Universidad de Córdoba

Máster en Inteligencia Computacional e Internet de las Cosas

Memoria de Prácticas

Autor: Alberto Fernández Merchán
Asignatura: Análisis Automático de Datos para las Ciencias Biomédicas, Medioambientales, Agroalimentarias.

Profesor: Javier Sánchez Monedero
Junio 2025

Índice

Índice de cuadros

1. Introducción

El análisis de datos y la construcción de modelos predictivos se han convertido en una parte fundamental del aprendizaje automático y la ciencia de datos. Para facilitar estas tareas, existen diversas herramientas que permiten a los usuarios aplicar técnicas de preprocesamiento, selección de características, entrenamiento y evaluación de modelos sin necesidad de programación avanzada. Entre ellas, destaca **WEKA** (Waikato Environment for Knowledge Analysis) [1], una plataforma de software libre desarrollada por la Universidad de Waikato, ampliamente utilizada en la comunidad académica para el análisis y minería de datos.

Este trabajo tiene como objetivo explorar el ciclo completo de desarrollo de un modelo predictivo utilizando WEKA, desde la preparación de los datos hasta la evaluación de su rendimiento. A lo largo del documento se aplican distintas técnicas sobre dos conjuntos de datos: `Audiology_soft` y `Glass Identification`. El primero se emplea para ilustrar las fases de preprocesamiento, mientras que el segundo se utiliza como caso práctico para construir y evaluar modelos de clasificación.

En primer lugar, se abordan tareas esenciales de limpieza y transformación de datos, como la eliminación e imputación de valores perdidos, así como la conversión de atributos nominales a binarios. Posteriormente, se analiza un caso práctico donde se exploran las variables del conjunto de datos, su correlación y su distribución, permitiendo comprender mejor las características que pueden influir en la predicción.

A continuación, se estudia el efecto de la normalización y estandarización de los atributos sobre el rendimiento de varios algoritmos de clasificación implementados en WEKA, incluyendo `SimpleLogistic`, `J48` y `RandomForest`. Para cada caso, se presentan los resultados de entrenamiento y prueba utilizando validación cruzada, y se comparan los clasificadores a través de métricas como la precisión, el coeficiente de Kappa, el MAE, el RMSE, el F1-score, el coeficiente de correlación de Matthews (MCC) y el área bajo la curva ROC (AUC).

Finalmente, se extraen conclusiones sobre el comportamiento de los clasificadores ante diferentes preprocesamientos y se discuten las ventajas y limitaciones del uso de WEKA como entorno de análisis de datos.

2. Preprocesado de Audiology_soft

En este apartado se detalla el proceso de preprocesado aplicado al conjunto de datos *Audiology_soft*, que trata sobre la clasificación de enfermedades auditivas a partir de diversos atributos clínicos. El conjunto original contiene 226 instancias y 10 atributos, además de la clase objetivo (*class*), como se muestra en la Figura 1 y la Figura 2.

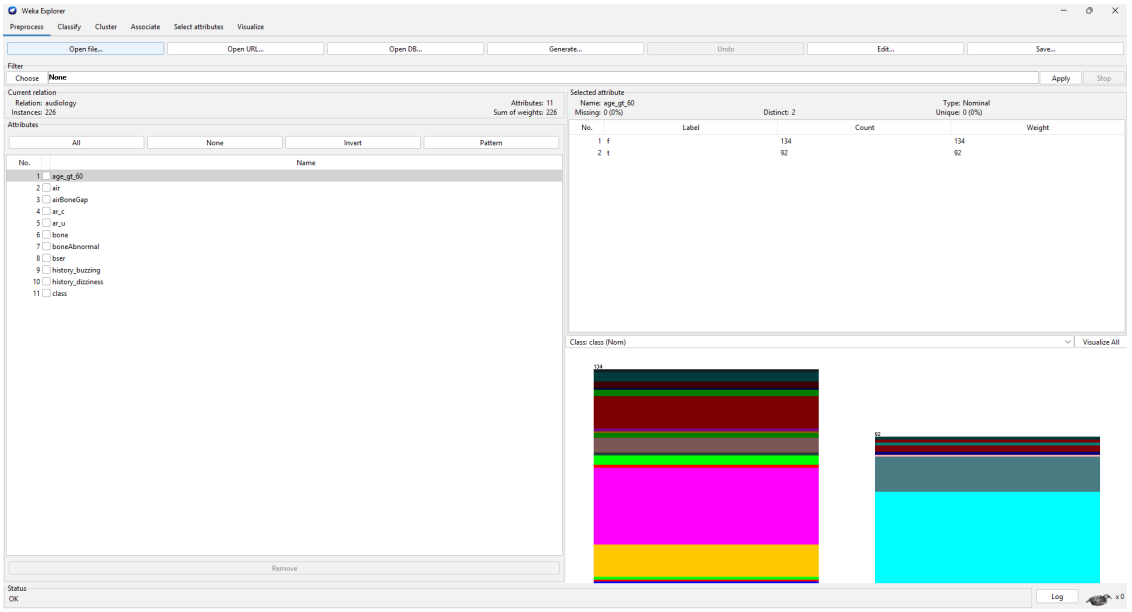


Figura 1: Pantalla de visualización de datos en Weka del dataset de Audiology_soft.

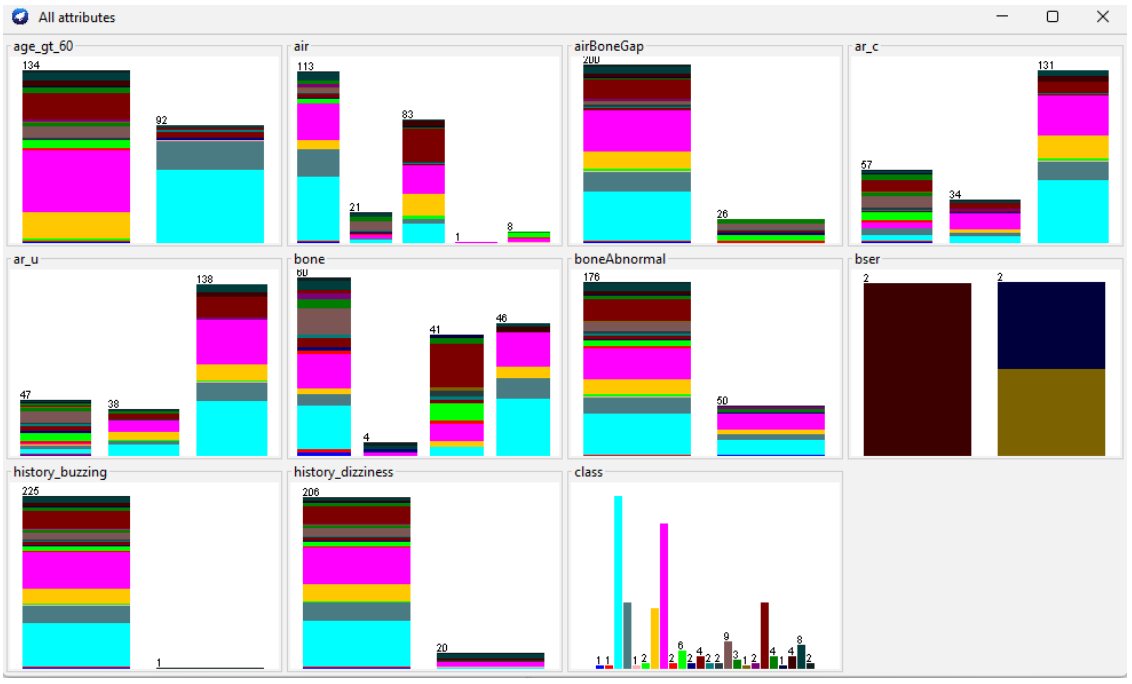


Figura 2: Distribución de los valores de los atributos del dataset.

2.1. Eliminación de atributos con demasiados valores perdidos

Se ha analizado el porcentaje de valores faltantes por cada atributo para determinar si eran susceptibles de imputación mediante la media o la moda. Se consideró que atributos con más de un 30 % de valores perdidos podrían introducir sesgos si se imputaban, por lo que estos fueron eliminados.

El atributo *bser* fue eliminado inmediatamente debido a que presentaba un 98 % de valores perdidos (Figura 3).

El atributo *bone* tenía un 33 % de valores faltantes (Figura 4), por lo que inicialmente se valoró su eliminación por simplicidad. Sin embargo, en un análisis de selección de atributos, se observó que *bone* tenía

Selected attribute

Name: bser
Missing: 222 (98%)
Distinct: 2
Type: Nominal
Unique: 0 (0%)

No.	Label	Count	Weight
1	degraded	2	2
2	normal	2	2

Figura 3: Información sobre el atributo *bser* con un 98 % de valores perdidos.

cierto valor predictivo (Figura 5).

Selected attribute

Name: bone
Missing: 75 (33%)
Distinct: 4
Type: Nominal
Unique: 0 (0%)

No.	Label	Count	Weight
1	mild	60	60
2	moderate	4	4
3	normal	41	41
4	unmeasured	46	46

Figura 4: Información sobre el atributo *bone* con un 33 % de valores perdidos.

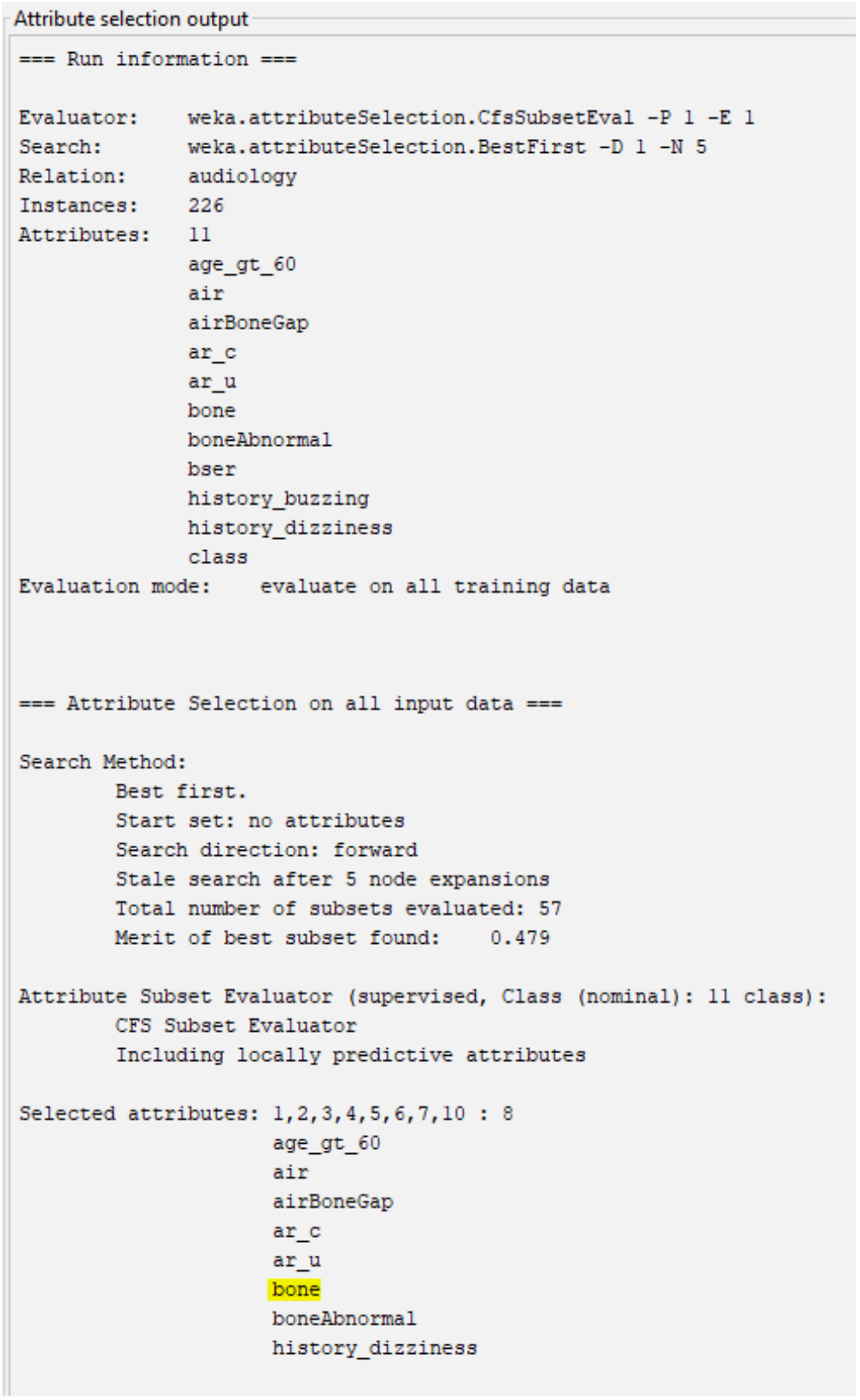
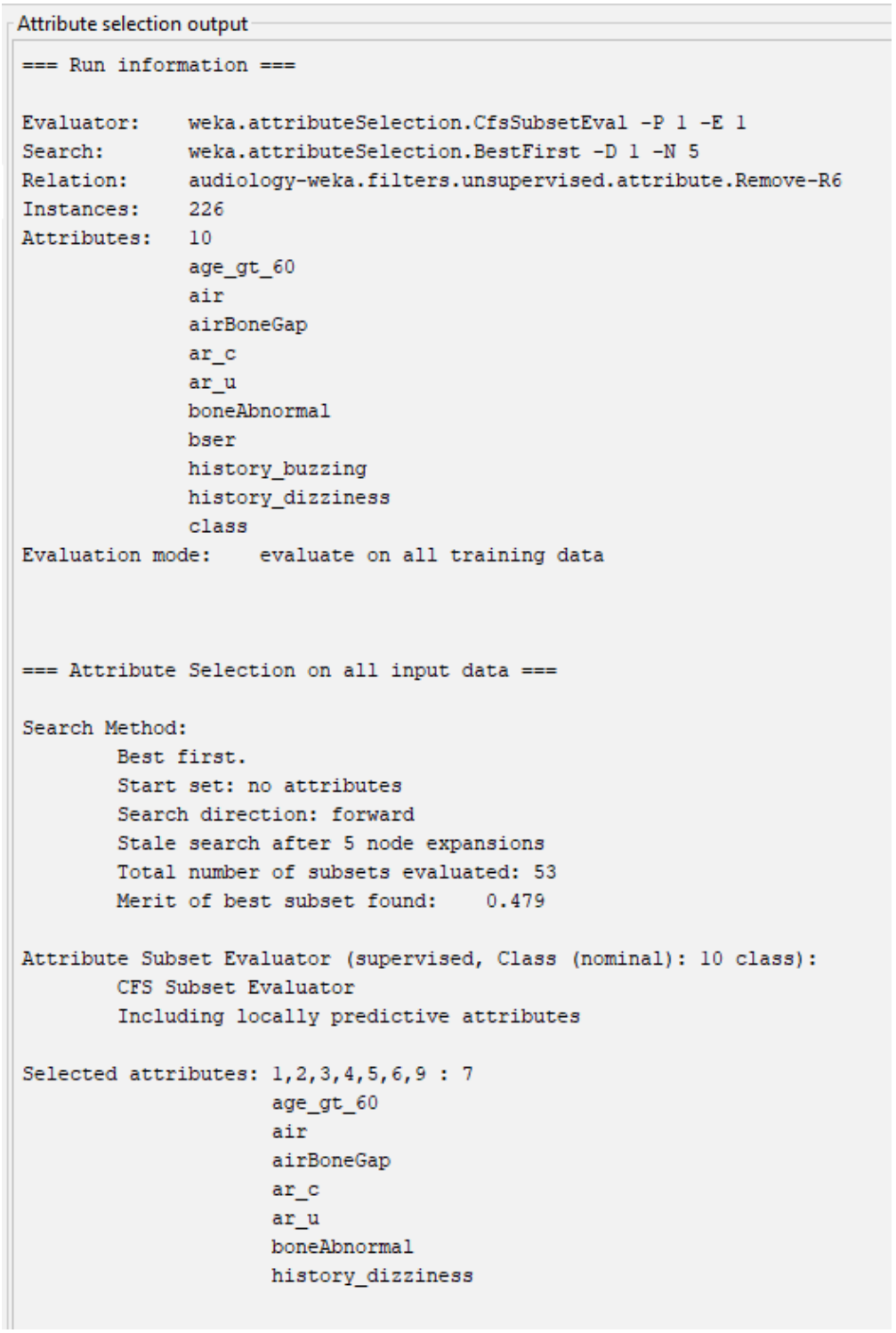


Figura 5: En la selección de atributos nos aparece que el atributo *bone* tiene valor predictivo.

Pese a ello, al repetir el proceso de selección sin dicho atributo, el mérito del subconjunto seleccionado fue idéntico (0.479), como se puede observar en la Figura 6, por lo que finalmente se decidió eliminar también el atributo *bone* para simplificar el conjunto.



Attribute selection output

```
=== Run information ===

Evaluator:    weka.attributeSelection.CfsSubsetEval -P 1 -E 1
Search:       weka.attributeSelection.BestFirst -D 1 -N 5
Relation:     audiology-weka.filters.unsupervised.attribute.Remove-R6
Instances:    226
Attributes:   10
              age_gt_60
              air
              airBoneGap
              ar_c
              ar_u
              boneAbnormal
              bser
              history_buzzing
              history_dizziness
              class
Evaluation mode:    evaluate on all training data



=== Attribute Selection on all input data ===

Search Method:
        Best first.
        Start set: no attributes
        Search direction: forward
        Stale search after 5 node expansions
        Total number of subsets evaluated: 53
        Merit of best subset found:    0.479

Attribute Subset Evaluator (supervised, Class (nominal): 10 class):
        CFS Subset Evaluator
        Including locally predictive attributes

Selected attributes: 1,2,3,4,5,6,9 : 7
                     age_gt_60
                     air
                     airBoneGap
                     ar_c
                     ar_u
                     boneAbnormal
                     history_dizziness
```

Figura 6: Selección de atributos sin incluir *bone*; el mérito del subconjunto permanece igual.

El dataset resultante queda reducido a 8 atributos más la clase (Figura 7).

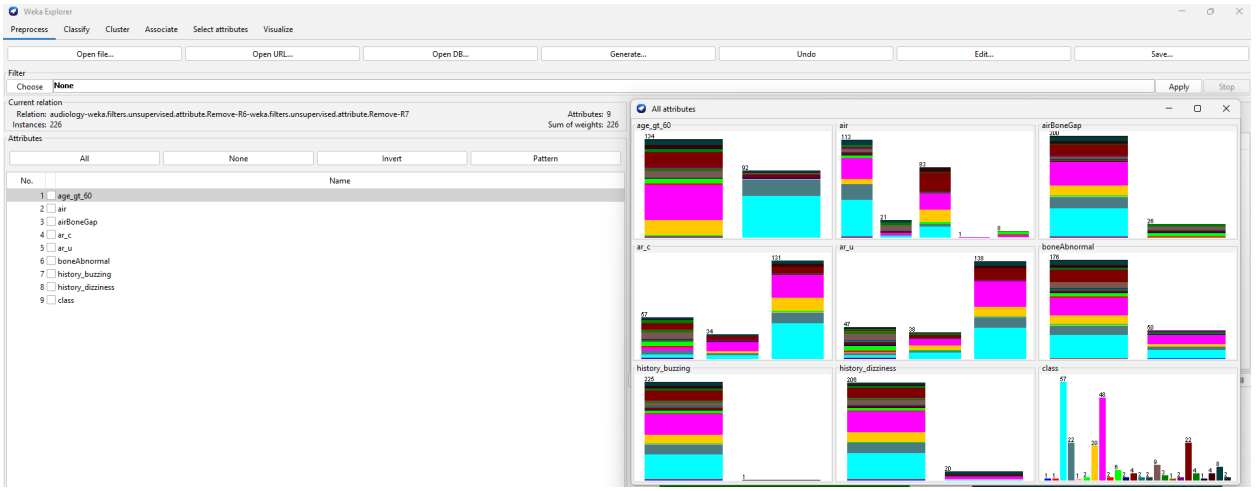


Figura 7: Conjunto de datos final con los atributos eliminados.

2.2. Imputación de valores perdidos

A continuación, se aplicó el filtro `ReplaceMissingValues`, que imputa automáticamente los valores faltantes restantes. Para atributos nominales, emplea la moda; para atributos numéricos, la media. Tras aplicar este filtro, se verificó que no quedaban valores faltantes, ya que la suma de valores por atributo coincide con el número total de instancias (Figura 8).

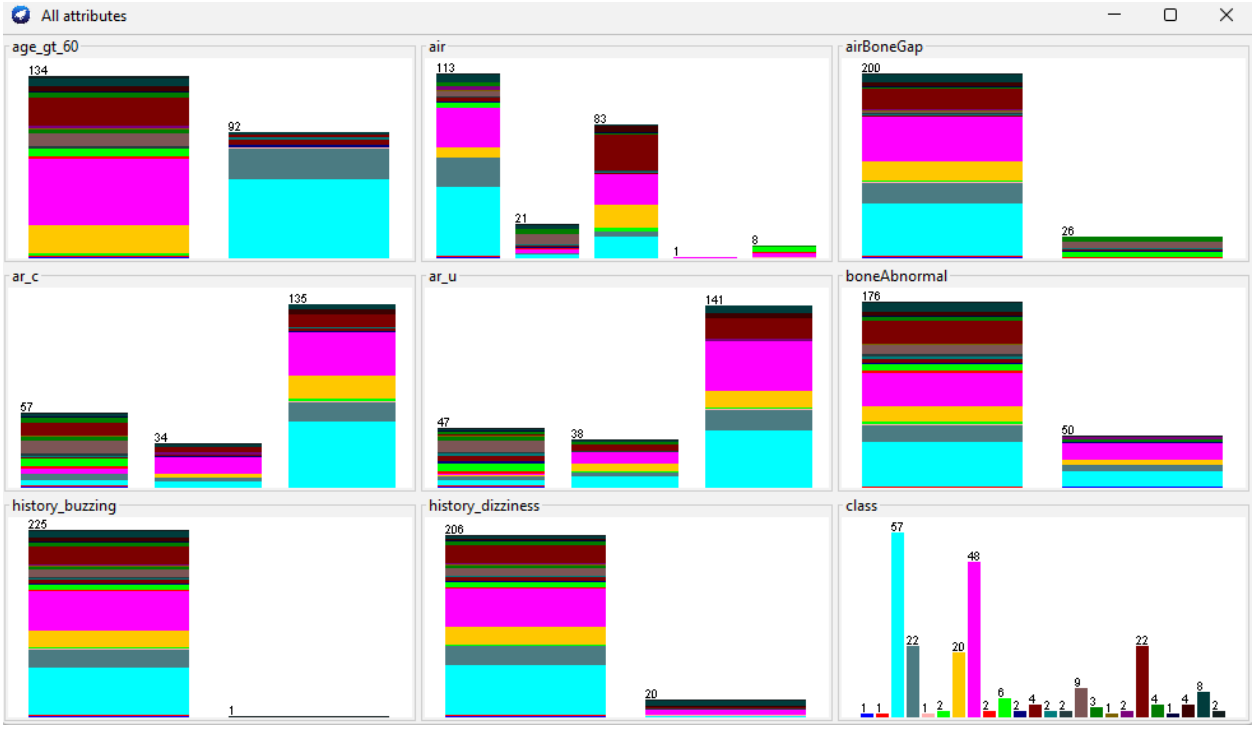


Figura 8: Conjunto de datos sin valores perdidos (todos los atributos suman 226, el número de instancias del dataset).

2.3. Conversión de atributos nominales a binarios

Finalmente, se aplicó el filtro `NominalToBinary` sobre los atributos nominales binarios y categórico como se muestra en la Figura 9.

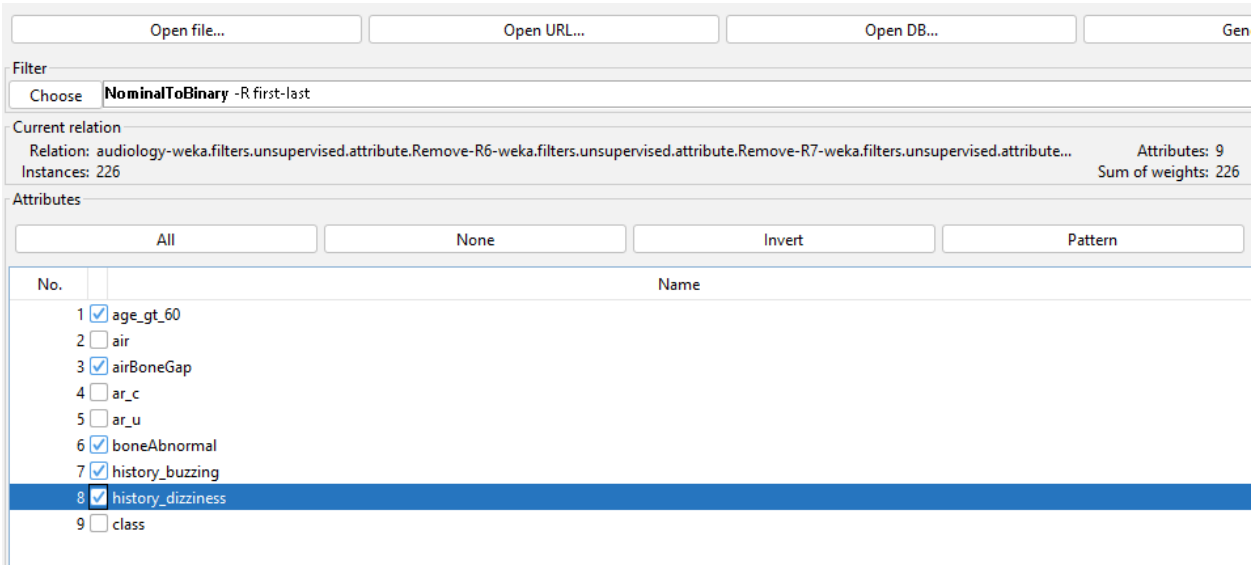


Figura 9: Selección de atributos con valores binarios del conjunto de datos.

Al aplicar el filtro, cada atributo nominal seleccionado fue transformado en una o más variables binarias, permitiendo representar la información de forma numérica (Figura 10).

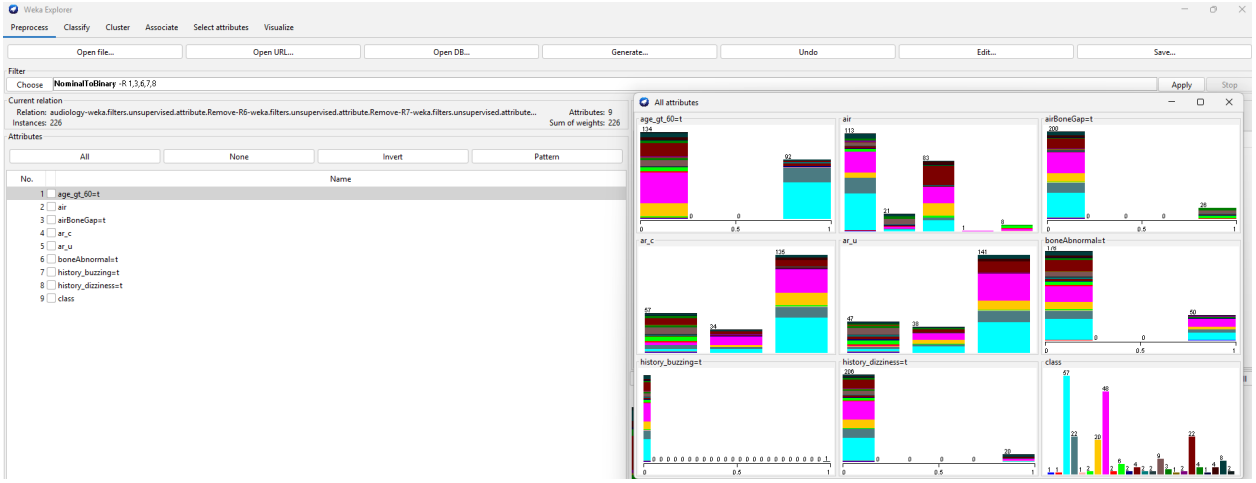


Figura 10: Conjunto de datos tras aplicar el filtro `NominalToBinary` a los atributos binarios (1, 3, 6, 7, 8).

3. Elección de Caso Práctico

El conjunto de datos seleccionado para el desarrollo del caso práctico es *Glass Identification*, disponible en el repositorio de aprendizaje automático de la Universidad de California en Irvine (UCI) [1]. Esta base de datos, proporcionada por el Servicio de Ciencias Forenses de los Estados Unidos, tiene como propósito la clasificación de distintos tipos de vidrio a partir de su composición química.

El objetivo principal consiste en determinar a qué tipo pertenece una muestra de vidrio, utilizando como base el porcentaje de diversos óxidos presentes en su estructura. Las clases recogidas en el conjunto de datos incluyen vidrios de ventanas (flotadas y no flotadas), vidrios de automóviles, utensilios, contenedores, entre otros.

El dataset está compuesto por 214 instancias y 10 atributos. Entre los atributos se encuentran las concentraciones de los siguientes compuestos: óxido de sodio (`Na`), óxido de magnesio (`Mg`), óxido de aluminio (`Al`), óxido de silicio (`Si`), óxido de potasio (`K`), óxido de calcio (`Ca`), óxido de bario (`Ba`) y óxido de hierro (`Fe`). Además, se incluye el índice de refracción (`RI`) y una variable identificadora que, por su naturaleza, no aporta valor predictivo.

La Figura 11 muestra una vista inicial del dataset cargado en el entorno de trabajo.

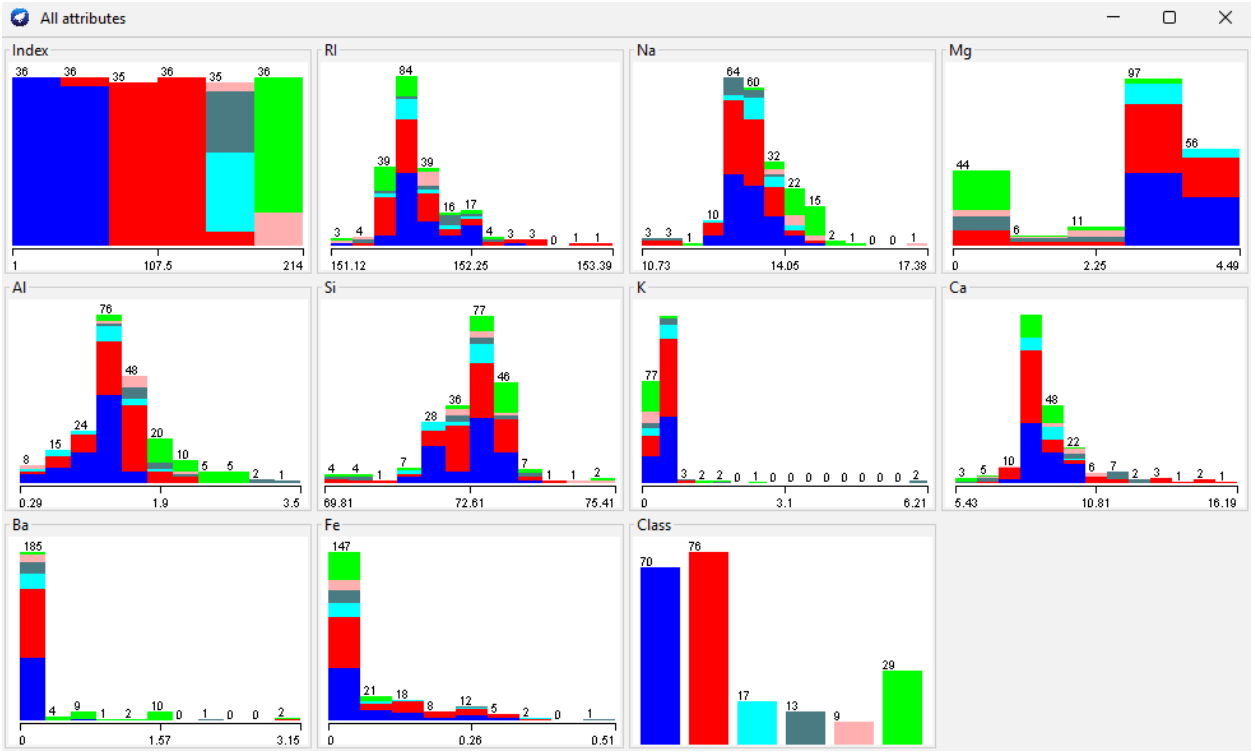


Figura 11: Vista inicial del dataset *Glass Identification* en Weka.

Dado que el identificador de la muestra no contiene información útil para el proceso de clasificación y puede introducir ruido en los modelos, se ha optado por eliminar esta variable durante la fase de preprocesamiento. El resto de atributos, todos de tipo numérico continuo, son aptos para ser utilizados en algoritmos estadísticos y de aprendizaje automático.

Una vez eliminado el identificador, el conjunto de datos queda como se observa en la Figura 12.

[1] https://archive.ics.uci.edu/dataset/42/glass+identification

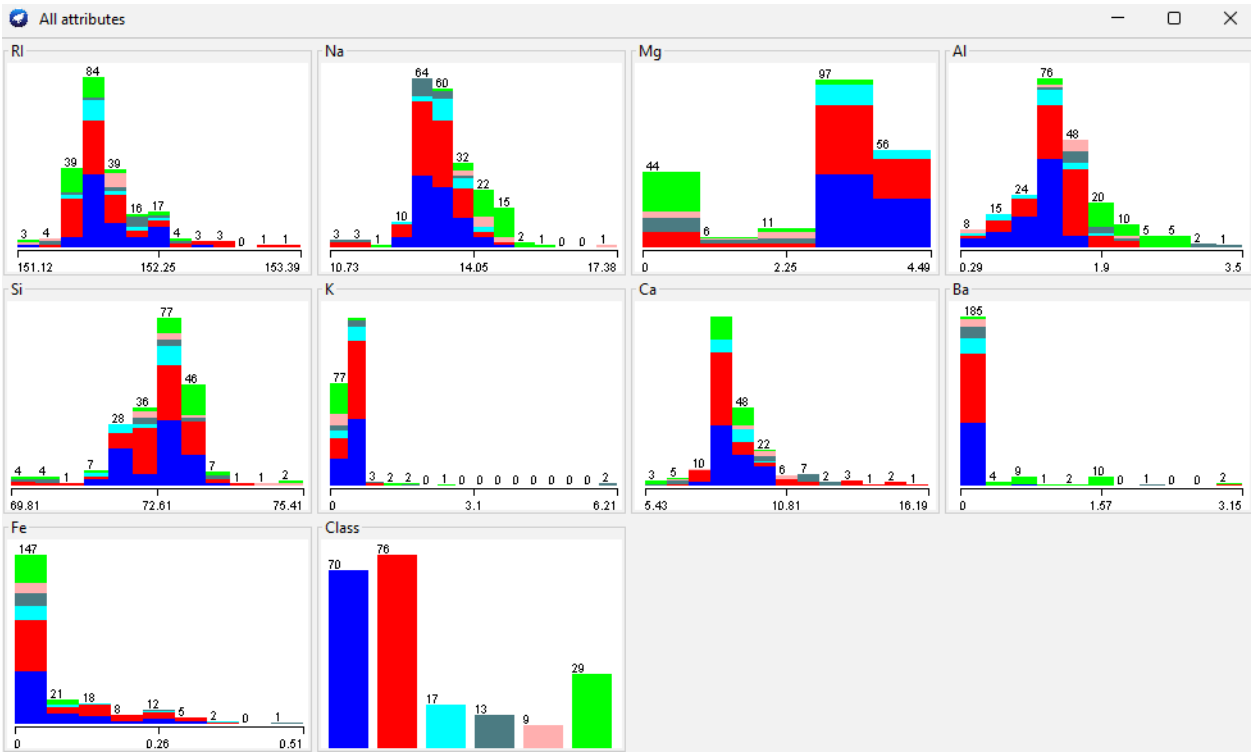


Figura 12: Vista del dataset tras eliminar la variable identificador.

4. Análisis Preliminar

Una vez cargado el dataset en el entorno `Preprocess` de Weka, se puede confirmar que se trata de un problema de clasificación multiclase, ya que la variable objetivo es de tipo categórico y toma uno de los siete posibles valores correspondientes a distintos tipos de vidrio. Todos los atributos predictivos son numéricos continuos, lo que permite aplicar algoritmos de análisis y clasificación.

La Tabla 1 resume las características principales de cada variable del conjunto de datos, incluyendo su nombre, una breve descripción, el tipo de dato y observaciones adicionales de interés.

Cuadro 1: Descripción de las variables del dataset *Glass Identification*.

Nombre	Descripción	Tipo	Observaciones
RI	Índice de refracción	Numérico	Medida óptica que varía ligeramente entre tipos de vidrio
Na	Porcentaje de óxido de sodio (%)	Numérico	Elemento habitual en la mayoría de los vidrios
Mg	Porcentaje de óxido de magnesio (%)	Numérico	Presente principalmente en vidrio de contenedores
Al	Porcentaje de óxido de aluminio (%)	Numérico	Mejora la resistencia del vidrio
Si	Porcentaje de óxido de silicio (%)	Numérico	Principal componente de la mayoría de los vidrios
K	Porcentaje de óxido de potasio (%)	Numérico	Presente en algunos tipos específicos de vidrio
Ca	Porcentaje de óxido de calcio (%)	Numérico	Común en vidrios para ventanas y botellas
Ba	Porcentaje de óxido de bario (%)	Numérico	Distintivo de algunos vidrios especiales
Fe	Porcentaje de óxido de hierro (%)	Numérico	Influye en el color del vidrio
Tipo	Clase del vidrio (1 a 7)	Nominal	Variable objetivo; representa el tipo de vidrio

El conjunto de datos se encuentra limpio, sin valores perdidos, y todos los atributos son relevantes desde el punto de vista químico para la identificación del tipo de vidrio. Esta estructura facilita el análisis exploratorio y la posterior aplicación de modelos de clasificación.

Desde el entorno `Preprocess` de Weka, al seleccionar una variable numérica, se muestra automáticamente un resumen estadístico que incluye, entre otros, el valor mínimo, máximo, la media y la desviación típica. Estos valores pueden consultarse en la parte derecha de la interfaz. A modo de ejemplo, en la Figura 13 se observa el resumen estadístico de la variable `Na`.

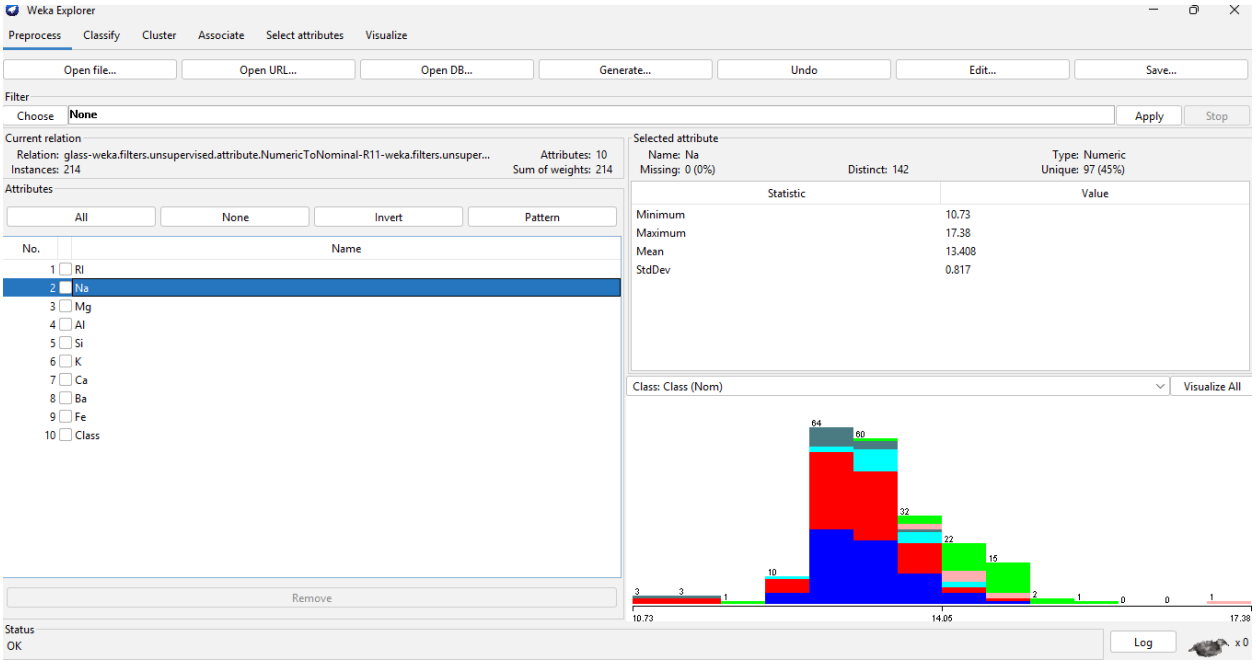


Figura 13: Resumen estadístico de la variable `Na` en Weka.

A continuación, se presenta una tabla con los valores estadísticos de interés para cada una de las variables cuantitativas:

Cuadro 2: Resumen estadístico de las variables cuantitativas del dataset.

Variable	Media	Mínimo	Máximo	Desviación típica
RI	151.837	151.115	153.393	0.304
Na	13.408	10.73	17.38	0.817
Mg	2.685	0	4.49	1.442
Al	1.445	0.29	3.5	0.499
Si	72.651	69.81	75.41	0.775
K	0.497	0	6.21	0.652
Ca	8.957	5.43	16.19	1.423
Ba	0.175	0	3.15	0.497
Fe	0.057	0	0.51	0.097

En el dataset *Glass Identification*, la única variable cualitativa es la variable de clase, que indica el tipo de vidrio al que pertenece cada muestra. Esta variable puede tomar uno de los siguientes siete valores:

- 1: Edificio - vidrio flotado.
- 2: Edificio - vidrio no flotado.
- 3: Vehículo - vidrio flotado.
- 4: Vehículo - vidrio no flotado (no está presente en los datos).
- 5: Contenedores.
- 6: Utensilios (vajilla).
- 7: Vidrio para luces delanteras (*headlamps*).

Desde el entorno `Preprocess` de Weka, al seleccionar la variable de clase (`class`) y observar el panel inferior, se puede ver un histograma que indica cuántas instancias hay de cada categoría, junto con su frecuencia absoluta y relativa. Se muestra en la Figura 14.

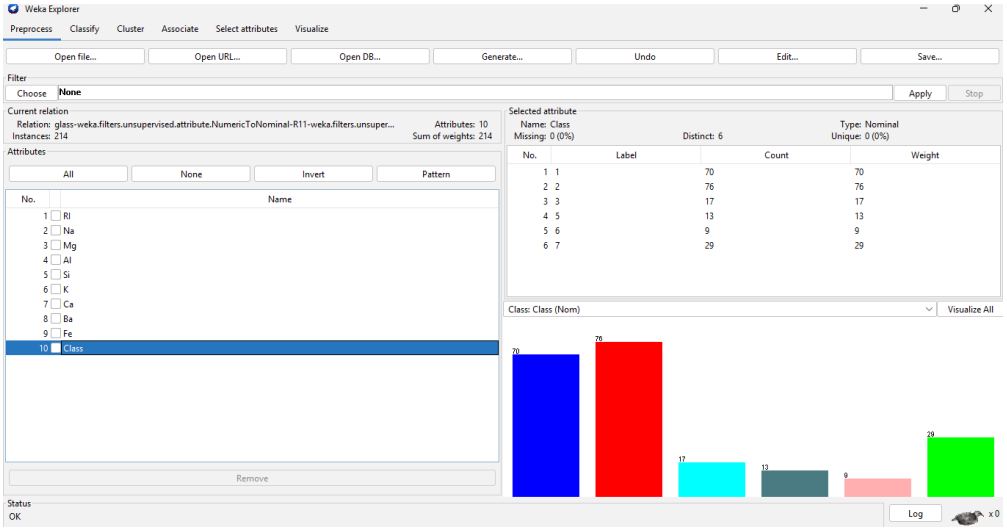


Figura 14: Distribución de frecuencias de la variable de clase en Weka.

La distribución de frecuencias de los distintos tipos de vidrio se resume en el Cuadro 3:

Cuadro 3: Frecuencia de cada tipo de vidrio en el dataset.

Clase	Descripción	Frecuencia (número de muestras)
1	Edificio - vidrio flotado	70
2	Edificio - vidrio no flotado	76
3	Vehículo - vidrio flotado	17
4	Vehículo - vidrio no flotado (no hay)	0
5	Contenedores	13
6	Vajilla	9
7	Faros	29

Durante la exploración inicial del conjunto de datos *Glass Identification*, se ha comprobado que no existen valores perdidos en ninguna de las variables. Esta información se obtiene a través del entorno `Preprocess` de Weka, que al cargar el conjunto de datos proporciona un resumen estadístico de cada atributo, incluyendo el número total de instancias, la cantidad de valores distintos y, en caso de haberlos, la cantidad de valores ausentes (*missing values*).

En este caso, todas las variables presentan un conteo completo de valores, sin datos faltantes, tal y como se muestra en la Figura 15. Por tanto, no ha sido necesario aplicar técnicas de imputación ni eliminar instancias.

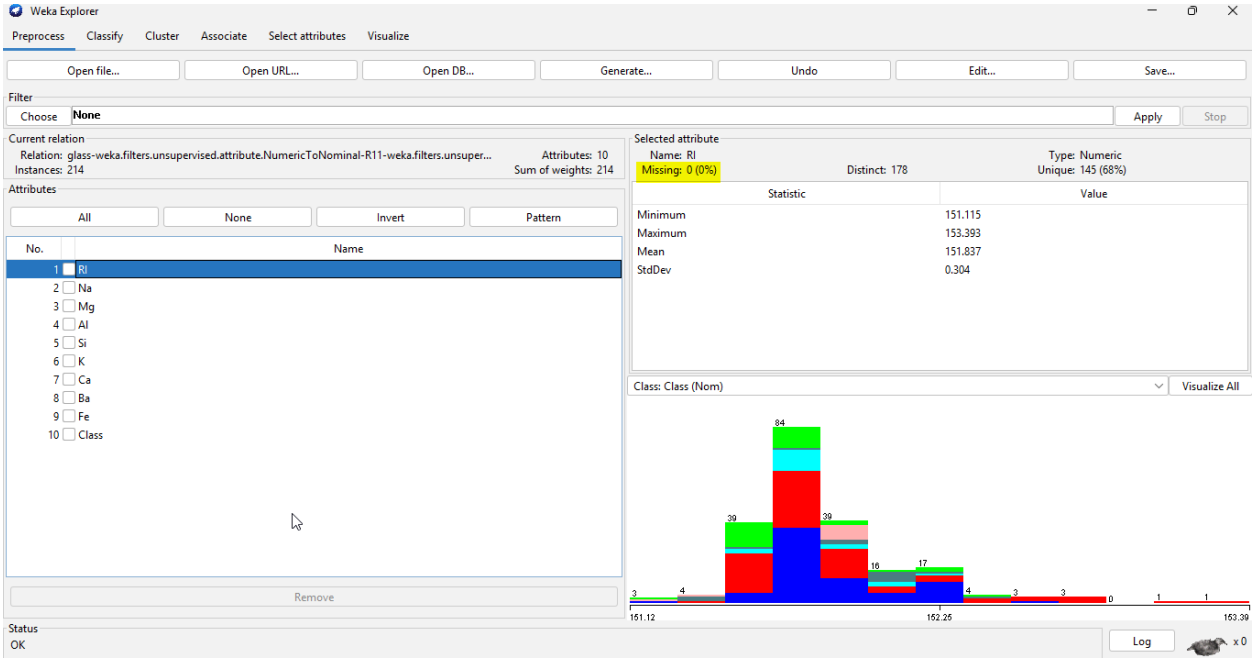


Figura 15: Porcentaje de valores perdidos en la variable RI (0 %). Todas las variables del conjunto de datos presentan un 0 % de valores ausentes.

Durante el análisis visual del conjunto de datos *Glass Identification* mediante la pestaña `Visualize` del entorno Weka, se han explorado posibles relaciones entre pares de atributos y su correspondencia con las clases objetivo. Para ello, se han utilizado herramientas como la opción `Jitter`, que permite dispersar ligeramente los puntos superpuestos para facilitar la observación, así como los controles `plot size` y `point size` para ajustar el tamaño de las gráficas y los puntos.

Como se observa en la Figura 16, algunos atributos muestran patrones de agrupación que podrían ser útiles para la tarea de clasificación.

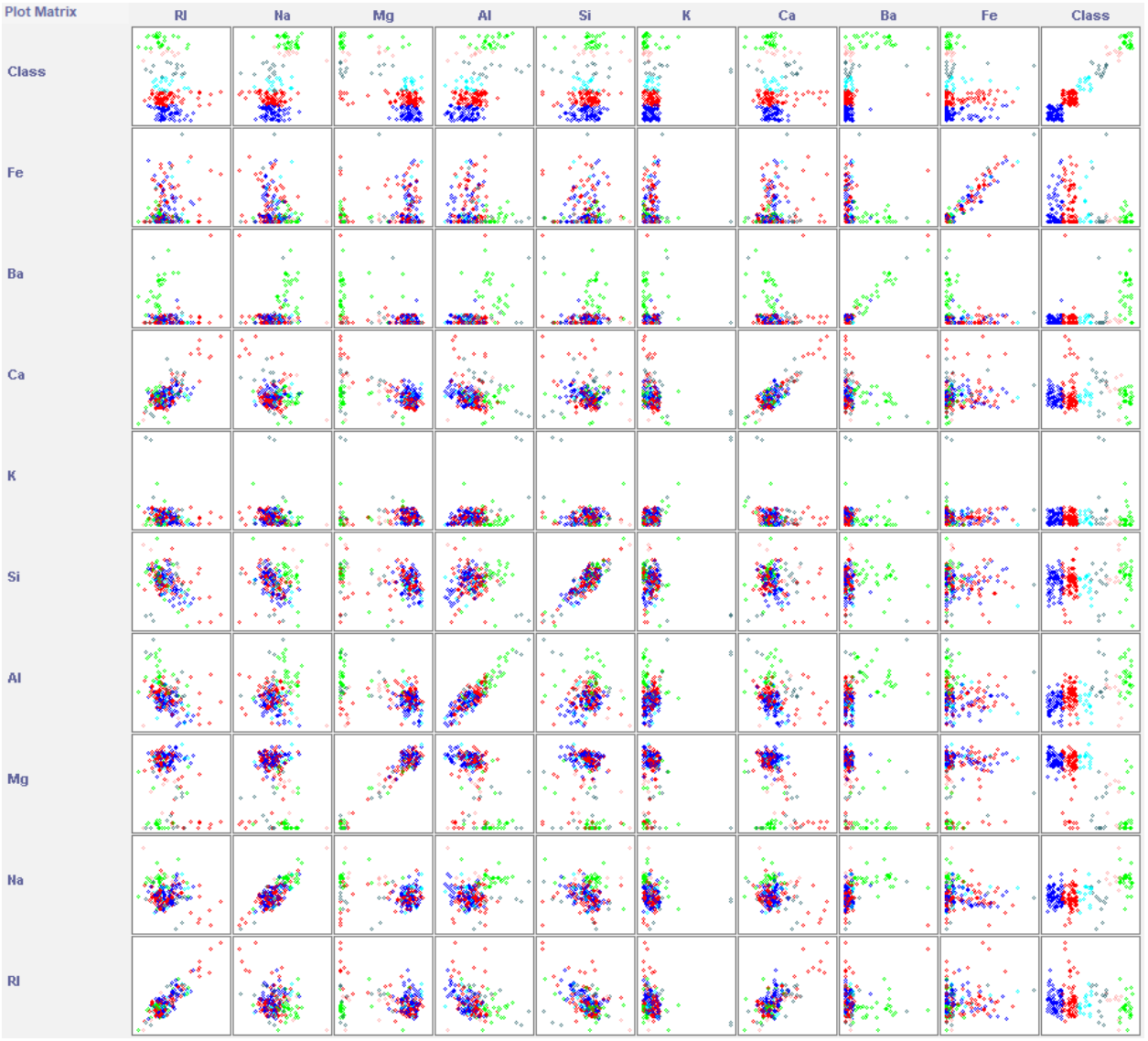


Figura 16: Gráficas de correlación

A través de este análisis, se han detectado ciertos patrones que pueden facilitar la separación de clases:

- **Tipo de vidrio con cada una de las variables**: Se diferencian bastante bien la división de clases con cada una de los atributos del dataset. Esto se puede ver en la Figura 17.

Figura 17: Correlación de la clase con cada una de las variables.

- **Óxido de bario (Ba)**: este elemento destaca especialmente para la clase 7, ya que las muestras de vajillas presentan niveles de Ba notablemente diferenciados respecto a las demás clases, facilitando su separación.

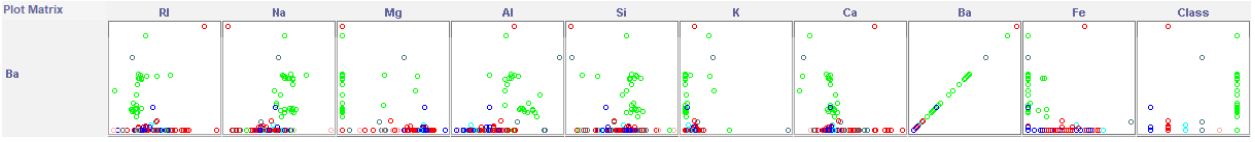


Figura 18: Correlación del atributo de Óxido de Bario con el resto de atributos.

A partir del análisis exploratorio del conjunto de datos *Glass Identification* en Weka, se identifican algunas tareas de preprocesamiento recomendables para mejorar la calidad y utilidad del dataset antes de aplicar modelos de clasificación. Esta información se obtiene principalmente del entorno `Preprocess` de Weka, donde se visualizan los atributos, sus tipos, rangos, y posibles anomalías.

Entre las acciones de preprocesamiento destacadas se encuentran:

- **Eliminación de la variable identificador**: Como se mencionó anteriormente, el atributo que actúa como identificador de la muestra no aporta información relevante para la clasificación y podría introducir ruido o sesgos en el modelo. En Weka, esta variable puede eliminarse fácilmente desde el panel de atributos en `Preprocess` seleccionándola y pulsando el botón `Remove`.

- **Normalización o estandarización de atributos**: Dado que las variables continuas presentan rangos y magnitudes muy diferentes (por ejemplo, el índice de refracción frente a las concentraciones de óxidos), es conveniente aplicar un escalado para que todas las variables contribuyan de forma equilibrada. En Weka, esto se puede realizar mediante el filtro `Normalize` o `Standardize`, disponibles en la sección de filtros del entorno `Preprocess`.

- **Revisión y tratamiento de valores atípicos**: Aunque no se detectaron valores perdidos, el análisis mediante el filtro `Interquartile Range` en Weka permitió identificar valores atípicos (outliers) en varias variables del conjunto de datos. Estos outliers pueden influir negativamente en el rendimiento de los modelos de clasificación al sesgar los parámetros o provocar un ajuste inadecuado. Para su detección, Weka muestra columnas adicionales que marcan si una instancia es un outlier o un valor extremo. El usuario puede visualizar estas instancias y, en caso necesario, aplicar filtros para eliminarlas o tratarlas, como el filtro `RemoveWithValues` para excluir las instancias con etiqueta de outlier.

  Tras eliminar los outliers, se obtiene un conjunto de datos más limpio y representativo, como se muestra en la Figura 21.

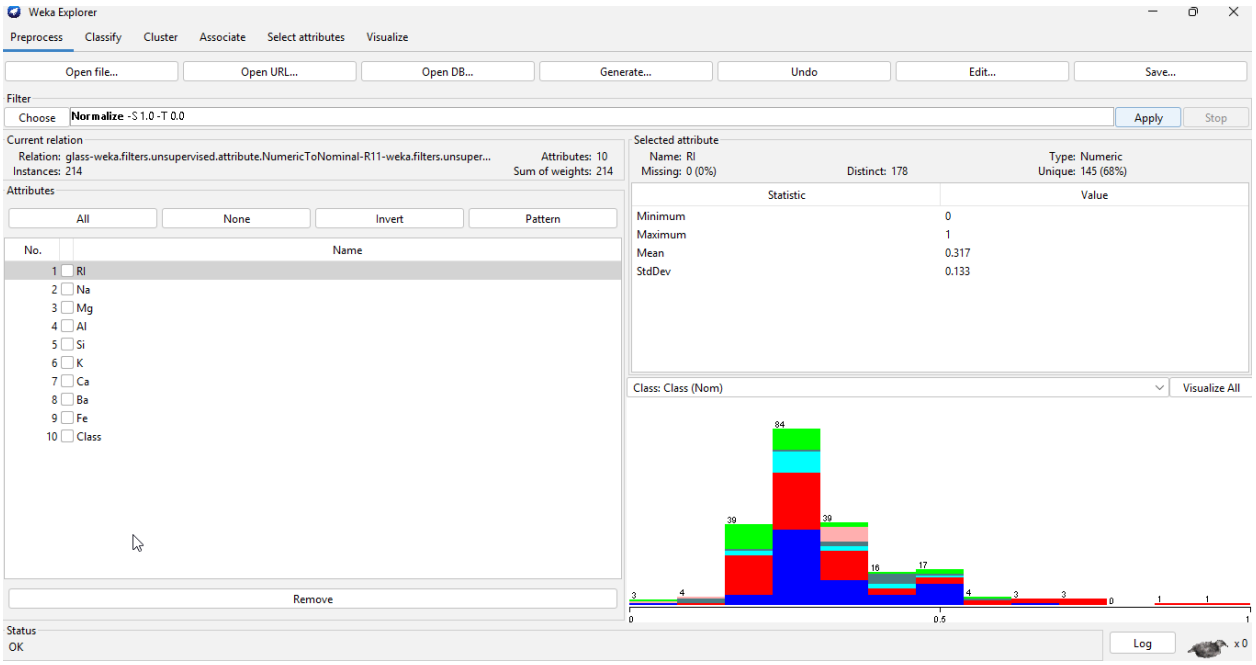


Figura 19: Aplicación del filtro de normalización en Weka. Se observa que los atributos han sido escalados a un rango [0, 1].

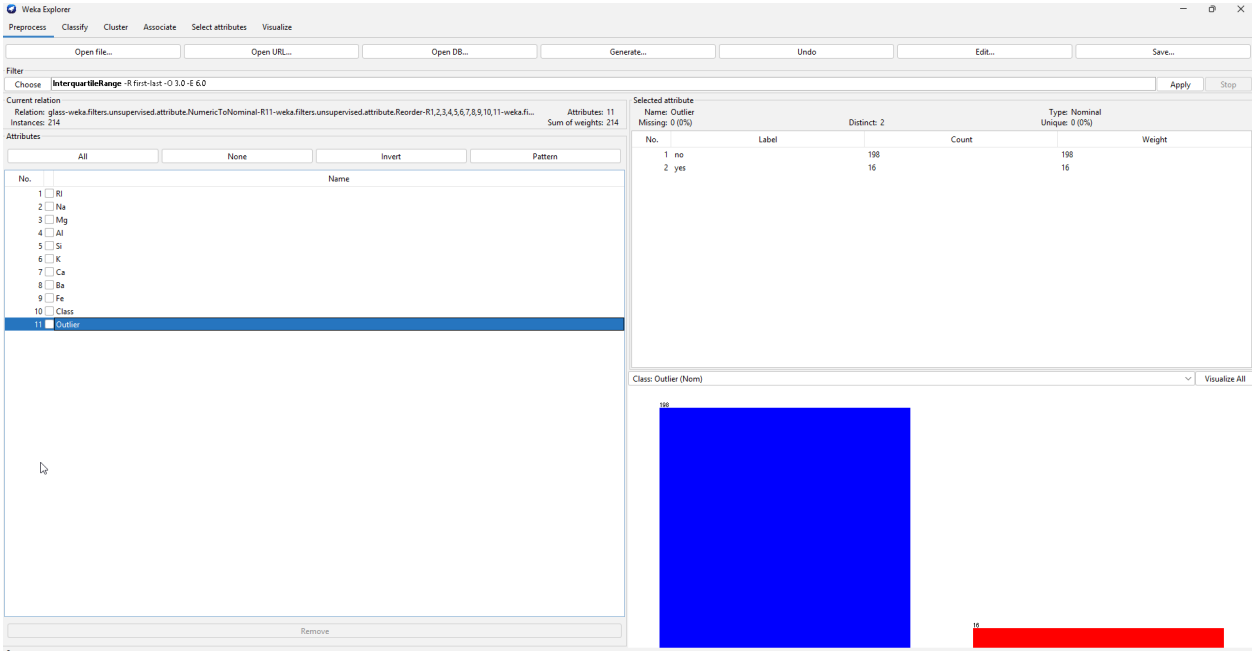


Figura 20: Outliers detectados mediante el filtro `Interquartile Range`.

5. Análisis de correlación de atributos

Para identificar qué atributos tienen mayor influencia en la variable de salida (`Class`), se ha utilizado el evaluador `InfoGainAttributeEval` junto con el método de búsqueda `Ranker`. Este método calcula la ganancia de información que aporta cada atributo respecto a la clase, permitiendo obtener un ranking de relevancia. Los atributos que mostraron mayor contribución fueron `Al` (0.637), `Mg` (0.565) y `K` (0.539), indicando que estas variables contienen más información útil para la clasificación del tipo de vidrio. En contraste,

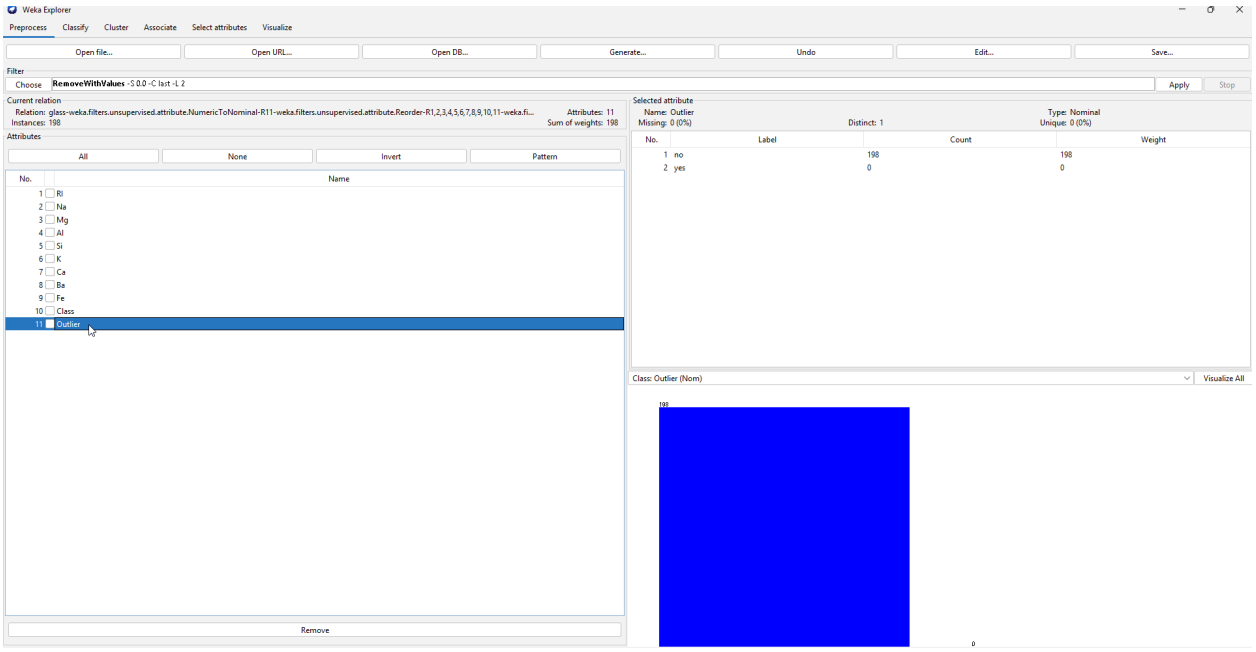


Figura 21: Dataset después de eliminar las instancias consideradas outliers.

atributos como `Fe` y `Si` mostraron una ganancia de información nula, lo que sugiere que podrían tener poca o ninguna utilidad para el modelo, al menos de forma individual. Este análisis permite reducir la dimensionalidad y centrarse en los atributos más informativos para mejorar la eficiencia de los algoritmos de clasificación.

Figura 22: Aplicación del algoritmo InfoGainAttributeEval + Ranker

El ranking obtenido mediante `InfoGainAttributeEval` refleja que los atributos con mayor ganancia de información respecto a la clase son `Al`, `Mg` y `K`. Este resultado tiene sentido desde el punto de vista químico, ya que la proporción de estos óxidos puede influir significativamente en las propiedades del vidrio (como

su resistencia, transparencia o punto de fusión), y por tanto en su uso final y clasificación. Por ejemplo, en el artículo de Rosales-Sosa et al. [2], se muestra que un mayor contenido en `Al` se asocia con vidrios más resistentes frente a grietas, mientras que otros estudios como [3] relacionan el contenido en `Mg` y `K` con mejoras en propiedades térmicas.

Por otro lado, atributos como `Fe` y `Si`, aunque relevantes en la composición general del vidrio, no presentan ganancia de información según el evaluador. Esto podría indicar que su valor no varía significativamente entre clases o que no tienen poder discriminativo individual. Es posible que su influencia sea redundante o dependa de interacciones con otras variables. En cualquier caso, el análisis respalda la utilidad del proceso de selección de atributos no solo desde un punto de vista computacional, sino también basado en el conocimiento del dominio químico.

6. Normalización y estandarización

Muchos algoritmos de aprendizaje automático requieren, o al menos se benefician significativamente, de trabajar con atributos normalizados o estandarizados. Este tipo de preprocesamiento facilita la convergencia de los modelos y evita que atributos con escalas muy diferentes dominen el proceso de aprendizaje.

En particular, clasificadores como `SimpleLogistic` y `J48` tienden a ser relativamente robustos frente a la escala de los datos, mientras que métodos basados en ensamblado, como `RandomForest`, pueden verse más afectados por variaciones en la magnitud de los atributos.

Aunque en la Sección 4 se realizó la normalización del conjunto de datos, en esta sección se analiza explícitamente el impacto que tiene el preprocesamiento mediante normalización y estandarización sobre el rendimiento de los modelos. Para ello, se han entrenado los clasificadores `SimpleLogistic`, `J48` y `RandomForest` con tres versiones del mismo conjunto de datos:

- Original, sin normalizar,
- Normalizado (valores escalados entre 0 y 1),
- Estandarizado (media cero y desviación típica uno).

A continuación se presentan los resultados obtenidos para cada caso, lo que permite observar cómo varía el rendimiento según el preprocesamiento aplicado.

Cuadro 4: Rendimiento de clasificadores con distintos preprocesamientos

Clasificador	**Tipo**	**Accur.**	**Kappa**	**MAE**	**RMSE**	**F1**	**MCC**	**ROC AUC**
SimpleLogistic	Sin normalizar	69.70	0.573	0.1417	0.2691	0.678	0.545	0.849
	Normalizado	69.70	0.573	0.1417	0.2691	0.678	0.545	0.849
	Estandarizado	69.70	0.573	0.1417	0.2691	0.678	0.545	0.849
J48	Sin normalizar	71.72	0.609	0.1040	0.2870	0.712	0.592	0.821
	Normalizado	71.72	0.609	0.1040	0.2870	0.712	0.592	0.821
	Estandarizado	71.72	0.609	0.1040	0.2870	0.712	0.592	0.821
RandomForest	Sin normalizar	81.31	0.741	0.1089	0.2177	0.811	0.736	0.947
	Normalizado	82.32	0.754	0.1087	0.2173	0.819	0.750	0.948
	Estandarizado	79.29	0.711	0.1104	0.2215	0.787	0.703	0.860

Como se observa en la Tabla 4, el rendimiento de los clasificadores `SimpleLogistic` y `J48` se mantiene prácticamente idéntico independientemente del tipo de preprocesamiento aplicado. Esto se debe a que ambos algoritmos son inherentemente poco sensibles a la escala de los atributos: `SimpleLogistic` utiliza una

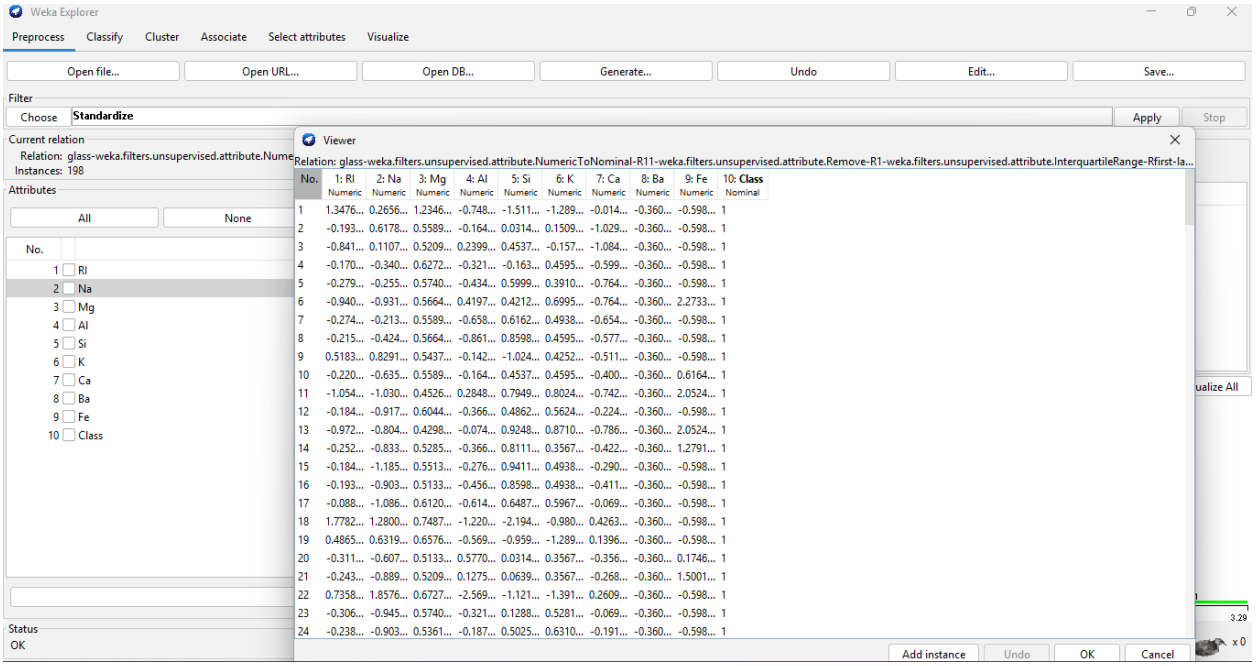


Figura 23: Ejemplo de dataset estandarizado

regresión logística con regularización que puede adaptarse bien a diferentes rangos, y `J48` es un árbol de decisión cuya división se basa en umbrales y no se ve afectado por la magnitud numérica en sí.

En cambio, el clasificador `RandomForest` sí muestra variaciones notables, mejorando su rendimiento con datos normalizados y experimentando una ligera caída cuando se usan datos estandarizados. Esto puede deberse a que el conjunto de árboles aleatorios se beneficia de la homogeneización de escalas para evaluar mejor las características, pero ciertos métodos de estandarización pueden alterar la distribución de los datos y afectar la construcción de los árboles.

7. Entrenamiento y prueba del modelo predictivo

Para evaluar la capacidad predictiva de los atributos seleccionados en el conjunto de datos, se ha usado el clasificador `SimpleLogistic`. Este algoritmo combina regresión logística multinomial con técnicas de regularización para evitar el sobreajuste, siendo adecuado para problemas multiclase y con atributos numéricos (ver Figura 24).

Se aplicó validación cruzada estratificada de 10 particiones (*10-fold cross-validation*) para obtener estimaciones robustas y reducir el sesgo en la evaluación del rendimiento del modelo.

El modelo fue entrenado con un total de 198 instancias y 9 atributos numéricos predictivos, junto con la variable objetivo de clasificación. El porcentaje de instancias correctamente clasificadas (CCR) alcanzó un 69.7 %, lo que indica una capacidad razonable de predicción en un problema multiclase, teniendo en cuenta la complejidad y el tamaño limitado del conjunto de datos.

Las métricas más relevantes por clase, basadas en la tasa de verdaderos positivos (TP Rate) y el área bajo la curva ROC (AUC), se muestran a continuación:

- **Clase 1**: TP Rate = 0.714, AUC = 0.820
- **Clase 2**: TP Rate = 0.691, AUC = 0.819

- **Clase 3**: TP Rate = 0.059, AUC = 0.773
- **Clase 5**: TP Rate = 0.857, AUC = 0.999
- **Clase 6**: TP Rate = 1.000, AUC = 0.999
- **Clase 7**: TP Rate = 0.929, AUC = 0.963

Como se observa, el modelo presenta un rendimiento notablemente superior en las clases 5, 6 y 7, reflejado en valores altos tanto en TP Rate como en AUC. En cambio, la clase 3 muestra un desempeño muy inferior, probablemente debido a su baja representación en el conjunto de entrenamiento (solo 17 instancias) o a una mayor confusión con clases similares, lo que resalta la influencia directa de la calidad y cantidad de datos por clase en el desempeño del modelo.

Para ampliar la comparación, se analizaron también los resultados de otros dos clasificadores populares: `J48` (árbol de decisión) y `RandomForest` (ensamblado de árboles). Las imágenes correspondientes a sus resultados se encuentran en las Figuras 25 y 26 respectivamente.

En la Tabla 5 se resumen las métricas principales de desempeño de los tres clasificadores evaluados bajo el mismo esquema de validación cruzada:

Cuadro 5: Comparación de clasificadores mediante validación cruzada (10-fold)

Clasificador	**CCR (%)**	**TP Rate media**	**AUC media**
SimpleLogistic	69.7	0.697	0.849
J48	65.2	0.652	0.812
RandomForest	73.8	0.738	0.876

Mientras que `SimpleLogistic` ofrece un modelo interpretable y con rendimiento competitivo en las clases más representadas, `RandomForest` supera ligeramente su desempeño global gracias a la combinación de múltiples árboles que mejora la generalización. Por su parte, `J48` obtiene resultados inferiores, posiblemente por su tendencia a sobreajustar con conjuntos pequeños y su menor capacidad para capturar relaciones complejas sin ajustes adicionales.

La baja precisión en la clase 3 indica la necesidad de aplicar estrategias adicionales, como:

- Recolección o generación de datos adicionales para las clases minoritarias.
- Técnicas de balanceo de datos, por ejemplo sobremuestreo (SMOTE) o submuestreo.
- Experimentación con modelos o métodos específicos para clases desequilibradas.

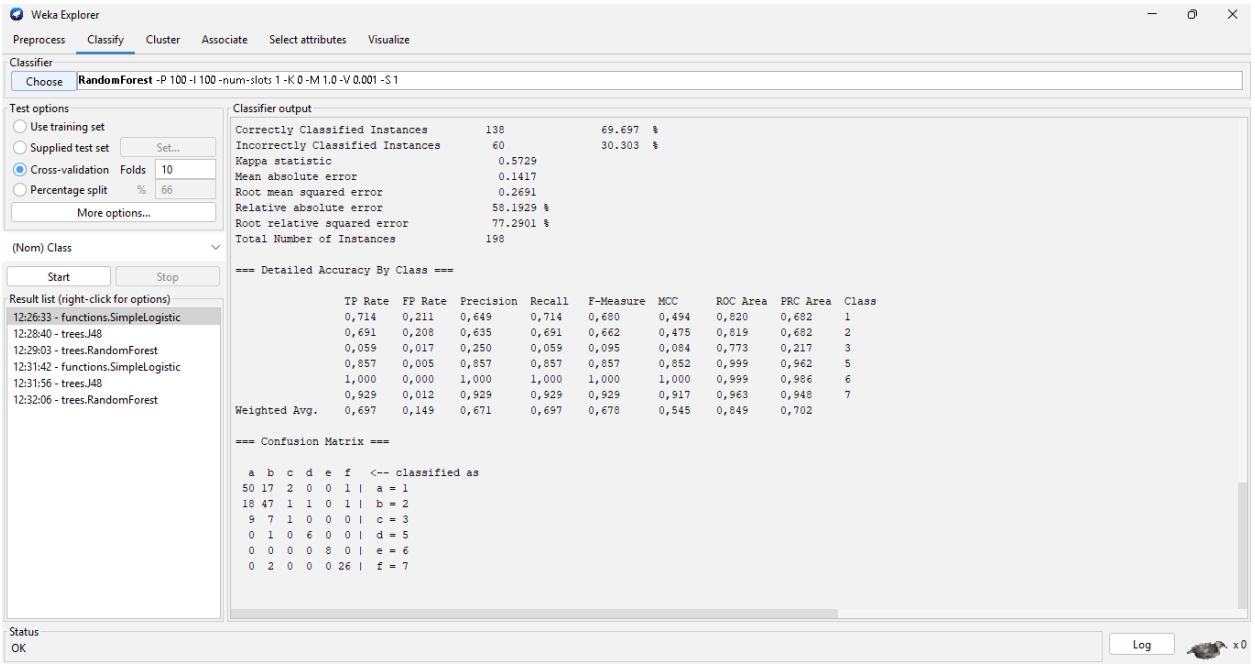


Figura 24: Resultados del algoritmo `SimpleLogistic` para clasificación del conjunto de datos.

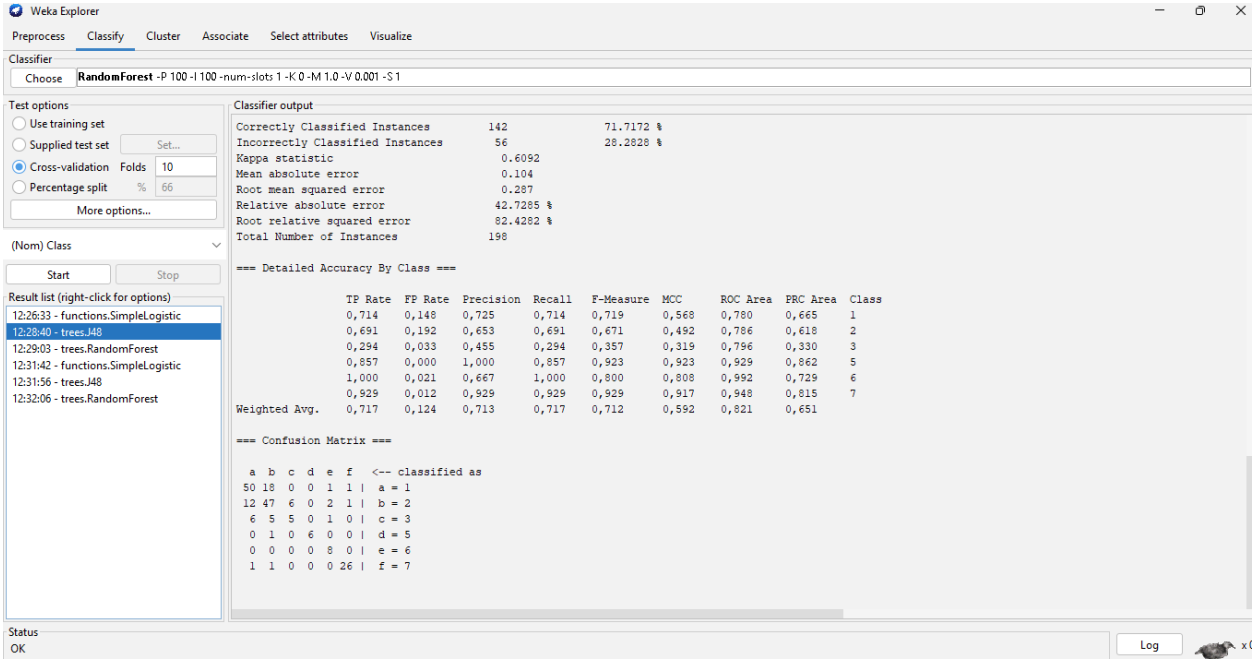


Figura 25: Resultados del algoritmo `J48` para clasificación del conjunto de datos.

Finalmente, se debe considerar que durante la validación cruzada Weka genera particiones diferentes para cada clasificador, lo que puede introducir pequeñas variaciones en los resultados. Para un análisis más riguroso, se recomienda fijar las particiones mediante semillas aleatorias y normalizar los datos usando únicamente las estadísticas del conjunto de entrenamiento para evitar filtración de información.

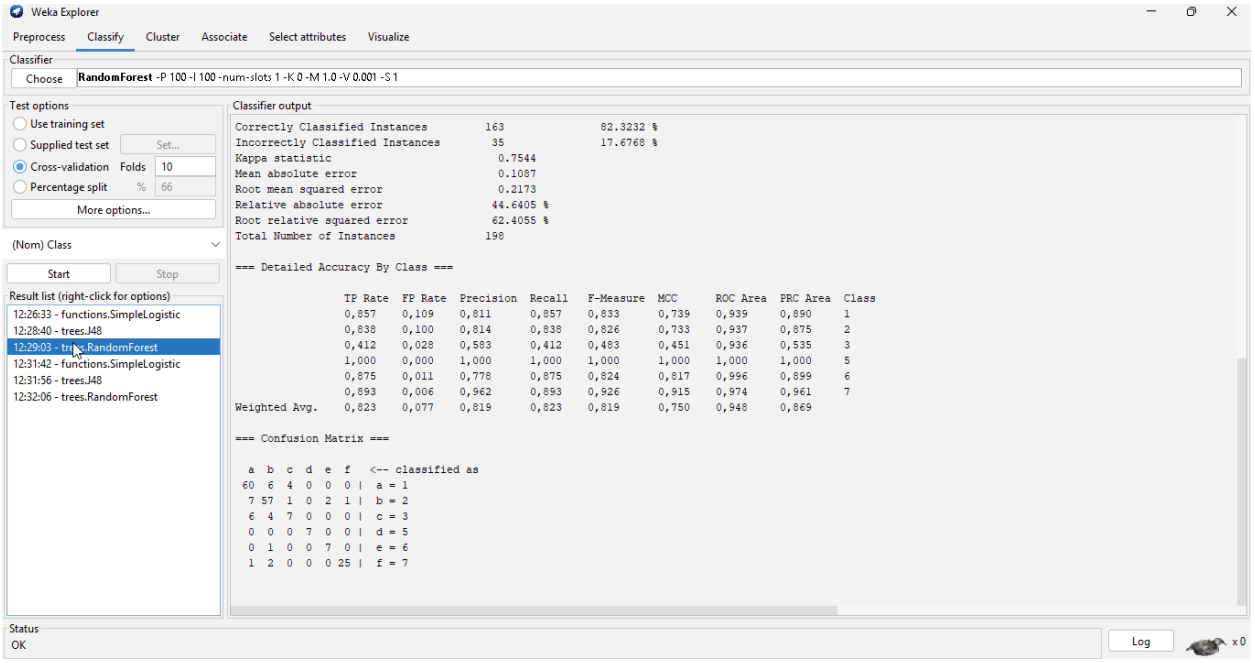


Figura 26: Resultados del algoritmo `RandomForest` para clasificación del conjunto de datos.

Referencias

[1] E. Frank, M. A. Hall, and I. H. Witten, "The weka workbench," 2016. Online Appendix for "Data Mining: Practical Machine Learning Tools and Techniques".

[2] G. A. Rosales-Sosa, A. Masuno, Y. Higo, and H. Inoue, "Crack-resistant al o –sio glasses," *Scientific Reports*, vol. 6, no. 1, p. 23620, 2016.

[3] T. Nishida, M. Yamada, H. Ide, and Y. Takashima, "Correlation between the structure and glass transition temperature of potassium, magnesium and barium tellurite glasses," *Journal of Materials Science*, vol. 25, pp. 3546–3550, 1990.