**영어음성학 12주차 정리**

* Data들은 반드시 vector의 형태로 존재해야 하며, 인공지능이라면 입력 vector가 어떤 형태이든 간에 들어갔을 때 결과값이 vector로 나와야 한다.
* 선형대수(Linear Algebra): 행렬을 이용한 것.
* 행렬에서 함수를 통과하는 방식이 곱셈인 것. 모든 데이터가 vector로 이루어져야 하는 이유가 행렬 곱셈을 해야 하기 때문이다.
* 행렬의 크기를 이야기할 때는 dimension(차원)을 이용한다. dimension만 같으면 곱셈이 가능하다.
* m\*n 행렬: m개의 행, n개의 열을 가진 행렬
* 세로로 길게 생긴 행렬은 column vector라고 부른다.
* Vector space: 무수하게 많은 여러 벡터들이 모여 만든 공간. 모든 가능한 덮여진 공간을 의미하며, 일부분은 vector space가 될 수 없다.
* Linear combinations: 행렬을 곱하고 더하는 그 자체를 의미한다. Linear combination을 무한히 하면, 결국 2차원 space를 가득 채우게 되며, 원래 있던 space를 절대 벗어나지 않는다.
* column vector에 의한 vector space는, 2 \* 2 행렬을 좌표 상에 표시했을 때, 아무리 linear combination을 하고 spanning으로 확장시켜봤자 line 선상에만 값이 나오지 나머지 공간은 채울 수가 없다.
* row vector에 의한 vector space도 존재하는데, whole space가 몇 차원인지는 몇 개의 row가 있는지 알아야 한다.
* column space는 그 중에서 몇 개의 independent한 것이 있는지도 알아야 한다.
* column vector의 관점에서 whole space와 column space가 있고, 똑같이 row vector의 관점에서 whole space와 column space가 존재한다. 두 개의 whole space와 column space가 존재하는 것.
* null space
* 나머지 비어있는 공간 (기하학적 정의)
* 어떤 행렬이 있을 때 무엇을 곱하든 null이 되는 것 (수학적 정의)
* 비어있는 공간이지만 null space 자체의 존재는 중요하다.
* Ax = b 에서, x는 입력 vector, b는 출력 vector, A는 x를 b로 바꿔주는 transformation matrix이다.
* 입력vector와 출력vector의 길이가 같을 필요는 없다. 출력 차원은 행렬의 행에 따라 달라진다. (여기서 행렬은 인공지능, 기계, 출력함수 등으로 볼 수 있으며, linear combination을 통해 transform하기 때문에 말 그대로 linear transformation matrix라고 부른다.)
* Detransformation: 역함수(inverse) 개념 (역행렬)
* transform을 했는데 다시 역함수를 취했을 때 원래 자리로 돌아갈 수 없는 경우가 존재한다. 이 경우는 dependent하게 합쳐져버려 원래 성질을 완전히 잊어버려서 원래 자리로 돌아갈 수 없게 된다.
* eigenvector는 transformation한 결과값이 원점에 놓이는 것이다. 어떤 행렬의 기하학적인 위치(좌표)를 원점과 쭉 연결했을 때 일직선 상에 있으면 그것은 eigenvector라고 할 수 없다.