

# A Appendix

## A.1 Matlab kode

```
% Ryan Yde (knb392), Thorsten Baek (kmg386), Jens Kinch (mqt845)

% fast L varier I

clc; clear all; close all;

f1 = importdata('fast_L_FafI.txt'); % datafilen hentes
L1 = 8*10^(-2); % laengden defineres
I1 = f1.data(:,1); % stroemstyrken defineres
m1 = f1.data(:,2); % massen-aendringen (vaegten) defineres
F1 = m1*9.8*10^(-3); % kraefte defineres
X = I1*L1; % x-aksen defineres

sigma_y_0 = 0.001*9.8*10^(-3); % usikkerhed paa vaegten
w = 1/sigma_y_0^2; % vaegtingen af datapunkterne - til fit
Delta = length(X)*w * sum(w.*X.^2) - ( sum( w.*X) )^2; % bruges i fit
A = ( sum(w.*X.^2) * sum(w.*F1) - sum(w.*X) * sum(w.*X.*F1) ) / Delta; % fit'tets skaering med y-aksen
B = ( length(X)*w *sum(w.*X.*F1) - sum(w.*X) * sum(w.*F1) ) / Delta % fit'tets haeldning
sigma_B = sqrt ( length(X)*w / Delta ) % usikkerhed paa fit'tets haeldning

fprintf('B=%d\n',B) % den foerste haeldningen vises
for i=1:20 % fittet gennemkoeres 20 gange ( det er rigeligt )
sigma_x_to_y = B*sqrt( (I1*0.001).^2 + (0.001*L1).^2 +(I1*L1*sin(pi/180)).^2 ); % usikkerheden paa x-aksen propageres til y-aksen
sigma_y = sqrt( sigma_y_0^2 + sigma_x_to_y.^2 ); % den 'nye' usikkerhed paa y-aksen
w = 1./sigma_y.^2; % den 'nye' vaegtning
Delta = sum(w) * sum(w.*X.^2) - ( sum( w.*X) )^2; % den 'nye' Delta
A = ( sum(w.*X.^2) * sum(w.*F1) - sum(w.*X) * sum(w.*X.*F1) ) / Delta; % den 'nye' skaerping med y-aksen
B = ( sum(w) *sum(w.*X.*F1) - sum(w.*X) * sum(w.*F1) ) / Delta; % den 'nye' haeldning
fprintf('B=%d\n',B) % den 'nye' haeldning vises
end

errorbar(X,F1,sigma_y,'o','Color',[0,0,0]) % usikkerheder paa hvert punkt plottes
title('Variabel: I') % plottets titel
xlabel('I*L [C*m/s]') % x-aksens label
ylabel('F [N]') % y-aksens label

xakse = 0.05:0.01:0.35; % x-aksens punkter til fittet
yakse = A + B.*xakse; % x-aksens vaerdier til fittet
hold on
plot(xakse,yakse) % fittet plottes

sigma_B = sqrt ( sum(w) / Delta ) % del endelige usikkerhed paa fittets haeldning vises

chi_sqrt = sum ( (F1 - (A + B .* X)).^2 ./ ( sigma_y .^2 ) ) % chi^2 for fittet vises
chi_sqrt_reduced = chi_sqrt / (length(X)-2) % reduceret chi^2 for fittet vises
chi_sqrt_prob = chi2cdf(chi_sqrt,length(X)-2) % chi^2 sandsynlighed vises

% de samme operationer udfoeres paa de foelgende delforsoeg, saa kommentarer
% udelades her.

%% fast I varier L

clc; clear all; close all;
I = 3.96;
f2 = importdata('fast_I_FafL.txt');
L1 = f2.data(:,1)*1e-2;
m2 = f2.data(:,2);
F1 = m2*9.8*10^(-3);
X = I*L1;

sigma_y_0 = 0.001*9.8*10^(-3);
w = 1/sigma_y_0^2;
Delta = length(X)*w * sum(w.*X.^2) - ( sum( w.*X) )^2;
A = ( sum(w.*X.^2) * sum(w.*F1) - sum(w.*X) * sum(w.*X.*F1) ) / Delta;
B = ( length(X)*w *sum(w.*X.*F1) - sum(w.*X) * sum(w.*F1) ) / Delta
sigma_B = sqrt ( length(X)*w / Delta )

fprintf('B=%d\n',B)
for i=1:20
sigma_x_to_y = B*sqrt( (I*0.001).^2 + (0.005*L1).^2 + (I*L1*sin(pi/180)).^2 );
sigma_y = sqrt( sigma_y_0^2 + sigma_x_to_y.^2 );
w = 1./sigma_y.^2;
Delta = sum(w) * sum(w.*X.^2) - ( sum( w.*X) )^2;
A = ( sum(w.*X.^2) * sum(w.*F1) - sum(w.*X) * sum(w.*X.*F1) ) / Delta;
```

```

B = ( sum(w) *sum(w.*X.*F1) - sum(w.*X) * sum(w.*F1) ) / Delta;
fprintf('B=%d\n',B)
end

errorbar(X,F1,sigma_y,'o','Color',[0,0,0])
title('Variabel: L')
xlabel('I*L [C*m/s]')
ylabel('F [N]')

xakse = 0:0.01:0.35;
yakse = A + B.*xakse;
hold on
plot(xakse,yakse)
sigma_B = sqrt ( sum(w) / Delta )

chi_sqrt = sum ( (F1 - (A + B .* X)).^2 ./ ( sigma_y .^2 ) )
chi_sqrt_reduced = chi_sqrt / (length(X)-2)
chi_sqrt_prob = chi2cdf(chi_sqrt,length(X)-2)

%% Varier vinkel
clc; clear all; close all;

f = importdata('variabelvinkel.txt');
vinkel = f.data(:,1);
theta = -vinkel.*(pi/180);
m = f.data(:,2);
F1 = m.*9.8;
I = 3.86;
L1 = 0.1;
X = sin(theta)*I*L1;

sigma_y_0 = 0.001*9.8*10^(-3);
w = 1/sigma_y_0^2;
Delta = length(X)*w * sum(w.*X.^2) - ( sum( w.*X) )^2;
A = ( sum(w.*X.^2) * sum(w.*F1) - sum(w.*X) * sum(w.*X.*F1) ) / Delta;
B = ( length(X)*w *sum(w.*X.*F1) - sum(w.*X) * sum(w.*F1) ) / Delta
sigma_B = sqrt ( length(X)*w / Delta )

fprintf('B=%d\n',B)
for i=1:20
sigma_x_to_y = B*sqrt( (I*0.001).^2 + (0.005*L1).^2 );
sigma_x_to_y = B*sqrt( (sin(pi/360)*I*L1).^2 + (sin(theta)*0.002*L1).^2 + (sin(theta)*I*0.001).^2 );
sigma_y = sqrt( sigma_y_0^2 + sigma_x_to_y.^2 );
w = 1./sigma_y.^2;
Delta = sum(w) * sum(w.*X.^2) - ( sum( w.*X) )^2;
A = ( sum(w.*X.^2) * sum(w.*F1) - sum(w.*X) * sum(w.*X.*F1) ) / Delta;
B = ( sum(w) *sum(w.*X.*F1) - sum(w.*X) * sum(w.*F1) ) / Delta;
fprintf('B=%d\n',B)
end

errorbar(X,F1,sigma_y,'o','Color',[0,0,0])
title('variabel: \theta')
xlabel('sin(\theta)*I*L [C*m/s]')
ylabel('F [N]')

xakse = -0.4:0.01:0.4;
yakse = A + B.*xakse;
hold on
plot(xakse,yakse)

sigma_B = sqrt ( sum(w) / Delta )
chi_sqrt = sum ( (F1 - (A + B .* X)).^2 ./ ( sigma_y .^2 ) )
chi_sqrt_reduced = chi_sqrt / (length(X)-2)
chi_sqrt_prob = chi2cdf(chi_sqrt,length(X)-2)

%% varier B (antal magnet)
clc; clear all; close all;

f = importdata('variabelB.txt');
magnet = f.data(:,1);
m = f.data(:,2);
F1 = m.*9.8*10^(-3);
I = 3.873;
L1 = 0.08;
X = magnet;

plot(X,F1,'o')

sigma_y_0 = 0.001*9.8*10^(-3);
w = 1/sigma_y_0^2;
Delta = length(X)*w * sum(w.*X.^2) - ( sum( w.*X) )^2 ;

```

```

A = ( sum(w.*X.^2) * sum(w.*F1) - sum(w.*X) * sum(w.*X.*F1) ) / Delta;
B = ( length(X)*w *sum(w.*X.*F1) - sum(w.*X) * sum(w.*F1) ) / Delta
sigma_B = sqrt ( length(X)*w / Delta )

fprintf('B=%d\n',B)
for i=1:20
sigma_x_to_y = B*sqrt( (sin(pi/180).*X).^2 );
sigma_y = sqrt( sigma_y_0^2 + sigma_x_to_y.^2 );
w = 1./sigma_y.^2;
Delta = sum(w) * sum(w.*X.^2) - ( sum( w.*X) )^2;
A = ( sum(w.*X.^2) * sum(w.*F1) - sum(w.*X) * sum(w.*X.*F1) ) / Delta;
B = ( sum(w) *sum(w.*X.*F1) - sum(w.*X) * sum(w.*F1) ) / Delta;
fprintf('B=%d\n',B)
end

% sigma_y = [sigma_y sigma_y sigma_y sigma_y sigma_y sigma_y];

errorbar(X,F1,sigma_y,'o','Color',[0,0,0])
title('variabel: B')
xlabel('B (antal magneter)')
ylabel('F [N]')

xakse = 0:0.1:7;
yakse = A + B.*xakse;
hold on
plot(xakse,yakse)

sigma_B = sqrt ( sum(w) / Delta )

chi_sqrt = sum ( (F1 - (A + B .* X)).^2 ./ ( sigma_y .^2 ) )
chi_sqrt_reduced = chi_sqrt / (length(X)-2)
chi_sqrt_prob = chi2cdf(chi_sqrt,length(X)-2)

```