

# Ingenieurgeodäsie

## Übung12: Regression zur Höhenberechnung



Ausarbeitung im Studiengang  
**Geodäsie und Geoinformatik**  
an der Universität Stuttgart

Ziqing Yu, 3218051

Stuttgart, Mai 2020

---

**Betreuer:** Dipl.-Ing. Otto Lerke  
Universität Stuttgart

# Inhaltsverzeichnis

1.1	Einleitung . . . . .	2
1.2	Aufgabe . . . . .	3
1.2.1	a . . . . .	3
1.2.2	b . . . . .	4
1.2.3	c . . . . .	5
1.2.4	d . . . . .	6
1.2.5	e . . . . .	7
1.2.6	f . . . . .	7
1.3	MatLab Code . . . . .	8

# Kapitel 1

## 1.1 Einleitung

Das amtliche Höhensystem in Deutschland basiert auf Normalhöhen  $H_N$ . Bezugsfläche dieses Höhensystems ist das Quasigeoid. In dieser Übung sind 30 Festpunkten mit Ellipsoidische Höhen gegeben, 20 davon haben bekannte Normalhöhen. Die übrige Normalhöhen sind angefragt.

## 1.2 Aufgabe

### 1.2.1 a

Höhenanomalie

$$\zeta = h - H_N$$

wobei

- $h$ : ellipsoidische Höhe
- $H_N$ : Normalhöhe

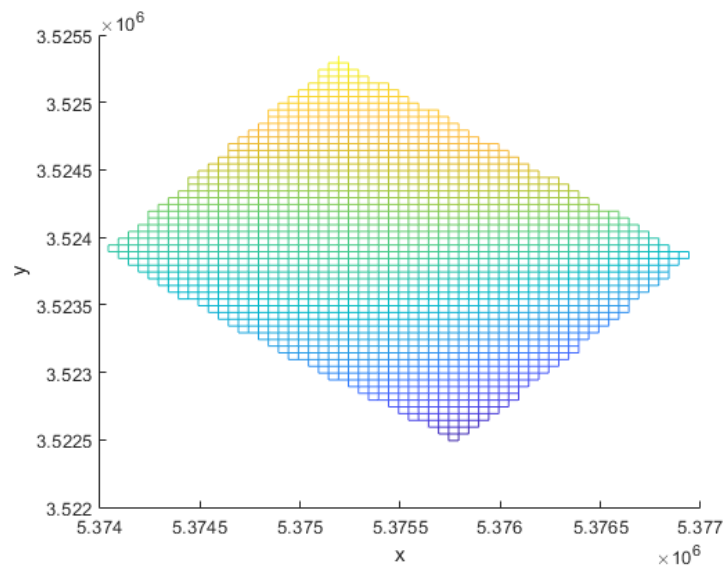
Höhenanomalie von Punkten 1 bis 20:

Pkt.Nr	Höhenanomalie [m]	Pkt.Nr	Höhenanomalie [m]
1	48,3548	11	48,3946
2	48,3928	12	48,4203
3	48,4118	13	48,4420
4	48,4159	14	48,4556
5	48,4290	15	48,4695
6	48,3750	16	48,4148
7	48,4098	17	48,4483
8	48,4290	18	48,4659
9	48,4360	19	48,4762
10	48,4487	20	48,4890

Standardabweichung

$$\sigma_\zeta = \sqrt{(\sigma_h^2 + \sigma_{H_N}^2)} = 0,0051 \text{ m}$$

Graphische Darstellung:



(a) Punkten

## 1.2.2 b

Die Schwerpunkt lautet:

$$x_s = \frac{1}{20} \sum_{i=1}^{20} x_i = 5375436,408 \quad y_s = \frac{1}{20} \sum_{i=1}^{20} y_i = 3523910,288$$



Der funktionale Modell:

$$\begin{aligned}
 x'_i &= x_i - x_s & y'_i &= y_i - y_s \\
 \zeta_i &= a_0 + a_1 \cdot y'_i + a_2 \cdot x'_i + a_3 \cdot y'_i \cdot x'_i + a_4 \cdot y'^2_i + a_5 \cdot x'^2_i \\
 \underbrace{\begin{bmatrix} \zeta_1 \\ \zeta_2 \\ \vdots \\ \zeta_{19} \\ \zeta_{20} \end{bmatrix}}_l &= \underbrace{\begin{bmatrix} 1 & y'_1 & x'_1 & y'_1 \cdot x'_1 & y'^2_1 & x'^2_1 \\ 1 & y'_2 & x'_2 & y'_2 \cdot x'_2 & y'^2_2 & x'^2_2 \\ \vdots & & & & & \\ 1 & y'_{19} & x'_{19} & y'_{19} \cdot x'_{19} & y'^2_{19} & x'^2_{19} \\ 1 & y'_{20} & x'_{20} & y'_{20} \cdot x'_{20} & y'^2_{20} & x'^2_{20} \end{bmatrix}}_A \cdot \underbrace{\begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \end{bmatrix}}_x \\
 \hat{x} &= (A' A)^{-1} A' l = \begin{bmatrix} 48,4361 \\ 4,3398 \cdot 10^{-5} \\ -2,5563 \cdot 10^{-6} \\ 8,1539 \cdot 10^{-9} \\ -4,9588 \cdot 10^{-9} \\ -5,7264 \cdot 10^{-9} \end{bmatrix}
 \end{aligned}$$

## 1.2.3 c

$n$  ist die Anzahl der übrigen Koeffizienten,  $r = 20 - n$   
 Kovarianzmatrix von alle Koeffizienten

$$\Sigma_a = \sigma_\epsilon^2 \cdot (A' A)^{-1}$$

Standardabweichung von Koeffizienten:

$$\sigma_a = \sqrt{\text{diag}(\Sigma_a)}$$

Testgröße für eine Koeffizient

$$\frac{|a_i - 0|}{\sigma_{a_i}}$$

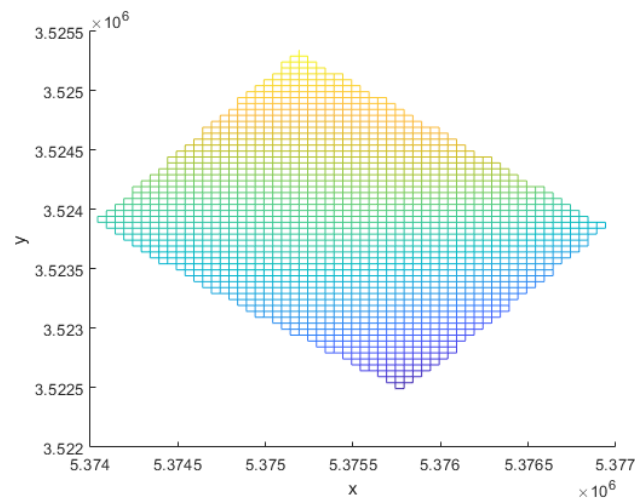
Quantil ist  $t_{97,5,r}^t$  in t-Verteilung.  
 Testgröße in jeder Schleife:

	Quantil	$T_{a_0}$	$T_{a_1}$	$T_{a_2}$	$T_{a_3}$	$T_{a_4}$	$T_{a_5}$
1	3.0688	0.0552	$6,657 \cdot 10^{-11}$	$3,645 \cdot 10^{-12}$	$2,472 \cdot 10^{-17}$	$6,628 \cdot 10^{-18}$	$7,002 \cdot 10^{-18}$
2	2.9467	$3,010 \cdot 10^4$	27,6830	1,8578	3,5185		2,0606
3	2,8131	$3,020 \cdot 10^4$	28,2715		3,7971		2,2270
4	2,6550	$4,105 \cdot 10^4$	28,5474		3,7364		

Koeffizienten in jeder Schleife:

	$a_0$	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$
1	48,4361	$4,3398 \cdot 10^{-5}$	$-2,5563 \cdot 10^{-6}$	$8,1539 \cdot 10^{-9}$	$-4,9588 \cdot 10^{-9}$	$-5,7264 \cdot 10^{-9}$
2	48,4323	$4,3178 \cdot 10^{-5}$	$-2,7177 \cdot 10^{-6}$	$1,1162 \cdot 10^{-8}$		$-3,4860 \cdot 10^{-9}$
3	48,4325	$4,3608 \cdot 10^{-5}$		$1,1940 \cdot 10^{-8}$		$-3,7536 \cdot 10^{-9}$
4	48,4301	$4,3887 \cdot 10^{-5}$		$1,1745 \cdot 10^{-8}$		





(a) Höhenanomalie

### 1.2.4 d

Wir haben jetzt das neue Flächenpolynom:

$$\zeta_i = a_0 + a_1 \cdot y_i + a_3 \cdot y_i \cdot x_i$$



Die Koordinaten der Neupunkten sind bekannt, damit darf man die Höhenanomalie berechnen.

Pkt.Nr	Höhenanomalie [ m]
21	48,4340
22	48,4345
23	48,4253
24	48,4258
25	48,4316
26	48,4429
27	48,4465
28	48,4425
29	48,4337
30	48,4226

Die ellipsoidesche Höhen sind bekannt, dann:



Pkt.Nr	Normalhöhe [ m]
21	800,0020
22	791,3315
23	614,9617
24	570,9782
25	504,6484
26	495,1981
27	466,7145
28	441,3225
29	412,4743
30	376,8034

## 1.2.5 e



Standardabweichung der Neupunktenormalhöhen sind durch Fehlerfortpflanzung bestimmbar.

$$\Sigma_{neu} = F \cdot \Sigma_{ll} \cdot F'$$

wobei:

$$F = \underbrace{\begin{bmatrix} 1 & 0 & 0 & \cdots & 0 & 0 & y_{21} & y_{21} \cdot x_{21} & y_{21}^2 & x_{21}^2 \\ 0 & 1 & 0 & \cdots & 0 & 0 & y_{22} & y_{22} \cdot x_{22} & y_{22}^2 & x_{22}^2 \\ 0 & 0 & 1 & \cdots & 0 & 0 & y_{23} & y_{23} \cdot x_{23} & y_{23}^2 & x_{23}^2 \\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 0 & y_{29} & y_{29} \cdot x_{29} & y_{29}^2 & x_{29}^2 \\ 0 & 0 & 0 & \cdots & 0 & 1 & y_{30} & y_{30} \cdot x_{30} & y_{30}^2 & x_{30}^2 \end{bmatrix}}_{10 \times 14}$$

Die Standardabweichungen sind:

Pkt.Nr	Standardabweichung [ mm]
21	5,1368
22	5,1329
23	5,1541
24	5,1437
25	5,1377
26	5,1553
27	5,2348
28	5,2199
29	5,1554
30	5,1358

## 1.2.6 f



Diese Standardabweichungen sind geringer als die Standardabweichung der Nivellement. aber die berechneten Standardabweichungen beziehen sich auf die GPS Messung, d.h. sie sind in sinnvoll, wo Nivellement eingeschränkt sind.



## 1.3 MatLab Code

Code und Data Auch als Anhang in E-mail.

```

%% Import Data
load data.mat

%% Aufgabe a
zeta_1 = data(1:20,4) - data(1:20,3); % öHhenanomalie 1 - 20
sigma_HN = 0.001; % Standardabweichung öNormalhhen
sigma_e = 0.005; % Standardabweichung Ellipsolid öHhe
sigma_zeta = sqrt(sigma_HN^2 + sigma_e^2); %
    Fehlerfortpflanzung
figure
scatter3(data(1:20,2),data(1:20,1),zeta_1)

%% Aufgabe b
A_1 = [ones(20,1), data(1:20,1), data(1:20,2), data(1:20,1).*
    data(1:20,2), data(1:20,1).^2, data(1:20,2).^2]; %
    Matrix bauen
a_bar = (A_1' * A_1) \ A_1' * zeta_1; % Ausgleichen

%% Aufgabe c
r = 20 - length(a_bar);
Sigma_a = sigma_zeta^2 * inv(A_1' * A_1);
sigma_a = sqrt(diag(Sigma_a));
T = abs(a_bar - 0) ./ sigma_a;
Q = abs(tinv(1 - 0.975, r)); % Quantil
idx = find(T < Q);
a_list = cell(6,1);
T_list = cell(6,1);
%%
i = 1;
id = zeros(6,1) * NaN;
check = zeros(6,1) * NaN;
check_list = 1:6;
while ~isempty(idx)
    a_list{i} = a_bar;
    T_list{i} = T;
    id(i) = find(T == min(T));
    check(i) = check_list(id(i));
    check_list(id(i)) = [];
    A_1(:,id(i)) = [];
    a_bar = (A_1' * A_1) \ A_1' * zeta_1;

    r = 20 - length(a_bar);
    Sigma_a = sigma_zeta^2 * inv(A_1' * A_1);
    sigma_a = sqrt(diag(Sigma_a));

```

```

T = abs(a_bar - 0) ./ sigma_a;
Q = abs(tinv(1 - 0.975, r));
idx = find(T < Q);
i = i + 1;
end
xq = min(data(1:20,2)):50:max(data(1:20,2));
yq = min(data(1:20,1)):50:max(data(1:20,1));
[xq,yq] = meshgrid(xq,yq);
vq = griddata(data(1:20,2),data(1:20,1),zeta_1,xq,yq);
figure,hold on
mesh(xq,yq,vq)
xlabel("x")
ylabel("y")
zlabel("öHhenanomalie")

%% Aufgabe d
A_2 = [ones(10,1), data(21:30,1), data(21:30,2), data
      (21:30,1).* data(21:30,2), data(21:30,1).^2, data(21:30,2)
      .^2];
for i=1:6
  if isnan(id(i))
    break
  else
    A_2(:,id(i)) = [];
  end
end
zeta_2 = A_2 * a_bar; % öHhenanomalie 21 - 30
NH_under = data(21:30,4) - zeta_2; % öNormalhhen 21 - 30
data(21:30,3) = NH_under;

%% Aufgabe e
F = [eye(10),A_2];
[~,l] = size(F);
Sigma_big = zeros(l,l);
Sigma_big(1:10,1:10) = 0.005^2 * eye(10);
Sigma_big(11:l,11:l) = Sigma_a;
Sigma_nh = F * Sigma_big * F';
sigma_nh = sqrt(diag(Sigma_nh));

```