矩阵代数与应用

上海大学计算机工程与科学学院

2023年12月

第三、四章 矩阵分解

3.0 矩阵分解基础

3.1 特征值分解

3.5 特征分析的应用

4.1 数值稳定性与条件数

4.2 奇异值分解

4.4 奇异值分解的工程案例

3.0 矩阵分解基础

- 初等变换
- 三角分解
- Schur分解
- QR分解
- 其他矩阵分解

"分解"是"分析"的重要手段之一！

◆初等变换

$$A_{m\times n}x_{n\times 1}=b_{m\times 1},$$

若A为正方矩阵（m=n）且非奇异，则

$$x=A^{-1}b.$$

（1）适定方程（α=β）

（2）欠定方程（α<β）

（3）超定方程（α>β，此时α指方程的个数？？？？）

- 独立方程个数：α，未知数个数：β
- 有解（唯一解，无穷多解），无解
- 矛盾方程（无解，广义解）

◆初等变换

✓方程组的初等行变换

✓矩阵的初等行变换

$$A_{m\times n}=P_{m\times m}\begin{pmatrix} I_{r\times r} & O_{r\times(n-r)} \\ O_{(m-r)\times r} & O_{(m-r)\times(n-r)} \end{pmatrix}Q_{n\times n},$$

$r=rank(A), Q, P$是非奇异的.

命题 1：若A为一m×n矩阵，则对A施行一次初等行（列）变换，相当于对A左（右）乘相应的m(n)阶初等矩阵.

命题 2：若矩阵A经有限次初等行（列）变换变成矩阵B，则A的行（列）向量组与B的**行（列）向量组**等价，而A的任意k个**列（行）向量**与B中对应的k个列（行）向量有相同的线性相关性. (行操作不影响列向量的线性相关性质)

命题 3：设A为可逆方阵，则存在有限个初等方阵$E_1,E_2,\cdots,E_p$，使 $A=E_1E_2\cdots E_p$.

推论1：m×n矩阵 $A\cong B$的充分必要条件是存在m阶可逆方阵P及n阶可逆方阵Q，使PAQ=B.（矩阵等价：初等变换下的矩阵相互等价）

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix} \quad \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

◆ 初等变换

定理1：任何一个矩阵 $A_{m\times n}$ 都与一个标准简约阶梯型矩阵等价，与一个简约阶梯型矩阵行等价(A和B行等价,就是说A经过若干次初等行变换可以变成B).

◆ 主元位置、主元列

主元(pivot element)，指在消去过程中起主导作用的变元.

m×n矩阵方程Ax=b的求解.

步骤 1：构造增广矩阵B=[A, b].

步骤 2：通过初等行变换，将增广矩阵B化成简约阶梯型矩阵，它与原增广矩阵等价.

步骤 3：从简化的矩阵得到对应的线性方程组，它与原线性方程组等价.

步骤 4：求新线性方程组的通解(general solution).

◆矩阵的等价标准型（canonical form）

等价：如果矩阵B可以由A经过一系列初等变换得到 那么矩阵A与B是等价的.

等价标准型：经过多次变换以后，得到一种左上角是一个单位矩阵，其余元素都为0的矩阵，此矩阵称为该矩阵的等价标准型.

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

◆ Jordan标准型

Jordan形矩阵：由Jordan块按对角排列组成的矩阵.

Jordan块：主对角线为矩阵的特征值，主对角线上方元素为1的矩阵.

定理：任意方阵A，都与一个Jordan形矩阵J相似，A~J.Jordan形矩阵J称为矩阵A的Jordan标准型.即，存在可逆矩阵P，使得 $P^{-1}AP = J$ ，其中

$$J = \begin{pmatrix} J_1 & & & \\ & J_2 & & \\ & & \ddots & \\ & & & J_m \end{pmatrix}$$

$$\begin{pmatrix} 3 & 1 & 0 \\ 0 & 3 & 1 \\ 0 & 0 & 3 \end{pmatrix}, \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & -1 \end{pmatrix}, \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 3 & 1 \\ 0 & 0 & 0 & 3 \end{pmatrix}$$

```
>> A=rand(5);
 B=jordan(A)
```

要使用 'jordan'，则必须授权、安装并启用了以下产品:
 Symbolic Math Toolbox

◆三角分解

- **Gauss消去法**

基本思想：利用初等行变换化增广矩阵[A b]为上阶梯形.

- **三角(LU)分解**

对于<u>任意矩阵A</u>，都存在分解：PA=LU. 其中，P为行置换阵，L为对角线上元素为1 的下三角阵，U是上三角阵.

MATLAB中的LU分解的函数为: [L,U,P]=lu(A).

行置换阵：单位阵通过几次行互换得到的矩阵.

1.0000	0	0	0
0.7106	1.0000	0	0
0.6702	0.2823	1.0000	0
0.6281	0.5282	0.4764	1.0000

0.8759	0.2077	0.8443	0.2277	0.4302	0.4389
0	0.3233	-0.3741	0.1493	0.5992	-0.0538
0	0	-0.2895	0.7287	0.5223	0.1298
0	0	0	-0.1333	-0.6507	-0.1979

1	0	0	0
0	0	1	0
0	0	0	1
0	1	0	0

```
A=rand(4,6);
[L,U,P]=lu(A);
disp(L);
disp(U);
disp(P);
```

◆ Cholesky分解(乔里斯基)

命题4：设A为n阶实对称正定矩阵，则存在非奇异下三角矩阵L，使

$$A = LL^T,$$

若限定L的对角元素为正，则这种分解是唯一的.

◆ C.Schur分解（舒尔）

定理2：设$A \in C^{n\times n}$的特征值为$\lambda_1, \lambda_2, \cdots, \lambda_n$，则存在酉矩阵P，使得

$$P^{-1}AP = P^H AP = \begin{bmatrix} \lambda_1 & * & \cdots & * \\ & \lambda_2 & \ddots & \vdots \\ & & \ddots & \\ & & & \lambda_n \end{bmatrix}$$

Schur分解：$A = PRP^H$.

MATLAB函数chol(X)用于对矩阵X进行Cholesky分解，其调用格式为：

R=chol(X);

产生一个上三角阵R，使R'R=X. 若X为非对称正定，则输出一个出错信息.

```
A=rand(4,4);
B=A'*A;
disp(B);
R=chol(B);
disp(R);
D=R'*R;
disp(D);
```

0.6456	0.4698	1.0377	0.7622
0.4698	0.6356	0.8850	0.6597
1.0377	0.8850	2.0964	1.1292
0.7622	0.6597	1.1292	1.0043

0.8035	0.5848	1.2915	0.9486
0	0.5419	0.2395	0.1938
0	0	0.6092	-0.2335
0	0	0	0.1112

D =

0.6456	0.4698	1.0377	0.7622
0.4698	0.6356	0.8850	0.6597
1.0377	0.8850	2.0964	1.1292
0.7622	0.6597	1.1292	1.0043

命题5：**<u>任意一个n阶方阵X</u>**可以分解为X=URU'，其中U为酉矩阵，R为上三角schur矩阵且其主对角线上的元素为X的特征值.（一般矩阵的Schur分解）

（Matlab函数 [U,R]=schur(X，'complex')）

X =

0.8147	0.6324	0.9575	0.9572
0.9058	0.0975	0.9649	0.4854
0.1270	0.2785	0.1576	0.8003
0.9134	0.5469	0.9706	0.1419

U =

-0.6621	-0.5327	0.5206	-0.0825
-0.4819	0.1301	-0.5815	-0.6424
-0.2766	0.8273	0.4848	-0.0637
-0.5029	0.1217	-0.3947	0.7593

R =

2.4021	-0.8133	-0.6225	-0.1304
0	-0.0346	-0.1940	0.2110
0	0	-0.7158	0.3496
0	0	0	-0.4400

Y =

0.8147	0.6324	0.9575	0.9572
0.9058	0.0975	0.9649	0.4854
0.1270	0.2785	0.1576	0.8003
0.9134	0.5469	0.9706	0.1419

```
X=rand(4,4);
[U,R]=schur(X,'complex');
Y=U*R*U';
disp(R);
```

```
X=rand(4,4);
 [U R]=schur(X,'real')
Y=U*R*U'
disp(R);
```

对Matlab函数schur，最好显式地给出是“实数形式分解”还是“复数形式分解”.不然，就使用默认方式.以上命题是针对“复数形式分解”的.

命题5*：任意一个n阶**实方阵X**可以分解为X=URU'，其中U为正交矩阵，R为实拟上三角矩阵. （Matlab函数 [U,R]=schur(X,'real')） （实Schur分解）

```
X=randn(4,4);
[U1,R1]=schur(X,'complex');
Y1=U1*R1*U1';
[U2,R2]=schur(X,'real');
disp(R1);
disp(R2);
S =0.5*[1- i    1-i
          1+i   -1-i ];
SS =0.5*[1+i   1- i
          1+i  -1+i ];
a=1.2; b=3.4;
A=[a b
   -b a];
B=SS*A*S;
disp(B);
D=SS*B*S;
disp(D);
E=S*B*SS;
disp(E);
```

$$R=\begin{bmatrix} R_1 & * & \cdots & * \\ & R_2 & \ddots & \vdots \\ & & \ddots & \\ & & & R_m \end{bmatrix}$$

R为拟上三角阵：其中R_i为实数，或两行两列具有一对共轭特征根的实矩阵.

```
1.1220 + 0.0000i   0.5579 + 0.1212i   0.0820 + 0.8244i   0.0434 + 0.0000i
0.0000 + 0.0000i  -1.0213 + 0.9275i  -0.7429 + 0.0000i   0.5927 + 0.9171i
0.0000 + 0.0000i   0.0000 + 0.0000i  -1.0213 - 0.9275i  -0.6207 - 0.8757i
0.0000 + 0.0000i   0.0000 + 0.0000i   0.0000 + 0.0000i  -0.8613 + 0.0000i

 1.1220    0.1463   -0.9955    0.0434
      0   -1.0213    1.3706   -1.1074
      0   -0.6277   -1.0213   -1.0574
      0         0         0   -0.8613

1.2000 + 3.4000i   0.0000 + 0.0000i
0.0000 + 0.0000i   1.2000 - 3.4000i

1.2000 + 0.0000i   0.0000 + 3.4000i
0.0000 + 3.4000i   1.2000 + 0.0000i
```

行（列）向量两两正交.

$$\frac{1}{2}\begin{pmatrix} 1 & -1 & 1 & 1 \\ 1 & 1 & -1 & 1 \end{pmatrix}$$

◆ QR分解

定理3（QR分解）：设实(复)矩阵 $A_{m\times n}$ 的n个列线性无关(n≤m)，则存在半正交矩阵 $Q_{m\times n}$ 和实(复)非奇异上三角矩阵 $R_{n\times n}$，使得

$$A = QR.$$

此式称为矩阵A的QR分解；且除去相差一个对角线元素的绝对值全等于1的对角矩阵因子外，该分解是唯一的.

MATLAB的函数qr可用于对矩阵进行QR分解，其调用格式为:

[Q,R]=qr(X)：产生一个正交矩阵Q和一个上三角矩阵R，使之满足X=QR.

[Q,R,E]=qr(X)：产生一个正交矩阵Q和一个上三角矩阵R，以及一个列置换矩阵E，使之满足XE=QR.

Matlab的分解格式与以上定理之间的关系:

$$Q_{m\times m}R_{m\times n} = (Q_{m\times n}, Q_{m\times(m-n)})\begin{pmatrix} R_{n\times n} \\ O_{(m-n)\times n} \end{pmatrix} = Q_{m\times n}R_{n\times n}$$

定理3*（QR分解）：设实(复)矩阵 $A_{m\times n}$ 的n个列线性无关(n≤m)，则存在正交矩阵$Q_{m\times m}$和实(复)非奇异上三角矩阵 $R_{m\times n}$，使得

$$A = QR.$$

称为矩阵A的QR分解；且除去相差**一个对角线元素的绝对值全等于1的对角矩阵因子**外，该分解是唯一的.

$$A = Q_{m\times m}R_{m\times n} = (Q_{m\times n}, Q_{m\times(m-n)})\begin{pmatrix} R_{n\times n} \\ O_{(m-n)\times n} \end{pmatrix} = Q_{m\times n}R_{n\times n}$$

一个对角线元素的绝对值全等于1的对角矩阵:

$$P = \begin{pmatrix} \alpha_1 & 0 & \cdots & 0 \\ 0 & \alpha_2 & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & \alpha_m \end{pmatrix}, |\alpha_i| = 1.$$

$$A = QPR.$$

```
X=rand(4,4)
[Q,R]=qr(X);
Q
R
```

```
X =
    0.2425    0.8972    0.4561    0.2973
    0.0538    0.1967    0.1017    0.0620
    0.4417    0.0934    0.9954    0.2982
    0.0133    0.3074    0.3321    0.0464
Q =
   -0.4783   -0.7892    0.3204    0.2136
   -0.1060   -0.1722    0.0689   -0.9769
   -0.8714    0.4651   -0.1562    0.0016
   -0.0262   -0.3621   -0.9318    0.0010
R =
   -0.5069   -0.5394   -1.1050   -0.4099
         0   -0.8098   -0.0347   -0.1234
         0         0   -0.3118    0.0098
         0         0         0    0.0034
```

```
X=rand(4,3)
[Q,R]=qr(X);
Q
R
```

```
Q =
   -0.2341   -0.1806    0.9317   -0.2110
   -0.2990   -0.9118   -0.2699   -0.0795
   -0.7336    0.3562   -0.2351   -0.5289
   -0.5636    0.0951    0.0621    0.8182
R =
   -0.8398   -1.0699   -1.2805
         0   -0.5603   -0.0849
         0         0    0.6462
         0         0         0
```

```
X=rand(3,4)
[Q,R]=qr(X);
Q
R
```

```
Q =
   -0.6279   -0.2625   -0.7327
   -0.6761   -0.2825    0.6805
   -0.3857    0.9226   -0.0001
R =
   -1.4189   -0.2873   -1.0138   -0.9161
         0    0.1590    0.4588   -0.0035
         0         0   -0.4430    0.4539
```

◆其他矩阵分解

正规矩阵

$$A^H A = AA^H$$

■ 特征值分解: $X = VDV^H$

任意一个**n阶可对角化方阵X**可以分解为XV=VD，其中D为X的特征值对角阵，V为X的特征向量矩阵.

MATLAB调用格式为:

[V,D]=eig(X)

注：特征值分解(Eigen decomposition)，又称谱分解(Spectral decomposition). 对称矩阵和正规矩阵都可进行特征值分解.

■ 奇异值分解(Singular Value Decomposition): $A = USV^H$

任意一个m×n维的矩阵X可以分解为X=USV'，U和V均为正交(酉)矩阵，S为m×n维的对角矩阵，其对角线元素为X的从大到小排序的非负奇异值.

MATLAB调用格式为:

[U,S,V]=svd(X)

U,V为正交阵，S为对角阵（对角线上元素为X的奇异值）

注：**奇异值是矩阵A^{H}A的特征值的算数平方根.**

$$A = QR$$
$$Q^H A = R$$
$$(Q^H A)^H (Q^H A) = R^H R$$
$$Q^H AP = RP = D$$

```
>> X=rand(4,4)
X =
    0.4218    0.6557    0.6787    0.6555
    0.9157    0.0357    0.7577    0.1712
    0.7922    0.8491    0.7431    0.7060
    0.9595    0.9340    0.3922    0.0318
>> [V,D]=eig(X)
V =
   0.4883 + 0.0000i  -0.2208 + 0.2328i  -0.2208 - 0.2328i  -0.4513 + 0.0000i
   0.4126 + 0.0000i   0.6612 + 0.0000i   0.6612 + 0.0000i   0.2778 + 0.0000i
   0.6205 + 0.0000i  -0.1999 + 0.1291i  -0.1999 - 0.1291i   0.6267 + 0.0000i
   0.4542 + 0.0000i  -0.2364 - 0.5894i  -0.2364 + 0.5894i  -0.5713 + 0.0000i
D =
   2.4478 + 0.0000i   0.0000 + 0.0000i   0.0000 + 0.0000i   0.0000 + 0.0000i
   0.0000 + 0.0000i  -0.5604 + 0.3177i   0.0000 + 0.0000i   0.0000 + 0.0000i
   0.0000 + 0.0000i   0.0000 + 0.0000i  -0.5604 - 0.3177i   0.0000 + 0.0000i
   0.0000 + 0.0000i   0.0000 + 0.0000i   0.0000 + 0.0000i  -0.0946 + 0.0000i
```

```
>> X=normrnd(0,1,4,4)
X =
    1.1093   -1.1135    0.3714    0.0326
   -0.8637   -0.0068   -0.2256    0.5525
    0.0774    1.5326    1.1174    1.1006
   -1.2141   -0.7697   -1.0891    1.5442

>> [U,S,V]=svd(X)
U =
   -0.2160   -0.4277   -0.8459   -0.2344
    0.3622    0.1590    0.0806   -0.9149
   -0.1707    0.8898   -0.4203    0.0500
    0.8905    0.0021   -0.3185    0.3249
S =
    2.6209         0         0         0
         0    2.3292         0         0
         0         0    1.3443         0
         0         0         0    0.2492
V =
   -0.6283   -0.2342   -0.4863    0.5602
   -0.2705    0.7888    0.4034    0.3766
   -0.5046    0.3423   -0.3385   -0.7167
    0.5267    0.4536   -0.6973    0.1751
```

正态分布随机数生成函数:

R = normrnd(MU,SIGMA,m,n)

3.1 特征值分解

定义1：对 $n \times n$ 矩阵A，若标量λ和$n \times 1$ 向量u满足

$$Au = \lambda u, u \neq 0$$

则称λ和u分别是矩阵A 的**特征值(eigen-value)**和**特征向量(eigen-vector)**，而二元组 (λ, u) 称为矩阵A 的**特征对(eigen-pair)**.

备注:

（**1**）上述定义并不要求矩阵A 是**Hermitian**矩阵.

（**2**）矩阵值λ 由特征多项式（特征方程）

$$(A - \lambda I)\text{奇异} \Leftrightarrow \det(A - \lambda I) = 0$$

的根决定.

定义2：(1) 称A的特征值具有**代数多重度(algebraic multiplicity)μ**，若λ是特征多项式$\det(A-zI)=0$ 的**μ**重根.

(2) 若特征值λ的代数多重度为**1**，则称该特征值为**单特征值(simple eigenvalue)**. 非单的特征值称为**多重特征值(multiple eigen-value)**. 若矩阵$\boldsymbol{A}$ 存在多重特征值，也称其具有退化特征值.

(3) 称A的特征值λ具有**几何多重度(geometric multiplicity)**γ，若与λ对应的线性无关特征向量 $\boldsymbol{u}$ 的个数为γ .

(4) 矩阵A 称为**减次矩阵(derogatory matrix)**，若至少有一个特征值的几何多重度大于**1**.

(5) λ称为**半单特征值(semi-simple eigen-value)**,若其代数多重度与几何多重度相等. 非半单特征值称为**亏损特征值(defective eigen-value)**.

特征值和特征向量的性质

（1）$n\times n$矩阵A具有$\boldsymbol{n}$个特征值，其中多重特征值按照其多重度计数.

（2）特征值与矩阵奇异性的关系：

①若$\boldsymbol{A}$奇异，则它至少有一个特征值为**0**.

②若$\boldsymbol{A}$非奇异，则它所有的特征值非零.

（3）若λ是$n\times n$矩阵A的特征值，则有

① λ也是A^{T} 的特征值.

② λ^*是A^{H} 的特征值.

③ $\lambda+\sigma^2$ 是$A+\sigma^2 I$ 的特征值.

④若A非奇异$(\lambda\neq 0)$，则λ^{-1}是 A^{-1}的特征值.

⑤ λ^k是矩阵A^k的特征值.

$\det(A-zI)=0$

（4）对角矩阵与三角矩阵的特征值即为其对角线元素.

（5）幂等矩阵$A^2=A$的所有特征值取**0**或者**1**.

（6）特征值与行列式的关系：$\prod_{i=1}^{n}\lambda_i=\det(A)=|A|$

（7）特征值与迹的关系：$\sum_{i=1}^{n}\lambda_i=tr(A)$ $\det(A-zI)=0$

（8）特征值与秩的关系：

①若$n\times n$ 矩阵A有r个非零特征值，则rank(A)≥r.

②若**0**是$n\times n$矩阵A的单特征值，则rank(A)=n-1.

③若$rank(A-\lambda I)\le n-1$，则λ是矩阵A的特征值.

（9）与不同特征值$\lambda_1,\lambda_2,\ldots,\lambda_n$对应的非零特征向量 $u_1,u_2,\ldots,u_n$ 线性无关.

（10）实对称矩阵或**Hermitian**矩阵具有实特征值，且不同特征值对应的特征向量相互正交. $u^H Av=v^H A^H u=v^H Au$

（11）一个实对称矩阵或**Hermitian**矩阵A是正定(或半正定)的，当且仅当其特征值是正(或者非负)的.

(12) 相似矩阵有相同的特征值.

(13) 若A的特征值互不相同，则一定可以找到一个相似矩阵使得 $S^{-1}AS=D$ (对角矩阵)，其对角元素即是矩阵A的特征值.

(14) 若 (λ,u) 为矩阵A的特征对，则

① $(f(\lambda)=\lambda^n+c_1\lambda^{n-1}+\ldots+c_{n-1}\lambda+c_n,u)$ 为矩阵多项式 $f(A)=A^n+c_1A^{n-1}+\ldots+c_{n-1}A+c_nI$ 的特征对.

② (e^{λ},u) 为矩阵指数函数 e^A 的特征对.

(15) 若 (λ_A,u_A) 和 (λ_B,u_B) 分别为矩阵A和B的特征对，则

① $(\lambda_A\lambda_B,u_A\otimes u_B)$ 为Kronecker积的 $A\otimes B$ 特征对. (???)

② $\left(\lambda_A,\begin{bmatrix}u_A\\0\end{bmatrix}\right)$ 与 $\left(\lambda_B,\begin{bmatrix}0\\u_B\end{bmatrix}\right)$ 均为矩阵直和 $A\oplus B$ 的特征对.

$ABu=\lambda u$

$BA(Bu)=\lambda(Bu)$ (???)

(16) 矩阵乘积AB与BA具有相同的非零特征值.

编写程序验证(15)(16)!!!

定义3： $n\times n$实矩阵A若与一对角矩阵相似，则称矩阵A是可对角化的(diagonalizable).

定理4： $n\times n$矩阵A是可对角化的，当且仅当A具有n个线性无关的特征向量.

推论2： 若$n\times n$ 矩阵A具有n个不同的特征值，则A可对角化.

定理5： 若矩阵$A\in C^{n\times n}$的特征值具有代数多重度$m_k, k=1,\dots,p$，并且 $\sum_{k=1}^{p} m_k = n$ ，则A具有n个线性无关的特征向量，当且仅当 $rank(A-\lambda_k I) = n - m_k, k=1,2,\dots,p$. 此时，$AU = U\Lambda$ 中的矩阵U是非奇异的，而且A可对角化为

$$U^{-1}AU = \Lambda$$

一个具有多重特征值、不可对角化的矩阵A，可以相似化简为Jordan标准型（Jordan canonical form)，$U^{-1}AU=J$，其中

$$J=\begin{pmatrix} J_1 & & O \\ & \ddots & \\ O & & J_p \end{pmatrix}, J_k=\begin{pmatrix} \lambda & 1 & & 0 \\ & \lambda & 1 & \\ & & \ddots & 1 \\ 0 & & & \lambda \end{pmatrix} \in C^{L_k \times L_k}$$

命题6：存在P，使得

$$(J_k-\lambda I)^{\mathrm{T}}P=\begin{pmatrix} 0 & 1 & & 0 \\ & 0 & 1 & \\ & & \ddots & 1 \\ 0 & & & 0 \end{pmatrix}^{\mathrm{T}} P=\begin{pmatrix} 0 & 1 & & 0 \\ & 0 & 1 & \\ & & \ddots & 1 \\ 0 & & & 0 \end{pmatrix}=J_k-\lambda I$$

$$(J_k-\lambda I)u=0 \Rightarrow (u^T P^T)(J_k-\lambda I)=0 \Rightarrow (u^T P^T)J_k=\lambda(u^T P^T)$$

特征值不分左右！！！

$\boxed{Au=\lambda u(\text{右特征向量}), uA=\lambda u(\text{左特征向量})}$ $J_k u=\lambda u \Rightarrow (u^T P^T)J_k=\lambda(u^T P^T)$

定义4: $cond(\lambda)=\dfrac{1}{\cos\theta(u,v)}$ 称为特征值的条件数. (???)

定义5: 矩阵$A\in C^{n\times n}$ 的所有特征值的集合称为**矩阵A的谱**，记作$\lambda(A)$.

而$\rho(A)=\max|\lambda|:\lambda\in\lambda(A)$ 称为**矩阵A的谱半径**.

定义6: 对称矩阵 $A\in R^{n\times n}$ 的惯性(Inertia), 定义为三元组

$$In(A)=(i_+(A),i\_(A),i_0(A))$$

其中 $i_+(A),i_(A)$和$i_0(A)$ 分别是A的正、负和零特征值的个数(多种特征值全部计算多重数在内). 另外, $i_+(A)-i\_(A)$ 称为矩阵A的符号差.

备注:

(1) 显然，对称矩阵A的秩由$rank(A)=i_+(A)+i_(A)$ 决定.

（2）给定一个对称矩阵A和一个同维数的非奇异矩阵U，则

$$In(A)=In(UAU^{-1})$$

命题7：设 $A\in C^{n\times n}$可对角化，且 (λ_i, p_i) 和 $(\lambda_i', q_i), i=1,\dots,n$

分别为A和A^H的特征对，则 $(\lambda_i' = \lambda_i^*)$ 且

$$p_i^H q_j = \delta_{ij} = \begin{cases} 1, i=j \\ 0, i\neq j \end{cases}$$

定理7：设 $A\in C^{n\times n}$ 可对角化且(λ_i, p_i)和 $(\lambda_i^*, q_i), i=1,\dots,n$

分别为A和A^H的特征对，则

$$A=\sum_{i=1}^{n}\lambda_i p_i q_i^H$$

$$X = VDV^{-H}$$

备注：$A=\sum_{i=1}^{n}\lambda_i p_i q_i^H$ 称为**矩阵A的谱分解**.

$$A=SDS^{-1}$$
$$S=(p_1,\cdots,p_i,\cdots,p_n),$$
$$S^{-1}=(q_1,\cdots,q_i,\cdots,q_n)^H$$
$$D=diag(\lambda_1,\cdots,\lambda_i,\cdots,\lambda_n)$$
$$\Rightarrow A=\sum_{i=1}^{n}\lambda_i p_i q_i^H$$

$$A=SDS^{-1}, A^H=S^{-H}D^HS^H \Rightarrow AS=DS, A^HS^{-H}=S^{-H}D^H$$
$$\Rightarrow S=(p_1,\cdots,p_i,\cdots,p_n), S^{-H}=(q_1,\cdots,q_i,\cdots,q_n), Ap_i=\lambda_i p_i, A^Hq_i=\lambda_i^* q_i$$
$$\Rightarrow I=S^HS^{-H}=(p_1,\cdots,p_i,\cdots,p_n)^H(q_1,\cdots,q_i,\cdots,q_n)=(p_i^Hq_j)_{n\times n}$$
$$\Rightarrow p_i^Hq_j=\delta_{ij}$$

◊基本矩阵

$$E_{ij}^{(m\times n)} = e_i^{(m)} (e_j^{(n)})^T$$

性质：

(1) $E_{ij}^{(m\times n)} E_{kl}^{(n\times r)} = \delta_{jk} E_{il}^{(m\times r)}$

(2) $\left(E_{ij}^{(m\times n)}\right)^T = E_{ji}^{(n\times m)}$

(3) $A = \sum_{i=1}^{m} \sum_{j=1}^{n} a_{ij} E_{ij}^{(m\times n)}$

(4) $E_{ij}^{(s\times m)} A E_{kl}^{(n\times r)} = a_{jk} E_{il}^{(s\times r)}$

(5) $\det(E_{ij}^{(m\times n)}) = 0, (m = n > 1)$

两个向量(列向量)的外积:

[illegible]

```
>> a=[0 1 0 0 0]
a =
     0     1     0     0     0

>> b=[0 0 0 0 1 0 0]'
b =
     0
     0
     0
     0
     1
     0
     0

>> b*a
ans =
     0     0     0     0     0
     0     0     0     0     0
     0     0     0     0     0
     0     0     0     0     0
     0     1     0     0     0
     0     0     0     0     0
     0     0     0     0     0
```

Hermitian 矩阵特征值与特征向量的性质：

（1）Hermitian矩阵A的特征值一定是实的.

（2）令(λ,u)是Hermitian矩阵A的特征对.若A可逆,则$(\frac{1}{\lambda},u)$是其逆的特征对.

（3）Hermitian矩阵的所有特征向量线性无关，并且相互正交.

Hermitian矩阵的正定性判据：一个复共轭对称矩阵A是正定的，当且仅当满足以下任一条件：

（1）二次型函数 $x^H Ax > 0, \forall x \neq 0$

（2）矩阵A的所有特征值都大于零.

（3）所有主子矩阵A_k,1≤k≤n都具有正的行列式，其中A_k=A(1:k,1:k).

（4）存在一个$n \times n$ 非奇异矩阵R，使得$A = R^H R$.

（5）存在一个$n \times n$非奇异矩阵P，使得共轭对称矩阵 $P^H AP$ 正定.

哪一个判据最好用？哪一个判据对算力的消耗最少？

任意一个n阶可对角化方阵X可以分解为XV=VD，其中D为X的特征值对角阵，V为X的特征向量矩阵. MATLAB调用格式为:

[V,D]=eig(X)

```
>> X=normrnd(0,1,5,5);
>> [V,D]=eig(X)
V =
   0.0823 - 0.2221i   0.0823 + 0.2221i   0.5765 + 0.0000i   0.5765 + 0.0000i   0.3602 + 0.0000i
  -0.0009 - 0.4728i  -0.0009 + 0.4728i  -0.1210 + 0.1786i  -0.1210 - 0.1786i  -0.6226 + 0.0000i
   0.7210 + 0.0000i   0.7210 + 0.0000i  -0.2449 - 0.1294i  -0.2449 + 0.1294i   0.5490 + 0.0000i
  -0.3603 - 0.2473i  -0.3603 + 0.2473i  -0.4987 - 0.0690i  -0.4987 + 0.0690i  -0.2397 + 0.0000i
  -0.0122 - 0.0973i  -0.0122 + 0.0973i   0.4946 + 0.2152i   0.4946 - 0.2152i   0.3517 + 0.0000i
D =
  -1.0217 + 1.9525i   0.0000 + 0.0000i   0.0000 + 0.0000i   0.0000 + 0.0000i   0.0000 + 0.0000i
   0.0000 + 0.0000i  -1.0217 - 1.9525i   0.0000 + 0.0000i   0.0000 + 0.0000i   0.0000 + 0.0000i
   0.0000 + 0.0000i   0.0000 + 0.0000i   1.5298 + 0.2746i   0.0000 + 0.0000i   0.0000 + 0.0000i
   0.0000 + 0.0000i   0.0000 + 0.0000i   0.0000 + 0.0000i   1.5298 - 0.2746i   0.0000 + 0.0000i
   0.0000 + 0.0000i   0.0000 + 0.0000i   0.0000 + 0.0000i   0.0000 + 0.0000i  -0.1868 + 0.0000i
X =
    0.0345    0.2367   -0.4789   -1.9090   -0.3613
   -1.8679    0.4319    0.8065   -1.4057    0.7911
   -1.4789   -1.8345   -1.1734   -0.6161   -0.6130
    0.4422    0.7225    1.3012    0.3693   -0.8261
   -0.7832    0.5899   -0.2038   -1.1895    1.1671
```

案例2：基于特征值分解的图像压缩

```
%读原图像
X=imread('test.jpg');
Y=rgb2gray(X);
figure(1);imshow(Y);

%将原图像整理为正方图像，宽高为H
H=100;
S=size(Y);%得图像的宽和高
Z=zeros(H,H);

for ii=1:H
    for jj=1:H
        Z(ii,jj)=Y(fix((S(1)-1)/H*ii) ...
                 +1,fix((S(2)-1)/H*jj)+1);
    end
end
figure(2); imshow(uint8(Z));
title('原图');

%特征值分解
[U,D]=eig(double(Z));
M=zeros(H,H);%此矩阵用来摘取成分
```

```
for LL=1:H/10-1;
        MM=M;
        for ii=1:(LL+1)*10
            MM(ii,ii)=1;
        end

        %基于特征值分解进行图像重建（还原）
        GG=U*MM*D/U;
        figure(3),subplot(3,3,LL),
        imshow(uint8(abs(GG)));
        title(['LL=',num2str(LL)]);
end

ZZ=zeros(10,10);
flag=1;
for ii=1:10
    for jj=1:10
        ZZ(ii,jj)=abs(D(flag,flag));
        flag=flag+1;
    end
end
disp(ZZ);
```

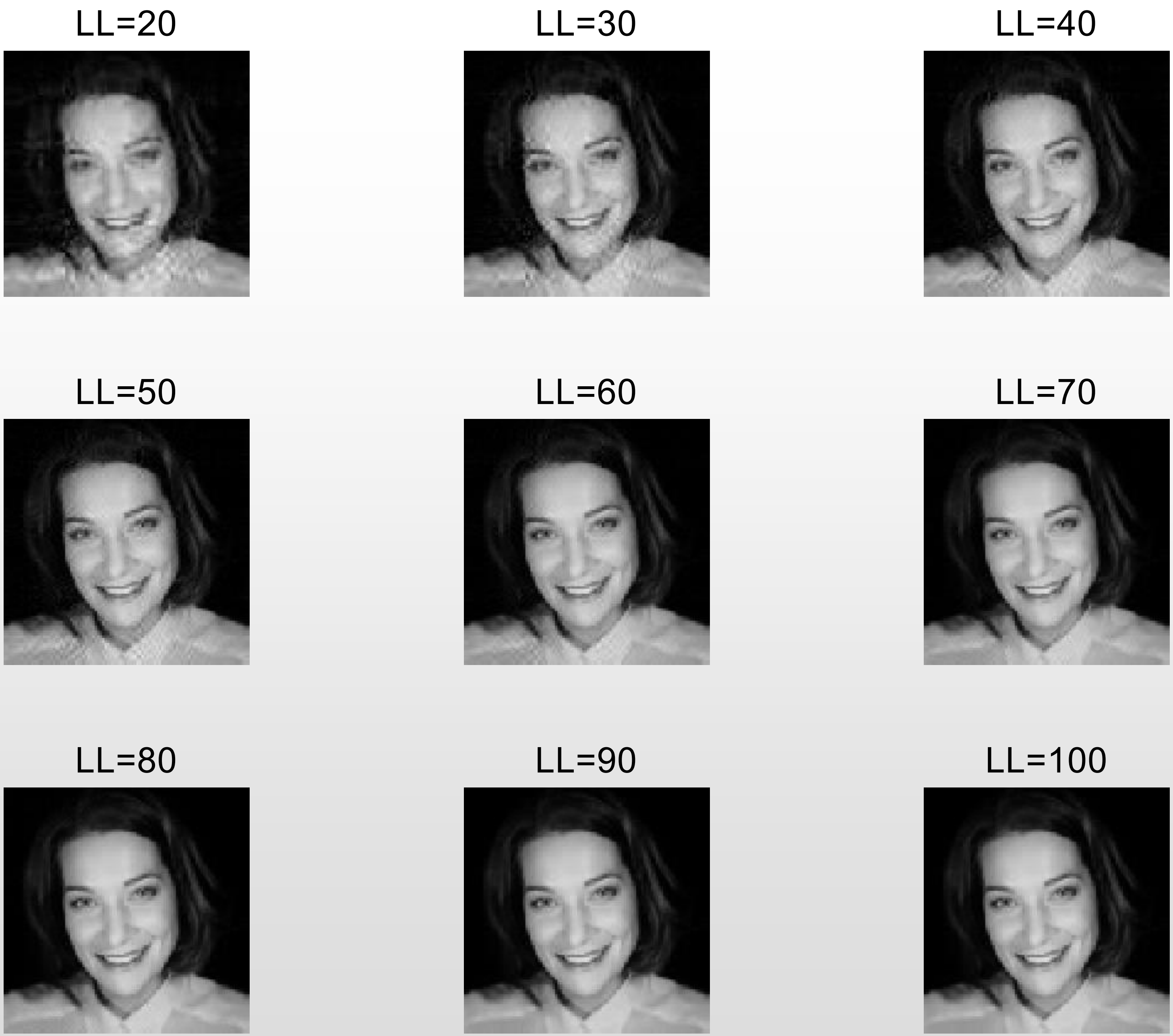


图1.基于特征值的图像压缩. LL:保留的特征值个数；不保留的用0代替.

特征值按绝对值大小排列（从大到小）(????)

1.0e+03 *

7.5101	2.5494	0.9648	0.6017	0.3363	0.3363	0.3211	0.3211	0.2639	0.2370
0.2370	0.2206	0.2206	0.1555	0.1555	0.1485	0.1485	0.1297	0.1297	0.1100
0.1100	0.0836	0.0747	0.0747	0.0662	0.0662	0.0652	0.0652	0.0584	0.0584
0.0522	0.0522	0.0393	0.0326	0.0326	0.0329	0.0329	0.0321	0.0321	0.0235
0.0235	0.0228	0.0228	0.0222	0.0222	0.0196	0.0196	0.0195	0.0195	0.0185
0.0185	0.0190	0.0190	0.0178	0.0178	0.0117	0.0103	0.0103	0.0097	0.0097
0.0093	0.0093	0.0094	0.0094	0.0084	0.0084	0.0083	0.0083	0.0078	0.0078
0.0062	0.0062	0.0062	0.0056	0.0056	0.0052	0.0052	0.0047	0.0047	0.0045
0.0045	0.0035	0.0028	0.0028	0.0027	0.0027	0.0023	0.0020	0.0020	0.0015
0.0007	0.0007	0.0006	0.0006	0.0006	0.0006	0.0004	0.0004	0.0004	0.0002

算法:

Step 1：将待处理数据用矩阵表示，X;（需方阵化）

Step 2：求该矩阵的特征值分解，[V,D]=eig(X);

Step 3： 根据D的特点，对D进行数据舍取，得A;

Step 4：重建或复原X，X=V*A/V.（或需反方阵化）

原理:

特征值的绝对值相当于权重，舍弃权重小的数据对复原矩阵的精度影响不大.

原图

- 默认情况下，eig 并不总是返回已排序的特征值和特征向量.可以使用 sort 函数将特征值按升序排序，并重新排序相应的特征向量.

- A 的特征值位于 D 的对角线上.但是，特征值并未排序.

- 使用 diag(D) 从 D 的对角线上提取特征值，然后按升序对得到的向量进行排序.sort 的第二个输出返回索引的置换向量.

```
A=normrnd(0,1,5,5);
[V,D] = eig(A);
[d,ind] = sort(diag(abs(D)));
Ds = D(ind,ind);
Vs=V(:,ind);
norm(V*D*V'-Vs*Ds*Vs')

ans =
  1.1102e-16
```

按特征值的绝对值大小排序后的分解：Vs*Ds*Vs'

```
d =
    0.4025
    1.8090
    2.9650
    2.9650
    3.6472

ind =
     3
     4
     1
     2
     5
```

direction — Sorting direction
'ascend' (default) | 'descend'

◆主成分分析（Principal Component Analysis, PCA)

● Hotelling Transform

设X是实随机P维列向量：

$$X = (x_1, x_2, \cdots, x_P)^T$$

其数学期望为：

$$\mathbf{E}X = (\mathbf{E}x_1, \mathbf{E}x_2, \cdots, \mathbf{E}x_P)^T$$

自协方差矩阵为：(注：也可以按照自相关矩阵这条概念线路讨论此问题)

$$\mathbf{R}_X = \mathbf{E}\{(X - \mathbf{E}X)(X - \mathbf{E}X)^T\} = \mathbf{E}\{XX^T\} - \mathbf{E}X \cdot \mathbf{E}X^T$$

对R_x进行特征值分解，得：

$$\mathbf{R}_X = U^T \Lambda U$$

其中，U为正交矩阵， $\Lambda = diag(\lambda_1, \lambda_2, \cdots, \lambda_P)$， 令

$$Y = U(X - \mathbf{E}X)$$ （**Hotelling Transform**)

推论3：$\mathbf{R}_Y = diag(\lambda_1, \lambda_2, \cdots, \lambda_P)$

推论4: $\| X - \mathbf{E}X \|_2^2 = \| Y \|_2^2 = \sum_{i=1}^{P} | \lambda_i |^2$

Hotelling变换的计算：（主要问题是如何得到数据样本的期望值和自相关矩阵）

（1）若知道了X的分布函数，则基于其分布函数可以得到 $\mathbf{E}X, \mathbf{R}_X$.（**理论**）

（2）根据样本数据得到 $\mathbf{E}X, \mathbf{R}_X$.(**数据实验**)

$$\mathbf{E}X = \frac{1}{K}\sum_{j=1}^{K} X_j, \mathbf{R}_X = \frac{1}{K}\sum_{j=1}^{K} X_j X_j^T - \mathbf{E}X \cdot \mathbf{E}X^T$$

- 主成分和次成分

可以将特征值按大小排序: $\mathbf{R}_Y = diag(\lambda_1, \lambda_2, \cdots, \lambda_P)$

定义7：随机向量X的自相关矩阵有K个主特征，与这些主特征对应的K个特征向量称为该随机向量的主成分.

定义8：随机向量X的自相关矩阵有K个主特征，P-K个次特征（小特征），与这些次特征对应的P-K个特征向量称为该随机向量的次成分.

定义9：只利用K个主特征进行的数据分析称为主成分分析(principal component analysis, PCA); 利用P-K个次特征进行的数据分析称为次成分分析(minor component analysis, MCA).

● 主成分分析算法

Step 1: 对数据进行Hotelling变换（一般是先对数据进行向量化或矩阵化），也就是求数据的特征值分解.（**正交化**）

Step 2:按功率或能量大小对特征进行由大到小排序.（**功率最大化**）

Step 3:选择恰当的策略(设定主特征个数K)，丢弃次特征(或主特征).(**降维**)

Step 4: 分析或使用降维以后的数据.（**分析或使用**）

以上算法，也适合次成分分析.

案例1：基于特征值分解的图像压缩

参见本PPT的第31页.

注：主成分分析不一定只用在图像处理领域.

如何得到功率谱：

（1）理论推导;

（2）对**全体数据**进行统计分析;

（3）在数据统计规律具有平稳性质的假定下，使用**过往数据**的功率谱.

◆基于特征脸的人脸识别

1987年由Sirovich和Kirby提出.（其它：LBP，稀疏表示，神经网络等）

设有M张人脸图像，每张人脸的类型是已知的(是谁是清楚的). 建设一个人脸数据库. 每张人脸图像的规格为$m\times n$，令$N=m\times n$. 令X_i表示第i张人脸图像，

$$\mu=\frac{1}{M}\sum_{i=1}^{M}X_i, x_i=vec(X_i), Z=(x_i-\mu)_{N\times M}$$

显然，ZZ^T和Z^TZ都是实对称矩阵. 令

$$Z^TZ=U^T diag(\lambda_1,\lambda_2,\cdots,\lambda_M)U, UZ^TZU^T=diag(\lambda_1,\lambda_2,\cdots,\lambda_M)$$

称$Y_{N\times M}=ZU^T$为特征脸.设对$Y_{N\times M}$进行降维得$Y_{d\times M}=[Y_{N\times M}]_d$ ($[\ ]_d$表示对列向量取前d个分量). 则重建图像为$Z^*=Y_{d\times M}U+\mu$.

算法:

T_1,T_2是决策阈值.

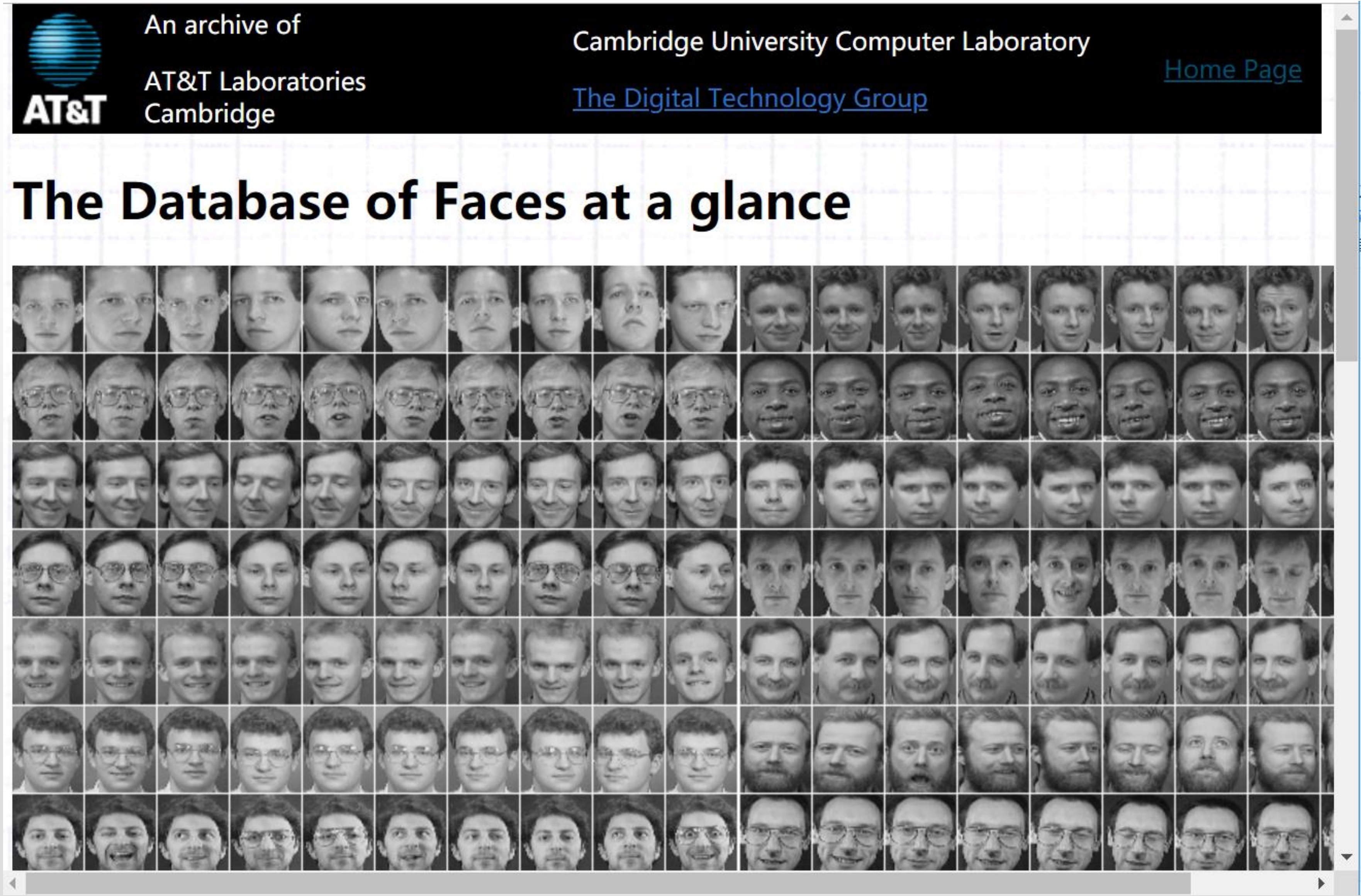
Step 1: 构建降维特征脸库.

Step 2: 输入人脸图像X, 计算其降维重建图像Y，若$\|X-Y\|>T_1$, 判负，终止.

Step 3: 遍历特征脸库$Y_{d\times M}$, 若$\|Y-Y_{d\times M}\|>T_2$, 判负，终止；否则，输出其类型.

案例3：基于特征脸的人脸识别（ORL数据集）

https://www.cl.cam.ac.uk/research/dtg/attarchive/facesataglance.html



ORL数据集

https://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html

AT&T

An archive of

AT&T Laboratories
Cambridge

Cambridge University Computer Laboratory

The Digital Technology Group

Home Page

The Database of Faces

Our Database of Faces, (formerly 'The ORL Database of Faces'), contains a set of face images taken between April 1992 and April 1994 at the lab. The database was used in the context of a face recognition project carried out in collaboration with the Speech, Vision and Robotics Group of the Cambridge University Engineering Department.

There are ten different images of each of 40 distinct subjects. For some subjects, the images were taken at different times, varying the lighting, facial expressions (open / closed eyes, smiling / not smiling) and facial details (glasses / no glasses). All the images were taken against a dark homogeneous background with the subjects in an upright, frontal position (with tolerance for some side movement). A preview image of the Database of Faces is available.

The files are in PGM format, and can conveniently be viewed on UNIX (TM) systems using the 'xv' program. The size of each image is 92x112 pixels, with 256 grey levels per pixel. The images are organised in 40 directories (one for each subject), which have names of the form `sX`, where `X` indicates the subject number (between 1 and 40). In each of these directories, there are ten different images of that subject, which have names of the form `Y.pgm`, where `Y` is the image number for that subject (between 1 and 10).

The database can be retrieved from http://www.cl.cam.ac.uk/Research/DTG/attarchive:pub/data/att_faces.tar.Z as a 4.5Mbyte compressed `tar` file or from

https://blog.csdn.net/hesays/article/details/39498375 ：(基于PCA的人脸识别的MATLAB实现代码)

[1] Turk,M.A.; Media Lab., MIT, Cambridge, MA, USA ; Pentland, A.P. [Face Recognitionusing Eigenfaces](). IEEE Signal Processing Society,1991.7(格式不对)

Face Recognition Using Eigenfaces

Matthew A. Turk and Alex P. Pentland

Vision and Modeling Group, The Media Laboratory
Massachusetts Institute of Technology

[2] John Wright , Allen Y. Yang, Arvind Ganesh, S. Shankar Sastry, Yi Ma. [Robust Face Recognition via Sparse Representation](). IEEE Transactions on Pattern Analysis and Machine Intelligence, 31(2):210 - 227, Feb. 2009.

Robust Face Recognition via Sparse Representation

John Wright, *Student Member, IEEE*, Allen Y. Yang, *Member, IEEE*, Arvind Ganesh, *Student Member, IEEE*, S. Shankar Sastry, *Fellow, IEEE*, and Yi Ma, *Senior Member, IEEE*

人脸识别的其它MATLAB代码：

- https://www.mathworks.com/matlabcentral/fileexchange/48479-face-recognition-using-eigenfaces（特征脸人脸识别）
- https://blog.csdn.net/liangjiubujiu/article/details/80278079（人脸稀疏表示）

4.1 数值稳定性与条件数

方程AX=b的解X，当A和b做微小扰动时，其相对误差如何？通过简单推导，可得：若b有微小扰动，则解的相对扰动为

$$\frac{\|\delta X\|_2}{\|X\|_2} \le (\|A\|_2 \cdot \|A^{-1}\|_2)\frac{\|\delta b\|_2}{\|b\|_2}$$

若A有微小扰动，则解的相对扰动为

$$\frac{\|\delta X\|_2}{\|X+\delta X\|_2} \le (\|A\|_2 \cdot \|A^{-1}\|_2)\frac{\|\delta \mathrm{A}\|_2}{\|\mathrm{A}\|_2}$$

考虑扰动对精度的影响！

数值稳定性

定义：矩阵A的条件数为

$$cond(A) = \|A\|_2 \cdot \|A^{-1}\|_2$$

条件数的性质

(1) $cond\,(A) \ge 1$. 特别地，若A为正交矩阵或酉矩阵，则$cond\,(A)=1$.

(2) $cond(aA) = cond(A), (a \ne 0)$.

(3) $cond(A^H A) = [cond(A)]^2$.

(4) $cond(QA) = cond(A)$, Q为酉矩阵.

$$AX = b$$
$$\Rightarrow A(X + \delta X) = b + \delta b$$
$$\Rightarrow AX + A\delta X = b + \delta b$$
$$\Rightarrow \delta X = A^{-1}\delta b$$
$$\| AX \|_2 = \| b \|_2,$$
$$\| A \|_2 . \| X \|_2 \geq \| AX \|_2 = \| b \|_2 \quad (1)$$
$$\delta X = A^{-1}\delta b,$$
$$\| \delta X \|_2 = \| A^{-1}\delta b \|_2 \leq \| A^{-1} \|_2 . \| \delta b \|_2 \quad (2)$$
$$\frac{\| \delta X \|_2}{\| X \|_2} \leq (\| A \|_2 . \| A^{-1} \|_2) \frac{\| \delta b \|_2}{\| b \|_2}$$

$$AX = b$$
$$\Rightarrow (A + \delta A)(X + \delta X) = b$$
$$\Rightarrow AX + A\delta X + \delta A X + \delta A \delta X = b$$
$$\Rightarrow A\delta X + \delta A X + \delta A \delta X = 0$$
$$\delta X = -A^{-1}\delta A(X + \delta X),$$
$$\| \delta X \|_2 = \| -A^{-1}\delta A(X + \delta X) \|_2,$$
$$\| \delta X \|_2 \leq \| A^{-1} \|_2 . \| \delta A \|_2 . \| (X + \delta X) \|_2,$$
$$\frac{\| \delta X \|_2}{\| (X + \delta X) \|_2} \leq (\| A \|_2 . \| A^{-1} \|_2) \frac{\| \delta A \|_2}{\| A \|_2}$$

$$AX = b$$
$$\Rightarrow (A + \delta A)(X + \delta X) = b + \delta b$$
$$\Rightarrow AX + A\delta X + \delta A X + \delta A \delta X = b + \delta b$$
$$\Rightarrow A\delta X + \delta A(X + \delta X) = \delta b$$
$$\delta X = -A^{-1}\delta A(X + \delta X) + A^{-1}\delta b,$$
$$\| \delta X \|_2 \leq \| -A^{-1}\delta A(X + \delta X) \|_2 + \| A^{-1}\delta b \|_2,$$
$$\| \delta X \|_2 \leq \| A^{-1} \|_2 . \| \delta A \|_2 . \| (X + \delta X) \|_2 + \| A^{-1}\delta b \|_2,$$
$$\frac{\| \delta X \|_2}{\| (X + \delta X) \|_2} \leq (\| A \|_2 . \| A^{-1} \|_2) \frac{\| \delta A \|_2}{\| A \|_2} + \frac{\| A^{-1}\delta b \|_2}{\| (X + \delta X) \|_2}$$
$$\frac{\| A^{-1}\delta b \|_2}{\| (X + \delta X) \|_2} = \frac{\| A^{-1}\delta b \|_2}{\| (A + \delta A)^{-1}(b + \delta b) \|_2}$$
$$\| (A + \delta A)^{-1}(b + \delta b) \|_2 . \| A + \delta A \|_2 \geq \| b + \delta b \|_2$$
$$\frac{\| A^{-1}\delta b \|_2}{\| (A + \delta A)^{-1}(b + \delta b) \|_2} \leq \frac{\| A + \delta A \|_2 . \| A^{-1}\delta b \|_2}{\| b + \delta b \|_2}$$
$$\leq \frac{\| A + \delta A \|_2 . \| A^{-1}\delta b \|_2}{\| b + \delta b \|_2}$$
$$\frac{\| \delta X \|_2}{\| (X + \delta X) \|_2} \leq (\| A \|_2 . \| A^{-1} \|_2) \frac{\| \delta A \|_2}{\| A \|_2}$$
$$+ (\| A + \delta A \|_2 . \| A^{-1} \|_2) \frac{\| \delta b \|_2}{\| (b + \delta b) \|_2}$$

例1：计算条件数

```
d=0.01;
D=diag([4,3,2,d]);
U=rand(4,4);
V=rand(4,4);
A=U*D*V;
invA=inv(A);
disp(A);
disp(invA);
cond(A)
norm(A)*norm(invA)
```

```
    2.5795    4.6312    1.7337    3.7619
    2.2776    4.1109    1.9356    3.0939
    3.9563    4.1405    2.4062    4.8637
    1.2411    1.2602    0.6975    1.5431

  -54.0508   72.0621 -177.5613  546.9398
  -15.3375   21.3188  -52.6344  160.5451
   24.0355  -32.7500   85.5925 -262.7109
   45.1329  -60.5646  147.1038 -451.6046


ans =

   9.9788e+03


ans =

   9.9788e+03
```

例2：计算条件数与矩阵奇异性之间的关系

```
%设置一个 小参数
d=0.01;
%生成两个满秩矩阵
U=rand(4,4);
V=rand(4,4);
%基于d生成一个矩阵序列
for I=1:100
    dd=d*I;
    D=diag([4,3,2,dd]);
    A=U*D*V;
    %计算矩阵条件数
    Y(I)=cond(A);
end
%画Y的函数图象
plot(Y);
xlabel('d的取值. 越小其性能越接近奇异矩阵');
ylabel('条件数');
```

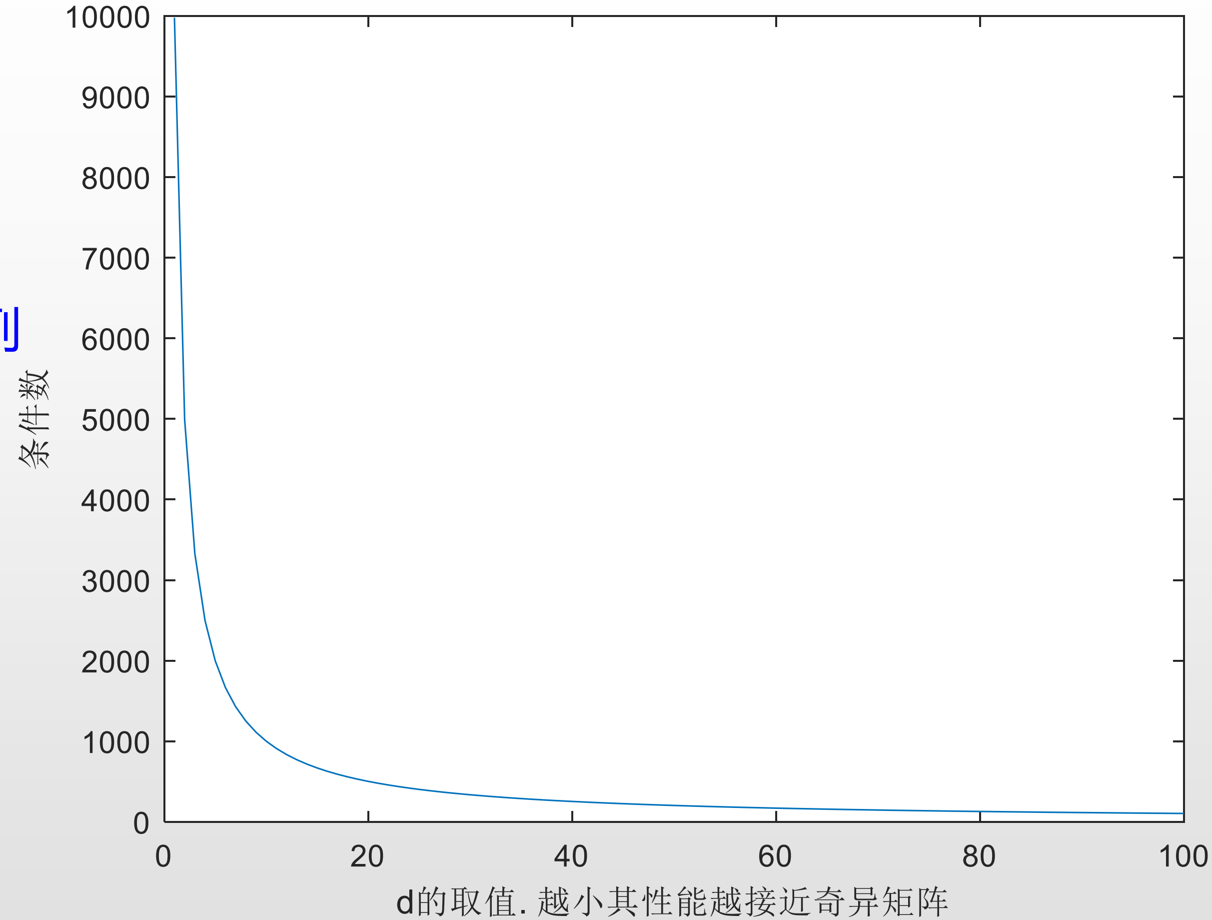


图2. 矩阵条件数与奇异性之间的关系

结论：矩阵越“靠近”奇异矩阵，条件数越大.

例3：条件数与方程解的精度之间的关系

```
dd=0.01;
b=rand(4,1);
dA=0.01*rand(4,4);
U=rand(4,4);
V=rand(4,4);
for I=1:100
      d=dd*I;
      D=diag([4,3,2,d]);
      A=U*D*V;
      Y(I)=cond(A)
      X=A\b;
      XX=(A+dA)\b;
      a1=norm(XX);
      a2=norm(XX-X);
      YY(I)=a2/a1;
end
plot(YY,Y);
```

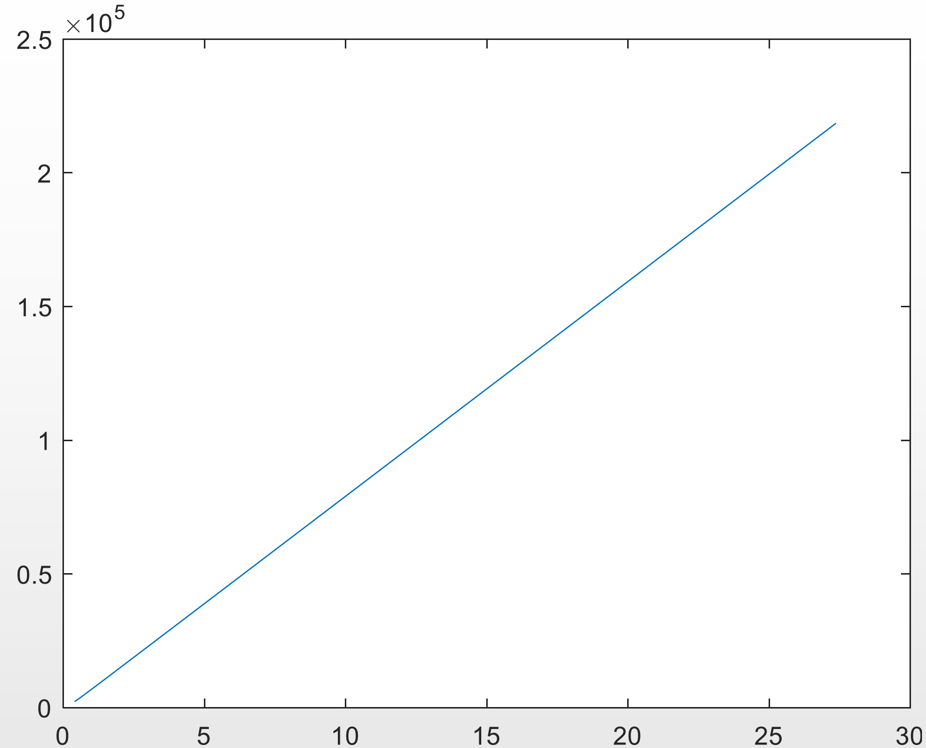


图3. 矩阵条件数与精度之间的关系

```
Y =
  1.0e+04 *
4.1751    2.0752    1.3753    1.0254    0.8155
0.6755    0.5756    0.5007    0.4424    0.3959
YY =
15.3007    7.5783    5.0043    3.7174    2.9453
2.4306    2.0630    1.7874    1.5731    1.4017
```

结论：相对精度与条件数成正比！

4.2 奇异值分解

定理（奇异值分解）：令 $A \in R^{m\times n}$ (或$C^{m\times n}$)，则存在正交(或酉)矩阵$U \in R^{m\times m}$ (或$C^{m\times m}$)和$V \in R^{n\times n}$(或 $C^{n\times n}$)使得

$$A = U\Sigma V^T \quad (\text{或 } U\Sigma V^H)$$

式中

$$\Sigma = \begin{bmatrix} \Sigma_1 & 0 \\ 0 & 0 \end{bmatrix}_{m\times n}$$

且 $\Sigma_1 = diag(\sigma_1, \sigma_2, \cdots, \sigma_r), r = rank(A)$，其对角元素按照由大到小顺序排列，即

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0$$

定理3*（QR分解）：设实(复)矩阵 $A_{m\times n}$ 的n个列线性无关(n≤m)，则存在正交矩阵 $Q_{m\times m}$和实(复)非奇异上三角矩阵 $R_{m\times n}$，使得

$$A = QR.$$

$$A_{m\times n} = Q_{m\times m}R_{m\times n}, R_{m\times n}^H = P_{n\times n}S_{n\times m}, A_{m\times n} = Q_{m\times m}S_{n\times m}^H P_{n\times n}^H$$ （？，从AA^H,A^HA下手）

关于奇异值和奇异值分解的几点解释和标记:

(1) V的列向量 v_i 称为A的右奇异向量(right singular vector)，V称为A的右奇异向量矩阵.

(2) U的列向量 u_i 称为A的左奇异向量(left singular vector)，并称U为A的左奇异向量矩阵.

(3) $u_i^H A v_i = \sigma_i, i = 1,2,\cdots,\min\{m,n\}$

$$U^H AV = \begin{bmatrix} \Sigma_1 & 0 \\ 0 & 0 \end{bmatrix}, \quad \Sigma_1 = \begin{bmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & \cdots & 0 & \sigma_r \end{bmatrix}$$

(4) 矩阵A的奇异值分解可以改写成向量表达形式:

$$A = \sum_{i=1}^{r} \sigma_i u_i v_i^H$$

这种表达有时称为A的**并向量（奇异值）分解**.

(5) 由奇异值分解公式易得

$$AA^H = U\left(\Sigma\Sigma^H\right)U^H$$

这表明，$m \times n$矩阵A的奇异值 σ_i 是矩阵乘积 AA^H 的特征值的正平方根.

(6) 设 $r = rank(A) < \min\{m, n\}$

$$A = U_r \Sigma_r V_r^H$$

称为矩阵A的**截尾奇异值分解(truncated SVD)**或薄奇异值分解(thin SVD).
式中

$$U_r = [u_1, u_2, \cdots, u_r]$$

$$V_r = [v_1, v_2, \cdots, v_r]$$

$$\Sigma_r = diag(\sigma_1, \sigma_2, \cdots, \sigma_r)$$

补充：外积

(1)列向量u和v的外积是指矩阵uv^T

$$A=\sum_{i=1}^{r}\sigma_i u_i v_i^H=U\Lambda V^H$$

SVD分解：任意矩阵都是其左奇异向量和右奇异向量的外积组合，组合系数是其奇异值.

(2)奇异值和特征值的联系

$$A^H A=(U\Lambda V^H)^H U\Lambda V^H=V\Lambda^H\Lambda V^H$$

$$AA^H=U\Lambda V^H(U\Lambda V^H)^H=U\Lambda\Lambda^H U^H$$

奇异值分解的性质

（1）非零奇异值的个数 r 和它们的值 $\sigma_1,\sigma_2,\cdots,\sigma_r$ 相对于矩阵*A*是唯一确定的，但**左右奇异向量矩阵***U*，*V*却不是唯一的.

（2）$m\times n$矩阵*A*的共轭转置 A^H 的奇异值分解为

$$A^H = V\Sigma^T U^H$$

（3）P 和Q分别为$m\times m$和$n\times n$酉矩阵时，PAQ^H 的奇异值分解为

$$PAQ^H = \tilde{U}\Sigma\tilde{V}^H$$

其中, $\tilde{U} = PU, \tilde{V} = QV$.

（4） $A^H A$ 和 AA^H 的奇异值分解分别为

$$A^H A = V\Sigma^T\Sigma V^H,\ AA^H = U\Sigma\Sigma^T U^H$$

其中

$$\Sigma^T\Sigma = diag\left(\sigma_1^2,\sigma_2^2,\cdots,\sigma_r^2;\overbrace{0,\cdots,0}^{n-r}\right)$$

$$\Sigma\Sigma^T = diag\left(\sigma_1^2,\sigma_2^2,\cdots,\sigma_r^2;\overbrace{0,\cdots,0}^{m-r}\right)$$

（5）**Moore-Penrose逆矩阵（摩尔-彭罗斯广义逆）**

$$A^\dagger = V\Sigma^\dagger U^H$$

定理（Eckart-Young）：设$A \in R^{m\times n}$ 的奇异值分解为$A=U\Sigma V^T$

且$\sigma_1 \ge \sigma_2 \ge \cdots \ge \sigma_r > 0, r = rank(A)$． 令

$$A_k = \sum_{i=1}^{k} \sigma_i u_i v_i^T, k \le r$$

则

$$\min_{rank(X)\le k} \|A-X\|_{spec} = \|A-A_k\|_{spec} = \sigma_{k+1} \quad \text{(谱范数逼近)}$$

$$\min_{rank(X)\le k} \|A-X\|_F = \|A-A_k\|_F = \sqrt{\sigma_{k+1}^2+\cdots+\sigma_r^2} \quad \text{(F范数逼近)}$$

$$-E = A - X \Rightarrow X = A + E$$

推论：设 $A \in R^{m\times n}$ 的奇异值为 $\sigma_1 \ge \cdots \ge \sigma_r > 0, r = rank(A)$，则

$$\sigma_k = \min_{E\in C^{m\times n}} \left\{ \|E\|_{spec} : rank(A+E) \le k-1 \right\}, k = 1, \cdots, r$$

且**最佳扰动矩阵**为

$$E_k = -\sum_{i=k}^{r} \sigma_i u_i v_i^T$$

此时，$\|E\|_{spec} = \sigma_k$ 且 $rank(A+E_k) = k-1.$

矩阵的低秩逼近解决：降维的可能性，“降维后信息损失有多大？”等问题？

矩阵的奇异值与条件数的关系

$$cond(A)=\frac{\sigma_1}{\sigma_p}, p=\min\{m,n\}, p\geq rank(A).$$

因此，（1）条件数是一个大于或等于1的整数.

（2）奇异矩阵的条件数为无穷大.

（3）条件数虽不是无穷大，但却很大时，A是接近奇异的，即此时其行或列线性相关性很强.

（4）$cond\left(A^H A\right)=\sigma_1^2/\sigma_p^2=[cond(A)]^2$

奇异值与矩阵范数的关系

（1）谱范数：$\|A\|_{spec}=\sigma_1$

（2）Frobenius范数：

$$\|A\|_F=\left[\sum_{i=1}^{m}\sum_{j=1}^{n}|a_{ij}|^2\right]^{1/2}=\sqrt{\sigma_1^2+\sigma_2^2+\cdots+\sigma_r^2}$$

奇异值分解与特征值之间的联系与区别

1）奇异值分解适用于任何长方形矩阵，特征值分解只适用于正方矩阵.

2）即使是同一个$n\times n$非Hermitian矩阵A，奇异值和特征值的定义也是完全不同的：**奇异值定义为**

最佳扰动矩阵

为了降维！

$$\sigma_k = \min_{E\in C^{m\times n}} \left\{ \|E\|_{spec} : rank(A+E) \le k-1 \right\}$$

$$\left(k \le \min\{m,n\}\right)$$

而**特征值定义为**特征多项式方程 $\det(A-\lambda I)=0$ 的根.

同一个正方矩阵的奇异值和特征值之间没有内在的关系，但是$m\times n$矩阵A的<u>非零奇异值</u>是 $n\times n$ Hermitian矩阵 $A^H A$或$m\times m$ Hermitian矩阵AA^H 的非零特征值的正平方根.

设$n \times n$正方矩阵A的特征值为

$$\lambda_1, \lambda_2, \cdots, \lambda_n \qquad \left(|\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_n|\right)$$

奇异值为

$$\sigma_1, \sigma_2, \cdots, \sigma_n \qquad \left(\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0\right)$$

则有

$$\sigma_1 \geq |\lambda_i| \geq \sigma_n \qquad \left(i = 1, 2, \cdots, n\right)$$

$$cond(A) \geq \frac{|\lambda_1|}{|\lambda_n|}.$$

特别地，当A(共轭)对称时，L_2-范数下的条件数

$$cond(A) = \frac{|\lambda_1|}{|\lambda_n|}.$$

（3）$m\times n$矩阵A的左、右**奇异向量**

$$u_i^H A v_i = \sigma_i$$

而$n\times n$矩阵A的左、右特征向量

$$u^H A = \lambda_i u^H, Av_i = \lambda_i v_i$$

因此，对于同一个$n\times n$非Hermitian矩阵A，其（左和右）奇异向量与（左和右）特征向量之间没有内在的关系.

矩阵$A\in C^{m\times n}$ 的左奇异向量u_i 是$m\times m$ Hermitian矩阵 AA^H 的特征向量. 类似地，矩阵$A\in C^{m\times n}$的右奇异向量v_i 是$n\times n$Hermitian矩阵 $A^H A$的特征向量.

任意一个$m\times n$维的矩阵X可以分解为X=USV'，U和V均为正交(酉)矩阵，S为$m\times n$维的对角矩阵，其对角线元素为X的从大到小排序的非负奇异值.

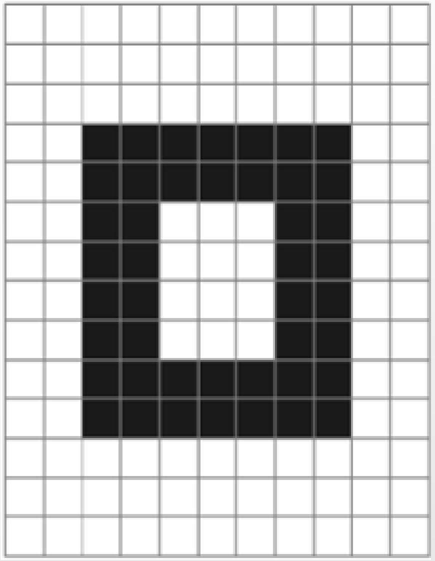
MATLAB调用格式为：

[U,S,V]=svd(X)

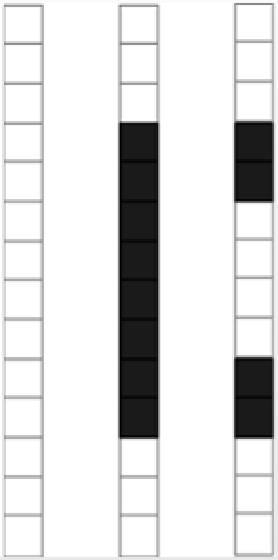
U,V为正交阵，S为对角阵（对角线上元素为X的奇异值）

注：**矩阵$X^H X$的特征值的算数平方根.**

案例4. 基于奇异值分解的图像压缩



14×11图像



3个列基矢量

教材4.4.2：图像压缩.

$$A=\begin{bmatrix}1&1&1&1&1&1&1&1&1&1&1\\1&1&1&1&1&1&1&1&1&1&1\\1&1&1&1&1&1&1&1&1&1&1\\1&1&0&0&0&0&0&0&0&1&1\\1&1&0&0&0&0&0&0&0&1&1\\1&1&0&0&1&1&1&0&0&1&1\\1&1&0&0&1&1&1&0&0&1&1\\1&1&0&0&1&1&1&0&0&1&1\\1&1&0&0&1&1&1&0&0&1&1\\1&1&0&0&0&0&0&0&0&1&1\\1&1&0&0&0&0&0&0&0&1&1\\1&1&1&1&1&1&1&1&1&1&1\\1&1&1&1&1&1&1&1&1&1&1\\1&1&1&1&1&1&1&1&1&1&1\end{bmatrix}_{14\times 11}$$

上述原始图像需要用14×11=154个数值来表示、存储.

对A进行奇异值分解$A = U\Sigma V^T$

$$\begin{aligned}\sigma_1 &= 9.7065 \\ \sigma_2 &= 3.3983 \\ \sigma_3 &= 2.0579\end{aligned} \quad \Rightarrow \quad A = \sum_{i=1}^{3} \sigma_i u_i v_i^T$$

这样，该图像可以用$\{(\sigma_i, u_i, v_i), i = 1,2,3\}$ 来完全刻画．此时，所需要的元素个数仅为(14+11+1)×3=78.

推而广之，一幅$m\times n$ 图像的像素矩阵A具有奇异值分解 $A = U\Sigma V^T$，其中奇异值按从大到小顺序排列．从中选择k个大奇异值及与之对应的左、右奇异向量，以此，可用$k(m+n+1)$个数值近似代替原来的$m\times n$个图像数据.

压缩比

$$\rho = \frac{m\times n}{k(m+n+1)}, k \le \min\{ m, n\}.$$

案例4(图像压缩MATLAB代码)

```
%读原图像
X=imread('test.jpg');
Y=rgb2gray(X);
figure(1);imshow(Y);

%奇异值分解
[U,D,V]=svd(double(Y));

SS=size(D);

M=zeros(SS(1),SS(2));
H=min(SS(1),SS(2));
dd=fix(H/9);
d=fix(sqrt(dd));

for L=1:d*d
MM=M;
for ii=1:(L)*9
    MM(ii,ii)=1;
end
rho=SS(1)*SS(2)/((SS(1)+SS(2)+1)*L);
%基于特征值分解进行图像重建（还原）
        GG=U*(MM.*D)*V';
        figure(3),subplot(d,d,L),
        imshow(uint8(abs(GG)));
        title(['L=',num2str(L*9),', rho=',num2str(rho)]);
end

ZZ=zeros(d,d);
flag=1;
for ii=1:d
    for jj=1:d
        ZZ(ii,jj)=abs(D(flag,flag));
        flag=flag+1;
    end
end
disp(ZZ);
```

test5.m

```
1.0e+04 *
   1.5504    0.5922    0.2357    0.1769
   0.1311    0.1057    0.0974    0.0727
   0.0607    0.0547    0.0494    0.0432
   0.0407    0.0399    0.0347    0.0330
```

前16个奇异值

图4. 基于SVD的图像压缩. L：保留下来的奇异值个数，rho:压缩比(原图像数据量/压缩后的图像数据量)

案例5： 基于奇异值分解的数字水印技术

（1）数字水印

为了版权保护和防止篡改，嵌入在数字载体(多媒体、文档、软件等)当中的标识信息；原则：该标志信息的嵌入不影响原载体的使用，也不容易被探知和再次修改，但可以被生产方识别和辨认.

按特性划分:

鲁棒水印(Robust Watermarking):用于在数字作品中标识著作权信息

脆弱水印(Fragile Watermarking):用于数字作品的完整性保护和认证

按用途划分:

票证防伪水印、版权保护水印、篡改提示水印和隐蔽标识水印

此三幅图像来源网络 仅做教学观摩使用 版权属于原作者

数字水印技术模型主要有三部分组成：

（1）水印信息

（2）水印嵌入算法

（3）水印提取、检测和验证算法

嵌入过程：

$$\begin{cases} A \Rightarrow USV^T \\ L \Leftarrow S + aW \\ L \Rightarrow U_1 S_1 V_1^T \\ A_w \Leftarrow U S_1 V^T \end{cases}$$

A： 原始图像
W： 水印
A_w: 含有数字水印的图像
a: 水印强度系数
L： 过渡矩阵

提取过程：

$$\begin{cases} P \Rightarrow U_p S_p V_p^{\ T} \\ F \Leftarrow U_1 S_p V_1^T \\ W_E \Leftarrow (F - S)/a \end{cases}$$

a, U1,V1,S: 保留参数

P： 待检测图像
F： 过渡矩阵
W_E： 水印提取数据

检测和验证过程：

基于保留参数库（**a, U1,V1,S,W**)判断W_E是否水印、是什么水印

是否需要遍历保留参数库？？？

```
AA=imread('A.jpg'); %原图
WW=imread('test.jpg');%水印
A(:,:)=AA(:,:,1);
W=rgb2gray(WW);
SW=size(W);
figure(1),imagesc(AA);
[U,S,V]=svd(double(A));
SS=size(S);
WWW=zeros(SS(1),SS(2));
WWW(1:SW(1),1:SW(2))=W(:,:);
figure(2),imagesc(WWW);
a=0.1;
%嵌入数字水印
L=a*double(WWW)+S;
[U1,S1,V1]=svd(double(L));
AW=U*S1*V';
AA(:,:,1)=AW(:,:);
figure(3),imagesc(AA);
%提取数字水印
[Up,Sp,Vp]=svd(double(AW));
F=U1*Sp*V1';
WE=(F-S)/a;
figure(4),imagesc(WE);
```

test6.m

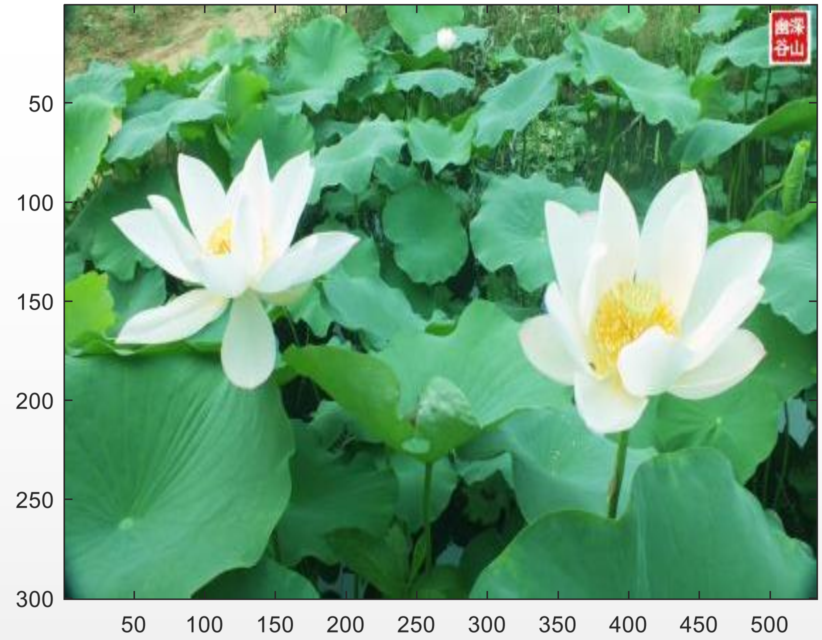


图5. 原图

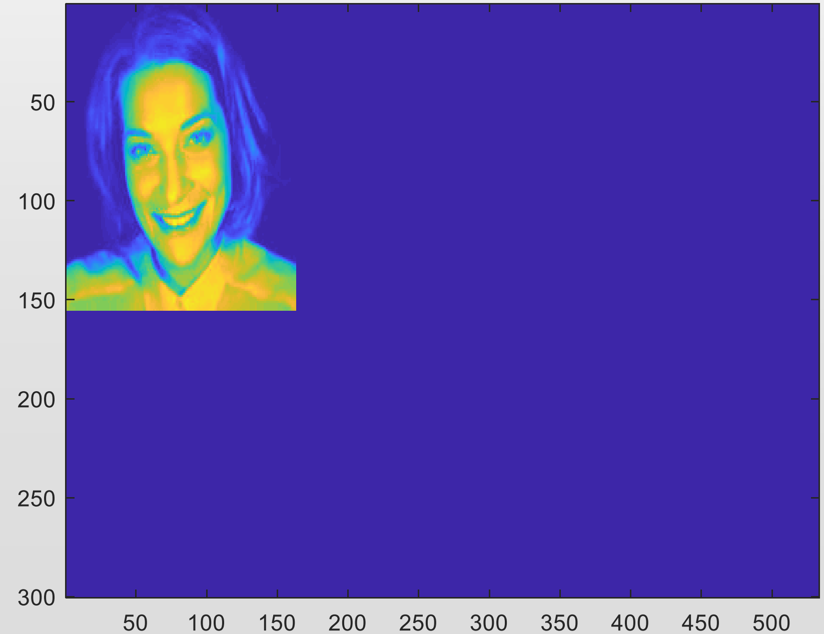


图6. 水印图像及其摆放位置

图7. 含有水印的图像，a=0.1

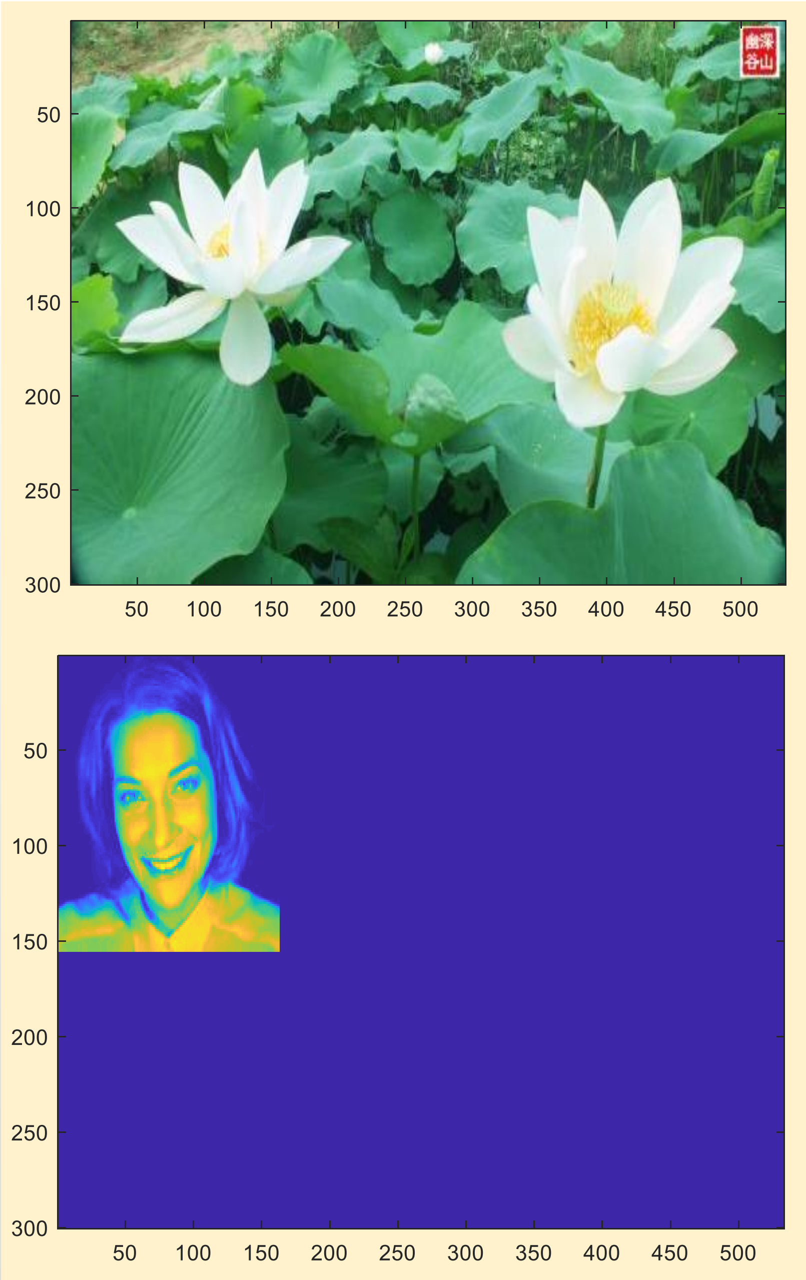
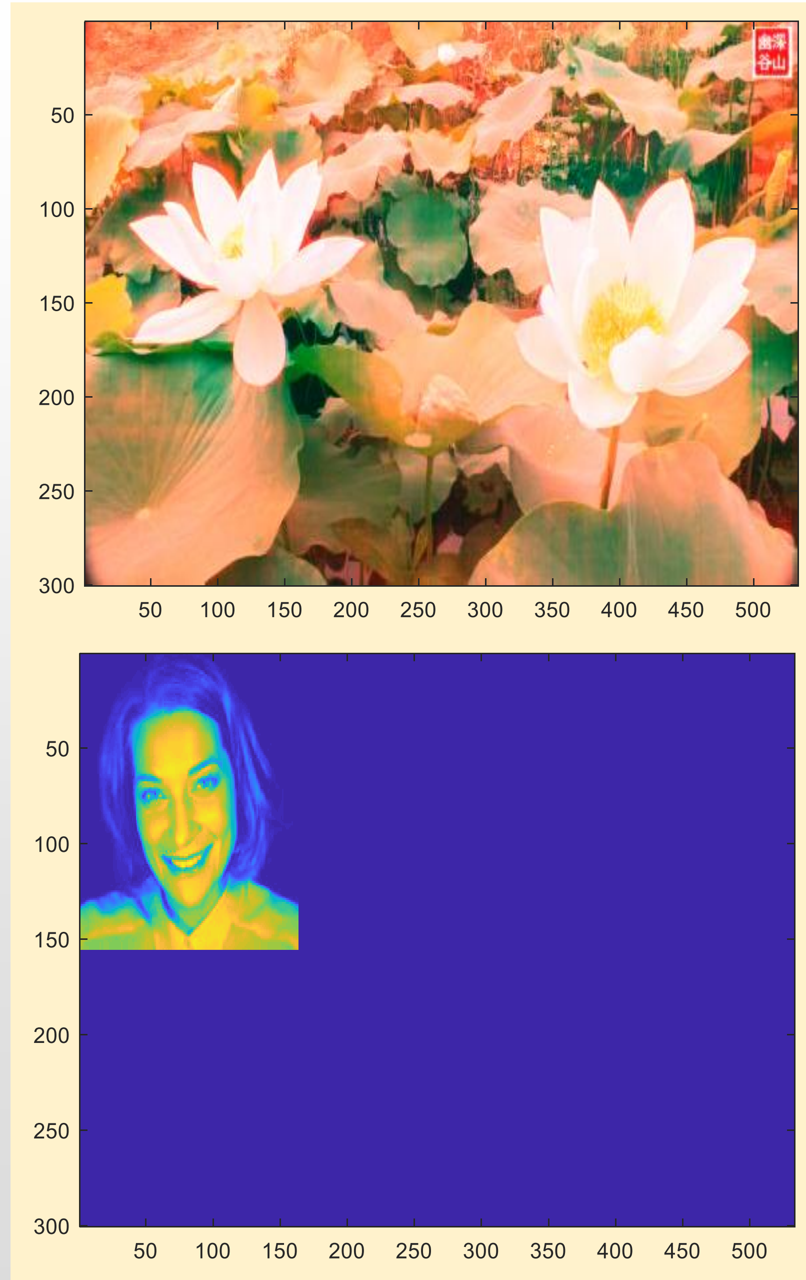


图8 含有水印的图像，a=10.1



现象：水印强度非常小的时候 也能检测出来

案例6： 数据的主成分分析

pca

princomp(Matlab2006a Before)

Principal component analysis of raw data

Syntax

coeff = pca(X)
coeff = pca(X,Name,Value)
[coeff,score,latent] = pca(___)
[coeff,score,latent,tsquared] = pca(___)
[coeff,score,latent,tsquared,explained,mu] = pca(___)

Matlab2012

Description

`coeff = pca(X)` returns the principal component coefficients, also known as loadings, for the n-by-p data matrix `X`. Rows of `X` correspond to observations and columns correspond to variables. The coefficient matrix is p-by-p. Each column of `coeff` contains coefficients for one principal component, and the columns are in descending order of component variance. By default, `pca` centers the data and uses the singular value decomposition (SVD) algorithm.

`coeff = pca(X,Name,Value)` returns any of the output arguments in the previous syntaxes using additional options for computation and handling of special data types, specified by one or more `Name,Value` pair arguments.

For example, you can specify the number of principal components `pca` returns or an algorithm other than SVD to use.

`[coeff,score,latent] = pca( ___ )` also returns the principal component scores in `score` and the principal component variances in `latent`. You can use any of the input arguments in the previous syntaxes.

Principal component scores are the representations of `X` in the principal component space. Rows of `score` correspond to observations, and columns correspond to components.

The principal component variances are the eigenvalues of the covariance matrix of `X`.

`[coeff,score,latent,tsquared] = pca( ___ )` also returns the Hotelling's T-squared statistic for each observation in `X`.

`[coeff,score,latent,tsquared,explained,mu] = pca( ___ )` also returns `explained`, the percentage of the total variance explained by each principal component and `mu`, the estimated mean of each variable in `X`.

Input Arguments

X:数据集. 若有n个样本, 每个样本是p维行向量, 则X是n-by-p的矩阵.

Name-Value Pair Arguments：

Example: 'Algorithm','eig','Centered',false,'Rows','all','NumComponents',3 specifies that pca uses eigenvalue decomposition algorithm, not center the data, use all of the observations, and return only the first three principal components.

'Algorithm' — Principal component algorithm

'svd' (default) | 'eig' | 'als'

'Centered' — Indicator for centering columns

true (default) | false

'Economy' — Indicator for economy size output

true (default) | false

'NumComponents' — Number of components requested

number of variables (default) | scalar integer

'Rows' — Action to take for NaN values

'complete' (default) | 'pairwise' | 'all'

'Weights' — Observation weights

ones (default) | row vector

'VariableWeights' — Variable weights

row vector | 'variance'

'Coeff0' — Initial value for coefficients

matrix of random values (default) | p-by-k matrix

'Score0' — Initial value for scores

matrix of random values (default) | k-by-m matrix

'Options' — Options for iterations

structure

```
load hald;% hald是Matlab自带的数据集之一
%用Matlab函数pca( )进行主成分分析
[coeff,score,latent,tsquared,explained] = pca(ingredients,'Algorithm','svd','Centered',true)
disp('--用Matlab函数pca( )进行主成分分析---结束');
%自编程序验证pca()
S=size(ingredients);%求数据维数
EX=zeros(1,S(2)); %用于存放数据期望值
%（1）求期望值
for II=1:S(1)
    EX=EX+ingredients(II,:);
end
EX=EX/S(1);
%（2）求协方差矩阵
DX=zeros(S(2),S(2));
for II=1:S(1)
    DX=DX+(ingredients(II,:)-EX)'*(ingredients(II,:)-EX);
end
DX=DX/(S(1)-1);
disp(EX);disp(DX);
%（3）特征值分解或奇异值分解
[U,SS,V]=svd(DX)
%[U,SS]=eig(DX)
Y=U*(ingredients'-EX'*ones(1,S(1)));%Hotelling 变换
disp(Y');disp('---自编程序验证pca()结束----');
```

X：原始数据，列表示一个变量，每行表示一个个案.

coeff：Hotelling变换矩阵(U).

score：Hotelling变换结果(Y).

latent：各主成分对应的奇异(特征)值.

tsquared：Hotelling T^2统计量.

explained：成分比重.

Definition 1: The Hotelling's T-square test statistic is

$$T^2 = n(\bar{X} - \mu^0)^T S^{-1}(\bar{X} - \mu^0)$$

where S is the covariance matrix of the sample for X, $\bar{X}$ is the mean of the sample, and where the sample for each random variable x_i in X has n elements.

test7.m

coeff =

-0.0678	-0.6460	0.5673	0.5062
-0.6785	-0.0200	-0.5440	0.4933
0.0290	0.7553	0.4036	0.5156
0.7309	-0.1085	-0.4684	0.4844

score =

36.8218	-6.8709	-4.5909	0.3967
29.6073	4.6109	-2.2476	-0.3958
-12.9818	-4.2049	0.9022	-1.1261
23.7147	-6.6341	1.8547	-0.3786
-0.5532	-4.4617	-6.0874	0.1424
-10.8125	-3.6466	0.9130	-0.1350
-32.5882	8.9798	-1.6063	0.0818
22.6064	10.7259	3.2365	0.3243
-9.2626	8.9854	-0.0169	-0.5437
-3.2840	-14.1573	7.0465	0.3405
9.2200	12.3861	3.4283	0.4352
-25.5849	-2.7817	-0.3867	0.4468
-26.9032	-2.9310	-2.4455	0.4116

latent =

517.7969
67.4964
12.4054
0.2372

U =

-0.0678	0.6460	-0.5673	0.5062
-0.6785	0.0200	0.5440	0.4933
0.0290	-0.7553	-0.4036	0.5156
0.7309	0.1085	0.4684	0.4844

SS =

517.7969	0	0	0
0	67.4964	0	0
0	0	12.4054	0
0	0	0	0.2372

Y =

36.8218	6.8709	4.5909	0.3967
29.6073	-4.6109	2.2476	-0.3958
-12.9818	4.2049	-0.9022	-1.1261
23.7147	6.6341	-1.8547	-0.3786
-0.5532	4.4617	6.0874	0.1424
-10.8125	3.6466	-0.9130	-0.1350
-32.5882	-8.9798	1.6063	0.0818
22.6064	-10.7259	-3.2365	0.3243
-9.2626	-8.9854	0.0169	-0.5437
-3.2840	14.1573	-7.0465	0.3405
9.2200	-12.3861	-3.4283	0.4352
-25.5849	2.7817	0.3867	0.4468
-26.9032	2.9310	2.4455	0.4116

(以上数据仅供参考，请运行行程序进行体验)

```
[U,SS,V]=svd(DX);
%[U,SS]=eig(DX)
Y=U'*(ingredients'-EX'*ones(1,S(1)));%Hotelling 变换
%coeff
disp('Coeff=');
disp(U);
%latent
disp('Latent=');
disp(diag(SS));
%score
disp('Score=');
disp(Y');
Explained=diag(SS);
Explained=100*Explained/sum(Explained);
%explained
disp('Explained=');
disp(Explained);
S1=DX;
n=S(2);
for II=1:S(1)
    X(II,:)=ingredients(II,:)-EX;
end
Tsquared=X*pinv(S1)*X';%Matlab如此计算tsquared!!!
T2=diag(Tsquared);
%tsquared
disp('Tsquared=');
disp(T2);
```

$$D(\mathbf{m},\mathbf{s}_i)=\sqrt{(\mathbf{s}_i-\mathbf{m})^T C^{-1}(\mathbf{s}_i-\mathbf{m})}$$

Mahalanobis距离

Explained=	explained =
86.5974	86.5974
11.2882	11.2882
2.0747	2.0747
0.0397	0.0397

Tsquared=	tsquared =
5.6803	5.6803
3.0758	3.0758
6.0002	6.0002
2.6198	2.6198
3.3681	3.3681
0.5668	0.5668
3.4818	3.4818
3.9794	3.9794
2.6086	2.6086
7.4818	7.4818
4.1830	4.1830
2.2327	2.2327
2.7216	2.7216

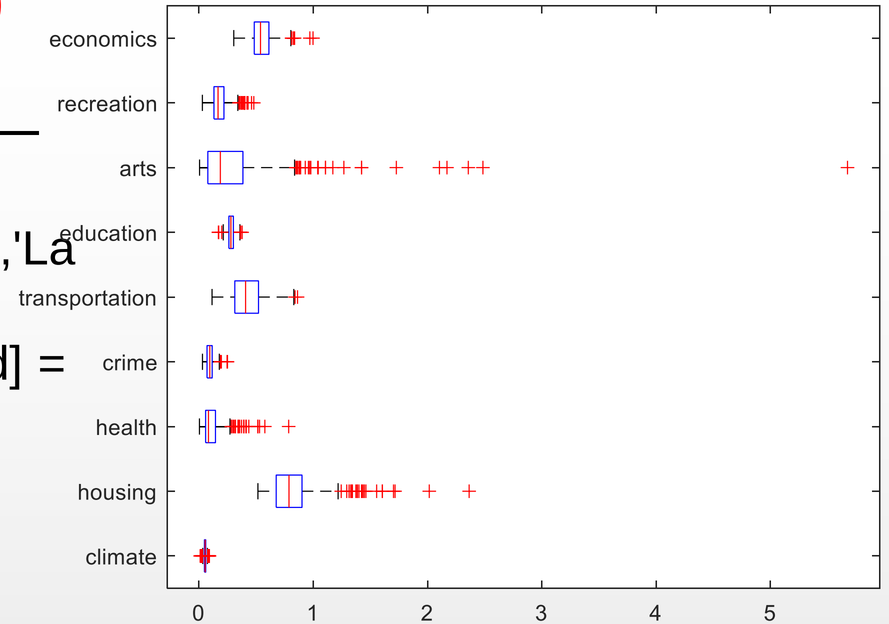
Analyze Quality of Life in U.S. Cities Using PCA
(用主成分分析法分析美国城市的生活质量)

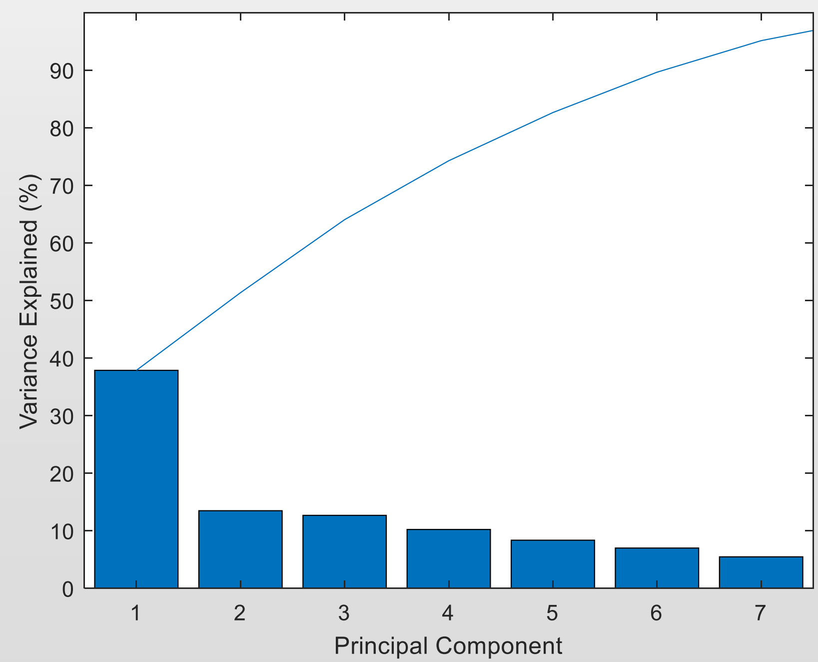
```
load cities%cities是matlab自带的数据集之一
figure()%画各列数据的盒形图
boxplot(ratings,'Orientation','horizontal','Labels',categories)
[wcoeff,score,latent,tsquared,explained] = pca(ratings,... %主成分分析
'VariableWeights','variance');
figure()%各成分比重及累计曲线
pareto(explained)
xlabel('Principal Component')
ylabel('Variance Explained (%)')
[st2,index] = sort(tsquared,'descend'); % sort in descending order
extreme = index(1);
names(extreme,:)
```

test8.m

```
ans =

    'New York, NY'
```





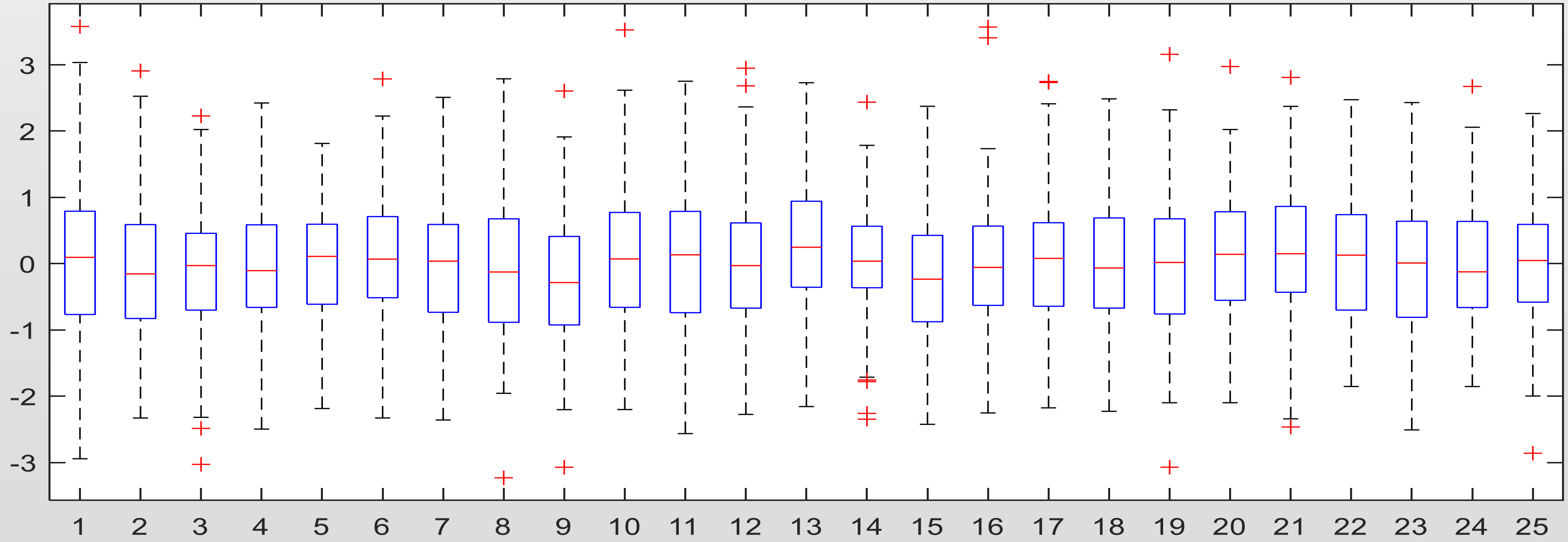
注记：

（1）盒形图中的“盒子”牵扯到六个要素值，请研读boxplot的帮助文档.

- ✓Q1=下四分位数
- ✓Q3=上四分位数
- ✓Median=样本的中值
- ✓最小值：sup{x:x<Q1}-alpha*(Q3-Q1), alpha=1,1.5或其他合适的值
- ✓最大值：inf{x:x>Q3}+alpha*(Q3-Q1), alpha=1,1.5或其他合适的值
- ✓离群值(Outliers)：{x:x<最小值||x>最大值}

（2）Hotelling's T-squared test属于统计学里面的方法之一.若想深入了解，请关注假设检验、统计推断等概念.

boxplot



数据实验

实验1：

1.编程验证Kronecker积与Khatri-Rao积的乘积的关系

$$(A \otimes B)(C * D) = AC * BD$$

2.编写程序，求矩阵方程 $AX + XB + CXD = E$

3.编程实现vec()的逆运算

4.编写程序，求一组数据的多项式**最小二乘拟合，并估算其精度**

以上实验提交截止日期：*月*日

试验2：

5.随机生成10000组数据，对其进行主成分分析

6.修改案例4中的程序，对图像进行次成分分析

以上实验提交截止日期：1月*日

作业:

3.6
3.9
3.29
4.2
4.3
4.7
412
4.13

所有数值计算，可以笔纸推导，也可以使用Python和MATLAB工具.

以上实验提交截止日期：1月*日