Problem 1

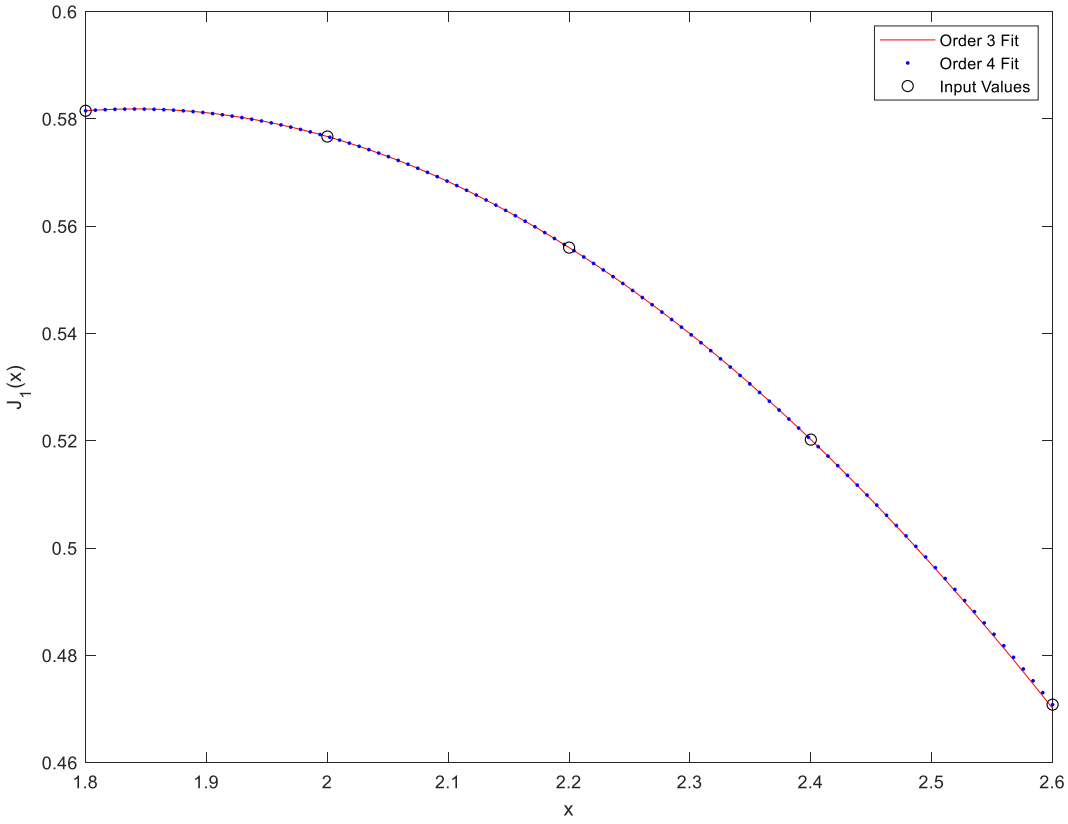
```
% PSET3_1
% Coded by Nigel F. Reuel on 9.22.2016
% Solves problme 17.13 in the text
x = [1.8 2 2.2 2.4 2.6];
J1 = [0.5815 0.5767 0.5560 0.5202 0.4708];
Tval = besselj(1,2.1); %<-- True value at 2.1
xx = linspace(1.8,2.6); % Vector to construct the fit curve
%
% For third order polynomial use four values closest to 2.1
(i.e. the first
% four)
clc
p1 = polyfit(x(1:4),J1(1:4),3);
yy1 = polyval(p1,xx);
f1 = polyval(p1,2.1);
E1 = ((f1-Tval)/Tval)*100;
fprintf('Percent relative error is %d percent for third
order interpolating polynomial\n',E1);
% For fourth order polynomial use all five values
p2 = polyfit(x,J1,4);
yy2 = polyval(p2,xx);
f2 = polyval(p2,2.1);
E2 = ((f2-Tval)/Tval)*100;
fprintf('Percent relative error is %d percent for fourth
order interpolating polynomial\n',E2);

plot(xx,yy1,':',xx,yy2,':',x,J1,'o')
legend('Order 3 Fit','Order 4 Fit','Input Values')
ylabel('J_1(x)')
xlabel('x')
```

Output

Percent relative error is -8.157349e-04 percent for third order interpolating polynomial

Percent relative error is 2.071205e-03 percent for fourth order interpolating polynomial


Order 3 Fit
Order 4 Fit
Input Values
0.6
0.58
0.56
0.54
0.52
0.5
0.48
0.46
$J_1(x)$
1.8
1.9
2
2.1
2.2
2.3
2.4
2.5
2.6
x

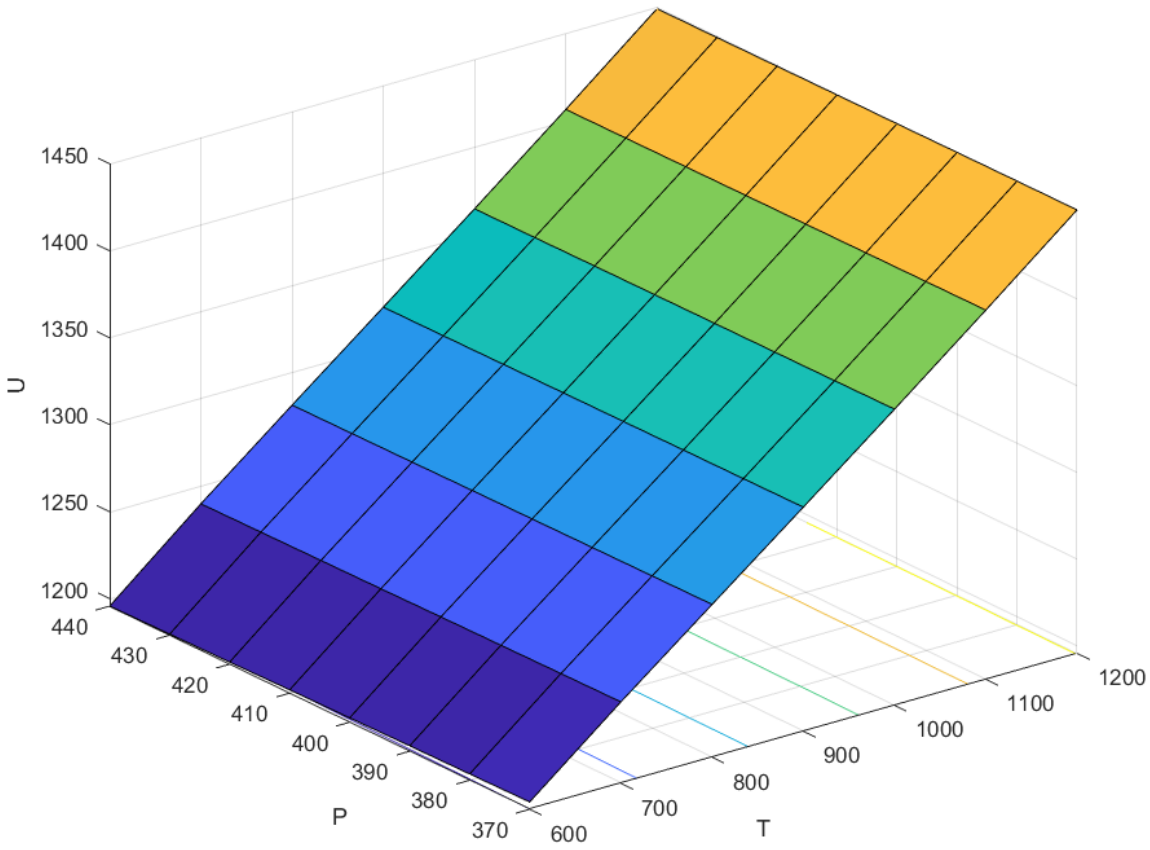
Problem 2

```
% PSET3_2
% Coded by Nigel F. Reuel on 9.2.2016
% This solves problem 5 of the problem set
%
% I used adobe acrobat to turn the image into text and then
cut and pasted
% the rows I needed into excel.  Here I import the data
table:
Data = xlsread('Steam.xls');
T = (6:12).*100;
P = (37:44).*10;
% Create surface plot
surfc(T,P,Data);
xlabel('T')
ylabel('P')
zlabel('U')
% Fit the values given
pi = [388 424 378 405 432 391];
ti = [743 908 1172 791 617 863];
display('The internal energies for these given values
are:')
U_out = interp2(T,P,Data,ti,pi)
```

Output

```
The internal energies for these given values are:

U_out = 1259.5    1326.5    1438.6    1278.8    1203.6    1308.9
```

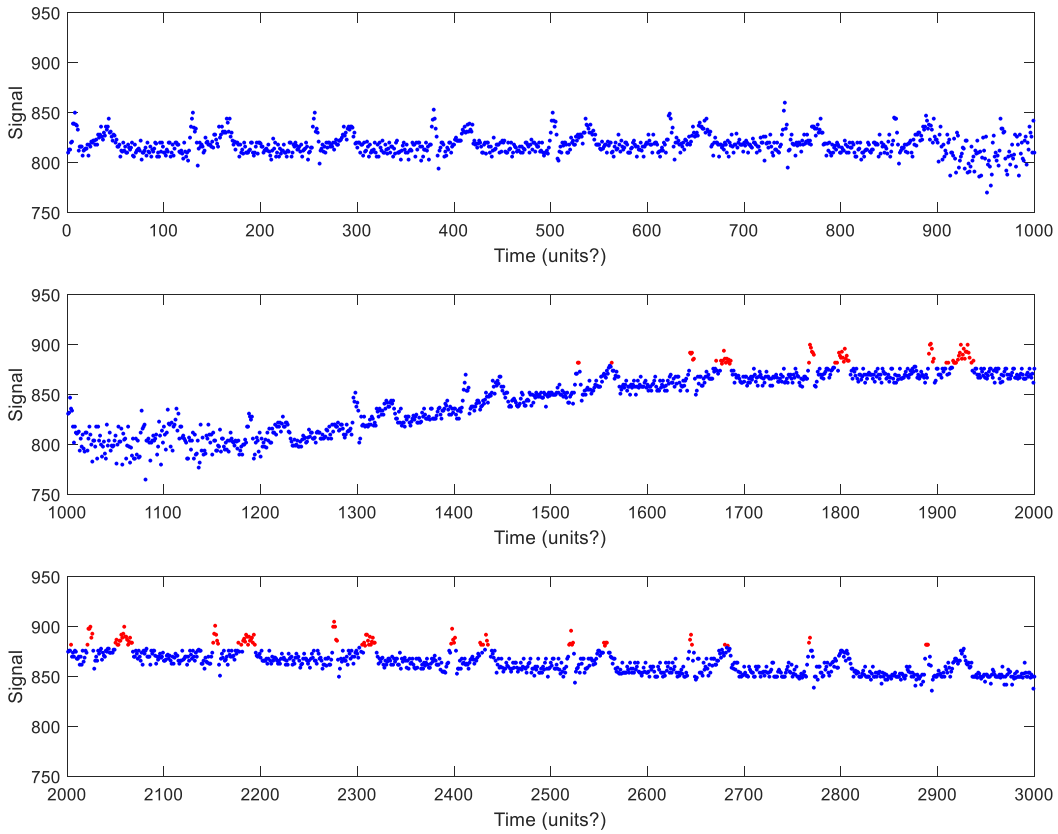

1450
1400
1350
1300
1250
1200
U
440
430
420
410
400
390
380
370
P
600
700
800
900
1000
1100
1200
T

Problem 3

```
% PSET 3_3
% Coded by Nigel F. Reuel on 9.7.2016
% This solves problem 3 of the third PSet
%
% Read in the EKG vector
Data = csvread('data.csv');
% Divide the data into three equal sections:
N = length(Data)/3;
x = 1;
% Like all REAL numerical problems, there are multiple
routes to the
% solution.  Here are a few:
% Slow Method - plot each point (take block comment off if
you want to run)
%{
while x <= length(Data)
    % Define the subplot position based on the point
location
    pos = ceil(x/N);
    subplot(3,1,pos)
    y = Data(x,1);
    if y > 880
        plot(x,y,'r.')
        ylim([750 950])
        hold on
    else
        plot(x,y,'b.')
        hold on
        ylim([750 950])
    end
    x = x+1;
end
xlabel('Time (units?)')
ylabel('Signal')
%}
% Fast Way - use logic on vectors
Dhigh = Data.*(Data>880);
Dlow = Data.*(Data<880);
tvec = 1:length(Data);
subplot(3,1,1)
plot(tvec(1:1000),Dhigh(1:1000),'r.',tvec(1:1000),Dlow(1:10
00),'b.')
```

```
ylim([750 950])
xlabel('Time (units?)')
ylabel('Signal')
subplot(3,1,2)
plot(tvec(1001:2000),Dhigh(1001:2000),'r.',tvec(1001:2000),
Dlow(1001:2000),'b.')
ylim([750 950])
xlabel('Time (units?)')
ylabel('Signal')
subplot(3,1,3)
plot(tvec(2001:end),Dhigh(2001:end),'r.',tvec(2001:end),Dlo
w(2001:end),'b.')
ylim([750 950])
xlabel('Time (units?)')
ylabel('Signal')
xlim([2000 3000])
```

Output

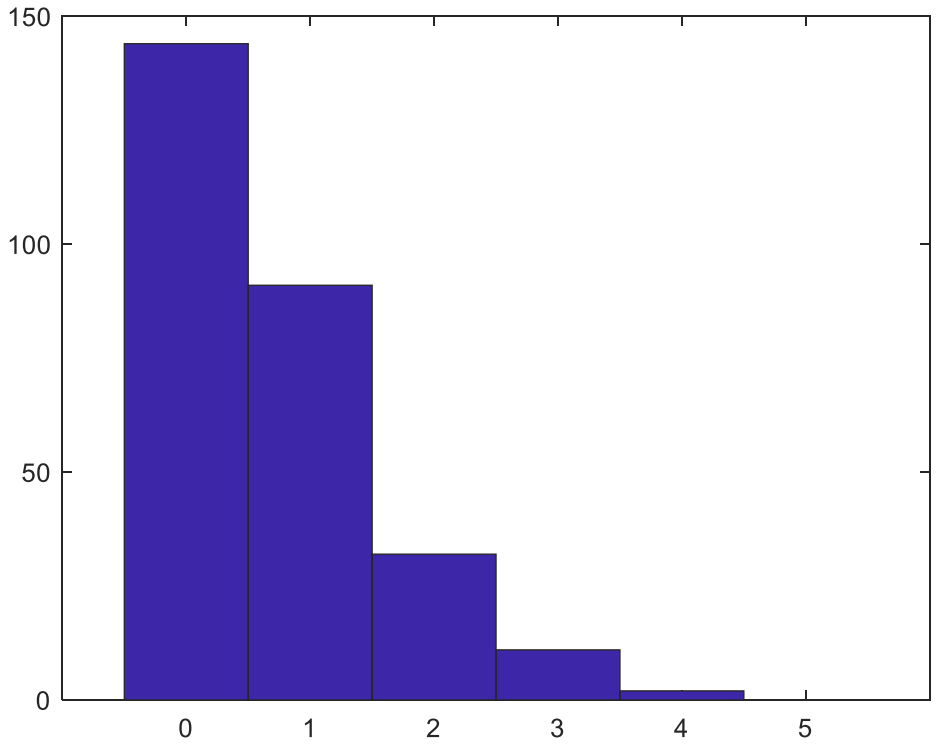


Problem 4

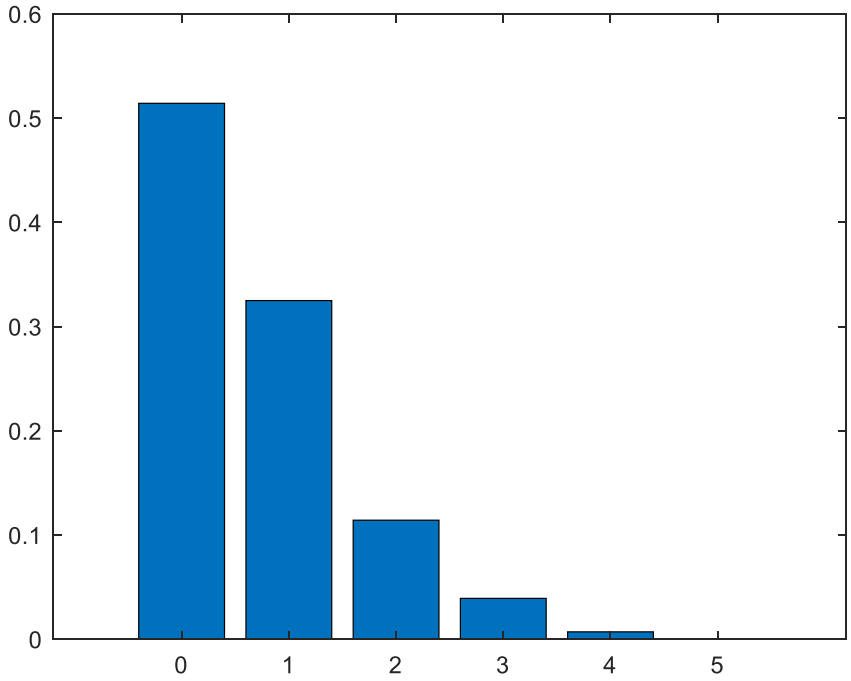
```
% PSET 3_4
% Coded by Nigel F. Reuel on 9.7.17
% Solves the Poisson Horse Kick Problem
% Part a)
[num,txt,~] = xlsread('HorseKickData.xlsx');
Data = num(:,2:end);
DataVec = Data(:);
bins = [0 1 2 3 4 5];
counts = hist(DataVec, bins);
% Part b and c)
Param = poissfit(DataVec);
% Adjust the histogram to show probability density
nc = length(DataVec);
Prob = counts./nc;
figure
bar(bins,Prob)
x = [0,1,2,3,4,5];
y = poisspdf(x,Param);
hold on
plot(x,y,':r')
hold off
% Part d)
% Why is the PDF undefined for non-integer values?
     % Answer - Poisson is used for EVENT counting...A
fractional event is un-defined...
% What is the probability of a calvary having >4 deaths in
one year from
% horse kicks?
% Part e and d)
ProbDeath = 1-poisscdf(4,Param)
% What is the average number of samples before >4 occurance
occurs?
obsInterval = 1/ProbDeath
% Part f)
% Simulate this probability with random number generator
NS = 1000;
CountVec = zeros(NS,1);
for i = 1:NS
    flag = 0;
    count = 1;
    while flag == 0
        if poissrnd(Param)> 4
```

```
            flag = 1;
            CountVec(i,1) = count;
        end
        count = count + 1;
    end
end
figure
hist(CountVec)
```

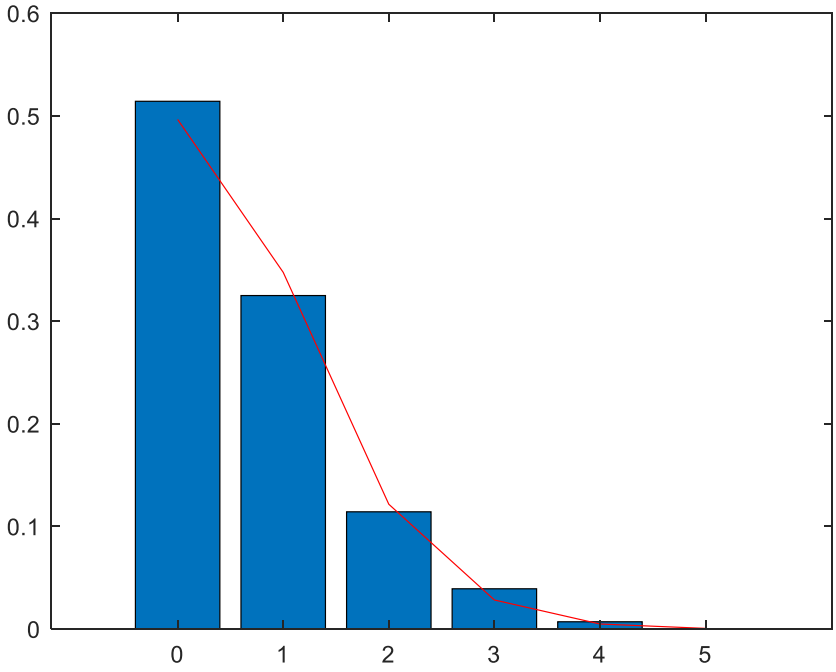
Part A



Part B



Part C

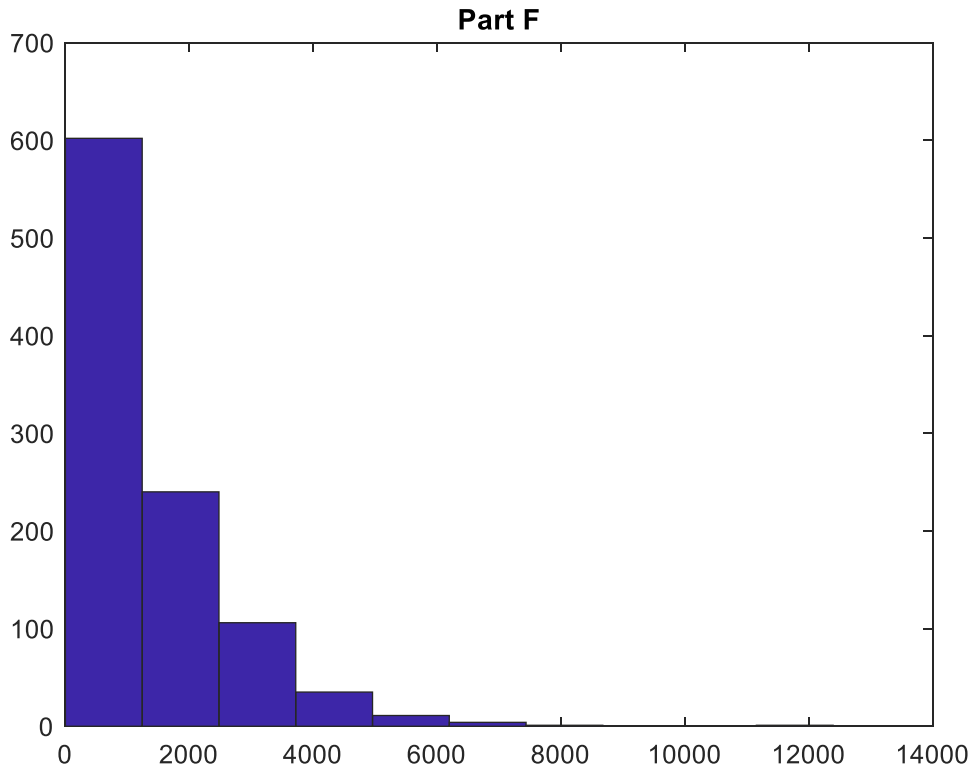


Part E

```
ProbDeath = 7.8554e-04

obsInterval = 1.2730e+03
```

Part F



Problem 5

```
% PSET 3_5
% Coded by Nigel F. Reuel on 9.8.2016
% This answers the group problem on the third problem set
%
% Preallocate memory for the error of the four functions
(C1 = 1/(1-x), C2
% = exp(x), C3 = cos(x), and C4 = sin(x)
Error = zeros(101,4);
% Start all the sum values at zero
S1 = 0;
S2 = 0;
S3 = 0;
S4 = 0;
% Evaluate at x = 4;
x = 0.5;
% Evaluate the true value for the four functions
T1 = 1/(1-x);
T2 = exp(x);
T3 = cos(x);
T4 = sin(x);
for i = 0:100
    S1 = S1 + x^(i);
    S2 = S2 + x^(i)/factorial(i);
    S3 = S3 + (-1)^(i)*x^(2*i)/factorial(2*i);
    S4 = S4 + (-1)^(i)*x^(2*i+1)/factorial(2*i+1);
    % Compute the error for the four functions
    E1 = abs((S1-T1)/T1)*100;
    E2 = abs((S2-T2)/T2)*100;
    E3 = abs((S3-T3)/T3)*100;
    E4 = abs((S4-T4)/T4)*100;
    % Place the errors into our storage matrix
    Error(i+1,1) = E1;
    Error(i+1,2) = E2;
    Error(i+1,3) = E3;
    Error(i+1,4) = E4;
end
% Generate the loglog plot
n = (0:1:100);
loglog(n,Error)
legend('1/(1-x)','e^x','cos(x)','sin(x)')
grid on % Personal preference when using log-log plots
% Make the marker
```

```
txt = '\leftarrow This function needs most terms at x =
0.5';
text(25,10^-5,txt)
```

Output