3.7 Linear Optimization

In the following we will introduce the concept of linear optimization for the example of least square fitting, i.e. we will search for the optimal linear combination of a set of functions to a set of measured data points. The vector $\vec{y}_m$ contains the measured numbers at the N points $\vec{x}_m$ with

$$\vec{y}_m = \begin{pmatrix} y_{m,1} \\ y_{m,2} \\ \vdots \\ y_{m,N} \end{pmatrix} \quad \text{and} \quad \vec{x}_m = \begin{pmatrix} x_{m,1} \\ x_{m,2} \\ \vdots \\ x_{m,N} \end{pmatrix}$$

We search for the best linear fit to a set of J functions $f_j(x)$, i.e. we define J coefficients a_j and

$$y_i(a_1, ..., a_J) := \sum_{j=1}^{J} a_j \, f_j(x_{m,i}) \,, \tag{3.1}$$

to calculate a least square function

$$\chi^2(\vec{a}) := \sum_{i=1}^{N} (y_{m,i} - y_i(\vec{a}))^2 \,, \tag{3.2}$$

where the coefficients are summarized in a vector

$$\vec{a} := \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_J \end{pmatrix} .$$

Eq. (3.1) can be written in a compact form

$$\vec{y} := \tilde{M}\vec{a} \,, \tag{3.3}$$

by defining a $J \times N$ dimensional matrix

$$\tilde{M} := \begin{pmatrix} f_1(x_{m,1}) & f_2(x_{m,1}) & \cdots & f_J(x_{m,1}) \\ f_1(x_{m,2}) & f_2(x_{m,2}) & & f_J(x_{m,2}) \\ \vdots & & \ddots & \vdots \\ f_1(x_{m,N}) & f_2(x_{m,N}) & \cdots & f_J(x_{m,N}) \end{pmatrix} \,,$$

which contains the values of the fit function at the measured points x_i.
Thus we can rewrite Eq. (3.2) in a very compact form

$$\chi^2(\vec{a}) := \sum_{i=1}^{N} \left(y_{m,i} - (\tilde{M}\vec{a})_i \right)^2 = \left| \vec{y}_m - \tilde{M}\vec{a} \right|^2 = \left\langle \vec{y}_m - \tilde{M}\vec{a} | \vec{y}_m - \tilde{M}\vec{a} \right\rangle \,. \tag{3.4}$$

Least square fitting of measured data now means to find the minimum of χ^2 with respect to the parameters a_j which in our case are linear coefficients, i.e. we perform a linear optimization. For solving the problem we have to calculate

$$\frac{\partial \chi^2}{\partial a_j} = 0 \qquad \text{for } j = 1 \text{ to } J \,. \tag{3.5}$$

Combining Eq. (3.4) and Eq. (3.5) we get

$$0 = \frac{\partial \chi^2}{\partial a_j} = 2 \left\langle \vec{y}_m - \tilde{M}\vec{a}_{opt} | - \tilde{M}\vec{e}_j \right\rangle \text{ for } j = 1 \text{ to } J \,, \tag{3.6}$$

where $\vec{e}_j$ is the j th base vector.
Eq. (3.6) can be rewritten as

$$\vec{0} = \tilde{M}^T \vec{y}_m - \tilde{M}^T \tilde{M}\vec{a}_{opt} \,, \tag{3.7}$$

or in it's final form

$$\vec{a}_{opt} = (\tilde{M}^T \tilde{M})^{-1} \tilde{M}^T \vec{y}_m \,. \tag{3.8}$$

Note that the matrix $\tilde{M}^T \tilde{M}$ has $J \times J$ components and can thus be inverted. Eq. (3.8) is the general solution for all linear optimization problems, i.e. all linear optimization problems can be solved just by one matrix inversion. The application of Eq. (3.8) to linear regression, i.e. fitting to a straight line, we will discuss as a homework.