SVD-矩阵奇异值分解 —— 原理与几何意义

1 简介

SVD 全称：Singular Value Decomposition。SVD 是一种提取信息的强大工具，它提供了一种非常便捷的矩阵分解方式，能够发现数据中十分有意思的潜在模式。

主要应用领域包括：

- 隐性语义分析 (Latent Semantic Analysis, LSA) 或隐性语义索引 (Latent Semantic Indexing, LSI)；
- 推荐系统 (Recommender system)，可以说是最有价值的应用点；
- 矩阵形式数据（主要是图像数据）的压缩。

2 线性变换

在做 SVD 推导之前，先了解一下线性变换，以 2*2 的线性变换矩阵为例，先看简单的对角矩阵：

$$M = \begin{bmatrix} 3 & 0 \\ 0 & 1 \end{bmatrix}$$

从集合上讲， M 是将二维平面上的点（x，y） 经过线性变换到另一个点的变换矩阵，如下所示：

$$\begin{bmatrix} 3 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 3x \\ y \end{bmatrix}$$

该变换的几何效果是，变换后的平面沿着x水平方向进行了3倍拉伸，垂直方向没有发生变化。

3 SVD 推导

该部分的推导从几何层面上去理解二维的SVD，总体的思想是：借助 SVD 可以将一个相互垂直的网格 (orthogonal grid) 变换到另外一个互相垂直的网格。

可以通过二维空间中的向量来描述这件事情。

首先，选择两个互相正交的单位向量 **v1**和 **v2**（也可称为一组正交基）。

M 是一个变换矩阵。

向量 **Mv1 , Mv2** 也是一组正交向量（也就是 **v1**和 **v2** 经过 **M** 变换得到的）。

u1， u2 分别是 **Mv1, Mv2** 的单位向量（即另一组正交基），且有：

$$Mv_1 = \sigma_1 u_1$$

$$Mv_2 = \sigma_2 u_2$$

则，**σ1，σ2** 分别为 **Mv1，Mv2** 的模（也称为**M**的奇异值）。

设任意向量x，有：

$$x = (v_1 \cdot x)v_1 + (v_2 \cdot x)v_2$$

$$x = (v_1 \cdot x)v_1 + (v_2 \cdot x)v_2$$

例如，当 $x = \begin{bmatrix} 3 \\ 2 \end{bmatrix}$ 时，$x = \left(\begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} 3 & 2 \end{bmatrix}\right) \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \left(\begin{bmatrix} 0 \\ 1 \end{bmatrix} \begin{bmatrix} 3 & 2 \end{bmatrix}\right) \begin{bmatrix} 0 \\ 1 \end{bmatrix}$.

那么，可得：

$$Mx = (v_1 \cdot x)Mv_1 + (v_2 \cdot x)Mv_2$$

$$Mx = (v_1 \cdot x)\sigma_1 u_1 + (v_2 \cdot x)\sigma_2 u_2$$

根据线代知识，向量的内积可用向量的转置来表示：

$v_1 \cdot x = v^T x$ ，则有：

$$Mx = v_1^T x \sigma_1 u_1 + v_1^T x \sigma_2 u_2$$

两边去掉 x ，得：

$$M = u_1 \sigma_1 v_1^T + u_2 \sigma_2 v_2^T$$

将下标相同的向量合并起来，则该式可通用地表达为：

$$M = U \Sigma V^T$$

至此，SVD 使用几何意义的形式推导完毕，其中：

- U 矩阵的列向量分别是 u_1, u_2 ；
- Σ 是一个对角矩阵，形如：$\begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix}$ ；
- V 矩阵的列向量分别是 v_1, v_2 。

关于 SVD 的一些重要的结论性总结：

- 任意的矩阵**M**是可以分解成三个矩阵；
- **V**表示了原始域的标准正交基；
- **U**表示经过**M**变换后的新标准正交基；
- ∑表示了**V**中的向量与**U**中相对应向量之间的比例（伸缩）关系；
- **∑**中的每个**σ**会按从大到小排好顺序，值越大代表该维度重要性越高；
- 在利用 SVD 做数据信息提取或压缩时，往往依据一些启发式策略，如直接设定只提取∑ 中的前 **k** 项，或者另一种较常用的做法是保留矩阵中一定百分比的能量信息，一般可设定为 90%，能量信息比例的计算可先求得所有奇异值平方总和，然后将奇异值的平方依次累加到总值的 90% 为止，形如：

$$k = \arg\min_k \frac{\sum_{i=0}^{k} \sigma_i^2}{\sum_{i=0}^{N} \sigma_i^2} \geqslant 0.9 \ .$$

```
# -*- coding: utf-8 -*-

import numpy as np
import numpy.linalg as la
import matplotlib.pyplot as plt
from sklearn import datasets
from skimage import io

def getImgAsMat(index):
    ds = datasets.fetch_olivetti_faces()
    return np.mat(ds.images[index])
```

```
def getImgAsMatFromFile(filename):
    img = io.imread(filename, as_grey=True)
    return np.mat(img)


 def plotImg(imgMat):
    plt.imshow(imgMat, cmap=plt.cm.gray)
    plt.show()


def recoverBySVD(imgMat, k):
    # singular value decomposition
    U, s, V = la.svd(imgMat)
    # choose top k important singular values (or eigens)
    Uk = U[:, 0:k]
    Sk = np.diag(s[0:k])
    Vk = V[0:k, :]
    # recover the image
    imgMat_new = Uk * Sk * Vk
    return imgMat_new


# -------------------- main --------------------- #
#A = getImgAsMat(0)
#plotImg(A)
#A_new = recoverBySVD(A, 20)
#plotImg(A_new)


A = getImgAsMatFromFile('D:/pic.jpg')
plotImg(A)
A_new = recoverBySVD(A, 30)
plotImg(A_new)
```

5 参考资料

1. We Recommend a Singular Value Decomposition (AMS 上一篇非常经典的关于SVD几何意义的介绍)
2. 奇异值分解 (SVD) 的几何意义中文翻译

作者：**张磊** 机器学习爱好者

知乎：https://zhuanlan.zhihu.com/c_184412713