Fregic

선형회귀 (Linear Regression)

- 21wp 시원

오늘 배울 내용

선형회귀
Linear Regression

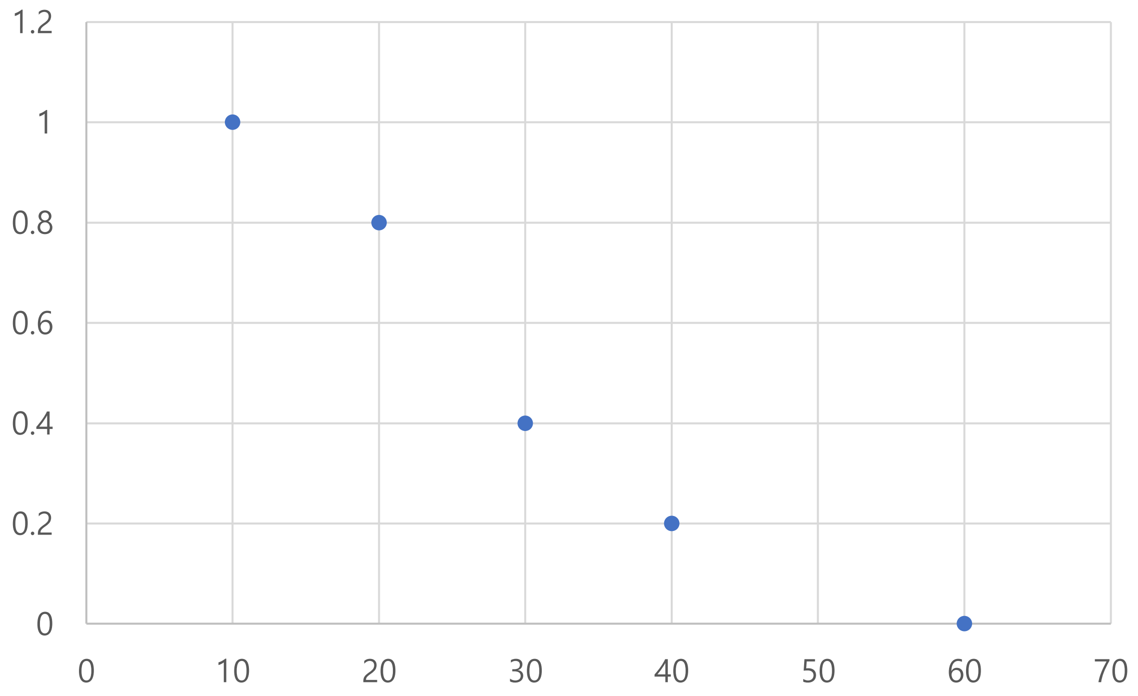
AI > **머신러닝** Machine Learning > **지도학습** supervised Learning > **회귀** Regression

https://plotly.com/~siwon/3/

머신러닝 (Machine Learning)

(일반적 상황서)
➡️ 최적의 함수를 찾는 방법

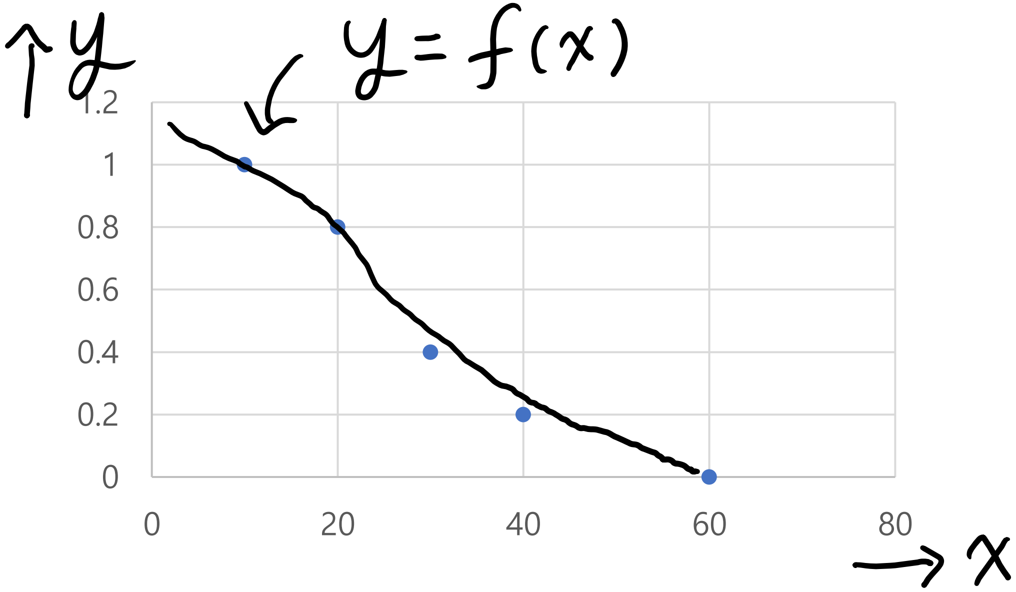
랩탑 사용 시간	시력
10	1
20	0.8
30	0.4
40	0.2
60	0



(일반적 상황서)

➡️ 최적의 함수를 찾는 방법

랩탑 사용 시간	시력
10	1
20	0.8
30	0.4
40	0.2
60	0



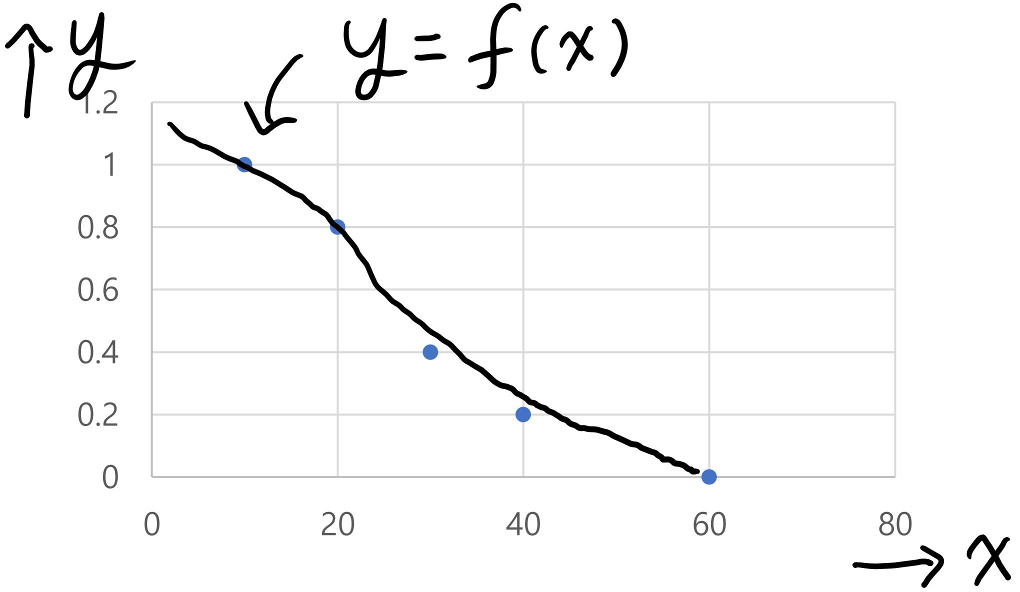
📂 목표: 랩탑 사용시간으로 시력을 예측하는 모델 제작

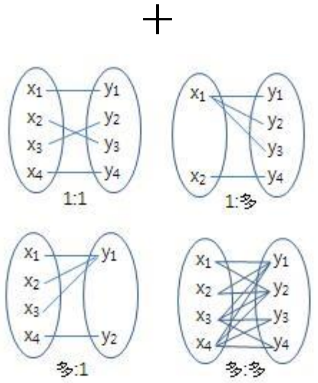
✨📂 모델 평가 방법: 모델이 예상한 값과 실제 학생의 시력을 비교함

➡️ 모델 예측값과 실제 값의 차이가 작을 수록 모델이 우수함

지도 학습 (Supervised Learning)

랩탑 사용 시간	시력
10	1
20	0.8
30	0.4
40	0.2
60	0





Mapping(사상): 대응관계 함수와 유사

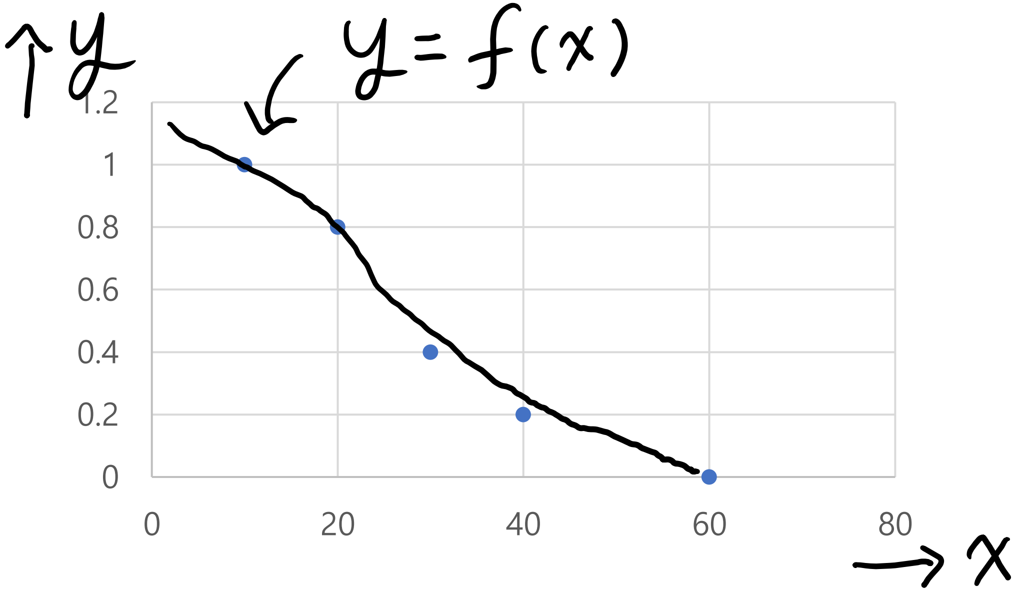
📁 10->1, 20->0.8, 30 ->0.4 …. Mapping(사상) 되어 있음

📁 지도 학습: 레이블 값(1, 0.8, 0.4…)을 모델이 정하지 않음

➡️ 사람이 레이블 값을 추가하는 과정이 필요

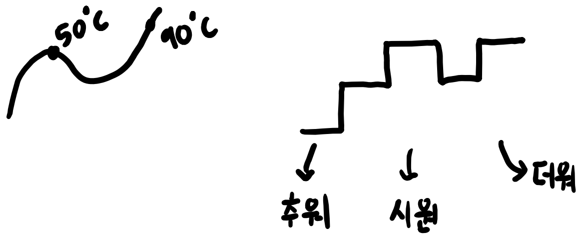
회귀 (Regression)

랩탑 사용 시간	시력
10	1
20	0.8
30	0.4
40	0.2
60	0



📁 회귀: 값을 예측함, 값이 '연속적 '

📁 회귀 vs 분류 ➡️ 연속적 vs 불연속적



선형 (Linear)

가산성

$$f(x_1 + x_2) = f(x_1) + f(x_2)$$

$$g(x) = 5x + 3$$
$$g(3+5) \neq g(3) + g(5)$$
$$g(3 \times 5) \neq 3g(5)$$

➡️ $5x + 3$은 선형이 아님

동차성

$$f(kx) = kf(x)$$

\+
선형 대수 (Linear Algebra)
: 벡터를 다루는 대수학 분야

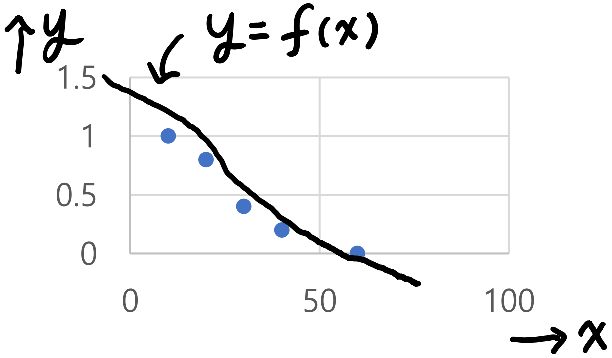
$f(x) = 5x + 3$: 선형 X,
Affine Space에서 정의됨

⬇️ Y축 -3 평행이동

$f(x) = 5x$: 선형 O

선형회귀, 왜 사용하지??

랩탑 사용 시간	시력
10	1
20	0.8
30	0.4
40	0.2
60	0



📂 목표: 랩탑 사용시간으로 시력을 예측하는 모델 제작

📂 가설: 랩탑 사용시간과 시력은 반비례 관계가 있을 거야!

➡️ 함수로 나타내면 '선형': $y = \theta x$꼴 (Wx+b)
➡️ 이 선형을 목표 모델로 만들자!

✨📂 모델 평가 방법: 모델이 예상한 값과 실제 학생의 시력을 비교함

➡️ 모델 예측값과 실제가 차이가 적을 수록 모델이 우수함

📂 선형회귀: 입력값 $x \in \mathbb{R}^D$에 대응하는 레이블된 함수값 $y \in \mathbb{R}$을 찾는 것

+ 함수를 찾기 위해서라면….

- 모형(모형 유형)과 모수화(parametrization) 선택
  - ➡️ 과연 함수가 $y = \theta x$꼴로 나올까??
- 좋은 모수(paramerer) 찾기
  - ➡️ θ에 어떤 값을 대입해야 좋은 모수라는 말을 들을까??
- 과적합(Overfitting) 및 모형 선택
  - ➡️ 혹시 모델이 학습한 데이터에서만 좋은 결과를 내지 않는가?? ➡️ 일반적 상황이 중요!
- 손실 함수와 사전 확률 모수(parameter priors) 사이의 관계
  + MAP
- 불확실성의 모델링
  - ➡️ 선형회귀는 최적의 근사값을 찾는 것이 일반적, 흔히 피할 수 없는 오차를 잡음이라 말하며 ϵ로 표현

parameter

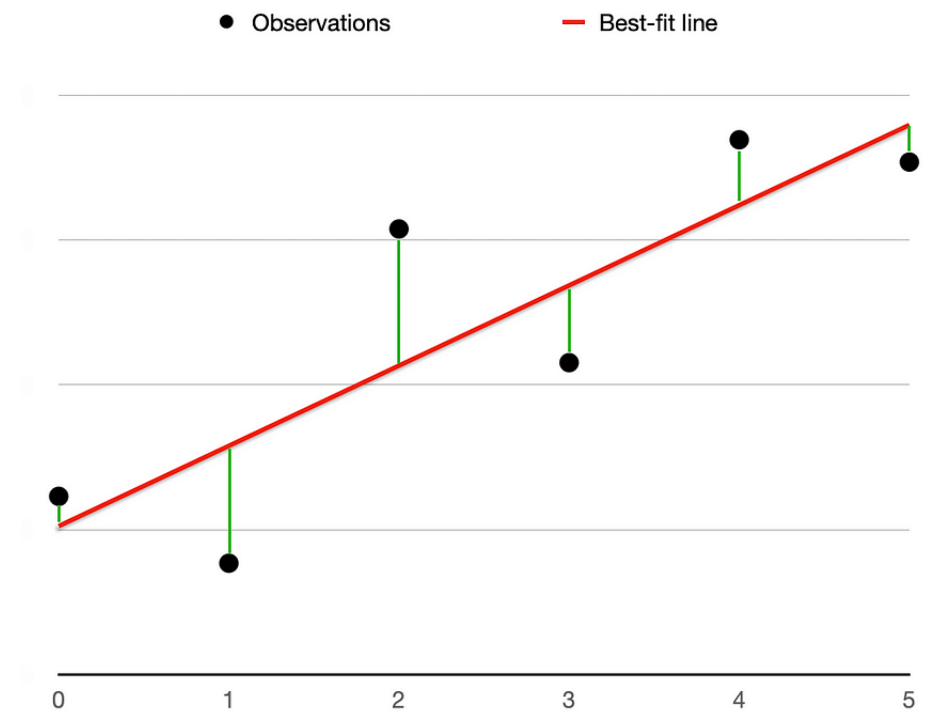
```
def fresh(logic):
    pass
```

- 어떤 책이 말하기를…

다시, 평가에 대하여

✨📁모델 평가 방법: 모델이 예상한 값과 실제 학생의 시력을 비교함

➡️ 모델 예측값과 실제가 차이가 적을 수록 모델이 우수함



📁빨간 선: 예측된 선
📁검은 점: 실제 데이터

➡️ 차이가 적을 수록 모델이 우수하니
차이는…

예시: 같은 x에 대하여 실제 데이터와 차이값 제곱을 모두 더해

$$\sum_{i=1}^{m}(f(x_i) - y_i)^2$$

$+$

$$\sum_{i=0}^{9} i$$

시그마 (Sigma)
: 그리스 문자 18번째,
수학서 모두 더하기

```
sum = 0
for i in range(10):
    sum += i
```

$$\prod_{i=0}^{9} i$$

파이 (Pi)
: 수학서 모두 곱하기

```
sum = 1
for i in range(10):
    sum *= i
```

최적의 함수 구하기

📂 최대 우도 추정법 (Maximum Likelihood Estimation, MLE)

➡️ 최적화된 함수 $y = \theta x$에 대하여 최적의 모수(parameter) θ를 찾는 것을 목표함

예시 상황을 가정해 보아

동전 앞면 H,
뒷면 T

5번 던지니…
H, T, T, H, H
(dataset)

그럼, MLE점 관점으로
H가 나올 확률은??

MLE

- D = [H, T, T, H, H]: Data set
- a^H: 데이터셋에서 H의 개수 -> 3, a^T: T의 개수 -> 2
- θ: H일 확률 ($0 \le \theta \le 1$), $1-\theta$: T일 확률 ($0 \le 1-\theta \le 1$),
- P: 확률 함수. P(D)라면 순서대로 HTTHH가 나올 확률
    - ➡️ $\theta(1-\theta)(1-\theta)\theta\theta$
    - ➡️ $\mathrm{P}(D|\theta)$: θ가 주어졌을 때, D의 조건부 확률 (D가 발생할 확률)

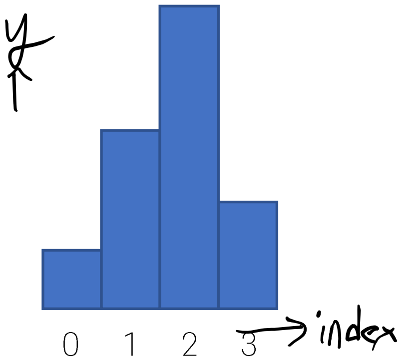
$$= \theta^{a^H}(1-\theta)^{a^T}$$

=>θ가 parameter 느낌?

$$\underbrace{p(\boldsymbol{x} \mid \boldsymbol{y})}_{\text{posterior}} = \frac{\overbrace{p(\boldsymbol{y} \mid \boldsymbol{x})}^{\text{likelihood}}\,\overbrace{p(\boldsymbol{x})}^{\text{prior}}}{\underbrace{p(\boldsymbol{y})}_{\text{evidence}}}$$

- argmax: 함수에서 y값이 최대가 되는 x값

```
>>> import numpy as np
>>> np.argmax([1,4,7,2])
2
```

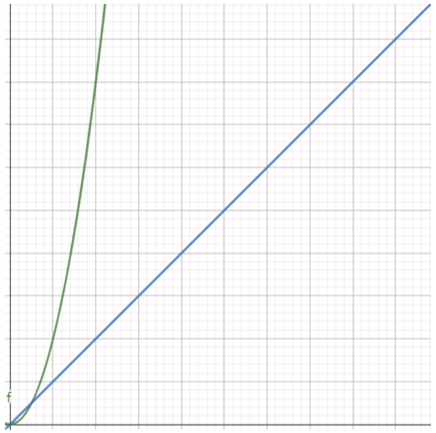


MLE 예시

$$P(D|\theta) = \theta^{a^H}(1-\theta)^{a^T}$$
$$\hat{\theta} = argmax_\theta \, P(D|\theta)$$
$$= argmax_\theta \, \theta^{a^H}(1-\theta)^{a^T}$$

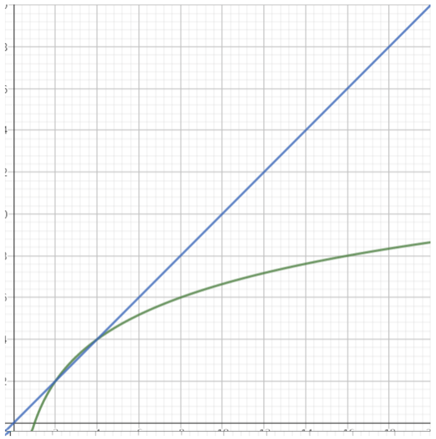
📂log 함수

$a^x = b$ => $x = \log_a b$



$y = x^2$
기하급수적

2



$y = \log_2 x^2$
아니 기하급수적

✨📂*를 +로 변환 가능

ab가 있는데... 로그를 취하면

$log(ab) = \log a + \log b$

➡️큰 수의 효율적 표현 가능

MLE 예시

$$P(D|\theta) = \theta^{a^H}(1-\theta)^{a^T}$$
$$\hat{\theta} = argmax_\theta\, P(D|\theta)$$
$$= argmax_\theta\, \theta^{a^H}(1-\theta)^{a^T}$$

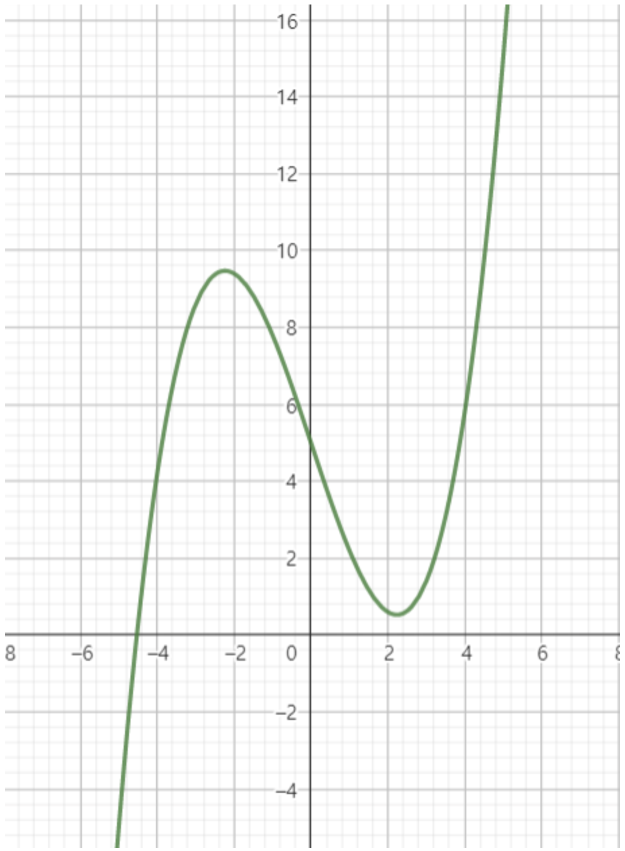
위스키 한 병과, 로그를 취해

$$= argmax_\theta \log\left(\theta^{a^H}(1-\theta)^{a^T}\right)$$ ➡️➡️➡️➡️

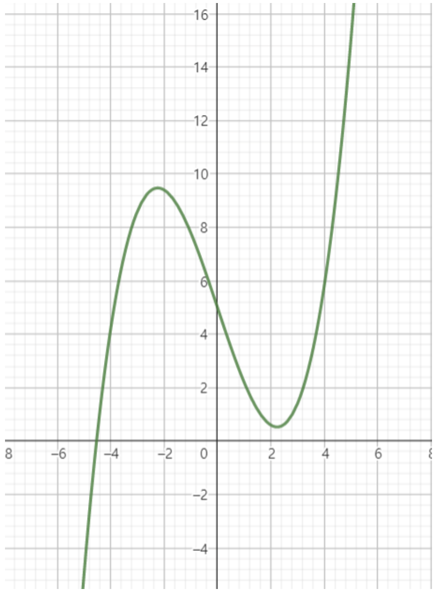
1. $a > b$이면 $\log a > \log b$
2. $argmax$: 함수에서 y값이 최대가 되는 x값
   즉 $argmax\ a = argmax \log a$
3. $\log a^b = b \log a$

$$= argmax_\theta \log(a^H \log\theta + a^T \log(1-\theta))$$

MLE 예시, argmax, 최댓값서 x좌표 어찌 구하나??



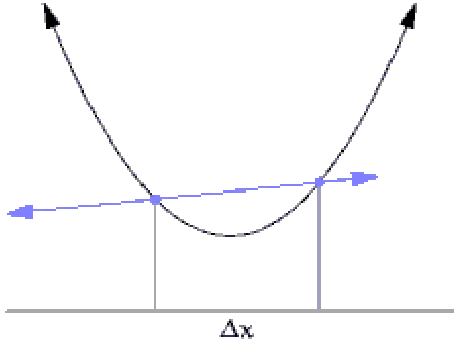
MLE 예시, argmax, 최댓값서 x좌표 어찌 구하나??

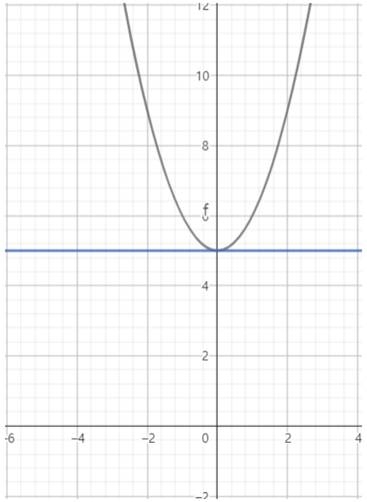


📂미분: 작게 나누다
'변화율', '기울기'

함수f가 x=0 지점에서
기울기는 0이야!

-> 접선이 y=0x+5,
즉 기울기가 0이야!

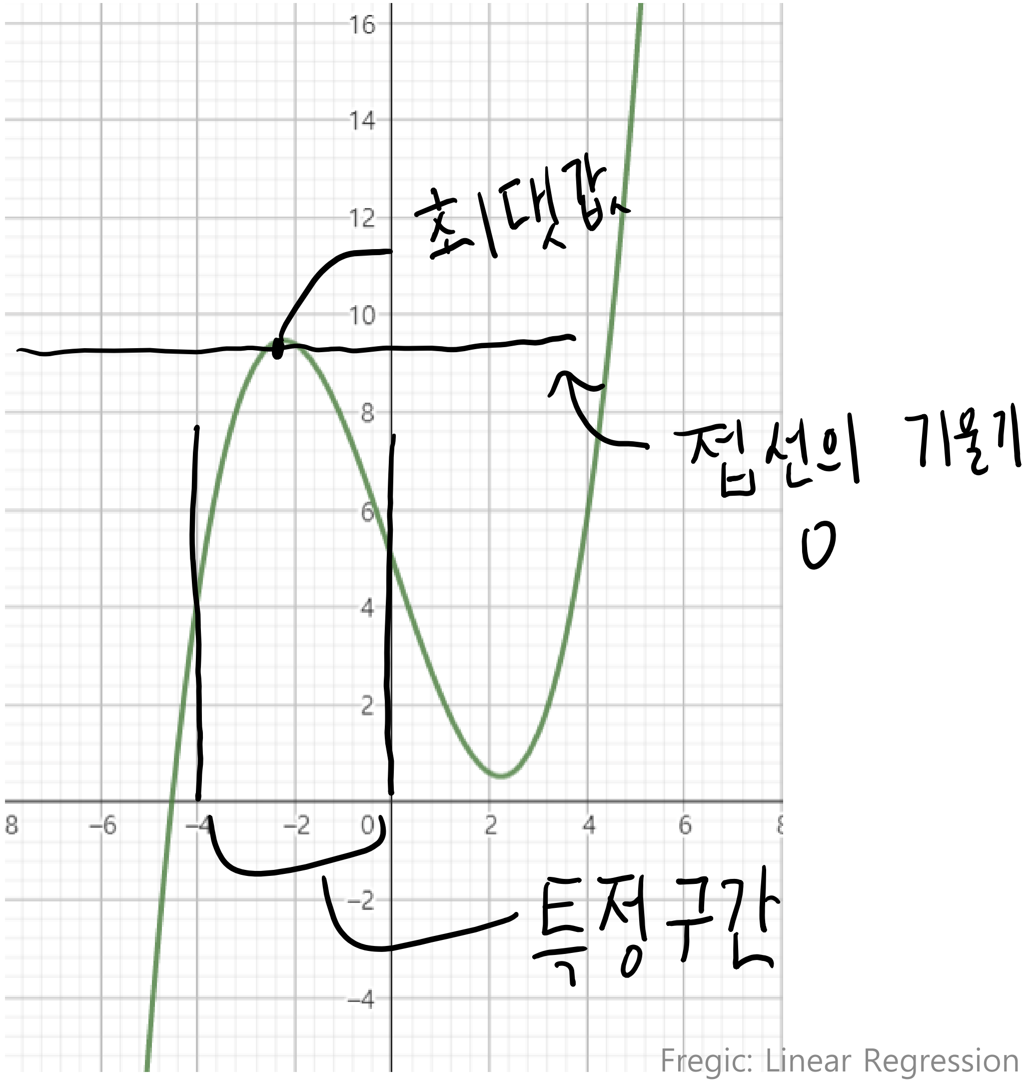




\+
엡실론-델타 논법

📂f(x)를 미분,
$\frac{d}{dx}f(x)$: x좌표서 접선의 기울기

MLE 예시, argmax, 최댓값서 x좌표 어찌 구하나??



📂미분
'변화율', '기울기'

함수f가 x=0 지점에서
기울기는 0이야!

-> 접선이 y=0x+5,
즉 기울기가 0이야!

MLE 예시

$$\mathrm{P}(D|\theta) = \theta^{a^H}(1-\theta)^{a^T}$$
$$\hat{\theta} = argmax_\theta\ \mathrm{P}(D|\theta)$$
$$= argmax_\theta\ \theta^{a^H}(1-\theta)^{a^T}$$

위스키 한 병과, 로그를 취해

$$= argmax_\theta \log\left(\theta^{a^H}(1-\theta)^{a^T}\right)$$
$$= argmax_\theta(a^H \log\theta + a^T \log(1-\theta))$$

밤 하늘의 별, 미분

$$\frac{d}{d\theta}(a^H \log\theta + a^T \log(1-\theta)) = 0 \Rightarrow \frac{a^H}{\theta} - \frac{a^T}{1-\theta} = 0 \Rightarrow \theta = \frac{a^H}{a^T + a^H} = \hat{\theta}$$

➡️D = [H, T, T, H, H], $a^H = 3, a^T = 2 \Rightarrow \hat{\theta} = \frac{3}{5} = 0.6$

H가 나올 확률: 60%

MLE를 선형회귀에 적용하기 전에.. 전치행렬

전치 행렬(Transposed Matrix)

A

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

행렬 A의 전치 행렬은 A^T로 표현

\+

https://www.siwonsw.com/paper/matrix

MLE를 선형회귀에

📂목표: 최적화된 함수 $y = \theta x$를 찾음 => 즉 우리가 찾고 싶은 값은 θ

랩탑 사용 시간	시력
10	1
20	0.8
30	0.4
40	0.2
60	0

📂랩탑 사용 시간을 바탕으로 시력을 예측
➡️x가 랩탑 사용 시간, y가 시력

'10, 20, 30…' 랩탑 사용 시간을 행열 X로 정의
➡️x = [[1, 10], [1, 20], [1, 30], [1, 40], [1, 60]]
첫 값은 무조건 1 (사실 0만 아니면 가능해요!)

📂$Y = \theta X, \ y \Rightarrow Wx + b$ ➡️행렬 곱(dot product) $\begin{array}{cc} \theta & \times \quad X \\ \begin{bmatrix} b \\ W \end{bmatrix} & [1 \quad x] = b + Wx \end{array}$

MLE를 선형회귀에

📂목표: 최적화된 함수 $y = \theta x$를 찾음 => 즉 우리가 찾고 싶은 값은 θ

랩탑 사용 시간	시력
10	1
20	0.8
30	0.4
40	0.2
60	0

📂랩탑 사용 시간을 바탕으로 시력을 예측

'1, 0.8, 0.4...' 시력을 행렬 Y로 정의

➡️y = [1, 0.8, 0.4, 0.2, 0]

📂실제 데이터 함수: f(X)

📂모델이 예측한 함수: $\hat{f}(X)$

➡️ $\hat{f}(x) = f(X) + \varepsilon$

➡️ε: noise, 잡음

MLE를 선형회귀에

📂모델 평가 방법: $\sum_{i=1}^{m}(f(x_i) - y_i)^2$ ➡️ x_i와 y_i를 행렬 x, y로 일반화, 시그마 필요 없어짐

즉: $\left(f(X) - \hat{f}(X)\right)^2$

$$\hat{\theta} = argmin_{\theta}\left(f(X) - \hat{f}(X)\right)^2 = argmin_{\theta}(Y - X\theta)^2$$

$$= \ argmin_{\theta}\ (Y - X\theta)^T(Y - X\theta) = argmin_{\theta}\ (\theta^T X^T X\theta - 2\theta^T X^T Y + Y^T Y)$$

$$= \ argmin_{\theta}(\theta^T X^T X\theta - 2\theta^T X^T Y)$$, 미분

$$+ \ \frac{d}{d\theta}(\theta^T X^T X\theta - 2\theta^T X^T Y) = 0$$

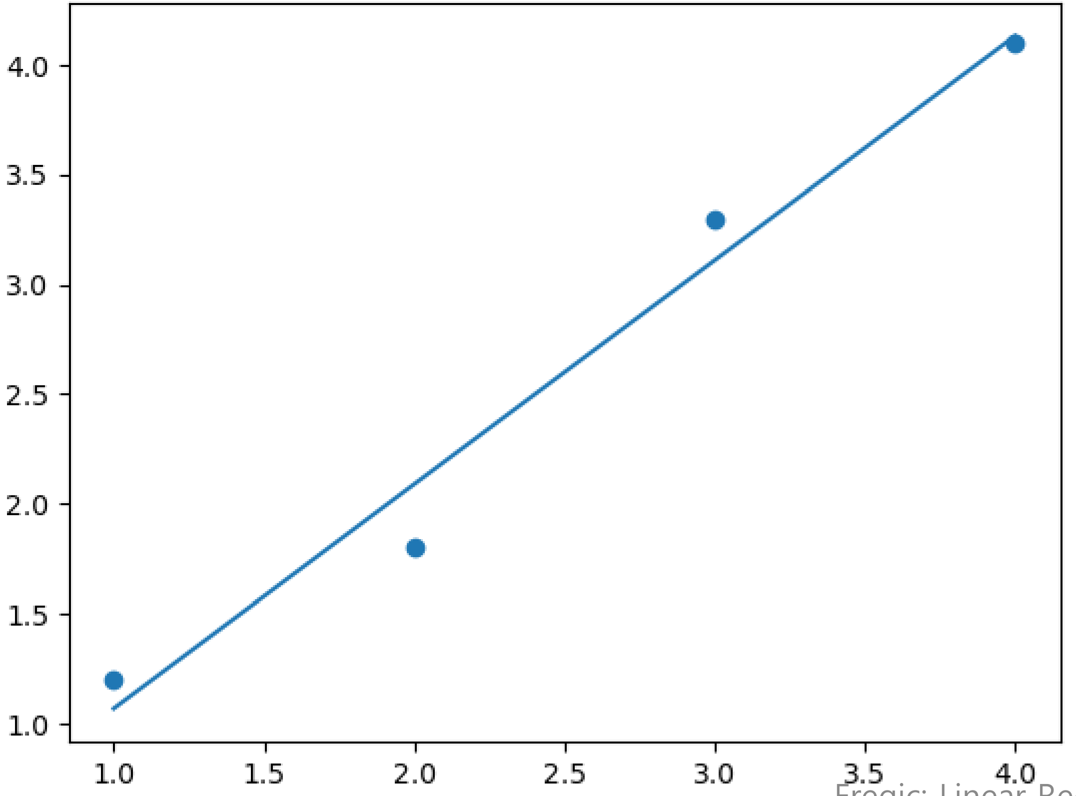
$$2X^T X\theta - 2\theta^T X^T Y = 0$$

$$\theta = (X^T X)^{-1} X^T Y = \hat{\theta}$$

MLE를 선형회귀에

📂$\theta = (X^T X)^{-1} X^T Y$를 파이썬으로… (예제1)

```
theta = np.linalg.inv(X.T.dot(X)).dot(X.T).dot(Y)
```



아름다운 결과 :)

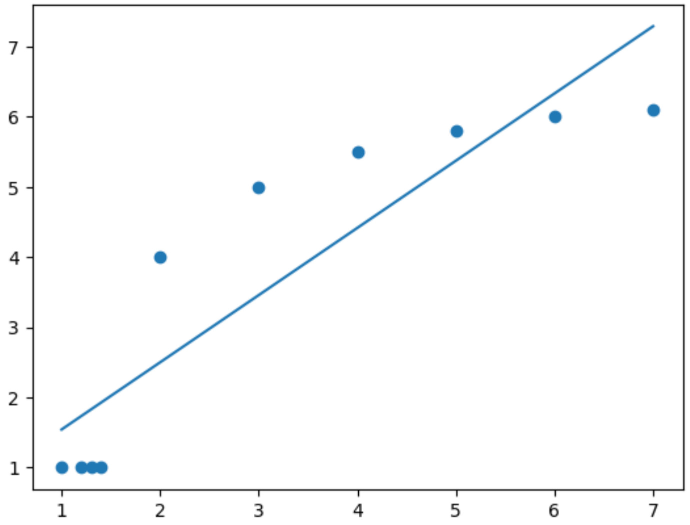
📂모델 평가 방법 $\left(f(X) - \hat{f}(X)\right)^2$

```
[0.0169 0.0841 0.0361 0.0009]
```

➡️작은 수, 만족해,
이보다 작을 수 없을걸?

+MLE 선형회귀서…

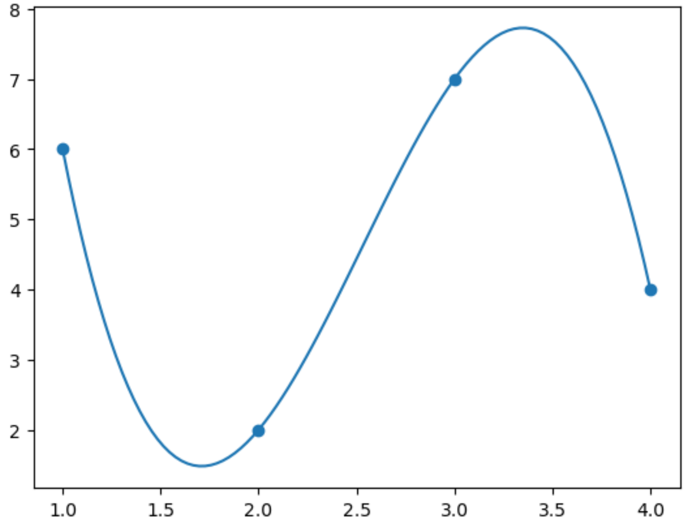
예제 2



📂Underfitting

+ MAP
+ 베이지안 선형 회귀

예제 3



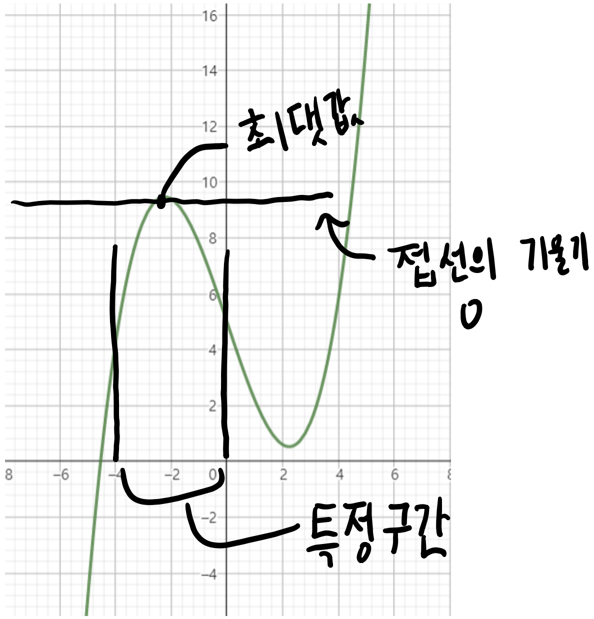
📂3차 방정식 꼴

+
'비선형 변환 φ에 대하여
$y = \varphi^T(x)\theta$도 선형 회귀 모델임 '

경사 하강법 (Gradient descent)

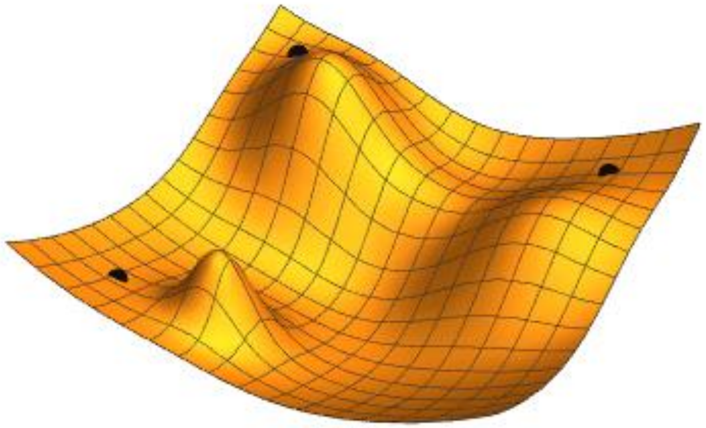
- 📂 (다음 시간에 배울) 로지스틱 회귀를 MLE 관점에서 해결 하려니…
  선형회귀와 같이 깔끔한 결과 X => 경사 상승법 필요
- 📂 (나중에 배울) 딥러닝을 위해 경사 하강법 / 상승법 필요

➡️ agrmax 구하는 방법 => 경사 상승법
➡️ agrmin 구하는 방법 => 경사 하강법



🗂️ 기울기가 양수일 때 왼쪽으로
기울기가 음수일 때 오른쪽으로
이동하면 최솟값을 구할 수 있을거야!

경사 하강법 (Gradient descent)

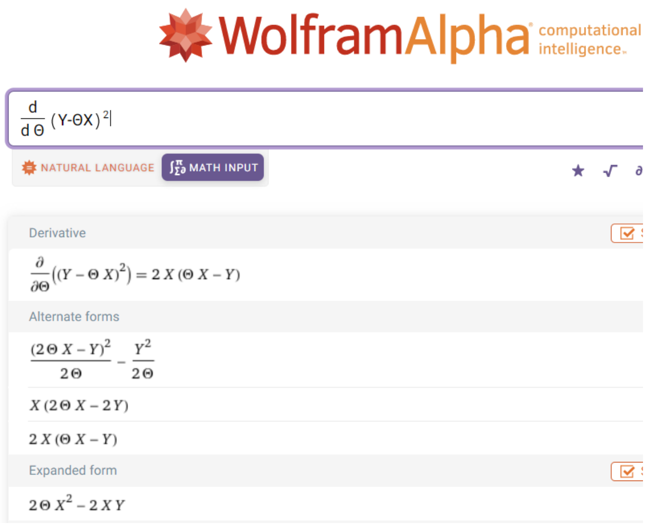


📁 모델 평가 방법: $\left(f(X) - \hat{f}(X)\right)^2$이 최대한 작아야함

📁 기울기 구하기: 미분 $\Rightarrow G = 2X^T(X\theta - Y)$
학습(Learning): θ를 구하는 과정

🌟📁 기울기가 양수일 때 왼쪽으로
기울기가 음수일 때 오른쪽으로
이동하면 최솟값을 구할 수 있을거야!

➡️ if(G > 0):
theta -= 1
else:
theta += 1



➡️ 기울기 절댓값이 크면 빠르게…
$new\ \theta = \theta - rG$ 반복 (학습)

r = Learning Late

방지

경사 하강법 (Gradient descent)

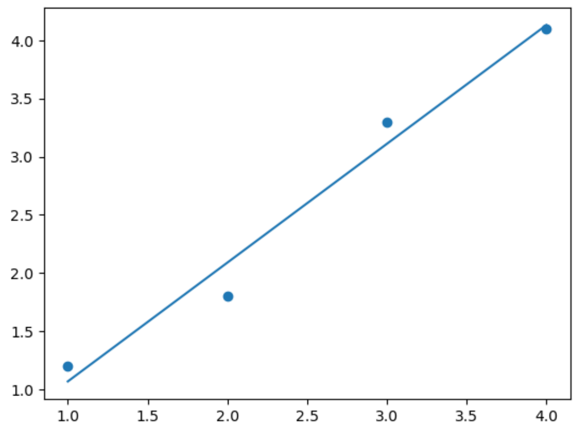
코드로 구현하면 (예제 4)

```python
# 경사 하강법
X = np.array([[1, 1], [1, 2], [1, 3], [1, 4]])
Y = np.array([1.2, 1.8, 3.3, 4.1])

theta = np.random.randn(2)
learning_rate = 0.01

for i in range(1000):
    grad = 2 * X.T.dot(X.dot(theta) - Y)
    theta = theta - learning_rate * grad

# plot을 통한 시각화
plt.scatter(X[:, 1], Y)
plt.plot(X[:, 1], X.dot(theta))
plt.show()
```



아름다운 결과 :)

모델 평가 방법 $\left(f(X) - \hat{f}(X)\right)^2$

```
[0.09875564 0.04028702 0.03397141 0.01707125]
```

➡️ 반복이 많을 수록 더 정확해짐!
= MLE 결과와 값이 비슷해짐 (수렴)

Fregic

선형회귀 (Linear Regression)