3.0 고급 선형대수

3.1 선형대수와 해석기하의 기초

1) 벡터의 길이

- ||a|| = $\sqrt{\vec{a}^T\vec{a}} = \sqrt{a_1^2+\cdots+a_n^2}$
- 단위벡터

2) 유클리드 거리

- 정의

3) 벡터의 내적과 삼각함수

- 내적 = $\vec{a}\cdot\vec{b} = \vec{a}^T\vec{b} = \|\vec{a}\|\,\|\vec{b}\|\cos\theta$
- sin, cos = $\frac{\text{opposite}}{\text{빗변}}$, $\frac{\text{adj}}{\text{빗변}}$

4) 직교

- 정의

5) 정규직교

- 정의 $\|v_i\|=1,\quad v_i^T v_i=1,\quad v_i^T v_j=0\ (i\neq j)$
- 정규직교 벡터의 역행렬 = 전치행렬

6) 코사인 유사도

- 정의
- 벡터 간 거리 : 코사인 거리 / 유클리드 거리

$=$ $\|\vec{a}-\vec{b}\|=$

03.01 선형대수와 해석기하의 기초

3.1 선형대수와 해석기하의 기초

7) 벡터의 분해와 성분

- a + b = c, c = a와 b로 분해될 수 있음. c라는 벡터의 분해는 (a,b)외에도 수 많은 조합으로 가능

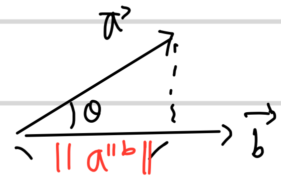
8) 정사영

- 정사영은 벡터와 직선(or벡터) 간 수직거리 이해에 필요

- 벡터 a를 정사영과 직교성분으로 분해
- 정사영 길이 = ||a|| cos
- 정사영 길이 = 내적값 / ||b||
- 정사영 길이 = 벡터 a와 단위벡터(b의) 내적
- 정사영 벡터 = 정사영 길이 * 단위벡터

- 벡터 a = a의 정사영 벡터 + a의 직교벡터

[정사영의 길이]



$= proj_b \vec{a}$ * $\frac{\|a^{\|b}\|}{\|a\|} = \cos\theta$

∴ $\|a^{\|b}\| = \|a\|\cos\theta = \frac{a^T b}{\|b\|}$

[정사영]

$a^{\|b} = proj\ \vec{a} = \underbrace{(\vec{a}\cdot\hat{u})}_{\text{스칼라}}\ \underbrace{\hat{u}}_{\text{단위벡터}} = \underbrace{\left(a^T \frac{b}{\|b\|}\right)}_{\text{스칼라,길이}} \underbrace{\frac{b}{\|b\|}}_{\text{벡터,단위}}$

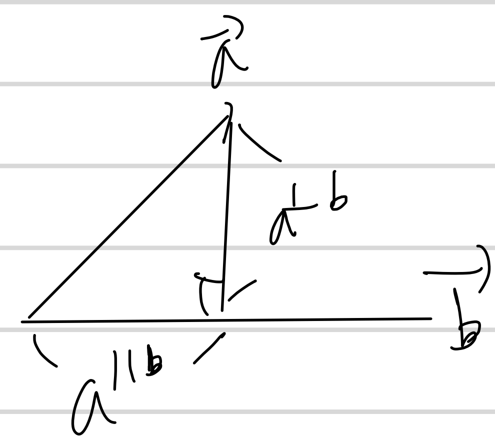
* "$\|a\|\cos\theta = \frac{a^T b}{\|b\|}$"

$a^T b = \vec{a}\cdot\vec{b} = \|a\|\|b\|\cos\theta$

$= \|b\|\ proj_b(a)$

$= \|b\|\ a^{\|b}$

$\vec{a} = a^{\|b} + a^{\perp b}$



3.0 고급 선형대수

3.1 선형대수와 해석기하의 기초

9) 직선의 방정식

벡터 W에 수직인 "임의의" 직선의방정식 (W_0 활용!)

- 벡터 w에 수직인 직선의 방정식을 구하라
- 그 직선과 원점사이의 거리를 구하라

1) $\vec{W}$ 와 $\vec{x}$ 을 잡고
2) 직교성질 이용, 방정식생성
3) $W \rightarrow W'$ 으로 확장!

$\|W\|$와 $\|W'\|$의 차이 확인

연습문제 3.1.10 풀기

10) 직선과 점사이 거리

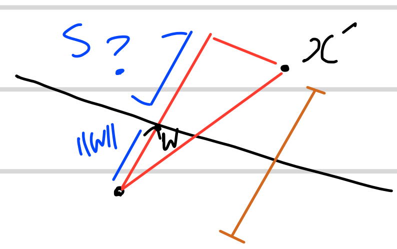
- 벡터 w에 수직인 직선과 x'의 거리를 구하라

Q. 직선과 x'의 거리 스칼라 (길이)

① $W \perp x'$의 그림그리기 (큰그림, 이해목적)
② W에 수직인 직선의 방정식 달기
③ $x'^{\|W}$
④ $\left| x'^{\|W'} - \|W'\| \right|$

연습문제 3.1.11 풀기

직선과 점(x')의 거리



$L : W^T x - \|W\|^2 = 0$

① W와 x' 사이 거리 (정사영)

$* \|a^{\|b}\| = a^T \frac{b}{\|b\|}$

$\|x'^{\|W}\| = x'^T \frac{W}{\|W\|} = \frac{W^T}{\|W\|} x' \qquad (x'^T W = W^T x')$

② 직선과 x' 사이 거리 S

$S = \left| \|x'^{\|W}\| - \|W\| \right| = \left| \frac{W^T}{\|W\|} x' - \|W\| \right|$

$= \left| \frac{W^T x' - \|W\|^2}{\|W\|} \right| = \left| \frac{W^T x' - W_0}{\|W\|} \right|$

03. 02 좌표와 변환

1) 선형독립과 선형종속

- 선형독립 : N(A)의 only solution이 영벡터

- 선형독립의 벡터들은 서로 직교한다.

(데카르트 좌표계의 x축과 y축이 수직인 이유. 서로 직교여야 독립)

Q1.

선형독립 벡터들은 왜 직교?

- 연습문제 3.2.3

선형독립 ⟷ 연립방정식 풀 ⟷ 회귀분석 O

종속 안풀 X

2) 선형독립과 연립방정식

Q2.

선형종속이면 왜 연립방정식 안풀리나?

- 선형독립의 논리적 식 (3.2.13식)

벡터 $x_1, x_2, \ldots, x_N$이 선형독립이라는 것을 논리기호로 나타내면 다음과 같다.

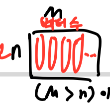
$$Xc = 0 \quad \rightarrow \quad c = 0$$

- 선형종속의 대표적인 경우 (03.02 3page)

1) 벡터의 갯수가 벡터의 차원보다 크면 선형종속

벡터수 m

벡터차원 n

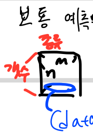


(m > n) 이 경우에 해당

보통 예측문제 입력데이터 = 특징행렬

종류

개수



$(data)^T$

ex) 면적, 층수, 뷰 ⇐ 종류

샘플데이터수 ⇐ 개수

최소 3집이상!

2) 값이 같은 벡터가 있으면 선형종속

3) 어떤 벡터가 다른 벡터의 조합이면 선형종속

일부가 종속이면,

전체가 종속이다.

$$x_1 \quad x_2 \quad \underbrace{x_3 \quad x_4 \quad x_5}_{\text{종속}}$$

종속

3) 랭크

≒ C(A)의 기저벡터 갯수

- 랭크의 정의 행렬 내 선형독립 벡터의 최대개수

ex) x_1 x_2 x_3 x_4 합계 x_5 평균

데이터에 x_4, x_5 같은 열 추가하면 선형종속이 됨

- 열랭크와 행랭크는 항상 같다

- rank A <= min(M,N) *A가 M,N행렬일 때 (03.02 4page)

- code) rank구하는 코드

3

5

행 랭크도

열랭크 = 최대 3개 (∴ 최대 3개)

4) 풀랭크

- 정의 $\text{rank } A_{(m \times n)} = \min(m, n)$ 이면, full rank

- Low-랭크 행렬

- 1랭크 행렬 : N차원 벡터x 1개를 이용해 만든 행렬

(nxn행렬은 사실 벡터1개의 복제품)

- 2랭크 행렬 : N차원 벡터x 2개를 이용해 만든 행렬

- 3랭크 행렬 : N차원 벡터x 3개를 이용해 만든 행렬

*왜 선형종속이면 역행렬 X?

(왜 선형독립으로만 채워져야 하는가)

- 랭크와 역행렬

- 정방행렬이 풀랭크면, 역행렬이 존재

(전단사함수 <-> 역행렬 존재)

(rankA=N, rankA=M, M=N => 정방행렬, 풀랭크 조건 = 역행렬 조건)

(즉, 모두 선형독립인 열벡터만으로 이뤄진 정방행렬 <-> 역행렬 존재)

(즉, 풀랭크 정방행렬 <-> 역행렬 존재)

5) 벡터공간과 기저벡터

- 벡터공간 : 선형독립인 벡터 N개의 선형결합으로 만들어지는 집합

- 기저벡터 : 벡터공간을 이루는 선형독립인 벡터들
    - 벡터공간의 차원 : 기저벡터의 갯수

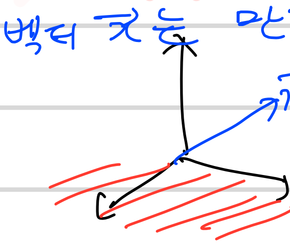
(기저벡터 2개로는 2차원 column space 만들 수 있음)

ex) 만약, 선형독립인 3차원 벡터 2개가 주어졌다면,

2차원 column space는 만들 수 있지만,

3차원 column space는 만들 수 없다.

2차원 열공간을 떠난 벡터 것는 만들 수 없다.



Q. C(A)는 pivot variable 수만큼 기저벡터 갯수?

- 정리 : M차원 벡터 N개가 *선형독립* 일 때 (M x N 행렬) full rank
    - N = M => 모든 N차원 C(A) 생성 가능 (span of 벡터N개)
    - N > M => 갯수가 더 많음. N개 중 일부는 선형종속. 따라서, 선형종속이다.

        기저벡터 갯수만큼 차원의 열공간 생성
    - N < M => 갯수가 더 적음. N차원 C(A)가능. (but, M차원 C(A)는 불가)

        ex) 3차원 벡터 2개가 선형독립

    => 2차원 C(A) 가능. (but, 3차원 C(A)는 불가.)

6) 정사영 (shadow to 벡터공간)

- 정규직교인 기저벡터의 M차원 공간 정사영

(03.02 10page그림, 03.02 11page)

- 정사영 =
- 정사영 길이 =

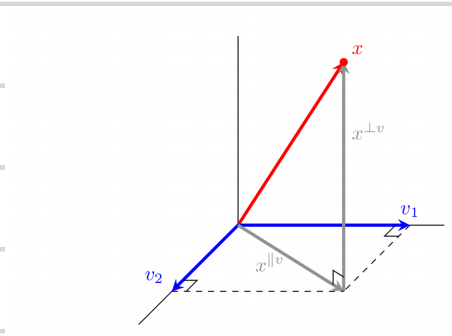


그림 3.2.1 : 3차원 벡터를 2차원 벡터공간에 투영하는 예

만약 기저벡터 $v_1, v_2, \cdots, v_M$가 정규직교(orthonormal)이면 투영벡터 $x^{\|v}$는 각 기저벡터에 대한 내적값으로 표현된다.

$x^{\|\hat{u}} = (x \cdot \hat{u})\hat{u}$

$$x^{\|V} = (x^T v_1)v_1 + (x^T v_2)v_2 + \cdots + (x^T v_M)v_M \tag{3.2.47}$$

그리고 투영벡터의 길이의 제곱은 각 기저벡터와의 내적의 제곱합이다.

$\|x^{\|\hat{u}}\| = x \cdot \hat{u}$

$$\|x^{\|V}\|^2 = \sum_{i=1}^{M} (x^T v_i)^2 \tag{3.2.48}$$

03. 02 좌표와 변환

7) 표준기저벡터

- e1, e2, ...
- 좌표 : 기저벡터를 선형조합하여 그 벡터를 나타내기 위한 계수벡터를 의미

(03.02 12page)

* $e_1, e_2 \rightarrow$ 보통 표준기저벡터

$$\begin{bmatrix} x \end{bmatrix} = x_{e_1} \begin{bmatrix} e_1 \end{bmatrix} + x_{e_2} \begin{bmatrix} e_2 \end{bmatrix}$$

좌표

8) 변환행렬

- 역할 : 기저벡터의 변환, 회전 및 스케일 (03.02 17page, 03.02 19page)

원래의 기저벡터가 아닌 새로운 기저벡터가 있다고 하자. 이 새로운 기저벡터들의 기존 기저벡터에 대한 좌표를 열벡터로 보고 이를 행렬로 묶은 행렬 A를 생각하자.

예를 들어, 기존의 기저벡터가 $\{e_1, e_2\}$이고 새로운 기저벡터 $\{g_1, g_2\}$간에 다음과 같은 관계가 성립한다면,

$$
\begin{aligned}
g_1 &= \frac{1}{\sqrt{2}}e_1 + \frac{1}{\sqrt{2}}e_2 \\
g_2 &= -\frac{1}{\sqrt{2}}e_1 + \frac{1}{\sqrt{2}}e_2
\end{aligned}
\tag{3.2.59}
$$

e_1, e_2 에 대한 g_1, g_2 의 좌표벡터는 다음처럼 열벡터로 나타낼 수 있다.

$$g_{1e} = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix}, \quad g_{2e} = \begin{bmatrix} -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix} \tag{3.2.60}$$

두 좌표벡터들을 합쳐서 행렬로 표시하면 다음과 같다.

$$\begin{bmatrix} g_1 & g_2 \end{bmatrix} = \begin{bmatrix} e_1 & e_2 \end{bmatrix} \begin{bmatrix} g_{1e} & g_{2e} \end{bmatrix} = \begin{bmatrix} e_1 & e_2 \end{bmatrix} A \tag{3.2.61}$$

$$A = \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \tag{3.2.62}$$

$$\underset{\{g_1, g_2\}\text{ 기저}}{\text{좌표}} = \underset{= A^{-1}}{\boxed{T}} \; \underset{\{e_1, e_2\}\text{ 기저}}{\text{좌표}}$$

$$\vec{x}_g = T(A^{-1})\, \vec{x}_e$$

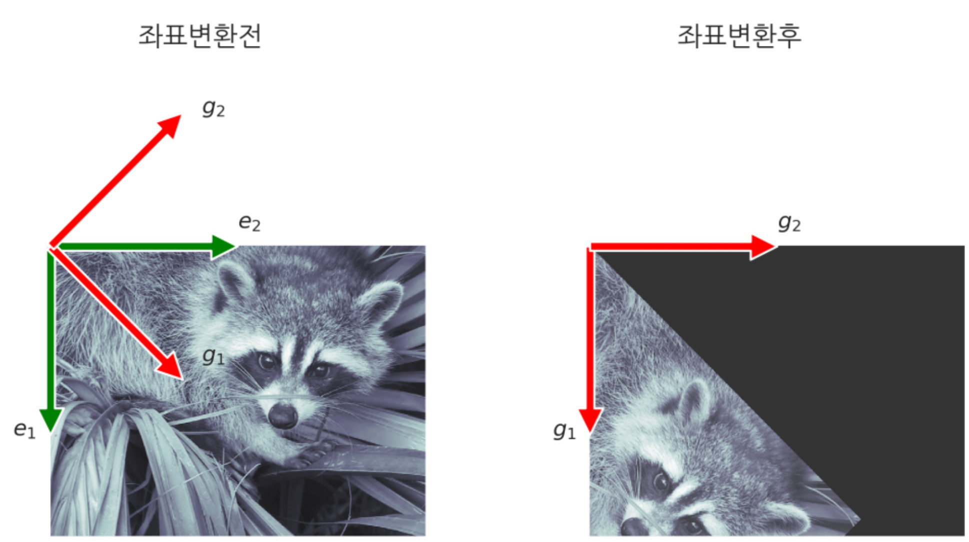
$\Leftrightarrow$

$$\vec{X}_e = A \vec{X}_g$$
$$A^{-1}\vec{X}_e = \vec{X}_g$$
$$\Leftrightarrow T\vec{X}_e = \vec{X}_g$$

- T (변환 행렬)의 역할 = 회전, 스케일

9) 이미지변환

- 03.02 23page
- 연습문제 3.2.9 (수기 및 코드 확인)



03.03 고유값 분해

1) 고유값과 고유벡터

- 정의

정방 행렬 A에 대해 다음 식을 만족하는 영벡터가 아닌 벡터 v, 실수 λ를 찾을 수 있다고 가정하자.

$$Av = \lambda v \tag{3.3.1}$$

위 식을 만족하는 실수 λ를 **고윳값**(eigenvalue), 벡터 v를 **고유벡터**(eigenvector)라고 한다. 고윳값과 고유벡터를 찾는 작업을 **고유분해(eigen-decomposition)** 또는 **고윳값 분해(eigenvalue decomposition)**라고 한다.

- 고유값 분해는 정방행렬에서만 가능
- 행렬을 곱해서 변환 시, 방향은 변하지 않고 스케일링 효과만 있을 때
  그때의 스케일러를 고유값, 방향을 고유벡터라고 한다.
- 대표적으로 고유벡터를 단위벡터로 상정해서 얘기한다.

2) 고유값 계산

- 특성방정식의 해 찾기 (03.03 2page)

행렬 A의 고유값은 $A - \lambda I$의 행렬식이 0이 되도록 하는 **특성방정식(characteristic equation)**의 해를 구하면 된다.

$$\det(A - \lambda I) = 0 \tag{3.3.16}$$

[고유값 갯수]

- N차 정방행렬의 고유값은 항상 N개 (중근 및 복소수 모두 별개라고 가정 시)

n차 → n차 변환. ∴ n차원 각각을 Scaling 해주기 위해선,
당연히 고윳값도 n개 있어야!

[고유값과 trace, determinent]

- tr(A) = 고유값 합
- det(A) = 고유값 곱 *고유값 중 0이 하나라도 존재하면, 역행렬이 존재하지 않는다.
  (det(A) = 0 이 될테니까)

3) 고유벡터 계산

- 특성방정식에 고유값 넣어서 계산

4) 대각화

- 고유값 행렬 = diag(람다i)
- 정방행렬 A 의 고유값분해 = 대각화 (03.03 8page)

고유벡터행렬 V은 고유벡터를 열벡터로 옆으로 쌓아서 만든 행렬이다.

$$V = [v_1 \cdots v_N] \tag{3.3.50}$$

$$V \in \mathbf{R}^{N \times N} \tag{3.3.51}$$

고윳값행렬 Λ은 고윳값을 대각성분으로 가지는 대각행렬이다.

$$\Lambda = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_N \end{bmatrix} \tag{3.3.52}$$

$$\Lambda \in \mathbf{R}^{N \times N} \tag{3.3.53}$$

위와 같이 고유벡터행렬과 고윳값행렬을 정의하면 **행렬과 고유벡터행렬의 곱은 고유벡터행렬과 고윳값행렬의 곱과 같다.**

$$\begin{aligned} AV &= A[v_1 \cdots v_N] \\ &= [Av_1 \cdots Av_N] \\ &= [\lambda_1 v_1 \cdots \lambda_N v_N] \\ &= [v_1 \cdots v_N] \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_N \end{bmatrix} \\ &= V\Lambda \end{aligned} \tag{3.3.54}$$

즉,

$$AV = V\Lambda \tag{3.3.55}$$

만약 **고유벡터행렬 V의 역행렬이 존재한다면** 행렬을 다음처럼 고유벡터행렬과 고윳값행렬의 곱으로 표현할 수 있다. 이를 행렬의 **대각화(diagonalization)**라고 한다.

$$A = V\Lambda V^{-1} \tag{3.3.56}$$

5) 대각화 가능 조건

- full rank는 역행렬. 역행렬은 full rank
- 고유벡터행렬이 full rank. 선형독립 set이어야 한다.

선형독립
|
기저벡터
|
직교?

직교하면
선형독립?

Ans)
직교하는 애들은
거의는
선형독립(?)

6) 대칭행렬의 고유분해

- 대칭행렬 => orthogonal 고유벡터행렬 + 실수 고유값 갖는다.

(고유값은 실수 + 고유벡터는 서로 직교)

- 대칭행렬은 항상 대각화가 가능함

(고유벡터 행렬이 full rank -> 고유벡터 행렬의 inverse 가능)

- 만약, 고유벡터들이 정규화된 상태라면
- 대칭행렬 => orthonormal 고유벡터행렬 + 실수 고유값 갖는다.

(고유값은 실수 + 고유벡터는 서로 정규직교)

- 고유벡터행렬 V는 정규직교행렬... 따라서, VV.T = I <==> V.inv = V.T

- 대칭행렬의 고유분해 (03.03 11page)

(고유벡터들이 정규화된 정규직교벡터set 행렬 V)

대칭행렬에 대해서는 다음 정리가 성립한다.

> **[정리] 행렬 A가 실수인 대칭행렬이면 고유값이 실수이고 고유벡터는 서로 직교(orthogonal)한다.**

만약 고유벡터들이 크기가 1이 되도록 정규화된 상태라면 고유벡터 행렬 V는 정규직교(orthonormal) 행렬이므로 **전치행렬이 역행렬**이다.

$$V^TV = VV^T = I \tag{3.3.68}$$

$$V^{-1} = V^T \tag{3.3.69}$$

따라서 대각화가 가능하고 다음처럼 쓸 수 있다.

$$A = V\Lambda V^T \tag{3.3.70}$$

이 사실로부터 다음 정리도 도출된다.

> **[정리] 실수인 대칭행렬은 항상 대각화가능**하다.

7) 대칭행렬을 랭크-1 행렬의 합으로 분해 (03.03 12page) (매우 중요)

- 부분행렬 3번 case로 해석 (수기로 해보기) + 예제 풀이

N차원 대칭행렬 A는 다음처럼 N개의 랭크-1 행렬 $A_i = v_iv_i^T$ 의 합으로 표시할 수 있다.

$$
\begin{aligned}
A &= V\Lambda V^T \\
&= \begin{bmatrix} v_1 & v_2 & \cdots & v_N \end{bmatrix}
\begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_N \end{bmatrix}
\begin{bmatrix} v_1^T \\ v_2^T \\ \vdots \\ v_N^T \end{bmatrix} \\
&= \begin{bmatrix} \lambda_1 v_1 & \lambda_2 v_2 & \cdots & \lambda_N v_N \end{bmatrix}
\begin{bmatrix} v_1^T \\ v_2^T \\ \vdots \\ v_N^T \end{bmatrix}
\end{aligned}
\tag{3.3.71}
$$

따라서 N차원 대칭행렬 A는

$$A = \sum_{i=1}^{N} \lambda_i v_i v_i^T = \sum_{i=1}^{N} \lambda_i A_i = \lambda_1 A_1 + \cdots + \lambda_N A_N \tag{3.3.72}$$

질문:
정방행렬이
양의정부호라면
고유값이
양수?
(대칭행렬만
부호 확인!)

8) 대칭행렬의 고유값 부호

- 대칭행렬이 양의정부호 <==> 고유값이 모두 양수 (매우 중요)
- 대칭행렬이 양의준정부호 <==> 고유값이 모두 양수 or 0
- 둘 다 증명 해보기(꼭 암기는 아니더라도, 해보기!)

* 부호 판별
- R^2 : $\begin{pmatrix} a & b \\ b & c \end{pmatrix}$에서 $a>0$, $ac>b^2$일 때 양의 정부호
- R^n $(n\geq 2)$: $f = x^TAx > 0$ 이면, 양의 정부호 / $\lambda_i > 0$ 이면 +

9) 분산행렬

- 분산행렬 : X.T X 인 정방행렬
- 분산행렬은 대칭행렬이다.
- 분산행렬은 양의정부호 + 고유값 >= 0

준

(분산행렬은 대칭행렬. 대칭행렬이 양의정부호이면, 고유값은 모두 양수)

- 분산행렬이 양의정부호 증명(꼭 해보기) 분산행렬은 항상 pos def 인 semi-posi def

준

10) 분산행렬의 역행렬

- (03.03 17page)

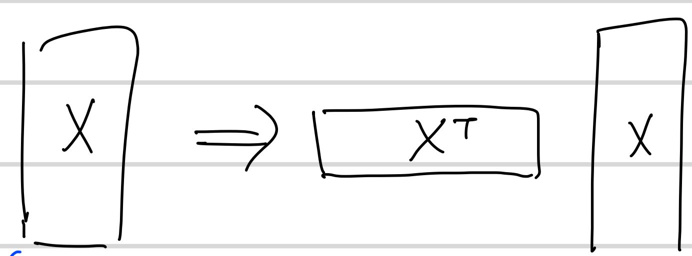
*X만 full rank라면, 역행렬 존재하지 않아도..

그것의 분산행렬인 X.TX의 역행렬이 존재한다는 것.

- X가 full rank가 아니라면, 분산행렬의 역행렬이 존재하지 않고

==> 최소제곱법(Ax = b의 근사해 찾기 위한 수도인버스) 불가

(03.03 17page 하단부)



full rank 여야

분산행렬 역행렬 존재해야

선형회귀 풀기위한 수도인버스(최소제곱법) 가능.

(연립방정식)

$X^T X$ 가 full rank 이고,

11) 고유분해의 성질 (03.03 18page)

요약: 고유분해의 성질

지금까지 나왔던 고유분해와 관련된 정리를 다시 한 번 요약하였다. 이 정리들은 데이터 분석에서 자주 사용되므로 잘 알아두자.

N 차원 정방행렬 A에 대해

1. 행렬 A는 N개의 고윳값-고유벡터를 가진다(복소수인 경우와 중복인 경우를 포함).
2. 행렬의 **대각합은 모든 고윳값의 합**과 같다.
3. 행렬의 **행렬식은 모든 고윳값의 곱**과 같다.
4. 행렬 A가 **대칭행렬**이면 N개의 **실수 고윳값**을 가지며 고유벡터들이 서로 **직교(orthogonal)**이다.
5. 행렬 A가 **대칭행렬**이고 고윳값이 모두 **양수**이면 **양의 정부호(positive-definite)이고 역행렬이 존재한다.** 역도 성립한다.
6. 행렬 A가 어떤 행렬 X의 **분산행렬 X^TX이면 0 또는 양의 고윳값을 가진다.**
7. 행렬 $X \in \mathbf{R}^{N \times M} (N \geq M)$가 **풀랭크이면 분산행렬 X^TX은 역행렬이 존재**한다.

↳ 양의정부호!

고윳값 분해 ⊃ 대칭행렬 ⊃ 분산행렬

$A = V \Lambda V^{-1}$
[A = square
V = 고유벡터 set (직교 아님)

$A_{sym} = V \Lambda V^T$
[A_{sym}
V = (정규)직교 고유벡터 set

$X^T X = V \Lambda V^T$
[$X^T X$ sym
V = (정규)직교 〃

· $A_{sym} = \sum_i \lambda_i A_i$
($* A_i = v_i v_i^T$, rank-1 행렬)

· $X^T X$는 posi or semi-posi

V^{-1} · V^{-1} 존재시 대각화 가능

V^{-1} · 항상 대각화 가능
(V = orthonormal, 선형독립 고유벡터 집합!
⇒ full rank, square ⇒ 역항상 존재)

· 항상 대각화가능

A^{-1} · A는 λ중 0 없으면
역행렬 존재

A^{-1} · A_{sym}이 positive definite라면,
$\lambda_i > 0$ ⇒ A_{sym} 역행렬존재

X가 full-rank라면
⇓
· $X^T X$는 posi definite
$\therefore \lambda_i \geq 0$ ⇒ $X^T X$ 역행렬 존재 가능

· 정방행렬 (A)의 역행렬 loop...

· 대칭행렬 (A_{sym} or $X^T X$) 의 역행렬 loop ...

full rank ↘
$\lambda_i \neq 0$ ↗ $D \neq 0$

A_{sym} ↘ (if)
양의 정부호 → $\lambda_i > 0$ → $D \neq 0$
$X^T X$ ↗ (X=full rank)

Q. 대칭행렬에선, 양의 정부호라면, full-rank가 되나?
ans) $A_{sym} = \sum \lambda_i A_i$ (A_{sym}은 rank-1 행렬의 합이니까 full..?)

SVD

$A = U\Sigma V^T$

· A = 직사각

U = 직교행렬 1
(선형독립)

V = 직교행렬 2
(선형독립)

Σ = 대각행렬
(특이값 행렬)

$\lambda_i = \sigma_i^2$
(A^TA로 확인가능)

· $U = AA^T$의 고유벡터 set
(직교, 선형독립)

· $V = A^TA$의 고유벡터 set
(직교, 선형독립)

고유분해

$A = V\Lambda V^{-1}$

A = 정방행렬

V = 고유벡터 set

· V^{-1} 존재해야 대각화 가능

inverse 조건 : square + full rank

∴ V^{-1} 존재하려면,
V가 full-rank 여야 역행렬
가능 (대각화 가능)

· V가 full-rank
= 모든 열벡터 (고유벡터)가
선형독립 이어야!
즉, 기저벡터 가능해야!
(선형독립이어서...)

고유분해(대칭)

$A = V\Lambda V^T$

A = 대칭행렬

V = 고유벡터 set
((정규)직교, 선형독립)
실수 고유값

· $V^TV = I$

· Asym 이,
positive def-이면,
$\lambda_i > 0$, inverse 가능

(물론, Asym이
posi-def 가 아니면,
$\lambda_i < 0$)

고유분해(분산)

$A^TA = V\Lambda V^T$

A^TA = 분산행렬

V = 고유벡터 set
((정규)직교, 선형독립)
실수 고유값

· A^TA는 항상
$\lambda_i \geq 0$

· A가 full-rank
라면,
A^TA
positive def
$\lambda_i > 0$
역행렬 존재

· 만약, A가 양의 정부호 라면,
$U\Lambda U^T = U\Sigma U^T$
(고유분해 = 특이분해)

대칭행렬 : 실수 고유값
양의정부호 행렬 : 양의 실수 고유값
(대칭행렬 중 일부, 분산행렬)

03. 04 특이값 분해

1) 특이값과 특이벡터 (03.04 1page)

$N \times M$ 크기의 행렬 A를 다음과 같은 3개의 행렬의 곱으로 나타내는 것을 **특이분해(singular-decomposition)** 또는 **특잇값 분해(singular value decomposition)**라고 한다.

$$A = U \Sigma V^T \tag{3.4.1}$$

여기에서 U, Σ, V는 다음 조건을 만족해야 한다.

- 대각성분이 양수인 대각행렬이어야 한다. 큰 수부터 작은 수 순서로 배열한다.

$$\Sigma \in \mathbf{R}^{N \times M} \tag{3.4.2}$$

- U는 N차원 정방행렬로 모든 열벡터가 단위벡터이고 서로 직교해야 한다.

$$U \in \mathbf{R}^{N \times N} \tag{3.4.3}$$

- V는 M차원 정방행렬로 모든 열벡터가 단위벡터이고 서로 직교해야 한다.

$$V \in \mathbf{R}^{M \times M} \tag{3.4.4}$$

위 조건을 만족하는 행렬 Σ의 대각성분들을 **특잇값(singular value)**, 행렬 U의 열벡터들을 **왼쪽 특이벡터**(left singular vector), 행렬 V의 행벡터들을 **오른쪽 특이벡터**(right singular vector)라고 부른다.

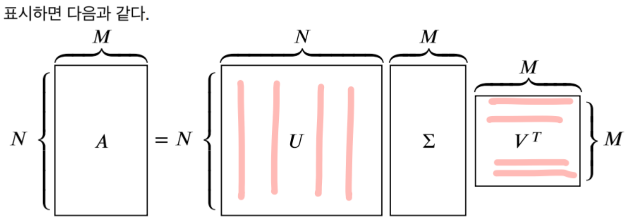
[정리] 특이분해는 모든 행렬에 대해 가능하다. 즉 어떤 행렬이 주어지더라도 위와 같이 특이분해할 수 있다.

2) 특이값 분해 행렬의 크기 (03.04 3page)

- 특이값 분해는 모든 행렬에 대해 가능하다.

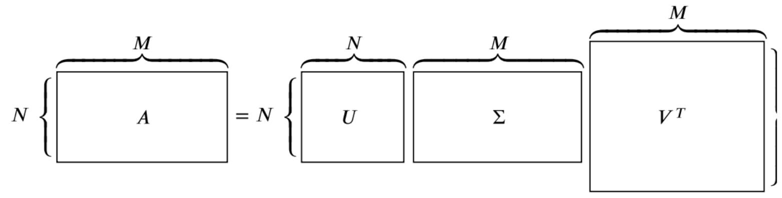
(고유값 분해는 정방행렬A에 대해 + 고유벡터set이 가역적일 때만 대각화 가능)

행렬의 크기만 표시하면 다음과 같다.



(3.4.7)

또는



3) 특이값 분해의 축소형 (03.04 3page)

예제

행렬 A

3X2

$$A = \begin{bmatrix} 3 & -1 \\ 1 & 3 \\ 1 & 1 \end{bmatrix} \tag{3.4.9}$$

3x3 3X2 3X2 2X2 2X2 2X2

는 다음처럼 특이분해할 수 있다.

$$A = \begin{bmatrix} -\frac{1}{\sqrt{6}} & \frac{2}{\sqrt{5}} & -\frac{1}{\sqrt{6}} \\ -\frac{2}{\sqrt{6}} & -\frac{1}{\sqrt{5}} & -\frac{2}{\sqrt{30}} \\ -\frac{1}{\sqrt{6}} & 0 & \frac{5}{\sqrt{30}} \end{bmatrix} \begin{bmatrix} \sqrt{12} & 0 \\ 0 & \sqrt{10} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix} \tag{3.4.10}$$

이 열은 있으나 마나!
(축소대상)

특잇값 분해의 축소형

특잇값 대각행렬에서 0인 부분은 사실상 아무런 의미가 없기 때문에 대각행렬의 0 원소 부분과 이에 대응하는 왼쪽(혹은 오른쪽) 특이벡터들을 없애고 다음처럼 축소된 형태로 해도 마찬가지로 원래 행렬이 나온다.

N이 M보다 큰 경우에는 왼쪽 특이벡터 중에서 $u_{M+1}, \cdots, u_N$을 없앤다.

$$A = \begin{bmatrix} u_1 & u_2 & \cdots & u_M \end{bmatrix} \begin{bmatrix} \boxed{\sigma_1} & 0 & 0 & \cdots & 0 \\ 0 & \boxed{\sigma_2} & 0 & \cdots & 0 \\ 0 & 0 & \boxed{\sigma_3} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \\ 0 & 0 & 0 & \cdots & \boxed{\sigma_M} \end{bmatrix} \begin{bmatrix} \boxed{v_1^T} \\ \boxed{v_2^T} \\ \vdots \\ \boxed{v_M^T} \end{bmatrix}$$

N이 M보다 작은 경우에는 오른쪽 특이벡터 중에서 $v_{N+1}, \cdots, v_M$을 없앤다.

$$A = \begin{bmatrix} u_1 & u_2 & \cdots & u_N \end{bmatrix} \begin{bmatrix} \boxed{\sigma_1} & 0 & 0 & \cdots & 0 \\ 0 & \boxed{\sigma_2} & 0 & \cdots & 0 \\ 0 & 0 & \boxed{\sigma_3} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \boxed{\sigma_N} \end{bmatrix} \begin{bmatrix} v_1^T \\ v_2^T \\ \vdots \\ v_N^T \end{bmatrix}$$

03. 04 특이값 분해

4) 파이썬을 활용한 특이분해(꼭 해보기) (연습문제 3.4.1)

5) 특이값과 특이벡터의 관계 (03.04 7page)

고유분해	특이분해
$A = V \Lambda V^{-1}$	$A = U \Sigma V^T$
$AV = V\Lambda$	$AV = U\Sigma$
$Av_i = \lambda_i v_i$	$Av_i = \sigma_i u_i$

즉, i번째 특잇값 σ_i와 특이벡터 u_i, v_i는 다음과 같은 관계가 있다.

$$Av_i = \sigma_i u_i \quad (i = 1, \ldots, \min(M, N))$$

이 관계는 고유분해와 비슷하지만 고유분해와는 달리 좌변과 우변의 벡터가 다르다.

6) 특이분해와 고유분해의 관계

아무행렬 $A = U\Sigma V^T$, ⇒ 분산행렬화 $A^TA = (U\Sigma V^T)^T(U\Sigma V^T)$

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

$= V\Sigma^2 V^T$ 고유분해

특이분해한 A는 분산행렬화하면,

$\lambda_i = \sigma_i^2$ 인 고유분해가 된다.

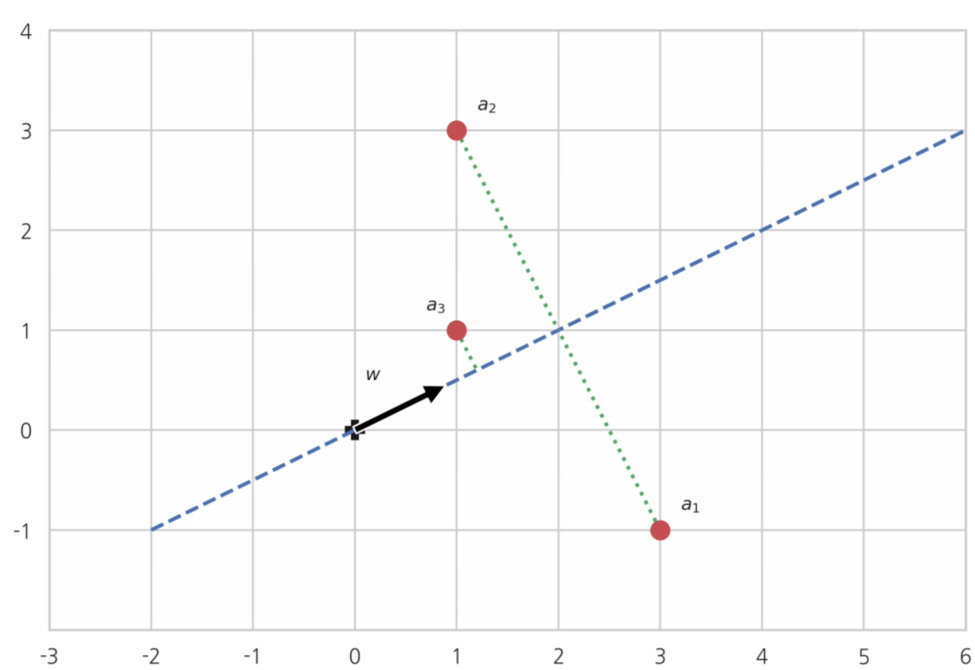
$(A^TA = V\Sigma^2 V^T = V \Lambda V^T)$

7) 1차원 근사

- (03.04 10page, 11page)

1차원 근사

2차원 평면 위에 3개의 2차원 벡터 a_1, a_2, a_3가 있다. 원점을 지나면서 모든 점들과 가능한 한 가까이 있는 직선을 만들고 싶다면 직선의 방향을 어떻게 해야 할까? 직선의 방향을 나타내는 단위 벡터를 w라고 하자.



* 부분행렬 $\boxed{A}\boxed{B}$

① $\boxed{\equiv}\boxed{B} = \begin{bmatrix} -a_1^TB- \\ \vdots \\ -a_n^TB- \end{bmatrix}$

② $\boxed{A}\boxed{||||} = \begin{bmatrix} Ab_1 & \cdots & Ab_n \end{bmatrix}$

③ $\boxed{||||}\boxed{\equiv} = |- + |- + \cdots$ 스칼라처럼

* $\sum (a_i^T w)^2 = \|Aw\|^2$

- $\begin{bmatrix} -a_1^T- \\ -a_2^T- \\ -a_3^T- \end{bmatrix} w = \begin{bmatrix} a_1^T w \\ a_2^T w \\ a_3^T w \end{bmatrix}$ (A)

- $\|Aw\| = \sqrt{(a_1^Tw)^2 + \cdots + (a_3^Tw)^2}$

$\|Aw\|^2 = (a_1^Tw)^2 + \cdots + (a_3^Tw)^2$

$= \sum (a_i^Tw)^2$

벡터 w와 점 a_i의 거리의 제곱은 다음처럼 계산할 수 있다.(연습 문제 3.1.9)

$$\|a_i^{\perp w}\|^2 = \|a_i\|^2 - \|a_i^{\parallel w}\|^2 = \|a_i\|^2 - (a_i^T w)^2 \qquad (3.4.34)$$

벡터 a_1, a_2, a_3를 행벡터로 가지는 행렬 A를 가정하면

$$A = \begin{bmatrix} a_1^T \\ a_2^T \\ a_3^T \end{bmatrix} \qquad (3.4.35)$$

행벡터의 놈의 제곱의 합은 행렬의 놈이므로 모든 점들과의 거리의 제곱의 합은 행렬의 놈으로 계산된다. (연습 문제 2.3.2)

$$\begin{aligned} \sum_{i=1}^{3} \|a_i^{\perp w}\|^2 &= \sum_{i=1}^{3} \|a_i\|^2 - \sum_{i=1}^{3} (a_i^T w)^2 \\ &= \|A\|^2 - \|Aw\|^2 \end{aligned} \qquad (3.4.36)$$

점 a_i의 위치가 고정되어 있으므로 행렬 A의 놈 값은 고정되어 있다. 따라서 이 값이 가장 작아지려면 $\|Aw\|^2$의 값이 가장 크게 만드는 w를 찾아야 한다.이 문제는 다음처럼 수식으로 쓸 수 있다.

$$\arg\max_w \|Aw\|^2 \qquad (3.4.37)$$