모든 것은 데이터로 되어있다.

따라서, 모든 것은 벡터값으로 표현할 수 있다.

$$\begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \cdots & a_{mn} \end{pmatrix}$$

행렬은 정의상 직사각형 형태로 되어있음

$$\begin{bmatrix} a_1 \\ \vdots \\ a_m \end{bmatrix} \begin{bmatrix} a_1 & \cdots & a_n \end{bmatrix}$$

가로는 column이라고 부르고 세로는 row이라고 부른다.

이 행렬의 크기를 얘기할 때 demention(차원)의 이 차원은 공간으로 전환되고 벡터 공간는 m*n의 형태 벡터들이 될 수 있는 모든 값으로 이뤄져 있는 공간(= 차원 / N개의 componant가 그 차원을 다 채우면 그 차원의 vector space 하나를 이룬다.)을 뜻하며 이것이 바로 column 측면에서는 m차원 row 측면에서는 n차원이 된다. 즉, 하나의 행렬은 m이라는 차원과 n이라는 차원을 whole space로 가진다.

공간은 whole space와 column/row space으로 나눌 수 있는데 whole space 란 이 벡터 자체가 가지고 있는 차원 전체를 뜻하고 column/row space는 이 column 혹은 row가 채울 수 있는 차원 전체를 뜻한다.

또한 null space가 있는데 whole space에서 어떤 column/row space를 정의하고 나서 그 나머지를 null space라고 한다.<기하학적 정의>

어떤 행렬이 있을 때 무엇을 곱하든 0으로 이루어진 행렬이 될 때, 이 모든 값들이 이루는 공간을 null space이라고 한다. <수학적 정의>

-> column서는 왼쪽에다 곱해야지 차원이 같아지기 때문에 left null space라고 한다. Row의 입장에서는 오른쪽에서 곱해야지 차원이 같아지기 때문에 null space라고 한다. 그리고 하나의 행렬에서 column입장에서 null space를 빼고 row입장에서 null space를 빼면 둘다 같은 차원을 가진다. 즉, whole space는 달라도 둘의 차원은 같다는 뜻이다.

Cf)

[영공간

Ax=0의 모든 해 집합을 행렬 A의 **영공간(null space)**이라고 합니다. A가 n개 열을 가졌다면 A의 영공간은 Rn의 부분공간이 됩니다.

영공간을 **선형변환(linear transformation)** 관점에서 이해할 수도 있습니다. T를 n차원 벡터 x를 m차원 영벡터로 변환하는 선형변환으로 둔다면 영공간 NulA는 아래 그림처럼 도식화할 수 있습니다.]

M*N인 행렬이 있을때

M은 column의 관점 / column space

-> M차원이 column의 whole space

만약에 column space가 whole space를 다 채우지 못한다면 이는 null space라고 한다.

<Null space는 그 벡터 값에 곱하면 0을 만들어주는 행렬>

또한 모든 column이 독립적이면 (한 일직선상에 존재하는 행렬이 하나식만 존재) column space는 whole space다.

여기서 독립적인 행렬은 whole space가 m차원이면 최대 m개까지만 존재가능하다.

-> m+1차원 행렬은 m차원의 행렬의 조합으로 표현가능하다.

만약에 column vector가 두개가 있다면 행렬의 덧셈으로 그 vector 값을 구할 수 있다.

이는 곧 두 vector에서의 평행사변형 꼴의 한 점이 된다.

Vector가 한 직선상으로 되어있지 않으면 두개의 rank으로 되어있는 행렬이다.

N은 row의 관점 -> N차원이 row의 whole space

2차원 row가 3개가 있다고 하면 2개의 조합으로 나머지 하나를 만들 수 있기 때문에 3개 전부 independent할 수 없다. 결국, n차원에서 independent한 벡터는 총 n개만 존재한다.

요약)

R^m –> m이 column의 whole space-> column space

R^n -> n이 row의 whole space -> row space

-> 그리고 둘의 차원은 같아야 한다. (whole space는 달라도)

그리고 이 모든 벡터 값은 이를 좌표축 상에 표시가 가능하다. 여기서 벡터는 크기와 방향성만 중요하지 이 벡터가 꼭 3,4의 위치에 있어야 하는 것은 아니다.

크기와 방향성만 정해지면 어디로 가든지 상관이 없다.

즉, 원점에서 시작할 필요가 없다는 뜻이다.

또한, 벡터 크기는 차원을 늘릴 뿐 점을 늘리지는 않는

다. 반대로, 벡터에 곱해진 값(상수)은 벡터의 길이에만 영향을 주지 크기에 영향을 주지 않는다. 이를 Spaning이라고 한다. 예를 들어 column 벡터 두개에서 이를 확장하면 하나의 삼각형이 만들어지고 이 삼각형이 무수히 확장되면 column vector space를 만들 수 있다.

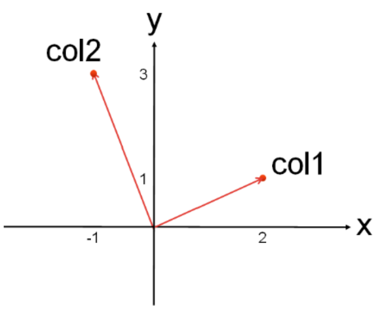
-> 상수의 곱셈은 벡터의 길이를 / 두 행렬의 덧셈은 (평행사변형 형태로)새로운 방향으로 바꿔준다. /두 행렬의 곱셈은 벡터 값의 차원을 틀어준다.

두개의 벡터값은 결합이 가능한데 이를 linear combination이라고 하고 이는 선형성(같은 직선위에 존재하는 여부 / 독립성 여부)를 판단하는 기준이 된다.

$$c \cdot v + d \cdot w$$

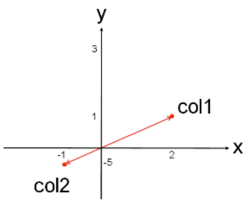
V는 벡터 W도 벡터(차원이 같음) / c,d는 곱하는 숫자

Linear combination은 이 형태만 가능



이 두개는 independent하다고 한다. -> 같은 선상에 없기 때문.

그래서 이 두개의 벡터의 결합으로 다른 점을 나타낼 수 있다. = col1과 col2의 linear combination을 통해



이렇게 한 직선상에 있으면 dependent하다고 한다.

-> 삼각형이 만들어 지지 않는다. -> 여기서 확장을 해도 line이 확장될 뿐 line 안쪽을 채울 수 없음

이를 독립성을 행렬식으로 표현하면

$$A = \begin{bmatrix} 1 & 1 & 2 \\ 2 & 1 & 3 \\ 4 & 1 & 5 \end{bmatrix}$$

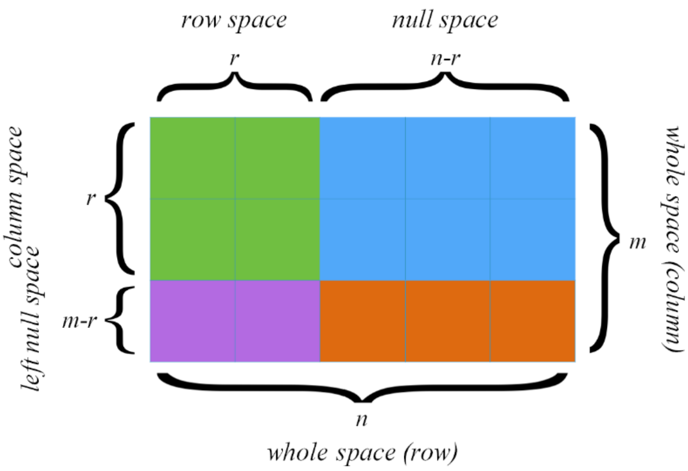
Column space -> 독립적이 아니다. -> linear combination으로 만들 수 있으면 이는 independent 하다고 할 수 없음. -> 1,2,4 / 1,1,1 이 두개에 의해 2,3,5를 만들 수 있기 때문에 2,3,5는 존재하지 않는다고 할 수 있음(만들어 지는 것이지 새로운 값이 아니다.) 결국 whole space는 3차원이지만 column space는 2차원이 된다.

$$A^T = \begin{bmatrix} 1 & 2 & 4 \\ 3 & 3 & 1 \end{bmatrix}$$

T는 transpose했다는 뜻이다. -> 3*2 -> 2*3로 바꿨다 -> whole space는 2차원 <여기서 whole space는 열의 관점에서 봤을때의 whole space / 2차원 값이 벡터가 3개 있는 것>

Colunm vector는 3개가 있는데 만들 수 있는 것은 2차원이기 때문에 column space는 2차원이다.

Transpose를 해도 여전히 whole과 vector의 개수는 달라져도 column space는 2차원



전체의 요약

m*n 행렬이 있다고 생각할 때 m 크기의 whole space<column>가 하나 있고 n 크기의 whole space<row>가 있다. M이랑 같거나 작게 column space<column>가 있고 N이랑 같거나 작게 column space<row>가 또 존재 한다. 여기서 column space의 차원을 판단하는 기준이 독립적이냐 독립적이지 않느냐에 따라 판단한다. 여기서의 나머지를 left null space / null space라고 한다.

$$Ax = b$$

X가 입력 벡터, b가 출력 벡터, A는 행렬

X의 벡터 차원과 b라는 출력 벡터의 차원은 같지 않아도 된다.

A 행렬이 그 역할을 해준다.

A라는 행렬은 x의 벡터 크기에 따라 크기가 정해지고

B라는 벡터는 A라는 행렬에 따라 크기가 정해진다.

-> A라는 행렬로 b라는 벡터가 달라지기 때문에, A를 linear transformation이라고 한다.

<차원도 바꾸고 값도 바꾸었다.>

이를 그래프 상으로 보면 basis vector의 차원을 틀어주는 것과 같다.

->

기하힉적 해석: 선형변환은 공간을 이동시키는 방법이며 격자선이 여전히 평행하고 균등간격을

유지한 변형이다. 그리고 원점은 고정되있음을 의미하고 이 변환들을 간단한 숫자들로 설명가능하다. 바로 기저 벡터들는 변환후의 좌표값이다.

행렬은 우리에게 이러한 변환을 설명하는 언어를 제공해줌.

행렬의 열들은 이 좌표 값을 나타내며, 행렬=벡터 곱셈은 단지 이것을 계산하는 방법임.

이 변환이 주어진 벡터에 적용한 결과를 보여줌.

우리가 행렬을 볼때마다 공간의 어떤 변환으로 생각가능.

$$A^{-1}b = x$$

역함수의 형태

출력에 해당되는 부분에 A의 역행렬을 집어넣어서 입력 값을 유도하는 과정

하지만, 만약에 A로 인하며 b가 일직선이 되면

여기 있는 점을 역으로 했을 때 이 점은 원래 자리로 찾아 갈 수 없음

직선에서 펼치려고 하면 어디서 왔는지, 어디로 가는 지 알 수 없음 -> 이를 invertible하다고 한다.

행렬을 배운 사람 determinant<선형대수학에서, **행렬식**은 정사각행렬에 수를 대응시키는 함수의 하나이다. 대략, 정사각행렬이 나타내는 선형 변환이 부피를 확대시키는 정도를 나타낸다.> 이 값이 0이라고 할 때 역행렬이 존재하지 않음

면적이 0일 된다면(독립적이면) -> 0이 됨과 동일해지고 왜 inverse가 존재하지 않느냐를 설명해준다.

어떤 행렬의 eigenvector란 무엇이냐?

어떤 벡터는 행렬 A와 transformation을 했음에도 불구하고 원점과의 일직선상에 있는 경우가 존재하는데 이 벡터를 eigenvector라고 한다.

행렬 A를 선형변환으로 봤을 때, 선형변환 A에 의한 변환 결과가 자기 자신의 상수배가 되는 0이 아닌 벡터를 고유벡터(eigenvector)라 하고 이 상수배 값을 고유값(eigenvalue)라 한다.