

CSE 6010 - Computational Problem Solving

Assignment 1 - Power Law with Least Squares Fitting

Ting Liao / GTID: 903278773

Slide 1

I structured my power law program by following these steps to translate the mathematics into code.

First, I put “`#include <stdio.h>`” and “`#include <math.h>`” at the top of my code, which can let me use the function of “`printf`” and the math function. Then, setting the program environment as “`int main(int argc, const char * argv[])`” to allow me to execute the program. Besides, I found that there were 19 items per data set, so I used “`#define arr_size 19`” to define the global `arr_size` to be 19.

Second, after setting up two data arrays (mass and max speed), I used three “for-loops” to help me deal with all cumulative mathematics. For example, convert mass and max speed into $\log(\text{mass})$ and $\log(\text{max speed})$, calculate the M matrix and the b vector. Due to that I have 19 rows of data, which means I have to convert or accumulate values for 19 times. In order to let the for-loop to run 19 times, I initialize `i` to be 0, and the testExpression is `i < arr_size` (`arr_size = 19`), so the range is from 0 to 18 (`i < 19`), and using `i++` as the updateStatement, so the for-loop will keep executing until `i = 19`. Now we can start to implement the math function into the for-loop.

For dealing with converting original data into $\log(\text{data})$, the M matrix and the b vector, I could use the same setting of for-loop because they all need to be executed 19 times. Then, I place in each math function into each for-loop. Take `x[i] = log(mass[i])` as an example. After placing this code into the for-loop, we can get the double `x[19]` array after executing 19 times. Besides, I can use the function `log()` because of “`#include <math.h>`”, same as `pow()` for `m22` and `exp()` calculating the coefficient.

What's more, the function `printf()` allows me to check each part of my code is correct or not by printing out the result value of each part. In the end, we can use the function `printf()` to print out the final result (exponent and coefficient) of our program.

Slide 2

Using Jupiter Notebook to make the plot via Python. As the figure and the Python code shown below, I believe that my program of power law works correctly. Also, in the C program, powerlaw.c, I have used a set of dummy data of mass and max speed to check whether I can derive the correct exponent and coefficient.

```
In [2]: # import numpy and matplotlib
import numpy as np
import matplotlib.pyplot as plt
```

```
In [11]: # For the mass & max speed data plot
# x axis values - mass
x = [4.22E-04, 4.50E-04, 3.90E-04, 0.0138, 5.22E-06, 6.86E-06, 1.30E-06, 1.76E-07,
      7.47E-07, 7.89E-08, 2.05E-07, 9.0E-05, 1.70E-04, 0.0015, 4.73E-05, 0.002,
      0.002, 6.52E-05, 1.01E-04]
# corresponding y axis values - max speed
y = [36, 27, 25.92, 36, 6.48, 4.8, 3.6,
      3.24, 6.84, 3.06, 4.32, 36, 36,
      25.56, 6, 19.08, 18, 10.8, 23.76]

# naming the x axis
plt.xlabel('Mass')
# naming the y axis
plt.ylabel('Max Speed')

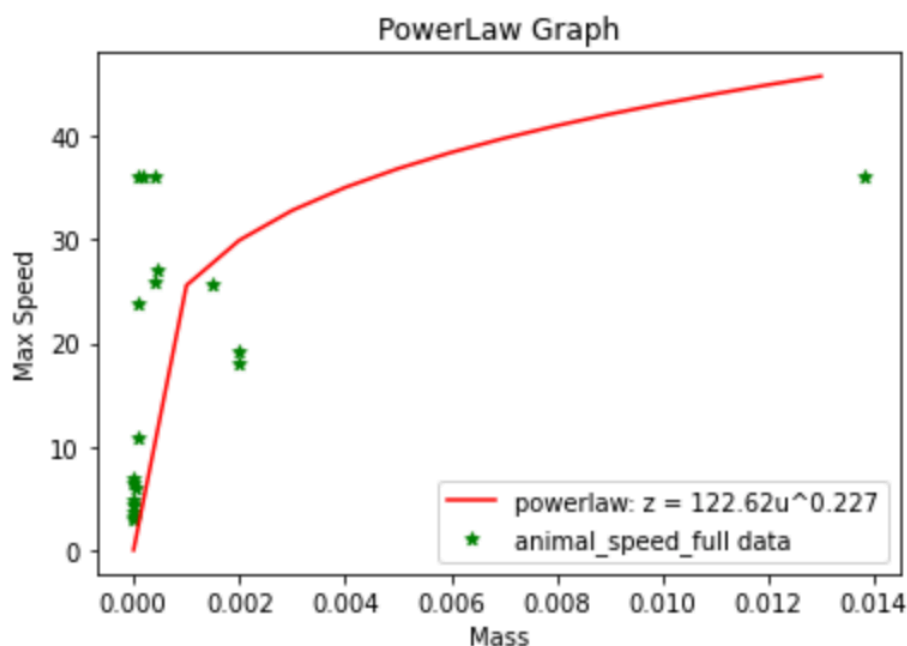
# For the powerlaw equation
# setting the x - coordinates
powerlaw_x = np.arange(0, 0.014, 0.001)
# setting the corresponding y - coordinates
powerlaw_y = 122.62*(powerlaw_x**0.227)

# plotting the points
plt.scatter(x, y, label= "animal_speed_full data", color= "green",
            marker= "*", s=30)
plt.plot(powerlaw_x, powerlaw_y, color= "red", label = "powerlaw: z = 122.62u^0.227")

# giving a title to my graph
plt.title('PowerLaw Graph')

# showing legend
plt.legend()

# function to show the plot
plt.show()
```



Slide 3

For this assignment, I think I have two most challenging aspects. One is that I have never used C programming before, so I have to struggle on dealing with the code environment setting, the format of data structures in C. Also, not pretty sure whether the extension package “Code Runner”, which help me compiled and ran the program, is my compiler in the beginning, but luckily got the answer from other classmates. Second, I haven’t written any documentation for explaining the program structure before, so I spent a while on how to write it.

Basically, the first assignment is quite simple and learning a new language is still a lot of fun. Except for being not familiar with C programming language and the documentation, everything else is fine.