VC Theory: Vapnik–Chervonenkis Dimension

http://freemind.pluskid.org/slt/vc-theory-vapnik-chervonenkis-dimension

posted on Free Mind on July 30, 2012
generated with pandoc on December 3, 2015
category: Statistical Learning Theory

tags: Binary Classification

上一次我们介绍了通过 Symmetrimization 的方法进行变形，从而得到了如下形式的不等式：

$$P\left(\sup_{f\in\mathcal{F}}(E(f)-E_N(f))>\epsilon\right)\leq 2P\left(\sup_{f\in\mathcal{F}}(E_N^*(f)-E_N(f))>\frac{\epsilon}{2}\right)$$

其中左边是我们感兴趣（希望 bound 住）的量，我们已经成功地把它转化为右边从某种意义上来说 “有限” 的量，本文中我们就要来对右边的部分进行分析，得到它的一个上界，从而回答我们最开始提出的 “Can we learn？” 的问题。

于是让我们把注意力集中到不等式的右边。首先我们注意到那个看起来很恐怖的对 $f\in\mathcal{F}$ 的上确界，其实等价于在一个有限的集合上求上确界，这个集合就是 $\mathcal{F}$ 到 $\{z_i\}_{i=1}^N$ 和 $\{z_i^*\}_{i=1}^N$ 上的投影：

$$\mathcal{F}(z_1,\ldots,z_N,z_1^*,\ldots,z_N^*)=\{(f(z_1),\ldots,f(z_N),f(z_1^*),\ldots,f(z_N^*)|f\in\mathcal{F}\}$$

这是由 $2N$ 维 binary 向量构成的集合，显然它的元素个数不超过 2^{2N}，于是我们可以简单地用 Union Bound 来进行处理。为了符号上的方便，以下我们将 $\mathcal{F}(z_1,\ldots,z_N,z_1^*,\ldots,z_N^*)$ 简单记作 $\mathcal{F}^P$（表示 Project 到数据上的 loss class）。

$$\begin{aligned}P\left(\sup_{f\in\mathcal{F}}(E_N^*(f)-E_N(f))>\frac{\epsilon}{2}\right)&\leq|\mathcal{F}^P|P(E_N^*(f)-E_N(f)>\frac{\epsilon}{2})\\&\leq 2^{2N}e^{-\epsilon^2N/2}\\&=\exp\left(\left(2\log 2-\frac{\epsilon^2}{2}\right)N\right)\end{aligned}\tag{1}$$

其中红色的部分是用 $\mathcal{F}^P$ 的元素个数的最简单的上界来实现的，而蓝色的部分则是直接套用最普通的（针对单个固定 f 的）Hoeffding 不等式。最后看起来似乎是得到了一个上界，比起直接在 $\mathcal{F}$ 上用 Union Bound 得到 ∞ 是进了一步，但是其实这个上界仍然不太有用。

因为 $2\log 2\approx 1.39$，几乎任何一个合理的（比如说，小于 1 的）ϵ 都会让我们的上界的指数部分是正数，从而随着 N 的增长迅速膨胀。也就

是说，数据点越多我们的上界反而越差。再看一个极端情况，当 $N = 0$ 的时候我们差不多能得到一个最好的上界，那就是 $e^0 = 1$，不过，这仍然是毫无意义的，因为任何一个概率值本来就是 ≤ 1 的呀。

让我们来稍微做一下反思：首先 Heoffding 不等式给我们带来了一个指数形式的上界，并且指数部分是随着 N 增大负增长的，这是很好的：只要让 N 增加，很快就能得到很好的上界数值。但是我们用了 Union Bound 之后，在前面乘上了一个系数，虽然这个系数是有限数，但是比较不幸的是它也是一个指数函数，并且指数部分随着 N 正增长。这样一来就把 Hoeffding 不等式给我们带来的好处一点不剩地抵消掉了。为了解决这个问题，我们希望前面乘的系数能小一点，这也并不是完全没有希望的，因为我们乘的是 2^{2N}，这是 $\mathcal{F}^P$ 所可能拥有的最多的元素个数，而如果它所拥有的实际元素个数比这个少的话，我们就有希望了！

接下来我们就来分析 $\mathcal{F}^P$ 的元素个数。这里我们先抛开之前的种种概念，来将问题抽象一下，顺便重新整理一下记号。首先我们有一个集合 $\mathcal{Z}$，以及一个 binary 函数的集合 $\mathcal{F} \subset \{0,1\}^{\mathcal{Z}}$，在这里的分析中我们不需要 $\mathcal{Z}$ 上有什么概率测度之类的。为了记号上方便，我们将 $\{z_i\}_{i=1}^N$ 简记为 $z_{1:N}$。$\mathcal{F}$ 到 $z_{1:N}$ 上的投影 $\mathcal{F}(z_{1:N})$ 的定义仍然和原来一样的，是由一些 N 维 binary 向量组成的集合。

显然 $\mathcal{F}(z_{1:N})$ 的元素个数同时依赖于 $\mathcal{F}$ 和 $z_{1:N}$ 的选取。例如，我们令 $\mathcal{Z} = \mathbb{R}^2$，并任意选取两个不重合的点 z_1 和 z_2。如果 $\mathcal{F}$ 是所有 Positive Rays 构成的集合[1]，那么

$$\mathcal{F}(z_1, z_2) = \{(0,0), (1,1), (0,1)\}$$

也就是说，此时 $|\mathcal{F}(z_1, z_2)| = 3$，严格小于 $2^2 = 4$，如图 1 所示。但是如果将 $\mathcal{F}$ 换成 Positive Intervals[2]，很容易知道我们将能够得到所有的四种可能的元素，我就不专门画图了。

像这种对于一个 $\mathcal{F}$ 和一个数据点集合 $z_{1:N}$，如果 $\mathcal{F}$ 投影到 $z_{1:N}$ 上能参生所有可能的 binary 向量，换句话说，如果 $|\mathcal{F}(z_{1:N})| = 2^N$ 的话，我们称 $\mathcal{F}$ shatters $z_{1:N}$。所以，Positive Intervals 可以 shatter 刚才的两个点，而 Positive Rays 却不行。

直观上来看，Positive Intervals 比 Positive Rays 更复杂（比如说，每个 positive interval 比 positive ray 的参数要多一个），所以投影到同样的两个点上之后，前者能产生更多的 binary 向量。回忆一下我们开始这些分析的初衷：2^N 这个系数乘到 Hoeffding 不等式的上界前面太大了，因此我们希望找到投影之后的元素（严格）小于 2^N 的情况。从这里的两个例子我们也可以大概看到一些端倪：如果 $\mathcal{F}$ 很复杂，能 shatter 给定的数据集的话，情况似乎就很悲观，但是如果我们选用简单一些的 $\mathcal{F}$，好像就有希望了。所以接下来让我们再接再厉，将这里的 “简单” 和 “复杂” 这两个模糊的概念严格地刻画出来。

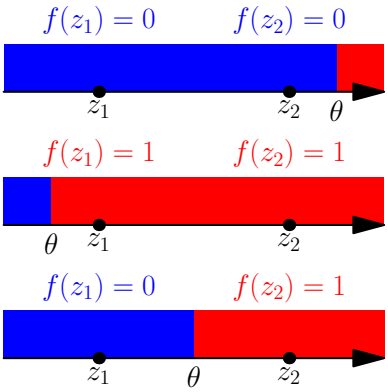

Figure 1: Positive rays projected on two points.

[1] 这个在课堂上定义过了，每个 positive ray f_θ 由一个参数 $\theta \in \mathbb{R}$ 确定，其取值为 $f_\theta(z) = \mathbf{1}\{z \geq \theta\}$。

[2] 同样在课中定义过，每个 positive interval $f_{a,b}$ 定义为 $f_{a,b}(z) = \mathbf{1}\{z \in [a,b]\}$。

刚才我们说过了，$|\mathcal{F}(z_{1:N})|$ 同时依赖于 $\mathcal{F}$ 和 $z_{1:N}$，我们刚才的例子已经说明了在 $z_{1:N}$ 一样的情况下，不同的 $\mathcal{F}$ 会得到不同的结果。下面我们再来举例说明一下在固定 $\mathcal{F}$ 的情况下，不同的数据点选取也会得到不同的结果。这次我们令 $\mathcal{Z} = \mathbb{R}^2$，而 $\mathcal{F}$ 则使用 Positive Rectangles，和 Positive Interval 类似的，落在给定的 rectangle 内的点取值为 1，而外面的点取值为 0。特别地，我们只考虑和坐标轴对齐的那种矩形，因此它可以由左上角和右下角的二维坐标这四个参数来确定。顺带一提，任意一个 $f \in \mathcal{F}$ 带入 $z_{1:N}$ 产生的这个 binary vector 通常称为一个 dichotomy。

在图 2 中我们展示了两种 layout，都取 $N = 4$，Layout 1 是可以被 shatter 的，虽然图中只画出了一种情况，但是剩下的情况也可以很容易给出。但是对于 Layout 2，也就是有某一个点处于其他三个点的 convex hull 内部的时候，就不好办了，当外围三个点都取值 1 的情况下，内部那个点由于被包围在 rectangle 之内，所以肯定也是取 1 而无法取到 0，所以至少有一种 dichotomy 是无法实现的，于是在这种 layout 下 Positive Rectangles 无法 shatter 这四个点。

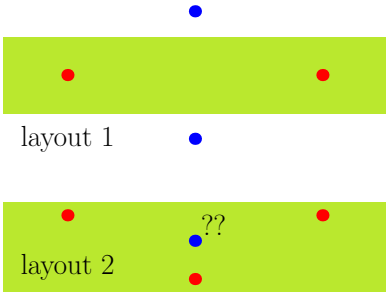

Figure 2: Layout 1 can be shattered by positive rectangles (only one dichotomy is shown), but not layout 2.

由于在学习问题中我们通常是无法控制训练数据点的选取的，所以为了能应付所有情况，我们必须最 "悲观" 地来考虑 $\mathcal{F}$ 是否能 shatter 某 N 个点的数据集这件事。

定义 1 (Growth Function) *对于给定的正整数 N 和函数空间 $\mathcal{F}$，growth function $S_{\mathcal{F}}(N)$ 的值定义为所有 N 个数据点的 layout 中 $\mathcal{F}$ 能产生的最多的 dichotomy 数：*

$$S_{\mathcal{F}}(N) = \sup_{z_{1:N}} |\mathcal{F}(z_{1:N})|$$

如果 $\mathcal{F}$ shatters $z_{1:N}$，那么必定有 $S_{\mathcal{F}}(N) = 2^N$；反过来，如果 $S_{\mathcal{F}}(N) = 2^N$，那么肯定至少存在一组 N 个点的数据 $z_{1:N}$，使得 $\mathcal{F}$ 可以 shatter $z_{1:N}$，但是我们却不能保证所有 N 个点的数据都能被 shatter。例如，对于刚才的 Positive Rectangles 而言，因为我们给出了 Layout 1 中的 4 个点是可以被 shatter 的[3]，所以 $S_{\mathcal{F}}(4) = 2^4$，但是即便如此，仍然存在像 Layout 2 这样的不能被 shatter 的 4 个点的情况。

这里很容易造成混淆，注意仔细理解一下我们 "悲观" 的意思：因为我们不希望 $z_{1:N}$ 被 shatter，那样的话表示 $\mathcal{F}$ 很复杂，难以得到合理的上界，只要任意的一组 $z_{1:N}$ 被 shatter 了，那我们就 "悲剧" 了。

回到 Positive Rectangles，我们来看一下 $S_{\mathcal{F}}(5)$。如果想要证明 $S_{\mathcal{F}}(5) = 2^5$ 的话，只要找到任意一组 5 个点的数据能被 shatter 就可以了，但是如果要证明 $S_{\mathcal{F}}(5) < 2^5$，则必须证明**任意的** 5 个点的数据集都无法被 shatter，这一般是要更加困难一些。不过对于 Positive Rectangles 来说也还是比较简单的。对于平面上的任意 5 个点，我们可以分为两种情况来考虑。

[3] 同时还因为对任意的 $\mathcal{F}$ 和 N，都有一个硬性的上界 $S_{\mathcal{F}}(N) \leq 2^N$。

第一种情况是其中至少有两个点位于和某一条坐标轴平行的上。记住我们这里处理的是和坐标轴对齐的矩形，所以对于这种情况，如果给共线的这两点分别标上 0 和 1，那么这种 dichotomy 显然是无法实现的，因为这两点要么同时位于矩形内部，要么同时位于外部。

第二种情况是不存在共线于坐标轴平行线的两点，此时最上面、最下面、最左面和最右面都必定由一个不同的点占据，而剩下的一个点位于他们“中间”，如果给这个点标上 0，而外围的点标上 1，这种 dichotomy 也无法由 Positive Rectangles 实现。

两种情况合起来，我们就证明了任意 5 个点的数据集都无法被 Positive Rectangles 所 shatter，所以 $S_{\mathcal{F}}(5) < 2^5$。此外很容易知道，对于某个 $\mathcal{F}$，如果存在 N_0，使得 $S_{\mathcal{F}}(N_0) < 2^{N_0}$，那么对任意的 $N' > N_0$，肯定也有 $S_{\mathcal{F}}(N') < 2^{N'}$。在课上这些使得 $S_{\mathcal{F}}(N) < 2^N$ 的 N 被称作 **break point**，不过我们这里要定义一个更重要（或者至少是更出名 `^_^bbb`）的量，也就是本文的标题。

定义 2 (Vapnik–Chervonenkis Dimension) *$\mathcal{F}$ 的 Vapnik–Chervonenkis Dimension，简称 VC Dimension，记作 $d_{\mathcal{F}}$，是最大的满足如下条件的整数 N*

$$S_{\mathcal{F}}(N) = 2^N$$

如果不存在这样的整数，我们记 $d_{\mathcal{F}} = \infty$。

显然所有大于 $d_{\mathcal{F}}$ 的整数都是 $\mathcal{F}$ 的 break point。VC 维的重要性在于它刻画了 $\mathcal{F}$ 的“复杂度”，这一点我们在这个 VC Theory 系列的一开始就提到了：如果 $\mathcal{F}$ 的 VC 维是有限的，那么在我们的设定下的问题就是 Learnable 的。不过，严格的证明还需要一些工作量，所以我们先来直观地感觉一下。简单来说，我们可以说 $\mathcal{F}$ 的 shatter 能力止于 $d_{\mathcal{F}}$：所有个数多于 $d_{\mathcal{F}}$ 的数据集 $\mathcal{F}$ 都无法将其 shatter，也就是说，当数据点的个数[4]大于 $d_{\mathcal{F}}$ 的时候，我们就可以在 Hoeffding 不等式的前面乘上一个比 2^N 更 nice 的系数了。

[4] 由于我们使用了 symmetrimization，所以实际上是数据点个数的二倍。

根据我们刚才的计算，Positive Rectangles 的 VC 维应该是 4，碰巧每个 positive rectangle 也是由 4 个参数所确定，这两者之间是不是有什么联系呢？遗憾的是，两者之间并没有什么必然联系，Yaser 教授[5]在课上举过一个把一堆 perceptron 串起来的例子，冗余参数的个数可以被堆到任意多个，但是它的 VC 维却永远和一个 perceptron 的 VC 维相等的；反过来的例子也有，例如这样的单参数函数集合

[5] 暂且以名字称吧，外国人的姓好难记……`^_^bbb`

$$\{\text{sign}(\sin(tz)) | t \in \mathbb{R}\}$$

它的 VC 维却是 ∞。

我们知道当 $N > d_{\mathcal{F}}$ 的时候，可以得到比 2^N 更好的系数，但是具体

是多少呢？当然形式上表示就是 $S_{\mathcal{F}}(N)$，不过这个具体数值的计算有点略复杂，课堂上 Yaser 教授举过几个例子，我们也见识过了，既然直接计算是不可行的，那么我们就来寻求一个上界吧，当然这个上界必须要比 2^N 要小，否则就毫无意义了。幸运的是，这里确实存在一个非常好的上界。

定理 1 (Vapnik and Chervonenkis, Sauer, Shelah) *设 $\mathcal{F}$ 的 VC 维 $d_{\mathcal{F}} < \infty$，则对任意正整数 N，我们有*

$$S_{\mathcal{F}}(N) \leq \sum_{i=0}^{d_{\mathcal{F}}} \binom{N}{i}$$

于是，对于 $N \leq d_{\mathcal{F}}$，$S_{\mathcal{F}}(N) = 2^N$，而当 $N > d_{\mathcal{F}}$ 时：

$$\begin{aligned}
\left(\frac{d_{\mathcal{F}}}{N}\right)^{d_{\mathcal{F}}} S_{\mathcal{F}}(N) &\leq \left(\frac{d_{\mathcal{F}}}{N}\right)^{d_{\mathcal{F}}} \sum_{i=0}^{d_{\mathcal{F}}} \binom{N}{i} \\
&\leq \sum_{i=0}^{d_{\mathcal{F}}} \left(\frac{d_{\mathcal{F}}}{N}\right)^{i} \binom{N}{i} \qquad \text{b/c. } \frac{d_{\mathcal{F}}}{N} < 1 \\
&\leq \sum_{i=0}^{N} \left(\frac{d_{\mathcal{F}}}{N}\right)^{i} \binom{N}{i} \\
&= \left(1 + \frac{d_{\mathcal{F}}}{N}\right)^{N} \\
&\leq e^{d_{\mathcal{F}}}
\end{aligned}$$

将左边的系数除到右边，得到：

$$S_{\mathcal{F}}(N) \leq \left(\frac{eN}{d_{\mathcal{F}}}\right)^{d_{\mathcal{F}}} \tag{2}$$

也就是说，$S_{\mathcal{F}}(N)$ 被一个关于 N 的 $d_{\mathcal{F}}$ 次多项式给 bound 住了。从指数函数到多项式函数不得不说是一次大跃进，因为多项式函数的增长速度和指数函数的增长速度是没法比的。于是我们迫不及待地对 (1) 进行修正。

注意原来的式子中我们处理的是一份数据加上一份 Ghost Sample，所以总数目是 $2N$，当 $d_{\mathcal{F}}$ 有限并且 $2N > d_{\mathcal{F}}$ 时，我们得到：

$$\begin{aligned}
P\left(\sup_{f \in \mathcal{F}}(E_N^*(f) - E_N(f)) > \frac{\epsilon}{2}\right) &\leq S_{\mathcal{F}}(2N) e^{-\epsilon^2 N/2} \\
&\leq \left(\frac{2eN}{d_{\mathcal{F}}}\right)^{d_{\mathcal{F}}} e^{-\epsilon^2 N/2}
\end{aligned}$$

再结合 Symmetrimization 得到的结论，我们最终得到：

$$P\left(\sup_{f\in\mathcal{F}}(E(f)-E_N(f))>\epsilon\right)\leq 2\left(\frac{2eN}{d_{\mathcal{F}}}\right)^{d_{\mathcal{F}}}e^{-\epsilon^2N/2}$$

令不等式右边等于 δ，反解 ϵ 得到

$$\epsilon=\sqrt{\frac{2}{N}\log 2+\frac{2d}{N}\log\left(\frac{2Ne}{d}\right)+\frac{2}{N}\log\frac{1}{\delta}} \tag{3}$$

换而言之，不论我们的学习算法给出什么样的 final hypothesis $f\in\mathcal{F}$，我们总能以不低于 $1-\delta$ 的概率保证

$$E(f)\leq E_N(f)+\epsilon(\delta,d_{\mathcal{F}},N)$$

其中 ϵ 依赖于 δ、N 以及 $\mathcal{F}$ 的 VC 维，它的具体定义如 (3)，式子有点复杂，不过如果我们只关注一下 N 的话，可以看到它是 $O(\sqrt{\frac{1}{N}\log N})$ 的，随着 N 的增大，这一项可以被变得任意小。

用人话总结一下的话，就是说，如果 $\mathcal{F}$ 的 VC 维是有限的，那么对于任意的精确度 ϵ 和确信程度 δ 的要求，只要我们把数据量 N 增加到足够大，就总能实现。我们把这称作是 Learnable 的。注意这里我们完全没有考虑用了什么学习算法，例如你可以用一个完全随机地返回任意一个 f 的算法，仍然能够满足我们这里给的 bound。

但是同时要注意的是我们这里的 bound 指的是 in-sample error 和 out-of-sample error 之间的差异。所以如果 in-sample error 很高的话，最终的结果也是没有多大意思的。而 in-sample error 是我们可以切实计算的，所以是否能做到让 in-sample error 很小就要看算法的好坏与问题本身的难度了（例如 Bayes Error 本身就很高的问题，再怎么都是无济于事的）。

末尾提一下我没有 cover 的一些东西，一个是常见的一些 VC 维，这个在 Yaser 教授的课里举了不少例子，例如 $\mathbb{R}^d$ 上的 perceptron 的 VC 维是 $d+1$，我在这里就不多讲了。另外我没有证明定理 1，本来也是打算要整理的，但是连写了三篇日志，完全没有力气了 :p，反正 Yaser 教授在课上已经讲得很生动详细了，另外也可以参考 [Kearns and Vazirani, 1994] 中 3.4 节给出的证明，相对比较简洁一点，但是其实本质上是一样的。

然后，这些东西还只是学习理论的冰山一角，就这个特定的问题而言，VC 维并不是刻画 $\mathcal{F}$ 复杂度的唯一量，我们所得到的 bound 也并不是已知最好的，而且只是处理 binary classification 的情况，而没有考虑 multiclass 的问题，也没有考虑诸如 Active Learning 等的情况。不同的设定下也会导出不同的问题和方法，仅从问题的 formulation 上来看的话，“古时候” 的统计学似乎比较喜欢研究给定数据的模型的情况下的一些问题，而发展到机器学习中时大家开始认为假定已知数据的分布或模

型是 “不科学” 的，因此发展的理论也主要集中在像这里这样的对 $\mathcal{Z}$ 上的任意分布都成立的背景下。然后呢，最近开始比较流行的另外一个叫做 Online Learning 或者 Theory of Individual Sequences 或者其他一些的名字的 subfield，他们认为假设数据全部采样自一个概率分布本身就是不科学的，虽然有一些人在研究 *Domain Adaptation* 之类的问题，通常假设训练数据和测试数据是来着不同的但是相关的概率分布，但是 Sequential Learning 学派更加 “激进”，直接抛弃了任何概率统计的假设，数据不必是从什么概率分布里采样出来的，而可以是任意的，更极端的情况下，数据甚至可以是某个 adversary 蓄意产生的 “最坏” 的数据——例如 spammer 和 anti-spam classifier 就是一个很典型的例子。

References

[Kearns and Vazirani, 1994] Kearns, M. J. and Vazirani, U. V. (1994). *An introduction to computational learning theory*. MIT Press, Cambridge, MA, USA.