9강

Rank

- Rank of matrix $A \in R^{m \times n}$ is defined as

$$rank(A) = dim(R(A))$$

- R(A) is a subspace spanned by columns of A, so rank is the number of independent columns of A

  → Ax는 linear combinations of columns

  → 그리고 그것의 dim임

- One can show that number of independent columns and rows are same for any matrix $A \in R^{m \times n}$

  → independent columns 와 independent rows의 갯수가 사실 같다는 것을 보일 수 있다.

- Rank of A is the number of independent rows/columns

  → independent rows = independent columns

- This implies rank(A) = rank(A^T)

  → T를 취해도 어차피 같다. row와 column을 바꿔도 같다.

Rank은 A의 자유도를 말해준다. Rank가 클수록 output 정보가 더 잘 보존된다.

input이 n차원일 때 그것의 output차원을 결정하는 것이 rank이다.

- we have

$$rank(A) \le min(m, n)$$

→ rank(A)는 m과 n의 최솟값보다는 작아야 한다.

  - Why? Suppose $m \le n$.
    - R(A) is subspace spanned by vectors in R^m, so rank(A) ≤ m ≤ n

      → col들은 R^m 안에 있기에, R^m보다 작은 span은 불가능하다.

- Why? Suppose m > n.
  - R(A) is subspace spanned by n vectors in R^m, so the basis of R(A) have at most n basis: rank(A) ≤ n < m

    → 아무리 조합을 해봤자, independent한 벡터의 수가 n개 이상될 수 없다.
- Rank can be considered as degree of freedom (information) preserved by going through

  linear system A
- y = Ax. x has DoF(Degree of Freedom) n:

  going through A, output y has DoF at most m – but this cannot exceed n!

  → x가 A를 통과하면 m이상의 자유도를 얻을 수는 없다.

  → 그리고 n이상이 될 수도 없다.

ex)

$$\begin{bmatrix} 1 & 2 \\ 2 & 4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} x_1 + 2x_2 \\ 2x_1 + 4x_2 \end{bmatrix} \\ = c \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

→ R^2차원의 점x가 R^1차원의 직선 위에만 움직일 수 있게된 것이다.

→ Rank(A) = 1이다.

Conservation of Dimension

차원의 보존.

- $$rank(A) + dim(N(A)) = n(A \in R^{mXn}) \\ dim(R(x)) + dim(N(A)) = n$$

Ax = 0을 만족시키는 x의 basis가 N(A)인 것이다.

- rank(A) is the dimension 'hit' by linear mapping y = Ax
- dim(N(A)) is the dimension 'crushed' by linear mapping y = Ax

→ Ax = 0이 되어서 결과에 영향을 미치지 못하는 것이다.

→ A안의 independent한 벡터들의 갯수를 k개 라고 할 때, 다음의 성질을 만족해야 한다.

$$\begin{bmatrix} a_1^T \\ a_2^T \\ ... \\ a_k^T \end{bmatrix} x = 0$$

즉, N(A)의 basis는 R(A)의 basis와 수직해야 한다는 것이다.

→ n - k 개

- Each dimension of input is either crushed to o or remains at the output
  - n is the degree of freedom (DOF) of input x
  - rank(A) is DOF left at output y = Ax
  - dim(N(A)) is DOF lost by y = Ax

rank of matrix products

- we have

$$rank(BC) \leq min(rank(B), rank(C))$$

- So if A = BC with B ∈ R^{m×r}, C ∈ R^{r×n} then

  y = BCx = B(Cx) first DoF reduces to no more than rank(C),

  but going through B again reduces rank no more than rank(B)

  → 결과값이 rank(B), rank(C)를 넘을 수가 없는 것이다.

  x → C → B → BCx

- conversely if rank(A) = r and r ≤ m, r ≤ n,

  then A∈R^{m×n} can be factorized to A = BC with B ∈R^{m×r}, C ∈R^{r×n}

  → 쪼개는 게 가능하다.

- Here r can be considered as the width of information bottleneck

→ 쪼개는 것이 가능할 경우, x는 m과n보다도 더 작은 r을 통과하게 되는 것이다.

→ 즉, r은 병목, r만큼 좁아져야 하는 길목이 되는 것이다.

Full Rank

- if

  $$rank(A) = min(m, n)$$

  we say A has full rank

- for square matrices, full rank means nonsingular (invertible)

  → 역행렬이 존재한다.

- for skinny matrices (m ≥n) full rank means all the columns are independent
- for fat matrices (m ≤n) full rank means all the rows are independent

Inverse

- A∈R^{n×n} is invertible or nonsingular if det A != 0
- columns of A are independent.
- rows of A are independent.
- columns/rows of A are basis of R^n

$$y = Ax$$
$$x = A^{-1}y$$

- y = Ax has unique solution x for any y (One-to-One 대응이 가능하다.)
- A has inverse

$$A^{-1}\ where\ AA^{-1} = A^{-1}A = I$$

- R(A) = R^n

Determinant

- Determinant is a function that maps a square matrix to a real number: det : R^{n×n} →R
- Definition: recursively defined using (n−1)×(n−1) determinant
- properties: (전부 column에 대한 것들이다.ㅈ)
  1. multilinear

$$
\begin{gathered}
det[a_1\ a_2\ ...\ v_1+v_2\ ...\ a_n] \\
= det[a_1\ ...\ v_1\ ...\ a_n] + det[a_1\ ...\ v_2\ ...\ a_n]
\end{gathered}
$$

  2. scaling

$$
det[a_1\ ...\ ca_i\ ...\ a_n] = c\, det[a_1\ ...\ a_i\ ...\ a_n]
$$

  3. exchange of columns

$$
det[a_1\ ...\ a_i\ a_j\ ...\ a_n] = -det[a_1\ ...\ a_j\ a_i\ ...\ a_n]
$$

- This implies that

$$
det(cA) = c^n det(A)
$$

- If any row or column of A is 0, then det(A) = 0

$$
\begin{gathered}
det[a_1\ a_2\ ...\ 0\ ...\ a_n] \\
= det[a_1\ a_2\ ...\ a_i - a_i\ ...\ a_n] \\
= det[a_1\ ...\ a_i\ ...\ a_n] + det[a_1\ ...\ -a_i\ ...\ a_n] \\
= det[a_1\ ...\ a_i\ ...\ a_n] - det[a_1\ ...\ a_i\ ...\ a_n] \\
= 0
\end{gathered}
$$

- If there exist linearly dependent rows/columns, det(A) = 0

  → dependent한게 하나라도 있으면 det(A) = 0

  → 동일한 column이 있으면 det(A) = 0

$$
\begin{gathered}
det([a_1\ a_1\ ...\ a_n]) = -det([a_1\ a_1\ ...\ a_n])\ (\because exchange\ of\ columns) \\
2\ det([a_1\ a_1\ ...\ a_n]) = 0
\end{gathered}
$$

$$
\begin{aligned}
&det([a_1\ a_2\ c_1a_1 + c_2a_2]) \\
&= det([a_1\ a_2\ c_1a_1]) + det([a_1\ a_2\ c_2a_2]) \\
&= c_1\ det([a_1\ a_2\ a_1]) + c_2\ det([a_1\ a_2\ a_2]) \\
&= 0
\end{aligned}
$$

- det(I) = 1

$$
\begin{gathered}
D = d_1e_1 + d_2e_2 + .. + d_ne_n \\
det(D) = d_1d_2...d_n\ det(e_1\ e_2\ ...\ e_n) \\
det(D) = d_1d_2...d_n
\end{gathered}
$$

- det(A) = det(A^T)
- determinant of triangular (either upper or lower) matrix A is the product of diagonal elements

$$
\prod_{i=1}^{n} a_{ii}
$$

$$
\begin{gathered}
det(\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & a_{22} & a_{23} \\ 0 & 0 & a_{33} \end{bmatrix}) \\
= det(\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & 0 & a_{23} \\ 0 & 0 & a_{33} \end{bmatrix}) + det(\begin{bmatrix} a_{11} & 0 & a_{13} \\ 0 & a_{22} & a_{23} \\ 0 & 0 & a_{33} \end{bmatrix}) \\
= 0 + det(\begin{bmatrix} a_{11} & 0 & a_{13} \\ 0 & a_{22} & a_{23} \\ 0 & 0 & a_{33} \end{bmatrix}) \\
= det(\begin{bmatrix} a_{11} & 0 & a_{13} \\ 0 & a_{22} & 0 \\ 0 & 0 & 0 \end{bmatrix}) + det(\begin{bmatrix} a_{11} & 0 & 0 \\ 0 & a_{22} & a_{23} \\ 0 & 0 & 0 \end{bmatrix}) + det(\begin{bmatrix} a_{11} & 0 & 0 \\ 0 & a_{22} & 0 \\ 0 & 0 & a_{33} \end{bmatrix}) \\
= 0 + 0 + det(\begin{bmatrix} a_{11} & 0 & 0 \\ 0 & a_{22} & 0 \\ 0 & 0 & a_{33} \end{bmatrix}) \\
= a_{11}a_{22}a_{33}
\end{gathered}
$$

- For square matrices A and B,

$$
det(AB) = det(A)det(B)
$$

so it follows that det(A^{-1}) = det(A)^{-1}

- can be shown using A = LUP factorization, and for any matrix B and triangular matrix T, we have det(AT) = det(A)det(T)

  → 이것을 확장하면 증명할 수 있다.