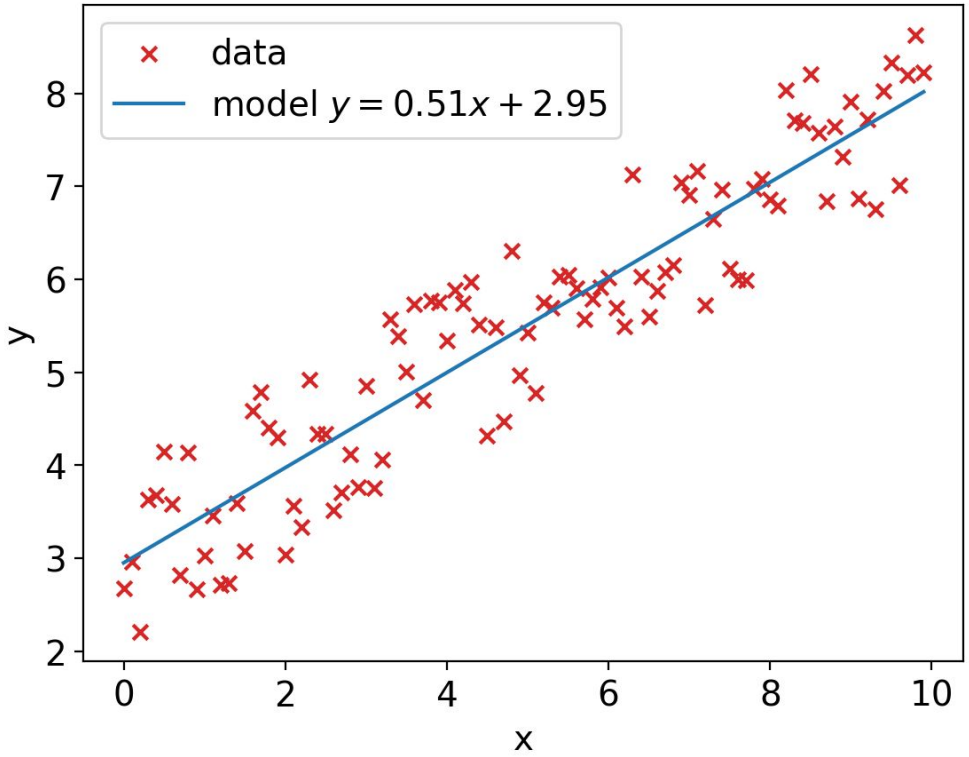
Giliojo mokymosi metodai

Modeliai, neuroniniai tinklai

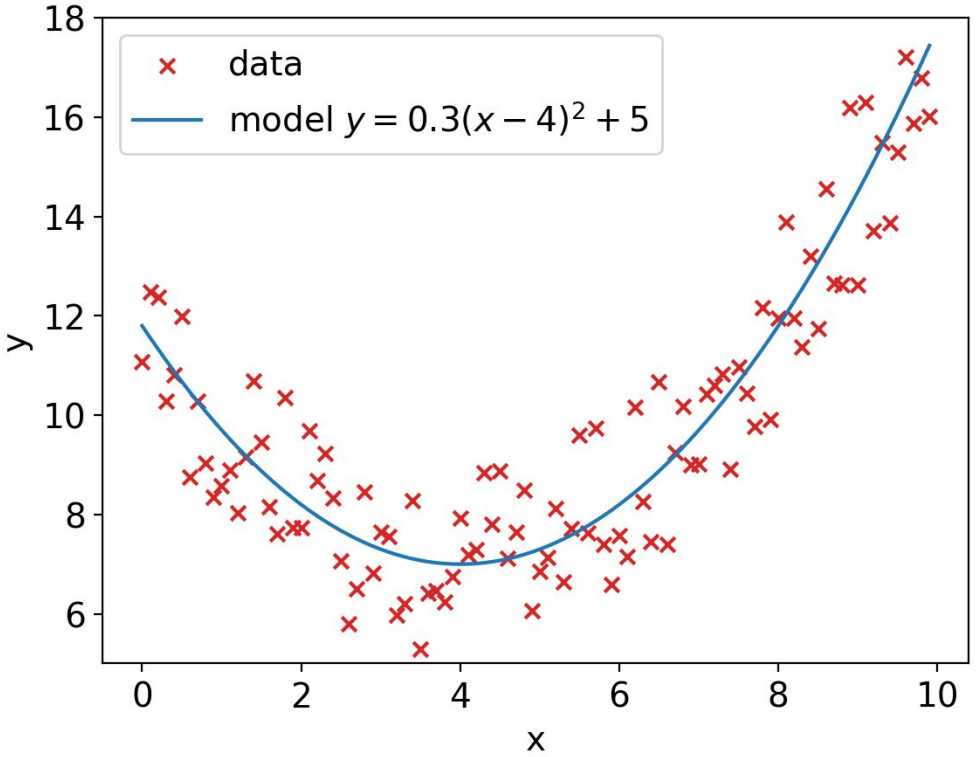
2024-02-12

Modeliai



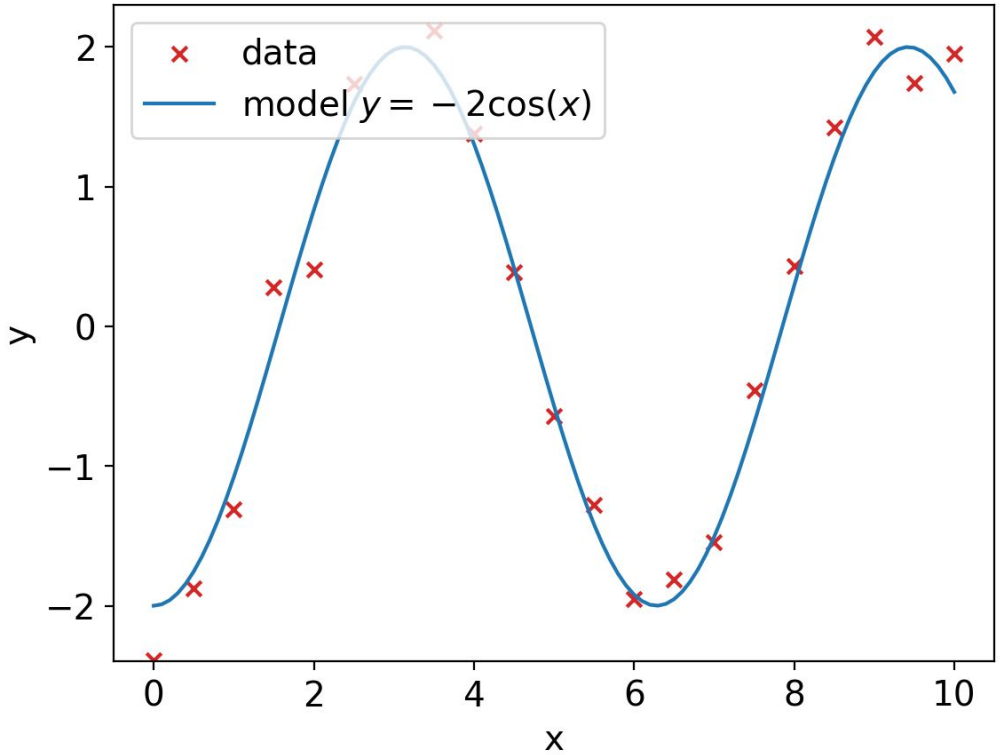
⇒ *Linear model*

Modeliai



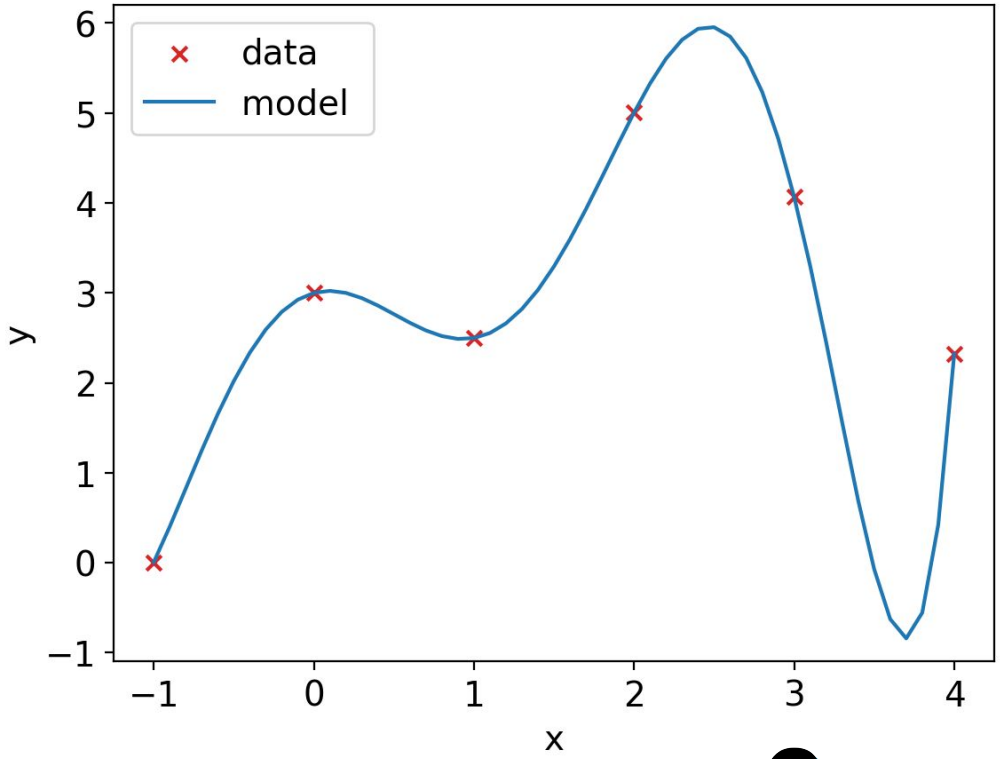
⇒ *Quadratic model*

Modeliai



⇒ *Cosine model*

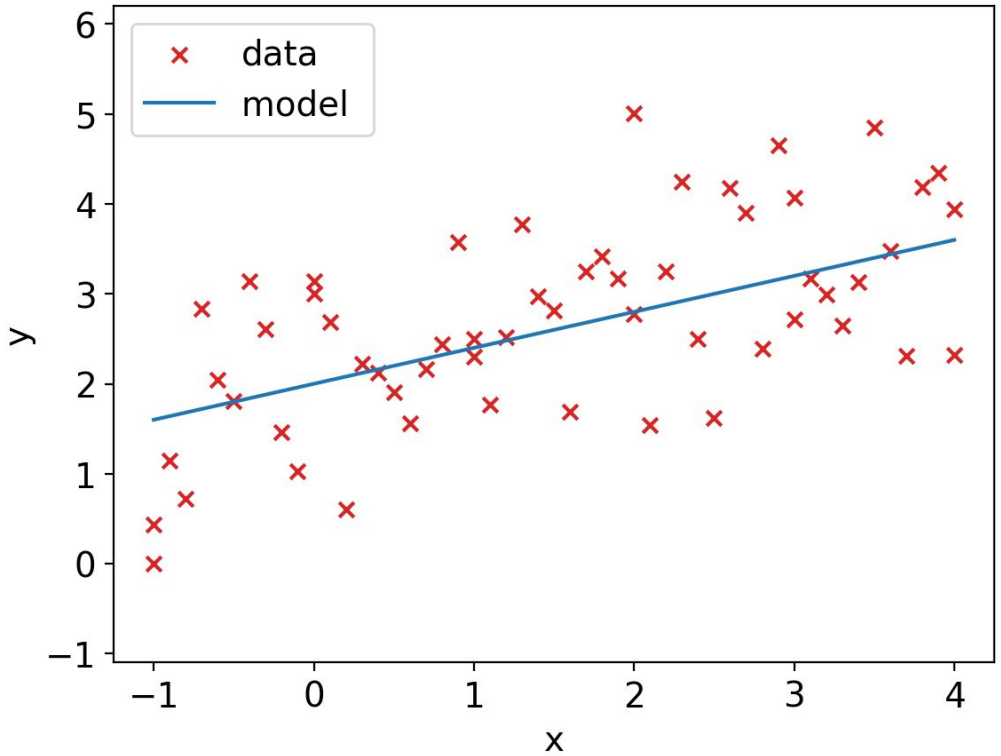
Modeliai



⇒ *Overfitting*

⇒ *Polynomial model* ?

Modeliai



⇒ *Linear model*

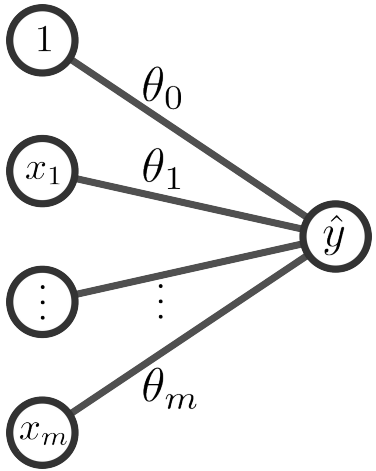
Python

linear_model.py

Neuroninis tinklas

$$\hat{y} = \boldsymbol{x}^\top \boldsymbol{\theta} = \sum_{i=0}^{m} x_i \theta_i$$

$$\boldsymbol{x} = \begin{bmatrix} 1 \\ x_1 \\ x_2 \\ \vdots \\ x_m \end{bmatrix} \quad \boldsymbol{\theta} = \begin{bmatrix} \theta_0 \\ \vdots \\ \theta_m \end{bmatrix}$$



1 Užduotis

žr. teorijos paskaitų skaidres

Loading the Data

size of dataset $\gg$ *computer memory*

$\Rightarrow$ *batching*

Stochastic Gradient Descent

$$\mathcal{L}_\theta = \frac{1}{N}\sum_i^N \ell_\theta(x_i, y_i) \qquad \text{e.g., } \ell_\theta(x_i, y_i) = \left(\hat{y}(x_i) - y_i\right)^2$$

(Full) Batch Gradient Descent $\theta_{t+1} = \theta_t - \gamma \frac{\partial \mathcal{L}}{\partial \theta_t}$

Stochastic Gradient Descent $\theta_{t+1} = \theta_t - \gamma \frac{\partial \ell}{\partial \theta_t}$

Mini-Batch Gradient Descent $\theta_{t+1} = \theta_t - \gamma \frac{\partial \frac{1}{|B|}\sum_{b \in B} \ell(x_b, y_b)}{\partial \theta_t}$

sasnauskas.net/gmm