



Lineární regrese

Metoda
postupného
klesání

Kdy metodu používáme a jak funguje?

- Máme hodnotovou funkci $J(\theta_0, \theta_1)$
- Chceme ji minimalizovat pro dané parametry θ_0, θ_1
- Začneme náhodnou inicializací parametrů θ_0, θ_1
- Měníme parametry, dokud není $J(\theta_0, \theta_1)$ minimální
- Minima ale nemusíme nikdy dosáhnout!
- Anglicky se metoda nazývá *gradient descent*

Jak na to?

$$\textit{opakovat dokud nekonverguje} \left\{ \begin{array}{l} \theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1) \end{array} \right. \}$$

α – míra učení (angl. *learning rate*) určuje velikost kroku, kterým se algoritmus blíží k minimu

$\frac{\partial}{\partial \theta_j}$ - parciální derivace hodnotové funkce přes parametry θ_0 a θ_1

Každým krokem se zmenšuje hodnota θ_j , od které se odečítá příslušná parciální derivace vážená mírou učení.

Na co si dát pozor?

- Nastavení míry učení α
 - Pokud je α malá, algoritmus může být velmi pomalý
 - Pokud je α velká, algoritmus nemusí konvergovat nebo dokonce může divergovat
- Počítat jednotlivé θ_j souběžně

$$\begin{array}{ccc} temp_0 := \theta_0 - \alpha \frac{\partial}{\partial \theta_0} J(\theta_0, \theta_1) & \longrightarrow & \theta_0 := temp_0 \\ temp_1 := \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1) & & \theta_1 := temp_1 \end{array}$$

- Nastavení počtu iterací