

MATH 2270-1
MAPLE PROJECT 2a - Power Laws
October 24, 2005

Using logarithms to discover power laws:

Here's how to use a least squares method to test whether a power law can adequately describe a correlation between two different variables. Suppose we have n data points,

$$\{[x_i, y_i]\}, i = 1 \dots n,$$

and we expect a power law

$$y = C x^p$$

to approximately relate the y_i 's to the x_i 's. Here p is the power, and we can call C the proportionality constant. Then we expect the logarithms of the original data points,

$$\{[\ln(x_i), \ln(y_i)]\}$$

to satisfy

$$\ln(y) = \ln(C) + p \ln(x).$$

In other words, the ln-ln data should satisfy the equation of a line, having slope equal to p and vertical axis intercept equal to $\ln(C)$. So we can take the least squares fit for the ln-ln data, and then use the slope and intercept to deduce p and C in the power law: take the power p equal to the slope of the least square fit to the ln-ln data, and take C to equal to the exponential of the vertical intercept, for the $\ln(y)$ vs. $\ln(x)$ line.

Example:

We work exercise #40 in section 5.4 of the text, page 231. This example relates to Johannes Kepler's discovery several hundred years ago that for circular planetary orbits the period of revolution is proportional to the radius or the orbit, to the $3/2$ power. Later Newton showed that the full set of Kepler's observational Laws was consistent (only) with the famous "inverse-square law" for gravitational attraction.

```
[ > restart:
> with(linalg):with(plots):
    #we use these two command libraries
Warning, the protected names norm and trace have been redefined and unprotected
Warning, the name changecoords has been redefined
> S:=[[.387,.241],[1,1],[5.20,11.86],
      [19.18,84.0],[39.53,248.5]]:
    #the actual data, in table form
A1:=convert(S,matrix); #and as a matrix
```

$$A1 := \begin{bmatrix} 0.387 & 0.241 \\ 1 & 1 \\ 5.20 & 11.86 \\ 19.18 & 84.0 \\ 39.53 & 248.5 \end{bmatrix}$$

Just for "fun" I have reproduced the table from the text. The units of distance are "Astronomical Units", i.e. the distance between the earth and the sun. The time units are (Earth) years.

```
> planets:=matrix(5,1,['Mercury','Earth','Jupiter',
  'Uranus','Pluto']):
headers:=matrix(1,3,['Planet','Mean radius','mean period']):
booktable:=stackmatrix(headers,augment(planets,A1));
#the table in the book
```

| <i>Planet</i> | <i>Mean radius</i> | <i>mean period</i> |
|----------------|--------------------|--------------------|
| <i>Mercury</i> | 0.387 | 0.241 |
| <i>Earth</i> | 1 | 1 |
| <i>Jupiter</i> | 5.20 | 11.86 |
| <i>Uranus</i> | 19.18 | 84.0 |
| <i>Pluto</i> | 39.53 | 248.5 |

Now that we've had our fun, let's get down to the task of discovering Kepler's power law, using the A1 matrix.

```
> A2:=map(ln, A1):
#the map command will apply a given function,
#in this case the natural logarithm, to each
#entry of the specified matrix. So the entries of
#A2 will be the ln of entries in A1. We used the
#map command when we made fractals!
```

```
> A3:=map(evalf,A2);
#get each decimal value (not necessary in
#this example, but needed for your work on BMI)
```

$$A3 := \begin{bmatrix} -0.9493305860 & -1.422958345 \\ 0. & 0. \\ 1.648658626 & 2.473171394 \\ 2.953868069 & 4.430816799 \\ 3.677059877 & 5.515442846 \end{bmatrix}$$

```
> rowdim(A3);
#rowdim computes the number of rows in
#a matrix. Of course, for this small matrix,
```

```
#we know there are five rows. Your B.M.I. matrix
#will be much bigger
```

5

We use $A2$ to construct the matrix A and right hand side b , for the least squares line fit. The first column of A will be the $\ln(x)$ values, the second column will be a vector of 1's and the vector b will be the corresponding $\ln(y)$ values, since we want a least squares line fit $\ln(y) = m \ln(x) + b$. (See class notes, or Example 5 on page 226 of the text.) Maple has commands to extract columns, augment matrices, etc.

```
> col2:=vector(rowdim(A3),1);
#this will be the second column, a vector of 1's
#for our least-squares line fit matrix
#(Out of habit I make this
#the second column of the matrix rather than the
#first, and then reverse m and b in the matrix equation.
#Sorry.) This command creates a vector
#with number of entries equal to the first argument
#and makes each entry equal to the second argument
```

```
col2 := [1, 1, 1, 1, 1]
```

```
> A4:=delcols(A3,2..2);
#remove the second column of A3,
```

```
A4 :=
```

$$\begin{bmatrix} -0.9493305860 \\ 0. \\ 1.648658626 \\ 2.953868069 \\ 3.677059877 \end{bmatrix}$$

```
> A:=augment(A4,col2); #This is our matrix for least squares,
#with the ln(x) terms in the first column, and 1's in
#the second.
```

```
A :=
```

$$\begin{bmatrix} -0.9493305860 & 1 \\ 0. & 1 \\ 1.648658626 & 1 \\ 2.953868069 & 1 \\ 3.677059877 & 1 \end{bmatrix}$$

```
> b:=delcols(A3,1..1); #This is the right-hand side for least
squares,
#it has the ln(y) values
```

$$b := \begin{bmatrix} -1.422958345 \\ 0. \\ 2.473171394 \\ 4.430816799 \\ 5.515442846 \end{bmatrix}$$

The next three commands find the least-squares solution, as in section 5.4.

```
> ATA:=evalm(transpose(A)*A);
ATb:=evalm(transpose(A)*b);
linsolve(ATA,ATb);
#solve the system (ATA)x=(ATb)
```

$$ATA := \begin{bmatrix} 25.86540974 & 7.330255986 \\ 7.330255986 & 5 \end{bmatrix}$$

$$ATb := \begin{bmatrix} 38.79693508 \\ 10.99647269 \end{bmatrix}$$

$$\begin{bmatrix} 1.499816413 \\ 0.0004868907611 \end{bmatrix}$$

Alternately, we could multiply the right-hand side by the inverse matrix:

```
> evalm(inverse(ATA)*ATb);
```

$$\begin{bmatrix} 1.499816413 \\ 0.000486890 \end{bmatrix}$$

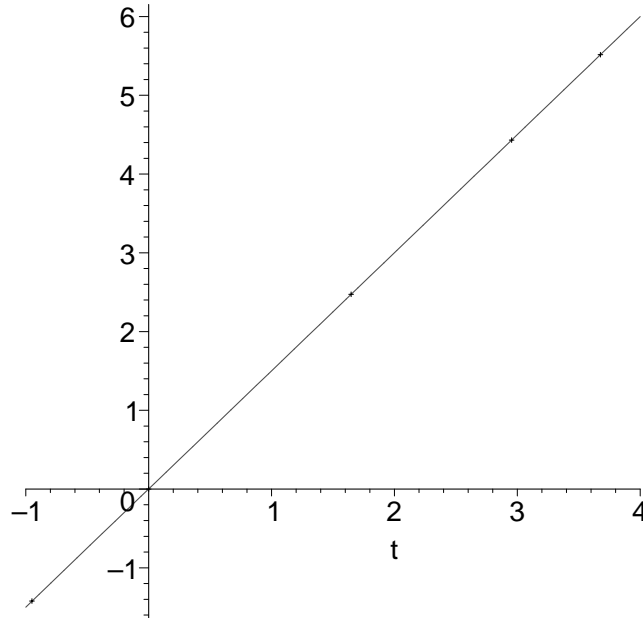
Actually, least squares for linear systems is a standard tool so Maple has the command built right in. You can read about it by searching the topic "leastsqrs".

The first entry above is the least-squares line slope (for the ln-ln data), the second point its intercept. See visually how the line-fit went:

```
> lnlnplot:=pointplot({seq([A3[i,1],A3[i,2]],i=1..rowdim(A3))}):
#those are the points from the ln-ln data in
#the A3 matrix. the index i ranges over all rows
#in A3.

> line:=plot(1.499816412*t + .4868908465e-3, t=-1..4):
#I used the mouse to paste in the coefficients
#from my work above. Here t is standing for the
#ln(x) variable.

> display({lnlnplot,line});
```

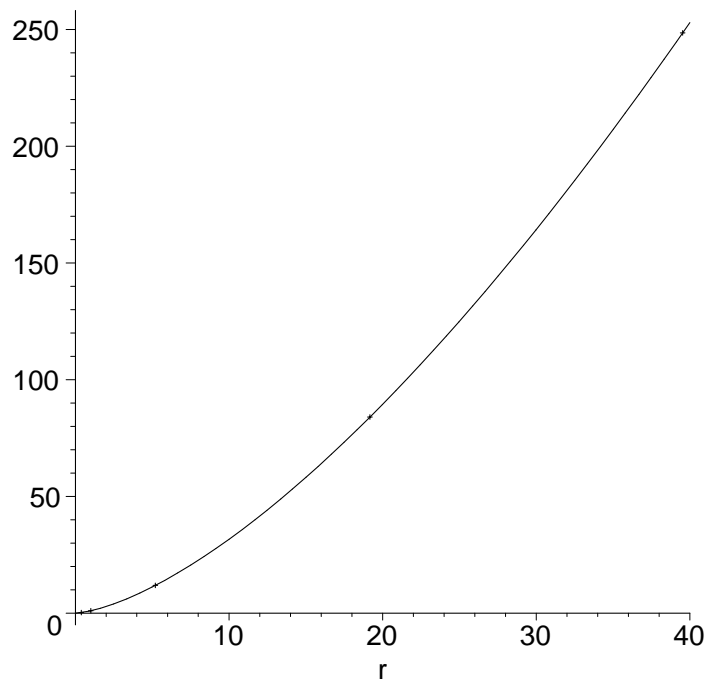


The line fit is pretty close! (Kepler didn't actually get to use Pluto in his data set, since it wasn't discovered until the 1900's.) The slope of this line will be our experimental power, its intercept will be the \ln of our proportionality constant. Now work backwards to get these values:

```
> C:=exp(.486890e-3); #proportionality constant
p:=1.499816413;
#power. I used the mouse to paste these in.
C:=1.000487009
p:=1.499816413
```

Notice, the power came out close to 1.5 and the proportionality constant came out close to 1. Now see how our power law works for the original (xi,yi) data:

```
> realplot:=pointplot({seq([A1[i,1],A1[i,2]],i=1..rowdim(A3))}):
> powerplot:=plot(C*r^p,r=0..40,color=black):
> display({realplot,powerplot});
```



Your job in the first part of this Maple project will be to carry out the same sort of analysis for the height-weight data which we have accumulated. If you go to our Maple page you may download the template for this part of your project.