模式识别与机器学习 23-24

第四章作业

韦诗睿

202328002509044

https://github.com/RuiNov1st/UCAS_PRML_2324

2023 年 11 月 5 日

第四章 KL 变换

题 1：

设有如下两类样本集，其出现的概率相等：
ω_1:$\{(0\ 0\ 0)^T, (1\ 0\ 0)^T, (1\ 0\ 1)^T, (1\ 1\ 0)^T\}$
ω_2:$\{(0\ 0\ 1)^T, (0\ 1\ 0)^T, (0\ 1\ 1)^T, (1\ 1\ 1)^T\}$
用 K-L 变换，分别把特征空间维数降到二维和一维，并画出样本在该空间中的位置。

答：

由题，两类样本出现的概率相等，即 $P(\omega_1) = P(\omega_2) = 0.5$。样本矩阵记为：

$$\mathbf{X} = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}\ \mathbf{Y} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \end{bmatrix}$$

i. **检查样本均值是否符合最佳变换条件**
将 ω_1 和 ω_2 两类模式作为一个整体考虑，有整体均值：

$$\begin{aligned}
\mathbf{M} &= P(\omega_1)\mathrm{E}(\mathbf{X}) + P(\omega_2)\mathrm{E}(\mathbf{Y}) \\
&= 0.5 \times \frac{1}{4} \begin{bmatrix} 0+1+1+1 \\ 0+0+0+1 \\ 0+0+1+0 \end{bmatrix} + 0.5 \times \frac{1}{4} \begin{bmatrix} 0+0+0+1 \\ 0+1+1+1 \\ 1+0+1+1 \end{bmatrix} \\
&= 0.5 \times \frac{1}{4} \begin{bmatrix} 3 \\ 1 \\ 1 \end{bmatrix} + 0.5 \times \frac{1}{4} \begin{bmatrix} 1 \\ 3 \\ 3 \end{bmatrix} \\
&= \frac{1}{2} \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}^T \neq \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T
\end{aligned}$$

样本均值不为 0，不符合最佳变换的条件。此时需将样本均值均值平移至原点，同时样本点同步移动，得到变换后的样本矩阵为：

$$\mathbf{X}' = \mathbf{X} - \mathbf{M} = \frac{1}{2} \begin{bmatrix} -1 & 1 & 1 & 1 \\ -1 & -1 & -1 & 1 \\ -1 & -1 & 1 & -1 \end{bmatrix}$$

$$\mathbf{Y}' = \mathbf{Y} - \mathbf{M} = \frac{1}{2} \begin{bmatrix} -1 & -1 & -1 & 1 \\ -1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 \end{bmatrix}$$

之后对样本矩 $\mathbf{X}'$ 和 $\mathbf{Y}'$ 进行变换。

ii. **计算自相关矩阵**
整体的自相关矩阵为：

$$
\begin{aligned}
\mathbf{R} &= P(\omega_1)\mathrm{E}(\mathbf{X'X'^T}) + P(\omega_2)\mathrm{E}(\mathbf{Y'Y'^T}) \\
&= 0.5 \times \frac{1}{4} \times \frac{1}{4} \begin{bmatrix} -1 & 1 & 1 & 1 \\ -1 & -1 & -1 & 1 \\ -1 & -1 & 1 & -1 \end{bmatrix} \begin{bmatrix} -1 & -1 & -1 \\ 1 & -1 & -1 \\ 1 & -1 & 1 \\ 1 & 1 & -1 \end{bmatrix} + 0.5 \times \frac{1}{4} \times \frac{1}{4} \begin{bmatrix} -1 & -1 & -1 & 1 \\ -1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 \end{bmatrix} \begin{bmatrix} -1 & -1 & 1 \\ -1 & 1 & -1 \\ -1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \\
&= \frac{1}{16} \begin{bmatrix} 4 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 4 \end{bmatrix} = \frac{1}{4} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}
\end{aligned}
$$

iii. **计算特征值和特征向量**
自相关矩阵 $\mathbf{R}$ 的特征方程为：

$$
|\lambda \mathbf{E} - \mathbf{R}| = \begin{vmatrix} \lambda - \frac{1}{4} & 0 & 0 \\ 0 & \lambda - \frac{1}{4} & 0 \\ 0 & 0 & \lambda - \frac{1}{4} \end{vmatrix} = (\lambda - \frac{1}{4})^3 = 0
$$

解得 $\mathbf{R}$ 的三重特征值为 $\lambda = \frac{1}{4}$。此时求特征向量的齐次线性方程组为：

$$
\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \mathbf{O}
$$

它的系数矩阵是零矩阵，所以任意三个线性无关的向量都是它的基础解系。在这里取特征向量为：

$$
\boldsymbol{\phi_1} = [1,0,0]^T \ \boldsymbol{\phi_2} = [0,1,0]^T \ \boldsymbol{\phi_3} = [0,0,1]^T
$$

iv. **降维**

- **降到二维**
  取 $\boldsymbol{\phi_1}$ 和 $\boldsymbol{\phi_2}$ 作为变换矩阵，即

$$
\boldsymbol{\Phi} = \begin{bmatrix} \boldsymbol{\phi_1} & \boldsymbol{\phi_2} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}
$$

  降维后的样本矩阵为：

$$
\mathbf{A_{X'}} = \boldsymbol{\Phi}^T \mathbf{X'} = \frac{1}{2} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} -1 & 1 & 1 & 1 \\ -1 & -1 & -1 & 1 \\ -1 & -1 & 1 & -1 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} -1 & 1 & 1 & 1 \\ -1 & -1 & -1 & 1 \end{bmatrix}
$$

$$
\mathbf{A_{Y'}} = \boldsymbol{\Phi}^T \mathbf{Y'} = \frac{1}{2} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} -1 & -1 & -1 & 1 \\ -1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} -1 & -1 & -1 & 1 \\ -1 & 1 & 1 & 1 \end{bmatrix}
$$

  降维后的样本位置为：

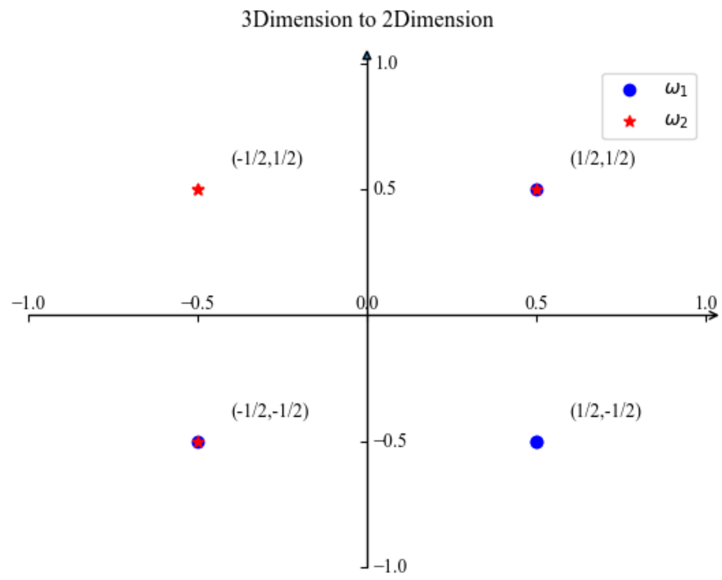

图 1: 降至二维的样本位置

- **降到一维**
  取 ϕ_1 作为变换矩阵，即

$$\mathbf{\Phi'} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}^T$$

降维后的样本矩阵为:

$$\mathbf{B_{X'}} = \mathbf{\Phi'}^T\mathbf{X'} = \frac{1}{2}\begin{bmatrix} 1 & 0 & 0 \end{bmatrix}\begin{bmatrix} -1 & 1 & 1 & 1 \\ -1 & -1 & -1 & 1 \\ -1 & -1 & 1 & -1 \end{bmatrix} = \frac{1}{2}\begin{bmatrix} -1 & 1 & 1 & 1 \end{bmatrix}$$

$$\mathbf{B_{Y'}} = \mathbf{\Phi'}^T\mathbf{Y'} = \frac{1}{2}\begin{bmatrix} 1 & 0 & 0 \end{bmatrix}\begin{bmatrix} -1 & -1 & -1 & 1 \\ -1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 \end{bmatrix} = \frac{1}{2}\begin{bmatrix} -1 & -1 & -1 & 1 \end{bmatrix}$$

降维后的样本位置为:

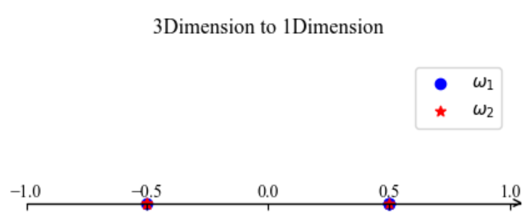

图 2: 降至一维的样本位置

AppendixI：绘图代码

```
import matplotlib.pyplot as plt
import numpy as np
import mpl_toolkits.axisartist as axisartist

# 字体设置
plt.rc('font',family='Times New Roman')

# 降至二维
X1 = np.array([[-0.5,-0.5],[0.5,-0.5],[0.5,-0.5],[0.5,0.5]])
Y1 = np.array([[-0.5,-0.5],[-0.5,0.5],[-0.5,0.5],[0.5,0.5]])

fig = plt.figure()

# 坐标轴设置
ax = axisartist.Subplot(fig, 111)
#将绘图区对象添加到画布中
fig.add_axes(ax)
#通过set_visible方法设置绘图区所有坐标轴隐藏
ax.axis[:].set_visible(False)
#ax.new_floating_axis代表添加新的坐标轴
ax.axis["x"] = ax.new_floating_axis(0,0)
#给x坐标轴加上箭头
ax.axis["x"].set_axisline_style("->", size = 1.0)
#添加y坐标轴，且加上箭头
ax.axis["y"] = ax.new_floating_axis(1,0)
ax.axis["y"].set_axisline_style("-|>", size = 1.0)
#设置x、y轴上刻度显示方向
ax.axis["x"].set_axis_direction("top")
ax.axis["y"].set_axis_direction("right")

# 散点图
plt.scatter(X1[:,0],X1[:,1],marker='o',c='b',label='$\omega_1$')
plt.scatter(Y1[:,0],Y1[:,1],marker='*',c='r',label='$\omega_2$')
# 坐标轴刻度
plt.xlim(-1,1)
plt.ylim(-1,1)
plt.xticks([-1,-0.5,0,0.5,1])
plt.yticks([-1,-0.5,0.5,1])
# 坐标标注
plt.annotate('(1/2,1/2)',xy=(0.5,0.5),xytext=(0.5+0.1,0.5+0.1))
plt.annotate('(1/2,-1/2)',xy=(0.5,-0.5),xytext=(0.5+0.1,-0.5+0.1))
plt.annotate('(-1/2,-1/2)',xy=(-0.5,-0.5),xytext=(-0.5+0.1,-0.5+0.1))
plt.annotate('(-1/2,1/2)',xy=(-0.5,0.5),xytext=(-0.5+0.1,0.5+0.1))

plt.legend()
```

```
plt.title('3Dimension to 2Dimension',y=1.05)
plt.savefig('2dim.png')
plt.close(fig)

# ----------------
# 降至一维
X2 = np.array([-0.5,0.5,0.5,0.5])
Y2 = np.array([-0.5,-0.5,-0.5,0.5])

fig = plt.figure(figsize=(5,3))
ax = axisartist.Subplot(fig, 111)
#将绘图区对象添加到画布中
fig.add_axes(ax)
#通过set_visible方法设置绘图区所有坐标轴隐藏
ax.axis[:].set_visible(False)
#ax.new_floating_axis代表添加新的坐标轴
ax.axis["x"] = ax.new_floating_axis(0,0)
#给x坐标轴加上箭头
ax.axis["x"].set_axisline_style("->", size = 1.0)
#设置x、y轴上刻度显示方向
ax.axis["x"].set_axis_direction("top")

plt.scatter(X2,[0]*len(X2),marker='o',c='b',label='$\omega_1$')
plt.scatter(Y2,[0]*len(Y2),marker='*',c='r',label='$\omega_2$')
plt.xlim(-1,1)
plt.xticks([-1,-0.5,0,0.5,1])
plt.annotate('1/2',xy=(0.5,0),xytext=(0.5+0.1,0.5))
plt.annotate('-1/2',xy=(-0.5,0),xytext=(-0.5+0.1,0.5))
plt.legend()
plt.title('3Dimension to 1Dimension',y=1.05)
plt.savefig('1dim.png')
plt.close(fig)
```

AppendixII：计算代码

```
import numpy as np

# data:
w1 = np.array([[0,0,0],[1,0,0],[1,0,1],[1,1,0]])
w2 = np.array([[0,0,1],[0,1,0],[0,1,1],[1,1,1]])
w1 = w1.T
w2 = w2.T
d = w1.shape[0] # dim
n = w1.shape[1] # samples number
p1 = 0.5
p2 = 0.5

# E(x):
w1_mean = np.reshape(np.mean(w1,axis=1),(d,1))
w2_mean = np.reshape(np.mean(w2,axis=1),(d,1))
Ex_all = p1*w1_mean+p2*w2_mean

# offset:
w1_off = w1-Ex_all
w2_off = w2-Ex_all

# 自相关矩阵:
R1 = 1/n*(np.dot(w1_off,w1_off.T))
R2 = 1/n*(np.dot(w2_off,w2_off.T))
R = p1*R1+p2*R2

# 特征值:
eigenvalue, featurevector = np.linalg.eig(R)
print("特征值: ",end=' ')
print(eigenvalue)
print("特征向量: ")
print(featurevector)

# 变换:
d2x = np.dot(featurevector[:,:2].T,w1_off)
d2y = np.dot(featurevector[:,:2].T,w2_off)
print("降至二维: ")
print("第一类")
print(d2x)
print("第二类")
print(d2y)
d1x = np.dot(featurevector[:,0].T,w1_off)
d1y = np.dot(featurevector[:,0].T,w2_off)
print("降至一维: ")
print("第一类")
```

```
print(d1x)
print("第二类")
print(d1y)

# ————————
# output:
# 特征值:  [0.25 0.25 0.25]
# 特征向量:
# [[1.  0.  0.]
#  [0.  1.  0.]
#  [0.  0.  1.]]
# 降至二维:
# 第一类
# [[-0.5  0.5  0.5  0.5]
#  [-0.5 -0.5 -0.5  0.5]]
# 第二类
# [[-0.5 -0.5 -0.5  0.5]
#  [-0.5  0.5  0.5  0.5]]
# 降至一维:
# 第一类
# [-0.5  0.5  0.5  0.5]
# 第二类
# [-0.5 -0.5 -0.5  0.5]
```