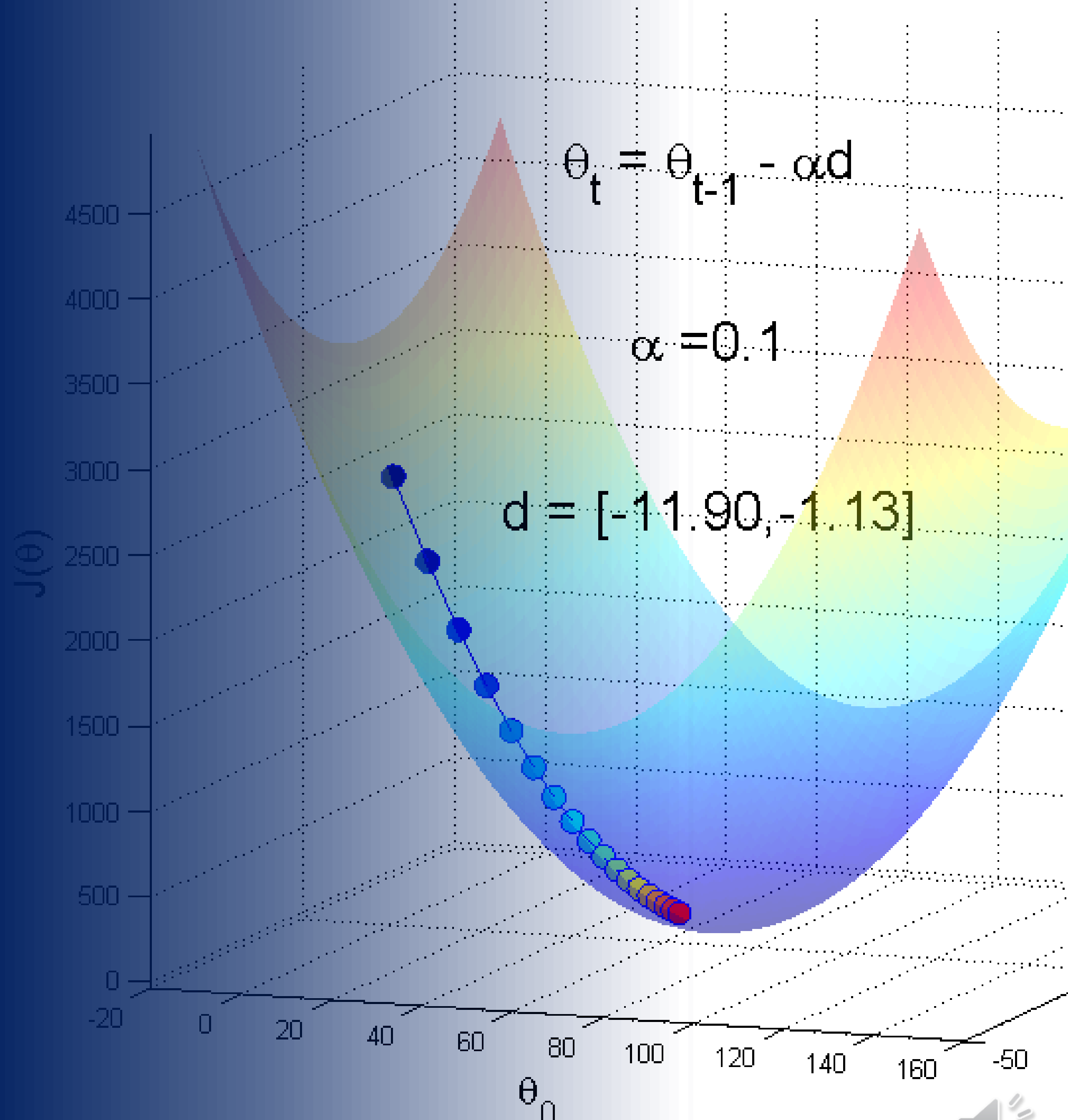
Obučavanje modela



Komponente obučavanja

Nepoznata ciljna funkcija
$f: X \to Y$

Trening skup
$T = \{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), \dots, (x^{(N)}, y^{(N)})\}$

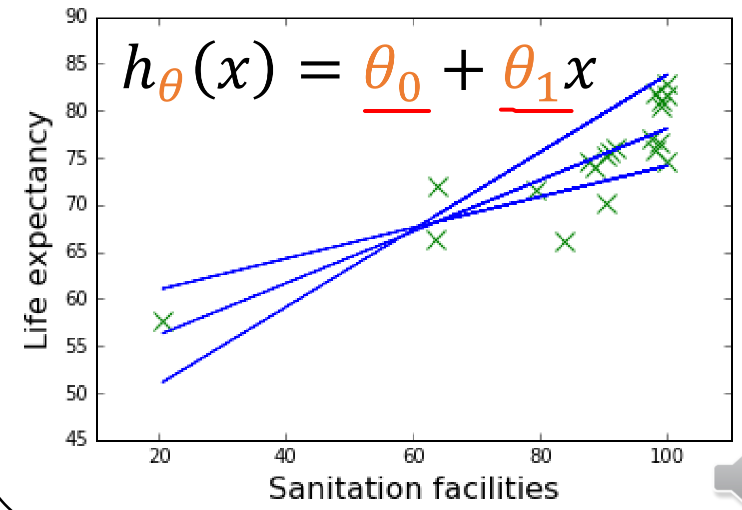
Kako odabrati "najbolju" liniju?

(onu koja se najbolje uklapa u podatke)

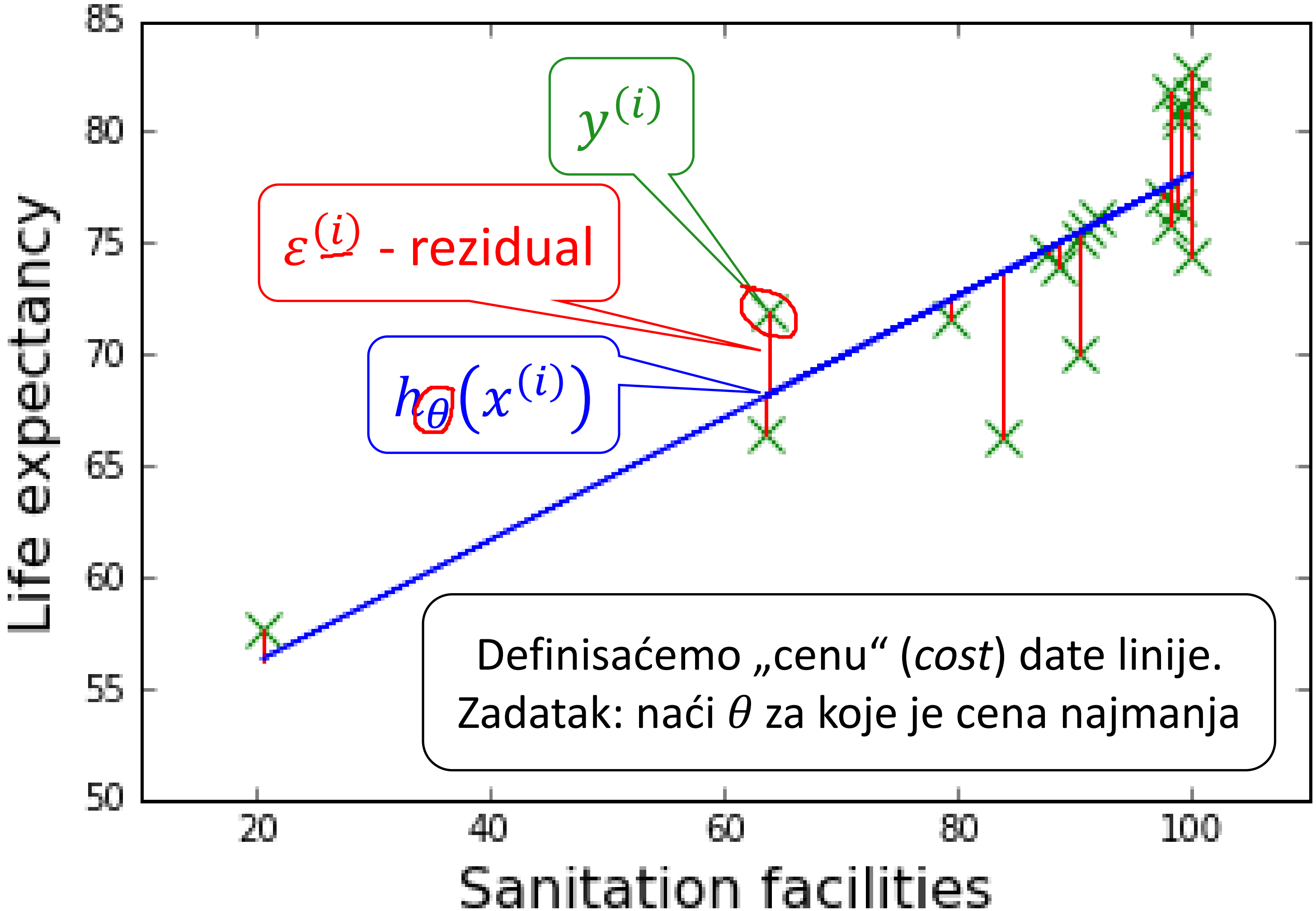
Obučavajući algoritam

Konačna hipoteza
$g \approx f$

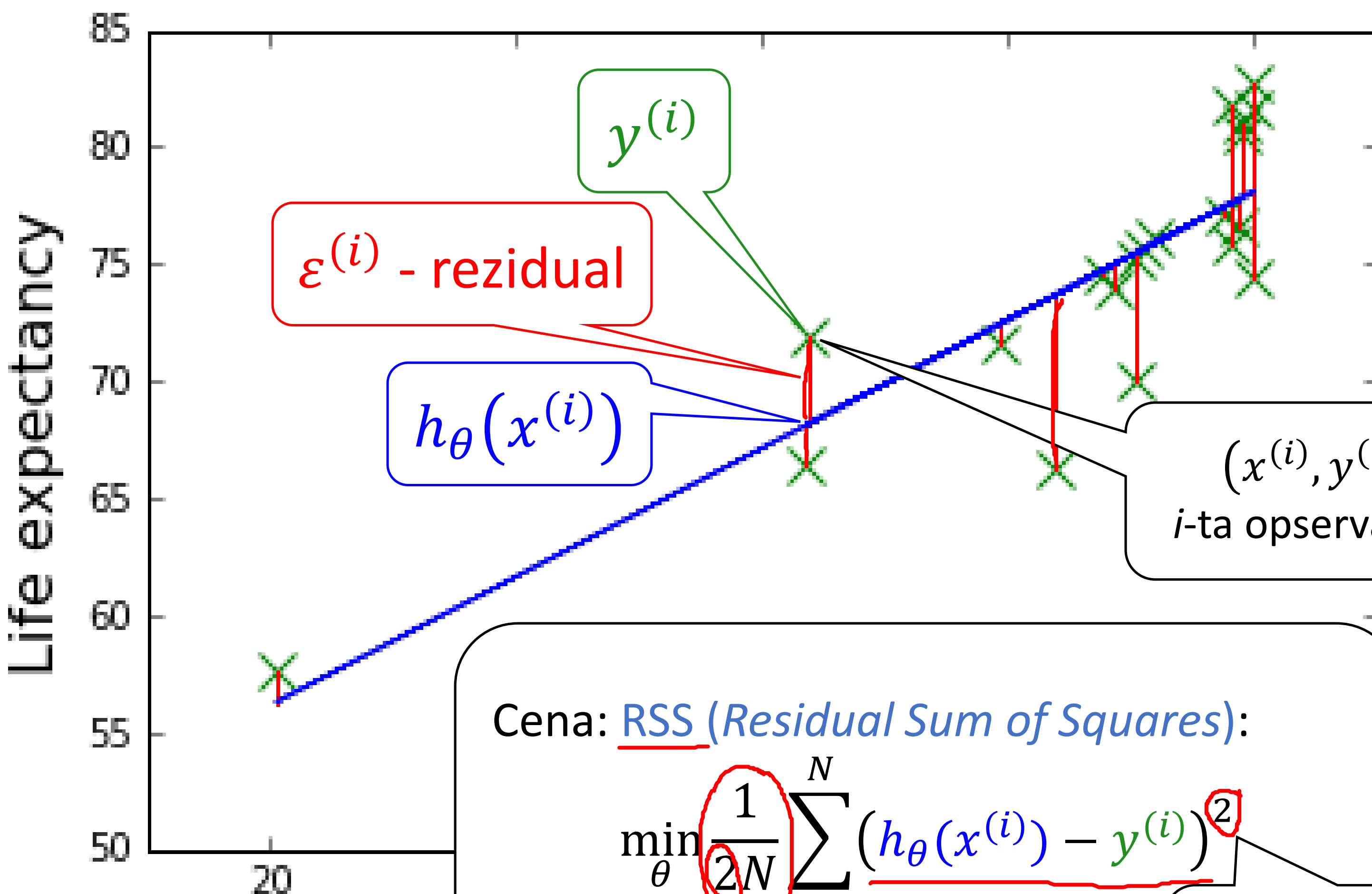
Skup hipoteza
$\mathcal{H}$



Obučavanje modela



Obučavanje modela



Cena: RSS (*Residual Sum of Squares*):

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

N – broj opservacija

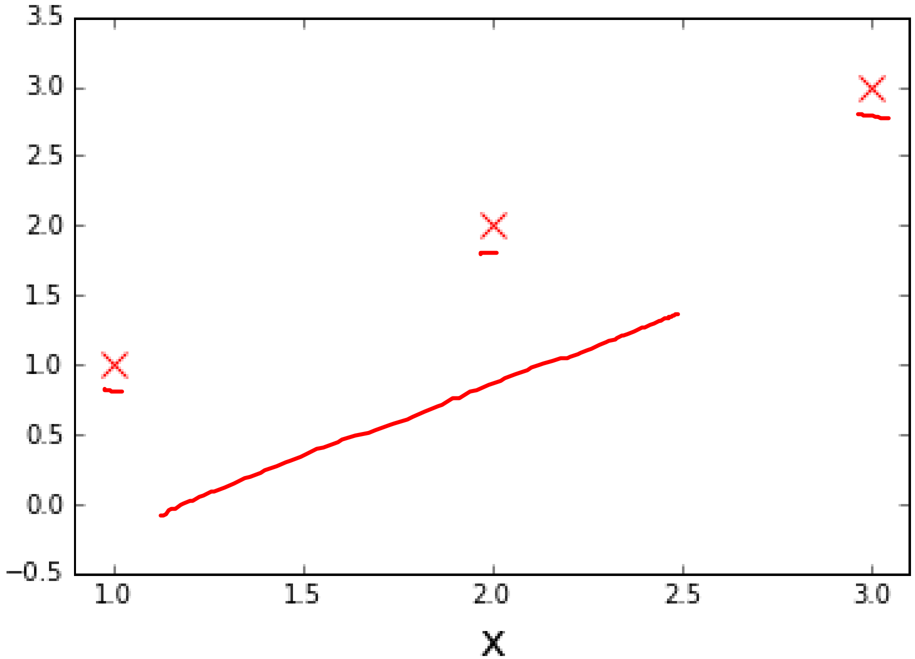
Kvadrat – jednostavnost analitičkog rešenja

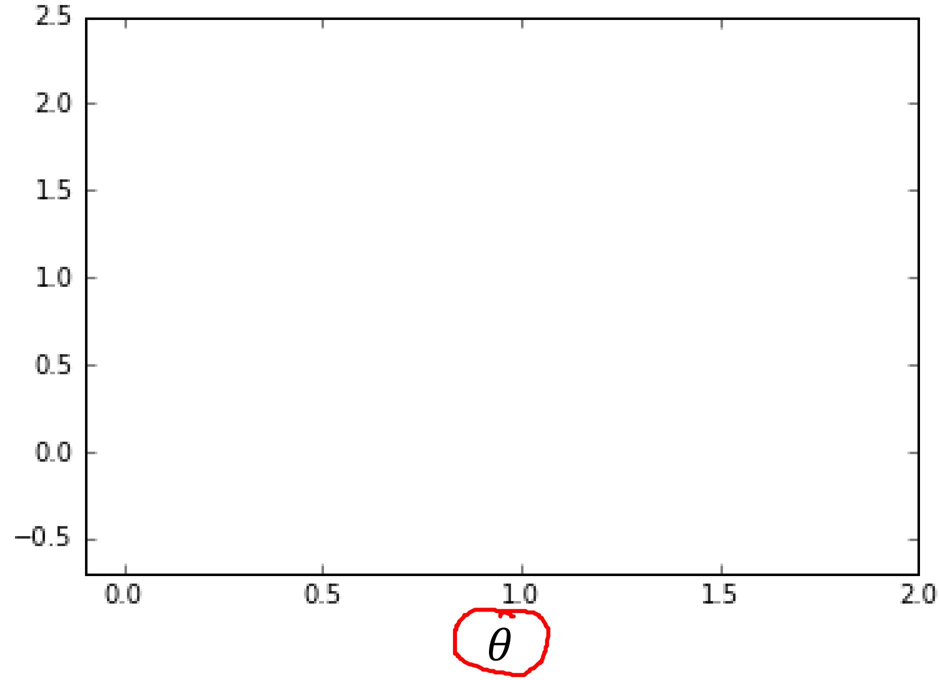
Funkcija greške – intuicija

Model: linija koja prolazi kroz $(0, 0)$

$$f_\theta(x) = \theta \cdot x$$

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





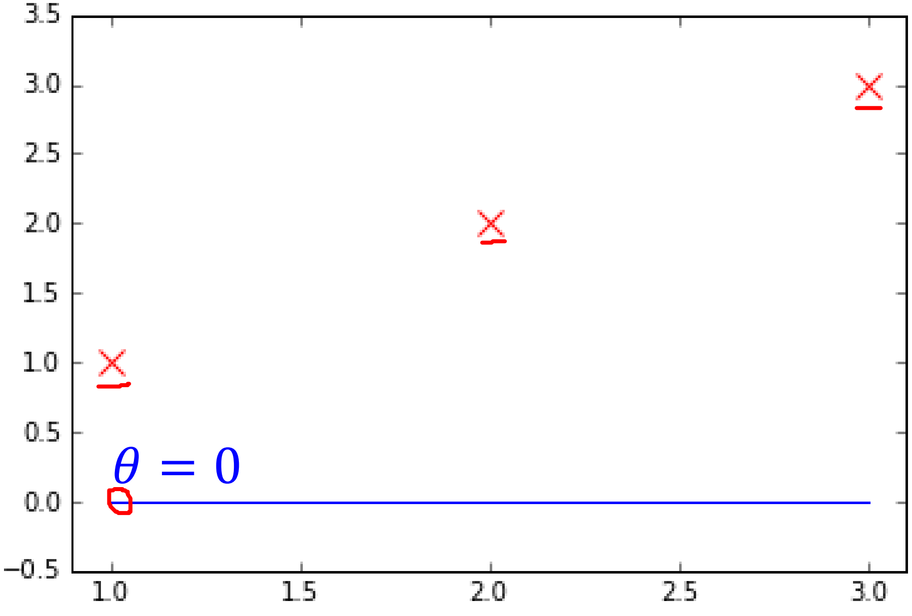
Promenom θ menjamo nagib linije

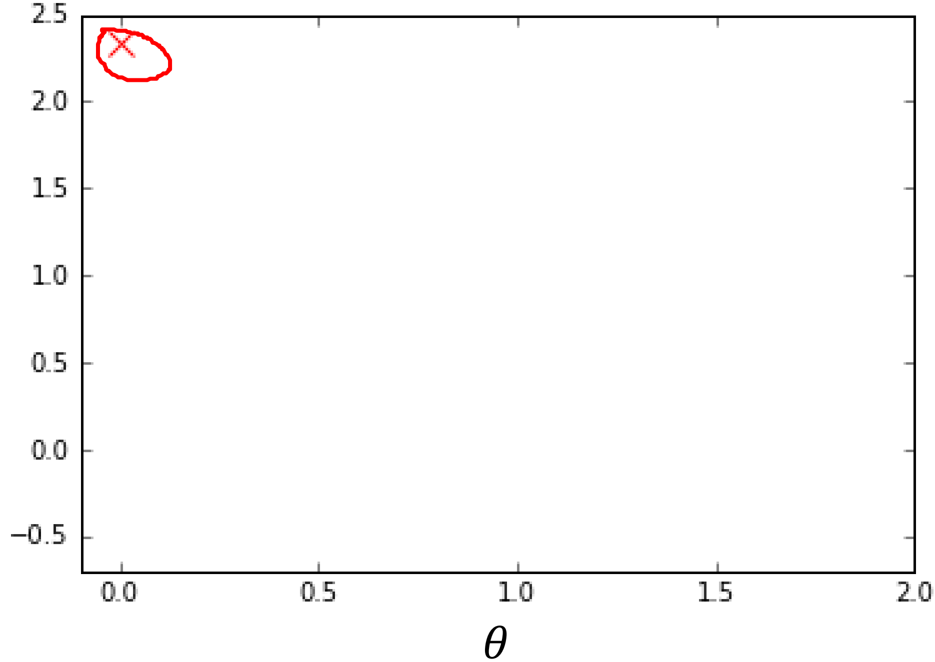
Posledica je da se menja $J(\theta)$ (greška je funkcija koja zavisi od θ)

Funkcija greške – intuicija

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

$$f_\theta(x) = \theta \cdot x$$



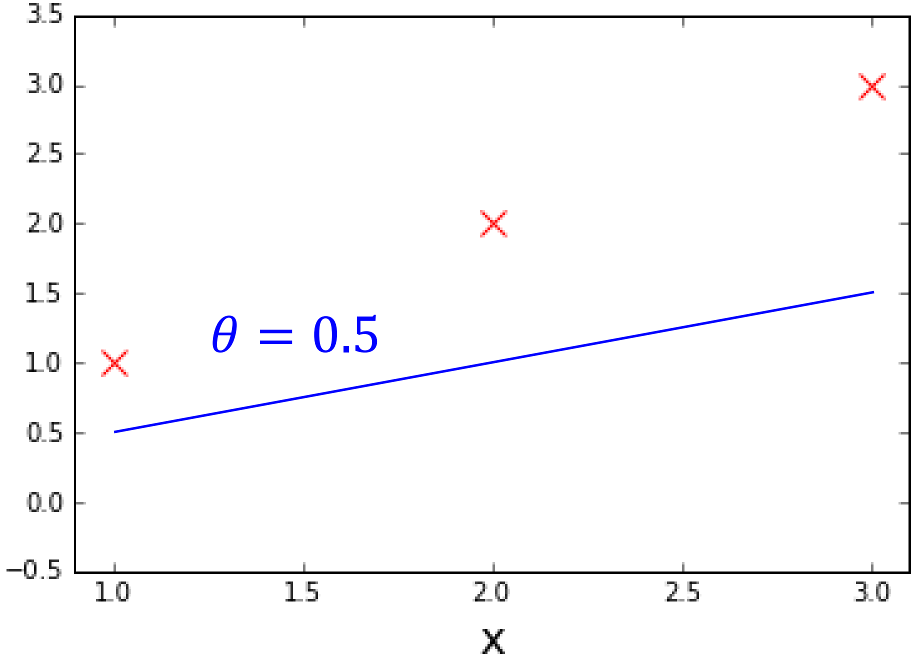


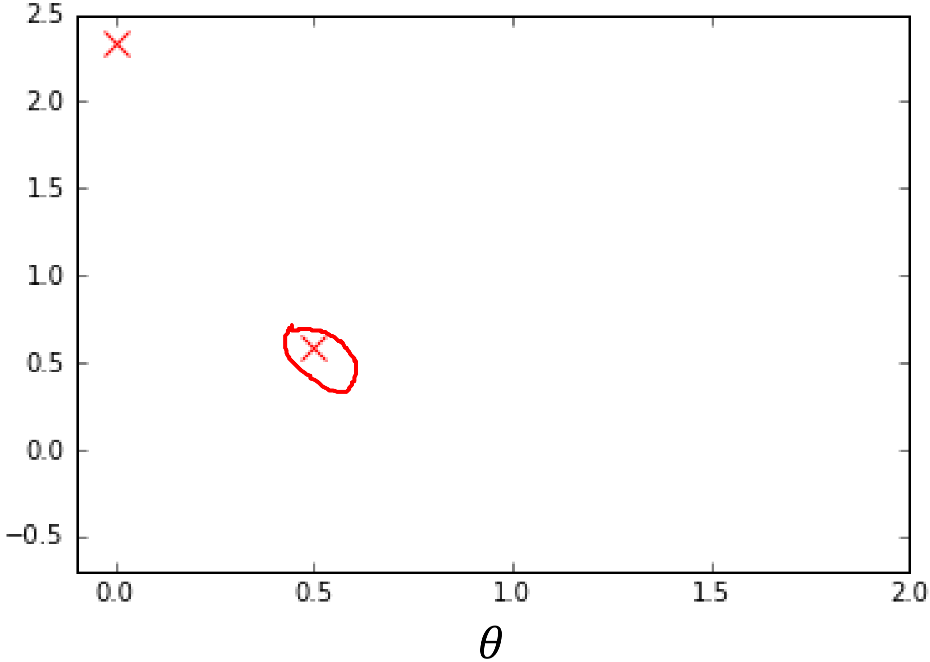
$$J(0) = 1/6\,\left((0*1-1)^2 + (0*2-2)^2 + (0*3-3)^2\right) \approx 2.33$$

Funkcija greške – intuicija

$$f_\theta(x) = \theta \cdot x$$

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



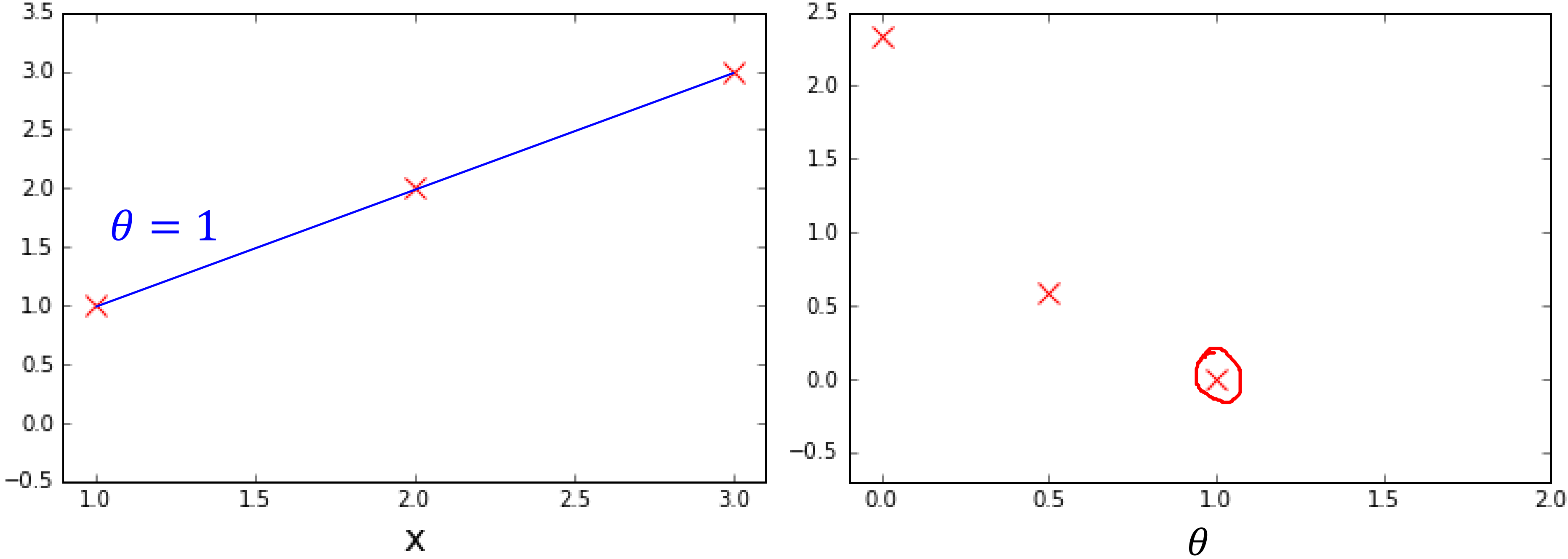


$$J(0.5) = 1/6\ \left((0.5 * 1 - 1)^2 + (0.5 * 2 - 2)^2 + (0.5 * 3 - 3)^2\right) \approx 0.58$$

Funkcija greške – intuicija

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

$$f_\theta(x) = \theta \cdot x$$

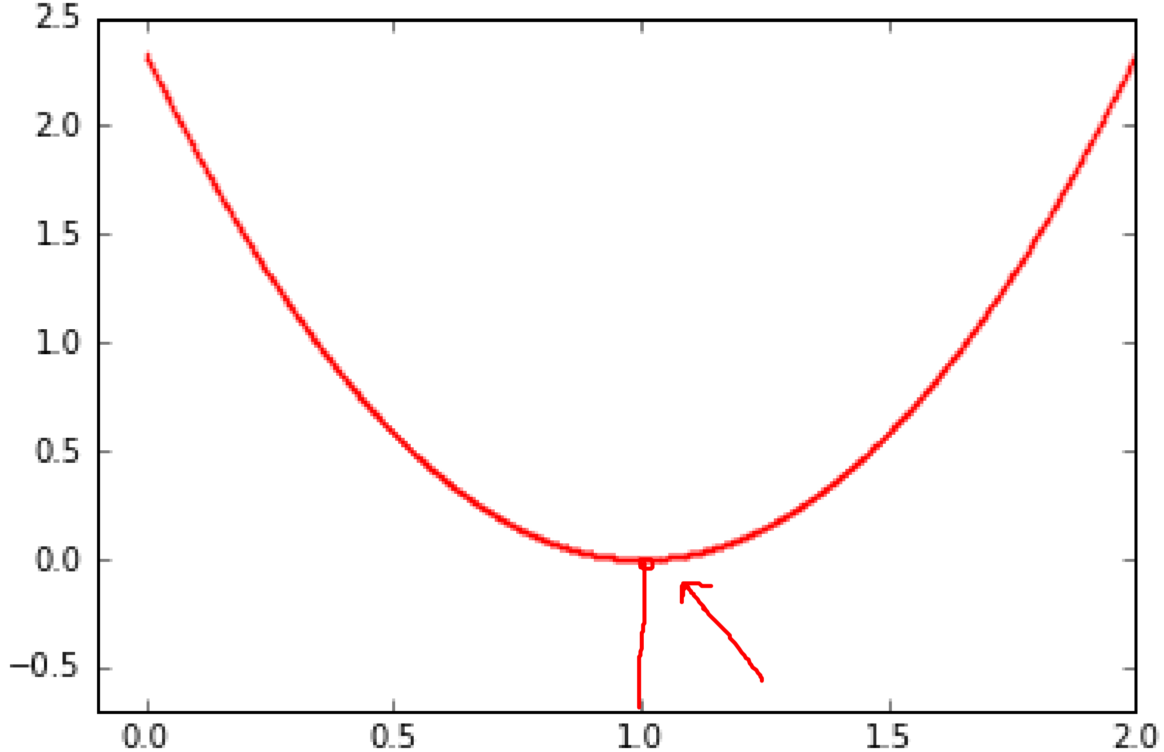


$$J(1) = 1/6\ \left((1*1-1)^2 + (1*2-2)^2 + (1*3-3)^2\right) = 0$$

Funkcija greške – intuicija

$$f_\theta(x) = \theta \cdot x$$

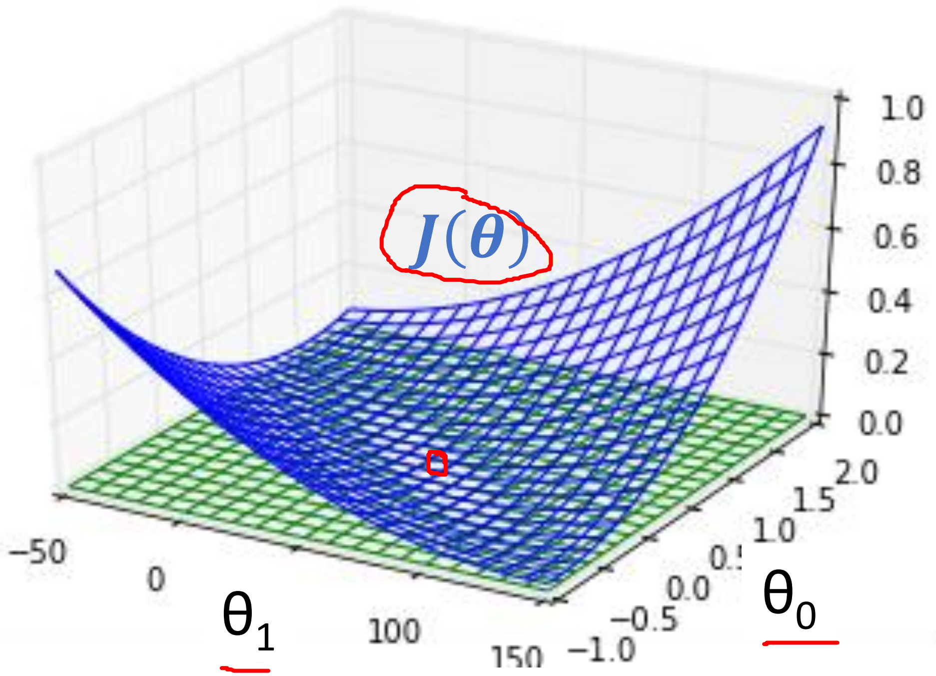
$J(\theta)$

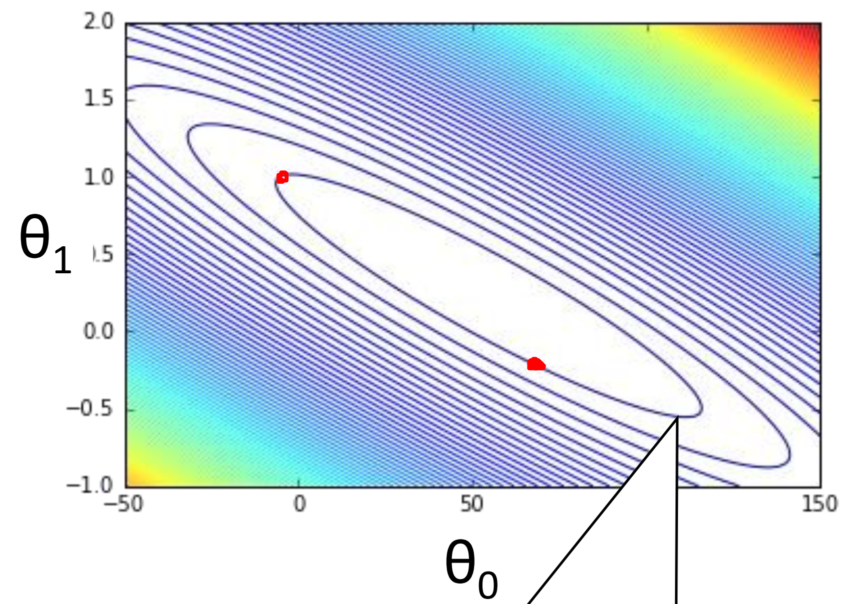


θ

Funkcija greške – intuicija

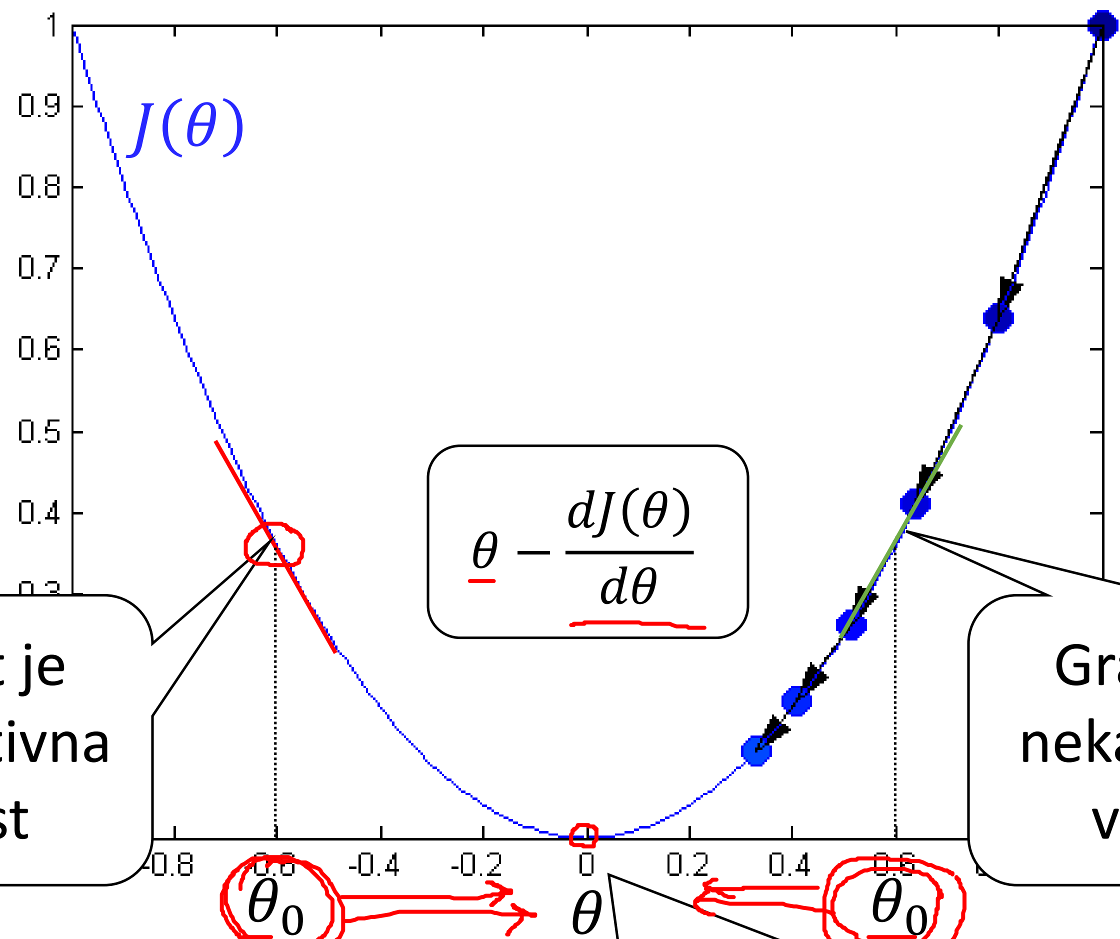
$$h_\theta(x) = \theta_1 x + \theta_0$$





Jedna kontura: sve tačke za koje je vrednost $J(\theta)$ jednaka

Kako pronaći minimum funkcije?



$$\theta - \frac{dJ(\theta)}{d\theta}$$

Gradijent je neka negativna vrednost

Gradijent je neka pozitivna vrednost

iterativno ćemo se pomerati ka minimumu

Obučavanje modela: gradijentni spust

- Optimizaciona tehnika
  - možemo pronaći ekstrem proizvoljne funkcije

Ulaz	• $J(\theta)$ – funkcija koja se optimizuje • $\theta_{\mathbf{0}}$ – početno rešenje • α – *learning rate* (veličina koraka) • *maxIters* – maksimalan broj iteracija
Postupak	for *t* = 1, 2,…, maxIters: for *d* = 1,2,…,*D* $\theta_d^{(t+1)} = \theta_d^{(t)} - \alpha \frac{\partial}{\partial \theta_d} J(\theta)$
Izlaz	θ – tačka u kojoj funkcija $J(\theta)$ ima minimum

GD_visualization_quick_convergence.m

Primena na linearnu regresiju

$$\min_{\theta} \mathrm{J}(\theta)$$

$$\theta_d^{(t+1)} = \theta_d^{(t)} - \alpha \boxed{\frac{\partial}{\partial \theta_d}} J(\theta)$$

$$J(\theta) = \frac{1}{2N} \sum_{i=1}^{N} (\underbrace{\theta_1 x^{(i)} + \theta_0}_{h_\theta(x)} - y^{(i)})^2$$

Za implementaciju bi bilo lakše da je pravilo ažuriranja jednako za sve θ_d

$$\frac{\partial J}{\partial \theta_0} = \frac{1}{N} \sum_{i=1}^{N} (\underbrace{\theta_1 x^{(i)} + \theta_0}_{h_\theta(x)} - y^{(i)}) \cdot 1 \qquad x_0^{(i)}$$

$$\frac{\partial J}{\partial \theta_1} = \frac{1}{N} \sum_{i=1}^{N} (\underbrace{\theta_1 x^{(i)} + \theta_0}_{h_\theta(x)} - y^{(i)}) \cdot x^{(i)} \qquad x_1^{(i)}$$

Primena na linearnu regresiju

$$X = \begin{bmatrix} 1 & x^{(1)} \\ 1 & x^{(2)} \\ \cdots & \cdots \\ 1 & x^{(N)} \end{bmatrix}_{N\times 2} \qquad \theta = \begin{bmatrix} \theta_0 \\ \theta_1 \end{bmatrix}_{2\times 1} \qquad h_\theta(x) = X\theta = \begin{bmatrix} \theta_0 + \theta_1 x^{(1)} \\ \theta_0 + \theta_1 x^{(2)} \\ \vdots \\ \theta_0 + \theta_1 x^{(N)} \end{bmatrix}_{N\times 1}$$

x_0 x_1

1

$$\frac{\partial J}{\partial \theta_0} = \frac{1}{N}\sum_{i=1}^{N}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)x_0^{(i)}$$

$$\frac{\partial J}{\partial \theta_1} = \frac{1}{N}\sum_{i=1}^{N}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)x_1^{(i)}$$

$$\boxed{\theta_d^{(t+1)} = \theta_d^{(t)} - \frac{\alpha}{N}\sum_{i=1}^{N}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)x_d^{(i)}}$$

LMS (*Least Mean Squares*) update rule

$$\theta_d^{(t+1)} = \theta_d^{(t)} - \frac{\alpha}{N}\sum_{i=1}^{N}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)x_d^{(i)}$$

Magnituda promene je proporcionalna grešci

Mala greška → nećemo mnogo menjati θ

Velika greška → mnogo ćemo promeniti θ

Navedite razliku

Batch gradient descent

Stochastic gradient descent

Batch Gradient Descent

for t = 1, 2,..., maxIters:

for d = 1,2,...,D

$$\theta_d^{(t+1)} = \theta_d^{(t)} - \frac{\alpha}{N}\sum_{i=1}^{N}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)x_d{}^{(i)}$$

Koristimo ceo skup podataka da ažuriramo model

Iteracija preko celog skupa podataka je skupa ako je N veliko

Stohastic Gradient Descent

for t = 1, 2,..., maxIters:

Shuffle dataset

for i = 1,2,..., N

for d = 1,2,...,D

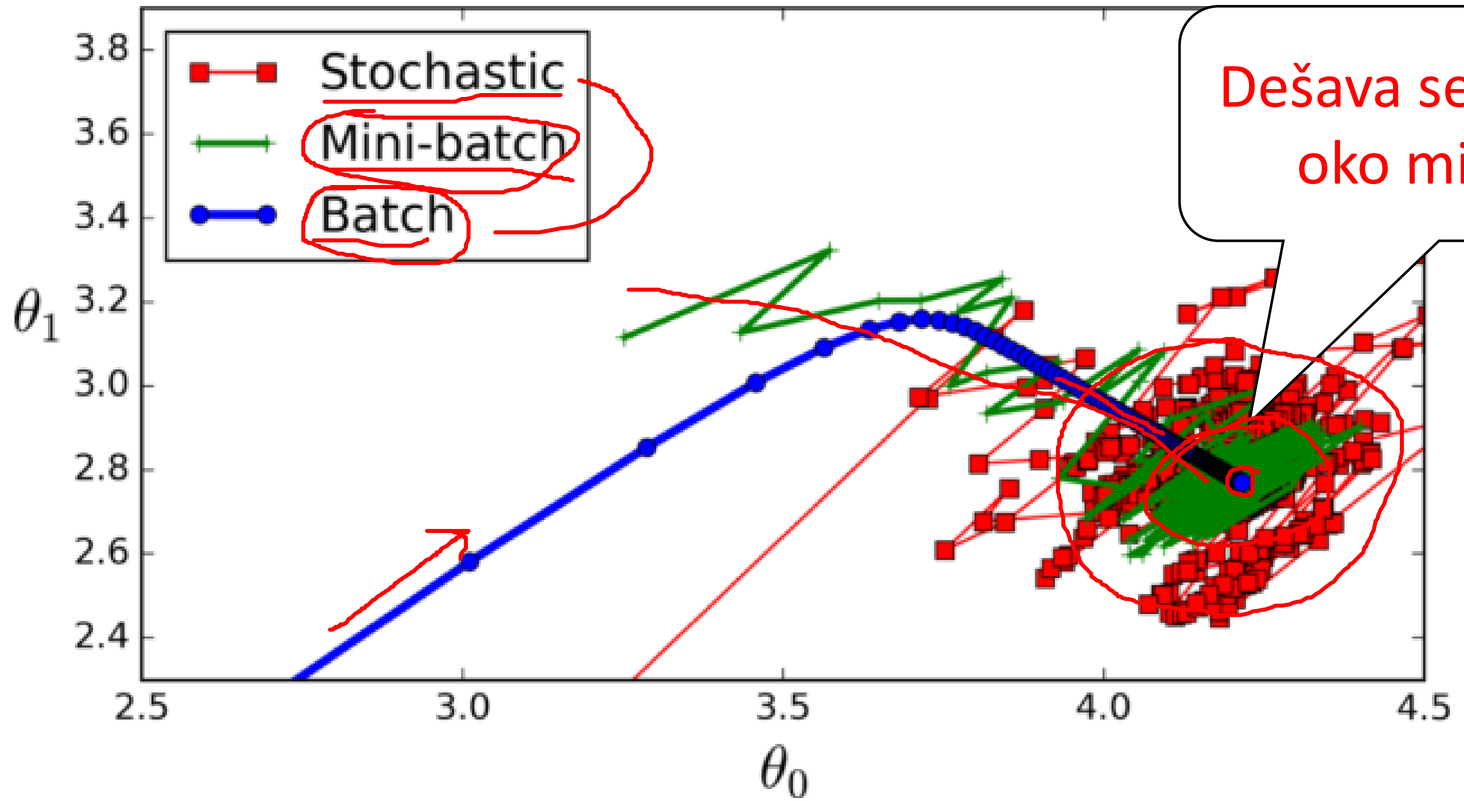
$$\theta_d^{(t+1)} = \theta_d^{(t)} - \alpha\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)x_d{}^{(i)}$$

Više puta prolazimo kroz skup podataka

Ažuriramo model na osnovu jednog trening podatka

Napredujemo sa svakim podatkom – ovo preferiramo kada je N veliko

Stohastic Gradient Descent



Dešava se da osciluje oko minimuma

- Obično će θ dovesti brže blizu minimuma od batch GD
- U praksi, blizu je često dovoljno dobro
- Zato ga preferiramo kada je N veliko

Rezime

- *Residual Sum of Squares*
- Gradijentni spust