UNIVERSIDAD D CÓRDOBA

Preprocesamiento de datos

Análisis automático de datos para ciencias biomédicas (Transversal Másteres Universitarios)

Juan Carlos Fernández Caballero
Departamento de Informática y Análisis Numérico
Universidad de Córdoba
curso 2021-2022

www.uco.es/ayrna

Agradecimientos

- Estas diapositivas se han elaborado con la colaboración de:
  - Pedro Antonio Gutiérrez Peña pagutierrez@uco.es.
  - Javier Sánchez-Monedero sanchez-monederoj@cardiff.ac.uk.
  - María Pérez-Ortiz maria.perez@ucl.ac.uk.
  - César Hervás Martínez chervas@uco.es.
  - Diversas fuentes relacionadas con la temática (consultar bibliografía).

Índice

Índice

¿Preprocesar?

Objetivo del preprocesamiento

Recordatorio: Etapa 2 del proceso de Ciencia de Datos.

Objetivo: Mejorar la calidad de los datos, de forma que:

- Se elimine el ruido o "suciedad" que pueda afectar a los modelos.
- Los datos sean interpretables por los algoritmos.
- Se pueda inferir (extraer) el máximo conocimiento.
- Se consiga un mejor rendimiento.

Datos de calidad $\longrightarrow$ **Resultados de calidad**.

Ruido en los datos

Los datos del mundo real están "*sucios*", con lo que aportan ruido a los módelos:

- **Incompletos:** Datos perdidos. ¿Se podrían recuperar o reemplazar, por ejemplo, por la media?
- **Mala representación y consistencia en el formato:**
  - Ej: Formato de números en separador de cientos y miles.
  - Ej: Formato de fechas: 2020-03-04; 4/3/2020.
  - Ej: Valor de mediciones: Temperatura en Celsius o Fahrenheit.
- **Transformación de variables:**
  - Ej: Calores nominales que no se pueden tratar por el algoritmo "X".
  - Ej: Normalización, de forma que los datos estén en la misma escala.
  - Ej: Discretización.
- **Duplicidad:** Existencia de duplicidad de patrones.
- **Mediciones erroneas y/o extremas:**
  - Ej: Errores en la toma o transcripción de datos.
  - Ej: Valores atípicos o extremos.

Ruido en los datos

- **Selección de información en cuanto a instancias:**
  - Ej: Demasiadas instancias o pocas instancias.
  - Ej: Patrones irrelevantes que se deban eliminar dependiendo del valor que tengan en un determinado atributo.
- **Selección de información en cuanto a atributos:**
  - Ej: Existen demasiados atributos o muy pocos atributos.
  - Ej: **¿Existen atributos redundantes?:** Análisis de correlaciones o selección de características para eliminar los redundantes.
  - Ej: **¿Existen atributos de diferentes fuentes que representen lo mismo?:** Distinto a correlación, posibilidad de eliminarlos.
  - Ej: **¿Existen atributos que no aporten información?:** Se llaman atributos identificadores, por ejemplo el dni de las personas o el número de un patrón.

Relevancia de los datos

¿Son los datos que tenemos **relevantes y suficientes** para nuestra tarea?.

Algunas reglas **orientativas** generalmente aceptadas (pero no aplicables siempre):

- Número de atributos: **Para cada atributo debería haber 10 o más instancias**.
- Número de instancias: **Por cada clase al menos 100 instancias**.

Base de datos **desbalanceada**: Aquella que tienen clase/s que tiene muchos menos patrones que el resto → aplicar métodos específicos para balancear.

- **Métodos de *Oversampling***: Aumentan el número de instancias de las clases menos representadas.
- **Métodos de *Undersampling***: Disminuyen el número de instancias de las clases más representadas.

  Buscar en la web: Oversampling and undersampling in data analysis.

Índice

Visualización

Visualización

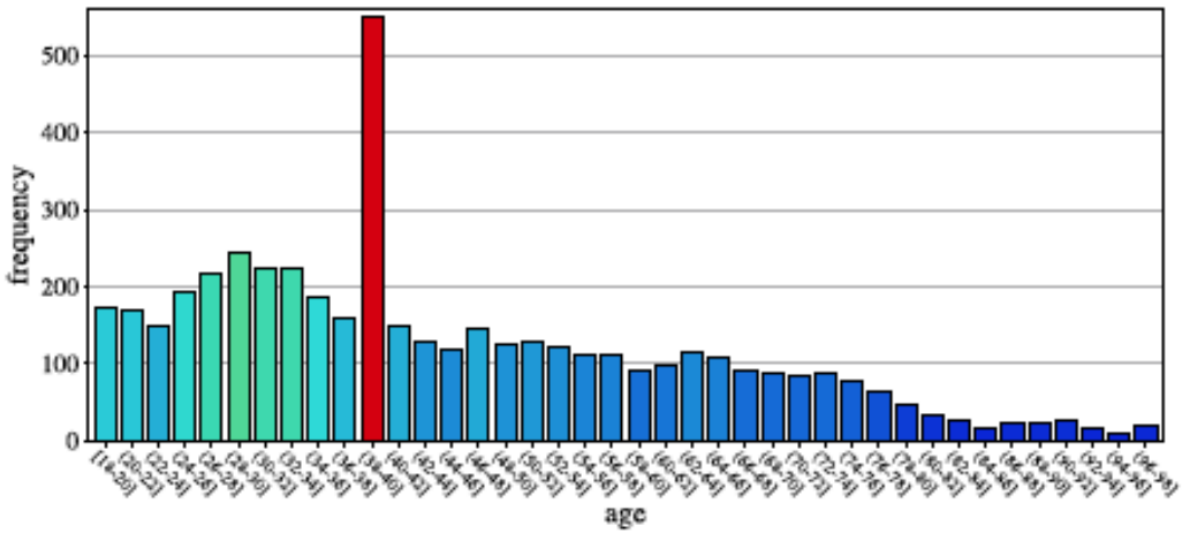
- Antes incluso del preprocesamiento, es necesario **analizar las características de los datos** para **conocer mejor el problema**.
- Par ello, podemos **convertir la información en una representación gráfica** que nos ofrezca una visión más coherente de los datos.
- Esto ayuda a **enfocar las tareas de preprocesamiento** a realizar y permite, por ejemplo:
  - Detectar posibles **datos erróneos**.
  - Detectar y/o comparar posibles **tendencias** o **frecuencias inusuales**.
  - Detectar **dependencias** o correlaciones.
  - Detectar *outliers* (**valores atípicos o extremos**).
  - Detectar **valores perdidos**.

Visualización

Los histogramas son diagramas de barras que pueden servir para:

- Dar una visión de la **distribución de la población respecto a una característica o atributo**.
- Mostrar el **grado de homogeneidad** o de **variabilidad** de los datos.
- Mostrar **frecuencias inusuales** que podrían venir de un valor etiquetado incorrectamente.

Buscar en la web: histogramas en machine learning o en análisis de datos.



Visualización

Diagramas de caja: *boxplot*

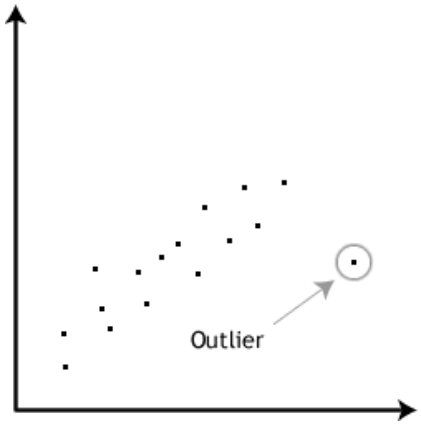
- Proporcionan el valor **máximo**, el **mínimo**, la **mediana** y los **cuartiles**.
- Ofrecen una visión de la **simetría y dispersión que siguen los datos**.
- Desvelan la presencia de posibles ***outliers*** **y valores extremos**.
- Buscar en la web: diagrama de caja para más información.

Figura: Boxplot atributo SalePrice.

Visualización

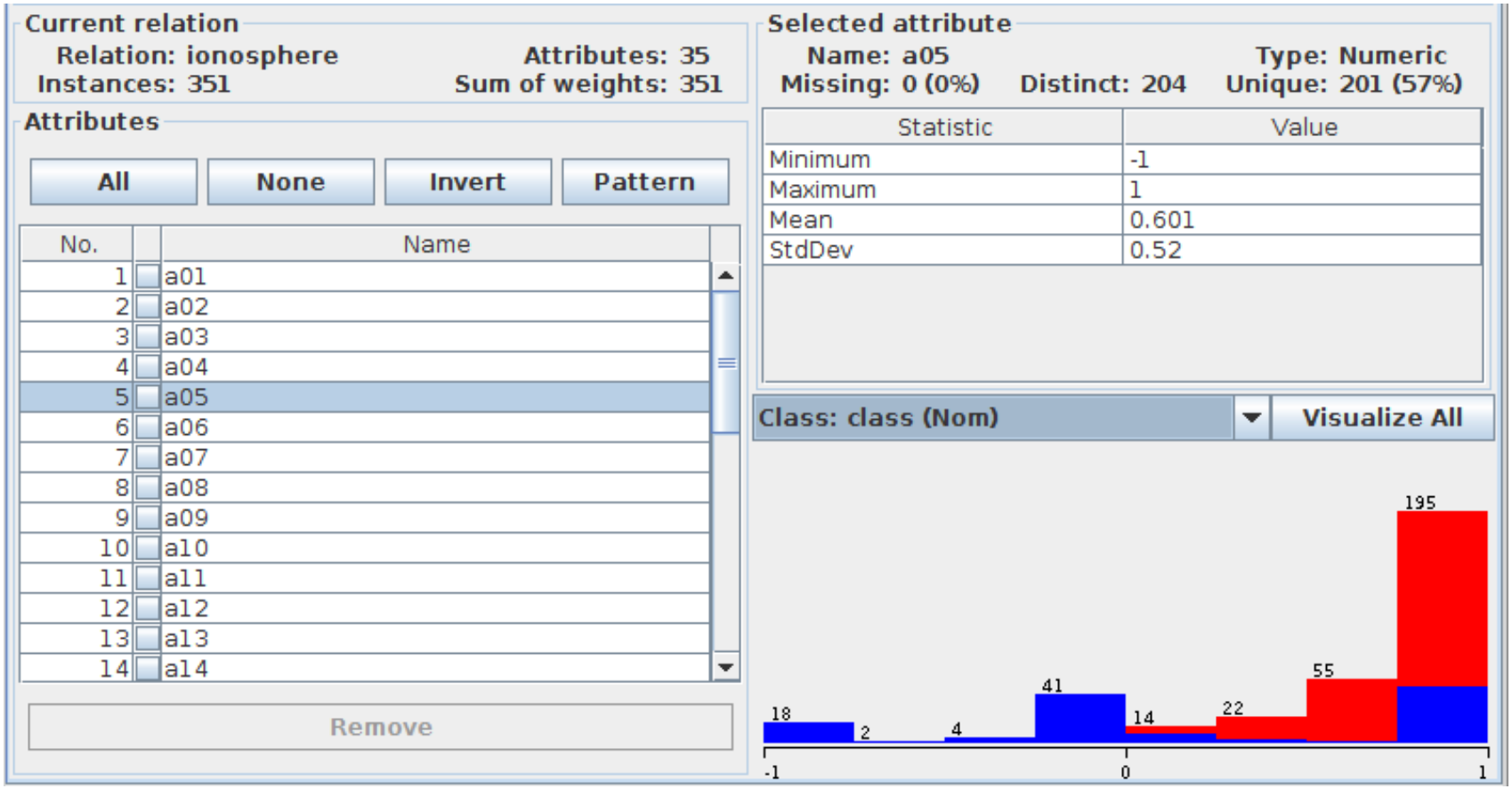
Gráficos de dispersión: *scatter plot*

- Estudian la **relación existente entre dos atributos**.
- Pueden **sugerir correlaciones entre los atributos**.
- También útiles para **detectar *outliers* y valores extremos**.
- Buscar en la web: diagrama de dispersión para más información.



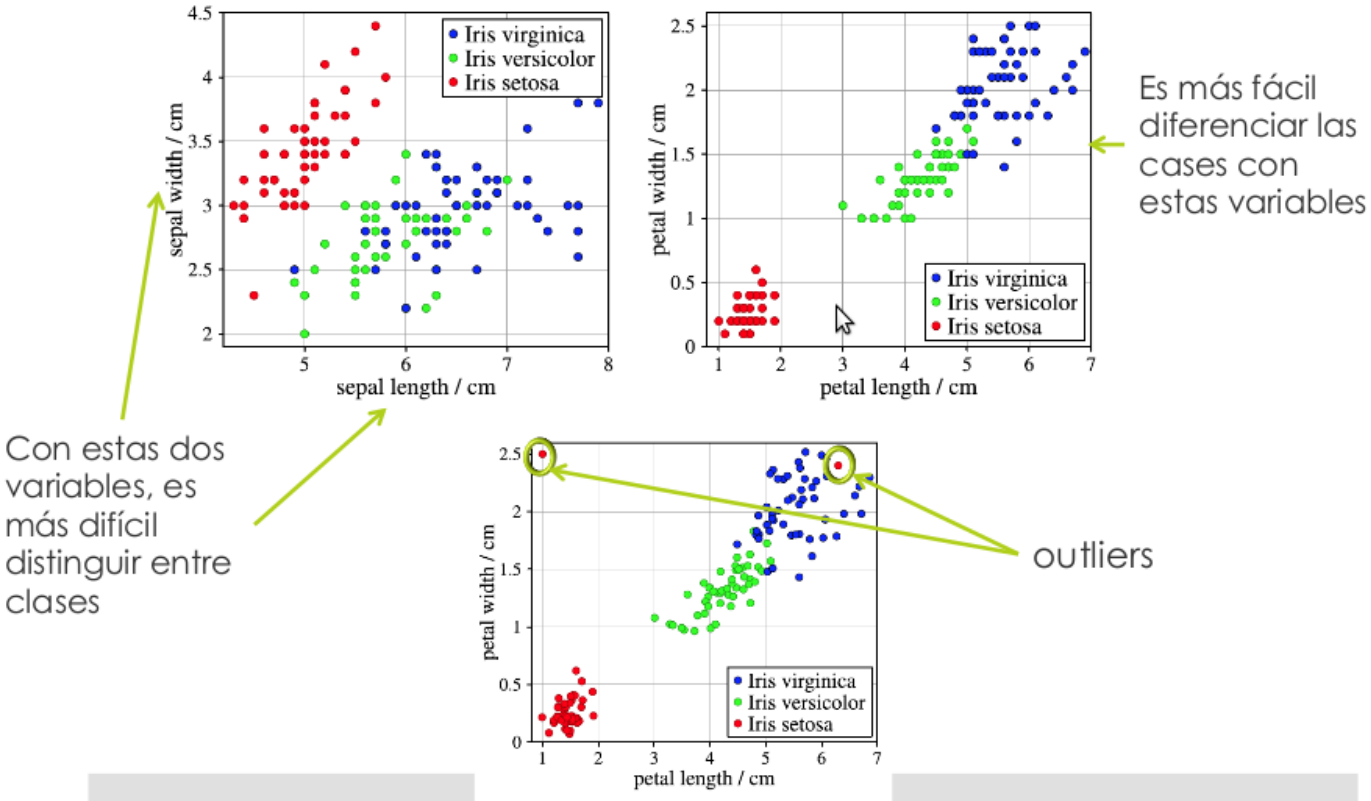
Histogramas en Weka

- En la pestaña **Explorer->Preprocess**, al seleccionar el atributo.



Scatter plots en Weka

- También la pestaña **Explorer->Visualize** nos permite ver graficos de dispersión (*scatter plots*) en pares de atributos.
- Doble clic en una gráfica permite individualizarla.



Índice

Operaciones de preprocesado

Preprocesado de datos en Weka

En Weka, las **herramientas de preprocesamiento** se denominan **filtros**, y hay dos tipos: supervisados y no supervisados.

- Filtros no supervisados: **No tienen en cuenta la variable objetivo** a predecir (clasificación/regresión).
  - **Ejemplos sobre instancias**: borrar porcentaje, borrar instancias con un valor dado en algún atributo, duplicados,etc.
  - **Ejemplos sobre atributos**: reemplazar valores perdidos, nominal a binario, normalizar, discretizar, reemplazar valores perdidos, etc.
- Filtros supervisados: **Tienen en cuenta la variable objetivo** a predecir para hacer operaciones sobre los datos.
  - **Ejemplos sobre instancias**: *Oversampling*, *Undersampling*, crear *k-folds* estratificados, etc.
  - **Ejemplos sobre atributos**: Selección de características, ordenación de clases, etc.

Filtro Tratar valores perdidos

La mayoría de métodos de Ciencia de Datos no pueden trabajar con valores perdidos. Algunas opciones:

1. Ignorar los patrones: Más del 5 % $\longrightarrow$ pérdida de información.
2. Rellenar manualmente si se conoce el valor (en general inviable).
3. Imputación de datos: substituir los valores por algo.
   - Rellenar usando **media/moda** de los datos/clase.
     - Variables **cuantitativas** $\rightarrow$ usamos la media.
     - Variables **categóricas** $\rightarrow$ usamos la moda (valor más frecuente).
   - Rellenar utilizando distancias.
   - Regresión a partir de otros atributos que no tengan valores perdidos.

Recuperar valores perdidos: ejemplo media

Precaución, este ejemplo presenta un problema:

Person	Highest Education	Salary
A	School	10000
B	Post Graduate	40000
C	Graduate	35000
D	School	11000
E	Graduate	NA
F	Post Graduate	42000
G	Post Graduate	39000
H	Graduate	25000
I	School	12000
J	School	NA
K	Graduate	31000
L	Post Graduate	39500

- Media global de la columna: 28450 (mismo salario para *School* que para *Graduate*).
- Solución: calcular la media por separado para *School* (11000) y para *Graduate* (30333).

Recuperar valores perdidos: ejemplo moda

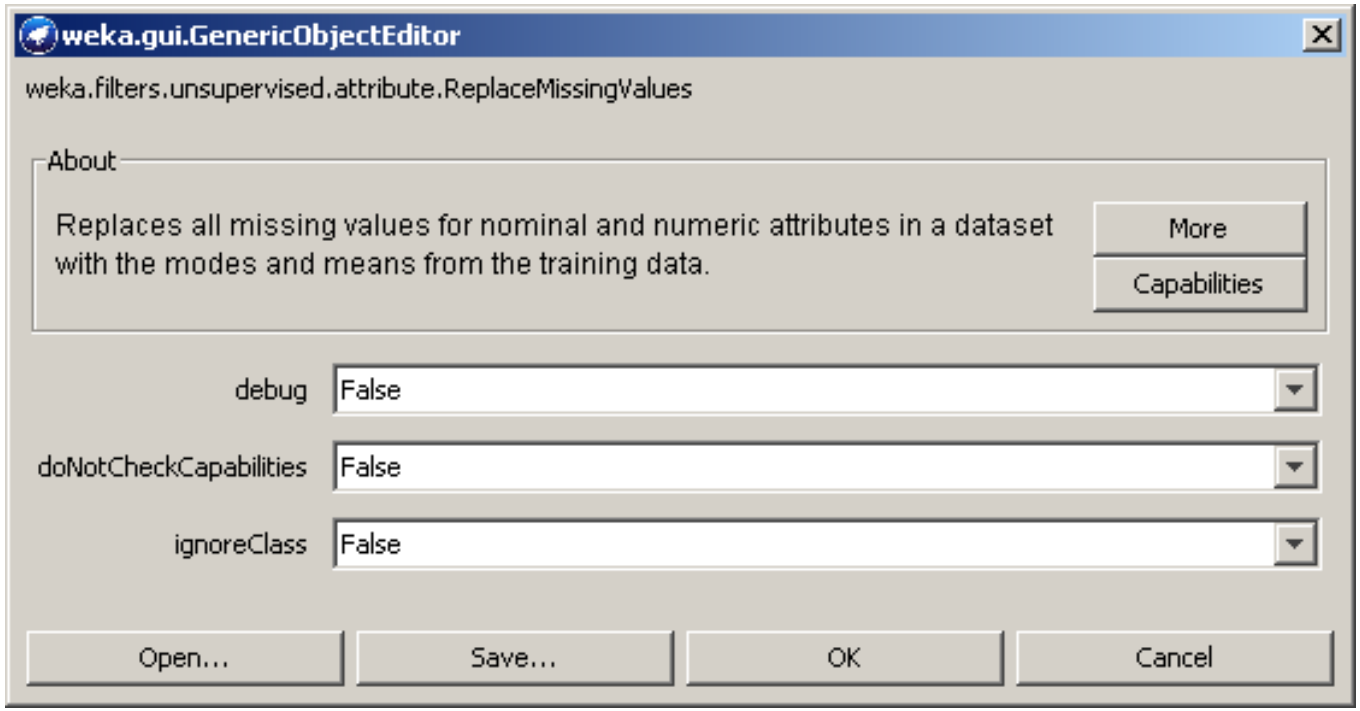
Product	Type	User Rating (0-5)
A	Grocery	5
B	Cream	3
C	Fashion	2
D	Fashion	3
E	Cream	NA
F	Fashion	4
G	Grocery	2
H	Cream	2
I	Cream	5
J	Grocery	1
K	Fashion	4
L	Grocery	4

- En este caso, no podemos usar medias, ya que la variable es **categórica ordinal** (*user rating* u opinión del usuario de 0 a 5).
- El valor medio sería 3,18 y no tiene sentido como categoría.
- Deberíamos calcular la **moda** (preferentemente condicionada a *Cream*, recuerde el problema de la media).

Filtro Valores perdidos en Weka

`filters→unsupervised→attribute→ ReplaceMissingValues`.

Compruebe usted mismo la aplicación del filtro usando cargando algún dataset con datos perdidos en Weka.



Filtro Discretización

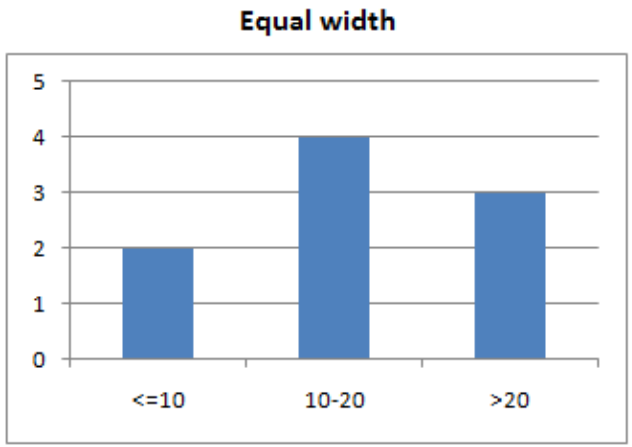
$\longrightarrow$ Algunos métodos de ML **solo permiten trabajar** (o trabajan mejor) con valores **nominales**.

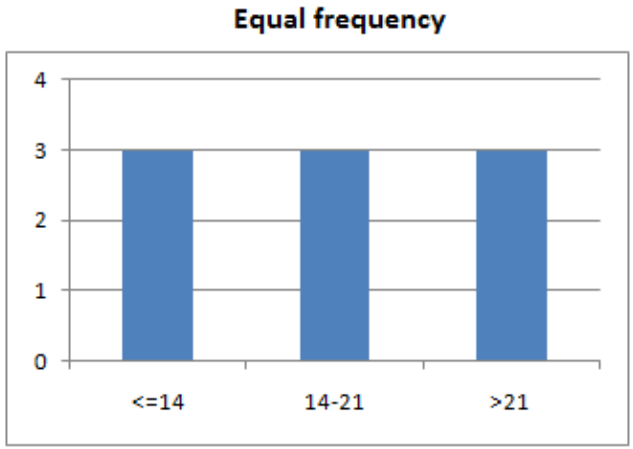
$\longrightarrow$ En otras ocasiones, una variable con valores discretos permite **reducir la cantidad de información** y hacer que los atributos sean más **fáciles de entender**.

- La solución podría consistir en discretizar una variable numérica.
- Por ejemplo, una variable **edad** que toma **valores de 5 a 64 años**. Se podría generar una variable **categórica nominal** con estas categorías: `{edad<=10, 10<edad<=30, 30<=edad<45, edad>=45}`
- **Métodos de discretización no supervisados**:
  - Igual amplitud.
  - Igual frecuencia.
  - *Clustering*: Se basa en agrupar instancias similares (buscar algoritmo *k*-medias en la web).

Filtro Discretización

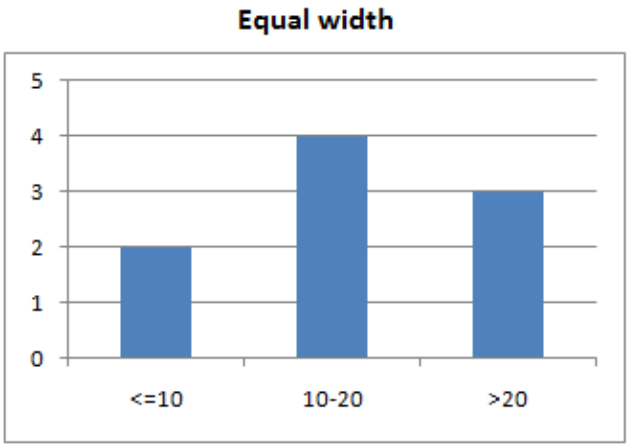
- **Igual amplitud**.
  - Divide el intervalo en k intervalos del mismo ancho W.
  - El valor que debe tomar k debe tener sentido según la semántica del problema $\longrightarrow$ Necesidad del experto.
  - Si m es el valor mínimo y M es el valor máximo, el ancho W será $W = \frac{M-m}{k}$.
  - Es la forma más simple, pero los *outliers* pueden dominar la conversión, ya que su valor es determinante en el ancho.
  - Además, puede generar desbalanceo de las categorías generadas, pocos o muchos patrones en un intervalo.

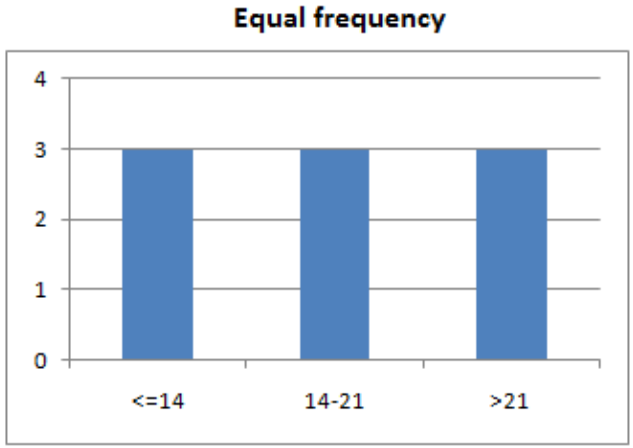




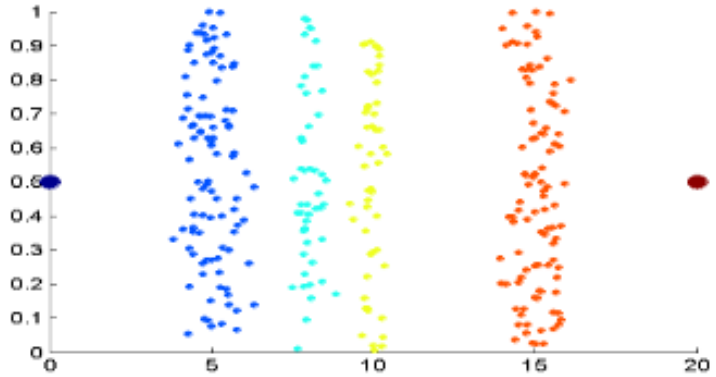
Filtro Discretización

- **Igual frecuencia**.
  - Divide el intervalo en k intervalos de distinto ancho, tratando de generar categorías (intervalos) **balanceadas** en cuanto al **número de patrones por intervalo**.
  - El valor que debe tomar k debe tener sentido según la semántica del problema $\longrightarrow$ Necesidad del experto.
  - Es decir, se fuerza a que, tras la discretización, el número de ejemplos en cada categoría sea, aproximadamente, el mismo.
  - Problema: Quizás haya patrones que debieran entrar en otro intervalo.

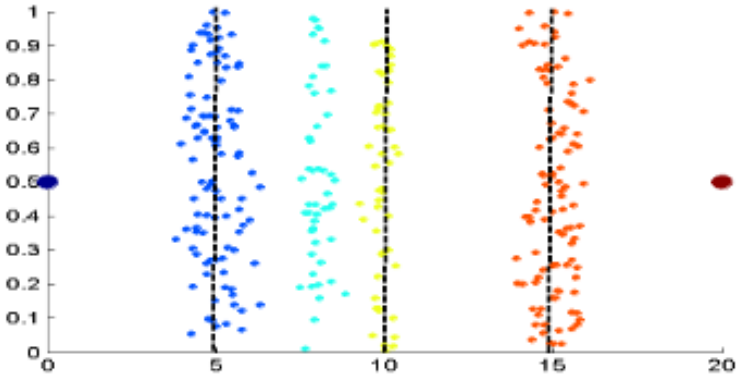




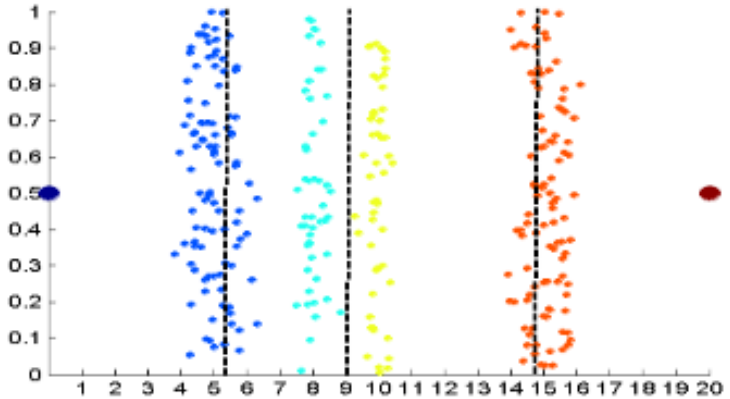
Filtro Discretización



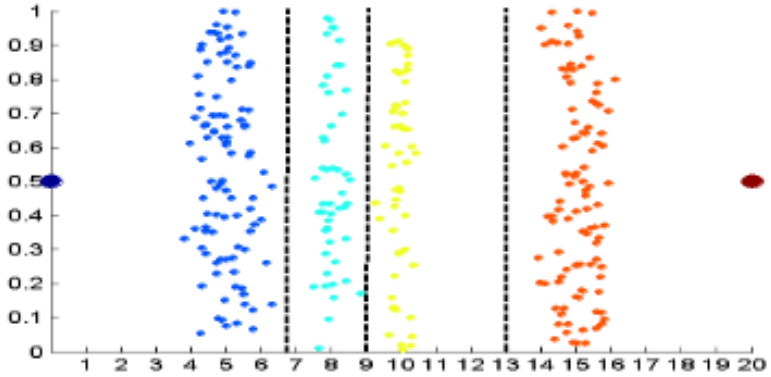
Datos



Igual anchura de intervalo

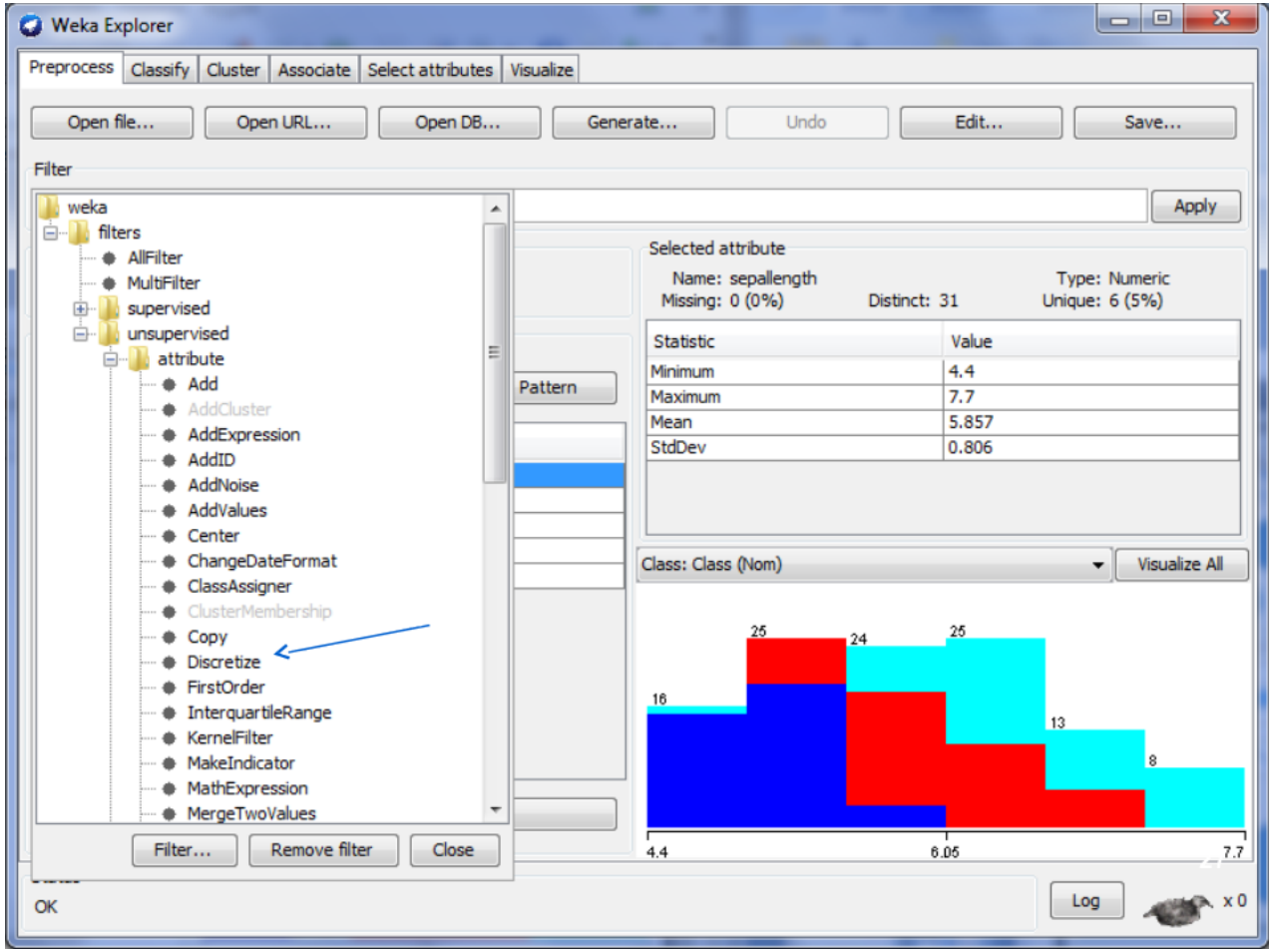


Igual frecuencia



K-medias

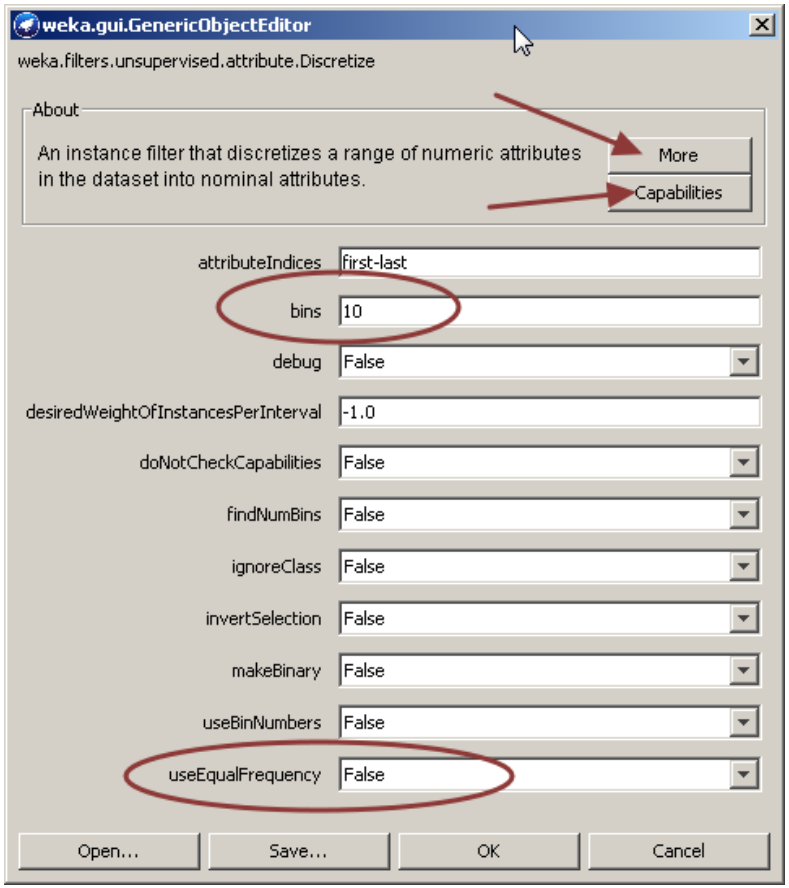
Filtro Discretización en Weka



Filtro Discretización en Weka

Al pichar sobre el nombre del filtro una vez seleccionado, nos aparecen sus parámetros configurables.

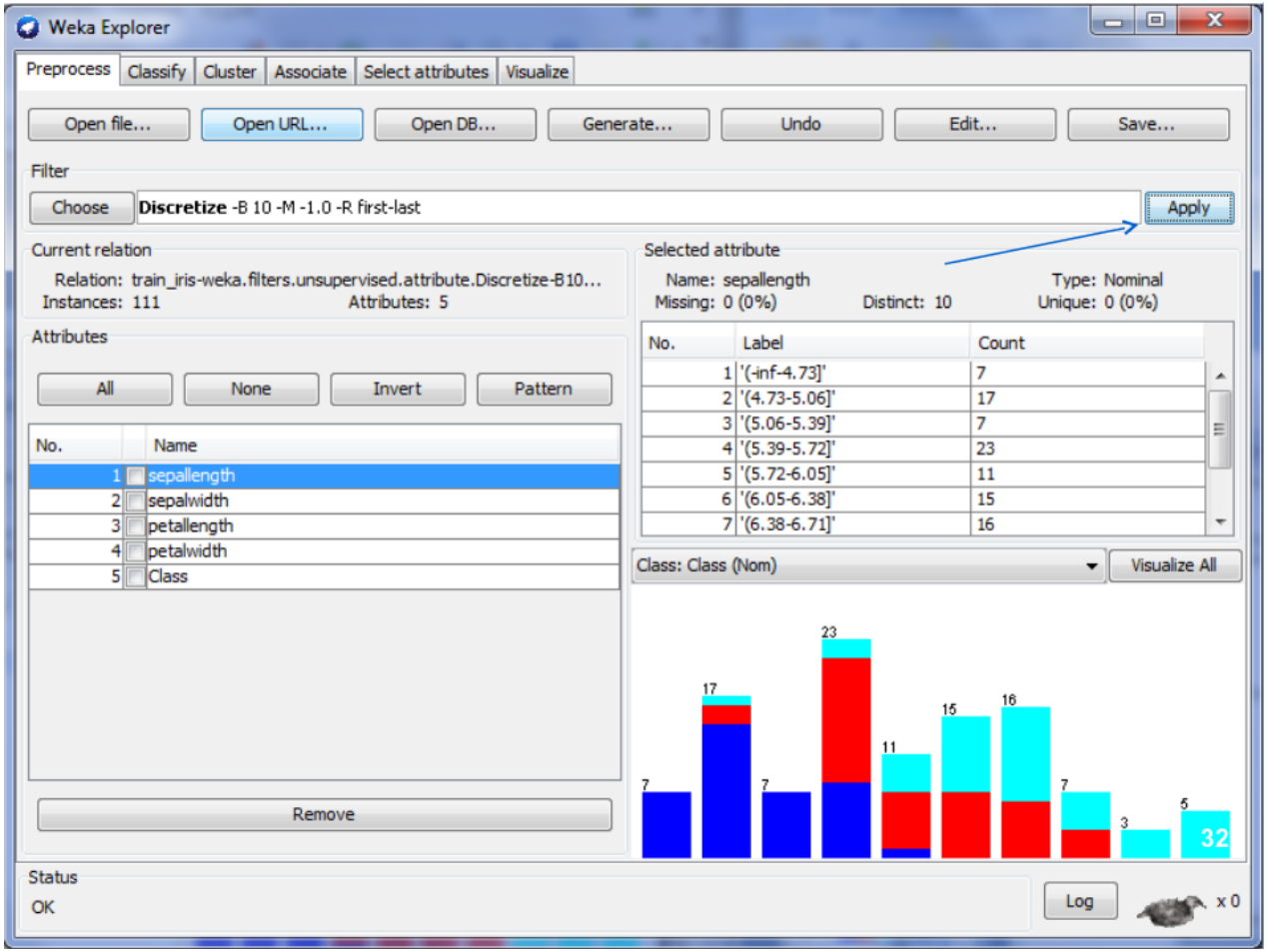
Podemos configurar por **frecuencia** o **amplitud** mediante las opciones "***bins***" y "***useEqualFrecuency***".



Filtro Discretización en Weka

Ejemplo para "**Iris**" con la configuración anterior (igual amplitud):

Compruebe usted mismo la aplicación del filtro usando Weka.



Filtro Normalización y estandarización

- **Normalizar**: Pasar los valores de todos los atributos a un **rango único**, para que todos adquieran la **misma importancia**.
- Permite una mejor **interpretabilidad del modelo, sobre todo en los problemas de regresión, no olvide usarlo**.
- La mayoría de metodologías de ML trabajan mejor con los datos normalizados.
  - En Weka, algunas lo hacen de manera automática y transparente para el usuario.
  - En otros casos conviene que el usuario pruebe el rendimiento normalizando los datos previamente.
- **Normalización** `min-max`: transformación de los datos, usualmente entre [0,1].
  - Atributo A, con valor mínimo m_A y valor máximo M_A.
  - Intervalo deseado: valor mínimo m_A^* y valor máximo M_A^*.

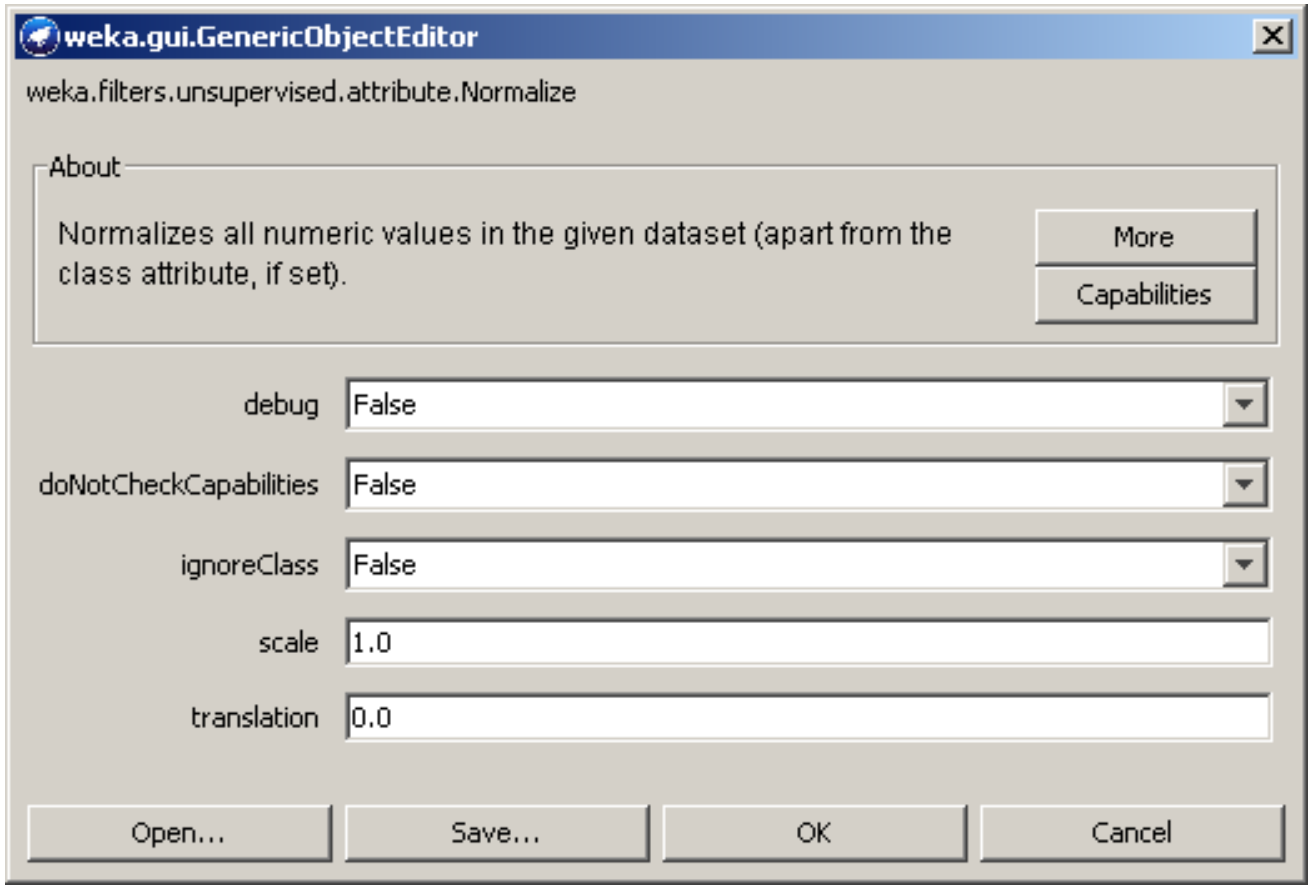
$$[m_A, M_A] \quad \Rightarrow \quad \left[m_A^*, M_A^*\right]$$
$$v \quad \Rightarrow \quad v^*$$
$$v^* = m_A^* + (v - m_A)\,\frac{M_A^* - m_A^*}{M_A - m_A}$$

  - Se conserva la relación entre los datos originales.

Filtro Normalización en Weka

`filters→unsupervised→attribute→ Normalize`

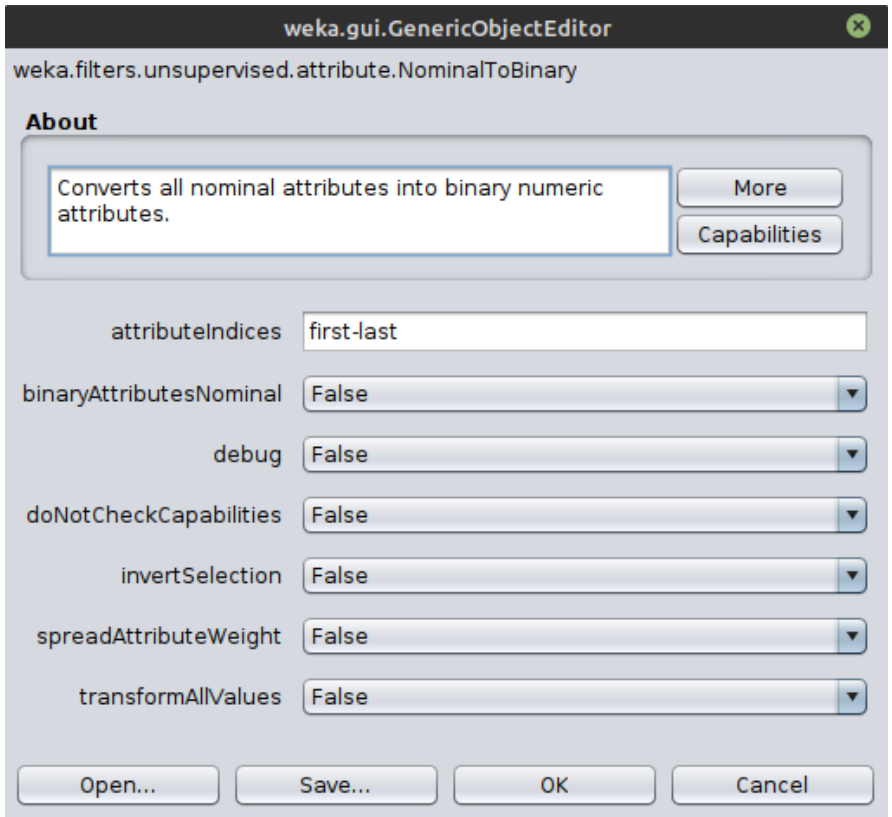
Compruebe usted mismo la aplicación del filtro usando Weka. La figura actual muestra una normalización [0,1].



Filtro Nominales a binarios

Algunos métodos como las Redes Neuronales y la regresión **solo trabajan con atributos numéricos** (al igual que otros, como algunos arboles de decisión, trabajan solo con nominales). Es necesaria una transformación:

En Weka `filters`→`unsupervised`→`attribute`→ `NominalToBinary`



Filtro Nominales a binarios

- Con la opción `BinaryAttributesNominal=False` (por defecto en Weka) todos los atributos nominales se transfoman a numericos (un nuevo atributo por cada etiqueta).
  - Los nominales que tenían **solo dos etiquetas en su lista** ({etiqueta1, etiqueta2}) se transforman a numéricos con dos valores posibles, 0 ó 1.
  - Con la opción `transformAllValues=True` los nominales que tenían solo dos etiquetas en su lista también se transforman a númericos (al igual que el resto de nominales) dando lugar a dos nuevos atributos binarizados.
- Con la opción `BinaryAttributesNominal=True` los atributos nominales **siguen siendo nominales**, pero por cada etiqueta de la lista de nominales se crea un nuevo atributo nominal con dos etiquetas posibles **{t,f}**.
  - Los nominales que tenían **solo dos etiquetas en su lista** ({etiqueta1, etiqueta2}) permanencen igual.
  - Con la opción `transformAllValues=True` los nominales que tenían solo dos etiquetas en su lista también **siguen siendo nominales** (al igual que el resto de nominales) dando lugar a dos nuevos atributos nominales con dos etiquetas posibles **{t,f}**.
- **Compruebe usted mismo la aplicación del filtro usando Weka.**

Filtro Datos anómalos (*outliers* y extremos)

- Los *outliers* son datos con valores sus **características considerablemente diferentes a la mayoría**.
  - **Ojo: Pueden ser correctos aunque sean anómalos estadísticamente** para la metodología de detección que se esté usando.
- Los valores extremos son datos mucho **más diferentes al resto que los *outliers***. Estos datos probablemente si sean datos anómalos, malas mediciones, etc.
- **Detección**:
  - Mediante distancias respecto a los demás datos (***boxplots***).
  - Gráficos de dispersión (***scatter plot***).
  - Mediante técnicas de agrupamiento que dejen fuera a patrones anómalos (**k-medias**). Busque información en la web si desea conocer su funcionamiento o metodología de trabajo.

Filtro Datos anómalos (*outliers* y extremos)

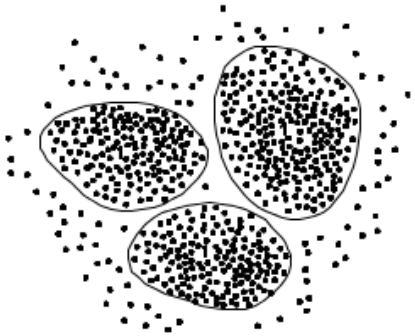


Figura: Agrupamiento o *clustering*

¿Qué hacemos si detectamos *outliers* y extremos?:

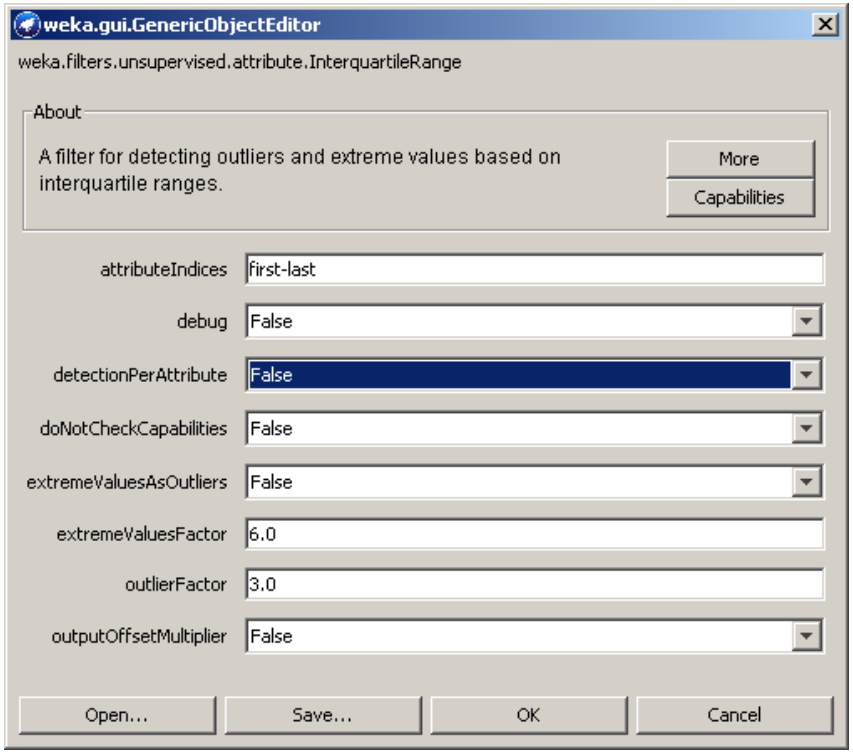
- Eliminar el patrón (más habitual).
- Ignorar: Hay modelos que son robustos a *outliers* y extremos.
- Reemplazar el *outlier* o extremo por la media del atributo u otro estadístico.

Filtro Datos anómalos (*outliers* y extremos) en Weka

`filters`→`unsupervised`→`attribute`→ `InterquartileRange`

Se basa en los **rangos interquartílicos**, como los *boxplots*.

¡Ojo!: solo se puede aplicar cuando TODOS los atributos son de **tipo numérico** (sino, aplicar antes el filtro **no supervisado *NominalToBinary***).



Filtro Datos anómalos (*outliers* y extremos) en Weka

Significado de los valores de configuración del filtro
`filters→unsupervised→attribute→ InterquartileRange`

- ***outlierFactor*:** Por defecto establecido a 3. A mayor valor (4, 5, 6, ...) hace que un patrón sea más dificil que **se considere como *outlier*** a pesar de que sea muy diferente al resto.
- ***extremeValueFactor***: Por defecto establecido a 6. A mayor valor (7, 8, 9, ...) hace que un patrón sea más dificil que **se considere como extremo o *extreme outlier*** a pesar de que sea muy diferente al resto.

Índice

Selección características

Selección de características

La selección de características implica una **reducción del tamaño de los datos**.

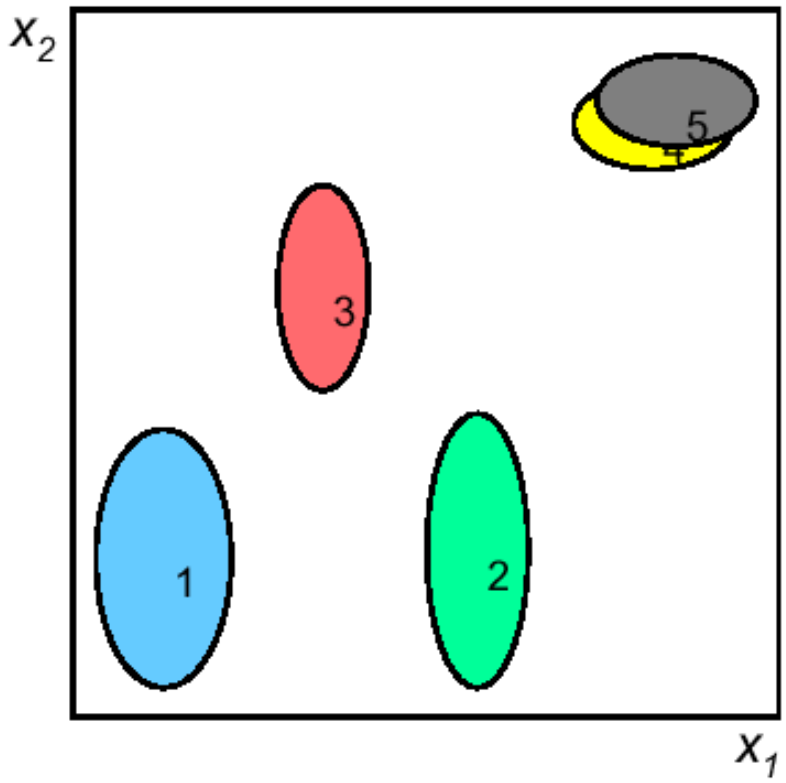
- Más atributos no significa más éxito en la clasificación: A mayor número de atributos mayor tiempo de cómputo y **probabilidad de tener sobreaprendizaje si hay atributos redundates**.
- Permite al método centrarse sólo en los atributos relevantes $\longrightarrow$ Se **mejora la calidad del modelo.**
- El modelo resultante tiene menos variables $\longrightarrow$ Se obtiene una mejor **interpretabilidad.**

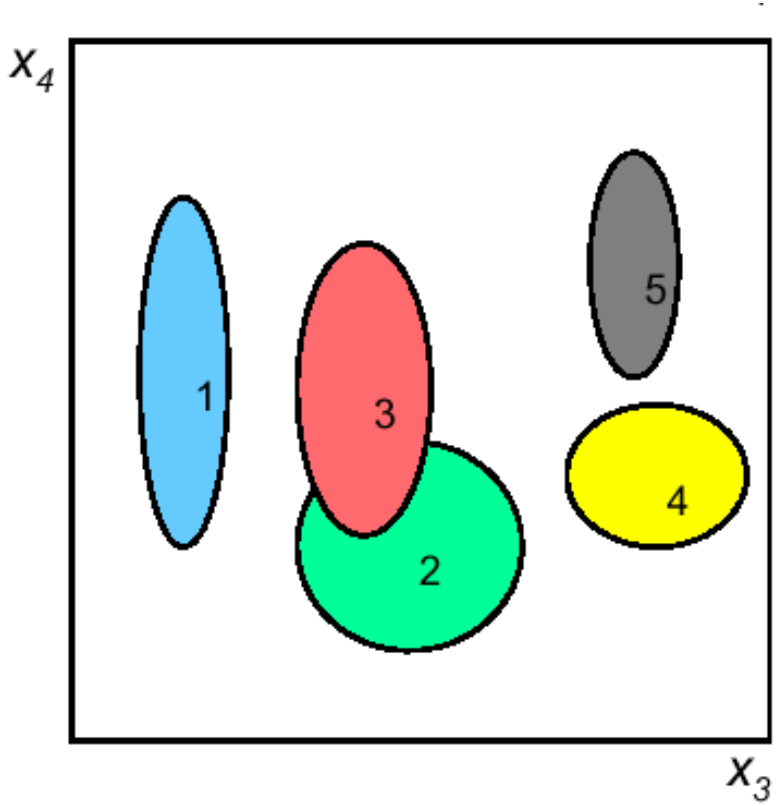
Var. 1. Var. 5 Var. 13

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
A	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
B	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
C	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
D	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
E	0	1	0	0	0	1	1	0	1	1	0	0	0	0	1	0
F	1	1	1	0	1	1	0	0	1	0	1	0	0	1	0	0

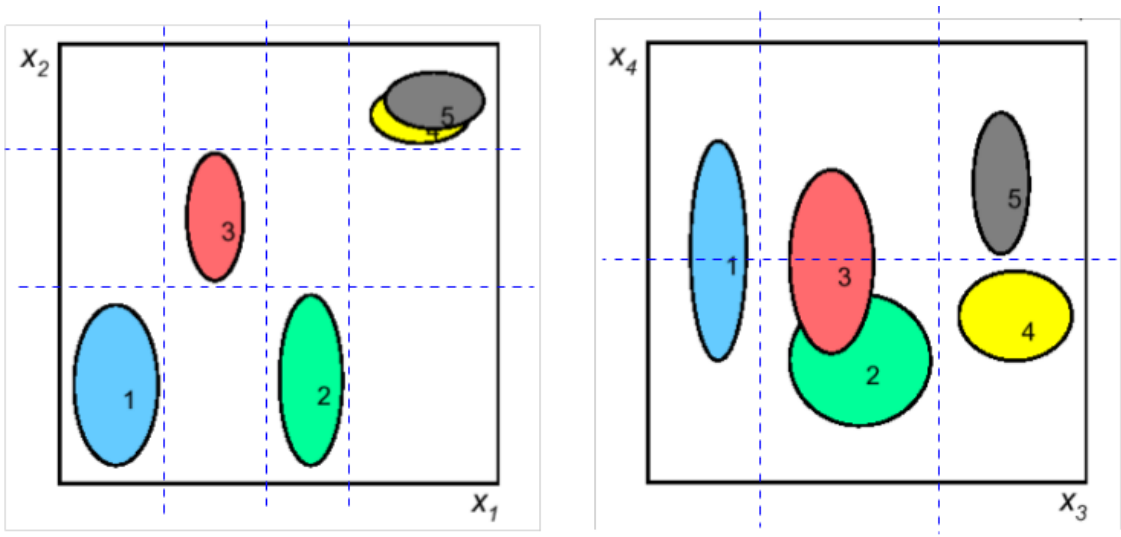
Selección de características: Ejemplo

Ejercicio: Seleccionar dos atributos para clasificar estos 5 patrones (4 dimensiones o atributos)





Selección de características: Ejemplo



$$X1 : [1, 2, 3, \{4, 5\}]$$

$$X2 : [\{1, 2\}, 3, \{4, 5\}]$$

$$X3 : [1, \{2, 3\}, \{4, 5\}]$$

$$X4 : [\{1, 2, 3\}, 4, 5]$$

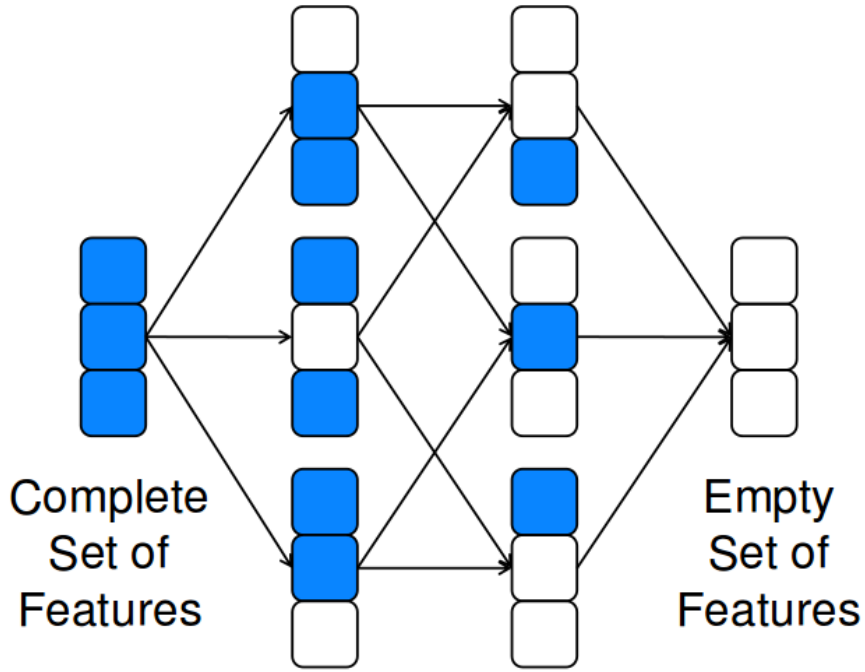
¿Elegiría $\{X_1, X_2\}$ o $\{X_1, X_3\}$?

Mejor solución: $\{X_1, X_4\}$

Selección de características

Una manera de seleccionar característica es mediante **técnicas de búsqueda que exploren el espacio de los posibles subconjuntos de características**.

- Se plantea como un problema de **búsqueda** y de **evaluación**.



Selección de características

Método de evaluación: Mediante una función de evaluación se determina la **bondad** de subconjuntos de atributos en su **discriminación sobre la clase de salida**.

Nos centraremos en tres.

- Pestaña ***Select attributes.Attribute Evaluator*** en Weka.
  - ***CfsSubsetEval***: Tiene en cuenta tanto lo **bueno que son los atributos** respecto a la salida, como lo **redundantes que son entre si**, indicando los atributos más importantes.
  - ***CorrelationAttributeEval***: Evalua lo **bueno que es cada atributo** respecto a la salida, basándose el coeficiente de **correlación de *Pearson*** respecto a la misma. Se suele utilizar más en problemas de **regresión** para correlaciones lineales. Indica la importancia de cada atributo mediante un **ranking** que va de **mayor a menor**.
    Si tenemos variables **nominales** se aconsejan pasarlas a **numéricas** (filtro **no supervisado *NominalToBinary*)**).
  - ***InfoGainAttributeEval***: Evalua lo **bueno que es cada atributo** respecto a la salida, basándose en la **ganancia de información** respecto a la misma. Se suele utilizar más en problemas de **clasificación**. Indica la importancia de cada atributo mediante un **ranking** que va de **mayor a menor**.

Selección de características

Los dos últimos métodos ***CorrelationAttributeEval*** y ***InfoGainAttributeEval*** nos pueden aportar información sobre el problema, pero hay que tener en cuenta lo siguiente:

- No seleccionan características de manera automática.
- No tienen en cuenta si dos atributos están correlados entre si.
- Para seleccionar características deberían usarse junto con métodos que nos indiquen la correlación entre los propios atributos (variables independientes), como la matriz de correlación de Pearson.
- La matriz de correlación de Pearson se puede obtener mediante la combinación ***PrincipalComponents*** + ***Ranker***.
- Con la matriz de correlación de Pearson podríamos probar a **eliminar aquellos atributos que estén muy correlados**.

Selección de características

Método de búsqueda: Mediante una metodología de búsqueda se determinan la selección de subconjuntos de atributos.

- Determinados métodos pueden provocar problemas combinatorios inabordables cuando crece el número de atributos.
- Otros métodos usan **estrategia (heurística)** para evitarlo. Es menos preciso pero también menos costoso.
- Pestaña ***Select attributes.Search Method*** en Weka (nos centraremos en dos).
  - ***Ranker***: No hace búsqueda de subconjuntos de atributos, sino que los ordena de mejor a peor según el algoritmo evaluador seleccionado. Seleccionar los k mejores.
  - ***BestFirst***: Método de búsqueda voraz de subconjuntos de atributos.
- Para saber sobre selección de características se requiere un estudio más extenso de cada método de evaluación y de búsqueda. Use la web y la bibliografía si desea ampliar conocimientos.

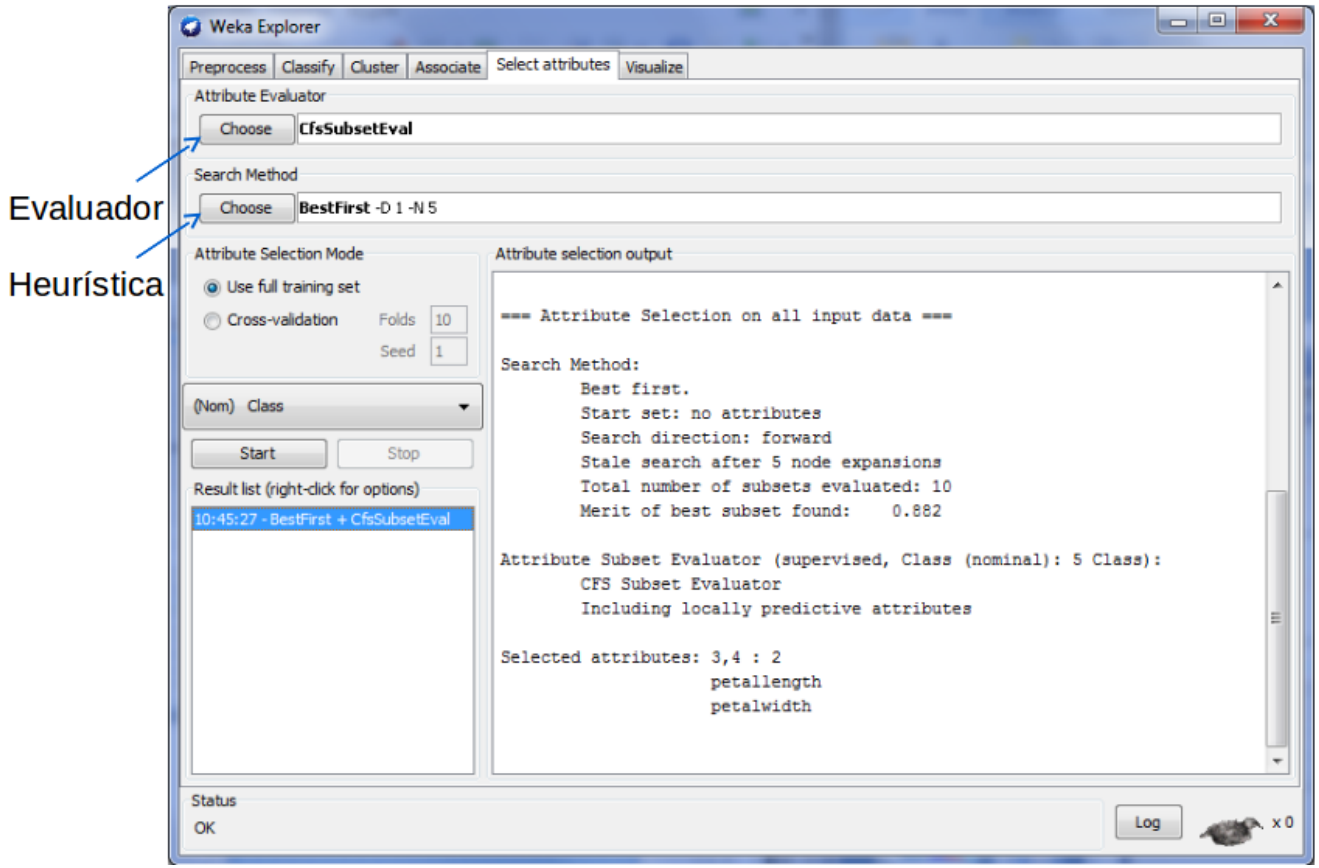
Selección de características

Combinaciones en las que nos centraremos:

- ***CfsSubsetEval*** + ***BestFirst***: Selecciona un subconjunto de atributos del total que podrían ser representativos de nuestro problema.
- ***CorrelationAttributeEval*** + ***Ranker***: Se basa en hacer ranking de los atributos usando el coeficiente de correlación de Pearson. Se nos ordenan de mejor a peor. Podríamos elegir los de mayor ranking y hacer pruebas para ir comprobando rendimientos (ojo, no indican la correlación entre los atributos independientes).
- ***InfoGainAttributeEval*** + ***Ranker***: Se basa en hacer ranking de los atributos usando la ganancia de información. Se nos ordenan de mejor a peor. Podríamos elegir los de mayor ranking y hacer pruebas para ir comprobando rendimientos (ojo, no indican la correlación entre los atributos independientes).

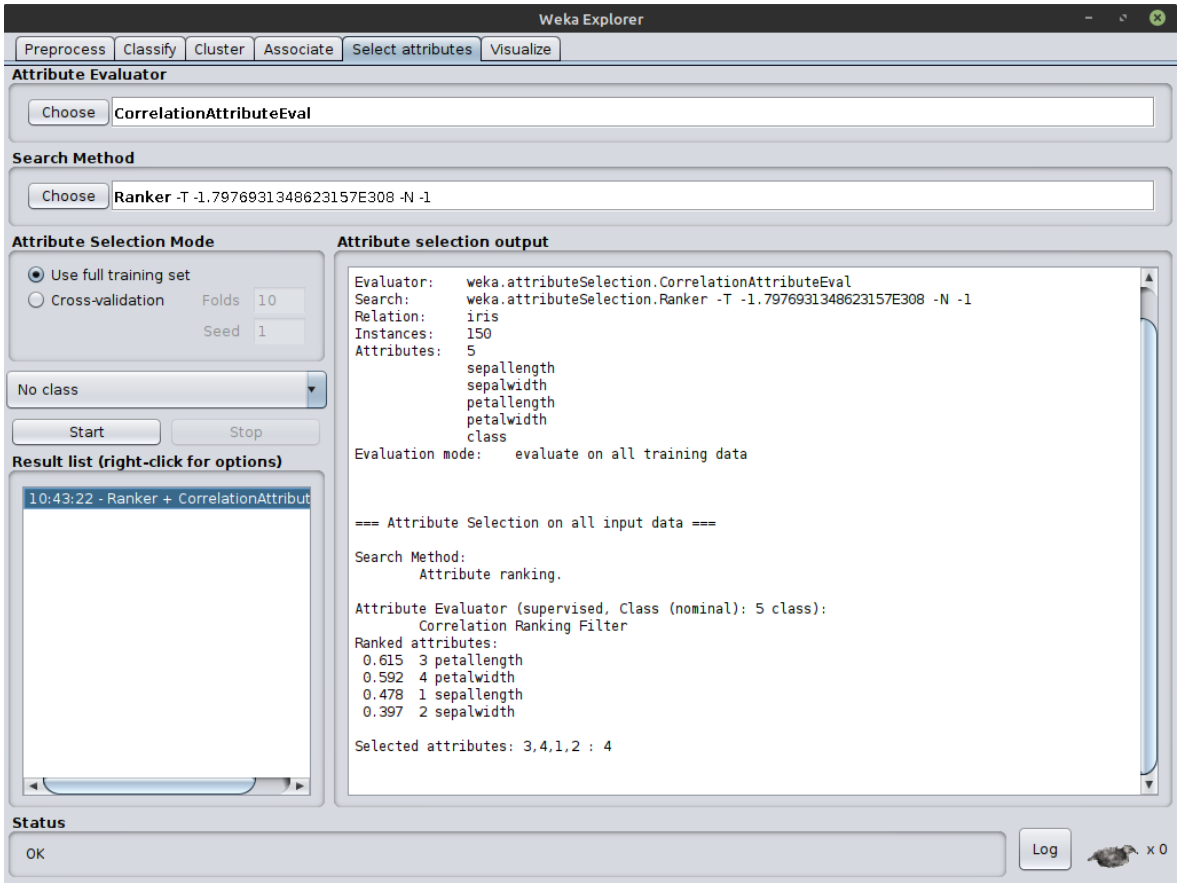
Selección de características en Weka

Selección de carácterísticas ***CfsSubsetEval*** + ***BestFirst*** en Weka en el entorno **Explorer->Select attributes**.



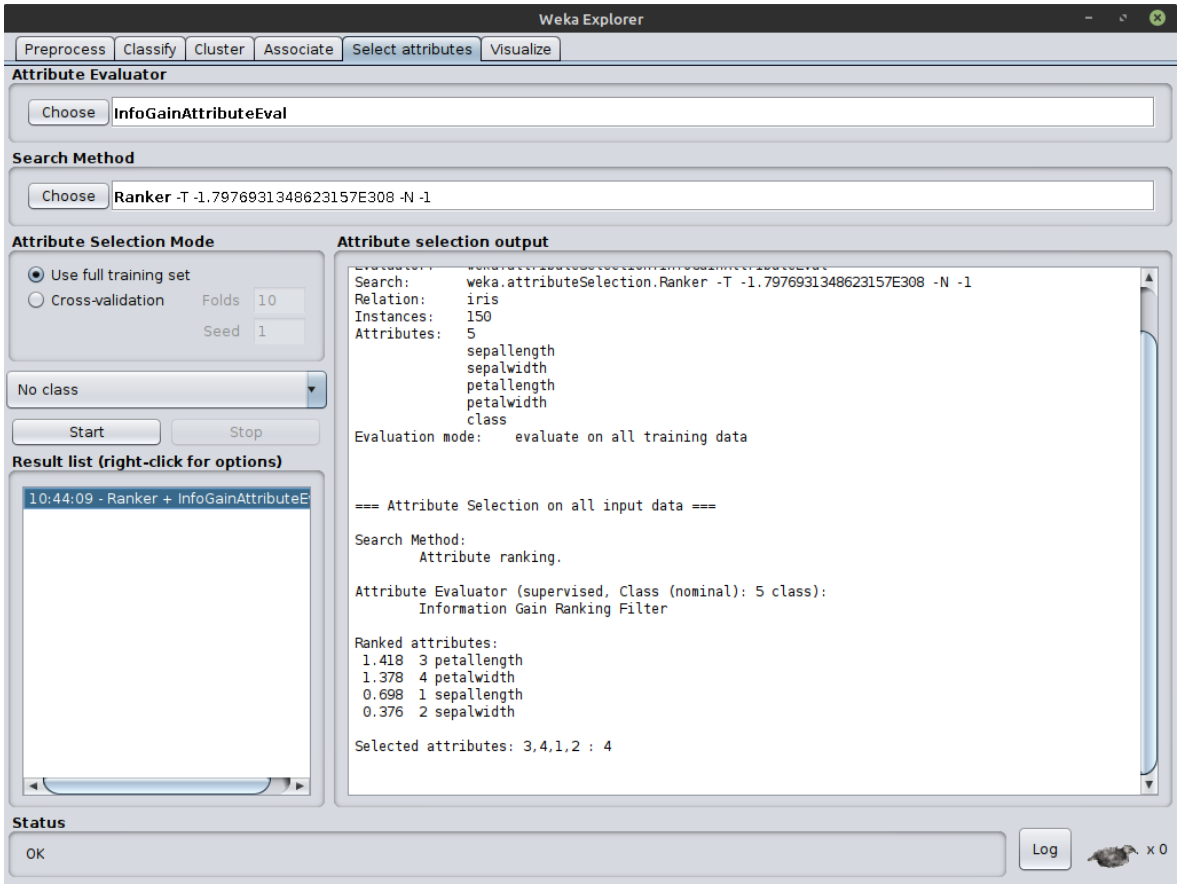
Selección de características en Weka

Selección de carácterísticas ***CorrelationAttributeEval*** + ***Ranker*** en Weka en el entorno **Explorer->Select attributes**.



Selección de características en Weka

Selección de carácterísticas ***InfoGainAttributeEval*** + ***Ranker*** en Weka en el entorno **Explorer->Select attributes**.



Índice

Selección instancias

Selección de instancias

La **selección de instancias** se emplea para:

- **Reducir el número de patrones** de las clases más numerosas (*under-sampling*).
- **Incrementar el número de patrones** en las clases minoritarias, introduciendo patrones sintéticos (*over-sampling*).

Ventajas:

- Acelera el proceso de entrenamiento.
- Mejor exactitud del modelo.
- Modelos más simples e interpretables.
- Reducción del ruido y patrones redundantes.
- Facilita el aprendizaje con grandes volúmenes de datos.

Selección de instancias

Ejemplo de **selección de instancias** representativas para **reducir el conjunto de datos**:

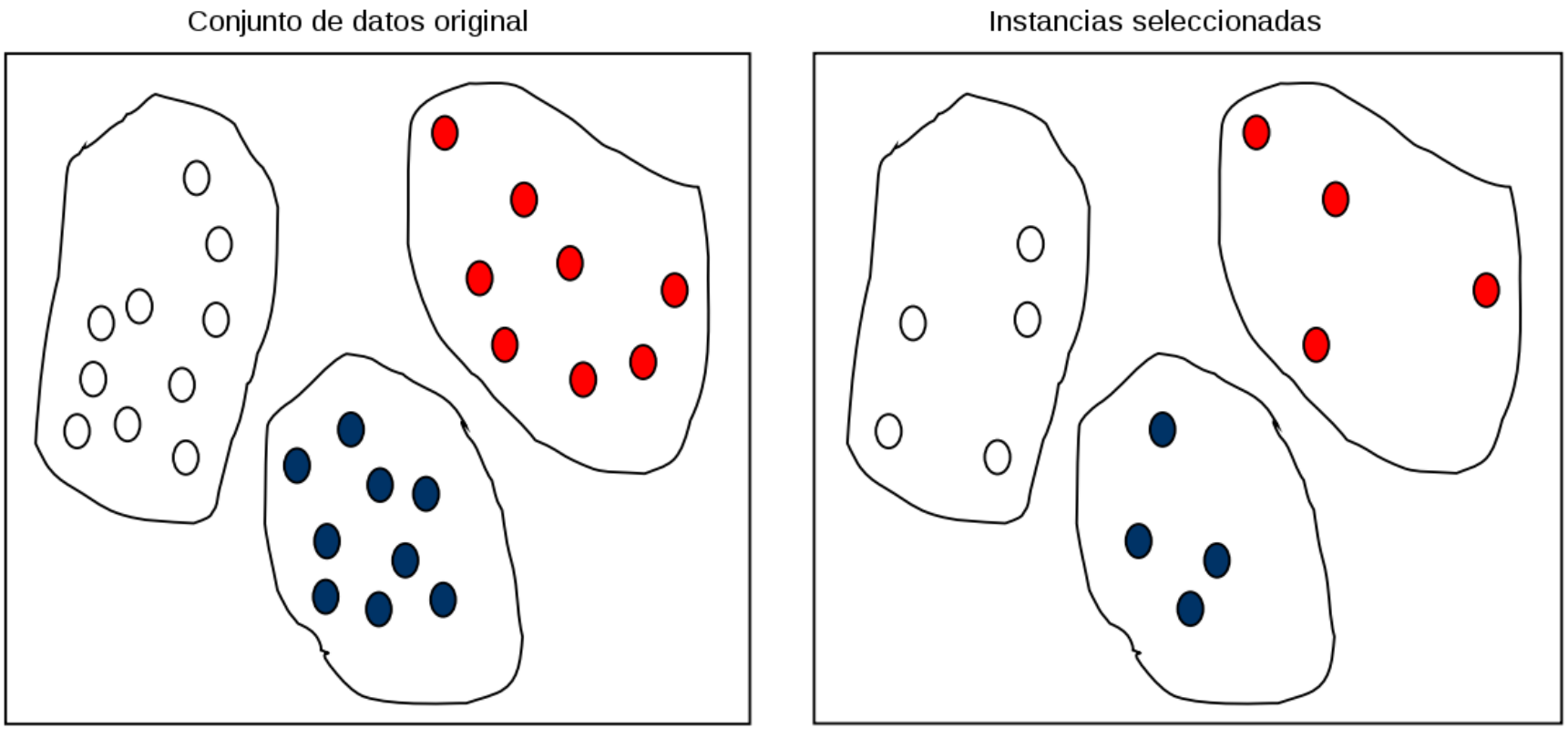
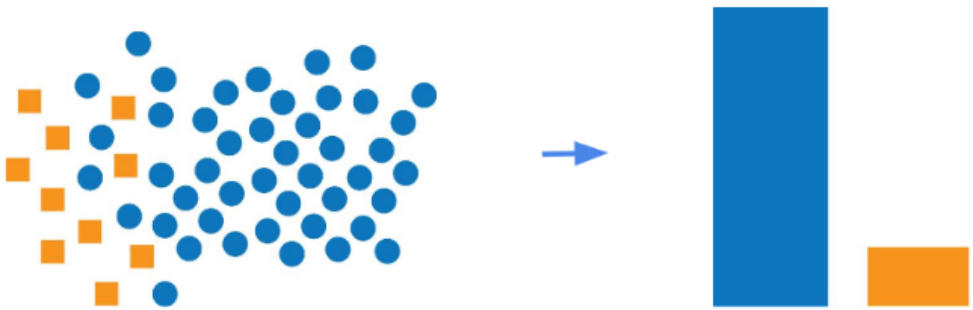


Figura: Ejemplo de selección de instancias.

Desbalanceo de los datos

En algunos casos las clases pueden tener una **frecuencia extremadamente desigual**:

- Ej. en diagnosis médica: 90 % saludables, 10 % enfermos.
- Ej. en seguridad: más del 99.99 % de los ciudadanos no son terroristas.
- ¡Mirar porcentaje de clasificación por clase!



Cuidado...

- **Clasificador inutil**: Pudiera ser incluso aquel con un [90 %-99.99 %] correcto...
- Ej: 100 patrones (90 clase A y 10 clase B). Reconoce a todos los de la clase A, pero **ninguno de la clase B**.

Desbalanceo de los datos

¿Qué hacer?

- **Sobremuestreo** (*over-sampling*): Generar nuevos patrones sintéticos de la clase minoritaria.
- **Inframuestreo** (*under-sampling*): Seleccionar una muestra de patrones de la clase mayoritaria.

Selección de instancias y desbalanceo en Weka

Filtros No supervisados más importantes (**no estratifican**):

- **RemoveDuplicates**: Eliminar patrones duplicados (*under-sampling*).
- **RemovePercentage**: Elimina (sin estratificar) aleatoriamente un porcentaje de patrones de la base de datos (*under-sampling*).
- **RemoveRange**: Eliminar patrones según su índice en la base de datos, p.ej. 1–500 (*under-sampling*).
- **RemoveWithValues**: Eliminar patrones que tienen unos valores concretos para determinados atributos (*under-sampling*).
- **Resample (para balanceo de datos)**: Aumenta/disminuye los patrones seleccionando (sin estratificar) aleatoriamente algunos de ellos (con reemplazamiento - se puede volver a elegir - o no) (*under-sampling-over-sampling*).

Selección de instancias y desbalanceo en Weka

Filtros Supervisados más importantes (**estratifican**):

- **Resample (para balanceo de datos): Como en el caso No Supervisado**, selecciona aleatoriamente algunos patrones de la base de datos (con reemplazamiento o no), pero manteniendo una determinada proporción de patrones por clase, es decir, es estratificado. (*under-sampling-over-sampling*)
- **SpreadSubsample (para balanceo de datos)**: Selecciona aleatoriamente algunos patrones de la base de datos de manera estratificada, eliminandolos de forman que el número de patrones por clase se ajuste a la más pequeña (*under-sampling*).
  - Parámetro *distributionSpread* $= 1.0$

Índice

¿Cómo analizar nuestros datos?

Realizar una **análisis preliminar** para entender sus **características** específicas:

- Hacer **más fácil el preprocesado y el aprendizaje**.
- En ocasiones necesidad de un **experto**.
- **Combinación de inspección humana y automática**: Utilizar técnicas automatizadas para identificar casos "extraños" y dejar al experto trabajar sobre estos datos.
- El no detectar **casos anómalos** (o en general, el obviar un análisis previo) puede llevar a muchos problemas *a posteriori*, por ejemplo al normalizar.
- **Balanceo de clases**: una o varias clases mucho menos frecuente.
  - Métodos de remuestreo de la clase minoritaria.
  - Selección de instancias de la clase mayoritaria.

¿Preguntas?

¡Gracias!