Table of Contents


```
clear all;
close all;
clc
```

Read in Data

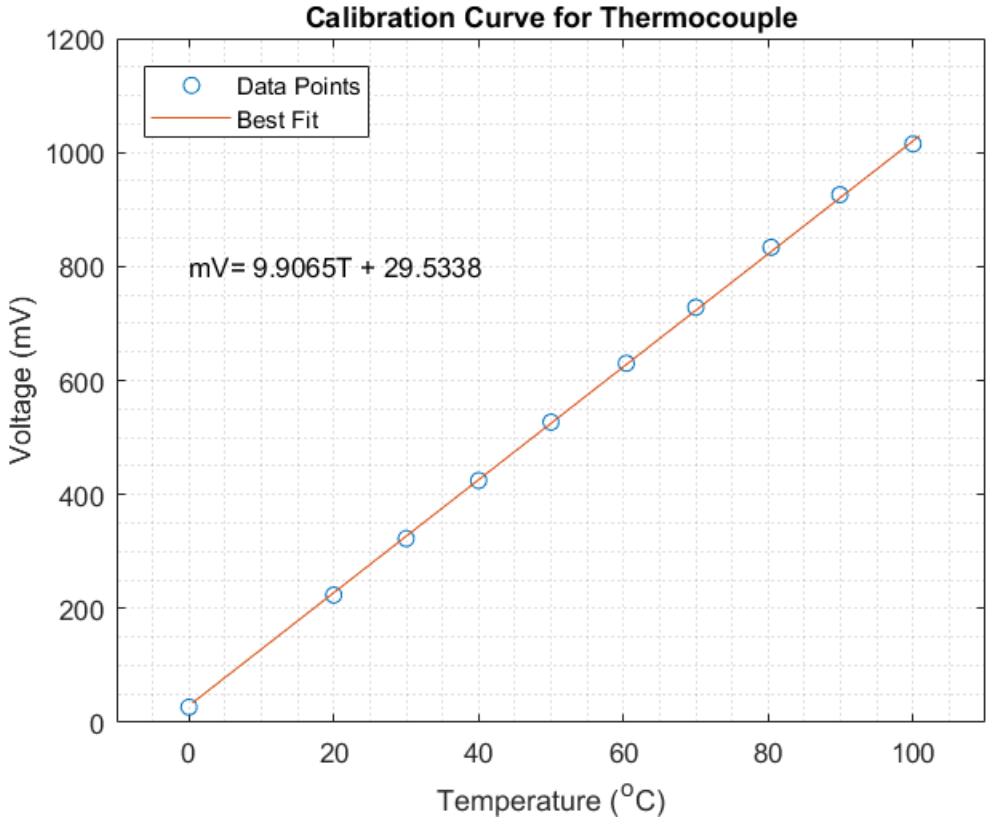
```
EIB_SS_time=xlsread('Lab_2_Data.xlsx', 'EIB', 'A9:A15000');%Embed Ice
 Boil
EIB_SS_volt=xlsread('Lab_2_Data.xlsx', 'EIB', 'B9:B15000');
EBI_SS_time=xlsread('Lab_2_Data_Cont.xlsx', 'EBI', 'A9:A150000');%Embed
 Boil Ice
EBI_SS_volt=xlsread('Lab_2_Data_Cont.xlsx', 'EBI', 'B9:B150000');
EIB_AL_time=xlsread('Lab_2_Data_Cont_2.xlsx', 'EIB_AL', 'A9:A150000');
EIB_AL_volt=xlsread('Lab_2_Data_Cont_2.xlsx', 'EIB_AL', 'B9:B150000');
EBI_AL_time=xlsread('Lab_2_Data_Cont_2.xlsx', 'EBI_AL', 'A9:A150000');
EBI_AL_volt=xlsread('Lab_2_Data_Cont_2.xlsx', 'EBI_AL', 'B9:B150000');
BIR_time=xlsread('Lab_2_Data.xlsx', 'BIR', 'A9:A150000');%Bare Ice
 Room
BIR_volt=xlsread('Lab_2_Data.xlsx', 'BIR', 'B9:B150000');
BIA_time=xlsread('Lab_2_Data.xlsx', 'BIA', 'A9:A150000');%Bare Ice Air
BIA_volt=xlsread('Lab_2_Data.xlsx', 'BIA', 'B9:B150000');
BBI_time=xlsread('Lab_2_Data.xlsx', 'BBI', 'A9:A150000');%Bare Boil
 Ice
BBI_volt=xlsread('Lab_2_Data.xlsx', 'BBI', 'B9:B150000');
BIB_time=xlsread('Lab_2_Data.xlsx', 'BIB', 'A9:A150000');%Bare Ice
 Boil
BIB_volt=xlsread('Lab_2_Data.xlsx', 'BIB', 'B9:B150000');
```

Hand Recorded Data

```
Temp_Zero=[0,0,0,0,0,0,0,0,0,0];%Array of 0's
Resist_Zero=[29.37, 29.16, 29.96, 27.93, 26.53, 28.98, 35.35, 27.88,
 28.44, 28.85];%Resistances from 0 bath
Volt_Zero=[15.35, 27.17, 28.56, 28.54, 28.03, 28.05, 27.67, 29.39,
 30.26, 27.86];%Voltage of 0 Bath
resist_Zero_avg=mean(Resist_Zero);
volt_Zero_avg=mean(Volt_Zero);
Temp_LED=[0,20,30,40,50,60.4,70,80.4,89.9,100];%LED temp measurement
Resist_thermistor=[resist_Zero_avg, 12.22, 8.13, 5.82, 3.88, 2.79,
 2.05, 1.47, 1.13, .854];%Thermistor Reistance in KOhm
Voltage_thermcouple=[volt_Zero_avg, 223.57, 322.76, 424.45, 526.79,
 630.10, 728.29, 833.49, 925.93, 1015];%Thermocouple voltages'
```

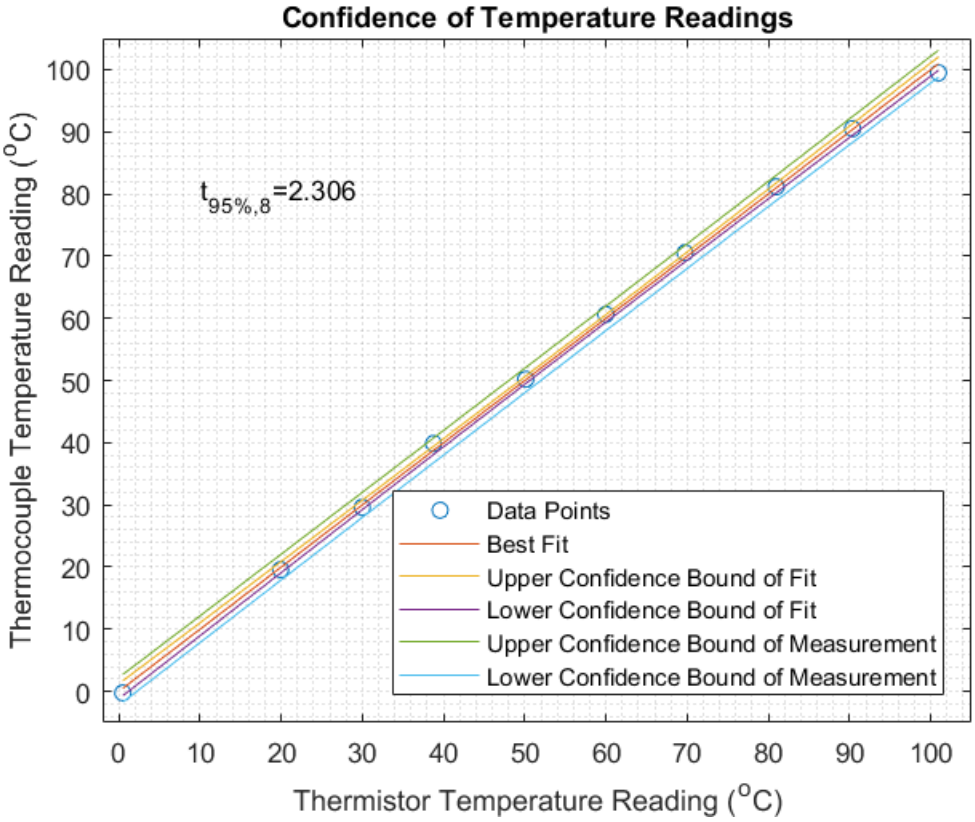
Part 3: Calibration Curve (Voltage)

```
plot(Temp_LED, Voltage_thermcouple, 'o')
hold on
    beta=3601;
    Ro=9915.06;
    To=298.15;
ylabel('Voltage (mV)')
xlabel('Temperature (^oC)')

for i=1:length(Temp_LED)

    Temp_Thermistor(i)=(((1/beta)*log((Resist_thermistor(i)*10^3)/
Ro)+(1/To))^-1)-273.15;
end
p=polyfit(Temp_Thermistor, Voltage_thermcouple, 1);
fit_1=p(1).*Temp_Thermistor+p(2);
plot(Temp_Thermistor, fit_1)
grid minor
text(0,800, ['mV= ' num2str(p(1)) 'T + ' num2str(p(2))])
axis([-10, 110, 0, 1200])
legend('Data Points', 'Best Fit', 'Location', 'Northwest')
title('Calibration Curve for Thermocouple')
figure
```


Calibration Curve for Thermocouple
Data Points
Best Fit
mV= 9.9065T + 29.5338
Voltage (mV)
Temperature (°C)
0
200
400
600
800
1000
1200
0
20
40
60
80
100

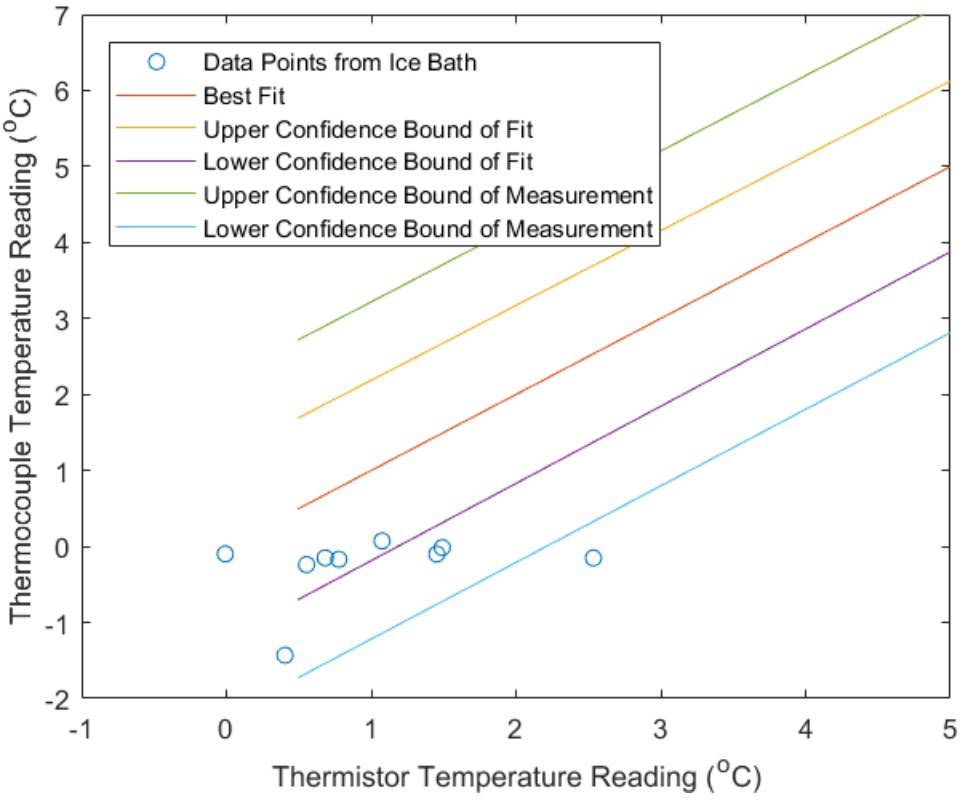
Part 4: Calibration Curve (Resistance)

```
%thermistor is x-direction and thermocouple is y-direction
for i=1:length(Voltage_thermcouple)
    Temp_Thermocouple(i)=(Voltage_thermcouple(i)-p(2))/p(1);

end
plot(Temp_Thermistor,Temp_Thermocouple, 'o')
hold on
xlabel('Thermistor Temperature Reading (^oC)')
ylabel('Thermocouple Temperature Reading (^oC)')
% axis([-10,110,-10,110]);
grid minor
p2=polyfit(Temp_Thermistor,Temp_Thermocouple,1);
fit_2=p2(1).*Temp_Thermistor+p2(2);

plot(Temp_Thermistor, fit_2)
hold on
legend('Data Points', 'Best Fit', 'Location', 'Northwest')
m=1;
nu=length(Temp_Thermistor)-(m+1);
tnup=tinv(.975, nu);
xbar=mean(Temp_Thermistor);
sumbot=sum((Temp_Thermistor-xbar).^2);
syx=(sum((Temp_Thermocouple-fit_2).^2)/nu).^.5;
confit=tnup*syx*(1/length(Temp_Thermistor)+(Temp_Thermistor-xbar).^2/
sumbot).^.5;
conmeas=tnup*syx*(1+1/length(Temp_Thermistor)+(Temp_Thermistor-
xbar).^2/sumbot).^.5;
plot(Temp_Thermistor, fit_2+confit, Temp_Thermistor, fit_2-confit,
 Temp_Thermistor, fit_2+conmeas, Temp_Thermistor, fit_2-conmeas)
text(10, 80,['t_9_5_%_,_8=' num2str(tnup)])
axis([-2,105, -5, 105])
title('Confidence of Temperature Readings')
legend('Data Points', 'Best Fit', 'Upper Confidence Bound of
 Fit', 'Lower Confidence Bound of Fit', 'Upper Confidence Bound
 of Measurement', 'Lower Confidence Bound of Measurement',
  'Location', 'southeast')
figure
```


Confidence of Temperature Readings
$t_{95\%,8}$=2.306
Thermocouple Temperature Reading (^{o}C)
Thermistor Temperature Reading (^{o}C)
Data Points
Best Fit
Upper Confidence Bound of Fit
Lower Confidence Bound of Fit
Upper Confidence Bound of Measurement
Lower Confidence Bound of Measurement
0
10
20
30
40
50
60
70
80
90
100

Part 5: Compare Zero Measurements

```
for i=1:length(Voltage_thermcouple)
    Temp_Thermocouple_zero(i)=(Volt_Zero(i)-p(2))/p(1);
    Temp_Thermistor_zero(i)=(((1/beta)*log((Resist_Zero(i)*10^3)/
Ro)+(1/To))^-1)-273.15;
end
plot(Temp_Thermistor_zero, Temp_Thermocouple_zero, 'o')
hold on
plot(Temp_Thermistor, fit_2)
plot(Temp_Thermistor, fit_2+confit, Temp_Thermistor, fit_2-confit,
 Temp_Thermistor, fit_2+conmeas, Temp_Thermistor, fit_2-conmeas)
legend('Data Points from Ice Bath', 'Best Fit', 'Upper Confidence
 Bound of Fit', 'Lower Confidence Bound of Fit', 'Upper Confidence
 Bound of Measurement', 'Lower Confidence Bound of Measurement',
  'Location', 'northwest')
axis([-1, 5, -2, 7])
xlabel('Thermistor Temperature Reading (^oC)')
ylabel('Thermocouple Temperature Reading (^oC)')
figure
```



Word Doc Stuff

thermocouple_measure_zero=(Volt_Zero/193.4); x=[0,10]; y=[3.1,101]; for i=1:length(Temp_Zero) thermocouple_measure_zero(i)=interp1(y,x,Volt_Zero(i)); end mean_thermocouple_measure=mean(thermocouple_measure_zero) stand_mean=std(thermocouple_measure_zero)/sqrt(length(Resist_Zero)) x_prime_pos=mean(thermocouple_measure_zero)+2.262*stand_mean x_prime_neg=mean(thermocouple_measure_zero)-2.262*stand_mean

Enter Amplifier Tables to Convert mV to C

```
x=[-94, 3.1, 101, 200, 250, 300, 401, 503, 606, 813, 1022, 1233];
y=[-10, 0, 10, 20, 25, 30, 40, 50, 60, 80, 100, 120];
```

BBI

```
for i=1:length(BBI_volt)
    BBI_Temp(i)=interp1(x,y,(BBI_volt(i)*10^3));
end
%smooth data to make anaylsis easier
span=45;
window=ones(span,1)/span;
BBI_Temp_smooth=conv(BBI_Temp,window,'same');

% filter off curving abnormalities at beginning
for i=1:length(BBI_time)
```

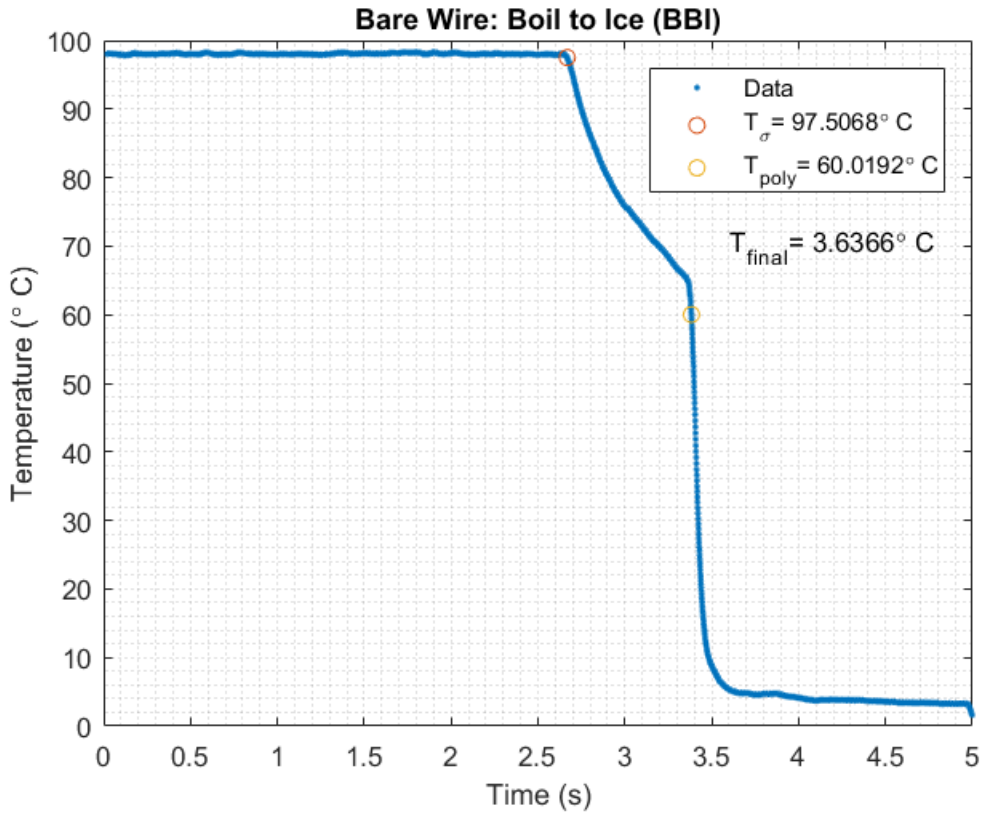
```
    if BBI_Temp_smooth(i)>=96
        break
    end
        if BBI_Temp_smooth(i)<96
            BBI_Temp_smooth(i)=NaN;

    end
end
plot(BBI_time,BBI_Temp_smooth, '.')
hold on
ylabel('Temperature (\circ C)')
xlabel('Time (s)')
title('Bare Wire: Boil to Ice (BBI)')
grid minor
%Find Ti with stand deviation method
deviation_BBI=std(BBI_Temp_smooth(100:2000));
mean_BBI=mean(BBI_Temp_smooth(100:2000));
for i=100:length(BBI_Temp_smooth)
    if BBI_Temp_smooth(i)<mean_BBI-5*deviation_BBI
        position_start_std_BBI=i;
        break
    end
end

time_start_std_BBI=BBI_time(position_start_std_BBI);
Temp_start_std_BBI=BBI_Temp_smooth(position_start_std_BBI);
plot(time_start_std_BBI,Temp_start_std_BBI,'o')

%Find Ti with polyfit method
start_poly_BBI=1;
for i=3300:4500
    BBI_poly(i,:)=polyfit(BBI_time(i:i+50)', BBI_Temp_smooth(i:i+50),
 1);
    if abs(BBI_poly(i))>abs(BBI_poly(start_poly_BBI))
        start_poly_BBI=i;
    end
end


plot(BBI_time(start_poly_BBI)',BBI_Temp_smooth(start_poly_BBI), 'o')

BBI_temp_final=mean(BBI_Temp_smooth(4000:4800));
text(3.6,70,['T_f_i_n_a_l= ' num2str(BBI_temp_final) '\circ C'])
legend('Data', ['T_\sigma= ' num2str(Temp_start_std_BBI) '\circ C'],
 ['T_p_o_l_y= ' num2str(BBI_Temp_smooth(start_poly_BBI)) '\circ C'])
figure
%Find Gamma std
new_time_start_std_BBI=BBI_time-BBI_time(position_start_std_BBI);
subplot(3,1,1)
for i=23:length(new_time_start_std_BBI)
    gamma_std_BBI(i)=(BBI_temp_final-BBI_Temp_smooth(i))/
(BBI_temp_final-Temp_start_std_BBI);
    if gamma_std_BBI(i)<=0
        break
```

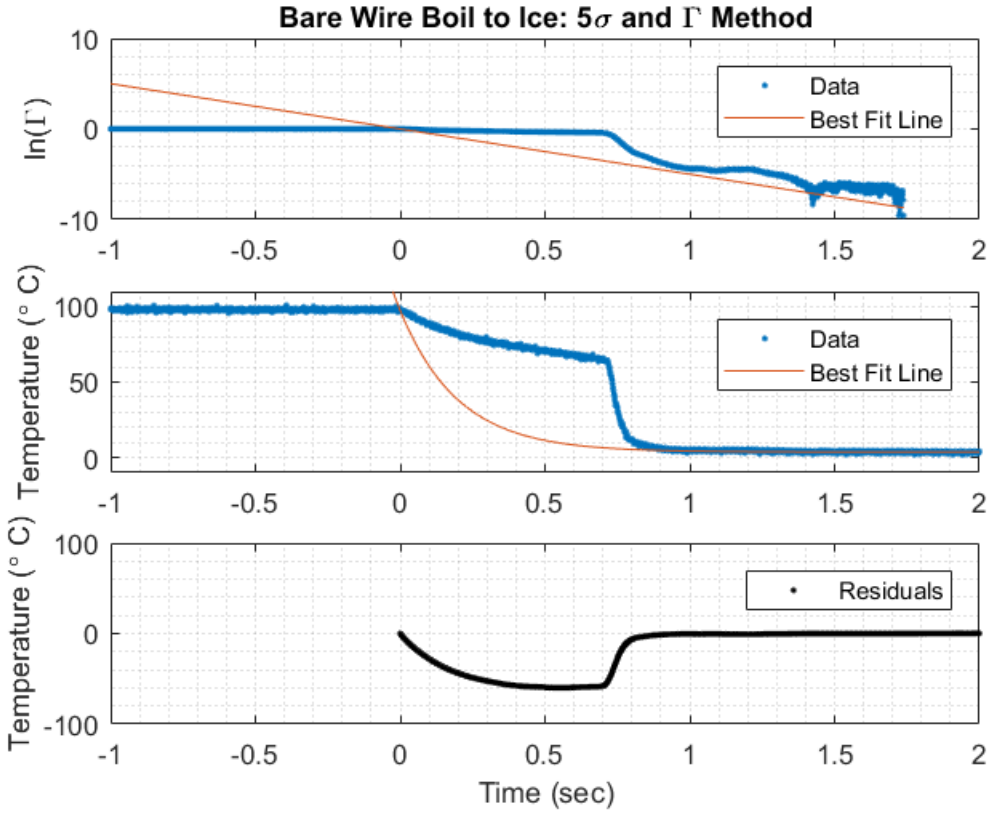
```
    end
    ln_gamma_std_BBI(i)=log(gamma_std_BBI(i));
end
new_time_std_range_BBI=new_time_start_std_BBI(1:length(ln_gamma_std_BBI));
plot(new_time_std_range_BBI, ln_gamma_std_BBI, '.')
hold on
find_tau_std_BBI=polyfit(new_time_std_range_BBI(position_start_std_BBI:length(ln_g
 ln_gamma_std_BBI(position_start_std_BBI:length(ln_gamma_std_BBI)),
 1);% find tau with stdfit
tau_std_BBI=-1/find_tau_std_BBI(1);
fit_tau_std_BBI=find_tau_std_BBI(1).*new_time_std_range_BBI;
plot(new_time_std_range_BBI, fit_tau_std_BBI);

ylabel('ln(\Gamma)')

axis([-1, 2, -10, 10])
% % text(2.8, -8, ['\tau= ' num2str(tau_std_BBI) ' sec'])
grid minor
legend('Data', 'Best Fit Line')
title('Bare Wire Boil to Ice: 5\sigma and \Gamma Method')
subplot(3,1,2)
% Estimate plot with time constant
for i=1:length(BBI_time)
    Temp_tau_std_BBI(i)=BBI_temp_final-((BBI_temp_final-
Temp_start_std_BBI)*exp(-new_time_start_std_BBI(i)/tau_std_BBI));
end




plot(new_time_start_std_BBI, BBI_Temp, '.')
hold on
plot(new_time_start_std_BBI, Temp_tau_std_BBI)


axis([-1, 2, -10, 110])
ylabel('Temperature (\circ C)')
legend('Data', 'Best Fit Line')
grid minor
% Residuals
subplot(3,1,3)

    residual_std_BBI=Temp_tau_std_BBI(position_start_std_BBI:end)-
BBI_Temp_smooth(position_start_std_BBI:end);

plot(new_time_start_std_BBI(position_start_std_BBI:end),
 residual_std_BBI, 'bla.')
grid minor
axis([-1, 2, -100, 100])
ylabel('Temperature (\circ C)')
xlabel('Time (sec)')
legend('Residuals')
figure
```

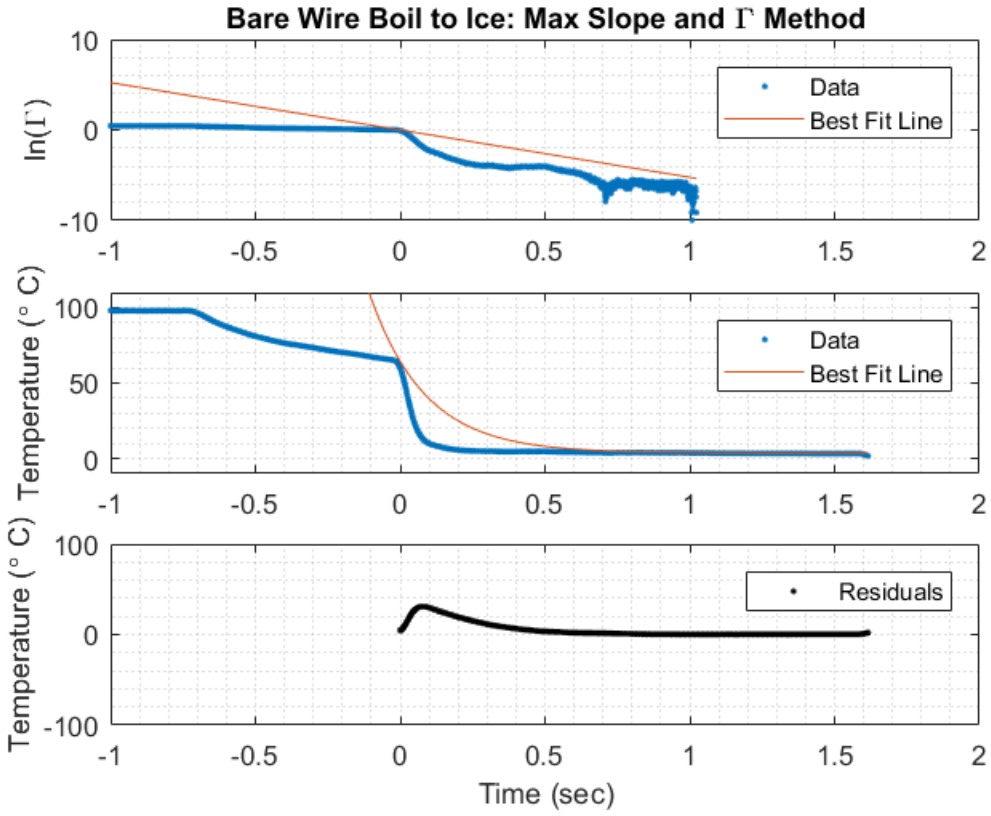
```
% Find Gamma poly
new_time_start_poly_BBI=BBI_time-BBI_time(start_poly_BBI);
subplot(3,1,1)
Temp_start_poly_BBI=(BBI_Temp(start_poly_BBI));
for i=23:length(new_time_start_poly_BBI)
    gamma_poly_BBI(i)=(BBI_temp_final-BBI_Temp_smooth(i))/
(BBI_temp_final-Temp_start_poly_BBI);
    if gamma_poly_BBI(i)<=0
        break
    end
    ln_gamma_poly_BBI(i)=log(gamma_poly_BBI(i));
end
new_time_poly_range_BBI=new_time_start_poly_BBI(1:length(ln_gamma_poly_BBI));
plot(new_time_poly_range_BBI, ln_gamma_poly_BBI, '.')
hold on
find_tau_poly_BBI=polyfit(new_time_poly_range_BBI(start_poly_BBI:length(ln_gamma_p
 ln_gamma_poly_BBI(start_poly_BBI:length(ln_gamma_poly_BBI)) , 1);%
 find tau with polyfit
tau_poly_BBI=-1/find_tau_poly_BBI(1);
fit_tau_poly_BBI=find_tau_poly_BBI(1).*new_time_poly_range_BBI;
plot(new_time_poly_range_BBI, fit_tau_poly_BBI);

ylabel('ln(\Gamma)')

% % text(2.8, -8, ['\tau= ' num2str(tau_poly_BBI) ' sec'])
grid minor
legend('Data', 'Best Fit Line')
title('Bare Wire Boil to Ice: Max Slope and \Gamma Method')
axis([-1, 2, -10, 10])
subplot(3,1,2)
% Estimate plot with time constant
for i=1:length(BBI_time)
    Temp_tau_poly_BBI(i)=BBI_temp_final-((BBI_temp_final-
Temp_start_poly_BBI)*exp(-new_time_start_poly_BBI(i)/tau_poly_BBI));
end




plot(new_time_start_poly_BBI, BBI_Temp_smooth, '.')
hold on
plot(new_time_start_poly_BBI, Temp_tau_poly_BBI)


ylim([-10, 110])
ylabel('Temperature (\circ C)')
axis([-1, 2, -10, 110])
legend('Data', 'Best Fit Line')
grid minor
% Residuals
subplot(3,1,3)

residual_poly_BBI=Temp_tau_poly_BBI(start_poly_BBI:end)-
BBI_Temp_smooth(start_poly_BBI:end);
```

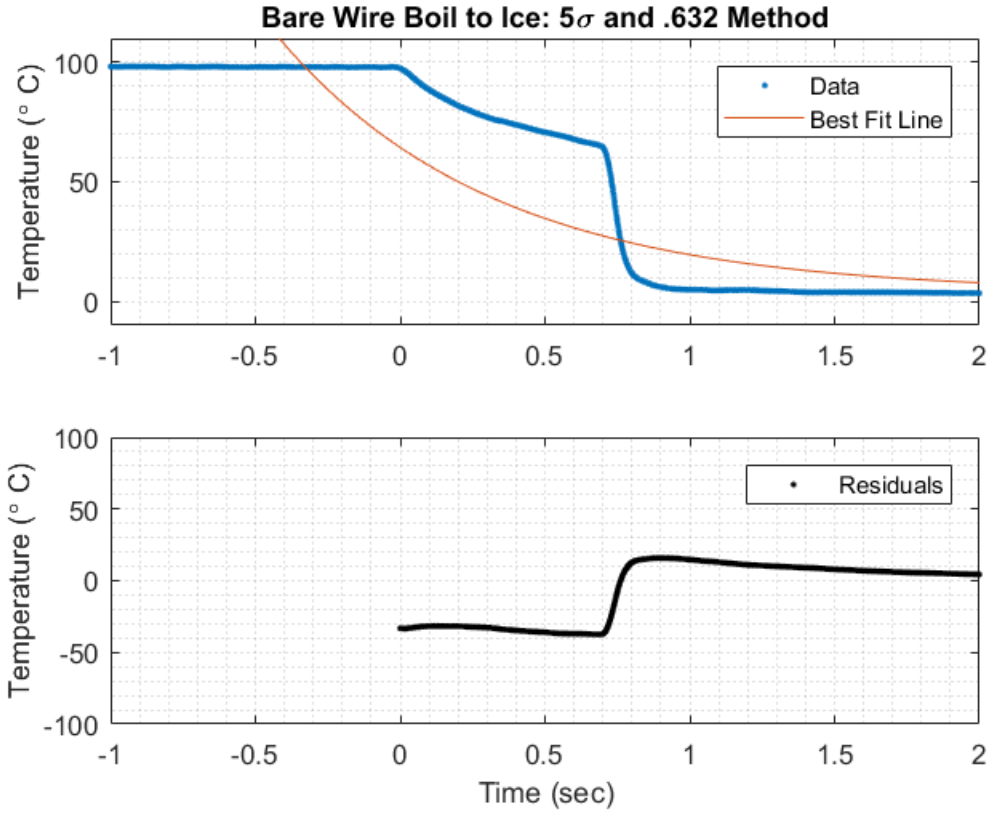
```
plot(new_time_start_poly_BBI(start_poly_BBI:end),
 residual_poly_BBI, 'bla.')
grid minor
axis([-1, 2, -100, 100])
ylabel('Temperature (\circ C)')
legend('Residuals')
xlabel('Time (sec)')
figure
%Calculate Tau with .632       STD
T_at_Tau_std_BBI=Temp_start_std_BBI + .632 * (BBI_temp_final -
 Temp_start_std_BBI);
for i=1:length(new_time_start_std_BBI)
    if BBI_Temp_smooth(i)<T_at_Tau_std_BBI
        Tau_std_value_BBI=new_time_start_std_BBI(i);
        break
    end
end
for i=1:length(new_time_start_std_BBI)
Temp_tau_std_value_BBI(i)=BBI_temp_final-((BBI_temp_final-
Temp_start_poly_BBI)*exp(-new_time_start_std_BBI(i)/
Tau_std_value_BBI));
end
subplot(2,1,1)
plot(new_time_start_std_BBI, BBI_Temp_smooth, '.')
title('Bare Wire Boil to Ice: 5\sigma and .632 Method')
ylabel('Temperature (\circ C)')
grid minor
hold on
plot(new_time_start_std_BBI, Temp_tau_std_value_BBI)
axis([-1, 2, -10, 110])
legend('Data', 'Best Fit Line')
%Residuals
subplot(2,1,2)
residual_value_std_BBI=Temp_tau_std_value_BBI-BBI_Temp_smooth;
plot(new_time_start_std_BBI(position_start_std_BBI:length(residual_value_std_BBI))
ylabel('Temperature (\circ C)')
xlabel('Time (sec)')
grid minor
axis([-1, 2, -100, 100])
legend('Residuals')
figure
T_at_Tau_poly_BBI=Temp_start_poly_BBI + .632 * (BBI_temp_final -
 Temp_start_poly_BBI);
for i=1:length(new_time_start_poly_BBI)
    if BBI_Temp_smooth(i)<T_at_Tau_poly_BBI
        Tau_poly_value_BBI=new_time_start_poly_BBI(i);
        break
    end
end
for i=1:length(new_time_start_poly_BBI)
Temp_tau_poly_value_BBI(i)=BBI_temp_final-((BBI_temp_final-
Temp_start_poly_BBI)*exp(-new_time_start_poly_BBI(i)/
Tau_poly_value_BBI));
```

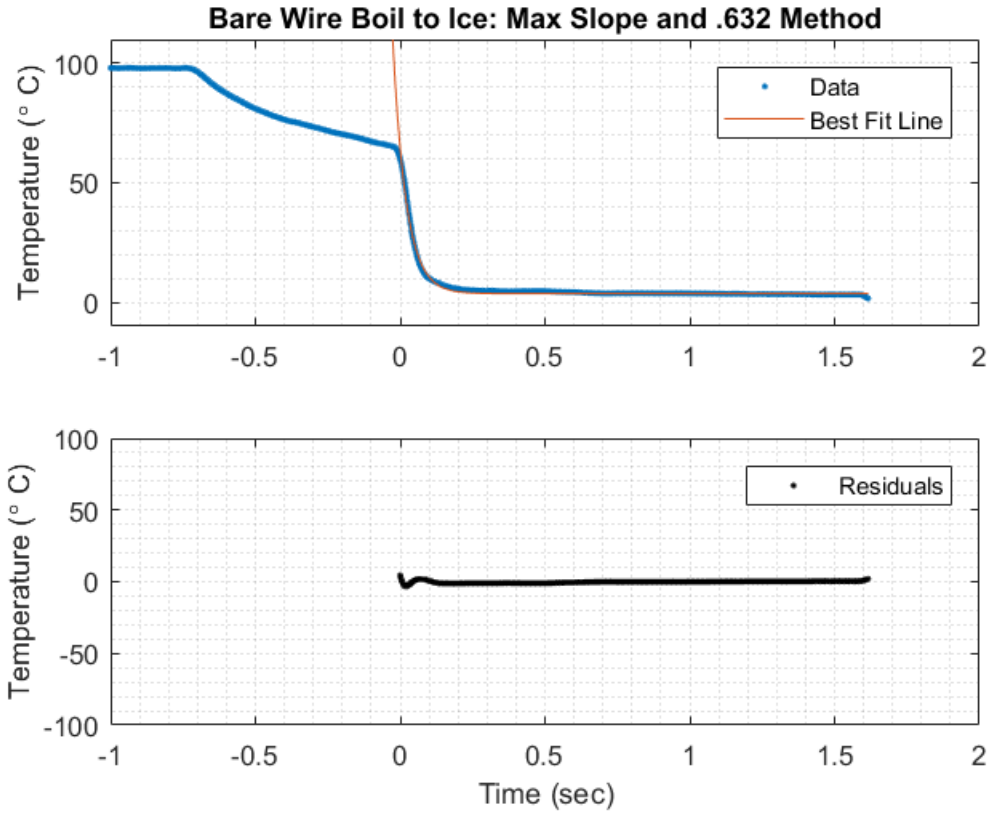
```
end
subplot(2,1,1)
plot(new_time_start_poly_BBI, BBI_Temp_smooth, '.')
hold on
plot(new_time_start_poly_BBI, Temp_tau_poly_value_BBI)
ylabel('Temperature (\circ C)')
title('Bare Wire Boil to Ice: Max Slope and .632 Method')
grid minor
axis([-1, 2, -10, 110])
legend('Data', 'Best Fit Line')
%Residuals
subplot(2,1,2)
residual_value_poly_BBI=Temp_tau_poly_value_BBI-BBI_Temp_smooth;
plot(new_time_start_poly_BBI(start_poly_BBI:length(residual_value_poly_BBI)),resid
ylabel('Temperature (\circ C)')
xlabel('Time (sec)')
grid minor
axis([-1, 2, -100, 100])
legend('Residuals')
figure
```




Bare Wire Boil to Ice: 5σ and Γ Method
ln(Γ)
Data
Best Fit Line
Temperature (° C)
Data
Best Fit Line
Temperature (° C)
Residuals
Time (sec)


Bare Wire Boil to Ice: Max Slope and Γ Method
ln(Γ)
Data
Best Fit Line
Temperature (° C)
Data
Best Fit Line
Temperature (° C)
Residuals
Time (sec)


Bare Wire Boil to Ice: 5σ and .632 Method
Temperature (° C)
Data
Best Fit Line
Residuals
Time (sec)


Bare Wire Boil to Ice: Max Slope and .632 Method
Temperature (° C)
Data
Best Fit Line
Residuals
Time (sec)

BIB

```
for i=1:length(BIB_volt)
    BIB_Temp(i)=interp1(x,y,(BIB_volt(i)*10^3));
end
%smooth data to make anaylsis easier
span=41;
window=ones(span,1)/span;
BIB_Temp_smooth=conv(BIB_Temp,window,'same');

% filter off curving abnormalities at beginning
for i=4000:length(BIB_time)
        if BIB_Temp_smooth(i)<95
            BIB_Temp_smooth(i)=NaN;
    end
end
plot(BIB_time,BIB_Temp_smooth, '.')
hold on
ylabel('Temperature (\circ C)')
xlabel('Time (s)')
title('Bare Wire: Ice to Boil (BIB)')
grid minor
deviation_BIB=std(BIB_Temp_smooth(50:500));
mean_BIB=mean(BIB_Temp_smooth(50:500));
for i=50:length(BIB_Temp_smooth)
```

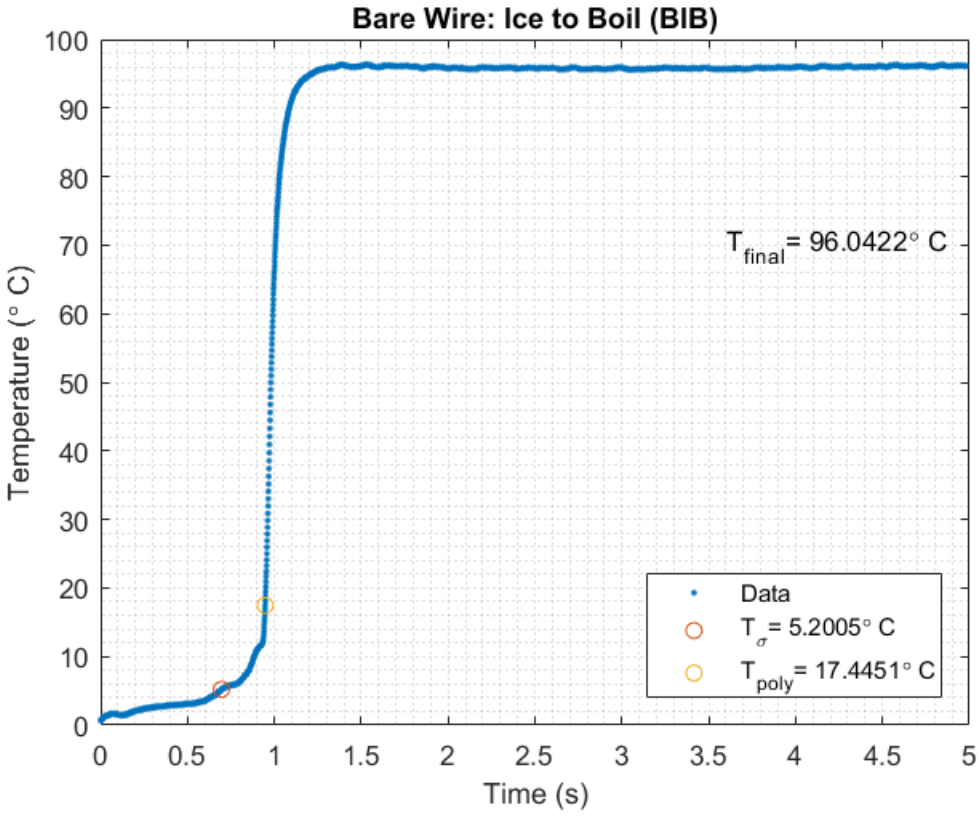
```
    if BIB_Temp_smooth(i)>mean_BIB+5*deviation_BIB
        position_start_std_BIB=i;
        break
    end
end

time_start_std_BIB=BIB_time(position_start_std_BIB);
new_time_start_std_BIB=BIB_time-BIB_time(position_start_std_BIB);
Temp_start_std_BIB=BIB_Temp_smooth(position_start_std_BIB);
plot(time_start_std_BIB,Temp_start_std_BIB,'o')
%Find Ti with polyfit method
start_poly_BIB=1;
for i=750:1250
    BIB_poly(i,:)=polyfit(BIB_time(i:i+50)', BIB_Temp_smooth(i:i+50),
 1);
    if abs(BIB_poly(i))>abs(BIB_poly(start_poly_BIB))
        start_poly_BIB=i;
    end
end


plot(BIB_time(start_poly_BIB)',BIB_Temp_smooth(start_poly_BIB), 'o')
new_time_start_poly_BIB=BIB_time-BIB_time(start_poly_BIB);
BIB_temp_final=mean(BIB_Temp_smooth(4000:4800));
text(3.6,70,['T_f_i_n_a_l= ' num2str(BIB_temp_final) '\circ C'])
legend('Data', ['T_\sigma= ' num2str(Temp_start_std_BIB) '\circ
 C'], ['T_p_o_l_y= ' num2str(BIB_Temp_smooth(start_poly_BIB)) '\circ
 C'], 'Location', 'southeast')

figure

%Find Gamma std
subplot(3,1,1)
for i=1:length(new_time_start_std_BIB)
    gamma_std_BIB(i)=(BIB_temp_final-BIB_Temp_smooth(i))/
(BIB_temp_final-Temp_start_std_BIB);
    if gamma_std_BIB(i)<=0
        break
    end
    ln_gamma_std_BIB(i)=log(gamma_std_BIB(i));
end
new_time_std_range_BIB=new_time_start_std_BIB(1:length(ln_gamma_std_BIB));
plot(new_time_std_range_BIB, ln_gamma_std_BIB, '.')
hold on
find_tau_std_BIB=polyfit(new_time_std_range_BIB(position_start_std_BIB:length(ln_g
 ln_gamma_std_BIB(position_start_std_BIB:length(ln_gamma_std_BIB)) ,
 1);% find tau with stdfit
tau_std_BIB=-1/find_tau_std_BIB(1);
fit_tau_std_BIB=find_tau_std_BIB(1).*new_time_std_range_BIB;
plot(new_time_std_range_BIB, fit_tau_std_BIB);

ylabel('ln(\Gamma)')
xlabel('Time (sec)')
legend('Data', 'Best Fit Line')
```

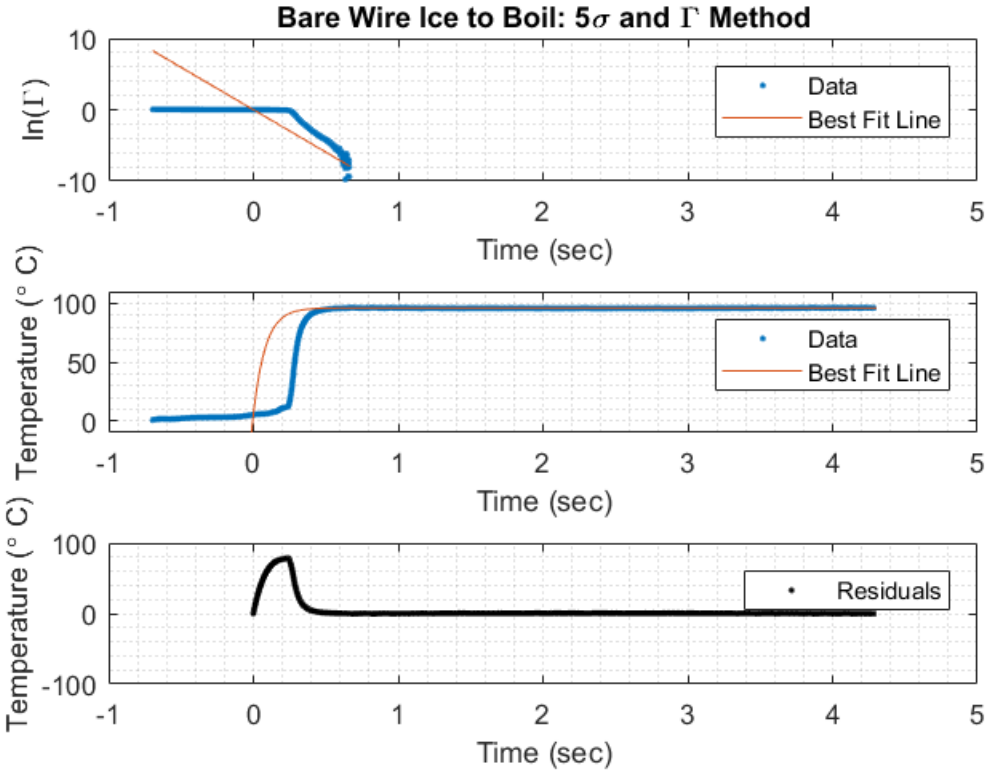
```
% % text(2.8, -8, ['\tau= ' num2str(tau_std_BIB) ' sec'])
grid minor
title('Bare Wire Ice to Boil: 5\sigma and \Gamma Method')
axis([-1, 5, -10, 10])
subplot(3,1,2)
% Estimate plot with time constant
for i=1:length(BIB_time)
    Temp_tau_std_BIB(i)=BIB_temp_final-((BIB_temp_final-
Temp_start_std_BIB)*exp(-new_time_start_std_BIB(i)/tau_std_BIB));
end



plot(new_time_start_std_BIB, BIB_Temp_smooth, '.')
hold on
plot(new_time_start_std_BIB, Temp_tau_std_BIB)
legend('Data', 'Best Fit Line')
ylim([-10, 110])
ylabel('Temperature (\circ C)')
xlabel('Time (sec)')
grid minor
% Residuals
subplot(3,1,3)

    residual_std_BIB=Temp_tau_std_BIB(position_start_std_BIB:end)-
BIB_Temp_smooth(position_start_std_BIB:end);

plot(new_time_start_std_BIB(position_start_std_BIB:end),
 residual_std_BIB, 'bla.')
grid minor
ylabel('Temperature (\circ C)')
xlabel('Time (sec)')
axis([-1,5, -100, 100])
legend('Residuals')
figure
% Find Gamma poly
subplot(3,1,1)
Temp_start_poly_BIB=(BIB_Temp(start_poly_BIB));
for i=1:length(new_time_start_poly_BIB)
    gamma_poly_BIB(i)=(BIB_temp_final - BIB_Temp_smooth(i)) /
 (BIB_temp_final - Temp_start_poly_BIB);
    if gamma_poly_BIB(i)<=0
        break
    end
    ln_gamma_poly_BIB(i)=log(gamma_poly_BIB(i));
end
new_time_poly_range_BIB=new_time_start_poly_BIB(1 :
 length(ln_gamma_poly_BIB));
plot(new_time_poly_range_BIB, ln_gamma_poly_BIB, '.')
hold on
find_tau_poly_BIB=polyfit(new_time_poly_range_BIB(start_poly_BIB:length(ln_gamma_p
 ln_gamma_poly_BIB(start_poly_BIB:length(ln_gamma_poly_BIB)) , 1);%
 find tau with polyfit
tau_poly_BIB=-1 / find_tau_poly_BIB(1);
```

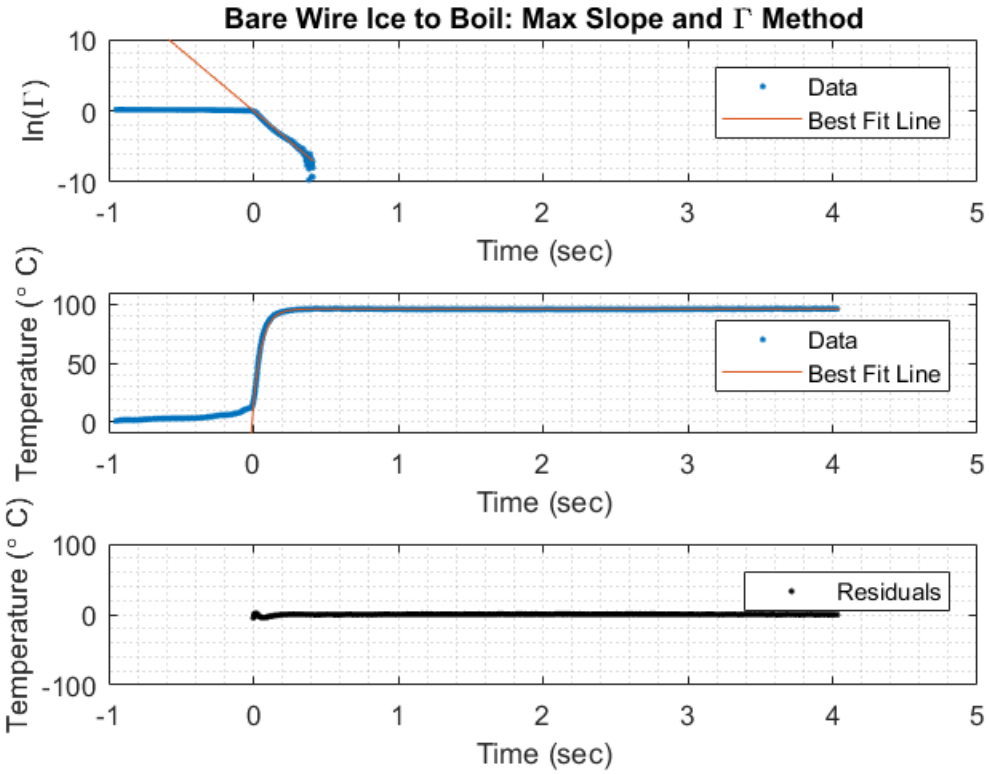
```
fit_tau_poly_BIB=find_tau_poly_BIB(1).*new_time_poly_range_BIB;
plot(new_time_poly_range_BIB, fit_tau_poly_BIB);

ylabel('ln(\Gamma)')
xlabel('Time (sec)')
% % text(2.8, -8, ['\tau= ' num2str(tau_poly_BIB) ' sec'])
grid minor
legend('Data', 'Best Fit Line')
title('Bare Wire Ice to Boil: Max Slope and \Gamma Method')
axis([-1, 5, -10, 10])
subplot(3,1,2)
% Estimate plot with time constant
for i=1:length(BIB_time)
    Temp_tau_poly_BIB(i)=BIB_temp_final-((BIB_temp_final-
Temp_start_poly_BIB)*exp(-new_time_start_poly_BIB(i)/tau_poly_BIB));
end



plot(new_time_start_poly_BIB, BIB_Temp_smooth, '.')

hold on
plot(new_time_start_poly_BIB, Temp_tau_poly_BIB)

ylim([-10, 110])
ylabel('Temperature (\circ C)')
xlabel('Time (sec)')
legend('Data', 'Best Fit Line')
grid minor
% Residuals
subplot(3,1,3)

residual_poly_BIB=Temp_tau_poly_BIB(start_poly_BIB:end)-
BIB_Temp_smooth(start_poly_BIB:end);

plot(new_time_start_poly_BIB(start_poly_BIB:end),
 residual_poly_BIB, 'bla.')
grid minor
ylabel('Temperature (\circ C)')
xlabel('Time (sec)')
axis([-1, 5, -100, 100])
legend('Residuals')
figure


%Calculate Tau with .632       STD
T_at_Tau_std_BIB=Temp_start_std_BIB + .632 * (BIB_temp_final -
 Temp_start_std_BIB);
for i=1:length(new_time_start_std_BIB)
    if BIB_Temp_smooth(i)>T_at_Tau_std_BIB
        Tau_std_value_BIB=new_time_start_std_BIB(i);
        break
    end
end
```

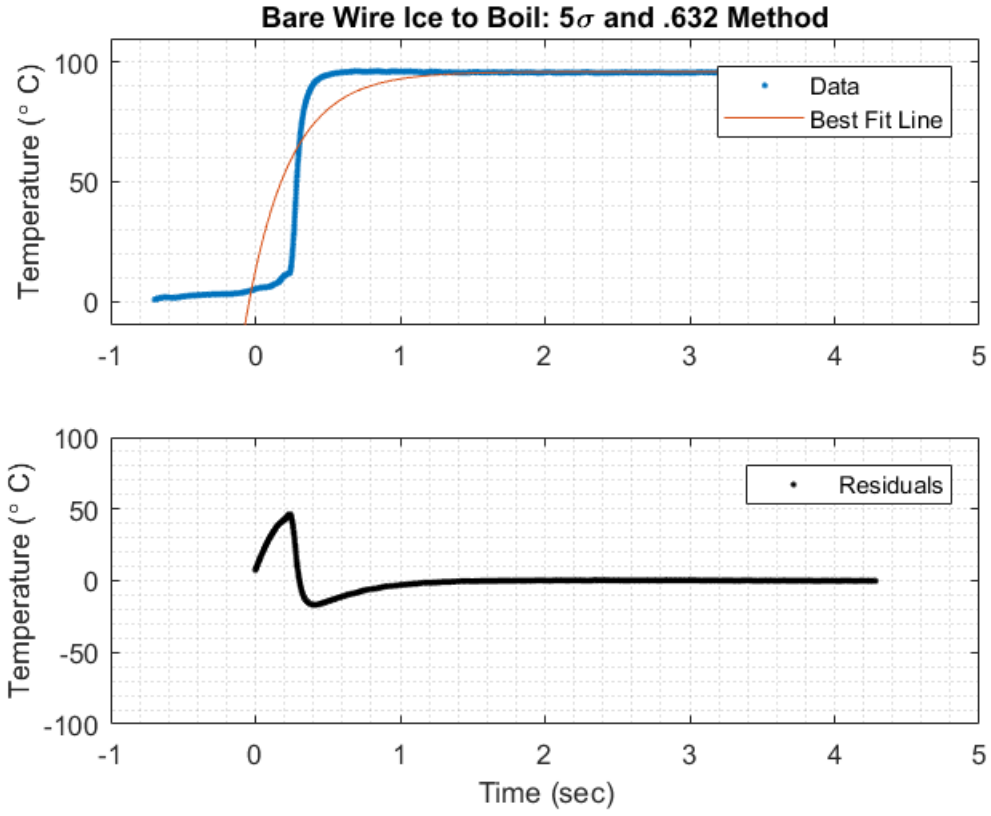
```
for i=1:length(new_time_start_std_BIB)
Temp_tau_std_value_BIB(i)=BIB_temp_final-((BIB_temp_final-
Temp_start_poly_BIB)*exp(-new_time_start_std_BIB(i)/
Tau_std_value_BIB));
end
subplot(2,1,1)
plot(new_time_start_std_BIB, BIB_Temp_smooth, '.')
title('Bare Wire Ice to Boil: 5\sigma and .632 Method')
ylabel('Temperature (\circ C)')
grid minor
hold on
plot(new_time_start_std_BIB, Temp_tau_std_value_BIB)
axis([-1, 5, -10, 110])
legend('Data', 'Best Fit Line')
%Residuals
subplot(2,1,2)
residual_value_std_BIB=Temp_tau_std_value_BIB-BIB_Temp_smooth;
plot(new_time_start_std_BIB(position_start_std_BIB:length(residual_value_std_BIB))
ylabel('Temperature (\circ C)')
xlabel('Time (sec)')
grid minor
axis([-1, 5, -100, 100])
legend('Residuals')
figure
T_at_Tau_poly_BIB=Temp_start_poly_BIB + .632 * (BIB_temp_final -
 Temp_start_poly_BIB);
for i=1:length(new_time_start_poly_BIB)
    if BIB_Temp_smooth(i)>T_at_Tau_poly_BIB
        Tau_poly_value_BIB=new_time_start_poly_BIB(i);
        break
    end
end
for i=1:length(new_time_start_poly_BIB)
Temp_tau_poly_value_BIB(i)=BIB_temp_final-((BIB_temp_final-
Temp_start_poly_BIB)*exp(-new_time_start_poly_BIB(i)/
Tau_poly_value_BIB));
end
subplot(2,1,1)
plot(new_time_start_poly_BIB, BIB_Temp_smooth, '.')
hold on
plot(new_time_start_poly_BIB, Temp_tau_poly_value_BIB)
ylabel('Temperature (\circ C)')
title('Bare Wire Ice to Boil: Max Slope and .632 Method')
grid minor
axis([-1, 5, -10, 110])
legend('Data', 'Best Fit Line')
%Residuals
subplot(2,1,2)
residual_value_poly_BIB=Temp_tau_poly_value_BIB-BIB_Temp_smooth;
plot(new_time_start_poly_BIB(start_poly_BIB:length(residual_value_poly_BIB)),resid
ylabel('Temperature (\circ C)')
xlabel('Time (sec)')
grid minor
axis([-1, 5, -100, 100])
```

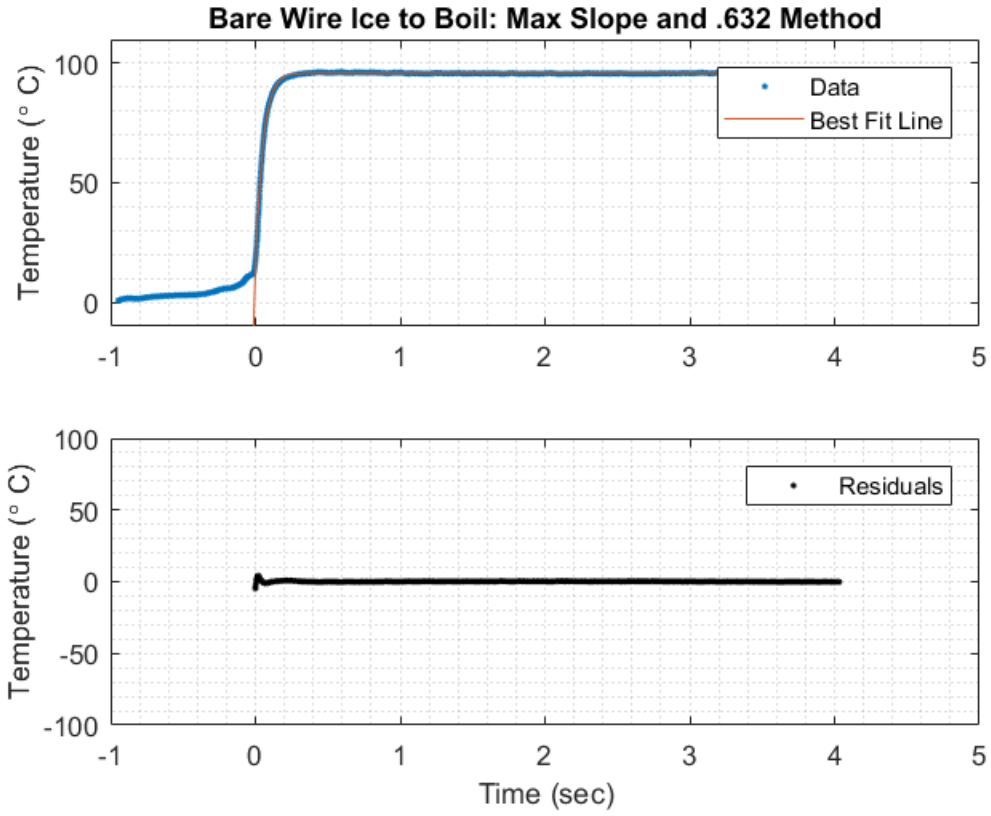
```
legend('Residuals')
figure
```




Bare Wire Ice to Boil: 5σ and Γ Method
ln(Γ)
Data
Best Fit Line
Time (sec)
Temperature (° C)
Data
Best Fit Line
Time (sec)
Temperature (° C)
Residuals
Time (sec)


Bare Wire Ice to Boil: Max Slope and Γ Method
ln(Γ)
Data
Best Fit Line
Time (sec)
Temperature (° C)
Data
Best Fit Line
Time (sec)
Temperature (° C)
Residuals
Time (sec)

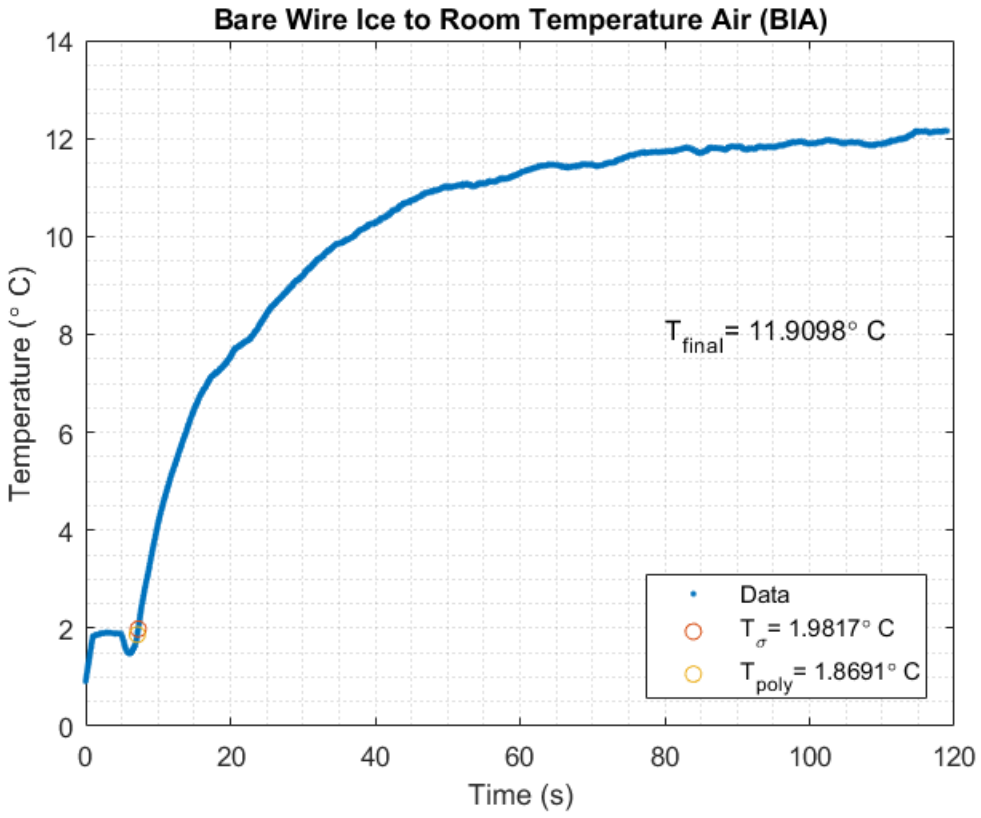

Bare Wire Ice to Boil: 5σ and .632 Method
Temperature (° C)
Data
Best Fit Line
Residuals
Time (sec)


Bare Wire Ice to Boil: Max Slope and .632 Method
Temperature (° C)
Data
Best Fit Line
Residuals
Time (sec)

BIA

```
for i=1:length(BIA_volt)
    BIA_Temp(i)=interp1(x,y,(BIA_volt(i)*10^3));
end
%smooth data to make anaylsis easier
span=200;
window=ones(span,1)/span;
BIA_Temp_smooth=conv(BIA_Temp,window,'same');

% filter off curving abnormalities at beginning
for i=length(BIA_time):-1:4000
        if BIA_Temp_smooth(i)<12.15
            BIA_Temp_smooth(i)=NaN;
        end
        if BIA_Temp_smooth(i)>12.15
            break
        end
end
plot(BIA_time,BIA_Temp_smooth, '.')
hold on
ylabel('Temperature (\circ C)')
xlabel('Time (s)')

grid minor
```

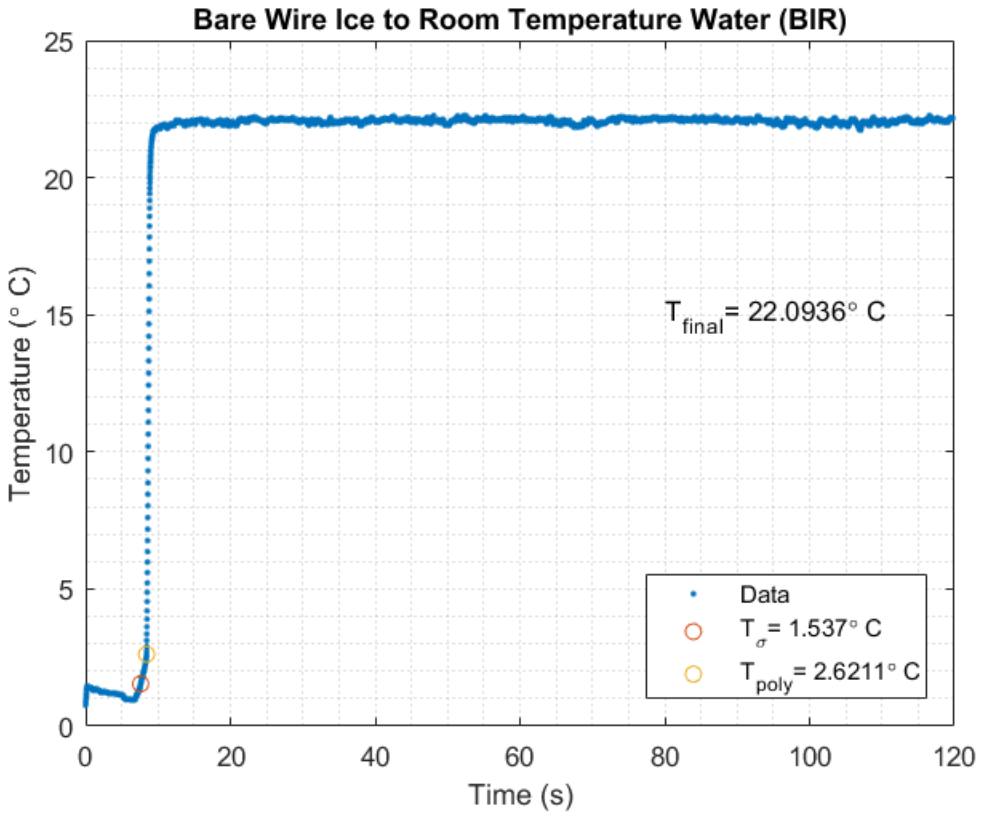
```
deviation_BIA=std(BIA_Temp_smooth(100:500));
mean_BIA=mean(BIA_Temp_smooth(100:500));
for i=300:length(BIA_Temp_smooth)
    if BIA_Temp_smooth(i)>mean_BIA+5*deviation_BIA
        position_start_std_BIA=i;
        break
    end
end

time_start_std_BIA=BIA_time(position_start_std_BIA);
Temp_start_std_BIA=BIA_Temp_smooth(position_start_std_BIA);
plot(time_start_std_BIA,Temp_start_std_BIA,'o')
%Find Ti with polyfit method
start_poly_BIA=1;
for i=300:2000
    BIA_poly(i,:)=polyfit(BIA_time(i:i+50)', BIA_Temp_smooth(i:i+50),
 1);
    if abs(BIA_poly(i))>abs(BIA_poly(start_poly_BIA))
        start_poly_BIA=i;
    end
end


plot(BIA_time(start_poly_BIA)',BIA_Temp_smooth(start_poly_BIA), 'o')
BIA_temp_final=mean(BIA_Temp_smooth(10000:11000));
text(80,8,['T_f_i_n_a_l= ' num2str(BIA_temp_final) '\circ C'])
legend('Data', ['T_\sigma= ' num2str(Temp_start_std_BIA) '\circ
 C'], ['T_p_o_l_y= ' num2str(BIA_Temp_smooth(start_poly_BIA)) '\circ
 C'], 'Location', 'southeast')
title('Bare Wire Ice to Room Temperature Air (BIA)')
figure
```


Bare Wire Ice to Room Temperature Air (BIA)
Temperature (° C)
Time (s)
0
2
4
6
8
10
12
14
0
20
40
60
80
100
120
T final = 11.9098° C
Data
T σ = 1.9817° C
T poly = 1.8691° C

BIR

```
for i=1:length(BIR_volt)
    BIR_Temp(i)=interp1(x,y,(BIR_volt(i)*10^3));
end
%smooth data to make anaylsis easier
span=40;
window=ones(span,1)/span;
BIR_Temp_smooth=conv(BIR_Temp,window,'same');

% filter off curving abnormalities at beginning
for i=length(BIR_time):-1:4000
        if BIR_Temp_smooth(i)<22.16
            BIR_Temp_smooth(i)=NaN;
        end
        if BIR_Temp_smooth(i)>22.16
            break
        end
end
plot(BIR_time,BIR_Temp_smooth, '.')
hold on
ylabel('Temperature (\circ C)')
xlabel('Time (s)')
title('Bare Wire Ice to Room Temperature Water (BIR)')
grid minor
deviation_BIR=std(BIR_Temp_smooth(100:500));
mean_BIR=mean(BIR_Temp_smooth(100:500));
for i=100:length(BIR_Temp_smooth)
    if BIR_Temp_smooth(i)>mean_BIR+5*deviation_BIR
        position_start_std_BIR=i;
        break
    end
end

time_start_std_BIR=BIR_time(position_start_std_BIR);
Temp_start_std_BIR=BIR_Temp_smooth(position_start_std_BIR);
plot(time_start_std_BIR,Temp_start_std_BIR,'o')
%Find Ti with polyfit method
start_poly_BIR=1;
for i=600:1250
    BIR_poly(i,:)=polyfit(BIR_time(i:i+50)', BIR_Temp_smooth(i:i+50),
 1);
    if abs(BIR_poly(i))>abs(BIR_poly(start_poly_BIR))
        start_poly_BIR=i;
    end
end


plot(BIR_time(start_poly_BIR)',BIR_Temp_smooth(start_poly_BIR), 'o')
BIR_temp_final=mean(BIR_Temp_smooth(4000:4800));
text(80,15,['T_f_i_n_a_l= ' num2str(BIR_temp_final) '\circ C'])
```

```
legend('Data', ['T_\sigma= ' num2str(Temp_start_std_BIR) '\circ
 C'], ['T_p_o_l_y= ' num2str(BIR_Temp_smooth(start_poly_BIR)) '\circ
 C'], 'Location', 'southeast')

figure
```



EIB_SS

```
for i=1:length(EIB_SS_volt)
    EIB_SS_Temp(i)=interp1(x,y,(EIB_SS_volt(i)*10^3));
end
%smooth data to make anaylsis easier
span=40;
window=ones(span,1)/span;
EIB_SS_Temp_smooth=conv(EIB_SS_Temp,window,'same');

% filter off curving abnormalities at beginning
for i=4500:length(EIB_SS_time)
        if EIB_SS_Temp_smooth(i)<98
            EIB_SS_Temp_smooth(i)=NaN;
    end
end
plot(EIB_SS_time,EIB_SS_Temp_smooth, '.')
hold on
ylabel('Temperature (\circ C)')
xlabel('Time (s)')
title('Embedded Stainless Steel Ice to Boil (EIB SS)')
grid minor
deviation_EIB_SS=std(EIB_SS_Temp_smooth(100:500));
mean_EIB_SS=mean(EIB_SS_Temp_smooth(100:500));
for i=100:length(EIB_SS_Temp_smooth)
```

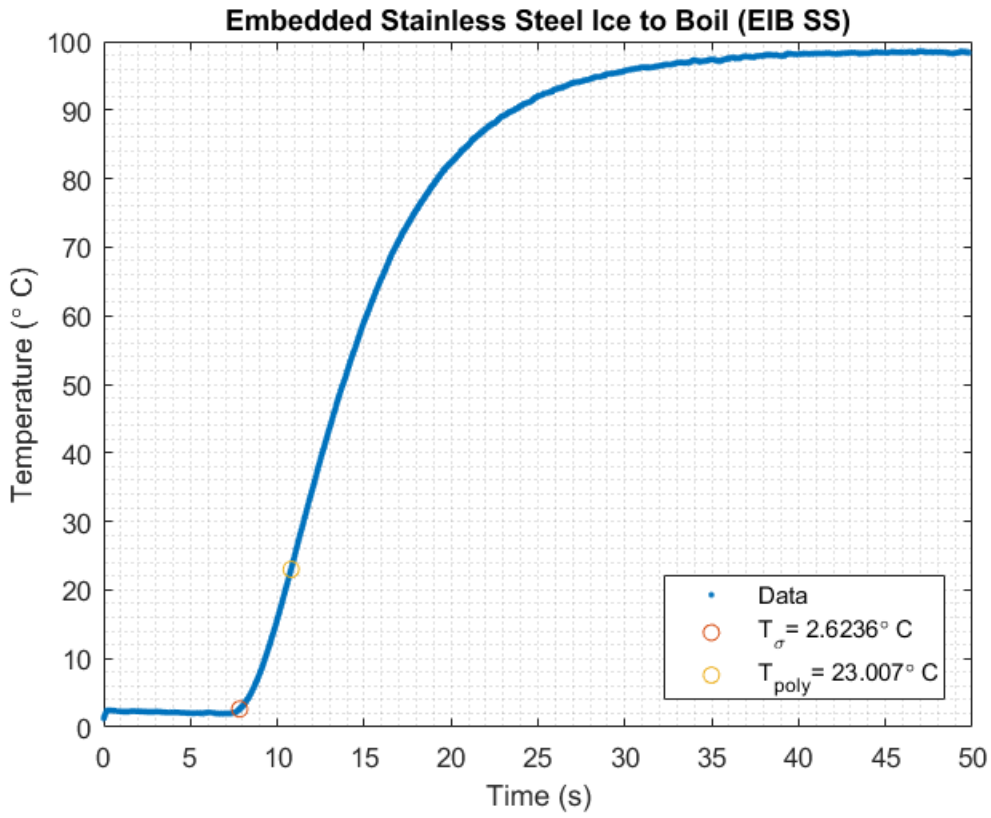
```
    if EIB_SS_Temp_smooth(i)>mean_EIB_SS+5*deviation_EIB_SS
        position_start_std_EIB_SS=i;
        break
    end
end

time_start_std_EIB_SS=EIB_SS_time(position_start_std_EIB_SS);
Temp_start_std_EIB_SS=EIB_SS_Temp_smooth(position_start_std_EIB_SS);
plot(time_start_std_EIB_SS,Temp_start_std_EIB_SS,'o')
%Find Ti with polyfit method
start_poly_EIB_SS=1;
for i=100:4000
    EIB_SS_poly(i,:)=polyfit(EIB_SS_time(i:i+50)',
 EIB_SS_Temp_smooth(i:i+50), 1);
    if abs(EIB_SS_poly(i))>abs(EIB_SS_poly(start_poly_EIB_SS))
        start_poly_EIB_SS=i;
    end
end


plot(EIB_SS_time(start_poly_EIB_SS)',EIB_SS_Temp_smooth(start_poly_EIB_SS), 'o')
EIB_SS_temp_final=mean(EIB_SS_Temp_smooth(4000:4800));
text(80,15,['T_f_i_n_a_l= ' num2str(EIB_SS_temp_final) '\circ C'])
legend('Data', ['T_\sigma= ' num2str(Temp_start_std_EIB_SS) '\circ
 C'], ['T_p_o_l_y= '
 num2str(EIB_SS_Temp_smooth(start_poly_EIB_SS)) '\circ
 C'], 'Location', 'southeast')

figure
%Find Gamma std
new_time_start_std_EIB_SS=EIB_SS_time-
EIB_SS_time(position_start_std_EIB_SS);
subplot(3,1,1)
for i=23:length(new_time_start_std_EIB_SS)
    gamma_std_EIB_SS(i)=(EIB_SS_temp_final-EIB_SS_Temp_smooth(i))/
(EIB_SS_temp_final-Temp_start_std_EIB_SS);
    if gamma_std_EIB_SS(i)<=0
        break
    end
    ln_gamma_std_EIB_SS(i)=log(gamma_std_EIB_SS(i));
end
new_time_std_range_EIB_SS=new_time_start_std_EIB_SS(1:length(ln_gamma_std_EIB_SS))
plot(new_time_std_range_EIB_SS, ln_gamma_std_EIB_SS, '.')
hold on
find_tau_std_EIB_SS=polyfit(new_time_std_range_EIB_SS(position_start_std_EIB_SS:le
 ln_gamma_std_EIB_SS(position_start_std_EIB_SS:length(ln_gamma_std_EIB_SS)) ,
 1);% find tau with stdfit
tau_std_EIB_SS=-1/find_tau_std_EIB_SS(1);
fit_tau_std_EIB_SS=find_tau_std_EIB_SS(1).*new_time_std_range_EIB_SS;
plot(new_time_std_range_EIB_SS, fit_tau_std_EIB_SS);

ylabel('ln(\Gamma)')
xlabel('Time (sec)')
axis([-5, 45,-10, 10])
```

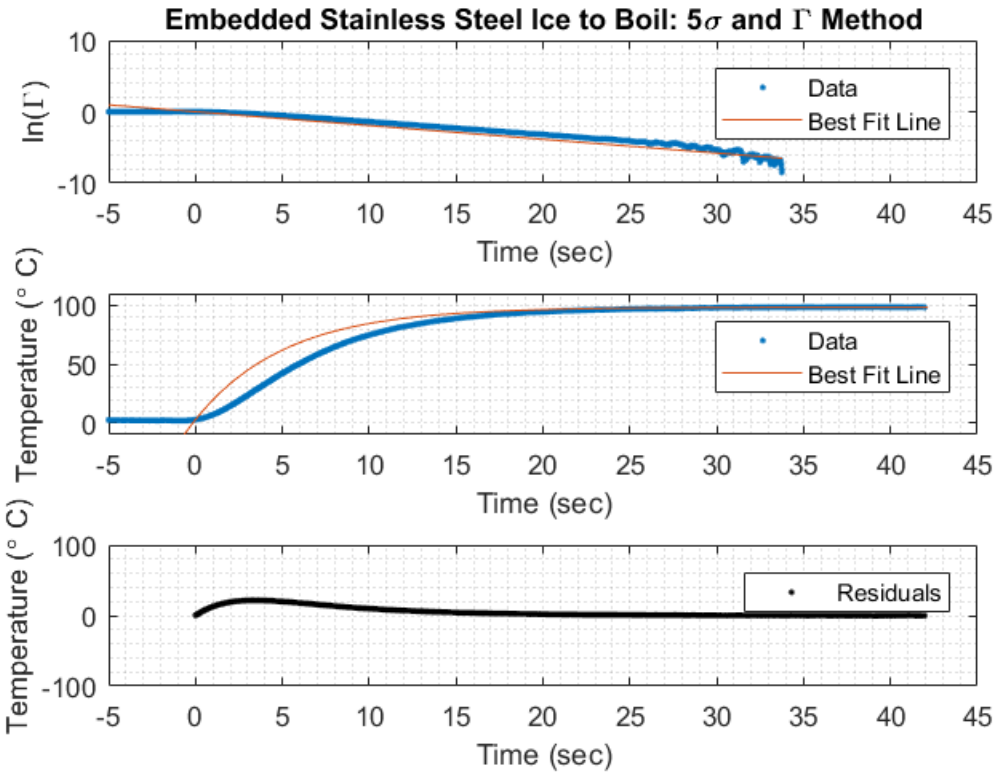
```
% % text(2.8, -8, ['\tau= ' num2str(tau_std_EIB_SS) ' sec'])
grid minor
legend('Data', 'Best Fit Line')
title('Embedded Stainless Steel Ice to Boil: 5\sigma and \Gamma
 Method')
subplot(3,1,2)
% Estimate plot with time constant
for i=1:length(EIB_SS_time)
    Temp_tau_std_EIB_SS(i)=EIB_SS_temp_final-((EIB_SS_temp_final-
Temp_start_std_EIB_SS)*exp(-new_time_start_std_EIB_SS(i)/
tau_std_EIB_SS));
end



plot(new_time_start_std_EIB_SS, EIB_SS_Temp_smooth, '.')
hold on
plot(new_time_start_std_EIB_SS, Temp_tau_std_EIB_SS)


ylim([-10, 110])
ylabel('Temperature (\circ C)')
xlabel('Time (sec)')
axis([-5, 45, -10, 110])
legend('Data', 'Best Fit Line')
grid minor
% Residuals
subplot(3,1,3)


 residual_std_EIB_SS=Temp_tau_std_EIB_SS(position_start_std_EIB_SS:end)-
EIB_SS_Temp_smooth(position_start_std_EIB_SS:end);

plot(new_time_start_std_EIB_SS(position_start_std_EIB_SS:end),
 residual_std_EIB_SS, 'bla.')
grid minor
ylabel('Temperature (\circ C)')
xlabel('Time (sec)')
legend('Residuals')
axis([-5, 45, -100, 100])
figure
% Find Gamma poly
new_time_start_poly_EIB_SS=EIB_SS_time-EIB_SS_time(start_poly_EIB_SS);
subplot(3,1,1)
Temp_start_poly_EIB_SS=(EIB_SS_Temp(start_poly_EIB_SS));
for i=23:length(new_time_start_poly_EIB_SS)
    gamma_poly_EIB_SS(i)=(EIB_SS_temp_final-EIB_SS_Temp_smooth(i))/
(EIB_SS_temp_final-Temp_start_poly_EIB_SS);
    if gamma_poly_EIB_SS(i)<=0
        break
    end
    ln_gamma_poly_EIB_SS(i)=log(gamma_poly_EIB_SS(i));
end
new_time_poly_range_EIB_SS=new_time_start_poly_EIB_SS(1:length(ln_gamma_poly_EIB_S
```

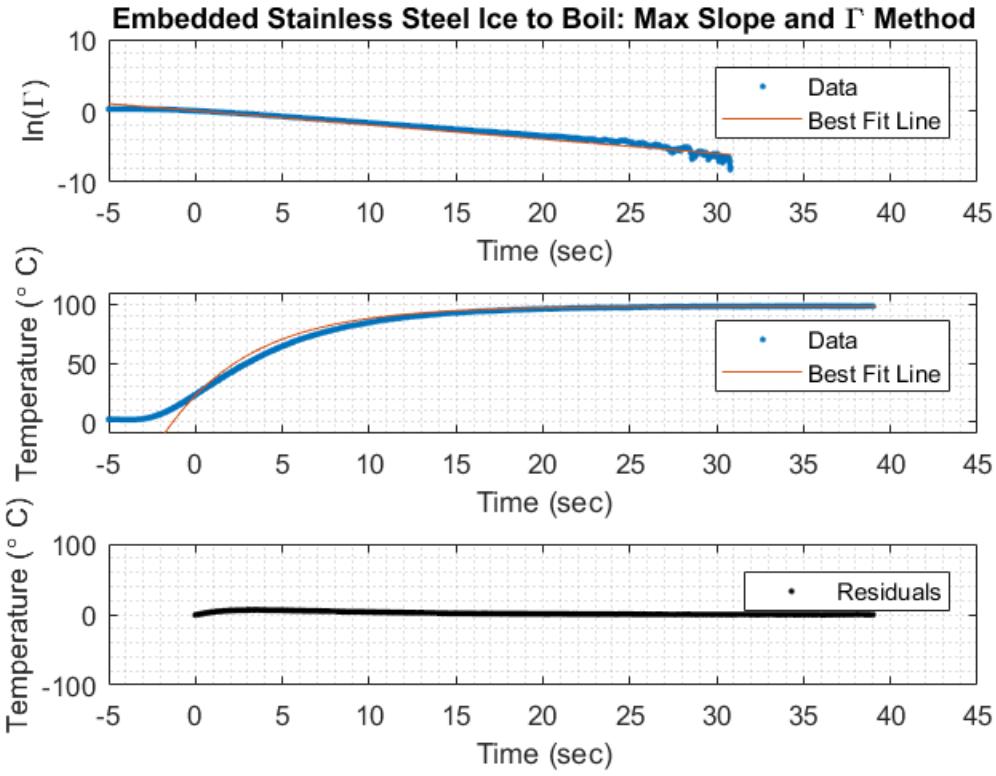
```
plot(new_time_poly_range_EIB_SS, ln_gamma_poly_EIB_SS, '.')
hold on
find_tau_poly_EIB_SS=polyfit(new_time_poly_range_EIB_SS(start_poly_EIB_SS:length(l
 ln_gamma_poly_EIB_SS(start_poly_EIB_SS:length(ln_gamma_poly_EIB_SS)) ,
 1);% find tau with polyfit
tau_poly_EIB_SS=-1/find_tau_poly_EIB_SS(1);
fit_tau_poly_EIB_SS=find_tau_poly_EIB_SS(1).*new_time_poly_range_EIB_SS;
plot(new_time_poly_range_EIB_SS, fit_tau_poly_EIB_SS);

ylabel('ln(\Gamma)')
xlabel('Time (sec)')
axis([-5, 45,-10, 10])
% % text(2.8, -8, ['\tau= ' num2str(tau_poly_EIB_SS) ' sec'])
grid minor
legend('Data', 'Best Fit Line')
title('Embedded Stainless Steel Ice to Boil: Max Slope and \Gamma
 Method')
subplot(3,1,2)
% Estimate plot with time constant
for i=1:length(EIB_SS_time)
    Temp_tau_poly_EIB_SS(i)=EIB_SS_temp_final-((EIB_SS_temp_final-
Temp_start_poly_EIB_SS)*exp(-new_time_start_poly_EIB_SS(i)/
tau_poly_EIB_SS));
end



plot(new_time_start_poly_EIB_SS, EIB_SS_Temp_smooth, '.')
hold on
plot(new_time_start_poly_EIB_SS, Temp_tau_poly_EIB_SS)


ylim([-10, 110])
ylabel('Temperature (\circ C)')
xlabel('Time (sec)')
axis([-5, 45, -10, 110])
legend('Data', 'Best Fit Line')
grid minor
% Residuals
subplot(3,1,3)

    residual_poly_EIB_SS=Temp_tau_poly_EIB_SS(start_poly_EIB_SS:end)-
EIB_SS_Temp_smooth(start_poly_EIB_SS:end);

plot(new_time_start_poly_EIB_SS(start_poly_EIB_SS:end),
 residual_poly_EIB_SS, 'bla.')
grid minor
ylabel('Temperature (\circ C)')
xlabel('Time (sec)')
axis([-5, 45, -100, 100])
legend('Residuals')
figure
%Calculate Tau with .632       STD
```

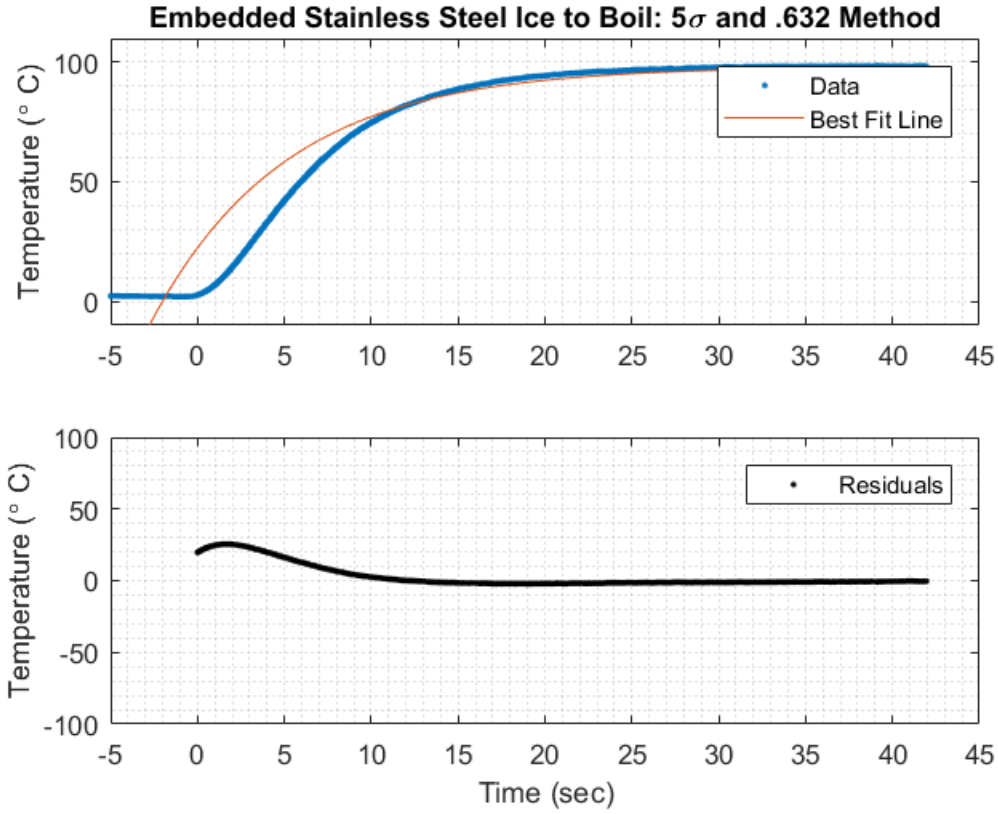
```
T_at_Tau_std_EIB_SS=Temp_start_std_EIB_SS + .632 * (EIB_SS_temp_final
 - Temp_start_std_EIB_SS);
for i=1:length(new_time_start_std_EIB_SS)
    if EIB_SS_Temp_smooth(i)>T_at_Tau_std_EIB_SS
        Tau_std_value_EIB_SS=new_time_start_std_EIB_SS(i);
        break
    end
end
for i=1:length(new_time_start_std_EIB_SS)
Temp_tau_std_value_EIB_SS(i)=EIB_SS_temp_final-((EIB_SS_temp_final-
Temp_start_poly_EIB_SS)*exp(-new_time_start_std_EIB_SS(i)/
Tau_std_value_EIB_SS));
end
subplot(2,1,1)
plot(new_time_start_std_EIB_SS, EIB_SS_Temp_smooth, '.')
title('Embedded Stainless Steel Ice to Boil: 5\sigma and .632 Method')
ylabel('Temperature (\circ C)')
grid minor
axis([-5, 45, -10, 110])
hold on
plot(new_time_start_std_EIB_SS, Temp_tau_std_value_EIB_SS)
legend('Data', 'Best Fit Line')
%Residuals
subplot(2,1,2)
residual_value_std_EIB_SS=Temp_tau_std_value_EIB_SS-
EIB_SS_Temp_smooth;
plot(new_time_start_std_EIB_SS(position_start_std_EIB_SS:length(residual_value_std
ylabel('Temperature (\circ C)')
xlabel('Time (sec)')
axis([-5, 45, -100, 100])
legend('Residuals')
grid minor

figure
T_at_Tau_poly_EIB_SS=Temp_start_poly_EIB_SS + .632 *
 (EIB_SS_temp_final - Temp_start_poly_EIB_SS);
for i=1:length(new_time_start_poly_EIB_SS)
    if EIB_SS_Temp_smooth(i)>T_at_Tau_poly_EIB_SS
        Tau_poly_value_EIB_SS=new_time_start_poly_EIB_SS(i);
        break
    end
end
for i=1:length(new_time_start_poly_EIB_SS)
Temp_tau_poly_value_EIB_SS(i)=EIB_SS_temp_final-((EIB_SS_temp_final-
Temp_start_poly_EIB_SS)*exp(-new_time_start_poly_EIB_SS(i)/
Tau_poly_value_EIB_SS));
end
subplot(2,1,1)
plot(new_time_start_poly_EIB_SS, EIB_SS_Temp_smooth, '.')
hold on
plot(new_time_start_poly_EIB_SS, Temp_tau_poly_value_EIB_SS)
ylabel('Temperature (\circ C)')
title('Embedded Stainless Steel Ice to Boil: Max Slope and .632
 Method')
```

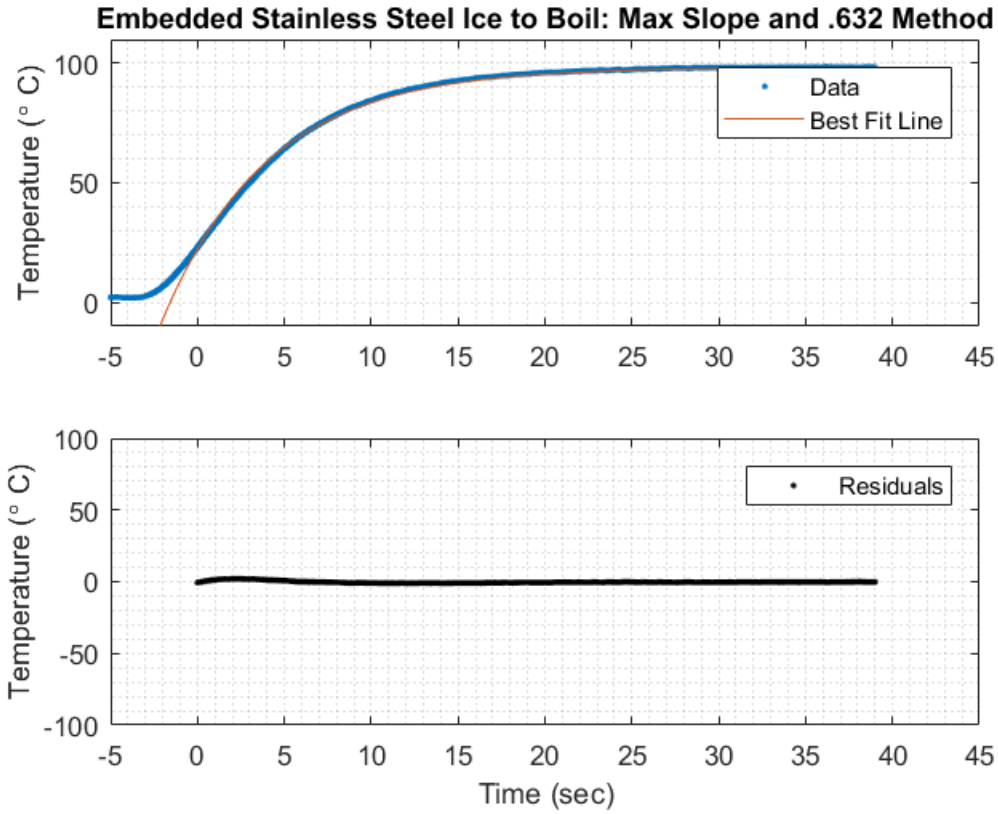
```
axis([-5, 45, -10, 110])
legend('Data', 'Best Fit Line')
grid minor

%Residuals
subplot(2,1,2)
residual_value_poly_EIB_SS=Temp_tau_poly_value_EIB_SS-
EIB_SS_Temp_smooth;
plot(new_time_start_poly_EIB_SS(start_poly_EIB_SS:length(residual_value_poly_EIB_S
ylabel('Temperature (\circ C)')
xlabel('Time (sec)')
grid minor
axis([-5, 45, -100, 100])
legend('Residuals')
figure
```




Embedded Stainless Steel Ice to Boil: 5σ and Γ Method
ln(Γ)
Data
Best Fit Line
Time (sec)
Temperature (° C)
Data
Best Fit Line
Time (sec)
Temperature (° C)
Residuals
Time (sec)


Embedded Stainless Steel Ice to Boil: Max Slope and Γ Method
ln(Γ)
Data
Best Fit Line
Time (sec)
Temperature (° C)
Data
Best Fit Line
Time (sec)
Temperature (° C)
Residuals
Time (sec)


Embedded Stainless Steel Ice to Boil: 5σ and .632 Method
Temperature (° C)
Data
Best Fit Line
Residuals
Time (sec)


Embedded Stainless Steel Ice to Boil: Max Slope and .632 Method
Temperature (° C)
Data
Best Fit Line
Residuals
Time (sec)

EBI_SS

```
for i=1:length(EBI_SS_volt)
    EBI_SS_Temp(i)=interp1(x,y,(EBI_SS_volt(i)*10^3));
end
%smooth data to make anaylsis easier
span=40;
window=ones(span,1)/span;
EBI_SS_Temp_smooth=conv(EBI_SS_Temp,window,'same');

% filter off curving abnormalities at beginning
for i=1:300
        if EBI_SS_Temp_smooth(i)<99
            EBI_SS_Temp_smooth(i)=NaN;
    end
            if EBI_SS_Temp_smooth(i)>99
                break
            end
end
for i=5000:-1:4500
    if EBI_SS_Temp_smooth(i)>13.2
        break
    end
    if EBI_SS_Temp_smooth(i)<13.2
        EBI_SS_Temp_smooth(i)=NaN;
```

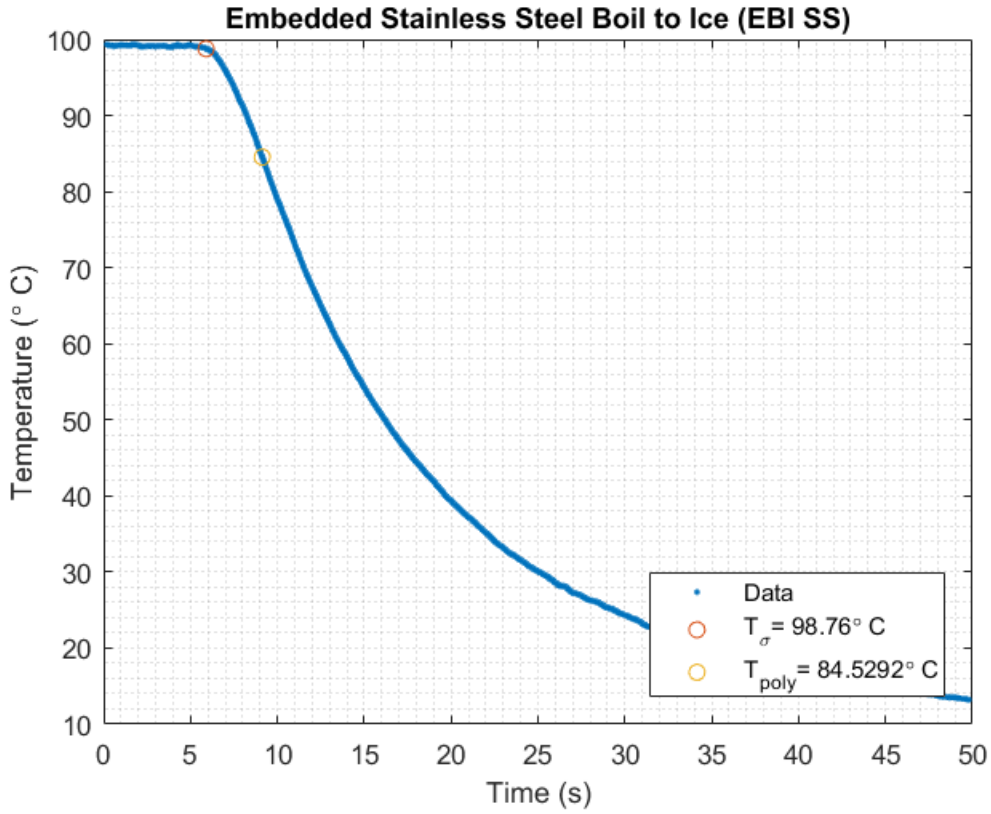
```
    end

end
plot(EBI_SS_time,EBI_SS_Temp_smooth, '.')
hold on
ylabel('Temperature (\circ C)')
xlabel('Time (s)')
title('Embedded Stainless Steel Boil to Ice (EBI SS)')
grid minor
deviation_EBI_SS=std(EBI_SS_Temp_smooth(50:500));
mean_EBI_SS=mean(EBI_SS_Temp_smooth(50:500));
for i=100:length(EBI_SS_Temp_smooth)
    if EBI_SS_Temp_smooth(i)<mean_EBI_SS-5*deviation_EBI_SS
        position_start_std_EBI_SS=i;
        break
    end
end

time_start_std_EBI_SS=EBI_SS_time(position_start_std_EBI_SS);
Temp_start_std_EBI_SS=EBI_SS_Temp_smooth(position_start_std_EBI_SS);
plot(time_start_std_EBI_SS,Temp_start_std_EBI_SS,'o')
%Find Ti with polyfit method
start_poly_EBI_SS=1;
for i=500:4000
    EBI_SS_poly(i,:)=polyfit(EBI_SS_time(i:i+70)',
 EBI_SS_Temp_smooth(i:i+70), 1);
    if abs(EBI_SS_poly(i))>abs(EBI_SS_poly(start_poly_EBI_SS))
        start_poly_EBI_SS=i;
    end
end


plot(EBI_SS_time(start_poly_EBI_SS)',EBI_SS_Temp_smooth(start_poly_EBI_SS), 'o')
EBI_SS_temp_final=mean(EBI_SS_Temp_smooth(4000:4800));
text(80,15,['T_f_i_n_a_l= ' num2str(EBI_SS_temp_final) '\circ C'])
legend('Data', ['T_\sigma= ' num2str(Temp_start_std_EBI_SS) '\circ
 C'], ['T_p_o_l_y= '
 num2str(EBI_SS_Temp_smooth(start_poly_EBI_SS)) '\circ
 C'], 'Location', 'southeast')

figure
%Find Gamma std
new_time_start_std_EBI_SS=EBI_SS_time-
EBI_SS_time(position_start_std_EBI_SS);
subplot(3,1,1)
for i=23:length(new_time_start_std_EBI_SS)
    gamma_std_EBI_SS(i)=(EBI_SS_temp_final-EBI_SS_Temp_smooth(i))/
(EBI_SS_temp_final-Temp_start_std_EBI_SS);
    if gamma_std_EBI_SS(i)<=0
        break
    end
    ln_gamma_std_EBI_SS(i)=log(gamma_std_EBI_SS(i));
end
new_time_std_range_EBI_SS=new_time_start_std_EBI_SS(1:length(ln_gamma_std_EBI_SS))
```

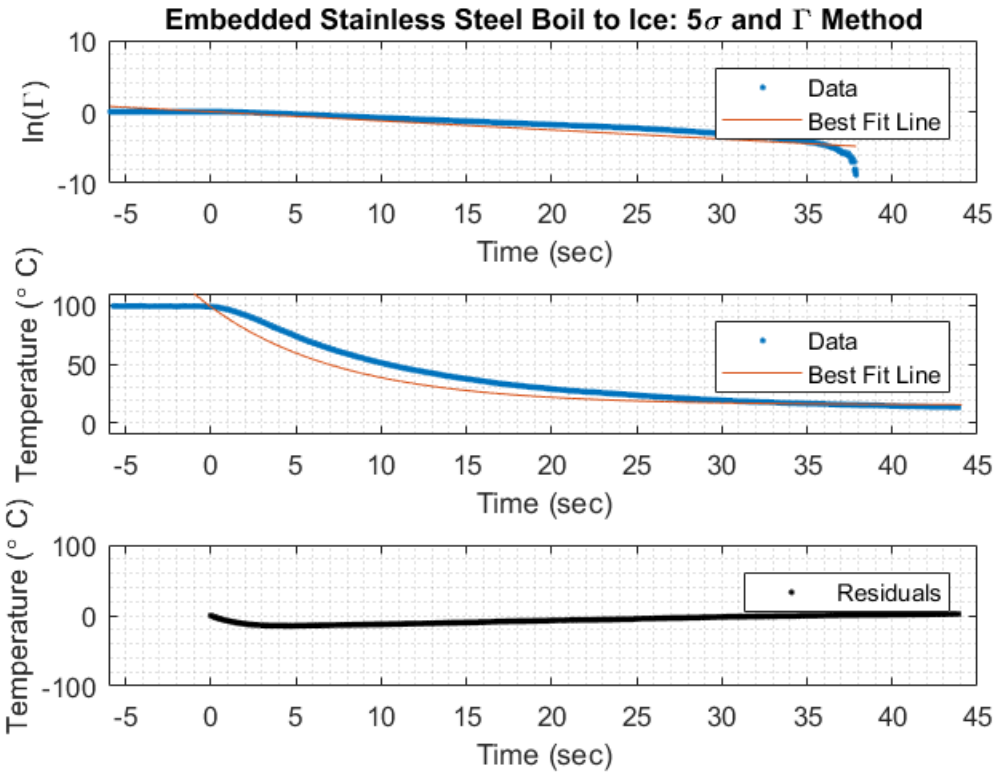
```
plot(new_time_std_range_EBI_SS, ln_gamma_std_EBI_SS, '.')
hold on
find_tau_std_EBI_SS=polyfit(new_time_std_range_EBI_SS(position_start_std_EBI_SS:le
 ln_gamma_std_EBI_SS(position_start_std_EBI_SS:length(ln_gamma_std_EBI_SS)) ,
 1);% find tau with stdfit
tau_std_EBI_SS=-1/find_tau_std_EBI_SS(1);
fit_tau_std_EBI_SS=find_tau_std_EBI_SS(1).*new_time_std_range_EBI_SS;
plot(new_time_std_range_EBI_SS, fit_tau_std_EBI_SS);

ylabel('ln(\Gamma)')
xlabel('Time (sec)')
axis([-6, 45, -10, 10])
% % text(2.8, -8, ['\tau= ' num2str(tau_std_EBI_SS) ' sec'])
grid minor
legend('Data', 'Best Fit Line')
title('Embedded Stainless Steel Boil to Ice: 5\sigma and \Gamma
 Method')
subplot(3,1,2)
% Estimate plot with time constant
for i=1:length(EBI_SS_time)
    Temp_tau_std_EBI_SS(i)=EBI_SS_temp_final-((EBI_SS_temp_final-
Temp_start_std_EBI_SS)*exp(-new_time_start_std_EBI_SS(i)/
tau_std_EBI_SS));
end



plot(new_time_start_std_EBI_SS, EBI_SS_Temp_smooth, '.')
hold on
plot(new_time_start_std_EBI_SS, Temp_tau_std_EBI_SS)

ylim([-10, 110])
ylabel('Temperature (\circ C)')
xlabel('Time (sec)')
axis([-6, 45, -10, 110])
legend('Data', 'Best Fit Line')
grid minor
% Residuals
subplot(3,1,3)


 residual_std_EBI_SS=Temp_tau_std_EBI_SS(position_start_std_EBI_SS:end)-
EBI_SS_Temp_smooth(position_start_std_EBI_SS:end);

plot(new_time_start_std_EBI_SS(position_start_std_EBI_SS:end),
 residual_std_EBI_SS, 'bla.')
grid minor
ylabel('Temperature (\circ C)')
xlabel('Time (sec)')
axis([-6, 45, -100, 100])
legend('Residuals')
figure
% Find Gamma poly
new_time_start_poly_EBI_SS=EBI_SS_time-EBI_SS_time(start_poly_EBI_SS);
```

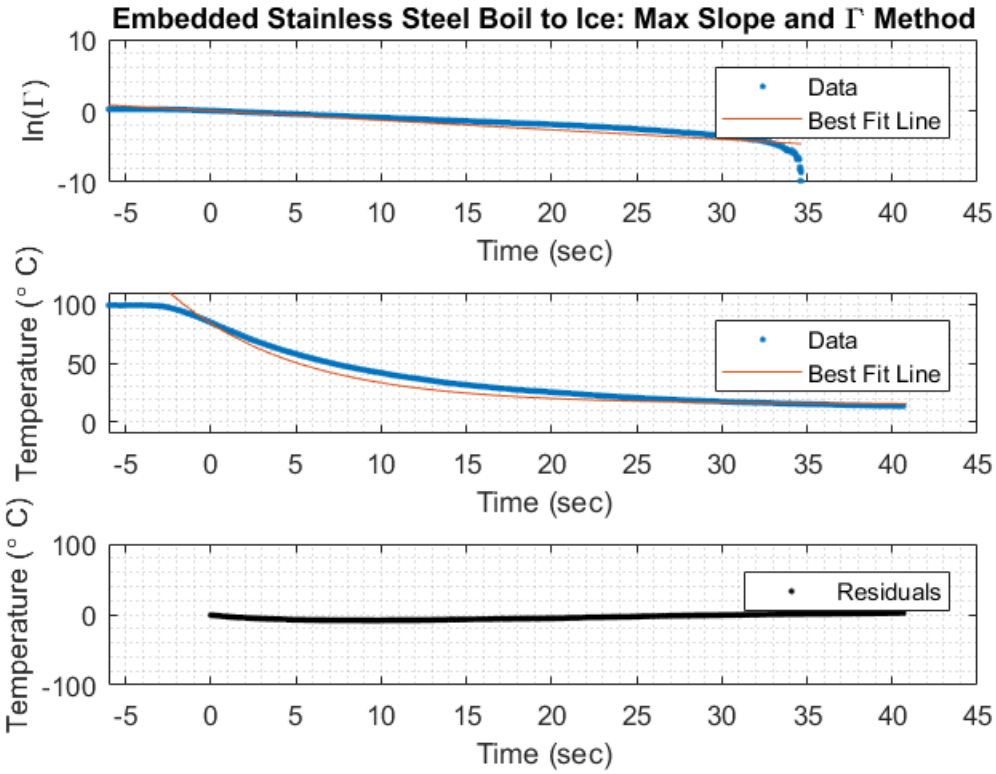
```
subplot(3,1,1)
Temp_start_poly_EBI_SS=(EBI_SS_Temp(start_poly_EBI_SS));
for i=23:length(new_time_start_poly_EBI_SS)
    gamma_poly_EBI_SS(i)=(EBI_SS_temp_final-EBI_SS_Temp_smooth(i))/
(EBI_SS_temp_final-Temp_start_poly_EBI_SS);
    if gamma_poly_EBI_SS(i)<=0
        break
    end
    ln_gamma_poly_EBI_SS(i)=log(gamma_poly_EBI_SS(i));
end
new_time_poly_range_EBI_SS=new_time_start_poly_EBI_SS(1:length(ln_gamma_poly_EBI_S
plot(new_time_poly_range_EBI_SS, ln_gamma_poly_EBI_SS, '.')
hold on
find_tau_poly_EBI_SS=polyfit(new_time_poly_range_EBI_SS(start_poly_EBI_SS:length(l
 ln_gamma_poly_EBI_SS(start_poly_EBI_SS:length(ln_gamma_poly_EBI_SS)) ,
 1);% find tau with polyfit
tau_poly_EBI_SS=-1/find_tau_poly_EBI_SS(1);
fit_tau_poly_EBI_SS=find_tau_poly_EBI_SS(1).*new_time_poly_range_EBI_SS;
plot(new_time_poly_range_EBI_SS, fit_tau_poly_EBI_SS);

ylabel('ln(\Gamma)')
xlabel('Time (sec)')
axis([-6, 45, -10, 10])
% % text(2.8, -8, ['\tau= ' num2str(tau_poly_EBI_SS) ' sec'])
grid minor
legend('Data', 'Best Fit Line')
title('Embedded Stainless Steel Boil to Ice: Max Slope and \Gamma
 Method')
subplot(3,1,2)
% Estimate plot with time constant
for i=1:length(EBI_SS_time)
    Temp_tau_poly_EBI_SS(i)=EBI_SS_temp_final-((EBI_SS_temp_final-
Temp_start_poly_EBI_SS)*exp(-new_time_start_poly_EBI_SS(i)/
tau_poly_EBI_SS));
end



plot(new_time_start_poly_EBI_SS, EBI_SS_Temp_smooth, '.')
hold on
plot(new_time_start_poly_EBI_SS, Temp_tau_poly_EBI_SS)

ylim([-10, 110])
ylabel('Temperature (\circ C)')
xlabel('Time (sec)')
axis([-6, 45, -10, 110])
legend('Data', 'Best Fit Line')
grid minor
% Residuals
subplot(3,1,3)

residual_poly_EBI_SS=Temp_tau_poly_EBI_SS(start_poly_EBI_SS:end)-
EBI_SS_Temp_smooth(start_poly_EBI_SS:end);
```

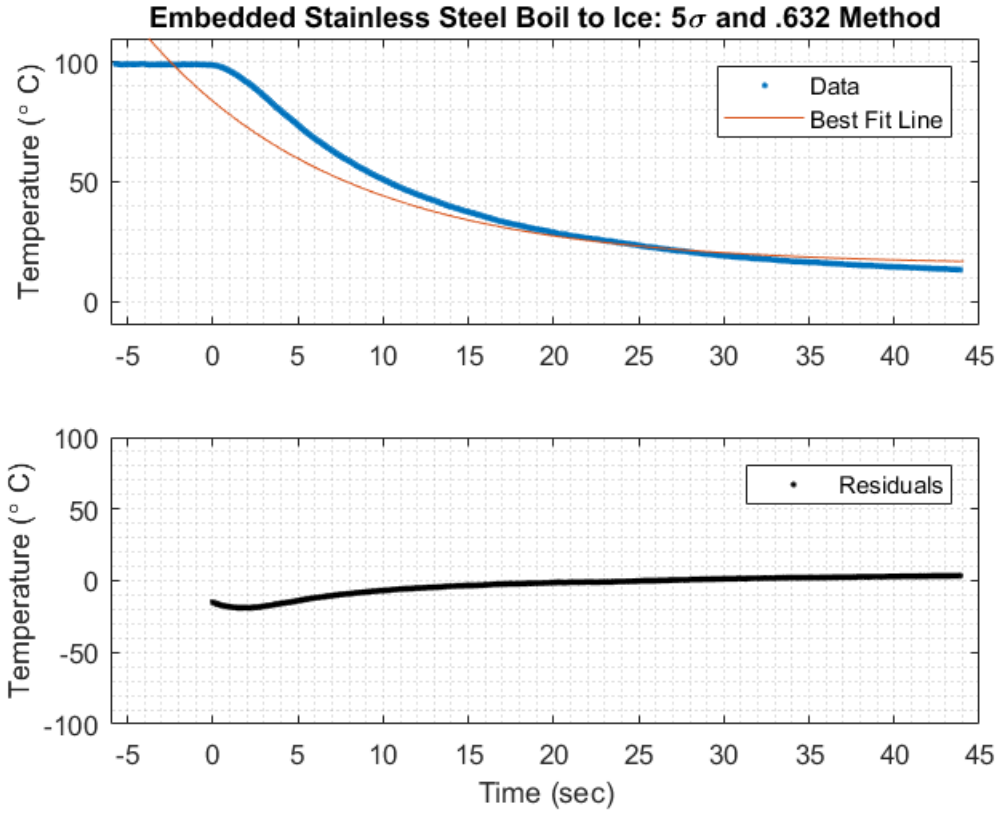
```
plot(new_time_start_poly_EBI_SS(start_poly_EBI_SS:end),
 residual_poly_EBI_SS, 'bla.')
grid minor
ylabel('Temperature (\circ C)')
xlabel('Time (sec)')
axis([-6, 45, -100, 100])
legend('Residuals')
figure

%Calculate Tau with .632       STD
T_at_Tau_std_EBI_SS=Temp_start_std_EBI_SS + .632 * (EBI_SS_temp_final
 - Temp_start_std_EBI_SS);
for i=1:length(new_time_start_std_EBI_SS)
    if EBI_SS_Temp_smooth(i)<T_at_Tau_std_EBI_SS
        Tau_std_value_EBI_SS=new_time_start_std_EBI_SS(i);
        break
    end
end
for i=1:length(new_time_start_std_EBI_SS)
Temp_tau_std_value_EBI_SS(i)=EBI_SS_temp_final-((EBI_SS_temp_final-
Temp_start_poly_EBI_SS)*exp(-new_time_start_std_EBI_SS(i)/
Tau_std_value_EBI_SS));
end
subplot(2,1,1)
plot(new_time_start_std_EBI_SS, EBI_SS_Temp_smooth, '.')
title('Embedded Stainless Steel Boil to Ice: 5\sigma and .632 Method')
ylabel('Temperature (\circ C)')
grid minor
axis([-6, 45, -10, 110])
hold on
plot(new_time_start_std_EBI_SS, Temp_tau_std_value_EBI_SS)
legend('Data', 'Best Fit Line')
%Residuals
subplot(2,1,2)
residual_value_std_EBI_SS=Temp_tau_std_value_EBI_SS-
EBI_SS_Temp_smooth;
plot(new_time_start_std_EBI_SS(position_start_std_EBI_SS:length(residual_value_std
ylabel('Temperature (\circ C)')
xlabel('Time (sec)')
axis([-6, 45, -100, 100])
grid minor
legend('Residuals')
figure
T_at_Tau_poly_EBI_SS=Temp_start_poly_EBI_SS + .632 *
 (EBI_SS_temp_final - Temp_start_poly_EBI_SS);
for i=1:length(new_time_start_poly_EBI_SS)
    if EBI_SS_Temp_smooth(i)<T_at_Tau_poly_EBI_SS
        Tau_poly_value_EBI_SS=new_time_start_poly_EBI_SS(i);
        break
    end
end
for i=1:length(new_time_start_poly_EBI_SS)
```

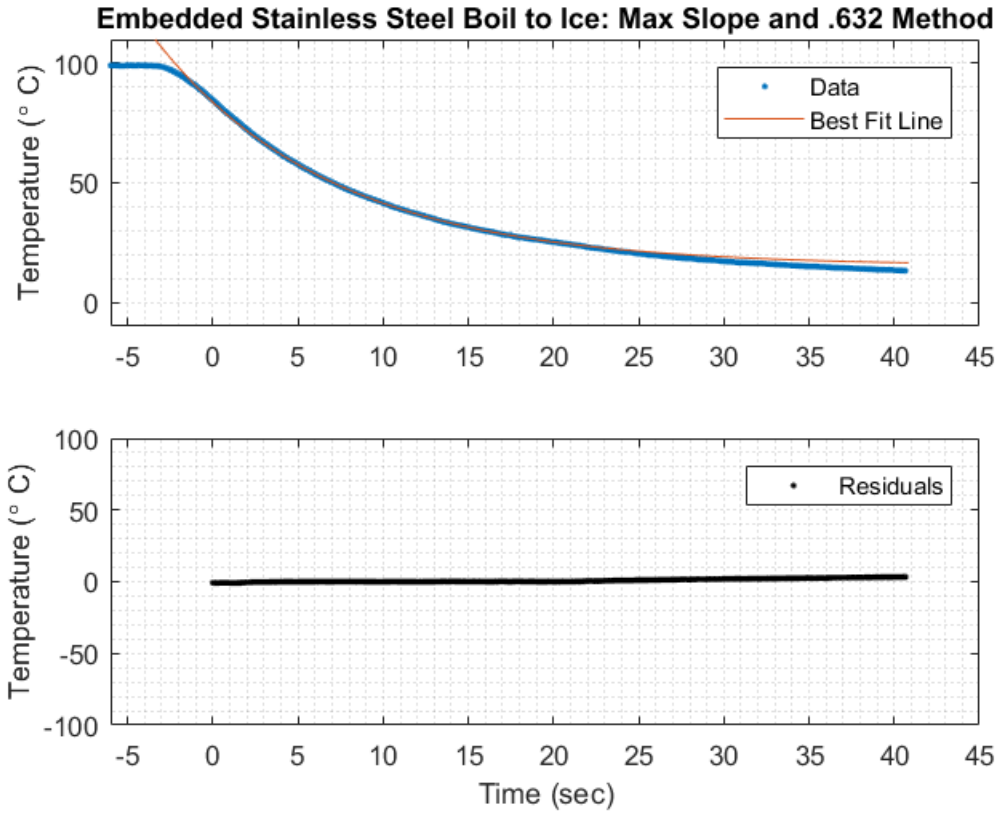
```
Temp_tau_poly_value_EBI_SS(i)=EBI_SS_temp_final-((EBI_SS_temp_final-
Temp_start_poly_EBI_SS)*exp(-new_time_start_poly_EBI_SS(i)/
Tau_poly_value_EBI_SS));
end
subplot(2,1,1)
plot(new_time_start_poly_EBI_SS, EBI_SS_Temp_smooth, '.')
hold on
plot(new_time_start_poly_EBI_SS, Temp_tau_poly_value_EBI_SS)
ylabel('Temperature (\circ C)')
title('Embedded Stainless Steel Boil to Ice: Max Slope and .632
 Method')
axis([-6, 45, -10, 110])
grid minor
legend('Data', 'Best Fit Line')
%Residuals
subplot(2,1,2)
residual_value_poly_EBI_SS=Temp_tau_poly_value_EBI_SS-
EBI_SS_Temp_smooth;
plot(new_time_start_poly_EBI_SS(start_poly_EBI_SS:length(residual_value_poly_EBI_S
ylabel('Temperature (\circ C)')
xlabel('Time (sec)')
grid minor
axis([-6, 45, -100, 100])
legend('Residuals')
figure
```




Embedded Stainless Steel Boil to Ice: 5σ and Γ Method
ln(Γ)
Data
Best Fit Line
Time (sec)
Temperature (° C)
Residuals


Embedded Stainless Steel Boil to Ice: Max Slope and Γ Method
ln(Γ)
Data
Best Fit Line
Time (sec)
Temperature (° C)
Residuals


Embedded Stainless Steel Boil to Ice: 5σ and .632 Method
Temperature (° C)
Data
Best Fit Line
Residuals
Time (sec)


Embedded Stainless Steel Boil to Ice: Max Slope and .632 Method
Temperature (° C)
Data
Best Fit Line
Residuals
Time (sec)

EIB_AL

```
for i=1:length(EIB_AL_volt)
    EIB_AL_Temp(i)=interp1(x,y,(EIB_AL_volt(i)*10^3));
end
%smooth data to make anaylsis easier
span=40;
window=ones(span,1)/span;
EIB_AL_Temp_smooth=conv(EIB_AL_Temp,window,'same');

%filter off curving abnormalities at beginning
for i=length(EIB_AL_time):-1:4500
        if EIB_AL_Temp_smooth(i)<98
            EIB_AL_Temp_smooth(i)=NaN;
        end
        if EIB_AL_Temp_smooth(i)>98
            break
        end
end
plot(EIB_AL_time,EIB_AL_Temp_smooth, '.')
hold on
ylabel('Temperature (\circ C)')
xlabel('Time (s)')
title('Embedded Aluminum Ice to Boil (EIB AL)')
grid minor
```

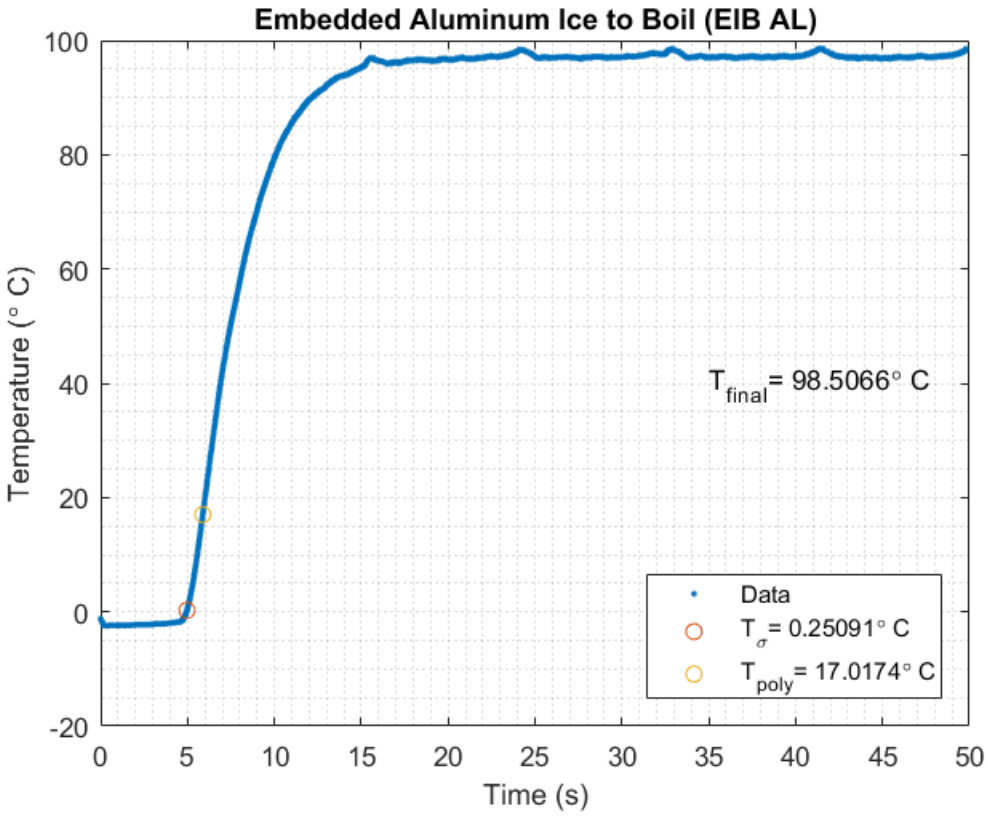
```
deviation_EIB_AL=std(EIB_AL_Temp_smooth(100:500));
mean_EIB_AL=mean(EIB_AL_Temp_smooth(100:500));
for i=100:length(EIB_AL_Temp_smooth)
    if EIB_AL_Temp_smooth(i)>mean_EIB_AL+5*deviation_EIB_AL
        position_start_std_EIB_AL=i;
        break
    end
end

time_start_std_EIB_AL=EIB_AL_time(position_start_std_EIB_AL);
Temp_start_std_EIB_AL=EIB_AL_Temp_smooth(position_start_std_EIB_AL);
plot(time_start_std_EIB_AL,Temp_start_std_EIB_AL,'o')
%Find Ti with polyfit method
start_poly_EIB_AL=1;
for i=100:4000
    EIB_AL_poly(i,:)=polyfit(EIB_AL_time(i:i+50)',
 EIB_AL_Temp_smooth(i:i+50), 1);
    if abs(EIB_AL_poly(i))>abs(EIB_AL_poly(start_poly_EIB_AL))
        start_poly_EIB_AL=i;
    end
end


plot(EIB_AL_time(start_poly_EIB_AL)',EIB_AL_Temp_smooth(start_poly_EIB_AL), 'o')
EIB_AL_temp_final=EIB_AL_Temp_smooth(4980);
text(35,40,['T_f_i_n_a_l= ' num2str(EIB_AL_temp_final) '\circ C'])
legend('Data', ['T_\sigma= ' num2str(Temp_start_std_EIB_AL) '\circ
 C'], ['T_p_o_l_y= '
 num2str(EIB_AL_Temp_smooth(start_poly_EIB_AL)) '\circ
 C'], 'Location', 'southeast')

figure
%Find Gamma std
new_time_start_std_EIB_AL=EIB_AL_time-
EIB_AL_time(position_start_std_EIB_AL);
subplot(3,1,1)
for i=23:length(new_time_start_std_EIB_AL)
    gamma_std_EIB_AL(i)=(EIB_AL_temp_final-EIB_AL_Temp_smooth(i))/
(EIB_AL_temp_final-Temp_start_std_EIB_AL);
    if gamma_std_EIB_AL(i)<=0
        break
    end
    ln_gamma_std_EIB_AL(i)=log(gamma_std_EIB_AL(i));
end
new_time_std_range_EIB_AL=new_time_start_std_EIB_AL(1:length(ln_gamma_std_EIB_AL))
plot(new_time_std_range_EIB_AL, ln_gamma_std_EIB_AL, '.')
hold on
find_tau_std_EIB_AL=polyfit(new_time_std_range_EIB_AL(position_start_std_EIB_AL:le
 ln_gamma_std_EIB_AL(position_start_std_EIB_AL:length(ln_gamma_std_EIB_AL)) ,
 1);% find tau with stdfit
tau_std_EIB_AL=-1/find_tau_std_EIB_AL(1);
fit_tau_std_EIB_AL=find_tau_std_EIB_AL(1).*new_time_std_range_EIB_AL;
plot(new_time_std_range_EIB_AL, fit_tau_std_EIB_AL);
```

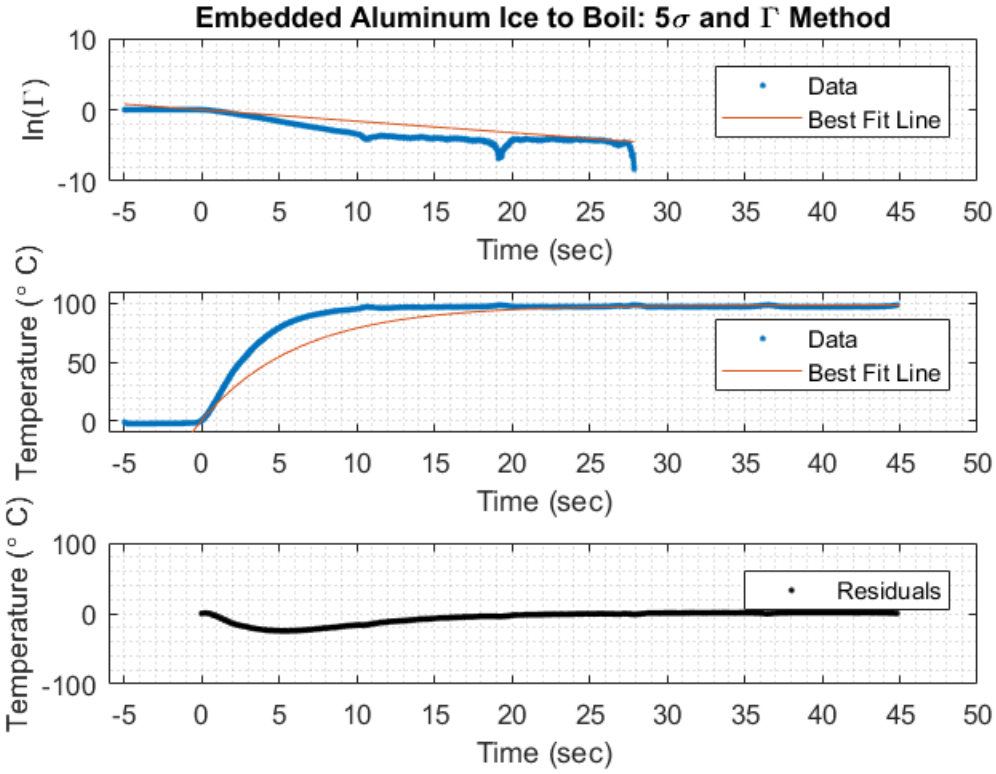
```
ylabel('ln(\Gamma)')
xlabel('Time (sec)')
axis([-6, 50, -10, 10])
% % text(2.8, -8, ['\tau= ' num2str(tau_std_EIB_AL) ' sec'])
grid minor
legend('Data', 'Best Fit Line')
title('Embedded Aluminum Ice to Boil: 5\sigma and \Gamma Method')
subplot(3,1,2)
% Estimate plot with time constant
for i=1:length(EIB_AL_time)
    Temp_tau_std_EIB_AL(i)=EIB_AL_temp_final-((EIB_AL_temp_final-
Temp_start_std_EIB_AL)*exp(-new_time_start_std_EIB_AL(i)/
tau_std_EIB_AL));
end



plot(new_time_start_std_EIB_AL, EIB_AL_Temp_smooth, '.')

hold on
plot(new_time_start_std_EIB_AL, Temp_tau_std_EIB_AL)

legend('Data', 'Best Fit Line')
ylabel('Temperature (\circ C)')
axis([-6, 50, -10, 110])
xlabel('Time (sec)')
grid minor
% Residuals
subplot(3,1,3)


 residual_std_EIB_AL=Temp_tau_std_EIB_AL(position_start_std_EIB_AL:end)-
EIB_AL_Temp_smooth(position_start_std_EIB_AL:end);

plot(new_time_start_std_EIB_AL(position_start_std_EIB_AL:end),
 residual_std_EIB_AL, 'bla.')
grid minor
ylabel('Temperature (\circ C)')
axis([-6, 50, -100, 100])
xlabel('Time (sec)')
legend('Residuals')
figure
% Find Gamma poly
new_time_start_poly_EIB_AL=EIB_AL_time-EIB_AL_time(start_poly_EIB_AL);
subplot(3,1,1)
Temp_start_poly_EIB_AL=(EIB_AL_Temp(start_poly_EIB_AL));
for i=23:length(new_time_start_poly_EIB_AL)
    gamma_poly_EIB_AL(i)=(EIB_AL_temp_final-EIB_AL_Temp_smooth(i))/
(EIB_AL_temp_final-Temp_start_poly_EIB_AL);
    if gamma_poly_EIB_AL(i)<=0
        break
    end
    ln_gamma_poly_EIB_AL(i)=log(gamma_poly_EIB_AL(i));
end
```

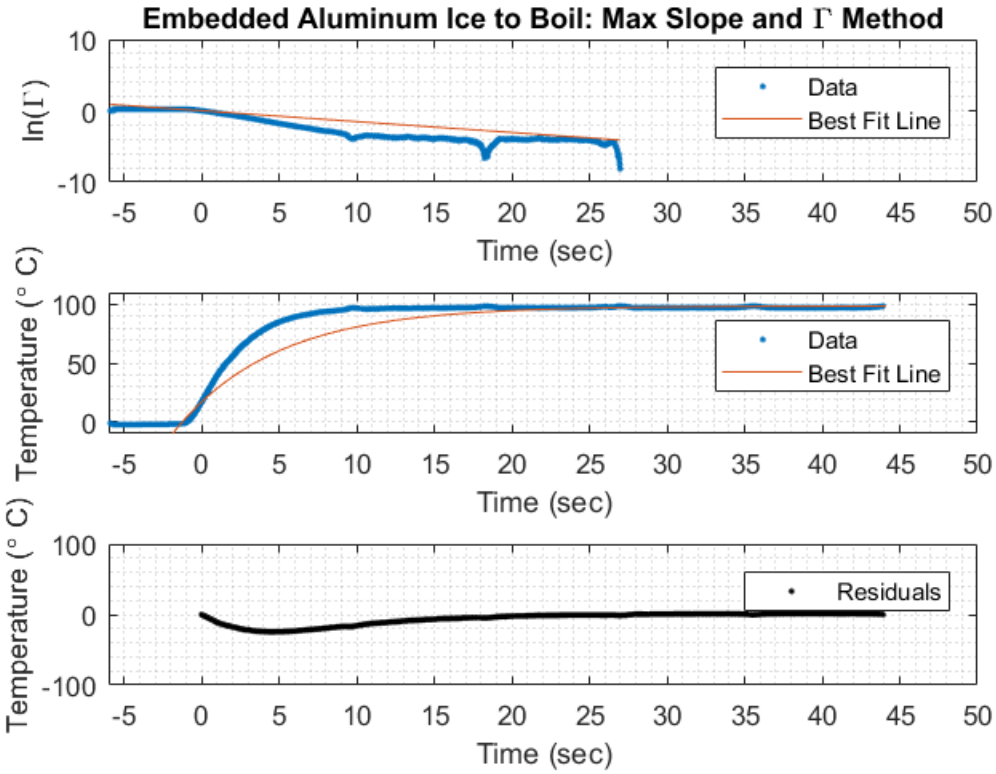
```
new_time_poly_range_EIB_AL=new_time_start_poly_EIB_AL(1:length(ln_gamma_poly_EIB_A
plot(new_time_poly_range_EIB_AL, ln_gamma_poly_EIB_AL, '.')
hold on
find_tau_poly_EIB_AL=polyfit(new_time_poly_range_EIB_AL(start_poly_EIB_AL:length(l
 ln_gamma_poly_EIB_AL(start_poly_EIB_AL:length(ln_gamma_poly_EIB_AL)) ,
 1);% find tau with polyfit
tau_poly_EIB_AL=-1/find_tau_poly_EIB_AL(1);
fit_tau_poly_EIB_AL=find_tau_poly_EIB_AL(1).*new_time_poly_range_EIB_AL;
plot(new_time_poly_range_EIB_AL, fit_tau_poly_EIB_AL);

ylabel('ln(\Gamma)')
xlabel('Time (sec)')
legend('Data', 'Best Fit Line')
% % text(2.8, -8, ['\tau= ' num2str(tau_poly_EIB_AL) ' sec'])
grid minor
axis([-6, 50, -10, 10])
title('Embedded Aluminum Ice to Boil: Max Slope and \Gamma Method')
subplot(3,1,2)
% Estimate plot with time constant
for i=1:length(EIB_AL_time)
    Temp_tau_poly_EIB_AL(i)=EIB_AL_temp_final-((EIB_AL_temp_final-
Temp_start_poly_EIB_AL)*exp(-new_time_start_poly_EIB_AL(i)/
tau_poly_EIB_AL));
end



plot(new_time_start_poly_EIB_AL, EIB_AL_Temp_smooth, '.')
hold on
plot(new_time_start_poly_EIB_AL, Temp_tau_poly_EIB_AL)
legend('Data', 'Best Fit Line')

ylabel('Temperature (\circ C)')
axis([-6, 50, -10, 110])

xlabel('Time (sec)')
grid minor
% Residuals
subplot(3,1,3)

    residual_poly_EIB_AL=Temp_tau_poly_EIB_AL(start_poly_EIB_AL:end)-
EIB_AL_Temp_smooth(start_poly_EIB_AL:end);

plot(new_time_start_poly_EIB_AL(start_poly_EIB_AL:end),
 residual_poly_EIB_AL, 'bla.')
grid minor
ylabel('Temperature (\circ C)')
axis([-6, 50, -100, 100])
xlabel('Time (sec)')
legend('Residuals')
figure

%Calculate Tau with .632       STD
```

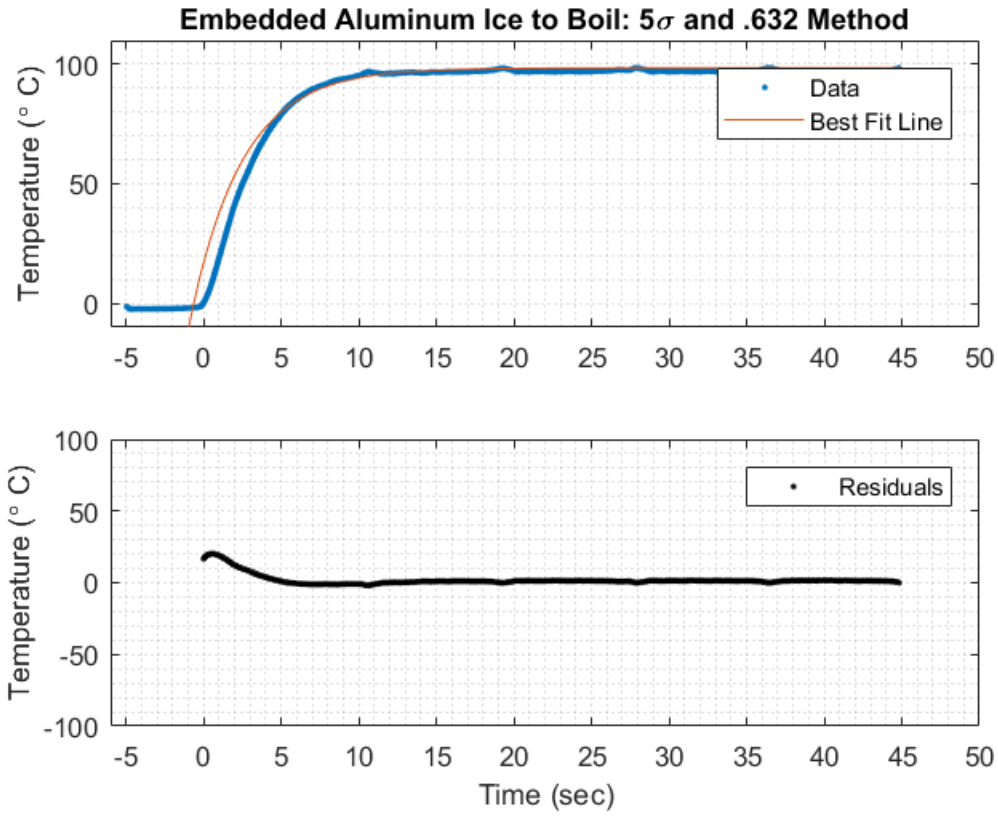
```
T_at_Tau_std_EIB_AL=Temp_start_std_EIB_AL + .632 * (EIB_AL_temp_final
 - Temp_start_std_EIB_AL);
for i=1:length(new_time_start_std_EIB_AL)
    if EIB_AL_Temp_smooth(i)>T_at_Tau_std_EIB_AL
        Tau_std_value_EIB_AL=new_time_start_std_EIB_AL(i);
        break
    end
end
for i=1:length(new_time_start_std_EIB_AL)
Temp_tau_std_value_EIB_AL(i)=EIB_AL_temp_final-((EIB_AL_temp_final-
Temp_start_poly_EIB_AL)*exp(-new_time_start_std_EIB_AL(i)/
Tau_std_value_EIB_AL));
end
subplot(2,1,1)
plot(new_time_start_std_EIB_AL, EIB_AL_Temp_smooth, '.')
title('Embedded Aluminum Ice to Boil: 5\sigma and .632 Method')
ylabel('Temperature (\circ C)')
grid minor
axis([-6, 50, -10, 110])
hold on
plot(new_time_start_std_EIB_AL, Temp_tau_std_value_EIB_AL)
legend('Data', 'Best Fit Line')
%Residuals
subplot(2,1,2)
residual_value_std_EIB_AL=Temp_tau_std_value_EIB_AL-
EIB_AL_Temp_smooth;
plot(new_time_start_std_EIB_AL(position_start_std_EIB_AL:length(residual_value_std
ylabel('Temperature (\circ C)')
xlabel('Time (sec)')
axis([-6, 50, -100, 100])
legend ('Residuals')
grid minor

figure
T_at_Tau_poly_EIB_AL=Temp_start_poly_EIB_AL + .632 *
 (EIB_AL_temp_final - Temp_start_poly_EIB_AL);
for i=1:length(new_time_start_poly_EIB_AL)
    if EIB_AL_Temp_smooth(i)>T_at_Tau_poly_EIB_AL
        Tau_poly_value_EIB_AL=new_time_start_poly_EIB_AL(i);
        break
    end
end
for i=1:length(new_time_start_poly_EIB_AL)
Temp_tau_poly_value_EIB_AL(i)=EIB_AL_temp_final-((EIB_AL_temp_final-
Temp_start_poly_EIB_AL)*exp(-new_time_start_poly_EIB_AL(i)/
Tau_poly_value_EIB_AL));
end
subplot(2,1,1)
plot(new_time_start_poly_EIB_AL, EIB_AL_Temp_smooth, '.')
hold on
plot(new_time_start_poly_EIB_AL, Temp_tau_poly_value_EIB_AL)
ylabel('Temperature (\circ C)')
title('Embedded Aluminum Ice to Boil: Max Slope and .632 Method')
axis([-6, 50, -10, 110])
```

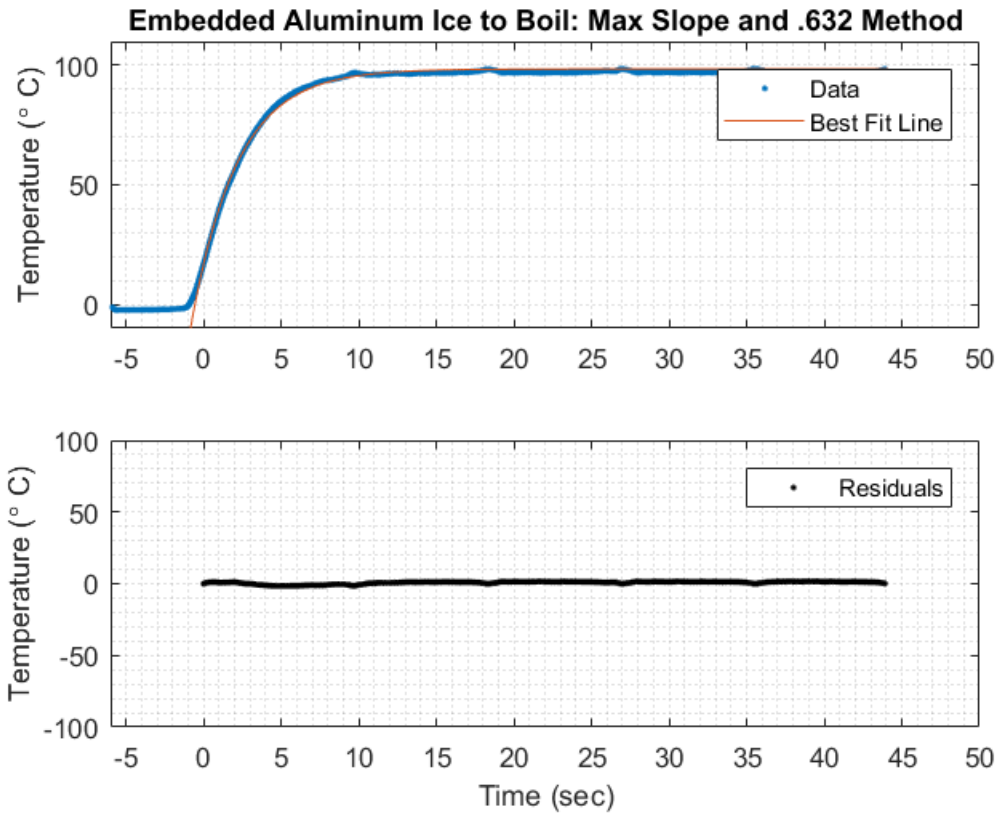
```
grid minor
legend('Data', 'Best Fit Line')
%Residuals
subplot(2,1,2)
residual_value_poly_EIB_AL=Temp_tau_poly_value_EIB_AL-
EIB_AL_Temp_smooth;
plot(new_time_start_poly_EIB_AL(start_poly_EIB_AL:length(residual_value_poly_EIB_A
ylabel('Temperature (\circ C)')
xlabel('Time (sec)')
grid minor
legend ('Residuals')
axis([-6, 50, -100, 100])
figure
```




Embedded Aluminum Ice to Boil: 5σ and Γ Method
ln(Γ)
Data
Best Fit Line
Time (sec)
Temperature (° C)
Data
Best Fit Line
Time (sec)
Temperature (° C)
Residuals
Time (sec)


Embedded Aluminum Ice to Boil: Max Slope and Γ Method
ln(Γ)
Data
Best Fit Line
Time (sec)
Temperature (° C)
Data
Best Fit Line
Time (sec)
Temperature (° C)
Residuals
Time (sec)


Embedded Aluminum Ice to Boil: 5σ and .632 Method
Temperature (° C)
Data
Best Fit Line
Residuals
Time (sec)


Embedded Aluminum Ice to Boil: Max Slope and .632 Method
Temperature (° C)
Data
Best Fit Line
Residuals
Time (sec)

EBI_AL

```
for i=1:length(EBI_AL_volt)
    EBI_AL_Temp(i)=interp1(x,y,(EBI_AL_volt(i)*10^3));
end
%smooth data to make anaylsis easier
span=40;
window=ones(span,1)/span;
EBI_AL_Temp_smooth=conv(EBI_AL_Temp,window,'same');

% filter off curving abnormalities at beginning
for i=1:300
        if EBI_AL_Temp_smooth(i)<96.7
            EBI_AL_Temp_smooth(i)=NaN;
    end
            if EBI_AL_Temp_smooth(i)>96.7
                break
            end
end

plot(EBI_AL_time,EBI_AL_Temp_smooth, '.')
hold on
ylabel('Temperature (\circ C)')
xlabel('Time (s)')
title('Embedded Aluminum Boil to Ice (EBI AL)')
```

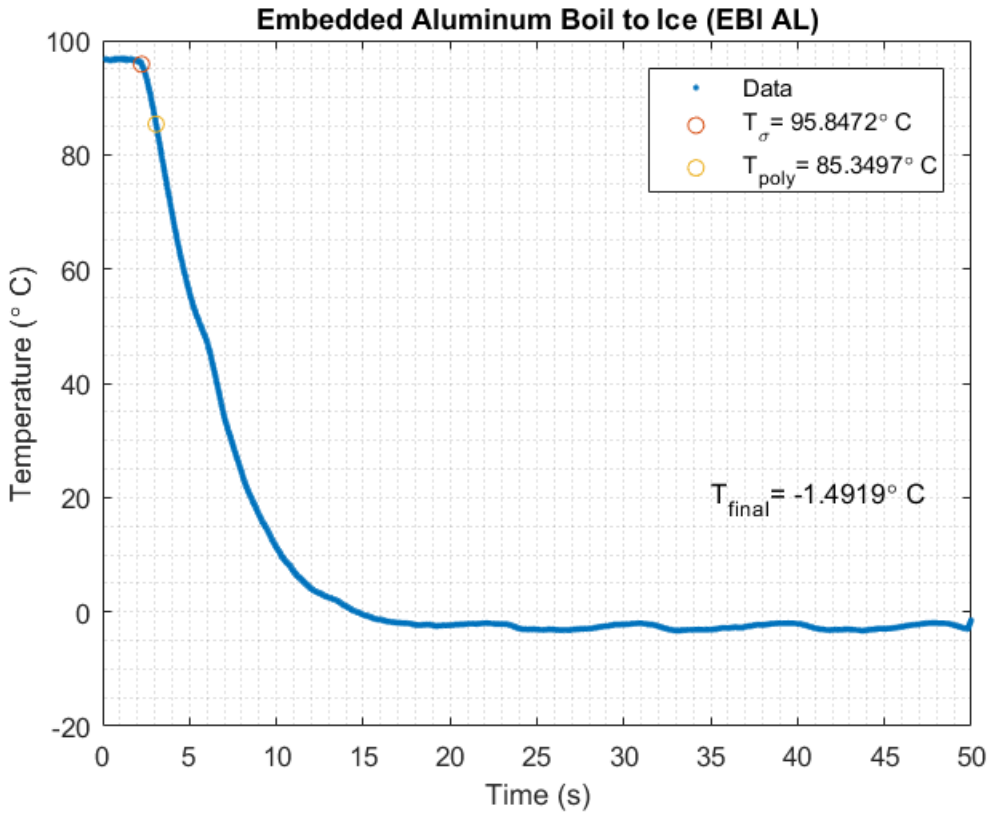
```
grid minor
deviation_EBI_AL=std(EBI_AL_Temp_smooth(50:220));
mean_EBI_AL=mean(EBI_AL_Temp_smooth(50:220));
for i=100:length(EBI_AL_Temp_smooth)
    if EBI_AL_Temp_smooth(i)<mean_EBI_AL-5*deviation_EBI_AL
        position_start_std_EBI_AL=i;
        break
    end
end

time_start_std_EBI_AL=EBI_AL_time(position_start_std_EBI_AL);
Temp_start_std_EBI_AL=EBI_AL_Temp_smooth(position_start_std_EBI_AL);
plot(time_start_std_EBI_AL,Temp_start_std_EBI_AL,'o')
%Find Ti with polyfit method
start_poly_EBI_AL=1;
for i=100:4000
    EBI_AL_poly(i,:)=polyfit(EBI_AL_time(i:i+70)',
 EBI_AL_Temp_smooth(i:i+70), 1);
    if abs(EBI_AL_poly(i))>abs(EBI_AL_poly(start_poly_EBI_AL))
        start_poly_EBI_AL=i;
    end
end


plot(EBI_AL_time(start_poly_EBI_AL)',EBI_AL_Temp_smooth(start_poly_EBI_AL), 'o')
EBI_AL_temp_final=EBI_AL_Temp_smooth(end);
text(35,20,['T_f_i_n_a_l= ' num2str(EBI_AL_temp_final) '\circ C'])
legend('Data', ['T_\sigma= ' num2str(Temp_start_std_EBI_AL) '\circ
 C'], ['T_p_o_l_y= '
 num2str(EBI_AL_Temp_smooth(start_poly_EBI_AL)) '\circ
 C'], 'Location', 'northeast')
figure
% %Find Gamma std
new_time_start_std_EBI_AL=EBI_AL_time-
EBI_AL_time(position_start_std_EBI_AL);
subplot(3,1,1)
for i=23:length(new_time_start_std_EBI_AL)
    gamma_std_EBI_AL(i)=(EBI_AL_temp_final-EBI_AL_Temp_smooth(i))/
(EBI_AL_temp_final-Temp_start_std_EBI_AL);
    if gamma_std_EBI_AL(i)<=0
        break
    end
    ln_gamma_std_EBI_AL(i)=log(gamma_std_EBI_AL(i));
end
new_time_std_range_EBI_AL=new_time_start_std_EBI_AL(1:length(ln_gamma_std_EBI_AL))
plot(new_time_std_range_EBI_AL, ln_gamma_std_EBI_AL, '.')
hold on
find_tau_std_EBI_AL=polyfit(new_time_std_range_EBI_AL(position_start_std_EBI_AL:le
 ln_gamma_std_EBI_AL(position_start_std_EBI_AL:length(ln_gamma_std_EBI_AL)) ,
 1);% find tau with stdfit
tau_std_EBI_AL=-1/find_tau_std_EBI_AL(1);
fit_tau_std_EBI_AL=find_tau_std_EBI_AL(1).*new_time_std_range_EBI_AL;
plot(new_time_std_range_EBI_AL, fit_tau_std_EBI_AL);
```

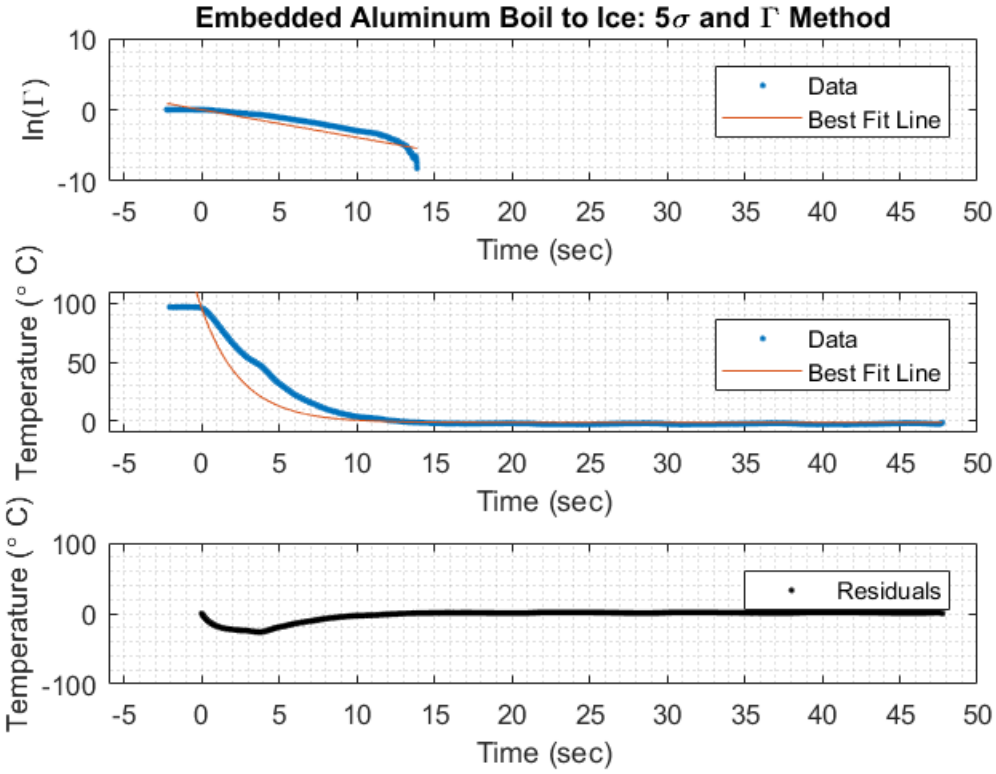
```
ylabel('ln(\Gamma)')
axis([-6, 50, -10, 10])
xlabel('Time (sec)')
legend('Data', 'Best Fit Line')
% % text(2.8, -8, ['\tau= ' num2str(tau_std_EBI_AL) ' sec'])
grid minor
title('Embedded Aluminum Boil to Ice: 5\sigma and \Gamma Method')
subplot(3,1,2)
% Estimate plot with time constant
for i=1:length(EBI_AL_time)
    Temp_tau_std_EBI_AL(i)=EBI_AL_temp_final-((EBI_AL_temp_final-
Temp_start_std_EBI_AL)*exp(-new_time_start_std_EBI_AL(i)/
tau_std_EBI_AL));
end



plot(new_time_start_std_EBI_AL, EBI_AL_Temp_smooth, '.')
hold on
plot(new_time_start_std_EBI_AL, Temp_tau_std_EBI_AL)
legend('Data', 'Best Fit Line')
axis([-6, 50, -10, 110])
ylabel('Temperature (\circ C)')
xlabel('Time (sec)')
grid minor
% Residuals
subplot(3,1,3)


 residual_std_EBI_AL=Temp_tau_std_EBI_AL(position_start_std_EBI_AL:end)-
EBI_AL_Temp_smooth(position_start_std_EBI_AL:end);

plot(new_time_start_std_EBI_AL(position_start_std_EBI_AL:end),
 residual_std_EBI_AL, 'bla.')
legend('Residuals')
grid minor
ylabel('Temperature (\circ C)')
axis([-6, 50, -100, 100])
xlabel('Time (sec)')
figure
% Find Gamma poly
new_time_start_poly_EBI_AL=EBI_AL_time-EBI_AL_time(start_poly_EBI_AL);
subplot(3,1,1)
Temp_start_poly_EBI_AL=(EBI_AL_Temp(start_poly_EBI_AL));
for i=23:length(new_time_start_poly_EBI_AL)
    gamma_poly_EBI_AL(i)=(EBI_AL_temp_final-EBI_AL_Temp_smooth(i))/
(EBI_AL_temp_final-Temp_start_poly_EBI_AL);
    if gamma_poly_EBI_AL(i)<=0
        break
    end
    ln_gamma_poly_EBI_AL(i)=log(gamma_poly_EBI_AL(i));
end
new_time_poly_range_EBI_AL=new_time_start_poly_EBI_AL(1:length(ln_gamma_poly_EBI_A
plot(new_time_poly_range_EBI_AL, ln_gamma_poly_EBI_AL, '.')
```

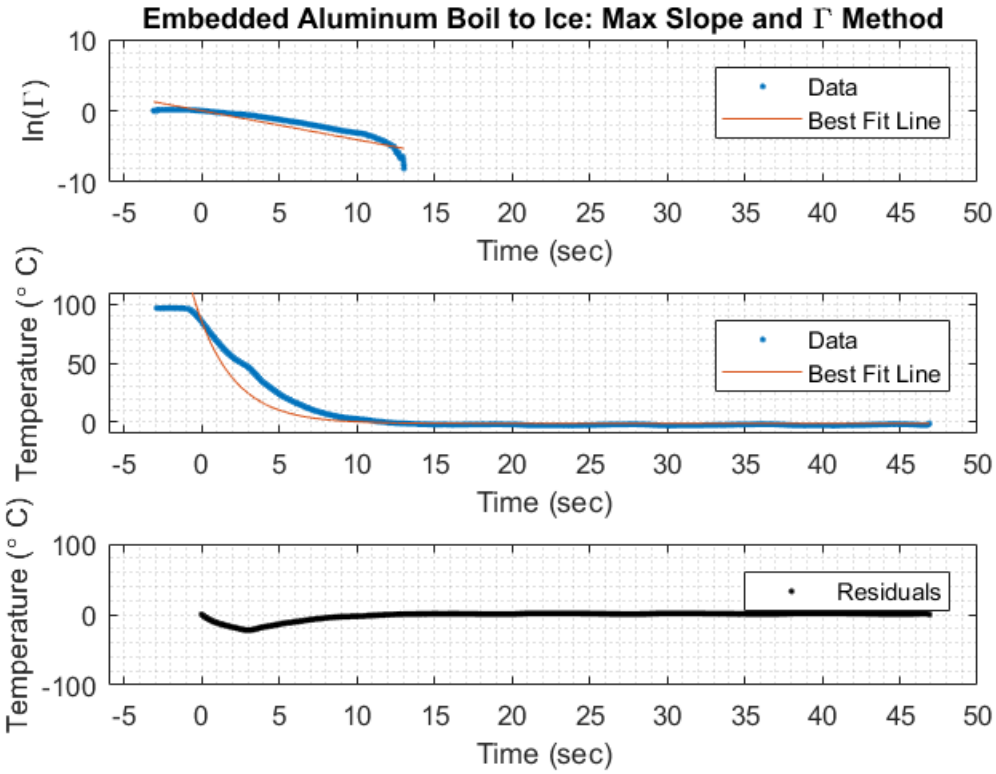
```
hold on
find_tau_poly_EBI_AL=polyfit(new_time_poly_range_EBI_AL(start_poly_EBI_AL:length(l
 ln_gamma_poly_EBI_AL(start_poly_EBI_AL:length(ln_gamma_poly_EBI_AL)) ,
 1);% find tau with polyfit
tau_poly_EBI_AL=-1/find_tau_poly_EBI_AL(1);
fit_tau_poly_EBI_AL=find_tau_poly_EBI_AL(1).*new_time_poly_range_EBI_AL;
plot(new_time_poly_range_EBI_AL, fit_tau_poly_EBI_AL);

ylabel('ln(\Gamma)')
xlabel('Time (sec)')
axis([-6, 50, -10, 10])
% % text(2.8, -8, ['\tau= ' num2str(tau_poly_EBI_AL) ' sec'])
grid minor
legend('Data', 'Best Fit Line')
title('Embedded Aluminum Boil to Ice: Max Slope and \Gamma Method')
subplot(3,1,2)
% Estimate plot with time constant
for i=1:length(EBI_AL_time)
    Temp_tau_poly_EBI_AL(i)=EBI_AL_temp_final-((EBI_AL_temp_final-
Temp_start_poly_EBI_AL)*exp(-new_time_start_poly_EBI_AL(i)/
tau_poly_EBI_AL));
end



plot(new_time_start_poly_EBI_AL, EBI_AL_Temp_smooth, '.')

hold on
plot(new_time_start_poly_EBI_AL, Temp_tau_poly_EBI_AL)
legend('Data', 'Best Fit Line')
axis([-6, 50, -10, 110])
ylabel('Temperature (\circ C)')
xlabel('Time (sec)')
grid minor
% Residuals
subplot(3,1,3)

residual_poly_EBI_AL=Temp_tau_poly_EBI_AL(start_poly_EBI_AL:end)-
EBI_AL_Temp_smooth(start_poly_EBI_AL:end);

plot(new_time_start_poly_EBI_AL(start_poly_EBI_AL:end),
 residual_poly_EBI_AL, 'bla.')
legend('Residuals')
grid minor
ylabel('Temperature (\circ C)')
axis([-6, 50, -100, 100])
xlabel('Time (sec)')
figure

%Calculate Tau with .632       STD
T_at_Tau_std_EBI_AL=Temp_start_std_EBI_AL + .632 * (EBI_AL_temp_final
 - Temp_start_std_EBI_AL);
for i=1:length(new_time_start_std_EBI_AL)
    if EBI_AL_Temp_smooth(i)<T_at_Tau_std_EBI_AL
```

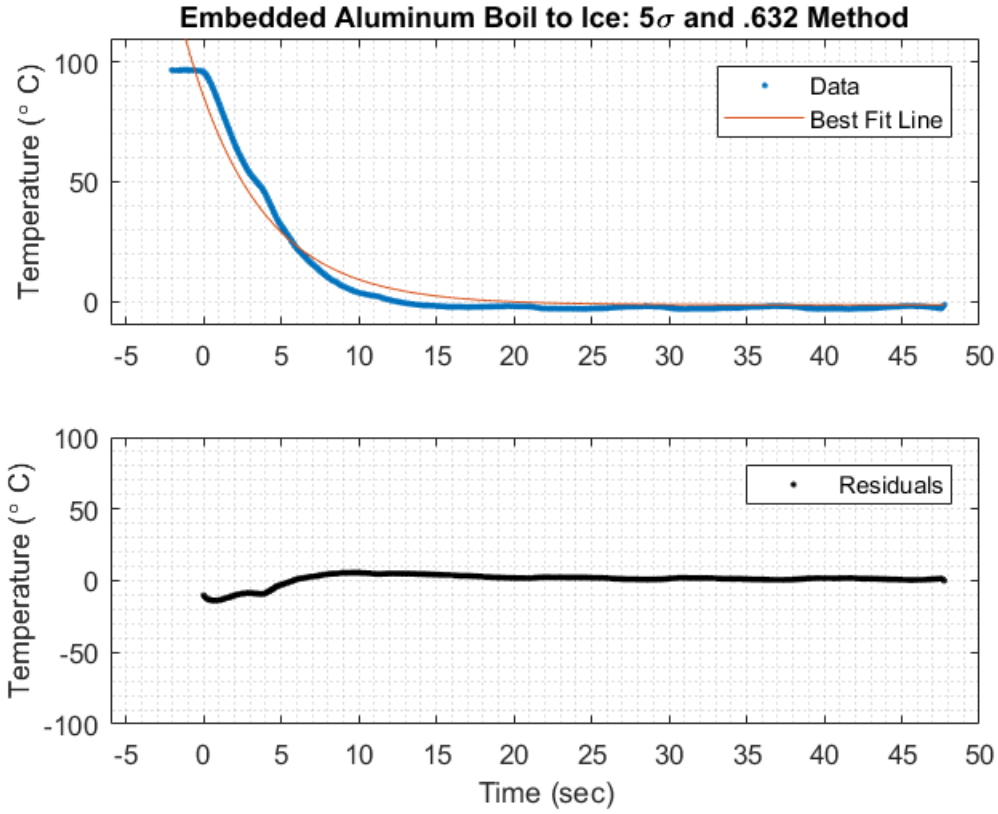
```
        Tau_std_value_EBI_AL=new_time_start_std_EBI_AL(i);
        break
    end
end
for i=1:length(new_time_start_std_EBI_AL)
Temp_tau_std_value_EBI_AL(i)=EBI_AL_temp_final-((EBI_AL_temp_final-
Temp_start_poly_EBI_AL)*exp(-new_time_start_std_EBI_AL(i)/
Tau_std_value_EBI_AL));
end
subplot(2,1,1)
plot(new_time_start_std_EBI_AL, EBI_AL_Temp_smooth, '.')
title('Embedded Aluminum Boil to Ice: 5\sigma and .632 Method')
ylabel('Temperature (\circ C)')
grid minor

axis([-6, 50, -10, 110])
hold on
plot(new_time_start_std_EBI_AL, Temp_tau_std_value_EBI_AL)
legend('Data', 'Best Fit Line')
%Residuals
subplot(2,1,2)
residual_value_std_EBI_AL=Temp_tau_std_value_EBI_AL-
EBI_AL_Temp_smooth;
plot(new_time_start_std_EBI_AL(position_start_std_EBI_AL:length(residual_value_std
ylabel('Temperature (\circ C)')
xlabel('Time (sec)')
legend('Residuals')
axis([-6, 50, -100, 100])
grid minor

figure
T_at_Tau_poly_EBI_AL=Temp_start_poly_EBI_AL + .632 *
 (EBI_AL_temp_final - Temp_start_poly_EBI_AL);
for i=1:length(new_time_start_poly_EBI_AL)
    if EBI_AL_Temp_smooth(i)<T_at_Tau_poly_EBI_AL
        Tau_poly_value_EBI_AL=new_time_start_poly_EBI_AL(i);
        break
    end
end
for i=1:length(new_time_start_poly_EBI_AL)
Temp_tau_poly_value_EBI_AL(i)=EBI_AL_temp_final-((EBI_AL_temp_final-
Temp_start_poly_EBI_AL)*exp(-new_time_start_poly_EBI_AL(i)/
Tau_poly_value_EBI_AL));
end
subplot(2,1,1)

plot(new_time_start_poly_EBI_AL, EBI_AL_Temp_smooth, '.')
hold on
plot(new_time_start_poly_EBI_AL, Temp_tau_poly_value_EBI_AL)
ylabel('Temperature (\circ C)')
title('Embedded Aluminum Boil to Ice: Max Slope and .632 Method')
axis([-6, 50, -10, 110])
legend('Data', 'Best Fit Line')
grid minor
```

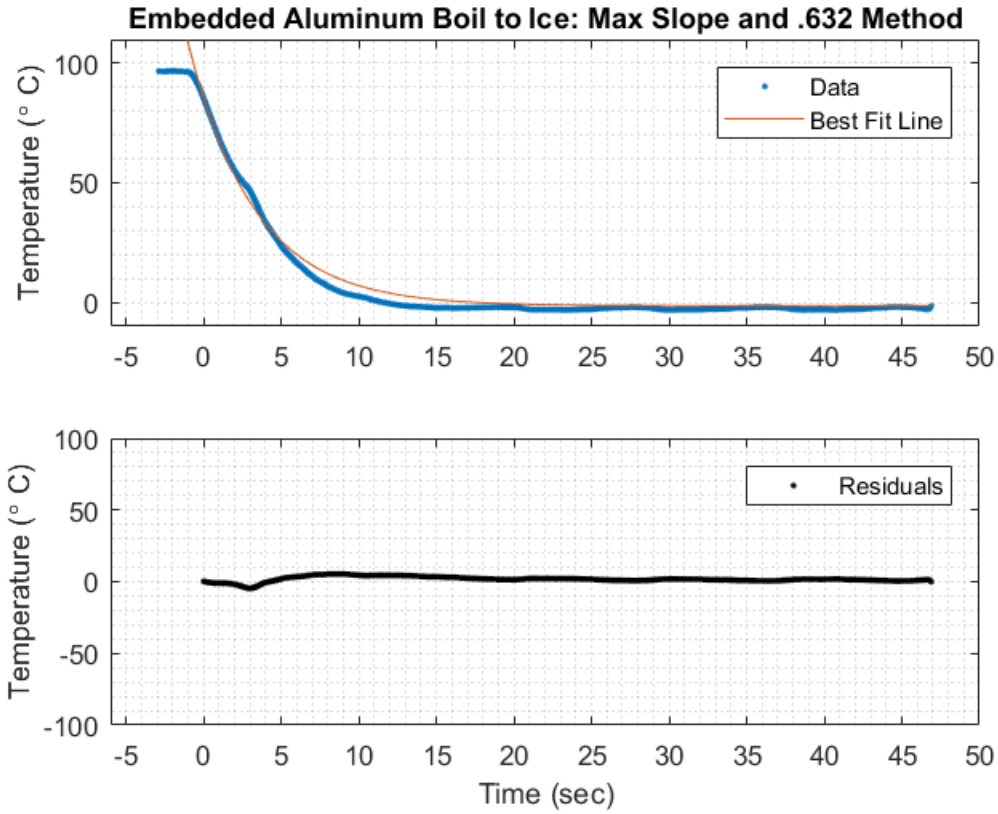
```
%Residuals
subplot(2,1,2)
residual_value_poly_EBI_AL=Temp_tau_poly_value_EBI_AL-
EBI_AL_Temp_smooth;
plot(new_time_start_poly_EBI_AL(start_poly_EBI_AL:length(residual_value_poly_EBI_A
ylabel('Temperature (\circ C)')
xlabel('Time (sec)')
grid minor
legend ('Residuals')
axis([-6, 50, -100, 100])
figure
```




Embedded Aluminum Boil to Ice: 5σ and Γ Method
ln(Γ)
Data
Best Fit Line
Time (sec)
Temperature (° C)
Data
Best Fit Line
Time (sec)
Temperature (° C)
Residuals
Time (sec)


Embedded Aluminum Boil to Ice: Max Slope and Γ Method
ln(Γ)
Data
Best Fit Line
Time (sec)
Temperature (° C)
Data
Best Fit Line
Time (sec)
Temperature (° C)
Residuals
Time (sec)


Embedded Aluminum Boil to Ice: 5σ and .632 Method
Temperature (° C)
Data
Best Fit Line
Residuals
Time (sec)


Embedded Aluminum Boil to Ice: Max Slope and .632 Method
Temperature (° C)
Data
Best Fit Line
Residuals
Time (sec)

Plots of Residuals

Plots of std residuals found with gamma

```
plot(new_time_start_std_BBI(position_start_std_BBI:end),
 residual_std_BBI)
hold on
plot(new_time_start_std_BIB(position_start_std_BIB:end),
 residual_std_BIB)
plot(new_time_start_std_EIB_SS(position_start_std_EIB_SS:end),
 residual_std_EIB_SS)
plot(new_time_start_std_EBI_SS(position_start_std_EBI_SS:end),
 residual_std_EBI_SS)
plot(new_time_start_std_EIB_AL(position_start_std_EIB_AL:end),
 residual_std_EIB_AL)
plot(new_time_start_std_EBI_AL(position_start_std_EBI_AL:end),
 residual_std_EBI_AL)

% Find Syx of each function
% syx=(sum((Temp_Thermocouple-fit_2).^2)/nu).^.5;
m_fits=1;
nu_BBI=length(BBI_Temp_smooth)-(m_fits+1);
sum_length_BBI_std_gamma=position_start_std_BBI:1:length(Temp_tau_std_BBI);
syx_gamma_std_BBI=(sum((BBI_Temp_smooth(sum_length_BBI_std_gamma)-
Temp_tau_std_BBI(sum_length_BBI_std_gamma)).^2)/nu_BBI).^.5;
syx_gamma_std_BBI_string=num2str(syx_gamma_std_BBI);
```

```
text(30, -5, ['S_Y_X_,_B_B_I = ' syx_gamma_std_BBI_string , ' \circ
 C'])

nu_BIB=length(BIB_Temp_smooth)-(m_fits+1);
sum_length_BIB_std_gamma=position_start_std_BIB:1:length(Temp_tau_std_BIB)-50;
syx_gamma_std_BIB=(sum((BIB_Temp_smooth(sum_length_BIB_std_gamma)-
Temp_tau_std_BIB(sum_length_BIB_std_gamma)).^2)/nu_BIB).^.5;
syx_gamma_std_BIB_string=num2str(syx_gamma_std_BIB);
text(30, -7, ['S_Y_X_,_B_I_B = ' syx_gamma_std_BIB_string , ' \circ
 C'])

nu_EIB_SS=length(EIB_SS_Temp_smooth)-(m_fits+1);
sum_length_EIB_SS_std_gamma=position_start_std_EIB_SS:1:length(Temp_tau_std_EIB_SS
syx_gamma_std_EIB_SS=(sum((EIB_SS_Temp_smooth(sum_length_EIB_SS_std_gamma)-
Temp_tau_std_EIB_SS(sum_length_EIB_SS_std_gamma)).^2)/nu_EIB_SS).^.5;
syx_gamma_std_EIB_SS_string=num2str(syx_gamma_std_EIB_SS);
text(30, -9, ['S_Y_X_,_E_I_B_ _S_S = ' syx_gamma_std_EIB_SS_string , '
 \circ C'])

nu_EBI_SS=length(EBI_SS_Temp_smooth)-(m_fits+1);
sum_length_EBI_SS_std_gamma=position_start_std_EBI_SS:1:length(Temp_tau_std_EBI_SS
syx_gamma_std_EBI_SS=(sum((EBI_SS_Temp_smooth(sum_length_EBI_SS_std_gamma)-
Temp_tau_std_EBI_SS(sum_length_EBI_SS_std_gamma)).^2)/nu_EBI_SS).^.5;
syx_gamma_std_EBI_SS_string=num2str(syx_gamma_std_BBI);
syx_gamma_std_EBI_SS_string=num2str(syx_gamma_std_EBI_SS);
text(30, -11, ['S_Y_X_,_E_I_B_ _S_S = '
 syx_gamma_std_EBI_SS_string , ' \circ C'])

nu_EIB_AL=length(EIB_AL_Temp_smooth)-(m_fits+1);
sum_length_EIB_AL_std_gamma=position_start_std_EIB_AL:1:length(Temp_tau_std_EIB_AL
syx_gamma_std_EIB_AL=(sum((EIB_AL_Temp_smooth(sum_length_EIB_AL_std_gamma)-
Temp_tau_std_EIB_AL(sum_length_EIB_AL_std_gamma)).^2)/nu_EIB_AL).^.5;
syx_gamma_std_EIB_AL_string=num2str(syx_gamma_std_BBI);
syx_gamma_std_EIB_AL_string=num2str(syx_gamma_std_EIB_AL);
text(30, -13, ['S_Y_X_,_E_I_B_ _A_L = '
 syx_gamma_std_EIB_AL_string , ' \circ C'])

nu_EBI_AL=length(EBI_AL_Temp_smooth)-(m_fits+1);
sum_length_EBI_AL_std_gamma=position_start_std_EBI_AL:1:length(Temp_tau_std_EBI_AL
syx_gamma_std_EBI_AL=(sum((EBI_AL_Temp_smooth(sum_length_EBI_AL_std_gamma)-
Temp_tau_std_EBI_AL(sum_length_EBI_AL_std_gamma)).^2)/nu_EBI_AL).^.5;
syx_gamma_std_EBI_AL_string=num2str(syx_gamma_std_BBI);
syx_gamma_std_EBI_AL_string=num2str(syx_gamma_std_EBI_AL);
text(30, -15, ['S_Y_X_,_E_B_I_ _A_L = '
 syx_gamma_std_EBI_AL_string , ' \circ C'])



title('Residuals of Standard Deviation Method found with Gamma')
ylabel('Temperature (\circ C)')
xlabel('Time (sec)')
legend('BBI', 'BIB', 'EIB SS', 'EBI SS', 'EIB AL', 'EBI AL')
ylim([-20, 20])
grid minor
```

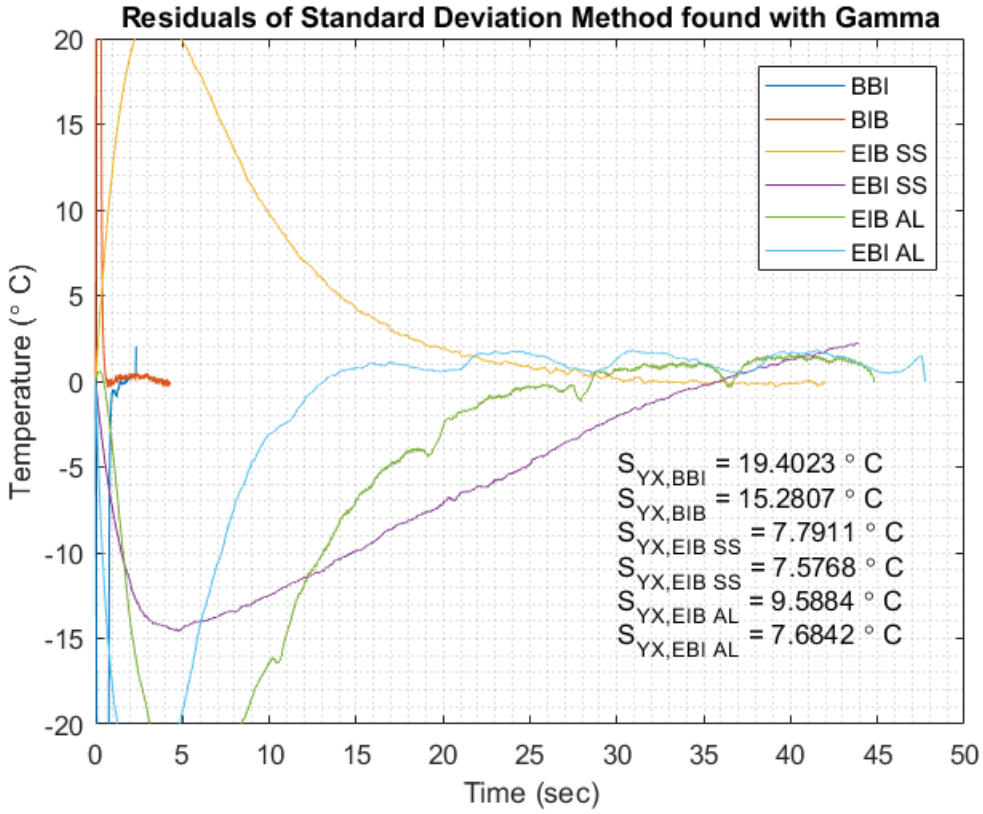
```
figure
%%%%%%%%%%%%%%%%%%%%%%%%%%%% Poly method with Gamma
plot(new_time_start_poly_BBI(start_poly_BBI:end), residual_poly_BBI)
hold on
plot(new_time_start_poly_BIB(start_poly_BIB:end), residual_poly_BIB)
plot(new_time_start_poly_EIB_SS(start_poly_EIB_SS:end),
 residual_poly_EIB_SS)
plot(new_time_start_poly_EBI_SS(start_poly_EBI_SS:end),
 residual_poly_EBI_SS)
plot(new_time_start_poly_EIB_AL(start_poly_EIB_AL:end),
 residual_poly_EIB_AL)
plot(new_time_start_poly_EBI_AL(start_poly_EBI_AL:end),
 residual_poly_EBI_AL)



nu_BBI=length(BBI_Temp_smooth)-(m_fits+1);
sum_length_BBI_poly_gamma=start_poly_BBI:1:length(Temp_tau_poly_BBI);
syx_gamma_poly_BBI=(sum((BBI_Temp_smooth(sum_length_BBI_poly_gamma)-
Temp_tau_poly_BBI(sum_length_BBI_poly_gamma)).^2)/nu_BBI).^.5;
syx_gamma_poly_BBI_string=num2str(syx_gamma_poly_BBI);
text(30, -5, ['S_Y_X_,_B_B_I = ' syx_gamma_poly_BBI_string , ' \circ
 C'])

nu_BIB=length(BIB_Temp_smooth)-(m_fits+1);
sum_length_BIB_poly_gamma=start_poly_BIB:1:length(Temp_tau_poly_BIB)-50;
syx_gamma_poly_BIB=(sum((BIB_Temp_smooth(sum_length_BIB_poly_gamma)-
Temp_tau_poly_BIB(sum_length_BIB_poly_gamma)).^2)/nu_BIB).^.5;
syx_gamma_poly_BIB_string=num2str(syx_gamma_poly_BIB);
text(30, -7, ['S_Y_X_,_B_I_B = ' syx_gamma_poly_BIB_string , ' \circ
 C'])

nu_EIB_SS=length(EIB_SS_Temp_smooth)-(m_fits+1);
sum_length_EIB_SS_poly_gamma=start_poly_EIB_SS:1:length(Temp_tau_poly_EIB_SS)-20;
syx_gamma_poly_EIB_SS=(sum((EIB_SS_Temp_smooth(sum_length_EIB_SS_poly_gamma)-
Temp_tau_poly_EIB_SS(sum_length_EIB_SS_poly_gamma)).^2)/
nu_EIB_SS).^.5;
syx_gamma_poly_EIB_SS_string=num2str(syx_gamma_poly_EIB_SS);
text(30, -9, ['S_Y_X_,_E_I_B_ _S_S = '
 syx_gamma_poly_EIB_SS_string , ' \circ C'])

nu_EBI_SS=length(EBI_SS_Temp_smooth)-(m_fits+1);
sum_length_EBI_SS_poly_gamma=start_poly_EBI_SS:1:length(Temp_tau_poly_EBI_SS)-25;
syx_gamma_poly_EBI_SS=(sum((EBI_SS_Temp_smooth(sum_length_EBI_SS_poly_gamma)-
Temp_tau_poly_EBI_SS(sum_length_EBI_SS_poly_gamma)).^2)/
nu_EBI_SS).^.5;
syx_gamma_poly_EBI_SS_string=num2str(syx_gamma_poly_BBI);
syx_gamma_poly_EBI_SS_string=num2str(syx_gamma_poly_EBI_SS);
text(30, -11, ['S_Y_X_,_E_I_B_ _S_S = '
 syx_gamma_poly_EBI_SS_string , ' \circ C'])

nu_EIB_AL=length(EIB_AL_Temp_smooth)-(m_fits+1);
sum_length_EIB_AL_poly_gamma=start_poly_EIB_AL:1:length(Temp_tau_poly_EIB_AL)-25;
```

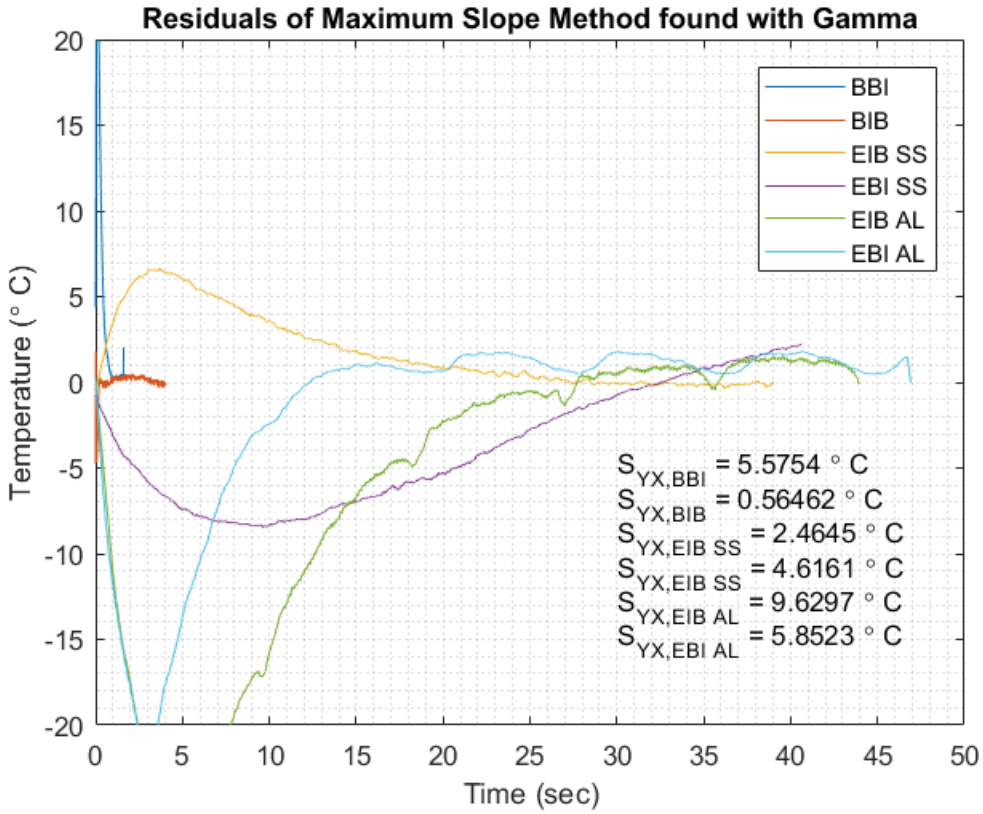
```
syx_gamma_poly_EIB_AL=(sum((EIB_AL_Temp_smooth(sum_length_EIB_AL_poly_gamma)-
Temp_tau_poly_EIB_AL(sum_length_EIB_AL_poly_gamma)).^2)/
nu_EIB_AL).^.5;
syx_gamma_poly_EIB_AL_string=num2str(syx_gamma_poly_BBI);
syx_gamma_poly_EIB_AL_string=num2str(syx_gamma_poly_EIB_AL);
text(30, -13, ['S_Y_X_,_E_I_B_ _A_L = '
 syx_gamma_poly_EIB_AL_string , ' \circ C'])

nu_EBI_AL=length(EBI_AL_Temp_smooth)-(m_fits+1);
sum_length_EBI_AL_poly_gamma=start_poly_EBI_AL:1:length(Temp_tau_poly_EBI_AL)-25;
syx_gamma_poly_EBI_AL=(sum((EBI_AL_Temp_smooth(sum_length_EBI_AL_poly_gamma)-
Temp_tau_poly_EBI_AL(sum_length_EBI_AL_poly_gamma)).^2)/
nu_EBI_AL).^.5;
syx_gamma_poly_EBI_AL_string=num2str(syx_gamma_poly_BBI);
syx_gamma_poly_EBI_AL_string=num2str(syx_gamma_poly_EBI_AL);
text(30, -15, ['S_Y_X_,_E_B_I_ _A_L = '
 syx_gamma_poly_EBI_AL_string , ' \circ C'])


title('Residuals of Maximum Slope Method found with Gamma')
ylabel('Temperature (\circ C)')
xlabel('Time (sec)')
legend('BBI', 'BIB', 'EIB SS', 'EBI SS', 'EIB AL', 'EBI AL')
ylim([-20, 20])
grid minor
figure


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%STD method with value equation
plot(new_time_start_std_BBI(position_start_std_BBI:length(residual_value_std_BBI))
hold on
plot(new_time_start_std_BIB(position_start_std_BIB:length(residual_value_std_BIB))
plot(new_time_start_std_EIB_SS(position_start_std_EIB_SS:length(residual_value_std
plot(new_time_start_std_EBI_SS(position_start_std_EBI_SS:length(residual_value_std
plot(new_time_start_std_EIB_AL(position_start_std_EIB_AL:length(residual_value_std
plot(new_time_start_std_EBI_AL(position_start_std_EBI_AL:length(residual_value_std

% Syx for value method

sum_length_std_value_BBI=position_start_std_BBI:length(residual_value_std_BBI);
nu_BBI=length(BBI_Temp_smooth)-(m_fits+1);
sum_length_BBI_std_value=position_start_std_BBI:1:length(Temp_tau_std_value_BBI);
syx_value_std_BBI=(sum((BBI_Temp_smooth(sum_length_std_value_BBI)-
Temp_tau_std_value_BBI(sum_length_std_value_BBI)).^2)/nu_BBI).^.5;
syx_value_std_BBI_string=num2str(syx_value_std_BBI);
text(30, -5, ['S_Y_X_,_B_B_I = ' syx_value_std_BBI_string , ' \circ
 C'])


sum_length_std_value_BIB=position_start_std_BIB:length(residual_value_std_BIB);
nu_BIB=length(BIB_Temp_smooth)-(m_fits+1);
sum_length_BIB_std_value=position_start_std_BIB:1:length(Temp_tau_std_value_BIB)-5
syx_value_std_BIB=(sum((BIB_Temp_smooth(sum_length_BIB_std_value)-
Temp_tau_std_value_BIB(sum_length_BIB_std_value)).^2)/nu_BIB).^.5;
```

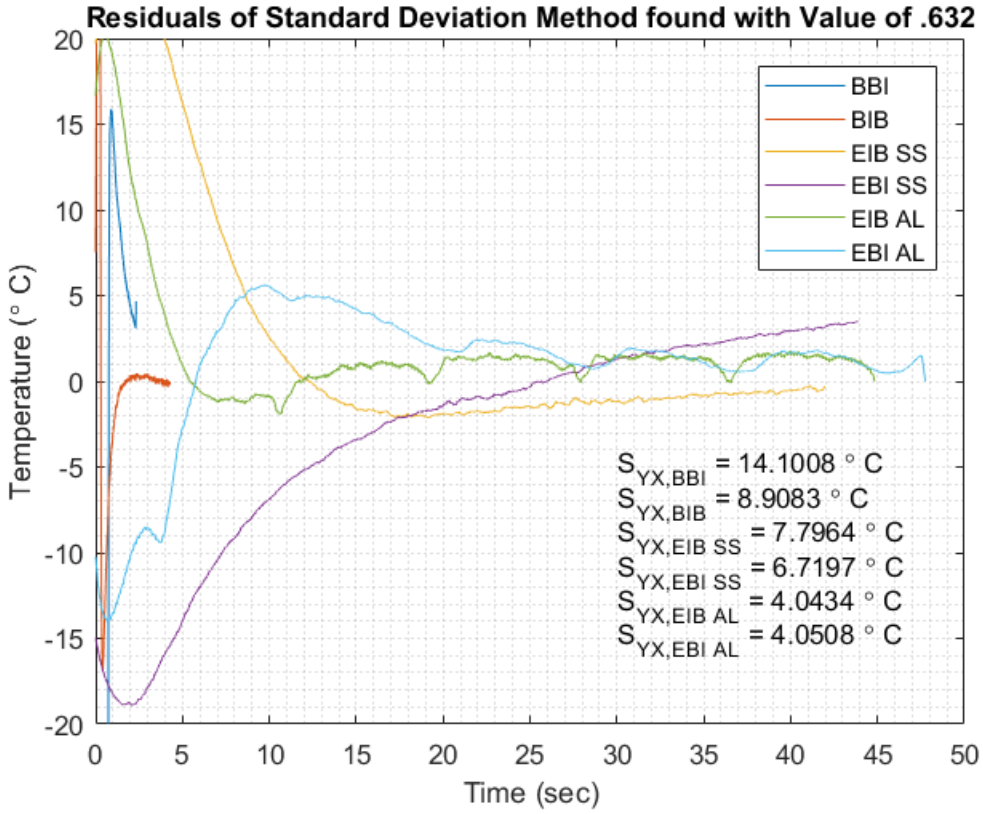
```
syx_value_std_BIB_string=num2str(syx_value_std_BIB);
text(30, -7, ['S_Y_X_,_B_I_B = ' syx_value_std_BIB_string , ' \circ
 C'])


sum_length_std_value_EIB_SS=position_start_std_EIB_SS:length(residual_value_std_EI
nu_EIB_SS=length(EIB_SS_Temp_smooth)-(m_fits+1);
sum_length_EIB_SS_std_value=position_start_std_EIB_SS:1:length(Temp_tau_std_value_
syx_value_std_EIB_SS=(sum((EIB_SS_Temp_smooth(sum_length_EIB_SS_std_value)-
Temp_tau_std_value_EIB_SS(sum_length_EIB_SS_std_value)).^2)/
nu_EIB_SS).^.5;
syx_value_std_EIB_SS_string=num2str(syx_value_std_EIB_SS);
text(30, -9, ['S_Y_X_,_E_I_B_ _S_S = ' syx_value_std_EIB_SS_string , '
 \circ C'])


sum_length_std_value_EBI_SS=position_start_std_EBI_SS:length(residual_value_std_EB
nu_EBI_SS=length(EBI_SS_Temp_smooth)-(m_fits+1);
sum_length_EBI_SS_std_value=position_start_std_EBI_SS:1:length(Temp_tau_std_value_
syx_value_std_EBI_SS=(sum((EBI_SS_Temp_smooth(sum_length_EBI_SS_std_value)-
Temp_tau_std_value_EBI_SS(sum_length_EBI_SS_std_value)).^2)/
nu_EBI_SS).^.5;
syx_value_std_EBI_SS_string=num2str(syx_value_std_EBI_SS);
text(30, -11, ['S_Y_X_,_E_B_I_ _S_S = '
 syx_value_std_EBI_SS_string , ' \circ C'])


sum_length_std_value_EIB_AL=position_start_std_EIB_AL:length(residual_value_std_EI
nu_EIB_AL=length(EIB_AL_Temp_smooth)-(m_fits+1);
sum_length_EIB_AL_std_value=position_start_std_EIB_AL:1:length(Temp_tau_std_value_
syx_value_std_EIB_AL=(sum((EIB_AL_Temp_smooth(sum_length_EIB_AL_std_value)-
Temp_tau_std_value_EIB_AL(sum_length_EIB_AL_std_value)).^2)/
nu_EIB_AL).^.5;
syx_value_std_EIB_AL_string=num2str(syx_value_std_EIB_AL);
text(30, -13, ['S_Y_X_,_E_I_B_ _A_L = '
 syx_value_std_EIB_AL_string , ' \circ C'])


sum_length_std_value_EBI_AL=position_start_std_EBI_AL:length(residual_value_std_EB
nu_EBI_AL=length(EBI_AL_Temp_smooth)-(m_fits+1);
sum_length_EBI_AL_std_value=position_start_std_EBI_AL:1:length(Temp_tau_std_value_
syx_value_std_EBI_AL=(sum((EBI_AL_Temp_smooth(sum_length_EBI_AL_std_value)-
Temp_tau_std_value_EBI_AL(sum_length_EBI_AL_std_value)).^2)/
nu_EBI_AL).^.5;
syx_value_std_EBI_AL_string=num2str(syx_value_std_EBI_AL);
text(30, -15, ['S_Y_X_,_E_B_I_ _A_L = '
 syx_value_std_EBI_AL_string , ' \circ C'])


title('Residuals of Standard Deviation Method found with Value
 of .632')
ylabel('Temperature (\circ C)')
xlabel('Time (sec)')
legend('BBI', 'BIB', 'EIB SS', 'EBI SS', 'EIB AL', 'EBI AL')
```

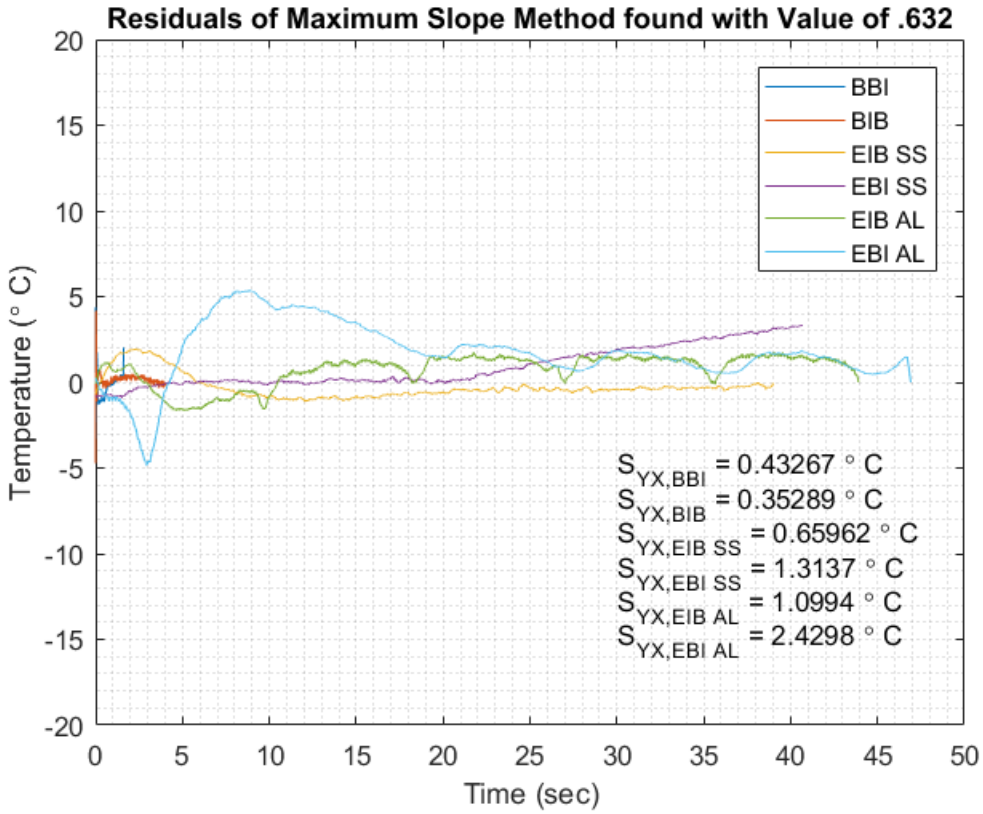
```
ylim([-20, 20])
grid minor
figure


%%%%%%%%Value method for Max Slope
plot(new_time_start_poly_BBI(start_poly_BBI:length(residual_value_poly_BBI)),resid
hold on
plot(new_time_start_poly_BIB(start_poly_BIB:length(residual_value_poly_BIB)),resid
plot(new_time_start_poly_EIB_SS(start_poly_EIB_SS:length(residual_value_poly_EIB_S
plot(new_time_start_poly_EBI_SS(start_poly_EBI_SS:length(residual_value_poly_EBI_S
plot(new_time_start_poly_EIB_AL(start_poly_EIB_AL:length(residual_value_poly_EIB_A
plot(new_time_start_poly_EBI_AL(start_poly_EBI_AL:length(residual_value_poly_EBI_A

%%%%%%%%%%%%%%%%%%%%%%%%%%%Find Syx Values for value method


sum_length_poly_value_BBI=start_poly_BBI:length(residual_value_poly_BBI);
nu_BBI=length(BBI_Temp_smooth)-(m_fits+1);
sum_length_BBI_poly_value=start_poly_BBI:1:length(Temp_tau_poly_value_BBI);
syx_value_poly_BBI=(sum((BBI_Temp_smooth(sum_length_poly_value_BBI)-
Temp_tau_poly_value_BBI(sum_length_poly_value_BBI)).^2)/nu_BBI).^.5;
syx_value_poly_BBI_string=num2str(syx_value_poly_BBI);
text(30, -5, ['S_Y_X_,_B_B_I = ' syx_value_poly_BBI_string , ' \circ
 C'])


sum_length_poly_value_BIB=start_poly_BIB:length(residual_value_poly_BIB);
nu_BIB=length(BIB_Temp_smooth)-(m_fits+1);
sum_length_BIB_poly_value=start_poly_BIB:1:length(Temp_tau_poly_value_BIB)-50;
syx_value_poly_BIB=(sum((BIB_Temp_smooth(sum_length_BIB_poly_value)-
Temp_tau_poly_value_BIB(sum_length_BIB_poly_value)).^2)/nu_BIB).^.5;
syx_value_poly_BIB_string=num2str(syx_value_poly_BIB);
text(30, -7, ['S_Y_X_,_B_I_B = ' syx_value_poly_BIB_string , ' \circ
 C'])


sum_length_poly_value_EIB_SS=start_poly_EIB_SS:length(residual_value_poly_EIB_SS);
nu_EIB_SS=length(EIB_SS_Temp_smooth)-(m_fits+1);
sum_length_EIB_SS_poly_value=start_poly_EIB_SS:1:length(Temp_tau_poly_value_EIB_SS
syx_value_poly_EIB_SS=(sum((EIB_SS_Temp_smooth(sum_length_EIB_SS_poly_value)-
Temp_tau_poly_value_EIB_SS(sum_length_EIB_SS_poly_value)).^2)/
nu_EIB_SS).^.5;
syx_value_poly_EIB_SS_string=num2str(syx_value_poly_EIB_SS);
text(30, -9, ['S_Y_X_,_E_I_B_ _S_S = '
 syx_value_poly_EIB_SS_string , ' \circ C'])


sum_length_poly_value_EBI_SS=start_poly_EBI_SS:length(residual_value_poly_EBI_SS);
nu_EBI_SS=length(EBI_SS_Temp_smooth)-(m_fits+1);
sum_length_EBI_SS_poly_value=start_poly_EBI_SS:1:length(Temp_tau_poly_value_EBI_SS
syx_value_poly_EBI_SS=(sum((EBI_SS_Temp_smooth(sum_length_EBI_SS_poly_value)-
Temp_tau_poly_value_EBI_SS(sum_length_EBI_SS_poly_value)).^2)/
nu_EBI_SS).^.5;
```

```
syx_value_poly_EBI_SS_string=num2str(syx_value_poly_EBI_SS);
text(30, -11, ['S_Y_X_,_E_B_I_ _S_S = '
 syx_value_poly_EBI_SS_string , ' \circ C'])


sum_length_poly_value_EIB_AL=start_poly_EIB_AL:length(residual_value_poly_EIB_AL);
nu_EIB_AL=length(EIB_AL_Temp_smooth)-(m_fits+1);
sum_length_EIB_AL_poly_value=start_poly_EIB_AL:1:length(Temp_tau_poly_value_EIB_AL
syx_value_poly_EIB_AL=(sum((EIB_AL_Temp_smooth(sum_length_EIB_AL_poly_value)-
Temp_tau_poly_value_EIB_AL(sum_length_EIB_AL_poly_value)).^2)/
nu_EIB_AL).^.5;
syx_value_poly_EIB_AL_string=num2str(syx_value_poly_EIB_AL);
text(30, -13, ['S_Y_X_,_E_I_B_ _A_L = '
 syx_value_poly_EIB_AL_string , ' \circ C'])


sum_length_poly_value_EBI_AL=start_poly_EBI_AL:length(residual_value_poly_EBI_AL);
nu_EBI_AL=length(EBI_AL_Temp_smooth)-(m_fits+1);
sum_length_EBI_AL_poly_value=start_poly_EBI_AL:1:length(Temp_tau_poly_value_EBI_AL
syx_value_poly_EBI_AL=(sum((EBI_AL_Temp_smooth(sum_length_EBI_AL_poly_value)-
Temp_tau_poly_value_EBI_AL(sum_length_EBI_AL_poly_value)).^2)/
nu_EBI_AL).^.5;
syx_value_poly_EBI_AL_string=num2str(syx_value_poly_EBI_AL);
text(30, -15, ['S_Y_X_,_E_B_I_ _A_L = '
 syx_value_poly_EBI_AL_string , ' \circ C'])


title('Residuals of Maximum Slope Method found with Value of .632')
ylabel('Temperature (\circ C)')
xlabel('Time (sec)')
legend('BBI', 'BIB', 'EIB SS', 'EBI SS', 'EIB AL', 'EBI AL')
ylim([-20, 20])
grid minor
figure
```


Residuals of Standard Deviation Method found with Gamma
BBI
BIB
EIB SS
EBI SS
EIB AL
EBI AL
$S_{YX,BBI}$ = 19.4023 ° C
$S_{YX,BIB}$ = 15.2807 ° C
$S_{YX,EIB\ SS}$ = 7.7911 ° C
$S_{YX,EIB\ SS}$ = 7.5768 ° C
$S_{YX,EIB\ AL}$ = 9.5884 ° C
$S_{YX,EBI\ AL}$ = 7.6842 ° C
Temperature (° C)
Time (sec)

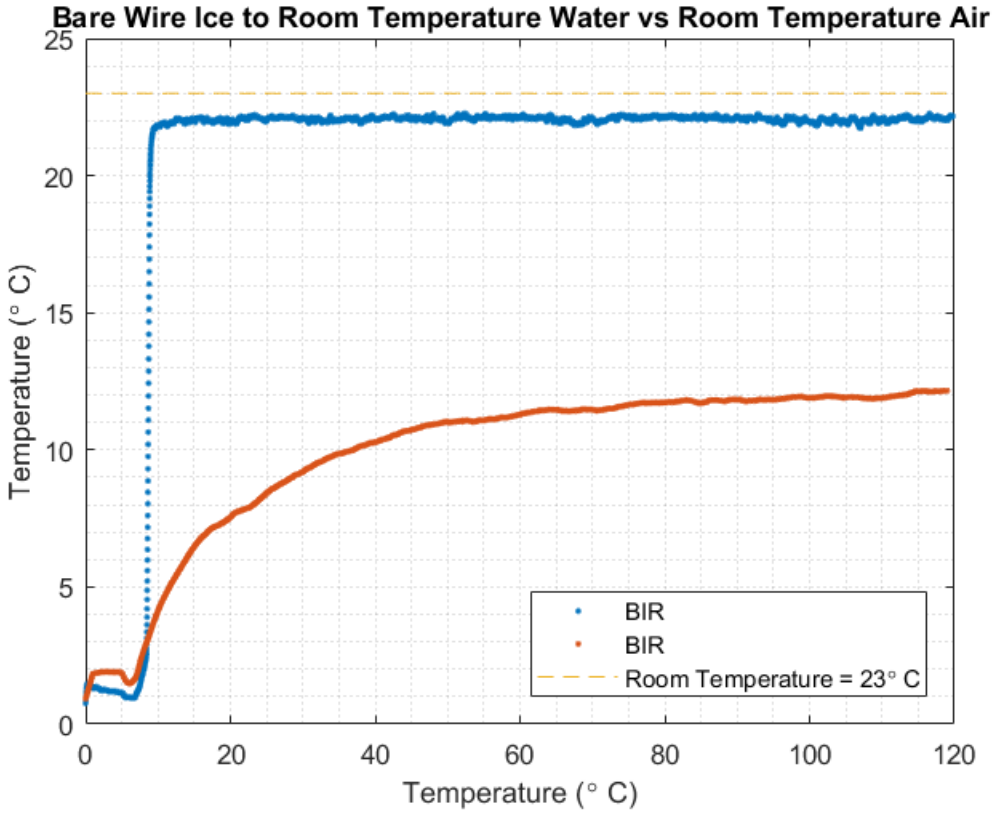

Residuals of Maximum Slope Method found with Gamma
BBI
BIB
EIB SS
EBI SS
EIB AL
EBI AL
$S_{YX,BBI}$ = 5.5754 ° C
$S_{YX,BIB}$ = 0.56462 ° C
$S_{YX,EIB\ SS}$ = 2.4645 ° C
$S_{YX,EIB\ SS}$ = 4.6161 ° C
$S_{YX,EIB\ AL}$ = 9.6297 ° C
$S_{YX,EBI\ AL}$ = 5.8523 ° C
Temperature (° C)
Time (sec)


Residuals of Standard Deviation Method found with Value of .632
Temperature (° C)
Time (sec)
BBI
BIB
EIB SS
EBI SS
EIB AL
EBI AL
S YX,BBI = 14.1008 ° C
S YX,BIB = 8.9083 ° C
S YX,EIB SS = 7.7964 ° C
S YX,EBI SS = 6.7197 ° C
S YX,EIB AL = 4.0434 ° C
S YX,EBI AL = 4.0508 ° C


Residuals of Maximum Slope Method found with Value of .632
Temperature (° C)
Time (sec)
BBI
BIB
EIB SS
EBI SS
EIB AL
EBI AL
S YX,BBI = 0.43267 ° C
S YX,BIB = 0.35289 ° C
S YX,EIB SS = 0.65962 ° C
S YX,EBI SS = 1.3137 ° C
S YX,EIB AL = 1.0994 ° C
S YX,EBI AL = 2.4298 ° C

Room Temp Water vs Room Temp Air

```
plot(BIR_time, BIR_Temp_smooth, '.')
hold on
for i=11900:length(BIR_time)
    if BIA_Temp_smooth(i)<12.1
        BIA_Temp_smooth(i)=NaN;
    end
end
plot(BIR_time, BIA_Temp_smooth, '.')
grid minor
room_temp=linspace((73.4-32)*(5/9), (73.4-32)*(5/9),
 length(BIR_time));%Room Temp in degrees Celsisus
plot(BIR_time, room_temp, '--')
legend('BIR','BIR', ['Room Temperature = '
 num2str((73.4-32)*(5/9)) '\circ C'], 'Location', 'southeast')
title('Bare Wire Ice to Room Temperature Water vs Room Temperature
 Air')
ylabel('Temperature (\circ C)')
xlabel('Temperature (\circ C)')
```



Make Arrays of Tau and Sxy to Put into Tables

```
table_std_gamma_Tau=[tau_std_BBI, tau_std_BIB, tau_std_EBI_SS,
 tau_std_EIB_SS, tau_std_EBI_AL, tau_std_EIB_AL];

table_poly_gamma_Tau=[tau_poly_BBI, tau_poly_BIB, tau_poly_EBI_SS,
 tau_poly_EIB_SS, tau_poly_EBI_AL, tau_poly_EIB_AL];

table_poly_value_Tau=[Tau_poly_value_BBI, Tau_poly_value_BIB,
 Tau_poly_value_EBI_SS, Tau_poly_value_EIB_SS, Tau_poly_value_EBI_AL,
 Tau_poly_value_EIB_AL];

table_std_value_Tau=[Tau_std_value_BBI, Tau_std_value_BIB,
 Tau_std_value_EBI_SS, Tau_std_value_EIB_SS, Tau_std_value_EBI_AL,
 Tau_std_value_EIB_AL];


%%%%%%%%%Syx

table_syx_std_gamma_syx=[syx_gamma_std_BBI, syx_gamma_std_BIB,
 syx_gamma_std_EBI_SS, syx_gamma_std_EIB_SS, syx_gamma_std_EBI_AL,
 syx_gamma_std_EIB_AL];
```

```
table_syx_poly_gamma=[syx_gamma_poly_BBI, syx_gamma_poly_BIB,
 syx_gamma_poly_EBI_SS, syx_gamma_poly_EIB_SS, syx_gamma_poly_EBI_AL,
 syx_gamma_poly_EIB_AL];

table_syx_std_value=[syx_value_std_BBI, syx_value_std_BIB,
 syx_value_std_EBI_SS, syx_value_std_EIB_SS, syx_value_std_EBI_AL,
 syx_value_std_EIB_AL];

table_syx_poly_value=[syx_value_poly_BBI, syx_value_poly_BIB,
 syx_value_poly_EBI_SS, syx_value_poly_EIB_SS, syx_value_poly_EBI_AL,
 syx_value_poly_EIB_AL];
```

Published with MATLAB® R2017a