第二章　学习理论

张皓

https://haomood.github.io/homepage/

zhangh0214@gmail.com

摘要

在本章, 我们介绍机器学习的理论基础. 其目的是分析学习任务的困难本质、为学习算法提供理论保证、并根据分析结果指导算法设计. 本章将介绍为什么机器学习是可能的, 以及对泛化误差进行分解.

本系列文章有以下特点: *(a).* 为了减轻读者的负担并能使尽可能多的读者从中收益, 本文试图尽可能少地使用数学知识, 只要求读者有基本的微积分、线性代数和概率论基础, 并在第一节对关键的数学知识进行回顾和介绍. *(b).* 本文不省略任何推导步骤, 适时补充背景知识, 力图使本节内容是自足的, 使机器学习的初学者也能理解本文内容. *(c).* 机器学习近年来发展极其迅速, 已成为一个非常广袤的领域. 本文无法涵盖机器学习领域的方方面面, 仅就一些关键的机器学习流派的方法进行介绍. *(d).* 为了帮助读者巩固本文内容, 或引导读者扩展相关知识, 文中穿插了许多问题, 并在最后一节进行问题的 "快问快答".

1　为什么机器学习是可能的

1.1　机器学习的目标

定义 1 (误差 (error))**.** 学习器的实际预测输出和样本的真实预测输出之间的差异. 学习器在训练集 D 上的误差称为训练误差 (training error) 或经验误差 (empirical error)

$$\hat{e}(h) := \frac{1}{m}\sum_{i=1}^{m}\mathbb{I}(h(\boldsymbol{x}_i)\neq y_i)\,. \tag{1}$$

学习器在新样本上的误差称为泛化误差 (generalization error)

$$e(h) := \mathbb{E}[\mathbb{I}(h(\boldsymbol{x})\neq y)] = \Pr(h(\boldsymbol{x})\neq y)\,. \tag{2}$$

定义 2 (泛化 (generalization))**.** 学得模型 h 适用于未见示例的能力.

学习的目的就是为了使得学习得到的假设 h 逼近真相 f. 更确切地说, 机器学习的目标是使学得的模型具有强泛化能力 (泛化误差 $e(h)$ 小), 能很好地适用于整个样本空间, 而不仅仅在训练样本上工作得很好 (训练误差 $\hat{e}(h)$ 小).

1.2　没有免费的午餐

定理 1 (没有免费的午餐定理 (no free lunch theorem))**.** 对任意两个学习算法 A 和 B, 若在某些问题上 A 比 B 好, 则必然存在另外一些问题 B 比 A 好.

Proof. 为简单起见, 假设样本空间 $\mathcal{X}$ 和假设空间 $\mathcal{H}$ 都是离散的. 对某个特定的学习算法, 令 $p(h\mid D)$ 为基于训练数据 D 产生假设 h 的概率. 则该算法在训练集外所有样本上的期望误差为

$$\begin{aligned} e &:= \mathbb{E}_h[\mathbb{E}_{\boldsymbol{x}\in\mathcal{X}-D}[\mathbb{I}(\boldsymbol{x})\neq f(\boldsymbol{x})]] \\ &= \sum_{h\in\mathcal{H}}\sum_{\boldsymbol{x}\in\mathcal{X}-D}\mathbb{I}(h(\boldsymbol{x})\neq f(\boldsymbol{x}))\,p(\boldsymbol{x})\,p(h\mid D)\,. \end{aligned} \tag{3}$$

考虑二分类问题, 真相 $f\colon \mathcal{X}\to\{0,1\}$ 可以是任何

函数, 对所有可能的真相按均匀分布对误差求和,

$$\begin{aligned}\sum_f e &= \sum_f \sum_{h\in\mathcal{H}} \sum_{\boldsymbol{x}\in\mathcal{X}-D} \mathbb{I}(h(\boldsymbol{x}) \neq f(\boldsymbol{x}))\, p(\boldsymbol{x})\, p(h \mid D) \\ &= \sum_{\boldsymbol{x}\in\mathcal{X}-D} p(\boldsymbol{x}) \sum_{h\in\mathcal{H}} p(h \mid D) \sum_f \mathbb{I}(h(\boldsymbol{x}) \neq f(\boldsymbol{x})) \\ &= \sum_{\boldsymbol{x}\in\mathcal{X}-D} p(\boldsymbol{x}) \sum_{h\in\mathcal{H}} p(h \mid D)\, \frac{1}{2}\, 2^{|\mathcal{X}|} \\ &= 2^{|\mathcal{X}|-1} \sum_{\boldsymbol{x}\in\mathcal{X}-D} p(\boldsymbol{x})\,. \end{aligned} \tag{4}$$

其中利用了若 f 是均匀分布, 则有一半的 f 对 $\boldsymbol{x}$ 的预测和 $h(\boldsymbol{x})$ 不一致. 可以看出, 总误差和学习算法无关. □

没有免费的午餐定理的启示. 在所有问题出现的机会相同、或所有问题同等重要时, 任意两个学习算法的期望性能都相同. 但事实上, f 并不是均匀分布. 另一方面, 在实际情况中我们只关注自己正在试图解决的问题, 希望为它找到一个解决方案, 至于这个解决方案在别的问题、甚至在相似的问题上是否为好方案, 我们并不关心. 因此, 没有免费的午餐定理最重要的寓意是让我们认识到, 脱离具体问题, 空泛地谈论 "什么学习算法更好" 毫无意义, 因为若考虑所有潜在的问题, 则所有学习算法都一样好. 学习算法自身的归纳偏好与问题是否匹配, 往往会起到决定性的作用.

1.3 概率近似正确

尽管训练集 D 通常通常只是样本空间的一个很小的采样, 我们仍希望它能很好地反映出样本空间的特性, 否则就很难期望在训练集上学得的模型能在整个样本空间上都工作得很好. 训练样例数目 m 越大, 我们得到关于 $\mathcal{D}$ 的信息越多, 这样就越有可能通过学习获得具有强泛化能力的模型.

定义 3 (独立同分布 (independent and identically distributed, iid) 假设)**.** 样本空间中全部样例 $(\boldsymbol{x}, y) \in \mathcal{X}\times\mathcal{Y}$ 服从一个未知的分布 $(\boldsymbol{x}, y) \sim \mathcal{D}$, 并且我们获得的每个样例都是从这个分布上采样得到的.

对于较困难的问题, f 通常不在假设空间 $\mathcal{H}$ 中. 或者, 即使 $f\in\mathcal{H}$, 但由于训练数据 D 往往仅包含有限数量的样例, 因此, 通常会存在一些在 D 上等效的假设, 学习算法无法对它们进行区别, 再如从分布 $\mathcal{D}$ 采样得到 D 的过程中有一定的偶然性, 即使对同样大小的不同训练集, 学得结果也可能有所不同. 因此, 给定训练集 D, 我们希望学习器学得的模型对应的假设 h 尽可能接近真相 f. 也就是说, 以较大的概率学得 $h\approx f$ 的模型.

引理 2 (Hoeffding 不等式 [3])**.** *若 $x_1, x_2, \ldots, x_n$ 是 n 个独立的随机变量, 且满足 $x_i\in[a_i, b_i]$, 令 $s := \sum_{i=1}^n X_i$,*

$$\forall\epsilon > 0.\ \Pr(s - \mathbb{E}[s] \geq \epsilon) \leq \exp\left(-\frac{2\epsilon^2}{\sum_{i=1}^n (b_i - a_i)^2}\right); \tag{5}$$

$$\forall\epsilon > 0.\ \Pr(\mathbb{E}[s] - s \geq \epsilon) \leq \exp\left(-\frac{2\epsilon^2}{\sum_{i=1}^n (b_i - a_i)^2}\right). \tag{6}$$

定理 3. *若 D 中所有样例满足独立同分布假设, 对任意固定 h,*

$$\forall\epsilon\in(0,1).\ \Pr(|e(h) - \hat{e}(h)| < \epsilon) > 1 - 2\exp(-2m\epsilon^2)\,. \tag{7}$$

令 $\delta := 2\exp(-2m\epsilon^2)$. 也就是说, 以至少 $1-\delta$ 的概率,

$$|e(h) - \hat{e}(h)| < \epsilon = \sqrt{\frac{\log\frac{2}{\delta}}{2m}} = \tilde{\mathcal{O}}\left(\sqrt{\frac{1}{m}}\right). \tag{8}$$

Proof. 可将 Hoeffding 不等式合并写作

$$\forall\epsilon > 0.\ \Pr(|s - \mathbb{E}[s]| < \epsilon) > 1 - 2\exp\left(-\frac{2\epsilon^2}{\sum_{i=1}^n (b_i - a_i)^2}\right). \tag{9}$$

进一步可以得到对任意 $\epsilon > 0$,

$$\Pr\left(\left|\frac{1}{n}s - \frac{1}{n}\mathbb{E}[s]\right| < \epsilon\right) > 1 - 2\exp\left(-\frac{2n^2\epsilon^2}{\sum_{i=1}^n (b_i - a_i)^2}\right). \tag{10}$$

令 $s := \sum_{i=1}^m \mathbb{I}(h(\boldsymbol{x}_i) \neq y_i)$, 由于 $\mathbb{I}(h(\boldsymbol{x}_i) \neq y_i)\in[0,1]$, 且 $\mathbb{E}[\hat{e}] = e$, 即证. □

概率近似正确的含义. 概率近似正确 (probably approximately correct, PAC) 是计算学习理论中最基本的学习理论 [5] , 其包含了两个参数 ϵ 和 δ. 准确率参数 ϵ 描述了训练误差和泛化误差的差异 (对应于 "近似正确"), 置信度参数 δ 描述了以多大的概率假设函数能满足该准确率要求 (对应于 "概率").

为什么是概率近似正确? 由于训练数据 D 是有限的, 总是会有训练数据 D 无法完全对分布 $\mathcal{D}$ 刻画到的细节. 准确率参数 ϵ 允许学习模型有微小的偏差.

注意, Hoeffding 不等式是对随机采样求概率, 因此定理 3 是对训练数据 $D\sim\mathcal{D}$ 求概率, 这需要 h 在产生

训练数据 D 之前固定. 当假设空间 $\mathcal{H}$ 中包含不止一个假设函数时, 虽然定理 3 对每个假设函数个体 h 成立, 但如果允许在生成训练数据 D 后改变 h , 使 h 依赖于 D, Hoeffding 不等式将不再成立.

然而, 使 h 依赖于 D 正是学习算法的目标, 例如选择在训练数据 D 上误差最小的假设函数作为输出 $h^\star$. 此时我们无法直接使用 Hoeffding 不等式. 解决方案是使泛化误差界不依赖于学习算法的输出. 当假设空间 $\mathcal{H}$ 是有限时, 令 $h_1, h_2, \ldots, h_{|\mathcal{H}|}$ 为假设空间 $\mathcal{H}$ 中的假设, 尽管 $h^\star$ 依赖于训练集 D, 但 $h^\star$ 总是 $h_1, h_2, \ldots, h_{|\mathcal{H}|}$ 其中之一.

引理 4. 当假设空间 $\mathcal{H}$ 有限时,

$$\forall \epsilon \in (0,1).\ \Pr(\forall h\colon |e(h)-\hat{e}(h)| < \epsilon) > 1-2|\mathcal{H}|\exp(-2m\epsilon^2)\,. \tag{11}$$

Proof. 由概率基本公理 $\Pr(\bigvee_i E_i) \le \sum_i \Pr(E_i)$,

$$\begin{aligned}
\Pr(\forall h\colon |e(h)-\hat{e}(h)| \le \epsilon) &= 1-\Pr(\exists h\colon |e(h)-\hat{e}(h)| \ge \epsilon) \\
&= 1-\Pr(\bigvee_{k=1}^{|\mathcal{H}|} |e(h_k)-\hat{e}(h_k)| \ge \epsilon) \\
&> 1-\sum_{k=1}^{|\mathcal{H}|} \Pr(|e(h_k)-\hat{e}(h_k)| \ge \epsilon) \\
&> 1-\sum_{k=1}^{|\mathcal{H}|} 2\exp(-2m\epsilon^2) \\
&= 1-2|\mathcal{H}|\exp(-2m\epsilon^2)\,.
\end{aligned} \tag{12}$$

□

但是, 现实学习任务中所面临的通常是无限假设空间 (例如 $\mathbb{R}^d$ 空间中的所有线性超平面), $|\mathcal{H}| = \infty$, 此时 $\Pr(\forall h\colon |e(h)-\hat{e}(h)| < \epsilon) > -\infty$ 失去意义. 出现这种情况的原因在于

$$\Pr(\bigvee_{k=1}^{|\mathcal{H}|} |e(h_k)-\hat{e}(h_k)| \ge \epsilon) \le \sum_{k=1}^{|\mathcal{H}|} \Pr(|e(h_k)-\hat{e}(h_k)| \ge \epsilon) \tag{13}$$

过于松, 例如对线性超平面稍微改变一点不会对事件 $|e(h_k)-\hat{e}(h_k)| \ge \epsilon$ 对训练集 D 的结果改变很大.

无限假设空间的处理思路. 在无限假设空间, 尽管假设函数个数有无穷多个, 但有很多假设函数是相似的. 我们设法将相似的假设函数归为一类, 因此可以将 $|\mathcal{H}|$ 取代为一个 "有效" 且有限的假设空间复杂度. 为了对此种情形的可学习性问题进行研究, 需要设法度量假设空间的复杂度.

1.4 VC 维

引入增长函数的动机. 尽管假设函数 $h \in \mathcal{H}$ 可能有无穷多个, 但训练数据 D 的样本数 m 是有限的. 样本空间 $\mathcal{H}$ 中不同的假设对 m 个样本赋予标记的结果可能相同, 也可能不同. 我们称对这 m 个样本赋予标记的每种可能结果为一种对分 (dichotomy). $\mathcal{H}$ 中所有假设能对 m 个样本赋予标记的可能结果数 (对分总数) 是有限的.

定义 4 (增长函数 (growth function))**.** 假设空间 $\mathcal{H}$ 对 m 个样本所能赋予标记的最大可能的结果数

$$\Pi_{\mathcal{H}}(m) := \max_{\{\boldsymbol{x}_i\}_{i=1}^m \subseteq \mathcal{X}} |\{(h(\boldsymbol{x}_1), h(\boldsymbol{x}_2), \ldots, h(\boldsymbol{x}_m)) \mid h \in \mathcal{H}\}|\,. \tag{14}$$

显然, $\Pi_{\mathcal{H}}(m) \le 2^m$. 当 $\Pi_{\mathcal{H}}(m) = 2^m$ 时, 即能实现所有对分时, 我们称 m 个样本能被假设空间打散 (shatter).

引入 VC 维的动机. 当假设空间 $\mathcal{H}$ 的表示能力越强, $\mathcal{H}$ 对样本能赋予标记的可能结果数越大, 对学习任务的适应能力也越强. 因此, 我们可以利用增长函数度量假设空间 $\mathcal{H}$ 的表示能力.

定义 5 (VC 维 (VC dimension))**.** 能被 $\mathcal{H}$ 打散的最大样本数

$$\mathrm{VC}(\mathcal{H}) := \max\{m \mid \Pi_{\mathcal{H}}(m) = 2^m\}\,. \tag{15}$$

也就是说, 若存在大小为 d 的样本集能被 $\mathcal{H}$ 打散, 但对任意大小为 $d+1$ 的样本集都不能被 $\mathcal{H}$ 打散, 则 $\mathcal{H}$ 的 VC 维是 d.

定理 5. $\mathbb{R}^d$ 空间中的超平面构成的假设空间

$$\mathcal{H} := \{h \mid \forall \boldsymbol{w} \in \mathbb{R}^d, b \in \mathbb{R}.\ h(\boldsymbol{x}) = \boldsymbol{w}^\top \boldsymbol{x} + b\} \tag{16}$$

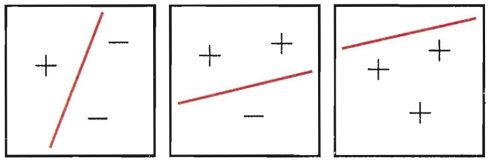
的 *VC* 维是 $d+1$, 如图 *1* 所示. 实际经验是, *VC* 维约为学习算法的自由参数个数.

Proof. 证明包括两个部分.

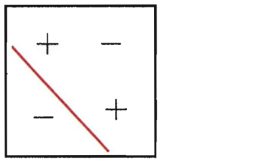
(1). 存在大小为 $d+1$ 的样本集能被 $\mathcal{H}$ 打散.

对任意 $i = 1, 2, \ldots, d$, 令 $\boldsymbol{x}_i := \mathbf{e}_i$, 令 $\boldsymbol{x}_{d+1} = \mathbf{0}$. 对任意标记 $y_1, y_2, \ldots, y_{d+1}$, 记

$$\boldsymbol{X} := \begin{bmatrix} \boldsymbol{x}_1^\top & 1 \\ \boldsymbol{x}_2^\top & 1 \\ \vdots & \\ \boldsymbol{x}_{d+1}^\top & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & \cdots & 0 & 1 \\ 0 & 1 & \cdots & 0 & 1 \\ 0 & 0 & \cdots & 1 & 1 \\ 0 & 0 & \cdots & 0 & 1 \end{bmatrix}; \tag{17}$$



(a) 示例集大小为 3



(b) 示例集大小为 4

Figure 1: $\mathbb{R}^2$ 平面中所有线性划分构成的假设空间的 VC 维是 3. 本图源于 [7] .

$$\boldsymbol{y} := \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_{d+1} \end{bmatrix}, \ \boldsymbol{\theta} := \begin{bmatrix} \boldsymbol{w} \\ b \end{bmatrix}. \tag{18}$$

由于 $\boldsymbol{X}$ 可逆, 因此对任意 $\boldsymbol{y}$, 我们总能找到 $\boldsymbol{\theta} := \boldsymbol{X}^{-1}\boldsymbol{y}$, 使得 $\boldsymbol{X}\boldsymbol{\theta} = \boldsymbol{y}$, 即 $h(\boldsymbol{x}_i) = \boldsymbol{w}^\top \boldsymbol{x}_i + b = y_i$. 换言之, 该样本集可以被 $\mathcal{H}$ 打散.

(2). 对任意大小为 $d+2$ 的样本集都不能被 $\mathcal{H}$ 打散.

对任意 $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{d+2}$, 记 $\tilde{\boldsymbol{x}} := \begin{bmatrix} \boldsymbol{x} \\ 1 \end{bmatrix} \in \mathbb{R}^{d+1}$. 由于样本数 $m := d+2$ 大于维度 $d+1$, 这些样本间必线性相关. 即存在不全为 0 的系数 $\alpha_1, \alpha_2, \ldots, \alpha_{d+1}$, 使得

$$\boldsymbol{x}_{d+2} = \sum_{i=1}^{d+1} \alpha_i \boldsymbol{x}_i \,. \tag{19}$$

$\mathcal{H}$ 无法实现标记

$$\boldsymbol{y} := \begin{bmatrix} \operatorname{sign} \alpha_1 \\ \operatorname{sign} \alpha_2 \\ \vdots \\ \operatorname{sign} \alpha_{d+1} \\ -1 \end{bmatrix}, \tag{20}$$

其中我们规定 $\operatorname{sign} 0 = 1$. 我们使用反证法, 假设能实现该特定标记. 即对任意 $i = 1, 2, \ldots, d+1$,

$$\operatorname{sign} \boldsymbol{\theta}^\top \tilde{\boldsymbol{x}}_i = y_i = \operatorname{sign} \alpha_i \,. \tag{21}$$

也就是说, $\boldsymbol{\theta}^\top \tilde{\boldsymbol{x}}_i$ 和 $\operatorname{sign} \alpha_i$ 同号. 那么对 $\boldsymbol{x}_{d+2}$,

$$\operatorname{sign} \boldsymbol{\theta}^\top \tilde{\boldsymbol{x}}_{d+2} = \operatorname{sign} \boldsymbol{\theta}^\top \sum_{i=1}^{d+1} \alpha_i \tilde{\boldsymbol{x}}_i = \operatorname{sign} \sum_{i=1}^{d+1} \boldsymbol{\theta}^\top \alpha_i \tilde{\boldsymbol{x}}_i = 1 \tag{22}$$

与 y_{d+2} 矛盾. 因此 $\mathcal{H}$ 无法打散任意大小为 $d+2$ 的样本集. □

1.5 基于 VC 维的泛化误差界

引理 6 (Sauer 引理 [4]).

$$\Pi_{\mathcal{H}}(m) \leq \sum_{d=0}^{\mathrm{VC}(\mathcal{H})} \binom{m}{d}. \tag{23}$$

Proof. 使用归纳法.

当 $m = 1, \mathrm{VC}(\mathcal{H}) = 0$ 或 $m = 1, \mathrm{VC}(\mathcal{H}) = 1$ 时, 定理成立.

假设定理对 $m-1, \mathrm{VC}(\mathcal{H}) - 1$ 和 $m-1, \mathrm{VC}(\mathcal{H})$ 成立. 对任意 $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m$, 令

$$\mathcal{H}(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m) := \{(h(\boldsymbol{x}_1), h(\boldsymbol{x}_2), \ldots, h(\boldsymbol{x}_m)) \mid h \in \mathcal{H}\}. \tag{24}$$

那么

$$\Pi_{\mathcal{H}}(m) = \max_{\{\boldsymbol{x}_i\}_{i=1}^m \subseteq \mathcal{X}} |\mathcal{H}(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m)|. \tag{25}$$

由于任何假设 h 对 $\boldsymbol{x}_m$ 的分类结果要么为 $+1$, 要么为 -1, 因此任何出现在 $\mathcal{H}(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{m-1})$ 中的序列 $(y_1, y_2, \ldots, y_{m-1})$ 都会在 $\mathcal{H}(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m)$ 中出现一次 (对应 $y_m = 1$ 异或 $y_m = -1$) 或两次 (对应 $y_m = 1$ 和 $y_m = -1$ 各一次).

记出现一次和两次的序列分别组成集合 $\mathcal{H}_1(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{m-1})$ 和 $\mathcal{H}_2(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{m-1})$, 我们有

$$\begin{aligned} &|\mathcal{H}(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m)| \\ &= |\mathcal{H}_1(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{m-1})| + 2|\mathcal{H}_2(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{m-1})| \\ &= |\mathcal{H}(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{m-1})| + |\mathcal{H}_2(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{m-1})|. \end{aligned} \tag{26}$$

对 $|\mathcal{H}(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{m-1})|$, 利用归纳假设

$$|\mathcal{H}(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{m-1})| \leq \Pi_{\mathcal{H}}(m-1) \leq \sum_{d=0}^{\mathrm{VC}(\mathcal{H})} \binom{m-1}{d}. \tag{27}$$

对 $|\mathcal{H}_2(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{m-1})|$ 则有 $\mathrm{VC}(\mathcal{H}_2) \leq \mathrm{VC}(\mathcal{H}) - 1$. 否则, 若 $\mathrm{VC}(\mathcal{H}_2)$ 能打散 $\mathrm{VC}(\mathcal{H})$ 个样本, 将 x_m 添加到 $\mathcal{H}_2$ 能打散的样本的集合中, $\mathcal{H}$ 将能打散 $\mathrm{VC}(H)+1$ 个样本, 与 VC 维定义矛盾. 因此

$$\begin{aligned} |\mathcal{H}_2(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{m-1})| &\leq \Pi_{\mathcal{H}_2}(m-1) \\ &\leq \sum_{d=0}^{\mathrm{VC}(\mathcal{H}_2)} \binom{m-1}{d} \\ &\leq \sum_{d=0}^{\mathrm{VC}(\mathcal{H})-1} \binom{m-1}{d}. \end{aligned} \tag{28}$$

将公式 27 和公式 28 代入公式 26 , 可以得到

$$\begin{aligned} |\mathcal{H}(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m)| &\leq \sum_{d=0}^{\mathrm{VC}(\mathcal{H})} \binom{m-1}{d} + \sum_{d=0}^{\mathrm{VC}(\mathcal{H})-1} \binom{m-1}{d} \\ &= \sum_{d=0}^{\mathrm{VC}(\mathcal{H})} \left(\binom{m-1}{d} + \binom{m-1}{d-1} \right) \\ &= \sum_{d=0}^{\mathrm{VC}(\mathcal{H})} \binom{m}{d}. \end{aligned} \tag{29}$$

由样本 $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m$ 的任意性, 即证. □

推论 7. 对任意整数 $m \geq VC(\mathcal{H})$,

$$\Pi_{\mathcal{H}}(m) \leq \left(\frac{\mathrm{e}m}{\mathrm{VC}(\mathcal{H})} \right)^{\mathrm{VC}(\mathcal{H})}. \tag{30}$$

Proof.

$$\begin{aligned} \Pi_{\mathcal{H}}(m) &\leq \sum_{d=0}^{\mathrm{VC}(\mathcal{H})} \binom{m}{d} \\ &\leq \sum_{d=0}^{\mathrm{VC}(\mathcal{H})} \binom{m}{d} \left(\frac{m}{\mathrm{VC}(\mathcal{H})} \right)^{\mathrm{VC}(\mathcal{H})-d} \\ &= \left(\frac{m}{\mathrm{VC}(\mathcal{H})} \right)^{\mathrm{VC}(\mathcal{H})} \sum_{d=0}^{\mathrm{VC}(\mathcal{H})} \binom{m}{d} \left(\frac{\mathrm{VC}(\mathcal{H})}{m} \right)^{d} \\ &\leq \left(\frac{m}{\mathrm{VC}(\mathcal{H})} \right)^{\mathrm{VC}(\mathcal{H})} \sum_{d=0}^{m} \binom{m}{d} \left(\frac{\mathrm{VC}(\mathcal{H})}{m} \right)^{d} \\ &= \left(\frac{m}{\mathrm{VC}(\mathcal{H})} \right)^{\mathrm{VC}(\mathcal{H})} \left(1 + \frac{\mathrm{VC}(\mathcal{H})}{m} \right)^{m} \\ &\leq \left(\frac{m}{\mathrm{VC}(\mathcal{H})} \right)^{\mathrm{VC}(\mathcal{H})} \mathrm{e}^{\mathrm{VC}(\mathcal{H})} \\ &= \left(\frac{\mathrm{e}m}{\mathrm{VC}(\mathcal{H})} \right)^{\mathrm{VC}(\mathcal{H})}. \end{aligned}$$

□

引理 8. 对假设空间 $\mathcal{H}$ 和样本数 m, 对任意 $h \in \mathcal{H}$ 和任意 $\epsilon \in (0,1)$,

$$\Pr(|e - \hat{e}| \leq \epsilon) \geq 1 - 4\Pi_{\mathcal{H}}(2m) \exp\left(-\frac{m\epsilon^2}{8} \right). \tag{31}$$

Proof. 见 [6] . □

推论 9. 对假设空间 $\mathcal{H}$ 和样本数 m, 对任意 $h \in \mathcal{H}$ 和任意 $\epsilon \in (0,1)$,

$$\Pr(|e - \hat{e}| \leq \epsilon) \geq 1 - 4\left(\frac{\mathrm{e}m}{\mathrm{VC}(\mathcal{H})} \right)^{\mathrm{VC}(\mathcal{H})} \exp\left(-\frac{m\epsilon^2}{8} \right). \tag{32}$$

Proof. 将推论 7 代入引理 8 即得. □

定理 10. 以 $1-\delta$ 的概率, 对任意 $h \in \mathcal{H}$, $m > \mathrm{VC}(\mathcal{H})$ 有

$$\begin{aligned} |e - \hat{e}| &\leq \sqrt{\frac{8}{m}\left(\mathrm{VC}(\mathcal{H}) \log \frac{2\mathrm{e}m}{\mathrm{VC}(\mathcal{H})} + \log \frac{4}{\delta} \right)} \\ &= \tilde{\mathcal{O}}\left(\sqrt{\frac{\mathrm{VC}(\mathcal{H})}{m}} \right). \end{aligned}$$

Proof. 令 $\delta := 4\left(\frac{\mathrm{e}m}{\mathrm{VC}(\mathcal{H})} \right)^{\mathrm{VC}(\mathcal{H})} \exp\left(-\frac{m\epsilon^2}{8} \right)$, 代入推论 9 即得. □

基于 *VC* 维的泛化误差界的适用范围. 泛化误差界只和样例数目 m 有关, 与数据分布 $\mathcal{D}$ 和训练集 D 无关. 因此, 基于 VC 维的泛化误差界是分布无关、数据独立的, 对任何数据分布都适用. 收敛速率是 $\tilde{\mathcal{O}}\left(\sqrt{\frac{1}{m}}\right)$.

基于 *VC* 维的泛化误差界的意义. 由于没有考虑数据自身, 基于 VC 维得到的泛化误差界通常比较松. 但是, 其对不同的学习算法松的程度大致相同, 因此可以用来比较不同学习算法的泛化误差. 从现实应用中发现, 具有比较小 VC 维的学习算法的泛化能力通常更好. 实际经验是, 为了使得学得模型有比较好的泛化能力, 需要训练样例数 $m \geq 10 \cdot \mathrm{VC}(\mathcal{H})$.

近似-泛化权衡. 如图 3 所示, 近似-泛化权衡 (approximation-generalization tradeoff) 是指, 当学习算法的 VC 维较高时, 其对训练数据拟合的比较好, 但是将有很高的泛化误差界. 当学习算法的 VC 维较低时, 训练误差和泛化误差之间的差异较小, 但是对训练数据拟合能力较弱. 如图 2 所示, 具有较低 VC 维的学习算法收敛较快, 但是最终的性能比较高 VC 的学习算法差.

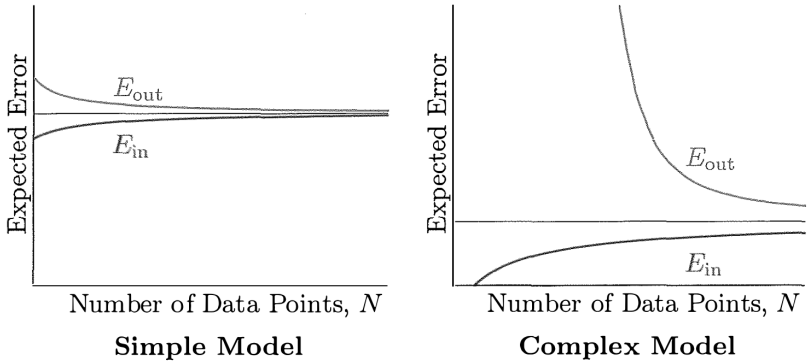


Figure 2: 不同 VC 维的学习算法的学习曲线. 本图源于 [1] .

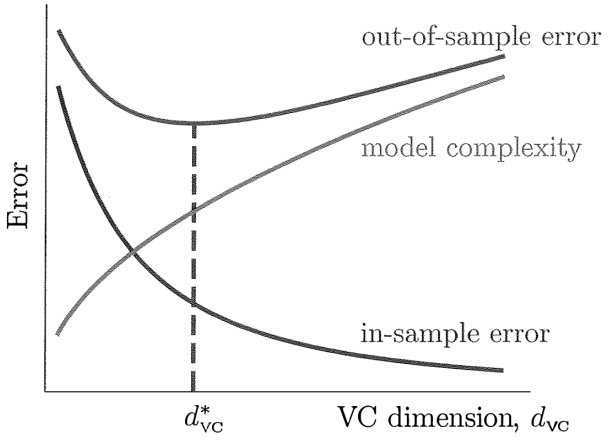


Figure 3: 近似-泛化权衡示意图. 本图源于 [1] .

计算机科学研究的算法通常考虑时间复杂度. 在机器学习算法中, 假定学习算法处理每个样本的时间为常数, 则学习算法的时间复杂度等价于样本复杂度. 因此, 我们对机器学习算法时间复杂度的关心转化为对样本复杂度的关心.

1.6 经验风险最小化

经验风险最小化. 为了得到泛化误差小的学习器, 我们努力使经验误差最小化, 也就是从训练样本中尽可能学出适用于所有潜在样本的普遍规律, 这样才能在遇到新样本时做出正确的判别. 在后面的章节中将介绍不同的学习算法如何最小化经验误差.

定理 11. 对任何 *VC* 维有限的假设空间 $\mathcal{H}$, 经验风险最小化的假设函数

$$h^\star := \arg\min_{h\in\mathcal{H}} \ \hat{e}(h) \tag{33}$$

以 $1-\delta$ 概率满足

$$e(h^\star) - \min_h e(h) < \epsilon\,. \tag{34}$$

Proof. 令

$$g := \arg\min_h \ e(h)\,. \tag{35}$$

以 $1-\delta$ 概率,

$$\begin{aligned} e(h^\star) - e(g) &\le \left(\hat{e}(h^\star) + \frac{\epsilon}{2}\right) - \left(\hat{e}(g) - \frac{\epsilon}{2}\right) \\ &= \hat{e}(h) - \hat{e}(g) + \epsilon \\ &\le \epsilon\,. \end{aligned} \tag{36}$$

□

2 泛化误差

在通过实验估计出学习算法的泛化性能之外, 我们还希望了解为什么具有这样的泛化性能.

2.1 泛化误差的成因

定理 12 (偏差-方差分解 (bias-variance decomposition) [2]). 对测试样本 $\boldsymbol{x}$, 定义 $h(\boldsymbol{x})$ 是从训练集 D 学得模型 h 在 $\boldsymbol{x}$ 上的预测输出, $f(\boldsymbol{x})$ 是 $\boldsymbol{x}$ 的真实标记, y 是 $\boldsymbol{x}$ 在数据集中的标记, 其可能存在标记噪声, 并假设噪声的期望为零 $\mathbb{E}_D[y] = f(\boldsymbol{x})$. 算法在不同训练集 D 上学得的结果很可能不同, 即使它们都来自同一个分布 $D \in \mathcal{D}$. 对于回归任务, 学习算法的期望泛化错误率可以拆解为

$$\begin{aligned} &\mathbb{E}_D[(h(\boldsymbol{x}) - y)^2] \\ &= (\bar{h}(\boldsymbol{x}) - f(\boldsymbol{x}))^2 + \mathbb{E}_D[(h(\boldsymbol{x}) - \bar{h}(\boldsymbol{x}))^2] + \mathbb{E}_D[(f(\boldsymbol{x}) - y)^2]\,, \end{aligned}$$

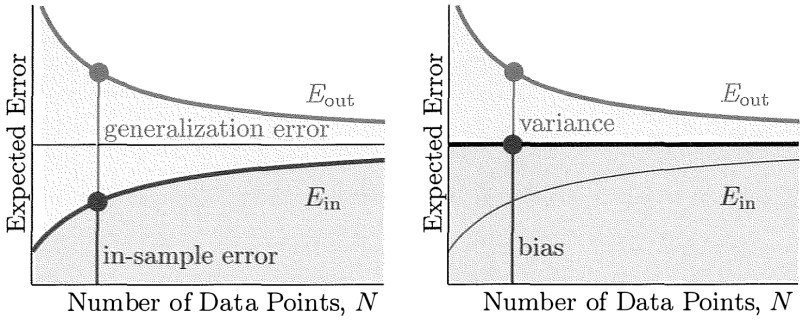


Figure 4: 对学习曲线的两种视角. 本图源于 [1] .

其中 $\bar{h}(\boldsymbol{x}) := \mathbb{E}_D[h(\boldsymbol{x})]$.

Proof. 为了使书写简单, 将 $h(\boldsymbol{x})$ 简记为 h, $\bar{h}(\boldsymbol{x})$ 简记为 $\bar{h}$, $f(\boldsymbol{x})$ 简记为 f.

$$\begin{aligned}
\mathbb{E}_D[e] &:= \mathbb{E}_D[(h-y)^2] \\
&= \mathbb{E}_D[(h-\bar{h}+\bar{h}-y)^2] \\
&= \mathbb{E}_D[(h-\bar{h})^2] + \mathbb{E}_D[(\bar{h}-f+f-y)^2] \\
&\quad + 2\,\mathbb{E}_D[(h-\bar{h})(\bar{h}-y)] \\
&= \mathbb{E}_D[(h-\bar{h})^2] + \mathbb{E}_D[(\bar{h}-f)^2] + \mathbb{E}_D[(f-y)^2] \\
&\quad + 2\,\mathbb{E}_D[(\bar{h}-f)(f-y)] \\
&= \mathbb{E}_D[(h-\bar{h})^2] + \mathbb{E}_D[(\bar{h}-f)^2] + \mathbb{E}_D[(f-y)^2] \\
&= (\bar{h}-f)^2 + \mathbb{E}_D[(h-\bar{h})^2] + \mathbb{E}_D[(f-y)^2],
\end{aligned}$$

其中用到了噪声不依赖于 h, 以及噪声的期望为 0. □

偏差-误差分解各项的含义.

- **偏差** $(\bar{h}(\boldsymbol{x})-y)^2$. 偏差度量了学习算法的期望预测 $\bar{h}$ 和真实标记 $f(\boldsymbol{x})$ 的偏离程度, 即刻画了学习算法本身的拟合能力.
- **方差** $\mathbb{E}_D[(h(\boldsymbol{x})-\bar{h}(\boldsymbol{x}))^2]$. 方差度量了同样大小的训练集的变动所导致的学习性能的变化, 即刻画了数据扰动所造成的影响.
- **噪声** $\mathbb{E}_D[(f(\boldsymbol{x})-y)^2]$. 噪声表达了在当前任务上任何学习算法所能达到的期望泛化误差的下界, 即刻画了学习问题本身的难度.

综上, 学习算法的泛化性能是由学习算法的能力、数据的充分性以及学习任务本身的难度共同决定的. 给定学习任务, 为了取得好的泛化性能, 则需使偏差较小, 即能够充分拟合数据, 并且使方差较小, 即使得数据扰动产生的影响小.

虽然偏差和方差确实反映了各类学习任务内在的误差决定因素, 但这样优美的数学形式仅在基于均方误差的回归任务中得以推导出. 对分类任务, 由于 0/1 损失函数的跳变性, 理论上推导出偏差-方差分解很困难. 已有多种方法对偏差和方差进行估计.

对学习曲线的两种视角. 如图 4 所示, 从 VC 的视角来看, 泛化误差 $e(h)$ 是由经验误差 $\hat{e}(h)$ 和泛化误差界构成. 在偏差-方差分解的视角来看, 泛化误差 $e(h)$ 是由偏差和方差构成. 当数据量 m 增大时, 泛化误差界和方差都随之减小.

2.2 欠拟合和过拟合

定义 6 (欠拟合 (underfitting)). 学习器对训练样本的一般性质尚未学好, 导致泛化能力低.

定义 7 (过拟合 (overfitting)). 学习器把训练样本学得"太好了", 以至于把训练样本自身的一些特点当作了所有潜在样本都会有的一般性质, 导致泛化能力下降.

如何应对欠拟合? 欠拟合通常是由于学习能力低下造成的. 克服方案例如在决策树学习中扩展分支、在神经网络学习中增加训练轮数等.

为什么过拟合是无法彻底避免的? 这是因为机器学习面临的问题通常是 NP 难甚至更难, 而有效的机器学习算法必然是在多项式时间内运行完成, 若可彻底避免过拟合, 则通过经验误差最小化就能获得最优解, 这意味着我们构造性的证明了 P=NP.

如何应对过拟合? 过拟合的主要原因包括: (1). 模型学习能力过于强大, VC 维过高. (2). 数据中噪声过大. (3). 数据量 m 过小. 缓解过拟合的手段包括: (1). 从简单模型开始 (奥卡姆剃刀原则)、正则化 (可以降低

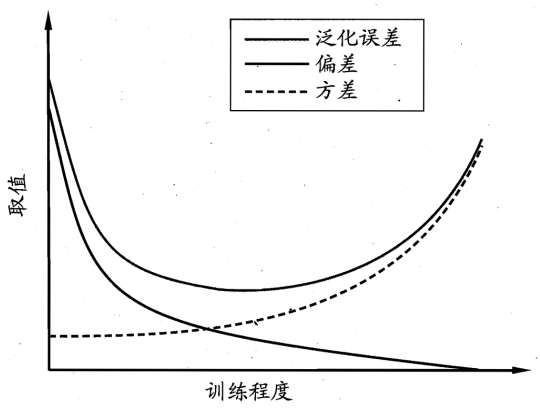


Figure 5: 泛化误差与偏差、方差的关系示意图. 本图源于 [7] .

模型有效的复杂度、减小噪声的影响). (2). 数据清洗和预处理 (效果不一定). (3). 数据扩充 (注意扩充的数据不满足独立同分布假设). (4). 使用验证集 (用于对泛化误差进行估计, 通常用于模型选择).

偏差-方差窘境 (bias-variance dilemma). 如图 5 所示, 给定学习任务, 在训练不足时, 学习器的拟合能力不够强, 训练数据的扰动不足以使学习器产生显著变化, 此时偏差主导了泛化错误率. 随着训练程度的加深, 学习器的拟合能力逐渐增强, 训练数据发生的扰动渐渐能被学习器学到, 方差 逐渐主导了泛化错误率. 在训练程度充足后, 学习器的拟合能力已非常强, 训练数据发生的轻微扰动都会导致学习器发生显著变化, 若训练数据自身的、非全局的特征被学习器学到了, 则将发生过拟合.

2.3 最好的学习器能做到多好

定理 13. *最小化分类错误率的贝叶斯最优分类器是对每个样本 $\boldsymbol{x}$ 选择能使后验概率最大的类别标记*

$$h^\star(\boldsymbol{x}) := \arg\max_c \ \Pr(y=c \mid \boldsymbol{x}) \,. \tag{37}$$

Proof.

$$\begin{aligned} h^\star(\boldsymbol{x}) &:= \arg\min_c \ \mathbb{E}[\mathbb{I}(y \neq c)] \\ &= \arg\min_c \ \mathbb{I}(y \neq c)\, p(y \mid \boldsymbol{x})\, p(\boldsymbol{x}) \\ &= \arg\min_c \ \sum_{k=1}^{C} \mathbb{I}(k \neq c)\ \Pr(y=k \mid \boldsymbol{x}) \\ &= \arg\min_c \ \Pr(y \neq c \mid \boldsymbol{x}) \\ &= \arg\max_c \ \Pr(y=c \mid \boldsymbol{x}) \,. \end{aligned} \tag{38}$$

□

贝叶斯最优分类器反映了分类器能达到的最好性能, 即通过机器学习所能产生的模型错误率的理论下限.

3 快问快答

为什么机器学习不能实现完全准确的预测? 答案见上文.

References

[1] Y. S. Abu-Mostafa, M. Magdon-Ismail, and H.-T. Lin. *Learning from Data: A Short Course*. AMLBook, 2012. 6, 7

[2] S. Geman, E. Bienenstock, and R. Doursat. Neural networks and the bias/variance dilemma. *Neural Computation*, 4(1):1–58, 1992. 6

[3] C. McDiarmid. On the method of bounded differences. *Surveys in Combinatorics*, 141(1):148–188, 1989. 2

[4] N. Sauer. On the density of families of sets. *Journal of Combinatorial Theory - Series A*, 13(1):145–147, 1972. 4

[5] L. G. Valiant. A theory of the learnable. *Communications of the ACM*, 27(11):1134–1142, 1984. 2

[6] V. N. Vapnik and A. Y. Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability and Its Applications*, 16(2):264–280, 1971. 5

[7] 周志华. 机器学习. 清华大学出版社, 2016. 4, 8