

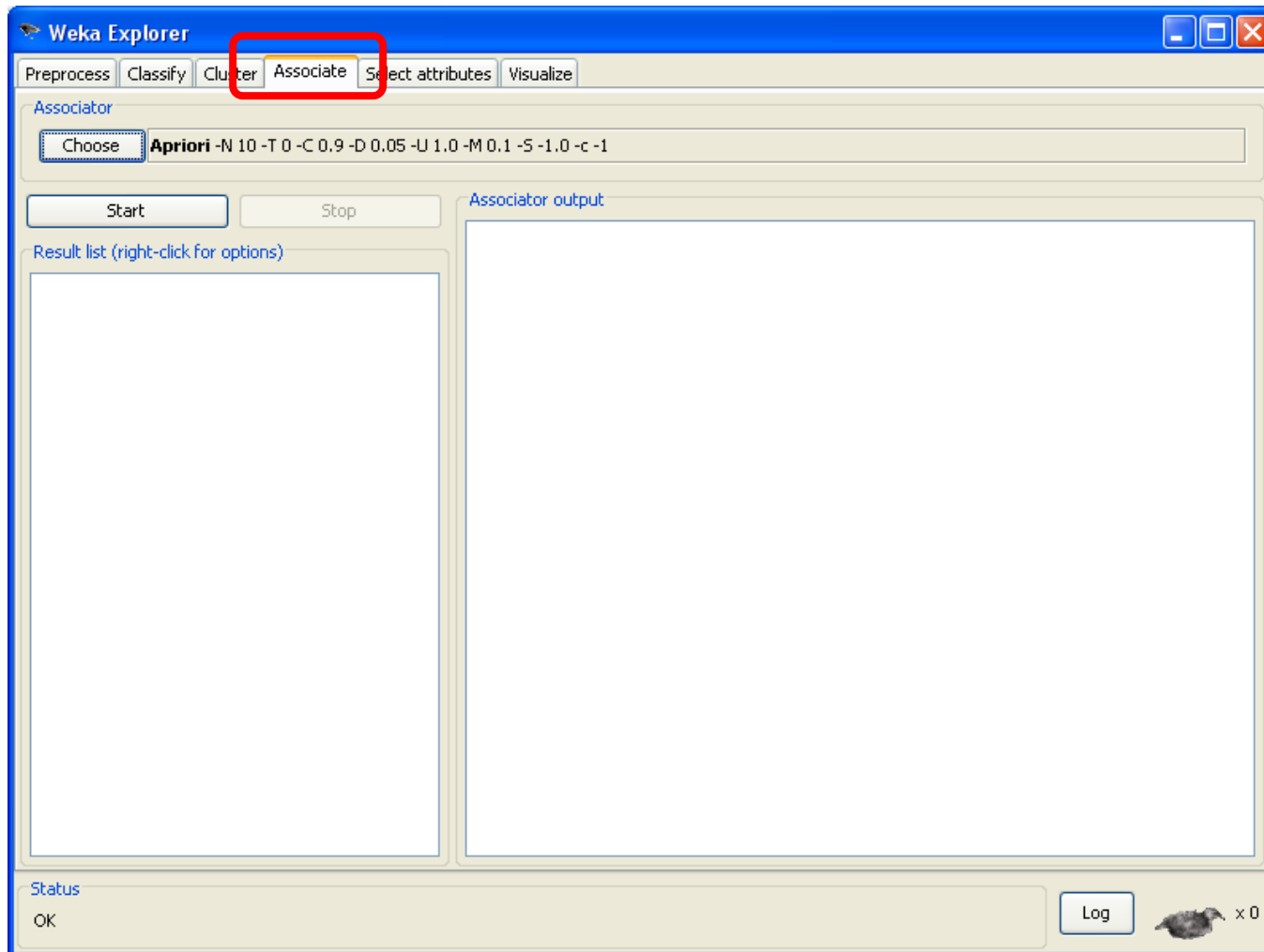
Association rules with WEKA

Lab 7

Lab 7. Association rules with WEKA

1. Market basket
2. Correlations in Human Development Index table
3. Ass 3: Correlations in Communities dataset

Association rules in WEKA Explorer



Parameters for Apriori

weka.gui.GenericObjectEditor

weka.associations.Apriori

About

Class implementing an Apriori-type algorithm.

More

Capabilities

car False

classIndex -1

delta 0.05

lowerBoundMinSupport 0.1

metricType Confidence

minMetric 0.9

numRules 10

outputItemSets False

removeAllMissingCols False

significanceLevel -1.0

upperBoundMinSupport 1.0

verbose False

Open... Save... OK Cancel

car -- If enabled class association rules are mined instead of (general) association rules: only rules which contain class attribute

Apriori in WEKA is iterative

- Starts looking for frequent itemsets with upper bound min support.
- If found the predefined number of rules, then stops.
- If did not find, then repeats with support decreased by delta
- Until it reaches the predefined lower bound for min support, or finds the predefined number of rules

Parameters for Apriori

weka.gui.GenericObjectEditor

weka.associations.Apriori

About

Class implementing an Apriori-type algorithm.

More

Capabilities

car False

classIndex -1

delta 0.05

lowerBoundMinSupport 0.1

metricType Confidence

minMetric 0.9

numRules 10

outputItemSets False

removeAllMissingCols False

significanceLevel -1.0

upperBoundMinSupport 1.0

verbose False

Open... Save... OK Cancel

lowerBoundMinSupport -
- Lower bound for
minimum support.

upperBoundMinSupport --
Upper bound for minimum
support. Start iteratively
decreasing minimum
support from this value.

How rules are ranked

- The rules are ranked by the selected parameter
- We are going to use confidence
- Lift and conviction are out of date
- So, the program produces the predefined number of top rules ranked according to the confidence

Parameters for Apriori

weka.gui.GenericObjectEditor

weka.associations.Apriori

About

Class implementing an Apriori-type algorithm.

More

Capabilities

car False

classIndex -1

delta 0.05

lowerBoundMinSupport 0.1

metricType Confidence

minMetric 0.9

numRules 10

outputItemSets False

removeAllMissingCols False

significanceLevel -1.0

upperBoundMinSupport 1.0

verbose False

Open... Save... OK Cancel

metricType -- Set the type of metric by which to rank rules. Confidence is the proportion of the examples covered by the premise that are also covered by the consequence

minMetric -- Minimum metric score. Consider only rules with scores higher than this value.

Parameters for Apriori

weka.gui.GenericObjectEditor

weka.associations.Apriori

About

Class implementing an Apriori-type algorithm. More
Capabilities

car

classIndex

delta

lowerBoundMinSupport

metricType

minMetric

numRules

outputItemSets

removeAllMissingCols

significanceLevel

upperBoundMinSupport

verbose

Open... Save... OK Cancel

outputItemSets -- If enabled the itemsets are output as well.

Requirements

- Attributes have to be NOMINAL
- APRIORI CAN BE RUN ONLY ON VERY SMALL DATASETS: memory problem
- For medium-large datasets use FP-growth

Small dataset: HDI index of countries

HDI-composite index measuring average achievement in three basic dimensions of human development—a long and healthy life, knowledge and a decent standard of living.

- Parameters are all numeric:
 - Life expectancy at birth
 - Mean years of schooling
 - Expected years of schooling
 - Gross national income per capita (\$)
 - Human development index (HDI)
 - HDI without income

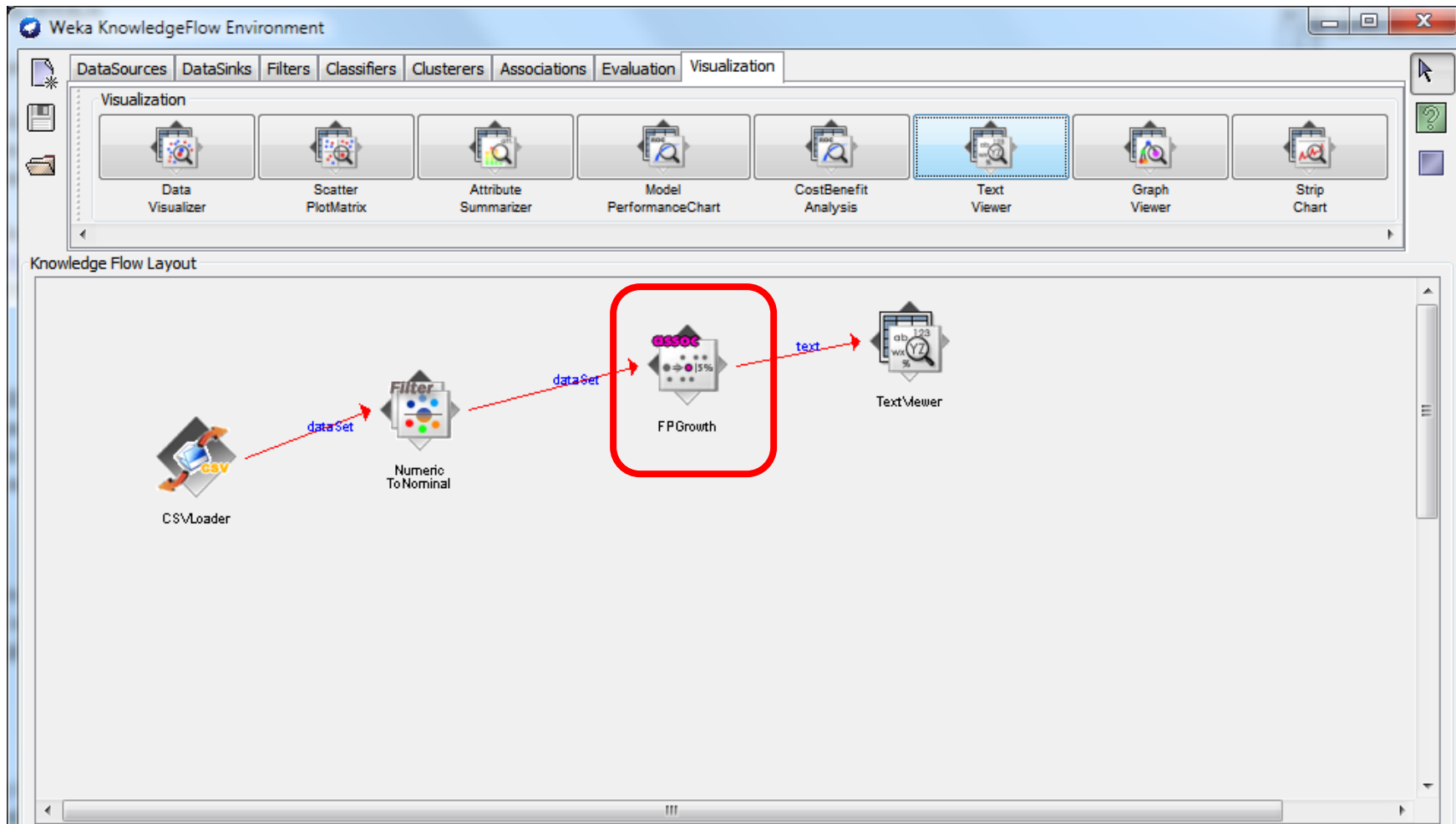
Small dataset: HDI index of countries

HDI-composite index measuring average achievement in three basic dimensions of human development—a long and healthy life, knowledge and a decent standard of living.

- Discretize parameters into bins, give meaningful names to the bins
- HDI and HDI no income are derivatives, so better remove them

How to scale up: larger datasets

Use WEKA knowledge flow interface



Large dataset

- Store transactions

Part of assignment 3

- Communities dataset:
- Combine skills obtained from parts 1 and 2, and output top 15 association rules for attributes of communities
- Perform the same task, but now each rule should contain a class attribute: percentage of violent crimes