선형 회귀(Linear Regression)

선형 회귀(Linear Regression)

시험 공부하는 시간을 늘릴수록 성적이 잘 나옴, 집의 평수가 클수록 집의 매매가격이 비쌈 → 어떤 변수 값에 따라서 특정 변수의 값이 영향을 받고 있음 → **다른 변수의 값을 변하게 하는 변수를 독립변수 x, 변수 x에 의하여 값이 종속적으로 변하는 변수를 종속변수 y라고 함 (단순 선형 회귀 분석 사용)**

또한, 집의 매매 가격은 단순히 집의 평수가 크다고 결정되는 것이 아니라 집의 층의 수, 방의 개수, 지하철 역과의 거리와도 영향을 받음 → **다수의 변수로 인해 특정변수의 값이 영향을 받는 경우 (다중 선형 회귀 분석 사용)**

- 단순 선형 회귀 분석(Simple Linear Regression Analysis)

  $$y = Wx + b$$

  W: 가중치(weight), b : 편향(bias) → W,b의 값을 적절히 찾아내면, x와 y의 관계를 적절히 모델링 한 것.

- 다중 선형 회귀 분석(Multiple Linear Regression Analysis)

  $$y = W_1x_1 + W_2x_2 + ... + W_nx_n + b$$

가설(Hypothesis) 세우기

x와 y의 관계를 유추하기 위해, 수학적으로 식을 세움 → 머신러닝에서는 x와 y의 관계에 대한 수학적 식을 가설이라고 함. example : $H(x) = Wx + b$

비용 함수(Cost function) : 평균 오차 제곱(MSE)

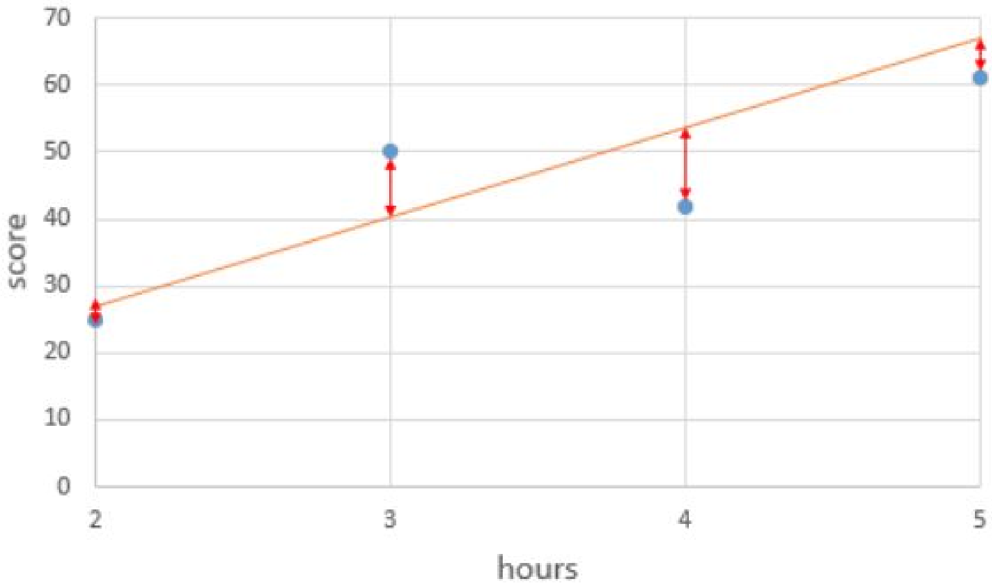
x와 y의 관계를 모델링하기 위해 적절한 W와 b의 값을 찾아야 함 → 이를 위해, 기계는 실제값과 가설로부터 얻은 예측값의 **오차를 계산하는 식(cost function)**을 세우고, 이 **식의 값을 최소화(Optimizer)**하는 최적의 W와 b를 찾아야함.

- 실제값과 예측값에 대한 오차를 나타내는 식을 **목적 함수(Objective function) 또는 비용 함수(Cost function) 또는 손실 함수(Loss function)**라고 함.
- cost function은 단순히 실제값과 예측값에 대한 오차를 표현하면 되는 것이 아니라, 예측값의 오차를 줄이는 일에 최적화된 식이어야함 → 회귀 문제의 경우, 대표적으로 평균 제곱 오차(Mean Squared Error, MSE)가 사용됨
- **평균 제곱 오차(Mean Squared Error, MSE)** :

  $$cost(W, b) = \frac{1}{n}\sum_{i=1}^{n}[y^{(i)} - H(x^{(i)})]^2$$

  n은 데이터의 갯수, $y^{(i)}$는 데이터 i의 실제값, $H(x^{(i)})$는 데이터 i의 예측값

- example



빨간 선은 오차를 의미

hours(x)	2	3	4	5
실제값	25	50	42	61
예측값	27	40	53	66
오차	-2	10	-7	-5

Squared Error = $(-2)^2 + 10^2 + (-7)^2 + (-5)^2 = 178$

Mean Squared Error = $178/4 = 44.5$

옵티마이저(Optimizer) : 경사하강법(Gradient Descent)

cost function을 최소화하는 매개 변수 W와 b를 찾기 위한 알고리즘 사용 → 옵티마이저 또는 최적화 알고리즘이라고 함

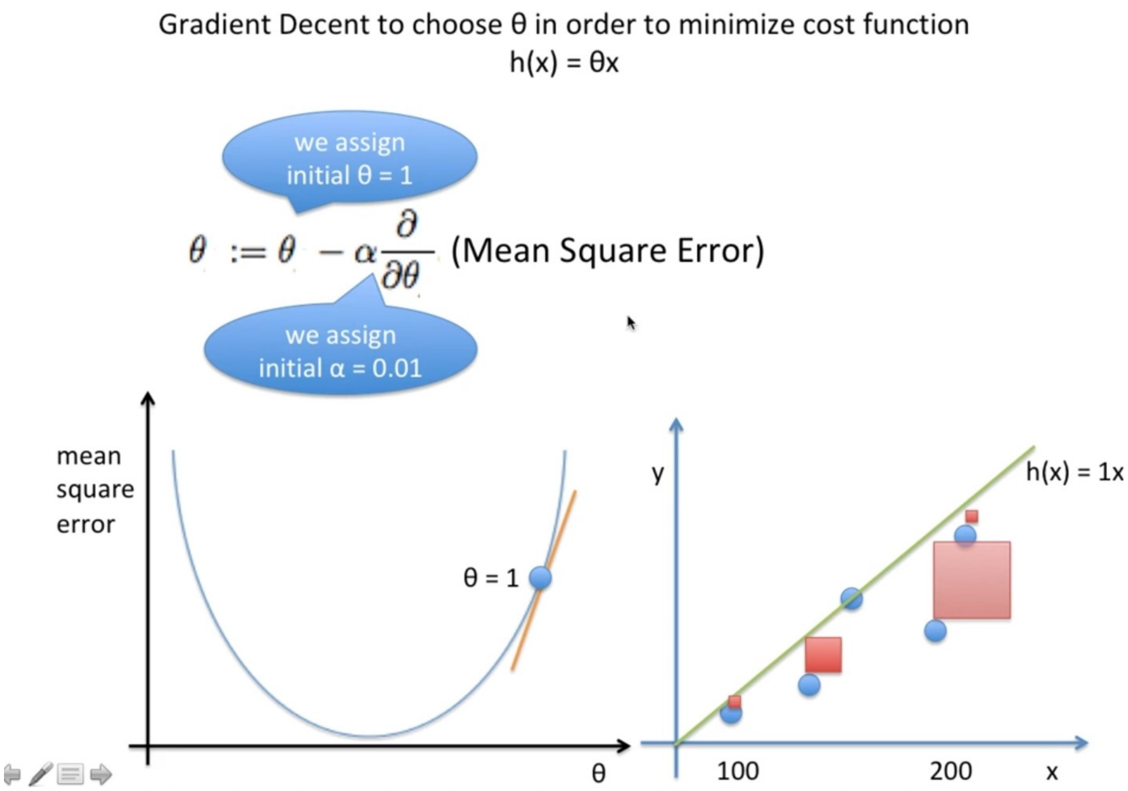
- optimizer를 통해 W와 b를 찾아내는 과정을 학습이라고 부름
- 옵티마이저 알고리즘은 대표적으로 경사 하강법(Gradient Descent)사용
- **경사 하강법(Gradient Descent)**

$$W = W - \alpha \frac{\partial}{\partial W} cost(W)$$

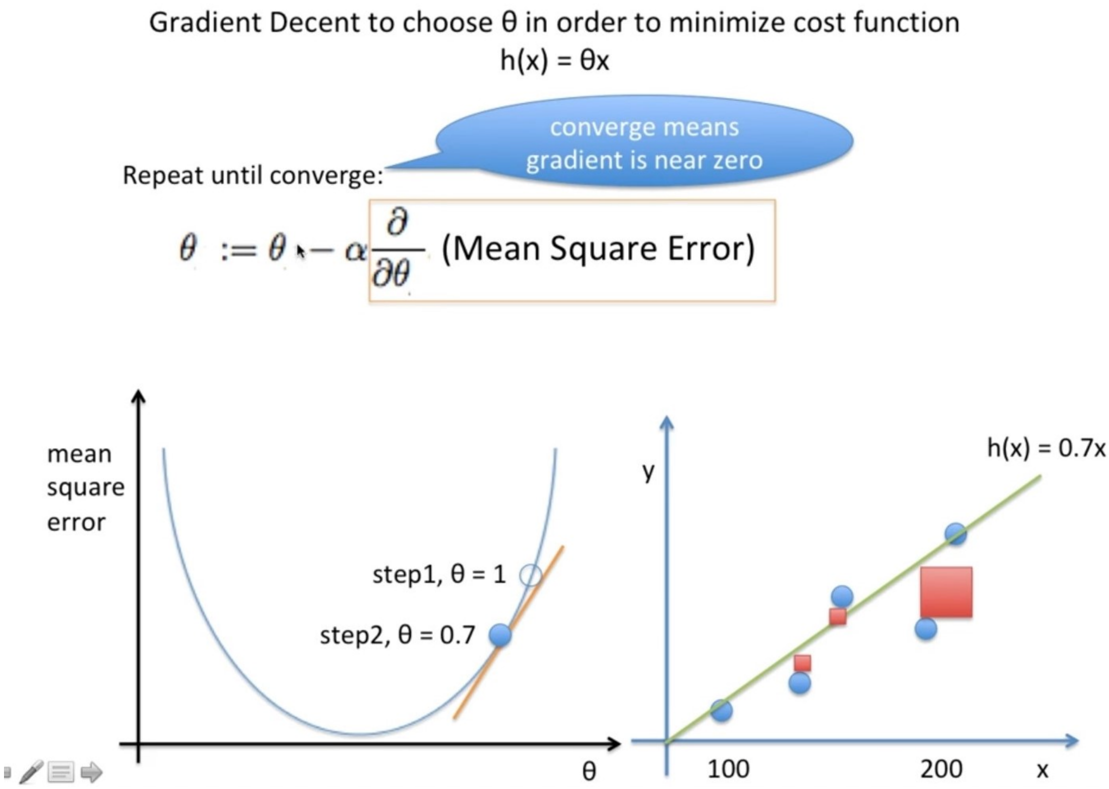
위 식을 $cost(W)$식의 접선의 기울기($\frac{\partial}{\partial W} cost(W)$)가 0이 될 때까지 반복

- 경사하강법 예제

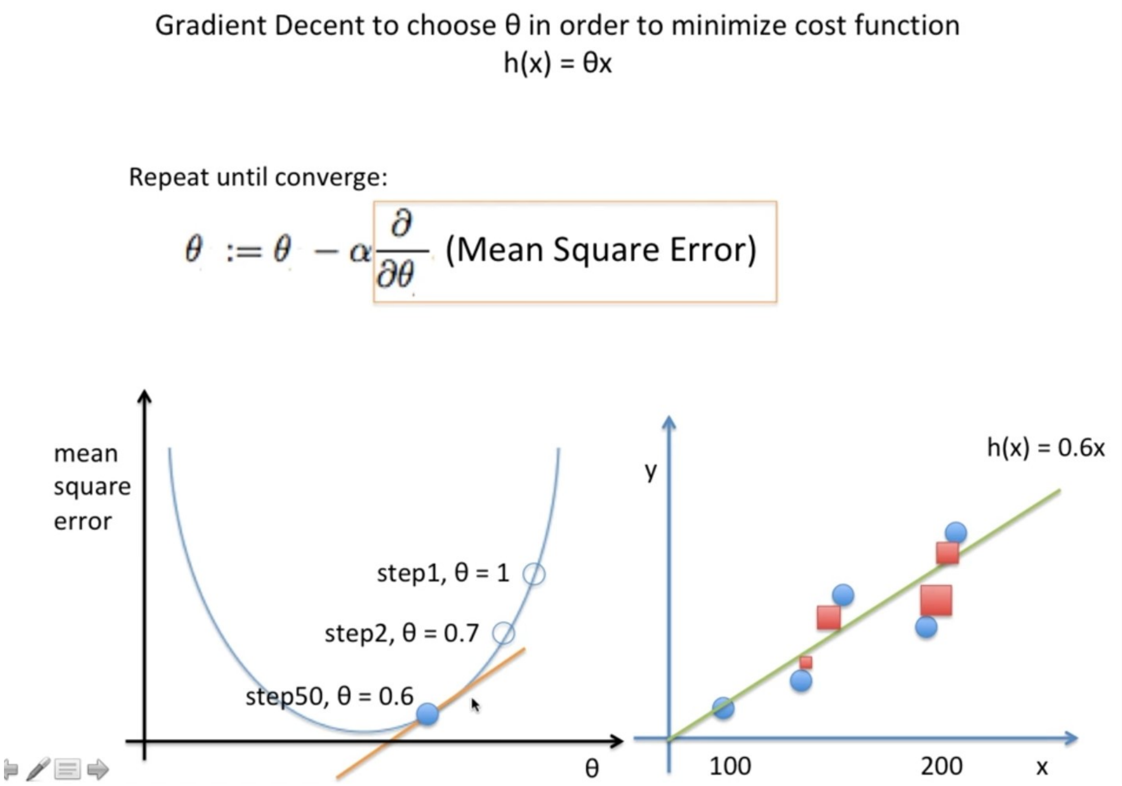
첫번째 step : 처음 θ는 1로 초기화 → 경사하강법을 통해 θ는 0.7로 업데이트 되었다고 가정



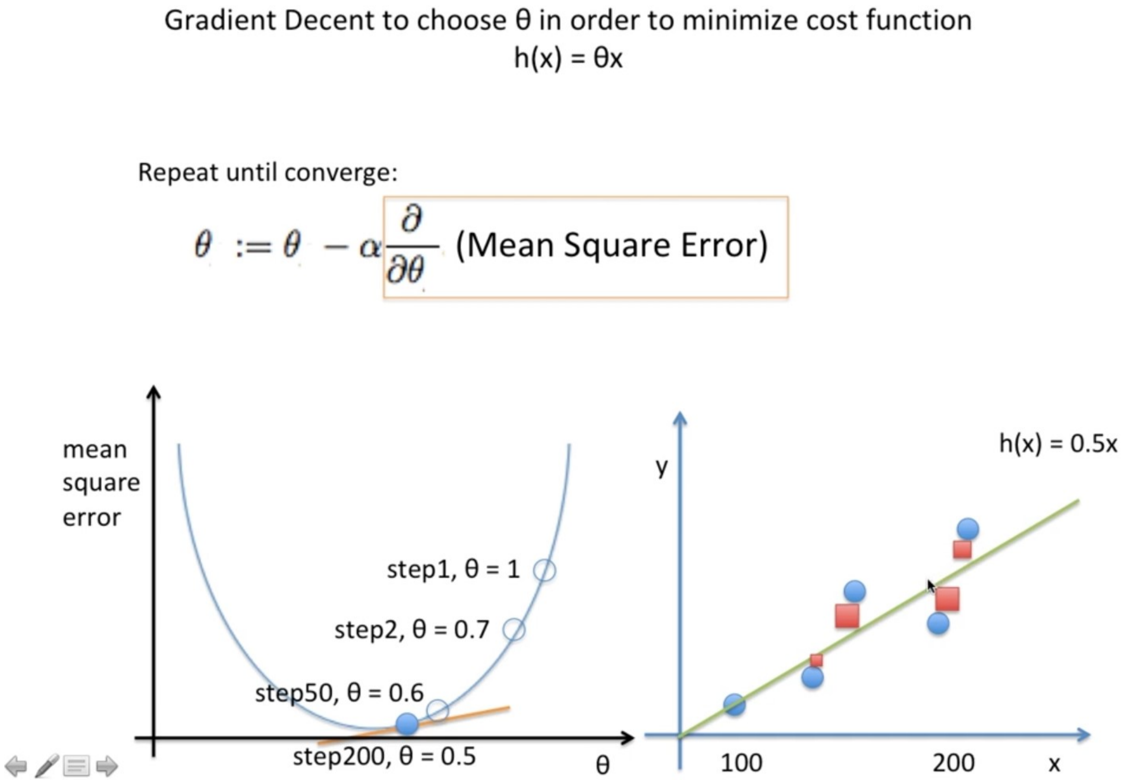
두번째 step : θ는 0.7→ 경사하강법을 통해 θ는 0.6로 업데이트 되었다고 가정



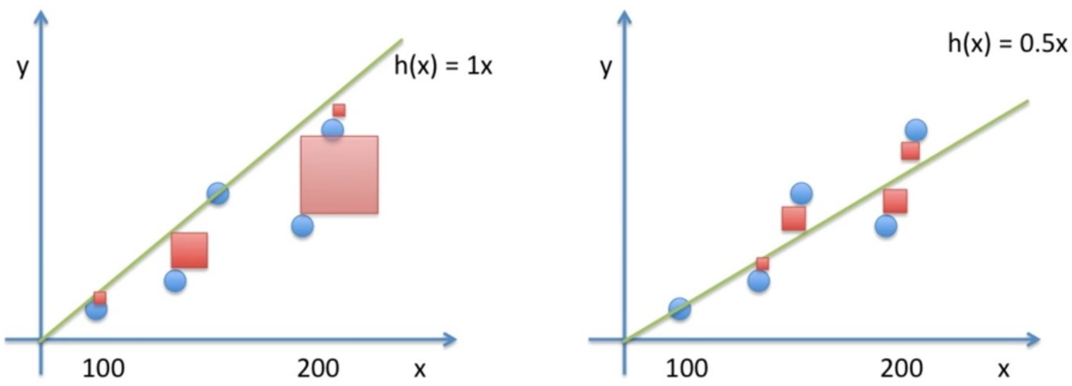
세번째 step : θ는 0.6→ 경사하강법을 통해 θ는 0.5로 업데이트 되었다고 가정



네번째 step : θ는 0.6→ 경사하강법을 통해 θ는 0.5로 업데이트 되었다고 가정, 이 이후의 cost function은 수렴한다고 보고 반복을 멈춘다.



결과 : 오른쪽 $h(x) = 1x$는 처음의 가설을 의미, $h(x) = 0.5x$는 optimizer를 통해 W 업데이트 과정을 거친 가설



출처

- https://wikidocs.net/21670
- https://www.youtube.com/watch?v=MwadQ74iE-k&list=PLVNY1HnUlO241gILgQloWAs0xrrkqQfKe&index=10