Table of Contents


```
x = linspace(1,5,5);
y = [0.5 1.7 3.4 5.7 8.4];

% given power equation is y = a(x.^b)
% After applying log OBS we get lny = ln(a) + bln(x)
% where slope is b and intercept is ln(a)
% and after applying antilog

lnx = log(x);
lny = log(y);

[m,i] = line_regression(lnx,lny);
b = m;
a = exp(i);
```

function handles

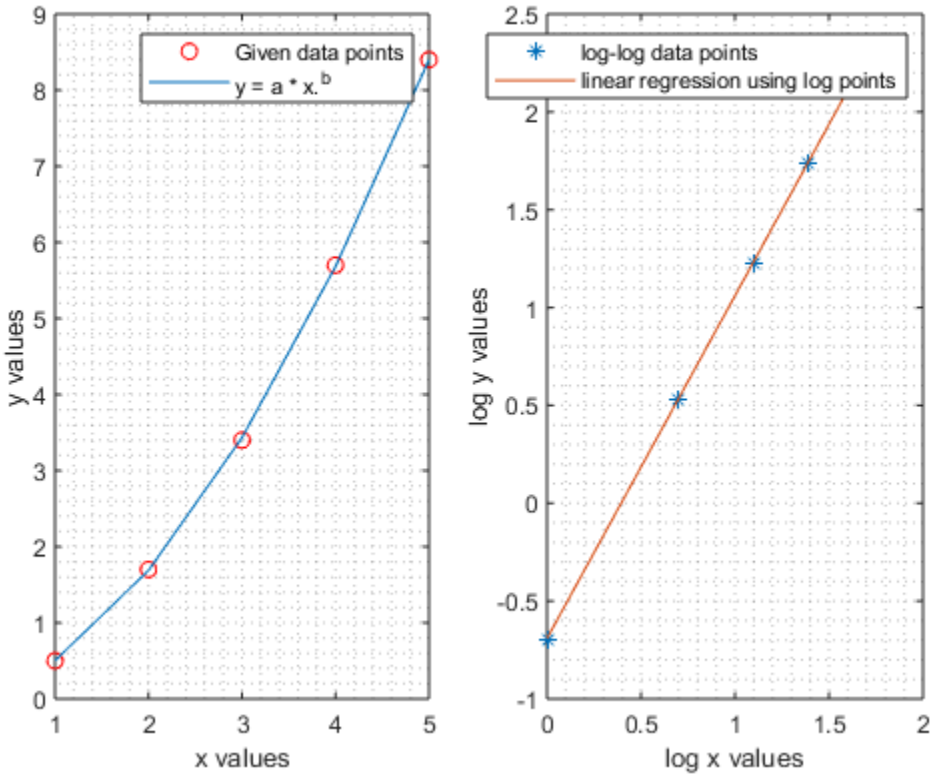
```
y_logregression =@(x) m*lnx + i;
y_actual =@(x) a*(x.^b);
```

plots

```
%convert this into log-log plot.
subplot(1,2,1)
plot(x,y,'ro');
hold on;
grid minor;
xlabel('x values');
ylabel('y values');
plot(x,y_actual(x))
legend('Given data points', 'y = a * x.^b');
hold off;

subplot(1,2,2)
plot(lnx,lny,'*');
hold on;
grid minor;
xlabel('log x values');
ylabel('log y values');
plot(lnx,y_logregression(lnx));
legend('log-log data points','linear regression using log points')
```

```
hold off;
```



to find value

```
yactual = y_actual(6)
```

```
yactual =

   11.5581
```

function calls

```
function [m,i] = line_regression(x,y)
     m = sum( y.*(x - mean(x)))/sum(x.*(x - mean(x)));
     i = mean(y) - m*mean(x);
end
```

Published with MATLAB® R2018b