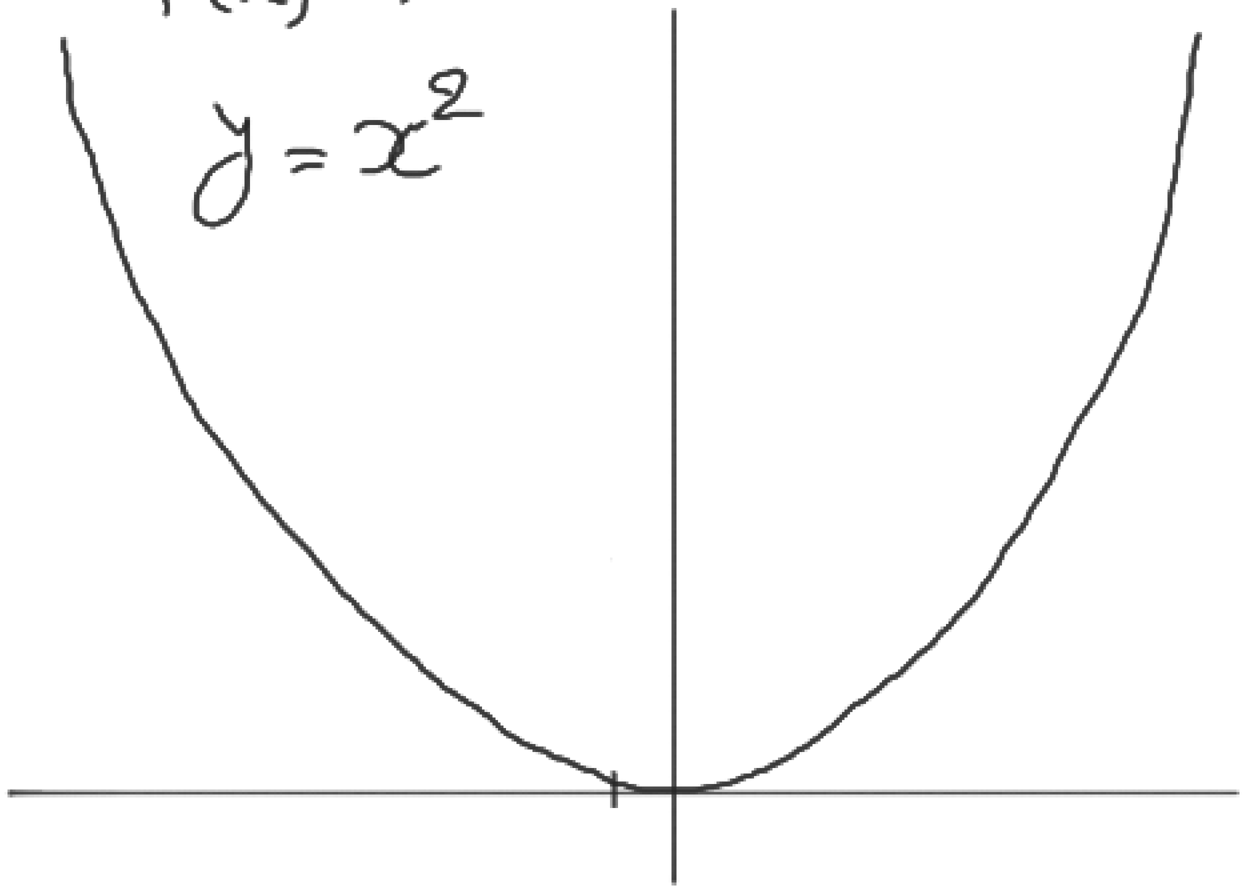
★ Partial Derivatives

What if our function involves more than one variables?

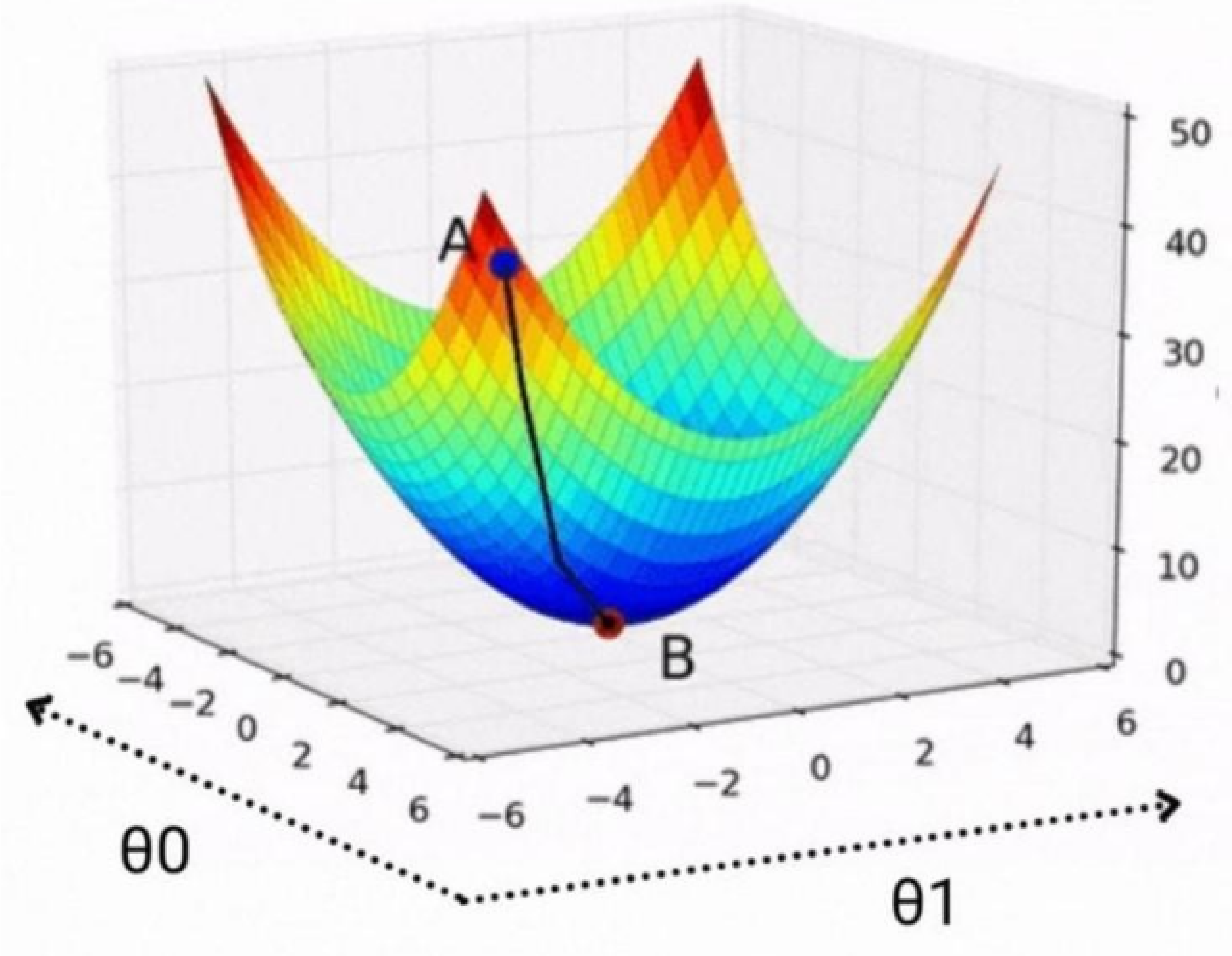
A function with one variable
(2-D function)

$f(x) = x^2$

$y = x^2$



A function with 2 variables
(3-D function)



Let's take such a function $f(x,y) = 2x^2y + 3y^3x^2 + 3y$

so now we can't take differentiation w.r.t. x

$\therefore$ we will differentiate $f(x,y)$ w.r.t. x & y one by one.

$$\frac{d}{dx} f(x,y) \rightarrow \frac{\partial}{\partial x} f(x,y) = \frac{\partial}{\partial x}\left(2x^2y + 3y^3x^2 + 3y\right)$$

$$\left.\begin{array}{l} \frac{\partial}{\partial x} f(x,y) = 4xy + 6y^3x \\ \text{similarly} \\ \frac{\partial}{\partial y} f(x,y) = 2x^2 + 9y^2x^2 + 3 \end{array}\right\} \rightarrow \begin{bmatrix} \frac{\partial}{\partial x} f \\ \frac{\partial}{\partial y} f \end{bmatrix} = \nabla f$$

= gradient

☆ In ML Context:

Our Loss Function $= \underset{\vec{\omega}, \omega_0}{\operatorname{argmin}} L(\underbrace{\omega_1, \omega_2, \omega_3, \ldots \omega_n}_{\vec{\omega}}, \omega_0)$

$\therefore$ To calculate gradient we must differentiate L by $\omega_1, \omega_2, \omega_3, \ldots, \omega_n, \omega_0$ one by one (partial derivative)

$$\nabla L = \begin{bmatrix} \partial L / \partial \omega_0 \\ \partial L / \partial \omega_1 \\ \partial L / \partial \omega_2 \\ \vdots \\ \partial L / \partial \omega_n \end{bmatrix}$$

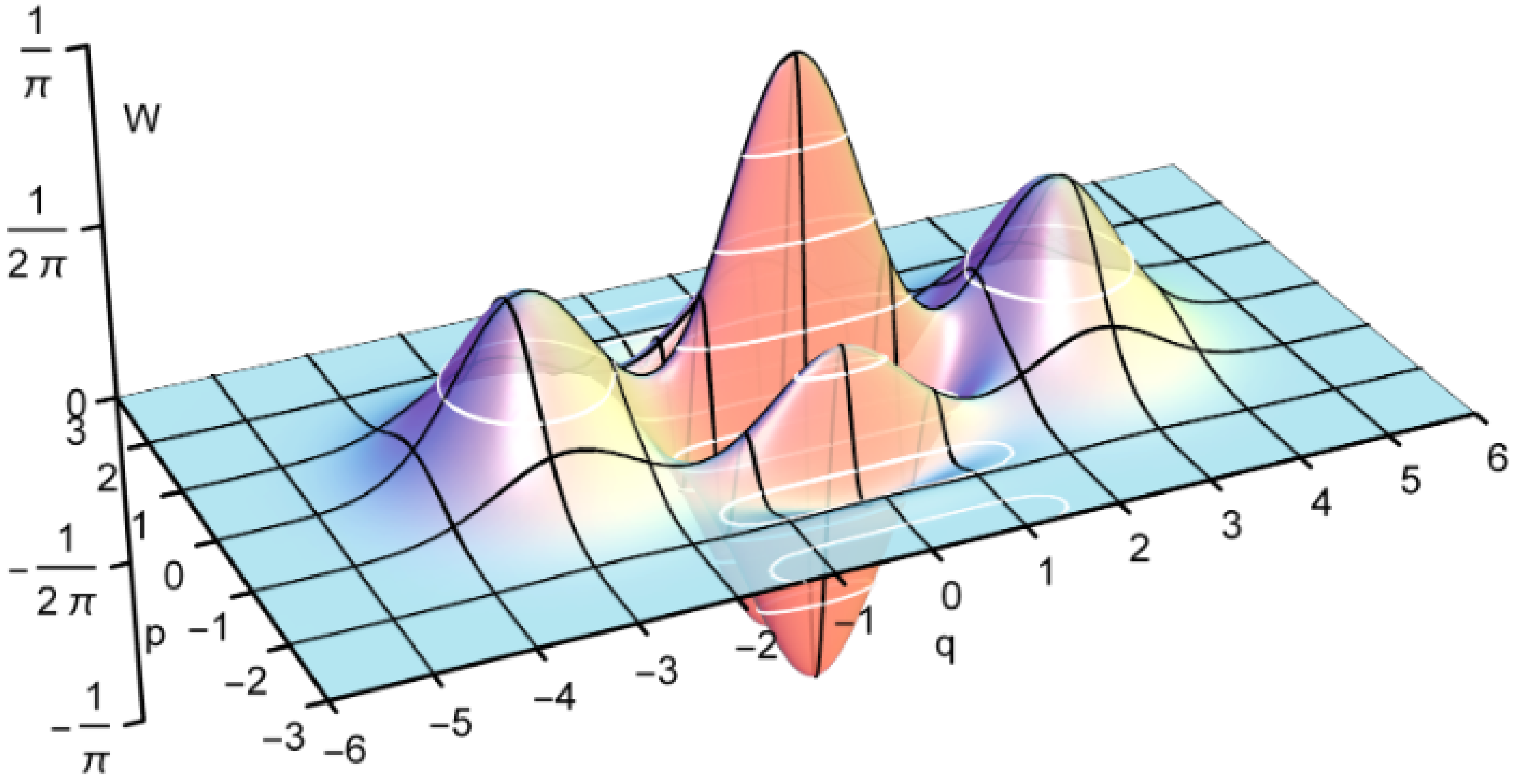
★ <u>An</u> <u>issue</u>: Can we always consider 'y' as constant while differentiating with respect to 'x'?

→ Let's take this function: $2x^2y^2 + 3xy + 4y$ where $y = 4x^2$

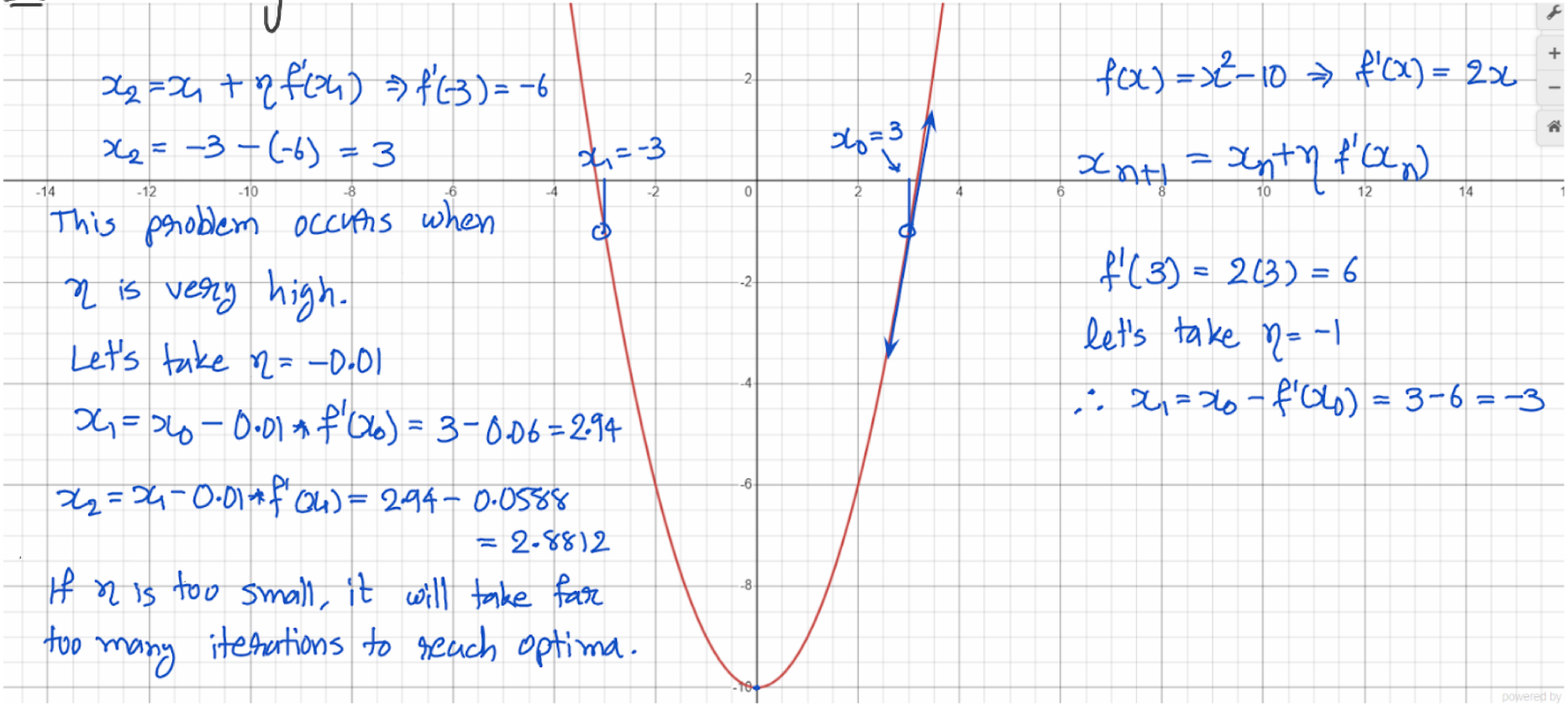
can we take y constant when we are differentiating w.r.t. x? <u>No, because y depends on x</u>

∴ If some feature is a function of other feature(s) (depends on other feature(s)) then we will not be able to differentiate the function partially. (That is also why they are called independent variables)

☆ Two more examples of a real-world functions:



★ Learning Rate - Let's take $f(x) = x^2 - 1$



$f(x) = x^2 - 10 \Rightarrow f'(x) = 2x$

$x_{n+1} = x_n + \eta f'(x_n)$

$f'(3) = 2(3) = 6$

let's take $\eta = -1$

$\therefore x_1 = x_0 - f'(x_0) = 3 - 6 = -3$

$x_2 = x_1 + \eta f'(x_1) \Rightarrow f'(-3) = -6$

$x_2 = -3 - (-6) = 3$

This problem occurs when η is very high.

Let's take $\eta = -0.01$

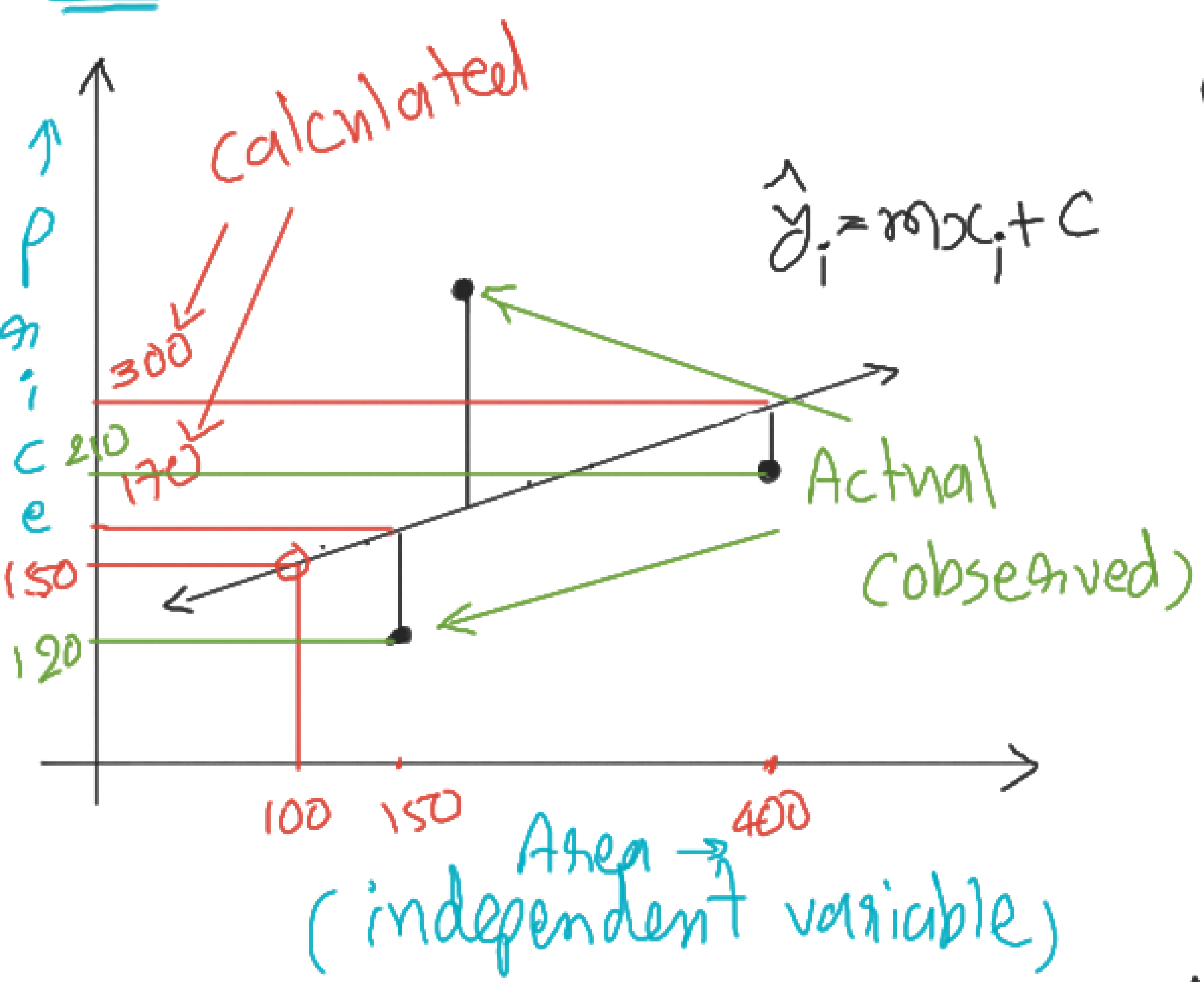
$x_1 = x_0 - 0.01 * f'(x_0) = 3 - 0.06 = 2.94$

$x_2 = x_1 - 0.01 * f'(x_1) = 2.94 - 0.0588$
$= 2.8812$

If η is too small, it will take far too many iterations to reach optima.

Our speed of convergence is controlled by η that's why it is called Learning Rate.

☆ Error calculation in Linear Regression (Loss F^n)



where $\hat{y}_i$ is calculated y while y_i is actual (observed) y.

$$\text{Error} = y_i - \hat{y}_i$$

$$\text{squaring the error} = (y_i - \hat{y}_i)^2$$

$$\text{sum of squares} = \sum (y_i - \hat{y}_i)^2$$

As $\hat{y}_i$ involves m & c, let's write this as a function of m & c $\Rightarrow$ $J(m, c) = \dfrac{\sum (y_i - \hat{y}_i)^2}{N}$

where N = no. of observations.

$$\therefore J(m, c) = \sum \frac{\left(y_i - (mx_i + c)\right)^2}{N}$$

Taking partial derivatives to find the gradient:

$$\frac{\partial}{\partial m} J(m, c) = \frac{1}{N} \sum 2\left(y_i - (mx_i + c)\right)^1 \cdot \frac{\partial}{\partial m}\left(y_i - mx_i - c\right)$$

$$\boxed{\frac{\partial}{\partial m} J(m, c) = \frac{-2}{N} \sum \left(y_i - (mx_i + c)\right) \cdot x_i}$$

similarly, $\frac{\partial}{\partial c} J(m, c) = \frac{1}{N} \sum 2\left(y_i - (mx_i + c)\right)^1 \cdot \frac{\partial}{\partial c}\left(y_i - mx_i - c\right)$

$$\boxed{\frac{\partial}{\partial c} J(m, c) = \frac{-2}{N} \sum \left(y_i - (mx_i + c)\right)}$$

* Constrained Optimization

Recap & new notation: In gradient descent, formula to find next guess was - $x_{t+1} = x_t - \eta f'(x)$

→ Above formula is applicable only for univariate g.d. (eg. predicting price of a house from only one variable - area). If we want multi-variate g.d. then the formula will change to:

$$X_{t+1} = X_t - \eta \left(\frac{\partial}{\partial x} f(x, y) + \frac{\partial}{\partial y} f(x, y) \right)$$ for two variables

For n variables:

$$X_{t+1} = X_t - \eta \left(\frac{\partial}{\partial x_1} f(x_1, x_2, \ldots x_n) + \frac{\partial}{\partial x_2} f(x_1, x_2, \ldots x_n) + \ldots + \frac{\partial}{\partial x_n} f(x_1, x_2, \ldots x_n) \right)$$

→ Using Σ notation:

$$X_{t+1} = X_t - \eta \sum_{x=1}^{n} \frac{\partial}{\partial x_i} f(x_1, x_2, \ldots x_n)$$

New notation: $X_t \to \theta^t$ or $X_t \to \theta_t$

$$\boxed{\therefore\ \theta^{t+1} = \theta^t - \eta \sum \frac{\partial}{\partial \theta} f}$$

☆ Constrained Optimization Problem

Our Loss Function is :

$$L = -\sum \frac{\vec{w}^T \cdot \vec{x} + w_0}{\|\vec{w}\|} \cdot y_i$$

If we partially differentiate this, it will be done as per division rule of differentiation. Hence, to calculate Loss of even one line, we need to apply division rule n times and the sum up them all. (where n is number of variables/features)

$\therefore$ This is going to take too many operations. Can we reduce this problem to a lower scale by putting some limitations (constraints) on it?

☆ The constraint :

① We don't use magnitude of $\vec{\omega}$ anywhere. We just need to see its direction. So if we take a special case where $||\vec{\omega}|| = 1$, it is not going to impact our algorithm adversly.

② But due to this 'constraint', our formula of the loss f^n will be very much simplified as the denominator will become 1 and hence differentiation will also get simple resulting in reducing no. of operations to be performed.

so now we will calculate :

$$\boxed{\underset{\vec{\omega}, \omega_0}{\text{argmin}} - \sum \frac{\vec{\omega}^T \cdot \vec{x} + \omega_0}{||\vec{\omega}||} \cdot y_i \quad s.t. \; ||\vec{\omega}|| = 1}$$

Here s.t. stands for "such that" or "subject to" and this new form of problem (with the constraint) is known as "constrained optimization problem".

→ Let's see how simple the problem becomes due to this:

Let the f^n be $f(x, y)$ $\therefore \frac{\partial}{\partial w_1} L$ & $\frac{\partial}{\partial w_2} L$ & $\vec{w} = [w_1, w_2]$

$$\therefore \frac{\partial}{\partial w_1} L = \frac{-(w_1 x_1 + w_2 x_2 + w_0) \cdot y_i}{1} = \frac{-\partial}{\partial w_1} \left(w_1 x_1 y_1 + \cancel{w_2 x_2 y_2} + \dots + \cancel{w_n x_n y_n} + \cancel{w_0}\right)$$

$$= -x_1 y_1$$

$$\frac{\partial}{\partial w_2} L = -x_2 y_2 \Rightarrow \frac{\partial}{\partial w_i} L = -\sum x_i y_i$$

$\therefore$ The formula for the next guess of ω in G.D.

(from the formula $x_{t+1} = x_t - \eta \sum \frac{\partial}{\partial x_i} f$):

$$\omega_1^{t+1} = \omega_1^t - \eta \frac{\partial}{\partial \omega_1^t} L$$

$$= \omega_1^t - \eta x_1 y_1$$

$$\boxed{\omega_i^{t+1} = \omega_i^t - \eta\left(-\sum x_i y_i\right) ; \; i \neq 0}$$

For ω_0: $\boxed{\omega_0^{t+1} = \omega_0^t - \eta\left(-\sum y_i\right)}$

☆ Suppose the function that we want to minimize is:

$f(x) = x^2 - 3x - 3$ & we also put a constraint that our minima must also satisfy $g(x) = x^2 - 2x - 3$

Example-1

∴ Our constraind problem looks like this:

$$\boxed{\underset{x}{\operatorname{argmin}} \; f(x) \;\; \text{s.t.} \; g(x)} \quad \text{I}$$

s.t. = 'such that' or 'subject to'

Goal: To do something so that $f(x)$ is also minimized and the constraint is also taken care of. Also there is a problem with I that how to differentiate "s.t." part?

Solution of all those issues is : Lagrange's Multiplier

Instead of computing I as it is, we will transform it as:

$$\boxed{\underset{x,\lambda}{\operatorname{argmin}} \; \underbrace{f(x)}_{\text{Hey, minimize the } f(x)!} + \underbrace{\lambda g(x)}_{\text{Also take care of this!}}} \qquad \text{II}$$

→ As above form of equation doesn't have any "s.t.", it is also known as "unconstrained problem".

→ Formula - II is only valid only when there is just one constraint. For multiple constraints, it will become :

$$\underset{x, \lambda_1, \lambda_2, \ldots, \lambda_n}{\operatorname{argmin}} \; f(x) + \lambda_1 g_1(x) + \lambda_2 g_2(x) + \ldots + \lambda_n g_n(x)$$

where $\lambda_1, \lambda_2, \ldots, \lambda_n$ are Lagrange's Multiplier & $g_1(x), g_2(x), \ldots g_n(x)$ are the constraints.

☆ Now let's try to solve Example-1 with this new method

$f(x) = x^2 - 3x - 3$; $g(x) = x^2 - 2x - 3$

$\therefore$ Unconstraind formula will be:

$$\underset{x,\lambda}{\operatorname{argmin}} \; f(x) + \lambda g(x) = \underset{x,\lambda}{\operatorname{argmin}} \; x^2 - 3x - 3 + \lambda x^2 - 2\lambda x - 3\lambda$$

To minimize this, we need to calculate gradient

$$\nabla h = \begin{bmatrix} \frac{\partial}{\partial x} h \\ \\ \frac{\partial}{\partial \lambda} h \end{bmatrix} = \begin{bmatrix} 2x - 3 + 2\lambda x - 2\lambda \\ \\ x^2 - 2x - 3 \end{bmatrix}$$

At the minima $\nabla h = 0$ $\therefore$ Equating both component of ∇h with 0 :

$2x + 2\lambda x - 2\lambda - 3 = 0$ —— (A) & $x^2 - 2x - 3 = 0$ —— (B)

$x^2 - 3x + x - 3 = 0$

$\therefore x(x-3) + 1(x-3) = 0$

$\therefore x = 3$ OR $x = -1$

For $x = 3$:

$6 + 6\lambda - 2\lambda - 3 = 0$

$\therefore 4\lambda = -3 \Rightarrow \boxed{\lambda = \frac{-3}{4}}$

For $x = -1$:

$-2 - 2\lambda - 2\lambda - 3 = 0$

$-4\lambda = 5 \Rightarrow \boxed{\lambda = \frac{-5}{4}}$

$\therefore$ Our two points (x, λ) are:

$\left(3, \frac{-3}{4}\right)$ & $\left(-1, \frac{-5}{4}\right)$

Finally, $f(x) + \lambda g(x)$ for both these points are:

For $(3, -\frac{3}{4})$: $f(x) + \lambda g(x) = -3$ & for $(-1, \frac{-5}{4})$: $f(x) + \lambda g(x) = 1$

$\therefore$ $f(x) + \lambda g(x)$ is minimum for $x = 3$ & $\lambda = \frac{-3}{4}$

$\therefore$ They are our solution.