

Manual de usuario del “Sistema de información para el estudio de metaheurísticas aplicadas a selección de atributos”

**Edición: 03
Fecha: 19 de septiembre
de 2022**

Índice

1	INTRODUCCIÓN	3
2	DESCRIPCIÓN DE LA INTERFAZ	3
	2.1 Pestaña Experiment	4
	2.1.1 Datasets	4
	2.1.2 Feature Selection	5
	2.1.3 Classifier	5
	2.1.4 Validation	6
	2.1.5 Ejecución	7
	2.2 Pestaña Results	7
	2.2.1 Predictions	7
	2.2.2 Metrics	9
	2.2.3 Statistical Tests	10
	2.2.4 Graph	11
3	CASO DE USO PRINCIPAL	12
	3.1 Experimentación	12
	3.2 Resultados	19
4	CONFIGURACIÓN DATABASEUTILS	27

1 INTRODUCCIÓN

La aplicación es un sistema de información que permite realizar estudios de selección de atributos en conjuntos de datos diseñados para aprendizaje supervisado. Para ello, se proporcionan algoritmos de optimización basados en metaheurísticas como algoritmos de búsqueda para encontrar los subconjuntos de atributos más relevantes, para ello se hace uso de la herramienta de análisis libre Weka. Se han incluido test estadísticos para comprobar la veracidad de los resultados obtenidos con los distintos algoritmos, concretamente, tests bayesianos, el cual se encuentra en un archivo Python (ya que la librería es de este lenguaje de programación) y con el cual se ha realizado una conexión desde Java.

Además, se ha llevado a cabo la integración del algoritmo CVOA desarrollado por el departamento de Data Science & Big Data de la Universidad Pablo de Olavide para tener un algoritmo más de optimización.

Al realizar un experimento, se permiten añadir y eliminar datasets así como modificar su classIndex y su clase positiva, también se permite su carga de nuevo en la pestaña Preprocess. Por otro lado, también se permite la gestión de algoritmos de búsqueda e integración, así como algoritmos de clasificación/regresión. También se permite elegir el método de validación (hold-out split, cross validation y leave-one out) y configurarlos (porcentaje destinado a entrenamiento en hold-out Split y número de bolsas en cross validation). Nos permite también ejecutar la experimentación con el número de hilos que queramos haciendo así la ejecución de forma paralela. Dicho progreso se muestra por una barra en la parte inferior.

Una vez se ha realizado la experimentación y obtenidos los datos, podemos ver dichos resultados agrupados en 4 componentes: Predictions (donde se muestran las predicciones con su valor real y predicho junto a la combinación a la que pertenece), Metrics (donde podemos elegir las métricas a mostrar al lado de cada combinación eligiendo entre 12 métricas de clasificación y regresión), Graph (donde nos permite visualizar una métrica a elegir en función del dataset, evaluador, buscador o clasificador) y Statistical tests (donde aparece información relativa a los tests estadísticos así como sus resultados). También, se incluye distintas formas de exportar los resultados como puede ser formato CSV, ARFF, etc, incluso a una DB relacional desde la cual también se pueden cargar los datos exportados anteriormente.

Todas y cada una de las funcionalidades se explicitarán y detallarán en los puntos siguientes.

2 DESCRIPCIÓN DE LA INTERFAZ

La interfaz desarrollada se compone de dos pestañas: Experiment y Results.

En la pestaña Experiment tenemos 4 componentes más el perteneciente a la ejecución de la experimentación, estos componentes son:

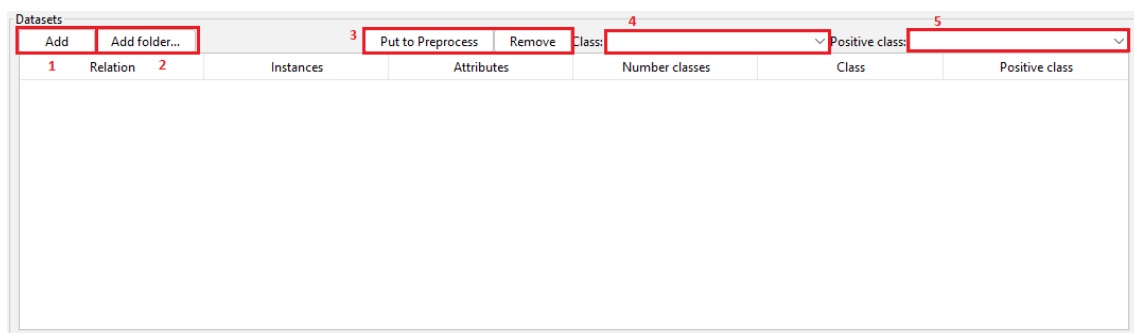
- Datasets
- Feature Selection
- Classifier
- Validation

2.1 Pestaña Experiment

2.1.1 Datasets

La sección Datasets está formada por los siguientes elementos:

1. **Botón Add:** añade el dataset previamente cargado en la pestaña de Preprocess de Weka a la tabla de Datasets y como tal, también a la experimentación a ejecutar.
2. **Botón Add folder:** añade todos los datasets que estén presentes en la carpeta que se seleccione en formato arff a la tabla de Datasets y como tal, también a la experimentación a ejecutar.
3. **Botones Actions:** el botón Put to Preprocess carga el dataset previamente seleccionado en la tabla en la pestaña Preprocess de Weka. El botón Remove elimina el dataset seleccionado previamente de la tabla, así como de la experimentación.
4. **ComboBox Class:** despliega los posibles valores de classIndex del dataset seleccionado en la tabla pudiendo de esta forma setear su valor actual. Al seleccionar un dataset en la tabla, el valor de dicho comboBox se setea al que posea el dataset seleccionado.
5. **ComboBox Positive Class:** despliega los posibles valores de positive class del dataset seleccionado en la tabla pudiendo de esta forma setear su valor actual. Al seleccionar un dataset en la tabla, el valor de dicho comboBox se setea al que posea el dataset seleccionado.
6. **Tabla:** tabla donde aparecerán los datasets añadidos a la experimentación con el botón Add. Sus campos son:
 - **Relation:** nombre del dataset.
 - **Instances:** número de filas del dataset.
 - **Attributes:** número de atributos del dataset.
 - **Number classes:** número de clases del dataset.
 - **Class:** indexClass actualmente seleccionado para el dataset.
 - **Positive class:** clase positiva actualmente seleccionada para el dataset.



2.1.2 Feature Selection

La sección Feature Selection está formada por los siguientes elementos:

1. **Botón Choose Evaluator:** al hacer click despliega un menú donde podemos elegir el algoritmo de evaluación que queramos entre los disponibles. Si queremos cambiar las opciones de dicho algoritmo, debemos hacer click en su textField y saltará un menú con sus opciones donde podremos cambiarlas.
2. **Botón Choose Search:** al hacer click despliega un menú donde podemos elegir el algoritmo de búsqueda que queramos entre los disponibles. Si queremos cambiar las opciones de dicho algoritmo, debemos hacer click en su textField y saltará un menú con sus opciones donde podremos cambiarlas.
3. **Botón Add:** añade los algoritmos previamente elegidos, configurados y cargados en sus respectivos textFields a la tabla, y, por consiguiente, a la experimentación.
4. **Botones Actions:** el botón Load carga la fila previamente seleccionada de la tabla (es decir, una combinación de Evaluator con Search) en sus respectivos textFields. El botón Remove elimina la fila seleccionada previamente de la tabla, así como de la experimentación.
5. **Tabla:** tabla donde aparecerán los algoritmos de evaluación y búsqueda añadidos a la experimentación con el botón Add. Sus campos son:
 - **Evaluator:** nombre y opciones del evaluador añadido.
 - **Search:** nombre y opciones del algoritmo de búsqueda añadido.

Feature Selection

Evaluator: 1 Choose CfsSubsetEval -P 1 -E 1

Search: 2 Choose BestFirst -D 1 -N 5

3 Add Actions: 4 Load Remove

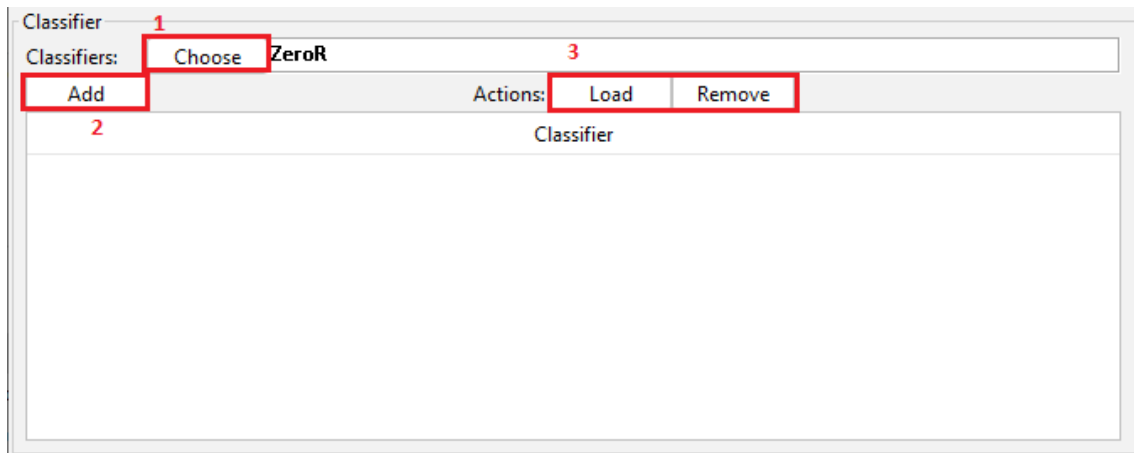
Evaluator	Search
-----------	--------

2.1.3 Classifier

La sección Classifier está formada por los siguientes elementos:

1. **Botón Choose:** al hacer click despliega un menú donde podemos elegir el algoritmo de clasificación/regresión que queramos entre los disponibles. Si queremos cambiar las opciones de dicho algoritmo, debemos hacer click en su textField y saltará un menú con sus opciones donde podremos cambiarlas.

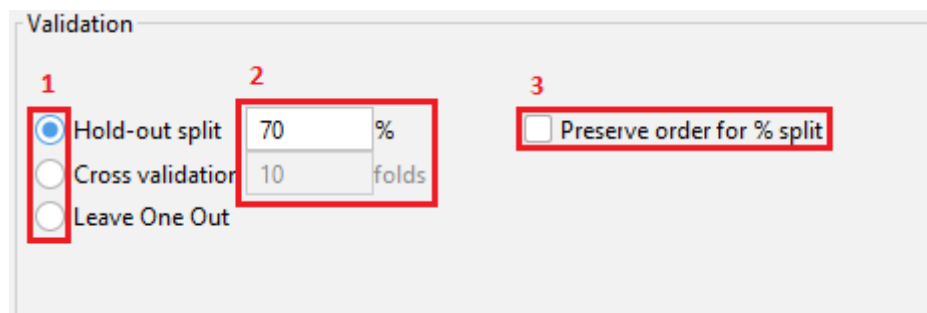
2. **Botón Add:** añade el algoritmo previamente elegido, configurado y cargado en su textField a la tabla, y, por consiguiente, a la experimentación.
3. **Botones Actions:** el botón Load carga la fila previamente seleccionada de la tabla en su textField. El botón Remove elimina la fila seleccionada previamente de la tabla, así como de la experimentación.
4. **Tabla:** tabla donde aparecerán los algoritmos de búsqueda/regresión añadidos con el botón Add previamente. Sus campos son:
 - **Classifier:** nombre y opciones del algoritmo de clasificación/regresión añadido.



2.1.4 Validation

La sección Validation está formada por los siguientes elementos:

1. **Radio buttons:** nos permite elegir el método de validación que queremos aplicar a la experimentación. Sólo se puede uno, y al elegirse dicho método, si posee configuración posible, su textField correspondiente se habilitará.
2. **TextFields de configuración:** nos permite configurar (en caso de haberlos seleccionados) las opciones disponibles para el método Hold-out Split (porcentaje destinado a entrenamiento), Cross validation (número de bolsas) y Preserve order Split (mantener el orden de las instancias en los datasets al hacer hold-out split).
3. **Checkbox:** nos permite mantener el orden de las instancias de los datasets al realizar la separación en la validación hold-out split.



2.1.5 Ejecución

La sección Ejecución está formada por los siguientes elementos:

1. **TextFields hilos:** nos permite configurar la cantidad de hilos con la cual se ejecutará la experimentación de forma paralela.
2. **Botones ejecución:** el botón Run comienza a ejecutar la experimentación, la cual puede detenerse con el botón Stop.
3. **Barra de progreso:** nos muestra el progreso actual de la ejecución, en caso de pararla, dicho valor se setea a cero.
4. **Guardar y cargar experimento:** nos muestra la configuración correspondiente para guardar los datos de un experimento (todo aquello que se ve en la pestaña Experiment). Podemos introducir el nombre del experimento y después, cargarlo seleccionado el lugar donde lo hayamos guardado previamente.



2.2 Pestaña Results

2.2.1 Predictions

La sección Predictions está formada por los siguientes elementos:

1. **Nomenclature:** muestra la leyenda de los distintos algoritmos utilizados (tanto de evaluación como de búsqueda como de regresión/clasificación).
2. **Listas:** listas donde aparecen en cada una de ellas los datasets, algoritmos de evaluación y búsqueda y algoritmos de clasificación/regresión respectivamente, con los cuales se ha llevado a cabo la experimentación. Además, aparecerá también el valor "original" en las listas de Evaluators y Search referentes a la combinación que no utiliza algoritmos de búsqueda ni de evaluación. Permite seleccionar y deseleccionar valores los cuales actúan como filtro.
- 3.
4. **TextField Attributes:** permite introducir los atributos que queramos que se muestren en la tabla como una columna más de los diferentes datasets que aparezcan (su valor puede ser vacío).
5. **Botones de acciones:** botones que nos permiten exportar e importar datos y actualizar la tabla, estos son:
 - **Save to CSV:** permite exportar los datos contenidos en la tabla en formato CSV al directorio elegido.
 - **Save to ARFF:** permite exportar los datos contenidos en la tabla en formato ARFF al directorio elegido.

- **Save to DB:** permite exportar los datos contenidos en la tabla de Predictions y en la tabla de Metrics (exporta los valores de todas las métricas para las combinaciones estén elegidas o no) a una base de datos con los parámetros indicados en el archivo DatabaseUtils.props (su configuración se verá más adelante). Si la base de datos o cualquier tabla no está previamente creada, se creará automáticamente mediante el uso de scripts sqls.
- **Put to Preprocess:** permite la carga de los datos de la tabla en la pestaña Preprocess de Weka.
- **Load DB:** permite cargar los datos previamente guardados en la base de datos a las tablas Predictions y Metrics.
- **Update:** actualiza la tabla con los datasets, algoritmos de evaluación y búsqueda y con los algoritmos de clasificación/regresión seleccionados previamente en sus respectivas listas. Además, añade los atributos indicados en el textField Attributes a la tabla en forma de columnas.

6. **Tabla:** tabla donde aparecerán los resultados obtenidos de la experimentación referentes a las predicciones. Sus campos son:

- **Actual value:** muestra el valor original de dicha instancia en el dataset.
- **Predicted value:** muestra el valor predicho para dicha instancia.
- **Dataset:** nombre del dataset al cual pertenece.
- **Evaluator:** nombre del evaluador junto con sus opciones con el cual se ha obtenido ese valor.
- **Search:** nombre del algoritmo de búsqueda junto con sus opciones con el cual se ha obtenido ese valor.
- **Classifier:** nombre del algoritmo de clasificación/regresión junto con sus opciones con el cual se ha obtenido ese valor.
- **Attributes i (opcional):** cada columna correspondiente a los atributos introducidos en el textField Attributes. Cada fila indica el valor de dicho atributo para la instancia de dicha fila.

The screenshot shows the FS-Studio interface. At the top, there's a 'Predictions' tab (labeled 5) with a table header: Actual, Predicted, Dataset, Evaluator, Search, Classifier. To the right, there's a 'Nomenclature' panel (labeled 1) with a table for Name and Definition. Below that is a panel (labeled 2) with tabs for Datasets, Evaluators, Searches, and Classifiers. At the bottom, there's a row of buttons: Save to CSV, Save to ARFF, Save to DB, Load DB, Put to Preprocess (labeled 4), and a panel (labeled 3) with an 'Attributes:' text field and an 'Update' button.

2.2.2 Metrics

La sección Metrics está formada por los siguientes elementos:

1. **Classification:** métricas referentes a la clasificación. Cada métrica tiene asociada un checkBox el cuál, al ser pulsado inserta la columna correspondiente a dicha métrica en la tabla.
2. **Regression:** métricas referentes a la regresión. Cada métrica tiene asociada un checkBox el cuál, al ser pulsado inserta la columna correspondiente a dicha métrica en la tabla.
3. **Botones de acciones:** botones que nos permiten exportar y cargar datos de la tabla, estos son:
 - **Save to CSV:** permite exportar los datos contenidos en la tabla en formato CSV al directorio elegido.
 - **Save to ARFF:** permite exportar los datos contenidos en la tabla en formato ARFF al directorio elegido.
 - **Save to Latex:** permite exportar los datos contenidos en la tabla en formato LATEX al directorio elegido.
 - **Put to Preprocess:** permite la carga de los datos de la tabla en la pestaña Preprocess de Weka.
4. **Tabla:** tabla donde aparecerán los resultados obtenidos de la experimentación referentes a las métricas para cada combinación (experimento). Sus campos son:
 - **Dataset:** nombre del dataset al cual pertenece.
 - **Evaluator:** nombre del evaluador junto con sus opciones con el cual se ha obtenido ese valor.
 - **Search:** nombre del algoritmo de búsqueda junto con sus opciones con el cual se ha obtenido ese valor.
 - **Classifier:** nombre del algoritmo de clasificación/regresión junto con sus opciones con el cual se ha obtenido ese valor.

The screenshot shows the 'Metrics' window with the following components:

- Table Headers:** Dataset, Evaluator, Search, Classifier, NumAttrib...
- Classification Metrics (Group 1):** Accuracy, Precision, Recall, F-measure, Kappa, MCC, AUC.
- Regression Metrics (Group 2):** MAE, MSE, RMSE, MAPE, R2.
- Action Buttons (Group 3):** Save to CSV, Save to ARFF, Save to Latex, Put to Preprocess.
- Table Area (Group 4):** A large empty table for displaying results.

2.2.3 Statistical Tests

La sección Statistical Tests está formada por los siguientes elementos:

1. **Metric:** comboBox para elegir la métrica con la que se calcularán los tests estadísticos. Las métricas a mostrar dependerán de los datasets elegidos al hacer la experimentación (clasificación o regresión). Por defecto, se calcularán los tests estadísticos con la métrica F-measure para clasificación y MAE para regresión. En cuanto se cambie la métrica, se volverán a calcular los tests con la métrica elegida.
2. **Number of datasets:** número de datasets con el que se realiza la experimentación.
3. **Number of samples:** número de ejemplos que contienen los tests estadísticos.
4. **Groups for leave-one-out:** número de grupos que se harán para obtener las métricas y las cuales servirán para hacer los tests estadísticos. Por defecto 30, sólo aplica en el caso de LOO.
5. **Rope:** the region of practical equivalence (rope) is specified by the caller and should correspond to what is “equivalent” in practice. Por defecto será 1, en cuanto su valor cambie y se presione “Enter” se calcularán los tests con ese nuevo valor.
6. **Tabla:** tabla donde aparecen los resultados de los tests estadísticos. Los resultados van comparados con otras combinaciones, de ahí la duplicidad de de las columnas Evaluator, Search y Algorithm. Sus campos son:
 - **Evaluator:** evaluador correspondiente a esa combinación.
 - **Search:** algoritmo de búsqueda utilizado en esa combinación
 - **Algorithm:** algoritmo de clasificación/regresión utilizado en esa combinación.
 - **p(left):** probabilidad del primer conjunto (evaluator #1, search #1, algorithm #1).
 - **p(no):** probabilidad de ninguno mejor.
 - **p(right):** probabilidad del segundo conjunto (evaluator #2, search #2, algorithm #2).
 - **Test:** indica cual combinación de algoritmos ha sido mejor, 1 en caso de la primera combinación (indicada con #1), 2 en caso de la segunda (indicada con #2) o 0 en caso de que ninguna de las dos es significativamente mejor que la otra.
7. **Filter:** nos permite filtrar los resultados obtenidos en la tabla por los valores de la columna Test, pudiendo mostrar todos (All, es el valor por defecto), sólo los positivos (Positives, valores 1 y 2) o sólo los negativos (Negatives, valor 0).
8. **Botones de acciones:** botones que nos permiten exportar y cargar datos de la tabla, estos son:
 - **Save to CSV:** permite exportar los datos contenidos en la tabla en formato CSV al directorio elegido.
 - **Put to Preprocess:** permite la carga de los datos de la tabla en la pestaña Preprocess de Weka.

2. **Botones de acciones:** botones que nos permiten exportar y crear la gráfica, estos son:
 - **Save to PNG:** exporta la gráfica al directorio elegido en formato PNG.
 - **Save to PDF:** exporta la gráfica al directorio elegido en formato PDF.
 - **Update:** crea una nueva gráfica sustituyendo a la que había previamente, configurada con los valores indicados en los comboBoxes Metric, X-axis (1), X-axis (2), Filter (1) y Filter (2).
3. **Gráfica:** lugar donde aparecerá la gráfica una vez generada.



3 CASO DE USO PRINCIPAL

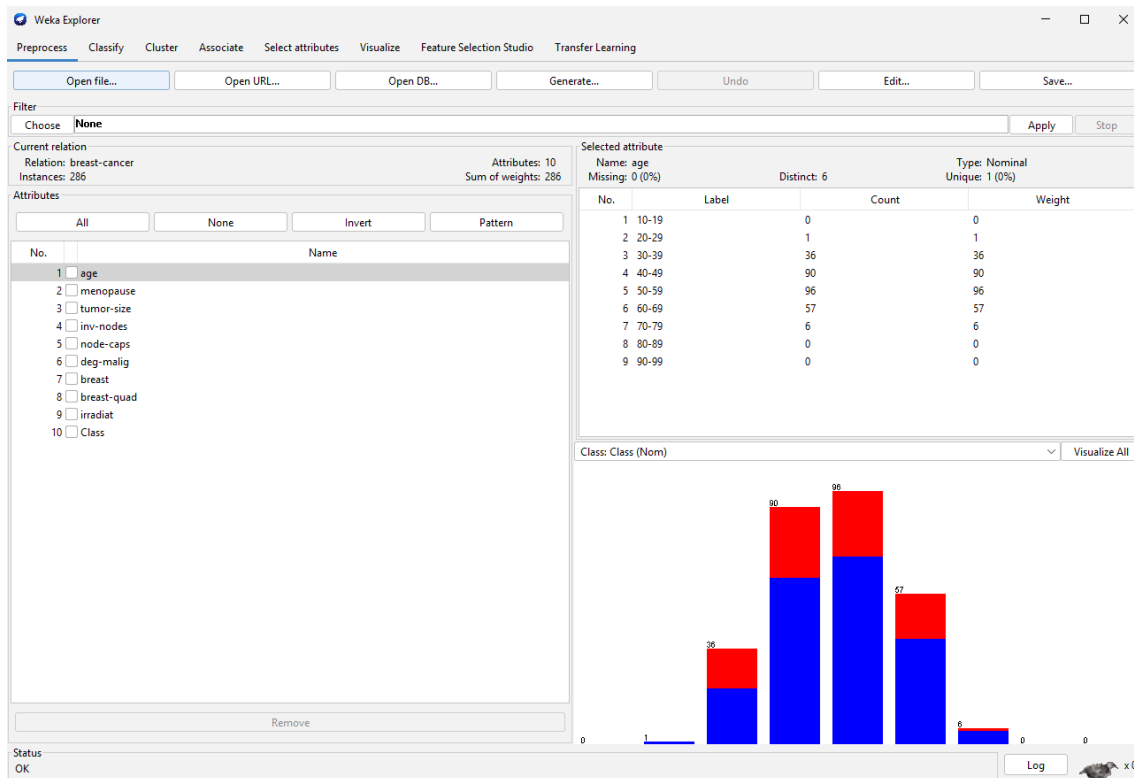
Una vez explicado y detallado cada una de las diferentes partes de las interfaces, y habiendo instalado previamente el paquete FeatureSelectionStudio en Weka, se muestra un ejemplo del flujo normal de un usuario al utilizar el sistema de información añadiendo varios conjuntos de datos, junto a varios algoritmos de búsqueda, evaluación, clasificación, así como eligiendo un método de evaluación y la cantidad de hilos con la cual se ejecutará la experimentación. También, veremos los resultados, los filtraremos, añadiremos nuevas columnas, exportaremos y cargaremos datos y generaremos gráficas:

3.1 Experimentación

PASO 1: Datasets

Lo primero que debemos hacer es en la pestaña Preprocess, cargar el dataset que queramos añadir al experimento. Mostraré los pasos para añadir 1 ya que con el resto es exactamente igual.

En Preprocess pulsamos en OpenFile, se nos abrirá el menú de directorios para seleccionar el dataset, en mi caso seleccionaré breast-cancer y le damos a abrir, procediendo así a su carga en la pestaña Preprocess:



Ahora pasamos a la pestaña FeatureSelectionStudio, lo que queremos hacer es añadir el dataset que acabamos de cargar a la experimentación, por lo que le damos a Add en la sección Datasets y vemos como se añade a la tabla:

Datasets

Relation	Instances	Attributes	Number classes	Class	Positive class
breast-cancer	286	9	2 Class		no-recurrence-events

Classifiers

Classifiers: Choose **ZeroR**

Feature Selection

Evaluator: Choose **CfsSubsetEval** - P 1 - E 1

Search: Choose **BestFirst - O 1 - N 5**

Validation

Hold-out split: 70 %

Cross validation: 10 folds

Leave One Out

Threads: 4

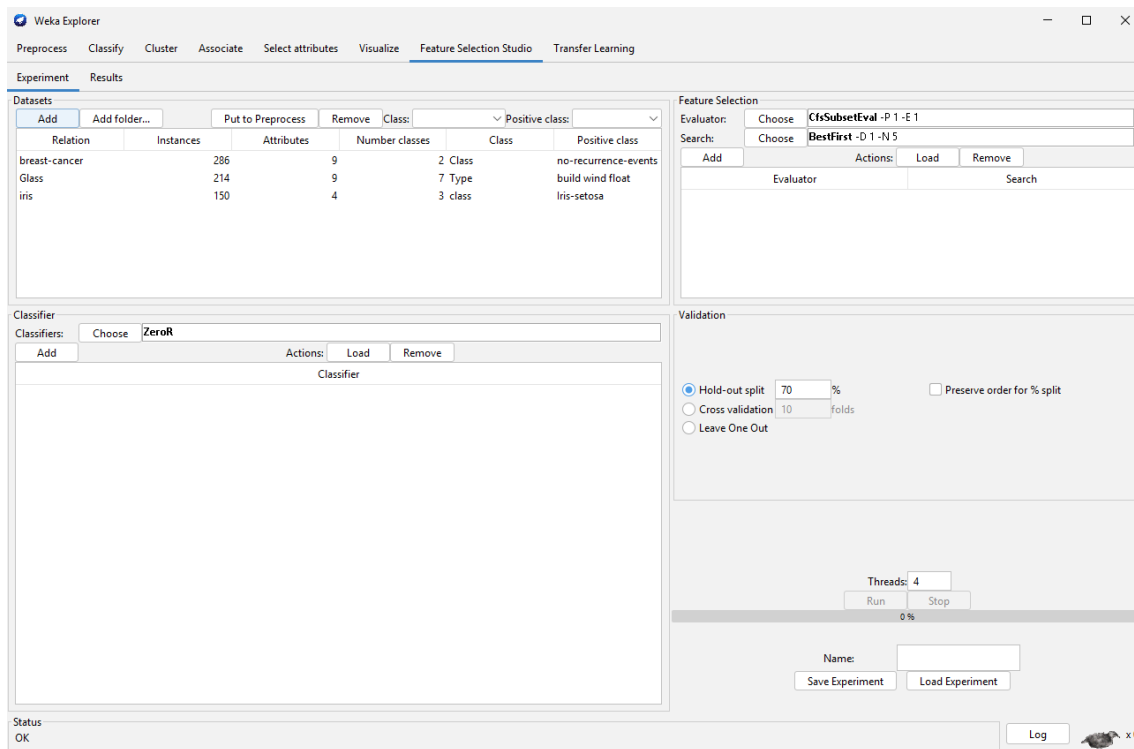
Run Stop

0 %

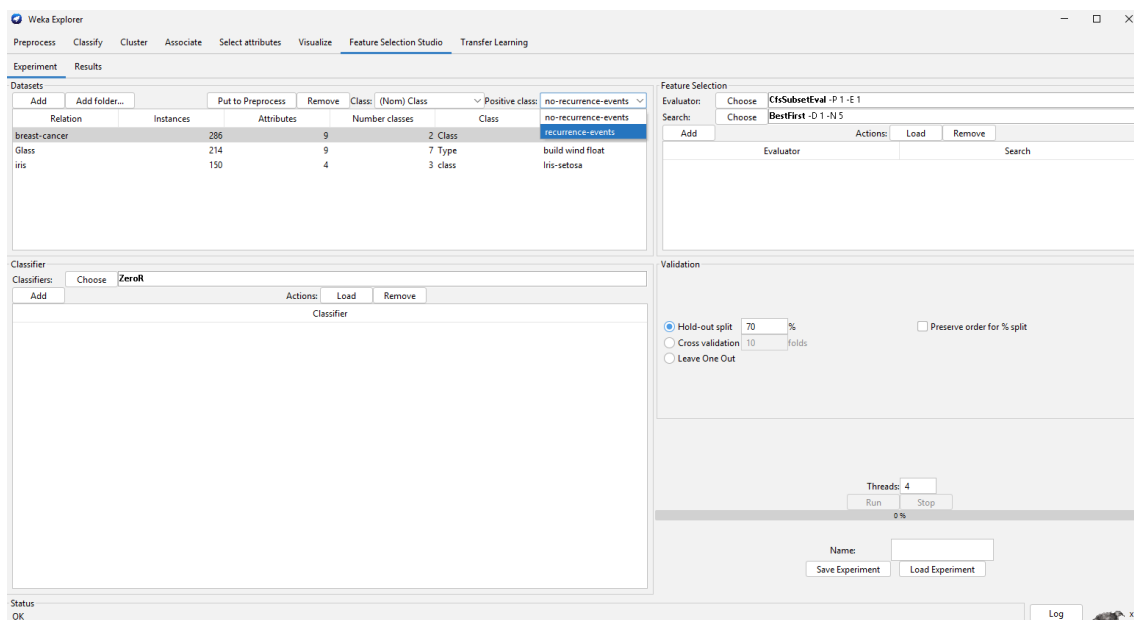
Name:

Save Experiment Load Experiment

Voy a añadir dos más, Glass e Iris (los tres añadidos vienen con Weka):



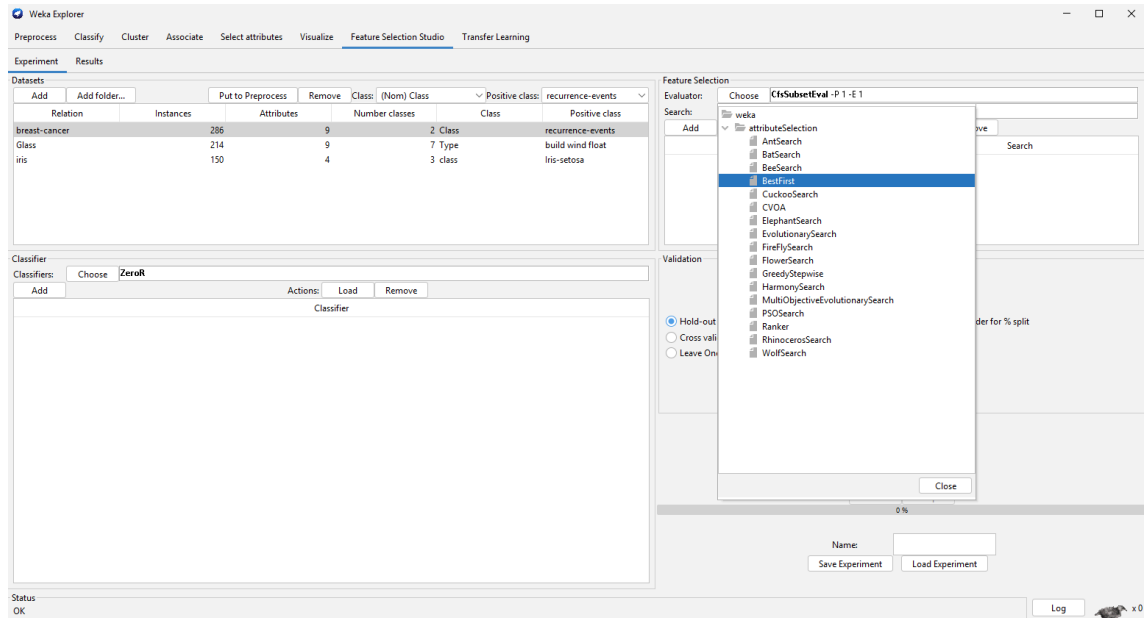
Como vemos aparecen todos los datasets añadidos junto con sus datos por cada línea, así que vamos a modificar la Positive class de breast-cancer, para ello seleccionamos la fila y en el comboBox de Positive class seleccionamos recurrence-events:



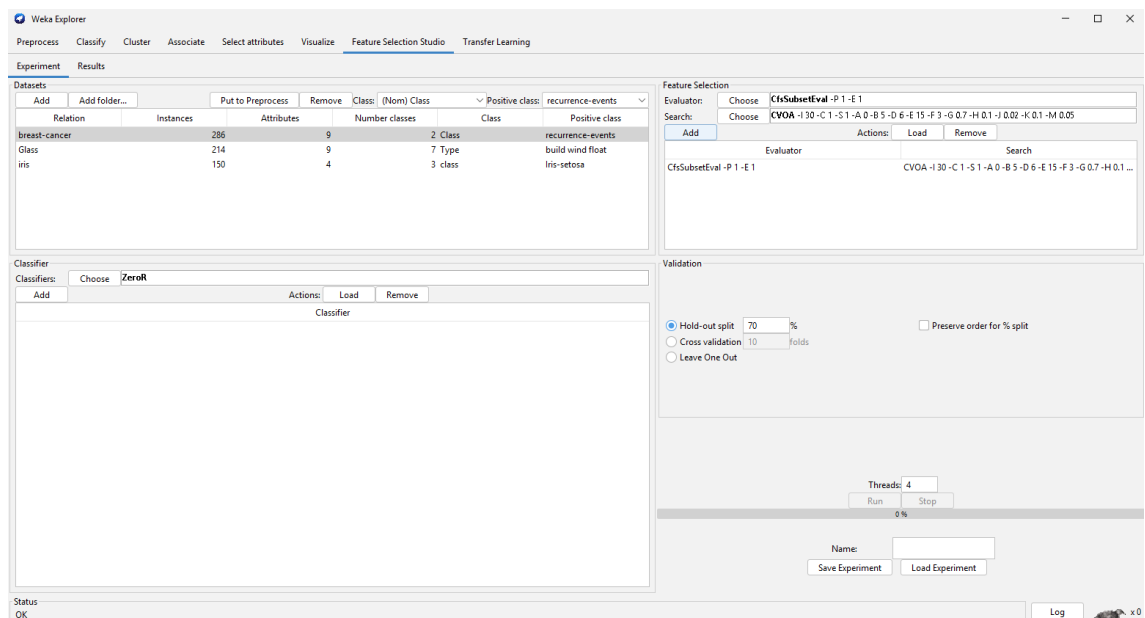
Una vez seleccionado, podemos observar que su valor se setea en la columna Positive class.

PASO 2: Algoritmos de búsqueda y evaluación

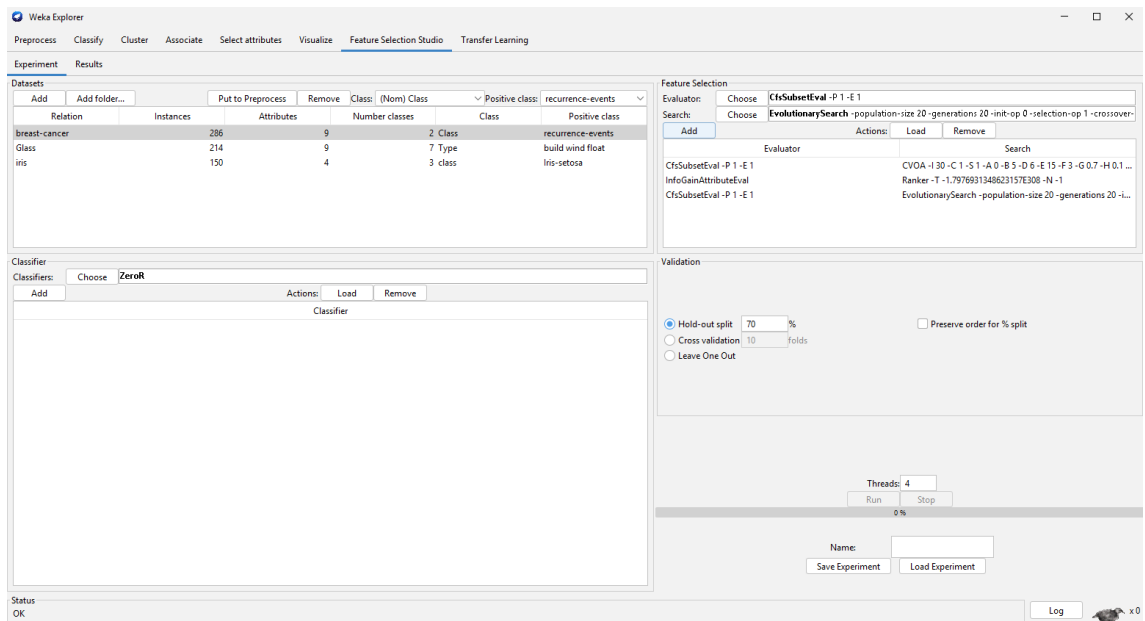
Haremos igual que antes, vamos a mostrar como se añade uno, pero vamos a añadir 3. Para ello, le damos a las botones Choose de Evaluator y Search respectivamente. Vamos a añadir CVOA (que es el que hemos integrado y adaptado a Weka). Hacemos click en Choose del apartado Search:



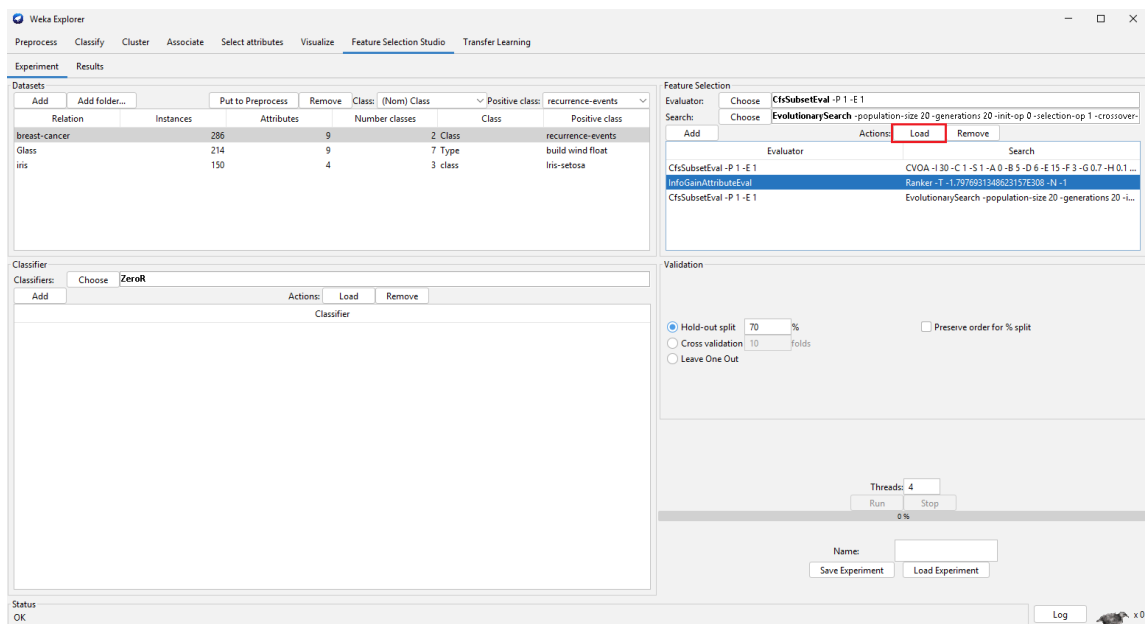
En ese menú elegimos CVOA, y como queremos ese evaluador, hacemos click en Add:



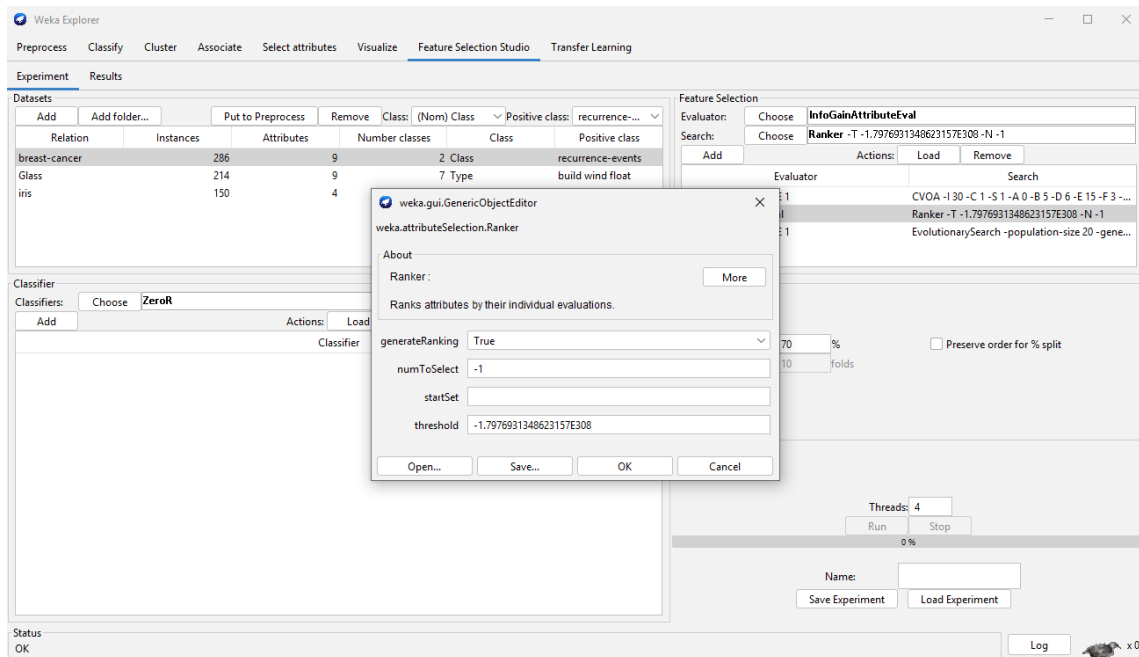
Añadiremos también Ranker y EvolutionarySearch:



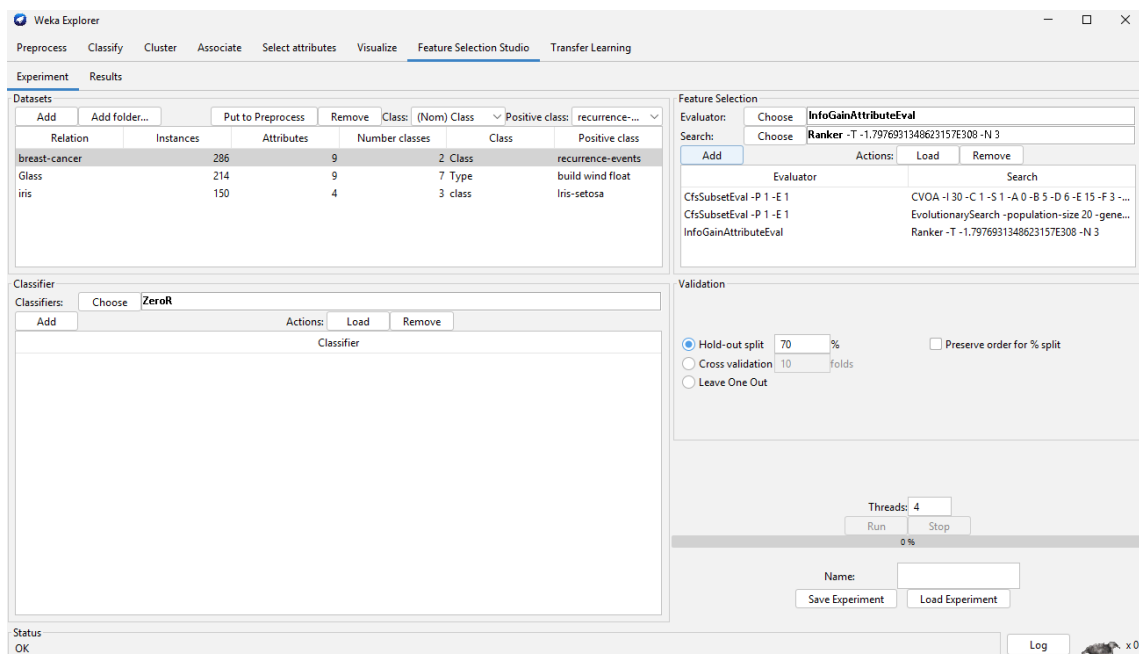
Como puede verse, en los textFields está cargado el último elemento añadido, vamos a cargar el Ranker junto con su evaluador y a modificar sus opciones (para realizar la carga de cualquier algoritmo o de un dataset se sigue el mismo proceso). Primero seleccionamos su fila y le damos a Load:



Como vemos, dichos algoritmos han sido cargados en sus respectivos textFields. Ahora para modificar sus opciones pulsamos en el textField en el cual aparece el Ranker y saldrá un menú con sus opciones:



En la opción numToSelect borramos el -1 y ponemos 3, y le damos a Ok. Para añadir este algoritmo modificado, borramos el anterior pulsando en Remove, ya que la fila ya está seleccionada, y luego volvemos a hacer click en Add:



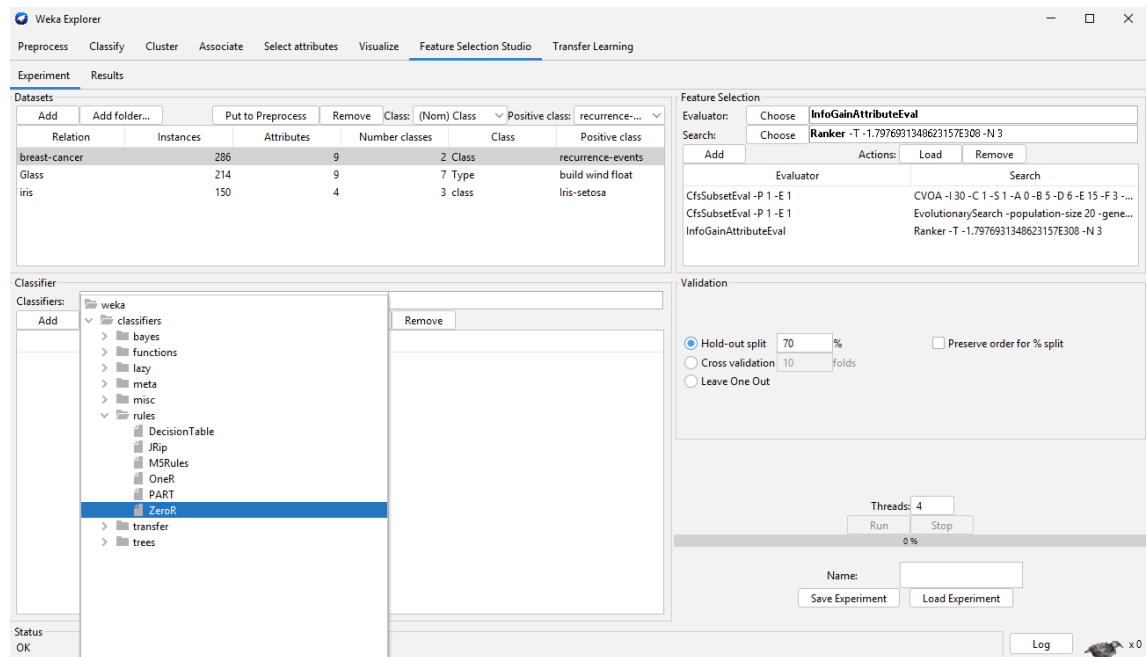
Cuando un algoritmo está cargado en su textField antes de añadirlo, podemos modificar sus opciones siguiendo el proceso descrito anteriormente sin la necesidad de eliminar ni seleccionarlo (ya que aún no estaría añadido) y añadirlo directamente. Hemos seguido este proceso para

mostrar cómo se cargaría, modificaría y eliminaría un algoritmo, ya que para el resto (algoritmo de evaluación y clasificación/regresión sería igual).

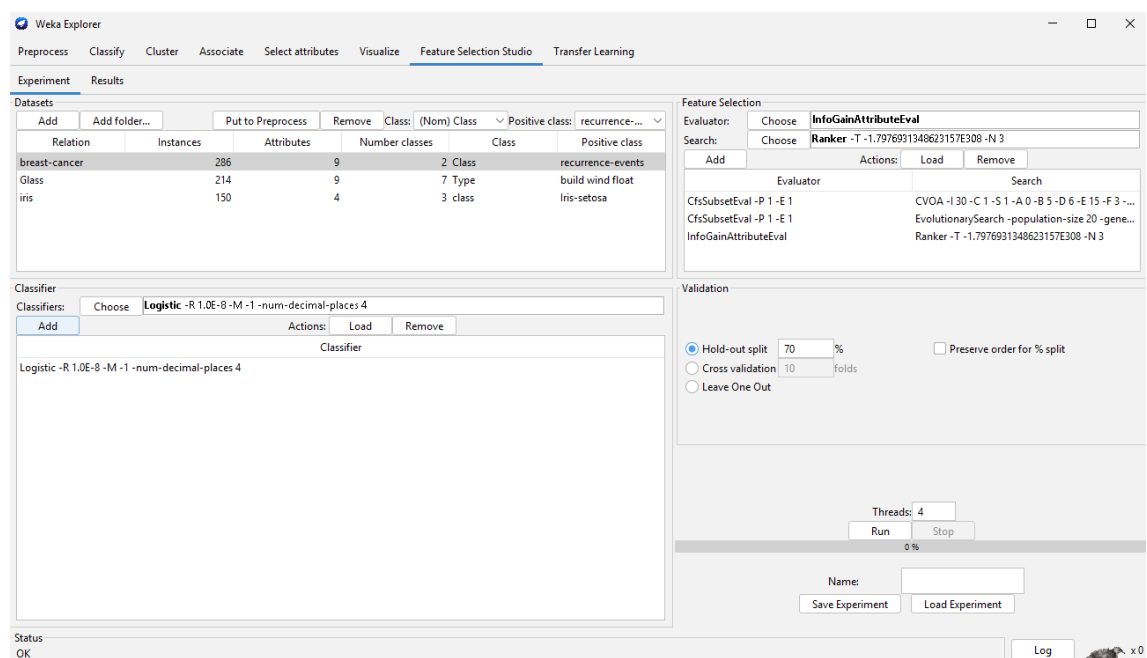
PASO 3: Algoritmo de clasificación/regresión

Vamos a mostrar el proceso para añadir uno, aunque añadiremos dos a esta experimentación.

Hacemos click sobre el botón Choose del apartado Classifier, aparecerá un menú donde nos iremos a functions y pulsamos sobre Logistic:



Ahora para añadirlo, click en Add:



Añadiré también el clasificador J48, que se encuentra en el apartado Trees.

PASO 4: Método de validación

Dejamos el método de validación y su configuración por defecto (hold-out Split y 70%). Si queremos cambiar el método de validación, simplemente lo elegimos haciendo click en su radio button. Y si queremos cambiar sus opciones, modificamos su valor en su respectivo textField.

PASO 5: Ejecución

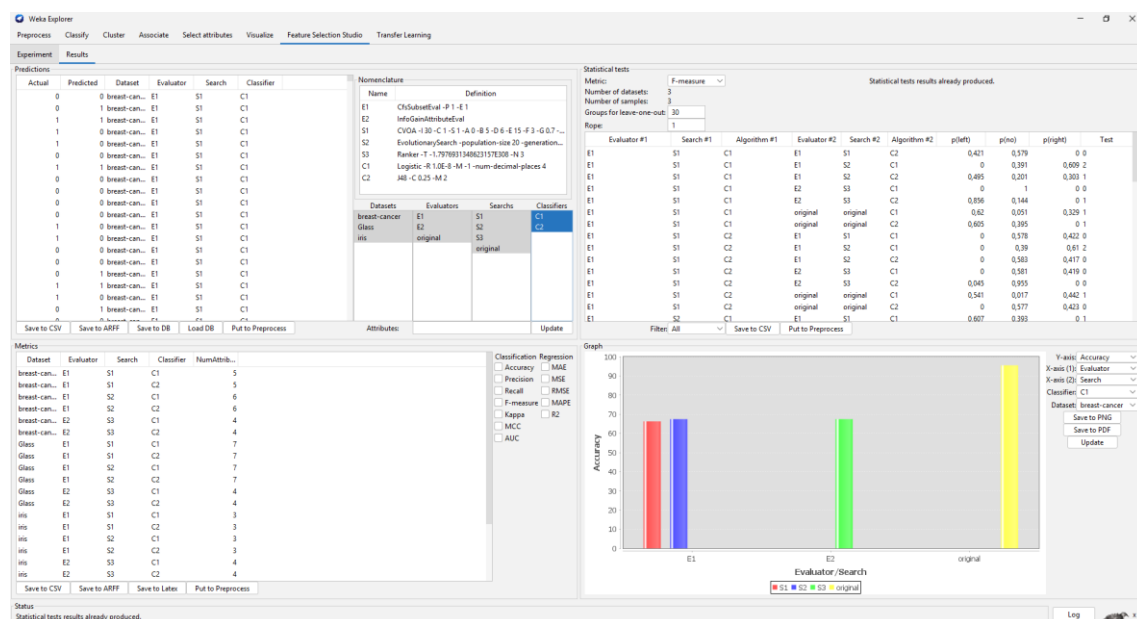
En esta ocasión, dejaremos también el número de hilos por defecto (4) que queremos utilizar para que la experimentación se lleve a cabo de forma paralela. Si queremos modificar este número, simplemente modificamos su valor en el textField correspondiente.

Ahora para ejecutar la experimentación, hacemos click en Run. En la barra de progreso se mostrará el porcentaje actual de ejecución, cuando llegue al 100% todos sus resultados se mostrarán en la pestaña de Results.

Justamente al llegar al 100%, se muestra el aviso de que, en caso de haber elegido el método de validación CV o LOO el número de atributos que aparecerá es el aplicado directamente al dataset y no el de una bolsa.

3.2 Resultados

Una vez ejecutada la experimentación, en la pestaña Results tendremos algo como esto:



PASO 6: Predicciones

Vamos a poner el caso de que queremos que en dicha tabla se muestren sólo los algoritmos bioinspirados (es decir, CVOA y EvolutionarySearch). Para ello, en las listas de Evaluators seleccionamos los dos evaluadores (E1, E2, podríamos también dejarlo tal cual está ya que van junto a sus algoritmos de búsqueda, y el filtrado de haría igualmente, pero es más correcto hacerlo de esta forma) y en la de Searches CVOA y EvolutionarySearch (S1, S2).

Ahora para que dichos cambios se apliquen a la tabla, hacemos click en el botón Update:

The screenshot shows the 'Results' tab in the Weka interface. The 'Predictions' table is filtered to show only results for E1, E2, S1, and S2. The table has columns: Actual, Predicted, Dataset, Evaluator, Search, and Classifier. The 'Nomenclature' panel on the right lists the definitions for E1, E2, S1, S2, S3, C1, and C2. Below the nomenclature panel, there are four columns: Datasets, Evaluators, Searches, and Classifiers, with a table showing the mapping of these names to specific datasets and search methods. At the bottom, there are buttons for 'Save to CSV', 'Save to ARFF', 'Save to DB', 'Load DB', 'Put to Preprocess', and an 'Update' button.

Actual	Predicted	Dataset	Evaluator	Search	Classifier
0.0	0.0	breast-can...	E1	S1	C1
0.0	1.0	breast-can...	E1	S1	C1
1.0	1.0	breast-can...	E1	S1	C1
1.0	0.0	breast-can...	E1	S1	C1
1.0	0.0	breast-can...	E1	S1	C1
0.0	0.0	breast-can...	E1	S1	C1
1.0	1.0	breast-can...	E1	S1	C1
0.0	0.0	breast-can...	E1	S1	C1
0.0	0.0	breast-can...	E1	S1	C1
0.0	0.0	breast-can...	E1	S1	C1
0.0	0.0	breast-can...	E1	S1	C1
1.0	0.0	breast-can...	E1	S1	C1
1.0	0.0	breast-can...	E1	S1	C1
0.0	0.0	breast-can...	E1	S1	C1
0.0	0.0	breast-can...	E1	S1	C1
0.0	1.0	breast-can...	E1	S1	C1
1.0	1.0	breast-can...	E1	S1	C1
1.0	0.0	breast-can...	E1	S1	C1
0.0	1.0	breast-can...	E1	S1	C1
0.0	0.0	breast-can...	E1	S1	C1

Nomenclature

Name	Definition
E1	CfsSubsetEval -P 1 -E 1
E2	InfoGainAttributeEval
S1	CVOA -I 30 -C 1 -S 1 -A 0 -B 5 -D 6 -E 15 -F 3 -G 0.7 -...
S2	EvolutionarySearch -population-size 20 -generation...
S3	Ranker -T -1.7976931348623157E308 -N 3
C1	Logistic -R 1.0E-8 -M -1 -num-decimal-places 4
C2	J48 -C 0.25 -M 2

Datasets	Evaluators	Searchs	Classifiers
breast-cancer	E1	S1	C1
Glass	E2	S2	C2
iris	original	S3	
		original	

Save to CSV Save to ARFF Save to DB Load DB Put to Preprocess Attributes: Update

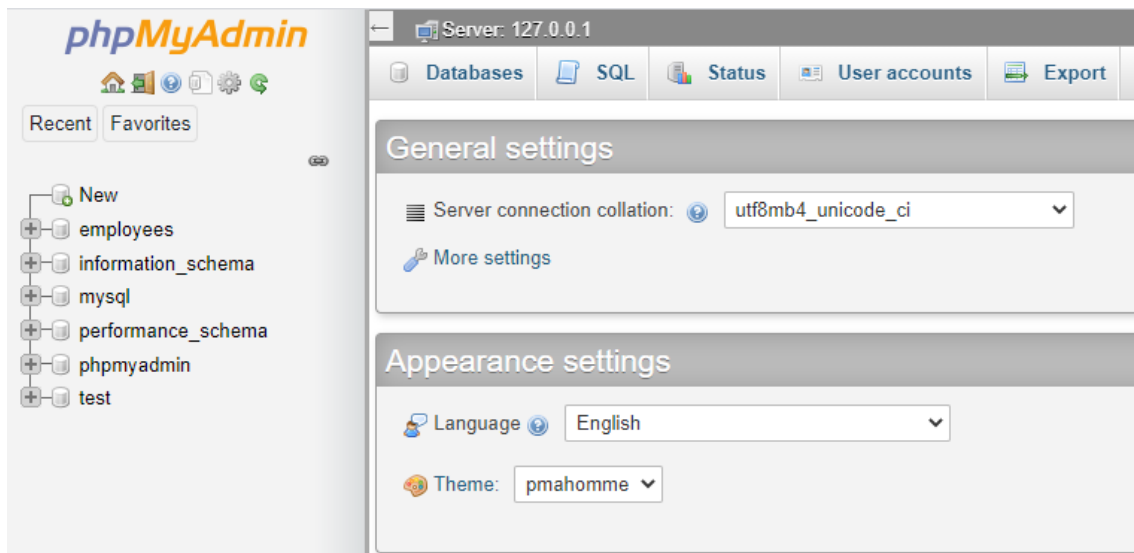
La tabla de predicciones ha sido filtrada. Como para realizar la carga de dichos datos en la pestaña Preprocess de Weka sólo hay que pulsar en el botón Load dataset, vamos a mostrar cómo se guardan los datos en CSV y en una base de datos (para esta segunda acción hay que tener configurado el fichero DatabaseUtils.props tal y como se explicita en el punto **4 CONFIGURACIÓN DATABASEUTILS** del presente documento).

Para guardar los datos en CSV, hace click en el botón Save to CSV de dicha sección apareciendo el menú donde elegiremos el nombre del fichero y su ubicación:

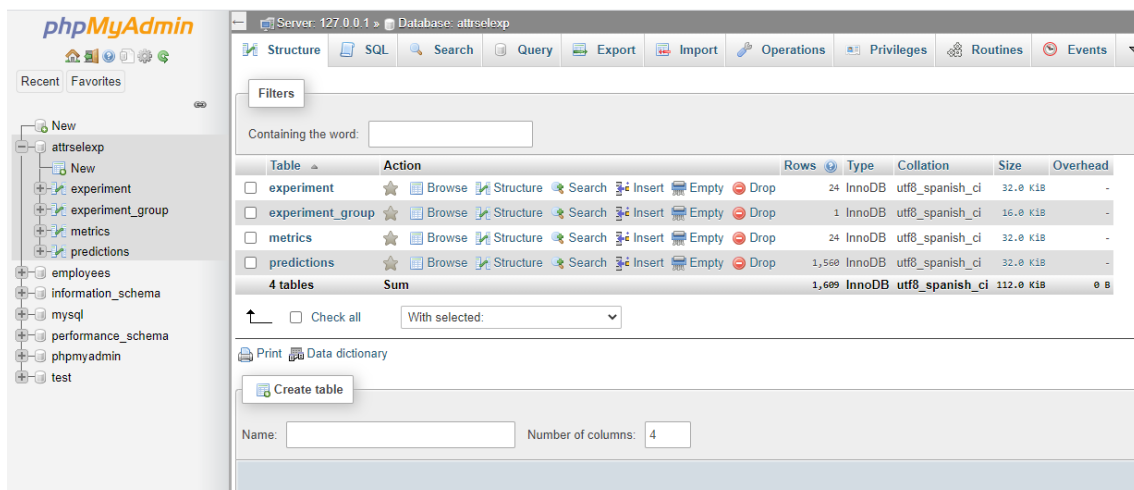
Y hacemos click en Guardar. Para comprobar que se ha guardado correctamente, nos vamos a dicha carpeta y abrimos el CSV:

Ahora guardaremos los datos en una base de datos, para ello y una vez configurado el fichero DatabaseUtils.props, hacemos click en Save to DB (si la base de datos no existe o no existe alguna tabla, se crearán automáticamente a partir de scripts sqls). Esta acción guarda tanto las predicciones como las métricas (en esta última, guarda todas las métricas se han elegido o no).

Primero y para mostrar que efectivamente crea la base de datos si no existe, con el gestor de base de datos xampp y a través de PhpMyAdmin vamos a ver las bases de datos que tenemos actualmente:



Ahora hacemos click en el botón Save to DB, y vemos que se ha creado la base de datos con los datos guardados:



Más adelante cargaremos los datos guardados actualmente.

PASO 7: Métricas

Para guardar los datos en CSV o ARFF, debemos seguir los mismos pasos llevados a cabo anteriormente para guardar las predicciones en CSV, y para cargarlos en la pestaña Preprocess, pues sería también pulsado el botón Load dataset, pero esta vez de esta sección. Por ello, vamos a mostrar cómo añadir métricas a la tabla.

Este proceso es tan fácil como hacer click en los checkboxes de las métricas que queramos añadir. Al tratarse de datasets de clasificación, añadiré las métricas Accuracy y F-measure. Añadiré también una métrica de regresión para ver que si los datasets son del tipo contrario dicha columna aparecen todos sus valores a cero:

Metrics									
Dataset	Evaluator	Search	Classifier	NumAttrib...	Accuracy	F-measure	MSE		
breast-can...	E1	S1	C1	5	66.2790697...	0.43137254...	0.0	<input checked="" type="checkbox"/> Accuracy	<input type="checkbox"/> MAE
breast-can...	E1	S1	C2	5	62.7906976...	0.0	0.0	<input type="checkbox"/> Precision	<input checked="" type="checkbox"/> MSE
breast-can...	E1	S2	C1	6	67.4418604...	0.48148148...	0.0	<input type="checkbox"/> Recall	<input type="checkbox"/> RMSE
breast-can...	E1	S2	C2	6	62.7906976...	0.0	0.0	<input checked="" type="checkbox"/> F-measure	<input type="checkbox"/> MAPE
breast-can...	E2	S3	C1	4	67.4418604...	0.44000000...	0.0	<input type="checkbox"/> Kappa	<input type="checkbox"/> R2
breast-can...	E2	S3	C2	4	62.7906976...	0.0	0.0	<input type="checkbox"/> MCC	
Glass	E1	S1	C1	7	59.375	0.60465116...	0.0	<input type="checkbox"/> AUC	
Glass	E1	S1	C2	7	60.9375	0.6	0.0		
Glass	E1	S2	C1	7	57.8125	0.61904761...	0.0		
Glass	E1	S2	C2	7	59.375	0.65000000...	0.0		
Glass	E2	S3	C1	4	64.0625	0.60465116...	0.0		
Glass	E2	S3	C2	4	59.375	0.58333333...	0.0		
iris	E1	S1	C1	3	95.555555...	1.0	0.0		
iris	E1	S1	C2	3	95.555555...	1.0	0.0		
iris	E1	S2	C1	3	95.555555...	1.0	0.0		
iris	E1	S2	C2	3	95.555555...	1.0	0.0		
iris	E2	S3	C1	4	95.555555...	1.0	0.0		
iris	E2	S3	C2	4	95.555555...	1.0	0.0		

Save to CSV Save to ARFF Save to Latex Put to Preprocess

PASO 8: Tests Estadísticos

Por defecto, las pruebas estadísticas se calculan con la métrica F-Measure para clasificación (y MAE para regresión), en cuanto esta métrica cambia en el comboBox, se recalculan las pruebas estadísticas. En nuestro ejemplo actual, la parte de las pruebas estadísticas se vería así:

Statistical tests									
Metric:		F-measure		Statistical tests results already produced.					
Number of datasets:		3							
Number of samples:		3							
Groups for leave-one-out:		30							
Rope:		1							
Evaluator #1	Search #1	Algorithm #1	Evaluator #2	Search #2	Algorithm #2	p(left)	p(no)	p(right)	Test
E1	S1	C1	E1	S1	C2	0	0,954	0,046	0
E1	S1	C1	E1	S2	C1	0	0,955	0,045	0
E1	S1	C1	E1	S2	C2	0	0,392	0,608	2
E1	S1	C1	E2	S3	C1	0	0,955	0,045	0
E1	S1	C1	E2	S3	C2	0,259	0,686	0,056	0
E1	S1	C1	original	original	C1	0,855	0,145	0	1
E1	S1	C1	original	original	C2	0,327	0,673	0	0
E1	S1	C2	E1	S1	C1	0,046	0,954	0	0
E1	S1	C2	E1	S2	C1	0	0,954	0,046	0
E1	S1	C2	E1	S2	C2	0	0,582	0,418	0
E1	S1	C2	E2	S3	C1	0	1	0	0
E1	S1	C2	E2	S3	C2	0,046	0,954	0	0
E1	S1	C2	original	original	C1	0,963	0,037	0	1
E1	S1	C2	original	original	C2	0,421	0,579	0	0
E1	S2	C1	E1	S1	C1	0,046	0,954	0	0

Filter: All Save to CSV Put to Preprocess

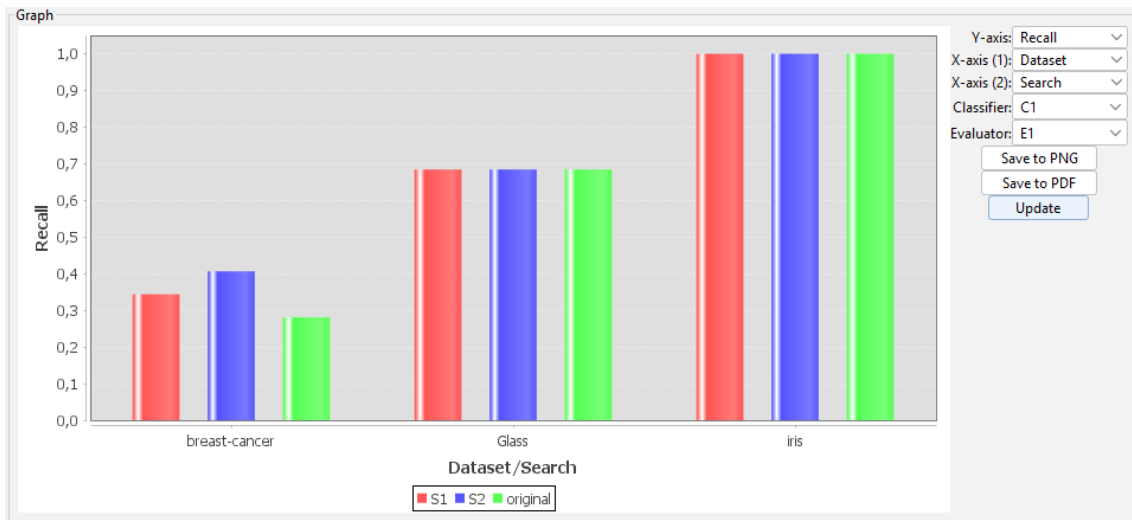
Para que se calculen las pruebas estadísticas, el usuario debe tener python, numpy (pip install numpy) y el paquete utilizado para realizar estas pruebas baycomp (pip install baycomp, <https://github.com/janezd/baycomp>) instalado en la computadora que ejecuta Weka.

PASO 9: Gráficas

Por defecto, se genera una gráfica con la métrica Accuracy o MAE (dependiendo si la experimentación llevada a cabo es de datasets de regresión o de clasificación), teniendo en el primer valor del eje X el evaluador y en el segundo el algoritmo de búsqueda. Vamos a generar

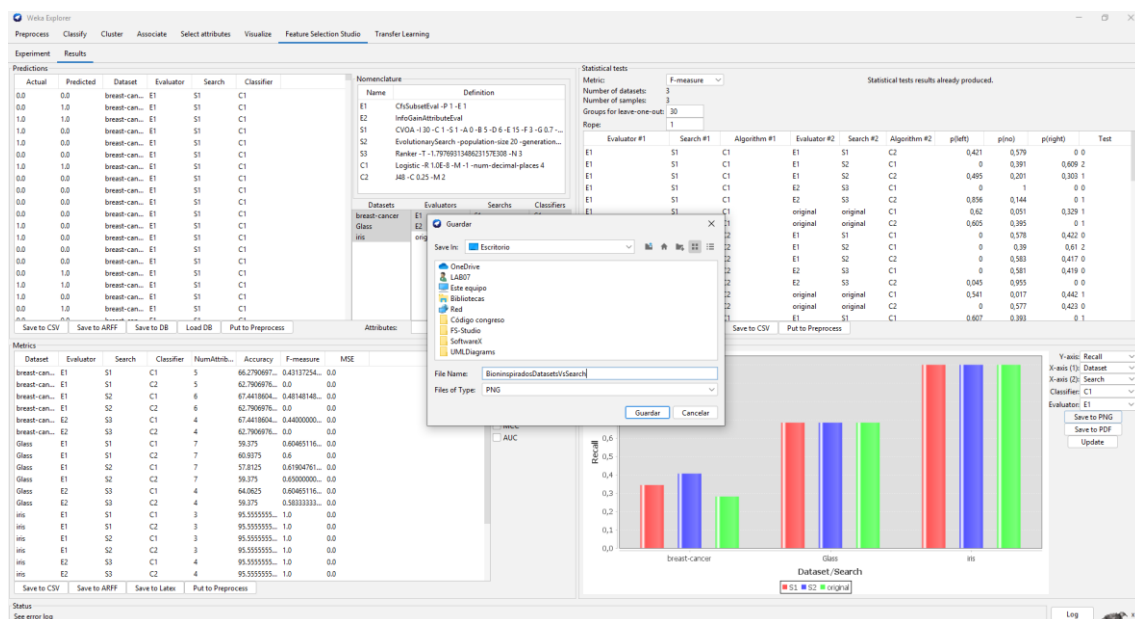
una nueva gráfica con otros valores y a exportarla en PNG por ejemplo (el proceso para PDF sería exactamente igual).

Para mostrar que las métricas insertadas en la tabla de métricas son independientes de que aparezcan o no en la gráfica (es decir, puedes tener unas métricas añadidas a la tabla, y aquí seleccionar cualquier otra), vamos a elegir la métrica Recall, el valor del eje X 1 pondremos Dataset y en el eje X 2 Search, en el del Classifier ponemos C1 y en el del Evaluador ponemos E1. Para ello simplemente hacemos click en sus respectivos checkboxes donde se desplegarán los valores posibles. Una vez hecho esto, para generar la nueva gráfica hacemos click en el botón Update:

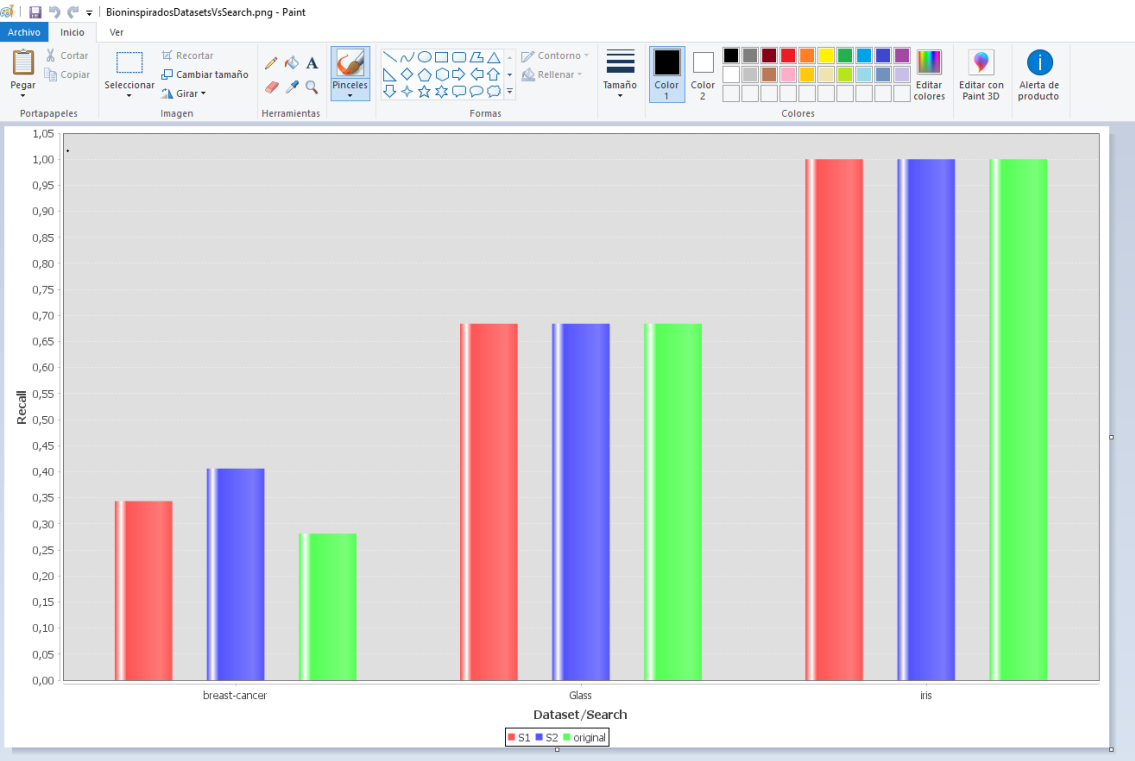


Se puede identificar fácilmente en este caso a los algoritmos de búsqueda utilizados por sus colores.

Para exportar la gráfica en formato PNG, hacemos click en el botón Save to PNG. Aparecerá un diálogo donde elegiremos donde guardarlo, y su nombre:



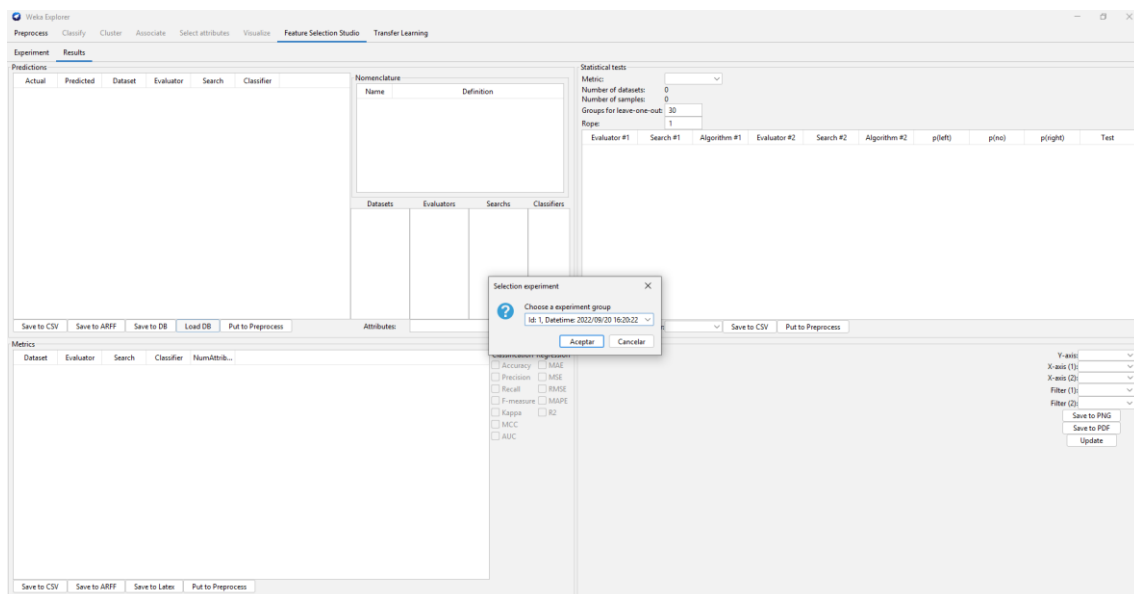
Y le damos a guardar. Para comprobar que se ha guardado bien, nos vamos a dicha carpeta y abrimos la imagen:



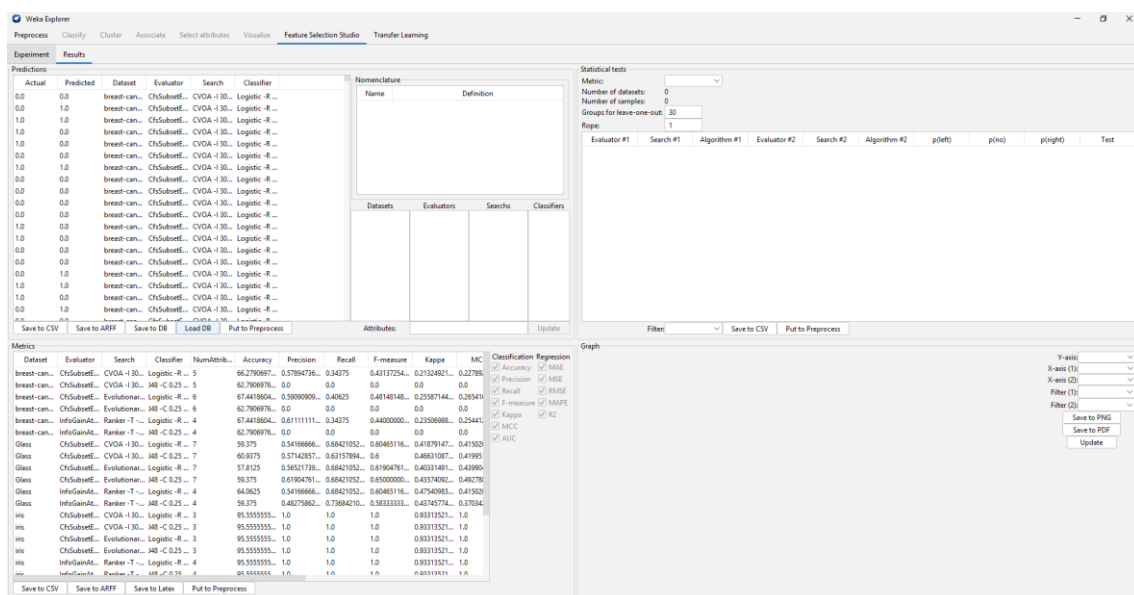
PASO EXTRA: Cargar datos desde la base de datos

Los datos pueden cargarse desde la base de datos sin necesidad de haber ejecutado un experimento. Es por ello que utilizaremos dicho camino para ver mejor las diferencias que se provocan en la interfaz (también pueden cargarse habiendo ejecutado una experimentación y obtenido los resultados sin ningún problema).

Para ello, nos vamos a la pestaña FeatureSelectionStudio en Weka, y aquí a la pestaña Results. Ahora hacemos click en Load (necesitamos tener configurado correctamente el fichero databaseUtils.prop, mirar apartado **4 CONFIGURACIÓN DATABASEUTILS** del presente documento) y nos aparecerá el siguiente diálogo:



Al seleccionar el checkBox que aparece, se desplegarán todas las experimentaciones guardadas representadas por su Id y la fecha y hora en la cual se guardó. Seleccionamos la que nos aparece ya que es la única que tenemos y le damos a Aceptar:



Y ya tendríamos cargados los datos desde la base de datos. Puede observarse que hay opciones que no están disponibles, es a causa de que lo que se guardan son datos y no objeto. Las opciones disponibles son: guardar en CSV, en ARFF, de nuevo a una base de datos y cargar datos otra vez desde la base de datos.

4 CONFIGURACIÓN DATABASEUTILS

Para poder utilizar las funcionalidades asociadas a la base de datos, es necesario introducir correctamente los datos que se solicitan a continuación y de la forma que se expone en el fichero **databaseUtils.props**, además de tener levantado el servidor perteneciente a la url que introduciremos más adelante. Los pasos a seguir para llevar a cabo la correcta configuración de los parámetros necesarios para acceder a la base datos son:

PASO 1

Una vez tenemos instalado en Weka el paquete FeatureSelectionStudio, al igual que el resto de los paquetes que instalemos, se descomprimirá en la dirección C:\Users\Usuario\wekafiles\packages de nuestro ordenador. Por lo que vamos a dicha dirección y entramos en la carpeta de FS-Studio:

Nombre	Fecha de modifica...	Tipo	Tamaño
EvolutionarySearch	22/04/2022 13:24	Carpeta de archivos	
FS-Studio	19/09/2022 15:34	Carpeta de archivos	
metaphorSearchMethods	15/09/2022 15:02	Carpeta de archivos	
MultiObjectiveEvolutionarySearch	22/04/2022 14:06	Carpeta de archivos	
PSOSearch	15/09/2022 15:03	Carpeta de archivos	
WekaTransfer	22/04/2022 13:14	Carpeta de archivos	
installedPackageCache.ser	19/09/2022 15:34	Archivo SER	8 KB

PASO 2

Una vez allí, abrimos el fichero DatabaseUtils.props con un editor de texto cualquiera, en nuestro caso, lo abriremos con el bloc de notas:

Nombre	Fecha de modifica...	Tipo	Tamaño
DB	19/09/2022 15:34	Carpeta de archivos	
doc	19/09/2022 15:34	Carpeta de archivos	
lib	19/09/2022 15:34	Carpeta de archivos	
bayesianComparator.py	19/09/2022 15:34	JetBrains PyCharm	1 KB
build_package.xml	19/09/2022 15:34	Documento XML	9 KB
DatabaseUtils.props	19/09/2022 15:34	Archivo PROPS	3 KB
Description.props	19/09/2022 15:34	Archivo PROPS	2 KB
Explorer.props	19/09/2022 15:34	Archivo PROPS	1 KB
fs-studio.jar	19/09/2022 15:34	Executable Jar File	276 KB

Al abrirlo nos debemos de centrar en las líneas comentadas de “#The url to the experiment database”, “#Your user” y “#Your password”:

DatabaseUtils.props: Bloc de notas

Archivo Edición Formato Ver Ayuda

```
# General information on database access can be found here:
# https://waikato.github.io/weka-wiki/databases/
#
# Version: $Revision: 15257 $

# The comma-separated list of jdbc drivers to use
#jdbcDriver=RmiJdbc.RJDriver,jdbc.idbDriver
#jdbcDriver=jdbc.idbDriver
#jdbcDriver=RmiJdbc.RJDriver,jdbc.idbDriver,org.gjt.mm.mysql.Driver,com.mckoi.JDBCdriver,org.hsqldb.jdbcDriver
jdbcDriver=com.mysql.jdbc.Driver

# The url to the experiment database
#jdbcURL=jdbc:rmi://expserver/jdbc:idb=experiments.prp
#jdbcURL=jdbc:idb=experiments.prp
#jdbcURL=jdbc:mysql://localhost:3306/

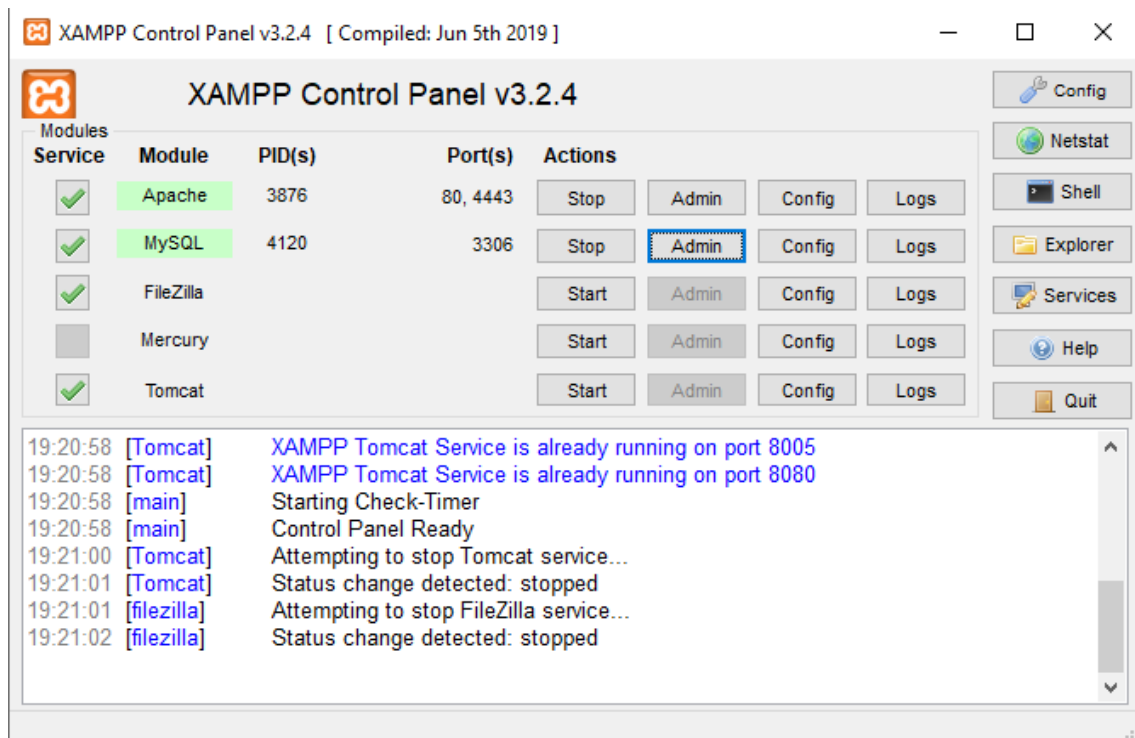
#Your user
#user=root

#Your password
#password=

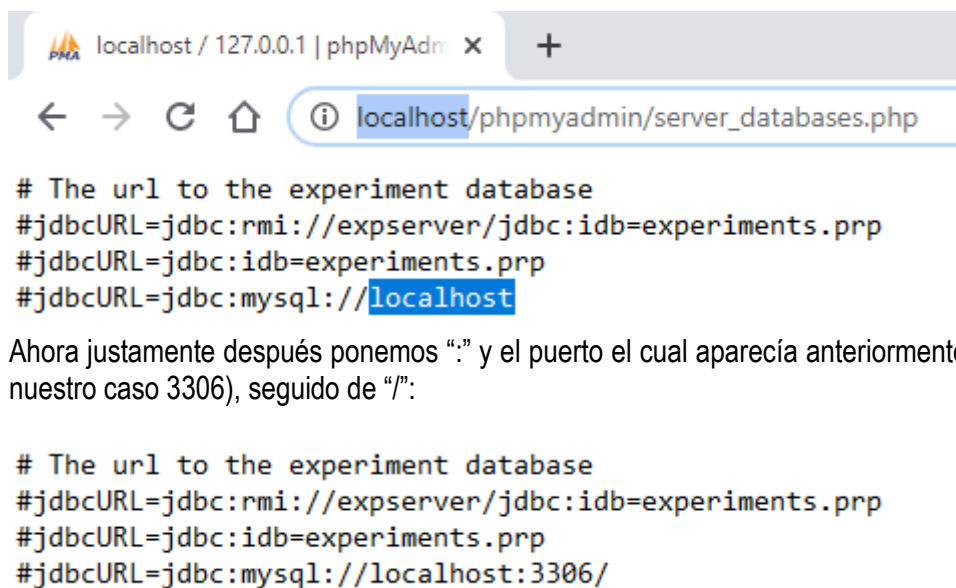
# the method that is used to retrieve values from the db
# (java datatype + ResultSet.<method>)
# string, getString() = 0; --> nominal
# boolean, getBoolean() = 1; --> nominal
```

Descomentamos la última línea de cada uno de ellos borrando el símbolo “#”. En caso de poseer dichos datos (url = localhost:3306/, user = root y password vacía), guardamos y ya estaría listo.

En caso de poseer otros datos, borramos los que vienen por defecto y ponemos lo nuestros. Para saber la url de nuestro servidor, en caso de tener un gestor de base de datos como Xampp y **asegurarnos de que está activo**, en la sección de MySql le damos a Admin:



Ahora se abrirá en nuestro navegador que tenemos puesto por defecto el gestor PhpMyAdmin, cogemos su url (sólo lo que aparece antes del primer "/") y la ponemos en su lugar correspondiente en el fichero:



Ahora justamente después ponemos ":" y el puerto el cual aparecía anteriormente en Xampp (en nuestro caso 3306), seguido de "/"

```
# The url to the experiment database
#jdbcURL=jdbc:rmi://expserver/jdbc:idb=experiments.prp
#jdbcURL=jdbc:idb=experiments.prp
#jdbcURL=jdbc:mysql://localhost:3306/
```

Y listo, ya podemos acceder a las funciones que interactúan con la base de datos.