Computing Assignment 2: GE Timing Test

By Kai Sackville-Hii Jan 28, 2019

power_law_plot.m (hl -- Jan. 2019)

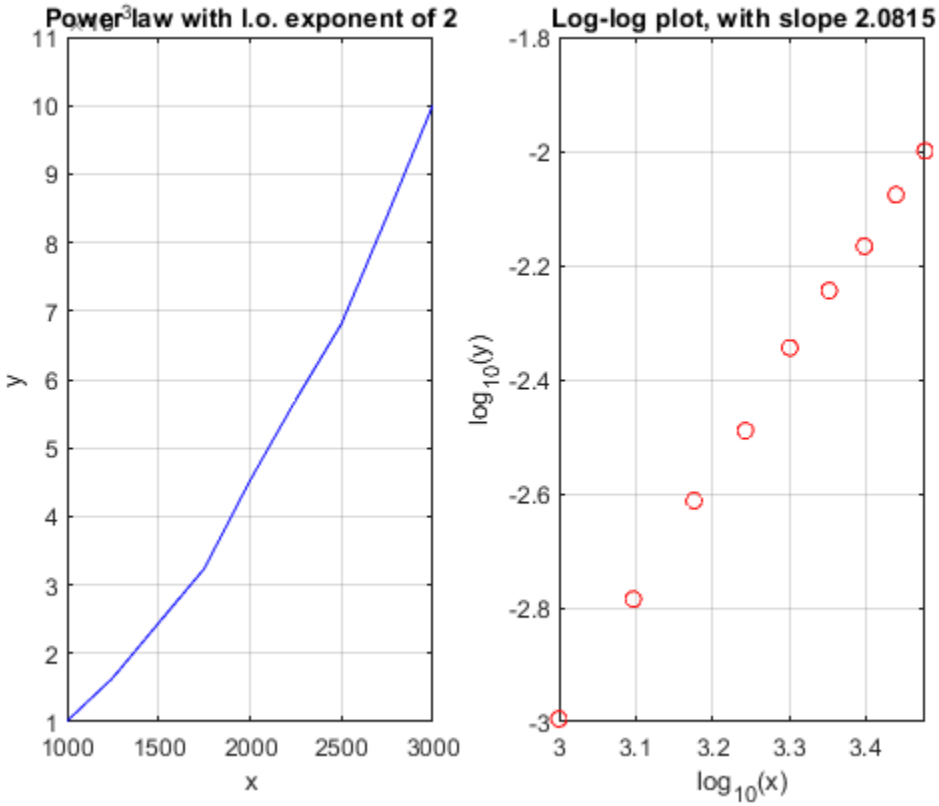
Purpose : To highlight how the slope of a log-log plot can be used to determine the leading order behaviour of a power law relationship

Instructions : Run the code as is, and check the output in the command line. How does lo_exp compare to the best fit slope? experiment by changing lo_exp.

```
clear;
load('times.mat');
x = 1000:250:3000;

% Our power law we wish to analyze
% change this to whichever average value in question
y = avg_sparse_time_array;

% Experiment by changing this value
lo_exp = 2;

% Compute log values
logx = log10(x);
logy = log10(y);

% Best fit line to log data
p = polyfit(logx,logy,1);

% Output
% How does the slope of the best fit compare to lo_exp?
display(['Leading order power law is : ',num2str(lo_exp)])
display(['Slope of best fit line is  : ',num2str(p(1))])

% Plotting
figure(1)
clf; hold on;

% Plot of raw data
% Is it clear exactly what power law is being plotted?
subplot(1,2,1)
plot(x,y,'b')
grid on
xlabel('x')
ylabel('y')
title(['Power law with l.o. exponent of ', ...
       num2str(lo_exp)])

% Plot of log data
% It should be clear that this relationship is linear
subplot(1,2,2)
plot(logx,logy,'ro')
```

```
grid on
xlabel('log_{10}(x)')
ylabel('log_{10}(y)')
title(['Log-log plot, with slope ', num2str(p(1))])
```

```
Leading order power law is : 2
Slope of best fit line is  : 2.0815
```



Published with MATLAB® R2018b