第一章 人工智能概览

人工智能概述

1950年，艾伦·图灵提出了“机器能否思考”的问题，并提出了图灵测试，即通过对话来判断机器是否具有智能。

1956年，约翰·麦卡锡在达特茅斯会议上提出“人工智能”这个概念。

多元智能的八种能力：

- 语言
- 逻辑推理
- 视觉空间 对空间位置的感受
- 肢体动觉
- 音乐
- 人际 与他人沟通时的反应
- 自我认知 认识自我的优缺点
- 自然 观察自然的各种形态

人工智能的定义分为两部分：

- 人工：由人设计，为人创造，服务人类
- 智能：像人一样行为

1950年人工智能

- 研究、开发用于模拟人的智能的理论、方法、应用系统的技术科学

1980年机器学习

- 基于样本数据构建模型，做出预测或决策
- 是实现人工智能的主要路径

2010年深度学习

- 机器学习的一个领域，源于人工神经网络的研究
- 基于人工神经网络模型模拟人脑处理信息的方式

人工智能主要学派

- 符号主义/逻辑主义/心理学派/计算机学派
    - 人工智能源于数理逻辑，人类认知的过程是各种符号进行推理运算的过程
- 连接主义/仿生学派/生理学派
    - 人工智能源于仿生学，人类的思维基于神经元，而不是符号处理过程

- 行为主义/进化主义/控制论学派
  - 人工智能源于控制论，智能取决于感知和行动，不需要知识、表示、推理

人工智能发展简史

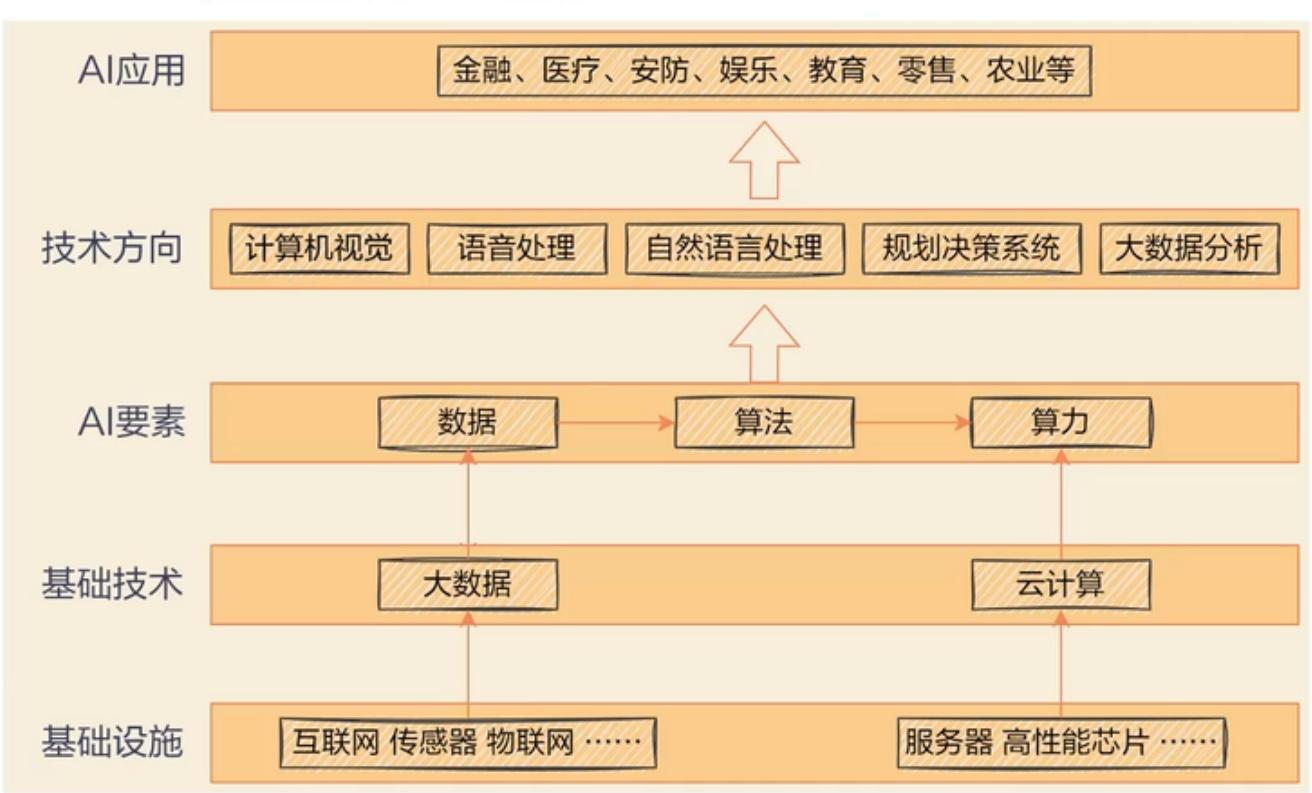
- 1956：达特茅斯会议提出人工智能
- 1959：阿瑟·萨缪尔提出机器学习
- 1985：决策树模型和多层人工神经网络
- 2006：Hinton开始研究深度学习
- 2010：大数据时代
- 2014：微软Cortana 第一个第一款个人智能助理
- 2016：AlphaGo战胜李世石
- 2018：基于Transformer的Bert模型
- 2020：OpenAI发布GPT-3
- 2022：Google发布扩散模型

人工智能的分类

- 强人工智能
  - 真正能推理和解决问题，具有自我意识，能理解、学习、创造
- 弱人工智能
  - 只能模拟人类的智能，不能真正理解、学习、创造，没有自主意识

人工智能的产业生态

人工智能四要素：数据、算法、算力、场景

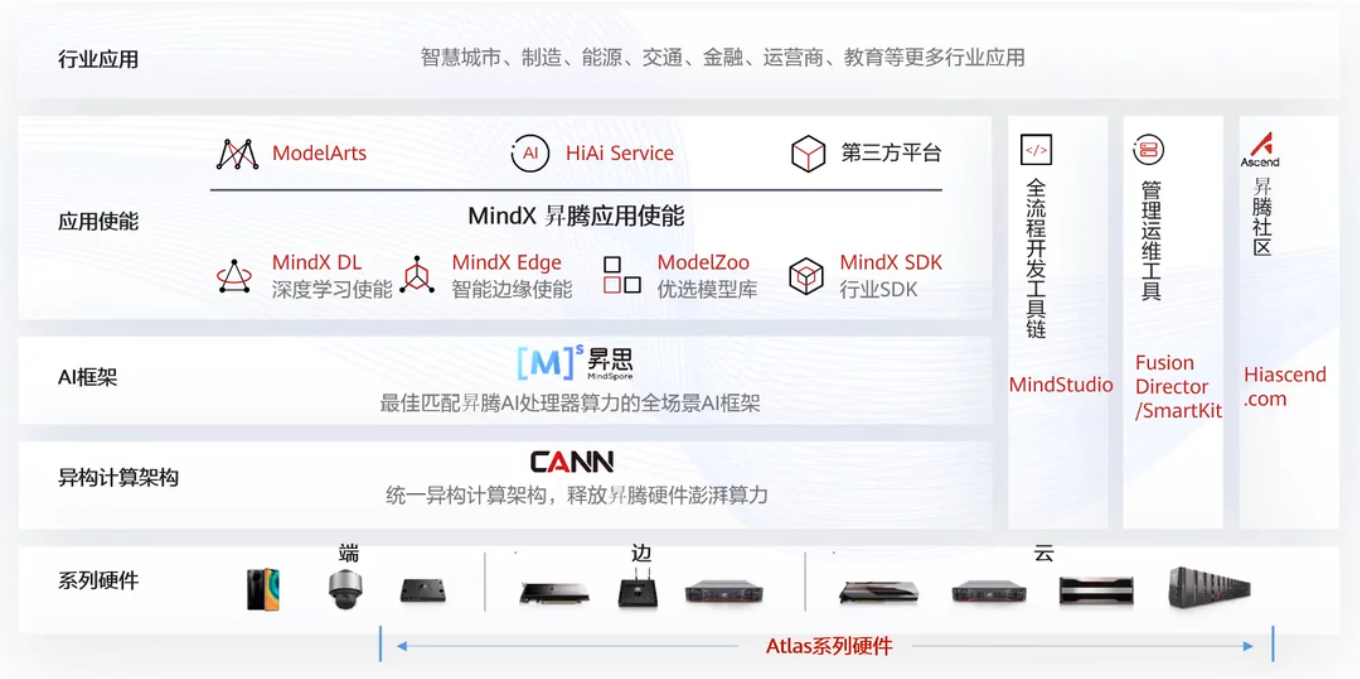


人工智能领域

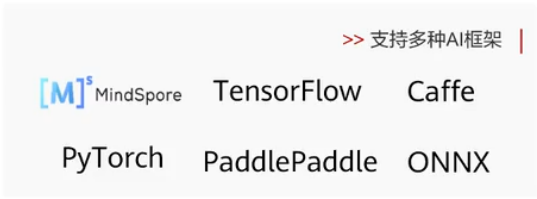
AI的通用技术方向主要为**计算机视觉**和**自然语言处理**

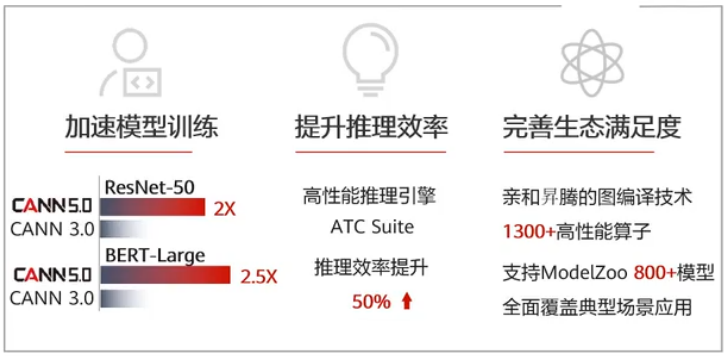
华为人工智能发展战略

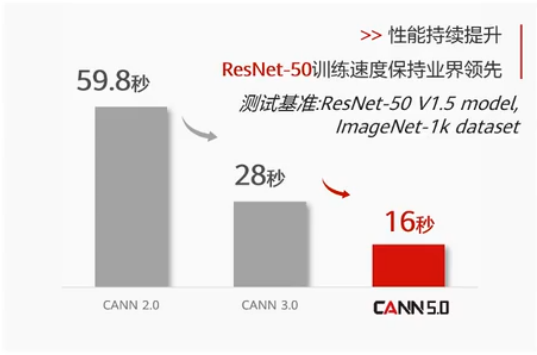
华为全栈全场景AI解决方案



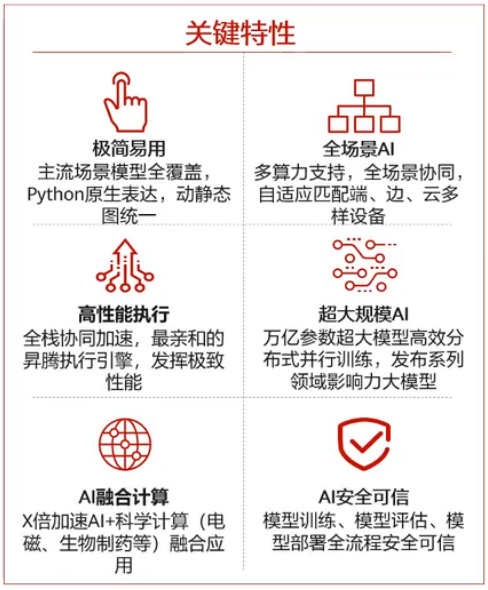
异构计算架构 CANN







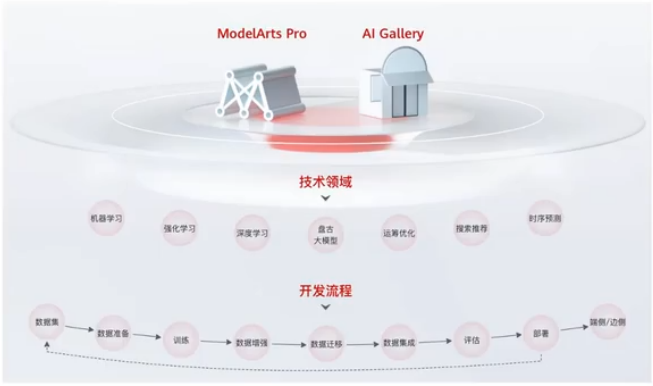
全场景AI计算框架 MindSpore



一站式AI开发平台 ModelArts

ModelArts 是面向开发者的一站式 AI 开发平台，为机器学习与深度学习提供海量数据预处理及交互式智能标注、大规模分布式训练、自动化模型生成，及端-边-云模型按需部署能力，帮助用户快速创建和部署模型，管理全周期 AI 工作流。

企业级开发平台ModelArts Pro，满足行业定制化需求。



昇腾应用使能 Mind X

从AI+到+AI

- AI+
  - 探索人工智能的自身能力
- +AI
  - 探索人工智能与行业结合的能力

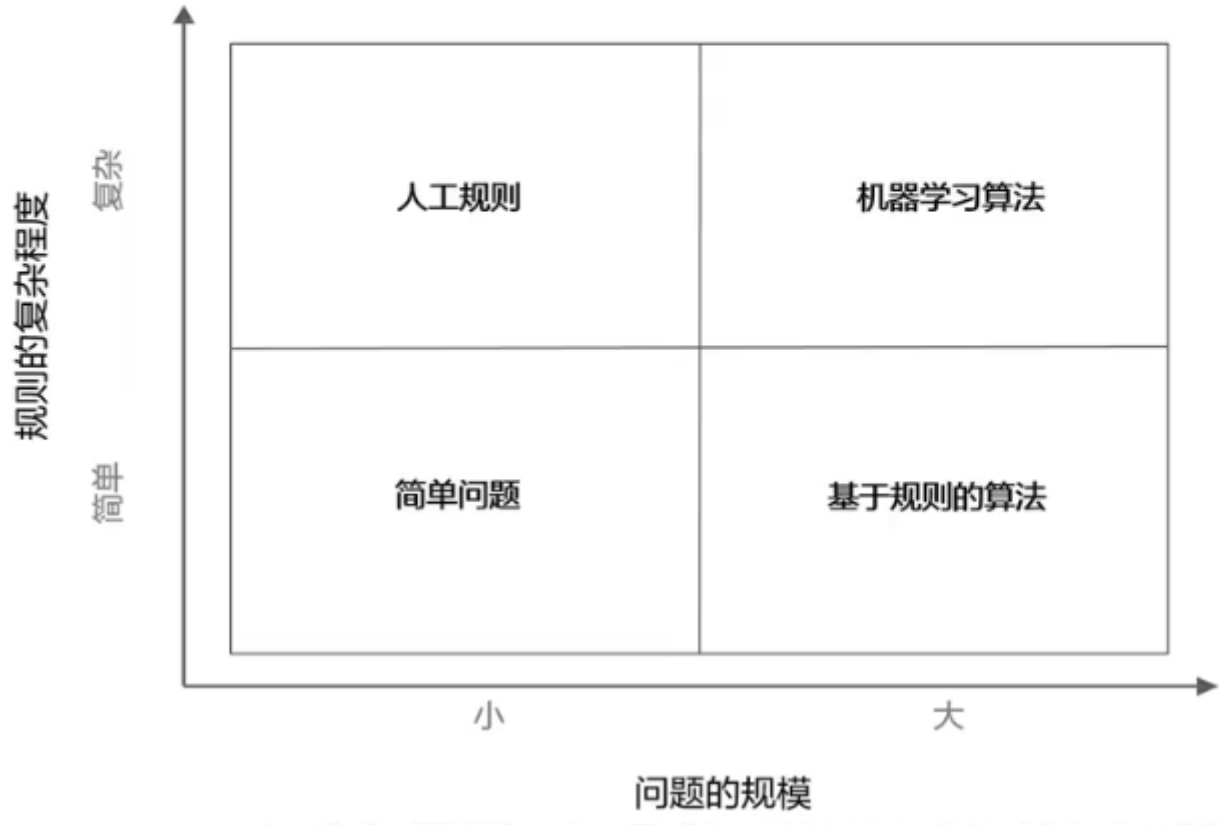
机器学习

机器学习算法

机器学习是研究"学习算法"的学科

若一个程序在任务T上以性能度量P衡量的性能随经验E而自我完善，则称这个程序在从经验E中学习，以改进在任务T上的性能P

什么时候该使用机器学习



机器学习算法的理性认识

假设目标方程为$f:X\to Y$

根据训练数据$D:(x_1,y_1),\ldots,(x_n,y_n)$，通过学习算法，学习到所有的特征参数，使得到的函数 g 尽可能逼近 f，即 $g\approx f$

机器学习解决的问题类型

- 分类
  - 根据样本特征，预测一个离散的输出标签
  - 识别物体类型、垃圾邮件分类
- 回归
  - 根据样本特征，预测一个连续的输出值
  - 预测房价、股票走势
- 聚类
  - 根据样本特征，将样本划分为若干个类别
  - 用户画像、新闻分类

机器学习算法分类

监督学习

用**已知类别**（称为标签）的样本，训练学习得到最优模型，再对未知类别的样本进行预测

回归：反映了样本数据集中样本的属性值的特性，通过函数表达样本映射的关系发现属性值之间的依赖关系

无监督学习

对**未知类别**的样本，学习算法将其划分为若干个类别，或学习样本之间的内在联系，然后对未知样本进行预测

聚类：将**无标签**的样本基于**数据内在相似度**进行分类，使得同一组内的样本相似度高，而不同组之间的样本相似度低，可以帮助发现数据的内在价值

半监督学习

从用少量有标记的数据和大量无标记的数据中学习

强化学习

感知**环境**，做出行动，根据状态和**奖惩**做出调整，寻找什么样的行为是最佳的

机器学习的整体流程

数据收集

数据集：机器学习中使用的一组数据，每个数据称为一个样本，每个样本包含若干个特征，反映样本的属性

训练集：训练过程中所使用的数据集，其中每个样本称为训练样本

测试集：训练过程中所使用的数据集，其中每个样本称为测试样本

数据集要划分为训练集与测试集，二者**不可以有交集**

数据质量决定了模型能力的上限

数据预处理

- 数据清理
  - 填充缺失值、删除异常值、去重
- 数据降维
  - 简化数据属性，降低训练复杂度，避免维度爆炸
- 数据标准化
  - 减小噪声，提高模型准确性
- 合并多个数据源的数据
  - 将来自多个数据源的数据进行合并，形成一个完整的数据集，使数据"没有歧视"，不存在偏见

脏数据：数据属性不完整，有异常值，数据不一致/有矛盾，无效重复，格式错误

数据的转换：将预处理后的数据转换为适合机器学习模型的表示形式

- 分类问题中，将类别数据编码为对应的数值表示
  - 独热编码（One-Hot Encoding）
    - 对每一个类别，创建一个新的二进制特征，每个类别用一个独立的二进制特征表示，类别之间没有顺序关系

        - 红：[1, 0, 0]，绿：[0, 1, 0]，蓝：[0, 0, 1]
    - 哑编码（Dummy Encoding）
        - 与独热编码相似，但减少了一个类别的特征，通常将其中一个类别作为基准，其编码为全0
        - 红：[1, 0]，绿：[0, 1]，蓝：[0, 0]
- 将数值数据转换为类别数据以减少变量的值（分段）
- 从文本数据中提取数据
- 将图像数据转换为数值数据

特征提取与选择

特征选择的必要性：

- 简化模型，使模型更容易被使用者所解释
- 降低训练难度，降低时间成本
- 避免维度爆炸
- 提高模型泛化性，避免过拟合

特征工程：对特征进行归一化、标准化，保证同一模型不同输入变量的值域都相同

特征扩充：对现有特征进行组合/转换以生成新的特征

特征选择的方法：

- 过滤法 Filter
    - 遍历所有特征，评估每个特征与目标属性之间的相关性，保留高相关性的特征
    - 过滤法选择特征时与模型本身无关
    - 适用于选择冗余的变量，不考虑特征之间的关系
- 包装器法 Wapper
    - 遍历所有特征，生成数个特征子集，根据**模型**的准确性对特征子集评分，选择，准确性更高的特征子集作为结果
    - 计算量特别大
    - 对**特定类型**的模型性能好，但是泛化能力弱
- 嵌入法 Embedded
    - 将特征选择作为模型构建的一部分
    - 遍历所有特征，生成一个特征子集，用学习算法对其进行效果评估，若效果不佳则重复生成特征子集
    - 最常见的嵌入式特征选择方法是**正则化方法**

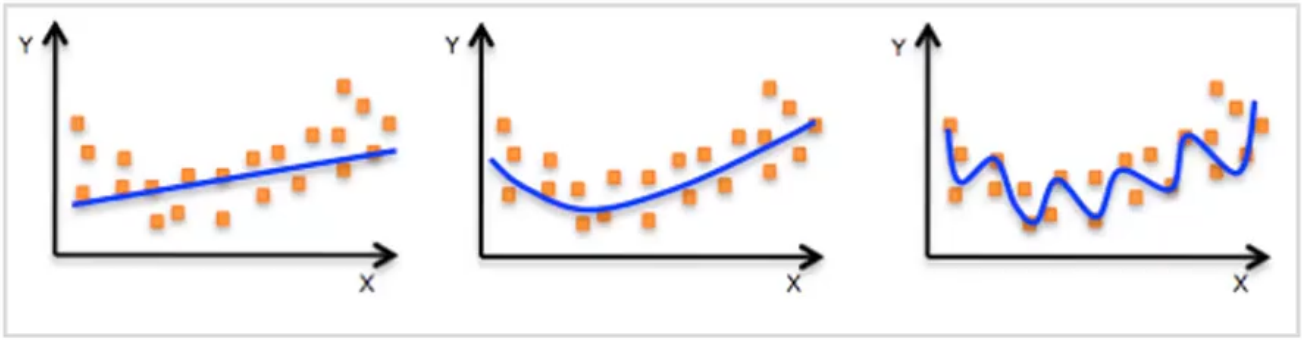
模型训练

模型评估

- 泛化能力（最重要）
    - 能否在实际的应用中获得较好的效果
- 可解释性
    - 预测的结果是否容易被解释
    - 提高模型的可信度，解释整个模型的思路
- 预测速率
    - 每一次预测需要多长时间

模型的有效性：

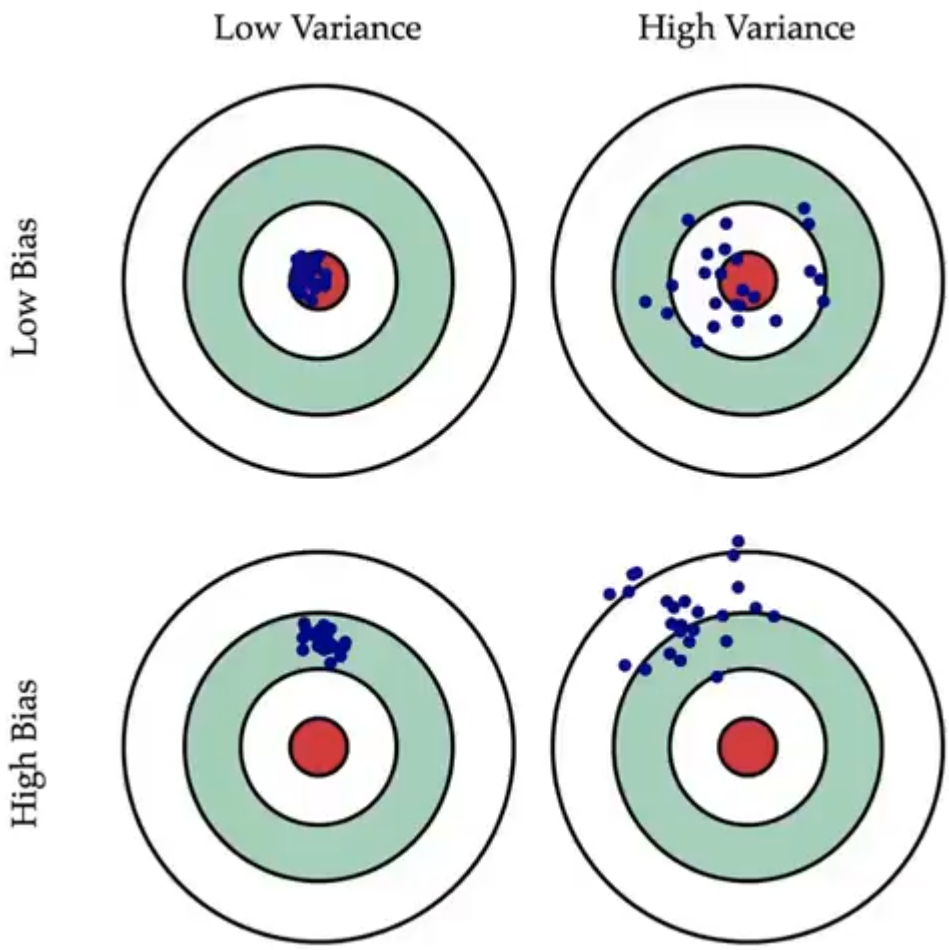
- 泛化能力（鲁棒性）
  - 模型是否适用于新的样本
- 误差：预测值与实际值之间的差异
  - 训练误差：模型在训练集上的误差
  - 泛化误差：模型在测试集上的误差，实际希望泛化误差更小
  - 预测总误差 $=$ 偏差^2 $+$ 方差 $+$ 不可消除的误差
    - 方差：模型的预测结果在**均值**附近的波动程度
    - 偏差：预测值与**实际值**之间的差异
- 欠拟合：训练误差较大
- 过拟合：训练误差较小但泛化误差较大

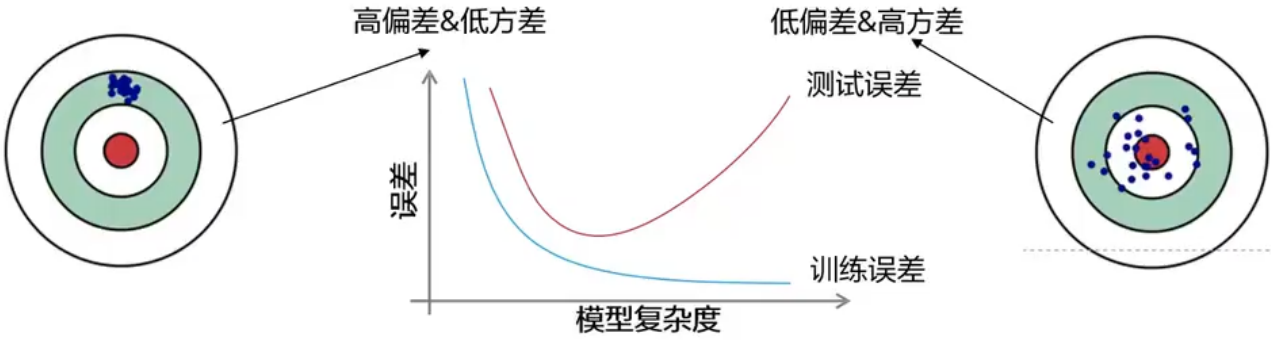


欠拟合
没学到特征

好的拟合

过拟合
学习了噪声





模型的容量（复杂度）：拟合各种函数的能力

- 容量小的模型不适用于复杂任务，容易欠拟合
- 容量大的模型适用于复杂任务，但容量高于任务所需时容易过拟合

对回归问题的评估方法

- 平均绝对误差 $\mathrm{MAE} = \frac{1}{m}\sum_{i=1}^{m}|y_i - \hat{y}_i|$
- 平均方差 $\mathrm{MSE} = \frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_i)^2$
- $R^2 = 1 - \frac{\text{RSS}}{\text{TSS}} = 1 - \frac{\sum_{i=1}^{m} (y_i - \hat{y}*i)^2}{\sum*{i=1}^{m} (y_i - \bar{y})^2}$，其中RSS为预测值与样本值的差异情况，TSS为样本之间的差异

对分类模型的评估方法

使用**混淆矩阵**

- 术语：
  - P：正元组，感兴趣的主要类的元组。
  - N：负元组，其他元组。
  - TP：真正例，被分类器正确分类的正元组。
  - TN：真负例，被分类器正确分类的负元组。
  - FP：假正例，被错误地标记为正元组的负元组。
  - FN：假负例，被错误的标记为负元组的正元组。

实际 \ 预测	yes	no	合计
yes	TP	FN	P
no	FP	TN	N
合计	P'	N'	$P+N$

混淆矩阵

度量	公式
准确率、识别率	$\frac{TP+TN}{P+N}$
错误率、误分类率	$\frac{FP+FN}{P+N}$
敏感度、真正例率、召回率（$recall$）	$\frac{TP}{P}$
特效性、真负例率	$\frac{TN}{N}$
精度（$precision$）	$\frac{TP}{TP+FP}$
F_1值，精度和召回率的调和均值	$\frac{2\times precision\times recall}{precision+recall}$
F_β值，其中β是非负实数	$\frac{(1+\beta^2)\times precision\times recall}{\beta^2\times precision+recall}$

混淆矩阵至少是一个m*m的表，理想情况下主对角线上的数字越大越好，其余数字趋于0

机器学习的重要方法

梯度下降法

将当前位置的负梯度方向，即斜率为负的最大值，作为搜索方向。越接近目标值，梯度越小。

$w_{k+1}=w_k-\eta\nabla f_{w_k}(x^i)$，其中 η 为学习率，i 表示第i个数据，权重参数 w 表示每次迭代变化的大小

- BGD 批量梯度下降
  - 每次迭代使用数据集中的所有样本在当前点的梯度之和对权重参数更新
  - 效果最好，最稳定，但是资源和时间成本高
- SGD 随机梯度下降
  - 每次迭代随机使用数据集中的一个样本的梯度对权重参数更新

    - 样本选取是随机的，导致不稳定性
- MBGD 小批量梯度下降
    - 每次迭代随机使用数据集中的一部分样本（称为**Batch**）的梯度对权重参数更新
    - 平衡BGD和SGD

参数与超参数

参数：模型中本身所需的权重，由模型**自动学习**

超参数：**人为设定**的参数，用于控制训练

在训练集上根据性能指标对**参数**进行优化
在验证集上根据性能指标对**超参数**进行优化

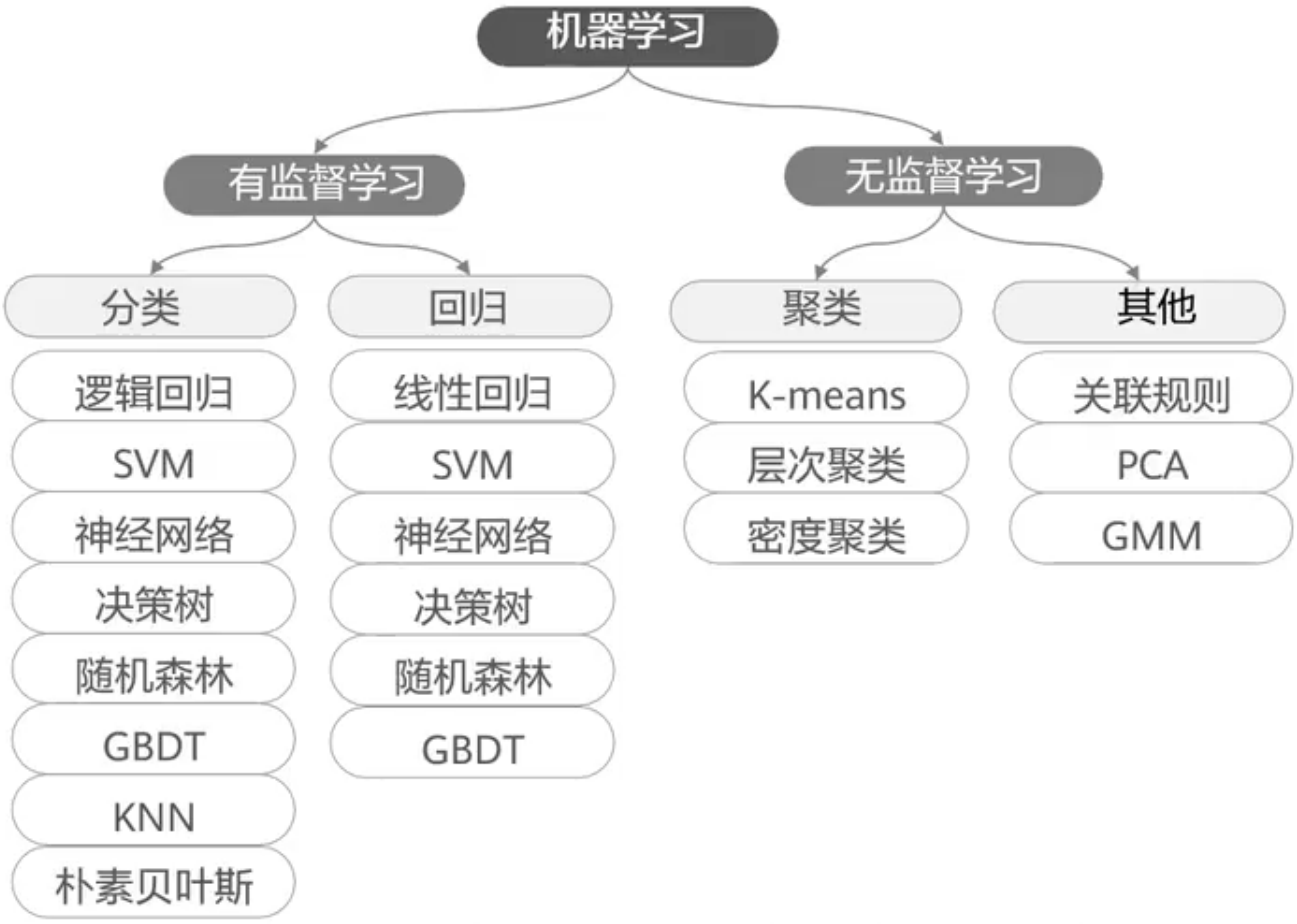
超参数的搜索方法：

- 网格搜索
    - 穷举所有可能的超参数组合
    - 成本高，耗时长，仅适用于超参数数目较少的机器学习算法
- 随即搜索
    - 随机选择超参数组合，试图找到最佳超参数子集

交叉验证：原始数据除了分为训练集和测试集，还有验证集，使用训练集训练模型，使用验证集评估模型用于调节超参数，使用测试集评估模型性能

- k折交叉验证（K-CV）
    - 将原始数据（平均）分为k组
    - 将其中一组作为验证集，其余k-1组作为训练集训练模型，重复k次，得到k个模型
    - 用k个模型最终的验证集的准确度的平均数作为次K-CV下分类器的性能指标

机器学习常见算法



线性回归

用数理统计中的回归分析，确定两个或以上变量间相互依赖的定量关系的一种统计分析方法

模型函数： $h_w(x) = w^T x + b$
误差 $\epsilon =$ 真实值$y - w^T \cdot x$
损失函数： $J(w) = \frac{1}{2m}\sum(h_w(x) - y)^2$

只要损失值最小，即可得到最优的模型

线性回归的扩展-多项式回归

模型函数： $h_w(x) = w_1 x + w_2 x^2 + \cdots + w_n x^n + b$

多项式回归仍是线性回归，因为权重参数 w 之间的关系是线性的

为了防止过拟合可以向损失函数中加入**正则项**：

- L1：Lasso回归，$\lambda\sum|w|$
  - $J(w) = \frac{1}{2}\sum(h_w(x) - y)^2 + \lambda\sum|w|$
- L2：Rigde回归，$\lambda\sum w^2$
  - $J(w) = \frac{1}{2}\sum(h_w(x) - y)^2 + \lambda\sum w^2$
- 对模型的复杂度进行约束，使得模型参数不会过度地拟合训练数据中的噪声，鼓励模型选择更简单、更平滑的解

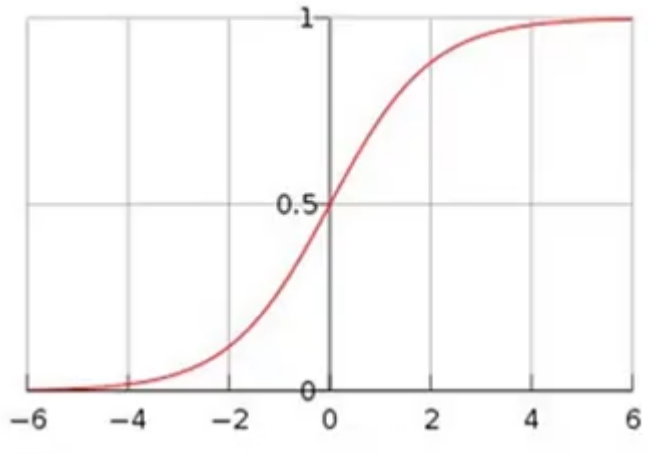
逻辑回归

一种分类模型，解决分类问题。

Sigmoid激活函数将线性回归的输出值映射到0-1之间，表示属于某一类的概率，取概率更大的类别作为预测结果

$P(Y=1|x)=\frac{e^{wx+b}}{1+e^{wx+b}}$

$P(Y=0|x)=\frac{1}{1+e^{wx+b}}$



用最大似然估计计算得到的逻辑回归损失函数：
$J(w)=\frac{1}{m}\sum(y\log h_w(x)+(1-y)\log(1-h_w(x)))$
其中 w 为权重参数，m 为样本数量，x 为样本，y 为真实值，$h_w(x)$ 为预测值

逻辑回归的扩展-Softmax回归

将Sigmoid的二分类扩展为多分类

$$p(y=k\mid x;w)=\frac{e^{w_k^Tx}}{\sum_{l=1}^{K}e^{w_l^Tx}},k=1,2,\ldots,K$$

- 首先计算每个类别的概率：$e^{w_k^Tx}$
- 然后计算所有类别的概率之和：$\sum_{l=1}^{K}e^{w_l^Tx}$
- 对每个类别的概率除以概率之和，得到输出概率
- 输出概率最大的类别为预测类别

决策树

决策树是一个树结构（可能是二叉或非二叉），每个非叶子节点表示一个特征属性上的测试，每个分支表示这个特征属性在某个值上的输出，每个叶子节点表示一个类别。

即对每个特征属性进行分支，直到可能的一个结果

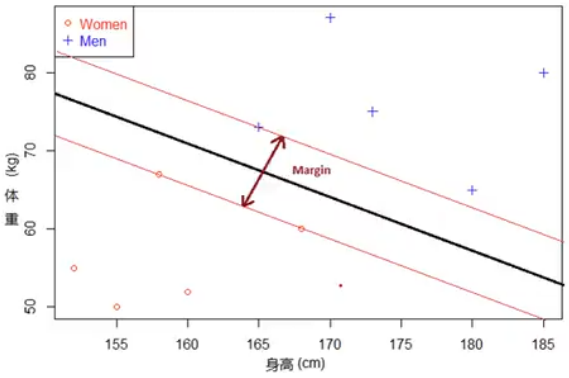
对特征属性的分类依据：

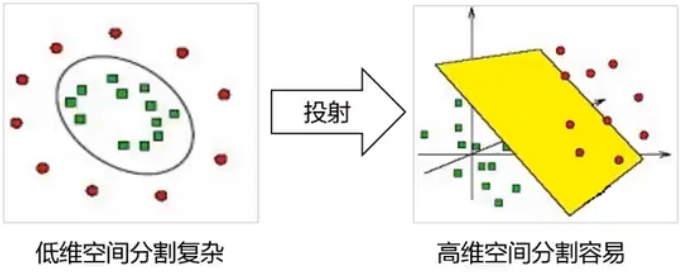
- 信息熵 $H(X)=-\sum_{k=1}^{K}p_k\log_2(p_k)$，其中 p_k 表示样本属于类别k的概率
- Gini系数 $Gini=1-\sum_{k=1}^{K}p_k^2$，
- 由信息熵和Gini系数来量化划分操作的"纯度"，选择"纯度"最高的属性作为分割数据集的数据点
- 分割前后的纯度差异越大，决策树越好

SVM 支持向量机

一种二分类模型，是定义在特征空间上的**间隔最大**的**线性**分类器。

支持向量机的学习算法是求解凸二次规划的最优化算法



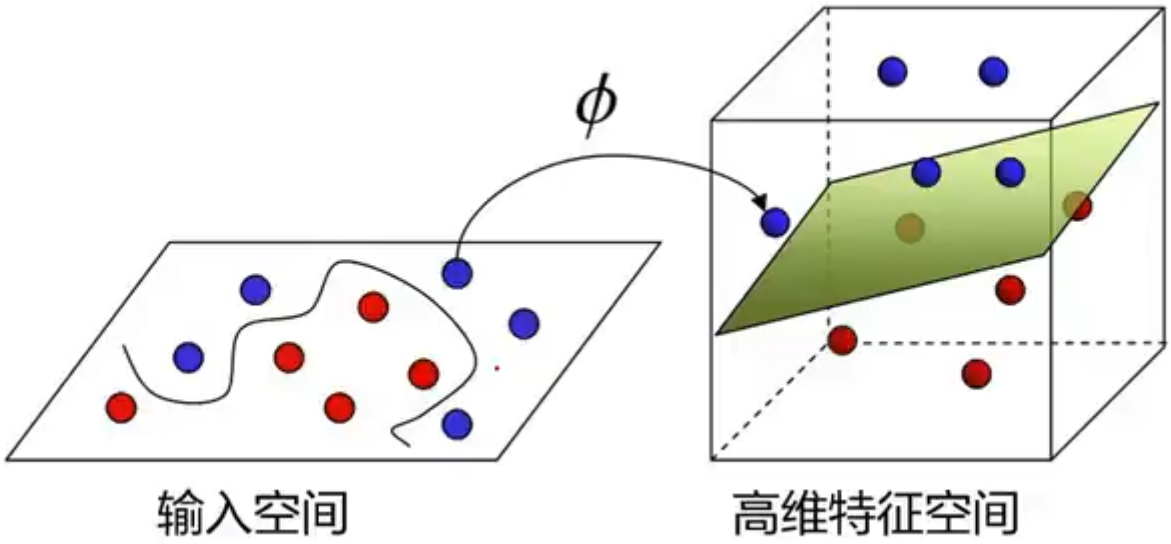


核心思路：

- 取离直线较近的点，称为**支持向量**，使支持向量到直线的距离最大
- 在二维空间中用**直线**分割，在三维空间用**平面**分割，在高维空间用**超平面**分割

非线性支持向量机：

- 使用**核函数**将数据映射到高维空间，从而用线性平面或超平面进行分割
- 常用核函数：线性、多项式、**高斯**、Sigmoid



KNN K最邻近算法

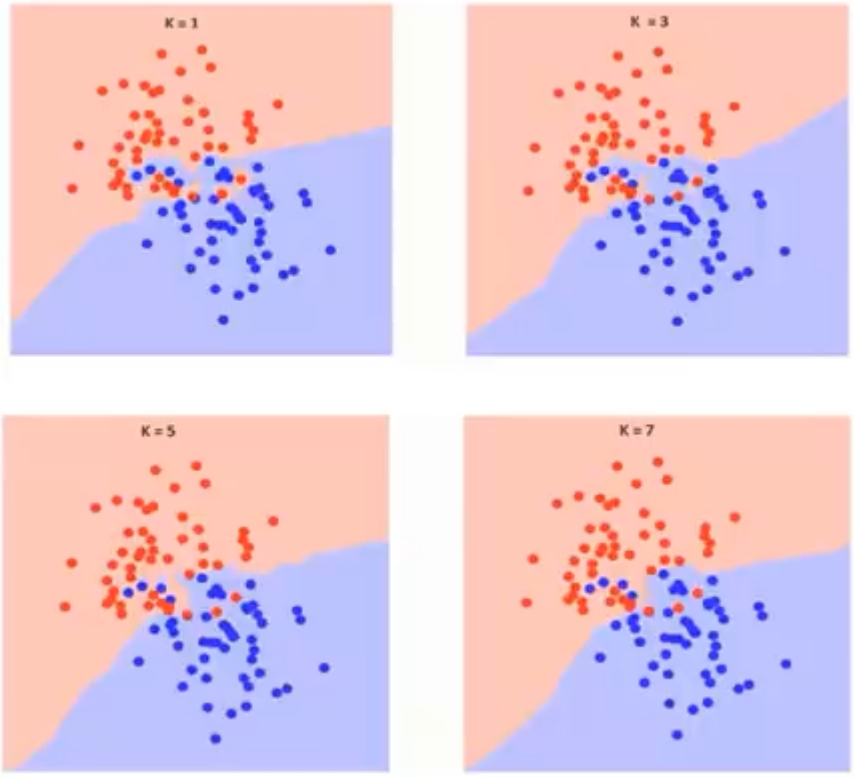
理论上成熟且最简单的机器学习的分类算法

给定空间中一个点和K值（属于超参数），在空间中找出离该点最近的K个点，这K个点中数量最多的类别即为该点的预测类别

- 近朱者赤 近墨者黑
- 分类预测时采用多数表决法，回归预测时采用平均值法
- 计算量很大

K值的选择：

- K值过小时分割过于细腻，容易过拟合
- K值过大时分割边界趋于平滑，容易欠拟合



朴素贝叶斯

基于贝叶斯定理的简单的多分类算法

假设**特征之间是独立**的，给定岩本特征X，则样本属于类别H的概率为：

$$P(C_k|X_1,\dots,X_n) = \frac{P(X_1,\dots,X_n|C_k)P(C_k)}{P(X_1,\dots,X_n)}$$

- $X_1,X_2....X_n$是数据的特征，通常用m个属性集的测量值描述，
  - 比如说颜色特征可能有红，黄和蓝三个属性。
- C_k表示该数据属于某个特定类C。
- $P(C_k \mid X_1,X_2....X_n)$ 是后验概率，或在条件C_k下，H的后验概率。
- $P(C_k)$是先验概率，$P(C_k)$独立于$X_1,X_2....X_n$。
- $P(X_1,X_2....X_n)$是X的先验概率。

即在知道先验概率的情况下，计算后验概率

如已知下雨和下雨时打伞的概率，求打伞时下雨的概率

集成学习

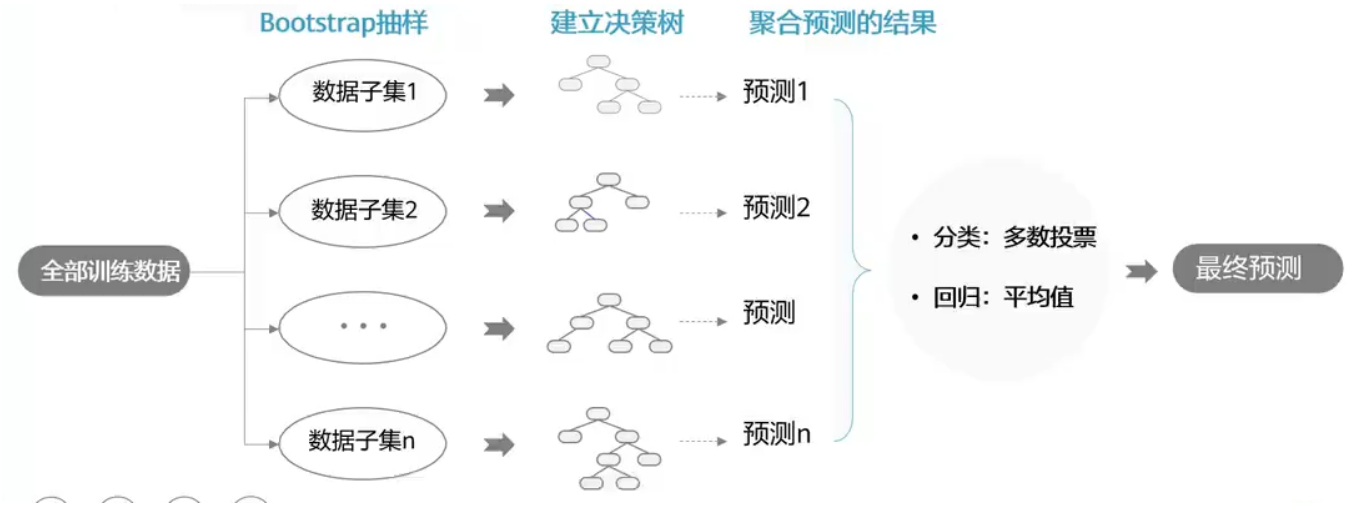
将多个学习器组合起来解决同一个问题

多个弱分类器组合起来，形成一个强分类器

集成学习的分类：

- Bagging
  - 构建多个基本学习器，平均其预测
- Boosting
  - 按顺序的方式构建多个基本学习器，逐步减少学习器的误差

随机森林：Bagging + 决策树
每一个决策树的数据集都是原数据集的一个子集



K-means

最基础的无监督学习算法

输入n个数据对象和聚类的最终个数k，使得同一聚类中的对象相似度较高，不同聚类中的对象相似度较低

层次聚类

无监督学习算法

在不同层次上对数据集进行划分，形成树形的聚类结构