```matlab
%-----------------------------------------------------------------------------
%
%           ADJUSTMENT THEORY I
%     Exercise 7: Adjustment Calculation - part II
%
%    Author         : Anastasia Pasioti
%    Version        : October 09, 2018
%    Last changes   : December 14, 2022
%
%-----------------------------------------------------------------------------
% Group 6
%
% Fahd Sbahi / 0501211
% Nils Lang / 0494447
% Felix Nöding / 0501313
% Egor Grekhov / 0501111
% Corban Rosenauer / 0494441
% Zilong Liu / 0501297
%-----------------------------------------------------------------------------

clc;
clear all;
close all;

%-----------------------------------------------------------------------------
%    Task 1: Adjustment of a straight line
%-----------------------------------------------------------------------------
x1 = 1;
x2 = 2;
x3 = 3;
x4 = 4;

y1 = 0.1;
y2 = 1.1;
y3 = 1.8;
y4 = 2.4;

s1 = 2;
s2 = 1;
s3 = 4;
s4 = 2;

%-----------------------------------------------------------------------------
%    Observations and redundancy
%-----------------------------------------------------------------------------
%Vector of observations
L = [y1; y2; y3; y4]; % or just L = [y1 y2 y3 y4]';

x = [x1 x2 x3 x4]';
```

```
%Number of observations
no_n = length(L);

%Number of unknowns
no_u = 2;

%Redundancy
r = no_n - no_u;

%--------------------------------------------------------------------------
%  Stochastic model
%--------------------------------------------------------------------------
%VC Matrix of the observations
s_L = [s1 s2 s3 s4] / 100; % [m]
S_LL = diag(s_L.^2);   %. means elementwise

%Theoretical reference standard deviation
sigma_0 = 1;           %a priori

%Cofactor matrix of the observations
Q_LL = 1 / sigma_0^2*S_LL;

%Weight matrix
P = inv(Q_LL);

%--------------------------------------------------------------------------
%  Adjustment
%--------------------------------------------------------------------------
%Design matrix
A = [x ones(no_n,1)];

%Normal matrix
N = A' * P * A;

%Vector of right hand side of normal equations
n = A' * P * L;

%Inversion of normal matrix / Cofactor matrix of the unknowns
Q_XX = inv(N);

%Solution of normal equation
X_hat = Q_XX * n;

%Estimated unknown parameters
a = X_hat(1);
b = X_hat(2);

%Vector of residuals
v = A * X_hat - L;
```

```
%Objective function
vTPv = v' * P * v;

%Vector of adjusted observations
L_hat = L + v;

%Final check


%Empirical reference standard deviation
s_0 = sqrt(vTPv / r);       %a posteriori

%VC matrix of adjusted unknowns
S_XX_hat = s_0^2 * Q_XX;

%Standard deviation of the adjusted unknowns
s_X = sqrt(diag(S_XX_hat));

%Cofactor matrix of adjusted observations
Q_LL_hat = A * Q_XX * A';

%VC matrix of adjusted observations
S_LL_hat = s_0^2 * Q_LL_hat;

%Standard deviation of the adjusted observations
s_L_hat = sqrt(diag(S_LL_hat));

%Cofactor matrix of the residuals
Q_vv = Q_LL - Q_LL_hat;

%VC matrix of residuals
S_vv = s_0^2 * Q_vv;

%Standard deviation of the residuals
s_v = sqrt(diag(S_vv));

disp('X_hat s_X_hat')
```

```
X_hat s_X_hat
```

```
[X_hat s_X]
```

```
ans = 2×2
    0.7360    0.0824
   -0.4432    0.1957
```

```


disp('v s_v L_hat s_L_hat')
```

```
v s_v L_hat s_L_hat
```

```
[v s_v L_hat s_L_hat]
```

```
ans = 4×4
    0.1928    0.1369    0.2928    0.1233
   -0.0712    0.0528    1.0288    0.0755
   -0.0352    0.3550    1.7648    0.0989
    0.1008    0.0807    2.5008    0.1656
```

```
clc;
clear all;
close all;

%------------------------------------------------------------------------------
%   Task 2: Adjustment of a parabola
%------------------------------------------------------------------------------
x1 = 1;
x2 = 2;
x3 = 3;
x4 = 4;
x5 = 5;

y1 = 1.112;
y2 = 0.880;
y3 = 0.768;
y4 = 0.830;
y5 = 1.175;

s = 0.02;

%------------------------------------------------------------------------------
%   Observations and redundancy
%------------------------------------------------------------------------------
%Vector of observations
L = [y1 y2 y3 y4 y5]';

x = [x1 x2 x3 x4 x5]';

%Number of observations
no_n = length(L);

%Number of unknowns
no_u = 3;

%Redundancy
r = no_n - no_u;

%------------------------------------------------------------------------------
%  Stochastic model
%------------------------------------------------------------------------------
%VC Matrix of the observations
s_L = [s s s s s];
S_LL = diag(s_L.^2);

%Theoretical reference standard deviation
sigma_0 = 1;        %a priori

%Cofactor matrix of the observations
Q_LL = 1 / sigma_0^2*S_LL;
```

```
%Weight matrix
P = inv(Q_LL);

%--------------------------------------------------------------------------
%  Adjustment
%--------------------------------------------------------------------------
%Design matrix
A = [x.^2 x ones(no_n,1)];

%Normal matrix
N = A' * P * A;

%Vector of right hand side of normal equations
n = A' * P * L;

%Inversion of normal matrix / Cofactor matrix of the unknowns
Q_XX = inv(N);

%Solution of normal equation
X_hat = Q_XX * n;

%Estimated unknown parameters
% a = X_hat(1);
% b = X_hat(2);
% c = X_hat(3);

%Vector of residuals
v = A * X_hat - L;

%Objective function
vTPv = v' * P * v;

%Vector of adjusted observations
L_hat = L + v;

%Final check

%Empirical reference standard deviation
s_0 = sqrt(vTPv / r);     %a posteriori

%VC matrix of adjusted unknowns
S_XX_hat = s_0^2 * Q_XX;

%Standard deviation of the adjusted unknowns
s_X = sqrt(diag(S_XX_hat));

%Cofactor matrix of adjusted observations
Q_LL_hat = A * Q_XX * A';
```

```
%VC matrix of adjusted observations
S_LL_hat = s_0^2 * Q_LL_hat;

%Standard deviation of the adjusted observations
s_L_hat = sqrt(diag(S_LL_hat));

%Cofactor matrix of the residuals
Q_vv = Q_LL - Q_LL_hat;

%VC matrix of residuals
S_vv = s_0^2 * Q_vv;

%Standard deviation of the residuals
s_v = sqrt(diag(S_vv));

disp('X_hat s_X_hat')
```

```
X_hat s_X_hat
```

```
[X_hat s_X]
```

```
ans = 3×2
    0.0949    0.0098
   -0.5615    0.0601
    1.5942    0.0788
```

```
disp('v s_v L_hat s_L_hat')
```

```
v s_v L_hat s_L_hat
```

```
[v s_v L_hat s_L_hat]
```

```
ans = 5×4
    0.0155    0.0124    1.1275    0.0346
   -0.0295    0.0291    0.8505    0.0224
   -0.0047    0.0263    0.7633    0.0256
    0.0357    0.0291    0.8657    0.0224
   -0.0171    0.0124    1.1579    0.0346
```