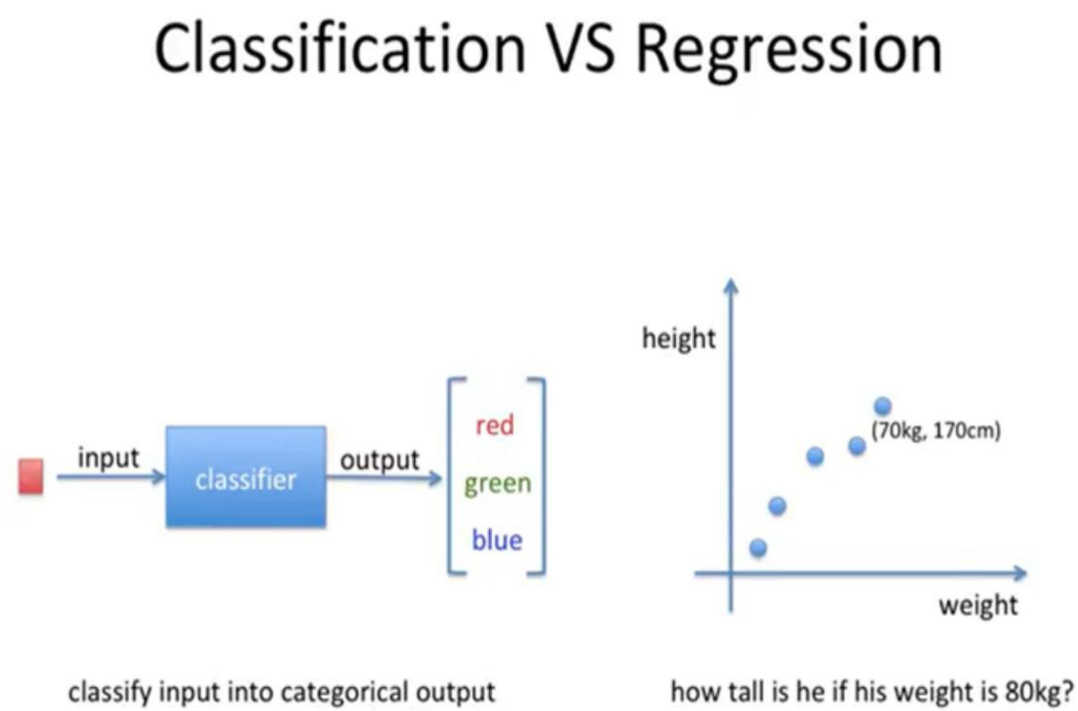
회귀와 분류 (regression and classification)

회귀(regresson)와 분류(classification)

- **회귀 모델**
  - 연속적인 값을 예측
    - **캘리포니아의 주택 가격이 얼마인가요?**
    - **사용자가 이 광고를 클릭할 확률이 얼마인가요?**
- **분류 모델**
  - 불연속적인 값을 예측
    - **주어진 이메일 메시지가 스팸인가요, 스팸이 아닌가요?**
    - **이 이미지가 강아지, 고양이 또는 햄스터의 이미지인가요?**



회귀의 어원

- **회귀 분석(regression analysis)**
  - 관찰된 연속형 변수들에 대해 두 변수 사이의 모형을 구한 뒤 적합도를 측정해 내는 분석 방법
  - 회귀분석은 시간에 따라 변화하는 데이터나 어떤 영향, 가설적 실험, 인과 관계의 모델링 등의 통계적 예측에 이용
- **회귀(영어: regress 리그레스[*])의 원래 의미**
  - 옛날 상태로 돌아가는 것을 의미
  - 영국의 유전학자 프랜시스 골턴은 "평균으로의 회귀(regression to the mean)"
    - **부모의 키와 아이들의 키 사이의 연관 관계를 연구하면서 부모와 자녀의 키 사이에는 선형적인 관계가 있고 키가 커지거나 작아지는 것보다는 전체 키 평균으로 돌아가려는 경향이 있다는 가설을 세웠으며 이를 분석하는 방법을 "회귀분석"이라고 함**
    - **이러한 경험적 연구 이후, 칼 피어슨은 아버지와 아들의 키를 조사한 결과를 바탕으로 함수 관계를 도출하여 회귀분석 이론을 수학적으로 정립**

선형 회귀
(linear regression)

선형 회귀와 로지스틱 회귀

- **단순 선형 회귀 분석(Simple Linear Regression Analysis)**
  - 입력: 특징이 하나
  - 출력: 하나의 값
    - **키로 몸무게 추정**

$$H(x) = Wx + b$$

- **다중 선형 회귀 분석(Multiple Linear Regression Analysis)**
  - 입력: 특징이 여러 개, 출력: 하나의 값
    - **역세권, 아파트 평수, 주소로 아파트값을 추정**

$$y = W_1x_1 + W_2x_2 + \ldots W_nx_n + b$$

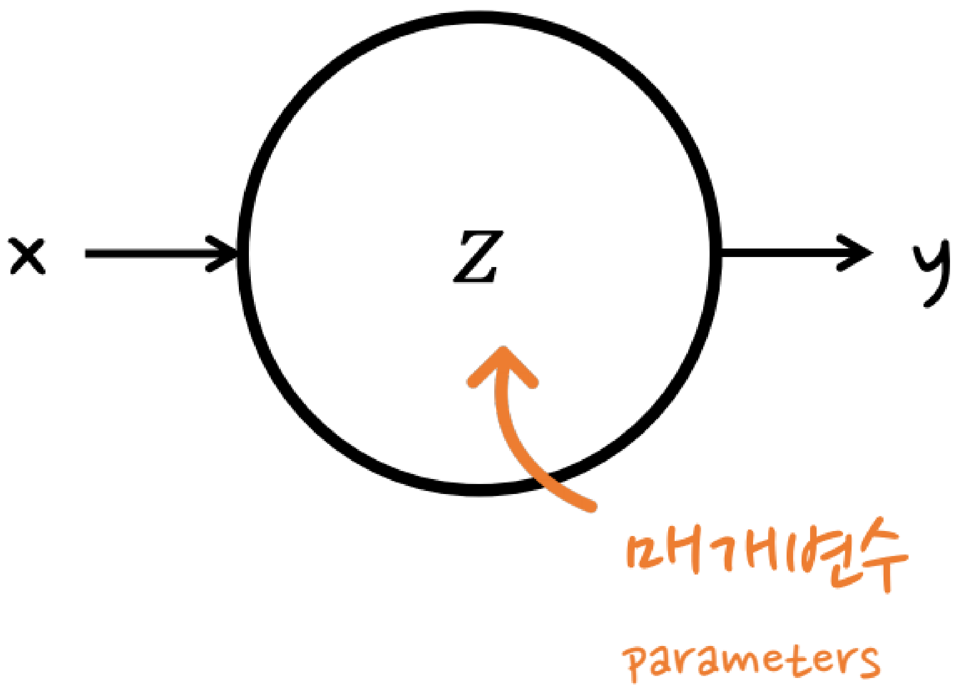
- **로지스틱 회귀(Logistic Regression)**
  - 이진 분류(Binary Classification)
  - 입력: 하나 또는 여러 개, 출력: 0 아니면 1
    - **타이타닉의 승객 정보로 죽음을 추정**

score(x)	result(y)
45	불합격
50	불합격
55	불합격
60	합격
65	합격
70	합격

인공지능이란? W와 b 구하기

- **다음 식에서 가중치 W와 편향 b를 구하기**
  - W와 b를 매개변수 함

$$H(x) = Wx + b$$



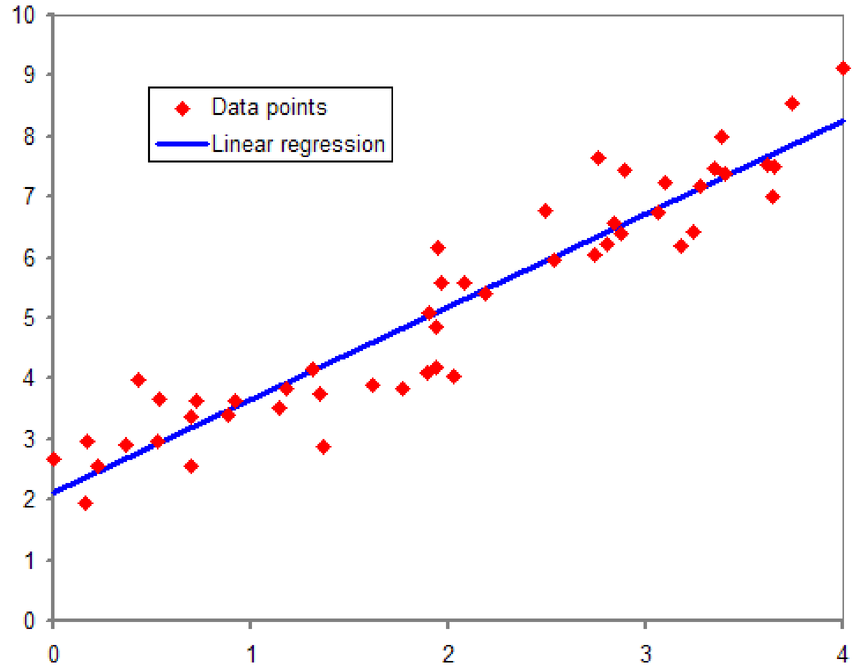
$$\theta = (w, b)$$

주요 용어 정리

- **가설(Hypothesis)**
  - 가중치(weight)와 편향(bias)
  - 기울기와 절편
- **손실 함수(Loss Function)**
  - MSE(Mean Square Error 평균제곱오차)
  - Categorical crossentropy
  - Sparse Categorical crossentropy
- **경사 하강법(Gradient Descent)**
  - 내리막 경사 따라 가기
- **학습률(learning rate)**
  - 대표적인 하이퍼패러미터

선형 회귀

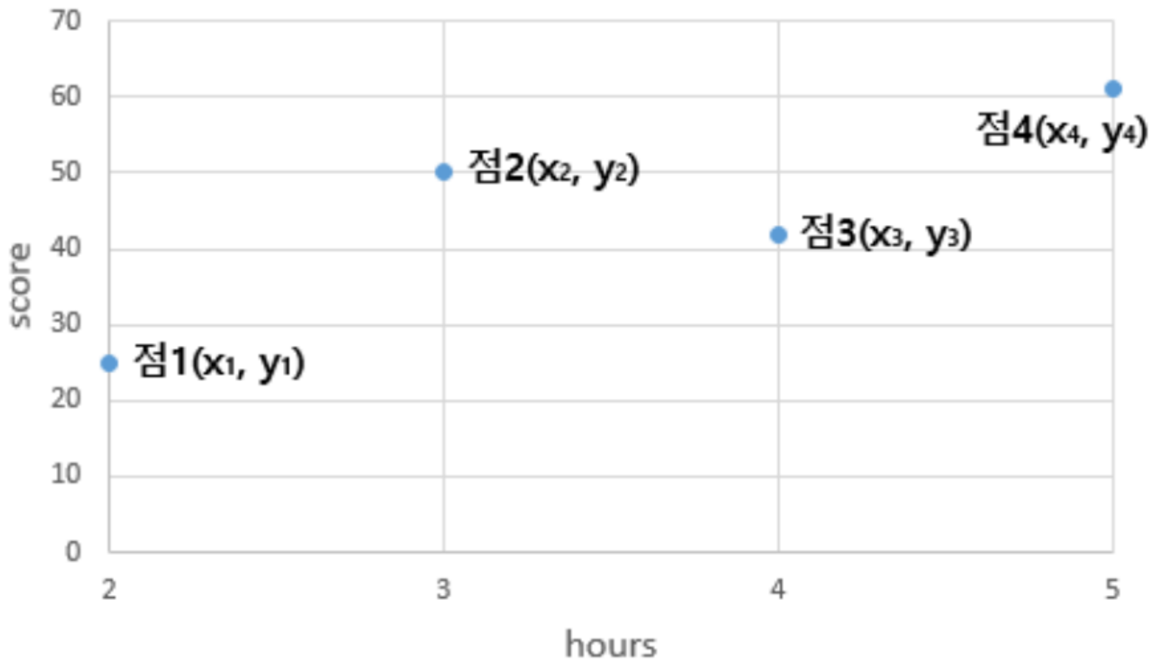
- **Linear regression**
  - 데이터의 경향성을 가장 잘 설명하는 하나의 직선을 예측하는 방법
    - **Y = aX + b**
      - **기울기 a와 절편인 b를 구하는 것**
  - 사례
    - **국어와 수학 성적**
    - **키와 몸무게**
    - **치킨과 맥주의 판매량**
    - **기저귀와 맥주의 판매량**



- **딥러닝 분야에서**
  - 선형 회귀
    - **Y = wX + b**
      - **가중치 w와 편향인 b를 구하는 것**

선형 회귀 문제 사례

- 공부 시간이 x라면, 점수는 y



hours(x)	score(y)
2	25
3	50
4	42
5	61

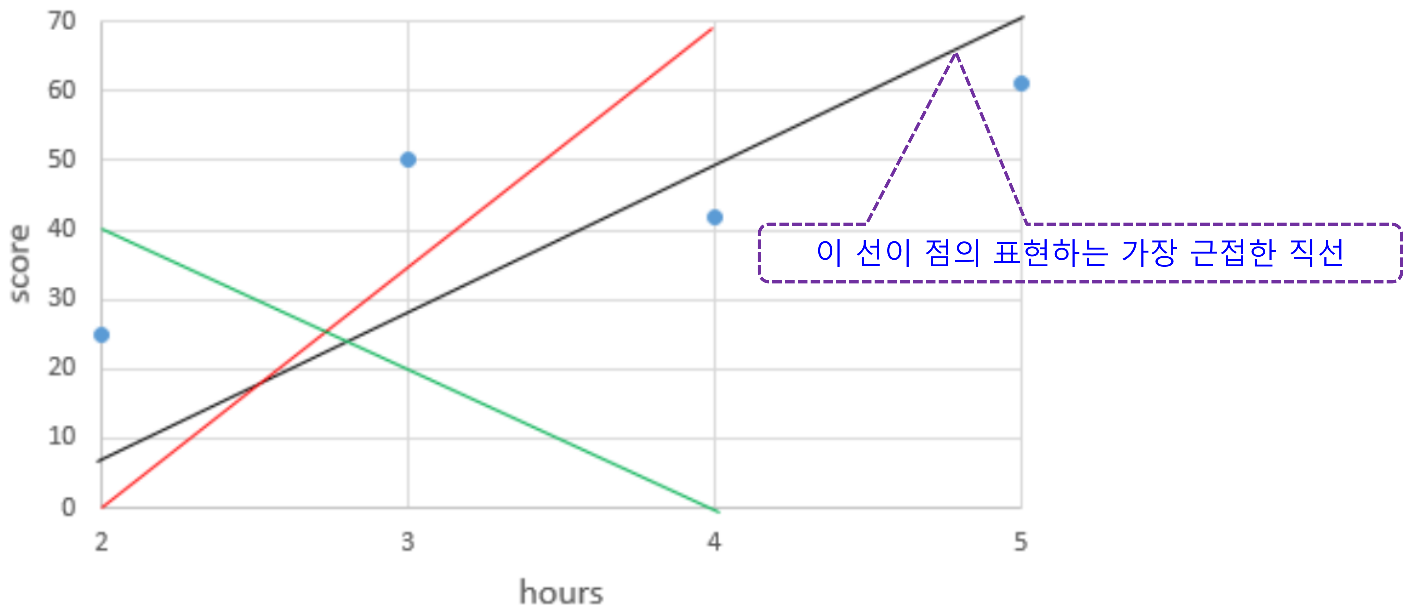
- 알려준 데이터로부터 x와 y의 관계를 유추
  - 학생이 6시간을 공부하였을 때의 성적
  - 그리고 7시간, 8시간을 공부하였을 때의 성적을 예측

가설

- **머신 러닝: y와 x간의 관계를 유추한 식을 가설(Hypothesis)**
  - H(x)에서 H는 Hypothesis를 의미

$$H(x) = Wx + b$$

W:기울기, 가중치
b: 절편, 편향



- **선형 회귀에서 해야할 일은 결국 적절한 W와 b를 찾아내는 일**
  - 딥러닝 알고리즘이 하는 것이 바로 적절한 W와 b를 찾아내는 일

손실 함수(Loss function)

- **머신 러닝은 W와 b를 찾기 위해서**
  - 손실 함수를 정의
    - **실제 값과 가설로부터 얻은 예측 값의 오차를 계산하는 식**
  - 손실 함수 값을 최소화하는 최적의 W와 b를 찾아내려고 노력
- **손실 함수(Loss function)**
  - 목적 함수(Objective function), 비용 함수(Cost function)라고도 부름
  - 실제 값과 예측 값에 대한 오차에 대한 식
    - **예측 값의 오차를 줄이는 일에 최적화 된 식**
  - 평균 제곱 오차(Mean Squared Error, MSE) 등을 사용

$$\frac{1}{n}\sum_{i}^{n}\left[y_i - H(x_i)\right]^2$$

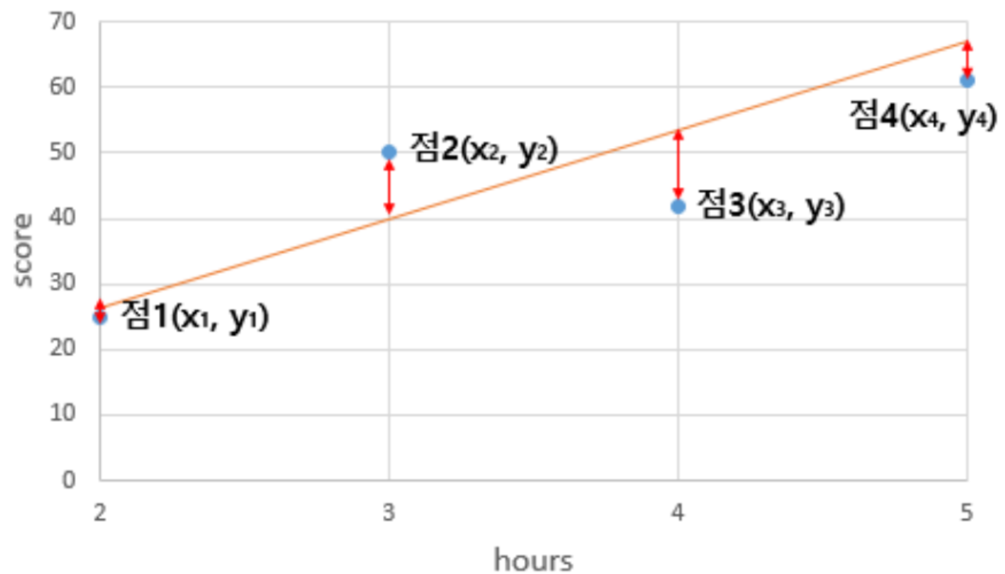
실제 값 예측 값

손실 함수: MSE

- **W 와 b의 값을 찾아내기 위해 오차의 크기를 측정할 방법이 필요**
  - W: 13 b: 1로 예측한다면 y=13x+1 직선이 예측한 함수로 예측 값을 추정

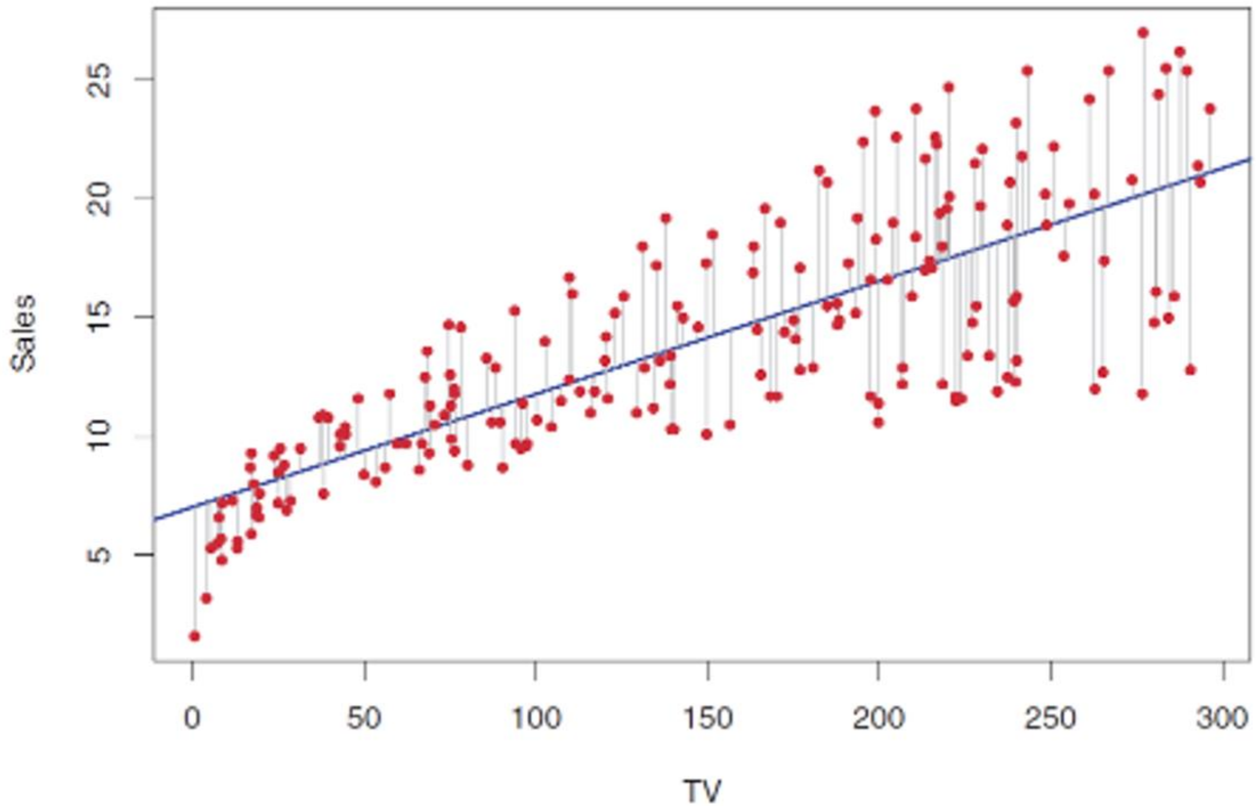
hours(x)	2	3	4	5
실제값	25	50	42	61
예측값	27	40	53	66
오차	-2	10	-7	-5

$$\frac{1}{n}\sum_{i}^{n}\left[y_i - H(x_i)\right]^2$$



손실 함수 MSE 이해

- **MSE**
  - 오차는 실제 데이터(빨간 점)와 예측 선(파란 선)의 차이의 제곱의 합



손실 함수를 W와 b의 함수로

- **평균 제곱 오차를 W와 b에 의한 비용 함수(Cost function)로 재정의**

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

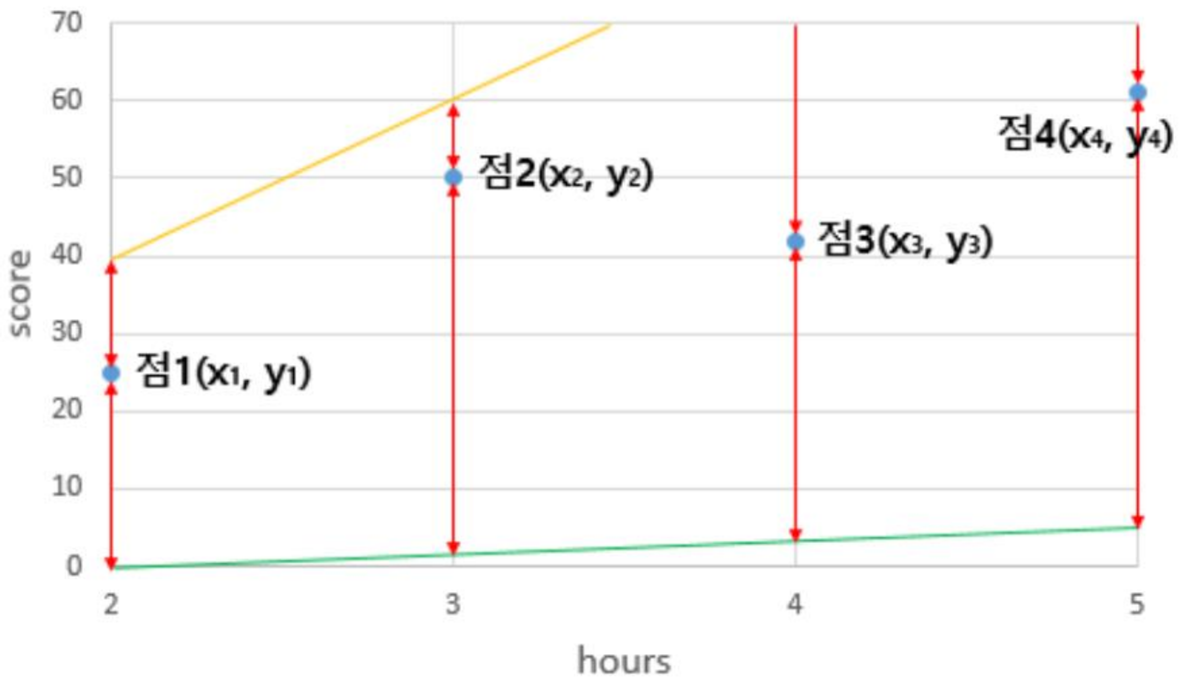
  - 모든 점 들과의 오차가 클수록 평균 제곱 오차는 커지며,
    - **오차가 작아질수록 평균 제곱 오차는 작아짐**

- **평균 제곱 오차**
  - cost(W, b)를 최소가 되게 만드는 W와 b를 구하면
    - **결과적으로 y와 x의 관계를 가장 잘 나타내는 직선을 그릴 수 있게 됨**

$$W, b \rightarrow minimize\ cost(W,b)$$

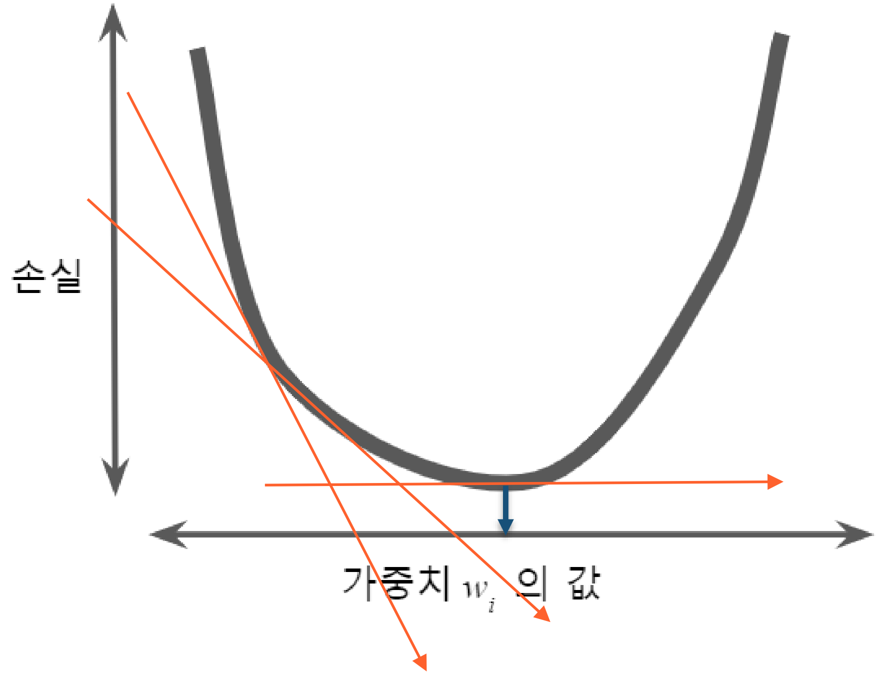
옵티마이저(Optimizer): 최적화 과정

- **머신 러닝에서 학습(training)**
  - 최적화 알고리즘(Optimizer algorithms)
  - 적절한 W와 b를 찾아내는 과정
    - **Gradient Descent(경사 하강법)**
      - **비용 함수(Cost Function)의 값을 최소로 하는 W와 b를 찾는 방법**
      - **경사 따라 내려 오기**



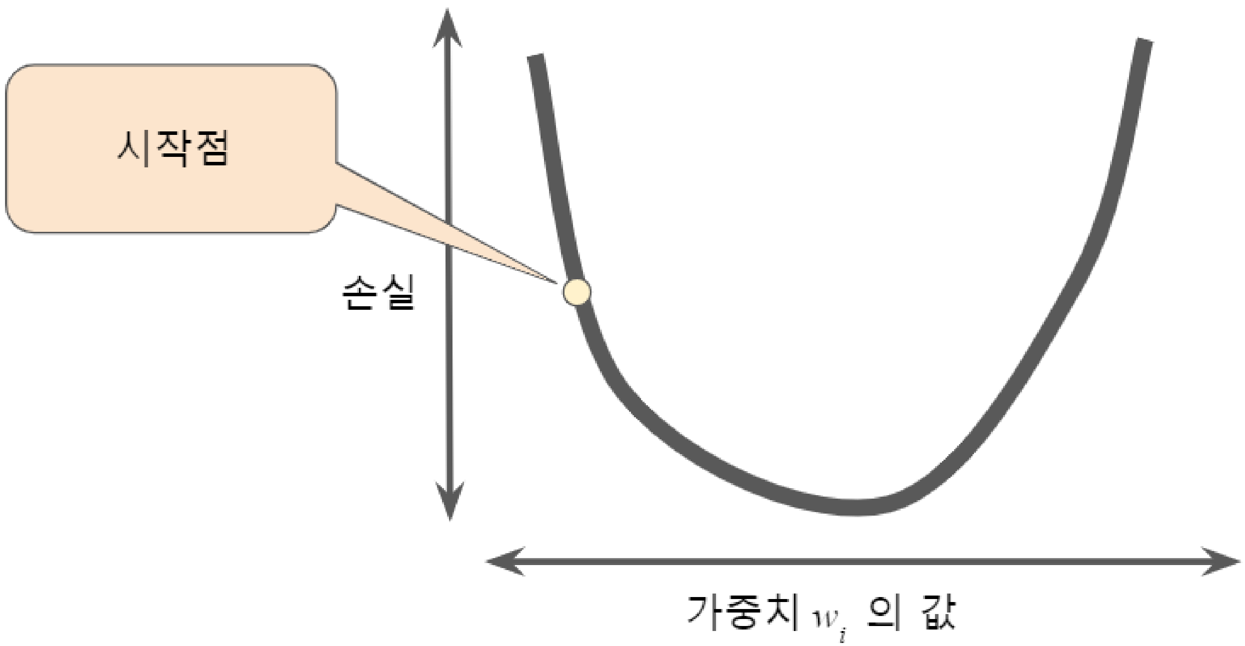
손실과 가중치

- **손실과 가중치 w_i을 대응한 그림**
  - 항상 볼록 함수 모양을 함
    - **도표가 다음과 같이 항상 그릇 모양으로 나타남**
- **볼록 문제에는 기울기가 정확하게 0인 지점인 최소값이 하나만 존재**
  - 이 최소값에서 손실 함수가 수렴
    - **결국 기울기를 구해야 함**



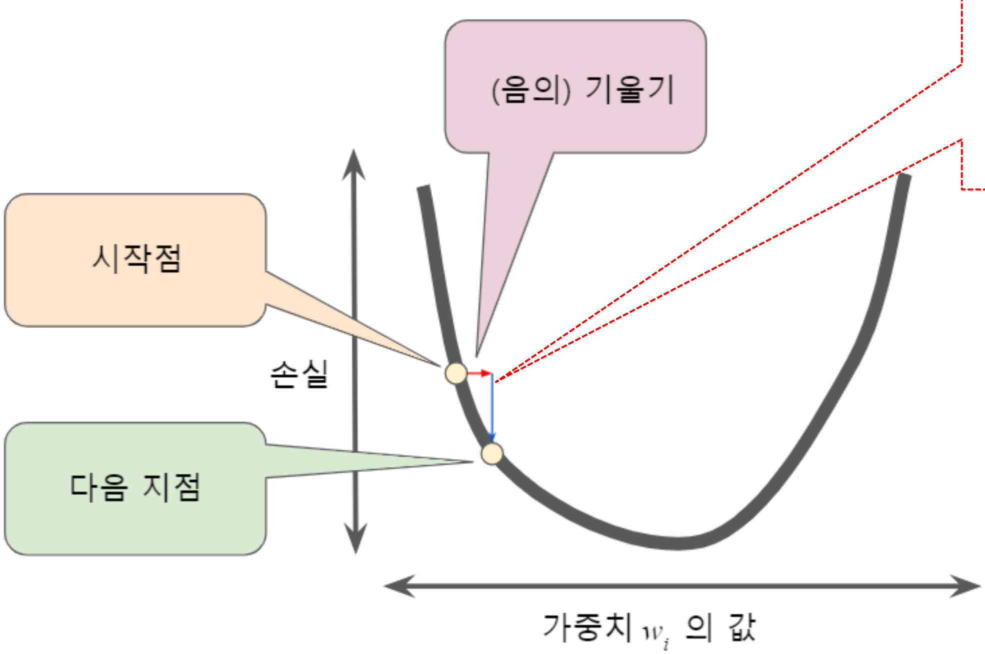
경사하강법

- **경사하강법의 첫 번째 단계**
  - 시작 값(시작점)을 선택
    - **시작점은 별로 중요하지 않음**
    - **따라서 많은 알고리즘에서는 0으로 설정하거나 임의의 값을 선택**
  - 시작점에서 손실 곡선의 기울기를 계산
    - **단일 가중치에 대한 손실의 기울기는 미분 값과 같음**



가중치의 조정

- **기울기가 0인 지점을 찾기 위해**
  - 기울기의 반대 방향으로 이동
    - **현재의 기울기가 음수이면**
      - **다음 가중치 값은 현재의 값보다 크게 조정**



가중치 값 W 조정

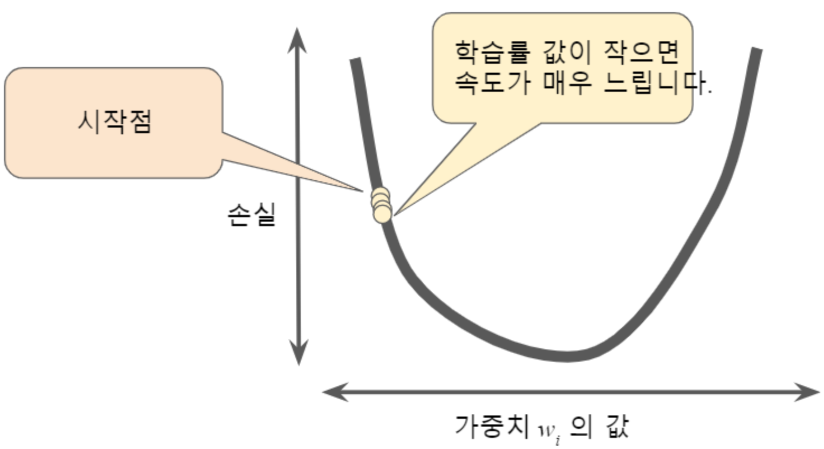
$$W := W - \alpha \boxed{\frac{\partial}{\partial W} cost(W)}$$

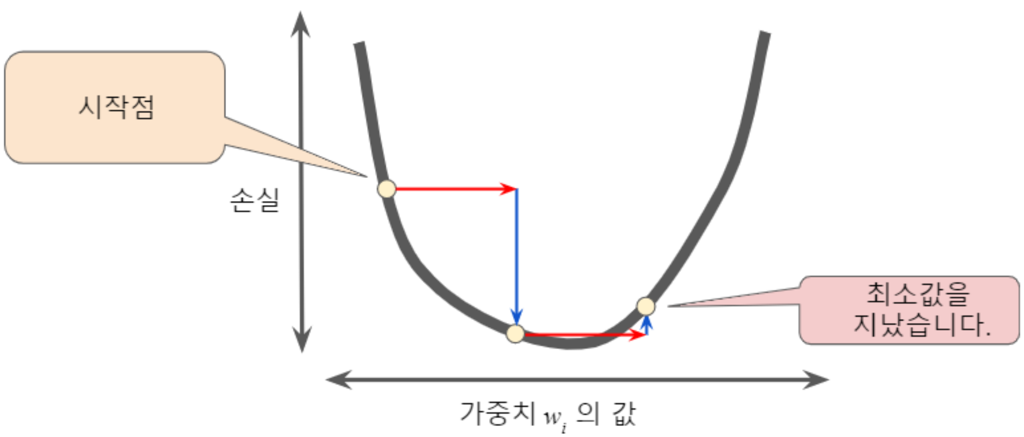
학습률(learning rate)

W 함수의 기울기: 미분 값

학습률

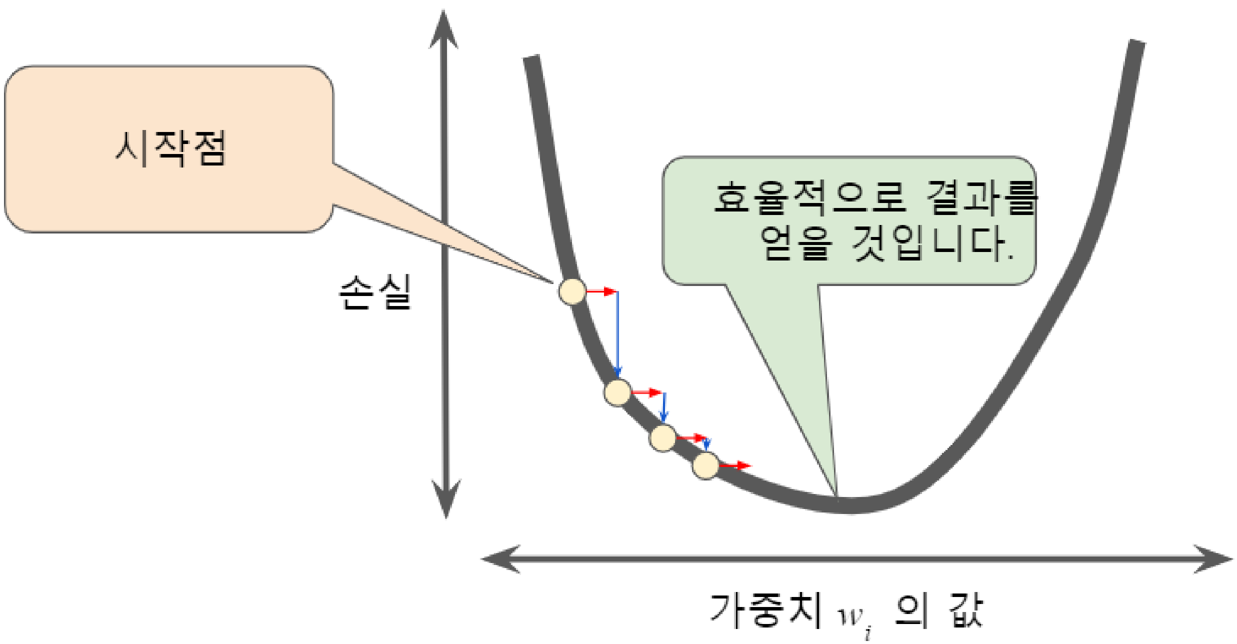
- **다음 가중치 값 결정 방법**
  - 기울기에 학습률(또는 보폭이라 불리는 스칼라)를 곱하여 다음 지점을 결정
    - **예를 들어 기울기가 -2.5이고 학습률이 0.01이면**
      - **w = w – (-2.5 x 0.01) = w + 0.025**
      - **경사하강법 알고리즘은 이전 지점으로부터 0.025 떨어진 지점을 다음 지점으로 결정**
- **학습률의 값**
  - 너무 작게 설정하면 학습 시간이 매우 오래 걸림
    - **반대로 학습률을 너무 크게 설정하면**
      - **다음 지점이 곡선의 최저점을 무질서하게 이탈할 우려가 있음**





적절한 학습률 설정

- **손실 함수의 기울기가 작다면 더 큰 학습률을 시도해 볼 수 있음**
  - 작은 기울기를 보완하고 더 큰 보폭을 만들어 냄

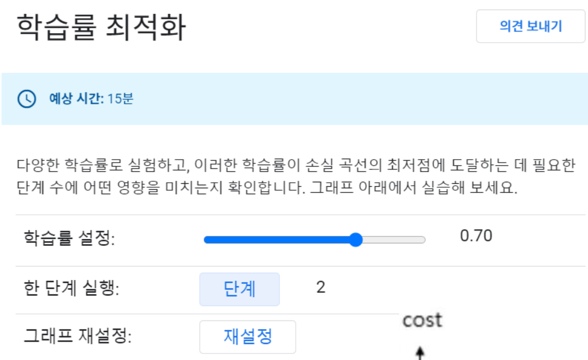


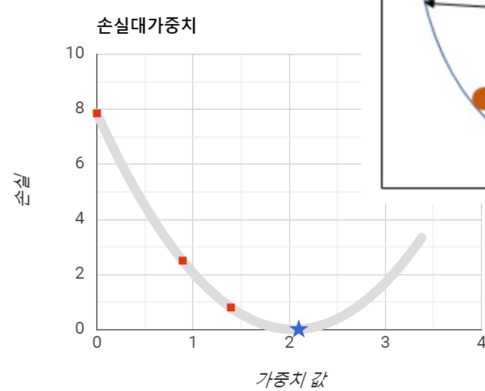
초매개변수와 학습률

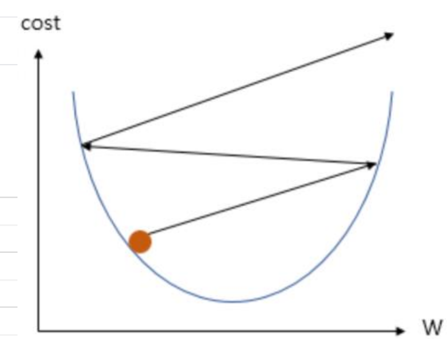
- **초매개변수(hyperparameter)**
  - 딥러닝에서 우리가 설정하는 값
    - **모델 학습을 연속적으로 실행하는 중에 개발자 본인에 의해 조작되는 '손잡이'**
  - 예를 들어 학습률은 초매개변수 중 하나
    - **매개변수와 대비되는 개념**

학습률 실험

- **다양한 학습률로 실험**
  - 이러한 학습률이 손실 곡선의 최저점에 도달하는 데 필요한 단계 수에 어떤 영향을 미치는지 확인
  - https://developers.google.com/machine-learning/crash-course/fitter/graph?hl=ko
  - 머신러닝 단기집중과정 메뉴
    - **손실 줄이기 | 학습률 최적화**
- **학습률 α**
  - W의 값을 변경할 때
    - **얼마나 크게 변경할지를 결정**
  - 얼마나 큰 폭으로 이동할 지를 결정
    - **학습률 α의 값을 무작정 크게 하면**
      - **W의 값이 발산하는 상황**
    - **학습률 α가 지나치게 낮은 값을 가지면**
      - **학습 속도가 느려지므로 적당한 α의 값을 찾아내는 것도 중요**
- **0.001에서 0.1 정도 사용**

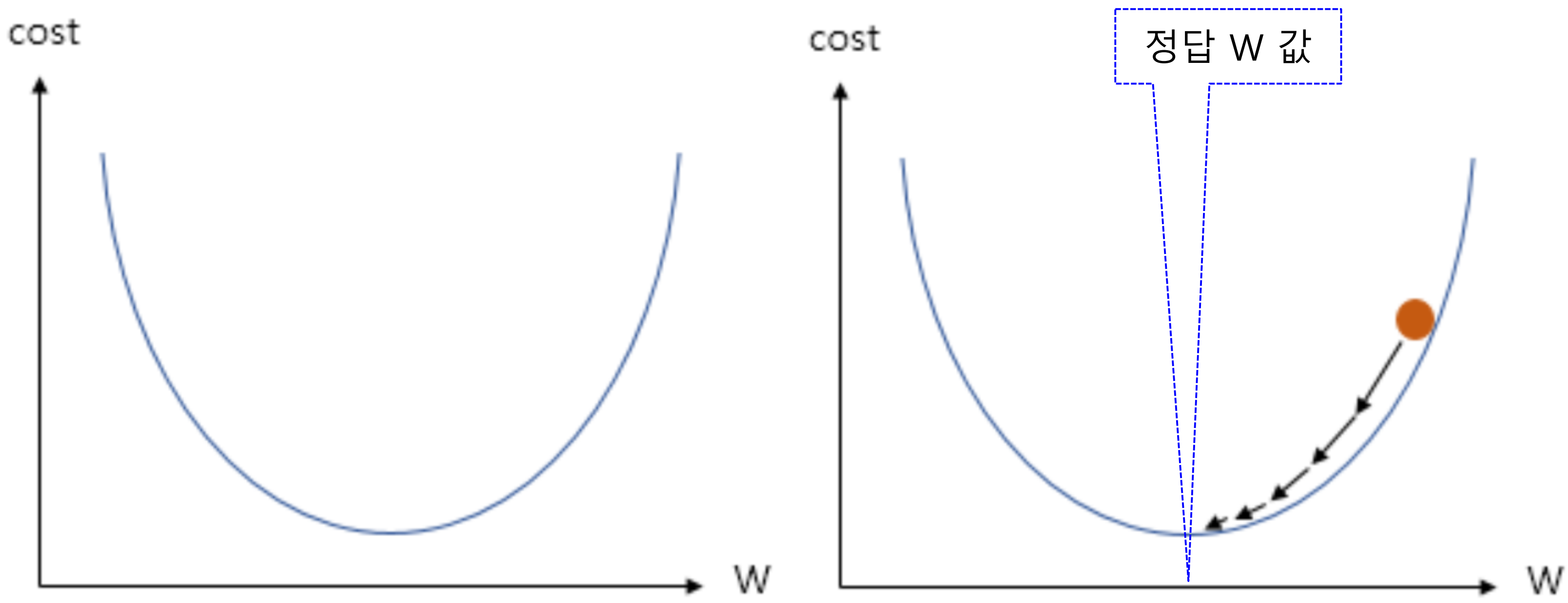






cost가 가장 최소값을 가지게 하는 W를 찾는 일

- **y = Wx라는 가설 H(x)**
  - 비용 함수의 값 cost(W)
    - **설명의 편의를 위해 편향 b가 없이 단순히 가중치 W만을 사용**



비용 함수와 최적의 W 구하기

- **비용 함수(Cost function)**

$$cost(W) = \frac{1}{n}\sum_{i}^{n}[y_i - H(x_i)]^2$$

- **Cost를 최소화하는 W를 구하기 위한 식**
  - 해당 식은 접선의 기울기가 0이 될 때까지 반복

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

학습률(learning rate)

W 함수의 기울기:
미분 값

  - 현재 W에서의 접선의 기울기와 α와 곱한 값을 현재 W에서 빼서 새로운 W의 값으로 다음 손실을 계산
  - 학습률(알파): 기울기가 최소인 다음 w로 가기 위한 비율

계산 과정의 의미

- **현재 W에서 현재 W에서의 접선의 기울기를 빼는 행위의 의미**
  - 접선의 기울기가 음수일 때

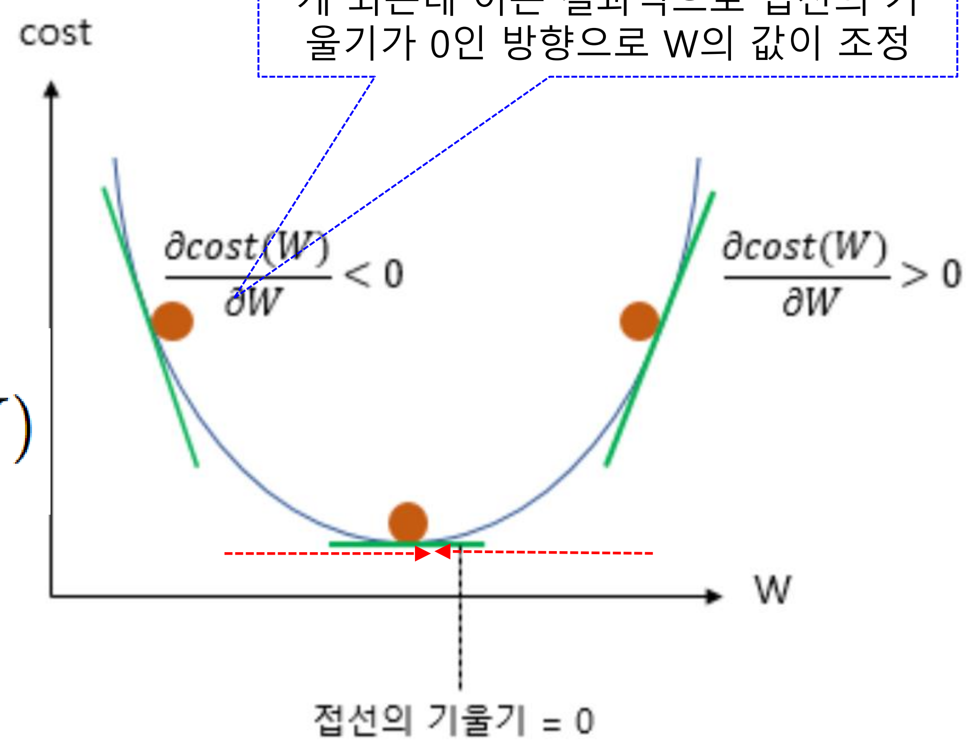
$$W := W - \alpha(\text{음수}) = W + \alpha(\text{양수})$$

  - 접선의 기울기가 양수일 때

$$W := W - \alpha(\text{양수})$$

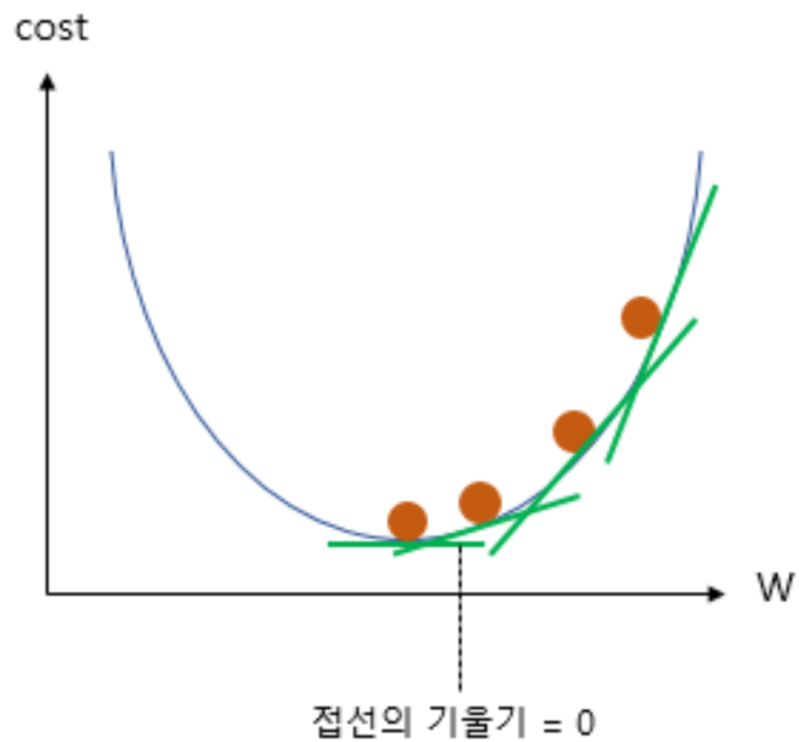
$$W := W - \alpha \frac{\partial}{\partial W} cost(W)$$

즉, 기울기가 음수면 W의 값이 증가하게 되는데 이는 결과적으로 접선의 기울기가 0인 방향으로 W의 값이 조정

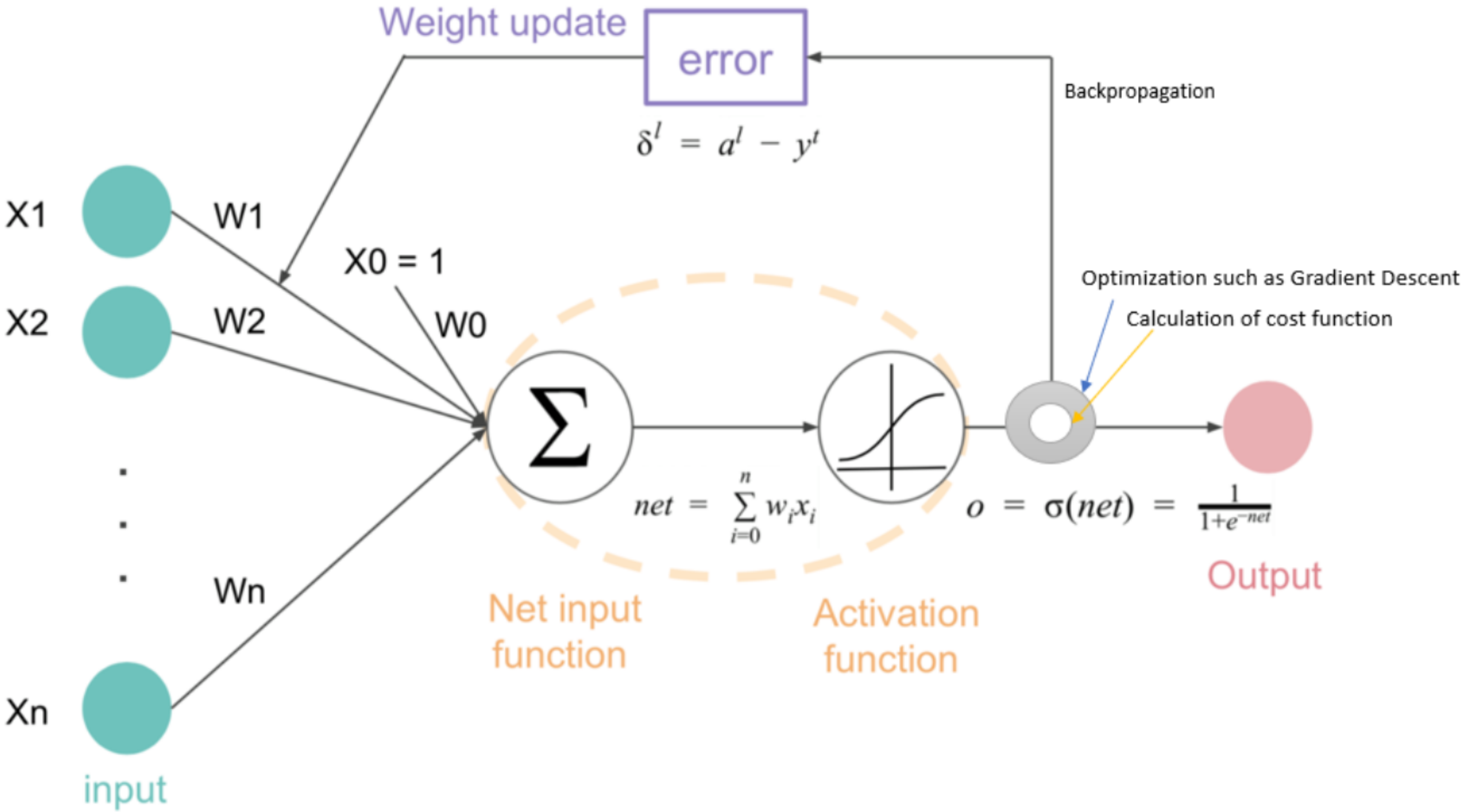


경사 하강법 정리

- **경사 하강법(Gradient Descent)**
  - 내리막 경사 따라 가기
  - 접선의 기울기
    - **맨 아래의 볼록한 부분에서는 결국 접선의 기울기가 0**
  - cost가 최소화가 되는 지점은 접선의 기울기가 0이 되는 지점
    - **또한 미분값이 0이 되는 지점**
  - 경사 하강법의 아이디어
    - **비용 함수(Cost function)를 미분하여 현재 W에서의 접선의 기울기를 구하고**
    - **접선의 기울기가 낮은 방향으로 W의 값을 변경하고 다시 미분하고**
    - **이 과정을 접선의 기울기가 0인 곳을 향해 W의 값을 변경하는 작업을 반복하는 것**

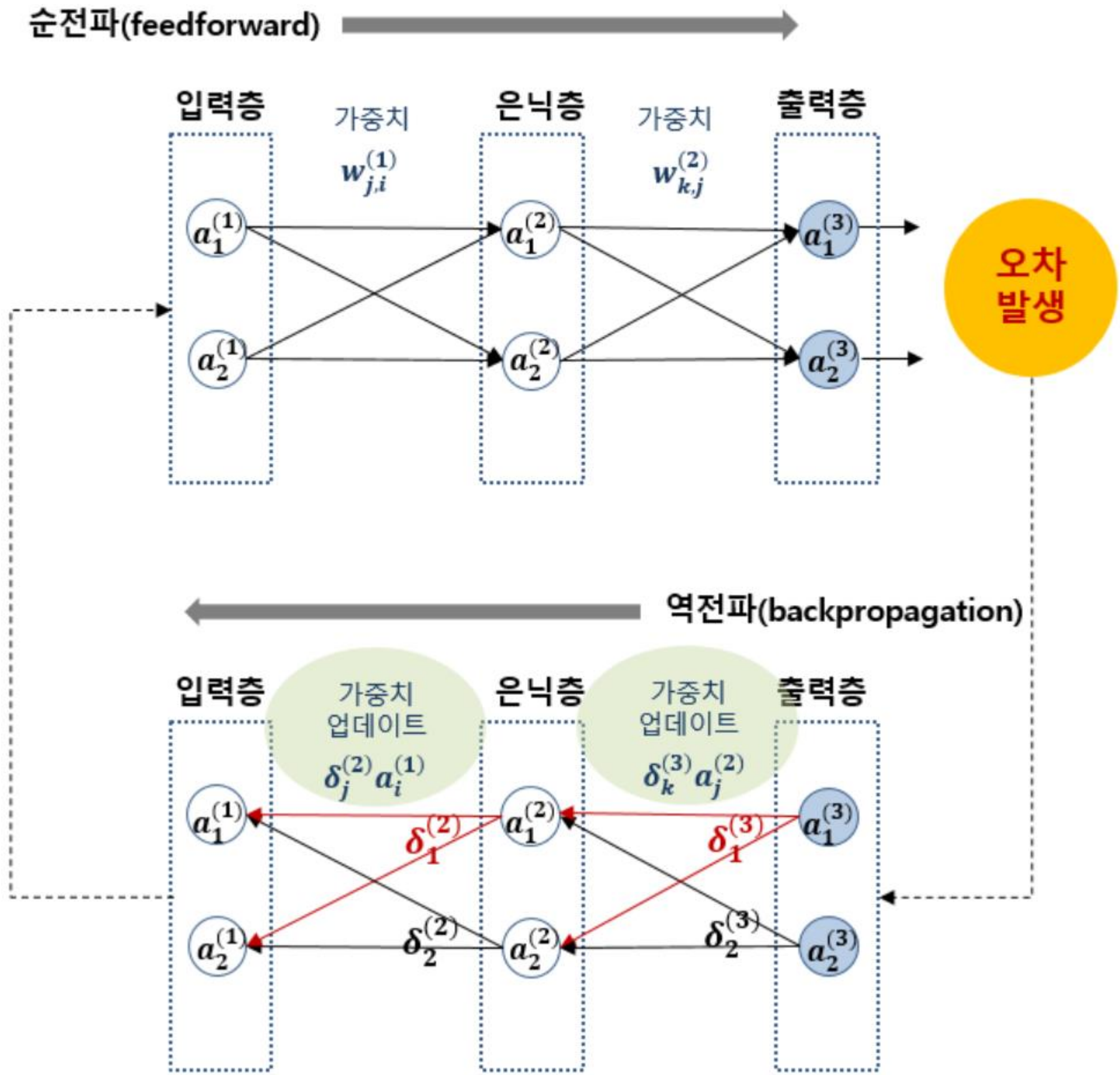


손실 함수를 최소로 하는 W와 b 구하는 과정



오차역전파

- **순전파와 역전파**
  - 순전파, 역전차, 가중치 수정, 순전파 ...를 계속 반복해 나가면서 오차 값이 적어지도록
- **1986년 제프리 힌튼이 적용**
  - 엄청난 처리 속도의 증가



https://blog.naver.com/samsjang/221033626685