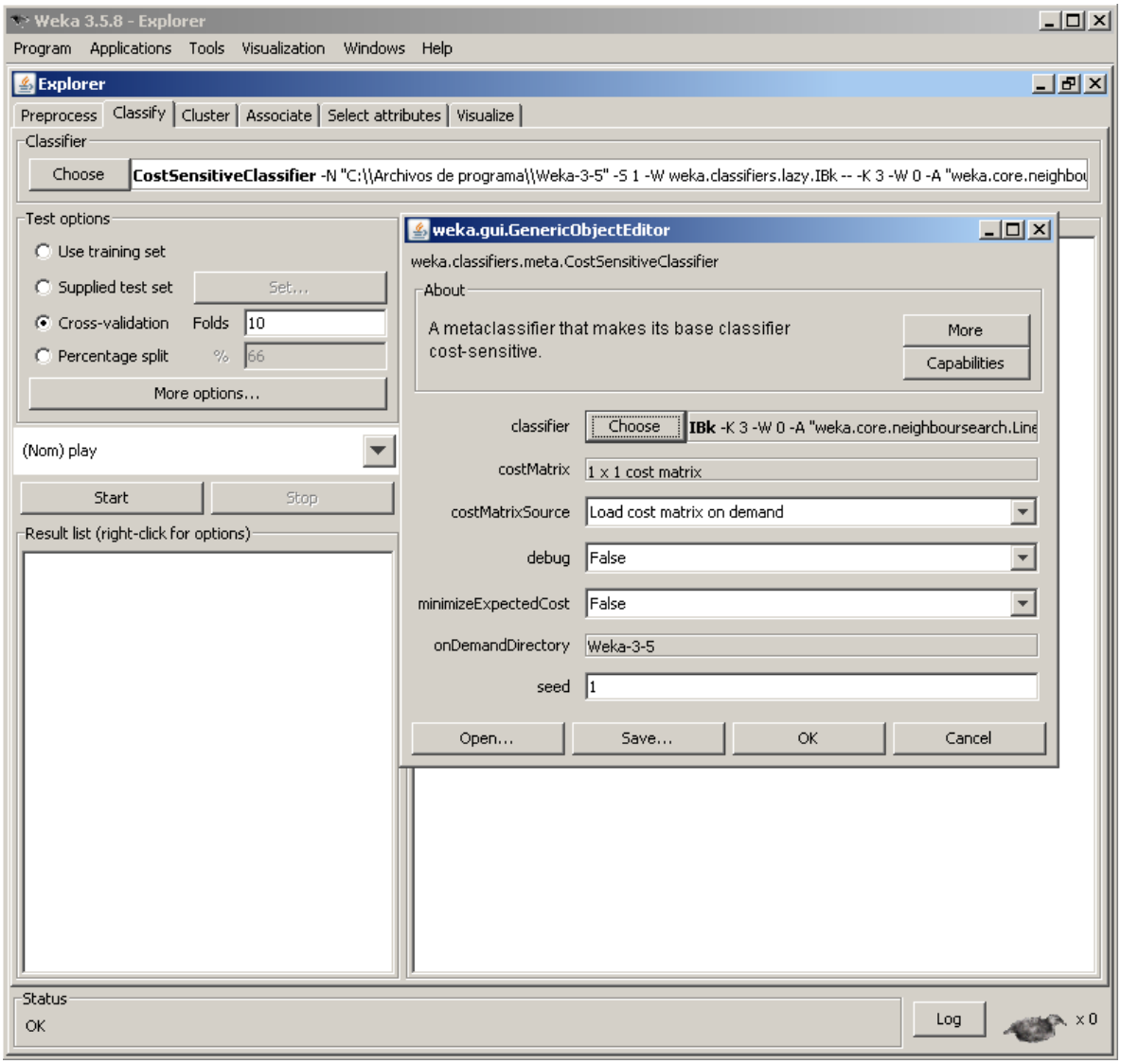
Configuramos las opciones, que son:

- *Clasificador que se emplea*. Podremos seleccionar un clasificador de entre los disponibles en Weka.

- *Matriz de costes*. Indicaremos la matriz de costes

- *Ubicación de la matriz de costes*. Podemos seleccionar si se usa la matriz de costes introducida anteriormente o alguna disponible en un fichero guardado. La ubicación del fichero será la indicada en la opción *onDemandDirectory*.

- *minimizeExpectedCost*. Esta opción configurará el clasificador según se ha indicado anteriormente.



Ejercicio para la segunda parte

En este ejercicio volveremos a usar la base de datos ***credit-g.arff***. Esta base de datos trae información de los costes de clasificación errónea, en concreto la siguiente matriz de coste:

		PREDICHA	
		SI	**NO**
REAL	**SI**	0	1
	NO	5	0

Esta tabla indica que es 5 veces más costoso si se otorga un crédito a una persona que no lo devuelve, que la situación contraria.

1. Realizar una clasificación con la asignación siempre la clase mayoritaria (*ZeroR*) y comprobar el coste asociado a dicho modelo, usando 10-validación cruzada. ¿qué precisión y costes se obtienen?

2. Volver a realizar la clasificación usando esta vez como clasificador el método *IBK* (los *k* vecinos más cercanos). Ajustar el valor de k hasta obtener una precisión que iguale o supere la precisión anterior. ¿Qué valor de k se consigue, y que costes y precisión se obtienen?

3. Repita la clasificación usando ahora otro método, por ejemplo Naive Bayes. ¿qué precisión y costes se obtienen?

4. Para todos ellos muestra la matriz de confusión y desglosa los cálculos de costes

Con esto lo único que hacemos es constatar el coste asociado a la precisión obtenida según la matriz de costes. Vamos a realizar ahora un aprendizaje basado en costes.

5. Usar el paquete *CostSensitiveClassifier* para realizar el aprendizaje sensible al coste, dejando *ZeroR* como clasificador base. Además hemos de indicarle al método la matriz de coste que ha de utilizar en el aprendizaje, fijando *CostMatrixSource* como *Use Explicit Matriz Cost*. Y en *Cost Matriz* introducimos la misma matriz de coste que hemos utilizado para la evaluación. ¿Qué precisión y costes se obtienen? ¿Cómo se explica?

6. Volver a usar el paquete *CostSensitiveClassifier* pero usando esta vez como clasificador base a *IBK*, con los parámetros del apartado 2. ¿Qué conclusiones pueden obtenerse de los resultados?

7. Repítelo de nuevo usando Naive Bayes como clasificador base e indica qué resultados se obtienen al aprender el modelo asignado las clases de manera que se minimicen los costes de clasificación errónea y qué resultados se obtienen al modificar los pesos asignados a cada clase.

8. Realiza una tabla resumen comparativa en la que se muestre la precisión obtenida y el coste asociado con cada uno de los métodos utilizados (basados y no en coste).

¿Cómo entregar la práctica?

- Utilizar un documento de texto para responder a las cuestiones y subirlo a través de la plataforma web