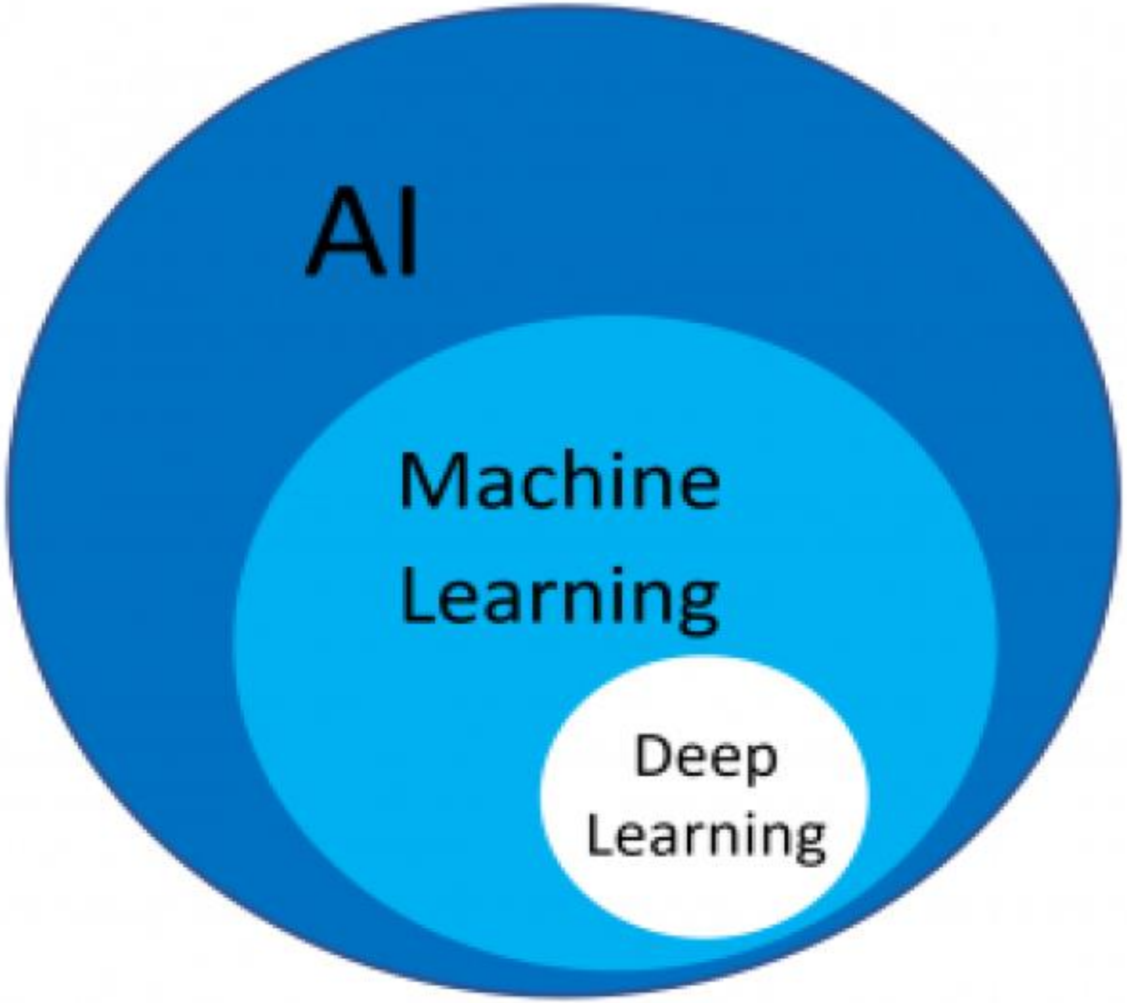
Relationship

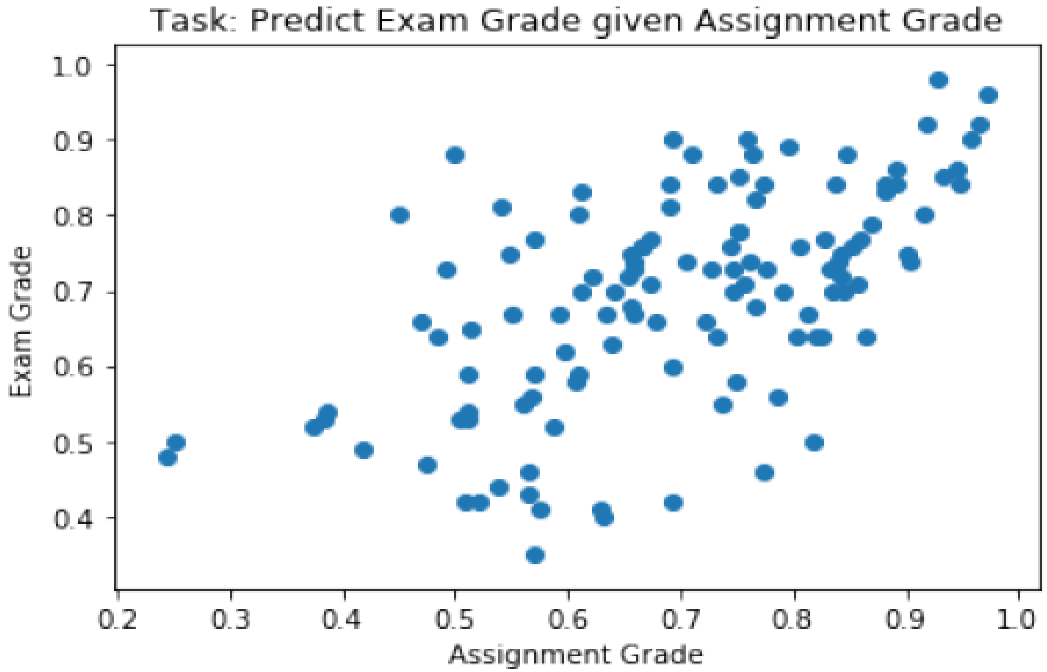


Supervised Learning Idea

- We have some data $(\mathbf{x}^{(1)}, t^{(1)}), (\mathbf{x}^{(2)}, t^{(2)}), \ldots\ (\mathbf{x}^{(N)}, t^{(N)})$
- We want to be able to make prediction y (of an unseen t) for a new value of $\mathbf{x}$
  - For example, predict the exam grade of a person who missed their exam
- How can we build a *model* to solve the prediction problem?

Supervised Learning Task: Exam Grade Prediction

(Definitely not real data from last term)



- Data: $(x^{(1)}, t^{(1)}), (x^{(2)}, t^{(2)}), \ldots (x^{(N)}, t^{(N)})$
- The $x^{(i)}$ are called *inputs*
- The $t^{(i)}$ are called *targets*

Linear Regression Model

A **model** is a set of assumptions about the underlying nature of the data we wish to learn about. The **model**, or **architecture** defines the set of allowable **hypotheses**.

In linear regression, our **model** will look like this

$$y = \sum_j w_j x_j + b$$

Where y is a prediction for t, and the w_j and b are **parameters** of the model, to be determined based on the data.

Linear Regression for Exam Grade Prediction

For the exam prediction problem, we only have a single feature, so we can simplify our model to:

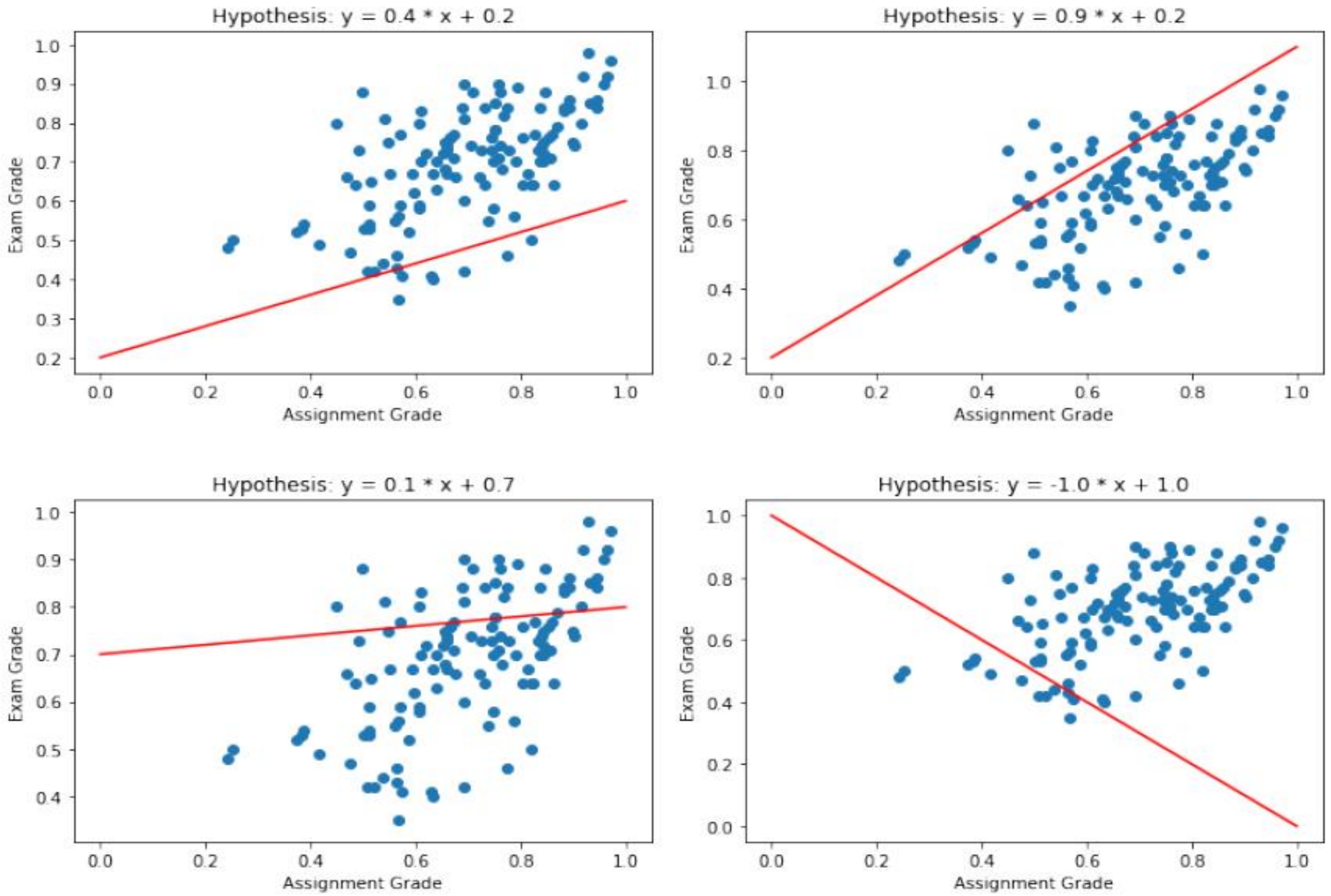
$$y = wx + b$$

Our **hypothesis space** includes all functions of the form $y = wx + b$. Here are some examples:

- $y = 0.4x + 0.2$
- $y = 0.9x + 0.2$
- $y = 0.1x + 0.7$
- $y = -x - 1$
- ...

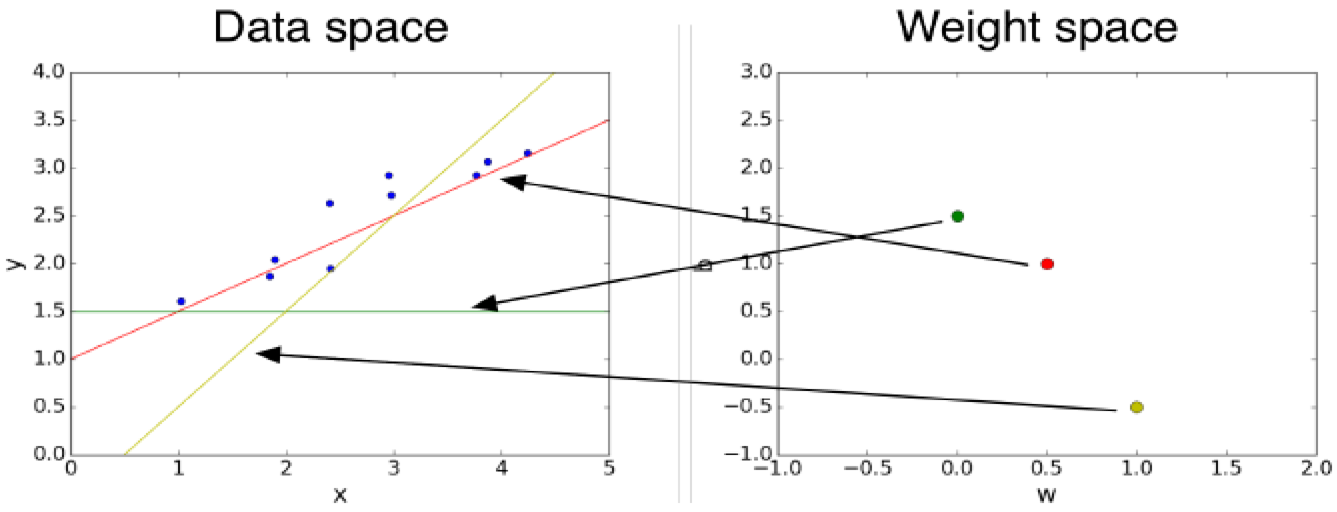
The variables w and b are called **weights** or **parameters** of our model. (Sometimes w and b are referred to as coefficients and intercept, respectively.)

Which hypothesis is better suited to the data?



Hypothesis Space

We can visualize the hypothesis space or **weight space**:



Each *point* in the weight space represents a hypothesis.

Quantifying the "badness" of a hypothesis

Idea:

- A good hypothesis should make good predictions about our labeled data $(x^{(1)}, t^{(1)})$, $(x^{(2)}, t^{(2)})$, ... $(x^{(N)}, t^{(N)})$
- That is, $y^{(i)} = wx^{(i)} + b$ should be "close to" $t^{(i)}$
- But how do we define the notion of "close to"?

We'll choose **square vertical distance**:

$$\mathcal{L}(y, t) = \frac{1}{2}(y - t)^2$$

This choice has some nice mathematical and statistical properties.

Cost Function (Loss Function)

The “badness” of an entire hypothesis is the average badness across our labeled data.

$$\begin{aligned}
\mathcal{E}(w, b) &= \frac{1}{N} \sum_i \mathcal{L}(y^{(i)}, t^{(i)}) \\
&= \frac{1}{2N} \sum_i (y^{(i)} - t^{(i)})^2 \\
&= \frac{1}{2N} \sum_i ((wx^{(i)} + b) - t^{(i)})^2
\end{aligned}$$

This is called the **loss** of a particular hypothesis.

Since the loss depends on the choice of w and b, we call $\mathcal{E}(w, b)$ the **loss function**.

Summary so far

Hypothesis	$y = wx + b$
Parameters	$w,\ b$
Loss Function	$\mathcal{E}(w, b) = \frac{1}{2N} \sum_i ((wx^{(i)} + b) - t^{(i)})^2$
Goal	Find w, b that minimize $L(w, b)$

Minimizing the Loss Function

Task: Find w and b that minimize the loss function:

$$\mathcal{E}(w, b) = \frac{1}{2N} \sum_i ((wx^{(i)} + b) - t^{(i)})^2$$

Potential Strategy: Direct Solution

Find a *critical point* by setting

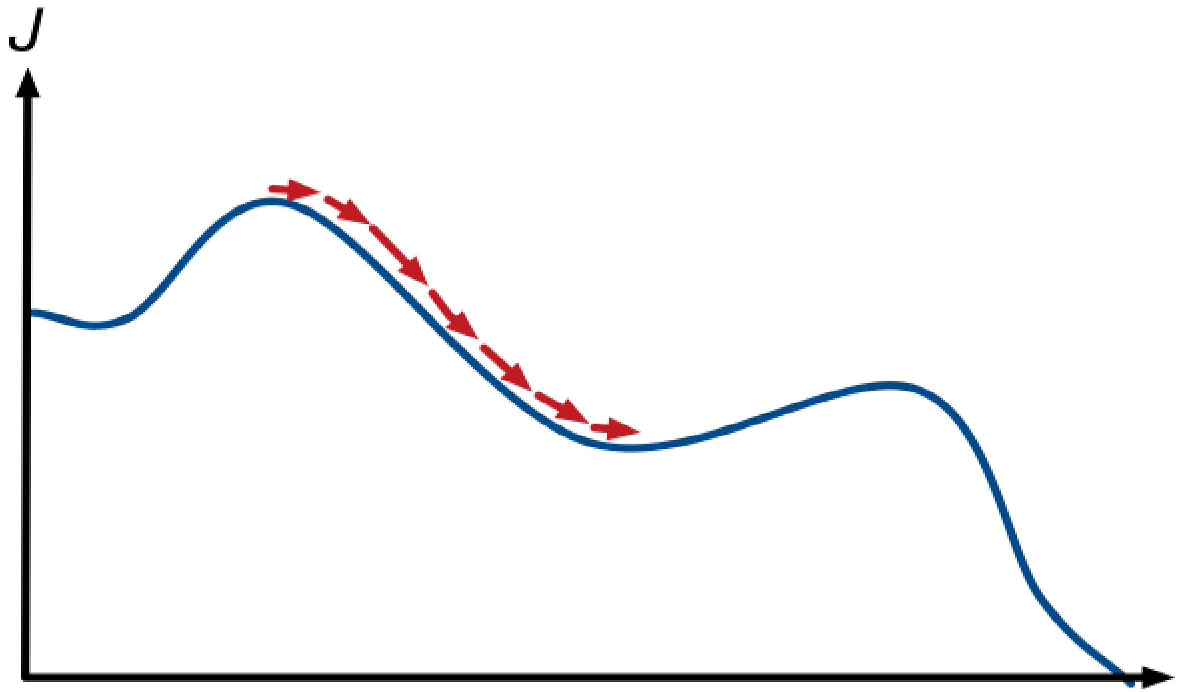
$$\frac{\partial \mathcal{E}}{\partial w} = 0$$
$$\frac{\partial \mathcal{E}}{\partial b} = 0$$

Possible for our hypothesis space, and are covered in the notes
. . . and the pre-requisite quiz! See what we did there?

However, let's use a technique that can also be applied to more general models.

Strategy: Gradient Descent

Minimizing a scalar function f(x)



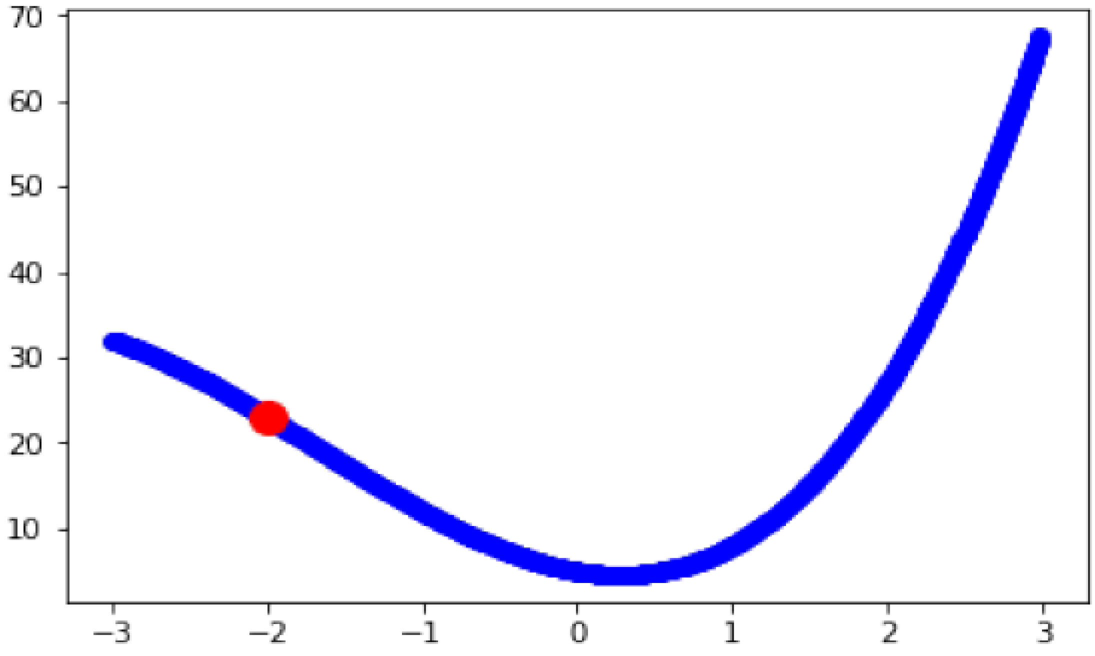
Gradient Descent is an iterative method used to find the minima of a function.

We'll start by thinking about a *scalar function* (1D)

To minimize a function f(x), we start with a random point x_0 and iterate an update rule that we will derive.

Deriving Gradient Descent Update

Consdier this function f(x)

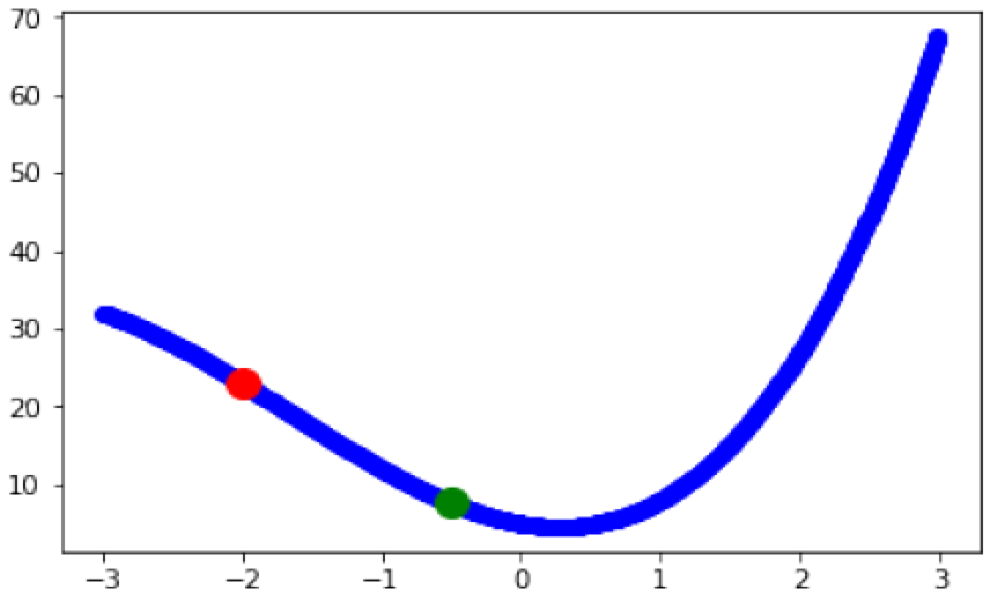


Q: If we want to move the red point closer to the minima, do we move left or right?

Q: At the red point x, is the derivative $f'(x)$ positive or negative?

We want to move x towards the negative direction of the gradient!

How much do we move?



Q: Should we make a larger jump at the red point or green?

The larger $|f'(x)|$, the more we should move. We *slow down* close to a minima.

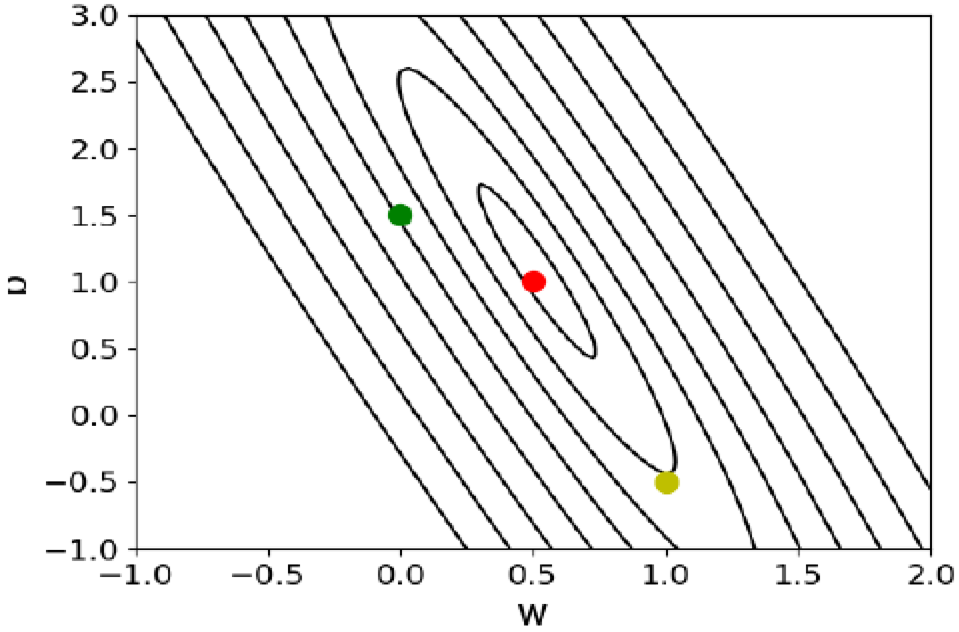
$$x \leftarrow x - \alpha f'(x)$$

The term α is the **learning rate**

Gradient Descent for Linear Regression (2D)

The same idea holds in higher dimensions:

$$w \leftarrow w - \alpha \frac{\partial \mathcal{E}}{\partial w}$$
$$b \leftarrow b - \alpha \frac{\partial \mathcal{E}}{\partial b}$$



Gradient Descent for Linear Regression (high dimensional)

Or, in general:

$$\mathbf{w} \leftarrow \mathbf{w} - \alpha \frac{\partial \mathcal{E}}{\partial \mathbf{w}}$$

$$\frac{\partial \mathcal{E}}{\partial \mathbf{w}} = \begin{bmatrix} \frac{\partial \mathcal{E}}{\partial w_1} \\ \cdots \\ \frac{\partial \mathcal{E}}{\partial w_D} \end{bmatrix}$$

It turns out that the gradient is the direction of the **steepest descent**.

Gradient Descent: when to stop?

In theory:

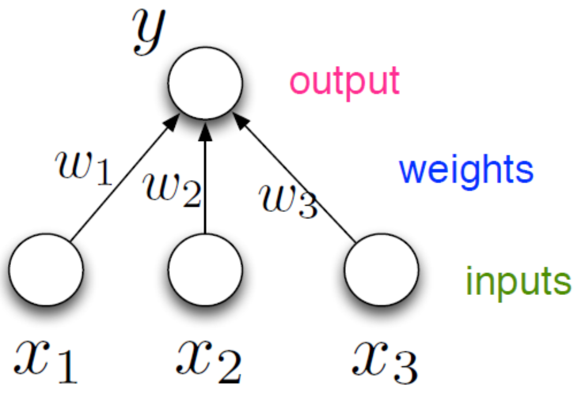
- Stop when w and b stop changing (convergence)

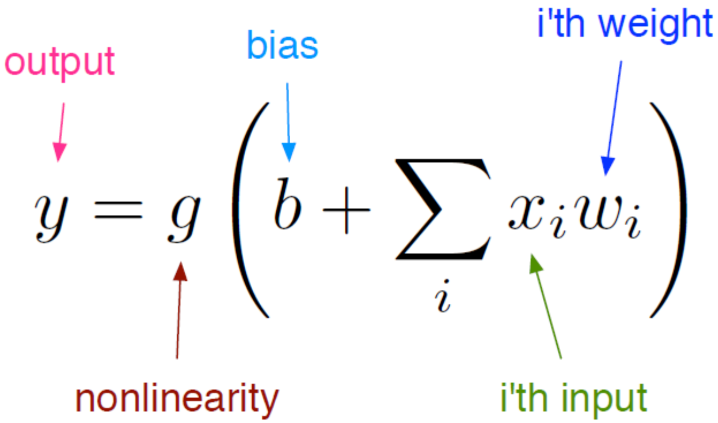
In practice:

- Stop when $\mathcal{E}$ almost stops changing (another notion of convergence)
- Stop until we're tired of waiting

What are neural networks?

- While neural nets originally drew inspiration from the brain, nowadays we mostly think about math, statistics, etc.





- Neural networks are collections of thousands (or millions) of these simple processing units that together perform useful computations.