Introduction to Optimization

Instructor: Dr. Mennatullah Siam

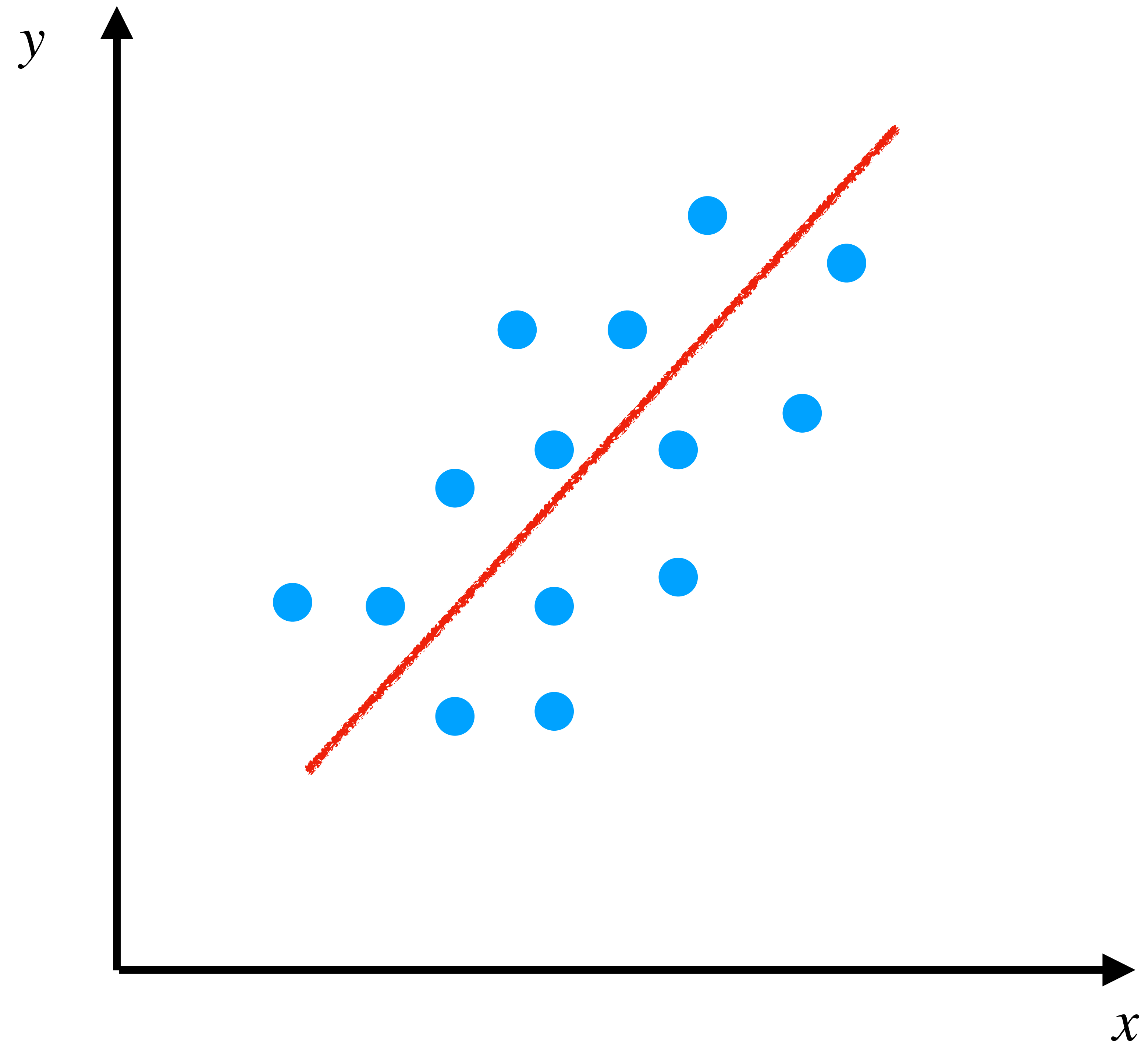
Recap

- Linear Regression
- Solving it using Normal Equation

$$Xw = y$$

$$w = (X^\top X)^{-1} X^\top y$$

General multivariate case



Optimization Intro

Goal:

- How to minimize an objective function?

Reference:

Heath, Michael T. *Scientific computing: an introductory survey, second edition*.

Objective Function

Fuel efficiency

Engine performance

Objective Function

Parameters: Vehicle Shape - Vehicle Weight

Objective Function

- Let's take linear regression as an example.

- We try to minimize this cost/objective function:

$$\min_{w} \frac{1}{n} \sum_{i=0}^{n} (x_i^\top w - y_i)^2$$

Objective Function

- Let's take linear regression as an example.

- We try to minimize this cost/objective function:

$$\min_{w} \frac{1}{n} \sum_{i=0}^{n} (x_i^\top w - y_i)^2$$

Matrix-Vector Multiplication Form

$$\min_{w} (Xw - y)^\top (Xw - y)$$

Closed Form Solution!

- We have seen the closed form solution last lecture.

$$w = (X^\top X)^{-1} X^\top y$$

Closed Form Solution!

- We have seen the closed form solution last lecture.
- But not all problems have a closed form solution!

Deep Neural Networks !!

$$w = (X^\top X)^{-1} X^\top y$$

Closed Form Solution!

- We have seen the closed form solution last lecture.
- But not all problems have a closed form solution!
- Also with large scale data!

$$X$$

$$1M \times 4096$$

1M examples in your dataset with 4096 feature vector per example

How about Random Search?

- Let's assume the simplest univariate case 1 feature per example.

$$\min_{w} \frac{1}{n} \sum_{i=0}^{n} (x_i w - y_i)^2$$

How about Random Search?

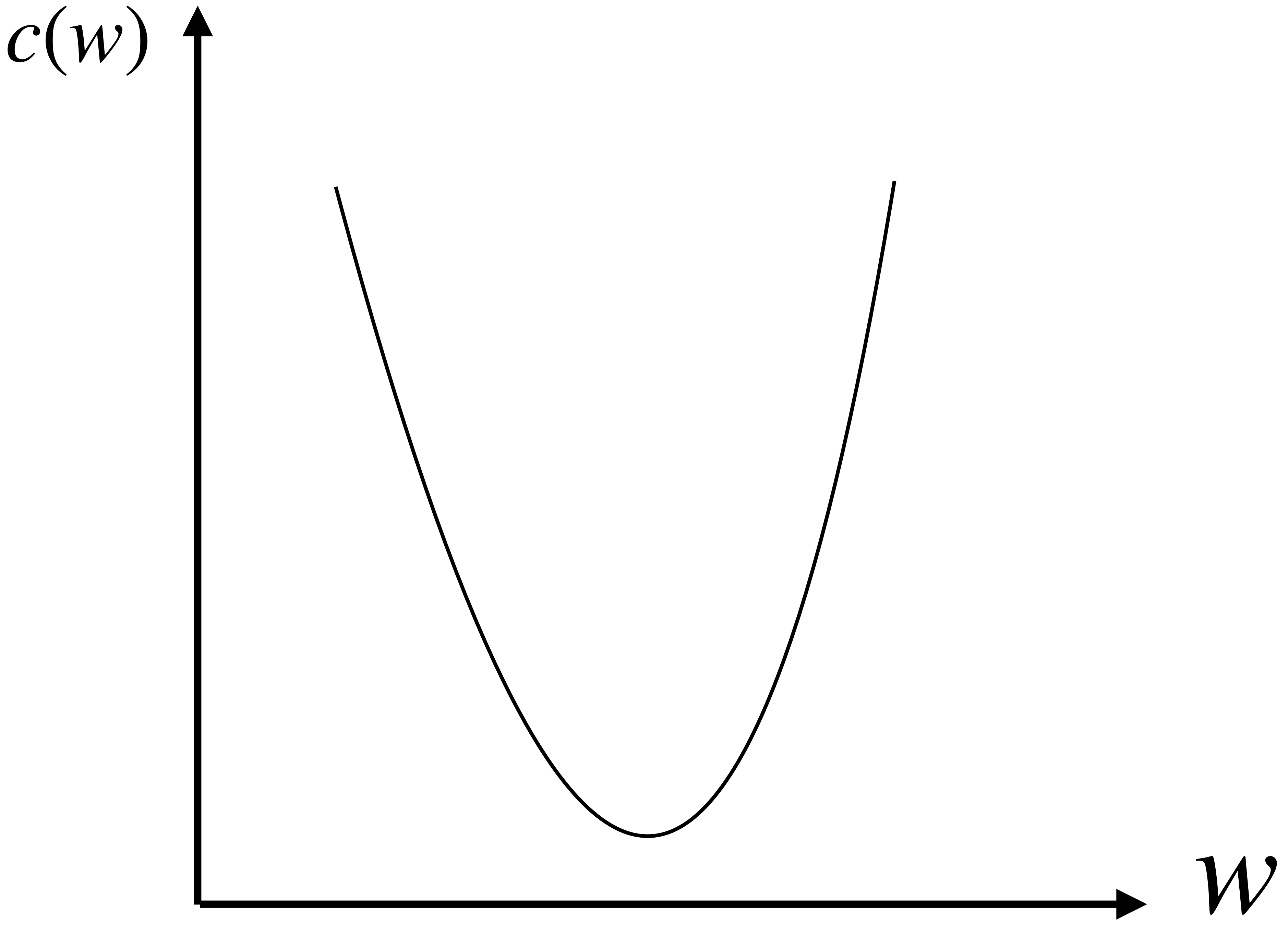
- Let's assume the simplest univariate case 1 feature per example.

Start with random w and select multiple random updates

$$w_1 = w_0 + r_1 \longrightarrow c(w_1)$$

$$w_2 = w_0 + r_2 \longrightarrow c(w_2)$$

$$c(w) = \frac{1}{n} \sum_{i=0}^{n} (x_i w - y_i)^2$$



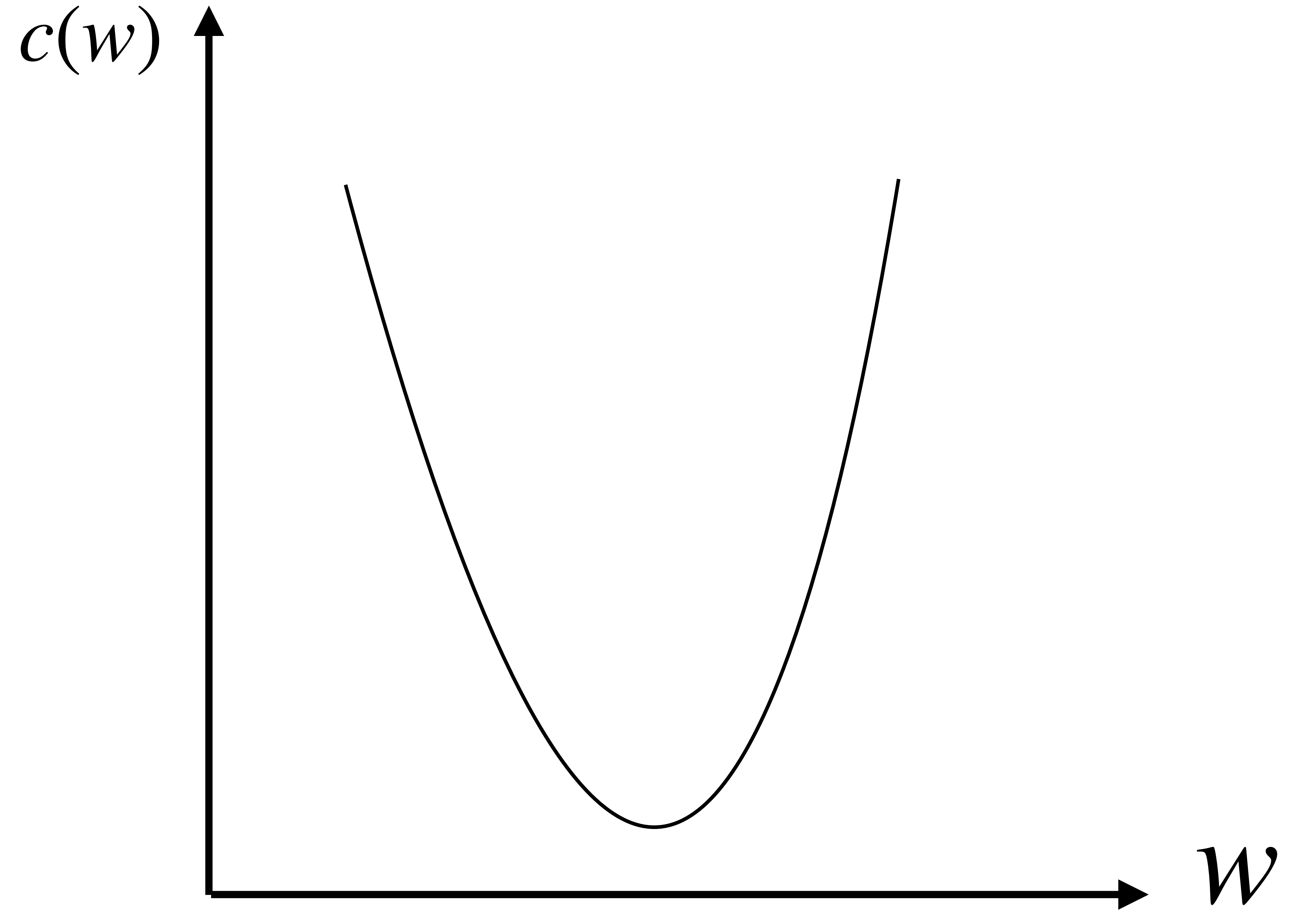
How about Random Search?

- Let's assume the simplest univariate case 1 feature per example.

Start with random w and select multiple random updates

$$w_2 = w_0 + r_2 \longrightarrow c(w_2)$$

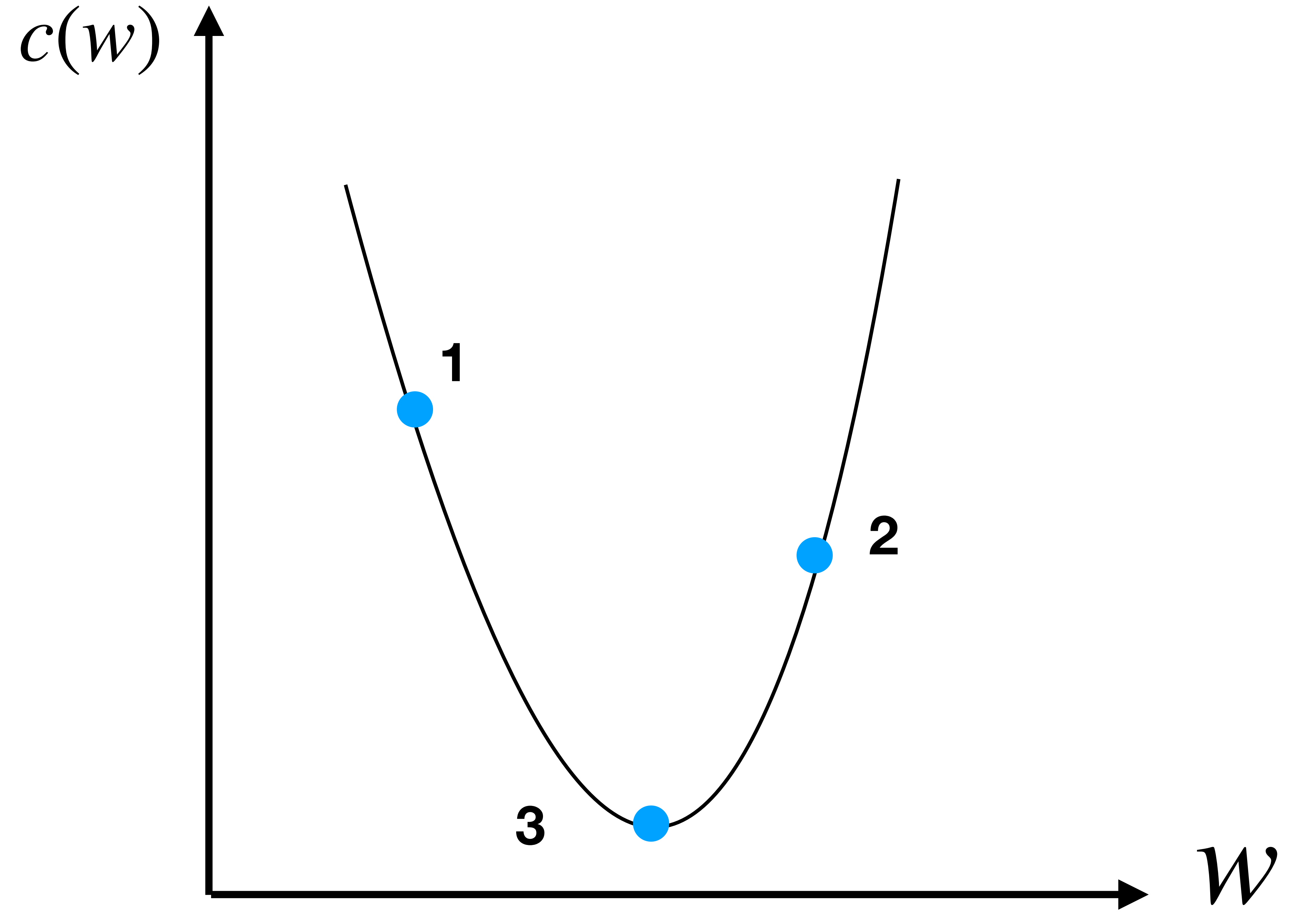
Can we do something smarter?



The smarter approach

- Let's assume the simplest univariate case 1 feature per example.

$$c(w) = \frac{1}{n}\sum_{i=0}^{n}(x_i w - y_i)^2$$



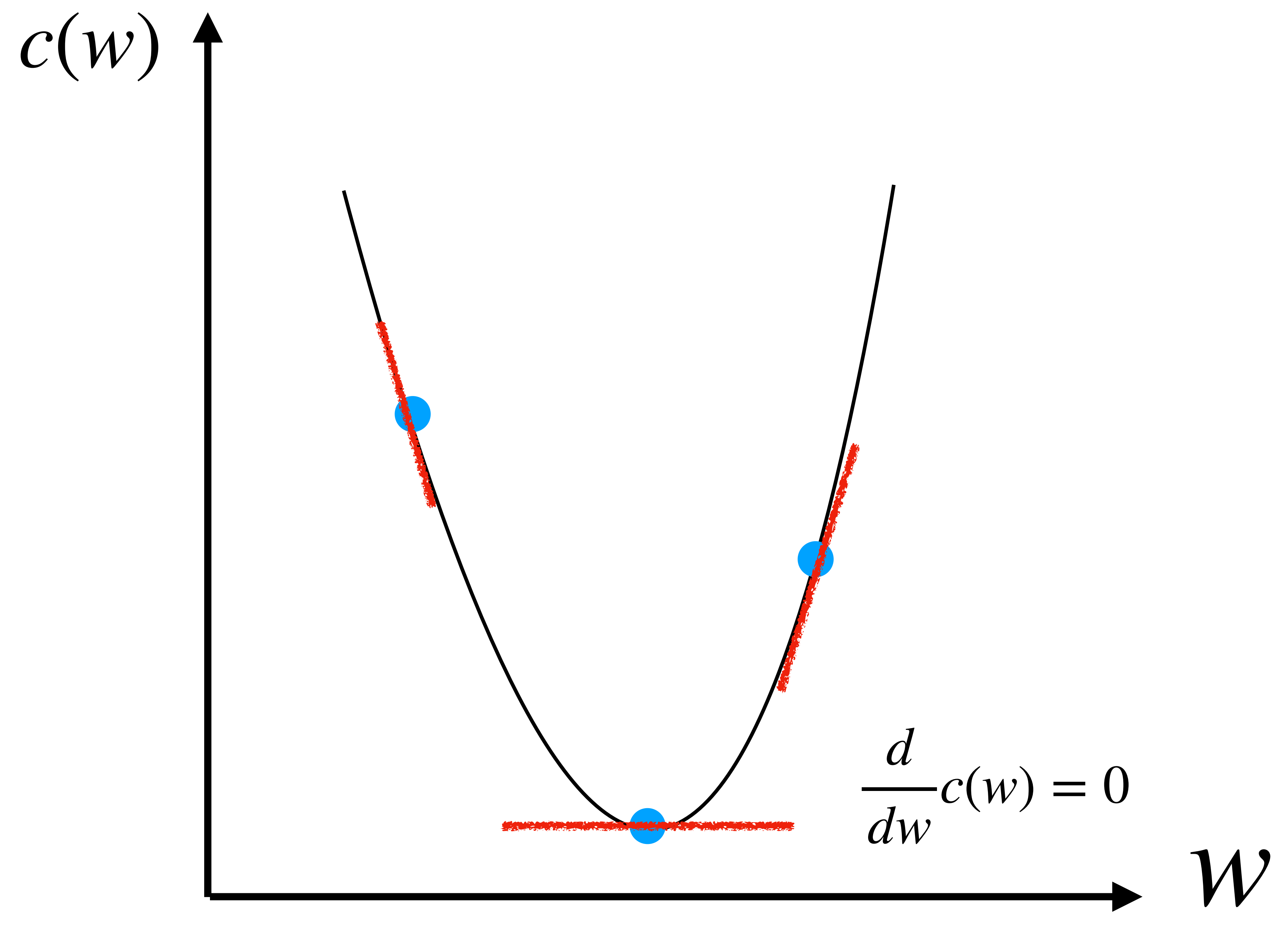
Which point has the lowest cost?

The smarter approach

- Let's assume the simplest univariate case 1 feature per example.

Use derivative!

$$c(w) = \frac{1}{n} \sum_{i=0}^{n} (x_i w - y_i)^2$$



Slope is zero

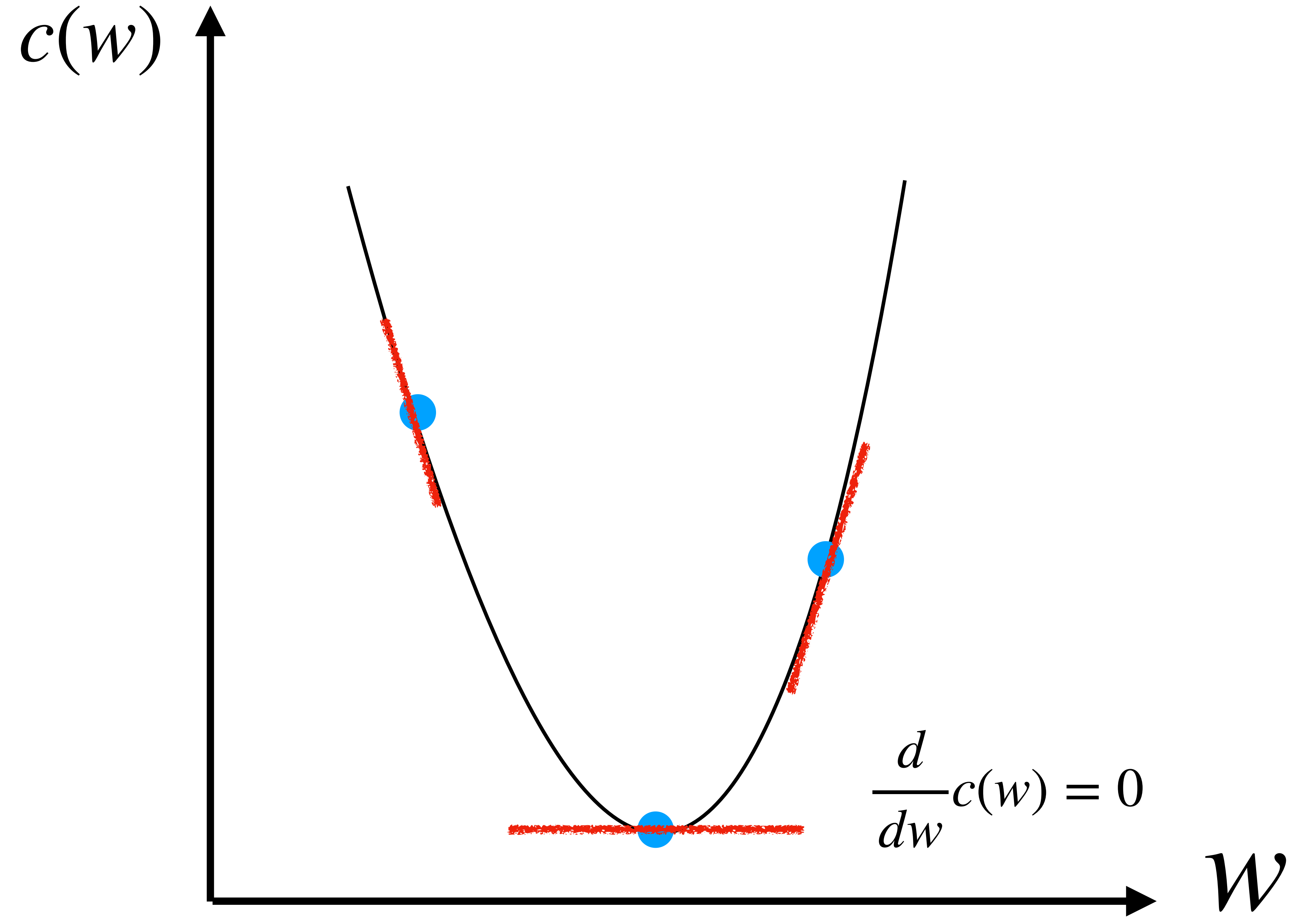
The smarter approach

- Let's assume the simplest univariate case 1 feature per example.

Use derivative!

$$c(w) = \frac{1}{n}\sum_{i=0}^{n}(x_i w - y_i)^2$$

$$g = \frac{d}{dw}c(w) = \frac{1}{n}\sum_{i=0}^{n}2x_i(x_i w - y_i)$$



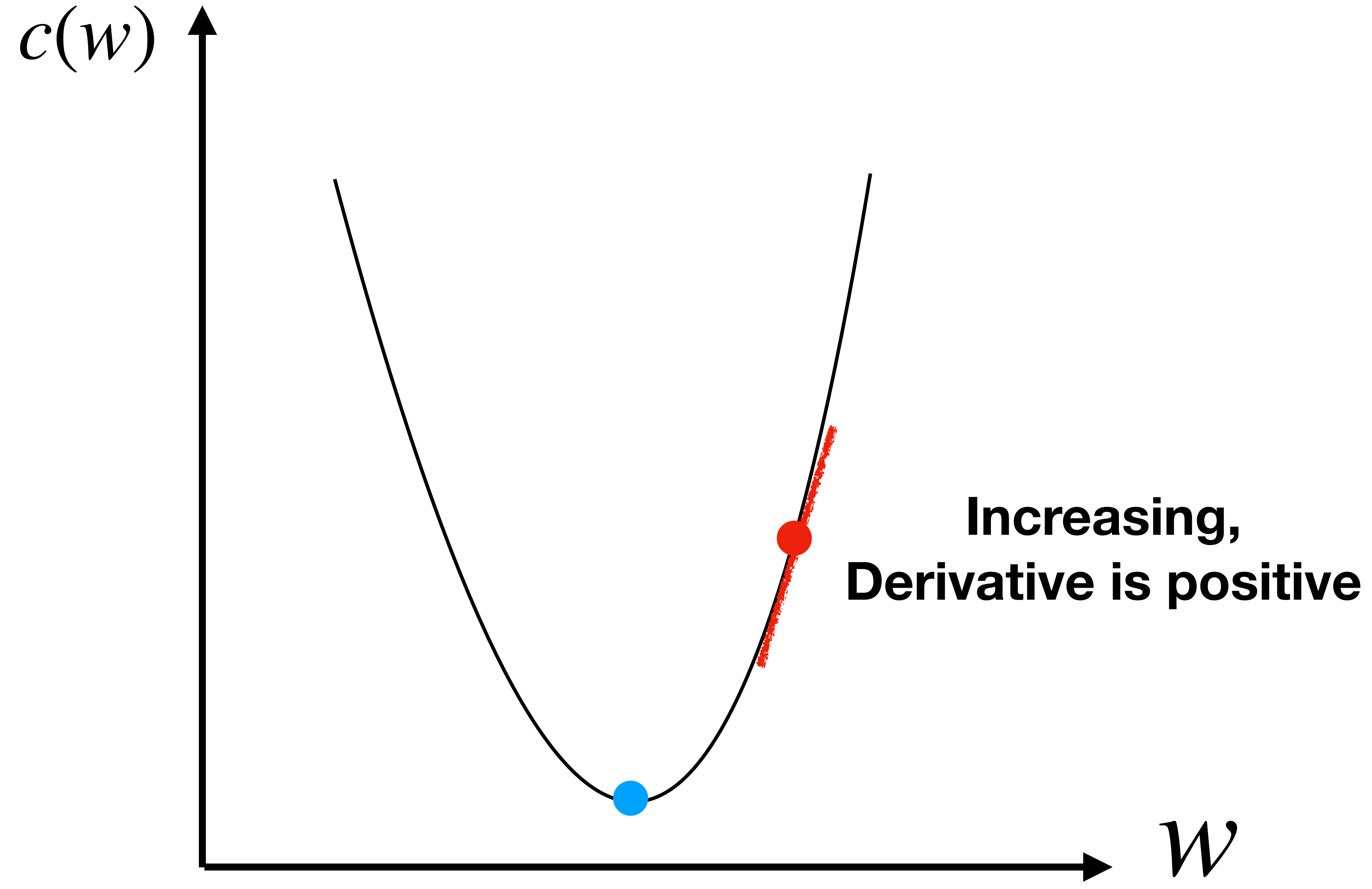
Slope is zero

The smarter approach

- Let's assume the simplest univariate case 1 feature per example.

Use derivative!

Update Function: $w_1 = w_0 - g$

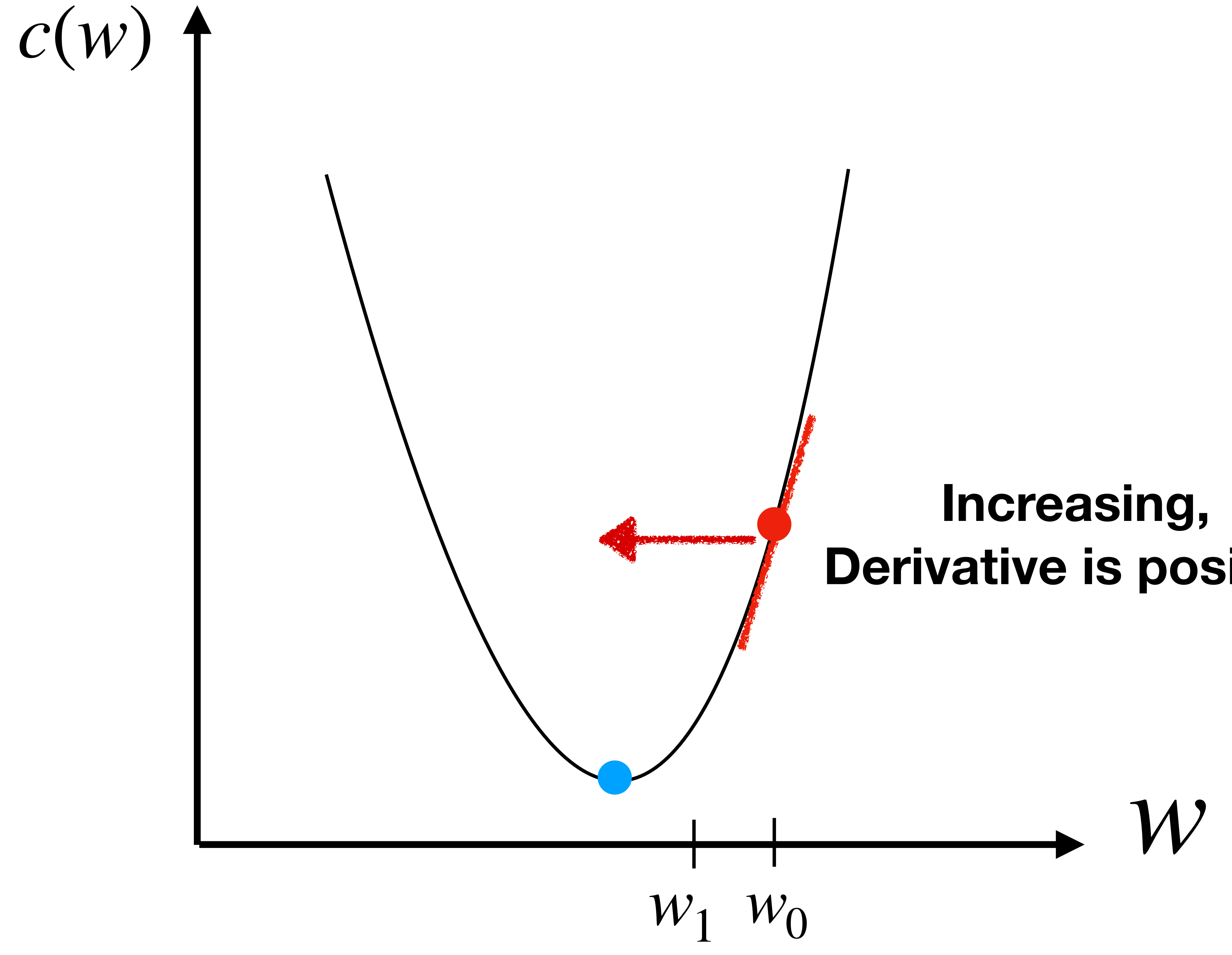


The smarter approach

- Let's assume the simplest univariate case 1 feature per example.

Use derivative!

Update Function: $w_1 = w_0 - g$

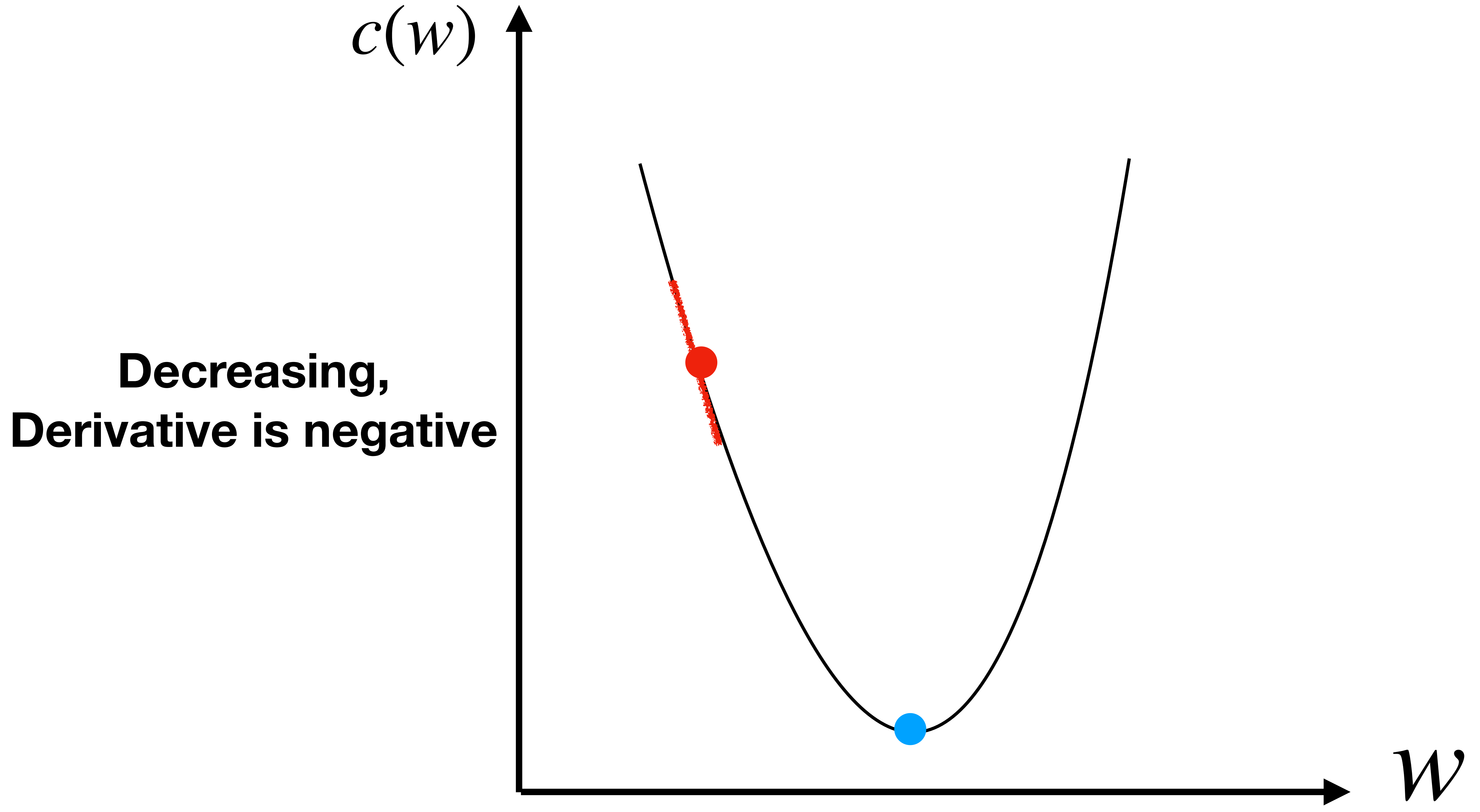


The smarter approach

- Let's assume the simplest univariate case 1 feature per example.

Use derivative!

Update Function: $w_1 = w_0 - g$

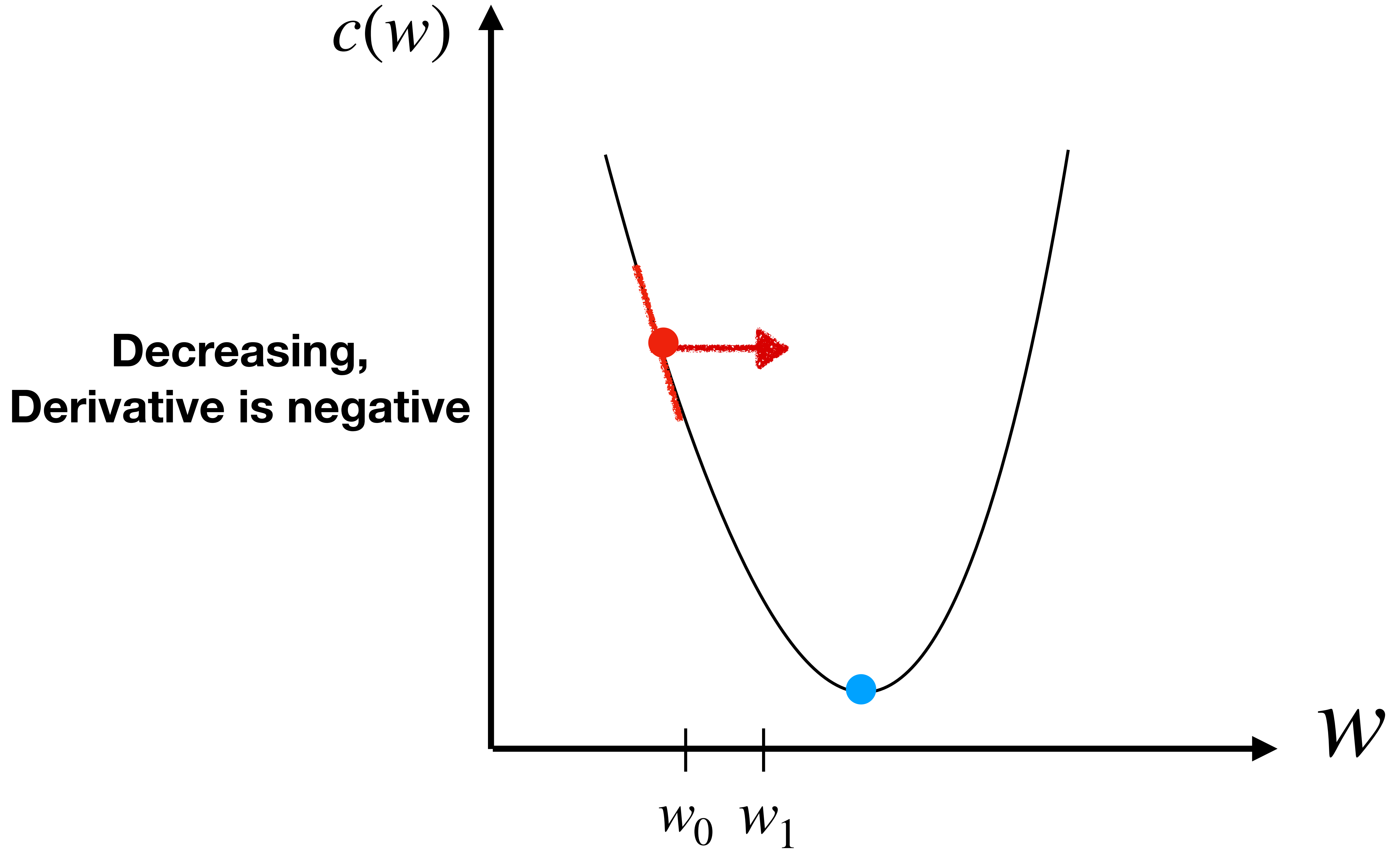


The smarter approach

- Let's assume the simplest univariate case 1 feature per example.

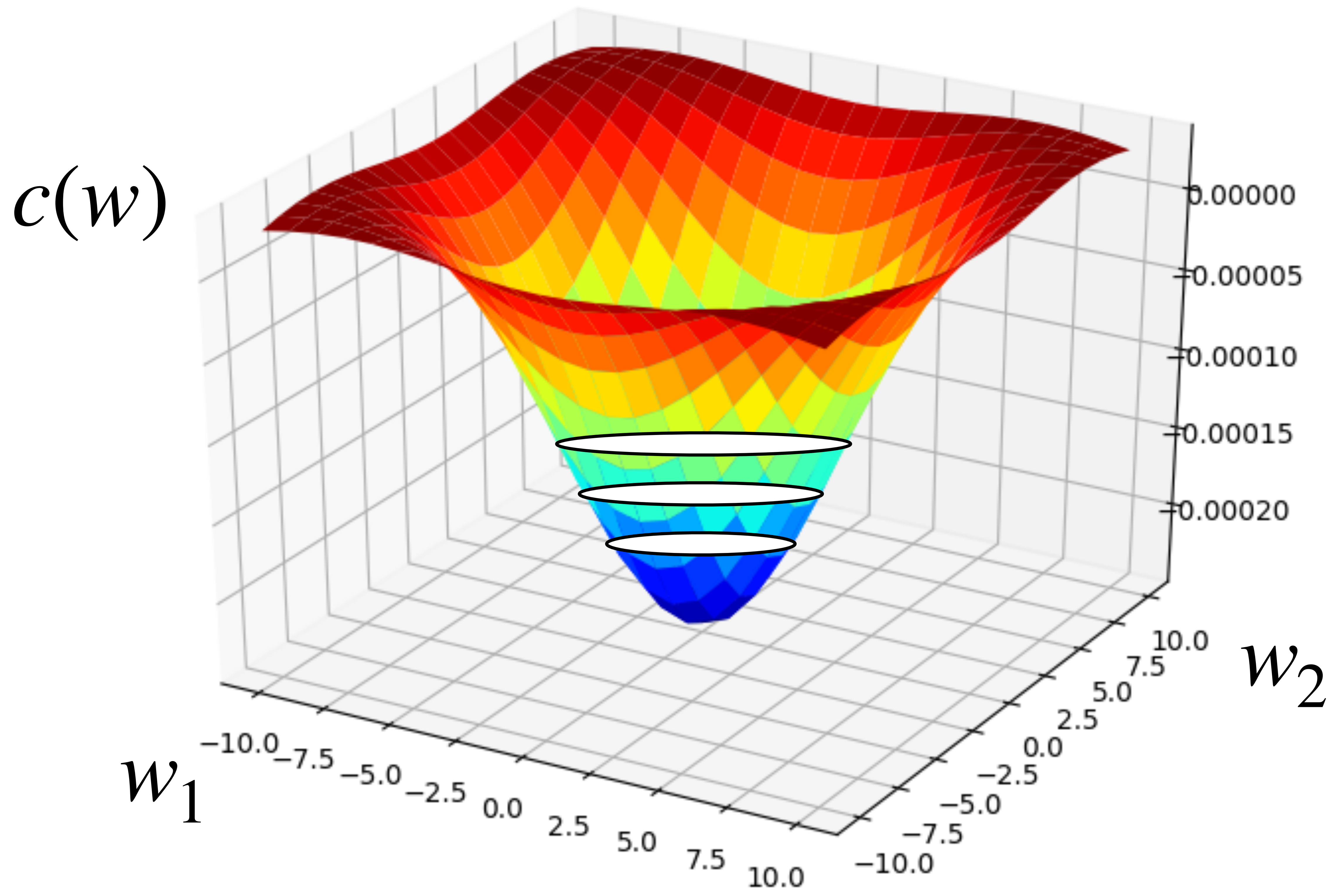
Use derivative!

Update Function: $w_1 = w_0 - g$



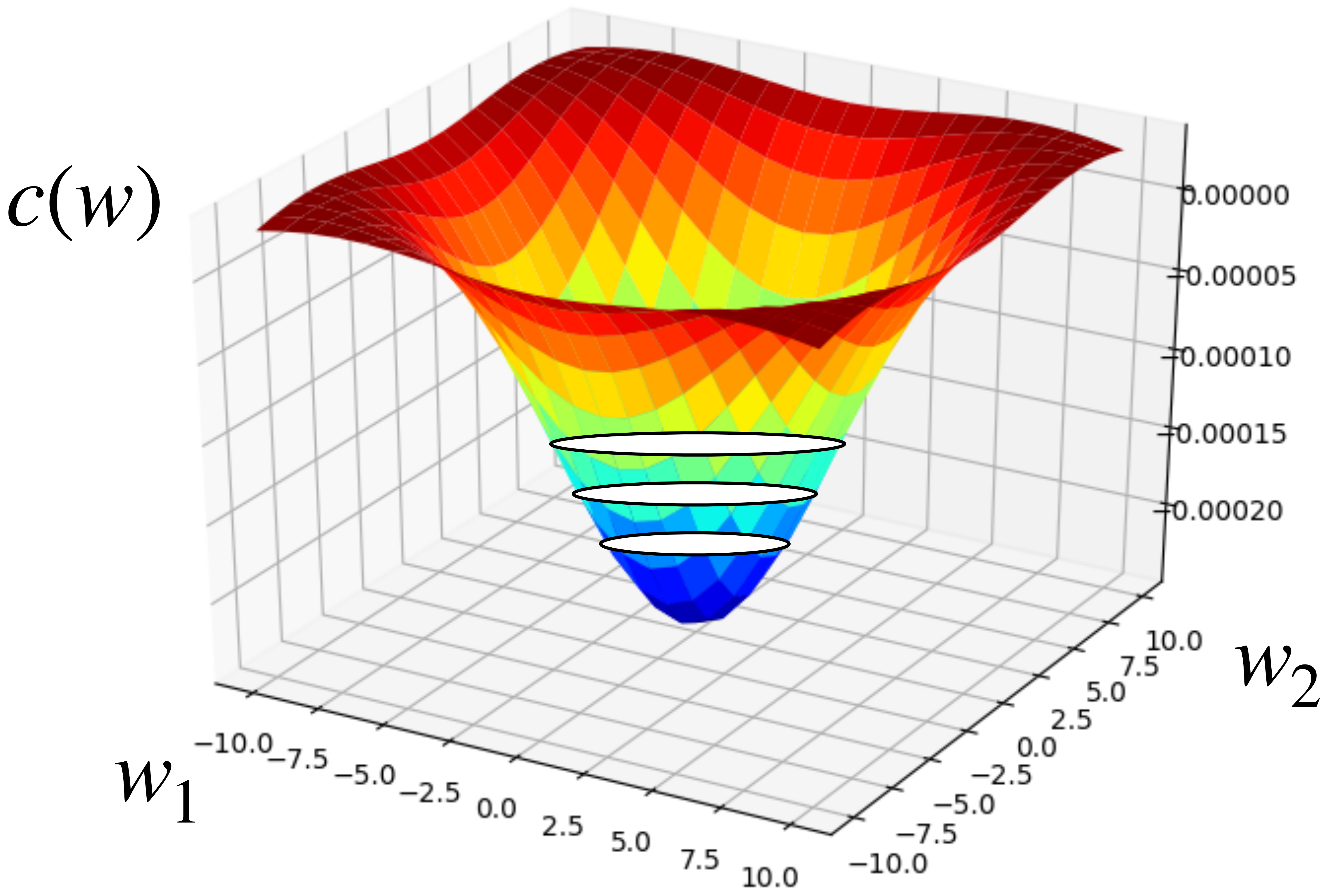
Multivariate Case

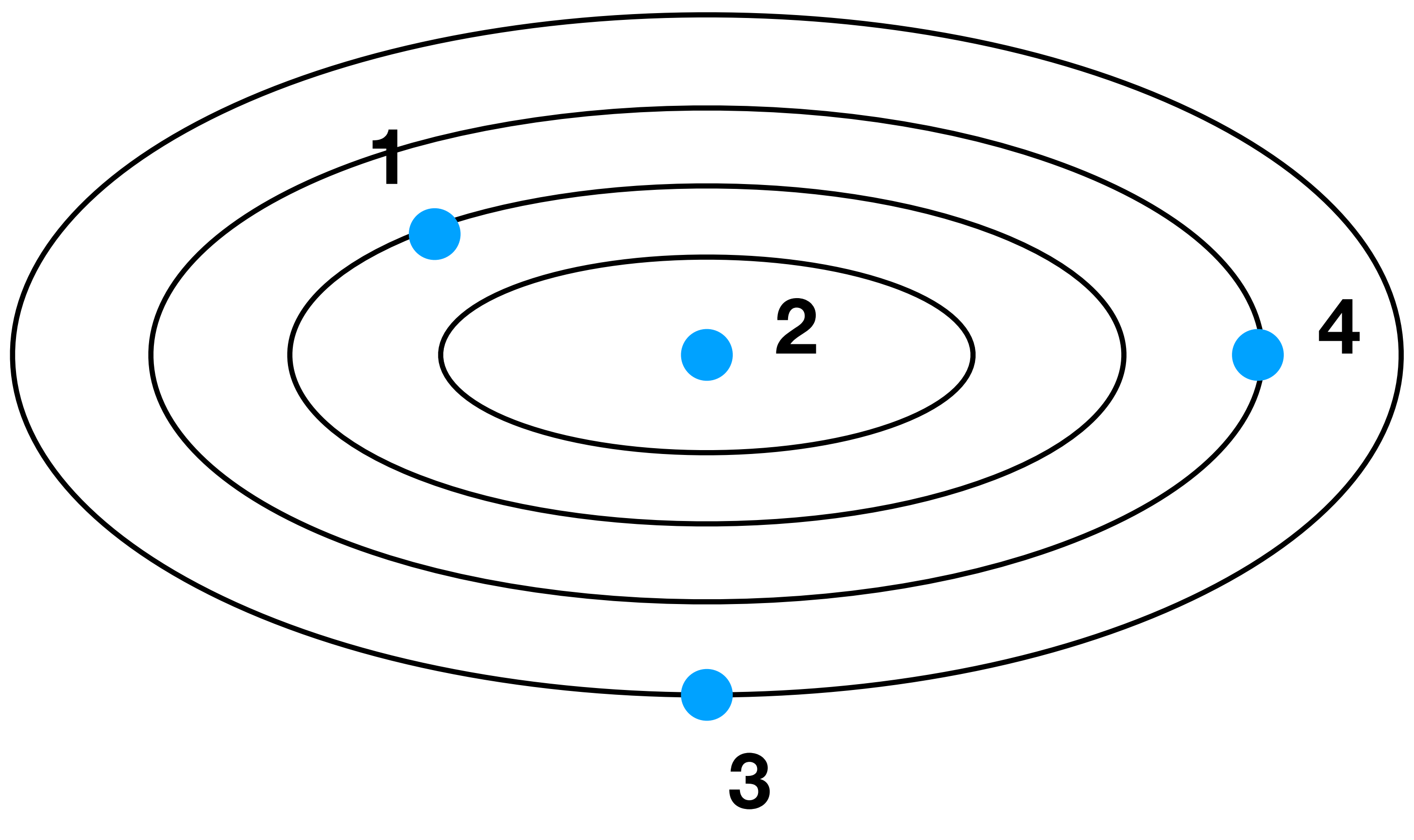
- Let's go to the multivariate case, I have 2 features per example.



Contour Plots

- Let’s go to the multivariate case, I have 2 features per example.





Which point has the lowest cost?

Multivariate Case

- Now we have gradients!

$$c(w) = \frac{1}{n} \sum_{i=0}^{n} (x_i^\top w - y_i)^2$$

$$\nabla c(w) = \begin{pmatrix} \frac{\partial c(w)}{\partial w_1} \\ \frac{\partial c(w)}{\partial w_2} \end{pmatrix}$$

Gradient Descent

- This is how it looks in the contour plot.

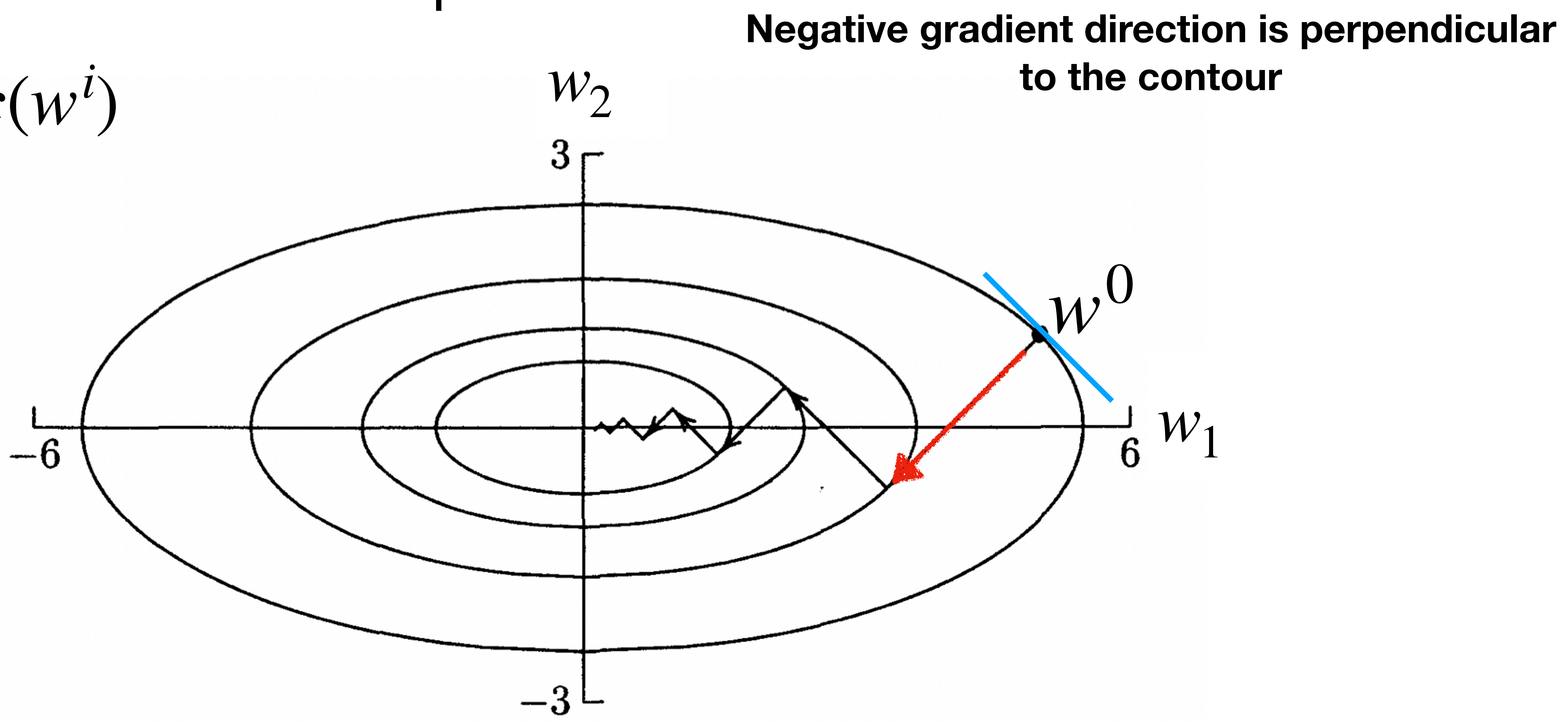
$$w^{i+1} = w^{i} - \nabla c(w^{i})$$

What are the shapes of the vectors above?

Gradient Descent

- This is how it looks in the contour plot.

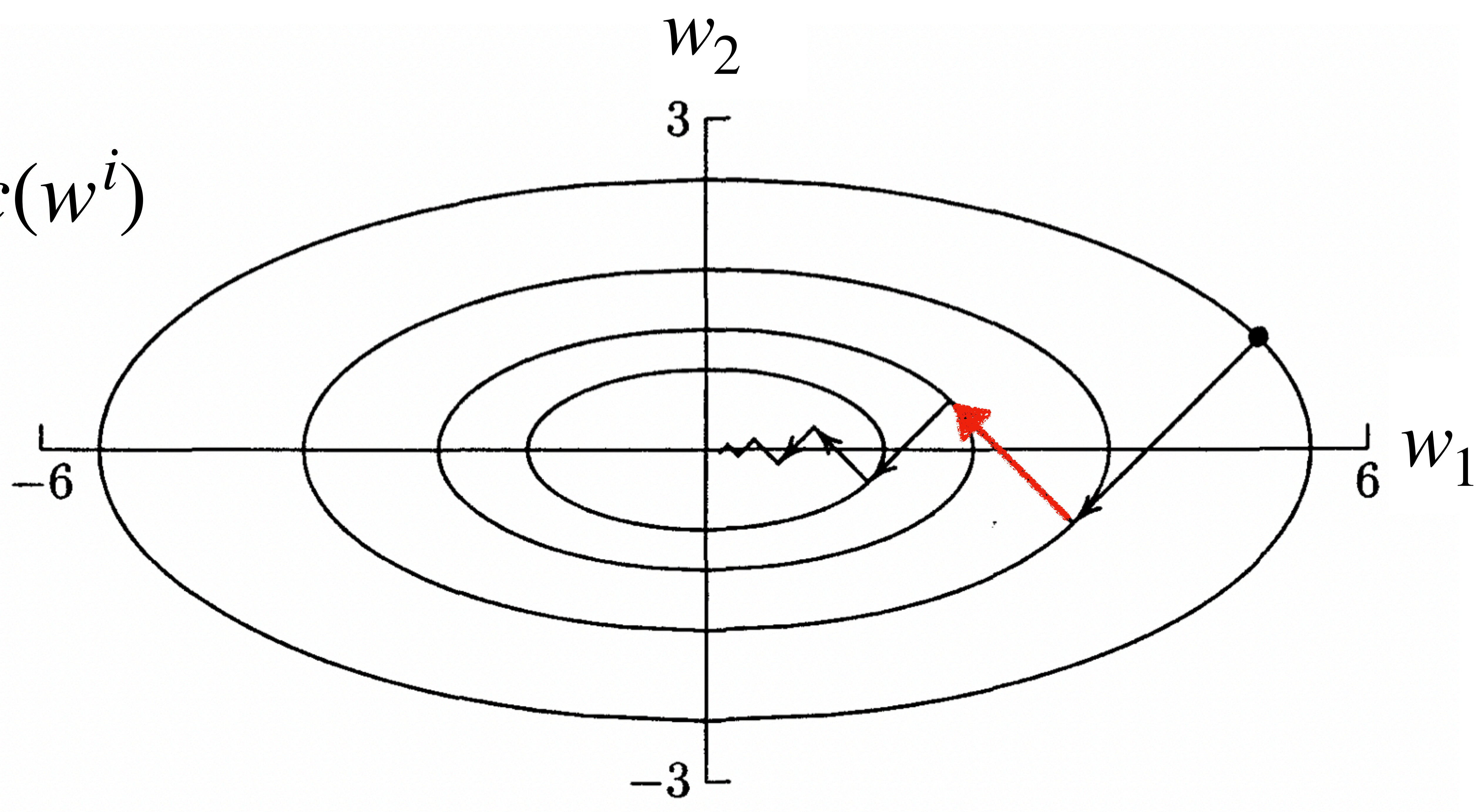
$$w^{i+1} = w^i - \nabla c(w^i)$$



Gradient Descent

- This is how it looks in the contour plot.

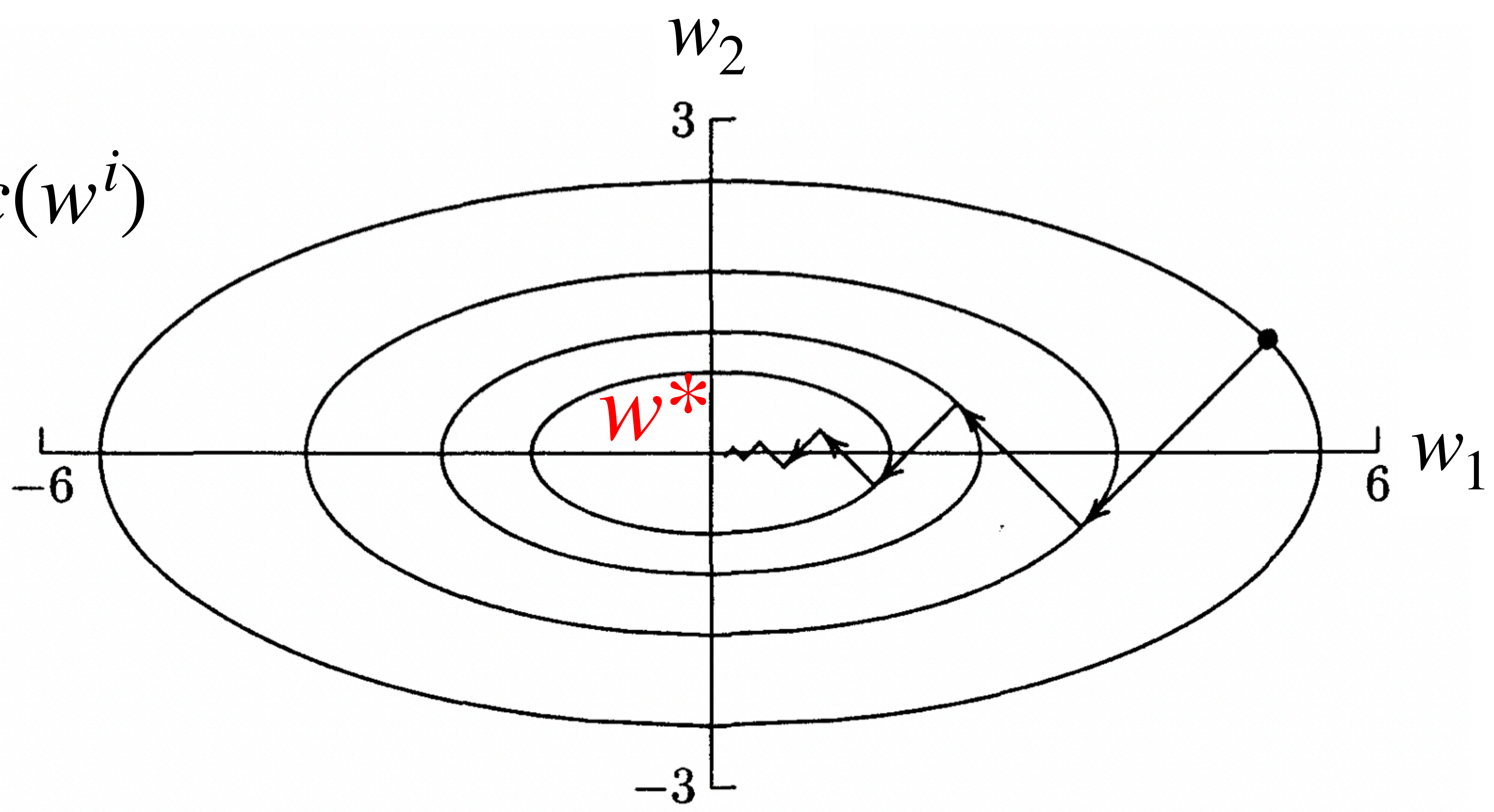
$$w^{i+1} = w^{i} - \nabla c(w^{i})$$



Gradient Descent

- This is how it looks in the contour plot.

$$w^{i+1} = w^{i} - \nabla c(w^{i})$$



Learning Rate

- Can we control our update!

$$w^{i+1} = w^i - \alpha \nabla c(w^i)$$

Learning rate

Gradient Descent

- Algorithm simply:

Initial guess w^o

For i=0, 1, …, M

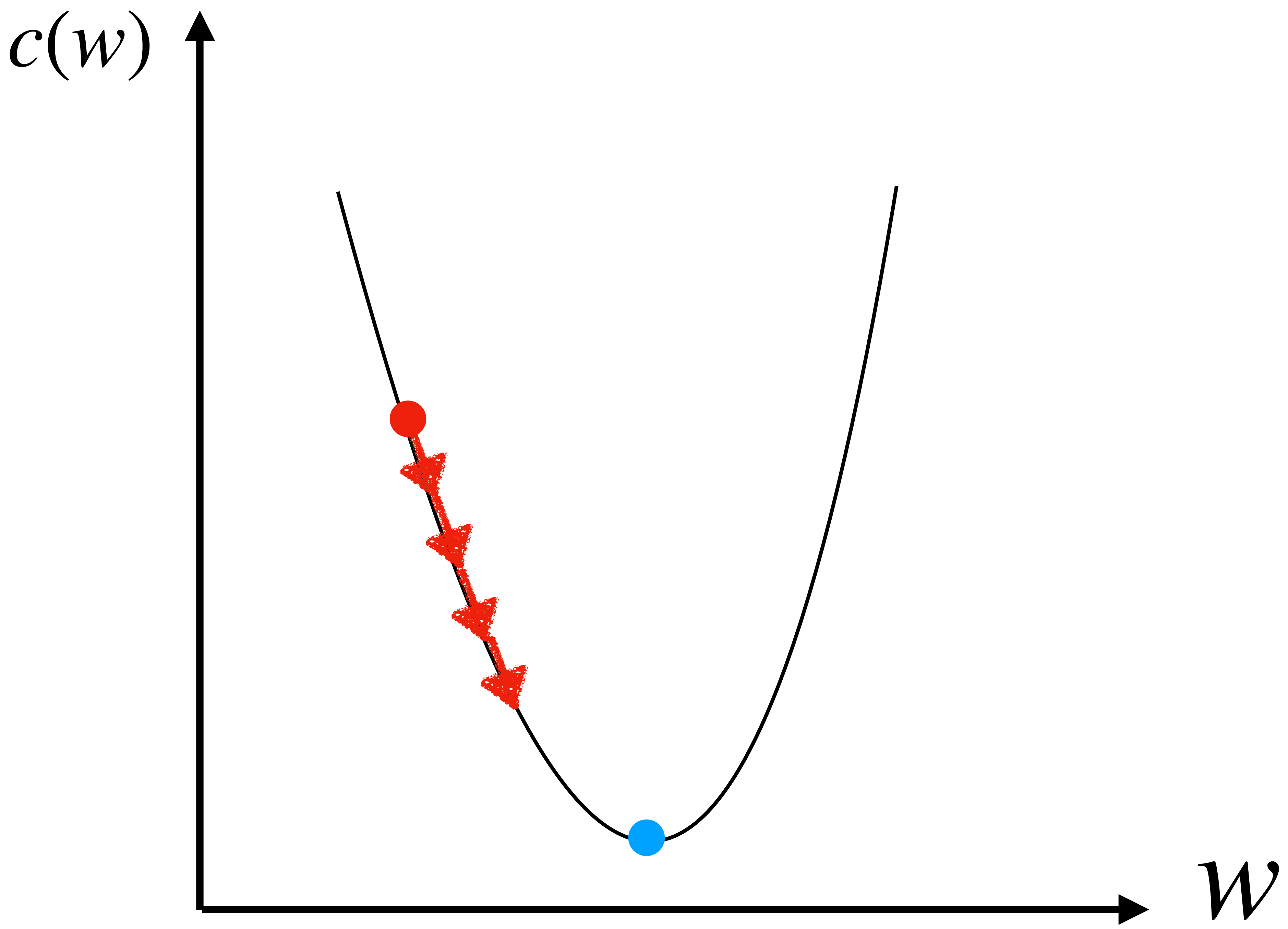
$$w^{i+1} = w^i - \alpha \nabla c(w^i)$$

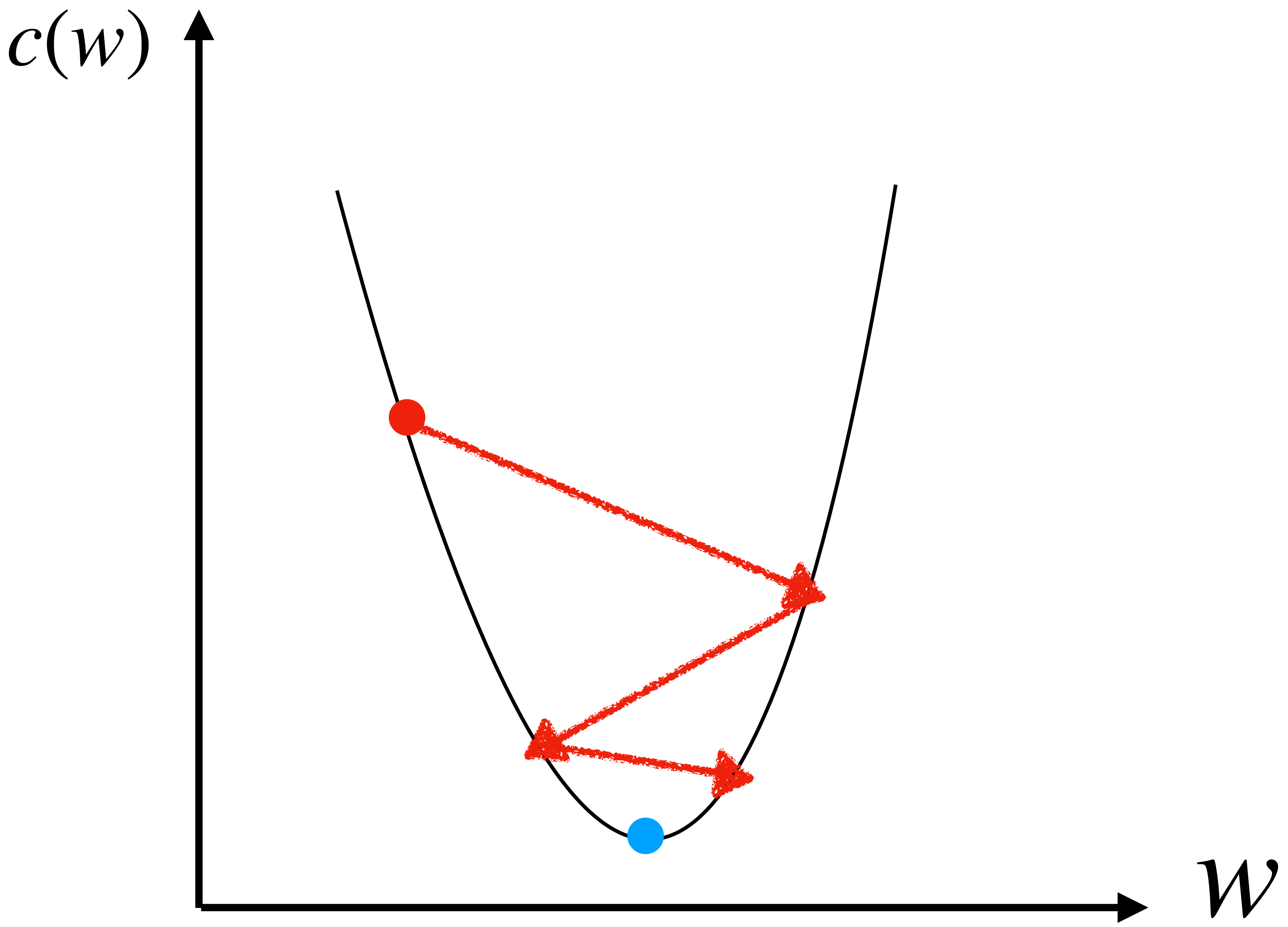
End

Learning Rate Choice

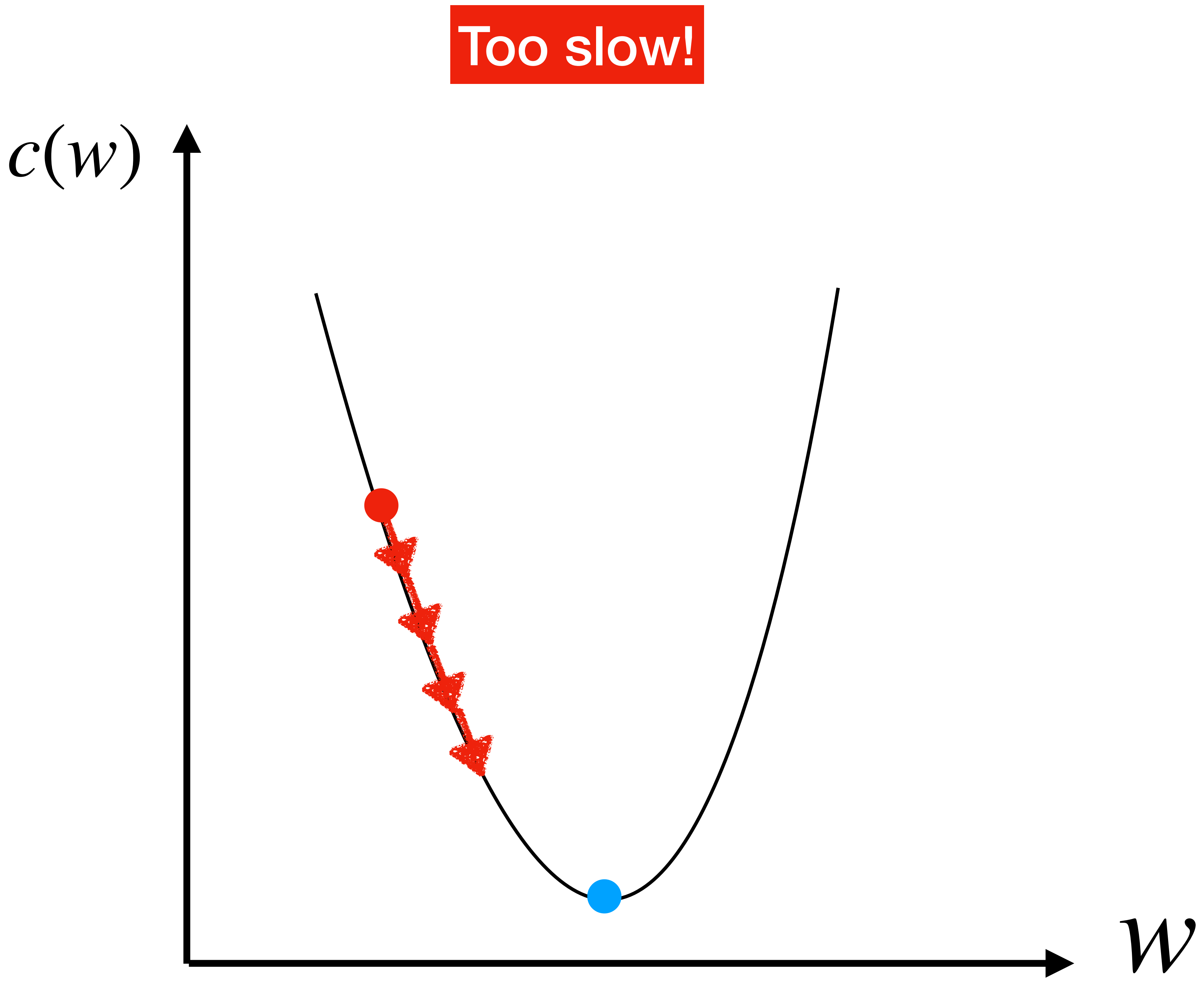
Which one is using a higher learning rate?

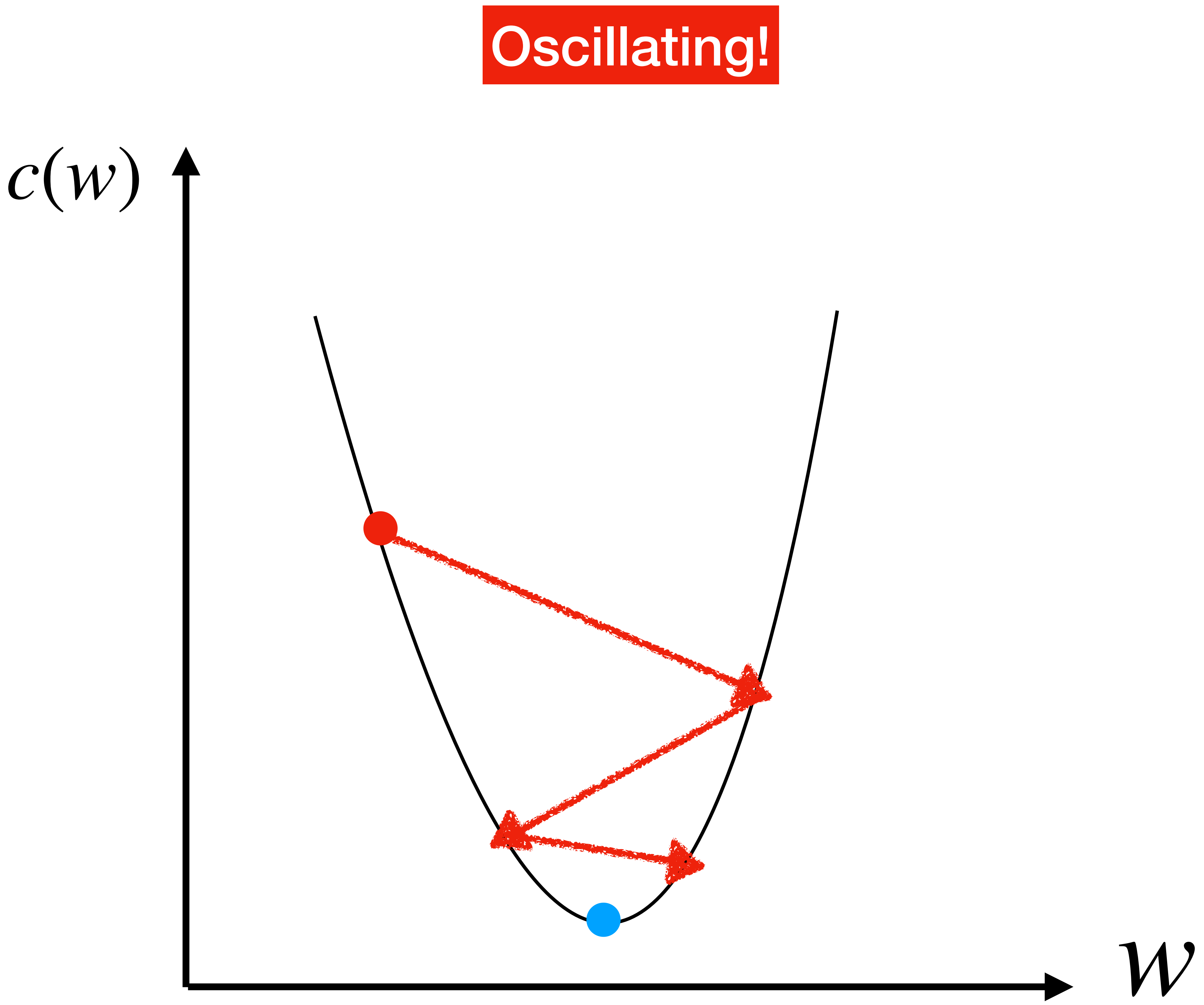
Which one is slower to converge?





Learning Rate Choice





Exact Line Search

- Algorithm simply:

How far should we go then?
How to choose α ?

Initial guess w^o

For i=0, 1, …, M

Compute $\nabla c(w^i)$

Compute $\alpha^i = \textbf{argmin}_\alpha \{c(w^i - \alpha \nabla c(w^i))\}$

$w^{i+1} = w^i - \alpha^i \nabla c(w^i)$

End

Numerical Example

- Let's try a simple example:

$$f(x) = \frac{1}{2}x_1^2 + \frac{5}{2}x_2^2$$

Gradient?

Numerical Example

- Let's try a simple example:

$$f(x) = \frac{1}{2}x_1^2 + \frac{5}{2}x_2^2$$

Gradient:

$$\nabla f(x) = \begin{pmatrix} x_1 \\ 5x_2 \end{pmatrix}$$

Numerical Example

- Let's try a simple example:

$$f(x) = \frac{1}{2}x_1^2 + \frac{5}{2}x_2^2 \qquad \nabla f(x) = \begin{pmatrix} x_1 \\ 5x_2 \end{pmatrix}$$

Initial guess: $$x^0 = \begin{pmatrix} 5 \\ 1 \end{pmatrix} \qquad \nabla f(x^0) = \begin{pmatrix} 5 \\ 5 \end{pmatrix}$$

Numerical Example

- Let's try a simple example:

$$f(x) = \frac{1}{2}x_1^2 + \frac{5}{2}x_2^2 \qquad \nabla f(x) = \begin{pmatrix} x_1 \\ 5x_2 \end{pmatrix}$$

$$x^0 = \begin{pmatrix} 5 \\ 1 \end{pmatrix} \qquad \nabla f(x^0) = \begin{pmatrix} 5 \\ 5 \end{pmatrix}$$

$$\alpha^i = \textbf{argmin}_\alpha \{ f(x^0 - \alpha \nabla f(x^0)) \}$$

How do you min. this function?

Numerical Example

- Let's try a simple example:

$$f(x) = \frac{1}{2}x_1^2 + \frac{5}{2}x_2^2 \qquad \nabla f(x) = \begin{pmatrix} x_1 \\ 5x_2 \end{pmatrix}$$

$$x^0 = \begin{pmatrix} 5 \\ 1 \end{pmatrix} \qquad \nabla f(x^0) = \begin{pmatrix} 5 \\ 5 \end{pmatrix}$$

ELS:

$$\frac{d}{d\alpha} f(x^0 - \alpha \nabla f(x^0)) = 0$$

Numerical Example

- Let's try a simple example:

$$f(x) = \frac{1}{2}x_1^2 + \frac{5}{2}x_2^2 \qquad \nabla f(x) = \begin{pmatrix} x_1 \\ 5x_2 \end{pmatrix}$$

$$x^0 = \begin{pmatrix} 5 \\ 1 \end{pmatrix} \qquad \nabla f(x^0) = \begin{pmatrix} 5 \\ 5 \end{pmatrix}$$

ELS:

$$\alpha = \frac{1}{3}$$

Numerical Example

- Let's try a simple example:

$$x^0 = \begin{pmatrix} 5 \\ 1 \end{pmatrix} \qquad \nabla f(x^0) = \begin{pmatrix} 5 \\ 5 \end{pmatrix}$$

Final update:

$$x^1 = x^0 - \alpha \nabla f(x^0) = \begin{pmatrix} 5 \\ 1 \end{pmatrix} - \frac{1}{3} \begin{pmatrix} 5 \\ 5 \end{pmatrix}$$

Numerical Example

- Let's try a simple example:

TextBook Example 6.11

Batch vs Stochastic Gradient Descent

- Algorithm simply:

Batch Gradient Descent

Initial guess w^o

For i=0, 1, …, M

$$w^{i+1} = w^i - \alpha \frac{1}{N} \sum_{j=1}^{N} \nabla c_j(w^i)$$

End

Batch vs Stochastic Gradient Descent

- Algorithm simply:

Stochastic Gradient Descent

Initial guess w^o

For i=0, 1, …, M

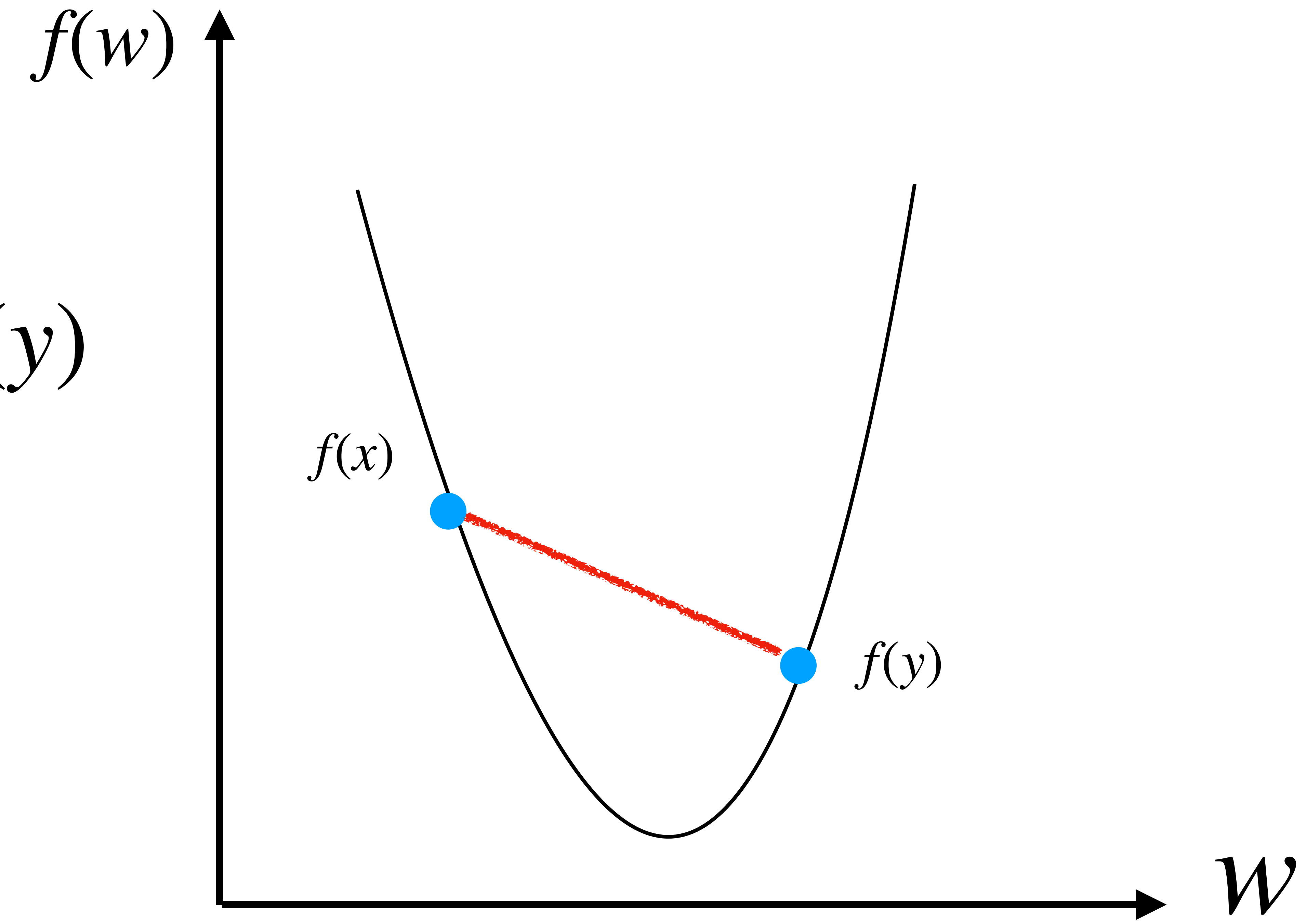
$$w^{i+1} = w^i - \alpha \nabla c_{j_i}(w^i) \qquad j_i \in \{1, \cdots, N\}$$

End

What is Convexity?

$$f(\alpha x + (1 - \alpha)y) \leq \alpha f(x) + (1 - \alpha)f(y)$$

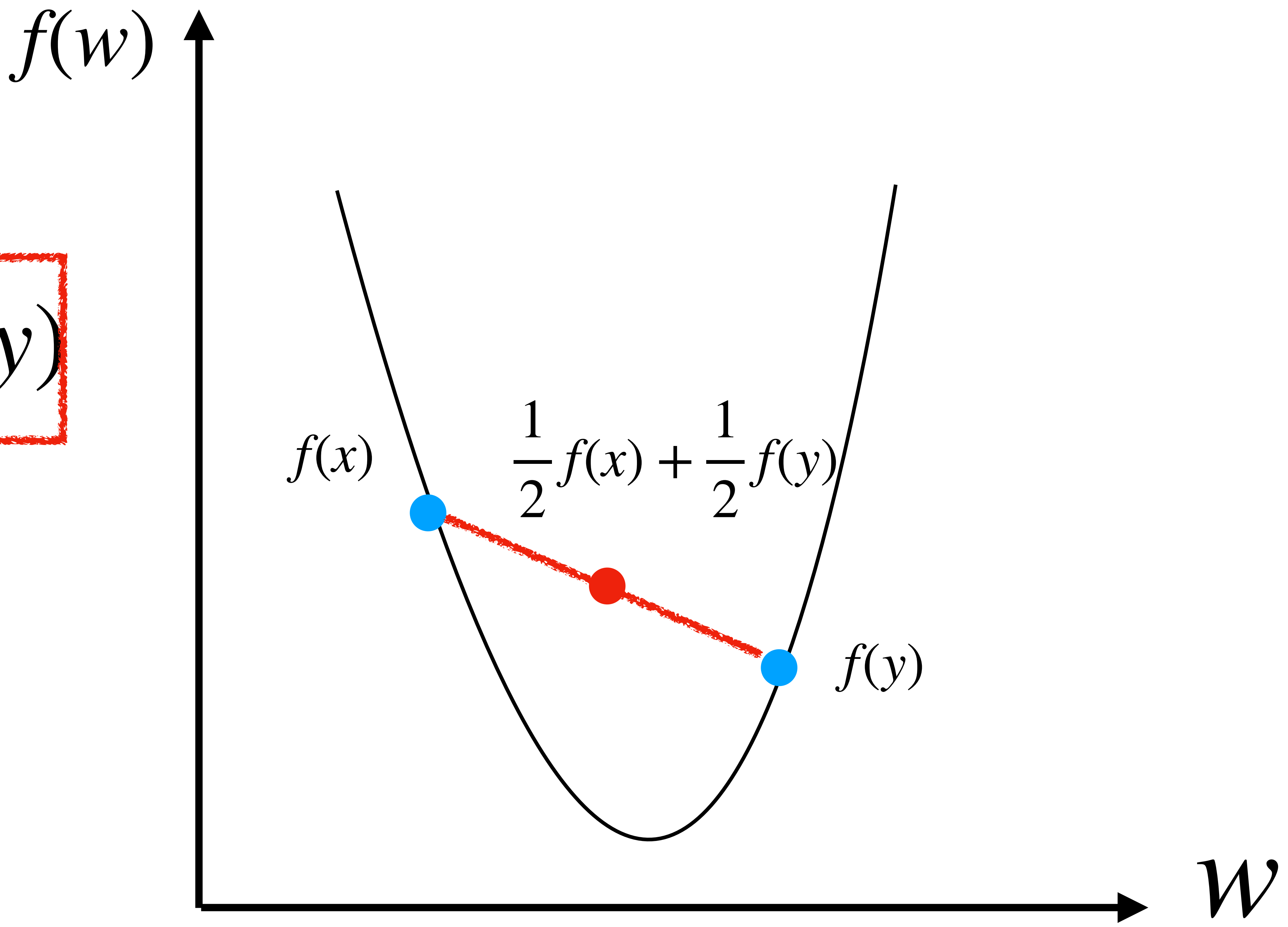
For all $\alpha \in [0,1]$



What is Convexity?

$$f(\alpha x + (1-\alpha)y) \leq \alpha f(x) + (1-\alpha) f(y)$$

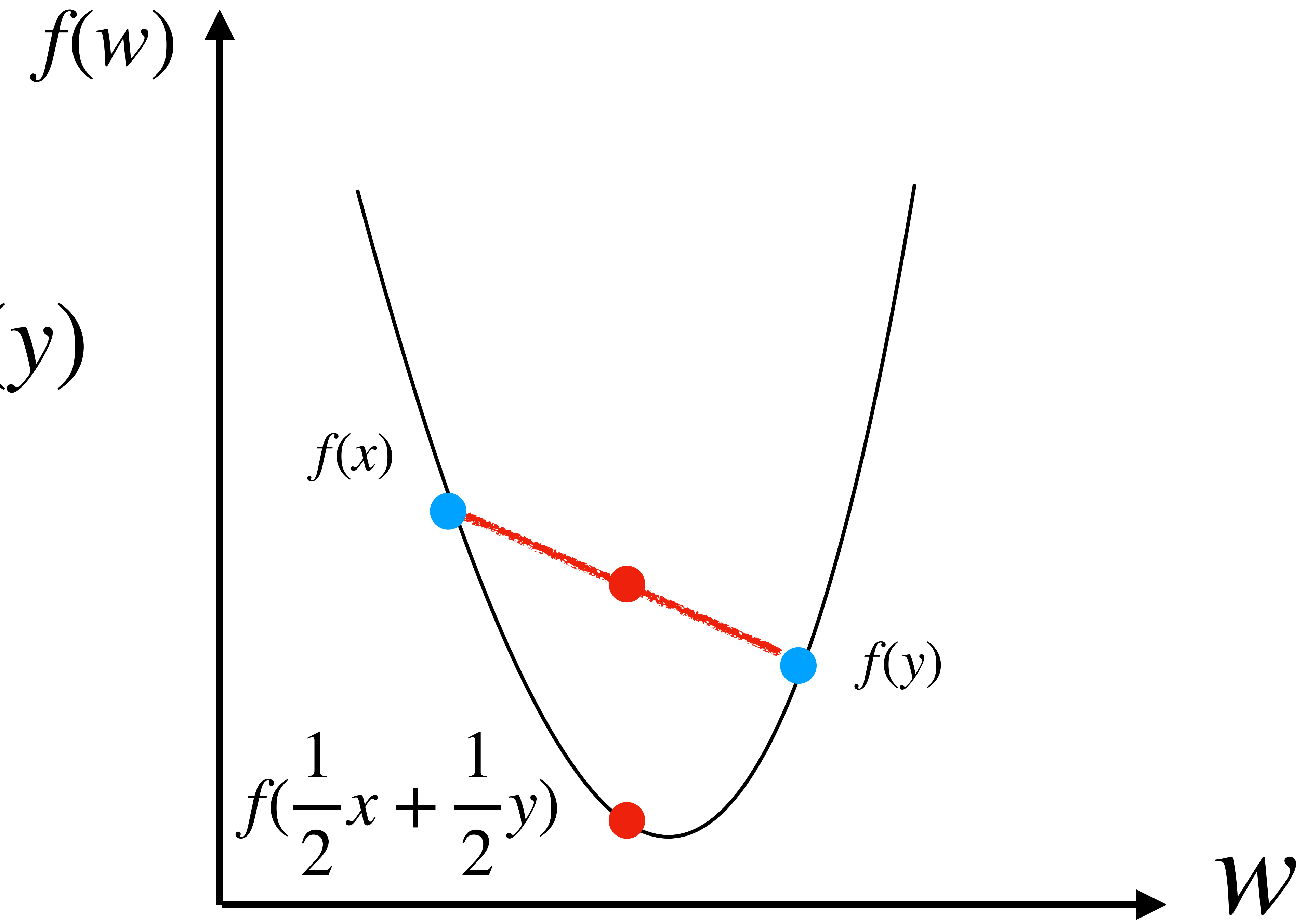
For all $\alpha \in [0,1]$



What is Convexity?

$$f(\alpha x + (1 - \alpha)y) \leq \alpha f(x) + (1 - \alpha)f(y)$$

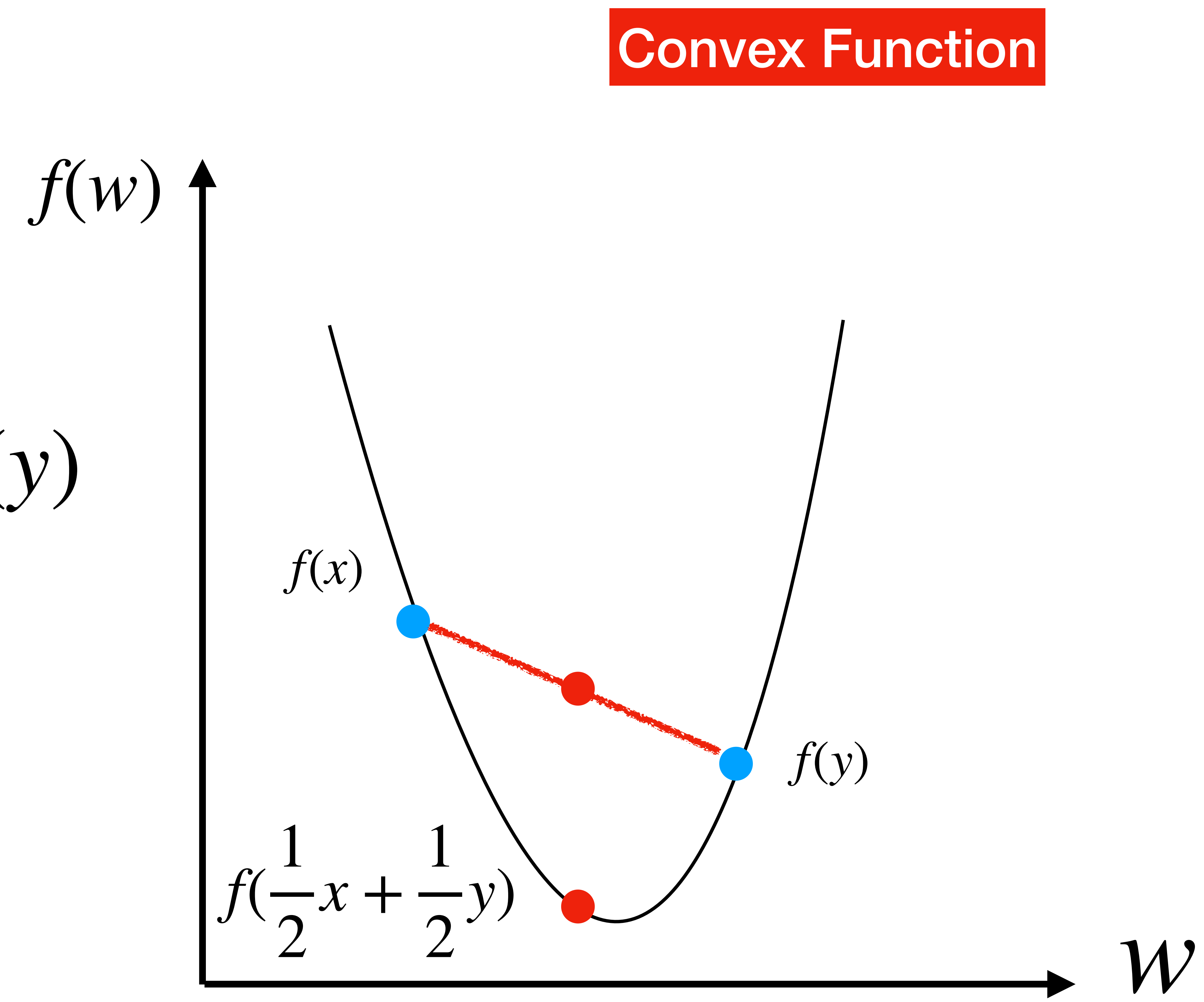
For all $\alpha \in [0,1]$



What is Convexity?

$$f(\alpha x + (1-\alpha)y) \leq \alpha f(x) + (1-\alpha)f(y)$$

For all $\alpha \in [0,1]$

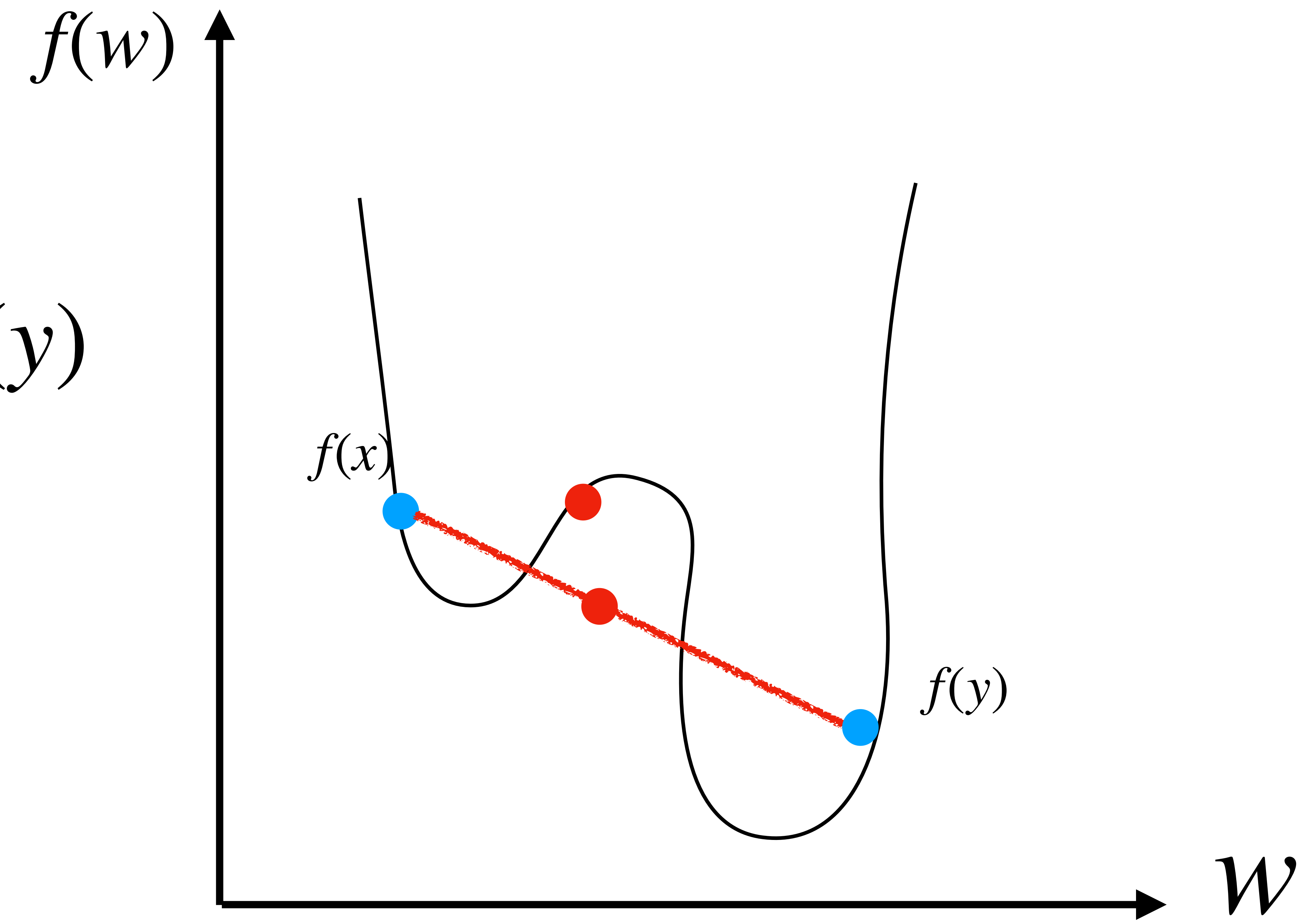


What is Convexity?

$$f(\alpha x + (1-\alpha)y) \leq \alpha f(x) + (1-\alpha)f(y)$$

For all $\alpha \in [0,1]$

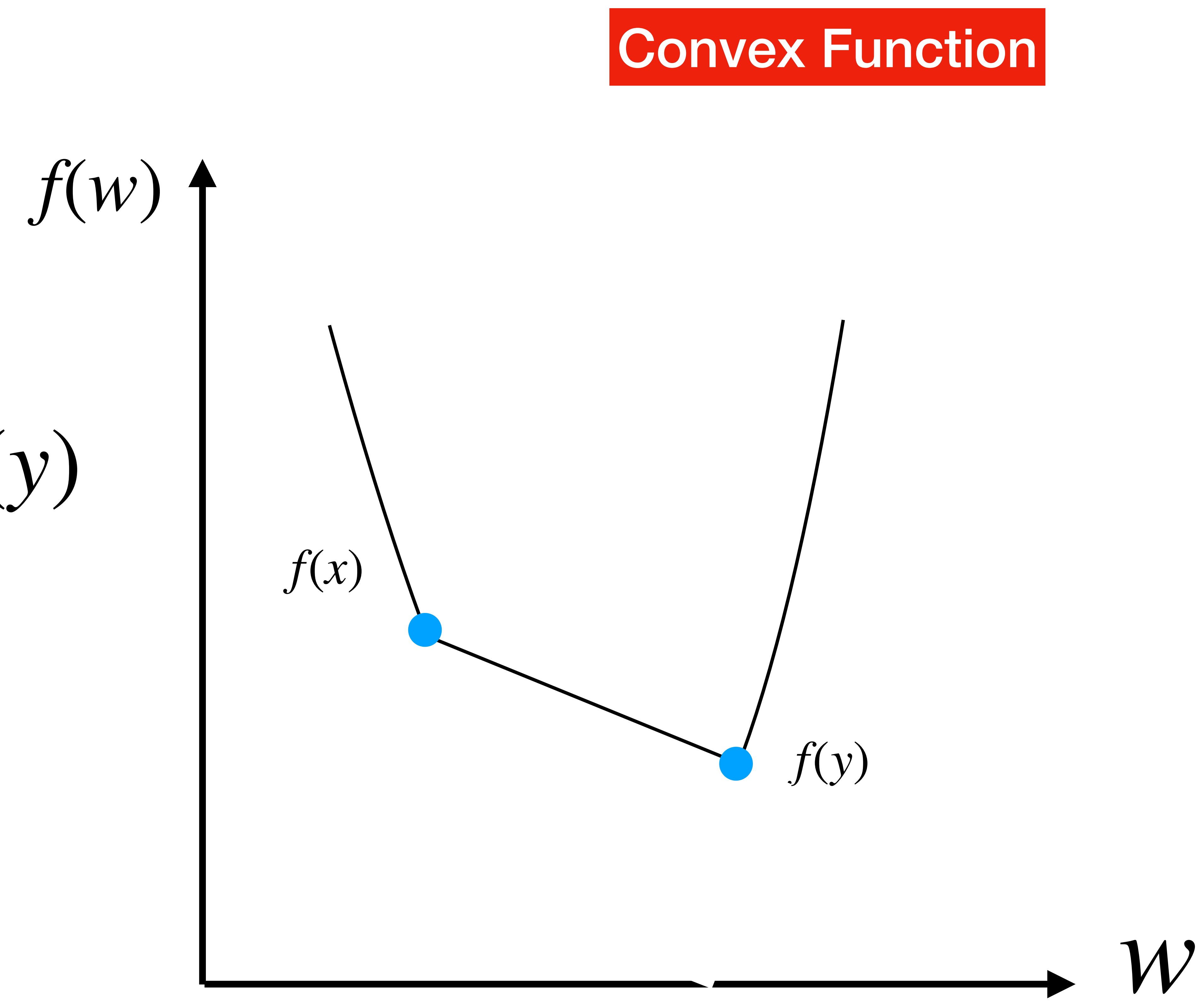
non-Convex Function



What is Convexity?

$$f(\alpha x + (1-\alpha)y) \leq \alpha f(x) + (1-\alpha) f(y)$$

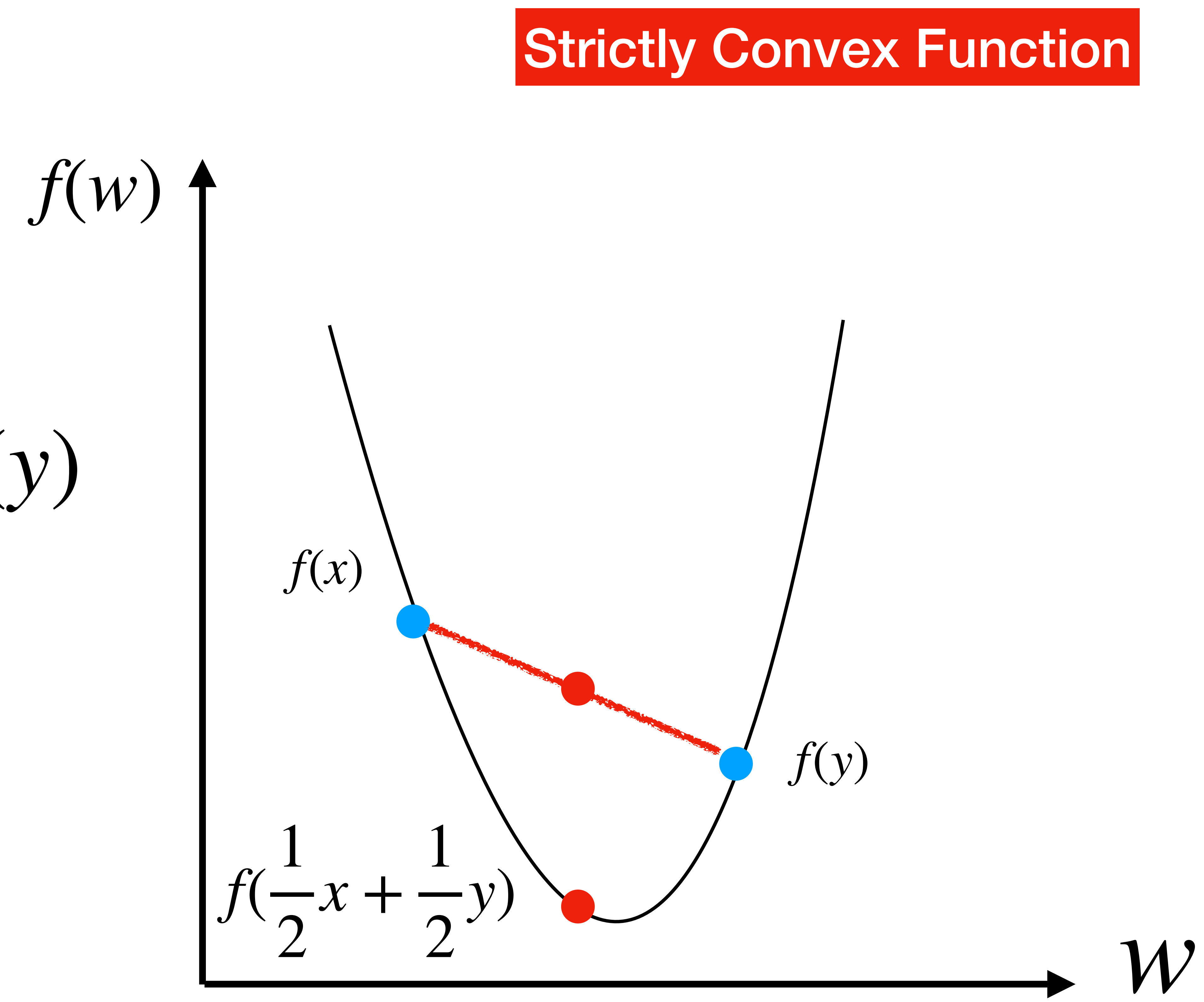
For all $\alpha \in [0,1]$



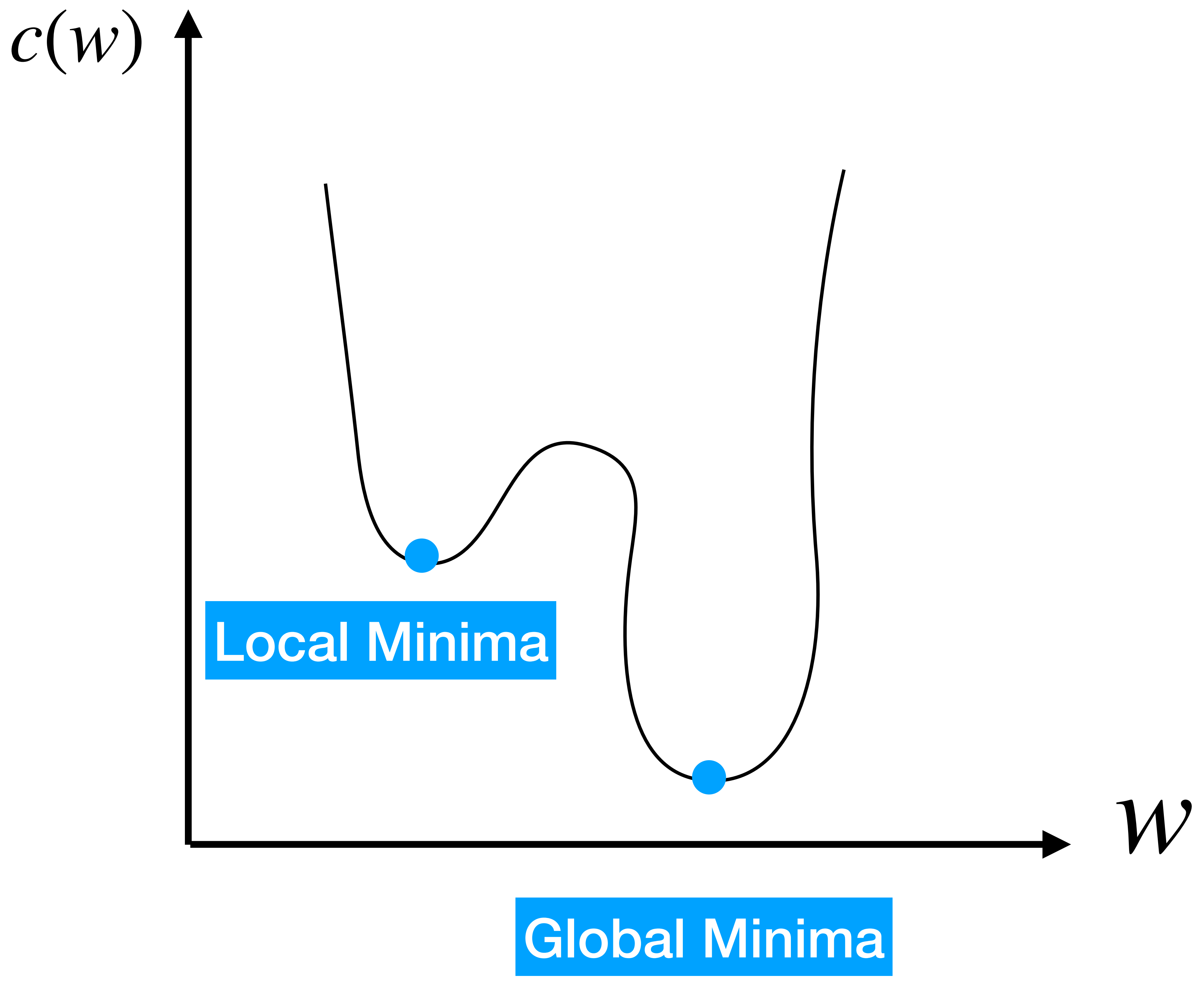
What is Convexity?

$$f(\alpha x + (1-\alpha)y) \leq \alpha f(x) + (1-\alpha)f(y)$$

For all $\alpha \in [0,1]$

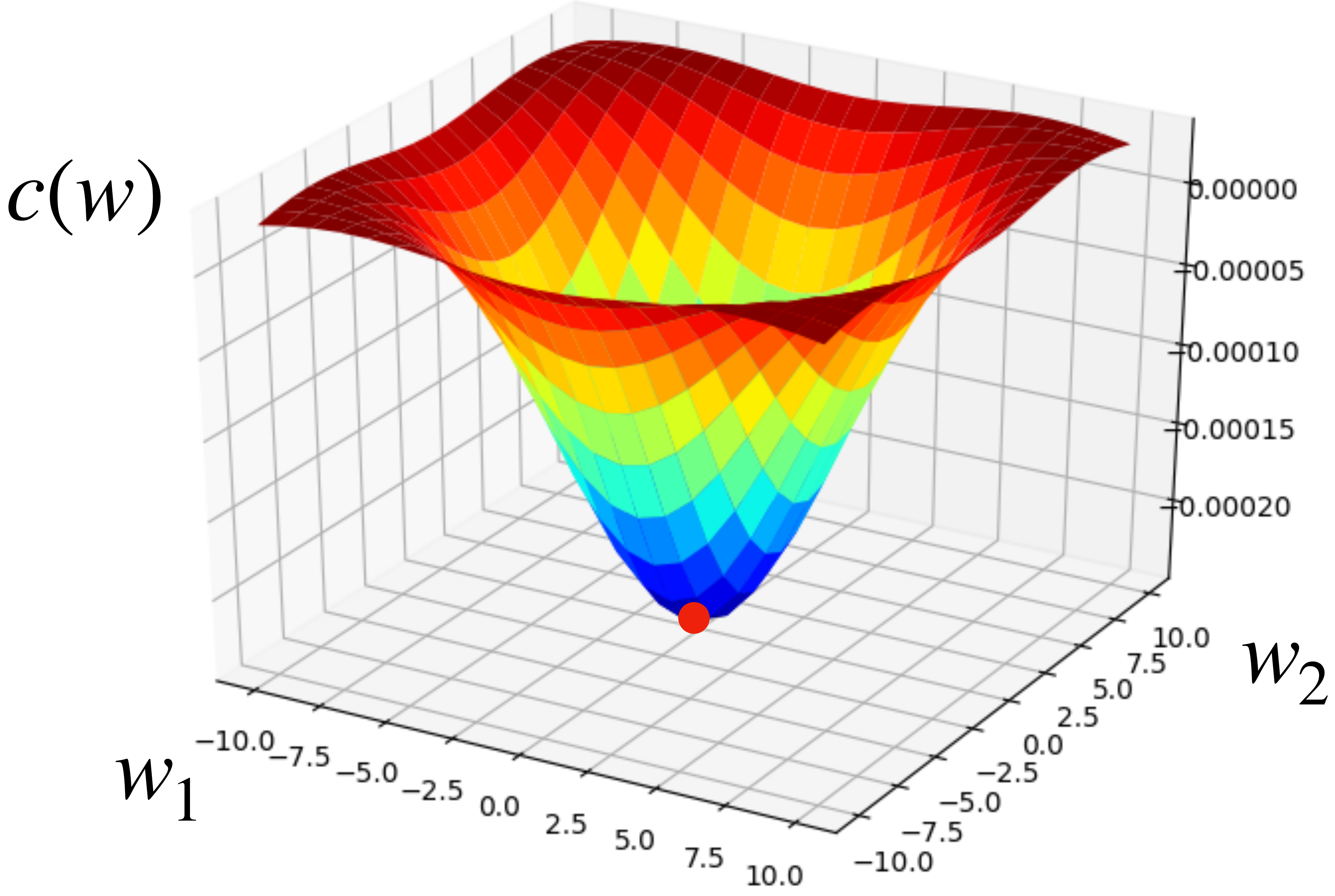


Global & Local Minima

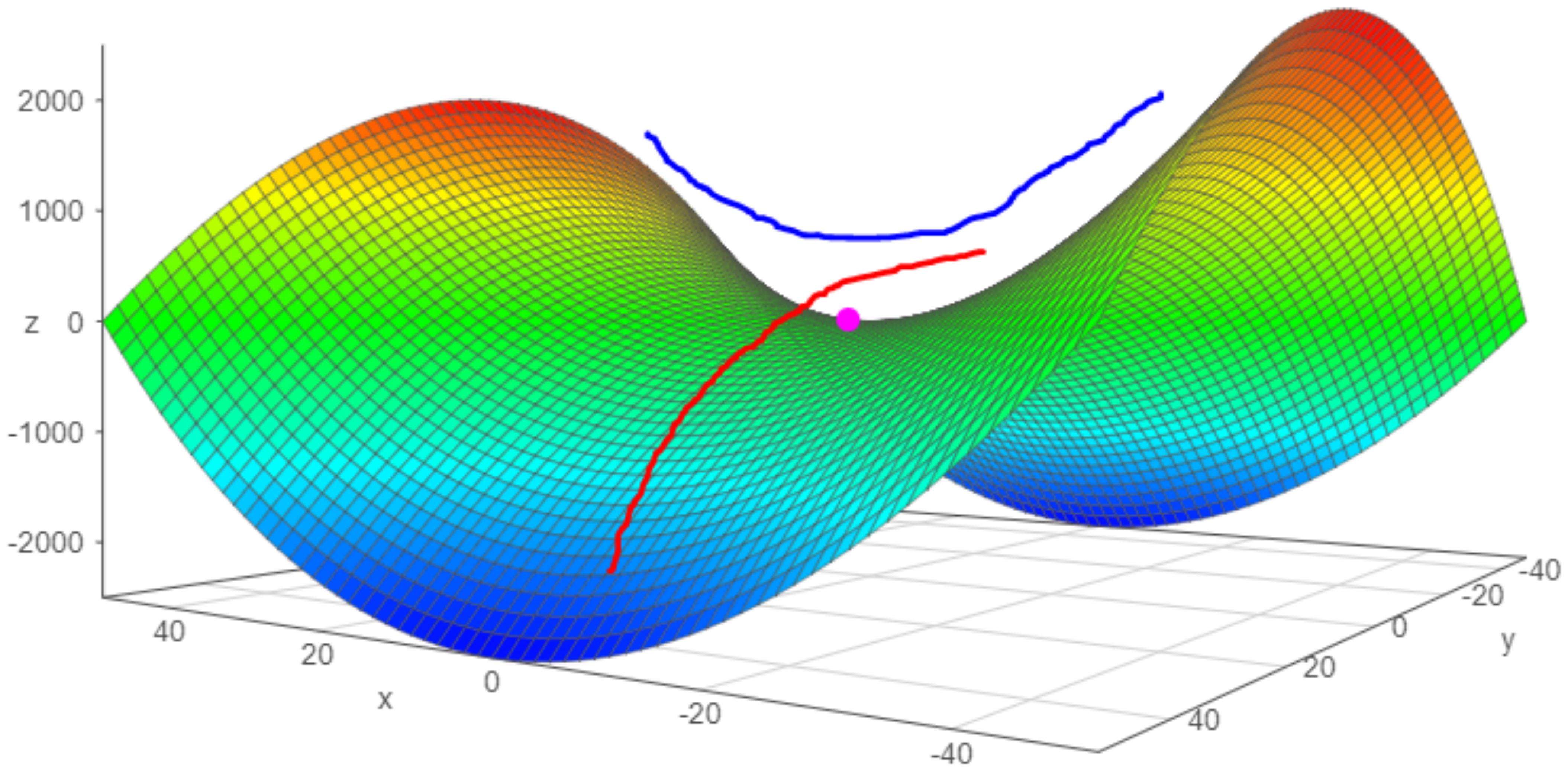


non-Convex Function

Global & Local Minima



Global Minima



Saddle point

1^{st} Order Method

Let's go back to univariate case.

We previously saw the first order version:

$$w^1 = w^0 - \alpha c'(w^0)$$

$$\Delta w = -\alpha c'(w)$$

Points down the hill

2^{nd} Order Method

$$c(w + \Delta w) = c(w) + c'(w)\Delta w + \frac{1}{2}c''(w)\Delta w^2$$

Use second order Taylor approximation this time not first order

How to get minimum of any function?

2^{nd} Order Gradient Descent

$$c(w + \Delta w) = c(w) + c'(w)\Delta w + \frac{1}{2}c''(w)\Delta w^2$$

$$\frac{d}{d\Delta w}c(w + \Delta w) = c'(w) + c''(w)\Delta w = 0$$

2^{nd} Order Gradient Descent

$$c(w + \Delta w) = c(w) + c'(w)\Delta w + \frac{1}{2}c''(w)\Delta w^2$$

$$\frac{d}{d\Delta w}c(w + \Delta w) = c'(w) + c''(w)\Delta w = 0$$

$$\Delta w = -\frac{c'(w)}{c''(w)}$$

1^{st} vs 2^{nd} Order Gradient Descent

Newton Method

First Order

$$\Delta w = -\alpha c'(w)$$

Second Order

$$\Delta w = -\frac{c'(w)}{c''(w)}$$

Faster

Newton’s Method

- Algorithm simply:

Initial guess w^o

For i=0, 1, …, M

$$w^{i+1} = w^i - \frac{c'(w^i)}{c''(w^i)}$$

End

Newton's Method

Can we do better?

$$w^{i+1} = w^i - \frac{c'(w^i)}{c''(w^i)}$$

Next part of the lecture we will look into quasi newton methods

Questions?