Table of Contents


```
% John Shuler
% GEOS597: Term Project
% Due: 12/9/2016

close all;
clear all;
clc;
```

Load in Data

```
fileID = fopen('./CR6Series_SoilPit1_2016_07_09.dat','r');
July9 = textscan(fileID, '%s%s%d%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f
%f', 'HeaderLines', 4);
fclose(fileID);

fileID = fopen('./CR6Series_SoilPit1_2016_07_22.dat','r');
July22 = textscan(fileID, '%s%s%d%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f
%f', 'HeaderLines', 4,'TreatAsEmpty','NaN');
fclose(fileID);

fileID = fopen('./CR6Series_SoilPit1_2016_08_06.dat','r');
Aug6 = textscan(fileID, '%s%s%d%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f
%f', 'HeaderLines', 4,'TreatAsEmpty','NaN');
fclose(fileID);

fileID = fopen('./CR6Series_SoilPit1_2016_08_23.dat','r');
Aug23 = textscan(fileID, '%s%s%d%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f
%f', 'HeaderLines', 4,'TreatAsEmpty','NaN');
fclose(fileID);

fileID = fopen('./CR6Series_SoilPit1_2016_09_09.dat','r');
Sep9 = textscan(fileID, '%s%s%d%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f
%f', 'HeaderLines', 4,'TreatAsEmpty','NaN');
```

```
fclose(fileID);

fileID = fopen('./CR6Series_SoilPit1_2016_09_30.dat','r');
Sep30 = textscan(fileID, '%s%s%d%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f%f
%f', 'HeaderLines', 4,'TreatAsEmpty','NaN');
fclose(fileID);
```

Parse Moisture Data, Delete Superfluous

```
July9datetime (:,1) = July9{1}(5471:9718);
July9datetime (:,2) = July9{2}(5471:9718);

July9moisture = zeros(4248,5);
for i = 4:8;
    July9moisture (:,i-3) = July9{i}(5471:9718);
end

July22datetime (:,1) = July22{1}(22:end);
July22datetime (:,2) = July22{2}(22:end);

July22moisture = zeros(1244,5);
for i = 4:8;
    July22moisture (:,i-3) = July22{i}(22:end);
end

Aug6datetime (:,1) = Aug6{1}(9:end);
Aug6datetime (:,2) = Aug6{2}(9:end);

Aug6moisture = zeros(1440,5);
for i = 4:8;
    Aug6moisture (:,i-3) = Aug6{i}(9:end);
end

Aug23datetime (:,1) = Aug23{1}(5:end);
Aug23datetime (:,2) = Aug23{2}(5:end);

Aug23moisture = zeros(1614,5);
for i = 4:8;
    Aug23moisture (:,i-3) = Aug23{i}(5:end);
end

Sep9datetime (:,1) = Sep9{1}(10:end);
Sep9datetime (:,2) = Sep9{2}(10:end);

Sep9moisture = zeros(1653,5);
for i = 4:8;
    Sep9moisture (:,i-3) = Sep9{i}(10:end);
end

Sep30datetime (:,1) = Sep30{1}(62:end);
Sep30datetime (:,2) = Sep30{2}(62:end);

Sep30moisture = zeros(2009,5);
```

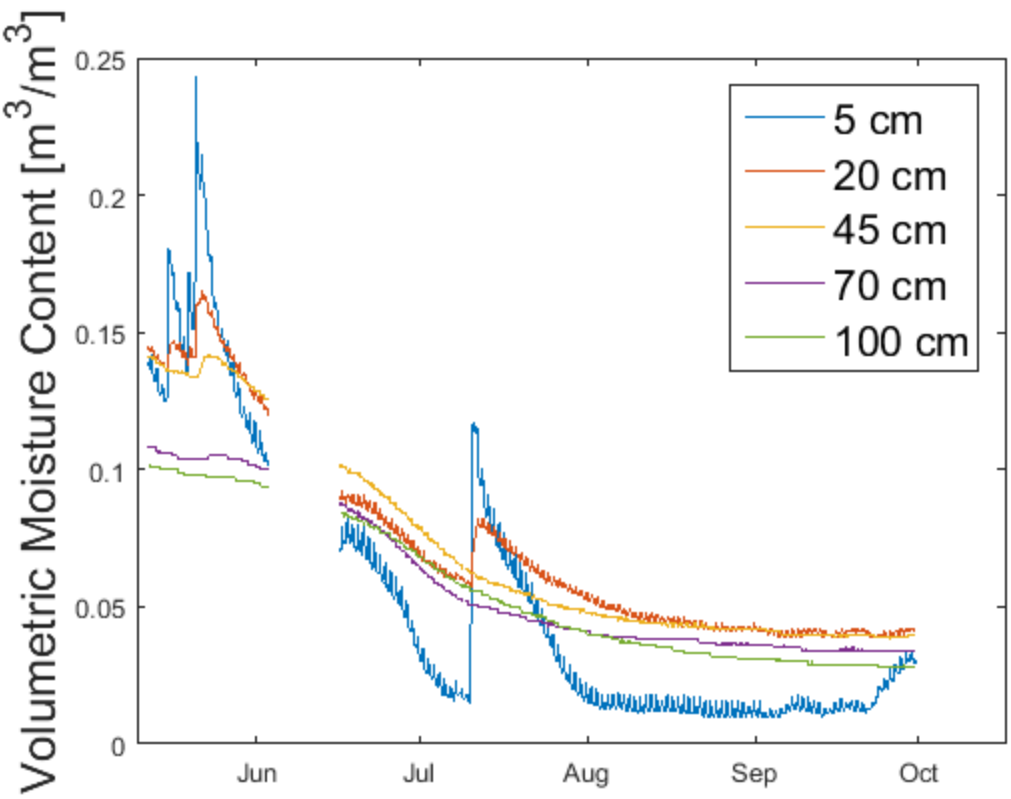
```
for i = 4:8;
    Sep30moisture (:,i-3) = Sep30{i}(62:end);
end

pit1Moisture = [July9moisture; July22moisture; Aug6moisture;
 Aug23moisture; Sep9moisture; Sep30moisture];

pit1Moisture(5492,:) = [];                  % find and delete
 duplicate moisture values
pit1Moisture(8546,:) = [];
pit1Moisture(10197,:) = [];
```

Create date vector, convert to string, delete duplicate dates

```
datetimestamp = [July9datetime; July22datetime; Aug6datetime;
 Aug23datetime; Sep9datetime; Sep30datetime];

A = char(datetimestamp(:,1));
B = char(datetimestamp(:,2));
dayTime = strcat(A, {' '}, B);
formatIn = 'mm/dd/yyyy HH:MM';
dateVec = datenum(dayTime,formatIn);

dateVec(5492) = [];                         % Remove duplicate dates
dateVec(8546) = [];
dateVec(10197) = [];
```

Plot soil moisture with new date vector with gaps

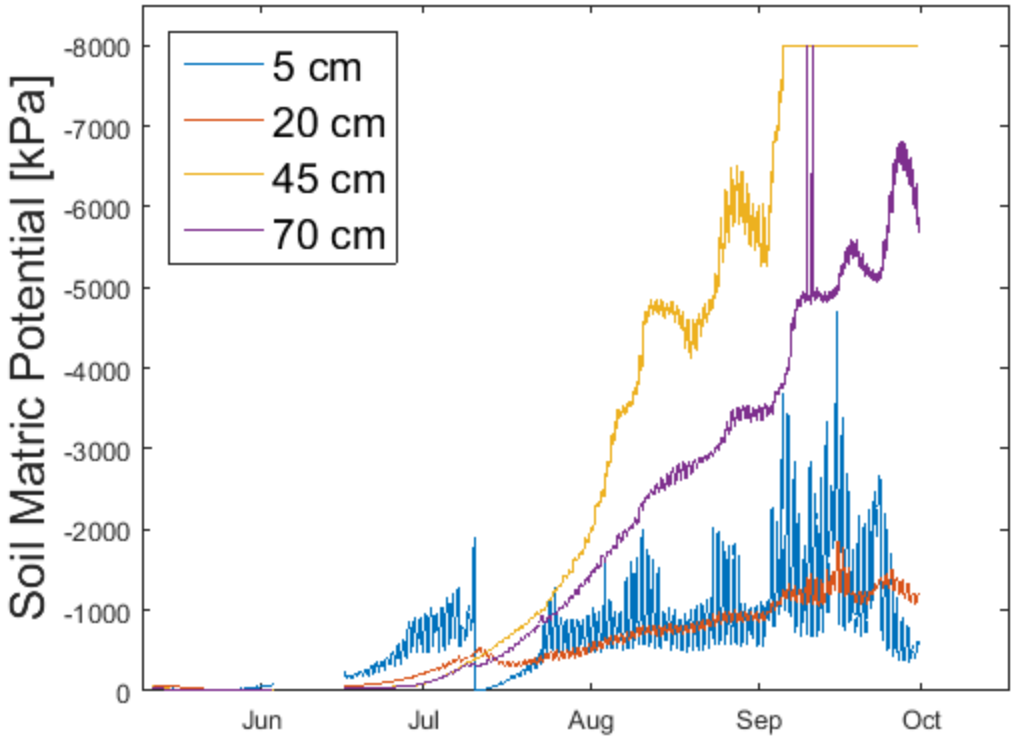
```
figure;
for i = 1:5;
    plot (dateVec, pit1Moisture(:,i));
    hold on;
end

legend({'5 cm','20 cm','45 cm','70 cm','100 cm'},'FontSize',16)
datetick('x','mmm','keeplimits')
%title('Raw Soil Moisture Data: May - September 2016')
ylabel('Volumetric Moisture Content [m^3/m^3]','FontSize',18)
print('RawSoilMoisture','-dpng')
```



Parse Tension Data, Delete Superfluous

```
July9tension (:,1) = July9{14}(5471:9718);
July9tension (:,2) = July9{16}(5471:9718);
July9tension (:,3) = July9{18}(5471:9718);
July9tension (:,4) = July9{20}(5471:9718);

July22tension (:,1) = July22{14}(22:end);
July22tension (:,2) = July22{16}(22:end);
July22tension (:,3) = July22{18}(22:end);
July22tension (:,4) = July22{20}(22:end);

Aug6tension (:,1) = Aug6{14}(9:end);
Aug6tension (:,2) = Aug6{16}(9:end);
Aug6tension (:,3) = Aug6{18}(9:end);
Aug6tension (:,4) = Aug6{20}(9:end);

Aug23tension (:,1) = Aug23{14}(5:end);
Aug23tension (:,2) = Aug23{16}(5:end);
Aug23tension (:,3) = Aug23{18}(5:end);
Aug23tension (:,4) = Aug23{20}(5:end);

Sep9tension (:,1) = Sep9{14}(10:end);
Sep9tension (:,2) = Sep9{16}(10:end);
Sep9tension (:,3) = Sep9{18}(10:end);
```

```
Sep9tension (:,4) = Sep9{20}(10:end);

Sep30tension (:,1) = Sep30{14}(62:end);
Sep30tension (:,2) = Sep30{16}(62:end);
Sep30tension (:,3) = Sep30{18}(62:end);
Sep30tension (:,4) = Sep30{20}(62:end);

pit1Tension = [July9tension; July22tension; Aug6tension; Aug23tension;
 Sep9tension; Sep30tension];

pit1Tension(5492,:) = [];               % Remove tension values
 associated with duplicate dates
pit1Tension(8546,:) = [];
pit1Tension(10197,:) = [];

figure;
for i = 1:4;
    plot (dateVec, pit1Tension(:,i));
    hold on;
end

legend({'5 cm','20 cm','45 cm','70
 cm'},'FontSize',16,'Location','Northwest')
datetick('x','mmm','keeplimits')
set(gca,'Ydir','reverse')
%title('Raw Soil Tension Data')
ylabel('Soil Matric Potential [kPa]','FontSize',18)
ylim([-8500 0]);
print('RawSoilTension','-dpng')
```



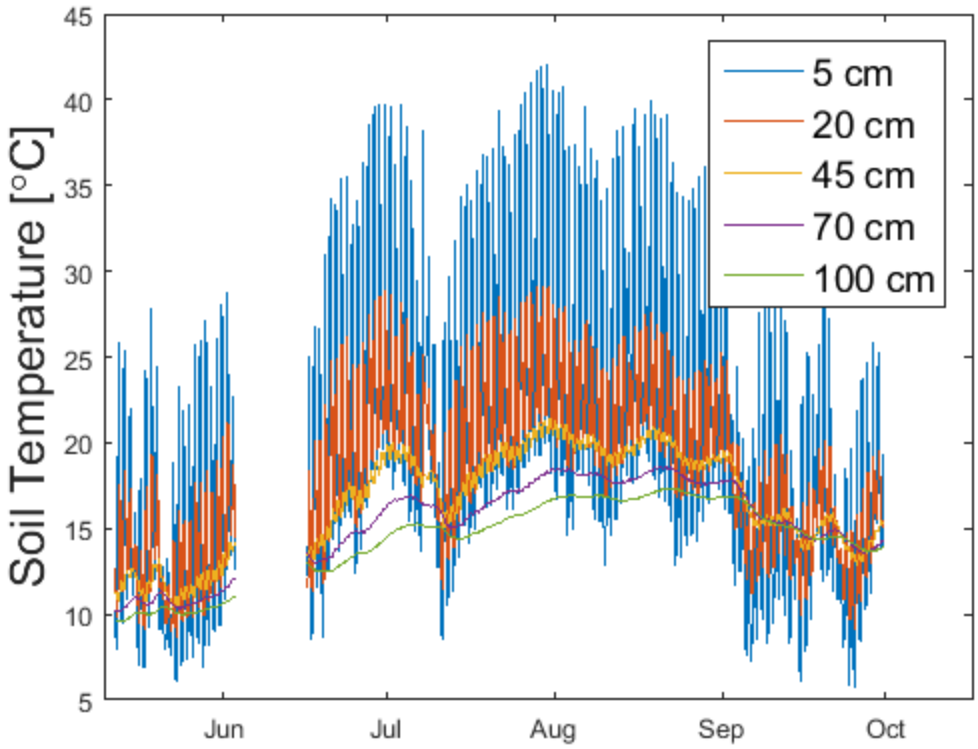
Replace dropouts/spikes with NaN

```
pit1Tension(pit1Tension<-7998) = NaN;    % Replace all dropout/
anomalous values with NaN
pit1Tension(10197:10222,4) = NaN;
pit1Tension(10293:10317,4) = NaN;
```

Parse temperature data; delete superfluous

```
July9temp (:,1) = July9{9}(5471:9718);
July9temp (:,2) = July9{10}(5471:9718);
July9temp (:,3) = July9{11}(5471:9718);
July9temp (:,4) = July9{12}(5471:9718);
July9temp (:,5) = July9{13}(5471:9718);

July22temp (:,1) = July22{9}(22:end);
July22temp (:,2) = July22{10}(22:end);
July22temp (:,3) = July22{11}(22:end);
July22temp (:,4) = July22{12}(22:end);
July22temp (:,5) = July22{13}(22:end);

Aug6temp (:,1) = Aug6{9}(9:end);
Aug6temp (:,2) = Aug6{10}(9:end);
Aug6temp (:,3) = Aug6{11}(9:end);
Aug6temp (:,4) = Aug6{12}(9:end);
```

```
Aug6temp (:,5) = Aug6{13}(9:end);

Aug23temp (:,1) = Aug23{9}(5:end);
Aug23temp (:,2) = Aug23{10}(5:end);
Aug23temp (:,3) = Aug23{11}(5:end);
Aug23temp (:,4) = Aug23{12}(5:end);
Aug23temp (:,5) = Aug23{13}(5:end);

Sep9temp (:,1) = Sep9{9}(10:end);
Sep9temp (:,2) = Sep9{10}(10:end);
Sep9temp (:,3) = Sep9{11}(10:end);
Sep9temp (:,4) = Sep9{12}(10:end);
Sep9temp (:,5) = Sep9{13}(10:end);

Sep30temp (:,1) = Sep30{9}(62:end);
Sep30temp (:,2) = Sep30{10}(62:end);
Sep30temp (:,3) = Sep30{11}(62:end);
Sep30temp (:,4) = Sep30{12}(62:end);
Sep30temp (:,5) = Sep30{13}(62:end);

pit1Temp = [July9temp; July22temp; Aug6temp; Aug23temp; Sep9temp;
 Sep30temp];

pit1Temp(5492,:) = [];          % Remove temp values associated with
 duplicate dates
pit1Temp(8546,:) = [];
pit1Temp(10197,:) = [];

pit1Temp(2134,:) = NaN;       % Replace zero values in this row with
 NaN

figure;
for i = 1:5;
    plot (dateVec, pit1Temp(:,i));
    hold on;
end

legend({'5 cm','20 cm','45 cm','70 cm','100 cm'},'FontSize',14)
datetick('x','mmm','keeplimits')
%title('Raw Temperature Data')
ylabel('Soil Temperature [\circC]','FontSize',18)
print('RawSoilTemp','-dpng')
```



Create new date vector

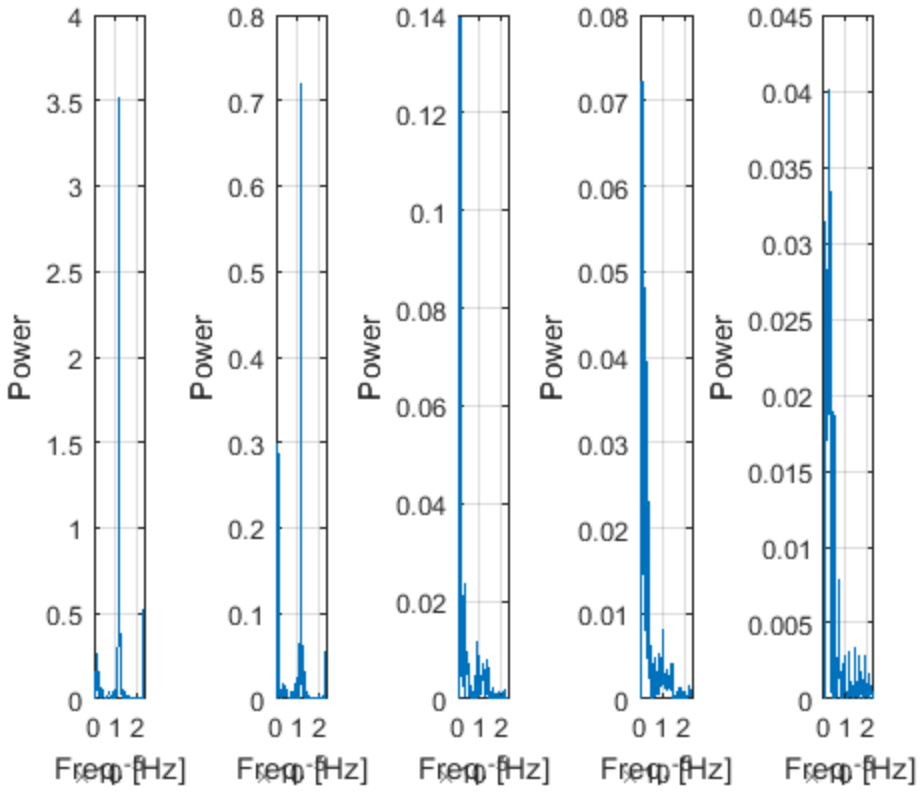
```
t0 = dateVec(1);
dt = dateVec(2)-dateVec(1);
dateVecNew = (t0:dt:dateVec(end))'; % new date vector with no gaps
```

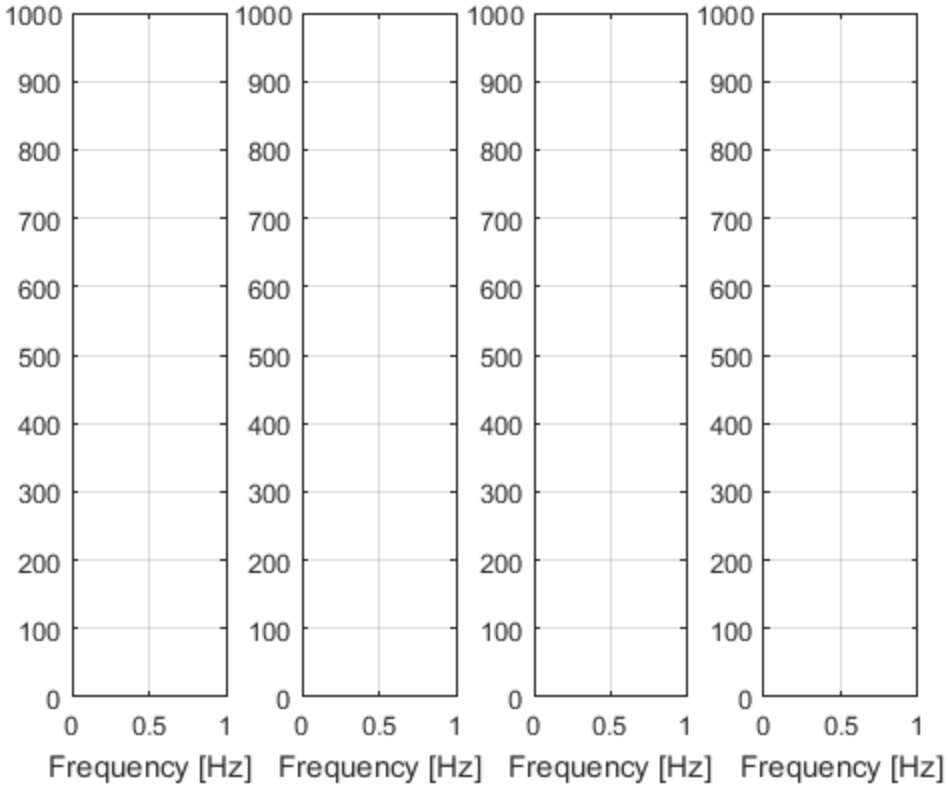
Periodogram of moisture, tension, temperature raw data

```
fs = 1/(15*60);               % sampling frequency in Hz
N = length(pit1Moisture)-1;
nfft = 2^nextpow2(N);
subset = pit1Moisture(8000:10000,1:5);

figure;
for i = 1:5;
    A = detrend(subset(:,i),'constant');
    B = detrend((A),'linear');
    [Pxx,f] = periodogram(B,[],nfft,fs);
    subplot(1,5,i)
    plot(f,Pxx);
    grid on;
    xlabel('Freq. [Hz]');
    xlim([0 1/(12*3600)])
```

```
    ylabel('Power')
end

figure;
for i = 1:4;
    [Pxx,f] = periodogram(pit1Tension(:,i),[],nfft,fs);
    subplot(1,4,i)
    title('Periodogram: Tension')
    plot(f,Pxx);
    grid on;
    xlabel('Frequency [Hz]');
    ylim([0 1000]);
end
```

```
Warning: Imaginary parts of complex X and/or Y arguments ignored
Warning: Imaginary parts of complex X and/or Y arguments ignored
Warning: Imaginary parts of complex X and/or Y arguments ignored
Warning: Imaginary parts of complex X and/or Y arguments ignored
```





Temperature Correction: Decagon Method

Choose 3 rain-free periods with similar start/end temps

```
A1 = pit1Moisture(1842:1937,1);
A2 = pit1Temp (1842:1937,1);

B1 = pit1Moisture(5594:5689,1);
B2 = pit1Temp (5594:5689,1);

C1 = pit1Moisture(8341:8436,1);
C2 = pit1Temp (8341:8436,1);

X1 = [dateVec(1842) dateVec(1937)];
X2 = [dateVec(5594) dateVec(5689)];
X3 = [dateVec(8341) dateVec(8436)];
Y1 = [A1(1) A1(end)];
Y2 = [B1(1) B1(end)];
Y3 = [C1(1) C1(end)];

p1 = polyfit(X1, Y1, 1);
p2 = polyfit(X2, Y2, 1);
p3 = polyfit(X3, Y3, 1);

yInt1 = polyval(p1, dateVec(1842:1937));
yInt2 = polyval(p2, dateVec(5594:5689));
```

```
yInt3 = polyval(p3, dateVec(8341:8436));

VMCmeasured = [A1; B1; C1];
temp        = [A2; B2; C2];
VMCint      = [yInt1; yInt2; yInt3];

X = [ones(size(VMCmeasured)) VMCmeasured temp VMCmeasured.*temp];
b = regress(VMCint,X);

C3 = b(1);
C1 = b(2);
C2 = b(3);

moistureCorrected = zeros(length(pit1Temp),5);
moistureCorrected(:,1) = C1.*(pit1Moisture(:,1)) + C2*(pit1Temp(:,1))
 + C3;

% 20 cm Sensor

A1 = pit1Moisture(1842:1937,2);
A2 = pit1Temp (1842:1937,2);

B1 = pit1Moisture(5594:5689,2);
B2 = pit1Temp (5594:5689,2);

C1 = pit1Moisture(8341:8436,2);
C2 = pit1Temp (8341:8436,2);

X1 = [dateVec(1842) dateVec(1937)];
X2 = [dateVec(5594) dateVec(5689)];
X3 = [dateVec(8341) dateVec(8436)];
Y1 = [A1(1) A1(end)];
Y2 = [B1(1) B1(end)];
Y3 = [C1(1) C1(end)];

p1 = polyfit(X1, Y1, 1);
p2 = polyfit(X2, Y2, 1);
p3 = polyfit(X3, Y3, 1);

yInt1 = polyval(p1, dateVec(1842:1937));
yInt2 = polyval(p2, dateVec(5594:5689));
yInt3 = polyval(p3, dateVec(8341:8436));

VMCmeasured = [A1; B1; C1];
temp        = [A2; B2; C2];
VMCint      = [yInt1; yInt2; yInt3];

X = [ones(size(VMCmeasured)) VMCmeasured temp VMCmeasured.*temp];
b = regress(VMCint,X);

C3 = b(1);
C1 = b(2);
C2 = b(3);
```

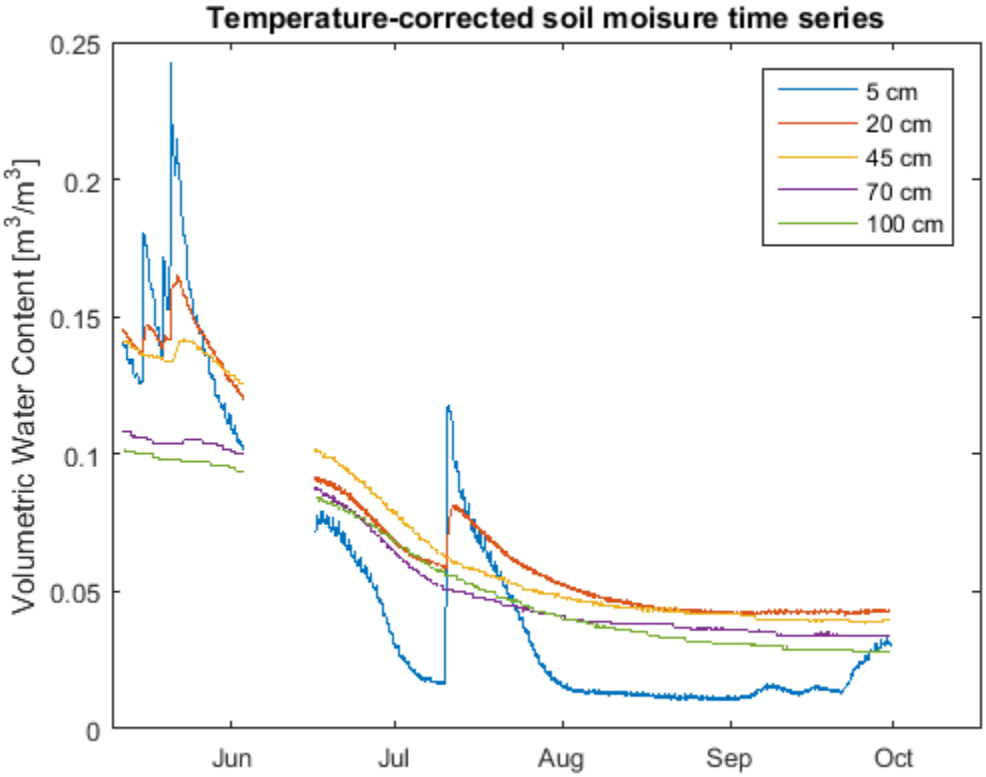
```
moistureCorrected(:,2) = C1.*(pit1Moisture(:,2)) + C2*(pit1Temp(:,2))
 + C3;

% 45 cm Sensor

% A1 = pit1Moisture(1842:1937,3);
% A2 = pit1Temp (1842:1937,3);
%
% B1 = pit1Moisture(5594:5689,3);
% B2 = pit1Temp (5594:5689,3);
%
% C1 = pit1Moisture(8341:8436,3);
% C2 = pit1Temp (8341:8436,3);
%
% X1 = [dateVec(1842) dateVec(1937)];
% X2 = [dateVec(5594) dateVec(5689)];
% X3 = [dateVec(8341) dateVec(8436)];
% Y1 = [A1(1) A1(end)];
% Y2 = [B1(1) B1(end)];
% Y3 = [C1(1) C1(end)];
%
% p1 = polyfit(X1, Y1, 1);
% p2 = polyfit(X2, Y2, 1);
% p3 = polyfit(X3, Y3, 1);
%
% yInt1 = polyval(p1, dateVec(1842:1937));
% yInt2 = polyval(p2, dateVec(5594:5689));
% yInt3 = polyval(p3, dateVec(8341:8436));
%
% VMCmeasured = [A1; B1; C1];
% temp        = [A2; B2; C2];
% VMCint      = [yInt1; yInt2; yInt3];
%
% X = [ones(size(VMCmeasured)) VMCmeasured temp VMCmeasured.*temp];
% b = regress(VMCint,X);
%
% C3 = b(1);
% C1 = b(2);
% C2 = b(3);
%
% moistureCorrected(:,3) = C1.*(pit1Moisture(:,3)) +
 C2*(pit1Temp(:,3)) + C3;
%
% % 70 cm Sensor
%
% A1 = pit1Moisture(1842:1937,4);
% A2 = pit1Temp (1842:1937,4);
%
% B1 = pit1Moisture(5594:5689,4);
% B2 = pit1Temp (5594:5689,4);
%
% C1 = pit1Moisture(8341:8436,4);
% C2 = pit1Temp (8341:8436,4);
%
```

```
% X1 = [dateVec(1842) dateVec(1937)];
% X2 = [dateVec(5594) dateVec(5689)];
% X3 = [dateVec(8341) dateVec(8436)];
% Y1 = [A1(1) A1(end)];
% Y2 = [B1(1) B1(end)];
% Y3 = [C1(1) C1(end)];
%
% p1 = polyfit(X1, Y1, 1);
% p2 = polyfit(X2, Y2, 1);
% p3 = polyfit(X3, Y3, 1);
%
% yInt1 = polyval(p1, dateVec(1842:1937));
% yInt2 = polyval(p2, dateVec(5594:5689));
% yInt3 = polyval(p3, dateVec(8341:8436));
%
% VMCmeasured = [A1; B1; C1];
% temp        = [A2; B2; C2];
% VMCint      = [yInt1; yInt2; yInt3];
%
% X = [ones(size(VMCmeasured)) VMCmeasured temp VMCmeasured.*temp];
% b = regress(VMCint,X);
%
% C3 = b(1);
% C1 = b(2);
% C2 = b(3);
%
% moistureCorrected(:,4) = C1.*(pit1Moisture(:,4)) +
 C2*(pit1Temp(:,4)) + C3;
%
% % 100 cm Sensor
%
% A1 = pit1Moisture(1842:1937,5);
% A2 = pit1Temp (1842:1937,5);
%
% B1 = pit1Moisture(5594:5689,5);
% B2 = pit1Temp (5594:5689,5);
%
% C1 = pit1Moisture(8341:8436,5);
% C2 = pit1Temp (8341:8436,5);
%
% X1 = [dateVec(1842) dateVec(1937)];
% X2 = [dateVec(5594) dateVec(5689)];
% X3 = [dateVec(8341) dateVec(8436)];
% Y1 = [A1(1) A1(end)];
% Y2 = [B1(1) B1(end)];
% Y3 = [C1(1) C1(end)];
%
% p1 = polyfit(X1, Y1, 1);
% p2 = polyfit(X2, Y2, 1);
% p3 = polyfit(X3, Y3, 1);
%
% yInt1 = polyval(p1, dateVec(1842:1937));
% yInt2 = polyval(p2, dateVec(5594:5689));
% yInt3 = polyval(p3, dateVec(8341:8436));
```

```
%
% VMCmeasured = [A1; B1; C1];
% temp        = [A2; B2; C2];
% VMCint      = [yInt1; yInt2; yInt3];
%
% X = [ones(size(VMCmeasured)) VMCmeasured temp VMCmeasured.*temp];
% b = regress(VMCint,X);
%
% C3 = b(1);
% C1 = b(2);
% C2 = b(3);
%
% moistureCorrected(:,5) = C1.*(pit1Moisture(:,5)) +
 C2*(pit1Temp(:,5)) + C3;

moistureCorrected(:,3:5) = pit1Moisture(:,3:5);

figure;
for i = 1:5;
    plot(dateVec, moistureCorrected(:,i));
    hold on;
end

legend('5 cm','20 cm','45 cm','70 cm','100 cm')
datetick('x','mmm','keeplimits')
title('Temperature-corrected soil moisure time series')
ylabel('Volumetric Water Content [m^3/m^3]')
```

Warning: Polynomial is badly conditioned. Add points with distinct X values,
reduce the degree of the polynomial, or try centering and scaling as described
in HELP POLYFIT.
Warning: Polynomial is badly conditioned. Add points with distinct X values,
reduce the degree of the polynomial, or try centering and scaling as described
in HELP POLYFIT.
Warning: Polynomial is badly conditioned. Add points with distinct X values,
reduce the degree of the polynomial, or try centering and scaling as described
in HELP POLYFIT.
Warning: Polynomial is badly conditioned. Add points with distinct X values,
reduce the degree of the polynomial, or try centering and scaling as described
in HELP POLYFIT.
Warning: Polynomial is badly conditioned. Add points with distinct X values,
reduce the degree of the polynomial, or try centering and scaling as described
in HELP POLYFIT.

```
Warning: Polynomial is badly conditioned. Add points with distinct X
 values,
reduce the degree of the polynomial, or try centering and scaling as
 described
in HELP POLYFIT.
```

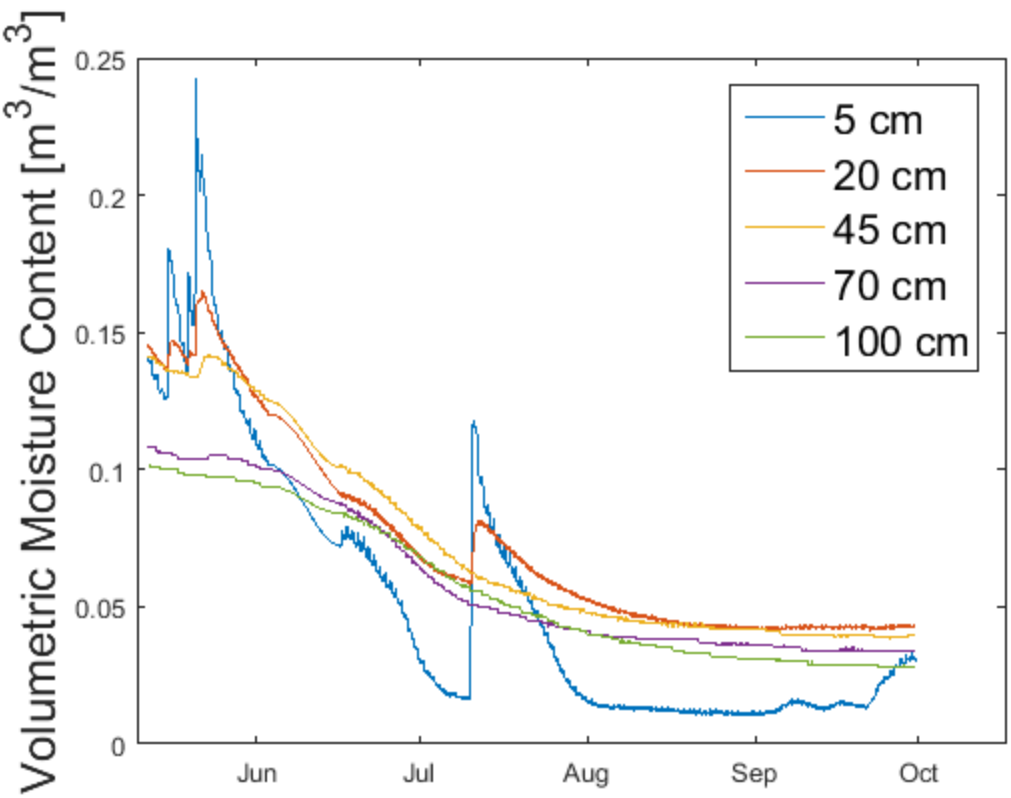


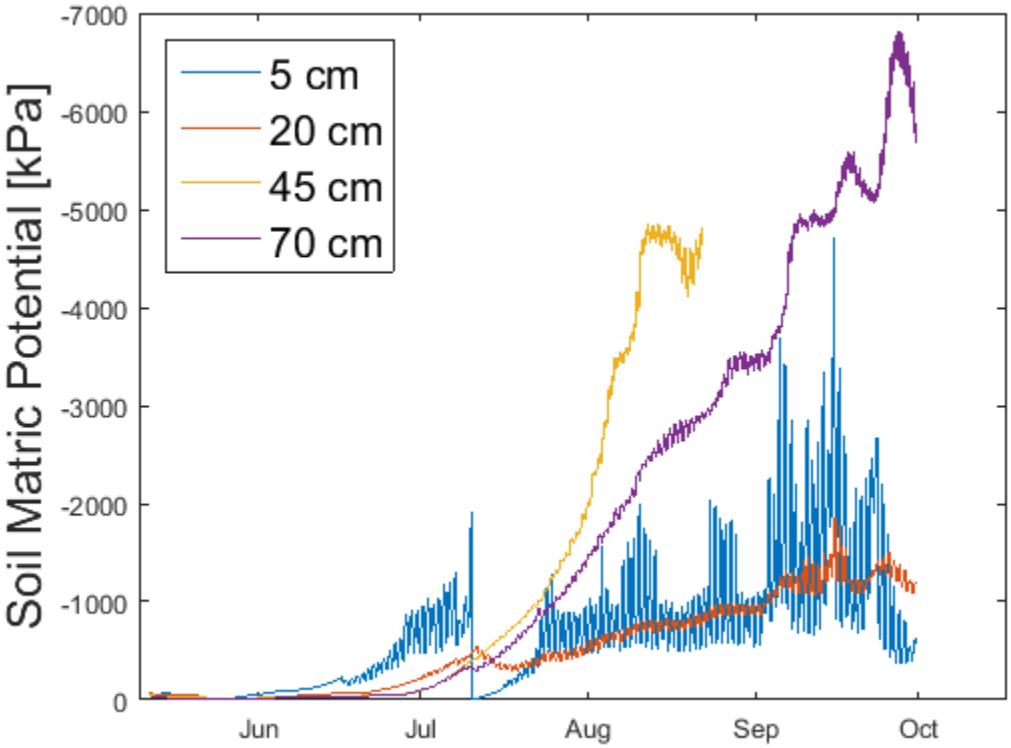
Interpolate missing values/data gaps

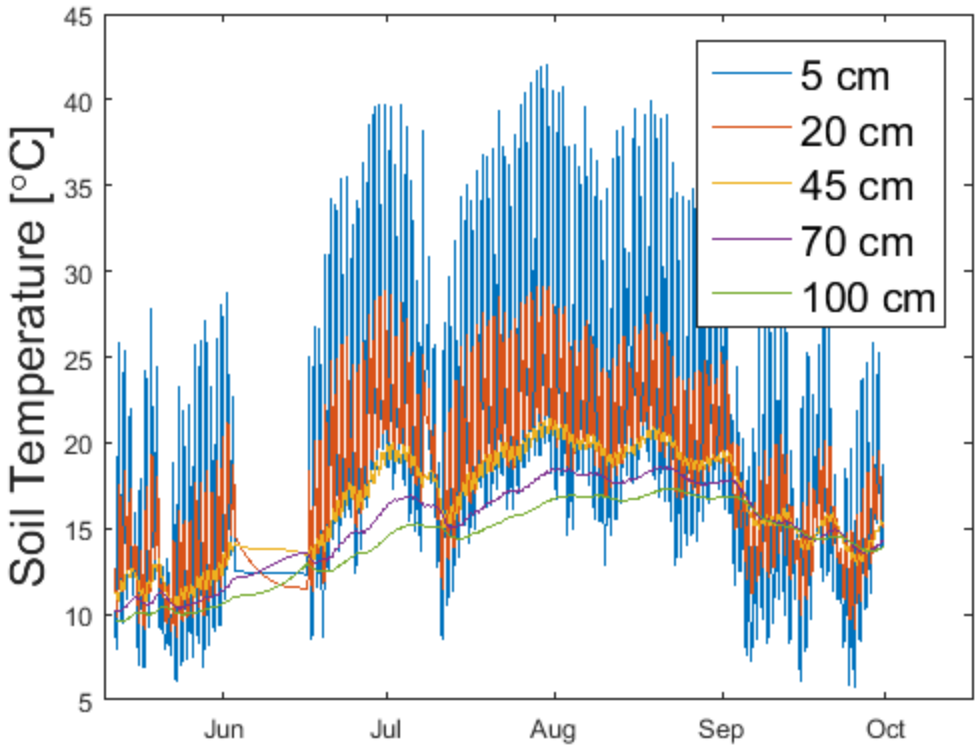
```
moistureInt = zeros(length(dateVecNew),5);
for i = 1:5;
    moistureInt(:,i) = interp1(dateVec, moistureCorrected(:,i),
 dateVecNew, 'pchip');
end

figure;
for i = 1:5;
    plot (dateVecNew, moistureInt(:,i));
    hold on;
end

legend({'5 cm','20 cm','45 cm','70 cm','100 cm'},'FontSize',16)
datetick('x','mmm','keeplimits')
ylabel('Volumetric Moisture Content [m^3/m^3]','FontSize',18)
print('ProcessedMoisture','-dpng')
%title('Soil moisture time series with interpolated values')
```

```
tensionInt = zeros(length(dateVecNew),4);
for i = 1:2;
    tensionInt(:,i) = interp1(dateVec, pit1Tension(:,i),
 dateVecNew, 'pchip');
end

A = zeros(13585,1);
A(9802:13585,1) = NaN;
B = interp1(dateVec(1:9801,1),pit1Tension(1:9801,3),
 dateVecNew(1:9801,1), 'pchip');
A(1:9801) = B;
tensionInt(:,3) = A;

tensionInt(:,4) = interp1(dateVec, pit1Tension(:,4),
 dateVecNew, 'pchip');

figure;
for i = 1:4;
    plot (dateVecNew, tensionInt(:,i));
    hold on;
end

legend({'5 cm','20 cm','45 cm','70
 cm'},'FontSize',16,'Location','Northwest')
datetick('x','mmm','keeplimits')
%title('Soil tension time series with interpolated values')
set(gca,'Ydir','reverse')
ylabel('Soil Matric Potential [kPa]','FontSize',18)
print('ProcessedTension','-dpng')


tempInt = zeros(length(dateVecNew),5);
for i = 1:5;
    tempInt(:,i) = interp1(dateVec, pit1Temp(:,i),
 dateVecNew, 'pchip');
end

figure;
for i = 1:5;
    plot (dateVecNew, tempInt(:,i));
    hold on;
end

legend({'5 cm','20 cm','45 cm','70 cm','100 cm'},'FontSize',16)
datetick('x','mmm','keeplimits')
%title('Soil temperature with interpolated values')
ylabel('Soil Temperature [\circC]','FontSize',18)
print('SoilTempFinal','-dpng')

Warning: Columns of data containing NaN values have been ignored
 during
interpolation.
Warning: Columns of data containing NaN values have been ignored
 during
```

```
interpolation.
Warning: Columns of data containing NaN values have been ignored
 during
interpolation.
Warning: Columns of data containing NaN values have been ignored
 during
interpolation.
Warning: Columns of data containing NaN values have been ignored
 during
interpolation.
Warning: Columns of data containing NaN values have been ignored
 during
interpolation.
Warning: Columns of data containing NaN values have been ignored
 during
interpolation.
Warning: Columns of data containing NaN values have been ignored
 during
interpolation.
Warning: Columns of data containing NaN values have been ignored
 during
interpolation.
Warning: Columns of data containing NaN values have been ignored
 during
interpolation.
Warning: Columns of data containing NaN values have been ignored
 during
interpolation.
Warning: Columns of data containing NaN values have been ignored
 during
interpolation.
Warning: Columns of data containing NaN values have been ignored
 during
interpolation.
Warning: Columns of data containing NaN values have been ignored
 during
interpolation.
```


Volumetric Moisture Content [m3/m3]
0.25
0.2
0.15
0.1
0.05
0
Jun
Jul
Aug
Sep
Oct
5 cm
20 cm
45 cm
70 cm
100 cm

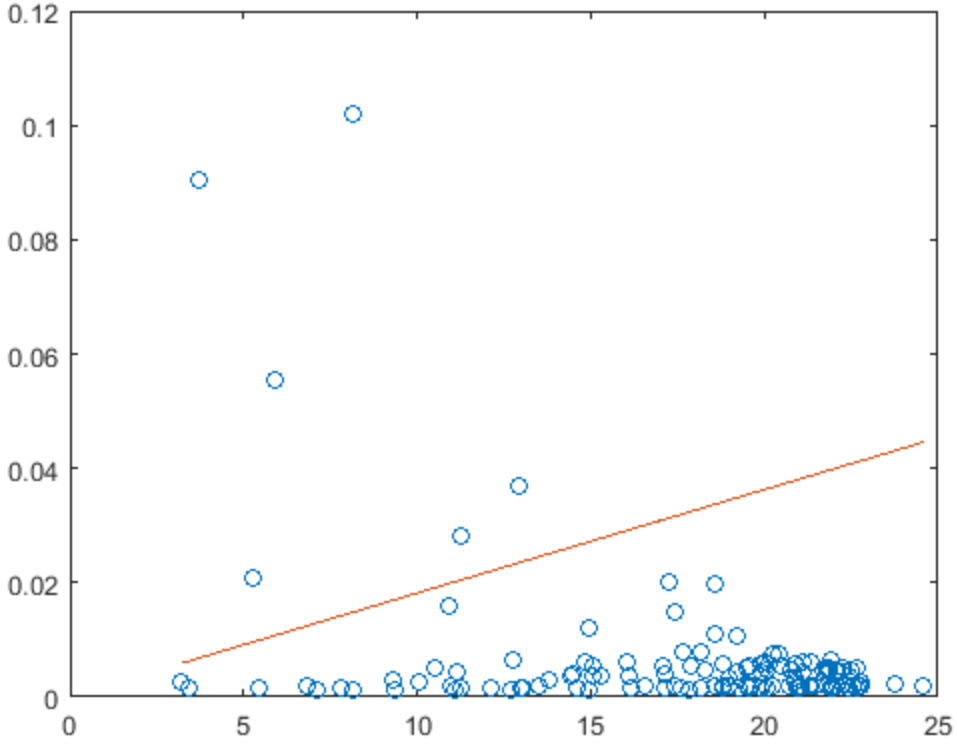

Soil Matric Potential [kPa]
-7000
-6000
-5000
-4000
-3000
-2000
-1000
0
Jun
Jul
Aug
Sep
Oct
5 cm
20 cm
45 cm
70 cm

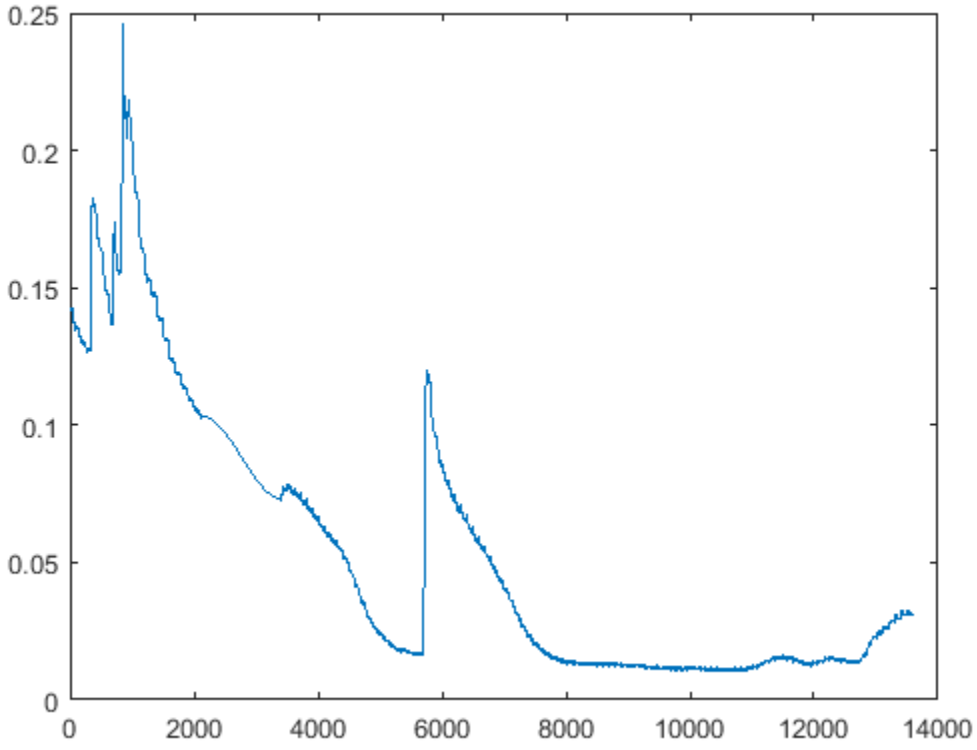


Temperature corrected moisture data: Lu Method

```
maxOmega = zeros(1,140);
for i = 2:140;
     maxOmega(1) = max(moistureInt(93:188));
     A = 93+(96*(i-1));
     B = (188+96*(i-1));
     maxOmega(i) = max(moistureInt (A:B));
end

minOmega = zeros(1,140);
for i = 2:140;
     minOmega(1) = min(moistureInt(93:188));
     A = 93+(96*(i-1));
     B = (188+96*(i-1));
     minOmega(i) = min(moistureInt (A:B));
end

Aomega = maxOmega - minOmega;
Aomega(23:34)= [];

maxTemp = zeros(1,140);
for i = 2:140;
```

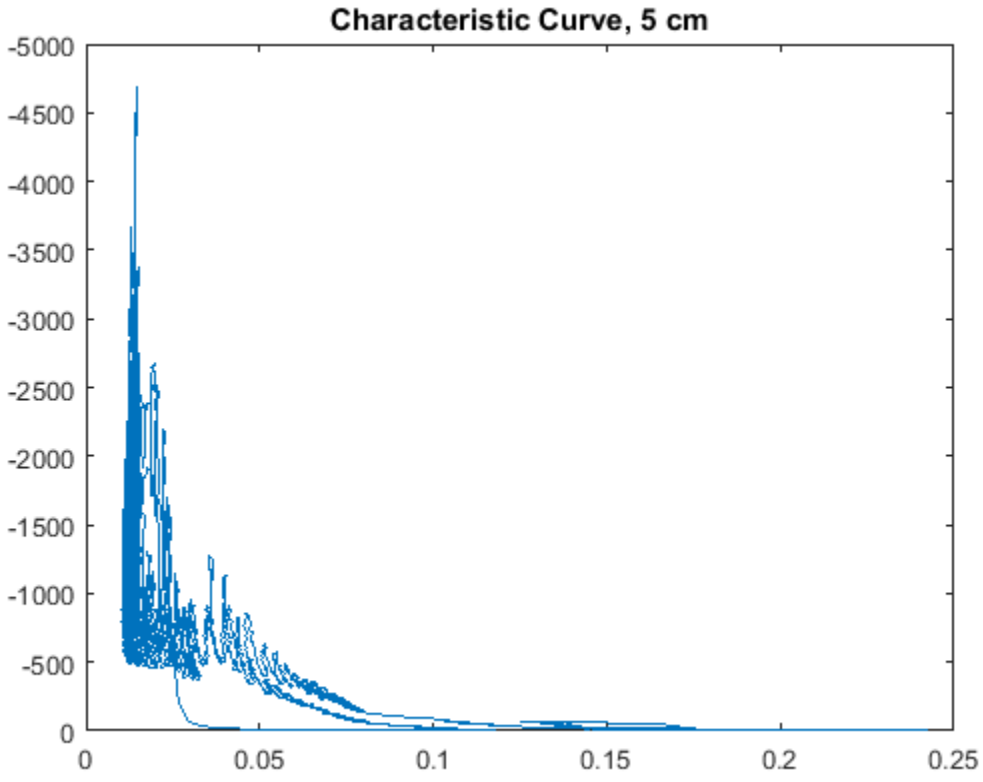
```
     maxTemp(1) = max(tempInt(93:188));
     A = 93+(96*(i-1));
     B = (188+96*(i-1));
     maxTemp(i) = max(tempInt (A:B));
end

minTemp = zeros(1,140);
for i = 2:140;
     minTemp(1) = min(tempInt(93:188));
     A = 93+(96*(i-1));
     B = (188+96*(i-1));
     minTemp(i) = min(tempInt (A:B));
end

Atemp = maxTemp - minTemp;
Atemp(23:34) = [];

brob = robustfit(Atemp, Aomega);
brobtrend = polyval(brob, Atemp);
figure;
plot(Atemp, Aomega,'o')
hold on;
plot(Atemp, brobtrend);

alpha = brob(1);
correctedOmega = moistureInt(:,1).*(1-(alpha*(tempInt(:,1)-20)));

figure;
plot(correctedOmega);
```

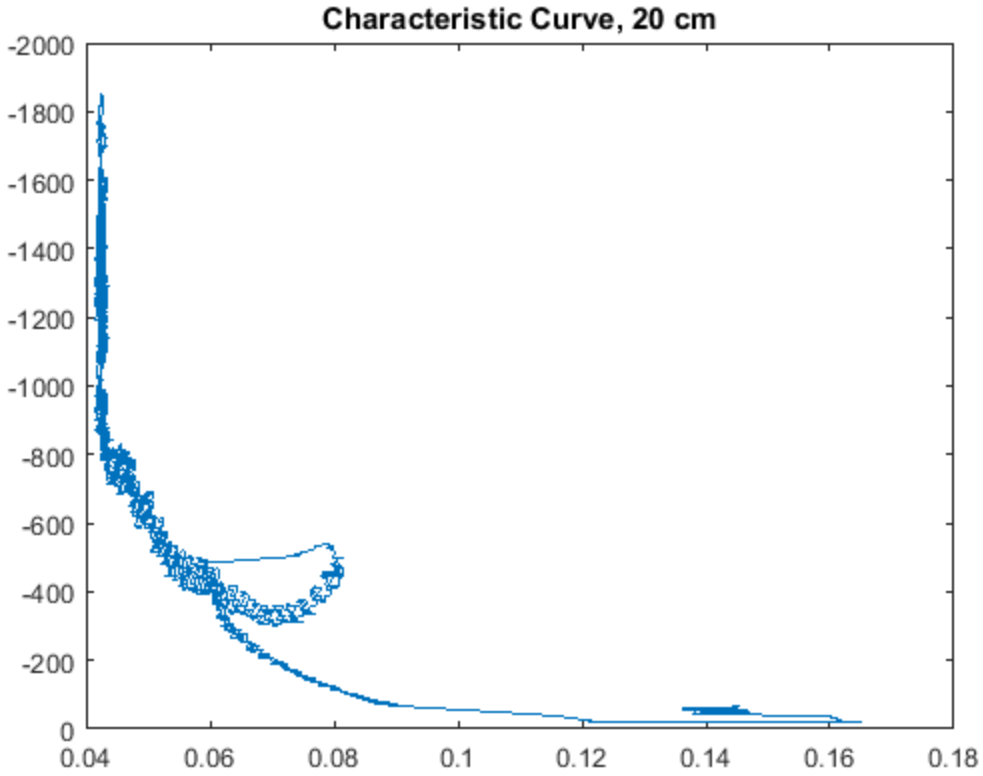

0.12
0.1
0.08
0.06
0.04
0.02
0
0
5
10
15
20
25

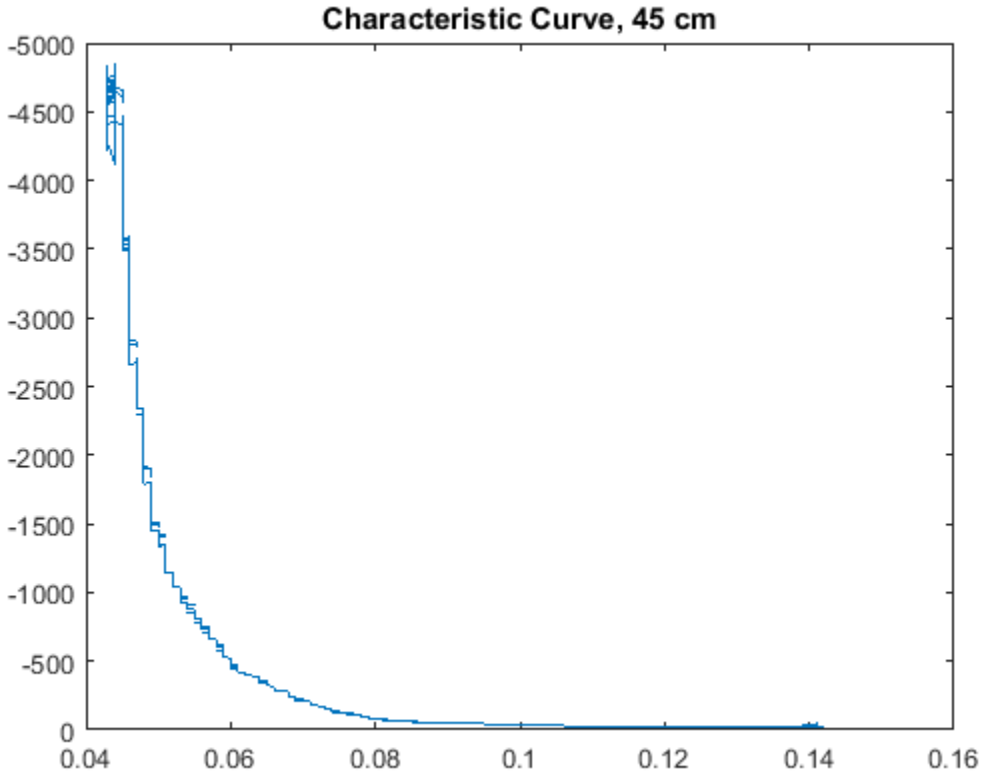

0.25
0.2
0.15
0.1
0.05
0
0
2000
4000
6000
8000
10000
12000
14000

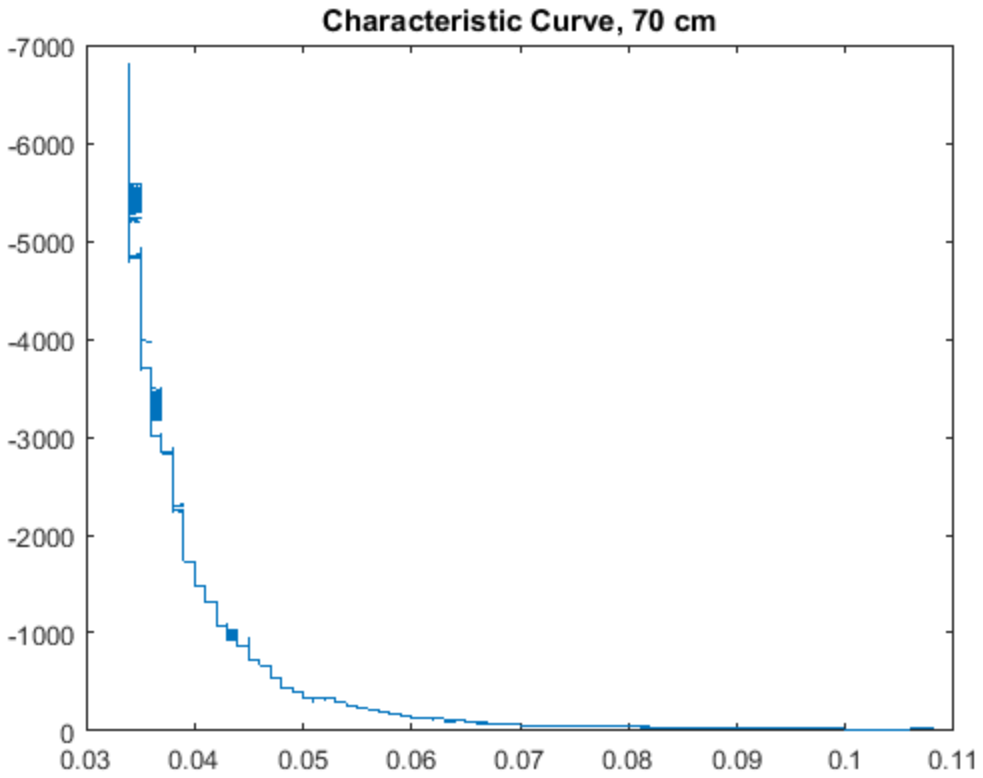
Create characteristic curves

```
figure;
plot(moistureInt(:,1),tensionInt(:,1));
set(gca,'Ydir','reverse')
title('Characteristic Curve, 5 cm')

figure;
plot(moistureInt(:,2),tensionInt(:,2));
set(gca,'Ydir','reverse')
title('Characteristic Curve, 20 cm')

figure;
plot(moistureInt(:,3),tensionInt(:,3));
set(gca,'Ydir','reverse')
title('Characteristic Curve, 45 cm')

figure;
plot(moistureInt(:,4),tensionInt(:,4));
set(gca,'Ydir','reverse')
title('Characteristic Curve, 70 cm')
```



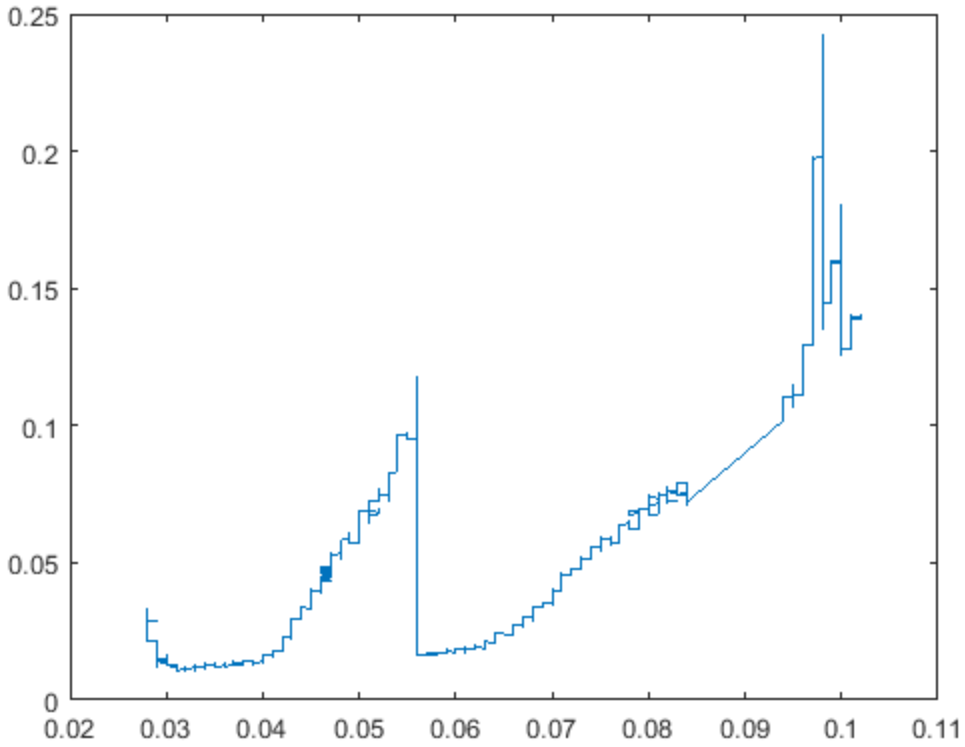

Characteristic Curve, 20 cm
-2000
-1800
-1600
-1400
-1200
-1000
-800
-600
-400
-200
0
0.04
0.06
0.08
0.1
0.12
0.14
0.16
0.18


Characteristic Curve, 45 cm
-5000
-4500
-4000
-3500
-3000
-2500
-2000
-1500
-1000
-500
0
0.04
0.06
0.08
0.1
0.12
0.14
0.16



Plot top and bottom moisture sensor against each other

```
figure;
plot(moistureInt(:,5),moistureInt(:,1));
```



Published with MATLAB® R2015a