# 机器学习导论 习题四

221300066, 季千$\boxed{\text{F}}$, qkjiai@smail.nju.edu.cn

2024 年 5 月 28 日

## 作业提交注意事项

1. 作业所需的 LaTeX 及 Python 环境配置要求请参考: [Link];
2. 请在 LaTeX 模板中第一页填写个人的学号、姓名、邮箱;
3. 本次作业需提交的文件为:
   (a) 作答后的 LaTeX 代码 — `HW4.tex`;
   (b) 由 (a) 编译得到的 PDF 文件 — `HW4.pdf`;
   (c) 题目 2.(2) 的求解代码文件 — `svm_qp_dual.py`
   (d) 题目 3 的求解代码文件 — `svm_kernel_solution.py`

   其他文件 (如其他代码、图片等) 无需提交. 请将以上文件**打包为 `学号_姓名.zip`** (例如 `221300001_张三.zip`) 后提交;
3. 若多次提交作业, 则在命名 .zip 文件时加上版本号, 例如 `221300001_张三_v1.zip`" (批改时以版本号最高的文件为准);
4. 本次作业提交截止时间为 **5 月 28 日 23:59:59**. 未按照要求提交作业, 提交作业格式不正确, **作业命名不规范**, 将会被扣除部分作业分数; 除特殊原因 (如因病缓交, 需出示医院假条) 逾期未交作业, 本次作业记 0 分; **如发现抄袭, 抄袭和被抄袭双方成绩全部取消**;
5. 学习过程中, 允许参考 ChatGPT 等生成式语言模型的生成结果, 但必须在可信的信息源处核实信息的真实性; **不允许直接使用模型的生成结果作为作业的回答内容**, 否则将视为作业非本人完成并取消成绩;
6. 本次作业提交地址为 [Link], 请大家预留时间提前上交, 以防在临近截止日期时, 因网络等原因无法按时提交作业.

# 1 [35pts] Soft Margin

考虑软间隔 SVM 问题, 其原问题形式如下:

$$
\begin{aligned}
\min_{\boldsymbol{w}, b, \xi_i} \quad & \frac{1}{2}\|\boldsymbol{w}\|_2^2 + C\sum_{i=1}^{m} \xi_i^p \\
\text{s.t.} \quad & y_i\left(\boldsymbol{w}^\top \boldsymbol{x}_i + b\right) \geq 1 - \xi_i \\
& \xi_i \geq 0, i \in [m].
\end{aligned} \tag{1.1}
$$

其中, 松弛变量 $\boldsymbol{\xi} = \{\xi_i\}_{i=1}^m, \xi_i > 0$ 表示样本 $\boldsymbol{x}_i$ 对应的间隔约束不满足的程度, 在优化问题中加入惩罚 $C\sum_{i=1}^m \xi_i^p, C > 0, p \geq 1$ 使得不满足约束的程度尽量小（$\xi_i \to 0$）. 课本式 (6.35) 即为 $p = 1$ 时对应的情况, 此时, 所有违反约束的样本都会受到相同比例的惩罚，而不考虑它们违反约束的程度. 这可能导致模型对较大偏差的样本不够敏感，不足以强调更严重的违规情况. 下面将考虑一些该问题的变式:

(1) **[2+7pts]** 我们首先考虑 $p = 2$ 的情况, 它对于违反约束程度较大的样本提供了更大的惩罚.

  (a) 如课本式 (6.34)-(6.35) 所述, $p = 1$ 的情况对应了 hinge 损失 $\ell_{hinge} : x \to \max(0, 1 - x)$. 请直接写出 $p = 2$ 的情况下对应的损失函数.

  (b) 请推导 $p = 2$ 情况下软间隔 SVM 的对偶问题.

(2) **[14pts]** $p = 1$ 的情况下, 相当于对向量 $\boldsymbol{\xi}$ 使用 $\mathrm{L}_1$ 范数惩罚: $\|\boldsymbol{\xi}\|_1 = \sum_i |\xi_i|$. 现在, 我们考虑使用 $\mathrm{L}_\infty$ 范数惩罚: $\|\boldsymbol{\xi}\|_\infty = \max_i \xi_i$, 这会使得模型着重控制最大的违背约束的程度，从而促使模型在最坏情况下的表现尽可能好. 请推导使用 $\mathrm{L}_\infty$ 范数惩罚的原问题和对偶问题.

(3) **[4+8pts]** 在(1.1)中, 正例和负例在目标函数中分类错误的“惩罚”是相同的. 然而在实际场景中, 很多时候正例和负例错分的“惩罚”代价是不同的 (参考教材 2.3.4 节). 比如考虑癌症诊断问题, 将一个确实患有癌症的人误分类为健康人, 以及将健康人误分类为患有癌症, 产生的错误影响以及代价不应该认为是等同的. 所以我们考虑对负例违反间隔约束的样本施加 $k > 0$ 倍于正例中违反间隔约束的样本的“惩罚”.

  (a) 令(1.1)中 $p = 1$, 并令所有正例样本的集合为 $D_+$, 负例样本的集合为 $D_-$. 请给出相应的 SVM 优化问题.

  (b) 请给出相应的对偶问题.

**Solution.** 此处用于写解答 (中英文均可)

(1) a. 对于 $p = 2$ 的情况，对应的损失函数为:

$$
\ell_{hinge}(x) = \begin{cases} 0 & \text{if } x \geq 1 \\ (1 - x)^2 & \text{otherwise} \end{cases}
$$

b. $p = 2$ 情况下软间隔 SVM 的对偶问题的推导:

$L(\boldsymbol{w}, \mathbf{b}, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\mu}) = \frac{1}{2}\|\boldsymbol{w}\|_2^2 + C\sum_{i=1}^m \xi_i^2 + \sum_{i=1}^m \alpha_i(1 - \xi_i - y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b)) - \sum_{i=1}^m \mu_i \xi_i$

其中 $\alpha_i \geq 0, \mu_i \geq 0$ 是拉格朗日乘子.
对 $\boldsymbol{w}, \mathbf{b}, \xi$ 求导可得:
$\frac{\partial L}{\partial \boldsymbol{w}} = \boldsymbol{w} - \sum_{i=1}^m \alpha_i y_i \boldsymbol{x}_i = 0 \Rightarrow \hat{\boldsymbol{w}} = \sum_{i=1}^m \alpha_i y_i \boldsymbol{x}_i$
$\frac{\partial L}{\partial b} = -\sum_{i=1}^m \alpha_i y_i = 0 \Rightarrow \sum_{i=1}^m \alpha_i y_i = 0$
$\frac{\partial L}{\partial \xi_i} = 2C\xi_i - \alpha_i - \mu_i = 0 \Rightarrow \alpha_i + \mu_i = 2C\xi_i$
带入可得:
$L(\boldsymbol{\alpha}, \boldsymbol{\mu}) = \sum_{i=1}^m \alpha_i - \frac{1}{2}\sum_{i=1}^m \sum_{j=1}^m \alpha_i \alpha_j y_i y_j \boldsymbol{x}_i^\top \boldsymbol{x}_j - \frac{1}{4C}\sum_{i=1}^n (\alpha_i + \mu_i)^2$

$$
\begin{aligned}
\max_{\boldsymbol{\alpha}, \boldsymbol{\mu}} \quad & \sum_{i=1}^m \alpha_i - \frac{1}{2}\sum_{i=1}^m \sum_{j=1}^m \alpha_i \alpha_j y_i y_j \boldsymbol{x}_i^\top \boldsymbol{x}_j - \frac{1}{4C}\sum_{i=1}^n (\alpha_i + \mu_i)^2 \\
\text{s.t.} \quad & \sum_{i=1}^n \alpha_i y_i = 0 \\
& 0 \leq \alpha_i, i = 1, 2, ..., m, \quad 0 \leq \mu_i, i = 1, 2, ..., m.
\end{aligned}
\tag{1.2}
$$

(2) 原问题的优化目标为:

$$\min_{\boldsymbol{w}, b, \xi, t} \quad \frac{1}{2}\|\boldsymbol{w}\|_2^2 + Ct$$

约束条件为:

$$y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) \geq 1 - \xi_i, i \in [m]$$

$$t \geq \xi_i \geq 0, i \in [m]$$

$L(\boldsymbol{w}, \mathbf{b}, \xi, \alpha, \mu, \beta, t) = \frac{1}{2}\|w\|_2^2 + Ct + \sum_{i=1}^m \alpha_i(1 - \xi_i - y_i(w^\top x_i + b)) - \sum_{i=1}^m \mu_i \xi_i + \sum_{i=1}^m \beta_i(\xi_i - t)$
其中 $\alpha_i \geq 0, \mu_i \geq 0, \beta_i \geq 0$ 是拉格朗日乘子.
对 $\boldsymbol{w}, \mathbf{b}, \xi, t$ 求导可得:
$\frac{\partial L}{\partial \boldsymbol{w}} = \boldsymbol{w} - \sum_{i=1}^m \alpha_i y_i \boldsymbol{x}_i = 0 \Rightarrow \boldsymbol{w} = \sum_{i=1}^m \alpha_i y_i \boldsymbol{x}_i$
$\frac{\partial L}{\partial b} = -\sum_{i=1}^m \alpha_i y_i = 0 \Rightarrow \sum_{i=1}^m \alpha_i y_i = 0$
$\frac{\partial L}{\partial \xi_i} = \beta_i - \alpha_i - \mu_i = 0 \Rightarrow \alpha_i + \mu_i = \beta_i$
$\frac{\partial L}{\partial t} = C - \sum_{i=1}^m \beta_i = 0 \Rightarrow \sum_{i=1}^m \beta_i = C$
带入可得对偶问题为:

$$
\begin{aligned}
\max_{\boldsymbol{\alpha}} \quad & \sum_{i=1}^m \alpha_i - \frac{1}{2}\sum_{i=1}^m \sum_{j=1}^m \alpha_i \alpha_j y_i y_j \boldsymbol{x}_i^\top \boldsymbol{x}_j \\
\text{s.t.} \quad & \sum_{i=1}^n \alpha_i y_i = 0 \\
& 0 \leq \alpha_i, i = 1, 2, ..., m, \quad \sum_{i=1}^m \alpha_i \leq C.
\end{aligned}
\tag{1.3}
$$

(3) a. 对于 $p = 1$ 的情况，正例和负例违反间隔约束的“惩罚”代价不同，可以引入权重系数 $k > 0$。此时，SVM 优化问题为:

$$\min_{\boldsymbol{w}, b, \xi} \quad \frac{1}{2}\|\boldsymbol{w}\|_2^2 + C\sum_{i \in D_+} k\xi_i + kC\sum_{i \in D_-} k\xi_i$$

约束条件为:

$$y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) \geq 1 - \xi_i,$$

$$\xi_i \geq 0, i \in [m]$$

b. 对应的对偶问题为:

$$\begin{aligned}
\max_{\boldsymbol{\alpha},\boldsymbol{\mu}} \quad & \sum_{i=1}^{m} \alpha_i - \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j \boldsymbol{x}_i^\top \boldsymbol{x}_j \\
\text{s.t.} \quad & \sum_{i=1}^{n} \alpha_i y_i = 0 \\
& 0 \leq \alpha_i \leq C, i \in D_+, \quad 0 \leq \alpha_i \leq kC, i \in D_-,.
\end{aligned} \tag{1.4}$$

## 2 [20pts] Primal and Dual Problem

给定一个包含 $m$ 个样本的数据集 $D = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^m$, 其中每个样本的特征维度为 $d$, 即 $\boldsymbol{x}_i \in \mathbb{R}^d$. 软间隔 SVM 的原问题和对偶问题可以表示为:

原问题:

$$\begin{aligned}\min_{\boldsymbol{w},b,\xi} \quad & \frac{1}{2}\boldsymbol{w}^\top \boldsymbol{w} + C\sum_{i=1}^n \xi_i \\ \text{s.t.} \quad & y_i\left(\boldsymbol{w}^\top \boldsymbol{x}_i + b\right) \geq 1 - \xi_i, \forall i \in [m] \\ & \xi_i \geq 0, \forall i \in [m]\end{aligned}$$

对偶问题:

$$\begin{aligned}\min_{\boldsymbol{\alpha}} \quad & \frac{1}{2}\boldsymbol{\alpha}^\top \boldsymbol{Q}\boldsymbol{\alpha} - \boldsymbol{1}_m^\top \boldsymbol{\alpha} \\ \text{s.t.} \quad & \boldsymbol{y}^\top \boldsymbol{\alpha} = 0 \\ & 0 \leq \alpha_i \leq C, \forall i \in [m]\end{aligned}$$

其中, 对于任意 $i, j \in [m]$ 有 $Q_{ij} \equiv y_i y_j \boldsymbol{x}_i^\top \boldsymbol{x}_j$.

上述的原问题和对偶问题都是都是二次规划 (Quadratic Programming) 问题, 都可以使用相关软件包求解. 本题目中我们将通过实践来学习凸优化软件包的使用, 并以软间隔 SVM 为例了解原问题、对偶问题在优化方面的特性.

(1) **[2pts]** 请直接写出原问题和对偶问题的参数量 (注意参数只包含分类器所保存的参数, 不包含中间变量).

(2) **[10pts]** 请参考 `lab2/svm_qp.py` 中对于原问题的求解代码, 编写对偶问题的求解代码 `lab2/svm_qp_dual.py`. (这里使用了 CVXPY 求解 QP 问题.) 请将代码提交至下方的解答处.

(3) **[8pts]** 设特征维度和样例数量的比值 $r = \frac{d}{m}$, 请绘制原问题和对偶问题的求解速度随着这个比值变化的曲线图. 并简述: 何时适合求解原问题, 何时适合求解对偶问题?

**Solution.** 此处用于写解答 (中英文均可)

(1) 原问题的参数量是 $d + 1 + m$, 对偶问题的参数量是 $m$

(2) 对偶问题的求解代码为:

```
def solve_dual(X, y, C):
    '''
    :参数 X: ndarray, 形状为(m, d), 样例矩阵
    :参数 y: ndarray, 形状为(m), 样例标签向量
    :参数 C: 标量, 含义与教材式(6.35)中C相同
    :返回: alpha, SVM的对偶变量
    '''
    m, d = X.shape
    alpha = cp.Variable(m)
    y = y.astype(float).reshape(-1, 1)
    c_ = np.full((m,1),C)

    K = X @ X.T
    q_1 = np.outer(y, y) * K

    prob = cp.Problem(
      cp.Minimize(0.5 * cp.quad_form(alpha, q_1) -cp.sum(alpha)), # 目标函数
        [alpha >= 0, alpha <= c_, y.T @ alpha == 0] # 约束
```

```
        )
    prob.solve()
    return alpha.value
```

(3) 曲线图为:

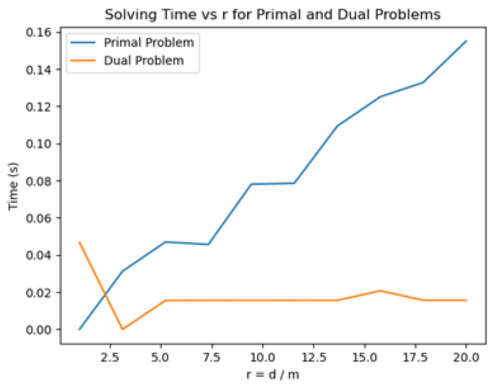



图 1: 原问题和对偶问题的求解速度随比值变化的曲线图

简述题: 当特征维度和样例数量的比值 $r = \frac{d}{m}$ 较大时，适合求解对偶问题；当 $r$ 较小时，适合求解原问题。这是因为在高维空间中，数据点之间的距离较远，对偶问题的参数量相对较少，使用对偶问题可以更好地处理这种情况。同理在低维空间中，数据点之间的距离较近，使用原问题可以更好地处理这种情况。

# 3 [15pts] Kernel Function in Practice

`lab3/svm_kernel.py` 中构造了异或 (XOR) 问题, 如图 2 所示. 该问题是线性不可分的.

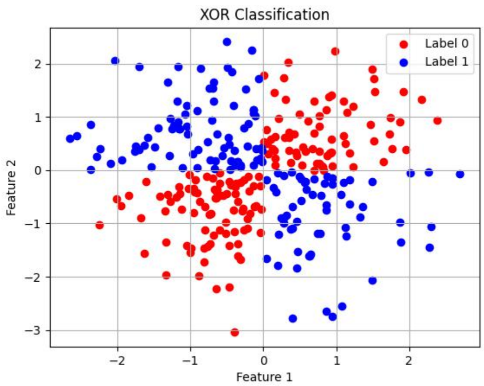



图 2: 异或 (XOR) 问题

本题中我们将通过实验了解核函数的选择对于 SVM 解决非线性问题的影响. 请使用 sklearn 包中的 SVM 分类器完成下述实验:

(1) **[6pts]** 请分别训练线性核 SVM 分类器和高斯核 (RBF 核) SVM 分类器, 并绘制出各自的决策边界.

(2) **[6pts]** sklearn 还允许自定义核函数, 参考 `lab3/svm_kernel_custom.py` 的用法, 编写核函数 $\kappa(\boldsymbol{x}, \boldsymbol{x}') = \frac{1}{1+\|\boldsymbol{x}-\boldsymbol{x}'\|_2^2}$, 训练该核函数的 SVM 分类器, 并绘制出决策边界.

具体的实验要求可以参考 `lab3/svm_kernel.py` 的 `main` 部分. 请将 `lab3/svm_kernel_solution.py` 中的代码和三个核函数分别对应的决策边界图提交至下方的解答处.

最后, 请直接回答 **[3pts]**: 三个核函数, 各自能够解决异或 (XOR) 分类问题吗?

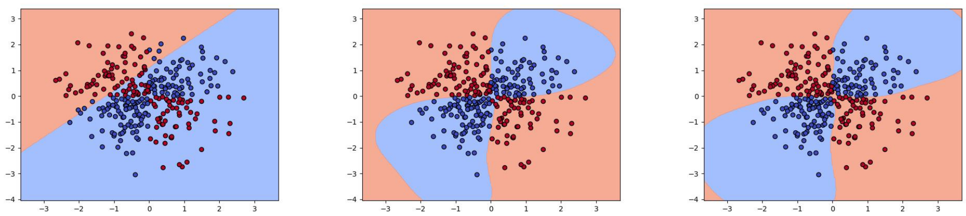
**Solution.** 此处用于写解答 (中英文均可)

求解代码为:

```python
from sklearn.svm import SVC
import numpy as np

def svm_kernel_linear(X, Y):
    '''
    :参数 X: ndarray, 形状(m, d), 样例矩阵
    :参数 Y: ndarray, 形状(m), 样例标签向量
    :返回: clf_linear, 训练好的分类器
    '''
    clf_linear = SVC(kernel='linear')
    clf_linear.fit(X, Y)
    return clf_linear

def svm_kernel_rbf(X, Y):
    '''
    :参数 X: ndarray, 形状(m, d), 样例矩阵
    :参数 Y: ndarray, 形状(m), 样例标签向量
```

```python
    :返回: clf_rbf, 训练好的分类器
    '''
    clf_rbf = SVC(kernel='rbf')
    clf_rbf.fit(X, Y)
    return clf_rbf

def custom_kernel(X1, X2):
    '''
    :参数 X1: ndarray, 形状(m, d)
    :参数 X2: ndarray, 形状(n, d)
    :返回: 形状为(m, n)的Gram矩阵, 第(i,j)个元素为X1[i]和X2[j]之间的核函数值
    '''
    gram_matrix  = np.sum(X1**2, axis=1).reshape(-1, 1) + np.sum(X2**2, axis=1)- 2 * np.dot(X1, X2.T)
    return  1 / (1 + gram_matrix)

def svm_kernel_custom(X, Y):
    '''
    :参数 X: ndarray, 形状(m, d), 样例矩阵
    :参数 Y: ndarray, 形状(m), 样例标签向量
    :返回: clf_custom, 训练好的分类器
    '''
    # hint: 需要使用 custom_kernel 函数
    clf_custom = SVC(kernel=custom_kernel)
    clf_custom.fit(X, Y)
    return clf_custom
```

决策边界为:



(a) clf_linear　(b) clf_rbf　(c) clf_custom

能否解决异或 (XOR) 分类问题: 线性核函数不能解决, 高斯核函数和自定义核函数可以解决.

## 4 [30pts] Maximum Likelihood Estimation

给定由 $m$ 个样本组成的训练集 $D=\{(\boldsymbol{x}_1,y_1),(\boldsymbol{x}_2,y_2),\cdots,(\boldsymbol{x}_m,y_m)\}$, 其中 $\boldsymbol{x}_i\in\mathbb{R}^d$ 是第 $i$ 个示例, $y_i\in\mathbb{R}$ 是对应的实值标记. 令 $\boldsymbol{X}\in\mathbb{R}^{m\times d}$ 表示整个训练集中所有样本特征构成的矩阵, 并令 $\boldsymbol{y}\in\mathbb{R}^m$ 表示训练集中所有样本标记构成的向量. 线性回归的目标是寻找一个参数向量 $\boldsymbol{w}\in\mathbb{R}^d$, 使得在训练集上模型预测的结果和真实标记之间的差距最小. 对于一个样本 $\boldsymbol{x}$, 线性回归给出的预测为 $\hat{y}=\boldsymbol{w}^\top\boldsymbol{x}$,[1] 它与真实标记 $y$ 之间的差距可以用平方损失 $(\hat{y}-y)^2$ 来描述. 因此, 在整个训练集上最小化损失函数的过程可以写作如下的优化问题:

$$\boldsymbol{w}^\star=\operatorname*{arg\,min}_{\boldsymbol{w}}\|\boldsymbol{X}\boldsymbol{w}-\boldsymbol{y}\|_2^2 \tag{4.1}$$

(1) **[8pts]** 考虑这样一种概率观点: 样本 $\boldsymbol{x}$ 的标记 $y$ 是从一个高斯分布 $\mathcal{N}(\boldsymbol{w}^\top\boldsymbol{x},\sigma^2)$ 中采样得到的. 这个高斯分布的均值由样本特征 $\boldsymbol{x}$ 和模型参数 $\boldsymbol{w}$ 共同决定, 而方差是一个额外的参数 $\sigma^2$. 基于这种概率观点, 我们可以基于观测数据对高斯分布中的参数 $\boldsymbol{w}$ 做极大似然估计. 请证明: $\boldsymbol{w}$ 的极大似然估计结果 $\boldsymbol{w}_{\mathrm{MLE}}$ 与式 (4.1) 中的 $\boldsymbol{w}^\star$ 相等;

(2) **[9pts]** 极大似然估计容易过拟合, 一种常见的解决办法是采用最大后验估计: 沿着上一小问的思路, 现在我们希望在概率建模下对参数 $\boldsymbol{w}$ 做最大后验估计. 为此, 引入参数 $\boldsymbol{w}$ 上的先验 $p(\boldsymbol{w})=\mathcal{N}(\boldsymbol{0},\lambda\boldsymbol{I})$. 其中, 均值 $\boldsymbol{0}$ 是 $d$ 维的全 0 向量, $\boldsymbol{I}$ 是 $d$ 维单位矩阵, $\lambda>0$ 是一个控制方差的超参数. 现在, 请推导对 $\boldsymbol{w}$ 做最大后验估计的目标函数, 并讨论一下该结果与 "带有 $\mathrm{L}_2$ 范数正则项的线性回归" 之间的关系;

(3) **[9pts]** 沿着上一小问的思路, 我们尝试给参数 $\boldsymbol{w}$ 施加一个拉普拉斯先验. 简便起见, 我们假设参数 $\boldsymbol{w}$ 的 $d$ 个维度之间是独立的, 且每一维都服从 0 均值的一元拉普拉斯分布, 即:

$$\begin{aligned} p(\boldsymbol{w}) &= \prod_{j=1}^{d} p(w_j)\ , \\ p(w_j) &= \mathrm{Lap}(w_j \mid 0,\lambda),\ j=1,2,\ldots,d\ . \end{aligned} \tag{4.2}$$

请推导对 $\boldsymbol{w}$ 做最大后验估计的目标函数, 并讨论一下该结果与 "带有 $\mathrm{L}_1$ 范数正则项的线性回归" 之间的关系;

*Note: 由参数 $\mu$, $\lambda$ 确定的一元拉普拉斯分布的概率密度函数为:*

$$\mathrm{Lap}(w\mid\mu,\lambda)=\frac{1}{2\lambda}\exp\left(-\frac{|w-\mu|}{\lambda}\right)\ . \tag{4.3}$$

(4) **[4pts]** 基于 (2) 和 (3) 的结果, 从概率角度讨论为什么 $\mathrm{L}_1$ 范数能使模型参数更稀疏.

**Solution.** 此处用于写解答 (中英文均可)

(1) 对于高斯分布，其对数似然函数为:

$$\begin{aligned} \log p(\boldsymbol{y}|\boldsymbol{w}) &= \log\prod_{i=1}^{m}\mathcal{N}(y_i|\boldsymbol{w}^\top\boldsymbol{x}_i,\sigma^2) \\ n &= \sum_{i=1}^{m}\log\mathcal{N}(y_i|\boldsymbol{w}^\top\boldsymbol{x}_i,\sigma^2) \\ n &= \frac{m}{2}\log(2\pi\sigma^2)+\frac{1}{2\sigma^2}\|\boldsymbol{X}\boldsymbol{w}-\boldsymbol{y}\|_2^2 \end{aligned}$$

[1]本题不考虑偏移 $b$, 可参考教材第 3 章将偏移 $b$ 吸收进 $\boldsymbol{w}$.

要找到一个 w 使得 $\ln L(\hat{\boldsymbol{w}}|\mathbf{X},\boldsymbol{y})$ 最大, 即 $\hat{\boldsymbol{w}}_{MLE}=\underset{\boldsymbol{w}}{\arg\max}\ln L(\hat{\boldsymbol{w}}|\mathbf{X},\boldsymbol{y})$ 前一项显然与 w 无关, 后一项和式 (4.1) 成反比. 最大化对数似然函数等价于最小化平方损失函数。因此, $\boldsymbol{w}_{\mathrm{MLE}}$ 与式 (4.1) 中的 $\boldsymbol{w}^{\star}$ 显然相等。

(2) 由题意知，此分布的概率密度函数为:

$$p(\boldsymbol{w})=\frac{1}{(2\pi\lambda)^{d/2}}\exp\left(-\frac{\|\boldsymbol{w}\|^2}{2\lambda}\right)$$

我们需要计算参数 $\boldsymbol{w}$ 的后验分布 $p(\boldsymbol{w}|\boldsymbol{y},\mathbf{X})$，根据贝叶斯公式:

$$p(\boldsymbol{w}|\boldsymbol{y},\mathbf{X})=\frac{p(\boldsymbol{y}|\mathbf{X},\boldsymbol{w})p(\boldsymbol{w})}{p(\boldsymbol{y}|\mathbf{X})}$$

$$\log p(\boldsymbol{w}|\boldsymbol{y},\mathbf{X})=\log p(\boldsymbol{y}|\mathbf{X},\boldsymbol{w})+\log p(\boldsymbol{w})-\log p(\boldsymbol{y}|\mathbf{X})=\log p(\boldsymbol{y}|\mathbf{X},\boldsymbol{w})+\log p(\boldsymbol{w})$$

$$\log p(\boldsymbol{w}|\boldsymbol{y},\mathbf{X})=-\frac{m}{2}\log(2\pi\sigma^2)-\frac{1}{2\sigma^2}\sum_{i=1}^{m}(y_i-\boldsymbol{w}^T x_i)^2-\frac{d}{2}\log(2\pi\lambda)-\frac{1}{2\lambda}\|\boldsymbol{w}\|^2$$

$$\boldsymbol{w}_{\mathrm{MAP}}=\underset{\boldsymbol{w}}{\arg\min}\left(\frac{1}{2\sigma^2}\sum_{i=1}^{m}(y_i-\boldsymbol{w}^T\mathbf{x}_i)^2+\frac{1}{2\lambda}\|\boldsymbol{w}\|^2\right)$$

$$\boldsymbol{w}_{\mathrm{MAP}}=\underset{\boldsymbol{w}}{\arg\min}\left(\|\mathbf{X}\boldsymbol{w}-\boldsymbol{y}\|^2+\sigma^2\lambda\|\boldsymbol{w}\|^2\right)$$

这个目标函数正是带有 $\mathrm{L}_2$ 正则项的线性回归，其正则化系数为 $\frac{\sigma^2}{\lambda}$.

(3) 推导过程:

$p(\boldsymbol{w})=\prod_{j=1}^{d}p(\boldsymbol{w}_j),\quad p(\boldsymbol{w}_j)=\mathrm{Lap}(\boldsymbol{w}_j|0,\lambda)=\frac{1}{2\lambda}\exp\left(-\frac{|\boldsymbol{w}_j|}{\lambda}\right)$

$\log p(\boldsymbol{w}|\boldsymbol{y},\mathbf{X})=\log p(\boldsymbol{y}|\mathbf{X},\boldsymbol{w})+\log p(\boldsymbol{w})-\log p(\boldsymbol{y}|\mathbf{X})=\log p(\boldsymbol{y}|\mathbf{X},\boldsymbol{w})+\log p(\boldsymbol{w})$

$\log p(\boldsymbol{w}|\boldsymbol{y},\mathbf{X})=-\frac{m}{2}\log(2\pi\sigma^2)-\frac{1}{2\sigma^2}\sum_{i=1}^{m}(y_i-\boldsymbol{w}^\top\boldsymbol{x}_i)^2+\sum_{j=1}^{d}\log\frac{1}{2\lambda}\exp(-|w_j|/\lambda)$

$\log p(\boldsymbol{w}|\boldsymbol{y},\mathbf{X})=-\frac{m}{2}\log(2\pi\sigma^2)-\frac{1}{2\sigma^2}\sum_{i=1}^{m}(y_i-\boldsymbol{w}^\top\boldsymbol{x}_i)^2-\sum_{j=1}^{d}\log(2\lambda)+|\boldsymbol{w}_j|/\lambda$

$\boldsymbol{w}_{\mathrm{MAP}}=\underset{\boldsymbol{w}}{\arg\min}\|\mathbf{X}\boldsymbol{w}-\boldsymbol{y}\|_2^2+\frac{2\sigma^2}{\lambda}\|\boldsymbol{w}\|_1$

这个目标函数正是带有 $\mathrm{L}_1$ 正则项的线性回归，其正则化系数为 $\frac{2\sigma^2}{\lambda}$.

(4) 从概率角度来说，$\mathrm{L}_1$ 范数对应于拉普拉斯分布，这种分布鼓励模型参数中的许多值接近零, 因为拉普拉斯分布在 0 处的陡峭, 可以使模型参数更稀疏。它对参数的绝对值进行惩罚，使得参数更倾向于取较小的值。这样，在最大后验估计中，参数的先验分布会偏向于较小的值，从而使得模型参数更加稀疏。

## Acknowledgments

允许与其他同样未完成作业的同学讨论作业的内容, 但需在此注明并加以致谢; 如在作业过程中, 参考了互联网上的资料 (包括生成式模型的结果), 且对完成作业有帮助的, 亦需注明并致谢.