**영어음성학 14주차 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
* null space의 수학적 정의는 Ax = 0 또는 Ax = [ 0 ] (행렬형태)

0

이 조건을 만족시키는 가능한 모든 x가 원점을 지나는 일직선 상에 있다. (그래프를 그리면 원점을 지나는 일직선으로 표현가능)

* null space는 값이 없어서 ouput에 직접적으로 영향을 주는 것처럼 보이지는 않지만 null space가 있어서 어떤 결과값이 더 풍성해진다.
* 기하학적으로 null space와 평행하는 값이 있을 때, 원래 이미지가 강아지였다면 그대로 형태만 조금 바뀐 강아지로 출력된다(똑같이 강아지의 범주에 속하는데 다른 종류의 강아지가 출력되는 것)
* null space와 평행하지 않게 간다면 강아지의 형태가 사라진다고 볼 수 있다.
* 따라서 vector는 방향이라고 좀 더 물리적으로 생각할 수 있다.
* Eigenvector의 개념에 대해서 생각할 때, 주어진 행렬 A(Given A)가 있을 때 Eigenvector가 뭐냐고 물어보는 것이 맞다. Given A가 있을 때 모든 vector가 원점과 입출력이 평행하는 순간이 존재한다.
* 각각의 Eigenvector에 대해 해당하는 Eigenvalue가 있다. (eigen은 한국어로는 ‘고유’라는 의미이다.
* Eigenvector의 정의는, transformation된 결과인 Av가 transformation되지 않은 그냥 v의 상수 배밖에 되지 않는다는 뜻이다. 기하학적으로 한 선에 있다는 뜻히다. 그 상수 배가 얼만큼인지, 그 ratio가 eigenvalue이다.
* 통계에서 상관관계의 의미는, 그래프를 봤을 때로 생각해보면 상대적으로 시각적으로 ‘같이가는 느낌’이 있는 그래프가 있는데, 이를 상관관계가 있다고 표현한다.
* -1 < r < 1 (r = correlation, 상관관계) 인데, 여기서 상관관계가 가장 낮은 지점은 0이다. 음의 상관관계도 상관관계이기 때문에, 얼마나 선에 가까운지가 r의 절댓값이라고 보면 된다.
* 여기서 선형대수가 쓰일 수 있는데, cosθ = r이다(그 값이 매우 유사하다). 아무리 몇백 차원으로 큰 차원의 행렬이더라도 결국 vector 두 개와 원점을 연결하면 세 점이 이어진 삼각형이 된다. 이 각도값에 cos을 붙인 것이 r값이다.
* 기하학적으로 inner product 값을 구하는 방법은, 한 벡터에서 다른 한 벡터를 원점과 연결한 점에 수직으로 내린 후 그걸 기준으로 원점에서의 두 길이를 곱하면 된다. 따라서 |a| \* cosθ \* |b|
* 이 inner product는 스펙트로그램을 직접만드는 데 사용된다.
* correlation이 많으면 inner product가 높게 나타나고, 이런 성질을 이용해서 스펙트로그램을 나타낼 수 있다.
* 두 개의 vector a, b를 좌표상에 표기했을 때 그 길이가 같다면 dot product의 크기는 angle이 결정한다. 이 angle이 90도에 가까워질수록 dot product는 0에 수렴하며 최소가된다. (90도의 dot product가 0이므로)
* 두 vector가 얼마나 가까운지를 angle로 표현하는 것이고, inner product를 할 때는 당연히 두 개의 signal이 같아야 한다.
* 아무리 복잡한 signal이어도 sin파들의 합인데, 이는 푸리에 법칙으로 설명가능하다.
* e^θi : complex number 🡪 이 것을 가지고 만든 파형이 complex wave 이다.
* 기존에 있는 실수 vector wave와 complex number의 절댓값과 inner product가 가능하다. 기하학적으로는 x축에 실수를(R), y축에 허수를(I) 표기하여 좌표상에 표시할 수 있다.
* amplitude(amp)에 허수는 절대 들어있을 수 없다. amp는 abs(absolute, 절대화)되었기 때문이다.
* 실험결과 그래프에서 bar의 개수는 sample의 수와 같다. 이 그래프에서 5000까지만 표현가능한데, 이를 나이퀴스트주파수라고 한다. 따라서 그래프에서 반정도만 의미가 있다.
* dot product는 a · b로 표현하고, cosθ = r (통계학적으로 말하는 correlation)이다.
* θ = 90도 일 때, cos(θ) = 0
* 두 개의 벡터를 좌표상에 표시했을 때 일직선 상에 놓여있는 경우 r = 1 이다. (단, 기울기는 양수)

이 일직선이 기울기가 음수일 경우 r = -1이다.

* sin그래프 파형과 cos그래프 파형은 각 wave의 진폭과는 상관없이 그 위상차이가 180도인데, 이를 위상차이라고 하며 영어로는 phase라고 칭한다.
* cosθ 를 기계학습, 인공지능에서는 cosine similarity라고 한다. a와 b가 얼마나 비슷한가를 수치적으로 나타낸 것이다. 다시 말하지만, 통계쪽으로는 이 값이 correlation을 나타내는 r값과 매우 유사하다.