机器学习导论
作业二

171840708, 张逸凯, zykhelloha@gmail.com

2020 年 4 月 2 日

1 [15 pts] Linear Regression

给定数据集$D = \{(\mathbf{x}_i, y_i)\}_{i=1}^m$，最小二乘法试图学得一个线性函数$y = \mathbf{w}^*\mathbf{x} + b^*$使得残差的平方和最小化，即

$$(\mathbf{w}^*, b^*) = \arg\min_{\mathbf{w},b} \sum_{i=1}^{m} [y_i - (\mathbf{w}\mathbf{x_i} + b)]^2. \tag{1.1}$$

“最小化残差的平方和”与“最小化数据集到线性模型的欧氏距离之和”或是“最小化数据集到线性模型的欧氏距离的平方和”一致吗？考虑下述例子

$$D = \{(-1, 0), (0, 0), (1, 1)\}, \tag{1.2}$$

并回答下列问题。

(1) **[5 pts]** 给出“最小化残差的平方和”在该例子中的解(w^*, b^*)。

(2) **[5 pts]** 给出“最小化数据集到线性模型的欧氏距离的平方和”在该例子中的数学表达式，并给出其解(w_E, b_E)，该解与(w^*, b^*)一致吗？

(3) **[5 pts]** 给出“最小化数据集到线性模型的欧氏距离之和”在该例子中的数学表达式，(w^*, b^*)是该问题的解吗？

Solution. ***(a)***

在残差的平方和里分别对$\boldsymbol{w}, b$进行求偏导并令其等于*0.*

得*:*

$$w^\star = \frac{\sum_{i=1}^m y_i \left(x_i - \bar{x}\right)}{\sum_{i=1}^m x_i^2 - \frac{1}{m}\left(\sum_{i=1}^m x_i\right)^2} = \frac{1}{2}$$

$$b^\star = \frac{1}{m}\sum_{i=1}^m \left(y_i - w x_i\right) = \frac{1}{3}$$

其中$\bar{x}$是x的均值*.*

(b)

最小化数据集到线性模型的欧氏距离之和等价于:

$$(w_E, b_E) = \arg\min_{\mathbf{w},b} \sum_{i=1}^{m} \frac{[y_i - (\mathbf{w}\mathbf{x_i} + b)]^2}{\mathbf{w^2} + 1}$$

对$\boldsymbol{w}, b$进行求偏导并令其等于0, 代入D, $\Rightarrow$

$$\sum_{i=1}^{m} \frac{2(y_i-(wx_i+b))\cdot(-x_i)(w^2+1) - (y_i-(wx_i+b))^2\cdot 2w}{(w^2+1)^2}$$
$$= \sum_{i=1}^{m} (y_i-(wx_i+b)) \cdot \frac{-2x_i(w^2+1)-(y_i-(wx_i+b))\cdot 2w}{(w^2+1)^2} = 0$$
$$\Rightarrow \sum_{i=1}^{m} (y_i-(wx_i+b)) \cdot (x_i + wy_i - wb) = 0$$

又 $b = \frac{1}{m}\sum_{i=1}^{m}(y_i - wx_i)$ 代入上式

$$\Rightarrow \sum_{i=1}^{m} \left(y_i - wx_i - \frac{1}{m}\sum_{i}^{m}(y_i - wx_i)\right)\left(x_i + wy_i - \frac{w}{m}\sum_{i}^{m}(y_i - wx_i)\right) = 0$$

$D = \{(-1,0), (0,0), (1,1)\} \Rightarrow \sum_{i}^{m}(y_i - wx_i) = 1$, $b = \frac{1}{3}$

$$\Rightarrow \left(w - \frac{1}{3}\right)\left(-1 - \frac{w}{3}\right) + \left(\frac{2}{3} - w\right)\left(1 + \frac{2}{3}w\right) + \frac{w}{9} = 0, \quad -w^2 - \frac{4}{3}w + 1 = 0$$

$$\Rightarrow w = -\frac{2}{3} \pm \frac{\sqrt{13}}{3}$$

由D数据集的分布知舍去负根

$\therefore w = \frac{\sqrt{13}}{3} - \frac{2}{3}$, $b = \frac{1}{3}$.

$\therefore$

$$w^\star = \frac{\sqrt{13} - 2}{3}$$
$$b^\star = \frac{1}{3}$$

与(w^*, b^*)不一致.

(c)

最小化数据集到线性模型的欧氏距离之和等价于:

$$(w_c, b_c) = \arg\min_{\mathbf{w},b} \sum_{i=1}^{m} \frac{|y_i - (\mathbf{w}\mathbf{x_i} + b)|}{\sqrt{\mathbf{w^2} + 1}}$$

$(w^\star, b^\star) = (\frac{1}{2}, \frac{1}{2})$时, 损失函数可以达到更小. 所以$(w^\star, b^\star)$不是该问题的解.

2 [40+5 pts] 编程题, Logistic Regression

请结合编程题指南进行理解

试考虑对率回归与线性回归的关系。最简单的对率回归的所要学习的任务仅是根据训练数据学得一个$\beta = (\omega; b)$，而学习β的方式将有下述两种不同的实现：

0. [闭式解] 直接将分类标记作为回归目标做线性回归，其闭式解为

$$\beta = (\hat{X}^T\hat{X})^{-1}\hat{X}^T y \tag{2.1}$$

, 其中$\hat{X} = (X; \vec{1})$

1. [数值方法] 利用牛顿法或梯度下降法解数值问题

$$\min_{\beta} \sum_{i=1}^{m} (-y_i \beta^T \hat{x}_i + \ln(1 + e^{\beta^T \hat{x}_i})). \tag{2.2}$$

得到β后两个算法的决策过程是一致的，即：

(1) $z = \beta X_i$

(2) $f = \frac{1}{1+e^{-z}}$

(3) 决策函数

$$y_i = \begin{cases} 1, & \text{if } f > \theta \\ 0, & \text{else} \end{cases} \tag{2.3}$$

其中θ为分类阈值。回答下列问题：

(1) **[10 pts]** 试实现用闭式解方法训练分类器。若设分类阈值$\theta = 0.5$，此分类器在Validation sets下的准确率、查准率、查全率是多少？

(2) **[10 pts]** 利用所学知识选择合适的分类阈值，并输出闭式解方法训练所得分类器在test sets下的预测结果。

(3) **[10 pts]** 利用数值方法重新训练一个新的分类器。若设分类阈值$\theta = 0.5$，此分类器在Validation sets下的准确率、查准率、查全率是多少？

(4) **[10 pts]** 利用所学知识选择合适的分类阈值，并输出数值方法训练所得分类器在test sets下的预测结果。

(5) **[选做][Extra 5 pts]** 谈谈两种方法下分类阈值的变化对预测结果的影响，简要说明看法。

Solution. *(1) 准确率 0.74, 查准率: P = 0.67, 查全率: R = 1.*

(2) 请见附件结果文件.

(3) 梯度下降法, 在足够多的迭代次数之后(这里进行了50次迭代, 学习率为0.01, 初始化参数为0), 得到准确率 1, 查准率: P = 1, 查全率: R = 1.

(4) 请见附件结果文件.

$$\ell(\beta) = \frac{1}{N}\sum_{i=1}^{N}\left[-y_i \ln \frac{1}{1+e^{\beta x_i}} - (1-y_i)\ln\left(1-\frac{1}{1+e^{\beta x_i}}\right)\right]$$

$$\frac{\partial \ell(\beta)}{\partial \beta_j} = \frac{1}{N}\sum_{i=1}^{N}\left[-y_i \frac{1}{S(x_i)} + (1-y_i)\frac{1}{1-S(x_i)}\right] S(x_i)\left(1-S(x_i)\right)x_{ij}$$

$$= \frac{1}{N}\sum_{i=1}^{N}\left[-y_i\left(1-S(x_i)\right) + S(x_i)(1-y_i)\right]x_{ij}$$

$$= \frac{1}{N}\sum_{i=1}^{N}\left[-y_i + S(x_i)\right]x_{ij}$$

$$= \frac{1}{N}X^{T}\left[S(X)-Y\right]$$

图 1: 梯度下降推导过程

(5) 附加:

在验证集上我们有:

- 闭式解: 我们观察闭式解的结果可以发现,

  预测值大于*0.71921136*的数据: 真实值为正.

  预测值小于*0.51404481*的数据: 真实值为负.

  阈值$\in [0.51404481, 0.71921136]$的准确率都是*1*.

- 梯度下降解: 同理观察数据可以发现(**迭代次数只有5次**)

  预测值大于*0.86078419*的数据: 真实值为正.

  预测值小于*0.03942457*的数据: 真实值为负.

  阈值$\in [0.03942457, 0.86078419]$的准确率都是*1*.

  如果迭代次数加高, 正负类会被分得更开.

  结合*sigmoid*(z)函数的图像:

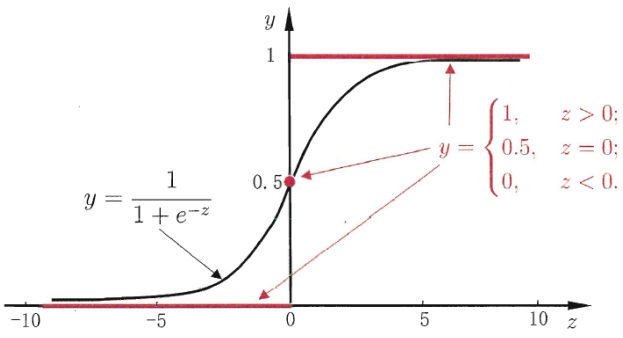

图 3.2 单位阶跃函数与对数几率函数

也就是说闭式解经过损失函数(*sigmoid*的作用于正确的类别预测)6求解的线性回归值z, 正负之间的*gap*没有那么开, 所以阈值的取值范围就更小(在完全把真实值为正负分开的前提下).

同理梯度下降对验证集的输出z将真实值为正负的分得较开, 准确率为*1*下阈值的取值范围更大.

3 [10 pts] Linear Discriminant Analysis

在凸优化中，试考虑两个优化问题，如果第一个优化问题的解可以直接构造出第二个优化问题的解，第二个优化问题的解也可以直接构造出第一个优化问题的解，则我们称两个优化问题是等价的。基于此定义，试证明优化问题**P1**与优化问题**P2**是等价的。

$$\max_{\mathbf{w}} \frac{\mathbf{w}^\top S_b \mathbf{w}}{\mathbf{w}^\top S_w \mathbf{w}}. \tag{3.1}$$

$$\begin{aligned} \min_{\mathbf{w}} \quad & -\mathbf{w}^\top S_b \mathbf{w} \\ \text{s.t.} \quad & \mathbf{w}^\top S_w \mathbf{w} = 1. \end{aligned} \tag{3.2}$$

Solution. • 令*P1, P2*的最优解为w_1, w_2.

(1) 从***P1***的解构造出***P2***的解*:*

不妨令

$$\mathbf{w_1} = argmax_{\mathbf{w}} \frac{\mathbf{w}^\top S_b \mathbf{w}}{\mathbf{w}^\top S_w \mathbf{w}}, \text{并且取最大值时}: t_w = \mathbf{w_1}^\top S_w \mathbf{w_1}.$$

则取$\mathbf{w_2} = \frac{1}{\sqrt{t_w}} \mathbf{w_1}$即可, 证明如下*:*

此时有$\mathbf{w_2}^\top S_w \mathbf{w_2} = 1$. 且

$$\frac{\mathbf{w_2}^\top S_b \mathbf{w_2}}{\mathbf{w_2}^\top S_w \mathbf{w_2}} = \frac{\mathbf{w_1}^\top S_b \mathbf{w_1}}{\mathbf{w_1}^\top S_w \mathbf{w_1}} = P1\text{的解}$$

反证法: 若此时$\mathbf{w_2}^\top S_b \mathbf{w_2}$不是最大值(即*P2*原问题中的取负为最小值), 则存在$R > \mathbf{w_2}^\top S_b \mathbf{w_2} = \frac{\mathbf{w_2}^\top S_b \mathbf{w_2}}{\mathbf{w_2}^\top S_w \mathbf{w_2}} = P1$的解, 这与*P1*是最优解不符, 所以$w_2$就是*P2*的解.

(2) 从***P2***的解构造出***P1***的解*:*

$\mathbf{w_1} = \mathbf{w_2}$即可.

反证法: 假设存在$\mathbf{w}^*$使*P1*更优, 令$t_w = \mathbf{w_*}^\top S_w \mathbf{w_*}$, $w_1^* = \frac{1}{\sqrt{t_w}} \mathbf{w_1}$, 则$\mathbf{w_1^*}^\top S_w \mathbf{w_1^*} = 1$, 且$\mathbf{w_1^*}^\top S_b \mathbf{w_1^*}$在*P2*中比$w_2$更优, 这与$w_2$是*P2*的最优解矛盾.

4 [35 pts] Multiclass Learning

在处理多分类学习问题的时候，我们通常有两种处理思路：一是间接求解，利用一些基本策略(OvO, OvR, MvM)将多分类问题转换为二分类问题，进而利用二分类学习器进行求解。二是直接求解，将二分类学习器推广到多分类学习器。

4.1 问题转换

(1) **[5 pts]** 考虑如下多分类学习问题：假设样本数量为n，类别数量为C，二分类器对于大小为m的数据训练的时间复杂度为$\mathcal{O}(m)$(比如利用最小二乘求解的线性模型)时，试分别计算在OvO、OvR策略下训练的总时间复杂度。

(2) **[10 pts]** 当我们使用MvM处理多分类问题时，正、反类的构造必须有特殊的设计，一种最常用的技术为“纠错输出码”(ECOC)，根据阅读材料(Error-Correcting Output Codes、Solving Multiclass Learning Problems via Error-Correcting Output Codes **[?]**；前者为简明版，后者为完整版)回答下列问题：

1) 假设纠错码之间的最小海明距离为n，请问该纠错码至少可以纠正几个分类器的错误？对于图2所示的编码，请计算该纠错码的最小海明距离并分析当两个分类器出错时该编码的纠错情况。

Class	Code Word							
	f_0	f_1	f_2	f_3	f_4	f_5	f_6	f_7
c_0	0	0	0	0	1	1	1	1
c_1	0	0	1	1	0	0	1	1
c_2	0	1	0	1	0	1	0	1

图 2: 3类8位编码

2) 令码长为8，类别数为4，试给出海明距离意义下的最优ECOC编码，并简述构造思路。

3) 试简述好的纠错码应该满足什么条件？(请参考完整版阅读资料)

4) ECOC编码能起到理想纠错作用的重要条件是：在每一位编码上出错的概率相当且独立，试分析多分类任务经ECOC编码后产生的二类分类器满足该条件的可能性及由此产生的影响。

(3) **[10 pts]** 使用OvR和MvM将多分类任务分解为二分类任务求解时，试论述为何无需专门这对类别不平衡进行处理。

4.2 模型推广

[10 pts] 对数几率回归是一种简单的求解二分类问题的广义线性模型，试将其推广到多分类问题上，其中标记为$y \in \{1, 2 \ldots, K\}$。

提示：考虑如下$K-1$个对数几率

$$\ln \frac{p(y=1|\mathbf{x})}{p(y=K|\mathbf{x})}, \ln \frac{p(y=2|\mathbf{x})}{p(y=K|\mathbf{x})}, \cdots, \ln \frac{p(y=K-1|\mathbf{x})}{p(y=K|\mathbf{x})}$$

Solution. *4.1 (1)*

OvO:

考虑一般情况(即使类别不均衡), 因为训练$\frac{C(C-1)}{2}$个分类器, 相当于对整个数据集遍历了$C-1$遍. 每个类别里的数据都要和其他$C-1$个配对, 训练一个二分类器, 所以共$(C-1)n$, 举个例子$C=4$: $D=\{A,B,C,D\}$, 其中A是第一个类别的所有数据,

那么6个二分类器要训练$AB;AC;AD;BC;BD;CD$这些数据, 也就是$C-1=3$个D. 所以复杂度**即:**

$$(C-1)\mathcal{O}(n)$$

.

OvR:

训练C个对于数据集(大小n)的二分类器, **即:**

$$C\,\mathcal{O}(n)$$

.

4.1 (2)

1) 至少可以纠正$\lfloor\frac{n-1}{2}\rfloor$个分类器的错误.

该纠错码最小海明距离是4, 不妨设测试数据的类别为c_0, 但是预测时某个分类器出错导致编码为`0 1 0 0 1 1 1 1`, 则它与c_0的海明距离为1, c_1为5, c_2为3, 所以这个编码仍然能分类到正确的类c_0.

2) 这里新构建一列, 用到了下一题的条件列分离, 要尽量和其他列以及其他列的反码的海明距离大.

由于$k=4$, 使用*Exhaustive Code*方法:

$class$	1	2	3	4	5	6	7	8
C_1	1	1	1	1	1	1	1	1
C_2	0	0	0	0	1	1	1	1
C_3	0	0	1	1	0	0	1	1
C_4	0	1	0	1	0	1	0	1

构造思路: 其中*Exhaustive Code*方法可以构造出前7个编码, 因为反码和本身使这个分类器是相关度极高的(由**下一题的原则2列分离**), 所以用前7个码位后还剩下$2^4-7\times 2=2$种四位编码: **全1和全0**, 这里我们选择全1全0作为添加的一列都是效果一样的. 因为前面14个相关度高的编码, 对于添加的全1或者全0的总海明距离是一致的.

3) **优化原则(条件):**

1. 行分离: 每个编码位应与其他充分隔开.

2. 列分离: 任意两个分类器的输出概率上应不相关, 独立无关联. 第i列的分类器**与其他分类器(每一列) 以及每一列的反码**之间的汉明距离也要尽可能大.

4) a. 每一位编码上出错的概率相当: 即每个分类器泛化误差相同. 书上说将多个类拆解为两个"类别子集", 所形成的两个类别子集的区分难度往往不同, 即其导致的二分类问题的难度不同. 比如有类别不平衡的问题. **综上,** 实际中实现这个还是有难度的.

b. 在每一位编码上出错的概率相互独立, 这在上一小题中提到过即不同分类器对应的输出编码距离越大越好, 比如k分类问题就可以有2^k个编码, 所以分的类别较多时实现的可能性较大*(*因为编码较多, 选择距离更大的可能就更多*)*.

(3) 对*OvR*、*MvM*来说, 由于对每个类进行了相同的处理, 其拆解出的二分类任务中类别不平衡的影响会相互抵消, 因此通常不需专门处理.

不失一般性, 不妨设C_i, C_j两类不平衡.

- *OvR:* 在所有划分中数据集D都被分为"*One*"类和"*Rest*"类, 对于$\frac{N(N-1)}{2}$个分类器, C_i, C_j都在这些"*One*"类中出现了一次, C_i, C_j都在这些"*Rest*"类中出现了$\frac{N(N-1)}{2}-1$次, 也就是说这$\frac{N(N-1)}{2}$个分类器中两个类别在"*One*"和"*Rest*"中出现是均等的. 所以相互抵消, 可以解决类不平衡问题.

- *MvM:* 不妨设此时的*MvM*是一个较优的编码中*(Exhaustive Code*算法生成的*)*, 不妨以上题中构造$C_1 \dots C_4$为例, 对于C_2, C_4两类, 他们在不同的分类器中为正类的出现次数一致, 并且可以看到$2^4/2 = 8$*(*除*2*因为反码分类器与原先分类器一致*)*已经使用了*8*种分类器, 所以分类器的覆盖是均匀的, 也就是考虑*MvM*整个方法, 样本作为稀疏类和稠密类被采样的概率都是一样的. 所以可以解决不平衡问题.

综上所述, 用这两种方法时无需专门对类别不平衡进行处理.

4.2

将$w^T x + b$简写为$\beta^T \hat{x}$. 其中β最后一列为b, $\hat{x}$最后一列全*1*.

由*hint*可以得到:

$$\ln \frac{\Pr(y_i = 1 \mid x_i)}{\Pr(y_i = K \mid x_i)} = \beta_1 \cdot x_i$$
$$\ln \frac{\Pr(y_i = 2 \mid x_i)}{\Pr(y_i = K \mid x_i)} = \beta_2 \cdot x_i$$
$$\cdots\cdots$$
$$\ln \frac{\Pr(y_i = K-1 \mid x_i)}{\Pr(y_i = K \mid x_i)} = \beta_{K-1} \cdot x_i$$

所以有:

$$\Pr(y_i = K \mid x_i) = 1 - \sum_{j=1}^{K-1} \Pr(y_i = j \mid x_i) = 1 - \sum_{j=1}^{K-1} \Pr(y_i = K \mid x_i)\, e^{\beta_j x_i}$$
$$\Rightarrow \Pr(y_i = K \mid x_i) = \frac{1}{1 + \sum_{j=1}^{K-1} e^{\beta_j \cdot x_i}}$$

同理可得:

$$\Pr(y_i = 1 \mid x_i) = \frac{e^{\beta_1 \cdot x_i}}{1 + \sum_{j=1}^{K-1} e^{\beta_j x_i}}$$

$$\Pr(y_i = 2 \mid x_i) = \frac{e^{\beta_2 \cdot x_i}}{1 + \sum_{j=1}^{K-1} e^{\beta_j x_i}}$$

$$\cdots\cdots$$

$$\Pr(y_i = K-1 \mid x_i) = \frac{e^{\beta_{K-1} x_i}}{1 + \sum_{j=1}^{K-1} e^{\beta_j x_i}}$$

对数似然函数为:

$$\ell(\boldsymbol{w}, b) = \sum_{i=1}^{m} \ln p\left(y_i | \boldsymbol{x}_i; \boldsymbol{w}, b\right)$$

其中似然项为:

$$p\left(y_i | \boldsymbol{x}_i; \boldsymbol{w}, b\right) = \Pr(y_i \mid x_i) = \mathbb{I}(y_i = k) \Pr(y_i = k \mid x_i) \text{ 其中: } \mathbb{I}(y_i = k) = \begin{cases} 0, & y_i = k \\ 1, & y_i \neq k \end{cases}$$

最大化上式即可求得原问题的最优解.

参考文献

.

[1] 机器学习. 周志华.

[2] Convex Optimization. Boyd and Vandenberghe.

[3] 模式识别. 吴建鑫.

[4] 概率论. 傅冬生等.

[5] https://www.wikipedia.org/

[6] http://users.isr.ist.utl.pt/ mir/pub/probability.pdf

[7] 白板推导机器学习. 哔哩哔哩.