

2021년
빅데이터 분석기반 비즈니스 개발자
양성 & 취업연계 과정

[선형대수학]

2021. 05. 28
김슬기

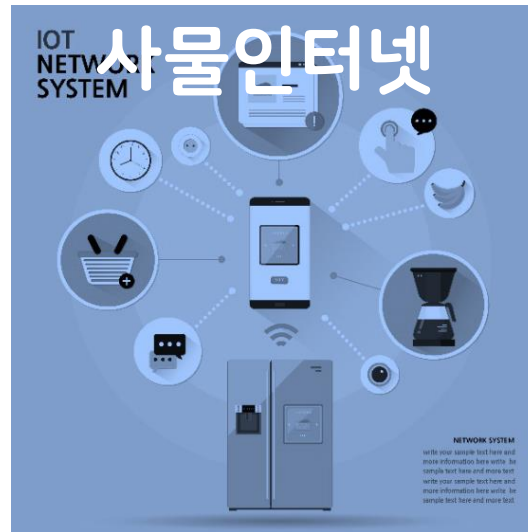


4차 산업혁명 시대의 사회 변화 및 특징

"Change your thoughts
and you change your world"

—Norman Vincent Peale—

핵심 키워드





머신 러닝이란?

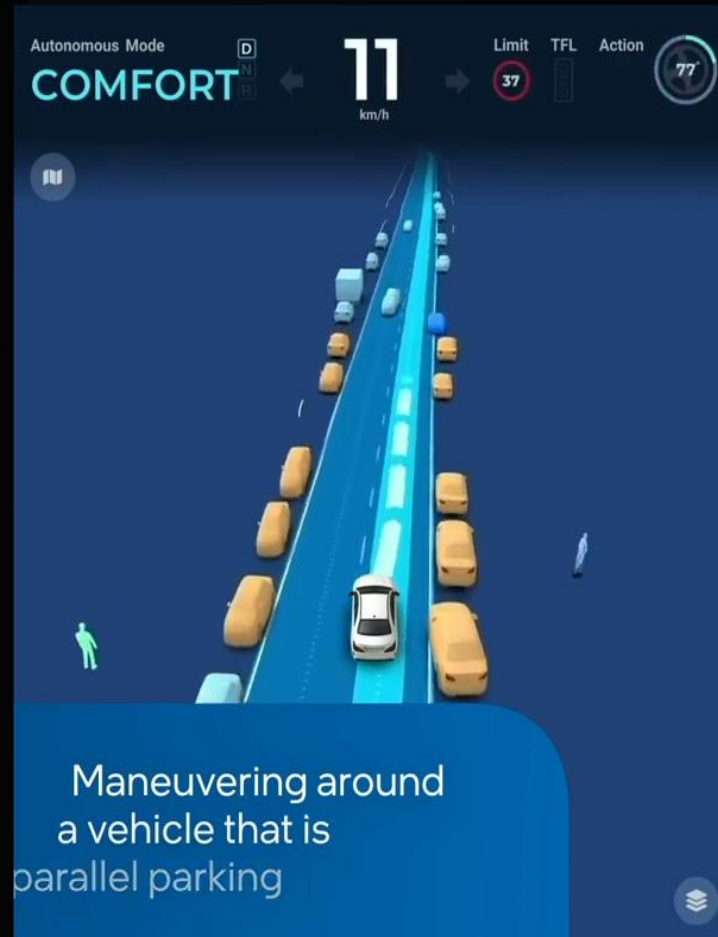
"Change your thoughts
and you change your world"

—Norman Vincent Peale—

Machine Learning

“기계 학습”: 기계가 학습을 통해 발전





2021 CES - 인텔 자회사 모빌아이, 레벨 4 수준 자율주행 테스트 영상

Program



Machine Learning Program



머신 러닝

컴퓨터 과학

+

수학

(선형대수, 미분, 통계, 확률)


```
from sklearn.linear_model import LinearRegression
```

```
:
```

```
model = LinearRegression()
```

```
model.fit(X_train, Y_train)
```

코드 몇 줄이면 할 수 있음!

```
from sklearn.linear_model import LogisticRegression
```

```
:
```

```
model = LogisticRegression()
```

```
model.fit(X_train, Y_train)
```

수학

선형대수, 미분, 통계와 확률

선형 대수학

많은 데이터 묶을 수 있음

$$A = \begin{bmatrix} 1 & 1 & 0 & 2 \\ 2 & 1 & 4 & 1 \\ 0 & 3 & 2 & 1 \end{bmatrix}$$

$$a = \begin{bmatrix} 1 \\ 0 \\ 2 \\ 4 \end{bmatrix}$$

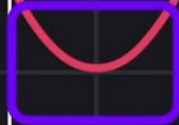
미적분학

$$y = f(x)$$

x

여기서 성능이 제일 좋음

최적화



통계

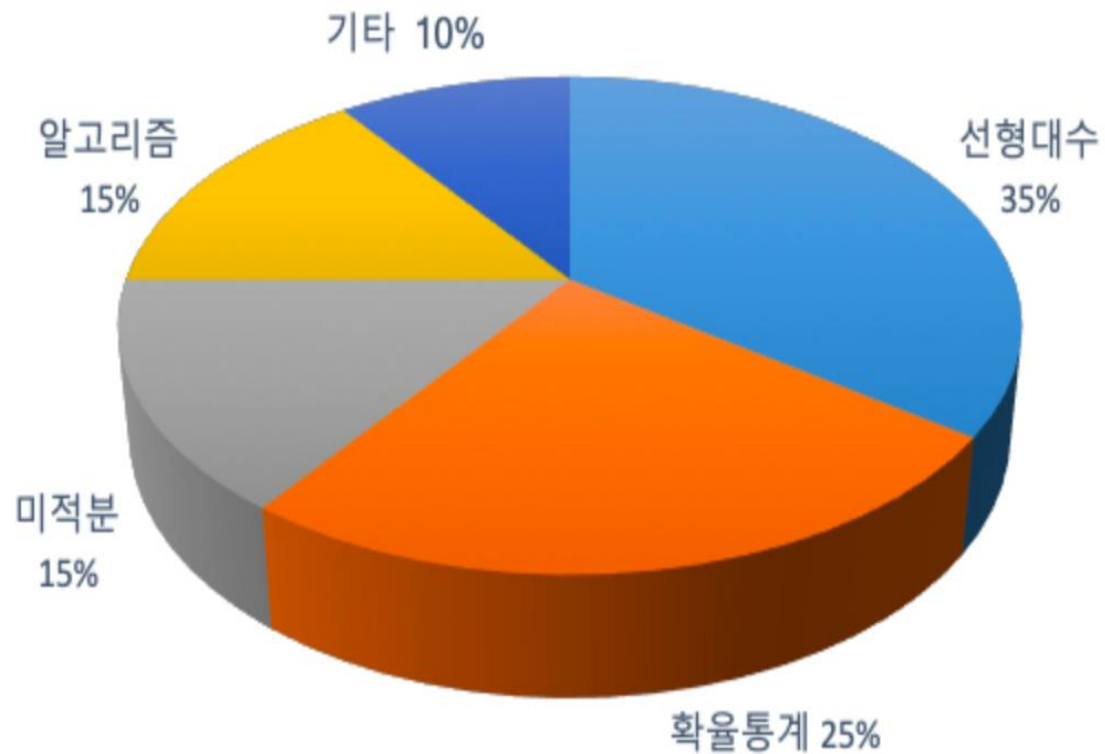
데이터에서
큰 흐름 파악!



예측에 사용



머신러닝에 필요한 수학의 중요도



기본 개념	스칼라 벡터 벡터 공간 스칼라 곱셈 벡터 사영 선형생성 선형 변환 사영작용소 일차 독립 집합 선형 결합 기저 기저 변경 행 벡터와 열 벡터 행 공간과 열 공간 직교 영공간 고윳값과 고유 벡터 외적 내적 공간 스칼라곱 전치행렬 그람-슈미트 과정 일차 방정식
벡터 대수	벡터곱 삼중곱 7차원 외적
다중선형 대수학	기하적 대수학 외대수 이중벡터 다중벡터 텐서 아우터모피즘
행렬	블록 분해 가역 소행렬 곱셈 계수 변환 크라메르 공식 가우스 소거법 행렬식
대수적 구성	쌍대 직합 함수 공간 몫공간 부분공간 텐서곱



선형대수학

"Change your thoughts
and you change your world"

—Norman Vincent Peale—

선형대수학이란?

- 일차식이나 일차 함수를 공부하는 학문

가장 높은 차수가 1인 **다항식**

$$\underline{2x^3} - \underline{y^2} + \underline{4y} + \underline{1}$$

항이 여러 개여서 **다항식**

$$\begin{array}{c} \text{계수} \quad \underline{2x^3} \quad \text{차수} \\ \text{변수} \end{array}$$

$$\begin{array}{c} \text{계수} \quad \underline{-y^2} \quad \text{차수} \\ \text{변수} \end{array}$$

$$\begin{array}{c} \text{계수} \quad \underline{1} \quad \text{차수} \\ \text{변수} \end{array}$$

$$6x + 4y - 7$$

차수가 1인 일차식!

일차 함수

$$y = 3x + 6$$

일차 함수

$$f(x) = 3x + 6$$

$$f(x, y) = 2x^3 - y^2 + 4y + 1$$

일차식 표기법

$$f(\underline{x}, y) = 3x + 6y + 4$$

$$f(x, y, z, a, b, c) = 4x + y - 2z + 16a + 7b + 3c$$

$$f(\underline{x_0, x_1, \dots, x_n}) = a_0 \underline{x_0} + a_1 \underline{x_1} + \dots + a_n \underline{x_n} + b$$

[행결과 벡터]

행렬: Matrix

$$A = \begin{bmatrix} 1 & 1 & 0 & 2 \\ 2 & 1 & 4 & 1 \\ 0 & 3 & 2 & 1 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & 1 & 0 & 2 \\ 2 & 1 & 4 & 1 \\ 0 & 3 & 2 & 1 \end{bmatrix}$$

행 (row)

열 (column)

3 x 4 행렬

$$A = \begin{bmatrix} 1 & 1 & 0 & 2 \\ 2 & 1 & 4 & 1 \\ 0 & 3 & 2 & 1 \end{bmatrix}$$


4 × 2 행렬

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

행: 4

열: 2

$$A = \begin{bmatrix} 1 & 1 & 0 & 2 \\ 2 & 1 & 4 & 1 \\ 0 & 3 & 2 & 1 \end{bmatrix}$$

A_{ij}  i 번째 행, j 번째 열에 있는 원소

$$A = \begin{bmatrix} 1 & 1 & 0 & 2 \\ 2 & 1 & 4 & 1 \\ 0 & 3 & 2 & 1 \end{bmatrix}$$

A_{23}

m x n 행렬

$$A = \begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1n} \\ A_{21} & A_{22} & \cdots & A_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ A_{m1} & A_{m2} & \cdots & A_{mn} \end{bmatrix}$$

벡터 : Vector

$$a = \begin{bmatrix} 1 \\ 0 \\ 2 \\ 4 \\ 1 \end{bmatrix}$$

$$b = [1 \quad 0 \quad 2 \quad 4]$$

열 벡터

$$a = \begin{bmatrix} 1 \\ 0 \\ 2 \\ 4 \\ 1 \end{bmatrix}$$

5차원 열벡터

행 벡터

$$b = [1 \quad 0 \quad 2 \quad 4]$$

4차원 행 벡터

[행렬 덧셈과 곱셈]

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

$$B = \begin{bmatrix} 3 & 1 \\ 2 & 4 \end{bmatrix}$$

$$A + B = \begin{bmatrix} & \\ & \end{bmatrix}$$

행렬 곱셈

i는 스칼라

$$i = 5$$

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

$$iA = \begin{bmatrix} 5 \times 3 & 5 \times 1 \\ 5 \times 2 & 5 \times 3 \end{bmatrix} = \begin{bmatrix} 15 & 5 \\ 10 & 15 \end{bmatrix}$$

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

$$AB = \begin{bmatrix} & \\ & \end{bmatrix}$$

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

$$1 \times 3 \quad 3 \times 2 \quad 1 \times 2$$

$$AB = \begin{bmatrix} & \\ & \end{bmatrix}$$

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

$$1 \times 3 \quad 3 \times 2 \quad 1 \times 1$$

$$AB = \begin{bmatrix} 20 & \end{bmatrix}$$

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

$$2 \times 5 + 2 \times 4 + 1 \times 3$$

$$AB = \begin{bmatrix} 20 & 13 \end{bmatrix}$$

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

행렬 A와 B를 곱하려면, A의 열과 B의 행의 수가 같아야함!

2 × 2 행렬

$$AB = \begin{bmatrix} 20 & 13 \\ 21 & 17 \end{bmatrix}$$

행: 2 열: 2

순서와 상관 **없음**

$$3 \times 4 = 4 \times 3 = 12$$

$$AB \neq BA$$

순서와 상관 **있음**

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

2 × 3

계산 가능!

AB

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

3 × 4

계산 불가능!

BA



NumPy

넘파이

숫자와 관련한 파이썬 도구



NumPy

왜 중요할까?

numpy 배열
(numpy array)

≈

python리스트



Python으로 Numpy array 만들기

"Change your thoughts
and you change your world"

—Norman Vincent Peale—

파이썬 리스트를 통해 생성

`numpy` 모듈의 `array` 메소드에 파라미터로 파이썬 리스트를 넘겨주면 `numpy array`가 리턴됩니다.

```
array1 = numpy.array([2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31])  
  
print(array1)
```

```
[ 2  3  5  7 11 13 17 19 23 29 31]
```


균일한 값으로 생성

`numpy` 모듈의 `full` 메소드를 사용하면, 모든 값이 같은 `numpy array`를 생성할 수 있습니다.

```
array1 = numpy.full(6, 7)
```

```
print(array1)
```

```
[7 7 7 7 7 7]
```

랜덤한 값들로 생성

어쩔 때는 임의의 값들로 배열을 생성시키고 싶습니다. 그럴 때는 `numpy`의 `random` 모듈의 `random` 함수를 사용하면 됩니다.

`numpy` 모듈 안에 `random`이라는 모듈이 있고, 그 안에 또 `random`이라는 함수가 있는 겁니다!

```
array1 = numpy.random.random(6)
array2 = numpy.random.random(6)

print(array1)
print()
print(array2)
```

연속된 값들이 담긴 numpy array 생성

`numpy` 모듈의 `arange` 함수를 사용하면 연속된 값들이 담겨 있는 `numpy array`를 생성할 수 있습니다.

`arange` 함수는 파이썬의 기본 함수인 `range`와 굉장히 비슷한 원리로 동작하는데요. 파라미터가 1개인 경우, 2개인 경우, 3개인 경우 모두 살펴봅시다.

파라미터 1개

`arange(m)` 을 하면 0부터 `m-1`까지의 값들이 담긴 `numpy array`가 리턴됩니다.

```
array1 = numpy.arange(6)
print(array1)
```

```
[0 1 2 3 4 5]
```

파라미터 2개

`arange(n, m)`을 하면 `n`부터 `m-1`까지의 값들이 담긴 numpy array가 리턴됩니다.

```
array1 = numpy.arange(2, 7)
print(array1)
```

```
[2 3 4 5 6]
```

파라미터 3개

`arange(n, m, s)`를 하면 `n`부터 `m-1`까지의 값들 중 간격이 `s`인 값들이 담긴 numpy array가 리턴됩니다.

```
array1 = numpy.arange(3, 17, 3)
print(array1)
```

```
[ 3  6  9 12 15]
```



실습

"Change your thoughts
and you change your world"

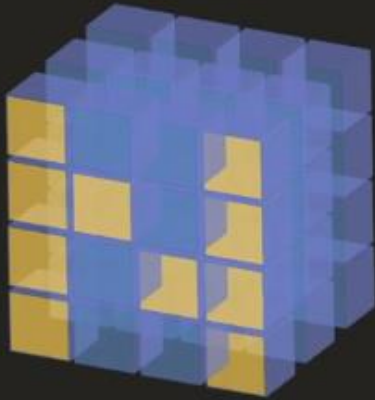
—Norman Vincent Peale—



Numpy array vs Python list

"Change your thoughts
and you change your world"

—Norman Vincent Peale—

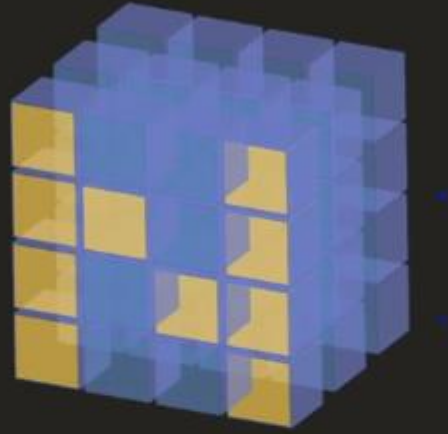


numpy array



python list

여러 값을 보관한다는 공통점



numpy array

굳이 numpy array를 쓰는 이유는?

문법 차이

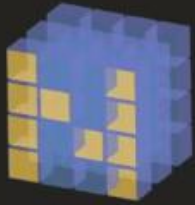


```
[10, 5, 3, 7, 1, 5]  
+ [10, 5, 3, 7, 1, 5]  
-----  
[20, 10, 6, 14, 2, 10]
```



```
[10, 5, 3, 7, 1, 5]  
+ [10, 5, 3, 7, 1, 5]  
-----  
[10, 5, 3, 7, 1, 5, 10, 5, 3, 7, 1, 5]
```

문법 차이



[10, 5, 3, 7, 1, 5]

+ 5

[15, 10, 8, 12, 6, 10]



[10, 5, 3, 7, 1, 5]

+ 5

~~백셈
나눗셈~~

문법 차이



`[10, 5, 3, 7, 1, 5]`

`* 3`

`[30, 15, 9, 21, 3, 15]`

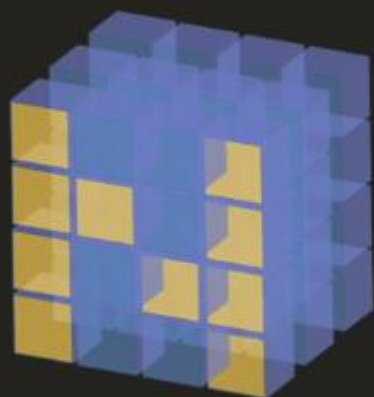


`[10, 5, 3, 7, 1, 5]`

`* 3`

`[10, 5, 3, 7, 1, 5, 10, 5, 3, 7, 1, 5,
10, 5, 3, 7, 1, 5]`

성능 차이



문법이 간단 + 뛰어난 성능

왜 이런 성능 차이가 있죠?



```
[17, 9, True, 'Hello', 1, 5]
```

속도 개선



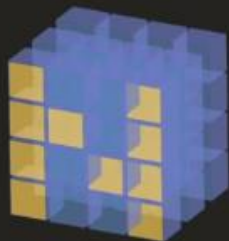
```
[17, 9, 10, 4, 1, 5]
```

```
['Hello', 'Cat', 'Banana', 'Pig']
```

언제 어떤 걸 써야 하나요?



값을 추가하고 제거하는 일



수치 계산이 많고 복잡할 때
행렬같은 다차원 배열의 경우



Python으로 Numpy 행렬 만들기

"Change your thoughts
and you change your world"

—Norman Vincent Peale—

실습과제

1) $A = \begin{pmatrix} 1 & 2 & 3 \\ -1 & 0 & 2 \\ 3 & 4 & 5 \\ 2 & -2 & -3 \end{pmatrix}$ 4X3 행렬 A를 numpy array로 정의해보기

2) $B = \begin{pmatrix} 0 & 2 \\ 1 & -1 \\ 3 & -4 \end{pmatrix}$ 3x2 행렬 B를 numpy array로 정의해보기

3) 정의한 행렬 A 의 2 행 2 열 원소 찾기

4) 정의한 행렬 B 의 3 행 1열 원소 찾기

(프로그래밍할 때는 인덱스를 0부터 세는 걸 잊지마세요!!)

질문 1

$$\begin{bmatrix} 1 & 2 & -1 \\ 1 & 1 & -3 \end{bmatrix} + \begin{bmatrix} 3 & 2 & -2 \\ -1 & 2 & 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix} \text{ 다음 식에서 } a_{13} \text{ 에 들어갈 값을 쓰세요.}$$

질문 2

$$\begin{bmatrix} 1 & 2 & -1 \\ 1 & 1 & -3 \end{bmatrix} \times \begin{bmatrix} 3 \\ 1 \\ 2 \end{bmatrix} = \begin{bmatrix} a_{11} \\ a_{21} \end{bmatrix} \text{ 다음 식에서 } a_{11} \text{ 에 들어갈 값을 고르세요.}$$

질문 3

$$\begin{bmatrix} 1 & 2 & -1 \\ 1 & 1 & -3 \end{bmatrix} \times \begin{bmatrix} 3 & 2 \\ 1 & 1 \\ 2 & -1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \text{ 다음 식에서 } a_{12} \text{ 에 들어갈 값을 쓰세요.}$$