

# 分类模型

授课教师：吴翔  
邮箱：wuhsiang@hust.edu.cn

1 统计学习概述

2 基本分类模型

3 聚类模型

4 树模型

## Section 1

# 统计学习概述

# 医疗健康领域中的统计学习

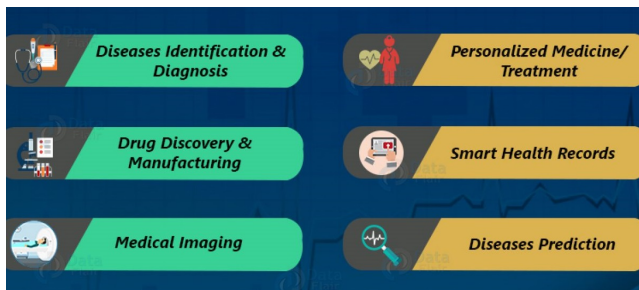


图 1: statistical learning in healthcare

## 通用模型

因变量 (响应变量, response variable) 记为  $y$ , 自变量 (预测变量, predictors) 集合记作向量  $\mathbf{x} = c(x_1, x_2, \dots, x_p)$ , 则通用模型可以写成:

$$y = \underbrace{f(\mathbf{x})}_{\text{Prediction}} + \underbrace{\epsilon}_{\text{Error}} .$$

从而由

$$\hat{y} = \hat{f}(\mathbf{x})$$

得到预测值。研究者可以选择不同的建模方法, 从而得到对  $f(\cdot)$  的不同估计  $\hat{f}(\cdot)$ 。

# 理解视角

将观测值  $y$  分为结构部分  $f(\mathbf{x})$  和随机部分  $\epsilon$ , 可以从**三个视角**来理解:

- **因果性** (计量经济领域): 观测项 = 机制项 + 干扰项
- **预测性** (统计学习领域): 观测项 = 预测项 + 误差项
- **描述性** (统计领域): 观测项 = 概括项 + 残差项

课堂讨论: (1) 三个视角在研究目的上有何区别? (2) 在建模思路和模型选取上有何差异?

# 模型误差

本质上，统计学习指一套估计  $f(\cdot)$  的方法。为此，我们需要了解在估计  $f(\cdot)$  中涉及的关键理论概念，以及评估准则。

给定  $\hat{f}(\mathbf{x})$  和  $\mathbf{x}$ ，那么：

$$E(y - \hat{y})^2 = E[f(\mathbf{x}) + \epsilon - \hat{f}(\mathbf{x})]^2 = \underbrace{[f(\mathbf{x}) - \hat{f}(\mathbf{x})]^2}_{\text{Reducible}} + \underbrace{\text{Var}(\epsilon)}_{\text{Irreducible}} .$$

第一项是**可约误差** (reducible error)，若使用更适当的方式估计  $f(\cdot)$ ，则可以减少可约误差；第二项是**不可约误差** (irreducible error)，它是由未被测量的因素导致的，因而不可消除。

# 自由度

模型自由度，表征了模型的复杂程度，是研究者在估计  $f(\cdot)$  时的重要考量因素。

在统计学习中，尤其需要理解模型的自由度。

*The number of **degrees of freedom** is the number of values in the final calculation of a statistic that are free to vary.*

— *In Statistics*

*The **degrees of freedom** are an accounting of how many parameters are estimated by the model and, by extension, a measure of complexity for linear regression models.*

— *In Statistical Learning*



## 自由度分解

假定多元线性模型  $y = \beta X + \epsilon$  中, 假定  $X$  包括一列常数和  $(p - 1)$  列变量, 那么待估计的参数个数为  $p$ , 方差和自由度的分解如下:

- SST: 自由度为  $n - 1$
- SSE: 自由度为  $n - p$
- SSR: 自由度为  $p - 1$

因而, 自由度的分解为:

$$n - 1 = (n - p) + (p - 1)$$

线性回归模型中, 模型的自由度等于预测变量的个数。

**课堂思考:** 假设模型有两个解释变量, 其中  $x_1$  是连续变量,  $x_2$  是包含 5 个分类的分类变量, SSR 的自由度为多少?

# 方差分析表

变异来源	平方和	自由度	均方
回归模型	SSR	$p - 1$	$MSR = SSR / (p - 1)$
误差	SSE	$n - p$	$MSE = SSE / (n - p)$
总变异	SST	$n - 1$	$MST = SST / (n - 1)$

假定在线性回归模型  $A$  的基础上, 加了几个变量得到模型  $B$ , 模型选择取决于构造的  $F$  检验:

$$F(\Delta df, df_{SSE}) = \frac{\Delta SSR / \Delta df}{MSE_B} ? > F_{\alpha}$$

# 预测精度 vs 可解释性

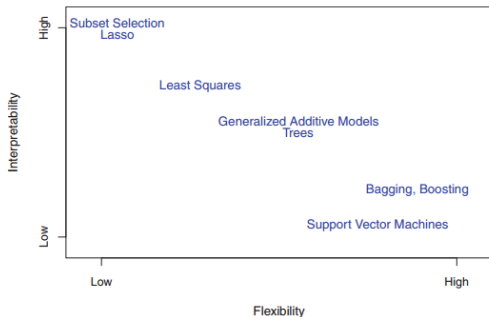
Q1: 如何选择函数  $f(\cdot)$ ?

随着模型自由度 (degree of freedom) 增加, 模型变得更加复杂。

- 预测精度 (accuracy): 尽可能减少可约误差, 因此要求自由度更大的模型。
- 可解释性 (interpretability): 尽可能用少数变量来解释  $\mathbf{x}$  如何影响  $y$ , 因此要求自由度更小的模型。

统计学习中, 大多数时候更加关注预测精度, 因而可以将  $f(\cdot)$  视作黑箱。

# 权衡预测精度与模型可解释性



**FIGURE 2.7.** A representation of the tradeoff between flexibility and interpretability, using different statistical learning methods. In general, as the flexibility of a method increases, its interpretability decreases.

图 2: Widely-used models

## 参数方法 vs 非参数方法

Q2: 函数  $f(\cdot)$  的形式, 是否有明确假设? 或说, 给定训练集 (training set) 数据, 如何估计函数  $f(\cdot)$ ?

参数方法 (parametric methods):

- 步骤: (1) 设定具体的函数形式, 包括线性或非线性函数; (2) 使用训练集数据, 拟合 (fit) 或说训练 (train) 模型, 得到参数估计值。
- 优点: 简化了  $f(\cdot)$  的估计问题, 估计一组参数通常很方便。
- 缺点: 一旦模型设定有误, 则会导致较大误差。

非参数方法 (non-parametric methods)

- 步骤: 使用附近的观测值来估计给定  $x$  时的预测值。
- 优点: 避免设定特定的函数形式, 从而规避了模型设定错误。
- 缺点: 无法将估计  $f(\cdot)$  这一问题变成少量参数的估计, 因而远远超过参数方法需要的观测值才能获得  $f(\cdot)$  的准确估计。

# 监督学习 vs 无监督学习

Q3: 有无已知的输出结果（即响应变量  $y$ ）作为参考？

- **监督学习** (supervised learning): 有已知的输出结果（当然也有输入结果）作为参考，即为训练集。
- **无监督学习** (unsupervised learning): 无已知的输出结果（仅有输入结果）作为参考。例如，市场细分研究中的**聚类分析**，依据消费者特征将其分为不同的细分市场。

# 回归问题 vs 分类问题

Q4: 响应变量  $y$  是离散还是连续的?

- **回归问题** (regression), 或说预测问题 (prediction): 响应变量  $y$  是连续的。
- **分类问题** (classification): 响应变量  $y$  是离散的。

因而, 二者在本质上并无太大差异。

# 统计学习方法

统计机器学习 (statistical machine learning) 可分为:

- 监督学习 (supervised learning) vs 无监督学习 (unsupervised learning): 聚类分析即为典型的无监督学习
- 参数方法 (parametric methods) vs 非参数方法 (non-parametric methods)
- 回归 (regression) 问题 vs 分类 (classification) 问题: 分别针对连续变量和分类变量



## 测试均方误差的分解

**测试集** (test set) 均方误差的期望值 (**expected test MSE**) 可以分解为如下三个部分:

$$E(y - \hat{f}(x))^2 = \underbrace{\text{Var}(\hat{f}(x))}_{\text{Variance}} + \underbrace{[\text{Bias}(\hat{f}(x))]^2}_{\text{Bias}} + \underbrace{\text{Var}(\epsilon)}_{\text{Irreducible}} .$$

- 模型方差 (variance): 针对不同的训练数据,  $\hat{f}$  的变化程度。
- 模型偏误 (bias): 通过相对简化的模型来**近似**真实世界的问题时所引入的误差。

## 模型复杂程度

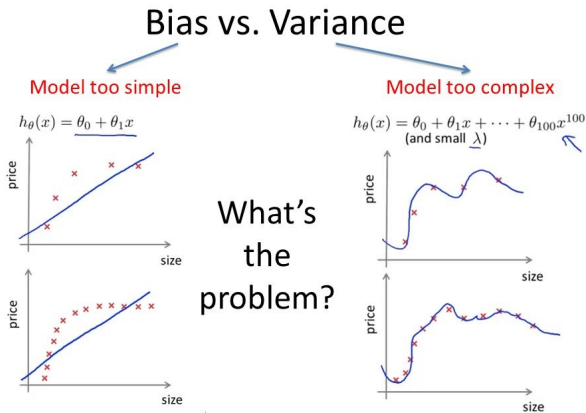


图 3: Model complexity

# 权衡模型偏误与方差

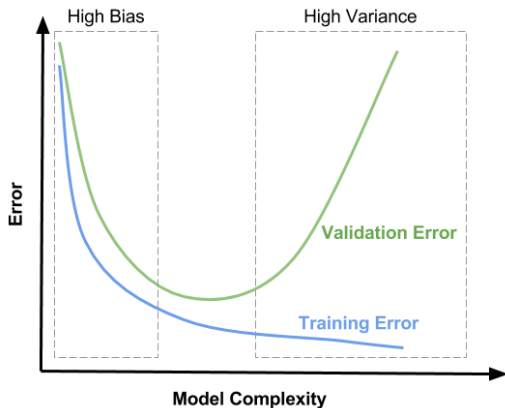


图 4: bias-variance trade-off

## 如何选择统计模型？

- 传统统计模型的局限：线性回归模型等统计模型通常最小化训练数据的均方误差，但是其测试均方误差（**test MSE**）却较大。换言之，传统统计模型执着于寻求“真实规律”，以致于将一些随机因素**误判**为  $f$  的真实性质。
- 权衡模型偏误与方差（**bias-variance trade-off**）：随着模型灵活性（或自由度）的增加，模型方差随之增大，但模型偏误则相应减小（过度拟合问题）。通过交叉验证来权衡两者。
- 权衡预测精度与可解释性（**accuracy-interpretability trade-off**）：诸如 bagging、boosting、support vector machines 等非线性模型具有很高的预测精度，但不易解释；linear models 等易于解释，但预测精度不高。两者的权衡取决于研究目的。

# 交叉验证

**交叉验证** (cross-validation) 将原始数据集分为训练集 (**training set**) 和验证集 (**validation set**), 并以验证集的错误率选择最佳模型。

- 留一交叉验证法 (leave-one-out cross validation, LOOCV)
- $k$  折交叉验证法 ( $k$ -fold CV): 将观测集随机分为  $k$  个大小基本一致的组, 或说折 (fold)。每次选取其中一折作为验证集, 而剩余  $k - 1$  折作为训练集。通常, 取  $k = 5$  或  $k = 10$ 。

**分类模型**验证集错误率:

$$CV_{(k)} = \frac{1}{k} \sum_{i=1}^k \text{Err}_k = \frac{1}{k} \sum_{i=1}^k \frac{1}{m_k} \sum_{i=1}^{m_k} I(y_i \neq \hat{y}_i).$$

# 分类模型概述

预测分类响应变量 (categorical response variable):

- ① 基本分类模型 (basic classifier)
- ② 树模型 (tree-based models)
- ③ 聚类模型 (clustering models)

# 分类模型的评价

		真实情况	
		患病（阳性）	正常（阴性）
		患病（阳性）	正常（阴性）
模型预测	患病（阳性）	灵敏度（TPR）	误诊率（FPR）
	正常（阴性）	漏诊率（FNR）	特异度（TNR）

图 5: confusion matrix

## 疾病筛查问题

采用乳房 X 光检查乳腺癌，得到以下混淆矩阵：

### Example: Breast Cancer Screening

	Breast Cancer		
Mammogram Results	Disease	No Disease	Total
Positive	132	983	1,115
Negative	45	63,650	63,695
Total	177	64,633	64,810



# 灵敏度

- 定义：灵敏度 (sensitivity) 也称为真阳性率、召回率 (recall rate)。指实际为阳性的样本中，被正确判断为阳性的比例。
- 疾病筛查：在患病人群中，成功检出患者的概率。
- 适用情况：用以避免假阴性。例如 HIV 的筛查。
- 结果解读：由于真阳性率高，因而假阴性低。亦即，若得到结果是阴性，则有把握认为未患病。

以上筛查技术的灵敏度为： $132 / 177 = 74.6\%$ 。

# 特异度

- 定义：**特异度** (specificity) 也称为真阴性率。指实际为阴性的样本中，被正确判断为阴性的比例。
- 疾病筛查：在未患病人群中，成功给出阴性结果的概率。
- 适用情况：用以**避免假阳性**。例如，治疗风险较大的疾病。
- 结果解读：由于真阴性率高，因而假阳性率低。亦即，若得到结果是阳性，则有把握认为患病。

以上筛查技术的特异度为： $63650 / 64633 = 98.5\%$ 。

联合筛查：先采用低成本、高灵敏度的筛查技术，排除未患病人群；再采用高成本、高特异度的筛查技术，确诊患病人群。

# 两类预测错误

(false positive)



(false negative)



图 7: Two types of error

## 误诊率与漏诊率

考虑两类预测错误，相应得到两个比率：

- **误诊率**：也称为假阳性率。指实际为阴性的样本中，被错误判断为阳性的比例。
- **漏诊率**：也称为假阴性率。指实际为阳性的样本中，被错误判断为阴性的比例。

以上筛查技术的误诊率为： $983 / 64633 = 1.5\%$ ，漏诊率为： $45 / 177 = 25.4\%$ 。

同时可以得知：

- 误诊率 =  $1 - \text{特异度}$
- 漏诊率 =  $1 - \text{灵敏度}$

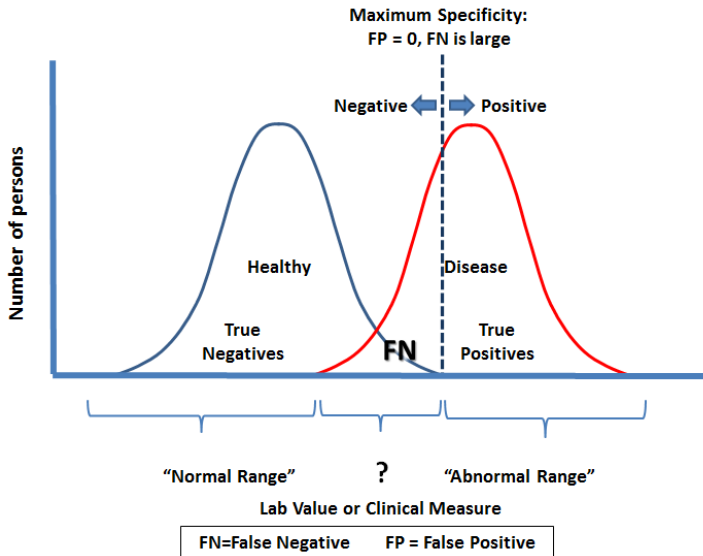
# 机场安检问题

- 方案一：
  - 措施：针对所有可疑的危险物品，均触发报警。
  - 评价：高灵敏度、低特异度。
- 方案二：
  - 措施：更复杂、成本更高的技术，尽可能方便未携带危险物品的乘客。
  - 评价：高特异度。

如何设计安检方案？

实践中的权衡：灵敏度 vs 特异度

# 权衡灵敏度和特异度



# ROC 曲线及 AUC

以疾病筛查为例，即有监督的二分类模型，模型预测结果为概率值，我们需要从中选取一个阈值来判断样本是否患病。确定阈值之后，超过此阈值定义为患病，低于此阈值定义为健康，就可以得出混淆矩阵。

**接收者操作特征曲线** (receiver operating characteristic curve, ROC 曲线) 绘制：

- 不断改变阈值，以误诊率 (FPR) 为横轴，灵敏度 (TPR) 为纵轴绘制

ROC 曲线下面积 (area under the curve of ROC, AUC) 常用于比较不同分类模型：

- $AUC = 1$ ，是完美分类器，通常不存在。
- $AUC = 0.5$ ，与随机猜测一样，模型没有预测价值。
- $0.5 < AUC < 1$ ，即是通常情形。

# 分类器及阈值

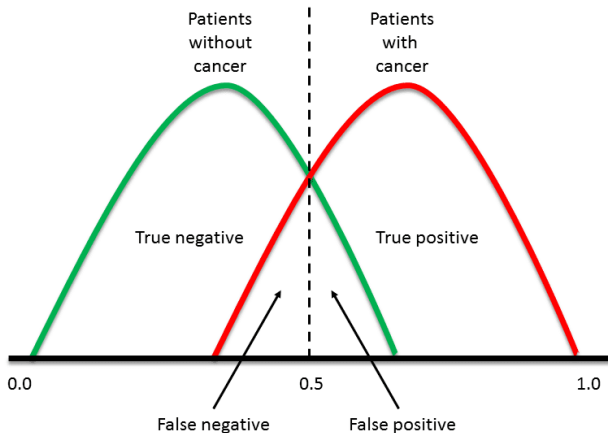


图 9: classifier and threshold



# ROC 曲线绘制

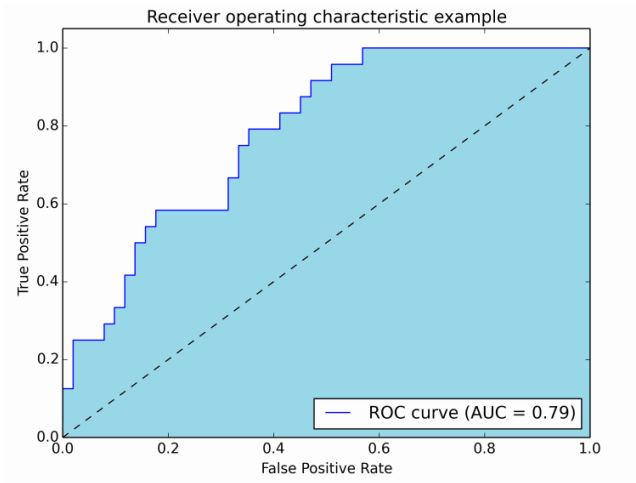


图 10: ROC curve and AUC

## Section 2

### 基本分类模型

# 基本分类模型 (basic classifier)

- ① 逻辑斯蒂回归 (logistic regression)
- ② 贝叶斯分类器 (bayes classifier)
- ③ 线性判别分析 (linear discriminant analysis, LDA)
- ④ 二次判别分析 (quadratic discriminant analysis, QDA)
- ⑤  $K$  最近邻 ( $K$ -nearest neighbor, KNN)

# 垃圾邮件分类问题

垃圾邮件分类时，需要区分垃圾邮件（spam）和正常邮件（non-spam）。这是典型的二分类问题。常用的特征包括：

- URL 和电子邮件特征：垃圾邮件中可能包含大量的链接或电子邮件地址；
- 个人信息相关词汇：垃圾邮件可能包含与个人信息相关的词汇，如“银行”、“密码”、“信用卡”等；
- 词频特征：某些词汇可能频繁出现在垃圾邮件中，例如“免费”、“优惠”、“促销”、美元或英镑等货币符号，这些词往往与垃圾邮件的内容相关联；
- 标点符号特征：垃圾邮件中可能包含大量的标点符号；
- ...

如上特征记为  $X_1, X_2, \dots, X_p$ 。

# 皮肤癌检测问题

皮肤癌检测问题，通常是指通过分析皮肤病变的图像，使用分类模型来区分良性的皮肤痣和恶性的黑色素瘤。常用的特征包括：

- 颜色特征：不同的颜色分布，如红色、黑色、棕色等，可能表明不同的皮肤病变类型；
- 形状特征：病变的形状不规则性，如边缘不整齐、不对称性，通常是恶性病变的迹象；
- 纹理特征：皮肤的纹理变化，如粗糙、鳞状或光滑，可以提供有关病变的重要信息；
- 大小和面积：病变的大小和面积可能与病变的恶性程度相关；
- 边界特征：恶性病变通常具有模糊或不清晰的边界；
- ...

这些特征可以通过图像处理技术自动提取，并用于训练分类模型。但我们依旧从如上特征  $X$  来理解分类模型。

# logistic 回归

给定  $X$  条件下事件  $Y$  发生的概率  $p(X) = \Pr(Y = 1|X)$ , 据此可以将发生比 (odd) 的对数建模为  $X$  的线性函数

$$\log\left[\frac{p(X)}{1 - p(X)}\right] = \beta X.$$

上式左侧称为对数发生比 (log-odd) 或分对数 (logit), 其取值范围在  $(-\infty, \infty)$ 。

当类别  $K \geq 2$  时, 则采用多类别 logistic 回归模型。

# 似然函数

可以通过**最大似然估计** (maximum likelihood estimation, MLE) 得到 logistic 回归的参数值。

参数记为  $\theta$ , 数据记为  $D$ 。**似然函数** (likelihood function) 是参数  $\theta$  的函数, 且定义为给定参数  $\theta$  时, 观测到数据  $D$  的概率:

$$l(\theta) = p(D|\theta).$$

例如, logistic 回归模型的似然函数

$$l(\beta) = \prod_{i=1}^n p(X_i)^{y_i} [1 - p(X_i)]^{1-y_i}.$$

# 贝叶斯定理

贝叶斯定理阐述了随机变量  $X$  和  $Y$  的条件概率之间的关系：

$$p(Y|X) = \frac{p(X, Y)}{p(X)} = \frac{p(Y) \cdot p(X|Y)}{p(X)}.$$

或从“数据-参数”的视角而言，参数  $\theta$  的**后验分布** $\pi(\theta) = p(\theta|D)$  正比于参数的**先验分布** $p(\theta)$  和**似然函数** $l(\theta)$  之积：

$$\pi(\theta) = \frac{p(\theta)p(D|\theta)}{p(D)} = \frac{p(\theta)l(\theta)}{p(D)}.$$

**课堂板书：贝叶斯定理推导及概念解释**



# 贝叶斯定理与分类

对于分类 (categorical) 响应变量  $Y$  而言, 运用贝叶斯定理:

$$p(Y = k|X = x) = \frac{p(Y = k) \cdot p(X = x|Y = k)}{p(X = x)}.$$

假定  $x$  是  $m$  维向量 (即特征数量), 简写为

$$p(C_k|x) = \frac{p(C_k) \cdot p(x|C_k)}{p(x)} \propto p(C_k) \prod_{i=1}^m p(x_i|C_k)$$

# 贝叶斯分类器

**贝叶斯分类器** (bayesian classifier) 选择后验概率  $p(C_k|x)$  最大的类别，作为分类结果，即  $\operatorname{argmax} p(C_k|x)$ 。

可以证明，贝叶斯分类器将产生最低的测试错误率，亦即**贝叶斯错误率**。相应于分类的边界，成为贝叶斯决策边界 (bayes decision boundary)。

问题在于，如何推导出后验概率  $p(C_k|x)$ ？我们需要更多**假设**。

# LDA

**线性判别分析** (linear discriminant analysis, LDA) 假定  $p(x|C_k) \sim N(\mu_k, \Sigma)$ 。  
LDA 即是条件概率  $p(x|C_k)$  为 (多元) 正态分布时的贝叶斯分类器, 其判别函数  $f(x)$  为线性函数。

考虑  $x$  是一维的情况,

$$p(x|C_k) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{1}{2\sigma^2}(x - \mu_k)^2\right],$$

由此根据后验概率  $p(C_k|x)$  的对数, 得到如下判别函数

$$f_k(x) = x \cdot \frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + \log[p(C_k)].$$

**课堂板书: 推导判别函数**

## LDA 示意图

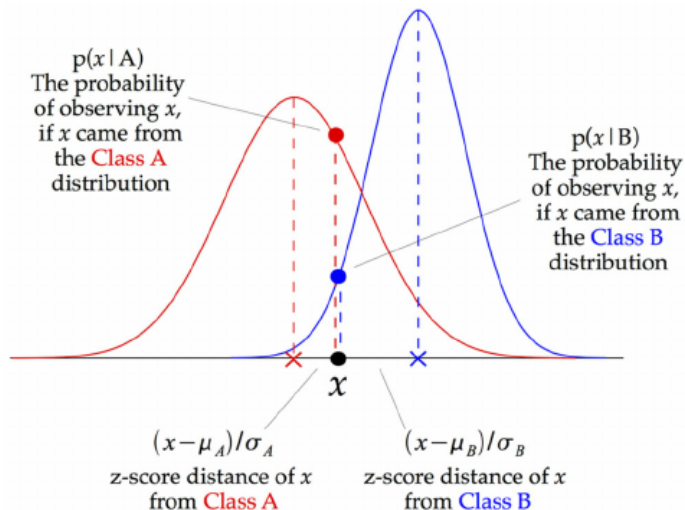


图 11: Illustration of LDA

# QDA

**二次判别分析** (quadratic discriminant analysis, QDA) 假定

$p(x|C_k) \sim N(\mu_k, \Sigma_k)$ 。QDA 即是条件概率  $p(x|C_k)$  为 (多元) 正态分布时的贝叶斯分类器, 其判别函数  $f(x)$  为二次函数。QDA 与 LDA 的**差别**在于, 协方差矩阵  $\Sigma_k$  是否假定相等。

$x$  为多维向量时, LDA 的判别函数为

$$f_k(x) = x^T \Sigma^{-1} \mu_k - \frac{1}{2} \mu_k^T \Sigma^{-1} \mu_k + \log[p(C_k)].$$

相应地, QDA 的判别函数为

$$f_k(x) = -\frac{1}{2} x^T \Sigma_k^{-1} x + x^T \Sigma_k^{-1} \mu_k - \frac{1}{2} \mu_k^T \Sigma_k^{-1} \mu_k + \log[p(C_k)].$$

# QDA 示意图

- 左图：对于两个类别，均有  $\rho(X_1, X_2) = 0.7$
- 右图：对于橙色类别， $\rho(X_1, X_2) = 0.7$ ；对于蓝色类别， $\rho(X_1, X_2) = -0.7$

## LDA versus QDA

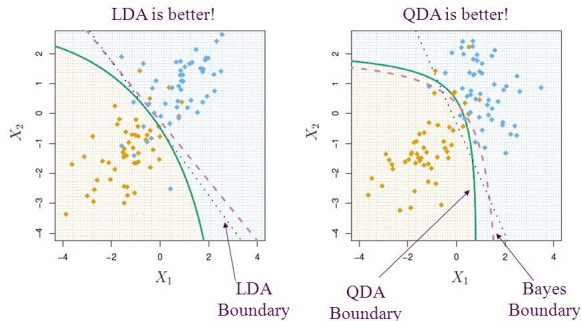


图 12: Illustration of QDA

# KNN

通常难以知道  $p(C_k|X)$  的分布。因而，可以设法估计条件分布  $p(C_k|X)$ 。

对给定正整数  $K$  和测试观测值  $x_0$ ， **$K$  最近邻** (KNN) 分类器首先识别训练集中  $K$  个最靠近  $x_0$  的点集  $A$ ，继而以集合  $A$  中的点估计条件概率：

$$p(C_j|x_0) = \frac{1}{K} \sum_{i \in A} I(y_i = j).$$

最后，运用贝叶斯规则将测试观测值  $x_0$  分到后验概率  $p(C_j|x_0)$  最大的类中。

- KNN 作为典型的**非参数方法** (non-parametric methods)，能够产生一个近似于最优贝叶斯分类器的效果
- $K$  值的选择对 KNN 分类器的效果有根本影响

## KNN 示意图

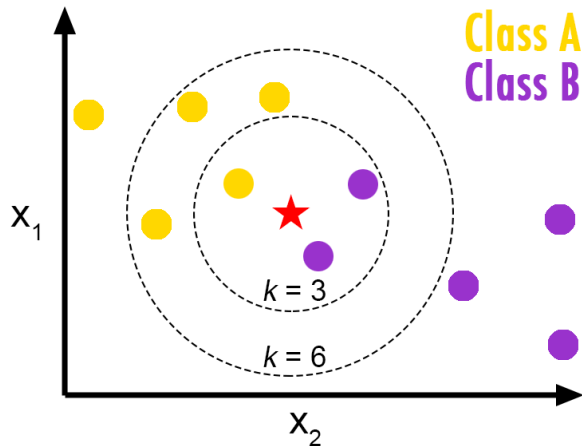


图 13: Illustration of KNN



## 模型讨论

- LDA 中的判别函数与 logistic 回归中的对数发生比 (log-odd) 均是  $x$  的线性函数, 因而二者都产生一个线性决策边界, 且分类结果相近。若  $p(x|C_k) \sim N(\mu_k, \Sigma)$  近似成立, 则 LDA 优于 logistic 回归; 反之, logistic 回归优于 LDA。
- KNN 作为非参数方法, 对决策边界的形状没有做出任何假设。因而当决策边界高度非线性时, KNN 优于 LDA 和 logistic 回归。
- QDA 是线性决策边界 (LDA 和 logistic 回归) 和非参数 KNN 方法的折衷方案, 采用了二次函数形式的决策边界。

案例: 股票市场走势预测

## Section 3

### 聚类模型

# 聚类模型 (clustering models)

聚类分析 (clustering) 试图从观测数据中寻找**同质子类**，属于**无监督学习** (unsupervised learning) 的范畴。基本聚类模型包括：

- ①  $K$  均值聚类 ( $K$ -means clustering)
- ② 层次聚类 (hierarchical clustering)

原理：将观测样本分割到不同的类 (cluster) 中，使每个类内的观测彼此相似，而不同类中的观测彼此差异很大。

**课堂讨论：比较聚类与 PCA、FA、ANOVA、线性回归**

# $K$ 均值聚类

$k$  均值聚类通过**最小化类内差异**而得到聚类结果：

$$\min \sum_{k=1}^K W(C_k).$$

$W(\cdot)$  衡量类内差异，例如可以采用欧氏距离计算。

$k$  均值聚类算法如下：

- ① 为每个观测样本随机分配一个初始类  $k(1 \leq k \leq K)$ 。
- ② 重复以下操作，直至类的重分配停止为止：
  - 分别计算  $K$  个类的中心。第  $k$  个类中心是其类内  $p$  维观测样本的均值向量。
  - 将每个观测样本分配到距离其最近的类中心所在的类中。

**课堂讨论：是否存在其它思路？**

# K 均值聚类示意图

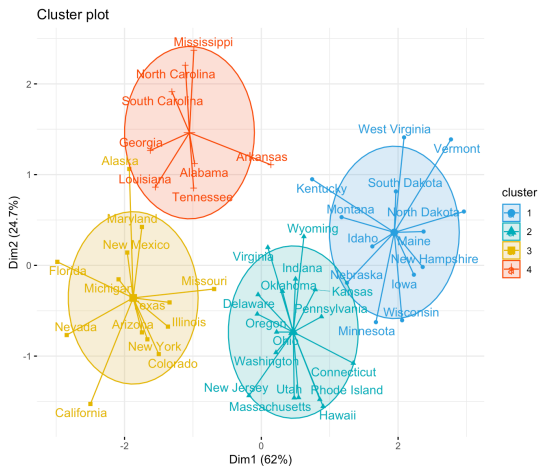


图 14: Illustration of K-means clustering

# 层次聚类

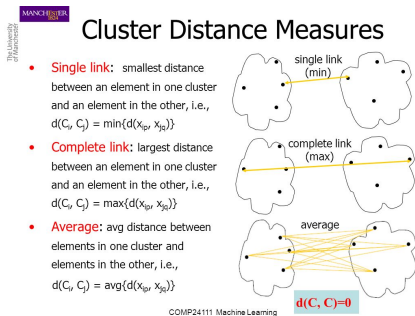
层次聚类 (hierarchical clustering) 算法如下:

- ① 每个观测样本自成一类, 共有  $n$  个初始类。计算所有  $n(n-1)/2$  对观测样本 (类) 之间的相异度。
- ② 令  $i = n, n-1, \dots, 2$ :
  - 在  $i$  个类中, 比较任意两类间的相异度, 找到相异度最小的两类, 将其合并起来。用两个类之间的相异度表示这两个类在谱系图中交汇的高度。
  - 计算剩下的  $i-1$  个新类中, 每两个类间的相异度。

层次聚类采用逐步归并的方式, 构建了谱系图 (dendrogram), 从而允许任意的类别数量。

# 距离测度

通常采用聚类来衡量相异度，常见距离形式包括：最长 (complete) 距离法、类平均法 (average)、最短 (single) 距离法和重心法 (centroid)。



5

图 15: Distance measures in hierarchical clustering

## 层次聚类示意图

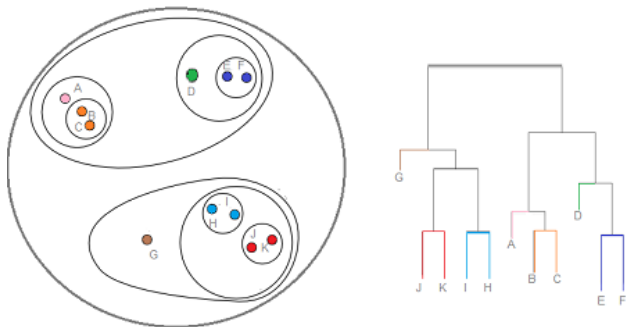


图 16: Illustration of hierarchical clustering



## Section 4

### 树模型

# 树模型 (tree-based models)

- ① 决策树 (decision tree)
- ② 装袋法 (bagging)
- ③ 随机森林 (random forest)
- ④ 提升法 (boosting)

# 决策树

- 步骤：(1) 根据分层 (stratifying) 或者分割 (segmenting) 的方式将预测变量空间划分为一系列的区域 (area); (2) 给定观测样本的预测值, 等于所属区域中训练集的平均值 (连续变量) 或者**众数** (分类变量)。
- 特征：划分预测变量空间的分割规则可以概括为一棵树。
- 适用范围：回归问题及分类问题。

## 课堂板书：与回归模型预测值比较

## 错误率的衡量

均方误差 MSE 适用于衡量回归问题的误差率。假定第  $m$  个区域第  $k$  个类别所占比例为  $p_{mk}$ ，则分类问题的错误率衡量指标包括：

- 分类错误率 (classification error rate)：此区域的训练集中非最常见类所占的比例,  $E = 1 - \max(p_{mk})$
- 基尼系数 (Gini index):  $G = \sum_{k=1}^K p_{mk}(1 - p_{mk})$
- 互熵 (cross entropy) :  $D = - \sum_{k=1}^K p_{mk} \log(p_{mk})$

后两个指标对节点纯度更加敏感。

## 预测变量空间的划分

- 形状：理论上，预测变量空间可以划分为任意形状。但为了简化模型和增强可解释性，通常划分为**高维矩阵**，亦即盒子 (box)。
- 算法：采用**递归二叉分割** (recursive binary splitting) 算法将预测变量空间划分为不同的盒子。递归二叉分割从树的根节点开始依次分割预测变量空间；每个分割点都产生两个新的分支；每一次分割空间都是**局部**最优的。
- 步骤
  - ① 考虑所有预测变量  $(X_1, \dots, X_p)$ ，从中选择预测变量  $X_j$  和分割点  $s$ ，将预测变量空间分割为  $R_1(j, s) = X|X_j < s$  和  $R_2(j, s) = X|X_j \geq s$  两个盒子，使训练集的错误率最小。
  - ② 针对新的预测变量空间，重复步骤 1，直至符合某个停止规则（例如，每个盒子观测值个数小于等于 5）为止。

课堂讨论：(1) 分类变量的处理？(2) 对比向前逐步选择 (forward stepwise selection)

## 树的剪枝

递归二叉分割采用了局部最优算法，且有可能造成数据的过度拟合，从而在测试集上预测效果不佳。我们希望选择更简单的树，从而降低模型方差。可行的解决方案包括：仅当分割使训练集的 MSE 或 Err 的减少量超过某阈值时，才分割树的节点。

可以从树  $T_0$  开始，通过剪枝 (prune) 得到子树 (subtree)。**代价复杂性剪枝** (cost complexity pruning) 在训练集的误差/错误率衡量公式中加入调整系数  $\alpha$ ，以权衡模型的精确性和复杂性：

$$\sum_{m=1}^{m_T} \sum_{i: x_i \in R_m} (y_i - \hat{y}_{R_m})^2 + \alpha |T| \text{ or } \sum_{m=1}^{m_T} \sum_{i: x_i \in R_m} [1 - \max(p_{mk})] + \alpha |T|$$

$|T|$  为树  $T$  的节点个数。最后，可以通过交叉验证选择最佳调整系数  $\alpha$ 。

**课堂讨论：比较 lasso 与代价复杂性剪枝**

## 决策树示例

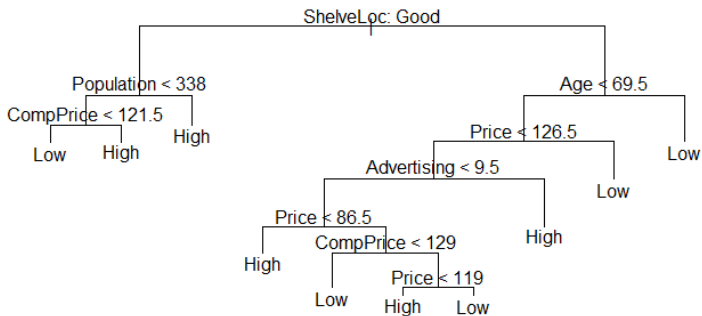


图 17: Illustration of decision tree

## 决策树的优缺点

- 优点：易于图示化，解释性较强，且更加接近人的决策模式
- 缺点：预测准确性通常低于其它回归和分类方法

因而，通常采用装袋法、随机森林、提升法等方法组合大量决策树，从而显著提高树的预测效果。



## 装袋法

决策树有着高方差 (high variance), 而自助法 (bootstrapping) 可以用以降低方差, 其原理为:  $n$  个独立观测值  $Z_1, \dots, Z_n$  的方差都为  $\sigma^2$ , 它们的均值  $\bar{Z}$  的方差为  $\sigma^2/n$ 。

装袋法 (bagging), 也称自助法聚集 (bootstrap aggregation), 从原始数据集中重抽样得到  $B$  个自助抽样训练集, 据此建立  $B$  棵回归树, 在计算相应预测值的均值 (连续变量) 或众数 (分类变量)。计算众数, 也称为**多数投票** (majority vote) 规则。当  $B$  增大到一定规模后, 就无法再降低模型误差了。因此, 只要  $B$  充分大即可。

装袋法的  $B$  颗树, 可以证明, 平均每棵树能利用约  $2/3$  的观测样本。对特定树而言, 剩余  $1/3$  的观测样本称为**袋外** (out-of-bag, OOB) 观测样本。可以用所有将第  $i$  个观测样本作为 OOB 的树来预测第  $i$  个观测样本的响应值。由此, 产生整体的 OOB 均方误差 (连续变量) 或 **OOB 分类误差** (分类变量)。实施 OOB 方法比交叉验证更为便利。

## 装袋法示意图

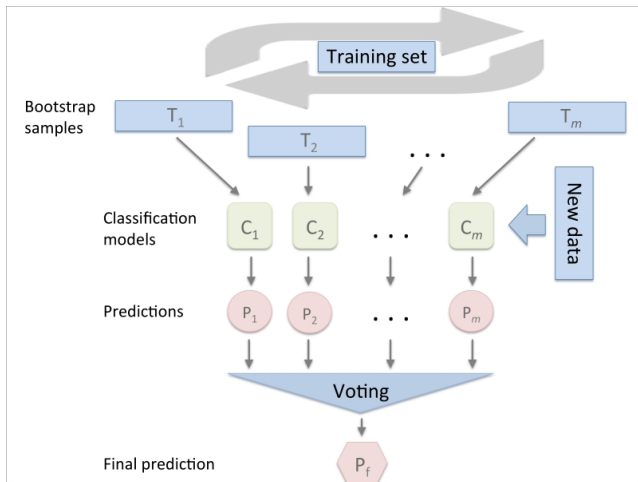


图 18: Illustration of bagging (bootstrap aggregation)

## 模型解释性

单棵决策树的结果易于解释，而采用多棵树得到的结果则难以解释。因而，装袋法以牺牲解释性为代价获得了更高的预测准确性。

可以计算**变量重要性** (variable importance)，从而获得装袋法的解释性。在装袋法建模过程中，记录下任一给定预测变量引发的分割而减少的误差量，并在所有  $B$  棵树上求平均。结果越大，表明变量越重要。

**课堂讨论：回归模型中变异的分解与解释**

# 随机森林

随机森林 (random forest) 沿袭了装袋法的思路，并进行了改进：每次分割的时候，从全部的  $p$  个预测变量中**随机**选择  $m \approx \sqrt{p}$  个预测变量实施分割。

随机森林对树**去相关** (decorrelate)，从而减少  $B$  棵树的均值或众数的方差。尤其是存在较强预测变量，或者预测变量之间相关度较高时，随机森林能有效改进装袋法。

# 随机森林示意图

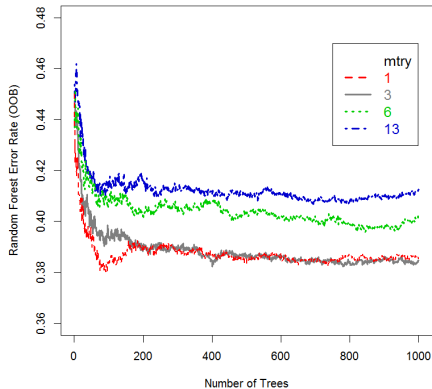


图 19: Number of trees

# 提升法

提升法 (boosting) 沿袭了装袋法的思路, 并进行了改进:  $B$  棵树按**顺序** (sequentially) 生成, 每棵树的构建都需要用到之前生成的树的信息, 采用现有模型的**残差信息**  $(X, \epsilon)$  生成决策树。回归情形下, 提升法的算法如下:

- ① 对训练集中所有观测样本  $i$ , 令  $\hat{f}(x) = 0$ ,  $\epsilon_i = y_i$ 。
- ② 对  $b = 1, 2, \dots, B$ , 重复以下过程:
  - 对训练数据  $(X, \epsilon)$  建立一棵有  $d$  个分割点 (亦即  $d + 1$  个节点) 的树  $\hat{f}^b$ 。
  - 将压缩后的新树加入模型以更新  $\hat{f}$ :  $\hat{f}(x) \leftarrow \hat{f}(x) + \lambda \hat{f}^b(x)$ 。
  - 更新残差:  $\epsilon_i \leftarrow \epsilon_i - \lambda \hat{f}^b(x_i)$ 。
- ③ 输出经过提升的模型,  $\hat{f}(x) = \sum_{b=1}^B \lambda \hat{f}^b(x)$ 。

## 提升法 (续)

提升法采用舒缓 (learning slowly) 训练模型的方法, 即利用现有残差而非原始响应变量  $y$  作为响应值。压缩参数  $\lambda$  使学习过程变得缓慢。通过充分利用之前生成的树的信息, 有效提高预测或分类准确度。

提升方法的调整参数包括:

- 树的总数  $B$ : 与装袋法和随机森林不同,  $B$  值过大时会出现过度拟合, 因而通常用交叉验证来选择  $B$  值。
- 压缩参数  $\lambda$ : 控制提升法的学习速度, 通常取极小的值, 如  $\lambda = 0.01$  或  $\lambda = 0.001$ 。压缩参数  $\lambda$  越小, 就需要越多的树, 即越大的  $B$  值。
- 每棵树的分割点数  $d$ : 控制整个提升模型的复杂程度。

提升法通常需要的树更小 ( $d$  值很小), 因为生成一棵特定的树已经考虑了其它已有的树。

## 提升法示意图

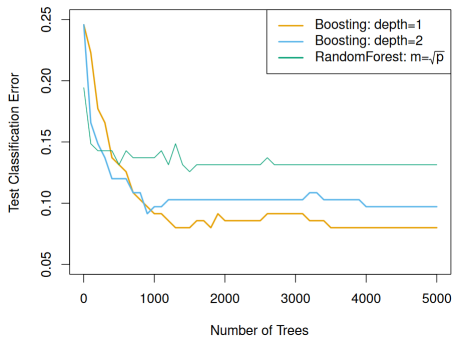


图 20: Illustration of boosting