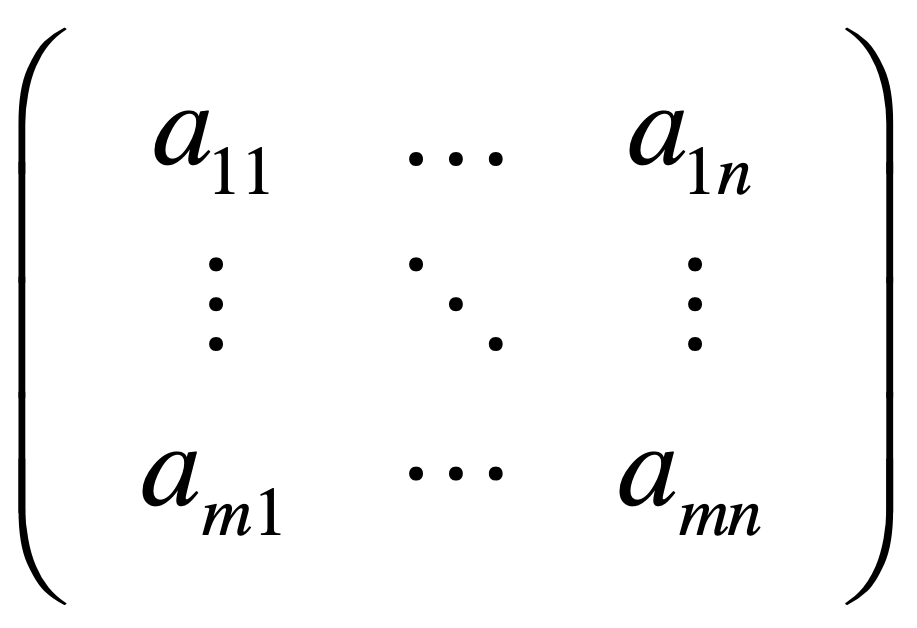
<Linear Algebra>

1. Matrics

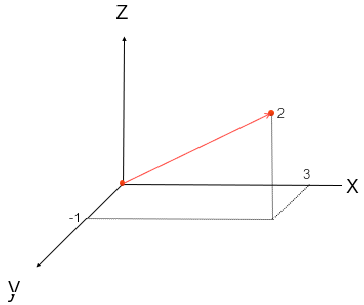
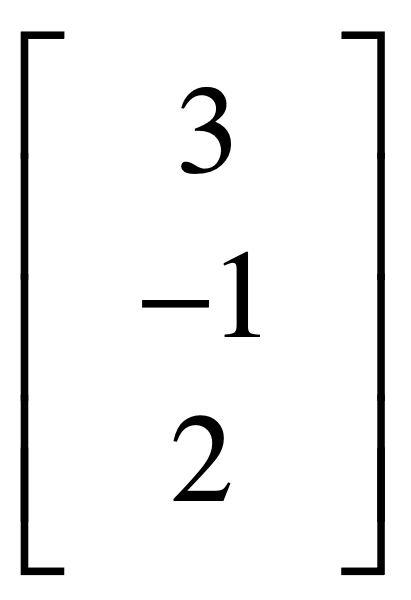


MxN행렬

1. Vectors

일종의 행렬

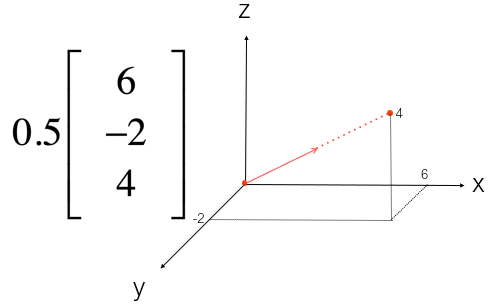
* Column vector(기하적으로)



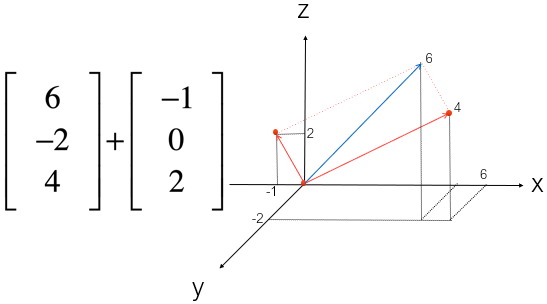
3차원에서 표현할 수 있는 점의 개수 하나

벡터 =하나의 점, 100차원이라 해도 점의 개수는 하나

* Vector Multiplication

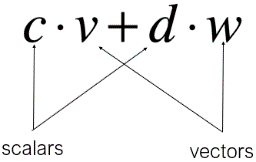
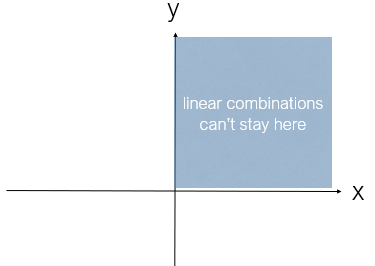


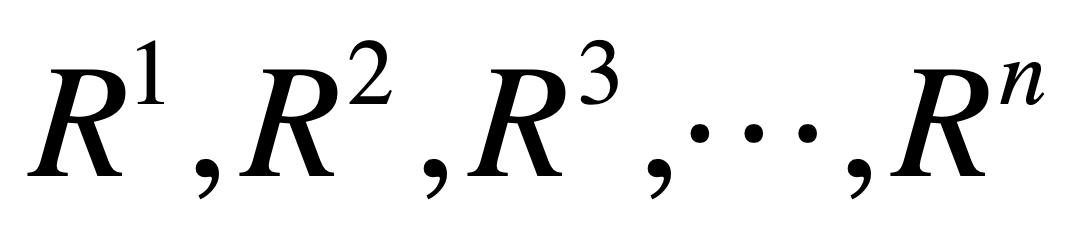
* Vector Addition



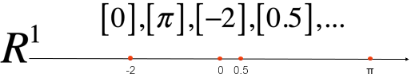
* Vector Spaces

Linear Combination로 cover가능한 모든 공간

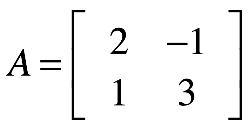
*  Real number

N차원의 공간은 n개의 구성요소로 이루오진 모든 벡터를 포함한다.

1차원의 벡터 space

* Column space

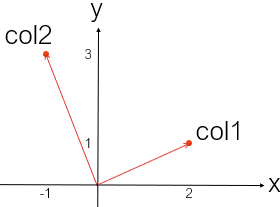
Columns of Amxn in Rm, all their linear combinations form a subspace: column space, C(A)

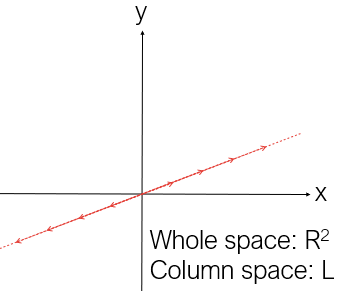
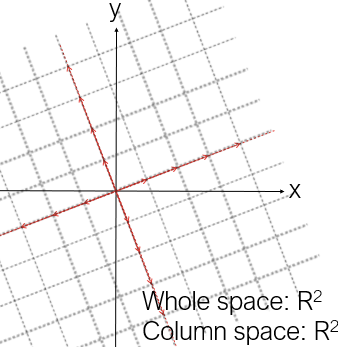
Column space 2개

2개를 2차원에 찍으면 점 2개

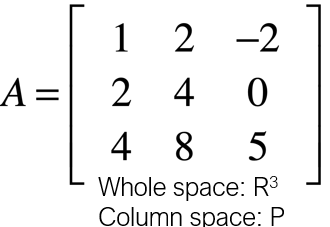
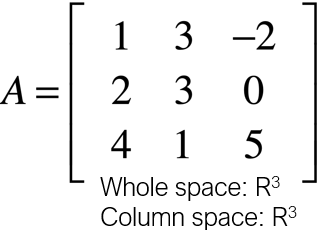
2개의 점을 가지고 linear combination

무한번 반복하면 하나의 plain이 된다.

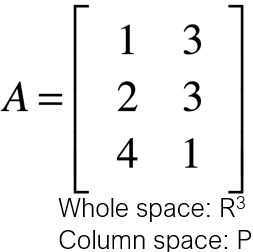
이 두개는 independent



Whole space의 dimention은 몇 개의 row가 있는지 찾는 것



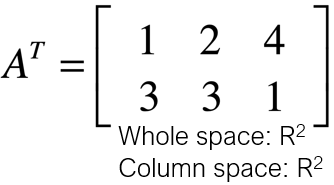
Plain 2차원

Column vecter 2개

점이 3차원에서 찍힘

Independent한 점이 2개이므로

Column space가 Plain이다.

Column vector 3개

점이 2차원에서 찍힘

2차원에서는 column vector가 100개 있어도 최대 2차원

A에서 column space 2차원 T(A)도 2차원

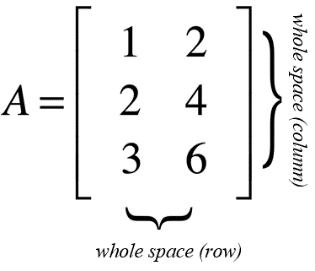
단 whole space는 달라진다

<Four spaces in a matirx>

mxn matrix has two whole spaces: Rm and Rn

Column vector 관점에서 whole space

Row vector 관점에서 ‘whole space

Column space

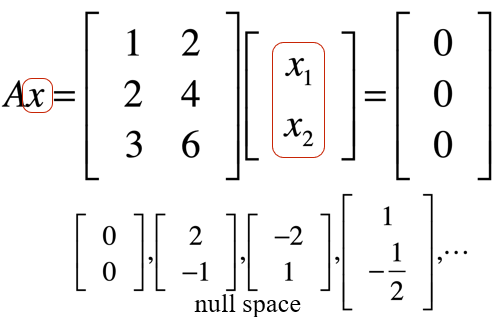
- dependent> 1차원

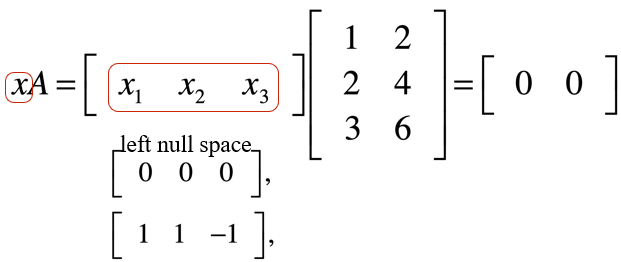
Whole space-column space

= 3 – 1 = 2(null space)

<Null Space>

Null space 의미 : 출력에 영향을 미치지 않는 공간들





한 평면에서 그 평면과 orthagonal하게 변형을 시키면 출력값에 영향을 주지 않는다.

실용적 해석 응용적 해석 수학적 해석

NULL space를 따라서 움직인다. = 강아지 사진 여러 종류의 변화

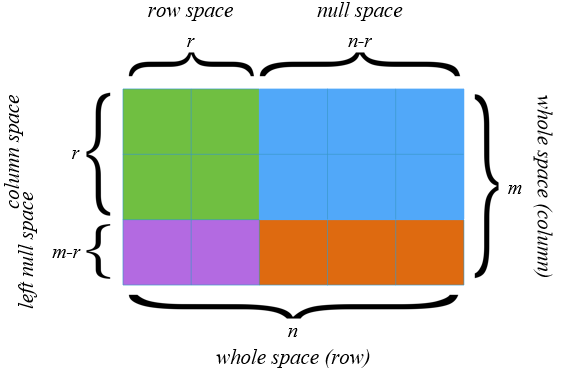
1. 강아지 그림들이 ''많이 변해도'' 같은 값(이 그림은 강아지 그림입니다)을 출력

2.강아지 라는 키워드를 입력 -> 각각 다른 이미지의 강아지이더라도 결국에는 '강아지'사진을 출력 (강아지 사진 여러 종류의 변화)

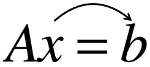
사회적 의미 :

목적은 사냥이다. 중간에 방해물이 있을 때 그 요인을 피해서 간다 하더라도 목적 달성은 동일하다.

NULL space를 확장해가는 과정이라고 할 수 있다.



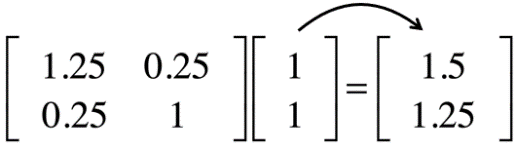
<Linear Transformation>

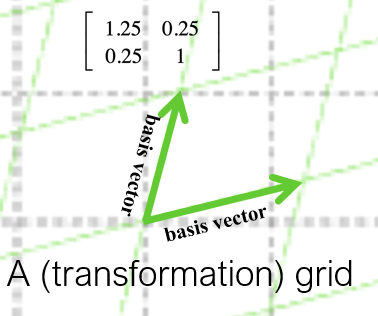
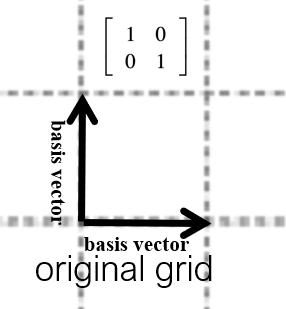
 행렬은 대문자, 소문자는 벡터

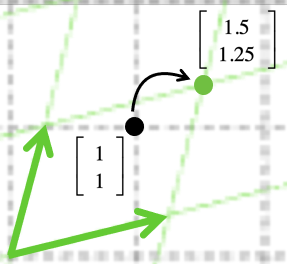
X : 입력 벡터, B : 출력벡터, A: transformation matrix

1. place *x* on original grid
2. transform *x* onto A grid
3. read transformed *x* on original grid (=b)

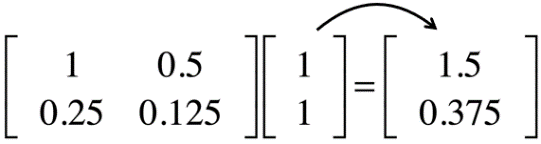
Ex.1

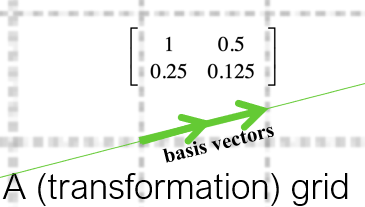
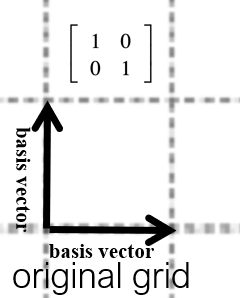


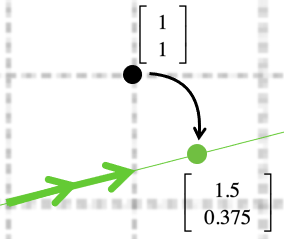




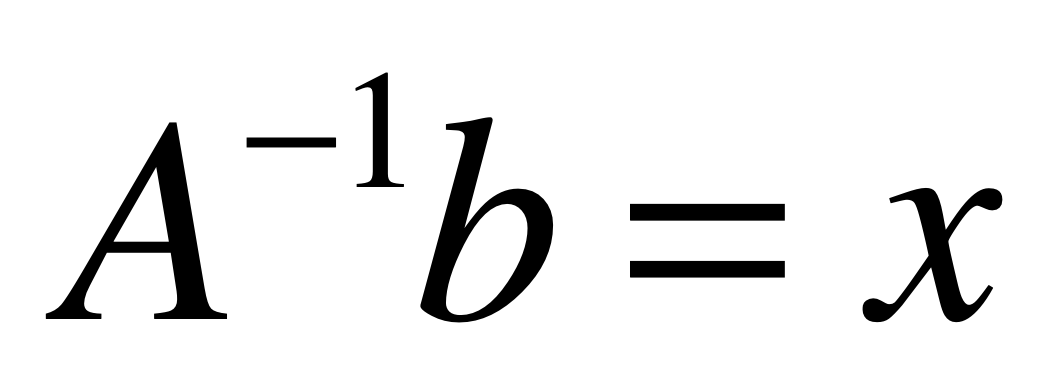
Ex.2



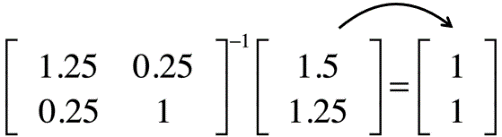


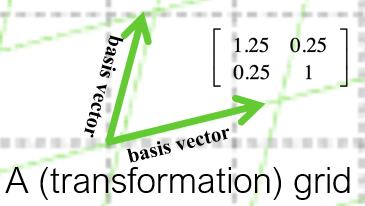


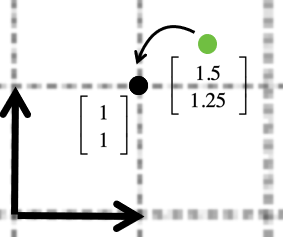
<Detransformation : Inverse matrix>



Ex.1





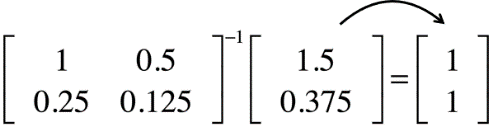


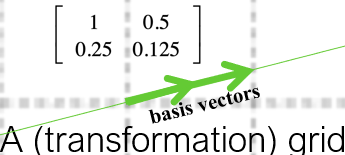
1.5 1.25 평행사변형 면적 = det의 면적

면적이 0이라는 것= 같은 선상에 열벡터 2개

>> 역함수를 구할 수 없음

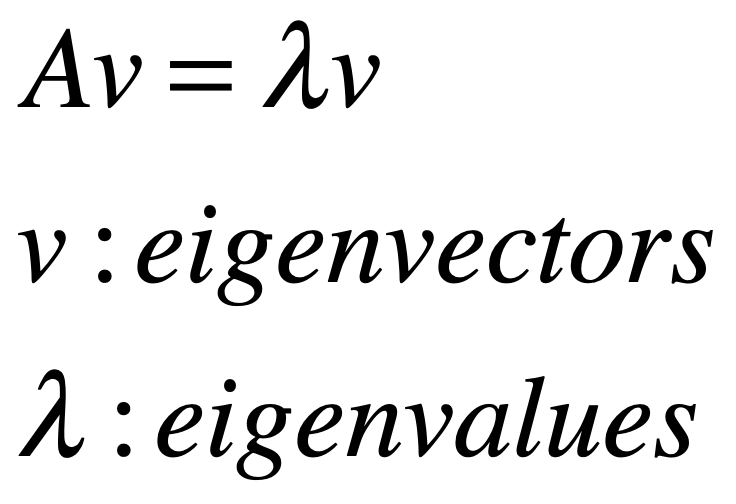
Ex.2





Dependent하면 그 전 점을 찾을 수 없으므로 not invertible

<Eigenvector>



Among all v, some v is parallel to Av. that v is eigenvector

given 행렬이 있을 때 eigen vector는 방향 전체를 나타내게 된다. = eigen vector space라고 한다.

* 2x2에서는 eigen vector 가 2개

3x3에서는 '' 3개

* Eigen value

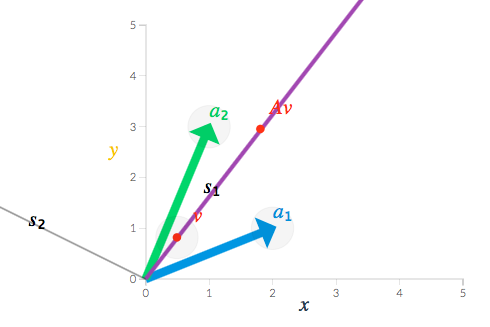
각각의 eigenvector에 따라서 eigenvalue 가 존재

v와 Av 간의 ratio

eigen 고유벡터 unique와 비슷한 느낌

서로 영향을 미치지 않는 여러 분야로 분리하고 싶을 때

ex. 국어 영어 수학 점수 분리하여 언어능력, 수리적 능력 을 분석하기



**191029**

* 숫자들의 나열 : 벡터

Ex. 흑백 그림

검정 : 0 , 흰색 : 10, 회색 : 그 사이 숫자들

* 숫자들의 행태나 특징 : 직사각형 = 행렬

행렬을 한 줄 씩 분리해서 늘어놓으면 벡터화

~ 소리도 늘어놓으면 벡터화된다

~ 텍스트 ex. 1000000000000

1번째 2번째 …5000 번째 자리에 1넣기

**191031**

X = np.array([[1,2,3],[4,5,6]])

X.shape

X.ndim

X.dtype : data type 을 알고 싶을 떄

X.astype : 안에 들어있는 data type을 바꾸고 싶을 때

np.zeros\_like(X) : 안에를 다 0으로 바꿔줌

X\*0 을 해도 같은 값이 나옴

data.shape

bins : 바구니 개수로 이해하면 편함.

\*who : 지금 available 한 변수들을 모두 보여주는 것

191119

데이터 기계 데이터

(함수)

벡터 벡터

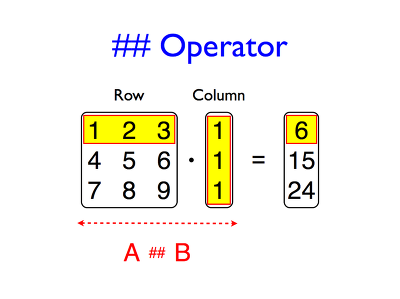
텍스트 음성

------------------------------------

데이터 형태 : 벡터

그 중간에 있는 것이 행렬, 행렬이 벡터를 다른 벡터로 변환

<행렬의 곱>



모든 데이터가 벡터 형태를 띄어야 하는 이유는 행렬 곱셈이 가능하게 하기 위해서

191126

<Correlation>

좌표에서 두개의 선 사이 각도 세타

cos thetha 는 R의 값과 같음

cos0 = 1 : 직선이 일치한다는 뜻 R=1

cos90= 0 : 직선이 직교한다는 뜻 R=0

<Inner Product> = dot product

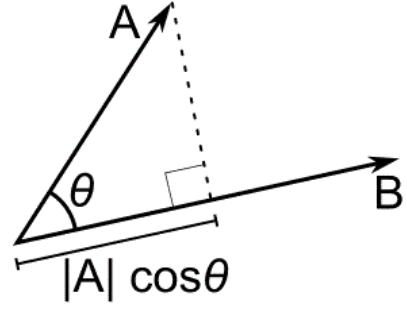
필요한 이유 :

2000Hz가 이 소리에 얼마나 들었는지 등을 파악할 수 있는 방법

여러개의 sin wave를 만들어서 inner product를 한다.

의미 잘 알아두기

* 기하학적 해석



B라는 기준 벡터에 A라는 벡터가 얼마나 B벡터의 성분이 들어있나?

cos simililarity =

cos90 = 0 (직교) , cos0 = 1(일치)

**Q.두개의 벡터 주어졌을 때 cos similarity 구하는 문제**

**Q. 두 직선 간 앵글이 작을수록 inner product 값은?**

**A. 길이(|a|,|b|)가 고정되어 있을 때 세타의 값이 0일 때 최대이고 90일 때 최소가 된다**

<Wave>

Wave도 시간 순으로 볼 때 벡터

Ex. 30개의 숫자가 있을 때

* Wave 속에 어떤 sin성분이 있는지 알기

= spectrogram - 한 시점에서의 analysis를 하는 것

= 어떤 frequency가 많이 들어있는지 아는 것

**<Fourier Transfrom>**

임의의 입력 신호를 다양한 주파수를 갖는 주기함수들의 합으로 분해하여 표현

<주의할 것>

1. phasor에는 두가지 중료가 있음

sin cos 같이 단순한 phasor

같이 complex한 phasor

단순한 phasor는 만약 시작점이 동일하다면 똑같이 나오지만 90도만 이동해도 완전 불일치 하게 보이는 문제가 발생한다.

Target wave에 대해서 여러가지 wave들을 만들어서 probe를 한다.

근데 Target 이 얼마나 shift하느냐에 따라서 값이 sensitive하게 변한다.

따라서

1. complex phasor를 사용 ex)

complex phasor로 만든 값들은 허수의 값도 포함

vector이긴 하므로 product할 수는 있지만 inner product값이 complex number가 나온다.

**★ complex number 가 inner product의 결과로 만들어진 후에**

**real value로 했을 때는 결과값이 그냥 real value가 나오므로 plotting이 가능하다.**

**그러나 complex number를 사용했을 경우에는 바로 plotting이 불가능하다.**

**따라서 절댓값을 씌운다.**

**complex number 절댓값하는 법**

**1. 실수값과 허수축의 (a,b)를 찍는다. 이 때 절댓값은 원점에서 (a,b)까지의 거리**

**원점에서 (a,b)까지의 거리 = |a + bi| 결과 실수값**

1. inner product 할 때 두개의 dimention 이 동일해야 한다. ex. 1xn nx2

**191205**

1. dot product

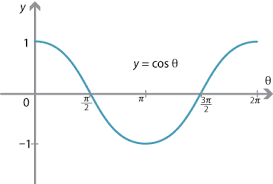
같은 사이즈의 벡터가 있을 때

[1xn] [nx1] = 1

1. 각도 thetha

cos(theta)과 correlation과 같음 확인하기

cos(theta) 그래프



theta 가 pi/2 , 즉90도 orthagonal일 때 0이다.

★★★이미지화 해서 알아두기★★★

* 2차원에서 데이터가 완전히 하나의 직선을 이룬다.

= 두개의 벡터 사이의 각도가 0이 된다.

그럼 correlation R값이 1이다.

* 2차원에서 데이터가 음의 방향으로 하나의 직선을 이룬다

correlation R값이 -1

* 3차원에서 두개의 벡터가 180도를 이룰 때

cos180 = -1이므로 R값이 -1

3. 두개의 주파수 관계에서 R값은

r값은 진폭과 관련이 있다