

# 高级机器学习

## 作业一

丁豪 181220010

2020 年 11 月 8 日

### 学术诚信

本课程非常重视学术诚信规范，助教老师和助教同学将不遗余力地维护作业中的学术诚信规范的建立。希望所有选课学生能够对此予以重视。<sup>1</sup>

- (1) 允许同学之间的相互讨论，但是**署你名字的工作必须由你完成**，不允许直接照搬任何已有的材料，必须独立完成作业的书写过程；
- (2) 在完成作业过程中，对他人工作（出版物、互联网资料）中文本的直接照搬（包括原文的直接复制粘贴及语句的简单修改等）都将视为剽窃，剽窃者成绩将被取消。**对于完成作业中有关键作用的公开资料，应予以明显引用；**
- (3) 如果发现作业之间高度相似将被判定为互相抄袭行为，**抄袭和被抄袭双方的成绩都将被取消**。因此请主动防止自己的作业被他人抄袭。

### 作业提交注意事项

- (1) 请在 LaTeX 模板中**第一页填写个人的姓名、学号、邮箱信息**；
- (2) 本次作业需提交该 pdf 文件、问题 3 可直接运行的源码 (学号 \_\_.py)，将以上两个文件压缩成 zip 文件后上传。zip 文件格式为**学号.zip**，例如 170000001.zip；pdf 文件格式为**学号 \_ 姓名.pdf**，例如 170000001\_ 张三.pdf。
- (3) 未按照要求提交作业，或提交作业格式不正确，将会**被扣除部分作业分数**；
- (4) 本次作业提交截止时间为**11 月 8 日 23:59:59**。除非有特殊情况（如因病缓交），否则截止时间后不接收作业，本次作业记零分。

<sup>1</sup>参考尹一通老师高级算法课程中对学术诚信的说明。

## 1 [30pts] VC Dimensions

本题探讨 VC 维的性质。

(1) [10pts] 请在样本空间  $\mathcal{X} = [0, 1]$  上构造一个有限假设空间  $\mathcal{H}$  使得  $VC(\mathcal{H}) = \lfloor \log_2(|\mathcal{H}|) \rfloor$ .

(2) [10pts] 定义轴平行四边形概念类  $\mathcal{H} = \{h_{(a_1, a_2, b_1, b_2)}(x, y) : a_1 \leq a_2 \wedge b_1 \leq b_2\}$ , 其中

$$h_{(a_1, a_2, b_1, b_2)}(x, y) = \begin{cases} 1 & \text{if } a_1 \leq x \leq a_2 \wedge b_1 \leq y \leq b_2 \\ 0 & \text{otherwise} \end{cases}$$

请证明  $\mathcal{H}$  的 VC 维为 4.

(3) [10 pts] 请证明最近邻分类器的假设空间的 VC 维可以为无穷大.

**Solution.** 此处用于写解答 (中英文均可)

(1)  $\mathcal{H} = \{h_1, h_2, h_3, h_4\}$ , 其中

$$h_1(x) = \begin{cases} 0 & x \geq -1 \\ 1 & x < -1 \end{cases}, h_2(x) = \begin{cases} 1 & x \geq -1 \\ 0 & x < -1 \end{cases}, h_3(x) = \begin{cases} 1 & x \geq 0.5 \\ 0 & x < 0.5 \end{cases}, h_4(x) = \begin{cases} 0 & x \geq 0.5 \\ 1 & x < 0.5 \end{cases}$$

则存在  $D = \{x_1 = 0.2, x_2 = 0.7\}$ , 则其可以被  $\mathcal{H}$  打散。对任意大小为 3 的示例集  $\{x_3, x_4, x_5\}$ , 不妨设  $x_3 < x_4 < x_5$ , 则  $\mathcal{H}$  中不存在任何假设能实现对分结果  $(1, 0, 1)$ 。因此  $VC(\mathcal{H}) = 2 = \lfloor \log_2(|\mathcal{H}|) \rfloor$

(2) 存在  $D = \{(0, -1), (-1, 0), (0, 1), (1, 0)\}$ , 则假设空间中存在  $\{h_{(0,0,0,0)}, h_{(-1,-1,0,0)}, h_{(1,1,0,0)}, h_{(0,0,1,1)}, h_{(0,0,-1,-1)}, h_{(-1,0,0,1)}, h_{(0,1,0,1)}, h_{(-1,0,-1,0)}, h_{(0,1,-1,0)}, h_{(-1,1,0,0)}, h_{(0,0,-1,1)}, h_{(-1,0,-1,1)}, h_{(-1,1,0,1)}, h_{(0,1,-1,1)}, h_{(-1,1,-1,0)}, h_{(1,1,1,1)}\}$  可以将  $D$  对分, 因此  $VC(\mathcal{H}) \geq 4$ 。与此同时, 对任意大小为 5 的示例集  $\{(x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4), (x_5, y_5)\}$ , 不妨设  $x_1 \leq x_2 \leq x_3 \leq x_4 \leq x_5$ 。若  $\mathcal{H}$  可以打散  $D$ , 则对任意  $(x_i, x_j), (x_x, x_y, x_z)$  的组合, 存在某  $h \in \mathcal{H}$  使得  $h(x_i) = h(x_j) = 1 \wedge h(x_x) = h(x_y) = h(x_z) = 0$ , 即可以用边平行于  $X$ 、 $Y$  轴的矩形将坐标平面上的 5 个点每两个与其余分隔开。由图论知识可以证明, 这是不可能的, 因而不存在这样的  $D$ 。所以  $\mathcal{H}$  的 VC 维为 4。

(3) 对任意  $m$ , 设某示例集  $D = \{x_1, x_2 \dots x_m\}$ , 考虑  $\mathcal{H}$  中以  $\{(x_1, y_1), (x_2, y_2) \dots (x_m, y_m)\}$  为邻居点产生的分类器  $\{h\}$ ,  $h|_D = \{y_1, y_2 \dots y_m\}$ , 则共有  $2^m$  种可能结果 (对  $y_i$  二进制赋值得到), 即  $\mathcal{H}$  可以将  $D$  打散, 所以  $VC(\mathcal{H}) \geq m$ 。由于对任意  $m$  都成立, 因此最近邻分类器的假设空间 VC 维可以为正无穷。

## 2 [40pts] Understanding the Parameters in LVW from A Probabilistic Perspective

课程中, 我们介绍了一种包裹式特征选择算法 Las Vegas Wrapper (简称 LVW), 该算法的流程如教材中图 11.1 所示。首先请各位回顾一下 LVW 算法, 接下来我们将对一种特殊情况

$p(T = 1)$	$p(T = 2)$	$p(T = 3)$

下的 LVW 做一些分析，过程中复习一些简单的概率知识，并从更理性的角度理解 LVW 中的参数。

现在，我们获得了一个数据集  $D$ ，该数据集中一共包含  $N$  个特征，设特征集合为  $A = \{f_1, f_2, \dots, f_N\}$ 。我们知道，对于某个具体任务，特征的质量参差不齐，高质量的特征会带来高质量的性能，低质量的特征会带来低质量的性能。我们设第  $i$  个特征的质量为  $2^{i-1}$ ，即  $N$  个特征的质量分别为  $1, 2, 4, \dots, 2^{N-1}$ 。我们规定：给定一个特征子集  $A'$ ，学习器在  $A'$  上的性能恰好等于  $A'$  中包含的所有特征的质量之和。设特征子集  $A'$  中特征的质量之和记作  $Q(A')$ 。

- (1) [5pts] 现执行一次图 11.1 中第 6 行的语句，产生一个特征子集  $A'$ ，试求数学期望  $\mathbb{E}[Q(A')]$ 。
- (2) [10pts] 现在，LVW 算法已经执行了一段时间了，当前得到的最优特征子集为  $B$ ，我们记  $R = Q(B)$ 。设函数  $\text{better}(n, r)$  表示从前  $n$  个特征中随机产生一个特征子集且该特征子集的质量大于  $r$  的概率。试求  $\text{better}(N, R)$ 。这里我们允许以递归式和递归边界来表达  $\text{better}(N, R)$ 。
- (3) [10pts] 从现在开始，我们设  $\text{better}(n, r)$  是一个已知函数。在 LVW 算法中，当连续  $T$  个随机生成的子集不比当前的最优子集更好时，算法结束。我们仍然设当前得到的最优特征子集为  $B$ ， $B$  的质量为  $R = Q(B)$ 。在本小问中，我们还设  $T$  是一个已知参数。设布尔型随机变量  $e$ ， $p(e = 1|T)$  表示经过恰好  $T$  次循环后 LVW 算法结束的概率， $p(e = 0|T)$  表示经过恰好  $T$  次循环后 LVW 算法没有结束的概率。试求分布  $p(e|T)$ 。
- (4) [5pts] 由第 (3) 小问可见，参数  $T$  越大，LVW 算法结束就越 ([回答“容易”或“困难”])；当前最优子集质量越高，LVW 算法结束就越 ([回答“容易”或“困难”])。
- (5) [10pts] 现在，我们引入 Bayes 观点，认为参数  $T$  是服从某个先验分布  $p(T)$  的随机变量，且是未知的。我们仍然设当前得到的最优子集为  $B$ ， $B$  的质量为  $R = Q(B)$ 。现在，LVW 算法继续执行，在恰好执行了  $T$  个循环后成功退出了。如果我们设  $T$  的先验分布为整数区间  $[1, 10]$  上的均匀分布，请对  $T$  做最大后验估计。这与最大似然估计的结果相同吗？为什么？
- (6) [Bonus 5pts] 现在，设随机变量  $T$  只能在  $\{1, 2, 3\}$  中取值，且小明设置了一个先验  $p(T)$ 。小红观察到，当前得到的最优子集为  $B$ ， $B$  的质量为  $R = Q(B)$ ，且 LVW 在执行恰好  $T$  个循环后成功退出了，于是小红试图对  $T$  做最大后验估计。小红发现， $T$  的后验概率在  $\{1, 2, 3\}$  上是均匀的，那么小明设置的先验  $p(T)$  有可能为：(填写一种可能的答案即可)

**Solution.** 此处用于写解答 (中英文均可)

(1) 每一个特征以  $\frac{2^{N-1}}{2^N - 1}$  的概率被选取 (分母减去 1 是去掉了空集的情况)，则

$$\mathbb{E}[Q(A')] = \sum_{i=1}^N \frac{2^{N-1}}{2^N - 1} \cdot 2^{i-1} = \frac{2^{N-1}}{2^N - 1} \cdot \frac{1 - 2^N}{1 - 2} = 2^{N-1}$$

(2)  $\text{better}(N, R)$  通过以下式子中  $n = N, r = R = Q(B)$  解出。

$$\text{better}(n, r) = \begin{cases} \frac{2^{N-1}-1}{2^{N-1}} \cdot \text{better}(n-1, r) + \frac{2^{N-1}}{2^{N-1}} \cdot \text{better}(n-1, r-2^{n-1}) & , n \geq 2 \\ \frac{2^{N-1}}{2^{N-1}} & , n = 1 \wedge r < 1 \\ 0 & , n = 1 \wedge r \geq 1 \end{cases}$$

通过将题目设定的  $R_i = 2^{i-1}$  代入, 可以将递归式化为一般表达式  $\text{better}(N, R) = \frac{2^N-1-R}{2^{N-1}}$

$$\begin{aligned} (3) \quad p(e=1|T) &= [1 - \text{better}(N, R)]^T \\ p(e=0) &= 1 - [1 - \text{better}(N, R)]^T \end{aligned}$$

(4) 困难, 容易

(5) 为了以示区别, 设恰好执行了  $T = T_0$  个循环后成功退出。

$$\text{最大后验估计: } \hat{T}_{MAP}(T_0) = \arg \max_{T \in [1, 10], T \in \mathbb{N}} P(T|T_0) = \arg \max_{T \in [1, 10], T \in \mathbb{N}} P(T_0|T)P(T)$$

$$\text{由于是均匀先验, } P(T) \text{ 为定值, 因此 } \hat{T}_{MAP}(T_0) = \arg \max_{T \in [1, 10], T \in \mathbb{N}} [1 - \text{better}(N, R)]^T$$

$$\text{极大似然估计: } \hat{T}_{MLE}(T_0) = \arg \max_{T \in \mathbb{N}^+} P(T_0|T) = \arg \max_{T \in \mathbb{N}^+} [1 - \text{better}(N, R)]^T$$

如果认为最大似然估计的  $T$  也需要符合  $[1, 10]$  约束, 则表达式完全相同, 故结果相同。若不然, 则二者不同, 因为最大后验具有  $T \in [1, 10] \wedge T \in \mathbb{N}$  限制, 因此不可能求出超过 10 的  $T$ 。而极大似然则只有  $T \in \mathbb{N}^+$  限制, 可以求出超过 10 的  $T$ 。

(6) 由极大后验估计公式:

$$[1 - \text{better}(N, R)]P(T=1) = [1 - \text{better}(N, R)]^2 P(T=2)$$

$$= [1 - \text{better}(N, R)]^3 P(T=3)$$

$$\therefore P(T=1) = [1 - \text{better}(N, R)]P(T=2) = [1 - \text{better}(N, R)]^2 P(T=3)$$

$$\therefore P(T=1) = \frac{R}{2^{N-1}} P(T=2) = \frac{R^2}{(2^{N-1})^2} P(T=3) \text{ 且 } P(T=1) + P(T=2) + P(T=3) = 1$$

$$\text{因此一种可能的先验概率为 } P(T=1) = \frac{\frac{R^2}{(2^{N-1})^2}}{\frac{R^2}{(2^{N-1})^2} + \frac{R}{2^{N-1}} + 1}, P(T=2) = \frac{\frac{R}{2^{N-1}} + 1}{\frac{R^2}{(2^{N-1})^2} + \frac{R}{2^{N-1}} + 1}, P(T=3) = \frac{1}{\frac{R^2}{(2^{N-1})^2} + \frac{R}{2^{N-1}} + 1}$$

### 3 [30pts] Semi-supervised SVM in practice

参照教材中图 13.4 所示的 TSVM 算法, 在所提供的半监督数据集上进行训练, 报告模型在未标记数据集以及测试集上的性能。

本次实验的数据集为一个二分类的数据集, 已提前划分为训练数据和测试数据, 其中训练数据划分为有标记数据和无标记数据。数据的特征维度为 30, 每一维均为数值类型。数据文件的具体描述如下:

- `label_X.csv, label_y.csv` 分别是有标记数据的特征及其标签。
- `unlabel_X.csv, unlabel_y.csv` 分别是无标记数据的特征及其标签。
- `test_X.csv, test_y.csv` 分别是测试数据的特征及其标签。

注意，训练阶段只可以使用 `label_X.csv`, `label_y.csv`, `unlabel_X.csv` 中的数据，其他的数据只可以在测试阶段使用。

(1) 本次实验要求使用 Python3 编写，代码统一集中在 `tsvm_main.py` 中，通过运行该文件就可以完成训练和测试，并输出测试结果。

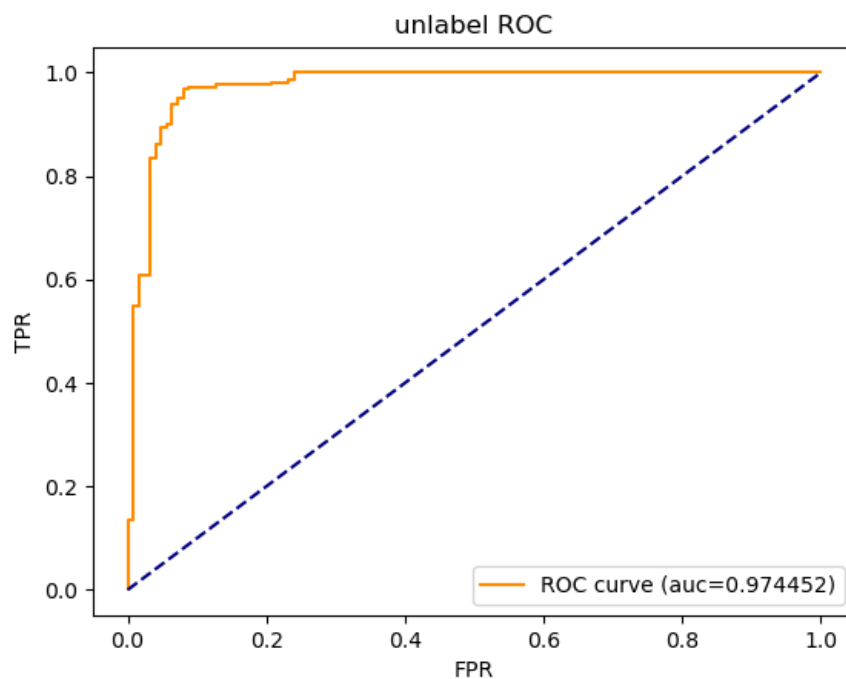
(2) 本次实验需要完成以下功能:

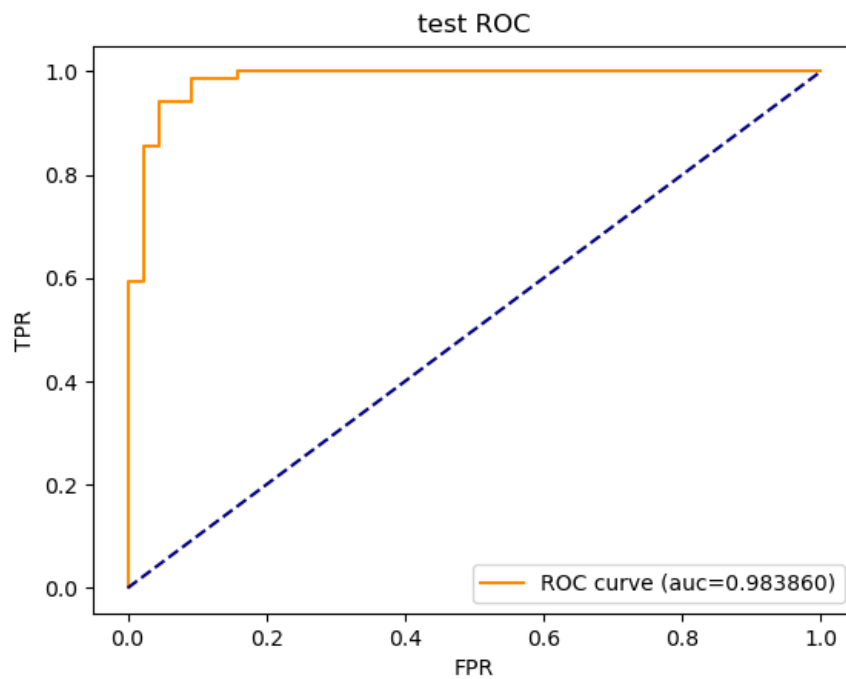
- [10pts] 参照教材中图 13.4，使用代码实现 TSVM 算法。要求：
  1. 不允许直接调用相关软件包中的半监督学习方法。
  2. 可以直接调用相关软件包的 SVM 算法。
  3. 可以使用诸如 `cvxpy` 等软件包求解 QP 问题。
- [10pts] 使用训练好的模型在无标记数据和测试数据上进行预测，报告模型在这两批数据上的准确率和 ROC 曲线以及 AUC 值。
- [10pts] 尝试使用各种方法提升模型在测试集上的性能，例如数据预处理，超参数调节等。报告你所采取的措施，以及其所带来的提升。

**Solution.** 此处用于写解答 (中英文均可)

1. 如 `181220010_.py` 所示，完成了本部分内容。使用了 `sklearn` 的 `svm.SVC` 包进行软间隔支持向量机求解。同时使用 `skleran.matrix` 中的 `accuracy_score`, `roc_curve`, `auc` 函数用于计算评价指标。

2. 此版本，设置  $C_l, C_u$  初始值为 1, 0.000001。得到结果为，  
在 `unlabel` 数据集上  $acc = 0.9385964912280702$ ,  $auc = 0.9744516129032258$   
在 `test` 数据集上  $acc = 0.9557522123893806$ ,  $auc = 0.9838603425559947$





### 3. 逐步改进过程如下

- 设置不同的  $C_l, C_u$  初始值，调整得到最优情况为 1, 0.000001，结果在上一小问已经呈现。
- 对数据特征进行标准化，即均值归 0，标准差归 1。  
得到 *unlabel* 上  $acc = 0.9649122807017544$ ,  $auc = 0.9902672811059908$   
*test* 上  $acc = 0.9646017699115044$ ,  $auc = 0.9976943346508564$