机器学习
Machine Learning - Overview

谢志鹏
复旦大学计算机学院
xiezp@fudan.edu.cn

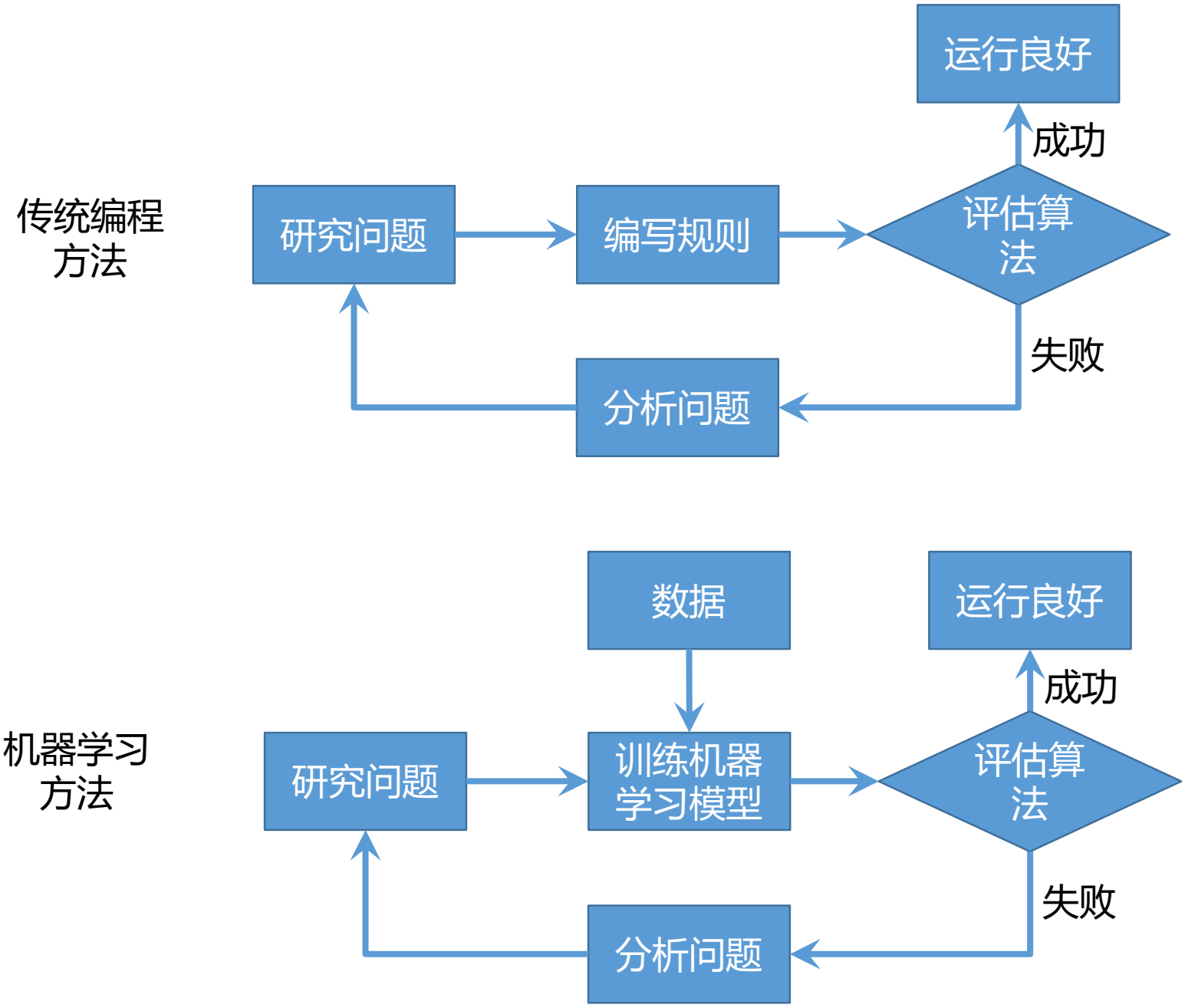
什么是机器学习?
What is machine learning?

- Arthur Samuel (1959):
  - "Field of study that gives computers the ability to learn without being explicitly programmed"
- Herbert Simon (1980):
  - “Learning is any process by which a system improves performance from experience”
- Tom Mitchell (1997):
  - A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E

为何需要使用机器学习？
Why?

- 普通程序难以求解的问题：
  - 不知道要写什么样的程序
  - 程序的复杂程度可能是令人恐怖的
  - 情况在不停地发生改变

- 如何处理？


传统编程
方法
研究问题
编写规则
评估算法
运行良好
成功
失败
分析问题
机器学习
方法
数据
研究问题
训练机器学习模型
评估算法
运行良好
成功
失败
分析问题

与人工智能的关系

机器学习可以被看作是**人工智能**的一个分支

将经验转化成技能或者**从复杂感知数据中检测出有意义的模式**都是人类智能或动物智能的基石

机器学习**并非**试图去构建对智能行为的自动模仿(automated imitation)，而是使用计算机的力量和特殊能力来**补充**人类智能

大数据时代
Big Data Era

- 大数据时代
  - 互联网：~10^{12} 网页@2008
  - 生物信息：人的基因组3.8*10^{9}个基对
  - 商业：沃尔玛100万项交易/秒；2.5PB数据库（2010）
- 如何进行自动数据分析？

机器学习理论基础的学习必要性

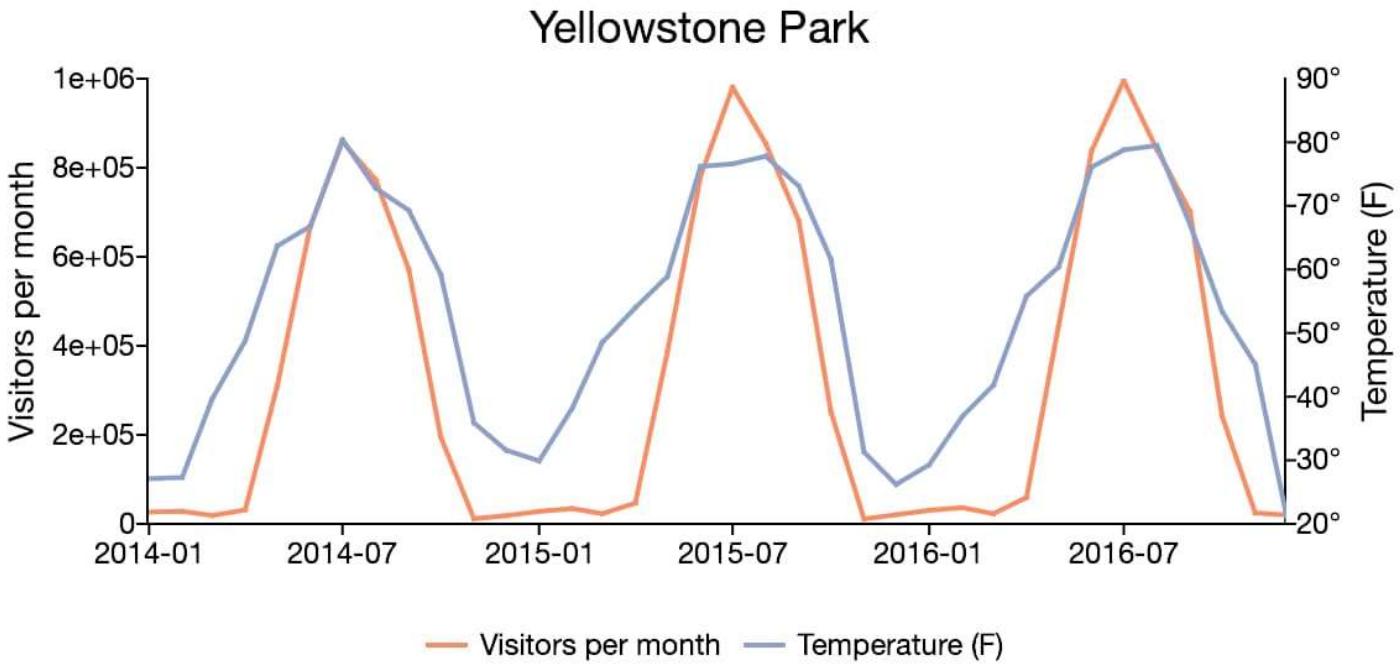
- 小的方面：调参的需要
  - 现有模型是如何工作的？超参的含义是什么？调整这些超参会产生哪些影响？等等

- 大的方面：解决实际问题（新问题）的需要
  - 新需求不断涌现，如何针对新问题建模并设计新目标函数
  - 具体的业务场景各不相同

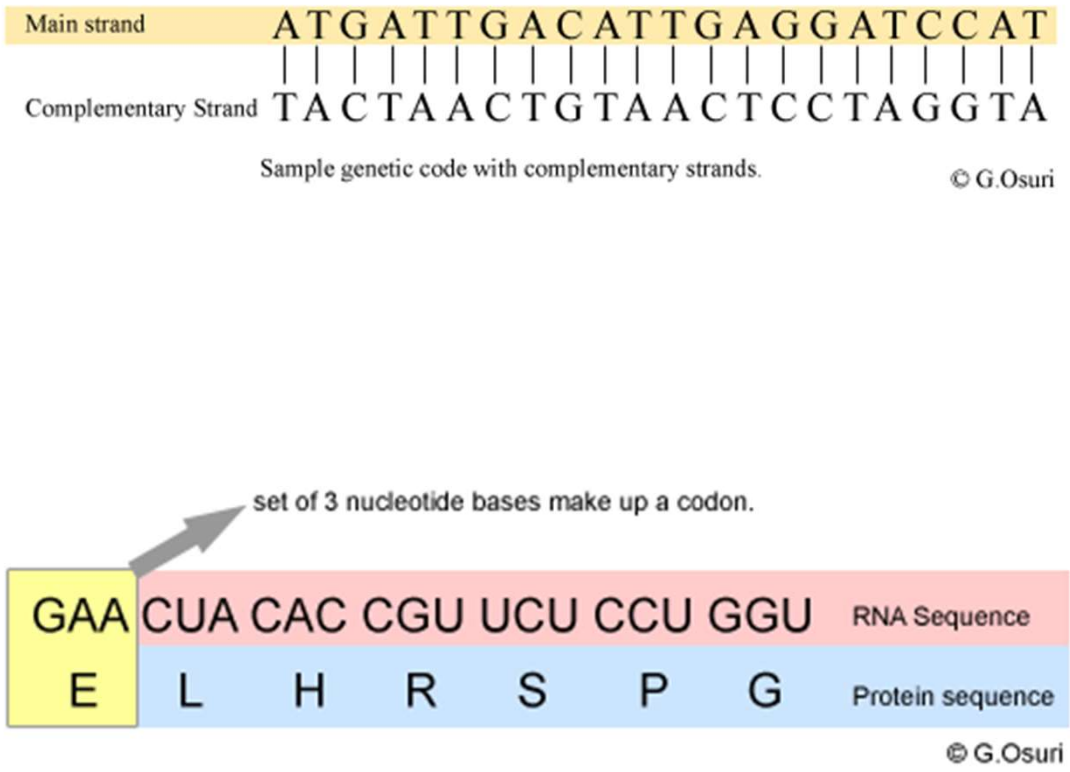
数据
Data

- 向量
- 序列数据
- 图和网络
- 文本
- 图像
- 视频
- 音频

时间序列

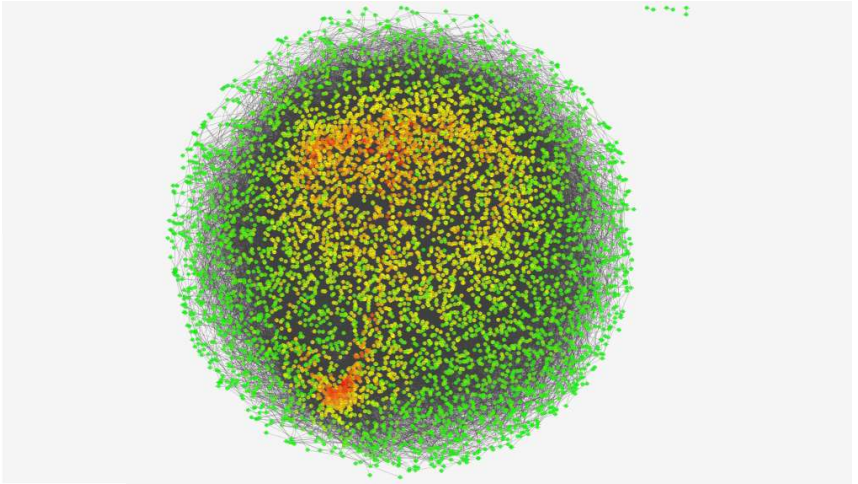

Yellowstone Park
Visitors per month
1e+06
8e+05
6e+05
4e+05
2e+05
0
2014-01
2014-07
2015-01
2015-07
2016-01
2016-07
Temperature (F)
90°
80°
70°
60°
50°
40°
30°
20°
Visitors per month
Temperature (F)

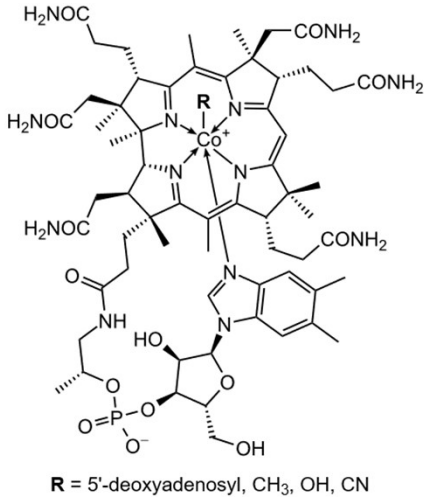
DNA-RNA-Protein序列



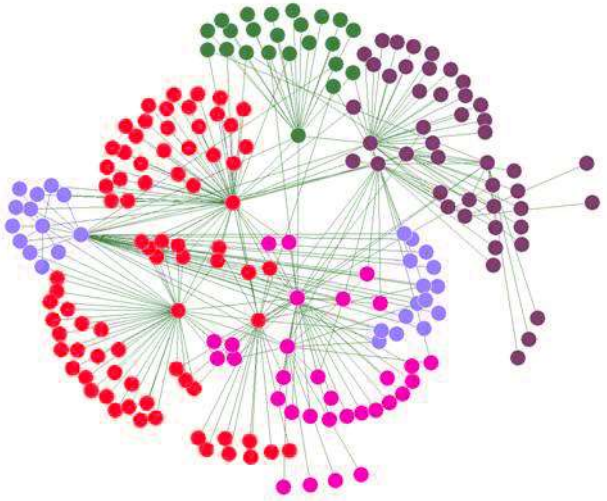
图/网络数据

基因共表达网络(Co-Expression Network)





化合物(Chemical Compounds)



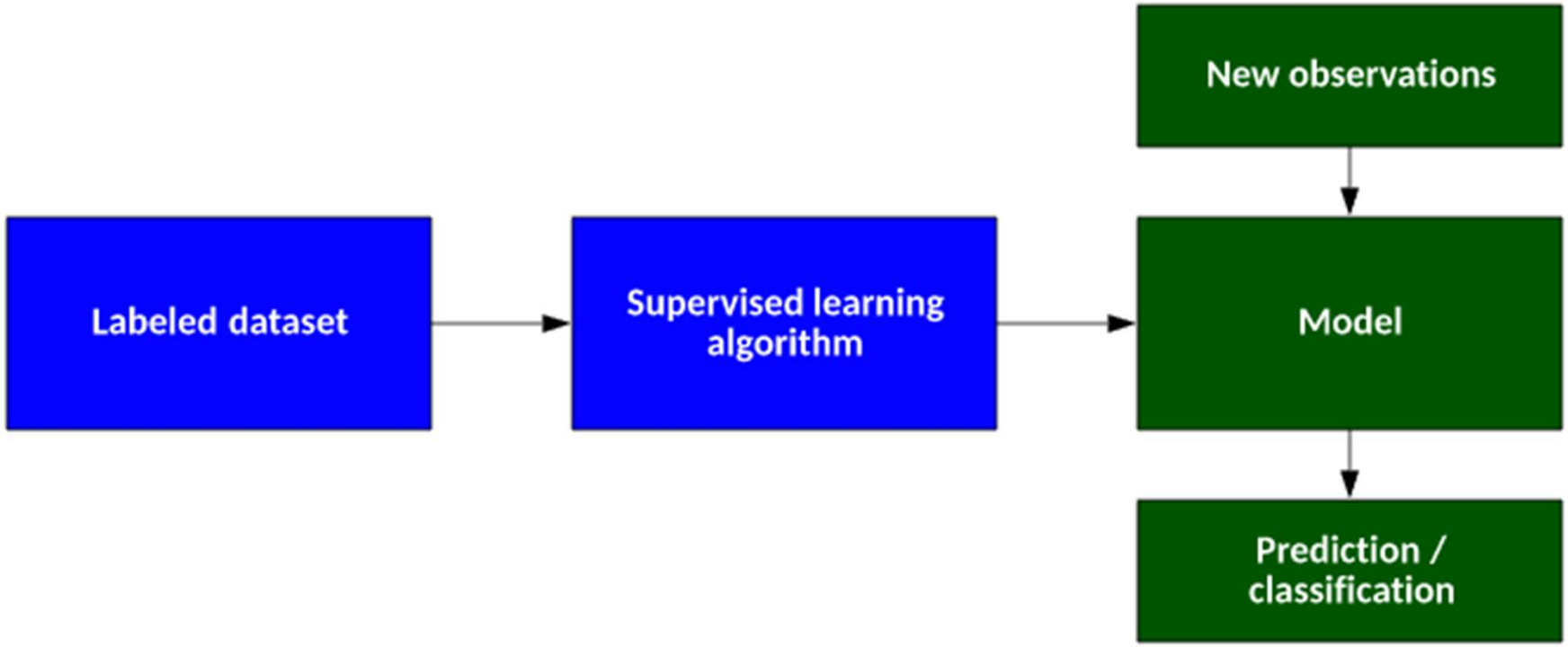
社交网络(Social Network)

学习任务的基础类型
Types of Basic Learning Tasks

- 有监督学习（Supervised Learning）
  - 分类(Classification)
  - 回归(Regression)
- 无监督学习（Unsupervised Learning)
  - 聚类(Clustering)
  - 潜因子发现(Latent Factor Discovering)
  - 关联(Association)
  - 矩阵填空(Matrix Completion)
- 强化学习（Reinforcement Learning）

- 有监督学习的目标：
  - 学习从输入到输出的映射，输出的正确值是由监督者所提供的

有监督学习
Supervised Learning



分类
Classification

分类任务定义：学习从输入$\mathbf{x}$到输出y的一个映射，其中$y \in \{1, \ldots, C\}$，而C是类别的数目。

如果$C = 2$，这称为**双类别分类**(Binary Classification)；
如果$C > 2$，则称为**多类别分类**。

如果类别标签不是互斥的，则称之为**多标签分类**。

- 目标：从数据中学习一个分类模型，它可以根据数据记录的条件属性取值来预测类别信息。
  - 换而言之，分类是学习从k个属性取值的向量到类别属性的一个映射。

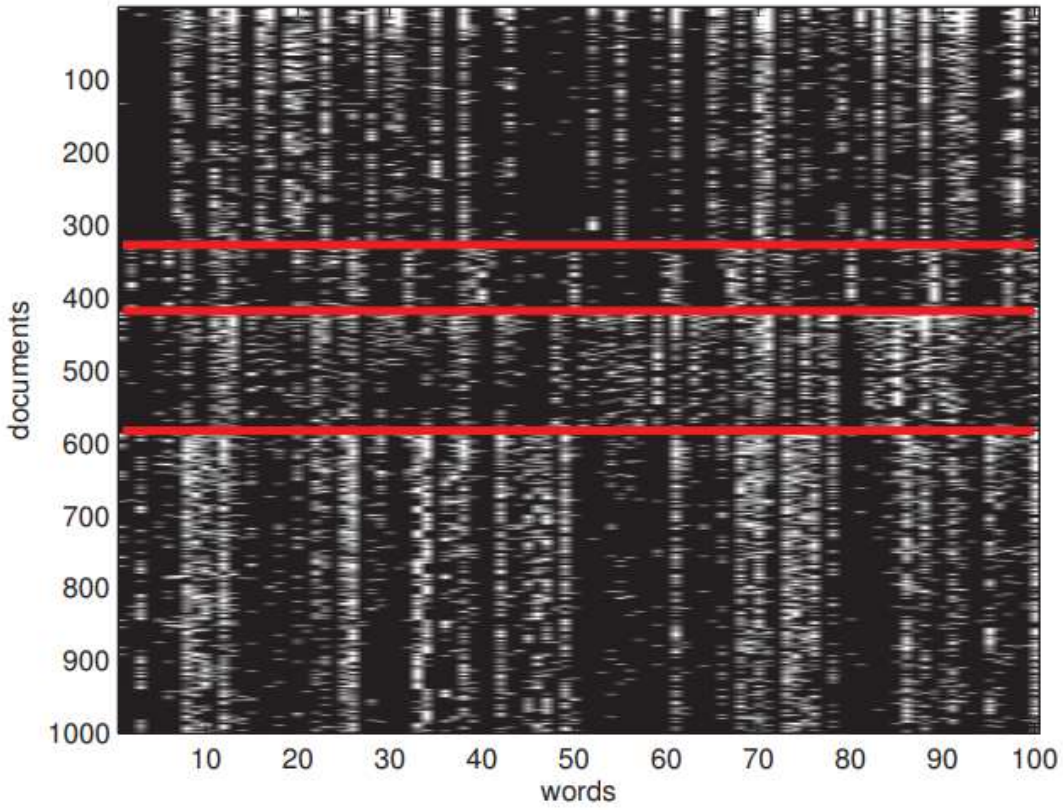
信用卡申请的批准决策

- 信用卡公司收到数千份信用卡的申请，每份申请都含有关于申请人的信息：
  - 年龄
  - 婚姻状态
  - 年薪
  - 债务
  - 信用等级
  - …
- 问题：决定一份申请是否应当被批准，或者说对申请进行分类：批准 与 不批准

文档分类与垃圾邮件过滤
Document classification and email spam filtering

- 文档分类的目标：将文档分类为C个不同类别中的一个

$$c^* = \underset{c}{\operatorname{argmax}}\, p(y = c|\mathbf{x}, \mathcal{D})$$



词袋**(bag of words)** 表示：如果词j j出现在文档i中，则定义$x_{ij} = 1$

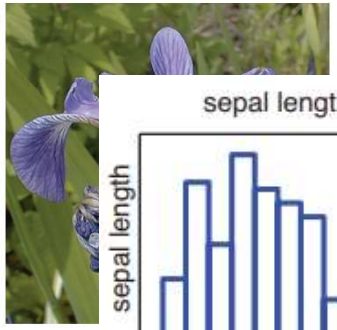
垃圾邮件过滤可以看作是文档分类：大多数垃圾邮件都以较大的概率含有一些词，如“buy”、“cheap”、“viagra”等等

例子：文档集

- 在一个文本文档数据集中，每份文档都可以被表示为一个关键词“袋”：
  - doc1: {Student, Teach, School }
  - doc2: {Student, School }
  - doc3: {Teach, School, City, Game}
  - doc4: {Baseball, Basketball}
  - doc5: {Basketball, Player, Spectator}
  - doc6: {Baseball, Coach, Game, Team}
  - doc7: {Basketball, Team, City, Game}

鸢尾花分类(Iris Classification)

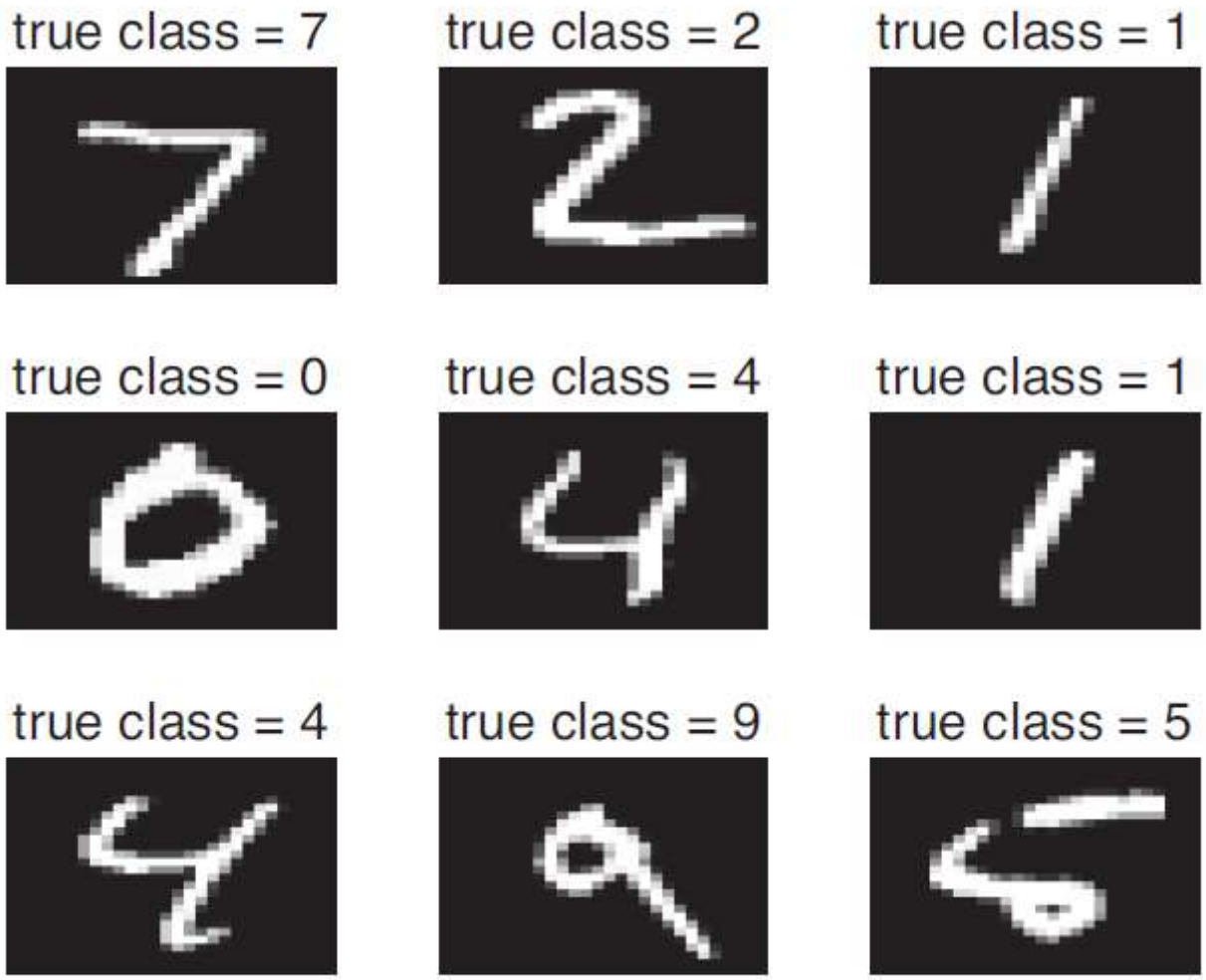
(a)



三类鸢尾花：setosa

探索性数据分析
(Exploratory
data analysis)

图像分类与手写体识别


true class = 7
true class = 2
true class = 1
true class = 0
true class = 4
true class = 1
true class = 4
true class = 9
true class = 5

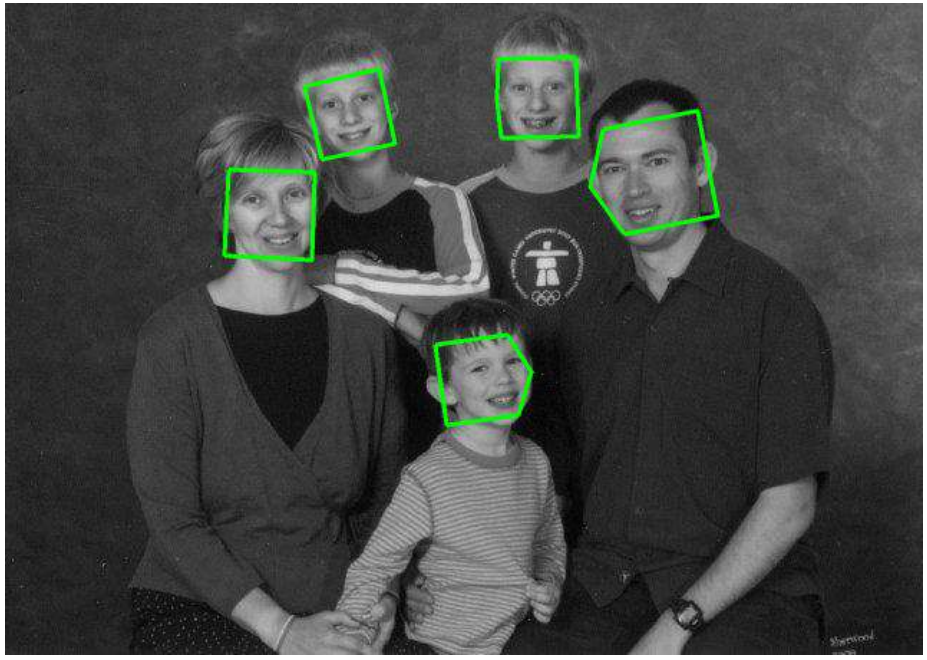
人脸检测
Face Detection

- 任务：在一幅图像中检测出人脸。

一张图片上可能有多个人脸，可以采用滑动窗口检测技术。

- 目前大多数数码相机已经提供了该功能，人脸的位置可以作为自动对焦的中心
- 另一个应用是Google的StreetView系统中模糊掉人脸

人脸检测例子



人脸识别
Face Recognition

检测出人脸后，接着可能要进行人脸识别，即估计出这个人的身份

- 类别标签数目可能会非常大
- 使用的特征应当不同于人脸检测问题
  - 识别：人脸间的细微差异对确定身份非常重要；
  - 检测：不考虑细节差异，只关注于人脸和非人脸的差异。

相关问题：视觉对象的检测与识别

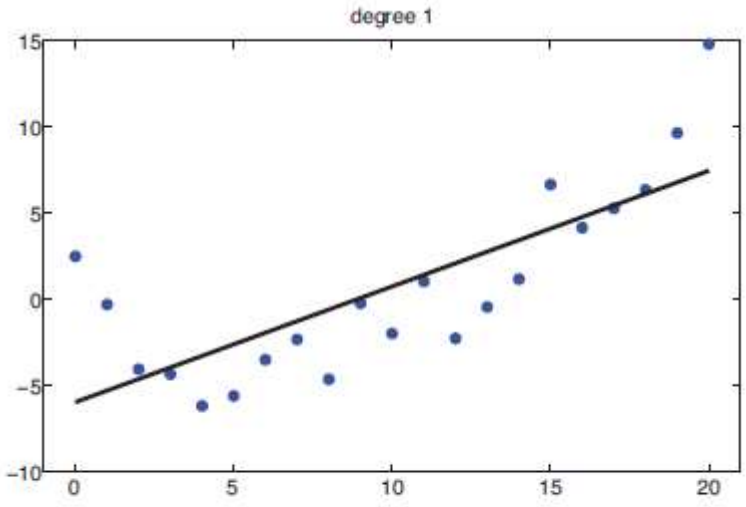
回归
Regression

回归与分类非常相似，不同之处在于回归问题的响应变量是连续型变量。

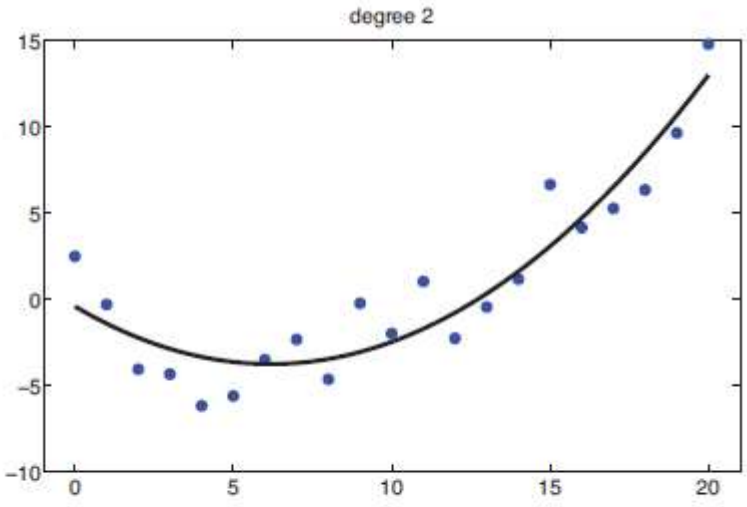
线性回归模型：建模一个标量响应变量和一个或多个解释变量之间线性关系

线性回归vs非线性回归

回归
一个简单的例子



(a)



(b)

回归：应用例子

- 根据当天的市场情况和其它可能的信息来预测明天的股市价格
- 预测正在YouTube上观看一部给定视频的用户的年龄
- 根据一系列不同的临床指标来预测体内的前列腺特异性抗原的量
- 使用气象数据、时间、门传感器等信息来预测一栋建筑物内部任一给定位置处的温度

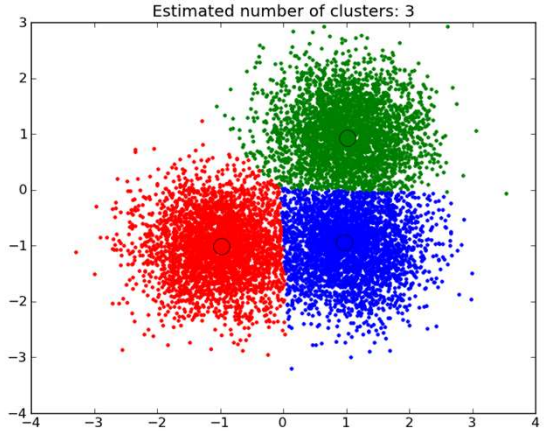
无监督学习

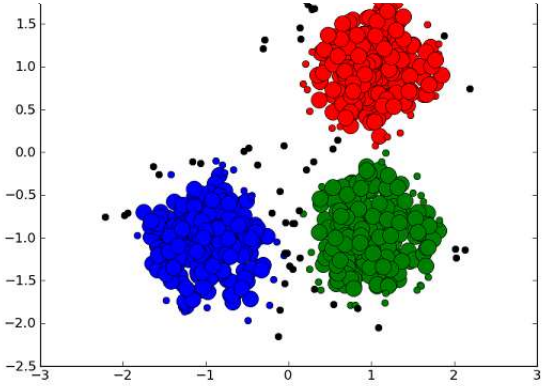
- **无监督学习**可以表述为密度估计任务： $p(\mathbf{x}_i|\boldsymbol{\theta})$的建模
  - 非条件的密度估计(unconditional density estimation)
  - 多元概率模型

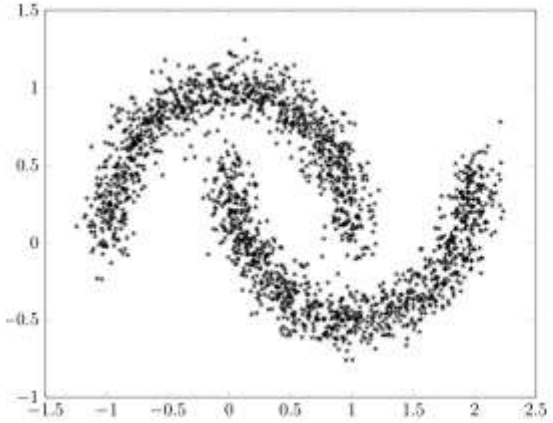
- **无监督学习的典型例子：**
  - 聚类分析
  - 发现潜因子
  - 矩阵填空

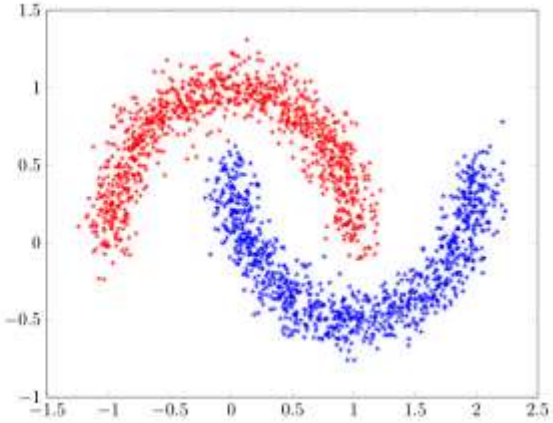
聚类分析
Clustering

- 聚类是一种公认的无监督学习的例子
  - 将数据对象分组成多个聚簇(clusters)
  - 同一个聚簇的对象之间具有较高的相似
  - 不同聚簇的对象之间相似度较低。

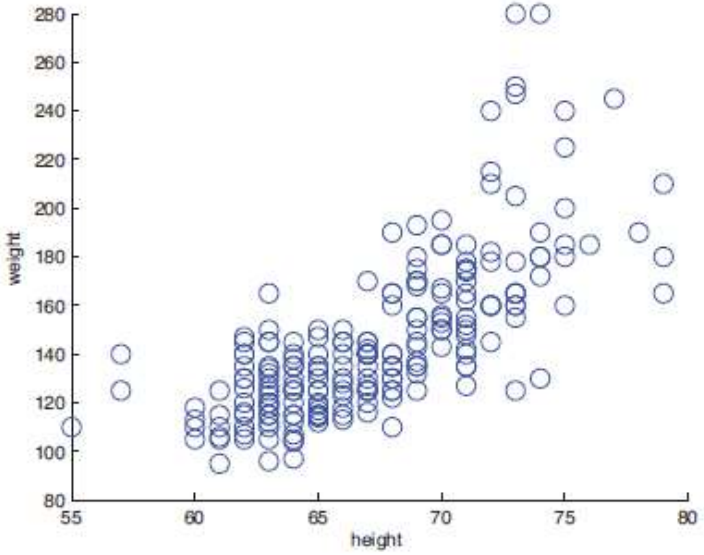




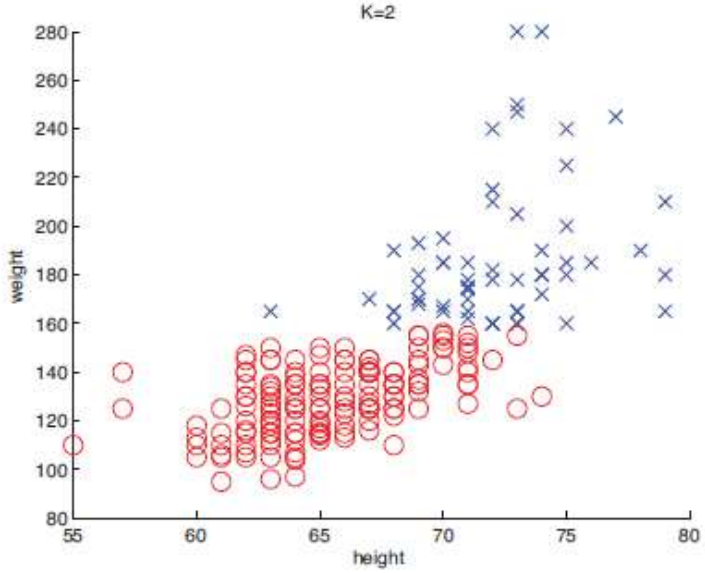




聚类分析



(a)



(b)

目标1：估计出在聚簇数目上的分布$p(K|\mathcal{D})$

$$K^* = \underset{K}{\operatorname{argmax}}\, p(K|\mathcal{D})$$

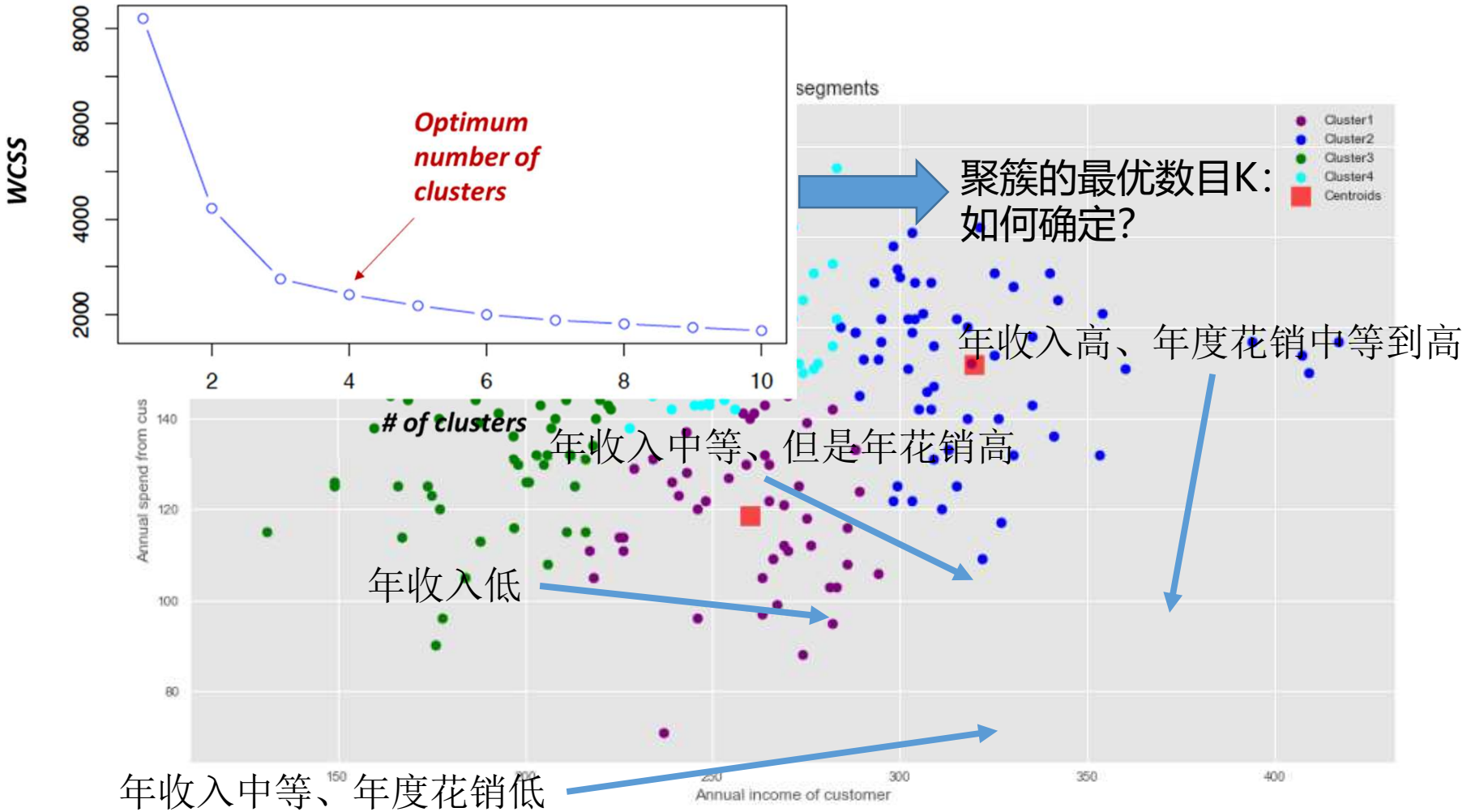
目标2：就是估计出每个数据点隶属于哪一个聚簇

$$z_i^* = \underset{k}{\operatorname{argmax}}\, p(z_i = k|\mathbf{x}_i, \mathcal{D})$$

聚类-可能的应用

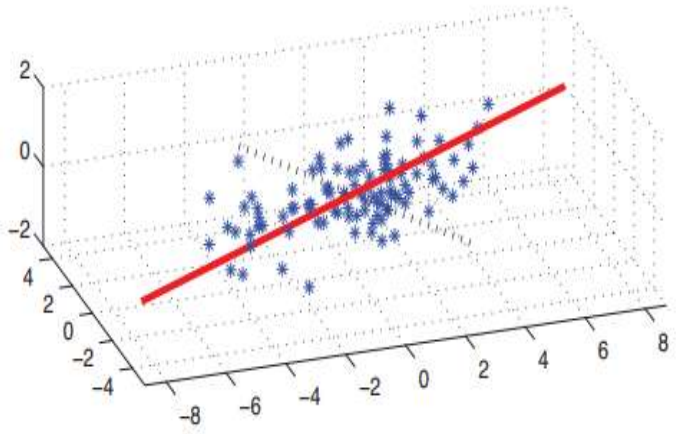
- 天文学中，新型恒星的发现
- 生物学中，不同细胞亚群的发现
- 电子商务中，用户群组发现→针对性营销广告
- 社会网络中，社团发现
- 搜索引擎中，搜索结果的分组

聚类应用场景
营销：客户细分

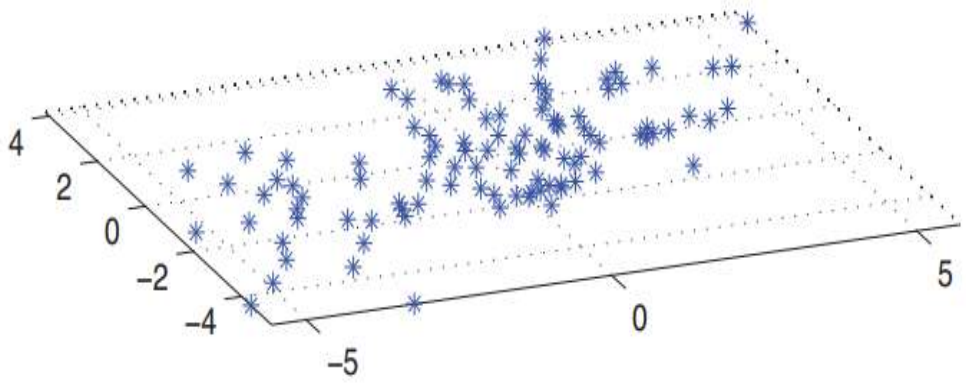


潜因子发现
Discovering Latent Factors

- 维度约简(Dimensionality Reduction)
  - 将高维数据投影到低维子空间来进行降维
  - 动机：高维数据往往只含有少量的潜因子
  - 用途：提升预测精度、快速近邻搜索、数据可视化



(a)



(b)

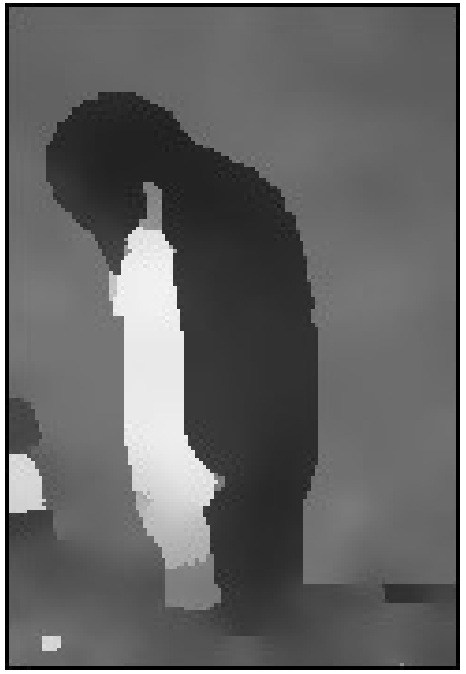
矩阵填空
Matrix Completion

- 数据有时会发生缺失的情况，即某些变量的值未知。
  - 问卷调查(survey)
  - 多个传感器故障。
- 数据矩阵中将含有一些“空洞”
  - 缺失数据项标记为NaN，代表“not a number”。

- 矩阵填空(Matrix Completion)：为这些缺失数据项推测出最可能的值。
  - 又称为“数据插补(imputation)”

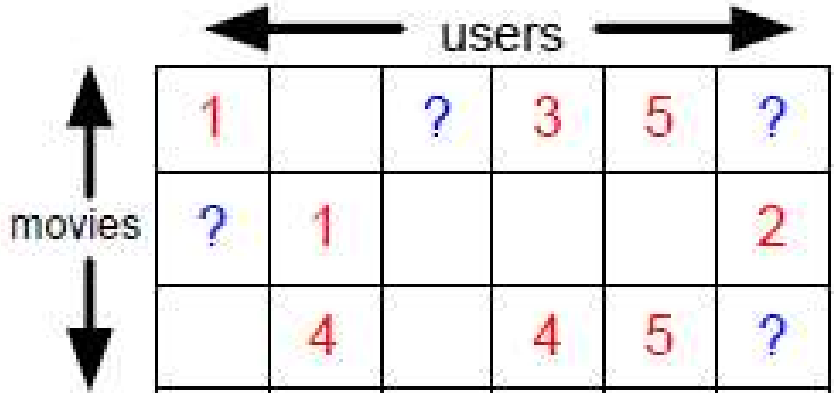
矩阵填充的例子：图像修复
Image Inpainting

- 图像修复的目标是使用逼真的纹理(realistic texture)来填充一幅图像中的空洞（可能是因为刮擦或是遮蔽所产生的）



矩阵填充的例子：
协同过滤(Collaborative Filtering)

- 协同过滤是另一种与“数据插补”相类似的任务。
- 协同过滤的一个常见例子是根据用户对过去看过的电影进行的评分来预测他们可能想看的电影是什么。
- 如果我们仅仅根据评分矩阵来进行预测，就是对于每个用户，需[illegible]平分的电影。



矩阵填充的例子：协同过滤
Collaborative Filtering

- DVD租赁公司Netflix在2006年启动了一项竞赛（http://netflixprize.com/）.
  - Netflix公司提供了一个大型的评分矩阵，评分标准为1到5，这是大约50万用户对1万8千部电影所创建的评分。
  - 整个矩阵将有大约 9×10^9 个数据项，但是其中仅可以观擦到大约1%的数据项，因此矩阵是非常稀疏的。
  - 这些可观察到的数据项的子集被用来训练，剩余部分用于测试。竞赛的目标是要能比Netflix的已有系统具有更高的预测精度。
- 2009年9月21日，奖金被授予了一个名为“BellKor’s Pragmatic Chaos”的研究者团队
  - 关于这个团队以及他们的方法可以在下列网址找到 http://www.netflixprize.com/community/viewtopic.php?id=1537.

矩阵填空的例子：购物篮分析
Market Basket Analysis

- 购物篮交易：

  t_1: {bread, cheese, milk}

  t_2: {apple, eggs, salt, yogurt}

  … …

  t_n: {biscuit, eggs, milk}

- 概念:
  - 项: 购物篮里面的一个物品
  - I: 商店里出售的所有物品的集合
  - 一次交易: 购物篮中所购买的物品集
  - 交易数据库: 所有交易的集合

关联规则
Association Rules

- 由 Agrawal等人在1993年提出
  - 它是一项重要的数据挖掘任务，得到了数据库和数据挖掘界的广泛研究。
- 它假定所有的数据都是范畴性的
  - 大多数算法都不能很好地处理数值数据。
- 最初用于购物篮(Market Basket)分析，以发现顾客所购买的商品项的关联性。
-
- Bread→Milk [sup = 5%, conf = 100%]

关联规则-数据模型

- 令$I = \{i_1, i_2, \ldots, i_m\}$为所有项的集合。
- 交易t，又称为事务:（transanction）
  - t是一个项集，即$t \subseteq I$.
- 交易/事务数据库Transaction Database T:
  - 交易/事务的集合$T = \{t_1, t_2, \ldots, t_n\}$.
- 关联规则是形如$X \rightarrow Y$的蕴涵式
  - 其中X和Y都是项集，且 $X \cap Y = \varnothing$

关联规则-强度度量

- 我们称交易 t 含有项集 X，如果 $X \subseteq t$.

- 支持度(Support):
  - 我们称规则X→Y成立的支持度为s，如果数据库T中有s%的交易含有$X \cup Y$，即$s = Pr(X \cup Y)$。
- 信任度(Confidence):
  - 规则X→Y在T中成立的信任度为c，如果c%含有X的交易也含有Y，即 $c = Pr(Y|X)$。
- An association rule is a pattern that states when X occurs, Y occurs with certain probability.

分类方法

- 逻辑斯蒂回归(Logistic Regression)
- 神经网络(Neural Network)
- 最近邻分类器(Nearest Neighbor)
- 决策树(Decision Tree)
- 贝叶斯分类器(Bayesian Classifier)
  - 朴素贝叶斯
  - 树状贝叶斯
  - 贝叶斯网络
- 支持向量机(Support Vector Machine)

- 分类器集成(Classifier Ensemble)
  - Bagging, AdaBoost, Stacking, Cascading, Random Forest, …

决策树分类

- 一种简单的算法框架（树结构通过自顶向下的递归方式构建）
  - 起初，所有的训练样例都位于根节点
  - 根据所选择的属性（或属性的组合）将训练样例进行递归的划分
  - 属性（或属性组合）的选择是基于某种纯度函数的（例如，信息增益）
- 停止划分的条件：
  - 给定节点的所有训练样例都属于同一个类别
  - 不存在属性可以进行进一步的划分（给定节点的所有训练样例在所有条件属性上都具有相同取值）
  - 给定节点不存在训练样例（或者训练样例的数目低于某个给定的阈值）

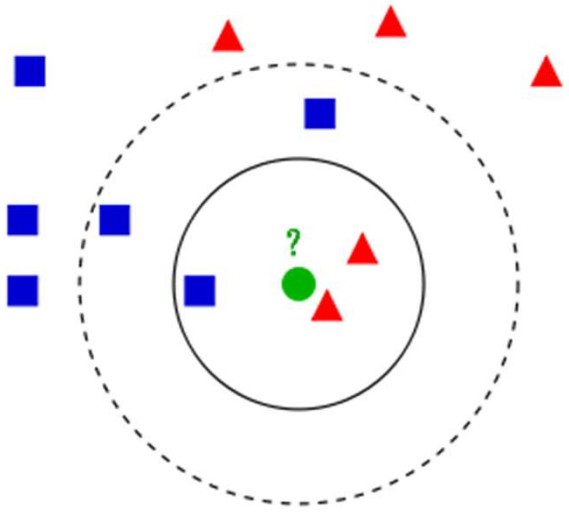
贝叶斯分类

- 设条件属性A_1到A_k都取离散值，决策属性为C
  - 给定的测试样例d的观测属性值为 a_1到a_k
- 贝叶斯分类采用概率的方法，它计算如下的后验概率：$P(C = c_i|A_1 = a_1, \dots, A_k = a_k)$
  - 具有最大后验概率值的那个类别c_i作为预测值输出
- 后验概率如何计算（贝叶斯定理）：
  - $P(C = c_i|A_1 = a_1, \dots, A_k = a_k) = \frac{P(A_1 = a_1, \dots, A_k = a_k|C = c_i) \times P(C=c_i)}{P(A_1=a_1,\dots,A_k=a_k)}$

k-最近邻分类

- k-近邻分类仅仅根据特征空间中与待预测样例最接近的k个训练样例来进行“多数票选”的分类决策
  - k-最近邻分类并不根据训练数据习得任何模型

- 什么是“最接近（或最相似）”？
  - 度量学习(Metric Learning) – to learn the distance function
  - 特征抽取(Feature Selection)
  - 维度约简(Dimensionality Reduction)
- 参数选择
- 数据约简(Data Reduction)
  - 仅仅部分数据对于分类重要



线性回归

$$y(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + \epsilon = \sum_{j=1}^{D} w_j x_j + \epsilon$$

逻辑斯蒂回归

$$\sigma(\eta) = \frac{1}{1 + \exp(-\eta)}$$

支持向量机分类

- 支持向量机是由 俄罗斯的V. Vapnik和他的同事在1970s发明的，但是在1992年才为世界所知晓。

- SVM是一种线性分类器，它寻找一个超平面来分割两类数据：正例和反例。
  - 利用核函数(Kernel functions)可以实现非线性的分割。

- SVM不仅具有良好的理论基础，在应用中还具有很高的分类精度，特别是对高维数据。
  - 它可能是用于文本分类的最优分类器。

分类器集成

- Bagging
- Boosting
- Stacking
- Cascading
- Random Forest
- ……

二元分类的混淆矩阵示例

		True/Actual	
		Positive ()	Negative
Predicted	Positive ()	7	0
	Negative	0	3

		True/Actual	
		Positive ()	Negative
Predicted	Positive ()	5 (TP)	1 (FP)
	Negative	2 (FN)	2 (TN)

二元分类问题
精度、召回、准确率

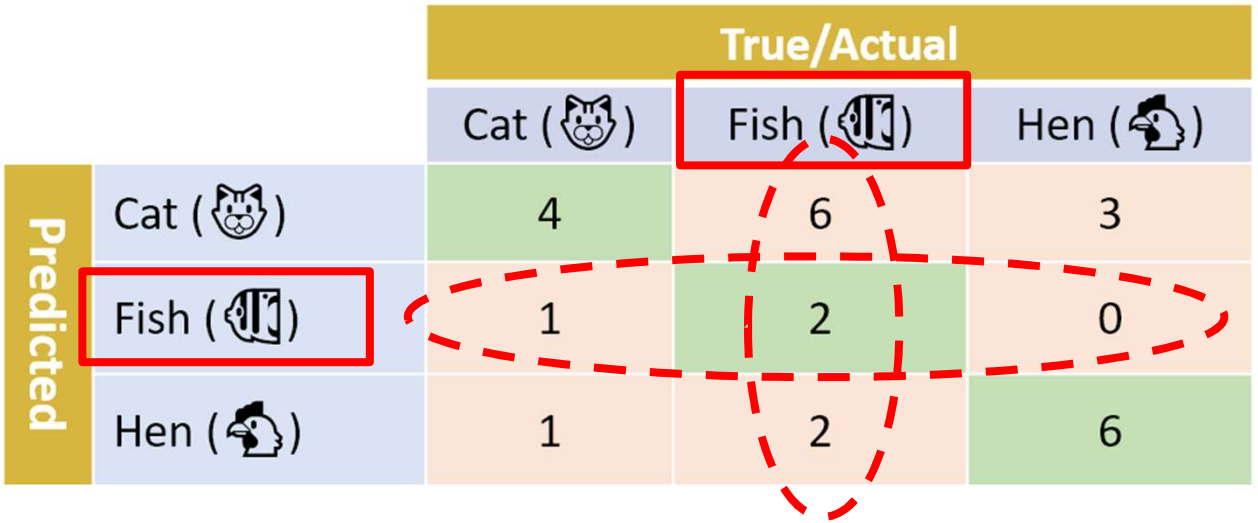
		True/Actual	
		Positive ()	Negative
Predicted	Positive ()	5 (TP)	1 (FP)
	Negative	2 (FN)	2 (TN)

$$Precision = \frac{TP}{TP + FP} = \frac{5}{5 + 1} = 83.3\%$$

$$Recall = \frac{TP}{TP + FN} = \frac{5}{7} = 71.4\%$$

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} = \frac{5 + 2}{5 + 1 + 2 + 2} = 70\%$$

多类别分类问题
混淆矩阵



$$Precision = \frac{TP}{TP + FP} = \frac{2}{2 + (1 + 0)} = 66.7\%$$

类别Fish:

$$Recall = \frac{TP}{TP + FN} = \frac{2}{2 + (6 + 2)} = 20.0\%$$

类似地，Cat: Precision=4/13=30.8%，Recall=4/6=66.7%
Hen: Precision=6/9=66.7%，Recall=6/9=66.7%

二元分类问题
F1值

		True/Actual	
		Positive ()	Negative
Predicted	Positive ()	5 (TP)	1 (FP)
	Negative	2 (FN)	2 (TN)

F1值是精度和召回的调和平均：

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} = \frac{2 \times 83.3\% \times 71.4\%}{83.3\% + 71.4\%} = 76.9\%$$

多类别分类任务 F1值

		True/Actual		
		Cat (🐱)	Fish (🐟)	Hen (🐔)
Predicted	Cat (🐱)	4	6	3
	Fish (🐟)	1	2	0
	Hen (🐔)	1	2	6

Class	Precision	Recall	F1-score
Cat	30.8%	66.7%	42.1%
Fish	66.7%	20.0%	30.8%
Hen	66.7%	66.7%	66.7%

组合每个类别的F1分值，得到单个F1分值：

- 宏平均(Macro)
- 加权平均(Weighted)
- 微平均(Micro)

Macro-F1 = (42.1% + 30.8% + 66.7%) / 3 = 46.5%

Weighted-F1 = (6 × 42.1% + 10 × 30.8% + 9 × 66.7%) / 25 = 46.4%

micro-F1 = micro-precision = micro-recall

分类任务的其他评测指标

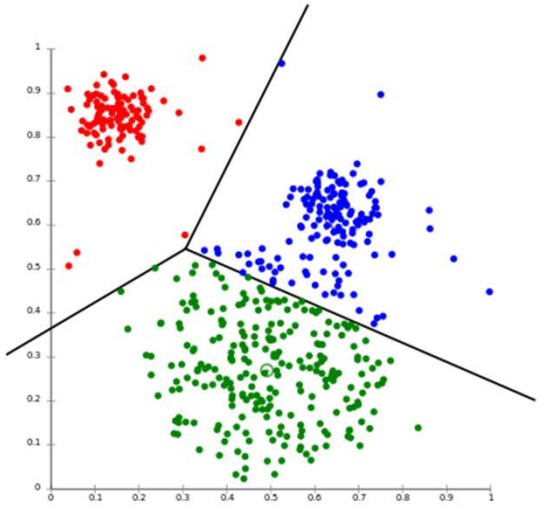
- ROC and its AUC
- PR Curve and its AUC
- Cohen’s Kappa score
- Mattew’s Correlation Coefficient

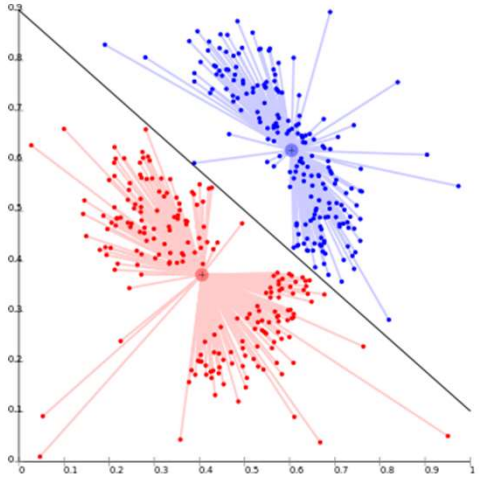
回归方法

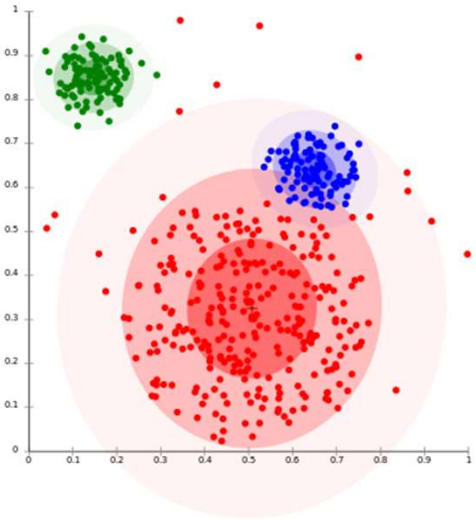
- 多元线性回归(Multiple Linear Regression)
- 最近邻回归(Nearest Neighbor Regression)
- 核回归(Kernel Regression)
- 反向传播算法（神经网络）
- 支持向量回归
- 回归树(Regression Tree)

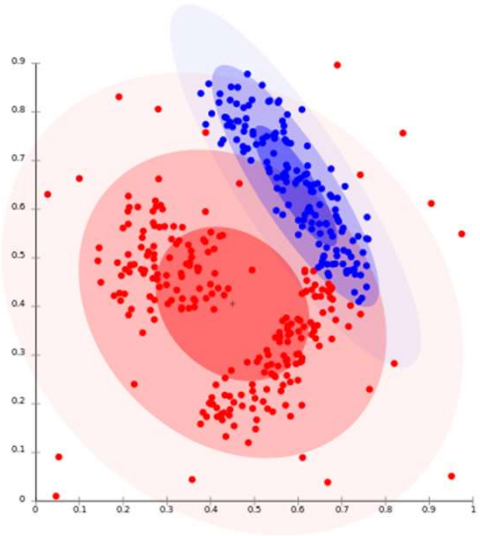
聚类方法

- 划分聚类方法
  - k-Means, k-Modes, k-Medoids
  - Gaussian Mixture
- 层次聚类方法
  - Complete-Linkage
  - Single-Linkage
  - Average-Linkage
- 谱聚类(Spectral Clustering)
- 聚类集成(Cluster Ensemble)
- 其它(数据挖掘)：
  - 基于密度的聚类方法
  - 基于网格的聚类方法



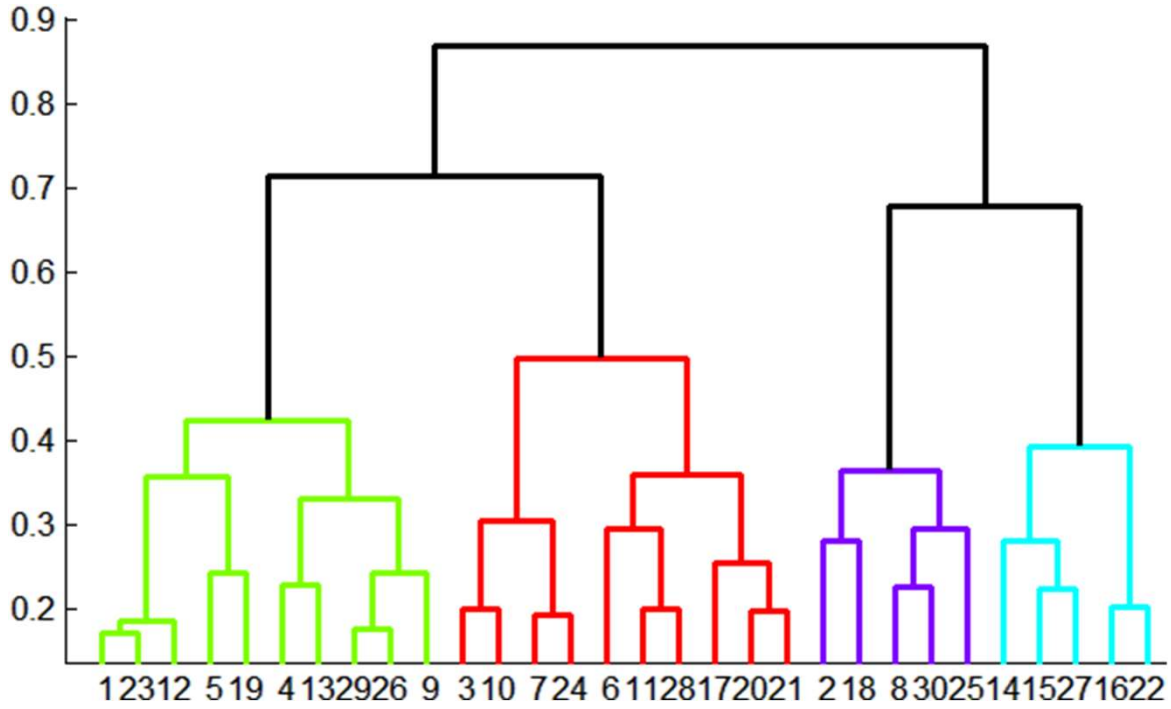






层次聚类算法

- 层次聚类将数据样本分组到聚簇的树状层次结构中。
  - 这个聚簇的层次结构被称为树状图(Dendrogram)。



层次聚类算法 - 凝聚与分裂

- 凝聚(Agglomerative)聚类：自底向上
  - 它从底层开始建立树状图(dendrogram)，它不断地合并最相似（或距离最近）的一组聚簇，直到所有的数据点都合并到一个单一的聚簇（根聚簇）才停止。
- 分裂(Divisive)聚类：自顶向下
  - 它从最顶部的根节点开始，根节点中含有所有的数据点；它将根节点分割成一系列子聚簇的集合，又进一步递归地分裂每一个子聚簇，直到每一个聚簇只含有单个的数据点为止。

机器学习的十大经典算法

- C4.5 （分类）
- k-Means（聚类）
- SVM（分类/回归）
- Apriori（关联）
- EM（期望最大化）
- PageRank（排序）
- AdaBoost（分类/回归）
- K-nearest neighbor（分类）
- Bayesian classification（分类）
- CART - Classification and Regression Tree（分类/回归）