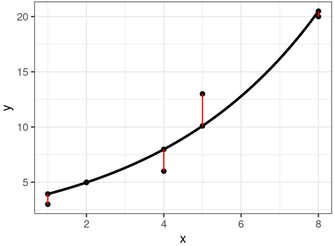
Optimization in Machine Learning

Second order methods: Gauss-Newton



Learning goals

- Least squares
- Gauss-Newton
- Levenberg-Marquardt

LEAST SQUARES PROBLEM

Consider the problem of minimizing a sum of squares

$$\min_{\boldsymbol{\theta}} \quad g(\boldsymbol{\theta})$$

$$\text{with} \quad g(\boldsymbol{\theta}) = \|r(\boldsymbol{\theta})\|_2^2 = \sum_{i=1}^{n} [r_i(\boldsymbol{\theta})]^2 = r(\boldsymbol{\theta})^\top r(\boldsymbol{\theta}).$$

r: map $\boldsymbol{\theta}$ to residuals

$$\begin{aligned} r : \mathbb{R}^d &\to \mathbb{R}^n, \\ \boldsymbol{\theta} &\mapsto r(\boldsymbol{\theta}) = \begin{pmatrix} r_1(\boldsymbol{\theta}) \\ ... \\ r_n(\boldsymbol{\theta}) \end{pmatrix} \end{aligned}$$

LEAST SQUARES PROBLEM

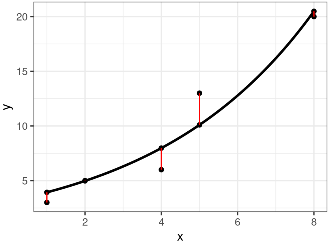
Risk minimization with squared loss $L\left(y, f(\mathbf{x})\right) = \left(y - f(\mathbf{x})\right)^2$

$$\mathcal{R}_{\text{emp}}(\boldsymbol{\theta}) = \sum_{i=1}^{n} L\left(y^{(i)}, f\left(\mathbf{x}^{(i)} \mid \boldsymbol{\theta}\right)\right) = \sum_{i=1}^{n} \underbrace{\left(y^{(i)} - f\left(\mathbf{x}^{(i)} \mid \boldsymbol{\theta}\right)\right)^2}_{[r_i(\boldsymbol{\theta})]^2}$$

also known as least squares regression is a least squares problem. $f\left(\mathbf{x}^{(i)} \mid \boldsymbol{\theta}\right)$ might be a nonlinear function. The r_i are commonly referred to as residuals.

Example:

$$\begin{aligned} \mathcal{D} &= \left(\left(\mathbf{x}^{(i)}, y^{(i)}\right)\right)_{i=1,\dots,5} \\ &= ((1,3),(2,7),(4,12),(5,13),(7,20)) \end{aligned}$$



LEAST SQUARES PROBLEM

Suppose we suspect an exponential relationship between x and y

$$f(\mathbf{x} \mid \boldsymbol{\theta}) = \theta_1 \cdot \exp(\theta_2 \cdot x), \quad \theta_1, \theta_2 \in \mathbb{R}.$$

Residuals:

$$r(\boldsymbol{\theta}) = \begin{pmatrix} \theta_1 exp(\theta_2 x^{(1)}) - y^{(1)} \\ \theta_1 exp(\theta_2 x^{(2)}) - y^{(2)} \\ \theta_1 exp(\theta_2 x^{(3)}) - y^{(3)} \\ \theta_1 exp(\theta_2 x^{(4)}) - y^{(4)} \\ \theta_1 exp(\theta_2 x^{(5)}) - y^{(5)} \end{pmatrix} = \begin{pmatrix} \theta_1 exp(1\theta_2) - 3 \\ \theta_1 exp(2\theta_2) - 7 \\ \theta_1 exp(4\theta_2) - 12 \\ \theta_1 exp(5\theta_2) - 13 \\ \theta_1 exp(7\theta_2) - 20 \end{pmatrix}.$$

LS problem:

$$g(\boldsymbol{\theta}) = r(\boldsymbol{\theta})^\top r(\boldsymbol{\theta}) = \sum_{i=1}^{5} \left(y^{(i)} - \theta_1 \exp\left(\theta_2 x^{(i)}\right)\right)^2.$$

NEWTON-RAPHSON IDEA

Approach: Calculate NR update direction by solving:

$$\nabla^2 g(\boldsymbol{\theta}^{[t]})\boldsymbol{d}^{[t]} = -\nabla g(\boldsymbol{\theta}^{[t]}).$$

The gradient is calculated by applying the chain rule

$$\nabla_\theta g(\boldsymbol{\theta}) = \nabla_\theta \left[r(\boldsymbol{\theta})^\top r(\boldsymbol{\theta}) \right] = 2 \cdot \nabla r(\boldsymbol{\theta})^\top r(\boldsymbol{\theta})$$

with $\nabla r(\boldsymbol{\theta})$ the Jacobian matrix of $r(\cdot)$.

In our example

$$\nabla r(\boldsymbol{\theta}) = \begin{pmatrix} \frac{\partial r_1(\theta)}{\partial \theta_1} & \frac{\partial r_1(\theta)}{\partial \theta_2} \\ \frac{\partial r_2(\theta)}{\partial \theta_1} & \frac{\partial r_2(\theta)}{\partial \theta_2} \\ \vdots & \vdots \\ \frac{\partial r_5(\theta)}{\partial \theta_1} & \frac{\partial r_5(\theta)}{\partial \theta_2} \end{pmatrix} = \begin{pmatrix} exp(\theta_2 x^{(1)}) & x^{(1)}\theta_1 exp(\theta_2 x^{(1)}) \\ exp(\theta_2 x^{(2)}) & x^{(2)}\theta_1 exp(\theta_2 x^{(2)}) \\ exp(\theta_2 x^{(3)}) & x^{(3)}\theta_1 exp(\theta_2 x^{(3)}) \\ exp(\theta_2 x^{(4)}) & x^{(4)}\theta_1 exp(\theta_2 x^{(4)}) \\ exp(\theta_2 x^{(5)}) & x^{(5)}\theta_1 exp(\theta_2 x^{(5)}) \end{pmatrix}$$

NEWTON-RAPHSON IDEA

Hessian is obtained by applying product rule and has elements

$$H_{jk} \quad = \quad 2 \sum_{i=1}^{n} \left(\frac{\partial r_i}{\partial \boldsymbol{\theta}_j} \frac{\partial r_i}{\partial \boldsymbol{\theta}_k} + r_i \frac{\partial^2 r_i}{\partial \boldsymbol{\theta}_j \partial \boldsymbol{\theta}_k} \right)$$

Problem with NR: 2nd derivatives can be challenging to compute!

GAUSS NEWTON FOR LEAST SQUARES

GN approximates H by dropping its second part:

$$\begin{aligned} H_{jk} &= 2\sum_{i=1}^{n}\left(\frac{\partial r_i}{\partial \boldsymbol{\theta}_j}\frac{\partial r_i}{\partial \boldsymbol{\theta}_k} + r_i \frac{\partial^2 r_i}{\partial \boldsymbol{\theta}_j \partial \boldsymbol{\theta}_k}\right) \\ &\approx 2\sum_{i=1}^{n}\left(\frac{\partial r_i}{\partial \boldsymbol{\theta}_j}\frac{\partial r_i}{\partial \boldsymbol{\theta}_k}\right) = 2\nabla r^\top \nabla r. \end{aligned}$$

assuming for all i that

$$\left|\frac{\partial r_i}{\partial \boldsymbol{\theta}_j}\frac{\partial r_i}{\partial \boldsymbol{\theta}_k}\right| \gg \left| r_i \frac{\partial^2 r_i}{\partial \boldsymbol{\theta}_j \partial \boldsymbol{\theta}_k}\right|.$$

This assumption may be valid if:

- Residuals r_i are small in magnitude
- Functions are only “mildly” nonlinear and $\frac{\partial^2 r_i}{\partial \boldsymbol{\theta}_j \partial \boldsymbol{\theta}_k}$ is small.

GAUSS NEWTON FOR LEAST SQUARES

If $\nabla r(\boldsymbol{\theta})^\top \nabla r(\boldsymbol{\theta})$ is invertible, the Gauss-Newton update direction is

$$\begin{aligned} \boldsymbol{d}^{[t]} &= -\left[\nabla^2 g(\boldsymbol{\theta}^{[t]})\right]^{-1} \nabla g(\boldsymbol{\theta}^{[t]}) \\ &= -\left[\nabla r(\boldsymbol{\theta})^\top \nabla r(\boldsymbol{\theta})\right]^{-1} \nabla r(\boldsymbol{\theta})^\top r(\boldsymbol{\theta}), \end{aligned}$$

Advantage: Reduced computational complexity because Hessian does not have to be computed.

LEVENBERG-MARQUARDT ALGORITHM

If $\nabla r(\boldsymbol{\theta}^{[t]})^\top \nabla r(\boldsymbol{\theta}^{[t]})$ singular, use $\nabla r(\boldsymbol{\theta}^{[t]})^\top \nabla r(\boldsymbol{\theta}^{[t]}) + \Delta$ with Δ non-negative diagonal matrix.

$$\Delta = \epsilon \cdot I$$

or

$$\Delta = \epsilon \cdot \mathrm{diag}\left(\nabla r(\boldsymbol{\theta}^{[t]})^\top \nabla r(\boldsymbol{\theta}^{[t]})\right)$$

LMA is an efficient and popular method for solving nonlinear optimization problems.

Note: The diag elements of a pd matrix are always ≥ 0