COS10022 – DATA SCIENCE PRINCIPLES

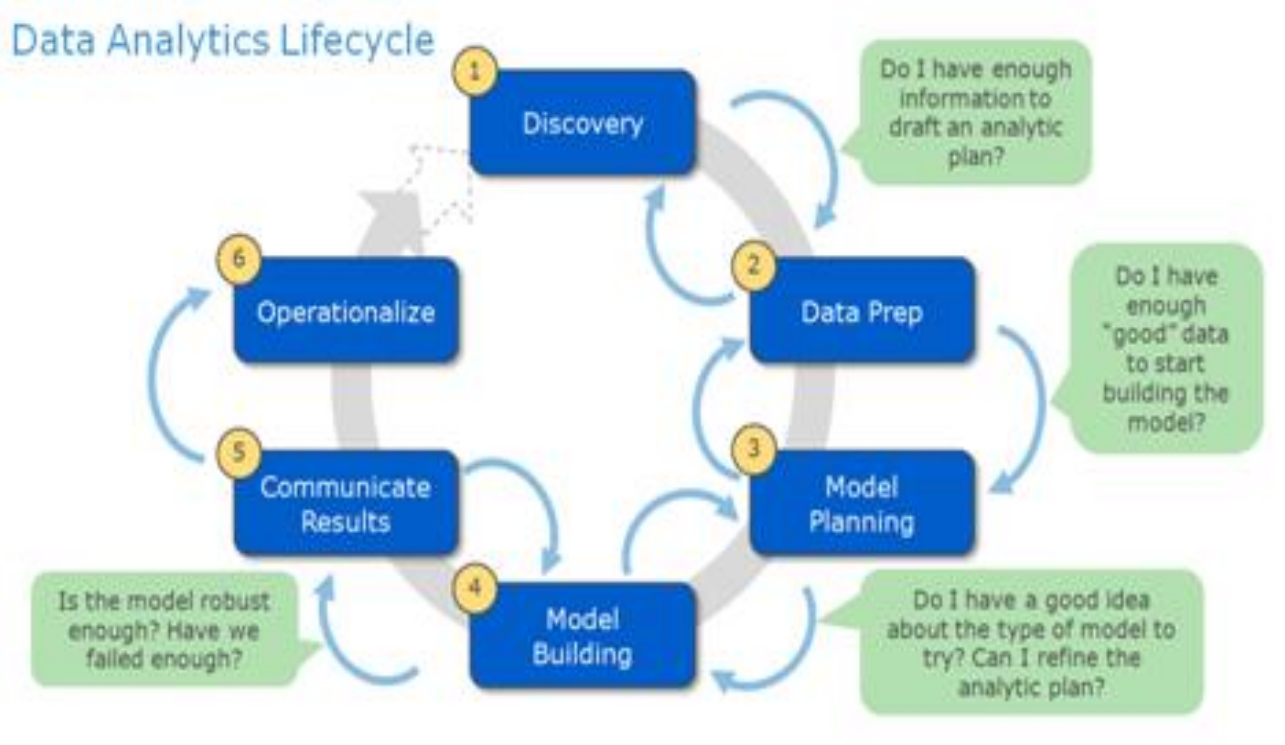
Dr Pei-Wei Tsai (Lecturer, Unit Convenor)
ptsai@swin.edu.au, EN508d

SWINBURNE UNIVERSITY OF TECHNOLOGY

Week 08
Data Preparation

COS10022 Data science Principles

Data Analytics Lifecycle



Phase 2: Data Preparation

Given the presence of an analytics sandbox, the data science team-work with data and perform analytics for the duration of the project. The team performs ETLT to get the data into the sandbox and familiarize themselves with the data thoroughly.

Outline

- **OVERVIEW**
  - **Data Quality**
  - **Major Tasks in Data Preparation**
- Data Cleaning
- Data Integration
- Data Reduction
- Data Transformation

Why is Data Preparation Important?

- **Data** have **quality** if they satisfy the requirements of the intended use.
- Factors comprising data quality:

Accuracy	Completeness	Consistency	Timeliness	Believability	Interpretability
• Degree to which the data **represents the reality.**	• Degree to which necessary data is **available for use.**	• Degree to which the data is **equal within and between datasets**.	• Degree to which the data is **available at the time it is needed**.	• Degree to which the data is **trusted by users.**	• Degree to which the data can be **easily understood.**

- Data Preparation is sometimes called Data Wrangling or Data Munging.

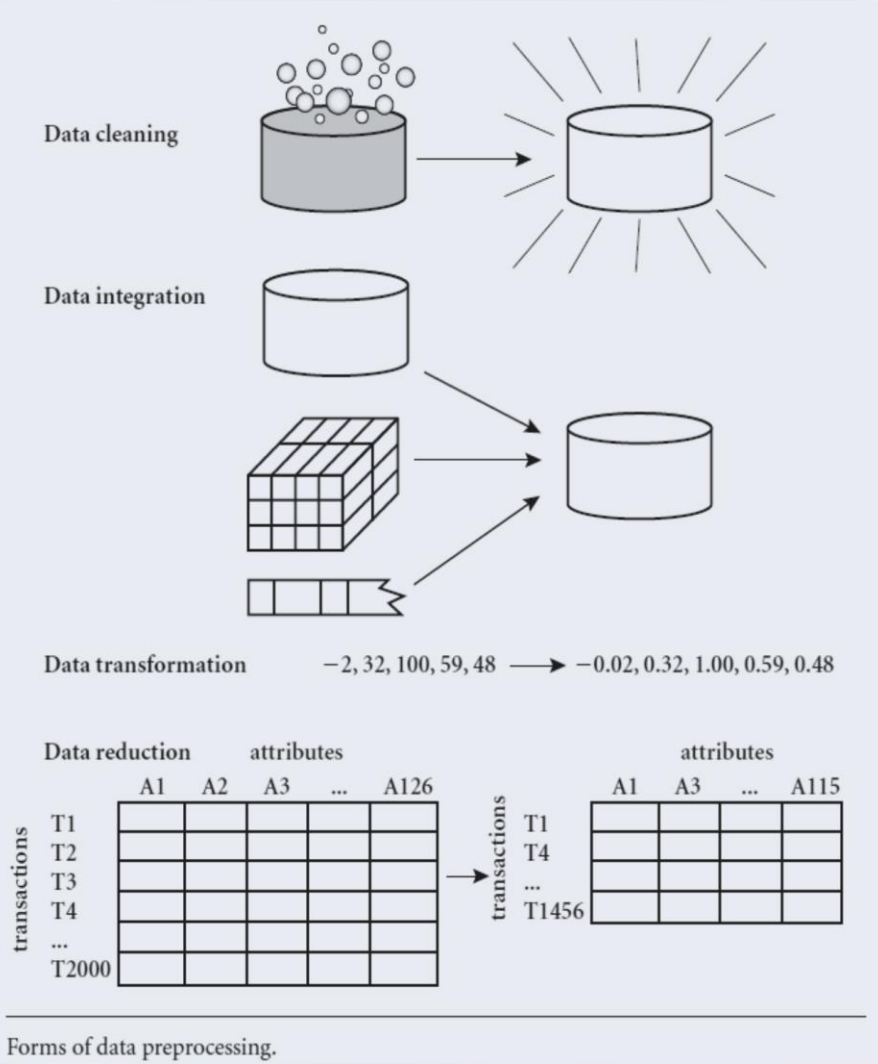
Major Tasks in Data Preparation

Data Cleaning

To fill in missing values, smooth noisy data, identify or remove outliers, and resolve inconsistencies.

Data Transformation

To modify the source data into different formats in terms of data types and values so that it is useful for mining and to make the output easier to understand.



Forms of data preprocessing.

Data Integration

To merge data from multiple data stores to help reduce redundancies and inconsistencies in the resulting dataset.

Data Reduction

To obtain a reduced representation of the dataset that is much smaller in volume, yet produces the same (or almost the same(analytical results.

Outline

- Overview
  - Data Quality
  - Major Tasks in Data Preprocessing
- **DATA CLEANING**
- Data Integration
- Data Reduction
- Data Transformation

Data Cleaning

Real-world data is **DIRTY**.

1. **Incomplete Data**:
   - Missing attribute values, lacking certain attributes of interest, or containing only aggregate data
   - E.g. Occupation = “”
2. **Noisy Data:**
   - Containing errors or outliers
   - E.g. Salary = “-100”
3. **Inconsistent Data:**
   - Containing discrepancies in codes or names
   - E.g. Discrepancy between duplicate records
   - E.g. Was rating “1, 2, 3”, Now rating “A, B, C”
   - E.g. Age = “36”, Birthday = “31/08/1984”

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	10000K	Yes
6	No	NULL	60K	No
7	Yes	Divorced	220K	NULL
8	No	Single	85K	Yes
9	No	Married	90K	No
9	No	Single	90K	No

A typo or a millionaire?

Missing values

Inconsistent duplicate entries

Incomplete Data

- **Incomplete data can occur for a number of reasons**:
  - Attributes of interest may not always be available.
  - Relevant data may not be recorded:
    - Because they were not considered important at the time of entry
    - Due to misunderstanding or equipment malfunctions.
  - Data that were inconsistent with other recorded data may have been deleted.
  - The recording of the data history or modifications may have been overlooked.

- **Missing data, particularly for tuples with missing values for some attributes, may need to be inferred.**

First Name	Gender	Age
Jason	M	22
May	F	17
Olivia	F	36
David		28

Incomplete Data

Class Label

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	NULL
4	Yes	Married	120K	No
5	No	Divorced	10000K	Yes

- **How to Handle Missing Data?**

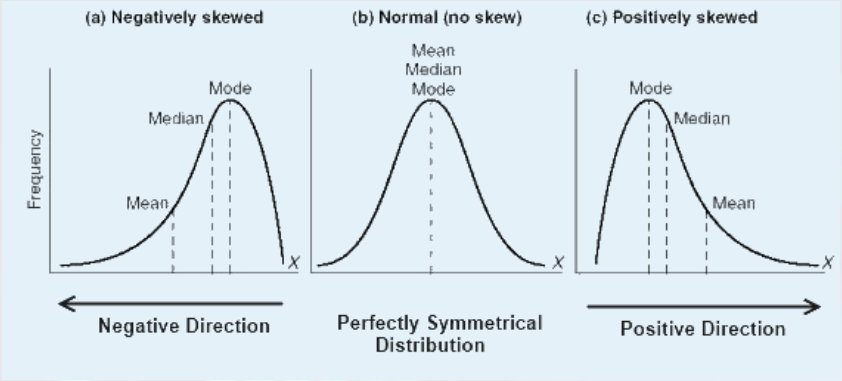
1. **Ignore the tuple**
   - This is usually done when **class label** is missing (assuming the mining task involves classification).
   - This method is not very effective, unless the tuple contains several attributes with missing values.

2. **Fill in the missing values with side information**
   - This method is **time consuming** and may not be feasible given a large dataset with many missing values.

3. **Use a global constant to fill in the missing values**
   - Replace all missing attribute values using the **same constant** (such as “Unknown”, “N/A”).
   - A mining program may mistakenly think that they form an interesting concept since they all have a value in common.

Incomplete Data

- **How to Handle Missing Data?**
  4. **Use a measure of central tendency for the attribute (e.g. the mean or medium) to fill in the missing values**
     - For normal data distributions, the mean can be used, while skewed data distribution should employ the median.
  5. **Use the attribute mean or medium for all samples belonging to the same class as the given tuples**
     - E.g. If classifying customers according to *credit risk*, replace the missing value with the average *income* value for customers in the same credit risk category as that of the given tuple.



Mean (Download Speed) = 130

Mobile ID	Mobile Package	Download Speed	Data Limit Usage
1	Fast+	157	80%
2	Lite	99	70%
3	Fast+	167	10%
4	Fast+	N/A	80%
5	Lite	76	70%
6	Fast+	155	10%
7	Fast+	N/A	95%
8	Lite	76	77%
9	Fast+	180	95%

Mobile ID	Mobile Package	Download Speed	Data Limit Usage
1	Fast+	157	80%
2	Lite	99	70%
3	Fast+	167	10%
4	Fast+	130	80%
5	Lite	76	70%
6	Fast+	155	10%
7	Fast+	130	95%
8	Lite	76	77%
9	Fast+	180	95%

Median (Download Speed) = 155

Mobile ID	Mobile Package	Download Speed	Data Limit Usage
1	Fast+	157	80%
2	Lite	99	70%
3	Fast+	167	10%
4	Fast+	N/A	80%
5	Lite	76	70%
6	Fast+	155	10%
7	Fast+	N/A	95%
8	Lite	76	77%
9	Fast+	180	95%

Mobile ID	Mobile Package	Download Speed	Data Limit Usage
1	Fast+	157	80%
2	Lite	99	70%
3	Fast+	167	10%
4	Fast+	155	80%
5	Lite	76	70%
6	Fast+	155	10%
7	Fast+	155	95%
8	Lite	76	77%
9	Fast+	180	95%

Incomplete Data

Mobile ID	Mobile Package	Download Speed	Data Limit Usage
1	Fast+	157	80%
2	Lite	99	70%
3	Fast+	167	10%
4	Fast+	N/A	80%
5	Lite	76	70%
6	Fast+	155	10%
7	Fast+	N/A	95%
8	Lite	76	77%
9	Fast+	180	95%

Mobile ID	Mobile Package	Download Speed	Data Limit Usage
1	Fast+	157	80%
3	Fast+	167	10%
4	Fast+	N/A	80%
6	Fast+	155	10%
7	Fast+	N/A	95%
9	Fast+	180	95%

Mobile ID	Mobile Package	Download Speed	Data Limit Usage
2	Lite	99	70%
5	Lite	76	70%
8	Lite	76	77%

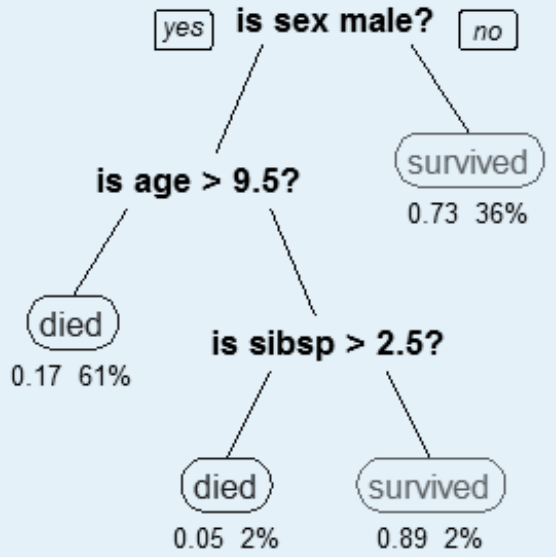
$$Mean_{Fast+} = 165$$

Replace the N/A with 165 in this example.

Incomplete Data

- **How to Handle Missing Data?**
  6. **Use the most probable value to fill in the missing value**
     - This may be determined with regression, interference-based tools using Bayesian formalism or decision tree induction.
     - E.g. Using the other passenger attributes in the *titanic* dataset, you may construct a decision tree to predict the **missing values** for ***sibsp*** *(Number of Siblings/Spouses Aboard).*

Sex	Age	Sibsp	Survived
Male	22	1	0
Female	38	1	1
Male	2	4	0
Male	5	1	1
Female	16	5	0
Male	1	?	0



Noisy Data

Noise is a random error or variance in a measured variable.

We can use the methods listed below to calibrate the data:

- Binning
- Regression
- Sliding Window (Moving Average)
- Outlier Analysis

Noisy Data

- **How to Handle Noisy Data?**
  1. **Binning (Data smoothing)**
     - This method smooth a sorted data value by consulting its “neighborhood”, that is, the values around it.
     - The sorted values are distributed into a number of “buckets”, or “bins”.

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

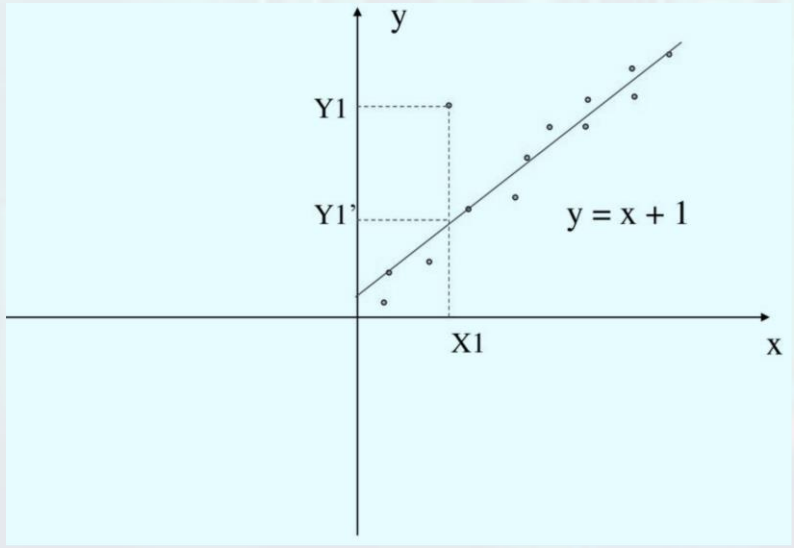
→ The data for price are first sorted and then partitioned into equal frequency bins of size 3 (i.e. each bin contains 4 values).

→ Each original value in a bin is replaced by the mean value of the bin (i.e. the value 9).

→ The min. and max. values in a given bin are identified. Each bin value is then replaced by the closed boundary value.

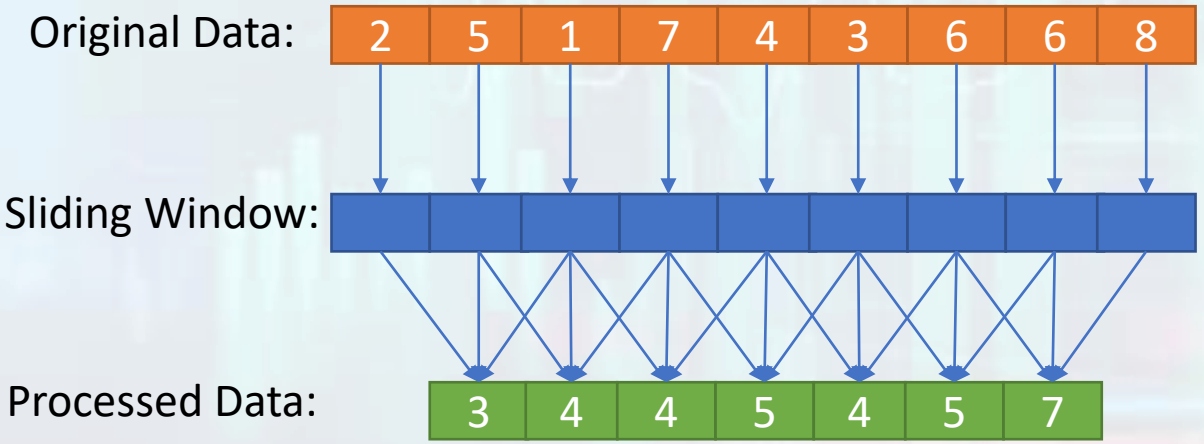
Noisy Data

- **How to Handle Noisy Data?**
  2. **Regression**
     - A technique that conforms data values to a function.
     - E.g. Linear regression involves finding the “best” line to fit two attributes so that one attribute can be used to predict the other.



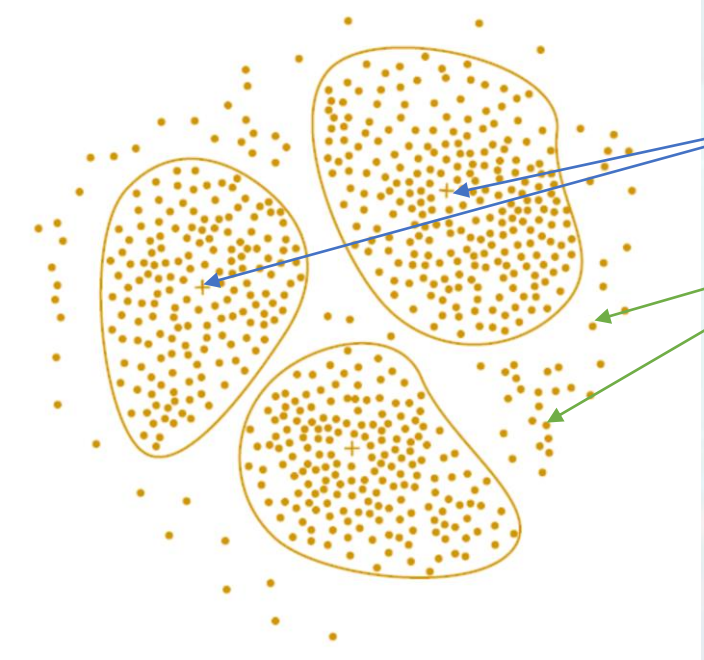
Noisy Data

- **How to Handle Noisy Data?**
  3. **Sliding Window (Moving Average) (Convolution)**
     - Using the neighbourhood data to find the average.
     - Slides through the whole data in sequence.



Noisy Data

- **How to Handle Noisy Data?**
  4. **Outlier analysis**
     - Outliers may be detected by clustering, for example, where similar values are organized into groups, or “clusters”.



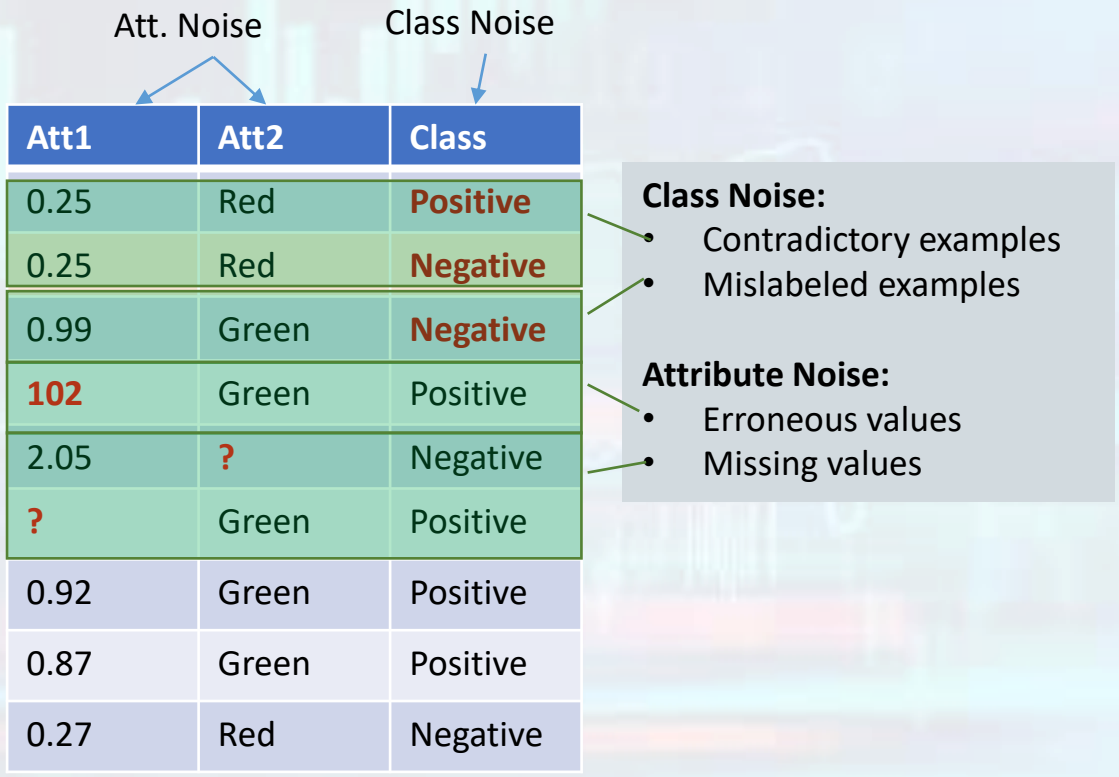
Each cluster centroid is marked with a “+”, representing the average point in space for that cluster.

Outliers may be detected as values that fall outside of the sets of clusters.

Fig. A 2-D plot of customer data with respect to customer locations in a city, showing three data clusters.

Noisy Data

- **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
  - inconsistent data



Data Cleaning as a Process

- The first step in data cleaning as a process is **discrepancy detection.**

- **Discrepancies** can be caused by:
  - Poorly designed data entry forms
  - Human errors in data entry
  - Deliberate errors
    - e.g., respondents not wanting to divulge information about themselves
  - Data decay
    - e.g., outdated addresses
  - Errors in instrumentation devices that record data
  - System errors
  - Inconsistencies due to data integration
    - e.g., where a given attribute can have different names in different databases

Data Cleaning as a Process

- **How to Detect Data Discrepancies?**
  1. **Metadata**
     - Use any knowledge that you may already have regarding properties of the data
     - E.g., What are acceptable values for each attribute? Do all values fall within the expected range? What are data type and domain of each attribute?
  2. **Check uniqueness rule, consecutive rule and null rule**
     - **Unique rule**: Each value of the given attribute must be different from all other values for that attribute
     - **Consecutive rule**: There can be no missing values between the lowest and highest values for the attribute, and that all values must also be unique.
     - **Null rule**: Specifies the use of blanks, question marks, special characters, or other strings that may indicate the null condition.

Data Cleaning as a Process

- **How to Detect Data Discrepancies? (Example)**

Unique Rule

UID	Gender	Height	Weight	Sex	DoB
0001	M	167	?	M	…
0002	F	158	52	F	…
0174	F	176	63	F	…
0001	M	181	88	F	…
0005	M	174	76	M	…

Consecutive Rule

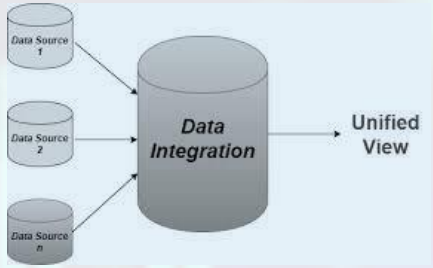
Data Cleaning as a Process

- **How to Detect Data Discrepancies?**
  3. **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)
     - For example: Microsoft Power BI. https://youtu.be/yKTSLffVGbk
  4. **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

Outline

- Overview
  - Data Quality
  - Major Tasks in Data Preprocessing
- Data Cleaning
- **DATA INTEGRATION**
- Data Reduction
- Data Transformation

Data Integration



- **Data integration** combines data from multiple sources (multiple databases, data cubes, or flat files) into a coherent store, as in data warehousing.

- How can equivalent real-world entities from multiple data sources be matched up? This is referred to as the **Entity Identification Problem.**
  - E.g.: Bill Clinton = William Clinton
  - E.g.: *customer_id* in one database = *cust_number* in another database

- Data integration can help detect and resolve **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)

Data Integration

- **Redundant data** often occurs when integrating 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 (I.e., annual revenue)

- Redundant attributes may be able to be detected by **correlation analysis.**
  - The analysis measure how strongly one attribute implies the other, based on the available data.
  - For categorical data, ***X^2 (Chi-Square)* test** is used.
  - For numerical data, ***Correlation Coefficient** and **Covariance*** are used.

Data Integration

- **How to Detect Redundant Attributes?**
  1. **Correlation Coefficient (*r*) for Numerical Data**
     - Also called Pearson's Product Moment Coefficient.

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

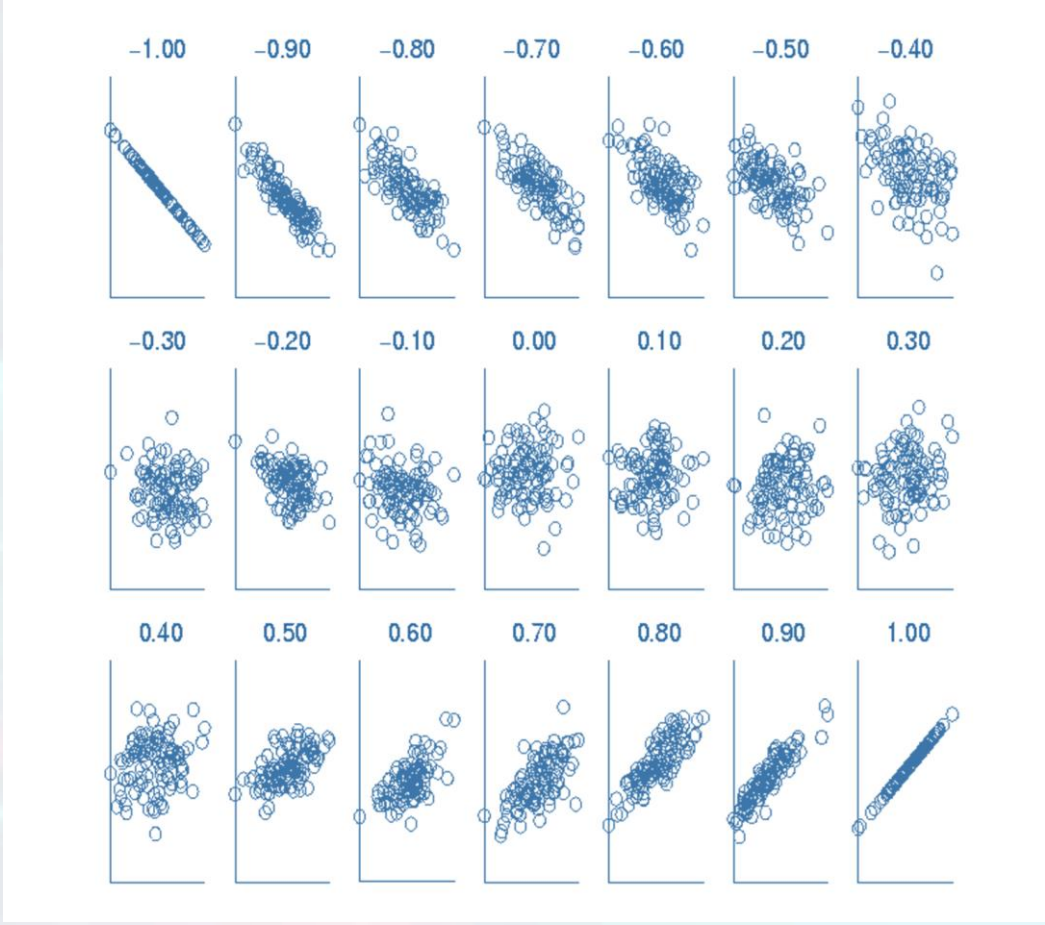
where n is the number of tuples, $\overline{A}$ and $\overline{B}$ are the respective means of A and B, σ_A and σ_B are the respective standard deviation of A and B, and $\Sigma(a_i b_i)$ is the sum of the AB cross-product.

- **$r_{A,B} > 0$: Positively correlated, (0, 1]**
- **$r_{A,B} = 0$: Independent, 0**
- **$r_{AB} < 0$: Negatively correlated, [-1, 0)**

- If you are interested in how this is calculated, you can watch this video online: https://www.youtube.com/watch?v=11c9cs6WpJU

Data Integration

- Visually evaluating correlation using scatter plots
- Scatter plots showing the correlation coefficient from -1 to 1.
  - *r* = 1.0 : A **perfect** positive relationship
  - *r* = 0.8 : A **fairly strong** positive relationship
  - *r* = 0.6 : A **moderate** positive relationship
  - *r* = 0.0 : **No relationship**
  - *r* = -1.0 : A **perfect** negative relationship



Data Integration

- **How to Detect Redundant Attributes?**
  - **2. Covariance (*Cov*) for Numerical Data**
    - Consider two numeric attributes *A* and *B*, and a set of *n* observations $\{(a_1, b_1), \ldots, (a_n, b_n)\}$.
    - The mean values of *A* and *B*, are also known as the **expected values** of *A* and *B*, that is:

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

    - The covariance between A and B is defined as:

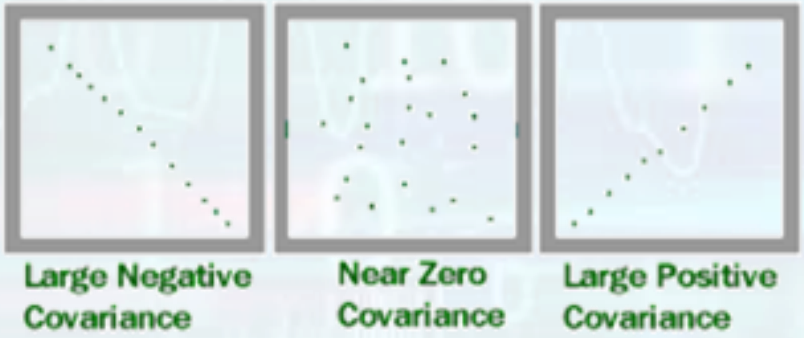
$$Cov(A, B) = E((A - \bar{A})(B - \bar{B})) = \frac{\sum_{i=1}^{n}(a_i - \bar{A})(b_i - \bar{B})}{n}$$

    - It can be simplified in computation as:

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

Data Integration

- Visually evaluating covariance between two variables using scatter plot.
  - $Cov(A, B) < 0$: A and B tend to move in **opposite** direction
  - $Cov(A, B) > 0$: A and B tend to move in **the same** direction
  - $Cov(A, B) = 0$: A and B are **independent**.
    - Note that: Zero covariance are not necessarily mean that the variables are independent. A non-linear relationship can exist that still would result in covariance value of zero.



Data Integration

EXAMPLE

The table below presents a simplified example of stock prices observed at five time points for *AllElectronics* and *HighTech*, some high tech company. If the stocks are affected by the same industry trends, will their price rise or fall together?

Time point	*AllElectronics*	*HighTech*
t1	6	20
t2	5	10
t3	4	14
t4	3	5
t5	2	5

$$E(AllElectronics) = \frac{6+5+4+3+2}{5} = \frac{20}{5} = \$4$$

$$E(HighTech) = \frac{20+10+14+5+5}{5} = \frac{54}{5} = \$10.8$$

$$\begin{aligned} Cov(AllElectroncis, HighTech) &= \frac{6\times20+5\times10+4\times14+3\times5+2\times5}{5} - 4 \times 10.8 \\ &= 50.2 - 43.2 = 7. \end{aligned}$$

Cov(*AllElectronics*, *HighTech*) **> 0**, the stock prices for both companies **rise together**.

Data Integration

- **How to Detect Redundant Attributes?**
  3. **Chi-Squared (X^2) test for Categorical Data**
     - 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.

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

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

Where:
Attribute A has *c* distinct values;
Attribute B has *r* distinct values;
e_{ij} is the expected frequency;
o_{ij} is the observed frequency;

Data Integration

EXAMPLE

Suppose that a group of 1500 people was surveyed. The gender of each person was noted. Each person was polled as to whether his or her preferred type of reading material was fiction or nonfiction. Thus, we have two attributes, *gender* and *preferred reading*. The observed frequency (or count) of each possible joint event is summarized in the contingency table

	male	*female*	*Total*
fiction	250 (90)	200 (360)	450
non_fiction	50 (210)	1000 (840)	1050
Total	300	1200	1500

Note: Are *gender* and *preferred_reading* correlated?

$$e_{11} = \frac{count(male) \times count(fiction)}{n} = \frac{300 \times 450}{1500} = 90$$

$$\chi^2 = \frac{(250-90)^2}{90} + \frac{(50-210)^2}{210} + \frac{(200-360)^2}{360} + \frac{(1000-840)^2}{840}$$
$$= 284.44 + 121.90 + 71.11 + 30.48 = 507.93.$$

Data Integration

EXAMPLE (Cont.)

Hypothesis: Gender and preferred reading are independent.

The **degree of freedom** is:
(r-1)(c-1) = (2-1)(2-1) = 1.

Result: 507.83 > 10.83
So hypothesis is rejected.

Conclusion: *Gender* and *preferred reading* are strongly correlated.

For 1-degree of freedom, the X^2 value needed to reject the hypothesis at the 0.001 significant level is 10.83.

Degrees of freedom (df)	X^2 value										
1	0.004	0.02	0.06	0.15	0.46	1.07	1.64	2.71	3.84	6.64	10.83
2	0.10	0.21	0.45	0.71	1.39	2.41	3.22	4.60	5.99	9.21	13.82
3	0.35	0.58	1.01	1.42	2.37	3.66	4.64	6.25	7.82	11.34	16.27
4	0.71	1.06	1.65	2.20	3.36	4.88	5.99	7.78	9.49	13.28	18.47
5	1.14	1.61	2.34	3.00	4.35	6.06	7.29	9.24	11.07	15.09	20.52
6	1.63	2.20	3.07	3.83	5.35	7.23	8.56	10.64	12.59	16.81	22.46
7	2.17	2.83	3.82	4.67	6.35	8.38	9.80	12.02	14.07	18.48	24.32
8	2.73	3.49	4.59	5.53	7.34	9.52	11.03	13.36	15.51	20.09	26.12
9	3.32	4.17	5.38	6.39	8.34	10.66	12.24	14.68	16.92	21.67	27.88
10	3.94	4.86	6.18	7.27	9.34	11.78	13.44	15.99	18.31	23.21	29.59
P value (Probability)	0.95	0.90	0.80	0.70	0.50	0.30	0.20	0.10	0.05	0.01	0.001
	Non-significant								Significant		

Outline

- Overview
  - Data Quality
  - Major Tasks in Data Preprocessing
- Data Cleaning
- Data Integration
- **DATA REDUCTION**
- Data Transformation

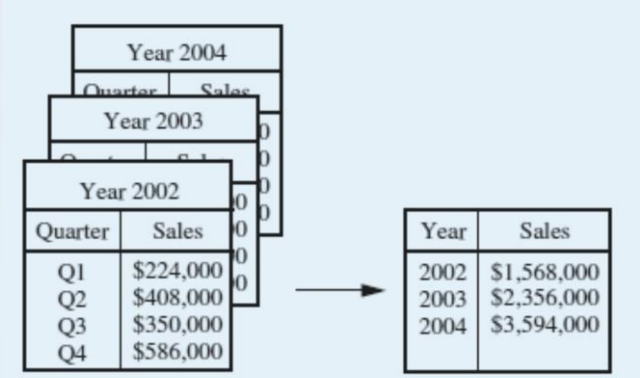
Data Reduction

- **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
- **What is 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

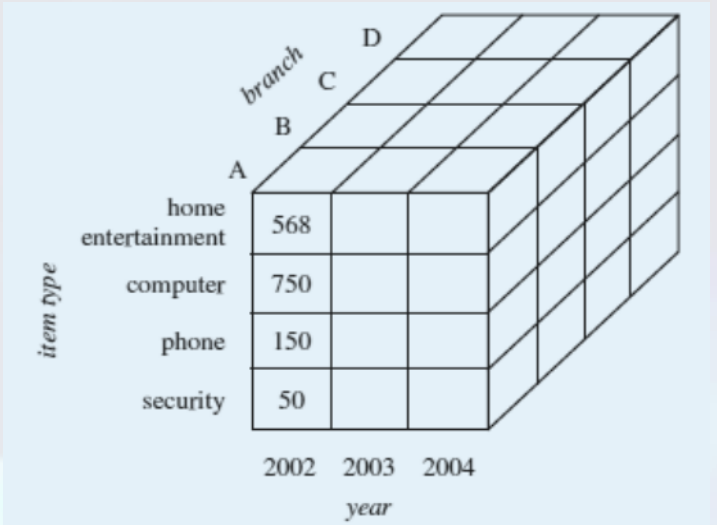
Data Reduction Strategies

1. **Data cube aggregation**
   - Aggregation operations are applied to the data in the construction of a data cube.
2. **Attribute subset selection**
   - Irrelevant, weakly relevant, or redundant attributes or dimensions may be detected and removed.
3. **Dimensionality reduction**
   - Encoding mechanisms are used to reduce the data set size.
4. **Numerosity reduction**
   - The data are replaced or estimated by alternative, smaller data representations
5. **Discretization and concept hierarchy generation**
   - Raw data values for attributes are replaced by ranges or higher conceptual levels.

Data Cube Aggregation



- These data consist of the *AllEletronic* **sales per quarter**, for the years 2002 to 2004.
- The data can be aggregated so that the resulting data **summarize** the **total sales per year** instead of per quarter.
- The resulting dataset is **smaller in volume**, without loss of information necessary for the analysis task.



- **Data cubes** store multidimensional analysis of sales data with respect to annual sales per item type for each *AllElectronic* branch.
  - Each cell holds **an aggregate data value**, corresponding to the data point in multidimensional space.
  - Data cubes provide fast access to precomputed, summarized data, thereby benefiting **on-line analytical processing** as well as data mining.

Attribute Subset Selection

- Attribute subset selection reduces the data set size by removing irrelevant or redundant attributes.
- Heuristic methods that explore a reduced search space are commonly used to find a ‘good’ subset of the original attributes.
  - **Stepwise forward selection**
  - **Stepwise backward elimination**
  - **Combination of forward selection and backward elimination**
  - **Decision tree induction**
- **The “best” (and “worst”) attributes are typically determined using tests of statistical significance, which assume that the attributes are independent of one another.**
- **Other attribute evaluation measures such as information gain is used in building decision trees for classification.**

Attribute Subset Selection

Stepwise forward selection

1. Start with an empty set of attributes
2. Determine the best of the original attributes and add it to the reduced set.
3. At each step, add the best of the remaining original attributes to the reduced set.

Forward selection	Backward elimination
Initial attribute set: $\{A_1, A_2, A_3, A_4, A_5, A_6\}$ Initial reduced set: $\{\}$ => $\{A_1\}$ => $\{A_1, A_4\}$ => Reduced attribute set: $\{A_1, A_4, A_6\}$	Initial attribute set: $\{A_1, A_2, A_3, A_4, A_5, A_6\}$ => $\{A_1, A_3, A_4, A_5, A_6\}$ => $\{A_1, A_4, A_5, A_6\}$ => Reduced attribute set: $\{A_1, A_4, A_6\}$

Stepwise backward elimination

1. Start with the full set of attributes
2. At each step, remove the worst attribute remaining in the set.

Forward selection + Backward elimination

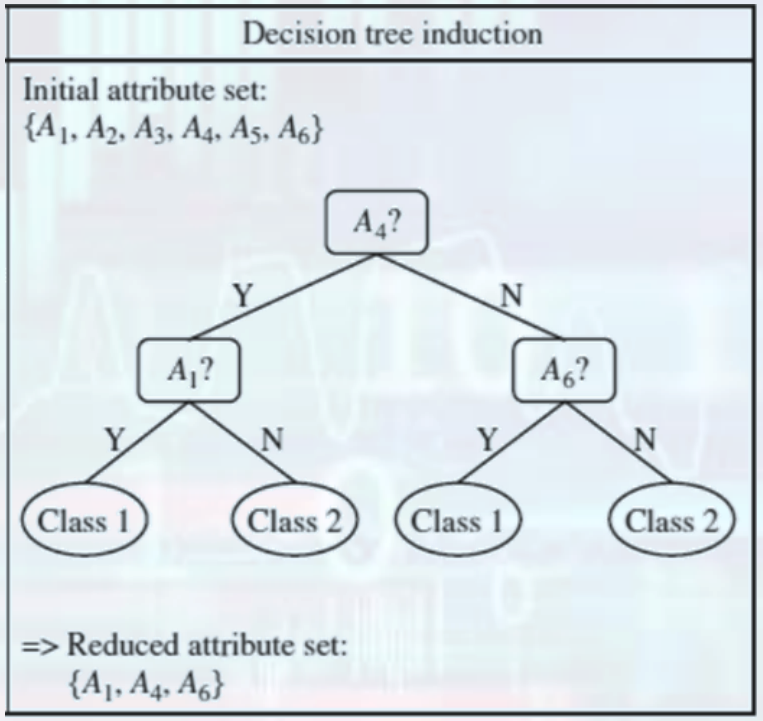
1. Start with an empty set of attributes
2. At each step, add the best attribute to the reduced set and removes the worst from among the remaining attributes.

Attribute Subset Selection

Decision Tree Induction

Decision tree induction constructs a flowchart-like structure where each internal (nonleaf) node denotes a test on an attribute, each branch corresponds to an outcome of the test, and each external (leaf) node denotes a class prediction. At each node, the algorithm chooses the **"best"** attribute to partition the data into individual classes.

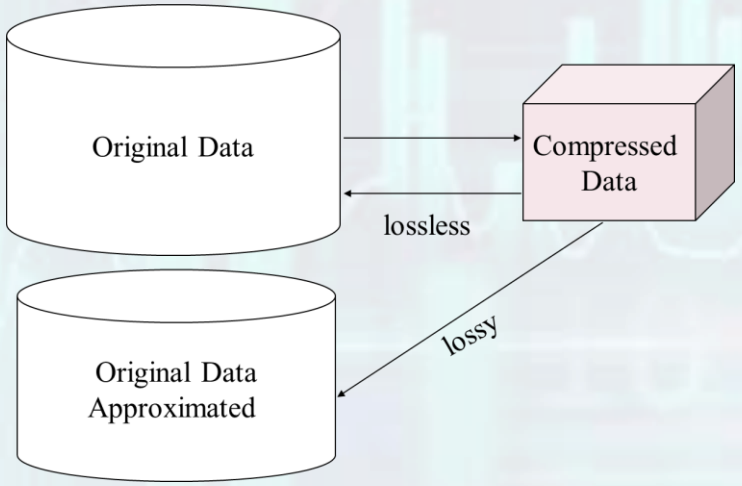
When decision tree induction is used for attribute subset selection, a tree is constructed from the given data. **All attributes that do not appear in the tree are assumed to be irrelevant.** The set of attributes appearing in the tree form the reduced subset of attributes.



Dimensional Reduction

- In dimensionality reduction, data encoding or transformations are applied to obtain a reduced or **“compressed”** representation of the original data.

If the original data can be reconstructed fro the compressed data without any loss of information, the data reduction is called **lossless.**



If only an approximation of the original data can be reconstructed from the compressed data, the data reduction is called **lossy**.

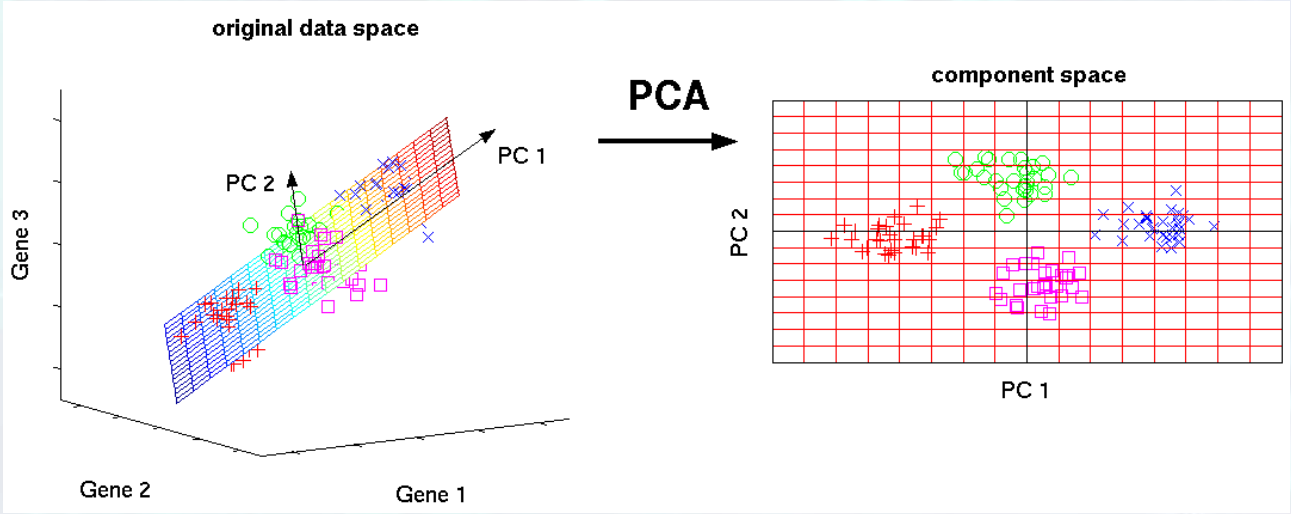
- An example of dimensional reduction method: **Principal Component Analysis (PCA)**
  - PCA main ideas in 5 minutes: https://www.youtube.com/watch?v=HMOI_lkzW08

Dimensional Reduction

- Principal Component Analysis (PCA) reduces the dimensionality (the number of features) of a dataset by maintaining as much variance as possible.

- Example:

  Gene Expression



- The original expression by 3 genes is projected to two new dimensions. Such two-dimensional visualization of the samples allow us to draw qualitative conclusions about the separability of experimental conditions (marked by different colors).

Numerosity Reduction

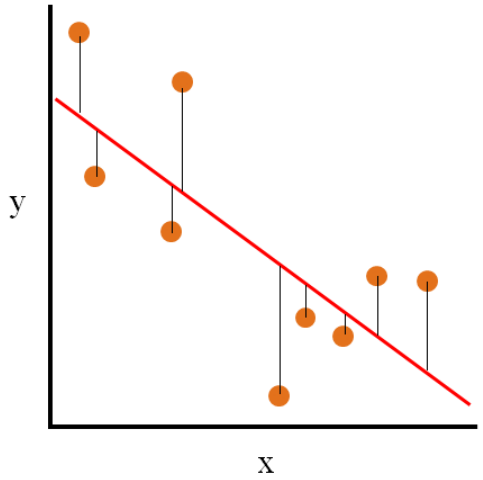
- Numerosity reduction techniques replace the original data volume by choosing alternative, smaller forms of data representation.

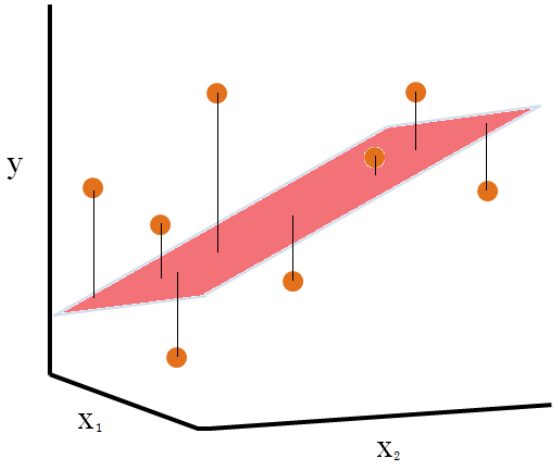
- **Parametric methods**
  - These methods assume that the data fits some models.
  - Models such as **regression** and **log-linear** model are used to estimate the data, so that only the data parameters need to be stored, instead of the actual data.

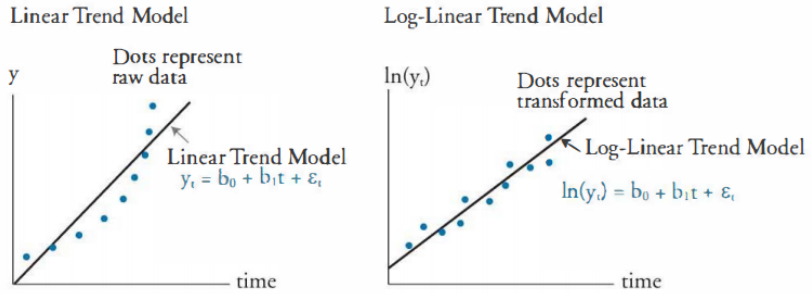
- **Non-parametric methods**
  - These methods do not assume models.
  - Methods such as **histogram**, **clustering**, **sampling** and **data cube aggregation** are used to store reduced representations of data

Numerosity Reduction: Parametric Method

Linear Regression	Multiple Linear Regression	Log-Linear Model
The data are modelled to fit a straight line. The least-square method is used to fit the line.	MLR allows a response variable Y to be modelled as a linear function of two or more predictor variables.	The model takes the form of a function whose logarithm is a linear combination of the parameters of the model.
$Y = b_0 + b_1 X_1$	$Y = b_0 + b_1 X_1 + b_2 X_2$	$\ln Y = b_0 + b_1 X_1 + \varepsilon$



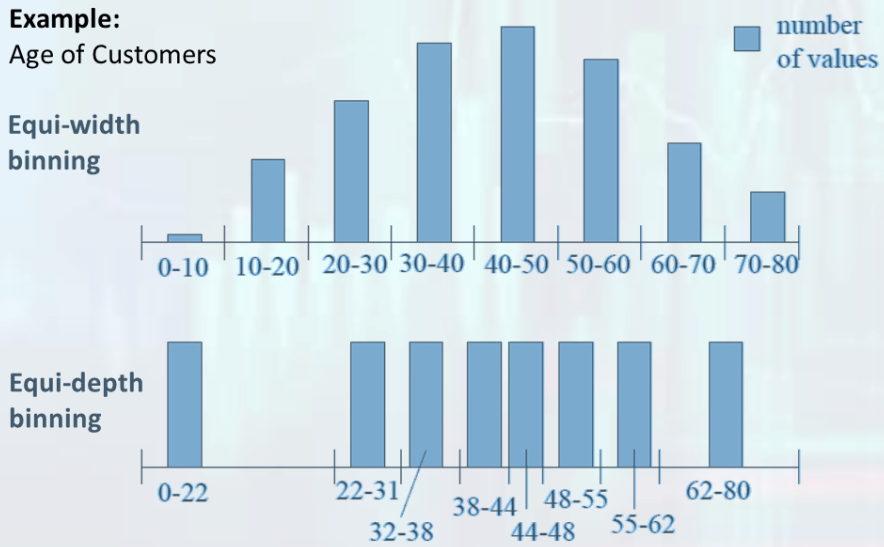




Numerosity Reduction: Non-Parametric Methods

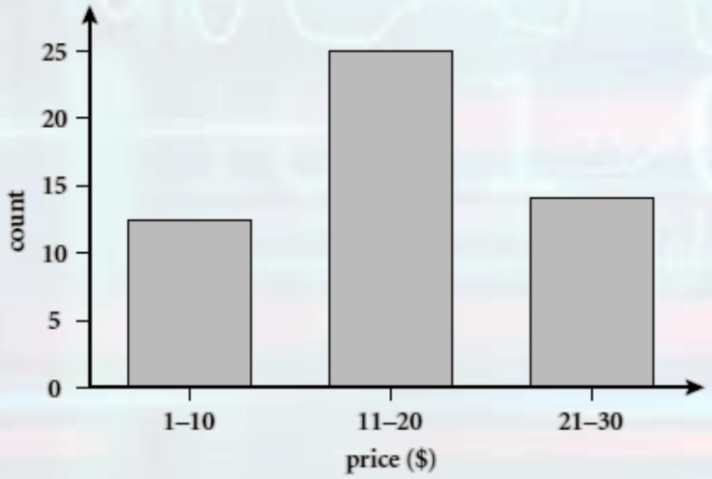
Binning

A top-down **unsupervised** splitting technique based on a specified number of bins.



Histogram

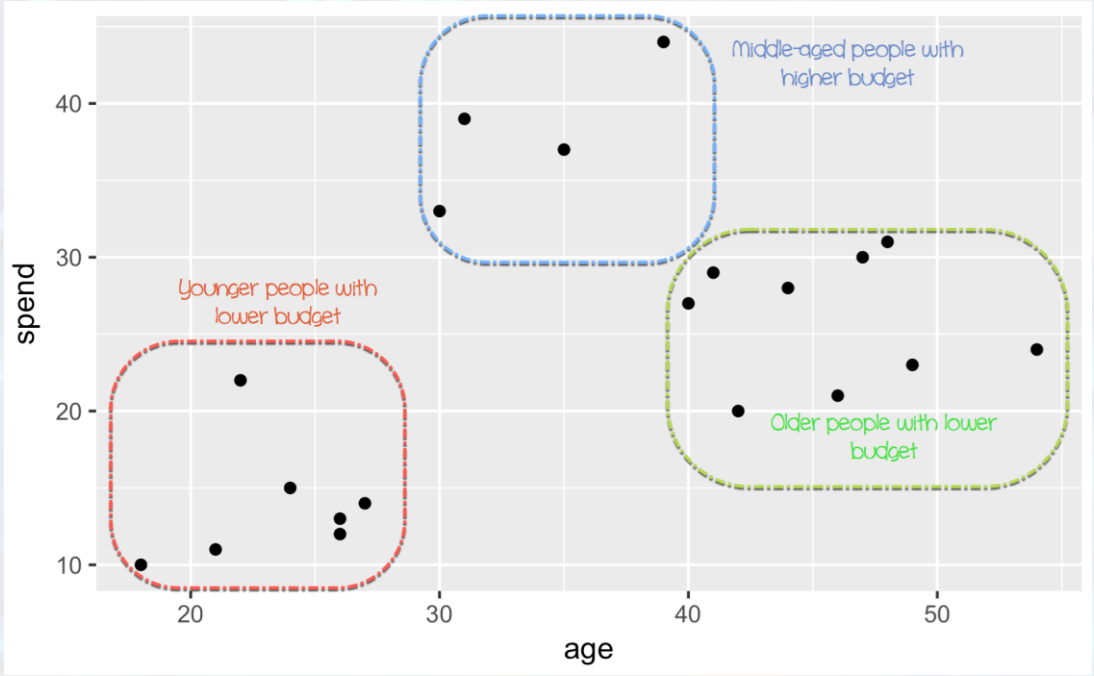
An **unsupervised** method to partition the values of an attribute into disjoint ranges called buckets or bins.



Numerosity Reduction: Non-Parametric Methods

Clustering

- A clustering algorithm can be applied to discretize a numerical attribute by partitioning the values of that attributes into clusters or groups.
  - **Unsupervised**, top-down split or bottom-up merge)
  - Partition dataset into clusters based on similarity
  - Effective if data is clustered but not if data is “smeared”
  - Cluster analysis using k-means (Lecture 6)

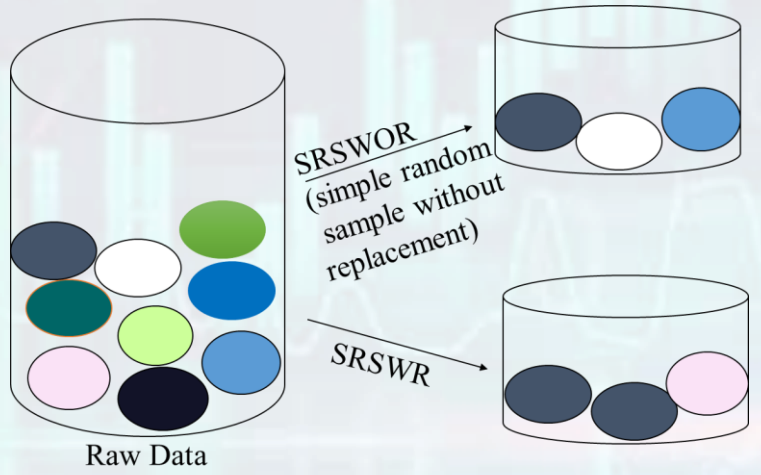


Numerosity Reduction: Non-Parametric Methods

Sampling

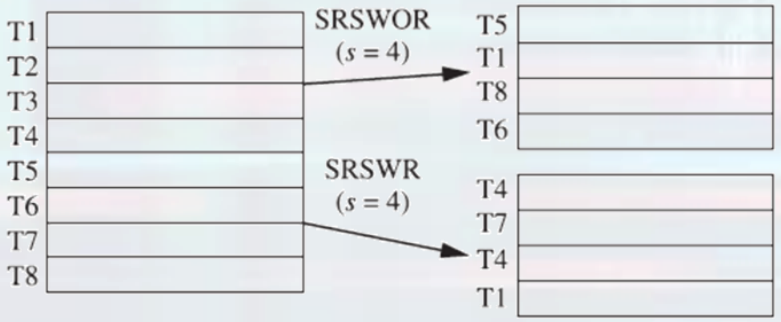
- Allows a large dataset to be represented by a much smaller random data sample (or subset).
- Sampling methods:
  1. **Sampling random sample without replacement (SRSWOR) of size s.**
  2. **Sampling random sample with replacement (SRSWR) of size s.**
  3. **Cluster sample.**
  4. **Stratified sample.**

Suppose that a large dataset, D, contain N tuples.



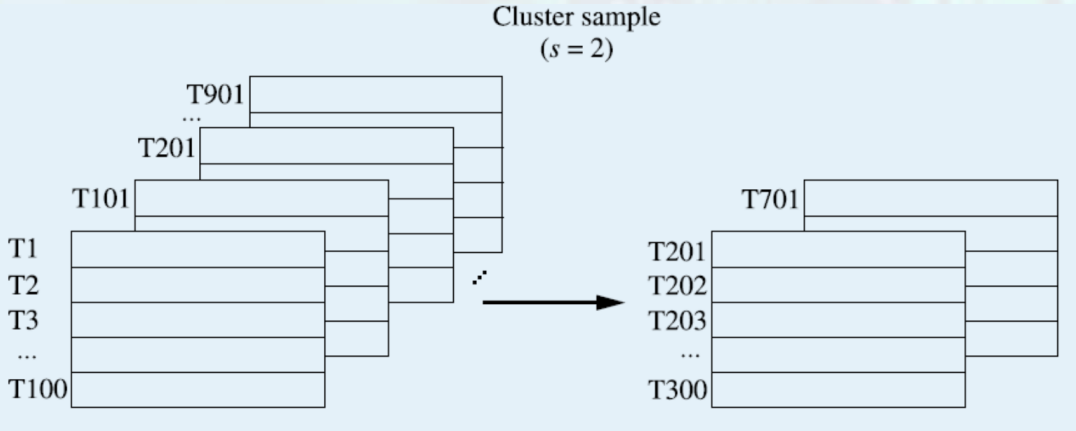
SRSWOR: All tuples are equally likely to be sampled.

SRSWR: After a tuple is drawn, it is placed back in D so that it may be drawn again.



Numerosity Reduction: Non-Parametric Method

Cluster sample: If the tuples in *D* are grouped into *M* mutually disjointed "clusters", then a simple random sample of *s* clusters can be obtained, where *s* < *M*.



Stratified sample: If ***D*** is divided into mutually disjoint parts called strata, a stratified sample of ***D*** is generated by obtaining an SRS at each stratum. This method is helpful when the data are skewed.

Stratified sample
(according to *age*)

T38	youth
T256	youth
T307	youth
T391	youth
T96	middle_aged
T117	middle_aged
T138	middle_aged
T263	middle_aged
T290	middle_aged
T308	middle_aged
T326	middle_aged
T387	middle_aged
T69	senior
T284	senior

T38	youth
T391	youth
T117	middle_aged
T138	middle_aged
T290	middle_aged
T326	middle_aged
T69	senior

Outline

- Overview
  - Data Quality
  - Major Tasks in Data Preprocessing
- Data Cleaning
- Data Integration
- Data Reduction
- **DATA TRANSFORMATION**

Data Transformation

- Data transformation strategies:
  1. **Smoothing**: Remove noise from data using techniques such as binning, regression and clustering.
  2. **Attribute/feature construction**: construct new attributes from the given set of attributes.
  3. **Aggregation**: Construct data cubes
  4. **Normalization**: Scale the attribute data to fall within a smaller, specified range such as -1.0 to 1.00, or 0.0 to 1.0.
  5. **Discretization**: Replace raw values of a numeric attribute (e.g. age) with interval label (e.g. 0-10 11-12) or conceptual labels (e.g. youth, adult, senior).
  6. **Concept hierarchy generation for nominal data**: Generalize attributes such as *street* to higher-level concepts such as *city* or *country*.

Normalisation

- Why normalisation?
  - Normalising the data attempts to put different attributes in the same scale.
  - Particularly useful for classification algorithms:
    - When using **neural network backpropagation algorithm** for classification mining, normalizing the input values for each attribute will speed up the learning phase.
    - When using **distance-based method** for clustering, normalization helps prevent attributes with initially large range (e.g. *income*) from outweighing attributes with initially smaller ranges (e.g. binary attributes).
      - Examples:
        - Income has range $3,000-$20,000
        - Age has range 10-80
        - Gender has domain Male/Female

Normalisation

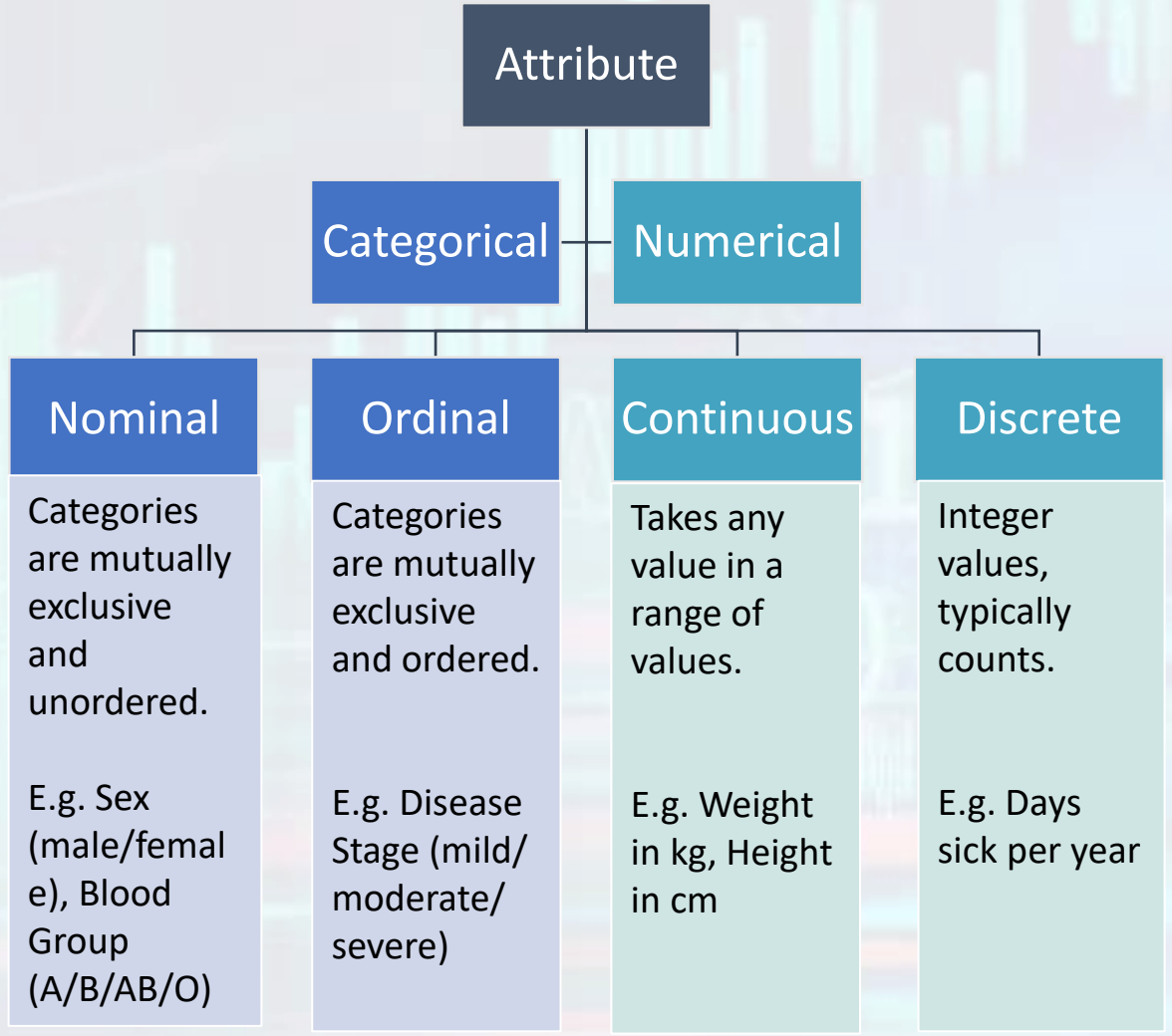
Min-Max	Z-score	Decimal scaling
Transforms the data into a desired range, usually [0, 1].	Useful when the actual min and max of attribute are unknown.	Transform data into a range between [-1, 1].
$v' = \frac{v - min_A}{max_A - min_A}(new\_max_A - new\_min_A) + new\_min_A$ Where, [min_A, max_A] is the initial range and [new_min_A, new_max_A] is the new range.	$v' = \frac{v - \mu_A}{\sigma_A}$ Where μ_A and σ_A are the mean and standard deviation of the initial data values.	$v' = \frac{v}{10^j}$ Where j is the smallest integer such that Max(|v'|) < 1.
Let income range \$12,000 to \$98,000 normalized to [0.0, 1.0]. Then \$73,000 is mapped to: $\frac{73{,}600 - 12{,}000}{98{,}000 - 12{,}000}(1.0 - 0) + 0 = 0.716$	Let μ = \$54,000, σ = \$16,000. Then \$73,600 is transformed to: $\frac{73{,}600 - 54{,}000}{16{,}000} = 1.225$	Suppose that the values of A range from -986 to 917. Divide each value by 1000 (i.e. j = 3): -986 normalizes to -0.986 and 917 normalizes to 0.917.

Discretisation

- Data discretisation transforms numeric data by mapping values to interval or concept label.

- Discretisation techniques:
  - Binning, Histogram analysis, Cluster analysis, Decision tree analysis, Correlation analysis

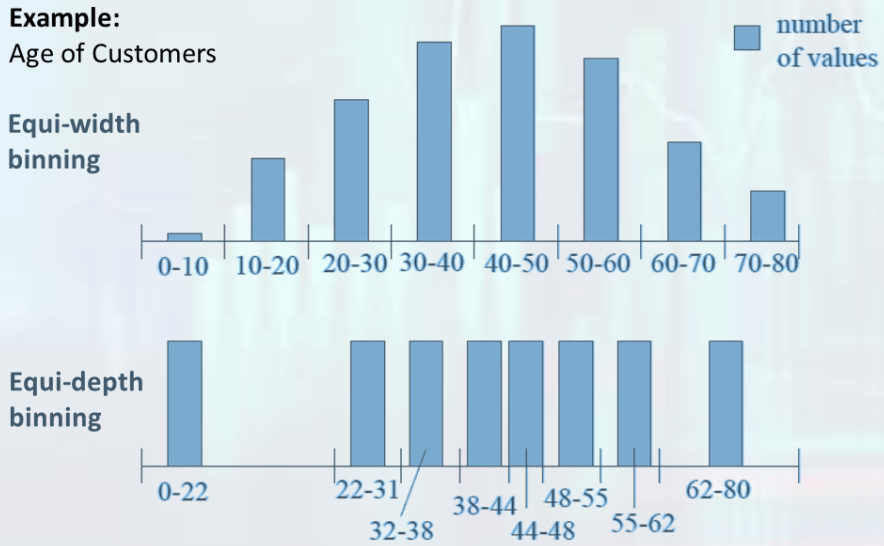
- For nominal data:
  - Concept hierarchy



Discretisation

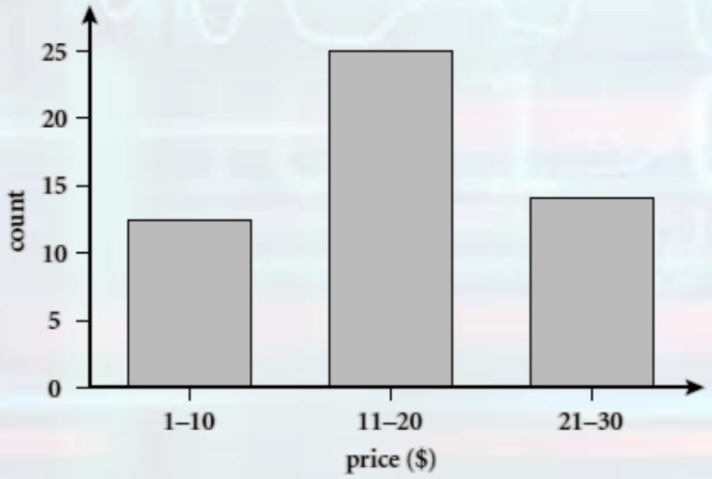
Binning

A top-down **unsupervised** splitting technique based on a specified number of bins.



Histogram

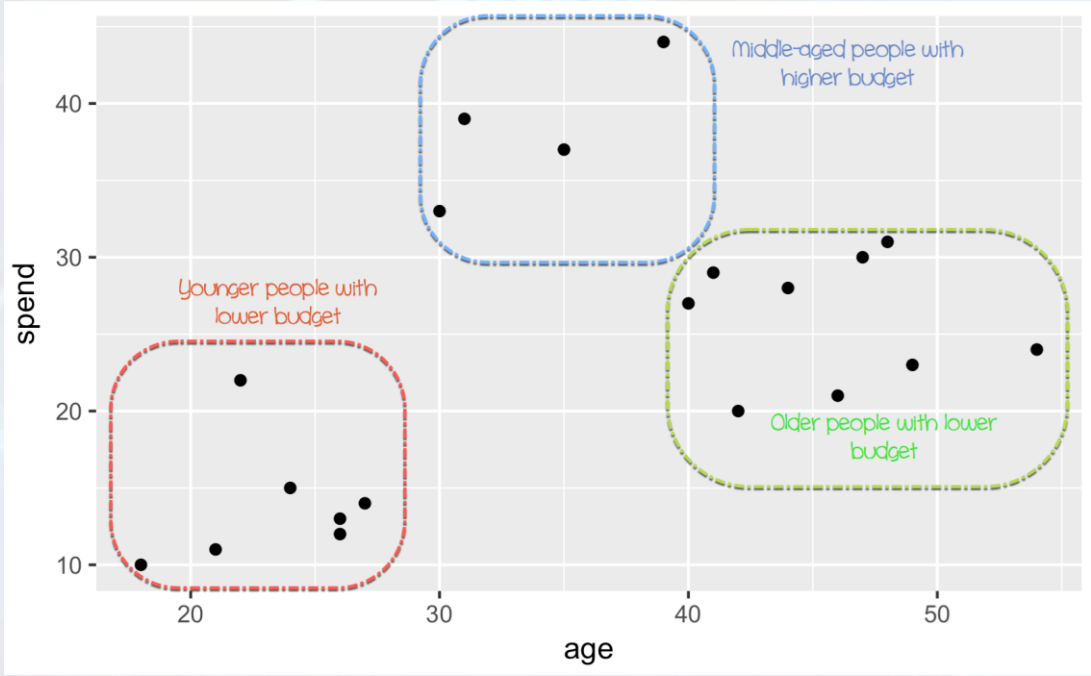
An **unsupervised** method to partition the values of an attribute into disjoint ranges called buckets or bins.



Discretisation

Cluster Analysis

- A clustering algorithm can be applied to discretize a numerical attribute by partitioning the values of that attributes into clusters or groups.
  - **Unsupervised**, top-down split or bottom-up merge)
  - Partition dataset into clusters based on similarity
  - Effective if data is clustered but not if data is “smeared”
  - Cluster analysis using k-means (Lecture 6)



Discretisation

- **Decision tree analysis**
  - Use a **top-down splitting** approach
  - **Supervised**: Make use of the class label (e.g., age < 40 and age >= 40)
  - Using ***entropy*** to determine split point (discretization point: the resulting partition contains as many tuples of the same class as possible)

$$Ent(U) = -\sum_{i=1}^{k} p_i \cdot \log_2 p_i$$

where:

- k – number of intervals
- p_i – the ratio of the value of the attribute in the i-th range to the number of all values of this attribute

Age	Buy
10	No
15	No
18	Yes
19	Yes
24	No
29	Yes
30	Yes
31	Yes
40	No
44	No
55	No
64	No

Split point = 35.5

Recursively find the best partition that minimizes entropy

	Yes	No
< 35.5	5	3
>= 35.5	0	4

$$E_1 = -\frac{5}{8}\log_2\frac{5}{8} - \frac{3}{8}\log_2\frac{3}{8} = 0.9544$$

$$E_2 = -\frac{0}{4}\log_2\frac{0}{4} - \frac{4}{4}\log_2\frac{4}{4} = 0$$

$$E_T = \frac{8}{12}E_1 + \frac{4}{12}E_2 = 0.6363$$

Discretisation

Correlation analysis

- Use a **bottom-up merge** approach
- **Supervised**: Make use of the class label (e.g. spam vs. genuine)
- ***ChiMerge*:** Find the best **neighboring intervals** (those having similar distributions of classes, i.e., low χ^2 values) to merge.

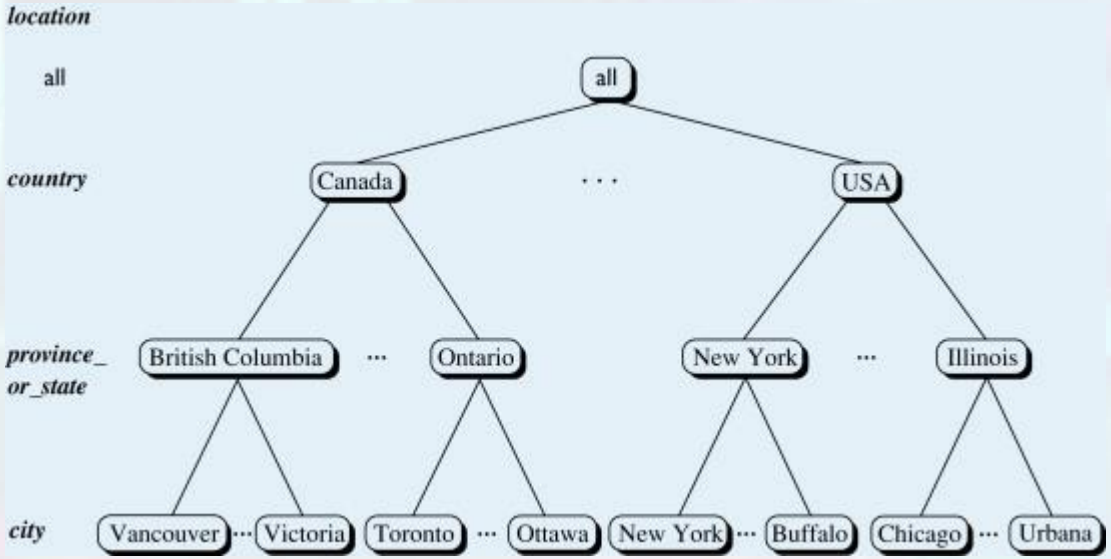
Sample	F	K	Intervals	Chi²
1	1	1	{0,2}	2
2	3	2	{2,5}	2
3	7	1	{5,7.5}	0
4	8	1	{7.5,8.5}	0
5	9	1	{8.5,10}	2
6	11	2	{10,17}	0
7	23	2	{17,30}	2
8	37	1	{30,38}	2
9	39	2	{38,42}	2
10	45	1	{42,45.5}	0
11	46	1	{45.5,52}	0
12	59	1	{52,60}	

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

Sample	F	K	Intervals	Chi²
1	1	1	{0,2}	2
2	3	2	{2,5}	4
3	7	1	{5,10}	5
4	8	1		
5	9	1		
6	11	2	{10,30}	3
7	23	2		
8	37	1	{30,38}	2
9	39	2	{38,42}	4
10	45	1	{42,60}	
11	46	1		
12	59	1		

Discretisation (Concept hierarchy generation for categorical data)

- **Nominal attributes** have a finite (but possibly large) number of distinct values, with no ordering among the values.
  - E.g. *geographic_location*, *job_category*, and *item_type*
- **Concept hierarchies** can be used to transform data into multiple levels of granularity.
- **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 *youth, adult*, or *senior*)



Discretisation (Concept hierarchy generation for categorical data)

Four methods for the generation of concept hierarchies:

1. **Specification of a partial ordering of attributes explicitly at the schema level by users or experts**
   - E.g. *street* < *city* < *state* < *country*

2. **Specification of a portion of a hierarchy by explicit data grouping**
   - E.g. {*Urbana, Champaign, Chicago*} < *Illinois*

3. **Specification of a set of attributes**
   - System automatically generates partial ordering by analysis of the number of distinct values
   - E.g. *street* < *town* < *country* < *country*

4. **Specification of only a partial set of attributes**
   - E.g. only *street* < *town*, not others

The attribute with the most distinct values is placed at the lowest level of the hierarchy

Summary

- **Data quality** is defined in terms of accuracy, completeness, consistency, timeliness, believability, and interpretability. These qualities are assessed based on the intended use of the data.
- **Data cleaning** routines attempt to fill in missing values, smooth out noise while identifying outliers, and correct inconsistencies in the data. Data cleaning is usually performed as an iterative two-step process consisting of discrepancy detection and data transformation.
- **Data integration** combines data from multiple sources to form a coherent data store. The resolution of semantic heterogeneity, metadata, correlation analysis, tuple duplication detection, and data conflict detection contribute to smooth data integration.

Summary

- **Data reduction** techniques obtain a reduced representation of the data while minimising the loss of information content. These include methods of dimensionality reduction, numerosity reduction, and data compression.
- **Data transformation** routines convert the data into appropriate forms for mining. For example, in normalization, attribute data are scaled so as to fall within a small range such as 0.0 to 1.0. Other examples are data discretization and concept hierarchy generation.
- **Data discretisation** transforms numeric data by mapping values to interval or concept labels. Such methods can be used to automatically generate concept hierarchies for the data, which allows for mining at multiple levels of granularity. Discretisation techniques include binning, histogram analysis, cluster analysis, decision tree analysis, and correlation analysis. For nominal data, concept hierarchies may be generated based on schema definitions as well as the number of distinct values per attribute.

References

- EMC Education Services. (2015). *Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data*. Wiley.

- Han, J., Pei, J., & Kamber, M. (2011). *Data mining: concepts and techniques*. Elsevier.