# 数据挖掘与机器学习
# 复习&回顾

# 课程目录

- 第一章 绪论（数据挖掘+机器学习）
- 第二章 数据预处理
- 第三章 分类与预测
- 第四章 聚类分析
- 第五章 关联规则

# 第一章 绪论

- 什么是数据挖掘?
- 为什么需要数据挖掘?
- 数据挖掘与传统数据分析方法的区别?
- 数据挖掘的任务是什么?
- 数据挖掘的一般流程(如何挖掘)？

# 什么是数据挖掘？

- **数据挖掘 (定义)**
  - 从大量的、不完全的、有噪声的、模糊的数据中挖掘那些有用的、隐含的、先前未知的模式或知识的过程。
- **其定义所包含的意义：**
  - 数据源必须是真实的、大量的；
  - 发现的是先前未知的知识；
  - 发现的知识要有用、可理解、可运用；
  - 这些知识是相对的，是有特定前提和约束条件的，在特定领域中具有实际应用价值。

# 为什么需要数据挖掘？

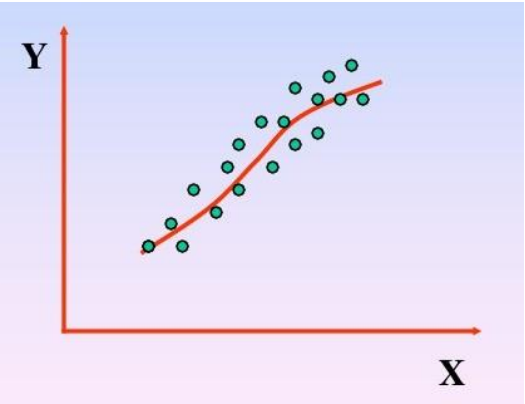
- 面临的问题
  - 我们虽然拥有丰富的数据，但却缺乏有用的信息。
- 解决途径
  - 数据挖掘：在大量的数据中挖掘感兴趣的信息（规则，规律，模式，约束）。
  - 因为隐藏在数据之后更深层次、更重要的信息能够描述数据的整体特征，可以预测发展趋势，在决策中具有重要价值。
  - 问题：数据挖掘与传统的数据分析方法有何区别？

# 数据挖掘与传统数据分析方法的区别

- **本质区别**
  - 数据挖掘是在没有明确假设的前提下去挖掘信息、发现知识；
  - 数据挖掘所得到的信息应具有先前未知，有效和可实用三个特征。
    - 先前未知是指信息是预先未曾预料到的，即：数据挖掘是要发现那些不能靠直觉发现的信息或知识，甚至是违背直觉的信息或知识，挖掘出的信息越是出乎意料，就可能越有价值。
    - 例：在商业应用中最典型的例子就是一家连锁超市通过数据挖掘发现了小孩尿布和啤酒之间有着惊人的联系。

# 数据挖掘的任务

- 分类预测型任务
  - 利用一些已知变量来预测未知的或其他变量将来的值。
  - 典型的方法是回归分析，即：利用大量的历史数据，以时间为变量建立线性(最小二乘法)或非线性(神经网络)回归方程。
  - 在预测时，只要输入任意的时间值，通过回归方程就可求出该时间的状态。



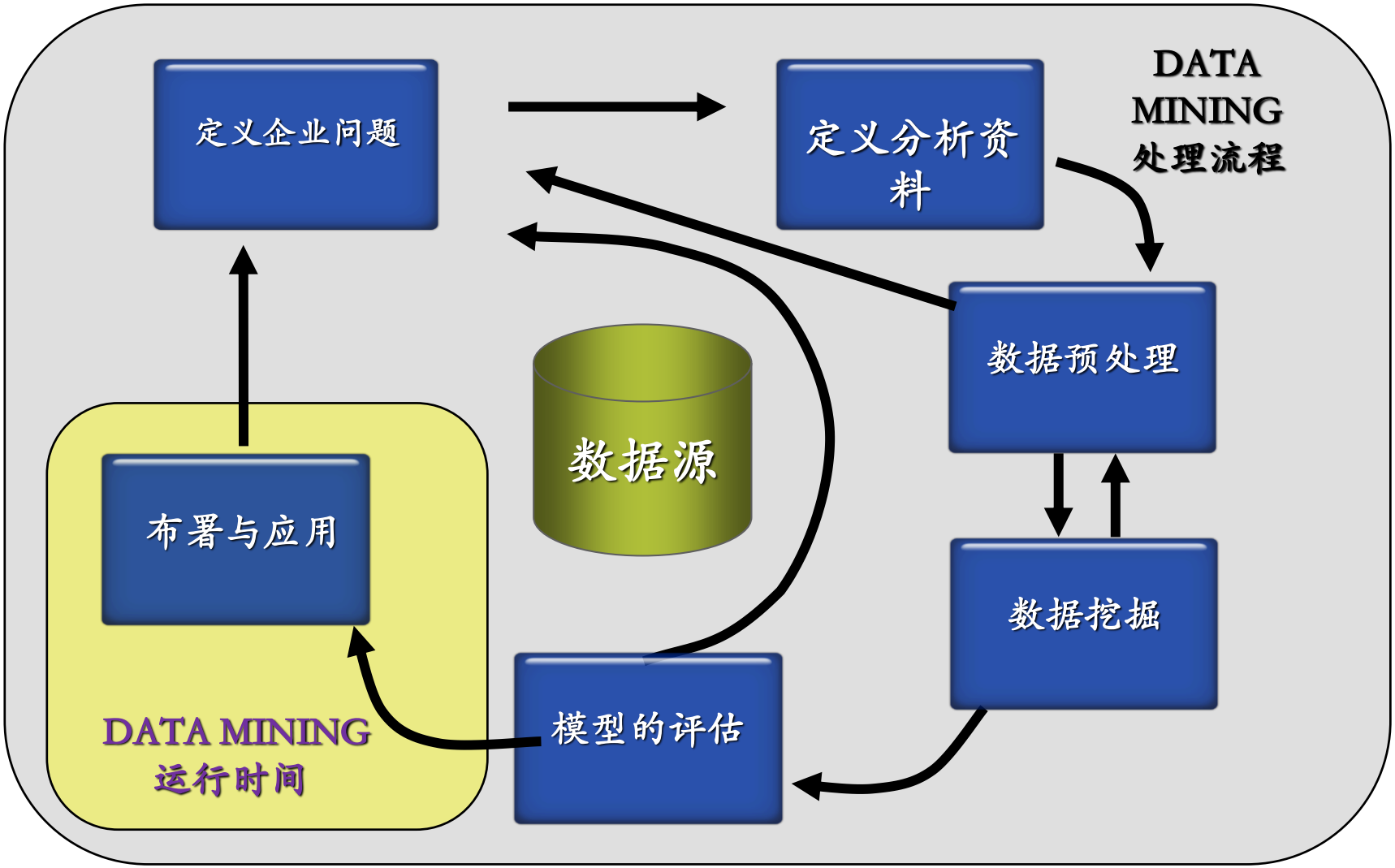
# 数据挖掘的任务

- 描述型任务
  - 找到人们可以解释的，能够描述数据的模式。
- 描述性任务主要包括：
  - 聚类：聚类任务用于把没有预定义类别的数据划分成几个合理的类别。
  - 摘要：摘要任务用于形成数据高度浓缩的子集及描述。
  - 依赖分析：依赖分析任务用于发现数据项之间的关系。

# 模式或知识发现（KDD）的过程

- 数据挖掘是知识发现的核心步骤

# 数据挖掘基本流程


定义企业问题
定义分析资料
DATA MINING
处理流程
数据预处理
数据源
布署与应用
数据挖掘
DATA MINING
运行时间
模型的评估

# 第一章 绪论

- 什么是机器学习?
- 机器学习三要素?
- 过拟合/欠拟合及其特点？
- 什么是机器学习的泛化能力？
- 如何平衡模型的方差和
- 机器学习的一般步骤？
- 监督学习与非监督学习的区别?
- 数据挖掘与机器学习的关系?

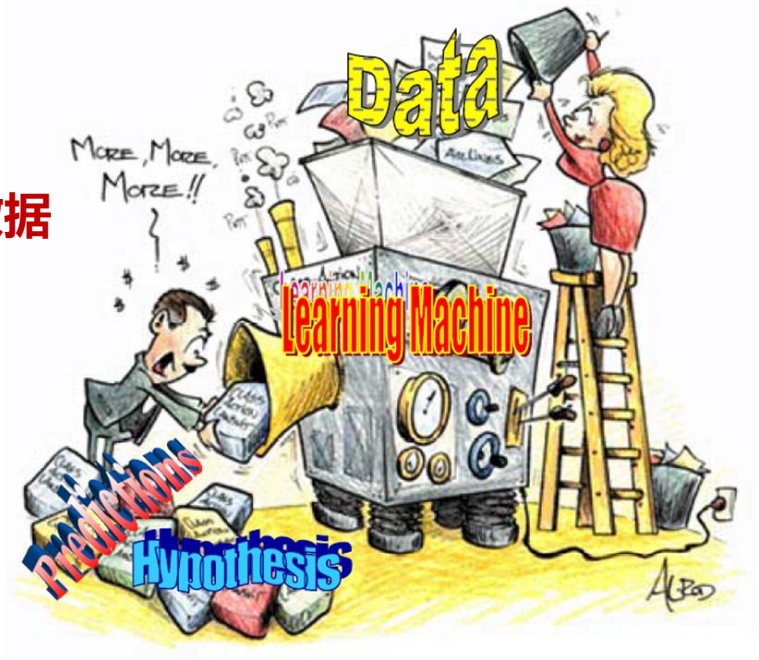
# 第一章 绪论

- 什么是机器学习?
- 机器学习三要素?
- 过拟合/欠拟合及其特点？
- 什么是机器学习的泛化能力？
- 如何平衡模型的方差和
- 机器学习的一般步骤？
- 监督学习与非监督学习的区别?
- 数据挖掘与机器学习的关系?

# 什么是机器学习？

- **机器学习 (通俗理解)**
  - 机器学习是指一种使计算机能够从已知数据中学习潜在规律和改进自身性能的技术和方法。简言之，它是让计算机在没有明确编程的情况下，通过分析数据产生“模型”，用于在新数据中识别模式、做出预测等

机器学习是一种**从数据中获得知识**的方法




所谓“**模型**”，本质上就是一个从输入到输出的**映射（函数）**

训练：
给定一组$\{(X, f(X))\}$，求$f$

预测：
对新的$X$，求$f(X)$

# 什么是机器学习（数学描述）

- 机器学习就是从给定的函数集$\{F(x,\alpha)\}$($\alpha$是参数)中，选择出能够最好地逼近训练器响应的函数。
- 机器学习的目标可以形式化地表示为：根据n个独立同分布的观测样本 $(x_1,y_1),(x_2,y_2),\cdots,(x_n,y_n)$ ，在一组函数 $\{F(x,\alpha)\}$ 中求出一个最优函数 $f(x,\alpha_0)$ 对训练器的响应进行估计，使期望风险最小

$$R(\alpha)=\int L(y,f(x,\alpha))dP(x,y)$$

其中 $L(y,f(x,\alpha))$ 是损失函数，对于不同类型的机器学习问题有不同形式的损失函数。$P(x,y)$是随机变量X, Y的联合概率分布函数，是未知的。

# 机器学习三要素

- 模型
  - 线性方法： $f(\mathbf{x}, \theta) = \mathbf{w}^{\mathrm{T}}\mathbf{x} + b$
  - 广义线性方法： $f(\mathbf{x}, \theta) = \mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}) + b$ ，其中$\phi(x)$为核函数
  - 非线性方法：根据情况而定，无统一的表示
- 学习策略（准则）
  - 损失函数：一次预测的好坏 $L(f(x), y)$
  - 期望损失（风险）：平均意义下模型预测的好坏

$$\mathcal{R}(f) = \mathbb{E}_{(\mathbf{x},y)\sim p(\mathbf{x},y)}[\mathcal{L}(f(\mathbf{x}), y)],$$

  - 模型训练的目标就是使得期望损失（风险）最小化，从而选择出最佳模型
- 优化方法

# 机器学习三要素

- 损失函数（常见）

**1. 0-1损失函数(0-1 Loss Function)**

$$\mathrm{L}\big(Y, f(X)\big) = \begin{cases} 1, & Y \neq f(X) \\ 0, & Y = f(X) \end{cases}$$

**2. 平方损失函数(Quadratic Loss Function)**

$$\mathrm{L}\big(Y, f(X)\big) = \big(Y - f(X)\big)^2$$

**3. 绝对损失函数(Absolute Loss Function)**

$$\mathrm{L}\big(Y, f(X)\big) = |Y - f(X)|$$

**4. 对数损失函数(Logarithmic Loss Function)**

$$\mathrm{L}\big(Y, P(Y|X)\big) = -\log P(Y|X)$$

# 机器学习三要素

- 损失函数将机器学习问题转化为最优化问题
  - 期望损失（风险）函数的值越小，模型性能越好，但是X,Y的联合概率分布是未知的，意味着我们无法求解期望损失。给定一个数据集，我们将训练数据集的平均损失称为经验损失（风险）

$$\frac{1}{N}\sum_{n=1}^{N} \mathrm{L}(y_n, f(x_n))$$

  - 当样本数量足够大时，根据大数定律，经验风险会近似于模型的期望风险，因此可以用经验风险来估计期望风险

  - 于是，期望风险最小化问题转化为在训练集上求解经验风险最小化的优化问题：

$$\min_{f} \frac{1}{N}\sum_{n=1}^{N} \mathrm{L}(y_n, f(x_n))$$

# 机器学习三要素

- 损失函数将机器学习问题转化为最优化问题
  - 然而，当样本数量不足时，单单利用经验风险最小化可能会导致“过拟合”的问题。
  - 解决办法一般是在原有的经验风险基础上加上用于控制模型复杂度的正则项(Regularizer)：

$$\min_{f} \frac{1}{N}\sum_{n=1}^{N} \mathrm{L}(y_n, f(x_n)) + \lambda J(f)$$

其中，$J(f)$ 代表对模型复杂度的惩罚。模型越复杂，$J(f)$ 越大，模型越简单，$J(f)$就越小。$\lambda$ 是一个正的常数，也叫正则化系数，用于平衡经验风险和模型复杂度。

# 过拟合与欠拟合

## 过拟合（Overfitting）

- **定义：** 过拟合指的是模型在训练数据上表现得非常好，但在新数据上的表现很差。换句话说，模型过于复杂，能够捕捉到训练数据中的噪声和细节，而这些噪声和细节在新数据中并不普遍存在。
- **特征：**

训练误差很低：模型在训练集上能够准确预测，大多数训练样本的误差很小。

测试误差较高：模型在测试集或验证集上的误差较大，表现不佳。

模型过于复杂：模型包含太多参数或特征，导致其能够学习到训练数据的具体细节和噪声。

- **解决方法：**

简化模型：减少模型的复杂度，如减少特征数量或层数。

正则化：应用正则化技术（如L1、L2正则化）来限制模型的复杂度。

交叉验证：使用交叉验证技术来选择模型参数，避免过于依赖训练集

数据增加：通过数据增强或收集更多训练数据来提高模型的泛化能力

# 过拟合与欠拟合

## 欠拟合（Underfitting）

- **定义**：欠拟合指的是模型在训练数据和新数据上的表现都很差。换句话说，模型过于简单，无法捕捉到数据中的重要模式和关系。
- **特征**：

  训练误差很高：模型在训练集上的误差也很高，说明模型不能很好地拟合训练数据。

  测试误差较高：模型在测试集或验证集上的误差较大，表现不佳。

  模型过于简单：模型的复杂度不足，无法捕捉数据中的复杂关系。
- **解决方法**：

  增加模型复杂度：使用更复杂的模型或增加特征数量，例如增加神经网络的层数或节点数。

  减少正则化：减少正则化强度，使模型能够拟合更多的训练数据。

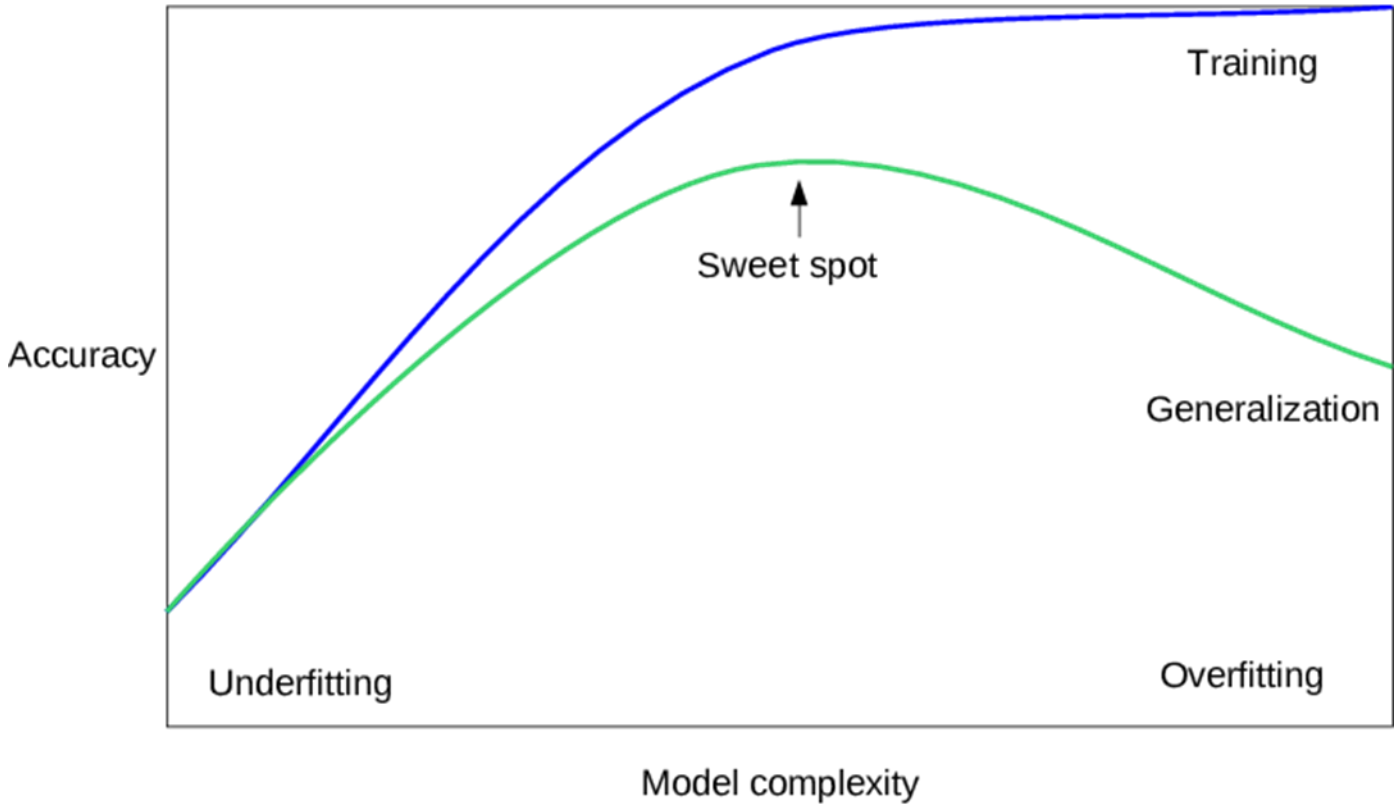
  特征工程：通过构造或选择更相关的特征来改善模型的表现。

  训练更久：增加训练时间，让模型有足够的时间学习数据中的模式。

# 过拟合与欠拟合

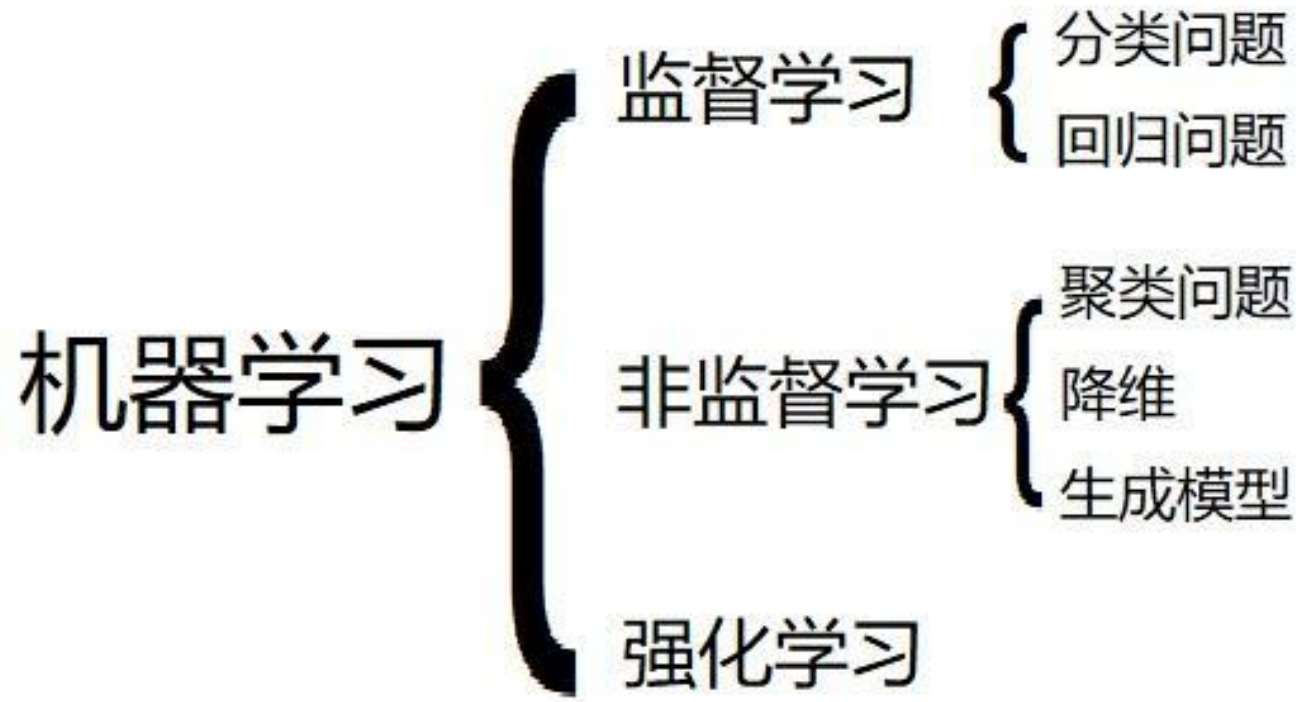
- **模型复杂度与泛化能力的关系**

泛化能力：机器学习的目标是使得学习到的模型都能很好的适用于“新样本”，而不仅仅是在训练集上报表现出色，我们将模型适用于新样本的能力称为泛化（Generalization）能力。



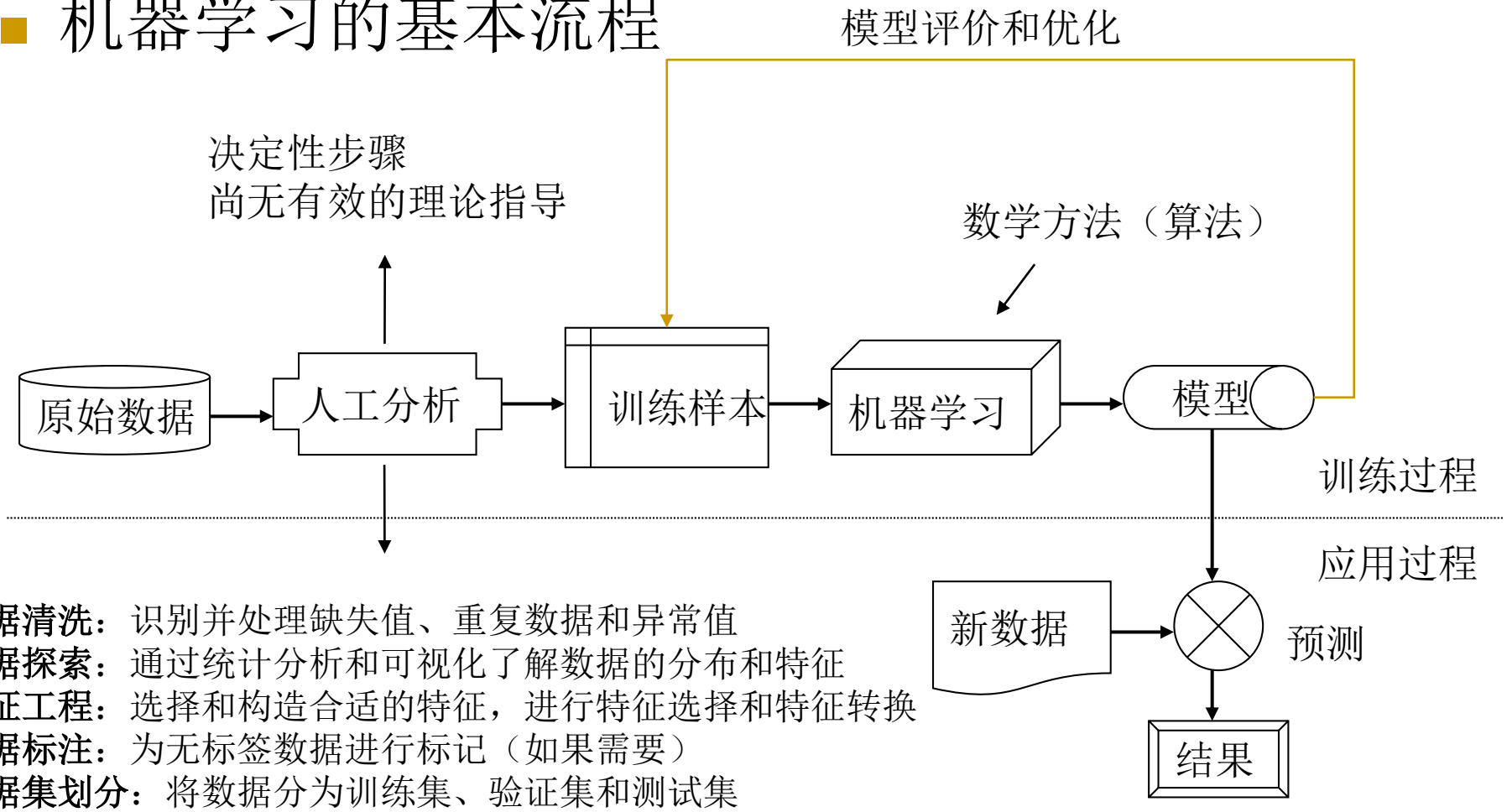
# 机器学习的分类

- 机器学习的三大领域（主流划分）



# 机器学习一般步骤

- 机器学习的基本流程




- **数据清洗：** 识别并处理缺失值、重复数据和异常值
- **数据探索：** 通过统计分析和可视化了解数据的分布和特征
- **特征工程：** 选择和构造合适的特征，进行特征选择和特征转换
- **数据标注：** 为无标签数据进行标记（如果需要）
- **数据集划分：** 将数据分为训练集、验证集和测试集
- **数据平衡：** 处理类别不平衡问题，如通过过采样或欠采样方法

# 思考：机器学习与数据挖掘的关系？

- 机器学习是数据挖掘的重要工具。
- 数据挖掘不仅仅要研究、拓展、应用一些机器学习方法，还要通过许多非机器学习技术解决数据仓储、大规模数据、数据噪音等等更为实际的问题。
- 机器学习的涉及面更宽，常用在数据挖掘上的方法通常只是“从数据学习”，然则机器学习不仅仅可以用在数据挖掘上，一些机器学习的子领域甚至与数据挖掘关系不大，例如强化学习与自动控制等等。
- 数据挖掘试图从海量数据中找出有用的知识。
- 大体上看，数据挖掘可以视为机器学习和数据库的交叉，它主要利用**机器学习界提供的技术来分析海量数据，利用数据库界提供的技术来管理海量数据**。

# 回顾
# Review

- 什么是正则化?
- 正则化有哪几种方式?
- 特征工程主要包括哪些内容？
- 模型的评估包括哪些方面?
- 分类任务和回归任务有哪些常用的评价指标?

# 什么是正则化

- 正则化（Regularization）是机器学习和统计建模中的一种技术，用于防止模型在训练数据上过拟合。过拟合发生在模型过于复杂，以至于它不仅学习到了数据中的实际模式，还学习到了数据中的噪声。正则化通过引入额外的约束、惩罚或干扰，来限制模型的复杂性，从而提高模型在新数据上的泛化能力。所有对抗过拟合，干扰优化的方法都是正则化
  - 约束目标函数：在目标函数中增加模型参数的正则化项
  - 约束模型结构：对模型的结构进行约束，如Dropout
  - 数据增强：通过对样本集中的样本进行额外的操作（通常是加入随机噪声），增加样本集的数据量，提高训练模型的鲁棒性，减少过拟合的风险。
  - 约束优化过程：在优化过程中施加额外步骤干扰，如Early Stop等

# 约束目标函数方法

- L1 正则化
  - 原理：L1 正则化通过将模型的参数绝对值的和添加到损失函数中来惩罚模型的复杂性。L1正则化具有稀疏性，它在优化过程中倾向于将一些参数收缩为零，从而实现特征选择。

$$w^* = \arg\min_w \frac{1}{N}\sum_{n=1}^{N} L(y_n, f(x_n); w) + \lambda||w||_1$$

其中，$\|w\|_1 = \sum_i |w_i|$ ，$w$是参数向量

# 约束目标函数方法

- L2 正则化
  - 原理：L2 正则化通过将模型的参数平方和添加到损失函数中来惩罚模型的复杂性。它会将参数值收缩但不会将它们变为零。

$$w^* = \arg\min_w \frac{1}{N} \sum_{n=1}^{N} L(y_n, f(x_n); w) + \lambda ||w||_2^2$$

其中，$\|w\|_2^2 = \sum_i^n w_i^2$ ，$w$是参数向量

# 约束目标函数方法

- L1/L2 正则化
  - 原理：L1/L2 正则化是 L1 和 L2 正则化的结合，结合了两者的优点。它可以同时进行特征选择和参数收缩。

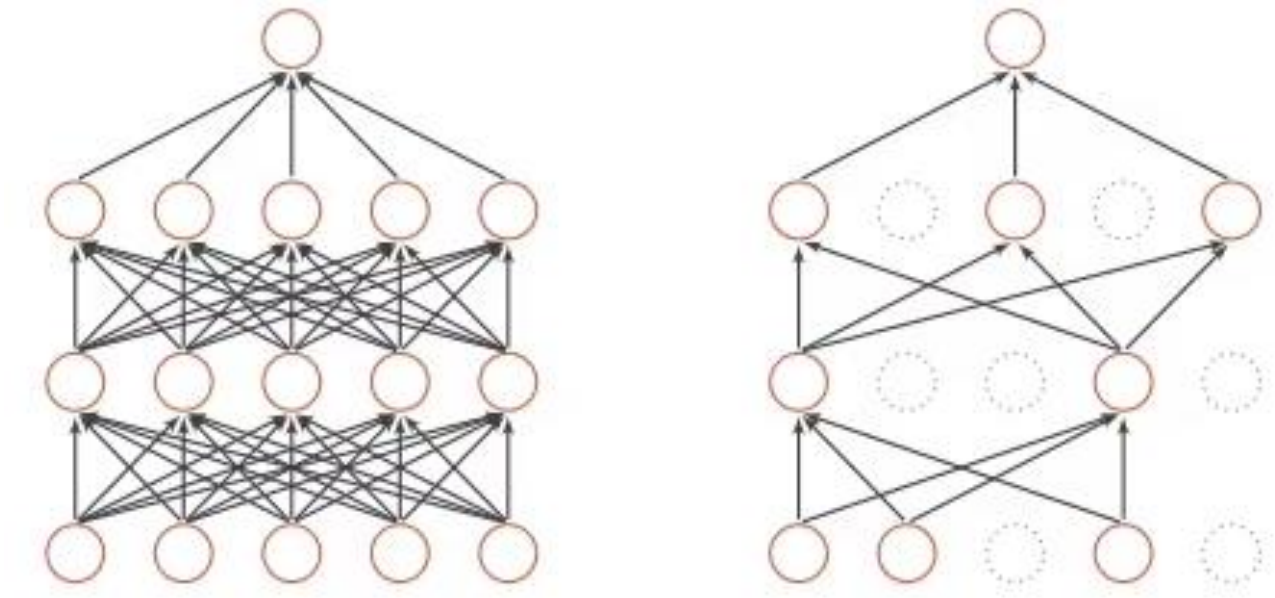
$$w^* = \arg\min_w \frac{1}{N} \sum_{n=1}^{N} L(y_n, f(x_n); w) + \lambda_2 ||w||_2^2 + \lambda_1 ||w||_1$$

- 其他方法

  谱正则化, 自正交性正则化, WEISSI正则化, 梯度惩罚等

# 约束模型结构的方法

- Dropout
  - 原理：Dropout 是一种用于神经网络的正则化方法，它在每次训练迭代中随机丢弃（即忽略）部分神经元。这样可以防止神经网络过度依赖某些特定的神经元，从而提高泛化能力



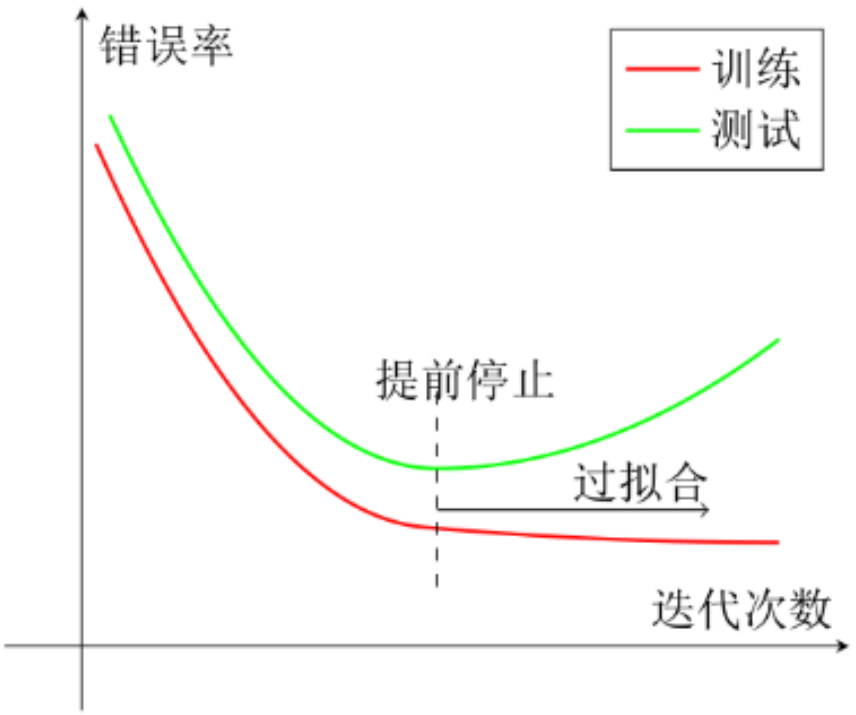
(a) 标准网络　　(b) 应用丢弃法后的网络

# 数据增强

- 图像数据的增强主要是通过算法对图像进行转变，引入噪声等方法来增加数据的多样性。
- 旋转（Rotation）：将图像按顺时针或逆时针方向随机旋转一定角度；
- 翻转（Flip）：将图像沿水平或垂直方法随机翻转一定角度；
- 缩放（Zoom In/Out）：将图像放大或缩小一定比例；
- 平移（Shift）：将图像沿水平或垂直方法平移一定步长；
- 加噪声（Noise）：加入随机噪声

# 约束优化过程的方法

- Early Stopping
  - 原理：指训练时当观察到验证集上的错误不再下降，就提前停止迭代



# 特征工程

- **没有高质量的数据，就没有高质量的挖掘结果**
  - 高质量的数据意味着我们能提取到高质量的特征。
  - 特征工程是指在机器学习过程中，从原始数据中提取、选择和转换特征的过程，其目的是提高模型的性能。
    - 特征理解
    - 特征增强
    - 特征选择与转换

# 特征工程

- **特征选择和变换**
  - 过滤方法（Filter Methods）：通过统计特征与目标变量的关系来选择特征，例如卡方检验、信息增益、相关系数等。这些方法通常不依赖于模型训练，计算效率高，但可能忽视特征之间的相互关系。
  - 包裹方法（Wrapper Methods）：将特征选择视为一个搜索问题，通过训练模型并评估模型性能来选择特征。例如，递归特征消除（RFE）就是一种包裹方法。它的优点是考虑了特征间的相互作用，但计算成本较高。
  - 嵌入方法（Embedded Methods）：将特征选择过程融入模型训练中。例如，Lasso回归（L1正则化）可以自动进行特征选择。嵌入方法结合了过滤方法和包裹方法的优点，但可能依赖于特定模型。

# 特征工程

- **特征选择和变换**
  - 主成分分析（PCA）：一种降维技术，通过线性变换将数据投影到主成分上，选择能够解释数据方差的大部分主成分。PCA并不是直接的特征选择方法，但它能通过提取重要的主成分来减少特征数量。
  - 线性判别分析（LDA）：与PCA类似，LDA用于降维，但它通过最大化类别间的分离度来选择特征，特别适用于分类问题。
  - 基于模型的方法：使用某些模型的特征重要性度量来选择特征，如随机森林、梯度提升树等。这些模型内置的特征重要性评估可以帮助选择关键特征。

# 模型的评估

- 模型评估是机器学习和数据挖掘中一个至关重要的过程，用于衡量和验证模型的性能。
  - 一方面，帮助我们从模型的假设空间中选择最佳模型（测试集）
  - 另一方面，帮助我们了解模型在未见数据上的表现，确保模型的泛化能力和有效性（验证集）
    - 交叉验证
    - 模型评价指标
    - 偏差-方差权衡（Bias-Variance Tradeoff）

# 模型的评估

- **交叉验证**
  - 主要用于防止模型过于复杂而引起的过拟合，是一种评价数据集泛化能力的统计方法。其基本思想是将原始数据进行划分，分成训练集（train_set）和测试集（test_set），训练集用来对模型进行训练，测试集用来测试训练得到的模型，以此作为模型的评价指标。
  - k折交叉验证（k-fold cross-validation）
  - 分层k折交叉验证（Stratified k-fold cross validation）
  - 留一交叉验证（Leave one out Cross-validation）
  - 打乱划分交叉验证（shuffle-split cross-validation）

# 模型的评估

## 模型评价指标

- 对于一个模型来说，如何评价该模型的好坏，针对不同的问题需要不同的模型评价标准，这是机器学习中的一个关键性的问题。具体来讲，评价指标有两个作用，其一是了解模型的泛化能力，可以通过同一个指标来对比不同模型，从而知道哪个模型相对较好；其二是可以通过这个指标来逐步优化我们的模型。因此，在选择模型与调参时，选择正确的指标是很重要的。
- 分类任务评价指标：准确率(Accuracy)、精确率(Precision)、召回率(Recall)、F1分数(F1 Score)、ROC曲线和AUC
- 回归任务评价指标：均方误差(MSE)、均方根误差(RMSE)、平均绝对误差（MAE）、决定系数(R-squared)

# 模型的评估

- **分类任务的评价**
  - **误分类：**误分类是指将被调查对象的特征错误地分到原本不属于它的类别中
  - 假阳性（false positive）属于第一类错误（type I error）
  - 假阴性（false negative）属于第二类错误（type II error）
  - 所有的分类器都存在偏好，因此都存在误分类的现象
  - 一个好的模型应该尽量减少第一类错误和第二类错误。可以通过调整分类器的阈值来平衡这些不同类型的错误，如何平衡和优化这两种错误取决于具体应用的需求和场景，这样我们的模型才有实际的应用价值

# 模型的评估

## 分类任务的评价

- **混淆矩阵（confusionmatrix）：**用于评估分类模型性能的工具，用n行n列的矩阵形式来表示。混淆矩阵是总结分类模型预测结果的情形分析表，以矩阵形式将数据集中的记录按照真实的类别与分类模型预测的类别进行汇总。其中矩阵的行表示真实值，矩阵的列表示预测值。下图以二分类问题为例展示混淆矩阵

| | 预测为正类 (Positive) | 预测为负类 (Negative) |
|---|---|---|
| 实际为正类 (Positive) | 真阳性 (True Positive, TP) | 假阴性 (False Negative, FN) |
| 实际为负类 (Negative) | 假阳性 (False Positive, FP) | 真阴性 (True Negative, TN) |

真阳性 (TP)：实际为正类的样本被正确预测为正类。
假阴性 (FN)：实际为正类的样本被错误预测为负类。
假阳性 (FP)：实际为负类的样本被错误预测为正类。
真阴性 (TN)：实际为负类的样本被正确预测为负类。

# 模型的评估

- **分类任务的评价**

| | 预测为正类 (Positive) | 预测为负类 (Negative) |
|---|---|---|
| 实际为正类 (Positive) | 真阳性 (True Positive, TP) | 假阴性 (False Negative, FN) |
| 实际为负类 (Negative) | 假阳性 (False Positive, FP) | 真阴性 (True Negative, TN) |

- 准确率 (Accuracy)：所有预测中正确预测的比例

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

- 精确率 (Precision)：预测为正类的样本中，实际为正类的比例，也叫查准率

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

# 模型的评估

- 分类任务的评价

| | 预测为正类 (Positive) | 预测为负类 (Negative) |
|---|---|---|
| 实际为正类 (Positive) | 真阳性 (True Positive, TP) | 假阴性 (False Negative, FN) |
| 实际为负类 (Negative) | 假阳性 (False Positive, FP) | 真阴性 (True Negative, TN) |

  - 召回率 (Recall)：所有实际为正类的样本中，被正确预测为正类的比例。也叫敏感性（sensitivity），又叫查全率。顾名思义，“查全”表明预测为真覆盖到了多少实际为真的样本，换句话说遗漏了多少

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

# 模型的评估

## 分类任务的评价

| | 预测为正类 (Positive) | 预测为负类 (Negative) |
|---|---|---|
| 实际为正类 (Positive) | 真阳性 (True Positive, TP) | 假阴性 (False Negative, FN) |
| 实际为负类 (Negative) | 假阳性 (False Positive, FP) | 真阴性 (True Negative, TN) |

- F1分数 (F1 Score)：精确率和召回率的调和平均数，是用来衡量分类模型综合性能的一种指标。它同时兼顾了分类模型的准确率和召回率。它的最大值是1，最小值是0。

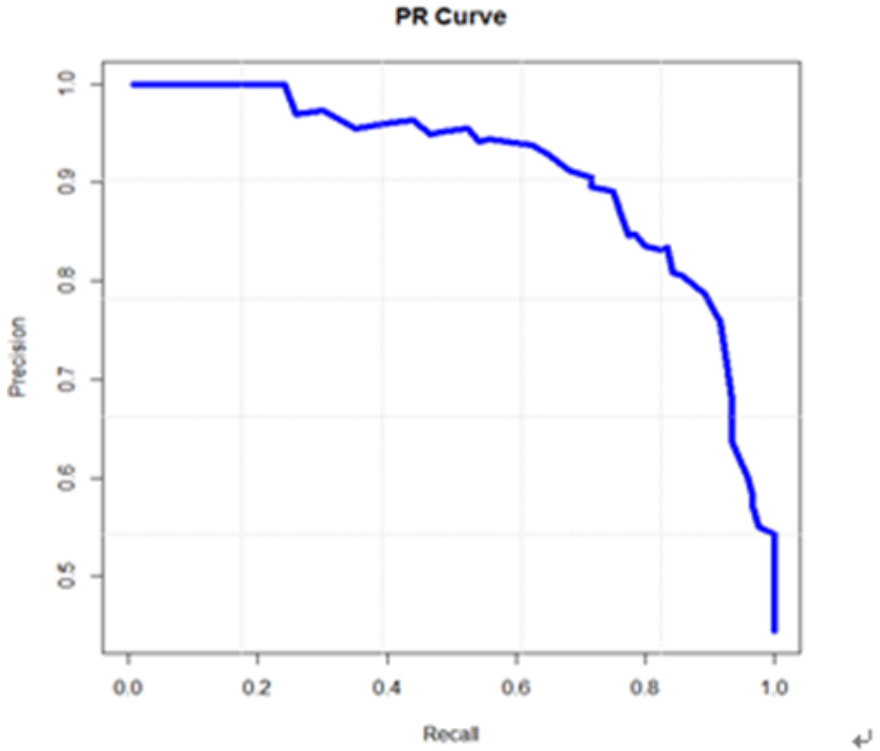
$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

- 调和平均数的性质是，当精确率和召回率二者都非常高的时候，它们的调和平均才会高。如果其中之一很低，调和平均就会被拉得接近于很低的那个数

# 模型的评估

## 分类任务的评价

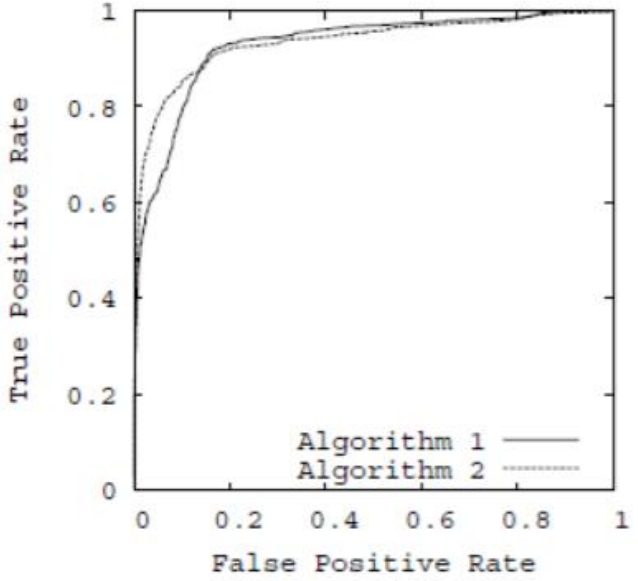
- 精确率-召回率曲线（P-R曲线）：也叫查准率-查全率曲线，是一对相互矛盾的性能指标，而事实上我们所期望的是既能保证查准率，又能提升查全率。对分类问题来讲，通过不断调整分类器的阈值，可以得到不同的Precision-Recall值，遍历所有可能的阈值，从而可以得到一条曲线。通常随着分类阈值从大到小变化，查准率减小，而查全率增加。



# 模型的评估

## 分类任务的评价

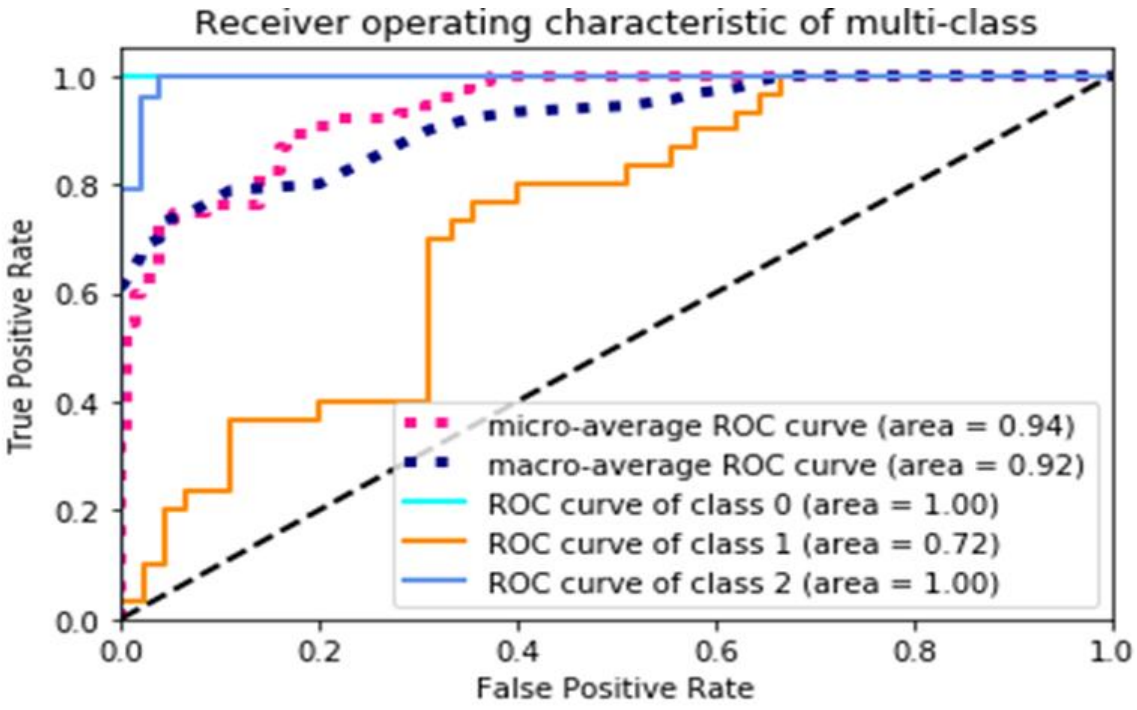
- ROC曲线:受试者工作特征曲线(receiver operator characteristic curve, ROC曲线),通常被用来评价一个二值分类器的优劣。ROC曲线的横坐标是假阳性率(false positive rate，FPR)，纵坐标是真阳性率(true positive rate，TPR)。
- TPR表示在所有实际为阳性的样本中，被正确地判断为阳性的比率，即TPR=TP/(TP+FN)。
- FPR表示在所有实际为阴性的样本中，被错误地判断为阳性之比率，即FPR=FP/(FP+TN)。



# 模型的评估

## 分类任务的评价

- TPR越高，FPR越低，则可以证明分类器分类效果越好。但是两者又是相互矛盾的，所以单凭TPR和FPR的两个值是没有办法比较两个分类器的好坏的，因此在机器学习里提出了ROC曲线。
- 也就是画出来的ROC曲线越靠近左上越好



# 模型的评估

## 分类任务的评价

- AUC（Area Under roc Curve）:AUC值为ROC曲线所覆盖的区域面积,是一种用来度量分类模型好坏的一个标准，显然，AUC越大，分类器分类效果越好。
- AUC = 1，是完美分类器，采用这个预测模型时，不管设定什么阈值都能得出完美预测。但绝大多数预测的场合，不存在完美分类器。
- 0.5 < AUC < 1，优于随机猜测，若妥善设定阈值，分类器将具有预测价值。
- AUC = 0.5，跟随机猜测一样，模型没有预测价值。
- AUC < 0.5，比随机猜测还差，但只要总是反预测而行，就优于随机猜测。

# 模型的评估

- 回归任务的评价
  - **均方误差**（Mean squared error，MSE），是反映估计量与被估计量之间差异程度的一种度量，其值越小说明拟合效果越好，所以常被用作线性回归的损失函数。

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

  - **均方根误差**（Root Mean Squared Error, RMSE）：MSE的平方根。
  - **平均绝对误差**（Mean absolute Error，MAE），预测目标值和实际目标值之间误差的绝对值的平均数，可以更好地反映预测值误差的实际情况，其值越小越好。

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

# 模型的评估

- 回归任务的评价
  - **中位绝对误差**（Median absolute error，MedAE）通过取目标值和预测值之间的所有绝对差值的中值来计算损失，其值越小越好

$$MedAE(y,\hat{y}) = median(|y_1 - \hat{y}_1|,...,|y_n - \hat{y}_n|)$$

  - R2决定系数（R Squared）表征回归方程在多大程度上解释了因变量的变化，或者说方程对观测值的拟合程度如何。R2决定系数的最优值为1（完全拟合），为0时，说明模型和样本基本没有关系，也可为负，为负时说明模型非常差

$$R^2(y,\hat{y}) = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

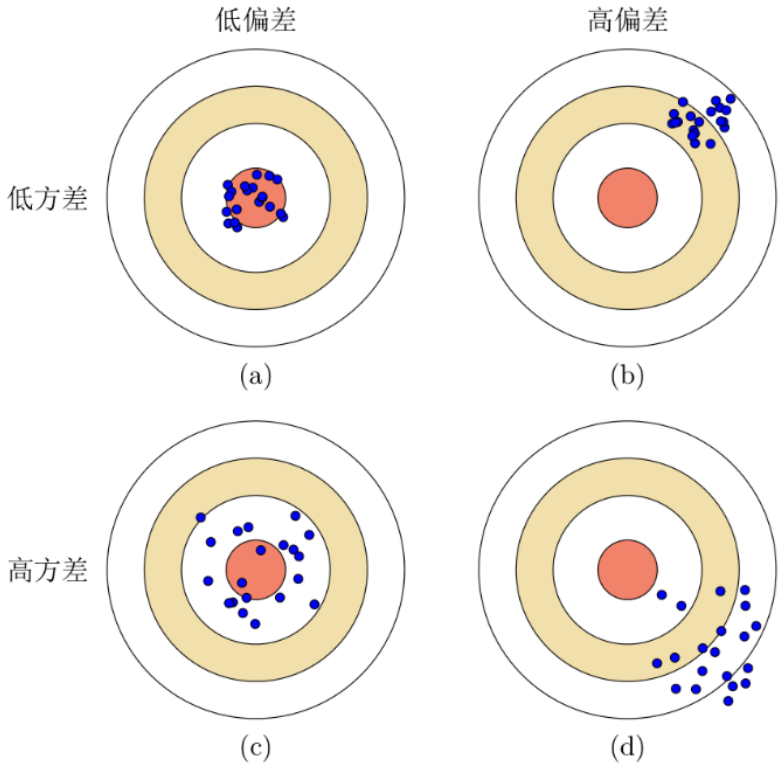
# 模型的评估

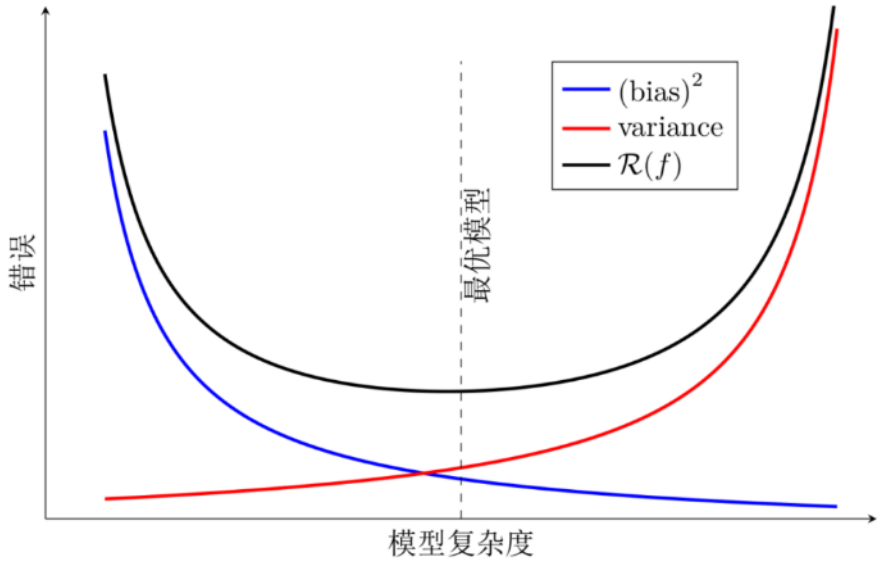
## 偏差和方差的权衡

- 模型的评估过程中还要考虑偏差（模型对训练数据的拟合程度）和方差（模型对训练数据的小波动的敏感程度）。一个理想的模型应该在这两者之间找到平衡，既能在训练数据上表现良好，也能在测试数据上保持较好的性能。
- 偏差（Bias）：模型预测的期望值与真实值之间的差距。高偏差通常表示模型过于简单，无法捕捉数据的复杂模式，导致系统性误差。偏差反映了模型的系统性错误。。
- 方差（Variance）：模型预测对训练数据的波动敏感性。高方差通常表示模型过于复杂，对训练数据的噪声和细节过度拟合，导致模型在不同数据集上的表现不稳定。方差反映了模型的随机误差。
- 噪声（Irreducible Error）：无法通过任何模型减少的误差，通常由数据本身的随机性或测量误差造成。

# 模型的评估

- 偏差和方差的权衡



# 模型的优化

- **梯度下降法**
  - 如何确定坡度最大的方向？
  - 在微积分里面，对多元函数的参数求偏导数，把求得的各个参数的偏导数以向量的形式写出来，就是梯度。比如函数f(x,y), 分别对x,y求偏导数，求得的梯度向量就是(∂f/∂x, ∂f/∂y)T,简称grad f(x,y)或者▽f(x,y)。

$$\text{gradf}(x, y) = \nabla f(x, y) = \left\{ \frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right\}$$

  - 从几何意义上讲，梯度就是函数变化最快的地方。具体来说，对于函数f(x,y),在点(x0,y0)，沿着梯度向量的方向就是(∂f/∂x0, ∂f/∂y0)T的方向是f(x,y)增加最快的地方。或者说，沿着梯度向量的方向，更加容易找到函数的最大值。反过来说，沿着梯度向量相反的方向，也就是 -(∂f/∂x0, ∂f/∂y0)T的方向，梯度减少最快，也就是更加容易找到函数的最小值

# 模型的优化

- **梯度下降法**

3. 算法过程：

1）确定当前位置的损失函数的梯度，对于$\theta$向量,其梯度表达式如下：

$$\frac{\partial}{\partial\theta}J(\theta)$$

2）用步长乘以损失函数的梯度，得到当前位置下降的距离，即$\alpha\frac{\partial}{\partial\theta}J(\theta)$对应于前面登山例子中的某一步。

3）确定$\theta$向量里面的每个值,梯度下降的距离都小于$\varepsilon$，如果小于$\varepsilon$则算法终止，当前$\theta$向量即为最终结果。否则进入步骤4.

4）更新$\theta$向量，其更新表达式如下。更新完毕后继续转入步骤1.

$$\theta=\theta-\alpha\frac{\partial}{\partial\theta}J(\theta)$$

新参数 = 原参数 - 学习率 x 梯度

较高的学习率可以使模型快速收敛，但也可能导致过度调整甚至发散(不收敛)。
较低的学习率虽然稳定，但收敛速度慢，可能需要更多的训练时间和迭代次数。
因此实际应用中需要二者的折中，设置最佳的学习率。

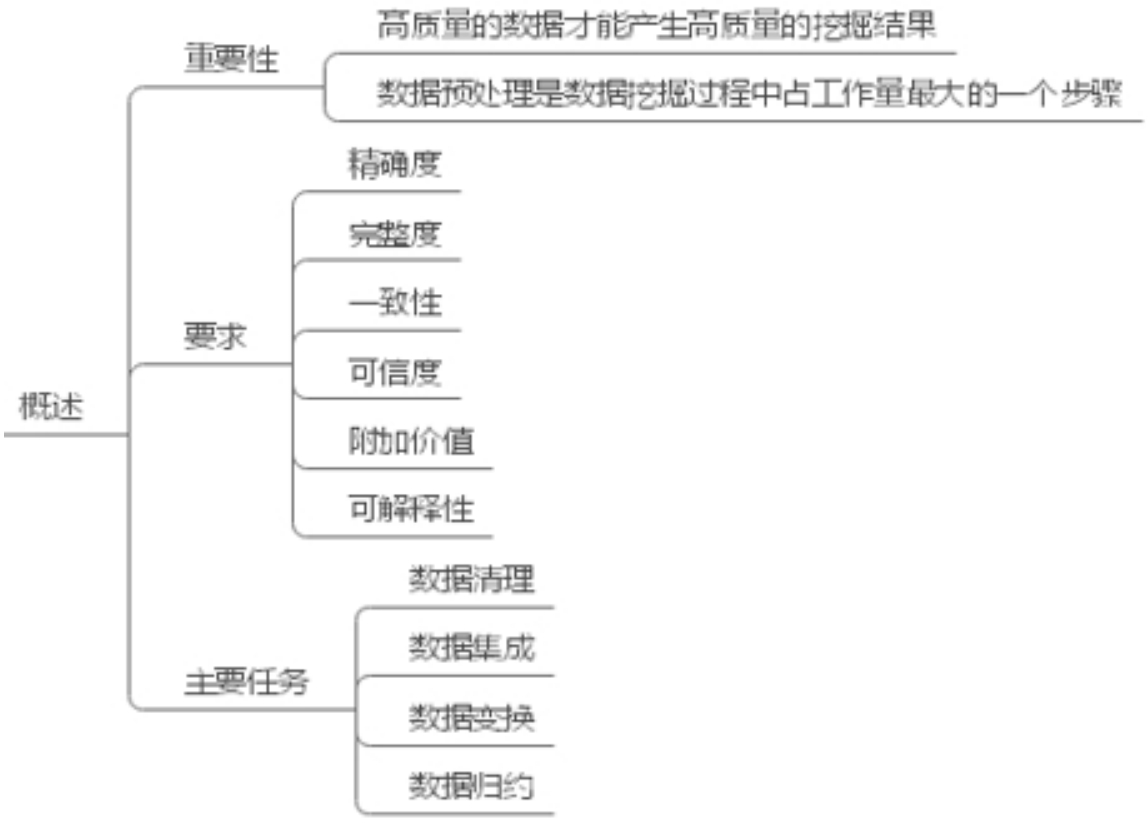
# 超参数（Hyperparameter）

- 什么是超参数
  - 超参数是用于控制机器学习算法学习过程的参数，与模型参数不同，模型参数是在训练过程中通过数据学习得到的，而超参数是在训练之前设置的，并且在训练过程中保持不变。
- 常见的超参数举例
  - 学习率（Learning Rate）：控制模型参数更新的步长
  - 正则化参数（Regularization Parameter）：控制模型复杂度，防止过拟合。
  - 神经网络结构（Neural Network Architecture）：例如层数、每层的神经元数量等。
  - 数据集的划分比例等

# 第二章 数据预处理

- **为什么对数据进行预处理**
- 数据的组织形式和属性
- 数据质量
- 数据清理
- 数据集成和变换
- 数据归约
- 如何操作？

# 第二章 数据预处理


概述
重要性
高质量的数据才能产生高质量的挖掘结果
数据预处理是数据挖掘过程中占工作量最大的一个步骤
要求
精确度
完整度
一致性
可信度
附加价值
可解释性
主要任务
数据清理
数据集成
数据变换
数据归约

# 第二章 数据预处理

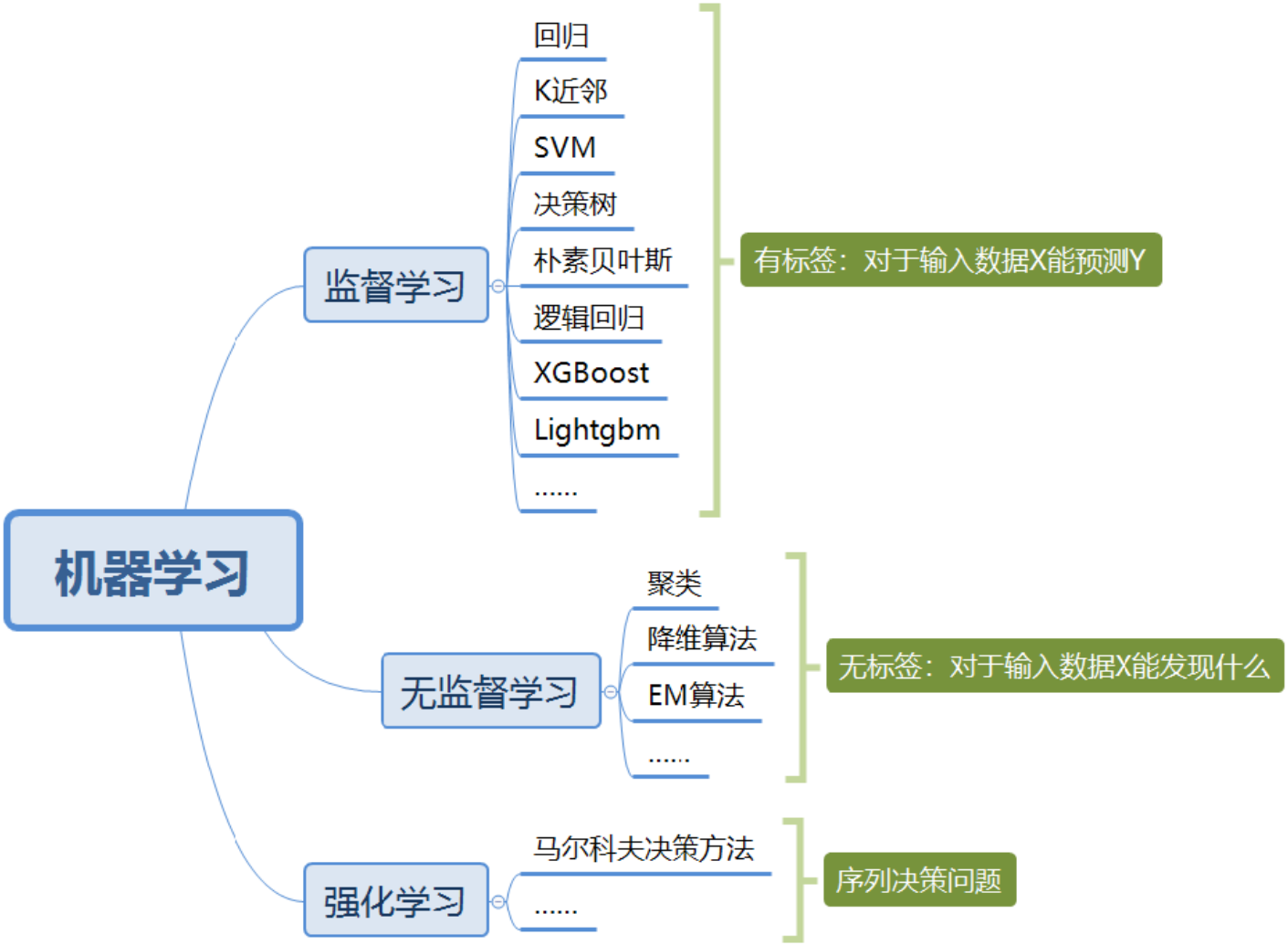

数据的统计描述
度量数据的中心趋势
平均值
中位数
中列数
众数
对称数据
倾斜数据
正倾斜
负倾斜
度量数据的离散趋势
极差
四分位数
IQR = Q3-Q1
离群点
挑出至少高于第三个四分位数或低于第一个四分位数1.5*IQR处的值
盒图
方差和标准差
相关的图形显示

# 第二章 数据预处理


数据清理
填写空缺的值
识别离群点和平滑噪声数据
有个分箱的概念
纠正不一致的数据
解决数据集成造成的冗余
数据集成和变换
数据集成
合并多个数据原中的数据，存放在一个一致的数据存储中，这当中要做到一致性
数据变换
将数据转换成适合挖掘的统一形式
最小-最大规范化
z-score规范化
最大最小值不知道时适用
数据归约
常用数据归约策略
维归约：移除不重要的属性
主成分分析PCA等等
数据压缩
有损压缩
无损压缩
数据归约
使用模型来表示数据，通过选择替代的或者较小的数据表示形式来减少数据量。
有参方法
用参数模型估计，最后只用存储参数
线性回归
多元回归
无参方法
直方图
聚类
采样
用较少的随机样本来表示原始数据集

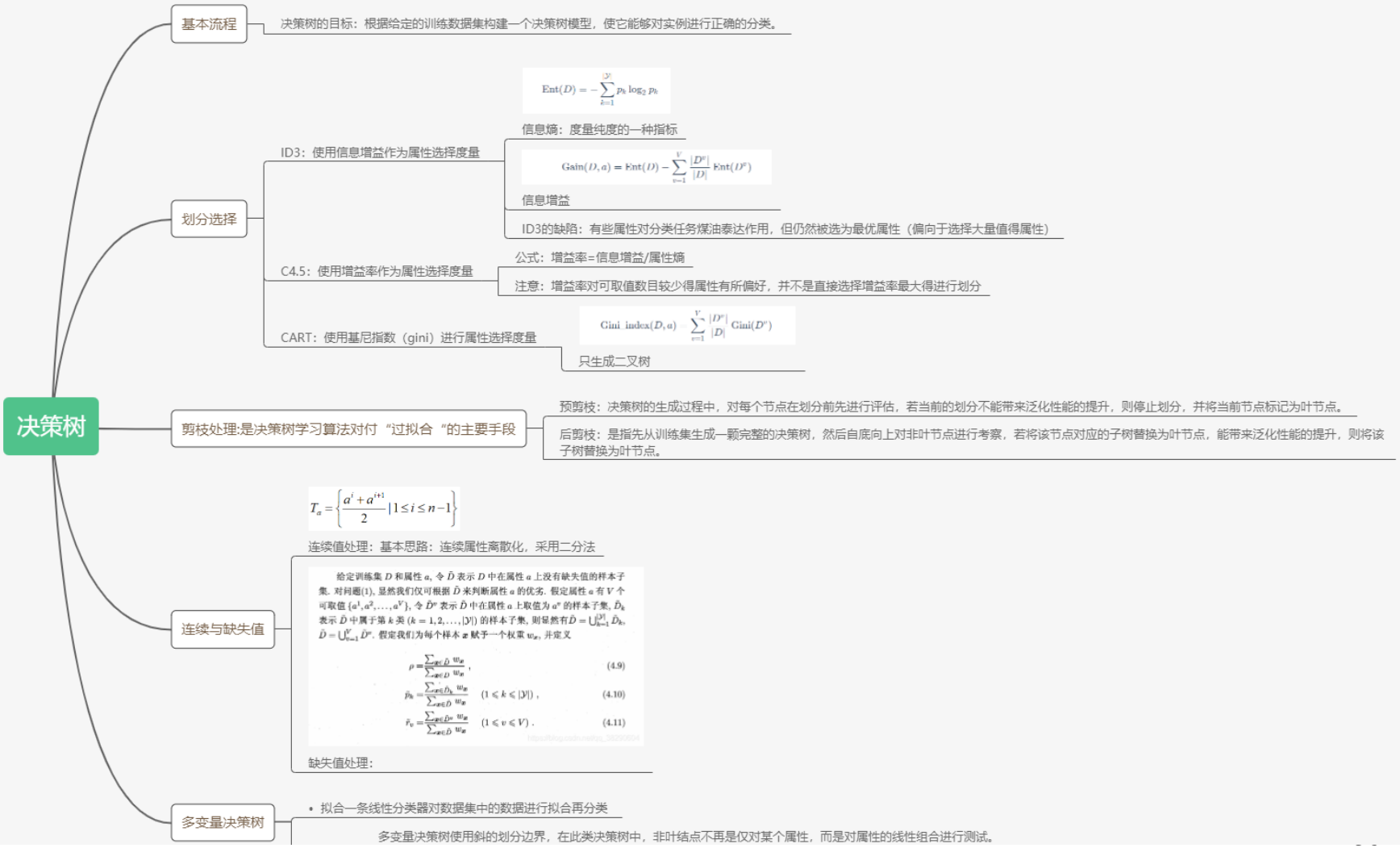
# 第三章 分类与预测

- 机器学习的算法概览



# 第三章 分类与预测

- 决策树



# 决策树

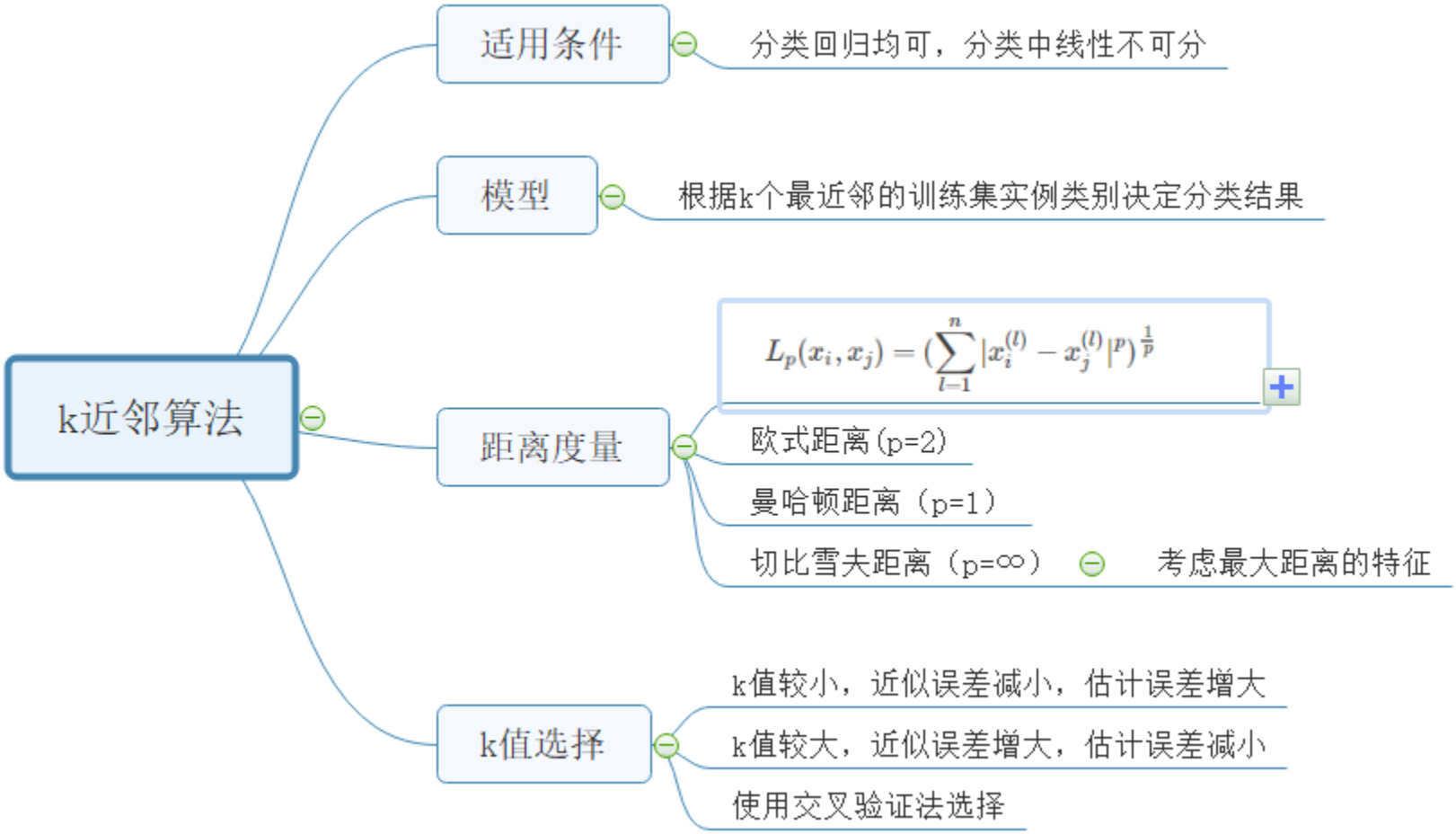
- 决策树算法：
  - 决策树算法是一种归纳分类算法，它通过对训练集的学习，挖掘出有用的规则，用于对新数据进行预测。
  - 决策树算法属于监督学习方法。
  - 决策树归纳的基本算法是贪心算法，自顶向下来构建决策树。
  - 贪心算法：在每一步选择中都采取在当前状态下最好优的选择。
  - 在决策树的生成过程中，分割方法即属性选择的度量是关键。

# 决策树

- 决策树算法关注的主要问题：
  - 特征选择（选择哪个属性作为分类依据）
    - 信息增益
    - 信息增益比
    - 基尼指数/平方误差
  - 决策树的生成算法
    - ID3算法
    - C4.5算法
    - CART算法
  - 决策树的剪枝策略：决策树的贪心特性容易生成过多的分枝造成过拟合，需要对其进行控制，以提高对未知数据分类的准确性。
    - 预剪枝方法
    - 后剪枝方法

# 第三章 分类与预测

- KNN



# KNN

- k近邻（k Nearest Neighbor, kNN）的核心思想
  - 对于分类问题：对新的样本，根据其$k$个最近邻的训练样本的类别，通过多数表决等方式进行预测。
  - 对于回归问题：对新的样本，根据其$k$个最近邻的训练样本标签值的均值作为预测值。
  - $k$近邻法的三要素：

    •距离度量

    • $k$值选择

    •决策规则

# KNN

- 距离的度量
  - 距离度量，简而言之，是一种衡量数据集中元素之间关系(相似性/差异性）的方法。
  - 它通过定义距离函数来实现，这个函数为数据集中的每个元素提供了一种相互关系的度量
  - 距离函数，本质上，是一种数学工具，它帮助我们量化数据集中任意两个元素之间的差异
  - 不同的距离度量采用不同的数学公式作为其距离函数
  - 如果两个元素之间的距离为零，可以认为它们是等同的
  - 如果距离大于零，则它们有所不同

# 第三章 分类与预测

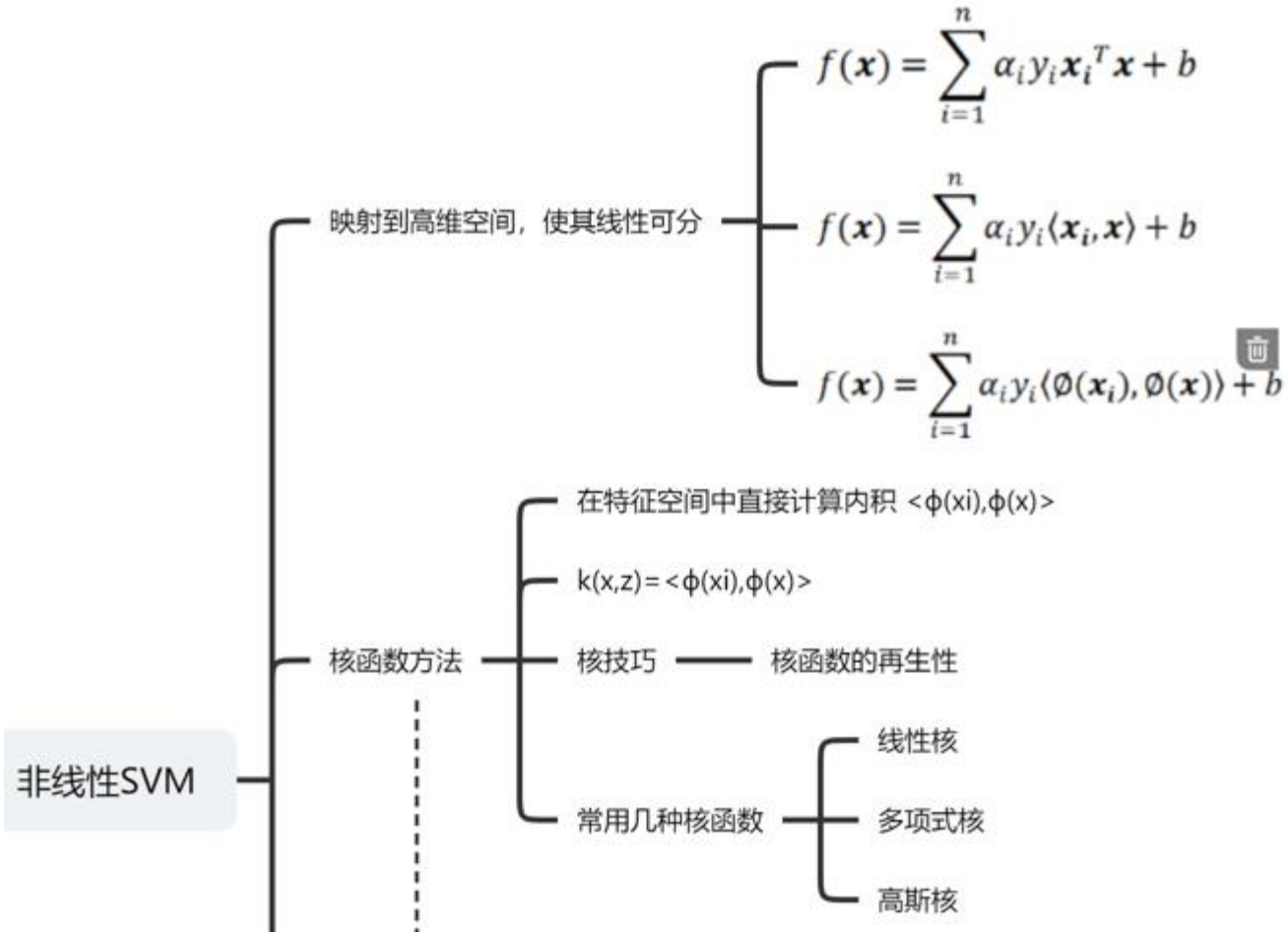
- SVM

# 第三章 分类与预测

- SVM

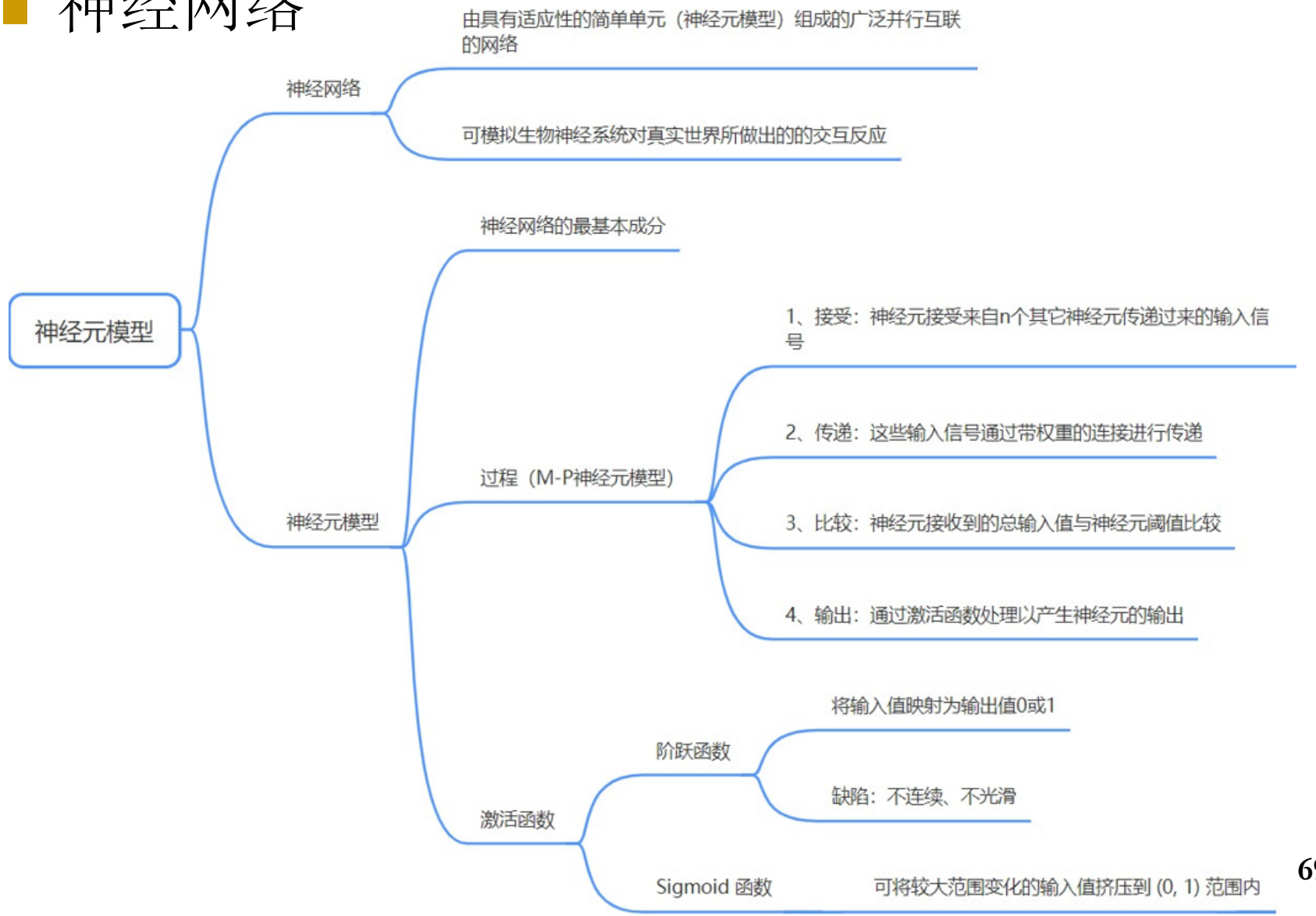
# 第三章 分类与预测

- SVM



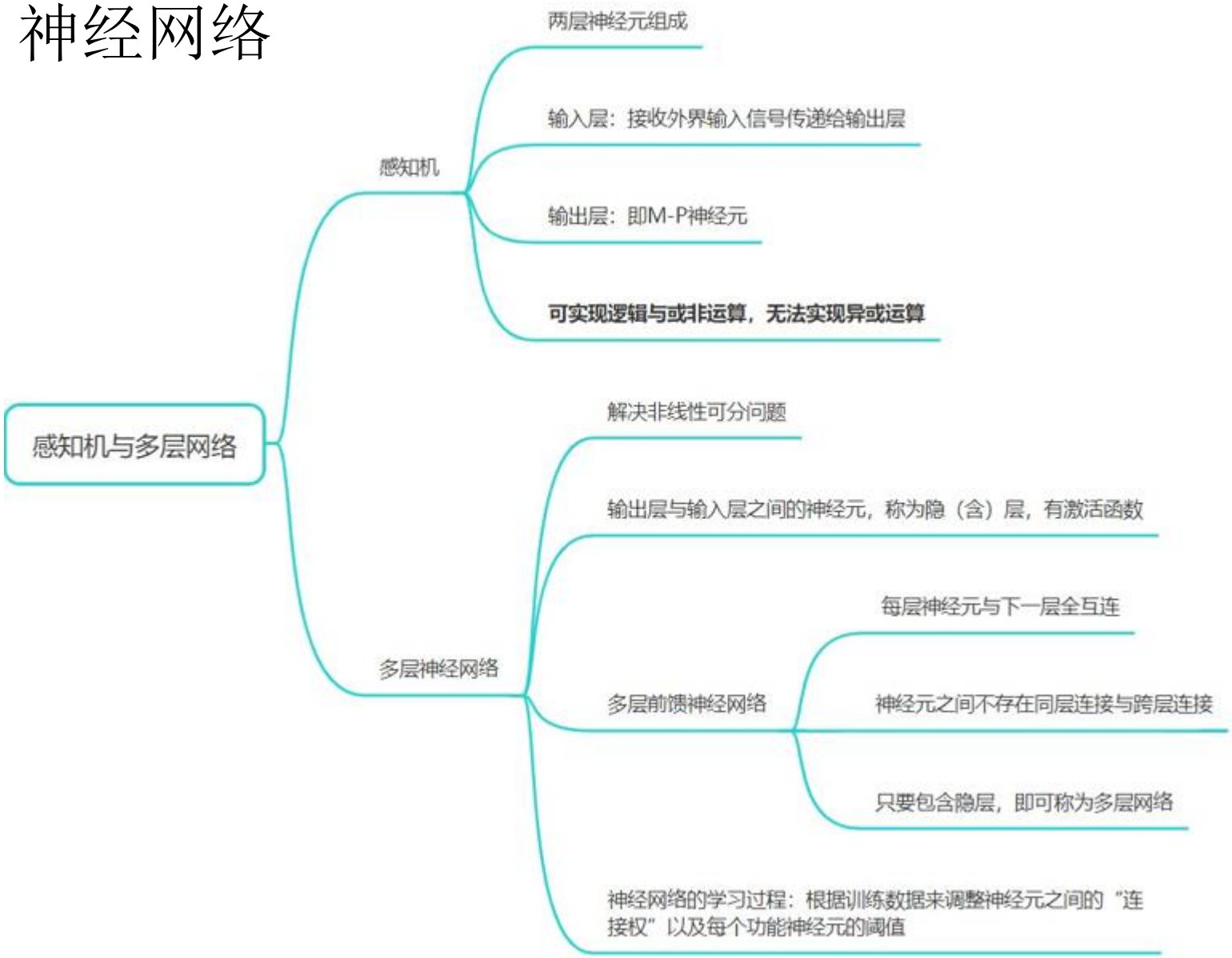
# 第三章 分类与预测

- 神经网络



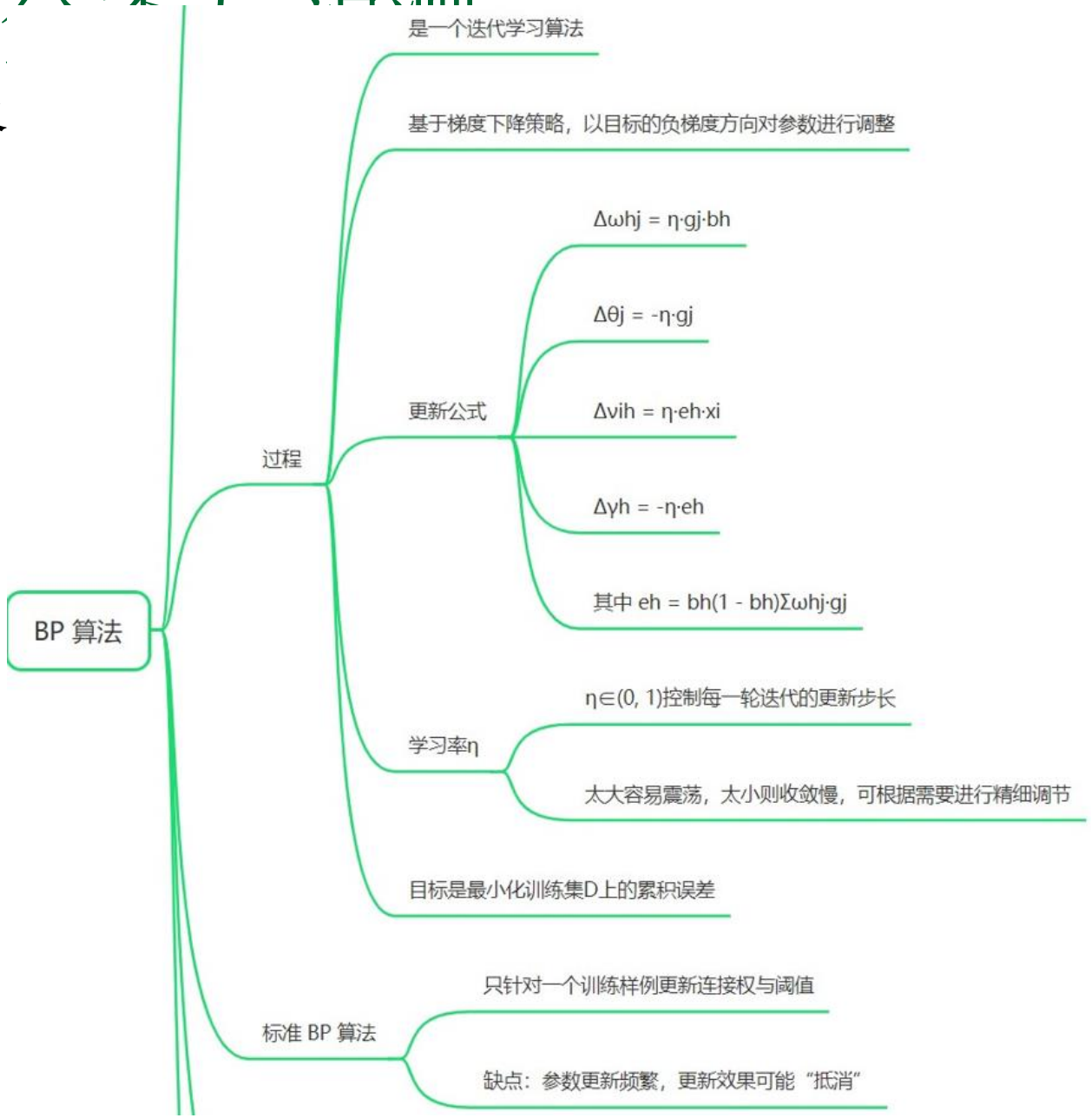
# 第三章 分类与预测

- 神经网络



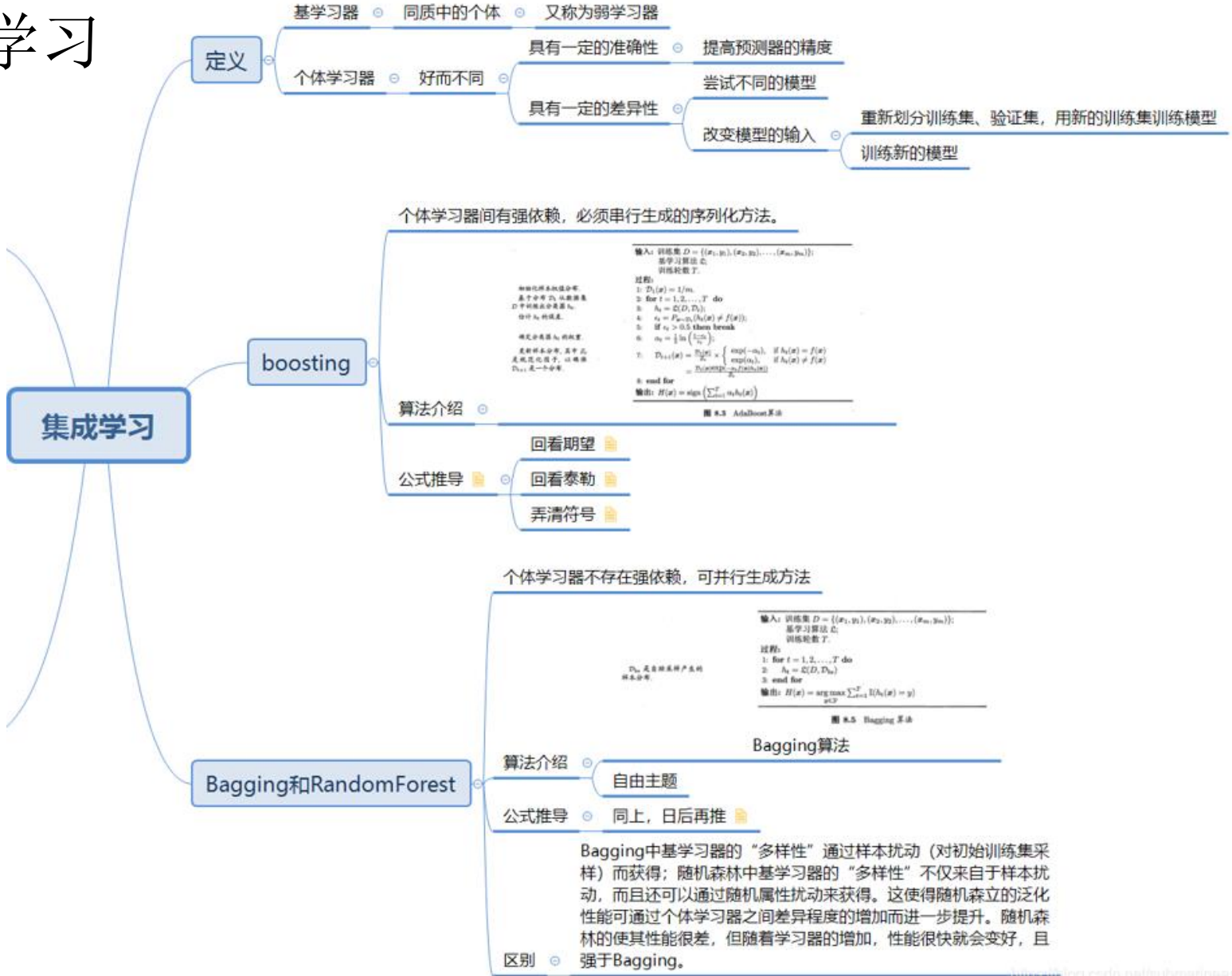
# 第三章

- 神经网络


BP 算法
过程
是一个迭代学习算法
基于梯度下降策略，以目标的负梯度方向对参数进行调整
更新公式
Δωhj = η·gj·bh
Δθj = -η·gj
Δvih = η·eh·xi
Δγh = -η·eh
其中 eh = bh(1 - bh)Σωhj·gj
学习率η
η∈(0, 1)控制每一轮迭代的更新步长
太大容易震荡，太小则收敛慢，可根据需要进行精细调节
目标是最小化训练集D上的累积误差
标准 BP 算法
只针对一个训练样例更新连接权与阈值
缺点：参数更新频繁，更新效果可能“抵消”

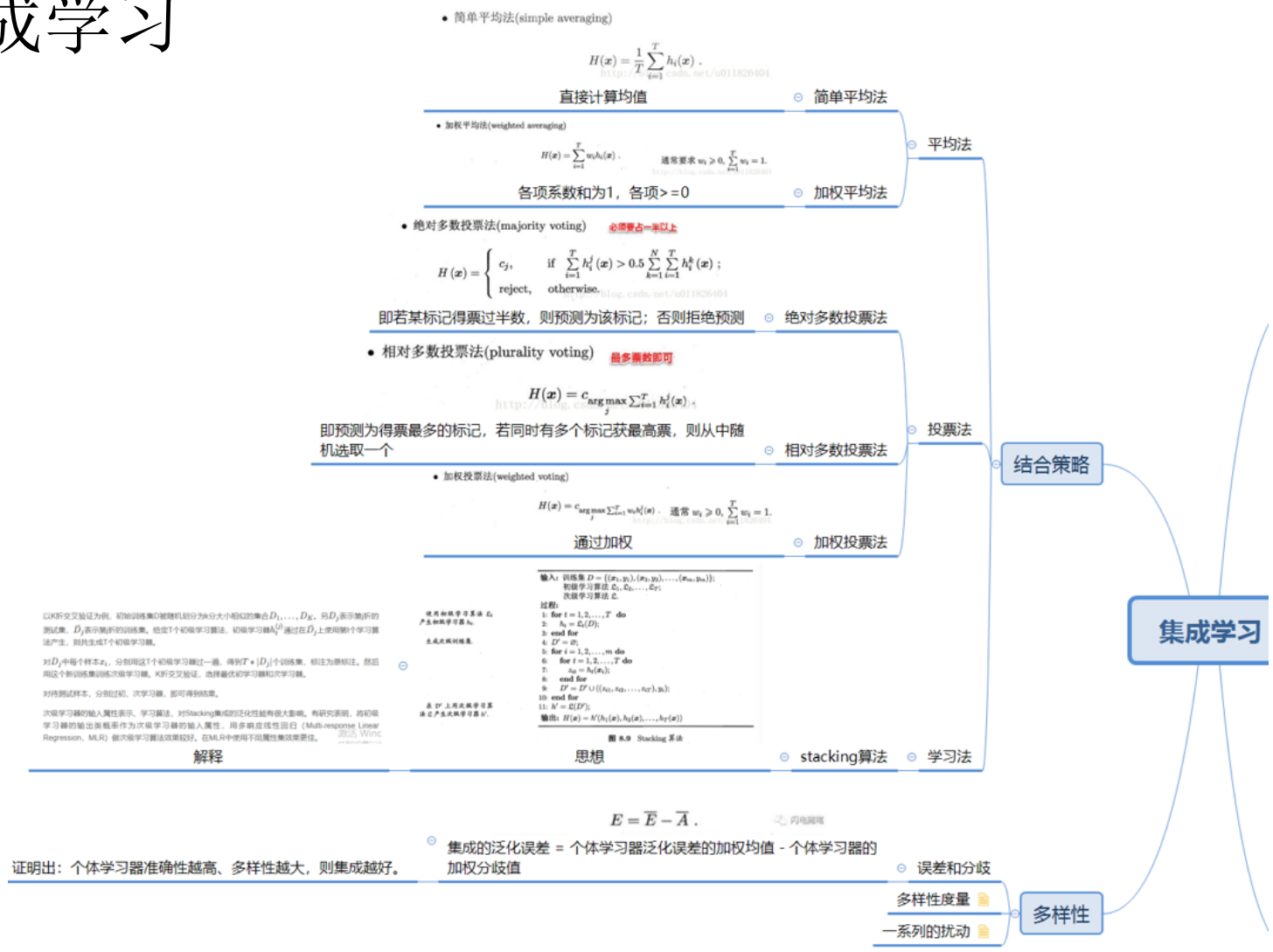
# 第三章 人类与预测

- 集成学习



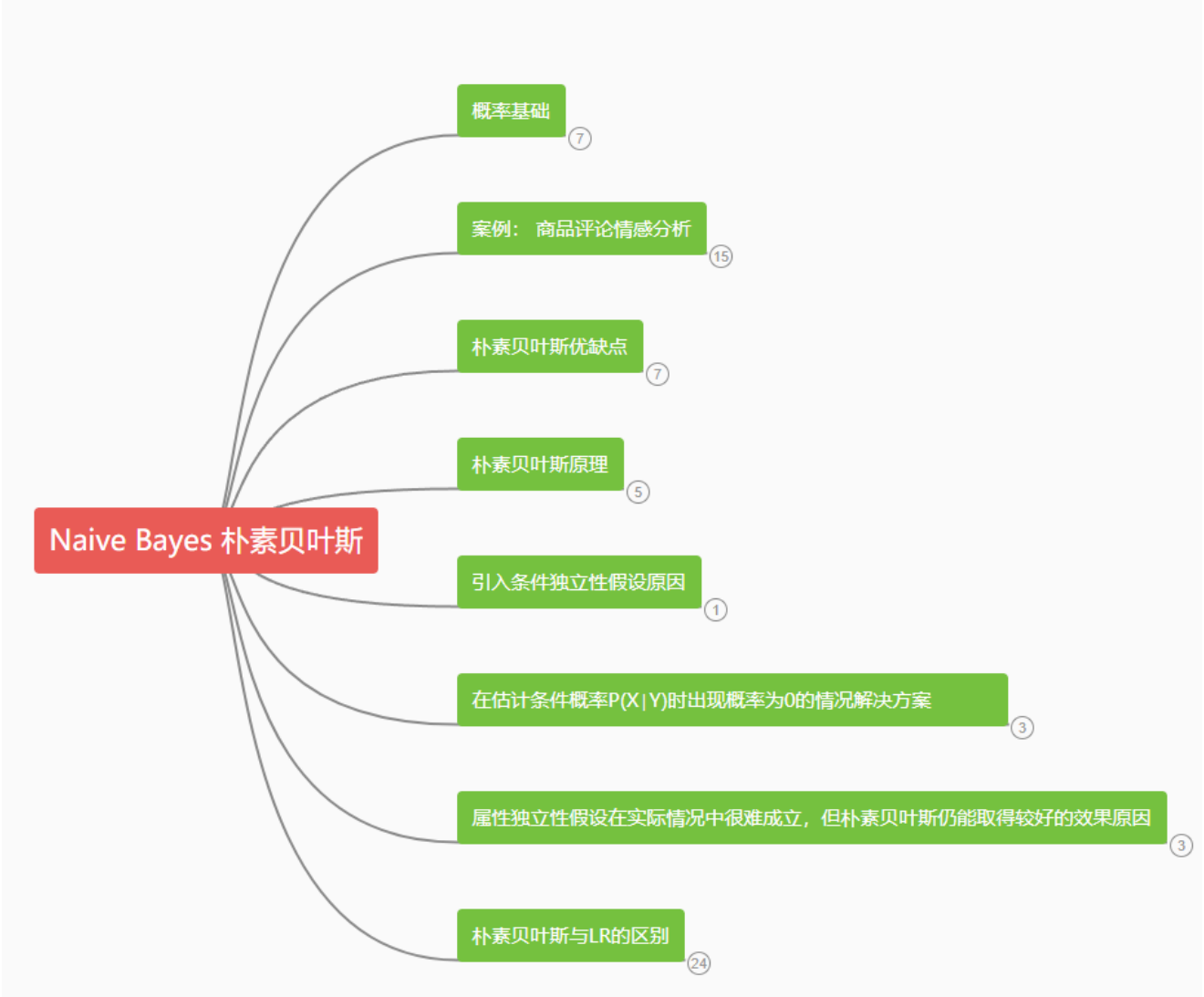
# 第三章 分类与预测

- 集成学习



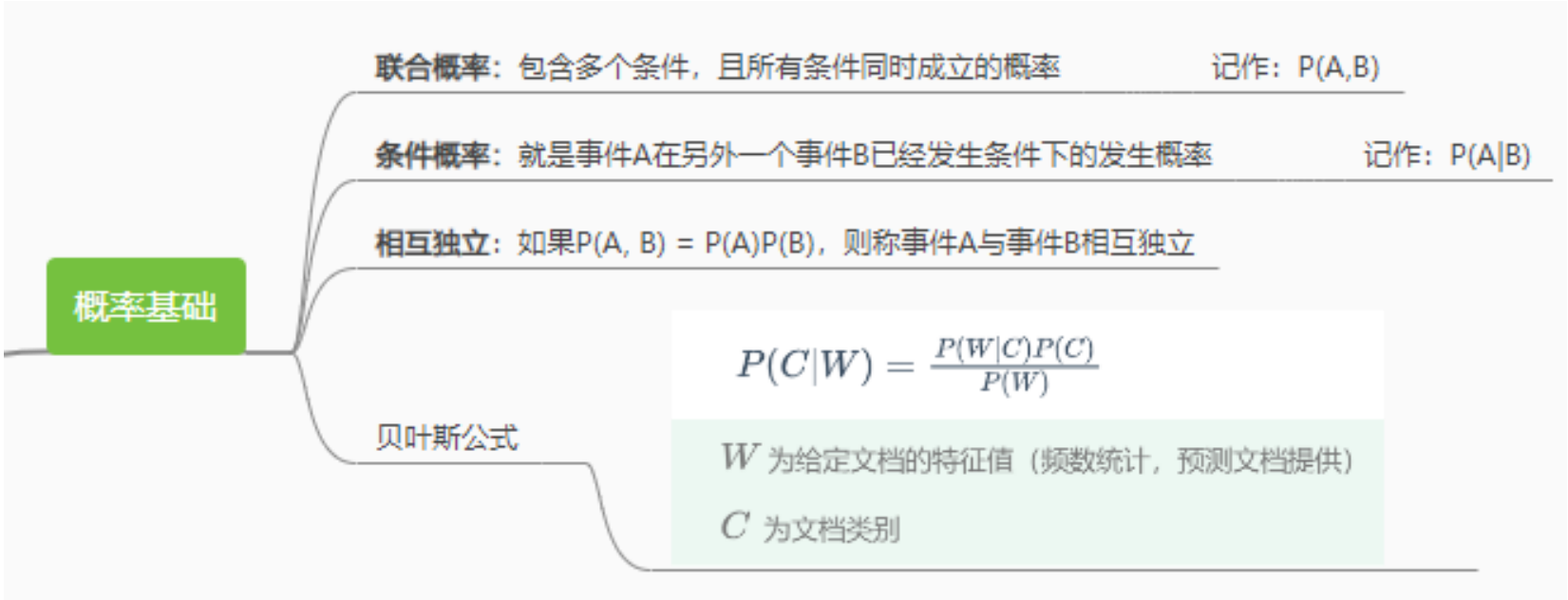
# 第三章 分类与预测

- 贝叶斯分类



# 第三章 分类与预测

- 贝叶斯分类



# 第三章 分类与预测

- 贝叶斯分类

**朴素贝叶斯优缺点**

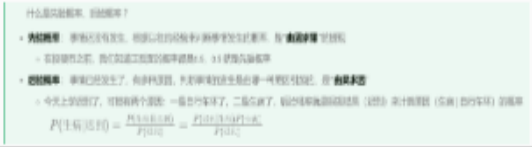
- 优点
  - 朴素贝叶斯模型发源于古典数学理论，**有稳定的分类效率**
  - **对缺失数据不太敏感**，算法也比较简单，常用于文本分类
  - 分类准确度高，速度快
- 缺点
  - 由于**使用了样本属性独立性的假设**，所以如果特征属性有关联时其效果不好
  - **需要计算先验概率**，而先验概率很多时候取决于假设，假设的模型可以有很多种，因此在某些时候会由于假设的先验模型的原因导致预测效果不佳；

**朴素贝叶斯原理**

- 基于贝叶斯定理与特征条件独立假设的分类方法
- 对于给定的待分类项 $x$，通过学习到的模型计算后验概率分布

在此项出现的条件下各个目标类别出现的概率，将后验概率最大的类作为x所属的类别

- 特点　当Y确定时，X的各个特征分量取值之间相互独立



# 第三章 分类与预测

- 贝叶斯分类