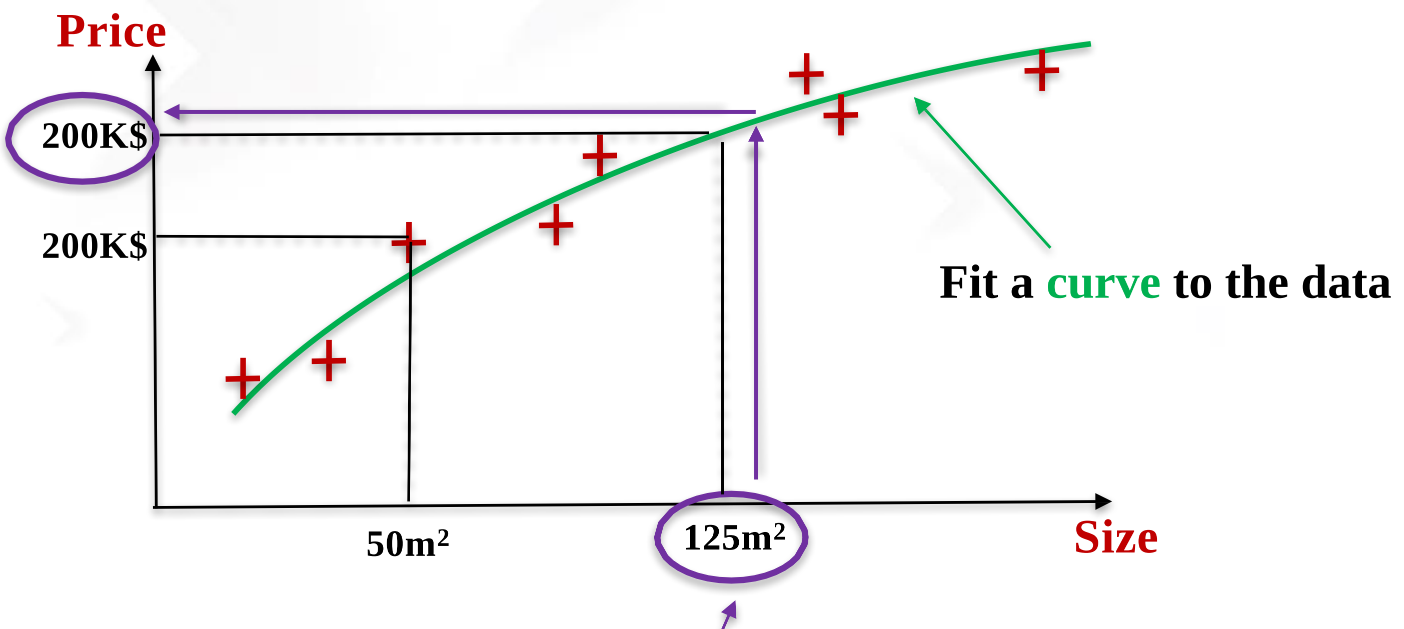
인공지능과 수학적 배경

Linear Regression

Housing price prediction

- **Supervised regression problem: Predict the price (continuous value) of houses given existing data features/properties (house size).**



What is the price for a house of size 125m²?
Supervised regression predicts280K$.

Linear Regression

Formalization

- **Supervised regression** as an example of **supervised learning**:

Training dataset (x, y)

For regression, y is a scalar

Input: x (data) → Supervised learning algorithm f → Output: y (data label)

here: house size

here: house price

f : prediction function

f maps data feature x to data label y

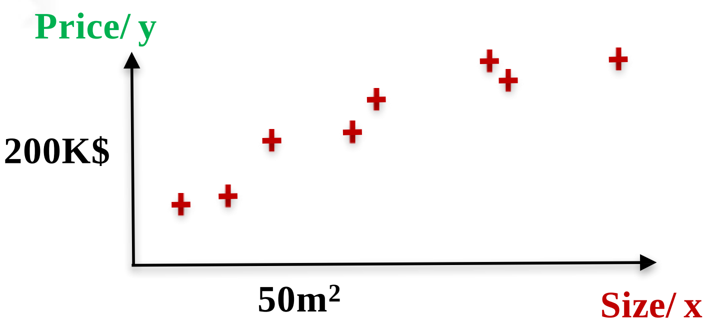
$f : x \rightarrow y$ or $y = f(x)$

Linear Regression

Training set

- All supervised learning techniques use a training set to design the prediction function *f*.

- Notations:
  - n=number of training data, here n=8
  - x=input variable/feature, here x=size
  - y=output variable/label, here y=price



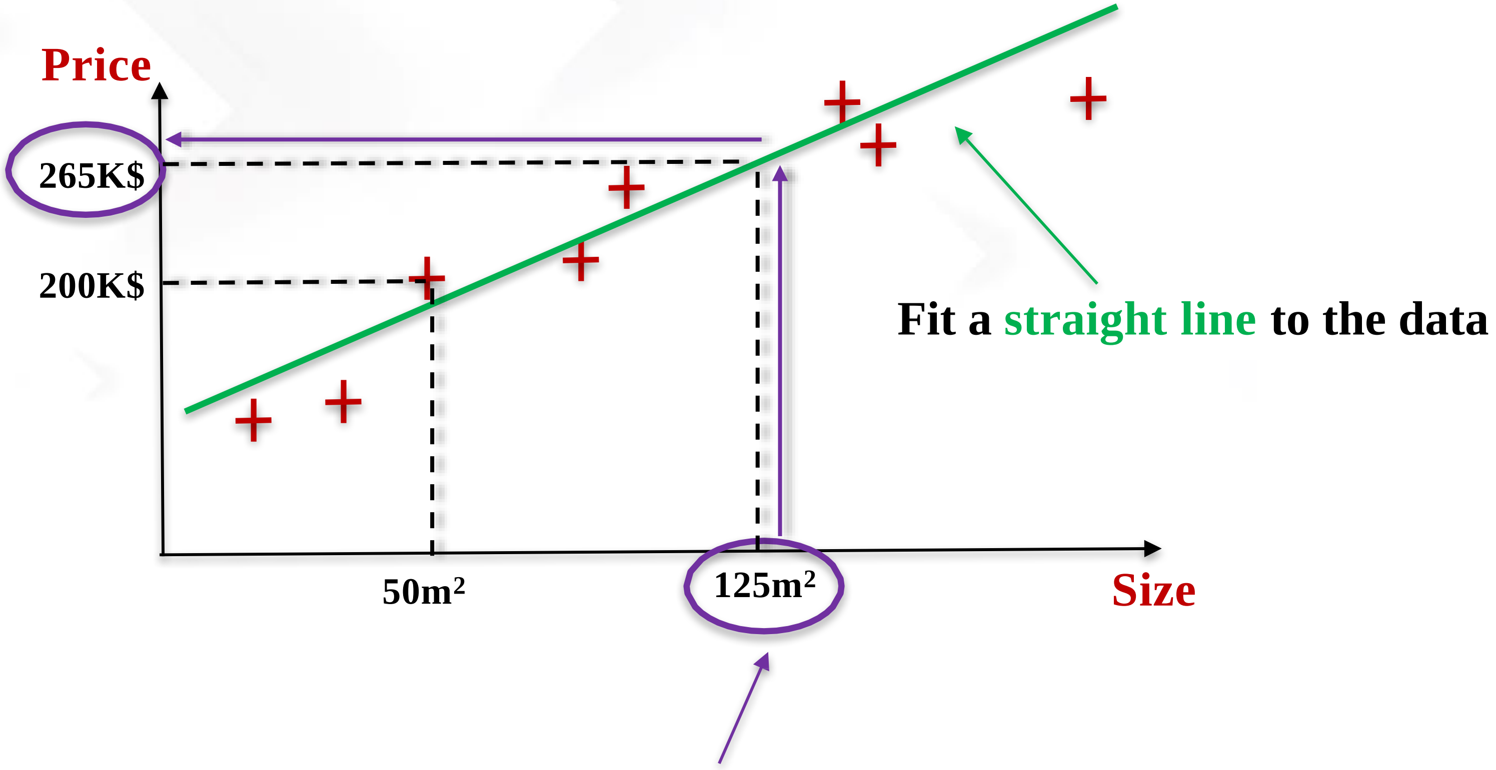
Size (x)	Price (y)
31	124
54	156
…	…

- One training data/sample is identified as (x_i,y_i) where i is the index of the training data. Examples: $(x_1,y_1)=(31,124)$, $(x_2,y_2)=(54,156)$.

Linear Regression

Linear regression

- What is the simplest model representation to regress the data?
  Straight line.

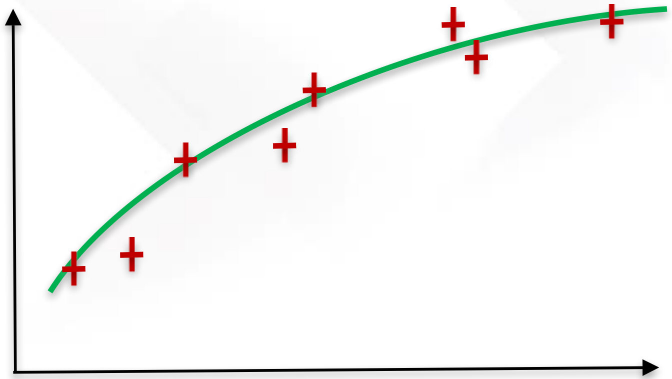


What is the price for a house of size 125m²?
Supervised regression predicts 265K$.

Linear Regression

Model representation

- How to represent the prediction function f ?

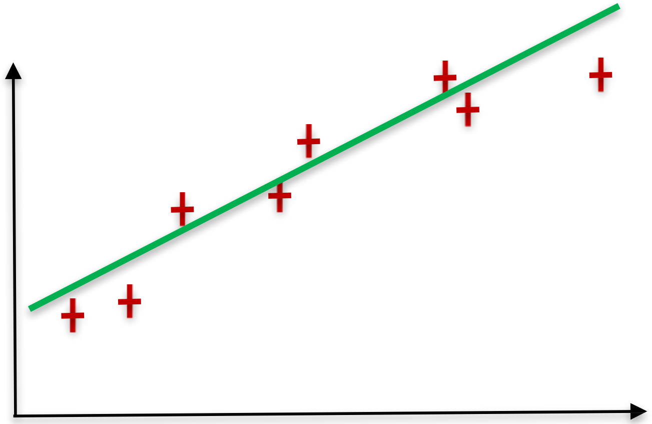


$$y = f(x)$$

Non-linear function

Example: Neural networks

- Case of linear regression:

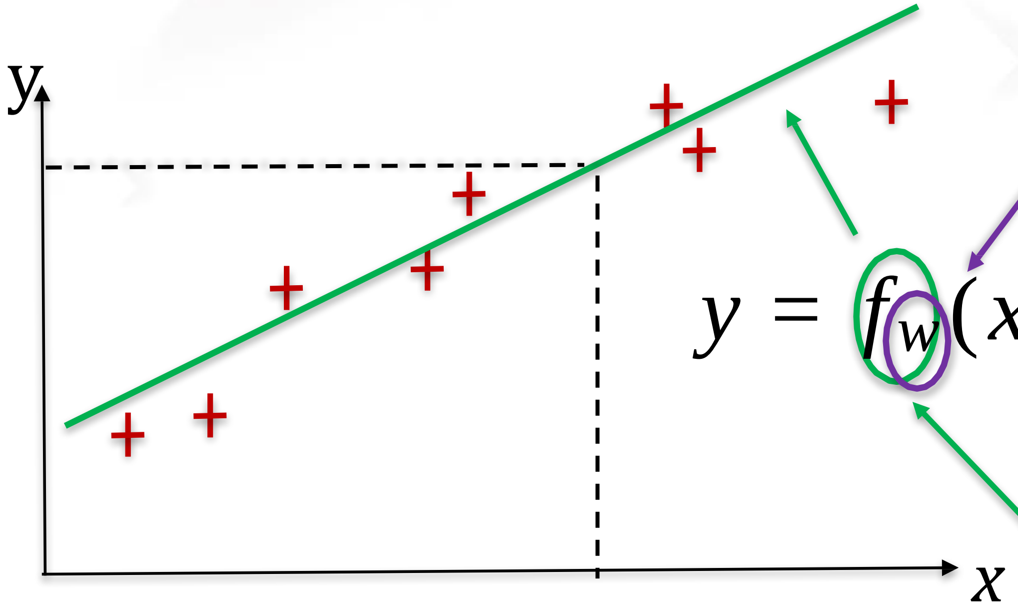


$$y = f(x)$$

Linear function

Linear representation

- **Predictor function *f* as a straight line:**



Compact representation of all variables/parameters of prediction function *f*

$$w = (w_0, w_1)$$

Data feature

$$y = f_w(x) = w_0 + w_1 x$$

Model parameters

Linear predictive function/
Linear regression model with one single variable *x*

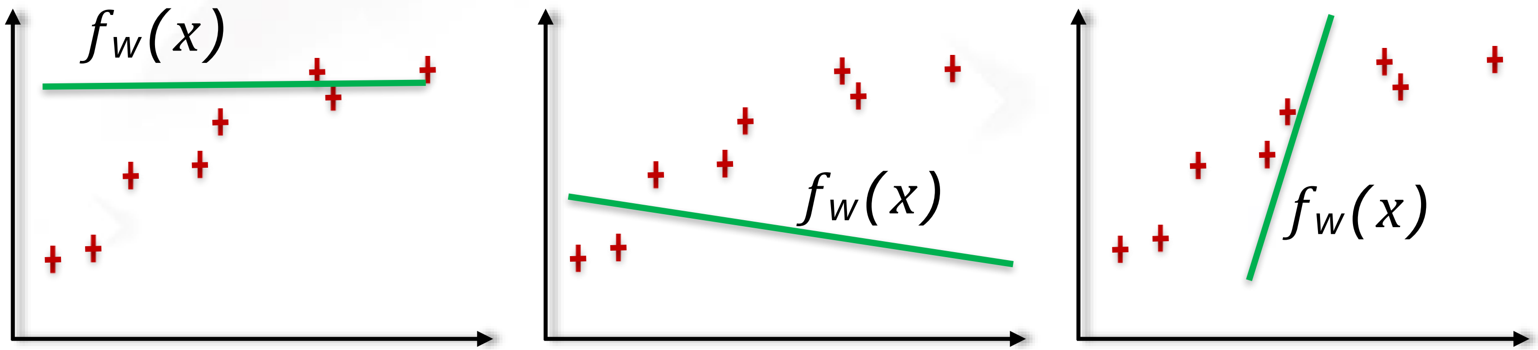
Linear Regression

Prediction function parameters

- **Prediction function:** $f_w(x) = w_0 + w_1 x$

Model parameters

- **Influence of different parameter values (w_0,w_1) on the prediction:**

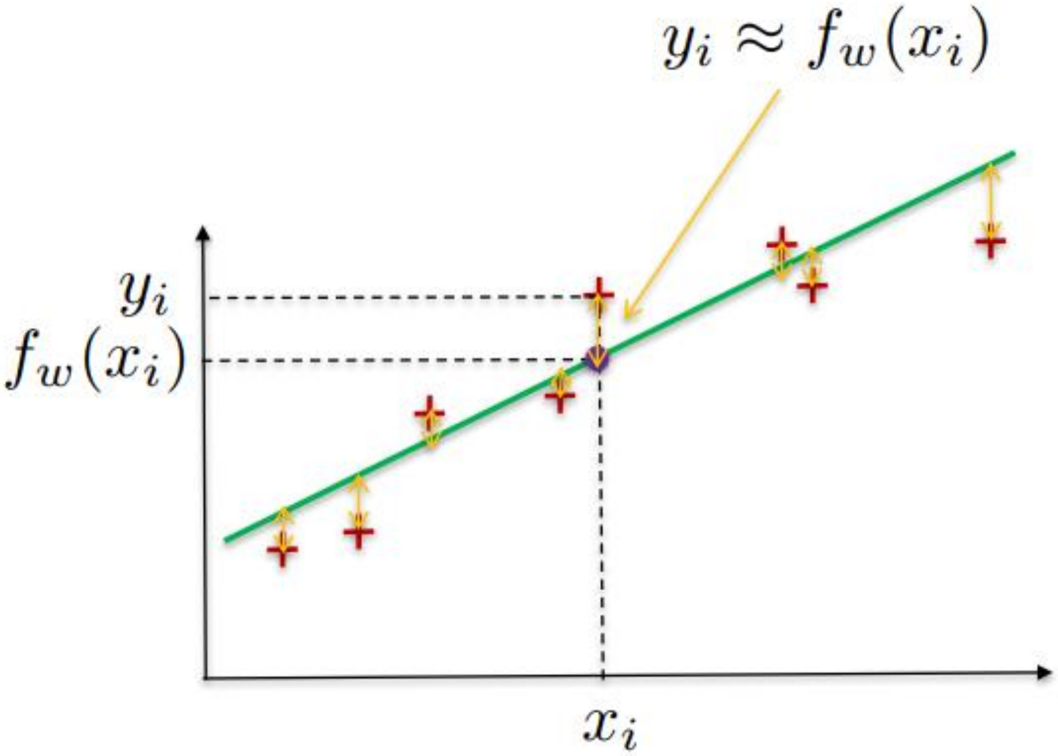


w_0=1.3, w_1=0 | w_0=2.9, w_1= -0.5 | w_0=-1.4, w_1=10.4

- **How to find the best possible straight line (w_0,w_1) that fits all data? A fitness measure is required $\Rightarrow$ Loss function.**

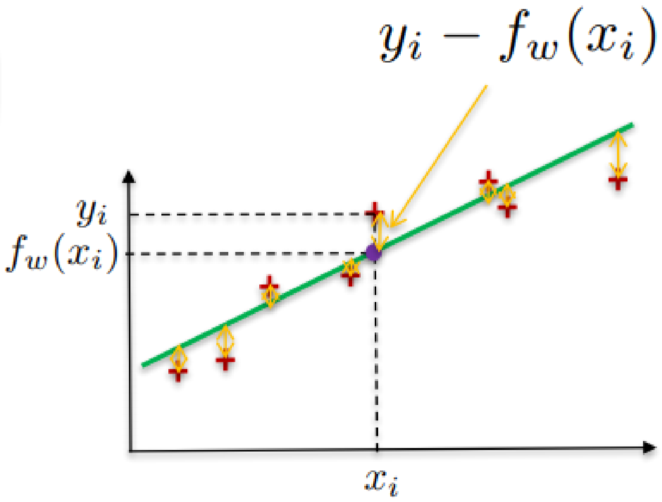
How to choose the parameters?

- **Idea: Choose parameters (w_0,w_1) such that $f_w(x_i)$ is close to y_i for all training data (x_i,y_i), that is**



Formalization

- Find parameters (w_0, w_1) by solving a **minimization** problem:



$$\min_{w=(w_0,w_1)} \underbrace{\overbrace{\frac{1}{n}}^{\text{Average prediction error}} \underbrace{\sum_{i=1}^{n}}_{\text{Sum over the } n \text{ training data}} \underbrace{\Big(\underbrace{f_w(x_i)}_{\text{Prediction given } x_i} - y_i\Big)^2}_{\text{Prediction error}}}_{L(w)\ \text{Loss function}}$$

- Remarks:
  - Generic optimization problem in supervised learning (f is not specified here).
  - Loss function is also called cost function.
  - This loss function is called mean square error (most used regression loss).

Linear regression loss

- **From generic to linear regression task**

$$\min_{w=(w_0,w_1)} \frac{1}{n}\sum_{i=1}^{n}\Big(f_w(x_i)-y_i\Big)^2$$

$$\Downarrow \quad f_w(x_i) = w_0 + w_1 x_i$$

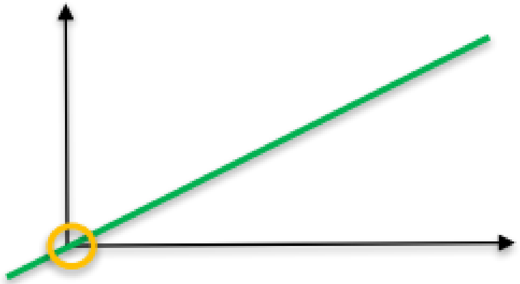
$$\min_{w=(w_0,w_1)} \underbrace{\frac{1}{n}\sum_{i=1}^{n}\Big(w_0 + w_1 x_i - y_i\Big)^2}_{L(w_0,\, w_1)}$$

Linear regression loss

Linear Regression
Loss analysis

- **Prediction function:** $f_w(x) = w_0 + w_1 x$
- **Parameters:** w_0, w_1
- **Loss function:** $L(w_0, w_1) = \frac{1}{n}\sum_{i=1}^{n}\left(w_0 + w_1 x_i - y_i\right)^2$
- **Optimization:** $\min_{w=(w_0,w_1)} L(w_0, w_1)$

- **Let us simplify with a single parameter: $w_0 = 0$, $w_1 = w$**



w_0=0

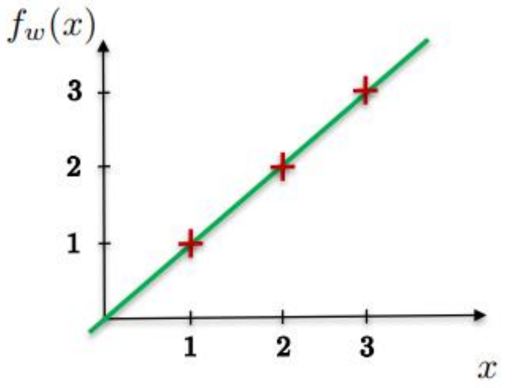
$$f_w(x) = wx$$

$$\min_w L(w) = \frac{1}{n}\sum_{i=1}^{n}\left(wx_i - y_i\right)^2$$

Linear Regression
Loss analysis

Prediction function $f_w(x)$

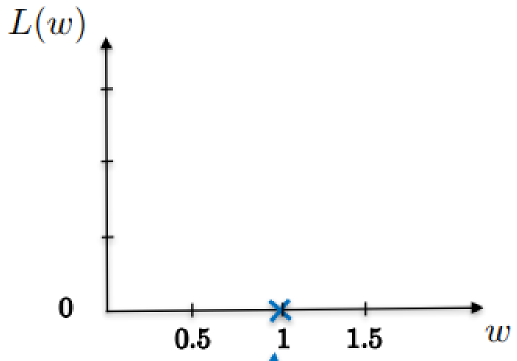
Data feature



$$f_w(x) = x, \; w = 1$$

Loss function $L(w)$

Prediction parameter



$$L(w=1) = \frac{1}{n}\sum_{i=1}^{n}\left(x_i - y_i\right)^2$$

$$\frac{1}{3}\left((1-1)^2 + (2-2)^2 + (3-3)^2\right) = 0$$

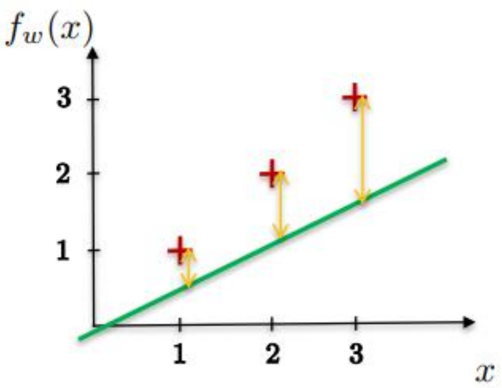
$$L(w=1) = 0$$

No error in prediction $\Rightarrow$ loss=0

Linear Regression
Loss analysis

Prediction function $f_w(x)$

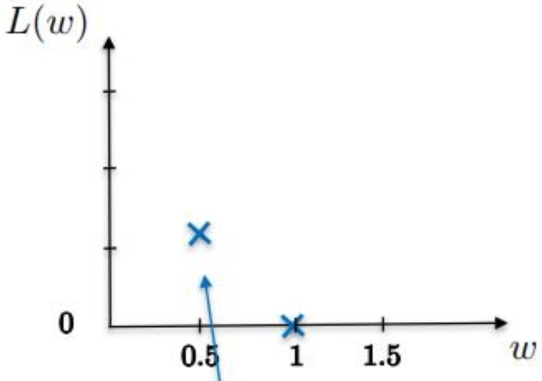
Data feature



$$f_w(x) = x, \ w = 0.5$$

Loss function $L(w)$

Prediction parameter



$$L(w=1) = \frac{1}{n}\sum_{i=1}^{n}\left(0.5x_i - y_i\right)^2$$

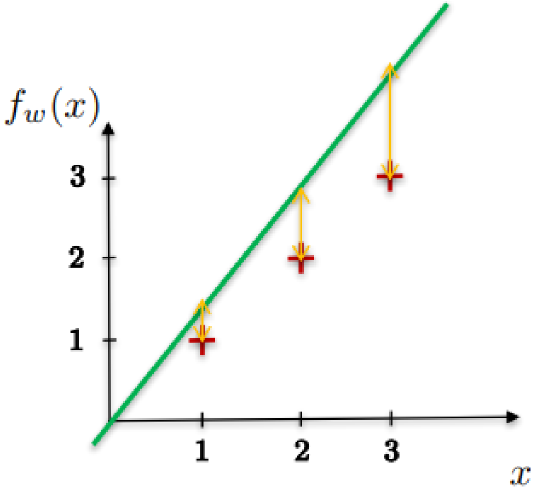
$$\frac{1}{3}\left((0.5-1)^2 + (1-2)^2 + (1.5-3)^2\right) = 1.16$$

$$L(w=0.5) = 1.16$$

Linear Regression
Loss analysis

Prediction function $f_w(x)$

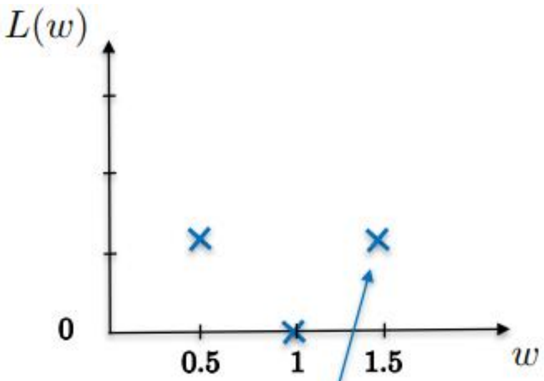
Data feature



$$f_w(x) = x, w = 1.5$$

Loss function $L(w)$

Prediction parameter



$$L(w = 1.5) = \frac{1}{n}\sum_{i=1}^{n}\left(1.5x_i - y_i\right)^2$$

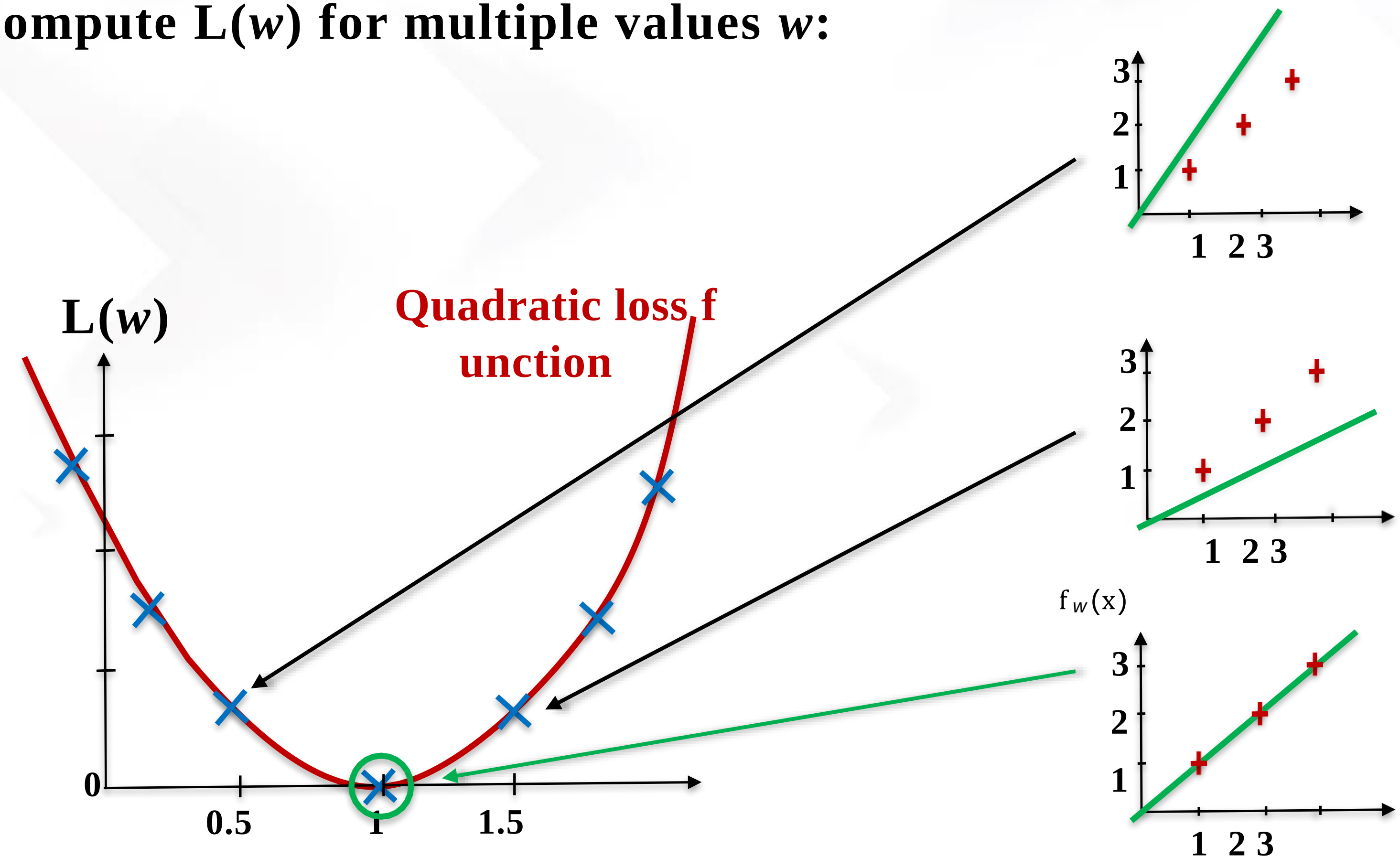
$$\frac{1}{3}\left((1.5-1)^2 + (3-2)^2 + (4.5-3)^2\right) = 1.16$$

$$L(w = 1.5) = 1.16$$

Linear Regression

Loss analysis

- **Compute L(*w*) for multiple values *w*:**



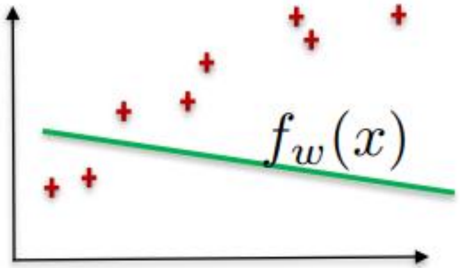
$$\min_{w} \; L(w) = \frac{1}{n}\sum_{i=1}^{n}\left(wx_i - y_i\right)^2 \quad \Rightarrow \quad w = 1$$

Parameter $w=1$ provides a perfect prediction function for all training data.

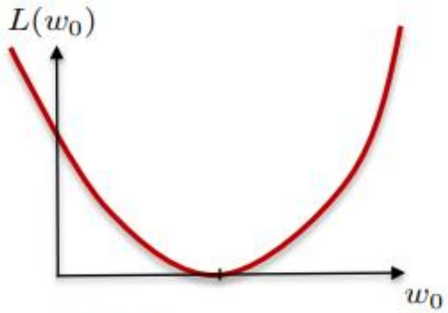
Linear Regression

Loss function with 2 parameters

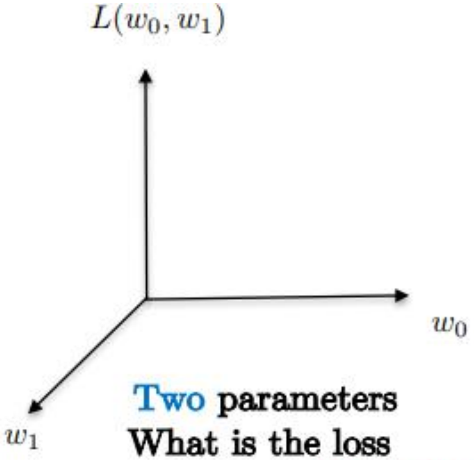
- **Prediction function:** $f_w(x) = w_0 + w_1 x$
- **Parameters:** w_0, w_1
- **Loss function:** $L(w_0, w_1) = \frac{1}{n}\sum_{i=1}^{n}\left(w_0 + w_1 x_i - y_i\right)^2$
- **Optimization:** $\min_{w=(w_0,w_1)} L(w_0, w_1)$



w_0=2.9
w_1=-0.5



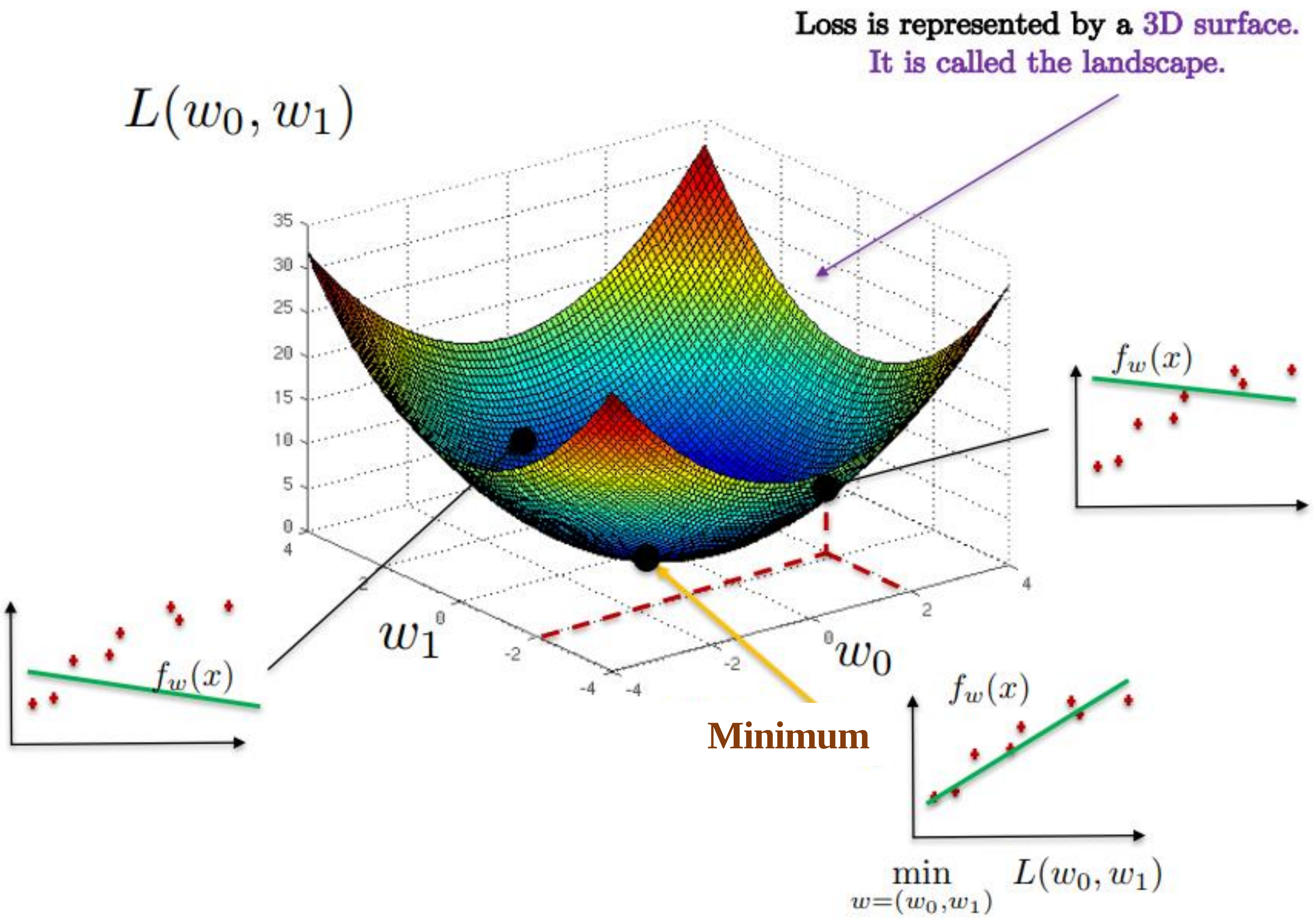
Single parameter
Loss is represented
by a curve



Two parameters
What is the loss
representation? 3D surface

Linear Regression

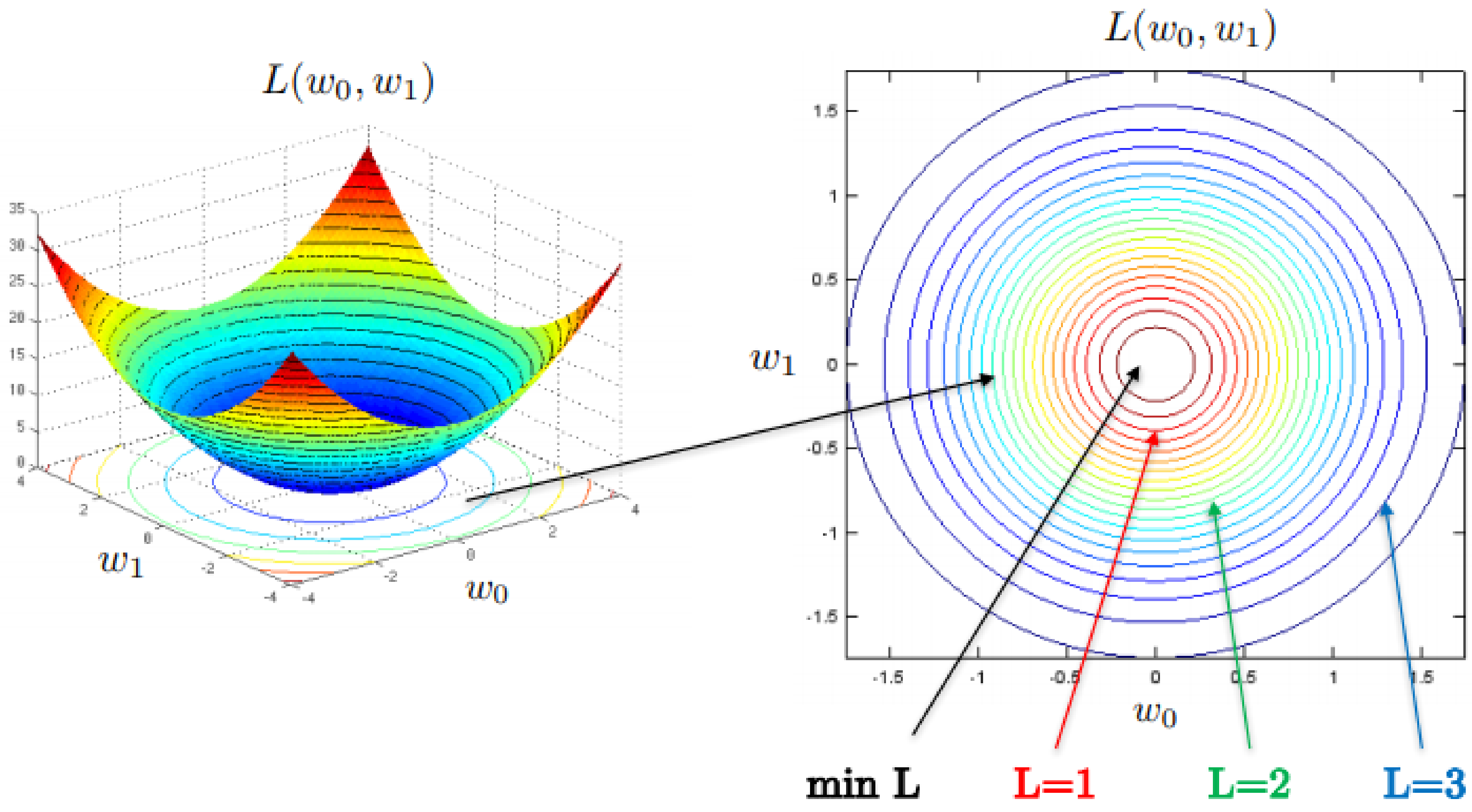
Loss function with 2 parameters



Linear Regression

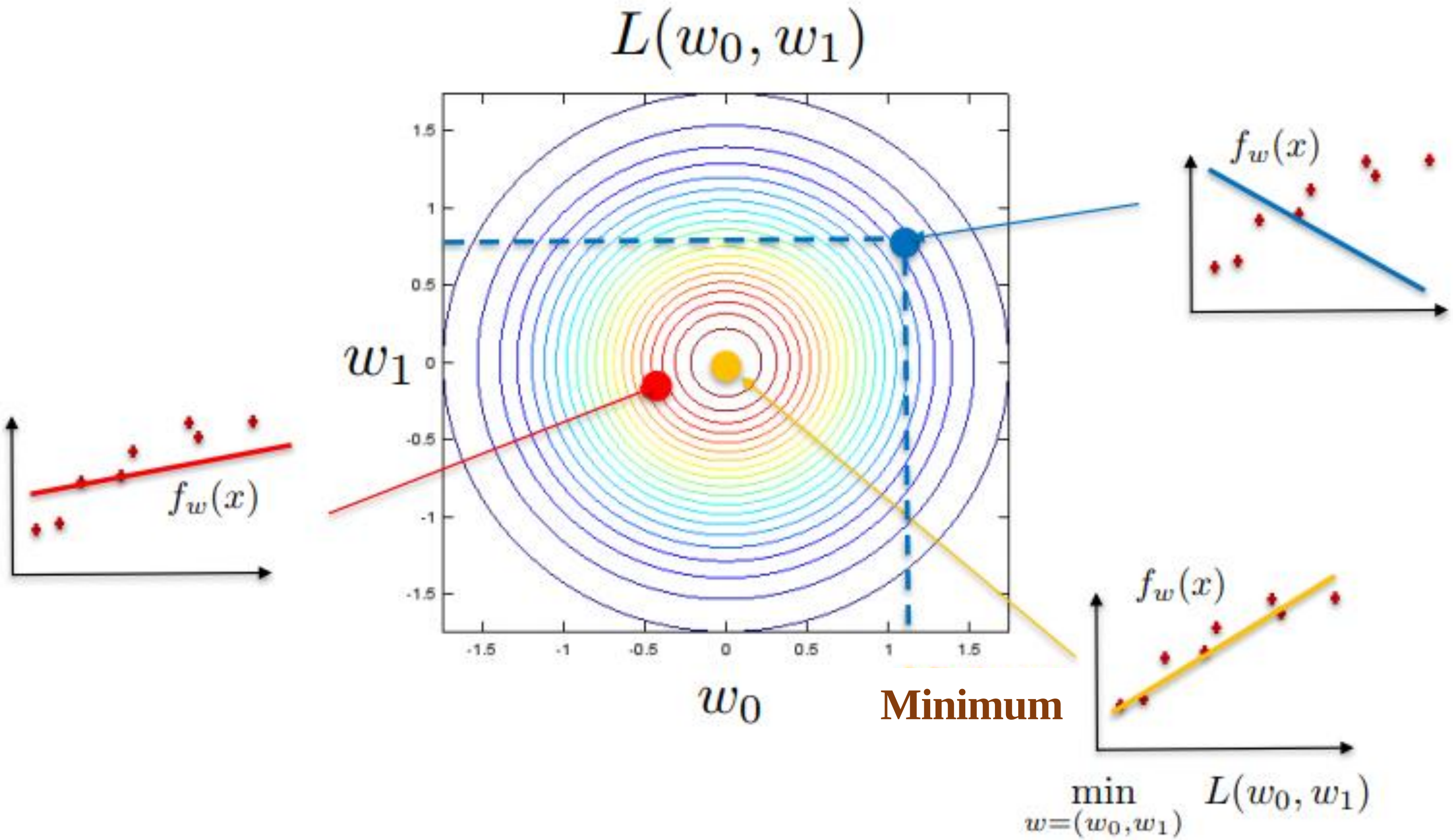
Contours of L

- Level sets/iso-contours = curves with the same loss values.
- They offer a 2D visualization of the 3D loss surface.



Linear Regression

Loss function with 2 parameters



- How to find the value of the parameters (w_0,w_1) that minimize the loss?
  ⇒ Gradient descent.

Optimization problem

- **Prediction function:** $f_w(x) = w_0 + w_1 x$
- **Parameters:** w_0, w_1
- **Loss function:** $L(w_0, w_1) = \frac{1}{n} \sum_{i=1}^{n} \left(w_0 + w_1 x_i - y_i \right)^2$
- **Optimization:** $\min_{w=(w_0, w_1)} \quad L(w_0, w_1)$