Author：Liu Jian

Time：2020-02-07

机器学习 I -模型选择

1 问题的引出：过拟合、欠拟合与偏差-方差分解

关于数据集的几个概念：

- 测试数据 (testing set)：学得的模型在实际中遇到的数据；
- 验证集 (validation set)：模型选择中用于评估测试的数据集，基于验证集进行模型选择和调参；
- 训练集 (training set)：训练模型的数据集；
- 测试集和训练集有时又统称为训练数据。

事实上，上述几种称呼在实际使用过程中很混乱，实际含义需根据具体语境进行判断。

机器学习希望得到泛化能力强的模型，为了学习潜在的普遍规律，我们使用模型对训练集进行拟合，这样做存在的两个问题：(1) 可能会把训练样本本身的一些特点当做所有潜在样本都会具有的一般性质而学得，也就是**过拟合 (overfitting)**；(2) 学习能力低下，对训练样本的一般性质没有学习好，也就是**欠拟合 (underfitting)**。这两种情况都意味着模型泛化能力的下降。因此，除了要尽量避免欠拟合，也就是尽量比较好地拟合样本数据外，我们还需要防止过拟合。**参数估计偏向于较好地拟合样本数据 (加入正则项的参数估计也在一定程度上考虑了过拟合的问题)，而模型选择除了考虑拟合样本数据外，还考虑了过拟合的问题，其最终的落脚点就是在考察模型的泛化能力。由此，引出了这篇文章的主题--模型选择。**

模型选择是指在不同类型的模型或类型相同但超参数不同的模型中选择泛化能力最好的模型，也就是不仅要考虑对训练数据的拟合程度，还需要考虑模型复杂度、过拟合等方面的问题。模型的**参数估计**则是指给定模型类型与超参数后基于训练数据对模型中待定的参数进行估计，偏向于更好地拟合训练数据 (加入正则项后的参数估计就考虑了过拟合，因此并非完全不考虑过拟合)。此外，所谓的**调参** (parameter tuning) 是指调整模型或者说算法的**超参数**，不要与参数估计弄混了，**调参对应的层面为模型选择。模型选择挑选的是模型的类型及超参数，因此，得到的并不是一个具体的模型，通过参数估计才能将模型中的待定参数具体化。显然，模型选择会涉及参数估计，因为在模型选择的过程中我们会基于训练集训练出具体的模型。**

为了进行模型选择，即评估模型的泛化能力，我们通常以模型在验证集上的的误差作为泛化误差的近似，为此，我们需要解决如下三个问题：(1) 给定样本数据后如何科学地划分验证集和训练集，也就是评估实验的设计，即评估方法；(2) 模型评估时用于衡量模型泛化能力的指标，也就是性能度量 (performance measure)；(3) 我们基于验证集得到的性能度量只是模型泛化能力的近似，那么如何科学地进行模型也就是学习器泛化性能的比较呢，这就涉及到假设检验的内容。

综上，**机器学习的流程**如下：我们首先根据**评估方法**划分数据集 D 为训练集 T 和验证集 V，基于训练集 T 训练模型，基于验证集 V 计算**性能度量**，基于性能度量采用**假设检验**的方法比较模型的泛化性能，进行**模型选择**，确定模型的类型及超参数。**最后，我们将挑选出来的模型在 D 上进行训练，作为最终提交给用户的模型。**

2 性能度量

回归任务中常用的性能度量是均方误差。也就是说，均方误差可用作学习或者说训练时的损失函数，又可用作性能度量时的评估指标，但二者基于的数据集不同：前者基于训练集 T ，后者基于验证集 V 。

接下来讨论的都是**分类任务**的性能度量。

2.1 错误率与精度

错误率，错误分类样本的占比；精度，正确分类样本的占比。显然，错误率 + 精度 = 1。

2.2 代价敏感 (cost-sensitive) 错误率

在上一小节的基础上考虑不同类型的错误所造成的损失不同，即**非均等代价**的情况。**二分类代价矩阵**：$[cost_{ij}]_{2\times 2}$ ，其中， $cost_{ij}$ 表示将第 i 类样本判断为第 j 类样本的代价，显然 $cost_{ii}=0$ 。由此可得代价敏感错误率和代价敏感精度。这里给出代价敏感错误率的计算公式：

$$\varepsilon(f,V,cost)=\frac{1}{N_v}\left(cost_{01}\sum_{x_i\in V^+}\mathbb{I}(f(x_i)\neq y_i)+cost_{10}\sum_{x_i\in V^-}\mathbb{I}(f(x_i)\neq y_i)\right)$$

其中， N_v 为验证集容量，第 0 类样本是指真实标记为正的样本，第 1 类样本是指真实标记为负的样本。**类似可定义多分类任务的代价敏感性能度量。**

2.3 基于 TP、FP、TN、FN 的性能度量

对于二分类问题，可根据分类结果和实际标记将验证集样本分为如下四类：真正例 (true positive, TP)、假正例 (false positive, FP)、真反例 (true negative, TN) 和假反例 (false negative, FN)。易知，验证集样本总数 = TP + FP + TN + FN。

2.3.1 P-R 曲线

查准率/准确率 (precision)： $P=\frac{TP}{TP+FP}$ ，预测为正的样本中真正为正样本的比例。

查全率/召回率 (recall)： $R=\frac{TP}{TP+FN}$ ，真正为正的样本中被识别出来的比例。

P-R 曲线：基于学习器对验证集进行排序，其中被学习器认为更可能是正例的样本点排在前面。按此顺序逐个以某个样本点为界，前面的样本点作为正例，后面的作为反例，每次就可计算出当前的查全率 P 和查准率 R ，以 P 为纵轴， R 为横轴画图，可得 P-R 曲线。**显然，对于某个学习器来说，其 P 值和 R 值是一个此消彼长的关系，P-R 曲线为一条从 $(0,1)$ 到 $(1,0)$ 的曲线。基于 P-R 曲线的模型评估结论如下**：

- 若一个学习器的 P-R 曲线被另一个学习器的曲线完全包住，则可断言后者的性能优于前者；
- 若两个学习器的 P-R 曲线出现交叉的情况：
  - 比较 P-R 曲线下面积的大小，面积越大，性能越好；
  - 基于平衡点 (break-even point, BEP) 进行比较。所谓平衡点就是 $P=R$ 的点，显然，其值越大，学习器的性能越好；
  - 可在某一具体的查准率或查全率条件下进行比较。可比较 查全率 P 和查准率 R 的调和平均即 F_1 度量：

$$\frac{1}{F_1}=\frac{1}{2}\left(\frac{1}{P}+\frac{1}{R}\right)$$

与算术平均和几何平均相比，调和平均更重视较小值。进一步，若考虑对查准率和查全率的重视程度的不同，可比较 F_β 度量：

$$\frac{1}{F_\beta}=\frac{1}{1+\beta^2}\left(\frac{1}{P}+\frac{\beta^2}{R}\right)$$

其中， $\beta>0$ 度量了查全率 R 对查准率 P 的相对重要性， $\beta=1$ 时， F_β 退化为 F_1 。

进一步，还有宏查准率 (macro-P)、宏查全率 (macro-R)、宏 F_1 (macro-F1) 以及微查准率 (micro-P)、微查全率 (micro-R)、微 F_1 (micro-F1) 等指标，这里不再赘述。

2.3.2 ROC 曲线

ROC (receiver operating characteristic) 曲线的绘制和 P-R 曲线类似，只是坐标轴不同：

- 纵坐标为真正率 (ture positive rate, TPR)：$TPR = \dfrac{TP}{TP+FN}$，**实际上就是 P-R 曲线的横坐标召回率** R，也就是验证集中真实标记为正的样本中，被学习器正确识别，也就是判定为正的比例；
- 横坐标为假正率 (false positive rate, FPR)：$FPR = \dfrac{FP}{TN+FP}$，表示验证集中真实标记为负的样本中，被学习器错误识别，也就是判定为正的比例。

ROC 曲线为一条从 $(0,0)$ 到 $(1,1)$ 的曲线。当其为 $(0,0)-(0,1)-(1,1)$ 的折线时，对应于理想模型的排序情况，即真实标记为正的样本排在所有真实标记为负的样本的前面；当其为对角线时，可推得 $\dfrac{TP}{FP} = \dfrac{TP+FN}{TN+FP}$，即 $\text{学习器判断为正中实际也为正的样本数}/\text{学习器判断为正中判断错误的样本数} = \text{真实为正的样本个数}/\text{真实为负的样本个数}$ **，学习器的这一判定结果类似于对验证集进行分层采样，分层采样所得的样本为验证集的子集，且其标记的比例分布与验证集一致，只是我们认为或者说学习器判定分层采样得到的样本点全部为正；学习器的判定过程也就相当于：若已知要从容量为 N_v 的验证集中判定 m 个样本点为正，则学习器会从 N_v 个样本点中以等概率的方式 (每个点被选取的概率为 $\dfrac{m}{N_v}$) 随机选取 m 个样本点作为正的样本点返回，也就是说，此时的 ROC 曲线对应于随机猜测模型的排序情况。**

基于 ROC 曲线的模型评估结论如下：

- 若一个学习器的 ROC 曲线被另一个学习器的曲线完全包住，则可断言后者的性能优于前者；
- 若两个学习器的 P-R 曲线出现交叉，则比较合理的判据是比较 ROC 曲线下的面积，即 AUC (area under ROC curve)：

$$AUC = \frac{1}{2}\sum_{i}^{N_v-1}(x_{i+1}-x_i)(y_i+y_{i+1})$$

其中，N_v 为验证集样本个数。

此外，若定义排序损失 l_{rank} 如下：

$$l_{rank} = \frac{1}{N_v^+ N_v^-}\sum_{x^+\in V^+}\sum_{x^-\in V^-}\left(\mathbb{I}(f(x^+)<f(x^-)) + \frac{1}{2}\mathbb{I}(f(x^+)=f(x^-))\right)$$

即考虑每一对真实标记为正、负的样本，根据学习器对二者进行排序，若正样本排在负样本后面，则记一个罚分；若相等，则记 $\frac{1}{2}$ 个罚分，则：

$$l_{rank} + AUC = 1$$

2.3.3 代价曲线 (cost curve)

考虑 2.2 节中的非均等代价，此时 ROC 曲线失效，可基于代价曲线进行性能度量。

- 横坐标是取值范围为 $[0,1]$ 的**正例概率代价**：将正例判断成反例的**相对**代价，

$$cost^+ = \frac{p\times cost_{01}}{p\times cost_{01} + (1-p)\times cost_{10}}$$

其中，p 表示样本为正的概率。

- 纵坐标是取值范围为 $[0,1]$ 的**归一化代价**：

$$cost_{norm} = FNR\times cost^+ + FPR\times(1-cost^+)$$

其中，FPR 为假正率，$FNR = 1-TPR$ 为假反率。

代价曲线的绘制：将 ROC 曲线上的每个点转化为代价平面上的一条线段，然后取所有线段的下界，围成的面积即为在所有条件下学习器的期望总代价。**显然，期望总代价越小，学习器越好。**

3 评估方法

评估方法就是设计模型评估的实验方案，也就是数据集 D 产生训练集 S 和验证集 T 的方案。

3.1 基于分层采样 (stratified sampling) 的实验方案

- 留出法 (hold-out)

单次留出法：将数据集 D 拆分为两个互斥的集合，其中一个集合作为训练集 S ，另一个集合作为验证集 T 。为了兼顾训练模型的性能和评估结果的稳定性和准确性，一般将 $2/3 \sim 4/5$ 的样本用于训练，剩余样本用于测试。此外，训练集和测试集的划分要尽可能保持数据分布的一致性，也就是它们各自的类别比例与数据集 D 中的类别比例保持一致，从采样的角度看待数据的划分过程即为**分层采样**。

单次留出法得到的估计结果往往不够稳定可靠，因此，一般采用若干次随机划分，重复进行试验评估后取平均值作为留出法的评估结果。

- 交叉验证法 (cross validation)

k **折交叉验证**：将数据集 D 划分为 k 个大小相似的互斥子集 $D_i\ (i=1,\cdots,k)$ ，每个子集 D_i 都尽可能保持数据分布的一致性 (分层采样)，则依次选择某个子集 D_i 为验证集，剩下 $k-1$ 个子集的并集为训练集，评估结果取 k 次实验结果的平均。

k **折交叉验证实际上就是多次留出法的特例。**一般地，取 $k=5/10/20$ ；当 $k=|D|$ 时，为**留一法 (leave-one-out, LOO)。**显然，留一法不受数据集划分方式的影响，但其计算量很大。考虑数据集划分方式的影响，可得 p **次** k **折交叉验证**：随机构造 p 个数据集 D 的 k 折划分，进行 p 次 k 折交叉验证，评估结果为 p 次 k 折交叉验证结果的平均。

3.2 基于自助采样 (bootstrap sampling) 的实验方案

自助法 (bootstrapping)：对数据集 D 进行有放回采样 (也就是自助采样)，得到容量相等的数据集 D' ($|D'|=|D|$)，则训练集 $T=D'$ ，验证集 $V=D\backslash D'$ 为在 D' 中未出现的 D 中的样本，其数量约占 D 的 $1/e$ 。自助法评估得到的结果亦称包外估计 (out-of-bag estimate)。

自助法在数据集较小、难以有效划分训练/验证集时很有用。此外，**自助法能从初始数据集中产生多个不同的训练集，这对集成学习等方法有很大的好处。**然而，自助法产生的数据集改变了初始数据集的分布，这会引入估计偏差，因此，在初始数据量足够时，留出法和交叉验证法更常用一些。

4 假设检验

模型基于验证集的性能度量只是对模型泛化能力的近似，为了科学地评估模型的泛化能力，比较两个模型的泛化能力是否存在显著差异，我们需要借助**假设检验。**

本章取错误率为性能度量，模型真实的错误率 (称为泛化错误率) 记为 ε ，模型基于验证集的错误率 (称为验证错误率) 记为 $\hat{\varepsilon}$ 。接下来，以 $\hat{\varepsilon}$ 为样本 (具有二重性)， ε 为参数，构建概率模型，进行假设检验，其中 α 为显著水平。假设检验的有关内容可参见笔记《4-统计推断》。

4.1 单个学习器泛化性能的假设检验

1. 只有单个样本数据

首先可构建验证错误率 $\hat{\varepsilon}$ 关于泛化错误率 ε 的概率模型：

$$P(\hat{\varepsilon}|\varepsilon)=\binom{N_v}{N_v\hat{\varepsilon}}\varepsilon^{N_v\hat{\varepsilon}}(1-\varepsilon)^{N_v-N_v\hat{\varepsilon}}$$

即以错误分类的样本个数 $N_v\hat{\varepsilon}$ 为随机变量，则其服从参数为 ε 的二项分布。

基于上述概率模型，我们可以对参数 ε 即泛化错误率进行假设检验。比如，提出关于 ε 的原始假设 $H_0:\varepsilon\leqslant\varepsilon_0$ 与备选假设 $H_0:\varepsilon\geqslant\varepsilon_0$ 后，首先，我们在显著水平 α 下由概率模型得到随机变量或者说样本 $\hat{\varepsilon}$ 的接收域： $[0,\varepsilon_1]$ ；接着，我们对 $\hat{\varepsilon}$ 进行一次观察 (即在验证集上运行学习器，求错误率)，得到一个结果 ε_2 ，若 $\varepsilon_2\in[0,\varepsilon_1]$ ，则接受原假设，否则拒绝原假设接受备选假设。

2. 存在多个样本数据

采用 k 折交叉验证等方法时可得 k 个验证错误率的样本，此时可采用 t 检验法 (t-tests) 对泛化错误率 ε 的有关假设进行检验，检验统计量 τ_t ：

$$\tau_t=\frac{\sqrt{k}(\mu-\varepsilon)}{\sigma}\sim t(k-1)$$

$$\text{with}\ \ \mu=\frac{1}{k}\sum_{i=1}^{k}\hat{\varepsilon}_i,\ \ \sigma=\frac{1}{k-1}\sum_{i=1}^{k}(\hat{\varepsilon}_i-\mu)^2$$

4.2 两个学习器泛化性能的假设检验

前面的假设检验只针对一个学习器的泛化性能，而这里介绍的假设检验方法则用于**比较两个学习器 A 和 B 的泛化性能。**

- 成对 t 检验 (paired t-tests)

记学习器 A 和 B 在 k 折交叉验证的第 i 折上错误率分别为 $\hat{\varepsilon}_i^A$ 和 $\hat{\varepsilon}_i^B$ （$i=1,2,\cdots,k$），二者之差 $\delta_i=\hat{\varepsilon}_i^A-\hat{\varepsilon}_i^B$ ，学习器 A 和 B 的泛化错误率之差 $\delta=\varepsilon^A-\varepsilon^B$ 。以 δ_i（$i=1,2,\cdots,k$）为样本，对参数 δ 进行假设检验，检验统计量 τ_t ：

$$\tau_t=\frac{\sqrt{k}(\mu-\delta)}{\sigma}\sim t(k-1)$$

$$\text{with }\ \mu=\frac{1}{k}\sum_{i=1}^{k}\delta_i,\ \ \sigma=\frac{1}{k-1}\sum_{i=1}^{k}(\delta_i-\mu)^2$$

原假设 H_0：学习器A和B的泛化性能相同，即$\delta=0$，若原假设成立，则:

$$\tau_t=\frac{\sqrt{k}\mu}{\sigma}\sim t(k-1)$$

此时， μ 应该在 0 附近，其绝对值不应过大，则接收域可由如下不等式给出：

$$|\tau_t|=\left|\frac{\sqrt{k}\mu}{\sigma}\right|<t_{\alpha/2,k-1}$$

代入样本的值，若上式成立则接受原假设，说明两个学习器的泛化能力没有明显差别；若上式不成立则拒绝原假设，说明两个学习器的泛化能力存在显著差别，此时，再比较两个学习器 k 折交叉验证的平均验证错误率，错误率较小的学习器泛化能力更强。

此外，还有如下两种检验方法，不再赘述：

- 5×2 交叉验证法
- McNemar 检验：针对二分类问题。

4.3 多个学习器泛化性能的假设检验

对于**多个学习器泛化性能的比较**，一种做法是基于前面提到的方法进行两两比较，另一种做法是采用 Friedman 检验对 H_0：所有学习器的泛化性能相同 这个假设进行检验，若原假设被接受则结束判断，否则，需要进行后续检验 (post-hoc test) 对学习器的泛化性能进行进一步地排序，比如基于 Nemenyi 后续检验可得 Friedman 检验图，由此可对各学习器的泛化性能进行排序。

附录 偏差-方差分解

真实模型： $y=f(x)+\epsilon$ ，其中 ϵ 表示噪声，均值为 0 ，方差为 σ^2 (即 $\mathbb{E}[\epsilon]=0,\mathbb{V}[\epsilon]=\sigma^2$)。

算法基于某个数据集 D 学得的模型为 $f_D(x)$ 。

算法的期望泛化误差 (也就是在某个未知点 (x^*,y^*) 处的期望误差，则 $y^*=f(x^*)+\epsilon^*$ ，且这里的期望是针对不同的训练集 D 而言的)：

$$\mathbb{E}_D[(f_D(x^*)-y^*)^2]$$

对上式进行恒等变形：

$$\mathbb{E}_D\Big[(f_D(x^*)-y^*)^2\Big]=\mathbb{E}_D\Big[(f_D(x^*)-f(x^*)-\epsilon^*)^2\Big]$$
$$=\mathbb{E}_D\Big[\Big(f_D(x^*)-\mathbb{E}_D[f_D(x^*)]+\mathbb{E}_D[f_D(x^*)]-f(x^*)-\epsilon^*\Big)^2\Big]$$
$$=\mathbb{E}_D\Big[\Big(f_D(x^*)-\mathbb{E}_D[f_D(x^*)]\Big)^2+\Big(\mathbb{E}_D[f_D(x^*)]-f(x^*)\Big)^2+(-\epsilon^*)^2+$$
$$2\Big(f_D(x^*)-\mathbb{E}_D[f_D(x^*)]\Big)\Big(\mathbb{E}_D[f_D(x^*)]-f(x^*)\Big)+2\Big(f_D(x^*)-\mathbb{E}_D[f_D(x^*)]\Big)\Big(-\epsilon^*\Big)+$$
$$2\Big(\mathbb{E}_D[f_D(x^*)]-f(x^*)\Big)\Big(-\epsilon^*\Big)\Big]$$
$$=\mathbb{E}_D\Big[\Big(f_D(x^*)-\mathbb{E}_D[f_D(x^*)]\Big)^2\Big]+\mathbb{E}_D\Big[\Big(\mathbb{E}_D[f_D(x^*)]-f(x^*)\Big)^2\Big]+\mathbb{E}_D\Big[(\epsilon^*)^2\Big]+0+0+0$$
$$=\mathbb{E}_D\Big[\Big(f_D(x^*)-\mathbb{E}_D[f_D(x^*)]\Big)^2\Big]+\Big(\mathbb{E}_D[f_D(x^*)]-f(x^*)\Big)^2+\mathbb{V}_D\Big[(\epsilon^*)^2\Big]+\Big(\mathbb{E}_D[\epsilon^*]\Big)^2$$
$$=\mathbb{V}_D\Big[f_D(x^*)\Big]+\Big(\mathbb{E}_D[f_D(x^*)]-f(x^*)\Big)^2+\sigma^2+0$$
$$=\text{Variance}+\text{Bias}+\text{Irreducible Error}$$

可见**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所共同决定的**：

- 方差 (Variance)： $\mathbb{V}_D\Big[f_D(x^*)\Big]$，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，即数据扰动所造成的影响，高方差意味着过拟合；
- 偏差 (Bias)： $\Big(\mathbb{E}_D[f_D(x^*)]-f(x^*)\Big)^2$，度量了学习算法的期望预测与真实结果的偏离程度，即刻画了学习算法本身的拟合能力，高偏差意味着欠拟合；
- 不可约错误 (Irreducible Error)： 刻画了学习问题本身的难度。

偏差-方差分解仅在基于均方误差的回归任务中得以推导出，对于分类任务，由于 $0-1$ 损失函数的跳跃性，理论上推导出偏差-方差分解很困难。

The bias–variance dilemma: Models with high variance are usually more complex (e.g. higher-order regression polynomials), enabling them to represent the training set more accurately. In the process, however, they may also represent a large noise component in the training set, making their predictions less accurate – despite their added complexity. In contrast, models with higher bias tend to be relatively simple (low-order or even linear regression polynomials) but may produce lower variance predictions when applied beyond the training set.

奥卡姆剃刀 (Occam's razor) 原则：a compromise between model complexity and goodness of fit (also known as the bias-variance trade-off).
