KECERDASAN BUATAN

3| REGRESI

Dr. Prima Dewi Purnamasari
Program Studi Teknik Komputer FTUI

Pendahuluan

Apa itu Regresi?

- Analisis/uji regresi merupakan suatu kajian dari hubungan antara satu variabel, yaitu variabel yang diterangkan (*the explained variabel*) dengan satu atau lebih variabel, yaitu variabel yang menerangkan (*the explanatory*).
- Apabila variabel bebasnya **hanya satu**, maka analisis regresinya disebut dengan regresi linear sederhana (**simple linear regression**).
- Apabila variabel bebasnya **lebih dari satu**, maka analisis regresinya dikenal dengan regresi linear berganda (**multiple regression**).
  - Dikatakan berganda karena terdapat beberapa variabel bebas yang mempengaruhi variabel tak bebas
- Hasil dari analisis/uji regresi berupa suatu persamaan regresi.
  - Persamaan regresi ini merupakan suatu fungsi prediksi variable yang mempengaruhi variabel lain

Simple Linear Regression

- Regresi Linear Sederhana adalah Metode Statistik yang berfungsi untuk menguji sejauh mana hubungan sebab akibat antara Variabel Faktor Penyebab (X) terhadap Variabel Akibatnya (Y).
- X = predictor
- Y = response
- Regresi Linear Sederhana atau sering disingkat dengan SLR (Simple Linear Regression) juga merupakan salah satu Metode Statistik yang dipergunakan dalam produksi untuk melakukan peramalan ataupun prediksi tentang karakteristik kualitas maupun kuantitas.

Contoh Penggunaan

- Contoh penggunaan analisis Regresi Linear Sederhana dalam kegiatan produksi, antara lain:
  - Hubungan antara lamanya kerusakan mesin dengan kualitas produk yang dihasilkan
  - Hubungan jumlah pekerja dengan output yang diproduksi
  - Hubungan antara suhu ruangan dengan cacat produksi yang dihasilkan.

Metode Regresi Linear Sederhana

diambil dari Machine Learning with Python cognitiveclass.ai

Regresi Untuk Melakukan Prediksi Pada Data yang Kontinyu

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	?

Misal: 1 independent variable X untuk memprediksi dependent variable Y

X: Independent variable

Y: Dependent variable

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	?

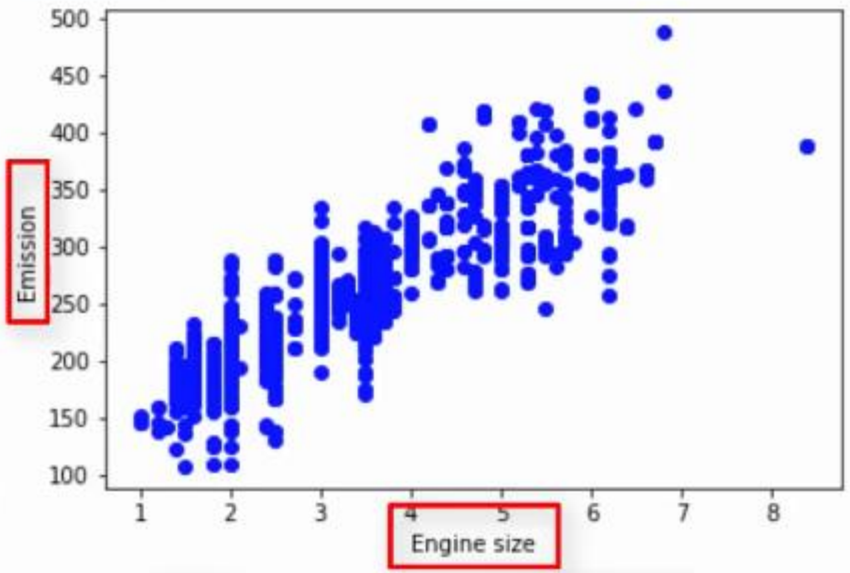
Continuous Values

Topologi Regresi Linear

- Regresi Linear Sederhana :
  - Prediksi emisi Co2 VS Ukuran mesin (engine size)
    - Variabel Independen (X): Ukuran mesin (engine size)
    - Variabel Dependen (Y): Emisi Co2
- Regresi Linear Berganda:
  - Prediksi emisi Co2 VS Ukuran mesin (engine size) dan Silinder
    - Variabel Independen (X): Ukuran mesin (engine size), Silinder
    - Variabel Dependen (Y): Emisi Co2

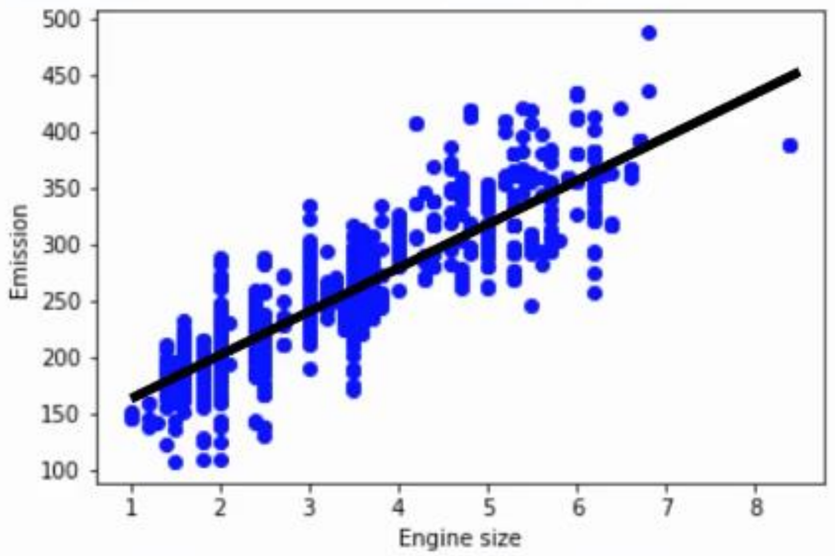
Bagaimana Regresi Linear Bekerja?

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	?



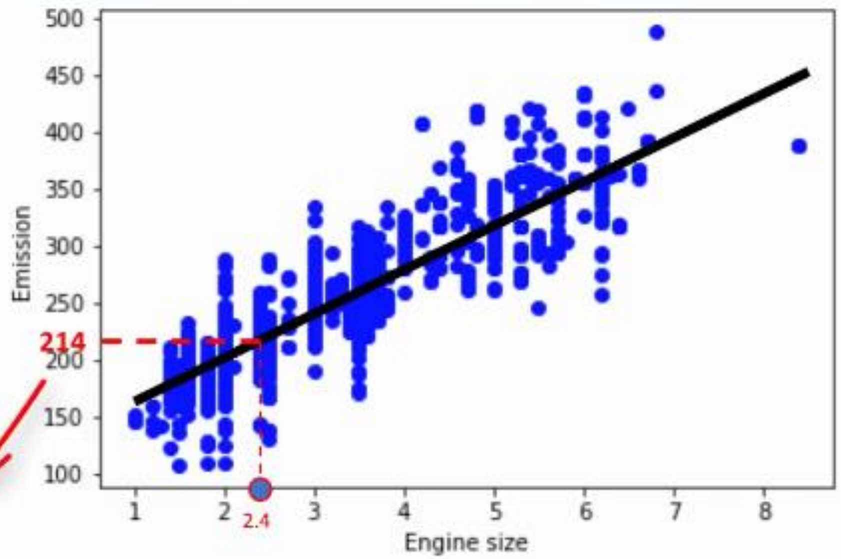
Bagaimana Regresi Linear Bekerja?

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	?

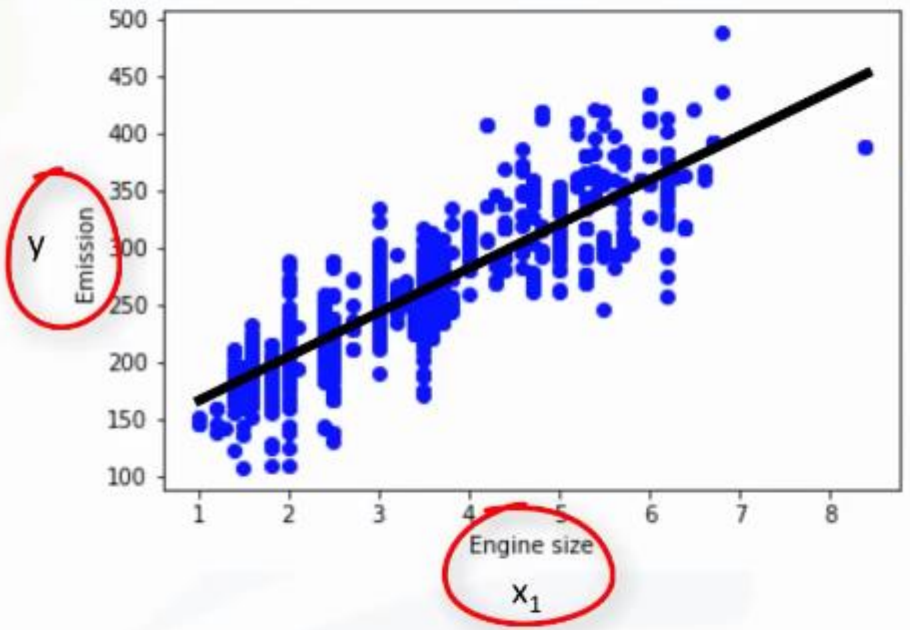


Bagaimana Regresi Linear Bekerja?

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	?

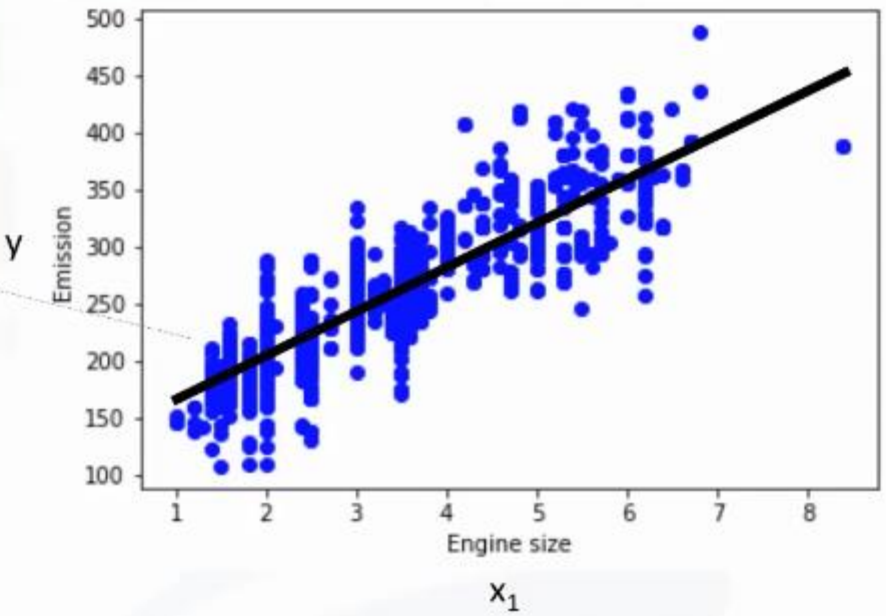


Representasi Model Regresi Linear



Representasi Model Regresi Linear

$$\hat{y} = \theta_0 + \theta_1 x_1$$

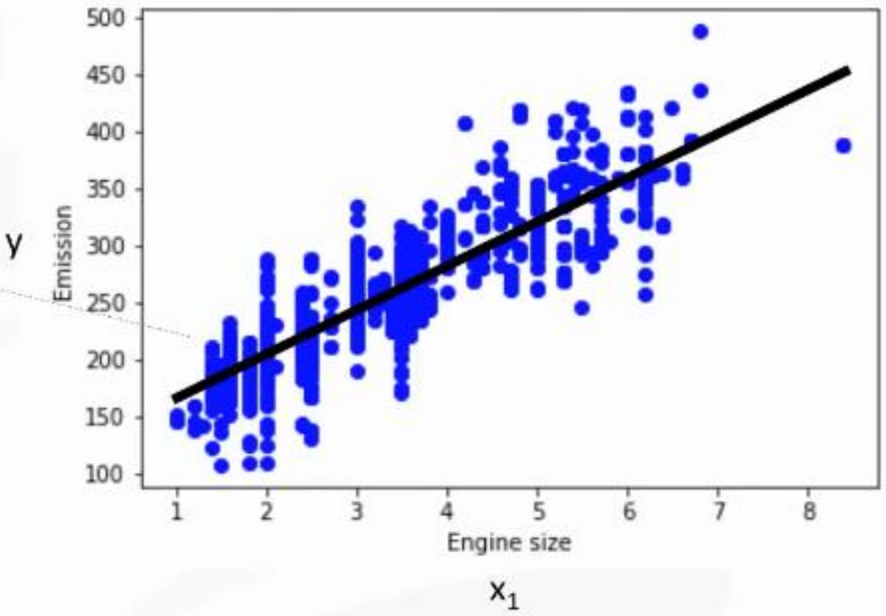


Representasi Model Regresi Linear

$$\hat{y} = \boxed{\theta_0 + \theta_1} x_1$$

a single predictor

response variable

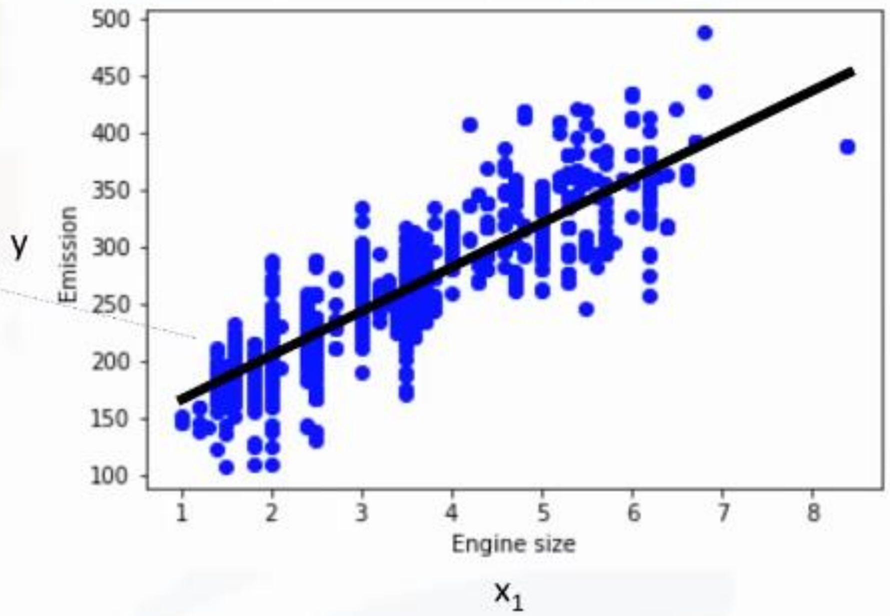


Representasi Model Regresi Linear

$$\hat{y} = \theta_0 + \theta_1 x_1$$

response variable

a single predictor

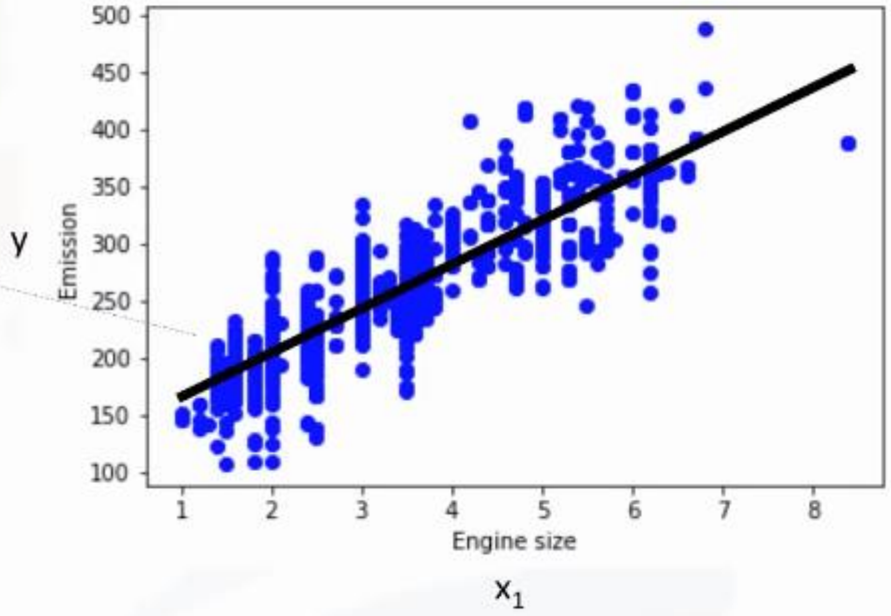


Representasi Model Regresi Linear

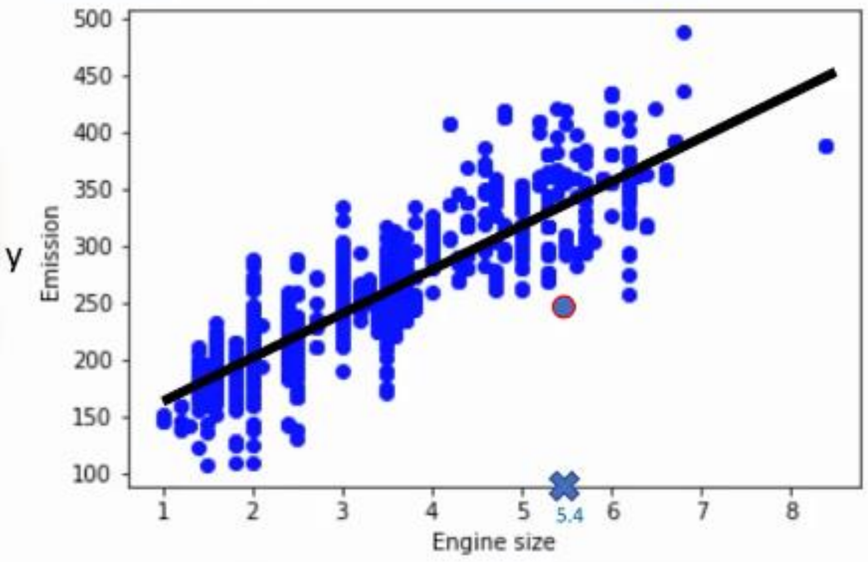
$$\hat{y} = \theta_0 + \theta_1 x_1$$

a single predictor

response variable



Bagaimana Mencari Model yang Fit

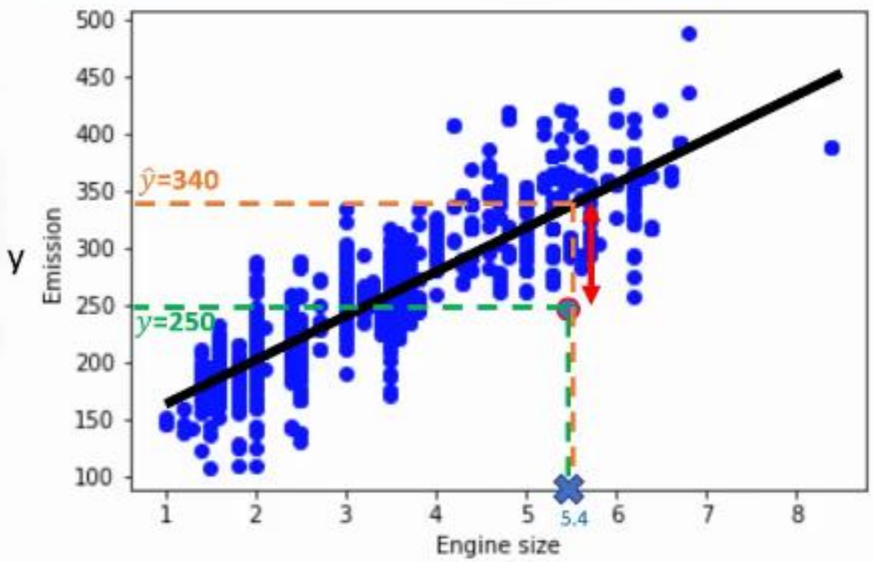


Bagaimana Mencari Model yang Fit

$x_1 = 2.4$ independent variable
$y = 250$ actual Co2 emission of x1

$\hat{y} = \theta_0 + \theta_1 x_1$
$\hat{y} = 340$ the predicted emission of x1

Error = $y - \hat{y}$
$= 250 - 340$
$= -90$



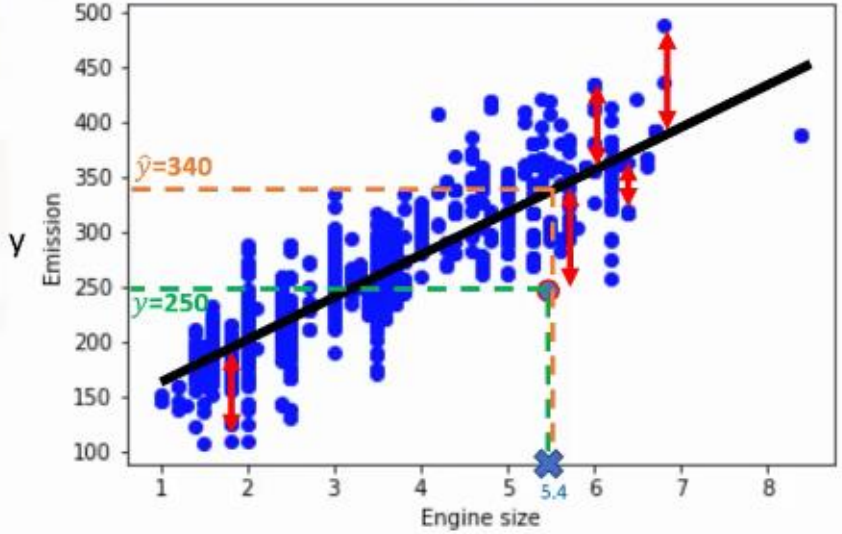
Bagaimana Mencari Model yang Fit

$x_1 = 2.4$ independent variable
$y = 250$ actual Co2 emission of x1

$\hat{y} = \theta_0 + \theta_1 x_1$
$\hat{y} = 340$ the predicted emission of x1

Error = $y - \hat{y}$
= $250 - 340$
= -90

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$



Mean Squared Error

Bagaimana Mencari Model yang Fit

$x_1 = 2.4$ independent variable

$y = 250$ actual CO_2 emission of x1

$\hat{y} = \theta_0 + \theta_1 x_1$

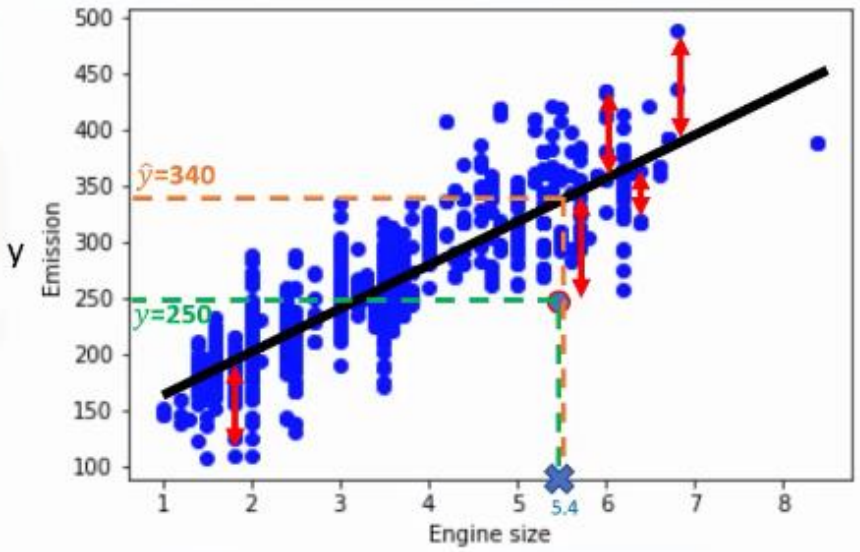
$\hat{y} = 340$ the predicted emission of x1

Error = $y - \hat{y}$

$= 250 - 340$

$= -90$

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$



Estimasi Parameter-Parameter

$$\hat{y} = \boxed{\theta_0 + \theta_1} x_1$$

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267

X_1 (ENGINESIZE), y (CO2EMISSIONS)

Estimasi Parameter-Parameter

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267

X_1 y

$$\hat{y} = \theta_0 + \theta_1 x_1$$

$$\theta_1 = \frac{\sum_{i=1}^{s}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{s}(x_i - \bar{x})^2}$$

$$\theta_0 = \bar{y} - \theta_1 \bar{x}$$

Estimasi Parameter-Parameter

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267

X_1 y

$$\hat{y} = \theta_0 + \theta_1 x_1$$

$$\theta_1 = \frac{\sum_{i=1}^{s}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{s}(x_i - \bar{x})^2}$$

$$\boxed{\bar{x} = (2.0 + 2.4 + 1.5 + \ldots)/9 = 3.34}$$

$$\bar{y} = (196 + 221 + 136 + \ldots)/9 = 256$$

$$\theta_0 = \bar{y} - \theta_1 \bar{x}$$

Estimasi Parameter-Parameter

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267

X_1 y

$$\hat{y} = \theta_0 + \theta_1 x_1$$

$$\theta_1 = \frac{\sum_{i=1}^{s}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{s}(x_i - \bar{x})^2}$$

$\bar{x} = (2.0 + 2.4 + 1.5 + \ldots)/9 = 3.34$

$\bar{y} = (196 + 221 + 136 + \ldots)/9 = 256$

$$\theta_0 = \bar{y} - \theta_1 \bar{x}$$

Estimasi Parameter-Parameter

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267

X_1 y

$$\hat{y} = \theta_0 + \theta_1 x_1$$

$$\theta_1 = \frac{\sum_{i=1}^{s}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{s}(x_i - \bar{x})^2}$$

$\bar{x} = (2.0 + 2.4 + 1.5 + \ldots)/9 = 3.34$

$\bar{y} = (196 + 221 + 136 + \ldots)/9 = 256$

$$\theta_0 = \bar{y} - \theta_1 \bar{x}$$

Estimasi Parameter-Parameter

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267

X_1 (ENGINESIZE)

y (CO2EMISSIONS)

$$\hat{y} = \theta_0 + \theta_1 x_1$$

$$\theta_1 = \frac{\sum_{i=1}^{s}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{s}(x_i - \bar{x})^2}$$

$\bar{x} = (2.0 + 2.4 + 1.5 + \ldots)/9 = 3.34$

$\bar{y} = (196 + 221 + 136 + \ldots)/9 = 256$

$$\boxed{\theta_1 = \frac{(2.0 - 3.34)(196 - 256) + (2.4 - 3.34)(221 - 256) + \ldots}{(2.0 - 3.34)^2 + (2.4 - 3.34)^2 + \ldots}}$$

$$\theta_0 = \bar{y} - \theta_1 \bar{x}$$

Estimasi Parameter-Parameter

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267

X_1 (ENGINESIZE), y (CO2EMISSIONS)

$$\hat{y} = \theta_0 + \theta_1 x_1$$

$$\theta_1 = \frac{\sum_{i=1}^{s}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{s}(x_i - \bar{x})^2}$$

$\bar{x} = (2.0 + 2.4 + 1.5 + \ldots)/9 = 3.34$

$\bar{y} = (196 + 221 + 136 + \ldots)/9 = 256$

$$\theta_1 = \frac{(2.0 - 3.34)(196 - 256) + (2.4 - 3.34)(221 - 256) + \ldots}{(2.0 - 3.34)^2 + (2.4 - 3.34)^2 + \ldots}$$

$$\theta_0 = \bar{y} - \theta_1 \bar{x}$$

Estimasi Parameter-Parameter

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267

X_1 (ENGINESIZE), y (CO2EMISSIONS)

$$\hat{y} = \theta_0 + \theta_1 x_1$$

$$\theta_1 = \frac{\sum_{i=1}^{s}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{s}(x_i - \bar{x})^2}$$

$\bar{x} = (2.0 + 2.4 + 1.5 + \dots)/9 = 3.34$

$\bar{y} = (196 + 221 + 136 + \dots)/9 = 256$

$$\theta_1 = \frac{(2.0 - 3.34)(196 - 256) + (2.4 - 3.34)(221 - 256) + \dots}{(2.0 - 3.34)^2 + (2.4 - 3.34)^2 + \dots}$$

$\boxed{\theta_1 = 39}$

$$\theta_0 = \bar{y} - \theta_1 \bar{x}$$

Estimasi Parameter-Parameter

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267

X_1 (ENGINESIZE) y (CO2EMISSIONS)

$$\hat{y} = \theta_0 + \theta_1 x_1$$

$$\theta_1 = \frac{\sum_{i=1}^{s}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{s}(x_i - \bar{x})^2}$$

$\bar{x} = (2.0 + 2.4 + 1.5 + \ldots)/9 = 3.34$

$\bar{y} = (196 + 221 + 136 + \ldots)/9 = 256$

$$\theta_1 = \frac{(2.0 - 3.34)(196 - 256) + (2.4 - 3.34)(221 - 256) + \ldots}{(2.0 - 3.34)^2 + (2.4 - 3.34)^2 + \ldots}$$

$\theta_1 = 39$

$$\theta_0 = \bar{y} - \theta_1 \bar{x}$$

$$\theta_0 = 256 - 39 * 3.34$$

$$\boxed{\theta_0 = 125.74}$$

Estimasi Parameter-Parameter

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267

X_1 (ENGINESIZE), y (CO2EMISSIONS)

$$\hat{y} = \theta_0 + \theta_1 x_1$$

$$\theta_1 = \frac{\sum_{i=1}^{s}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{s}(x_i - \bar{x})^2}$$

$$\bar{x} = (2.0 + 2.4 + 1.5 + \ldots)/9 = 3.34$$

$$\bar{y} = (196 + 221 + 136 + \ldots)/9 = 256$$

$$\theta_1 = \frac{(2.0 - 3.34)(196 - 256) + (2.4 - 3.34)(221 - 256) +}{(2.0 - 3.34)^2 + (2.4 - 3.34)^2 + \ldots}$$

$$\theta_1 = 39$$

$$\theta_0 = \bar{y} - \theta_1 \bar{x}$$

$$\theta_0 = 256 - 39 * 3.34$$

$$\theta_0 = 125.74$$

$$\boxed{\hat{y} = 125.74 + 39x_1}$$

Prediksi dengan Model Garis (*line model*)

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	?

Prediksi dengan Model Garis (*line model*)

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	?

$$\hat{y} = \theta_0 + \theta_1 x_1$$

$$Co2Emission = \theta_0 + \theta_1\, EngineSize$$

$$\boxed{Co2Emission = 125 + 39\, EngineSize}$$

Prediksi dengan Model Garis (*line model*)

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	?

$$\hat{y} = \theta_0 + \theta_1 x_1$$

$$Co2Emission = \theta_0 + \theta_1 \, EngineSize$$

$$Co2Emission = 125 + 39 \, EngineSize$$

$$Co2Emission = 125 + 39 \times 2.4$$

$$Co2Emission = 218.6$$

Metode tadi adalah metode statistika.

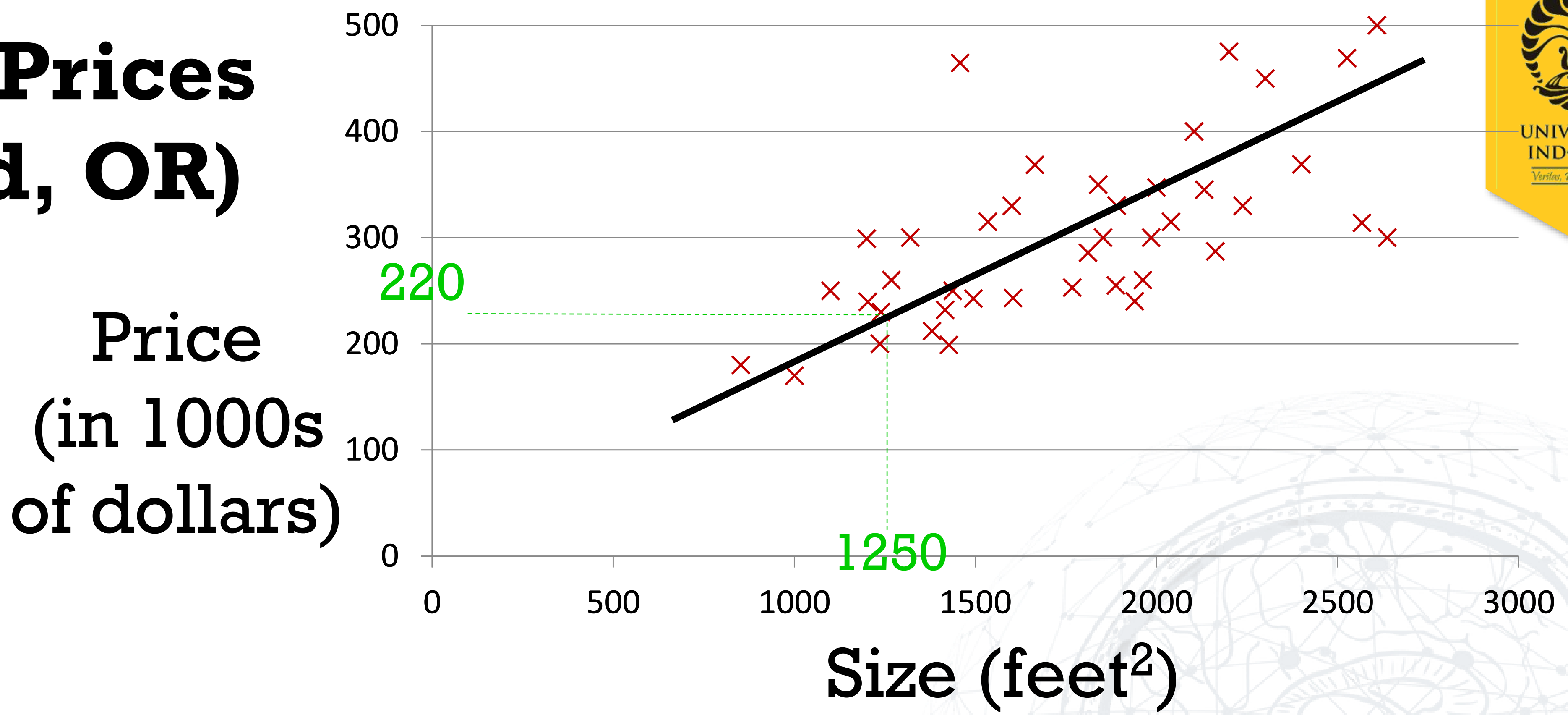
Bagaimana metode machine learning bekerja?

- The following slides are taken from:
- https://www.coursera.org/learn/machine-learning/home/welcome

- All credit to Prof. Andrew Ng

Regression Model

Housing Prices (Portland, OR)



Supervised Learning

Given the "right answer" for each example in the data.

Regression Problem

Predict real-valued output

Classification : Discrete-valued output

Training set of housing prices (Portland, OR)

Size in feet² (x)	Price ($) in 1000's (y)
2104	460
1416	232
1534	315
852	178
…	…

m

Notation:

m = Number of training examples
x's = "input" variable / features
y's = "output" variable / "target" variable

$x^{(1)} = 2104$
$x^{(2)} = 1416$
$y^{(1)} = 460$

(x, y) – one training example
$(x^{(i)}, y^{(i)})$ – ith trainingg example

Training Set

↓

Learning Algorithm

↓

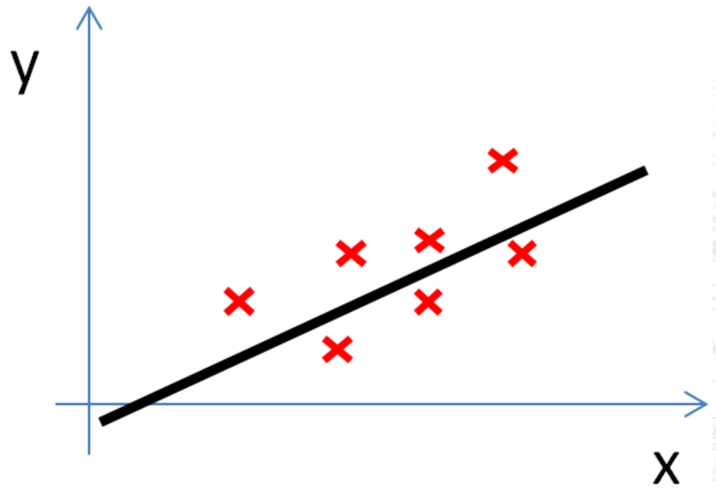
Size of house → h → Estimated price

x — hypothesis — Estimated value

h maps from x's to y's

How do we represent h ?

$$h_{\Theta}(x) = \Theta_0 + \Theta_1 x$$



Linear regression with one variable

Univariate linear regression.

One variable

COST FUNCTION

Training Set

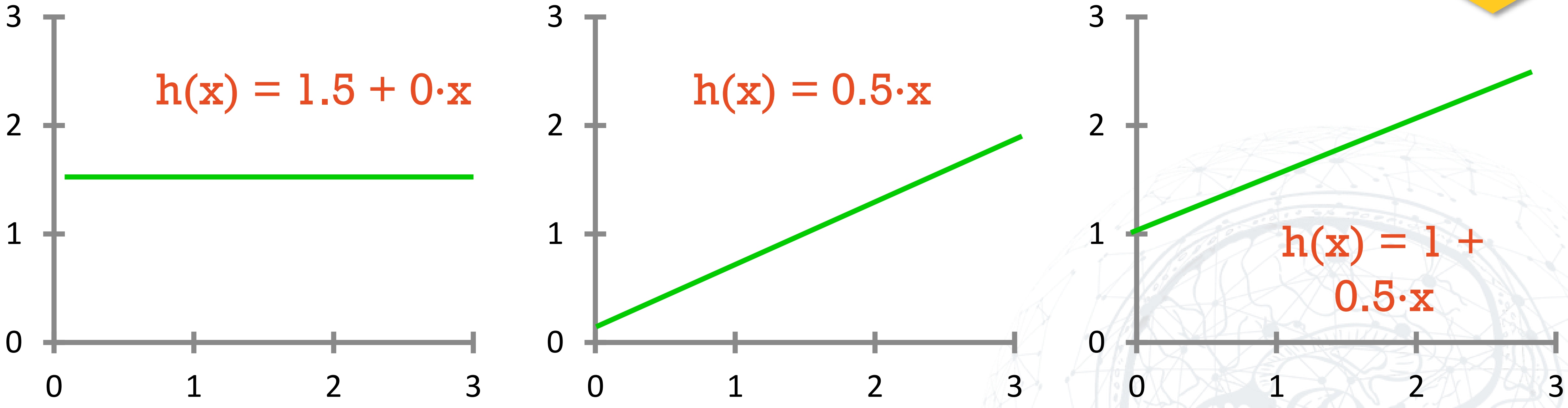
Size in feet² (x)	Price ($) in 1000's (y)
2104	460
1416	232
1534	315
852	178
…	…

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

θ_i's: Parameters

How to choose θ_i 's ?

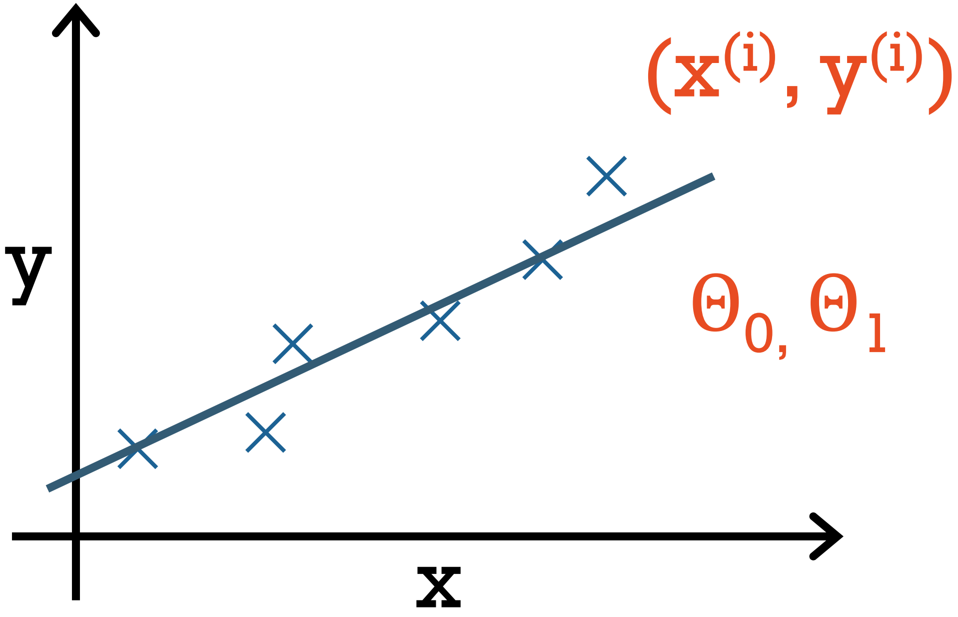
$$h_\theta(x) = \theta_0 + \theta_1 x$$



$\theta_0 = 1.5$
$\theta_1 = 0$

$\theta_0 = 0$
$\theta_1 = 0.5$

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



$$\underset{\Theta_0\,\Theta_1}{\text{minimize}}\ \frac{1}{2m}\sum_{i=1}^{m}\left(h_\Theta(x^{(i)}) - y^{(i)}\right)^2$$

$$h(x) = \Theta_0 + \Theta_1 x^{(i)}$$

$$J(\Theta_0, \Theta_1) = \frac{1}{2m}\sum_{i=1}^{m}\left(h_\Theta(x^{(i)}) - y^{(i)}\right)^2$$

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

$\underset{\Theta_0\,\Theta_1}{\text{Minimize}}\ J(\Theta_0, \Theta_1)$: Cost Function

Squared error function

Simplified

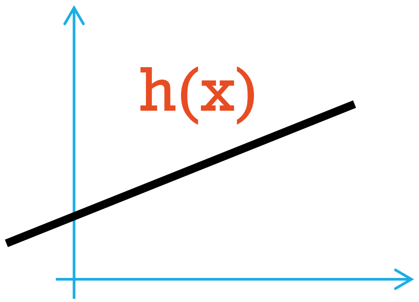
Hypothesis:

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

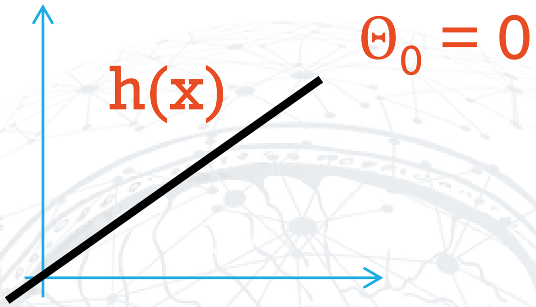
$$h_\theta(x) = \theta_1 x$$

Parameters:

$$\theta_0, \theta_1$$



$$\theta_1$$



Cost Function:

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

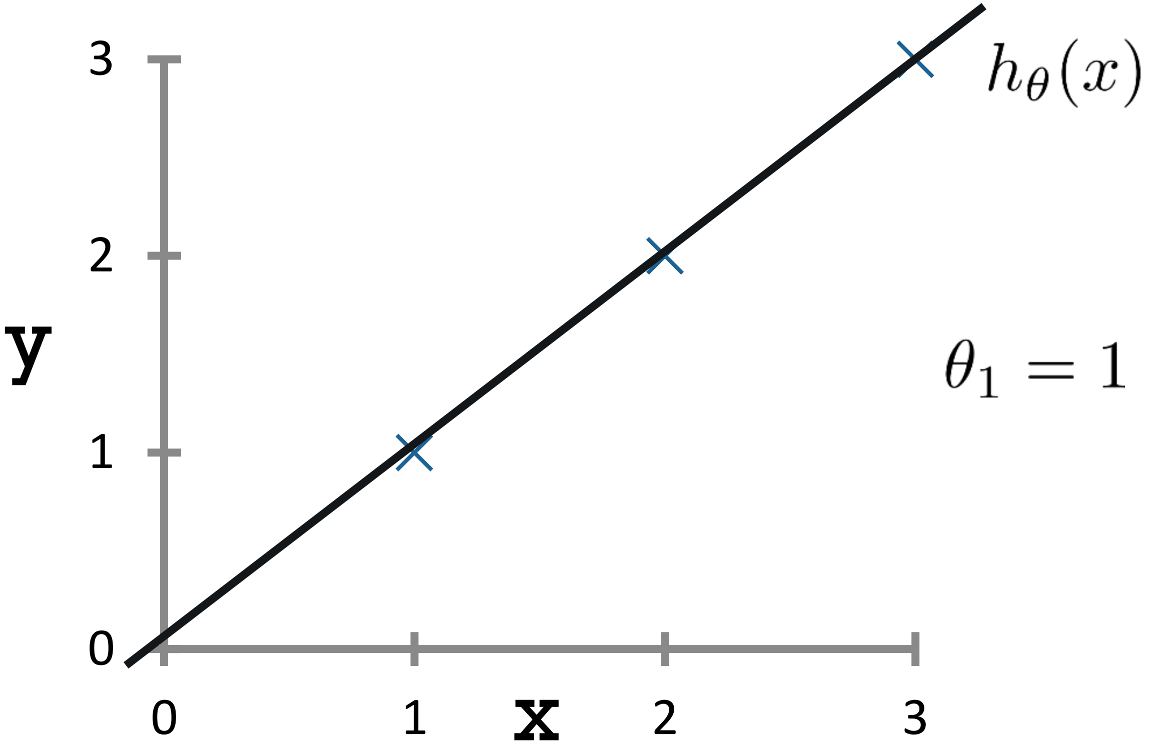
$$J(\theta_1) = \frac{1}{2m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)}\right)^2$$

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

$$\underset{\theta_1}{\text{minimize}}\ J(\theta_1)$$

$h_\theta(x)$ $\quad h_\theta(x) = \theta_1 x$

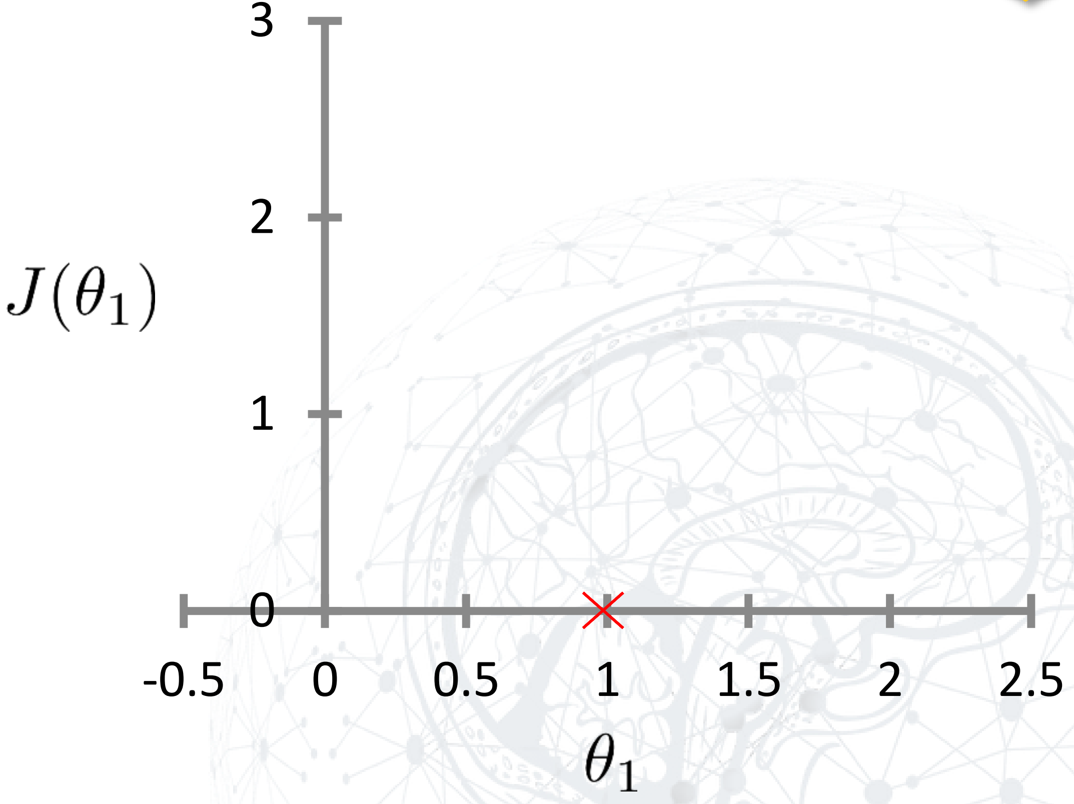
(for fixed θ_1, this is a function of x)



$$J(\Theta_1) = \frac{1}{2m}\sum_{i=1}^{m}(h_\Theta(x^{(i)}) - y^{(i)})^2$$
$$= \frac{1}{2m}\sum_{i=1}^{m}(\Theta_1 x^{(i)} - y^{(i)})^2$$
$$= \frac{1}{2m}(0^2 + 0^2 + 0^2)$$

$J(\theta_1)$

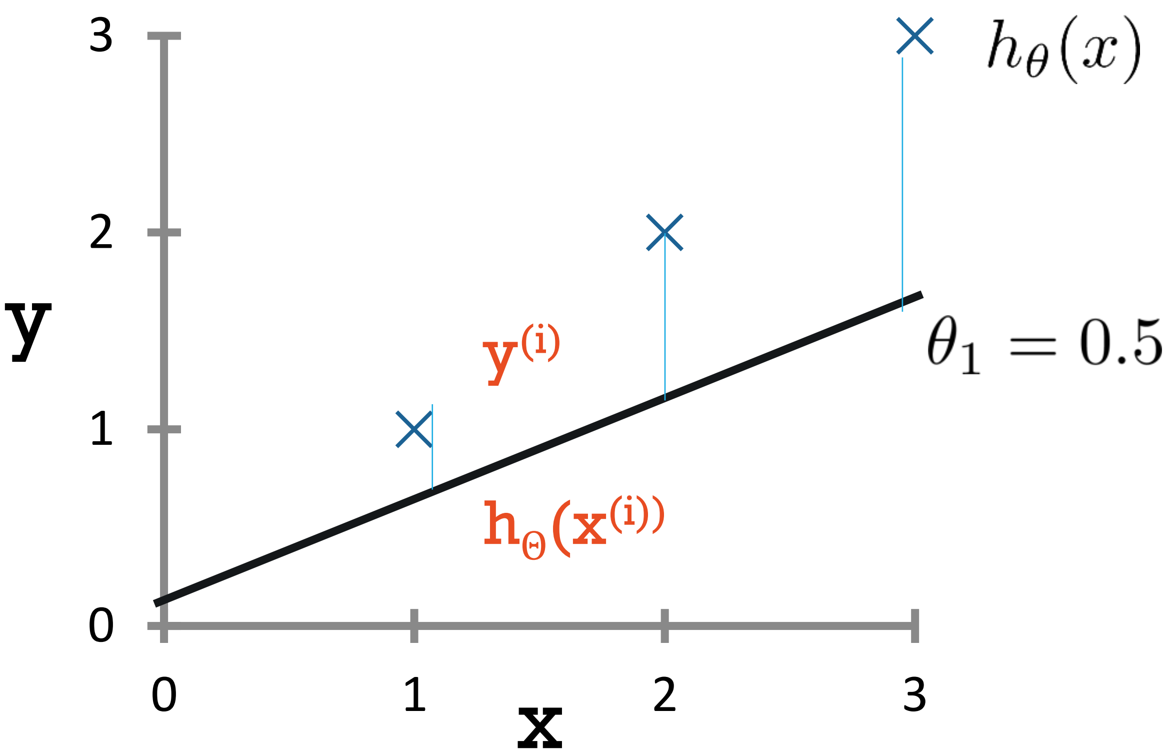
(function of the parameter θ_1)



$J(1) = 0$

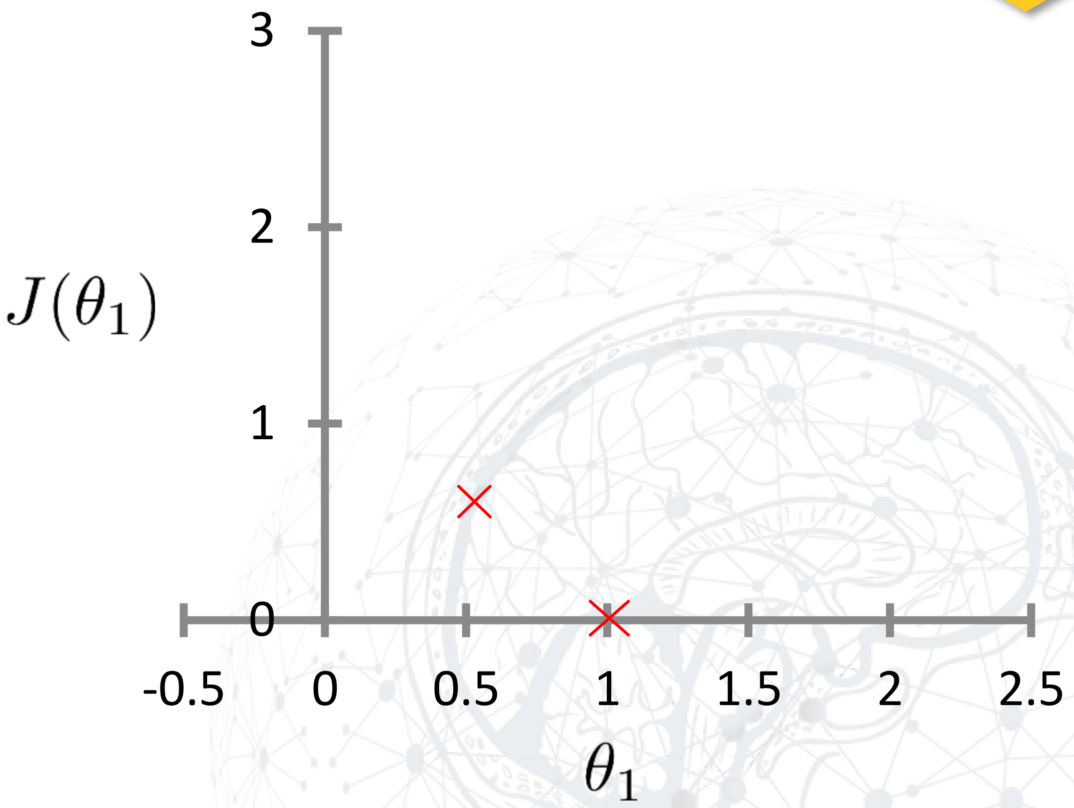
$h_\theta(x)$

(for fixed θ_1, this is a function of x)



$J(\theta_1)$

(function of the parameter θ_1)

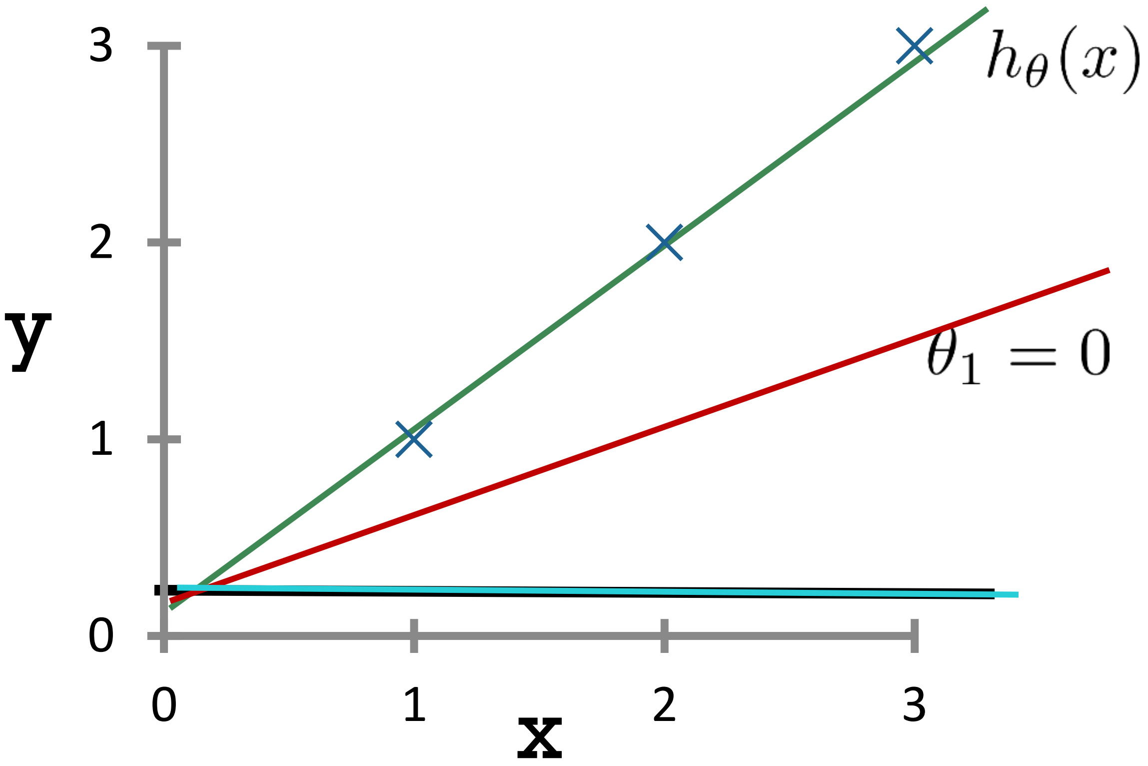


$$J(0.5) = \frac{1}{2\cdot 3}\sum_{i=1}^{3}[(0.5-1)^2 + (1-2)^2 + (1.5-3)^2]$$

$$= \frac{1}{6}\cdot(3.5) = 0.58$$

$h_\theta(x)$

(for fixed θ_1 , this is a function of x)

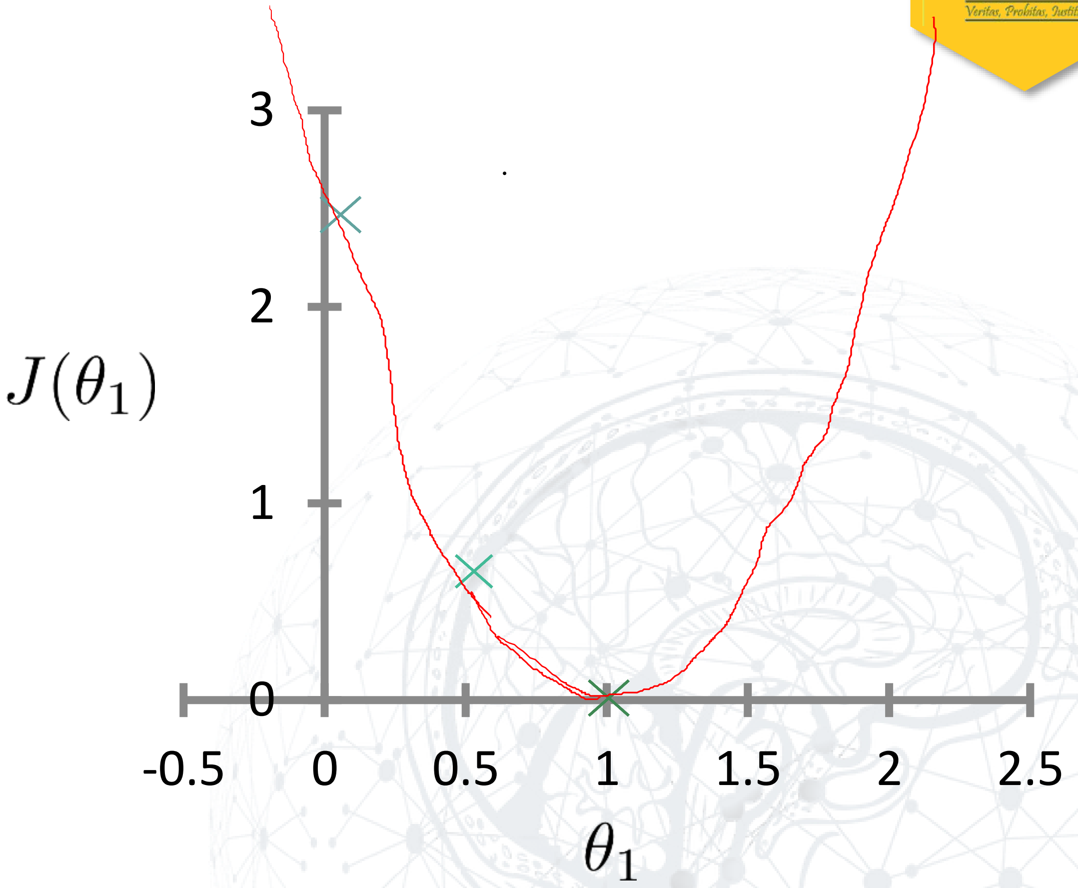


$$J(0) = \frac{1}{2 \cdot 3} \sum_{i=1}^{3} [1^2 + 2^2 + 3^2]$$

$$= \frac{1}{6} \cdot 14 = 2.3$$

$J(\theta_1)$

(function of the parameter θ_1)



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

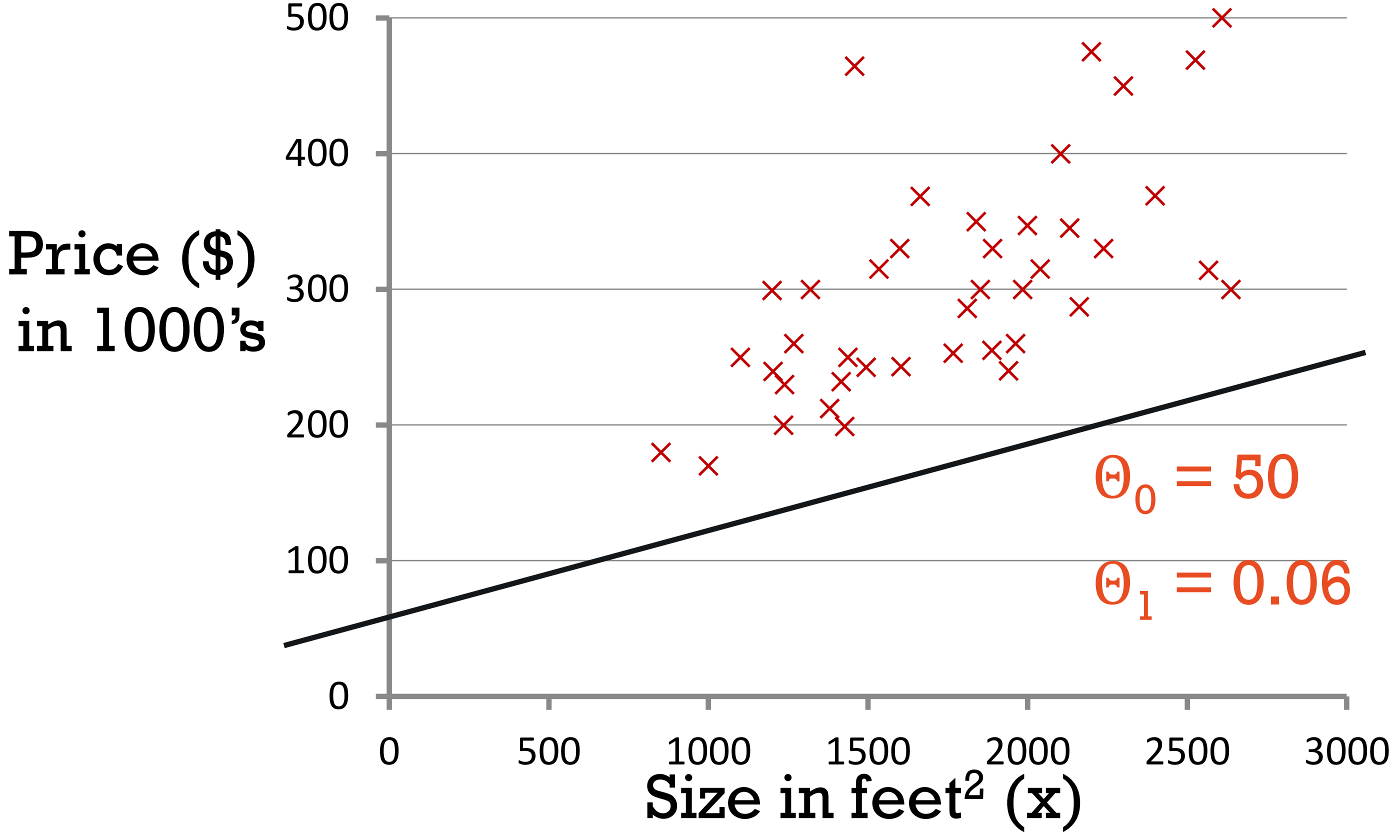
Parameters: $$\theta_0, \theta_1$$

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

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

$h_\theta(x)$

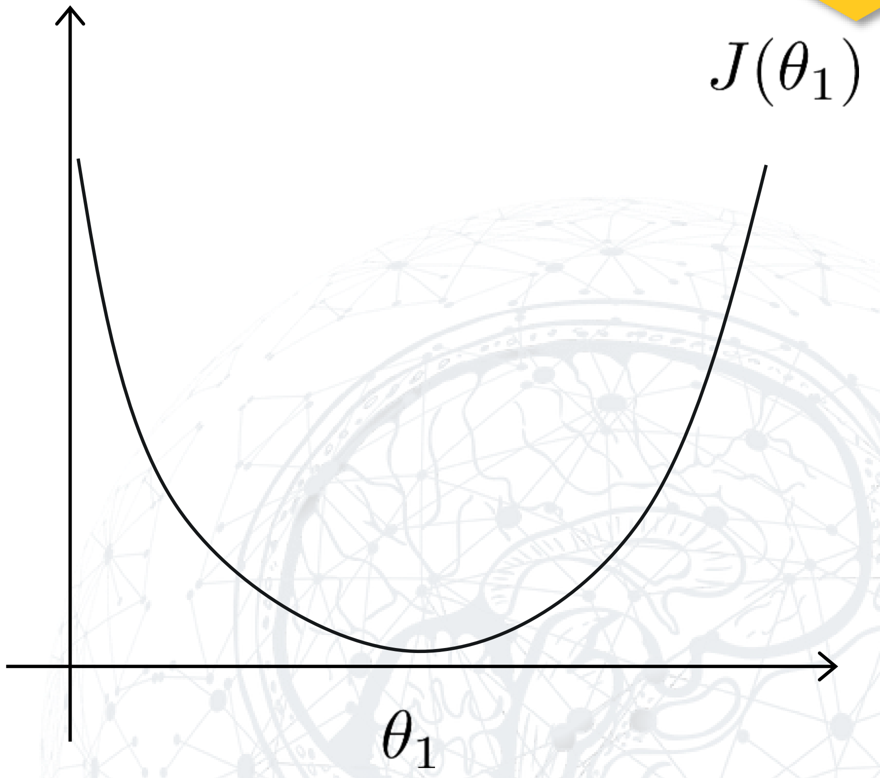
(for fixed θ_0, θ_1, this is a function of x)



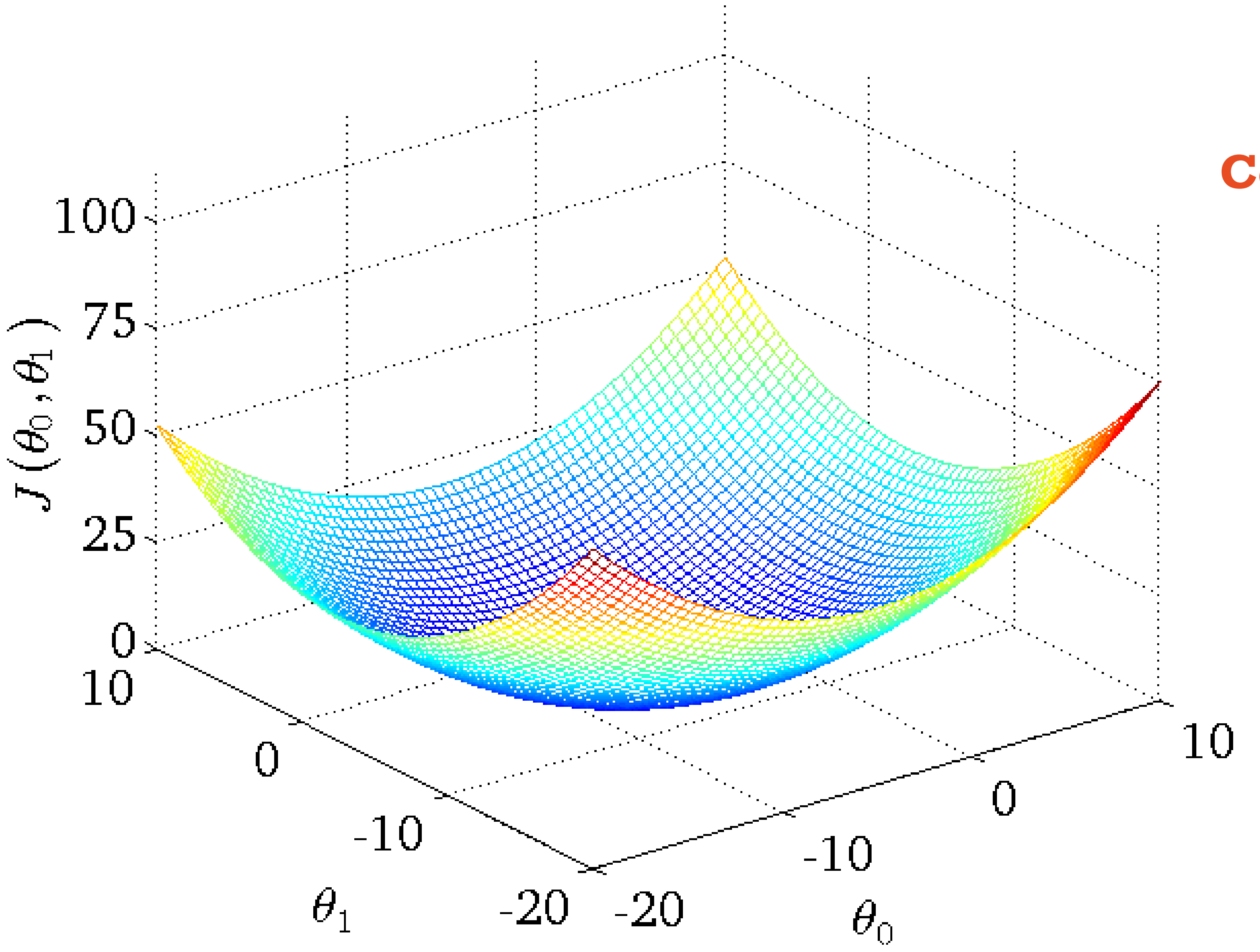
$$h_\theta(x) = 50 + 0.06x$$

$J(\theta_0, \theta_1)$

(function of the parameters θ_0, θ_1)

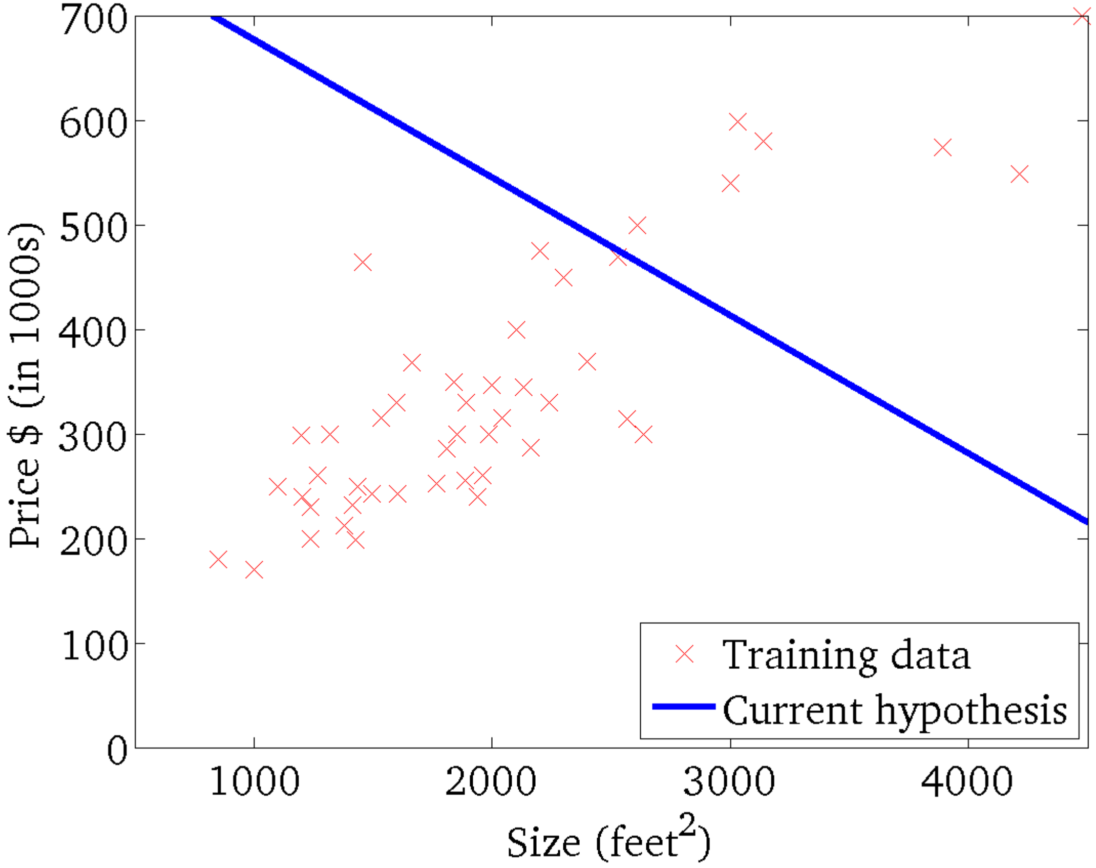


Contour plots



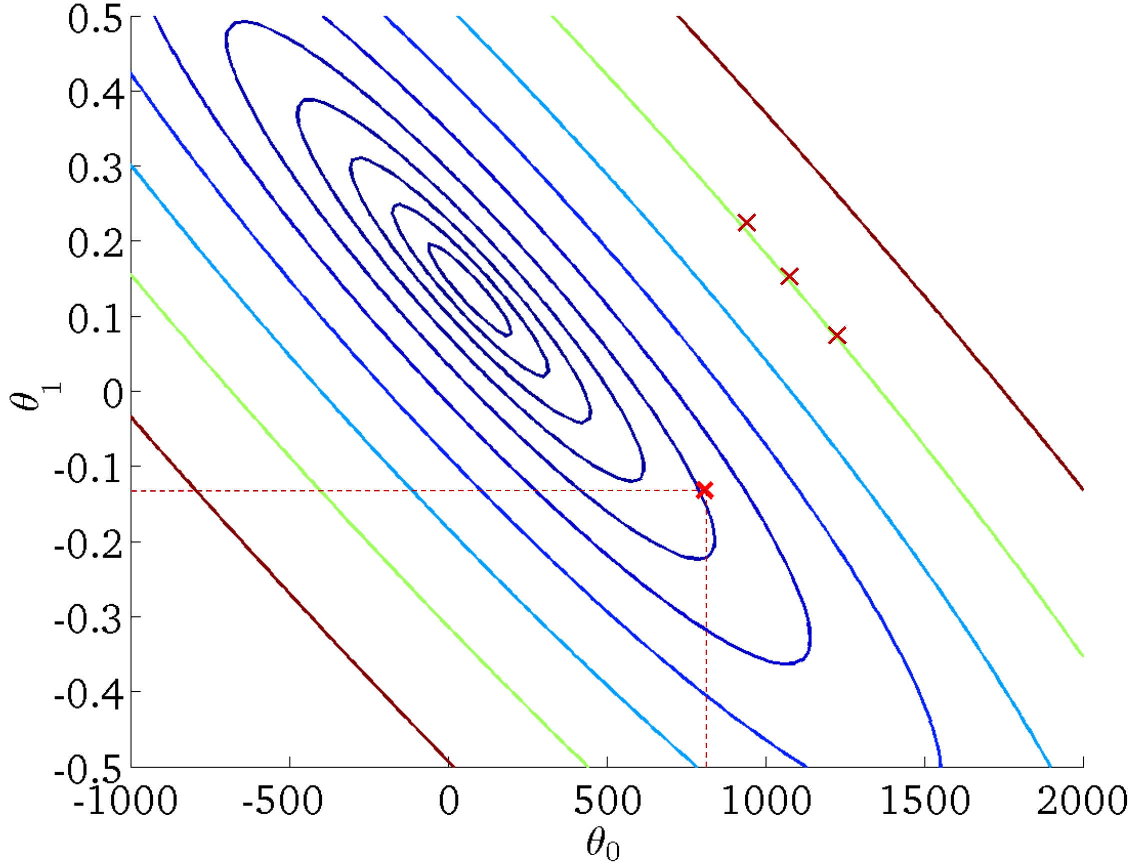
$h_\theta(x)$

(for fixed θ_0, θ_1 , this is a function of x)



$J(\theta_0, \theta_1)$

(function of the parameters θ_0, θ_1)

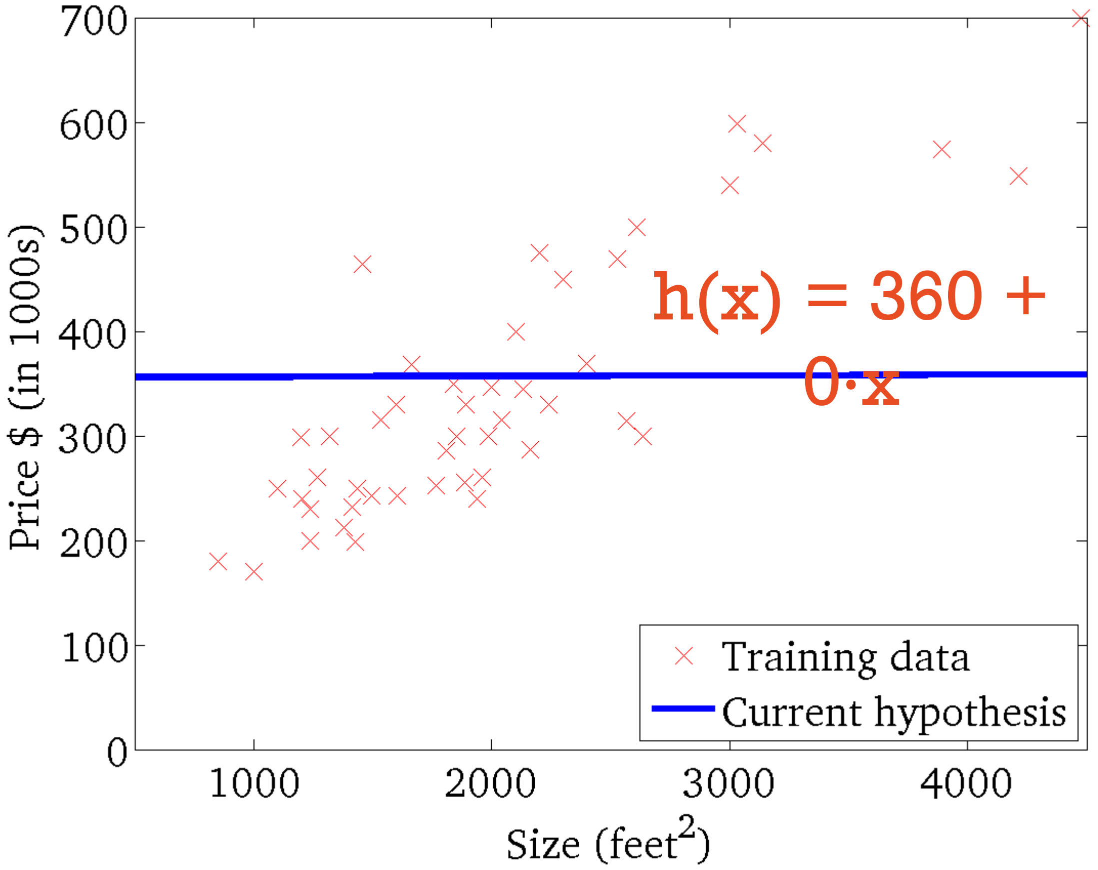


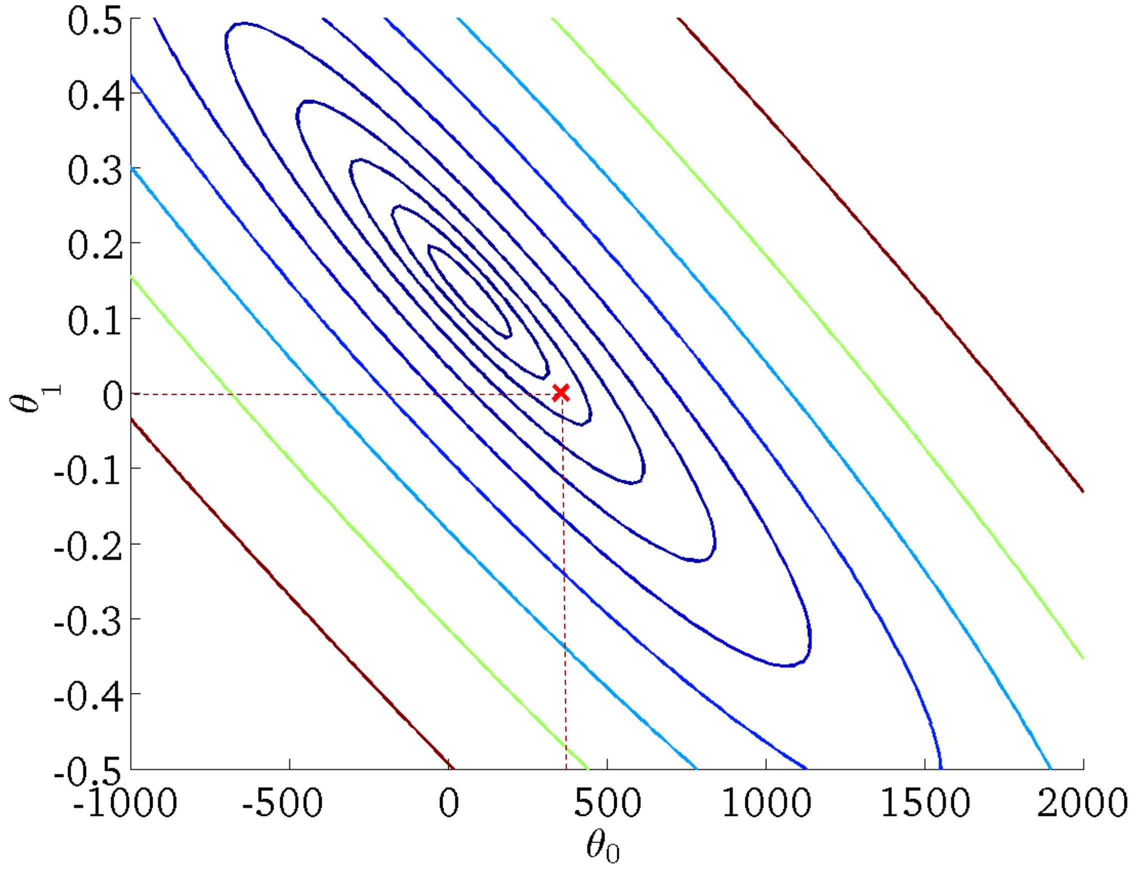
$h_\theta(x)$

(for fixed θ_0, θ_1 , this is a function of x)

$J(\theta_0, \theta_1)$

(function of the parameters θ_0, θ_1)





$\Theta_0 = 360$

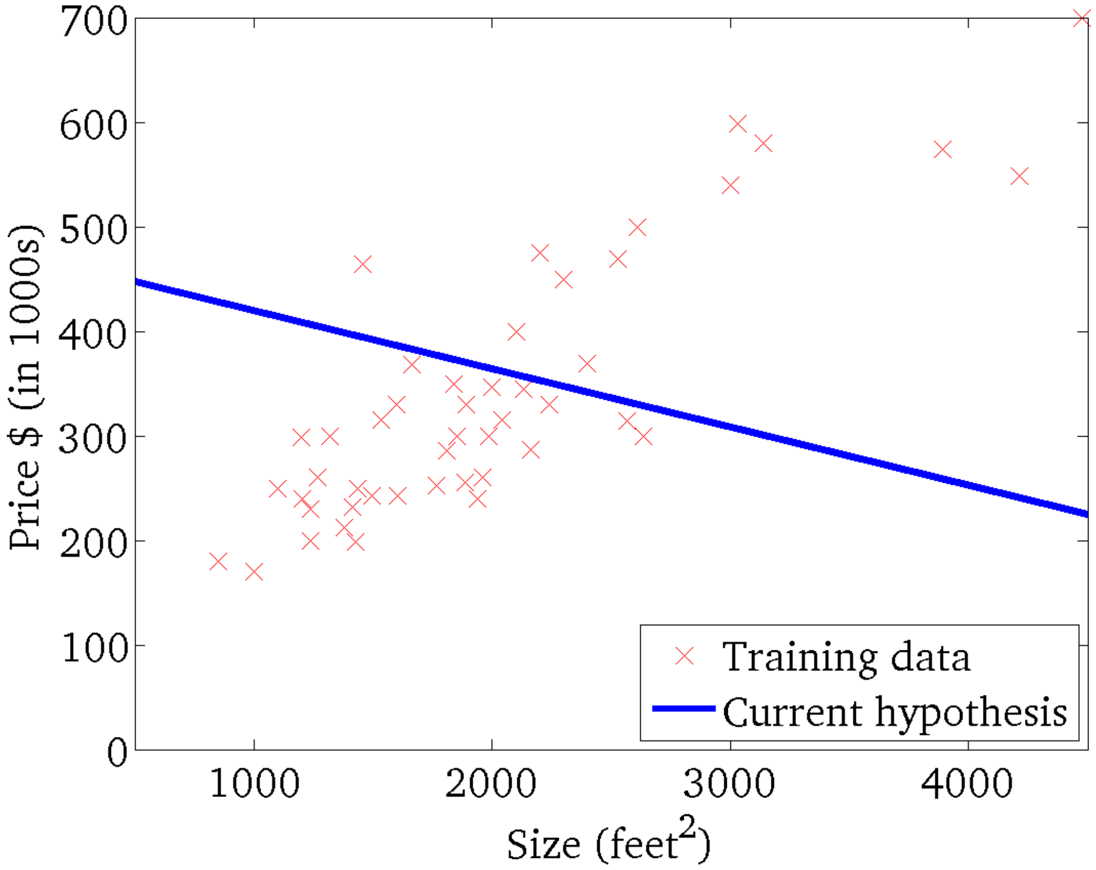
$\Theta_1 = 0$

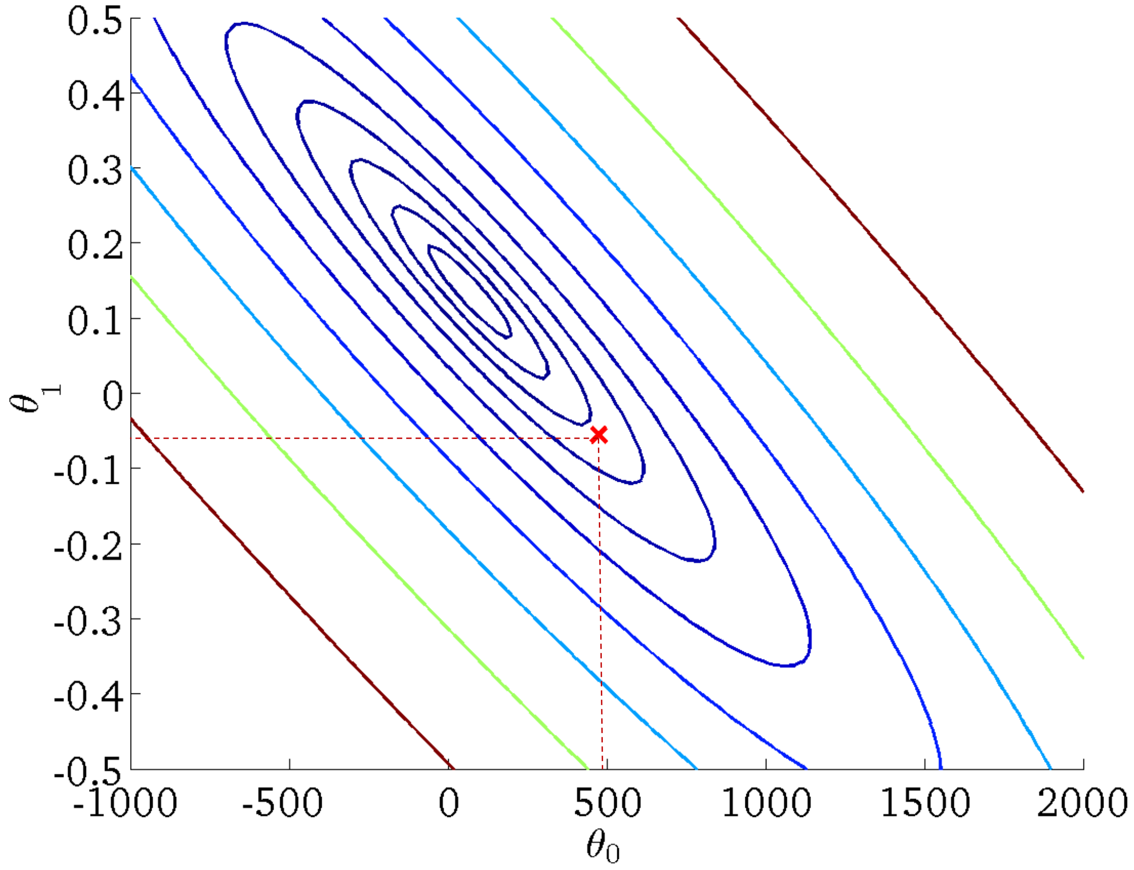
$$h_\theta(x)$$

(for fixed θ_0, θ_1 , this is a function of x)

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

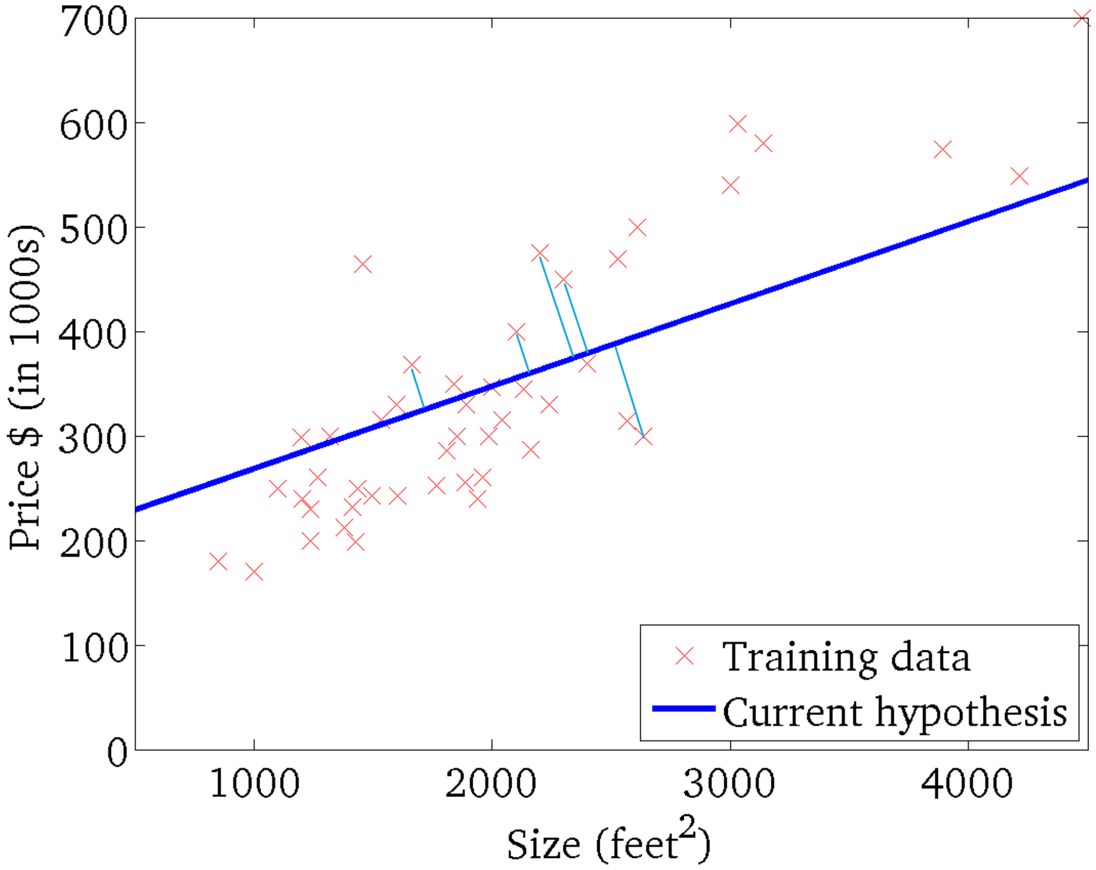
(function of the parameters θ_0, θ_1)





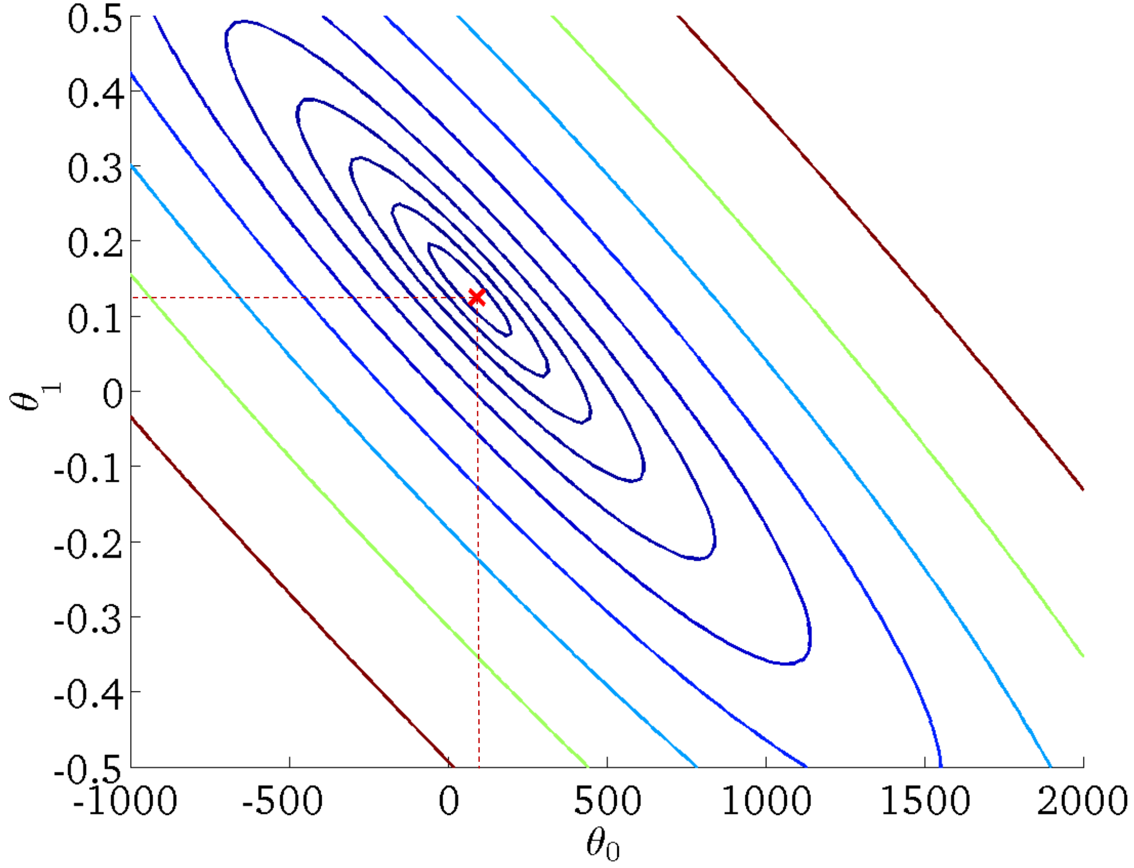
$h_\theta(x)$

(for fixed θ_0, θ_1 , this is a function of x)



$J(\theta_0, \theta_1)$

(function of the parameters θ_0, θ_1)



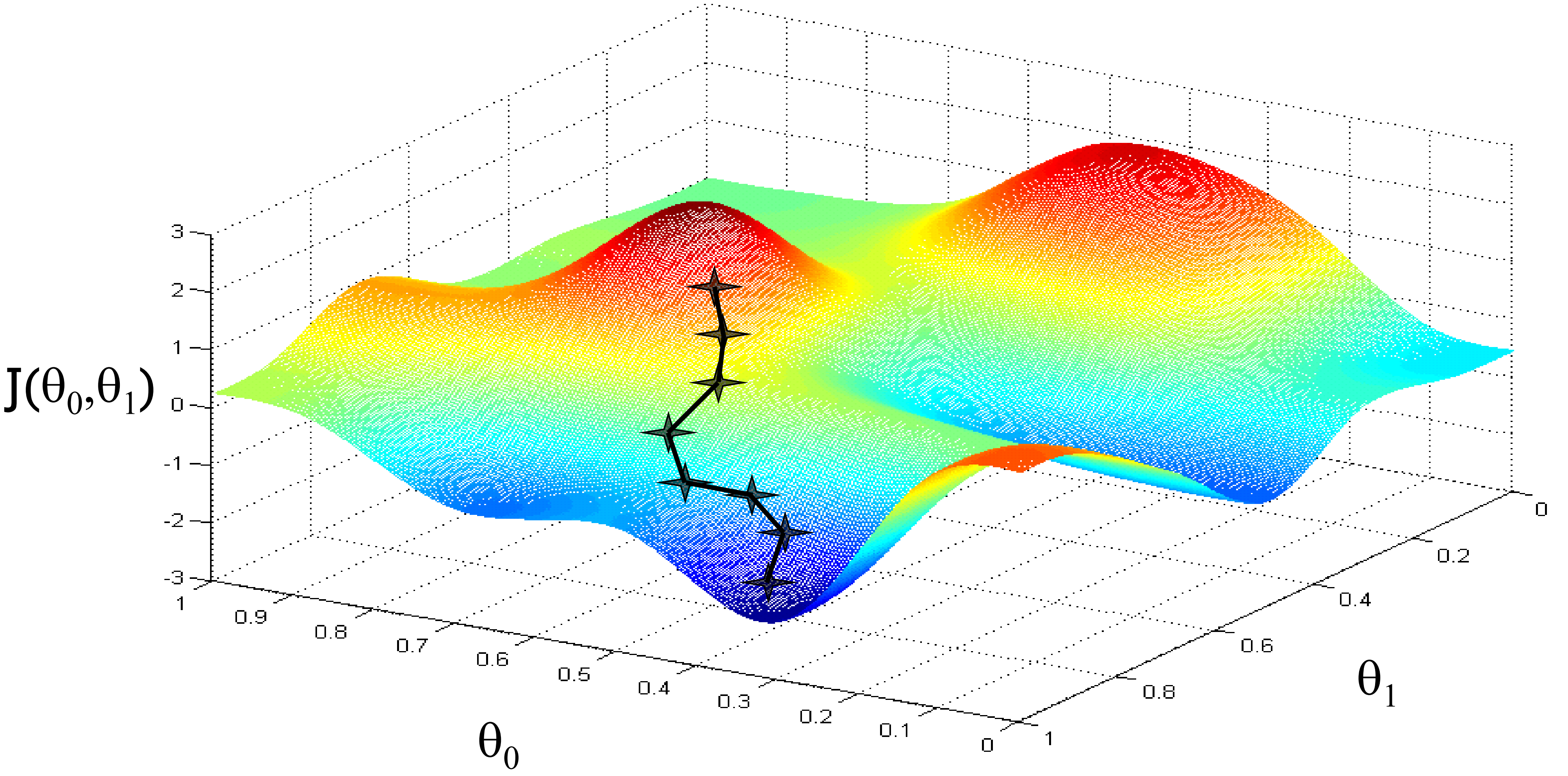
Gradient Descent

Have some function $J(\theta_0, \theta_1)$

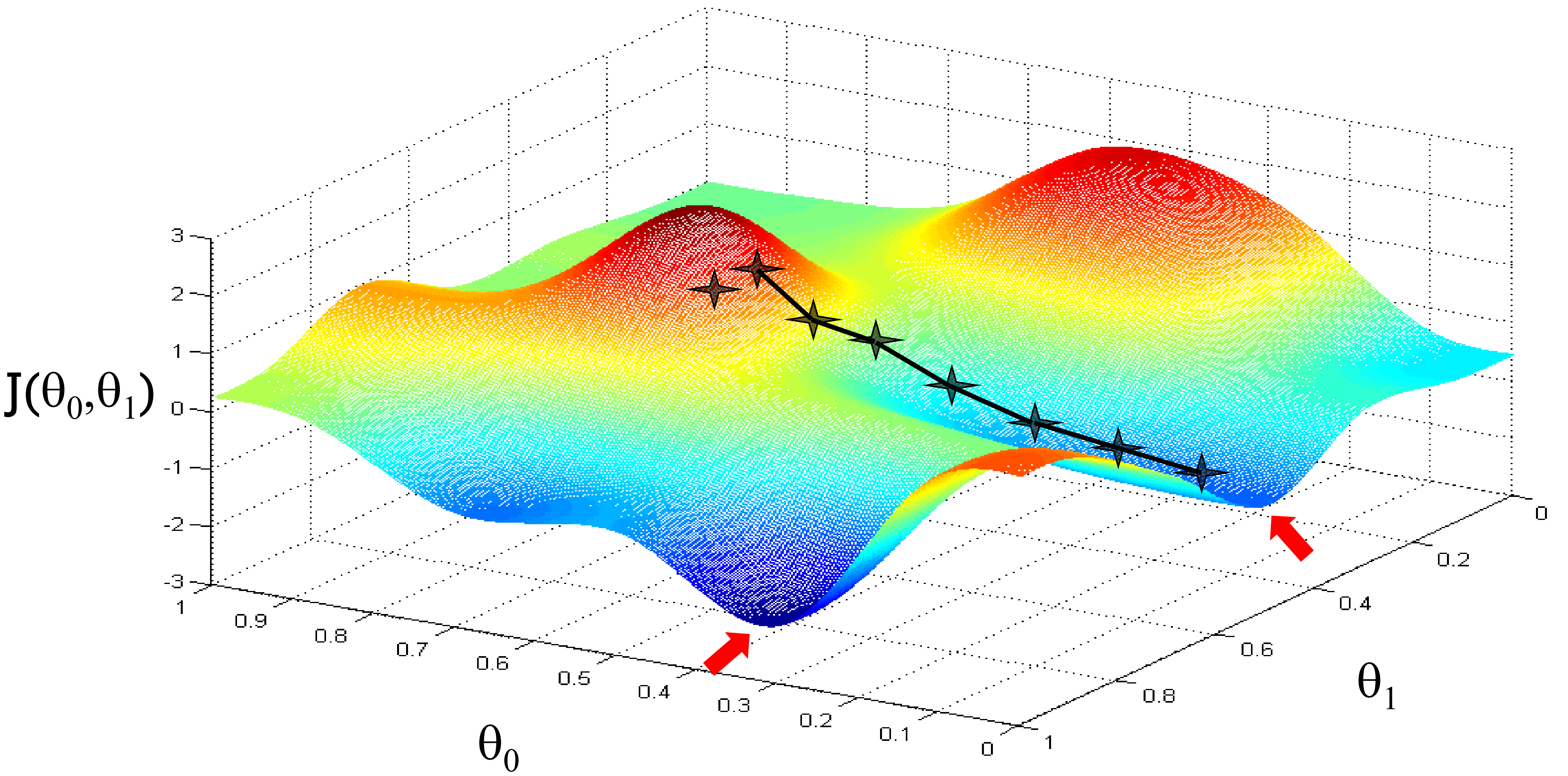
Want $\min_{\theta_0, \theta_1} J(\theta_0, \theta_1)$

Outline:

- Start with some θ_0, θ_1
- Keep changing θ_0, θ_1 to reduce $J(\theta_0, \theta_1)$ until we hopefully end up at a minimum



$J(\theta_0,\theta_1)$
3
2
1
0
-1
-2
-3
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
θ_0
0
0.2
0.4
0.6
0.8
1
θ_1


J(θ0,θ1)
θ0
θ1
3
2
1
0
-1
-2
-3
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0
0.2
0.4
0.6
0.8
1

Gradient descent algorithm

assignment

a:=b

repeat until convergence {

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

}

Learning rate

Simultaneously update Θ_0 & Θ_1

Correct: Simultaneous update

$$\text{temp0} := \theta_0 - \alpha \frac{\partial}{\partial \theta_0} J(\theta_0, \theta_1)$$
$$\text{temp1} := \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1)$$
$$\theta_0 := \text{temp0}$$
$$\theta_1 := \text{temp1}$$

Incorrect:

$$\text{temp0} := \theta_0 - \alpha \frac{\partial}{\partial \theta_0} J(\theta_0, \theta_1)$$
$$\theta_0 := \text{temp0}$$
$$\text{temp1} := \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1)$$
$$\theta_1 := \text{temp1}$$

Gradient descent algorithm

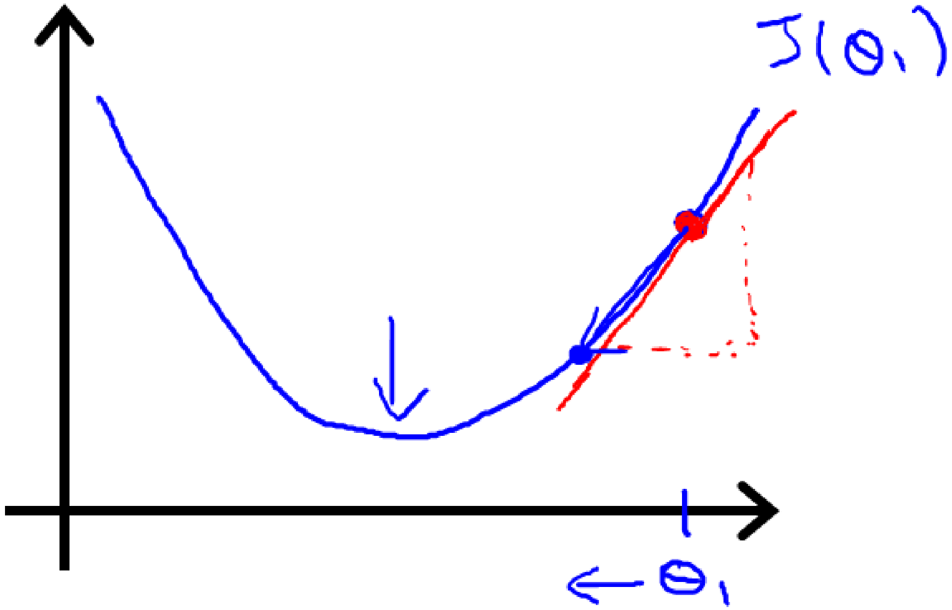
$$\text{repeat until convergence } \{$$

$$\theta_j := \theta_j - \alpha \boxed{\frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1)} \quad \text{(simultaneously update } j = 0 \text{ and } j = 1\text{)}$$

$$\}$$

Learning rate

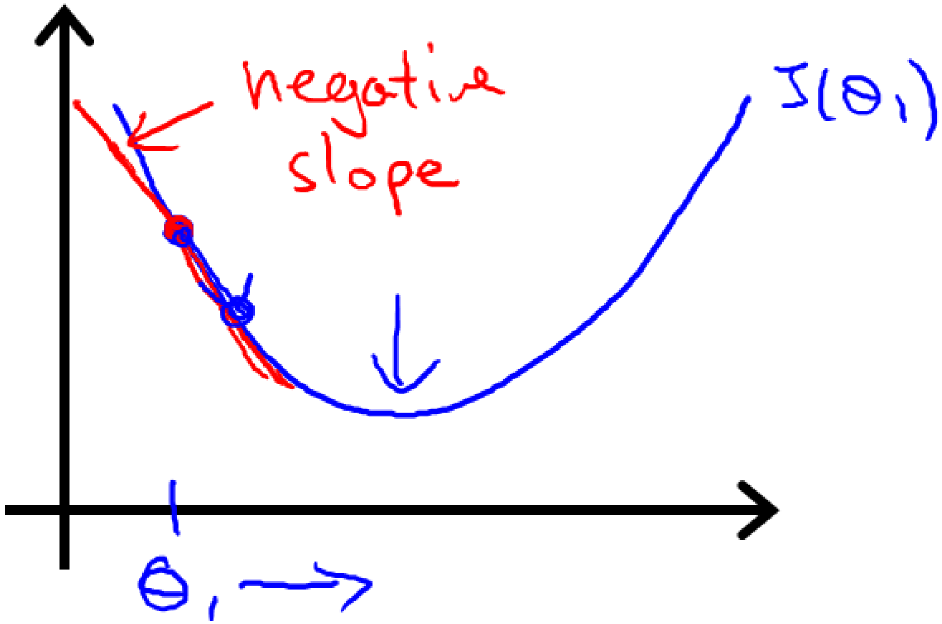
derivative



$$\Theta_1 := \Theta_1 - \alpha \boxed{\frac{d}{d\theta_1} J(\theta_1)}$$

$$\geq 0$$

$$\Theta_1 := \Theta_1 - \alpha \cdot J(positive\ number)$$



$$\Theta_1 := \Theta_1 - \alpha \boxed{\frac{d}{d\theta_1} J(\theta_1)}$$

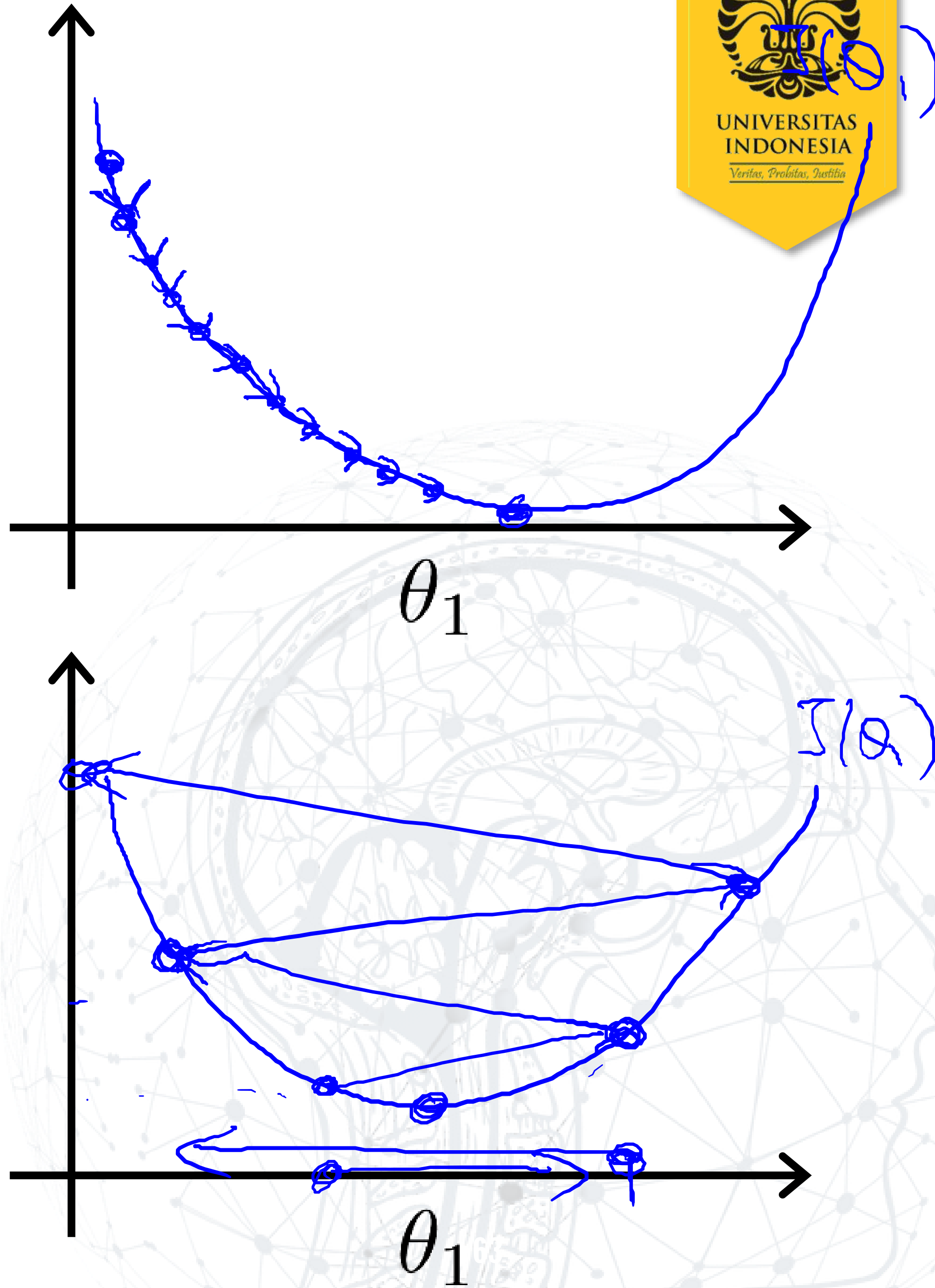
$$\leq 0$$

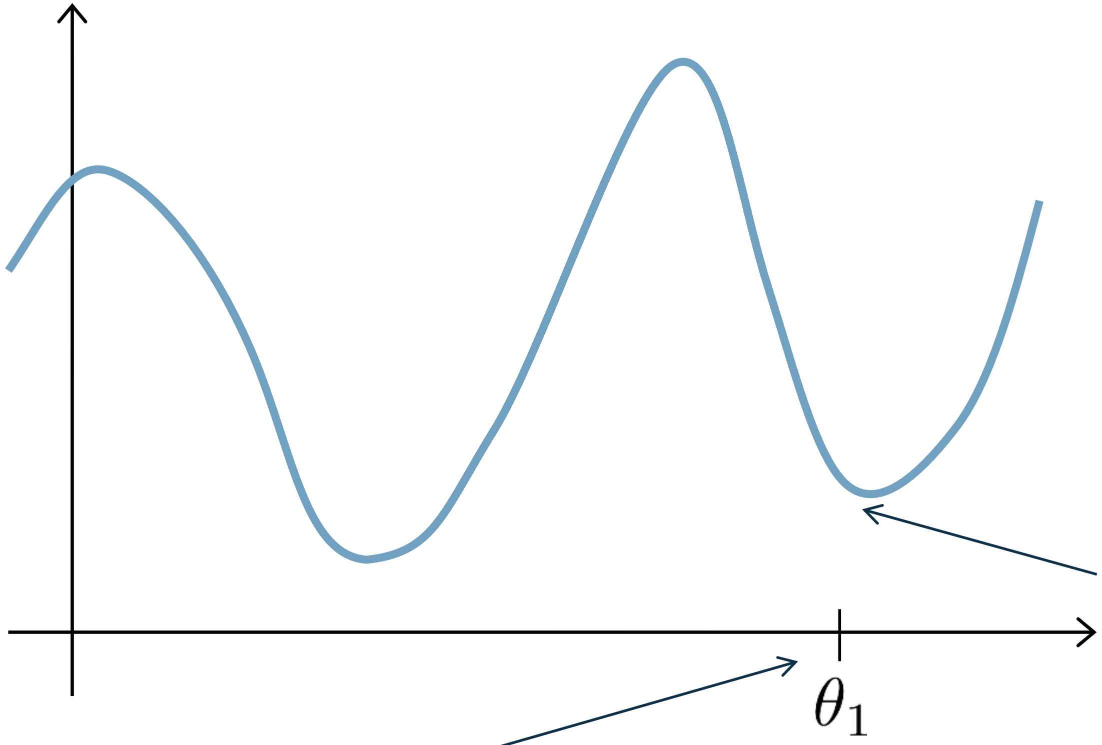
$$\Theta_1 := \Theta_1 - \alpha \cdot J(negative\ number)$$

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

If α is too small, gradient descent can be slow.

If α is too large, gradient descent can overshoot the minimum. It may fail to converge, or even diverge.





θ_1 at local optima

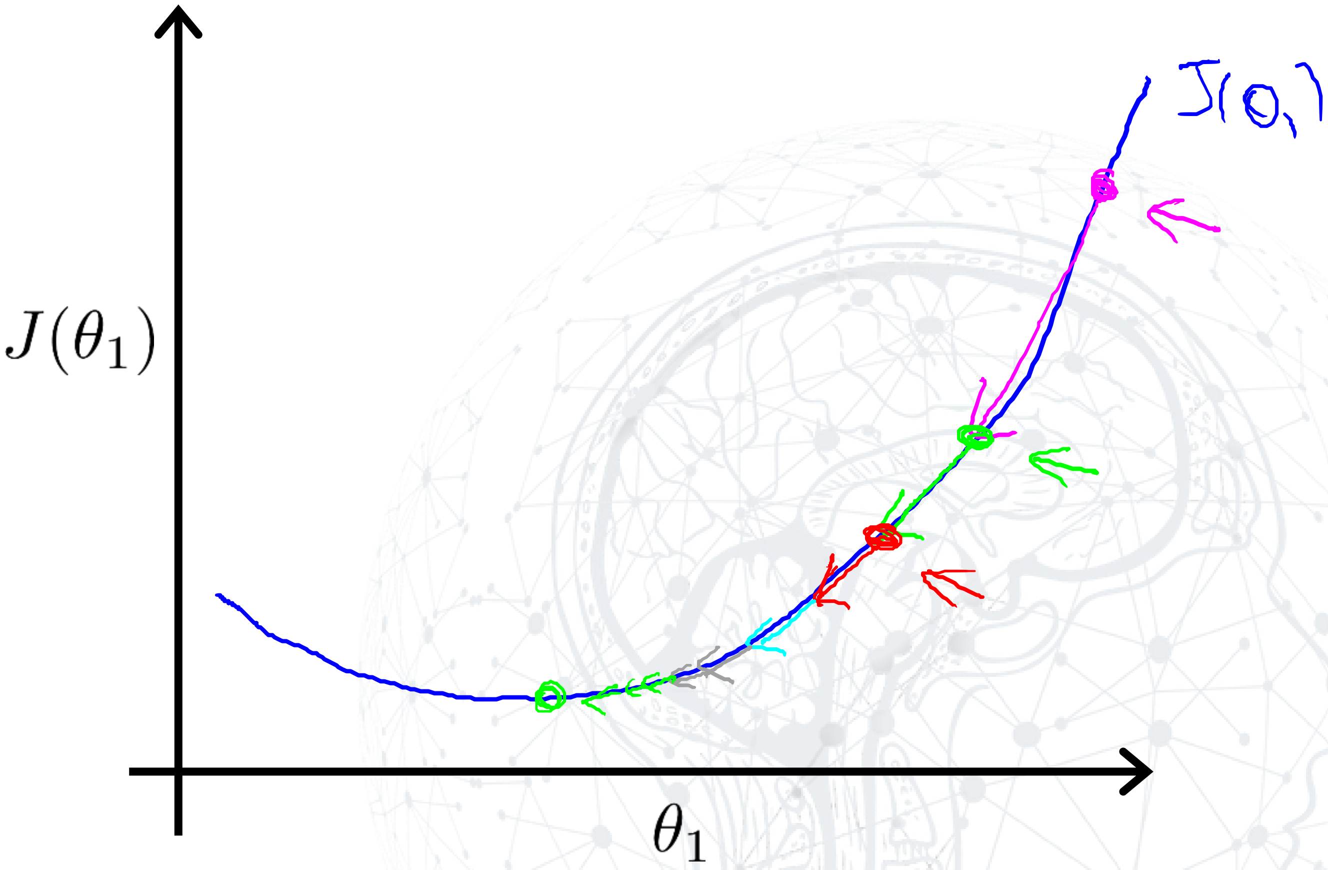
Current value of θ_1

$$\theta_1 := \theta_1 - \alpha \frac{d}{d\theta_1} J(\theta_1)$$

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

$$\theta_1 := \theta_1 - \alpha \frac{d}{d\theta_1} J(\theta_1)$$

As we approach a local minimum, gradient descent will automatically take smaller steps. So, no need to decrease α over time.



Gradient descent algorithm

repeat until convergence {

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

$$(\text{for } j = 1 \text{ and } j = 0)$$

}

Linear Regression Model

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

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

$$\frac{\partial}{\partial\theta_j} J(\theta_0, \theta_1) = \frac{d}{d\theta_j} \cdot \frac{1}{2m} \sum_{i=1}^{m} (\mathbf{h}_\Theta(\mathbf{x}^{(i)}) - \mathbf{y}^{(i)})^2$$

$$= \frac{d}{d\theta_j} \cdot \frac{1}{2m} \sum_{i=1}^{m} (\theta_0 + \theta_1 \cdot \mathbf{x}^{(i)} - \mathbf{y}^{(i)})^2$$

$$j = 0 : \frac{\partial}{\partial\theta_0} J(\theta_0, \theta_1) = \frac{1}{m} \sum_{i=1}^{m} (\mathbf{h}_\Theta(\mathbf{x}^{(i)}) - \mathbf{y}^{(i)})$$

$$j = 1 : \frac{\partial}{\partial\theta_1} J(\theta_0, \theta_1) = \frac{1}{m} \sum_{i=1}^{m} (\mathbf{h}_\Theta(\mathbf{x}^{(i)}) - \mathbf{y}^{(i)}) \cdot x^{(i)}$$

Gradient descent algorithm

$$\frac{d}{d\theta_0} \cdot J(\theta_0, \theta_1)$$

repeat until convergence {

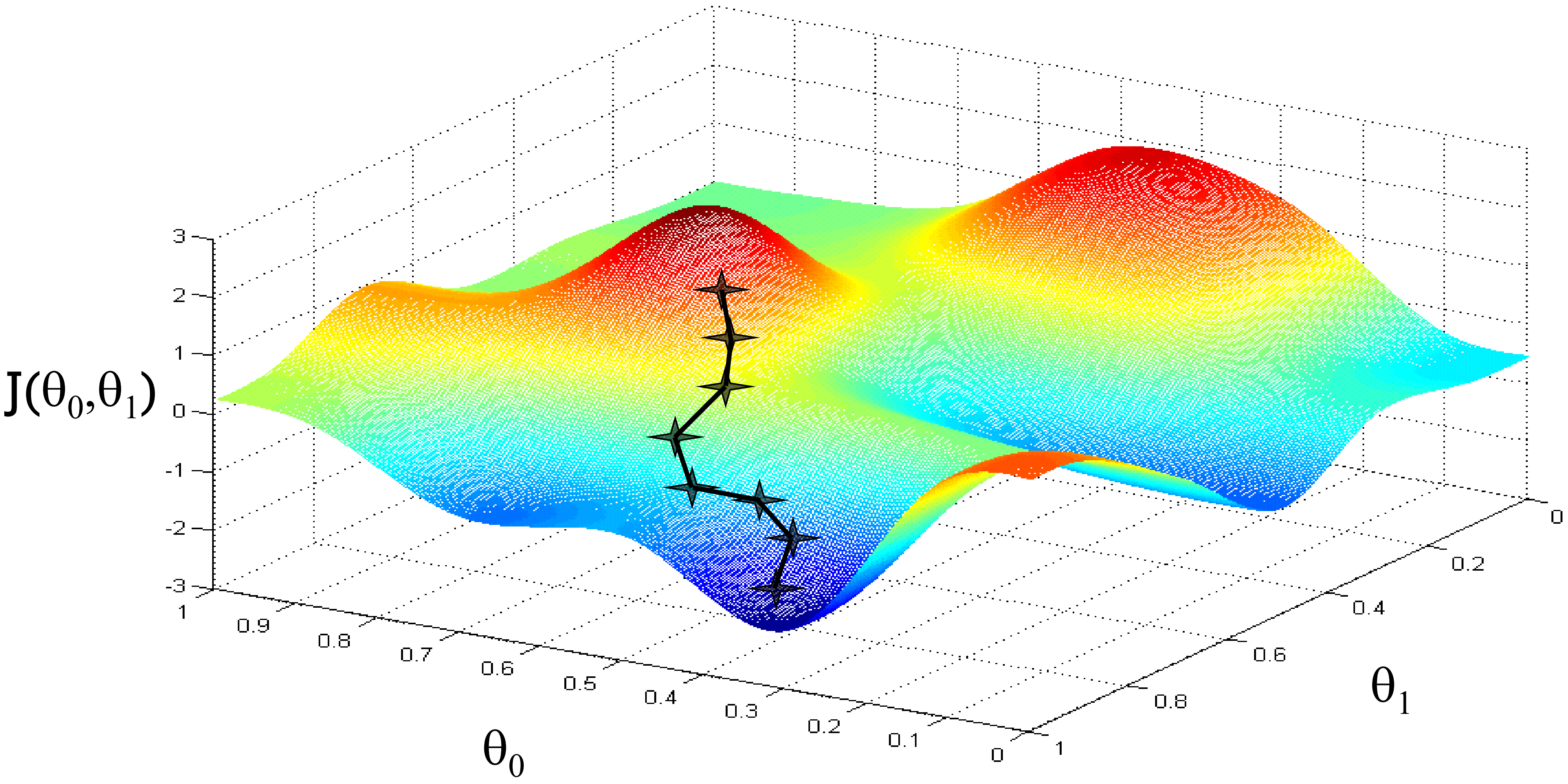
$$\theta_0 := \theta_0 - \alpha \boxed{\frac{1}{m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)}\right)}$$

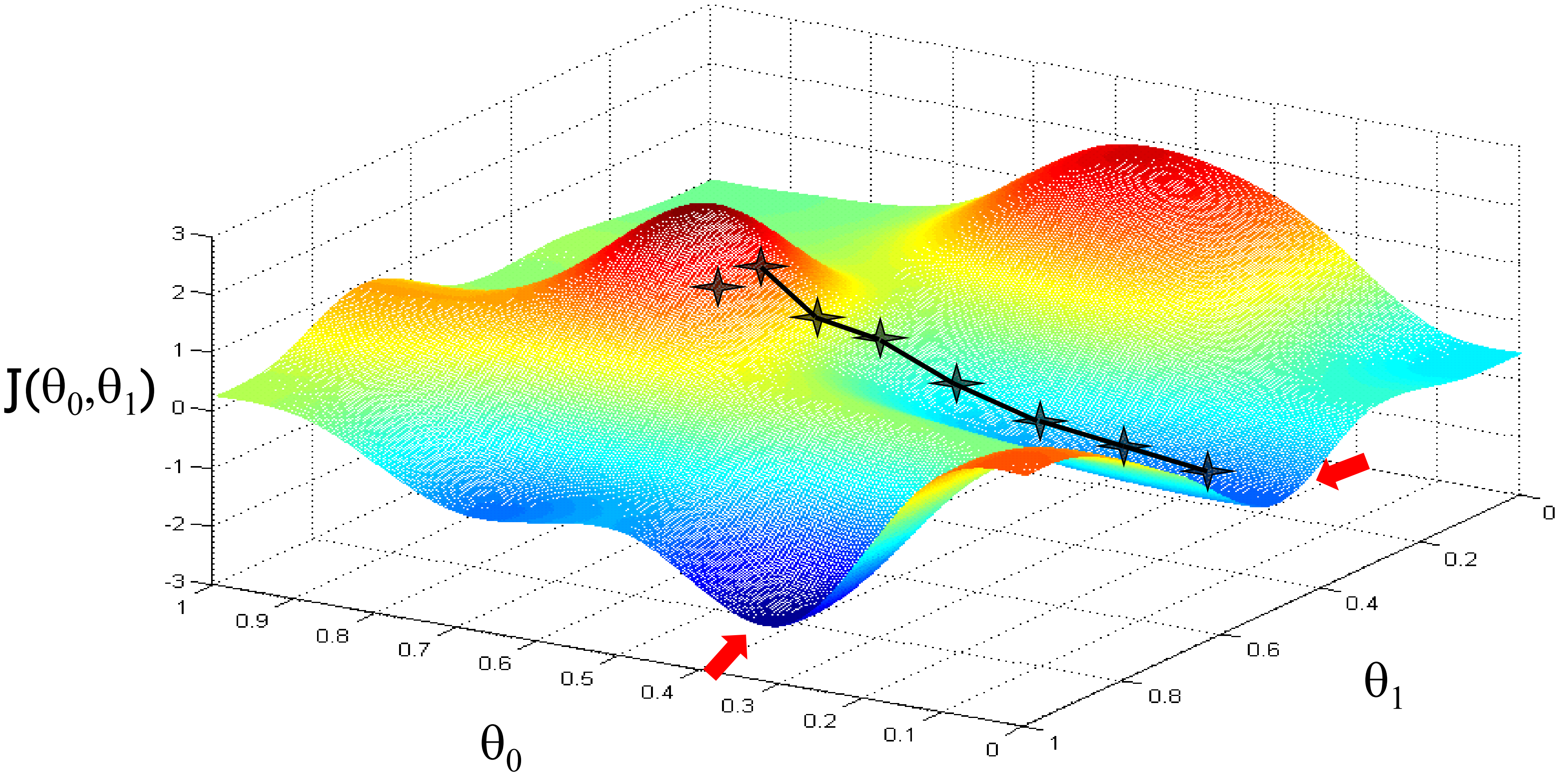
$$\theta_1 := \theta_1 - \alpha \boxed{\frac{1}{m} \sum_{i=1}^{m} \left(h_\theta(x^{(i)}) - y^{(i)}\right) \cdot x^{(i)}}$$

}

$$\frac{d}{d\theta_1} \cdot J(\theta_0, \theta_1)$$

update θ_0 and θ_1 simultaneously


J(θ0,θ1)
3
2
1
0
-1
-2
-3
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
θ0
0
0.2
0.4
0.6
0.8
1
θ1



$J(\theta_0,\theta_1)$
3
2
1
0
-1
-2
-3
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
θ_0
0
0.2
0.4
0.6
0.8
1
θ_1

STOCHASTIC VS BATCH Gradient Descent

Stochastic Gradient Descent

- Repeatedly run through the training set, and each time we encounter a training example, we update the parameters according to the gradient of the error with respect to that single training example only.
- Also called incremental gradient descent

BATCH Gradient Descent

- This method looks at every example in the entire training set on every step, to update the θ

- Modifikasi: mini batch → sekelompok data digunakan dalam 1x iterasi, misal batch_size = 16 artinya 16 buah data digunakan dalam 1x iterasi

STOCHASTIC

Loop {

for i=1 to m, {

$$\theta_j := \theta_j + \alpha \left(y^{(i)} - h_\theta(x^{(i)})\right) x_j^{(i)} \qquad \text{(for every } j\text{)}.$$

}

}

BATCH

Repeat until convergence {

$$\theta_j := \theta_j + \alpha \sum_{i=1}^{m} \left(y^{(i)} - h_\theta(x^{(i)})\right) x_j^{(i)} \qquad \text{(for every } j\text{)}.$$

}

- Whereas batch gradient descent has to scan through the entire training set before taking a single step—a costly operation if m is large—stochastic gradient descent can start making progress right away, and continues to make progress with each example it looks at.
- Often, stochastic gradient descent gets θ “close” to the minimum much faster than batch gradient descent.
- Note however that it may never “converge” to the minimum, and the parameters θ will keep oscillating around the minimum of $J(\theta)$; but in practice most of the values near the minimum will be reasonably good approximations to the true minimum.
- For these reasons, particularly when the training set is large, stochastic gradient descent is often preferred over batch gradient descent

MULTIPLE FEATURES/ Multiple variable regression

Multiple features (variables).

Size ($feet^2$) x	Price (\$1000) y
2104	460
1416	232
1534	315
852	178
…	…

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

Multiple features (variables).

Size (feet^2)	Number of bedrooms	Number of floors	Age of home (years)	Price ($1000)
2104	5	1	45	460
1416	3	2	40	232
1534	3	2	30	315
852	2	1	36	178
…	…	…	…	…

Notation:

n = number of features

$x^{(i)}$ = input (features) of i^{th} training example.

$x_j^{(i)}$ = value of feature j in i^{th} training example.

Hypothesis:

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

$$h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n$$

For convenience of notation, define $x_0 = 1$.

Multivariate linear regression.

Gradient descent for multiple variables

Hypothesis: $h_\theta(x) = \theta^T x = \theta_0 x_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n$

Parameters: $\theta_0, \theta_1, \ldots, \theta_n$

Cost function:

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

Gradient descent:

Repeat $\{$

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta_0, \ldots, \theta_n)$$

$\}$ (simultaneously update for every $j = 0, \ldots, n$)

Gradient Descent

Previously (n=1):

Repeat {

$$\theta_0 := \theta_0 - \underbrace{\alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})}_{\frac{\partial}{\partial \theta_0} J(\theta)}$$

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

(simultaneously update θ_0, θ_1)

}

New algorithm $(n \geq 1)$:

Repeat {

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

(simultaneously update θ_j for $j = 0, \ldots, n$)

}

$$\theta_0 := \theta_0 - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) x_0^{(i)}$$

$$\theta_1 := \theta_1 - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) x_1^{(i)}$$

$$\theta_2 := \theta_2 - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) x_2^{(i)}$$

$\ldots$

Now, how to program?
See the notebook given,
and do the task in TK02