泛化界

PAC 学习框架导读 (下)

Jiaxuan Zou

西安交通大学数学与统计学院

2025 年 5 月 1 日

回顾确定性场景下的泛化误差

在确定性场景中，每个输入 x 都有一个唯一的标签 y。泛化误差 $R(h)$ 定义为：

$$R(h) = \Pr_{x \sim D}[h(x) \neq c(x)] = \mathbb{E}_{x \sim D}[1_{h(x) \neq c(x)}]$$

其中 $h(x)$ 是模型预测的标签，$c(x)$ 是真实的标签，$1_{h(x) \neq c(x)}$ 是指示函数，当预测错误时取值为 1，否则为 0。

随机性场景下的泛化误差

在随机性场景中，输出标签是输入的概率函数，即对于每个输入 x，标签 y 可能不唯一。泛化误差 $R(h)$ 定义为：

$$R(h) = \Pr_{(x,y)\sim D}[h(x) \neq y] = \mathbb{E}_{(x,y)\sim D}[1_{h(x)\neq y}]$$

这里，$h(x)$ 是模型预测的标签，y 是实际的标签，$1_{h(x)\neq y}$ 是指示函数，当预测错误时取值为 1，否则为 0。
例如，如果我们根据个人身高和体重的输入对来预测性别，那么标签通常不是唯一的。对于大多数对，男性和女性都是可能的性别。对于每个固定的对，标签为男性的概率分布是存在的。

不可知 PAC 学习

将 PAC 学习框架自然扩展到这种设置被称为不可知 PAC 学习 (Agnostic PAC-learning)。

2.4.2 贝叶斯错误率和噪声

在确定性情况下，存在一个没有泛化误差的目标函数 f：$R(h) = 0$。在随机情况下，任何假设存在一个最小非零误差。

定义 2.5 贝叶斯错误率

给定分布 D 在 $\mathcal{X} \times \mathcal{Y}$ 上，贝叶斯错误率 R^* 定义为可测函数 $h : \mathcal{X} \to \mathcal{Y}$ 所能达到的错误的下确界：

$$R^* = \inf_{h \text{ measurable}} R(h).$$

使得 $R(h) = R^*$ 的假设 h 称为贝叶斯假设或贝叶斯分类器。

定义 2.6 噪声

给定分布 D 在 $\mathcal{X} \times \mathcal{Y}$ 上，点 $x \in \mathcal{X}$ 处的噪声定义为：

$$noise(x) = \min\{\Pr[1|x], \Pr[0|x]\}.$$

这表示在给定输入 x 的情况下，标签 y 的不确定性。

平均噪声

与分布 D 相关的平均噪声或噪声定义为 $noise(x)$ 的期望值：

$$\text{noise} = \mathbb{E}[noise(x)].$$

平均噪声精确地是贝叶斯错误率：$\text{noise} = \mathbb{E}[noise(x)] = R^*$。
噪声是学习任务的特征，表明了其难度水平。对于 $\mathcal{X}$ 中的点 x，如果 $noise(x)$ 接近 $1/2$，则有时称为 嘈杂，这当然是准确预测的挑战。

估计和逼近误差

假设 $h \in H$ 的错误与贝叶斯错误之间的差异可以分解为：

$$R(h) - R^* = (R(h) - R(h^*)) + (R(h^*) - R^*)$$

其中 h^* 是具有最小错误的假设，或最佳类别内假设。第一项是估计误差，它取决于所选的假设 h。它衡量了假设 h 相对于最佳类别内假设的质量。第二项称为逼近误差，因为它衡量了贝叶斯错误可以通过 H 近似的程度。它是假设集 H 的属性，衡量其丰富性

不可达的逼近误差

$$R(h) - R^* = (R(h) - R(h^*)) + (R(h^*) - R^*)$$

逼近误差通常不可达，因为一般不知道底层分布 D。即使有各种噪声假设，估计逼近误差也很困难。

算法的估计误差

$$R(h) - R^* = (R(h) - R(h^*)) + (R(h^*) - R^*)$$

算法 A 的估计误差，即在样本 S 上训练后返回的假设 h_S 的估计误差，有时可以用泛化误差来界定。

泛化界限的应用

例如，设 h_S^{ERM} 表示经验风险最小化算法返回的假设，即返回具有最小经验误差的假设 h_S^{ERM}。然后，可以使用定理 2.2 或任何其他界限 $\sup_{h\in H}\left|R(h)-\hat{R}(h)\right|$ 来界定经验风险最小化算法的估计误差。

估计误差的界定

通过重写估计误差，我们可以得到：

$$R(h_S^{\mathrm{ERM}}) - R(h^*) = R(h_S^{\mathrm{ERM}}) - \hat{R}(h_S^{\mathrm{ERM}}) + \hat{R}(h_S^{\mathrm{ERM}}) - R(h^*)$$

由于 $\hat{R}(h_S^{\mathrm{ERM}}) \leq \hat{R}(h^*)$，我们有：

$$R(h_S^{\mathrm{ERM}}) - R(h^*) \leq R(h_S^{\mathrm{ERM}}) - \hat{R}(h_S^{\mathrm{ERM}}) + \hat{R}(h^*) - R(h^*)$$

进一步简化，我们得到：

$$R(h_S^{\mathrm{ERM}}) - R(h^*) \leq 2 \sup_{h \in H} |R(h) - \hat{R}(h)|$$

这表明估计误差的两倍上界是所有假设中真实风险和经验风险之间差异的最大值。

总结

$$R(h_S^{\mathrm{ERM}}) - R(h^*) \leq 2 \sup_{h \in H} |R(h) - \hat{R}(h)|$$

通过这个推导，我们可以看到经验风险最小化算法的泛化误差可以被估计误差的两倍上界所限制。这为我们提供了一种评估和改进学习算法性能的方法。

估计误差和逼近误差的分解为我们提供了理解和分析学习算法性能的有力工具。通过控制这两个误差，我们可以优化学习算法的设计和性能。

ERM 算法的局限性

ERM 算法仅寻求最小化训练样本上的错误，可能会忽略复杂性项。实际上，ERM 算法的性能通常很差，而且在很多情况下，确定 ERM 解是计算上不可行的。

结构风险最小化（SRM）

另一种称为结构风险最小化（SRM）的方法包括考虑经验误差和复杂性项：

$$h_S^{\mathrm{SRM}} =_{h \in H_n} \hat{R}_S(h) + \mathrm{complexity}(H_n, m).$$

其中复杂性项取决于 H_n 的大小（或更一般地说，容量）和样本大小 m。

正则化方法

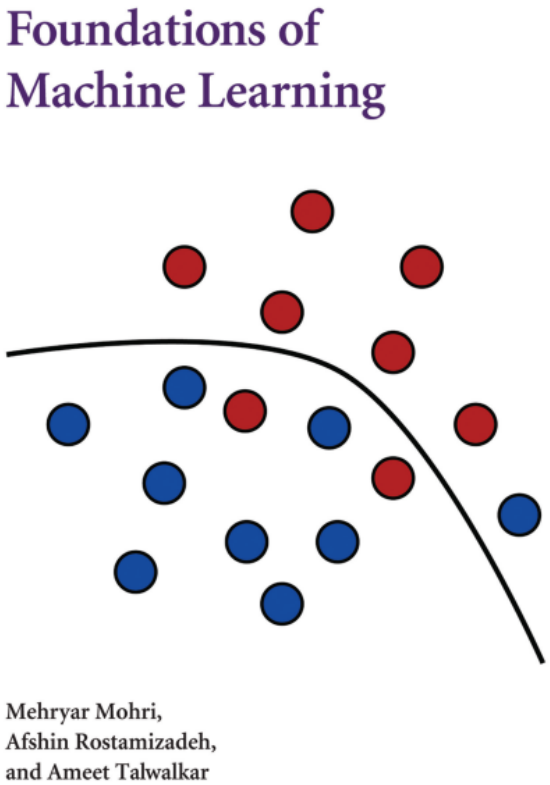
另一种算法族基于更直接的优化，包括最小化经验误差和正则化项的和，正则化项惩罚更复杂的假设：

$$h_S^{\mathrm{REG}} =_{h\in H} \hat{R}_S(h) + \lambda\|h\|^2.$$

其中 $\lambda \geq 0$ 是一个正则化参数，用于确定经验误差最小化和复杂性控制之间的权衡。

总结

模型选择是机器学习中的一个重要步骤，它涉及到在经验误差和模型复杂性之间找到平衡。不同的方法，如 ERM 和 SRM，提供了不同的策略来实现这一目标。



https://github.com/SmartPig-Joe/Foundations-of-Machine-Learning

Thanks!