**开发记录-2020-03-12-20点01分**

开发内容：依据老师上午将的内容，对之前自己完成的SVD代码进行debug，解决其存在的问题。

解决思想：利用算法中的的特性，对U和V的特征列向量进行逐个检查。如果出现相差一个负号的现象，就将一个取负，使之成立。

具体实现代码如下：

**function** **[**sortedEigVecB**,**singularValueMatrix**,**sortedEigVecA**]=**mysvd**(**matrixToCompute**)**

%% 形成左右矩阵。其中A对应右侧V，B对应左侧U

matrixA **=** matrixToCompute**.'\***matrixToCompute**;**

matrixB **=** matrixToCompute**\***matrixToCompute**.';**

%% 记待求矩阵为A，求解A'A和AA'

**[**eigVecA**,**eigValueA**]=**eig**(**matrixA**);**

**[**eigVecB**,**eigValueB**]=**eig**(**matrixB**);**

%% 对特征值和特征向量做排序

% 排序算法来自 https://ww2.mathworks.cn/help/matlab/ref/eig.html

**[**eigValueElementA**,**sortedIndexA**]** **=** sort**(**diag**(**eigValueA**),**'descend'**);**

sortedEigValA**=**eigValueA**(**sortedIndexA**,**sortedIndexA**);**

sortedEigVecA **=** eigVecA**(:,**sortedIndexA**);**

%% 对 matrixB 的特征值和特征向量做同样的排序操作

**[**eigValueElementB**,**sortedIndexB**]** **=** sort**(**diag**(**eigValueB**),**'descend'**);**

sortedEigValB**=**eigValueB**(**sortedIndexB**,**sortedIndexB**);**

sortedEigVecB **=** eigVecB**(:,**sortedIndexB**);**

%% 截取有效部分

**if** length**(**sortedEigValB**)>**length**(**sortedEigValA**)**

singularValueMatrix**=**sqrt**(**sortedEigValB**(**1**:**size**(**matrixToCompute**,**1**),**1**:**size**(**matrixToCompute**,**2**)));**

**else**

singularValueMatrix**=**sqrt**(**sortedEigValA**(**1**:**size**(**matrixToCompute**,**1**),**1**:**size**(**matrixToCompute**,**2**)));**

**end**

%% 检查左右U,V是否能够满足中间的奇异值分解主对角线元素非负。不满足则修改左侧使之满足

**for** i**=** 1**:** length**(**find**(**eigValueA**>**1e-6**))**

**if** max**(**abs**((**1**/**sqrt**(**sortedEigValA**(**i**,**i**))\***matrixToCompute**\***sortedEigVecA**(:,**i**)** **+** sortedEigVecB**(:,**i**))))<**1e-6

sortedEigVecB**(:,**i**)=-**sortedEigVecB**(:,**i**);**

**end**

**end**

%% 验证SVD是否成功

sortedEigVecB**\***singularValueMatrix**\***sortedEigVecA**.';**

**end**

为了检验上述代码的正确性，这里选用了MATLAB自带的SVD命令作为参考的标准结果，和自己完成的mysvd进行对比检验。如果相差很小，如，则认为该误差是由于计算机自身的计算误差引起的。

在测试文件中，通过反复测试不同尺寸和数值的矩阵，来验证算法的有效性。

以下是实现代码：

clc**;**clear**;**close all**;**

%% 初始化保存运算误差的两个变量

errS**=[];**

err**=[];**

%% 循环检验算法的有效性

**for** i**=**1**:**1000

a**=**fix**(**200**\***rand**)+**2**;**%随机行数

b**=**fix**(**200**\***rand**)+**2**;**%随机列数

mat**=**rand**(**a**,**b**);**%随机矩阵

**[**stdU**,**stdS**,**stdV**]=**svd**(**mat**);**%标准参考

**[**myU**,**myS**,**myV**]=**mysvd**(**mat**);**%自己的运算结果

errS**=[**errS**,**max**(**abs**(**myS**-**stdS**))];**%奇异值的差

err**=[**err**,**max**(**max**(**abs**(**myU**\***myS**\***myV**'-**mat**)))];**%还原之后相比于原矩阵的差

**end**

%% 输出最大值

max**(**errS**)**

max**(**err**)**

通过计算，可以发现运算结果中，奇异值分解最大误差为，还原成原矩阵的最大误差为。

注：该验证基于随机产生的矩阵，以上结果具体数值不具有可重复性，但是其数量级具有可重复性。