Ausgleich von Messwerten

Torsten Hehl

Physikalisches Institut Tübingen

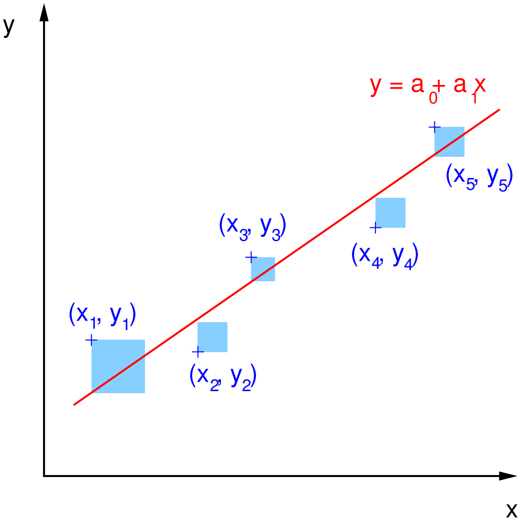
18.12.2023

Anpassung von Messwerten y_i an Modellfunktion y:

$$y(x) = y(x; a_1 \dots a_M)$$

Methode der kleinsten Quadrate (C.F. Gauss)

$$\sum_{i=1}^{N} [y_i - y(x_i; a_1 \dots a_M)]^2 \to \text{Min.}$$



Annahme: Messwerte y_i seien jeweils normalverteilt:

$$P_i = \frac{1}{\sqrt{2\pi}\sigma_i} e^{-\frac{(y_i - y)^2}{2\sigma_i^2}}$$

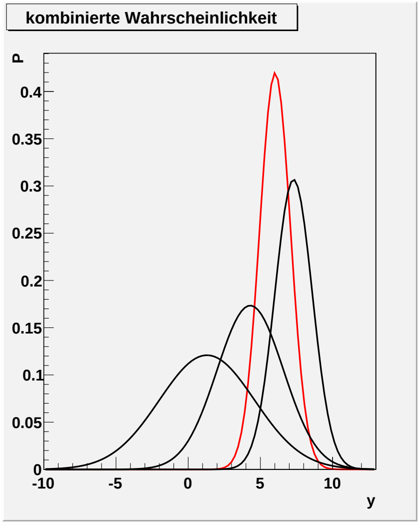
max. Wahrscheinlichkeit für 1 Punkt: $y = y_i$

Viele Punkte $\rightarrow$ kombinierte Wahrscheinlichkeit

Maximum Likelihood für Normalverteilungen

$$P = \prod_{i=1}^{N} P_i \sim \prod_{i=1}^{N} e^{-\frac{1}{2}\left(\frac{y_i - y(x_i)}{\sigma_i}\right)^2} \rightarrow \text{Max.}$$

$$-\ln P \sim \chi^2 = \sum_{i=1}^{N} \left(\frac{y_i - y(x_i)}{\sigma_i}\right)^2 \rightarrow \text{Min.}$$



χ^2-Verteilungsfunktion (Helmert 1875, Pearson 1900)

$$F(\chi^2) = \frac{1}{\Gamma(\lambda) 2^{\lambda}} \int_0^{\chi^2} u^{\lambda-1} e^{-\frac{1}{2}u} du \quad .$$

$$\lambda = \frac{N-M}{2} \,, \quad E(\chi^2) = (N-M) \,, \quad \sigma^2(\chi^2) = 2(N-M)$$

$\longrightarrow \frac{\chi^2}{N-M}$: Maß für Güte der Anpassung (reduziertes χ^2)

- ≈ 1: Gute Anpassung
- $\gg 1$: Inkonsistenz zwischen Daten und Modellfunktion, meist Fehler zu klein angegeben
- $\ll 1$: Anpassung ist konsistent, aber Fehler zu groß oder Daten geschönt

Bsp.: Messung der Lichtgeschwindigkeit (1948)

Bergstrand (1948)	299 793(2) km/s
Essen et al. (1947)	299 792.0(45) km/s
Jones (1947)	299 782(25) km/s

$$\chi^2 = \sum_{i=1}^{3} \left(\frac{c_i - \bar{c}}{\Delta c_i} \right)^2 \to \text{Min.}$$

$$\frac{d\chi^2}{d\bar{c}} = 0 = \sum_{i=1}^{3} 2 \frac{c_i - \bar{c}}{\Delta c_i^2} = 0$$

$$\longrightarrow \bar{c} = \frac{\sum\limits_{i=1}^{3} \frac{c_i}{\Delta c_i^2}}{\sum\limits_{i=1}^{3} \frac{1}{\Delta c_i^2}}$$

Lineare Modellfunktion

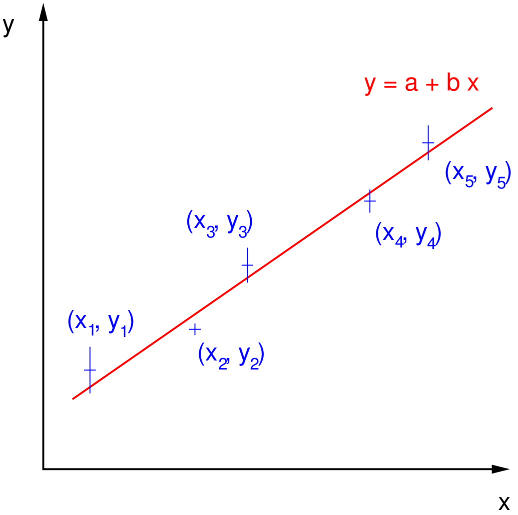
$$y(x) = y(x; a, b) = a + bx$$

$$\chi^2(a, b) = \sum_{i=1}^{N} \left(\frac{y_i - a - bx_i}{\sigma_i} \right)^2 \quad \rightarrow \quad \text{Min.}$$

Daraus folgen die Bedingungen

$$0 = \frac{\partial \chi^2}{\partial a} = -2 \sum_{i=1}^{N} \frac{y_i - a - bx_i}{\sigma_i^2}$$

$$0 = \frac{\partial \chi^2}{\partial b} = -2 \sum_{i=1}^{N} \frac{x_i(y_i - a - bx_i)}{\sigma_i^2}$$



Mit den Abkürzungen

$$S \equiv \sum_{i=1}^{N} \frac{1}{\sigma_i^2} \;, \quad S_x \equiv \sum_{i=1}^{N} \frac{x_i}{\sigma_i^2} \;, \quad S_y \equiv \sum_{i=1}^{N} \frac{y_i}{\sigma_i^2}$$

$$S_{xx} \equiv \sum_{i=1}^{N} \frac{x_i^2}{\sigma_i^2} \;, \quad S_{xy} \equiv \sum_{i=1}^{N} \frac{x_i y_i}{\sigma_i^2}$$

erhalten wir das Gleichungssystem:

$$\begin{aligned} aS + bS_x &= S_y \\ aS_x + bS_{xx} &= S_{xy} \end{aligned}$$

Lösungen: (mit der Determinante $D = SS_{xx} - S_x^2$):

$$a = \frac{S_{xx}S_y - S_x S_{xy}}{D}$$

$$b = \frac{SS_{xy} - S_x S_y}{D}$$

Fehler der Koeffizienten (aus Gaussschem FFG)

$$\sigma_f^2 = \sum_{i=1}^{N} \sigma_i^2 \left(\frac{\partial f}{\partial y_i} \right)^2$$

Hier:

$$\begin{aligned} \frac{\partial a}{\partial y_i} &= \frac{S_{xx} - S_x x_i}{\sigma_i^2 D} \\ \frac{\partial b}{\partial y_i} &= \frac{S x_i - S_x}{\sigma_i^2 D} \end{aligned}$$

Damit erhält man:

$$\begin{aligned} \sigma_a^2 &= \frac{S_{xx}}{D} \\ \sigma_b^2 &= \frac{S}{D} \end{aligned}$$

Lineare Koeffizienten

Modellfunktion als Summe von beliebigen $f(x)$ mit linearen Koeffizienten:

$$y(x; a_1, \ldots a_M) = \sum_{j=1}^{M} a_j f_j(x)$$

Dann ergibt sich lineares Gleichungssystem M-ter Ordnung:

$$\frac{\partial}{\partial a_j} \sum_{i=1}^{N} \frac{\left(y_i - \sum_{j=1}^{M} a_j f_j(x_i) \right)^2}{\sigma_i^2} = 0$$

Beispiele:
Polynom $M-1$-ten Grades: $y = a_1 + a_2 x + a_3 x^2 + \ldots + a_M x^{M-1}$
Legendrepolynome: $P_{j-1}(x) \equiv f_j(x)$ mit Rekursion:

$$P_0(x) = 1, \qquad P_1(x) = x$$

$$(n+1)P_{n+1}(x) = (2n+1)xP_n(x) - nP_{n-1}(x)$$

Transformation von y so, dass lineare Funktion y' entsteht:
Beispiel:

$$y(x; a, b) = e^{a+bx_i}$$

Transformation (Logarithmus):

$$y' = \ln y = a + bx_i$$

Wichtig: Transformation der Fehler σ_i nicht vergessen!

$$(\sigma_i')^2 = \sigma_i^2 \frac{dy_i'}{dy_i} = \sigma_i^2 \frac{d}{dy_i} \ln y_i = \frac{\sigma_i^2}{y_i}$$

Zu minimieren ist dann:

$$\chi^2(a, b) = \sum_{i=1}^{N} \frac{y_i(\ln y_i - a - bx_i)^2}{\sigma_i^2} \rightarrow \mathrm{Min.}$$

Suche des Minimums von χ^2 im M-dimensionalen Raum der Parameter a_j

- M Minimalbedingungen führen zu nichtlinearem GS
- sinnvoller Startvektor a_o nahe Minimum

Näherung durch quadratische Form

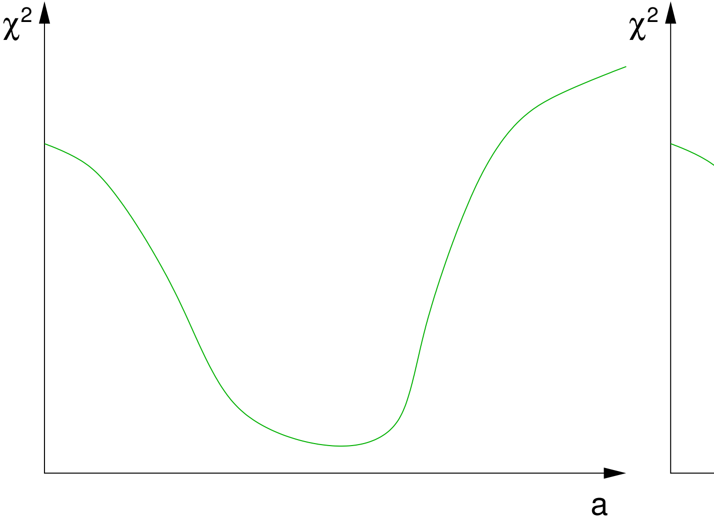
Eindimensionaler Fall:

$$\frac{d\chi^2}{da} = b \qquad \frac{d^2\chi^2}{da^2} = A$$

$\rightarrow$ verbesserter Parameter

$$a' = a_o + \frac{b}{A}$$

Minimum der Parabel



Quadratische Näherung

$$\chi^2(\mathrm{a}) = c + \mathrm{b} \cdot \mathrm{a} + \frac{1}{2}\mathrm{a}^T \mathrm{A}\mathrm{a}$$

Gradient b_k

$$\mathrm{b} = \mathrm{grad}\,\chi^2(\mathrm{a}_0)\ , \qquad b_i = \left.\frac{\partial \chi^2}{\partial a_i}\right|_{\mathrm{a}=\mathrm{a}_0}$$

Hesse-Matrix A_{ik}:

$$A_{ik} = \left.\frac{\partial^2 \chi^2}{\partial a_i \partial a_k}\right|_{\mathrm{a}=\mathrm{a}_0}$$

$$\rightarrow \mathrm{a}' = \mathrm{a} + \mathrm{b}\mathrm{A}^{-1}$$

Probleme:

- ineffizienter Zick-Zack-Kurs bei Verfolgung des steilsten Abfalls
- Problem mit A^{-1}, wenn A nicht positiv definit

Lösung:

- konjugierte Richtungen u und v nötig mit $\mathrm{uAv} = 0$
- Näherung der invertierten Hessematrix in variabler Metrik $\rightarrow$ Quasi-Newton-Verfahren

Nelder u. Mead 1965

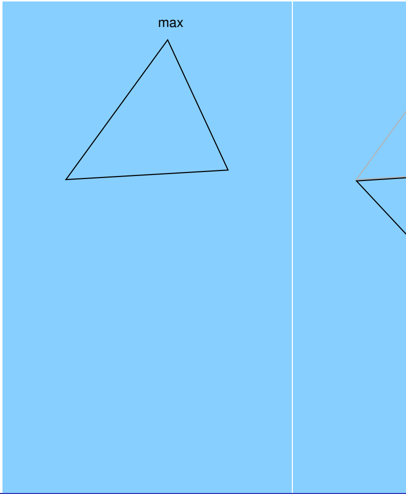
Simplex

$M + 1$ Punkte a_i im M-dimensionalen Raum der anzupassenden Parameter (2-dim.: Dreieck)

Schrittweise Näherung an Minimum durch:

- Spiegelung von schlechtestem Punkt
- Streckung, wenn gespiegelter Punkt der beste ist
- Kontraktion, wenn gespiegelter Punkt größer als zweitgrößter ist

Verfahren ist robust, aber langsam



Weitere Verfahren

- Unbeschränkte Parameter
  - **Newton-CG** (Nonlinear conjugate gradient method)
  - **BFGS** (Broyden–Fletcher–Goldfarb–Shanno, modifiziertes Quasi-Newton-Verfahren)
  - **dogleg** (dog-leg trust-region algorithm)
  - **trust-ncg, trust-krylov, trust-exact**
- Beschränkte/gebundene Parameter
  - **Powell-Methode** (modifiziertes CG-Verfahren)
  - **L-BFGS-B** (Variante von BFGS)
  - **TNC** (Truncated Newton method, ohne Hesse-Matrix)
  - **COBYLA** (Constrained Optimization BY Linear Approximation)
  - **SLSQP** (Sequential Least SQuares Programming, mod. BFGS)
  - **trust-constr**

Umsetzung in Python: scipy.optimize