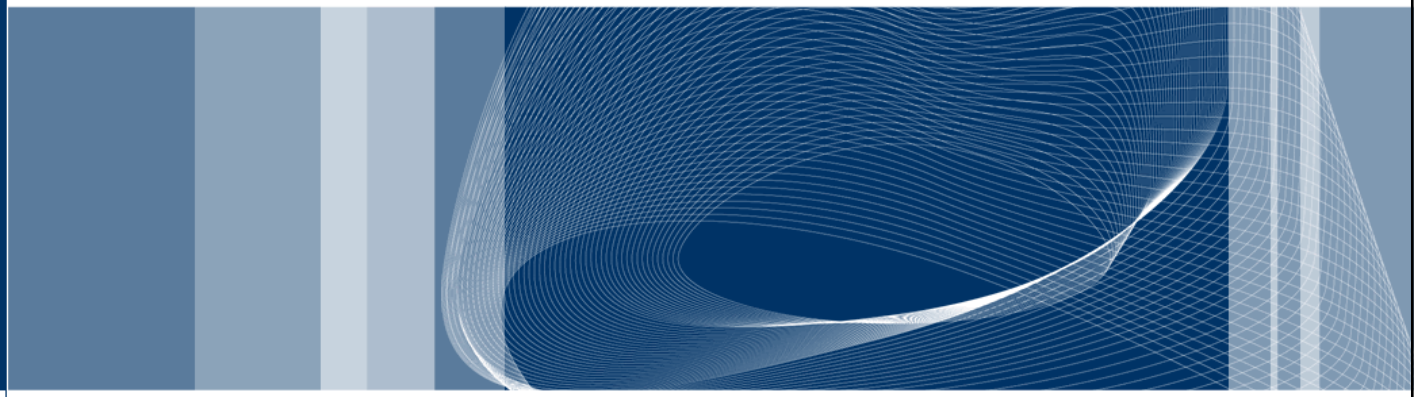
POLITECNICO DI MILANO



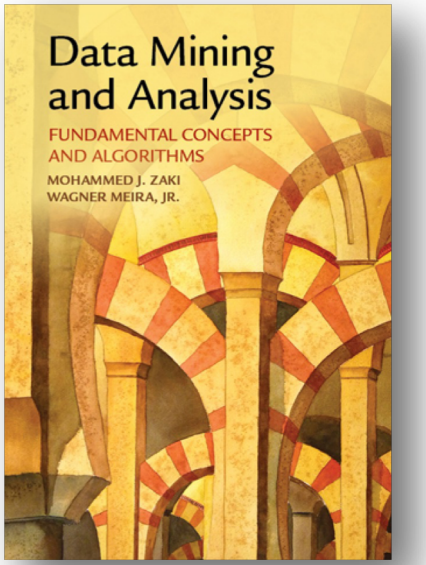
Data Preparation

Data Science for Mobility

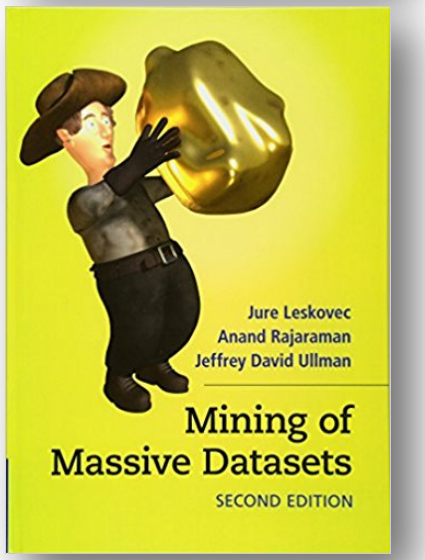
Prof. Pier Luca Lanzi

Readings

- "Data Mining and Analysis" by Zaki & Meira Chapter 7 http://www.dataminingbook.info



- "Mining of Massive Datasets" by Leskovec, Rajaraman, and Ullman; Chapter 11



Why Is Data Preprocessing Important?

- No quality in data, no quality in the mining results!
  (trash in, trash out)
- E.g., duplicate or missing data may cause incorrect or even misleading statistics.
- Data warehouses need consistent integration of quality data
- Data extraction, cleaning, and transformation comprises the majority of the work of building a data warehouse

Data Cleaning

Missing Values

- Reasons for missing values
  - Information is not collected
    (e.g., people decline to give their age and weight)
  - Attributes may not be applicable to all cases
    (e.g., annual income is not applicable to children)

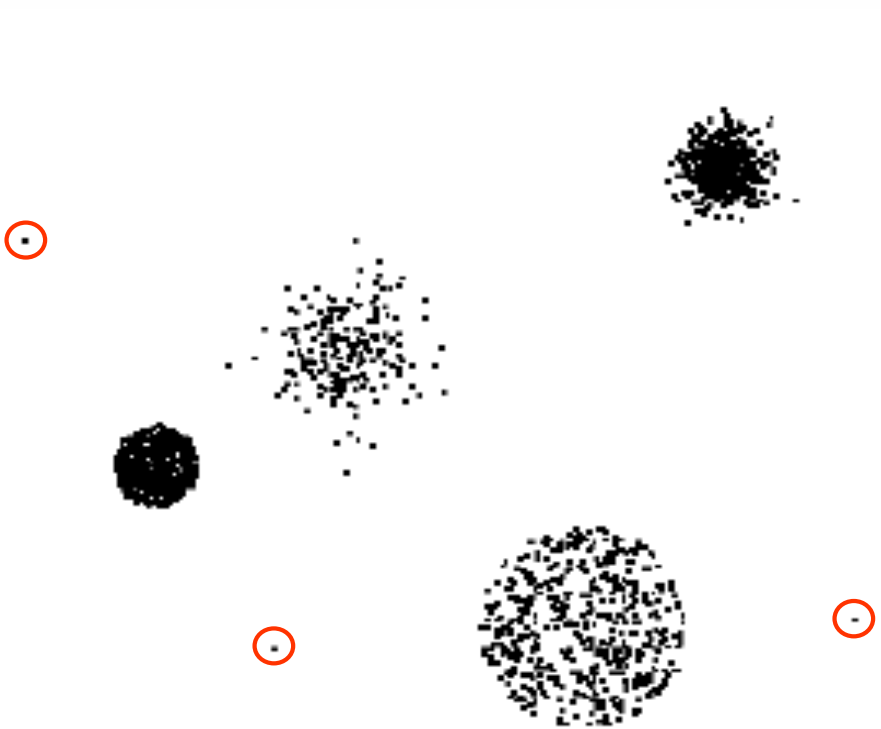
- Handling missing values
  - Eliminate with missing values
  - Imputation
  - Ignore them
  - …

Duplicate Data

- Data set may include data objects that are duplicates, or almost duplicates of one another
- Major issue when merging data from heterogeous sources
- For example, same person with multiple email addresses

Outliers

- Outliers are data objects with characteristics that are considerably different than most of the other data objects in the data set



Sampling

Sampling

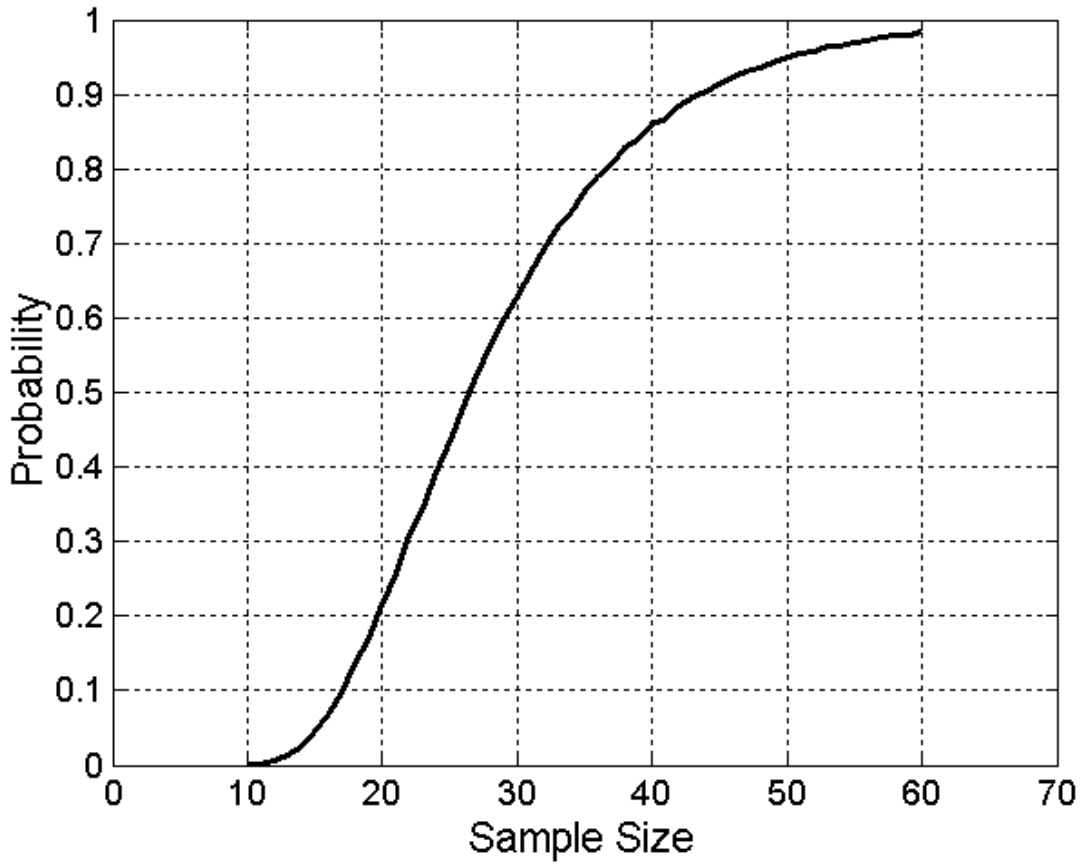
- Sampling is the main technique employed for data selection
- It is often used for both the preliminary investigation of the data and the final data analysis
- Statisticians sample because obtaining the entire set of data of interest is too expensive or time consuming
- Sampling is used in data mining because processing the entire set of data of interest is too expensive or time consuming
- Using a sample will work almost as well as using the entire data sets, if the sample is representative
- A sample is representative if it has approximately the same property (of interest) as the original set of data

Types of Sampling

- Simple Random Sampling
  - There is an equal probability of selecting any particular item
- Sampling without replacement
  - As each item is selected, it is removed from the population
- Sampling with replacement
  - Objects are not removed from the population as they are selected for the sample.
  - In sampling with replacement, the same object can be picked up more than once
- Stratified sampling
  - Split the data into several homogeneous partitions
  - Then draw random samples from each partition

Sample Size

- What sample size is necessary to get at least one object from each of 10 groups.



Aggregation

Aggregation

- Combining two or more attributes (or objects) into a single attribute (or object)
- Data reduction
  - Reduce the number of attributes or objects
- Change of scale
  - Cities aggregated into regions
  - States, countries, etc
- More "stable" data
  - Aggregated data tends to have less variability

Feature Creation

Feature Creation

- Create new attributes that can capture the important information in a data set much more efficiently than the original attributes
- For instance
  - Given the birthday, create the attribute age
  - Given an address, generate longitude and latitude coordinates
- Three general methodologies
  - Feature extraction (domain-specific)
  - Mapping data to a new space
  - Feature construction (combining features)

Discretization

Discretization

- Three types of attributes
  - Nominal: values from an unordered set, e.g., color
  - Ordinal: values from an ordered set,
    e.g., military or academic rank
  - Continuous: real numbers,
    e.g., integer or real numbers
- Discretization
  - Divide the range of a continuous attribute into intervals
  - Some classification algorithms only accept categorical attributes
  - Reduce data size by discretization
  - Prepare for further analysis

Discretization Approaches

- Supervised
  - Attributes are discretized using the class information
  - Generates intervals that tries to minimize the loss of information about the class
- Unsupervised
  - Attributes are discretized solely based on their values

Feature Selection
(Dimensionality Reduction)

Feature Selection (Dimensionality Reduction)

- Purpose
  - Avoid curse of dimensionality
  - Reduce amount of time and memory required by data mining algorithms
  - Allow data to be more easily visualized
  - May help to eliminate irrelevant features or reduce noise
- Techniques
  - Principle Component Analysis
  - Singular Value Decomposition
  - Others: supervised and non-linear techniques

Feature Selection

- Redundant features
  - Duplicate much or all of the information contained in one or more other attributes
  - For example, the purchase price of a product and the amount of sales tax paid
- Irrelevant features
  - Contain no information that is useful for the data mining task at hand
  - For example, the students' ID is often irrelevant to the task of predicting students' average grade

Feature Selection

- Brute-force approach
  - Try all possible feature subsets as input to data mining algorithm
- Embedded approaches
  - Feature selection occurs naturally as part of the data mining algorithm (e.g., Lasso)
- Filter approaches
  - Features are selected using a procedure that is independent from a specific data mining algorithm
  - E.g., attributes are selected based on correlation measures
- Wrapper approaches
  - Use a data mining algorithm as a black box to find best subset of attributes
  - E.g., apply a genetic algorithm and an algorithm for decision tree to find the best set of features for a decision tree

Filter Approach

Dimensionality Reduction: Principal Component Analysis (PCA)

- Principal Component Analysis (PCA) that seeks a basis that best captures the variance in the data
- The direction with the largest projected variance is called the first principal component
- The orthogonal direction that captures the second largest projected variance is called the second principal component, and so on

Dimensionality Reduction: Principal Component Analysis (PCA)

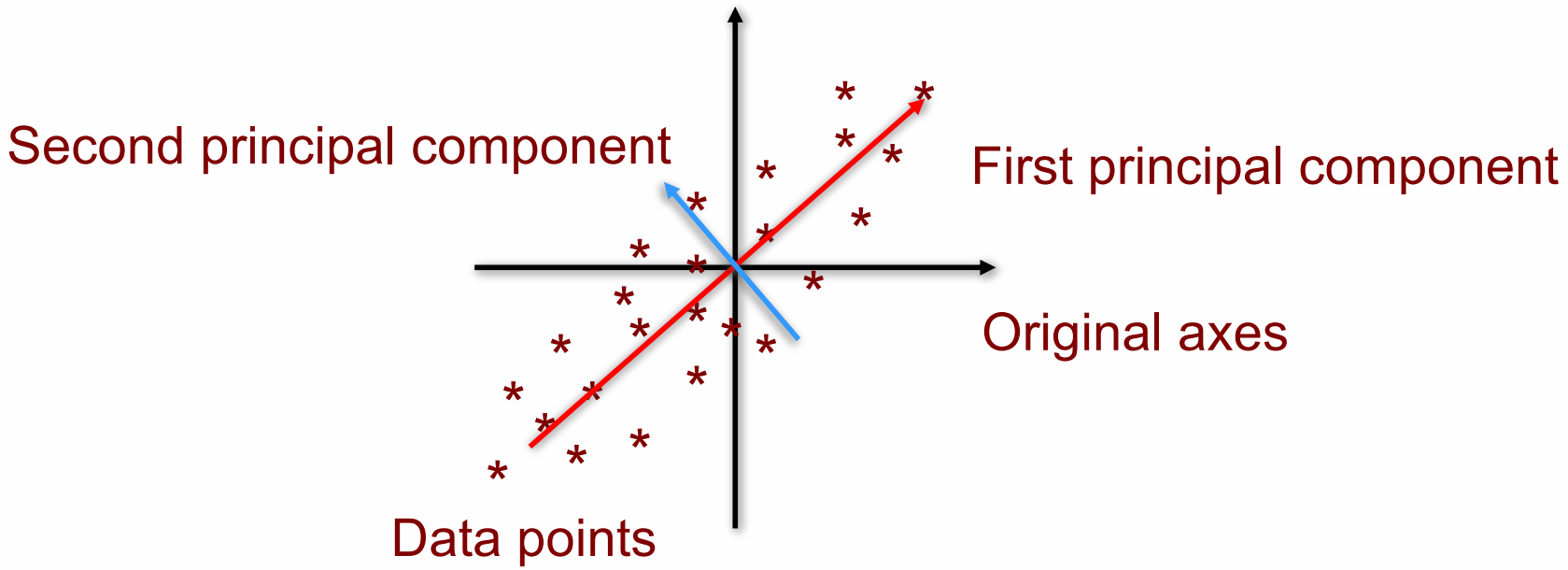
- Given N data vectors from n-dimensions, find k<n orthogonal vectors (the principal components) that can be best used to represent data
- Works for numeric data only
- Used when the number of dimensions is large

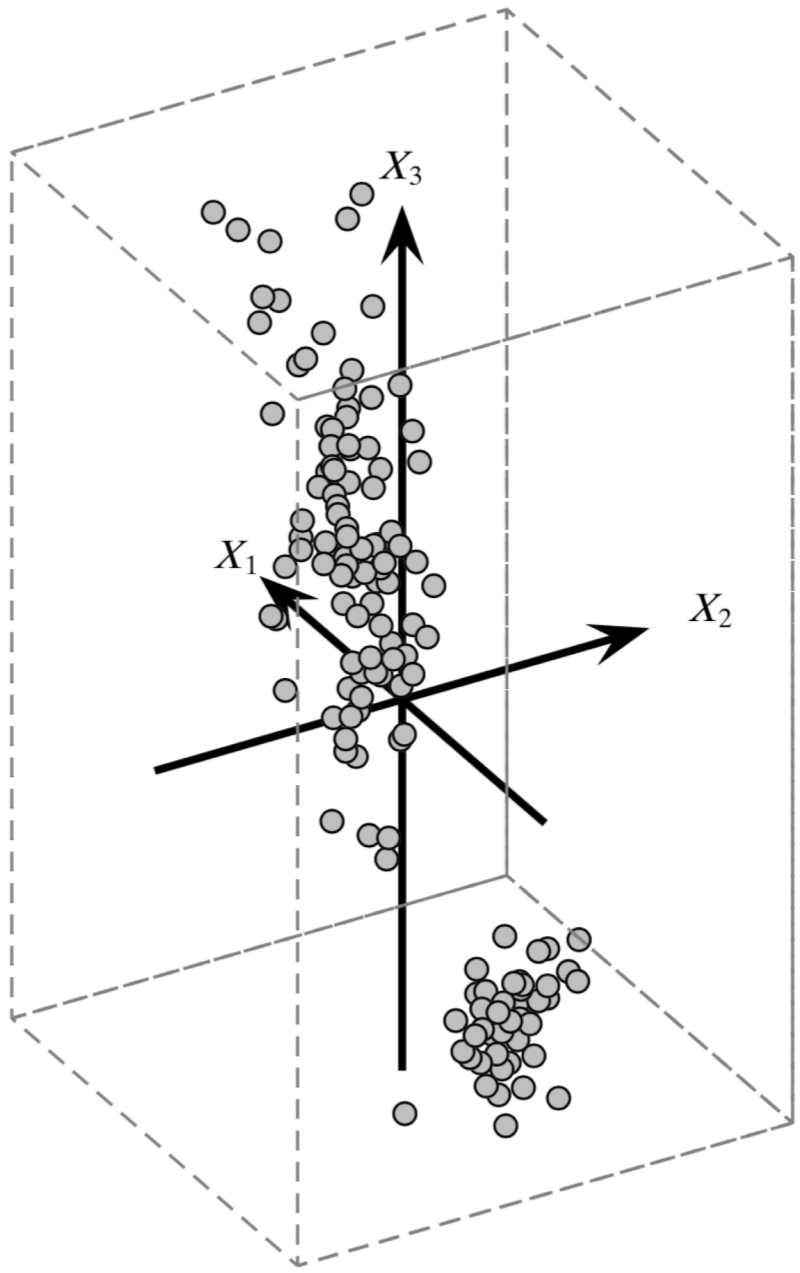
Dimensionality Reduction: Principal Component Analysis (PCA)

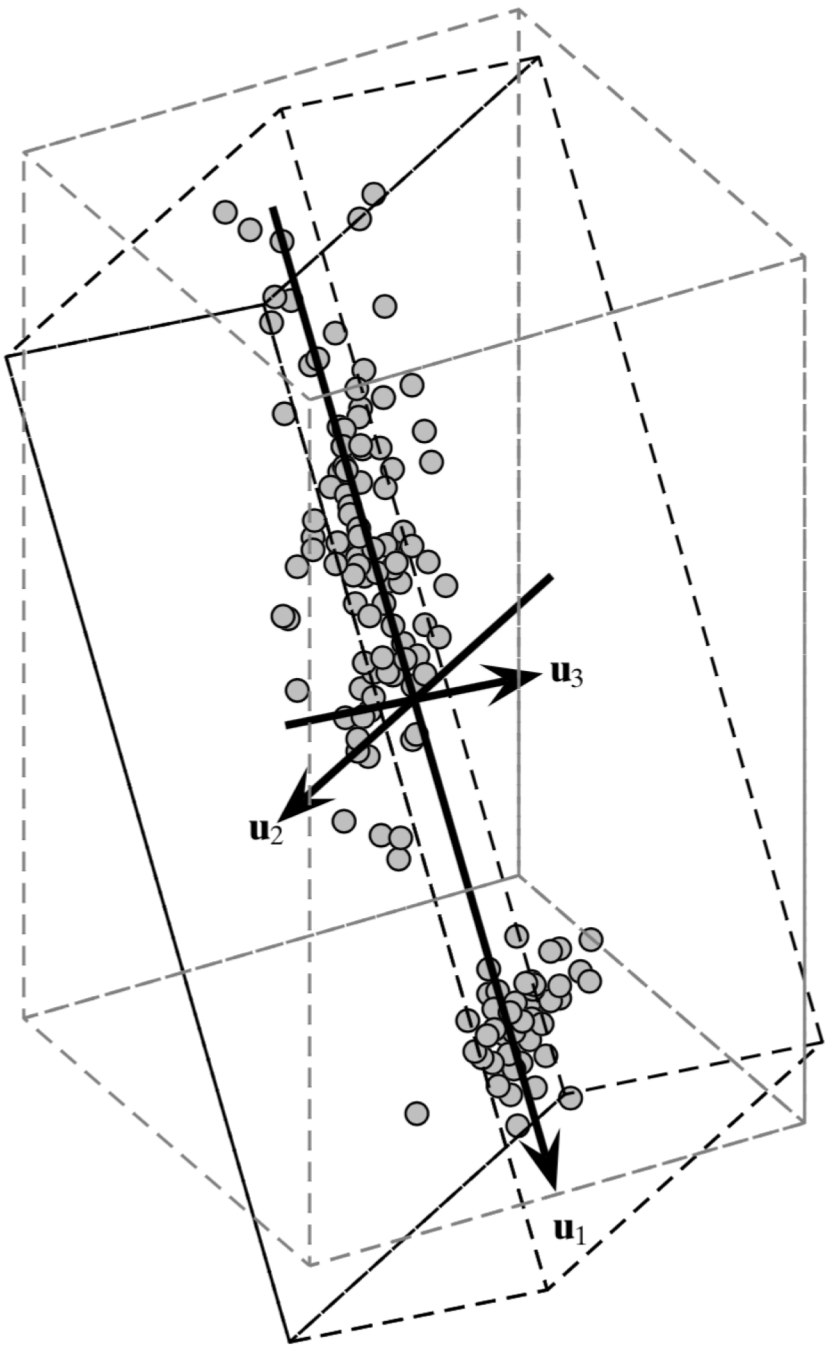
- Normalize input data
- Compute k orthonormal (unit) vectors, i.e., principal components
- Each input data (vector) is a linear combination of the k principal component vectors
- The principal components are sorted in order of decreasing "significance" or strength
- Since the components are sorted, the size of the data can be reduced by eliminating the weak components, i.e., those with low variance. (i.e., using the strongest principal components, it is possible to reconstruct a good approximation of the original data

Dimensionality Reduction: Principal Component Analysis (PCA)

- Goal is to find a projection that captures the largest amount of variation in data

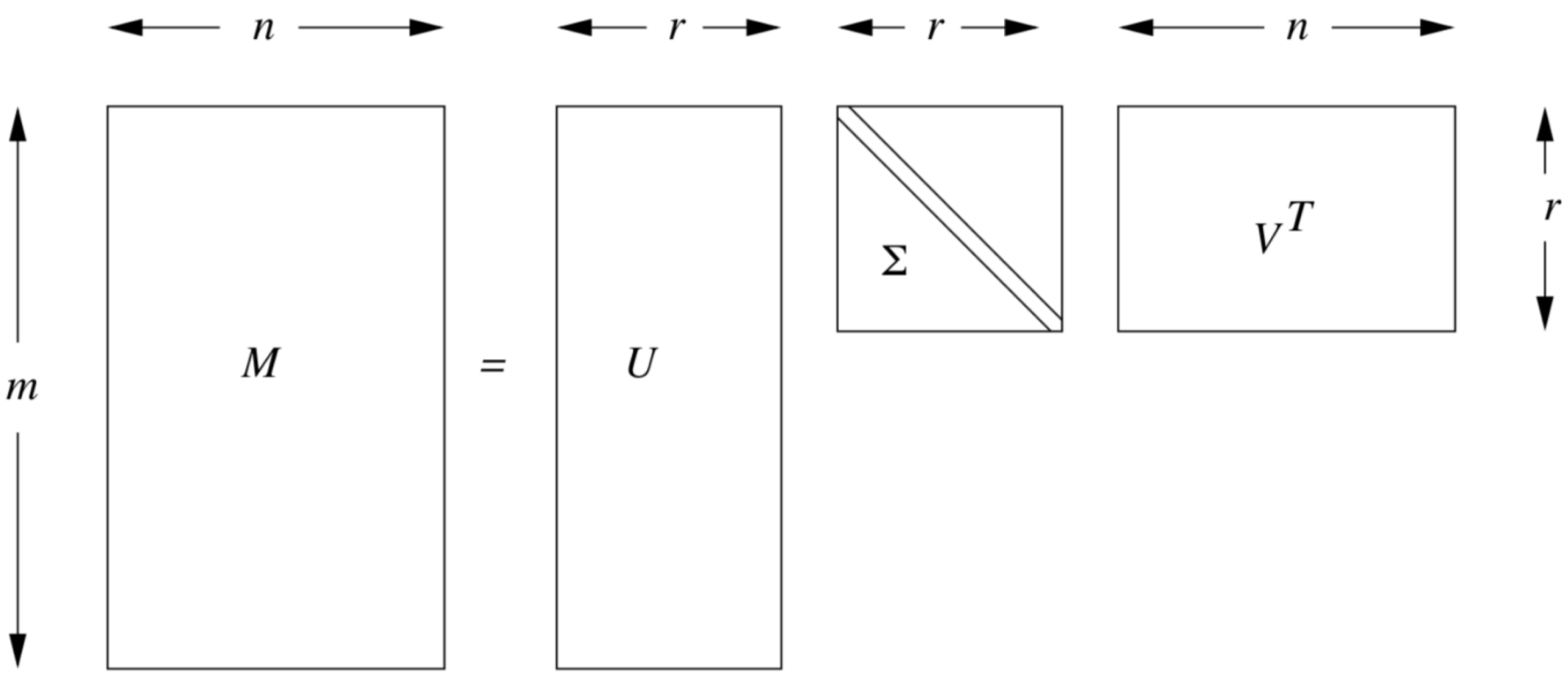



X_3
X_1
X_2


$\mathbf{u}_3$
$\mathbf{u}_2$
$\mathbf{u}_1$

Singular Value Decomposition (SVD)

- Principal components analysis is a special case of a more general matrix decomposition method called Singular Value Decomposition (SVD)
- It is always possible to decompose matrix M into $M = U\mathbf{\Sigma}V^T$ where U, $\mathbf{\Sigma}$, V are unique
- U, V are column orthonormal, i.e., columns are unit vectors, orthogonal to each other, so that $U^TU = I$; $V^TV = I$
- $\mathbf{\Sigma}_{i,i}$ are the singular values, they are positive, and sorted in decreasing order



The form of a singular-value decomposition
(Mining Massive Datasets)

Singular Value Decomposition

- M stores the original data, U represents the n examples using r new concepts/attributes
- Σ represents the strength of each 'concept' (r is the rank of A)
- V contain m terms, r concepts
- $\Sigma_{i,i}$ are called the singular values of M

	Matrix	Alien	Star Wars	Casablanca	Titanic
Joe	1	1	1	0	0
Jim	3	3	3	0	0
John	4	4	4	0	0
Jack	5	5	5	0	0
Jill	0	0	0	4	4
Jenny	0	0	0	5	5
Jane	0	0	0	2	2

Ratings of movies by users
(Fig 11.6 from Mining Massive Datasets)

relates people to movies

concepts strength (sf & romance)

relates movies to people

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 3 & 3 & 3 & 0 & 0 \\ 4 & 4 & 4 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 0 & 0 & 4 & 4 \\ 0 & 0 & 0 & 5 & 5 \\ 0 & 0 & 0 & 2 & 2 \end{bmatrix} = \begin{bmatrix} .14 & 0 \\ .42 & 0 \\ .56 & 0 \\ .70 & 0 \\ 0 & .60 \\ 0 & .75 \\ 0 & .30 \end{bmatrix} \begin{bmatrix} 12.4 & 0 \\ 0 & 9.5 \end{bmatrix} \begin{bmatrix} .58 & .58 & .58 & 0 & 0 \\ 0 & 0 & 0 & .71 & .71 \end{bmatrix}$$

M U Σ V^{T}

SVD for matrix M
(Fig 11.6 from Mining Massive Datasets)

	Matrix	Alien	Star Wars	Casablanca	Titanic
Joe	1	1	1	0	0
Jim	3	3	3	0	0
John	4	4	4	0	0
Jack	5	5	5	0	0
Jill	0	2	0	4	4
Jenny	0	0	0	5	5
Jane	0	1	0	2	2

Ratings of movies by users as matrix M'
(Fig 11.8 from Mining Massive Datasets)

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 3 & 3 & 3 & 0 & 0 \\ 4 & 4 & 4 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 2 & 0 & 4 & 4 \\ 0 & 0 & 0 & 5 & 5 \\ 0 & 1 & 0 & 2 & 2 \end{bmatrix} =$$

M'

$$\begin{bmatrix} .13 & .02 & -.01 \\ .41 & .07 & -.03 \\ .55 & .09 & -.04 \\ .68 & .11 & -.05 \\ .15 & -.59 & .65 \\ .07 & -.73 & -.67 \\ .07 & -.29 & .32 \end{bmatrix} \begin{bmatrix} 12.4 & 0 & 0 \\ 0 & 9.5 & 0 \\ 0 & 0 & 1.3 \end{bmatrix} \begin{bmatrix} .56 & .59 & .56 & .09 & .09 \\ .12 & -.02 & .12 & -.69 & -.69 \\ .40 & -.80 & .40 & .09 & .09 \end{bmatrix}$$

U Σ V^{T}

SVD for matrix M'
(Fig 11.6 from Mining Massive Datasets)

by eliminating some of the least important concepts …

we can simplify the representation

$$\begin{bmatrix} .13 & .02 \\ .41 & .07 \\ .55 & .09 \\ .68 & .11 \\ .15 & -.59 \\ .07 & -.73 \\ .07 & -.29 \end{bmatrix} \begin{bmatrix} 12.4 & 0 \\ 0 & 9.5 \end{bmatrix} \begin{bmatrix} .56 & .59 & .56 & .09 & .09 \\ .12 & -.02 & .12 & -.69 & -.69 \end{bmatrix}$$

$$= \begin{bmatrix} 0.93 & 0.95 & 0.93 & .014 & .014 \\ 2.93 & 2.99 & 2.93 & .000 & .000 \\ 3.92 & 4.01 & 3.92 & .026 & .026 \\ 4.84 & 4.96 & 4.84 & .040 & .040 \\ 0.37 & 1.21 & 0.37 & 4.04 & 4.04 \\ 0.35 & 0.65 & 0.35 & 4.87 & 4.87 \\ 0.16 & 0.57 & 0.16 & 1.98 & 1.98 \end{bmatrix}$$

Dropping the lowest singular value from the decomposition of M'
(Fig 11.10 from Mining Massive Datasets)

How Many Singular Values?

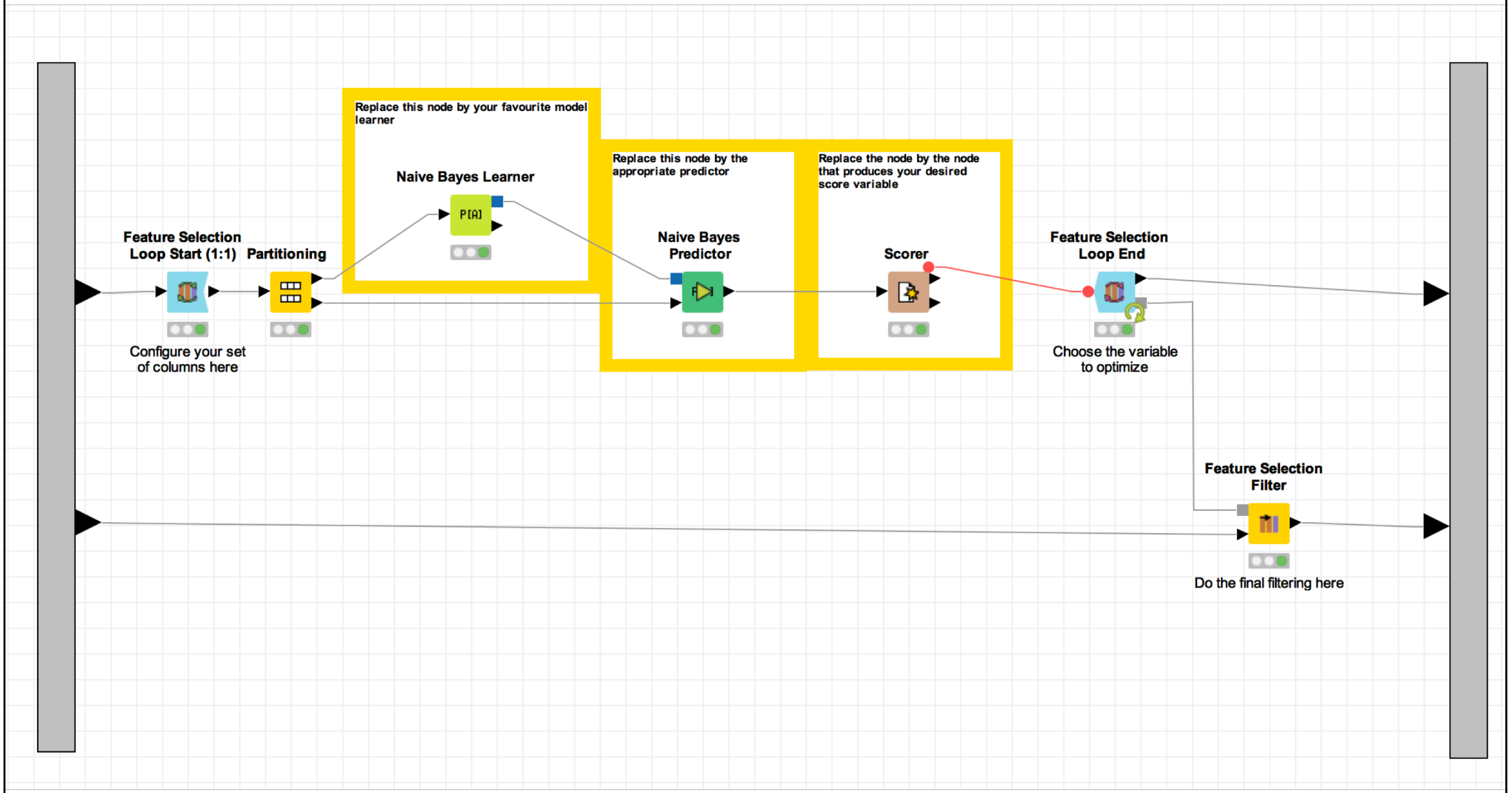
- The rule of thumb is to retain enough singular values to make up 90% of the energy in Σ.
- That is, the sum of the squares of the retained singular values should be at least 90% of the sum of the squares of all the singular values.
- In the previous example,
  - The total energy is $(12.4)^2 + (9.5)^2 + (1.3)^2 = 245.70$
  - The retained energy is $(12.4)^2 + (9.5)^2 = 244.01$. Thus, we have retained over 99% of the energy.
  - However, if we would eliminate the second singular value, 9.5, the retained energy would be only about 63%.

Should we include the class attribute in the feature selection process?

Wrapper Approach

Wrapper Approach

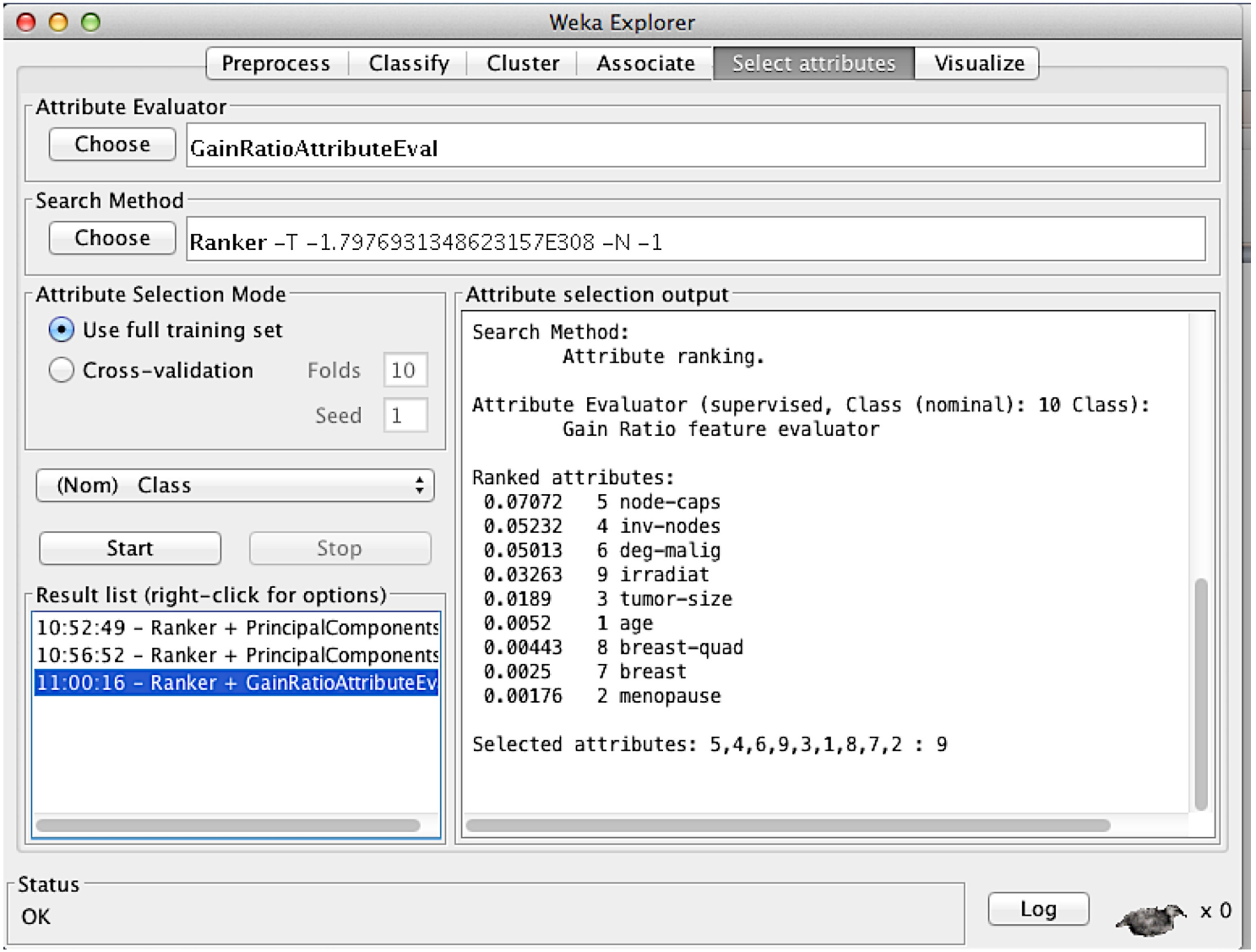
- Focus on a specific data mining algorithm and find the best subset of attributes for that specific algorithm
- For example, if you are planning to use a decision tree, you are looking for the subset of features that maximize the tree performance
- The target algorithm is used as a black box and a search method is applied to find the subset of features that maximize the performance or minimizes the error



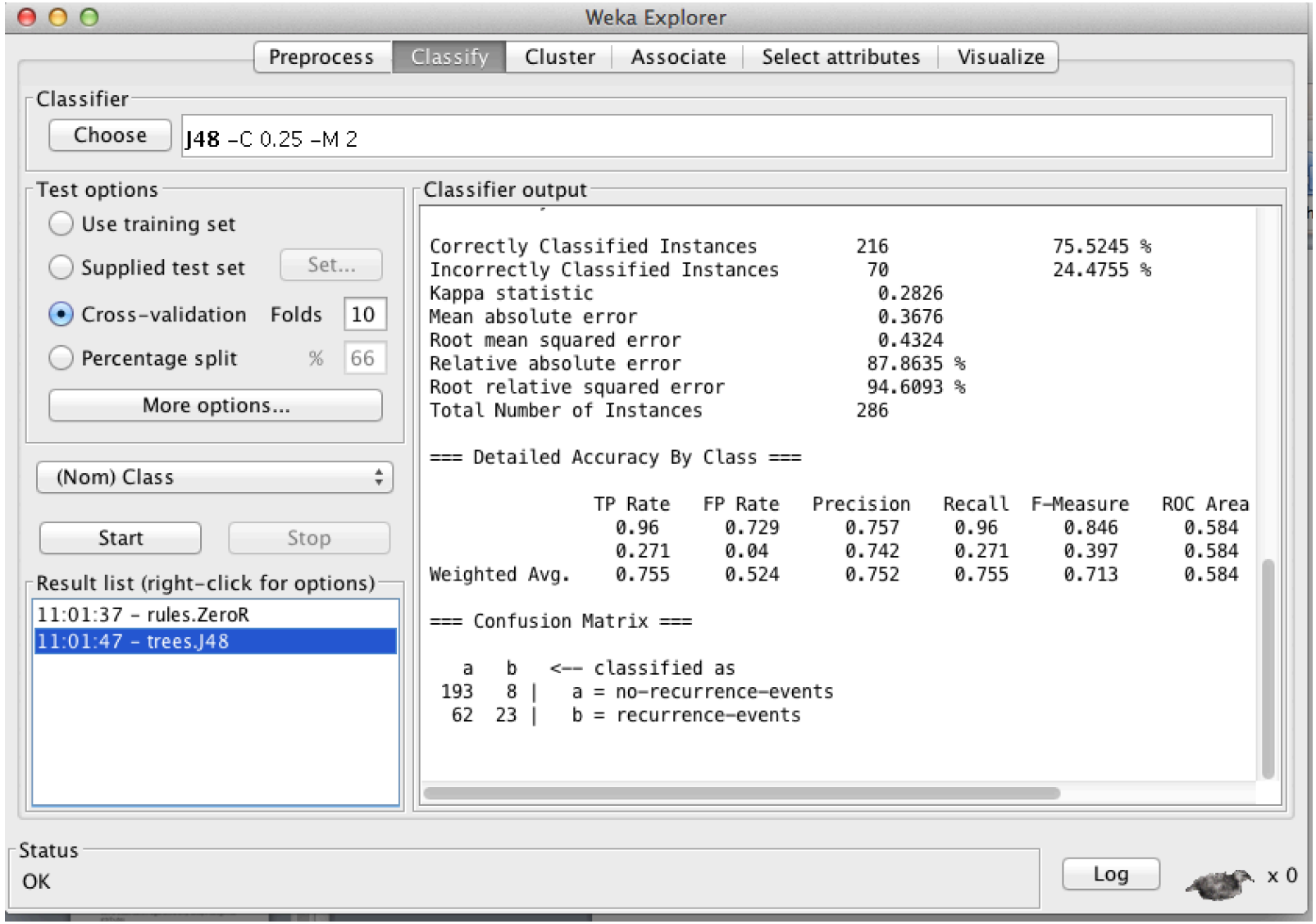
Example of KNIME workflow implementing a wrapper approach to feature selection.

Examples of Wrapper Approaches

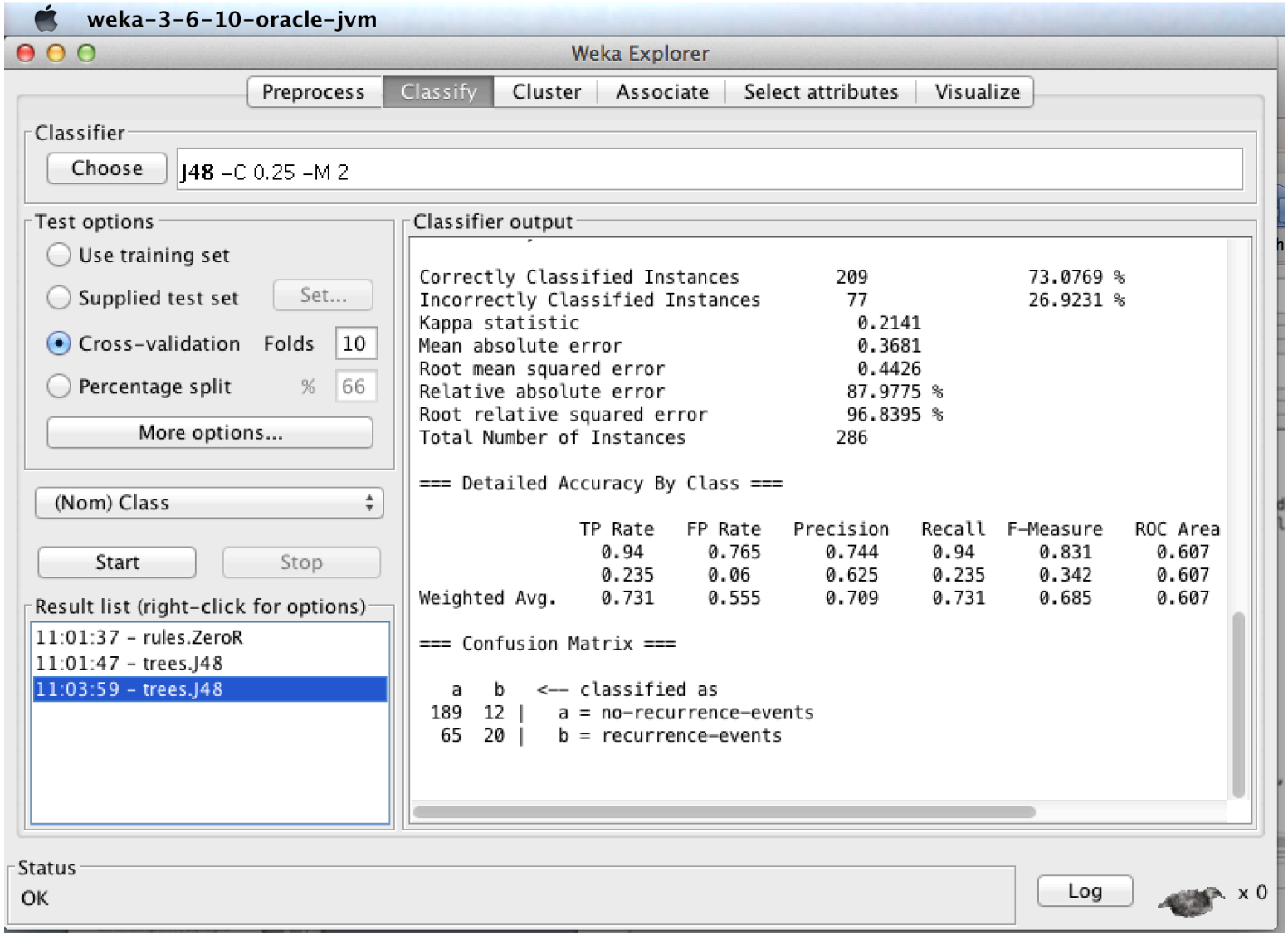
- Backward Feature Elimination
  - Start training on n input features. Then remove one input feature at a time and train the same model on n-1 input
  - The input feature whose removal has produced the smallest increase in the error rate is removed.
  - Each iteration k produces a model trained on n-k features and an error rate e(k). The process stops when the maximum tolerable error rate i reached.
- Forward Feature Construction
  - The inverse process to the Backward Feature Elimination. It starts with one feature and progressively adds the feature that the highest increase in performance.
- Recursive Feature elimination
  - It repeatedly creates models and keeps aside the best or the worst performing feature at each iteration. It constructs the next model with the left features until all the features are exhausted. It then ranks the features based on the order of their elimination.
- …



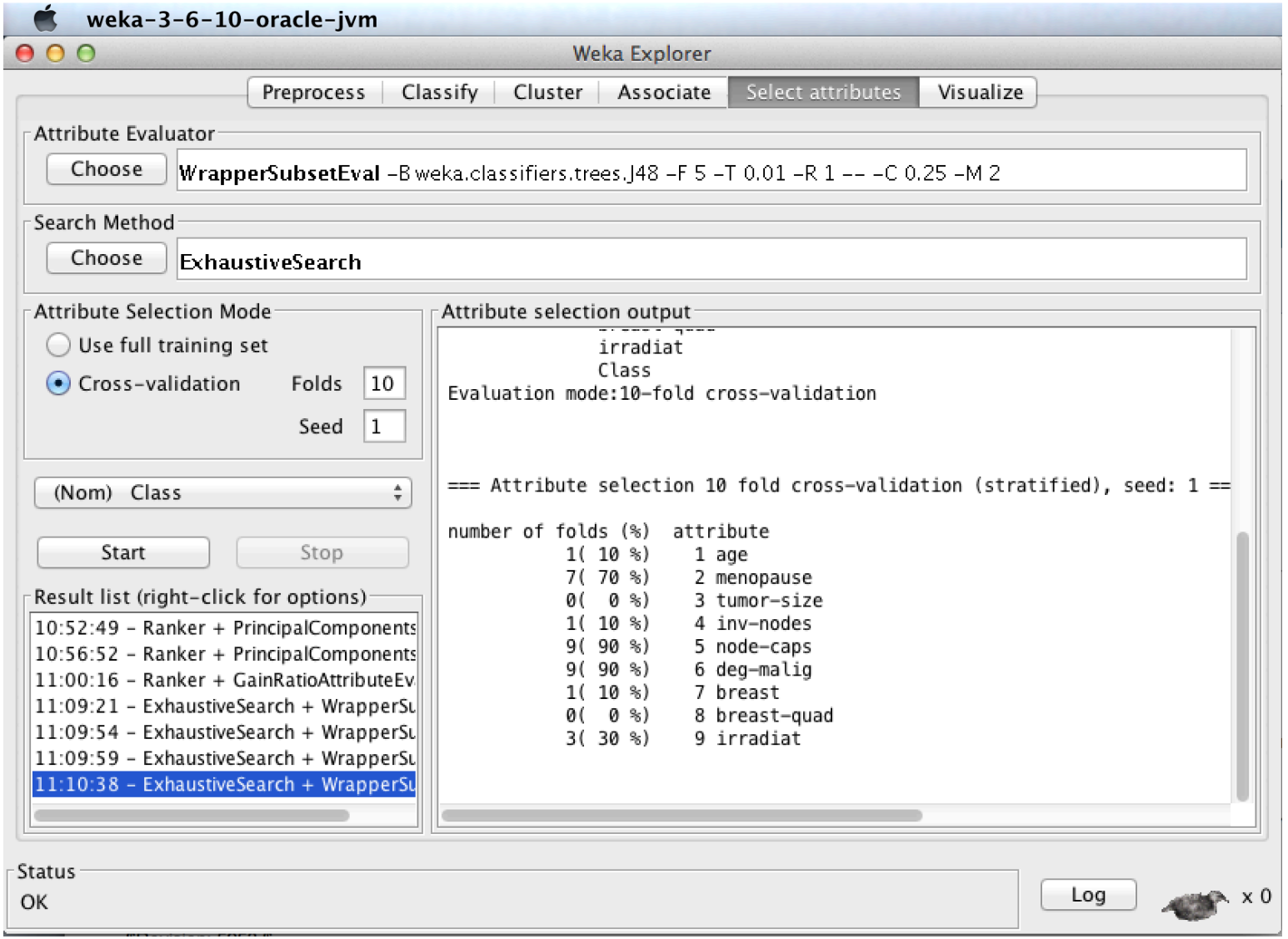
Example of using the information gain to rank the importance of attributes.



Decision trees applied to all the attributes.



Decision trees applied to all the breast dataset without the four least ranked attributes.



Wrapper approach using exhaustive search and decision trees.

Filter Methods vs Wrapper Methods

- Filter methods focus on the relevance of features while wrapper methods measure the usefulness of a subset of feature by actually training a model
- Filter methods are much faster as they do not involve training
- Wrapper methods are computationally very expensive as well
- Filter methods use statistical methods for evaluation of a subset of features while wrapper methods use cross validation
- Filter methods might fail to find the best subset of features in many occasions but wrapper methods can always provide the best subset of features
- Using the subset of features from the wrapper methods make the model more prone to overfitting

Examples from Scikit-Learn

- Filter Approach
  - VarianceThreshold removes all features whose variance doesn't meet some threshold. By default, it removes all zero-variance features.
  - Univariate feature selection methods works by selecting the best features based on univariate statistical tests (SelectKBest, SelectPercentile, SelectFdr, SelectFwe, GenericUnivariateSelect)
- Wrapper Approach
  - Recursive feature elimination
  - SelectFromModel selects features based on the model coef_ or feature_importances_ fields

Run the Python notebooks
for this lecture