Data Preprocessing

Overview

1. Data preprocessing an overview
2. Why data preprocessing?
3. Major steps in preprocessing
4. Data cleaning
5. Data Integration

Why Data Preprocessing?

- Real world databases are highly susceptible to noise, missing and inconsistent data.
- Low quality data will lead to low-quality mining results.
- Different data preprocessing techniques can improve the data quality
  - They improve the accuracy and efficiency of mining process.
  - Data cleaning, data integration, data reduction and data transformation are different preprocessing techniques.
  - Techniques are not mutually exclusive they may work together (e.g : Data cleaning may involve data transformations to correct wrong data)

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=“ ” (Missing Data)
  - **Inaccurate or noisy:** containing errors or values deivate from the expected
    - e.g., Salary=“-10” (an error)
  - **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
  - **Intentional:** (e.g., disguised missing data)
    - Jan. 1 as everyone’s birthday?

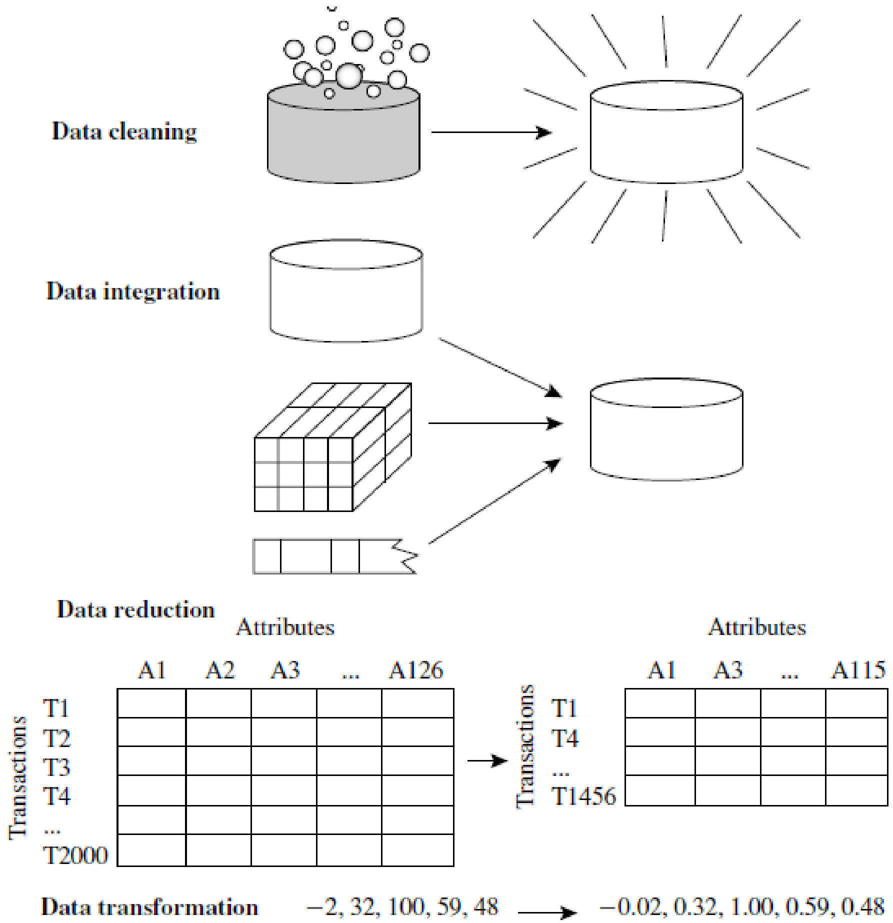
Multi-Dimensional Measure of Data Quality

- **Measures for data quality: A multidimensional view**
  - **Accuracy:** correct data, accurate value, no deviations
  - **Completeness:** not recorded, unavailable, only aggregate data
  - **Consistency:** some modified but some not, no discrepancy
  - **Timeliness:** timely update?
  - **Believability**: how much the data are trusted by users?
  - **Interpretability:** how easily the data can be understood?

Why Data is dirty?

- There are possible reasons for inaccurate data:
  - Faulty data collection instruments
  - Human or computer error at data entry
  - Users may purposely submit incorrect data values for mandatory fields (disguised missing data)
  - Errors in data transmission
  - Technology limitations such as limited buffer size for data transfer
  - Missing attributes or values of the attributes
  - Naming conventions or inconsistent formats for input fields

Major Tasks of Data Preprocessing



Major Tasks of Data Preprocessing

- **Data cleaning:** Can be applied to remove the noise and correct inconsistent of data.
  - filling missing data, smoothing noisy data, identifying or removing outliers, and resolving inconsistencies
- **Data integration** : Merges data from multiple sources into coherent data store such as data warehouse.
  - Integration of multiple databases, data cubes, or files
  - Attributes have different names in different data bases
  - Naming inconsistencies may also occur for attribute values

Major Tasks of Data Preprocessing

- **Data reduction:** Obtains reduced representation of data set smaller in volume but produces the same analytical results**.**
  - **Dimensionality reduction** :
    - Data encoding schemes are applied to obtain compressed data (Eg: Wavelet transform and PCA)
    - Attribute subset selection (removing irrelevant attributes)
    - Attribute construction (useful attributes)
  - **Numerosity reduction**: Smaller representation of data using parametric models or non-parametric models

Major Tasks of Data Preprocessing

- **Data transformation :** Normalization, Data discretization and concept hierarchy generation are forms of data transformation
  - **Normalization** : Data to be analyzed can be scaled to smaller range of values
  - **Discretization and Concept hierarchy generation**: Raw data values are replaced by ranges or higher conceptual levels

Data Cleaning

- Data cleaning:
  - Real world data tend to be incomplete, noisy and inconsistent.
  - Data cleaning routine attempts
    - to fill in missing values
    - Smooth out noise while identifying outliers
    - Correct inconsistencies in the data

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 (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 be due to
  - faulty data collection instruments
  - data entry problems
  - data transmission problems
  - technology limitation
  - inconsistency in naming convention
- **Other data problems** which require data cleaning
  - duplicate records, incomplete data and inconsistent data

How to Handle Noisy Data?

- **Binning**
  - First sort data and partition into (equal-frequency) bins
  - Perform local smoothing by consulting neighborhood values.
  - Then one can smooth by bin means, smooth by bin median, smooth by bin boundaries, etc.
- **Regression**
  - Smoothing the data by conforms data values to a functions
- **Outlier Analysis:**
  - Detect and remove outliers by organizing data into clusters.

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: $W = (B - A)/N$.
  - 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:

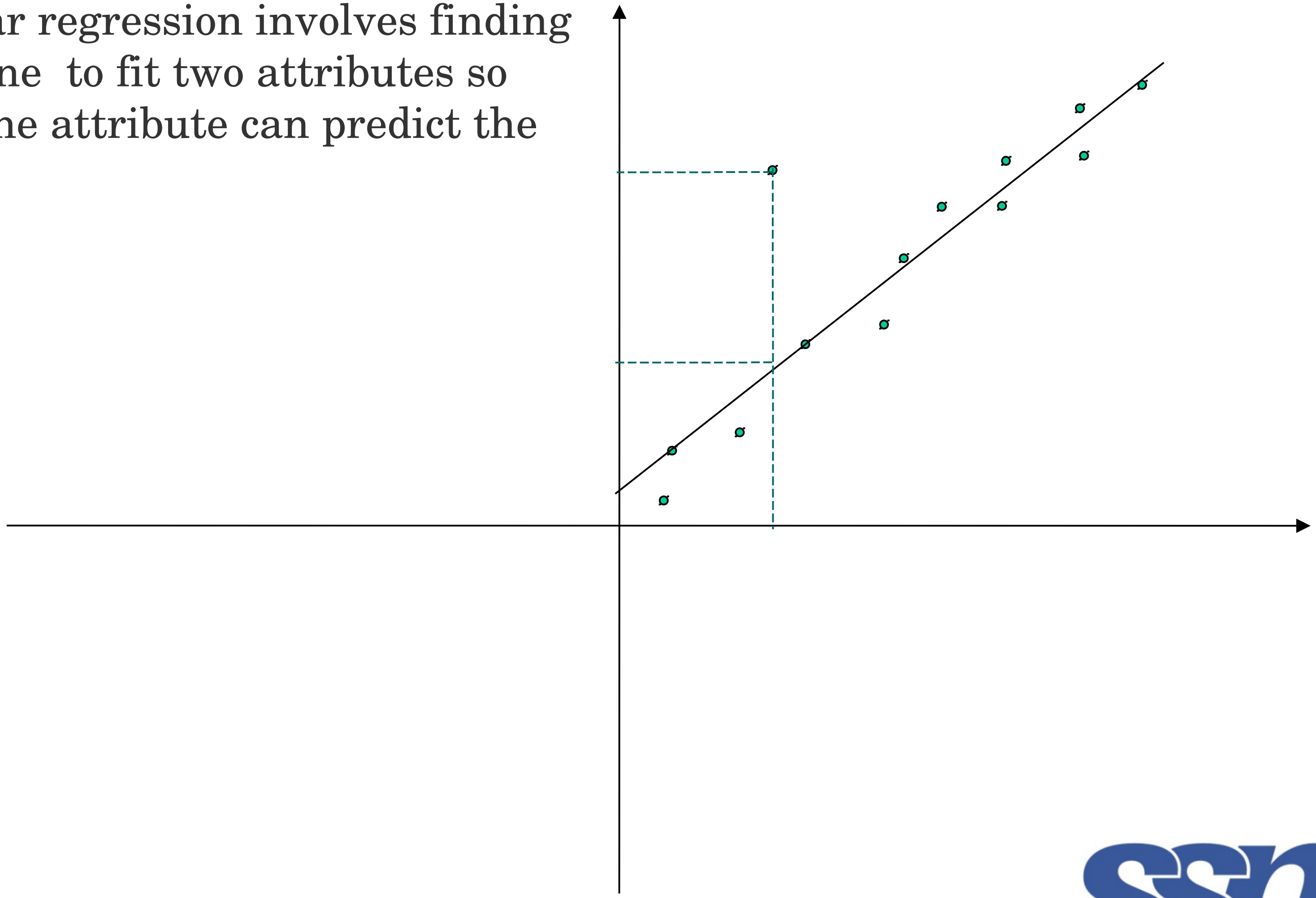
- Bin 1: 9, 9, 9, 9
- Bin 2: 23, 23, 23, 23
- Bin 3: 29, 29, 29, 29

Smoothing by bin boundaries:

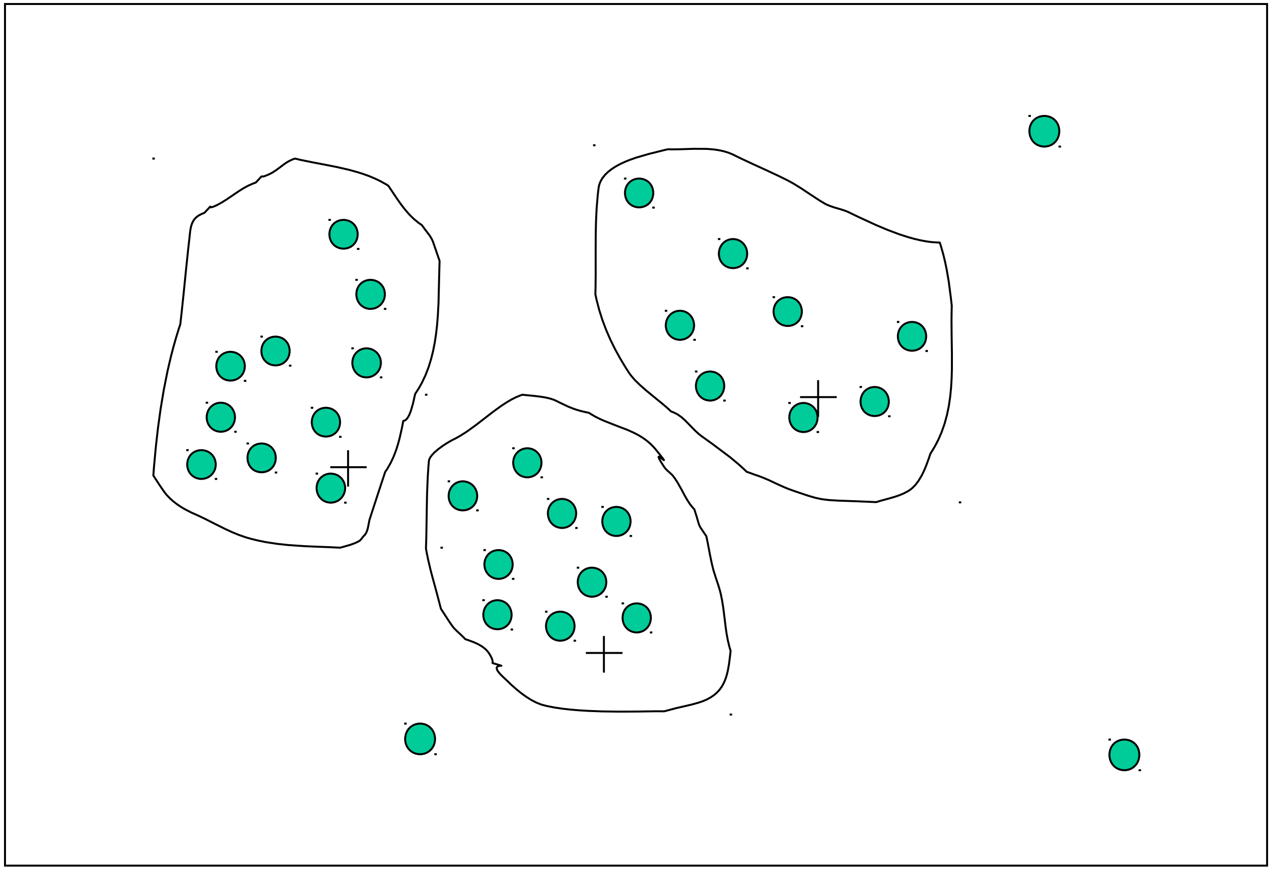
- Bin 1: 4, 4, 4, 15
- Bin 2: 21, 21, 25, 25
- Bin 3: 26, 26, 26, 34

Regression

- Linear regression involves finding best line to fit two attributes so that one attribute can predict the other.



Cluster Analysis



Data Cleaning as a Process

- The first step in data cleaning is to detect the discrepancy
- The discrepancy may due to several reasons:
    - Poorly designed data entry forms
    - Deliberate errors
    - Data decay
    - Human error and system error

Data discrepancy detection

Use metadata (e.g., domain, range, dependency, distribution)

Check field overloading

Check uniqueness rule, consecutive rule and null rule

Data Cleaning as a Process

- **Data discrepancy detection**
  - 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
- **Potter's Wheels**
  - Iterative and interactive
  - Integrates discrepancy detection and transformation

Data Integration

- **Data integration**:
  - Combines data from multiple sources into a coherent store
- 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
- Careful integration of the data from multiple sources may help reduce/avoid redundancies and inconsistencies and improve mining speed and quality

Entity Identification Problem

- Schema integration and object matching are tricky
- Schema integration: e.g., A.cust-id ≡ B.cust-#
  - Integrate metadata from different sources
- Object matching:
  - Identify real world entities from multiple data sources, e.g., Bill Clinton = William Clinton
  - By checking with their metadata and null rules
  - Special attention can be made on the structure of the data
    - Functional dependencies and the referential constraints in the source and the target system must match.

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* and *covariance analysis*
- *For nominal data => chi square test*
- *Numeric attributes=> correlation coefficient or covariance*

Correlation Analysis (for Nominal Data)

- The c^2 test is used to determine whether an association (or relationship) between 2 categorical variables in a sample.
- The test reflects a real association between these 2 variables in the population.
- Suppose A has c distinct values namely a_1, a_2,….a_c B has r distinct values b_1, b_2,….b_r
- The data tuples described by A and B forms the contigency table with c coulmns and r rows
- Let (A_i, B_j) joint event that A takes on value a_i and B takes a value b_j

Correlation Analysis (for Nominal Data)

- **X^2 (chi-square) test:**

$$\chi^2 = \sum_{i=1}^{c} \sum_{j=1}^{r} \frac{(o_{ij} - e_{ij})^2}{e_{ij}},$$

where o_{ij} is the *observed frequency* (i.e., actual count) of the joint event (A_i, B_j) and e_{ij} is the *expected frequency* of (A_i, B_j), which can be computed as

$$e_{ij} = \frac{count(A = a_i) \times count(B = b_j)}{n},$$

- Null hypothesis: The two distributions are independent
- The larger the X^2 value, the more likely the variables are related

Chi-Square Calculation: An Example

	Male	Female	Sum (row)
Like science fiction	250 (90)	200 (360)	450
Not like science fiction	50 (210)	1000 (840)	1050
Sum(col.)	300	1200	1500

How to derive 90?
300*450/1500
= 90

We can reject the null hypothesis of independence at a confidence level of 0.001

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

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

- For this 2*2 contigency table the degree of freedom is (r-1)*(c-1) .
- For 1 degree of freedom the chi square need to reject the hypothesis at the significance of 0.001 is 10.820

Chi-Square Calculation: An Example

- Since the computed value is higher we can reject the hypothesis that gender and preferred reading are independent.
- If the hypothesis can be rejected then the attribute are statistically correlated.
- The two attributes are correlated for the given group of people.
- Stronger correlation indicates any one attribute may be removed as redundancy.

Correlation Analysis (Numeric Data)

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

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

where n is the number of tuples, $\bar{A}$ and $\bar{B}$ are the respective means of A and B, σ_A and σ_B are the respective standard deviation of A and B, and $\Sigma(a_i b_i)$ 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.
  - Higher value indicate A or B may be removed as redundancy
- $r_{A,B} = 0$: independent; $r_{AB} < 0$: negatively correlated

Covariance (Numeric Data)

- Covariance is similar to correlation

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

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

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

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

Covariance (Numeric Data)

- **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 one attribute larger than its expected value and other attribute less than the expected value.

- **Independence**: $Cov_{A,B} = 0$ Both attributes are independent.

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: will their prices rise or fall together according to industry
  - 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) = (2×5+3×8+5×10+4×11+6×14)/5 − 4 × 9.6 = 4
- Thus, A and B rise together since Cov(A, B) > 0.

Tuple Duplication

- Tuple detection should be detected at tuple level.
  - Use of denormalized tables
  - Inaccurate data entry
  - Updating some but not all occurrences.

Data value Conflict Detection and Resolution

- Data integration involves the detection and resolution of data value conflicts
- Attribute values from different sources may differ.
  - May to due to difference in reprsentation, scaling or encoding.

  Eg: weight attribute stored in metric units in one sytsem and British units in another system
  - Schools with different grading and curriculum scheme
- Attributes may differ at abstraction level