2021-1-AI기초수학-OT
담당교수 : 장 영 호

강의계획서
평가계획서

수업 목표	본 강의는 인공지능(Artificial intelligence)을 이해하는데 필수적인 수학 내용인 선형대수학, 다변수 미적분학, 기초 통계와 확률, 특이값 분해(SVD), 주성분 분석(PCA) 및 그래디언트 알고리즘을 포함한 내용을 고교, 대학 수학 과정의 수준으로 설명하고, 이어서 배운 개념들이 실제로 인공지능을 개발할 때 어떻게 쓰이는지, 잘 알려진 구체적인 예와 알고리즘을 이용하여 최대한 쉽게 설명하는 것을 목표로 하고 있다.
Office Hour	카카오톡 단체 대화방 혹은 이메일 219514@inha.ac.kr로 상담 환영

평가 항목	횟수	평가 비율	비고
출석		10%	우리대학 출석 관련 규정 및 지침에 따른 평가
과제		20%	미 제출 0점, 불성실 과제 5점, 성실 과제 10점 (누적점수를 20점으로 환산)
중간고사	1	35%	7주차까지의 세부 교육목표(주제)를 평가한다.
기말고사	1	35%	14주차까지의 세부 교육목표(주제)를 평가한다.

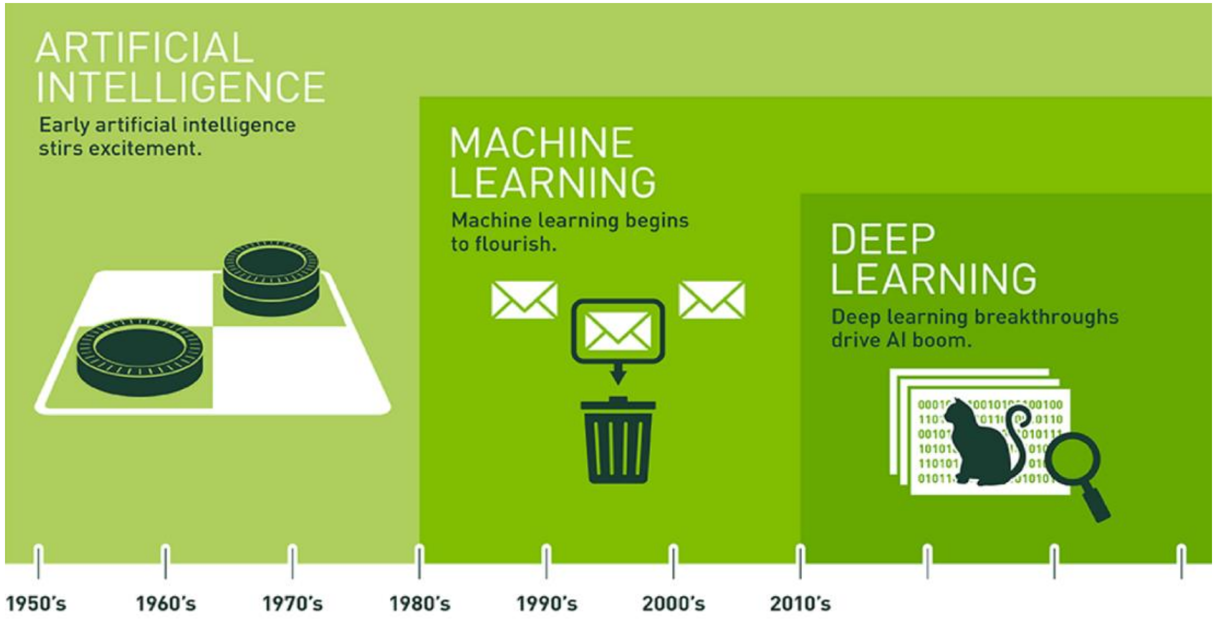
강의계획서 : 내용은 변경될 수도 있습니다.

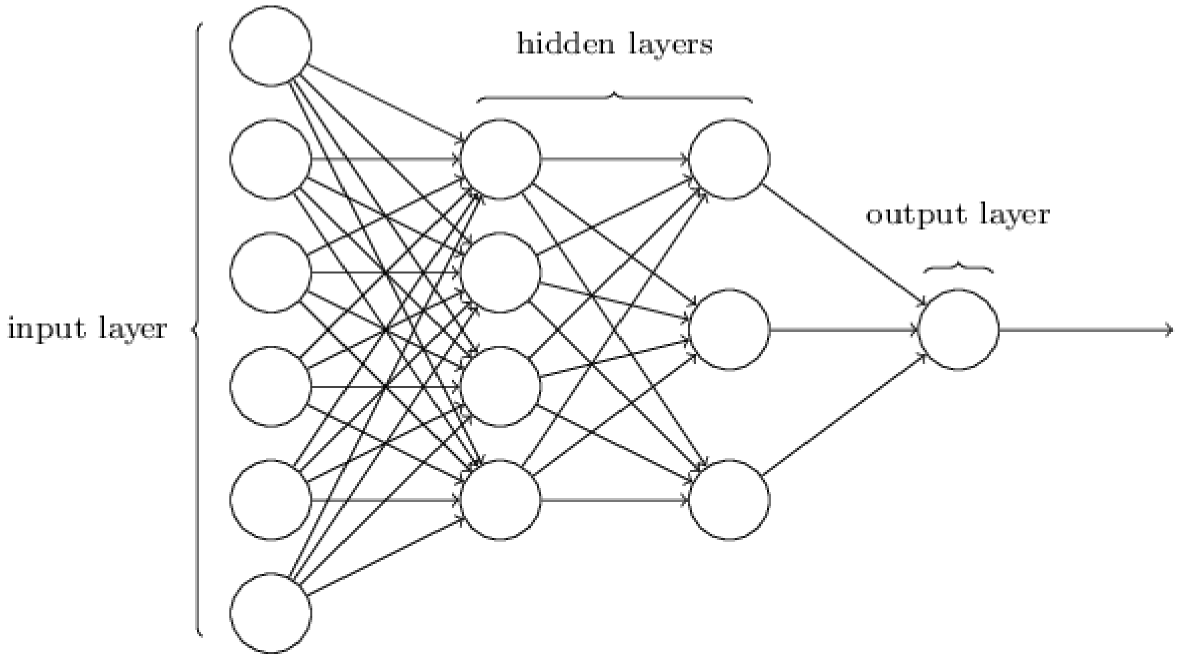
<table>
<tr><th colspan="2">수업내용</th><th>핵심 개념</th></tr>
<tr><td>3월</td><td>SageMath(Python/Sage/R) 언어 사용법

코딩 연습

Part I. 행렬과 데이터분석 (선형대수학)</td><td>1. 벡터
2. 선형 연립방정식
3. 행렬과 행렬식
4. 일차변환과 기저 및 차원
5. 선형변환
6. 고유값, 고유벡터, 대각화
7. SVD (특이값 분해)
8. 이차형식(Quadratic form)</td></tr>
<tr><td>4월</td><td>Part II. 다변수 미적분과 최적화</td><td>1. 일변수함수와 미적분
2. 다변수함수와 미적분</td></tr>
<tr><td>5월</td><td>Part III. 확률통계와 빅데이터</td><td>1. 순열, 조합, 확률
2. 확률변수
3. 확률분포
4. 데이터 활용의 실제</td></tr>
<tr><td>6월</td><td>Part IV. 빅데이터와 인공지능</td><td>1. 주성분분석
2. 신경망
3. 프로젝트
인공지능 알고리즘에 응용하는 수학, 선형회귀, 자연어 처리, 이미지 인식</td></tr>
</table>

Artificial Intelligence, Machine Learning, Deep Learning : What's the difference?





1. 인공지능 > 머신러닝 > 딥러닝

2. 인공지능을 구현하는 방법은 크게 규칙기반 시스템(Rule-based system)과 머신러닝으로 나눌 수 있다. 즉, 머신러닝은 인공지능을 구현하는 한 방법이다.

3. 머신러닝은 사람이 일일히 규칙을 정의하지 않아도 데이터를 통해 배우는 것이 핵심이다. 즉,

Knowledge from experience

로 인공지능을 구현한다.

4. 딥러닝은 생물체의 뇌 구조에서 영감을 얻은 머신러닝 기법 중 하나로 핵심 원리는

다계층 구조(MLP, Multi-layer perceptron)

를 이용한 Representation Learning을 통해 스스로 데이터속에서 유용한 feature를 찾아내는 것이다. 딥러닝은 기존 머신러닝과는 다르게

인간 고유의 영역, 이미지 분석, 언어 인식

과 같은 직관적이고 고차원적인 사고를 요하는 분야에 강점이 있다.

출처 - Deep learning, Youshua Bengio et al. (2016)

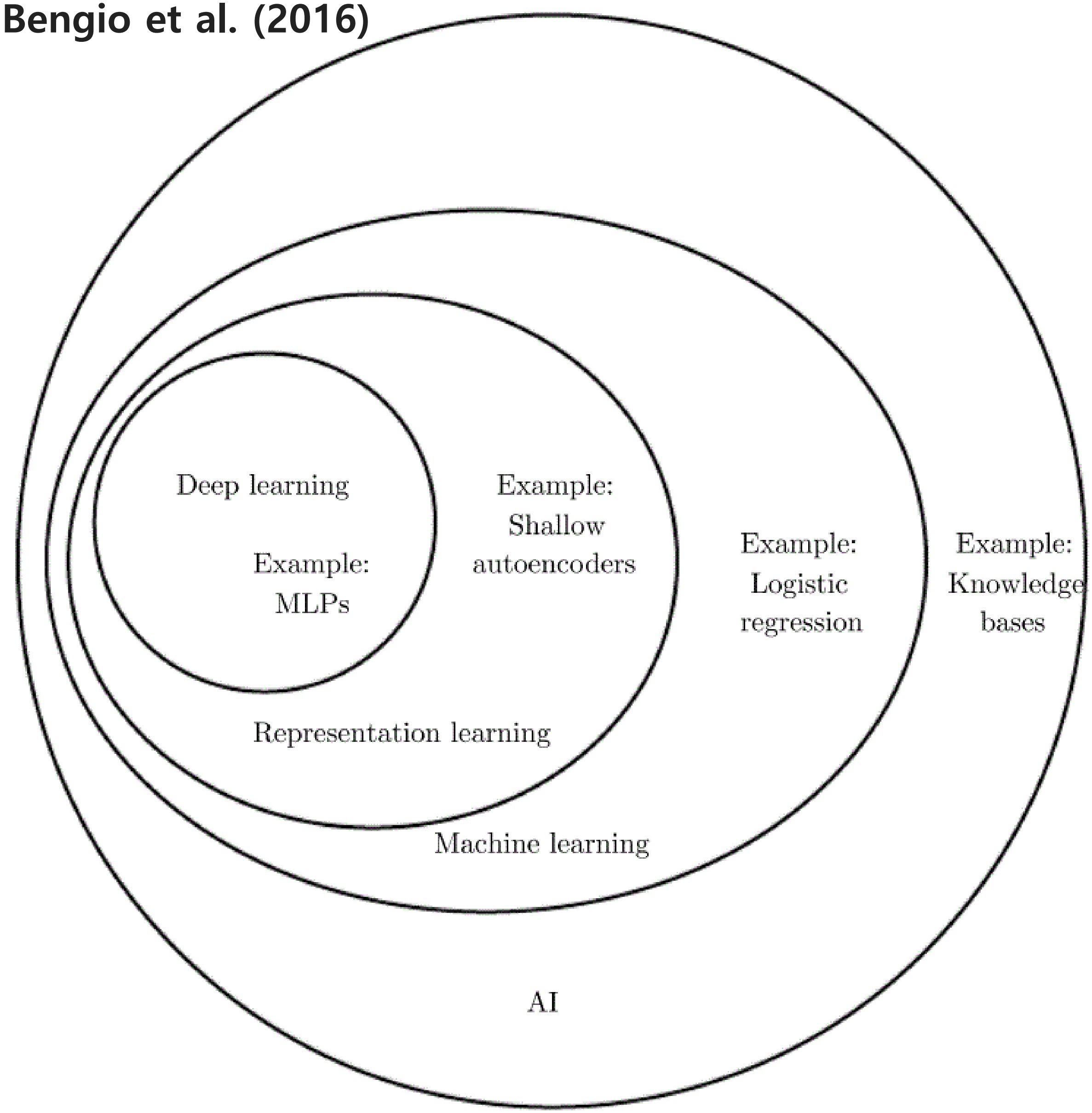


Figure 1.4: A Venn diagram showing how deep learning is a kind of representation learning, which is in turn a kind of machine learning, which is used for many but not all approaches to AI. Each section of the Venn diagram includes an example of an AI technology.

Sage Tutorial 2005년에 처음 발표된 Sage는 **서버용 무료 공개 수학연산 프로그램**으로서, 사용자의 PC에 직접 설치할 필요가 없이 인터넷 환경만 제공되면 언제 어디서나 사용이 가능하다. 특히 Sage를 이용하여 워크시트(worksheet)를 제작하고, 그 워크시트에서 Sage 명령과 수학연산에 적합한 파이썬(python) 명령어들을 직접 실행해볼 수 있다.

Sage는 기본적으로 웹을 통해 서비스되기 때문에, Sage를 이용하여 제작된 워크시트 또한 웹을 통해 다른 사람들과 공유할 수 있다. 따라서 새로운 수학연산을 시도할 때 그 수학연산에 대한 Sage 명령어의 지식이 부족하다 하더라도, 웹을 통해 이미 공개된 다양한 자료들을 통해 Sage는 그 해법을 제시할 수 있다.

Sage의 수많은 명령어 및 새롭게 개발된 함수들은 구글(Google)과 같은 웹 검색 사이트를 이용하여 관련된 검색이 가능하다. 그 명령어들을 단순히 복사하여 붙여 넣는 방식만으로도 새로운 수학연산을 할 수 있다. 또한 이와 같이 다양하게 공개된 워크시트의 자료들을 아래의 주소에서 한 번에 얻을 수도 있다.

위치	서버명	주소
성균관 대학교	Math1	http://math1.skku.ac.kr/pub
	Math3	http://math3.skku.ac.kr/pub
	Sagenb	http://sagenb.skku.edu/pub
	SKKU Sage Cell 1	http://sage.skku.edu
	SKKU Sage Cell 2	http://sage1.skku.edu

위치	서버명	주소
한국방송통신 대학교	수학 사이버랩 연산서버	http://sage.knou.ac.kr/pub
SAGE	Alpha	http://alpha.sagenb.org/pub

온라인으로 사용하기 위해서는
https://cloud.sagemath.com
접속하여 사용한다(크롬사용).

Sage의 대부분의 명령어 알고리즘들은 새롭게 개발된 것들이 아니라 아래 표와 같이 이미 지난 40여 년 동안 수학의 각 분야에서 유용하다고 검증된 오픈소스 프로그램들의 함수들이 파이선 형식의 명령어로 다시 종합되었다.

분야	오픈소스 수학연산 프로그램
Algebra and calculus	Maxima, SymPy
High precision arithmetic	GMP, MPFR, MPFI, quaddouble, Givaro
Commutative algebra	Singular
Number theory	PARI, NTL, mwrank, ECM, FLINTQS, GMP-ECM
Exact linear algebra	LinBox, IML
Group theory	GAP
Scientific computation	GSL, SciPy, NumPy, cvxopt
Statistical computation	R
Graphics (2d and 3d)	Matplotlib, Tachyon3d, Jmol

따라서, 우리는 이와 같이 강력한 수학연산을 제공하는 Sage를 이용하여 제작된 공학적 도구를 사용하여 대수적인 답을 찾을 수 있고 Sage의 상호작용 함수(interact function)를 활용하여 다양한 시뮬레이션을 실행할 수도 있다. 이렇듯 Sage는 한국형 CAS(Computer Algebra System) 도구로써 공학적 도구 도입의 단점을 모두 극복하여 새로운 도구의 대안이 될 수 있다. 게다가 이미 Sage의 한글화에 성공하여 언어의 제약도 따르지 않는다. 또한, 새롭게 Sage 명령어를 익히지 않고서도 기존의 CAS 도구(Mathematica, Matlab, Maple 등)들의 언어를 그대로 Sage에서 사용할 수도 있다.

Basic Sage Codes

1) 정의

- ● 변수 x var('x')
- ● 벡터 $<1,2,3>$ vector([1, 2, 3])
- ● 벡터함수 $(2-5t,\ 4t,\ 1+3t)$ vector([2-5*t, 4*t, 1+3*t])
- ● 행렬 matrix([[1, 2], [3, 4]])
- ● 값 정의(예: a=3) a=3
- ● 함수의 정의 f(x)=x^2

2) 벡터와 행렬

- ● 벡터의 크기 a.norm()
- ● 벡터의 내적 b.dot_product(a)
- ● 벡터의 외적 b.cross_product(a)
- ◆ 행렬식 det(A)
- ◆ 행렬의 크기 len(A)
- ◆ 행렬의 고유값 A.eigenvalues()
- ◆ 행렬의 고유벡터 A.eigenvectors_left()

3) 미분과 적분

- ● 도함수 diff(f(x), x)
- ◆ 부정적분 integral(f(x), x)
- ◆ $[a,\ b]$ 에서 적분 integral(f(x), x, a, b)
- ◆ 이중적분 integral(integral(f(x), (x, 0, 1)), (y, 0, 1))
- ◆ 삼중적분 integral(integral(integral(f(x), (x, 0, 1)), (y, 0, 1)), (z, 0, 1))

4) 그래프 그리기

- ● 일반함수(2D) plot(f(x), (x, -4, 4))
- ● 매개변수함수(2D) parametric_plot((f(x), g(x)), (x, 0, 2*pi))
- ● 음함수(2D) implicit_plot(f(x)==0, (x, -2, 2), (y, -2, 2))
- ● 선분(2D) line([(1, 1), (2, 2)], color='red')
- ● 일반함수(3D) plot3d(f(x, y), (x, -2, 2), (y, -2, 2))
- ● 매개변수함수(3D) parametric_plot3d((f(x), g(x), h(x)), (x, 0, 2*pi))
- ● 음함수(3D) implicit_plot3d(f(x, y, z)==5, (x, -3,3), (y,-3,3), (z,-3,3))
- ● 선분(3D) line([(1, 1, 1), (2, 2, 2)], color='red')
- ● Contour Plot contour_plot(f(x, y), (x, -1, 1), (y, -1, 1), cmap='hsv', labels=True)
- ● 벡터필드(2D) plot_vector_field((x+y, x), (x, -3, 3), (y, -3, 3))
- ● 벡터필드(3D) plot_vector_field3d((0, 0, 1), (x, -3, 3), (y,-3,3), (z,-3,3))

5) 극한

- a에서 극한 limit(f(x), x=a)
- a에서 우극한 limit(f(x), x=a, dir='+')
- a에서 좌극한 limit(f(x), x=a, dir='-')
- $+\infty$ 에서의 극한 limit(f(x), x=+oo)
- $-\infty$ 에서의 극한 limit(f(x), x=-oo)

6) 기타

- 방정식 풀이 solve(f(x)==0,x)
- 급수($\sum_{n=1}^{\infty} p(n)$) sum(p(n),n,1,+∞)
- 변수 a의 값 보이기 print a
- 정의된 함수 f 보이기 show(f)
- 조건문 if 조건:

  statement(s) (조건이 True이면 실행할 명령어)

  else:

  statement(s) (조건이 False이면 실행할 명령어)
- 반복문 for i in range(10):

  print i (i=0-9 까지 i를 출력)

SageMath 기본 사용법

https://sites.google.com/site/dkjangsage/home/sage3

SageMath 연습 1:

(1) 집합 $\{3x - 2 \mid x \in (0, 1, \ldots, 9) \text{ and } x \text{ is odd}\}$을 구하여라.

(2) 집합 $\{3x^2 + 1 \mid x \in (0, 1, \ldots, 9) \text{ and } x \text{ is even}\}$을 구하여라.

(3) $\sum_{k=1}^{10} (2k)(2k+1)$의 값을 구하여라. **(for loop 이용)**

(4) $f(x) = x^4 + x^2$일 때, $f(3)$을 구하여라.

(5) 구간 $(0, 5)$에서 함수 $f(x) = -x^2 + 4x + 1$의 그래프를 그리고, 그래프가 x축과 만드는 영역을 표현하고, 그 영역의 넓이를 구하여라.

(6) 구간 $(0, 5)$에서 함수 $f(x) = -x^2 + 4x + 1$의 그래프를 x축에 대하여 회전시킨 회전체를 그리고, 그 회전체의 부피를 구하여라.

(7) 임의로 tree(graph)를 하나 만들어라.

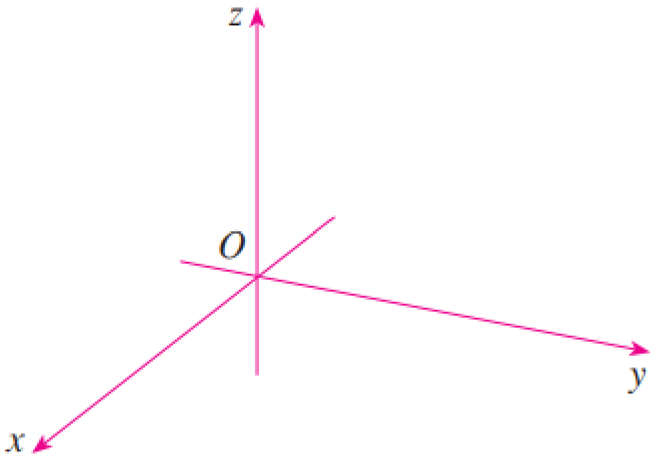
10분 휴식

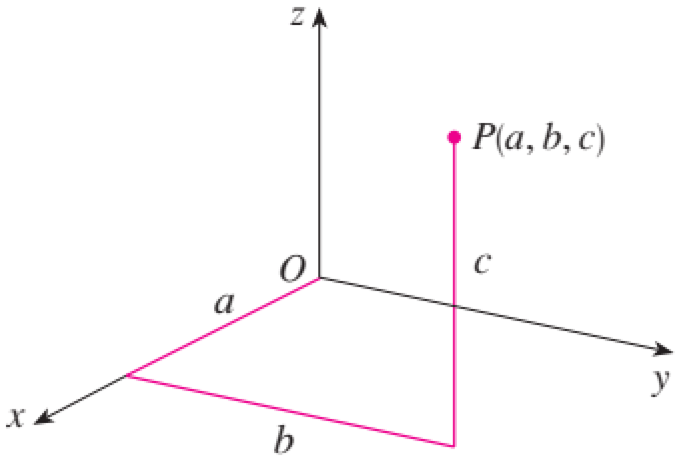
Part 1. 행렬과 데이터 분석

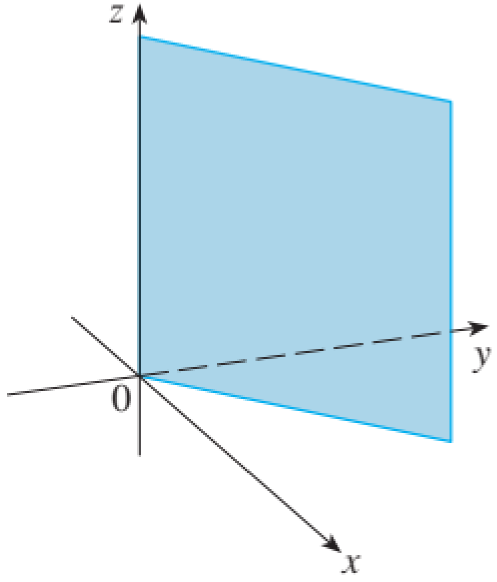
1. 벡터
2. 선형 연립방정식
3. 행렬과 행렬식
4. 일차변환과 기저 및 차원
5. 선형변환
6. 고유값, 고유벡터, 대각화
7. SVD(특이값 분해)
8. dlckgudtlr(Quadratic form)

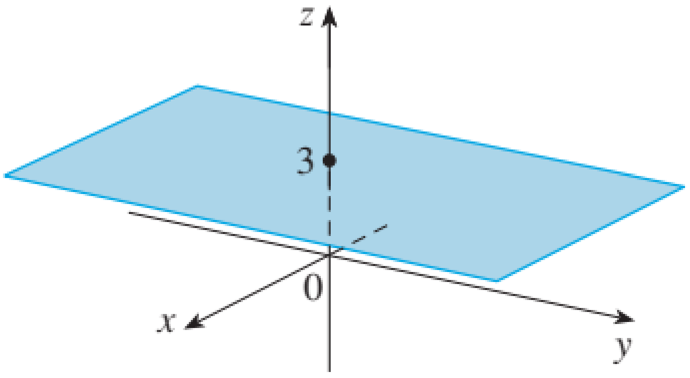
Goal: 선형대수학 개념의 이해와 공학적 도구(CAS)를 활용한 문제해결력을 기르고 그 과정을 과제로 제출하고, 설명할 수 있는 능력을 키우고, 그 향상된 능력에 대하여 평가한다.

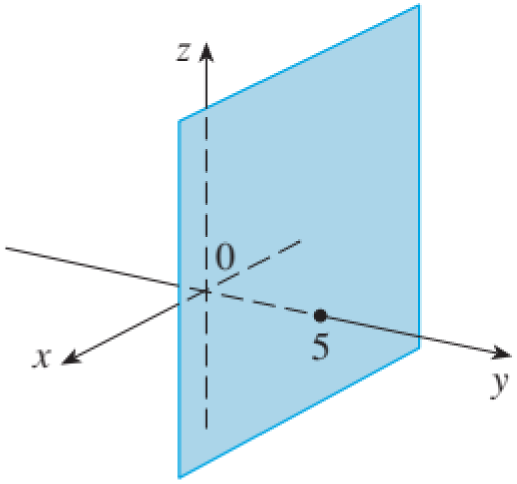
선형대수학은 인공지능뿐만 아니라 자연과학과 공학, 사회과학의 여러 분야에서 활용되고 있다. 1890년에 혜성 궤도를 관측할 때부터 쓰이기 시작했고, 현대에는 빅데이터 분석에도 쓰이고 있다. 또, 우리가 많이 쓰는 구글 검색 엔진속에도 선형대수학이 이용된다. 전 세계 1위 브랜드 가치를 가진 구글 신화를 창조할 수 있었던 이유는 선형대수학을 이용한 검색엔진 덕분이었던 것이다.

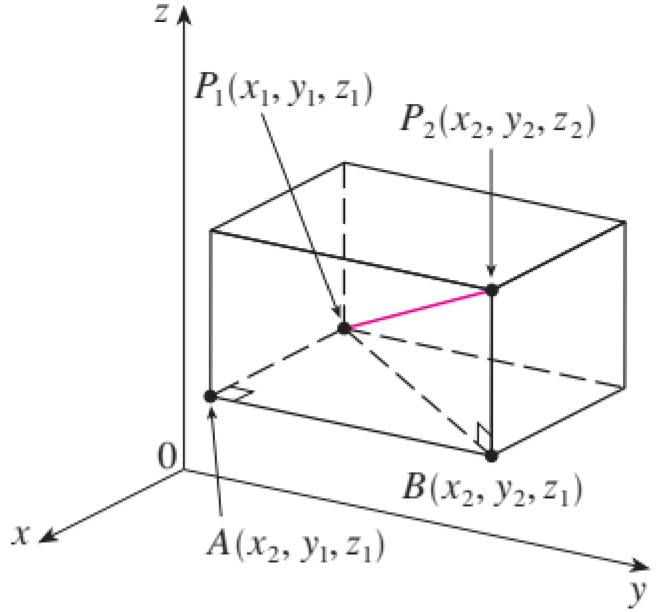








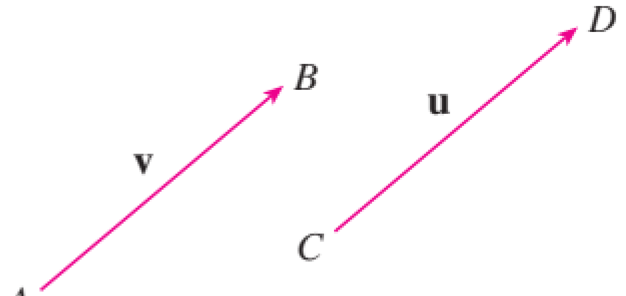




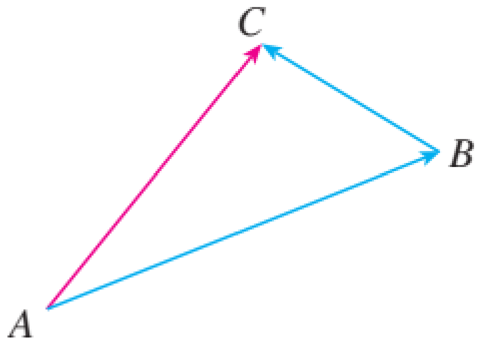
$$|P_1P_2| = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}$$

스칼라(scalar): 길이, 넓이, 질량, 온도 등과 같이 크기만 주어진 물리량

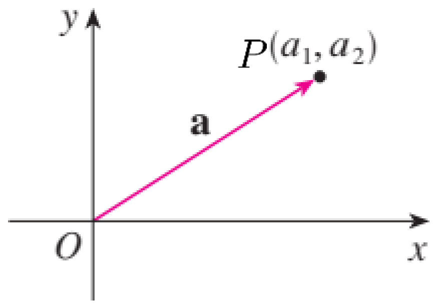
벡터(vector): 속도, 위치이동, 힘 등과 같이 크기와 방향이 함께 주어진 물리량



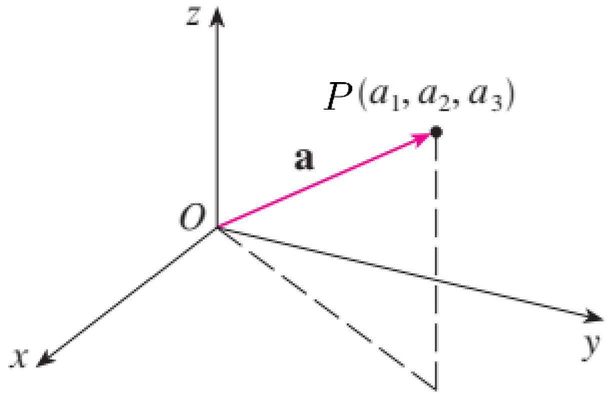
$\mathbf{u} = \mathbf{v}$.



$\overrightarrow{AC} = \overrightarrow{AB} + \overrightarrow{BC}$

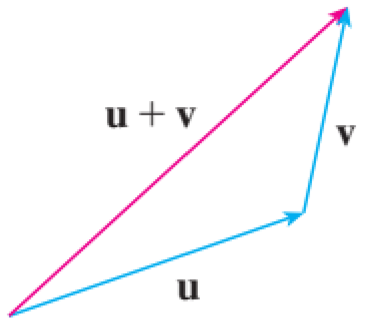


$\mathbf{a} = \langle a_1, a_2 \rangle$

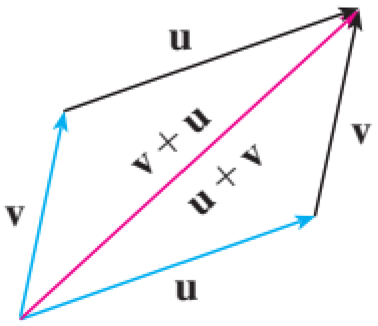


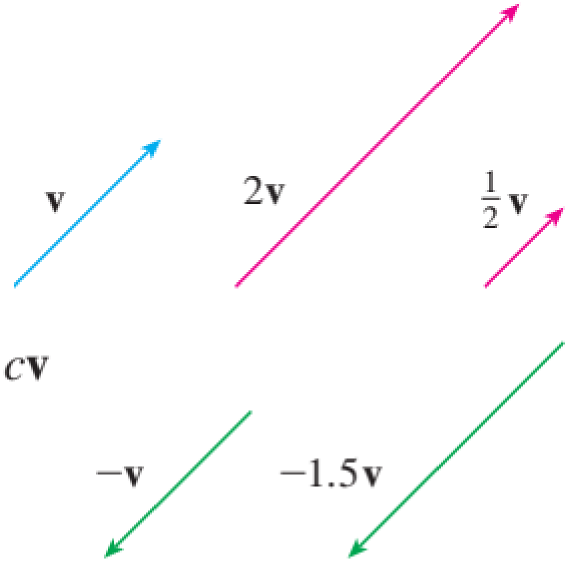
$\mathbf{a} = \langle a_1, a_2, a_3 \rangle$

position vector of the point P.



sum u + v





scalar multiple $c\mathbf{v}$

$\mathbf{a} = (a_1, a_2, a_3)$ $\quad \mathbf{a} = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}$

norm, length:

$|a| = \sqrt{a_1^2 + a_2^2 + a_3^2}$

If $\mathbf{a} = \langle a_1, a_2 \rangle$ and $\mathbf{b} = \langle b_1, b_2 \rangle$, then

$$\mathbf{a} + \mathbf{b} = \langle a_1 + b_1, a_2 + b_2 \rangle \qquad \mathbf{a} - \mathbf{b} = \langle a_1 - b_1, a_2 - b_2 \rangle$$

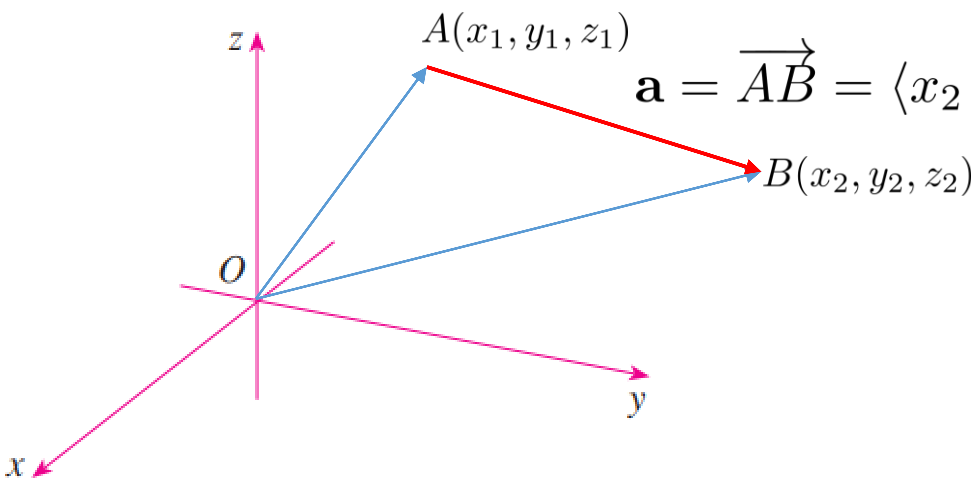
$$c\mathbf{a} = \langle ca_1, ca_2 \rangle$$

Similarly, for three-dimensional vectors,

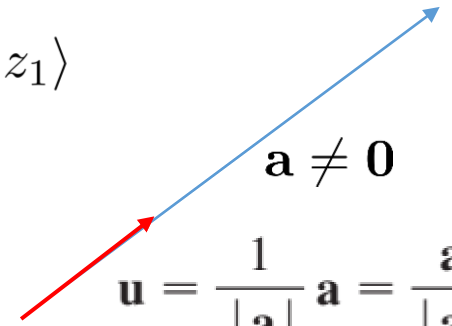
$$\langle a_1, a_2, a_3 \rangle + \langle b_1, b_2, b_3 \rangle = \langle a_1 + b_1, a_2 + b_2, a_3 + b_3 \rangle$$

$$\langle a_1, a_2, a_3 \rangle - \langle b_1, b_2, b_3 \rangle = \langle a_1 - b_1, a_2 - b_2, a_3 - b_3 \rangle$$

$$c\langle a_1, a_2, a_3 \rangle = \langle ca_1, ca_2, ca_3 \rangle$$



$$\mathbf{a} = \overrightarrow{AB} = \langle x_2 - x_1, y_2 - y_1, z_2 - z_1 \rangle$$



$$\mathbf{a} \neq \mathbf{0}$$

$$\mathbf{u} = \frac{1}{|\mathbf{a}|}\mathbf{a} = \frac{\mathbf{a}}{|\mathbf{a}|}$$

벡터 **a** 방향으로의 **unit vector**

$\mathbb{R}^n$의 벡터 $\mathbf{x},\ \mathbf{y},\ \mathbf{z}$와 스칼라 $h,\ k$에 대하여 다음이 성립한다.

(1) $\mathbf{x}+\mathbf{y}=\mathbf{y}+\mathbf{x}$
(2) $(\mathbf{x}+\mathbf{y})+\mathbf{z}=\mathbf{x}+(\mathbf{y}+\mathbf{z})$
(3) $\mathbf{x}+\mathbf{0}=\mathbf{x}=\mathbf{0}+\mathbf{x}$
(4) $\mathbf{x}+(-\mathbf{x})=\mathbf{0}=(-\mathbf{x})+\mathbf{x}$
(5) $k(\mathbf{x}+\mathbf{y})=k\mathbf{x}+k\mathbf{y}$
(6) $(h+k)\mathbf{x}=h\mathbf{x}+k\mathbf{x}$
(7) $(hk)\mathbf{x}=h(k\mathbf{x})$
(8) $1\mathbf{x}=\mathbf{x}$

$0\mathbf{x}=\mathbf{0}$
$k\mathbf{0}=\mathbf{0}$
$(-1)\mathbf{x}=-\mathbf{x}$

집합에 두 연산 **덧셈(addition)**과 **스칼라 곱(scalar multiplication)**에 대해 위와 같은 8가지 성질을 만족하는 집합을 벡터공간(vector space)이라 한다.

$\mathbb{R}^n$을 **n-차원 유클리드 벡터공간(Euclid n-dimensional vector space)**이라 한다.

$\mathbf{v}_1,\ \mathbf{v}_2, \dots, \mathbf{v}_n \in \mathbb{R}^n$이고 $c_1, c_2, \dots, c_n \in \mathbb{R}$일 때,

$$c_1\mathbf{v}_1 + c_2\mathbf{v}_2 + \cdots + c_n\mathbf{v}_n = \mathbf{0}$$

인 형태를 $\mathbf{v}_1,\ \mathbf{v}_2, \dots, \mathbf{v}_n$의 **일차결합**이라 한다.
(linear combination)

[Example 1] SageMath를 사용하여 답하세요.

(1) $\mathbb{R}^2$의 점 $P_1(0,-4),\ P_2(-3,1),\ Q(2,3),\ Q_1(2,-1),\ Q_2(-1,4)$에 대하여 $\overrightarrow{OQ},\ \overrightarrow{P_1Q_1},\ \overrightarrow{P_2Q_2}$를 성분으로 표시하여라.

(2) $\mathbb{R}^3$의 벡터 $\mathbf{x}=(1,2,3),\ \mathbf{y}=(-2,4,-3)$에 대하여 $\mathbf{x}+\mathbf{y},\ 3\mathbf{x}-4\mathbf{y}$를 구하여라.

(3) 벡터 $\mathbf{a}=(2,-1,-2)$ 방향으로의 단위벡터를 구하여라.

(4) 벡터 $\mathbf{x}=(2,-1,-2,5),\ \mathbf{y}=(-2,1,-4,7),\ \mathbf{z}=(5,3,-4,-5)$일 때, 일차결합 $3x-5y+10z$을 구하여라.

(5) 두 벡터 $\mathbf{x}=(2,-1,-2,5),\ \mathbf{y}=(-2,1,-4,7)$ 사이의 거리를 구하여라.