斯坦福 ML 公开课笔记 10

事情繁忙，竟然过了三个半月才更新了斯坦福 ML 公开课笔记 10，在此对自己表达深深的歉意，没有好好的利用暗时间。在整个寒假期间，预期更新到笔记 15。在此先言明，以达到督促自己的效果。

本篇是 ML 公开课的第 10 个视频，上接第 9 个视频，都是讲学习理论的内容。本篇的主要内容则是 VC 维、模型选择(Model Selection)。其中 VC 维是上篇笔记中模型集合无限大时的扩展分析；模型选择又分为交叉检验(Cross Validation)和特征选择(Feature Selection)两大类内容。

VC 维

VC 由两位大牛的名字首字母拼合而成，Vapnik 和 Chervonenkis。在讲述 VC 维以前，先使用另外一种技巧对模型集合无限大时的一致收敛定理的推导考虑。所谓的一致收敛定理，如上篇笔记所述，就是训练误差与泛化误差随着样本数目的增大而更加接近的意思。

对于一个模型来说，比如 logistic 模型，如果有 n 个 feature，那么该模型会有 d=n+1 个参数。虽然理论上说 d 个参数的取值都有无穷多个，使得模型集合无限大。但实际上，在计算机的表达中，比如每个参数都以 64 位 Double 型表示，那么共需要 64d 位来表达这个模型集合，考虑到每个位有 0、1 两种状态，那么在计算机的表达中，这个无限大的模型集合的大小其实是 2^{64d}，即：

$$|\mathcal{H}| = 2^{64d} = k \tag{1}$$

因此，按照上一篇笔记里的结论，则有：

$$\mathrm{m} \geq \frac{1}{2\gamma^2} log \frac{2k}{\sigma} = \mathrm{O}(\frac{d}{\gamma^2} log \frac{1}{\sigma}) \tag{2}$$

所以，我们得到结论，一个包含 d 个参数的无限模型集合至少有 1-σ的概率使 $\varepsilon(\hat{h}) \leq \min_{h\in\mathcal{H}} \varepsilon(h) + 2\gamma$成立的前提是至少有与特征数目同一数量级的样本数目。

这样的结论虽然符合我们的直观感觉，但是却并不正式。使用参数数目对模型复杂度进行衡量的方法有缺陷，比如对于同样的模型，不同的表达形式就有不同的参数数目。比如，对于 logistic 模型来说，可以用公式 3 和公式 4 表示：

$$h_\theta(x) = I\{\theta_0 + \theta_1 x_1 + \cdots + \theta_n x_n \geq 0\} \tag{3}$$

$$h_{u,v}(x) = I\{(u_0^2 - v_0^2) + (u_1^2 - v_1^2)x_1 + \cdots + (u_n^2 - v_n^2)x_n \geq 0\} \tag{4}$$

虽然模型一样，但是参数数目却是两倍的关系。

为了更准确的对模型复杂度进行衡量，先介绍如下概念。

定义一：

给定一个集合$\mathbf{S} = \{\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \ldots, \boldsymbol{x}^{(d)}\}$，我们称模型集合$\mathcal{H}$可以分散 S 当且仅当对于集合 s 的任意一种标记方式，$\mathcal{H}$中总存在一种假设 h，可以将其线性分开。

比如，在二维平面中，对于三个点的一个集合，我们假设要判断的模型集合为二维平面上所有的直线，模型集合表示为$h_\theta(x) = I\{\theta_0 + \theta_1 x_1 + \theta_2 x_2 \geq 0\}$。那么则有如图 1 所示的分割面。

当然，当三个点在一条直线上或三个点重合的时候，直线集合并不能将其分散，但这无关紧要，因为只要存在一个 3 个点的集合，不论以任何的标记方式，直线集合都能将其线性分开的话，我们就可以认为直线集合能够分散的点的数目为 3 了。对于 4 个点的集合，可以证明，不管是怎样的 4 个点，对于直线集合来说，都不能将其分散。所以这里我们就得到，在二维平面下，对于直线集合，它能分散的点的最大数目为 3。从而，我们得到了 VC 维的定义。

定义二：

对一个模型集合$\mathcal{H}$来说，它的 VC 维，记为 VC($\mathcal{H}$)，是其能够分散的最大集合的大小。

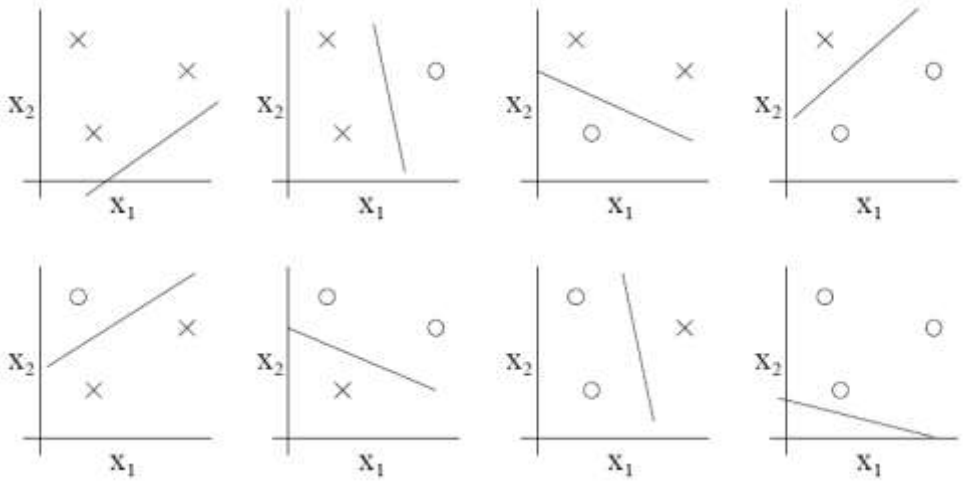

图 1 二维平面下所有直线模型集合将三点数据集分散的方式

对于上文的直线集合，它的 VC 维是 3。

更一般的，对于 n 维线性分类模型来说，它的 VC 维为 n+1。

所以，对于一个模型集合来说，不论它是有限的还是无限的，我们可以根据它的 VC 维来得到其一致收敛定理。由于 VC 维下一致收敛的证明比较繁琐，本笔记不涉及，只对定理进行解释。

定理一：

对于模型集合$\mathcal{H}$，令 d=VC($\mathcal{H}$)，那么至少以有 1-σ的概率下，对于模型集合中的所有模型 h 来说，我们有

$$|\varepsilon(h) - \hat{\varepsilon}(h)| \leq O\left(\sqrt{\frac{d}{m}log\frac{m}{d} + \frac{1}{m}log\frac{1}{\sigma}}\right) \quad (5)$$

从而，得到在至少以有 1-σ的概率下，有

$$\varepsilon(\hat{h}) \leq \hat{\varepsilon}(h^*) + O\left(\sqrt{\frac{d}{m}log\frac{m}{d} + \frac{1}{m}log\frac{1}{\sigma}}\right) \quad (6)$$

由公式 5 和公式 6，可以看到，当一个模型集合的 VC 为有限的时候，随着样本数目的变大，训练误差与泛化误差将会一致收敛。

与笔记九中类似，我们可以保持几个参数不变，得到剩余一个参数的不等式。

引理一：

为使$|\varepsilon(h) - \hat{\varepsilon}(h)| \leq \gamma$对模型集合中的所有 h 都以 1-σ的概率成立，那么样本数目 m 必须满足：

$$m = O_{\gamma,\sigma}(d) \quad (7)$$

一般情况下，VC 维和模型的参数数目线性相关，因而我们才有上面的结论，即训练模型需要的样本数目与参数呈线性关系。更一般的是，为了使模型可以达到较好的效果，需要的样本数必须与模型的 VC 维在同一数量级。

VC 维解释 SVM

由笔记 6、7、8 对 svm 的阐述可知，SVM 通过核函数将数据映射到高维空间，那么相

应的，其 VC 维应该变大，要达到较好效果所需的数据量应该增大才对。但 svm 只在原数据上就达到了比其他模型更优的效果。这是为什么呢？

虽然 svm 将数据映射到了高维空间，但是其仍然有最大间隔分类器的假设。而对于最大间隔分类器来说，其 VC 维并不依赖 x 的维度。对于最大间隔为γ的分类器来说，其在γ半径内的数据点数目设为 k，则分类器的 VC 维服从如下公式：

$$\mathrm{VC}(\mathcal{H}) \leq \left\lceil \frac{k^2}{\gamma} \right\rceil + 1 \tag{8}$$

而 SVM 算法则会自动寻找一个具有较小 VC 维的假设类，这样反而降低了 VC 维，使得数据量变得相对更加充分，提高了模型的效果。

ERM 的直观意义

ERM，经验风险最小化，笔记 9 与笔记 10 的上半部分的定理都是以 ERM 为基础进行分析的。那么 ERM 的作用到底是什么呢？

回顾笔记 9 中的训练误差的计算公式：

$$\hat{\varepsilon}(h_\theta) = \hat{\varepsilon}_S(h_\theta) = \frac{1}{m}\sum_{i=1}^{m} I\{h_\theta(x^{(i)}) \neq y^{(i)}\}$$

更简单的，我们以单个样本为例，其误差函数为$I\{h_\theta(x) \neq y\}$，很显然，这是一个非凸函数，使用机器学习的方法并不能很好的对其进行优化。因而产生了一些算法对该误差函数进行凸性近似，以期能够更好的优化。以 svm 和 logistic 为例，如图 2 所示：

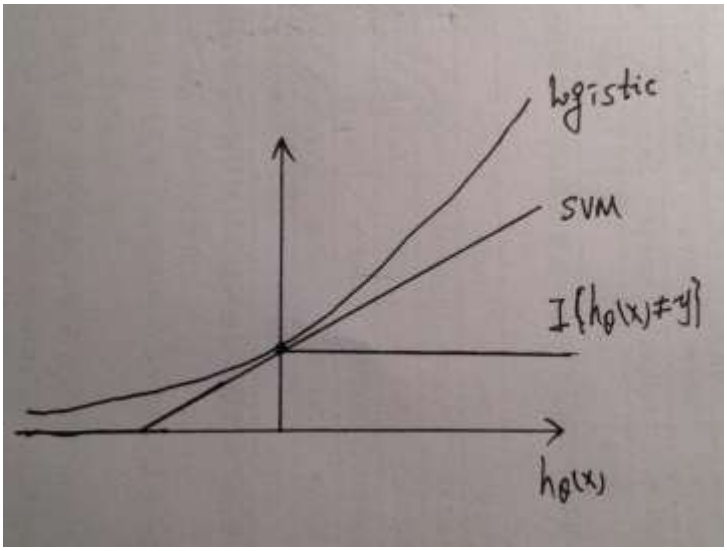

图 2 logistic 与 svm 对误差函数的近似

logistic 模型采用极大似然估计方法，它尝试令负的对数似然最小，因而如图 2 中的曲线所示。Svm 的对应关系我现在也不是太清楚，算是一个 TODO 项吧。

虽然 logistic 和 svm 都不是直接的 ERM 算法，但基于对 ERM 的近似而产生，因而可见 ERM 的一致性定理在实际中的威力。

模型选择

机器学习的模型有很多，如何在多个模型中选择最好的一个？即便对于同一个模型来说，又如何选择最好的参数？这就是本节要解决的问题。具体举几个例子，比如多项式模型如何选择阶数？svm 模型如何选择惩罚参数 C 和正则化项？局部加权回归如何选择带宽参数？

如何从神经网络与 svm 模型中选择一个较好的模型？等等。

对于模型选择，本文讲述常用的交叉检验和特征选择。

交叉检验

选择模型最简单的方式是，对每个模型 M，取训练误差最小的模型。显然，这样最终会容易选择那些过拟合的模型。简单的对其进行修改，我们就可以得到成为保留交叉验证的模型选择方法。

保留交叉验证(hold-out cross validation or simple cross validation)的做法如下：

1) 将标注数据集随机切分为 S_{train}(如 70%)和 S_{cv}(如 30%)
2) 在 S_{train} 上训练模型；
3) 在 S_{cv} 上进行测试；
4) 去测试误差最小的那个模型。

通过在非训练集上进行测试，我们得到了一个对模型更好的估计。在实际使用过程中，模型将在全部的数据集上重新训练，以利用更多的数据，达到更好的效果。

该方法的劣势在于分出过多的数据用来测试，对于标注数据难得的实际问题来说，这是不能容忍的。因而产生了如下的改进方法，成为 k 重交叉检验。

k 重交叉检验(k-fold cross validation)，做法如下：

1) 将标注数据集随机平均切分为 k 份；
2) 对于每一份来说，
   2.1) 以该份为测试集，其余份为训练集；
   2.2) 在训练集上得到模型；
   2.3) 在测试集上得到误差结果，这样就对每个样例都有一个预测结果。
3) 计算误差结果；
4) 取误差最小的模型

常用的做法是取 k=10。极端的做法取 k=m，m 为样例数，这样就变成了留一交叉验证(leave-one-out cross validation)。

特征选择

特征选择是一类比较特殊的模型选择方法。设想这样的问题，训练集中有 m 个样本，每个样本有 n 个特征，其中 n>>m。如果要使用简单的线性模型的话，那么按照之前 VC 维的分析，n 个特征会有 n+1 个参数，则需要 O(n+1)级别的样例数才能得到一个较好的模型，对于远小于 m 的样例数来说，欠拟合的风险比较大。这时，进行特征选择的意义在于降低 VC 维，使得目前的样例数目变得相对充分，从而能得到更为有效的模型。

对于 n 个特征来说，特征子集的个数有 2^n 个，如何进行选择呢？穷举法计算量太大，必然不可行。本文介绍一种启发式的算法，前向选择法(Forward Search)。

前向选择法如下：

1) 初始化特征子集为$\mathcal{F} = \Phi$
2) 对于不属于$\mathcal{F}$的每个特征，计算添加该特征后模型精度的提升
3) 选择提升最大的特征
4) 重复第 2 步和第 3 步，直到模型精度不再上升为止

该算法也被称为 wrapper model feature selection。因为它将模型的训练和评测包含在算法的内部。

根据前向选择法，我们可以很容易的得到后向选择法(Backward Search)。即每次删除对精度影响最不大的特征。

上面的方法虽然可以达到较优的特征选择结果，但是由于其反复多次调用模型训练算法，其计算量会相当的大，尤其在训练数据量比较大的时候。为了是特征选择更简便，提出了一种更简单的特征选择方法——过滤法。

特征过滤

过滤特征选择(Filter Feature Selection)，采用一种启发式的规则对特征进行评分，选择评分较优的特征，其计算量相对于前向后向搜索算法来说，非常小。

一个可能的评分函数是衡量 x_i 和 y 的相关性，从而选择出与类别标号 y 最相关的特征 x_i。而相关性可以用互信息(mutual information,MI)来表示，当 x_i 是离散型变量的时候，MI 的计算公式如下：

$$\mathrm{MI}(x_i,\mathrm{y}) = \sum_{x_i}\sum_{y} p(x_i,y) log \frac{p(x_i,y)}{p(x_i)p(y)} \tag{9}$$

其中，$p(x_i,y)$等是概率，可以通过在训练集中统计得到估计值。

事实上，互信息也可以表示为 KL(Kullback-Leibler)距离的形式：

$$\mathrm{MI}(x_i,\mathrm{y}) = \mathrm{KL}(p(x_i,y)||p(x_i)p(y)) \tag{10}$$

KL 距离的作用是衡量分布之间的差异，就此例来说，如果x_i和 y 相互独立，那么它们之间的 KL 距离为 0；如果它们之间的关联关系比较强，那么 KL 距离会变大。

使用 MI 进行衡量后，我们得到了各个特征的评分，那么选择多少个特征可以让模型的效果达到最好呢？标准的方法还是采用交叉检验的方式进行选择。

最后，举一个特征选择应用的例子——文本分类，对于该问题来说，每个样例(文本)有多个特征(词语)。而特征数(词语数目)往往很大，多于样例数目。这时就需要采用特征选择了。