Lecture10. The Four Fundamental Subspaces

▼ **1. Four fundamental subspaces**

▼ **1) Column space and Null space**

- Column space는 행렬 A의 column vector들의 Linear combination으로 형성되는 공간이다. (Lecture5 참고)
- Null space는 A**x**=0를 만족시키는 해들의 Linear combination으로 형성되는 공간이다. (Lecture6 참고)

▼ **2) Row space**

- Row space는 행렬 A의 row vector들의 Linear combination으로 형성되는 공간이다.
- Row vector들은 row space를 span한다.
- 만약 행렬A의 row vector들이 Independent이면 row space의 basis가 되고, Dependent일 경우, basis가 아니다.
- Row space는 다음과 같이 표현할 수도 있다.

  $$Row\ space\ = All\ combinations\ of\ columns\ of\ A^T = C(A^T)$$

  - 행렬A를 transpose하고 column vector들의 Linear combination을 통해 column space를 정의하게 되면 원래 row space들에 대한 공간이 정의된다.

▼ **3) Left null space**

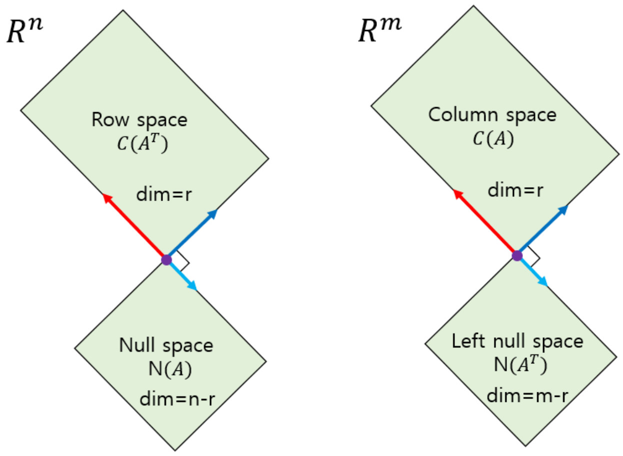
- Left null space는 행렬 A의 transpose에 대한 Null space를 의미한다.

  $$Left\ null\ space\ = null\ space\ of\ A^T\ = N(A^T)$$

- 4개의 주요 subspace들이 어떤 전체 공간에 존재하는지 정리해보면 다음과 같다.

  $$Ax = 0 \Rightarrow \underset{m \times n}{\begin{bmatrix} - & - & - & - \\ - & - & - & - \\ - & - & - & - \end{bmatrix}} \begin{bmatrix} - \\ - \\ - \\ - \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

  - Column space는 R^m의 공간에 존재한다.
  - Null space는 R^n의 공간에 존재한다.
  - Row space는 행렬A의 transpose 행렬에 대한 column space이므로 R^n의 공간에 존재한다.
  - Left null space는 행렬 A의 transpose 행렬에 대한 null space이므로 R^m의 공간에 존재한다.
- 위에서 나타낸 것을 그림으로 표현해보면 다음과 같다.



- Row space와 null space, column space와 left null space는 직교한다.
- Row space와 column space의 차원은 항상 dim=r로 같다. 즉, 두 공간의 rank가 같다는 뜻이다.

- 예시를 들어 이해해보자

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$$

행렬 A는 2개의 row와 3개의 column으로 구성되어 있다.

먼저 A의 row space를 보자. row space는 A의 row vector들의 선형조합으로 형성하는 공간이다. n=3이므로 3차원 공간상에 존재한다. A의 두 개의 row vector들이 3차원 공간에 아래와 같이 표현되었다고 가정해보자.

위 두 개의 row vector들은 선형 조합(Linear combination)을 통해 3차원 공간상에서 무한대로 2차원 평면을 채울 수 있지만 3차원 공간을 모두 채울 수 없다. 두 벡터를 직교하는 또 하나의 벡터가 존재하지 않기 때문이다. 이때 **두 row vector를 직교하여 3차원 공간을 형성할 수 있도록 하는 것이 바로 Null space다 즉 row space와 null space는 직교(orthogonal)한다**.

null space가 row space와 직교하는지 알아보기 위해 실제로 구해서 내적을 해보자.

$$Ax = 0 \Rightarrow \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$
$$\text{null space} \Rightarrow \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} -1 \\ 2 \\ -1 \end{bmatrix}$$

내적했을 때 둘 다 0이 되므로 null space vector는 row1, row2에 동시에 직교하는 벡터가 된다.

▼ 2. Basis and Dimension of 4 fundamental subspaces

- 각 subspace를 이해하기 위해서 basis와 차원을 알아야한다.

▼ 1) Column space

- Column space의 basis를 알기 위해서 공간을 형성하기 위한 최소한의 필수 벡터의 개수를 알아야한다. 이 개수가 차원이 되고 rank가 된다.
- Rank는 행렬 A의 소거 과정에서 드러나는 pivot들의 개수이고 pivot이 존재하는 column이 바로 pivot column이 된다. 이 pivot column들이 결국 A의 basis가 된다.

▼ 2) Null space

- Null space는 행렬 A의 free variable에 0과 1을 번갈아 설정하여 Ax=0의 해를 구한 special solution이 null space의 basis가 된다.

- 차원은 free column의 개수이므로 n-r이 된다.

▼ 3) Row space

- 행렬 A에서 소거를 통해 row reduction(소거를 통해 row를 줄여나가는 행위)을 통해 pivot row를 취하는 방법으로 row space의 basis를 구할 수 있다.
- 다음과 같이 행렬 A를 소거해서 reduced row echelon form을 만들어보자.

$$A = \begin{bmatrix} 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 1 \\ 1 & 2 & 3 & 1 \end{bmatrix} \Rightarrow \begin{bmatrix} 1 & 2 & 3 & 1 \\ 0 & -1 & -1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \Rightarrow \begin{bmatrix} 1 & 2 & 3 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \Rightarrow \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} = R$$

  - 행렬A와 행렬 R의 column space가 같지 않다는 것을 알 수 있다. $C(A) \neq C(R)$

- 행렬A를 소거할때 한 연산은 row reduction이다. Row에 대해 상수를 곱하고 row끼리 Linear combination 연산을 했기때문에 row space는 변하지 않는다. 그렇기 때문에 소거된 행렬 R의 row들은 row space상에 존재한다. 하지만 column space는 원시 행렬A에 row 연산을 통해 변하게 된다.

> ⭐ 행렬 A를 소거하여 R을 만들었을때 **A와 R의 row space는 같지만 column space는 다르다.**

- **행렬 A의 row space의 basis는 R행렬의 처음 r개의 row 벡터들이다.** Pivot의 개수인 rank만큼의 기저 벡터가 존재하며, pivot이 위치한 R의 row벡터들이 행렬 A의 기저가 되는 것이다.

  위와 같이 R의 첫 번째와 두 번째 row vector들은 독립(independence)이며 A의 subspace인 2차원 평면을 "span"한다. 따라서 R의 row1과 row2가 행렬 A의 기저(basis)가 된다.

  R의 row1, row2의 선형 조합으로 A의 row space를 만들어낼 수 있다는 것을 어떻게 알 수 있을까? 이를 확인하기 위해선 바로 A->R을 만드는 순서를 거꾸로 수행해보면 된다.

▼ 4) Left null space

- 아래와 같이 A^T에 대한 null space를 구하면 Left null space를 구할 수 있다.

$$A^T x = 0 \Rightarrow \begin{bmatrix} 1 & 1 & 1 \\ 2 & 1 & 2 \\ 3 & 2 & 3 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \Rightarrow x_1 \begin{bmatrix} 1 \\ 2 \\ 3 \\ 1 \end{bmatrix} + x_2 \begin{bmatrix} 1 \\ 1 \\ 2 \\ 1 \end{bmatrix} + x_3 \begin{bmatrix} 1 \\ 2 \\ 3 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

- Left null space의 차원은 행렬 A의 row의 수에서 rank를 뺀 m-r이다. 이 예시에서는 3-2=1이다. 다음과 같이 A^T 소거를 통해 column3이 free column인 것을 알 수 있다. x3에 임의의 값을 설정하여 나머지를 풀면 Left null space를 구할 수 있다.

$$A^T = \begin{bmatrix} 1 & 1 & 1 \\ 2 & 1 & 2 \\ 3 & 2 & 3 \\ 1 & 1 & 1 \end{bmatrix} \Rightarrow \begin{bmatrix} 1 & 1 & 1 \\ 0 & -1 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \Rightarrow \begin{bmatrix} 1 & 0 & 1 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

- Gauss-Jordan소거법을 이용하여 A를 R로 만드는 과정을 써보면 다음과 같다.

$$rref \begin{bmatrix} A_{m \times n} & I_{m \times m} \end{bmatrix} \rightarrow \begin{bmatrix} R_{m \times n} & E_{m \times m} \end{bmatrix}$$

  - 행렬 A의 뒤에 단위 행렬을 붙여서 augmented matirx를 만들고 reduced row echelon form을 만들어준다. 이때 A가 R이 되고 I가 E가 된다. 이 과정을 하나의 행렬을 곱하여 rref를 만드는 과정으로 다시 써보면 다음과 같다.

$$E \begin{bmatrix} A_{m \times n} & I_{m \times m} \end{bmatrix} \rightarrow \begin{bmatrix} R_{m \times n} & E_{m \times m} \end{bmatrix}$$
$$EA = R$$
$$EI = E$$

- 위의 식에서 E행렬이 rref를 만드는 역할을 한다.

- 행렬 A를 이 과정을 통해 R로 만들어보면 다음과 같다.

$$rref\begin{bmatrix}A_{m\times n} & I_{m\times m}\end{bmatrix} \rightarrow \begin{bmatrix}R_{m\times n} & E_{m\times m}\end{bmatrix}$$

$$\left[\begin{array}{cccc|ccc} 1 & 2 & 3 & 1 & 1 & 0 & 0 \\ 1 & 1 & 2 & 1 & 0 & 1 & 0 \\ 1 & 2 & 3 & 1 & 0 & 0 & 1 \end{array}\right]$$

$$\left[\begin{array}{cccc|ccc} \color{blue}{1} & 2 & 3 & 1 & 1 & 0 & 0 \\ 0 & -1 & -1 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 1 \end{array}\right]$$

$$\left[\begin{array}{cccc|ccc} \color{blue}{1} & 2 & 3 & 1 & 1 & 0 & 0 \\ 0 & \color{blue}{1} & 1 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 1 \end{array}\right]$$

$$\left[\begin{array}{cccc|ccc} \color{blue}{1} & 0 & 1 & 1 & -1 & 2 & 0 \\ 0 & \color{blue}{1} & 1 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 1 \end{array}\right] = RE$$

  - 이를 통해 A->R이 되고, I->E가 되었다. E행렬이 A를 R로 만들어주는 행렬인지 다음과 같이 확인해볼 수 있다.

$$Left\ null\ space \rightarrow \begin{bmatrix} -1 & 2 & 0 \\ 1 & -1 & 0 \\ \color{green}{-1} & \color{green}{0} & \color{green}{1} \end{bmatrix} \begin{bmatrix} 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 1 \\ 1 & 2 & 3 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$Left\ null\ space \rightarrow -1\begin{bmatrix}1 & 2 & 3 & 1\end{bmatrix} + 0\begin{bmatrix}1 & 1 & 2 & 1\end{bmatrix} + 1\begin{bmatrix}1 & 2 & 3 & 1\end{bmatrix} = \begin{bmatrix}0 & 0 & 0 & 0\end{bmatrix}$$

  - **Left null space는 A의 row vector의 Linear combination으로 영벡터를 만들수 있어야한다.**
  - R행렬의 row3가 영벡터이고 영벡터가 존재한다는 것은 A벡터의 Linear combination을 통해 영벡터를 만들 수 있다는 의미이다. 따라서 1차원의 null space가 존재한다.

C(A): Column space	N(A): Null space	C(A^T): Row space	N(A^T): Left Null space
basis: pivot columns of A	basis: special solutions of $Ax = 0$	basis: pivot columns of A^T	basis: special solutions of $A^Tx = 0$
dimension: r	dimension: n-r	dimension: r	dimension: m-r