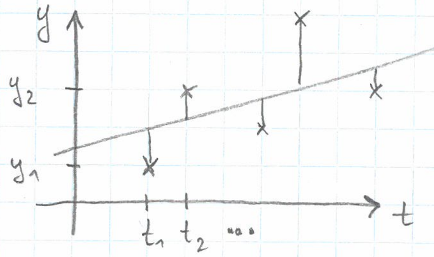
Quadratische Optimierung: Regression

Problemstellung: Fitten einer Gerade durch n Datenpunkte.



$y = d + kt$ ... Fit

$d + kt_i - y_i =: e_i$ ... i-ter Fehler

Gesucht: optimale Werte für d und k.

Vorgehensweise: Das lin. Glg.system

$$\begin{aligned} d + kt_1 &= y_1 \\ d + kt_2 &= y_2 \\ &\vdots \\ d + kt_n &= y_n \end{aligned}$$

lautet in Matrixform

$$\underbrace{\begin{pmatrix} 1 & t_1 \\ 1 & t_2 \\ \vdots & \vdots \\ 1 & t_n \end{pmatrix}}_{A} \underbrace{\begin{pmatrix} d \\ k \end{pmatrix}}_{x} = \underbrace{\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}}_{b}$$

und ist nicht lösbar.

Stattdessen soll der „Fehler minimiert" werden. Genauer:

Der Fehlervektor $e = Ax - b$ soll minimale Länge haben.

Das heißt $\|e\| = \sqrt{e_1^2 + e_2^2 + \ldots + e_n^2}$ soll minimal sein, was äquivalent zu $\|e\|^2 = e_1^2 + e_2^2 + \ldots + e_n^2$ soll minimal sein ist.

→ Methode der kleinsten Quadrate von Gauß, Regression, least squares

Formulierung in Matrixform: $\boxed{\min \|Ax - b\|}$

Lösung in Matrixform: $\boxed{\hat{x} = (A^T A^{-1}) A^T b}$ falls $\ker(A) = \{0\}$

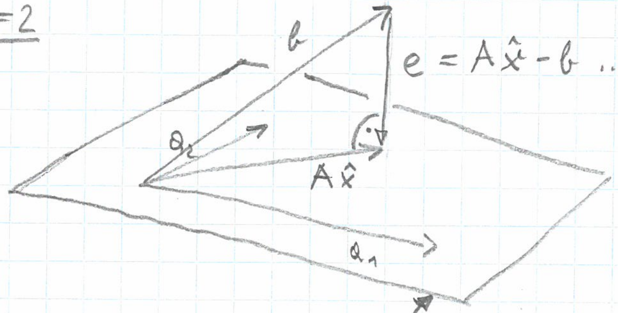
Lösung in Python: $\boxed{\text{lstsq}(A, b)}$

Geometrie:

$$Ax = \Big(\underbrace{a_1 \; a_2 \; \ldots \; a_m}_{\text{Spalten}}\Big) \begin{pmatrix} x_1 \\ \vdots \\ x_m \end{pmatrix} = x_1 a_1 + x_2 a_2 + \ldots + x_m a_m$$

Linearkombination der Spalten von A soll möglichst nahe an b kommen.

$\underline{m=2}$



$e = A\hat{x} - b$ ... opt. Fehlervektor

$A\hat{x} = \hat{x}_1 a_1 + \hat{x}_2 a_2$ ... opt. Linearkomb.

lineare Hülle von a_1 und a_2

Probe, ob zwischen $A\hat{x}$ und $e = A\hat{x} - b$ ein rechter Winkel ist:

$$(A\hat{x})^T e = \hat{x}^T A^T (A\hat{x} - b) =$$
$$= \hat{x}^T A^T \left(A (A^T A)^{-1} A^T b - b\right) =$$
$$= \hat{x}^T \underbrace{\underbrace{A^T A}\,\underbrace{(A^T A)^{-1}}}_{I} A^T b - \hat{x}^T A^T b = 0. \checkmark$$

Tricks:

- Polynomialer Fit: z.B. von Ordnung 2, d.h. quadratisch

$$c_0 + c_1 t_i + c_2 t_i^2 \sim y_i$$

$$\underbrace{\begin{pmatrix} 1 & t_1 & t_1^2 \\ 1 & t_2 & t_2^2 \\ \vdots & \vdots & \vdots \\ 1 & t_n & t_n^2 \end{pmatrix}}_{A} \underbrace{\begin{pmatrix} c_0 \\ c_1 \\ c_2 \end{pmatrix}}_{x} \sim \underbrace{\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}}_{b} \quad \rightarrow \min \|Ax - b\|$$

- Transformation von nicht-linearen Gleichungen zu linearen:

Bsp. Moore's Law

$$n = \alpha^{t - t_0} \quad | \log(\,)$$
$$\log(n) = (t - t_0) \log(\alpha)$$
$$\log(n) = -t_0 \log(\alpha) + \log(\alpha) \cdot t$$
$$\text{vgl.} \quad y = d + k \cdot t$$