아니 그래서, 행렬 $\begin{bmatrix} a & c \\ b & d \end{bmatrix}$은 도대체 뭐야?

I. 대각화는 행렬의 주민번호!

HUN LEARNING × ESC

ESC 21-겨울방학
선형대수 직관 얻기 프로젝트
@hun_learning

벡터란 무엇인가?

대충 $x = [x_1, x_2, \dots, x_n]^T$ 이런 것들…?

그냥 화살표 같은 거…?

스칼라는 크기만, 벡터는 크기와 방향을 모두 가진 어쩌고…?

아니다! 집어치워라! 잊어버려라! 너를 헷갈리게 할 뿐이다!
이런 기생충같은 오해들 때문에 선형대수가 어려워지는 거다!!

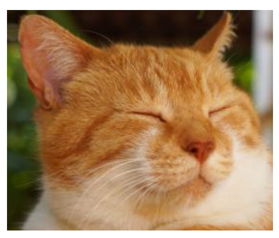
수학을 통한 계몽은 바로 정의를 받아들이는 것에서 시작한다.
벡터는 벡터공간의 원소이다!

[Definition]
다음의 성질을 만족하는 집합을, 필드 F에서 정의된 벡터공간 V라고 한다.

1. $(+,\times)$가 적절히 잘 정의됨
2. $u, v \in V \rightarrow (u+v) \in V$ (Closed under addition)
3. $k \in F, v \in V \rightarrow k \times v \in V$ (Closed under scalar multiplication)

이때 필드의 원소를 "스칼라", 벡터공간의 원소를 "벡터"라고 한다.

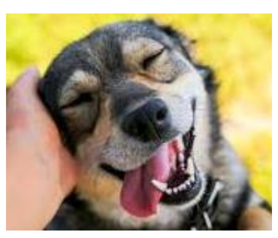
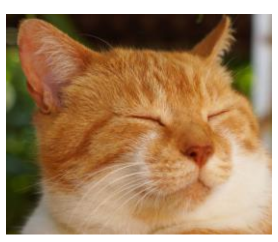
굉장히 널널한 정의다. 함 볼까?

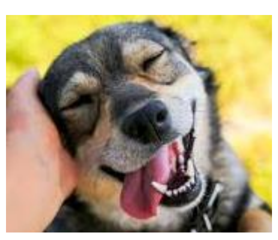
Let $V = \{$  ,  ,  ,  , …. $\}$

and $F = \{$  ,  , ….. $\}$

If we can show that, given $(+,\times)$ is well defined *(which is, of course, not..)*,

1.  $+$  $\in V$

2.  $\times$  $\in V$

then  $\in V$ is a vector, and  $\in F$ is a scalar.

벡터란 무엇인가?

웃음기 빼고 예시를 들면…

$$P_2 = \{ax^2 + bx + c \mid a, b, c \in R\}$$

최대 n차인 다항식의 공간은 실수에서 정의된 벡터공간이다.
임의의 원소들, 예컨대

$$p_1(x) = x^2 + x$$
$$p_2(x) = 3x + 1$$

두 다항식을 더해도, 하나에 실수를 곱해도 그대로 최대 2차이기 때문.
(그러나 "정확히" 2차인 다항식의 공간은 벡터공간이 아니다. 왜?)
p_1은 벡터다. 이게 화살표인가? 크기와 방향을 가진 뭐시기인가?

흔히 벡터하면 떠오르는 것은 사실 "좌표"이다. 벡터공간은 선형 연산으로 정의되었기 때문에, 벡터공간의 모든 원소는 어떤 벡터들의 선형결합으로 쓸 수 있다. 위 예시를 보면

$$P_2 = sp\{1, x, x^2\}$$

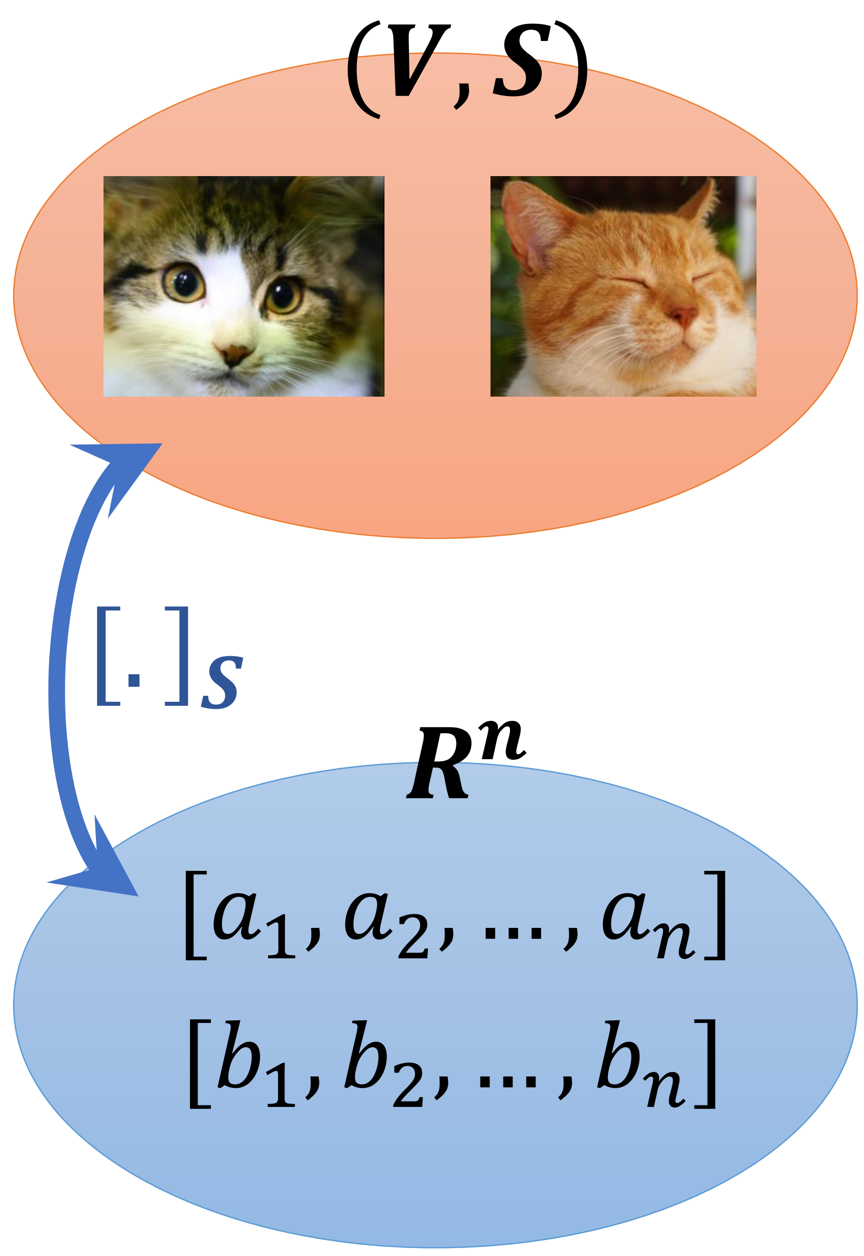
즉 모든 벡터는 세 개의 벡터 $S = \{1, x, x^2\}$의 선형결합으로 쓸 수 있다.

$$p_1(x) = (0 \times 1) + (1 \times x) + (1 \times x^2)$$

이렇게 뼈대가 되는 벡터들을 "기저"라고 한다
이 기저 S라는 잣대를 들이밀었을때 벡터 p_1의 좌표는 다음과 같은데, 좌표 자체는 R^3 벡터다.

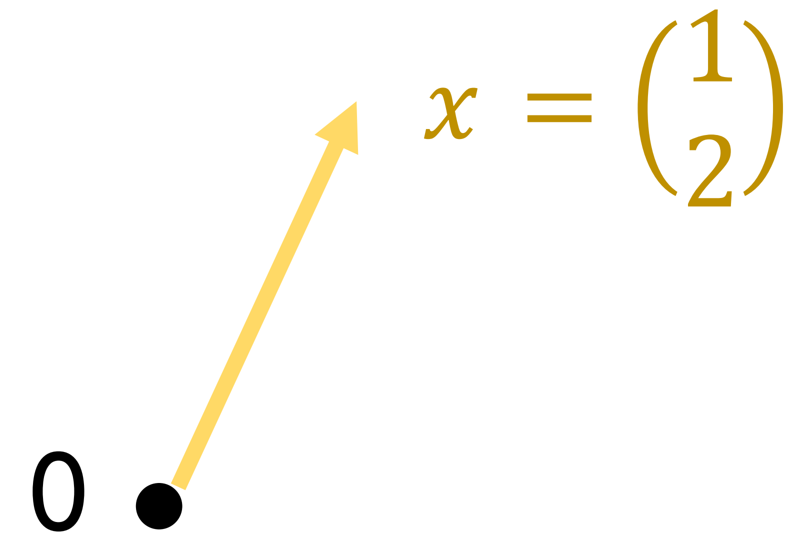
$$[p_1]_S = [0,1,1]^T \in R^3$$

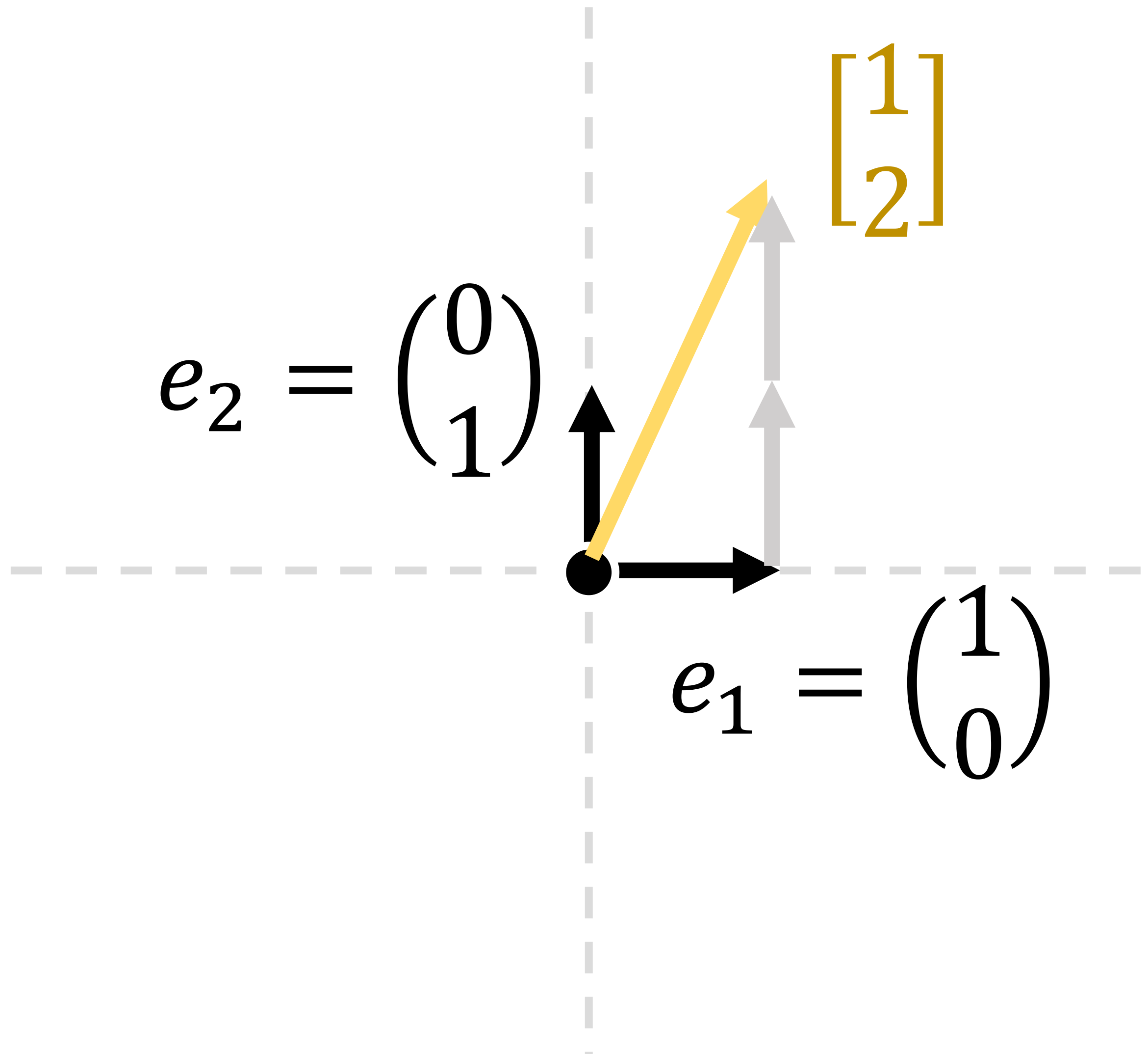
(다른 기저, 예컨대 $Q = \{2, x-1, x^2\}$에 대한 좌표는 $[p_1]_Q = [1/2,1,1]^T$로 다르다!)
어떤 벡터공간의 기저 벡터들이 유한하면 "**finite-dimensional**"이며,
모든 n차원 벡터공간과 R^n 벡터공간에는 위와 같은 대응관계, "**isomorphism**"이 존재한다.

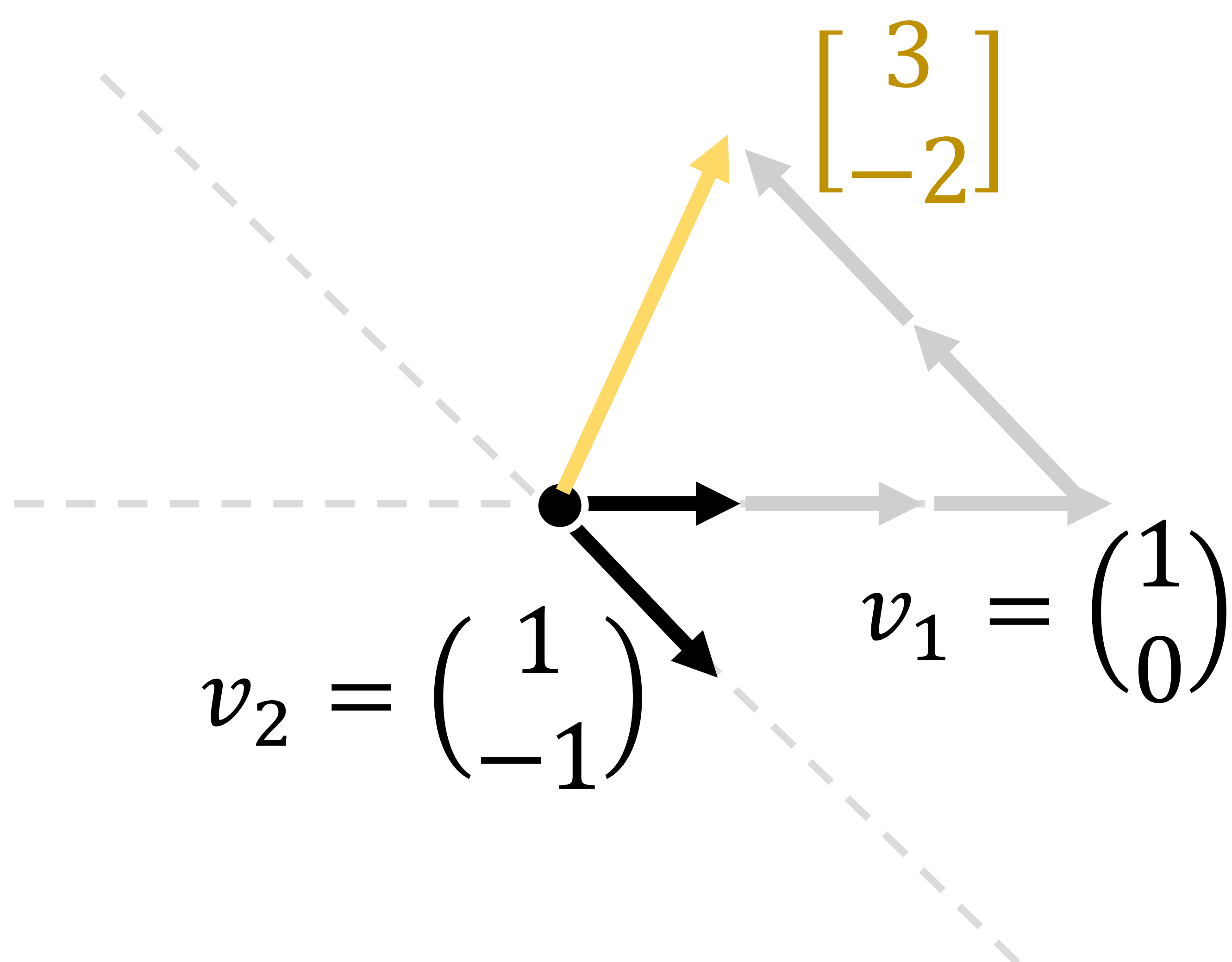


지금부터는 $\boldsymbol{V} = \boldsymbol{R}^n$에 대해서만 이야기하자.

그냥 덩그러니 놓여진 벡터 $x \in \boldsymbol{R}^2$
이 벡터의 주소, **"[좌표]"**는
내가 어떤 기저에서 보냐에 따라,
즉 어떤 좌표평면에서 보냐에 따라 다르다.







$$x = 1e_1 + 2e_2$$
$$\rightarrow [x]_S = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

S 기저에서의 x의 좌표는 $\begin{bmatrix} 1 \\ 2 \end{bmatrix}$

$$x = 3v_1 - 2v_2$$
$$\rightarrow [x]_V = \begin{bmatrix} 3 \\ -2 \end{bmatrix}$$

V 기저에서의 x의 좌표는 $\begin{bmatrix} 3 \\ -2 \end{bmatrix}$

좌표 $[x]_S$를 $[x]_V$로, 그 반대로도 자유자재로 읽는 방법? Change of Basis

$$x = 1e_1 + 2e_2 = 3v_1 - 2v_2$$
$$= 1\begin{pmatrix} 1 \\ 0 \end{pmatrix} + 2\begin{pmatrix} 0 \\ 1 \end{pmatrix} = 3\begin{pmatrix} 1 \\ 0 \end{pmatrix} - 2\begin{pmatrix} 1 \\ -1 \end{pmatrix}$$
$$= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}[x]_S = \begin{pmatrix} 1 & 1 \\ 0 & -1 \end{pmatrix}[x]_V$$

$$[x]_V = \begin{pmatrix} 1 & 1 \\ 0 & -1 \end{pmatrix}^{-1}[x]_S$$
$$[x]_S = \begin{pmatrix} 1 & 1 \\ 0 & -1 \end{pmatrix}[x]_V$$

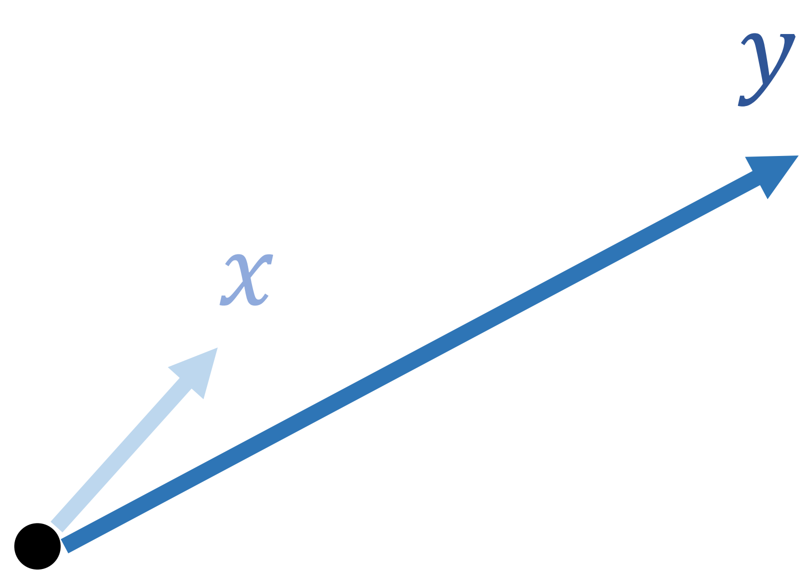
때문에,

$$\boldsymbol{[I]_{S,V} = (v_1 \quad \cdots \quad v_n)^{-1}}$$
$$\boldsymbol{[I]_{V,S} = (v_1 \quad \cdots \quad v_n)}$$

선형 연산과 행렬의 탄생

$T(x) = y$ 오른쪽 그림과 같은 걸 하고 싶다.

직접 T를 하지 말고,
S 기저에서 y의 좌표 $[y]_S$ 만 알 수 있을까?



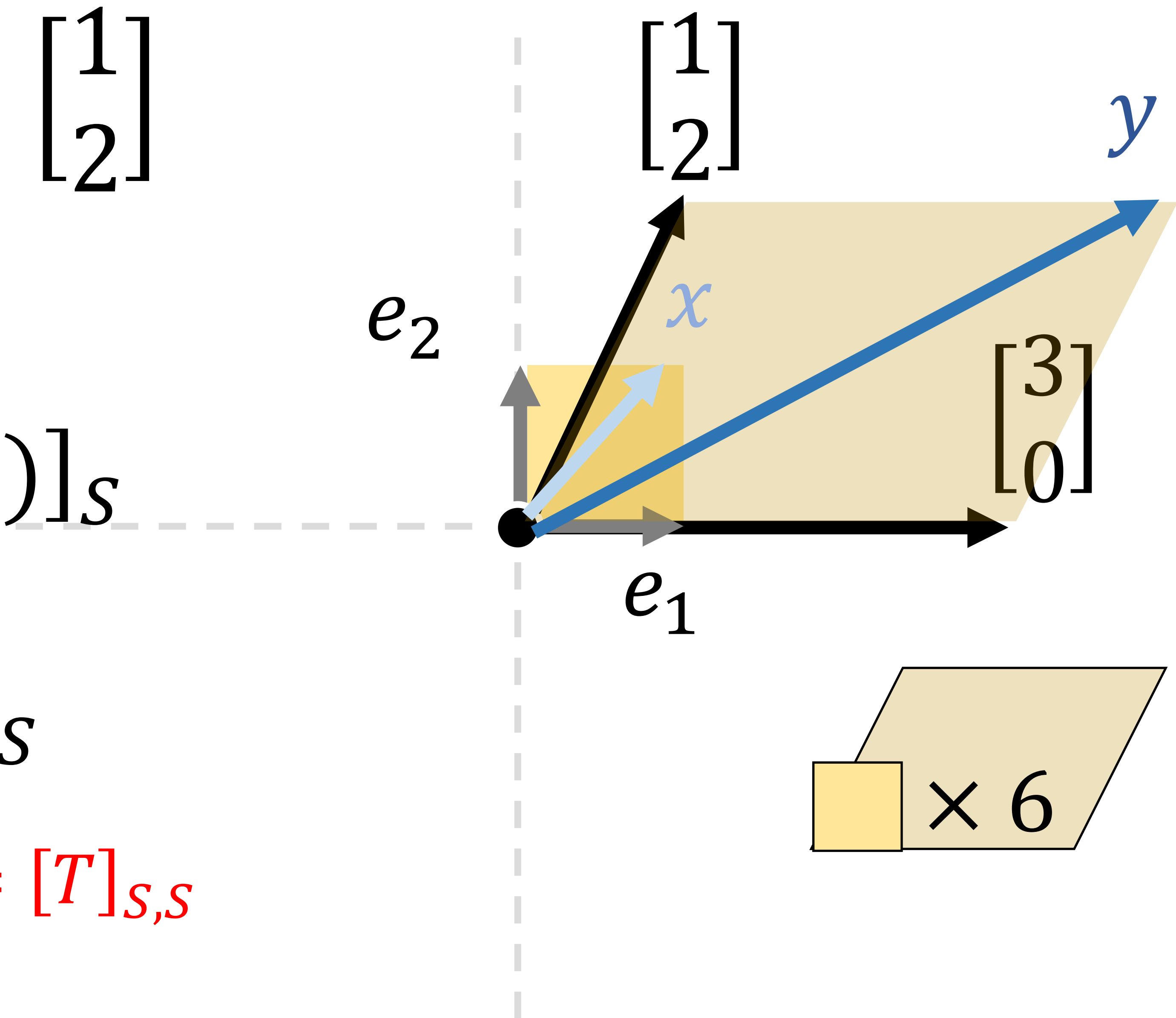
어차피 x는 기저 e_1, e_2 의 선형 결합이다.
먼저 T를 기저 e_1, e_2에 조지고 좌표를 보자.

$$[T(e_1)]_S = \begin{bmatrix}3\\0\end{bmatrix}, [T(e_2)]_S = \begin{bmatrix}1\\2\end{bmatrix}$$

그럼 임의의 x에 대해

$$[T(x)]_S = [T(x_1e_1 + x_2e_2)]_S$$
$$= [x_1T(e_1) + x_2T(e_2)]_S$$
$$= x_1[T(e_1)]_S + x_2[T(e_2)]_S$$
$$= \begin{bmatrix}3 & 1\\0 & 2\end{bmatrix}[x]_S$$

이때 $\begin{bmatrix}3 & 1\\0 & 2\end{bmatrix} = [T]_{S,S}$

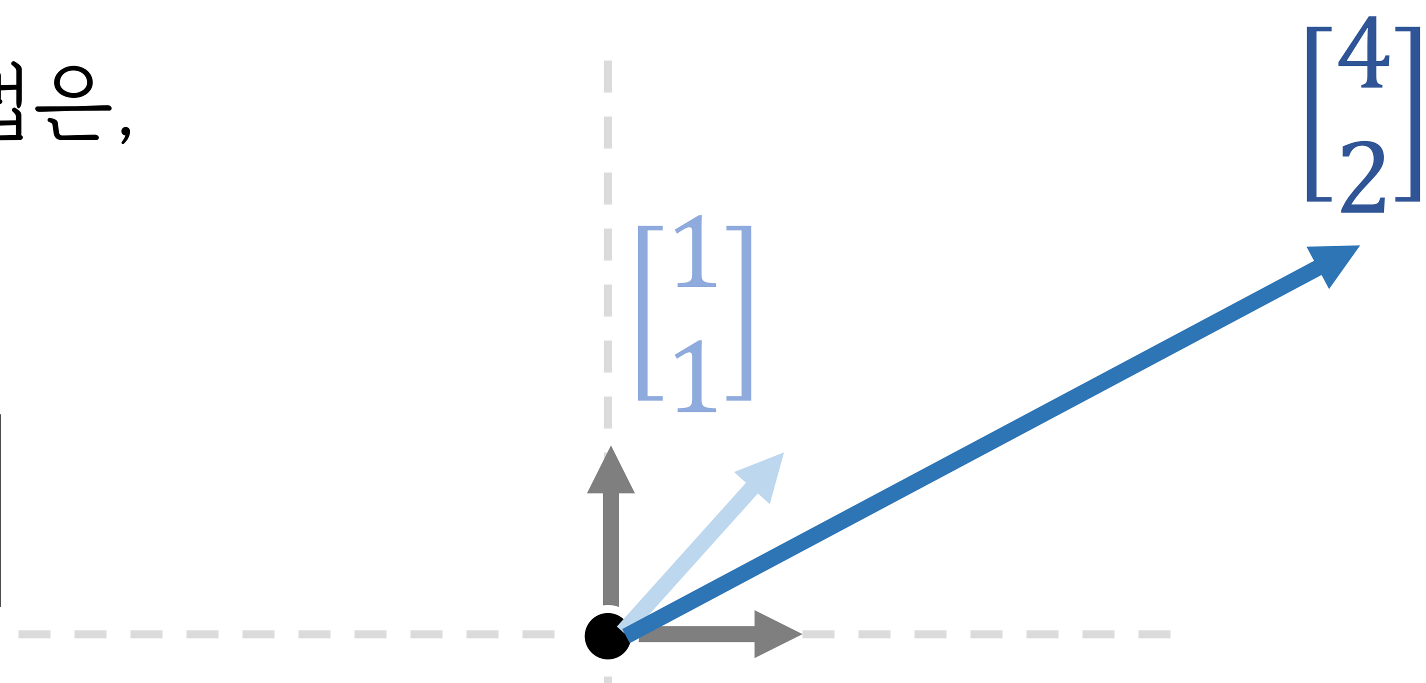


직접 T를 하지 말고,
S 기저에서 y의 좌표 $[y]_S$ 만 아는 방법은,

First, read x in S: $[x]_S = \begin{bmatrix}1\\1\end{bmatrix}$

And multiply $\begin{bmatrix}3 & 1\\0 & 2\end{bmatrix}\begin{bmatrix}1\\1\end{bmatrix} = \begin{bmatrix}4\\2\end{bmatrix}$

Then y in S is: $[y]_S = \begin{bmatrix}4\\2\end{bmatrix}$

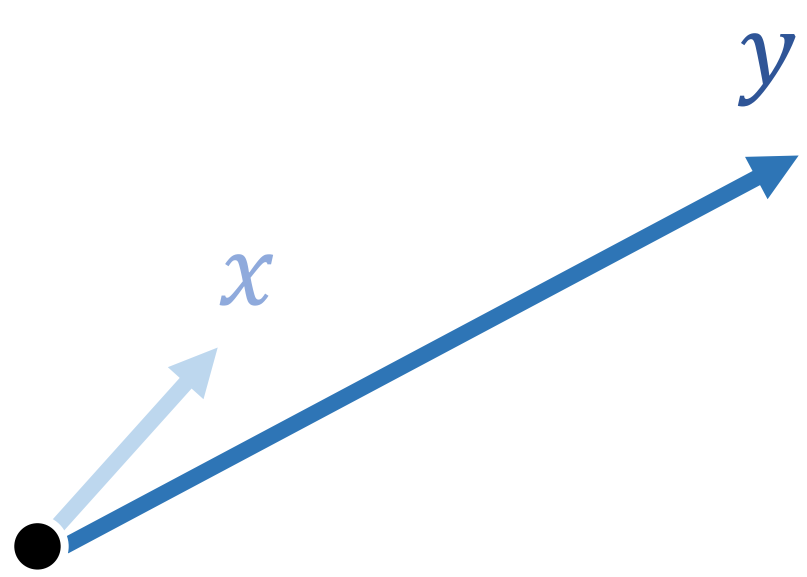


행렬의 목적은 바로 선형변환을 묘사하는 것!
1) 선형변환 T, 2) 정의역의 기저, 3) 공역의 기저에 의해 결정된다!

선형 연산과 행렬의 탄생

$T(x) = y$ 오른쪽 그림과 같은 걸 하고 싶다.

직접 T를 하지 말고,
V 기저에서 y의 좌표 $[y]_V$ 만 알 수 있을까?



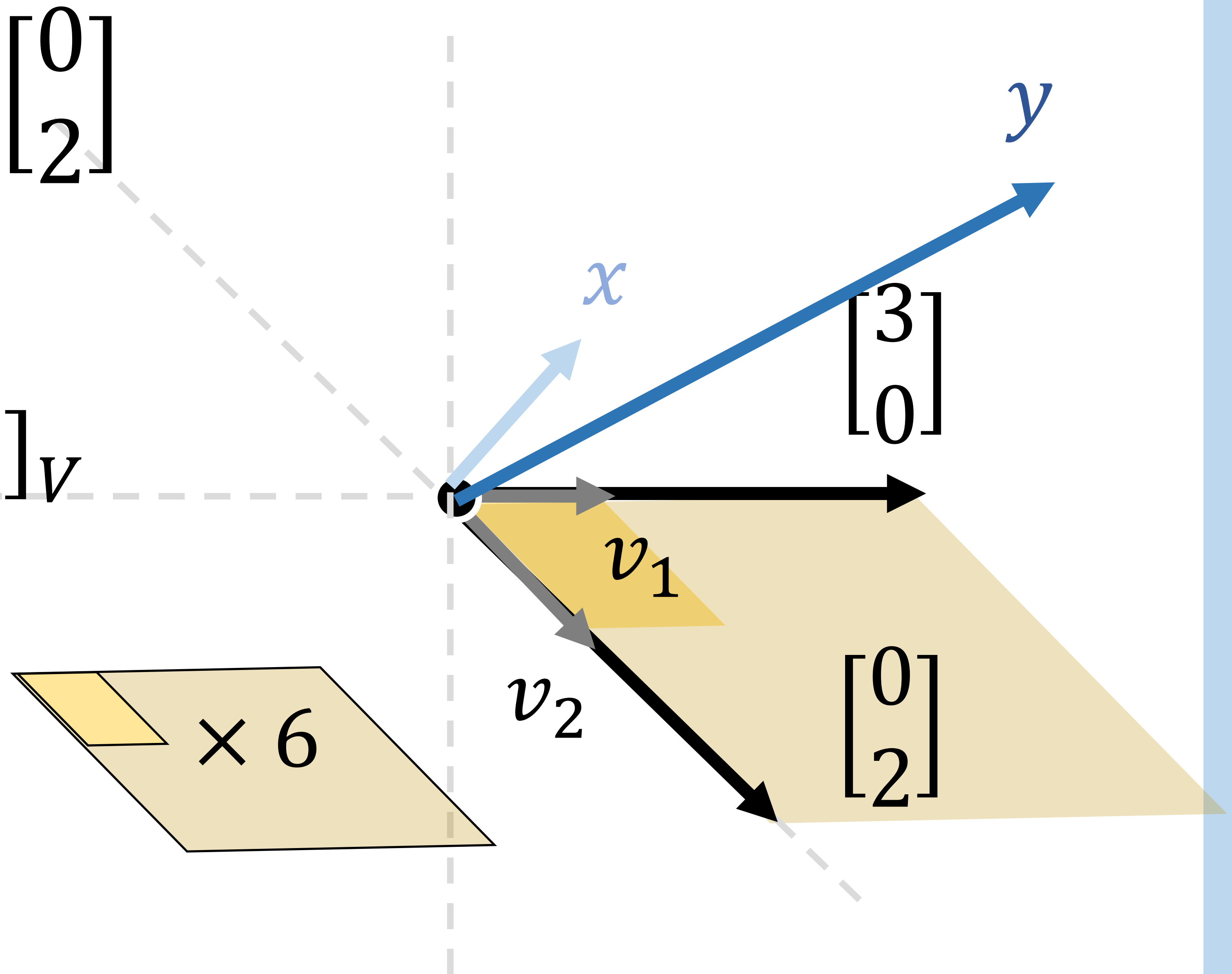
어차피 x는 기저 v_1, v_2 의 선형 결합이다.
먼저 T를 기저 v_1, v_2에 조지고 좌표를 보자.

$$[T(v_1)]_V = \begin{bmatrix}3\\0\end{bmatrix}, [T(v_2)]_V = \begin{bmatrix}0\\2\end{bmatrix}$$

그럼 임의의 x에 대해

$$[T(x)]_V = [T(x_1v_1 + x_2v_2)]_V$$
$$= \cdots$$
$$= \begin{bmatrix}3 & 0\\0 & 2\end{bmatrix}[x]_V$$

이때 $\begin{bmatrix}3 & 0\\0 & 2\end{bmatrix} = [T]_{V,V}$라고 쓸 수 있다.

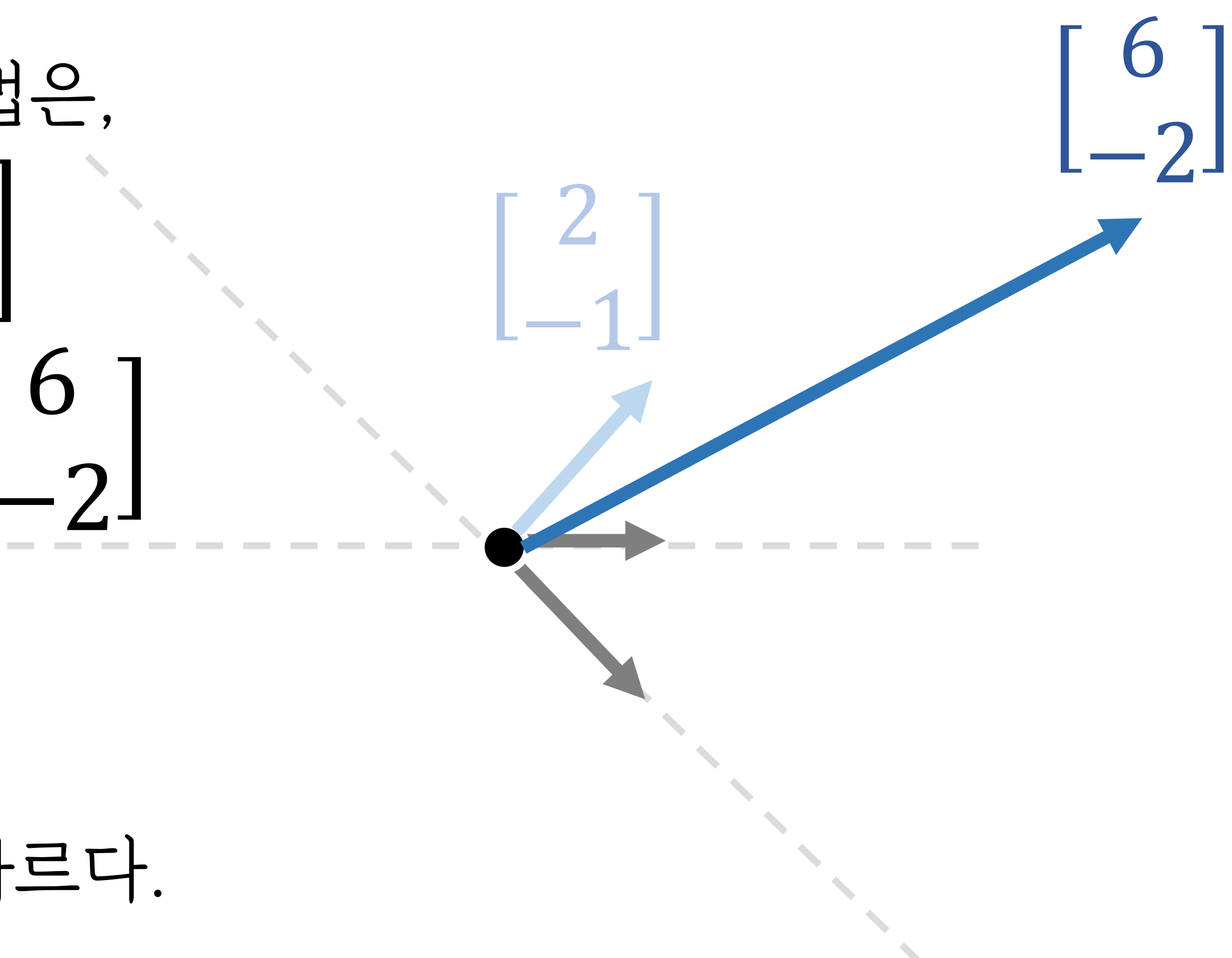


직접 T를 하지 말고,
V 기저에서 y의 좌표 $[y]_v$ 만 아는 방법은,

First, read x in S: $[x]_V = \begin{bmatrix}2\\-1\end{bmatrix}$

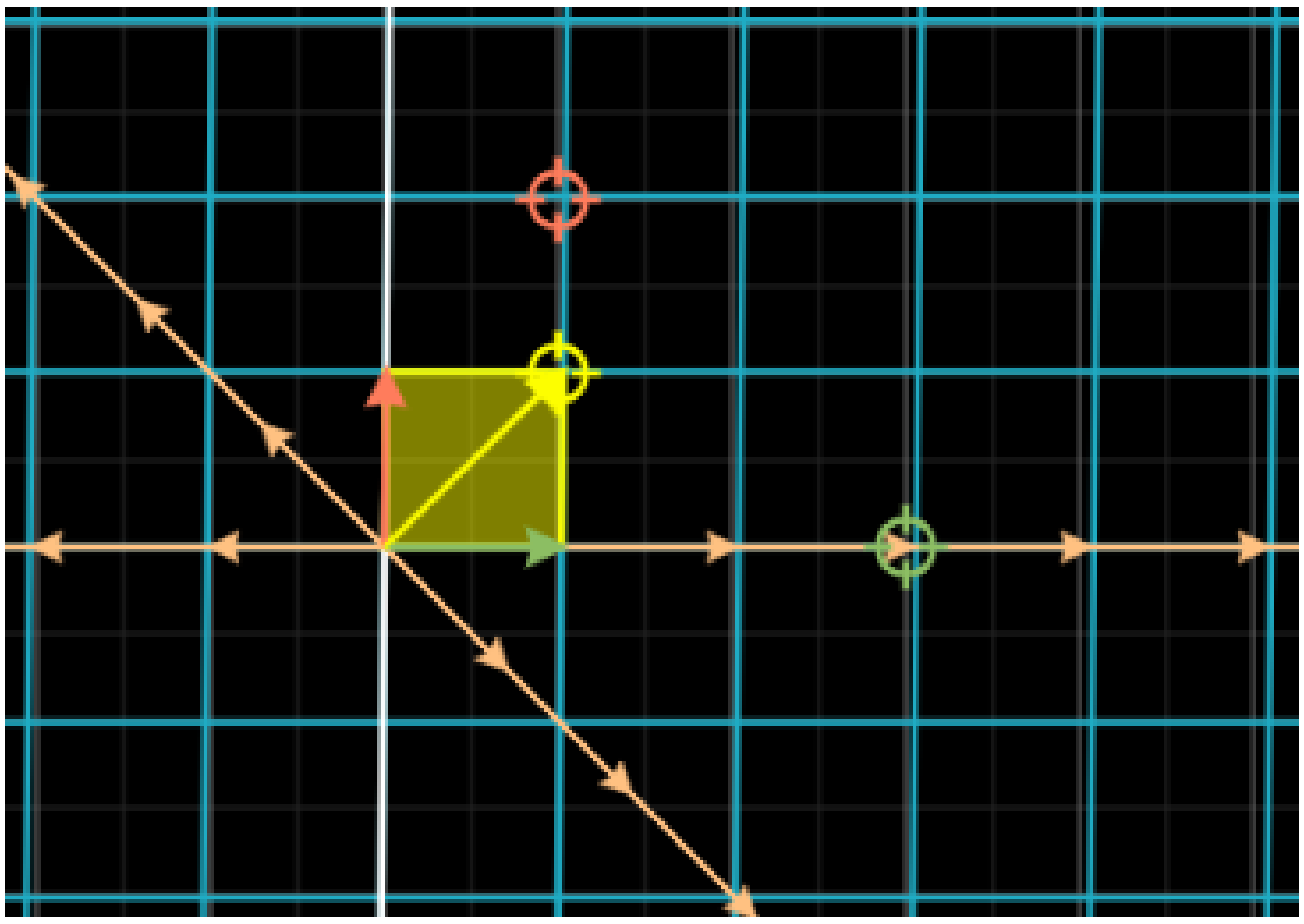
And multiply $\begin{bmatrix}3 & 0\\0 & 2\end{bmatrix}\begin{bmatrix}2\\-1\end{bmatrix} = \begin{bmatrix}6\\-2\end{bmatrix}$

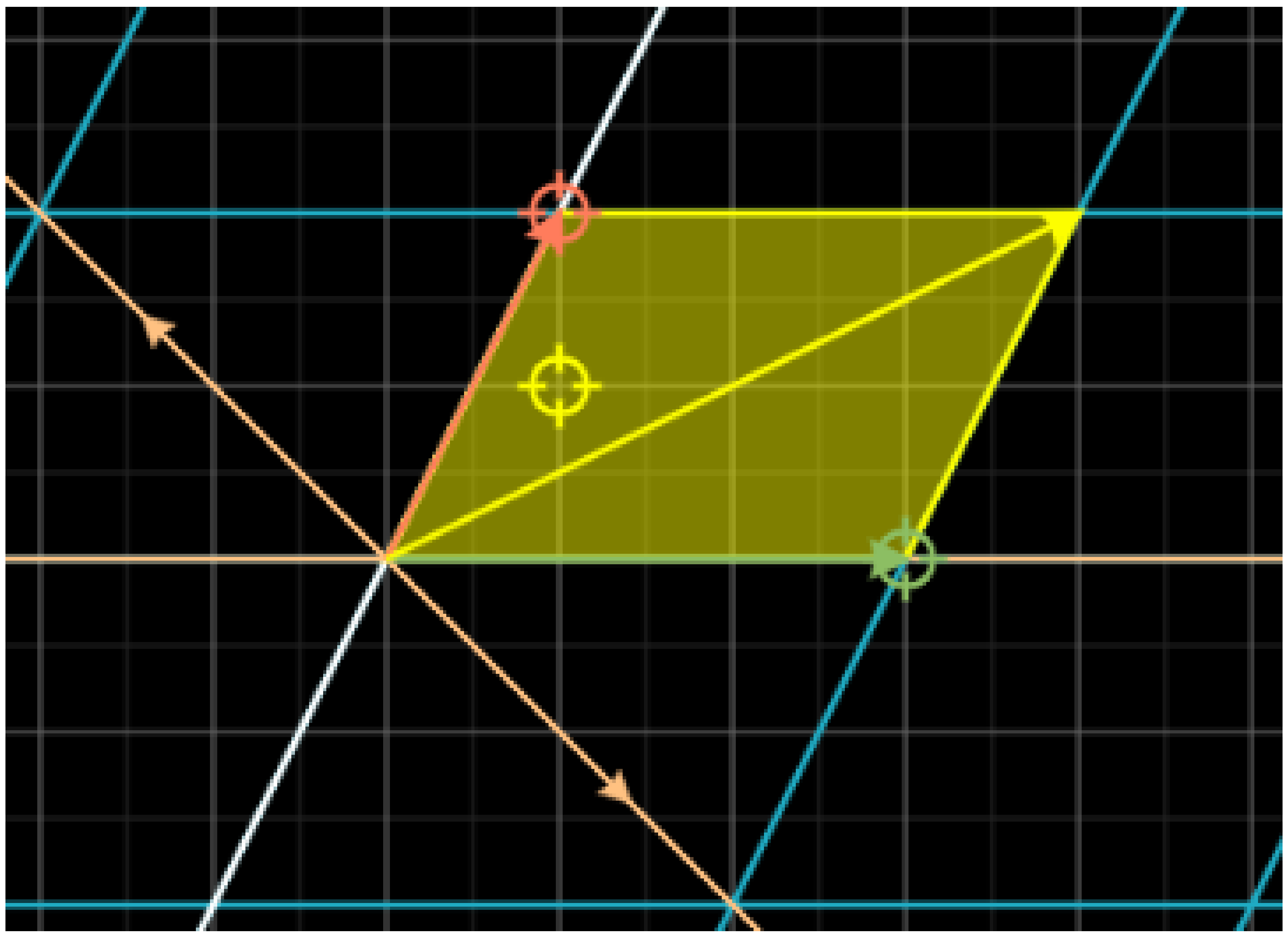
Then y in V is: $[y]_V = \begin{bmatrix}6\\-2\end{bmatrix}$



아까와 똑같은 선형연산인데 행렬이 다르다.
정의역과 공역의 기저가 다르기 때문!
근데, 얘는 왜 대각행렬일까? v_1, v_2이 선형변환의 고유벡터이기 때문!

DIAGONALIZATION $T: R^2 \to R^2$ by $\begin{pmatrix} 3 & 1 \\ 0 & 2 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}$





Interactive Matrix Visualization (shad.io)

선형변환 T 은 좌표평면의 선형적인 "왜곡"이다.
좌표평면이 방향이 틀어지고, 늘어나거나 줄어들면서 위치를 바꾼다.

그러나 그 와중에도 도도하게 자기 자리를 유지하고,
오직 늘어나거나 줄어드는 등 스케일만 바뀌는 벡터들이 있다.
이 벡터들을 선형변환 T의 **축**, **"고유벡터"**라고 하고,
이 축을 따라서 좌표평면이 느는/주는 **스케일**을 **"고유값"**이라고 하며,
기저가 만드는 단위공간이 느는/주는 스케일이 **행렬의 크기**, **"행렬식"**이다.

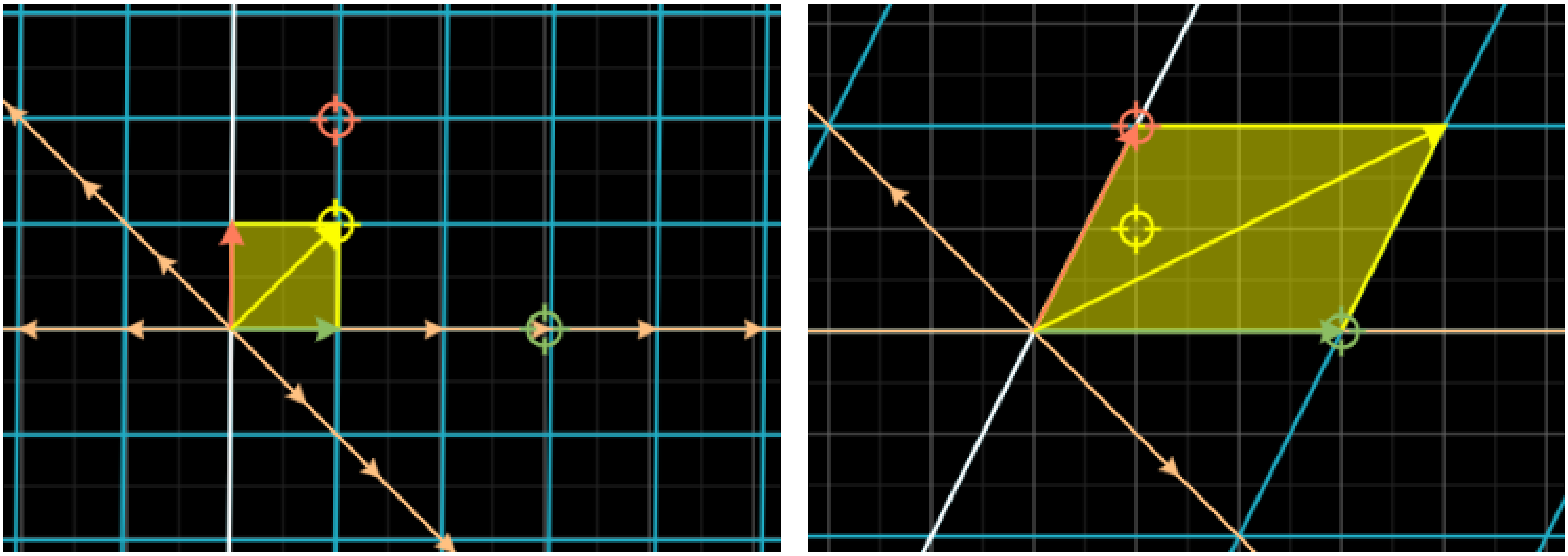
위 그림과 같은 선형변환을 행렬로 나타내는 데는 두 가지 방법이 있다.

1. 표준기저 $e_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$, $e_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$를 기저로 삼는다.
   이들이 변환 이후에 어디로 가는지 $T(e_1), T(e_2)$를 추적하고,
   그 좌표를 다시 표준기저로 기록한다.
   $$[y]_S = \begin{bmatrix} 3 & 1 \\ 0 & 2 \end{bmatrix}[x]_S$$

2. 고유벡터 $v_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$, $v_2 = \begin{pmatrix} 1 \\ -1 \end{pmatrix}$ 를 대체기저로 삼는다. (Change of Basis)
   이들은 고유벡터이므로 변환을 해도 고유값만큼만 곱해지고 끝.
   즉 $T(v_1) = \lambda_1 v_1, T(v_2) = \lambda_2 v_2$ 이다. 그 결과를 다시 표준기저로 읽어준다.

$$[y]_S = \begin{bmatrix} 1 & 1 \\ 0 & -1 \end{bmatrix}\begin{bmatrix} 3 & 0 \\ 0 & 2 \end{bmatrix}\begin{bmatrix} 1 & 1 \\ 0 & -1 \end{bmatrix}^{-1}[x]_S$$

DIAGONALIZATION $\quad T: R^2 \to R^2 \ by \begin{pmatrix} 3 & 1 \\ 0 & 2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}$



Interactive Matrix Visualization (shad.io)

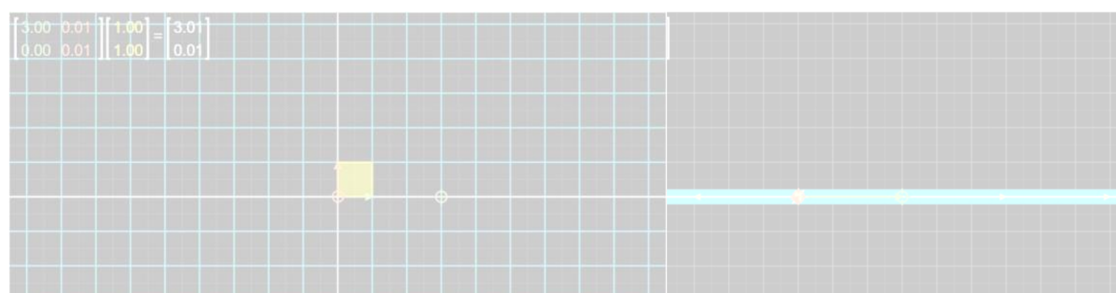
$$[y]_S = \begin{bmatrix} 3 & 1 \\ 0 & 2 \end{bmatrix} [x]_S = \begin{bmatrix} 1 & 1 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} 3 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 0 & -1 \end{bmatrix}^{-1} [x]_S$$

이 결과를 일반화하면 다음과 같이 쓸 수 있다.

$$[T]_{S,S} = [I]_{V,S}[T]_{V,V}[I]_{S,V}$$

좀더 세련되게 쓰면, DIAGONALIZATION:

$$A = VDV^{-1}$$



A가 singular → 축을 따라서 뭉개짐!, $|A| = 0$

대각화의 의미는 무엇일까?

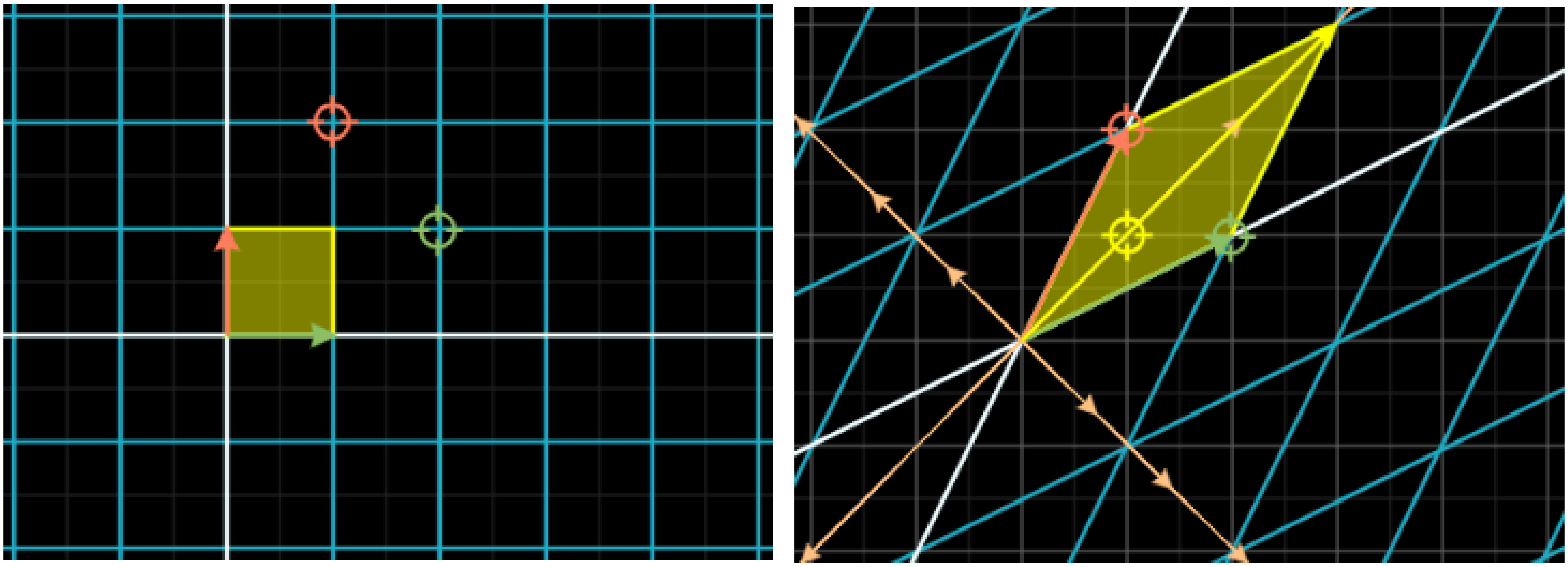
대수적으로는 ⋯
어떤 행렬 A의 고유값 d_i과 고유벡터 v_i를 구하면 $AV = VD$로 쓸 수 있다.
이때 $v_1, \ldots, v_n$ 고유벡터들이 모두 독립이면 V^{-1}이 존재하므로,
위 식과 같이 쓸 수 있는 것이다.

그림으로 상상해보면 ⋯
A가 나타내는 **선형변환의 축이 모두 살아있으면**, 선형변환의 엑기스를
선형변환의 축 V과 축에서의 스케일 D로 요약할 수 있는 것이다!
이런 의미에서 **대각화는 행렬을 해부**하는 것과 같다!

그러나…

1. 역행렬 구하는 것은 너무 빡세다. $A = VDV^{-1}$

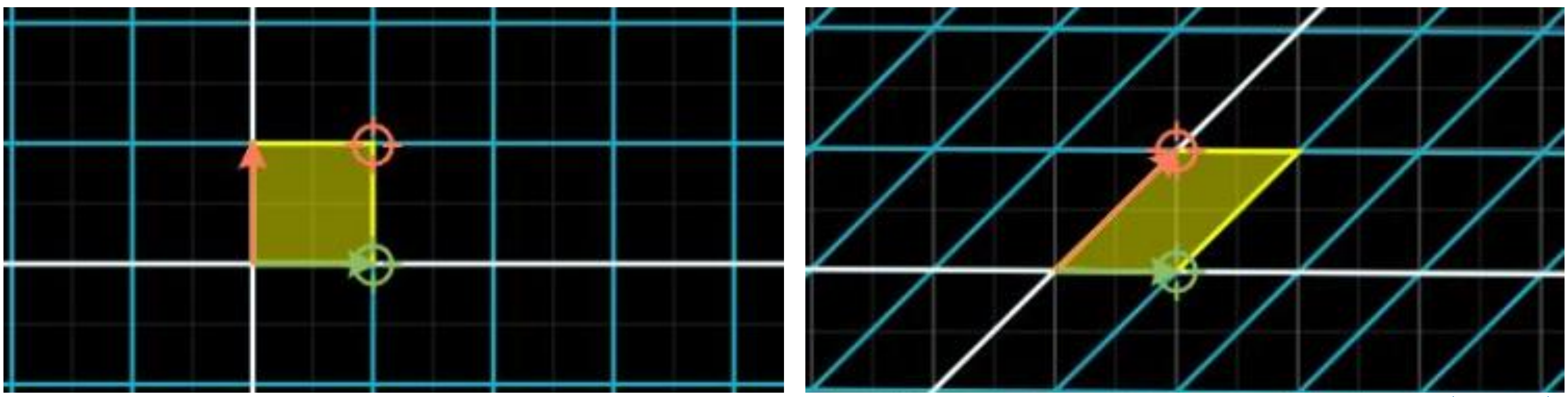
그러나 $A = A^T$이면, A의 선형변환의 축들이 모두 **"직교"**하게 된다.
즉 $v_i \cdot v_j = 0\ (i \neq j)$ 이므로 $VV^T = V^TV = I \rightarrow V^T = V^{-1}$ **직교대각화**



Interactive Matrix Visualization (shad.io)

2. 모든 행렬이 대각화 가능한 것은 아니다.

$$T: R^2 \rightarrow R^2\ \ by \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}$$



Interactive Matrix Visualization (shad.io)

선형 변환의 축이 하나가 없다.. 이런 애들을 "defective"

근데 이런 애들도 결국 A^TA 이렇게 어거지로 대칭으로 만들어줄 수 있고, 이건 당연히 직교대각화 가능하다. 이를 이용하면 A도 대각화가 가능하다. Singular Value Decomposition

아니 그래서,
행렬 $\begin{bmatrix} a & c \\ b & d \end{bmatrix}$ 은
도대체 뭐야?

II. 왜 대칭행렬을 좋아할까?

HUN LEARNING × ESC

ESC 21-겨울방학
선형대수 직관 얻기 프로젝트
@hun_learning

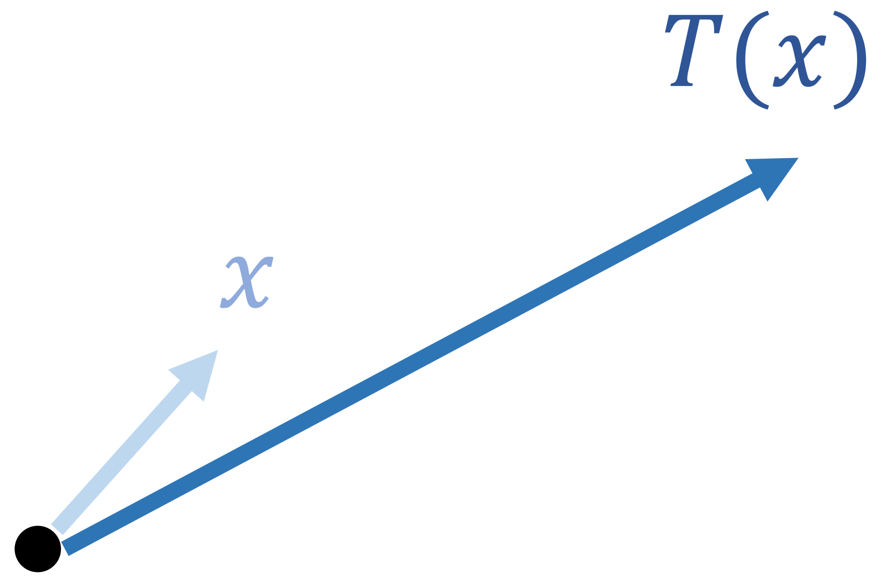
아니 그래서,
행렬 $\begin{bmatrix} a & c \\ b & d \end{bmatrix}$은
도대체 뭐야?

III. 돌리고 당기고 돌리고 SVD!

HUN LEARNING × ESC

ESC 21-겨울방학
선형대수 직관 얻기 프로젝트
@hun_learning

대각화 리뷰



$$T: (R^2, \boldsymbol{\alpha}) \to (R^2, \boldsymbol{\beta}) \ \ by\ T(x) = y.$$

Find $[T]_{\alpha,\beta}$ s.t. $[T(x)]_\beta = \boldsymbol{[T]_{\alpha,\beta}}[x]_\alpha$

대각화 문제의 핵심은,
어떻게 기저 α, β를 깔쌈하게 잡아서 $[T]_{\alpha,\beta}$를 이쁘게 만들어볼까? 이다.

그냥 우리가 보는 흔해빠진 (대각화가 되지 않은 행렬)을 $[T]_{S,S}$로 보고,
n개의 독립인 고유벡터를 찾아 기저 V를 만들어,

$$[T]_{S,S} = [I]_{V,S}[T]_{V,V}[I]_{S,V} \iff A = VDV^{-1}$$

를 만들어줬고, 이때 고맙게도 A가 대칭이면 $A = VDV^T$가 되었다.

근데 이때 **정의역과 공역의 기저가 같을 필요는 없다**.
즉 똑같은 R^n에 대해서 여러가지 수많은 기저가 있는데,
그 중에서 $\boldsymbol{[T]_{\alpha,\beta}}$가 절묘하게 대각화가 되는 그런 기저가 있으며,
그것을 찾는 방법을 Singular value decomposition이라고 부른다.

SVD의 핵심은 바로 니가 어떤 A를 갖고 와도 A^TA는 대칭이므로 대각화가 가능하다는 점이다. 때문에 SVD는 A가 full rank이든 아니든, 정방이든 아니든 하튼 뭐든 간에 아주그냥 유니바샬리 하게 성립하는 테크닉이다.

SVD를 대수적으로 보이는 것은 헛웃음이 나올 정도로 쉽다.
문제는 직관이다. 미리 말하면, SVD의 핵심은 다음과 같다.

"돌리고! 땡기고! 돌리고!"

먼저 SVD를 어떻게 하는지 보고, 저게 뭔 말인지 그림으로 살펴보자.

SVD를 이해하면 행렬을 압축하는 여러 알고리즘들을 쉽게 이해할 수 있다.

SVD는 그냥 말장난임

아무렇게나 생긴 A (n $\geq p$)

p

$n \quad A \in R^{n \times p}$

진짜 아무거나 다 됨.
찌그러져도 (p ≠ n),
열벡터 선형독립
아니어도 노상관

그래도 $A^T A$는 대각행렬

$$p \; \underset{p}{A^T A} = p \; A^T \; \underset{p}{A}$$

당연히 $A^T A$는 직교대각화 가능

$A^T A$의 모든 고유값은 ≥ 0 $\quad$ v_i들은 모두 오쏘노말, $R^p = sp\{v_1, \dots, v_p\}$

$$A^T A = \begin{bmatrix} v_1 & \cdots & v_p \end{bmatrix} \begin{bmatrix} \sigma_1^2 & & \\ & \ddots & \\ & & \boxed{\sigma_p^2} \end{bmatrix} \begin{bmatrix} v_1 & \cdots & v_p \end{bmatrix}^T$$

A가 not full rank이면 $A^T A$는 singular, 이중에 몇 개는 0

지금부터 편치라인

1. $\boldsymbol{Av_i \in R^n}$들도 서로 직교하네!

$$(Av_i) \cdot (Av_j) = A^T A v_i \cdot v_j = \sigma_i^2 v_i \cdot v_j = 0 \; (i \neq j)$$

오쏘노말하게 하기 위해 $\mathrm{u_i} = Av_i / \sigma_i$ 라고 해보자.

2. $\boldsymbol{u_i = \sigma_i^{-1} A v_i}$들로 $\boldsymbol{R^n}$`의 <u>오쏘노말 기저를!</u>

$n = p$이면 편안하게 $R^n = sp\{u_1, \dots, u_p\}$

$n > p$이면? ㄱㅊㄱㅊ 부족한건 암거나 채워. 오쏘노말하게만 하면 ok

$$R^n = sp\{\boxed{u_1, \dots, u_p}, \boxed{u_{p+1}, \dots, u_n}\}$$ 훼이크다!

v_i들에서 가져옴

이제부턴 그냥 말장난, $\sigma_i u_i = A v_i$이므로

$$\underbrace{\begin{bmatrix} u_1 & \cdots & u_p & \cdots & u_n \end{bmatrix}}_{U(n \times n)} \underbrace{\begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_p \\ & & \end{bmatrix}}_{\Sigma(n \times p)} = \underset{n \times p}{A} \overbrace{\begin{bmatrix} v_1 & \cdots & v_p \end{bmatrix}}^{V(p \times p)}$$

$n < p$이면? $A^T = V \Sigma U^T$구하고 transpose!

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

$A = U\Sigma V^T$, 그래서 뭔 말인데?

$$\underset{n \times p}{A} = \underset{U(n \times n)}{\begin{bmatrix} u_1 & \cdots & u_p & \cdots & u_n \end{bmatrix}} \underset{\Sigma(n \times p)}{\begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_p \\ & & \end{bmatrix}} \underset{V(p \times p)}{\begin{bmatrix} v_1 & \cdots & v_p \end{bmatrix}}^T$$

1. Change of Basis로 이해!

Change of Basis를 기억하는가?

$$[\boldsymbol{I}]_{S,V} = (\boldsymbol{v_1} \quad \cdots \quad \boldsymbol{v_n})^{-1}$$
$$[\boldsymbol{I}]_{V,S} = (\boldsymbol{v_1} \quad \cdots \quad \boldsymbol{v_n})$$

SVD는 딱 그 꼴이다. U를 R^n의 기저, V를 R^p의 기저로 생각하면,

$$A = U\Sigma V^T \Leftrightarrow [T]_{S,S} = [I]_{U,S}[T]_{V,U}[I]_{S,V}$$

V는 오쏘노말 기저니까 $V^{-1} = V^T$

T라는 선형변환을

1. 정의역 기저 S, 공역 기저 S를 고르면 노잼 존못 $[T]_{S,S} = A$
2. 정의역 기저 V, 공역 기저 U를 고르면 대유잼 존잘 $[T]_{V,U} = \Sigma$

어떻게 이런 깔쌈한 대각화가 가능한 것인가?

대각화 : $A = [T]_{S,S}$이라는 행렬은 선형변환을 "표준기저가 변환 후에 어디로 갈지"를 정의함으로써 프로그램한다. 그러나 이 선형변환은

$$\boldsymbol{Av_i = \lambda_i v_i}$$
$$(AAv_i = \lambda_i^2 v_i)$$

이므로, v_i라는 변환의 축과 변환의 세기 λ_i를 안다면 선형변환을 아는 것.

SVD: $A = [T]_{S,S}$라는 선형변환이 다음을 만족함에 주목한다.

$$\boldsymbol{Av_i = \sigma_i u_i}$$
$$(A^T A v_i = \sigma_i^2 v_i)$$

A가 대각화가 가능하지 않아도 $A^T A$는 대각화가 되며,
따라서 선형변환을, $A^T A$의 고유벡터가 A 변환 이후에 어디로 가는지를 지정하는 것으로 프로그램할 수 있다.
이때 $u_i = \sigma_i^{-1} A v_i$ 라는 기저를 쓰면, 어디로 가는지를 단순히 u_i 의 상수배로 나타낼 수 있다. 때문에 대각행렬이 나오는 것!

$A = U\Sigma V^T$, 그래서 뭔 말인데?

$$\underset{n \times p}{A} = \underbrace{\left[\begin{array}{ccc|ccc} u_1 & \cdots & u_p & & \cdots & u_n \end{array}\right]}_{U(n \times n)} \underbrace{\begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_p \\ \hline & & \end{bmatrix}}_{\Sigma(n \times p)} \underbrace{\begin{bmatrix} v_1 & \cdots & v_p \end{bmatrix}^T}_{V(p \times p)}$$

2. 전부 다 $[T]_{S,S}$으로 이해!

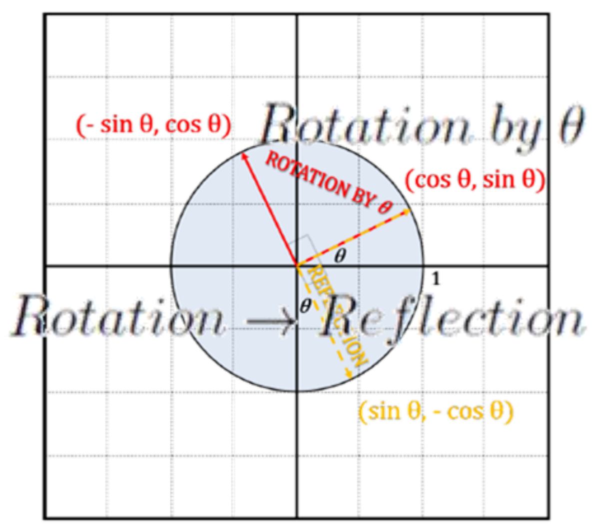
$$A = U\Sigma V^T \Leftrightarrow [T]_{S,S} = [R_2]_{S,S}[E]_{S,S}[R_1]_{S,S}$$

그냥 처음부터 끝까지 S 표준기저에 대해서만 생각하자. 그럼 U, V^T가 하는 역할은 Change of Basis가 아니라, **표준기저 $e_1, e_2, .. e_p$를 또 다른 직교하는 기저들로 바꿔주는 선형연산**으로 이해할 수 있다.

(왜? $U, \tilde{V} = V^T$ 모두 Orthogonal matrix, 즉 $U^TU = I, \tilde{V}^T\tilde{V} = I$ 이므로, 각각의 열벡터끼리는 서로 오쏘노말하기 때문!)

Orthonormal Basis

Orthonormal basis is actually either a **rotation** of the standard basis, **reflection**, or both. We can see this in $\mathbb{R}^2$, but the intuition holds for higher dimensions.



$$\begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$

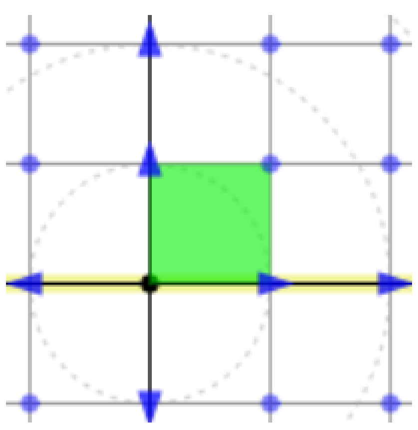
$$\begin{bmatrix} \cos\theta & \sin\theta \\ \sin\theta & -\cos\theta \end{bmatrix}$$

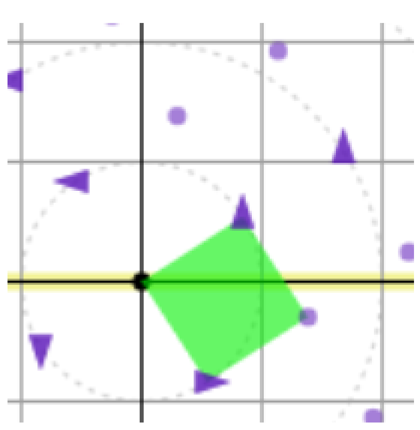
[훈러닝 직관적인 선형대수](#)

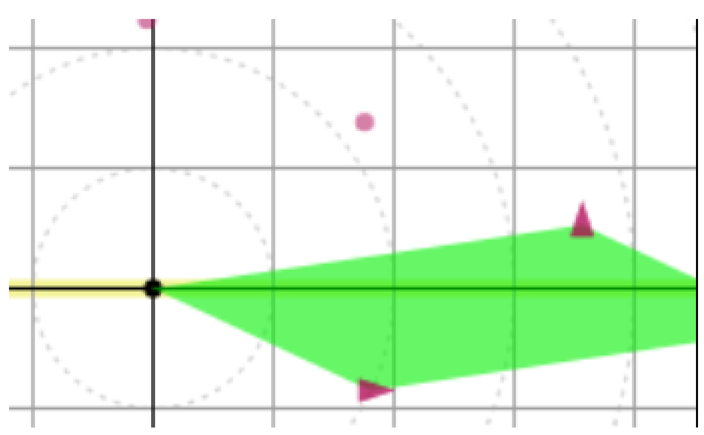
SVD의 핵심:

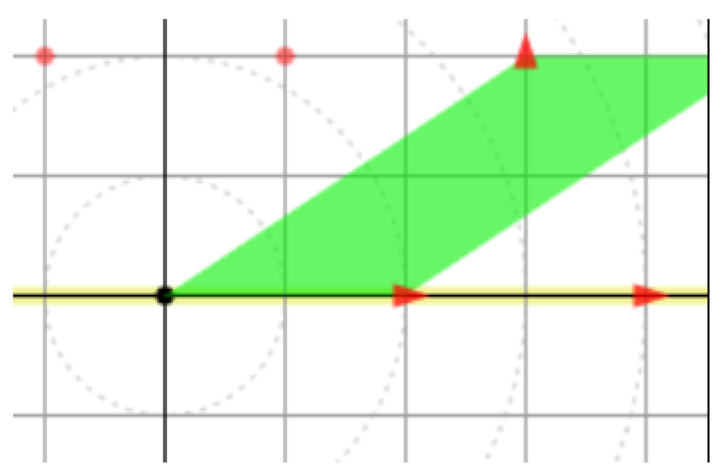
1. 하나의 직교기저들이 다른 직교기저로 이동하는 것은, 좌표평면이 회전하는 것으로 생각할 수 있다. (Rotate)
2. 거기에다가 (준)대각행렬 $[E]_{S,S}$를 곱하는 것은 그냥 벡터를 늘리는 거고 (Extend)
3. 다시 또 다른 직교기저로 회전시키는 것 (Rotate)

$$\begin{bmatrix} 2 & 3 \\ 0 & 2 \end{bmatrix} = \begin{bmatrix} 2/\sqrt{5} & -1/\sqrt{5} \\ 1/\sqrt{5} & 2/\sqrt{5} \end{bmatrix} \begin{bmatrix} 4 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{5}} & -\frac{2}{\sqrt{5}} \\ \frac{2}{\sqrt{5}} & \frac{1}{\sqrt{5}} \end{bmatrix}^T$$









[Linear Transform Visualizer (sakura.ne.jp)](#)