CSE408 – Data Warehousing and Data Mining

Unit-1: Data Preprocessing

Prepared by,

Prof. Gayathri. P

Assistant Professor (Senior)

SCSE

Data Preprocessing

- Improves the quality of the data and consequently mining results.
- Real world dbs are highly noisy, has missing and inconsistent data due to their typically huge size.
- Low quality data leads to low quality mining results.
- Improves the data quality

Data Quality: Why Preprocess the Data?

- Measures for data quality
  - Accuracy: correct information
  - Completeness: complete information
  - Consistency: no discrepancies in the information

Major Tasks in Data Preprocessing

- **Data cleaning**
  - Remove noise and correct inconsistencies in the data
- **Data integration**
  - Merges data from multiple sources into coherent data store such as data warehouse.
- **Data transformation**
  - Improves accuracy and efficiency of mining algorithms.
- **Data reduction**
  - Reduce the data size by aggregating, eliminating redundant features.

Data Cleaning

- Data in the Real World Is
  - Dirty: Lots of potentially incorrect data, e.g., instrument faulty, human or computer 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)
    - Age = “200” (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

Tasks done in Data cleaning

- Fill in missing values
- Smooth noisy data
- Identify or remove outliers
- Resolve inconsistencies

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

How to Handle Missing Data?

- Ignore the tuple: usually done when class label is missing (when doing classification)—not effective
- 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 – eg. Loan decision = “safe”
  - the most probable value: inference-based such as Bayesian formula or decision tree

Noisy Data

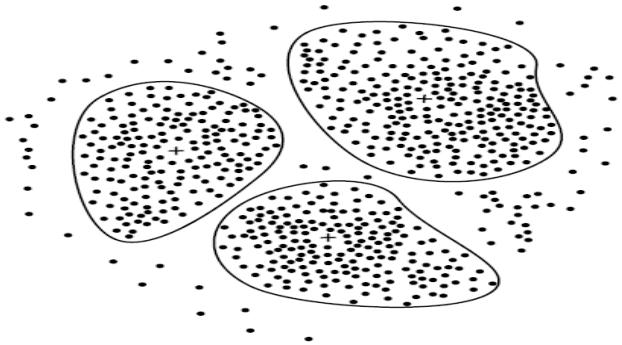
- Noise: random error or variance in a measured variable
- Incorrect attribute values may be due to
  - faulty data collection instruments
  - data entry problems
  - technology limitation
  - inconsistency in naming convention

How to Handle Noisy Data?

- Binning
  - first sort data
  - partition into (equal-frequency) bins
  - Then smooth by bin means, smooth by bin median, smooth by bin boundaries, etc.
- Regression
  - smooth by fitting the data into regression functions
  - Linear regression – finding the best line to fit two attributes. So that one attribute can be used to predict the other.
  - Multiple linear regression – More than two attributes are involved.

How to Handle Noisy Data?

- Clustering
  - detect and remove outliers
  - Similar values are organized into groups or clusters.
  - Values that fall outside the set of clusters may be considered as outliers.



A 2-D plot of customer data with respect to customer locations in a city, showing three data clusters. Each cluster centroid is marked with a "+", representing the average point in space for that cluster. Outliers may be detected as values that fall outside of the sets of clusters.

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**: each value in the bin is replaced by the mean value of the bin
  - Bin 1: 9, 9, 9, 9
  - Bin 2: 23, 23, 23, 23
  - Bin 3: 29, 29, 29, 29
* Smoothing by **bin medians**: each bin value is replaced by bin median
  - Bin 1: 8.5,8.5,8.5,8.5
  - Bin 2: 22.5,22.5,22.5,22.5
  - Bin 3: 28.5,28.5,28.5,28.5

Binning Methods for Data Smoothing

- Smoothing by **bin boundaries**: minimum and maximum values in given bin are identified as bin boundaries. Each bin value is then replaced by the closest boundary value.

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

Try yourself

1. Use smoothing by bin means to smooth the following data, using a bin depth of 3.

13 15 16 16 19 20 20 21 22 22 25 25 25 25 30 33 33 35 35 35 35 36 40 45 46 52 70

2. Suppose a group of 12 sales price records has been sorted as follows

5 10 11 13 15 35 50 55 72 92 204 215

Partition them into 3 bins by each of the following methods

a. Equal frequency (eqidepth) partitioning
b. Equal width partitioning
c. clustering

Answer

1. Step 1: Sort the data. (This step is not required here as the data are already sorted.)

Step 2: Partition the data into equidepth bins of depth 3.

Bin 1: 13, 15, 16 Bin 2: 16, 19, 20 Bin 3: 20, 21, 22

Bin 4: 22, 25, 25 Bin 5: 25, 25, 30 Bin 6: 33, 33, 35

Bin 7: 35, 35, 35 Bin 8: 36, 40, 45 Bin 9: 46, 52, 70

Step 3: Calculate the arithmetic mean of each bin.

Step 4: Replace each of the values in each bin by the arithmetic mean calculated for the bin.

Bin 1: 142/3, 142/3, 142/3 Bin 2: 181/3, 181/3, 181/3 Bin 3: 21, 21, 21

Bin 4: 24, 24, 24 Bin 5: 262/3, 262/3, 262/3 Bin 6: 332/3, 332/3, 332/3

Bin 7: 35, 35, 35 Bin 8: 401/3, 401/3, 401/3 Bin 9: 56, 56, 56

Answer

2. (a)equal-frequency (equidepth) partitioning

bin 1- 5,10,11,13

bin 2- 15,35,50,55

bin 3 -72,92,204,215

(b) equal-width partitioning

The width of each interval will be (215 – 5)/3 = 70.

bin 1- 5,10,11,13,15,35,50,55,72

bin 2 - 92

bin 3 - 204,215

(c) clustering

- We will use a simple clustering technique, divide the data along the 2 biggest gaps in the data.

bin 1- 5,10,11,13,15

bin 2 - 35,50,55,72,92 bin 3- 204,215

Data Integration

- Combines data from multiple sources into a coherent data store

Issues:

- Entity identification problem:
  - Identify real world entities from multiple data sources.
  - e.g., cust_id in one db and cust_number in other db
- 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., dob – dd/mm/yyyy in one db and mm/dd/yyyy in other db
  - Weight in g and kg
- Redundant attributes

Handling Redundancy in Data Integration

- Redundant data occur often when integration of multiple databases take place
- Redundancy occurs due to
  - The same attribute or object may have different names in different databases. Eg., gender, sex
  - An attribute can be derived from another attribute or set of attributes. Eg., age and dob
- Redundant attributes may be able to be detected by *correlation analysis*

Correlation Analysis

- Correlation between two attributes A and B can be evaluated by computing
  - X^2 (chi-square) value if attributes are categorical
  - Correlation coefficient if attributes are numerical

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

  Expected frequency eij = (count(A=ai)*count(B=bj))/N

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 i.e using formula.
- e11 = count (like science fiction)*count(playchess))/N = (450*300)/1500 = 90

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

Chi-Square Calculation: An Example

- The chi-square statistic tests the hypothesis that A and B are independent.
- The test is based on significance level, with (r-1)*(c-1) degrees of freedom (DoF).
- If the hypothesis is rejected, we say A and B are statistically related or associated.
- Example problem: DoF = (2-1)*(2-1) = 1
- For 1 DoF, the chi-square value to reject the hypothesis at 0.001 significance level is 10.828 (taken from table).
- Since our computed value is above this, we can reject the hypothesis that preferred_reading and preferred_paying are independent.
- It shows that preferred_reading and preferred_paying are correlated for the group of people.

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

Try yourself

1. Calculate correlation coefficient. Are these two variables positively or negatively correlated?

age	23	23	27	27	39	41	47	49	50
%fat	9.5	26.5	7.8	17.8	31.4	25.9	27.4	27.2	31.2
age	52	54	54	56	57	58	58	60	61
%fat	34.6	42.5	28.8	33.4	30.2	34.1	32.9	41.2	35.7

Answer

The *Pearson correlation coefficient* is 0.82, the variables are positively correlated.

Data Transformation

- Data are transformed or consolidated into forms appropriate for mining
- **Methods:**
  - Smoothing:
    - Remove noise from data.
    - Includes binning, regression, clustering.
  - Aggregation:
    - summary or aggregation operations are applied to the data.
    - Eg. Daily sales data may be aggregated to compute monthly total amount.
  - Generalization:
    - Low level primitive raw data are replaced by high-level concepts.
    - Eg. Street generalized to city, age generalized to youth, middle-aged,old

Data Transformation

- Normalization:
  - Attribute values are scaled to fall within a smaller, specified range such as -1.0 to 1.0 or 0.0 to 1.0
  - min-max normalization
  - z-score normalization
  - normalization by decimal scaling
- Attribute construction (feature construction):
  - New attributes are constructed and added from the given set of attributes to help the mining process

Normalization

- **Min-max normalization**:
  - Performa liner transformation on the original data.
  - Let $\min_A$ and $\max_A$ be minimum and maximum values of attribute A.
  - Min-max normalization maps value v of A to v' in the range [new_min_A, new_max_A] by using the formula

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

  - Ex. Let income range \$12,000 to \$98,000 normalized to [0.0, 1.0]. Then \$73,000 is mapped to

$$\frac{73{,}600 - 12{,}000}{98{,}000 - 12{,}000}(1.0 - 0) + 0 = 0.716$$

Normalization

- **Z-score normalization** (zero mean normalization)
  - Attribute values are normalized based on mean and SD of A.
  - μ: mean, σ: standard deviation
  - Value v of A is normalized to v' by using $$v' = \frac{v - \mu_A}{\sigma_A}$$
  - Ex. Let μ = 54,000, σ = 16,000. Then $73,000 is mapped to $$\frac{73{,}600 - 54{,}000}{16{,}000} = 1.225$$
- **Normalization by decimal scaling**
  - Normalizes by moving the decimal point of values of attribute A
  - The number of decimal points moved depends on max absolute value of A.
  - Value v of A is normalized to v' by using $$v' = \frac{v}{10^j}$$

Where j is the smallest integer such that Max(|v'|) < 1

Normalization

- Example:

Suppose values of attribute A range from -986 to 917.

Maximum absolute value of A is 986.

So we divide each value by 1000 (j=3)

Therefore, for value -986

v’ = -0.986

For value 917,

v’ = 0.917

Try yourself

1. Suppose that the data for analysis includes the attribute *age*. The *age* values for the data tuples are (in increasing order) 13, 15, 16, 16, 19, 20, 20, 21, 22, 22, 25, 25, 25, 25, 30, 33, 33, 35, 35, 35, 35, 36, 40, 45, 46, 52, 70.

   (a) Use min-max normalization to transform the value 35 for *age* onto the range [0.0, 1.0].

   (b) Use z-score normalization to transform the value 35 for *age*, where the standard deviation of *age* is 12.94 years.

   (c) Use normalization by decimal scaling to transform the value 35 for *age*.

2. Normalize the two variables based on z-score normalization.

age	23	23	27	27	39	41	47	49	50
%fat	9.5	26.5	7.8	17.8	31.4	25.9	27.4	27.2	31.2

age	52	54	54	56	57	58	58	60	61
%fat	34.6	42.5	28.8	33.4	30.2	34.1	32.9	41.2	35.7

Answer

1. (a) Use min-max normalization to transform the value 35 for *age* onto the range $[0.0, 1.0]$.
Using the corresponding equation with $min_A = 13$, $max_A = 70$, $new\_min_A = 0$, $new\_max_A = 1.0$, then $v = 35$ is transformed to $v' = 0.39$.

   (b) Use z-score normalization to transform the value 35 for *age*, where the standard deviation of *age* is 12.94 years.
Using the corresponding equation where $A = 809/27 = 29.96$ and $\sigma_A = 12.94$, then $v = 35$ is transformed to $v' = 0.39$.

   (c) Use normalization by decimal scaling to transform the value 35 for *age*.
Using the corresponding equation where $j = 2$, $v = 35$ is transformed to $v' = 0.35$.

2.

age	23	23	27	27	39	41	47	49	50
z-age	-1.83	-1.83	-1.51	-1.51	-0.58	-0.42	0.04	0.20	0.28
%fat	9.5	26.5	7.8	17.8	31.4	25.9	27.4	27.2	31.2
z-%fat	-2.14	-0.25	-2.33	-1.22	0.29	-0.32	-0.15	-0.18	0.27

age	52	54	54	56	57	58	58	60	61
z-age	0.43	0.59	0.59	0.74	0.82	0.90	0.90	1.06	1.13
%fat	34.6	42.5	28.8	33.4	30.2	34.1	32.9	41.2	35.7
z-%fat	0.65	1.53	0.0	0.51	0.16	0.59	0.46	1.38	0.77

Data Reduction

- **Data reduction**:

  Obtain a reduced representation of the data set that is much smaller in volume but yet produces the same (or almost the same) analytical results

- Why data reduction?

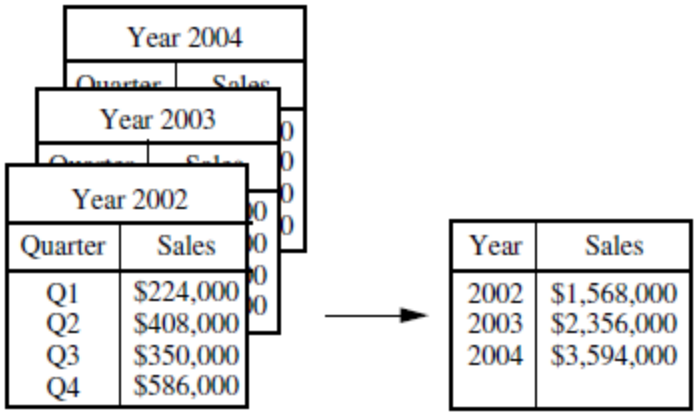
  A database/data warehouse may store terabytes of data. Complex data analysis may take a very long time to run on the complete data set.

Data Reduction Strategies

- Data cube aggregation
- Attribute subset selection
- Dimensionality reduction
- Discretization and concept hierarchy generation

Data cube aggregation

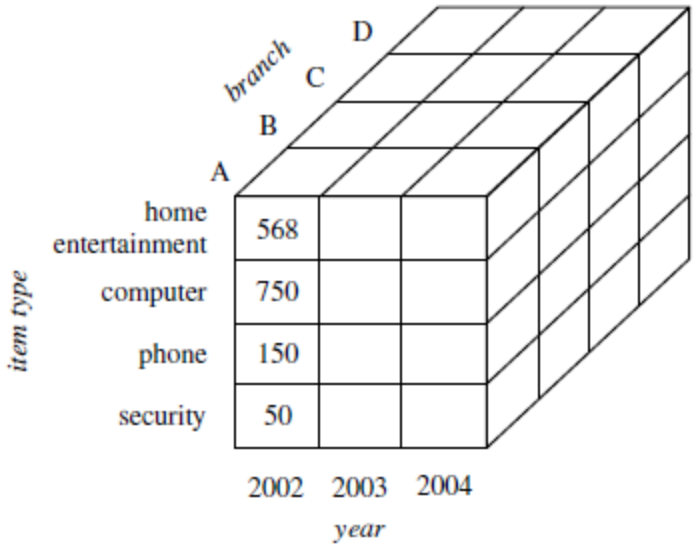
- Aggregation operations are applied to the data in the construction of data cube.
  - Example – suppose data consist of sales per quarter details and you are interested in the annual sales. Data can be aggregated to obtain total sales per year



Sales data for a given branch of *AllElectronics* for the years 2002 to 2004. On the left, the sales are shown per quarter. On the right, the data are aggregated to provide the annual sales.

Data cube aggregation

Data cube – store multi-dimensional aggregated information



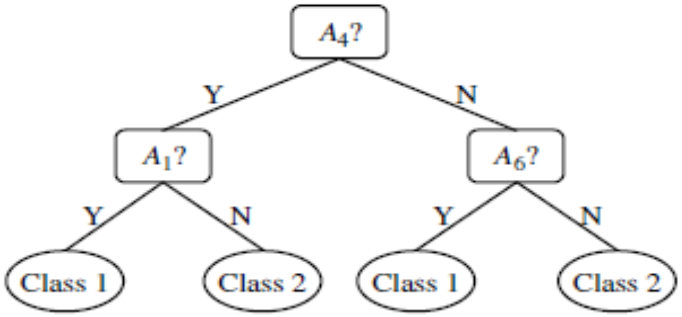
A data cube for sales at *AllElectronics*.

Attribute subset selection

Irrelevant, weakly relevant or redundant attributes are detected and removed.

Techniques:

1. Stepwise forward selection – adds best attribute in each step
2. Stepwise backward selection – removes worst attribute in each step
3. Decision tree induction – constructs tree. Internal node(test on attribute), branch(out come of test), external node (class prediction)

Forward selection	Backward elimination	Decision tree induction
Initial attribute set: $\{A_1, A_2, A_3, A_4, A_5, A_6\}$ Initial reduced set: $\{\}$ $\Rightarrow \{A_1\}$ $\Rightarrow \{A_1, A_4\}$ $\Rightarrow$ Reduced attribute set: $\{A_1, A_4, A_6\}$	Initial attribute set: $\{A_1, A_2, A_3, A_4, A_5, A_6\}$ $\Rightarrow \{A_1, A_3, A_4, A_5, A_6\}$ $\Rightarrow \{A_1, A_4, A_5, A_6\}$ $\Rightarrow$ Reduced attribute set: $\{A_1, A_4, A_6\}$	Initial attribute set: $\{A_1, A_2, A_3, A_4, A_5, A_6\}$  $\Rightarrow$ Reduced attribute set: $\{A_1, A_4, A_6\}$

Dimensionality reduction

Encoding mechanisms are used to reduce the data set size (i.e) gives compressed representation of the original data.

Methods:

1. Lossless – original data can be reconstructed from the compressed data without any loss of information
2. Lossy – approximation of original data is constructed from original data.

Example techniques:

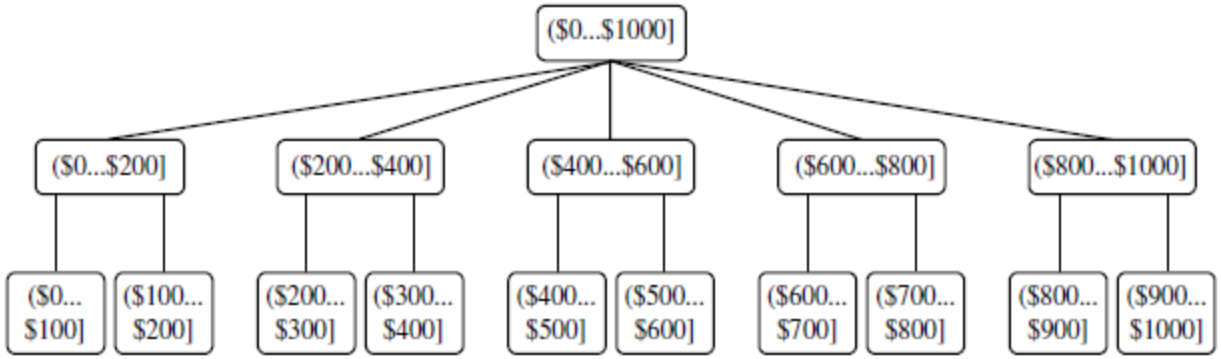
- Wavelet transforms
- Principal Components Analysis (PCA)

Dimensionality Reduction

- **Curse of dimensionality**
  - When dimensionality increases, data becomes increasingly sparse
  - Density and distance between points, which is critical to clustering, 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

Discretization and concept hierarchy generation

- Raw data values for attributes are replaced by ranges or higher conceptual levels.
- It is a type of numerosity reduction



A concept hierarchy for the attribute *price*, where an interval ($X ... $Y] denotes the range from $X (exclusive) to $Y (inclusive).