第二章 模型评估与选择

王星 参考文献：周志华第二章

中国人民大学统计学院

March 7, 2022

噪声及其影响

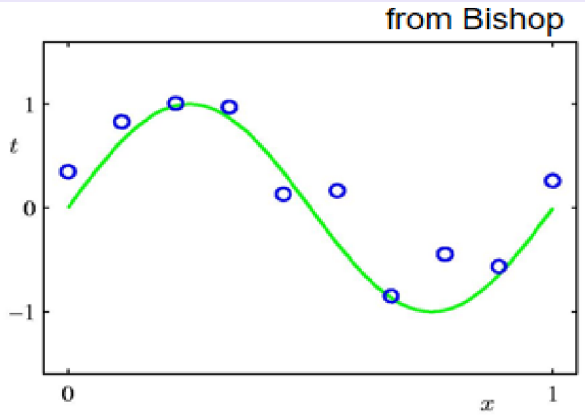
- 噪声是数据中不可避免的异常，有三种情况
  - 输入变量的数值不正确；
  - 响应变量y_i标记不准确；
  - 未见的、隐藏的模式，不能直接观测的属性没有考虑但却会对y产生影响；
- 噪声带来的后果是：学习越来越困难；
- 概念学习中设定的假设要更简单一些而不是更复杂，原因是
  - 方便使用也容易训练（少参数，速度更快）；
  - 易于解释也便于理解
  - 稳定性较好，但需要损失偏差；
- 如何训练是个难题

大纲

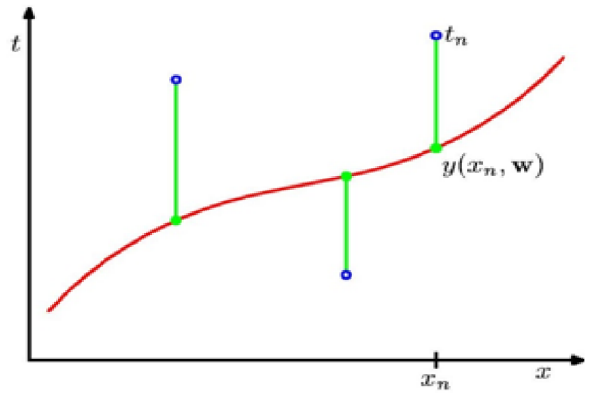
- 统计学习三要素
- 期望风险、经验风险和结构风险
- 泛化误差的几种近似计算（留出法，交叉验证和自助法）
- 性能指标（错误率、查准率，查全率，F1,ROC,AUC,代价敏感曲线CC）；
- 比较不同学习期的性能的假设检验
- 性能差异的理解：偏差与方差

一个简单的例子:拟合多项式

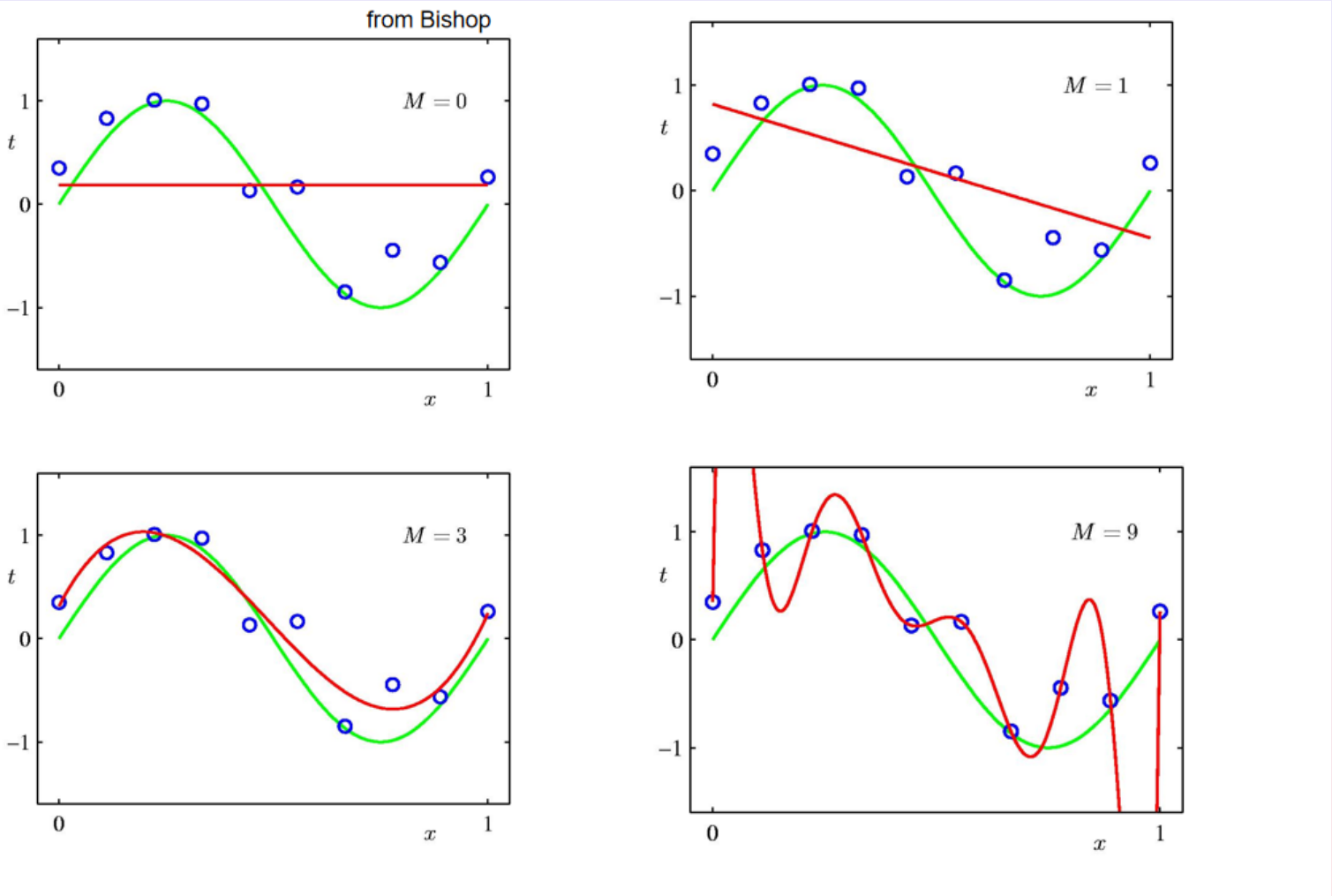
- 绿色的曲线是真实的函数，蓝色的点是从该曲线产生的带噪声的数据
- 假设数据点在x上是均匀分布的，但是在 y上有噪声.



- 用真实标记的y(x)和在x上拟合的结果之间的差异作为损失.
- 使用多项式函数作为拟合函数，把每个点在纵向方向上的拟合误差平方和相加作为整体的拟合损失.



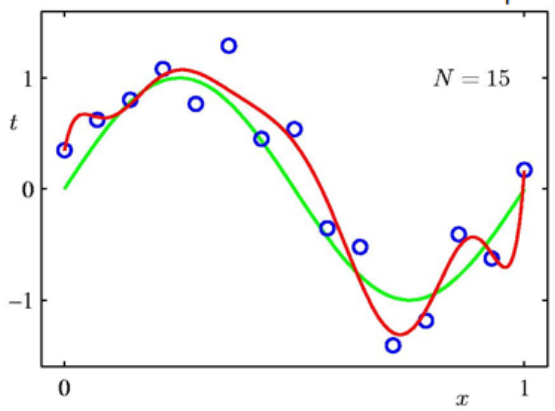
不同阶数多项式会产生不同的拟合结果



一种简单的提高模型适配度的方式是选择简单的模型

对拟合后系数过大的的项实施惩罚，就会抑制拟合曲线出现剧烈震荡

from Bishop PRML



正则后的损失函数
penalized loss function

起正则作用的参数
regularization
parameter

$$\widetilde{E}(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\{y(x_n, \mathbf{w}) - t_n\}^2 + \frac{\lambda}{2}\|\mathbf{w}\|^2$$

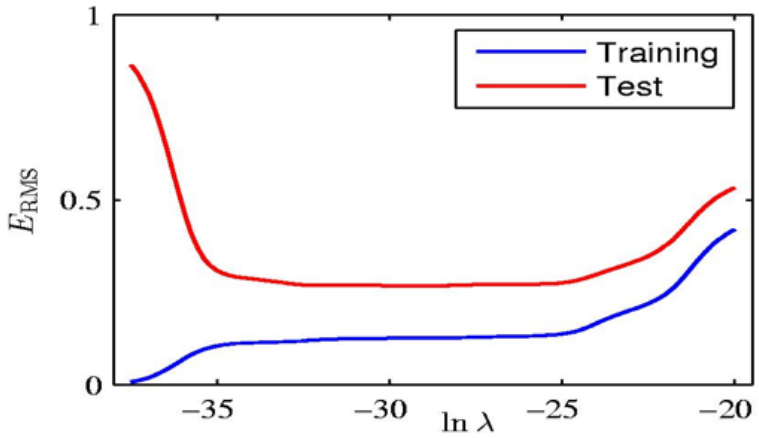
目标值target value

如何选择参数

随着多项式阶数增加，系数的估计会变大

	$\ln \lambda = -\infty$	$\ln \lambda = -18$	$\ln \lambda = 0$
$w_0^\star$	0.35	0.35	0.13
$w_1^\star$	232.37	4.74	-0.05
$w_2^\star$	-5321.83	-0.77	-0.06
$w_3^\star$	48568.31	-31.97	-0.05
$w_4^\star$	-231639.30	-3.89	-0.03
$w_5^\star$	640042.26	55.28	-0.02
$w_6^\star$	-1061800.52	41.32	-0.01
$w_7^\star$	1042400.18	-45.95	-0.00
$w_8^\star$	-557682.99	-91.53	0.00
$w_9^\star$	125201.43	72.68	0.01

正则化: E_{RMS} vs. $\ln \lambda$



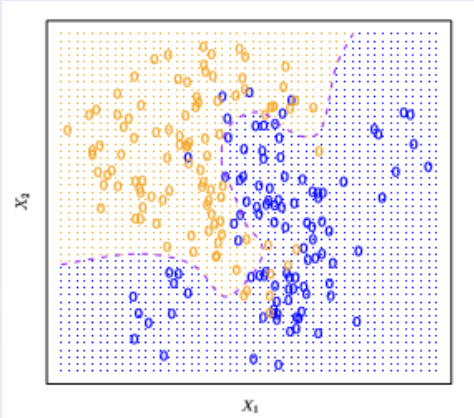
- 随着多项式阶数增大，系数通常会变大，函数表现出巨大的震荡，有着更大的M表示多项式被过分调参，多项式与目标值的随机噪声更相符;
- 当数据集规模增加的时候，过拟合问题又变得不太严重，这表明数据量越大，用来拟合数据点的模型应该是更加复杂的，这里的启发是：数据点的数量不应该小于模型可调节参数数量的若干倍，然而后面可以看到的是应要求数据量和可调参数之间的比例结构以达到理想的拟合效果是徒劳的，还不如去审视我们的模型空间是不是合适。

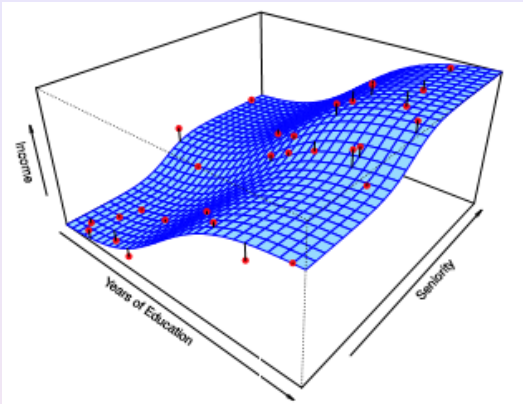
什么是统计学习?

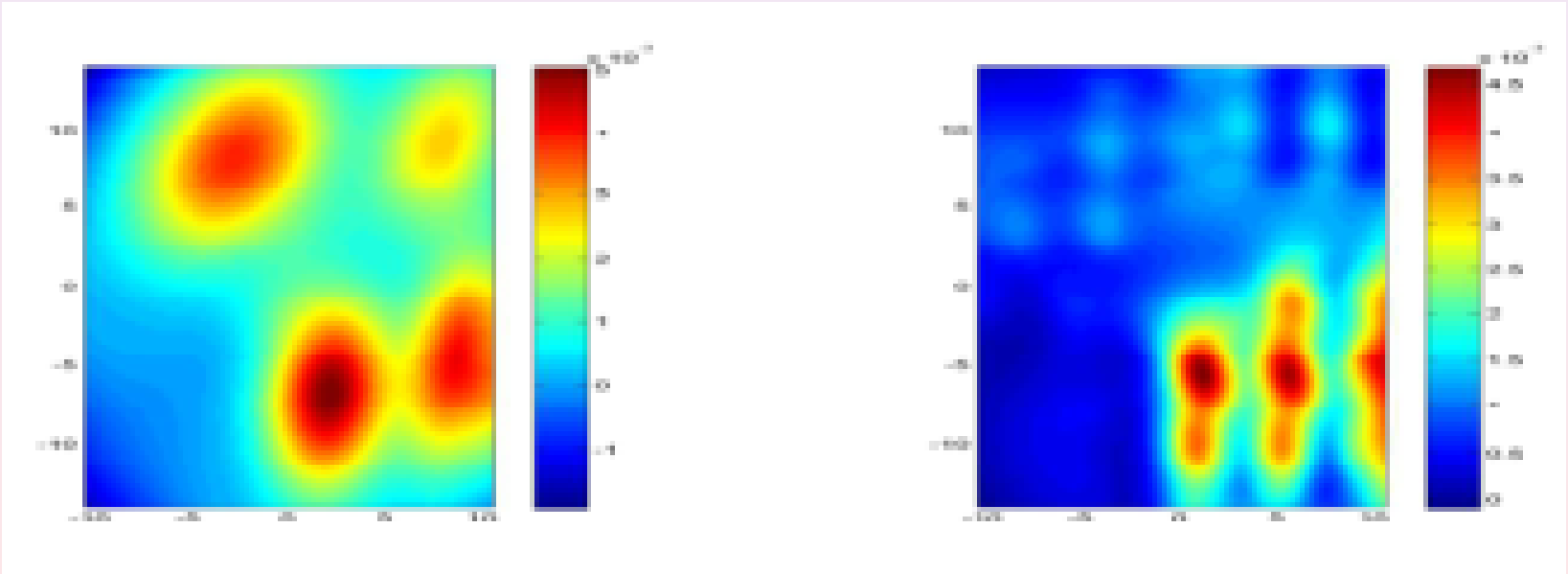
- 统计学习是关于计算机基于数据构建概率统计模型并运用模型对数据进行预测与分析的一个学科—- 李航
- 统计学习(Statistical Learning)是一套以理解数据为目的的庞大的工具集，数据是信息的载体，但不全是信息。
- 统计学习中主要分为预测（prediction）和推断(inference)
  - 预测的例子：哪些人对这个产品会有兴趣？
    - Example: direct mailing prediction: Interested in predicting how much money an individual will donate based on observations from 90,000 people on which we have recorded over 400 different characteristics.
    - Don't care too much about each individual characteristic.
    - Just want to know: For a given individual should I send out a mailing?

- 推断的例子：有兴趣的人群有怎样的特征和分布,A type of relationship between y and the $x's$.
  - Example: Which particular predictors actually affect the response?
  - 影响的方向Is the relationship positive or negative?
  - 影响的模式Is the relationship a simple linear one or is it more complicated etc.?
  - Example: housing inference:Wish to predict median house price based on 14 variables.
  - 影响的程度Probably want to understand which factors have the biggest effect on the response and how big the effect is.
  - 影响的范围For example how much impact does a river view have on the house value etc.
- 在这些例子中通常会有模式的存在性认识，随机特征的辨析，可能性的估计和参数的推断等一系列复杂的建模问题。

几个例子







统计学习三要素

- 方法=模型+策略+算法
- 模型: 假设空间中的函数，如果假设空间用$\mathcal{F}$ 表示，那么假设空间就是
  - 决策函数：
    $$\mathcal{F} = \{f|Y = f(X)\},$$
    如果模型是由参数控制的，又称为参数空间：
    $$\mathcal{F} = \{f|Y = f_\theta(X), \theta \in R^n\},$$
  - 条件概率：$\mathcal{F} = \{P|P(Y|X)\}$;或$\mathcal{F} = \{P|P_\theta(Y|X), \theta \in R^n\}$;
- 策略: 有了模型的假设空间，需要考虑按照怎样的准则来学习到最优的模型
  - 损失函数（代价函数）：在假设空间$\mathcal{F}$上选取函数f 作为决策函数，对于给定的输入，由f 产生一个相应的输出Y, 这个输出与真实的Y 之间可能一致也可能不一致，用损失函数或代价函数$L(Y,f(X))$度量错误的程度。
- 算法: 算法是指学习模型的具体计算方法，很多统计学习问题归为最优化问题，如果最优化问题有显示解析解，求解的方法比较直接，如果没有显式解，如何保证解的全局最优和高效性，是算法中的一个重要问题。

损失函数

- 统计学习中的常用的损失函数如下：
  - 0-1损失

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

  - 平方损失函数(Quadratic Loss Function):

$$L(Y,f(X)) = (Y - f(X))^2;$$

  - 绝对损失函数(Absolute Loss Function):

$$L(Y,f(X)) = |Y - f(X)|;$$

  - 对数损失函数(Logarithmic Loss Function):

$$L(Y,P(Y|X)) = -\log P(Y|X).$$

- 损失函数性质：损失函数越小，模型越好，然而模型的输入和输出(X,Y) 是随机变量，与联合分布$P(X,Y)$ 有关系，所以通常用损失函数的期望来表达：
- 风险函数(期望损失)：

$$R_{\exp}(f) = E_P[L((Y,f(X)))] = \int_{x,y} L(y,f(x))P(x,y)\mathrm{d}x\mathrm{d}y;$$

损失函数（2）

- 学习的目标是期望风险最小，也就是说：

$$f^* = \text{argmin} R_{\text{exp}}(f)$$

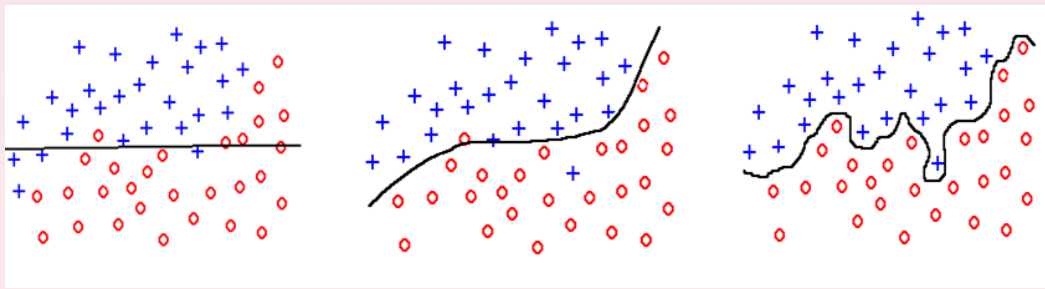
- 问题：$p(x,y)$未知。
- 给定一个训练数据$\mathcal{D} = \{(x_1,y_1),(x_2,y_2),\cdots,(x_N,y_N)\}$, 称模型$f(X)$ 关于训练数据的平均损失为经验风险（Empirical Risk）,记作：

$$R_{\text{emp}}(f) = \frac{1}{N}\sum_{i=1}^{N} L(y_i, f(x_i)).$$

损失函数（3）

[定理]x_i, y_i独立同分布的情况下，经验风险是期望风险的无偏估计

$$
\begin{aligned}
\mathrm{E}[R_{\mathrm{emp}}(f)] &= \mathrm{E}\left[\frac{1}{N}\sum_{i=1}^{N} L(y_i, f(x_i))\right] \\
&= \frac{1}{N}\mathrm{E}\sum_{i=1}^{N}\left[L(y_i, f(x_i))\right] \quad x_i\text{独立} \\
&= \frac{1}{N}\mathrm{E}\sum_{i=1}^{N} L\left[(y, f(x))\right] \quad x_i\text{同分布} \\
&= R_{\mathrm{exp}}(f)
\end{aligned}
$$



结构风险最小化

- Empirical Risk Minimization(ERM):

$$f^*(D_N) = \text{argmin}_{f\in\mathcal{F}} L(f, D_N) = \text{argmin}_{f\in\mathcal{F}} \frac{1}{N}\sum_{i=1}^{N} L(y_i, f(x_i))$$

- 训练数据总是有限的，有噪声甚至数量不足，用训练数据上的经验风险估计期望风险并不理想，需要对经验风险作一定的矫正—结构风险最小化
- Structure Risk Minimization(SRM):

$$R_{\text{srm}}(f) = \frac{1}{N}\sum_{i=1}^{N} L(y_i, f(x_i)) + \lambda J(f)$$

$J(f)$是模型的复杂度，f越简单，复杂度$J(f)$就越小，f越复杂，复杂度$J(f)$ 就越大。

$$f^*(D_N) = \text{argmin}_{f\in\mathcal{F}} L(f, D_N) = \text{argmin}_{f\in\mathcal{F}} \frac{1}{N}\sum_{i=1}^{N} L(y_i, f(x_i)) + \lambda J(f)$$

算法（续）

- 算法是指学习模型的具体算法;
- 例如：最优化学习算法，梯度学习算法，深度学习算法等;

例1

- 对于二分类问题$Y = \{\pm 0, 1\}$,
- 假设$X \times Y$的联合分布由(μ, η)表示，其中μ是X的边缘分布，η 是Y 的条件分布，$\eta(X) := P(Y = 1|X)$;
- 损失函数$L[(\hat{y}, y) = 1|y \neq \hat{y}]$,
- 于是可以计算任意一个可测函数f的风险

$$R(f) = \mathrm{E}L(f(X), Y) = P(f(X)) \neq Y).$$

- 定义R^*为最优风险，$R^* = \mathrm{argmin}_f R(f)$.
- 定义：0-1损失

$$f^*(x) = \begin{cases} 1, & \eta(X) \geq 1/2 \\ 0, & otherwise \end{cases}$$

定理 对任意的$f: X \to Y$,

$$R(f) - R(f^*) = E(1[f(X) \neq f^*(X)]|2\eta(X) - 1|)$$

证明

$$\begin{aligned}
& R(f) - R(f^*) \\
= & \mathbb{E}_{X,Y}\{(1[f(X) \neq Y]) - (1[f^*(X) \neq Y])\} \\
= & \mathbb{E}_{X,Y}\{1[f(X) \neq f^*(X)]\,([1[f(X) \neq Y] - 1[f^*(X) \neq Y]])\} \\
= & \mathbb{E}_{X,Y}\{1[f(X) \neq f^*(X)](2 \cdot 1[f^*(X) = Y] - 1)\} \\
= & \mathbb{E}_X\{1[f(X) \neq f^*(X)][2\mathbb{E}_{Y|X}1(f^*(X) = Y) - 1]\} \\
= & \mathbb{E}_X\{1[f(X) \neq f^*(X)][2(1/2 + |\eta(X) - 1/2|) - 1]\} \\
= & \mathbb{E}_X 1[f(X) \neq f^*(X)](|2\eta(X) - 1|)
\end{aligned}$$

这个定理表明在二分类问题中，最优的决策函数f^*形式是由后验概率$\eta(x)$.定义的

2.1 经验误差与过拟合

经验误差:假设学习到的模型是$Y = \hat{f}(x)$

- 训练误差(training error)：在训练集上预测的结果与训练集的真实结果的差异，也称为经验误差（empirical error）,代表学习器预测的结果与实际真实结果之间的差异：

$$R_{\text{emp}}(\hat{f}) = \frac{1}{N}\sum_{i=1}^{N} L(y_i, \hat{f}(x_i))$$

  N是样本量
- 训练错误率：分类错误的样本数(a)占训练样本总数(N)的比例$E = a/N$,称$1 - a/N$ 为精度
- 泛化误差（generalization error ）：测试集或新样本上的预测误差

$$R_{\text{test}} = \frac{1}{N_{test}}\sum_{i=1}^{N_{test}} L(y_i, \hat{f}(x_i))$$

- 很明显，要使分类器尽可能的有用，应该要让泛化误差仅可能的小。然而现实环境中，很难知道新样本是什么样的，所以我们实际能做的只有努力使经验误差最小化，但经验误差又不合适。

在泛化误差未知的情况下，如何对模型进行评估呢？

- Larson于20世纪30年代就提出：在相同的数据上训练算法和评价算法的性能将会得到“过于乐观”的结果。
- 思路是：既然没法拿到新样本来进行泛化误差计算，那么可以从训练样本中取出一部分来，假装它是一组新样本，并用这些样本计算出来的误差作为泛化误差的近似。这组选出来的样本被称为“测试集”，测试集上的误差被称为测试误差。
- 留出法（hold-out）：将数据集S分为$D = S \cup T, S \cap T = \emptyset$,$S$ 训练模型，T 拿来做测试。N_T是T的样本量，泛化误差表示为：

$$\hat{R}^{HO}_{\text{test}}(\hat{f}) = \frac{1}{N_T} \sum_{x_i \in T} L(y_i, \hat{f}(x_i))$$

- 交叉验证法（cross validation）先将数据划分为k个大小相似的互斥子集，$D = D_1, \cup, \cdots, \cup D_k, D_i \cap D_j = \emptyset, \forall i \neq j$ 分为多次Hold-out, 留p 交叉,Repeated learning-testing(RLT) 交叉验证估计和k- 折交叉。
- 自助法（bootstrapping）:从m个样本的数据集D，随机采样生成训练集D'. 用D' 去估计泛化误差。

留出法

- 要有足够的样本量，以保证训练模型的效果;
- 在划分时注意保证数据分布的一致性（如：500个样本中正例和反例的比为2:3，则在训练集和测试集中正例和反例的比也要求为2:3），只需要采用随机分层抽样即可;
- 为降低随机划分的影响，重复划分训练集和测试集，对得到的多次结果取平均作为最后的结果
- 优点：Hold-out估计法的提出打破了传统的基于相同的数据进行训练和测试的分析，避免了训练和测试数据重叠引起的过拟合。
- 缺点:Hold-out估计过分依赖于某一次数据划分，数据的划分方式直接影响着估计的精度，经验上2/3 ~ 4/5.

```
from sklearn.model_selection import train_test_split
#使用train_test_split划分训练集和测试集
train_X , test_X, train_Y ,test_Y = train_test_split(
        X, Y, test_size=0.2,random_state=0)
'''
X为原始数据的自变量，Y为原始数据因变量;
train_X, test_X是将X按照8:2划分所得;
train_Y, test_Y是将X按照8:2划分所得;
test_size是划分比例;
random_state设置是否使用随机数
```

交叉验证法1:多次Hold-out估计法

- 注意到hold-out估计依赖于数据的一次划分，容易受到数据划分随机性的影响，Geisser(1975)[1]提出了包含多次hold-out 估计平均的交叉验证方法的一个一般的表示，实现了从验证估计到交叉验证估计的过渡。因为多次数据划分导致数据之间有交叉，称为交叉验证

  [1]GEISSER S．The predictive sample reuse method with applications[J]．Journal of the American Statistical Association,1975,70(35):320-328

- 如果记$S_1, ..., S_K$为D的K个非空真子集,$T_1, ..., T_K$ 是分别对应的补集，即$S_i \cap T_i = \emptyset, S_i \cup T_i = D, \forall i = 1, ..., K$ 此时泛化误差估计为：

$$\hat{R}_{\text{test}}^{MHO}(\hat{f}) = \frac{1}{K}\sum_{k=1}^{K}\hat{R}_{test(T_k)}^{HO}$$

- 优点:方法依赖于数据的多次重复划分，可去掉划分随机性的影响(在统计分析中就是通过多次重复实验来减小方差.)
- 不足：交叉验证数据划分方式有很多种，随着样本量的增加，划分方式的组合数也在急剧地增加，现实的计算中很难穷尽，计算复杂度非常高。

交叉验证法2: 留p(leave-p-out:LPO)交叉验证估计

- 为减小多次Hold-out交叉验证中数据划分的组合数，Shao[2]提出了每次数据划分中测试样本个数都相同的留p 交叉验证

  [2]SHAO Jun．Linear model selection by cross-validation[J],Journal of the American Statistical Association, 1993，88(422):486-494．

  即训练样本容量$N_S = N - p$, 测试集容量$N_T = p$，其中$p \in 1, \cdots N-1$ 。这样，如果有K 种划分$T_1, \cdots, T_K$，测试集样本量均为p，留p 交叉验证泛化误差估计为：

$$\hat{R}_{\text{test}}^{LPO}(\hat{f}) = \frac{1}{K}\sum_{k=1}^{K}\frac{1}{p}\sum_{x_j \in T_j} L(y_j, \hat{f}(x_j))$$

- 优点：留p交叉验证相比于多次留出给出的交叉验证的划分组合数从$\sum_{p=1}^{N-1} C_N^p$ 减少到了C_N^p。
- 不足：对于较大的样本量N，组合数计算复杂度C_N^p仍然很大。
- $p = 1$时为留一(leave-one-out:LOO) 交叉验证估计。

$$\hat{R}_{\text{test}}^{LOO}(\hat{f}) = \frac{1}{N}\sum_{x_j \in D_{-j}} L(y_j, \hat{f}(x_j))$$

交叉验证法3:RLT Repeated learning-testing

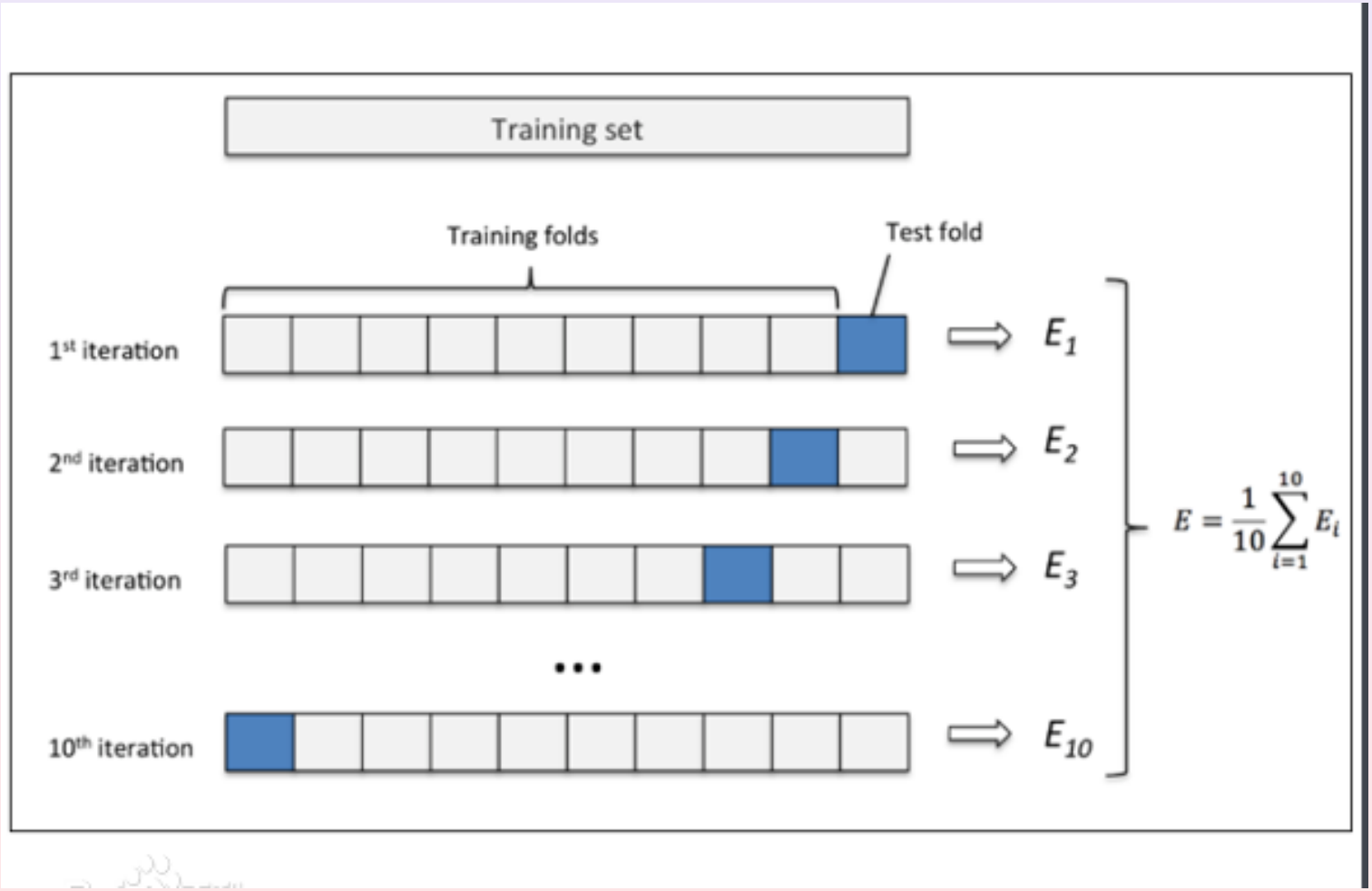
- 与留p交叉验证考虑所有数据划分相比,RLT交叉验证只基于部分数据划分进行,选择任意大于0 的K次数据划分进行泛化误差的估计.
- RLT交叉验证泛化误差估计为:

$$\hat{R}_{\text{test}}^{RLT}(\hat{f}) = \frac{1}{K}\sum_{k=1}^{K}\frac{1}{n_T}\sum_{x_j \in T_j} L(y_j, \hat{f}(x_j))$$

- 优点:RLT自从1981年被提出以后在实际应用中就被广泛使用，因为它有可接受的计算开销且操作简单。
- 缺点:K的大小的选择一直是这个方法的最大问题，不同文献中有不同的结论，如文献[5] 中建议$K = 15$, 训练和测试集比例的选择也没有一个确定的结论，往往不同的研究者使用不同的训练和测试容量。
- [3]ARLOT S,CELISSE A．A survey of cross-validation procedures for model selection[J]．Statistics Surveys,2010,4:40-79．
- [4]ZHANG Ping．Model selection via multifold cross validation[J]．Annals of Statistics,1993,21(1):299-313．
- [5]NADEAU C，BENGIO Y．Inference for the generalization error[J]．Machine Learning,2003,52(3):239-281．

交叉验证法4: K-折交叉验证估计

将D划分为互斥的k个子集，每次训练用其中$(k-1)$个数据，用剩下的一个作为测试集。这样就能获得k组训练/测试集，对这k组分别进行训练和测试，最终返回k个测试结果的均值



自助法

从m个样本的数据集D，随机采样生成训练集D'.用D'去顾及泛化误差。样本在m次采样中始终不被采集到的概率如下：

$$\lim_{m\to\infty}(1-1/m)^m = e = 0.368.$$

自助法在数据集较小、难以有效划分训练/测试集时很有用,自助法产生的数据集改变了初始数据集的分布，这会引入估计偏差。因此，在初始数据量足够时，留出法和交叉验证法更常用一些

2022.3.9课后作业

统计学习导论第5题，第6题；对于数据Auto,

- (a)将数据集按照留出法分为训练集(比例0.8)和测试集(比例0.2)，对于函数$mpg = 40 - 0.15 \times horsepower$, 编写程序估计平方损失下的泛化误差,自行设定试验次数，进行泛化;
- (b)将数据集按照留p交叉验证的方式提取测试集，$K = 20$划分采取无放回抽样$p = 10$,对于函数$mpg = 40 - 0.15 \times horsepower$, 编写程序计算平方损失下的泛化误差;
- (c)建立一个二元变量，每加仑里程量（mpg01），1表示每加仑汽油该型号车所跑里程数（mpg）在3/4 分位数以上，0 表示每加仑汽油该型号车所跑里程数（mpg）在3/4分位数以下。将数据集按照留出法随机分为训练集(比例0.8)和测试集(比例0.2)，对于函数$P(mpg01 = 1) = \frac{\exp\{3.85-0.01\times weight\}}{1+\exp\{3.85-0.01\times weight\}}$ 编写程序估计平方损失下的泛化误差;
- (d)将数据集按照分层等比例分按照训练集(比例0.8)和测试集(比例0.2)，对于函数$P(mpg01 = 1) = \frac{\exp\{3.85-0.01\times weight\}}{1+\exp\{3.85-0.01\times weight\}}$ 编写程序估计平方损失下的泛化误差;
- (e)根据以上实验，结合教材，分析分层抽样和不同的抽样方式对泛化误差的影响;
- (f)书上的2.1.