# Data Mining:
# Concepts and Techniques

## — Chapter 2 —

Jiawei Han

Department of Computer Science

University of Illinois at Urbana-Champaign

[www.cs.uiuc.edu/~hanj](www.cs.uiuc.edu/~hanj)

# Chapter 2: Data Preprocessing

- **Why preprocess the data?**

- Descriptive data summarization

- Data cleaning

- Data integration and transformation

- Data reduction

- Discretization and concept hierarchy generation

- Summary

# Why Data Preprocessing?

- Data in the real world is dirty
  - incomplete: lacking attribute values, lacking certain attributes of interest, or containing only aggregate data
    - e.g., occupation=" "
  - noisy: containing errors or outliers
    - e.g., Salary="-10"
  - inconsistent: containing discrepancies in codes or names
    - e.g., Age="42" Birthday="03/07/1997"
    - e.g., Was rating "1,2,3", now rating "A, B, C"
    - e.g., discrepancy between duplicate records

# Why Is Data Dirty?

- Incomplete data may come from
  - "Not applicable" data value when collected
  - Different considerations between the time when the data was collected and when it is analyzed.
  - Human/hardware/software problems
- Noisy data (incorrect values) may come from
  - Faulty data collection instruments
  - Human or computer error at data entry
  - Errors in data transmission
- Inconsistent data may come from
  - Different data sources
  - Functional dependency violation (e.g., modify some linked data)
- Duplicate records also need data cleaning

# Why Is Data Preprocessing Important?

- No quality data, no quality mining results!
  - Quality decisions must be based on quality data
    - e.g., duplicate or missing data may cause incorrect or even misleading statistics.
  - Data warehouse needs consistent integration of quality data
- Data extraction, cleaning, and transformation comprises the majority of the work of building a data warehouse

# Multi-Dimensional Measure of Data Quality

- A well-accepted multidimensional view:
    - Accuracy
    - Completeness
    - Consistency
    - Timeliness
    - Believability
    - Value added
    - Interpretability
    - Accessibility
- Broad categories:
    - Intrinsic, contextual, representational, and accessibility

# Major Tasks in Data Preprocessing

- ## Data cleaning
    - Fill in missing values, smooth noisy data, identify or remove outliers, and resolve inconsistencies

- ## Data integration
    - Integration of multiple databases, data cubes, or files

- ## Data transformation
    - Normalization and aggregation

- ## 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
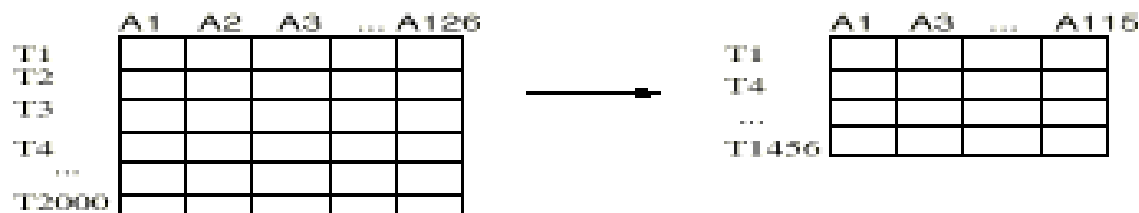
# Forms of Data Preprocessing

Data Cleaning

[water to clean dirty-looking data]

[show soap suds on data]

['clean'-looking data]
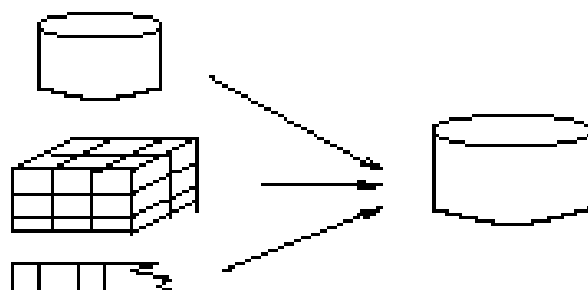
Data Integration

Data Transformation     -2, 32, 100, 59, 48     ⟶     -0.02, 0.32, 1.00, 0.59, 0.48

Data Reduction

| | A1 | A2 | A3 | ... A126 |
|---|---|---|---|---|
| T1 | | | | |
| T2 | | | | |
| T3 | | | | |
| T4 | | | | |
| ... | | | | |
| T2000 | | | | |

| | A1 | A3 | ... | A115 |
|---|---|---|---|---|
| T1 | | | | |
| T4 | | | | |
| ... | | | | |
| T1456 | | | | |

# Chapter 2: Data Preprocessing

- Why preprocess the data?

- Descriptive data summarization

- Data cleaning

- Data integration and transformation

- Data reduction

- Discretization and concept hierarchy generation

- Summary

# Mining Data Descriptive Characteristics

- **Motivation**
  - To better understand the data: central tendency, variation and spread
- **Data dispersion characteristics**
  - median, max, min, quantiles, outliers, variance, etc.
- **Numerical dimensions** correspond to sorted intervals
  - Data dispersion: analyzed with multiple granularities of precision
  - Boxplot or quantile analysis on sorted intervals
- **Dispersion analysis on computed measures**
  - Folding measures into numerical dimensions
  - Boxplot or quantile analysis on the transformed cube

# Measuring the Central Tendency

- **Mean (algebraic measure) (sample vs. population):** $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$     $\mu = \frac{\sum x}{N}$

    - Weighted arithmetic mean:

    - Trimmed mean: chopping extreme values     $\bar{x} = \dfrac{\sum_{i=1}^{n} w_i x_i}{\sum_{i=1}^{n} w_i}$

- <u>Median</u>: A holistic measure

    - Middle value if odd number of values, or average of the middle two values otherwise

    - Estimated by interpolation (for *grouped data*):

$$median = L_1 + \left(\frac{n/2 - (\sum f)l}{f_{median}}\right)c$$

- <u>Mode</u>

    - Value that occurs most frequently in the data

    - Unimodal, bimodal, trimodal

    - Empirical formula:     $mean - mode = 3 \times (mean - median)$

# Symmetric vs. Skewed Data

- Median, mean and mode of symmetric, positively and negatively skewed data
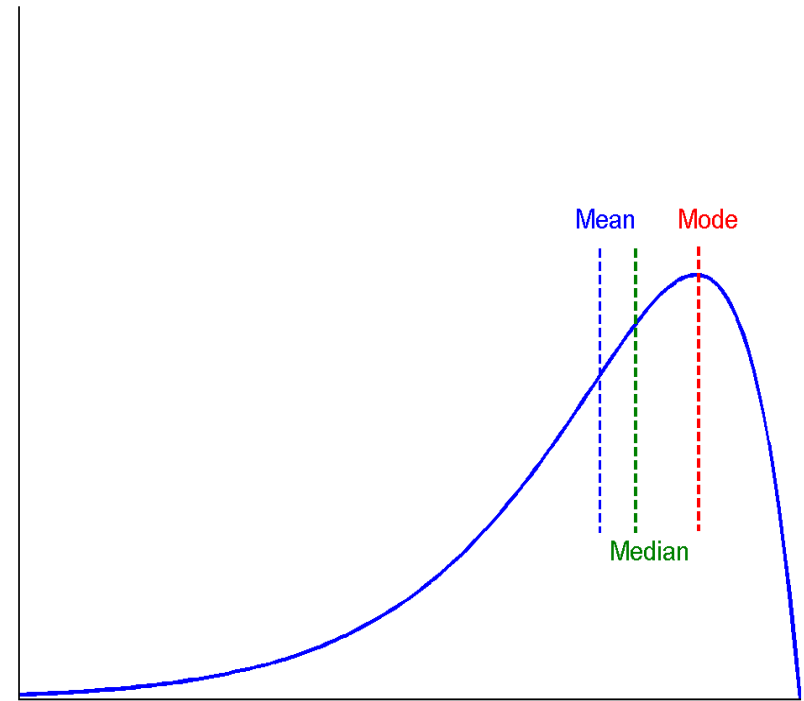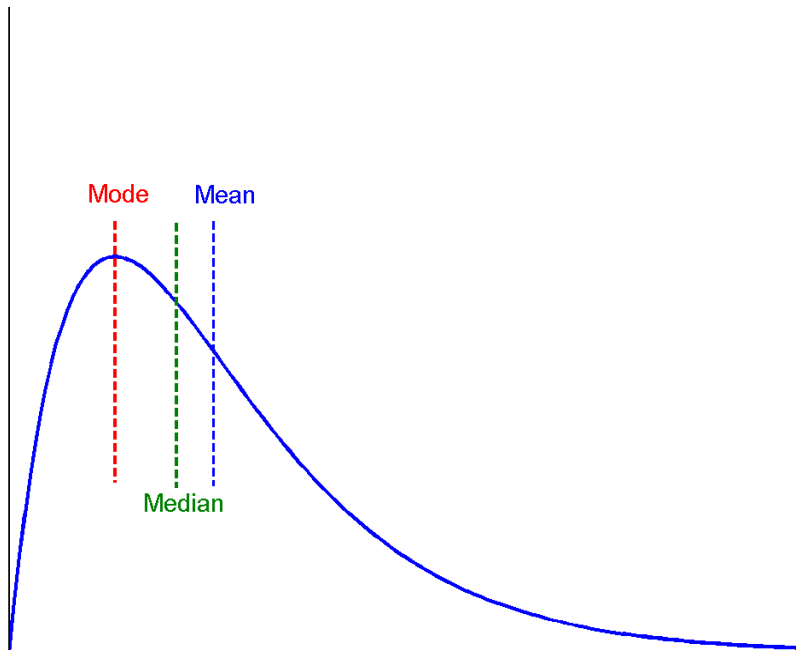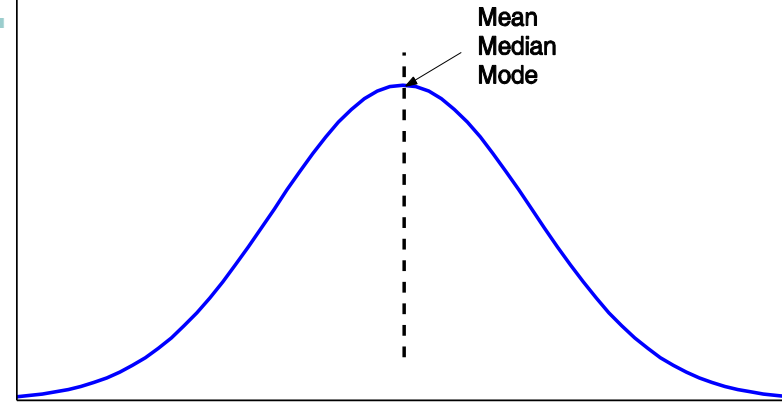
# Measuring the Dispersion of Data

- Quartiles, outliers and boxplots

    - Quartiles: $Q_1$ (25$^{th}$ percentile), $Q_3$ (75$^{th}$ percentile)

    - Inter-quartile range: IQR = $Q_3 - Q_1$

    - Five number summary: min, $Q_1$, M, $Q_3$, max

    - Boxplot: ends of the box are the quartiles, median is marked, whiskers, and plot outlier individually

    - Outlier: usually, a value higher/lower than 1.5 x IQR

- Variance and standard deviation (*sample: s, population: σ*)

    - Variance: (algebraic, scalable computation)

$$s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2 = \frac{1}{n-1}[\sum_{i=1}^{n}x_i^2 - \frac{1}{n}(\sum_{i=1}^{n}x_i)^2] \qquad \sigma^2 = \frac{1}{N}\sum_{i=1}^{n}(x_i - \mu)^2 = \frac{1}{N}\sum_{i=1}^{n}x_i^2 - \mu^2$$

    - Standard deviation *s (or σ)* is the square root of variance *s² (or σ²)*

# Properties of Normal Distribution Curve

- The normal (distribution) curve
  - From μ–σ to μ+σ: contains about 68% of the measurements (μ: mean, σ: standard deviation)
  - From μ–2σ to μ+2σ: contains about 95% of it
  - From μ–3σ to μ+3σ: contains about 99.7% of it
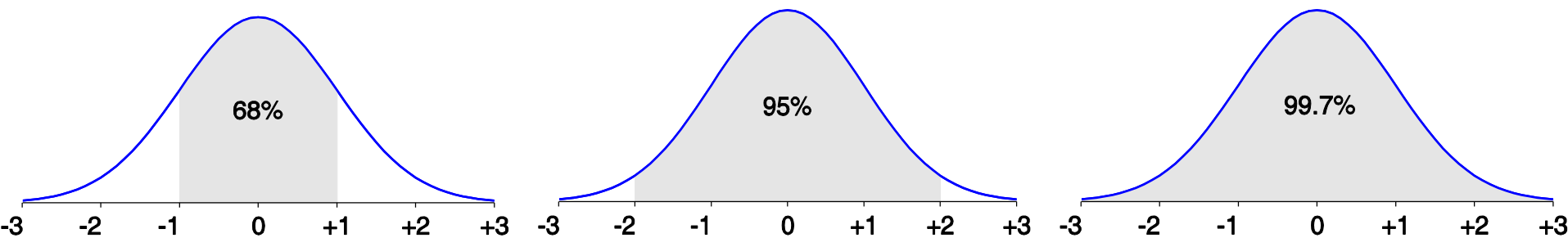
# Boxplot Analysis


unit price (S)
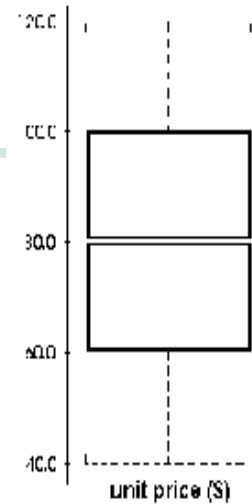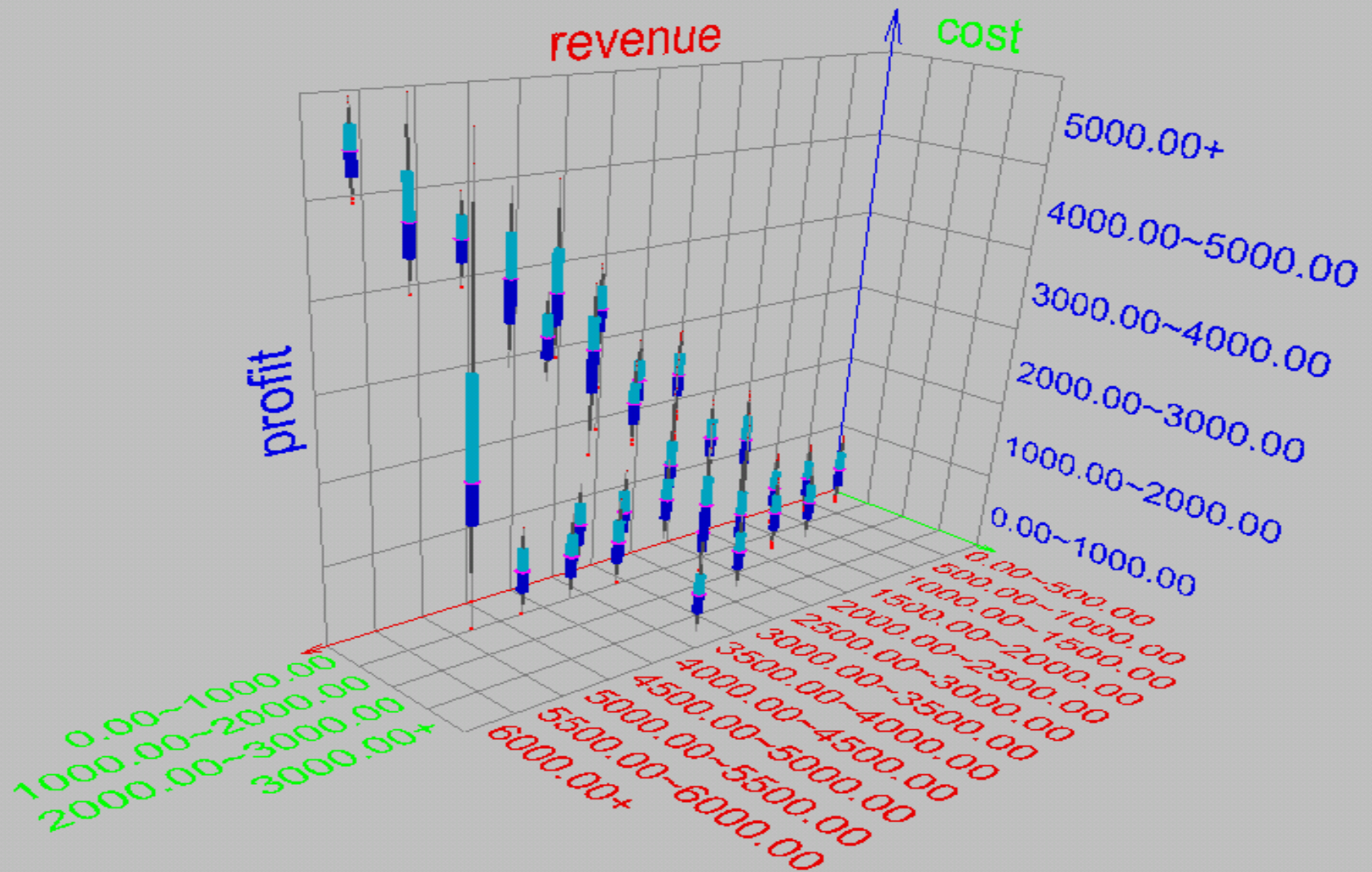
- **Five-number summary** of a distribution:

  Minimum, Q1, M, Q3, Maximum

- **Boxplot**

  - Data is represented with a box

  - The ends of the box are at the first and third quartiles, i.e., the height of the box is IRQ

  - The median is marked by a line within the box

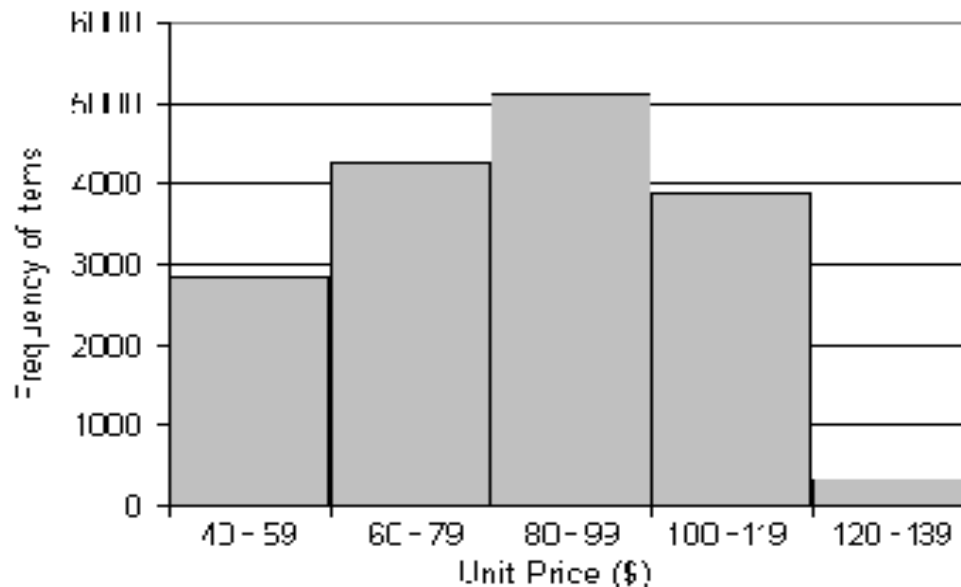  - Whiskers: two lines outside the box extend to Minimum and Maximum

# Visualization of Data Dispersion: Boxplot Analysis

# Histogram Analysis

- **Graph displays of basic statistical class descriptions**
  - **Frequency histograms**
    - A univariate graphical method
    - Consists of a set of rectangles that reflect the counts or frequencies of the classes present in the given data

# Quantile Plot

- Displays all of the data (allowing the user to assess both the overall behavior and unusual occurrences)
- Plots quantile information
    - For a data $x_i$ data sorted in increasing order, $f_i$ indicates that approximately 100 $f_i$% of the data are below or equal to the value $x_i$
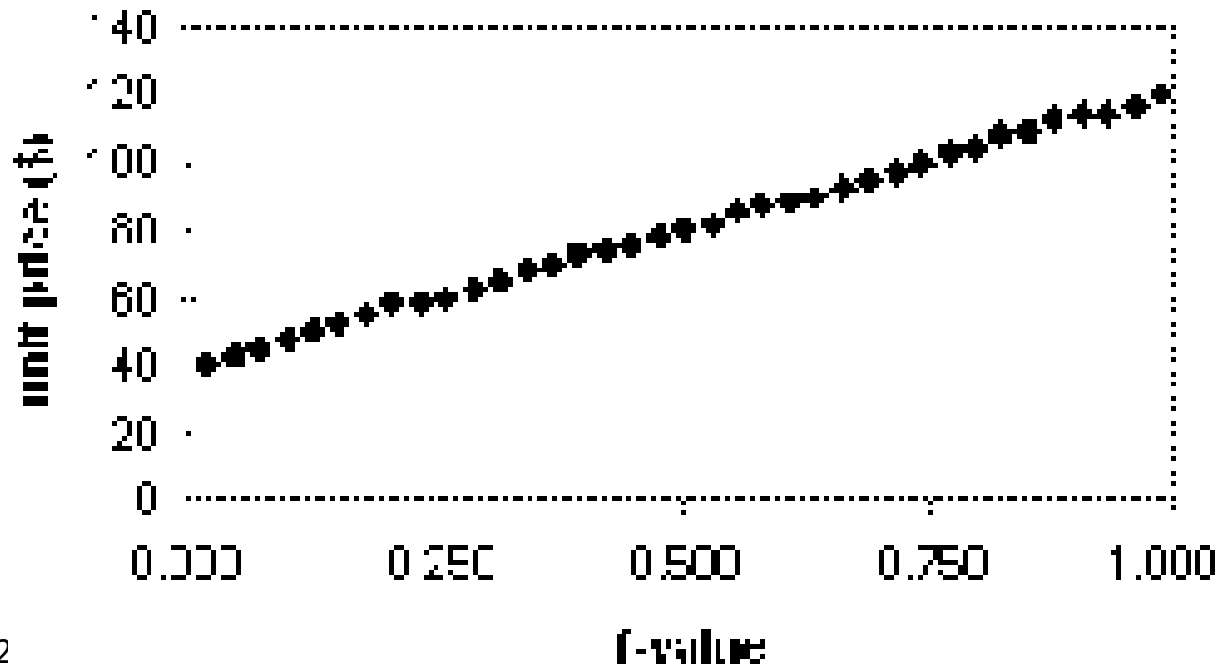
# Quantile-Quantile (Q-Q) Plot

- Graphs the quantiles of one univariate distribution against the corresponding quantiles of another

- Allows the user to view whether there is a shift in going from one distribution to another
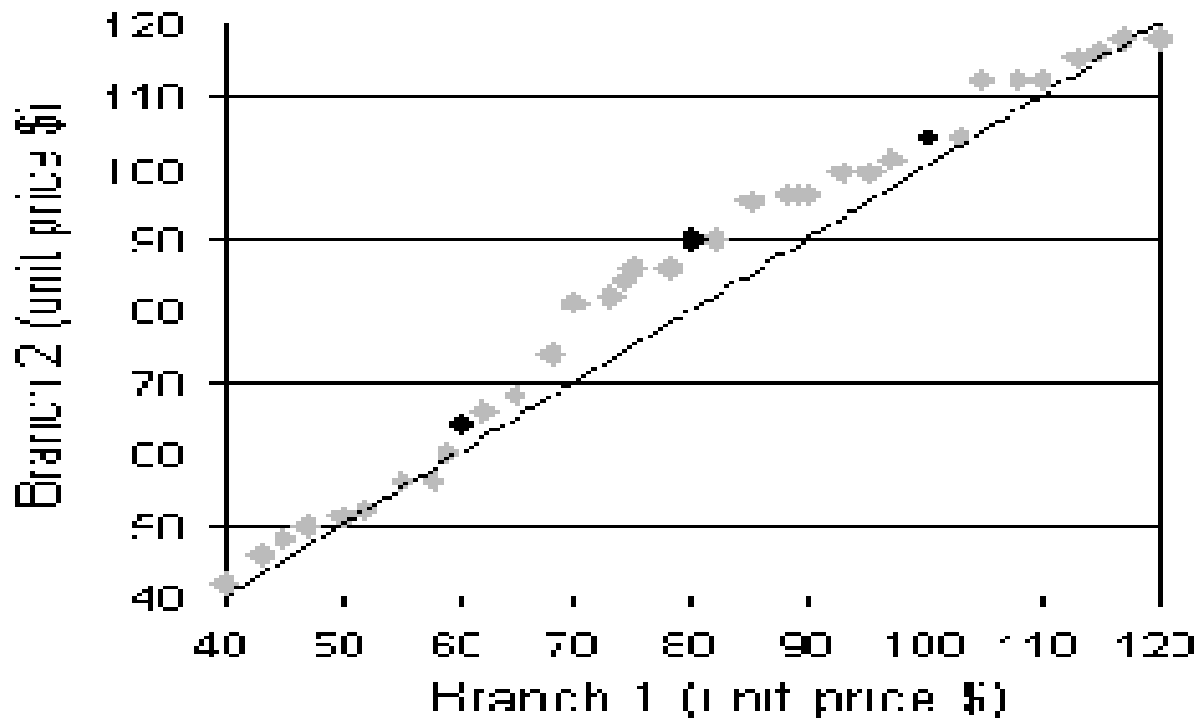
# Scatter plot

- Provides a first look at bivariate data to see clusters of points, outliers, etc
- Each pair of values is treated as a pair of coordinates and plotted as points in the plane

# Loess Curve

- Adds a smooth curve to a scatter plot in order to provide better perception of the pattern of dependence

- Loess curve is fitted by setting two parameters: a smoothing parameter, and the degree of the polynomials that are fitted by the regression
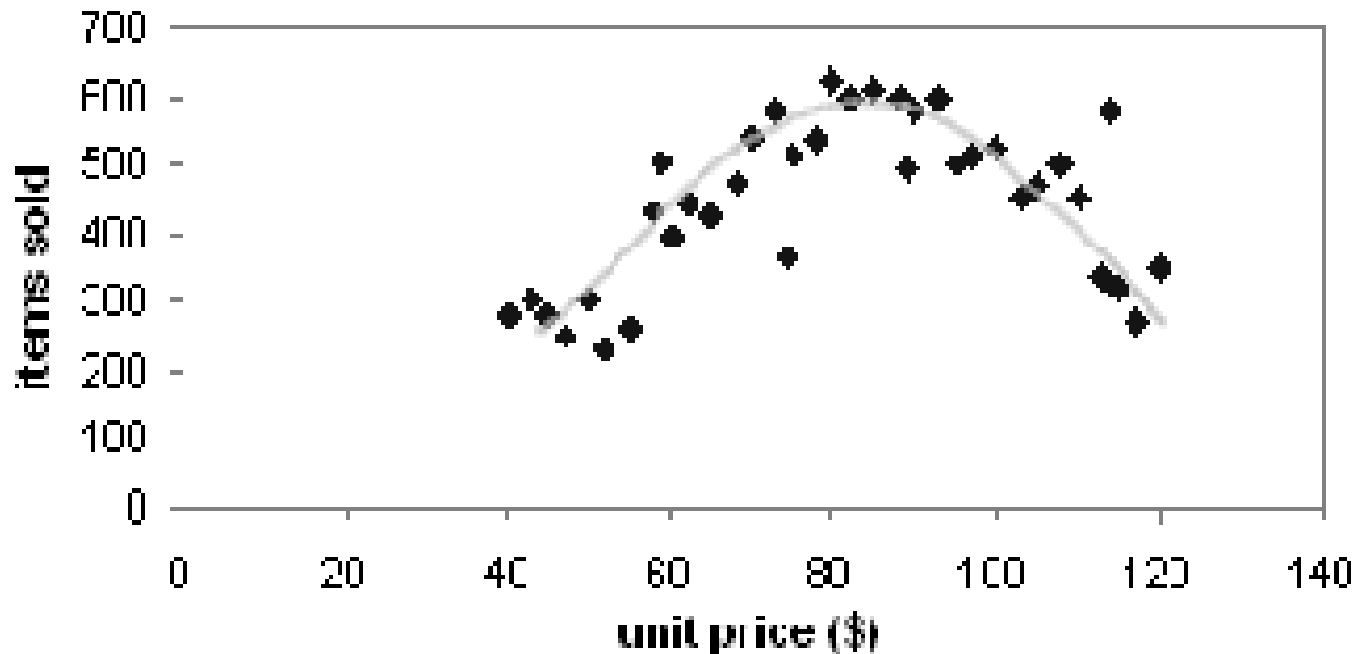
# Positively and Negatively Correlated Data

Data Mining: Concepts and Techniques
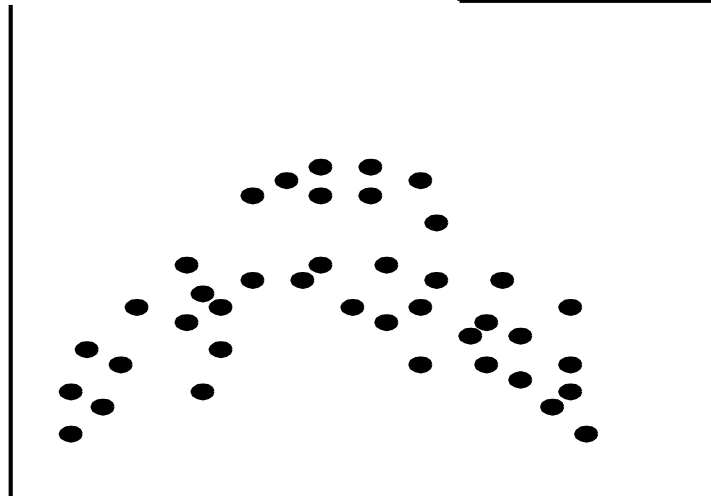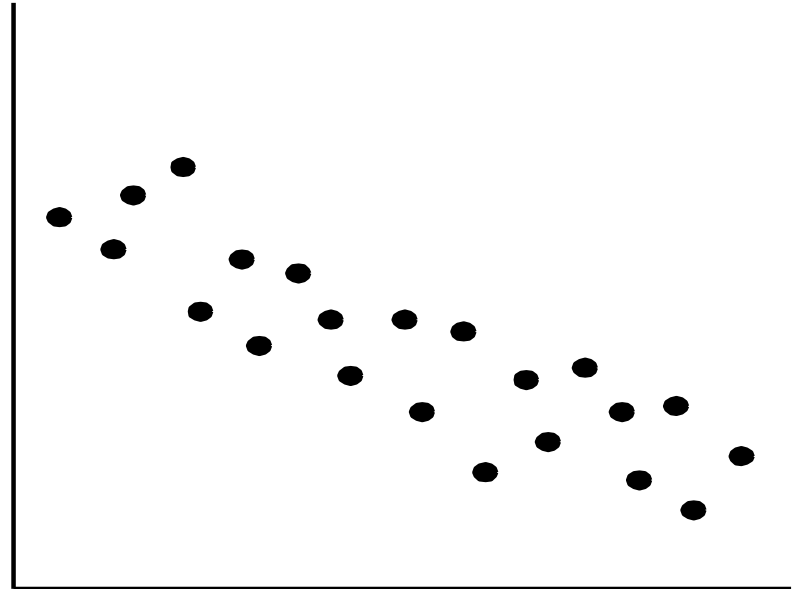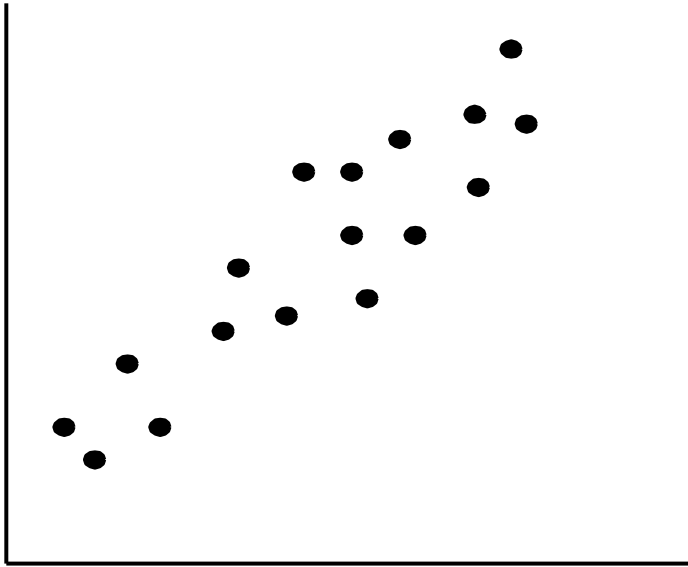
# Not Correlated Data
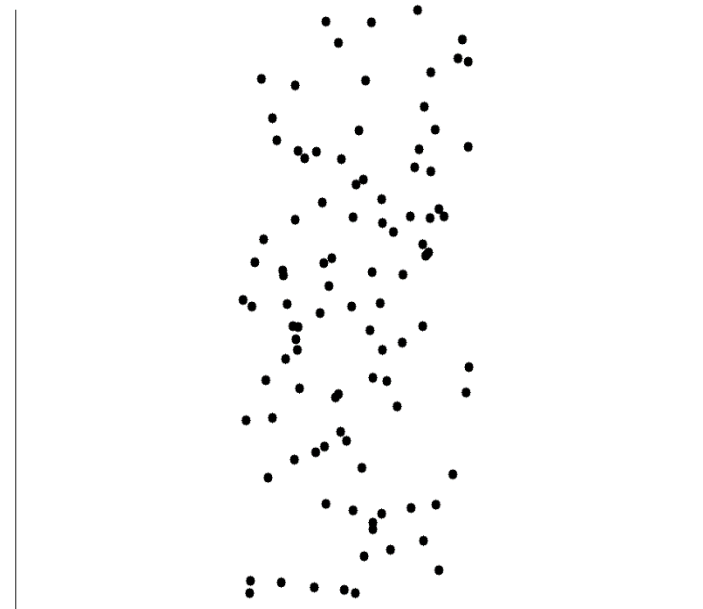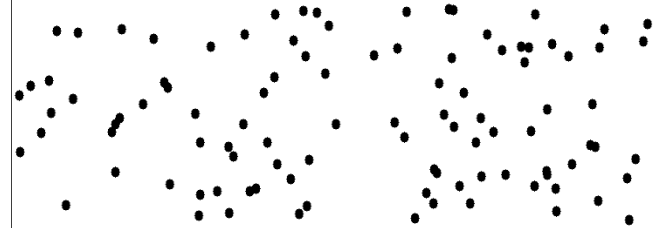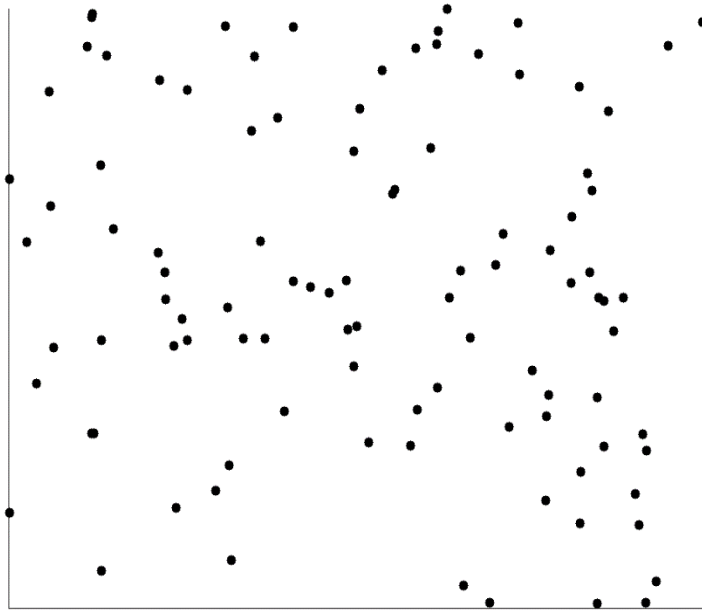
# Graphic Displays of Basic Statistical Descriptions

- Histogram: (shown before)
- Boxplot: (covered before)
- Quantile plot: each value $x_i$ is paired with $f_i$ indicating that approximately 100 $f_i$% of data are $\leq x_i$
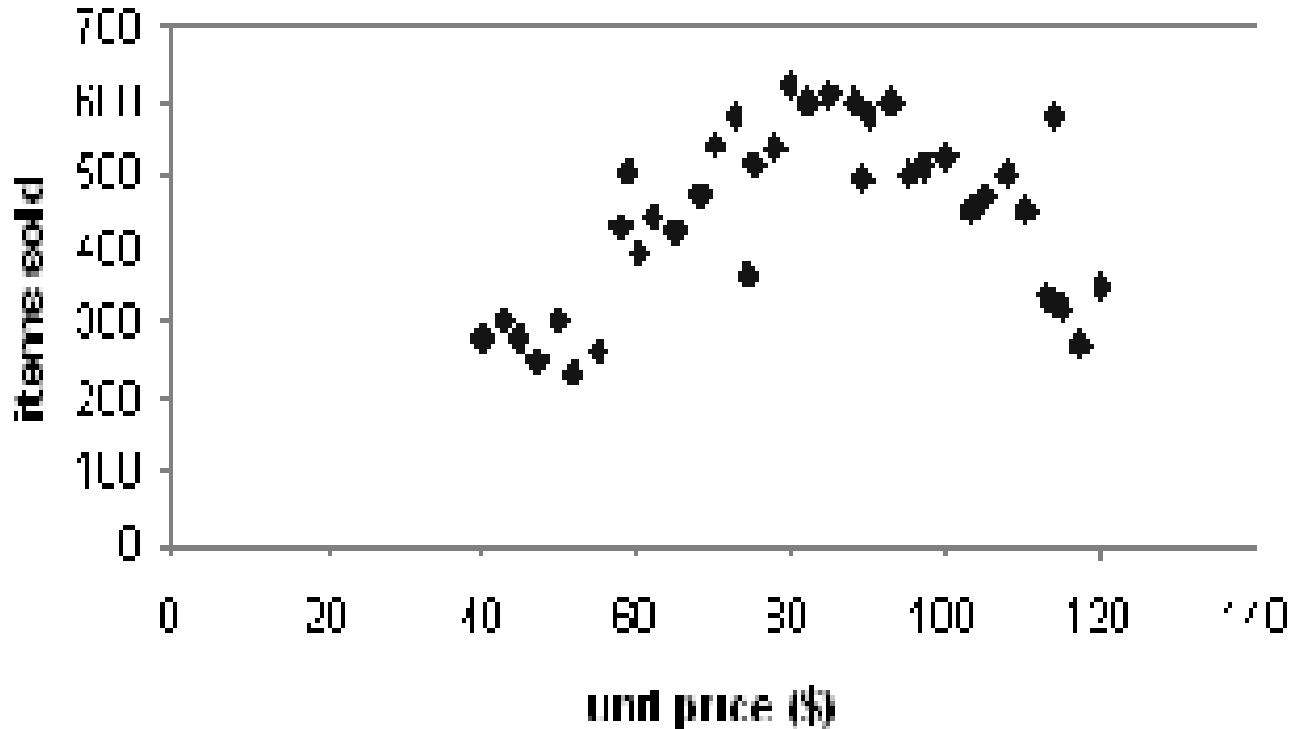- Quantile-quantile (q-q) plot: graphs the quantiles of one univariant distribution against the corresponding quantiles of another
- Scatter plot: each pair of values is a pair of coordinates and plotted as points in the plane
- Loess (local regression) curve: add a smooth curve to a scatter plot to provide better perception of the pattern of dependence

# Chapter 2: Data Preprocessing

- Why preprocess the data?

- Descriptive data summarization

- <span style="color:red">Data cleaning</span>

- Data integration and transformation

- Data reduction

- Discretization and concept hierarchy generation

- Summary

# Data Cleaning

- Importance
  - "Data cleaning is one of the three biggest problems in data warehousing"—Ralph Kimball
  - "Data cleaning is the number one problem in data warehousing"—DCI survey

- Data cleaning tasks

  - Fill in missing values

  - Identify outliers and smooth out noisy data

  - Correct inconsistent data

  - Resolve redundancy caused by data integration

# Missing Data

- Data is not always available

    - E.g., many tuples have no recorded value for several attributes, such as customer income in sales data

- 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?

- Ignore the tuple: usually done when class label is missing (assuming the tasks in classification—not effective when the percentage of missing values per attribute varies considerably.

- Fill in the missing value manually: tedious + infeasible?

- Fill in it automatically with

    - a global constant : e.g., "unknown", a new class?!

    - the attribute mean

    - the attribute mean for all samples belonging to the same class: smarter

    - the most probable value: inference-based such as Bayesian formula or decision tree

# Noisy Data

- Noise: random error or variance in a measured variable
- 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 which requires data cleaning
    - duplicate records
    - incomplete data
    - inconsistent data

# How to Handle Noisy Data?

- Binning
  - first sort data and partition into (equal-frequency) bins
  - then one can <span style="color:red">smooth by bin means, smooth by bin median, smooth by bin boundaries</span>, etc.
- Regression
  - smooth by fitting the data into regression functions
- Clustering
  - detect and remove outliers
- Combined computer and human inspection
  - detect suspicious values and check by human (e.g., deal with possible outliers)

# Simple Discretization Methods: 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 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

- **Equal-depth** (frequency) partitioning

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

  - Good data scaling

  - Managing categorical attributes can be tricky

# Binning Methods for Data Smoothing

- ❑  Sorted data for price (in dollars): 4, 8, 9, 15, 21, 21, 24, 25, 26, 28, 29, 34
- *  Partition into equal-frequency (equi-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

# Regression



$$y = x + 1$$

Data Mining: Concepts and Techniques

# Cluster Analysis
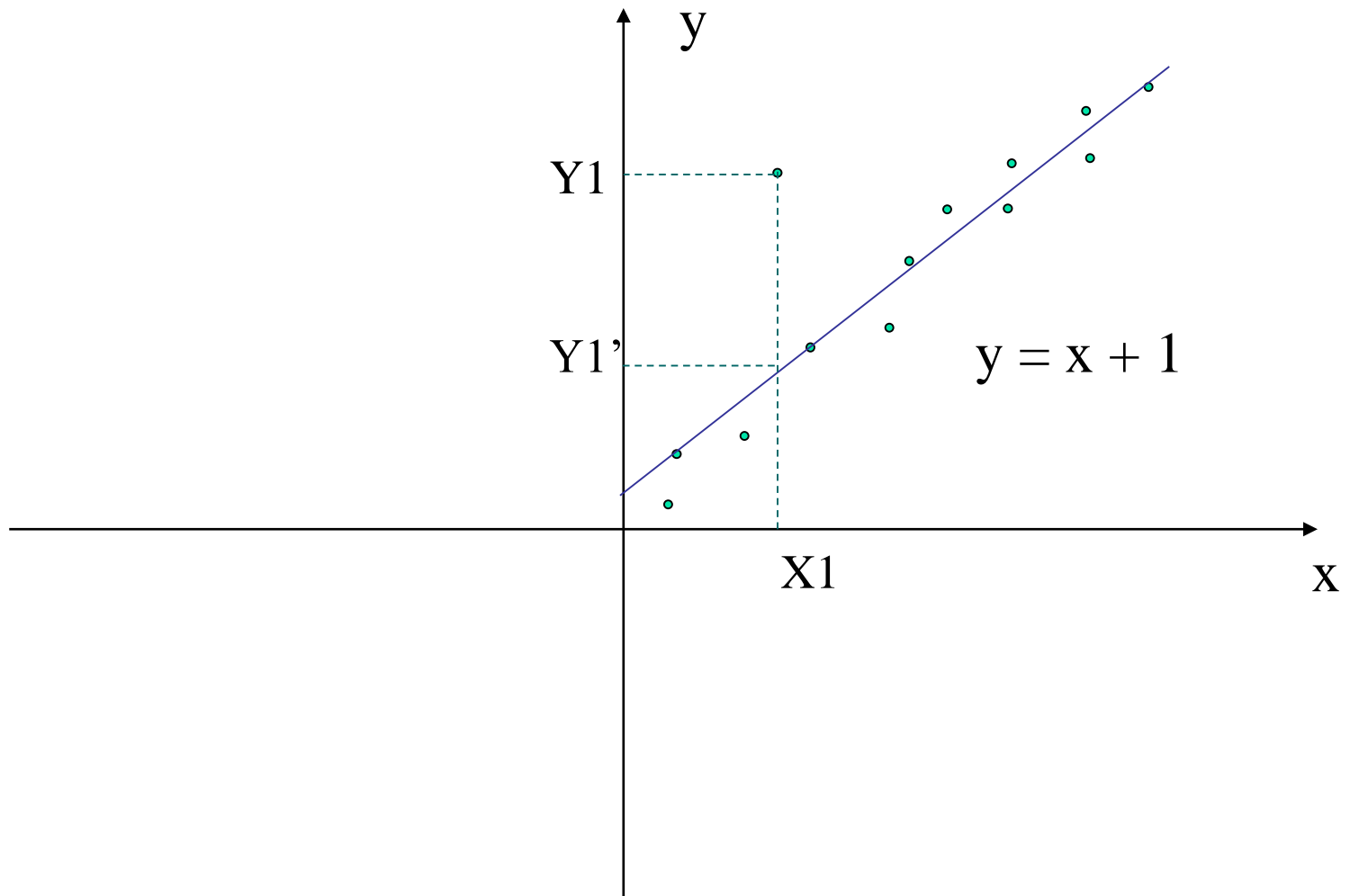


Data Mining: Concepts and Techniques
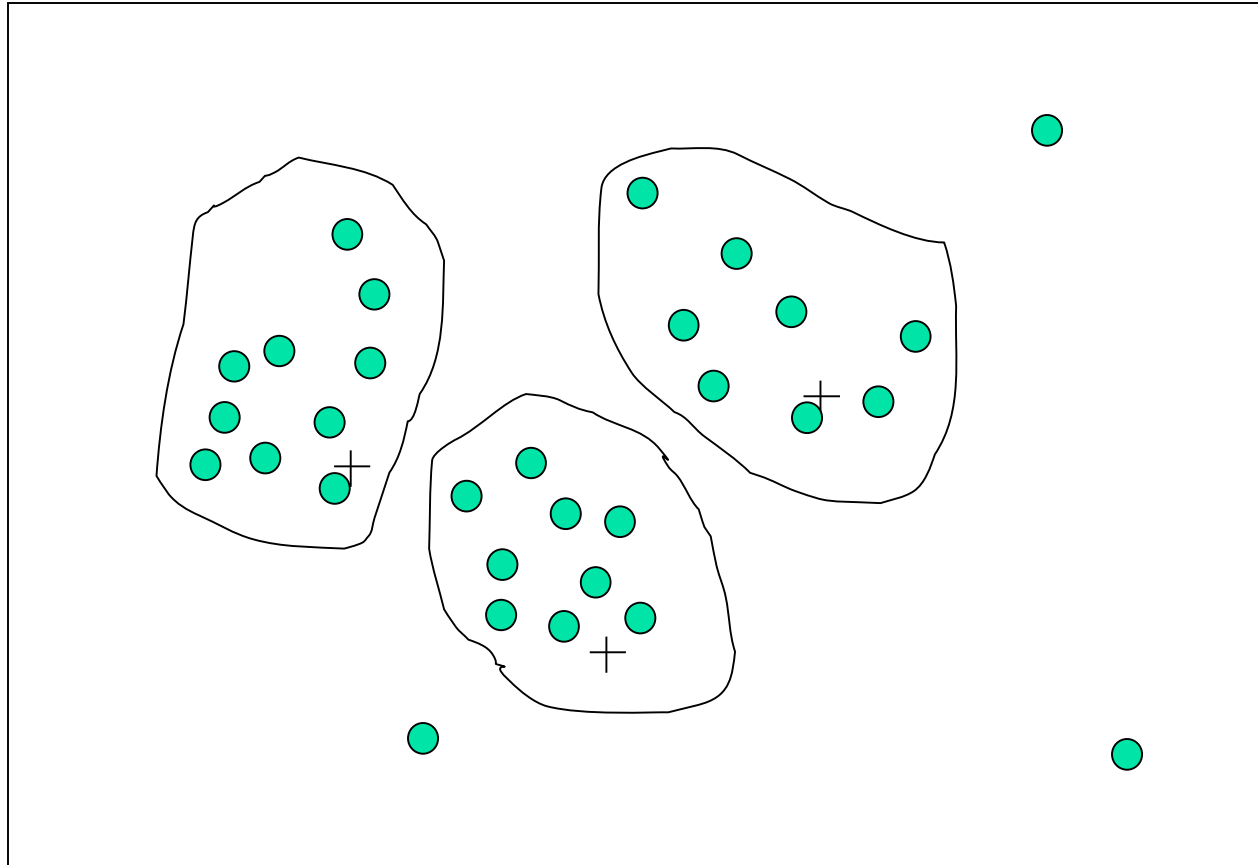
# Data Cleaning as a Process

- Data discrepancy detection
    - Use metadata (e.g., domain, range, dependency, distribution)
    - Check field overloading
    - Check uniqueness rule, consecutive rule and null rule
    - Use commercial tools
        - Data scrubbing: use simple domain knowledge (e.g., postal code, spell-check) to detect errors and make corrections
        - Data auditing: by analyzing data to discover rules and relationship to detect violators (e.g., correlation and clustering to find outliers)
- Data migration and integration
    - Data migration tools: allow transformations to be specified
    - ETL (Extraction/Transformation/Loading) tools: allow users to specify transformations through a graphical user interface
- Integration of the two processes
    - Iterative and interactive (e.g., Potter's Wheels)

# Chapter 2: Data Preprocessing

- Why preprocess the data?

- Data cleaning

- <span style="color:red">Data integration and transformation</span>

- Data reduction

- Discretization and concept hierarchy generation

- Summary

# Data Integration

- Data integration:
  - Combines data from multiple sources into a coherent store
- Schema integration: e.g., A.cust-id $\equiv$ B.cust-#
  - Integrate metadata from different sources
- Entity identification problem:
  - Identify real world entities from multiple data sources, e.g., Bill Clinton = William Clinton
- 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

# Handling Redundancy in Data Integration

- Redundant data occur often when integration of multiple databases

  - *Object identification*:  The same attribute or object may have different names in different databases

  - *Derivable data:* One attribute may be a "derived" attribute in another table, e.g., annual revenue

- Redundant attributes may be able to be detected by *correlation analysis*

- Careful integration of the data from multiple sources may help reduce/avoid redundancies and inconsistencies and improve mining speed and quality

# Correlation Analysis (Numerical Data)

- Correlation coefficient (also called Pearson's product moment coefficient)

$$r_{A,B} = \frac{\sum (A - \bar{A})(B - \bar{B})}{(n-1)\sigma_A \sigma_B} = \frac{\sum (AB) - n\bar{A}\bar{B}}{(n-1)\sigma_A \sigma_B}$$

  where n is the number of tuples, $\bar{A}$ and $\bar{B}$ are the respective means of A and B, $\sigma_A$ and $\sigma_B$ are the respective standard deviation of A and B, and $\Sigma(AB)$ is the sum of the AB cross-product.

- If $r_{A,B} > 0$, A and B are positively correlated (A's values increase as B's). The higher, the stronger correlation.

- $r_{A,B} = 0$: independent; $r_{A,B} < 0$: negatively correlated

# Correlation Analysis (Categorical Data)

- $X^2$ (chi-square) test

$$\chi^2 = \sum \frac{(Observed - Expected)^2}{Expected}$$

- The larger the $X^2$ value, the more likely the variables are related

- The cells that contribute the most to the $X^2$ value are those whose actual count is very different from the expected count

- Correlation does not imply causality
  - # of hospitals and # of car-theft in a city are correlated
  - Both are causally linked to the third variable: population

# Chi-Square Calculation: An Example

|  | Play chess | Not play chess | Sum (row) |
|---|---|---|---|
| Like science fiction | 250(90) | 200(360) | 450 |
| Not like science fiction | 50(210) | 1000(840) | 1050 |
| Sum(col.) | 300 | 1200 | 1500 |

- $X^2$ (chi-square) calculation (numbers in parenthesis are expected counts calculated based on the data distribution in the two categories)

$$\chi^2 = \frac{(250-90)^2}{90} + \frac{(50-210)^2}{210} + \frac{(200-360)^2}{360} + \frac{(1000-840)^2}{840} = 507.93$$

- It shows that like_science_fiction and play_chess are correlated in the group

# Data Transformation

- Smoothing: remove noise from data

- Aggregation: summarization, data cube construction

- Generalization: concept hierarchy climbing

- Normalization: scaled to fall within a small, specified range

  - min-max normalization

  - z-score normalization

  - normalization by decimal scaling

- Attribute/feature construction

  - New attributes constructed from the given ones

# Data Transformation: Normalization

- 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 range \$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

$$v' = \frac{v}{10^j} \qquad \text{Where } j \text{ is the smallest integer such that } Max(|v'|) < 1$$

# Chapter 2: Data Preprocessing

- Why preprocess the data?

- Data cleaning

- Data integration and transformation

- Data reduction

- Discretization and concept hierarchy generation

- Summary

# Data Reduction Strategies

- Why data reduction?
    - A database/data warehouse may store terabytes of data
    - Complex data analysis/mining may take a very long time to run on the complete data set
- Data reduction
    - Obtain a reduced representation of the data set that is much smaller in volume but yet produce the same (or almost the same) analytical results
- Data reduction strategies
    - Data cube aggregation:
    - Dimensionality reduction — e.g., remove unimportant attributes
    - Data Compression
    - Numerosity reduction — e.g., fit data into models
    - Discretization and concept hierarchy generation

# Data Cube Aggregation

- The lowest level of a data cube (base cuboid)
  - The aggregated data for an <span style="color:red">individual entity of interest</span>
  - E.g., a customer in a phone calling data warehouse
- Multiple levels of aggregation in data cubes
  - Further reduce the size of data to deal with
- Reference appropriate levels
  - Use the smallest representation which is enough to solve the task
- Queries regarding aggregated information should be answered using data cube, when possible

# Attribute Subset Selection

- Feature selection (i.e., attribute subset selection):
  - Select a minimum set of features such that the probability distribution of different classes given the values for those features is as close as possible to the original distribution given the values of all features
  - reduce # of patterns in the patterns, easier to understand
- Heuristic methods (due to exponential # of choices):
  - Step-wise forward selection
  - Step-wise backward elimination
  - Combining forward selection and backward elimination
  - Decision-tree induction

# Example of Decision Tree Induction

Initial attribute set:
{A1, A2, A3, A4, A5, A6}

```
                        ┌─────────┐
                        │  A4 ?   │
                        └─────────┘
                       /           \
                ┌────────┐        ┌────────┐
                │  A1?   │        │  A6?   │
                └────────┘        └────────┘
               /         \       /         \
        ( Class 1 ) ( Class 2 ) ( Class 1 ) ( Class 2 )
```
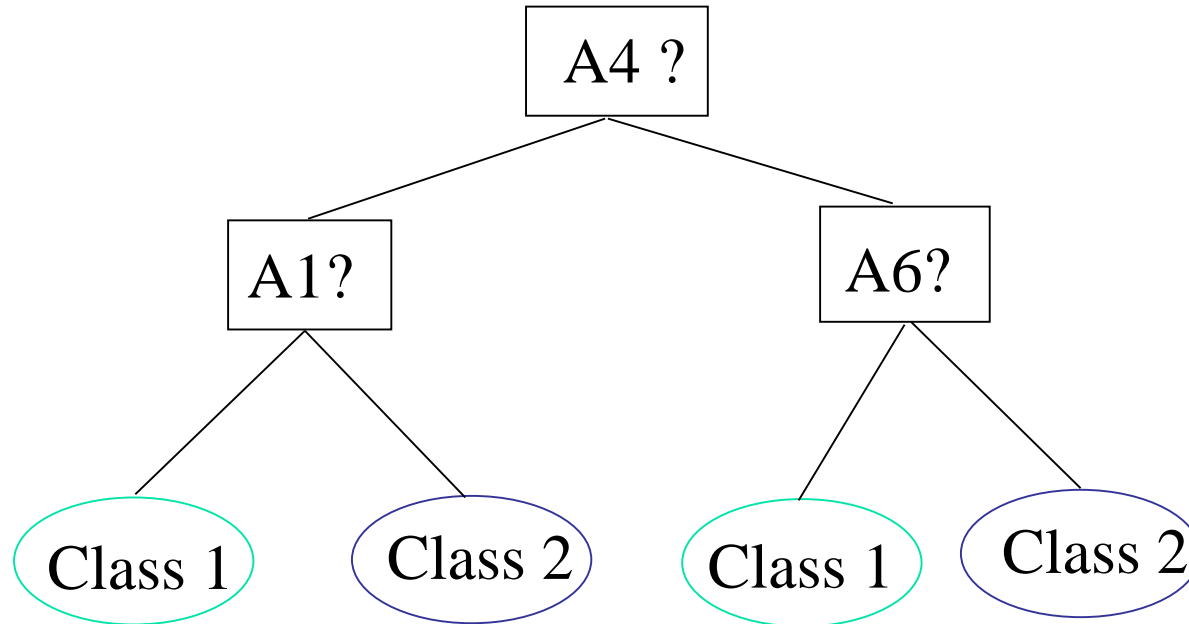
-------> Reduced attribute set: {A1, A4, A6}

# Heuristic Feature Selection Methods

- There are $2^d$ possible sub-features of $d$ features
- Several heuristic feature selection methods:
  - Best single features under the feature independence assumption: choose by significance tests
  - Best step-wise feature selection:
    - The best single-feature is picked first
    - Then next best feature condition to the first, ...
  - Step-wise feature elimination:
    - Repeatedly eliminate the worst feature
  - Best combined feature selection and elimination
  - Optimal branch and bound:
    - Use feature elimination and backtracking

# Data Compression

- String compression
  - There are extensive theories and well-tuned algorithms
  - Typically lossless
  - But only limited manipulation is possible without expansion
- Audio/video compression
  - Typically lossy compression, with progressive refinement
  - Sometimes small fragments of signal can be reconstructed without reconstructing the whole
- Time sequence is not audio
  - Typically short and vary slowly with time

# Data Compression



The diagram shows "Original Data" cylinder with an arrow pointing right to "Compressed Data" box. An arrow labeled "lossless" points back from Compressed Data to Original Data. An arrow labeled "lossy" points from Compressed Data down to "Original Data Approximated" cylinder.
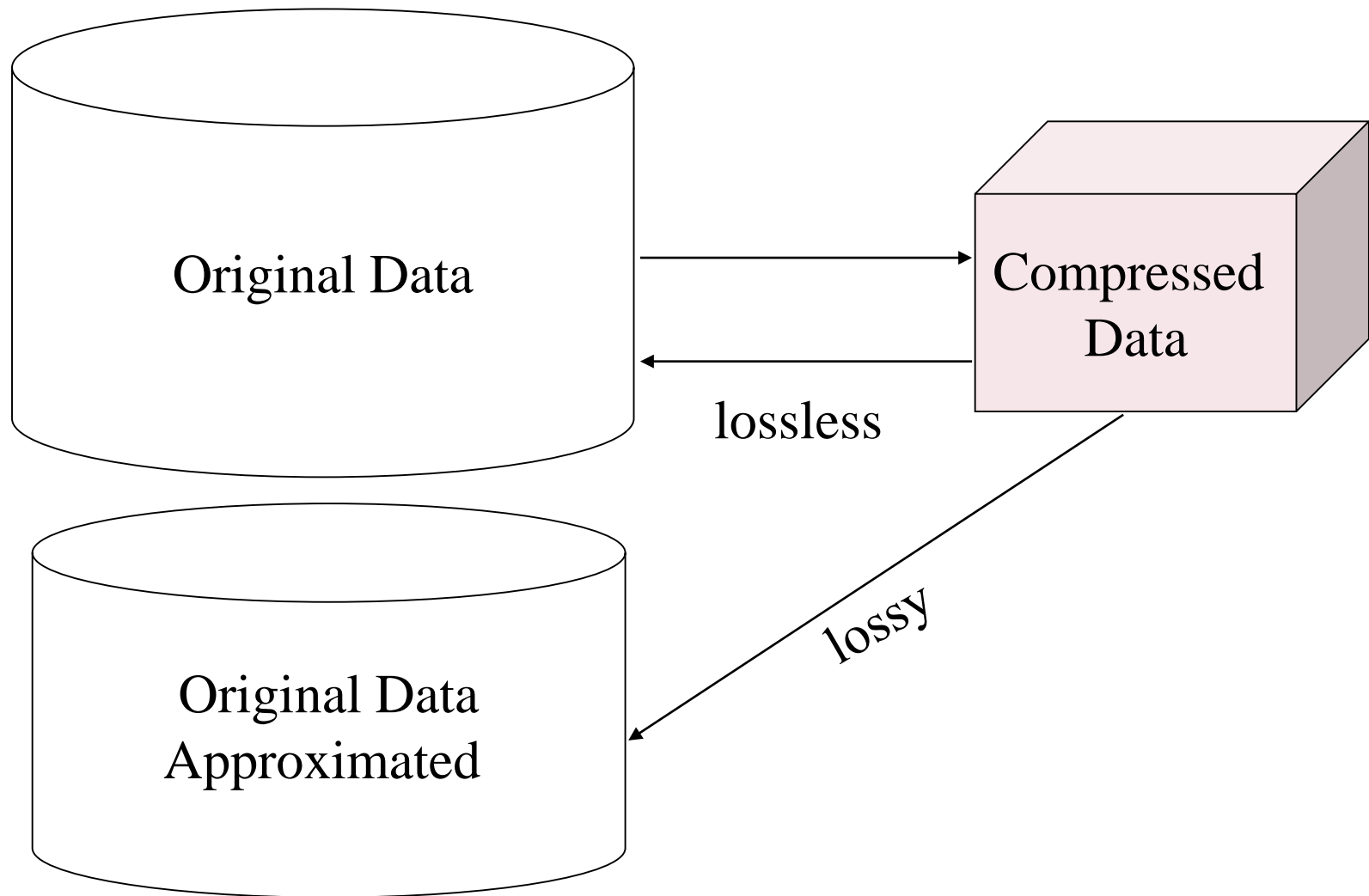
# Dimensionality Reduction: Wavelet Transformation

Haar2     Daubechie4
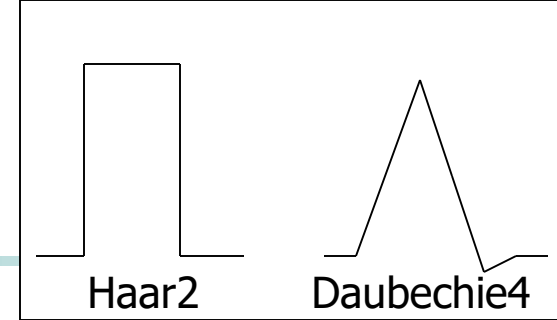
- Discrete wavelet transform (DWT): linear signal processing, multi-resolutional analysis

- Compressed approximation: store only a small fraction of the strongest of the wavelet coefficients

- Similar to discrete Fourier transform (DFT), but better lossy compression, localized in space

- Method:
  - Length, L, must be an integer power of 2 (padding with 0's, when necessary)
  - Each transform has 2 functions: smoothing, difference
  - Applies to pairs of data, resulting in two set of data of length L/2
  - Applies two functions recursively, until reaches the desired length

# DWT for Image Compression

- Image

```
          Image
         /      \
   Low Pass    High Pass
     /    \
Low Pass   High Pass
   /    \
Low Pass  High Pass
```
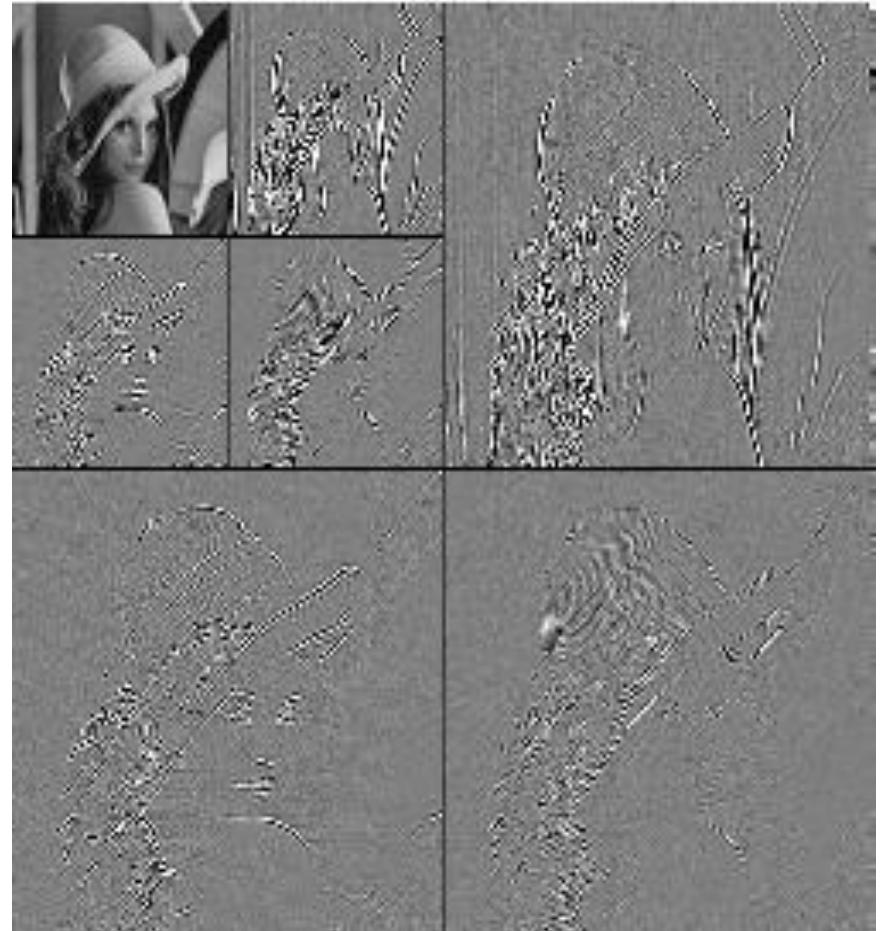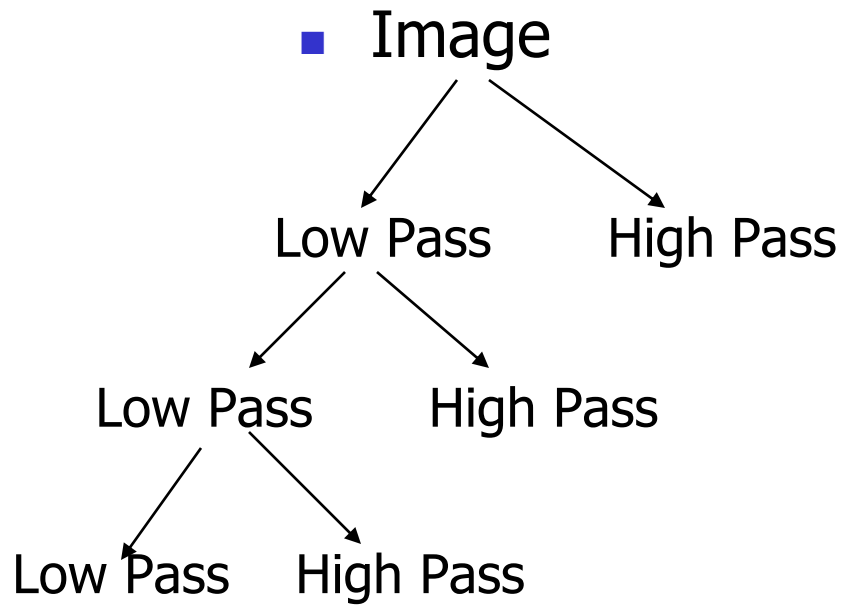
# Dimensionality Reduction: Principal Component Analysis (PCA)

- Given $N$ data vectors from $n$-dimensions, find $k \leq n$ orthogonal vectors (*principal components*) that can be best used to represent data
- Steps
  - Normalize input data: Each attribute falls within the same range
  - 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
- Works for numeric data only
- Used when the number of dimensions is large

# Principal Component Analysis



Data Mining: Concepts and Techniques
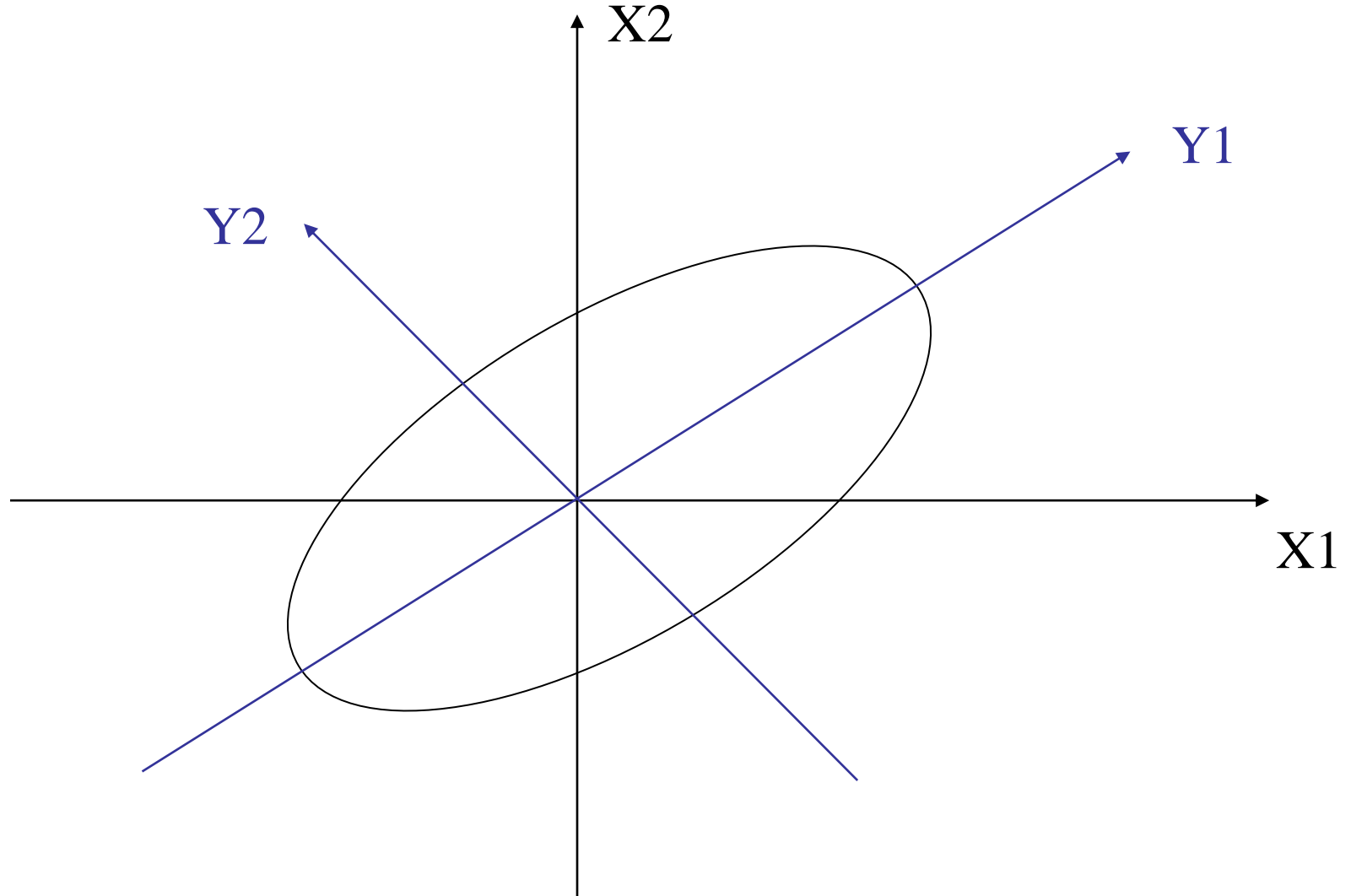
# Numerosity Reduction

- Reduce data volume by choosing alternative, smaller forms of data representation
- Parametric methods
  - Assume the data fits some model, estimate model parameters, store only the parameters, and discard the data (except possible outliers)
  - Example: Log-linear models—obtain value at a point in m-D space as the product on appropriate marginal subspaces
- Non-parametric methods
  - Do not assume models
  - Major families: histograms, clustering, sampling

# Data Reduction Method (1): 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

# Regress Analysis and Log-Linear Models

- Linear regression: $Y = w X + b$
    - Two regression coefficients, $w$ and $b$, specify the line and are to be estimated by using the data at hand
    - Using the least squares criterion to the known values of $Y_1, Y_2, ..., X_1, X_2, ....$
- 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: $p(a, b, c, d) = \alpha_{ab} \beta_{ac} \chi_{ad} \delta_{bcd}$

# Data Reduction Method (2): Histograms

- Divide data into buckets and store average (sum) for each bucket

- Partitioning rules:

    - Equal-width: equal bucket range

    - Equal-frequency (or equal-depth)

    - V-optimal: with the least *histogram variance* (weighted sum of the original values that each bucket represents)

    - MaxDiff: set bucket boundary between each pair for pairs have the $\beta-1$ largest differences
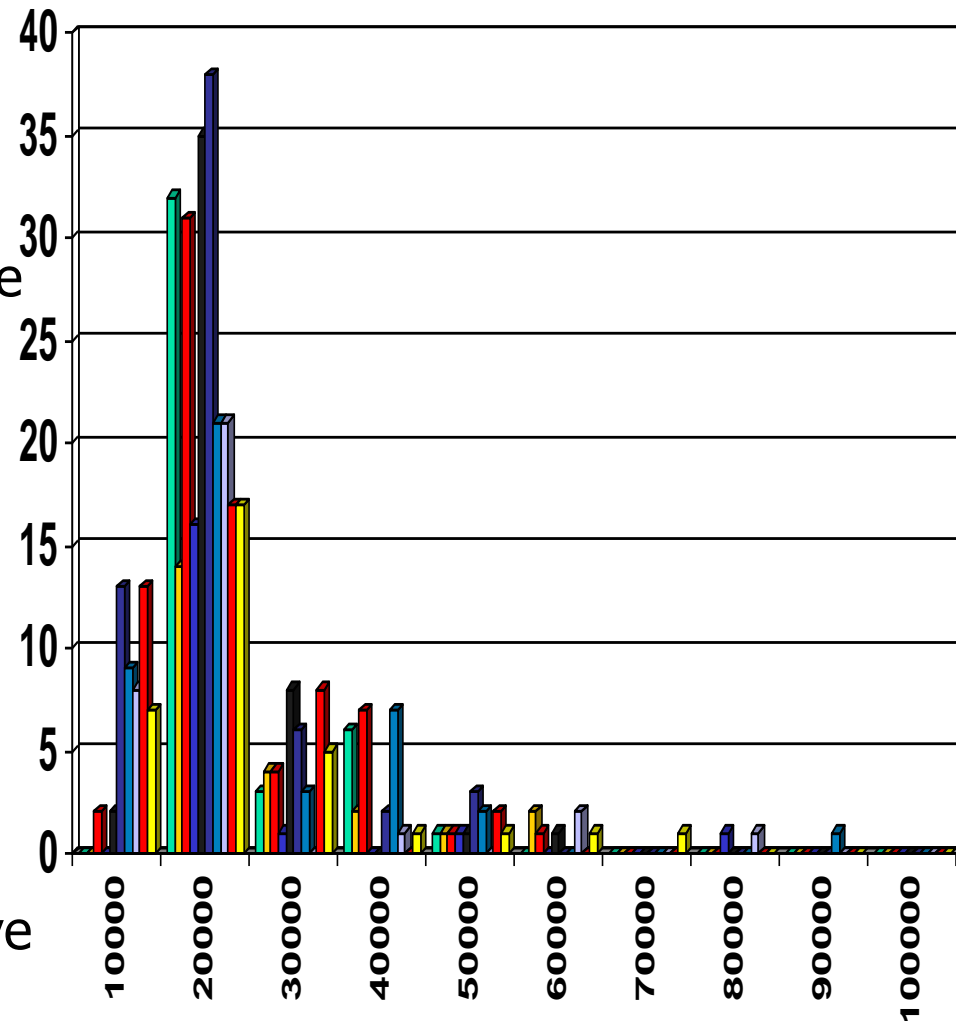
# Data Reduction Method (3): Clustering

- Partition data set into clusters based on similarity, and store cluster representation (e.g., centroid 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 depth in Chapter 7

# Data Reduction Method (4): Sampling

- Sampling: obtaining a small sample $s$ 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
- Choose a <span style="color:red">representative</span> subset of the data
    - Simple random sampling may have very poor performance in the presence of skew
- Develop 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
- Note: Sampling may not reduce database I/Os (page at a time)

# Sampling: with or without Replacement



SRSWOR
(simple random sample without replacement)

SRSWR
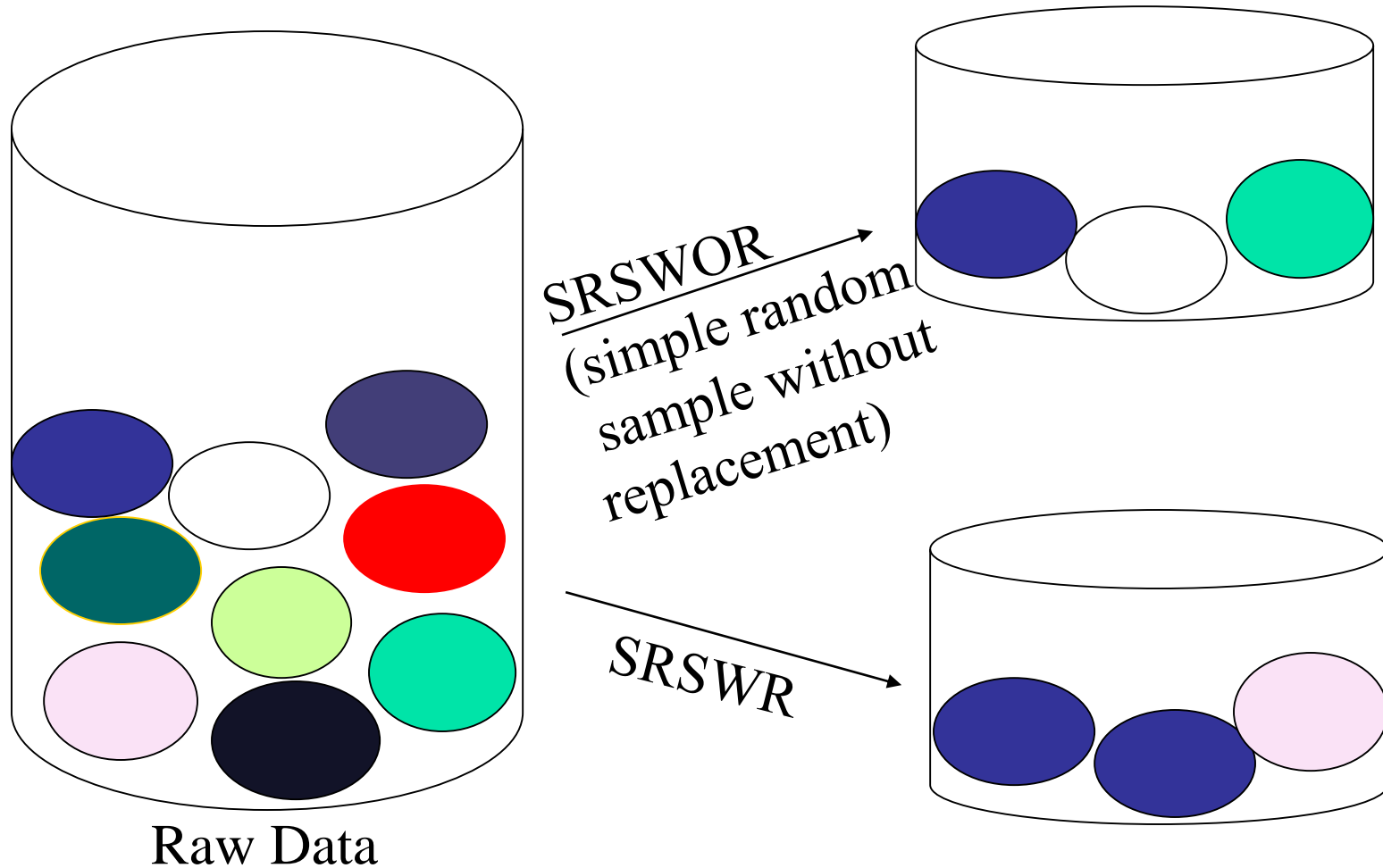
Raw Data

# Sampling: Cluster or Stratified Sampling

Raw Data
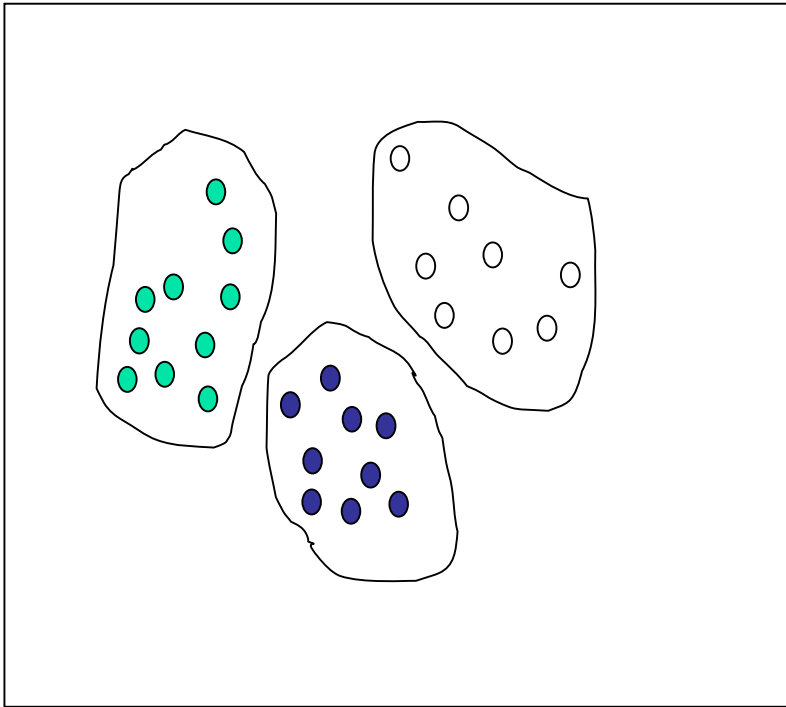
Cluster/Stratified Sample
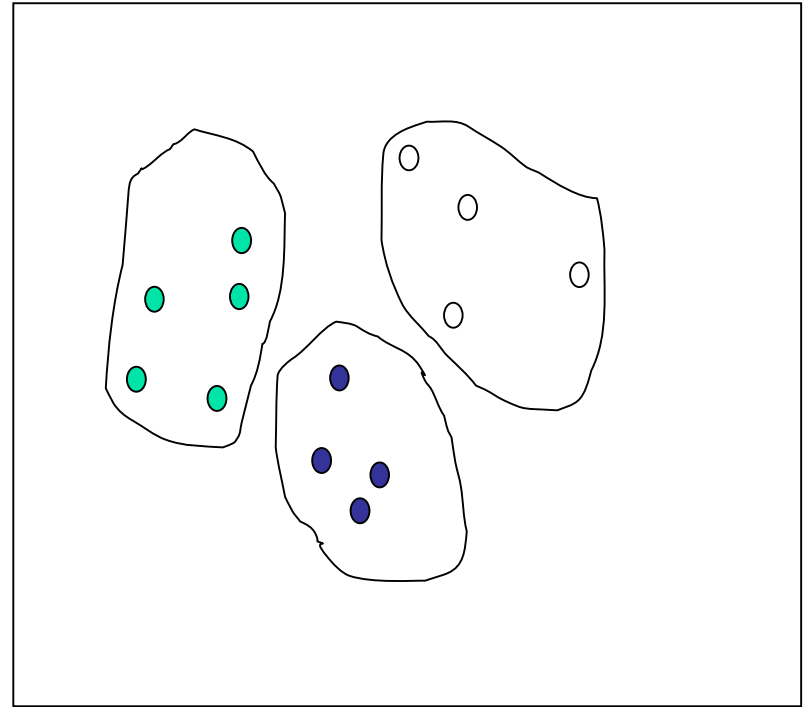
# Chapter 2: Data Preprocessing

- Why preprocess the data?

- Data cleaning

- Data integration and transformation

- Data reduction

- <span style="color:red">Discretization and concept hierarchy generation</span>

- Summary

# Discretization

- Three types of attributes:

    - Nominal — values from an unordered set, e.g., color, profession

    - 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 and Concept Hierarchy

- Discretization

  - Reduce the number of values for a given continuous attribute by dividing the range of the attribute into intervals

  - Interval labels can then be used to replace actual data values

  - Supervised vs. unsupervised

  - Split (top-down) vs. merge (bottom-up)

  - Discretization can be performed recursively on an attribute

- Concept hierarchy formation

  - Recursively reduce the data by collecting and replacing low level concepts (such as numeric values for age) by higher level concepts (such as young, middle-aged, or senior)

# Discretization and Concept Hierarchy Generation for Numeric Data

- Typical methods: All the methods can be applied recursively

  - Binning (covered above)

    - Top-down split, unsupervised,

  - Histogram analysis (covered above)

    - Top-down split, unsupervised

  - Clustering analysis (covered above)

    - Either top-down split or bottom-up merge, unsupervised

  - Entropy-based discretization: supervised, top-down split

  - Interval merging by $\chi^2$ Analysis: unsupervised, bottom-up merge

  - Segmentation by natural partitioning: top-down split, unsupervised

# Entropy-Based Discretization

- Given a set of samples S, if S is partitioned into two intervals $S_1$ and $S_2$ using boundary T, the information gain after partitioning is

$$I(S,T) = \frac{|S_1|}{|S|} Entropy(S_1) + \frac{|S_2|}{|S|} Entropy(S_2)$$

- Entropy is calculated based on class distribution of the samples in the set. Given *m* classes, the entropy of $S_1$ is

$$Entropy(S_1) = -\sum_{i=1}^{m} p_i \log_2(p_i)$$

where $p_i$ is the probability of class *i* in $S_1$

- The boundary that minimizes the entropy function over all possible boundaries is selected as a binary discretization

- The process is recursively applied to partitions obtained until some stopping criterion is met

- Such a boundary may reduce data size and improve classification accuracy

# Interval Merge by $\chi^2$ Analysis

- Merging-based (bottom-up) vs. splitting-based methods

- Merge: Find the best neighboring intervals and merge them to form larger intervals recursively

- ChiMerge [Kerber AAAI 1992, See also Liu et al. DMKD 2002]

  - Initially, each distinct value of a numerical attr. A is considered to be one interval

  - $\chi^2$ tests are performed for every pair of adjacent intervals

  - Adjacent intervals with the least $\chi^2$ values are merged together, since low $\chi^2$ values for a pair indicate similar class distributions

  - This merge process proceeds recursively until a predefined stopping criterion is met (such as significance level, max-interval, max inconsistency, etc.)

# Segmentation by Natural Partitioning

- A simply 3-4-5 rule can be used to segment numeric data into relatively uniform, "natural" intervals.

  - If an interval covers 3, 6, 7 or 9 distinct values at the most significant digit, partition the range into 3 equi-width intervals

  - If it covers 2, 4, or 8 distinct values at the most significant digit, partition the range into 4 intervals

  - If it covers 1, 5, or 10 distinct values at the most significant digit, partition the range into 5 intervals

# Example of 3-4-5 Rule



Step 1:   -$351   -$159                profit            $1,838    $4,700
          Min     Low (i.e, 5%-tile)                     High(i.e, 95%-0 tile)   Max

Step 2:   msd=1,000        Low=-$1,000     High=$2,000

Step 3:                          (-$1,000 - $2,000)

            (-$1,000 - 0)      (0 -$ 1,000)      ($1,000 - $2,000)

Step 4:                          (-$400 -$5,000)

        (-$400 - 0)       (0 - $1,000)          ($1,000 - $2, 000)        ($2,000 - $5, 000)

(-$400 -        (0 -            ($1,000 -              ($2,000 -
-$300)          $200)          $1,200)                $3,000)

                ($200 -        ($1,200 -
(-$300 -        $400)          $1,400)
-$200)
                                                       ($3,000 -
                               ($1,400 -               $4,000)
(-$200 -        ($400 -        $1,600)
-$100)          $600)                                           ($4,000 -
                                                                $5,000)
                               ($1,600 -    ($1,800 -
(-$100 -        ($600 -        $1,800)      $2,000)
0)              $800)      ($800 -
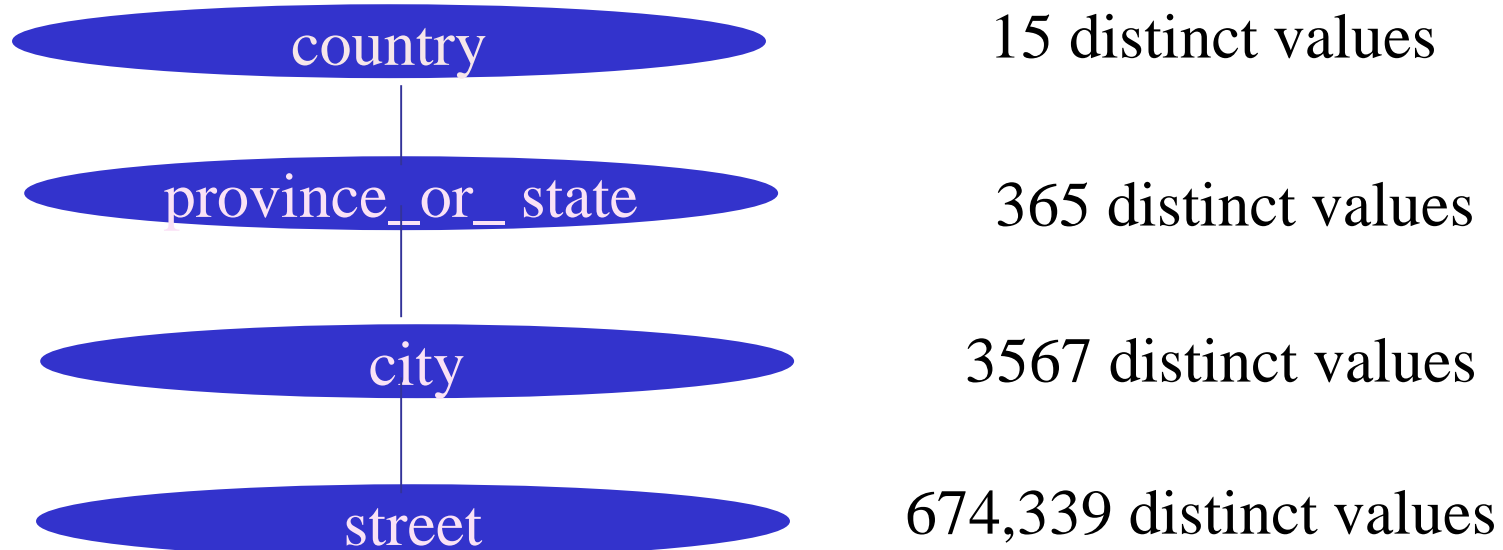                           $1,000)
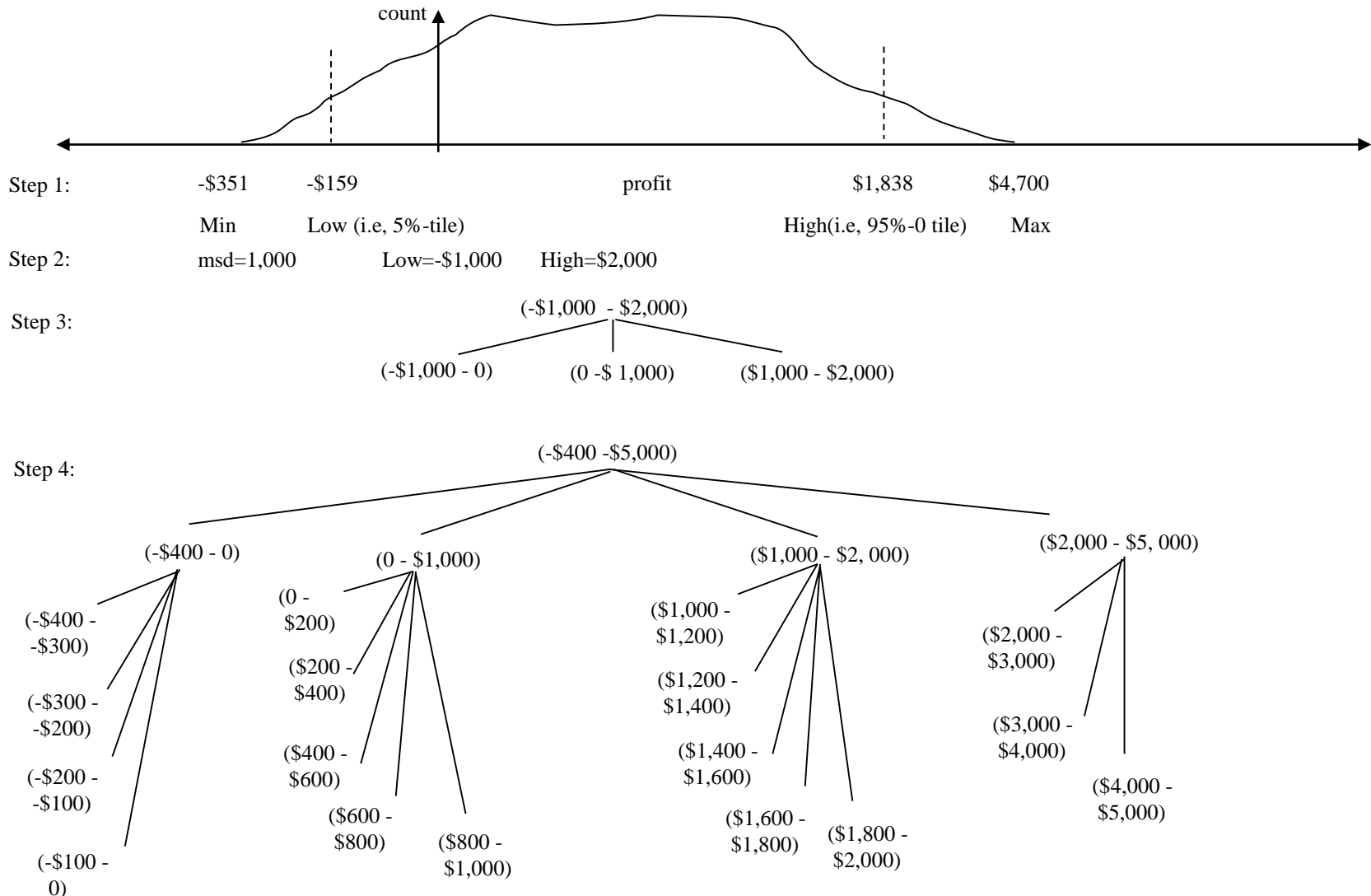
# Concept Hierarchy Generation for Categorical Data

- Specification of a partial/total ordering of attributes explicitly at the schema level by users or experts
  - street < city < state < country
- Specification of a hierarchy for a set of values by explicit data grouping
  - {Urbana, Champaign, Chicago} < Illinois
- Specification of only a partial set of attributes
  - E.g., only street < city, not others
- Automatic generation of hierarchies (or attribute levels) by the analysis of the number of distinct values
  - E.g., for a set of attributes: {street, city, state, country}

# Automatic Concept Hierarchy Generation

- Some hierarchies can be automatically generated based on the analysis of the number of distinct values per attribute in the data set
  - The attribute with the most distinct values is placed at the lowest level of the hierarchy
  - Exceptions, e.g., weekday, month, quarter, year

country                        15 distinct values

province_or_ state             365 distinct values

city                           3567 distinct values

street                         674,339 distinct values

# Chapter 2: Data Preprocessing

- Why preprocess the data?

- Data cleaning

- Data integration and transformation

- Data reduction

- Discretization and concept hierarchy generation

- <span style="color:red">Summary</span>

# Summary

- Data preparation or preprocessing is a big issue for both data warehousing and data mining

- Discriptive data summarization is need for quality data preprocessing

- Data preparation includes

  - Data cleaning and data integration

  - Data reduction and feature selection

  - Discretization

- A lot a methods have been developed but data preprocessing still an active area of research

# References

- D. P. Ballou and G. K. Tayi. Enhancing data quality in data warehouse environments. Communications of ACM, 42:73-78, 1999

- T. Dasu and T. Johnson. Exploratory Data Mining and Data Cleaning. John Wiley & Sons, 2003

- T. Dasu, T. Johnson, S. Muthukrishnan, V. Shkapenyuk. Mining Database Structure; Or, How to Build a Data Quality Browser. SIGMOD'02.

- H.V. Jagadish et al., Special Issue on Data Reduction Techniques. Bulletin of the Technical Committee on Data Engineering, 20(4), December 1997

- D. Pyle. Data Preparation for Data Mining. Morgan Kaufmann, 1999

- E. Rahm and H. H. Do. Data Cleaning: Problems and Current Approaches. *IEEE Bulletin of the Technical Committee on Data Engineering. Vol.23, No.4*

- V. Raman and J. Hellerstein. Potters Wheel: An Interactive Framework for Data Cleaning and Transformation, VLDB'2001

- T. Redman. Data Quality: Management and Technology. Bantam Books, 1992

- Y. Wand and R. Wang. Anchoring data quality dimensions ontological foundations. Communications of ACM, 39:86-95, 1996

- R. Wang, V. Storey, and C. Firth. A framework for analysis of data quality research. IEEE Trans. Knowledge and Data Engineering, 7:623-640, 1995