

大数据管理方法与应用作业 1

大数据 81 郭英明

1. 求下列矩阵的一个完全奇异值分解，并写出相应的紧奇异值分解:

$$A = \begin{bmatrix} 1 & -1 \\ -2 & 2 \\ 2 & -2 \end{bmatrix}$$
$$AA^T = \begin{bmatrix} 1 & -2 & 2 \\ -1 & 2 & -2 \\ 2 & -2 & 2 \end{bmatrix} \begin{bmatrix} 1 & -1 \\ -2 & 2 \\ 2 & -2 \end{bmatrix} = \begin{bmatrix} 9 & -9 \\ -9 & 9 \\ 4 & -4 \end{bmatrix}$$
$$A^T A = \begin{bmatrix} 1 & -1 \\ -2 & 2 \\ 2 & -2 \end{bmatrix} \begin{bmatrix} 1 & -2 & 2 \\ -1 & 2 & -2 \end{bmatrix} = \begin{bmatrix} 2 & -4 & 4 \\ -4 & 8 & -8 \\ 4 & -8 & 8 \end{bmatrix}$$

对于 $A^T A$

$$\det(A^T A - \lambda E) = 0$$
$$\Rightarrow \lambda_1 = 18 \quad v_1 = \begin{bmatrix} \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} \\ -\frac{\sqrt{2}}{2} \end{bmatrix} \quad \lambda_2 = 0 \quad v_2 = \begin{bmatrix} \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} \end{bmatrix}$$

同理, 对于 AA^T

$$\lambda_1 = 18 \quad u_1 = \left[\frac{1}{3}, -\frac{2}{3}, \frac{2}{3} \right]^T, \quad \lambda_2 = 0 \quad u_2 = \left[\frac{4}{3\sqrt{2}}, \frac{1}{3\sqrt{2}}, -\frac{1}{3\sqrt{2}} \right]^T$$
$$\lambda_3 = 0, \quad u_3 = \left[0, \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}} \right]^T$$
$$\therefore A = \begin{bmatrix} \frac{18}{3} & \frac{4}{3\sqrt{2}} & 0 \\ -\frac{2}{3} & \frac{1}{3\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{2}{3} & -\frac{1}{3\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 3\sqrt{2} & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \end{bmatrix}$$

紧奇异值分解:

$$A = \begin{bmatrix} \frac{1}{3} \\ -\frac{2}{3} \\ \frac{2}{3} \end{bmatrix} [3\sqrt{2}] \begin{bmatrix} \frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2} \end{bmatrix}$$

2. 利用奇异值分解压缩下列图片，对比保留不同比例（1%，2%，5%，10%，20%，30%）特征值时重构的图片与原图片的异同。

代码：

```
import numpy as np
from PIL import Image

def imgCompress(channel, percent):
    U, sigma, V_T = np.linalg.svd(channel)
    m = U.shape[0]
    n = V_T.shape[0]
    reChannel = np.zeros((m, n))

    for k in range(len(sigma)):
        reChannel = reChannel + sigma[k] * np.dot(U[:, k].reshape(m, 1),
            V_T[k, :].reshape(1, n))
        if float(k) / len(sigma) > percent:
            reChannel[reChannel < 0] = 0
            reChannel[reChannel > 255] = 255
            break

    return np rint(reChannel).astype("uint8")

oriImage = Image.open(r'D:\桌面\奇异值.jpg', 'r')
imgArray = np.array(oriImage)

R = imgArray[:, :, 0]
G = imgArray[:, :, 1]
B = imgArray[:, :, 2]
# A = imgArray[:, :, 3]

for p in [0.01, 0.02, 0.05, 0.1, 0.2, 0.3]:
    reR = imgCompress(R, p)
    reG = imgCompress(G, p)
    reB = imgCompress(B, p)
    # reA = imgCompress(A, p)
    reI = np.stack((reR, reG, reB), 2)

    Image.fromarray(reI).save("{}".format(p) + "img.png")
```

图片结果：

图片顺序依次为取特征值 1%， 2%， 5%， 10%， 20%， 30%





