Chapter 3: Data Preprocessing

Dong-Kyu Chae

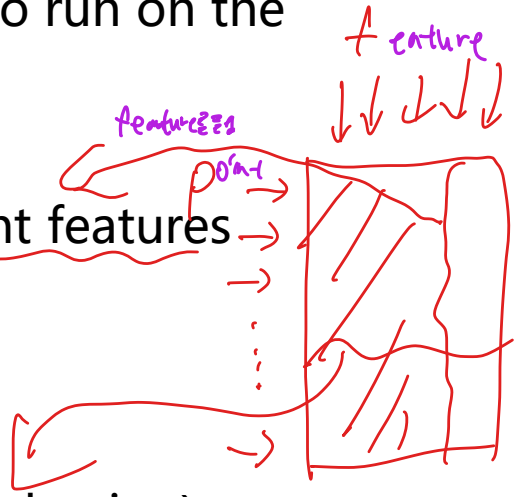
PI of the Data Intelligence Lab @HYU
Department of Computer Science & Data Science
Hanyang University

Data
Intelligence
LAB

Contents

Data Reduction Strategies

- **Data reduction:** Obtain a reduced representation of the data
  - Goal: to make data size **much smaller** in volume, but produce **almost the same** data mining results
- **Why data reduction?**
  - A database/data warehouse may store terabytes of data
  - Complex data analysis may take a very long time to run on the complete data set
- **Data reduction strategies**
  - Dimensionality reduction, e.g., remove unimportant features
    - Principal Components Analysis (PCA)
    - Feature selection via correlation analysis



  - Numerosity reduction (some simply call it: Data Reduction)
    - Replace data objects with a model that well-represents them
    - Regression, clustering, sampling

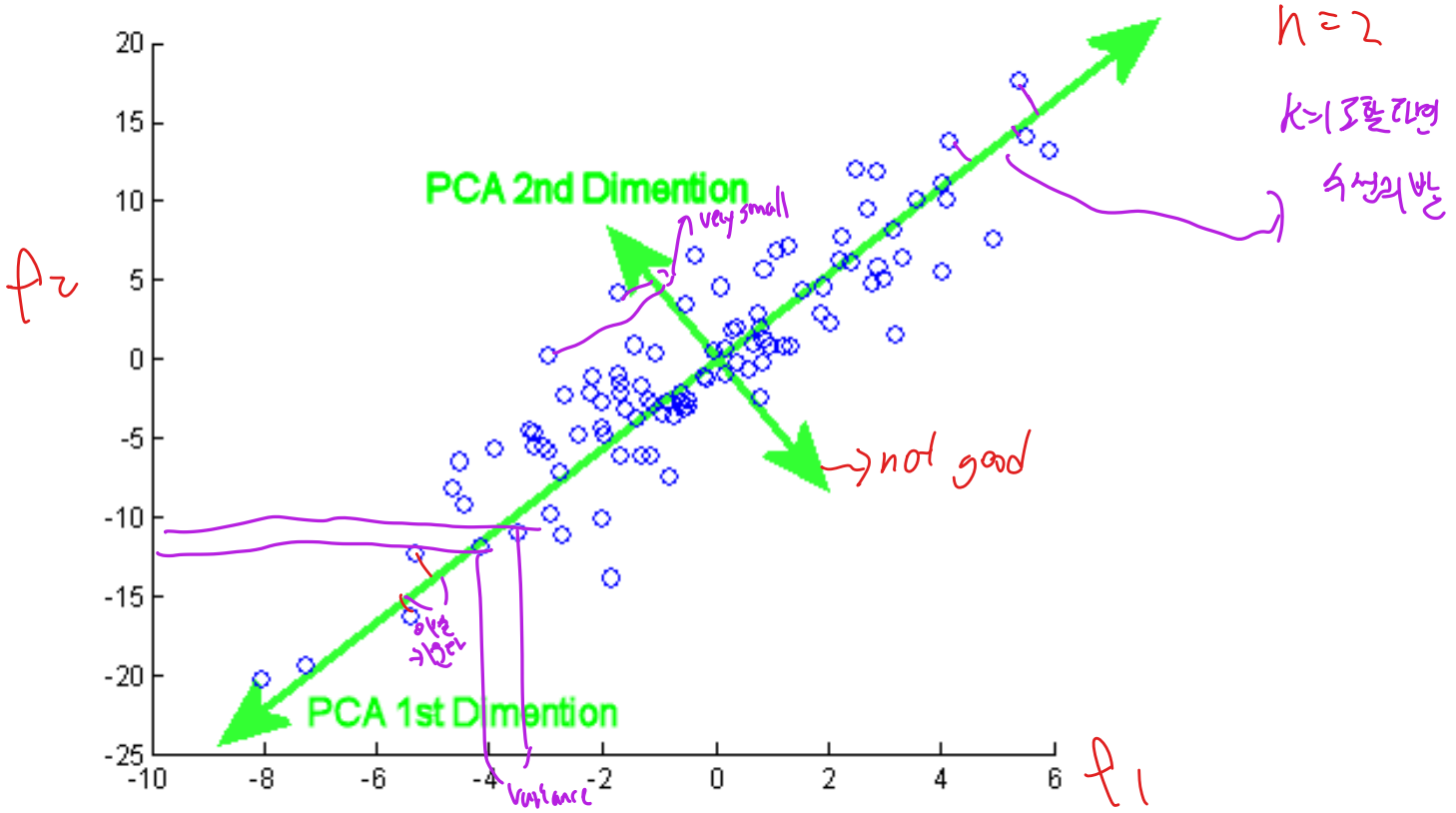
Dimensionality Reduction

- **Benefits**
  - Avoid the curse of dimensionality
  - Help eliminate irrelevant features and reduce noise
  - Reduce time and space required in data mining
  - Allow easier visualization

- **Dimensionality reduction techniques**
  - Mathematically
    - Principal Component Analysis (PCA)
  - Heuristically
    - Feature subset selection via correlation analysis
    - Feature creation

Principal Component Analysis (PCA)

- **Find new dimension(s) that result in the largest amount of variation in given data**
  - variance among data
- **Original data are projected onto a much smaller space**
  - Resulting in dimensionality reduction



Principal Component Analysis (PCA)

- **Given *N* data points from *n*-dimensions, find *k* ≤ *n* orthogonal (unit) vectors (*principal components*) that can be best used to represent data**
- **Step by step (skip, learned from the linear algebra class)**
  - Normalize input data: Each feature falls within the same range
  - Compute *k* orthonormal (unit) vectors, i.e., *principal components*
  - The principal components are sorted in order of decreasing "significance" or strength
  - The size of the data can be reduced by eliminating the *weak components*, i.e., those with low strength
    - Using the strong principal components, it is possible to **reconstruct** given data, which is a **good approximation** of the original data
- **Works for numeric data only**

Feature Selection

$n \rightarrow k$ $(k \leq n)$

- **Another way to reduce dimensionality of data**
- **Remove redundant features**
  - **Price** of a product and the VAT of a product
  - They tend to have **high correlation** with each other
- **Remove irrelevant features to the target problem**
  - Contain no information that is useful for the data mining task at hand
  - E.g., students' ID is often irrelevant to the task of predicting students' GPA

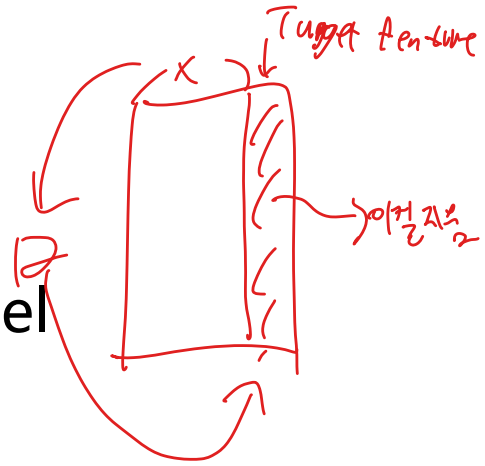
select feature subset

Feature Selection

- **Goal: among n features, find optimal d features**
  - Full search: There are 2^{d-1} possible feature combinations of d features to examine (**infeasible**!)
- **Approximation: heuristic feature selection methods**
  - 1. Best feature set under the feature independence assumption: choose d features one by one, by significance tests on **a validation set**
  - 2. Best step-wise feature selection:
    - The best single-feature is picked first
    - Then next best feature condition to the first, ...
  - 3. Step-wise feature elimination:
    - Repeatedly eliminate the worst feature

Numerosity Reduction

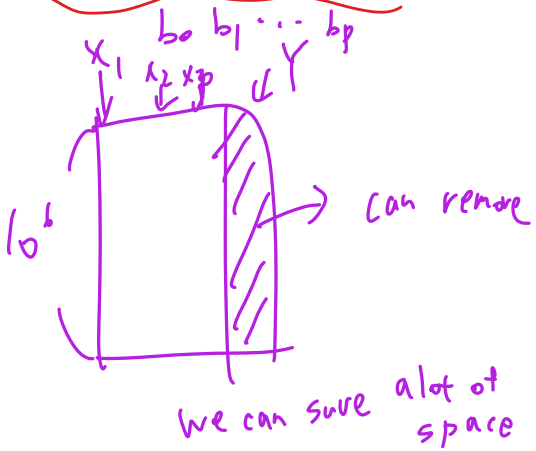
- **Reduce data volume by choosing alternative, *smaller forms* of data representation**
- **Parametric methods (e.g., regression)**
  - Assume some features' values fit a model
  - Estimate the model parameters
  - Store only the model parameters, and discard all the feature values (except possible outliers)
- **Non-parametric methods**
  - Do not assume any model parameters
  - Clustering, sampling, …



Parametric Method with a Regression Model

- **Linear regression:** $Y = b_1 X + b_0$
  - Two regression coefficients, b_0 and b_1, specify the line. They are to be estimated via optimization task
- **Multiple regression:** $Y = b_0 + b_1 X_1 + b_2 X_2 + \ldots + b_p X_p$
  - Linear function involving multiple features (they are assumed to be independent)

- **All the model parameters $b_0\ b_1 \ldots b_p$ *are stored instead of data Y***



Clustering

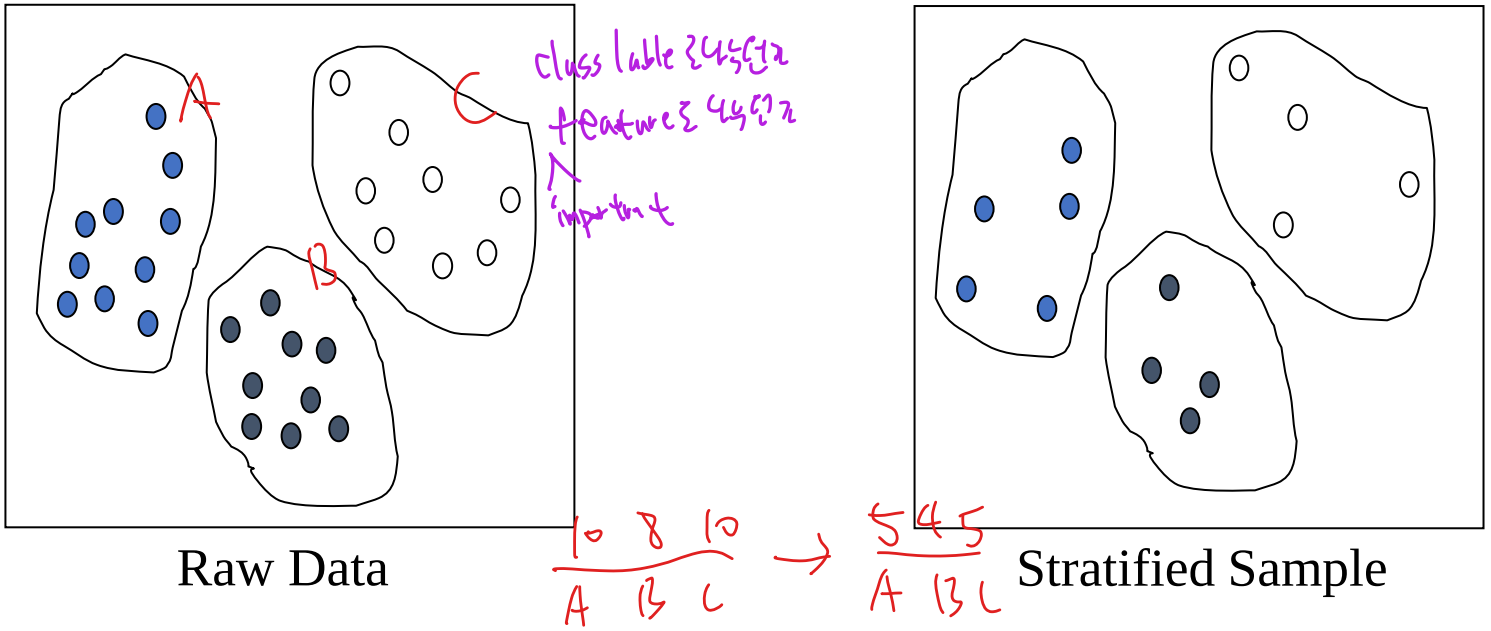
- Partition data set into clusters based on similarity
- Then, store cluster representation (e.g., centroids and diameter) only
- Can be very effective if data is clustered but not if data is "smeared"
- Can have hierarchical clustering and be stored in multi-dimensional index tree structures
- There are many choices of clustering definitions and clustering algorithms
  - Cluster analysis will be studied in the next several lectures.

Sampling

- **Sampling:** obtaining a small set of samples S to represent the whole data set N
- **Allow a mining algorithm to run in complexity that is potentially sub-linear to the size of the data**
- **How to choose a representative subset of the data?**
  - Simple random sampling may have poor performance
  - Develop adaptive sampling methods, e.g., stratified sampling

Stratified Sampling

- **Stratified sampling:**
  - Partition the data set, and draw samples from each partition **proportionally**
    - Approximately the same percentage of the data
  - Used when sampling small number of objects from skewed dataset



Contents

Data Transformation

- **Maps the values of a given feature into the new values within a specific range, or new categories**
- **Methods**
  - **Normalization**: Scaled to fall within a smaller, specified range
    - Min-max normalization
    - Z-score normalization
    - Normalization by decimal scaling
  - **Discretization**: fall within a discete category
    - Binning

Z-Score Normalization

- **Why normalization?**
  - If scale is different, proximity among data may not be computed meaningfully.

Age 1~100

Salary 50,000 ~

- **Z-score:**
  - x: raw score to be standardized, μ: mean of the population, σ: standard deviation

$$z = \frac{x - \mu}{\sigma}$$

  - Meaning: the distance between the raw score and the population mean in units of the standard deviation
    - "–" when the raw score is below the mean
    - "+" when the raw score is above the mean

Normalization Examples

- **Min-max normalization: to [new_min$_A$, new_max$_A$]**

$$v' = \frac{v - min_A}{max_A - min_A}(new\_max_A - new\_min_A) + new\_min_A$$

  - Ex. Let income that ranges from \$12,000 to \$98,000, normalized to [0.0, 1.0]. Then \$73,000 is mapped to: $\frac{73{,}600 - 12{,}000}{98{,}000 - 12{,}000}(1.0 - 0) + 0 = 0.716$

- **Z-score normalization (μ: mean, σ: standard deviation):**

$$v' = \frac{v - \mu_A}{\sigma_A}$$

  - Ex. Let μ = 54,000, σ = 16,000. Then $\frac{73{,}600 - 54{,}000}{16{,}000} = 1.225$

- **Normalization by decimal scaling**

  - $v' = \frac{v}{10^j}$ where j is the smallest integer such that any Max(|v'|) < 1

Discretization

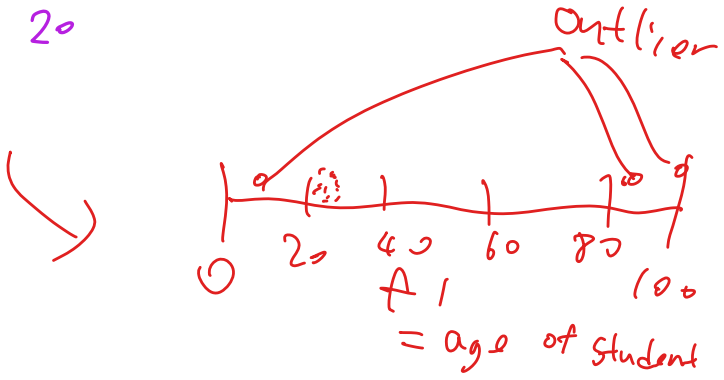
- **Three types of features**
  - Nominal—values from an unordered set, e.g., color, job, …
  - Ordinal—values from an ordered set, e.g., size
  - Numeric—continuous real numbers, e.g., integer or real numbers

- **Discretization: divide the range of a numeric (continuous) feature into intervals**
  - Labels are assigned to intervals to replace actual data values
  - After discretization, similar values become identical
  - Used for further analysis, e.g., classification

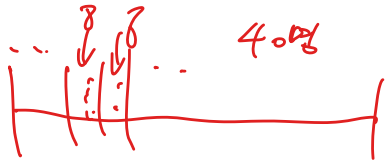
Simple Discretization: Binning

Equal-width (distance) partitioning

- Divides the range into *N* intervals of equal size: uniform grid
- if *A* and *B* are the lowest and highest values of the feature, the width of intervals will be: $W = (B - A)/N$.
- The most straightforward
- Problems
  - Outliers may dominate presentation
  - Skewed data is not handled well



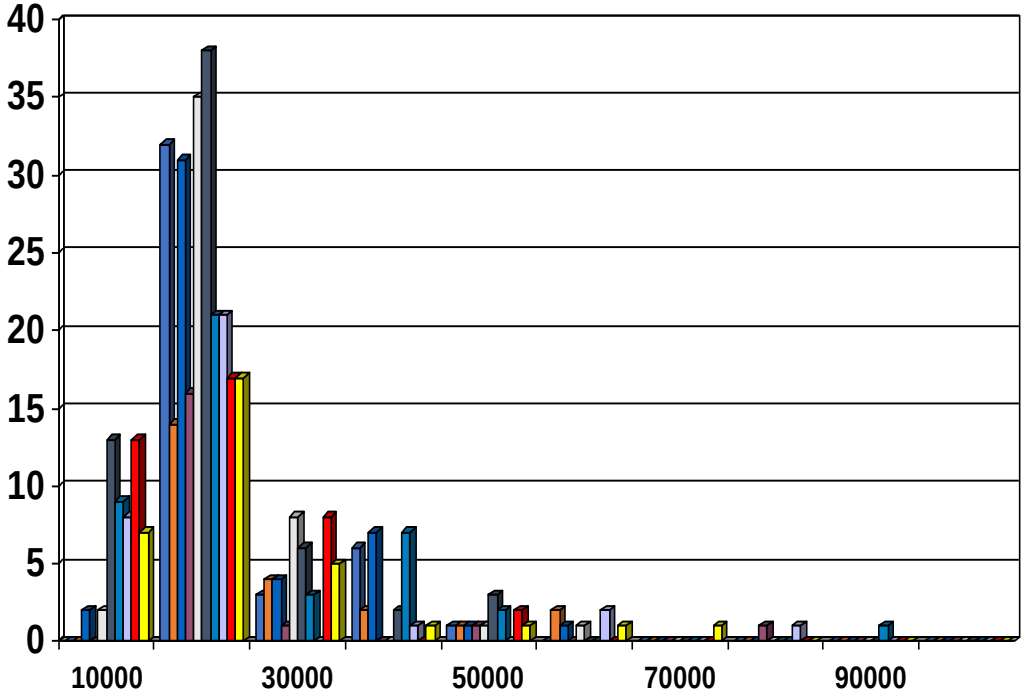
Equal-depth (frequency) partitioning



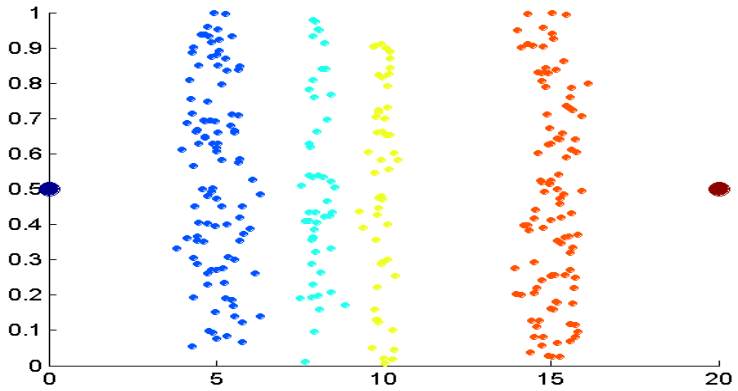
- Divides the range into *N* intervals, each containing approximately same number of samples

Histogram Analysis of Numerical Features

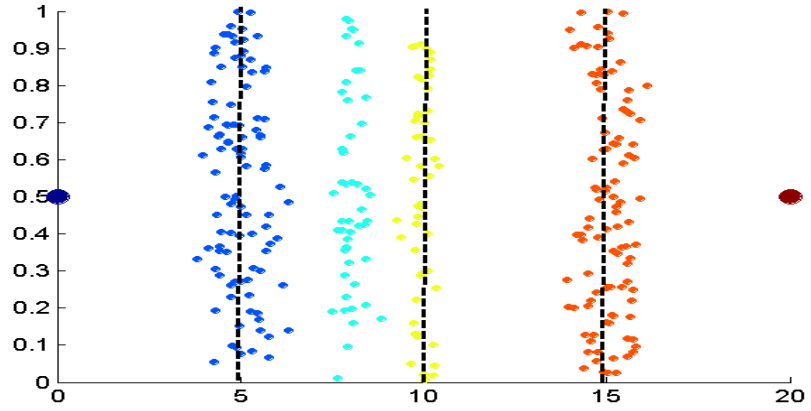
- **Bining methods:**
  - Equal-width
    - Equal bucket range
  - Equal-frequency
    - (or equal-depth)
    - Equal depth for buckets



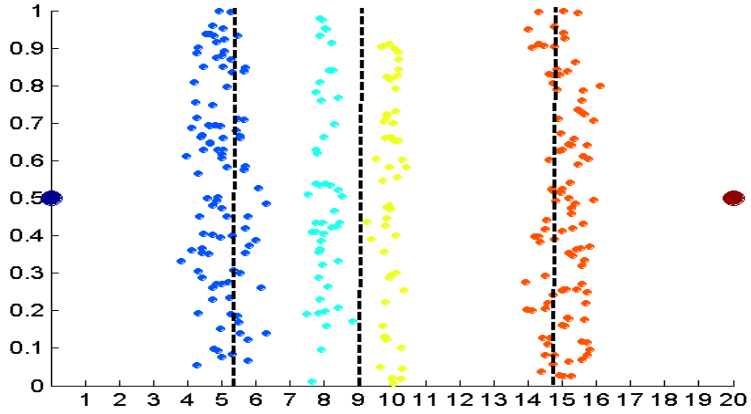
Limitation of Binning



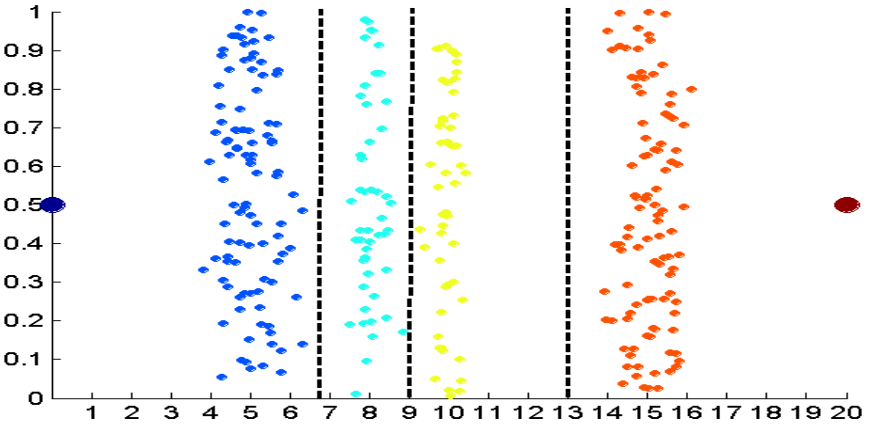
data



Equal width (binning)



Equal frequency (binning)



clustering leads to better results

Summary

- **Why data pre-processing?:** accuracy, completeness, consistency, timeliness, believability, interpretability
- **Data cleaning:** to handle missing/noisy values, outliers
- **Data integration from multiple sources:**
  - Remove redundancies via correlation analysis
  - Detect inconsistencies
- **Data reduction**
  - Dimensionality reduction
  - Numerosity reduction
- **Data transformation**
  - Discretization
  - Normalization

Thank You

Data
Intelligence
Lab