10. Matrices and linear algebra II

- Eigenvectors and eigenvalues
- Singular value decomposition
- Rank of a matrix
- Low-rank approximation of matrices

Eigenvalues and eigenvectors

- `[V,D]=eig(A)`: calculates eigenvectors and eigenvalues of a square matrix
  - Eigenvalues are stored in ascending order in a diagonal matrix

$$\underbrace{\mathbf{A}\mathbf{v}_i}_{\text{eigenvector}} = \underbrace{d_i\mathbf{v}_i}_{\text{eigenvalue}} \iff \mathbf{AV} = \mathbf{VD}$$

$$\mathbf{V} = \begin{bmatrix}\mathbf{v}_1 \ \mathbf{v}_2 \ \cdots \ \mathbf{v}_n\end{bmatrix} \qquad \mathbf{D} = \begin{bmatrix} d_1 & & & \\ & d_2 & & \\ & & \ddots & \\ & & & d_n \end{bmatrix}$$

```
>> A=randn(3,3);
>> [V,D]=eig(A)
V =
   0.52988 + 0.00000i  -0.05375 - 0.34548i  -0.05375 + 0.34548i
   0.68932 + 0.00000i   0.84473 + 0.00000i   0.84473 - 0.00000i
   0.49404 + 0.00000i   0.12431 + 0.38565i   0.12431 - 0.38565i
D =
Diagonal Matrix
   0.04533 + 0.00000i                    0                    0
                    0   2.09047 + 1.25277i                    0
                    0                    0   2.09047 - 1.25277i
>> A*V-V*D
ans =
   1.6306e-16 + 0.0000e+00i  -1.1102e-15 - 2.2204e-15i  -1.1102e-15 + 2.2204e-15i
  -1.5959e-16 + 0.0000e+00i   0.0000e+00 + 4.4409e-16i   0.0000e+00 - 4.4409e-16i
  -2.6368e-16 + 0.0000e+00i  -1.1102e-16 + 3.3307e-16i  -1.1102e-16 - 3.3307e-16i
```

Eigenvectors/values of symmetric matrices

- Symmetric matrices always have *real* eigenvectors/values
  - Nonsymmetric matrices have complex eigenvectors/values in general as in the last slide
  - Many matrices we encounter in engineering will be symmetric
  - Eigenvectors of symmetric matrices are orthogonal, i.e., $\mathbf{V}^\top\mathbf{V} = \mathbf{V}\mathbf{V}^\top = \mathbf{I}$
  - The symmetric matrix is 'diagonalized' by V as $\mathbf{V}^\top\mathbf{A}\mathbf{V} = \mathbf{D}$

```
>> X = randn(3,3);
>> A=X'*X;
>> [V,D]=eig(A)
V =
   0.960179   0.267639   0.080159
  -0.226697   0.914040  -0.336363
  -0.163292   0.304796   0.938315

D =
Diagonal Matrix
   0.015584          0          0
          0   1.752624          0
          0          0   6.254892
```

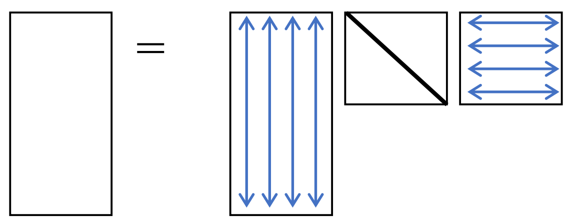
```
>> A*V-V*D
ans =
   1.2143e-17  -2.7756e-16   3.3307e-16
   9.1507e-17   0.0000e+00  -4.4409e-16
  -1.5179e-16   0.0000e+00   0.0000e+00

>> V'*A*V
ans =
   1.5584e-02   4.2718e-17  -1.7391e-16
  -1.3878e-17   1.7526e+00   2.2204e-16
  -2.2204e-16   4.4409e-16   6.2549e+00
```

Singular value decomposition of matrices (1/2)

- Any $m \times n$ real matrix can be decomposed into a product of orthogonal matrices U and V and a diagonal matrix W as follows:

$$\mathbf{X} = \mathbf{U}\mathbf{W}\mathbf{V}^\top$$



singular values

$$\mathbf{W} = \begin{bmatrix} w_1 & & & \\ & w_2 & & \\ & & \ddots & \\ & & & w_n \end{bmatrix}$$

$$\mathbf{U}^\top\mathbf{U} = \mathbf{I} \qquad \mathbf{V}^\top\mathbf{V} = \mathbf{V}\mathbf{V}^\top = \mathbf{I}$$

- Remark: The decomposition is *unique* when we fix the order of the singular values (say, in descending order)

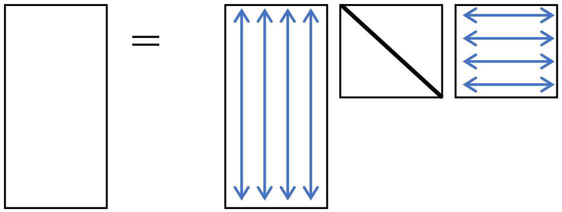
Singular value decomposition of matrices (2/2)

- `svd`: calculates singular value decomposition
  - Singular value decomposition is often abbreviated as SVD

Economy-size decomposition

```
>> X=randn(5,3);
>> [U,W,V]=svd(X,“econ“);
>> W
W =
Diagonal Matrix
   3.07321         0         0
         0   1.73673         0
         0         0   0.82822
```

$$\mathbf{X} = \mathbf{U}\mathbf{W}\mathbf{V}^{\top}$$



```
>> norm(U*W*V'-X)
ans =    1.5822e-15
>> norm(U'*U-eye(3))
ans =    6.7963e-16
>> norm(V'*V-eye(3))
ans =    2.3629e-16
```

Relation between SVD and eigenproblem

- Column vectors of V of SVD of X coincides with eigenvecotrs of A=X'X

$$
\begin{aligned}
\mathbf{A} &= \mathbf{X}^\top \mathbf{X} \\
&= (\mathbf{U}\mathbf{W}\mathbf{V}^\top)^\top (\mathbf{U}\mathbf{W}\mathbf{V}^\top) \\
&= \mathbf{V}\mathbf{W}^\top \mathbf{U}^\top \mathbf{U}\mathbf{W}\mathbf{V}^\top \\
&= \mathbf{V}\mathbf{W}^2\mathbf{V}^\top
\end{aligned}
$$

```
>> X=randn(10,3);
>> [V1,D]=eig(X'*X);
>> [U,W,V2]=svd(X, "econ")
>> V1
V1 =
 -0.69324  -0.61974  -0.36789
  0.14611   0.37901  -0.91379
  0.70574  -0.68722  -0.17219

>> V2
V2 =
  0.36789   0.61974   0.69324
  0.91379  -0.37901  -0.14611
  0.17219   0.68722  -0.70574
```

- Singular values of X are equal to the square roots of eigenvalues of A=X'X

```
>> sqrt(D)
ans =
Diagonal Matrix
   1.9851        0        0
        0   3.9417        0
        0        0   4.7675
```

```
>> W
W =
Diagonal Matrix
   4.7675        0        0
        0   3.9417        0
        0        0   1.9851
```

Properties of SVD

- Pseudo inverse of X can be written using its SVD as

$$\mathbf{X} = \mathbf{U}\mathbf{W}\mathbf{V}^\top$$

$$\downarrow$$

$$\mathbf{X}^\dagger = \mathbf{V}\mathbf{W}^{-1}\mathbf{U}^\top$$

```
>> X=randn(5,3);
>> pinv(X)
ans =
   0.037163  -0.070115  -0.329386   0.373801  -0.323277
  -0.116157  -0.215984  -0.339899  -0.014697  -0.207555
  -0.066574  -0.156025   0.513828   0.023533  -0.501137

>> [U,W,V]=svd(X, "econ");
>> V*inv(W)*U'
ans =
   0.037163  -0.070115  -0.329386   0.373801  -0.323277
  -0.116157  -0.215984  -0.339899  -0.014697  -0.207555
  -0.066574  -0.156025   0.513828   0.023533  -0.501137
```

- The number of non-zero singular values of X is called the rank of X

```
>> X=randn(5,2)*randn(2,4)
X =
  -0.065735  -0.053739   1.626185   1.734253
  -0.022809  -0.020869   0.637444   0.672717
   0.151451   0.140834  -4.307108  -4.539055
   0.563733   0.153728  -3.832887  -5.067102
  -0.246376  -0.082560   2.181361   2.705379
>> rank(X)
ans =  2
>> svd(X)
ans =
   9.9100e+00
   6.0159e-01
   2.4902e-16
   1.6489e-17
```

Approximation of matrices by SVD

- Consider the following problem: given a matrix A, we wish to obtain a matrix of a fixed rank *r* that approximates A as accurately as possible
- It can be formulated as a *constrained minimization* problem:

$$\min_{\hat{\mathbf{A}}} \|\mathbf{A} - \hat{\mathbf{A}}\|_F \quad \text{subject to} \quad \text{rank}(\mathbf{A}) = r$$

- Its solution is simply given by SVD of A in the following way:

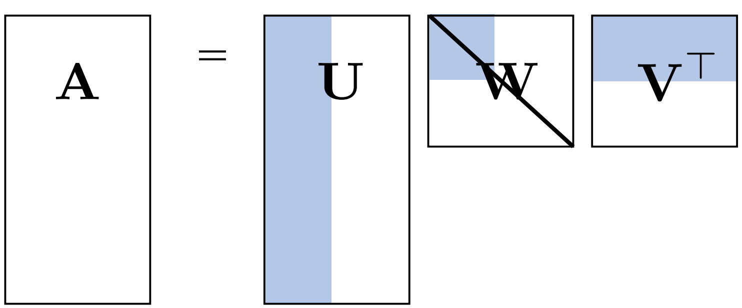
$$\mathbf{A} = \mathbf{U}\mathbf{W}\mathbf{V}^\top$$

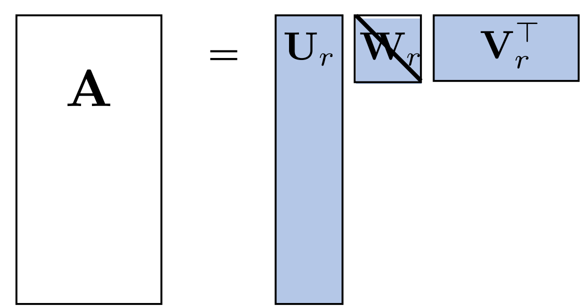
$$\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_r, \ldots, \mathbf{u}_n]$$
$$\mathbf{W} = \text{diag}[w_1, \ldots, w_r, \ldots, w_n]$$
$$\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_r, \ldots, \mathbf{v}_n]$$

Simply remove (r+1)th to nth column vectors

$$\hat{\mathbf{A}} = \mathbf{U}_r\mathbf{W}_r\mathbf{V}_r^\top$$

$$\mathbf{U}_r = [\mathbf{u}_1, \ldots, \mathbf{u}_r]$$
$$\mathbf{W}_r = \text{diag}[w_1, \ldots, w_r]$$
$$\mathbf{V}_r = [\mathbf{v}_1, \ldots, \mathbf{v}_r]$$

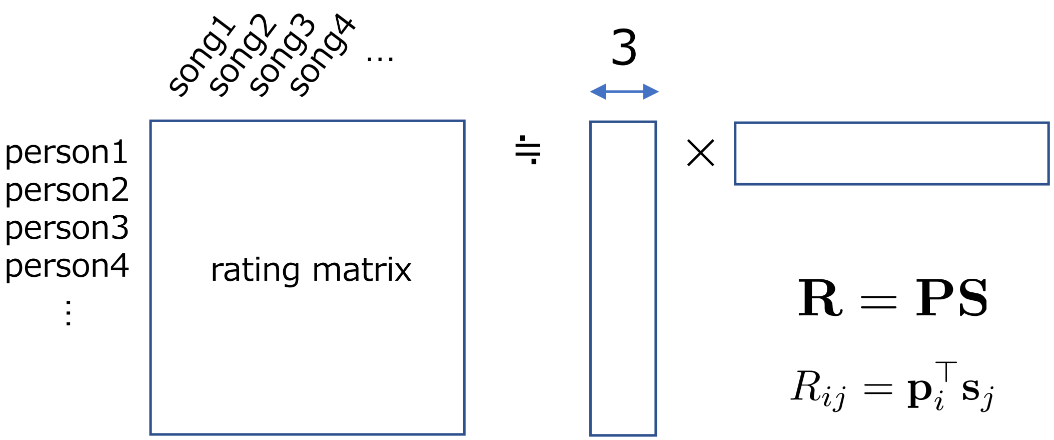




Exercise 10.1

- We wish to predict how a person rates songs

- Some people have similar tastes about like/dislike of music
  - That said, there will be no two persons having exactly the same taste
  - This kind of problems is known as *collaborative filtering*
- We approximate the rating matrix by a matrix of rank=3



$$\mathbf{R} = \mathbf{P}\mathbf{S}$$

$$R_{ij} = \mathbf{p}_i^\top \mathbf{s}_j$$

$$\mathbf{P} = \begin{bmatrix} \mathbf{p}_1^\top \\ \mathbf{p}_2^\top \\ \vdots \\ \mathbf{p}_{15}^\top \end{bmatrix} \quad \mathbf{S} = [\mathbf{s}_1, \mathbf{s}_2, \cdots, \mathbf{s}_{20}]$$

Exercise 10.1

- Ratings of 20 songs by 15 persons are available
  - Download `rating.txt` from the course page and read into `R` by

```
>> load('rating.txt')
```

  - Rating is represented by an integer in the range of [1,5]
  - R(2,4)=3 means person2 gave rating=3 for song4
- Suppose a new (i.e., 16th) person gives ratings for three songs
  - song1=4, song3=2, song7=3, i.e., $R_{16,1} = 4,\ R_{16,3} = 2,\ R_{16,7} = 3$
- Estimate ratings by this person for other songs
  - First, find a rank-3 approximation of R, i.e., obtain 15x3 P and 3x20 S
  - Second, find p_{16} that satisfies the following equations using S:

$$R_{16,1} = \mathbf{p}_{16}^\top \mathbf{s}_1$$
$$R_{16,3} = \mathbf{p}_{16}^\top \mathbf{s}_3$$
$$R_{16,7} = \mathbf{p}_{16}^\top \mathbf{s}_7$$

  - Finally, calculate prediction of ratings by $R_{16,j} = \mathbf{p}_{16}^\top \mathbf{s}_j$
  - True ratings are:

4 3 2 2 3 3 3 2 3 1 2 3 2 2 3 4 3 3 3 3