Data Science 9

Tilburg University

Understanding Society

Overview

- Pre-processing
- Normalization
- Outlier removal
- Feature Selection
- Dimensionality Reduction
- Error measures

Pre-processing

What are the features?

-> what is available? what domain knowledge do we have (access to)?

What are their distributions (min, max, mean, histogram, outliers)?

-> does the distribution looks approximately normal, uniform, …?

Which features are relevant and why?

-> effect of inclusion/exclusion on prediction performance

-> domain knowledge

Which features are highly correlated? (“collinearity”)

-> correlation matrix

Are there missing feature values? How can they be dealt with?

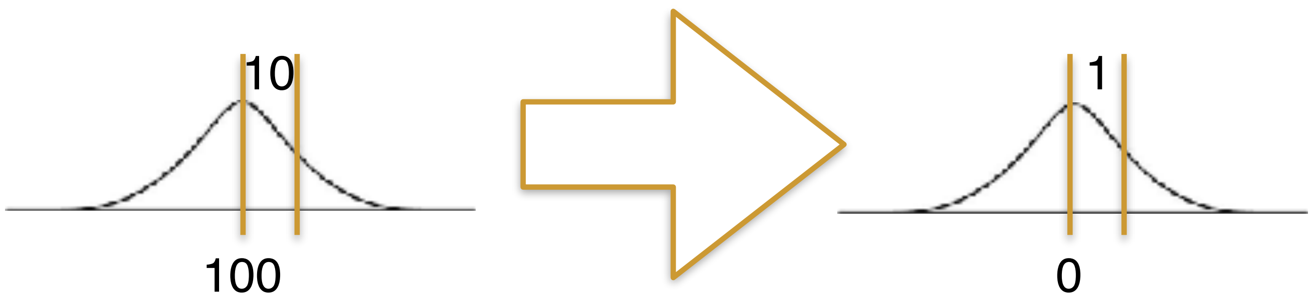
-> imputation, prediction

Which information is missing in the features and can it be represented by constructed features?

-> domain knowledge!

Data normalization

- Centering by mean = subtract mean from all values
- Scale by std = divide all values by the standard deviation

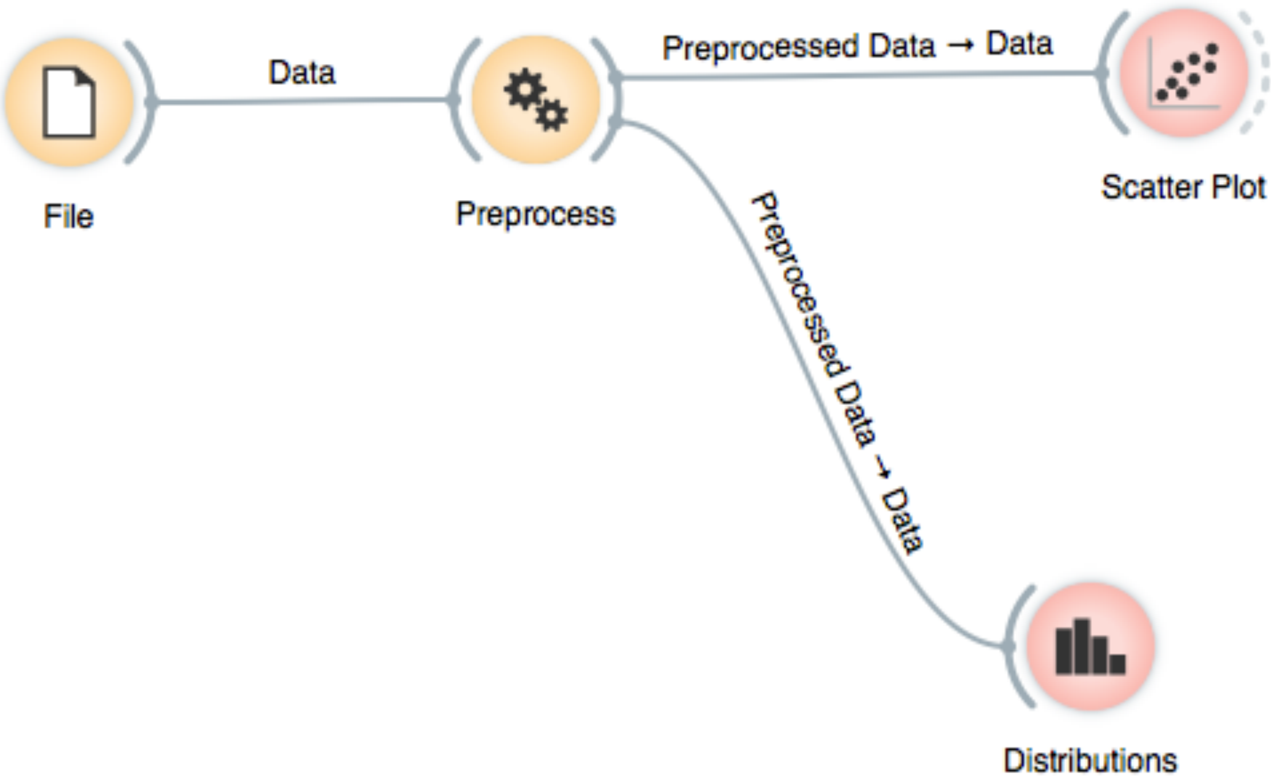


- Centering by median = subtract median from all values
- Scale by span = divide all values by (max. value - min. value)

(less sensitive to outliers)

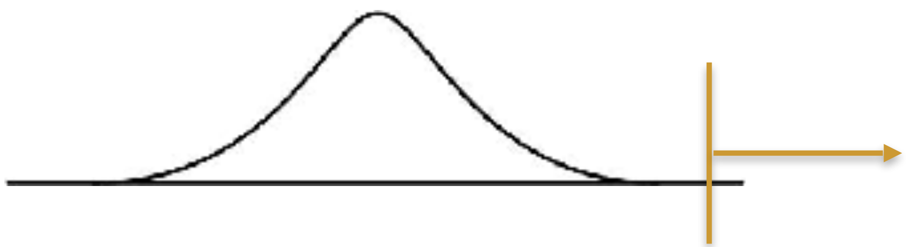
Normalization

- ionosphere.tab



Outlier Removal/Detection

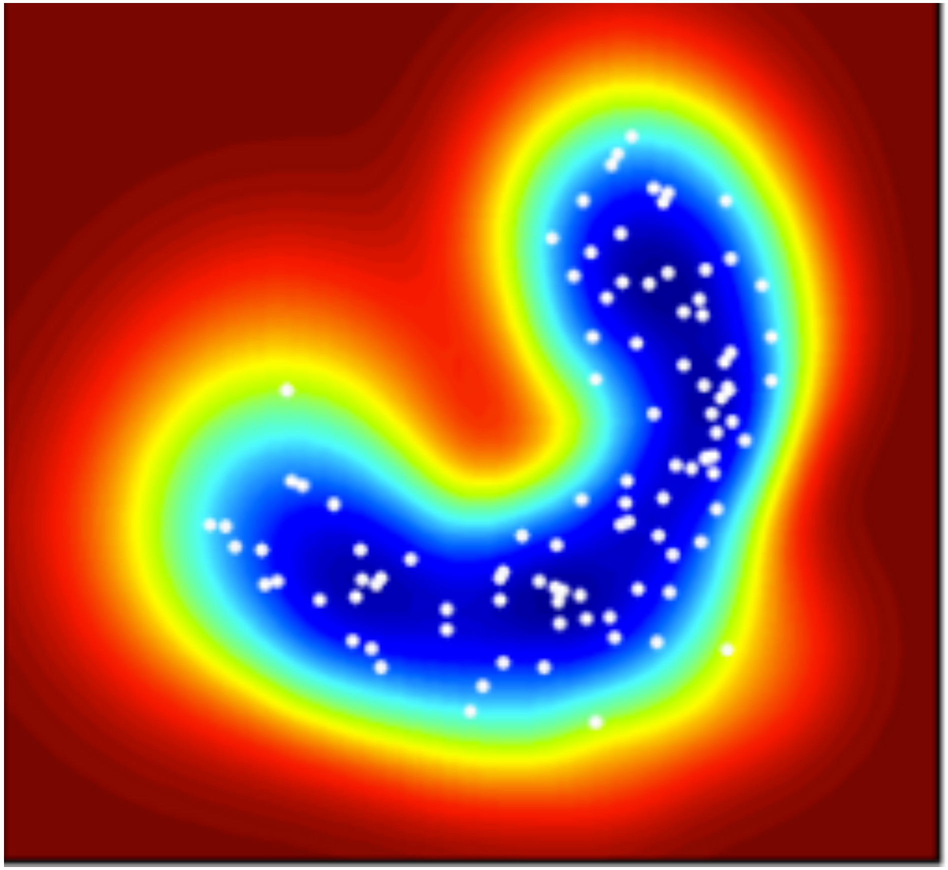
- What is an outlier?
- In statistics defined in terms of standard deviations…



- In Orange: two methods

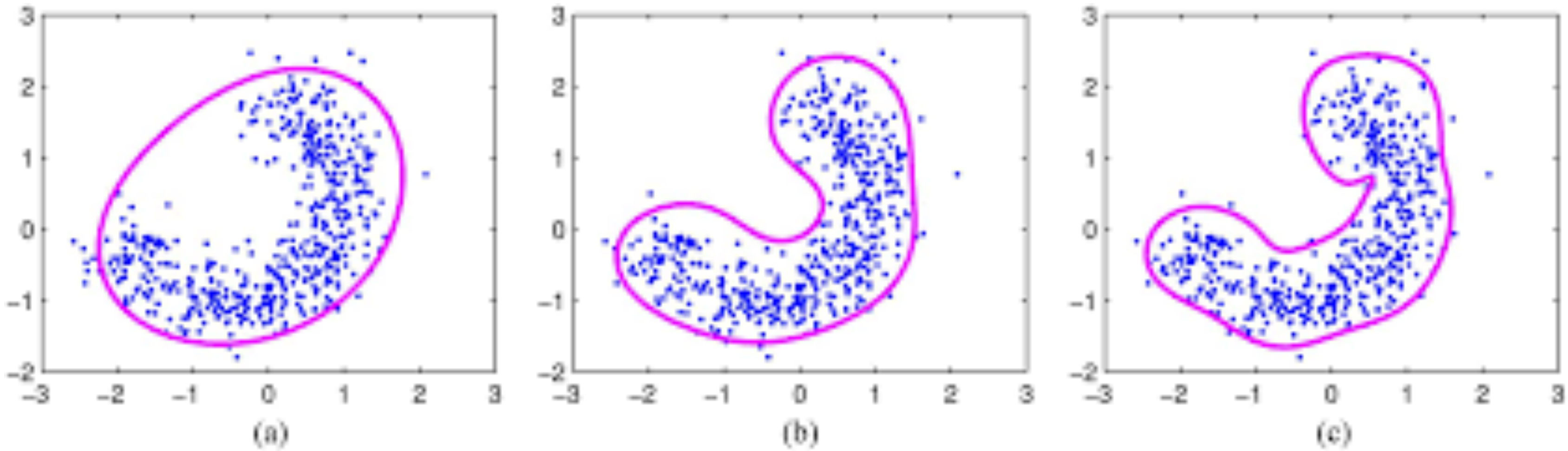
- One-class classification (SVM with RBF kernel) -> for non-Gaussian data
- Covariance estimator (for Gaussian data)

One-class classification



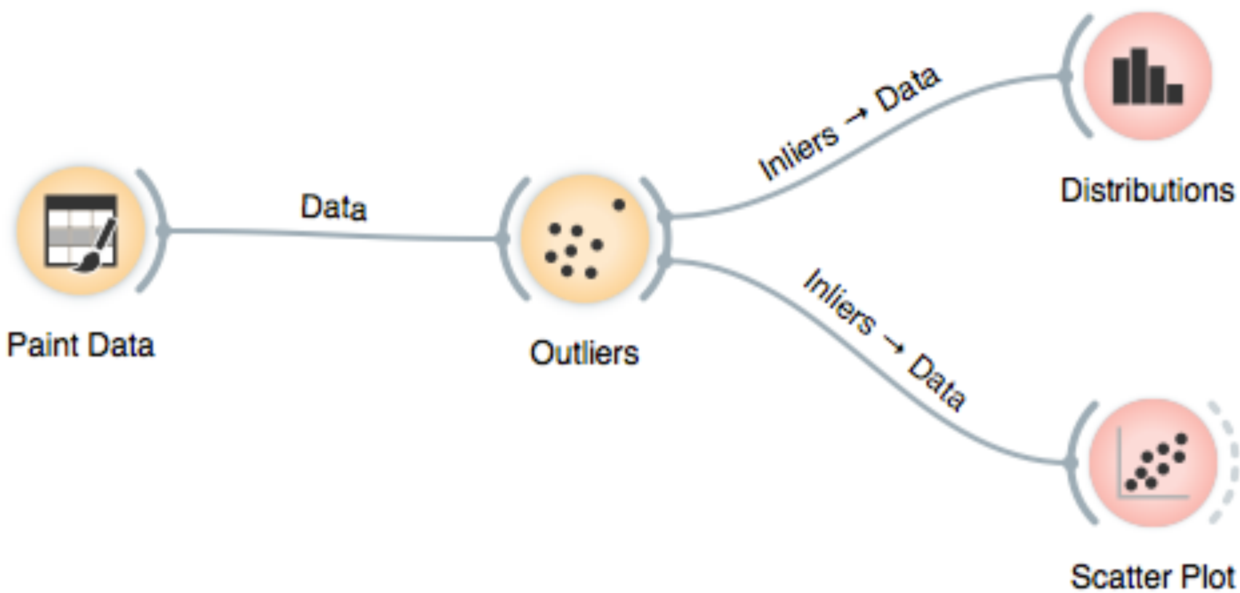
SVM for one-class classification

- The banana-shaped decision boundary is formed by the SVM

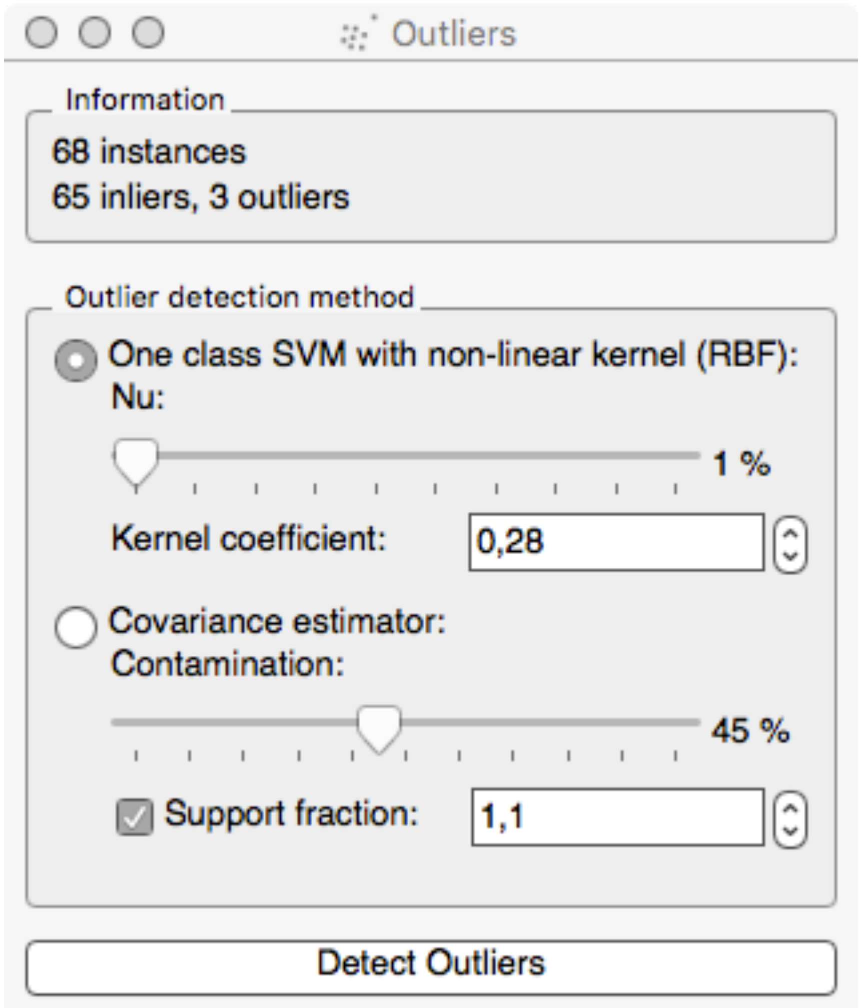


Tilburg University

Outlier removal/detection with SVM



SVM parameters



percentage of support vectors
= minimum complexity of boundary

larger value = smoother boundary

Tilburg University

Feature Selection

- Three methods for feature selection in Orange

- Information Gain (entropy/information)
- Gain Ratio (entropy/information)
- Gini Index (impurity measure)

Information Theory

The odd-one-out of a dozen

You are given 12 balls, all equal in weight except for one that is either heavier or lighter. You are also given a two-pan balance to use. In each use of the balance you may put any number of the 12 balls on the left pan, and the same number on the right pan, and push a button to initiate the weighing; there are three possible outcomes: either the weights are equal, or the balls on the left are heavier, or the balls on the left are lighter. Your task is to design a strategy to determine which is the odd ball and whether it is heavier or lighter than the others in as few uses of the balance as possible.

(a) How can one measure information?

(b) When you have identified the odd ball and whether it is heavy or light, how much information have you gained?

(c) Once you have designed a strategy, draw a tree showing, for each of the possible outcomes of a weighing, what weighing you perform next. At each node in the tree, how much information have the outcomes so far given you, and how much information remains to be gained?

(d) How much information is gained when you learn

(i) the state of a flipped coin;

(ii) the states of two flipped coins;

(iii) the outcome when a four-sided die is rolled?

(e) How much information is gained on the first step of the weighing problem if 6 balls are weighed against the other 6? How much is gained if 4 are weighed against 4 on the first step, leaving out 4 balls?

The odd-one-out of a dozen

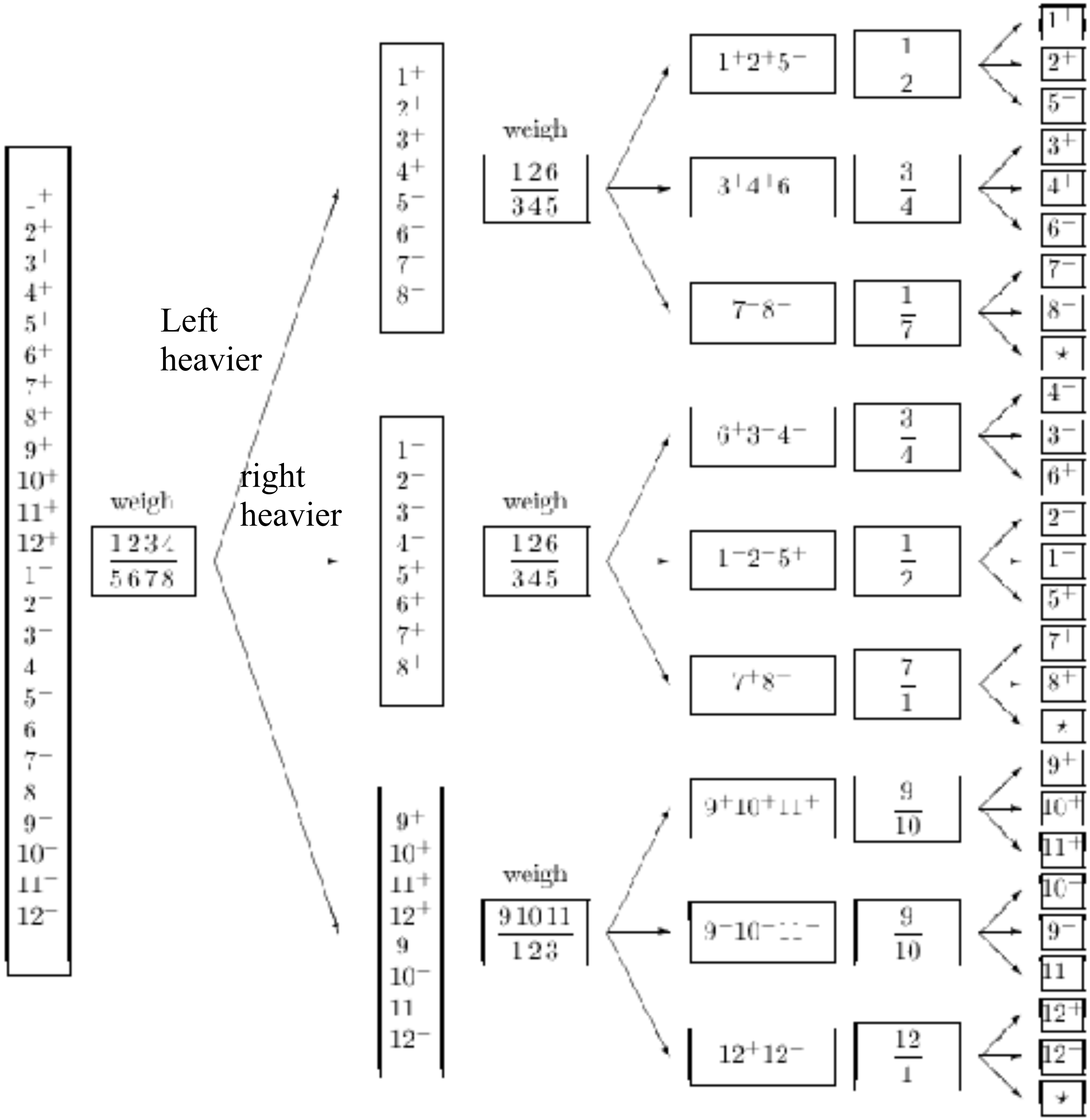
- Number of possible outcomes for K uses of the balance equals 3^K (= 27 for K = 3)
- Number of possible states equals 24
  - The odd ball can be any of 12 and can be lighter or heavier
- Hence three weighings suffice

- HINT: what weighing has the maximal information content?

Optimal strategy

- The three outcomes
  - Left heavier
  - Right heavier
  - Balance
- Should be as close as possible to equiprobable

- E.g., starting with balancing balls 1-6 agains 7-12 is suboptimal because “balance” has probability zero in this case



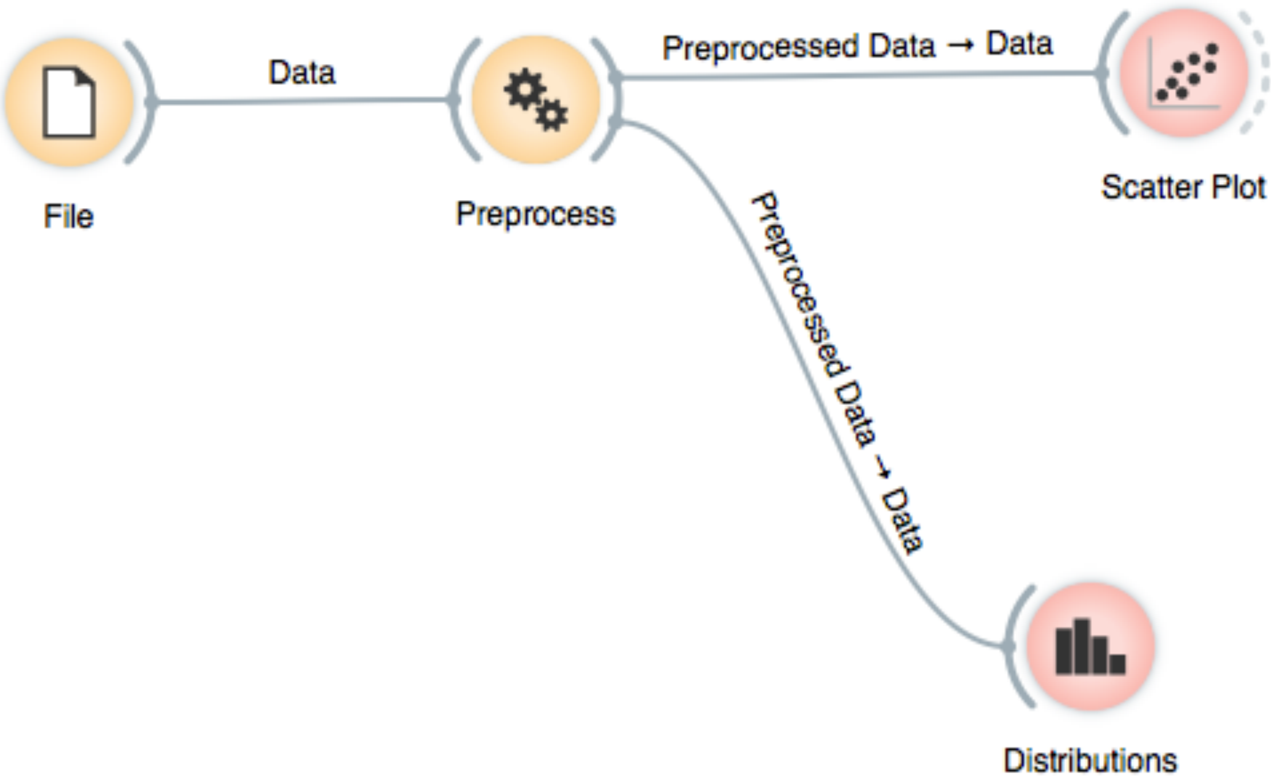
weigh
1234
5678
Left heavier
right heavier
weigh
126
345
weigh
126
345
weigh
91011
123
1+2+5−
3+4+6
7−8−
6+3−4−
1−2−5+
7+8−
9+10+11+
9−10−11−
12+12−

Conclusion

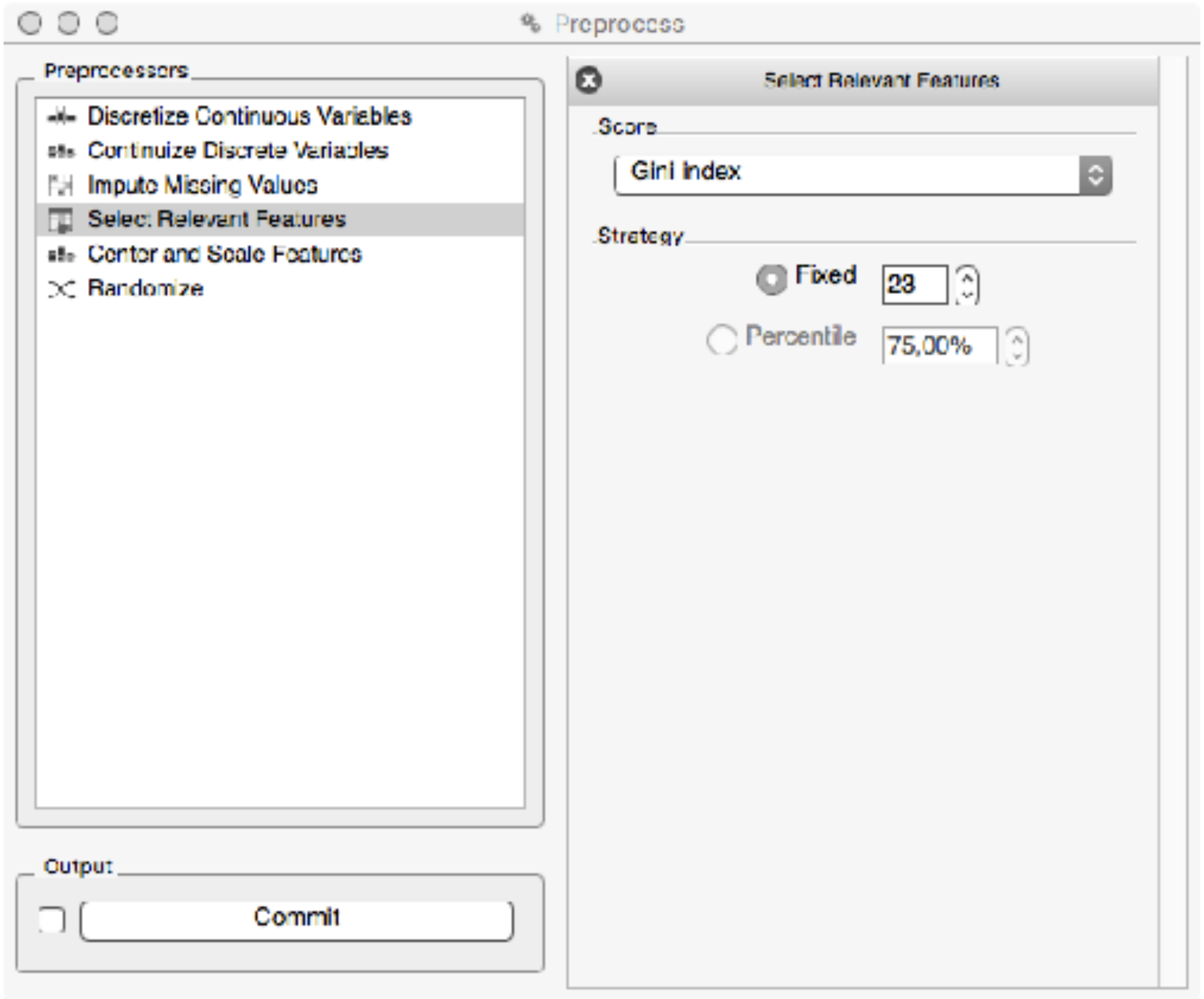
The outcome of a random experiment is to be most informative if the probability distribution over outcomes is uniform

Feature Selection

- ionosphere.tab



Feature Selection



Number of features

Dimensionality Reduction

- Principal Component Analysis
  - continuous features

- Correspondence Analysis
  - discrete features (not for preprocessing)

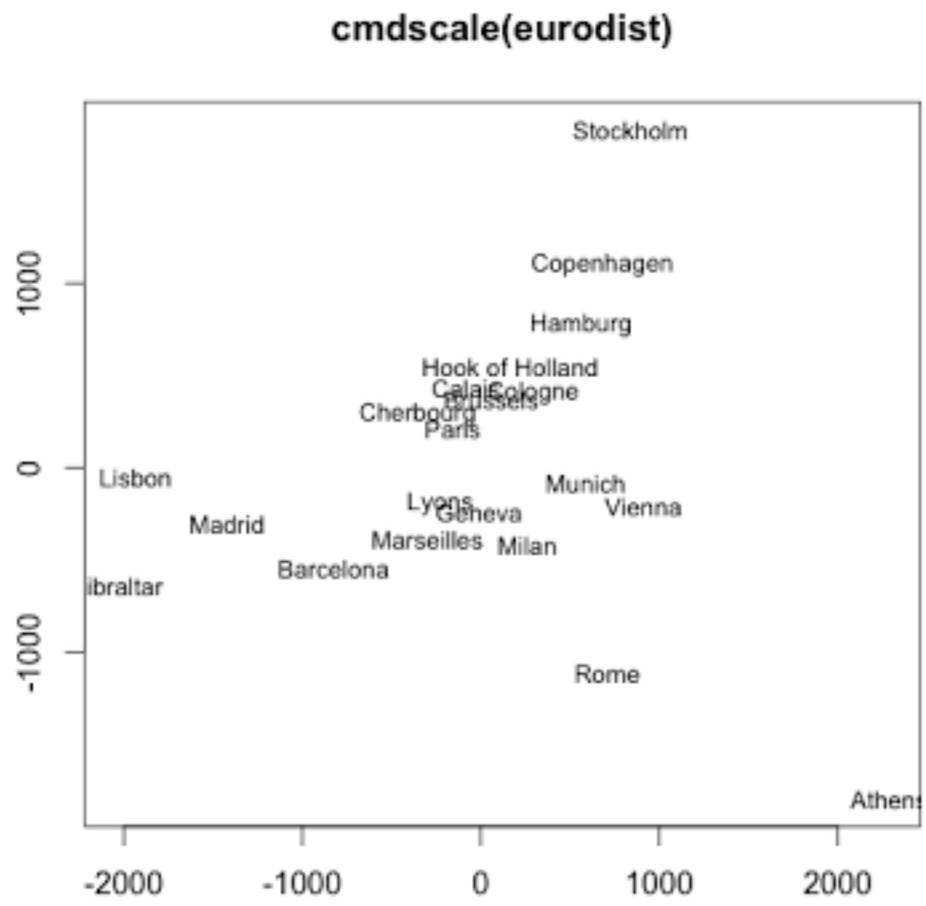
- Multidimensional Scaling (MDS)
  (not for preprocessing)

- k-Means clustering

- hierarchical clustering

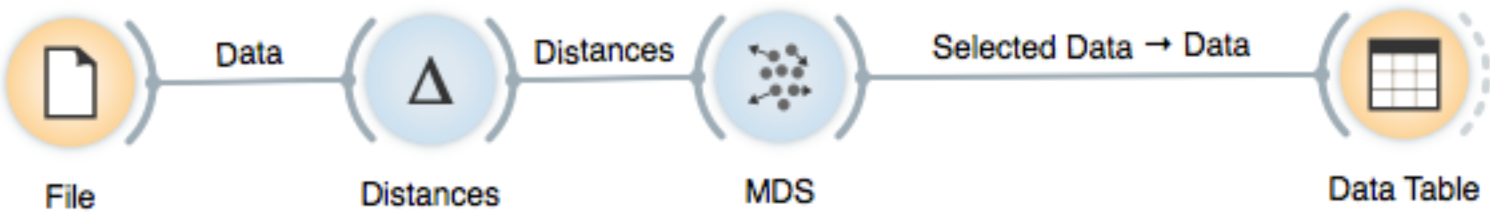
MDS

Input:
Distance table of European cities



TILBURG UNIVERSITY

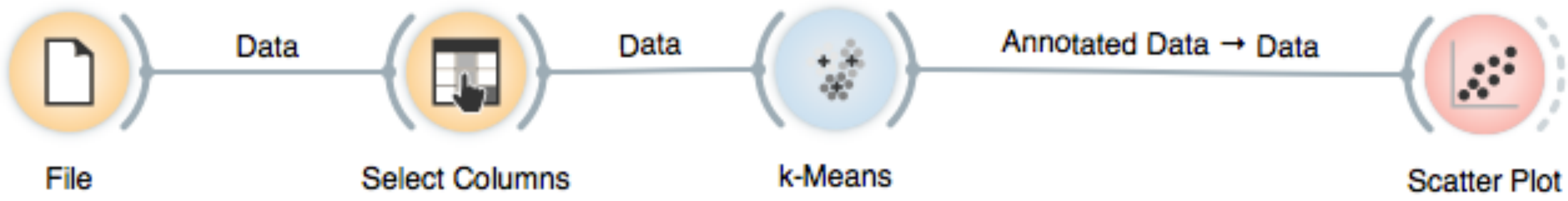
MDS in action



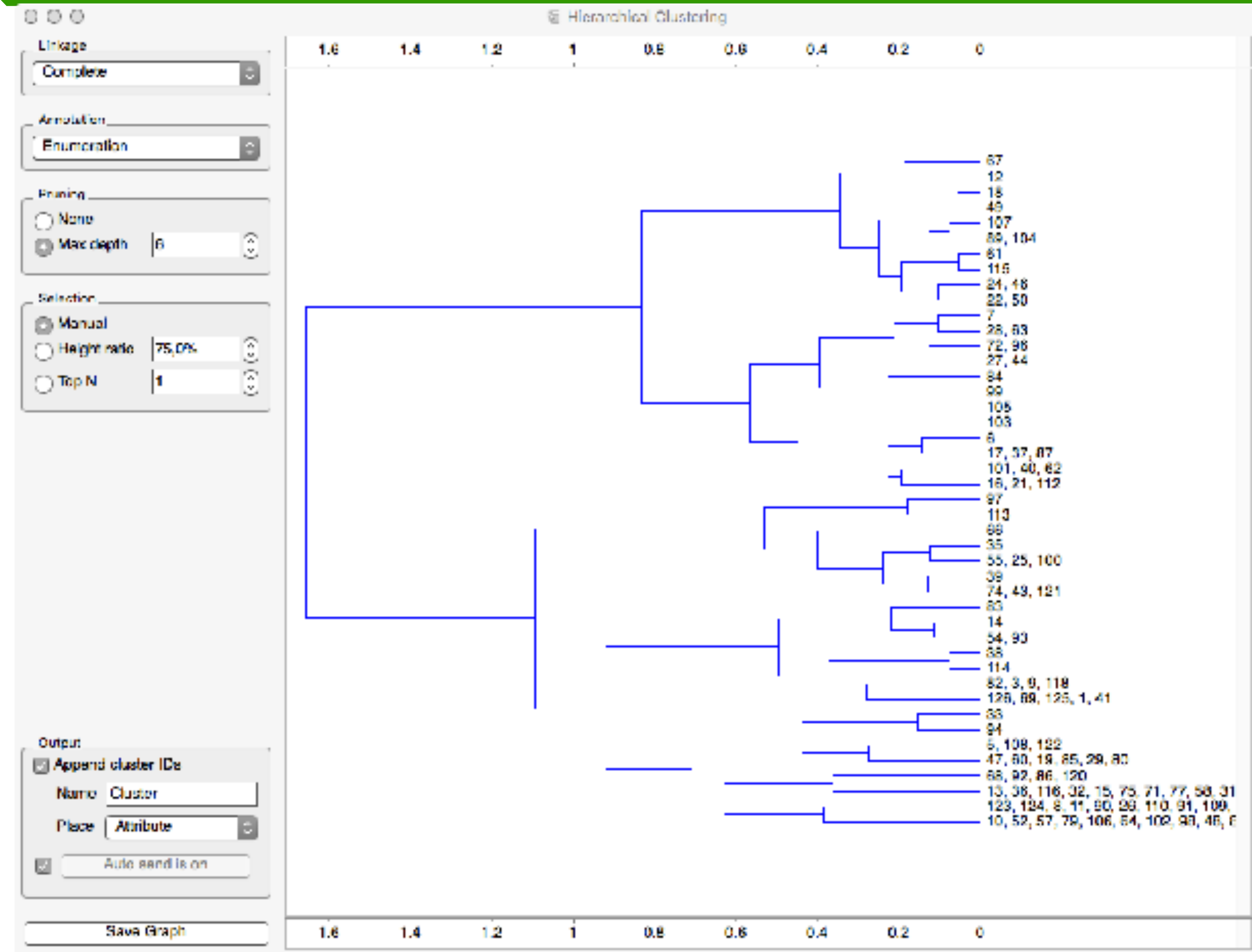
k-means clustering

- Given k clusters, k-means clustering assigns instances to their nearest cluster
- Animation of k-means clustering in action: https://youtu.be/BVFG7fd1H30
- Best clustering is obtained for a specific value of k
- This value can be determined automatically

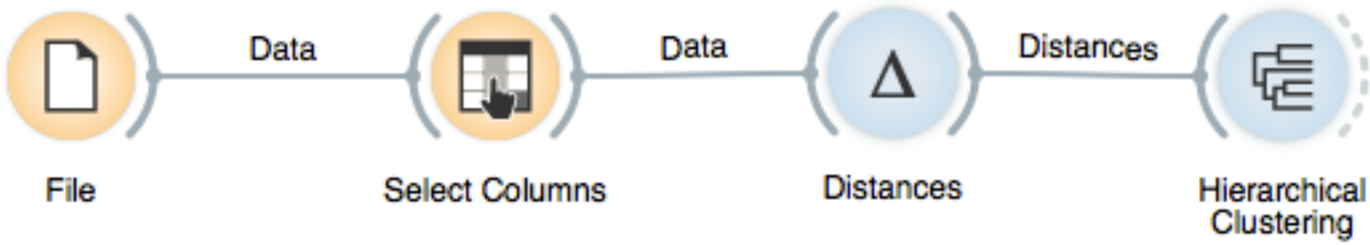
k-means clustering in Orange



Hierarchical clustering



Hierarchical clustering in Orange



Error Measures for Regression

MSE = Mean Squared Error

RMSE = Square Root of Mean Squared Error

MAE = Mean Absolute Error

R2 = R-squared (coefficient of determination) = proportion of the variance in the target that is predictable from the feature(s)

Error Measures for Classification

- CA = Classification Accuracy

- Precision = the fraction of detected instances that are relevant

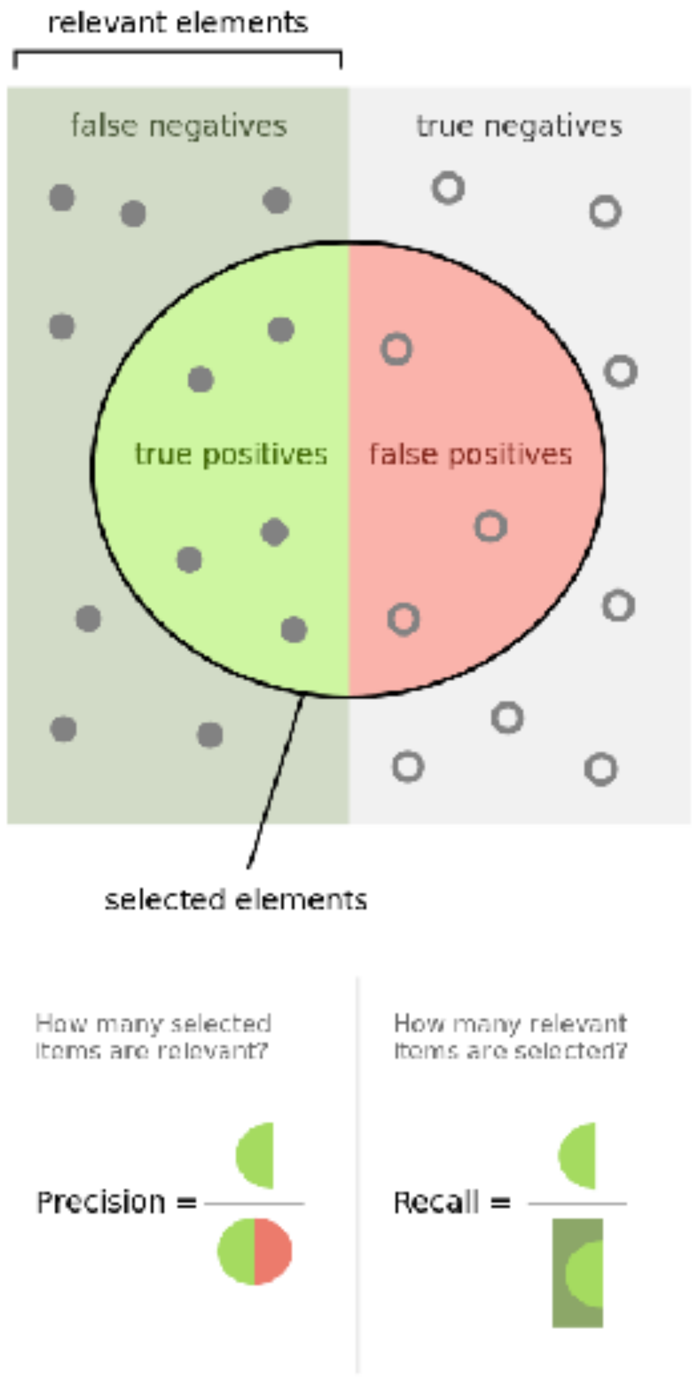
- Recall = the fraction of relevant instances that are detected

$$F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

Confusion Table is preferred

- precision is the fraction of retrieved instances that are relevant

- recall is the fraction of relevant instances that are retrieved



TILBURG UNIVERSITY

$$SS_T = \sum_i (y_i - \bar{y})^2, \quad SS_E = \sum_i (y_i - \hat{y}_i)^2$$

$$SA_T = \sum_i (y_i - \bar{y})^2, \quad SA_R = \sum_i |\hat{y}_i - \bar{y}|$$

mean-squared error (MSE)	SS_E/n
root mean-squared error (RMSE)	$\sqrt{SS_E/n}$
mean absolute error (MSE)	SA_R/n
relative squared error (RSE)	SS_E/SS_T
root relative squared error (RRSE)	$\sqrt{SS_E/SS_T}$
relative absolute error (RAE)	SA_R/SA_T
R-squared (R2)	$1 - SS_E/SS_T$