


# Chapter 3: Data Preprocessing

---

- Data Preprocessing: An Overview 
  - Data Quality
  - Major Tasks in Data Preprocessing
- Data Cleaning
- Data Integration
- Data Reduction
- Data Transformation and Data Discretization
- Summary



# Data Quality: Why Preprocess the Data?

---

- Measures for data quality: A multidimensional view
  - Accuracy: correct or wrong, accurate or not
  - Completeness: not recorded, unavailable, ...
  - Consistency: some modified but some not, dangling, ...
  - Timeliness: timely updated?
  - Believability: how trustable the data are?
  - Interpretability: how easily the data can be understood?



# 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 reduction**

- Dimensionality reduction
- Numerosity reduction
- Data compression


- **Data transformation and data discretization**

- Normalization
- Concept hierarchy generation



# Chapter 3: Data Preprocessing

---

- Data Preprocessing: An Overview
  - Data Quality
  - Major Tasks in Data Preprocessing
- Data Cleaning 
- Data Integration
- Data Reduction
- Data Transformation and Data Discretization
- Summary



# Data Cleaning

- Data in the Real World Is Dirty: Lots of potentially incorrect data, e.g., instrument faulty, human or computer error, transmission error
  - incomplete: lacking attribute values, lacking certain attributes of interest, or containing only aggregate data
    - e.g., *Occupation*=" " (missing data)
  - noisy: containing noise, errors, or outliers
    - e.g., *Salary*="−10" (an error)
  - inconsistent: containing discrepancies in codes or names, e.g.,
    - *Age*="42", *Birthday*="03/07/2010"
    - Was rating "1, 2, 3", now rating "A, B, C"
    - discrepancy between duplicate records
  - Intentional (e.g., *disguised missing* data)
    - Jan. 1 as everyone's birthday?



# Incomplete (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 (when doing classification)—not effective when the % of missing values per attribute is large
- 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 be due to
  - Faulty data collection instruments
  - Data entry problems
  - Data transmission problems
  - Inconsistency in naming convention (cm => inch)
- **Other data problems** which require data cleaning
  - Duplicate records
  - Inconsistent data





# How to Handle Noisy Data?


---

- Clustering
  - Detect and remove outliers
- Combined computer and human inspection
  - Detect suspicious values and check by human (e.g., deal with possible outliers)



# Chapter 3: Data Preprocessing

---

- Data Preprocessing: An Overview
  - Data Quality
  - Major Tasks in Data Preprocessing
- Data Cleaning
- Data Integration 
- Data Reduction
- Data Transformation and Data Discretization
- 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., cm vs. inch, meter vs. mile



# 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., birthdate vs. age
- Redundant attributes may be able to be detected by *correlation analysis* and *covariance analysis*
- Careful integration of the data from multiple sources may help reduce/avoid redundancies and inconsistencies
  - Improves mining speed and quality



# Correlation Analysis (Nominal Data)

---

	Play chess	Not play chess	Sum (row)
Like science fiction	250	200	450
Not like science fiction	50	1000	1050
Sum(col.)	300	1200	1500

- We want to know that **like\_science\_fiction** and **play\_chess** are **correlated** in the group



# Correlation Analysis (Nominal Data)

---

- **X<sup>2</sup> (chi-square) test**

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

- The larger the X<sup>2</sup> value, the more likely the variables are related
  - The cells that contribute the most to the X<sup>2</sup> 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

- $\chi^2$  (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



# Correlation Analysis (Numeric Data)

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

$$r_{A,B} = \frac{\sum_{i=1}^n (a_i - \bar{A})(b_i - \bar{B})}{(n-1)\sigma_A\sigma_B} = \frac{\sum_{i=1}^n (a_i b_i) - n\bar{A}\bar{B}}{(n-1)\sigma_A\sigma_B}$$

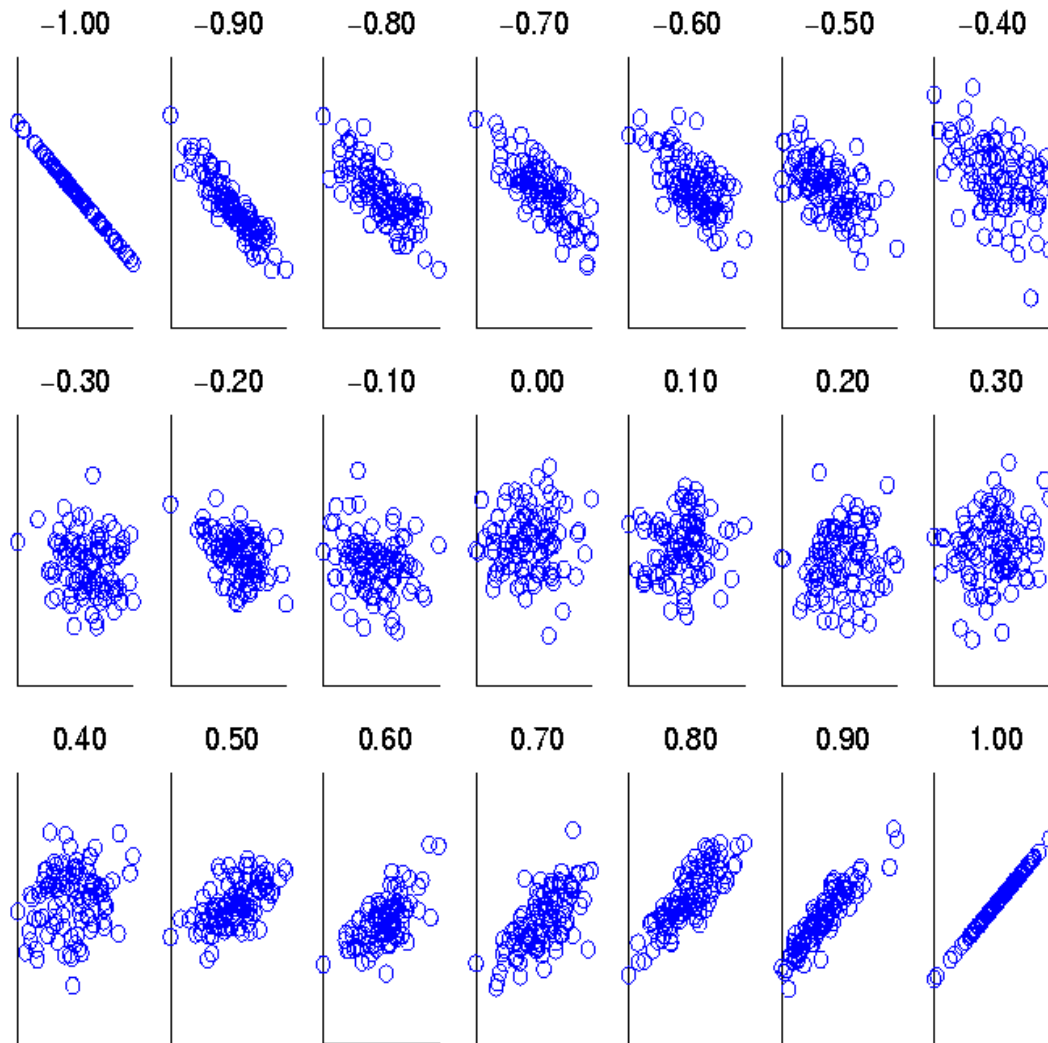
where n is the number of tuples,  $\bar{A}$  and  $\bar{B}$  are the respective means of A and B,  $\sigma_A$  and  $\sigma_B$  are the respective standard deviation of A and B, and  $\sum(a_i b_i)$  is the sum of the AB cross-product.

- 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
- $r_{AB} < 0$ : negatively correlated





# Visually Evaluating Correlation



**Scatter plots  
showing the  
similarity from  
-1 to 1.**



# Covariance (Numeric Data)

- Covariance is similar to correlation

$$Cov(A, B) = E((A - \bar{A})(B - \bar{B})) = \frac{\sum_{i=1}^n (a_i - \bar{A})(b_i - \bar{B})}{n}$$

Correlation coefficient:  $r_{A,B} = \frac{Cov(A, B)}{\sigma_A \sigma_B}$

where  $n$  is the number of tuples,  $\bar{A}$  and  $\bar{B}$  are the respective mean or **expected values** of  $A$  and  $B$ ,  $\sigma_A$  and  $\sigma_B$  are the respective standard deviation of  $A$  and  $B$ .

- **Positive covariance:** If  $Cov_{A,B} > 0$ , then  $A$  and  $B$  both tend to be larger than their expected values.
- **Negative covariance:** If  $Cov_{A,B} < 0$  then if  $A$  is larger than its expected value,  $B$  is likely to be smaller than its expected value.
- **Independence:**  $Cov_{A,B} = 0$



# Co-Variance: An Example

$$Cov(A, B) = E((A - \bar{A})(B - \bar{B})) = \frac{\sum_{i=1}^n (a_i - \bar{A})(b_i - \bar{B})}{n}$$

- It can be simplified in computation as


$$Cov(A, B) = E(A \cdot B) - \bar{A}\bar{B}$$

- Suppose two stocks A and B have the following values in one week:  
(2, 5), (3, 8), (5, 10), (4, 11), (6, 14).
- Question: If the stocks are affected by the same industry trends, will their prices rise or fall together?
  - $E(A) = (2 + 3 + 5 + 4 + 6) / 5 = 20/5 = 4$
  - $E(B) = (5 + 8 + 10 + 11 + 14) / 5 = 48/5 = 9.6$
  - $Cov(A, B) = (2 \times 5 + 3 \times 8 + 5 \times 10 + 4 \times 11 + 6 \times 14) / 5 - 4 \times 9.6 = 4$
- Thus, A and B rise together since  $Cov(A, B) > 0$ .



# Chapter 3: Data Preprocessing

---

- Data Preprocessing: An Overview
  - Data Quality
  - Major Tasks in Data Preprocessing
- Data Cleaning
- Data Integration
- Data Reduction 
- Data Transformation and Data Discretization
- Summary



# Data Reduction Strategies

---

- **Data reduction:** Obtain a reduced representation of the data set
  - Much smaller in volume but yet produces the same (or almost the same) analytical 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 attributes
    - Wavelet transforms; Principal Components Analysis (PCA)
    - Feature subset selection, feature creation
  - **Numerosity reduction** (some simply call it: Data Reduction)
    - Regression
    - Histograms, clustering, sampling
  - **Data compression**



# Data Reduction 1: Dimensionality Reduction

## ■ Curse of dimensionality

- When dimensionality increases, data becomes increasingly sparse
- Density and distance between points (which is critical to clustering and outlier analysis) becomes less meaningful

## ■ Dimensionality reduction

- 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

- Wavelet transforms
- Principal Component Analysis



# Wavelet Transformation

---

- Discrete wavelet transform (DWT)
  - For linear signal processing and multi-resolution 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



# Wavelet Decomposition

- Wavelets: A math tool for space-efficient hierarchical decomposition of functions
- $S = [2, 2, 0, 2, 3, 5, 4, 4]$  can be transformed to  $S_{\wedge} = [2^{3/4}, -1^{1/4}, 1/2, 0, 0, -1, -1, 0]$
- Compression:
  - many small detail coefficients can be replaced by 0's
  - only the significant coefficients are retained

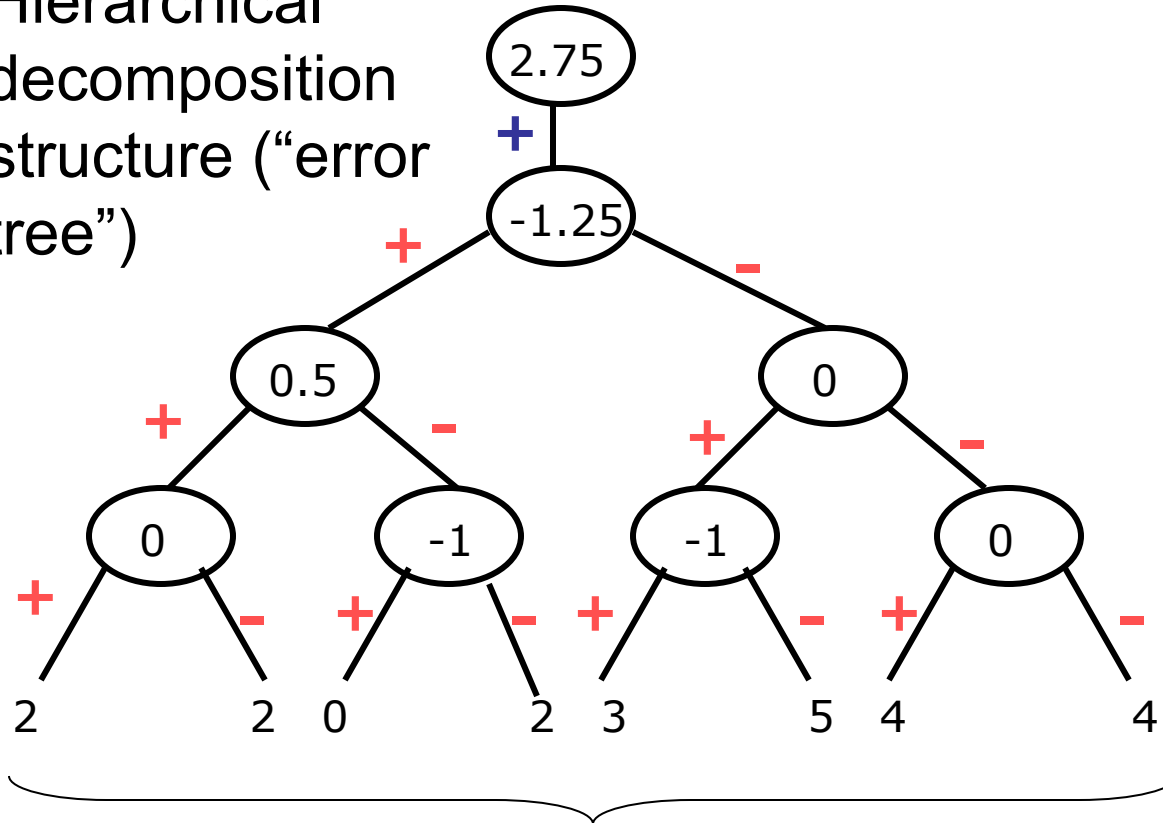
Resolution	Averages	Detail Coefficients
8	$[2, 2, 0, 2, 3, 5, 4, 4]$	
4	$[2, 1, 4, 4]$	$[0, -1, -1, 0]$
2	$[1\frac{1}{2}, 4]$	$[\frac{1}{2}, 0]$
1	$[2\frac{3}{4}]$	$[-1\frac{1}{4}]$





# Haar Wavelet Coefficients

Hierarchical decomposition structure ("error tree")



Original frequency distribution

Coefficient "Supports"

