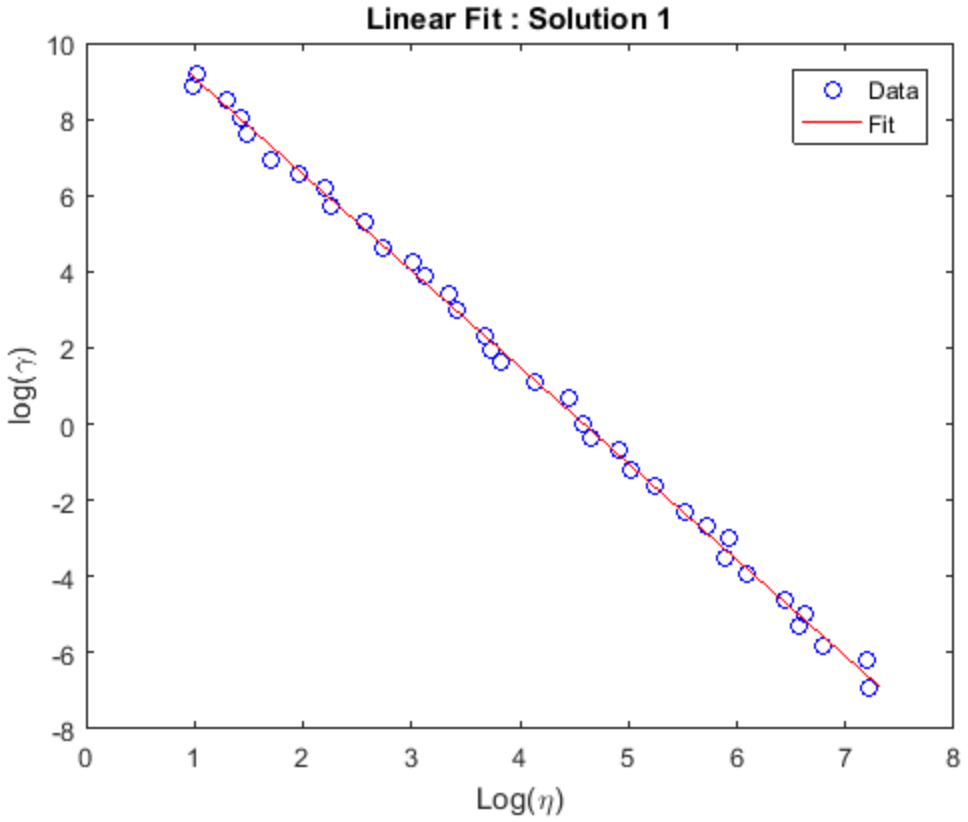
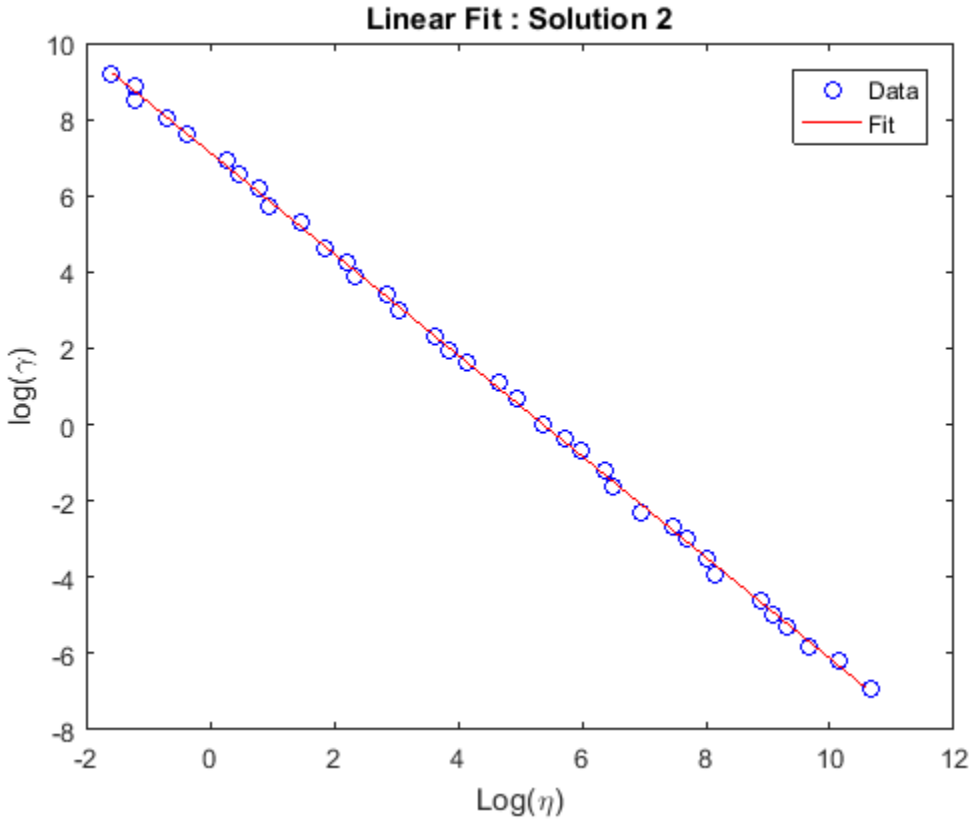
```
function problem4
%Navneet Singh (nsingh1@andrew.cmu.edu)
%HW-4 Prb 4

clc       %clear screen
clear all %clearing all stored variables
close all %close previous plots

%loding given data
load('midterm_data.csv')

%Extracting values
shr = midterm_data(:,1);
vis1 = midterm_data(:,2);
vis2 = midterm_data(:,3);

%Part (a)
%By taking log, we have linearised the system. System is represented
 as
% Ax=b.

%Solution 1
%Defining A and b matrices.
A = zeros(size(shr,1),2);
A(:,1) = 1;
A(:,2) = log(shr);
b = log(vis1);

%solving unknowns using pseudo-inverse
c = ((inv(A'*A))*A')*b;

fprintf('\nUsing Linearising\nFor solution 1,\nValue of K = %f\nValue
 of n = %f',exp(c(1)),c(2))

%Calculating fir from our data
fit = c(1) + c(2)*log(shr);
figure
plot(log(vis1), log(shr),'bo',fit,log(shr),'r');
title('Linear Fit : Solution 1')
legend('Data','Fit')
xlabel('Log(\eta)')
ylabel('log(\gamma)')

%Solution 2
%Defining A and b matrices
A = zeros(size(shr,1),2);
A(:,1) = 1;
A(:,2) = log(shr);
b = log(vis2);

c = ((inv(A'*A))*A')*b;
```

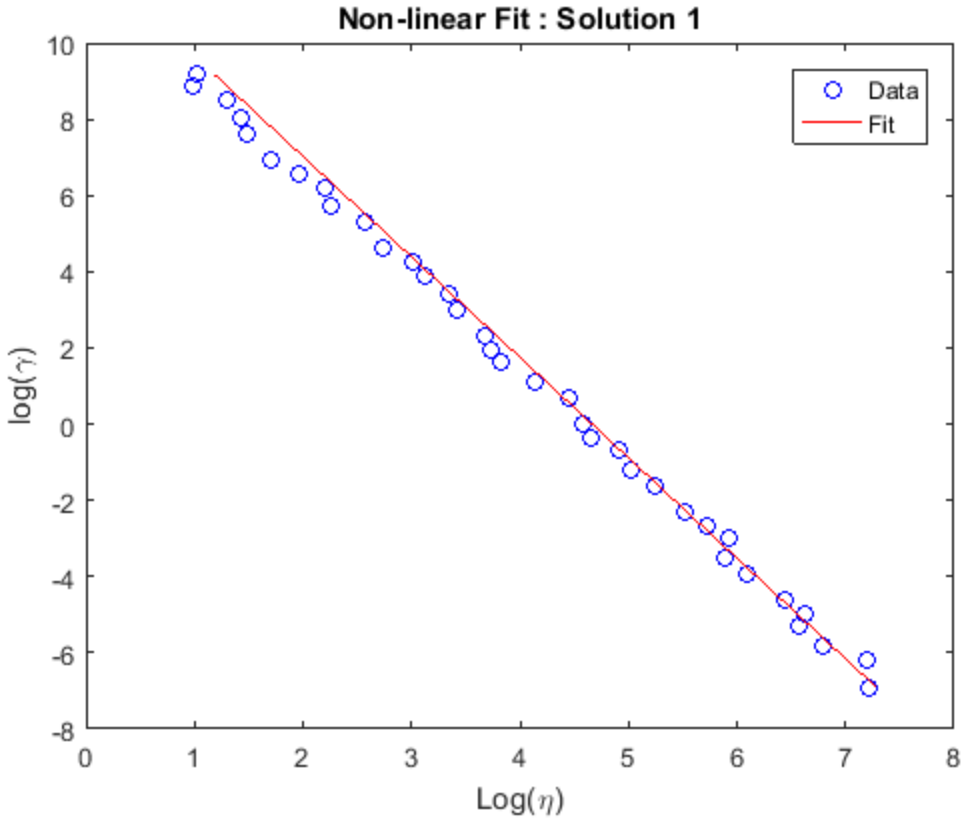
```
fprintf('\n\nFor solution 2,\nValue of K = %f\nValue of n =
 %f',exp(c(1)),c(2))

fit = c(1) + c(2)*log(shr);
figure
plot(log(vis2), log(shr),'bo',fit,log(shr),'r');
title('Linear Fit : Solution 2')
legend('Data','Fit')
xlabel('Log(\eta)')
ylabel('log(\gamma)')

%Non-linear optimization

%Solution 1
guess = [1,-1];
options = optimoptions(@fminunc,'display','off','Algorithm','quasi-
newton');
sol = fminunc(@obj, guess,options);

viscf = sol(1)*(shr.^sol(2));
fprintf('\n\nUsing non-linear optimization,\nFor solution 1,\nValue of
 K = %f\nValue of n = %f',sol(1),sol(2))

figure
plot(log(vis1), log(shr),'bo',log(viscf),log(shr),'r');
title('Non-linear Fit : Solution 1')
legend('Data','Fit')
xlabel('Log(\eta)')
ylabel('log(\gamma)')

%Our objective (cost) function that we have to minimize
function f = obj(x)
     err =  vis1 - x(1)*(shr.^x(2));
     f = 0.5*dot(err,err);
end


%non-linear2

guess = [5,-1];
options = optimoptions(@fminunc,'display','off','Algorithm', 'quasi-
newton');
sol2 = fminunc(@obj2, guess,options);

viscf2 = sol2(1)*(shr.^sol2(2));
fprintf('\nFor solution 2,\nValue of K = %f\nValue of n =
 %f',sol2(1),sol2(2));

figure
plot(log(vis2), log(shr),'bo',log(viscf2),log(shr),'r');
title('Non-linear Fit : Solution 2')
legend('Data','Fit')
xlabel('Log(\eta)')
ylabel('log(\gamma)')
```

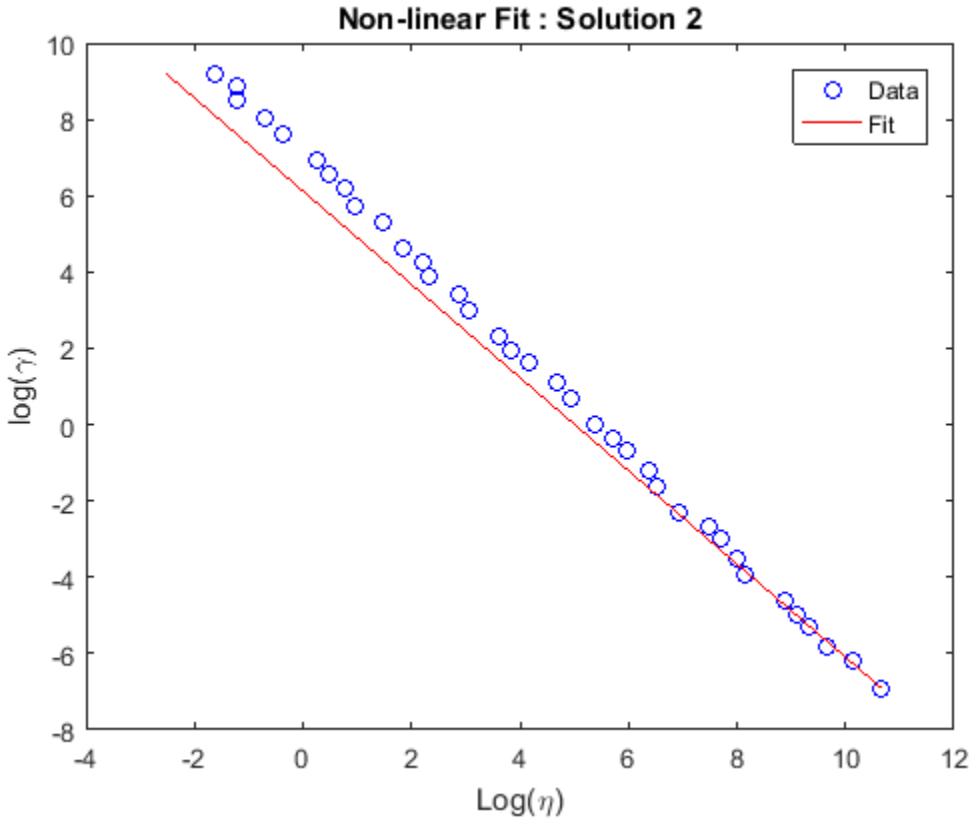
```
%Our objective (cost) function that we have to minimize
function f = obj2(x)
     err =  vis2 - x(1)*(shr.^x(2));
     f = 0.5*dot(err,err);
end


end
```

```
Using Linearising
For solution 1,
Value of K = 98.637078
Value of n = -0.395376

For solution 2,
Value of K = 216.389546
Value of n = -0.753974

Using non-linear optimization,
For solution 1,
Value of K = 106.089919
Value of n = -0.379400
For solution 2,
Value of K = 149.818454
Value of n = -0.818576
```


Linear Fit : Solution 1
Data
Fit
log(γ)
Log(η)
10
8
6
4
2
0
-2
-4
-6
-8
0
1
2
3
4
5
6
7
8


Linear Fit : Solution 2
Data
Fit
log(γ)
Log(η)
10
8
6
4
2
0
-2
-4
-6
-8
-2
0
2
4
6
8
10
12


Non-linear Fit : Solution 1
Data
Fit
log(γ)
Log(η)
10
8
6
4
2
0
-2
-4
-6
-8
0
1
2
3
4
5
6
7
8


Non-linear Fit : Solution 2
Data
Fit
log(γ)
Log(η)
10
8
6
4
2
0
-2
-4
-6
-8
-4
-2
0
2
4
6
8
10
12

*Published with MATLAB® R2016a*