2. LINEAR REGRESSION

Chapter 2

Linear Regression

Model Representation - Linear Regression - Chapter 2 $\boxed{1}$

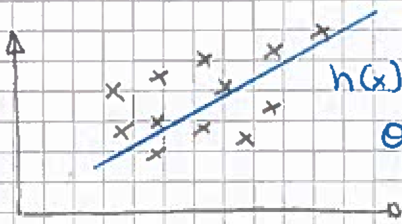
Notation:

m - number of training examples
x - input variable / feature
y - output variable
(x,y) simple training example
$(x^{(i)}, y^{(i)})$ specific example

Training Set
↓
Learn algorithm
↓
Size of house → $\boxed{h}$ → Estimated price
hypothesis

Representation of h

$\Rightarrow \quad h_\theta(x) = \theta_0 + \theta_1 x \quad \Rightarrow$



$h(x) = \theta_0 + \theta_1 x$

$\Rightarrow ax + b$

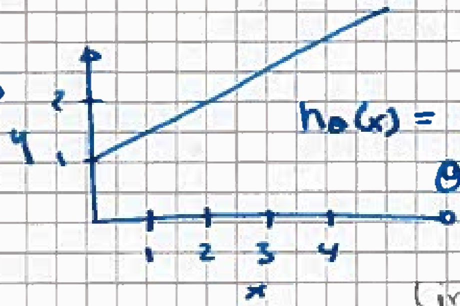
↳ shorthand $h(x)$

$$\underbrace{}$$

Linear Regression with one variable
Univariate Linear Regression

Cost Function

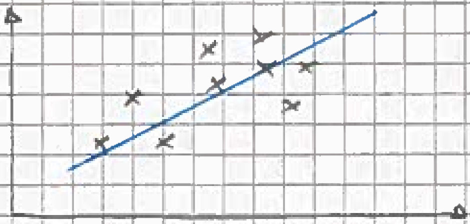
$$h_\theta(x) = \theta_0 + \theta_1 x \Rightarrow$$



$h_\theta(x) = 1 + 0{,}5x$

$\theta_0 = 1 \quad \theta_1 = 0.5$

Lineare Funktion
$ax+b$



Idea: Chose θ_0, θ_1 so that $h_\theta(x)$ is close to y for training example (x, y)

$$\underset{\theta_0, \theta_1}{\text{minimize}} \quad \frac{1}{2m} \sum_{i=1}^{m} \Big(\underbrace{h_\theta(x^{(i)})}_{\text{Predicted value}} - \underbrace{y^{(i)}}_{\text{real value}} \Big)^2$$

m — # of training examples

$\Downarrow$

$h(x^{(i)}) = \theta_0 + \theta_1 x \Rightarrow$ given by the data

$\Downarrow$

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)} \right)^2$$

minimize $= J(\theta_0, \theta_1) \Rightarrow$ Cost Function

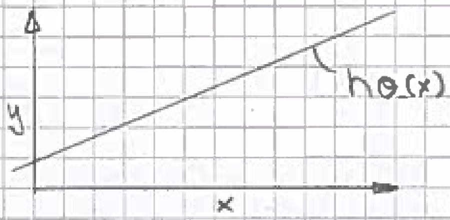
$\Rightarrow$ Squared error function

Cost function intuition I

Summary:

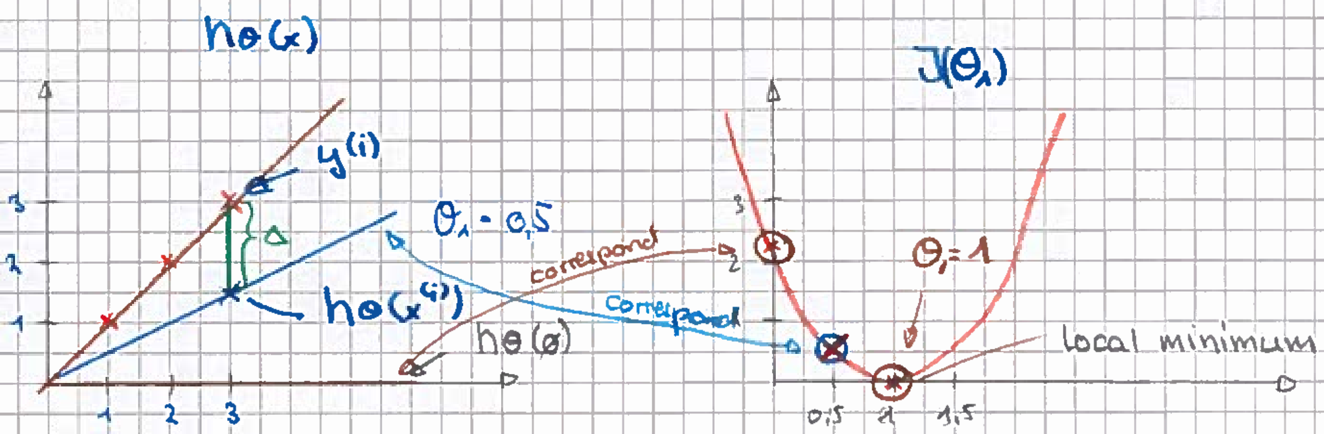
Hypothesis: $h_\theta(x) = \theta_0 + \theta_1 x$

Parameters: θ_0, θ_1



Cost function: $J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h(x^{(i)}) - y^{(i)}\right)^2$

Goal: $\underset{\theta_0, \theta_1}{\text{minimize}} \; J(\theta_0, \theta_1)$



$$J(0{,}5) = \frac{1}{2m} \sum_{i=1}^{m} \Rightarrow \left[(0{,}5-1)^2 + (1-2)^2 + (1{,}5-3)^2\right]$$

$$\Rightarrow \frac{1}{2 \cdot 3} \cdot 3{,}5 = 0{,}58$$

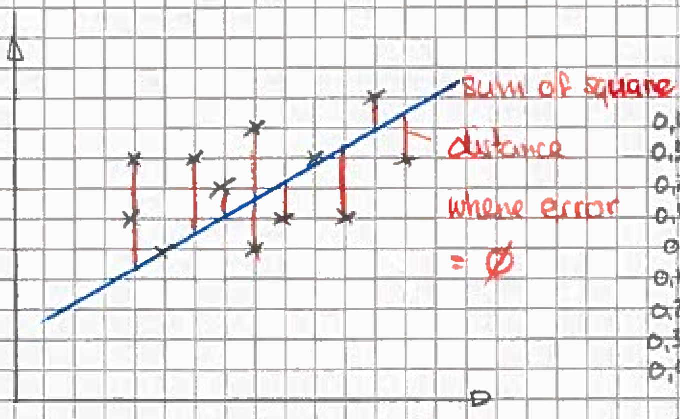
Δ difference between predicted and effective value

minimize $J(\theta_1)$

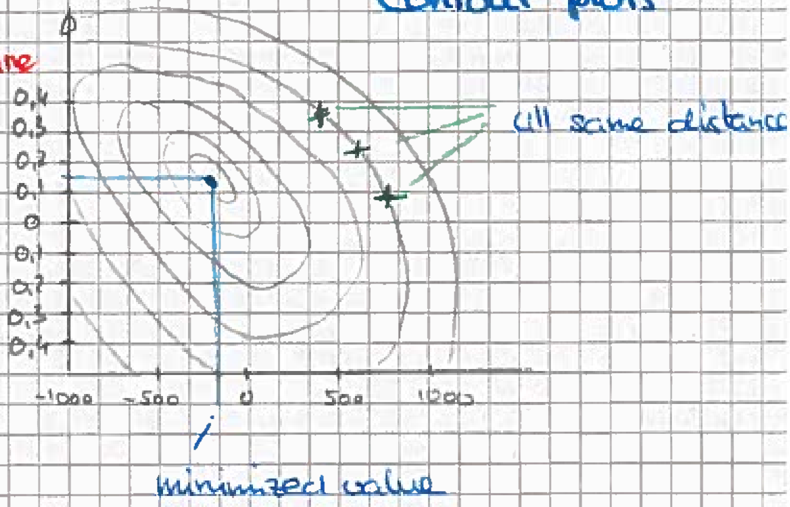
$$J(0) \Rightarrow \frac{1}{2m} \cdot \left[(0-1)^2 + (0-2)^2 + (0-3)^2\right] = \frac{1 \cdot 14}{2 \cdot 3} = \sim 2{,}3$$

$$J(1) \Rightarrow \frac{1}{2m} \cdot \left[(1-1)^2 + (2-2)^2 + (3-3)^2\right] = \frac{0}{6} = 0$$

Cost function intuition II



Contour plots



Gradient descent algorithm

⇒ want to get the local/global minimum

$$\min J(\theta_0,\theta_1)$$

⇒ Gradient descent applies to more general functions

$$J(\theta_0,\theta_1 \ldots \theta_n) \Rightarrow \min J(\theta_0,\theta_1 \ldots \theta_n)$$

Repeat until convergence {

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial_j} J(\theta_0,\theta_1) \quad (\text{for } j=0,\ j=1)$$

}

⇒ simultaniou update

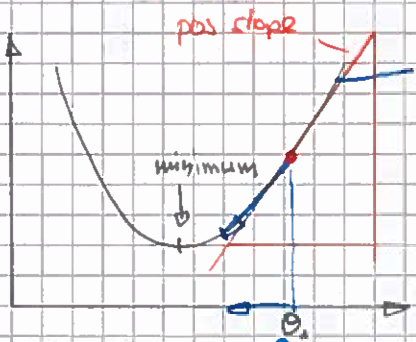
α = learning rate

$$temp\emptyset := \theta_0 - \alpha \frac{\partial}{\partial\theta_0} J(\theta_0,\theta_1)$$

$$temp1 := \theta_1 - \alpha \underbrace{\frac{\partial}{\partial\theta_1} J(\theta_0,\theta_1)}_{\text{derivative term}}$$

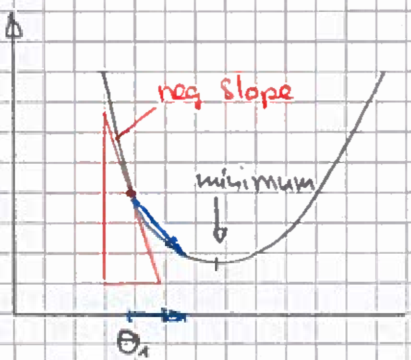
$$\theta_0 := temp\emptyset$$

$$\theta_1 := temp1$$



$J(\theta_1) \Rightarrow$

$\theta_1 \in \mathbb{R}$



$$\theta_1 := \theta_1 - \alpha \underbrace{\frac{\partial}{\partial\theta_1} \cdot J(\theta_1)}_{\geq 0}$$

$$\underbrace{\frac{\partial}{\partial\theta_1} \cdot J(\theta_1)}_{\leq \emptyset}$$

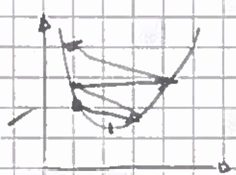
$\theta_1 := \theta_1 - \alpha \cdot (\text{pos. number})$

always pos.

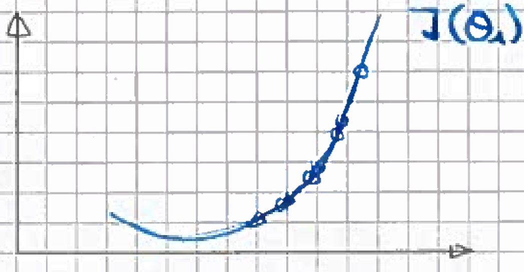
$\theta_1 := \theta_1 - \alpha \cdot (\text{neg. number})$

make sure that α is not too small or to large

- too small might take to many steps to converge
- too big might overshoot the min. and might never converge.



Gradient descent can converge to a local minimum, even with the learning rate α fixed.



$$\theta_1 := \theta_1 - \alpha \underbrace{\frac{\partial}{\partial \theta_1} J(\theta_1)}_{x}$$

As we approach local minimum, gradient descent will take smaller steps. No need to decrease α.

term x will decrease therefore value will be smaller & smaller

Linear Regression with One Variable

Apply gradient descent to linear regression function

$$\frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1) = \frac{\partial}{\partial \theta_1} \cdot \frac{1}{2m} \cdot \sum_{i=1}^{m} \left(\underbrace{h_\theta(x^{(i)})} - y^{(i)}\right)^2$$

$$\Rightarrow \frac{\partial}{\partial \theta_1} \cdot \frac{1}{2m} \sum_{i=1}^{m} \left(\theta_0 + \theta_1 x^{(i)} - y^{(i)}\right)^2$$

$$\theta_0 :\Rightarrow j = 0 : \frac{\partial}{\partial \theta_0} \cdot J(\theta_0, \theta_1) = \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)}\right)$$

$$\theta_1 :\Rightarrow j = 1 : \frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1) = \frac{1}{m} \sum \left(h_\theta(x^{(i)}) - y^{(i)}\right) \cdot x^{(i)}$$

Gradient descent algorithm

repeat until convergence {

$$\theta_0 := \theta_0 - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)}\right)$$

$$\theta_1 := \theta_1 - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)}\right) \cdot x^{(i)}$$

}

always update simultaniously.