MTH2051

IT in Business

许炜源

21-2 Introduction

1. Basics of Information System

① Definition

a. 按成分定义

1) Information systems is a combination of

1. Hardware ← 计算机硬件
2. Software
3. Some **infrastructure** ← 除了计算机之外的硬件
4. and **trained people** who are organized to facilitate planning, control, coordination, and decision making in organizations

概念区分: Core components

Core components of an IS

Data Database Information Processor

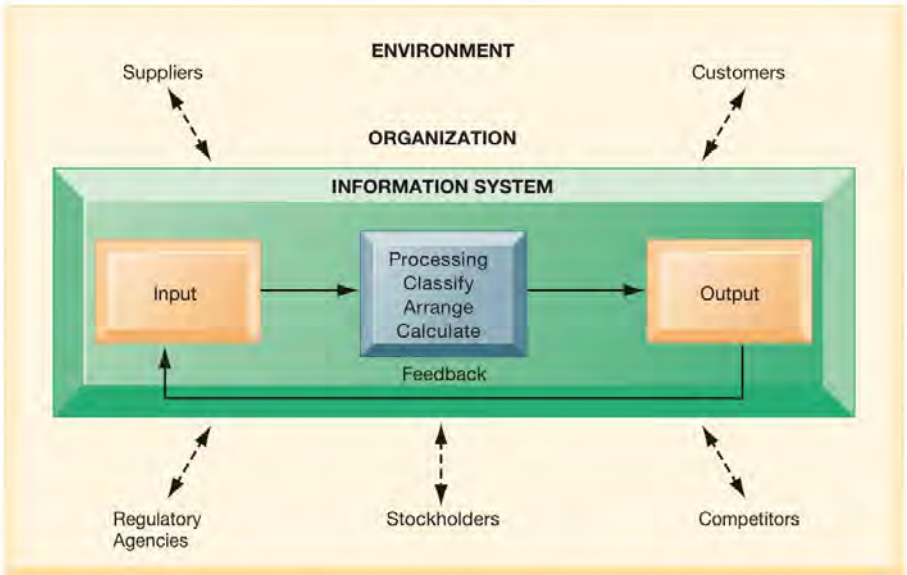
DIDIII

- The car is driving at a speed of 130km/h
- The car is speeding
- The company total sales during last month was 20,000 dollars
- The company total sales has increased by 10% in the last month
- The call-center receives 500 calls per day on average
- 50% of the calls that are coming into the call center are between 10 am and 1 pm.

b. 按用途定义

2) Information systems is the study of hardware and software that people and organizations use to collect, filter, process, create, and distribute data

3) An information system is an organized system for the collection, organization, storage and **communication of information.**



An information system contains information about an organization and its surrounding environment. Three basic activities—input, processing, and output—produce the information organizations need. Feedback is output returned to appropriate people or activities in the organization to evaluate and refine the input. Environmental actors, such as customers, suppliers, competitors, stockholders, and regulatory agencies, interact with the organization and its information systems.

② IS vs IT

IS: Hardware. Software. Infrastructure. People

IT: Hardware. Software

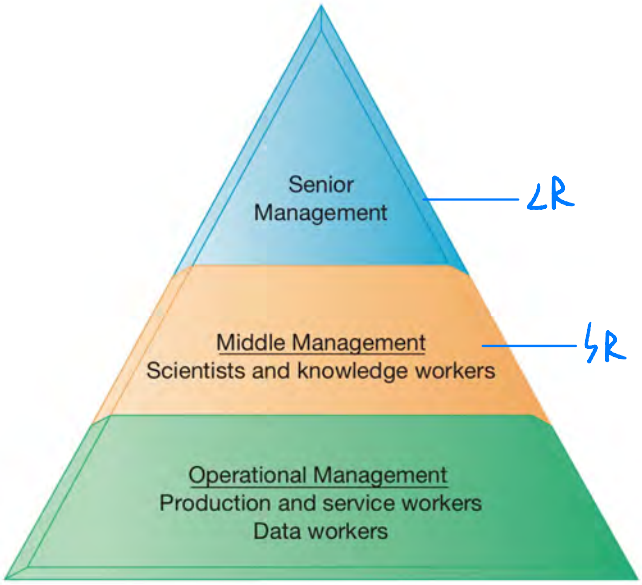
③ Four types

1. The executive support systems (ESS) which is found at the **strategic** level 行政管理
2. The management information systems (MIS) which is found at the **management** level
3. The decision-support systems (DSS) which is also found at the **management** level
4. Transaction processing systems (TPS) which is found at the **operational** level.

人

机器

事



LR

SR

利用数据的人

加工数据的机器

收集数据的事

④ Two topics

DataBase = Business Intelligence (Descriptive)

DataMining = Business Analytics (Predictive)

⑤ Extra knowledge

a. Analytics

When we have some data, we do (Descriptive analytics / Predictive analytics)

b. Statistics vs Machine Learning

Statistics : Inference / Descriptive analytics

ML : Prediction analytics

This distinction, however, have become a bit blurry as ML methods are used to improve statistical inference and statistics is used to improve predictions.

2. E-commerce

① E-business vs E-commerce

E-business

E-commerce : deal with buying and selling ← a component

② Types

The first one create value

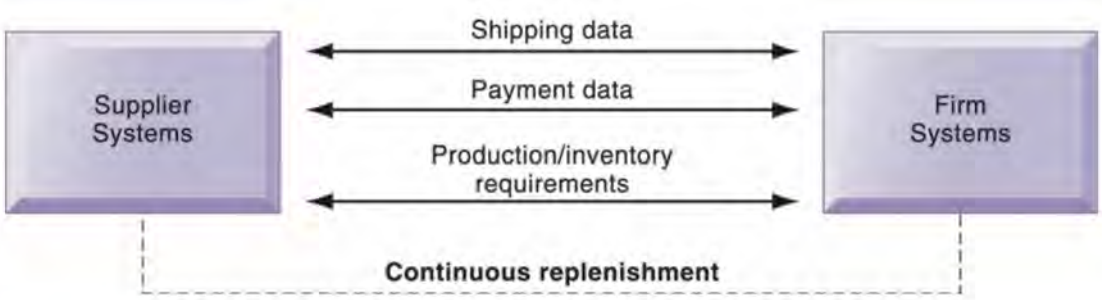
The second consumes such a value

	Consumer	Business	Government
Consumer	C2C	C2B	C2G
Business	B2C	B2B	B2G
Government	G2C	G2B	G2G

C2C: Taobao

C2B: feedback, sacrificing data

B2B is based on Electronic Data Interchange (EDI) :



Companies use EDI to automate transactions for B2B e-commerce and continuous inventory replenishment. Suppliers can automatically send data about shipments to purchasing firms. The purchasing firms can use EDI to provide production and inventory requirements and payment data to suppliers.

③Two models

a. Business models (按工作内容分类)

CATEGORY	DESCRIPTION	EXAMPLES
E-tailer 电商零售商	Sells physical products directly to consumers or to individual businesses.	Amazon Blue Nile
Transaction broker 交易经纪商	Saves users money and time by processing online sales transactions and generating a fee each time a transaction occurs.	ETrade.com Expedia
Market creator 市场创造者	Provides a digital environment where buyers and sellers can meet, search for products, display products, and establish prices for those products; can serve consumers or B2B e-commerce, generating revenue from transaction fees.	eBay Priceline.com Exostar Elemica
Content provider 内容提供商	Creates revenue by providing digital content, such as news, music, photos, or video, over the web. The customer may pay to access the content, or revenue may be generated by selling advertising space.	WSJ.com GettyImages.com iTunes.com Games.com
Community provider 社区提供商	Provides an online meeting place where people with similar interests can communicate and find useful information.	Facebook Google+ Twitter
Portal 门户网站	Provides initial point of entry to the web along with specialized content and other services.	Yahoo Bing Google
Service provider 服务提供商	Provides applications such as photo sharing, video sharing, and user-generated content as services; provides other services such as online data storage and backup.	Google Apps Photobucket.com Dropbox

现实世界有很多平台都是混合的!

b. Revenue models (按盈利方式分类)

1. **Advertising revenue model** (e.g., spots on YouTube videos, advertised results on the first Google search page)
2. **Sales revenue model** (Amazon, iTunes, online stores)
3. **Subscription revenue model** (Netflix, dating websites)
4. **Free/freemium revenue model**: here the basic service is free, but they charge a premium for advanced or special features (YouTube
5. **Transaction fee revenue model**: receiving a fee for enabling or executing a transaction (eBay) ← E-Tailer
6. **Affiliate revenue model**: Web sites (called "affiliate websites") send visitors to other websites in return for a referral-fee or a percentage of the revenue from any resulting sale (individual YouTubers, Podcasters, etc..)
7. And we also have blends of such models called **Mixed revenue model.**

④ Characteristics of consumers - SoLoMo

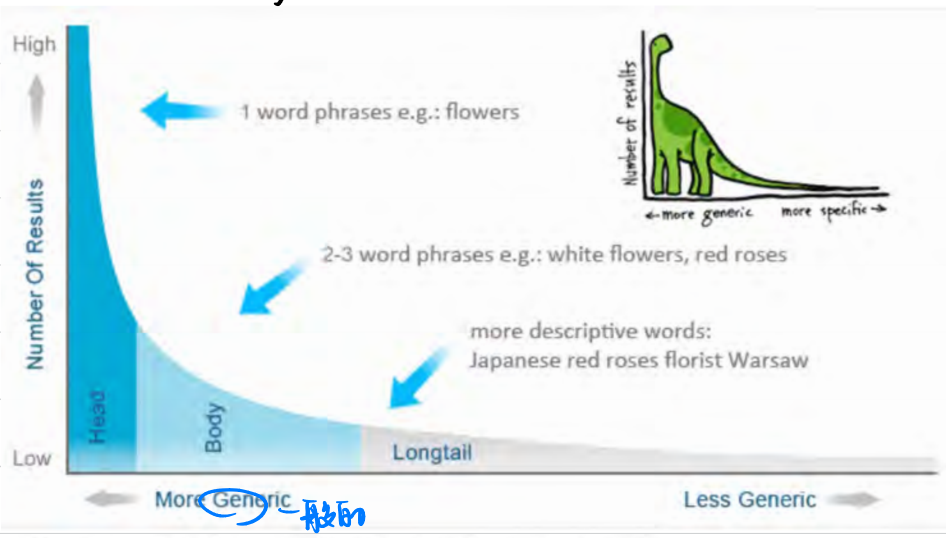
- SoLoMo stands for Social, Local and Mobile
- **Social:** Consumers are social, they no longer look at brands as the primary source of information
- they are more likely to consult third party resources and other members of their social networks (influencers, peers, compliance and conformity)
- Common social activities are: seeking advice, reading or writing reviews, and sharing experiences via a social network
- **Local:** Consumers get social where they are and where they live
- Smartphone owners use location-based services
- Platforms provide hyper-local offers by tracking where people live
- **Mobile:** Today's Consumers are largely defined by their mobility
- People buy goods and services while commuting to work, on the bus, train etc.
- Moreover, the market place is always available through mobile 24/7.

⑤ Success

Long-tail marketing + Behavioral targeting

a. Long-tail marketing

- adv: Less competitors
- disadv: Less consumers



Long-tail marketing in action

- The idea is to put an accent on less popular products, developing a business-model based upon products in the "long tail."
- Drawback: for such strategy there are supply-side considerations of managing a large inventory
- Thus it is suitable mainly for **pure** online businesses
- E.g., niche books can be found on Amazon, but cannot be found in a physical book-store (e.g., Waterstone)
- This is because it is not profitable for Waterstone to keep items that only the 0.0001 of the book buyers would buy (inventory costs)
- Amazon does not have such limitations because it is a virtual marketplace.

amazon
- niche product
- general product

general product

→ 享受的好处，不被地域影响

b. Behavioral / Psychological targeting

收集clickstream → 把它们投入更efficient的market (相当于个性化)

↑ Search engine optimization (SEO)

Increase the position of a search engine results pages (SERP)

e.g. Includes frequently searched keywords on your platform

3. AI for Business

Core techs:

- Massive amount of data : Data assessment, storage, management and cleaning
- Algorithms : e.g. ML algorithms
- Computing power : New processors, parallel computing and cloud computing

2.3 Introduction to Database

Database Structure

1. Database → 类比为 Computer

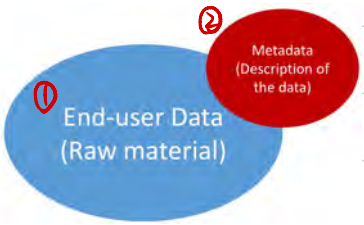
① Definition

A **shared, integrated** computer structure that stores 2 types of data:

1) **End-user data**: these are raw facts of interest to end users 终端数据

2) **Metadata**: this is data about data (i.e., a description of data characteristics and relationships). 元数据

↳ Metadata gives a map



Example: End-user data

Student Table

StudentNumber	StudentName	EmailAddress
10111	Ada Lam	ada@cuhk.edu.cn
10222	Bob Kent	bob@cuhk.edu.cn
10333	Charles Robertson	charles@cuhk.edu.cn
10444	Davis Chan	davis@cuhk.edu.cn

Class Table

ClassNumber	Name	Term	Section
MAT2090A	Real analysis	Spring	1
MAT2090B	Real analysis	Spring	2
DMS2080	Stochastic modeling	Fall	1

Grade Table

StudentNumber	ClassNumber	Grade
10111	MAT2090A	66.20
10111	DMS2080	81.35
10222	MAT2090B	98.20
10222	DMS2080	72.15
10333	MAT2090A	90.50
10333	DMS2080	92.65
10444	MAT2090A	58.00
10444	DMS2080	49.00

This is an example of End-user data, which is different from the Metadata.

Example: Metadata

描述表

USER_TABLES Table

TableName	NumberColumns	PrimaryKey
STUDENT	3	StudentNumber
CLASS	4	ClassNumber
GRADE	3	(StudentNumber, ClassNumber)

描述列

USER_COLUMNS Table

ColumnName	TableName	DataType	Length (bytes)
StudentNumber	STUDENT	Integer	4
StudentName	STUDENT	Text	50
EmailAddress	STUDENT	Text	50
ClassNumber	CLASS	Integer	4
Name	CLASS	Text	50
Term	CLASS	Text	5
Section	CLASS	SmallInteger	2
StudentNumber	GRADE	Integer	4
ClassNumber	GRADE	Integer	4
Grade	GRADE	Decimal	(3, 2)

maximum

In summary:

- **The End-user tables give information about he actual raw data**
- **The Metadata tables give information about the End-user tables.**

② Format

Often by tables (But not the unique way)

③ Relational database

In such database, a set of specific types of tables are related to each other through keys.

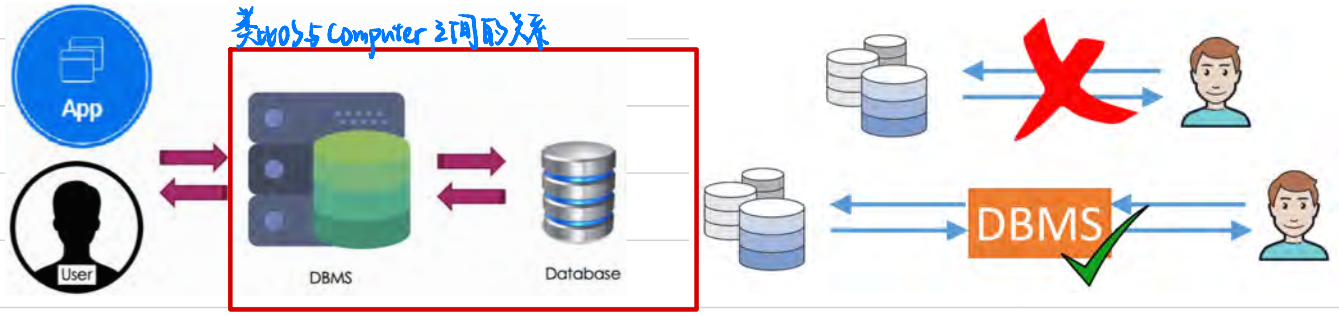
④ Types

- Number of users
  - Single-user database (Desktop/personal database)
  - Multiuser database (Work-group database or Enterprise database)
- Location (Physical location)
  - Centralized database
  - Distributed database
- Type of use
  - Operational database (e.g. transactional DB, production DB)
  - Data warehouse.

2. DBMS → 类似为 Operation System

① Definition

A collection of system, that generate, store and retrieve data in the database.



类似DBMS与Computer之间的关系

② Data management vs Data science

Data management

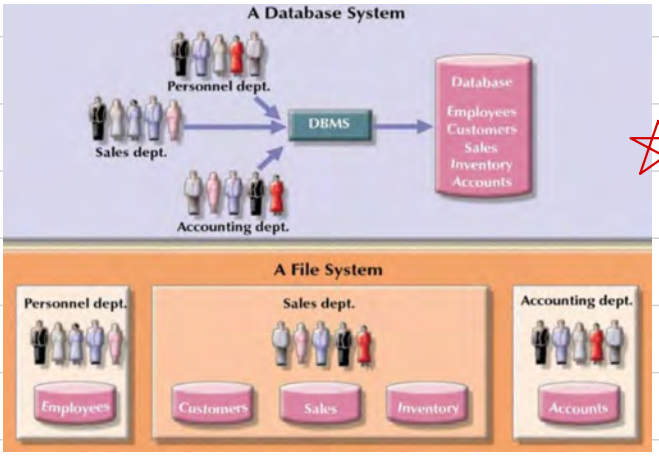
Data science : More concerned with data analysis.

③ Historical evolution

Flat-file system → Database system (Computer system)

(e.g. paper) (e.g. spreadsheet in Excel)

两者之间的一个大区别在于是否有Relationship



Disadvantages of Flat-file system

1) Data **redundancy**: i.e., different files contain same information (ID, name, address, etc...)

2) **Isolation** of data in **separate** systems: i.e., there is no data integration across systems

3) Data **inconsistency**: e.g., we may find different address values for the same person in different data files

4) Lack of data **integrity**: i.e., data may not be accurate, it may be missing or it may not be verifiable

5) Data **anomalies**: i.e., all changes may not be made successfully in each department

e.g., **Update** / **Insertion** / **Deletion** anomalies

6) Structural and data **dependency**: i.e., changing the file structure or data characteristics (e.g., data type) in the table will affect the application's ability to access data

7) It is difficulty to program **security** features and get information to end-users in a **timely manner**.

比如改变excel中一列的type，则可能会丢失一些数据

④ Advantages & Disadvantages

a. Advantages

- Minimal data **redundancy**
- Data **consistency**
- **Integration** of data
- Improved data **sharing**
- Enforcement of **standards**
- Ease of **application development**
- Uniform **security**, **privacy**, and **integrity**
- Data **independence** from applications
  - The "self-describing" data are stored in a data dictionary (i.e., the metadata).

Flat-file system 一般没有Metadata

b. Disadvantages

- Increased **costs**
- Management **complexity** (因为多了很多Relationship)
- Maintaining **currency** (i.e., performing frequent updates)

Cost

- It's vendor **dependence** (e.g., Oracle, Microsoft, MySQL)
- You need frequent **upgrade** and **replacements**.

⑤ Function

1. Data **dictionary** management: DBMS **stores definitions** of the data **elements** and their **relationships** (metadata) in a data dictionary
2. Data **storage** management: DBMS creates **complex structures** required for data storage, relieving the user from the difficult task of **defining** and **programming** the **physical** data characteristics
3. Data **transformation and presentation**: DBMS transforms entered data to **conform** to required data structures. e.g. date and time formats
4. **Security** management: DBMS creates a security system that **enforces** user security and data privacy
   - E.g., Authentication; security rules
5. **Multiuser access** control: DBMS uses sophisticated algorithms to ensure that multiple users can access database **concurrently** without compromising the integrity of the database.
6. **Backup and recovery** management: DBMS provides backup and data recovery to ensure data safety and integrity
7. Data **integrity** management: DBMS promotes and enforces integrity rules, minimizing data **redundancy** and maximizing data **consistency**
8. Database **access languages** and application programing **interfaces**: DBMS provides data access through a query language
   - Structural Query Language (SQL) is the query language and data access standard supported by the majority of DBMS vendors
9. Database **communication** interfaces: current DBMSs accept end-user requests via multiple, different network environments.

A map

3. Data model ≜ Database model (How to design a database)

Style

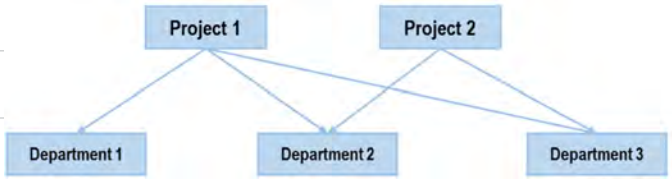
1. Hierarchical database model
2. Network database model
3. Relational database model.

↓ upgrade

a. Hierarchical

一对多

b. Network



多对多

c. Relational

- It was initially described by the English computer scientist Edgar F. Codd (Turing Award Winner) in his's landmark paper – "A Relational Model of Data for Large Shared Data Banks" (1970)

1. Project Table: (Entity; Attribute)

ProjectID	ProjectName	DptID	LeaderID
023	Payroll	11	1553
196	Alb[illegible] release	15	8817

2. Department Table: (Relationship)

DptID	DptName	DptManagerID
11	Human Resource	1553
14	Production	2566
15	Marketing	9022

- This model represent data using **multiple tables**
- These tables are **related** to each other by certain keys
- E.g., the Department Table is linked to the Project Table via the Department ID.

FK 对应 PK (referenced)

3. Personnel Table:

EmpID	Name	DptID
1553	Ian Lee	11
2566	Mack Hollins	14
8817	Amanda Nash	15
9022	Philip Tan	15

Instance

The 4 main components:

- Entity : An object / concept about which data is collected and stored
- Instance (Entity instance) : Row
- Attribute : Columns (必须同时Format/Data type Numeric, Character, Date, Logical; 各有Attribute domain) — True or False
- Relationship : Connections among data (in a DB, keys are used to connect tables)

The order of rows and cols is immaterial to the DBMS.

Attributes的命名方式: dpt_name / DptName

L4 Relational Data Model

1. Keys

① Definition

Keys are attributes or combination of attributes used to:

对于单个表: Ensure that each row in a table is uniquely identifiable (Identify each instance)

对于多个表: Establish relationships between tables

i.e. A key consists of one or more attributes that determine other attributes.

StudentTable

Here StudentNumber determines the other 2 attributes (StudentName and EmailAddress

StudentNumber	StudentName	EmailAddress
10111	Ada Lam	ada@cuhk.edu.cn
10222	Bob Kent	bob@cuhk.edu.cn
10333	Charles Robertson	charles@cuhk.edu.cn
10444	Davis Chan	davis@cuhk.edu.cn

Examples 找到有决定性作用的最小属性组

BookTable

BookID	BookTitle	Publisher	Location
QC.453.1	Introduction to Art	DK	4/F
AD.02.086	Gone with the wind	Macmillan	2/F
P.291.4	Amusing ourselves to death	Viking Press	3/F

GradeTable

StudentNumber	ClassNumber	Grade
10111	MAT2090A	66.20
10111	DMS2080	81.35
10222	MAT2090B	98.20
10222	DMS2080	72.15
10333	MAT2090A	90.50
10333	DMS2080	92.65
10444	MAT2090A	58.00
10444	DMS2080	49.00

StudentTable

StudentID	StudentName	ClassName	Credit	Grade
10111	Ada Lam	Real analysis	2	66.20
10111	Ada Lam	Stochastic modeling	3	81.35
10222	Bob Kent	Real analysis	2	98.20
10222	Bob Kent	Stochastic modeling	3	72.15
10333	Charles Robertson	Real analysis	2	90.50
10333	Charles Robertson	Stochastic modeling	3	92.65
10444	Davis Chan	Real analysis	2	58.00

② Functional dependency

a. A Determines B: A→B 知道A属性→我们也知道B属性

b. B is Functionally dependent on A: A→B (one and only one value in B)

c. B is Fully Functionally dependent on A: A→B (A决定B，且在A的subset里都不能决定B) ↑这时A是最小决定属性集

d. Null value: No data entry, no value (0和空格也不行) (PK不允许NV存在)
尽可能后 通常可以帮我们筛选出PK

③ Types

a. Composite key: A key with more than one attribute ← 这时叫"key attribute"

b. Superkey: A key that uniquely identifies each row

Examples:

- BookID (each value in this column is unique)
- BookID, BookTitle (each combo of values in these columns is unique)
- BookID, BookTitle, Publisher (same as above).

Book Table:

BookID	BookTitle	Publisher	Location
QC.453.1	Introduction to Art	DK	4/F
AD.02.086	Gone with the wind	Macmillan	2/F
P.291.4	Amusing ourselves to death	Viking Press	3/F

c. Candidate key: Minimal superkey

Examples: Which of the following is the candidate key?

- BookID ✔
- BookID, BookTitle
- BookID, BookTitle, Publisher.

The candidate key is BookID. Why?
Because it is the minimal superkey
i.e., BookID alone can determine all the other row values in the table

Book Table:

BookID	BookTitle	Publisher	Location
QC.453.1	Introduction to Art	DK	4/F
AD.02.086	Gone with the wind	Macmillan	2/F
P.291.4	Amusing ourselves to death	Viking Press	3/F

= Identifier

d. Primary key (PK): Optimal candidate key ← Determined by designer's choice or user's requirement

Unique
No null values
⇐ Entity integrity (单)

e. Foreign key: 在parent table中是PK，在child table中是FK

Table Name: Student
Primary Key: StudentNumber
Foreign key: none

parent

StudentNumber	StudentName	EmailAddress
10111	Ada Lam	ada@cuhk.edu.cn
10222	Bob Kent	bob@cuhk.edu.cn
10333	Charles Robertson	charles@cuhk.edu.cn
10444	Davis Chan	davis@cuhk.edu.cn

Link

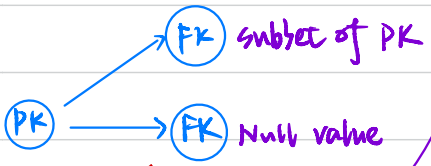
child

Table Name: BorrowRecord
Primary Key: BookID, StudentNu.
Foreign key: StudentNumber
一个表里可以有多个FK

BookID	StudentNumber	BorrowedDate	DueDate
QC.453.1	10222	Dec 19, 2018	Jan 1, 2019
AD.02.086	10111	Jan 3, 2019	Jan 17, 2019
P.291.4	10444	Jan 5,2019	Jan 19, 2019

如果FK中出现PK中没有的instance，则此时不满足Referential integrity

只有两种情况：



Referential integrity (多)
子的attribute的某一项在父的同一attribute中能否找到
(不能直接删掉)

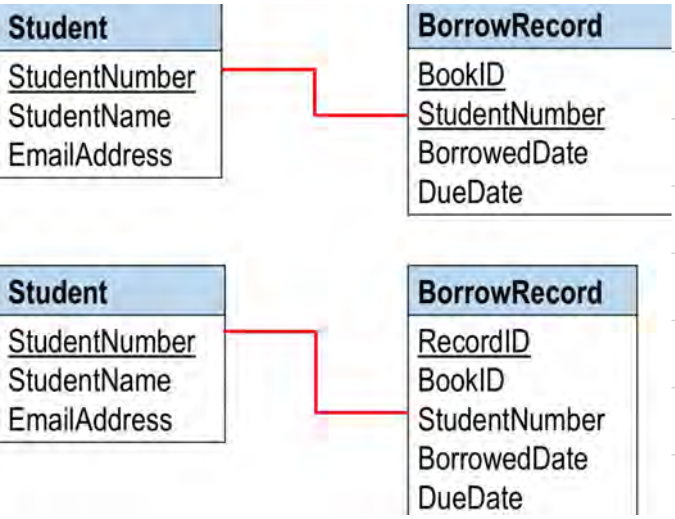
当忘记PK时

J. Secondary key: An alternative key that can be used to retrieve the information 要除去PK

(虽然不一定是unique → Secondary key's effectiveness (how restrictive it is)

越高越好

④ Rational Diagram (RD)



对underscore的attributes是PK

Link连接PK与FK

L5 Conceptual Model : Entity Relationship Model (ER Model)

1. Database development process

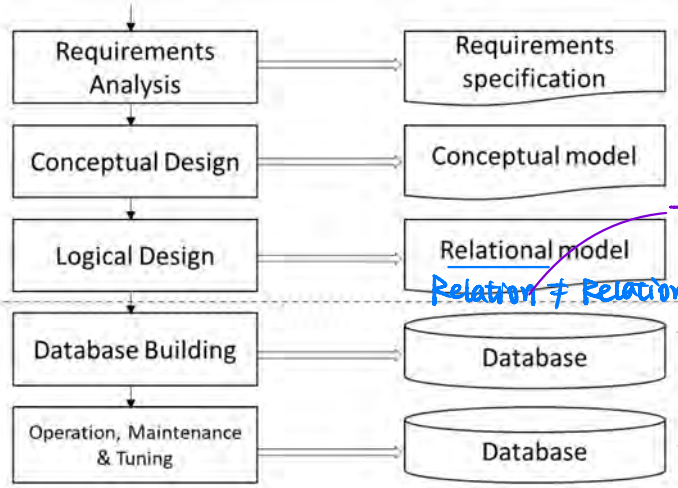


Diagram is a component of Model

Model = Description

Diagram: Graphic presentation

=> 外模式 = Diagram

=> 外模式: Schema & Diagram

ER Model : Collection of entities and relations between those entities — Entity-specific

Relational Model : Collection of tables and relations between those tables — Table-specific

2. ER Model

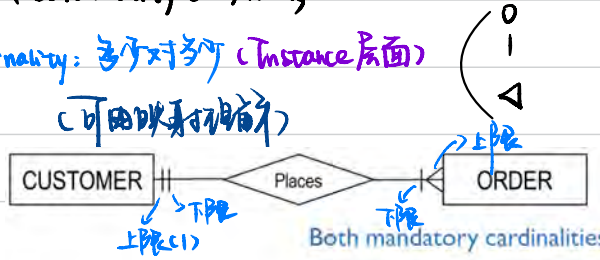
① Process to build up an ER Model

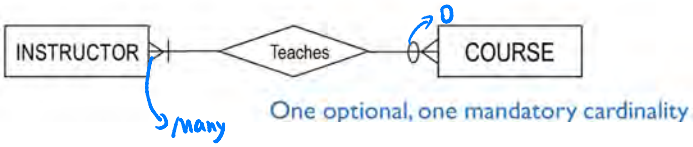
a. Find entities

b. Find the relationship between entities

c. Find cardinality constraints

Cardinality: 多少对多少 (Instance层面)





这些symbols表示Cardinality

- We have some Entities (with some attributes):

Entity	Attributes
Customer	CustomerID, Name, Address, Phone, Balance
Product	ProductID, Description, Price
Order	OrderID, Date, Status

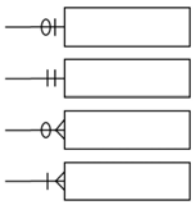
- We also have some relationships:
  - A Customer places zero or many Orders
  - An Order is placed by one and only one Customer
  - An Order has one or many Products
  - A Product can be on none or many Orders.

这叫做 Type of relationship

如何判断?: 看两个方向!

Optional & Mandatory

- Optional one (0 or 1)
- Mandatory one (exactly 1)
- Optional many (>=0)
- Mandatory many (>0).

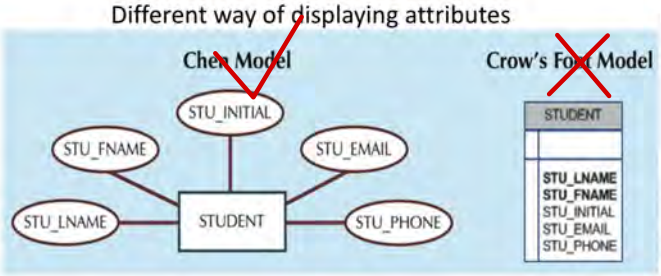


d. Find attributes of entities

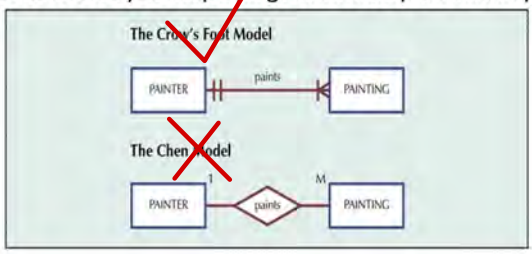
e. Find candidate and primary key

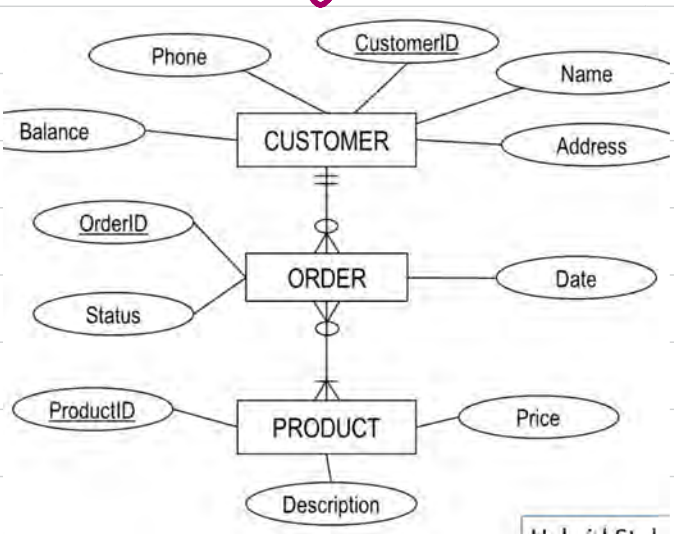
3. ER Diagram

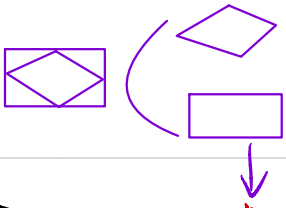
Different way of displaying attributes



Different way of displaying relationship cardinality



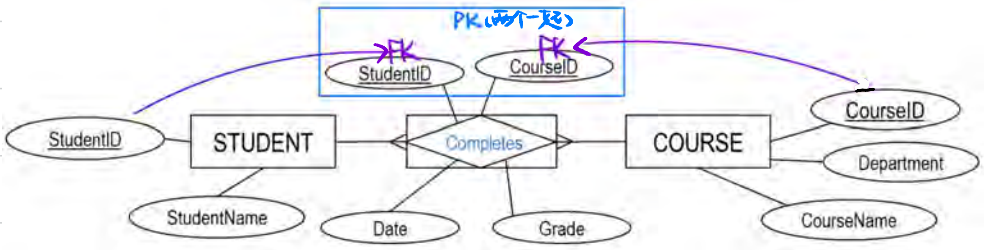




① Components

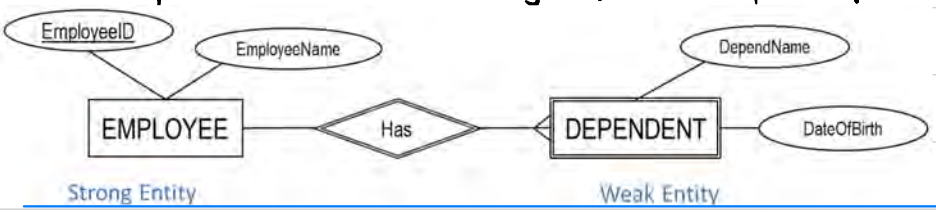
Entity

Associative entity：一些entities的属性attributes组成一个新的entity（实际上不是Entity）



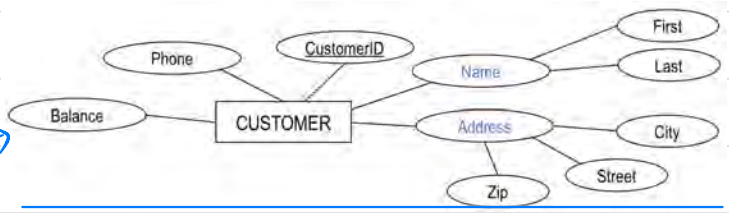
Strong：Independent：可单PK为identify ⇒ Parent

Weak：Dependent：没有PK，通过与Strong entity的Relationship为identify（e.g. Classroom）⇒ Child

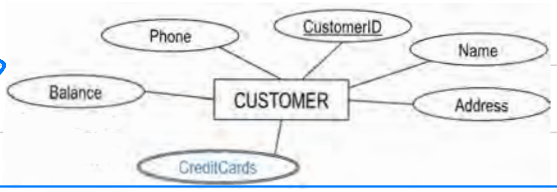


Attribute

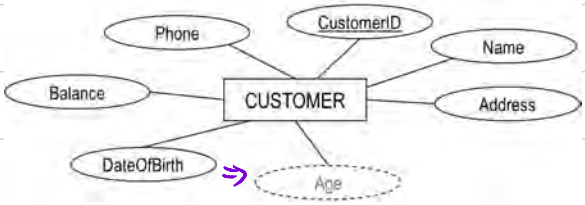
- Simple
- Composite
- Single-valued
- Multiple-valued
- Derived



多个oval



双圆

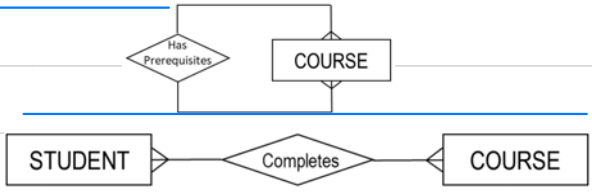


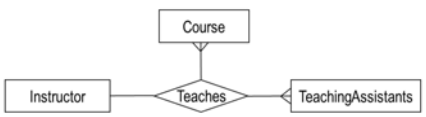
虚心的

Relationship　A verb

Degree of Relationship

- Unary　Degree 1　("Recursive")
- Binary　Degree 2　(most common)
- Ternary　Degree 3　(Confusing)





先与到一个Relationship的Entity的数目

(续上)

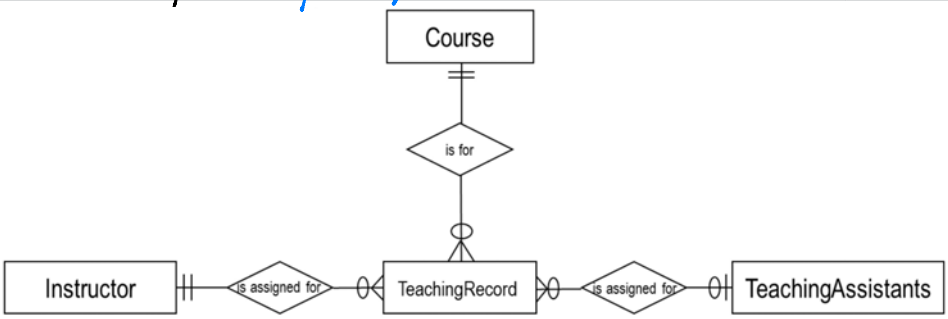
N-ary relationship

When it comes to N-ary relationship, we get confused with the direction and cardinality

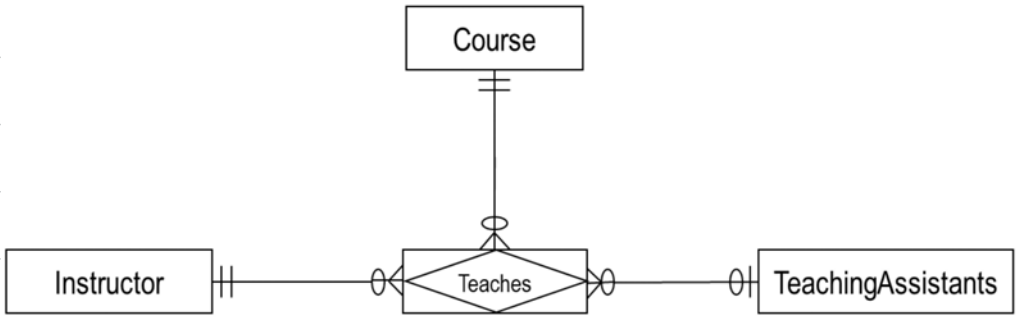
(Ternary is one kind of N-ary relationship)

Then we come up with some options to deal with this case.

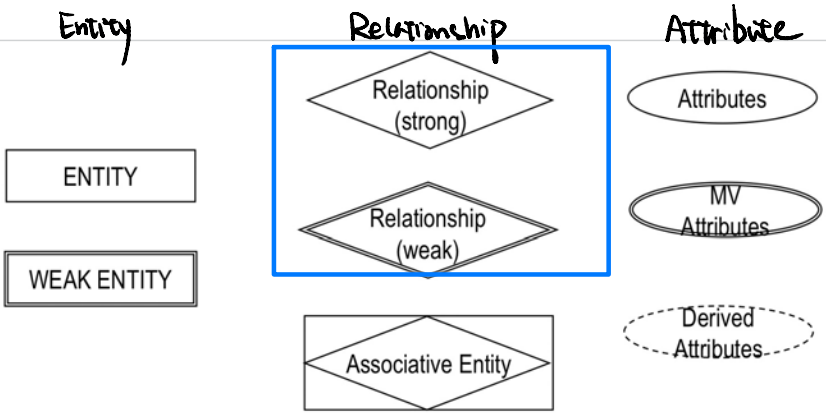
Option 1: Add an entity Ternary → Binary



Option 2: Add an associate entity



② Summary of symbols



Strong relationship Strong Entity & Strong Entity

Weak relationship Strong Entity & Weak Entity

4. Draw ER Diagram

① Entity

② Relationship & Cardinality

③ Attribute

④ 备注：

- Entity
  - Strong / Weak — 1
- Relationship
  - Degree
  - Strong / Weak — 3
  - Associative entity
- Attribute
  - Primary key
  - Simple / Composite — 4
  - Single-valued / Multi-valued
  - Derived

我们使用EER来处理更复杂的数据，这个意味着EER比ER好。

L6 Conceptual Model: Enhanced Entity Relationship Model (EER Model)

1. EER Model

① Definition

i.e. 我们进一步地 expand Entity

EER Model = ER Model + Additional semantic concepts
- Specialization / Generalization (这门课教这个)
- Aggregation
- Composition

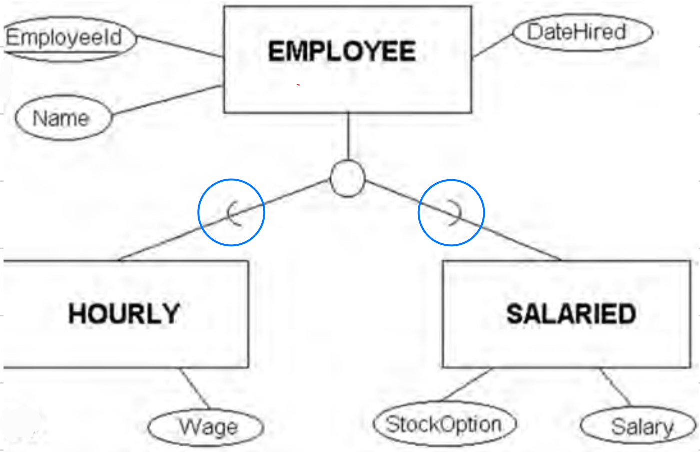
+ 继承关系

→ 通常用于有相同attribute又有不同attribute的情况

② Specialization / Generalization

a. Concepts

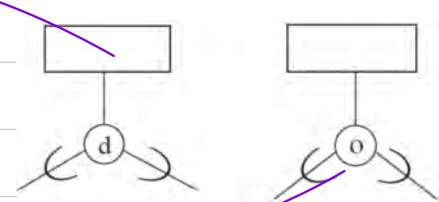
- Supertype 父类
- Subtype 子类 继承了父类的attribute & relationship，但也有自己的新attribute & relationship



b. Inheritance details

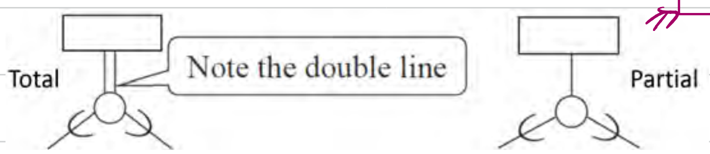
↑ ① 一个subtype 可有多个supertypes: Multiple inheritance (继承多个父类)

↓ ② 一个supertype 的多个subtypes 之间可能有Overlap
- Disjoint
- Overlap (Non-disjoint)



Summary ⇒
- Disjoint Total
- Disjoint Partial
- Overlap Total
- Overlap Partial

↓ ③ 一个supertype 的instance 可能不属于任何subtypes
- Total specialization (全部有个性)
- Partial specialization (部分有个性)



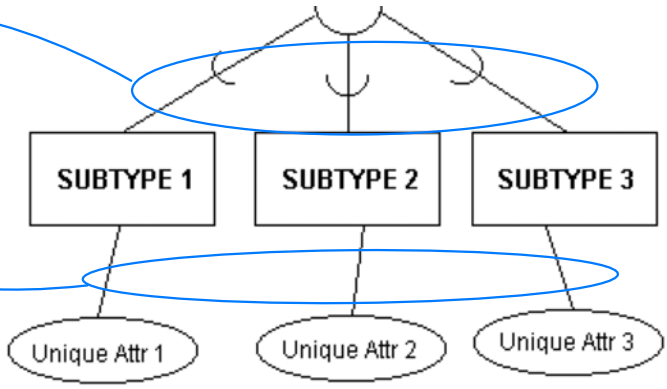
☆ 注意：当一个entity 只有共同点，没有不同点时，不需创建其为subtype

Total / Partial 由具体情况讨论

c. Way of identifying Heritance

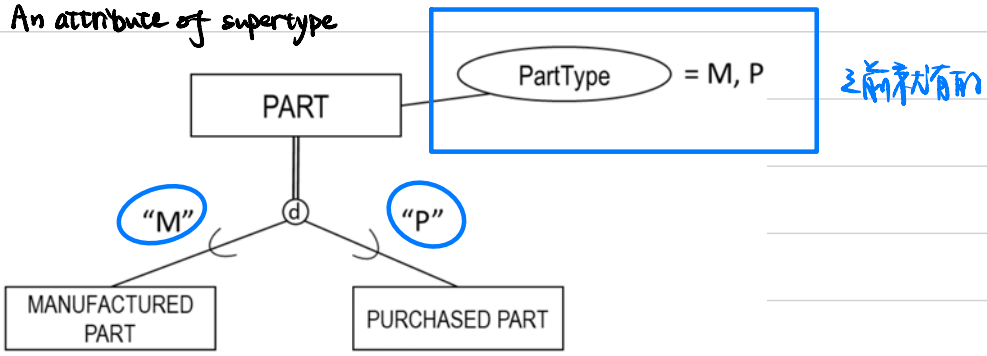
Generalization 找共同

Specialization 找不同



d. Subtype discriminator

An attribute of supertype



之前有讲到

③ Function

a. Avoid unnecessary null values

b. Enable some particular type to participate in some unique relationships.

2. Draw EER Diagram (具体例子为W3-2作业)

① Draw ER Diagram

② Heredity
- Subtype
- Hereditary method
  - Total / Partial
  - Disjoint / Overlap
- Subtype's attribute

L7-8 Logical Design

1. Logical Design

Conceptual Model → Relational Model (Entity = Relation)

能转换更多信息

与别人沟通：Conceptual Model 更好

建DB：Relational Model 更好

2. Relational Schema & Relational Diagram

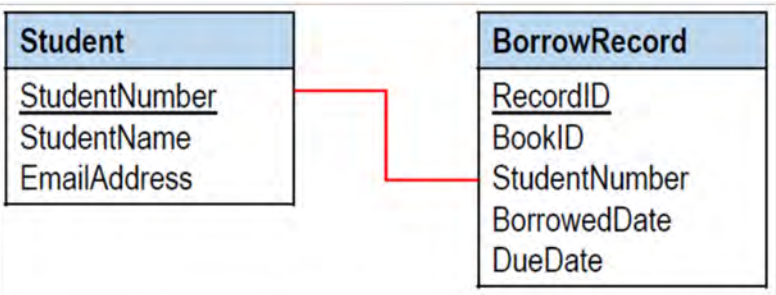
① Definition

Schema = Textual representation

Diagram = Graphic representation

Entity Attribute

Student (StudentNumber, StudentName, EmailAddress).



② Transformation: ERM → RM (具体到必为M-1上课例子)

a. Construct an EER Diagram

Step 0: Identify entities, relationships and attributes

b. Entity

- Step 1: Strong entity types
- Step 2: Weak entity types

⚠ 只有 Supertype 才分 Strong & Weak

⚠ 在这里先忽略 Multi-valued attribute 和 Derived attribute

c. Relationship

- Step 3: Many-to-many binary relationships
- Step 4: One-to-many binary relationships
- Step 5: One-to-one binary relationships
- Step 6: Recursive (Unary) relationships
- Step 7: n-ary relationships (n ≥ 3)
- Step 8: Supertype/subtype relationships
- Step 9: Weak entity types (revisit)

Type

Degree

Other

d. Attribute

- Step 10: Multi-valued attributes
- Step 11: Derived attributes
- Step 12: Additional constraints

③Normalization

a. Function

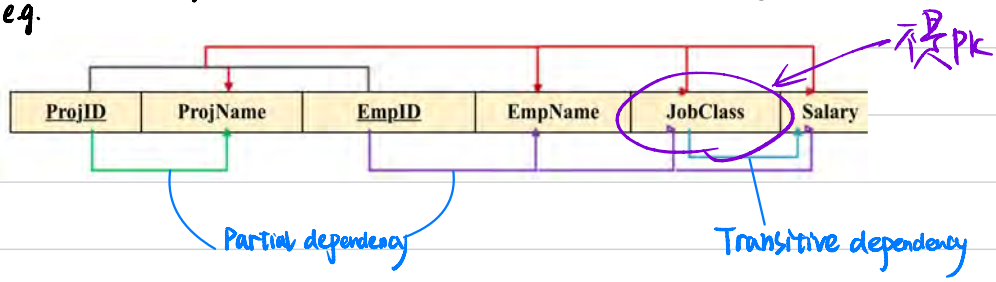
Solving
- Data redundancy
- Data anomalies (inconsistency)
  - Insertion
  - Deletion
  - Modification

  in row

b. Dependency (可以被函数决定)

- Partial dependency : 部分函数决定关系 ⟶ Redundancy
- Transitive dependency : $X \to Y, Y \to Z \Rightarrow X \to Z$ ⟶ Anomaly

e.g.



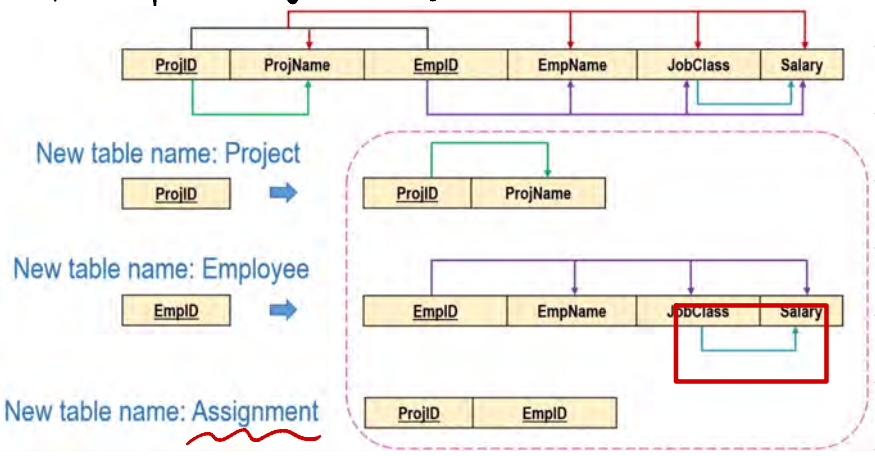
c. Normal forms

① First normal form (1NF)

For any relation, attributes are all single-valued (atomic)

② Second normal form (2NF) (Fully functionally dependent)

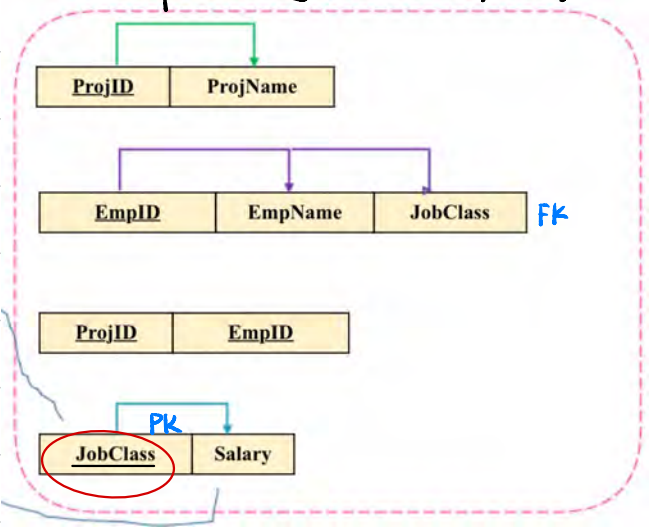
2NF = 1NF + Solving partial dependency



② Third normal form

3NF = 1NF + Solving partial dependency + Solving transitive dependency

= 2NF + Solving transitive dependency



其JobClass也設為Primary key

L9-10 SQL Basics

1. Data type

- INTEGER
- DECIMAL (precision, scale) Precision 是 # of digits, Scale 是 # of digits after the decimal point
- CHAR(n) 那一列一定为 n 个字符
- VARCHAR(n) 那一列最多为 n 个字符
- DATE 'YYYY-MM-DD'
- DATETIME 'YYYY-MM-DD HH:MM:SS'

Invalid dates & times 将会变成 '0000-00-00'

2. Data definition language (DDL)

① Purpose

Create a database
- Relation
- Relationship

② Commands

a. CREATE TABLE

```
CREATE TABLE tablename (
        column1         data type       [constraint],
        column2         data type       [constraint],
        PRIMARY KEY  (column),
        FOREIGN KEY  (column) REFERENCES tablename
        );
```

Optional

```
CREATE TABLE product (
        product_id          VARCHAR(10)  NOT NULL  UNIQUE,
        product_name        VARCHAR(20)  NOT NULL,
        product_price       DECIMAL(8,2) NOT NULL,
        vendor_id           INTEGER,
        PRIMARY KEY  (product_id),
        FOREIGN KEY  (vendor_id) REFERENCES vendor ON UPDATE CASCADE
        );
```

可能有多个

(可空的PK)

PK变 → FK变 (自动的)

指向

不同表中可以有同名的列

b. ALTER TABLE

c. DROP TABLE

3. Data manipulation language (DML)

① Purpose

Manipulate data in the database

② Commands

a. Standard query

△ SELECT 可以选多列，用","隔开

```
SELECT [DISTINCT | ALL] {column(s) | * [AS new_name]}
    FROM table_name [ AS alias]
    [WHERE condition(s)]
    [GROUP BY column_list] [HAVING condition]
    [ORDER BY column_list]
```

or　all

!! Required

[] Optional

△ WHERE

流通不是"=="

=, <, >, <=, >=

AND, OR, NOT

BETWEEN, IN, LIKE

条件可以用()圈起来

- Between　e.g. WHERE x Between y And z
- IN　e.g. WHERE x IN ('Lee', 'Lea') list
- Like　e.g. WHERE name LIKE '%Lee'
  - % 0至n的多符
  - _ 1个字符
  - e.g. WHERE name LIKE '___Lee'

NULL values

e.g. WHERE x 为 NULL

WHERE x 为 NOT NULL

△ ORDER BY

```
SELECT * FROM People ORDER BY FirstName DESC, YearOfBirth ASC
```

descend　ascend 默认

先排前面，后排后面

△ AS

```
SELECT x AS 'new_name1', x * y AS 'new_name2' FROM ...
```

← Column alias

可以 rename column，也可以 rename function

△ DISTINCT

- SELECT DISTINCT FRUIT_NAME FROM Fruits

用于去除重复的 Row

⚠COUNT( )

COUNT (xxx)

COUNT (*)

COUNT (DISTINCT xxx) (跳过不是总计算) — col、cols、distinct col、*、func.

⚠GROUP BY

```
SELECT item, SUM (quantity * price) AS sales
        FROM salesitem GROUP BY item
        ORDER BY item ASC
```

item 主体 ← 研究对象

只能 GROUP BY 列，不能 GROUP BY 函数

⚠HAVING

```
SELECT item, SUM (quantity * price) AS sales
      FROM salesitem GROUP BY item
      HAVING SUM (quantity * price) > 1000
```

当条件是个function时，我们需要用HAVING来代替WHERE

GROUP BY 之后只能接 HAVING

⚠Subquery (SELECT( ) → 后)

```
SELECT [column_names]
        FROM [tablename | (SELECT...)
        WHERE [criteria <=> (SELECT...)]
```

b. Action query

⚠INSERT INTO

```
INSERT INTO tablename
        [(col1, col2, col3...)]
        VALUES (value1, value2, value3...)
```

⚠UPDATE

```
UPDATE tablename
      SET col1 = value1, col2 = value2,...
      WHERE criteria (is true)   e.g., where StudentID = "102012"
```

⚠DELETE FROM

```
DELETE FROM tablename
        WHERE criteria (is true)
```

L11-12 SQL Advanced

1. Join (连接两个表)

① Cross join

ProjID	EmpID	Hours
A	001	16
B	003	28

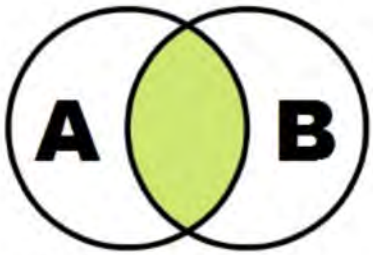
EmpID	EmpName
001	Anna
002	Bella
004	Donna

ProjID	EmpID	Hours	EmpID	EmpName
A	001	16	001	Anna
A	001	16	002	Bella
A	001	16	004	Donna
B	003	28	001	Anna
B	003	28	002	Bella
B	003	28	004	Donna

```
SELECT      [column names]
FROM        [Table1]
CROSS JOIN  [Table2]
```

```
SELECT [column names]
FROM   [Table1], [Table2]
```

② Inner join




```
SELECT *
FROM A
INNER JOIN B
ON A.id = B.id
```

ProjID	EmpID	Hours
A	001	16
B	003	28

EmpID	EmpName
001	Anna
002	Bella
004	Donna

ProjID	EmpID	Hours	EmpID	EmpName
A	001	16	001	Anna

Repeated column for the matching key

```
SELECT column_names FROM table1, table2
    WHERE table1.col1 = table2.col2
```

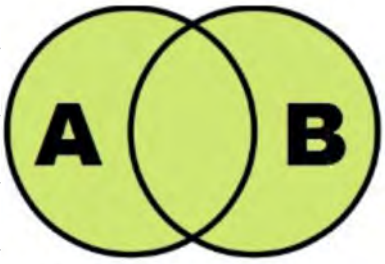
(old-style)

当多张表时，则用link连接：WHERE (a.PK = b.FK) AND (b.PK = c.FK) AND ...

```
SELECT column_names FROM table1 JOIN table2
    ON table1.col1 = table2.col2
```

```
Select sname from student, instructor where (student.advisorid = instructor.iid) and (instructor.taught>30)
Select count(*) from student join instructor on student.advisorid = instructor.iid where instructor.name = 'Tess Pearson'
```

③Full outer join (SQLite不支持)




```
SELECT *
FROM A
FULL OUTER JOIN B
ON A.id = B.id
```

ProjID	EmpID	Hours
A	001	16
B	003	28

EmpID	EmpName
001	Anna
002	Bella
004	Donna

ProjID	EmpID	Hours	EmpID	EmpName
A	001	16	001	Anna
B	003	28		
			002	Bella
			004	Donna

Cross join vs Full outer join

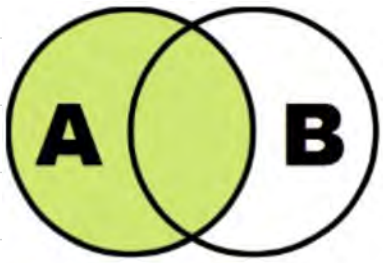
ProjID	EmpID	Hours
A	001	16
B	003	28

EmpID	EmpName
001	Anna
002	Bella
004	Donna

ProjID	EmpID	Hours	EmpID	EmpName
A	001	16	001	Anna
A	001	16	002	Bella
A	001	16	004	Donna
B	003	28	001	Anna
B	003	28	002	Bella
B	003	28	004	Donna

ProjID	EmpID	Hours	EmpID	EmpName
A	001	16	001	Anna
B	003	28		
			002	Bella
			004	Donna

④Left outer join



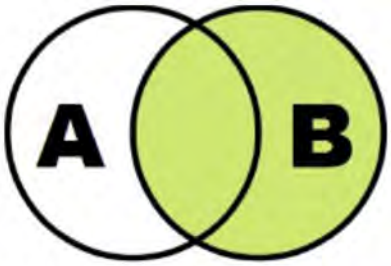

```
SELECT *
FROM A
LEFT JOIN B
ON A.id = B.id
```

ProjID	EmpID	Hours
A	001	16
B	003	28

EmpID	EmpName
001	Anna
002	Bella
004	Donna

ProjID	EmpID	Hours	EmpID	EmpName
A	001	16	001	Anna
B	003	28		

⑤ Right outer join (SQLite 不支持)




```
SELECT *
FROM A
RIGHT JOIN B
ON A.id = B.id
```

ProjID	EmpID	Hours
A	001	16
B	003	28

EmpID	EmpName
001	Anna
002	Bella
004	Donna

⇩

EmpID	EmpName	ProjID	EmpID	Hours
001	Anna	A	001	16
002	Bella			
004	Donna			

Residual information

```
SELECT table1.col1, table2.col2, ...
    FROM table1 [LEFT|RIGHT] OUTER JOIN table2
    ON table1.col1 = table2.col2
```

2. CREATE VIEW ... AS ...

```
CREATE VIEW view_name AS
    SELECT column1, column2, ...
    FROM table_name
    WHERE condition;
```

创建了一个虚拟(临时)的表

```
CREATE VIEW [Brazil Customers] AS
SELECT CustomerName, ContactName
FROM Customers
WHERE Country = 'Brazil';
```

L13-14 Introduction to Data Mining

1. Definition

Data mining ≈ KDD (Knowledge Discovery in Database)

2. Types of analysis

① Descriptive analysis

找 x-y 关系

② Predictive analysis

通过 x 预测 y

3. CRISP-DM Model 跨行业标准的数据挖掘流程

① Definition

CRoss Industry Standard Process for Data Mining

② Steps

a. Business understanding

Context

b. Data understanding

What data we need to get, where to get it, and how?

c. Data preparation

- Duplicate data
- Missing data
- Outlier data

→

- Data cleaning
- Data transformation
- Data sampling

d. Modeling

DM Task	Learning Method	Popular Algorithms
Regression	Supervised	Linear/nonLinear Regression, SVM
Classification	Supervised	Decision trees, SVM, logistic regression, KNN
Clustering	Unsupervised	K-means, DBSCAN, ANN/SOM
Association	Unsupervised	Apriori, FP-Growth

有 label

无 label

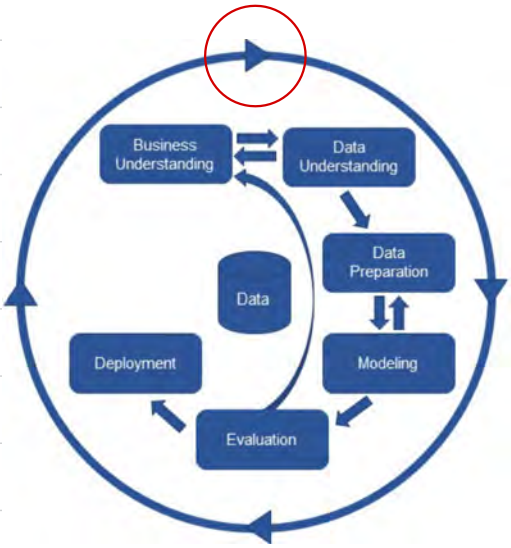
e. Evaluation

Find the best model

- Hold-out validation
- Cross-validation

f. Deployment

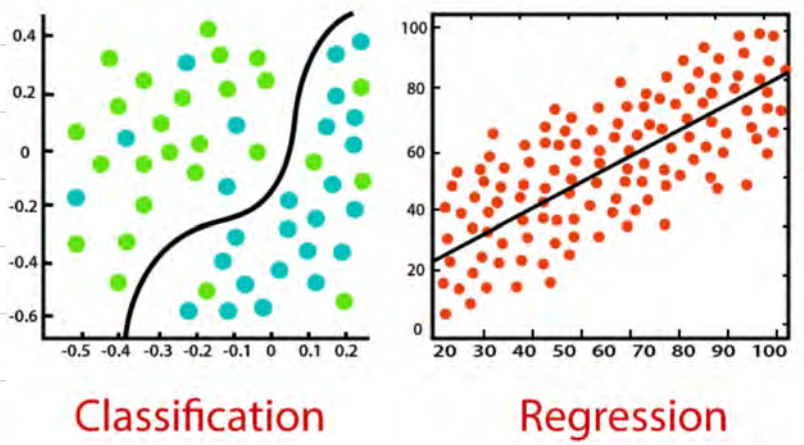
a→e 之间的跌 repetitive的



4. Intro to ML / Predictive analysis

① Types

- Regression ⟶ y is Cont
- Classification ⟶ y is disc



② Concepts

a. Overfitting

- Training dataset 拟合得过好
- Test dataset 拟合得不好

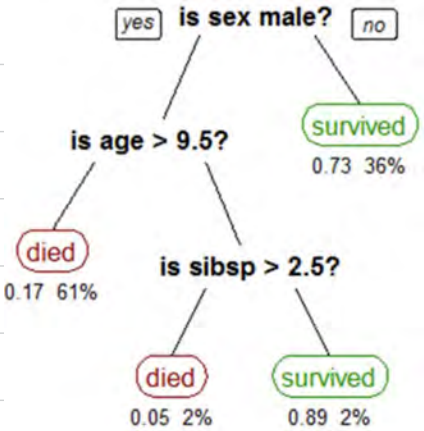
⟶ 因为有太多 Training dataset 不具有的点

- Stat 不 split dataset ⟶ 不考虑过拟合
- ML split dataset ⟶ 考虑过拟合

b. Rule-based model (e.g. Decision tree)

For instance, if the student height > 185, the hair color = blonde, the eye color = blue and the shoe size > 44, then the student = swinish

- 红: Variables
- 蓝: Threshold
- 绿: Prediction



(Algorithm ≠ Model: Algorithm is the tool to get model)

L15 Classification

1. Overview of Classification

① Definition

<x, y>

Attribute/Feature　　Label / Target variable

② Types of Classfiers (Algorithms)

a. Base classfiers

- **Decision Tree based Methods**
- Rule-based Methods
- Nearest-neighbor
- Neural Networks/Deep Learning
- Naïve Bayes/Bayesian Belief Networks
- Logistic Regression/Support Vector Machines

b. Ensemble classfiers (集中到一起)

- Boosting, Bagging, Random Forests

2. Decision Tree

① Merit

- Easy to setup
- Easy to interpret
- Computationally cheap
- Robust to noise

- Almost all data mining packages include DTs

② Concepts

- Root node: no parent, zero or more children
- Internal node: one parent, two or more children
- Leaf node/terminal node: one parent, no child

③ The order of attributes ⟶ Based on impurity

a. Different measures of node impurity

- Gini Index

$$GINI(t) = 1 - \sum_j [p(j|t)]^2$$

- Entropy

$$Entropy(t) = -\sum_j p(j|t) \log p(j|t)$$

$\Sigma - p_i \cdot \log_2 p_i$

- Misclassification error

$$Error(t) = 1 - \max_i P(i|t)$$

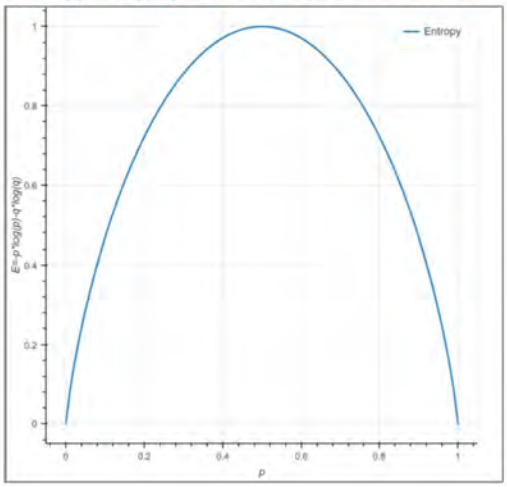
b. More on Entropy (详细) 无序程度 ⟶ 不确定性

⚠ Computation

Entropy = $\Sigma - p_i \cdot \log_2 p_i$ (i 指的是第 i 个 class)

⚠ Properties

Entropy and proportion for a two-class variable



(i) 关于 p = 0.5 对称

(ii) Entropy ∈ [0, 1]

(X, 0) (X, X)

c. Criterion of choosing nodes

We chose the nodes (attributes) that make class purer

⚠ max Gain = P − M

P: 前 impurity

M: 后 impurity (Weighted average: Weight 是分类后子类内部个数)

d. Stopping criteria

⚠ Use up all the attributes

⚠ Early-stop ⟶ 有3个条件:

1. No attribute satisfies a minimum information gain (再分类也无 Gain)
2. A Maximal tree depth is reached ⟶ If too deep, then we have the risk of Overfitting
3. There are less than a certain number of examples in the current subtree

2.1b Model Evaluation

1. Criterion

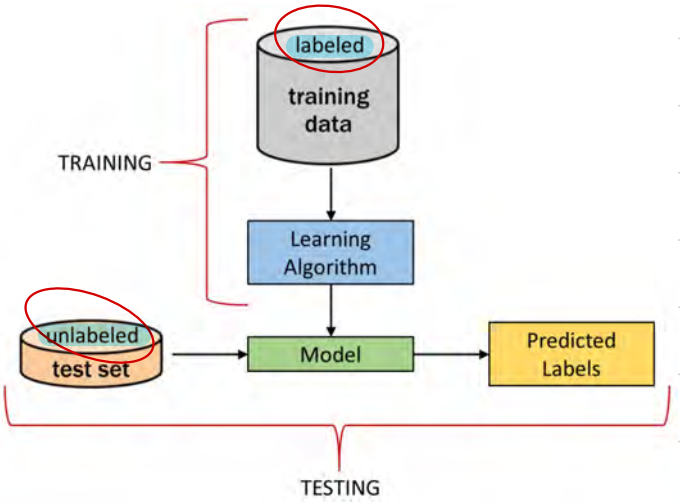
Find the balance between Underfitting & Overfitting ⟶ To successfully generalize the relationship between x & y

2. Tests

① Training set

② Validation set (用于adjust parameter, before evaluation)

③ Test set



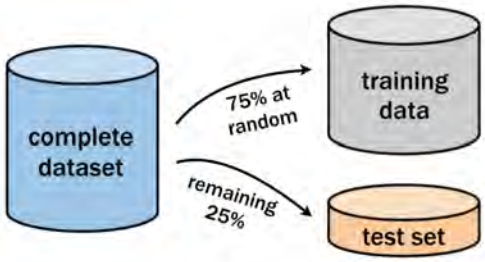
3. Problems

① Dataset is small

② Sample is under-representative ⟶ More noise

4. Validation method

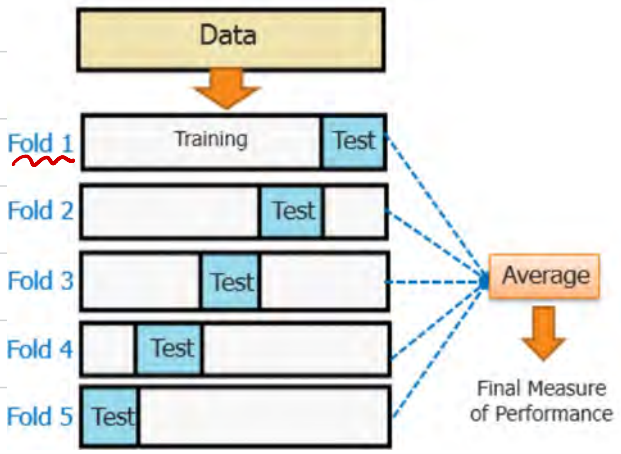
① Holdout validation

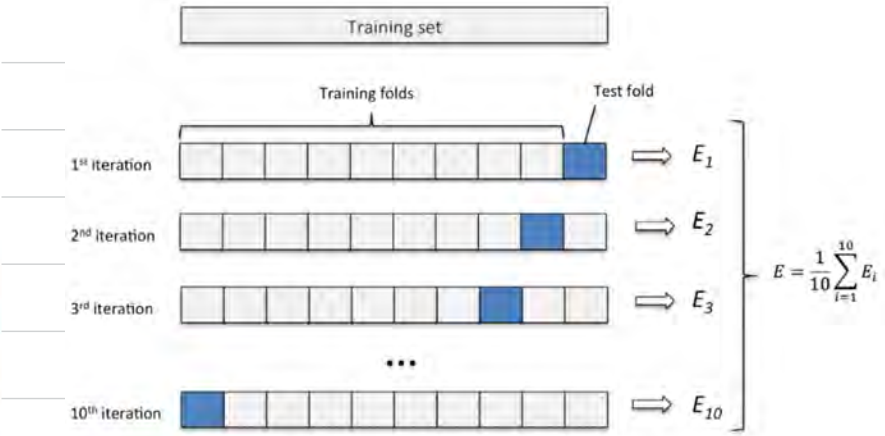


75% (Randomly)

25%

② Cross validation





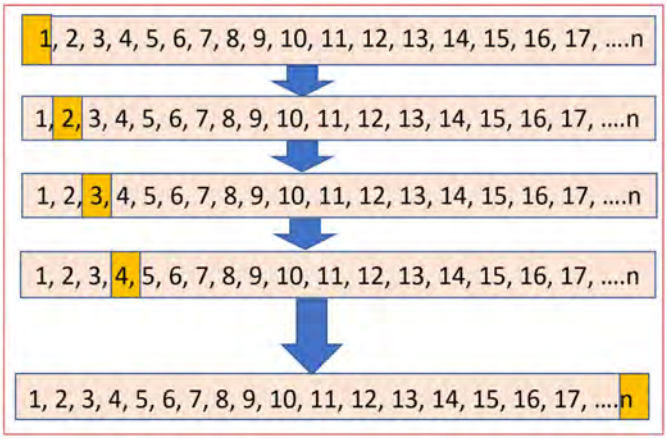
通常 # of folds = 5 | 10

Cross validation 比 Holdout validation 更好：

- Training set 大小不会减小
- Training set 与 Test set 互动更多

③ The Leave-one-out validation

Take # of folds = k, k is # of data



5. Evaluation measure

① Confusion matrix

a. Setup

		Actual class (observation)	
		yes	**no**
Predicted class (expectation)	**yes**	TP	FP ← Type 1 error
	no	FN	TN

Type 2 error

b. Measure

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+FN+TN}$$

When "Class imbalance problem" (eg. $C_0 = 99$, $C_1 = 1$), then it's no a good measure.

总是会有高的Accuracy，是不对的

$$\text{Precision (p)} / \text{Positive predictive value (PPV)} = \frac{TP}{TP+FP}$$

$$\text{Recall (r)} / \text{Sensitivity} / \text{TP rate} = \frac{TP}{TP+FN}$$

$$\text{F-measure (F)} = \frac{2TP}{2TP+FP+FN}$$

一般都会综合以上几种 matrices 进行分析.

c. With cost matrix

Model 1	Actual class (observation)		
Predicted class (expectation)		yes	no
	yes	3	3
	no	2	7

Model 2	Actual class (observation)		
Predicted class (expectation)		yes	no
	yes	3	1
	no	4	7

Cost Matrix	Actual class (observation)		
Predicted class (expectation)		yes	no
	yes	-1	5
	no	20	0

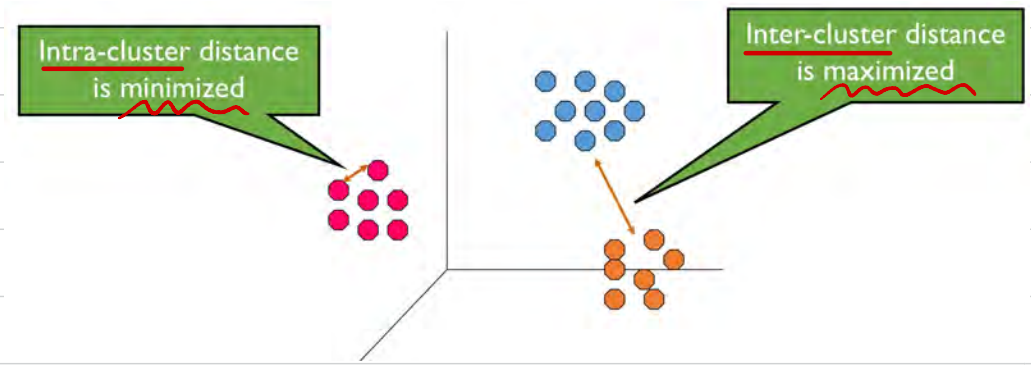
C(M1) = -(3x1) + (3x5) + (2x20) + (7x0) = 52
C(M2) = -(3x1) + (1x5) + (4x20) + (7x0) = 82.

Total costs

17-18 Clustering

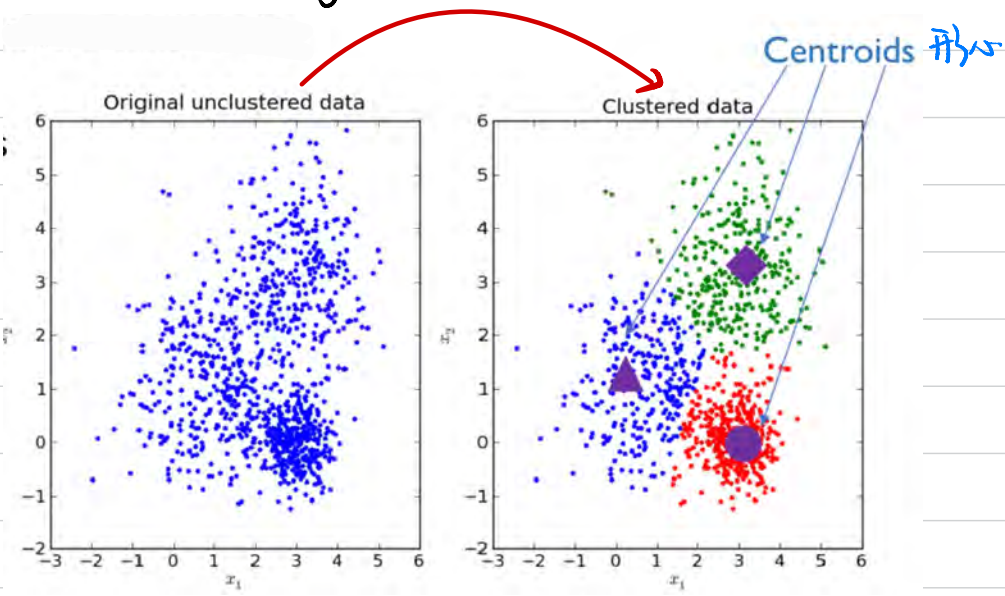
1. Goal

- Maximize similarity within a cluster
- Maximize dissimilarity between clusters



2. Types

① Center-based clustering



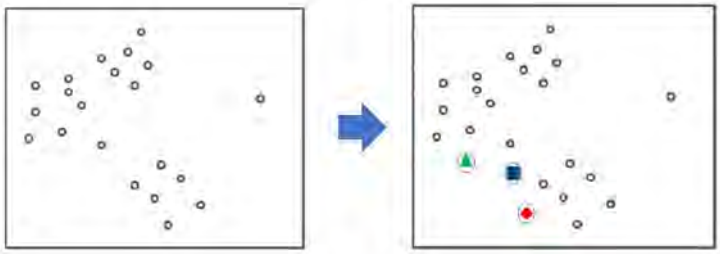
② K-Means clustering

a. Definition

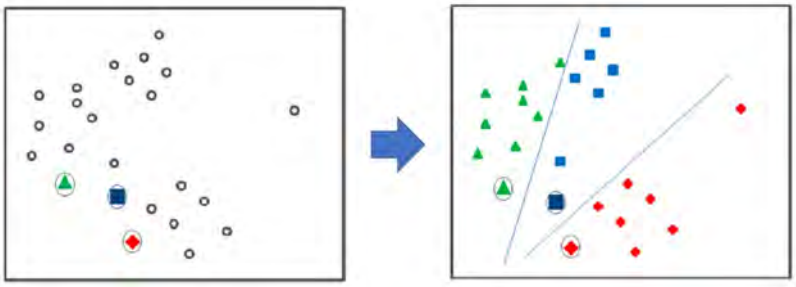
K = # of centroids = # of exclusive clusters

b. Process

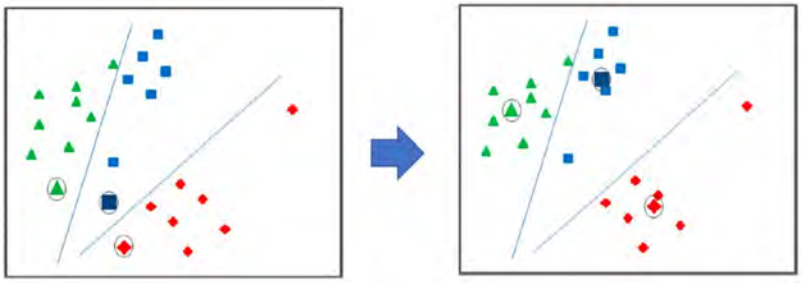
① Initiate K centroids 由于 initial 的 centroids 不同，clustering 的结果也会不同



② Assign each data point to form clusters according to Euclidian distance 也可以是其他 distance



③ Choose the mean of each cluster as new centroid



④ Repeat Step ② & ③ until convergence ≙ No further change in assignment of data points

(Few 也可以)

c. Evaluating

⚠ Minimize SSE (With sum of squares) ⟶ Cluster cohesion

- C_i is the i^{th} cluster
- x is the data point in C_i
- m_i is the representative point for cluster C_i which is also the center/mean of the C_i

$$SSE = \sum_{i=1}^{K} \sum_{x \in C_i} dist^2(m_i, x)$$

The objective function (cost function) of the method is to minimize the SSE.

⟶ 比較不同的clustering model

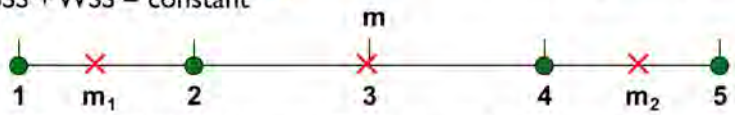
⚠ Maximize BSS (Between sum of squares) ⟶ Cluster separation

$$BSS = \sum_i |C_i|(m - m_i)^2$$

- Where $|C_i|$ is the size of cluster *I*
- *mi is the mean of cluster i (centroid)*
- *m is the grand mean*

- Example: SSE
  - BSS + WSS = constant

Minimizing WSS (cohesion) is equivalent to maximizing BSS (separation)



K=1 cluster:

$$SSE = WSS = (1-3)^2 + (2-3)^2 + (4-3)^2 + (5-3)^2 = 10$$

$$BSS = 4 \times (3-3)^2 = 0$$

$$Total = 10 + 0 = 10$$

K=2 clusters:

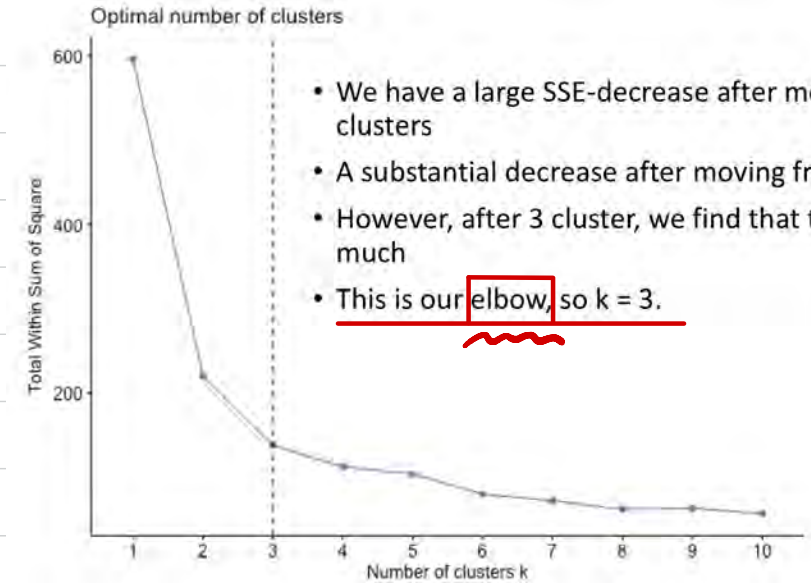
$$SSE = WSS = (1-1.5)^2 + (2-1.5)^2 + (4-4.5)^2 + (5-4.5)^2 = 1$$

$$BSS = 2 \times (3-1.5)^2 + 2 \times (4.5-3)^2 = 9$$

$$Total = 1 + 9 = 10$$

⚠ How to choose K ?

a. Elbow method



- We have a large SSE-decrease after moving from 1 cluster to 2 clusters
- A substantial decrease after moving from 2 to 3 clusters
- However, after 3 cluster, we find that the SSE does not decrease much
- This is our elbow, so k = 3.

L19 Association

1. Association rule

① Definition

$X \rightarrow Y$

② Remarks

X & Y are item set : X is called premise, Y is called conclusion

$X \cap Y = \phi$

③ Examples

- {diaper} → {beer}
- {news pages, finance pages} → {sports pages}
- {youth book, reference book} → {child book, geography book}.

item

item set

2. Support count

① Support count

Absolute frequency of an itemset

- σ ({news, finance}) = 3
- σ ({news}) = 4

② Support

Relative frequency of an itemset

- S({news, finance}) = 3/6
- S({news}) = 4/6

> • A Frequent itemset is an itemset whose support is greater than or equal to a **minsup** (a threshold defined by the user)
> • i.e., minsup (minimum support) is a parameter.

③ Confidence

Measure how often items in Y appear in transactions that contain X

$$c = \frac{\sigma(X, Y)}{\sigma(X)}$$

- E.g. {news, finance} → {sports}
- $s = \frac{\sigma(\{news, finance, sports\})}{N} = \frac{1}{6} \approx 0.167$
- $c = \frac{\sigma(\{news, finance, sports\})}{\sigma(\{news, finance\})} = \frac{1}{3} \approx 0.33$

3. Task

Given a set of transactions T, the goal of association rule learning is to **find all rules** having:

- Support ≥ minsup threshold
- Confidence ≥ minconf threshold

4. Approaches to learn rules

① Brute-force approach 1.0

1. List all possible association rules
2. Compute the support and confidence for each rule
3. Prune rules that fail the *minsup* and *minconf* thresholds

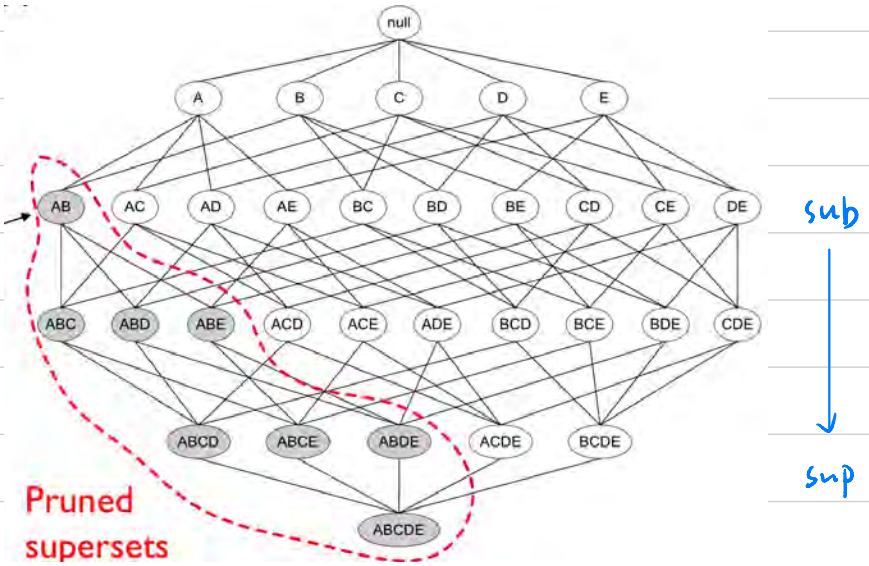
② Brute-force approach 2.0

1. Generate all itemsets whose support ≥ minsup threshold (i.e. frequent itemset generation)
2. Generate rules which have high confidence from each frequent itemset (i.e., strong rule generation)

③ Aprior algorithm

Subset ⊆ Superset ⟶ Superset must be infrequent

Subset is infrequent



④ Rule generation algorithm

→ Find out candidates satisfying minSup

{news}, {finance}, {sports}, {arts}, {news, finance}, {finance, sports}

→ Rule out 1-item itemset

~~{news}, {finance}, {sports}, {arts},~~ {news, finance}, {finance, sports}.

1-item set 2-item set

→ Compute confidence

- An association rule can be extracted by **partitioning** the itemset Y
  - e.g., **Y** = {news, finance}
- into 2 non-empty subsets, **X** and **Y – X**
  - e.g. **X** = {news}
  - e.g. **Y – X** = {news, finance} - {news} = {finance}

- such that the association **X → Y –X** satisfies a given **confidence threshold.**
- {news} → {finance}: $c = \frac{\sigma(\{news, finance\})}{\sigma(\{news\})} = \frac{3}{4} = 0.75$
- {finance} → {news}: $c = \frac{\sigma(\{news, finance\})}{\sigma(\{finance\})} = \frac{3}{4} = 0.75$
- {finance} → {sports}: $c = \frac{\sigma(\{sports, finance\})}{\sigma(\{finance\})} = \frac{2}{4} = 0.5$
- {sports} → {finance}: $c = \frac{\sigma(\{sports, finance\})}{\sigma(\{sports\})} = \frac{2}{2} = 1$

→ Pattern evaluation (Problem)

{finance} → {news}: $c = \frac{\sigma(\{news, finance\})}{\sigma(\{finance\})} = \frac{3}{4} = 0.75$

- Confidence: P(news|finance) = 3/4 = 0.75
- Confidence > 50%, meaning people who read finance pages are more likely to read news pages than not reading finance pages
- This rule seems reasonable, right? Is anything wrong here?
- Yes, **P(N|F) ≠ 1 – P(N|F$^-$).**

→ Pattern evaluation (Solution)

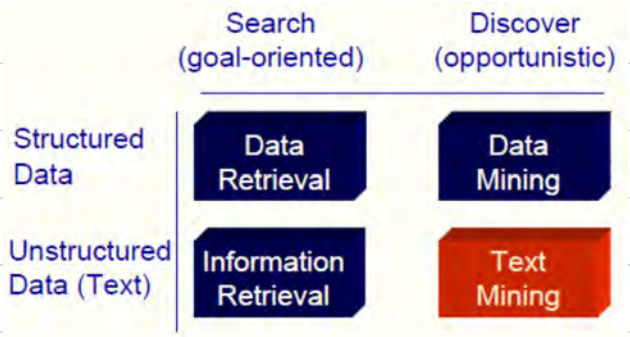
$$lift(X \rightarrow Y) = \frac{c(X \rightarrow Y)}{s(Y)} \text{ or } \frac{P(Y|X)}{P(Y)}$$

$$= \frac{f_{11}/f_{1+}}{f_{+1}/N} = N\left(\frac{f_{11}}{f_{+1}f_{1+}}\right) = \frac{P(X,Y)}{P(X)P(Y)}$$

- If P(X,Y) > P(X) * P(Y), lift > 1 : X & Y are **positively** correlated
- If P(X,Y) < P(X) * P(Y), lift < 1 : X & Y are **negatively** correlated.

L21-22 Text Mining

1. Overview



2. Text classification

① Definition

input (Text) ⟶ output (Classes)

② Naive Bayes

Naive Bayes is a probabilistic model: $P(x|Y=y)$

a. Assumption

Words are Conditional independent (Naive): $P(w_1,\cdots,w_N|Y=y) = \prod_i P(w_i|y)$ ⟶ Fast

b. Process

△ Compute conditional probability

Example: Movie Reviews

No.	Movie Review	Y
1	The movie is great!	1
2	I like the movie.	1
3	I hate the movie.	0
4	Like it! It is great!	1

- Here we have 4 movie reviews, with labels.
- We want to learn how to classify reviews as positive or negative (1/0).
- We have a bag of words with 8 unique words.
- We use these as our features for classification (the scale here is discrete, i.e., count).

8 unique words: {the, movie, is, great, I, like, hate, it}

Doc	the	movie	is	great	i	like	hate	it	Y
1	1	1	1	1					1
2	1	1			1	1			1
3	1	1			1		1		0
4			1	1		1		2	1

The prior probability is the probability for each outcome: $P(1) = 3/4$ Prior probability
$P(0) = 1/4$

The likelihood is the probability of each word given the outcome (i.e., the label):
$P(the|1), P(movie|1), P(is|1), P(great|1), P(i|1), P(like|1), P(hate|1), P(it|1)$
$P(the|0), P(movie|0), P(is|0), P(great|0), P(i|0), P(like|0), P(hate|0), P(it|0)$

△ Do additive smoothing for the conditional probability Likelihood

$$P(w_i|y) = \frac{count(w_i,y)}{count(y)}$$

→ $$P(w_i|y) = \frac{count(w_i,y)+1}{count(y)+N_{unique}}$$ Smoothing

N_{unique}: word 的个数

△ Judge new text

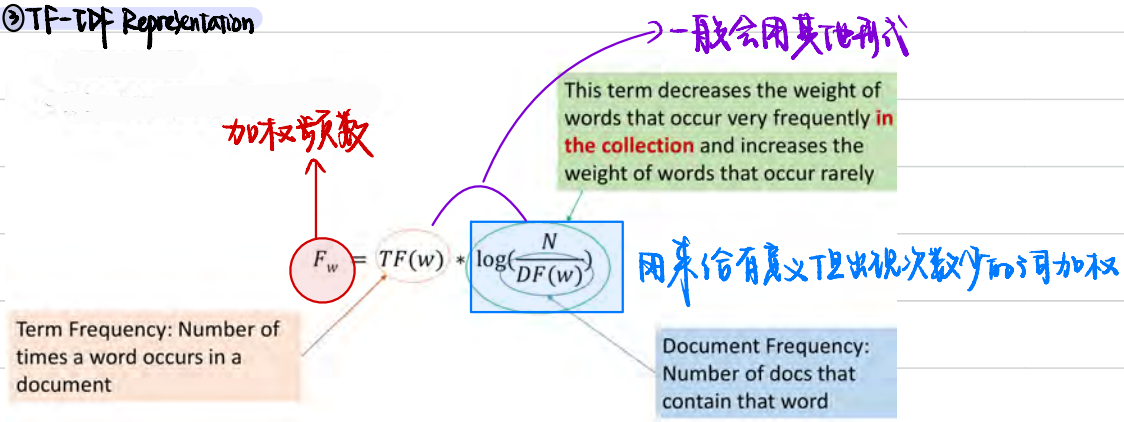
A new movie review: Great movie!

Classify the movie review according to: $f(x)$

If class = 1, $P(1) * P(great|1) * P(movie|1) \approx 0.0556$
If class = 0, $P(0) * P(great|0) * P(movie|0) \approx 0.0062$ ➡ class = 1

↑ prior pro ↑ likelihood

③TF-IDF Representation



Example: Movie Reviews (with TF-IDF)

No.	Movie Review	Y
1	The movie is great!	1
2	I like the movie.	1
3	I hate the movie.	0
4	Like it! It is great!	1

Now we go back to the movie review example and we apply the TF-IDF representation.

8 unique words: {the, movie, is, great, I, like, hate, it}

Doc	the	movie	is	great	i	like	hate	it	Y
1	1*log(4/3)	1*log(4/3)	1*log(4/2)	1*log(4/2)					1
2	1*log(4/3)	1*log(4/3)			1*log(4/2)	1*log(4/2)			1
3	1*log(4/3)	1*log(4/3)			1*log(4/2)		1*log(4/1)		0
4			1*log(4/2)	1*log(4/2)		1*log(4/2)		2*log(4/1)	1

④Bag-of-words model

a. Procedure

- The first step is to create a list of unique words

It was the best of times,
it was the worst of times,
it was the age of wisdom,
it was the age of foolishness,

- "it"
- "was"
- "the"
- "best"
- "of"
- "times"
- "worst"
- "age"
- "wisdom"
- "foolishness"

- Now I can create a document vector for each sentence


```
"it was the best of times" =  [1, 1, 1, 1, 1, 1, 0, 0, 0, 0]
"it was the worst of times" = [1, 1, 1, 0, 1, 1, 1, 0, 0, 0]
"it was the age of wisdom" = [1, 1, 1, 0, 1, 0, 0, 1, 1, 0]
"it was the age of foolishness" = [1, 1, 1, 0, 1, 0, 0, 1, 0, 1]
```

b. Pros & Cons

Pro: Unstructured text → Structured text

Con: Vocabulary
Sparsity
Meaning

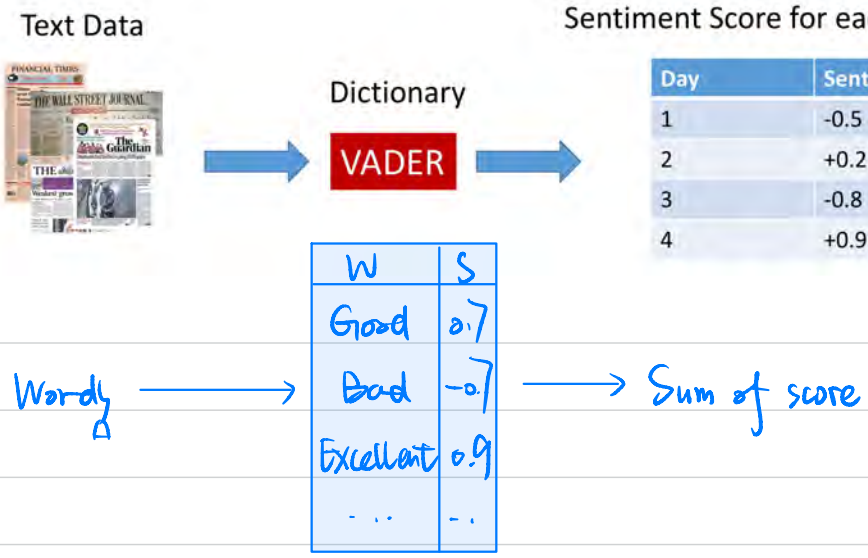
⑤ Sentiment analysis (本质上是classification)

a. Dictionary

Text —Dic→ Label ∈ [-1, 1] 是个范围

Dictionary is dataset-specific: 不同领域或应用不同的Dictionary

b. Process



L23-24 Social Network Analysis

1. Social Network

① Definition

Graph dataset (Heterogeneous
Multi-relational

② Small-world Network

a. Motivation: Why we look at small-world network

Often happen in real life

b. Definition

Combination of regular & random networks

	Regular	Small-world	Random
Degree of local clustering	High	High	Low
Degree of separation	High	Low	Low

→ 近的有联系，远的也有联系

c. Property

Dynamic: Nodes & Links are added and deleted over time

2. Some basics of Social Network Analysis (SNA)

① Notation

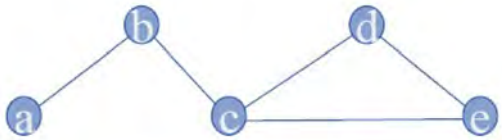
- Node/actor/vertex: V
- Link/tie/edges: E
- Network: G(V, E)

- Null Graph: $G(V,E), V = E = \emptyset$
- Empty Graph: $G(V,E), E = \emptyset$
- Directed Graph: $G(V,E), E_{i,j} \neq E_{j,i}$
- Undirected Graph: $G(V,E), E_{i,j} = E_{j,i}$
- Weighted Graph: $G(V,E,W), |E| = |W|, W$: E → N or R The #edges = #weights
- Simple graphs vs. Multigraphs: *whether multiple edges between two nodes can exist.*

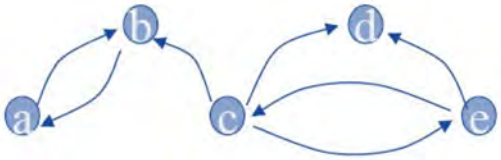
② Type of Network graph

Binary / Valued

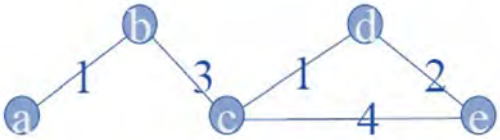
Undirected / Directed



Undirected, binary



Directed, binary

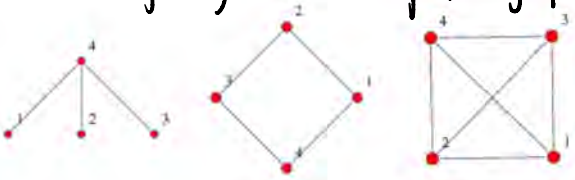


Undirected, Valued

Directed, Valued

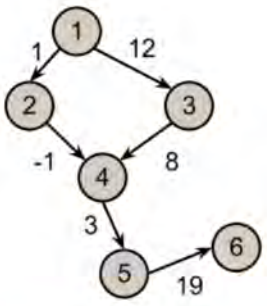
③ Mathematical representation of graph

We use adjacency matrix to represent graph



$$\begin{pmatrix} 0&0&0&1\\ 0&0&0&1\\ 0&0&0&1\\ 1&1&1&0 \end{pmatrix} \quad \begin{pmatrix} 0&1&0&1\\ 1&0&1&0\\ 0&1&0&1\\ 1&0&1&0 \end{pmatrix} \quad \begin{pmatrix} 0&1&1&1\\ 1&0&1&1\\ 1&1&0&1\\ 1&1&1&0 \end{pmatrix}$$

Undirected graph



Weighted Directed Graph

	1	2	3	4	5	6
1	0	1	12	0	0	0
2	-1	0	0	-1	0	0
3	-12	0	0	8	0	0
4	0	1	-8	0	3	0
5	0	0	0	-3	0	19
6	0	0	0	0	-19	0

Adjacency Matrix

被指的看行次数

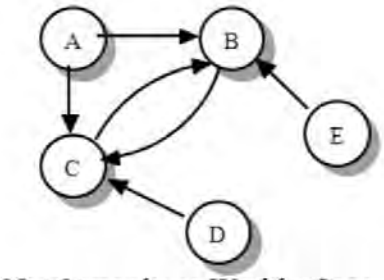
④ Connectivity

a. Connected node & graph

- Connected node: 被连接的点
- Connected graph: 任意两点都有连接

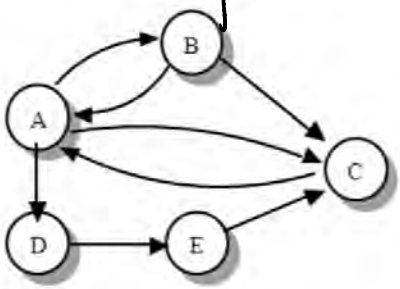
b. Strongly / Weakly connected directed graph 有向

- Strongly connected: 任意两点都有同连接
- Weakly connected: 任意两点都有连接，但不满足都有同连接



Not Strongly or Weakly Connected
(No path E to D or D to E)

(a)



Strongly Connected

(b)

⑤ Component

Connected subgraph of a graph

⑥ Diameter

Longest shortest path of a graph

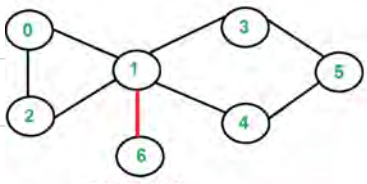
- The lengths of the shortest path between any 2 nodes are
- AB = 1
- AC = 1
- BC = 1
- BD = 1
- CD = 1
- AD = 2
- So the longest is AD = 2
- This is the diameter.

⑦ Bridge & Articulation

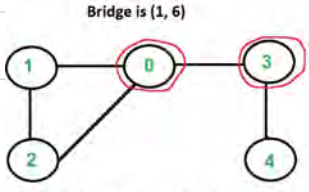
- Bridge: Connecting edge
- Articulation: Connecting node

↓

除去会使得graph disconnected



Bridge is (1, 6)

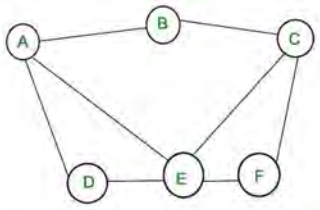


Articulation points are 0 and 3

⑧ Metrics

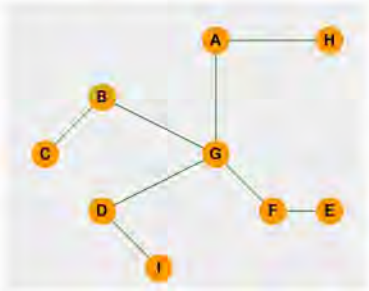
- Size and density: size $= N$; density $= \frac{\#links}{\left(\frac{N(N-1)}{2}\right)}$
- Nodes' degree: the number of edges incident to each node
- Distance between a pair of nodes: the shortest-path length of the two nodes
- Network diameter: the maximum distance between pairs of nodes
- Average distance: the mean of the distances between pairs
- Effective diameter: the minimum distance, d, such that for at least 90% of the reachable node pairs, the path length is at most d.

1. A's degree
2. E's degree
3. Network diameter
4. Number of articulation nodes



Q1: 3
Q2: 4
Q3: 3
Q4: 0

1. A's degree
2. G's degree
3. Network diameter
4. Number of articulation nodes



Q1: 2
Q2: 4
Q3: 4
Q4: 5

⑨ Measures of centrality

- Degree centrality is the number of links incident upon a node (i.e., the degree of the node)
- Closeness centrality: the reciprocal of the sum of the length of the shortest-path between the node and all other nodes in the graph
- Betweenness centrality is the number of times a node acts as a bridge along the shortest path between two other nodes
- Eigenvector centrality: a node's centrality is a function of its neighbor's centralities.

a. Degree centrality

Degree Centrality, of a node v:

$$C_D(v) = \deg(v)$$ ← # of edges incident upon node v

In Python this is normalized by dividing by n – 1 ⟹ $C_D(v) = \frac{\deg(v)}{N-1}$

b. Closeness centrality

Closeness Centrality, of a node x:

$$C(x) = \frac{1}{\sum_y d(y,x)}$$ ← Distance between nodes y and x

This is usually normalized by multiplying by n – 1 ⟹ $C(x) = \frac{N-1}{\sum_y d(y,x)}$

c. Betweenness centrality

Betweenness Centrality, of a node v:

$$C_B(v) = \sum_{s \neq v \neq t \in V} \frac{\sigma_{st}(v)}{\sigma_{st}}$$

$\sigma_{st}(v)$: # of shortest paths from node s to node t, passing through v

σ_{st}: # of shortest paths from node s to node t

Note that here the paths to be considered exclude any path to or from v

d. Eigenvector centrality

Eigenvector Centrality, of a node v for a given graph G:

$$x_v = \frac{1}{\lambda} \sum_{t \in M(v)} x_t = \frac{1}{\lambda} \sum_{t \in G} a_{v,t} x_t$$

- λ: a predefined value for lambda (a constant)
- x_t: centrality the vertexes connected to v
- $a_{v,t}$: weight of the path between v and t (a matrix of 0/1)
- $M(v)$: a set of the neighbors of v
- x_v: relative centrality of vertex v

For Betweenness Centrality see:
https://www.youtube.com/watch?v=ptqt2zr9ZRE

⑩ Clustering coefficient

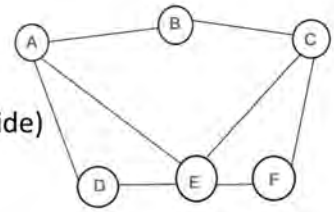
- It is a measure of the portions of i's neighbors that are connected together
- For a node i with degree k_i, the clustering coefficient is give by the following

$$C_i = \frac{2e_i}{k_i(k_i - 1)}$$

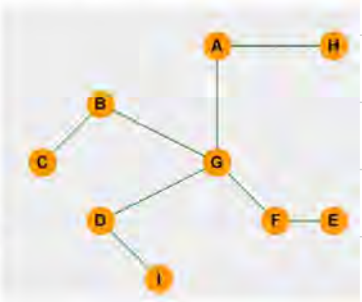
- Where e_i is the number of edges between the neighbors of node i

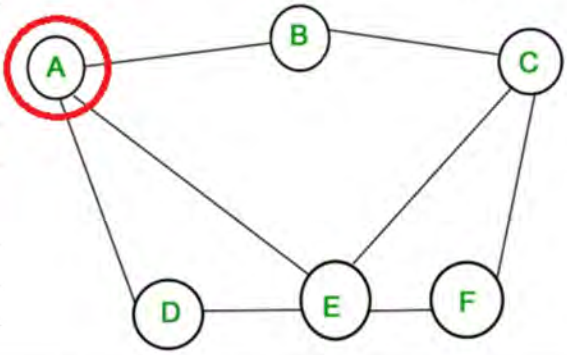
1. A's degree centrality
2. A's closeness centrality
3. A's betweenness centrality
4. Which node has the highest eigenvector centrality? (homework)
5. A's cluster coefficient

Q1: 3 (or 3/5)
Q2: 5/7 = 0.71
Q3: 1.5 (see next slide)
Q4: Need a PC
Q5: 1/3



Q1: 2 (or 2/8)
Q2: 0.47
Q3: 7
Q4: Need a PC
Q5: 0





Path	σ_{st}	$\sigma_{st(v)}$	$\sigma_{st(v)}/\sigma_{st}$
DB	1	1	1/1 = 1
DC	1	0	0
DE	1	0	0
DF	1	0	0
BC	1	0	0
BE	2	1	1/2 = 0.5
BF	1	0	0
CE	1	0	0
CF	1	0	0
EF	1	0	0
		SUM =	1+0.5 = 1.5