



Instituto Superior Universitario Tecnológico del Azuay
Tecnología Superior en Big Data

**Guía Practica N°3 - Kaggle, OpenRefine y Weka como
herramientas para el análisis de datos**

Integrantes:

Eduardo Mendieta
Freddy Montalván

Materia:

Introducción a Big Data

Docente:

MSc. Ing. Carmen Tacuri Vintimilla

Ciclo:

Primer Ciclo

Fecha:

22 de julio de 2024

Periodo Académico:

Abril 2024 - Agosto 2024

Índice

1.	Introducción	2
2.	Objetivos	2
2.1.	Objetivo general	2
2.2.	Objetivos específicos	2
3.	Paso a paso	3
3.1.	Kaggle Datasets	3
3.2.	OpenRefine	3
3.3.	Weka	14
4.	Errores y correcciones realizadas a la data	26
5.	Resultados del análisis de la data	27
6.	Conclusiones y recomendaciones	28
6.1.	Conclusiones	28
6.2.	Recomendaciones	28
7.	Bibliografía	29

Guía Practica N°3: Kaggle, OpenRefine y Weka como herramientas para el análisis de datos

1. Introducción

La capacidad de analizar datos de manera efectiva se ha convertido en una habilidad fundamental en el panorama actual, donde la información abunda y su interpretación correcta puede proporcionar insights valiosos para la toma de decisiones informadas. Dentro de este contexto, herramientas como Kaggle, OpenRefine y Weka se destacan como recursos indispensables para aprender y aplicar técnicas de análisis de datos de manera práctica y accesible.

Kaggle destaca como una plataforma central para aprender análisis de datos, ofreciendo conjuntos de datos reales y desafíos competitivos que permiten aplicar técnicas avanzadas de aprendizaje automático. OpenRefine simplifica la limpieza y preparación de datos con una interfaz intuitiva, ideal para principiantes en análisis de datos. Weka ofrece una amplia gama de algoritmos de aprendizaje automático y una interfaz gráfica intuitiva, facilitando la experimentación con técnicas como clasificación, regresión y agrupamiento.

Utilizando un dataset real como el del naufragio del Titanic, estas herramientas se vuelven especialmente relevantes al aplicarlas directamente en un contexto práctico. Este dataset histórico no solo proporciona datos reales para el análisis, sino que también motiva el aprendizaje al enfrentar problemas reales de ciencia de datos. Así, los estudiantes podemos desarrollar habilidades fundamentales mientras exploramos las capacidades de Kaggle, OpenRefine y Weka en la manipulación y análisis de datos significativos.

2. Objetivos

2.1. Objetivo general

Abordar los ejercicios prácticos delineados en la guía para fomentar el pensamiento analítico a través de la aplicación de herramientas especializadas como Kaggle, OpenRefine y Weka.

2.2. Objetivos específicos

- Instalar las herramientas OpenRefine y Weka siguiendo los pasos de instalación proporcionados en los enlaces de descarga recomendados.
- Descargar el archivo titanic.csv desde la plataforma Kaggle para su análisis posterior.
- Utilizar OpenRefine para preparar y limpiar los datos del archivo titanic.csv, siguiendo los procedimientos descritos en la guía práctica.
- Realizar un análisis gráfico de los datos previamente limpiados con OpenRefine utilizando Weka, siguiendo los procedimientos descritos en la guía práctica.
- Presentar los resultados del análisis previo de los datos.

3. Paso a paso

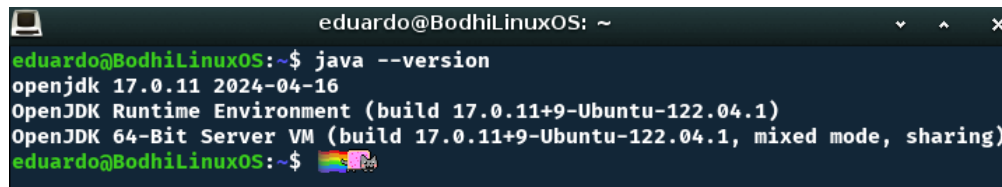
3.1. Kaggle Datasets

Kaggle Datasets es una plataforma en línea que forma parte de la comunidad de Kaggle, ofreciendo una amplia variedad de conjuntos de datos para proyectos de ciencia de datos, análisis y aprendizaje automático. Los usuarios pueden buscar, explorar y descargar datasets en diversos formatos y temáticas, desde datos de salud hasta imágenes y textos. La plataforma facilita la colaboración a través de Notebooks de Kaggle, donde los usuarios pueden analizar datos, construir modelos y compartir sus resultados. Además, los datasets a menudo están asociados con competiciones y proyectos, y la comunidad activa permite discutir y aprender de otros científicos de datos.

3.2. OpenRefine

OpenRefine es una poderosa herramienta gratuita y de código abierto para trabajar con datos desordenados: limpiarlos, transformarlos de un formato a otro, y extenderlos con servicios web y datos externos.

- En primer lugar, es necesario contar con una versión de Java igual o superior a *Java 8*. Para ello, descargamos la versión adecuada para Ubuntu desde su repositorio oficial utilizando los siguientes comandos: *sudo apt update*, *sudo apt install openjdk-17-jdk* y *java -version*.

A terminal window titled 'eduardo@BodhiLinuxOS: ~' with standard window controls. The terminal shows the command 'java --version' being executed. The output is: 'openjdk 17.0.11 2024-04-16', 'OpenJDK Runtime Environment (build 17.0.11+9-Ubuntu-122.04.1)', and 'OpenJDK 64-Bit Server VM (build 17.0.11+9-Ubuntu-122.04.1, mixed mode, sharing)'. The prompt returns to 'eduardo@BodhiLinuxOS:~\$' with a rainbow flag emoji.

```
eduardo@BodhiLinuxOS:~$ java --version
openjdk 17.0.11 2024-04-16
OpenJDK Runtime Environment (build 17.0.11+9-Ubuntu-122.04.1)
OpenJDK 64-Bit Server VM (build 17.0.11+9-Ubuntu-122.04.1, mixed mode, sharing)
eduardo@BodhiLinuxOS:~$ 🌈🏳️‍🌈
```

Figura 1: Verificación de la instalación de Java

- Descargamos OpenRefine desde su página oficial, seleccionando la versión para el sistema operativo Ubuntu. Esto descargará un archivo *.tar.gz*, el cual descomprimos en el directorio */home*. A continuación, ingresamos en la carpeta descomprimida y otorgamos permisos de ejecución al archivo *refine* utilizando el comando *chmod +x refine*. Posteriormente, ejecutamos el archivo con *./refine*. Este proceso iniciará un servidor que se abrirá automáticamente en un navegador, permitiéndonos comenzar a utilizar la herramienta.

Download OpenRefine

OpenRefine is free software released under the [BSD 3-clause license](#), brought to you by our contributors.



Figura 2: Intalación y ejecución de OpenRefine

- Procedemos a descargar el archivo *titanic.csv* desde la plataforma *Kaggle* o desde la dirección de [Google Drive](#) propuesta en la guía.

- Podemos cambiar el idioma desde la opción de *Language Settings* del menu lateral izquierdo de la herramienta a español.
- Creamos un nuevo proyecto y cargamos el archivo *titanic.csv* previamente descargado, dando clic en *siguiente* y en *Crear Proyecto* en la esquina superior derecha.

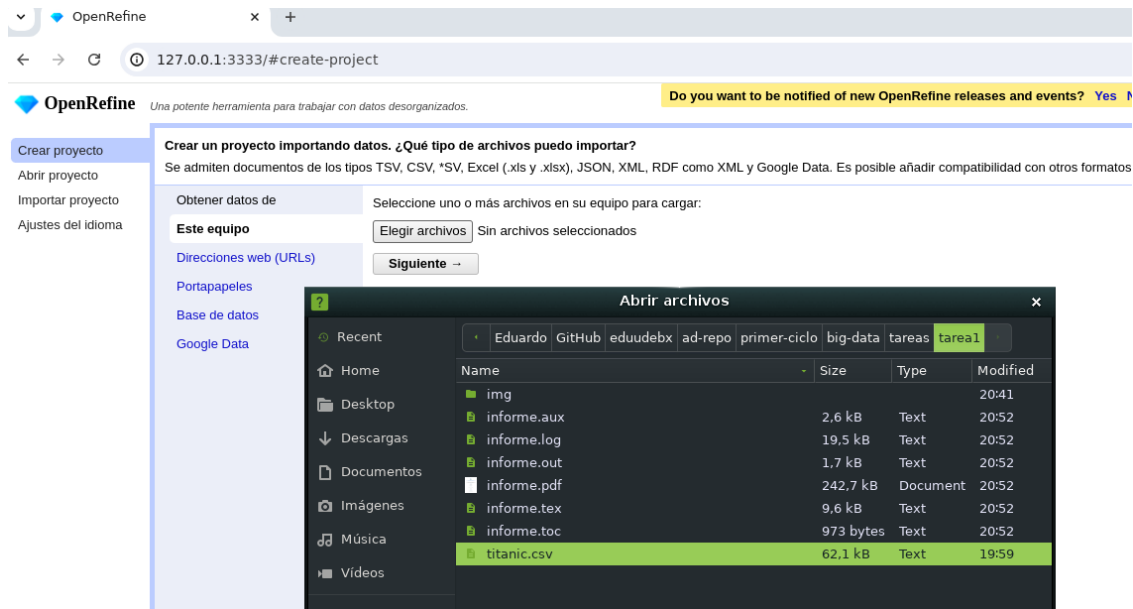


Figura 3: Creación de un nuevo proyecto

- Podemos renombrar las columnas, en el caso de la columna *Column*, la podemos renombrar a *Vacia*.

891 filas

Mostrar como: **filas** registros Mostrar: 5 10 25 50 100 500 1000 filas

▼ Todo	▼ Passengerid	▼ Survived	▼ Pclass	▼ Name	▼ Sex	▼ Age	▼ SibSp	▼ Parch	▼ Ticket	▼ Fare	▼ Cabin	▼ Embarked	▼ Column
1	1	0	3	Braund, Mr. Owen Harris	male	22	1	0	A/5 21171	7.25		S	
2	2	1	1	Cummings, Mrs. John Bradley (Florence Briggs Thayer)	female	38	1	0	PC 17599	71.2833	C85	C	
3	3	1	3	Heikkinen, Miss. Laina	female	26	0	0	STON/O2. 3101282	7.925		S	
4	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35	1	0	113803	53.1	C123	S	
5	5	0	3	Allen, Mr. William Henry	male	35	0	0	373450				
6	6	0	3	Moran, Mr. James	male		0	0	330677				
7	7	0	1	McCarthy, Mr. Timothy J	male	54	0	0	17463				
8	8	0	3	Palsson, Master. Gosta Leonard	male	2	3	1	349909				
9	9	1	3	Johsson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)	female	27	0	2	347742				
10	10	1	2	Nasser, Mrs. Nicholas (Adele Achem)	female	14	1	0	237736				
11	11	1	3	Sandstrom, Miss. Marguerite Rut	female	4	1	1	PP 9549				
12	12	1	1	Bonnell, Miss. Elizabeth	female	58	0	0	113783				
13	13	0	3	Saunderscock, Mr. William Henry	male	20	0	0	A/5. 2151				
14	14	0	3	Andersson, Mr. Anders Johan	male	39	1	5	347082				
15	15	0	3	Vestrom, Miss. Hulda Amanda Adolfina	female	14	0	0	350406				
16	16	1	2	Hewlett, Mrs. (Mary D Kingcome)	female	55	0	0	248706				
17	17	0	3	Rice, Master. Eugene	male	2	4	1	382652				
18	18	1	2	Williams, Mr. Charles Eugene	male		0	0	244373				
19	19	0	3	Vander Planke, Mrs. Julius (Emelia Maria Vandemoortele)	female	31	1	0	345763	18		S	

Extensi

« primera « anterior 1 sigue

Facetas
Filtro de texto
Editar celdas
Editar columna
Transponer
Ordenar...
Ver
Cotejar

Dividir en varias columnas...
Unir las columnas...
Agregar columna basada en esta columna...
Agregar columna accediendo a URLs...
Añadir columnas de valores cotejados...
Renombrar esta columna...
Quitar esta columna
Mover columna al principio
Mover columna al final
Mover columna a la izquierda
Mover columna a la derecha

Figura 4: Renombrando columnas

- En el siguiente paso buscamos los valores no vacíos por columna en todo el proyecto dando clic en la columna *Todo* → *Facetas* → *Valores no vacíos por columna*.



Figura 5: Buscando valores no vacíos por columna

- Se puede observar que el número total de registros por columna es diferente, por lo cuál podemos inferir que existen valores que no están definidos dentro de estas, por lo que será necesario tratarlos.

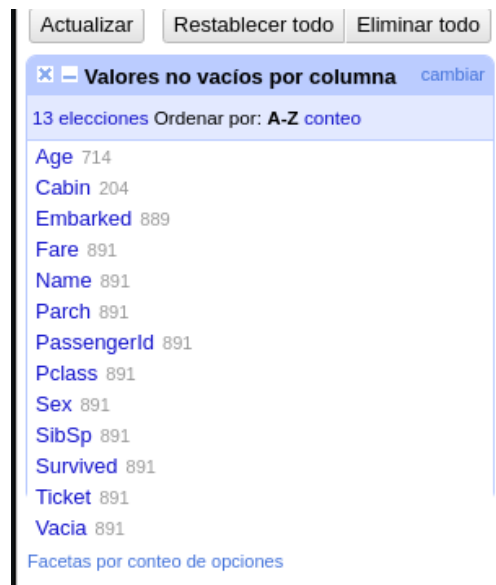


Figura 6: Valores no vacíos por columna

- Procedemos a retirar la columna *Vacía* puesto que ya no es de utilidad.

Extensión						
« primera			« anterior		1	siguiente »
ibSp	Parch	Ticket	Fare	Cabin	Embarked	Vacia
0	PC	17604	82.1708	C		Facetas
0		113789	52	S		Filtro de texto
0		2677	7.2292	C		Editar celdas
0	A/5	2152	8.05	S		Editar columna
0		345764				Dividir en varias columnas...
0		2651				Unir las columnas...
0		7546				Transponer
0		11668				Ordenar...
0		349253				Ver
2	SC/Paris	2123				Cotejar
0		330958				
0	S.C./A.4.	23567				
0		370371				
0		14311				
0		2662				
0		349237				
1		3101295				
0	A/4	39886				

Figura 7: Eliminando columna *Vacia*

- La información cargada inicialmente en el proyecto esta en formato de cadena por lo que seria conveniente transformar las columnas que contienen valores numericos, siendo estas *Survived*, *PClass*, *Sibsp* y *Parch* como números enteros y *Age* y *Fare* en formato decimal.

▼ Todo	▼ PassengerId	▼ Survived	▼ Pclass	▼ Name	▼ Sex	▼ Age	▼ SibSp	▼ Parch	▼ Ticket	▼ Fare	▼ Cabin	▼ Embarked
1.	1.	0	3	Braund, Mr. Owen Harris	male	22	1	0	A/5 21171			S
2.	2.	1	1	Cummings, Mrs. John Bradley (Florence Briggs Thayer)	female	38	1	0	PC 17599			C
3.	3.	1	3	Heikinen, Miss. Laina	female	26	0	0	STON/O2. 3101282			S
4.	4.	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35	1	0	113803			C
5.	5.	0	3	Allen, Mr. William Henry	male	35	0	0				S
6.	6.	0	3	Moran, Mr. James	male		0	0				C
7.	7.	0	1	McCarthy, Mr. Timothy J	male	54	0	0				S
8.	8.	0	3	Patson, Master. Gosta Leonard	male	2	3	1				C
9.	9.	1	3	Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)	female	27	0	2				S
10.	10.	1	2	Nasser, Mrs. Nicholas (Adele Achem)	female	14	1	0	A Tipo oración			S
11.	11.	1	3	Sandstrom, Miss. Marguerite Rut	female	4	1	1	A mayúsculas			S
12.	12.	1	1	Bonnell, Miss. Elizabeth	female	58	0	0	A minúsculas			S
13.	13.	0	3	Saunderscock, Mr. William Henry	male	20	0	0	A número			S
14.	14.	0	3	Andersson, Mr. Anders Johan	male	39	1	5	A fecha			S
15.	15.	0	3	Vestrom, Miss. Hulda Amanda Adolfina	female	14	0	0	A texto			S
16.	16.	1	2	Hewlett, Mrs. (Mary D Kingcome)	female	55	0	0	Establecer celdas en nulo			S
17.	17.	0	3	Rice, Master. Eugene	male	2	4	1	A la cadena vacia			S
18.	18.	1	2	Williams, Mr. Charles Eugene	male		0	0				S
19.	19.	0	3	Vander Planke, Mrs. Julius (Emelia Maria Vandemoortele)	female	31	1	0				S
20.	20.	1	3	Masselmani, Mrs. Fatima	female		0	0	2649	7.225		C

Figura 8: Transformación de columnas a tipo numerico

- Podemos observar los valores vacios por columna en todo el proyecto dando clic en la columna *Todo* → *Facetas* → *Valores vacíos por columna*. Podemos observar que las columnas *Age*, *Cabin* y *Embarked* contienen valores nulos, por lo que debemos tratar estas columnas.



Figura 9: Columnas con valores nulos

- Entonces, para completar los valores que faltan en el atributo Embarked, solo lo llenamos con *S* sabiendo que los pasajeros realmente embarcaron en Southampton. Para lo cual filtramos por el valor faltante, al dar click en el panel de resumen en el atributo *Embarked*, luego, en la columna *Embarked* → *Editar celdas* → *Reemplazar* → *Reemplazar por* actualizando de esta manera las casillas vacías por la letra *S*.



Figura 10: Reemplazando valores nulos

- Para el atributo *Cabin*, al no saber la cabina real de cada pasajero, podemos agregar una columna adicional basada en esta llamada *HaveCabin* con un *1* pasajero si la cabina existe y *0* si no existe. Esto se puede lograr con *Cabin* → *Editar columna* → *Agregar columna basada en esta columna*. Agregamos el nombre *HaveCabin* y la siguiente línea de código `if(value.toString() == "", 0, 1)`. por último click en *Aceptar*.

Añadir columna basada en otra Cabin

Nombre nuevo de la columna

En error ☒ cambiar a en blanco ☐ guardar error ☐ copiar valor de la columna original

Expresión No hay error de sintaxis.

Lenguaje

Previsualización [Historial](#) [Con estrella](#) [Ayuda](#)

row	value	if(value.toString() == "", 0, ...
1.	null	0
2.	C85	1
3.	null	0
4.	C123	1
5.	null	0
6.	null	0

Figura 11: Creando columnas a partir de otras columnas

- Para el atributo de *Age*, calculamos la media de todos los valores de la columna. Para esto creamos una nueva columna llamada *Record*, basada en la primera columna *PassengerId* con el proposito de agrupar los datos y poder sacar el promedio. Agregamos el código `if(value.toString() != '', 1, 0)`.

Añadir columna basada en otra PassengerId

Nombre nuevo de la columna

En error ☒ cambiar a en blanco ☐ guardar error ☐ copiar valor de la columna original

Expresión No hay error de sintaxis.

Lenguaje

Previsualización [Historial](#) [Con estrella](#) [Ayuda](#)

row	value	if(value.toString() != "", 1, ...
1.	1	1
2.	2	1
3.	3	1
4.	4	1
5.	5	1
6.	6	1

Figura 12: Creando columnas a partir de otras columnas

- Reubicamos la columna *Record* en la primera posición haciendo click en la columna *Todo* → *Editar columnas* → *Reordenar/quitar columnas*, arrastramos la columna *Record* a la primera posición y click en *aceptar*.

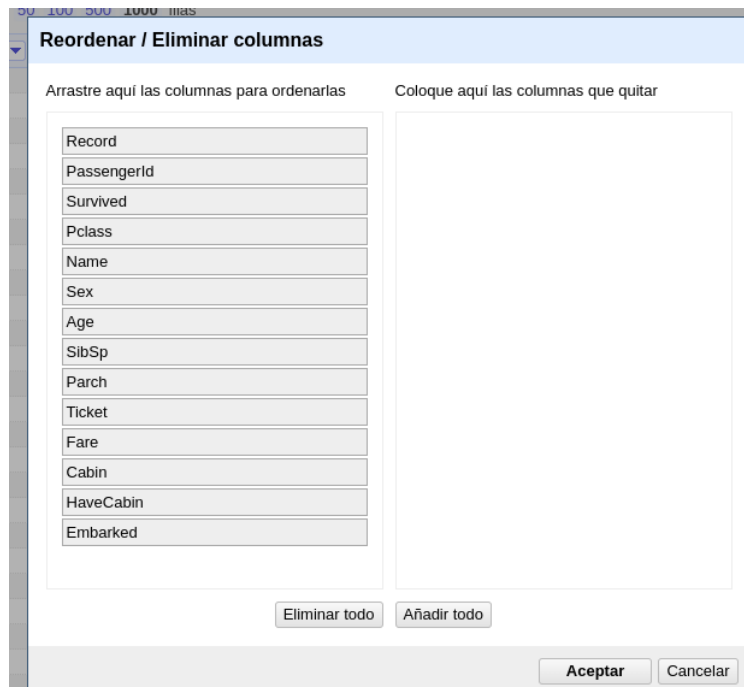


Figura 13: Reordenando columnas

- Agrupamos los datos dando click en *Todo* → *Editar celdas* → *Vaciar hacia abajo*.
- Creamos una nueva columna a partir de *Age* con el nombre *AgeProm*, le agregamos el siguiente código `round(sum(row.record.cells['Age'].value) / length(row.record.cells['Age'].value))` y click en *Aceptar*.

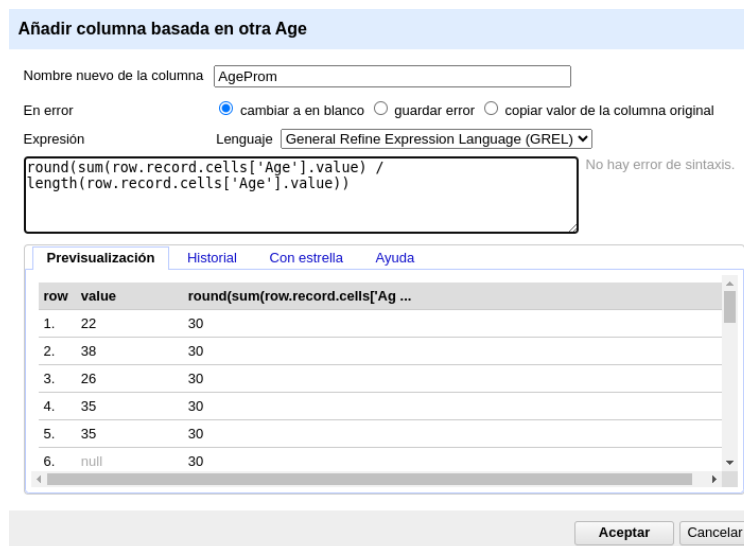


Figura 14: Creando columnas a partir de otras columnas

- Para reemplazar los valores faltantes de la columna *Age* por la media damos click en *Age* → *Editar celdas* → *Transformar*, agregamos el código `if(value.toString() == "`

30, value) y click en *Aceptar*.

Transformación personalizada en Age

Expresión Lenguaje General Refine Expression Language (GREL) ▼

`if(value.toString() == '', 30, value)` No hay error de sintaxis.

Previsualización Historial Con estrella Ayuda

row	value	if(value.toString() == '', 30, ...
1.	22	22
2.	38	38
3.	26	26
4.	35	35
5.	35	35
6.	null	30

En error ☒ mantener original ☐ Re-transformar hasta veces hasta que no haya cambios
☐ cambiar a en blanco
☐ guardar error

Aceptar **Cancelar**

Figura 15: Agregando la media de las edades en la columna Age

- Una vez preprocesada la data, ya no es necesaria la columna *Record*, *Cabin*, *AgeProm*, y podemos eliminarlas, para evitar campos vacíos.
- Buscamos si existen valores duplicados, en el caso de la columna *PassengerId*, ordenamos la información dando click en *PassengerId* → *Ordenar* → seleccionamos *números* y *menores primero*, click en *Aceptar*.

Ordenar por PassengerId

Ordenar valores como Posición de blancos y errores

☐ texto ☐ Distingue mayúsculas y minúsculas
☒ números
☐ fechas
☐ booleano

Valores validos
 Errores
 Blancos

Arrastre para ordenar

☒ menores primero ☐ mayores primero

Aceptar **Cancelar**

Figura 16: Ordenando la columna PassengerId

- Para visualizar si existen duplicados damos click en *PassengerId* → *Facetas* → *Facetas personalizadas* → *Faceta por duplicados*.

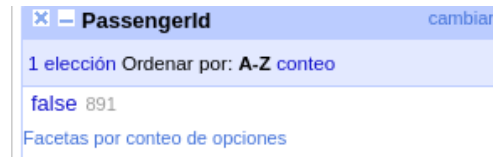


Figura 17: Buscando duplicados en PassengerId

- En el caso de existir duplicados, damos clic en *PassengerId* → *Editar celdas* → *vaciar hacia abajo*, luego en *PassengerId* → *Facetas* → *Facetas personalizadas* → *Faceta por blanco (nulo o cuerda vacía)* y en el caso de existir valores vacíos seleccionar y eliminarlos.
- Para el caso de la edad creamos una columna llamada *TypeAge* en base a la columna *Age*, donde se considera si ≥ 18 el valor será 0, y si es < 18 el valor será 1.

Añadir columna basada en otra Age

Nombre nuevo de la columna

TypeAge

En error

☒ cambiar a en blanco
 ☐ guardar error
 ☐ copiar valor de la columna original

Expresión

Lenguaje

General Refine Expression Language (GREL)

if(value >= 18, 1, 0)

No hay error de sintaxis.

Previsualización

Historial

Con estrella

Ayuda

row	value	if(value >= 18, 1, 0)
1.	22	1
2.	38	1
3.	26	1
4.	35	1
5.	35	1
6.	30	1

Aceptar

Cancelar

Figura 18: Creando columnas a partir de otras columnas

- Regresamos todos los datos a tipo cadena.

891 filas

Mostrar como: **filas** registros Mostrar: 5 10 25 50 100 500 1000 filas Ordenar ▼

« primera < anterior 1 siguiente > última »

▼ Todo	▼ PassengerId	▼ Survived	▼ Pclass	▼ Name	▼ Sex	▼ Age	▼ TypeAge	▼ SibSp	▼ Parch	▼ Ticket	▼ Fare	▼ HaveCabin	▼ Embarked
1	1	0	3	Braund, Mr. Owen Harris	male	22	1	1	0	A/5 21171	7.25	0	
2	2	1	1	Cummings, Mrs. John Bradley (Florence Briggs Thayer)	female	38	1	1	0	PC 17599	71.2833	1	
3	3	1	3	Helikinen, Miss. Laina	female	26	1	0	0	STON/O2. 3101282			
4	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35	1	1	0	113803			
5	5	0	3	Allen, Mr. William Henry	male	35	1	0	0	373450			
6	6	0	3	Moran, Mr. James	male	30	1	0	0	330877			
7	7	0	1	McCarthy, Mr. Timothy J	male	54	1	0	0	17463			
8	8	0	3	Palsson, Master. Gosta Leonard	male	2	0	3	1	349909			
9	9	1	3	Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)	female	27	1	0	2	347742			
10	10	1	2	Nasser, Mrs. Nicholas (Adele Achem)	female	14	0	1	0	237736			
11	11	1	3	Sandstrom, Miss. Marguerite Rut	female	4	0	1	1	PP 9549			
12	12	1	1	Bonnell, Miss. Elizabeth	female	58	1	0	0	113783			
13	13	0	3	Saunderscock, Mr. William Henry	male	20	1	0	0	A/5 2151			
14	14	0	3	Andersson, Mr. Anders Johan	male	39	1	1	5	347082	31.275	0	
15	15	0	3	Vestrom, Miss. Hilda Amanda Adolfin	female	14	0	0	0	350406	7.8542	0	
16	16	1	2	Hewlett, Mrs. (Mary D Kingcome)	female	55	1	0	0	248706	16	0	
17	17	0	3	Rice, Master. Eugene	male	2	0	4	1	382652	29.125	0	
18	18	1	2	Williams, Mr. Charles Eugene	male	30	1	0	0	244373	13	0	
19	19	0	3	Vander Planke, Mrs. Julius (Emelia Maria Vandemoortele)	female	31	1	1	0	345763	18	0	

Transformar... Editar celdas

Transformaciones comunes

- Quitar espacios al inicio y final
- Contrair espacios consecutivos
- Des-escapar entidades HTML
- Reemplazar comillas inteligentes por ASCII
- A Tipo oración
- A mayúsculas
- A minúsculas
- A número
- A fecha
- A texto
- Establecer celdas en nulo
- A la cadena vacía

Figura 19: Transformado a tipo de dato cadena

- Finalmente, exportamos el archivo dando click *Exportar* → *Valor delimitado por comas* en la esquina superior derecha. Se descarga un archivo *titanic-csv.csv*. Tambien podemos exportar los datos en formato excel.

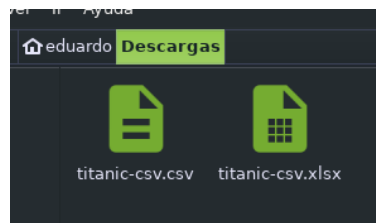


Figura 20: Datos pre-procesados

3.3. Weka

Weka es una plataforma de software libre que permite a los usuarios aplicar diversos algoritmos de aprendizaje automático y técnicas de minería de datos a sus conjuntos de datos. El nombre "Weka" proviene del nombre de un ave nativa de Nueva Zelanda y fue desarrollado por la Universidad Waikato en Nueva Zelanda.

- Descargamos Weka para Ubuntu. Realizamos el mismo procedimiento de instalación de *OpenRefine*, este caso, los permisos de ejecución se deben otorgar al archivo *weka.sh* y para ejecutarlo con *./weka.sh*.

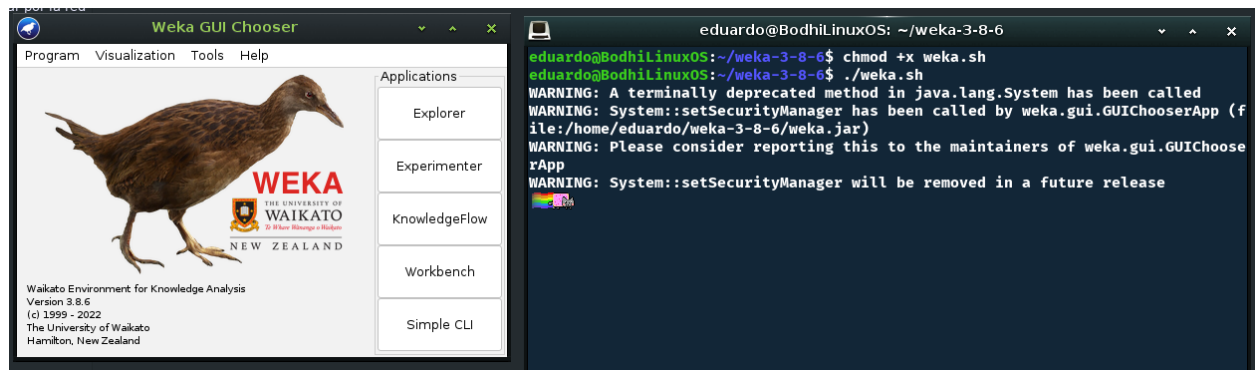


Figura 21: Ejecutando Weka

- Weka ofrece las opciones posibles de interfaces de trabajo.
 - **Explorer:** es la opción que permite ejecutar los algoritmos de análisis y comparar resultados sobre un único conjunto de datos.
 - **Experimenter:** es la opción que permite definir experimentos complejos y almacenar resultados.
 - **Knowledge Flow:** es la opción que permite llevar a cabo las mismas operaciones que Experimenter pero representado como un grafo dirigido.
 - **Simple CLI:** es *Command Line Interfaz* es una ventana de comandos java para ejecutar las clases WEKA.
- Antes de importar el archivo *.csv* preprocesado con *OpenRefine* abrimos el archivo para editar caracteres que pueden ocasionar errores, en este caso reemplazamos todas las por un texto vacío utilizando *Leafpad*, similar al *Notepad++*.

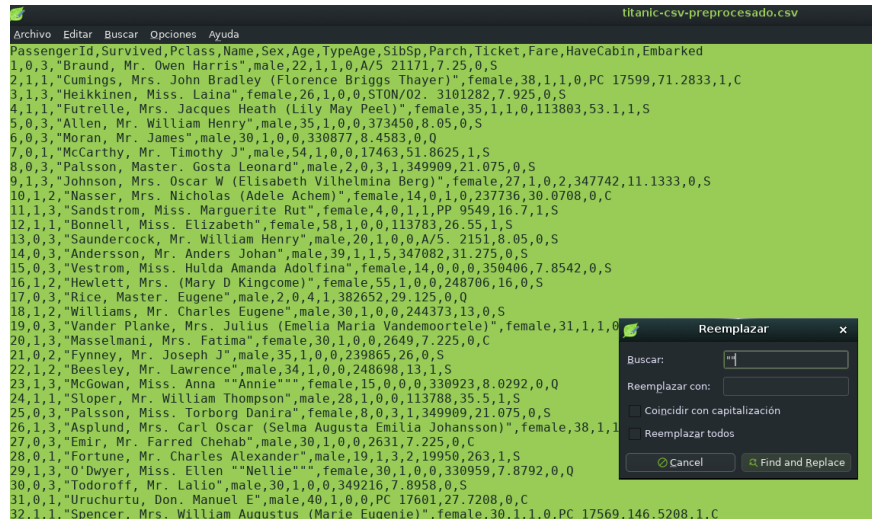


Figura 22: Eliminando caracteres que ocasionan errores

- Seleccionamos la opción Explorer e importamos el archivo .csv preprocesado.

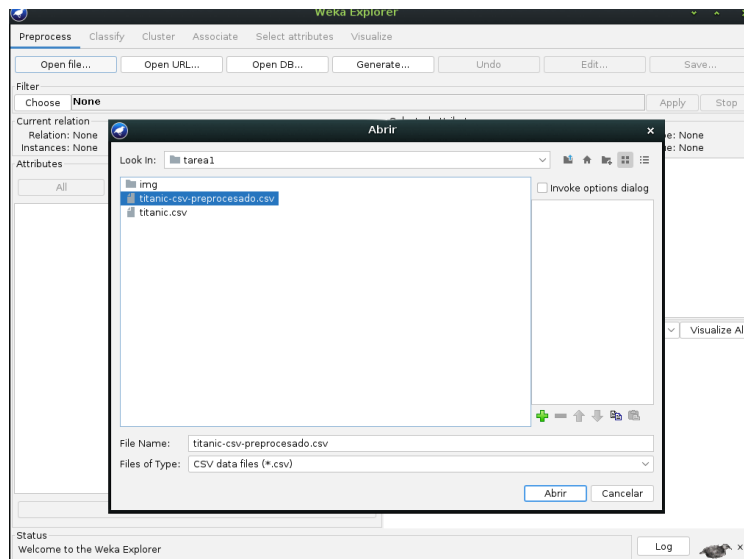


Figura 23: Cargando una archivo .csv pre-procesado

- En esta guía se pide considerar unicamente las variables: *Pclass*, *Age*, *TypeAge*, *Sex*, *Survived*, *Emkarked* por lo que vamos a eliminar los atributos que no se van a analizar, dando click en el botón *Remove*.
- Los atributos a analizar se componen:
 - PClass (0 = tripulación, 1 = primera, 2 = segunda, 3 = tercera)
 - TypeAge (1 = adulto, 0 = niño)
 - Sex (1 = hombre, 0 = mujer)

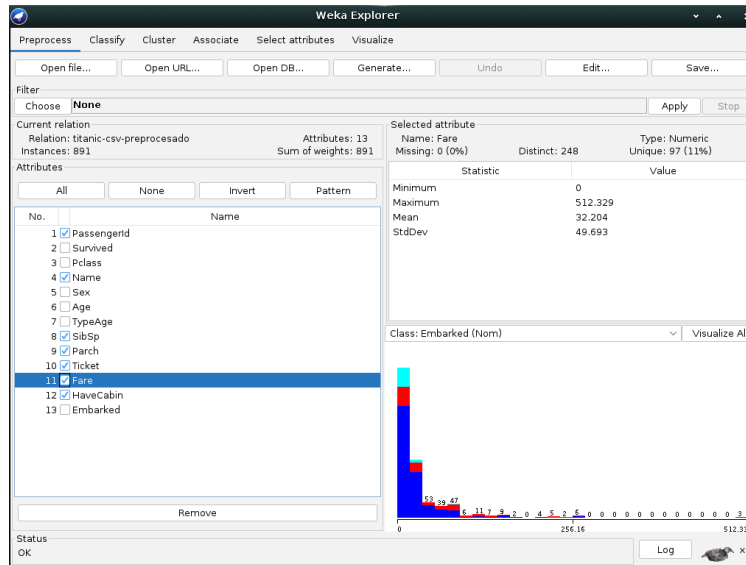


Figura 24: Eliminando columnas inservibles

- Survived (1 = sí, 0 = no)

Al dar click en el botón Visualize All, nos muestra un resumen (histograma) de todas las variables:

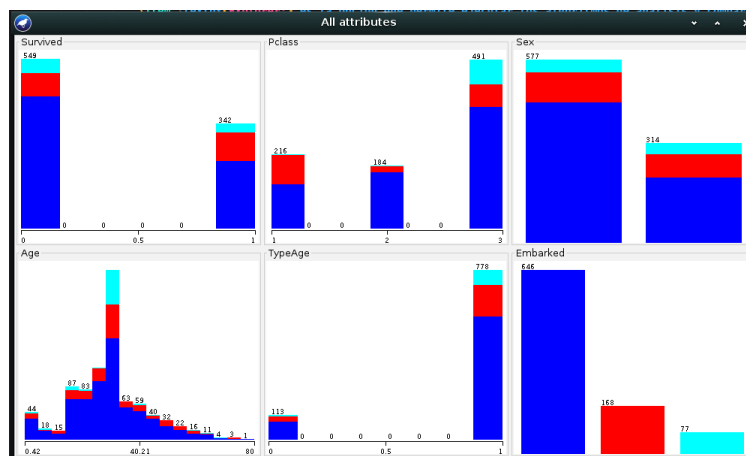


Figura 25: Histograma de todas las variables

- seleccionamos *Filter* → *Choose* → *filters* → *unsupervised* → *attribute* → *NumericTo-Nominal*, y seleccionamos las siguientes opciones:

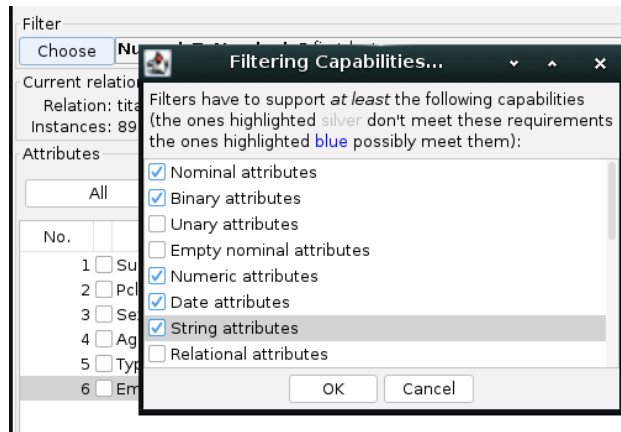


Figura 26: Seleccionando filtros

- seleccionamos el botón *Apply* de la sección de *Filter*.

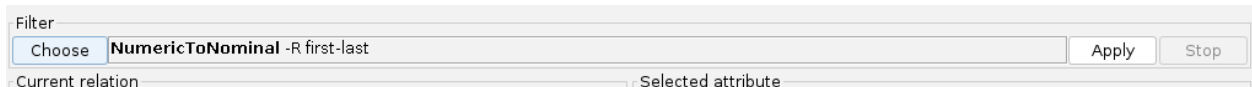


Figura 27: Aplicando filtros

- Ahora podemos hacer la tarea de clasificación *Classify*, Weka ofrece 4 opciones en el Test Options:
 - **Use training set:** la muestra es usada para entrenar y probar al mismo tiempo. Los resultados obtenidos no corresponden a la realidad.
 - **Supplied test set:** los atributos de los datos son escritos en un nuevo archivo de formato ARFF sobre el cual se efectuará la clasificación.
 - **Cross-validation:** permite dividir la muestra en k partes, sobre estas se procede a entrenar el clasificador con las k-1 partes y evaluar con la k parte actual.
 - **Percentage split:** indica el porcentaje de la muestra que empleara para probar el clasificador.
- Weka ofrece 8 opciones para clasificar:
 - **Bayes:** métodos basados en el aprendizaje de Bayes.
 - **Functions:** métodos matemáticos.
 - **Lazy:** métodos basados en el aprendizaje perezoso.
 - **Meta:** métodos que resultan de la combinación de diferentes métodos de aprendizaje.
 - **Mi:** métodos que aprenden mediante la variación de la densidad de los algoritmos.
 - **Misc:** métodos que aprenden como si leyera los datos.

- **Trees:** métodos que aprenden mediante árboles de decisión.
- **Rules:** métodos que aprenden y estos se pueden expresar como reglas.

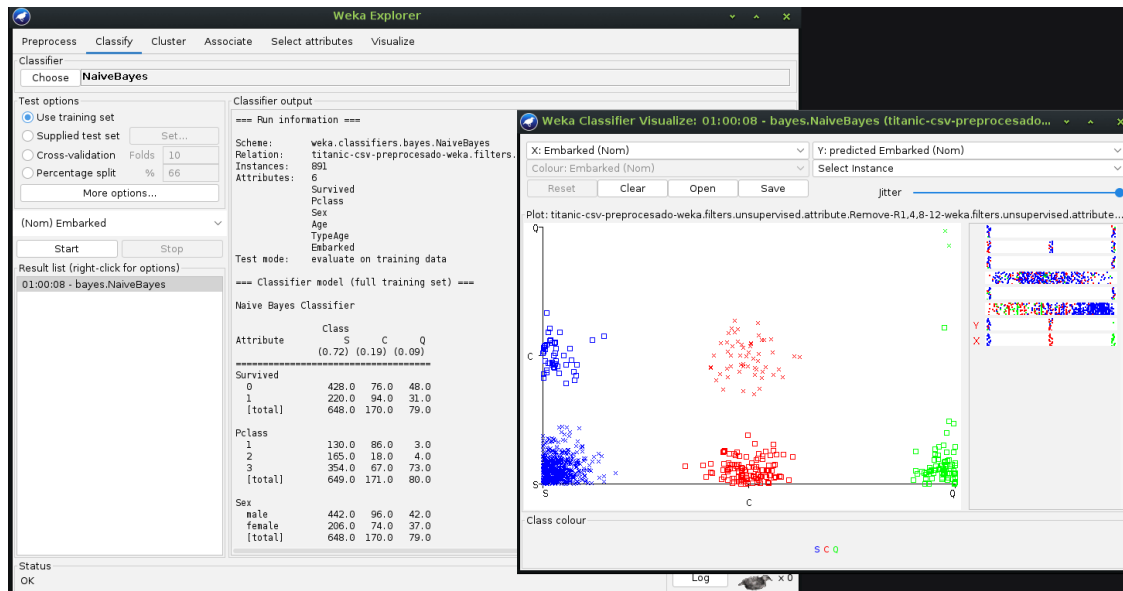


Figura 28: Test *Use training set* con opción de clasificación *Bayes* - errores del clasificador

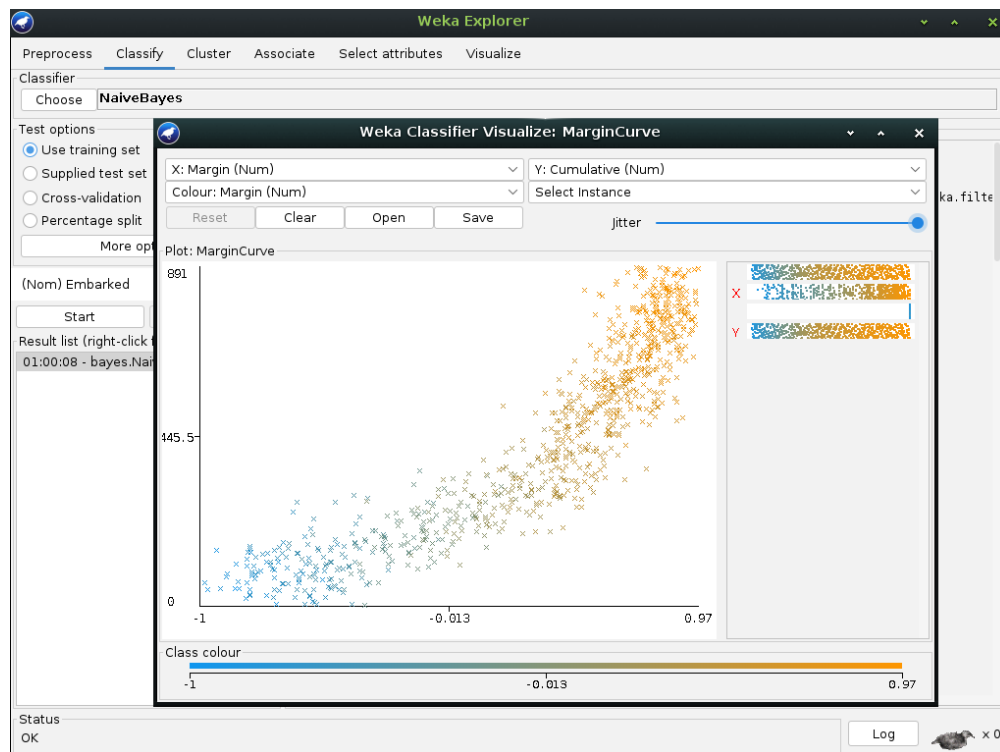


Figura 29: Test *Use training set* con opción de clasificación *Bayes* - curva de margen

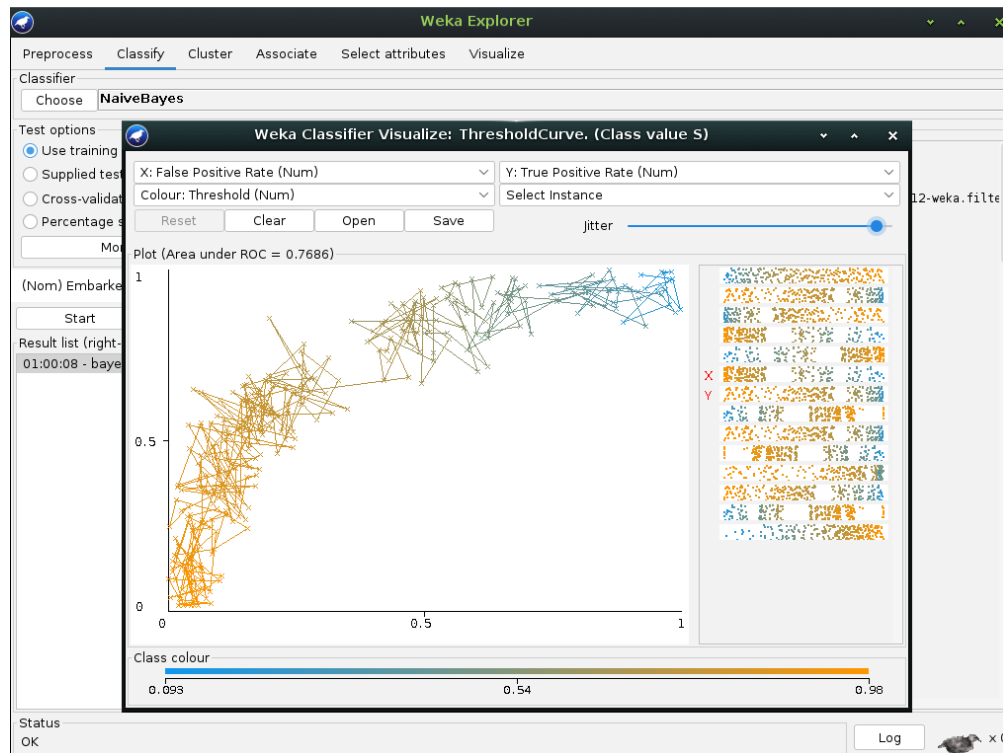


Figura 30: Test *Use training set* con opción de clasificación *Bayes* - curva de umbral S

- Principales algoritmos utilizados para la clasificación:
 - **BayesNet:** Aprende Reyes Bayesianas.
 - **NaiveBayes:** Clasificador discriminador de Bayes.
 - **Id3:** Árboles de decisión usando el divide y vencerás.
 - **J48:** Árboles de decisión usando el C4.5.
 - **RandomForest:** Construye un bosque aleatorio.
 - **JRip:** Construye reglas con el algoritmo RIPPER.
 - **M5Rules:** Construye reglas M5 desde árboles.
 - **LinearRegression:** Utiliza la regresión lineal.
 - **MultilayerPerceptron:** Usa red neuronal de Retroprogramación.
 - **RBFNetwork:** Usa Red de función en Radio base.
 - **SMO:** basado en vectores de soporte.
 - **Ibk:** Usa k vecinos más cercanos.
 - **LWL:** Aprendizaje basados en Pesos Locales.
 - Entre muchos otros.

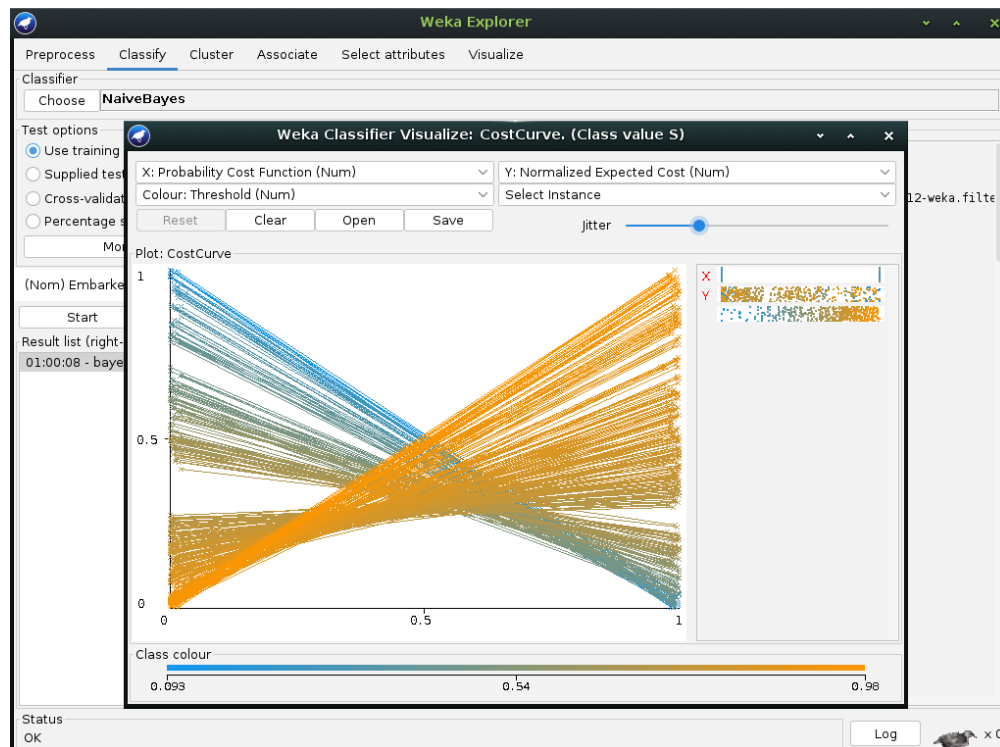


Figura 31: Test *Use training set* con opción de clasificación *Bayes* - curva de costos S

- Weka ofrece cuatro opciones en Cluster mode:
 - **Use training set:** la muestra es usada para entrenar y probar al mismo tiempo. Los resultados obtenidos no corresponden a la realidad.
 - **Supplied test set:** los atributos de los datos son escritos en un nuevo archivo de formato ARFF sobre el cual se efectuará la clasificación.
 - **Percentage split:** indica el porcentaje de la muestra que empleara para probar el clasificador.
 - **Classes to cluster evaluation:** permite escoger el atributo a agrupar.

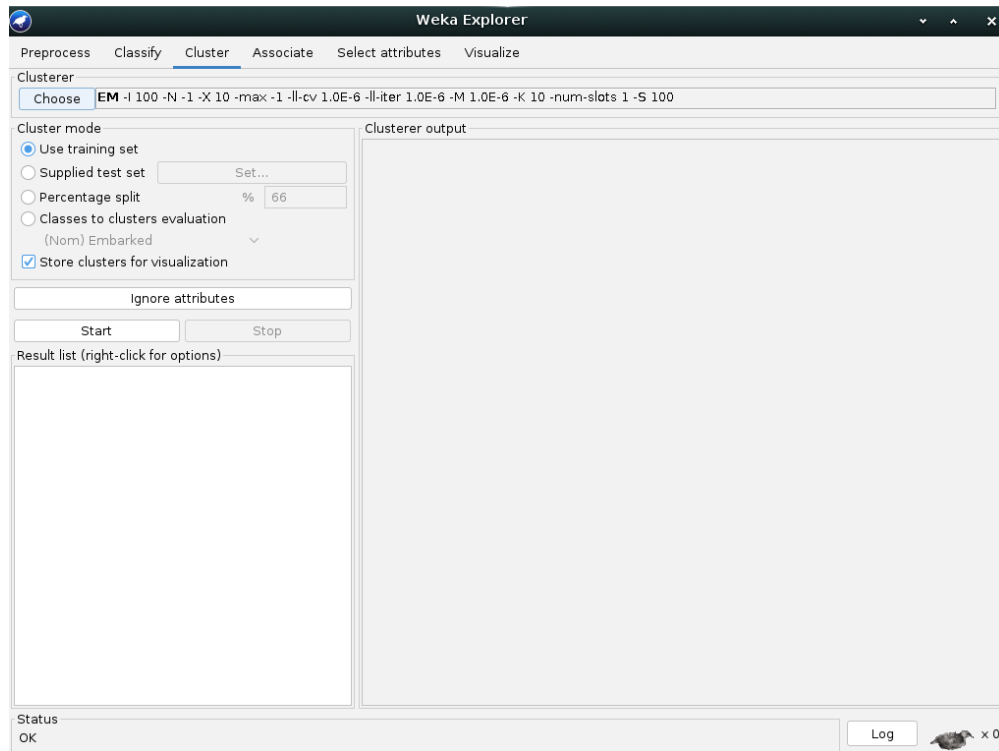


Figura 32: Opción de Cluster

- Weka ofrece los algoritmos para agrupar datos:
 - **Canopy:** Utiliza el algoritmo Canopy.
 - **CobWeb:** Utiliza el algoritmo CoWeb.
 - **EM:** Utiliza el algoritmo EM.
 - **FarthestFirst:** Utiliza el algoritmo FarthestFirst.
 - **FilteredClusterer:** agrupa los datos arbitrariamente y luego son pasados por un filtro arbitrario.
 - **HierarchicalClusterer:** agrupa los datos jerárquicamente y luego son pasados por un filtro arbitrario.
 - **MakeDensityBasedClusterer:** los datos son envueltos en clases y devuelven su distribución y densidad.
 - **OPTICS:** utiliza el algoritmo OPTICS.
 - **SimpleKMeans:** utiliza el algoritmo de k-medias.

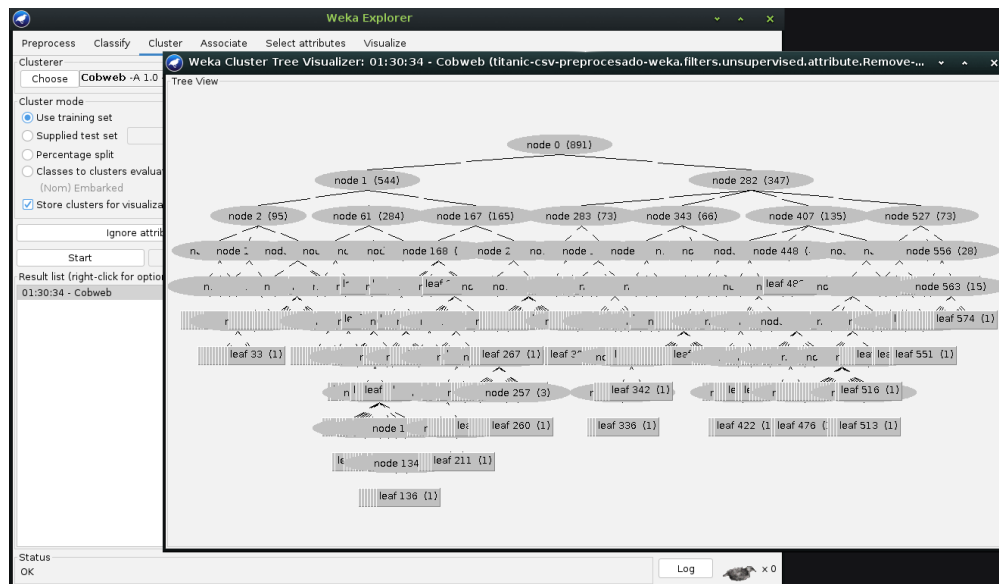


Figura 33: Cluster - *CobWeb* - *Use training set* - visualizar árbol

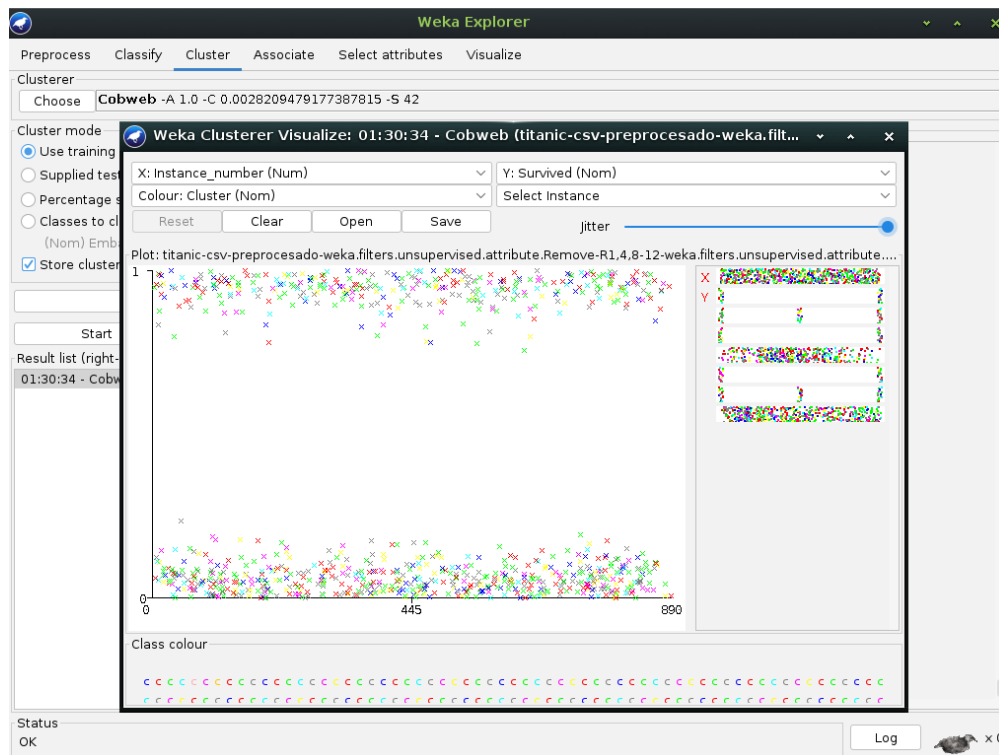


Figura 34: Cluster - *CobWeb* - *Use training set* - visualizar asignaciones de clúster

- Para ejecutar los métodos en Weka de reglas de asociación, seleccionamos la ventana de *associate*, Weka ofrece los algoritmos para asociar datos:
 - **Apriori:** Utiliza el algoritmo Apriori.

- **FilteredAssociator:** utiliza el algoritmo que asocia los datos arbitrariamente y también los filtra arbitrariamente.
- **FPGrowth:** El algoritmo de crecimiento FP representa la base de datos en forma de árbol llamado árbol de patrones frecuentes o árbol FP.

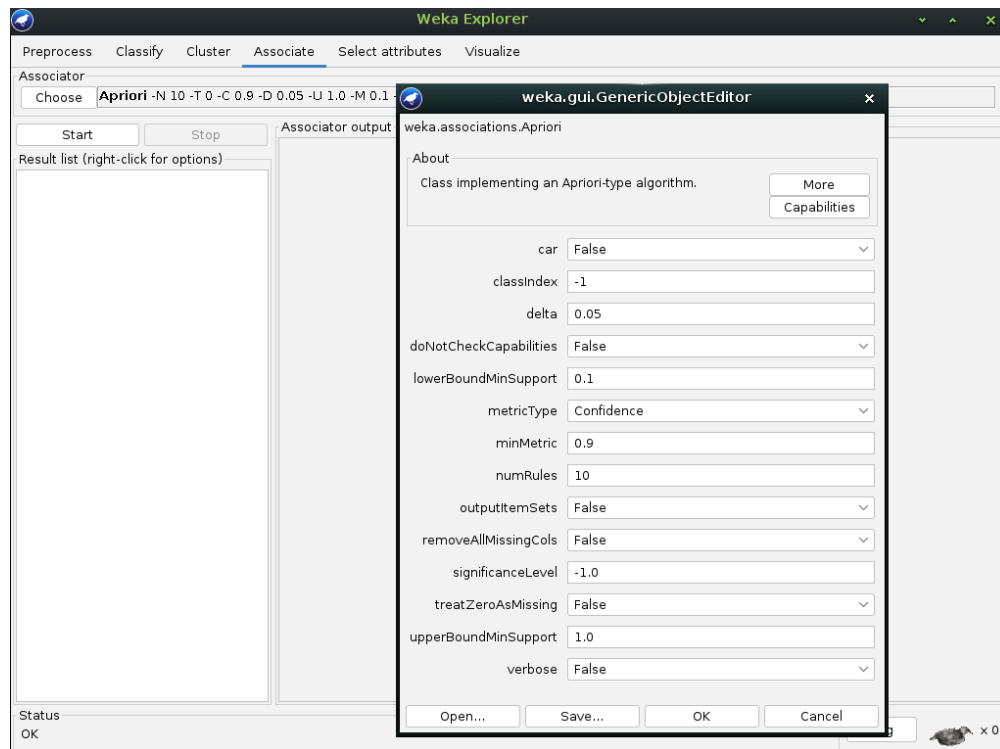


Figura 35: Técnica de minería de datos - Apriori por defecto

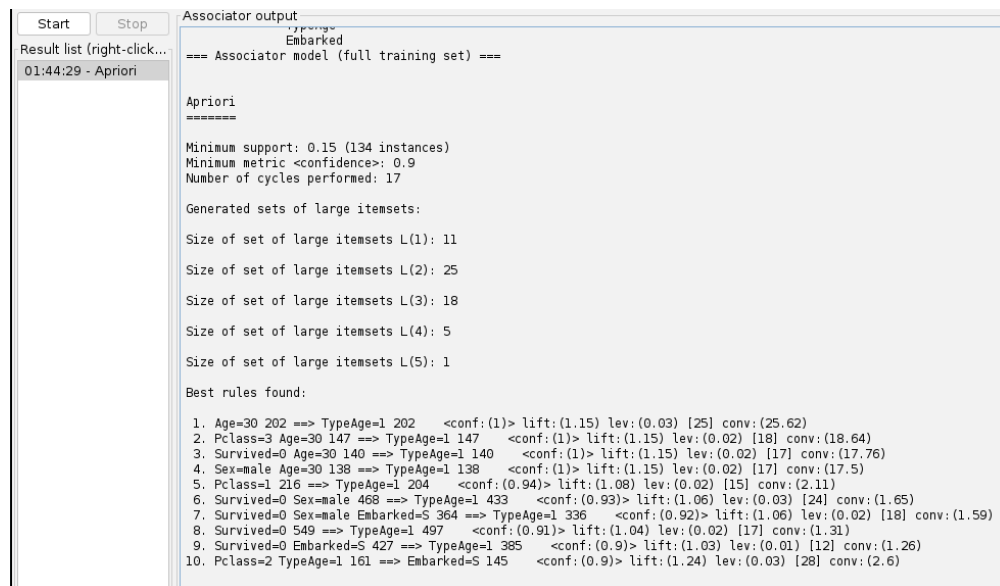


Figura 36: Técnica de minería de datos - Apriori por defecto

En cada regla, tenemos la cobertura de la parte izquierda y de la regla, así como la confianza de la regla. Por ejemplo, la regla 1 indica que, como se supone todas las personas de la clase 1 son adultas. La regla 4 nos indica lo mismo que regla 1, pero teniendo en cuenta a los varones. Parecidas conclusiones se pueden observar de la regla 2,5 y 10. La regla 6 nos indica que los varones que murieron fueron en su mayoría adultos 93 %. La regla 8 nos indica que la mayoría que murieron eran adultos 91 %. Y finalmente, la regla 7 nos indica que la mayoría de muertos fueron hombres 91 %.

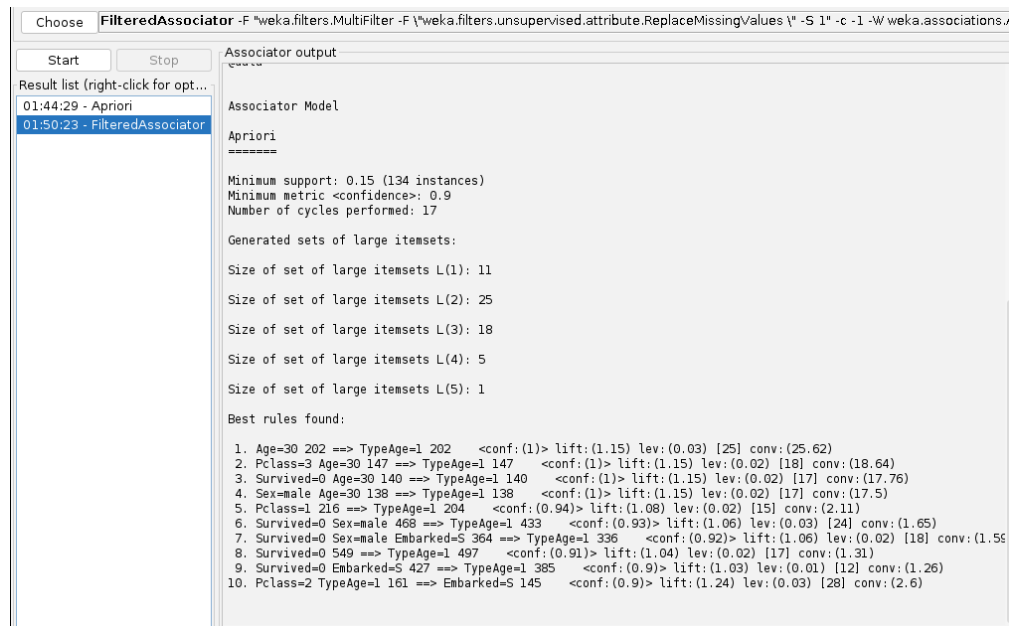


Figura 37: Técnica de minería de datos - FilteredAssociator

- Visualización:

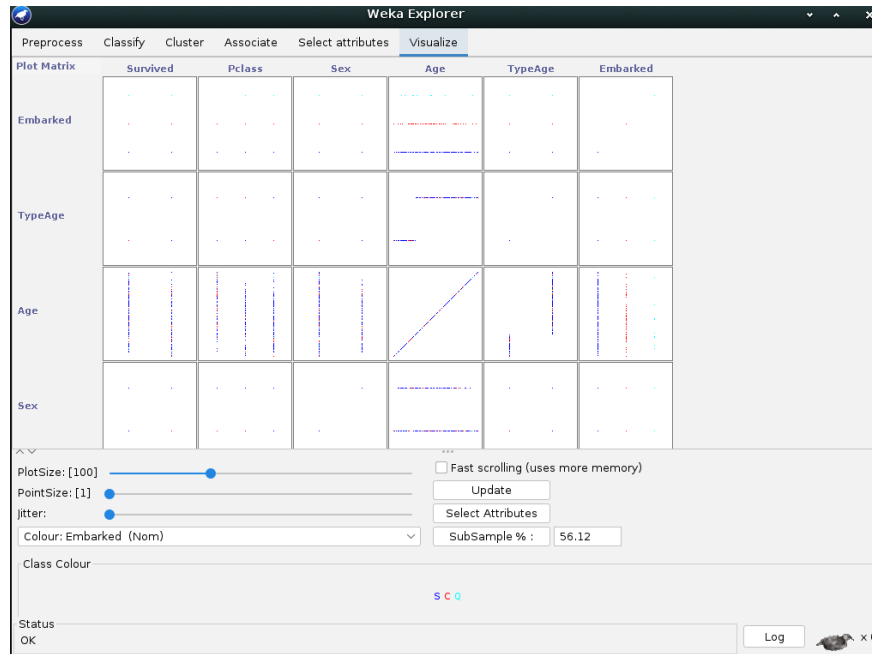


Figura 38: Visualización

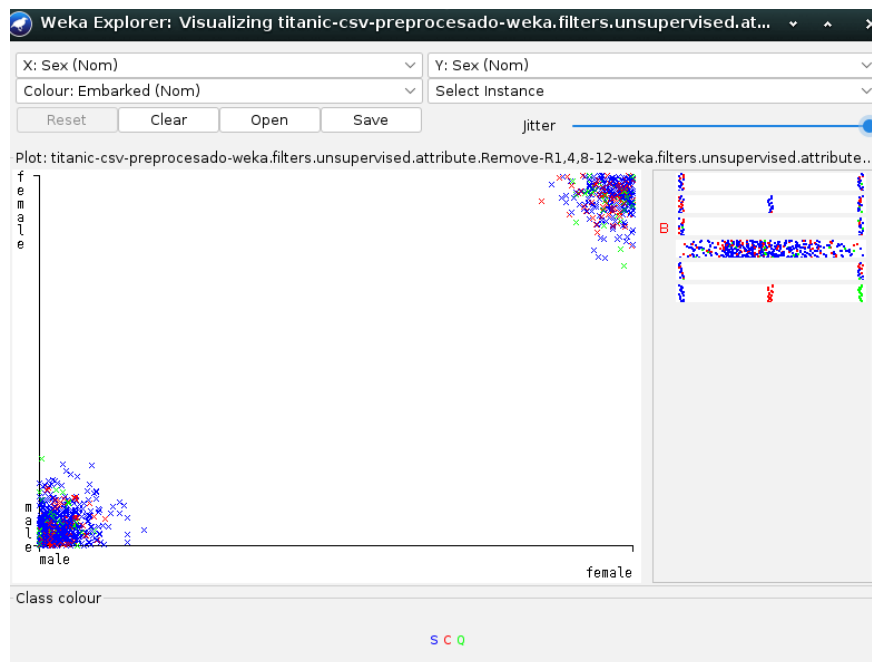


Figura 39: Visualización

4. Errores y correcciones realizadas a la data

Durante el proceso de limpieza y preprocesamiento de la data del archivo titanic.csv, se identificaron varios errores típicos:

- Primero, se renombraron columnas con nombres genéricos, como Column, a nombres más descriptivos, como Vacía.
- Se transformaron columnas con valores numéricos (Survived, PClass, Sibsp, Parch, Age, Fare) de formato de cadena a sus respectivos tipos numéricos.
- Se identificaron valores nulos en las columnas Age, Cabin y Embarked.
- Para Embarked, los valores faltantes se reemplazaron con "S", indicando Southampton.
- En el caso de Cabin, se creó una columna adicional llamada HaveCabin para marcar con 1 si la cabina existe y con 0 si no existe.
- Los valores nulos en Age se reemplazaron con la media de la columna.
- Se eliminaron las columnas innecesarias, como Vacía, Record, Cabin, y AgeProm, una vez que ya no eran útiles.
- Para manejar posibles duplicados en PassengerId, se identificaron y eliminaron registros duplicados.
- Adicionalmente, se creó la columna TypeAge para clasificar edades en menores de 18 (0) y mayores o iguales a 18 (1).
- Finalmente, todos los datos se regresaron a tipo cadena para mantener la consistencia en el formato.

5. Resultados del análisis de la data

Usando Weka para analizar los datos preprocesados del Titanic, se pueden obtener varios resultados:

- Los algoritmos de clasificación, como NaiveBayes, J48 (árbol de decisión), y Random-Forest, permitirán predecir la probabilidad de supervivencia de los pasajeros basándose en atributos como Pclass, TypeAge, Sex, y Embarked. Por ejemplo, se puede identificar que los pasajeros de primera clase tenían una mayor probabilidad de supervivencia.
- Utilizando reglas de asociación como el algoritmo Apriori, se pueden descubrir patrones significativos, como que la mayoría de los adultos varones no sobrevivieron, mientras que muchas mujeres y niños sí lo hicieron. Además, con algoritmos de agrupamiento como k-means, se pueden segmentar los pasajeros en grupos con características similares, lo cual podría revelar insights sobre la distribución y comportamiento de diferentes grupos en el barco.
- La visualización de estos análisis a través de histogramas y gráficos facilitará la interpretación y comunicación de los resultados, permitiendo una comprensión más clara de los factores que influyeron en la supervivencia durante el hundimiento del Titanic.

6. Conclusiones y recomendaciones

6.1. Conclusiones

1. Aprendimos a utilizar Kaggle para acceder a conjuntos de datos reales.
2. Simplificamos la limpieza y preparación de datos utilizando OpenRefine, mejorando nuestra habilidad en la manipulación de datos.
3. Repasamos de manera muy superficial sobre los diferentes algoritmos de aprendizaje automático con Weka, como clasificación y agrupamiento.
4. Analizamos el dataset del Titanic para obtener insights prácticos sobre la supervivencia en contextos históricos.
5. Desarrollamos habilidades analíticas al aplicar técnicas de análisis de datos en un problema reales de ciencia de datos.
6. Integramos Kaggle, OpenRefine y Weka para mejorar nuestra capacidad en la toma de decisiones basadas en datos.

6.2. Recomendaciones

1. Proporcionar una explicación más detallada sobre el uso de la aplicación Weka, enfocándose en hacer la herramienta accesible para estudiantes con menos experiencia técnica.
2. Explorar con mayor profundidad conceptos avanzados en aprendizaje automático y sus diversos algoritmos.

7. Bibliografía

- Invitado. (2016, 6 julio). Open Refine – qué es + tutorial. Escuela de Datos.
<https://es.schoolofdata.org/2014/06/30/openrefine/index.html>
- Isaiaranda. (2021, 16 diciembre). ¿QUE ES WEKA? - Isaiaranda - Medium. Medium.
<https://medium.com/@isaiaranda15/que-es-weka-926c05050d44>
- Guía practica proporcionada por la docente.