Prof. Dr. Volker Roth
volker.roth@unibas.ch

Vitali Nesterov
vitali.nesterov@unibas.ch

Department of Mathematics and Computer Science
Spiegelgasse 1
4051 Basel

Exercise 4

Due date: **Wednesday, April 1st 2020**

4.1: Maximum likelihood estimate

In the regression setting, the relationship between observations and model parameters is captured. Thereby, model parameters are optimized, such that observed data is best explained by a *trained* model. Assume d-dimensional observations $(x_1, x_2, \ldots, x_d) \in \mathbb{R}^d$ and adopt the following notation for a single data point $\boldsymbol{x} = (1, x_1, x_2, \ldots, x_d)^\top \in \mathbb{R}^{d+1}$ and adjustable model parameters $\boldsymbol{\beta} = (\beta_0, \beta_1, \beta_2, \ldots, \beta_d)^\top \in \mathbb{R}^{d+1}$, where β_0 is considered as a bias. Labels for each data point $\boldsymbol{x_i}$ are denoted with $y_i \in \mathbb{R}$.

Definitions

- The **linear regression model** is defined as follows:

$$y_i \triangleq \boldsymbol{\beta}^\top \boldsymbol{x}_i + \eta_i, \quad \eta_i \sim \mathcal{N}\left(0, \sigma^2\right),$$

  such that η_i models the observation noise.

- The **probability density function** of the normal distribution $\mathcal{N}\left(\mu, \sigma^2\right)$ is

$$f\left(y|\mu, \sigma^2\right) \triangleq \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(y-\mu)^2}{2\sigma^2}\right).$$

- The **likelihood function** is given by

$$\begin{aligned}
L\left(\boldsymbol{y}|X, \boldsymbol{\beta}, \sigma^2\right) &= \prod_{i=1}^{n} f\left(y_i|\boldsymbol{\beta}^\top \boldsymbol{x}_i, \sigma^2\right) \\
&\triangleq \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(y_i - \boldsymbol{\beta}^\top \boldsymbol{x}_i)^2}{2\sigma^2}\right) \\
&= \left(\frac{1}{2\pi\sigma^2}\right)^{n/2} \exp\left(-\frac{\sum_{i=1}^{n}\left(y_i - \boldsymbol{\beta}^\top \boldsymbol{x}_i\right)^2}{2\sigma^2}\right) \\
&= \left(\frac{1}{2\pi\sigma^2}\right)^{n/2} \exp\left(-\frac{(\boldsymbol{y} - X\boldsymbol{\beta})^\top (\boldsymbol{y} - X\boldsymbol{\beta})}{2\sigma^2}\right).
\end{aligned}$$

- A **maximum likelihood estimator** is a solution to the maximization problem:

$$\widehat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} L(\boldsymbol{y}|X, \boldsymbol{\theta}),$$

with respect to model parameters $\boldsymbol{\theta} = (\boldsymbol{\beta}, \sigma^2)$.

Exercise

Maximize the likelihood function with respect to model parameters $\boldsymbol{\theta}$ and show that the estimator for $\boldsymbol{\beta}$ is

$$\widehat{\boldsymbol{\beta}} = \left(X^\top X\right)^{-1} X^\top \boldsymbol{y}$$

and the variance

$$\widehat{\sigma}^2 = \frac{1}{n} \left(\boldsymbol{y} - X\boldsymbol{\beta}\right)^\top \left(\boldsymbol{y} - X\boldsymbol{\beta}\right).$$

Why is it a good idea to assume that the noise η_i is gaussian distributed? Why is the mean parameter of the noise set to zero? In order to maximize parameters, the likelihood function is normally mapped to the log-space. What is the advantage and why is it a valid operation in terms of the parameter maximization?

4.2: Python implementation

In the above derived results, the maximum likelihood estimates for the model parameters are given. Write a Python program for polynomial regression with $x \in \mathbb{R}$ and a polynomial degree of $d - 1$.

- A polynomial basis expansion of a polynomial degree $d - 1$ for a data point $x \in \mathbb{R}$ is given with

$$g(x; \boldsymbol{\beta}) = \beta_0 + \beta_1 x^1 + \cdots + \beta_d x^d.$$

  Compute the expansion in the polynomial basis for each data point x_i and store the result in a matrix.

- Use a numpy package in order to compute the inverse.
- Evaluate the regression function and plot the corresponding curve and observed data points.
- Try out different values for the polynomial degree $d - 1$. Which conclusions can you draw?