Fundamentals of Data Mining

3. DATA PREPROCESSING

Hamid Fadishei, Assistant Professor

CE Department, University of Bojnord

fadishei@yahoo.com, http://www.fadishei.ir

Introduction

Recorded data may be inaccurate, incomplete, and inconsistent

- Data collection instruments used may be faulty
- There may have been human or computer errors occurring at data entry
- Users may purposely submit incorrect data values for mandatory fields when they do not wish to submit personal information
- Errors in data transmission can also occur
- Incorrect data may also result from inconsistencies in naming conventions or data codes, or inconsistent formats for input fields
- Data may not be included simply because they were not considered important at the time of entry

Low-quality data will lead to low-quality mining results

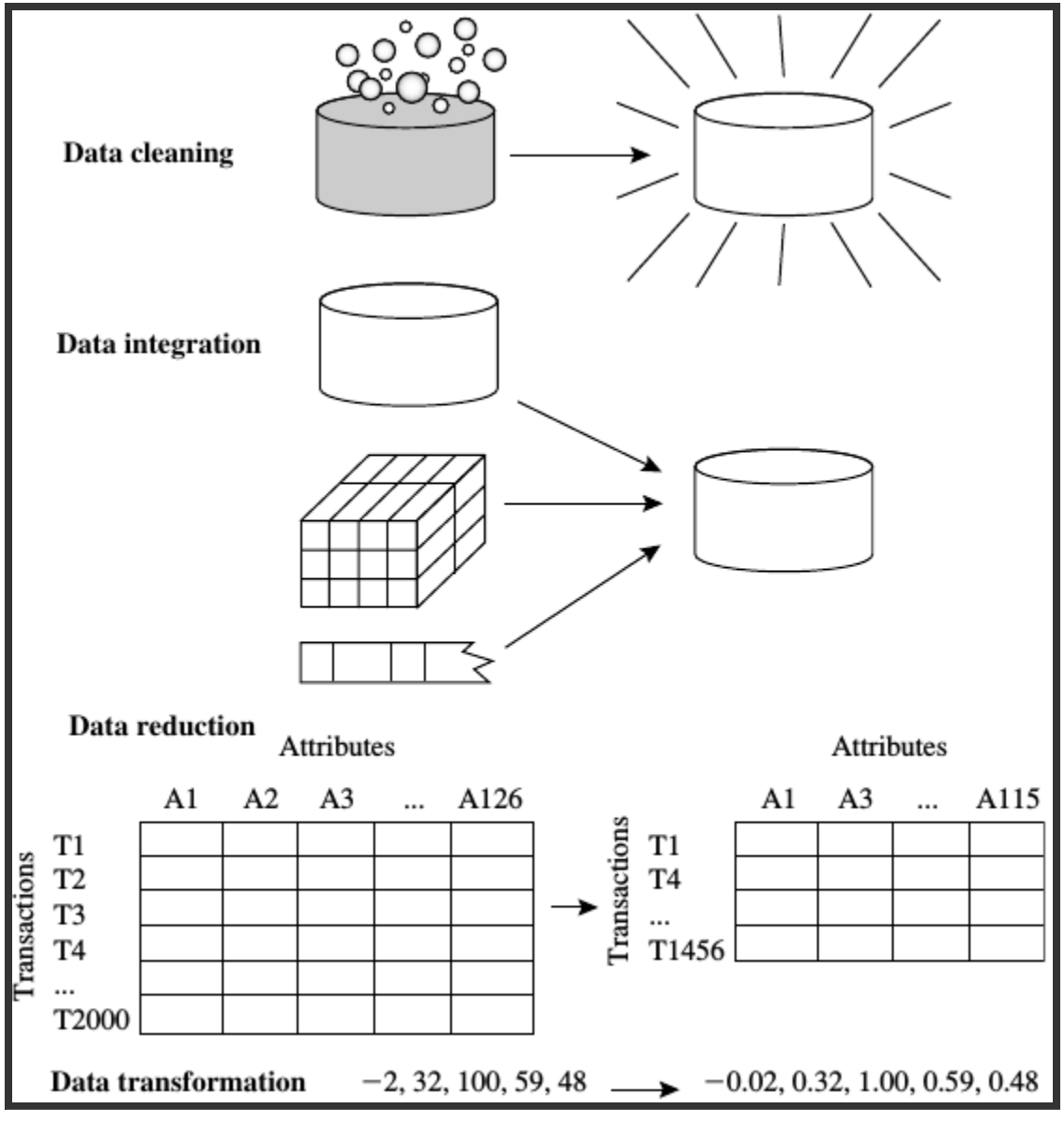
Data preprocessing

Preprocessing improves the quality of data

Data preprocessing includes…

- Data cleaning
  - Removes noise and correct inconsistencies in data
- Data integration
  - Merges data from multiple sources into a coherent data store such as a data warehouse
- Data reduction
  - Reduces data size or complexity by, for instance, aggregating, eliminating redundant features, or clustering
- Data transformation
  - For example, normalization which scales data to a new range like 0.0 to 1.0

Data preprocessing



Data cleaning

Data cleaning involves treating:

- Missing values
- Noisy data

Missing values

Methods for treating missing values

- Ignore the tuple
  - For example, when the class attribute is missing. Not very effective. Discarded attributes could have been used
- Fill in the missing value manually
  - Time consuming. Sometimes not feasible
- Use a global constant to fill in the missing value
  - Use constants like "Unknown" or "0". Simple but not foolproof
- Use a measure of central tendency for the attribute to fill in the missing value
  - For example, mean for normal data or median for skewed data.
- Use the attribute mean or median for all samples belonging to the same class as the given tuple
  - Use mean or median values specific to each class
- Use the most probable value to fill in the missing value
  - Methods like decision tree, regression, Bayesian, etc. (We'll see them later)

Noisy data

Noise: random error in a measured variable

Smoothing techniques are used to remove noise

- Binning
- Regression
- Outlier analysis

Binning

- Smooth a sorted data value by consulting its neighborhood
- Partition data into equal-sized bins
- Replace the values in each bin by a smoothed version of it

Sorted data for *price* (in dollars): 4, 8, 15, 21, 21, 24, 25, 28, 34

Partition into (equal-frequency) bins:

Bin 1: 4, 8, 15
Bin 2: 21, 21, 24
Bin 3: 25, 28, 34

Smoothing by bin means:

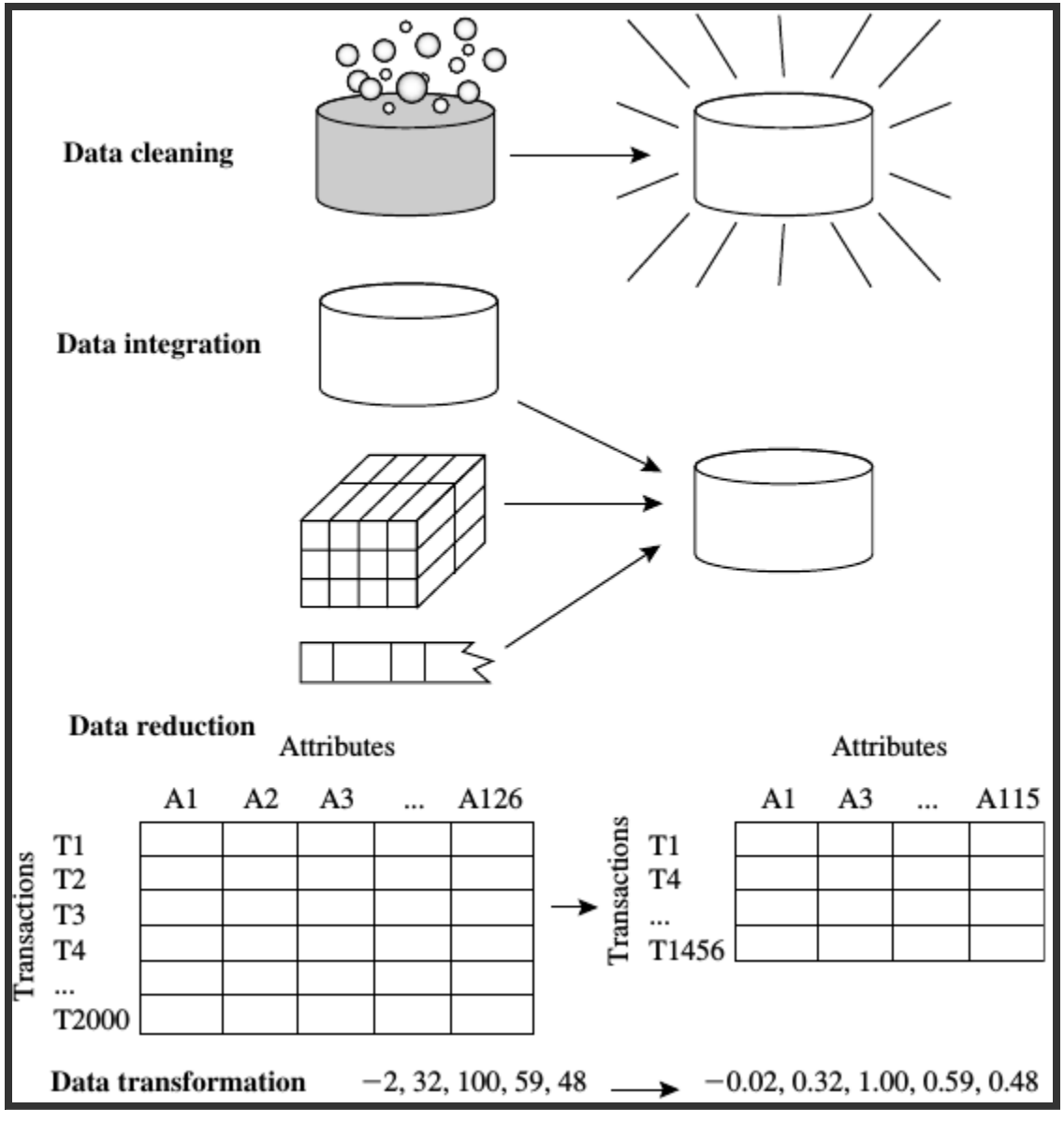
Bin 1: 9, 9, 9
Bin 2: 22, 22, 22
Bin 3: 29, 29, 29

Smoothing by bin boundaries:

Bin 1: 4, 4, 15
Bin 2: 21, 21, 24
Bin 3: 25, 25, 34

Mini-break #4

Data preprocessing



Data reduction

Mining complex and huge data takes a long time and consumes lots of resources

Data reduction techniques can be applied to obtain a reduced representation of the data set that is much smaller in volume, yet closely maintains the integrity of the original data

Mining on the reduced data set should be more efficient yet produce the same (or almost the same) analytical results

Data reduction

Data reduction strategies

- Dimensionality reduction
  - Reduce the number of attributes of a dataset
- Numerosity reduction
  - Reduce the number of dataset instances

Principal component analysis

PCA is a method of dimensionality reduction

- A method of dimensionality reduction
- Combines the essence of attributes by creating an alternative, smaller set of variables
- Searches for k n-dimensional orthogonal vectors that can best be used to represent the data, where $k \leq n$

Attribute subset selection

- A method of dimensionality reduction
- Reduces the data set size by removing irrelevant or redundant attributes
- Additional benefit: It reduces the number of attributes appearing in the discovered patterns, helping to make the patterns easier to understand.
- Exhaustive search is prohibitively expensive (There are 2^n subsets)
- Greedy methods are typically used

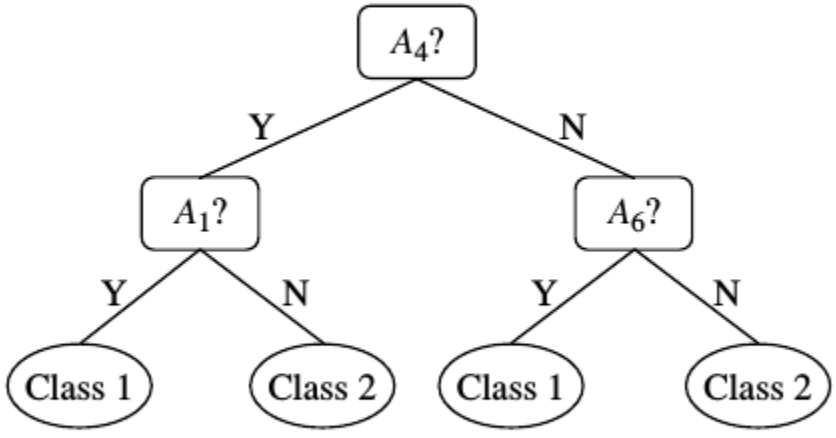
Attribute subset selection

Greedy methods for feature subset selections

- Stepwise forward selection
  - Starts with an empty set of attributes. The best of the original attributes is determined and added to the reduced set iteratively
- Stepwise backward elimination
  - Starts with the full set of attributes. At each step, it removes the worst attribute remaining in the set
- Decision tree induction
  - We'll see later

Attribute subset selection

Greedy methods for feature subset selections

Forward selection	Backward elimination	Decision tree induction
Initial attribute set: $\{A_1, A_2, A_3, A_4, A_5, A_6\}$ Initial reduced set: $\{\}$ => $\{A_1\}$ => $\{A_1, A_4\}$ => Reduced attribute set: $\{A_1, A_4, A_6\}$	Initial attribute set: $\{A_1, A_2, A_3, A_4, A_5, A_6\}$ => $\{A_1, A_3, A_4, A_5, A_6\}$ => $\{A_1, A_4, A_5, A_6\}$ => Reduced attribute set: $\{A_1, A_4, A_6\}$	Initial attribute set: $\{A_1, A_2, A_3, A_4, A_5, A_6\}$  => Reduced attribute set: $\{A_1, A_4, A_6\}$

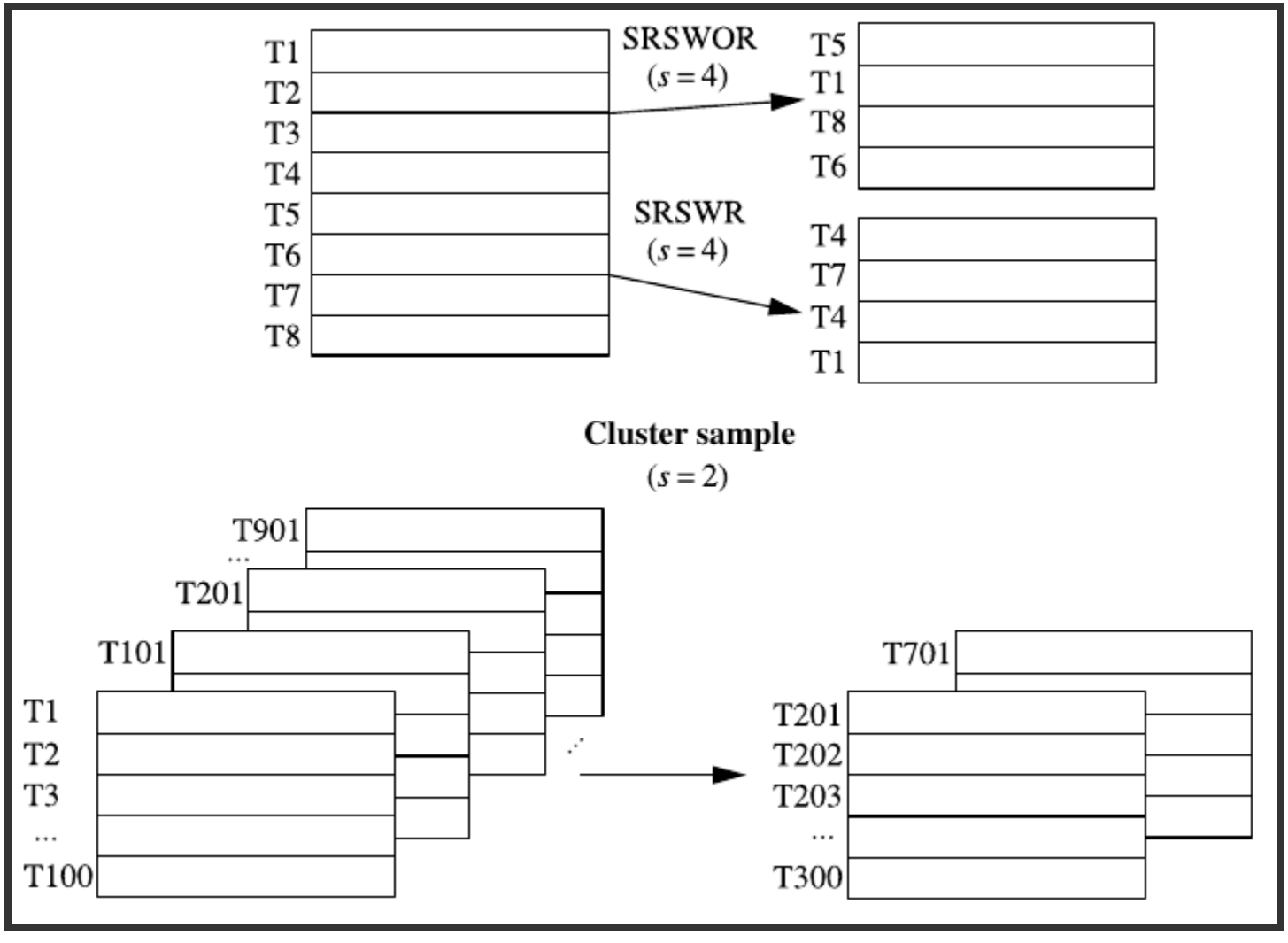
Sampling

Reduces the number of data instances by selecting a subset of them

- Simple random sample without replacement (SRSWOR)
  - draws s of the N tuples from dataset D ($s < N$), where the probability of drawing any tuple in D is $\frac{1}{N}$
  - All tuples are equally likely to be sampled
- Simple random sample with replacement (SRSWR)
  - Similar to SRWOR, but after a tuple is drawn, it is placed back in D so that it may be drawn again
- Cluster sample
  - Tuples in D are grouped into M clusters, then an SRS of s clusters can be obtained, where $s < M$
- Stratified sample

  Dataset D is divided into mutually disjoint parts called strata, then an SRS is obtained from each stratum.

Sampling


T1
T2
T3
T4
T5
T6
T7
T8
SRSWOR
(s = 4)
T5
T1
T8
T6
SRSWR
(s = 4)
T4
T7
T4
T1
Cluster sample
(s = 2)
T901
...
T201
T101
T1
T2
T3
...
T100
T701
T201
T202
T203
...
T300

Sampling

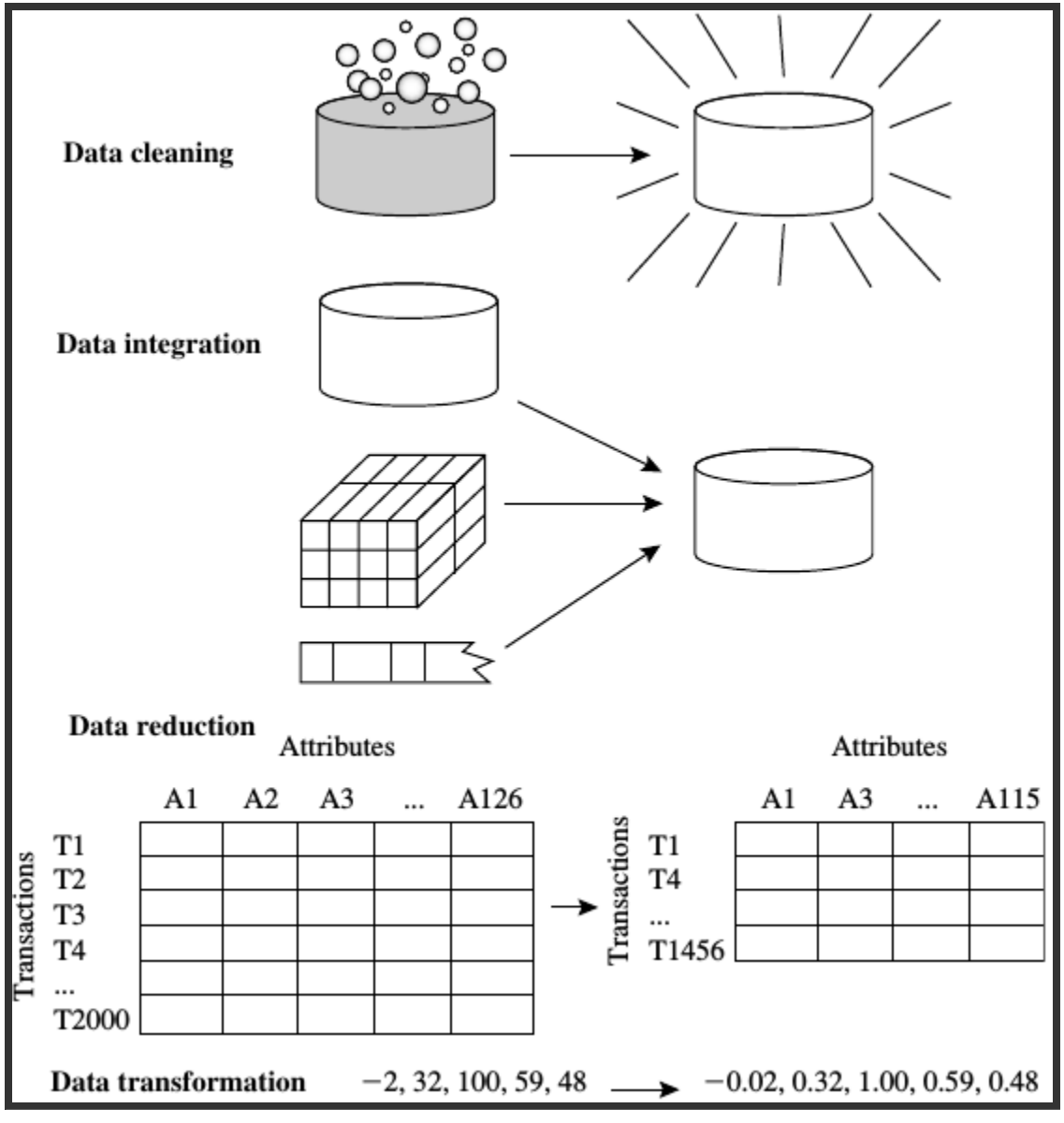
Startified sample
(according to *age*)

T38	youth
T256	youth
T307	youth
T391	youth
T96	middle_aged
T117	middle_aged
T138	middle_aged
T263	middle_aged
T290	middle_aged
T308	middle_aged
T326	middle_aged
T387	middle_aged
T69	senior
T284	senior

T38	youth
T391	youth
T117	middle_aged
T138	middle_aged
T290	middle_aged
T326	middle_aged
T69	senior

Mini-break #5

Data preprocessing



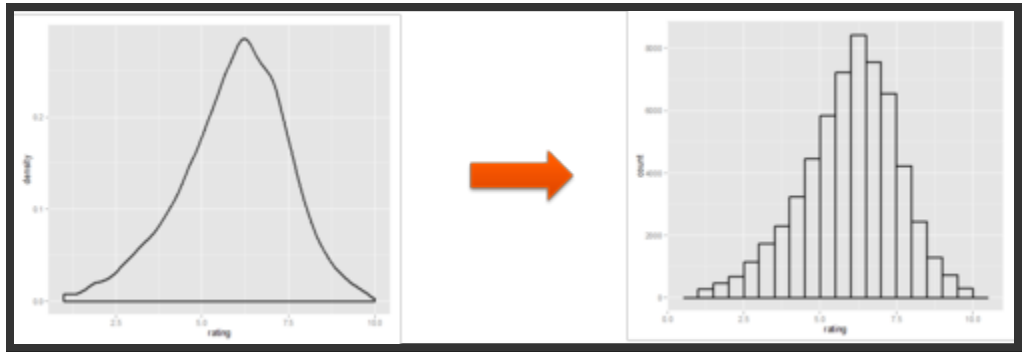
Data transformation

- Discretization
- Normalization

Discretization

Discretization

- Replaces raw data by a smaller number of interval or concept labels
- Simplifies the original data and makes the mining more efficient
- Some data mining algorithm can only operate with categorical data. Discretization can convert numeric data to categorical
- One discretization method is to split data into ranges (of equal width, or equal frequency). Replace each numeric value by the range it belongs to



Normalization

Normalization

- Measurement unit can affect data analysis
- To avoid dependence on measurement units, data should be normalized (standardized)
- Some data mining methods require normalized data to work on (Neural networks, similarity calculations, ...)
- Normalization transforms data to fall within a smaller or common range such as [−1,1] or [0.0, 1.0]

Normalization methods

Min-max normalization

$$v'_i = \frac{v_i - min_A}{max_A - min_A}(newmax_A - newmin_A) + newmin_A$$

Out-of-bounds error for future values

Example

- Suppose that the minimum and maximum values for the attribute income are 12,000 and 98,000
- We would like to map income to the range [0.0, 1.0]
- What is the min-max normalized value for income = 73600?
- Answer = $\frac{73600-12000}{98000-12000}(1.0 - 0) + 0 = 0.716$

Normalization methods

Z-score normalization

$$v'_i = \frac{v_i - \bar{A}}{\sigma_A}$$

Useful when actual minimum and maximum of data are unknown

Example

- Suppose that the mean and standard deviation of the values for the attribute income are 54,000 and 16,000
- We would like to map income to the range [0.0, 1.0]
- What is the z-score normalized value for income = 73600?
- Answer = $\frac{73600-54000}{16000} = 1.225$

Normalization methods

Decimal scaling

$$v'_i = \frac{v_i}{10^j}$$

Where j is the smallest integer such that $max(|v'_i|) < 1$

Example

- Suppose that the recorded values of A range from −986 to 917
- The maximum absolute value of A is 986
- To normalize by decimal scaling, we therefore divide each value by 1000 (i.e., j = 3)