Computational Methods in Statistics with Applications
Singular Value Decomposition

Maya Neytcheva, Lars Eldén

SeSe, September 2014

Matrix factorizations/decompositions

Matrix factorizations I

- LU, LDU, $Cholesky\ LDL^T$
- Tridiagonalization $Q^TAQ = T$, A - symmetric
  Aasen's algorithm: $A = LTL^T$
- Bidiagonalization $Q^TAV = B$, $A(m,n)$, B - upper bi-diagonal
- $\dots$
- QR

Golub, Van Loan, *Matrix Computations*, many editions.
Note: Some of the algorithms are not numerically stable.

Matrix factorizations

- Schur decomposition $A = Q * T$

Any real matrix A can be decomposed into a unitary matrix U times an upper triangular matrix T, which has the eigenvalues of A on its diagonal. Note: Eigenvalue-revealing factorization

Matrix factorizations

- Schur decomposition $A = Q * T$

Any real matrix A can be decomposed into a unitary matrix U times an upper triangular matrix T, which has the eigenvalues of A on its diagonal. Note: Eigenvalue-revealing factorization

- Eigenvalue decomposition

A - square. If all its eigenvectors are linearly independent, then $A = QDQ^T$, where Q is orthogonal and D is diagonal, containing the eigenvalues of A.

Matrix factorizations

- Schur decomposition $A = Q * T$

Any real matrix A can be decomposed into a unitary matrix U times an upper triangular matrix T, which has the eigenvalues of A on its diagonal. Note: Eigenvalue-revealing factorization

- Eigenvalue decomposition

A - square. If all its eigenvectors are linearly independent, then $A = QDQ^T$, where Q is orthogonal and D is diagonal, containing the eigenvalues of A.

- Singular value decomposition SVD

Question: Can we diagonalize a general matrix using unitary matrices?

$$Q_1 A Q_2^T = \Sigma$$

Singular value decomposition

SVD

Let $A(m, n)$, $n \leq m$ or $n \geq m$, $rank(A) = rank(A^*)k$.

Definition

If there exist $\mu \neq 0$ and vectors $\mathbf{u}$ and $\mathbf{v}$, such that

$$A\mathbf{v} = \mu\mathbf{u} \quad \text{and} \quad A^*\mathbf{u} = \mu\mathbf{v}$$

then μ is called a singular value of A, and $\mathbf{u}, \mathbf{v}$ are a pair of singular vectors, corresponding to μ.

The existence of singular values and vectors is shown...

via the following construction:

$$A\mathbf{v} = \mu\mathbf{u}, \quad A^*\mathbf{u} = \mu\mathbf{v}$$

can be written as

$$\widetilde{A}\begin{bmatrix}\mathbf{v}\\ \mathbf{u}\end{bmatrix} = \begin{bmatrix}0 & A\\ A^* & 0\end{bmatrix}\begin{bmatrix}\mathbf{v}\\ \mathbf{u}\end{bmatrix} = \mu\begin{bmatrix}\mathbf{v}\\ \mathbf{u}\end{bmatrix}$$

The matrix $\widetilde{A}$ is selfadjoint, has real eigenvalues and a complete eigenvector space.

Furthermore, μ^2 is an eigenvalue of A^*A with eigenvector $\mathbf{u}$ and of AA^* with eigenvector $\mathbf{v}$, because

$$\begin{aligned} A\mathbf{v} = \mu\mathbf{u}, &\quad \rightarrow \quad A^*A\mathbf{v} = \mu A^*\mathbf{u} = \mu^2\mathbf{v}\\ A^*\mathbf{u} = \mu\mathbf{v}, &\quad \rightarrow \quad AA^*\mathbf{u} = \mu A\mathbf{v} = \mu^2\mathbf{u} \end{aligned}$$

Singular Value Decomposition

Theorem (SVD)

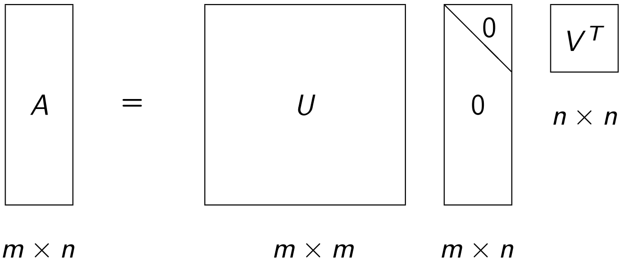
Any $m \times n$ matrix A with dimensions, say, $m \geq n$, can be factorized as

$$A = U \begin{pmatrix} \Sigma \\ 0 \end{pmatrix} V^T,$$

where $U \in \mathbb{R}^{m \times m}$ and $V \in \mathbb{R}^{n \times n}$ are orthogonal, and $\Sigma \in \mathbb{R}^{m \times n}$ is diagonal,

$$\Sigma = \mathrm{diag}(\sigma_1, \sigma_2, \ldots, \sigma_n),$$
$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0.$$

SVD



A

$=$

U

0

0

V^T

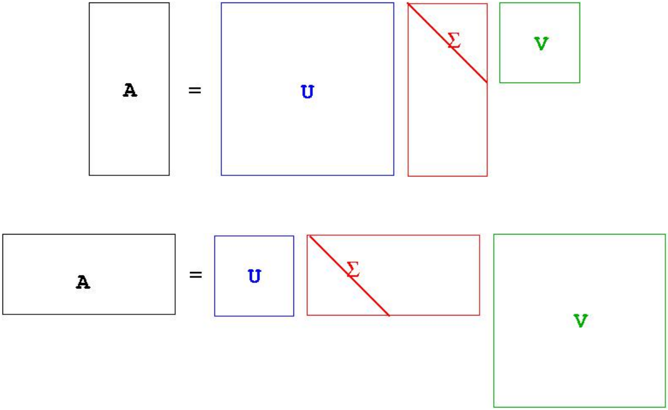
$n \times n$

$m \times n$

$m \times m$

$m \times n$

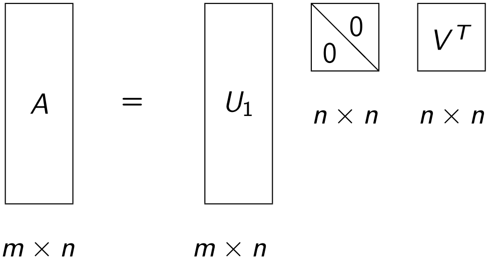
SVD



Thin SVD

Partition $U = (U_1\ U_2)$, where $U_1 \in \mathbb{R}^{m\times n}$,

$$A = U_1 \Sigma V^T,$$

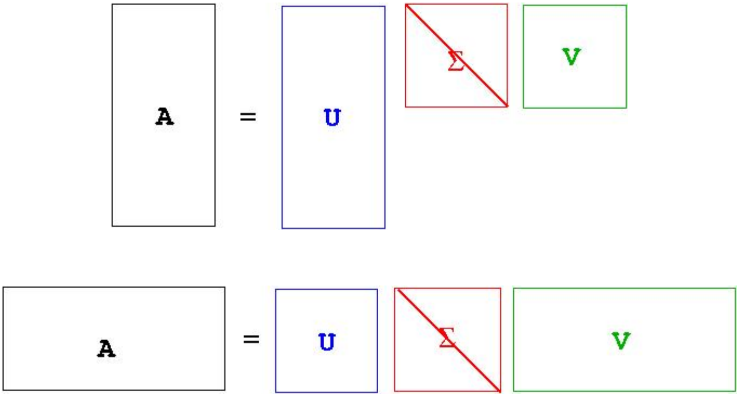


A = U_1 0 0 V^T

$n \times n$ $n \times n$

$m \times n$ $m \times n$

Thin SVD



Fundamental Subspaces I

The *range of the matrix A:*

$$\mathcal{R}(A) = \{y \mid y = Ax, \text{ for arbitrary } x\}.$$

Assume that A has rank r:

$$\sigma_1 \geq \cdots \geq \sigma_r > \sigma_{r+1} = \cdots = \sigma_n = 0.$$

Outer product form:

$$y = Ax = \sum_{i=1}^{r} \sigma_i u_i v_i^T x = \sum_{i=1}^{r} (\sigma_i v_i^T x) u_i = \sum_{i=1}^{r} \alpha_i u_i.$$

Fundamental Subspaces II

The *null-space of the matrix A*:

$$\mathcal{N}(A) = \{x \mid Ax = 0\}.$$

$$Ax = \sum_{i=1}^{r} \sigma_i u_i v_i^T x$$

Any vector $z = \sum_{i=r+1}^{n} \beta_i v_i$ is in the null-space:

$$Az = (\sum_{i=1}^{r} \sigma_i u_i v_i^T)(\sum_{i=r+1}^{n} \beta_i v_i) = 0.$$

Fundamental Subspaces

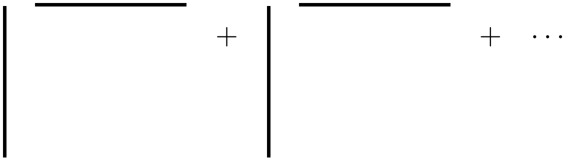
Theorem (Fundamental subspaces)

1. *The singular vectors $u_1, u_2, \dots, u_r$ are an orthonormal basis in $\mathcal{R}(A)$ and*
$$\mathrm{rank}(A) = \dim(\mathcal{R}(A)) = r.$$
2. *The singular vectors $v_{r+1}, v_{r+2}, \dots, v_n$ are an orthonormal basis in $\mathcal{N}(A)$ and*
$$\dim(\mathcal{N}(A)) = n - r.$$
3. *The singular vectors $v_1, v_2, \dots, v_r$ are an orthonormal basis in $\mathcal{R}(A^T)$.*
4. *The singular vectors $u_{r+1}, u_{r+2}, \dots, u_m$ are an orthonormal basis in $\mathcal{N}(A^T)$.*

SVD matrix expansion

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

$$A = \sum_{i=1}^{n} \sigma_i u_i {v_i}^T =$$



SVD of a matrix with full column rank I

```
A =   1      1
      1      2
      1      3
      1      4
```

```
>> [U,S,V]=svd(A)
```

SVD of a matrix with full column rank II

```
U = 0.2195   -0.8073    0.0236    0.5472
    0.3833   -0.3912   -0.4393   -0.7120
    0.5472    0.0249    0.8079   -0.2176
    0.7110    0.4410   -0.3921    0.3824

 S = 5.7794         0           V = 0.3220 -0.9467
          0    0.7738               0.9467  0.3220
          0         0
          0         0
```

Thin SVD

```
>> [U,S,V]=svd(A,0)

U = 0.2195   -0.8073
    0.3833   -0.3912
    0.5472    0.0249
    0.7110    0.4410

S = 5.7794         0
         0    0.7738

V = 0.3220   -0.9467
    0.9467    0.3220
```

Rank deficient matrix I

```
>> A(:,3)=A(:,1)+0.5*A(:,2)

A = 1.0000    1.0000    1.5000
    1.0000    2.0000    2.0000
    1.0000    3.0000    2.5000
    1.0000    4.0000    3.0000

>> [U,S,V]=svd(A,0)

U = 0.2612   -0.7948   -0.5000
    0.4032   -0.3708    0.8333
    0.5451    0.0533   -0.1667
    0.6871    0.4774   -0.1667
```

Rank deficient matrix II

```
S = 7.3944         0         0
         0    0.9072         0
         0         0         0

V = 0.2565   -0.6998    0.6667
    0.7372    0.5877    0.3333
    0.6251   -0.4060   -0.6667
```

SVD is rank-revealing!

Null Space

The third column of V is a basis vector in $N(A)$:

```
>> A*V(:,3)

ans =
    1.0e-15 *
         0
   -0.2220
   -0.2220
         0
```

Historical notes

SVD has many different names:

- First derivation of the SVD by Eugenio Beltrami (1873)
- Full proof by Camille Jordan (1874)
- James Joseph Sylvester (1889), independently discovers SVD
- Erhard Schmidt (1907), first to derive an optimal, low-rank approximation of a larger problem
- Hermann Weyl (1912) - determination of the rank in the presence of errors
- Eckart-Young decomposition and optimality properties of SVD (1936), psychometrics
- Numerically efficient algorithms to compute the SVD - works by Gene Golub 1970 (Golub-Kahan)

Best approximation / Eckart-Young Property I

Frobenius norm

$$\|A\|_F = \left(\sum_{i,j} a_{ij}^2 \right)^{1/2}$$

Best approximation / Eckart-Young Property II

Theorem

Assume that the matrix $A \in \mathbb{R}^{m\times n}$ has rank r and choose k, such that $r > k$. The Frobenius norm matrix approximation problem

$$\min_{\mathrm{rank}(Z)=k} \|A - Z\|_F$$

has the solution

$$Z = A_k = U_k \Sigma_k V_k^T,$$

where $U_k = (u_1, \ldots, u_k)$, $V_k = (v_1, \ldots, v_k)$, and $\Sigma_k = \mathrm{diag}(\sigma_1, \ldots, \sigma_k)$.

Best approximation / Eckart-Young Property III

Proof:

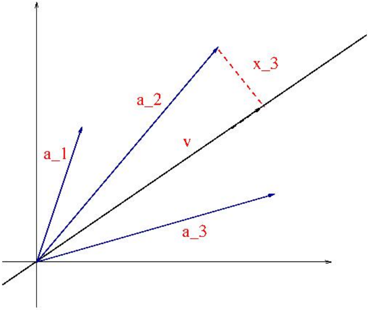
(1) Observe: if $A_k = \sum_{j=1}^{k} \sigma_j u_j v_j^*$, then $\|A - A_k\| = \sigma_{k+1}$.

(2) Observe: Consider the subspace, spanned by the first $k+1$ singular vectors of A, W. Then, $\|Aw\|_2 \geq \sigma k + 1\|w\|_2, w \in W$.

(3) Assume that there exists a matrix B of rank k, such that $\|A - B\|_2 < \sigma_{k+1}$. Then, there exists a subspace $\widehat{W}$ of size $n-k$, such that $Bw = 0, w \in \widehat{W}$.

$\|Aw\|_2 = \|(A-B)w\|_2 \leq \|A - B\|_2\|w\|_2 \leq \sigma_{k+1}\|w\|_2$. From dinemsion argiments $W \cap \widehat{W} \neq \emptyset$.

Singular vectors, another view



v+x_3=a_3

$\underbrace{(v,v)}_{||v||\hat{}2}+\underbrace{(v,x\_3)}_{=0}=(v,a\_3)$

Consider the rows of $A(m,n)$ as points in an n-dimensional space and find the best linear fit through the origin.

$$\mathbf{v}_1 = arg \max_{\|\mathbf{v}\|=1} \|A\mathbf{v}\|_2^2, \ \sigma_1 = \|A\mathbf{v}_1\|_2$$

$$\mathbf{v}_2 = arg \max_{\|\mathbf{v}\|=1, \mathbf{v}\perp\mathbf{v_1}} \|A\mathbf{v}\|_2^2$$

Principal Component Analysis (PCA) I

Data matrix $\mathbb{R}^{m \times n} \ni X = U\Sigma V^T$

Each column of X is an observation of a real-valued random vector with mean zero.

The right singular vectors v_i are called *principal components directions* of X. The vector

$$z_1 = Xv_1 = \sigma_1 u_1$$

has the largest sample variance amongst all normalized linear combinations of the columns of X:

$$\mathrm{Var}(z_1) = \mathrm{Var}(Xv_1) = \frac{\sigma_1^2}{m}.$$

Principal Component Analysis (PCA) II

The normalized variable u_1 is called the *normalized first principal component* of X.

The second principal component is the vector of largest sample variance of the deflated data matrix $X - \sigma_1 u_1 v_1^T$, and so on.

Test example borrowed from

Computational Statistics with Application to Bioinformatics

Prof. William H. Press Spring Term, 2008, The University of Texas at Austin

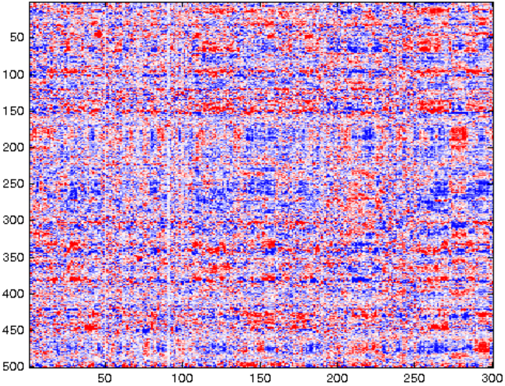
Example

Consider some gene expression data, represented by the so-called 'design matrix' $X = \{X_{ij}\}$
Each column of X corresponds to a separate observation, in this case, a separate micro array experiment under a different condition.
N rows are genes (1:500) and M columns are the corresponding responses.
Assumptions:
- the individual experiments (columns of X) have zero mean.
- scale data to unit standard deviation.

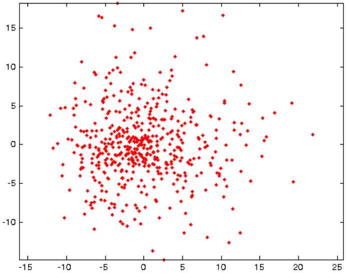
```
load yeastarray_t2.txt;
size(yeastarray_t2)

ans = 500 300
yclip = prctile(yeastarray_t2(:),[1,99])
yclip = -204 244

data = max(yclip(1),min(yclip(2),yeastarray_t2));
dmean= mean(data,1);
dstd = std(data,1);
data = (data - repmat(dmean,[size(data,1),1]))./...
               repmat(dstd,[size(data,1),1]);
genecolormap = [min(1,(1:64)/32); 1-abs(1-(1:64)/32);
                min(1,(64-(1:64))/32)]';
figure(1),clf,colormap(genecolormap);
image(20*data+32)
```

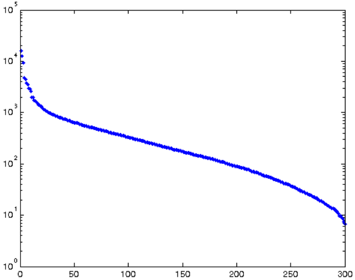

50
100
150
200
250
300
350
400
450
500
50
100
150
200
250
300

```
[U S V] = svd(data, 0);
PCAcoords = U*S;
plot(PCAcoords(:,1),PCScoords(:,2),'r.')
axis equal
```



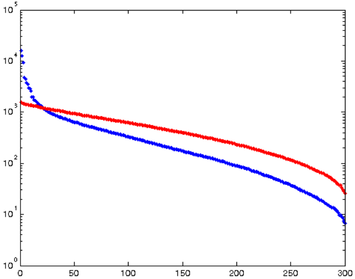
The squares of the singular values are proportional to the portion of the total variance (L_2 norm of X) that each accounts for.

```
ssq = diag(S).^2;
semilogy(ssq,'.b')
```



We can produce fake data and compare:

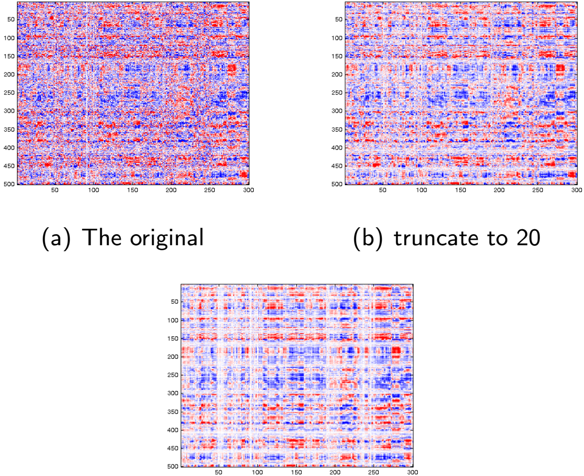
```
fakedata = randn(500,300);
[Uf Sf Vf] = svd(fakedata,0);
sfsq = diag(Sf).^2;
semilogy(sfsq,’.r’)
```



For the data in this example, a sensible use of PCA (i.e., SVD) would be to project the data into the subspace of the first 20 SVs, where we can be sure that it is not noise.

```
% Truncate the first 20 singular values/vectors
strunc = diag(S);
strunc(21:end) = 0;
filtdata20 = U*diag(strunc)*V’;
figure(2),clf,colormap(genecolormap);
image(20*filtdata20+32)

% Truncate the first 5 singular values/vectors
strunc(6:end) = 0;
filtdata5 = U*diag(strunc)*V’;
figure(3),clf,colormap(genecolormap);
image(20*filtdata5+32)
```

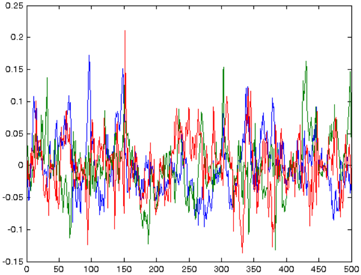


(a) The original

(b) truncate to 20

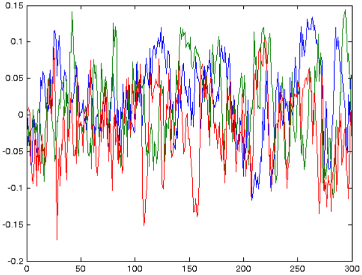
How to interpret the singular vectors? The first three vectors u are 'eigengenes', the linear combination of genes that explain the most data.

```
plot(U(:,1:3))
```



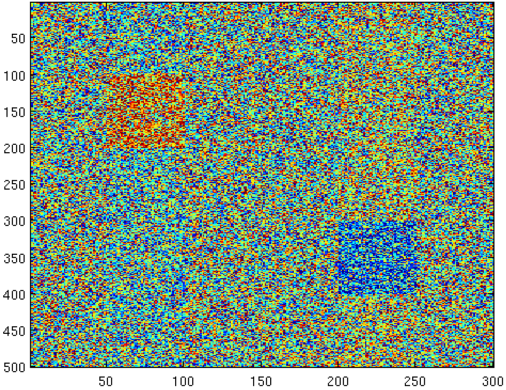
The first three vectors v are 'eigenarrays', the linear combination of experiments that explain the most data.

```
plot(V(:,1:3))
```



Consider a toy example

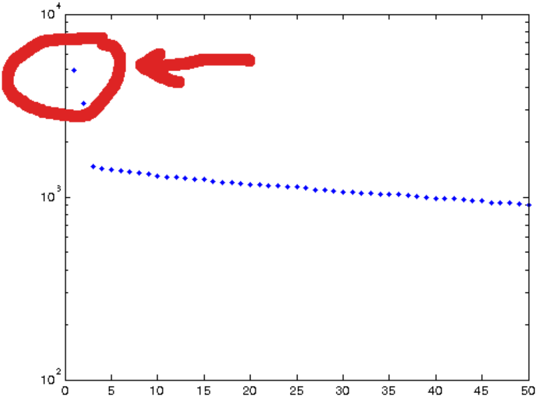
```
pdata = randn(500,300);
pdata(101:200,51:100) = pdata(101:200,51:100) + 1;
pdata(301:400,201:250) = pdata(301:400,201:250) - 1;
pmean = mean(pdata,1);
pstd = std(pdata,1);
pdata = (pdata - repmat(pmean,[size(pdata,1),1]))./...
                 repmat(pstd,[size(pdata,1),1]);
colormap(genecolormap)
image(20*pdata+32)
```


50
100
150
200
250
300
350
400
450
500
50
100
150
200
250
300

Consider a toy example

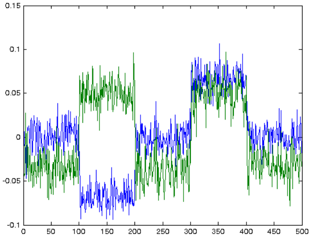
```
[Up Sp Vp] = svd(pdata,0);
spsq = diag(Sp).^2;
semilogy(spsq(1:50),’.b’)
```

Should we expect the eigengenes/eigenarrays to show the separate main effects?

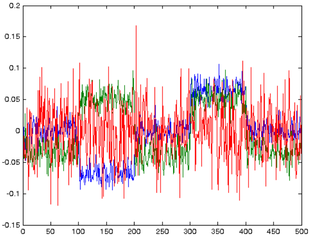


10^4
10^3
10^2
0
5
10
15
20
25
30
35
40
45
50

```
plot(Up(:,1:2)),
plot(Up(:,1:3))
```

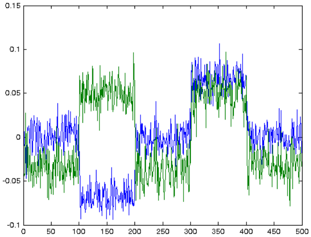


(d)

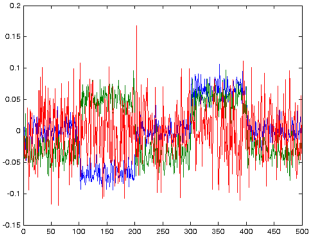


(e)

```
plot(Vp(:,1:2)),
plot(Vp(:,1:3))
```



(f)



(g)

Solving Least Squares problems by SVD

$$A\mathbf{x} = \mathbf{b}, A(m, n)$$

$$A = U\Sigma V$$

$$U\Sigma V\mathbf{x} = \mathbf{b} \ \rightarrow \mathbf{x} = V(\Sigma^{-1}(U^T\mathbf{b}))$$

Least Squares by SVD I

```
A =   1      1                b = 7.9700
      1      2                   10.2000
      1      3                   14.2000
      1      4                   16.0000
      1      5                   21.2000

>> [U1,S,V]=svd(A,0)

U1 =0.1600   -0.7579
    0.2853   -0.4675
    0.4106   -0.1772
    0.5359    0.1131
    0.6612    0.4035
```

Least Squares by SVD II

```
S = 7.6912    0          V = 0.2669  -0.9637
     0    0.9194             0.9637   0.2669

      >> x=V*(S\(U1'*b))

      x = 4.2360
          3.2260
```

Least Squares by SVD, in R I

```
> A.svd<-svd(A)
> A.svd
$d
[1] 7.6912131 0.9193696

$u
          [,1]       [,2]
[1,] 0.1600071  0.7578903
[2,] 0.2853078  0.4675462
[3,] 0.4106086  0.1772020
[4,] 0.5359094 -0.1131421
[5,] 0.6612102 -0.4034862
```

Least Squares by SVD, in R II

```
$v
          [,1]       [,2]
[1,] 0.2669336  0.9637149
[2,] 0.9637149 -0.2669336

> x=A.svd$v %*% diag(1/A.svd$d) %*% t(A.svd$u) %*%
> x
      [,1]
[1,] 4.236
[2,] 3.226
```

Linear dependence – SVD

Theorem

Let the singular values of A satisfy

$$\sigma_1 \geq \cdots \geq \sigma_r > \sigma_{r+1} = \cdots = \sigma_n = 0.$$

Then the rank of A is equal to r.

Rank $=$ the number of linearly independent columns of A.

Linear dependence I

```
A=[1 1; 1 2; 1 3; 1 4]
singval=svd(A)

% Third col=linear combination of first two
A1=[A A(:,1)+0.5*A(:,2)]
singval1=svd(A1)
```

Linear dependence II

Result:

```
A =  1     1         singval = 5.7794
     1     2                   0.7738
     1     3
     1     4

A1 = 1.0000    1.0000    1.5000
     1.0000    2.0000    2.0000
     1.0000    3.0000    2.5000
     1.0000    4.0000    3.0000

singval1 = 7.3944
           0.9072
                0
```

Almost linear dependence I

```
A2=[A A(:,1)+0.5*A(:,2)+0.0001*randn(4,1)]
singval2=svd(A2)

--------------------------------------

A2 = 1.0000    1.0000    1.4999
     1.0000    2.0000    2.0001
     1.0000    3.0000    2.5000
     1.0000    4.0000    3.0001

singval2 = 7.3944
           0.9072
           0.0001
```

Almost linear dependence? I

Run Matlab demo

`~/.../STAT/Labs/Lab_QR_SVD/Small_singular_values.m`

Computing the SVD in a numerically efficient way

Computing the SVD

1. Transform A to bidiagonal form by unitary transformations

$$Q_L A Q_R = B = \begin{bmatrix} * & * & & & \\ & * & * & & \\ & ddots & \ddots & * & \\ & & & & * \end{bmatrix}$$

2. Diagonalize B by two orthogonal transformations

$$\widetilde{Q}_L B \widetilde{Q}_R = \widetilde{Q}_L Q_L A Q_R \widetilde{Q}_R = \Sigma$$

The cost for the bidiagonalization is $4mn^2 - 4/3n^3$.
The cost for SVD: $4m^2n + 8mn^2 + 9n^3$.