

Analítica Social de los Datos y Herramientas para Analítica de Datos

Jose Aguilar
CEMISID, Escuela de Sistemas
Facultad de Ingeniería
Universidad de Los Andes
Mérida, Venezuela

Analítica Social de los Datos

El análisis sociales de datos es un estilo de análisis en el que es considerado las personas trabajan en un contexto social, de colaboración, para darle sentido a los datos.

El análisis de datos sociales se compone de dos partes:

- Los datos generados a partir de los sitios de redes sociales (o a través de aplicaciones sociales), y
- Análisis de los datos, en tiempo real (o casi en tiempo real), en los cuales se incluyen medidas para entender, y apropiadamente pesar, factores como la influencia, alcance y relevancia del contexto de los datos, y se incluye el horizonte de tiempo.

La analítica social de datos implica el análisis de las redes sociales, con el fin de comprender la percepción y actividad social que se inserta dentro de los datos.
Por ejemplo, analizar

- Interacciones directas con los demás (por ejemplo, mensajería, etc.),
- Uso de plataformas (por ejemplo, búsquedas , etiquetados).

Analítica Social de los Datos

En un sistema de análisis social de datos:

- los usuarios almacenan conjuntos de datos y crean representaciones visuales.
- Los conjuntos de datos y representaciones son accesibles para otros usuarios de la red o sitio web.
- Los usuarios pueden crear nuevas e interesantes representaciones, así como comentarios asociados a las.
- Se pueden usar un blogs y wikis para conducir esta inteligencia social.

Por lo general, podemos recuperar los datos sociales de una variedad de redes sociales, como Twitter, Facebook, nos sentimos bien, Wikipedia, etc.

- Dado que la mayoría de las redes sociales nos proporcionan la API, no es difícil para nosotros para recuperar los datos.
- El uso de API para obtener los datos es como enviar una solicitud a la página web y luego el sitio web envía los datos solicitados en forma de XML o en forma de JSON.

Analítica Social de los Datos

Métodos de análisis

- En la mayoría de los casos, queremos averiguar las relaciones entre los datos sociales y otro evento, o queremos predecir algunos eventos.
- Con el fin de lograr estos objetivos, necesitamos los métodos adecuados para hacer los análisis: métodos estadísticos, de aprendizaje de máquinas o de minería de datos.

Cuando se habla de análisis de datos sociales, hay una serie de factores que es importante tener en cuenta:

- **Análisis de datos sofisticados:** El análisis de datos sociales debe tomar en consideración una serie de factores (contexto, el contenido, el sentimiento) para proporcionar información adicional.
- **La consideración del tiempo:** Lo más relevante de un día (o incluso una hora) puede no ser en la siguiente. Ser capaz de ejecutar con rapidez el análisis es imperativo.
- **Análisis de la influencia:** la comprensión del impacto potencial de individuos específicos puede ser clave en la comprensión de cómo los mensajes podrían estar resonando. No se trata sólo de la cantidad, también tiene mucho que ver con la calidad.
- **Análisis de las Redes:** los datos social migran, crece n(o mueren) en base a cómo los datos se propagan a través de la red. Es como una actividad viral, que se inicia y se propaga.

Analítica Social de los Datos

Por ejemplo, Analítica Social del aprendizaje (SLA) centra la atención en los elementos de aprendizaje que son relevantes cuando se está aprendiendo en una cultura participativa.

- Ella busca hacer visible, comportamientos y patrones en el ambiente de aprendizaje.
- Los alumnos no son solitarios, y hacen actividades sociales, ya sea interactuando con los demás, o dejando trazas en las plataformas de sus actividades
- Los logros individuales, se transmiten a través de la interacción y la colaboración.

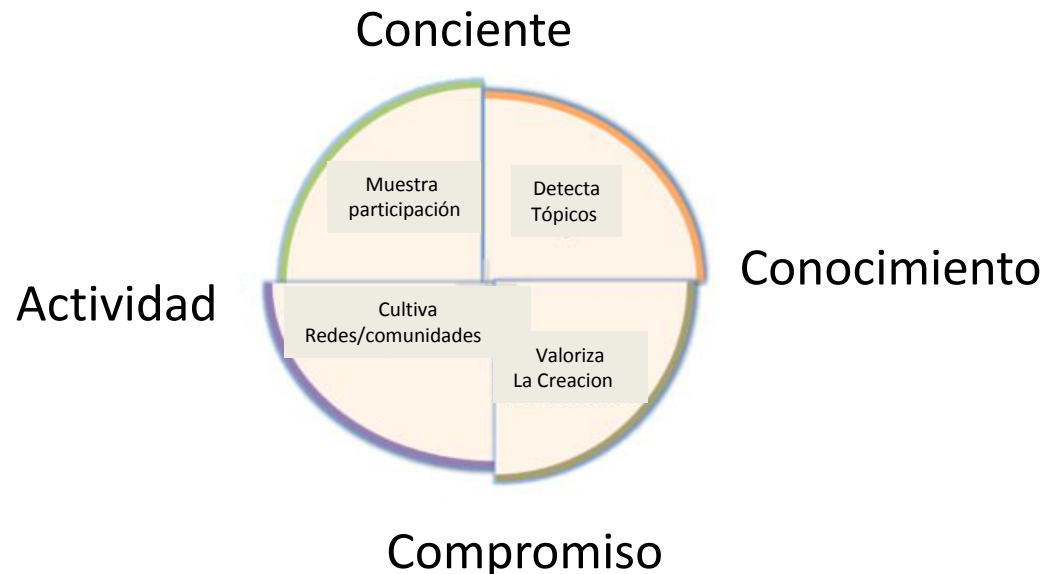
Los aprendices construyen el conocimiento, influenciados por sus objetivos, los sentimientos y las relaciones, las cuales cambian según el contexto.

- El éxito es una combinación de los conocimientos, habilidades individuales, el medio ambiente, el uso de herramientas, y la capacidad de trabajar juntos.
- La comprensión de aprendizaje obliga a prestar atención a los procesos del grupo en la construcción del conocimiento.
- La atención debe centrarse no sólo en los alumnos, sino también en las herramientas y contextos.

Analítica Social de los Datos

El análisis de aprendizaje desde una perspectiva social destaca un tipo de analítica que es empleada para darle sentido a la actividad de aprendizaje en un entorno social.

- Se deben identificar los comportamientos sociales y los patrones que significan un proceso eficaz, en entornos de aprendizaje emergente.
- Debe ser aplicable a diferentes escalas: redes nacionales e internacionales, pequeños grupos y alumnos individuales.



Analítica Social de los Datos

Algunos horizontes nuevos de análisis que permite SLA.

Necesitamos desarrollar nuevos conjuntos de análisis que se pueden utilizar para apoyar el aprendizaje y la enseñanza, considerando:

- Factores tecnológicos
- El cambio a "libre" y "abierto"
- La demanda de habilidades para la era del conocimiento
- La innovación requerida en el aprendizaje social
- Los nuevos desafíos de las instituciones educativas.

Analítica Social del aprendizaje

Factores tecnológicos

Los cambios en la tecnología no necesariamente implican cambios en la pedagogía.

Aquellos que ven la educación como la transferencia de información, utilizará medios interactivos para el almacenamiento, y la transmisión de la información;

Los que buscan el cambio conceptual explotan sus cualidades interactivas.

- Buscan con las tecnologías conocimientos que se basen en conjuntos de datos, para comprender el movimiento hacia el aprendizaje social en línea y analizar los procesos asociados.
- Conceptualmente, es introducir la idea de conocimiento distribuido, colaboración o innovación.

Analítica Social del aprendizaje

El cambio a lo libre y abierto

El Internet hace posibles modelos completamente nuevos de generación de ingresos

- Compañías en línea son capaces de ofrecer servicios de calidad de forma gratuita, produciendo una enorme variedad de herramientas y fuentes de contenido libre hospedado 'en la nube'.
- Los Recursos Abiertos Educativos (REA) son un vehículo de gran alcance, que hacen materiales de aprendizaje de alta calidad disponibles, no sólo de forma gratuita, sino también en formatos que promueven la re-mezcla,

Esto es amplificado por los esfuerzos para hacer que los datos sean abiertos al procesamiento de la máquina, así como a la interpretación humana.

- Ejemplos son las comunidades de Linked Data y de Web Semántic, el Gobierno Abierto, Ciencia 2.0 y 2.0 de la Salud
- La provisión de contenidos, las herramientas, etc., pueden esperarse que sean de forma gratuita, mientras que los estudiantes paguen por otros servicios: tutorías personalizadas, orientación profesional, acreditación

Analítica Social del aprendizaje

La demanda de habilidades en la era del conocimiento

- Las habilidades de la era del conocimiento están relacionadas no sólo a un imperativo económico, sino a un deseo y al derecho a saber, una extensión de las oportunidades educativas,
- Además, es un aprendizaje que debe desarrollar habilidades y competencias.
- Esto implica la necesidad de apoyar el desarrollo de la creatividad y la curiosidad, habilidades de colaboración

La innovación requiere aprendizaje social

El aprendizaje social es la única manera en que las organizaciones pueden hacer frente en el cambiante mundo de hoy.

Nuestra comprensión de los contenidos se construye socialmente a través de conversaciones sobre ese contenido ya través de interacciones

Analítica Social del aprendizaje

Una característica importante del paradigma de la Web 2.0 es el grado de personalización que los usuarios finales esperan ahora.

Centrado en el aprendiz, no en el usuario

- Lo que quiero no es necesariamente lo que necesito, porque mi comprensión de la materia, es incompleta
- El trabajo es enseñar a la gente a pensar, y que requiere un aprendizaje más profundo, dejando al lado la seguridad cognitiva y emocional donde las hipótesis son meramente reforzadas.
- Esto implica un desafío para llevar a los alumnos fuera de su zona de confort, lo que subraya la importancia de la afirmación y el aliento al alumno para darle seguridad de salir.

Analítica Social del aprendizaje

Dos analíticas inherentemente sociales, y tres de análisis socializados:

- ***Inherentemente sociales tienen sentido en un contexto colectivo:***
 - Analítica de Redes Sociales- relaciones interpersonales en plataformas sociales.
 - Analítica del Discurso- el lenguaje es una herramienta fundamental para la construcción del conocimiento.
- ***Análisis socializados: son analíticas personales, pertinentes socialmente en un contexto colectivo:***
 - Analítica del contenido -es una de las características definitorias de la Web 2.0
 - Analítica de la Disposición- Motivación para aprender (centro de la innovación)
 - Analítica del Contexto- clave en entornos móviles, aprendizaje informal, etc.

Analítica Social del aprendizaje

Análisis de redes sociales

El aprendizaje en red implica el uso de las TIC para promover conexiones entre un alumno y otros alumnos, entre alumnos y tutores, y entre las comunidades de aprendizaje y recursos de aprendizaje.

- Estas redes se componen de actores (tanto de personas como de recursos) y las relaciones entre ellos.
- Actores con una relación entre ellos se dice que están vinculados, y estos lazos pueden ser clasificados como fuertes o débiles, dependiendo de su frecuencia, calidad o importancia.
- El análisis de redes sociales investiga los procesos en la red, las propiedades de las relaciones, los roles y la formación de la red, para entender cómo la gente desarrolla y mantiene estas relaciones para apoyar el aprendizaje

Ejemplo implementación de análisis de redes sociales de aprendizaje es SNAPP (Social Networks Adapting Pedagogical Practice), una herramienta de visualización de la red que incluye:

- Identificar a los estudiantes desconectados
- Identificar a los agentes de información claves dentro de una clase
- Indicar el grado en que una comunidad de aprendizaje se está desarrollando

Analítica Social del aprendizaje

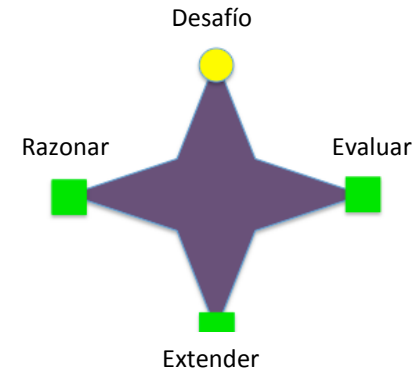
La analítica del discurso aprendizaje social

El análisis del discurso es el término colectivo para el análisis de la serie de eventos comunicativos.

- Algunos de estos enfoques se centran en la cara-a-cara o interacción oral.
- Otros proporcionan nuevas formas de entender las grandes cantidades de texto generados en los cursos en línea y conferencias.
- Otros lo usan para comprender la relación entre las dimensiones interactivas, cognitivas, la interacción en línea, y el seguimiento de los intercambios (IRF).
- En las discusiones asíncronas, muestra cómo los grupos crean y mantienen la comunidad y la coherencia a través del uso de dispositivos discursivos.

Dos ejemplos de análisis de discurso son:

- *El diálogo exploratorio*, que incluye descubrir en los foros desafíos ('pero si', 'no creo'), evaluaciones ("Buen punto", "importante",), el razonamiento('quiere decir eso', 'mi entendimiento es') y las extensiones ('Siguiendo paso', 'se refiere a'), por las palabras y frases claves, y
- *El desarrollo Mapas Conceptuales* para la deliberación y argumentación.



Analítica Social del aprendizaje

Analítica de la disposición para el aprendizaje

- Es la única que se origina del campo de la investigación educativa
- Es definida por un modelo de siete dimensiones
- Mezcla la experiencia, la motivación y la capacidad de una persona para toda la vida aprender e identificar a las oportunidades de aprendizaje
- Disposición para el aprendizaje no es "estilo de aprendizaje": las disposiciones de aprendizaje varían según el contexto,

El compromiso en el aprendizaje es una combinación compleja de las identidades de los alumnos, disposiciones, valores, actitudes y habilidades.

Cuando éstos son frágiles, los estudiantes luchan por alcanzar su potencial en las evaluaciones convencionales, y no están preparados para la novedad, la complejidad de los retos, y las muchas otras esferas de la vida que requieren cualidades como la capacidad de recuperación, el pensamiento crítico y habilidades de colaboración.

Las disposiciones de aprendizaje pueden ser modelados como un constructo multidimensional, llamado **Aprendizaje de energía**, actualmente evaluados a través de un cuestionario web llamado ELLI

ELLI genera una visualización analítica que se utiliza para apoyar la auto-reflexión y cambio



Analítica Social del aprendizaje

Análisis de contenidos de aprendizaje social

- Originario de los campos relacionados con los sistemas de recomendación y recuperación de la información
- Es la variedad de métodos automatizados que se pueden utilizar para examinar, indizar y filtrar los recursos multimedia en línea, con la intención de guiar a los estudiantes a través del océano de los recursos potenciales disponibles para ellos.
- En combinación con el análisis de contexto de aprendizaje o con los términos de búsqueda definidos, análisis de contenidos se pueden utilizar para proporcionar recomendaciones de recursos que se adaptan bien a las necesidades de un individuo o de las necesidades de un grupo de alumnos.
- La recuperación de información representa las técnicas para la indexación automática y el filtrado de contenido, ya sea textual o multimedia (por ejemplo, imágenes, vídeo o música).
- En conjunto, estos elementos se pueden utilizar para proporcionar nuevos métodos para sugerir, de navegación o búsqueda de medios educativos.

Analítica Social del aprendizaje

Análisis de contenidos de aprendizaje social

Los componentes básicos de análisis de contenidos de aprendizaje social, ofrecen la opción de ver las palabras clave, enlaces directos o imágenes relacionadas con la página web abierta. Esta información sobre las imágenes se puede combinar con la búsqueda de similitud visual, para identificar y recomendar otros recursos que hacen uso de estas imágenes,

The screenshot shows a news website with a red header bar containing the text "NEWS SCIENCE & ENVIRONMENT". Below the header is a navigation bar with links: Home, World, UK, England, N. Ireland, Scotland, Wales, Business, Politics, Health, Education, Sci/Environment, Technology, and Entertainment & Arts. The date "5 October 2011" and time "Last updated at 16:36" are displayed. The main article is titled "Nobel win for crystal discovery" and features a large image of a colorful, fractal-like crystal structure. To the right of the image is a "Watch/Listen" section with a play button icon. Below the main article, there are three smaller articles: "Cull 'no help' to Tasmanian devil" with a photo of a devil, "New push to eradicate guinea worm" with a photo of a worm, and "NZ condemns Japan's whaling plans" with a photo of a whale. A sidebar on the right, titled "Images Keywords Links", lists several image-related suggestions: "Guinea worm...", "Jonathan Amos...", "Richard Black...", "Hurricane Isabel...", "Astronomer Mark Thompson...", "A yeti creature...", "Hand on a remote control...", "Cave of Rouffignac...", "Insects mating on branch...", "Euclid (Esa)...", "American bullfight...", "Leo and Hope...", "Daspletosaurus", and "Click: 04/10/2011...". A small image of a dinosaur is shown next to the "Daspletosaurus" link. The bottom of the page features a blue navigation bar with icons for Home, Apps, Learning paths, Bookmarks, and Search, along with a social media share icon.

Analítica Social del aprendizaje

Análisis del contexto de aprendizaje

- El aprendizaje se pueden aplicar a una amplia variedad de contextos que se extiende mucho más allá de los sistemas institucionales.
- Pueden ser utilizados en contextos formales como escuelas, colegios y universidades, en contextos informales en las que los estudiantes eligen tanto el proceso como el objetivo de su aprendizaje.
- En algunos casos, los estudiantes están en entornos síncronos, sobre la base de que los participantes son co-presente en el tiempo, o se encuentran en entornos asíncronos, donde se supone que van a participar en diferentes momentos.
- Puede ocurrir en una red, en un grupo de afinidad, en las comunidades de investigación, comunidades de interés, etc.

Bajo el título análisis del "contexto" se agrupan las diversas herramientas analíticas que exponen, hacen uso de, o tratan de comprender estos contextos en relación con el aprendizaje.

Analítica Social del aprendizaje

Análisis del contexto de aprendizaje

Se puede caracterizar el contexto de una entidad (por ejemplo, un estudiante) en función de cinco categorías distintas:

- **contexto Individualidad**, incluye información sobre la entidad dentro del contexto. En el caso de los estudiantes, esto podría incluir su idioma, su comportamiento, sus preferencias y sus objetivos
- **contexto del Tiempo**, que incluye puntos en los tiempos, y oscila historias, por lo que puede tener en cuenta el flujo de trabajo, cursos de larga duración y las historias de interacción
- **contexto de Localización**, puede incluir la ubicación absoluta, ubicación en relación con personas o recursos, o la ubicación virtual (dirección IP)
- **contexto de la Actividad**, se refiere a los objetivos, tareas y acciones
- **contexto de la Relaciones**, capta las relaciones de la entidad con otras entidades, por ejemplo, entre los alumnos, los profesores y los recursos.

Analítica Social del aprendizaje

- Si las herramientas y los datos de SLA se colocan en las manos de los estudiantes, el equilibrio de poder cambia significativamente.
- Si la analítica están llamando la atención a los alumnos para su desarrollo como, estudiantes, intrínsecamente los motiva conscientes de sí mismos, moviendo en la dirección opuesta a depender pasivamente en la institución o plataforma para decirles cómo lo están haciendo y qué hacer a continuación.
- Si la analítica se centran en proporcionar retroalimentación formativa para mejorar el proceso de aprendizaje, en lugar de hacer juicios automatizados sobre el dominio en un tema determinado, puede haber un menor número de preocupaciones en torno a la eliminación de los tutores humanos del circuito de retroalimentación.

"Libre y abierto" es una expectativa fundamental y dinámico dentro de aprendizaje social en línea.

- Muchas herramientas de SLA están disponibles en las versiones de código abierto.
- Se convierte en normal que los patrones de SLA y los datos sean abiertos.
- Alumnos están dispuestos a pagar por características más potentes, una vez que las herramientas de mayor éxito se han ganado su derecho a cobrar.

Analítica Social del aprendizaje

Las instituciones que carecen de la infraestructura necesaria para SLA pedirá los servicios de SLA en la computación "en la nube",

- Los alumnos individuales o comunidades que necesitan este tipo de servicios también utilizarán estos servicios.
- instituciones educativas explotarán su experiencia pedagógica para proporcionar servicios de consultoría de SLA.

Aspiraciones de todas las culturas han ido cambiando hacia un creciente deseo de participación y libre expresión. La web social es una expresión de este cambio,

- Las herramientas SLA se convierten en una parte importante del sentido de la identidad de los individuos, y su capacidad para evidenciar sus habilidades. Por ejemplo, podríamos ver Placas tales como: "yo puedo llevar debates complejos ", "yo puedo guiar a los estudiantes en la construcción de su creatividad."

Analítica Social del aprendizaje

La innovación en entornos complejos y turbulentos requiere infraestructuras de creación de conocimiento y negociación sociales construidas sobre las relaciones de calidad y conversaciones, en orden a trabajar juntos para resolver "problemas perversos".

- SLA se convierten en una parte integral del juego de herramientas,

El papel de las instituciones educativas está cambiando. Se están moviendo cada vez más para proporcionar apoyo personalizado para aprender a pensar profundamente, y aprender cómo ser un miembro efectivo de las comunidades.

- Las instituciones educativas ya no son la única opción para evidenciar el aprendizaje avanzado.
- Analítica se convierten en una nueva forma de pruebas de confianza, que se genera a partir de conjuntos de datos verificables, públicas, privadas, que no podrían haber sido fabricados.

Weka



Weka

(Data Mining Tool)

- Colección de algoritmos open source de aprendizaje automático
 - pre-procesamiento
 - clasificadores
 - clustering
 - Reglas de asociación
- Principales características:
 - Herramientas de pre-procesamiento, algoritmos de aprendizaje y métodos de evaluación
 - Interfaz Grafica (incl. visualización de datos)
 - Ambiente para comparar algoritmos de aprendizaje

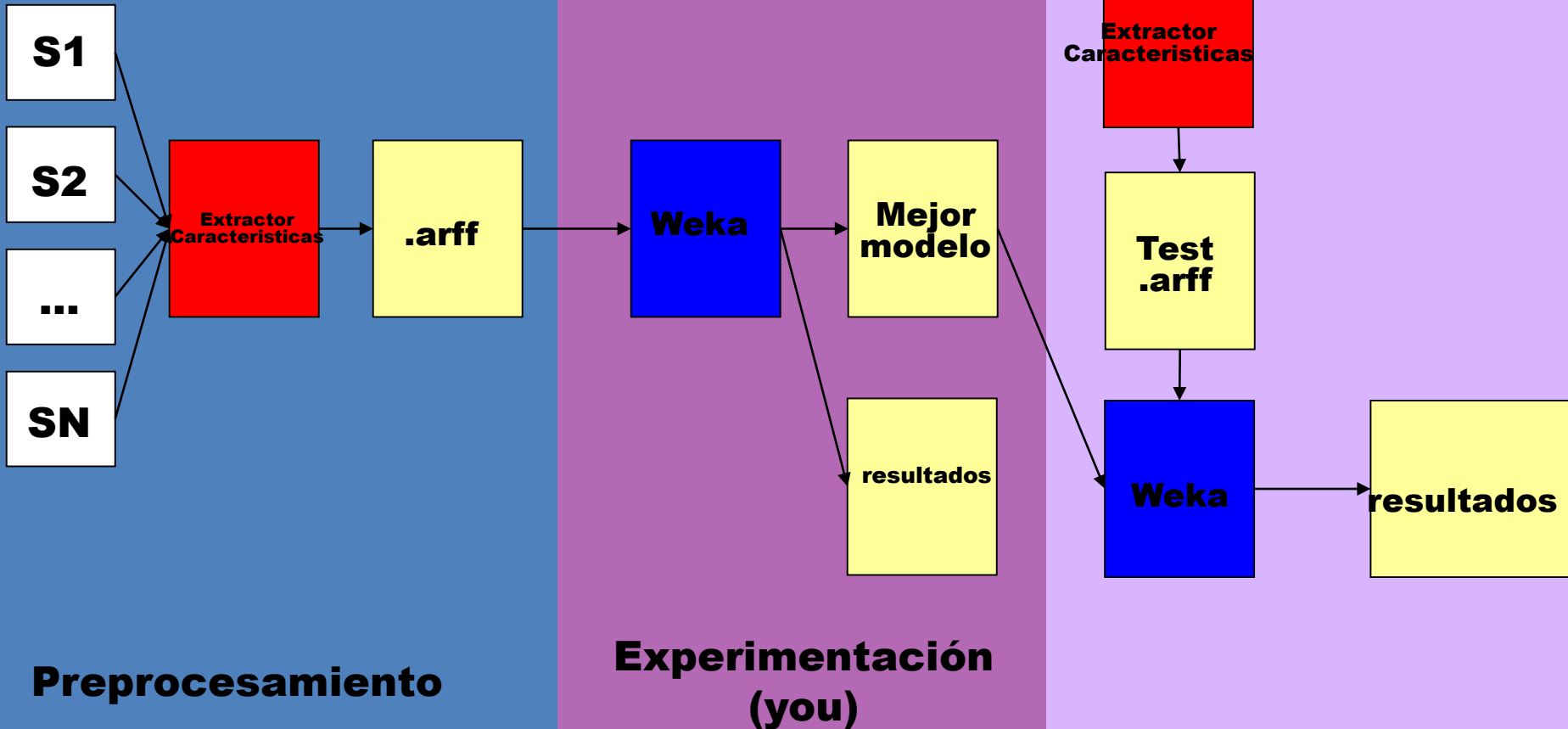


Weka

(Data Mining Tool)

- Weka es una herramienta de minería de datos de código abierto desarrollado en Java.
- Se utiliza para la investigación, la educación, y las aplicaciones.
- Basado en Java <http://www.cs.waikato.ac.nz/ml/weka/>
- Se puede ejecutar en Windows, Linux y Mac.
- 3 modos of operation
 - GUI: aplicar directamente a un conjunto de datos (con interfaz gráfica de usuario)
 - Linea de comando
 - Java API: llamados desde su propio código Java (usando biblioteca Weka de Java).

Flujo de trabajo con Weka



Weka

- Entrada de datos a Weka (Input)
 - Usa archivos planos
 - Formatos “.arff”, C4.5, CSV,
 - Datos pueden ser leídos desde URL o BD SQL (usando JDBC)
- Weka para Data Mining
- Salida desde Weka (Output)

Entrada de datos a Weka (Input)

- El más popular formato is “arff” (“arff” es la extensión del nombre del archivo).

FILE FORMAT

@relation RELATION_NAME

@attribute ATTRIBUTE_NAME ATTRIBUTE_TYPR

@attribute ATTRIBUTE_NAME ATTRIBUTE_TYPR

@attribute ATTRIBUTE_NAME ATTRIBUTE_TYPR

@attribute ATTRIBUTE_NAME ATTRIBUTE_TYPR

@data

DATAROW1

DATAROW2

DATAROW3

Entrada de datos a Weka (Input)

archivo “arff”

@relation heart-disease-simplified

@attribute age numeric

@attribute sex { female, male}

@attribute chest_pain_type { typ_angina, asympt, non_anginal, atyp_angina}

@attribute cholesterol numeric

@attribute exercise_induced_angina { no, yes}

@attribute class { present, not_present}

@data

63,male,typ_angina,233,no,not_present

67,male,asympt,286,yes,present

67,male,asympt,229,yes,present

38,female,non_anginal,?,no,not_present

...

Atributo numérico

Atributo nominal

Con interfaz gráfica de usuario

The central image is the **Weka GUI Chooser** window, titled "Waikato Environment for Knowledge Analysis". It features a bird image and copyright information: "(c) 1999 - 2003 University of Waikato New Zealand". At the bottom, it presents four GUI options: **Simple CLI**, **Explorer**, **Experimenter**, and **KnowledgeFlow**. Blue arrows connect these options to their respective interface screenshots:

- Simple CLI**: A terminal window titled "Weka SimpleCLI" showing a welcome message and a list of commands: `java <classname> <args>`, `break`, `kill`, `cls`, `exit`, and `help <command>`.
- Explorer**: The **Weka Knowledge Explorer** window, showing the "Iris" dataset with 150 instances and 5 attributes. It displays a histogram for the "class" attribute and a "Visualize All" button.
- Experimenter**: The **Weka Experiment Environment** window, showing configuration options for experiments, including "Results Destination", "Experiment Type" (Cross-validation), and "Number of folds".
- KnowledgeFlow**: The **Weka KnowledgeFlow Environment** window, showing a visual workflow diagram with nodes like "dataSet", "CrossValidationFoldMaker", "AttributeSelection", "testSet trainingSet", "batchClassifier", "ClassifierPerformanceEvaluator", "DataVisualizer", "AttributeSummarizer", "ScatterPlotMatrix", and "TextViewer".

o usando biblioteca Weka de Java

Weka GUI

Weka 3.5.5
Program Applications Tools Visualization Windows Help

Explorer
Preprocess Classify Cluster Associate Select attributes Visualize

Open file... Open URL... Open DB... Generate... Undo Edit... Save...

Filter
Choose **None** Apply

Current relation
Relation: labor-neg-data
Instances: 57
Attributes: 17

Attributes
All None Invert Pattern

No.	Name
1	duration
2	wage-increase-first-year
3	wage-increase-second-year
4	wage-increase-third-year
5	cost-of-living-adjustment
6	working-hours
7	pension
8	standby-pay
9	shift-differential
10	education-allowance
11	statutory-holidays
12	vacation
13	longterm-disability-assistance
14	contribution-to-dental-plan
15	bereavement-assistance
16	contribution-to-health-plan
17	class

Remove

Herramientas de análisis

Selected attribute
Name: class
Missing: 0 (0%)
Distinct: 2
Type: Nominal
Unique: 0 (0%)

Label	Count
bad	20
good	37

Valores de un atributo escogido

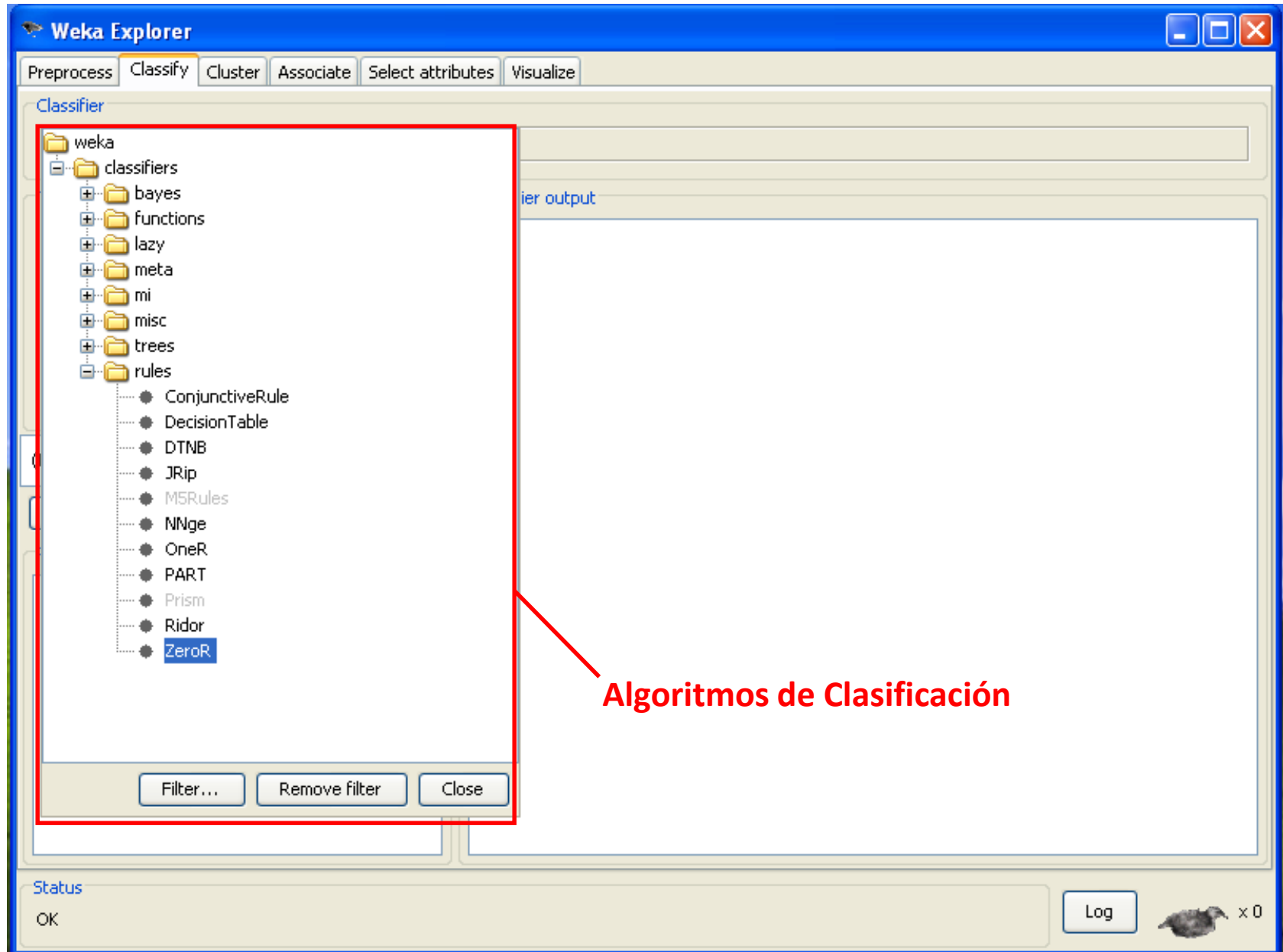
Class: class (Nom) Visualize All

20 37

Atributos a escoger

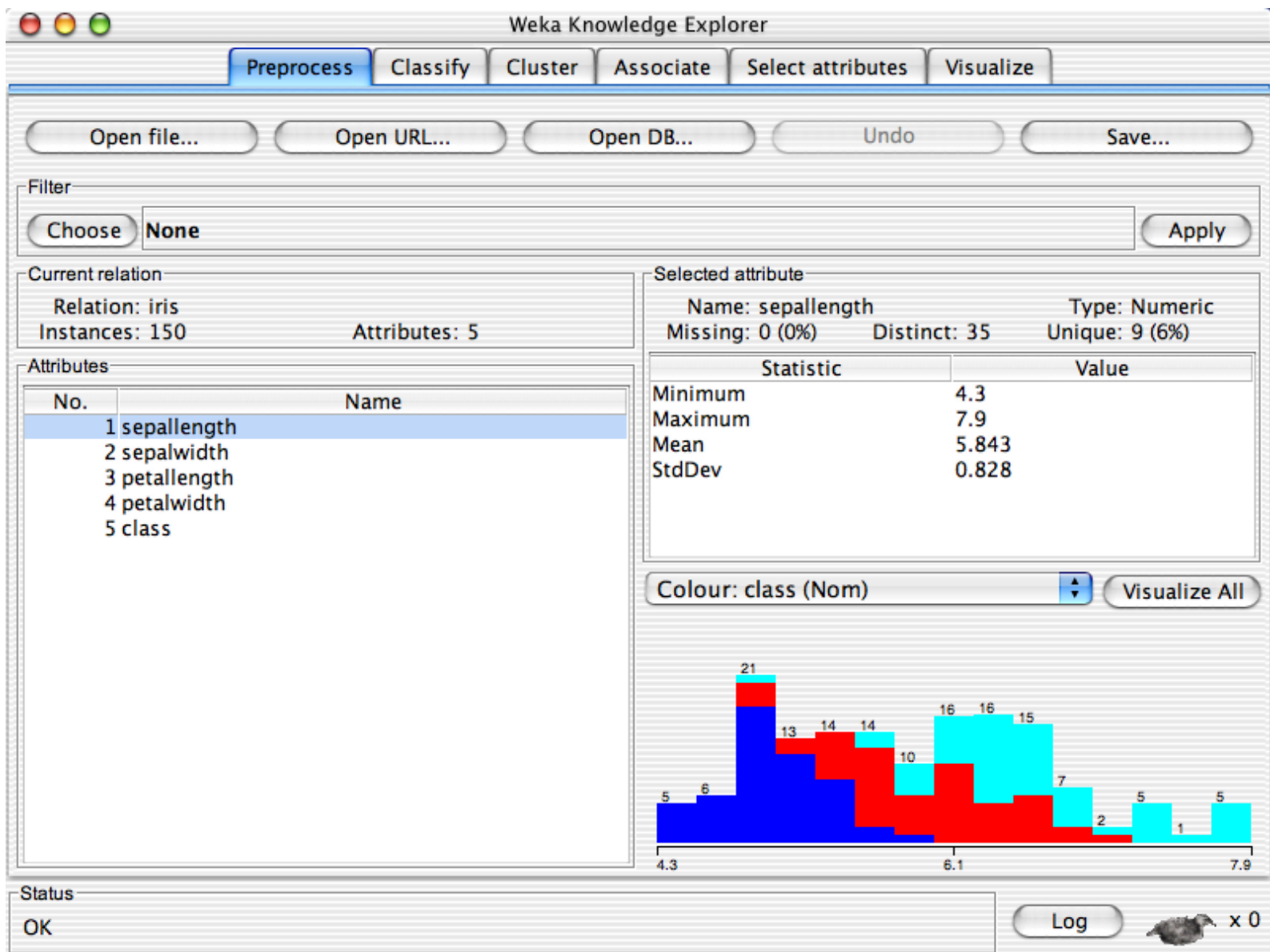
Status
OK Log x 0

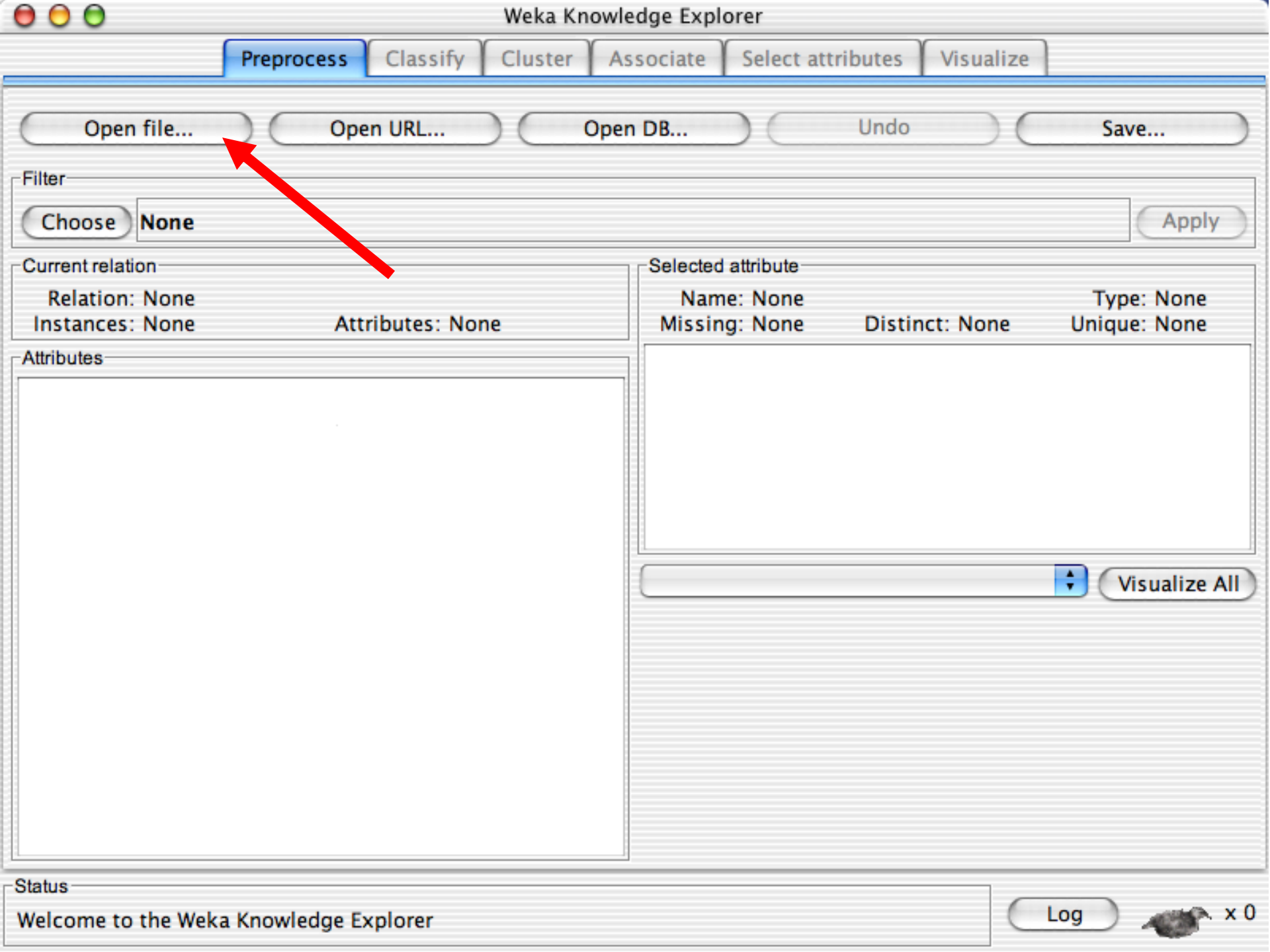
Weka GUI

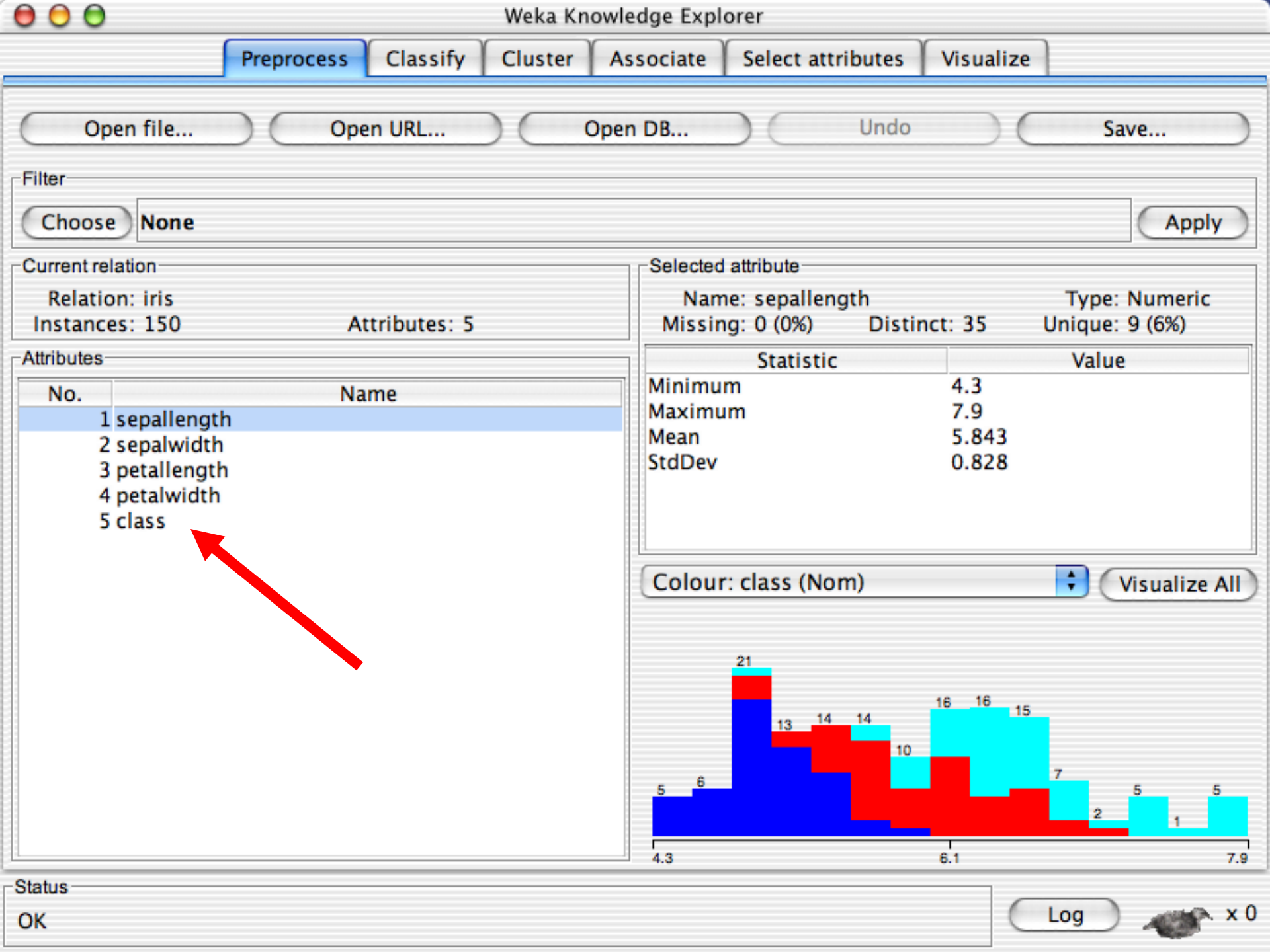


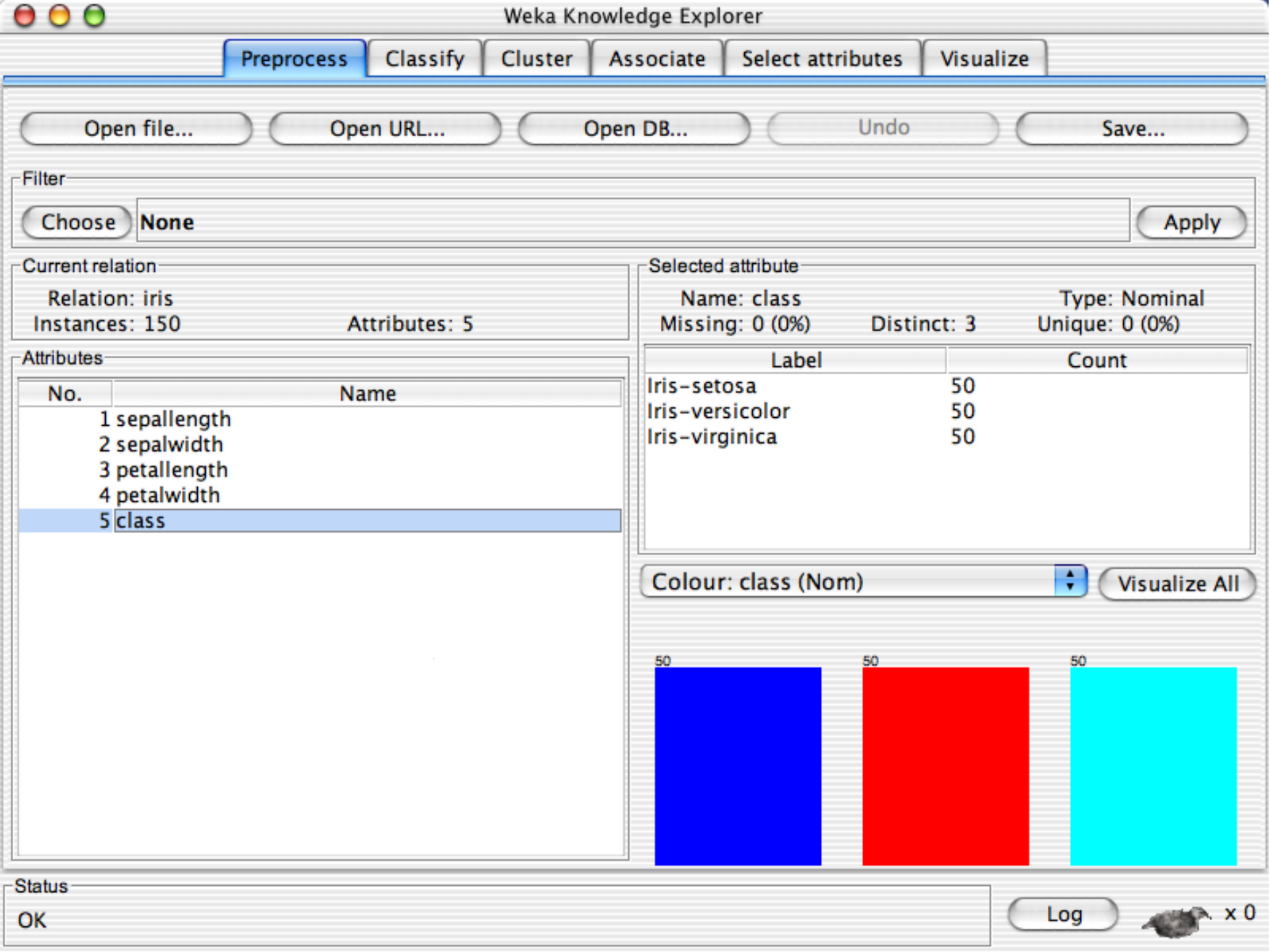
Explorar: Preprocesamiento

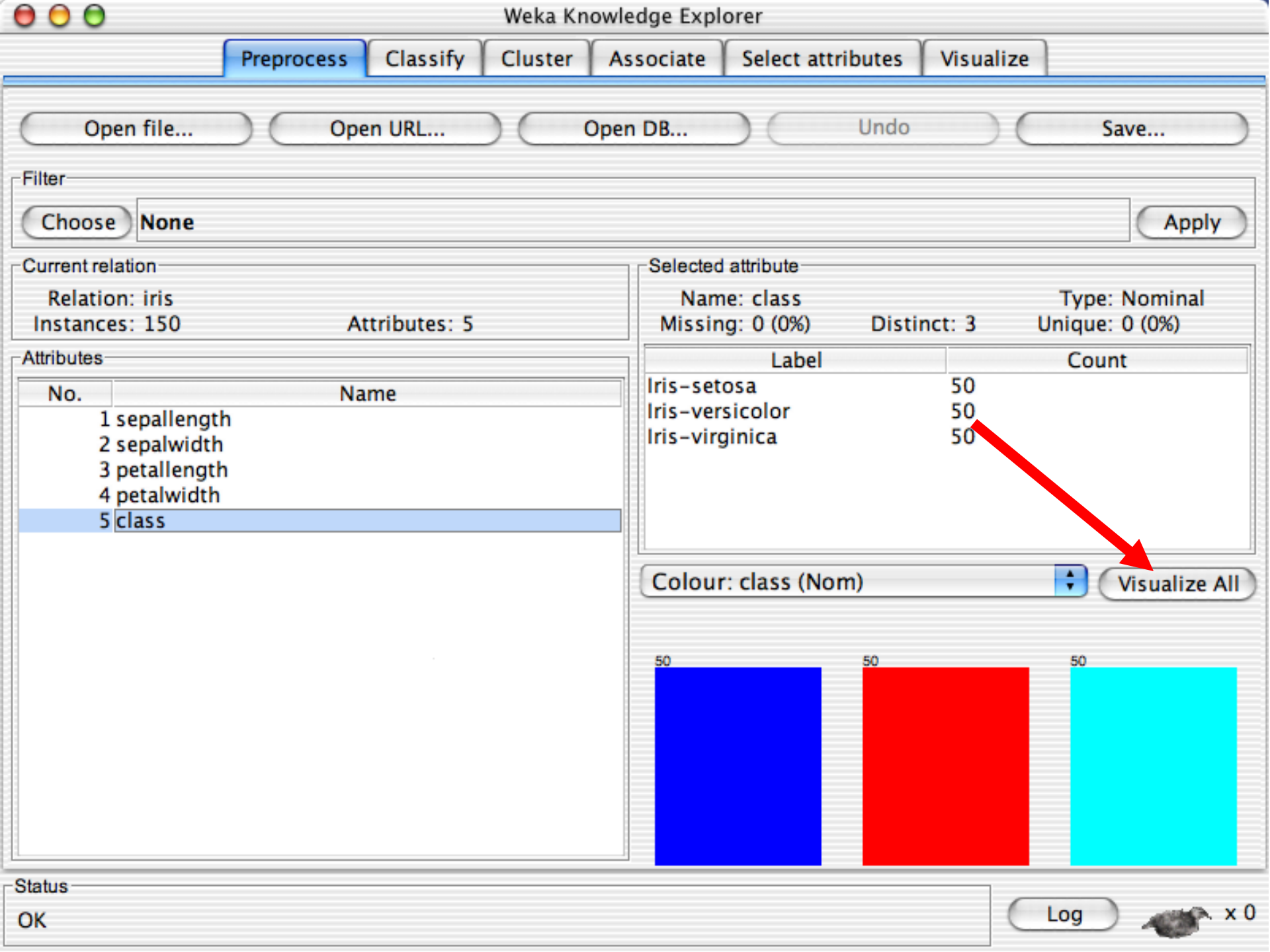
- Las herramientas de pre-procesamiento en WEKA son llamados “filtros”
- Los filtros son:
 - Discretizar, normalizar, selección de atributos, transformar, combinar atributos, etc.

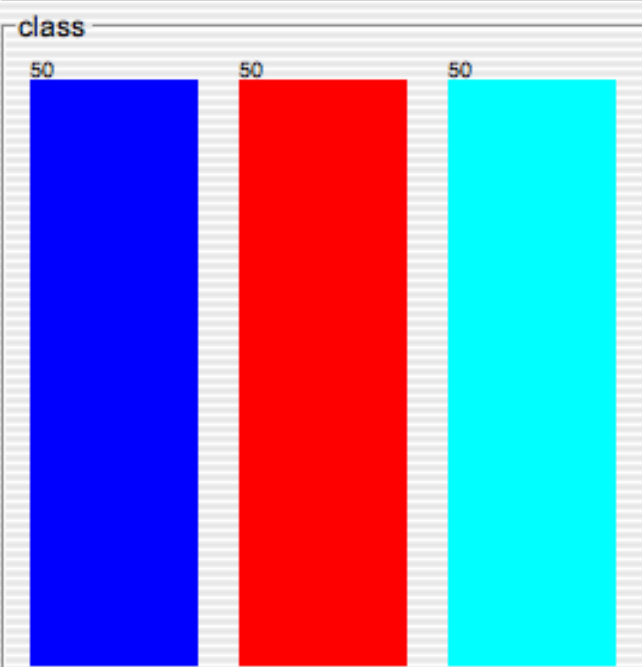
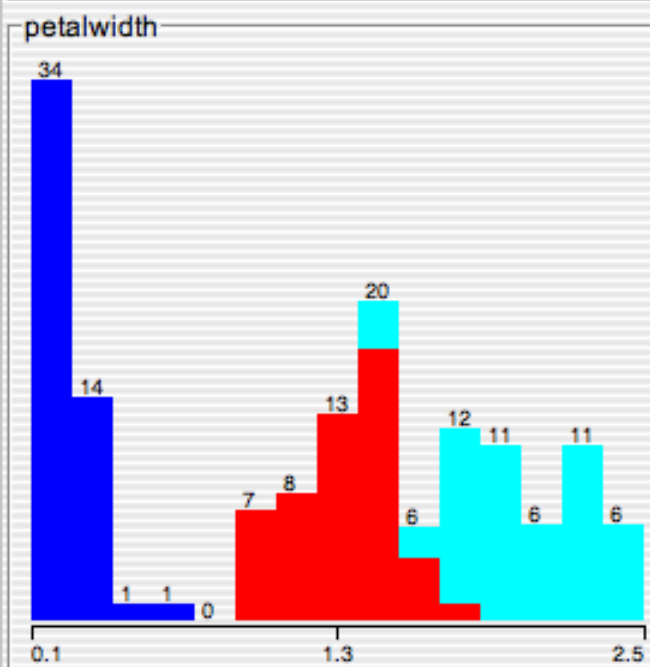
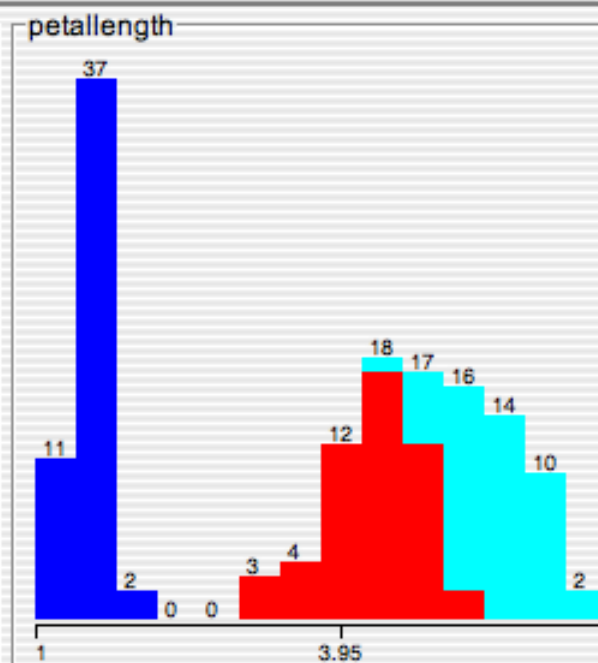
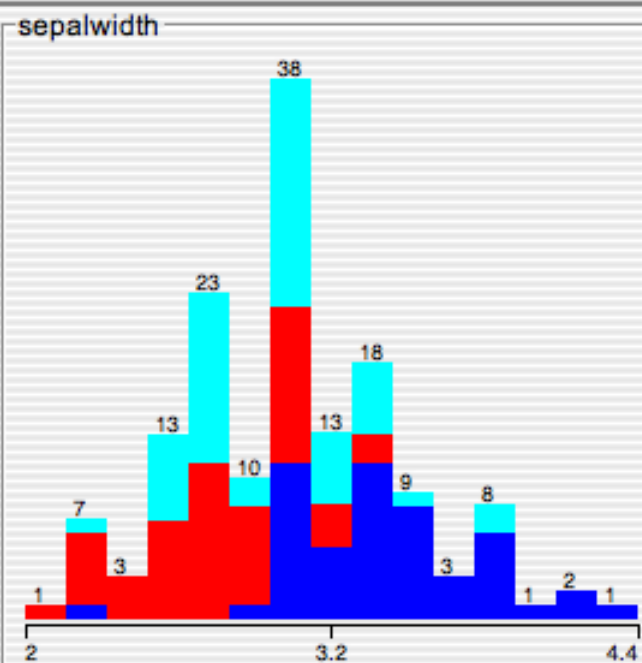
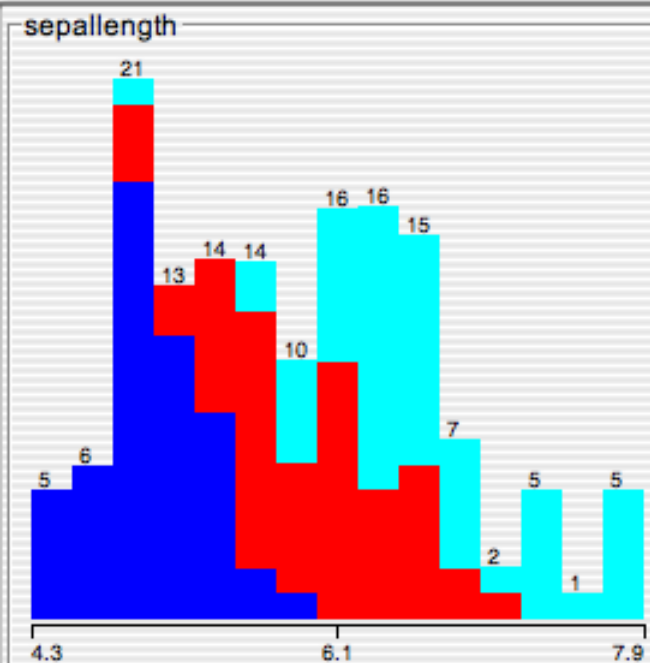














Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Open file...

Open URL...

Open DB...

Undo

Save...

Filter

Choose

None

Apply

Current relation

Relation: iris

Instances: 150

Attributes: 5

Attributes

No.	Name
1	sepal.length
2	sepal.width
3	petal.length
4	petal.width
5	class

Selected attribute

Name: petal.length

Type: Numeric

Missing: 0 (0%)

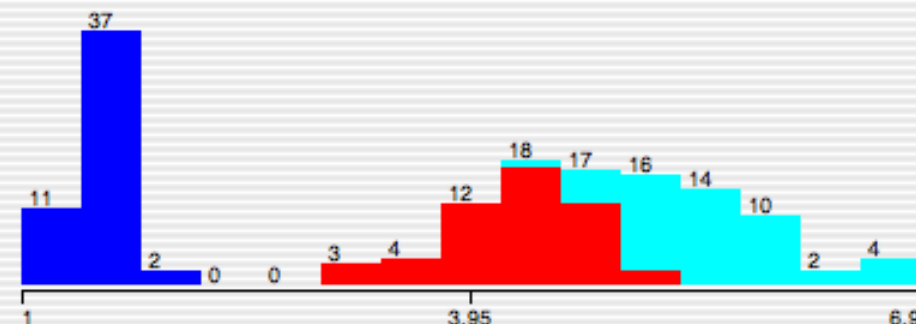
Distinct: 43

Unique: 10 (7%)

Statistic	Value
Minimum	1
Maximum	6.9
Mean	3.759
StdDev	1.764

Colour: class (Nom)

Visualize All

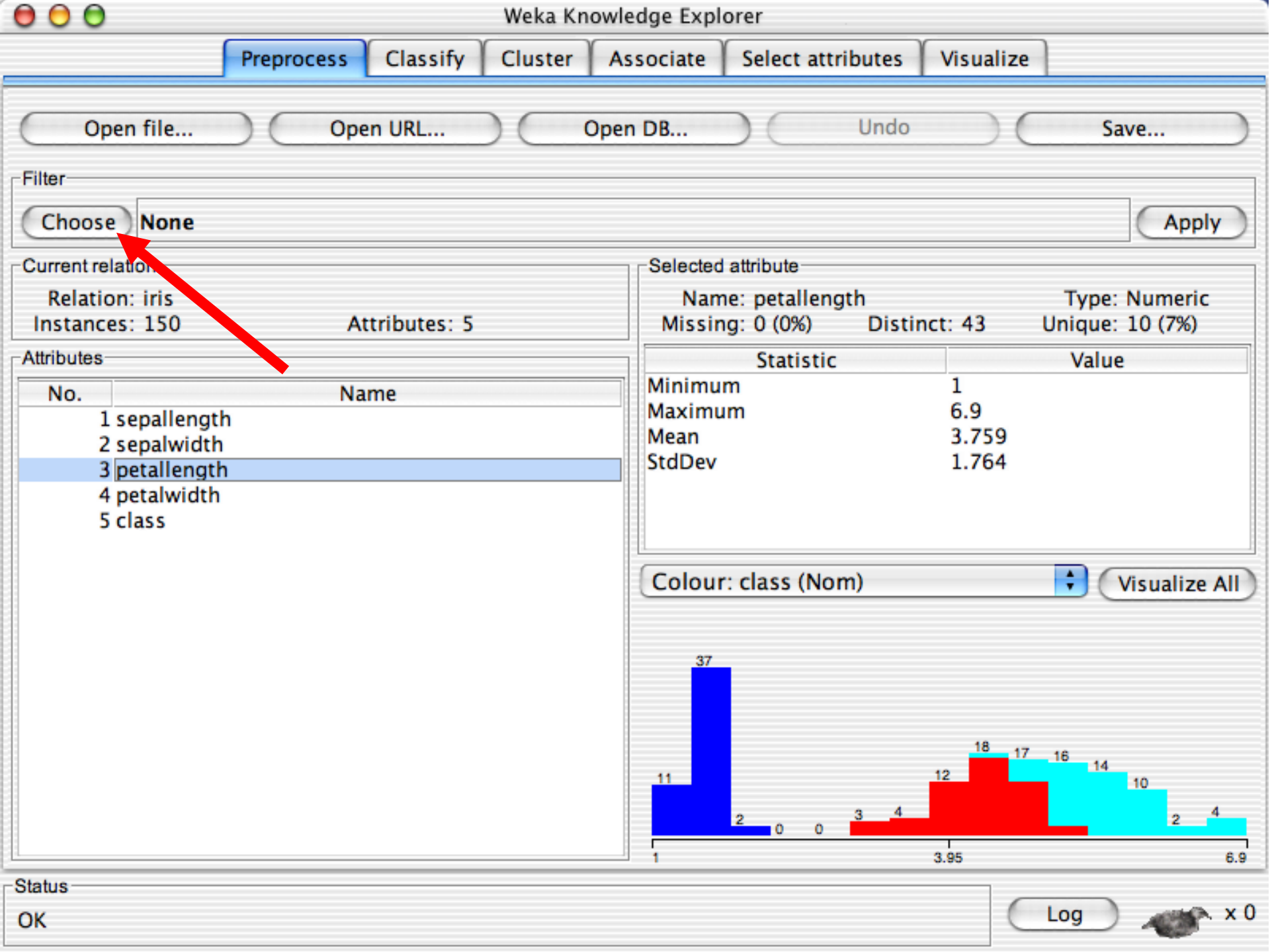


Status

OK

Log

 x 0





Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Open file...

Open URL...

Open DB...

Undo

Save...

Filter

- weka
 - filters
 - unsupervised
 - attribute
 - instance

Apply

Selected attribute

Name: petallength

Type: Numeric

Missing: 0 (0%)

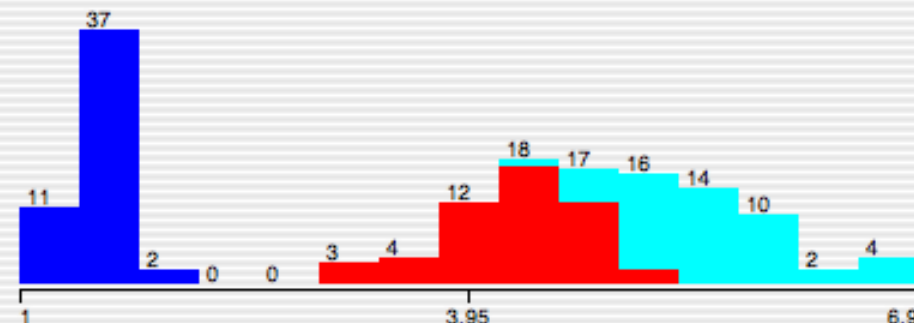
Distinct: 43

Unique: 10 (7%)

Statistic	Value
Minimum	1
Maximum	6.9
Mean	3.759
StdDev	1.764

Colour: class (Nom)

Visualize All

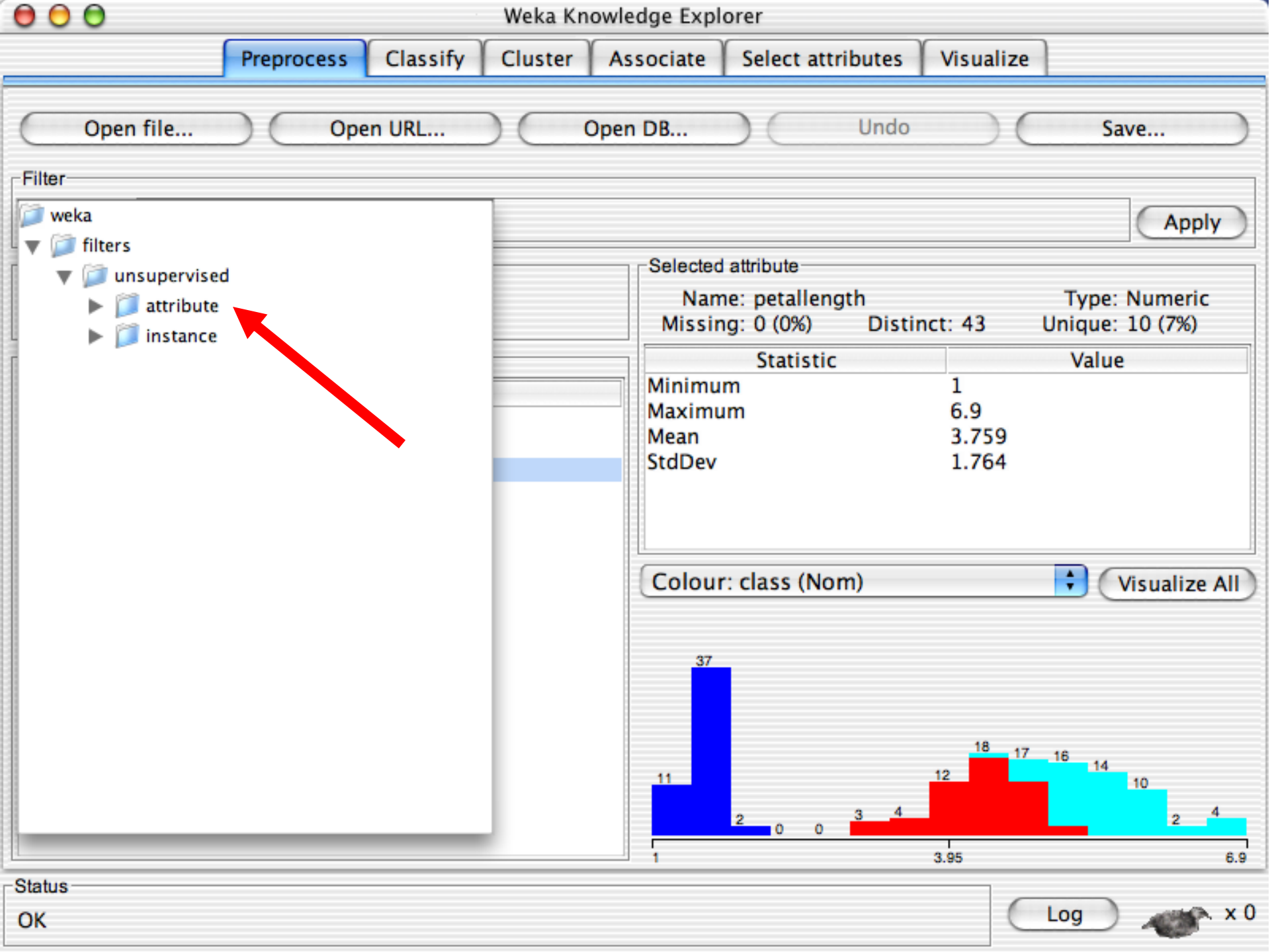


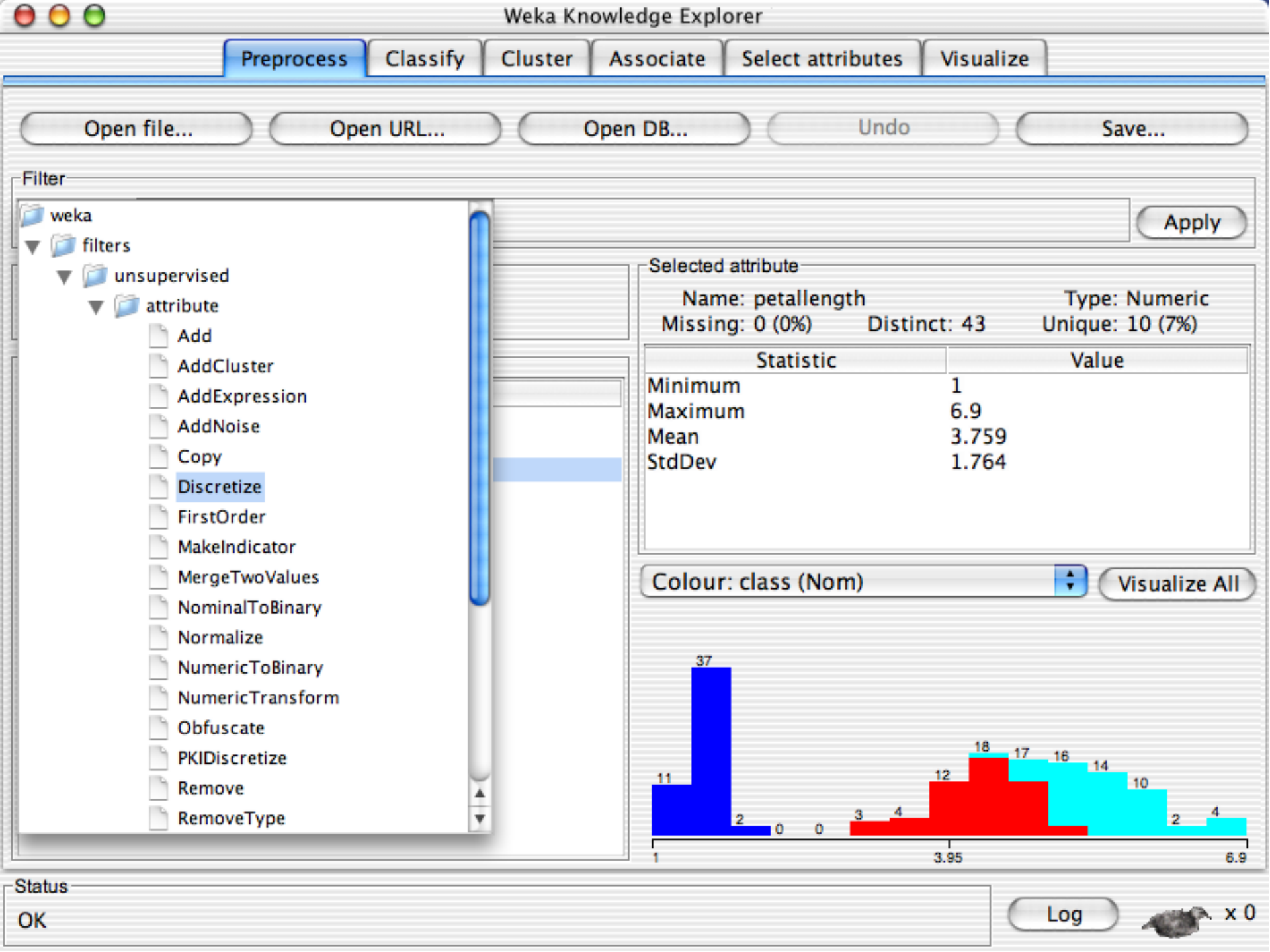
Status

OK

Log

 x 0







Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Open file...

Open URL...

Open DB...

Undo

Save...

Filter

Choose

Discretize -B 10 -R first-last

Apply

Current relation

Relation: iris

Instances: 150

Attributes: 5

Attributes

No.	Name
1	sepal.length
2	sepal.width
3	petal.length
4	petal.width
5	class

Selected attribute

Name: petal.length

Type: Numeric

Missing: 0 (0%)

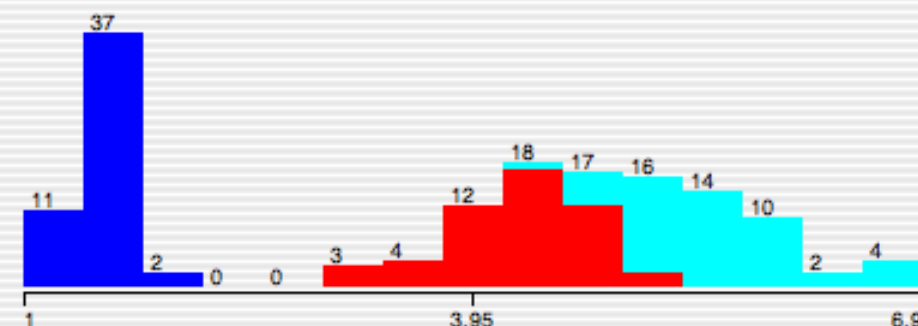
Distinct: 43

Unique: 10 (7%)

Statistic	Value
Minimum	1
Maximum	6.9
Mean	3.759
StdDev	1.764

Colour: class (Nom)

Visualize All



Status

OK

Log

 x 0



Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Open file...

Open URL...

Open DB...

Undo

Save...

Filter

Choose

Discretize -B 10 -R first-last

Apply

Current relation

Relation: iris

Instances: 150

Attributes: 5

Attributes

No.	Name
1	sepal.length
2	sepal.width
3	petal.length
4	petal.width
5	class

Selected attribute

Name: petal.length

Type: Numeric

Missing: 0 (0%)

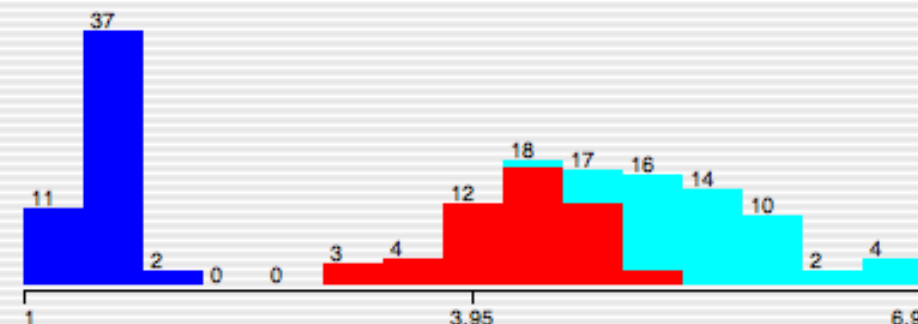
Distinct: 43

Unique: 10 (7%)

Statistic	Value
Minimum	1
Maximum	6.9
Mean	3.759
StdDev	1.764

Colour: class (Nom)

Visualize All

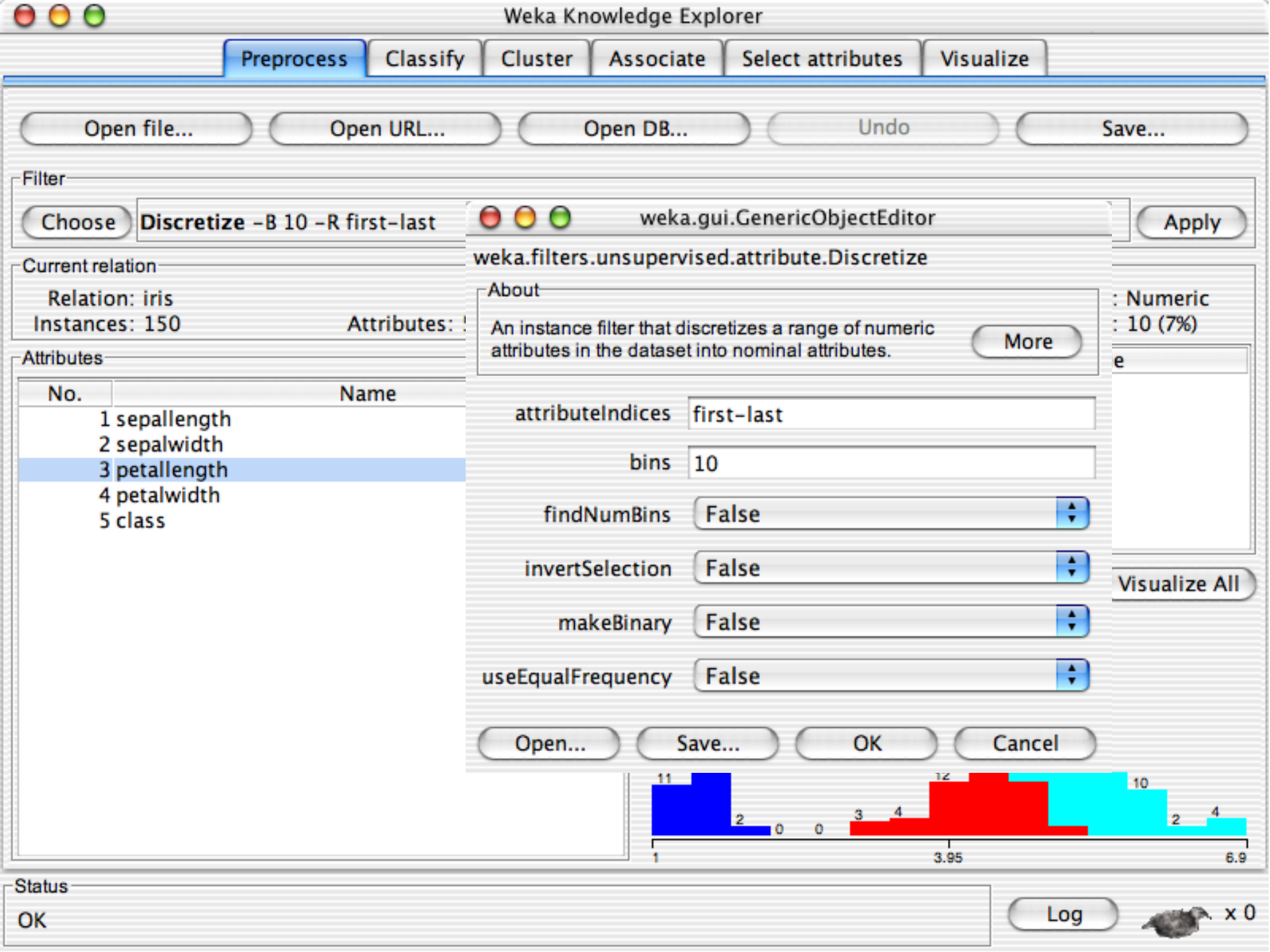


Status

OK

Log

 x 0



Weka Knowledge Explorer

Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Open file...

Open URL...

Open DB...

Undo

Save...

Filter

Choose

Discretize -B 10 -R first-last

Current relation

Relation: iris

Instances: 150

Attributes: 5

Attributes

No.	Name
1	sepal.length
2	sepal.width
3	petal.length
4	petal.width
5	class

weka.gui.GenericObjectEditor

weka.filters.unsupervised.attribute.Discretize

About

An instance filter that discretizes a range of numeric attributes in the dataset into nominal attributes.

More

attributeIndices

first-last

bins

10

findNumBins

False

invertSelection

False

makeBinary

False

useEqualFrequency

False

Open...

Save...

OK

Cancel

Visualize All

11

2

0

0

3

4

14

10

2

4

1

3.95

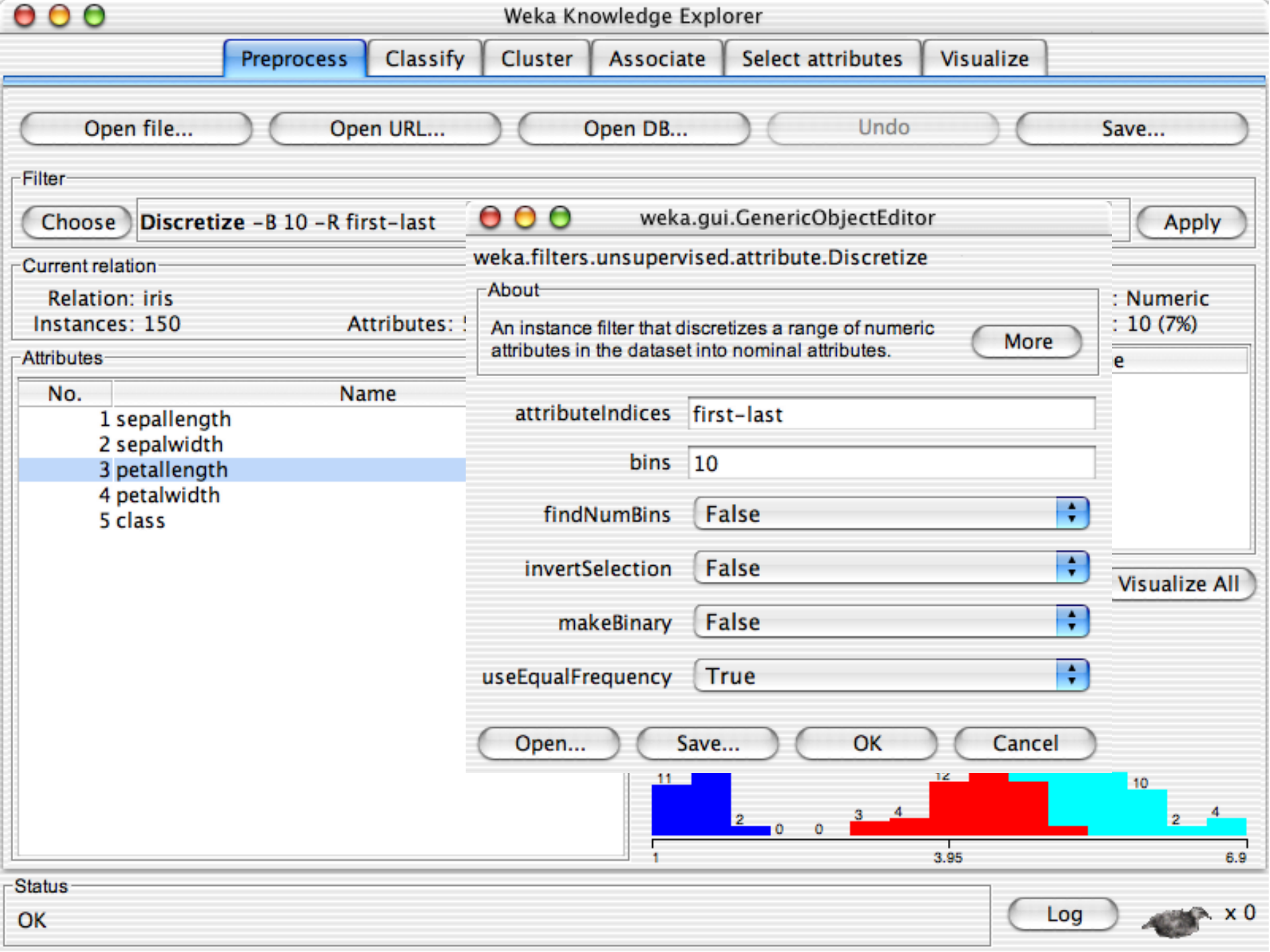
6.9

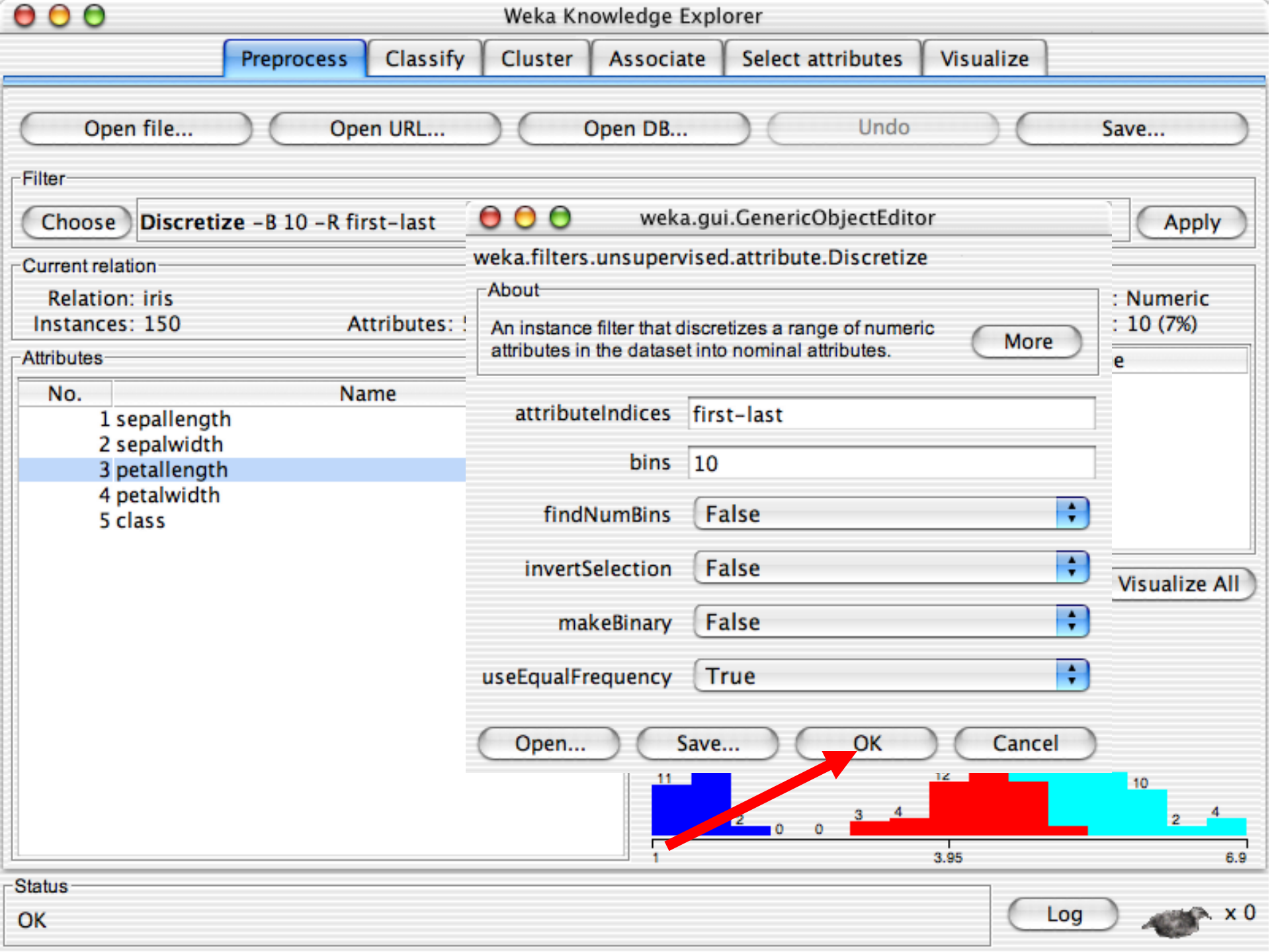
Status

OK

Log

x 0







Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Open file...

Open URL...

Open DB...

Undo

Save...

Filter

Choose

Discretize -F -B 10 -R first-last

Apply

Current relation

Relation: iris

Instances: 150

Attributes: 5

Attributes

No.	Name
1	sepal.length
2	sepal.width
3	petal.length
4	petal.width
5	class

Selected attribute

Name: petal.length

Type: Numeric

Missing: 0 (0%)

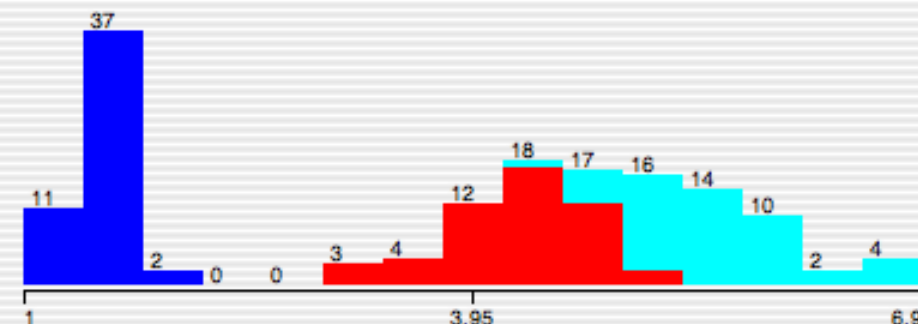
Distinct: 43

Unique: 10 (7%)

Statistic	Value
Minimum	1
Maximum	6.9
Mean	3.759
StdDev	1.764

Colour: class (Nom)

Visualize All

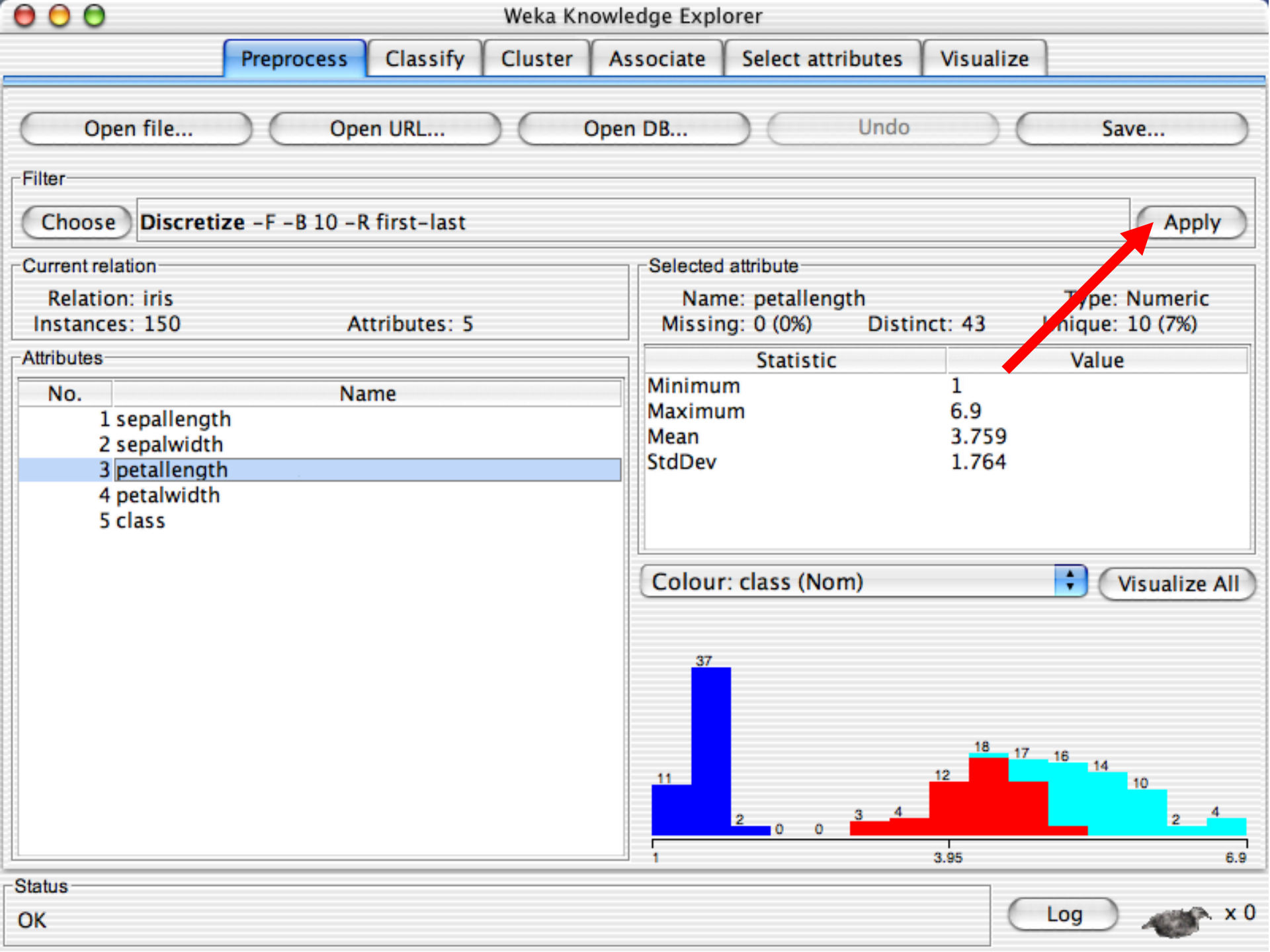


Status

OK

Log

 x 0





Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Open file...

Open URL...

Open DB...

Undo

Save...

Filter

Choose

Discretize -F -B 10 -R first-last

Apply

Current relation

Relation: iris-weka.filters.unsupervised.attribute.Disc...

Instances: 150

Attributes: 5

Attributes

No.	Name
1	sepal.length
2	sepal.width
3	petal.length
4	petal.width
5	class

Selected attribute

Name: petal.length

Type: Nominal

Missing: 0 (0%)

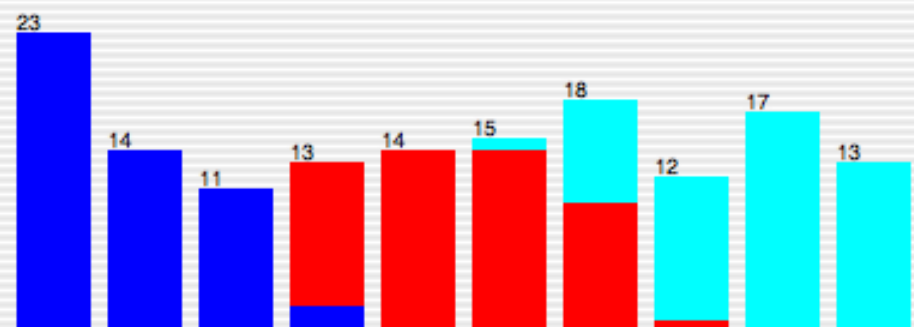
Distinct: 10

Unique: 0 (0%)

Label	Count
'(-inf-1.45]'	23
'(1.45-1.55]'	14
'(1.55-1.8]'	11
'(1.8-3.95]'	13
'(3.95-4.35]'	14
'(4.35-4.65]'	15
'(4.65-5.05]'	18

Colour: class (Nom)

Visualize All



Status

OK

Log

 x 0

Explorar: construir “clasificadores”

- Son modelos para predecir cantidades nominales o numericas
- Ejemplo son:
 - Árboles de decisión
 - Support vector machines,
 - Perceptron Multi-capas,
 - Regresión lógica,
 - Red de Bayes, ...



Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose

ZeroR

Test options

☐ Use training set☐ Supplied test set

Set...

☒ Cross-validation Folds 10☐ Percentage split % 66

More options...

(Nom) class



Start

Stop

Result list (right-click for options)

Classifier output

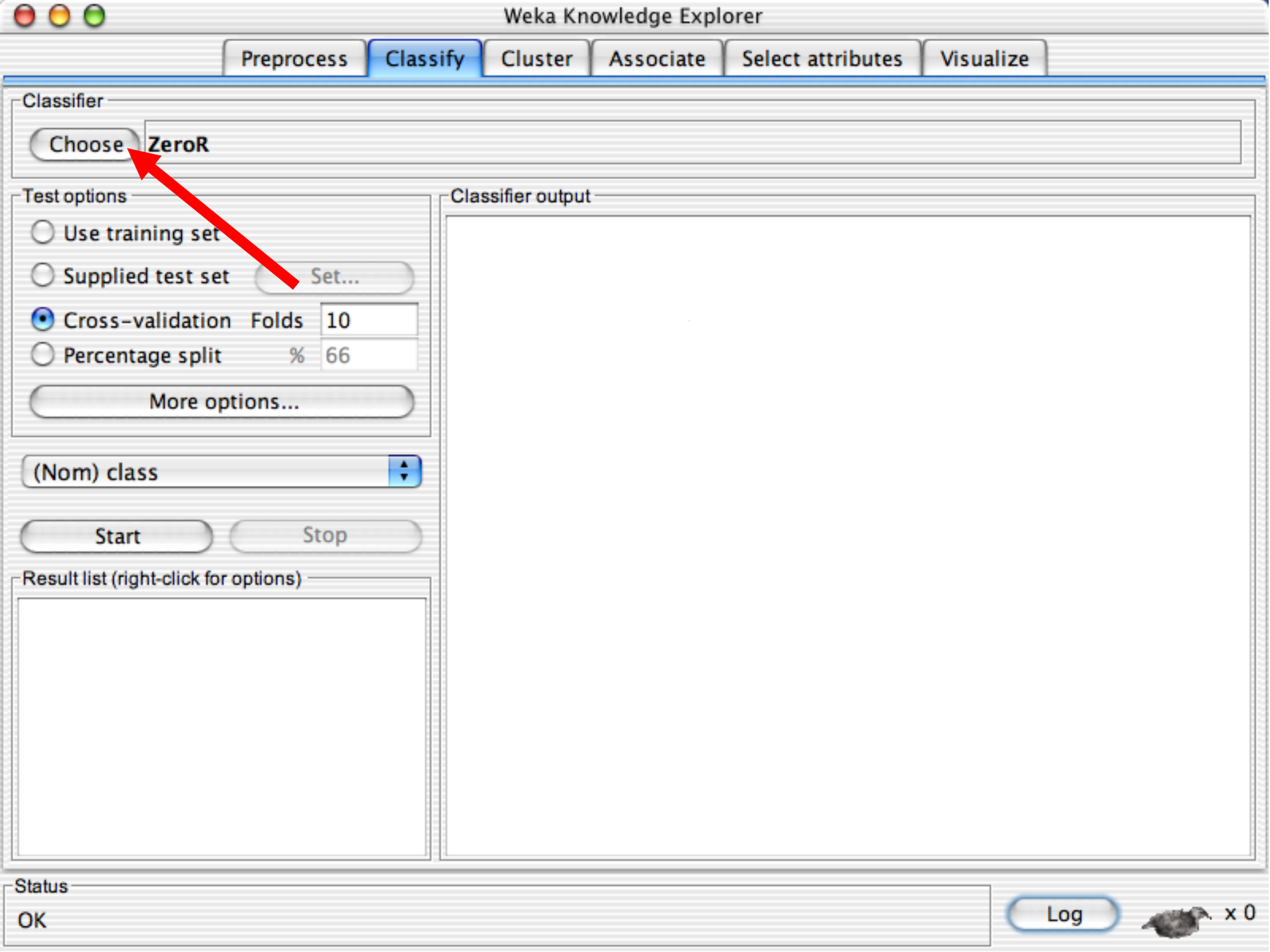
Status

OK

Log



x 0



Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose

ZeroR

Test options

☐ Use training set

☐ Supplied test set

Set...

☒ Cross-validation

Folds

10

☐ Percentage split

%

66

More options...

(Nom) class



Start

Stop

Result list (right-click for options)

Classifier output

Status

OK

Log



x 0



Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

- weka
 - classifiers
 - bayes
 - functions
 - lazy
 - meta
 - misc
 - trees
 - adtree
 - DecisionStump
 - Id3
 - j48
 - J48
 - lmt
 - m5
 - RandomForest
 - RandomTree
 - REPTree
 - UserClassifier
 - rules

Classifier output

Status

OK

Log



x 0



Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose

J48 -C 0.25 -M 2

Test options

☐ Use training set☐ Supplied test set

Set...

☒ Cross-validation Folds 10☐ Percentage split % 66

More options...

(Nom) class



Start

Stop

Result list (right-click for options)

Classifier output

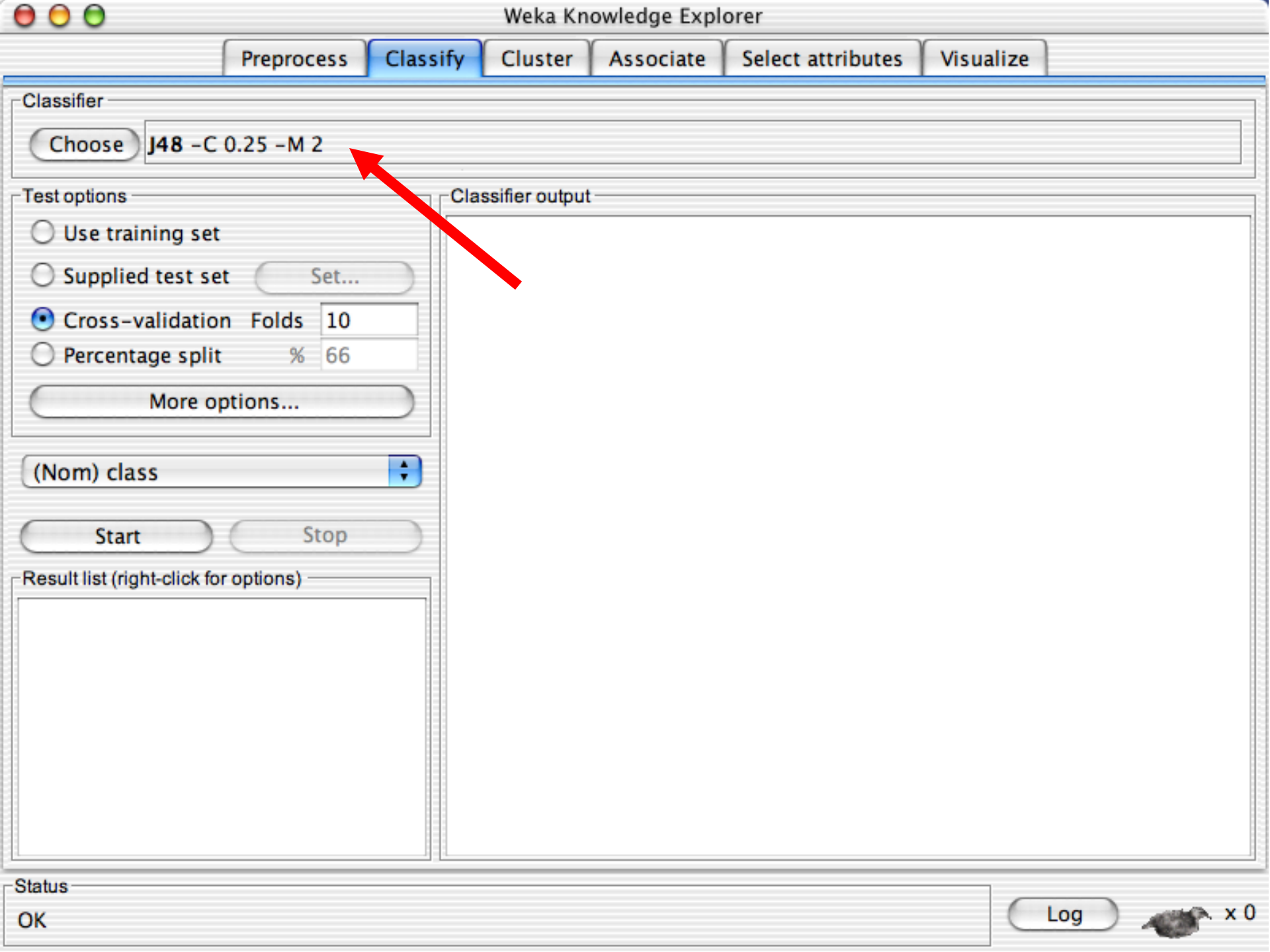
Status

OK

Log



x 0



Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose

J48 -C 0.25 -M 2

Test options

☐ Use training set☐ Supplied test set

Set...

☒ Cross-validation Folds 10☐ Percentage split % 66

More options...

(Nom) class

Start

Stop

Result list (right-click for options)

Classifier output

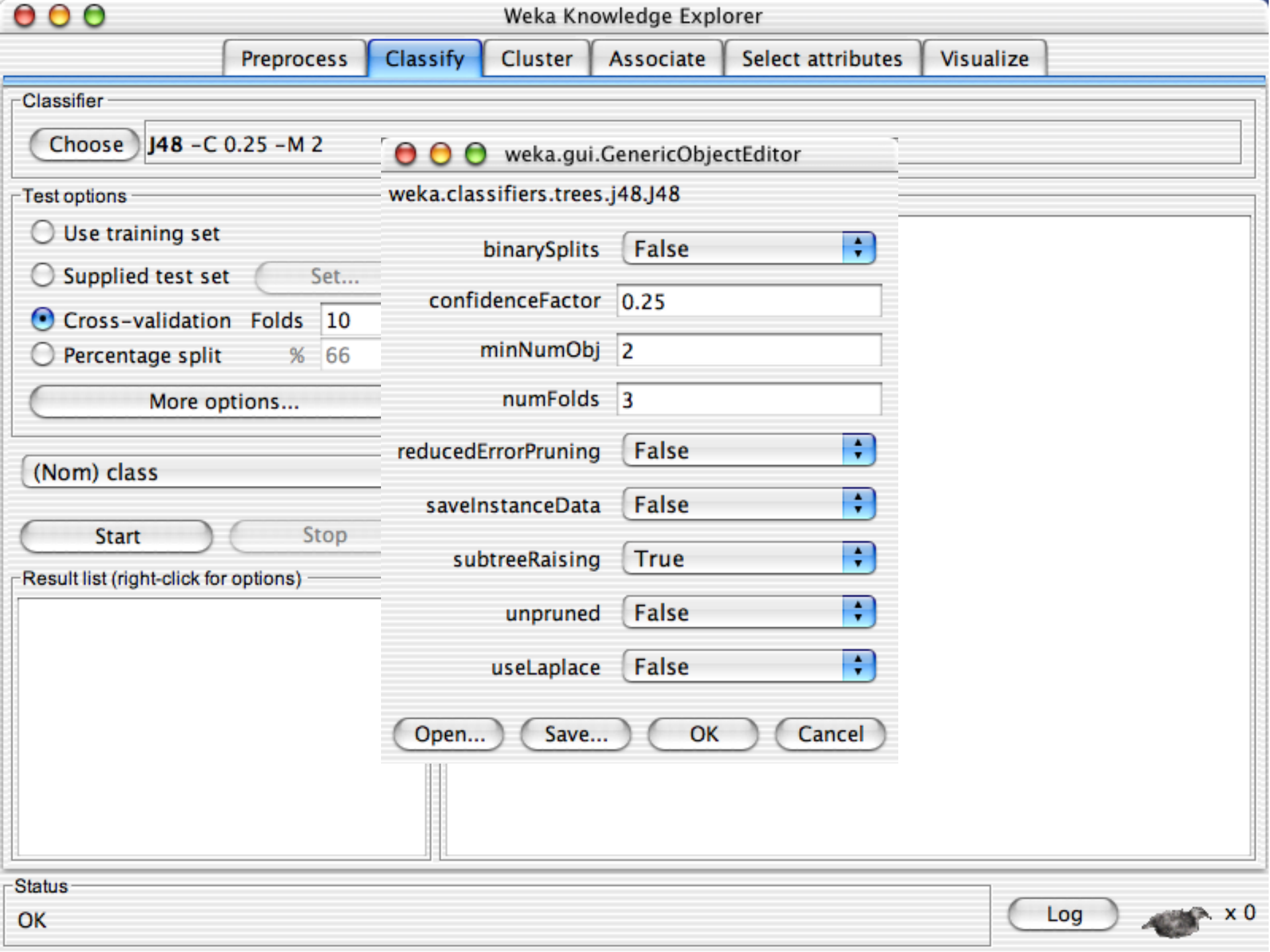
Status

OK

Log



x 0



Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose J48 -C 0.25 -M 2

Test options

☐ Use training set☐ Supplied test set Set...☒ Cross-validation Folds 10☐ Percentage split % 66

More options...

(Nom) class

Start

Stop

Result list (right-click for options)

weka.gui.GenericObjectEditor

weka.classifiers.trees.J48.J48

binarySplits False

confidenceFactor 0.25

minNumObj 2

numFolds 3

reducedErrorPruning False

saveInstanceData False

subtreeRaising True

unpruned False

useLaplace False

Open...

Save...

OK

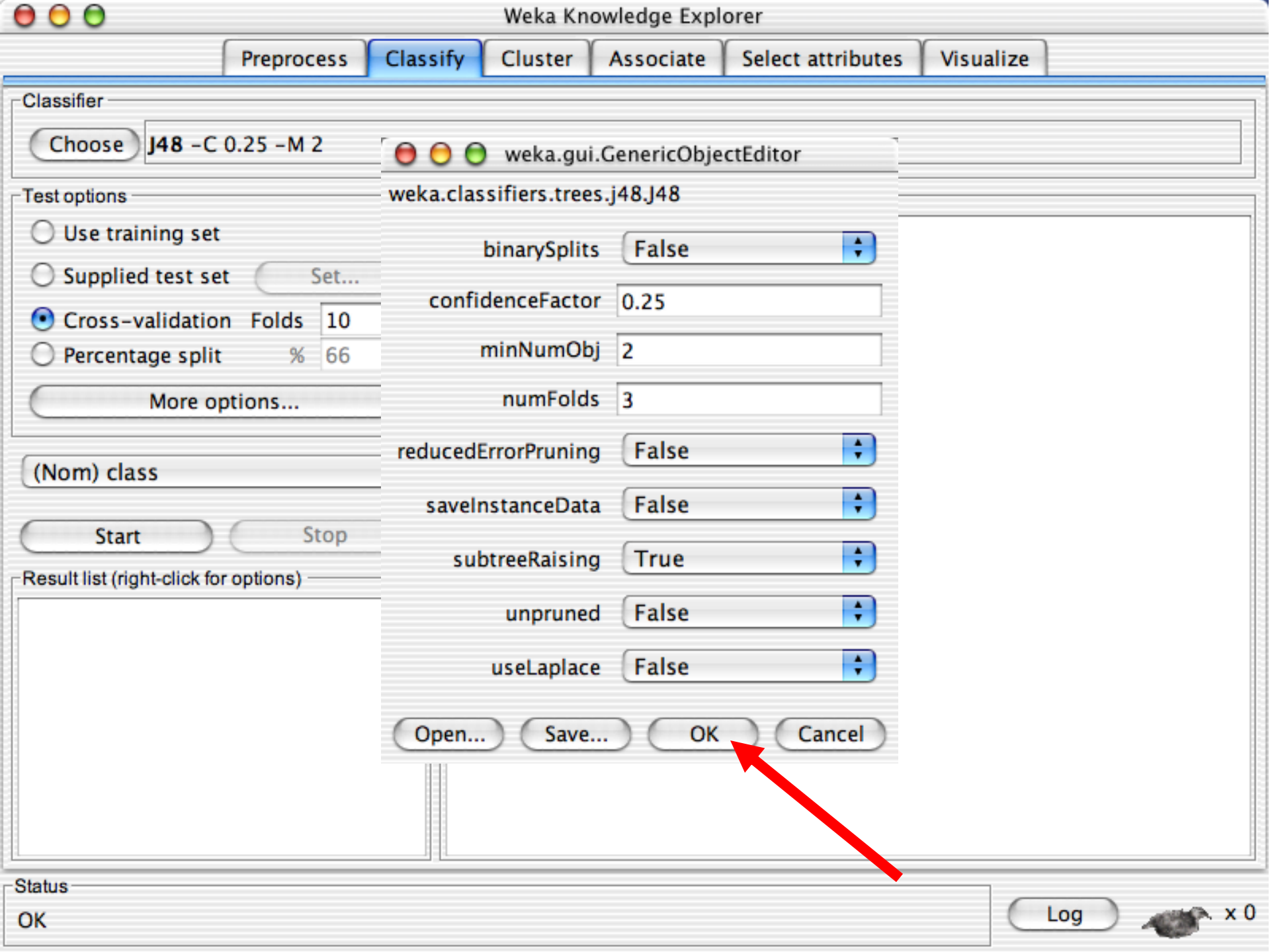
Cancel

Status

OK

Log

x 0



Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose

J48 -C 0.25 -M 2

Test options

☐ Use training set☐ Supplied test set

Set...

☒ Cross-validation Folds 10☐ Percentage split % 66

More options...

(Nom) class

Start

Stop

Result list (right-click for options)



weka.gui.GenericObjectEditor

weka.classifiers.trees.j48.J48

binarySplits False

confidenceFactor 0.25

minNumObj 2

numFolds 3

reducedErrorPruning False

saveInstanceData False

subtreeRaising True

unpruned False

useLaplace False

Open...

Save...

OK

Cancel

Status

OK

Log

x 0



Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose

J48 -C 0.25 -M 2

Test options

☐ Use training set☐ Supplied test set

Set...

☒ Cross-validation Folds 10☐ Percentage split % 66

More options...

(Nom) class



Start

Stop

Result list (right-click for options)

Classifier output

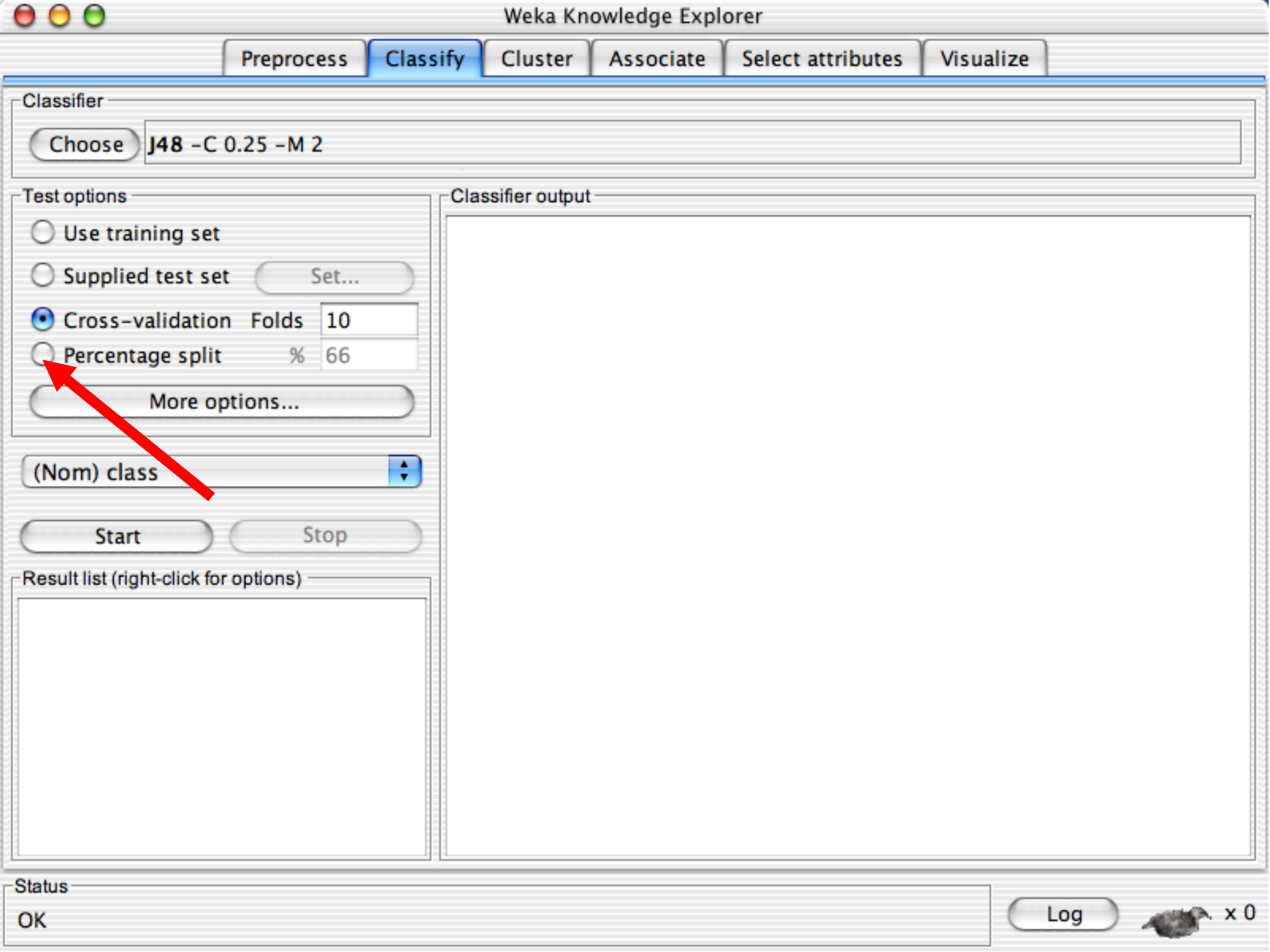
Status

OK

Log



x 0



Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose

J48 -C 0.25 -M 2

Test options

☐ Use training set☐ Supplied test set

Set...

☒ Cross-validation Folds 10☐ Percentage split % 66

More options...

(Nom) class



Start

Stop

Result list (right-click for options)

Classifier output

Status

OK

Log



x 0



Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose

J48 -C 0.25 -M 2

Test options

☐ Use training set☐ Supplied test set

Set...

☐ Cross-validation Folds 10☒ Percentage split % 66

More options...

(Nom) class



Start

Stop

Result list (right-click for options)

Classifier output

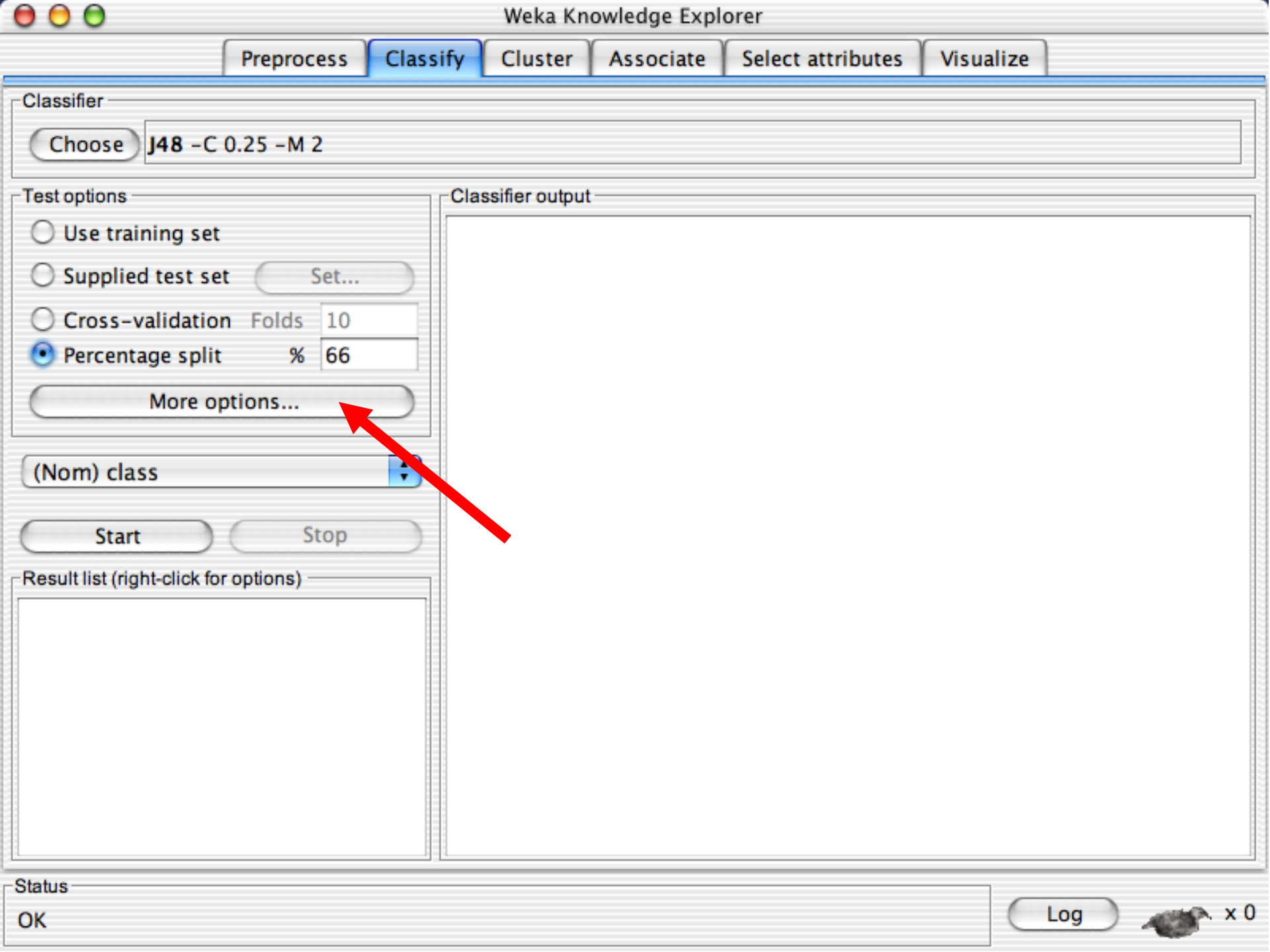
Status

OK

Log



x 0



Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose

J48 -C 0.25 -M 2

Test options

☐ Use training set

☐ Supplied test set

Set...

☐ Cross-validation Folds 10

☒ Percentage split % 66

More options...

(Nom) class

Start

Stop

Result list (right-click for options)

Classifier output

Status

OK

Log



x 0



Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose J48 -C 0.25 -M 2

Test options

☐ Use training set☐ Supplied test set Set...☐ Cross-validation Folds 10☒ Percentage split % 66

More options...

(Nom) class

Start

Stop

Result list (right-click for options)

Classifier output

Classifier evaluation opt

☒ Output model☒ Output per-class stats☐ Output entropy evaluation measures☒ Output confusion matrix☒ Store predictions for visualization☐ Output text predictions on test set☐ Cost-sensitive evaluation Set...

Random seed for XVal / % Split 1

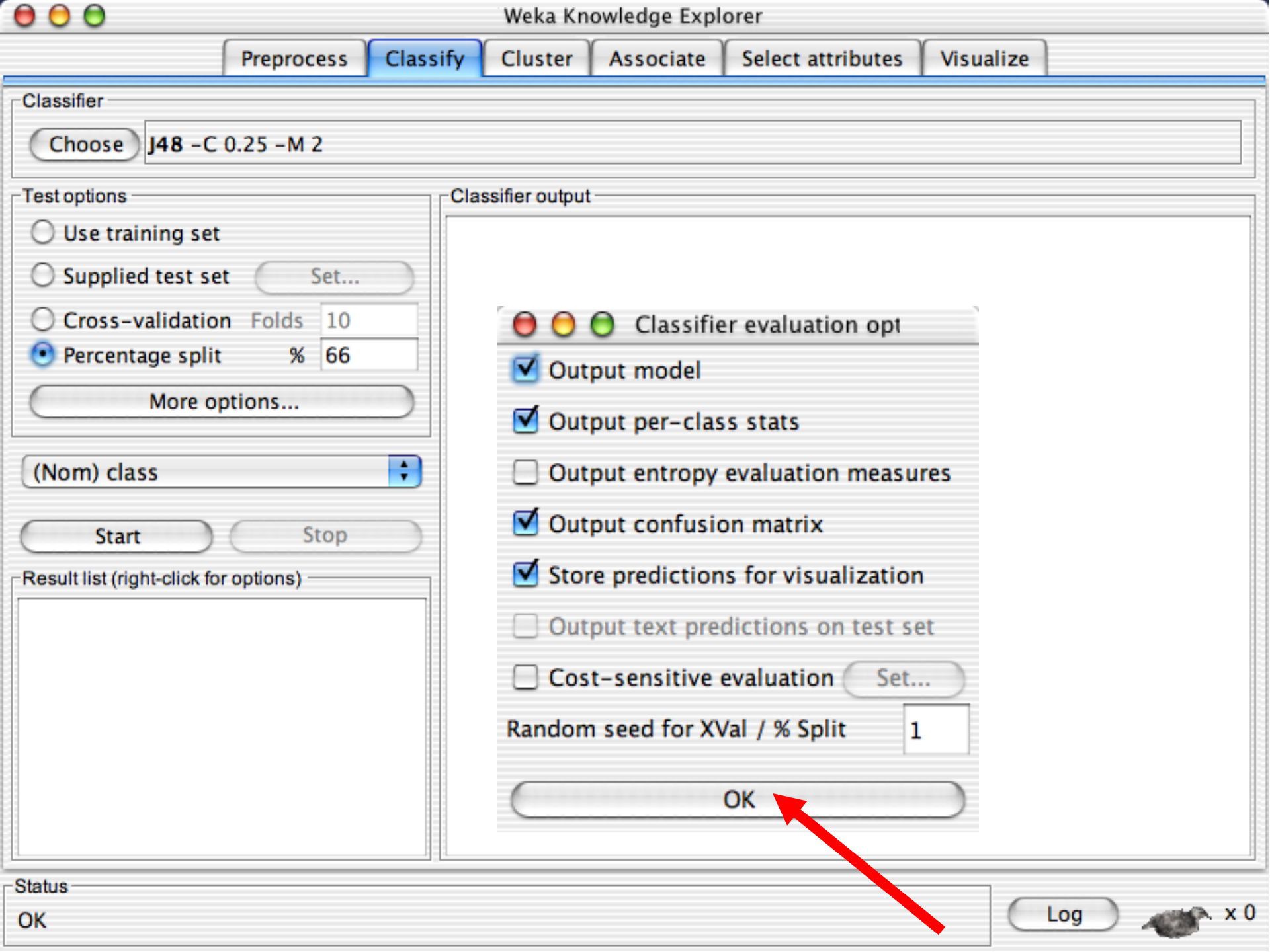
OK

Status

OK

Log

x 0





Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose

J48 -C 0.25 -M 2

Test options

☐ Use training set☐ Supplied test set

Set...

☐ Cross-validation Folds 10☒ Percentage split % 66

More options...

(Nom) class



Start

Stop

Result list (right-click for options)

Classifier output

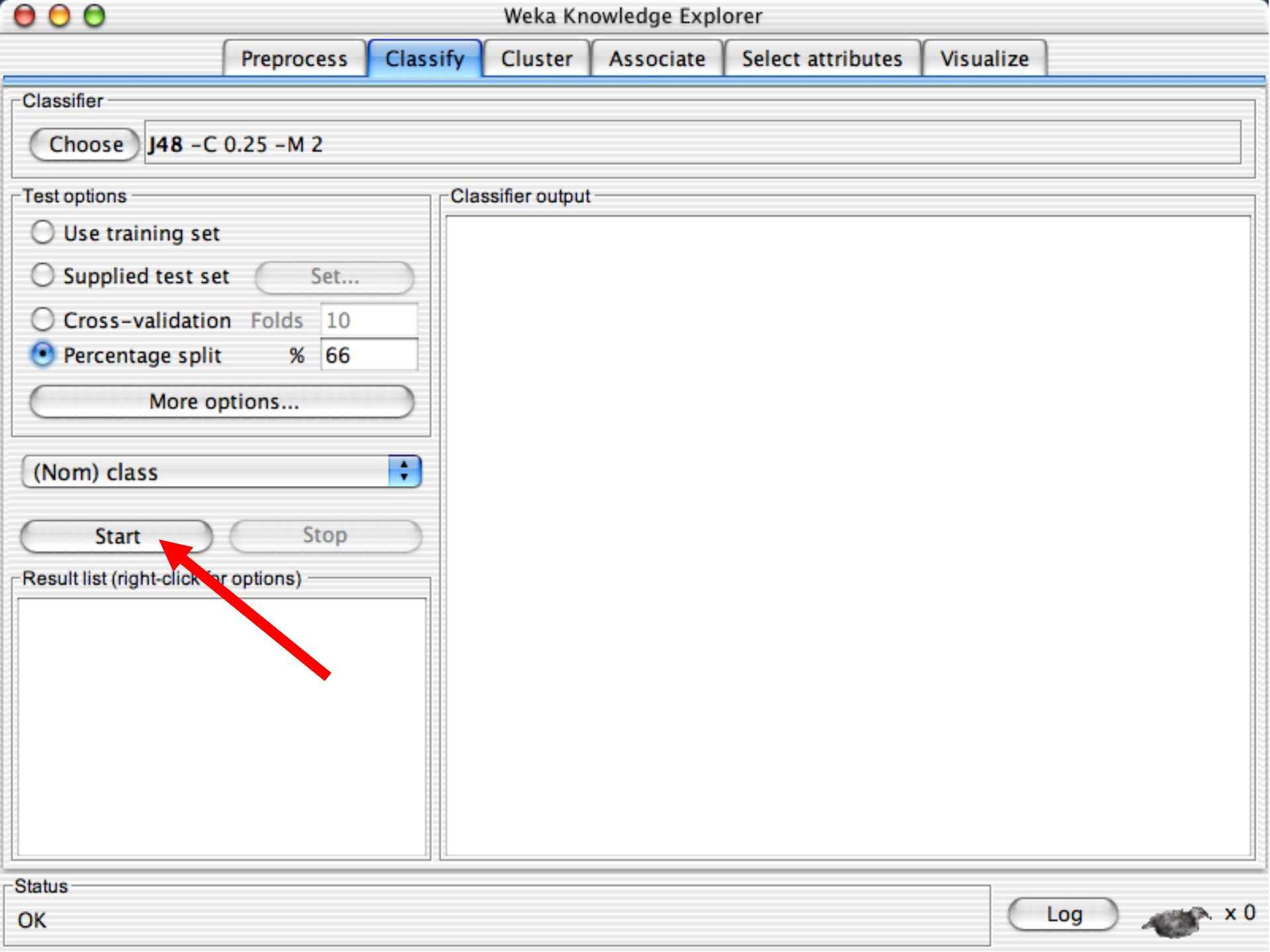
Status

OK

Log



x 0



Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose

J48 -C 0.25 -M 2

Test options

☐ Use training set

☐ Supplied test set

Set...

☐ Cross-validation Folds 10

☒ Percentage split % 66

More options...

(Nom) class

Start

Stop

Result list (right-click for options)

Classifier output

Status

OK

Log



x 0



Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose J48 -C 0.25 -M 2

Test options

☐ Use training set☐ Supplied test set Set...☐ Cross-validation Folds 10☒ Percentage split % 66

More options...

(Nom) class

Start

Stop

Result list (right-click for options)

11:49:05 - trees.j48.J48

Classifier output

=== Run information ===

Scheme: weka.classifiers.trees.j48.J48 -C 0.25 -M 2
Relation: iris
Instances: 150
Attributes: 5

sepalwidth
sepalwidth
petalwidth
petalwidth
class

Test mode: split 66% train, remainder test

=== Classifier model (full training set) ===

J48 pruned tree

```
-----  
petalwidth <= 0.6: Iris-setosa (50.0)  
petalwidth > 0.6  
| petalwidth <= 1.7  
| | petalwidth <= 4.9: Iris-versicolor (48.0/1.0)  
| | petalwidth > 4.9  
| | | petalwidth <= 1.5: Iris-virginica (3.0)  
| | | petalwidth > 1.5: Iris-versicolor (3.0/1.0)  
| petalwidth > 1.7: Iris-virginica (46.0/1.0)
```

Number of Leaves : 5

Status

OK

Log

 x 0

Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose J48 -C 0.25 -M 2

Test options

☐ Use training set☐ Supplied test set

Set...

☐ Cross-validation Folds 10☒ Percentage split % 66

More options...

(Nom) class

Start

Stop

Result list (right-click for options)

11:49:05 - trees.j48.J48

Classifier output

=== Run information ===

Scheme: weka.classifiers.trees.j48.J48 -C 0.25 -M 2
Relation: iris
Instances: 150
Attributes: 5

sepalwidth
sepalwidth
petalwidth
petalwidth
class

Test mode: split 66% train, remainder test

=== Classifier model (full training set) ===

J48 pruned tree

```
-----  
petalwidth <= 0.6: Iris-setosa (50.0)  
petalwidth > 0.6  
| petalwidth <= 1.7  
| | petalwidth <= 4.9: Iris-versicolor (48.0/1.0)  
| | petalwidth > 4.9  
| | | petalwidth <= 1.5: Iris-virginica (3.0)  
| | | petalwidth > 1.5: Iris-versicolor (3.0/1.0)  
| petalwidth > 1.7: Iris-virginica (46.0/1.0)
```

Number of Leaves : 5



Status

OK

Log

 x 0

Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose J48 -C 0.25 -M 2

Test options

☐ Use training set☐ Supplied test set

Set...

☐ Cross-validation Folds 10☒ Percentage split % 66

More options...

(Nom) class

Start

Stop

Result list (right-click for options)

11:49:05 - trees.j48.J48

Classifier output

Time taken to build model: 0.24 seconds

=== Evaluation on test split ===

=== Summary ===

Correctly Classified Instances	49	96.0784 %
Incorrectly Classified Instances	2	3.9216 %
Kappa statistic	0.9408	
Mean absolute error	0.0396	
Root mean squared error	0.1579	
Relative absolute error	8.8979 %	
Root relative squared error	33.4091 %	
Total Number of Instances	51	

=== Detailed Accuracy By Class ===

TP Rate	FP Rate	Precision	Recall	F-Measure	Class
1	0	1	1	1	Iris-setosa
1	0.063	0.905	1	0.95	Iris-versicolor
0.882	0	1	0.882	0.938	Iris-virginica

=== Confusion Matrix ===

a	b	c	<-- classified as
15	0	0	a = Iris-setosa
0	19	0	b = Iris-versicolor
0	2	15	c = Iris-virginica

Status

OK

Log

x 0

Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose J48 -C 0.25 -M 2

Test options

☐ Use training set☐ Supplied test set Set...☐ Cross-validation Folds 10☒ Percentage split % 66

More options...

(Nom) class

Start

Stop

Result list (right-click for options)

11:49:05 - trees.j48.J48

Classifier output

Time taken to build model: 0.24 seconds

=== Evaluation on test split ===
=== Summary ===

Correctly Classified Instances	49	96.0784 %
Incorrectly Classified Instances	2	3.9216 %
Kappa statistic	0.9408	
Mean absolute error	0.0396	
Root mean squared error	0.1579	
Relative absolute error	8.8979 %	
Root relative squared error	33.4091 %	
Total Number of Instances	51	

=== Detailed Accuracy By Class ===

TP Rate	FP Rate	Precision	Recall	F-Measure	Class
1	0	1	1	1	Iris-setosa
1	0.063	0.905	1	0.95	Iris-versicolor
0.882	0	1	0.882	0.938	Iris-virginica

=== Confusion Matrix ===

a	b	c	<-- classified as
15	0	0	a = Iris-setosa
0	19	0	b = Iris-versicolor
0	2	15	c = Iris-virginica

Status

OK

Log

x 0

Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose J48 -C 0.25 -M 2

Test options

☐ Use training set☐ Supplied test set

Set...

☐ Cross-validation Folds 10☒ Percentage split % 66

More options...

(Nom) class

Start

Stop

Result list (right-click for options)

11:49:05 - trees.j48.J48

View in main window

View in separate window

Save result buffer

Load model

Save model

Re-evaluate model on current test set

Visualize classifier errors

Visualize tree

Visualize margin curve

Visualize threshold curve

Visualize cost curve

Classifier output

Time taken to build model: 0.24 seconds

=== Evaluation on test split ===

=== Summary ===

Correctly Classified Instances	49	96.0784 %
Incorrectly Classified Instances	2	3.9216 %
Kappa statistic	0.9408	
Mean absolute error	0.0396	
Root mean squared error	0.1579	
Relative absolute error	8.8979 %	
Root relative squared error	33.4091 %	
Total Number of Instances	51	

=== Detailed Accuracy By Class ===

Recall	F-Measure	Class
1	1	Iris-setosa
1	0.95	Iris-versicolor
0.882	0.938	Iris-virginica

Status

OK

Log

x 0

Weka Knowledge Explorer

Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose

J48 - C 0.25 - M 2

Test options

Use training set

Supplied test set

Cross-validation

Percentage split

More options

(Nom) class

Start

Result list (right-click for details)

11:49:05 - trees.j48.J48

Weka Classifier Tree Visualizer: 11:49:05 - trees.j48.J48 (iris)

Tree View

petalwidth

<= 0.6

Iris-setosa (50.0)

> 0.6

petalwidth

<= 1.7

petallength

<= 4.9

Iris-versicolor (48.0/1.0)

> 4.9

petalwidth

<= 1.5

Iris-virginica (3.0)

> 1.5

Iris-versicolor (3.0/1.0)

> 1.7

Iris-virginica (46.0/1.0)

96.0784 %

3.9216 %

ass

is-setosa

is-versicolor

is-virginica

15 0 0 | a = Iris-setosa

0 19 0 | b = Iris-versicolor

0 2 15 | c = Iris-virginica

Status

OK

Log

 x 0



Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose J48 -C 0.25 -M 2

Test options

☐ Use training set☐ Supplied test set

Set...

☐ Cross-validation Folds 10☒ Percentage split % 66

More options...

(Nom) class

Start

Stop

Result list (right-click for options)

11:49:05 - trees.j48.J48

View in main window

View in separate window

Save result buffer

Load model

Save model

Re-evaluate model on current test set

Visualize classifier errors

Visualize tree

Visualize margin curve

Visualize threshold curve

Visualize cost curve

Classifier output

Time taken to build model: 0.24 seconds

=== Evaluation on test split ===

=== Summary ===

Correctly Classified Instances	49	96.0784 %
Incorrectly Classified Instances	2	3.9216 %
Kappa statistic	0.9408	
Mean absolute error	0.0396	
Root mean squared error	0.1579	
Relative absolute error	8.8979 %	
Root relative squared error	33.4091 %	
Total Number of Instances	51	

=== Detailed Accuracy By Class ===

Recall	F-Measure	Class
1	1	Iris-setosa
1	0.95	Iris-versicolor
0.882	0.938	Iris-virginica

lor
ca

Status

OK

Log

x 0

Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose J48 -C 0.25 -M 2

Test options

- ☐ Use training set
- ☐ Supplied test set
- ☐ Cross-validation
- ☒ Percentage split

More options

(Nom) class

Start

Result list (right-click for)

11:49:05 - trees.j48.J

Weka Classifier Visualize: 11:49:05 - trees.j48.J48 (iris)

X: petallength (Num)

Y: petalwidth (Num)

Colour: class (Nom)

Select Instance

Reset

Clear

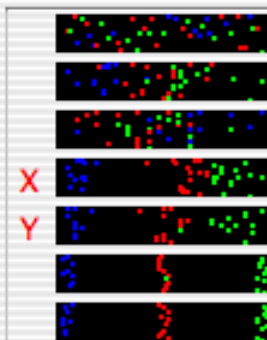
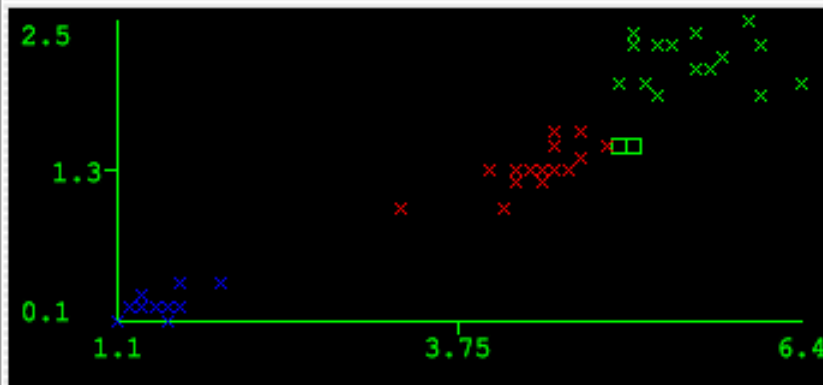
Save

Jitter

96.0784 %

3.9216 %

Plot: iris_predicted



Class colour

Iris-setosa Iris-versicolor Iris-virginica

```
0 15 0 | D = Iris-versicolor
0 2 15 | c = Iris-virginica
```

ass
is-setosa
is-versicolor
is-virginica

Status

OK

Log

x 0

Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose J48 -C 0.25 -M 2

Test options

☐ Use training set☐ Supplied test set

Set...

☐ Cross-validation Folds 10☒ Percentage split % 66

More options...

(Nom) class

Start

Stop

Result list (right-click for options)

11:49:05 - trees.j48.J48

Classifier output

Time taken to build model: 0.24 seconds

=== Evaluation on test split ===

=== Summary ===

Correctly Classified Instances	49	96.0784 %
Incorrectly Classified Instances	2	3.9216 %
Kappa statistic	0.9408	
Mean absolute error	0.0396	
Root mean squared error	0.1579	
Relative absolute error	8.8979 %	
Root relative squared error	33.4091 %	
Total Number of Instances	51	

=== Detailed Accuracy By Class ===

TP Rate	FP Rate	Precision	Recall	F-Measure	Class
1	0	1	1	1	Iris-setosa
1	0.063	0.905	1	0.95	Iris-versicolor
0.882	0	1	0.882	0.938	Iris-virginica

=== Confusion Matrix ===

a	b	c	<-- classified as
15	0	0	a = Iris-setosa
0	19	0	b = Iris-versicolor
0	2	15	c = Iris-virginica

Status

OK

Log

x 0

Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose J48 -C 0.25 -M 2

Test options

☐ Use training set☐ Supplied test set Set...☐ Cross-validation Folds 10☒ Percentage split % 66

More options...

(Nom) class

Start

Stop

Result list (right-click for options)

11:49:05 - trees.j48.J48

Classifier output

Time taken to build model: 0.24 seconds

=== Evaluation on test split ===

=== Summary ===

Correctly Classified Instances	49	96.0784 %
Incorrectly Classified Instances	2	3.9216 %
Kappa statistic	0.9408	
Mean absolute error	0.0396	
Root mean squared error	0.1579	
Relative absolute error	8.8979 %	
Root relative squared error	33.4091 %	
Total Number of Instances	51	

=== Detailed Accuracy By Class ===

TP Rate	FP Rate	Precision	Recall	F-Measure	Class
1	0	1	1	1	Iris-setosa
1	0.063	0.905	1	0.95	Iris-versicolor
0.882	0	1	0.882	0.938	Iris-virginica

=== Confusion Matrix ===

a	b	c	<-- classified as
15	0	0	a = Iris-setosa
0	19	0	b = Iris-versicolor
0	2	15	c = Iris-virginica

Status

OK

Log

x 0



Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

- weka
 - classifiers
 - bayes
 - functions
 - LeastMedSq
 - LinearRegression
 - Logistic
 - neural
 - NeuralNetwork
 - pace
 - supportVector
 - SimpleLinearRegression
 - SimpleLogistic
 - VotedPerceptron
 - Winnow
 - lazy
 - meta
 - misc
 - trees
 - rules

output

Time taken to build model: 0.24 seconds

Evaluation on test split ===

Summary ===

Correctly Classified Instances	49	96.0784 %
Incorrectly Classified Instances	2	3.9216 %
Kappa statistic	0.9408	
Mean absolute error	0.0396	
Mean squared error	0.1579	
Root mean squared error	8.8979 %	
Relative squared error	33.4091 %	
Number of Instances	51	

Detailed Accuracy By Class ===

	FP Rate	Precision	Recall	F-Measure	Class
1	0	1	1	1	Iris-setosa
2	0.063	0.905	1	0.95	Iris-versicolor
3	0	1	0.882	0.938	Iris-virginica

Confusion Matrix ===

```
a b c <-- classified as
15 0 0 | a = Iris-setosa
0 19 0 | b = Iris-versicolor
0 2 15 | c = Iris-virginica
```

Status

OK

Log



Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose NeuralNetwork -L 0.3 -M 0.2 -N 500 -V 0 -S 0 -E 20 -H a

Test options

☐ Use training set☐ Supplied test set

Set...

☐ Cross-validation Folds 10☒ Percentage split % 66

More options...

(Nom) class

Start

Stop

Result list (right-click for options)

11:49:05 - trees.j48.J48

Classifier output

=== Evaluation on test split ===

=== Summary ===

Correctly Classified Instances	49	96.0784 %
Incorrectly Classified Instances	2	3.9216 %
Kappa statistic	0.9408	
Mean absolute error	0.0396	
Root mean squared error	0.1579	
Relative absolute error	8.8979 %	
Root relative squared error	33.4091 %	
Total Number of Instances	51	

=== Detailed Accuracy By Class ===

TP Rate	FP Rate	Precision	Recall	F-Measure	Class
1	0	1	1	1	Iris-setosa
1	0.063	0.905	1	0.95	Iris-versicolor
0.882	0	1	0.882	0.938	Iris-virginica

=== Confusion Matrix ===

```

a b c <-- classified as
15 0 0 | a = Iris-setosa
0 19 0 | b = Iris-versicolor
0 2 15 | c = Iris-virginica

```

Status

OK

Log

x 0

Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose NeuralNetwork -L 0.3 -M 0.2 -N 500 -V 0 -S 0 -E 20 -H a

Test options

☐ Use training set☐ Supplied test set

Set...

☐ Cross-validation Folds 10☒ Percentage split % 66

More options...

(Nom) class

Start

Stop

Result list (right-click for options)

11:49:05 - trees.j48.J48

Classifier output

=== Evaluation on test split ===

=== Summary ===

Correctly Classified Instances	49	96.0784 %
Incorrectly Classified Instances	2	3.9216 %
Kappa statistic	0.9408	
Mean absolute error	0.0396	
Root mean squared error	0.1579	
Relative absolute error	8.8979 %	
Root relative squared error	33.4091 %	
Total Number of Instances	51	

=== Detailed Accuracy By Class ===

TP Rate	FP Rate	Precision	Recall	F-Measure	Class
1	0	1	1	1	Iris-setosa
1	0.063	0.905	1	0.95	Iris-versicolor
0.882	0	1	0.882	0.938	Iris-virginica

=== Confusion Matrix ===

```

a  b  c  <-- classified as
15  0  0 | a = Iris-setosa
 0 19  0 | b = Iris-versicolor
 0  2 15 | c = Iris-virginica

```

Status

OK

Log

x 0

Preprocess

Classify

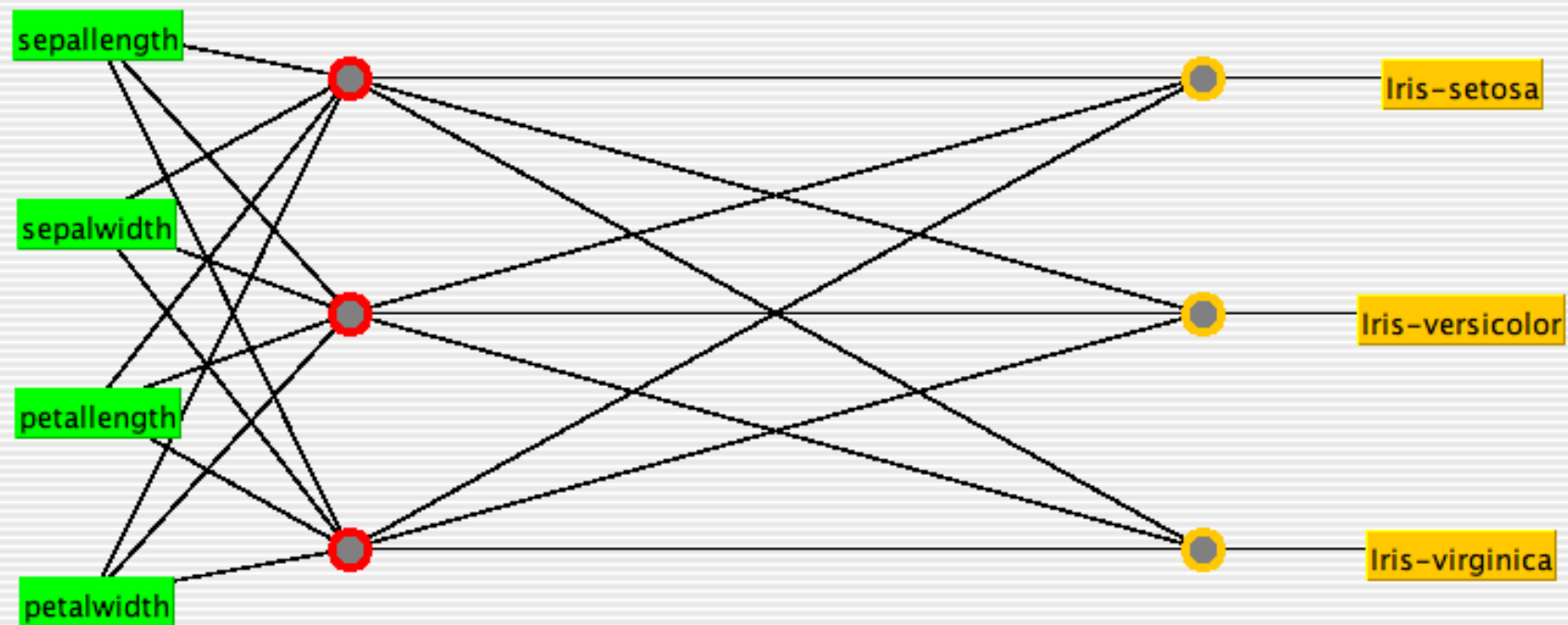
Cluster

Associate

Select attributes

Visualize

Neural Network



Controls

Start

Epoch 0

Num Of Epochs 500

Accept

Error per Epoch = 0

Learning Rate = 0.3

Momentum = 0.2

building model on training data...

Preprocess

Classify

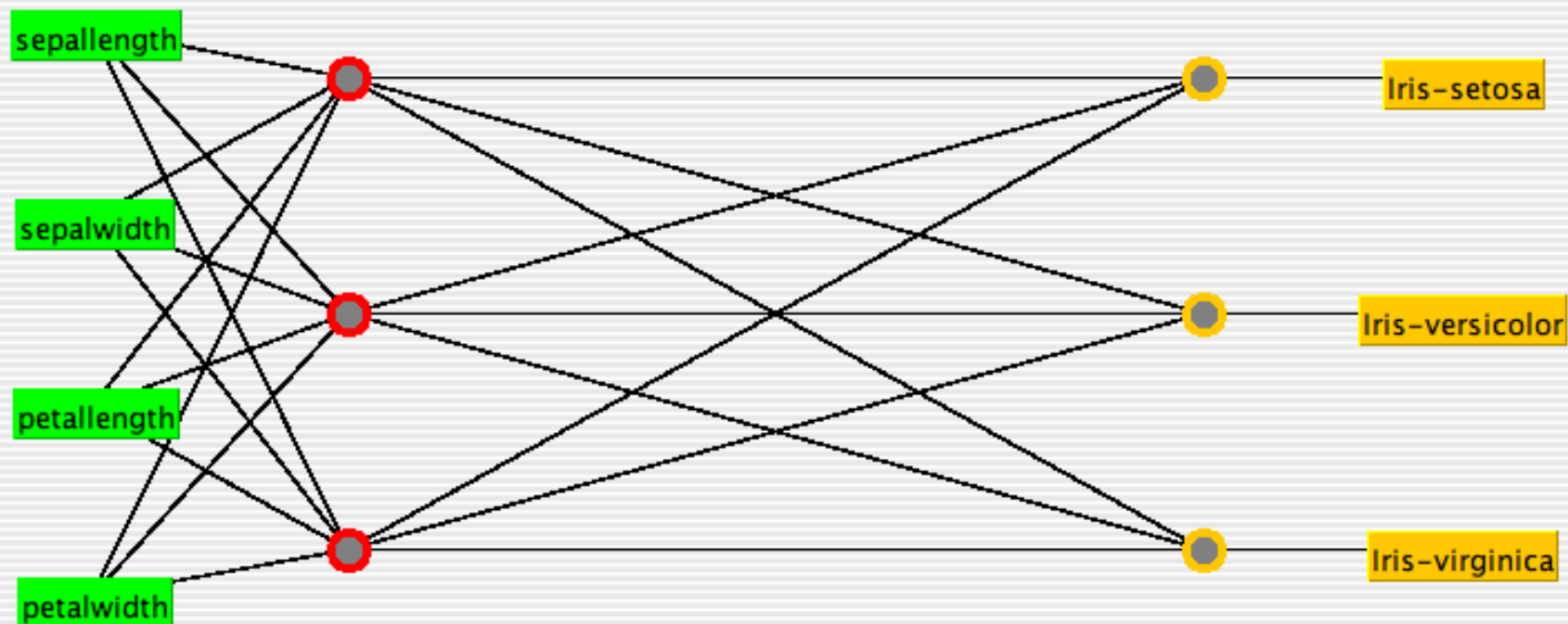
Cluster

Associate

Select attributes

Visualize

Neural Network



Controls

Start

Epoch 0

Num Of Epochs 500

Accept

Error per Epoch = 0

Learning Rate = 0.3

Momentum = 0.2

building model on training data...

Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose NeuralNetwork -L 0.3 -M 0.2 -N 500 -V 0 -S 0 -E 20 -H a -G -R

Test options

☐ Use training set☐ Supplied test set Set...☐ Cross-validation Folds 10☒ Percentage split % 66

More options...

(Nom) class

Start

Stop

Result list (right-click for options)

11:49:05 - trees.j48.J48

14:34:28 - functions.neural.NeuralNetwork

Classifier output

=== Evaluation on test split ===

=== Summary ===

Correctly Classified Instances	50	98.0392 %
Incorrectly Classified Instances	1	1.9608 %
Kappa statistic	0.9704	
Mean absolute error	0.0239	
Root mean squared error	0.1101	
Relative absolute error	5.3594 %	
Root relative squared error	23.2952 %	
Total Number of Instances	51	

=== Detailed Accuracy By Class ===

TP Rate	FP Rate	Precision	Recall	F-Measure	Class
1	0	1	1	1	Iris-setosa
1	0.031	0.95	1	0.974	Iris-versicolor
0.941	0	1	0.941	0.97	Iris-virginica

=== Confusion Matrix ===

```

a b c  <-- classified as
15 0 0 | a = Iris-setosa
0 19 0 | b = Iris-versicolor
0 1 16 | c = Iris-virginica

```

Status

OK

Log

x 0

Preprocess

Classify

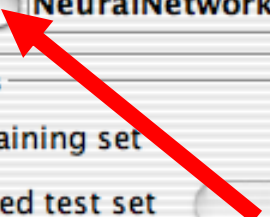
Cluster

Associate

Select attributes

Visualize

Classifier

Choose NeuralNetwork -L 0.3 -M 0.2 -N 500 -V 0 -S 0 -E 20 -H a -G -R

Test options

☐ Use training set☐ Supplied test set Set...☐ Cross-validation Folds 10☒ Percentage split % 66

More options...

(Nom) class

Start

Stop

Result list (right-click for options)

11:49:05 - trees.j48.J48

14:34:28 - functions.neural.NeuralNetwork

Classifier output

=== Evaluation on test split ===

=== Summary ===

Correctly Classified Instances	50	98.0392 %
Incorrectly Classified Instances	1	1.9608 %
Kappa statistic	0.9704	
Mean absolute error	0.0239	
Root mean squared error	0.1101	
Relative absolute error	5.3594 %	
Root relative squared error	23.2952 %	
Total Number of Instances	51	

=== Detailed Accuracy By Class ===

TP Rate	FP Rate	Precision	Recall	F-Measure	Class
1	0	1	1	1	Iris-setosa
1	0.031	0.95	1	0.974	Iris-versicolor
0.941	0	1	0.941	0.97	Iris-virginica

=== Confusion Matrix ===

a	b	c	<-- classified as
15	0	0	a = Iris-setosa
0	19	0	b = Iris-versicolor
0	1	16	c = Iris-virginica

Status

OK

Log

 x 0

Weka Knowledge Explorer

Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

weka

classifiers

bayes

AODE

BayesNetK2

BayesNetB

NaiveBayes

NaiveBayesMultinomial

NaiveBayesSimple

NaiveBayesUpdateable

functions

lazy

meta

misc

trees

rules

Classifier output

== Evaluation on test split ==

== Summary ==

Correctly Classified Instances

Incorrectly Classified Instances

Kappa statistic

Mean absolute error

Root mean squared error

Relative absolute error

Root relative squared error

Total Number of Instances

50

1

0.9704

0.0239

0.1101

5.3594 %

23.2952 %

51

98.0392 %

1.9608 %

== Detailed Accuracy By Class ==

TP Rate	FP Rate	Precision	Recall	F-Measure	Class
1	0	1	1	1	Iris-setosa
1	0.031	0.95	1	0.974	Iris-versicolor
0.941	0	1	0.941	0.97	Iris-virginica

== Confusion Matrix ==

a	b	c	<-- classified as
15	0	0	a = Iris-setosa
0	19	0	b = Iris-versicolor
0	1	16	c = Iris-virginica

QuickTime™ and a TIFF (LZW) decompressor are needed to see this picture.

Log

 x 0

Weka Knowledge Explorer

Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose

NaiveBayes

Test options

☐ Use training set

☐ Supplied test set

☐ Cross-validation

☒ Percentage split

Set...

Folds

%

10

66

More options...

(Nom) class

Start

Stop

Result list (right-click for options)

11:49:05 - trees.j48.J48

14:34:28 - functions.neural.NeuralNetwork

Classifier output

=== Evaluation on test split ===

=== Summary ===

Correctly Classified Instances	50	98.0392 %
Incorrectly Classified Instances	1	1.9608 %
Kappa statistic	0.9704	
Mean absolute error	0.0239	
Root mean squared error	0.1101	
Relative absolute error	5.3594 %	
Root relative squared error	23.2952 %	
Total Number of Instances	51	

=== Detailed Accuracy By Class ===

TP Rate	FP Rate	Precision	Recall	F-Measure	Class
1	0	1	1	1	Iris-setosa
1	0.031	0.95	1	0.974	Iris-versicolor
0.941	0	1	0.941	0.97	Iris-virginica

=== Confusion Matrix ===

a b c <-- classified as

15 0 0 | a = Iris-setosa

0 19 0 | b = Iris-versicolor

0 1 16 | c = Iris-virginica

QuickTime™ and a TIFF (LZW) decompressor are needed to see this picture.

Log

 x 0

Weka Knowledge Explorer

Preprocess **Classify** Cluster Associate Select attributes Visualize

Classifier

Choose **NaiveBayes**

Test options

☐ Use training set

☐ Supplied test set Set...

☐ Cross-validation Folds 10

☒ Percentage split % 66

More options...

(Nom) class

Start Stop

Result list (right-click for options)

11:49:05 - trees.j48.J48

14:34:28 - functions.neural.NeuralNetwork

Classifier output

=== Evaluation on test split ===

=== Summary ===

Correctly Classified Instances	50	98.0392 %
Incorrectly Classified Instances	1	1.9608 %
Kappa statistic	0.9704	
Mean absolute error	0.0239	
Root mean squared error	0.1101	
Relative absolute error	5.3594 %	
Root relative squared error	23.2952 %	
Total Number of Instances	51	

=== Detailed Accuracy By Class ===

TP Rate	FP Rate	Precision	Recall	F-Measure	Class
1	0	1	1	1	Iris-setosa
1	0.031	0.95	1	0.974	Iris-versicolor
0.941	0	1	0.941	0.97	Iris-virginica

=== Confusion Matrix ===

a	b	c	<-- classified as
15	0	0	a = Iris-setosa
0	19	0	b = Iris-versicolor
0	1	16	c = Iris-virginica

Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose NaiveBayes

Test options

☐ Use training set☐ Supplied test set Set...☐ Cross-validation Folds 10☒ Percentage split % 66

More options...

(Nom) class

Start

Stop

Result list (right-click for options)

11:49:05 - trees.j48.J48

14:34:28 - functions.neural.NeuralNetwork

14:48:05 - bayes.NaiveBayes

Classifier output

=== Evaluation on test split ===

=== Summary ===

Correctly Classified Instances	48	94.1176 %
Incorrectly Classified Instances	3	5.8824 %
Kappa statistic	0.9113	
Mean absolute error	0.0447	
Root mean squared error	0.1722	
Relative absolute error	10.0365 %	
Root relative squared error	36.4196 %	
Total Number of Instances	51	

=== Detailed Accuracy By Class ===

TP Rate	FP Rate	Precision	Recall	F-Measure	Class
1	0	1	1	1	Iris-setosa
0.947	0.063	0.9	0.947	0.923	Iris-versicolor
0.882	0.029	0.938	0.882	0.909	Iris-virginica

=== Confusion Matrix ===

```

a b c  <-- classified as
15 0 0 | a = Iris-setosa
0 18 1 | b = Iris-versicolor
0 2 15 | c = Iris-virginica

```

Status

OK

Log

x 0

Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose NaiveBayes

Test options

☐ Use training set☐ Supplied test set Set...☐ Cross-validation Folds 10☒ Percentage split % 66

More options...

(Nom) class

Start

Stop

Result list (right-click for options)

11:49:05 - trees.j48.J48

14:34:28 - functions.neural.NeuralNetwork

14:48:05 - bayes.NaiveBayes

Classifier output

=== Evaluation on test split ===

=== Summary ===

Correctly Classified Instances	48	94.1176 %
Incorrectly Classified Instances	3	5.8824 %
Kappa statistic	0.9113	
Mean absolute error	0.0447	
Root mean squared error	0.1722	
Relative absolute error	10.0365 %	
Root relative squared error	36.4196 %	
Total Number of Instances	51	

=== Detailed Accuracy By Class ===

TP Rate	FP Rate	Precision	Recall	F-Measure	Class
1	0	1	1	1	Iris-setosa
0.947	0.063	0.9	0.947	0.923	Iris-versicolor
0.882	0.029	0.938	0.882	0.909	Iris-virginica

=== Confusion Matrix ===

a	b	c	<-- classified as
15	0	0	a = Iris-setosa
0	18	1	b = Iris-versicolor
0	2	15	c = Iris-virginica

Status

OK

Log

x 0



Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose NaiveBayes

Test options

☐ Use training set☐ Supplied test set Set...☐ Cross-validation Folds 10☒ Percentage split % 66

More options...

(Nom) class

Start

Result list (right-click for)

11:49:05 - trees.j48.J

14:34:28 - functions.

14:48:05 - bayes.Nai

View in main window

View in separate window

Save result buffer

Load model

Save model

Re-evaluate model on current test set

Visualize classifier errors

Visualize tree

Visualize margin curve

Visualize threshold curve

Visualize cost curve

Classifier output

=== Evaluation on test split ===

=== Summary ===

Correctly Classified Instances	48	94.1176 %
Incorrectly Classified Instances	3	5.8824 %
Kappa statistic	0.9113	
Mean absolute error	0.0447	
Root mean squared error	0.1722	
Relative absolute error	10.0365 %	
Root relative squared error	36.4196 %	
Total Number of Instances	51	

=== Detailed Accuracy By Class ===

	Precision	Recall	F-Measure	Class
	1	1	1	Iris-setosa
	0.9	0.947	0.923	Iris-versicolor
	0.938	0.882	0.909	Iris-virginica

x ===

classified as
Iris-setosa
Iris-versicolor
Iris-virginica

Iris-setosa

Iris-versicolor

Iris-virginica

Status

OK

Log



Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose NaiveBayes

Test options

☐ Use training set☐ Supplied test set☐ Cross-validation For☒ Percentage split

More options

(Nom) class

Start

Result list (right-click for options)

11:49:05 - trees.j48.J48

14:34:28 - functions.neu

14:48:05 - bayes.NaiveBa

Weka Classifier Visualize: ThresholdCurve. Class value Iris-versicolor)

X: False Positive Rate (Num)

Y: True Positive Rate (Num)

Colour: Threshold (Num)

Select Instance

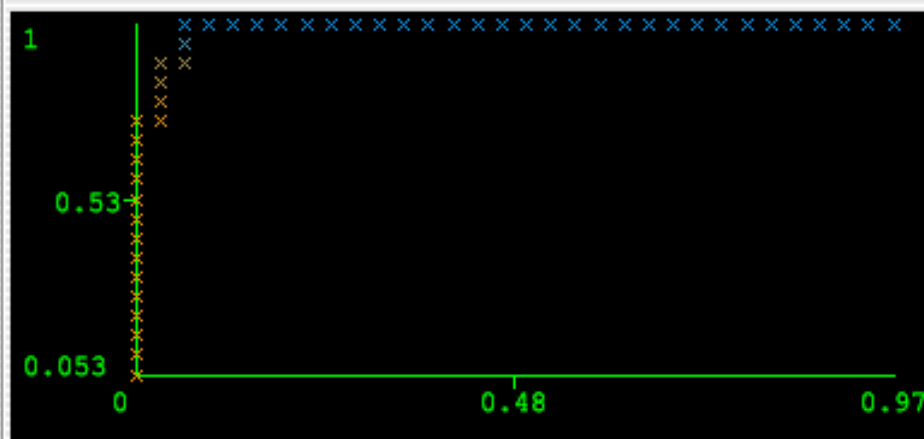
Reset

Clear

Save

Jitter

Plot: ThresholdCurve



Class colour

0

0.5

1

X

Y

osa
sicolor
ynica

Log

x 0

Status

OK

Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose NaiveBayes

Test options

☐ Use training set☐ Supplied test set Set...☐ Cross-validation Folds 10☒ Percentage split % 66

More options...

(Nom) class

Start

Stop

Result list (right-click for options)

11:49:05 - trees.j48.J48

14:34:28 - functions.neural.NeuralNetwork

14:48:05 - bayes.NaiveBayes

Classifier output

=== Evaluation on test split ===

=== Summary ===

Correctly Classified Instances	48	94.1176 %
Incorrectly Classified Instances	3	5.8824 %
Kappa statistic	0.9113	
Mean absolute error	0.0447	
Root mean squared error	0.1722	
Relative absolute error	10.0365 %	
Root relative squared error	36.4196 %	
Total Number of Instances	51	

=== Detailed Accuracy By Class ===

TP Rate	FP Rate	Precision	Recall	F-Measure	Class
1	0	1	1	1	Iris-setosa
0.947	0.063	0.9	0.947	0.923	Iris-versicolor
0.882	0.029	0.938	0.882	0.909	Iris-virginica

=== Confusion Matrix ===

```

a b c  <-- classified as
15 0 0 | a = Iris-setosa
0 18 1 | b = Iris-versicolor
0 2 15 | c = Iris-virginica

```

Status

OK

Log

x 0

Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose NaiveBayes

Test options

☐ Use training set☐ Supplied test set Set...☐ Cross-validation Folds 10☒ Percentage split % 66

More options...

(Nom) class

Start

Stop

Result list (right-click for options)

11:49:05 - trees.j48.J48

14:34:28 - functions.neural.NeuralNetwork

14:48:05 - bayes.NaiveBayes

Classifier output

=== Evaluation on test split ===

=== Summary ===

Correctly Classified Instances	48	94.1176 %
Incorrectly Classified Instances	3	5.8824 %
Kappa statistic	0.9113	
Mean absolute error	0.0447	
Root mean squared error	0.1722	
Relative absolute error	10.0365 %	
Root relative squared error	36.4196 %	
Total Number of Instances	51	

=== Detailed Accuracy By Class ===

TP Rate	FP Rate	Precision	Recall	F-Measure	Class
1	0	1	1	1	Iris-setosa
0.947	0.063	0.9	0.947	0.923	Iris-versicolor
0.882	0.029	0.938	0.882	0.909	Iris-virginica

=== Confusion Matrix ===

a	b	c	<-- classified as
15	0	0	a = Iris-setosa
0	18	1	b = Iris-versicolor
0	2	15	c = Iris-virginica

Status

OK

Log

x 0



Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

weka

- classifiers
 - bayes
 - functions
 - lazy
 - meta
 - misc
 - trees
 - adtree
 - DecisionStump
 - Id3
 - j48
 - lmt
 - m5
 - RandomForest
 - RandomTree
 - REPTree
 - UserClassifier
 - rules

Classifier output

== Evaluation on test split ==

== Summary ==

Correctly Classified Instances	48	94.1176 %
Incorrectly Classified Instances	3	5.8824 %
Kappa statistic	0.9113	
Mean absolute error	0.0447	
Root mean squared error	0.1722	
Relative absolute error	10.0365 %	
Root relative squared error	36.4196 %	
Total Number of Instances	51	

== Detailed Accuracy By Class ==

P Rate	FP Rate	Precision	Recall	F-Measure	Class
1	0	1	1	1	Iris-setosa
0.947	0.063	0.9	0.947	0.923	Iris-versicolor
0.882	0.029	0.938	0.882	0.909	Iris-virginica

== Confusion Matrix ==

a	b	c	<-- classified as
15	0	0	a = Iris-setosa
0	18	1	b = Iris-versicolor
0	2	15	c = Iris-virginica

Status

OK

Log

x 0

Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose

Test options

☐ Use training set☐ Supplied test set

Set...

☐ Cross-validation Folds 10☒ Percentage split % 66

More options...

(Nom) class

Start

Stop

Result list (right-click for options)

11:49:05 - trees.j48.J48

14:34:28 - functions.neural.NeuralNetwork

14:48:05 - bayes.NaiveBayes

Classifier output

=== Evaluation on test split ===

=== Summary ===

Correctly Classified Instances	48	94.1176 %
Incorrectly Classified Instances	3	5.8824 %
Kappa statistic	0.9113	
Mean absolute error	0.0447	
Root mean squared error	0.1722	
Relative absolute error	10.0365 %	
Root relative squared error	36.4196 %	
Total Number of Instances	51	

=== Detailed Accuracy By Class ===

TP Rate	FP Rate	Precision	Recall	F-Measure	Class
1	0	1	1	1	Iris-setosa
0.947	0.063	0.9	0.947	0.923	Iris-versicolor
0.882	0.029	0.938	0.882	0.909	Iris-virginica

=== Confusion Matrix ===

a	b	c	<-- classified as
15	0	0	a = Iris-setosa
0	18	1	b = Iris-versicolor
0	2	15	c = Iris-virginica

Status

OK

Log

x 0

Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose

UserClassifier

Test options

☐ Use training set☐ Supplied test set☐ Cross-validation For☒ Percentage split

More options

(Nom) class

Start

Result list (right-click for options)

11:49:05 - trees.j48.J48

14:34:28 - functions.neu

14:48:05 - bayes.NaiveBa

15:26:57 - trees.UserClas

Tree Visualizer

Data Visualizer

Tree View

[Iris-setosa, 50.0]
[Iris-versicolor, 50.0]
[Iris-virginica, 50.0]

Status

Building model on training data...

Log



x 1

Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose

UserClassifier

Test options

☐ Use training set☐ Supplied test set☐ Cross-validation☒ Percentage split

More...

(Nom) class

Start

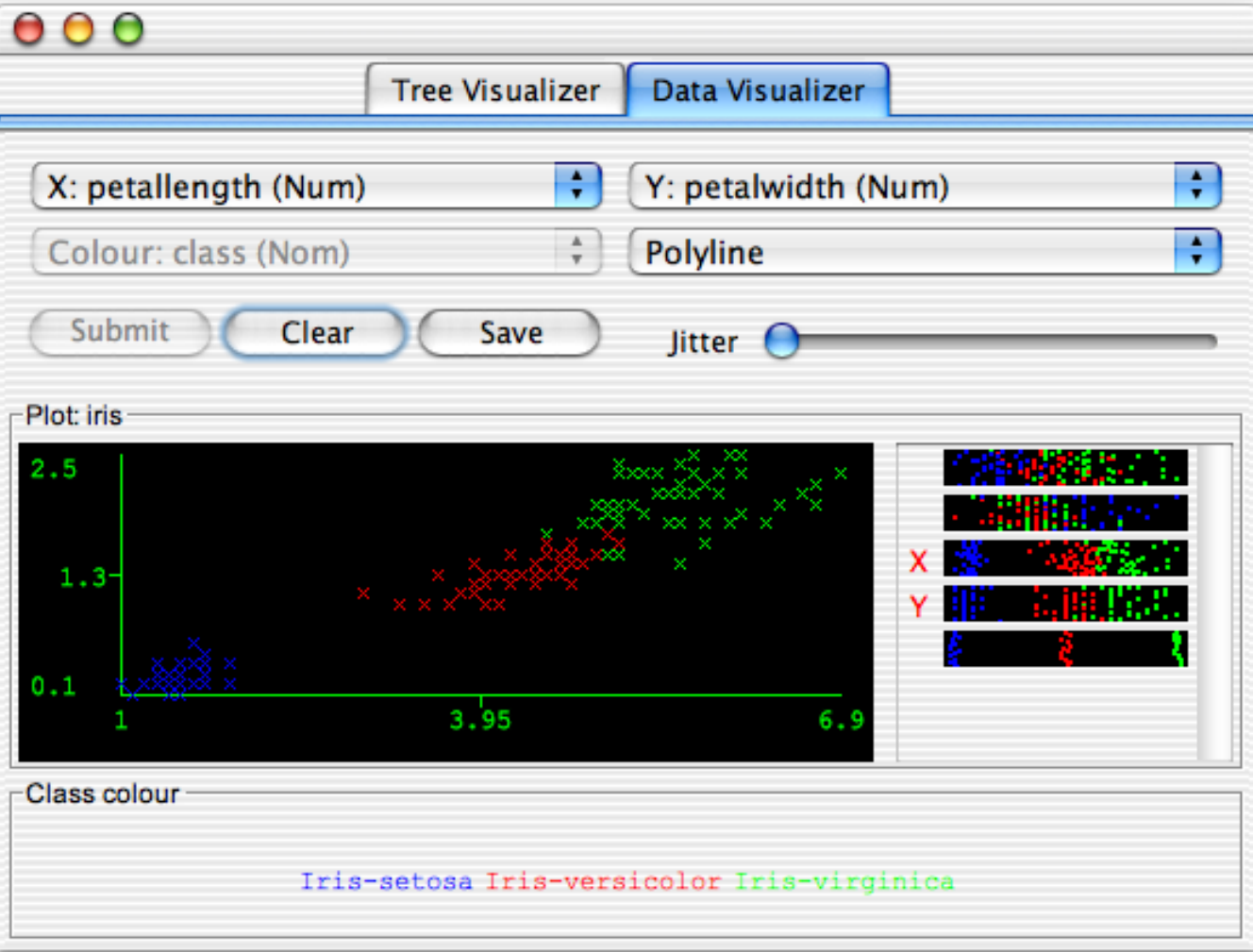
Result list (right-click for)

11:49:05 - trees.j48.

14:34:28 - functions

14:48:05 - bayes.Naï

15:26:57 - trees.Use



Status

Building model on training data...

Log



x 1

Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose

UserClassifier

Test options

☐ Use training set☐ Supplied test set☐ Cross-validation☒ Percentage split

More...

(Nom) class

Start

Result list (right-click for)

11:49:05 - trees.j48.

14:34:28 - functions

14:48:05 - bayes.Naï

15:26:57 - trees.Use



Tree Visualizer

Data Visualizer

X: petallength (Num)

Y: petalwidth (Num)

Colour: class (Nom)

Polyline

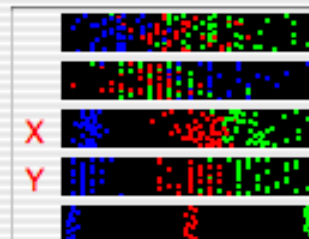
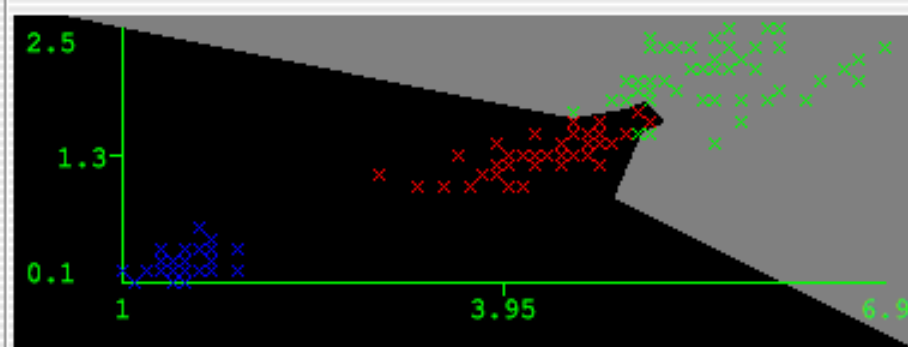
Submit

Clear

Save

Jitter

Plot: iris



Class colour

Iris-setosa Iris-versicolor Iris-virginica

Status

Building model on training data...

Log



x 1

Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose

UserClassifier

Test options

☐ Use training set☐ Supplied test set☐ Cross-validation☒ Percentage split

More options

(Nom) class

Start

Result list (right-click for context menu)

11:49:05 - trees.j48.J

14:34:28 - functions.

14:48:05 - bayes.Naïve

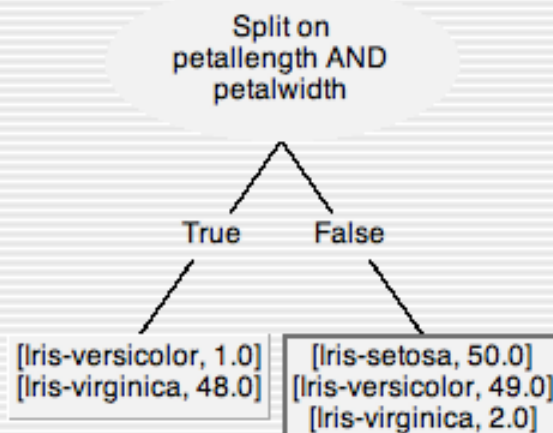
15:26:57 - trees.User



Tree Visualizer

Data Visualizer

Tree View



Status

Building model on training data...

Log



x 1



Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose

UserClassifier

Test options

☐ Use training set☐ Supplied test set

Set...

☐ Cross-validation Folds 10☒ Percentage split % 66

Classifier output

=== Evaluation on test split ===

=== Summary ===

Correctly Classified Instances	49	96.0784 %
Incorrectly Classified Instances	2	3.9216 %
Kappa statistic	0.9408	
Mean absolute error	0.0319	
Root mean squared error	0.1622	
Relative absolute error	7.1634 %	
Root relative squared error	34.312 %	
Total Number of Instances	51	

=== Detailed Accuracy By Class ===

TP Rate	FP Rate	Precision	Recall	F-Measure	Class
1	0	1	1	1	Iris-setosa
1	0.063	0.905	1	0.95	Iris-versicolor
0.882	0	1	0.882	0.938	Iris-virginica

=== Confusion Matrix ===

a	b	c	<-- classified as
15	0	0	a = Iris-setosa
0	19	0	b = Iris-versicolor
0	2	15	c = Iris-virginica

11:49:05 - trees.j48.J48

14:34:28 - functions.neural.NeuralNetwork

14:48:05 - bayes.NaiveBayes

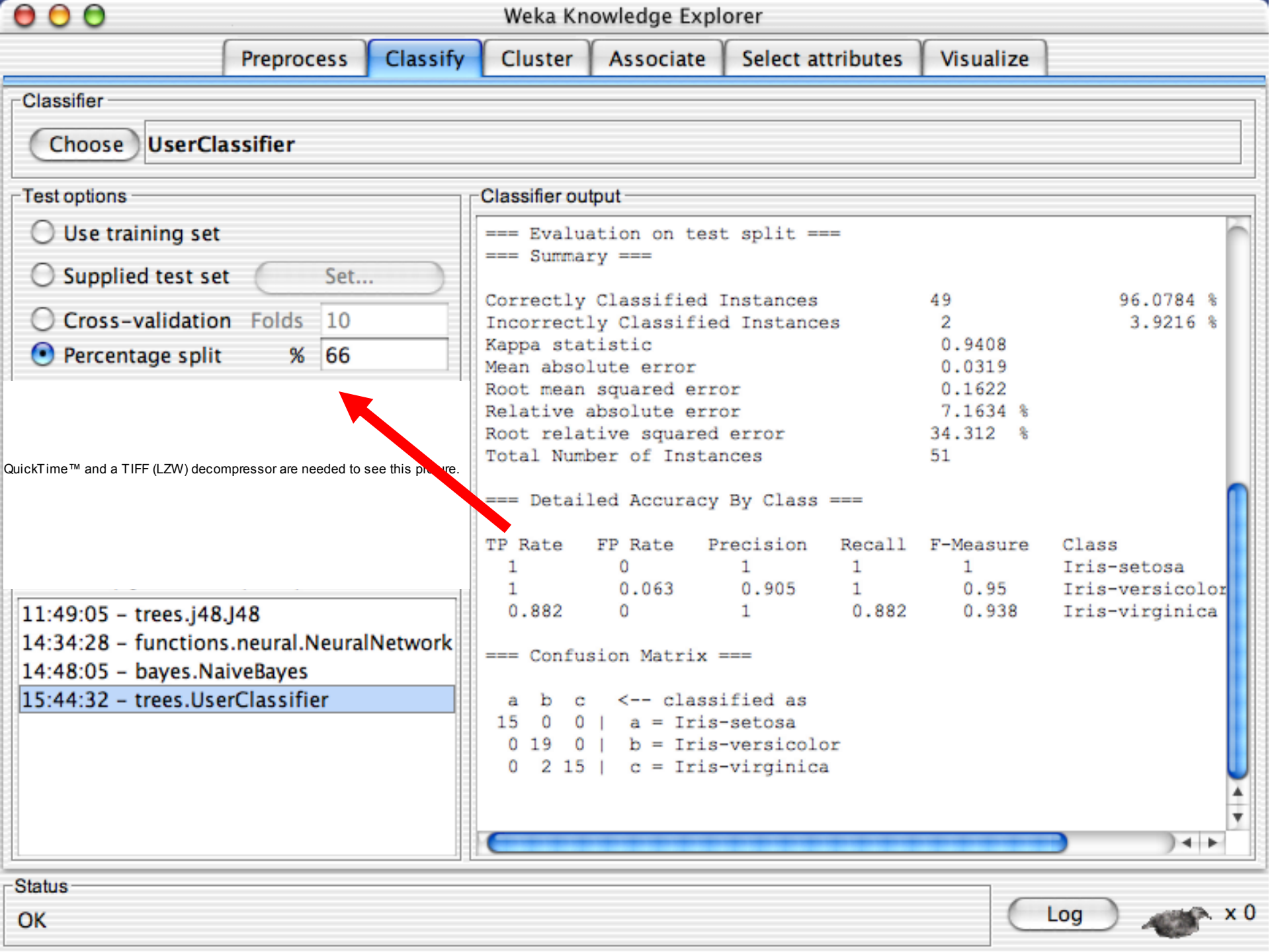
15:44:32 - trees.UserClassifier

Status

OK

Log

x 0



Classifier

Choose

UserClassifier

Test options

☐ Use training set☐ Supplied test set

Set...

☐ Cross-validation

Folds

10

☒ Percentage split

%

66

Classifier output

=== Evaluation on test split ===

=== Summary ===

Correctly Classified Instances	49	96.0784 %
Incorrectly Classified Instances	2	3.9216 %
Kappa statistic	0.9408	
Mean absolute error	0.0319	
Root mean squared error	0.1622	
Relative absolute error	7.1634 %	
Root relative squared error	34.312 %	
Total Number of Instances	51	

=== Detailed Accuracy By Class ===

TP Rate	FP Rate	Precision	Recall	F-Measure	Class
1	0	1	1	1	Iris-setosa
1	0.063	0.905	1	0.95	Iris-versicolor
0.882	0	1	0.882	0.938	Iris-virginica

=== Confusion Matrix ===

a	b	c	<-- classified as
15	0	0	a = Iris-setosa
0	19	0	b = Iris-versicolor
0	2	15	c = Iris-virginica

11:49:05 - trees.j48.J48

14:34:28 - functions.neural.NeuralNetwork

14:48:05 - bayes.NaiveBayes

15:44:32 - trees.UserClassifier

Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Classifier

Choose

UserClassifier

Test options

☐ Use training set☐ Supplied test set

Set...

☐ Cross-validation Folds 10☒ Percentage split % 66

(Num) sepallength

(Num) sepalwidth

✓ (Num) petallength

(Num) petalwidth

(Nom) class

Result list (right-click for options)

11:49:05 - trees.j48.J48

14:34:28 - functions.neural.NeuralNetwork

14:48:05 - bayes.NaiveBayes

15:44:32 - trees.UserClassifier

Classifier output

=== Evaluation on test split ===

=== Summary ===

Correctly Classified Instances	49	96.0784 %
Incorrectly Classified Instances	2	3.9216 %
Kappa statistic	0.9408	
Mean absolute error	0.0319	
Root mean squared error	0.1622	
Relative absolute error	7.1634 %	
Root relative squared error	34.312 %	
Total Number of Instances	51	

=== Detailed Accuracy By Class ===

TP Rate	FP Rate	Precision	Recall	F-Measure	Class
1	0	1	1	1	Iris-setosa
1	0.063	0.905	1	0.95	Iris-versicolor
0.882	0	1	0.882	0.938	Iris-virginica

=== Confusion Matrix ===

```

a b c  <-- classified as
15 0 0 | a = Iris-setosa
0 19 0 | b = Iris-versicolor
0 2 15 | c = Iris-virginica

```

Status

OK

Log

x 0

Explorar: clustering

- WEKA contiene “clusterers” para buscar grupos similares
- Algunos algoritmos:
 - *k*-Means, EM, Cobweb, X-means, FarthestFirst
- Clusters pueden ser visualizados y comparados con el correcto cluster (si es dado)



Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Clusterer

Choose

EM -I 100 -N -1 -S 100 -M 1.0E-6

Cluster mode

☒ Use training set☐ Supplied test set

Set...

☐ Percentage split

% 66

☐ Classes to clusters evaluation

(Nom) class

☒ Store clusters for visualization

Ignore attributes

Start

Stop

Result list (right-click for options)

Clusterer output

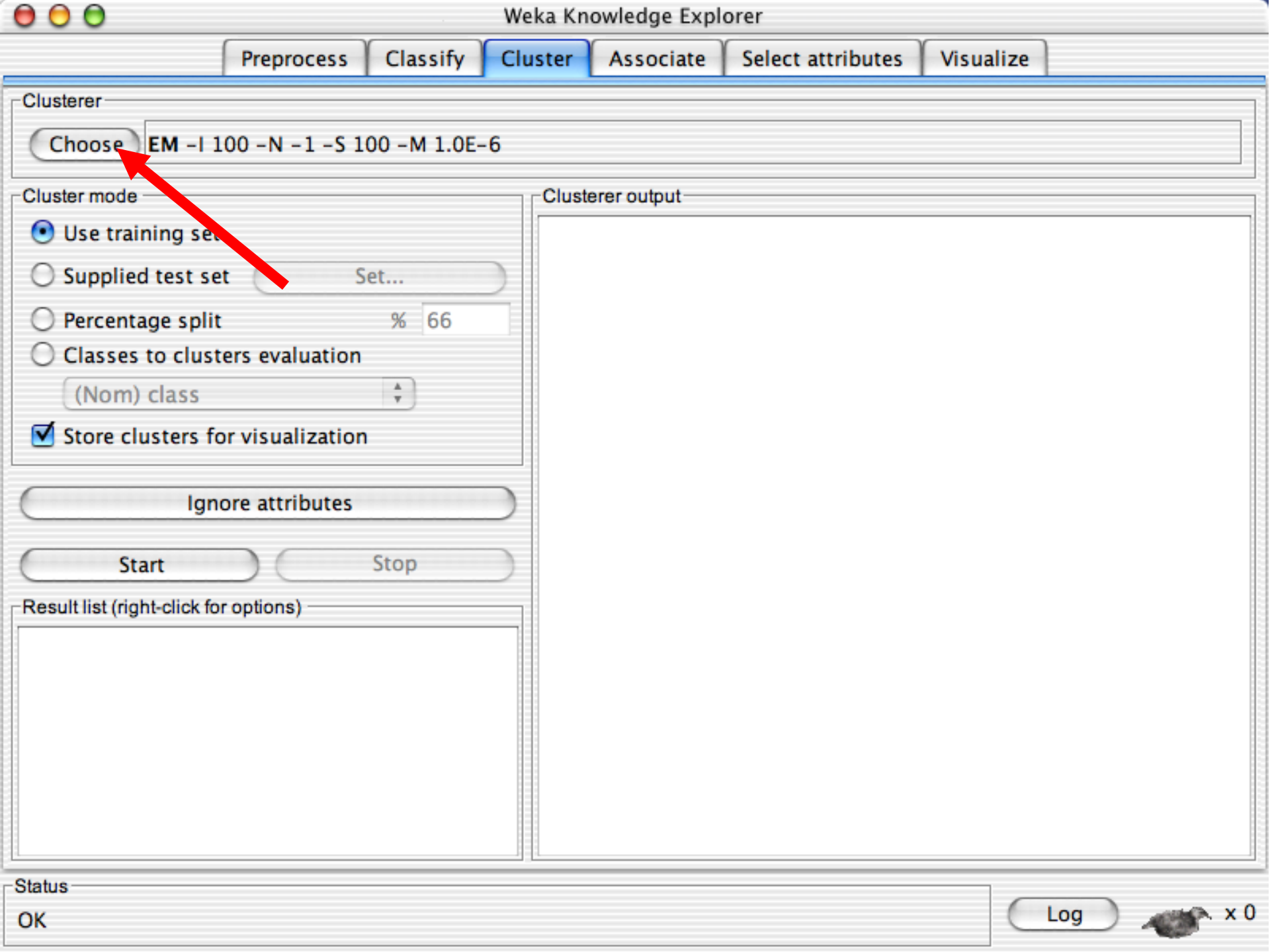
Status

OK

Log



x 0



Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Clusterer

- weka
 - clusters
 - EM
 - SimpleKMeans
 - Cobweb
 - FarthestFirst
 - XMeans

77387815

Clusterer output

Status

OK

Log



x 0



Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Clusterer

Choose

Cobweb -A 1.0 -C 0.0028209479177387815

Cluster mode

☒ Use training set☐ Supplied test set

Set...

☐ Percentage split

%

66

☐ Classes to clusters evaluation

(Nom) class

☒ Store clusters for visualization

Ignore attributes

Start

Stop

Result list (right-click for options)

Clusterer output

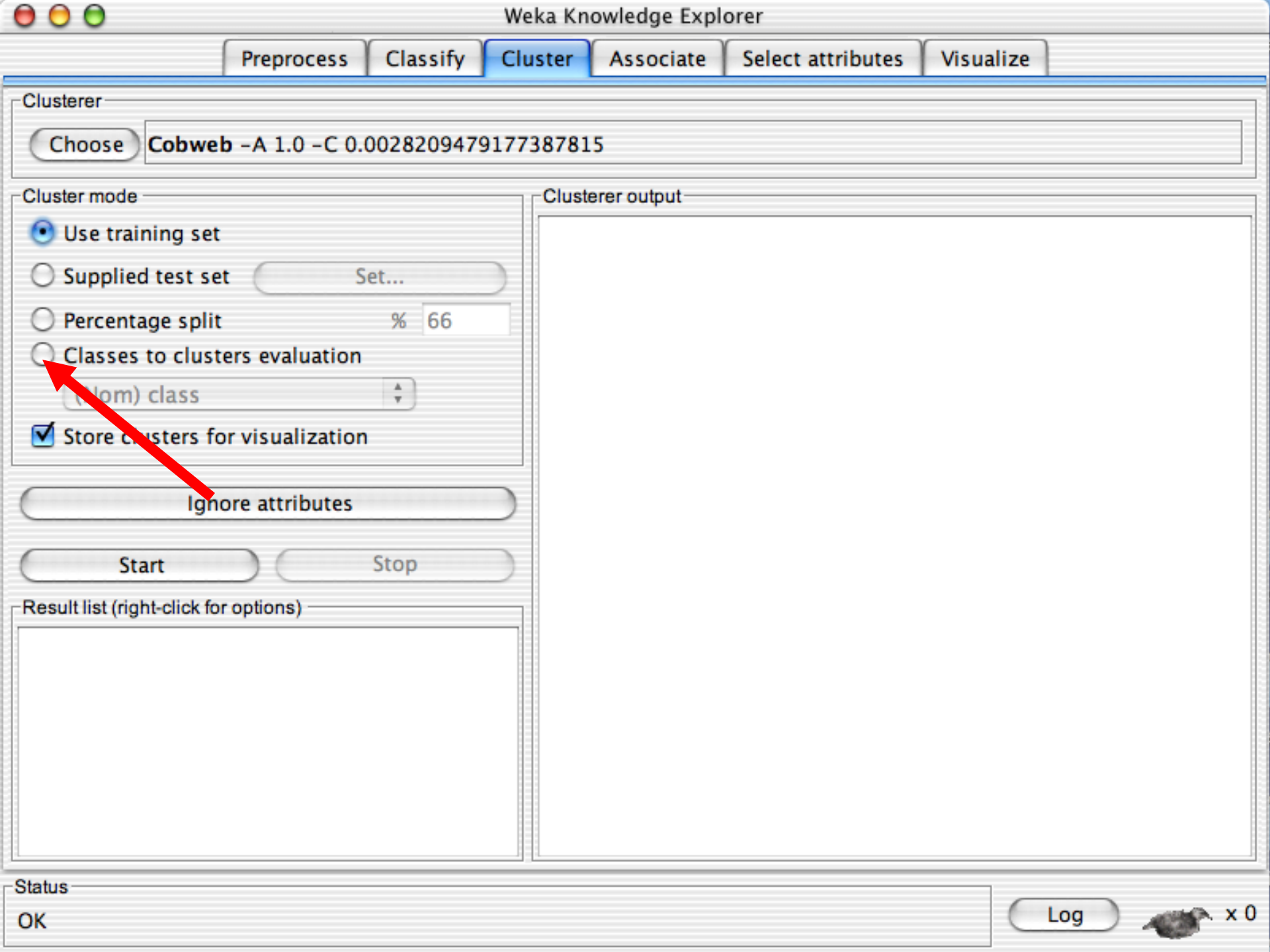
Status

OK

Log



x 0



Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Clusterer

Choose

Cobweb -A 1.0 -C 0.0028209479177387815

Cluster mode

☒ Use training set

☐ Supplied test set

Set...

☐ Percentage split

%

66

☐ Classes to clusters evaluation

(Nom) class

☒ Store clusters for visualization

Ignore attributes

Start

Stop

Result list (right-click for options)

Clusterer output

Status

OK

Log



x 0



Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Clusterer

Choose

Cobweb -A 1.0 -C 0.0028209479177387815

Cluster mode

☐ Use training set☐ Supplied test set

Set...

☐ Percentage split

% 66

☒ Classes to clusters evaluation

(Nom) class

☒ Store clusters for visualization

Ignore attributes

Start

Stop

Result list (right-click for options)

Clusterer output

Status

OK

Log



x 0

Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Clusterer

Choose

Cobweb -A 1.0 -C 0.0028209479177387815

Cluster mode

☐ Use training set☐ Supplied test set

Set...

☐ Percentage split

% 66

☒ Classes to clusters evaluation

(Nom) class

☒ Store clusters for visualization

Ignore attributes

Start

Stop

Result list (right-click for options)

Clusterer output

Log

x 0

Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Clusterer

Choose

Cobweb -A 1.0 -C 0.0028209479177387815

Cluster mode

☐ Use training set☐ Supplied test set

Set...

☐ Percentage split

% 66

☒ Classes to clusters evaluation

(Nom) class

☒ Store clusters for visualization

Ignore attributes

Start

Stop

Result list (right-click for options)

16:05:58 - Cobweb

Clusterer output

=== Run information ===

Scheme: weka.clusterers.Cobweb -A 1.0 -C 0.002820947917
Relation: iris
Instances: 150
Attributes: 5
sepallength
sepalwidth
petallength
petalwidth

Ignored:

class

Test mode: Classes to clusters evaluation on training data

=== Clustering model (full training set) ===

Number of merges: 0
Number of splits: 0
Number of clusters: 3

node 0 [150]
| leaf 1 [96]
node 0 [150]
| leaf 2 [54]

=== Evaluation on training set ===



Status

OK

Log

x 0



Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Clusterer

Choose

Cobweb -A 1.0 -C 0.0028209479177387815

Cluster mode

☐ Use training set☐ Supplied test set

Set...

☐ Percentage split

% 66

☒ Classes to clusters evaluation

(Nom) class

☒ Store clusters for visualization

Ignore attributes

Start

Stop

Result list (right-click for options)

16:05:58 - Cobweb

Clusterer output

=== Run information ===

Scheme: weka.clusterers.Cobweb -A 1.0 -C 0.002820947917
Relation: iris
Instances: 150
Attributes: 5
sepallength
sepalwidth
petallength
petalwidth

Ignored:

class

Test mode: Classes to clusters evaluation on training data

=== Clustering model (full training set) ===

Number of merges: 0
Number of splits: 0
Number of clusters: 3

node 0 [150]
| leaf 1 [96]
node 0 [150]
| leaf 2 [54]

=== Evaluation on training set ===

Log

x 0

Status

OK



Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Clusterer

Choose

Cobweb -A 1.0 -C 0.0028209479177387815

Cluster mode

☐ Use training set☐ Supplied test set

Set...

☐ Percentage split

% 66

☒ Classes to clusters evaluation

(Nom) class

☒ Store clusters for visualization

Ignore attributes

Start

Stop

Result list (right-click for options)

16:05:58 - Cobweb

Clusterer output

Number of clusters: 3

node 0 [150]

| leaf 1 [96]

node 0 [150]

| leaf 2 [54]

Clustered Instances

1 100 (67%)

2 50 (33%)

Class attribute: class

Classes to Clusters:

1 2 <-- assigned to cluster

0 50 | Iris-setosa

50 0 | Iris-versicolor

50 0 | Iris-virginica

Cluster 1 <-- Iris-versicolor

Cluster 2 <-- Iris-setosa

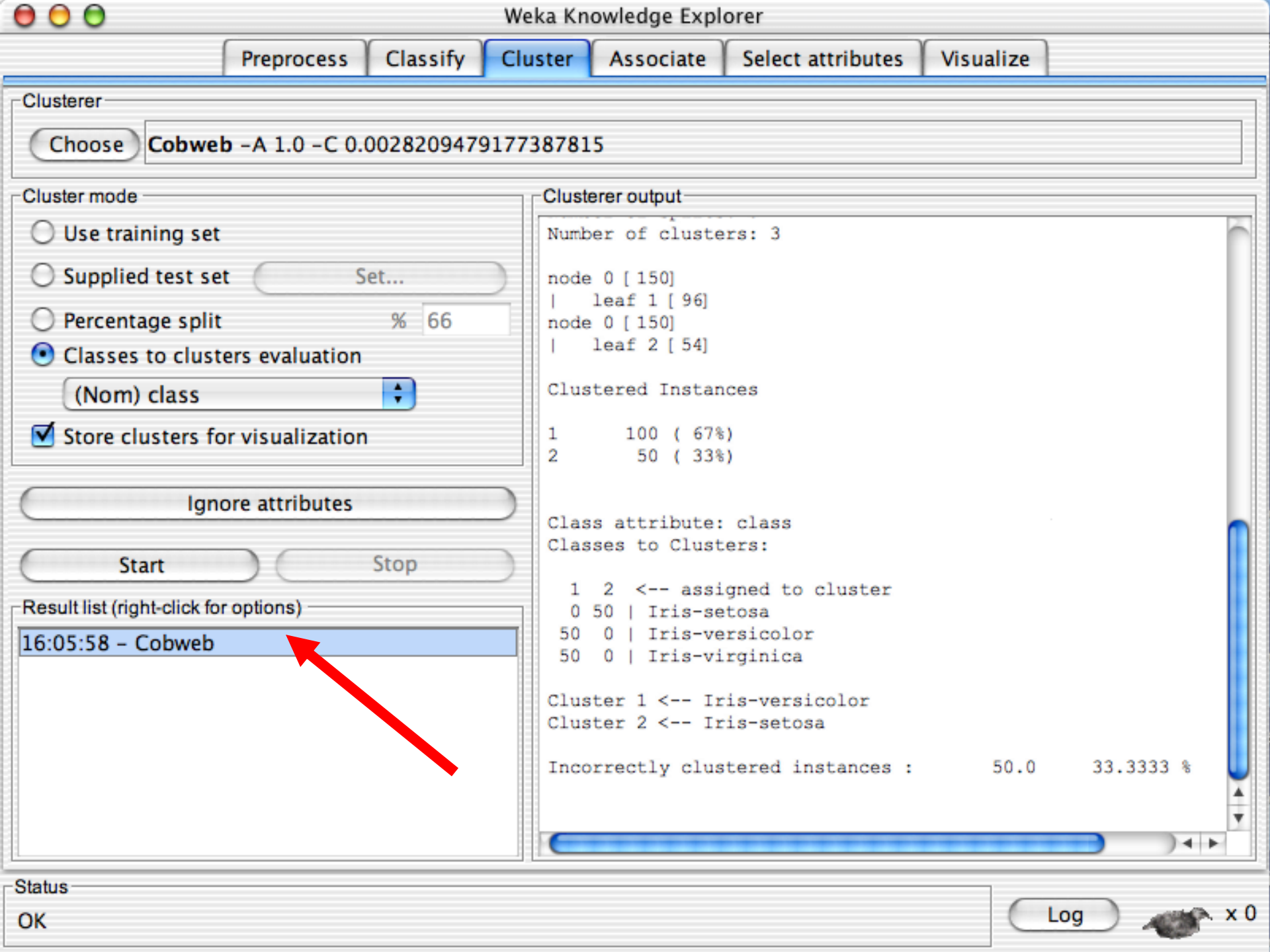
Incorrectly clustered instances : 50.0 33.3333 %

Status

OK

Log

 x 0



Clusterer
Choose **Cobweb -A 1.0 -C 0.0028209479177387815**

Cluster mode

☐ Use training set

☐ Supplied test set Set...

☐ Percentage split % 66

☒ Classes to clusters evaluation

(Nom) class

☒ Store clusters for visualization

Ignore attributes

Start Stop

- Result list (right-click for options)
- 16:05:58 - Cobweb

Clusterer output

Number of clusters: 3

node 0 [150]
| leaf 1 [96]
node 0 [150]
| leaf 2 [54]

Clustered Instances

1	100 (67%)
2	50 (33%)

Class attribute: class

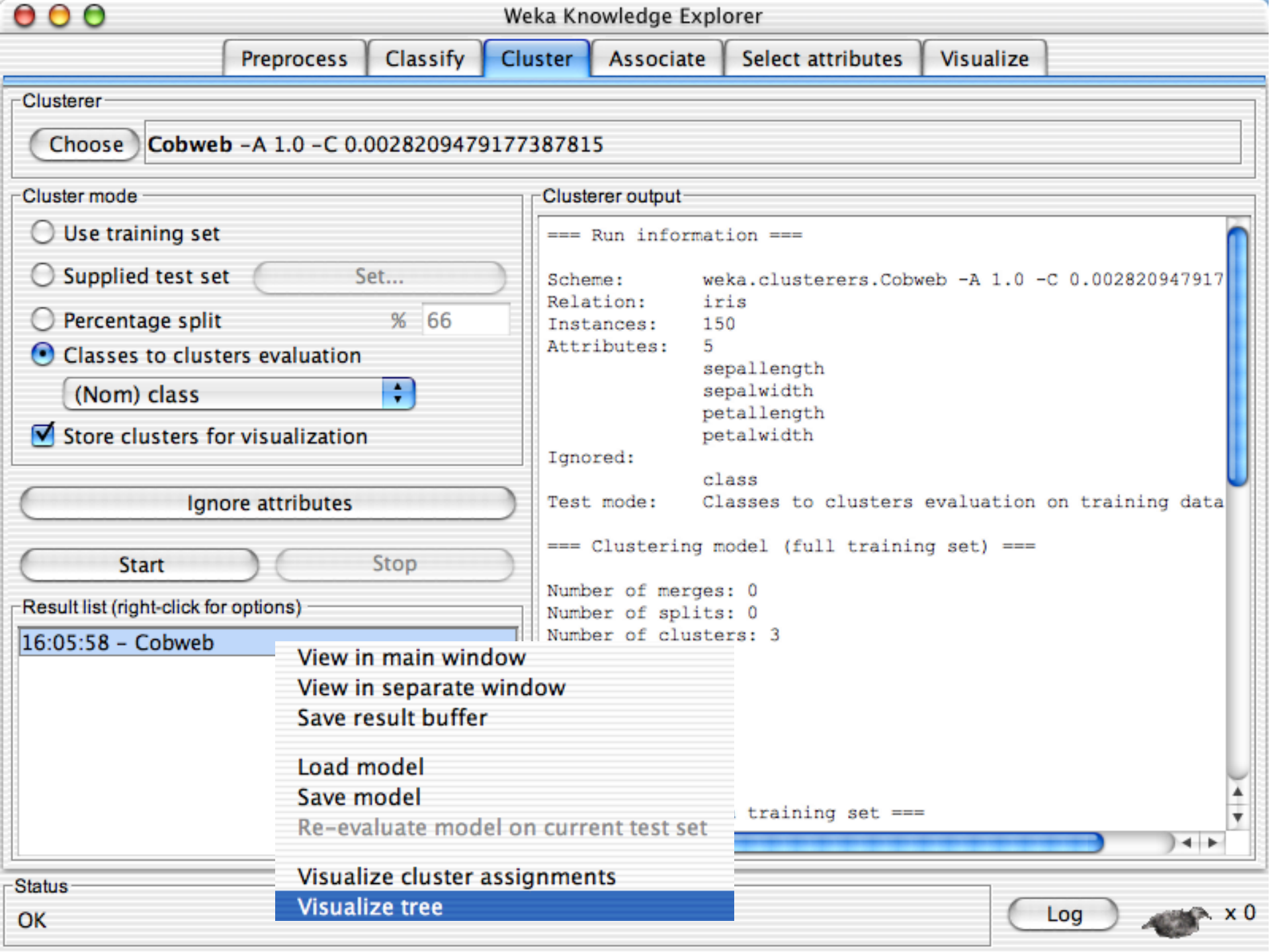
Classes to Clusters:

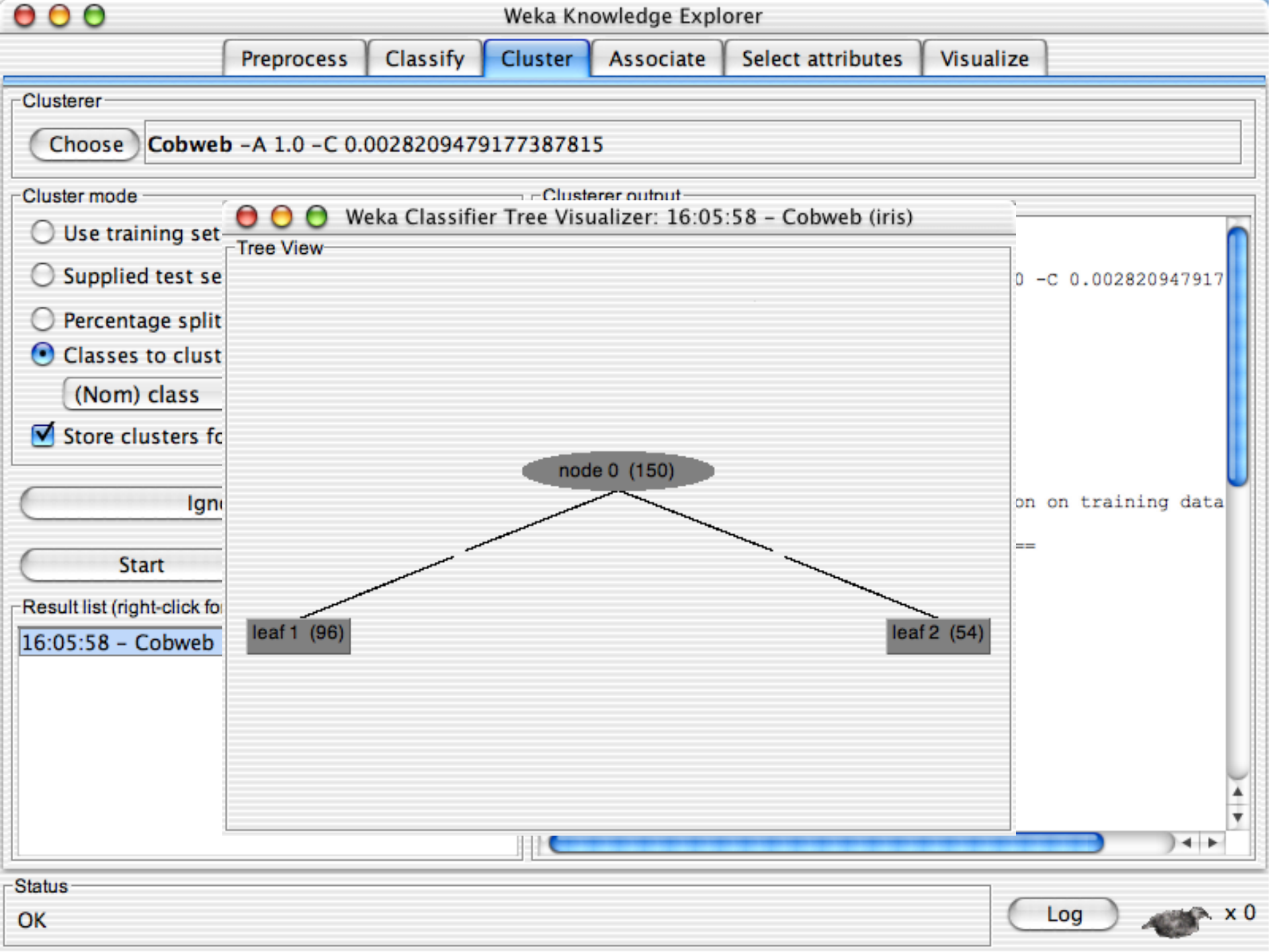
1	2	<-- assigned to cluster
0	50	Iris-setosa
50	0	Iris-versicolor
50	0	Iris-virginica

Cluster 1 <-- Iris-versicolor

Cluster 2 <-- Iris-setosa

Incorrectly clustered instances : 50.0 33.3333 %







Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Clusterer

Choose

Cobweb -A 1.0 -C 0.0028209479177387815

Cluster mode

☐ Use training set☐ Supplied test set

Set...

☐ Percentage split

% 66

☒ Classes to clusters evaluation

(Nom) class

☒ Store clusters for visualization

Ignore attributes

Start

Stop

Result list (right-click for options)

16:05:58 - Cobweb

View in main window

View in separate window

Save result buffer

Load model

Save model

Re-evaluate model on current test set

Visualize cluster assignments

Visualize tree

Clusterer output

=== Run information ===

Scheme: weka.clusterers.Cobweb -A 1.0 -C 0.002820947917
Relation: iris
Instances: 150
Attributes: 5
sepallength
sepalwidth
petallength
petalwidth

Ignored:

class

Test mode: Classes to clusters evaluation on training data

=== Clustering model (full training set) ===

Number of merges: 0
Number of splits: 0
Number of clusters: 3

on training set ===

Status

OK

Log

x 0

Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Clusterer

Choose

Cobweb -A 1.0 -C 0.0028209479177387815



Weka Clusterer Visualize: 16:05:58 - Cobweb (iris)

Cluster mode

☐ Use training se

X: petallength (Num)



Y: petalwidth (Num)

☐ Supplied test s

Colour: Cluster (Nom)



Select Instance

☐ Percentage spli☒ Classes to clus

Reset

Clear

Save

Jitter



(Nom) class

Plot: iris_clustered

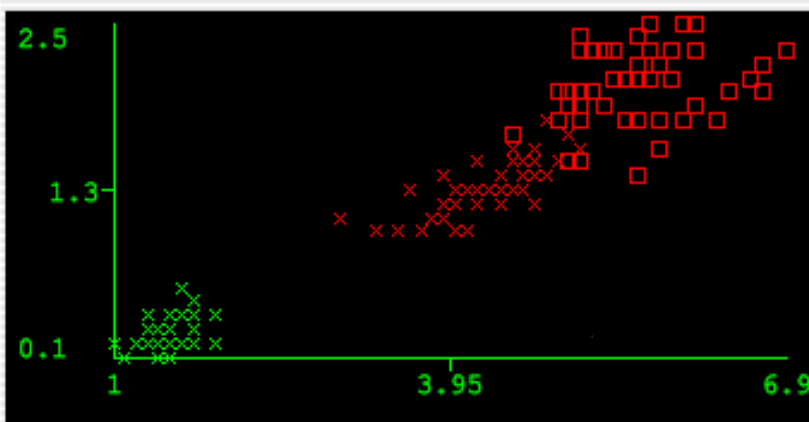
☒ Store clusters f

lgr

Start

Result list (right-click f

16:05:58 - Cobweb

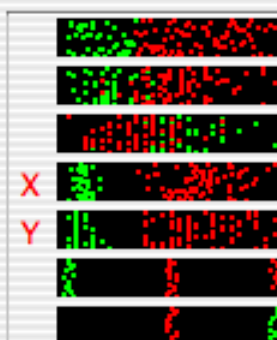


Class colour

cluster0

cluster1

cluster2



ion on training data

=== Evaluation on training set ===



Status

OK

Log



x 0

Explorar: asociaciones

- WEKA implementa el algoritmo Apriori para aprender reglas de asociación
 - Solo con datos discretos
- Puede identificar dependencias estadísticas entre grupos de atributos:
 - Leche, matequilla $\Rightarrow$ pan, huevos(con confianza de 0.9 y soporte de 2000)
- Apriori puede calcular reglas con mínimo soporte y que excedan una dada confianza



Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Associator

Choose

Apriori -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1 -S -1.0

Start

Stop

Result list (right-click for options)

Associator output

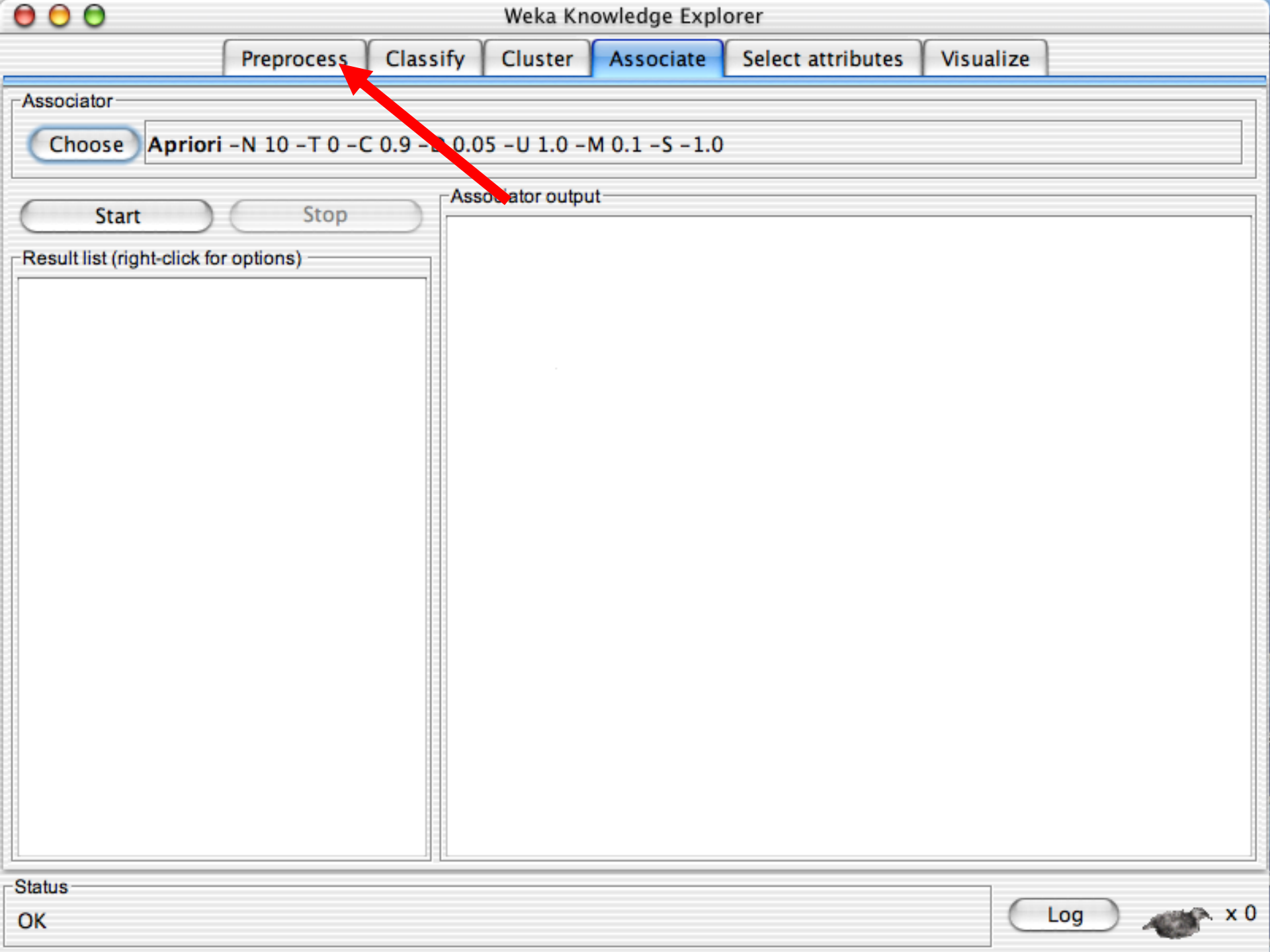
Status

OK

Log



x 0



Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Associator

Choose

Apriori -N 10 -T 0 -C 0.9 -E 0.05 -U 1.0 -M 0.1 -S -1.0

Start

Stop

Result list (right-click for options)

Associator output

Status

OK

Log

x 0



Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Open file...

Open URL...

Open DB...

Undo

Save...

Filter

Choose

None

Apply

Current relation

Relation: vote

Instances: 435

Attributes: 17

Attributes

No.	Name
1	handicapped-infants
2	water-project-cost-sharing
3	adoption-of-the-budget-resolution
4	physician-fee-freeze
5	el-salvador-aid
6	religious-groups-in-schools
7	anti-satellite-test-ban
8	aid-to-nicaraguan-contras
9	mx-missile
10	immigration
11	synfuels-corporation-cutback
12	education-spending
13	superfund-right-to-sue
14	crime
15	duty-free-exports
16	export-administration-act-south-africa
17	Class

Selected attribute

Name: handicapped-infants

Type: Nominal

Missing: 12 (3%)

Distinct: 2

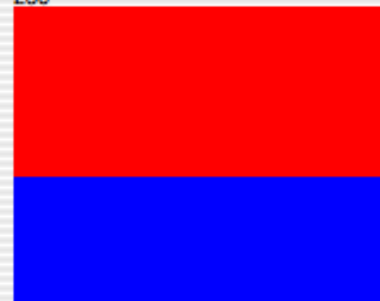
Unique: 0 (0%)

Label	Count
n	236
y	187

Colour: Class (Nom)

Visualize All

236



187



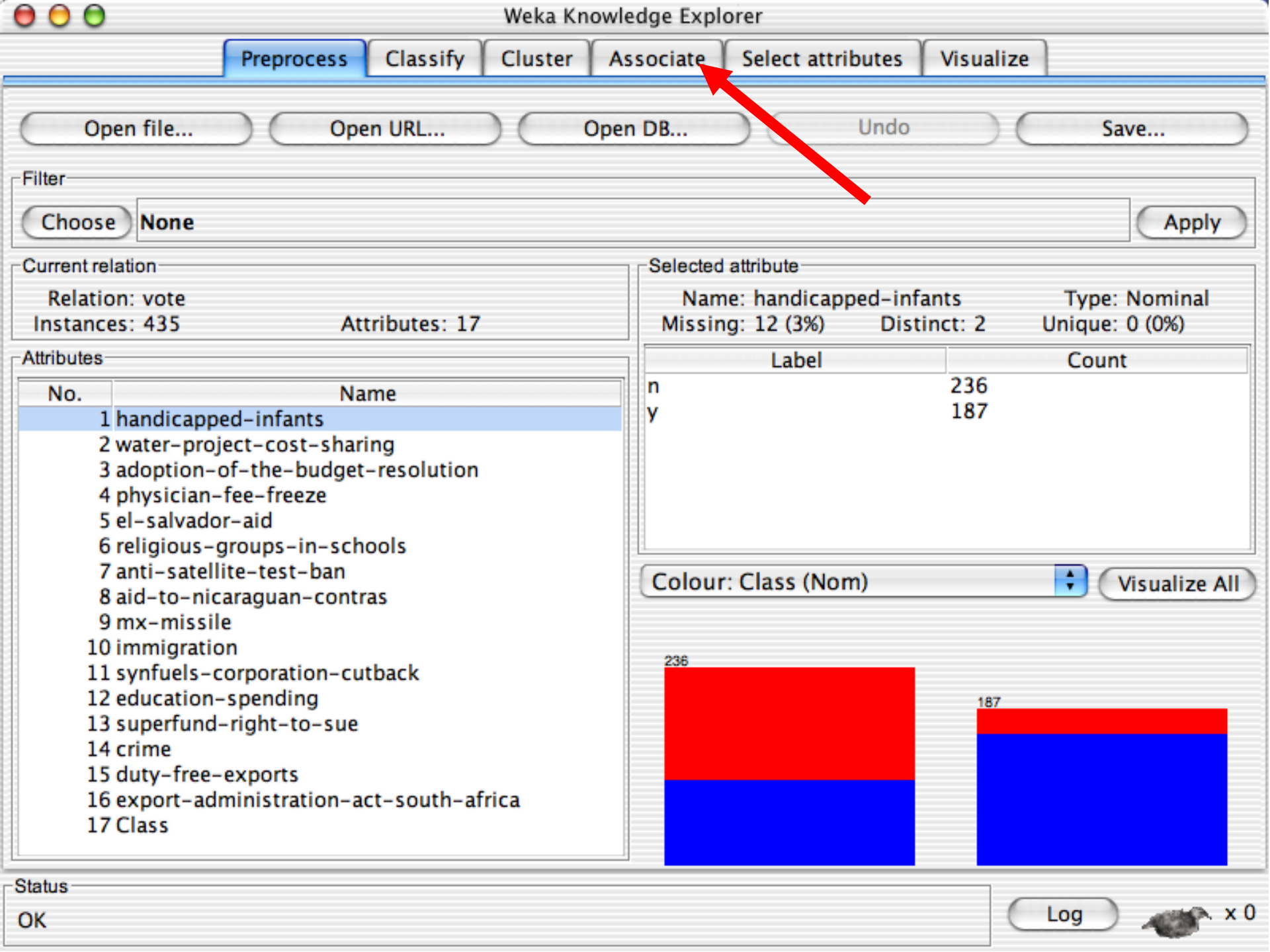
Status

OK

Log



x 0





Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Associator

Choose

Apriori -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1 -S -1.0

Start

Stop

Result list (right-click for options)

Associator output

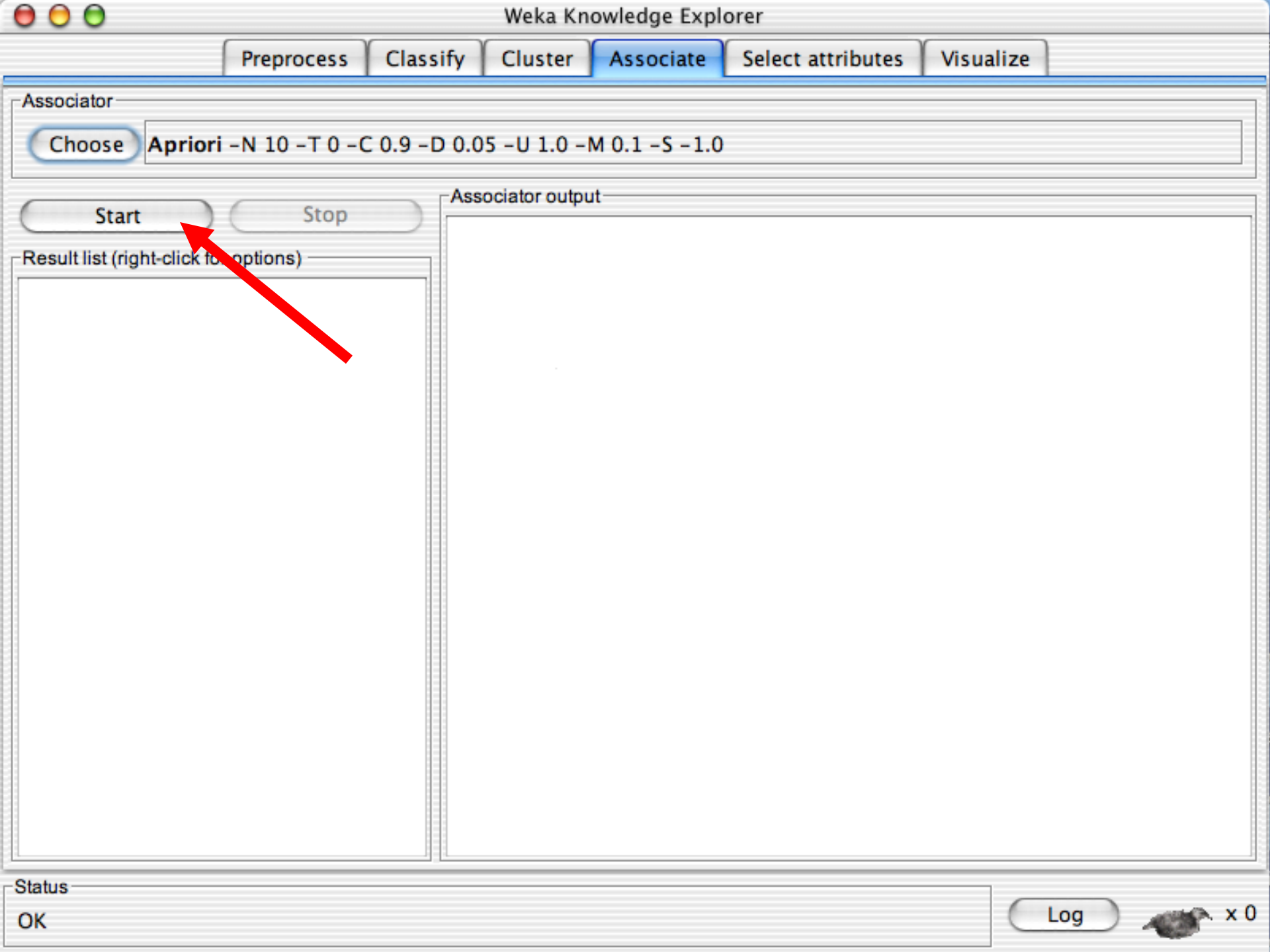
Status

OK

Log



x 0



Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Associator

Choose

Apriori -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1 -S -1.0

Start

Stop

Associator output

Result list (right-click for options)

Status

OK

Log

x 0

Preprocess

Classify

Cluster

Associate

Select attributes

Visualize

Associator

Choose Apriori -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1 -S -1.0**Start****Stop**

Result list (right-click for options)

16:29:37 - Apriori

Associator output

Minimum metric <confidence>: 0.9

Number of cycles performed: 11

Generated sets of large itemsets:

Size of set of large itemsets L(1): 20

Size of set of large itemsets L(2): 17

Size of set of large itemsets L(3): 6

Size of set of large itemsets L(4): 1

Best rules found:

1. adoption-of-the-budget-resolution=y physician-fee-freeze=n 219 ==> Class=democrat 219
2. adoption-of-the-budget-resolution=y physician-fee-freeze=n aid-to-nicaraguan-contras=y 210 ==> Class=democrat 210
3. physician-fee-freeze=n aid-to-nicaraguan-contras=y 211 ==> Class=democrat 210
4. physician-fee-freeze=n education-spending=n 202 ==> Class=democrat 201 conf: (0.99)
5. physician-fee-freeze=n 247 ==> Class=democrat 245 conf: (0.99)
6. el-salvador-aid=n Class=democrat 200 ==> aid-to-nicaraguan-contras=y 197 conf: (0.98)
7. el-salvador-aid=n 208 ==> aid-to-nicaraguan-contras=y 204 conf: (0.98)
8. adoption-of-the-budget-resolution=y aid-to-nicaraguan-contras=y Class=democrat 204
9. el-salvador-aid=n aid-to-nicaraguan-contras=y 204 ==> Class=democrat 197 conf: (0.98)
10. aid-to-nicaraguan-contras=y Class=democrat 218 ==> physician-fee-freeze=n 210

Status

OK

Log

x 0

Explorar: selección de atributos

- Panel para investigar los atributos más predictivos
- Métodos de selección:
 - Métodos de búsqueda: best-first, aleatorio, exhaustivo, algoritmos genéticos, ranqueo
 - Métodos de evaluación: basado en correlación, wrapper, ganancia de información, chi-squared, ...
- Se pueden combinar

Explorar: visualizar datos

- Visualizar es muy útil y práctico: p.e. ayuda a determinar dificultades en los procesos de aprendizaje
- WEKA puede visualizar un atributo simple (1-d) y pares de atributos (2-d)

Weka Java library

- Clases para carga de datos
- Clases para los clasificadores
- Clases para la evaluación

Clases para carga de datos

- `weka.core.Instances`
- `weka.core.Attribute`

Cada `DataRow` -> `Instance`,

```
# Load a file as Instances
FileReader reader;
reader = new FileReader(path);
Instances instances = new Instances(reader);
```

– ¿Cómo recuperar un valor de una instancia?

```
# Get Instance
Instance instance = instances.instance(index);
# Get Instance Count
int count = instances.numInstances();
```

– ¿Cómo recuperar un atributo?

```
# Get Attribute Name
Attribute attribute = instances.attribute(index);
# Get Attribute Count
int count = instances.numAttributes();
```

Clases para carga de datos

- ¿Cómo recuperar el valor del atributo para cada Instancia?

```
# Get value  
instance.value(index);    or  
instance.value(attrName);
```

- **Indice de Clase**

```
# Get Class Index  
instances.classIndex();           or  
instances.classAttribute().index();  
# Set Class Index  
instances.setClass(attribute);    or  
instances.setClassIndex(index);
```

Clases para los clasificadores

- Clases Weka para C4.5, Naïve Bayes, and SVM
 - Clasificadores:
 - C4.5: `weka.classifier.trees.J48`
 - NaiveBayes: `weka.classifiers.bayes.NaiveBayes`
 - SVM: `weka.classifiers.functions.SMO`
- Cómo construir un clasificador?

```
# Build a C4.5 Classifier
Classifier c = new weka.classifier.trees.J48();
c.buildClassifier(trainingInstances);
Build a SVM Classifier
Classifier e = weka.classifiers.functions.SMO();
e.buildClassifier(trainingInstances);
```

Clases para la evaluación

- weka.classifiers.CostMatrix
- weka.classifiers.Evaluation

- Como usarlas?

```
# Use Classifier To Do Classification
CostMatrix costMatrix = null;
Evaluation eval = new Evaluation(testingInstances, costMatrix);

for (int i = 0; i < testingInstances.numInstances(); i++){
    eval.evaluateModelOnceAndRecordPrediction(c,testingInstances.instance(i));
    System.out.println(eval.toSummaryString(false));
    System.out.println(eval.toClassDetailsString()) ;
    System.out.println(eval.toMatrixString());
}
```

Clases para la evaluación

Validación Cruzada

- Divide un único conjunto de datos en N partes iguales.
- Toma la $N-1$ como un conjunto de datos de entrenamiento, el resto se utilizará como prueba de conjunto de datos.

Clases para la evaluación

- Obtención conjunto de entrenamiento

```
Random random = new Random(seed);
instances.randomize(random);
instances.stratify(N);

for (int i = 0; i < N; i++)
{
    Instances train = instances.trainCV(N, i , random);
    Instances test = instances.testCV(N, i , random);
}
```



Tres pasos para el uso

- Asignar el archivo de datos primero
- Seleccionar funcionalidad
- Ejecutar Función utilizando rápido Minero

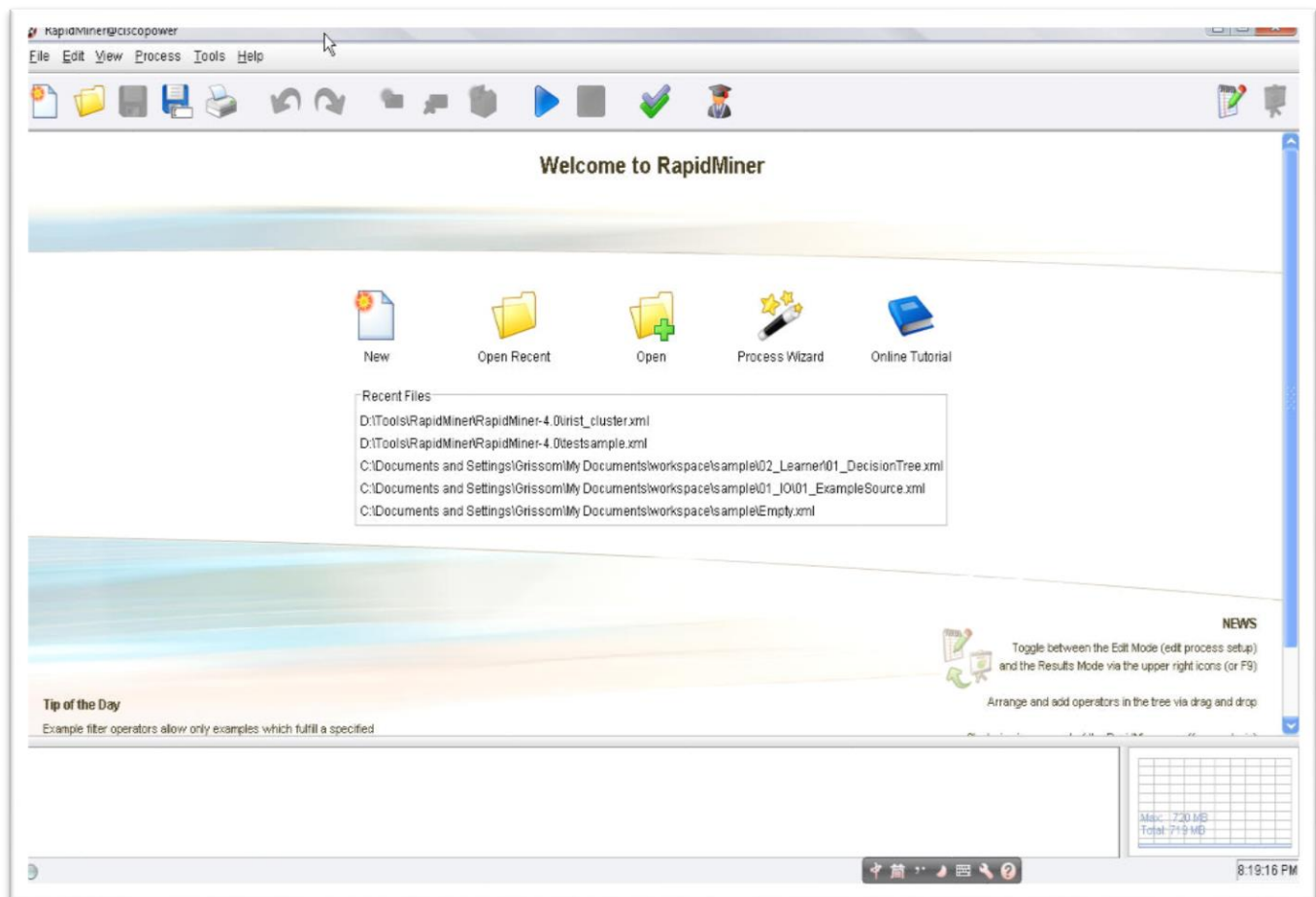
Los conjuntos de datos pueden utilizar formato ARFF, pero otros formatos también

RapidMiner

- Software open-source software que permite
 - Analisis inteligente de datos
 - Minería de datos
 - Descubrir Conocimiento
 - Aprendizaje automático
 - Analitica predictiva
 - Inteligencia de Negocio.
- Esta en Java y es GPL licencia
- <http://rapid-i.com>

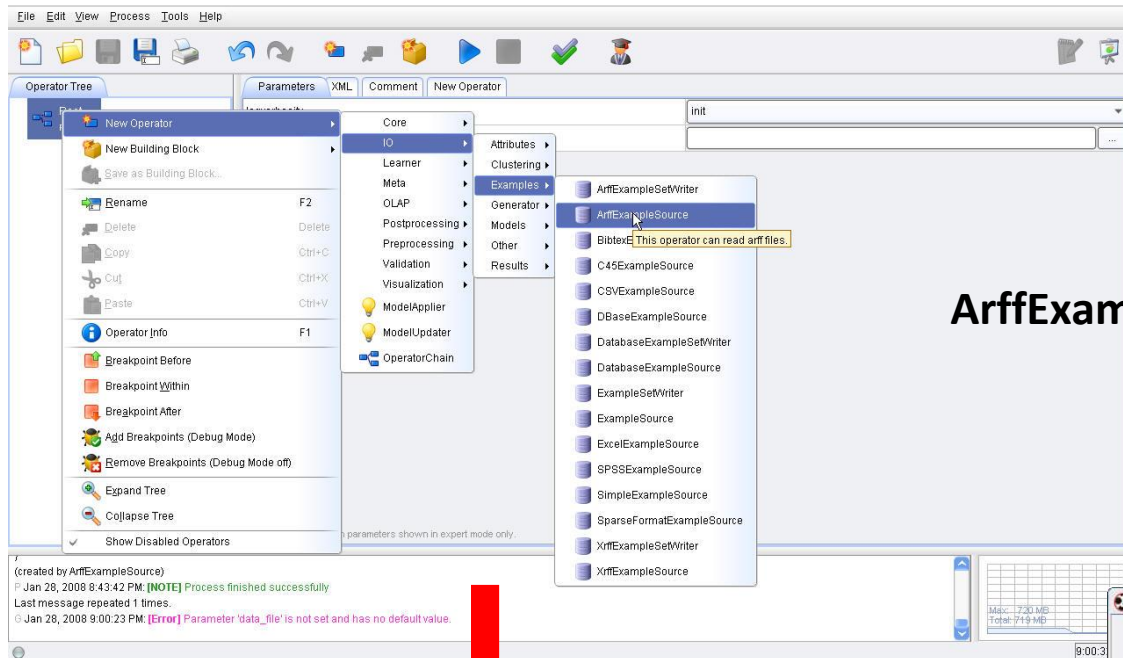
<http://rapid-i.com/content/view/130/82/>

➤ Crear un proyecto Rapid Miner (opción “new”)

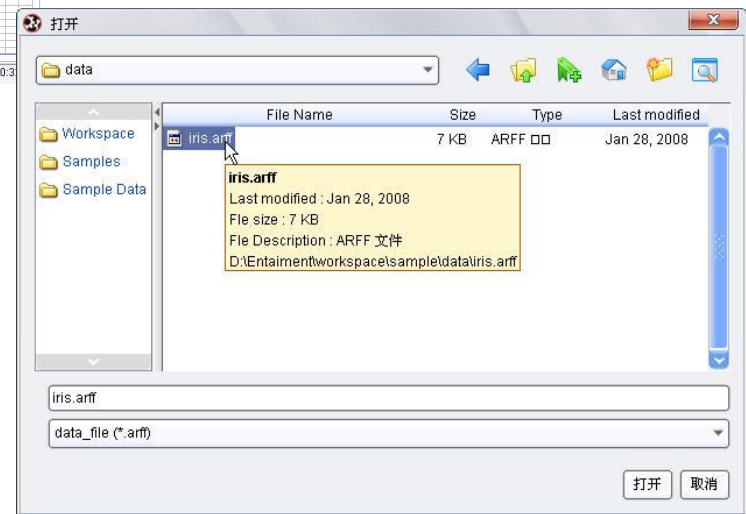
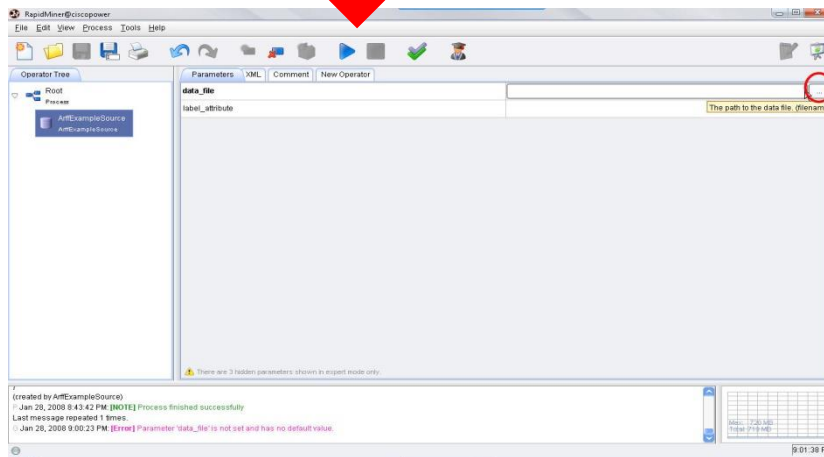


Establecer una fuente de datos para el proyecto

➤ Arff, excel , etc

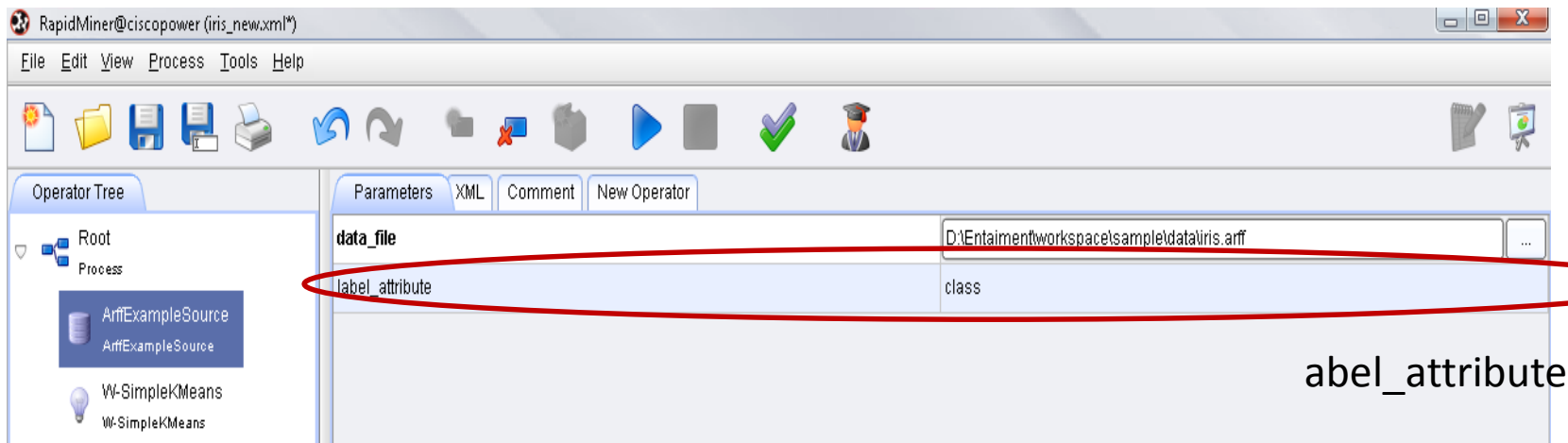


ArffExampleSource" menu



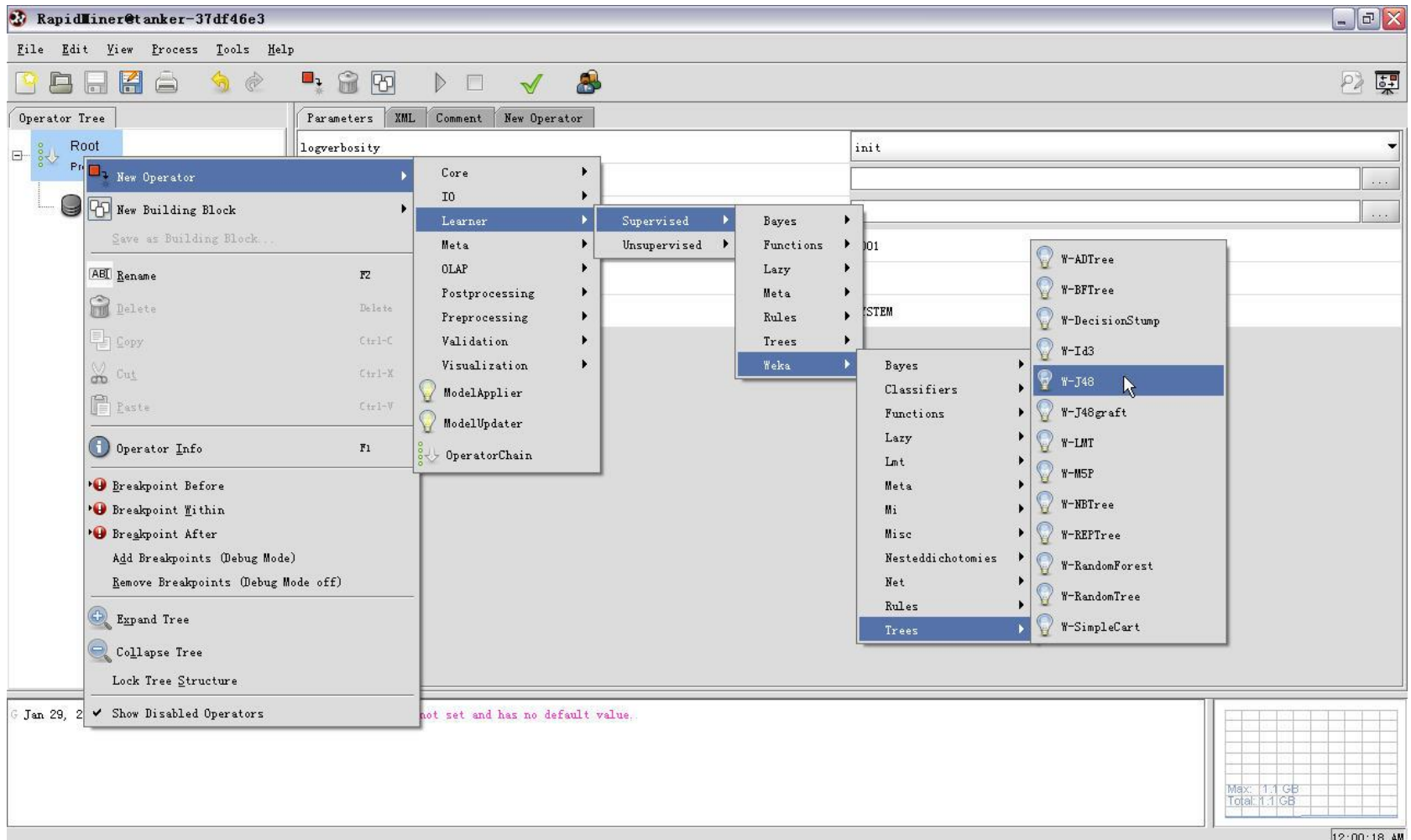
Escoger etiqueta atributo

➤ qué atributo en el origen de datos es el atributo de clase.



Selecccionar funcionalidad

clustering, classification, decision tree, etc.



Escoger parámetros

The screenshot displays the RapidMiner interface with the 'Parameters' tab selected. The 'Operator Tree' on the left shows a process starting with 'Root', followed by 'Process', 'AmExampleSource', and 'W-J48'. The 'Parameters' panel on the right lists various settings for the J48 operator, with several parameters highlighted by blue boxes:

- keep_example_set**: Indicates if this input object should also be returned as output. (boolean, default: false)
- U**: Use unpruned tree. (boolean, default: false)
- C**: Set confidence threshold for pruning. (default 0.25) (real, -∞+∞) 0.25
- M**: Set minimum number of instances per leaf. (default 2) (real, -∞+∞)
- R**: Use reduced error pruning. (boolean, default: false)
- N**: Set number of folds for reduced error pruning. One fold is used as pruning set. (default 3) (string)
- B**: Use binary splits only. (boolean, default: false)
- S**: Don't perform subtree raising. (boolean, default: false)
- L**: Do not clean up after the tree has been built. (boolean, default: false)
- A**: Laplace smoothing for predicted probabilities. (boolean, default: false)
- Q**: Seed for random data shuffling (default 1) (string)

At the bottom, a summary of the generated decision tree is shown:

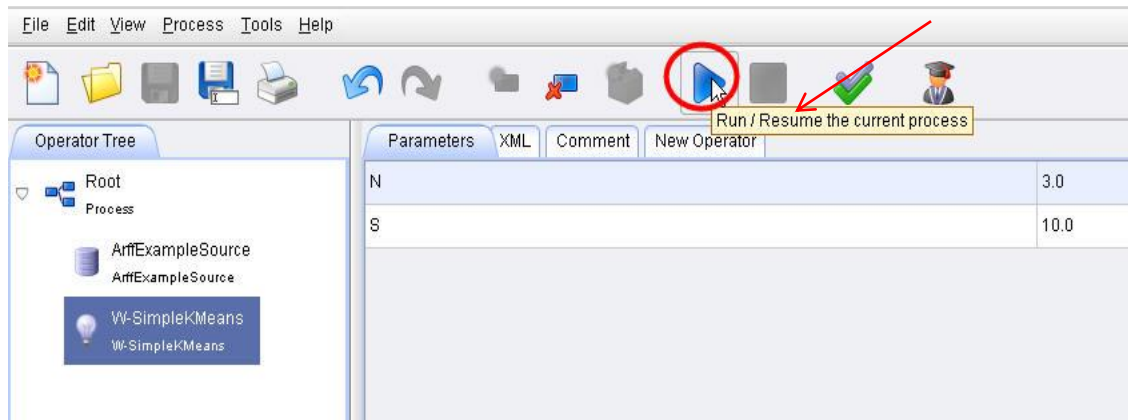
```
| | | petalwidth > 1.5: Iris-versicolor (3.0/1.0)
| | | petalwidth > 1.7: Iris-virginica (46.0/1.0)
Number of Leaves : 5
Size of the tree : 9
]
(created by Root)
```

The bottom right corner shows a small table with memory usage statistics:

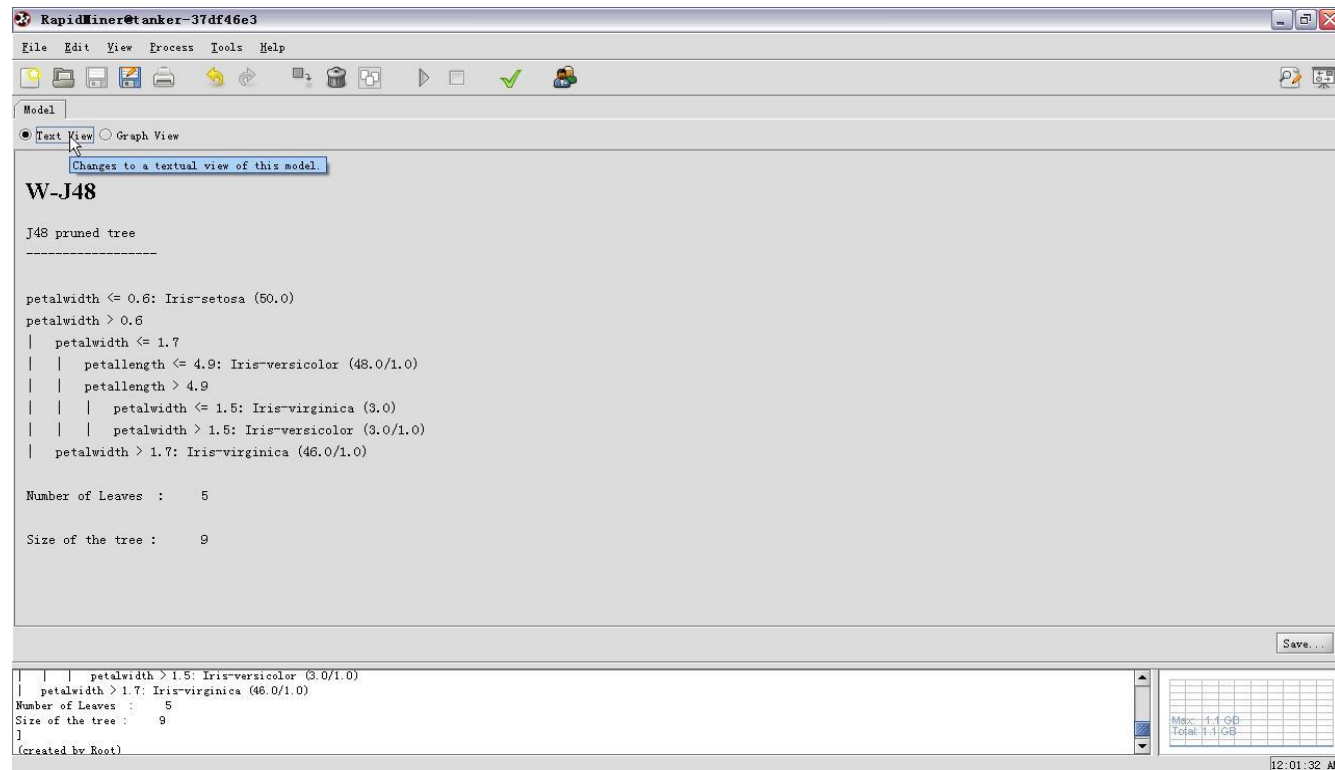
Mem:	1.1 GB
Total:	1.1 GB

The system clock in the bottom right corner indicates the time is 12:03:02 AM.

Ejecutar



resultados





Qué (y cómo) es R

- R es un entorno **libre** de **análisis estadístico** de datos y de **creación de gráficos estadísticos**
- Se basa en una interfaz de usuario de **líneas de comandos**
- Lenguaje Interpretado
- Guarda, analiza y gráfica datos
- Software Open Source

Razones para utilizar (y no) utilizar

- **Pros**

- Es libre
- Flexibilidad
- Procedimientos disponibles
- Se aprende estadística
- Estado del arte en estadística
- 2do detras de MATLAB para graficos.
- Comunidad de usuarios
- Lo obligue a pensar al usuarios sobre el analisis
- Interfaz con BD (SQL)

- **Contras**

- Al principio suele ser durp
- Cuesta un poco si se está habituado a trabajar con otro tipo de programas
- No amigable: curva de aprendizaje
- Fácil cometer errores

Definiciones Básicas

- Objetos
 - Funciones
 - *nombre.de.La.función(argumento/s=, opción/es=)*
- Espacio o área de trabajo
- Directorio de trabajo
- Paquetes
- Archivos de comandos o *scripts*

Como trabajar en R...

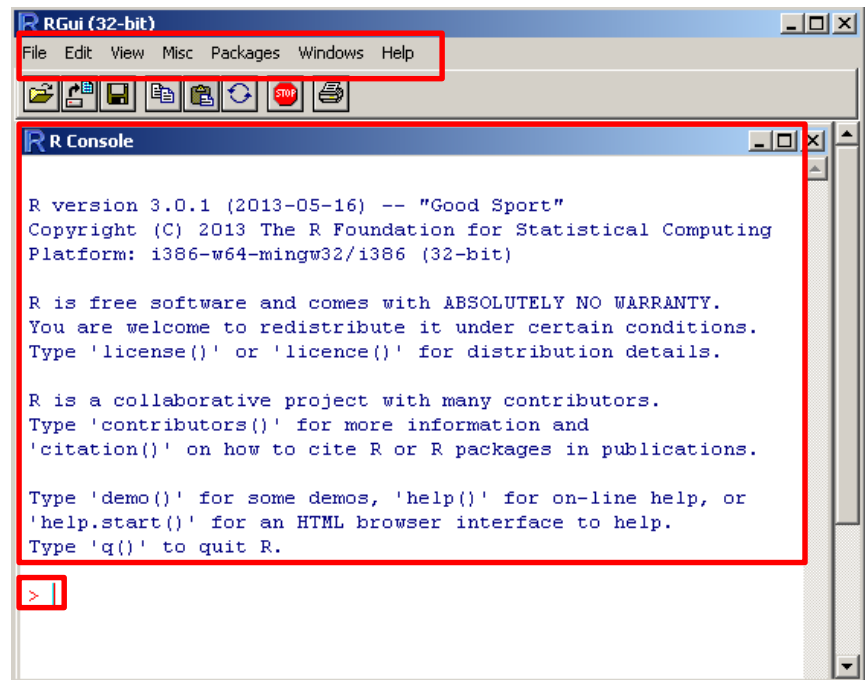
- Leer datos desde otras fuentes
- Usar paquetes, librerías, y funciones
- Escribir funciones si es necesario
- Conducir Análisis estadístico de Datos
Guardar salidas en archivos
- Guardar espacio de trabajo de R si es necesario

Corriendo R

- **Sistemas de ventanas(Console)**
- **Usando Editores**
 - Notepad, WinEdt, Tinn-R: Windows
 - Xemacs, ESS (Emacs speaks Statistics)
- **En el Editor:**
 - `source("filename.R")`
 - Salida puede ser direccionada con
 - `sink("filename.Rout")`

Descripción de la interfaz gráfica de usuario en R

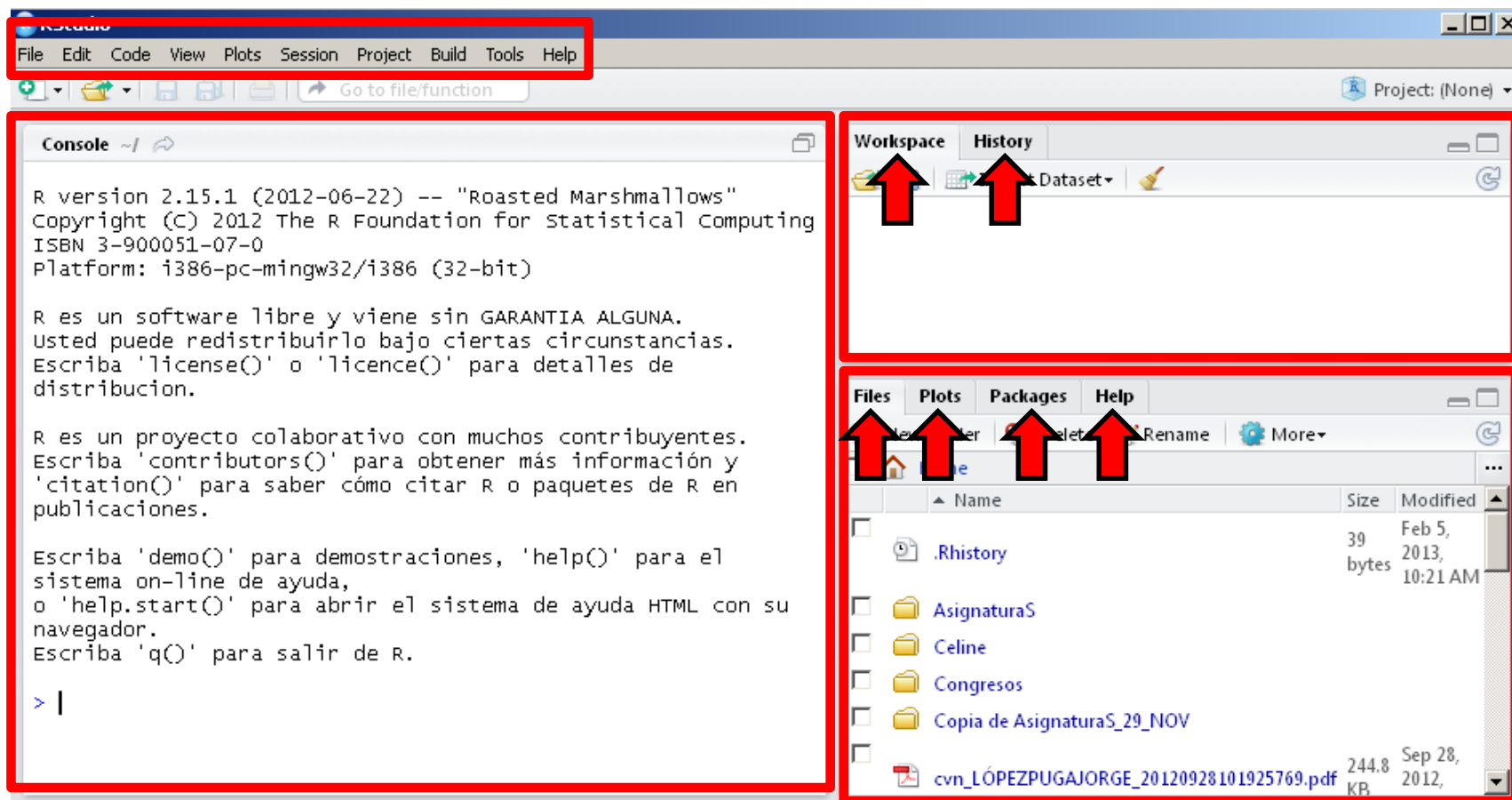
- Ejecutar el programa o el acceso directo del escritorio
- Apertura de la *consola de comandos*

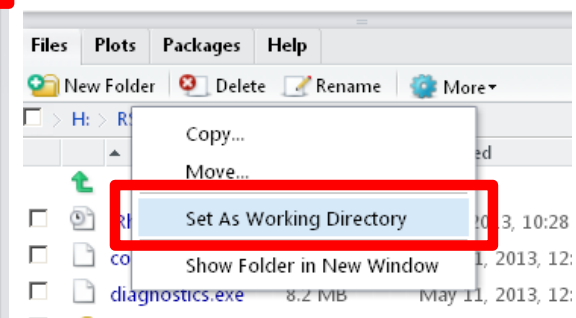
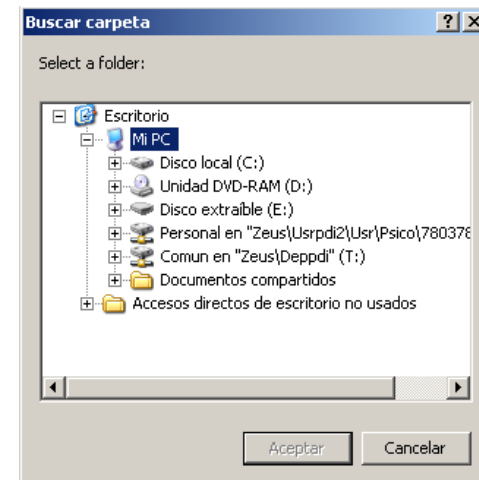
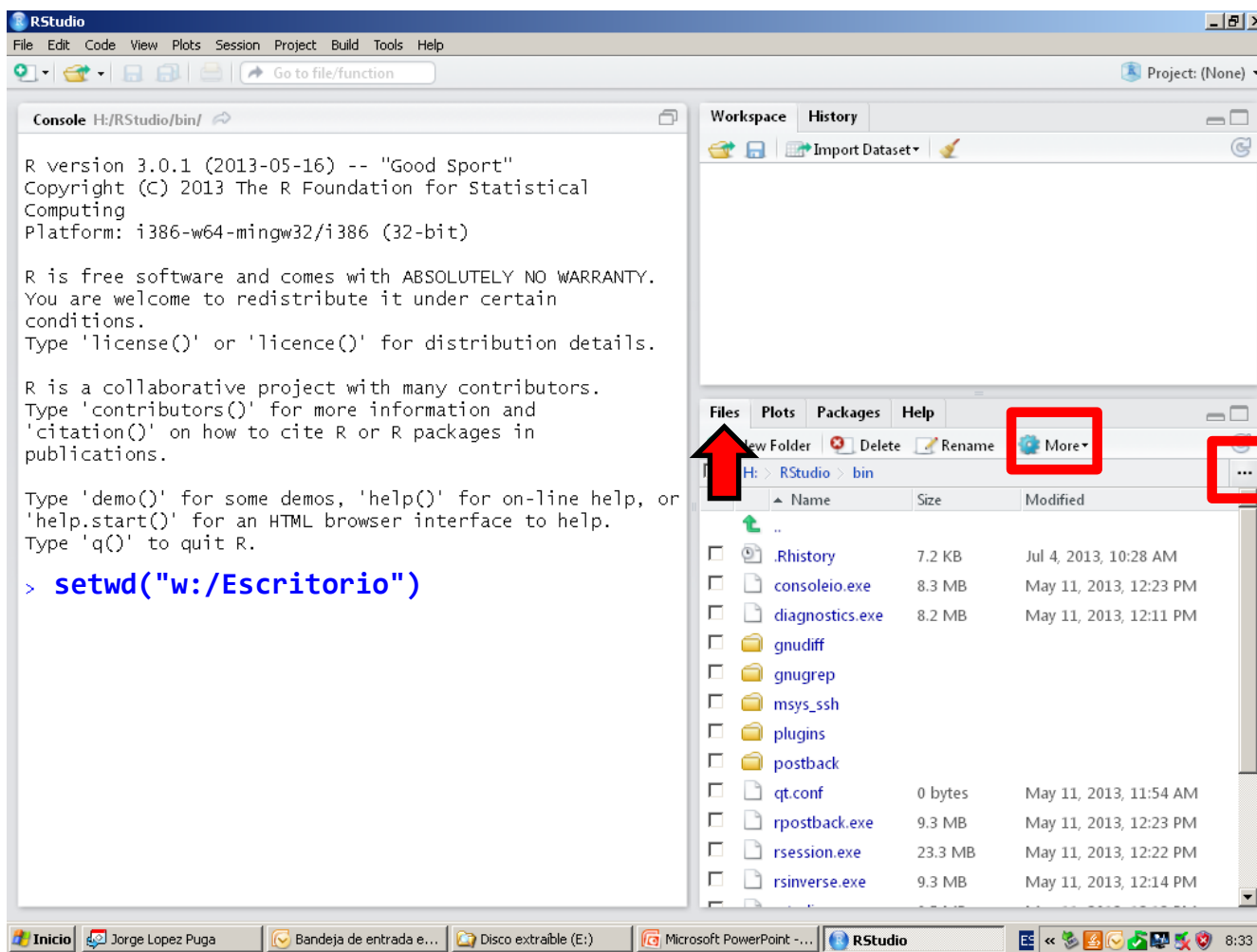


Otras Interfaces Interesantes

- Hay variedad de interfaces para
 - *R Commander* – es un paquete de R
 - *RKward* – <http://rkward.sourceforge.net>
 - *Tinn-R* – <http://www.sciviews.org/Tinn-R>
 - *Emacs* – <http://www.gnu.org/software/emacs>
 - *RStudio* – <http://www.rstudio.com>

Interfaz gráfica de *RStudio*





Comenzar con R

- Basico: asignaciones y operaciones.
- Operaciones Aritmeticas:
 - $+$, $-$, $*$, $/$, $^$.
- Aritmetica de Matrices.
 - $*$ is element wise multiplication
 - $\%*\%$ es una multiplicación de matrices
- Asignación
 - “ $<-$ ”

Descripción de paquetes

- Función `packageDescription()`
- Función `library(help="nombre.del.paquete")`

Cargar paquetes

- La función `library()`

Instalar paquetes

- La función `install.packages("nombre.del.paquete")`

Desactivar paquetes

- La función

`detach("package:nombre.del.paquete")`

Desinstalar paquetes

- La función

`remove.packages("nombre.del.paquete")`

Actualizar paquetes

- La función `update.packages()`

Los objetos de R

- Uno **objeto** es una estructura de datos con la que R puede trabajar
- Por ejemplo:
 - Un conjunto de datos – base de datos
 - El resultado de un análisis estadístico
 - Una tabla de datos
 - Una función

Tipos de objetos

Soporta cualquier tipo de datos

- Vector (numéricos, caracteres, lógicos)
- Matrices (*y arrays*)
- Listas (pueden contener información de diferente tipo)
- Tablas (de frecuencia o de contingencia)
- *Data frame* o base de datos

variables

```
> a = 49  
> sqrt(a)  
[1] 7
```

numeric

```
> a = "The dog ate my homework"  
> sub("dog","cat",a)  
[1] "The cat ate my homework"
```

character
string

```
> a = (1+1==3)  
> a  
[1] FALSE
```

logical

Datos

	Lineal	Rectangular
Del mismo Tipo	VECTORS	MATRIX*
Mezcla	LIST	DATA FRAME

Tareas específicas

- **search():** ver cuales directories y datos existen
- **ls():** ver objetos guardados
- **attach(NameOfTheDataset, expression):** incluir un conjunto de datos en el camino para proceso de análisis
- **detach(NameOfTheDataset):** quitar un conjunto de datos del camino del proceso de análisis despues de culminarlo

Creación de objetos

- La creación de objetos se lleva a cabo realizando una **asignación**
- Para ello utilizamos los símbolos $\leftarrow$, $\rightarrow$ o $=$
 - Los códigos parecidos a flecha funcionan en ambos sentidos
 - El símbolo igual sólo en un sentido

R como calculador

```
> log2(32)
```

```
[1] 5
```

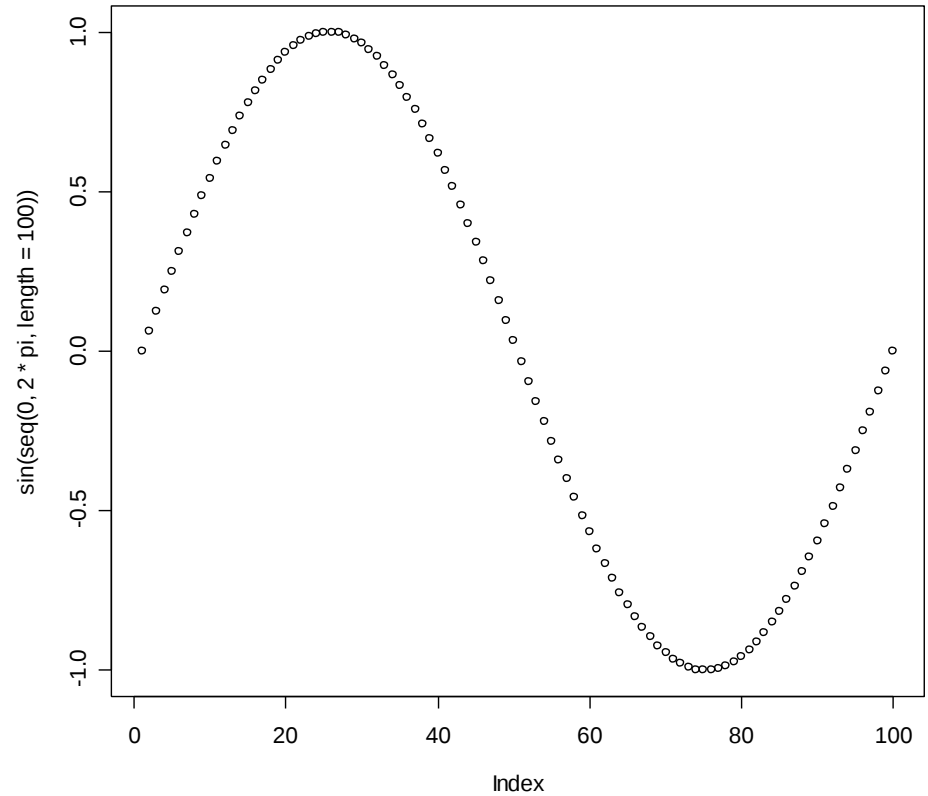
```
> sqrt(2)
```

```
[1] 1.414214
```

```
> seq(0, 5, length=6)
```

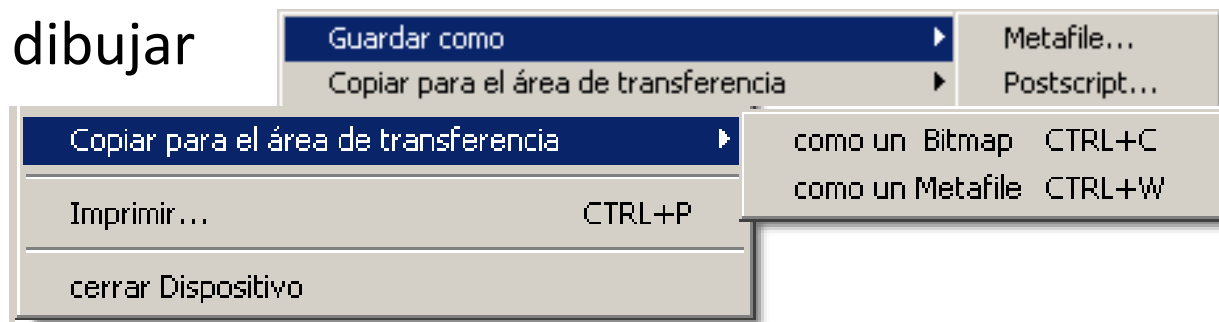
```
[1] 0 1 2 3 4 5
```

```
> plot(sin(seq(0, 2*pi, length=100)))
```



Archivos de código

- R no genera archivos de gráficos propios
- Por tanto, es conveniente guardarlos como *scripts*
- Recordar la extensión .R
- Comentar líneas para aclaraciones
- El dispositivo gráfico
- Abrir un dispositivo gráfico
 - Al ejecutar una función de alto nivel – consola de R
 - Con la función `windows()` → dibujar gráfico



Utilizando código

- Habría que dar tres pasos
 1. Inicializar – generar el archivo
 2. Generar el gráfico
 3. Cerrar el dispositivo gráfico

- Ejemplo:

```
jpeg("imagen.jpeg")  
hist(rivers, col="blue")  
dev.off()
```

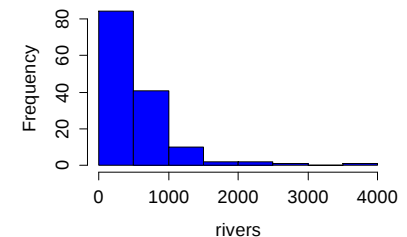


imagen.jpeg

0 bytes



Histogram of rivers



Creación de vectores

- La función `c()`
- La función `assignn()`
- Los dos puntos `:`
- La función `seq()`
- Distribuciones estadísticas
- Ejemplos

Modos de crear matrices y arrays

- A partir de un vector con la función `dim()`
- Con la función `matrix()`
- Con la función `array()`

vectores, matrices

vector: mismo tipo

```
> a = c(1,2,3)
```

```
> a*2
```

```
[1] 2 4 6
```

- En R, un numero es un especial caso de un vector con 1 elemento
- Other vector types: character strings, logical

matriz: tabla rectangular de datos del mismo tipo

- **ejemplo:** biopsias descritas por 10000 genes y 30 tejidos: una matriz de 10000 filas y 30 columnas.

Listas

- **vector:** acceso.

```
> a = c(7,5,1)
```

```
> a[2]
```

```
[1] 5
```

lista: colección de datos de tipo arbitrario.

```
> doe = list(nombre="jose",edad=28,casado=F)
```

```
> doe$nombre
```

```
[1] "jose"
```

```
> doe$edad
```

```
[1] 28
```

- Elementos vectores/matrices se accesan por indices (un entero):
x[1], y[1,] y[,1] ,
- Elementos listas/frames por sus nombres (cadena caracter):
myframe\$age.
- Ahora, ambos tipos soportam ambos métodos.

Creación de tablas y su utilidad

- Creación (entre otras)
 - Con la función `table()`
 - Con la función `as.table()`
- Utilidad:
 - Para realizar análisis de frecuencias
 - Para crear gráficos (por ejemplo, de barras o de sectores)

Creación de tablas y su utilidad

- Función `read.table()`
 - Lee datos directamente
 - La primera línea del archivo tiene nombre de cada variable en la tabla
 - Cada línea adicional comienza con una etiqueta de fila y despues los valores de cada variable.

	Precio	Piso	Area	Hab	Edad	Calent
01	52.00	11	830	5	6.2	no
02	54.75	8	710	5	7.5	no
03	57.50	1	1000	5	4.2	no
04	57.50	13	690	6	8.8	no
05	59.75	9	900	5	1.9	yes

...

```
HousePrice <- read.table("houses.data", Calent=yes)
```

¿Qué es una base de datos?

- En R se le llama *data frame*
- Una especie de matriz bi-dimensional
 - Columnas representan variables
 - Filas representan personas, registros o casos
- Contiene diferentes tipos de datos (es un tipo de *lista*)
- Ejemplo: *iris*

Creación de bases de datos

- Existe la función `data.frame()`
 - Hay que crear previamente un conjunto de vectores
- A partir de otros objetos `as.data.frame()`
- También podemos importar datos desde archivos externos (Excel, SPSS, .csv, .txt)

Data frames

Tabla **rectangular** de filas y columnas; y datos en cada columna son del mismo tipo (p.e. numero, texto, lógico), pero diferentes columnas pueden tener diferentes tipos.

> a

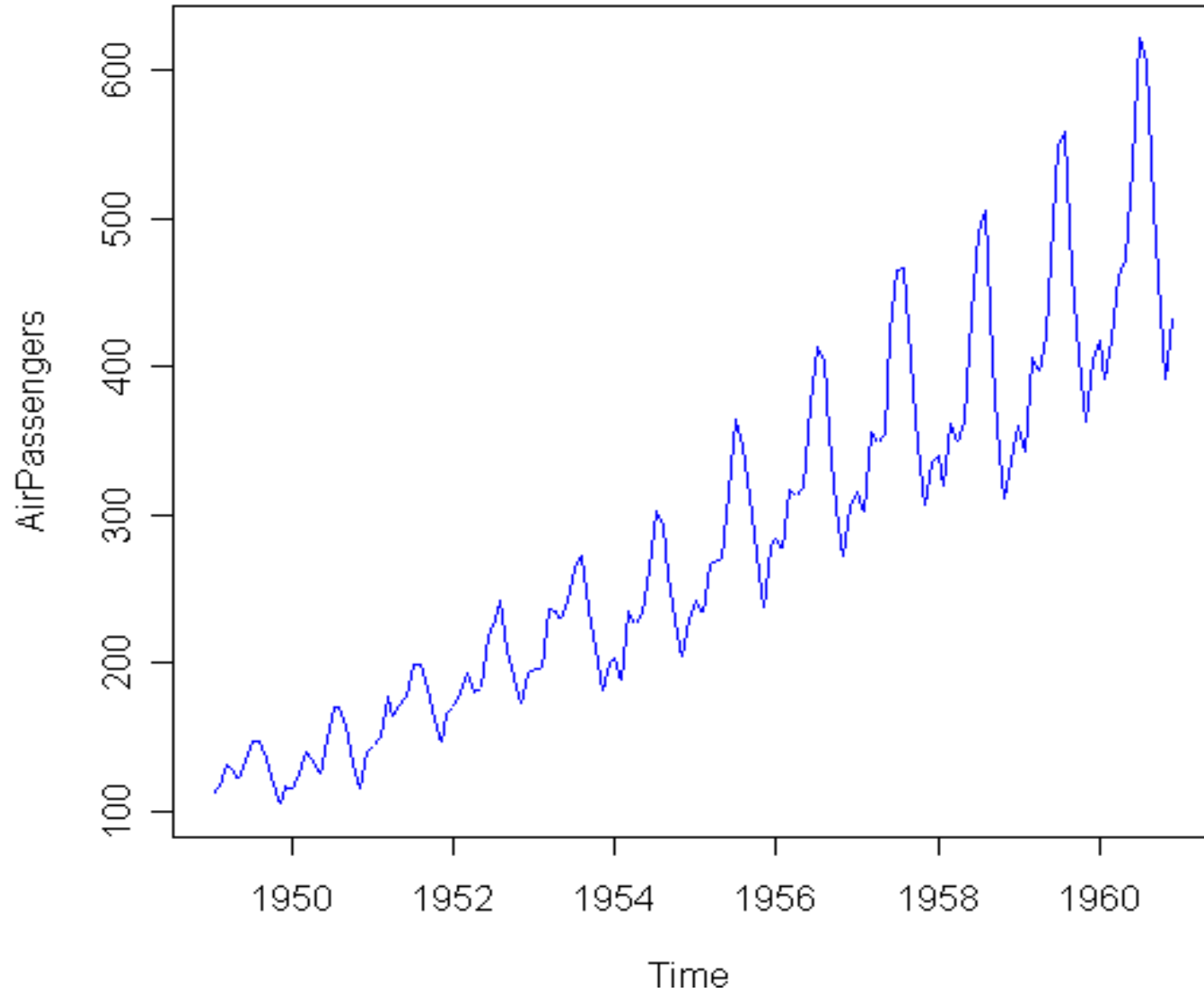
	localización	tamatumor	progreso
XX348	proximo	6.3	FALSE
XX234	distan	8.0	TRUE
XX987	proximo	10.0	FALSE

Gráficos

- Dibujar un objeto: `plot(num.vec)`
- Dibujo enviar a un dispositivo gráfico
 - Debe decirse dispositivo
 - postscript, gif, jpeg, etc...
 - Se pueden apagar o encender: `dev.off()`
- 2 tipos
 - Alto nivel: dibujo gráfico
 - Bajo nivel: agregar información a uno existente

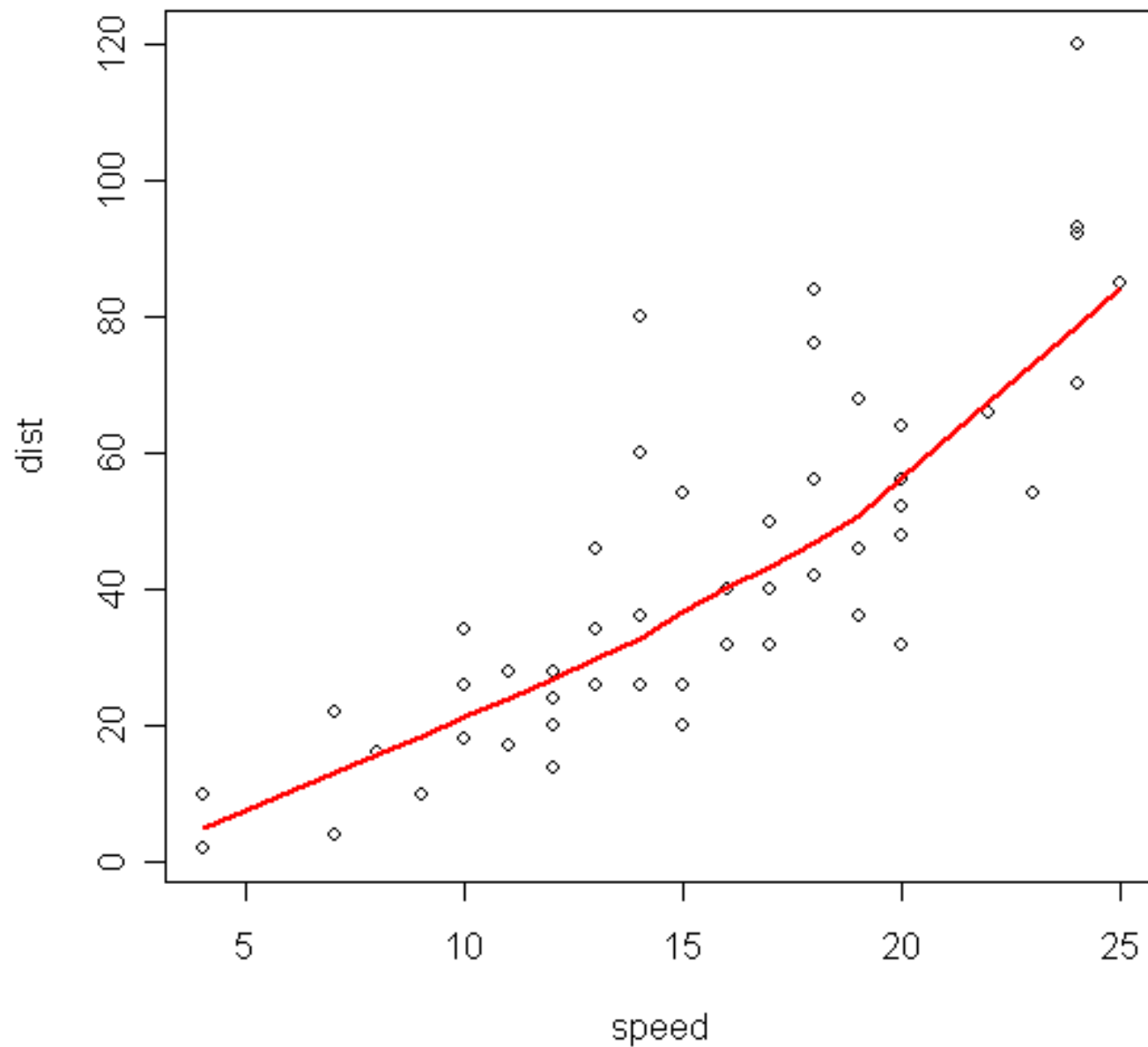
Alto nivel: generado con plot()

Number of Airline Passengers over time



Bajo nivel

distance vs speed



Programando con R

- **Funciones & Operadores**
- **Expresiones definidas en {}**
- **Código separado por nueva linea, “;” no es necesario**
- **Repetición**
- **Condicional**

Condicional

```
if (logical expression) {  
    statements  
} else {  
    alternative statements  
}
```

else branch is optional

Lazos

```
for(i in 1:10) {  
  print(i*i)  
}
```

```
i=1  
while(i<=10) {  
  print(i*i)  
  i=i+sqrt(i)  
}
```

- repeat expr statement.
- instrucción break usado para terminar un lazo.
- instrucción next usado para saltar un ciclo y pasar al siguiente.

lapply, apply

- Si una tarea debe ejecutarse sobre los elementos de una lista o matriz

`lapply(li, function )`

```
> li = list("klaus","martin","georg")
```

```
lapply(li, toupper)
```

- `[[1]]`
- `[1] "KLAUS"`
- `[[2]]`
- `[1] "MARTIN"`
- `[[3]]`
- `[1] "GEORG"`

- Aplicar la función fct en una dimension de una matriz arr, y regresarlo en un vector o matriz de apropiada tamaño.

`apply( arr, margin, fct )`

```
> x
```

```
  [,1] [,2] [,3]  
[1,]  5  7  0  
[2,]  7  9  8  
[3,]  4  6  7  
[4,]  6  3  5
```

```
> apply(x, 1, sum)
```

```
[1] 12 24 17 14
```

```
> apply(x, 2, sum)
```

```
[1] 22 25 20
```

Guardar datos

```
> save(x, file="x.Rdata")  
> load("x.Rdata")
```

Importar y exportar datos

```
> x = read.delim("filename.txt")
```

also: read.table, read.csv

```
> write.table(x, file="x.txt", sep="\t")
```

Funciones estadísticas en R

- **Descriptivas**

- mean, median, first, third quartiles,
- stem() gives stem-leaf plots
- table() gives tabulation of categorical variables

- **Modelado**

- Regresión: Lineal y Lógica
- Series de tiempo
- Más de 400 funciones
 - lm, glm, aov, ts
- Muchas librerías & paquetes
 - survival, coxph, tree (recursive trees), nls, ...

- **Multivariables**

Descriptivas

- Tendencia central
 - Media
 - Mediana
 - Segmentación por factores
- Opciones interesantes

Tendencia central

- Para describir variables de manera general
- La función `median()`
- Por ejemplo:

```
median(rivers)      [1] 425
```

- La media aritmética
- Función `mean()`
- Por ejemplo:

```
mean(austres)      [1] 15273.45
```

Opciones interesantes

- Aprovechando que acabamos de hablar de la media...
- El argumento `trim`
 - Sirve para estimar medias recortadas – valor de 0 a 0.5
- Por ejemplo:

```
mean(austres, trim = 0.025)
```

```
mean(austres, trim = 0.05)
```

```
[1] 15268.83 95%
```

```
[1] 15263 90%
```

Descriptivas

- Dispersión:
 - La función para la varianza: `var()`
 - La función para la desviación típica: `sd()`
- Forma
 - La función para el sesgo o asimetría: `skewness()`
 - Función para el apuntamiento o curtosis:
`kurtosis()`

Dispersión

- Está referida a la heterogeneidad-homogeneidad
- La función `var()` para estimar la varianza
- Puede funcionar
 - Con vectores o variables
 - Con matrices
- Por ejemplo:

```
var(rivers)
```

```
[1] 243908.4
```

Dispersión

- Cuando el argumento es una matriz
 - Matriz de varianzas-covarianzas

```
data(iris)
```

```
iris$Species <- NULL
```

```
var(iris)
```

	Sepal.Length	Sepal.width	Petal.Length	Petal.width
Sepal.Length	0.6856935	-0.0424340	1.2743154	0.5162707
Sepal.width	-0.0424340	0.1899794	-0.3296564	-0.1216394
Petal.Length	1.2743154	-0.3296564	3.1162779	1.2956094
Petal.width	0.5162707	-0.1216394	1.2956094	0.5810063

Dispersión

- La función `sd()` para estimar la desviación típica
- El argumento (`x`) ha de ser un vector
- Acepta el argumento `na.rm` para casos perdidos
- Por ejemplo:

```
sd(rivers) [1] 493.8708
```

```
vp <- c(45, 75, 32, NA, 22, 14, 85)  
sd(vp, na.rm = TRUE) [1] 28.83574
```

Como modelar

- **Especificar modelo:**

$$y \sim x_i + c_i,$$

- **y = variable resultado, x_i = variables de entrada, c_i = constantes, + = operador**

- **Operadores**

- **+ = agrega, : = interactua, / = anida, etc...**

```
carReg <- lm(speed~dist, data=cars)  
carReg = es un objeto
```

Ejemplo de Modelo de Regresión

plot

- La función `plot()`
- Gráficos de dispersión
- Gráficos de caja
- Gráficos de series temporales
- Gráficos de barras
- Gráficos de Mosaico
- Gráficos de densidad

La función `plot()`

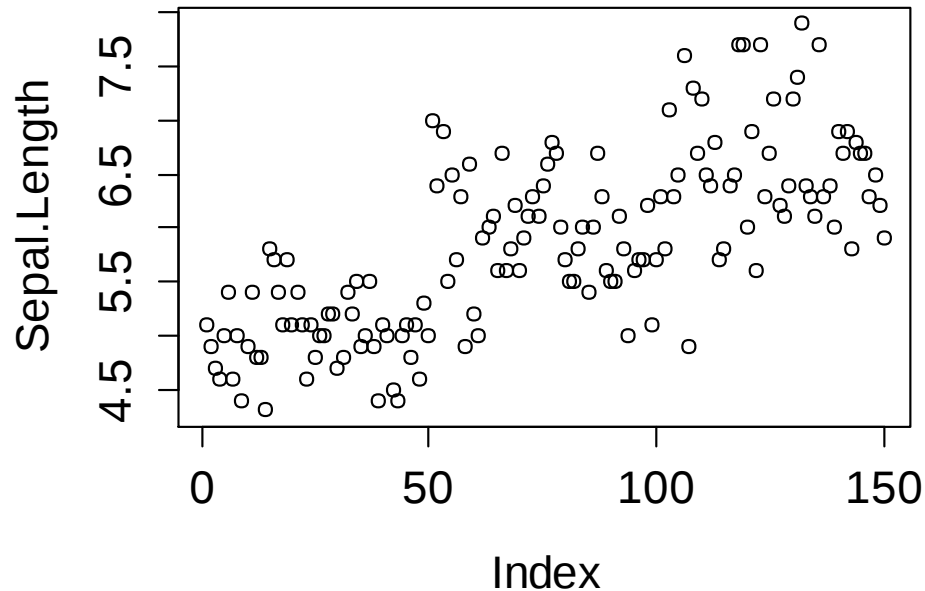
- Es una función genérica para dibujar gráficos *x-y*
- Sus dos argumentos básicos son `x` e `y`
- Otro argumento interesante que puede tomar es `type`
 - `p` – puntos (por defecto)
 - `l` – línea
 - `b` – ambos
 - `o` – superpuestos
 - `h` – histograma
 - `s` – salto de “escalera”

La función `plot()`

- Ejemplos
- Usemos la variable `Sepal.Length` contenida en la base de datos `iris`
- Recuerda utilizar `attach()`
- Es decir:
 - `attach(iris)`

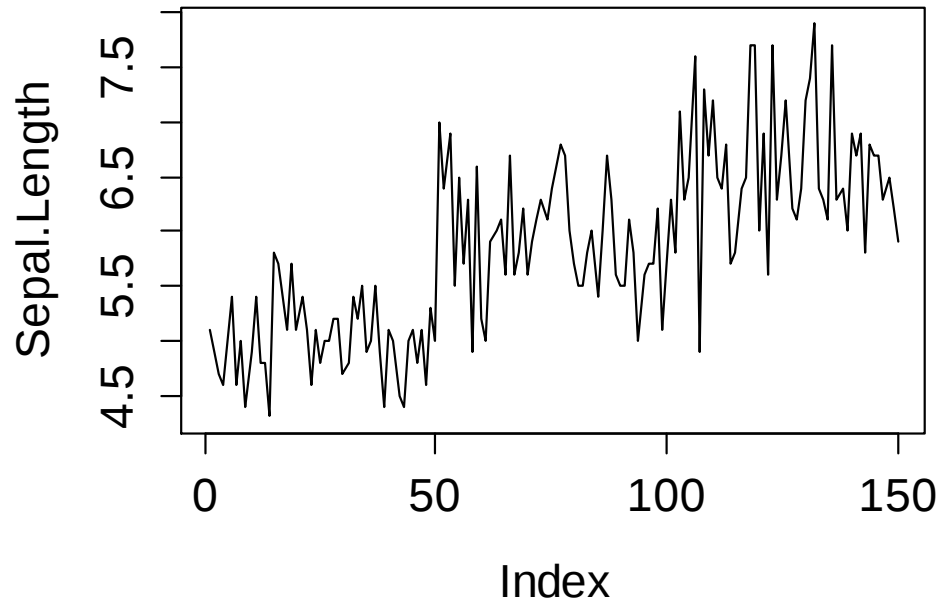
La función `plot()`

- `plot(Sepal.Length, type="p") = plot(Sepal.Length)`



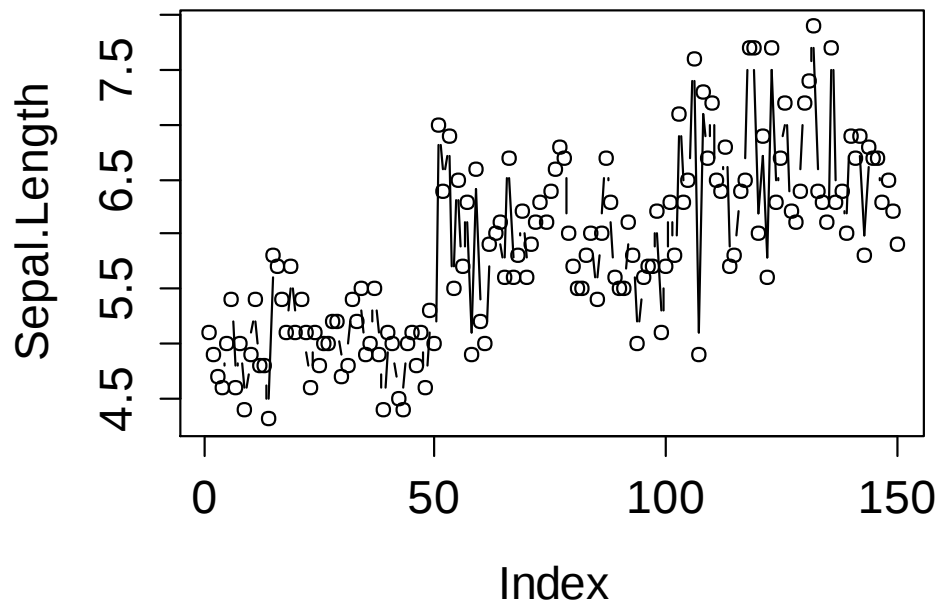
La función `plot()`

- `plot(Sepal.Length, type="l")`



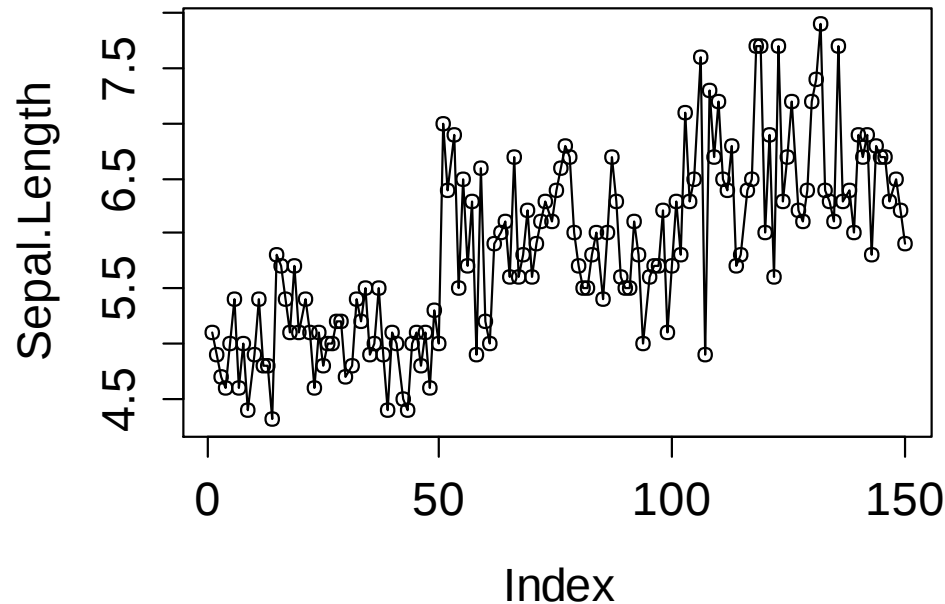
La función `plot()`

- `plot(Sepal.Length, type="b")`



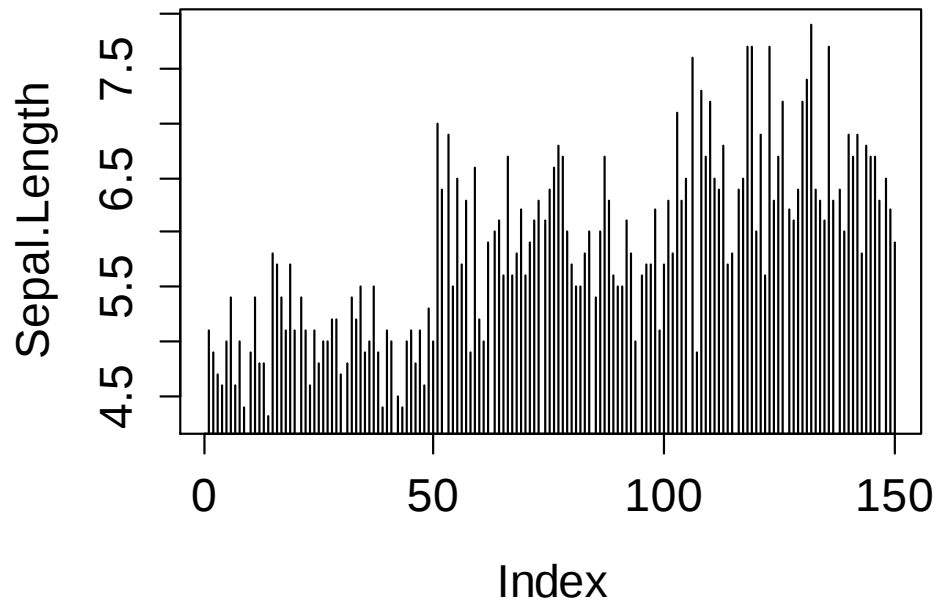
La función `plot()`

- `plot(Sepal.Length, type="o")`



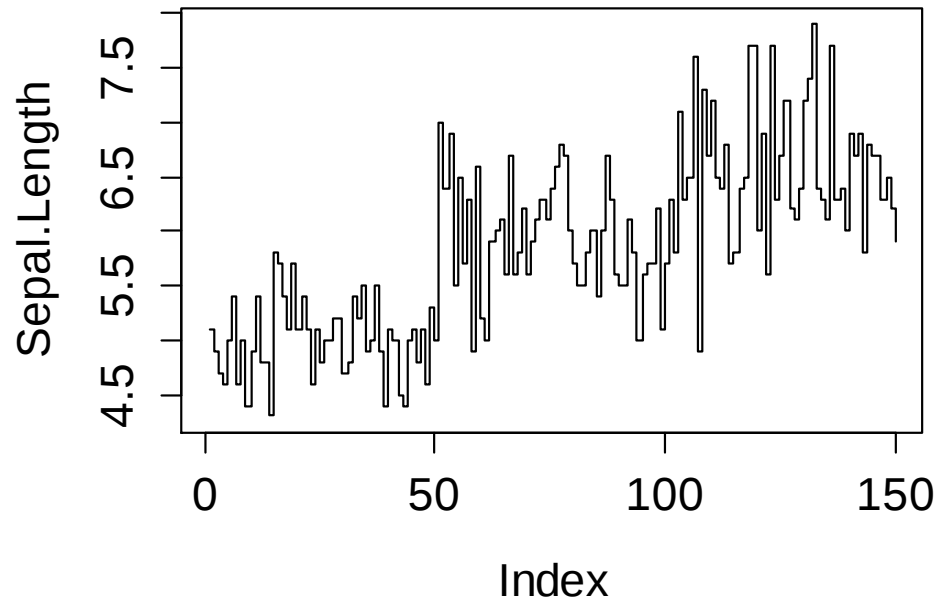
La función `plot()`

- `plot(Sepal.Length, type="h")`



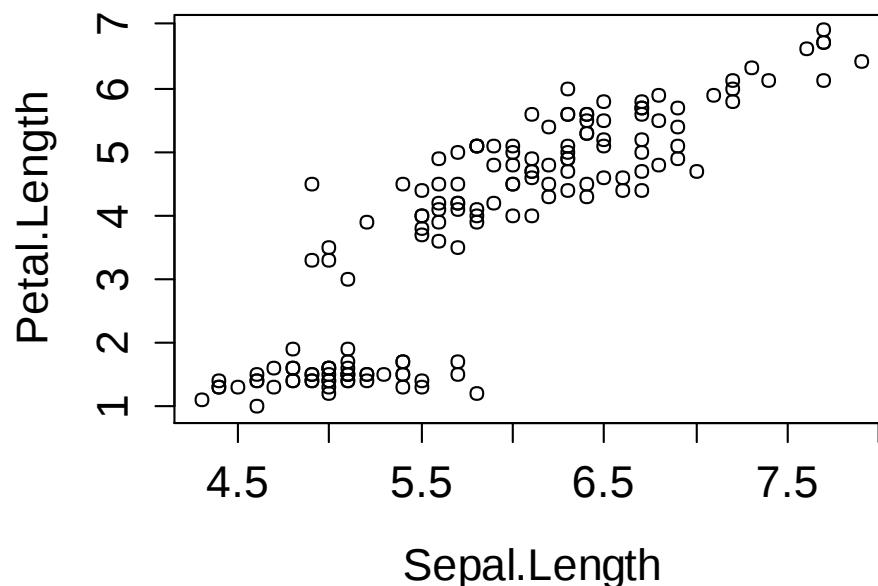
La función `plot()`

- `plot(Sepal.Length, type="s")`



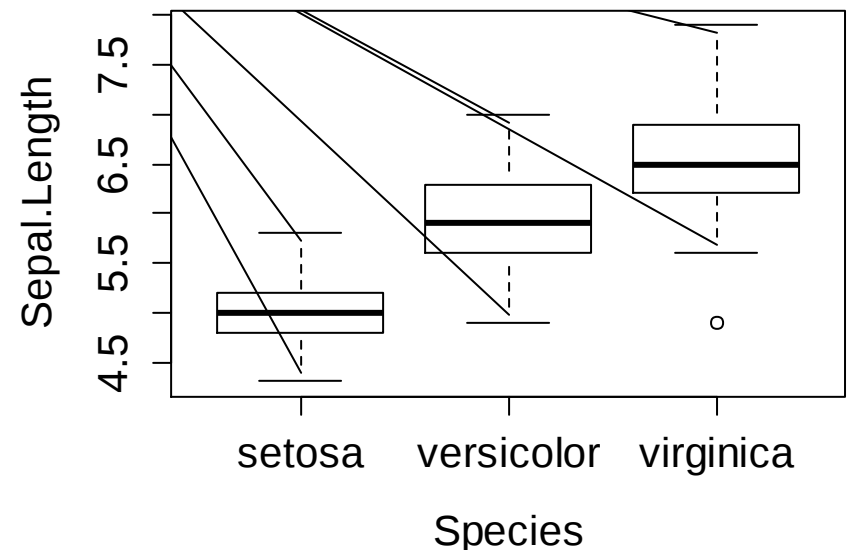
Gráficos de Dispersión

- Necesitamos dos variables (x e y)
- Como coordenadas (x, y)
 - `plot(Sepal.Length, Petal.Length)`
- Con interfaz tipo fórmula
 - `plot(Petal.Length ~ Sepal.Length)`



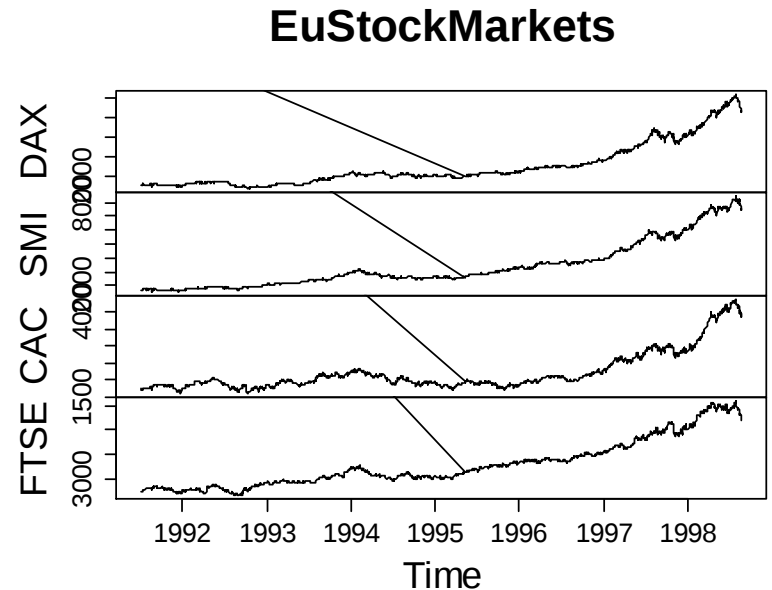
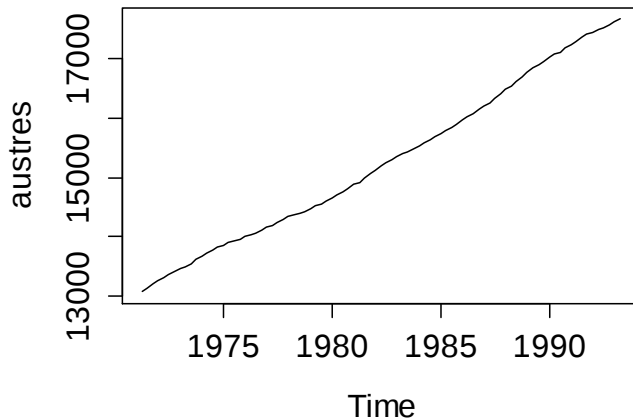
Gráficos de Caja

- Combinando un **factor** y una variable **cuantitativa**
- Aludiendo a lo representado en los ejes
 - `plot(Species, Sepal.Length)`
- Interfaz tipo fórmula
 - `plot(Sepal.Length ~ Species)`



Series Temporales

- Cuando se utiliza como objeto una serie temporal
 - `plot(austres)`
 - `plot(EuStockMarkets)`



Gráficos de Barras

- Cuando el argumento es una tabla
 - `plot(table(Sepal.Length))`

Frecuencia
absoluta

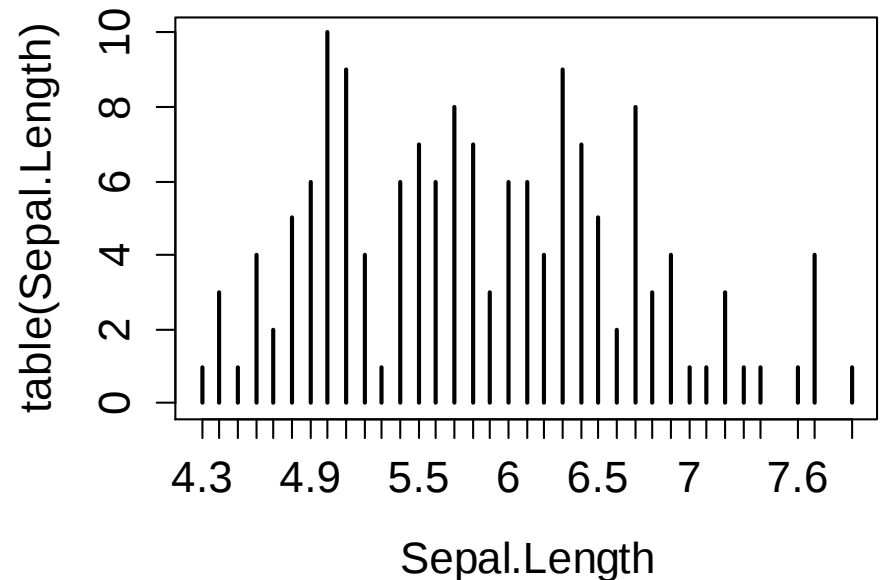
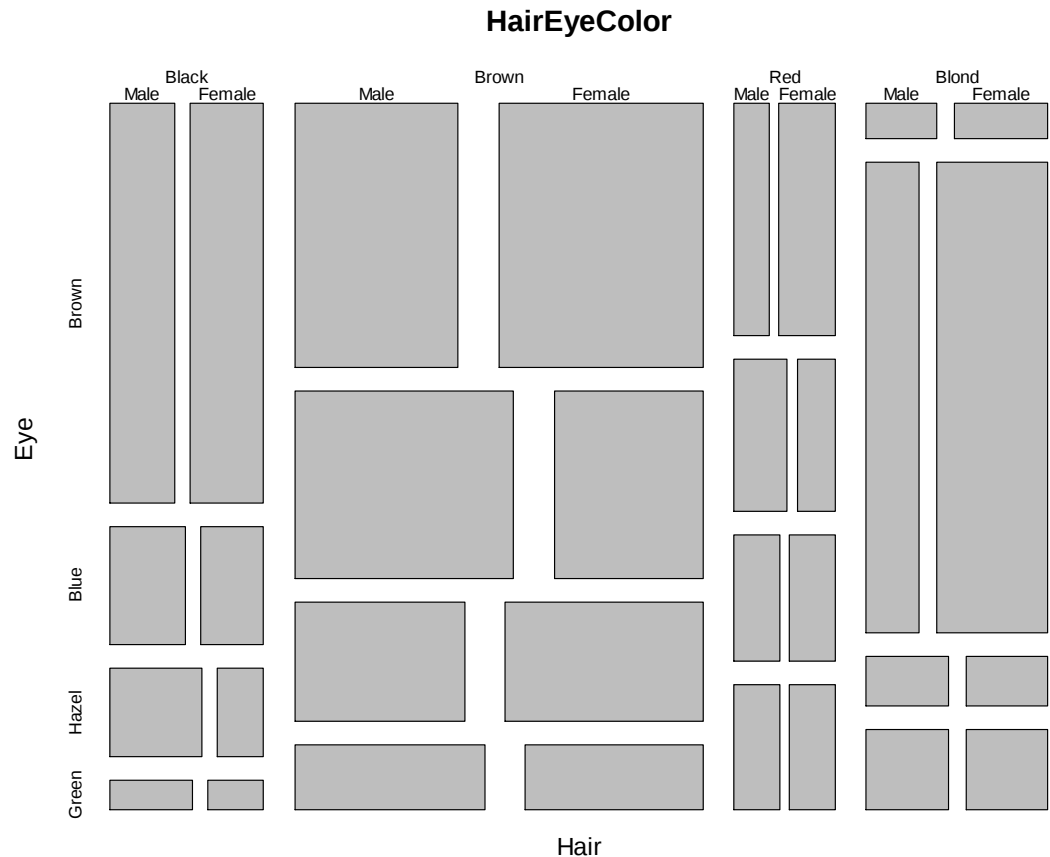


Gráfico de Mosaico

- Cuando el argumento es una tabla *bi* o *multi* dimensional
- Por ejemplo
 - HairEyeColor
 - plot(HairEyeColor)

Variables:

Color del Pelo
Color de los ojos
Sexo



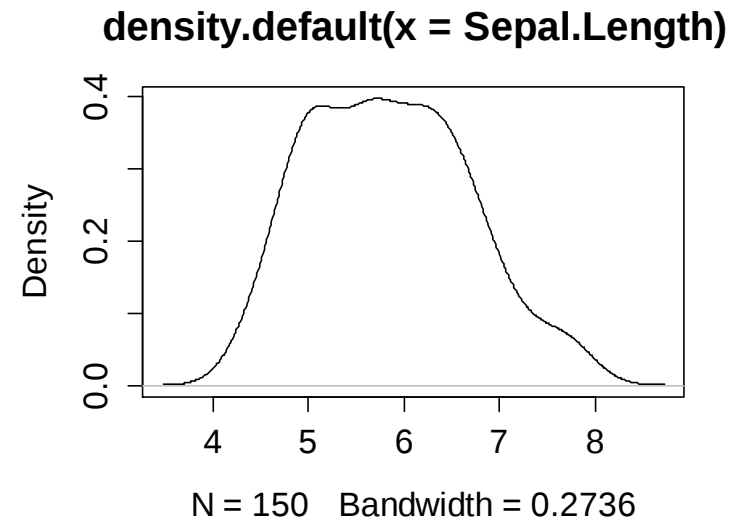
Gráficos de Densidad

- Podemos obtener la función de densidad de una variable utilizando la función `density()`
- Podríamos trazar el gráfico de densidad *incrustando* la función `density()` dentro de `plot()`

Consideremos a la variable

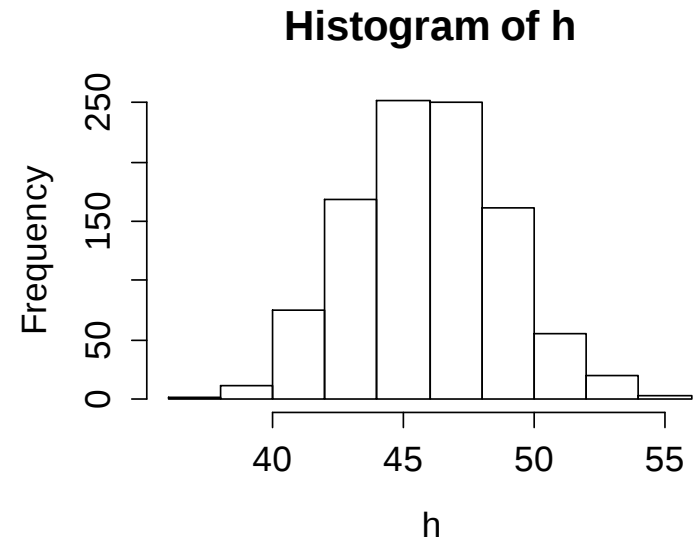
`Sepal.Length`

`plot(density(Sepal.Length))`



La función `hist()`

- Es una función genérica para dibujar histogramas
- Toma como argumento un vector
- Por ejemplo:
 1. `h = rnorm(1000, 46, 3)`
 2. `hist(h)`

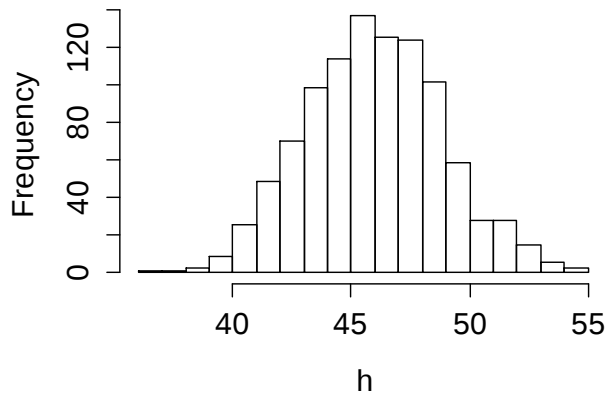


Opciones específicas de los histogramas

- La opción `breaks`

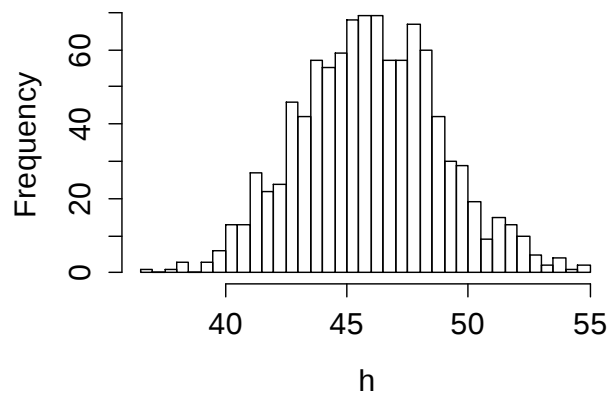
`hist(h, breaks=15)`

Histogram of h



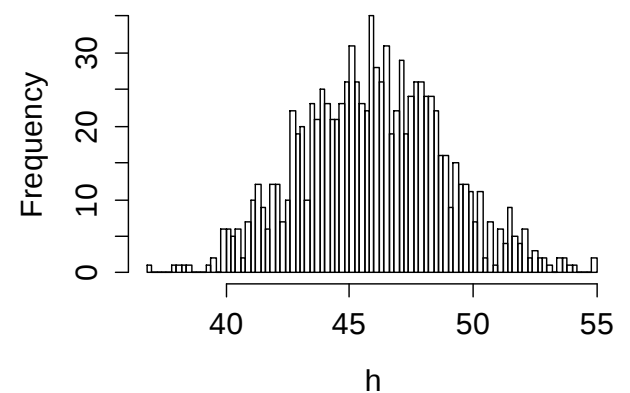
`hist(h, breaks=30)`

Histogram of h



`hist(h, breaks=70)`

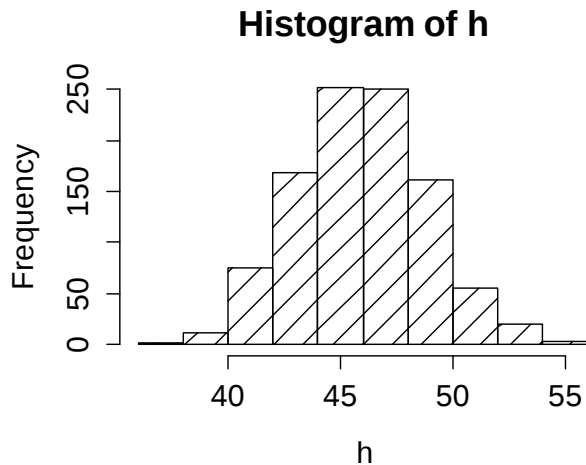
Histogram of h



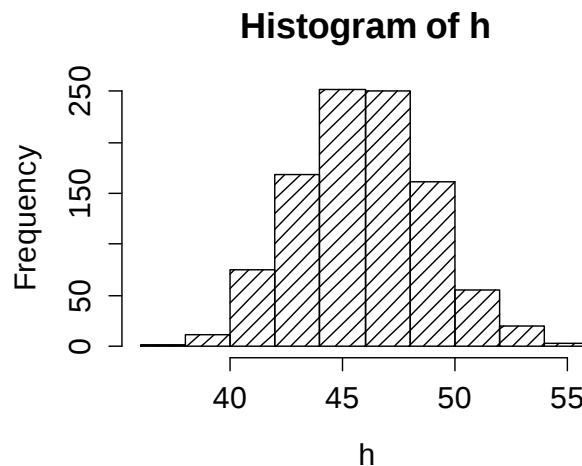
Opciones específicas de los histogramas

- La opción **density**

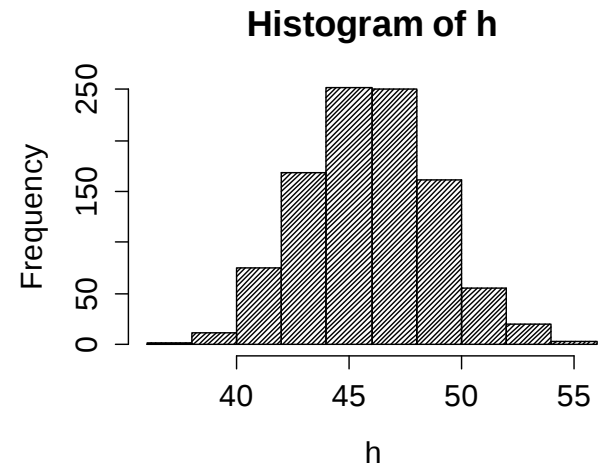
`hist(h, density=5)`



`hist(h, density=10)`



`hist(h, density=30)`

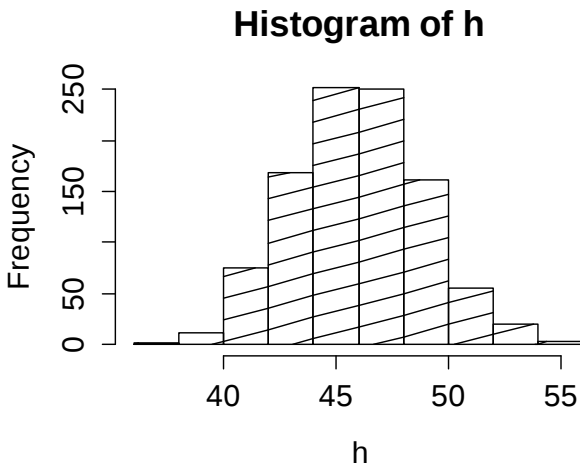


Opciones específicas de los histogramas

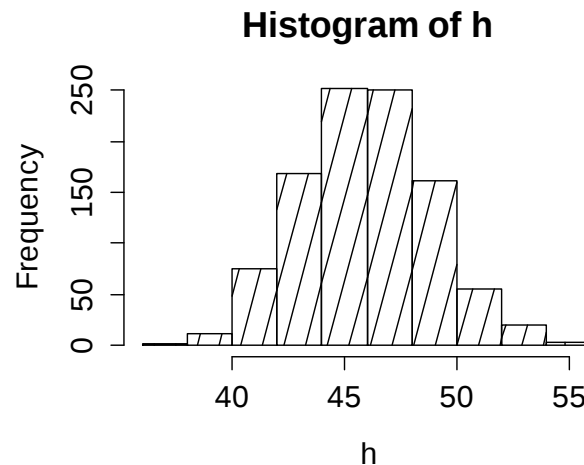
- La opción **angle**

density=5

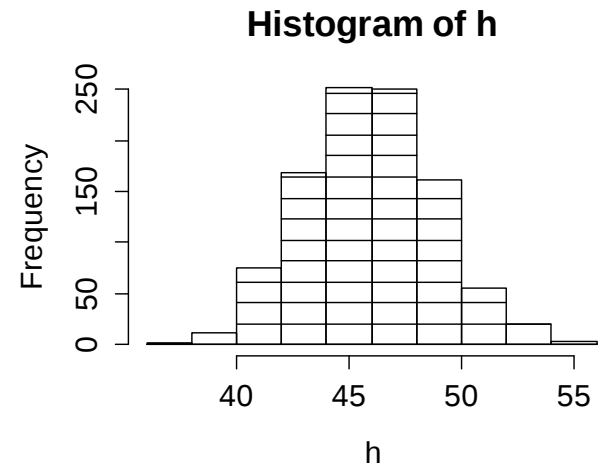
angle=15



angle=75



angle=180



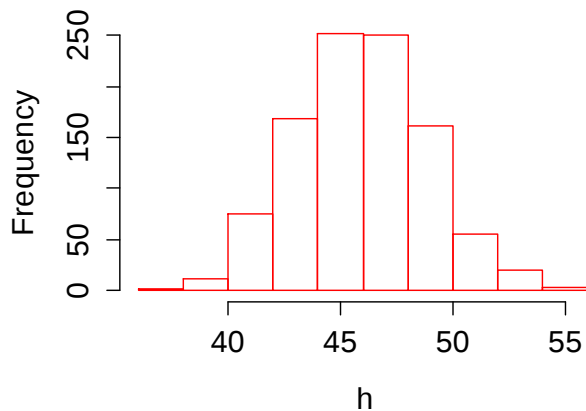
Opciones específicas de los histogramas

- La opción `border`

`colors()`

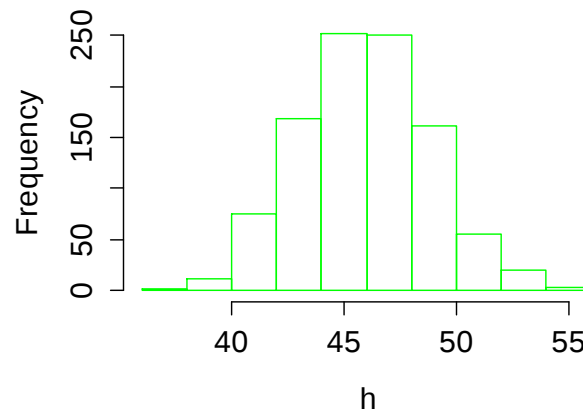
`hist(h, border="red")`

Histogram of h



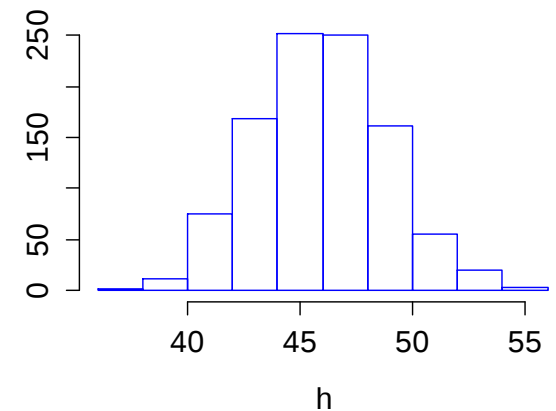
`hist(h, border="green")`

Histogram of h



`hist(h, border="blue")`

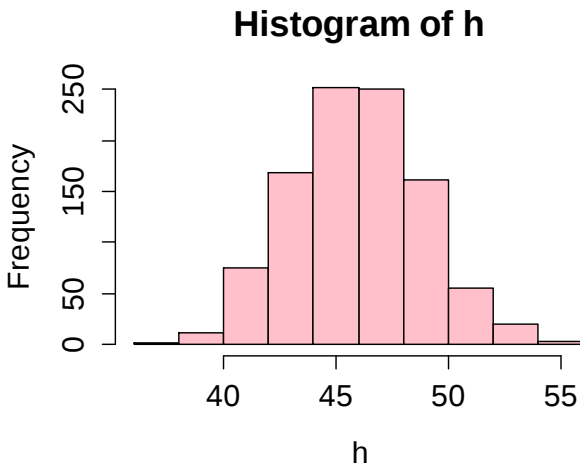
Histogram of h



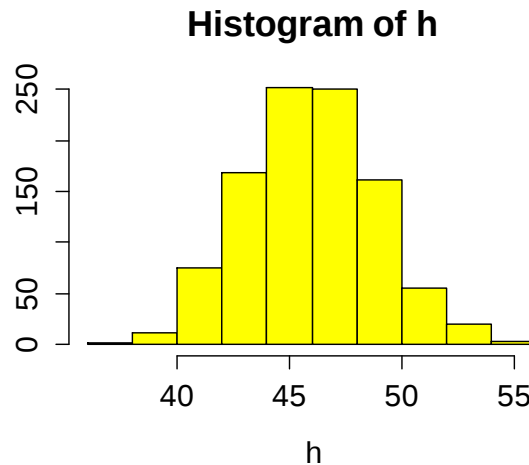
Opciones específicas de los histogramas

- La opción `col`

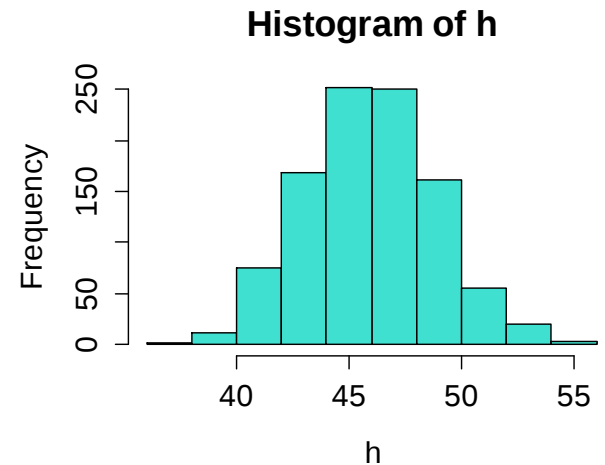
```
hist(h, col="pink")
```



```
hist(h, col="yellow")
```



```
hist(h, col="turquoise")
```

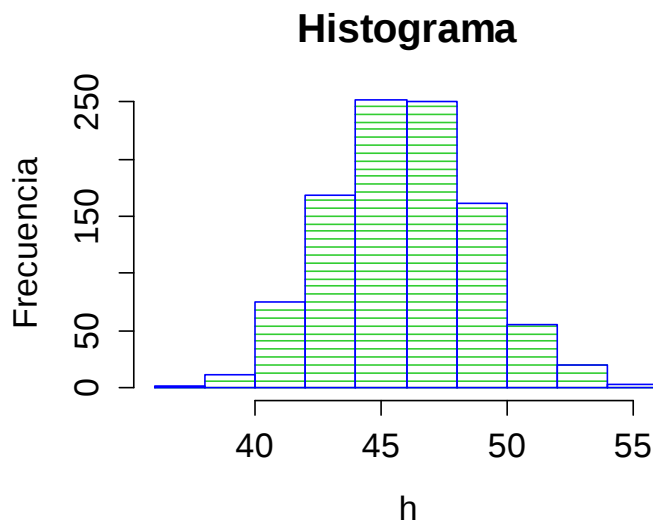


Varias opciones en un mismo histograma

- Las opciones que hemos visto (y más) se pueden combinar

- Ejemplo:

```
hist(h,  
     col = "limegreen",  
     border = "blue",  
     density = 15,  
     angle = 180,  
     main = "Histograma",  
     ylab = "Frecuencia"  
)
```

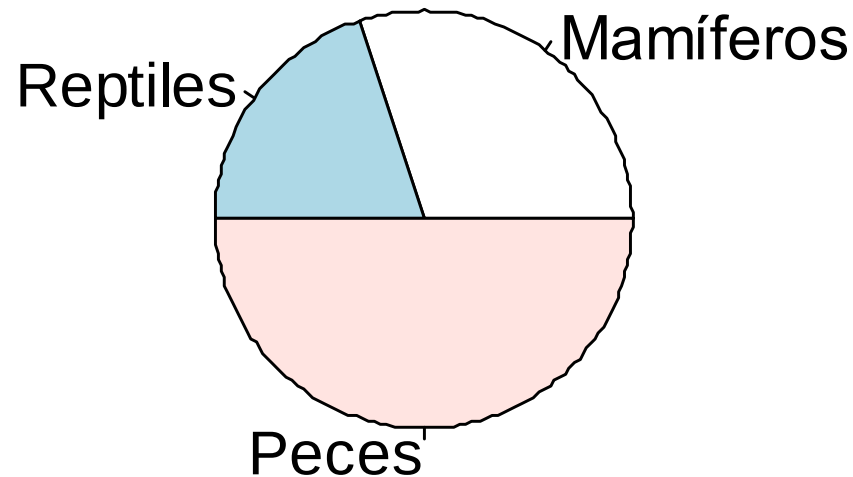


La función `pie()`

- Sirve para crear gráficos de sectores
- El argumento principal es `x`
 - Vector de cantidades no negativas – áreas del gráfico
- Otro argumento interesante es `labels`
- Ejemplo:

```
pie(x=c(3,2,5), labels=c("Mamíferos", "Reptiles", "Peces"))
```

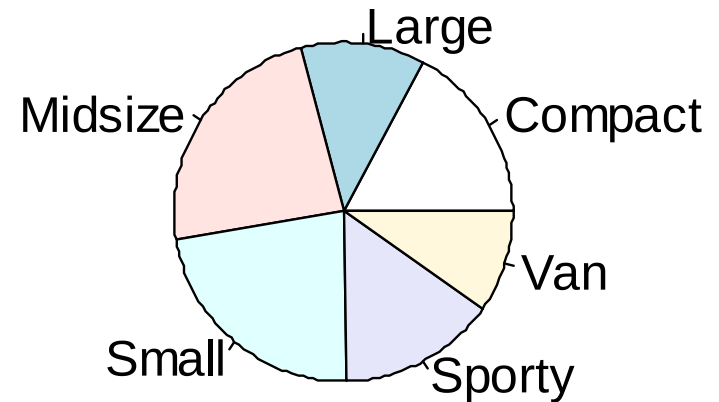
La función `pie()`



```
pie(x=c(3,2,5), labels=c("Mamíferos", "Reptiles", "Peces"))
```

La función `pie()`

- ¿Cómo funciona con bases de datos?
- Tenemos que crear una tabla de frecuencias
- Ejemplo:

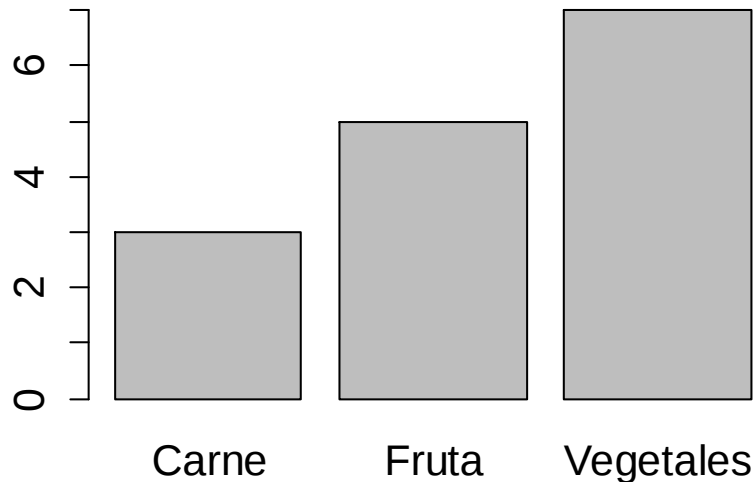


```
data(Cars93, package="MASS")  
attach(Cars93)  
pie(table(Type))
```

La función `barplot()`

- Sirve para generar gráficos de barras
- El argumento principal son alturas de barras - `height`
- El argumento que controla las etiquetas de las barras es `names.arg`
- Ejemplo:

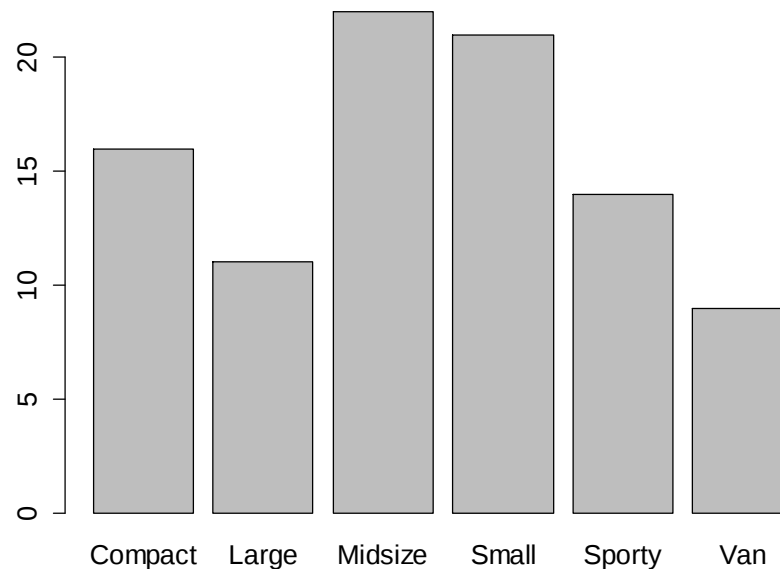
```
barplot(height=c(3,5,7),names.arg=c("Carne","Fruta","Vegetales"))
```



La función `barplot()`

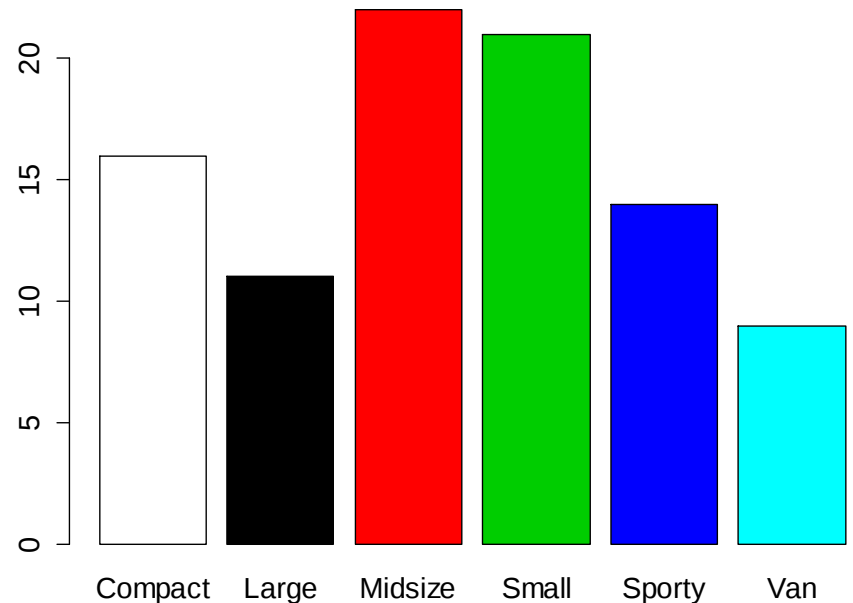
- ¿Cómo lo haríamos desde una base de datos?
- Tendríamos que generar una tabla de frecuencias
- Ejemplo:
- Utilizando la base de datos `Cars93` y la variable `Type`

`barplot(table(Type))`



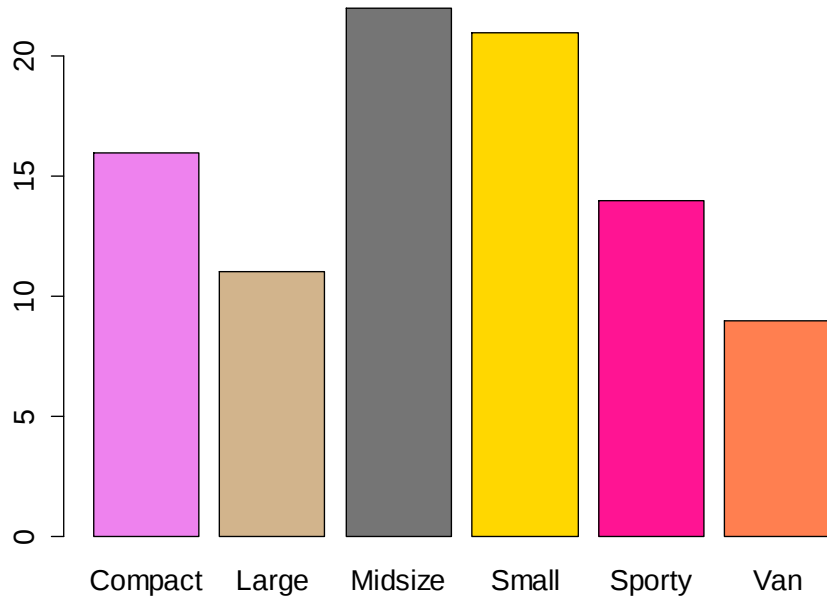
Dar color

- Con números – del 0 al 8
- Con su nombre – lista en `colors()`
- Paletas específicas
 - `rainbow`
 - `topo.colors`
 - `cm.colors`

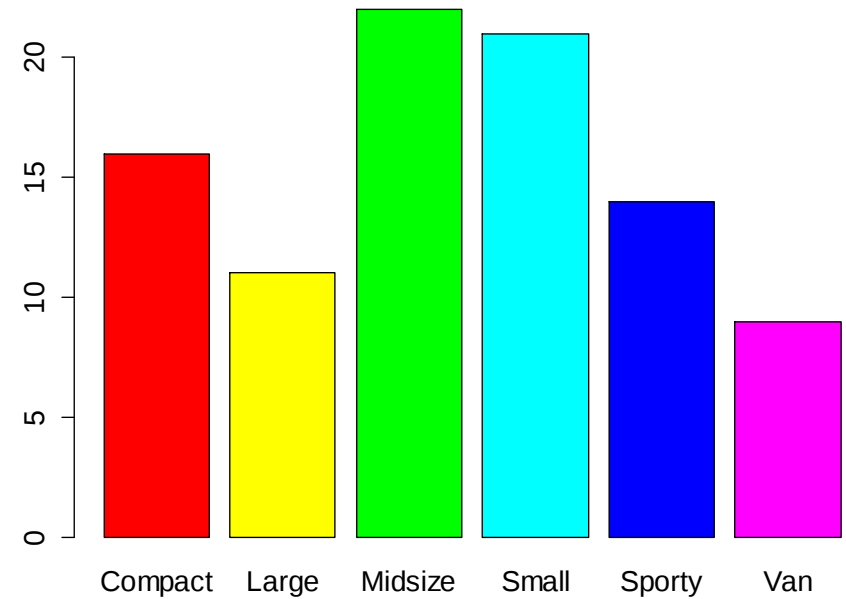


```
barplot(table(Type), col=0:5)
```

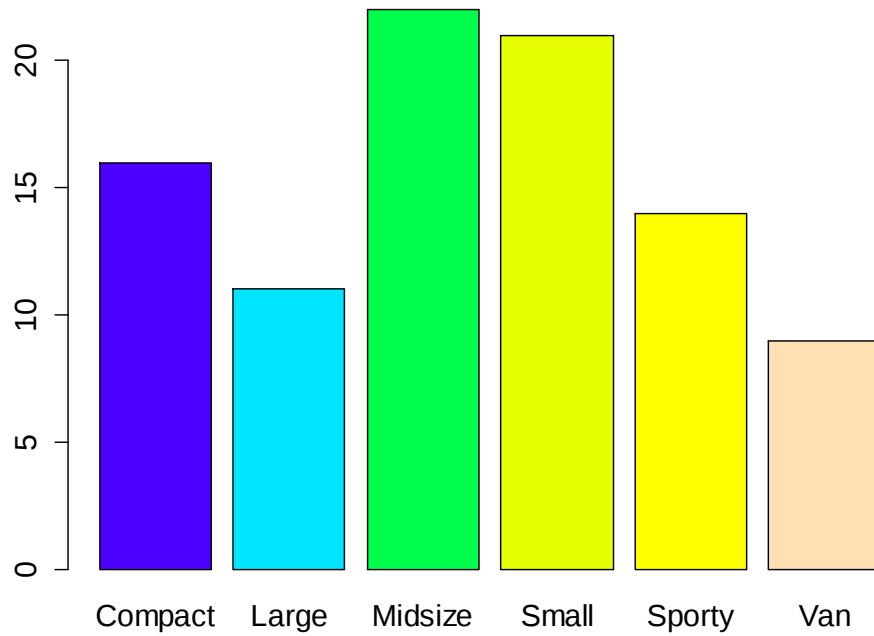
```
barplot(table(Type), col=c("violet","tan","gray46","gold","deeppink","coral"))
```



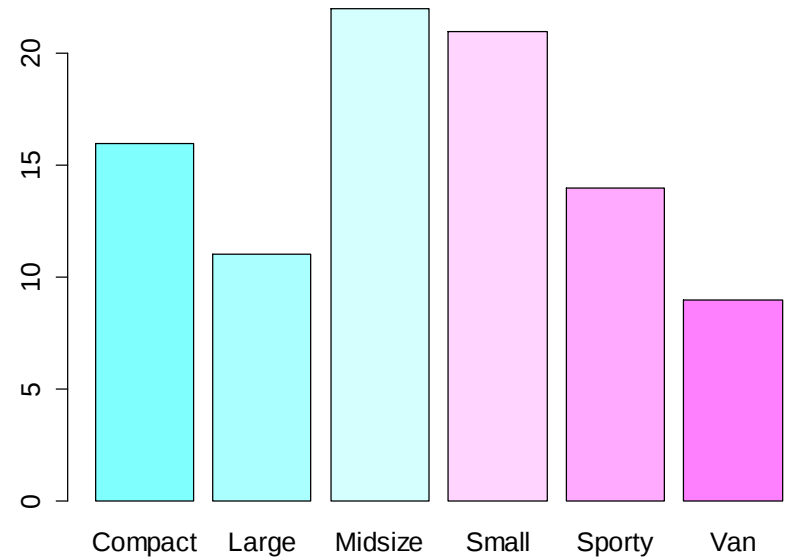
```
barplot(table(Type), col=rainbow(6))
```



```
barplot(table(Type), col=topo.colors(6))
```



```
barplot(table(Type), col=cm.colors(6))
```

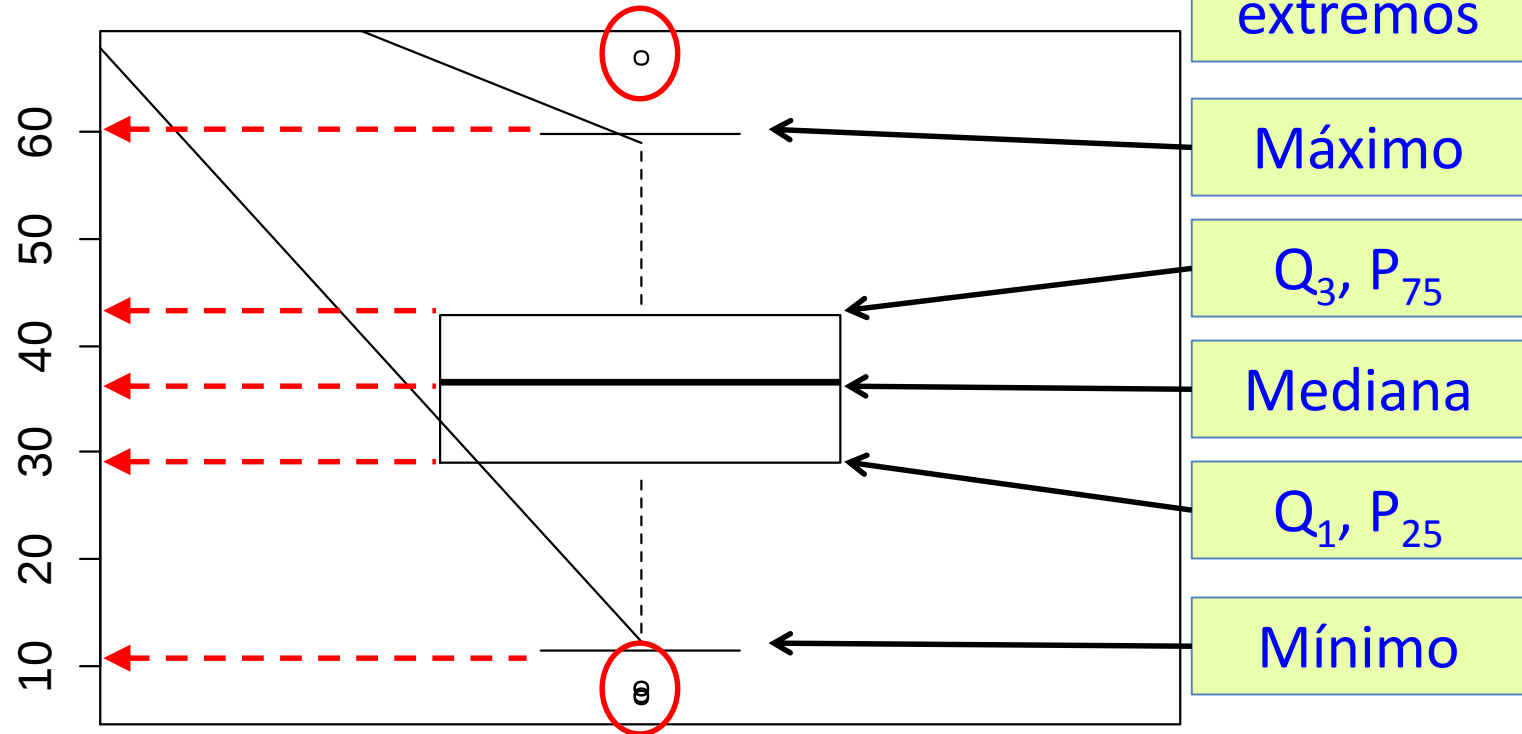


¿Qué es un diagrama de caja?

- Una caja que con sus bordes delimita ciertos cuantiles
- En concreto señala dónde están
 - El mínimo – *bajo ciertas condiciones*
 - El máximo – *bajo ciertas condiciones*
 - Los cuartiles – Percentiles 25, 50 y 75
 - Datos extremos, alejados o atípicos
- Ejemplo:

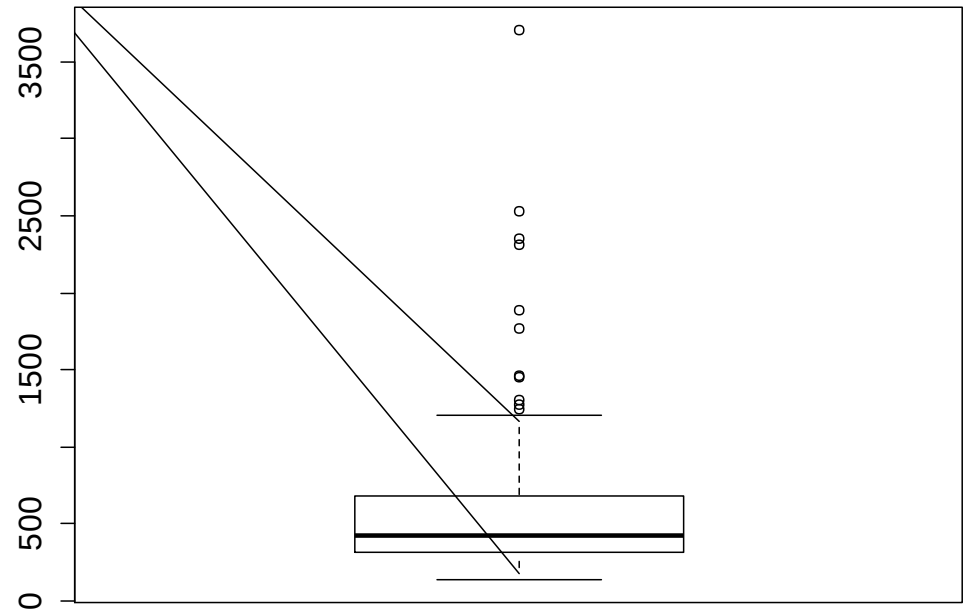
¿Qué es un diagrama de caja?

precip



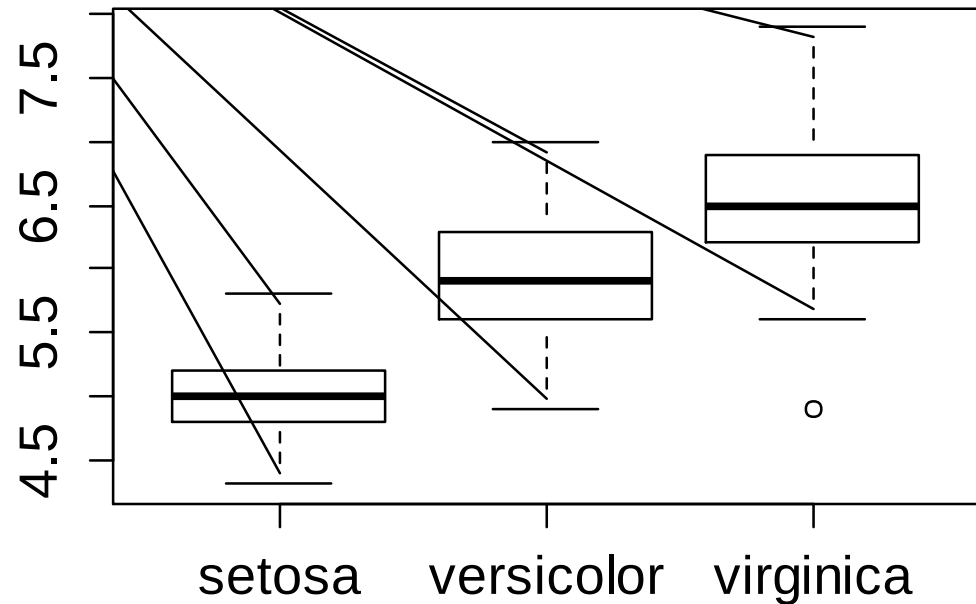
La función **boxplot()**

- Sirve para dibujar gráficos de caja
- Su principal argumento es **x**
- Ejemplo:
boxplot(rivers)



La función `boxplot()`

- Puede tomar una interfaz de tipo fórmula
- Ejemplo:

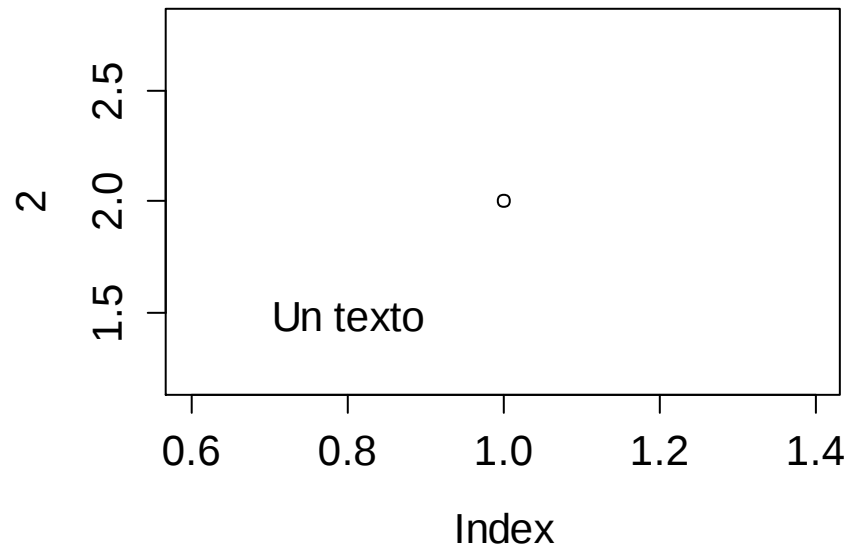


```
boxplot(Sepal.Length ~ Species)
```

La función `text()`

- Sirve para añadir texto en coordenadas concretas

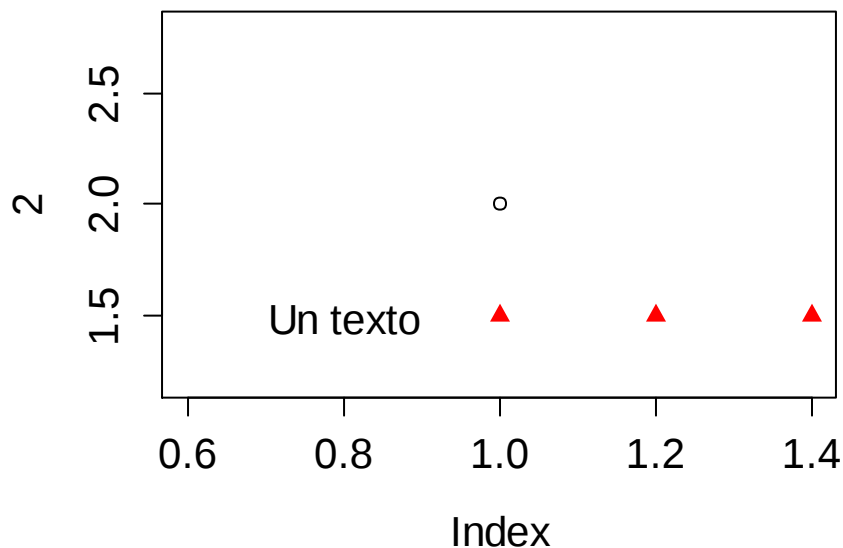
`text(0.8, 1.5, "Un texto")`



La función `points()`

- Para añadir puntos en coordenadas concretas

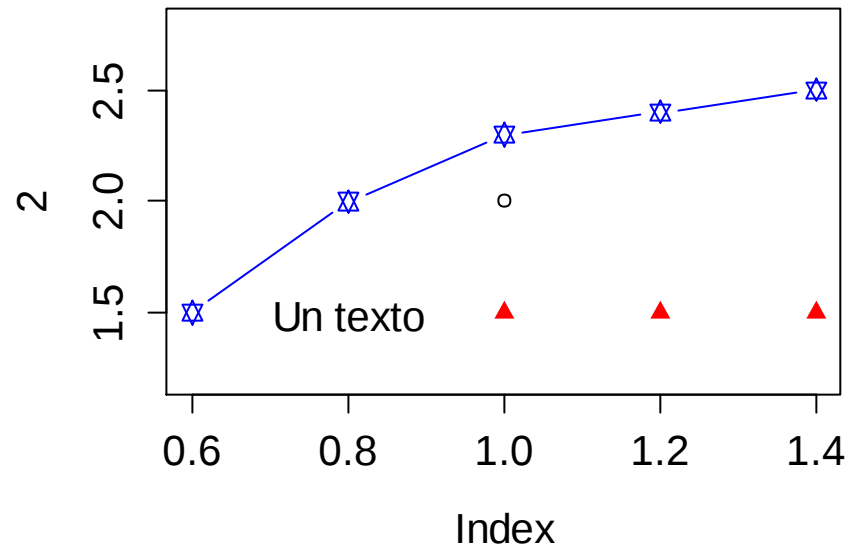
```
points(  
  x = c(1,1.2,1.4),  
  y = c(rep(1.5,3)),  
  pch = 17,  
  col = "red"  
)
```



La función **lines()**

- Sirve para añadir líneas

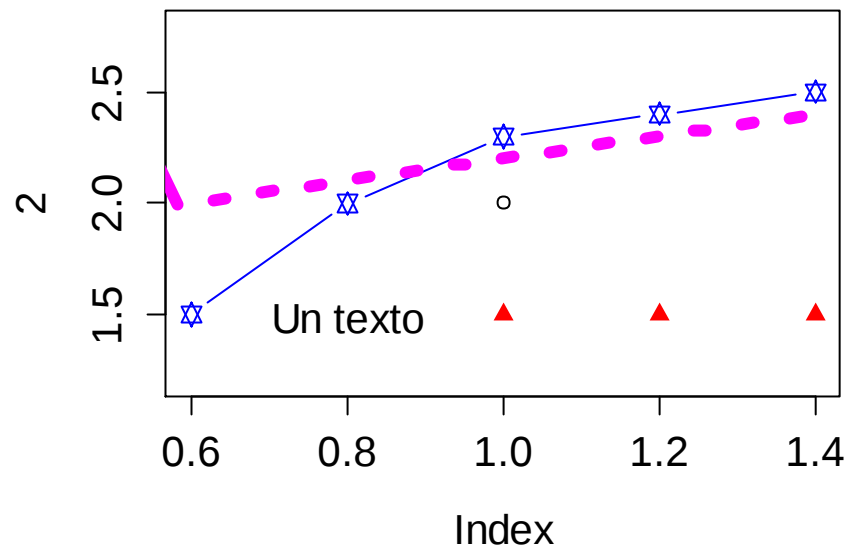
```
lines(  
  x = c(0.6,0.8,1,1.2,1.4),  
  y = c(1.5,2,2.3,2.4,2.5),  
  type = "b",  
  col = 4,  
  pch = 11  
)
```



La función `abline()`

- Añade una línea con intercepto y pendiente arbitrarios

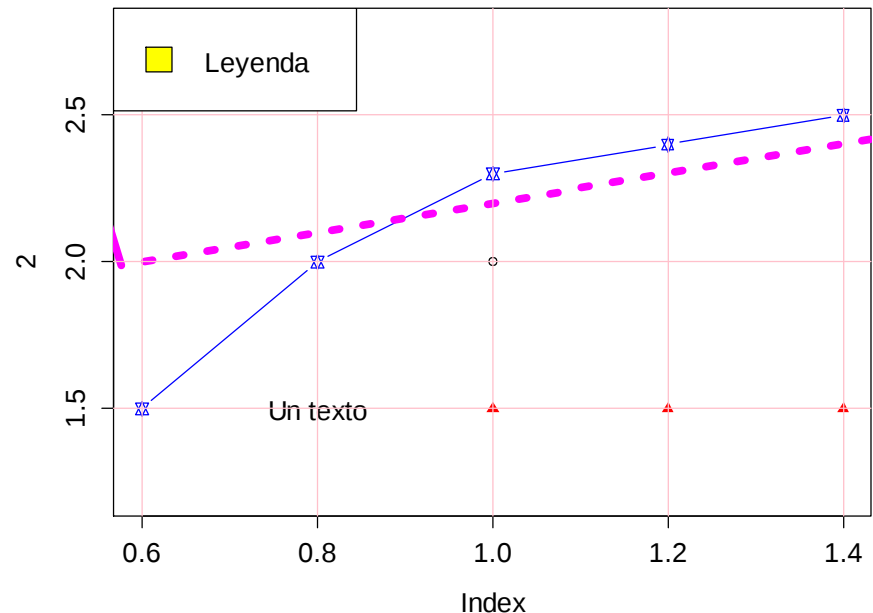
```
abline(  
  a = 1.7,  
  b = 0.5,  
  lwd= 6,  
  lty = 3,  
  col = 6  
)
```



La función **legend()**

- Sirve para añadir leyendas

```
legend(  
    x = "topleft",  
    legend = "Leyenda",  
    fill = "yellow"  
)
```



La función `title()`

- Sirve para añadir o modificar títulos

```
title(  
    main = "Es una prueba",  
    sub = "Base",  
    col.main = "green"  
)
```

