机器学习

李成龙

安徽大学人工智能学院
“多模态认知计算”安徽省重点实验室
合肥综合性国家科学中心人工智能研究院

内容安排

- 什么是机器学习
- 机器如何学习
- 如何让机器学习的更好
- 为什么机器能学习

内容安排

- 机器如何学习
  - 有监督学习
    - 感知机
    - 支持向量机
    - 朴素贝叶斯分类
    - 决策树
    - 集成学习（Bagging算法与随机森林、Boosting算法）
    - 线性回归
    - 逻辑回归
    - Softmax回归
    - 神经网络与深度学习
  - 无监督学习
    - 聚类
    - 主成分分析

本节目录

本节目录

集成学习

- **集成学习概念**
  - 集成学习将多个性能一般的普通模型进行有效集成，形成一个性能优良的集成模型
  - 通常将这种性能一般的普通模型称为个体学习器
  - 如果所有个体学习器都属于同类模型，则称由这些个体学习器产生的集成模型为同质集成模型，并称这些属于同类模型的个体学习器为基学习器
  - 将属于不同类型的个体学习器进行组合产生的集成模型称为异质集成模型

集成学习

- **集成学习概念**
  - 若某学习问题能被个体学习器高精度地学习，则称该学习问题是强可学习问题，并称相应个体学习器为强学习器
  - 反之，则可定义弱可学习问题，并称相应个体学习器为弱学习器
  - 当直接构造其强学习器比较困难时，可通过构造一组弱学习器生成强学习器，将强可学习问题转化为弱可学习问题

集成学习

- **集成学习概念**
  - 合理地选择弱学习器是集成学习首要必须解决的问题
  - 对于图所示的二分类任务（圆圈表示分类正确，叉号表示分类错误），图中每个分类器的分类正确率均为1/3，则由少数服从多数原则进行组合得到集成模型的分类正确率为 0

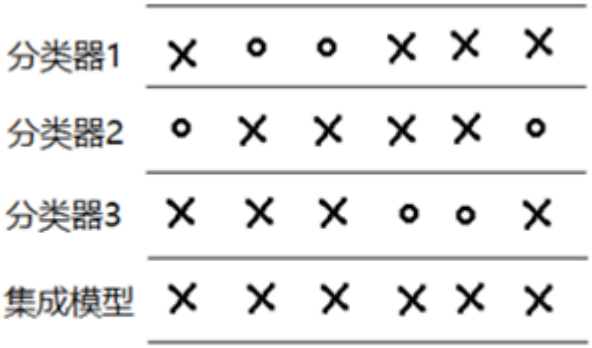


图　　过弱泛化性能个体学习器集成效果

集成学习

- **集成学习概念**
  - 上述情况是由于弱学习器泛化性能均太弱造成的
  - 在集成学习的实际应用当中，应尽可能选择泛化性能较强的弱学习器进行组合
  - 如图所示，当每个弱分类器分类错误的样本各不相同时，则能得到一个效果优异的集成模型

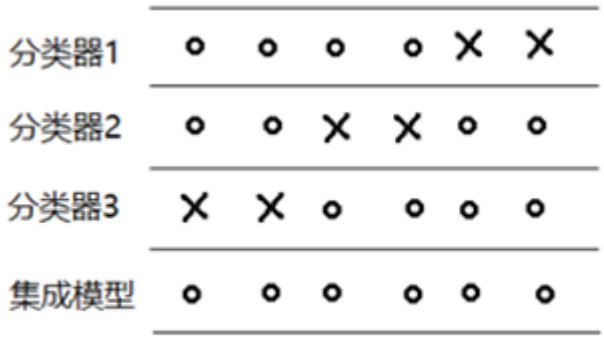


图　　个体学习器的差异对集成结果的影响

集成学习

- **集成学习概念**
  - 在选定弱学习器时，集成学习需要通过一定的组合策略将它们组合起来形成较高性能的强学习器
  - 使用单个学习器会带来过多的模型偏好，从而产生模型泛化能力不强的现象，结合多个弱学习器则可以有效降低此类风险

集成学习

- **集成学习基本范式**
  - 集成学习包括两个基本步骤
    - 首先根据数据集构造弱学习器
    - 对弱学习器进行组合得到集成模型

集成学习

- **集成学习基本范式**
  - 构造弱学习器
    - 给定样本数据集D,对该样本数据集进行某种随机采样方式生成多个具有一定差异的训练样本集$D_1, D_2, \cdots, D_m$
    - 分别通过这些训练样本集产生若干具有一定差异的弱学习器$L_1, L_2, \cdots, L_m$， 以满足集成学习的要求
    - 可以如图所示并行执行弱学习器构造过程

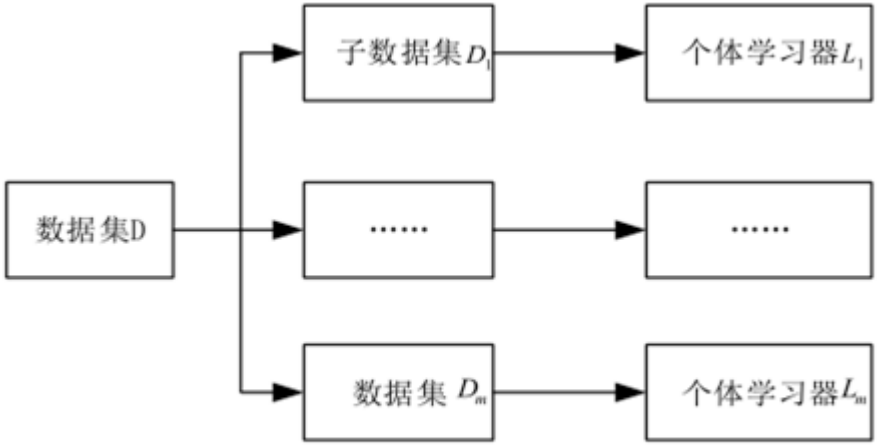


图　　弱学习器并行构造方式

集成学习

- **集成学习基本范式**
  - 构造弱学习器
    - 并行构造方式忽略了弱学习器之间的某些联系，有时候丢掉一些重要信息
    - 可以用串行的方式逐个构造弱学习器，使得各弱学习器之间可以存在一定的关联
    - 弱学习器的串行构造过程如图所示

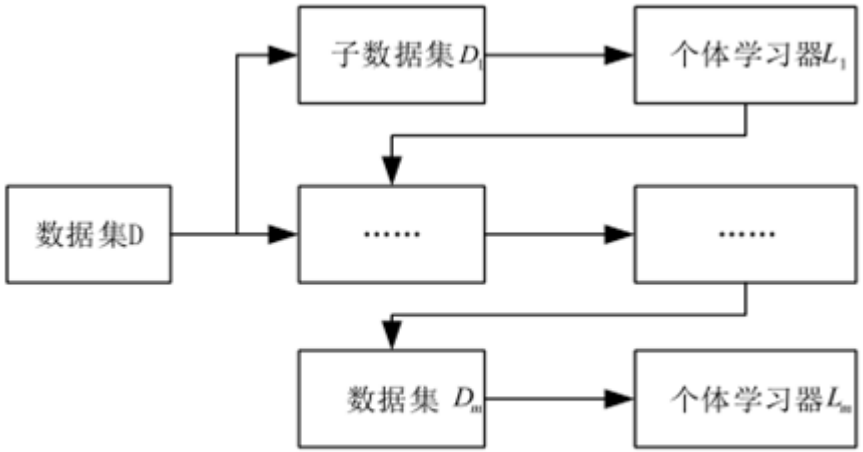


图　　弱学习器串行构造方式

集成学习

- **集成学习基本范式**
  - 组合弱学习器
    - 在集成学习的弱学习器组合阶段，不同学习任务所用组合策略会有所不同
    - 对于输出空间为实数域的回归任务，通常使用平均法实现多个弱回归器的组合
    - 设有m个弱回归器，第i个弱回归器对样本输入X的预测输出为$L_i(X)$，则可取集成模型$L(X)$的输出为各个弱回归器输出的简单平均值，即有

$$L(X) = \frac{1}{m}\sum_{i=1}^{m} L_i(X)$$

这种简单的组合策略被称为简单平均法

集成学习

- **集成学习基本范式**
  - 组合弱学习器
    - 简单平均法规定每个弱回归器对集成模型输出的贡献都相同
    - 而不同弱回归器的重要性通常会有一些差异，此时简单平均法会导致集成学习的预测输出因过分依赖不太重要的弱回归器而降低泛化性能
    - 可用权重对弱回归器的重要性进行加权计算，通过加权平均法实现多个弱回归器的组合。令ω_i为弱回归器$L_i(X)$的权重，则有

$$L(X) = \sum_{i=1}^{m} \omega_i L_i(X)$$

集成学习

- **集成学习基本范式**
  - 组合弱学习器
    - 对于输出空间为离散集合的分类任务，通常用投票法实现多个弱分类器的组合
    - 设有m个弱分类器，输出空间为$s_{out} = \{c_1, \cdots, c_n\}$,其中$c_j$表示第$j$类的类标。令$L_i(X, c_j) = 1$表示一个布尔值，当且仅当第$i$个弱分类器$L_i(X)$对于样本输入$X$的预测输出为$c_j$时，$L_i(X, c) = 1$
    - 可将集成模型输出$L(X)$定义为

$$L(X) = c_{arg_j max \sum_{i=1}^{m} L_i(X, c_j)}$$

    - $L(X)$的输出值为得票数最多的类别，将其作为预测结果。通常称这种投票方法为相对多数投票法

集成学习

- **集成学习基本范式**
  - 组合弱学习器
    - 在票数比较分散的情况下，相对多数投票法的最多票数可能会很小，此时大大增加集成模型出现错误分类的概率
    - 如果限制模型$L(X)$预测输出类型的最低得票数不得小于弱分类器数目m的一半，否则$L(X)$拒绝输出预测结果，称这种改进相对多数投票法为绝对多数投票法

集成学习

- **集成学习基本范式**
  - 组合弱学习器
    - 相对多数投票法和绝对多数投票法显然均未考虑不同弱分类器在重要性方面的差异
    - 对于一组重要性不相同的弱分类器 $L_i(X)$，令 ω_i 为 $L_i(X)$ 的权重，则可通过带加权计算的投票方法对其进行组合，得到集成模型的预测输出为

$$L(X) = c_{arg_j max} \sum_{i=1}^{m} \omega_i L_i(X)$$

集成学习

- **集成学习泛化策略**
  - 集成学习目标是获得具有较好泛化性能机器学习模型
  - 现以回归任务为例，假设回归任务的真实映射为f, 集成模型L由m个弱回归器$L_1, \cdots, L_m$通过简单平均法组合生成，即对于输入样本X，集成模型$L(X)$的预测输出为

$$L(X) = \frac{1}{m}\sum_{i=1}^{m} L_i(X)$$

  - 则集成模型L关于输入样本X的误差可表示为

$$Q(L, X) = (f(X) - L(X))^2$$

集成学习

- **集成学习泛化策略**
  - 弱回归器L_i对输入样本X的预测$L_i(X)$与集成模型预测结果$L(X)$的差异可表示为

$$D(L_i, X) = (L_i(X) - L(X))^2$$

  - 一组弱回归器的差异度或多样性可表示为该组所有弱回归器关于集成模型输出偏差的平均值，即有

$$aveD = \frac{1}{m}\sum_{i=1}^{m}(L_i(X) - L(X))^2$$

  - 令$Q(L_i, X)$表示所有弱回归器对于输入样本X的平均误差：

$$Q(L_i, X) = \frac{1}{m}\sum_{i=1}^{m}(f(X) - L_i(X))^2$$

  - 则有：$Q(L, X) = Q(L_i, X) - aveD$

集成学习

- **集成学习泛化策略**
  - $Q(L,X)= \ Q(L_i,X) - aveD$
  - 由以上公式可知，集成模型L关于输入样本X的预测误差$Q(L,X)$等于所有弱回归器关于输入样本X的平均误差减去这组弱回归器的差异度ave$D$
  - 这个结论为集成模型泛化性能的提高给出了两个基本思路，即降低个体学习器的泛化误差和提高个体学习器的多样性

集成学习

- **集成学习泛化策略**
  - 降低弱学习器的泛化误差：样本扩充、范数惩罚等机器学习正则化策略
  - 提高个体学习器的多样性：改变训练样本和改变模型训练参数

集成学习

- **集成学习泛化策略**
  - 提高个体学习器的多样性：改变训练样本和改变模型训练参数
    - 改变训练样本的角度：通过对样本数据采样增加输入样本随机性，由此提高弱学习器多样性
      - 具体地说，通过样本数据集D构造m个不同的弱学习器，使用某种采样方法从D生成m个有差别的训练样本数据子集$\{D_1, D_2, \ldots, D_m\}$，分别用这些训练子集进行训练就可以构造出$m$个有差别弱学习器
    - 改变模型训练参数的角度：由于弱学习器自身参数的不同设置以及在不同训练阶段的产生的不同参数也会产生不同的弱学习器，故可从改变神经网络初始连接权重、隐层神经元个数等参数的角度增加弱学习器的多样性
      - 通常综合使用多种泛化策略构造同一个集成模型
      - 例如在构建某个集成模型时可能既采用数据样本采样方法，又对模型参数进行随机选择以提高弱学习器的多样性

集成学习

- **集成学习泛化策略**
  - 事实上，弱学习器的个数也在一定程度上影响集成模型的泛化性能
  - 例如，对于某个二分类任务的集成学习问题，假设每个弱学习器的错误率均为ε，则可从理论上证明其集成模型的泛化误差上界为

$$H(m) = e^{-\frac{1}{2}m(1-\varepsilon)^2}$$

其中m为组成集成模型的弱学习器个数。$H(m)$作为关于弱学习器个数m的函数，取值m随着的增加而减小。因此，增加弱学习器的个数也能达到提升集成模型泛化性能的目的

本节目录

- 集成学习
- Bagging集成策略
- 随机森林

Bagging集成策略

- **自助采样法**
  - 对于给定的样本数据集D,Bagging集成学习主要通过自助采样法生成训练样本数据子集
  - 假设D中包含有n个样本数据，自助采样对D进行n次有放回的随机采样并将采样样本纳入训练集
  - 可将这些未被抽到的样本构成测试集，用于测试弱学习器的泛化性能
  - 对样本数据集D进行多次自助采样就可以分别生成多个具有一定差异的训练样本子集$D_1, D_2, \cdots, D_K$,可分别通过对这些子集的训练构造出所需的弱学习器
  - 一般通过简单平均法集成多个弱回归器，通过相对多数投票法集成多个弱分类器

Bagging集成策略

- **自助采样法**
  - Bagging集成学习的基本流程图如图所示

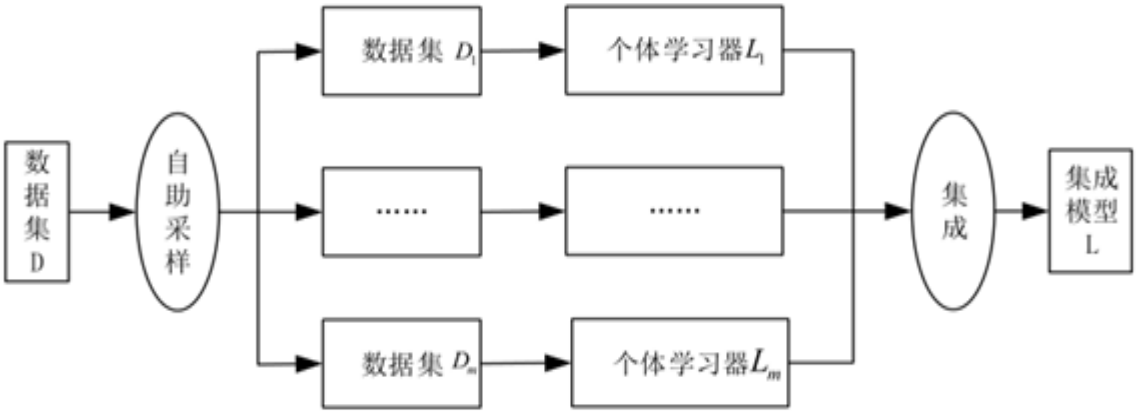


图　　Bagging 集成学习流程图

Bagging集成策略

- **例题1：**现有一组某市房屋价格与房屋位置数据如表 所示，其中X表示房屋到市中心的直线距离。试用Bagging集成学习方法构造一个包含三个线性回归模型的集成模型，并使用该集成模型预测距离市中心5.5千米的房屋价格

表 房屋价格与位置的数据样本集D

序号	1	2	3	4	5	6	7	8	9
$X(\mathrm{km})$	4.2	7.1	6.3	1.1	0.2	4.0	3.5	8	2.3
$y(元/\mathrm{m}^2)$	8600	6100	6700	12000	14200	8500	8900	6200	11200

Bagging集成策略

- 补充知识：线性回归
  - 在一维或者多维空间里，线性回归的目标是找到一条直线(对应一维)、一个平面(对应二维)或者更高维的超平面，使样本集中的点更接近它，也就是残留误差最小化
  - 使用平方损失函数，也被称为最小二乘法

$$l\left(g\left(\boldsymbol{x}^i;\boldsymbol{w}\right),y^i\right)=\left(g\left(\boldsymbol{x}^i;\boldsymbol{w}\right)-y^i\right)^2$$

$$L=\frac{1}{n}\sum_{i=0}^{n}\left(\boldsymbol{w}^T\boldsymbol{x}^i-y^i\right)^2$$

  - 对于此类线性回归问题，优化二次凸函数，一阶导数为零，即：

$$\boldsymbol{X}\boldsymbol{w}-\boldsymbol{y}=0$$

$$\boldsymbol{w}=\boldsymbol{X}^{+}\boldsymbol{y}$$

其中，$\boldsymbol{X}^{+}$是$\boldsymbol{X}$的伪逆矩阵(pseudo-inverse):

$$\boldsymbol{X}^{+}=(\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{y}$$

Bagging集成策略

序号	1	2	3	4	5	6	7	8	9
X(km)	4.2	7.1	6.3	1.1	0.2	4.0	3.5	8	2.3
y(元/m^2)	8600	6100	6700	12000	14200	8500	8900	6200	11200

- 依题意知，需要构造三个弱学习器。如表所示，可通过对样本数据集D进行三次自助采样获得如下表所示的三个训练样本子集D_1, D_2, D_3

表 由数据样本集生成的样本子集

训练样本子集D_1

序号	1	2	3	4	5	6	7	8	9
$X(km)$	4.2	4.2	4.2	6.3	1.1	0.2	3.5	3.5	2.3
y(元/m^2)	8600	8600	8600	6700	12000	14200	8900	8900	11200

训练样本子集D_2

$X(km)$	4.2	4.2	7.1	7.1	1.1	4.0	4.0	3.5	2.3
y(元/m^2)	8600	8600	6100	6100	12000	8500	8500	8900	11200

训练样本子集D_3

$X(km)$	4.2	1.1	0.2	4.0	4.0	4.0	4.0	3.5	8
y(元/m^2)	8600	12000	14200	8500	8500	8500	8500	8900	6200

Bagging集成策略

- 假设线性回归模型为$L(X)=\theta_0 X+\theta_1$，则可分别通过训练集D_1, D_2, D_3构造相应的弱学习器L_1, L_2, L_3。使用最小二乘法, 不难得到L_1, L_2, L_3的具体表达式如下

$$L_1(X)=-1216.488X+13731.8219$$

$$L_2(X)=-984.0959X+12822.6216$$

$$L_3(X)=-1015.2945X+13044.9688$$

- 使用简单平均法集成L_1, L_2, L_3，得到如下$L(X)$集成模型
- 带入具体数据可以算得

$$L(5.5)=\frac{bias(L_1)+bias(L_2)+bias(L_2)}{3}=\mu=7304$$

- 由此可见，通过Bagging集成学习产生的集成模型$L(X)$并未改善对弱回归器的预测偏差。假设三个弱回归器的预测方差var(L)均为σ^2，则集成模型L的预测方差为$var(L)=var\left[\frac{l_1+l_2+l_3}{3}\right]=\frac{\sigma^2}{3}$
- 集成模型的预测方差仅为弱回归器预测方差的1/3。因此通过$Bagging$集成策略可以有效降低模型输出预测的方差

Bagging集成策略

- **例题2：**对于某产品的二分类任务，表所示数据为该产品样本数据集D，其中X表示产品的某个属性，y表示类标号（1或-1）。令α为分类阈值，分类器通过比较X与α值的大小进行分类。试用 Bagging 集成学习方法生成 5 个弱分类器并将它们进行组合构建集成成模型$L(X)$，并对$L(X)$的分类误差进行分析

表 某产品样本数据集D

编号	1	2	3	4	5	6	7	8	9	10
X	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
y	1	1	1	−1	−1	−1	−1	−1	1	1

Bagging集成策略

编号	1	2	3	4	5	6	7	8	9	10
X	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
y	1	1	1	−1	−1	−1	−1	−1	1	1

- 依题意对所示样本数据集D进行 5 次自助法随机采样，获得如表所示的 5 个训练样本子集D_1, D_2, D_3, D_4, D_5

编号	1	2	3	4	5	6	7	8	9	10
X	0.1	0.4	0.5	0.6	0.6	0.7	0.8	0.8	0.9	0.9
y	1	−1	−1	−1	−1	−1	−1	−1	1	1

X	0.1	0.2	0.3	0.4	0.5	0.8	0.9	1	1	1
y	1	1	1	−1	−1	−1	1	1	1	1

X	0.1	0.2	0.3	0.4	0.4	0.5	0.7	0.7	0.8	0.9
y	1	1	1	−1	−1	−1	−1	−1	−1	1

X	0.1	0.1	0.2	0.5	0.6	0.7	0.7	0.8	0.9	0.9
y	1	1	1	−1	−1	−1	−1	−1	1	1

X	0.1	0.1	0.2	0.5	0.6	0.6	0.6	1	1	1
y	1	1	1	−1	−1	−1	−1	1	1	1

Bagging集成策略

编号	1	2	3	4	5	6	7	8	9	10
X	0.1	0.4	0.5	0.6	0.6	0.7	0.8	0.8	0.9	0.9
y	1	−1	−1	−1	−1	−1	−1	−1	1	1
X	0.1	0.2	0.3	0.4	0.5	0.8	0.9	1	1	1
y	1	1	1	−1	−1	−1	1	1	1	1
X	0.1	0.2	0.3	0.4	0.4	0.5	0.7	0.7	0.8	0.9
y	1	1	1	−1	−1	−1	−1	−1	−1	1
X	0.1	0.1	0.2	0.5	0.6	0.7	0.7	0.8	0.9	0.9
y	1	1	1	−1	−1	−1	−1	−1	1	1
X	0.1	0.1	0.2	0.5	0.6	0.6	0.6	1	1	1
y	1	1	1	−1	−1	−1	−1	1	1	1

- 分别对样本子集D_1, D_2, D_3, D_4, D_5构造出相应的弱分类器L_1, L_2, L_3, L_4, L_5。不难得到这些弱分类器的具体表达式如下

$$L_1(X) = \begin{cases} -1, X \le 0.75 \\ 1, X > 0.75 \end{cases} \quad L_2(X) = \begin{cases} -1, X \le 0.65 \\ 1, X > 0.65 \end{cases} \quad L_3(X) = \begin{cases} 1, X \le 0.35 \\ -1, X > 0.35 \end{cases}$$

$$L_4(X) = \begin{cases} 1, X \le 1 \\ -1, X > 1 \end{cases} \quad L_5(X) = \begin{cases} 1, X \le 0.4 \\ -1, X > 0.4 \end{cases}$$

- 令C_1, C_2, C_3, C_4, C_5分别表示弱分类器L_1, L_2, L_3, L_4, L_5的分类准确率，则对于表所示样本数据集D，不难得到C_1= 70%，C_2= 60%, C_3= 90%、C_4 = 50%、C_5= 70%

Bagging集成策略

编号	1	2	3	4	5	6	7	8	9	10
X	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
y	1	1	1	−1	−1	−1	−1	−1	1	1

- 使用相对多数投票法将弱分类器C_1, C_2, C_3, C_4, C_5的分类结果进行融合，得到表所示集成模型分类结果。通过对比表所示预测类别与实际类别，可知 Bagging 集成学习获得集成分类器L具有90%的分类准确

表 Bagging集成模型L的分类结果

	1	2	3	4	5	6	7	8	9	10
类别求和	1	1	1	-1	-3	-3	-1	1	1	1
预测类别	1	1	1	-1	-1	-1	-1	1	1	1
实际类别	1	1	1	-1	-1	-1	-1	-1	1	1

本节目录

随机森林

- **模型结构**
  - 决策树是一类简单有效的常用监督学习模型
  - Bagging集成学习方法将多个决策树模型作为弱学习器集成起来，构建一个较强泛化性能的森林模型作为强学习器
  - 称由这些决策树作为弱学习器组合而成的森林模型为随机森林模型，通常简称为随机森林

随机森林

- **模型结构**
  - 下图表示由某个贷款数据集通过随机性自助采样方式构造而成三个决策树模型。这三个决策树模型的结构有一定差异，对新客户是否会拖欠贷款的预测也有所不同

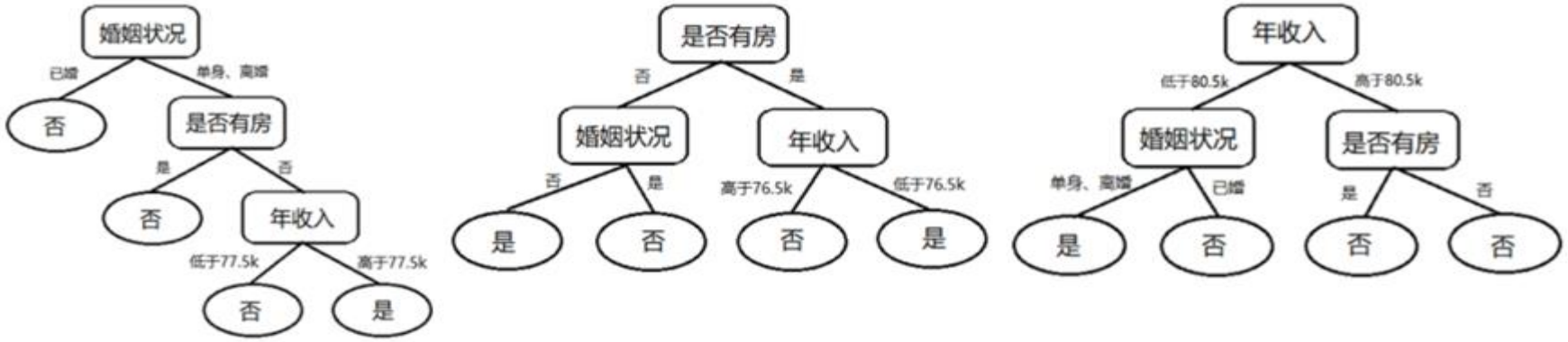


图　三个作为弱学习器的决策树

随机森林

- **模型结构**
  - 可用相对多数投票法将这三个决策树模型作为弱学习器进行集成，构建一个如图所示具有更高预测性能的随机森林模型

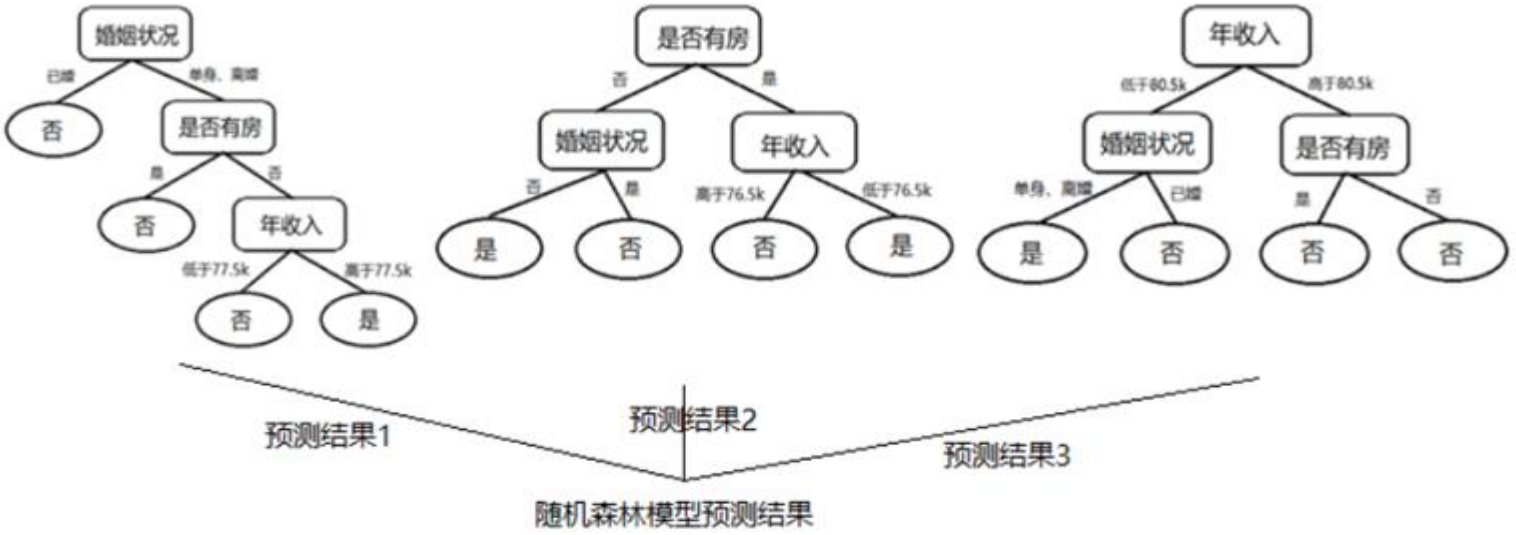
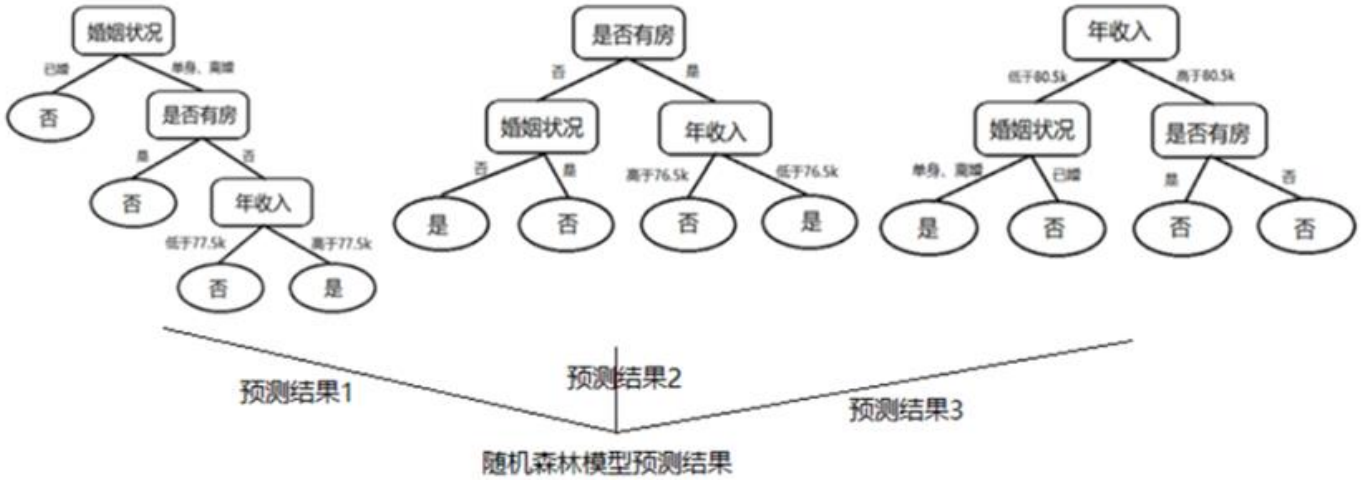


图　　随机森林模型

随机森林

- **模型结构**
  - 对于输入样本$X = \{$婚姻状况 = 单身，是否有房 = 有，年收入 = 67.2k$\}$，图所示随机森林模型对该样本的预测输出应为“是”，表示该客户可能会拖欠贷款。这是由于尽管图中最左侧决策树对该样本的预测值为“否”，但其它两棵决策树对该样本的预测值均为“是”，故根据相对多数投票法可得随机森林模型的预测输出为“是”



随机森林

- **模型结构**
  - 随机森林模型在Bagging集成策略基础上进一步增加了弱学习器之间的差异性（决策树学习过程中引入随机性），这使得随机森林模型能有效解决许多实际问题

随机森林

- **学习算法**
  - 随机森林模型基于 Bagging 集成学习方法构建，故训练构造随机森林模型过程基本上遵从 Bagging 集成学习的基本流程
  - 具体地说，对于一个包含n个样本的数据集D，首先对D做k次随机性自助采样l个训练样本子集$D_1, D_2, \cdots, D_k$，然后分别由$D_1, D_2, \cdots, D_k$训练构造k棵决策树并这些决策树进行组合便可得到随机森林模型

随机森林

- **学习算法**
  - 与基本 Bagging 集成学习不同的是，随机森林训练算法通过在决策树的构造环节随机性进一步提升弱学习器的个体差异性，使得生成的随机森林模型具有更好的泛化性能
  - 以使用训练样本子集D_i构造第i棵决策树T_i为例介绍作为弱学习器的决策树模型具体构造过程
  - 假设在确定决策树 T_i中某个结点的划分属性时，该结点所对应样本特征属性集合为$A_i = \{a_{i1}, a_{i2}, \cdots, a_{im}\}$
  - 则可使用某个度量指标通过比较该特征集合上各属性指标值的方式确定决策树节点的划分属性

随机森林

- **例题：**下表是一个感冒诊断样本数据集，试用该数据集构造一颗作为随机森林弱学习器的 CART 决策树，在确定某结点的划分属性时，若该结点所对应属性集合具有m个特征，则规定从中随机选择$s = [\log_2 m]$个属性计算用于确定划分属性的基尼指数

编号	体温	流鼻涕	肌肉疼	头疼	感冒
1	较高	是	是	否	是
2	非常高	否	否	否	否
3	非常高	是	否	是	是
4	正常	是	是	是	是
5	正常	否	否	是	否
6	较高	是	否	否	是
7	较高	是	否	是	是
8	非常高	是	是	否	是
9	较高	否	是	是	是
10	正常	是	否	否	否
11	正常	是	否	是	是
12	正常	否	是	是	是
13	较高	否	否	否	否
14	非常高	否	是	否	是
15	非常高	否	是	否	是
16	较高	否	否	是	是

随机森林

编号	体温	流鼻涕	肌肉疼	头疼	感冒
1	较高	是	是	否	是
2	非常高	否	否	否	否
3	非常高	是	否	是	是
4	正常	是	是	是	是
5	正常	否	否	是	否
6	较高	是	否	否	是
7	较高	是	否	是	是
8	非常高	是	是	否	是
9	较高	否	是	是	是
10	正常	是	否	否	否
11	正常	是	否	是	是
12	正常	否	是	是	是
13	较高	否	否	否	否
14	非常高	否	是	否	是
15	非常高	否	是	否	是
16	较高	否	否	是	是

- 表中有 4 个属性，即$m = 4$。故从中随机选择$s = [\log_2 4] = 2$个属性用于计算确定该决策树第一个结点的划分属性。通过随机抽样，选择“流鼻涕”和“肌肉疼”这两个属性进行计算。首先考察“流鼻涕”属性，根据是否流鼻涕可以将数据集划分为
$D_1 = \{1,3,4,6,7,8,10,11\}$; $D_2 = \{2,5,9,12,13,14,15,16\}$

- 分别计算D_1和 D_2的基尼指数：$Gini(D_1) = 1 - \left(\frac{7}{8}\right)^2 - \left(\frac{1}{8}\right)^2 = 0.21875$，$Gini(D_2) = 1 - \left(\frac{5}{8}\right)^2 - \left(\frac{3}{8}\right)^2 = 0.46875$

随机森林

编号	体温	流鼻涕	肌肉疼	头疼	感冒
1	较高	是	是	否	是
2	非常高	否	否	否	否
3	非常高	是	否	是	是
4	正常	是	是	是	是
5	正常	否	否	是	否
6	较高	是	否	否	是
7	较高	是	否	是	是
8	非常高	是	是	否	是
9	较高	否	是	是	是
10	正常	是	否	否	否
11	正常	是	否	是	是
12	正常	否	是	是	是
13	较高	否	否	否	否
14	非常高	否	是	否	是
15	非常高	否	是	否	是
16	较高	否	否	是	是

- 在使用“流鼻涕”这一属性对集合D进行划分时，得到的基尼指数为

$$Gini(D,\text{流鼻涕}) = \frac{8}{16} \times Gini(D_1) + \frac{8}{16} \times Gini(D_2) = 0.34375$$

- 再考察属性“肌肉疼”，根据肌肉是否疼痛可以将数据集划分为

$$D_1 = \{1,4,8,9,12,14,15\}; \quad D_2 = \{2,3,5,6,7,10,11,13,16\}$$

- 分别计算D_1和D_2的基尼指数：

$$Gini(D_1) = 1 - \left(\frac{7}{7}\right)^2 - \left(\frac{0}{7}\right)^2 = 0，\ Gini(D_2) = 1 - \left(\frac{4}{9}\right)^2 - \left(\frac{5}{9}\right)^2 = 0.4938$$

- 在使用“肌肉疼”这一属性对集合D进行划分时，得到的基尼指数为

$$Gini(D,\text{肌肉疼}) = \frac{7}{16} \times Gini(D_1) + \frac{9}{16} \times Gini(D_2) = 0.2778$$

随机森林

编号	体温	流鼻涕	肌肉疼	头疼	感冒
1	较高	是	是	否	是
2	非常高	否	否	否	否
3	非常高	是	否	是	是
4	正常	是	是	是	是
5	正常	否	否	是	否
6	较高	是	否	否	是
7	较高	是	否	是	是
8	非常高	是	是	否	是
9	较高	否	是	是	是
10	正常	是	否	否	否
11	正常	是	否	是	是
12	正常	否	是	是	是
13	较高	否	否	否	否
14	非常高	否	是	否	是
15	非常高	否	是	否	是
16	较高	否	否	是	是

- 根据上述计算结果，选择“肌肉疼”作为决策树根节点的划分属性，得到如图所示的初始决策树，其左右叶子节点所对应的数据子集分别为

$$D_1 = \{1,4,8,9,12,14,15\}; \quad D_2 = \{2,3,5,6,7,10,11,13,16\}$$

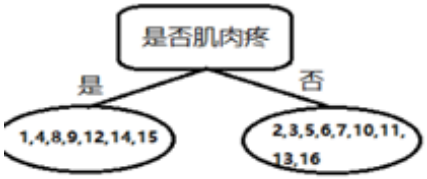


图　初始决策树

图　更新后的决策树

总结

- **集成学习**
  - 概念
  - 基本范式
  - 泛化策略
- **Bagging集成策略**
  - 自助采样法
- **随机森林模型**
  - 模型结构
  - 学习算法

思考题

- 试分析Bagging算法为何难以提升朴素贝叶斯分类器的性能
- 试分析随机森林为何比决策树Bagging集成的训练速度更快

练习题

- 试编程实现Bagging，以决策树桩为基学习器，在西瓜数据集3.0α上训练一个Bagging集成