

Bei der wissenschaftlichen Analyse von Daten in den biologischen Wissenschaften („life sciences“) wird sehr viel „gesündigt“, und manchmal wird es auch schlichtweg falsch gemacht. Besonders augenfällig ist das am Beispiel der Anpassung (Modellfindung) von nichtlinearen Kurven an gemessene Daten (curve-fitting) in veröffentlichten Originalarbeiten zu studieren.

Meistens wird dabei nach folgendem Schema vorgegangen: Linearisieren der Daten, Berechnung der Ausgleichsgeraden nach der Methode der kleinsten Quadrate, daraus Rückrechnung der Kurvenparameter. Aber das ist falsch! Zwar wird bei diesem Verfahren für die Gerade die Sum-

me der Abstandsquadrate ein Minimum, aber für die Kurve mit den errechneten Parametern gilt das nicht mehr. Die so ermittelte Kurve ist den Meßdaten (leider) nicht angepaßt, d. h. das Modell ist nicht richtig verifiziert worden.

Ein Beispiel soll das verdeutlichen. Für die sehr oft in der Praxis vorkommende Funktion $y = f(x) = a \cdot e^{bx}$ sollen die zwei Parameter a und b aufgrund der Meßdaten $x_i \mapsto y_i$ ermittelt werden. Üblicherweise wird die Funktion durch Logarithmieren linearisiert, d. h. es gilt dann $\log y = \log a + b x \log e$. Daraus werden a und b bestimmt. Was passiert nun, wenn $b < 0$ ist (abklingende e-Funktion)? In diesem Fall können „verrauschte“ Daten,

die echt sind, für große x -Werte ein $y = 0$ erzwingen. Ein drastischer Effekt, denn der Logarithmus von 0 ist Unendlich. Es stellt sich somit die Frage, mit welcher Genauigkeit dann überhaupt noch a und b bestimmt werden können. Oft werden solche „störenden“ Werte nicht in die Berechnung einbezogen, was natürlich keine Lösung ist.

In diesem Merkblatt soll gezeigt werden, wie dieses Problem wissenschaftlich sauber und auf solider mathematischer Basis – im Sinne einer „Art of Scientific Computing“ [4] – zu lösen ist. Allerdings wird das dazu notwendige iterative Berechnungsverfahren etwas Rechenzeit auf einem Computer „kosten“.

1. Definition von Größen und Notation:

- e = Eulersche Zahl = 2,718 281 828 ...
- x = Unabhängige Variable.
- y = Von x abhängige Variable.
- Q = Qualitätsfunktion.
- $\dots \mapsto \dots = \dots$ ist zugeordnet ...
- $\dots \in \dots = \dots$ ist Element von ...
- $\dots \subseteq \dots = \dots$ enthalten in ...
- $\|\dots\|$ = Norm von ...
- $|\dots|$ = Betrag von ...
- $\dots|_a = \dots$ an der Stelle a .
- \mathbf{R}^n = Euklidischer Raum von n Dimensionen.
- $L^2(\mathbf{R}^n)$ = Hilbert Raum, ein Sonderfall eines Banachraumes.
Menge der meßbaren, quadratintegrablen n -dimensionalen Funktionen.

- l^2 = Hilbertscher Folgenraum. Menge der Folgen $s = (a_n)$ komplexer (und reeller) Zahlen für die $\sum_{n=1}^{\infty} |a_n|^2 < \infty$ ist, also konvergieren.
- f, g = Reelle Funktionen im $L^2(\mathbf{R})$.
- \mathbf{p} = Parametervektor.
- \mathbf{F} = M Funktionen f_1, \dots, f_M in einem Gebiet $G \in \mathbf{R}^n$.
- $\frac{\partial \mathbf{F}}{\partial \mathbf{p}}|_a$ = Gateaux - Ableitung von \mathbf{F} an der Stelle a ; das ist eine Funktionalmatrix [3].
- \mathbf{A} = Matrix.
- \mathbf{A}^T = Transponierte Matrix von \mathbf{A} .
- \mathbf{A}^{-1} = Zu \mathbf{A} inverse Matrix.

2. Die Problemstellung:

Gemessene Werte: Es seien N Werte y_i in Abhängigkeit von x_i gemessen worden, d. h. in der x, y -Ebene sind N Wertepaare $x_i \mapsto y_i$ ($i = 1, 2, \dots, N$) mit $x_i \neq x_j$ für $i \neq j$ gegeben.

Gesuchte Funktion: Gesucht ist eine von x und M Parametern p_k abhängige Funktion f (Modell), deren Graph sich „möglichst gut“ den gemessenen Punkten anpaßt.

$$f = f(x, p_1, p_2, \dots, p_k, \dots, p_M) \text{ mit } M < (N - 1). \quad (1)$$

Was heißt „möglichst gut“? Der Abstand zweier quadratintegrabler Funktionen $f(x)$ und $g(x)$ ist die euklidische Norm der Funktion $f - g$, d. h. die Zahl

$$\|f - g\| = \sqrt{\int_a^b |f(x) - g(x)|^2 dx} \quad (2)$$

Dieser Abstand ist die mittlere quadratische Abweichung zwischen den Funktionen f und g , was auch das geeignete Maß für die Güte der gesuchten Anpassung (curve-fitting) ist. Als Bedingung ist nun zu fordern, daß diese Norm minimal werden soll. Im diskreten Fall ist das Integral durch eine Summe zu ersetzen, womit sich ergibt

$$Q = \|f - y_i\|^2 = \sum_{i=1}^N [f(x_i, p_1, p_2, \dots, p_M) - y_i]^2 \Rightarrow \text{Min} \quad (3)$$

3. Ermittlung der Parameter p_k :

Die Parameter p_k müssen also nach der Methode der kleinsten Abstandsquadrate (Gauß) so gewählt werden, daß

$$Q = Q(p_1, p_2, \dots, p_k, \dots, p_M) \quad (4)$$

ein Minimum wird. Q ist (allgemein) eine Hyperebene in \mathbf{R}^M und sei zweimal stetig differenzierbar. Nach der Theorie der relativen Extrema gelten als notwendige Bedingungen:

$$\frac{\partial Q}{\partial p_k} = 0 \quad (k = 1, \dots, M) \quad (5)$$

$$\frac{\partial Q}{\partial p_k} = 2 \cdot \sum_{i=1}^N [f(x_i, p_1, p_2, \dots, p_M) - y_i] \cdot \frac{df}{dp_k} = 0 \quad (6)$$

$$F(p_k) = \sum_{i=1}^N f \frac{df}{dp_k} - \sum_{i=1}^N y_i \frac{df}{dp_k} = 0 \quad (k = 1, \dots, M) \quad (7)$$

Mit diesen M Gleichungen (7) ist zwar das Problem der Ermittlung der M Parameter p_k eindeutig bestimmt, dennoch ist dieses Normalgleichungssystem im allgemeinen – insbesondere für nichtlineare Funktionen f – nicht explizit lösbar. Im Zeitalter der Computer macht das aber nichts. Es muß dann eben ein iteratives Lösungsverfahren angewandt werden.

4. Das iterative Lösungsverfahren:

Mit Glchg. (7) sind $F_1(\mathbf{p}), F_2(\mathbf{p}), \dots, F_k(\mathbf{p}), \dots, F_M(\mathbf{p})$ stetige Funktionen von M unabhängigen Variablen $\mathbf{p} = (p_1, p_2, \dots, p_M)$ über einen gemeinsamen Definitionsbereich $\mathbf{D} \subseteq \mathbf{R}^M$ gegeben. Gibt es nun einen Vektor $\mathbf{p}_L \in \mathbf{D}$, für den für alle $F_k(\mathbf{p}_L) = 0$ gilt, so ist \mathbf{p}_L der Lösungsvektor des Gleichungssystems (7) $\mathbf{F}(\mathbf{p}) = 0$. Die Elemente von \mathbf{p}_L sind die gesuchten Parameter p_k .

Entwickelt man (7) nach dem Verfahren von Newton-Kantorowitch [1] in eine Taylorreihe, wobei \mathbf{p}_0 eine 0. Näherung von \mathbf{p}_L sei, ergibt sich in Vektorschreibweise

$$\mathbf{F}(\mathbf{p}) = \mathbf{F}(\mathbf{p}_0) + \left. \frac{\partial \mathbf{F}}{\partial \mathbf{p}} \right|_{\mathbf{p}=\mathbf{p}_0} \cdot (\mathbf{p} - \mathbf{p}_0) + \text{Rest}(\mathbf{p}) = 0. \quad (8)$$

Dabei ist $\mathbf{A} = \left. \frac{\partial \mathbf{F}}{\partial \mathbf{p}} \right|_{\mathbf{p}=\mathbf{p}_0}$ die $(M \times M)$ -Funktionalmatrix (Jacobische Matrix) an der Stelle $\mathbf{p} = \mathbf{p}_0$, die nicht von den Meßwerten abhängt.

$$\mathbf{A} = \left. \frac{\partial \mathbf{F}}{\partial \mathbf{p}} \right|_{\mathbf{p}=\mathbf{p}_0} = \begin{bmatrix} \frac{\partial F_1}{\partial p_1} & \dots & \frac{\partial F_M}{\partial p_1} \\ \vdots & \ddots & \vdots \\ \frac{\partial F_1}{\partial p_M} & \dots & \frac{\partial F_M}{\partial p_M} \end{bmatrix}_{\mathbf{p}=\mathbf{p}_0} \quad (9)$$

Läßt man das Restglied weg, erhält man aus (8) mit

$$\mathbf{q} = \mathbf{p} - \mathbf{p}_0 \quad (10)$$

als Linearisierung die Matrixgleichung

$$\mathbf{F}(\mathbf{p}) = \mathbf{A}\mathbf{q} + \mathbf{F}(\mathbf{p}_0) = 0 \quad (11)$$

$$\mathbf{A}\mathbf{q} = -\mathbf{F}(\mathbf{p}_0) \quad (12)$$

Die M Gleichungen (7) lassen sich in einer Vektorgleichung zusammenfassen:

$$\mathbf{F}(\mathbf{p}) = \sum_{i=1}^N \frac{d\mathbf{f}}{d\mathbf{p}} \Big|_{x_i} \cdot [\mathbf{f} - \mathbf{y}] \quad (13)$$

Darin ist

$$\mathbf{D} = \left. \frac{\partial \mathbf{f}}{\partial \mathbf{p}} \right|_{\mathbf{x}} = \begin{bmatrix} \frac{\partial f}{\partial p_1} \Big|_{x_1} & \dots & \frac{\partial f}{\partial p_1} \Big|_{x_i} & \dots & \frac{\partial f}{\partial p_1} \Big|_{x_N} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \frac{\partial f}{\partial p_k} \Big|_{x_1} & \dots & \frac{\partial f}{\partial p_k} \Big|_{x_i} & \dots & \frac{\partial f}{\partial p_k} \Big|_{x_N} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \frac{\partial f}{\partial p_M} \Big|_{x_1} & \dots & \frac{\partial f}{\partial p_M} \Big|_{x_i} & \dots & \frac{\partial f}{\partial p_M} \Big|_{x_N} \end{bmatrix} \quad (14)$$

die $(M \times N)$ -Funktionalmatrix im durch den Vektor \mathbf{x} (die unabhängige Variable der N Meßwerte) gegebenen Punkt. \mathbf{D} ist also von den Meßwerten abhängig. Aus (13) wird damit

$$\mathbf{F}(\mathbf{p}) = \mathbf{D} \cdot [\mathbf{f} - \mathbf{y}]. \quad (15)$$

Es läßt sich nun zeigen, daß sich die Matrix \mathbf{A} aus \mathbf{D} berechnet mit

$$\mathbf{A} = \mathbf{D} \circ \mathbf{D}^T. \quad (16)$$

Aus (12) ergibt sich für das iterative Lösungsverfahren mit (15) die endgültige Matrixgleichung

$$\mathbf{A}\mathbf{q} = -\mathbf{F}(\mathbf{p}) = \mathbf{D} \cdot [\mathbf{f} - \mathbf{y}] \quad (17)$$

mit $\mathbf{q} = \mathbf{p} - \mathbf{p}_0$.

Die Lösung dieser Matrixgleichung, d. h. die Suche nach den Lösungsvektoren von \mathbf{q} bzw. \mathbf{p} , für den (17)

erfüllt ist, kann mit den Standardverfahren der numerischen Mathematik gefunden werden.

5. Der Algorithmus:

Aufgrund der Matrixgleichung (17) ergibt sich der iterative Algorithmus zur Ermittlung der Parameter p_k :

1. Auswahl einer Näherung \mathbf{p}_n des Lösungsvektors \mathbf{p}_L .
2. Berechnung der Matrix \mathbf{D} nach (14).
3. Berechnung von $\mathbf{F}(\mathbf{p}_n)$ nach (15).
4. Berechnung der Matrix $\mathbf{A}(\mathbf{p}_n)$ nach (16).
5. Lösung von (17) $\mathbf{A}(\mathbf{p}_n)\mathbf{q} = -\mathbf{F}(\mathbf{p}_n)$ liefert \mathbf{q}_n , womit sich
6. eine neue Näherung \mathbf{p}_{n+1} aufgrund von (10) ergibt: $\mathbf{p}_{n+1} = \mathbf{p}_n + \mathbf{q}_n$.
7. Mit \mathbf{p}_{n+1} wird die Iteration solange wiederholt bis $|\mathbf{A}\mathbf{q} + \mathbf{F}(\mathbf{p}_n)| < \varepsilon$, einer vorgegebenen Fehlergrenze ist.
8. Wenn das der Fall ist, ist der Lösungsvektor $\mathbf{p}_L = \mathbf{p}_{n+1}$, womit die Parameter p_k der gesuchten Funktion f gefunden sind.

Eine Wichtung der Meßwerte \mathbf{y} ist in diesem Algorithmus nicht dargestellt. Durch Einführung eines geeigneten Gewichtsvektors \mathbf{w} kann dieses berücksichtigt werden.

6. Die Realisierung:

Neue Softwarepakete der Signaldatenverarbeitung stellen besonders ausgefeilte Programme zur direkten Lösung der Matrixgleichung vom Typ $\mathbf{A}\mathbf{x} = \mathbf{b}$ zur Verfügung, die auch (fast) singuläre Matrizen handhaben können.

Stehen solche Programme nicht zur Verfügung, kann man versuchen (17) iterativ mit elementaren Routinen der Matrizenalgebra zu lösen, denn aus (17) folgt:

$$\mathbf{p}_{n+1} = \mathbf{A}^{-1} \mathbf{D} [\mathbf{y} - \mathbf{f}] + \mathbf{p}_n \quad (18)$$

Auf dieser Basis entstand bereits 1971 in der Abt. Wiss. Datenverarbeitung für den DEC Computer PDP-15 zusammen mit einem einfachen Matrizen-Paket das FORTRAN-Programm „APPRX“ (P 229), das in sehr vielen Fällen des „Curve-Fittings“ hilfreich war; aber nicht in den Fällen, wenn die Meßdaten fast singuläre Matrizen ergaben.

Mit dem heute den Wissenschaften zur Verfügung stehenden Softwarepaket „IDL – Interactive Data Language“ [6] ist eine wesentlich bessere Lösung möglich. IDL stellt mit den Routinen *SVBKS* und *SVD* Programme zur direkten Lösung von Gleichung (17) zur Verfügung.

Damit läßt sich ein universelles „Curve-Fitting“ realisieren [7], was die eingangs dargestellten Probleme vermeidet.

7. Literatur:

- [1] Bronstein, I. N. und K. A. Semendjajew: Taschenbuch der Mathematik. Thun: Harri Deutsch 1980 (19. Auflage). ISBN: 3-871-44492-8. Preis: 39,80 DM. – [BU 693].
- [2] Reinhardt, F. und H. Soeder: DTV-Atlas zur Mathematik. Band 1: Grundlagen, Algebra und Geometrie. München: DTV 1980 (4. Auflage). ISBN: 3-423-03007-0. Preis: 14,80 DM. – [BU 608].
- [3] Reinhardt, F. und H. Soeder: DTV-Atlas zur Mathematik. Band 2: Analysis und angewandte Mathematik. München: DTV 1980 (3. Auflage). ISBN: 3-423-03008-9. Preis: 14,80 DM. – [BU 609].
- [4] Press, Flannery, Teukolsky, and Vetterling: Numerical Recipes – The Art of Scientific Computing. Cambridge (USA): Cambridge University 1987.
- [5] Meschkowski, H.: Mathematisches Begriffswörterbuch. BI Hochschul-Taschenbücher Nr. 99/99a. Mannheim: Bibliographisches Institut 1966 (2. Auflage). – [BU 131].
- [6] Research Systems, Inc.: IDL – Interactive Data Language. Version 2.1. Boulder (USA): Research Systems 1991 (Edition vom 2.4.1991). E-Mail: idl@boulder.colorado.edu.
- [7] Dittberner, K.-H.: Vektoren und Matrizen in der Signaldatenverarbeitung. FU Berlin (IfP): wdv-notes Nr. 175, 1979–1991.