

第二讲：数据及数据分析基础

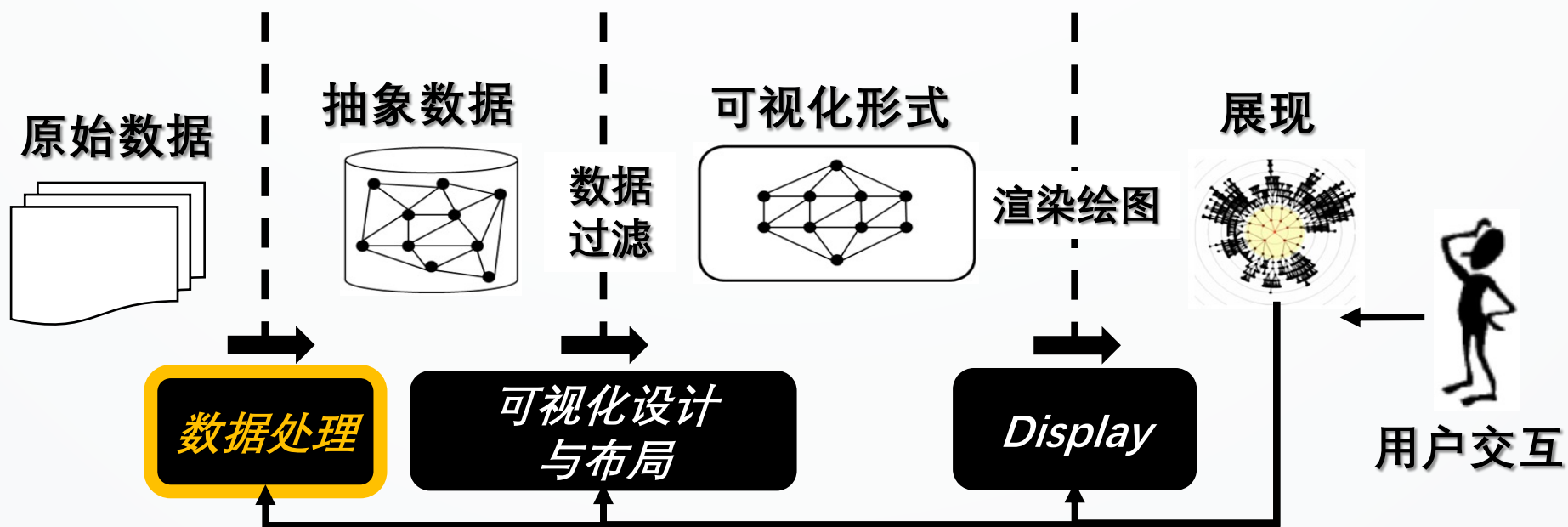
信息可视化

曹楠（教授）

<https://idvxlabs.com>

同济大学

怎样对数据进行可视化？

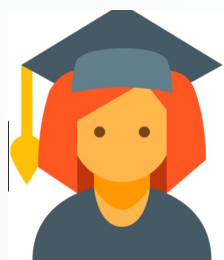


信息可视化参考模型

课程大纲

- 数据元素及元素的属性
- 数据元素之间的差异与关联
- 有监督学习与无监督学习
- 有监督学习：回归与分类
- 无监督学习：聚类分析

数据的维度



Student

[22, Male, 3000, 20, 30, 5]

Age

Sex

scholarship








Skills

Machine Learning
Data Mining Visualization

- 数据的维度，是数据中用于描述数据元素的各种属性。例如，在学生数据集中，一个学生可以通过她的年龄、性别、助学金的额度、以及学生在相关课程上的技能，通过花费在课程上的时间来衡量，等
- 真实世界中的数据一般都是多维度的

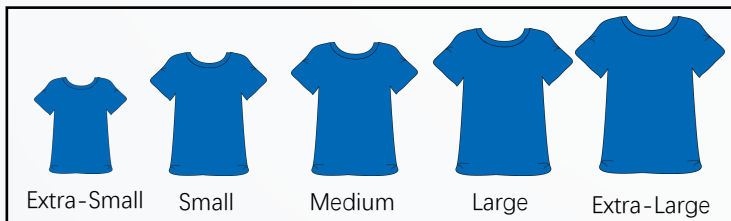
数据维度的类型

Nominal Data

Point	airport 	town 	mine 	capital 
Line	river 	road 	boundary 	pipeline 
Area	orchard 	desert 	forest 	water 

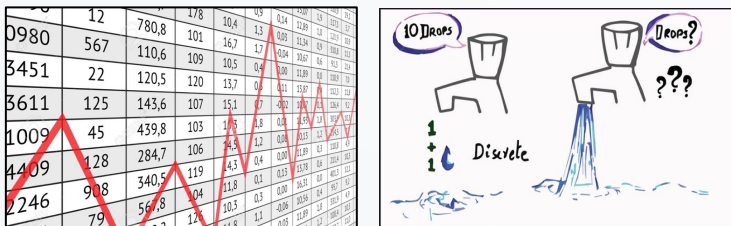
- 分类属性 (Nominal , Categorical): 该类型属性的取值代表了数据的类别, 且相互间是无法排序的

Ordinal



- 有序属性 (Ordinal) : 该类型属性的取值是有顺序的

Numerical



- 数值属性 (Ordinal) : 该类型属性的取值是数字。根据其取值的特点可以进一步分为 离散数值属性及连续的数值属性

一维数值属性的统计特征

- 均值 (Mean)
- 中位数 (Median)
- 方差 (Variance)
- 标准差 (Standard Deviation)

$$\bar{x} = \frac{\sum_{i=1}^N x_i}{N} = \frac{x_1 + x_2 + \cdots + x_N}{N}.$$

$$Q_{\frac{1}{2}}(x) = \begin{cases} x'_{\frac{n+1}{2}}, & \text{if } n \text{ is odd.} \\ \frac{1}{2}(x'_{\frac{n}{2}} + x'_{\frac{n}{2}+1}), & \text{if } n \text{ is even.} \end{cases}$$

$$\sigma^2 = \frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^2 = \left(\frac{1}{N} \sum_{i=1}^N x_i^2 \right) - \bar{x}^2,$$

$$SD = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - \mu)^2}$$

课程大纲

- 数据元素及元素的属性
- 数据元素之间的差异与关联
- 有监督学习与无监督学习
- 有监督学习：回归与分类
- 无监督学习：聚类分析

数据元素之间的差异性

- 数据元素的差异性是通过数据元素在各个维度上的取值的差异性来加以度量的
- 整个数据集所有元素两两之间的差异性可以用差异矩阵来表示：
 - 差异矩阵是一个对称矩阵
 - 差异矩阵的每一行及每一列代表数据集中的一个数据元素
 - 差异矩阵中的每一个取值，代表对应元素之间的数据距离
- 不同类型的数据维度之间有不同的距离计算方法

$$\begin{bmatrix} 0 & & & & \\ d(2,1) & 0 & & & \\ d(3,1) & d(3,2) & 0 & & \\ \vdots & \vdots & \vdots & \ddots & \\ d(n,1) & d(n,2) & \dots & \dots & 0 \end{bmatrix}$$

Dissimilarity Matrix

数据元素之间的差异性

- **分类属性** 之间的距离 通过 “不匹配率” 进行衡量

- p 在某一分类属性上所有可能的取值的总数
- m 代表数据元素在该属性上的取值相同情况的数目

$$d(i, j) = \frac{p - m}{p}$$

- 当分类属性只有两种可能的取值时（即为 binary），上述距离可以通过 “Jaccard coefficient” 计算：

$$d(i, j) = \frac{r + s}{q + r + s}$$

q: 两个二进制字符串中相互匹配的总位数

r: 第一个字符串中取值为1，但是在第二个字符串中取值为0的位数

s: 第二个字符串中取值为1，但是在第一个字符串中取值为0的位数

String1: 0100010

String2: 1000010

$q = 4, r = 1, s = 1$

$d(1, 2) = (1 + 1) / (4 + 1 + 1) = 1/3 = 0.333$

数据元素之间的差异性

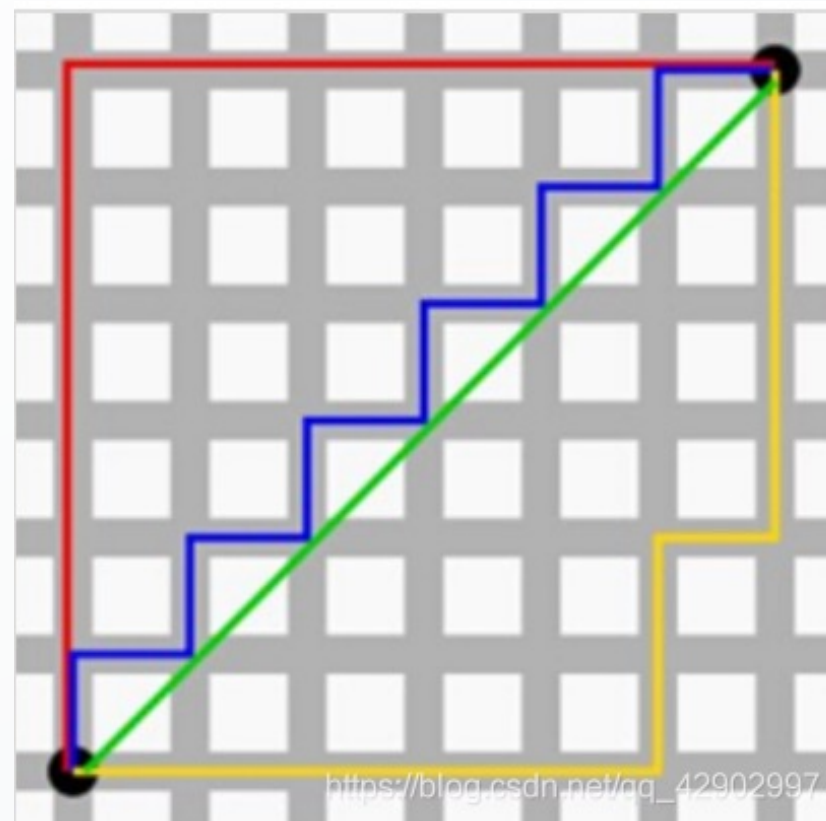
- 数值属性 之间的距离 通过 **明科夫斯基距离** 进行计算

- $P = (x_1, x_2, \dots, x_n), Q = (y_1, y_2, \dots, y_n)$

- 明科夫斯基距离 (L_q) -
$$\left[\sum_{i=1}^n |x_i - y_i|^q \right]^{\frac{1}{q}}$$

- 曼哈顿距离 (L_1) - $q = 1$


- 欧几里得距离 (L_2) - $q = 2$



数据元素之间的差异性


- 数据维度取值量的大小会影响 欧式距离 的大小
- 有的时候并不希望计入这样的差异，需要用到 “角距离”（Cosine Distance）：
例如，做文本分析时，为了计算文档与文档之间的差异性，文档的特征是通过文档中特定关键词出现的频次加以度量的。在这里，每一文档是一个数据元素，每一个关键词是数据元素的维度，而关键词在文档中出现的次数代表数据元素在该维度上的取值
- 如图所示，角距离度量的是两个向量之间夹角的大小（用 θ 表示），而不是空间中点A与点B的距离远近（用 d 表示），后者会因为向量的模的大小而改变。

Example:

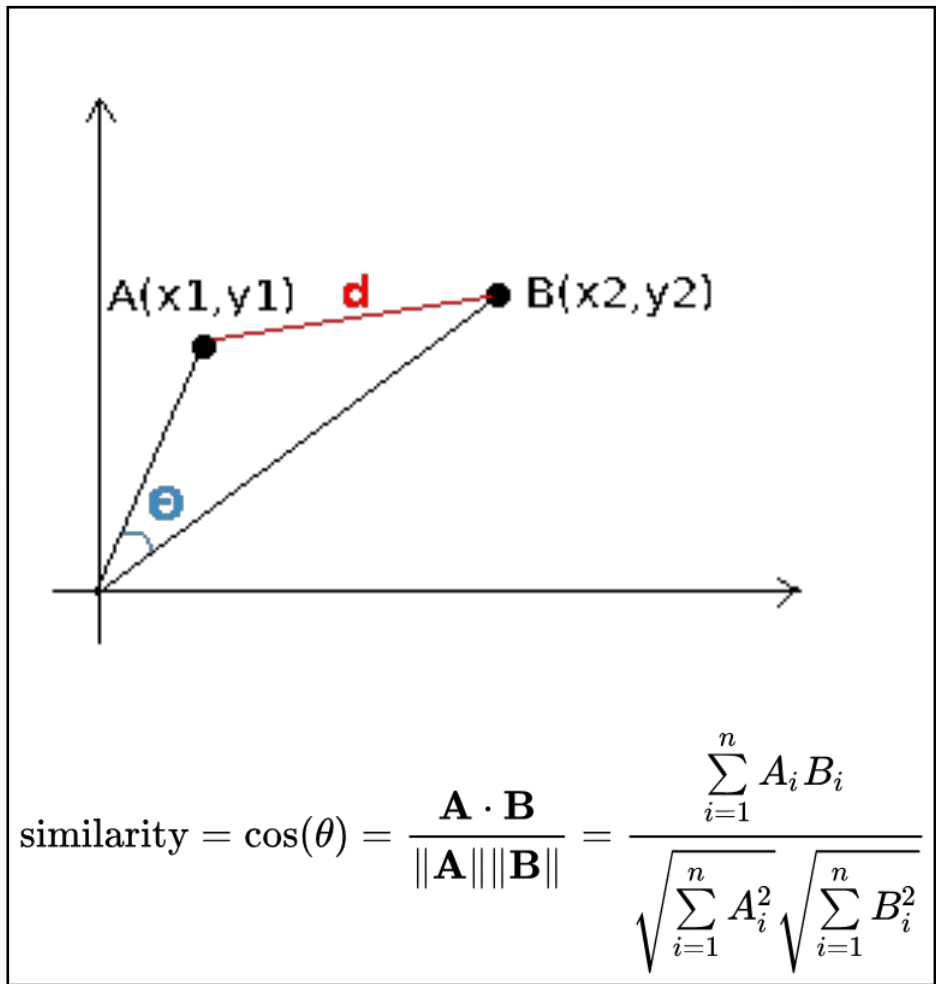


New, Vis, Semester, Course, Data, Score

A = [10, 3, 5, 20, 12, 11] |A| = 28.26



B = [5, 3, 6, 5, 3, 3] |B| = 10.64



数据属性之间相关性

- 相关性分析 (Correlation Analysis) : X 与 Y 之间是否有关联? 用户判断属性之间是否独立
- 回归分析 (Regression Analysis) : 如果有关联, 能否根据 X 的取值预测 Y 得取值? 当时数据中有缺失值时能不能通过其他属性评估缺失值的大小?

	X	Y	Z
数据元素 1	0.1	0.1	0.1
	0.2	0.2	0.2
	0.5	0.5	0.5
	0.3	0.3	0.3
	0.7	0.7	0.7
	0.4	0.4	0.4
数据元素 N	0.2	0.2	0.2
	0.5	0.5	0.5

相关性分析

- 相关性分析用于解释数据属性之间关联度的强弱
- 相关性分析揭示了一个属性的取值 是如何 跟随另外一个属性的取值 发生变化的内在规律
- 相关性分析不同于因果关系分析

相关性分析

- 方差 (variance)、标准差 (standard deviation) 及 协方差 (covariance)

- **方差及标准差：** 方差及标准差是用来衡量 单一变量 (属性) 取值变化程度的统计量

$$S_x^2 = \frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n-1}$$

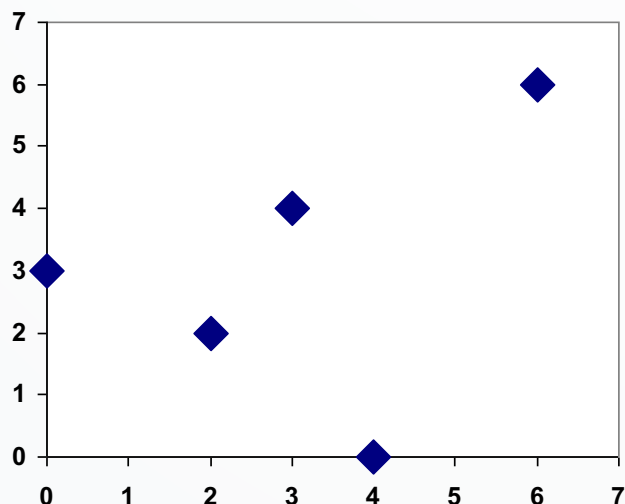
$$s = \sqrt{\frac{\sum_{i=1}^N (x_i - \bar{x})^2}{N-1}}$$

- **协方差：** 当一个变量 Y 的取值 (y_1, y_2, \dots, y_n) 随另外一个变量 X 的取值 (x_1, x_2, \dots, x_n) 变化而变化时, 协方差是用来衡量相互间变化程度的统计量

$$\text{cov}(x, y) = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{n-1}$$

- **自由度：** 上边的公式中, n 是所有取值样本的数量, $n-1$ 被称为自由度, 反映了所有能够自由改变的样本数量, 当均值固定时, n 个样本中只有 $n-1$ 个样本可以自由取值, 第 n 个样本的取值可以通过 均值及 前 $n-1$ 个样本计算出来

相关性分析 - 协方差



x	y	$x_i - \bar{x}$	$y_i - \bar{y}$	$(x_i - \bar{x})(y_i - \bar{y})$
0	3	3	0	0
2	2	-1	-1	1
3	4	0	1	0
4	0	1	-3	-3
6	6	3	3	9
$\bar{x}=3$	$\bar{y}=3$			$\Sigma=7$

$$\text{cov}(x, y) = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{n-1} = \frac{7}{4} = 1.75$$

这个数值告诉了我们什么？

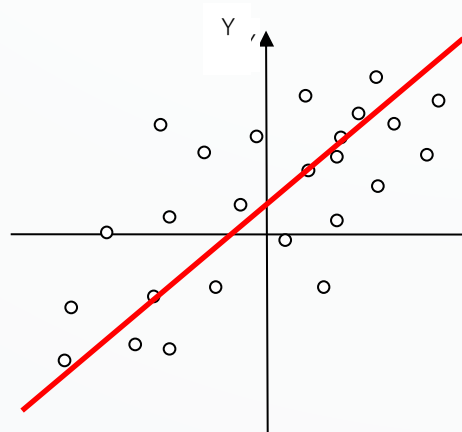
相关性分析 – 协方差

	High variance data				Low variance data		
Subject	x	y	x error * y error		x	y	x error * y error
1	101	100	2500		54	53	9
2	81	80	900		53	52	4
3	61	60	100		52	51	1
4	51	50	0		51	50	0
5	41	40	100		50	49	1
6	21	20	900		49	48	4
7	1	0	2500		48	47	9
Mean	51	50			51	50	
Sum of x error * y error :			7000		Sum of x error * y error :		28
Covariance:			1166.67		Covariance:		4.67

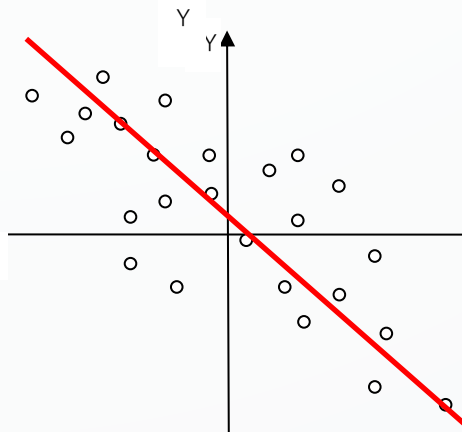
相关性分析 – 皮尔森系数

- 协方差并不能真正告诉我们有用的信息
- 皮尔森系数就是使用标准差对协方差进行正则化处理，让计算数值变得可以相互比较

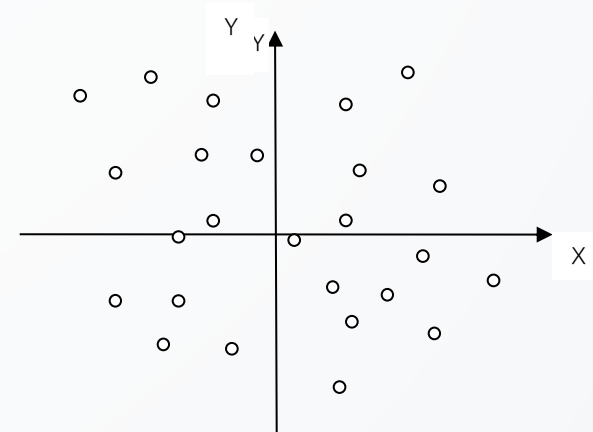
$$r_{xy} = \frac{\text{COV}(x, y)}{s_x s_y}$$



Positive correlation



Negative correlation



No correlation

相关性分析 – 协方差

$$\text{cov}(x, y) = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{n-1}$$

- When $X \uparrow$ and $Y \uparrow$: $\text{cov}(x, y) = \text{pos.}$
- When $X \downarrow$ and $Y \uparrow$: $\text{cov}(x, y) = \text{neg.}$
- When no constant relationship: $\text{cov}(x, y) = 0$

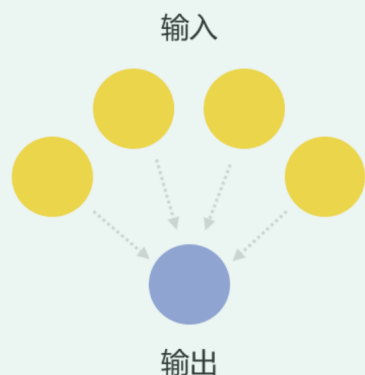
课程大纲

- 数据元素及元素的属性
- 数据元素之间的差异与关联
- 有监督学习与无监督学习
- 有监督学习：回归与分类
- 无监督学习：聚类分析

机器学习分类

监督学习

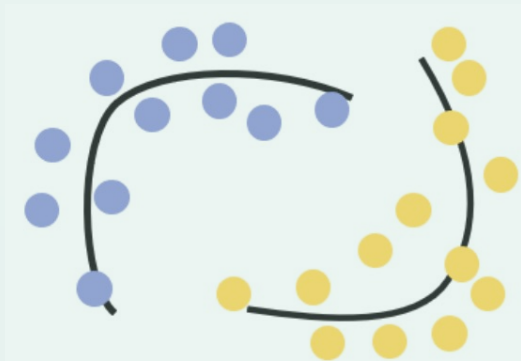
Supervised Learning



从**标记**的训练数据中学习，尝试找到输入和输出之间的映射关系。

无监督学习

Unsupervised Learning



从**未标记**的数据中学习，尝试找到数据的内在结构和模式。

强化学习

Reinforcement Learning



通过与环境的交互来学习，以实现某个**目标**或**任务**。

机器学习的分类

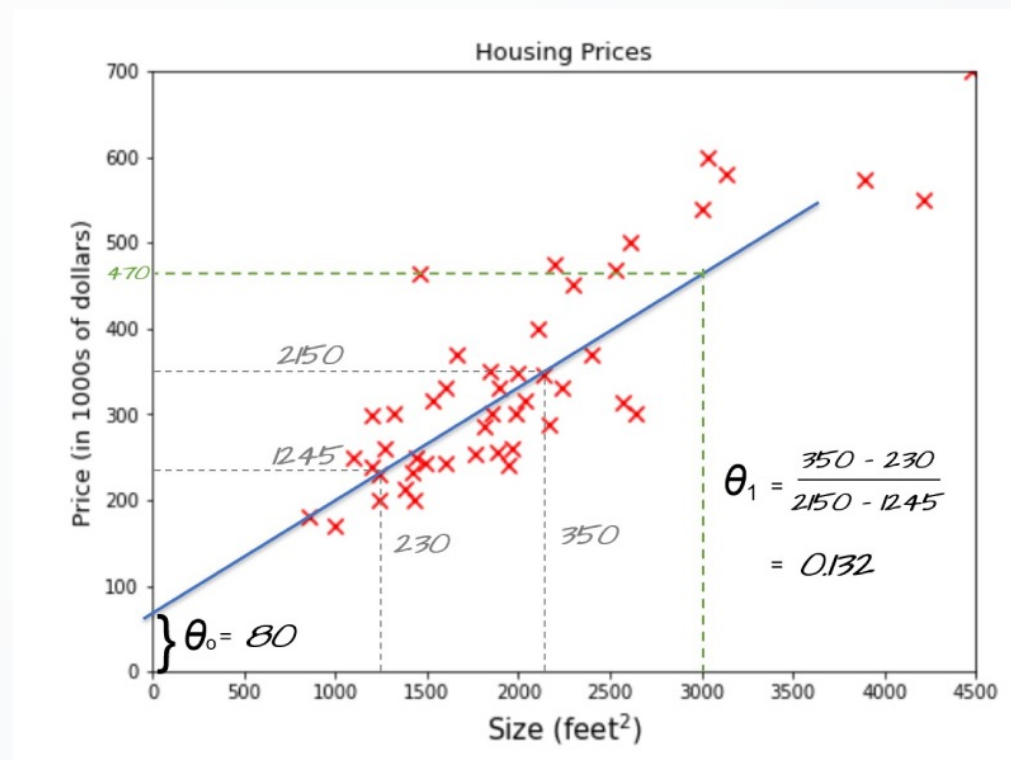
- 监督学习的主要任务
 - 回归：预测数据属性的取值
 - 分类：预测数据元素的标签
 - 生成：根据样本生成新的样本
- 无监督学习的主要任务
 - 聚类：寻找数据中的群组模式
 - 关联规则学习：检测数据频繁关联
 - 异常检测：检测离群点

课程大纲

- 数据元素及元素的属性
- 数据元素之间的差异与关联
- 有监督学习与无监督学习
- **有监督学习：回归与分类**
- 无监督学习：聚类分析

回归分析

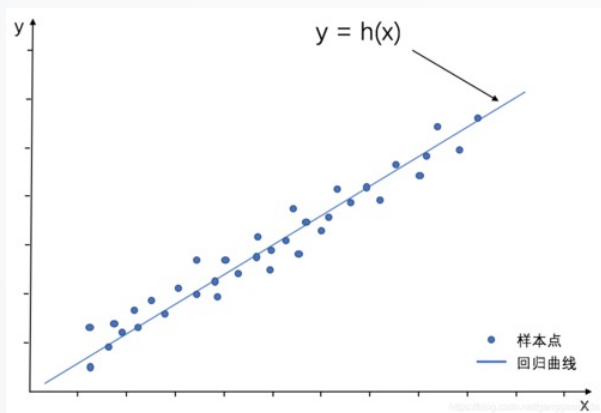
- 对于回归任务，其基本原理是利用已有的数据学习一个**映射函数**，该函数能够将输入（如房屋特征）映射到一个连续的输出值（如房价）。
- 假如你希望预测出一个房子的销售价格。给定房子的一些特征（如面积、卧室数量、地理位置等），你可以使用回归算法来预测出一个准确的价格。



回归分析

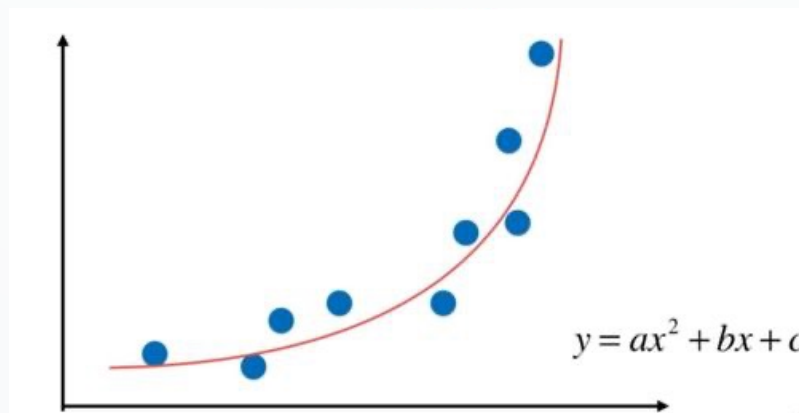
线性回归

通过拟合最佳直线来预测连续值



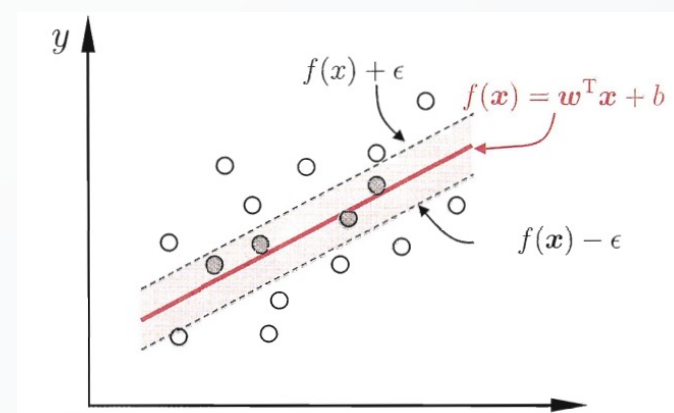
多项式回归

使用多项式方程来拟合数据

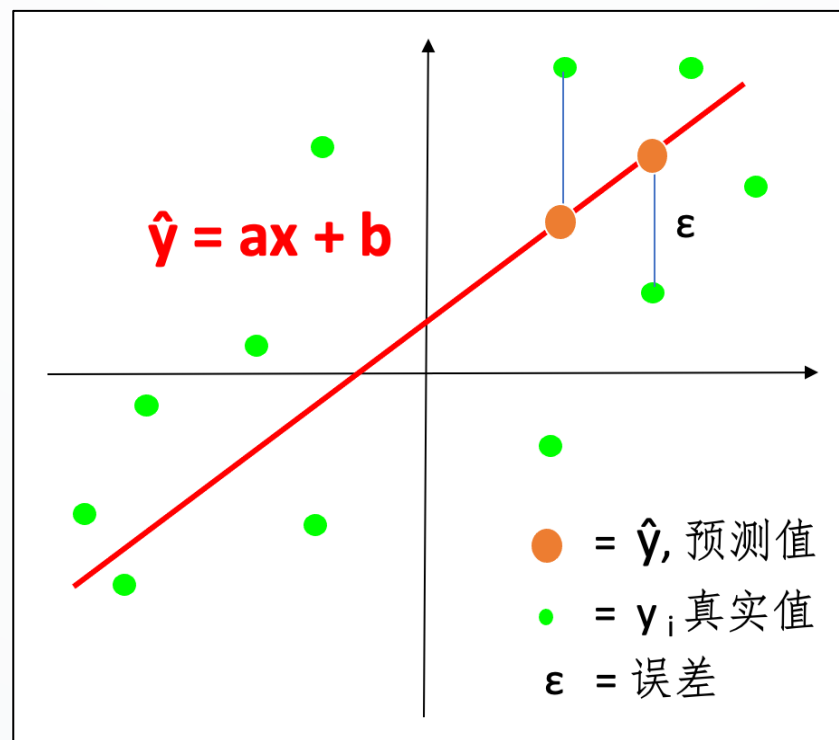


向量机回归

使用支持向量机的原理来解决回归问题



线性回归分析



在空间中找到一根直线
来近似刻画数据属性之间的联系

回归分析

- 为了找到最优的直线，我们需要尽可能减小预测的误差，需要求解一下的优化问题

模型直线: $\hat{y} = ax + b$ a = 斜率, b = 截距

误差 $(\varepsilon) = y - \hat{y}$

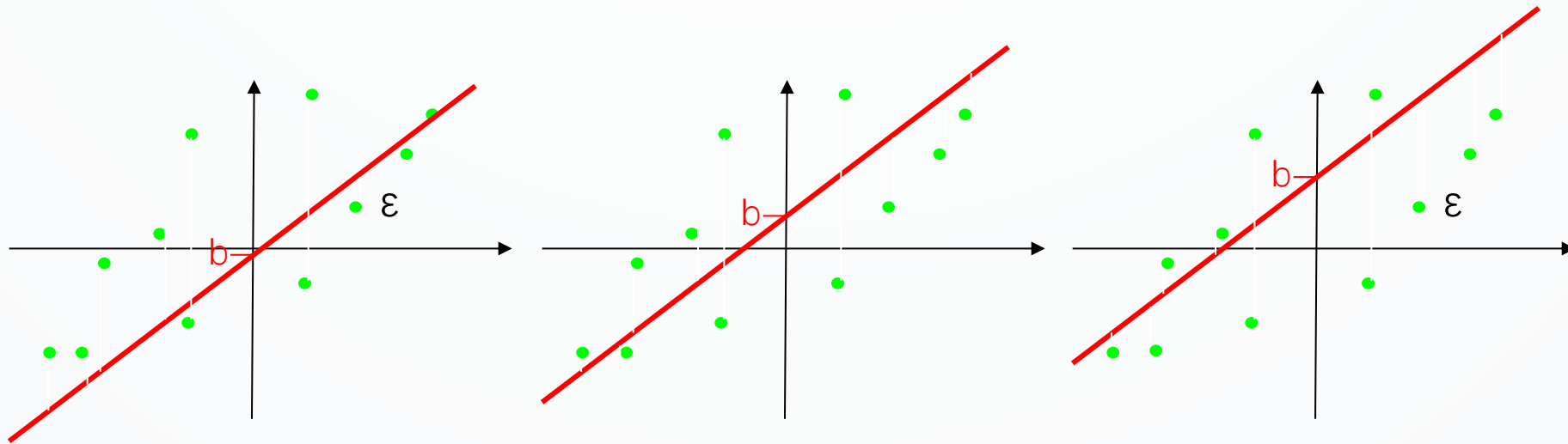
误差的平方和 = $\sum (y - \hat{y})^2$

找到使误差的平方和最小的 斜率 a 及 截距 b

$\min \sum (y - \hat{y})^2$

Finding b

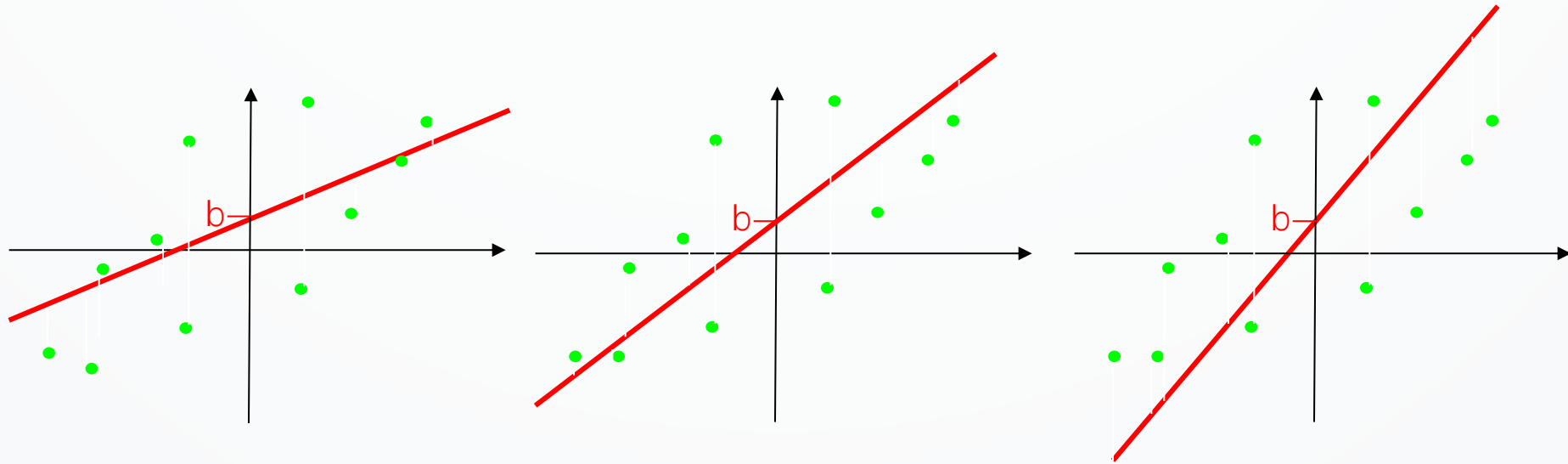
First we find the value of b that gives the min sum of squares



Trying different values of b is equivalent to shifting the line up and down the scatter plot

Finding a

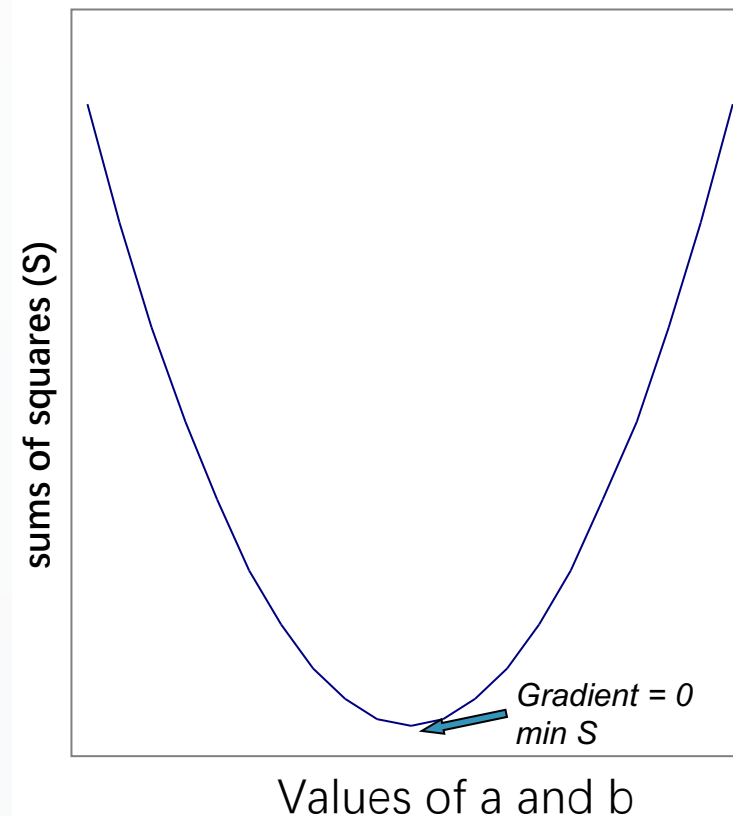
Find the value of a that gives the min sum of squares



Trying out different values of a is equivalent to changing the slope of the line, while b stays constant

最小化误差

- 需要选择能够使 $\sum (y - \hat{y})^2$ 最小的直线拟合方案
- 考虑到 $\hat{y} = ax + b$
- 所以需要最小化： $\sum (y - ax - b)^2$
- 上述二次多项式再导数为 0 时取得最小值
- 有 a、b 两个变量，需要分别求偏导数



求解回归方程

$$a = \frac{r s_y}{s_x}$$

r = 皮尔森相关系数
 s_y = y 的标准差
 s_x = x 的标准差

- X , Y 相关性越大, 线的斜率越大
- Y 的标准差越大, 斜率越大
- X 的标准差越大, 斜率越小

求解回归方程

- 考虑到模型是 $\hat{y} = ax + b$
- 这条线必须经过均值

$$\bar{y} = a\bar{x} + b \quad \longrightarrow \quad b = \bar{y} - a\bar{x}$$

- 把 a 的取值带入，可以得到 b 的计算公式

$$b = \bar{y} - \frac{r s_y}{s_x} \bar{x}$$

求解回归方程

- 把 a, b 的计算结果带入模型:

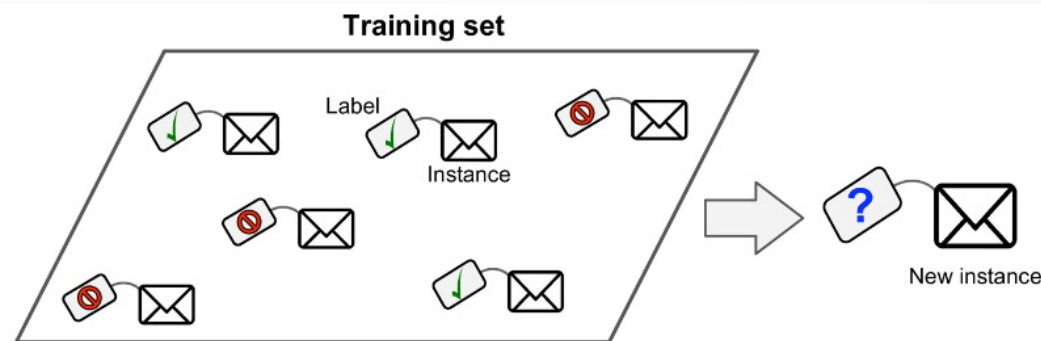
$$\hat{y} = ax + b = \frac{r s_y}{s_x} x + \bar{y} - \frac{r s_y}{s_x} \bar{x}$$

- 整理公式后的到:

$$\hat{y} = \frac{r s_y}{s_x} (x - \bar{x}) + \bar{y}$$

分类 (Classification)

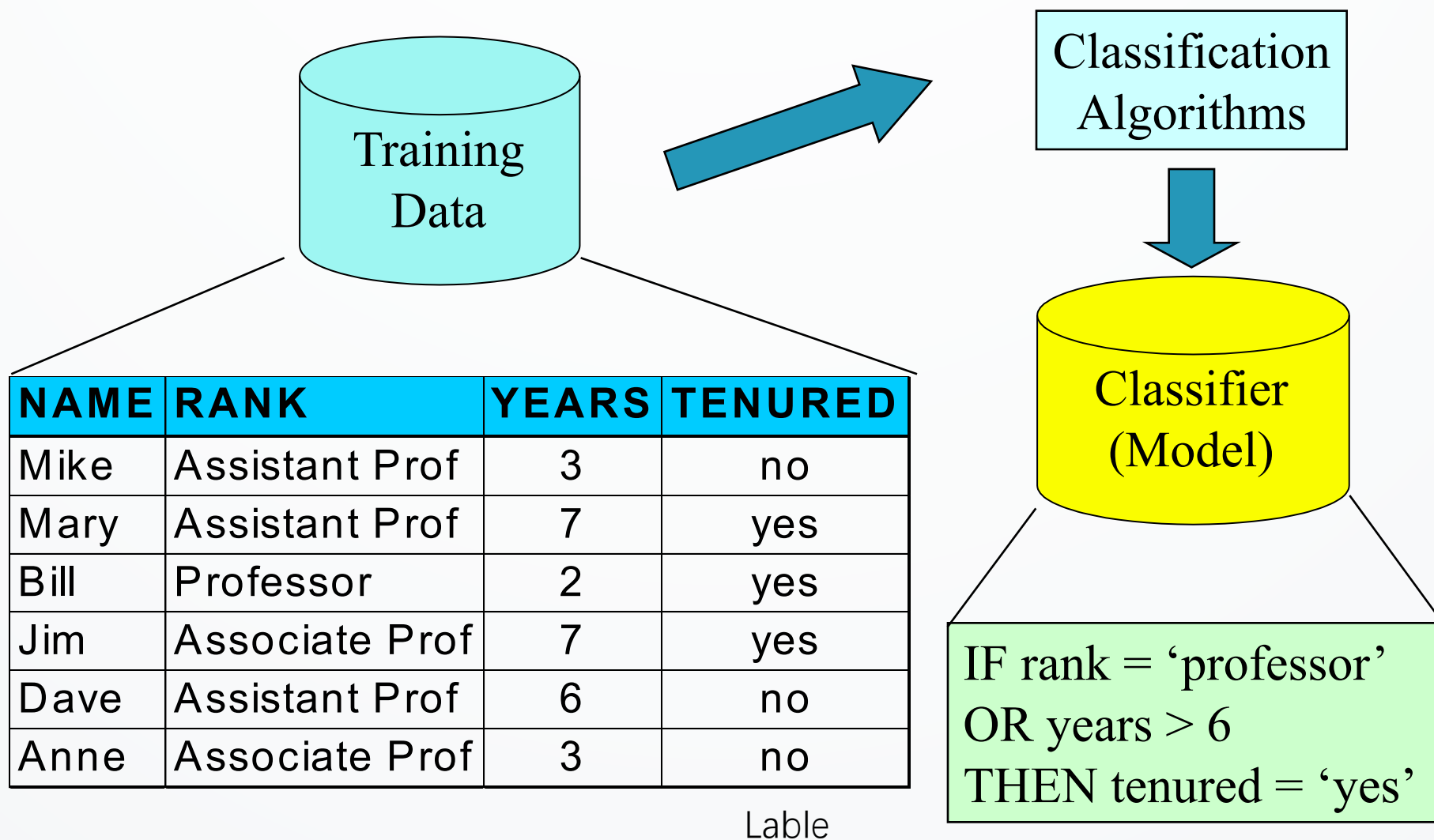
- 分类：预测数据的分类标签。基本原理是利用已有的数据学习一个映射函数/策略，将输入数据元素根据其特征映射到一个**离散**的标签。
- 回归：预测数据的取值。
- 应用：垃圾邮件识别；放贷风险判断；医疗诊断；文本主题分类，等等



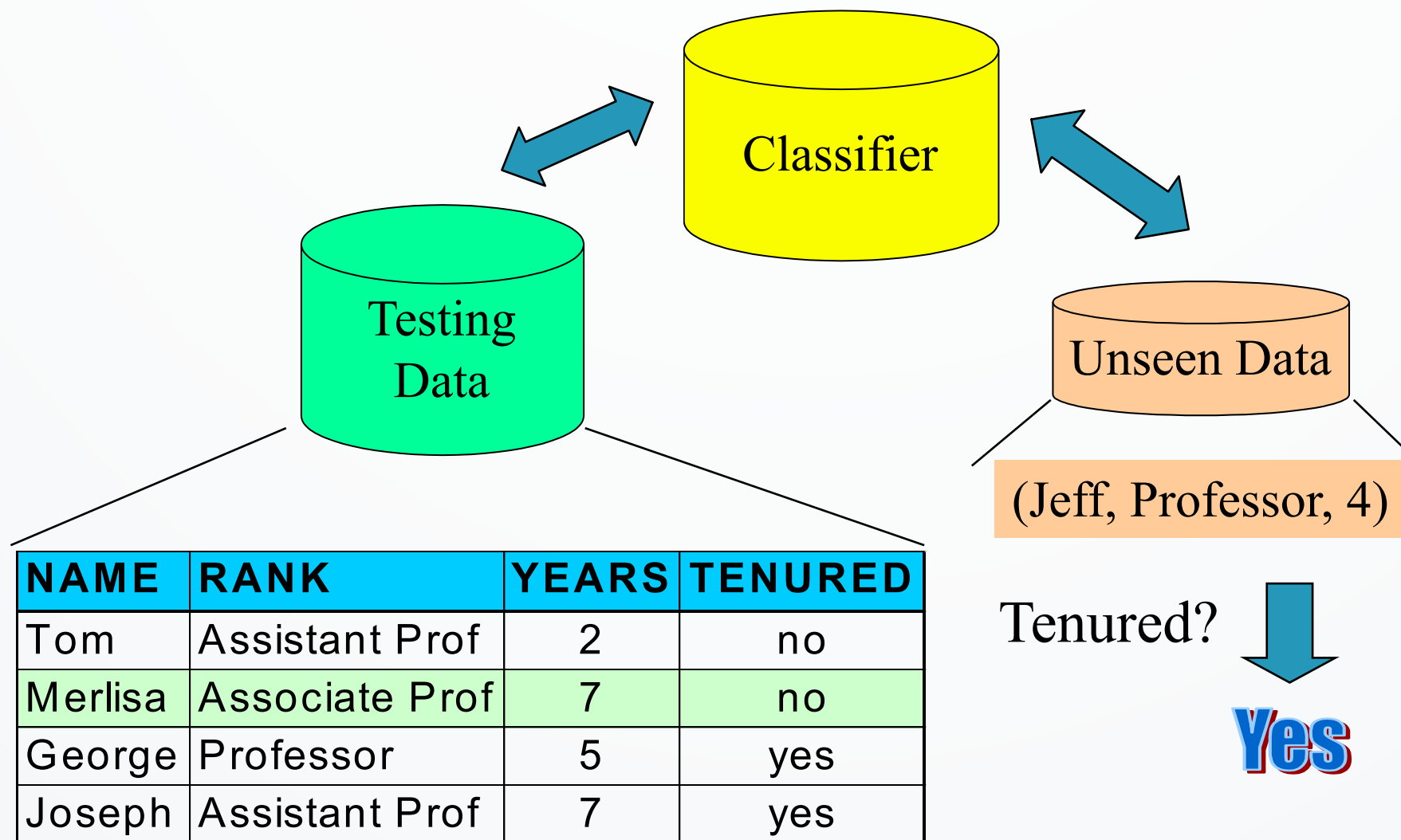
分类计算的两个步骤

- 构建分类模型：计算分类算法的关键参数
 - 用一组拥有标签的数据训练调整模型参数，使得数据中的数据元素能够被正确的划分到所属的类别中
 - 用于构建分类模型的带有标签的数据被称为 **训练数据 (Training Set)**
 - 分类算法多种多样：可以是一组策略、决策树、数据公式等
- 模型的使用：使用训练好的模型对未知类别的数据进行分类
 - 使用检测数据，对模型进行正确性检验 (test set)
 - 如果检测指标达标，则可以利用模型对未知数据进行分类
 - 用于检测模型的数据叫做 **检测数据 (Testing Set)**
 - 当检测数据用于帮助选择不同的分类模型时，这被称作 **验证数据 (Validation Set)**

第一步：模型训练



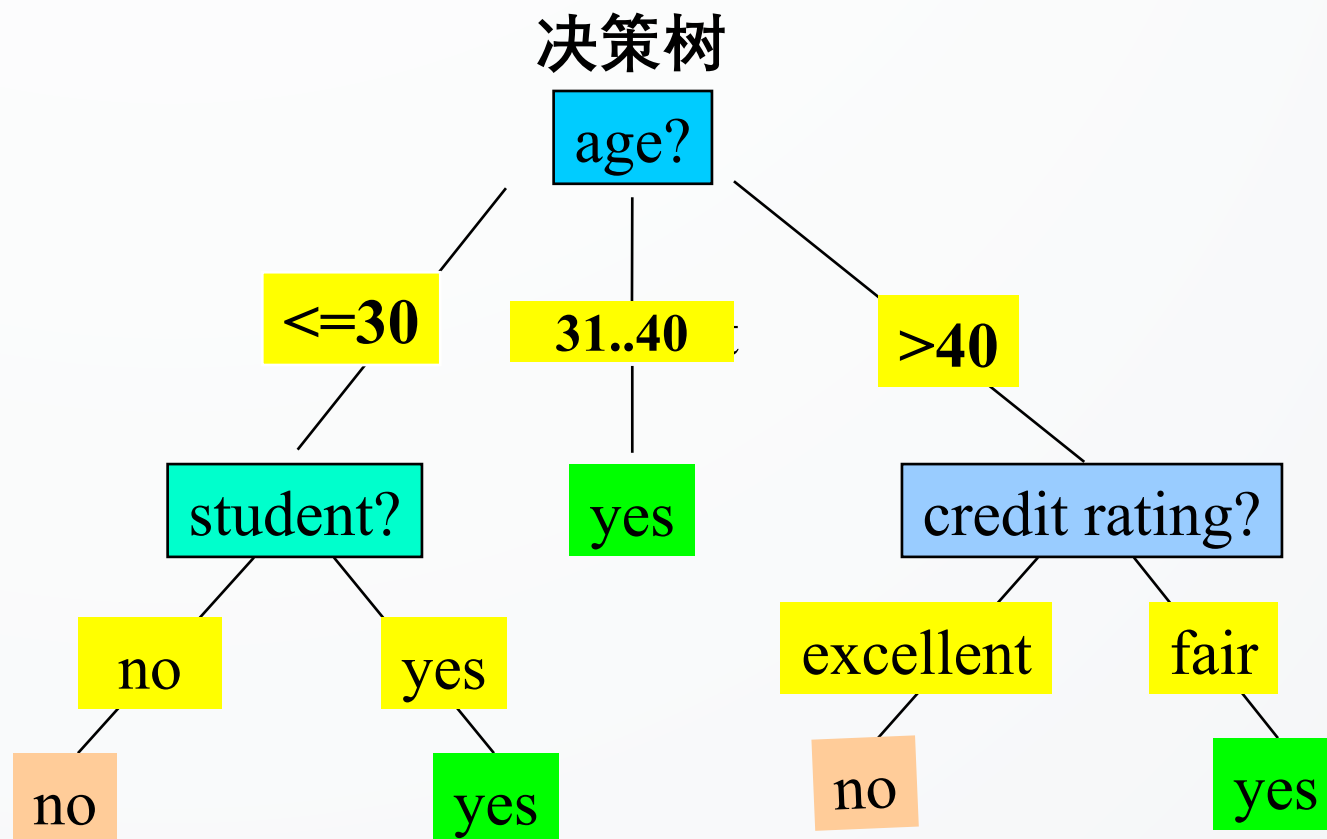
第二步：对数据进行分类



分类算法（1）：决策树

训练数据

age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
31...40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31...40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31...40	medium	no	excellent	yes
31...40	high	yes	fair	yes
>40	medium	no	excellent	no



构建决策树的方法

- 基本算法思想
 - 自顶向下的递归分治 构建
 - 初始状态下，所有训练样本都位于根节点
 - 属性必须为分类属性（如果是连续数值属性，则预先进行离散化处理），根据选择的属性递归地划分示例
 - 每一层级用于分类的属性是根据启发式策略或统计度量（例如，信息增益）来选择的
- 算法的终止条件
 - 给定节点的所有样本属于相同的类别（完美情况）
 - 没有剩余的属性可用于进一步的划分
 - 没有剩余的训练样本了

分类算法（2）：贝叶斯分类

- **统计分类器**：执行概率性预测，即预测类别成员概率
- **数学基础**：基于贝叶斯定理。
- **算法性能**：一个简单的贝叶斯分类器，例如，朴素贝叶斯分类器，常常具有与决策树和选定的神经网络分类器相媲美的性能
- **增量训练**：每个训练样本都可以逐步增加/减少假设正确性的概率 - 先验知识可以与观察数据相结合
- **标准算法**：即使在计算上难以处理时，贝叶斯方法可以提供一种标准的最佳决策制定，用以衡量其他方法。

贝叶斯分类器的原理

- D 是训练数据集, X 是 D 中的一个数据元素, 用一系列特征进行描述 $\mathbf{X} = [x_1, x_2, \dots, x_n]$
- 假设已知数据可以被划分为 m 类, 每一类的标签是 C_1, C_2, \dots, C_m
- 计算 $P(C_i | \mathbf{X})$, 即数据 X 在分类 C_i 中的概率, 概率最大的 就是 X 所对应的分类

$$P(C_i | \mathbf{X}) = \frac{P(\mathbf{X} | C_i)P(C_i)}{P(\mathbf{X})}$$



$$P(C_i | \mathbf{X}) = P(\mathbf{X} | C_i)P(C_i)$$

只用计算使上式子最大的 i 的取值就可以了

朴素贝叶斯分类器 (Naïve Bayes Classifier)

- 考虑到 X 包含多个属性, 上述公式中联合条件概率 $P(x_1, \dots, x_n | C_i)$ 在实际问题中往往难以计算, 因此可以通过 假设 X 的属性 相互独立, 将计算化简为:

$$P(\mathbf{X} | C_i) = \prod_{k=1}^n P(x_k | C_i) = P(x_1 | C_i) \times P(x_2 | C_i) \times \dots \times P(x_n | C_i)$$

- 极大化简了计算复杂度, $P(x_k | C_i)$ 可以通过统计训练集中 x_k 的条件分布直接获得
 - 如果 x_k 是分类型属性 (Categorical), 择可以直接通过计数的到
 - 如果 x_k 是连续数值型属性 (Numerical), 择可以假设 x_k 符合某种分布 (例如正太分布), 并通过训练数据计算该分布的参数 (均值、标准差) 从而得到相应的概率

$$g(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

$$P(\mathbf{X} | C_i) = g(x_k, \mu_{C_i}, \sigma_{C_i})$$

练习题

Class:

C1:buys_computer = 'yes'

C2:buys_computer = 'no'

Data to be classified:

X = (age <=30,

Income = medium,

Student = yes

Credit_rating = Fair)

Will the person X by a
computer ?

age	income	student	credit_rating	com
<=30	high	no	fair	no
<=30	high	no	excellent	no
31...40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31...40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31...40	medium	no	excellent	yes
31...40	high	yes	fair	yes
>40	medium	no	excellent	no

答案

- $P(C_i)$: $P(\text{buys_computer} = \text{"yes"}) = 9/14 = 0.643$
 $P(\text{buys_computer} = \text{"no"}) = 5/14 = 0.357$
- Compute $P(X|C_i)$ for each class and each attributes

$P(\text{age} = \text{"<=30"} \mid \text{buys_computer} = \text{"yes"}) = 2/9 = 0.222$
 $P(\text{age} = \text{"<= 30"} \mid \text{buys_computer} = \text{"no"}) = 3/5 = 0.6$
 $P(\text{income} = \text{"medium"} \mid \text{buys_computer} = \text{"yes"}) = 4/9 = 0.444$
 $P(\text{income} = \text{"medium"} \mid \text{buys_computer} = \text{"no"}) = 2/5 = 0.4$
 $P(\text{student} = \text{"yes"} \mid \text{buys_computer} = \text{"yes"}) = 6/9 = 0.667$
 $P(\text{student} = \text{"yes"} \mid \text{buys_computer} = \text{"no"}) = 1/5 = 0.2$
 $P(\text{credit_rating} = \text{"fair"} \mid \text{buys_computer} = \text{"yes"}) = 6/9 = 0.667$
 $P(\text{credit_rating} = \text{"fair"} \mid \text{buys_computer} = \text{"no"}) = 2/5 = 0.4$

-
- **$X = (\text{age} \leq 30, \text{income} = \text{medium}, \text{student} = \text{yes}, \text{credit_rating} = \text{fair})$**

$P(X|C_i)$:

$P(X \mid \text{buys_computer} = \text{"yes"}) = 0.222 \times 0.444 \times 0.667 \times 0.667 = 0.044$

$P(X \mid \text{buys_computer} = \text{"no"}) = 0.6 \times 0.4 \times 0.2 \times 0.4 = 0.019$

$P(X|C_i) \cdot P(C_i)$:

$P(X \mid \text{buys_computer} = \text{"yes"}) \cdot P(\text{buys_computer} = \text{"yes"}) = 0.028$

$P(X \mid \text{buys_computer} = \text{"no"}) \cdot P(\text{buys_computer} = \text{"no"}) = 0.007$

X belongs to class ("buys_computer = yes")

$$P(C_i|\mathbf{X})=P(\mathbf{X}|C_i)P(C_i)$$

$$P(\mathbf{X}|C_i)=\prod_{k=1}^n P(x_k|C_i)=P(x_1|C_i)\times P(x_2|C_i)\times...\times P(x_n|C_i)$$

age	income	student	credit_rating	com
<=30	high	no	fair	no
<=30	high	no	excellent	no
31...40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31...40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31...40	medium	no	excellent	yes
31...40	high	yes	fair	yes
>40	medium	no	excellent	no

朴素贝叶斯分类的优点与缺点

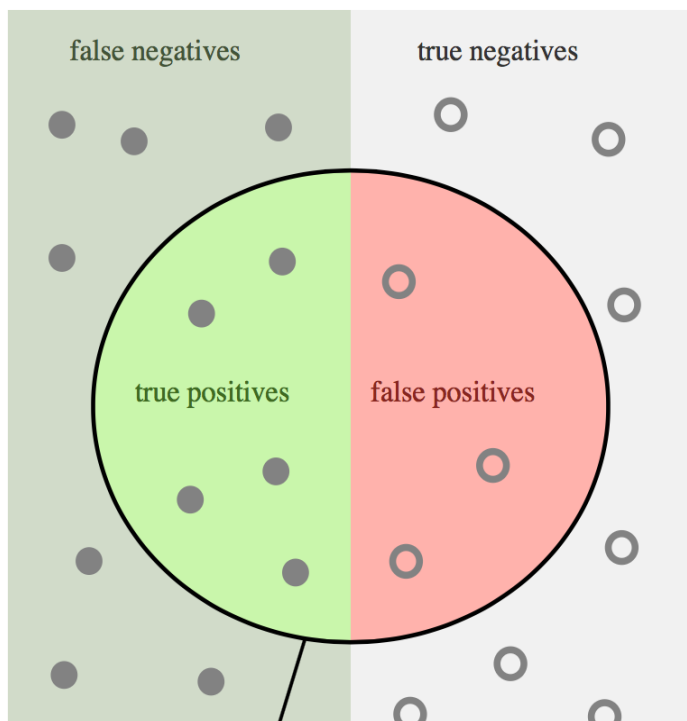
- 优点：
 - 易于实现
 - 绝大多数情况下性能优良
- 缺点：
 - 真实数据的属性往往无法完全满足相互独立（怎么办？）

分类模型的评估方法

- 评估数据集：在评估准确性时，使用带有类别标签的数据进行测试
- 估算分类器准确性的方法：
 1. **交叉验证 (Cross-validation)**：将数据分成多个折叠 (folds)，然后依次将每个折叠用作验证集，其余用作训练集，以多次评估分类器性能并取平均值。
 2. **随机采样 (Holdout by random subsampling)**：随机将数据集划分为训练集和测试集，通常按照一定比例划分，然后使用测试集评估分类器的性能。
- 比较分类器：
 - 精确度 (Precision)，召回率 (Recall)
 - Cost-benefit analysis and ROC Curves

混淆矩阵 (Confusion Matrix)

The data actually
in the class



Prediction Results

(the data that are predicted to
be inside a class)

Confusion Matrix

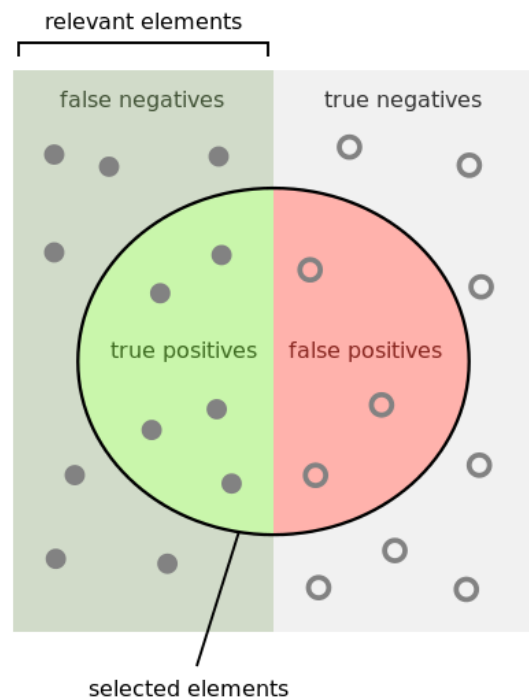
Prediction Results

Label		C	$\neg C$
	C	True Positives (TP)	False Negatives (FN)
	$\neg C$	False Positives (FP)	True Negatives (TN)

Example

Actual class\Predicted class	buy_computer = yes	buy_computer = no	Total
buy_computer = yes	6954	46	7000
buy_computer = no	412	2588	3000
Total	7366	2634	10000

评估指标：精确度、召回率



How many selected items are relevant?

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$

How many relevant items are selected?

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

- 精确度：在分类器识别的所有结果当中，有多少数据元素实际上应该属于 C 的类别（百分比）？

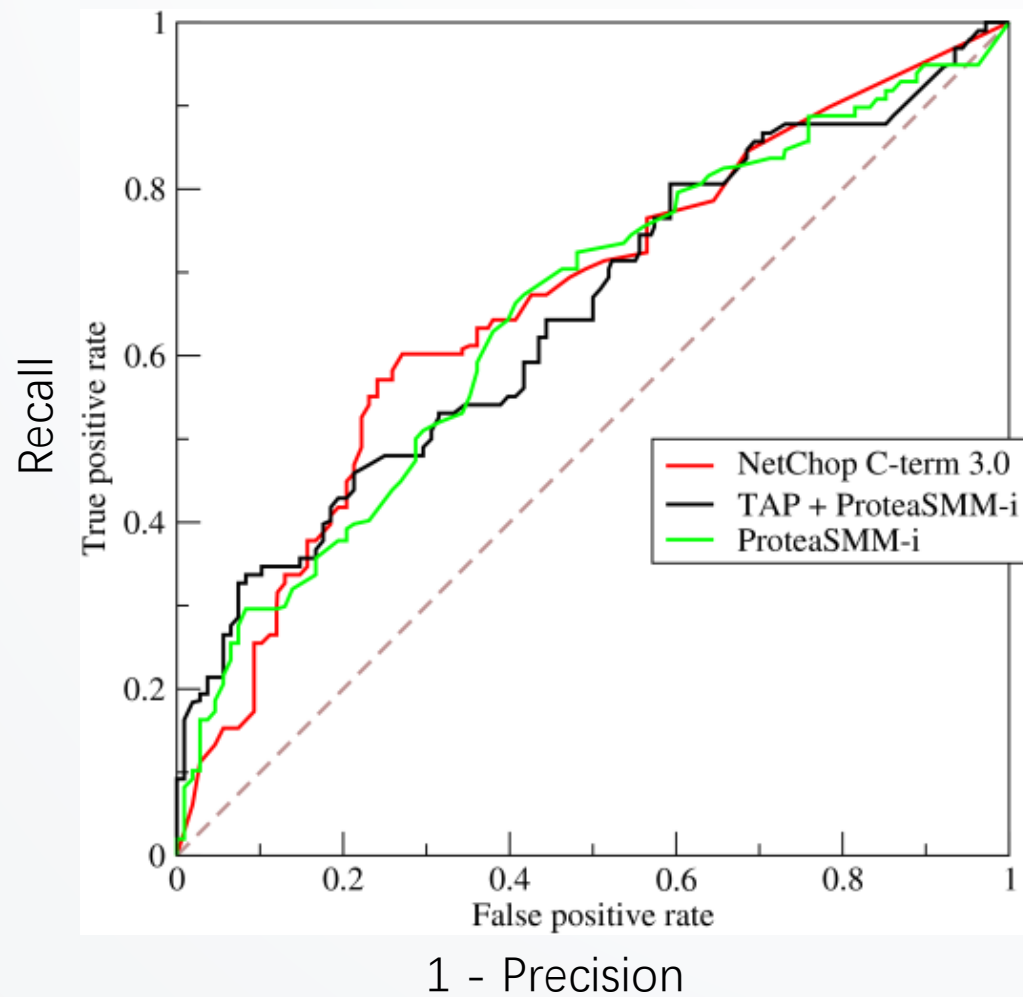
$$\text{precision} = \frac{TP}{TP + FP}$$

- 召回率：在 C 所包含的所有数据中，分类器成果识别了多少（百分比）？

$$\text{recall} = \frac{TP}{TP + FN}$$

- 精确度和召回率之间存在反向关系。精确度高，召回率往往比较低，召回率高，精度往往比较低。

ROC 曲线



- ROC 曲线展现了 精确度与召回率之间的关系变化关系
- AUC Value: ROC 曲线下覆盖的面积值
- 用来对比 多个分类模型, AUC 越大越好, 取值范围往往在 $[0.5, 1]$ 之间
- 问题: ROC 曲线是怎样绘制的?

课程大纲

- 数据元素及元素的属性
- 数据元素之间的差异与关联
- 有监督学习与无监督学习
- 有监督学习：回归与分类
- **无监督学习：聚类分析**

聚类分析

- 一组相互 关联/相似 的数据构成的集合称为簇
- 通过计算发现簇的过程称为聚类分析
- 聚类分析被应用于很多方面：在商业上，被用来进行用户画像，即用于发现不同的客户群，并且通过购买模式刻画不同的客户群的特征；在生物上，聚类分析被用来进基因分类及物种划分，获取对种群固有结构的认识；在互联网上，聚类分析被用于文档归类，主题提取与分析等

聚类分析

- 聚类分析是一种无监督学习算法，即对数据分类不依赖于任何先验知识及数据标签，仅仅取决于数据的内在属性
- 好的聚类分析算法应该能够分析得到高质量的聚类结果
 - 属于同一个类中的数据元素之间应该相互相似
 - 属于不同类中的数据元素之间应该有显著的差异
- 聚类分析的好坏与算法本身息息相关 也与算法所采用的数据度量方法相关

常见的聚类分析策略

- 基于数据划分的聚类分析
- 层次化的聚类分析
- 基于密度的聚类分析

基于数据划分的聚类分析

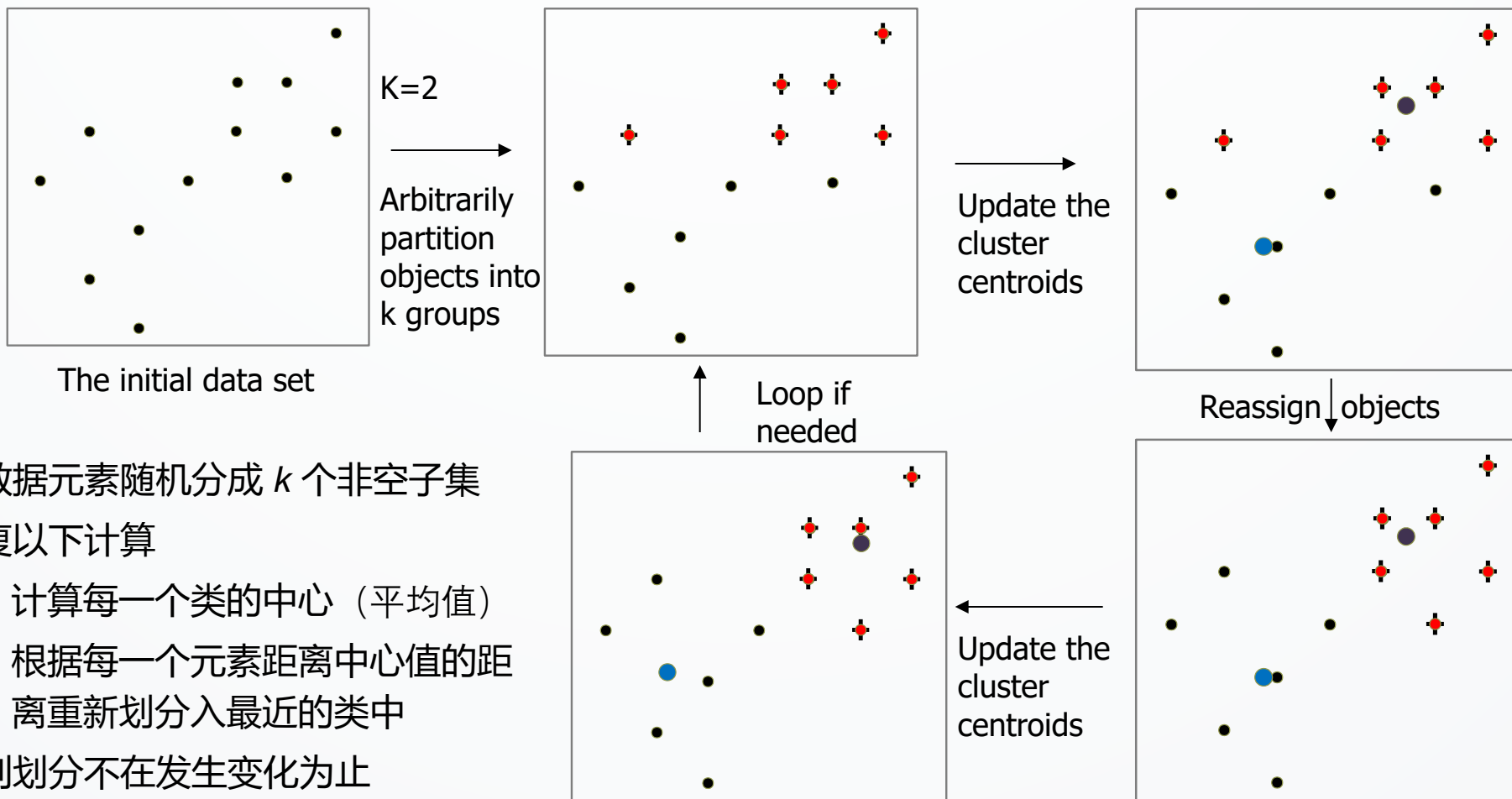
- 基于数据划分的聚类分析
- 讲一个数据集 D 中的数据元素根据其相似度划分为 K 份，并使得一下目标函数最小

$$E = \sum_{i=1}^k \sum_{p \in C_i} (p - c_i)^2$$

类中的
数据元素 类的
中心点

- 求解最优结果需要穷举所有一切可能的划分方法
- 近似解法同样能够很好的解决问题

K-Means 聚类算法

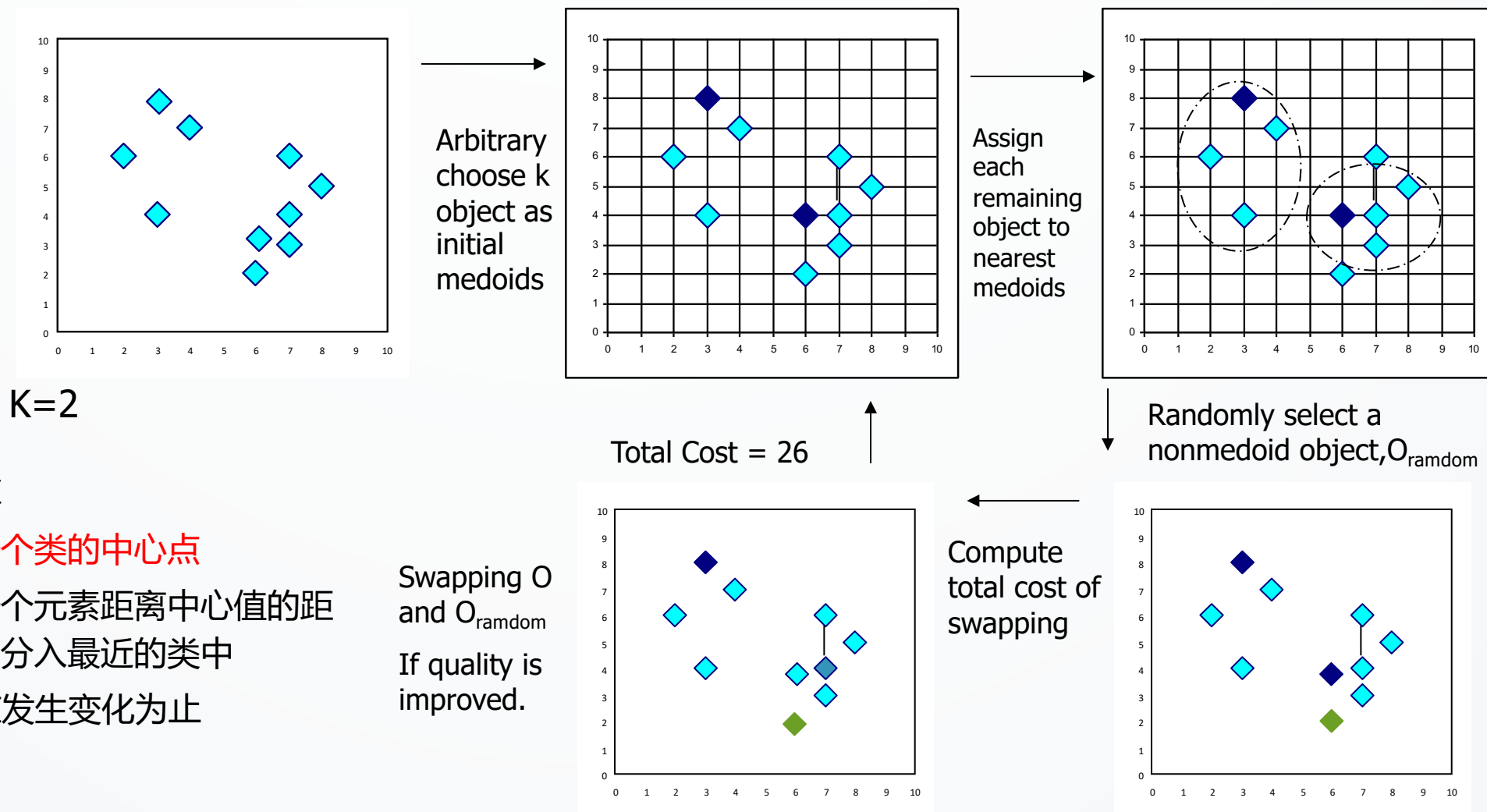


- 将数据元素随机分成 k 个非空子集
- 重复以下计算
 - 计算每一个类的中心（平均值）
 - 根据每一个元素距离中心值的距离重新划分入最近的类中
- 直到划分不在发生变化为止

K-Means 的优缺点

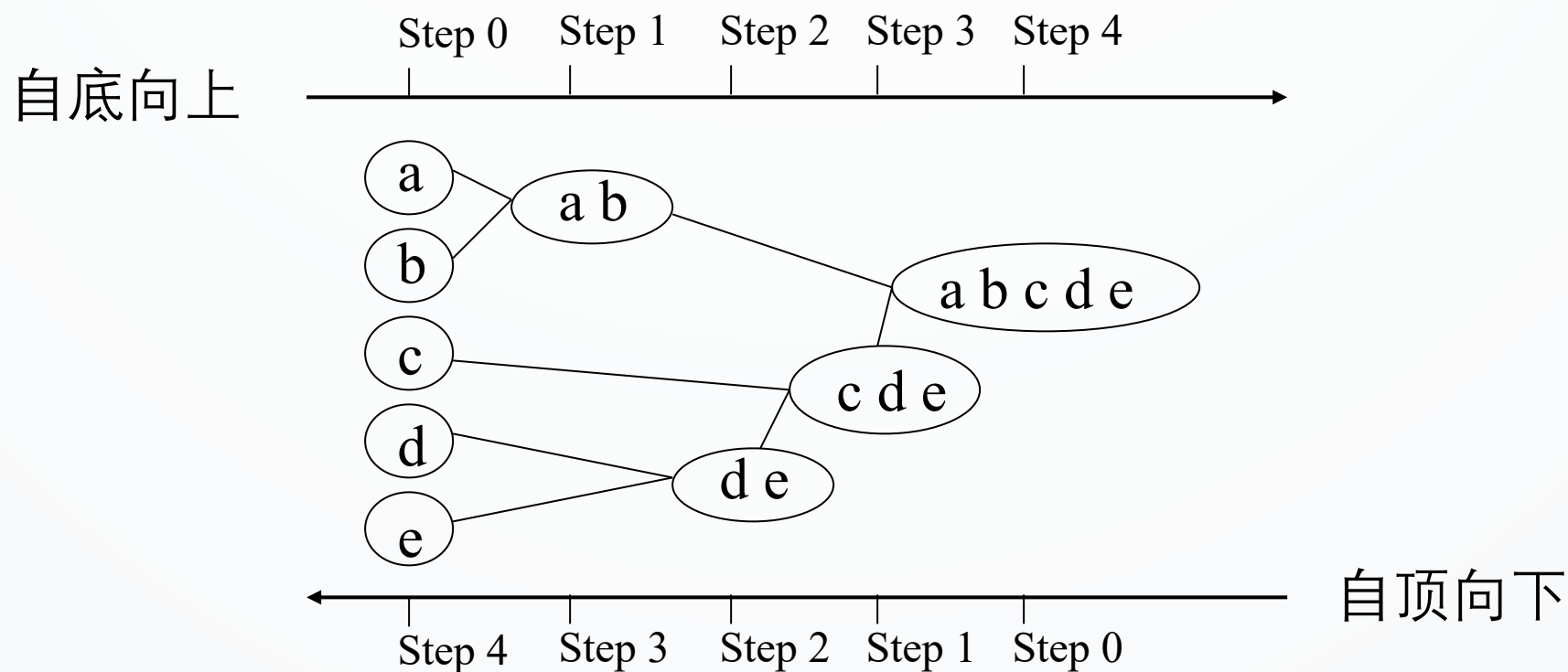
- 非常高效 $O(tkn)$, t 是迭代次数、 k 是簇的数量, n 是数据元素的数量, 正常情况下 $k, t \ll n$.
- 近似算法, 最终结果往往不是最优解
- 需要输入 K
- 因为需要计算 means, 所以需要数据用于聚类的属性必须是数值型属性
- 对离群点 (数据噪声、异常) 比较敏感
- 只适用于凸包的情况

K-Centroid 聚类算法



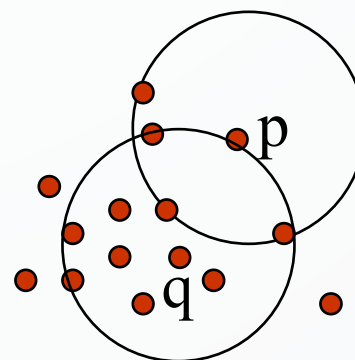
- 重复以下计算
 - 查找每一个类的中心点
 - 根据每一个元素距离中心值的距离重新划分入最近的类中
- 直到划分不在发生变化为止

层次化的聚类分析



基于数据密度的聚类分析方法

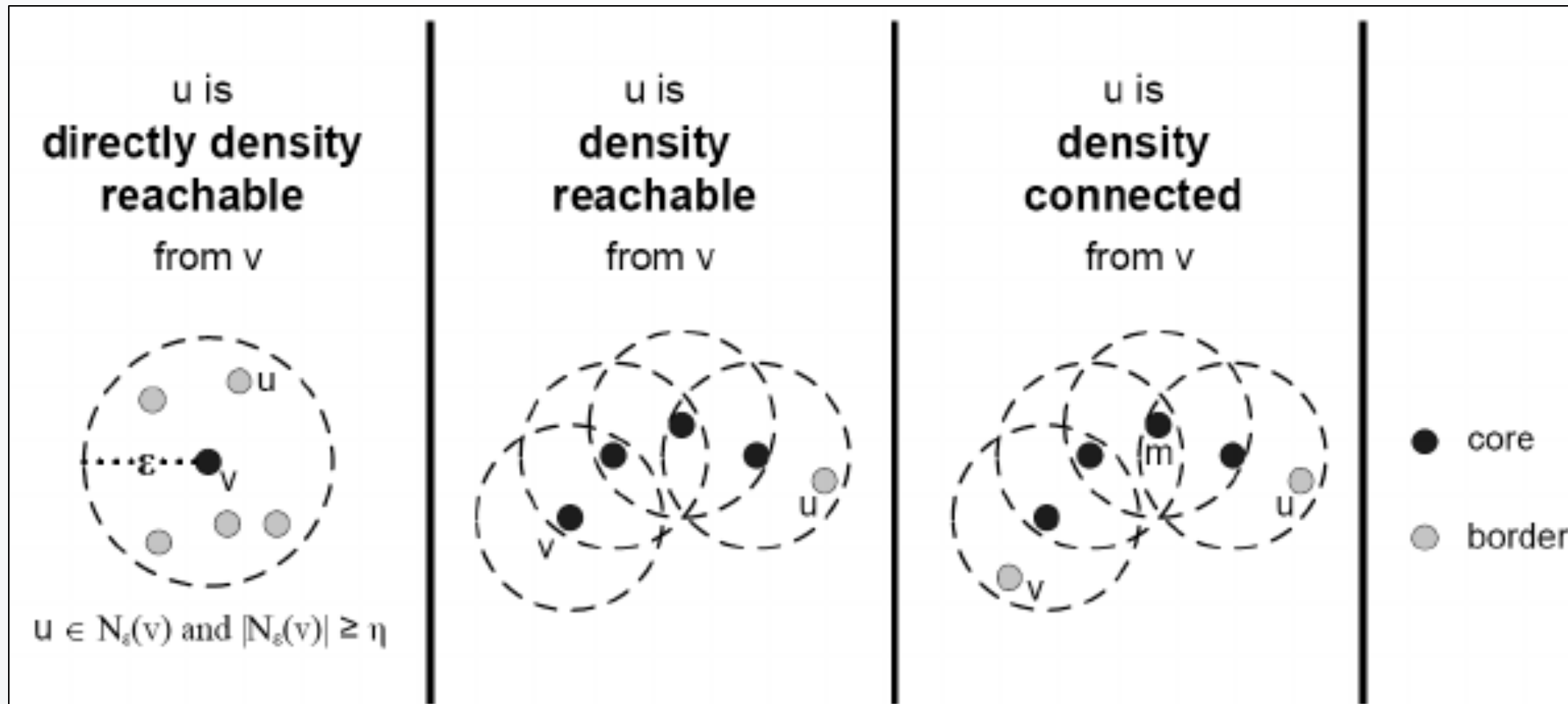
- 两个重要参数：
 - 最大检测半径 Eps
 - 最小相邻数据元素数 $MinPts$
- 点 p 的邻居 $N(p)$ 可以定义为以 p 为中心, 以 Eps 为半径的圆所涵盖的所有数据元素



$MinPts = 5$

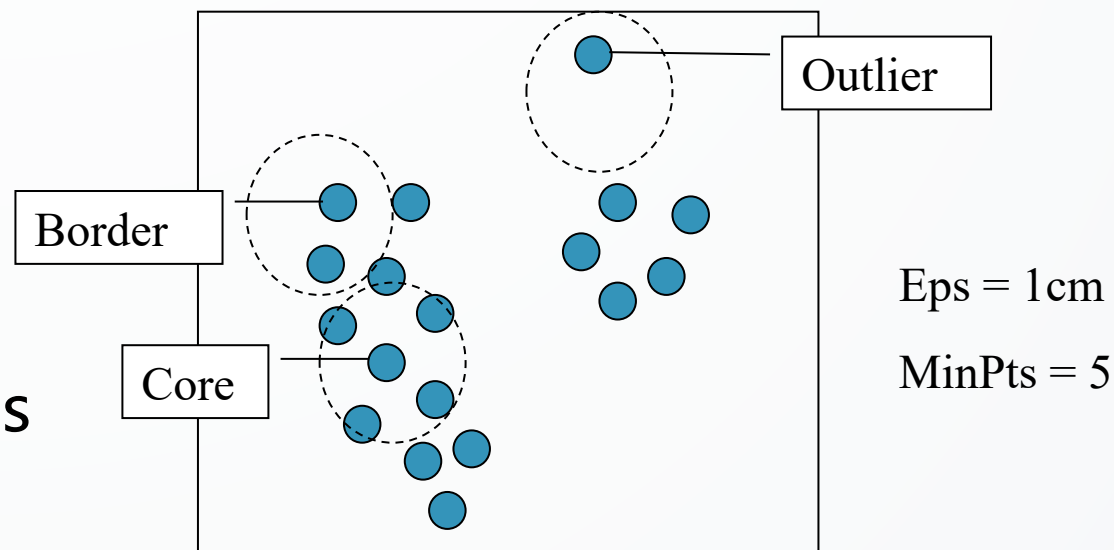
$Eps = 1\text{ cm}$

基于数据密度的聚类分析方法



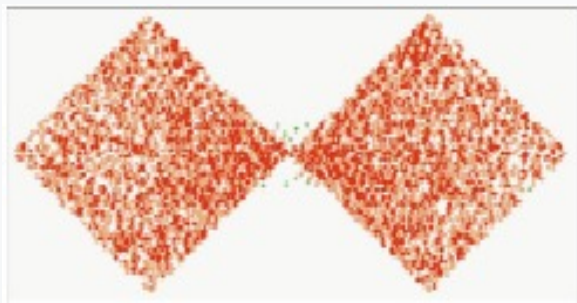
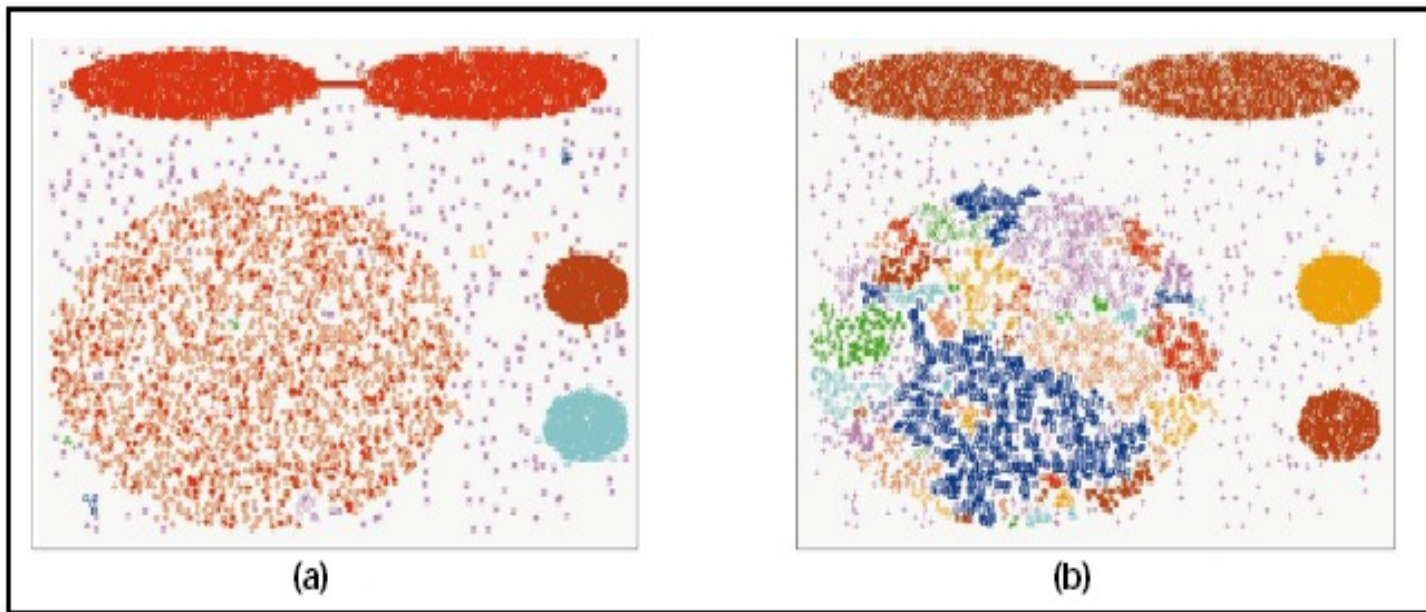
基于数据密度的聚类分析方法

- 根据数据分布进行扫描
- 可以检测到任何形状的簇
- 不会收到数据噪声的影响
- 不用输入类的数目但是算法结果受到Eps以及 MinPts 的影响

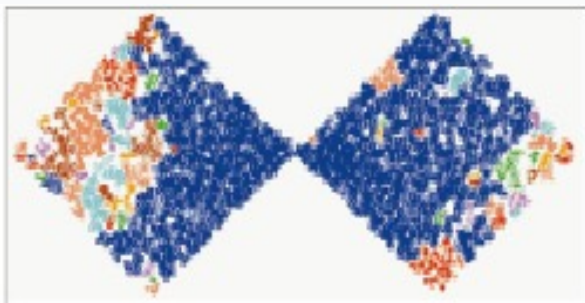


基于数据密度的聚类分析方法

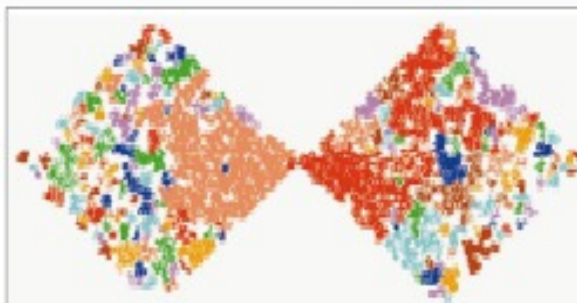
Figure 8. DBScan results for DS1 with MinPts at 4 and Eps at (a) 0.5 and (b) 0.4.



(a)



(b)



(c)

课程总结

- 本节课
 - 数据元素及元素的属性
 - 数据元素之间的差异与关联
 - 有监督学习与无监督学习
 - 有监督学习：回归与分类
 - 无监督学习：聚类分析
- 下节课
 - 可视化的基本设计准则