

Berechnung von HELMERT-Transformationen mit dem NEWTON-Verfahren

Helmut Späth

Je nach Raumdimension ($n = 2, n = 3, n > 3$) und Maßstabsfaktoren (keine, einen, mehrere) ergeben sich unterschiedliche Arten von HELMERT-Transformationen. Für den Fall $n = 2$ und einem Maßstabsfaktor zeigen wir, wie man das NEWTON-Verfahren einsetzen kann. Das Vorgehen kann prinzipiell auf beliebige Dimension und beliebige Maßstabsfaktoren übertragen werden.

1 Problemstellung

Gegeben seien Vektoren $x_i, y_i \in \mathbb{R}^n, i = 1, \dots, m$. Die allgemeinste Form einer HELMERT-Transformation ist $x_i \rightarrow t + UQx_i$, wobei

$t \in \mathbb{R}^n$ ein Verschiebungsvektor,

$U = \text{diag}(u_1, \dots, u_n)$ eine Diagonalmatrix von Maßstabsfaktoren und

$Q \in \mathbb{R}^{n \times n}$ eine orthogonale Drehmatrix ist.

Diese Größen werden so gesucht, dass

$$S(t, U, Q) = \frac{1}{2} \sum_{i=1}^m \|y_i - t - UQx_i\|_2^2 \quad (1)$$

minimal wird, d.h. dass die Bildpunkte $t + UQx_i$ von den y_i im Sinne der kleinsten Quadrate möglichst wenig abweichen. Spezialfälle hiervon erhält man, indem man die Dimension

Fall A: $n = 2$

Fall B: $n = 3$

Fall C: $3 < n < \infty$

und unterschiedlicher Maßstabsfaktoren

Fall α : $U = I$ (Einheitsmatrix), d.h. kein Maßstabsfaktor

Fall β : $U = uI$ ($u \in \mathbb{R}, u \neq 1$), d.h. ein Maßstabsfaktor

Fall γ : $U = \text{diag}(u_1, \dots, u_n)$, d.h. mehrere Maßstabsfaktoren

beliebig miteinander kombiniert, was 9 Möglichkeiten ergibt.

Viele verschiedene Fälle und zugehörige numerische Verfahren sind z.B. für $A\alpha$ in [6], $A\beta$ und $A\gamma$ in [8], $B\alpha$ in [5], $B\beta$ in [10], $B\gamma$ in [2, 3, 7] und $C\alpha$ in [1, 9, 11] diskutiert. Für den Fall A kann

$$Q = \begin{pmatrix} \cos \varphi & \sin \varphi \\ -\sin \varphi & \cos \varphi \end{pmatrix},$$

für den Fall B kann Q als Produkt von den drei elementaren Drehungen und im Fall C kann Q als Produkt von $\frac{n(n-1)}{2}$ elementaren Drehungen angesetzt werden. Die bisher vorgestellten Verfahren erscheinen für alle Kombinationen i.a. nur linear konvergent, d.h. benötigen i.a. sehr viele Iterationen. Betrachtet man (1) zusammen mit der Nebenbedingung (Orthogonalität von Q)

$$Q^T Q = I \quad (2)$$

und wendet auf die notwendigen Bedingungen für die zu (1) und (2) gehörige LAGRANGE-Funktion das NEWTON-Verfahren an, so spart man sich die Faktorisierung von Q durch elementare Drehungen und erhält im Falle der Konvergenz eine quadratisch konvergente Methode. Dies wird für den Fall $A\beta$ vorgeführt, kann aber auch prinzipiell für die restlichen acht Kombinationsmöglichkeiten angewandt werden.

2 Der Fall $n = 2$ und $U = uI$

Setzen wir

$$t = \begin{pmatrix} x \\ y \end{pmatrix}, Q = \begin{pmatrix} p & q \\ r & s \end{pmatrix}, x_i = \begin{pmatrix} a_i \\ b_i \end{pmatrix}, y_i = \begin{pmatrix} c_i \\ d_i \end{pmatrix} \quad (3)$$

so wird aus (1)

$$S(x, y, u, p, q, r, s) = \frac{1}{2} \sum_{i=1}^m [(c_i - x - upa_i - uqb_i)^2 + (d_i - y - ura_i - usb_i)^2] \quad (4)$$

und aus der Nebenbedingung (2) wird

$$p^2 + r^2 = 1, q^2 + s^2 = 1, pq + rs = 0. \quad (5)$$

Genau für $n = 2$ hätte man wegen $p = s = \cos \varphi$ und $q = -r = \sin \varphi$ z.B. die beiden letzten Bedingungen in (5) auch weglassen können. Dies tun wir deswegen nicht, da für $n > 2$ ein derartiges Vorgehen nicht möglich wäre. Für $n = 3$ würde man 6 zu (5) ähnliche Bedingungen erhalten.

Die LAGRANGE-Funktion zu (4) und (5) lautet

$$L(x, y, u, p, q, r, s, \lambda, u, v) = S(x, y, u, p, q, r, s) + \lambda(p^2 + r^2 - 1) + \mu(q^2 + s^2 - 1) + v(pq + rs).$$

Das Verschwinden deren partieller Ableitungen nach den jetzt zehn Variablen ergibt die notwendigen Bedingungen für ein Minimum. Wir erhalten

$$\frac{\partial L}{\partial x} = \sum_{i=1}^m (c_i - x - upa_i - uqb_i) = 0, \quad (6)$$

$$\frac{\partial L}{\partial y} = \sum_{i=1}^m (d_i - y - ura_i - usb_i) = 0, \quad (7)$$

$$\frac{\partial L}{\partial u} = u \sum_{i=1}^m ((pa_i + qb_i)^2 + (ra_i + sb_i)^2) - \sum_{i=1}^m ((pa_i + qb_i)(c_i - x) + (ra_i + sb_i)(d_i - y)) = 0, \quad (8)$$

$$\frac{\partial L}{\partial p} = u \left[\sum_{i=1}^m c_i a_i - x \sum_{i=1}^m a_i - up \sum_{i=1}^m a_i^2 - uq \sum_{i=1}^m a_i b_i \right] + 2\lambda p + vq = 0, \quad (9)$$

$$\frac{\partial L}{\partial q} = u \left[\sum_{i=1}^m c_i b_i - x \sum_{i=1}^m b_i - up \sum_{i=1}^m a_i b_i - uq \sum_{i=1}^m b_i^2 \right] + 2\mu q + vp = 0, \quad (10)$$

$$\frac{\partial L}{\partial v} = u \left[\sum_{i=1}^m d_i a_i - y \sum_{i=1}^m a_i - ur \sum_{i=1}^m a_i^2 - us \sum_{i=1}^m a_i b_i \right] + 2\lambda r + vs = 0, \quad (11)$$

$$\frac{\partial L}{\partial s} = u \left[\sum_{i=1}^m d_i b_i - y \sum_{i=1}^m b_i - ur \sum_{i=1}^m a_i b_i - us \sum_{i=1}^m b_i^2 \right] + 2\mu s + vr = 0, \quad (12)$$

$$\frac{\partial L}{\partial \lambda} = p^2 + r^2 - 1 = 0, \quad (13)$$

$$\frac{\partial L}{\partial \mu} = q^2 + s^2 - 1 = 0, \quad (14)$$

$$\frac{\partial L}{\partial v} = pq + rs = 0. \quad (15)$$

Da die Werte λ , μ und v bei Gleichungen als Nebenbedingungen nicht interessieren, kann man sie auch eliminieren. Dazu lösen wir (9) und (11) nach λ und $v = v_1$ und (10) und (12) nach μ und $v = v_2$ auf. Die Formeln für λ und μ sind nicht weiter interessant. Es muss aber $v_1 = v_2$ gelten. Dies ergibt sich für $ps \neq qr$ (bei einer orthogonalen Matrix stets erfüllt) die Gleichung

$$-\sum_{i=1}^m (d_i - y)(a_i p + b_i q) + \sum_{i=1}^m (c_i - x)(a_i r + b_i s) = 0. \quad (16)$$

Somit haben wir jetzt noch 7 nichtlineare Gleichungen (6), (7), (8), (13), (14), (15) und (16) mit den 7 Unbekannten x , y , u , p , q , r , s numerisch aufzulösen. Da bietet sich das NEWTON-Verfahren an (z.B. Subroutine TAYLOR aus [4]), das im Falle der Konvergenz quadratisch nach ca. 6–8 Iterationen konvergiert.

Um Startwerte zu erhalten, wählen wir den Drehwinkel φ zufällig und gleichverteilt im Intervall $(0, 2\pi]$ und setzen $p = s = \cos \varphi$, $q = -r = \sin \varphi$. Für u wählen wir eine um $\frac{1}{2}$ erhöhte gleichverteilte Zufallszahl im Intervall $(0, 1]$. Für x und y lösen wir die Gleichungen (6) bzw. (7) danach auf, nachdem wir dort die bereits gewählten Werte für p , q , r , s und u eingesetzt haben. Für die beiden folgenden Beispiele haben wir jeweils 10 Startwerte für φ genommen und zu jedem hiervon 10 neue Werte für u , was insgesamt 100 Startwerte ergibt.

Beispiel 1: Die Daten ($m = 5$) stammen aus [1] und wurden schon in [8] benutzt:

a_i	b_i	c_i	d_i
1	0	0.993808	0.004709
1	1	1.007303	1.493057
0	1	-0.015174	0.981590
0	0	-0.009153	-0.021163
.5	.5	0.381458	0.498160

Die Resultate

$$x = -.0267, \quad y = -.0395, \quad u = 1.1369$$

$$p = s = .9932, \quad q = -r = -.1165$$

wurden bei allen 100 Startwerten erzielt und stimmten mit denen in [8] überein.

Beispiel 2: Die Daten ($m = 10$) wurden willkürlich gewählt:

a_i	b_i	c_i	d_i
2	3	9	6
2	-3	2	-6
-4	-5	-2	3
6	5	5	-8
9	-8	-7	0
-1	8	7	-3
0	-5	-7	3
3	5	-4	4
-3	0	0	-4
-3	-4	2	-7

Die Ergebnisse

$$x = .7958, \quad y = -.7827, \quad u = .4369$$

$$p = s = -.2649, \quad q = -r = .9643$$

wurden für 85 von 100 Startwerten erzielt. Es gab also 15 Startwerte ohne Konvergenz oder mit Abbruch des Verfahrens aus anderen Gründen [4].



Literatur

- [1] CASPARY, W.; BEINEKE, D.: Robuste HELMERT-Transformation, AVN 7/2003, 242–247
- [2] KOCH, K.-R.; FRÖHLICH, H.; BRÖKER, G.: Transformation räumlicher variabler Koordinaten, AVN 8–9/2000, 293–295
- [3] KOCH, K.-R.: Räumliche HELMERT-Transformation variabler Koordinaten im GAUSS-HELMERT- und im GAUSS-MARKOFF-Modell, zfv 3/2002, 147–152
- [4] SPÄTH, H.: Algorithmen für multivariable Ausgleichsmo-
delle. R. Oldenbourg Verlag, München 1974
- [5] SPÄTH, H.: Identifying spatial point sets, Math. Commun. 8 (2003), 69–75
- [6] SPÄTH, H.: Zur Identifizierung von Lochmustern, Techn. Messen 70 (2003), 38–41
- [7] SPÄTH, H.: A Numerical Method for Determining the Spatial HELMERT-Transformation in the Case of Different Scale Factors, zfv 4/2004, 255–257
- [8] SPÄTH, H.: Zur numerischen Berechnung der ebenen HELMERT-Transformation bei unterschiedlichen Maßstabsfaktoren, AVN 7/2004, 270–272
- [9] SPÄTH, H.: Fitting affine and orthogonal transformation between two sets of points, Math. Commun. 9 (2004), 27–34
- [10] WATSON, G. A.: Computing HELMERT transformation, J. Comput. Appl. Math. 197, 387–394 (2006)
- [11] AL-SUBAIHI; WATSON, G. A.: An algorithm for matching point sets using the l_1 norm, Numerical Algorithms 41, 203–217 (2006)

Anschrift des Verfassers:

Prof. Dr. HELMUTH SPÄTH, Fakultät V, Institut für
Mathematik, Carl von Ossietzky Universität Oldenburg,
Postfach 25 03, D-26111 Oldenburg, Germany
e-mail: spaeth@mathematik.uni-oldenburg.de