

Instituto Tecnológico de Ciudad Madero

Miranda Martínez Diego Ismael

19071551

Inteligencia artificial

12:00 – 1:00 pm

Tarea 3. Weka DM

30 de abril de 2023

ÍNDICE

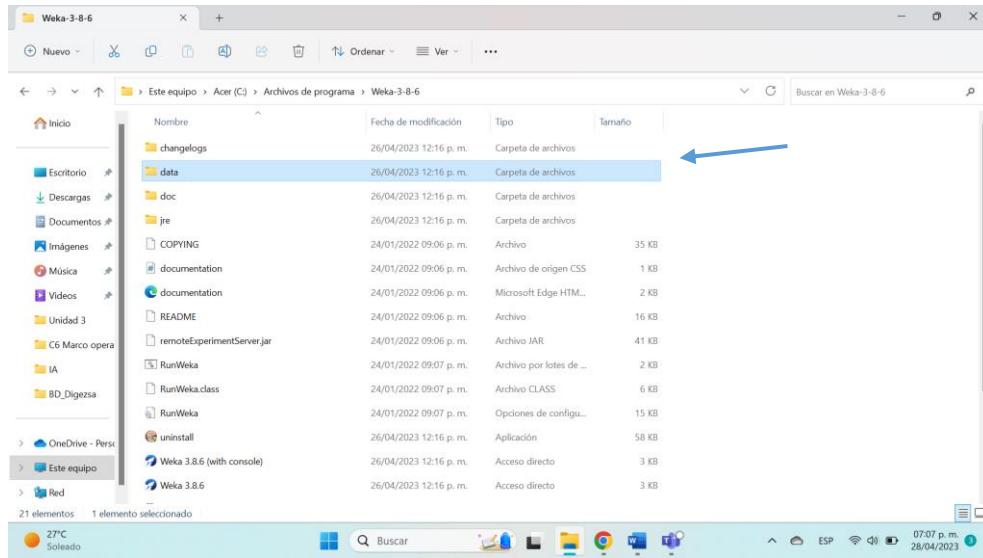
Exploring the explorer	1, 2
Exploring datasets	2, 3
Building a classifier	3, 4, 5
Using filter	5, 6, 7, 8
Visualizing your data	8, 9
Be a classifier	9, 10
Training and testing	10, 11, 12
Repeated training and testing	12, 13
Baseline accuracy	13, 14, 15
Cross-validation	15, 16
Cross-validation results	16
Simplicity first!	16, 17
Overfitting	17, 18
Using probabilities	18, 19
Decision trees	19
Pruning decision trees	20, 21
Nearest neighbor	21
Classification boundaries	22, 23
Linear regression	23
Classification by regression	24
Logistic regression	24, 25

Support vector machines 25

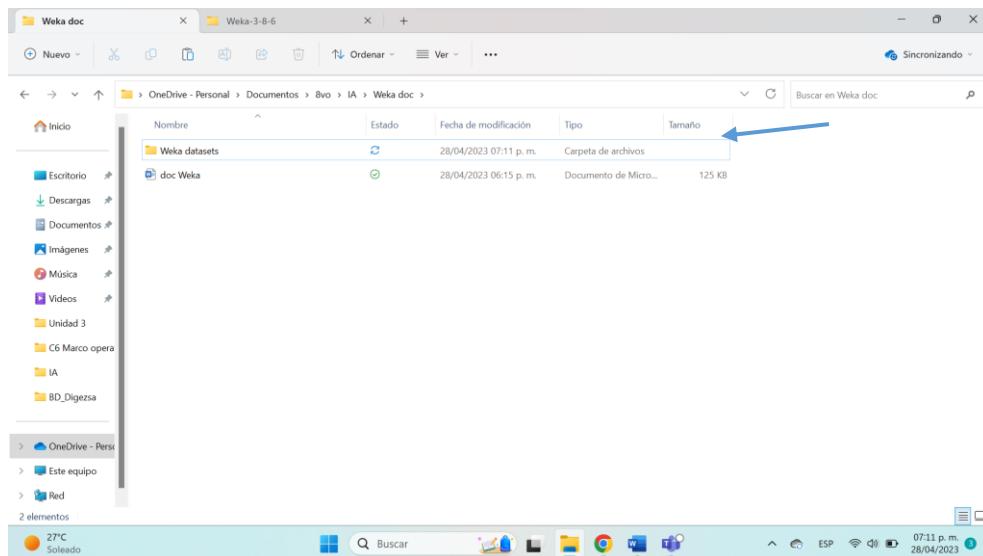
Ensemble learning 26

WEKA – 1.2 Exploring the explorer

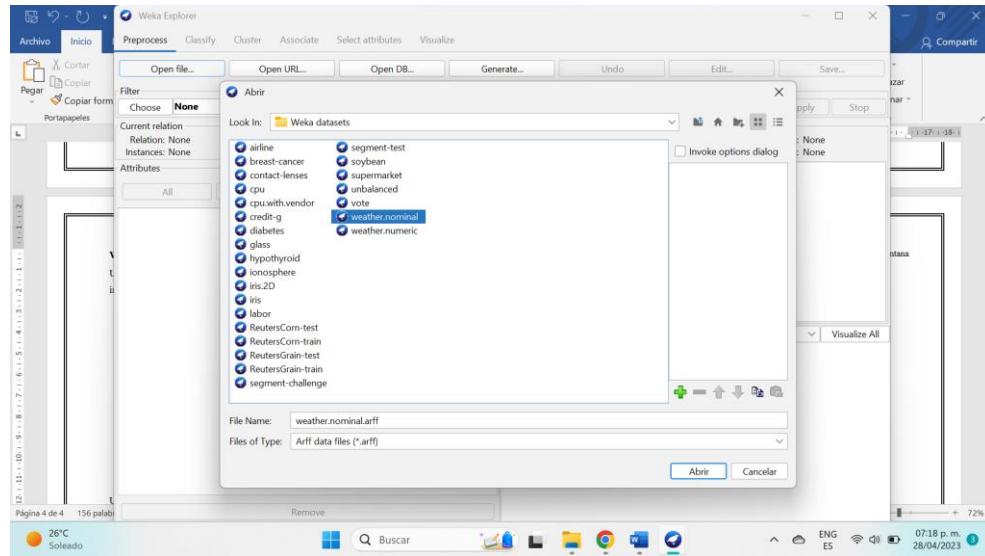
Una vez instalada la aplicación de Weka, procedemos abrir los archivos que genera la instalación, debemos ubicar la carpeta “Data”.



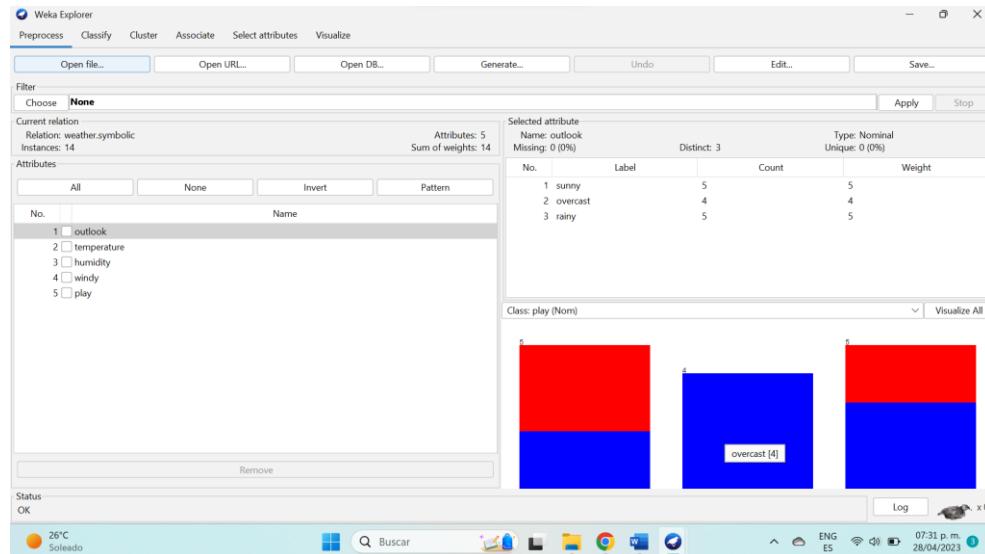
Una vez que hayamos ubicado la carpeta, la copiamos y la pegamos en “Documentos”, pudiendo crear una carpeta o pegarla directa y al finalizar cambiarle el nombre por *Weka datasets*.



Ejecutamos la aplicación de Weka y buscamos el botón “Explorer” y se nos abrirá una ventana en donde abriremos el archivo de ejercicios de nombre Weka datasets y abrimos el ejercicio weather.nominal.



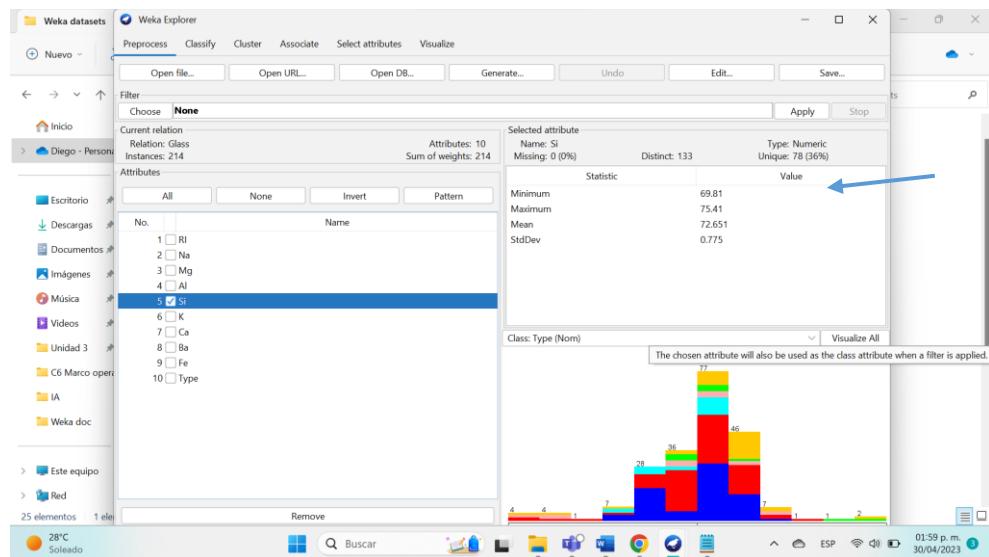
Cuando abrimos el ejercicio, nos aparecerá una gráficas y datos que podemos cambiar dependiendo que checkbox seleccionemos, también podemos editar los valores de los datos y después guardarlos para que se actualicen y podamos verlos.



WEKA – 1.3 Exploring datasets

En este capítulo se visualizarán algunos datos del ejercicio weather.numeric.

Después se procede a abrir el ejercicio *glass* el cual tiene 214 instancias de valores dependiendo la clasificación (clases de vidrio) que se está manejando.



Para confirmar que los porcentajes de cada tipo de clasificación que existe, nos dirigimos a la carpeta de datasets y damos click en el ejercicio que estamos trabajando, después abrimos el ejercicio como tipo bloc de notas y buscamos la sección donde se encuentra los valores para confirmar.

```

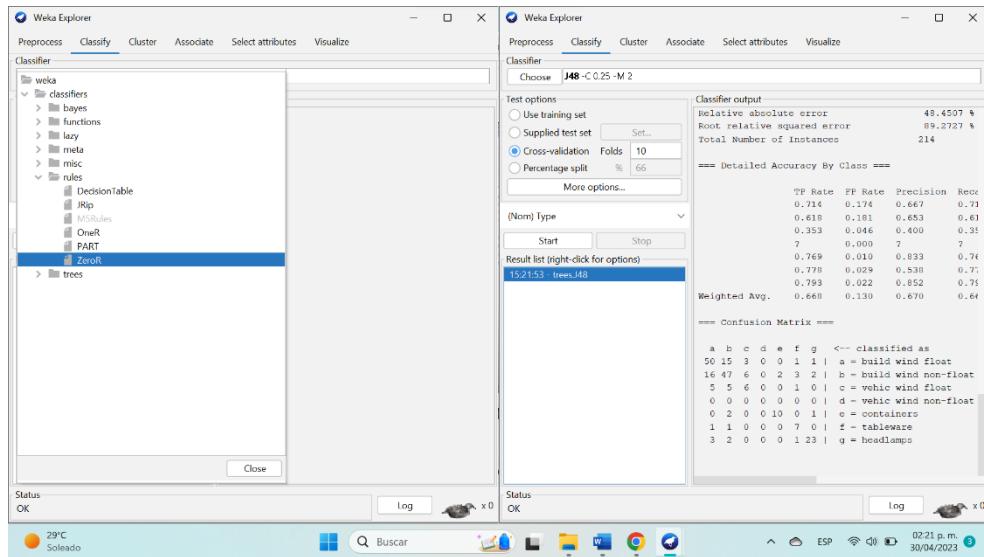
% 11. Type of glass (class attribute)
%   -- 1 building_windows_float_processed
%   -- 2 building_windows_non_float_processed
%   -- 3 vehicle_windows_float_processed
%   -- 4 vehicle_windows_non_float_processed (none in this database)
%   -- 5 containers
%   -- 6 tableware
%   -- 7 headlamps
%
% 8. Missing Attribute Values: None
%
% Summary Statistics:
% Attribute: Min Max Mean SD Correlation with class
% 2. RI: 1.5112 1.5339 1.5184 0.0030 -0.1642
% 3. Na: 10.73 17.38 13.4879 0.8166 0.5920
% 4. Mg: 0 4.49 2.6845 1.4424 -0.7447
% 5. Al: 0.29 3.5 1.4449 0.4993 0.5988
% 6. Si: 69.81 75.41 72.6509 0.7745 0.1515
% 7. K: 0 6.21 0.4971 0.6522 -0.0100
% 8. Ca: 5.43 16.19 8.9570 1.4232 0.0007
% 9. Ba: 0 3.15 0.1750 0.4972 0.5751
% 10. Fe: 0 0.57 0.0570 0.0974 -0.1879
%
% 9. Class Distribution: (out of 214 total instances)
%   -- 163 Window glass (building windows and vehicle windows)
%   -- 87 float processed
%     -- 78 building windows
%     -- 17 vehicle windows
%   -- 76 non-float processed
%     -- 76 building windows
%     -- 0 vehicle windows
%   -- 51 Non-window glass
%     -- 13 containers
%     -- 9 tableware
%     -- 0 headlamps
%
Ln 1, Col 1
28°C Soleado

```

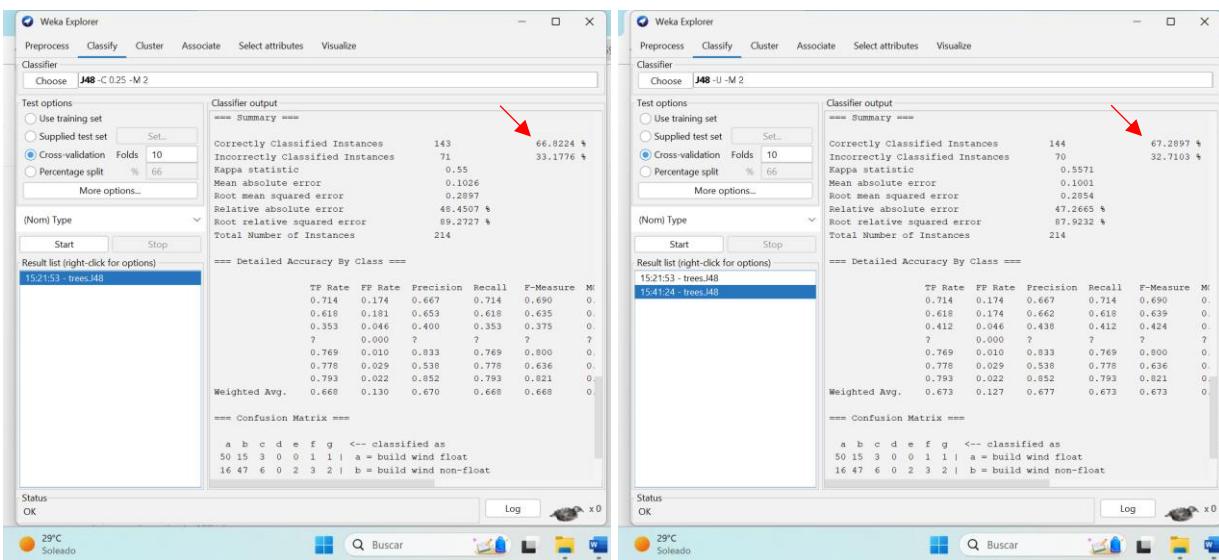
WEKA – 1.4 Building a classifier

En este capítulo se visualizará las diferentes clasificaciones con las que se pueden trabajar. La cual se trabajará con la clasificación J48 para analizar los datos del ejercicio glass.

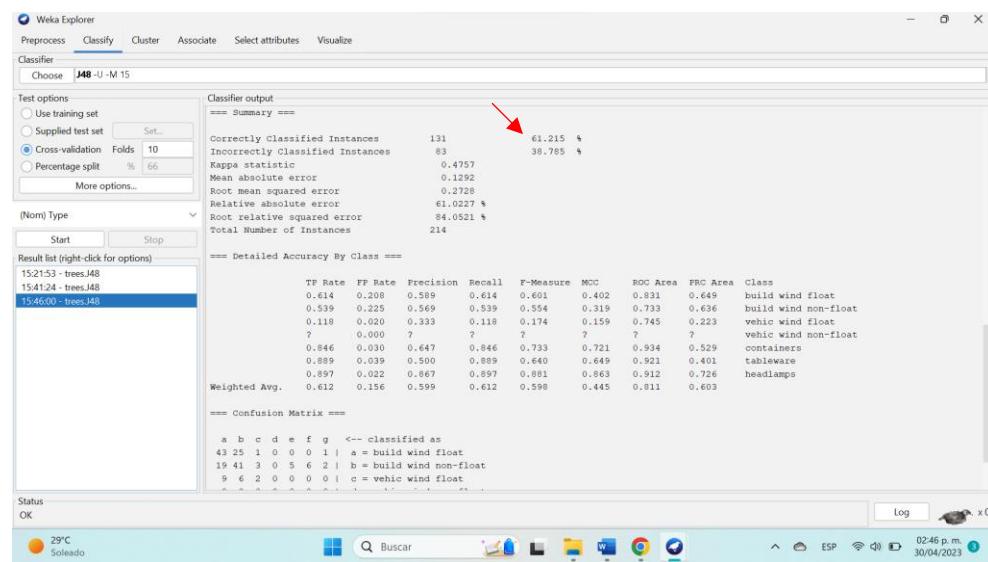
- La pantalla del lado izquierdo muestra todas las clasificaciones.
- La pantalla del lado derecho muestra el análisis del ejercicio glass con la clasificación J48.



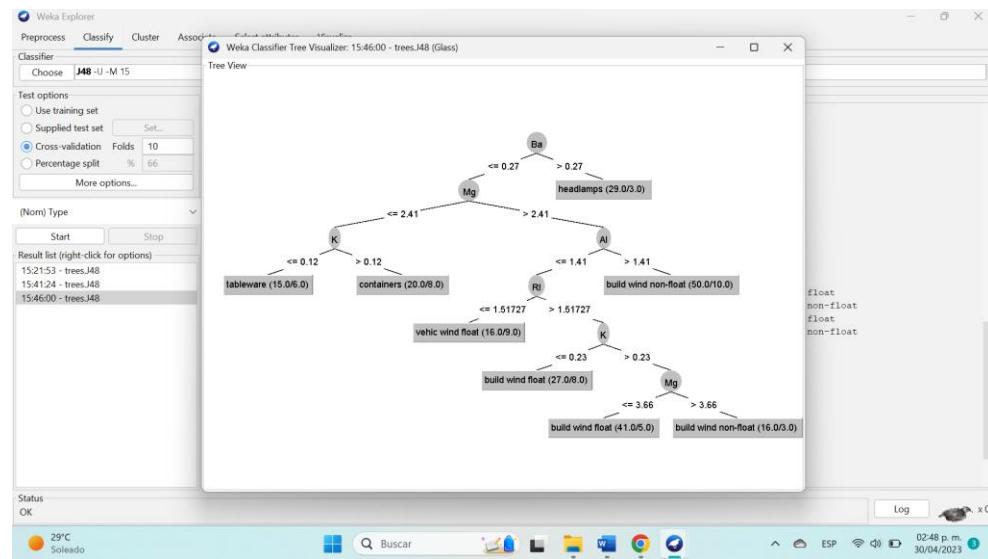
En los parámetros de la clasificación se hará un pequeño cambio para poder visualizar que diferencia de porcentajes existe entre ese cambio (unpruned → true).



Se vuelve hacer otro cambio a un parámetro y se visualiza que volvió a cambiar el porcentaje de objetos correctos.

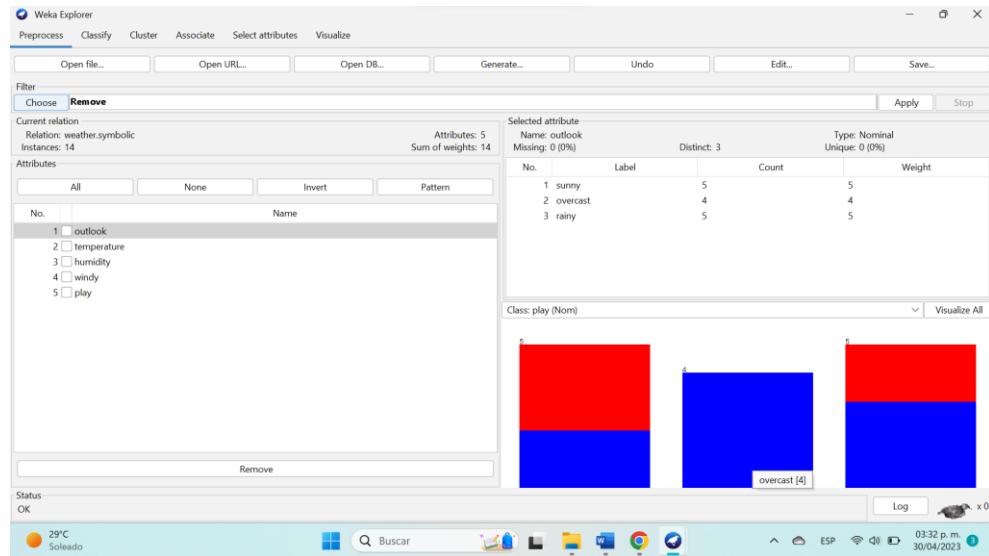


Este último análisis lo vamos a visualizar en forma de árbol, mostrando los valores correspondientes.

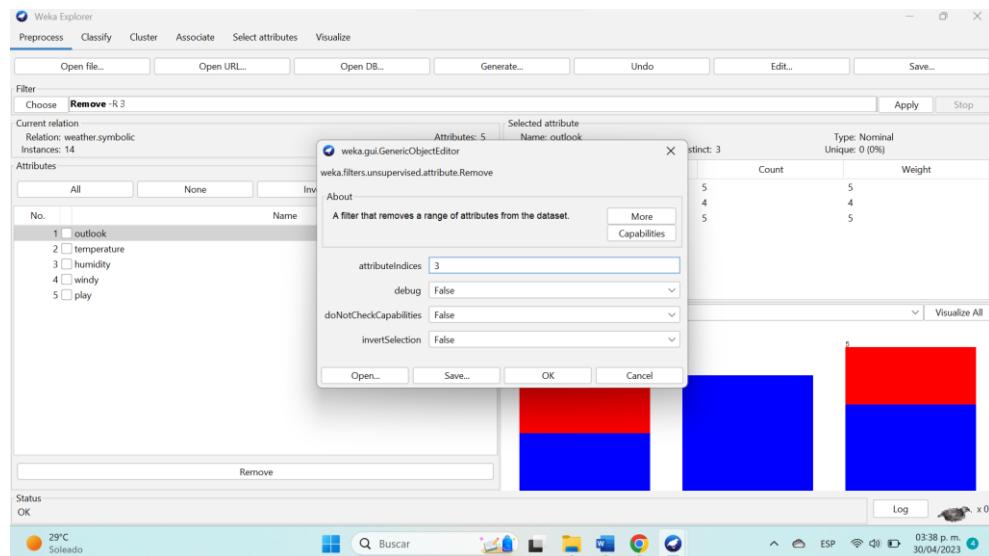


WEKA – 1.5 Using filter

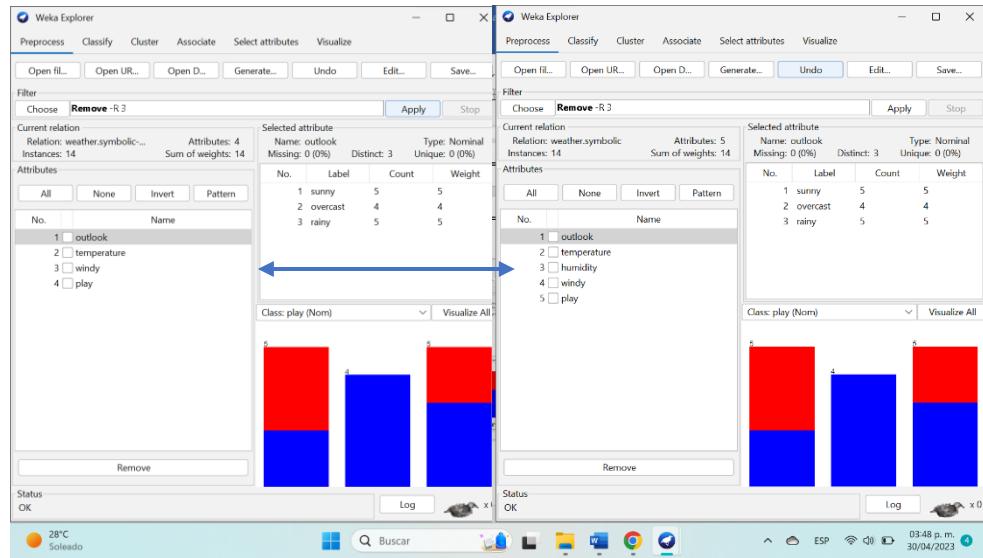
Veremos los tipos de filtración para datos que se pueden ocupar, en este caso ejercicio se ocupara el filtrado de **remove** que se encuentra en la carpeta unsupervised → attribute.



Procedemos a visualizar las propiedades del filtrado y a cambiar unos valores como lo puede ser el rango de los atributos e invertir la selección y al finalizar de modificar los parámetros le damos en el botón aplicar.

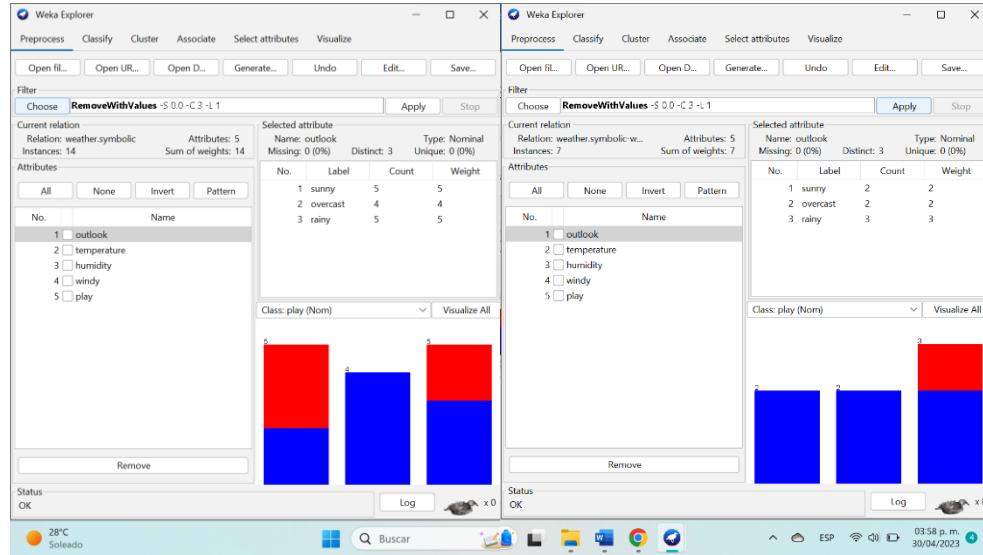


Una vez aplicando el cambio de parámetro, esto quiere decir que eliminamos el atributo numero 3, el cual es “Humedad” es así como podemos eliminar un atributo, lo cual al finalizar con este ejemplo, reinvertimos el cambio y el atributo “Humedad” vuelve aparecer.



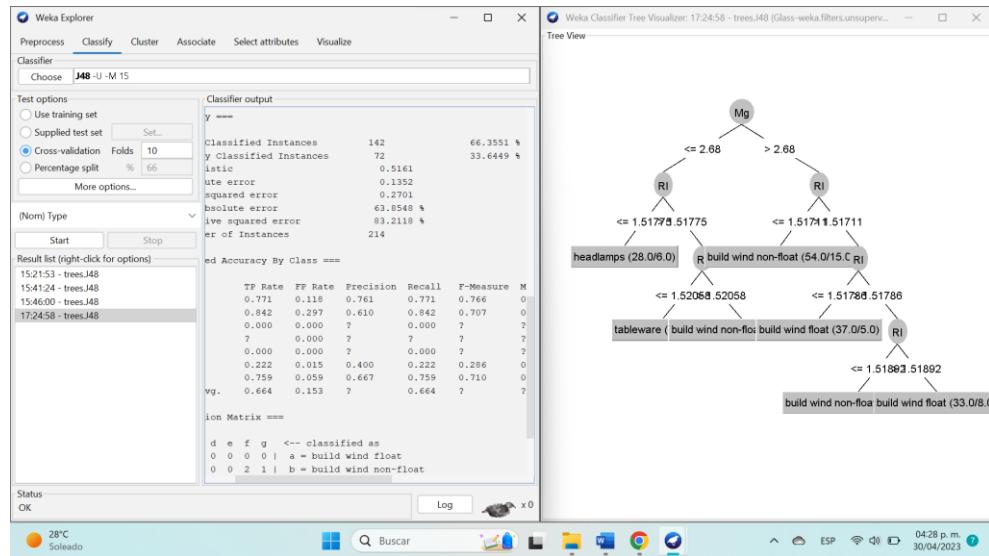
El otro tipo de filtrado es de la carpeta **instance** el cual es remover con valores (RemoveWithValues) y en donde podemos también modificar los distintos parámetros que puede haber en el tipo de filtrado. Al aplicar el filtro se eliminan algunos valores.

- Lado izquierdo filtro sin aplicar y lado derecho filtro ya una vez aplicado.



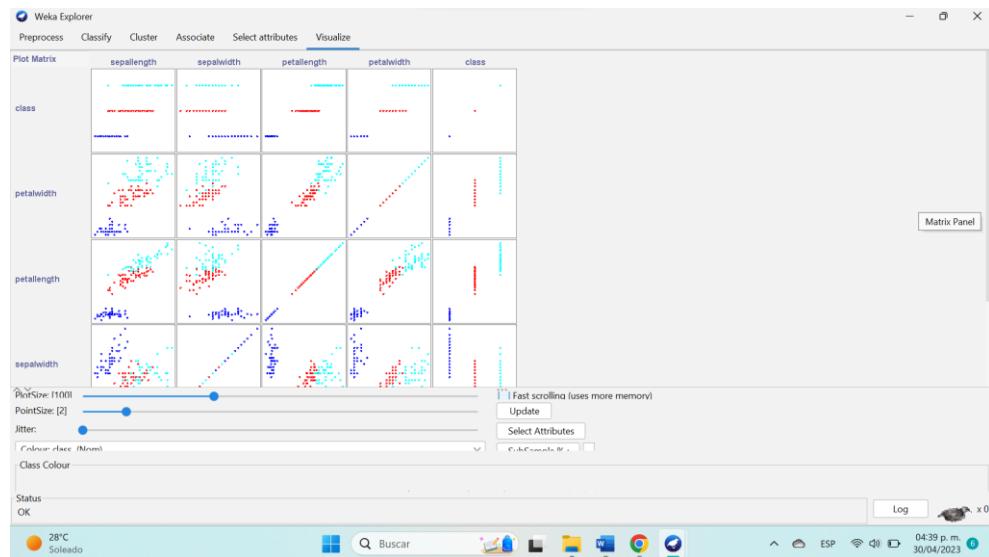
El otro ejemplo que podemos visualizar la precisión de los datos del conjunto, haciendo uso de la clasificación j48 y usando el mismo filtrado, también eliminando algunos tipos.

Esto lo que busca es poder ver la precisión de los datos y ver de una mejor manera el árbol que se crea.



WEKA – 1.6 Visualizing your data

Abrimos el ejercicio de **Iris** y podemos ver que hay un apartado donde aparecerá una matriz de parcelas bidimensionales de 5x5 del tipo de datos que hay.

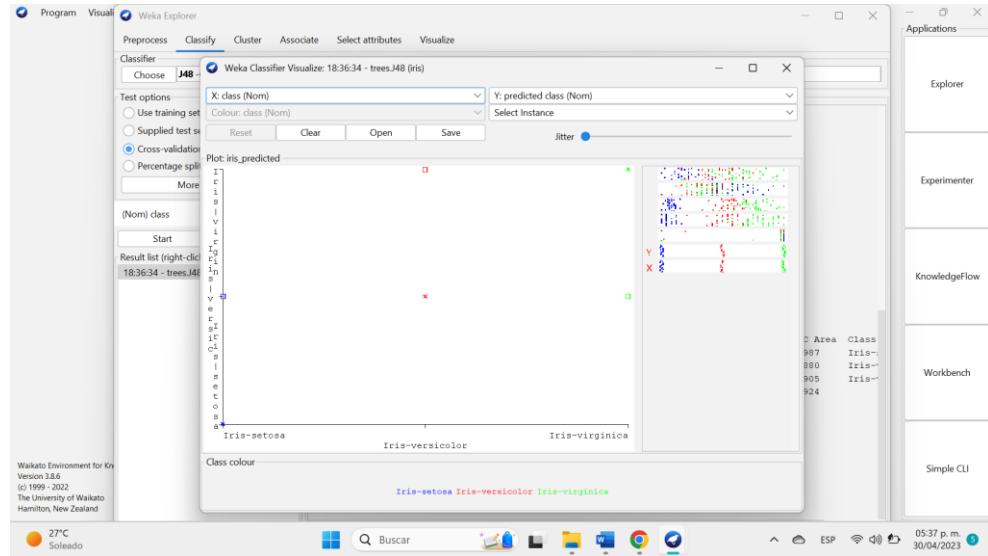


Una vez visualizamos esto, elegimos un cuadro para ver los datos más a fondo, el cual es **Petalwidth – sepawidth**. En donde este cuadro abarca 3 clases y podemos modificar los colores de cada punto que existe.

Y al darle click a cada punto, nos despliega una ventana respecto a la información que contiene es punto en específico.

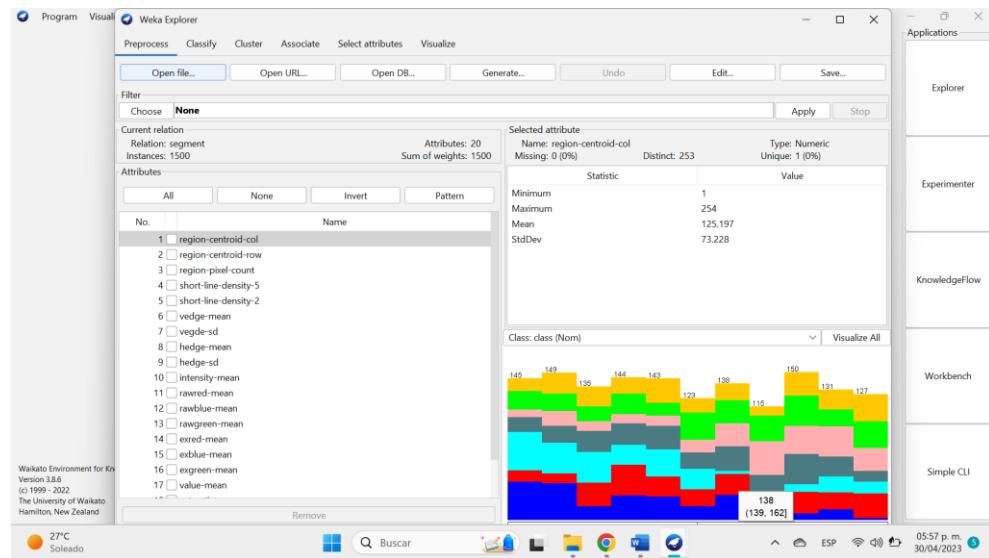


Después en el mismo ejercicio, podemos hacer una clasificación con j48 y ver la visualización de los errores que hay en los diferentes ejes. En donde cada cuadro del cuadrante es un error que dándole click se ve más información del mismo.

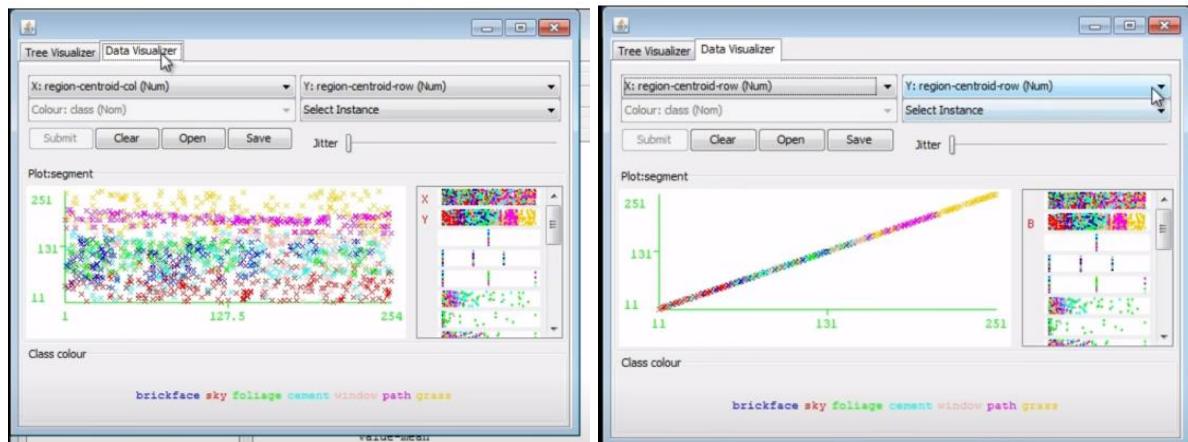


WEKA – 2.1 Be a classifier

Se trabajará con el conjunto de datos llamado **segment-challenger** que trae valores de clase como lo son: ladrillo, cielo, cemento, ventana, etc.



- Usaremos un clasificador el cual es UserClassifier (clasificador de usuarios), antes de empezar a trabajar en la opción cambiaremos a un “equipo de pruebas suministrado”.
- En donde aparecerá un mapa con los atributos de diferente color y se trazará una línea centroide para los datos que hay en el visualizador de datos.

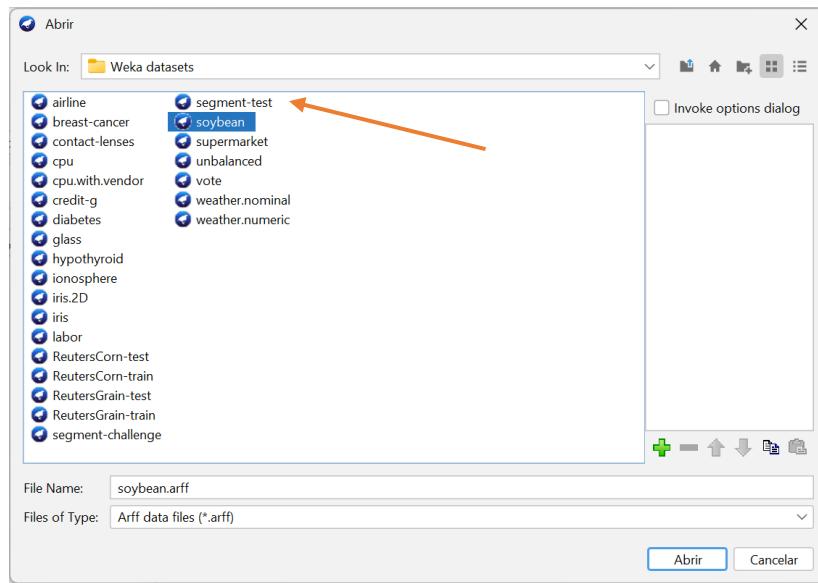


- Por último, el ejercicio trata de ir haciendo un árbol con los conjuntos de datos que vamos seleccionando, podemos seleccionar un cuadrado o hacer una figura que servirán para abarcar un área de datos y mandarlo al árbol.

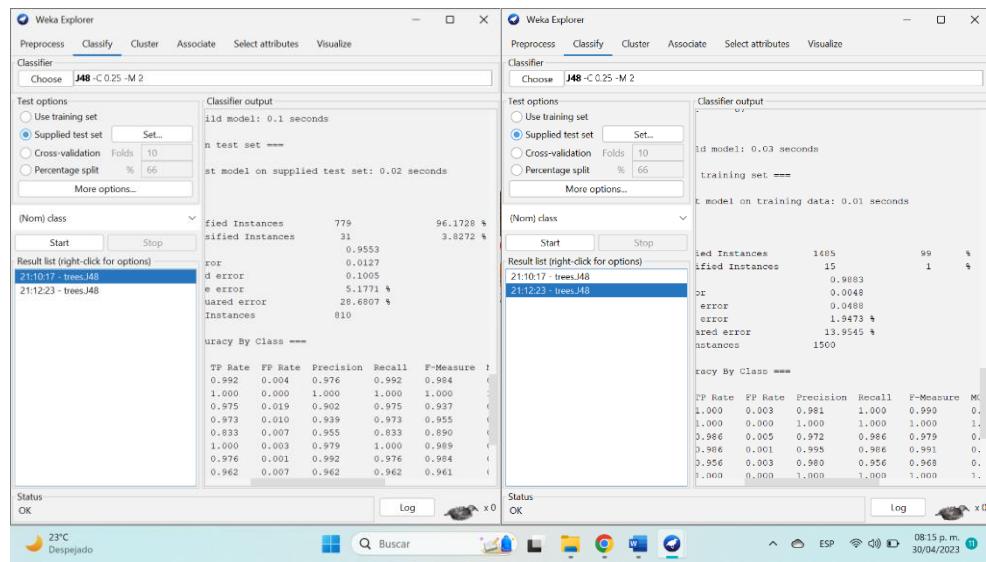
WEKA – 2.2 Training and testing

Trabajaremos con el ejercicio segment challenge, el cual usaremos el clasificador j48.

Una vez que elegimos j48, volvemos a elegir un equipo para pruebas. Antes de finalizar tenemos que agregar el archivo de pruebas el cual es el siguiente:

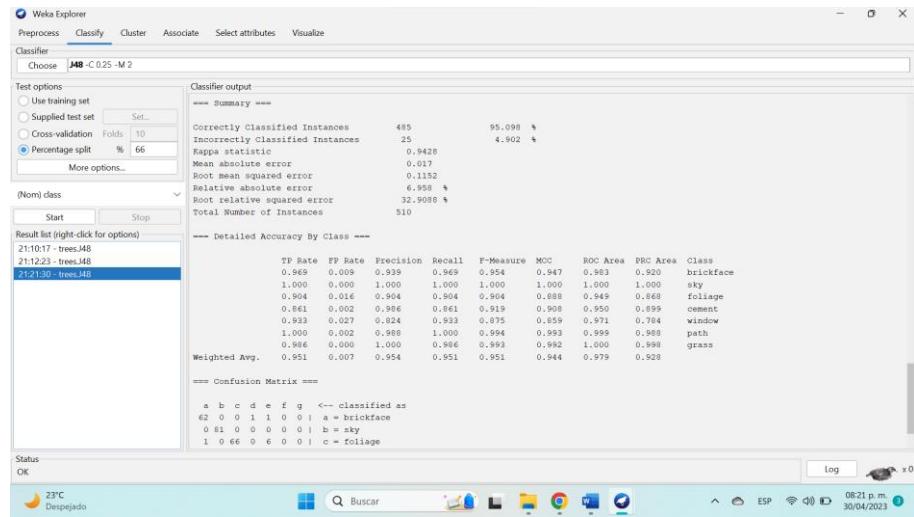


La tasa de precisión es del %96, lo cual j48 es mucho que el clasificador de usuarios, pero si usamos los datos de entrenamiento vemos que son un poco engañosos (imagen derecha) a diferencia de la 1ra opción (imagen izquierda).



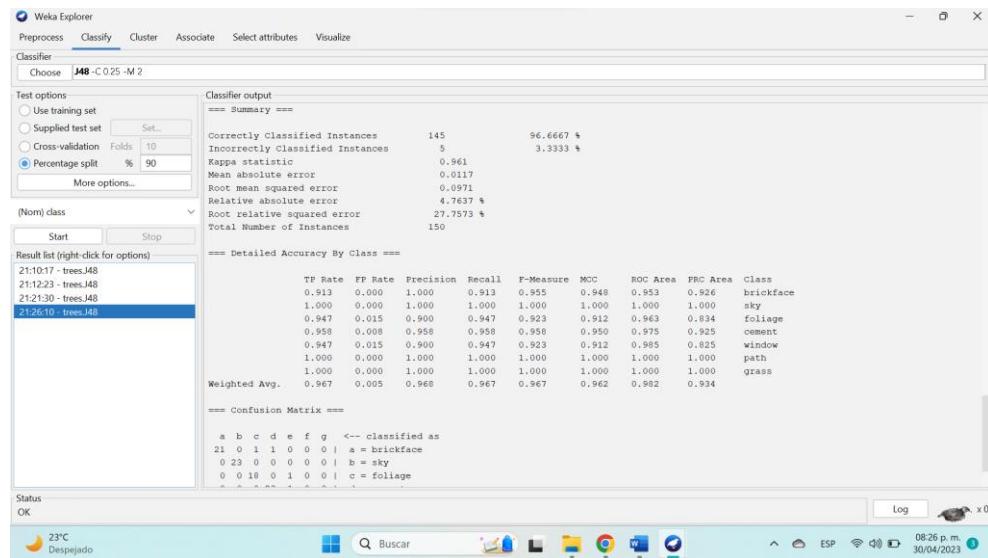
En el caso de que nos falten datos para hacer el análisis, tenemos la opción de “Percentage Split” que básicamente da el 66% de datos de entrenamiento y el %34 de los datos de prueba del archivo.

Lo volvemos a ejecutar y nos aparecerá una precisión del %95, pero si queremos ver resultados diferentes debemos cambiar sus valores.



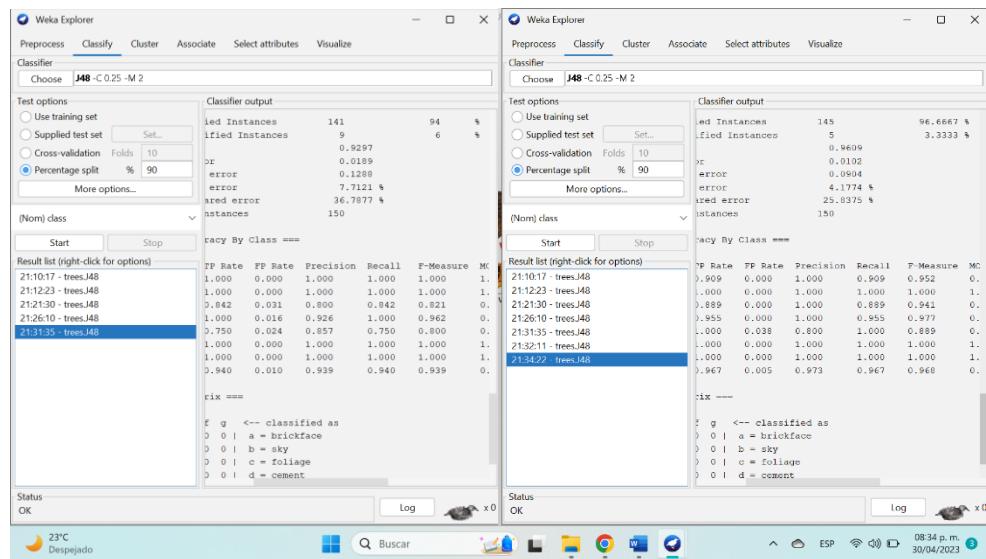
WEKA – 2.3 Repeated training and testing

En este ejercicio se volverá a trabajar con el ejercicio del anterior punto, siguiendo con el mismo clasificador y con el valor de 90% de entrenamiento y %10 de pruebas y vemos que es el mismo porcentaje de precisión que cuando había %66 de entrenamiento.



Se volverá hacer el calculo mediante este proceso, pero modificando el numero de la semilla aleatoria a 2.

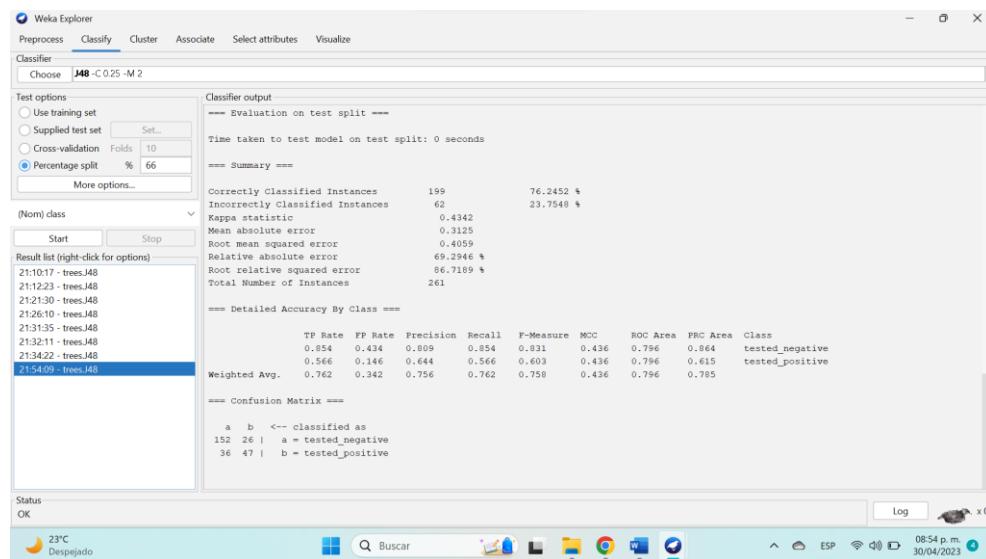
La precisión salió del **%94 (imagen derecha)**. Después se volvió a ejecutar con la semilla aleatoria en 4 y la precisión fue **%96 (imagen izquierda)**.



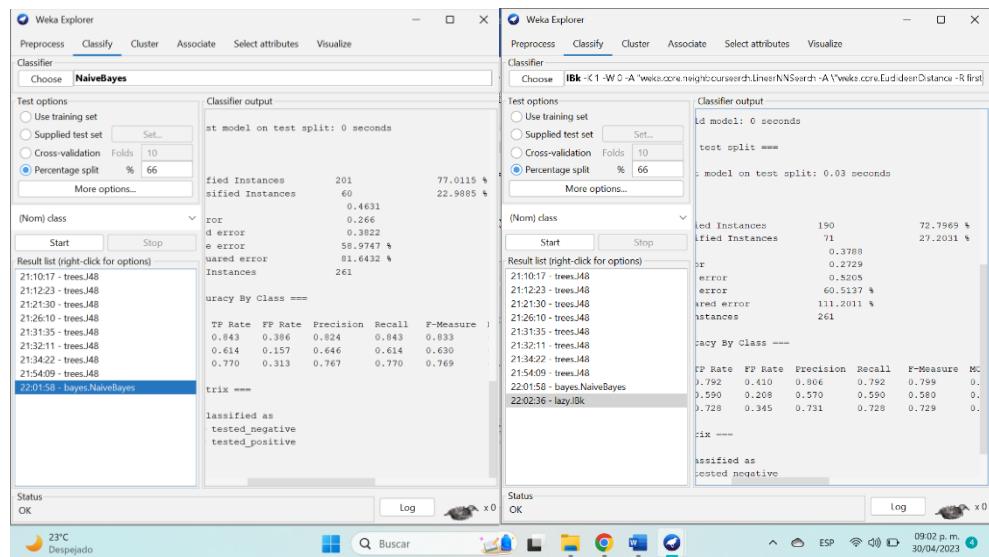
Los resultados son a base de funciones como lo son la media, desviación estándar, variancia, etc.

WEKA – 2.4 Baseline accuracy

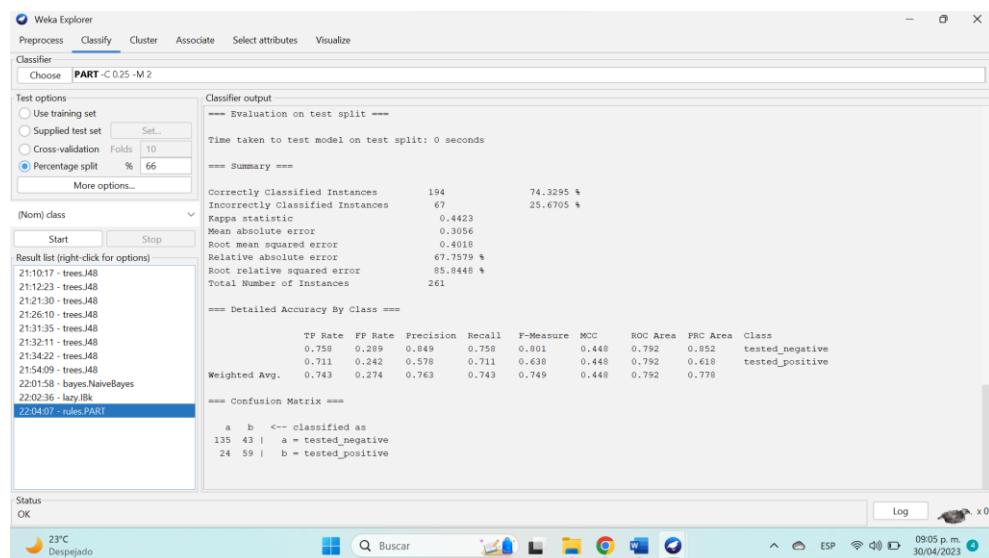
Trabajaremos con el archivo de diabetes en los indios y que contiene dos clases; datos positivos y datos negativos. Pasamos a dividir en porcentajes y los valores los dejamos por default con el clasificador j48, en el cual obtenemos **%76** de precisión.



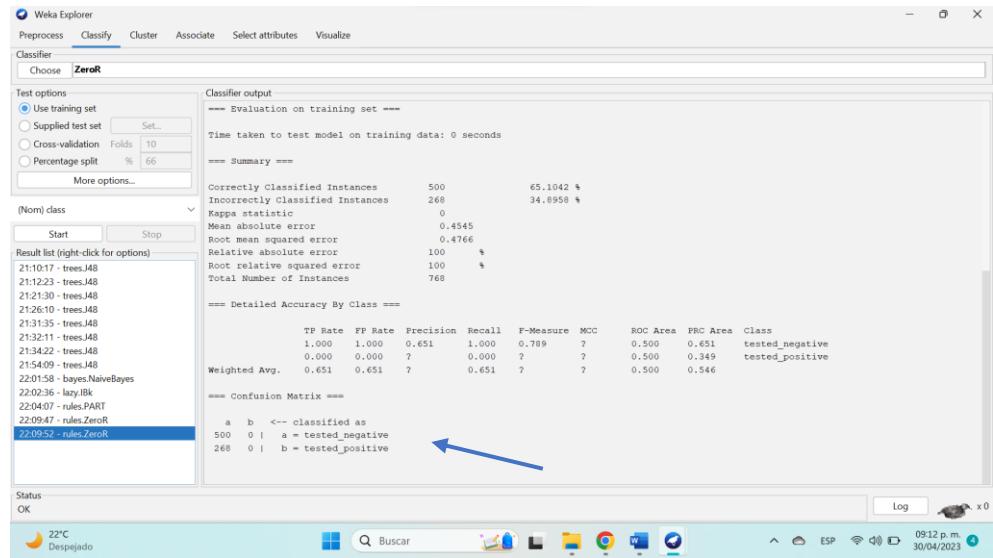
Ahora aplicamos el clasificador NaiveBayes y obtenemos una precisión de **%77** (img izquierda) y luego utilizamos el IBK con una precisión del **%72** (img derecha).



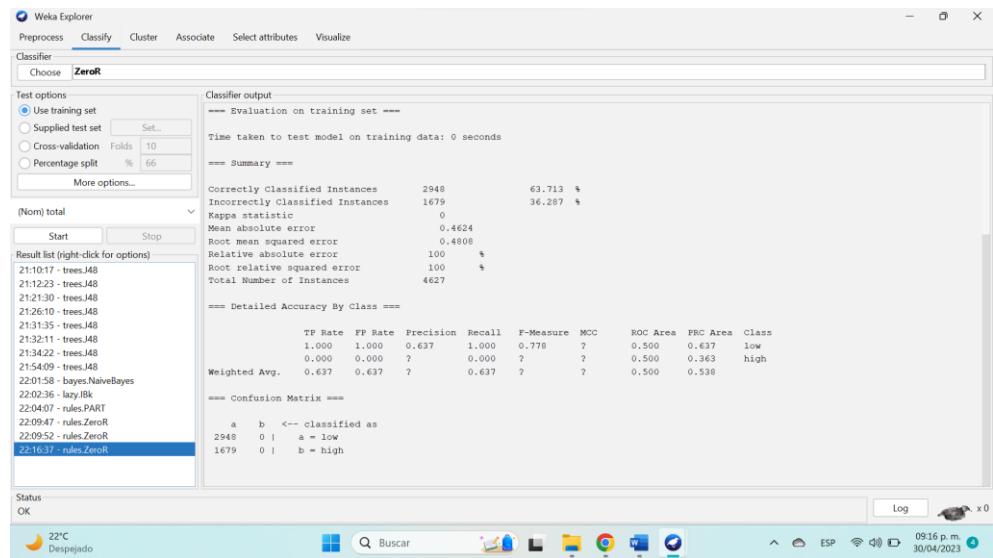
Después aplicamos PART con una precisión de **%74**.



Ocuparemos el clasificador ZeroR para encontrar la clase más probable con los datos de entrenamiento, contando que los datos negativos son 500 y el total de datos sería 768, que en este caso adivina los negativos sobre 500/768.



Aplicaremos ZeroR en otro archivo con le conjunto de datos de un supermercado y obtenemos una precisión de **%64**.

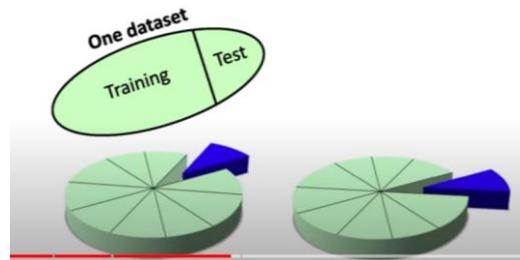


Repetimos el mismo método de comparar con algunos clasificadores del punto anterior y rondan porcentajes muy similares. Por lo tanto, los resultados muestran que la precisión de la línea base es en realidad mayor que la precisión de cualquiera de los clasificadores.

WEKA – 2.5 Cross-validation

En este capítulo habla sobre la validación cruzada el cual es un método para reducir el error, también la validación cruzada estratificada.

La validación cruzada es un ejemplo, que un conjunto de datos que tengamos lo dividimos en 10 y hacemos una simulación 10 veces. En donde usamos otro conjunto de 9 como datos de entrenamiento.

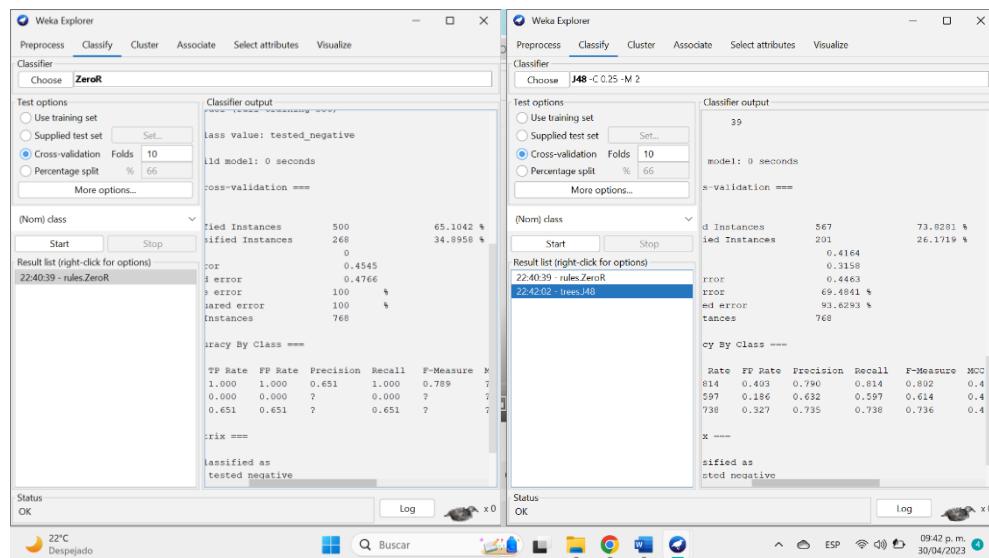


WEKA – 2.6 Cross-validation results

En este capítulo usaremos el conjunto de datos “Diabetes” y el clasificador ZeroR al ejecutarlo se evalúa con validación cruzada, pero para conseguir una línea base debemos utilizar los datos de entrenamiento.

Imagen izquierda es el clasificador zeroR con la precisión de **%65**.

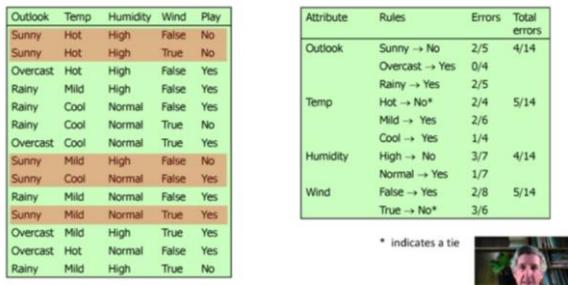
Imagen derecha usando J48 nos da la precisión de **%73** esto es dependiendo a la semilla de aleatoriedad que pongamos.



WEKA – 3.1 Simplicity first!

En este paso se trabajará con el clasificador OneR, que busca la manera de simplificar.

Por ejemplo, utilizamos el OneR para un conjunto de datos sobre el estado del tiempo y se lo aplicamos a cada uno.



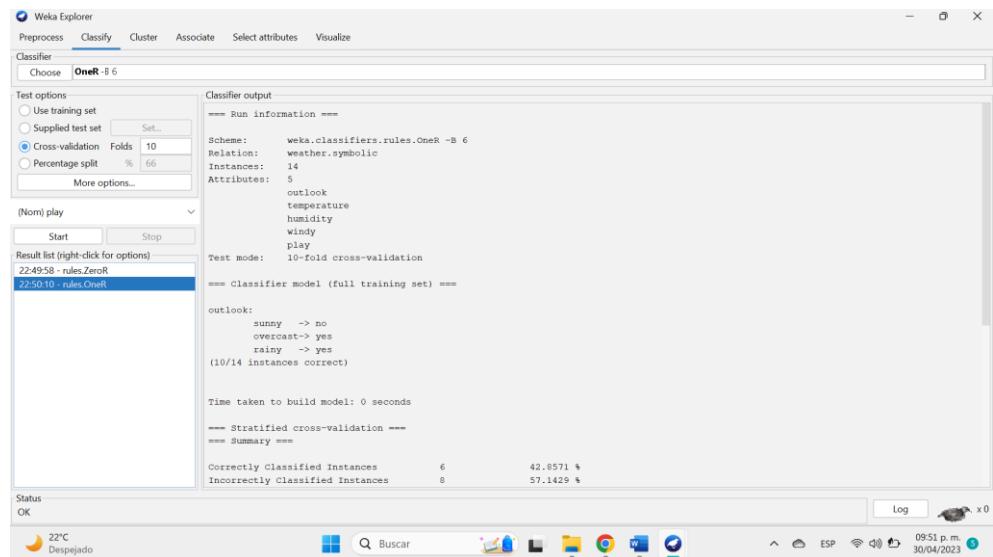
The image shows two tables side-by-side. The left table is a 14x5 grid representing the 'weather.nominal' dataset with columns: Outlook, Temp, Humidity, Wind, and Play. The right table is a summary of the generated rules, showing the attribute, rule, error count, and total errors. Below the tables is a small portrait of a man.

Outlook	Temp	Humidity	Wind	Play
Sunny	Hot	High	False	No
Sunny	Hot	High	True	No
Overcast	Hot	High	False	Yes
Rainy	Mild	High	False	Yes
Rainy	Cool	Normal	False	Yes
Rainy	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Sunny	Mild	High	False	No
Sunny	Cool	Normal	False	Yes
Rainy	Mild	Normal	False	Yes
Sunny	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Rainy	Mild	High	True	No

Attribute	Rules	Errors	Total errors
Outlook	Sunny → No Overcast → Yes Rainy → Yes	2/5 0/4 2/5	4/14
Temp	Hot → No* Mild → Yes Cool → Yes	2/4 2/6 1/4	5/14
Humidity	High → No Normal → Yes	3/7 1/7	4/14
Wind	False → Yes True → No*	2/8 3/6	5/14

* indicates a tie

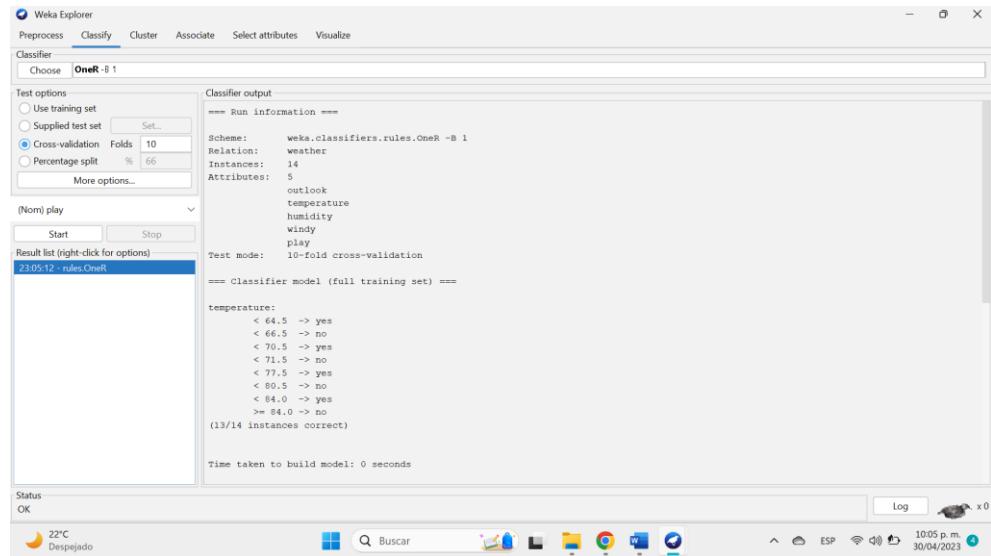
Cargamos el archivo de weather nominal, usamos el OneR con un conjunto de entrenamiento y obtenemos 10/14 instancias correctas, claramente no es muy razonable para un conjunto pequeño.



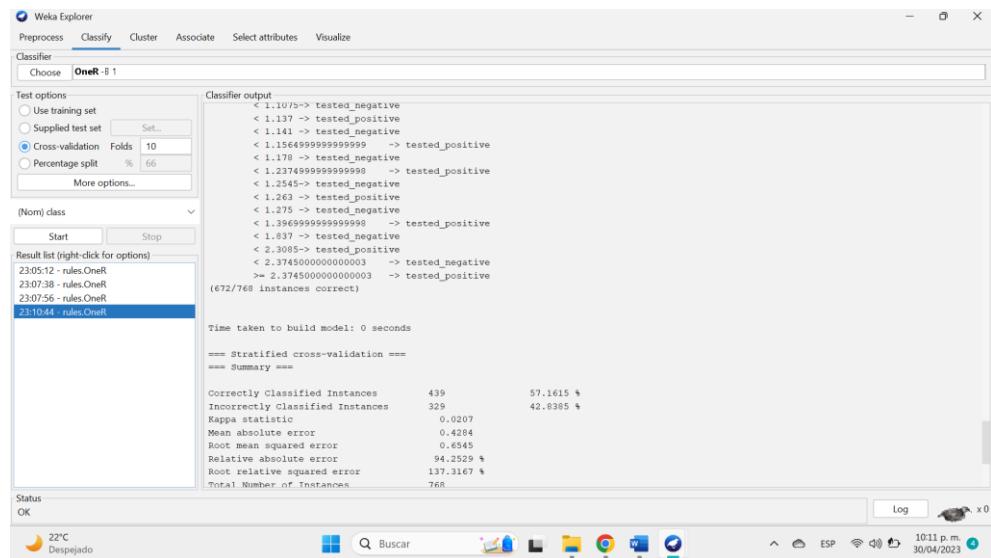
WEKA – 3.2 Overfitting

Realizamos el trabajo con el clasificador OneR y con el ejercicio weather numeric, en donde crearemos una regla que crean en función del atributo de perspectiva.

Para el análisis eliminaremos el atributo de clase Outlook y utilizaremos la configuración predeterminada de OneR, en un parámetro de OneR cambiaremos de 6 a 1, para ver el tipo de reglas que está dando.

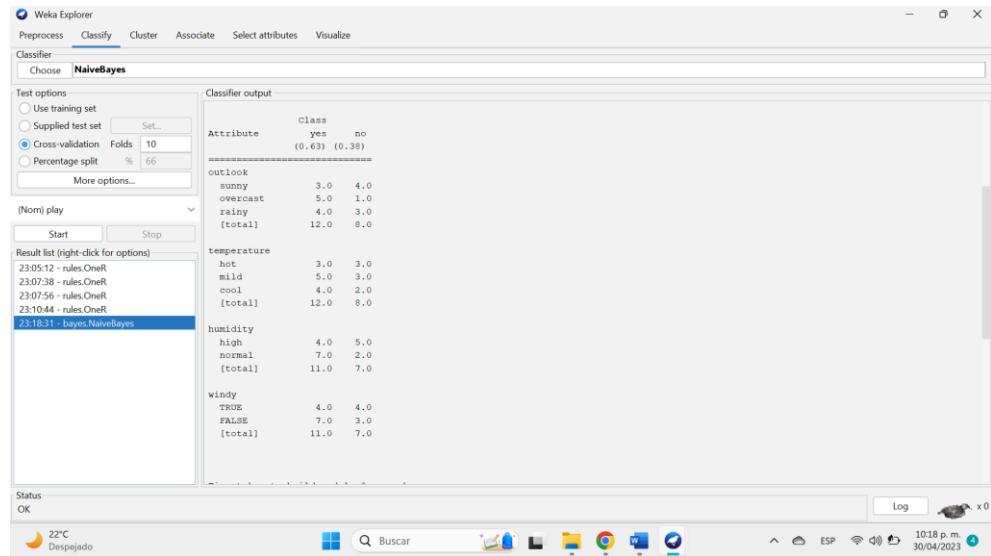


Ahora vemos el análisis del conjunto de datos diabetes que cuenta con dos clases ya antes mencionadas aplicando el mismo valor cambiada en la prueba anterior a 1 y vemos las reglas que da las cuales son demasiadas y retorna una precisión del **%57**.



WEKA – 3.3 Using probabilities

Utilizaremos el clasificador NaiveBayes con el ejercicio que se viene manejando en el anterior punto, el cual es el de weather nominal. En donde la probabilidad del éxito se obtiene por validación cruzada.

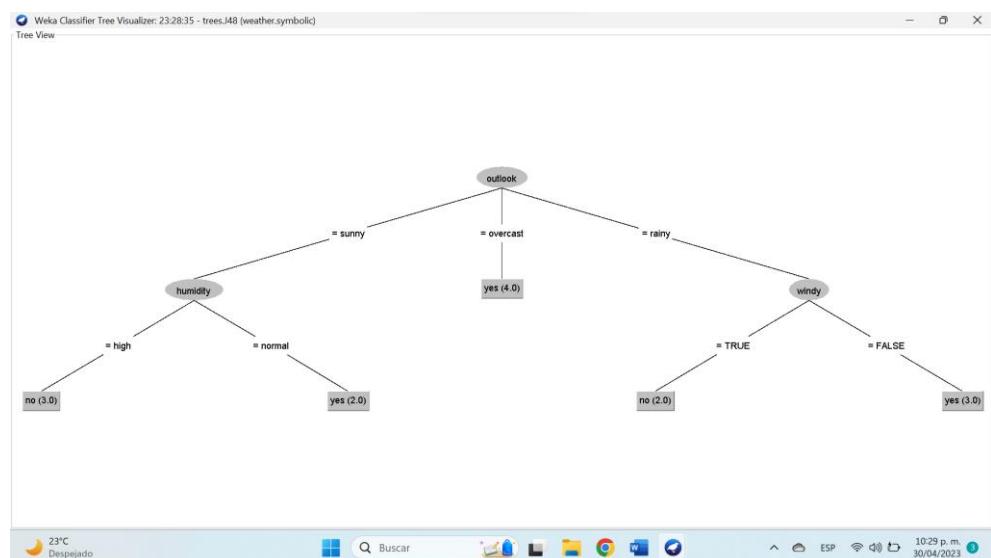


En este método se concluye que su premisa es para lo que se va a predecir y que todos los atributos son igual de importantes e independientes entre sí.

WEKA – 3.4 Decision trees

Como herramientas utilizaremos J48. Una vez cargado el ejercicio de weather nominal y el clasificador listo, nos mostrara un árbol en donde tren 3 instancias dependiendo el dato.

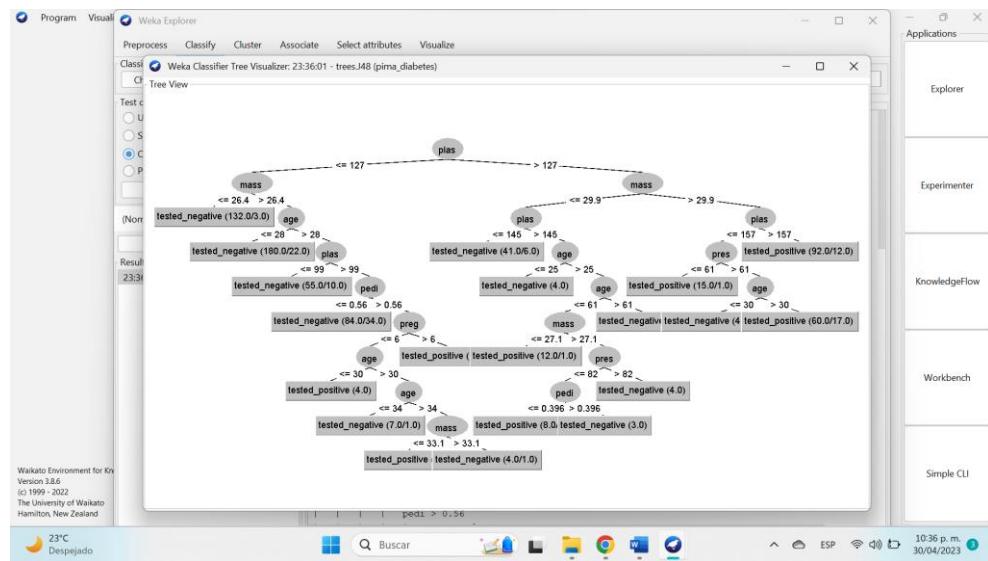
Por lo tanto para la validación de este árbol se hace por el método de validación cruzada.



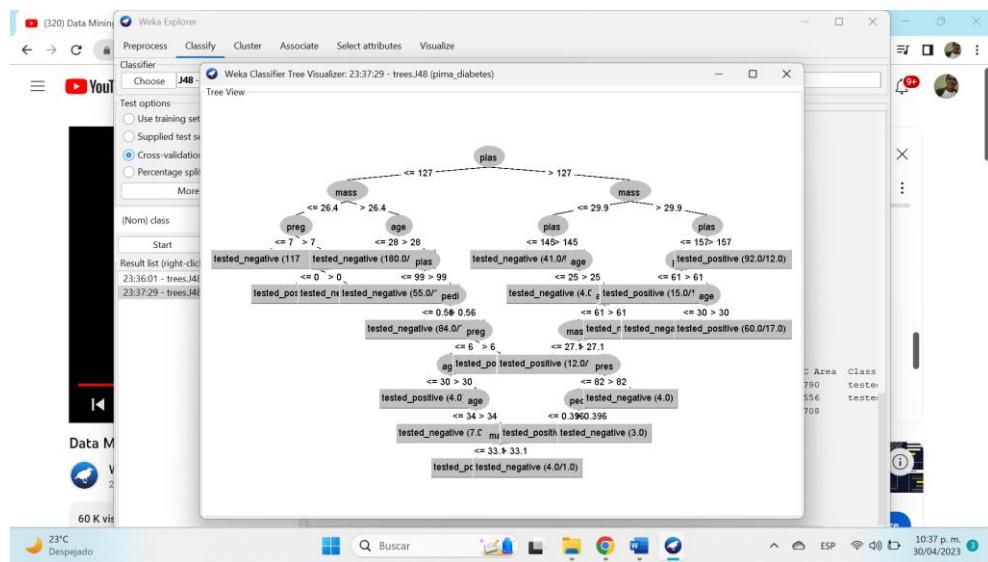
WEKA – 3.5 Pruning decision trees

Hay técnicas simples y complejas para decisión de tomas en un árbol.

Una técnica simple es dejar de dividir el nodo que contiene muy pocas instancias. El análisis se hará para el ejercicio de diabetes con J48. En donde tendrá una validación cruzada del **%73.8** y contiene 39 nodos con 20 nodos hojas y 19 internos.



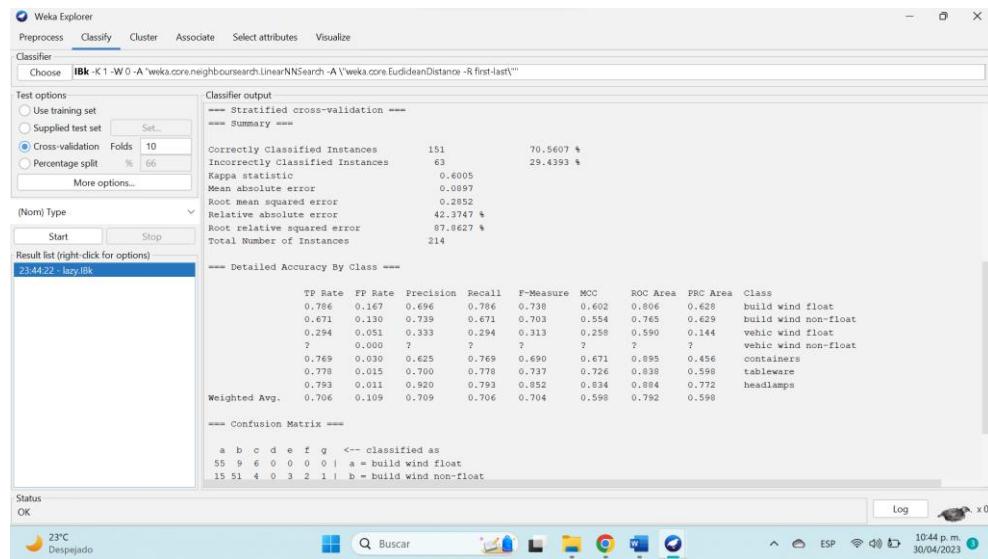
Al cambiar un valor de un parámetro, tendremos un árbol de dimensiones más grandes, por ejemplo: 22 hojas de 43 nodos que habrá.



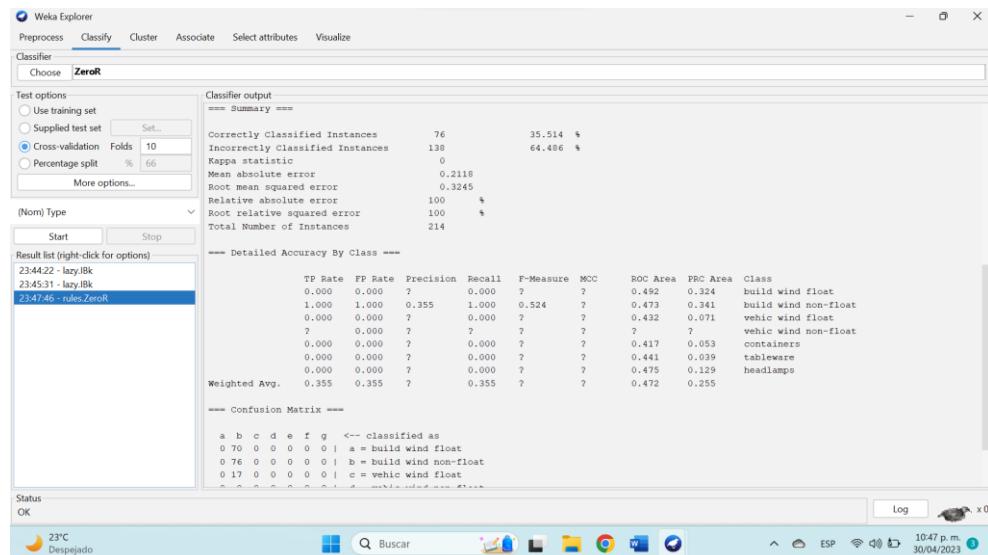
Si hacemos esto con el ejercicio sobre datos del cáncer, obtenemos dos árboles: un árbol pequeño y un árbol grande esto debido a una sola modificación de un valor.

WEKA – 3.6 Nearest neighbor

Utilizaremos el método del k-vecino más cercano. Con el archivo glass utilizaremos el clasificador IBK obteniendo un porcentaje del **%76** y aumentando un número es un poco pero el porcentaje de precisión **%67**.

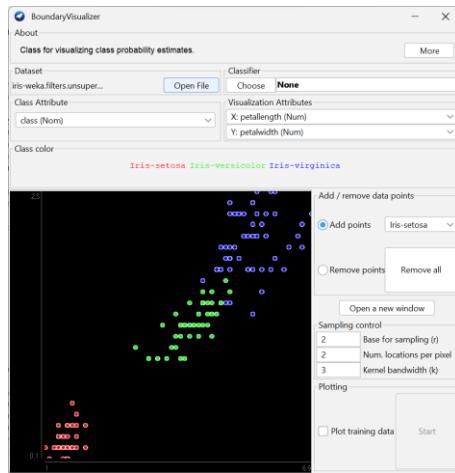


Al hacer que encuentre las ultimas 100 instancias regresa un porcentaje de precisión del **%35** el cual comparando con el método zeroR es el mismo.

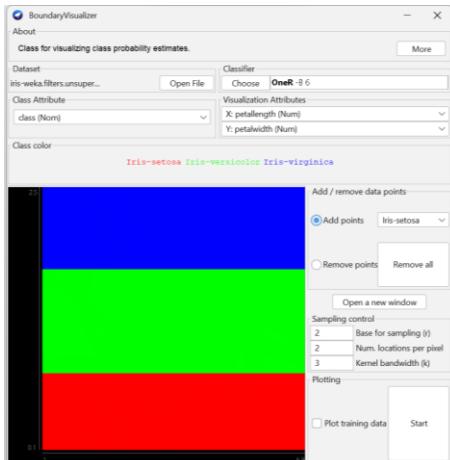


WEKA – 4.1 Classification boundaries

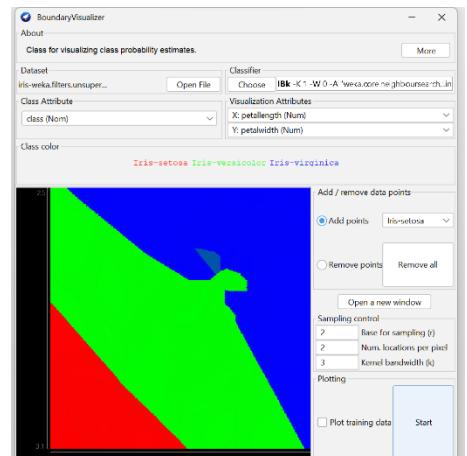
Abrimos el conjunto de datos iris2D y también utilizaremos el visualizador de límites y vemos una gráfica de datos.



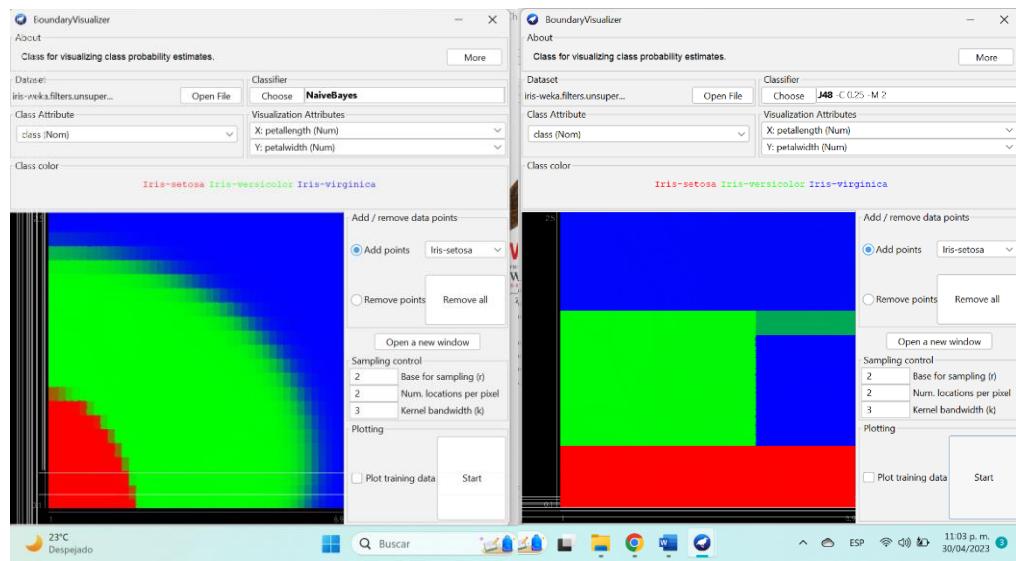
Después de verlo en una grafica procedemos aplicarle un clasificador oneR para que nos muestre la diferencia del punto anterior sin clasificador.



Ahora aplicamos el método IBK en donde obtendremos datos diferentes a los anteriores. Al cambiar las ultimas instancias el gráfico se vera muy diferente.

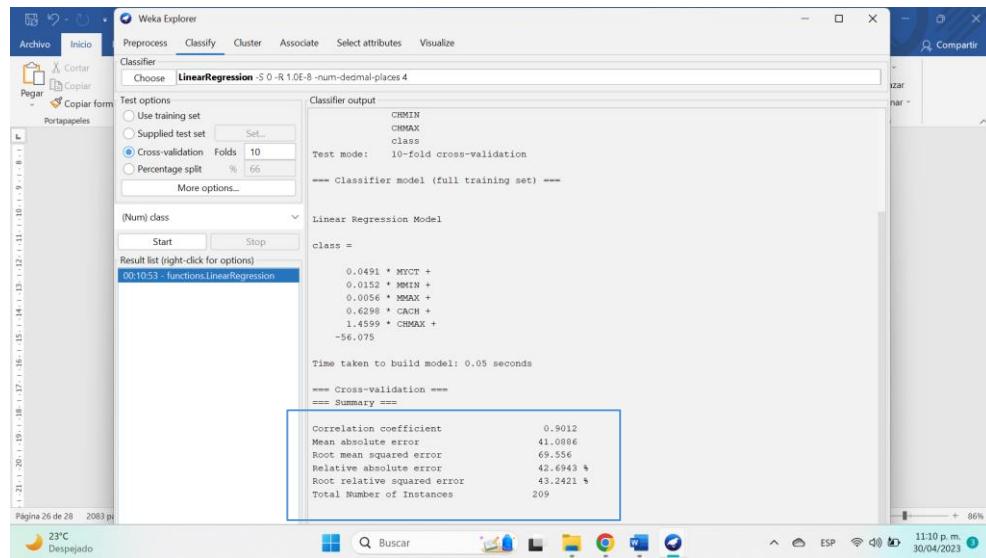


Aplicaremos los siguientes métodos que es el NaiveBayes (imagen izquierda) y J48 (imagen derecha).



WEKA – 4.2 Linear regression

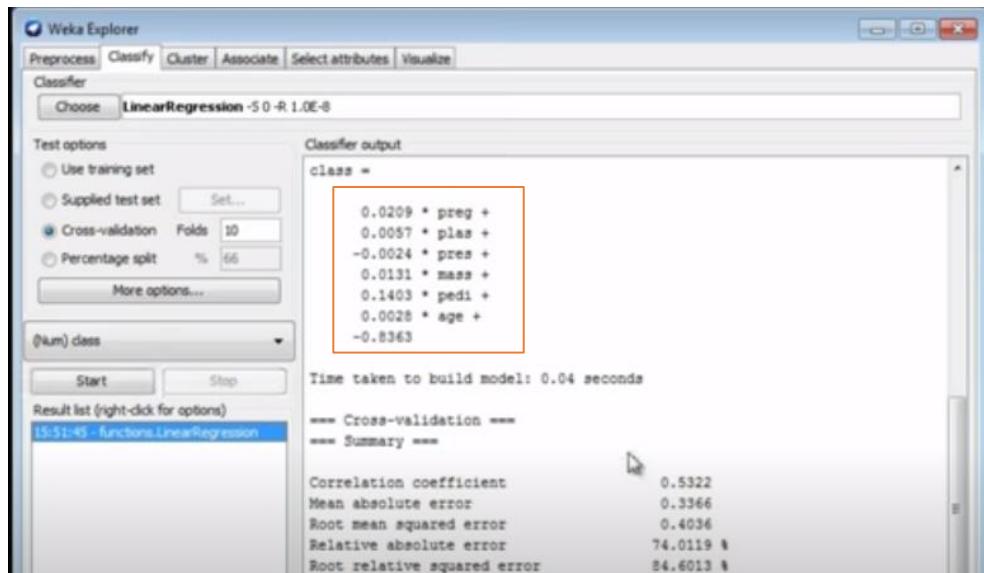
Procedemos a predecir el valor de un número con la función llamada LinearRegression con el archivo de datos de CPU. En la parte del rectángulo se usa una formula en la que se puede ver su tasa de éxito para los datos de entrenamiento.



Sus resultados son todos aproximados, pero sin antes debemos elegir las circunstancias específicas a elegir.

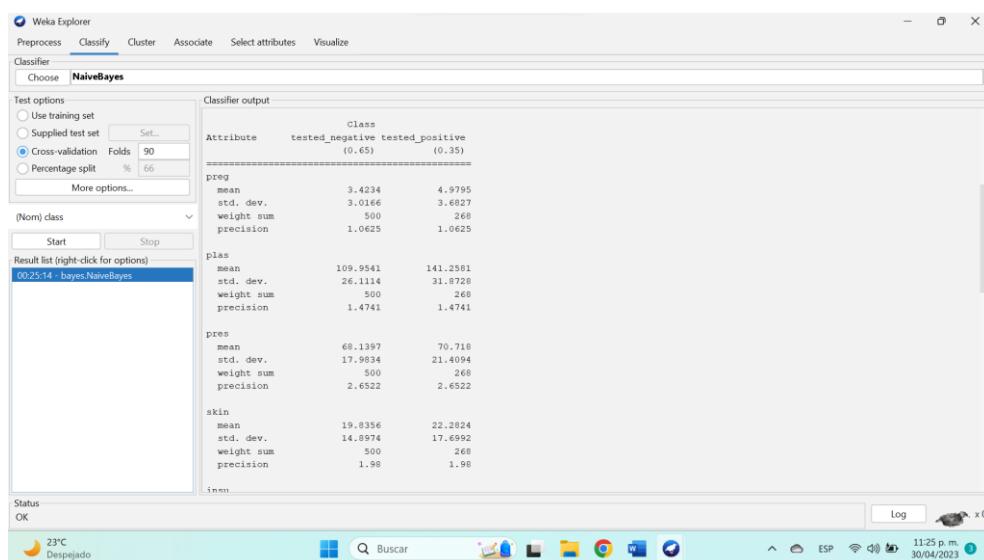
WEKA – 4.3 Classification by regression

Usaremos el archivo diabetes para usar el nominalToBinary, lo que haremos será convertirlos a datos numéricos, ya que la clasificación que usaremos solo ocupa datos numéricos, el rectángulo muestra la línea de regresión.

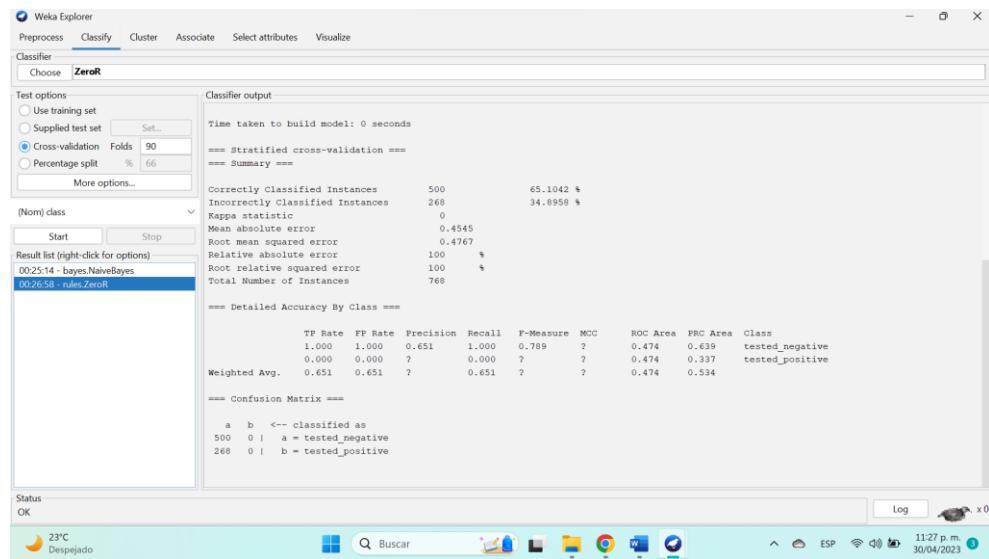


WEKA – 4.4 Logistic regression

Ejecutamos NaiveBayes para el archivo de diabetes, con la proporción de división del %90 y editamos la predicción de salida.



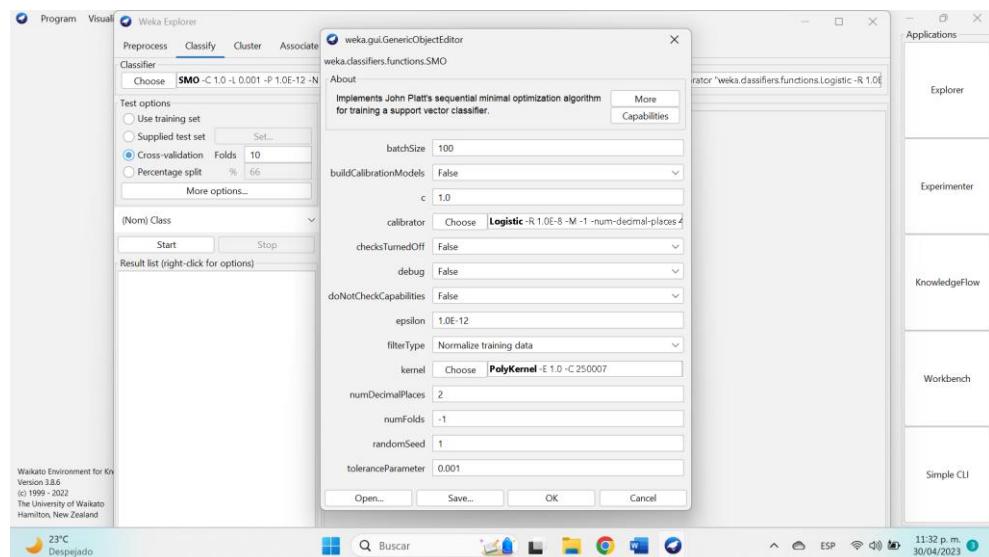
Volvemos aplicar el método zeroR para comparar unos datos del método anterior y confirmar mediante este método.



WEKA – 4.5 Support vector machines

En este punto se hablará sobre las máquinas de vector de soporte, otra forma de aprendizaje automático avanzado.

Abrimos las configuraciones de método SMO, en donde encontramos para entrenar clasificadores de máquinas de vectores de soporte se llama Optimización Mínima Secuencial y diferentes opciones de núcleo.



WEKA – 4.6 Ensemble learning

Esta es una nueva técnica del aprendizaje automático, a continuación, se analizarán cuatro métodos: embolsado, aleatorización, impulso y apilado.

- Embolsado: se necesitan varias estructuras de decisión diferentes
- Aleatorización: cuando se usa un árbol de decisión, se llama bosque aleatorio.
- Impulsar: es un algoritmo iterativo, una nueva ronda de modelos que se basan en los resultados de clasificación de modelos anteriores.
- Apilamiento: se puede elegir un meta clasificador diferente, así como el número de pliegues.

WEKA – 5.1 The data mining process

WEKA – 5.2 Pitfalls and pratfalls

WEKA – 5.3 Data mining and ethics

WEKA – 5.4 Summary