Ferramentas de Mineração de Dados

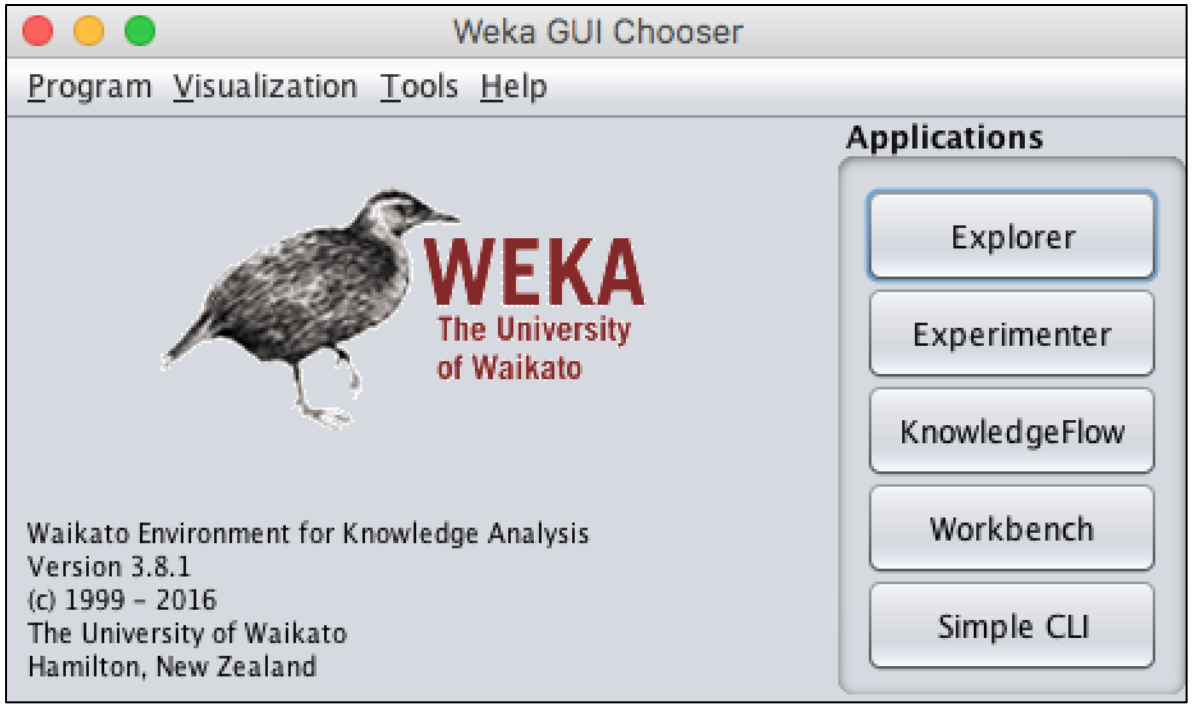
Profª: Solange Oliveira Rezende

Ferramentas

- Existem várias ferramentas para Mineração de Dados ou Textos
- Nesta aula:
  - Weka
    - Pré-processamento, Classificação, Agrupamento e Regras de Associação
  - Apriori
    - Regras de Associação
  - Torch
    - Pré-processamento, Agrupamento de Textos e Visualização
  - Outras ferramentas

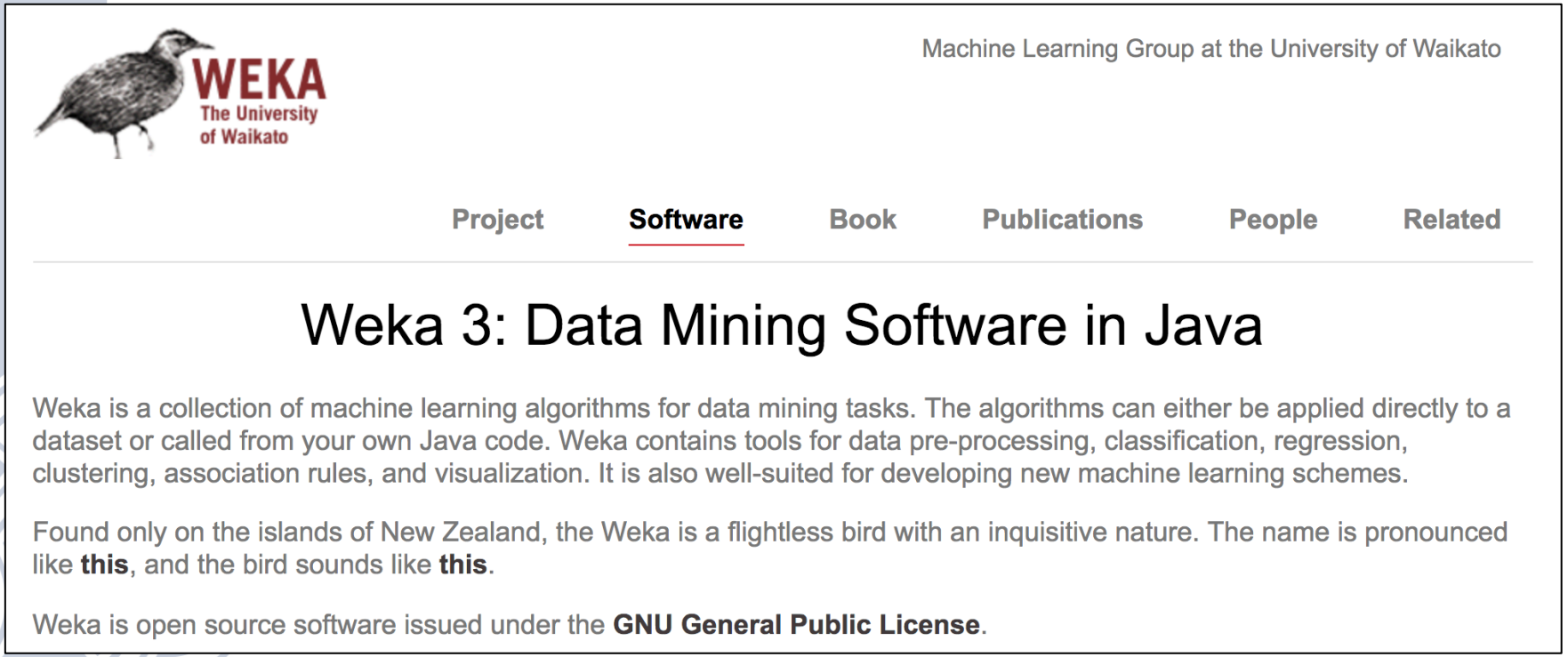
Weka – Waikato Environment for Knowledge Analysis

- Software popular e código aberto de aprendizagem de máquina.
- Pode ser utilizado através da interface gráfica, linha de comando ou Java API.



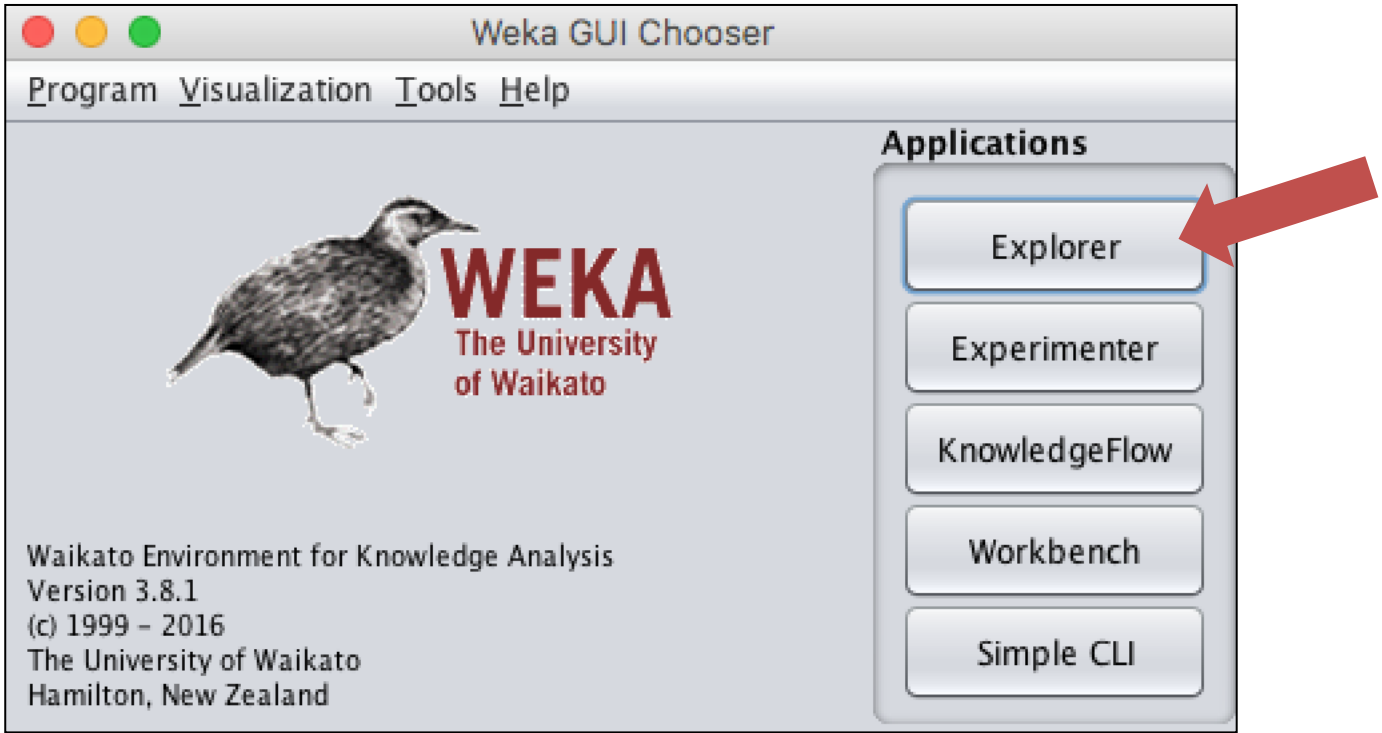
Weka – Instalação

- Instalação simples.
- Versões para Windows, Linux e Mac.
- Site: https://www.cs.waikato.ac.nz/ml/weka/



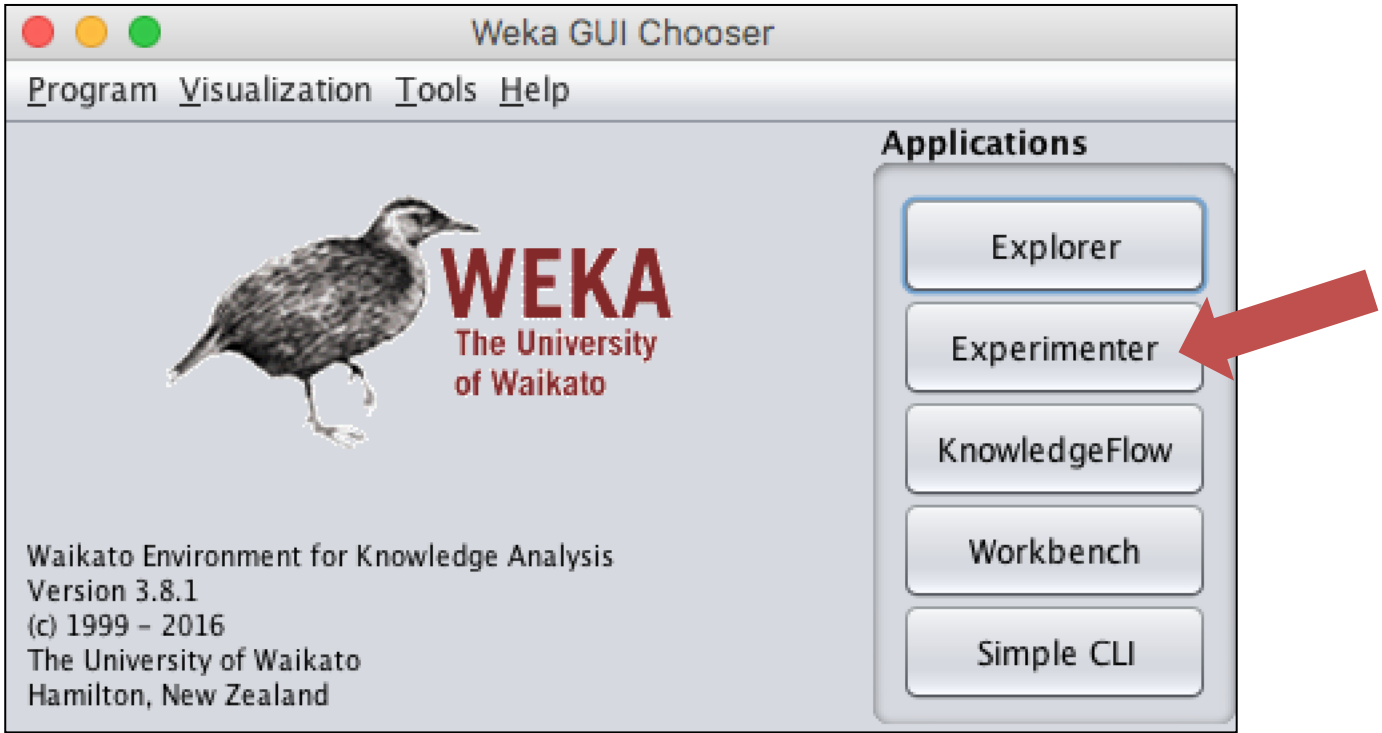
Weka

- Explorer (aplicação principal)
  - Explorar o conjunto de dados, visualizar dados, aplicar filtros;
  - Realizar classificação, agrupamento, regras de associação e seleção de atributos.



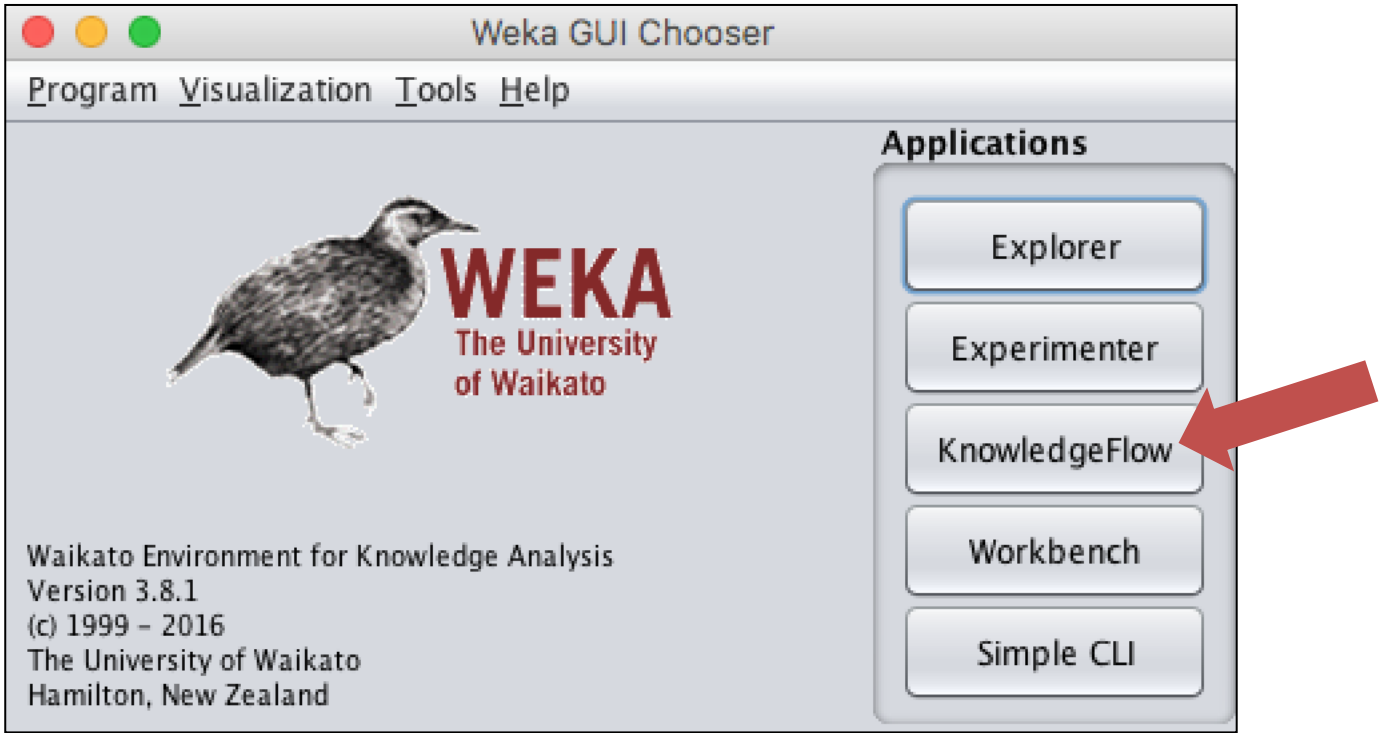
Weka

- Experimenter
  - Permite a avaliação de desempenho de algoritmos diferentes em base de dados diferentes.



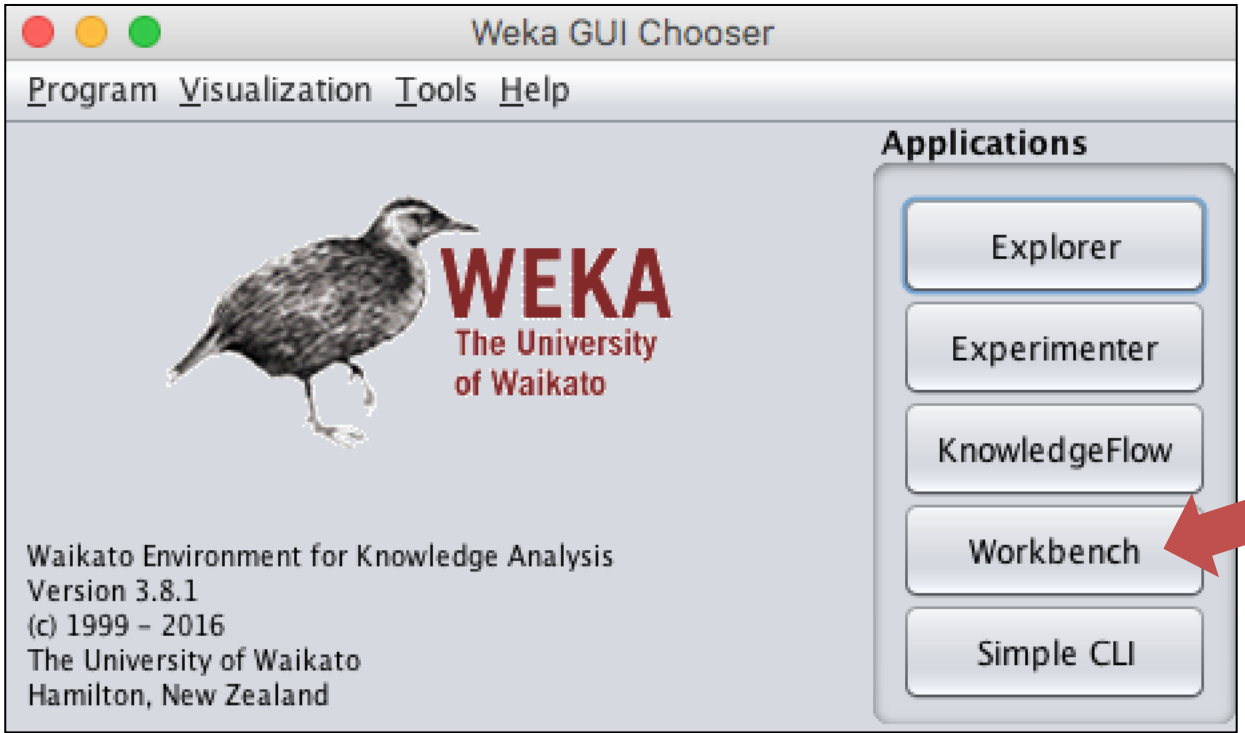
Weka

- KnowledgeFlow
  - Permite construir o processo de aprendizado de máquina na forma de fluxograma, arrastando e configurando atividades. Essencialmente o mesmo que o Experimenter, porém com interface que permite arrastar e soltar.



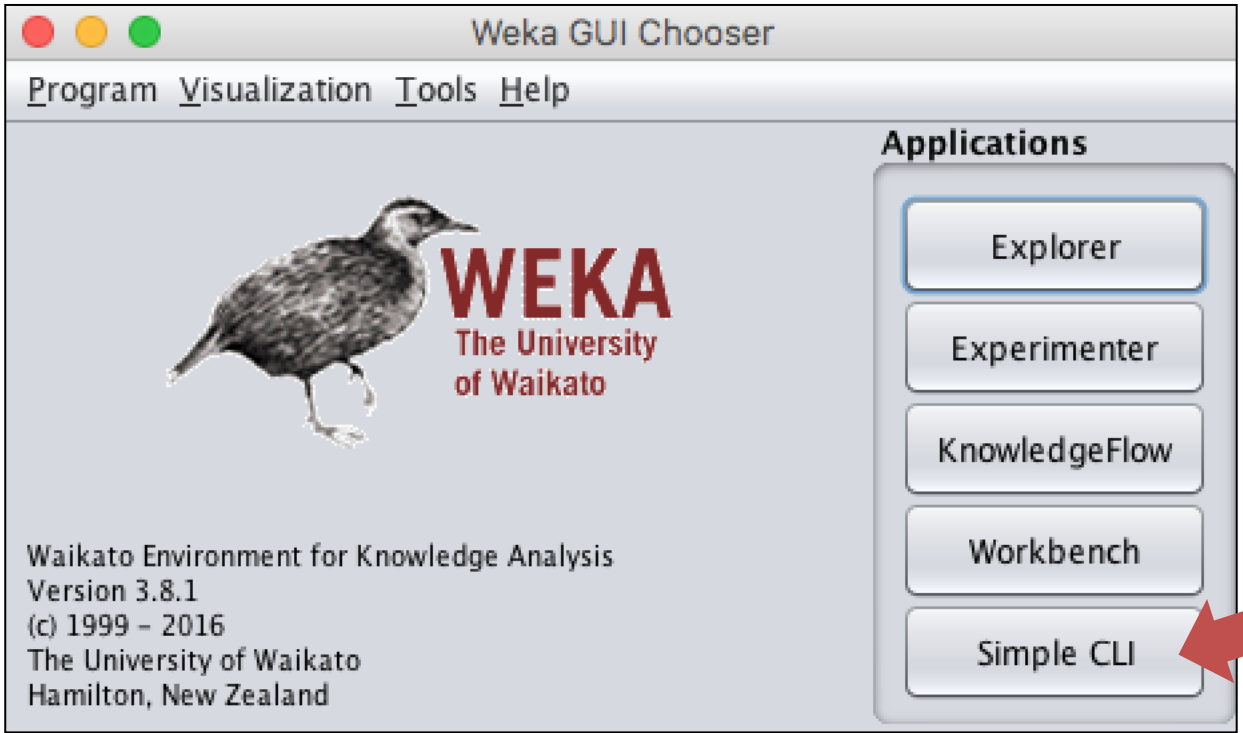
Weka

- Workbench
  - Aplicação *all-in-one* que combina toda, porém o usuário é livre para configurar a interface.



Weka

- Simple CLI
  - Permite utilizar o Weka através de linha de comando.



Weka

- Funcionalidades
  - Pré-processamento
  - Classificação
  - Regressão
  - Agrupamento
  - Regras de Associação
  - Visualização

- Distribuído sob a licença GPL

Weka

- Os dados podem estar nos formatos: ARFF, CSV, entre outros.
- O formato de arquivo de dados próprio da ferramenta WEKA é o **ARFF** (Attribute-Relation File Format)

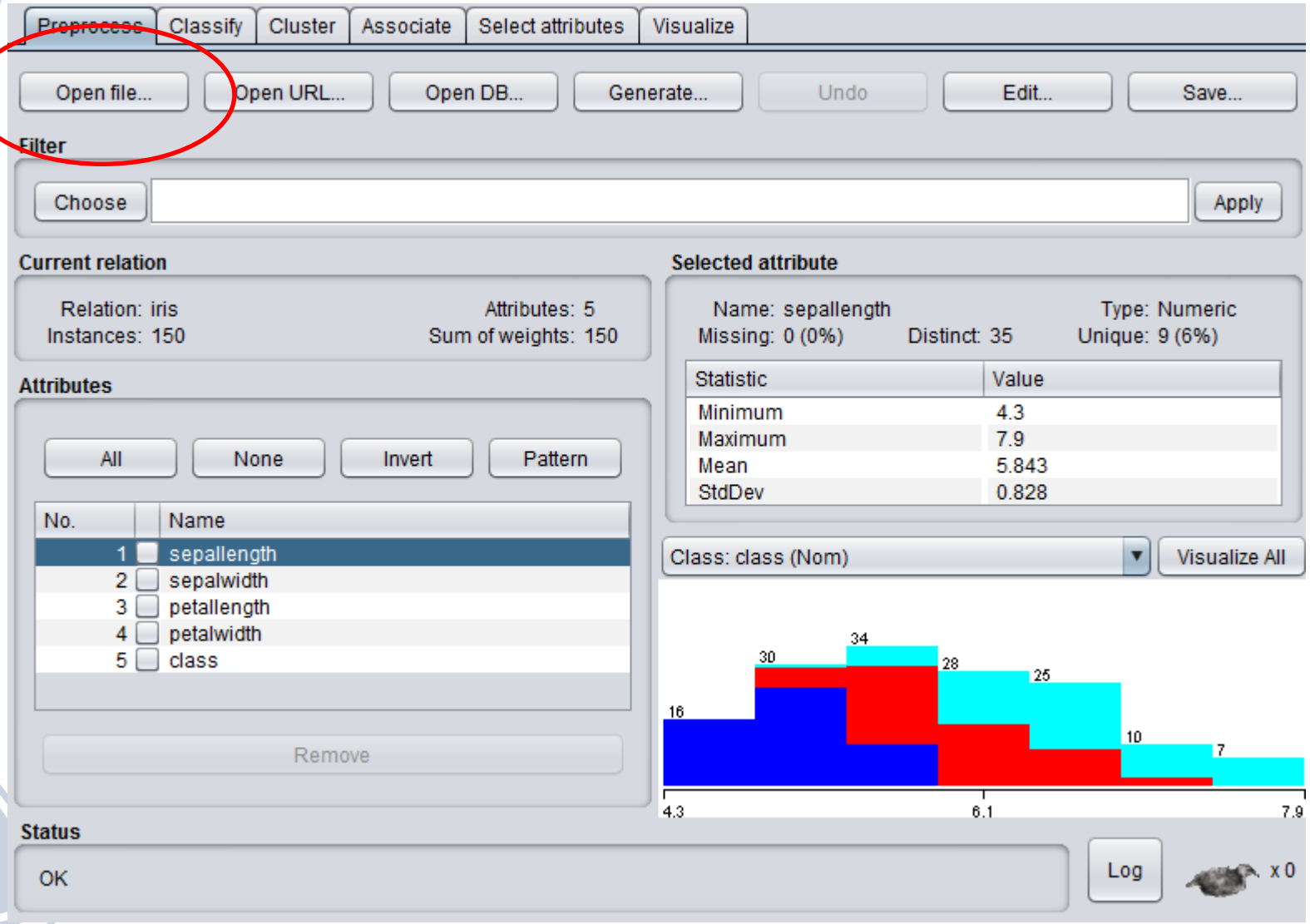
Weka – Exemplo de ARFF

```
%Exemplo do formato ARFF
@relation alunos_graduacao

@attribute nome string
@attribute idade numeric
@attribute sex {fem,masc}
@attribute notaP1 numeric
@attribute class {aprovado,reprovado}

@data
Roberta,25,fem,10.0,aprovado
Pedro,20,masc,8.0,aprovado
Maria,22,fem,?,reprovado
Joana,25,fem,9.0,aprovado
```

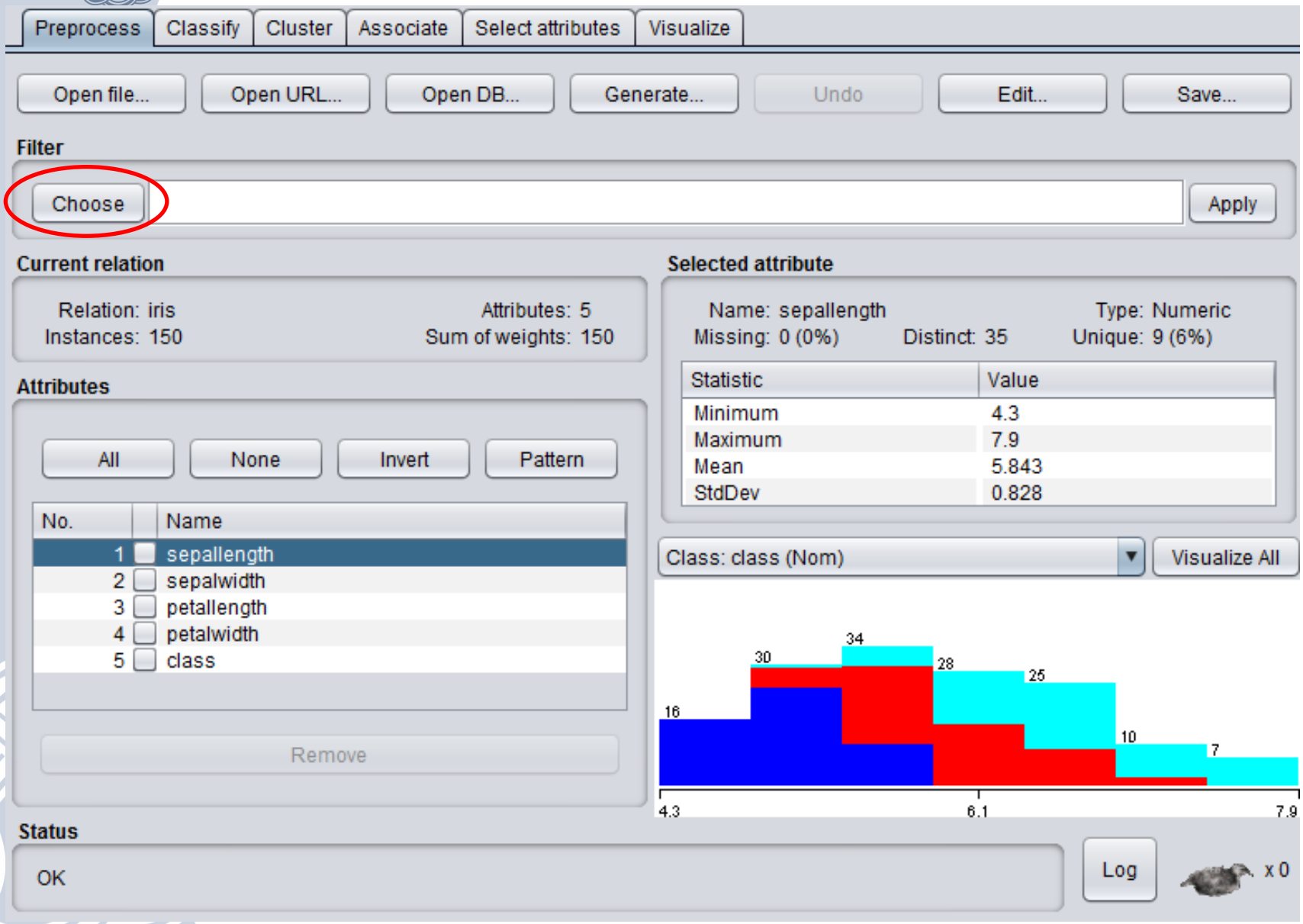
Weka – Carregamento dos Dados



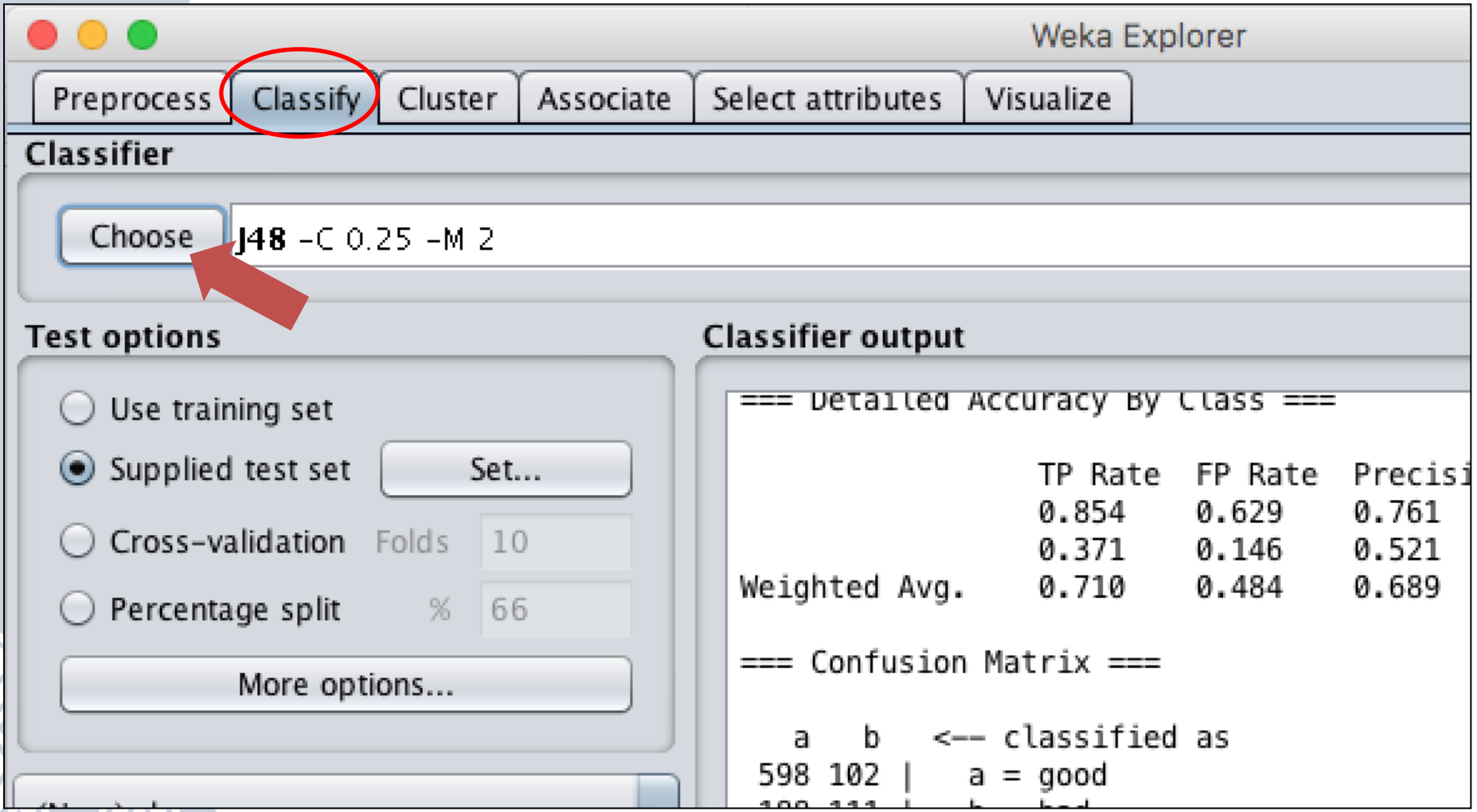
Weka – Pré-processamento

- As ferramentas de pré-processamento do Weka são chamadas de filtros
  - Discretização
  - Normalização
  - Seleção de atributos
  - Reamostragem

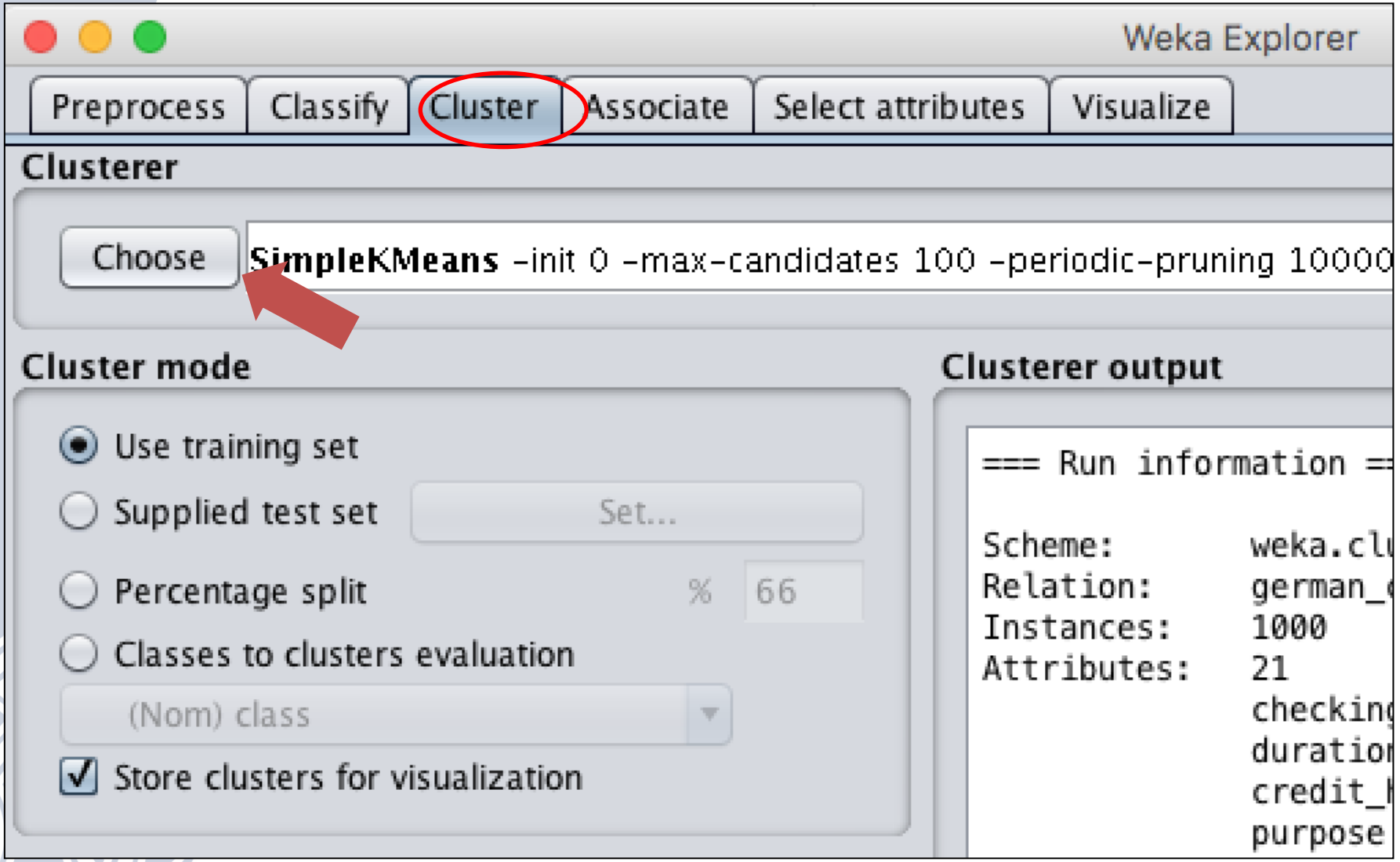
Weka – Pré-processamento



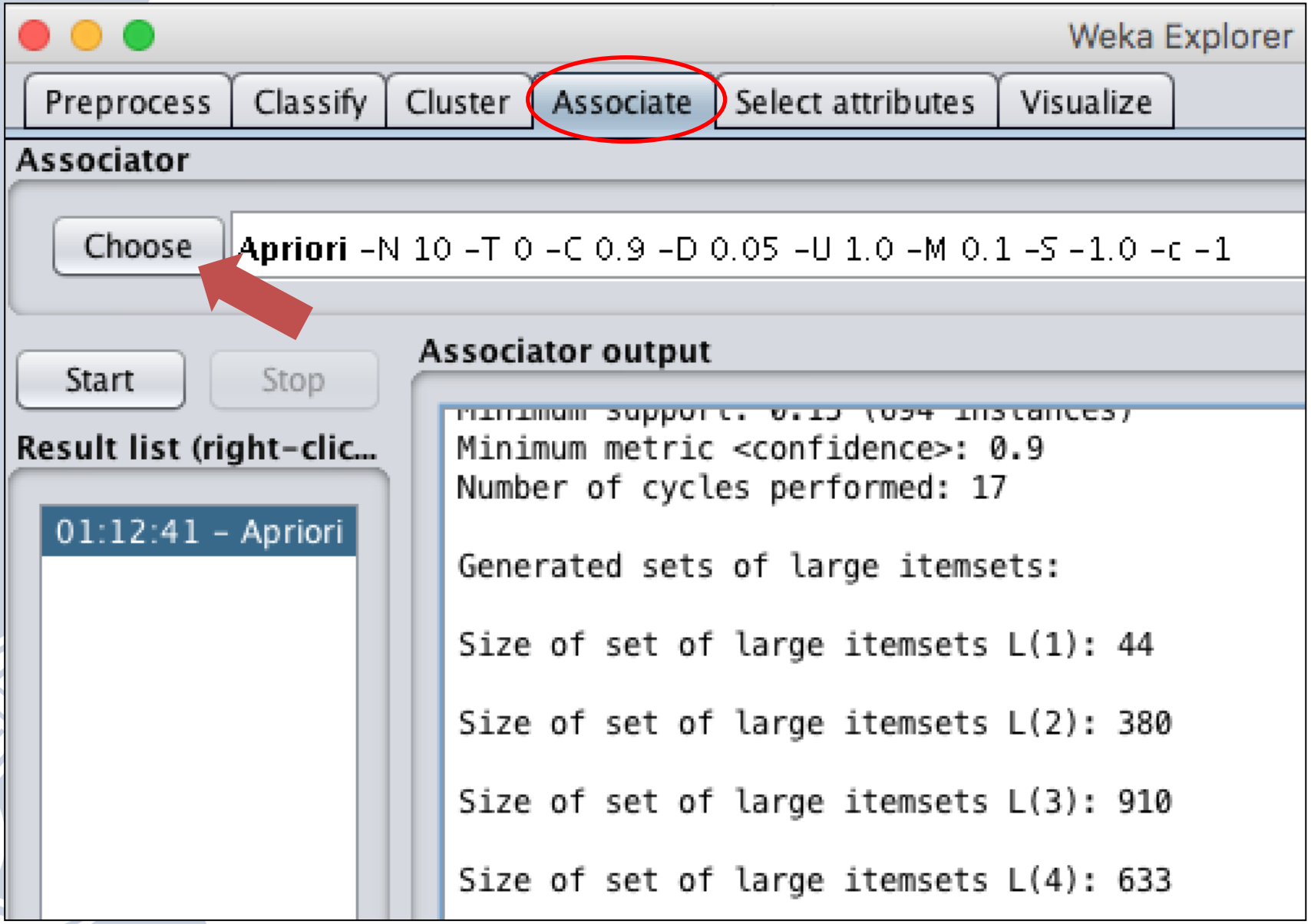
Weka - Classificação



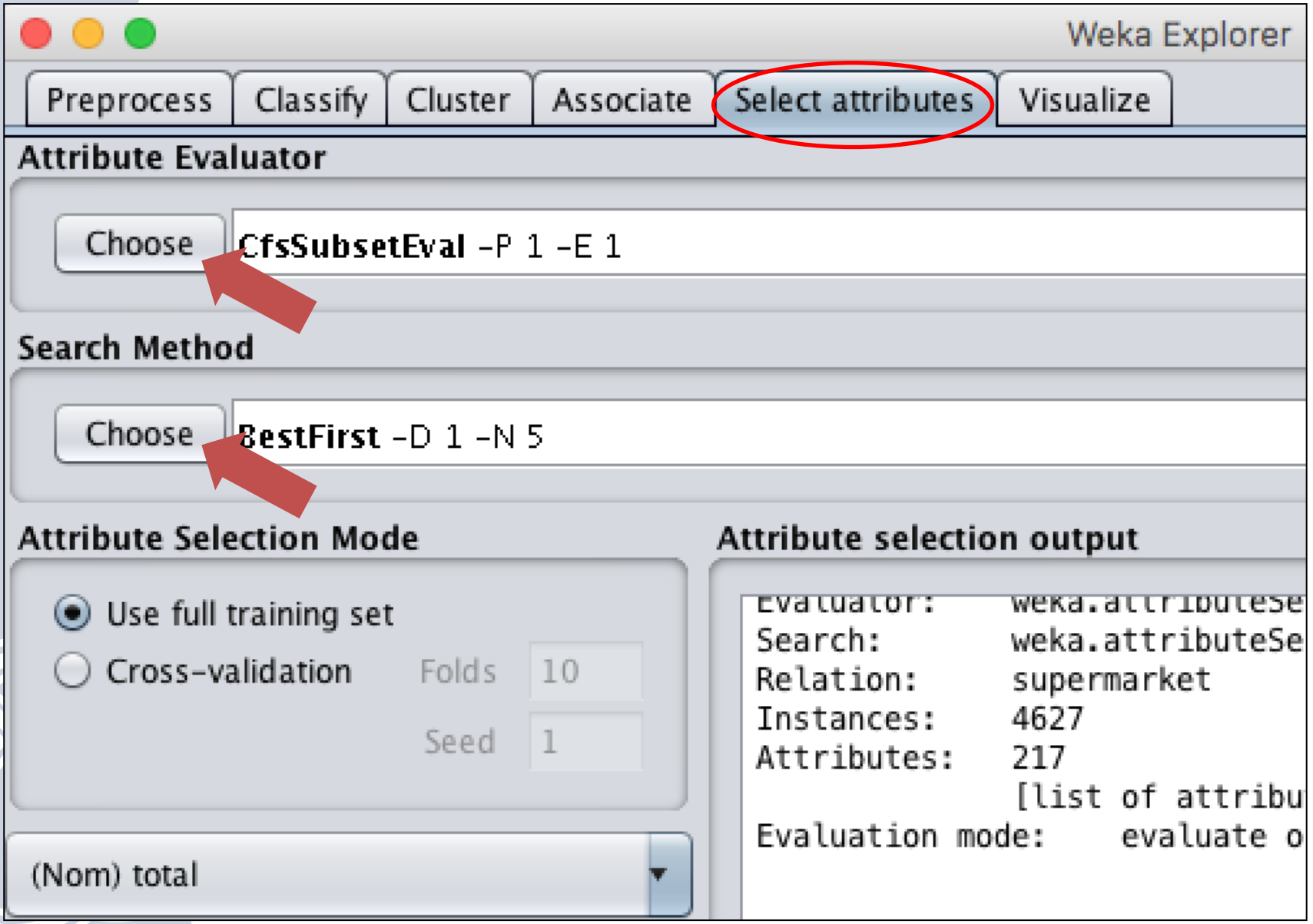
Weka - Agrupamento



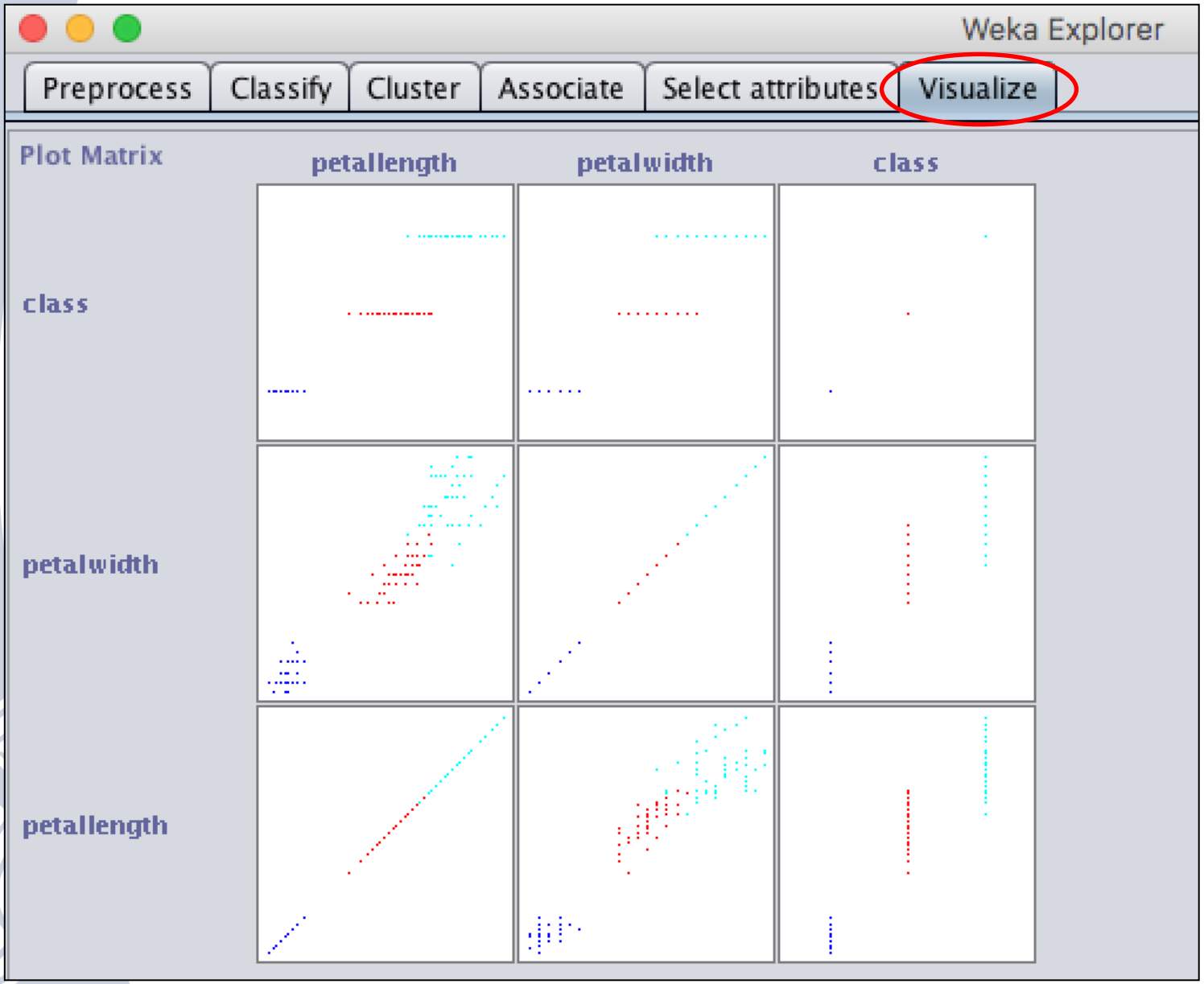
Weka – Regras de Associação



Weka – Seleção de Atributos

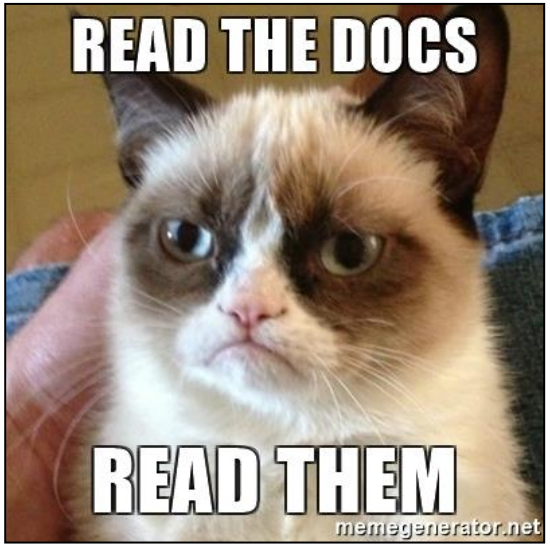


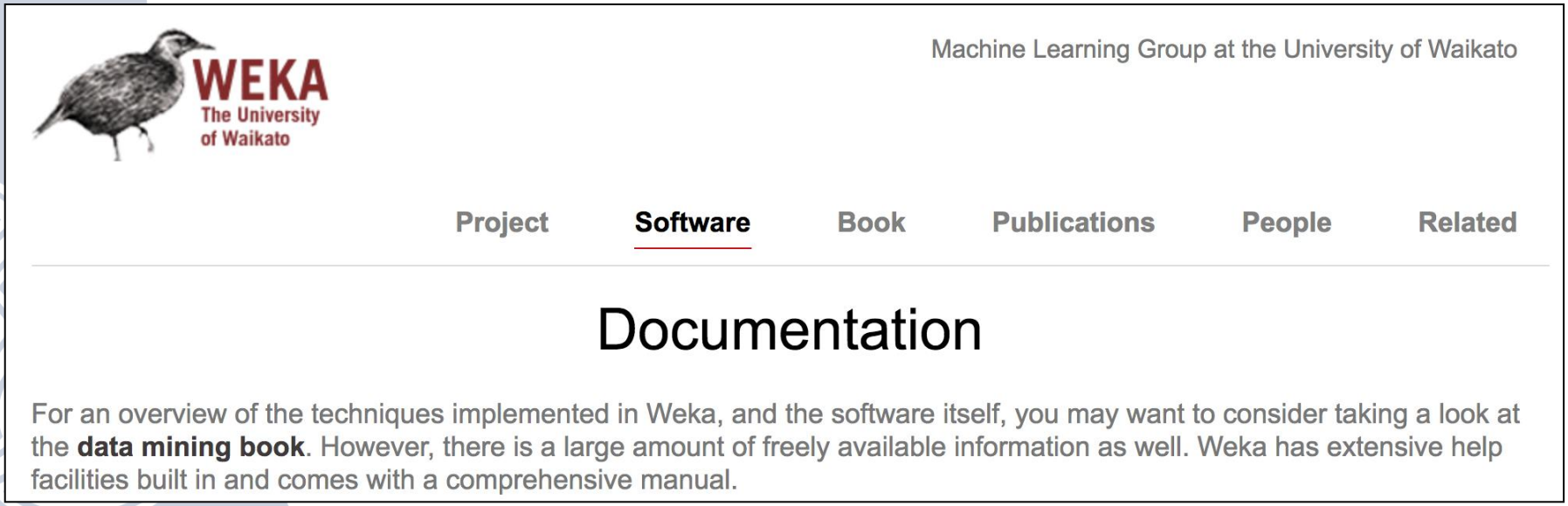
Weka – Visualização dos Dados



Weka – Documentação

- https://www.cs.waikato.ac.nz/ml/weka/documentation.html





Linguagens mais utilizadas

- R (http://www.r-project.org/)
  - dplyr, plyr e data.table (manipulação de dados)
  - stringr (manipulação de strings)
  - zoo (time-series)
  - ggvis, lattice e ggplot2 (gráficos)
  - caret (Machine Learning)Site para download e manuais:

- Python (https://www.python.org)
  - SciPy / NumPy (computação científica)
  - Pandas (manipulação de dados)
  - Matplotlib (gráficos)
  - Scikit-learn (Machine Learning)

Python ou R?

- R ou Python para análise de dados?
  - http://www.cienciaedados.com/r-ou-python-para-analise-de-dados/
- Discussão dos prós e contras de cada um:
  - https://www.kdnuggets.com/2015/05/r-vs-python-data-science.html
- Infográfico interessante:
  - https://www.datacamp.com/community/tutorials/r-or-python-for-data-analysis

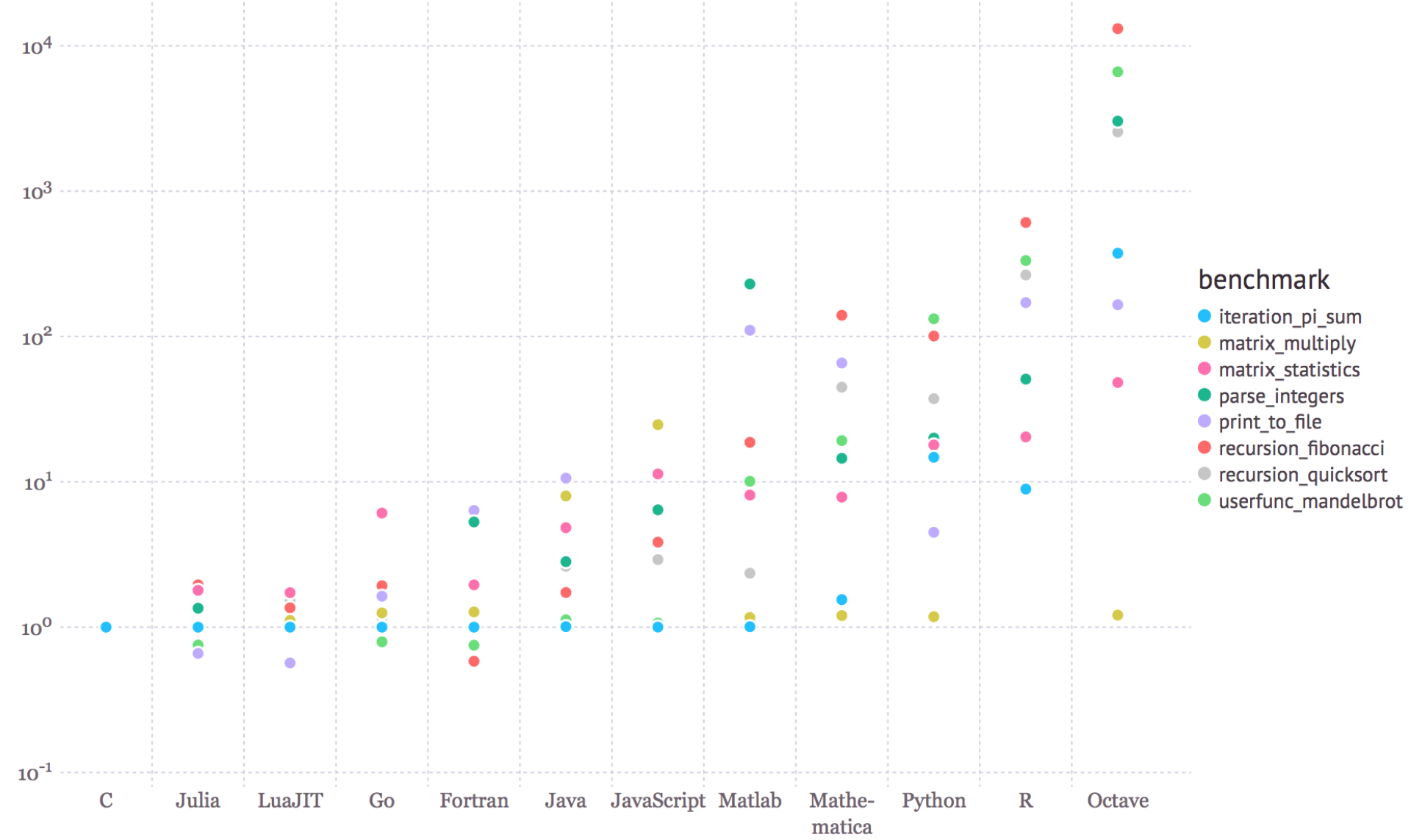
Exemplo – Mineração de Tweets

Julia?

- Julia é uma linguagem *open-source* de alto nível e alta performance para computação numérica.
  - Compilada *just-in-time* (JIT);
  - Sintaxe simples (parecida com Python);
  - Tipagem dinâmica (também permite especificar o tipo);
  - Bibliotecas do Python, C e Fortran podem ser utilizadas;
  - Permite metaprogramação;
  - Gerenciamento automático de memória.

- Link : http://julialang.org/

Julia - benchmark



Apriori

- Implementação do algoritmo Apriori em linguagem C desenvolvida por Cristian Borgelt
- Encontra
  - regras de associação
  - itemsets frequentes
  - itemsets máximos
  - itemsets fechados

Apriori

- Página para download e instruções para execução:

http://www.borgelt.net/apriori.html

- Também é disponibilizado no site uma interface gráfica para visualização de regras desenvolvida em linguagem Java, o ARView
- Outras ferramentas também são disponibilizadas

Apriori – Formato de Entrada

transacoes

```
cafe pao manteiga
leite cerveja pao manteiga
cafe pao manteiga
leite cafe pao manteiga
cerveja
menteiga
pao
feijao
arroz feijao
arroz
```

Apriori - Execução

- Exemplo de linha de comando

./apriori -tr -o -c70.0 -s30.0 transacoes saida

- Exemplo do arquivo de saída

```
manteiga <- cafe (30, 100)
cafe <- manteiga (40, 75)
pao <- cafe (30, 100)
pao <- manteiga (40, 100)
manteiga <- pao (50, 80)
```

Apriori – Parâmetros de Execução

Parâmetro	Descrição	Padrão
-t#	tipo de alvo (s: itemsets frequentes, c: fechados, m: máximos, g: geradores, r: regras de associação)	s
-m#	número mínimo de itens por conjunto/regra	1
-n#	número máximo de itens por conjunto/regra	sem limite
-s#	suporte mínimo para um conjunto/regra	10%
-S#	suporte máximo para um conjunto/regra (positivo: porcentagem, negativo: número absoluto)	100%
-o	uso da definição de suporte da regra original	cabeça e corpo
-c#	confiança mínima de uma regra	80%
-e#	medida de avaliação adicional	nenhum
-a#	modo de agregação para medida de avaliação	nenhum
-d#	limiar para medida de avaliação adicionada	10%
-i#	melhora mínima de medida de avaliação	sem limite
-z	ignora avaliação abaixo do suporte esperado	avaliar todos
-p#	(tamanho mínimo para) poda com avaliação (< 0: para frente ("fraco"), > 0 para frente ("forte"), $= 0$: poda para traz)	sem poda
-q#	ordenar itens considerando a frequência (1: ascendente, -1: descendente, 0: não ordenar, 2: ascendente, -2: descendente considerando a soma do tamanho das transações)	2

Apriori – Parâmetros de Execução

Parâmetro	Descrição	Padrão
-u#	filtrar itens não usados das transações (0: não filtra itens, <0: fração de itens removidos para filtragem, >0: considera taxas de vezes de execução)	0.01
-x	não podar com extensões perfeitas	podar
-y	poda a-posteriori de conjuntos de itens	
-T	não organizar transações com árvores de prefixo	
-Z	imprimir estatísticas dos conjuntos de itens	
-k#	separador de itens para saída	" "
-I#	sinal de implicação das regras de associação	< –
-v#	formato da informação de saída dos conjuntos/regras	(%S)
-l#	ordenar itens na saída pelo tamanho (< 0: descendente, > 0: ascendente)	sem ordenar
-r#	separador de registros/transações	\n
-f#	separadores de itens/campos	\t
-C#	caracteres de comentários	\#
-!	imprimir informação de opção adicional	
infile	arquivo de transações	[obrigatório]
outfile	arquivo de saída	[opcional]
appfile	arquivo definindo itens selecionados	[opcional]

Torch

- Conjunto de ferramentas para
  - Pré-processamento
  - Agrupamento de Textos (vários algoritmos)
  - Classificação Hierárquica e Visualização de Hierarquias de Tópicos
- Desenvolvida por Ricardo Marcacini – LABIC-ICMC-USP

- Página principal da Torch:

http://sites.labic.icmc.usp.br/torch/

Torch - JPreText

- Ferramenta para pré-processamento de textos
- Principais funcionalidades:
  - Remoção de stopwords
  - Stemming
  - Criação da bag-of-words
- Download e instruções de uso:

http://sites.labic.icmc.usp.br/torch/msd2011/jpretext/

Torch - JPreText

- Entrada: diretório com um documento em cada arquivo (.txt)
  - "nome_classe.ID_ARQUIVO.txt"
- Caso a classe de cada arquivo seja conhecida, ela pode ser usada para avaliação dos agrupamentos gerados pela Torch
- Execução:

```
java -Xmx2G -cp jpretext.jar pretext.Main
./config.ini
```

Torch - JPreText

- Arquivo de configurações (config.ini):

```
#Text Collection (e.g. Reuters)
Text Source: ./MinhaColecaoDeTextos

# Preprocessing
# Stemming language options are English, Portuguese or None
Stem Language: Portuguese
Max. Keywords: 20
Stopwords File: ./stopwords.txt

# Term Selection using Document Frequency
Min. DF: 2

# Term weighting can ben TF or TFIDF
Term Weighting: TFIDF
Normalization: Yes

# Output
CSV Data File: MinhaColecaoDeTextos.csv
```

Torch – TopHClust

- Algoritmo para agrupamento hierárquico (Bisecting K-means)
- Página para download e instruções:

http://sites.labic.icmc.usp.br/torch/msd2011/tophclust/

- Execução:

```
java -cp tophclust.jar cluster.XSectingKmeans ./config.ini
```

Torch - ClusterMap

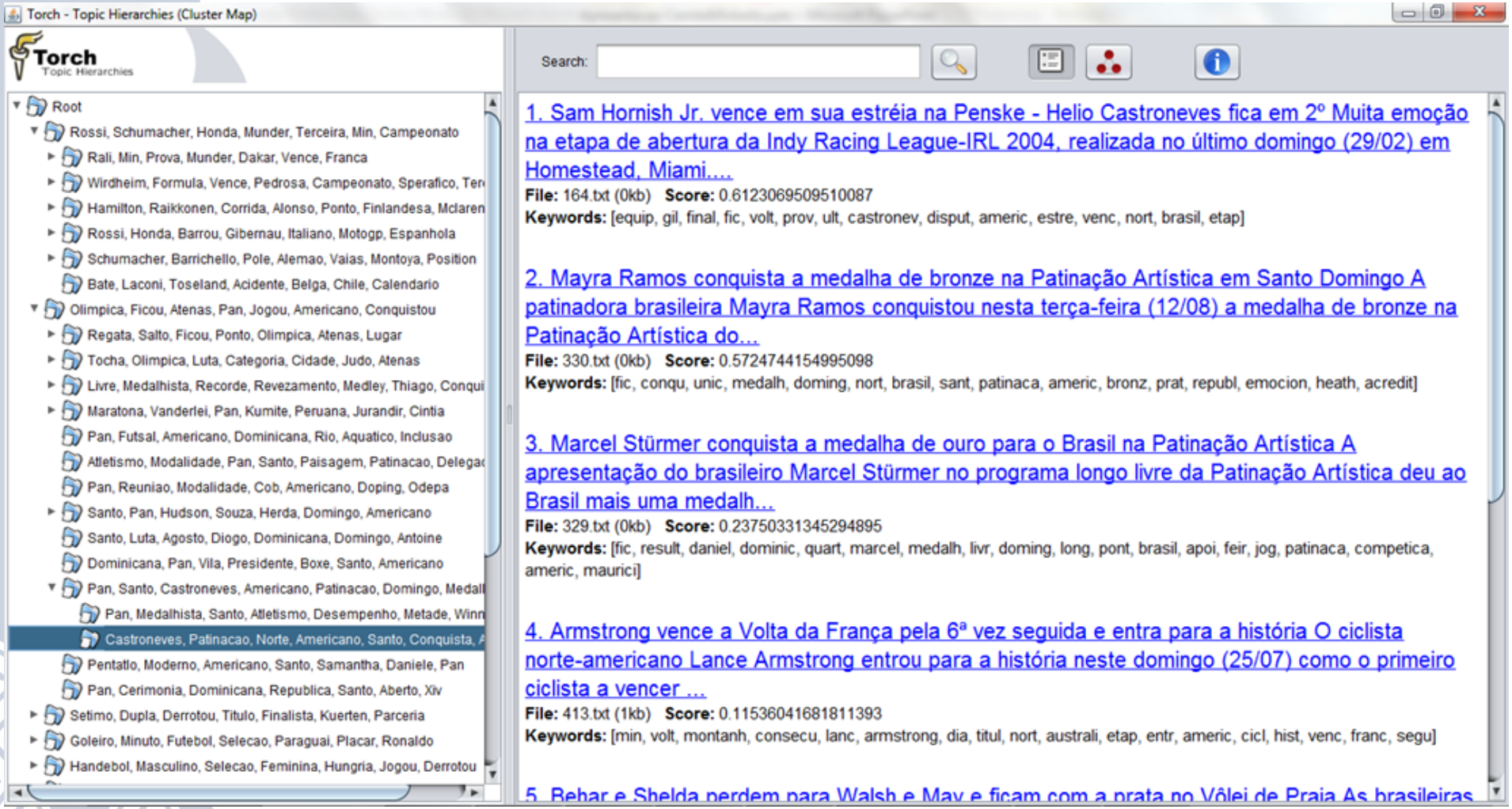
- Classificação Hierárquica e Visualização de coleções em uma hierarquia de tópicos
- Download e instruções:

http://sites.labic.icmc.usp.br/torch/msd2011/clustermap/

- Execução:

java -Xmx1G -cp clustermap.jar torch.clustermap.Explorer ./config.ini

Torch - ClusterMap



Outras Ferramentas

- Orange
  - Uma ferramenta para visualização e análise de dados tanto para usuários iniciantes quanto especialistas
  - Componentes para aprendizado de máquina e consequentemente mineração de dados
  - Permite selecionar dados de treino e teste visualmente
  - Requer Python instalado
  - Download http://orange.biolab.si/download/

Outras Ferramentas

- Tanagra
  - Ferramenta para mineração de dados de proposito acadêmico
  - Métodos para analise exploratória de dados, aprendizado estatístico e aprendizado de máquina
  - Projeto open-source
  - Download: http://eric.univ-lyon2.fr/~ricco/tanagra/en/tanagra.html

Outras Ferramentas

- C4.5
  - Implementação do algoritmo C4.5 de Ross Quinlan
  - Inclui procedimento para amostragem na etapa de avaliação do algoritmo
  - Download: http://www.rulequest.com/Personal/

Outras Ferramentas

- MCL++
  - Biblioteca de classes em C++ para aprendizado de máquina
  - supervisionado
  - Domínio público
  - Download: http://www.sgi.com/tech/mlc/

Outras Ferramentas

- MineSet
  - Provê 5 ferramentas para exploração visual de dados e para resultados de mineração de dados
  - Facilita o entendimento de grandes quantidades de dados
  - Download: http://www.dcc.uchile.cl/~rbaeza/cursos/visual/sg/index.html

Outras Ferramentas

- Clementine
  - Suíte comercial popular para mineração de dados
  - Embarcado em vários sistemas de mineracão de dados
  - Download: http://www.spss.com/clementine

Outras Ferramentas

- Knime
  - Desenvolvido em Java e assim como o Weka, sua biblioteca pode ser facilmente incorporada em outros códigos
  - Permite denir um uxo de dados
  - Download: http://www.knime.org

Outras Ferramentas

- Keel
  - Ferramenta open-source desenvolvida em Java
  - Desenvolvido para fins de pesquisa e educação
  - Implementa algoritmos evolutivos, fuzzy e demais algoritmos para regressão, classificação, agrupamento, etc.
  - Análise de resultados utilizando testes de significância estatística
  - Inclui técnicas para pré-processamento dos dados
  - Download: http://www.keel.es/

Outras Ferramentas

- Cluto
  - Software de agrupamento para conjuntos dados de alta dimensionalidade
  - Análise de características dos grupos
  - Disponibiliza uma interface gráfica e uma interface web
  - Download: http://glaros.dtc.umn.edu/gkhome/views/cluto

Agradecimentos

- Material desenvolvido com ajuda de
  - Rafael Geraldeli Rossi
  - Roberta Akemi Sinoara
  - Camila Vaccari Sundermann
  - Felipe Coutinho