인공지능수학

(선형대수학)

개 요

벡터, 행렬

특징	연산	특수형태
노름, 거리, 차원, 내적, 행렬식	벡터합, 스칼라곱, 행렬합, 행렬곱	영행렬, 음행렬, 단위행렬, 역행렬, 대각행렬, 전치행렬 등

선형문제
- 선형결합
- 기본행변환
- 가우스-조르단 소거법

알고리즘
- LU분해
- QR분해
- 최소제곱법
- 그람-슈미트 방법

행렬의 특성값
- 고윳값&고유벡터
- 특잇값
  - 특잇값분해 (SVD)

고유치와 고유벡터

➲ 고유치가 단근인 경우

$$(\lambda_1 \neq \lambda_2 \neq \dots \neq \lambda_n)$$

행렬 $\mathbf{A}$의 i번째 고유치 λ_i에 대해, 다음을 만족하는 영이 아닌 벡터 $n \times 1$차원의 $\mathbf{X}_\mathrm{i}$를 행렬 $\mathbf{A}$의 **고유벡터**라고 한다.

$$\mathbf{A}\mathbf{X}_\mathrm{i} = \lambda_\mathrm{i}\mathbf{X}_\mathrm{i} \tag{2.79}$$

그런데 이 식이 임의의 스칼라 상수 k에서도 다음 방정식을 만족한다.

$$\mathbf{A}(k\mathbf{X}_\mathrm{i}) = \lambda_\mathrm{i}(\mathrm{k}\mathbf{X}_\mathrm{i}) \tag{2.80}$$

이 경우 해 벡터는 $k\mathbf{X}_\mathrm{i}$가 되므로, 고유벡터는 일의적(unique)이 아닌 임의의 길이가 설정된다.

고유치와 고유벡터

➲ 고유치가 / 중근인 경우 $(\lambda_1 = \lambda_2 = \ldots = \lambda_l \neq \ldots \neq \lambda_n)$

행렬 $\mathbf{A}$의 고유치가 중근이면, 앞의 방법으로는 n개의 선형 독립인 고유벡터를 구할 수 없으며 대각화도 할 수 없다. 그러나 고유치가 l 중근인 경우, 즉 $\lambda_1 = \lambda_2 = \cdots = \lambda_l$이면, 대각행렬과 유사한 다음과 같은 방법으로 표현할 수 있다.

$$\Lambda = \begin{bmatrix} \lambda_1 & 1 & 0 & 0 & \cdot & \cdot & \cdot & 0 \\ 0 & \lambda_2 & 1 & 0 & 0 & \cdot & \cdot & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & \cdot & \cdot & \lambda_{l-1} & 1 & \cdot & 0 \\ 0 & 0 & 0 & \cdot & 0 & \lambda_l & \cdot & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \lambda_n \end{bmatrix} \tag{2.81}$$

한편, 고유치가 l 중근인 경우, $\lambda_1 = \lambda_2 = \cdots = \lambda_l$에 대한 고유벡터는 단근인 경우의 식 (2.79)와 같이

$$\mathbf{A}\mathbf{X}_1 = \lambda_1 \mathbf{X}_1 \tag{2.82}$$

로부터 $\mathbf{X}_1$을 먼저 구한 다음,

고유치와 고유벡터

➲ 고유치가 / 중근인 경우

$$(\lambda_1 = \lambda_2 = \dots = \lambda_l \neq \dots \neq \lambda_n)$$

$$\begin{aligned} \mathbf{AX}_2 &= \mathbf{X}_1 + \lambda_1 \mathbf{X}_2 \\ \mathbf{AX}_3 &= \mathbf{X}_2 + \lambda_1 \mathbf{X}_3 \\ &\vdots \\ &\vdots \\ \mathbf{AX}_l &= \mathbf{X}_{l-1} + \lambda_1 \mathbf{X}_l \end{aligned} \tag{2.83}$$

또는

$$\begin{aligned} (\mathbf{A} - \lambda_1 \mathbf{I})\mathbf{X}_2 &= \mathbf{X}_1 \\ (\mathbf{A} - \lambda_1 \mathbf{I})\mathbf{X}_3 &= \mathbf{X}_2 \\ &\vdots \\ &\vdots \\ (\mathbf{A} - \lambda_1 \mathbf{I})\mathbf{X}_l &= \mathbf{X}_{l-1} \end{aligned} \tag{2.84}$$

에서 차례대로 구한다. 나머지 $(n-l)$개의 고유치는 단일 고유치를 구하는 방법과 같다.

고윳값 분해

고윳값 분해의 문제

- 고윳값 분해는 오로지 정방행렬에 대해서만 가능하다.
- 하지만 정방행렬이 아닌 일반적인 행렬에 대해서도 고윳값을 이용한 분석을 하고싶다면, 어떻게 해야할까?

$$A \qquad\qquad\qquad\qquad \mathrm{P}\,\Sigma\,\mathrm{P}^{-1}$$

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

$$B \qquad\qquad C$$

$$\begin{pmatrix} 3 & 1 \\ 2 & -1 \\ 1 & 2 \end{pmatrix}, \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix} \Rightarrow \ ?$$

비정방행렬의 분해는 어떻게 해야할까?

SVD (Singular Value Decomposition)

특잇값&특이벡터

■ 특잇값 & 특이벡터

행렬 A(m×n)에 대해 A^TA 를 하면 정방행렬 n×n 이 된다.
따라서 벡터 x, 스칼라 λ 가 다음 식 $(A^TA)x = \lambda x$ 을 만족한다면,

x : A의 특이벡터 ------ 고유벡터
$+\sqrt{\lambda} = \sigma$: A의 x에 대한 특잇값 -- 고유값

■ 특징

$A = [x_1\ x_2 \cdots x_n]$가 $m \times n$행렬이고 $x_1, x_2, \cdots, x_n$이 일차독립이면,

A^TA 와 AA^T의 0을 제외한 고윳값(특이값)은 동일하다.

>> 이에 따라 이후 A를 $m \geq n$인 $m \times n$행렬로 가정한다.

특이치 분해

특이치 행렬 S는 AA' 또는 $A'A$의 고유치의 제곱근이 대각에 있는 대칭행렬이다. 여기서 U의 열벡터는 AA' 행렬의 고유벡터이며, 각 열벡터는 행렬 A의 **좌측 특이벡터**다. 이와 유사하게 V의 열벡터는 $A'A$ 행렬의 고유벡터이며, 각 열벡터는 행렬 A의 **우측 특이벡터**다.

- A가 다음처럼 세 행렬의 곱으로 표현됩니다.

$$\boxed{A = USV^T}$$

$$A'A = VS'U'USV' = V(S'S)V'$$
$$AA' = USV'VS'U' = U(SS')U'$$

- A가 $m \times n$ 행렬일때

→ U는 $m \times m$ 행렬 : **U** 는 AA'의 고유벡터행렬

→ V는 $n \times n$ 행렬 : **V** 는 $A'A$의 고유벡터행렬

→ S는 $m \times n$ 행렬 : 특이치 대각 행렬

S의 주대각 성분을 행렬 A의 특잇값(singular value)이라고 부르며,

AA′ 과 A′A의 고유값의 제곱근이다.

특이치 분해 – 경우의 수에 따른 방법

◆ (1) 행렬 A 의 퇴화로 랭크가 부족한 경우

경우 1 방정식의 수와 미지수의 수가 같은 경우

다음 행렬 $\boldsymbol{A}$는 정방행렬이지만, 대칭행렬은 아니다.

$$\boldsymbol{A} = \begin{bmatrix} 1 & 1 \\ 2 & 2 \end{bmatrix}$$

이 행렬에서 제1행과 제2행은 서로 종속이므로, $|\boldsymbol{A}| = 0$인 특이치가 있다. 이 경우, 원래의 행렬은 한 행이 퇴화했으므로 랭크는 1이 된다. 따라서 $\boldsymbol{A}^{-1}$은 존재하지 않는다.
그러나 특이치 분해를 이용하면, 고유치와 고유벡터는 내적과 외적 행렬에서 다음을 얻게 된다.

$$A^T A = \begin{bmatrix} 5 & 5 \\ 5 & 5 \end{bmatrix} \quad \text{및} \quad AA^T = \begin{bmatrix} 2 & 2 \\ 4 & 8 \end{bmatrix}$$

이것은 모두 대칭행렬이며, 고유치는 $\lambda_1 = 0$ 및 $\lambda_2 = 10$이다. 결국 행렬 $\boldsymbol{A}$의 특이치는 $\sigma_1 = 0$ 및 $\sigma_2 = \sqrt{10}$이 된다. 따라서 행렬 $\boldsymbol{A}$의 랭크는 1이며, 다음과 같이 특이치 분해로 풀 수 있다.

$$\boldsymbol{A} = \boldsymbol{USV}^T = \begin{bmatrix} 2/\sqrt{5} & 1/\sqrt{5} \\ -1/\sqrt{5} & 2/\sqrt{5} \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & \sqrt{10} \end{bmatrix} \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ -1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix}^T = \begin{bmatrix} 1 & 1 \\ 2 & 2 \end{bmatrix}$$

특이치 분해 – 경우의 수에 따른 방법

경우 2 방정식의 수가 미지수의 수보다 적은 경우

다음 행렬 B는 비정방행렬이며, 행렬의 랭크는 2이다.

$$B = \begin{bmatrix} 3 & 1 & 1 \\ -1 & 3 & 1 \end{bmatrix}$$

이 행렬에서 제1행과 제2행은 서로 종속이므로, $|B| = 0$인 특이치가 있다. 이 경우, 원래의 행렬은 한 행이 퇴화했으므로 랭크는 1이 된다. 따라서 B^{-1}은 존재하지 않는다.
그러나 특이치 분해를 이용하면, 고유치와 고유벡터는 내적과 외적 행렬에서 다음을 얻게 된다.

$$B^T B = \begin{bmatrix} 10 & 0 & 2 \\ 0 & 10 & 4 \\ 2 & 4 & 2 \end{bmatrix} \text{ 및 } BB^T = \begin{bmatrix} 11 & 1 \\ 1 & 11 \end{bmatrix}$$

이것은 모두 대칭행렬이며, 전자의 고유치는 $\lambda_1 = 12$ 및 $\lambda_2 = 10$, $\lambda_3 = 0$이고, 후자의 고유치는 $\lambda_1 = 12$ 및 $\lambda_2 = 10$이다. 결국 행렬 B의 특이치는 $\sigma_1 = \sqrt{12}$ 및 $\sigma_2 = \sqrt{10}$이 된다. 따라서 행렬 B의 랭크는 2이며, 다음과 같이 특이치 분해로 풀 수 있다.

$$\begin{aligned} B &= USV^T \\ &= \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ 1/\sqrt{2} & -1/\sqrt{2} \end{bmatrix} \begin{bmatrix} \sqrt{12} & 0 & 0 \\ 0 & \sqrt{10} & 0 \end{bmatrix} \begin{bmatrix} 1/\sqrt{6} & 2/\sqrt{5} & 1/\sqrt{30} \\ 2/\sqrt{6} & -1/\sqrt{5} & 2/\sqrt{30} \\ 1/\sqrt{6} & 0 & -5/\sqrt{30} \end{bmatrix} = \begin{bmatrix} 3 & 1 & 1 \\ -1 & 3 & 1 \end{bmatrix} \end{aligned}$$

특이치 분해 – 경우의 수에 따른 방법

경우 3 방정식의 수가 미지수의 수보다 많은 경우(랭크가 결여된 경우)

다음 행렬 C는 비정방행렬이다.

$$C = \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 0 & 0 \end{bmatrix}$$

이 행렬에서 제1행과 제2행은 서로 종속이며, 고유치와 고유벡터는 내적과 외적 행렬에서 다음을 얻게 된다.

$$C^T C = \begin{bmatrix} 2 & 2 \\ 2 & 2 \end{bmatrix} \quad \text{및} \quad C C^T = \begin{bmatrix} 2 & 2 & 0 \\ 2 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

이것은 모두 대칭행렬이며, 전자의 고유치는 $\lambda_1 = 4$ 및 $\lambda_2 = 0$이고, 후자의 고유치는 $\lambda_1 = 4$ 및 $\lambda_2 = 0$, $\lambda_3 = 0$이다. 결국 행렬 C의 특이치는 $\sigma_1 = \sqrt{4}$ 및 $\sigma_2 = 0$이 된다. 따라서 행렬 C의 랭크는 1이며, 다음과 같이 특이치 분해로 풀 수 있다.

$$C = USV^T = \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} & 0 \\ 1/\sqrt{2} & -1/\sqrt{2} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \sqrt{4} & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ 1/\sqrt{2} & -1/\sqrt{2} \end{bmatrix}^T = \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 0 & 0 \end{bmatrix}$$

특이치 분해 – 경우의 수에 따른 방법

◆ (2) 랭크가 행렬 A 의 열 또는 행과 같아지는 경우

경우 4 랭크가 행렬 A의 열 또는 행과 같아지는 경우

랭크가 행렬 A의 열 또는 행과 같아지면, 의사 역행렬을 이용한다. 식 (2.54)의 미지수 벡터는 $m > n$일 때, 다음과 같이 좌 의사 역행렬 A_{LM}^{-1}으로 구할 수 있으며,

$$A^T A x = A^T b$$

$$x = (A^T A)^{-1} A^T b = A_{LM}^{-1} b \qquad (2.55)$$

$n > m$일 때, 다음과 같이 우 의사 역행렬 A_{RM}^{-1}으로 구할 수 있다.

$$A A^T x = A^T b$$

$$x = A^T (A A^T)^{-1} b = A_{RM}^{-1} b \qquad (2.56)$$

일반적으로 방정식의 수가 미지수의 수보다 많은 경우, 일종의 최소제곱법과 유사하게 좌 의사 역행렬로 다음을 구하고,

$$x = (A^T A)^{-1} A^T b = U \Lambda^{-1} V^T b \qquad (2.57)$$

방정식의 수가 미지수의 수보다 적은 경우, 우 의사 역행렬로 다음을 구한다.

$$x = A^T (A A^T)^{-1} b = A_{RM}^{-1} b \qquad (2.58)$$

특잇값 분해(SVD)

Singular Value Decomposition(SVD)

$$\underset{A}{\begin{bmatrix} \star & \star & \star \\ \star & \star & \star \\ \star & \star & \star \\ \star & \star & \star \end{bmatrix}} \xrightarrow{\text{SVD}} \underset{U}{\begin{bmatrix} \blacksquare & \blacksquare & \blacksquare & \blacksquare \\ \blacksquare & \blacksquare & \blacksquare & \blacksquare \\ \blacksquare & \blacksquare & \blacksquare & \blacksquare \\ \blacksquare & \blacksquare & \blacksquare & \blacksquare \end{bmatrix}} \underset{\Sigma}{\begin{bmatrix} \sigma_1 & 0 & 0 \\ 0 & \sigma_2 & 0 \\ 0 & 0 & \sigma_3 \\ 0 & 0 & 0 \end{bmatrix}} \underset{V^T}{\begin{bmatrix} \blacklozenge & \blacklozenge & \blacklozenge \\ \blacklozenge & \blacklozenge & \blacklozenge \\ \blacklozenge & \blacklozenge & \blacklozenge \end{bmatrix}}$$

A : $m \times n$ 행렬

σ_i : A의 특잇값 $(\sigma_1 \geq \cdots \geq \sigma_n)$

U : $AA^T = U(\Sigma\Sigma^T)U^T$를 만족하는 $m \times m$ 정규직교행렬

V : $A^TA = V(\Sigma^T\Sigma)V^T$를 만족하는 $n \times n$ 정규직교행렬

특잇값 분해(SVD)

SVD 계산과정

Input

$m \times n$ **matrix**
A

Output

$A = \mathrm{U}\,\Sigma\,V^T$

Find $\sigma = \sqrt{\lambda}$
s.t.
$A^T A x = \lambda x$

$\text{Make } \Sigma$

특잇값 계산

$\text{Get } x \text{ s.t.}$
$(AA^T - \lambda I)x = 0$

$\text{Get } y \text{ s.t.}$
$(A^T A - \lambda I)y = 0$

$A^T A$와 AA^T의
고유벡터 계산

$\text{Make } \dot{U} \text{ s.t.}$
$\dot{U} = [x_1\ x_2 \cdots x_m]$

$\text{Make } \dot{V} \text{ s.t.}$
$\dot{V} = [y_1\ y_2 \cdots y_n]$

$\dot{\mathrm{U}}, \dot{\mathrm{V}}$를
정규직교화하여
U, V 생성

QR법

LU분해

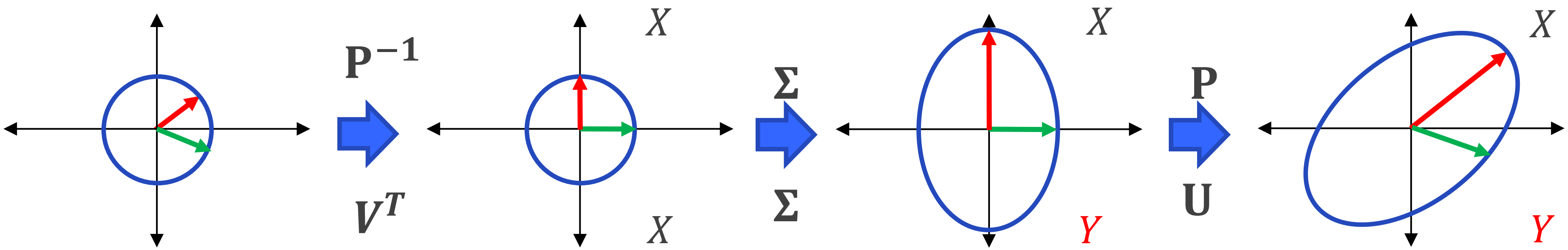
정규직교화과정

특잇값 분해(SVD)

고윳값 분해와 SVD 비교

고윳값 분해 $\boxed{\mathbf{A} = \mathbf{P}\,\boldsymbol{\Sigma}\,\mathbf{P}^{-1}}$ 정방행렬 $A: X \to X$

벡터공간 X



벡터공간 X

SVD $\boxed{\mathbf{A} = \mathbf{U}\,\boldsymbol{\Sigma}\,V^T}$ 행렬 $A: X \to Y$

SVD는 고윳값 분해를 일반화한 개념이라고 볼 수 있다.

특잇값 분해(SVD)

SVD의 기하학적 의미

만약 고윳값들이 $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n$ 를 만족한다면,

$$A = \sigma_1 x_1 y_1 + \sigma_2 x_2 y_2 + \cdots + \sigma_{n-1} x_{n-1} y_{n-1} + \sigma_n x_n y_n$$

중요함 ⟷ 무시해도 됨

특징의 중요도

위의 특징을 이용한 응용

- A의 중요한 특징만 조사하고자 한다. → 특징 추출 (Feature extraction)
- 필요없는 특징들은 제거하고자 한다. → 디노이징 (Denoising)

고윳값 분해와 동일하다!

특잇값 분해(SVD)

Python을 이용한 SVD 계산

Simulation results

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

Input

```
>>> A
array([[1, 2],
       [2, 3],
       [3, 4],
       [4, 5]])
```

The results of SVD

U, 특잇값, V^T

```
>>> u, s, vt = np.linalg.svd(A)
>>> u
array([[-0.24054726, -0.80133452, -0.40008743, -0.37407225],
       [-0.3934325 , -0.38106544,  0.25463292,  0.79697056],
       [-0.54631774,  0.03920365,  0.69099646, -0.47172438],
       [-0.69920298,  0.45947273, -0.54554195,  0.04882607]])
>>> s
array([9.15211593, 0.48864503])
>>> vt
array([[-0.59693053, -0.80229293],
       [ 0.80229293, -0.59693053]])
```

특잇값 분해(SVD)

Python을 이용한 SVD 계산

Simulation results

```
>>> A
array([[1, 2],
       [2, 3],
       [3, 4],
       [4, 5]])
```

Σ

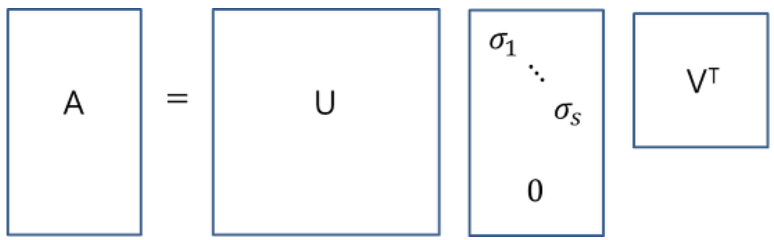
검증

$U \Sigma V^T$

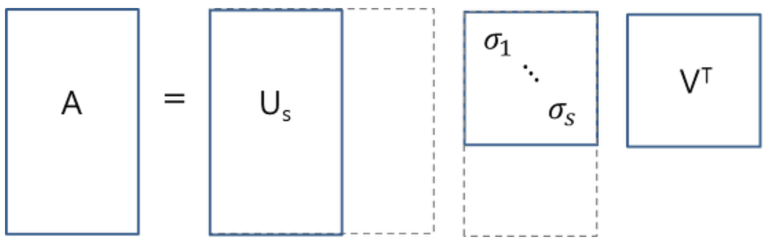
```
>>> S
array([[9.15211593, 0.        ],
       [0.        , 0.48864503],
       [0.        , 0.        ],
       [0.        , 0.        ]])
>>> B = np.matmul(np.matmul(u,S),vt)
>>> B
array([[1., 2.],
       [2., 3.],
       [3., 4.],
       [4., 5.]])
```

절단된 SVD

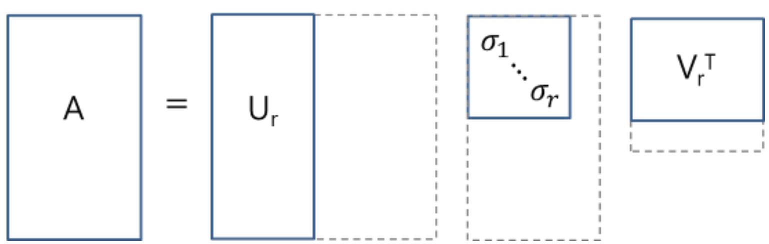
절단된 SVD



Full SVD

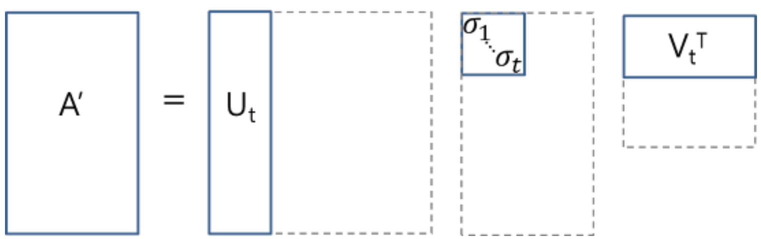


Thin SVD
(square matrix)



Compact SVD
($\sigma_i = 0$ 인것 제거)

*Thin SVD와 다르게
$\sigma_i = 0$인 특잇값도 삭제한다.



Truncated SVD

$$A_k = \sigma_1 u_1 v^T + \cdots + \sigma_k u_k v^T$$

♣ 유의미한 고윳값 σ_i들을 제외하고 전부 삭제한다.

절단된 SVD

■ Eckart-Young theorem

$A_k = \sigma_1 u_1 v^T + \cdots + \sigma_k u_k v^T$를 A 의 truncated matrix라 하자.

만약 B가 rank k인 행렬이라면, $|A - B| \geq |A - A_k|$가 항상 성립한다.

Truncated matrix A_k는 A와 가장 비슷한 rank k 행렬이다!!

이에 따라 Truncated SVD의 결과를 A의 근사값으로 사용할 수 있다.

특잇값 분해(SVD)

SVD의 응용

무어-펜로즈 의사역행렬

$$A = U\,\Sigma\,V^T$$
$$\Rightarrow \underline{A^+} = V\,\Sigma^+\,U^T$$

$$\Sigma^+ = \begin{bmatrix} \frac{1}{\sigma_1} & 0 & 0 \\ 0 & \frac{1}{\sigma_2} & 0 \\ 0 & 0 & \frac{1}{\sigma_3} \\ 0 & 0 & 0 \end{bmatrix}^T$$

만약 A가 $m \times n$행렬이면,
$AA^+ = I_m$.

데이터 압축
(절단 SVD의 r의 값)