Table of Contents


```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% ENGR 132
% Program Description
% You must use create a program to predict Uranium recovery from sea
 water.
%
% Assigment Information
%   Assignment:   PS 05, Problem 1
%   Author:    Ethan Hotson, ehotson@purdue.edu
%   Team ID:   009-01
%   Paired Partner:  cjennewe@purdue.edu
%                       wpopovic@purdue.edu
%                       omeccia@purdue.edu
%
%   Contributor:  Name, login@purdue [repeat for each]
%   Our contributor(s) helped us:
%     [ ] understand the assignment expectations without
%         telling us how they will approach it.
%     [ ] understand different ways to think about a solution
%         without helping us plan our solution.
%     [ ] think through the meaning of a specific error or
%         bug present in our code without looking at our code.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

INITIALIZATION

```
u_data = csvread('Data_uranium_adsorption.csv',1,0); %Imports the
 Uranium data into matlab
```

```
u_dataX = u_data(:,1); %X column of u data
u_dataY = u_data(:,2); %Y column of u data
```

SUBPLOT FIGURE

```
%This will display the 4 types of graphs for the data given in the
 Uranium
%data file.

figure;
subplot(2,2,1)
plot(u_data(:,1),u_data(:,2),'.b'); %Linear scale
grid on;
xlabel 'Time (hrs)';
ylabel 'Uranium Uptake (\mug/g)';
title 'Linear Scale Uranium Data';

subplot(2,2,2)
semilogx(u_data(:,1),u_data(:,2),'.b'); %Log x scale
grid on;
xlabel '[Time (hrs)]';
ylabel 'Uranium Uptake (\mug/g)';
title 'Logarithmic X Scale Uranium Data';

subplot(2,2,3)
semilogy(u_data(:,1),u_data(:,2),'.b'); %Log y scale
grid on;
xlabel 'Time (hrs)';
ylabel '[Uranium Uptake (\mu/g)]';
title 'Logarithmic Y Scale Uranium Data';

subplot(2,2,4)
loglog(u_data(:,1),u_data(:,2),'.b'); %Log Log scale
grid on;
xlabel '[Time (hrs)]';
ylabel '[Uranium Uptake (\mug/g)]';
title 'LogLog Scale Uranium Data';
```

LINEARIZATION

```
linear_u = [log10(u_dataX),log10(u_dataY)];
u_reg = polyfit(linear_u(:,1), linear_u(:,2),1);

fprintf("The Linear Equation for the data is: Log[Uranium Uptake
 (mu*g/g)] = %.4f * (Log[Time (hrs)]) + %.4f\n",u_reg(1),u_reg(2));
```

```
figure; %This will plot the linearized data with the linear trendline.
plot(linear_u(:,1),linear_u(:,2),'.b'); %Linearized scale
hold all;
grid on;
xlabel 'Log[Time (hrs)]';
ylabel 'Lo[Uranium Uptake (\mug/g)]';
title 'LogLog Scale Uranium Data';
hline = refline(u_reg(:,1),u_reg(:,2));
hline.Color = 'r';
```

*The Linear Equation for the data is: Log[Uranium Uptake (mu*g/g)] =*
*0.4202 * (Log[Time (hrs)]) + 1.4526*

UPTAKE MODEL

```
%This prints the most general power equation
fprintf("The Power Equation of this Data is: [Uranium Uptake (\\mu*g/
g)] = %.4f * (Log[Time (hrs)]) ^ (%.4f)\n",u_reg(2),u_reg(1));
figure;
plot(u_data(:,1),u_data(:,2),'k.');
hold all;
grid on;
xlabel '[Time (hrs)]';
ylabel '[Uranium Uptake (\mug/g)]';
title 'Power Function Uranium Data';
y_data = 10 ^ (u_reg(2)) * (u_dataX .^ u_reg(1));
plot(u_dataX,y_data,'r-')
```

*The Power Equation of this Data is: [Uranium Uptake (\mu*g/g)] =*
*1.4526 * (Log[Time (hrs)]) ^ (0.4202)*

PREDICTIONS

```
%Predicts the value of the theoretical best fit function at 10, 100
 and 250
%hours.
tenHrs = 10 ^ (u_reg(2)) * 10 ^ u_reg(1);
hunHrs= 10 ^ (u_reg(2)) * 100 ^ u_reg(1);
twfiHrs= 10 ^ (u_reg(2)) * 250 ^ u_reg(1);
```

ANALYSIS

-- Q1

The data is best represented by the power model, as it represents the trend without changing either of the axis to linear. Additionally, the log log plot's linearity indicates that the power model is best.

-- Q2

After 10 hours, our model predicts a Uranium uptake of 74.6197 micrograms per gram of water. After 100 hours, our model predicts a Uranium uptake of 196.3796 micrograms per gram of water. After 250 hours a Uranium uptake of 288.6208 micrograms per gram of water is predicted. This last prediction is flawed, because if heavily extrapolates outside of our data, which is dangerous and likely innacurate.

ACADEMIC INTEGRITY STATEMENT

We have not used source code obtained from any other unauthorized source, either modified or unmodified. Neither have we provided access to our code to another. The script we are submitting is our own original work.

Published with MATLAB® R2018b