Aufgabe 1 (ML)

Sonntag, 27. März 2022 10:54

Exercises - Introduction Machine Learning

Prof. Dr.-Ing. Steffen Schober

SoSe 22

1 Minimization of functions

Given the function

$$f(x_1, x_2) = (x_1 - 1)^2 - 2x_1x_2 + 2x_2^2 + 1$$

Solve

$$\min_{x_1, x_2} f(x_1, x_2).$$

Note that for a minima at some point, it is necessary that the partial derivative for each variable is zero there. Is this a local or global minima? Sketch the function, or plot it with Python, to see it!

2 Polynomial Regression

In the lecture we discussed polynomial regression. For given training data $(x^{(i)}, y^{(i)}), i = 1, \ldots, N$ we have to solve

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

where

$$y(x, \boldsymbol{\theta}) = \sum_{k=0}^{K} \theta_k x^k \quad \boldsymbol{\theta} = (\theta_1, \ldots, \theta_K)^T$$

Let $K = 2$. Show that the solution of the minimization problem can be found by solving a system of linear equations (you don't have to compute the solution). Hint: first compute the partial derivative of $y(x, \boldsymbol{\theta})$ for θ_k. To simplify notation write x_i for $x^{(i)}$ and y_i for $y^{(i)}$.

3 Polynomial Regression

Let us consider the polynomial regression example from the lecture where we fitted a polynomial of degree K. In the lecture we saw that noise in the target lead to large coefficients when the order of the polynomial increases. Let's check this your self! The code below shows how to generate and plot the data.

```
%matplotlib inline
import matplotlib.pyplot as plt
import numpy as np

N = 10
s = 0.5 # strength of noise
x = np.linspace(0, 1, N)
e = np.random.normal(0, s, N)
```



```

y = np.sin(2*np.pi*x) + e
plt.scatter(x,y)

# for plotting
xx = np.linspace(0,1, 100)
plt.plot(xx, np.sin(2*np.pi*xx))
```

To fit a polynomial by minimizing the squared error, use the function `np.polyfit` (read the documentation!), which returns the coefficients (weights) of the fitted polynomial. For plotting the function found use `np.polyval` to compute the values of a polynomial given the coefficients. Check the coefficients obtained with and without noise (set $s = 0$). Then iterate over K and save the L_2 norm of the weights, e.g.,

$$\|\mathbf{w}\|_2 = \sum_{i=1} w_i^2$$

---

(1)

$$f(x_1, x_2) = (x_1 - 1)^2 - 2x_1x_2 + 2x_2^2 + 1$$

$$f'_{x_1} = 2(x_1 - 1) - 2x_2 = 2x_1 - 2x_2 - 2$$

$$f'_{x_2} = -2x_1 + 4x_2$$

$$\begin{matrix} f'_{x_1} = 0 \\ f'_{x_2} = 0 \end{matrix} \rightarrow \begin{array}{cc|c} x_1 & x_2 & \\ \end{array}$$

$$\left(\begin{array}{cc|c} 2 & -2 & 2 \\ -2 & 4 & 0 \end{array}\right)$$

$$\downarrow$$

$$\left(\begin{array}{cc|c} 2 & -2 & 2 \\ 0 & 2 & 2 \end{array}\right) \rightarrow x_2 = 1$$

$$\rightarrow x_1 = 2$$

$$\min_{x_1, x_2} f(x_1, x_2) = f(2, 1)$$

$$= -1 \rightarrow \text{Global Minima} \rightarrow$$

Eigenwerte der Hesse-Matrix

- ↳ positiv definite → minima
- ↳ negativ definite → maxima
- ↳ indefinite → sattelpunkt

Gradiend:

$$f(x_1, x_2, x_3, \ldots, x_n)$$

$$\frac{\partial f}{\partial x_1}(\underline{x}) = 0$$

$$\frac{\partial f}{\partial x_2}(\underline{x}) = 0 \rightarrow \nabla = \begin{pmatrix} \frac{\partial}{\partial x_1} \\ \frac{\partial}{\partial x_2} \\ \vdots \\ \frac{\partial}{\partial x_n} \end{pmatrix} \quad \nabla f = \underline{0} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

$$\frac{\partial f}{\partial x_3}(\underline{x}) = 0$$

$$\vdots$$

$$\frac{\partial f}{\partial x_n}(\underline{x}) = 0$$

(2)

$$y(x, \theta) = \theta_1 + \theta_2 x + \theta_3 x^2 \quad (K = 2)$$

$$y'_{\theta_1} = 1$$

$$y'_{\theta_2} = x \rightarrow \text{nicht Abhängig von } \theta_x \rightarrow \text{linear (konst.)}$$

$$y'_{\theta_3} = x^2$$

$$\min_{\theta} L(\theta) = \min_{\theta} \frac{1}{2} \sum_{i=1}^{N} \left(y(x_i, \theta) - y_i \right)^2$$

$$\Rightarrow \min_{\theta} \frac{1}{2} \sum_{i=1}^{N} \left(\theta_1 + \theta_2 x_i + \theta_3 x_i^2 - y_i \right)^2$$

$$L'(\theta) = \underbrace{\sum_{i=1}^{N} \left(y(x_i, \theta) - y_i \right)}_{\text{linear von } \theta} \cdot \underbrace{\frac{y'(x_i, \theta)}{\partial \theta}}_{\text{konst.}}$$

$$L'(\theta) = 0 \rightarrow \text{for } i = 1, \ldots, N$$

$$\rightarrow \text{LGS dimension } N \text{ von } \theta$$

$$\left(\begin{array}{l} L_1(\theta) = \frac{1}{2} \left(\theta_1 + \theta_2 x_1 + \theta_3 x_1^2 - y_1 \right)^2 \\ L'_{1\,\theta_1}(\theta) = \left(\theta_1 + \theta_2 x_1 + \theta_3 x_1^2 - y_1 \right) \cdot 1 \\ L'_{1\,\theta_2}(\theta) \quad \left(\theta_1 + \theta_2 x_1 + \theta_3 x_1^2 - y_1 \right) \cdot x_1 \\ L_{1\,\theta_3}(\theta) \quad \left(\theta_1 + \theta_2 x_1 + \theta_3 x_1^2 - y_1 \right) \cdot x_1^2 \end{array} \right)$$

$$\frac{\partial}{\partial \theta_0} L(\underline{\theta}) = 0$$

$$\frac{\partial}{\partial \theta_1} L(\underline{\theta}) = 0$$

$$\vdots$$

$$\frac{\partial}{\partial \theta_n} L(\underline{\theta}) = 0$$

Muste

Allgem

Allgemein:

$$\frac{\partial}{\partial \theta_k} L(\underline{\theta}) = \frac{\partial}{\partial \theta_k} \cdot \frac{1}{2} \sum_{i=1}^{N} \left(y(x_i, \underline{\theta}) - y_i \right)^2$$

$$= \sum_{i=1}^{N} \left(y(x_i, \underline{\theta}) - y_i \right) \cdot \frac{\partial y(x_i, \underline{\theta})}{\partial \theta_k}$$

$$= \sum_{i=1}^{N} \left(y(x_i, \underline{\theta}) - y_i \right) \cdot x_i^k \longrightarrow$$

$$(k = 2) \rightarrow \frac{\partial}{\partial \theta_k} L(\theta) = \sum_{i=1}^{N} \left(\theta_0 + \right.$$

Aufgabe 3 ist im JupyterNotebook bearbeitet

$\left(\text{mit } \frac{\partial}{\partial \Theta_k} y(X, \underline{\Theta}) = x^k\right)$

Linear abhäng von $\underline{\Theta}$

$\Theta_1 x_i + \Theta_2 x_i^2 - y_i) \cdot x_i^k$