可学习性

前言

机器学习理论研究的是关于机器学习的理论基础，主要内容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.

对一个任务，通常我们先要考虑它是不是“可学习的(learnable)”

本章就是介绍关于可学习性的基本知识:

PAC(概率近似正确)，可学习性，不可知PAC可学习性，概念类，假设空间.

数据与分布

给定样例集 $D = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_m, y_m)\}$ 其中 $\boldsymbol{x}_i \in \mathcal{X}$
若无特别说明, $y_i \in \mathcal{Y} = \{-1, +1\}$, 即本章主要讨论二分类问题.

基本假设:

$\mathcal{X}$ 中的所有样本服从一个隐含未知的分布 $\mathcal{D}$, D 中的所有样本都是独立地从 $\mathcal{D}$ 上采样得到, 记作 $D \sim \mathcal{D}^m$ 即独立同分布(independent and identically distributed 简称 $i.i.d.$)样本.

泛化误差与经验误差

令 h 为从 $\mathcal{X}$ 到 $\mathcal{Y}$ 的一个映射，其泛化误差为

$$E(h;\mathcal{D}) = P_{\boldsymbol{x}\sim\mathcal{D}}\big(h(\boldsymbol{x}) \neq y\big) = \mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}}\big[\mathbb{I}\big(h(\boldsymbol{x}) \neq y\big)\big], \tag{2.1}$$

h 在 D 上的经验误差为

$$\widehat{E}(h;D) = \frac{1}{m}\sum_{i=1}^{m}\mathbb{I}\big(h(\boldsymbol{x}_i) \neq y_i\big)\ . \tag{2.2}$$

两者之间的关系：由于D 是 $\mathcal{D}$ 的独立同分布采样，因此 h 的经验误差的期望等于其泛化误差.

在上下文明确时，$E(h;\mathcal{D})$ 和 $\widehat{E}(h;D)$分别简记为 $E(h)$和 $\widehat{E}(h)$.

误差参数 一致性 不合

误差参数 ϵ

ϵ 为 $E(h)$的上界，即 $E(h) \leqslant \epsilon$ ，表示预先设定的学得模型所应满足的误差要求，亦称误差参数.

一致性:

若 h 在数据集 D 上的经验误差为 0 , 则称 h 与 D 一致, 否则称其与 D 不一致.

不合（disagreement）：

对任意两个映射 $h_1, h_2 \in \mathcal{X} \to \mathcal{Y}$, 可通过其“不合”来度量它们之间的差别:

$$dis(h_1, h_2) = P_{\boldsymbol{x} \sim \mathcal{D}}(h_1(\boldsymbol{x}) \neq h_2(\boldsymbol{x})) \ . \qquad (2.3)$$

概念

概念:

令 c 表示概念(concept), 这是从样本空间 $\mathcal{X}$ 到标记空间 $\mathcal{Y}$ 的映射, 它决定示例 $\boldsymbol{x}$ 的真实标记 y

目标概念:

若对任何样例 $(\boldsymbol{x}, y)$, 都有 $c(\boldsymbol{x}) = y$ 成立, 则称 c 为目标概念

概念类:

所有希望学得的目标概念所构成的集合称为概念类(concept class), 用符号 $\mathcal{C}$ 表示.

概念的举例

例：三角形概念

目标概念“三角形”把所有三角形映射为1，把其它图形映射为-1。

例：概念类

一个概念类则是想要学习的概念构成的一个集合. 假设我们在学习任务中考虑“三角形”“四边形”“五边形”这三种概念，则它们组成的集合{三角形，四边形，五边形}就构成了一个概念类。

假设与假设空间

假设空间(hypothesis space):

给定学习算法 $\mathfrak{L}$，它所考虑的所有可能的概念构成的集合称为假设空间，用符号 $\mathcal{H}$ 表示.

学习算法 $\mathfrak{L}$ 事先并不知道概念类的真实存在，故 $\mathcal{H}$ 和 $\mathcal{C}$ 通常是不同的，学习算法会把自认为可能的目标概念集中起来构成 $\mathcal{H}$

假设(hypothesis):

对 $h \in \mathcal{H}$，由于并不能确定它是否真是目标概念，因此称为假设，显然假设 h 也是从示例空间 $\mathcal{X}$ 到标记空间 $\mathcal{Y}$ 的映射.

可分性

可分(separable):

若目标概念 $c \in \mathcal{H}$, 则 $\mathcal{H}$ 中存在假设能将样本正确地完全分开，我们称以 c 为目标的这个学习问题对假设空间$\mathcal{H}$ 是可分的.

不可分(non-separable):

若 $c \notin \mathcal{H}$, 则 $\mathcal{H}$ 中不存在任何假设能将所有样本正确地完全分开, 则称该学习问题对假设空间$\mathcal{H}$ 是不可分的.

PAC的基本思想

概率近似正确（Probably Approximately Correct ,PAC），[Valiant, 1984].

希望以比较大的把握学得比较好的模型，即以较大的概率学得误差满足预设上限的模型.

对于给定训练集 D，希望基于学习算法 $\mathfrak{L}$ 学得的模型对应的假设 h 尽可能地接近目标概念 c .

为什么不是希望精确地学得目标概念 c 呢?

机器学习过程受很多因素的制约：

获取的训练集 D 往往仅包含有限数量的样本，因此通常会存在一些在 D 上等效的假设，学习算法有时无法区分它们

从分布 $\mathcal{D}$ 采样得到数据集 D 有一定偶然性，即使对同样大小的不同训练集 D，学得结果也可能有所不同.

PAC辨识

形式化地说, 令 δ 表示置信参数, 可定义:

PAC辨识 (PAC Identify):

对 $0 < \epsilon, \delta < 1$, 所有 $c \in \mathcal{C}$ 和分布 $\mathcal{D}$, 若存在学习算法 $\mathfrak{L}$, 其输出的假设 $h \in \mathcal{H}$ 满足

$$P(E(h) \leqslant \epsilon) \geqslant 1 - \delta \ , \tag{2.4}$$

则称学习算法 $\mathfrak{L}$ 能从假设空间 $\mathcal{H}$ 中辨识概念类 $\mathcal{C}$.

这样的学习算法 $\mathfrak{L}$ 能以较大的概率(至少 $1 - \delta$)学得目标概念 c 的近似(误差最大为 ϵ)

在此基础上我们可以定义PAC可学习

PAC可学习

PAC可学习 (PAC Learnable) :

令 m 表示从分布 $\mathcal{D}$ 中独立同分布采样得到的样例数目,$0 < \epsilon, \delta < 1$, 对所有分布 $\mathcal{D}$,若存在学习算法 $\mathfrak{L}$ 和多项式函数 $poly(.,.,.,.)$, 使得对于任何

$$m \geqslant poly(1/\epsilon, 1/\delta\ ,\ size(x), size(c))$$

$\mathfrak{L}$ 能从假设空间 $\mathcal{H}$ 中 PAC 辨识 概念类 $\mathcal{C}$, 则称概念类 $\mathcal{C}$ 对假设空间 $\mathcal{H}$ 而言是 PAC 可学习的, 有时也简称概念类 $\mathcal{C}$ 是PAC可学习的.

如果一个概念类 $\mathcal{C}$ 是PAC 可学习的，则学习算法能够在观察一定数量的样本后以较高概率（至少 $1-\delta$）返回目标概念 c 近似正确的假设（泛化错误率小于 ϵ ）

PAC学习算法

PAC可学习性描述的是概念类 $\mathcal{C}$ 和假设空间 $\mathcal{H}$ 的性质，若考虑到对应学习算法 $\mathfrak{L}$ 的时间复杂度，则有：

PAC学习算法(PAC Learning Algorithm)：

若学习算法 $\mathfrak{L}$ 使概念类 $\mathcal{C}$ PAC可学习的，且 $\mathfrak{L}$ 的运行时间也是多项式函数

$$\mathrm{poly}(1/\epsilon, 1/\delta, \mathrm{size}(\boldsymbol{x}), \mathrm{size}(c))$$

则称概念类 $\mathcal{C}$ 是高效PAC可学习(efficiently PAC learnable)的，称 $\mathfrak{L}$ 为概念类 $\mathcal{C}$ 的PAC学习算法。

样本复杂度

假定学习算法 $\mathfrak{L}$ 处理每个样本的时间为常数, 则 $\mathfrak{L}$ 的时间复杂度等价于样本复杂度:

样本复杂度(Sample Complexity):

满足 PAC 学习算法 $\mathfrak{L}$ 所需的 $m \geq \text{poly}(1/\epsilon, 1/\delta, \text{size}(x), \text{size}(c))$ 最小的 m, 称为学习算法 $\mathfrak{L}$ 的样本复杂度.

PAC框架的意义

PAC学习给出了一个抽象地刻画机器学习能力的框架，基于这个框架能对很多重要问题进行理论探讨：

研究学习任务在什么样的条件下可学得较好的模型？

研究学习任务需多少训练样本才能获得较好的模型？

研究算法在什么样的条件下可进行有效的学习？

……

可分情形：恰PAC可学习

恰PAC可学习(properly PAC learnable):

假设空间 $\mathcal{H}$ 包含了学习算法 $\mathfrak{L}$ 所有可能输出的假设，在PAC学习中假设空间 $\mathcal{H}$ 与概念类 $\mathcal{C}$ 完全相同，即 $\mathcal{H}=\mathcal{C}$ ，这称为恰PAC可学习.

直观上看，这意味着学习算法的能力与学习任务“恰好匹配.”

然而在现实应用中，由于对概念类 $\mathcal{C}$ 通常一无所知，设计一个假设空间与概念类恰好相同的学习算法通常是不切实际的

研究的重点：假设空间与概念类不同的情形，即 $\mathcal{H}\neq\mathcal{C}$.

不可分情形

对较为困难的学习问题，目标概念 c 往往不存在于假设空间 $\mathcal{H}$ 中，假定对于任何 $h \in \mathcal{H}$，$\widehat{E}(h) \neq 0$，也就是说，$\mathcal{H}$ 中的任意一个假设都会在训练集上出现或多或少的错误.

【Hoeffding 不等式】 对 m 个独立随机变量 $X_i \in [0,1]$ $(i = 1, \ldots, m)$, 有

$$P\left(\frac{1}{m}\sum_{i=1}^{m} X_i - \mathbb{E}\left[\frac{1}{m}\sum_{i=1}^{m} X_i\right] \geqslant \epsilon\right) \leqslant e^{-2m\epsilon^2} . \tag{1.26}$$

基于Hoeffding不等式可得

引理2.1：若训练集 D 包含 m 个从分布 $\mathcal{D}$ 上独立同分布采样而得的的样本，$0 < \epsilon < 1$ ，则对任意 $h \in \mathcal{H}$ ，有

$$P\big(\widehat{E}(h) - E(h) \geqslant \epsilon\big) \leqslant \exp(-2m\epsilon^2) , \tag{2.5}$$

$$P\big(E(h) - \widehat{E}(h) \geqslant \epsilon\big) \leqslant \exp(-2m\epsilon^2) , \tag{2.6}$$

$$P\Big(\big|E(h) - \widehat{E}(h)\big| \geqslant \epsilon\Big) \leqslant 2\exp(-2m\epsilon^2) . \tag{2.7}$$

不可分情形

定理2.1：若训练集 D 包含 m 个从分布 $\mathcal{D}$ 独立同分布采样而得的样例，$0 < \epsilon < 1$ ，则对任意 $h \in \mathcal{H}$，下式至少以 $1-\delta$ 的概率成立：

$$\widehat{E}(h) - \sqrt{\frac{1}{2m}\ln\frac{2}{\delta}} < E(h) < \widehat{E}(h) + \sqrt{\frac{1}{2m}\ln\frac{2}{\delta}} \,. \qquad (2.8)$$

上述推论表明，样例数目 m 较大时，h 的经验误差是其泛化误差很好的近似.

不可分情形

定理2.1的证明:

不妨令

$$\delta = 2\exp(-2m\epsilon^2)\ , \tag{2.9}$$

即

$$\epsilon = \sqrt{\frac{1}{2m}\ln\frac{2}{\delta}}\ , \tag{2.10}$$

将上面两个式子代入 $P\left(\left|E(h)-\widehat{E}(h)\right| \geqslant \epsilon\right) \leqslant 2\exp(-2m\epsilon^2)\ .$ (2.7)

得

$$P\left(\left|E(h)-\widehat{E}(h)\right| \geqslant \sqrt{\frac{1}{2m}\ln\frac{2}{\delta}}\right) \leqslant \delta\ , \tag{2.11}$$

即至少以 $1-\delta$ 的概率有

$$\left|E(h)-\widehat{E}(h)\right| < \sqrt{\frac{1}{2m}\ln\frac{2}{\delta}}\ , \tag{2.12}$$

即式（2.8）成立，定理2.1得证 □

不可分情形

当 $c \notin \mathcal{H}$ 时，

学习算法 $\mathfrak{L}$ 无法学得目标概念 c 的 ϵ 近似.

当假设空间 $\mathcal{H}$ 给定时，其中必存在一个泛化误差最小的假设

找出此假设的 ϵ 近似也不失为一个较好的目标.

$\mathcal{H}$ 中泛化误差最小的假设表示为 $\operatorname{argmin}_{h \in \mathcal{H}} E(h)$

于是，以此为目标可将PAC学习推广到 $c \notin \mathcal{H}$ 的情形，称为不可知学习(agnostic learning).相应的，我们可以定义不可知PAC可学习：

不可知PAC可学习

不可知 PAC 可学习 (agnostic PAC learnable):

令 m 表示从分布 $\mathcal{D}$ 中独立同分布采样得到的样例数目，$0 < \epsilon, \delta < 1$ ，对所有分布 $\mathcal{D}$，若存在学习算法 $\mathfrak{L}$ 和多项式函数 $poly(.,.,.,.)$，使得对于任何 $m \geqslant poly(1/\epsilon, 1/\delta, size(x), size(c))$，$\mathfrak{L}$ 能从假设空间 $\mathcal{H}$ 中输出满足下式的假设 h：

$$P\big(E(h) - \min_{h' \in \mathcal{H}} E(h') \leqslant \epsilon\big) \geqslant 1 - \delta \ , \tag{2.13}$$

则称假设空间 $\mathcal{H}$ 是不可知PAC可学习的.

若此处学习算法 $\mathfrak{L}$ 的运行时间也是多项式函数 $poly(1/\epsilon, 1/\delta, size(x), size(c))$ 则称假设空间 $\mathcal{H}$ 是高效不可知PAC可学习的,学习算法 $\mathfrak{L}$则称为假设空间$\mathcal{H}$的不可知PAC学习算法, 满足上述要求的最小m称为学习算法$\mathfrak{L}$ 的样本复杂度.

PAC框架的要点

- PAC框架是一种分布无关的模型，未对产生样本的分布 $\mathcal{D}$ 作任何假设。
- 训练样本和用来计算错误率的测试样本都来自于同一个分布 $\mathcal{D}$. 要使PAC模型能够得到推导，这是一个必不可少的假定。
- PAC模型处理的是某个概念类 $\mathcal{C}$ 的可学习性，而不是某一个特定的概念. 通常目标概念 $c \in \mathcal{C}$ 对于学习算法是未知的。

复杂度

假设空间$\mathcal{H}$的复杂度是影响可学习性的重要因素之一.一般而言，$\mathcal{H}$ 越大，其包含的任意目标概念的可能性越大，但从中找到某个具体目标概念的难度就越大

$|\mathcal{H}|$有限时，我们称 $\mathcal{H}$ 为有限假设空间，否则称为无限假设空间.

有限假设空间包含的概念数有限，可以用概念个数$|\mathcal{H}|$来衡量其复杂度

对于无限假设空间，需使用$|\mathcal{H}|$之外的方法度量假设空间的复杂度：

- VC 维（Vapnik-Chervonekis dimension）
- Rademacher 复杂度（Rademacher Complexity）

复杂度的概念和性质将在第3章进行进一步介绍

泛化性

PAC可学习考虑的是学习算法 $\mathfrak{L}$ 输出假设的泛化误差与最优假设的泛化误差之间的差别(在可分情况下，最优假设的泛化误差为0) .

由于真实分布 $\mathcal{D}$ 未知，泛化误差通常无法直接计算.

不过，由于经验误差与泛化误差有密切联系，我们可以借助经验误差来进行比较，于是有必要考虑经验误差与泛化误差之间的差距，我们将在第4章进行讨论.

分析实例

证明一个概念类是PAC可学习的，需显示出存在某个学习算法，它在使用了一定数量的样本后能够PAC辨识概念类.

将分析以下几个概念类的PAC可学习性：

- 布尔合取式的学习
- 3-DNF和3-CNF的学习
- 轴平行矩形的学习

布尔合取式的学习

令示例 $\boldsymbol{x} \in \mathcal{X}_n = \{0,1\}^n$ 表示对 n 个布尔变量 $b_i\ (i = 1, \cdots, n)$ 的一种赋值。

布尔合取式（Boolean Conjunctions）：

形如 $b_i,\ \neg b_i$ 的文字所构成的合取式.

例如: $c = b_1 \wedge \neg b_3 \wedge b_4$ 意味着对于示例集 $\{\boldsymbol{x} \in \mathcal{X}_n : x_1 = 1, x_3 = 0, x_4 = 1\}$ 有 $c(\boldsymbol{x}) = 1$

所有这样的概念类就组成了布尔合取式概念类 $\mathcal{C}_n$ ， 下面证明概念类 $\mathcal{C}_n$ 是PAC可学习的.

布尔合取式的学习

下面证明布尔合取式概念类 $\mathcal{C}_n$是PAC可学习的:

对假设空间 $\mathcal{H} = \mathcal{C}_n$，根据PAC可学习定义，需要找到学习算法 $\mathfrak{L}$，使得存在多项式函数 $\mathrm{poly}(\cdot,\cdot,\cdot,\cdot)$，当样本集大小 $m \geq \mathrm{poly}(1/\epsilon, 1/\delta, \mathrm{size}(\boldsymbol{x}), \mathrm{size}(c))$ 时，$\mathfrak{L}$输出的假设满足PAC可学习的要求 $P(E(h) \leqslant \epsilon) \geqslant 1-\delta$. $\mathrm{size}(\boldsymbol{x})$ 和 $\mathrm{size}(c)$ 对应于合取式中的文字个数，$\forall c \in C_n$有 $\mathrm{size}(\boldsymbol{x}) = \mathrm{size}(c) \leq 2n$ (考虑 b_i和 $\neg b_i$，一共有 $2n$个布尔文字)，因此样本集大小 m 应该是关于 $1/\epsilon, 1/\delta, n$ 的多项式.

布尔合取式的学习

下面证明布尔合取式概念类 $\mathcal{C}_n$是PAC可学习的:

构造这样一个算法:

初始 $h = b_1 \wedge \neg b_1 \wedge \cdots \wedge b_n \wedge \neg b_n$, 对于任何赋值初始 h 都返回 0 .

算法忽略样本中所有反例, 仅使用样本集中的正例来进行学习.

对于样本集中的正例 $(\boldsymbol{x}, 1)$, 算法按下面的操作来更新:

h: $\forall i \in \{1, \cdots, n\}$, 若 $x_i = 0$ 则从 h 中删除 b_i, 若 $x_i = 1$ 则从 h 中删除$\neg b_i$.

这样, 算法就从 h 中删除了所有与正例矛盾的文字.

假设目标概念为 c , 则 c 包含的文字在上述任何时刻仍出现在 h 中. 这是由于初始 h 包含所有文字, 一个文字仅当在某个正例中的对应值为 0 才会被删除, 而 c 中文字在任何正例中的对应值都不会为0 .

布尔合取式的学习

下面证明布尔合取式概念类 $\mathcal{C}_n$是PAC可学习的:

现在考虑出现在 h 中但未出现在 c 中的文字 $\tilde{b}$. 对满足 $\tilde{b}=0$的正例 $\boldsymbol{x}$, h 将由于包含 $\tilde{b}$ 而在 $\boldsymbol{x}$ 上出错; 但同时, 样例 $\boldsymbol{x}$ 也恰好能使算法从 h 中删除 $\tilde{b}$.

令 $P(\tilde{b})$ 表示此类样例出现的概率(这里我们把 $\tilde{b}$ 也看作一个概念), 有:

$$P(\tilde{b})=P_{\boldsymbol{x}\in\mathcal{D}}(c(\boldsymbol{x})=1\wedge\tilde{b}(\boldsymbol{x})=0)\ . \tag{2.14}$$

由于 h 所犯的每个错误都可以归因于 h 中至少有一个文字 $\tilde{b}$, 从而可得:

$$E(h)\leqslant P(\cup_{\tilde{b}\in h}\tilde{b})\leqslant\sum_{\tilde{b}\in h}P(\tilde{b})\ . \tag{2.15}$$

上式中第二个不等号是由于Union Bound不等式 $P(X\cup Y)\leqslant P(X)+P(Y)$

我们称满足 $P(\tilde{b})\geq\frac{\epsilon}{2n}$ 的文字 $\tilde{b}$ 为 “坏字” . 若 h 不包含任何坏字, 则有

$$E(h)\leqslant\sum_{\tilde{b}\in h}P(\tilde{b})\leqslant 2n\cdot\frac{\epsilon}{2n}=\epsilon\ . \tag{2.16}$$

布尔合取式的学习

下面证明布尔合取式概念类 $\mathcal{C}_n$是PAC可学习的:

对任何一个给定的坏字 $\tilde{b}$ ，随机抽取一个样例导致其被删除的概率为 $P(\tilde{b})$ ，于是，算法在使用了 m 个样例后坏字 $\tilde{b}$ 仍未从 h 中被删除的概率至多为 $(1-\epsilon/2n)^m$. 考虑所有 $2n$个文字, 则 h 中存在坏字未被删除的概率至多为 $2n(1-\epsilon/2n)^m$. h 不包含任何坏字得概率至少为$1-2n(1-\epsilon/2n)^m$,由(2.16):

$$P\big(E(h)\leq\epsilon\big)\geq 1-2n\left(1-\frac{\epsilon}{2n}\right)^m \ . \tag{2.17}$$

于是，欲使 $P(E(h)\leq\epsilon)\geq 1-\delta$ 成立，仅需:

第一个不等号成立是由于 $(1-x)^m\leqslant e^{-mx}$.

$$\left(1-\frac{\epsilon}{2n}\right)^m\leqslant\exp\left(-\frac{m\epsilon}{2n}\right)\leqslant\frac{\delta}{2n}\ , \tag{2.18}$$

即

$$m\geqslant\frac{2n}{\epsilon}\ln\frac{2n}{\delta}\ . \tag{2.19}$$

上面构造的算法仅需(2.19)中的样本数，就能以至少 $1-\delta$ 的概率得到满足 $E(h)\leq\epsilon$ 的假设h , 于是根据可学习定义可知概念类$\mathcal{C}_n$ 是PAC可学习的.

□

布尔合取式的学习

$$m \geqslant \frac{2n}{\epsilon} \ln \frac{2n}{\delta} . \tag{2.19}$$

事实上，由于该问题的假设空间有限可分，因此其必然是PAC可学习的. 但若直接使用最简单的经验风险最小化算法(对每个假设逐一计算经验误差，然后返回经验误差最小的假设)所推出的样本复杂度上界不如这里的紧致。

注意到，学习算法 $\mathfrak{L}$处理每个样例$(\boldsymbol{x}, y)$所需的计算时间至多为 n 的线性函数(学习算法 $\mathfrak{L}$仅利用正例进行更新，对于正例 $(\boldsymbol{x}, 1), \forall i \in \{1, \cdots, n\}$,若 $x_i = 0$ 则从 h 中删除 b_i; 若 $x_i = 1$ 则从 h 中删除 $\neg b_i$)

因此概念类 $\mathcal{C}_n$是高效PAC可学习的.

3-DNF和3-CNF的学习

在PAC学习理论中，学习算法 $\mathfrak{L}$ 从给定的假设空间 $\mathcal{H}$ 中输出假设来逼近目标概念. 若 $\mathfrak{L}$ 的运行时间是多项式函数 $\mathrm{poly}(1/\epsilon, 1/\delta, \mathrm{size}(\boldsymbol{x}), \mathrm{size}(c))$ 则假设空间 $\mathcal{H}$ 是高效PAC可学习的.

一个假设空间是否高效PAC可学习，有时还取决于假设的具体表示形式. 本节给出一个具体的例子.

3-DNF和3-CNF的学习

先给出析取范式和合取范式的定义：

析取范式(Disjunctive Normal Form, 简称DNF)：

亦称析合范式, 是多项布尔合取式的析取. k-DNF是k个合取项的析取.

例如：$(x_1 \wedge \neg x_2 \wedge x_3) \vee (\neg x_1 \wedge x_3) \vee (\neg x_1 \wedge x_2)$ 是一个3-DNF公式.而之前讨论的布尔合取式可以看作1-DNF公式.

合取范式(Conjunctive Normal Form, 简称CNF)：

亦称合析范式, 是多项布尔析取式的合取. k-CNF是若干个析取项的合取, 每个析取项中至多包含k个文字.

例如：$(x_1 \vee \neg x_2 \vee x_3) \wedge (\neg x_1 \vee x_3)$ 是一个3-CNF公式.

3-DNF和3-CNF的学习

已经知道3-DNF这个概念类不是高效PAC可学习的，除非 $\mathcal{RP} = \mathcal{NP}$.

[Kearns and Vazirani, 1994].

现在我们把3-DNF换一种表示形式：

注意到布尔代数中 $\vee$ 对 $\wedge$ 满足分配律，即对于布尔变量 a, b, e, f 有

$$(a \wedge b) \vee (e \wedge f) = (a \vee e) \wedge (a \vee f) \wedge (b \vee e) \wedge (b \vee f) \ , \qquad (2.20)$$

即2-DNF公式可以等价转化为一个每项至多包含2个文字的布尔合取式. 一般的，一个k-DNF公式可以等价转化为一个k-CNF公式. 因此，对3-DNF概念类的学习可以等价转化为对3-CNF概念类的学习.

3-DNF和3-CNF的学习

下面证明3-CNF这个概念类是高效PAC可学习的:

证明的主要思路是将3-CNF的PAC学习问题归约(reduce)为2.3.1节已证明的布尔合取式概念类的PAC学习问题.

对于包含原始的 n 个布尔变量的集合 $B=\{b_1,\cdots,b_n\}$, 考虑其中任意三个变量 $u,v,w\in B$ 形成的三元组, 构造一个新的布尔变量集合 $A=\{a_{u,v,w}=u\vee v\vee w\}$, $|A|=(2n)^3$. 注意到 $u=v=w$ 时有 $a_{u,v,w}=u$, 因此 $B\subseteq A$, 即所有原来的变量都包含在 A 中. 于是, B 上任意的3-CNF概念 c 都能转化为 A 上的布尔合取式概念 c', 仅需把 $(u\vee v\vee w)$ 替换为 $a_{u,v,w}$ 即可.

这样就把3-CNF概念的学习通过简单替换操作等价转化为了一个在更大变量集合上的布尔合取式概念的学习, 而上一节已经证明了布尔合取式概念类是高效PAC可学习的, 于是3-CNF概念类也是高效PAC可学习的.

□

3-DNF和3-CNF的学习

如前所述，任何一个3-DNF公式都可以等价转化为一个3-CNF公式. 因此，若允许假设表示成3-CNF公式的形式，则3-DNF概念类是高效PAC可学习的；但若必须把假设表示成3-DNF公式形式，则3-DNF概念类不是高效PAC可学习的.

进一步，这一论断 $\forall k \geq 2$ 的k-DNF 和k-CNF 都成立. 这揭示出一个从PAC学习理论得到的重要洞察：即便对同一个概念类，选择不同的表示方式可能会导致不同的可学习性结论.

在实际应用中，对现实问题进行不同的形式化抽象，可能导致性质相去甚远的假设空间表示，进而导致迥然不同的结果

轴平行矩形的学习

有限假设空间总能通过一种通用方法（对每个假设逐一计算经验误差，然后返回经验误差最小的假设）进行PAC学习。无限假设空间则不存在这种方法，不过某些情况下可通过概念类本身特性构造学习算法进行分析，例如：

轴平行矩形(Axis-Parallel Rectangle, 简称APR)：

轴平行矩形是平面 $\mathbb{R}^2$上四条边均与坐标轴平行的矩形区域. $\mathbb{R}^2$中每个点对应于一个数据样本，即 $\mathcal{X} = \mathbb{R}^2$. 概念 c 是某个特定的轴平行矩形，对该矩形中的点 $\boldsymbol{x}$ 有 $c(\boldsymbol{x}) = 1$，该矩形之外的点 $c(\boldsymbol{x}) = -1$.概念类 $\mathcal{C}$ 是 $\mathbb{R}^2$ 上所有轴平行矩形的集合.

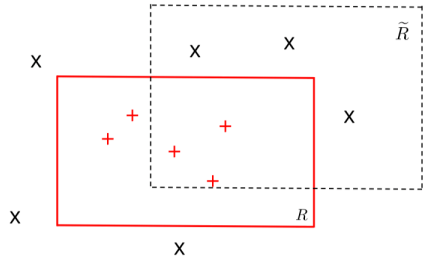

图 2.1 轴平行矩形目标概念 R 与假设 $\tilde{R}$. 图中红色 + 为正例, 黑色 × 为反例

[Shalev-Shwartz and Ben-David, 2014].

轴平行矩形的学习

下面证明轴平行矩形概念类是高效PAC可学习的:

图2.1中轴平行矩形 R 表示目标概念, $\tilde{R}$ 表示一个假设. 由图中可看出, $\tilde{R}$ 的错误区域为 $(R-\tilde{R})\cup(\tilde{R}-R)$, 即位于 R 内但在 $\tilde{R}$ 外的区域, 以及在 $\tilde{R}$ 内但在 R 外的区域. $\tilde{R}$ 会将前一个区域中的点错误地判断为反例, 而将后一个区域中的点错误地判断为正例.

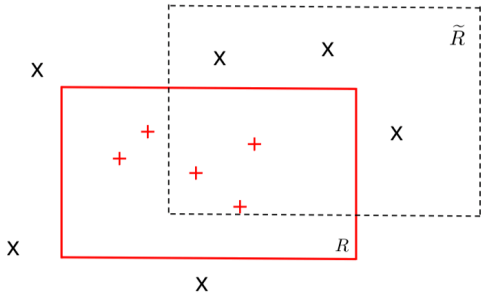

图 2.1 轴平行矩形目标概念 R 与假设 $\tilde{R}$. 图中红色 + 为正例, 黑色 × 为反例

轴平行矩形的学习

下面证明轴平行矩形概念类是高效PAC可学习的:
考虑这样一个简单的学习算法 $\mathcal{L}$:
对于给定数据集 D , $\mathcal{L}$ 输出一个包含了 D 中所有正例的最小轴平行矩形 R^D .

如图2.2所示. 显然, R^D中的点一定包含在目标概念 R 中, 因此 R^D 不会将反例误判为正例, 它犯错误的区域都包含在 R 中. 令 $P(R)$ 表示 R 区域的概率质量, 即按照分布 $\mathcal{D}$ 随机生成的点落在区域 R 中的概率. 由于算法 $\mathcal{L}$ 的错误仅可能出现在 R 内的点上, 不妨设 $P(R) > \epsilon$, 否则无论输入什么训练样本集 D , R^D的错误率都不会超过 ϵ .

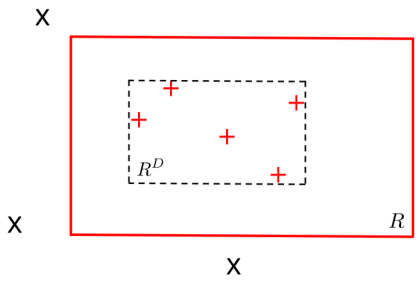

图 2.2 学习算法 $\mathcal{L}$ 输出的包含了数据集 D 中所有正例的最小轴平行矩形 R^D

轴平行矩形的学习

下面证明轴平行矩形概念类是高效PAC可学习的:

因为 $P[R] > \epsilon$ ，所以我们可以沿 R 的四条边定义4个轴平行矩形区域 r_1, r_2, r_3, r_4 ，使得每个区域的概率质量均为 $\epsilon/4$ ，如图2.3所示.

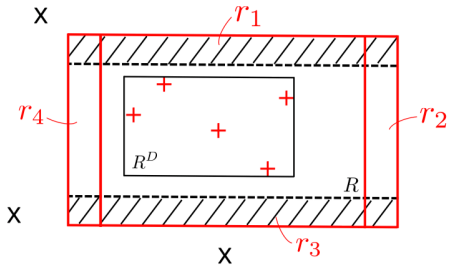

图 2.3 区域 $r_1, \cdots, r_4$ 的位置情况

于是, 有 $P[r_1 \cup r_2 \cup r_3 \cup r_4] \leq \epsilon$.

有一个重要的观察: 由于 R^D 位于 R 内部, 且它和 r_1, r_2, r_3, r_4 都是轴平行矩形, 因此若 R^D 与 r_1, r_2, r_3, r_4 都相交, 则对于任何一个 r_i $(i = 1, \cdots, 4)$, R^D 都必有一条边落在 r_i 之内. 此时 R^D 的错误区域被这4个区域完全覆盖, 其概率质量 $P[R^D] \leq \epsilon$.

轴平行矩形的学习

下面证明轴平行矩形概念类是高效PAC可学习的:

于是, 若泛化误差 $E(R^D) > \epsilon$, 则 R^D 必然至少与 r_1, r_2, r_3, r_4 中的某一个不相交. 数据集 D 中的每个样本是从分布 $\mathcal{D}$ 随机采样得到, 其出现在 r_i 中的概率为 $\epsilon/4$. 设 D 包含 m 个样本, 则有

$$
\begin{aligned}
P_{D\sim\mathcal{D}^m}\left(E(R^D) > \epsilon\right) &\leq P_{D\sim\mathcal{D}^m}\left(\cup_{i=1}^{4}\{R^D \cap r_i = \emptyset\}\right) \\
\boxed{\text{由于 } P(X\cup Y) \leqslant P(X) + P(Y)} \quad &\leq \sum_{i=1}^{4} P_{D\sim\mathcal{D}^m}\left(\{R^D \cap r_i = \emptyset\}\right) \qquad (2.21) \\
&\leq 4(1-\epsilon/4)^m \\
\boxed{(1-x)^m \leqslant e^{-mx}.} \quad &\leq 4e^{-m\epsilon/4}.
\end{aligned}
$$

令 $4e^{-m\epsilon/4} \leq \delta$ 即可确保

$$
\begin{aligned}
P_{D\sim\mathcal{D}^m}\left(E(R^D) \leq \epsilon\right) &= 1 - P_{D\sim\mathcal{D}^m}\left(E(R^D) > \epsilon\right) \qquad (2.22) \\
&\geq 1-\delta \ ,
\end{aligned}
$$

轴平行矩形的学习

下面证明轴平行矩形概念类是高效PAC可学习的:

$$\begin{aligned} P_{D\sim\mathcal{D}^m}\left(E(R^D)\leq\epsilon\right) =& 1-P_{D\sim\mathcal{D}^m}\left(E(R^D)>\epsilon\right) \\ \geq& 1-\delta\ , \end{aligned} \tag{2.22}$$

于是,

$$m\geqslant\frac{4}{\epsilon}\ln\frac{4}{\delta}\ . \tag{2.23}$$

上面构造的算法仅需(2.23)的样本数，就能以至少 $1-\delta$ 的概率得到满足 $E(R^D)\leq\epsilon$ 的假设R^D，且该学习算法所涉及的 $\mathbb{R}^2$ 平面上轴平行矩形的计算时间为常数（一个轴平行矩形可由四个角点确定），于是可知轴平行矩形概念类是高效PAC可学习的.

□