Fitting Power Law Correlations in MATLAB

Introduction

One of the most frequent types of correlation in the sciences is the *power law* correlation. Two variables t and n are said to obey a power law correlation if the relationship between them can be expressed as

$$t = c\,n^p. \tag{1}$$

In Equation (1), p is called the power (or exponent) and c is the proportionality constant. The purpose of this document is to describe how to use MATLAB to perform a power law fit to experimental data.

How can the values of c and p in Equation (1) be determined? There are several ways to do this, but the easiest is to use the notion of linear regression or best fit in the logarithmic domain. To do this, Take the base-10 log of both sides of equation (1) and do some manipulation:

$$\begin{aligned}
\log_{10}(t) &= \log_{10}(c\,n^p) \\
\log_{10}(t) &= \log_{10}(c) + \log_{10}(n^p) \\
\log_{10}(t) &= \log_{10}(c) + p\,\log_{10}(n)
\end{aligned}$$

Now set $T = \log_{10}(t)\,, N = \log_{10}(n)$ and $b = \log_{10}(c)\,$. Then the last equation becomes

$$T = p\,N + b. \tag{2}$$

Equation (2) is the equation of a line with slope p and y-intercept b. This results in the following process for obtaining the desired correlation:

a) Take $\log_{10}$ of the data values.

b) Perform a linear least squares fit of the logarithmic data.

c) The slope of the line computed in b) will be the power p. The value of the proportionality constant c can be computed from $c = 10^b$.

In practice, you can use any base logarithm but base-10 is the most commonly used in scientific experiments.

Example

Enter the following program in the MATLAB editor: Save this function as `logfit.m`. Test that it is working properly by entering and running the following script:

```
% Driver script for the power law fitting
clear;
close all;
 n = [1.2 1.3 1.6 1.9 2.0 2.1]';
 t = [13.2 21.9 81.1 239.5 330.9 450.0]';
 N = log10(n);
 T = log10(t);
 coeff = polyfit(N,T,1);
 power = coeff(1);
 const = 10^(coeff(2));
```

If you did everything correctly, you should have obtained $p \approx 6.3$, $c \approx 4.2$. You can now use the values of p and c to predict the value of t for a given n (or vice versa). For example, if $n = 1.7$, you can compute t from

$$t = 4.2 \cdot 1.7^{6.3} \approx 118.87 \tag{3}$$