

Preprocessing Lecture Notes (cse352)

Professor Anita Wasilewska

Data Preprocessing

- Why preprocess the data?
- Data cleaning
- Data integration and transformation
- Data reduction
- Discretization and concept hierarchy generation
- Summary

TYPES OF DATA (1)

- Generally we distinguish:
 - Quantitative Data
 - Qualitative Data
- **Bivaluated:** often very useful
- Remember: Null Values are not applicable
- Missing data usually not acceptable

Why Data Preprocessing?

- Data in the real world is dirty
 - incomplete:** lacking attribute values, lacking certain attributes of interest, or containing only aggregate data
 - noisy:** containing errors or outliers
 - inconsistent:** containing discrepancies in codes or names

Why Data Preprocessing?

- No quality data, no quality results!

Quality decisions must be based on quality data
Data warehouse needs consistent integration of
quality data

Data Quality

- A well-accepted multidimensional view of data quality:
 - Accuracy
 - Completeness
 - Consistency
 - Timeliness
 - Believability
 - Interpretability
 - Accessibility

Major Tasks in Data Preprocessing

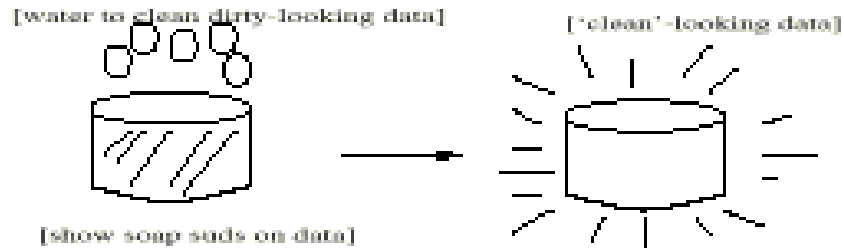
- Data cleaning
 - Fill in missing values, smooth noisy data, identify or remove outliers, and resolve inconsistencies
- Data integration (if needed)
 - Integration of multiple databases, data cubes, or files
- Data transformation
 - Normalization and aggregation

Major Tasks in Data Preprocessing

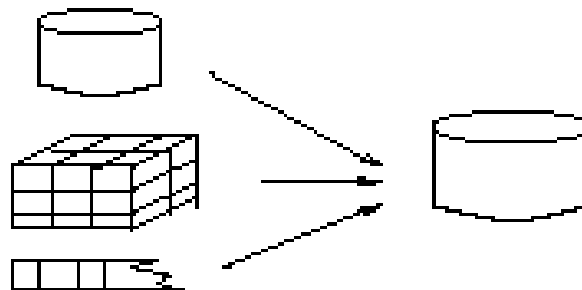
- Data reduction
 - Obtains reduced representation in volume but produces the same or similar analytical results
- Data discretization
 - Part of data reduction but with particular importance, especially for numerical data

Forms of data preprocessing

Data Cleaning



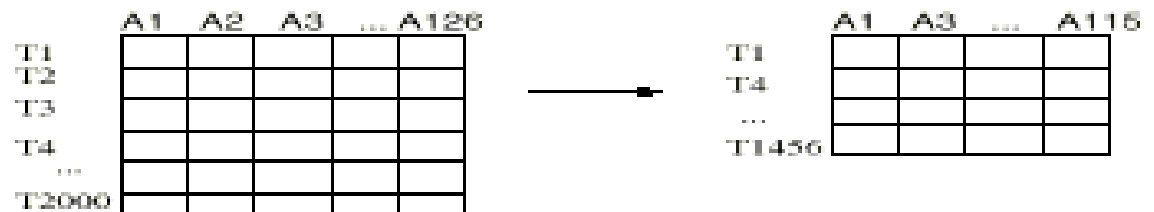
Data Integration



Data Transformation

-2, 32, 100, 59, 48 → -0.02, 0.32, 1.00, 0.59, 0.48

Data Reduction



Data Cleaning

- Data cleaning tasks
 - Fill in missing values
 - Identify outliers and smooth out noisy data
 - Correct inconsistent data

Missing Data

- Data is not always available
- Missing data may be due to
 - equipment malfunction
 - inconsistent with other recorded data and thus deleted
 - data not entered due to misunderstanding
 - certain data may not be considered important at the time of entry
 - not register history or changes of the data
- Missing data may need to be inferred.

How to Handle Missing Data?

- (1). **Ignore** the tuple (record) : usually done when class label (a value of the classification attribute) is missing (assuming the tasks in classification)
- It is not effective when the percentage of missing values per attribute varies considerably.
- (2) **Fill in** the missing value manually: tedious + often infeasible?
- (3) **Fill in** the missing value automatically (methods to follow)

Fill in Missing Data

- (1) **Use a global constant** to fill in the missing value (not efficient and often incorrect)
- (2) **Use the attribute values mean** to fill in the missing value
- (3) Use the attribute values mean for all **samples belonging to the same class (in case of the classification)** to fill in the missing value: smarter than (2)
- (4) **Use the most probable value** to fill in the missing value

Noisy Data

- **Noise:** random error or variance in a measured variable (numeric attribute value)
- **Incorrect attribute values** may due to faulty data collection instruments, data entry problems, data transmission problems, technology limitation, inconsistency in naming convention

Other Data Problems

- Other data problems which requires data cleaning
 - duplicate records
 - incomplete data
 - inconsistent data

How to Handle Noisy Data?

- **Binning method:**
 - first sort data (values of the attribute we consider) and partition them into (equal-depth) bins
 - then apply one of the methods:
 - **smooth by bin means**, (replace noisy values in the bin by the bin mean)
 - **smooth by bin median**, (replace noisy values in the bin by the bin median)
 - **smooth by bin boundaries**, (replace noisy values in the bin by the bin boundaries)

Binning method is also used for **data discretization**

How to Handle Noisy Data?

- **Clustering**
 - Are used to detect and remove outliers (in the attributes values)
- **Combined computer and human inspection**
 - detect suspicious attribute values and check by human
- **Regression**
 - smooth by fitting the attribute values into regression functions

Simple Discretization Methods:

Binning

- **Equal-width** (distance) partitioning:
 - It divides the range (values of a given attribute)
 - into N intervals of equal size: uniform grid
 - if A and B are the lowest and highest values of the attribute, the width of intervals will be:
$$W = (B-A)/N.$$
 - The most straightforward
 - But outliers may dominate presentation
 - Skewed data is not handled well.

Simple Discretization Methods:

Binning

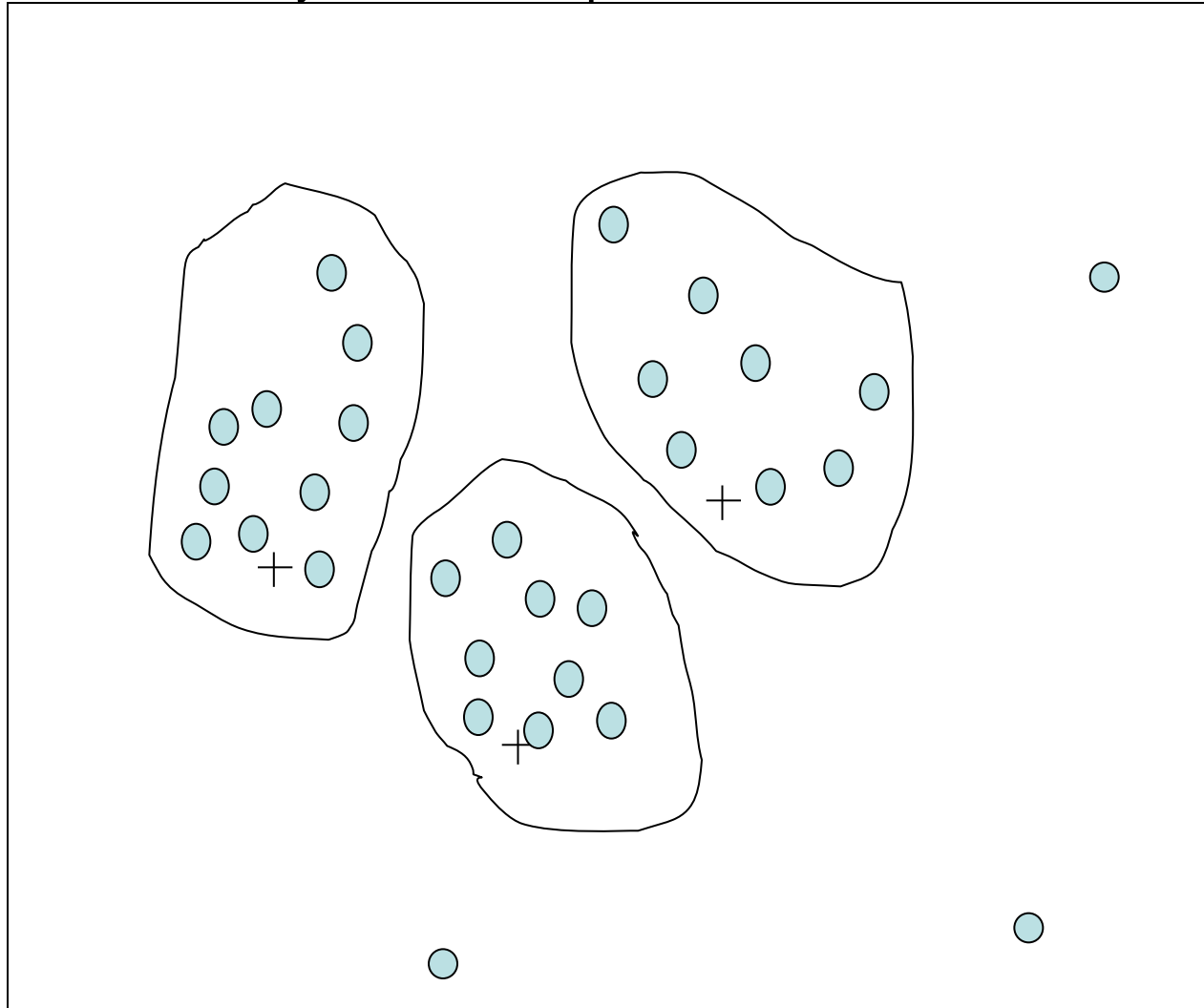
- **Equal-depth** (frequency) partitioning:
 - It divides the range (values of a given attribute)
 - into N intervals, each containing approximately same number of samples (elements)
 - Good data scaling
 - Managing categorical attributes can be tricky.

Binning Methods for Data Smoothing (example)

- Sorted data (attribute values) for price (attribute: price in dollars):
4, 8, 9, 15, 21, 21, 24, 25, 26, 28, 29, 34
- Partition into (equal-depth) bins:
 - Bin 1: 4, 8, 9, 15
 - Bin 2: 21, 21, 24, 25
 - Bin 3: 26, 28, 29, 34
- Smoothing by bin means:
 - Bin 1: 9, 9, 9, 9
 - Bin 2: 23, 23, 23, 23
 - Bin 3: 29, 29, 29, 29
- Smoothing by bin boundaries:
 - Bin 1: 4, 4, 4, 15
 - Bin 2: 21, 21, 25, 25
 - Bin 3: 26, 26, 26, 34
- Replace all values in a BIN by ONE value , or corresponding 2 values (smoothing values)

Cluster Analysis

Perform clustering on attributes values and replace all values in the cluster by a cluster representative



Data Integration

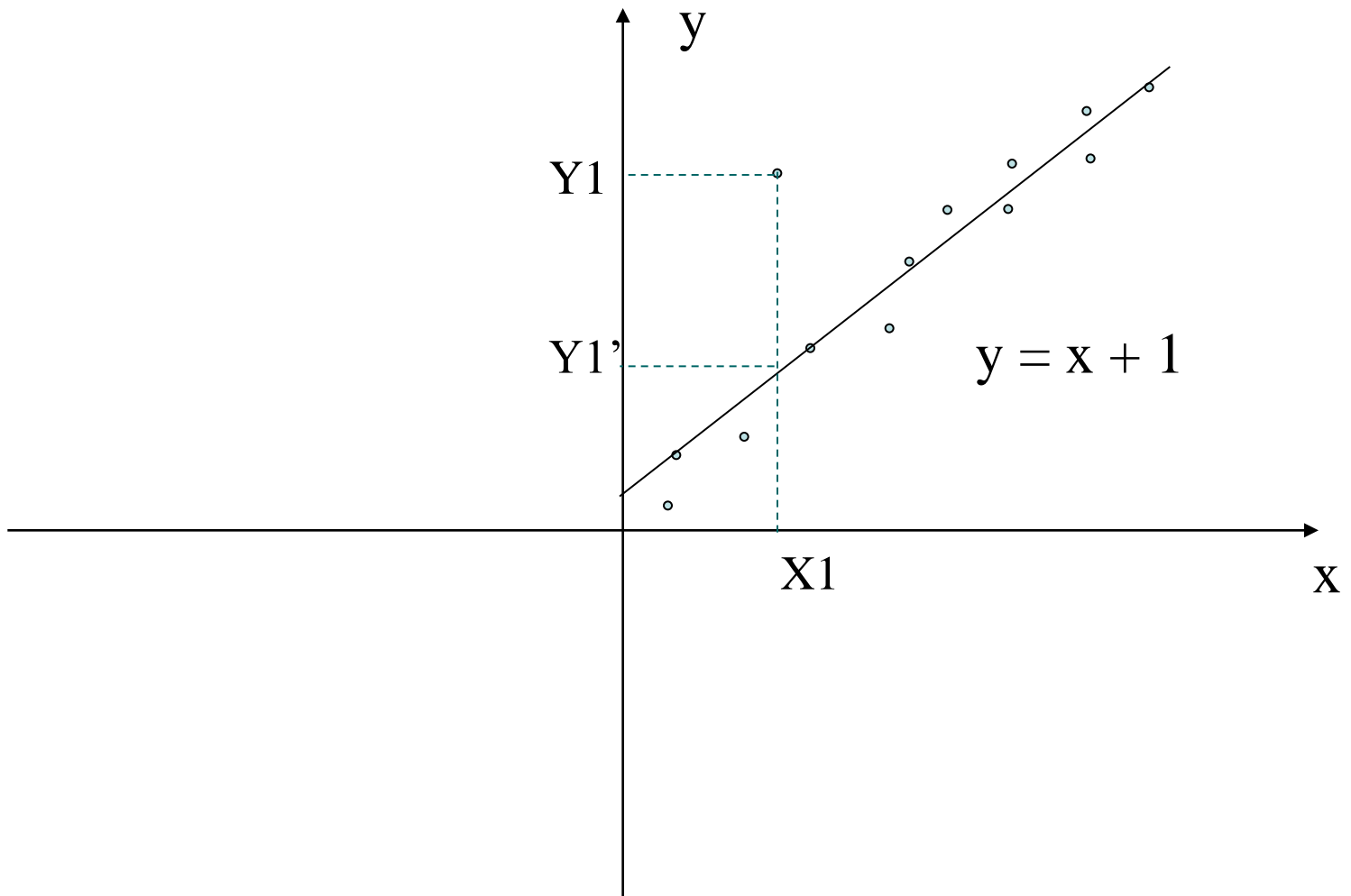
- **Data integration:**
 - combines data from multiple sources into a coherent store
- **Schema integration**
 - integrate metadata from different sources
 - Entity identification problem: identify real world entities from multiple data sources, e.g., A.cust-id \equiv B.cust-#
- **Detecting and resolving data value conflicts**
 - for the same real world entity, attribute values from different sources are different
 - possible reasons: different representations, different scales, e.g., metric vs. British units

Regression and Log-Linear Models

- **Linear regression:** Data are modeled to fit a straight line
 - Often uses the least-square method to fit the line
- **Multiple regression:** allows a response variable Y to be modeled as a linear function of multidimensional feature vector
- **Log-linear model:** approximates discrete multidimensional probability distributions

Linear Regression

Use regression analysis on values of attributes to fill missing values.

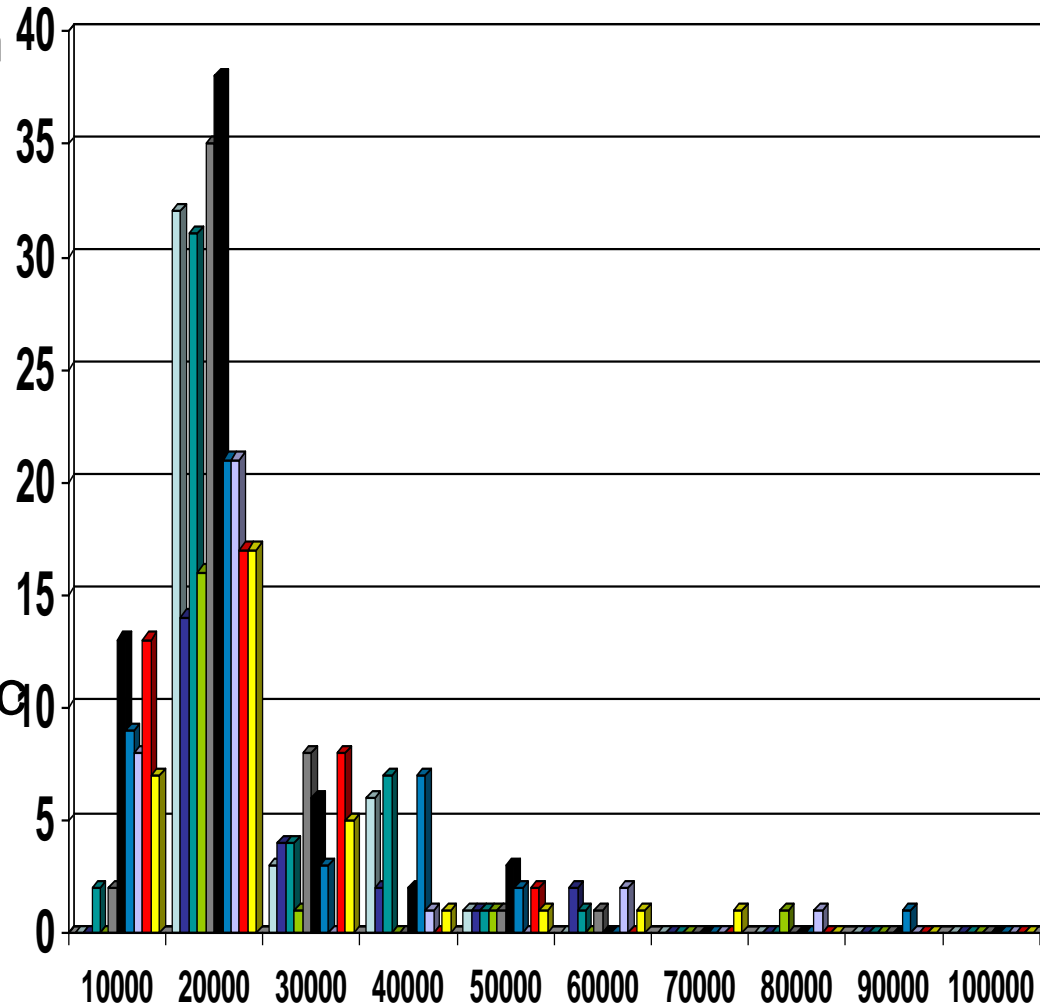


Regression and Log-Linear Models

- Linear regression: $Y = \alpha + \beta X$
 - Two parameters, α and β specify the line and are to be estimated by using the data at hand.
 - using the least squares criterion to the known values of $Y_1, Y_2, \dots, X_1, X_2, \dots$
- Multiple regression: $Y = b_0 + b_1 X_1 + b_2 X_2$.
 - Many nonlinear functions can be transformed into the above.
- Log-linear models:
 - The multi-way table of joint probabilities is approximated by a product of lower-order tables.
 - Probability: $p(a, b, c, d) = \alpha_{ab} \beta_{ac} \chi_{ad} \delta_{bcd}$

Histograms

- A popular data reduction technique
- Divide data (values of a given attribute) into buckets and store average (sum) for each bucket
- Can be constructed optimally in one dimension using dynamic programming
- Related to quantization problems.



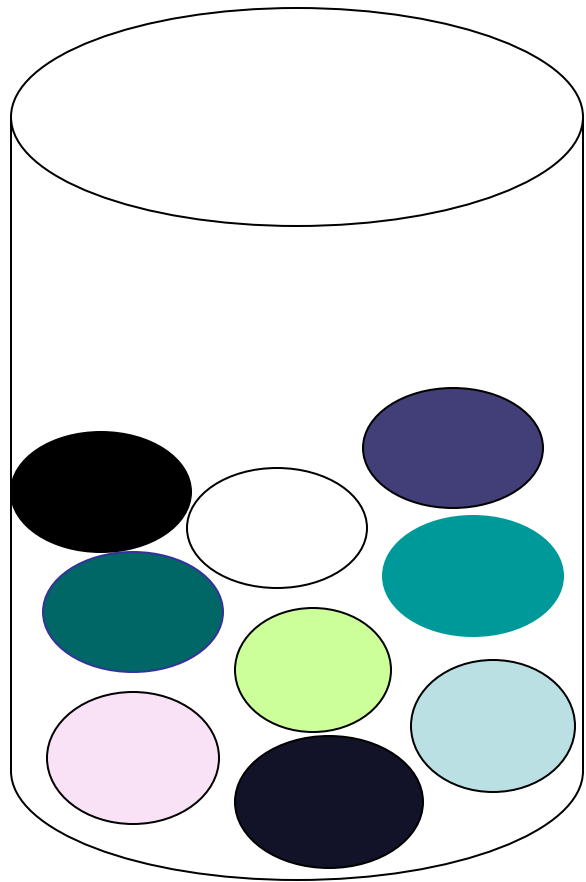
Clustering

- **Partition data set** (it means the values of an attribute in case of preprocessing) **into clusters**, and one can store cluster representation only, i.e. replace all values of the cluster by the one value representing this cluster.
- We also can use hierarchical clustering that can be stored in multi-dimensional index tree structures
- There are many choices of clustering definitions and clustering algorithms (some in Lecture notes on Clustering)

Sampling

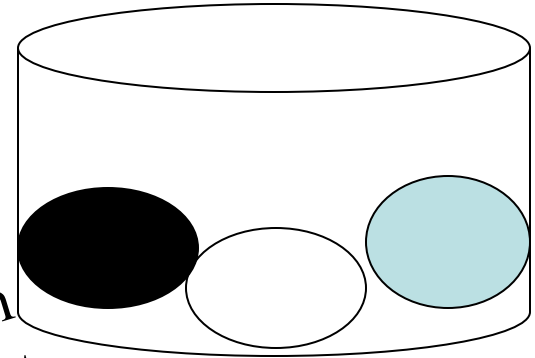
- Sampling allows the learning algorithm to run in complexity that is potentially sub-linear to the size of the data
- **Sampling** is a method of **choosing a representative** subset of the data
 - Simple random sampling may have very poor performance in the presence of skew data
- There are adaptive sampling methods
 - **Stratified sampling:**
 - Approximate the percentage of each class (or subpopulation of interest) in the overall database
 - Used in conjunction with skewed data

Sampling

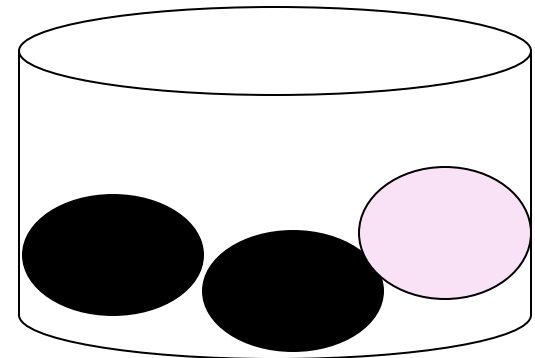


Raw Data

SRSWOR
(simple random
sample without
replacement)



SRSWR



Discretization

- Three types of attributes:
 - **Nominal** — values from an unordered set
 - **Ordinal** — values from an ordered set
 - **Continuous** — real numbers
- **Discretization:** divide the range (values) of a continuous attribute into intervals and replace all values in the interval by one representative (can be a non-numerical)
 - We use discretization to reduce data (attributes values) size
 - Prepare for further analysis
 - Some classification algorithms only accept categorical (non-numerical) attributes.

Discretization and Concept Hierachy

- Discretization
 - reduce the number of values for a given continuous attribute by dividing the range of the attribute (values of the attribute) into intervals.
 - Interval labels are then used to replace actual data values.
- Concept hierarchies
 - reduce the data by collecting and replacing low level concepts (such as numeric values for the attribute age) by higher level concepts (such as young, middle-aged, or senior).

Discretization and concept hierarchy generation for numeric data

- Discretization:
- Binning (see slides before)
- Histogram analysis (see slides before)
- Clustering analysis (see slides before)
- Segmentation by natural partitioning

Segmentation by natural partitioning

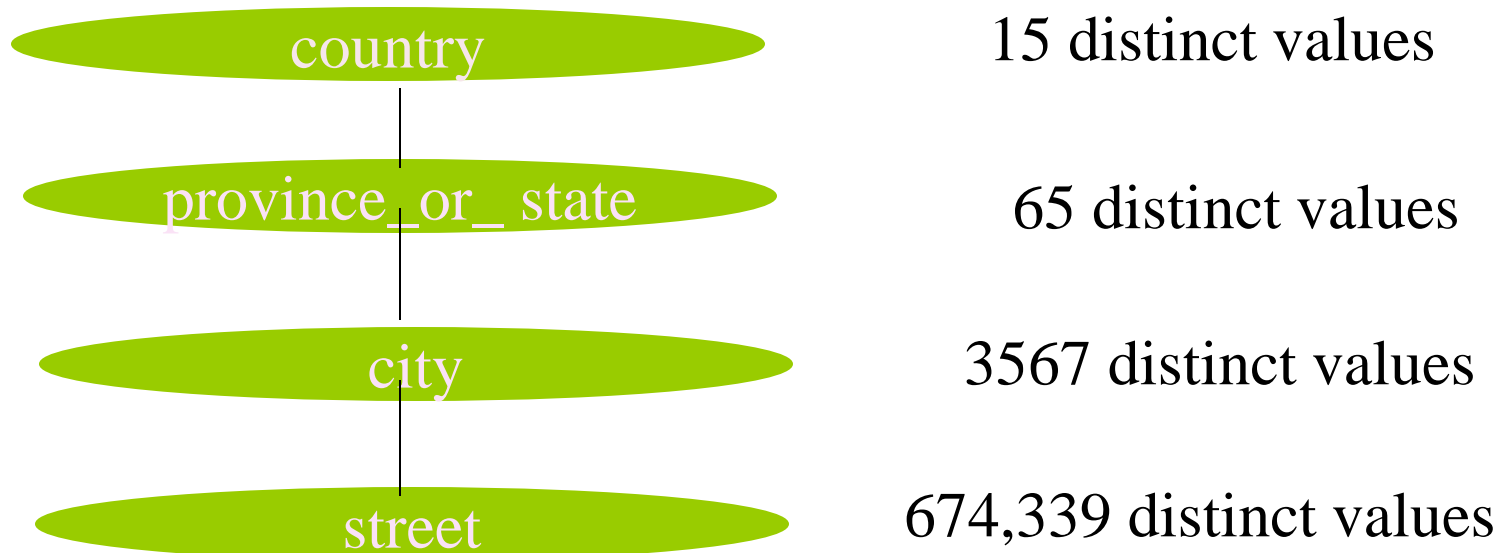
- 3-4-5 rule can be used to segment numeric data (attribute values) into relatively uniform, “natural” intervals.
- If an interval covers 3, 6, 7 or 9 ... $(3n)$ distinct values at the most significant digit, partition the range into 3 equi-width intervals
 - If it covers 2, 4, or 8 ... $(2n)$ distinct values at the most significant digit, partition the range into 4 intervals
 - If it covers 1, 5, or 10 ... $(5n)$ distinct values at the most significant digit, partition the range into 5 intervals

Concept Hierarchy generation for categorical data

- **Concept hierarchy is:**
- Specification of a partial ordering of attributes explicitly at the schema level by users or experts
- Specification of a portion of a hierarchy by explicit data grouping
- Specification of a set of attributes, but not of their partial ordering
- Specification of only a partial set of attributes

Specification of a set of attributes

Concept hierarchy can be automatically generated based on the number of distinct values per attribute in the given attribute set. The attribute with the most distinct values is placed at the lowest level of the hierarchy.



Summary

- Data preparation and preprocessing is a big issue for learning, data mining
- Data preprocessing includes
 - Data cleaning and data integration
 - Data reduction and attributes selection
 - Discretization
- A lot a methods have been developed but still an active area of research