机器学习实验报告 (二)

[redacted]

2018/11/18

目录

1 题目描述与分析

1.1 问题描述

该问题要求我们考虑不同维数下的高斯概率密度模型均值与方差的最大似然估计。

一共分为六小问，前三问要求我们编写程序计算一维、二维、三维数据情形下均值与方差 (协方差) 的的最大似然估计结果；第四问考查可分离三维高斯模型下均值与协方差矩阵；后两问让我们比较并解释前四问中计算出来的均值与方差。

1.2 问题分析

参数估计问题是统计学中的经典问题，最常见的是最大似然估计与贝叶斯估计。两者最大的区别在于，前者把待估计的参数看作取值未知的确定的量，而后者把待估计的参数看成是符合某种先验概率分布的随机变量。本题主要考察最大似然估计。

(a) 此小问让我们求解一维情况下的特征均值与方差的最大似然估计。均值 μ 与方差 σ^2 均未知。在单变量的情况下，参数向量 $\boldsymbol{\theta}$ 的组成成分是:$\theta_1=\mu$,$\theta_2=\sigma^2$ 。我们知道一维正态分布形式如下:

$$f\left(x\right)=\frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{\left(x-\mu\right)^2}{2\sigma^2}\right) \tag{1}$$

因此对于单个训练样本，简单变化成对数似然函数有如下形式:

$$\ln p\left(x_k|\boldsymbol{\theta}\right)=-\frac{1}{2}\ln 2\pi\theta_2-\frac{1}{2\theta_2}\left(x_k-\theta_1\right)^2 \tag{2}$$

对 (2) 式关于 $\boldsymbol{\theta}$ 求导可得:

$$\boldsymbol{\nabla}_{\boldsymbol{\theta}}l=\boldsymbol{\nabla}_{\boldsymbol{\theta}}\ln p\left(x_k|\boldsymbol{\theta}\right)=\begin{bmatrix}\dfrac{1}{\theta_2}\left(x_k-\theta_1\right)\\-\dfrac{1}{2\theta_2}+\dfrac{\left(x_k-\theta_1\right)^2}{2\theta_2^2}\end{bmatrix} \tag{3}$$

又因为

$$\nabla_{\boldsymbol{\theta}} l = 0 \tag{4}$$

是求解最大似然估计值 $\boldsymbol{\theta}$ 的必要条件，因此对全部样本对数似然函数求和可以得到如下极值条件:

$$\sum_{k=1}^{n} \frac{1}{\widehat{\theta}_2} \left(x_k - \widehat{\theta}_1 \right) = 0 \tag{5}$$

和

$$-\sum_{k=1}^{n} \frac{1}{\widehat{\theta}_2} + \sum_{k=1}^{n} \frac{\left(x_k - \widehat{\theta}_1 \right)^2}{\widehat{\theta}_2^2} \tag{6}$$

将 $\widehat{\theta}_1 = \widehat{\mu}$,$\widehat{\theta}_2 = \widehat{\sigma^2}$ 代入 (6) 式，化简可得:

$$\widehat{\mu} = \frac{1}{n} \sum_{k=1}^{n} x_k \tag{7}$$

和

$$\widehat{\sigma^2} = \frac{1}{n} \sum_{k=1}^{n} \left(x_k - \widehat{\mu} \right)^2 \tag{8}$$

(7)(8) 两式就是我们要求解的值，代入相应的样本即可

(b) 此小问让我们求解二维情况下的特征均值与协方差矩阵的最大似然估计。我们知道一般的 d 维多元正态密度的形式如下

$$f(\mathbf{x}) = \frac{1}{(2\pi)^{d/2} |\mathbf{\Sigma}|^{1/2}} \exp\left[-\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu})^t \mathbf{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}) \right] \tag{9}$$

类似于 (a) 小问, 均值 μ 与协方差矩阵 $\mathbf{\Sigma}$ 均未知。在单变量的情况下，参数向量 $\boldsymbol{\theta}$ 的组成成分是:$\theta_1 = \boldsymbol{\mu}$,$\theta_2 = \mathbf{\Sigma}$ 。于是我们可以得到单训练样本的对数似然函数:

$$\ln p(\boldsymbol{x}_k | \boldsymbol{\theta}) = -\frac{1}{2} \ln\left[(2\pi)^d |\theta_2| \right] - \frac{1}{2} (\boldsymbol{x}_k - \theta_1)^t \theta_2^{-1} (\boldsymbol{x}_k - \theta_1) \tag{10}$$

对上式两边对 $\boldsymbol{\theta}$ 求导以及运用 (4) 式最终可以得到:

$$\widehat{\boldsymbol{\mu}} = \frac{1}{n}\sum_{k=1}^{n}\boldsymbol{\mu}_k \tag{11}$$

$$\widehat{\boldsymbol{\Sigma}} = \frac{1}{n}\sum_{k=1}^{n}\left(\boldsymbol{x}_k - \widehat{\boldsymbol{\mu}}\right)\left(\boldsymbol{x}_k - \widehat{\boldsymbol{\mu}}\right)^t \tag{12}$$

以上两式即为要求的多元高斯分布的均值与协方差矩阵的最大似然估计结果，代入相应的值即可

(c) 此小问让我们求解三维情况下的特征均值与协方差矩阵的最大似然估计。与 (b) 问公式相同，带入相应的值即可

(d) 题目假设这个三维高斯模型是可分离的，协方差矩阵为对角矩阵，则我们可以推断出 ω_2 的三个特征值是两两独立的，从而协方差矩阵对角线元素分别为三个特征的方差，根据 (7) 式代入相应的值就能求出协方差矩阵中的三个参数。ω_2 中的均值可使用 (11) 式求出。

(e) 与 **(f)** 在求解出结果来以后再作分析。

2 具体实现

2.1 库的导入及数据准备

```
import numpy as np #矩阵运算
lists1=np.array([[0.42,-0.087,0.58],
                 [-0.2,-3.3,-3.4],
                 [1.3,-0.32,1.7],
                 [0.39,0.71,0.23],
                 [-1.6,-5.3,-0.15],
                 [-0.029,0.89,-4.7],
```

```
                 [-0.23,1.9,2.2],
                 [0.27,-0.3,-0.87],
                 [-1.9,0.76,-2.1],
                 [0.87,-1.0,-2.6]])
lists2=np.array([[-0.4,0.58,0.089],
                 [-0.31,0.27,-0.04],
                 [0.38,0.055,-0.035],
                 [-0.15,0.53,0.011],
                 [-0.35,0.47,0.034],
                 [0.17,0.69,0.1],
                 [-0.011,0.55,-0.18],
                 [-0.27,0.61,0.12],
                 [-0.065,0.49,0.0012],
                 [-0.12,0.054,-0.063]])
```

2.2 (a) 问

在这里，我们定义了一个 $getVar1()$ 函数, 用于求解方差。传入的参数 $points$ 代表特征的坐标。为了求解均值向量，我们定义 $getMean1()$ 函数。具体代码如下:

```
def getMean1(points):
    return np.mean(points)
def getVar1(points):
    u = getMean1(points)
    n = len(points)
    re = 0
    for i in points:
        re = re + (i-u)**2
    return re/n
```

随后编写调用函数，对 (a) 小问进行求解:

```
for i in range(3):
    print("特征x",i+1,"的均值的最大似然估计为",
    '%.8f' % getMean1(lists1[:,i]))
    print("特征x",i+1,"的方差的最大似然估计为",
    '%.8f' % getVar1(lists1[:,i]))
```

所得结果如下所示:

特征 x_1 的均值 μ 的最大似然估计为 *-0.07090000*
特征 x_1 的方差 σ^2 的最大似然估计为 *0.90617729*
特征 x_2 的均值 μ 的最大似然估计为 *-0.60470000*
特征 x_2 的方差 σ^2 的最大似然估计为 *4.20071481*
特征 x_3 的均值 μ 的最大似然估计为 *-0.91100000*
特征 x_3 的方差 σ^2 的最大似然估计为 *4.54194900*

2.3 (b) 问

首先我们定义了 *getTheta*() 函数来求解协方差矩阵的最大似然估计，需要传入的参数 *points* 为 *numpy.ndarray* 类型，具体代码如下:

```
#均值的最大似然估计为全体样本的平均值
def getMean(points):
#输入为矩阵  求列均值
    return np.mean(points,0)
def getTheta(points):
    u = getMean(points)
    n,d = points.shape
    re_matrix=np.zeros([d,d])
    for i in points:
        temp = np.matrix(i)-u
```

```
        re_matrix = re_matrix + np.dot(temp.T,temp)
    return re_matrix/n
```

在这我们编写调用函数求解任意两个组合均值与协方差矩阵的最大似然估计:

```
print("特征x1,x2的均值的最大似然估计为\n",
 getMean(lists1[:,[0,1]]))
print("特征x1,x2的协方差矩阵的最大似然估计为\n",
getVar(lists1[:,[0,1]]))
print("特征x1,x3的均值的最大似然估计为\n",
getMean(lists1[:,[0,2]]))
print("特征x1,x3的协方差矩阵的最大似然估计为\n",
 getVar(lists1[:,[0,2]]))
print("特征x2,x3的均值 的最大似然估计为\n",
 getMean(lists1[:,[1,2]]))
print("特征x2,x3的协方差矩阵的最大似然估计为\n",
 getVar(lists1[:,[1,2]]))
```

整理得到的结果为:

特征 x_1,x_2 的均值 μ 的最大似然估计为 $\begin{bmatrix} -0.0709 & -0.6047 \end{bmatrix}$

特征 x_1,x_2 的协方差矩阵 σ^2 的最大似然估计为 $\begin{bmatrix} 0.90617729 & 0.56778177 \\ 0.56778177 & 4.20071481 \end{bmatrix}$

特征 x_1,x_3 的均值 μ 的最大似然估计为 $\begin{bmatrix} -0.0709 & -0.911 \end{bmatrix}$

特征 x_1,x_3 的协方差矩阵 σ^2 的最大似然估计为 $\begin{bmatrix} 0.90617729 & 0.3940801 \\ 0.3940801 & 4.541949 \end{bmatrix}$

特征 x_2,x_3 的均值 μ 的最大似然估计为 $\begin{bmatrix} -0.6047 & -0.911 \end{bmatrix}$

特征 x_2,x_3 的协方差矩阵 σ^2 的最大似然估计为 $\begin{bmatrix} 4.20071481 & 0.7337023 \\ 0.7337023 & 4.541949 \end{bmatrix}$

2.4 (c) 问

与 (b) 问相似，直接编写调用函数

```
print("w1的均值的最大似然估计为")
print(getMean(lists1))
print("w1协方差矩阵的最大似然估计为")
print(getVar(lists1))
```

整理得到的结果为:

ω_1 的均值 μ 的最大似然估计为

$\begin{bmatrix} -0.0709 & -0.6047 & -0.911 \end{bmatrix}$

ω_1 协方差矩阵 σ^2 的最大似然估计为

$$\begin{bmatrix} 0.90617729 & 0.56778177 & 0.3940801 \\ 0.56778177 & 4.20071481 & 0.7337023 \\ 0.3940801 & 0.7337023 & 4.541949 \end{bmatrix}$$

2.5 (d) 问

根据前边的分析，协方差矩阵中三个参数即为 ω_2 中三个特征各自的方差，而均值可直接求出，下面是调用函数:

```
print("w1的均值的最大似然估计为")
print(getMean(lists2))
for i in range(3):
    print("特征x",i+1,"的均值的最大似然估计为 ",
    '%.8f' % getVar1(lists2[:,i]))
```

整理得到结果如下:

ω_2 的均值 μ 的最大似然估计为

$\begin{bmatrix} -0.1126 & 0.4299 & 0.00372 \end{bmatrix}$

特征 x_1 的方差 σ^2 的最大似然估计为 *0.05392584*

特征 x_2 的方差 σ^2 的最大似然估计为 *0.04597009*

特征 x_3 的方差 σ^2 的最大似然估计为 *0.00726551*

3 问题再分析

3.1 (e) 问

通过观察我们可以发现，对于特征的均值 μ, 每个特征 x_i 对应的均值是相同的。多维数据的均值有单个特征的均值组合而成。这是因为我们在求组合的均值时，仍然是按照每个特征单独计算的

3.2 (f) 问

可以看到，多维协方差矩阵对角线元素为对应特征的方差，而非对角线元素 Σ_{ij} 表示了特征 x_i 与 x_j 之间的协方差，当 x_i 与 x_j 相互独立时，他们之间的协方差 $Cov\left(x_i, x_j\right) = 0$，因此当三维高斯模型可独立时，特征之间两两独立，此时协方差矩阵也就变成了对角矩阵，其对角线元素为各特征各自的方差 σ^2

4 总结与收获

通过这次实验，我更加深刻的理解了最大似然估计，以及求均值与协方差矩阵的公式推导过程。在学习过程中，又厘清了独立、不相关、协方差为 0 各自的含义以及他们之间的关系，相信在日后的学习中会应用到。