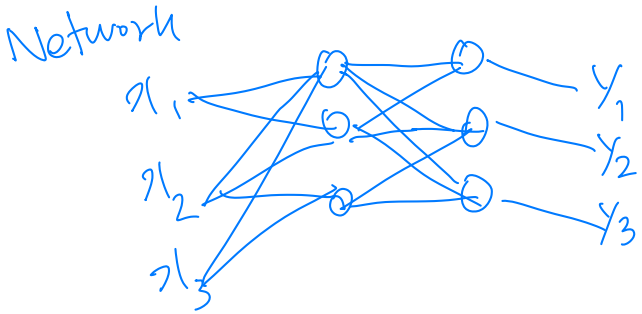
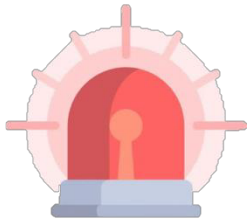


DNN

Logistic Regression

Regression with Gradient Descent Method



Complex: Differentiation, Derivative, and Optimization

One of difficult parts in this class, but
it will be a foundation of Deep Neural Networks!!

Regression

- **Solve the following problem**

$$Given\ Data = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \cdots, (\mathbf{x}_n, y_n)\}$$

Model you choose $f(\mathbf{x}; \mathbf{w})$

Find w to minimizes E (Error)

$$E(\mathbf{w}) = \sum_{(\mathbf{x},y)\in Data} \left(y - f(\mathbf{x}; \mathbf{w})\right)^2$$

Regression is an Optimization problem !!

Linear Additive Model

- **E is a quadratic function of w**

Find $w_1, w_2, \ldots, w_m$ to minimize

$$E(w_1, w_2, \ldots, w_m) = \sum_{(\mathbf{x},y)\in Data} \left(y - f(\mathbf{x}; w_1, w_2, \ldots, w_m)\right)^2$$

f is Linear fun of w

  - **It is a quadratic function optimization. <- Easy**

$$\frac{\partial E}{\partial w_1} = 0$$
$$\frac{\partial E}{\partial w_2} = 0$$
$$\ldots$$
$$\frac{\partial E}{\partial w_m} = 0$$

Solve the linear equations. Then we have the exact solution !!

But..

$E = -2(y - w - e^{wx} - \sin wx)(we^{wx} + w\cos x) = 0$ ∴ 복잡해 보임..

$f(w)$

↳ $\frac{dE}{dw} = 0$

- **What if E is not quadratic nor a polynomial of w**

$$f(w_0, w_1, w_2) = w_0 + e^{w_1 x_1} + \sin w_2 x_2$$

or $w_0 + w_1 e^{x} + w_2 \sin x$

$$\frac{\partial E}{\partial w_1} = 0$$
$$\frac{\partial E}{\partial w_2} = 0$$
$$\cdots$$
$$\frac{\partial E}{\partial w_m} = 0$$

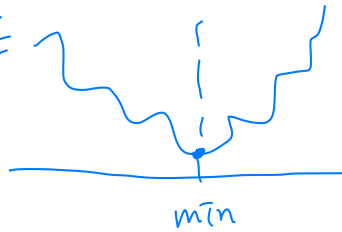
Can we solve the equations?

일반적으로, 찾기 어렵다 (계산이 복잡해서)

- **Is there a solution?** **Yes!**

값을 알긴 어렵지만 분명히 존재함

$$E(w_1, w_2, \ldots, w_m) = \sum_{(\mathbf{x},y)\in Data} \left(y - f(\mathbf{x}; w_1, w_2, \ldots, w_m)\right)^2$$



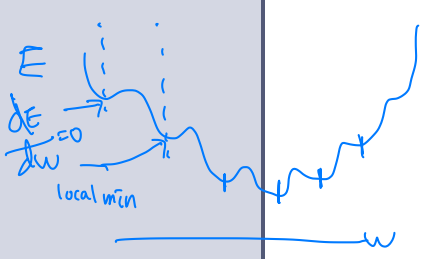
- **Can we find the solution?** **We canNOT say "Yes"**
- **What shall we do? :(**

General Approach for Optimization

- Because the original problem may not be solved

Find w that (globally) minimizes

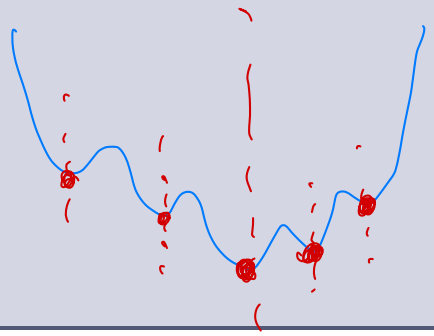
$$E(\mathbf{w}) = \sum_{(\mathbf{x},y)\in Data} \left(y - f(\mathbf{x};\mathbf{w})\right)^2$$



- We change the problem as follows:

Find w that locally minimizes

$$E(\mathbf{w}) = \sum_{(\mathbf{x},y)\in Data} \left(y - f(\mathbf{x};\mathbf{w})\right)^2$$



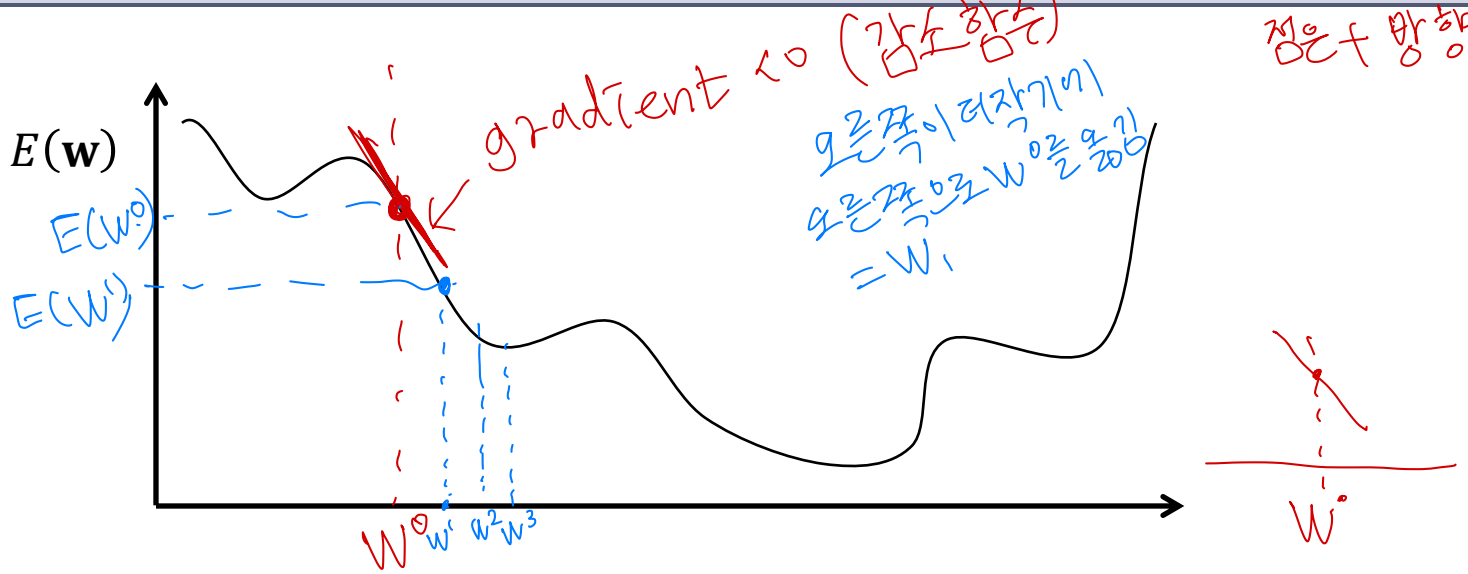
General Approach for Optimization

- **Local Minimization instead of Global Minimization**

Find w that locally minimizes

$$E(\mathbf{w}) = \sum_{(\mathbf{x},y)\in Data} \left(y - f(\mathbf{x};\mathbf{w})\right)^2$$

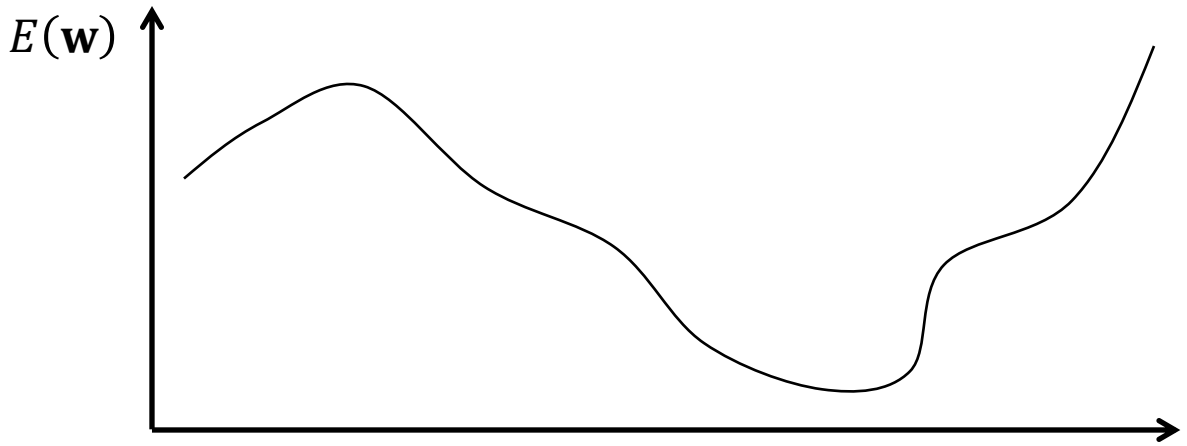
repeat
① Random W^0
② $\frac{\partial E}{\partial w}$ at W^0
③ move opposite direction
(기울기 – 면 접선 + 방향이동)



General Approach for Optimization

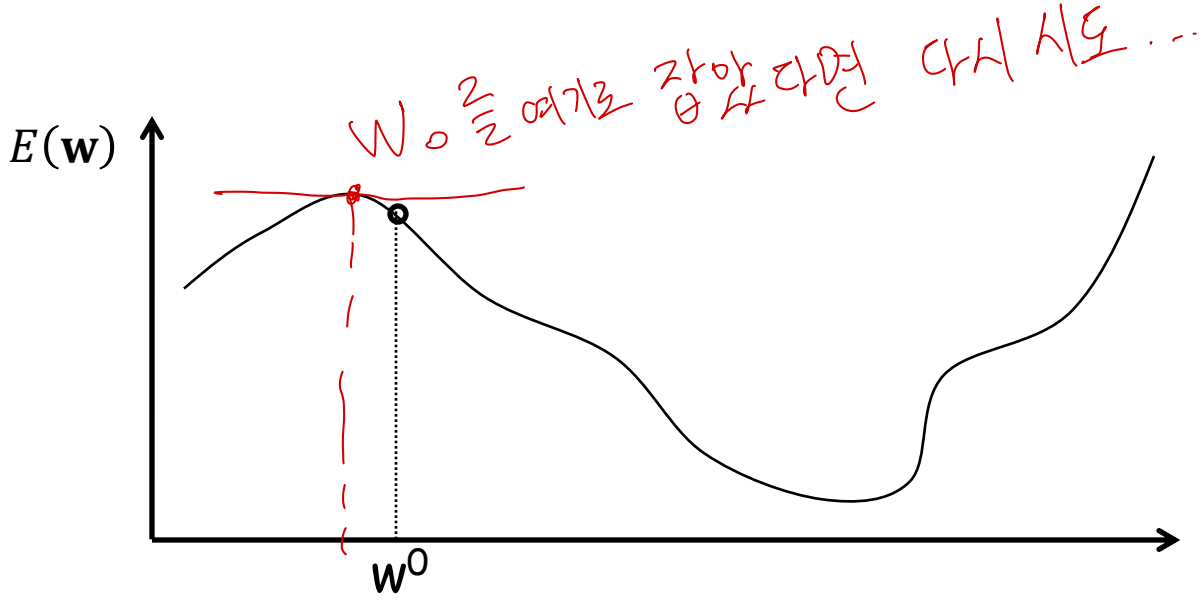
- **OK, but.. Is it easy to find a local minimum? Yes!!**

But, how?



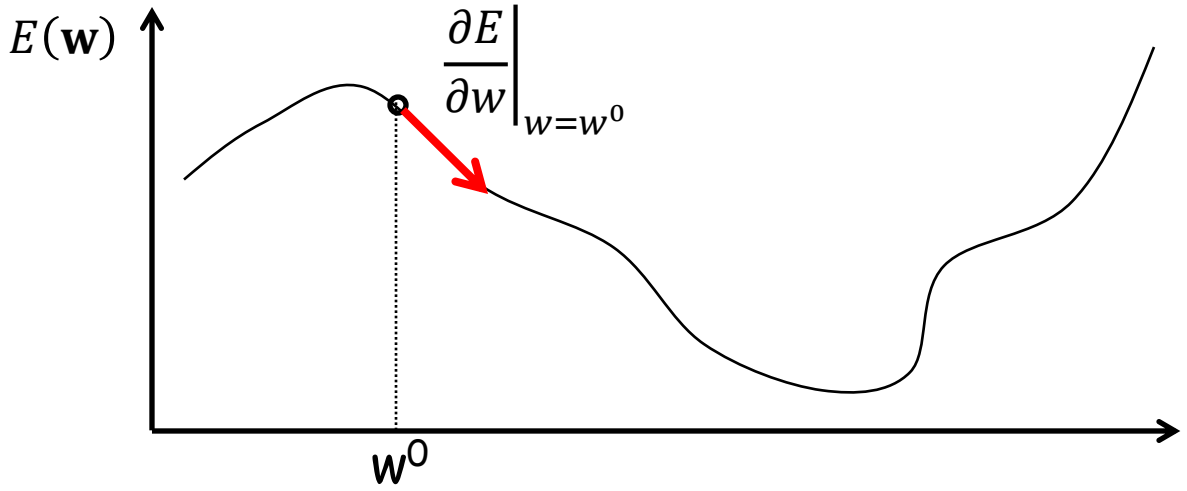
Gradient Descent Method

- 1. Randomly choose an initial point



Gradient Descent Method

- **2. Evaluate the gradient**
  - E(w) is differentiable -> We can obtain the gradient

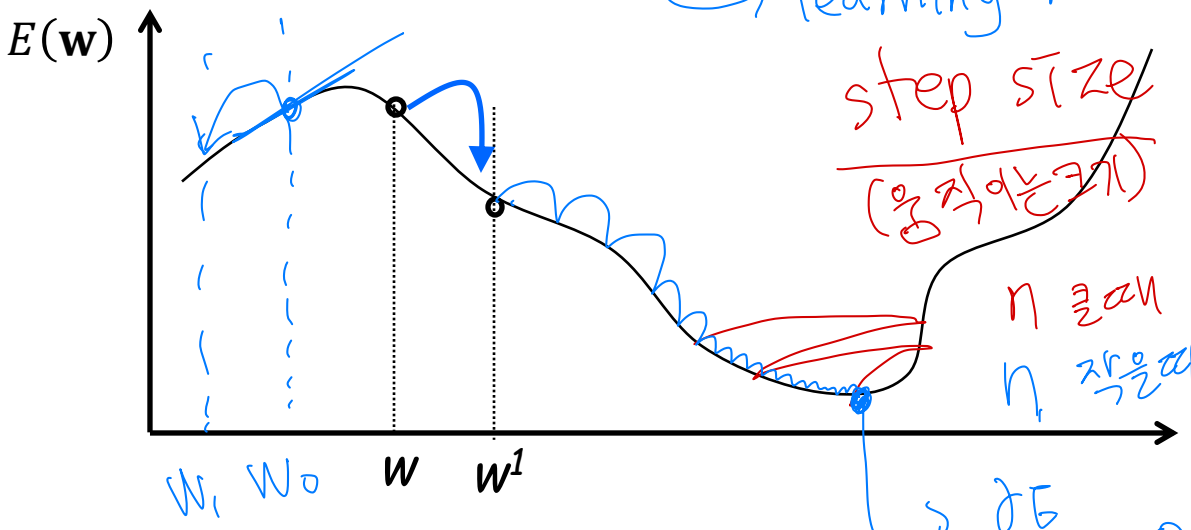


Gradient Descent Method

- 3. Move downward
  - Make a small jump

Why to jump reverse direction of the gradient?
- It's minimization

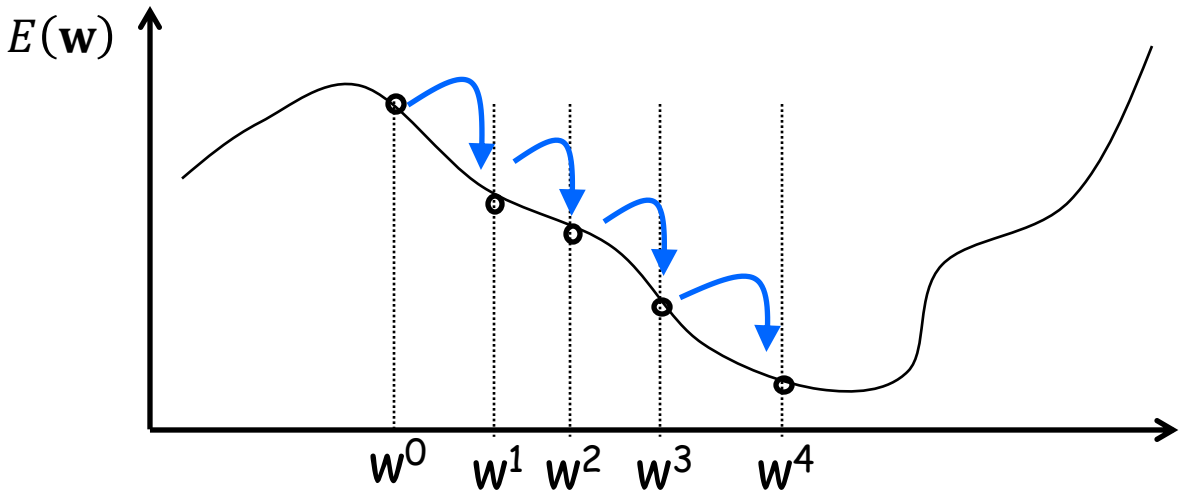
$$w^1 = w^0 - \eta \left.\frac{\partial E}{\partial w}\right|_{w=w^0}$$



Gradient Descent Method

- 4. Repeat the steps until the gradient is zero

$$w^{t+1} = w^t - \eta \left.\frac{\partial E}{\partial w}\right|_{w=w^t}$$



Gradient Descent Method

- **Steps**

learning rate (hyperparameter)

Randomly choose an initial solution, w^0

$t = 0$

Repeat

$$w^{t+1} = w^t - \eta \left. \frac{dE}{dw} \right|_{w=w^t}$$

$t = t + 1$

Until stopping condition is satisfied

① ≈0 : 학습 안 됨 (점프를 조금씩 하게끼 정확한 시기를 알게됨)

- ① gradient is very small
- ② fixed number of iterations

Gradient Descent Method

- **Steps**

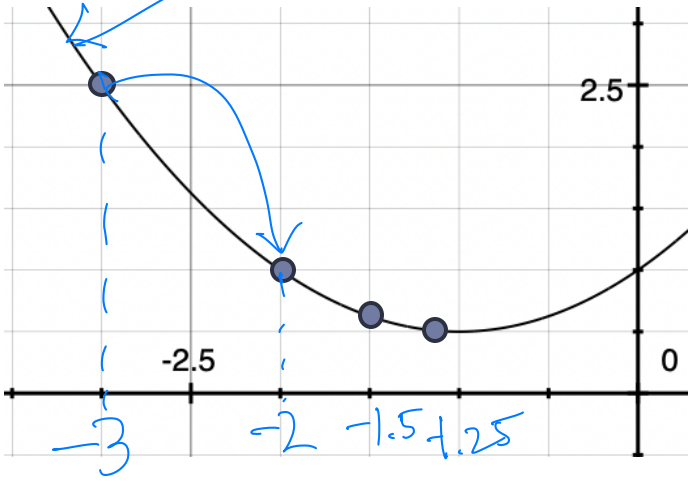
Randomly choose an initial solution, w^0

Repeat

$$w^{t+1} = w^t - \eta \left.\frac{dE}{dw}\right|_{w=w^t}$$

$$t = t + 1$$

Until stopping condition is satisfied



$$E = \frac{1}{2}w^2 + w + 1$$

$$\frac{dE}{dw} = w + 1$$

$$\eta = 0.5$$

$$w^0 = -3$$

$$w^1 = w^0 - \eta \left.\frac{dE}{dw}\right|_{w=w^0} = -3 - 0.5 \times (-2) = -2$$

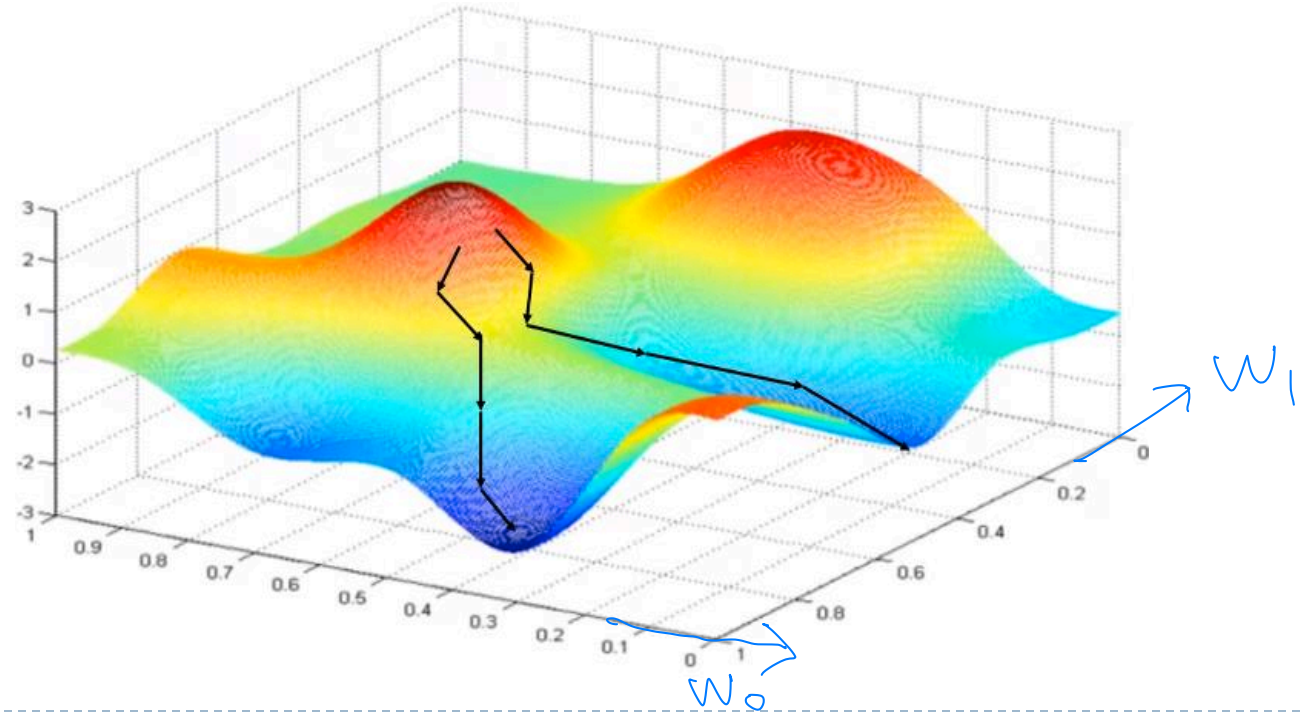
$$w^2 = w^1 - \eta \left.\frac{dE}{dw}\right|_{w=w^1} = -2 - 0.5 \times (-1) = -1.5$$

$$w^3 = w^2 - \eta \left.\frac{dE}{dw}\right|_{w=w^2} = -1.5 - 0.5 \times (-0.5) = -1.25$$

Gradient Descent Method

- **Multivariate Case**

$$E(\mathbf{w}) = \sum_{(\mathbf{x},y)\in Data} \big(y - f(\mathbf{x};\mathbf{w})\big)^2$$



Gradient Descent Method

- **Multivariate Case**
  - **Separately apply the method to each variable**

Randomly choose an initial solution, $w_0^0\ w_1^0$

$t = 0$

Repeat

$$w_0^{t+1} = w_0^t - \eta \left.\frac{\partial E}{\partial w_0}\right|_{w_0=w_0^t, w_1=w_1^t}$$

$$w_1^{t+1} = w_1^t - \eta \left.\frac{\partial E}{\partial w_1}\right|_{w_0=w_0^t, w_1=w_1^t}$$

$t = t + 1$

Until stopping condition is satisfied

Gradient Descent Method

optimization

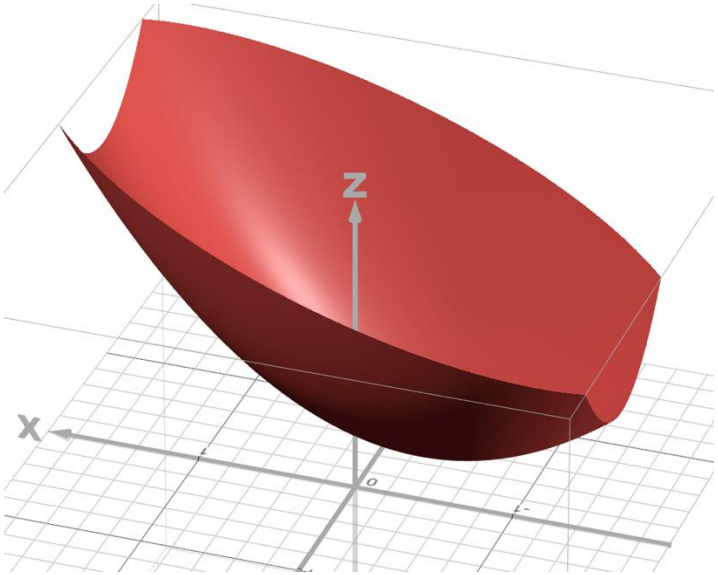
- **Steps**

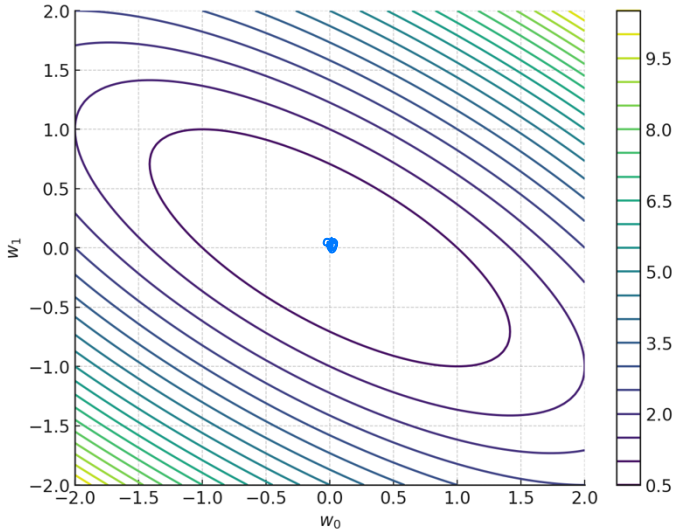
$w_0=0$ $w_1=0$

$$E = \frac{1}{2}w_0^2 + w_1^2 + w_0 w_1 + \frac{1}{2}$$

$$\frac{\partial E}{\partial w_0} = w_0 + w_1$$

$$\frac{\partial E}{\partial w_1} = 2w_1 + w_0$$





Gradient Descent Method

- **Steps**

Randomly choose an initial solution, $w_0^0\ w_1^0$

$t = 0$

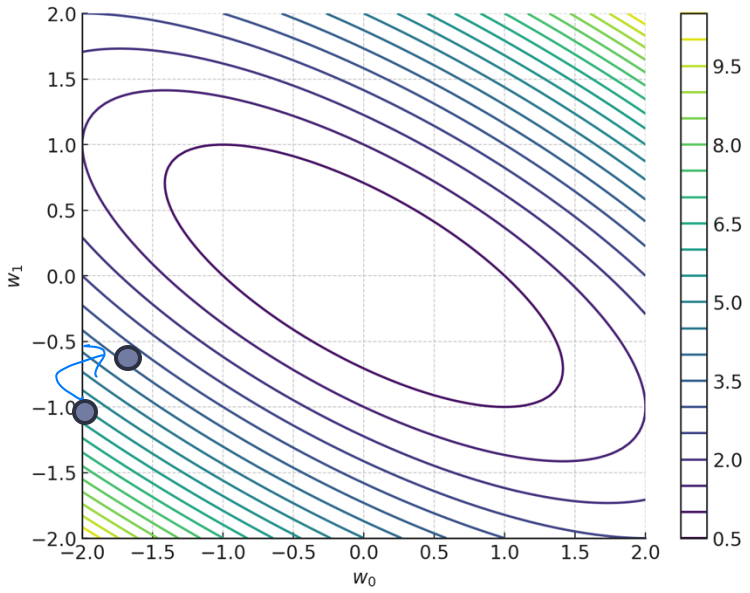
Repeat

$$w_0^{t+1} = w_0^t - \eta \left.\frac{\partial E}{\partial w_0}\right|_{w_0=w_0^t, w_1=w_1^t}$$

$$w_1^{t+1} = w_1^t - \eta \left.\frac{\partial E}{\partial w_1}\right|_{w_0=w_0^t, w_1=w_1^t}$$

$t = t + 1$

Until stopping condition is satisfied



$$E = \frac{1}{2}w_0^2 + w_1^2 + w_0 w_1 + \frac{1}{2}$$

$$\frac{\partial E}{\partial w_0} = w_0 + w_1$$

$$\frac{\partial E}{\partial w_1} = 2w_1 + w_0$$

$\eta = 0.1$

$w_0^0 = -2 \quad w_1^0 = -1$

$w_0^1 = -1.7 \quad w_1^1 = -0.6$

$$w_0^1 = w_0^0 - \eta \left.\frac{\partial E}{\partial w_0}\right|_{w_0=w_0^0, w_1=w_1^0} = -2 - 0.1 \times (-3) = -1.7$$

$$w_1^1 = w_1^0 - \eta \left.\frac{\partial E}{\partial w_1}\right|_{w_0=w_0^0, w_1=w_1^0} = -1 - 0.1 \times (-4) = -0.6$$

Gradient Descent Method

- **Steps**

Randomly choose an initial solution, $w_0^0\ w_1^0$

$t = 0$

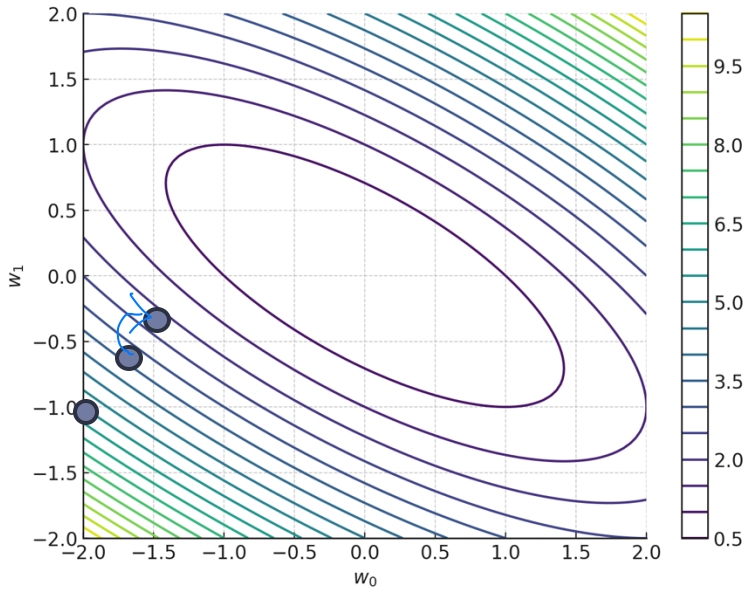
Repeat

$$w_0^{t+1} = w_0^t - \eta \left.\frac{\partial E}{\partial w_0}\right|_{w_0=w_0^t, w_1=w_1^t}$$

$$w_1^{t+1} = w_1^t - \eta \left.\frac{\partial E}{\partial w_1}\right|_{w_0=w_0^t, w_1=w_1^t}$$

$t = t + 1$

Until stopping condition is satisfied



$$E = \frac{1}{2}w_0^2 + w_1^2 + w_0 w_1 + \frac{1}{2}$$

$$\frac{\partial E}{\partial w_0} = w_0 + w_1$$

$$\frac{\partial E}{\partial w_1} = 2w_1 + w_0$$

$$\eta = 0.1$$

$w_0^1 = -1.7 \quad w_1^1 = -0.6$

$w_0^2 = -1.47 \quad w_1^2 = -0.31$

$$w_0^2 = w_0^1 - \eta \left.\frac{\partial E}{\partial w_0}\right|_{w_0=w_0^1, w_1=w_1^1} = -1.7 - 0.1 \times (-2.3) = -1.47$$

$$w_1^2 = w_1^1 - \eta \left.\frac{\partial E}{\partial w_1}\right|_{w_0=w_0^1, w_1=w_1^1} = -0.6 - 0.1 \times (-2.9) = -0.31$$

Gradient Descent Method

- **Steps**

Randomly choose an initial solution, $w_0^0\ w_1^0$

$t = 0$

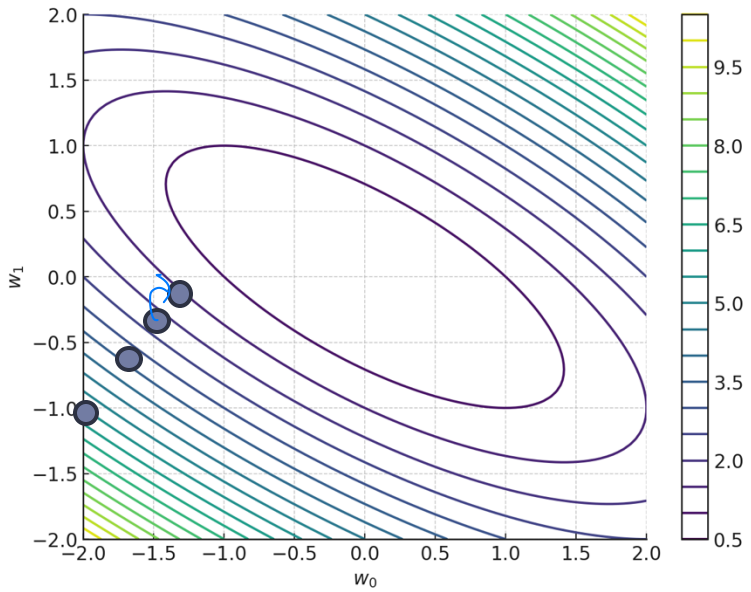
Repeat

$$w_0^{t+1} = w_0^t - \eta \left.\frac{\partial E}{\partial w_0}\right|_{w_0=w_0^t, w_1=w_1^t}$$

$$w_1^{t+1} = w_1^t - \eta \left.\frac{\partial E}{\partial w_1}\right|_{w_0=w_0^t, w_1=w_1^t}$$

$t = t + 1$

Until stopping condition is satisfied



$$E = \frac{1}{2}w_0^2 + w_1^2 + w_0 w_1 + \frac{1}{2}$$

$$\frac{\partial E}{\partial w_0} = w_0 + w_1$$

$$\frac{\partial E}{\partial w_1} = 2w_1 + w_0$$

$$\eta = 0.1$$

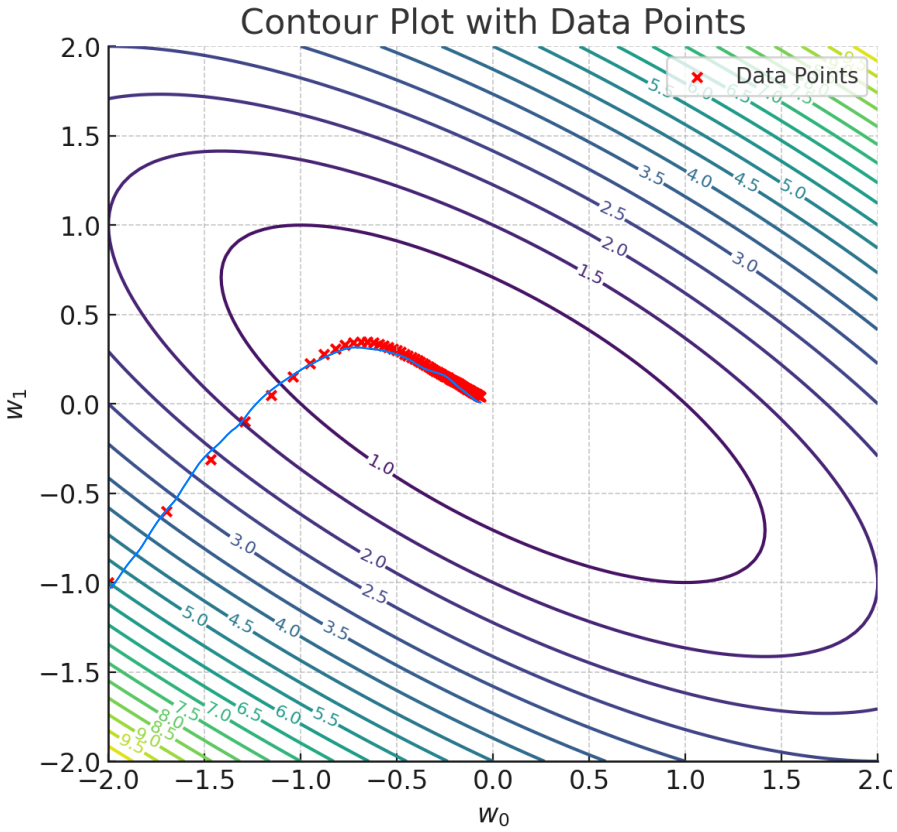
$w_0^2 = -1.47\ \ w_1^2 = -0.31$

$w_0^2 = -1.292\ \ w_1^2 = -0.101$

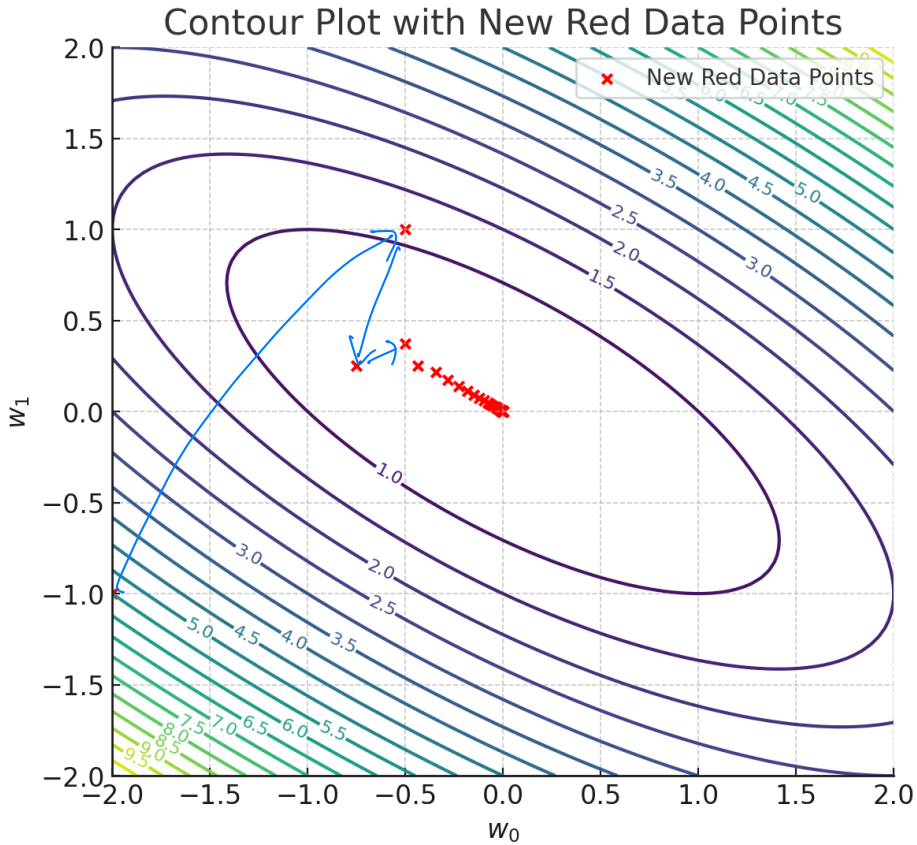
$$w_0^3 = w_0^2 - \eta \left.\frac{\partial E}{\partial w_0}\right|_{w_0=w_0^2, w_1=w_1^2} = -1.47 - 0.1 \times (-1.78) = -1.292$$

$$w_1^3 = w_1^2 - \eta \left.\frac{\partial E}{\partial w_1}\right|_{w_0=w_0^2, w_1=w_1^2} = -0.31 - 0.1 \times (-2.09) = -0.101$$

Gradient Descent Method



$\eta = 0.1$



$\eta = 0.5$

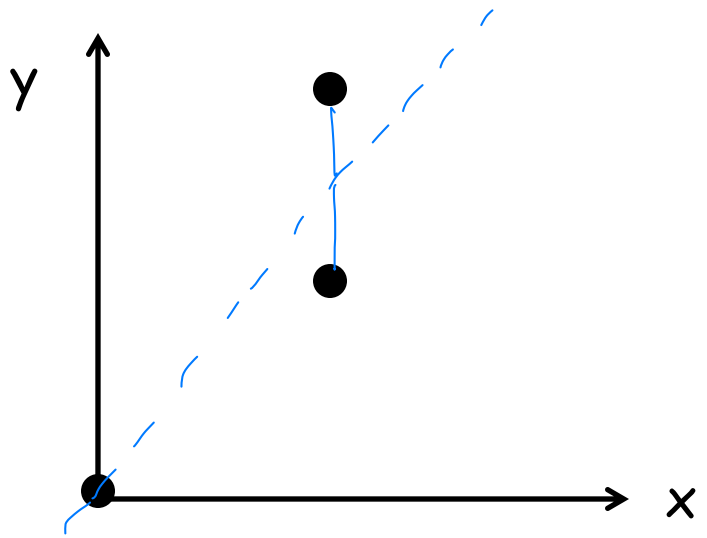
Large eta 값이 클수록 ... 작은 값으로 가자

Linear Regression with Gradient Descent

Linear Regression

- **Find the best-fit line**

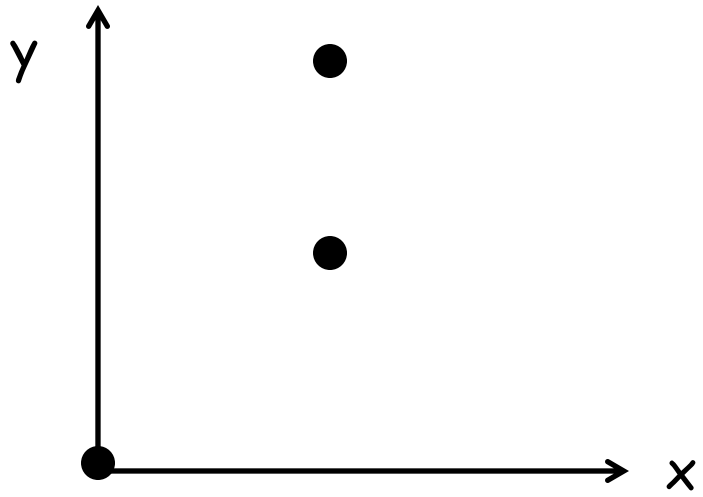
$$(0.0, 0.0)\ (1.0, 1.0)\ (1.0, 2.0)$$



Linear Regression

- **Step 1: Choose a model**

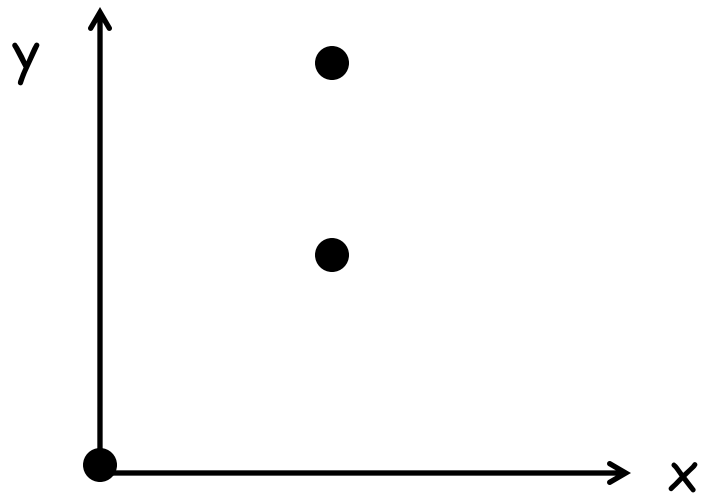
$$f(x; w_0, w_1) = w_1 x + w_0$$



Steps of Machine Learning

- **Step 2: Find w to minimize E by GDM**

$$f(x; w_0, w_1) = w_1 x + w_0$$



Steps of Machine Learning

- **Step 2-1: Obtain a closed form of E**
  - **Error Function**

$$E(w_0, w_1) = \sum_{(\mathbf{x},y)\in Data} \left(y - f(\mathbf{x}; w_1, w_0)\right)^2$$

  - **Obtain a closed form of E by applying the given data**

$(0.0, 0.0)\ (1.0, 1.0)\ (1.0, 2.0)$ $\quad f(x; w_0, w_1) = w_1 x + w_0$

$$E(w_0, w_1) = \left(0.0 - f(0.0; w_0, w_1)\right)^2 + \left(1.0 - f(1.0; w_0, w_1)\right)^2 + \left(2.0 - f(1.0; w_0, w_1)\right)^2$$

$$E(w_0, w_1) = (0.0 - w_0)^2 + \left(1.0 - (w_1 + w_0)\right)^2 + \left(2.0 - (w_1 + w_0)\right)^2$$

$$E(w_0, w_1) = 2{w_1}^2 + 3{w_0}^2 - 6w_1 - 6w_0 + 4w_1 w_0 + 5$$

Steps of Machine Learning

- **Step 2-2: Obtain partial derivatives of E**

error fun

E를 최소화하는 w_0, w_1 찾기 = optimization problem ≠ machine learning problem

$$E(w_0, w_1) = 2{w_1}^2 + 3{w_0}^2 - 6w_1 - 6w_0 + 4w_1w_0 + 5$$

$$\frac{\partial E}{\partial w_0} = 4w_1 + 6w_0 - 6$$

$$\frac{\partial E}{\partial w_1} = 4w_1 + 4w_0 - 6$$

Steps of Machine Learning

- **Step 2-3: Plug the derivatives into the algorithm**

$$\frac{\partial E}{\partial w_0} = 4w_1 + 6w_0 - 6$$

$$\frac{\partial E}{\partial w_1} = 4w_1 + 4w_0 - 6$$

Randomly choose an initial solution, w_0^0 w_1^0

$t = 0$

Repeat

$$w_0^{t+1} = w_0^t - \eta \left.\frac{\partial E}{\partial w_0}\right|_{w_0=w_0^t, w_1=w_1^t}$$

$$w_1^{t+1} = w_1^t - \eta \left.\frac{\partial E}{\partial w_1}\right|_{w_0=w_0^t, w_1=w_1^t}$$

$t = t + 1$

Until stopping condition is satisfied

- **Final algorithm**

Randomly choose an initial solution, w_0^0 w_1^0

$t = 0$

Repeat

$$w_0^{t+1} = w_0^t - \eta(4w_1^t + 6w_0^t - 6)$$

$$w_1^{t+1} = w_1^t - \eta(4w_1^t + 4w_0^t - 6)$$

$t = t + 1$

Until stopping condition is satisfied

Steps of Machine Learning

- **Step 2-4: Run the algorithm**
  - Choose a small value for η ($\eta = 0.1$)
  - Randomly choose w_0^0, w_1^0. ($w_0^0 = 1, w_1^0 = 1$)

Randomly choose an initial solution, $w_0^0\ w_1^0$

$t = 0$

Repeat

$$w_0^{t+1} = w_0^t - \eta \left.\frac{\partial E}{\partial w_0}\right|_{w_0=w_0^t, w_1=w_1^t}$$

$$w_1^{t+1} = w_1^t - \eta \left.\frac{\partial E}{\partial w_1}\right|_{w_0=w_0^t, w_1=w_1^t}$$

$t = t + 1$

Until stopping condition is satisfied

$$w_0^0 = 1$$
$$w_1^0 = 1$$

Jump

$$w_0^1 = 1 - 0.1(4 \times 1 + 6 \times 1 - 6) = 0.6$$
$$w_1^1 = 1 - 0.1(4 \times 1 + 4 \times 1 - 6) = 0.8$$

Jump

$$w_0^2 = 0.6 - 0.1(4 \times 0.8 + 6 \times 0.6 - 6) = 0.54$$
$$w_1^2 = 0.8 - 0.1(4 \times 0.8 + 4 \times 0.6 - 6) = 0.84$$

Jump

$$w_0^3 = 0.54 - 0.1(4 \times 0.84 + 6 \times 0.54 - 6) = 0.480$$
$$w_1^3 = 0.84 - 0.1(4 \times 0.84 + 4 \times 0.54 - 6) = 0.888$$

Steps of Machine Learning

- **Step 2-4: Run the algorithm**

$$w_0^3 = 0.54 - 0.1(4 \times 0.84 + 6 \times 0.54 - 6) = 0.480$$
$$w_1^3 = 0.84 - 0.1(4 \times 0.84 + 4 \times 0.54 - 6) = 0.888$$

$$w_0^4 = 0.480 - 0.1(4 \times 0.888 + 6 \times 0.480 - 6) = 0.4368$$
$$w_1^4 = 0.888 - 0.1(4 \times 0.888 + 4 \times 0.480 - 6) = 0.9408$$

…

$$w_0^{100} = 0.00007713$$
$$w_1^{100} = 1.49989171$$

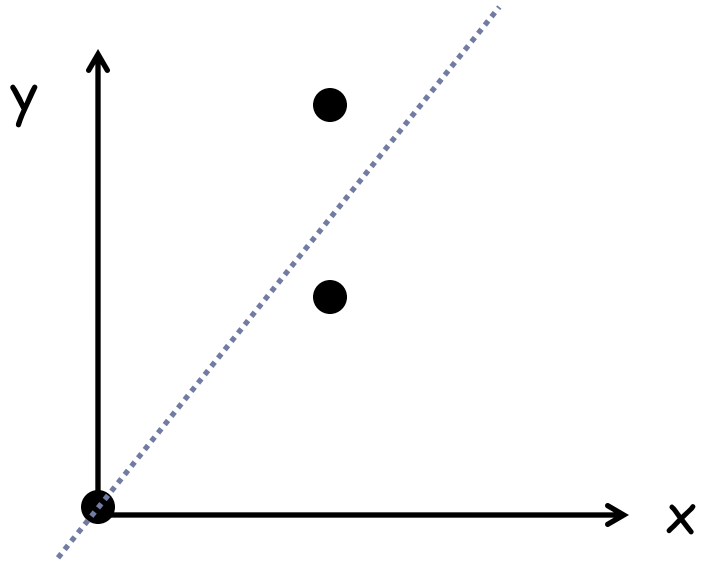
$w_0 \approx 0$

$w_1 \approx 1.5$

Steps of Machine Learning

- **Final Result**

$$f(x)=1.49989171x + 0.00007713$$



But.. if I have millions of samples

- **If there are a small number of samples, we can manually obtain derivatives**

$$E(w_0, w_1) = \sum_{(\mathbf{x},t)\in Data} \left(t - f(\mathbf{x}; w_1, w_0)\right)^2$$

definition

  - **For three samples, we manually obtain the closed form of E**

"

(sample 3개 정의하면서)

$$E(w_0, w_1) = \left(0.0 - f(0.0; w_0, w_1)\right)^2 + \left(1.0 - f(1.0; w_0, w_1)\right)^2 + \left(2.0 - f(1.0; w_0, w_1)\right)^2$$

$$E(w_0, w_1) = (0.0 - w_0)^2 + \left(1.0 - (w_1 + w_0)\right)^2 + \left(2.0 - (w_1 + w_0)\right)^2$$

$$E(w_0, w_1) = 2{w_1}^2 + 3{w_0}^2 - 6w_1 - 6w_0 + 4w_1w_0 + 5$$

  - **We manually obtain derivatives**

$$\frac{\partial E}{\partial w_0} = 4w_1 + 6w_0 - 6 \qquad \frac{\partial E}{\partial w_1} = 4w_1 + 4w_0 - 6$$

manually : 실용적인 방법은 아니다!

- **How can I MANUALLY obtain derivatives? if I have millions of samples.. :(**

But.. if I have millions of samples

definition

$$E(w_0, w_1) = \sum_{(\mathbf{x},y)\in Data} \left(y - f(\mathbf{x}; w_1, w_0)\right)^2$$

Closed form of E manually

$$E(w_0, w_1) = 2{w_1}^2 + 3{w_0}^2 - 6w_1 - 6w_0 + 4w_1w_0 + 5$$

Differentiation

$$\frac{\partial E}{\partial w_0} = 4w_1 + 6w_0 - 6$$

$$\frac{\partial E}{\partial w_1} = 4w_1 + 4w_0 - 6$$

sample 안 생각하면 ↓

Differentiation without closed form of E directly differentiable

closed formol 필요없다!

$$\frac{\partial E}{\partial w_0} = \sum_{(\mathbf{x},y)\in Data} \frac{\partial}{\partial w_0}\left(y - f(\mathbf{x}; w_1, w_0)\right)^2$$

$$\frac{\partial E}{\partial w_1} = \sum_{(\mathbf{x},y)\in Data} \frac{\partial}{\partial w_1}\left(y - f(\mathbf{x}; w_1, w_0)\right)^2$$

But.. if I have millions of samples

- **Another way to obtain derivatives at a point**

$$E(w_0, w_1) = \sum_{(\mathbf{x}_i, t_i) \in Data} \left(t_i - f(\mathbf{x}_i; w_1, w_0)\right)^2$$

Sum of E

$$= \sum_{(\mathbf{x}_i, t_i) \in Data} E_i(\mathbf{x}_i, t_i; w_0, w_1)$$

where $E_i(\mathbf{x}_i, t_i; w_1, w_0) = \left(t_i - f(\mathbf{x}_i; w_1, w_0)\right)^2$

- **then**

$$\frac{\partial}{\partial w_j} E(w_0, w_1) = \frac{\partial}{\partial w_j} \sum_{(\mathbf{x}_i, t_i) \in Data} E_i(\mathbf{x}_i, t_i; w_1, w_0)$$

$$= \sum_{(\mathbf{x}_i, t_i) \in Data} \frac{\partial}{\partial w_j} E_i(\mathbf{x}_i, t_i; w_1, w_0)$$

But.. if I have millions of samples

- **Another way to obtain derivatives**
  - **Derivative of E**

$$\frac{\partial E}{\partial w_j} = \sum_{(\mathbf{x}_i, t_i) \in Data} \frac{\partial E_i}{\partial w_j}$$

  - **Derivative of E at (w_0^t, w_1^t)**

$$\left.\frac{\partial E}{\partial w_j}\right|_{w_0 = w_0^t, w_1 = w_1^t} = \sum_{(\mathbf{x}_i, t_i) \in Data} \left.\frac{\partial E_i}{\partial w_j}\right|_{w_0 = w_0^t, w_1 = w_1^t}$$

But.. if I have millions of samples

- **Another way to obtain derivatives**

Randomly choose an initial solution, $w_0^0\ w_1^0$

$t = 0$

Repeat

$$w_0^{t+1} = w_0^t - \eta \left.\frac{\partial E}{\partial w_0}\right|_{w_0=w_0^t, w_1=w_1^t}$$

$$w_1^{t+1} = w_1^t - \eta \left.\frac{\partial E}{\partial w_1}\right|_{w_0=w_0^t, w_1=w_1^t}$$

$t = t + 1$

Until stopping condition is satisfied

$$\sum_{(\mathbf{x}_i, t_i) \in Data} \left.\frac{\partial E_i}{\partial w_0}\right|_{w_0=w_0^t, w_1=w_1^t}$$

$$\sum_{(\mathbf{x}_i, t_i) \in Data} \left.\frac{\partial E_i}{\partial w_1}\right|_{w_0=w_0^t, w_1=w_1^t}$$

But.. if I have millions of samples

- **New Algorithm**

* DNN에서 주로 사용! (Deep Neural Network)
여기엔 보통 엄청 많은 Data를 사용하기 때문

Randomly choose an initial solution, w_0^0 w_1^0

$t = 0$

New algorithm of GDM (sample 몇 개씩)

Repeat

$$w_0^{t+1} = w_0^t - \eta \sum_{(\mathbf{x}_i, t_i) \in Data} \left. \frac{\partial E_i}{\partial w_0} \right|_{w_0 = w_0^t, w_1 = w_1^t}$$

python에서 for, while로 구현

$$w_1^{t+1} = w_1^t - \eta \sum_{(\mathbf{x}_i, t_i) \in Data} \left. \frac{\partial E_i}{\partial w_1} \right|_{w_0 = w_0^t, w_1 = w_1^t}$$

$t = t + 1$

Until stopping condition is satisfied

But.. if I have millions of samples

- **New Algorithm**

Randomly choose an initial solution, w_0^0 w_1^0

$t = 0$

Repeat

$g_0^t = 0;\ g_1^t = 0$

for all $(\mathbf{x}_i, t_i) \in Data$

$$g_0^t = g_0^t + \frac{\partial E_i}{\partial w_0}\Big|_{w_0=w_0^t, w_1=w_1^t}$$

$$g_1^t = g_1^t + \frac{\partial E_i}{\partial w_0}\Big|_{w_0=w_0^t, w_1=w_1^t}$$

$\Sigma \frac{\partial E_i}{\partial w}$ 의 값을

$$w_0^{t+1} = w_0^t - \eta g_0^t$$

$$w_1^{t+1} = w_1^t - \eta g_1^t$$

$$t = t + 1$$

Until stopping condition is satisfied

But.. if I have millions of samples

- **Example**
  - **Given data:** $\{(0.0, 0.0), (1.0, 1.0), (1.0, 2.0)\}$
  - **Model chosen:** $f(x; w_0, w_1) = w_1 x + w_0$
  - **Find w to minimize**

$$E(w_0, w_1) = \sum_{(\mathbf{x}_i, t_i) \in Data} \left(t_i - f(\mathbf{x}_i; w_1, w_0)\right)^2$$

But.. if I have millions of samples

- **Example**
  - **Step 1: Formulate** $\left.\frac{\partial E_i}{\partial w_0}\right|_{w_0=w_0^t, w_1=w_1^t}$ **and** $\left.\frac{\partial E_i}{\partial w_0}\right|_{w_0=w_0^t, w_1=w_1^t}$

$$g_0^t = g_0^t + \left.\frac{\partial E_i}{\partial w_0}\right|_{w_0=w_0^t, w_1=w_1^t} \qquad g_1^t = g_1^t + \left.\frac{\partial E_i}{\partial w_0}\right|_{w_0=w_0^t, w_1=w_1^t}$$

$$E_i = \left(t_i - (w_1 x_i + w_0)\right)^2$$

$$\frac{\partial E_i}{\partial w_0} = -2\left(t_i - (w_1 x_i + w_0)\right)$$

$$\frac{\partial E_i}{\partial w_1} = -2\left(t_i - (w_1 x_i + w_0)\right) \cdot x_i$$

$$\left.\frac{\partial E_i}{\partial w_0}\right|_{w_0=w_0^t, w_1=w_1^t} = -2\left(t_i - (w_1^t x_i + w_0^t)\right)$$

$$\left.\frac{\partial E_i}{\partial w_1}\right|_{w_0=w_0^t, w_1=w_1^t} = -2\left(t_i - (w_1^t x_i + w_0^t)\right) \cdot x_i$$

But.. if I have millions of samples

- **Example**
  - **Step 2: Plug the formulae into the algorithm**

Randomly choose an initial solution, $w_0^0\ w_1^0$

$t = 0$

Repeat

$$g_0^t = 0;\ g_1^t = 0$$

for all $(\mathbf{x}_i, t_i) \in Data$

$$g_0^t = g_0^t - 2\left(t_i - (w_1^t x_i + w_0^t)\right)$$

$$g_1^t = g_1^t - 2\left(t_i - (w_1^t x_i + w_0^t)\right) \cdot x_i$$

$$w_0^{t+1} = w_0^t - \eta g_0^t$$

$$w_1^{t+1} = w_1^t - \eta g_1^t$$

$$t = t + 1$$

Until stopping condition is satisfied

But.. if I have millions of samples

- **Example**
  - **Step 3: Randomly initialize w_0^t and w_1^t: $w_0^0 = 1, w_1^0 = 1$**
    Set η with a small value: $\eta = 0.1$
    Run the algorithm

Randomly choose an initial solution, $w_0^0\ w_1^0$
$t = 0$
Repeat

$g_0^t = 0;\ g_1^t = 0$

for all $(\mathbf{x}_i, t_i) \in Data$

$$g_0^t = g_0^t - 2\left(t_i - (w_1^t x_i + w_0^t)\right)$$
$$g_1^t = g_1^t - 2\left(t_i - (w_1^t x_i + w_0^t)\right) \cdot x_i$$

$$w_0^{t+1} = w_0^t - \eta g_0^t$$
$$w_1^{t+1} = w_1^t - \eta g_1^t$$
$$t = t + 1$$

Until stopping condition is satisfied

$$Data = \{(0.0, 0.0), (1.0, 1.0), (1.0, 2.0)\}$$

$$w_0^0 = 1, w_1^0 = 1$$

$$g_0^0 = \sum -2\left(t_i - (w_1^0 x_i + w_0^0)\right)$$
$$= -2\big(0 - (0 + 1)\big) - 2\big(1 - (1 + 1)\big)$$
$$-2\big(2 - (1 + 1)\big) = 4$$

$$g_1^0 = \sum -2\left(t_i - (w_1^0 x_i + w_0^0)\right) \cdot x_i$$
$$= -2\big(0 - (0 + 1)\big) \cdot 0 - 2\big(1 - (1 + 1)\big) \cdot 1$$
$$-2\big(2 - (1 + 1)\big) \cdot 1 = 2$$

$$w_0^1 = 1 - 0.1 \cdot 4 = 0.6$$
$$w_1^1 = 1 - 0.1 \cdot 2 = 0.8$$

Question and Answer