Numerical Analysis Lab: Least Squares Methods for Curve Fitting

Name _____

1. In MATLAB change to your N:\m439 directory (with **cd N:\m439**)
2. In Theorem 5.2, page 257, we developed a method for curve fitting called the least squares power fit.
The curve $y = Ax^M$ is a least squares power fit for the data points (x_1, y_1), (x_2, y_2), ..., (x_N, y_N) with power M, a known constant where A is given by

$$A = \frac{\sum_{k=1}^{N} x_k^M y_k}{\sum_{k=1}^{N} x_k^{2M}}$$

We can apply this model to determining the gravitational constant (as in example 5.3 and exercise 2).
In this the known constant is M = 2 and the gravitational constant g will be 2A.
The *x* coordinates represent the elapsed time and the *y* coordinates represent the observed distance at that time.
It is based on the following formula discovered by Gallileo for how far a body will fall under gravity if air resistance is ignored.

$d = (1/2)gt^2$

where

d is the vertical distance traveled
g is the acceleration of gravity on the planet's surface
t is the amount of time the body falls.

"*The ancients believed that if two object were dropped, the heavier one would fall faster (and hence travel further in a given time). By dropping cannon balls from the leaning tower at Pisa, Galileo showed that all objects will travel the same distance and speed regardless of the mass / weight of the body*". (Reference: http://www.krysstal.com/formulas.html)

3. In MATLAB we can compute A for the power fit very easily using some of MATLAB's commands for matrices:

1. **X .^ M**
will raise all of the elements in the matrix X to the M'th power.
2. **X .* Y**
will multiply the corresponding elements of the matrix X with elements of the matrix Y
3. **sum(X)**
will sum all of the elements of the vector X

Exercise 1 Part 1: Write a MATLAB function (store in m-file) that will compute the A for the power fit using the formula $A = \dfrac{\sum_{k=1}^{N} x_k^M y_k}{\sum_{k=1}^{N} x_k^{2M}}$.

4. The above three commands make the task easy. (Your program will be about four lines of code or less, depending on how you write it
(You could actually do in one line, but that is not necessarily desirable because of clarity).

5. Name the function **lspower** (So save it on you N: drive under the name **lspower.m**)
The code for the function in your m-file should be:

```
function A = lspower(X,Y,M)
% Provide your name
% and appropriate comments documenting what function does
% and what A, X, Y, and M are
MATLAB commands necessary to compute and assign appropriate value to A
```

6. Test your function by calling it on the data for example 5.3, page 258 (table at top).
Initialize vector X to store the times, vector Y the distances. Then call the function with

```
A = lspower(X,Y,2)
g = 2 * A
```

7. Compare your answer to the answers reported in the text (**A = 4.9073**, **g = 2*A = 9.8146 (note book is wrong for g**). If you don't have this answer, debug!!

Exercise Part 2:

8. Now modify your function to include calculation of the root-mean-square error E_2. = $\left(\dfrac{1}{N}\sum_{k=1}^{N}(Ax_k^M - y_k)^2\right)^{1/2}$.
Change the heading of your function to
function [A, E] = lspower(X,Y,M)
and add the code to also compute and assign to E the root-mean-square error.
For this you will need to divide by the number of elements in X, so you will need to use the MATLAB function **length(X)**
that returns the number of elements in the vector X .You can use **sqrt** for the square root function.

9. After making the necessary changes and saving your m-file, test it with:

```
[A, E] = lspower(X,Y,2)
```

10. Plot the **data points** and the approximating **power fit function** on the **same graph**, using

```
>> clf                          Clear any previous graph
>> plot(x,y,'r*')               Plot the original data points in red.
>> hold on;
>> ezplot( '4.9073*x^2', [0,1.2] )   Plot the power fit curve for 0<x<1.2
```

11. a) Put a title on the graph with your name with **title ('Least Squares Power Fit -- your name')** . Print the graph.

b) On the back of your printout write the answers to the exercise below. Also print your m-file **lspower.m**

c) Use your function to calculate the answers to Algorithms and Programs 2 b and 3, page 262.
(Note for problem 3 the exponent should be M = 3/2 (or 1.5). (Write answers on above printout).