Installing gnuplot and Using gnuplot

You may have gnuplot already installed on your machine. You can test this the same way we tested for g++:

```
which gnuplot
```

If it returns a path, you have gnuplot installed! If not, use your favorite package manager to install it. I'm an Ubuntu user, so I had to run:

```
sudo apt-get install gnuplot
```

If you're on a different distribution, you'll probably need to use `yum`, or some GUI tool. On Mac OS X, an optional package manager is Brew: http://brew.sh/, which will help you out.

By looking around on stackoverflow, I found a sample brew install command:

```
brew install gnuplot --wx --cairo --pdf --with-x --tutorial
```

Which will let you output PDFs as well as to the screen (that's the whole `with-x` and `wx`), I imagine. If you get stuck, let us know!

To test out gnuplot in OS X or Linux, run:

```
gnuplot
```

from the terminal. This will put you in an interactive gnuplot terminal.

Plotting Fitting and Saving the Plot

The basic problem is to plot and fit data and save the figure. Here is a simple example So now you can have fun fitting and saving a figure, using the sorted data for the find exercise

```
plot "Sorted100.txt"        #plot out in column format
f(x) = a + b*x + c*x*x      #parameterize fitting function.
fit f(x) "Sorted100.txt" via a,b,c
set term postscript color   #one option that gives a .ps figure
set output "SillyFit.eps"   #one option that gives a .ps figure
replot                      #one option that gives a .ps figure
set term x11                #return to interactive view.
                            #On linux, you may need ‘‘wxt’’ instead of x11
```

The fit was

```
Final set of parameters            Asymptotic Standard Error
=======================            ==========================
```

```
a               = -27217.6          +/- 5164         (18.97%)
b               = 11715             +/- 238.7        (2.037%)
c               = -19.3785          +/- 2.309        (11.92%)
```

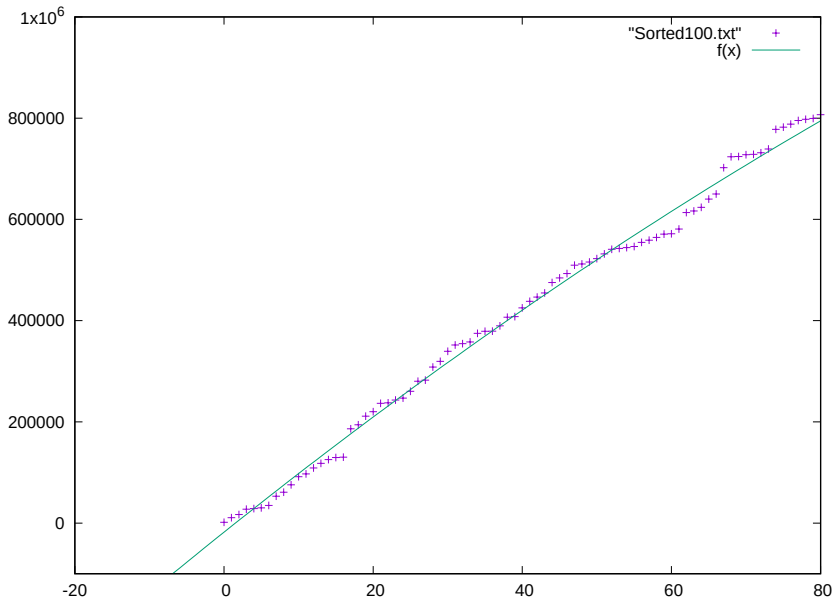

Figure 1: The plot looks pretty nice.

Fitting with Error of Climate data

This compares a polynomial fit vs a Fourier fit. **This is the script `plotSave` on GitHub in `GlobalWarmingExample`**

```
f8(x)=a0 +a1*cos(x*w)+b1*sin(x*w) +a2*cos(2*x*w)+b2*sin(2*x*w) +a3*cos(3*x*w)
+b3*sin(3*x*w)+ a4*cos(4*x*w)+b4*sin(4*x*w) +a5*cos(5*x*w)+b5*sin(5*x*w)
+a6*cos(6*x*w) +b6*sin(6*x*w)+a7*cos(7*x*w)+b7*sin(7*x*w)+a8*cos(8*x*w)+b8*sin(8*x*w)

w =0.01

fit f8(x) 'Complete_TAVG_summary.txt' using 1:4 via w, a0,a1,b1 ,a2,b2 ,a3,b3,a4,
b4 ,a5,b5,a6,b6,a7,b7,a8,b8

plot  'Complete_TAVG_summary.txt' using 1:4:5 w errorbars
replot f8(x)
```

```
f11(x) = c0 + c1*x + c2*x**2 + c3*x**3+ c4*x**4 + c5*x**5
#+ c6*x**6 + c7*x**7 + c8*x**8 + c9*x**9 + c10*x**10 + c11*x**11

fit f11(x) "Complete_TAVG_summary.txt" u 1:4 via  c0,c1,c2,c3,c4, c5
#,c6,c7,c8,c9, c10,c11

replot f11(x)

set output "ThisPlot.eps"
set terminal postscript color

replot

set term x11
```

A Few Random Tricks!

You can do almost anything using gnuplot. Just google to find more than you want or need

```
# Hashes aren't for twitter, they're for comments in gnuplot!
plot sin(x) # plot the sine function
f(x) = cos(x) # assign a function
plot sin(x), f(x) # plot two functions at once.
set xrange [0:2] # change the x axis.
set yrange [-2:2] # change the y axis range.
replot # update the plot with your new axis.
set yrange [-5:-2] # change the y axis range again.
replot # you won't see anything! So do...
reset # ... because you've messed up!
set xrange [-1:1]
plot x*sin(1/x) # This will look really bad!
set samples 1000 # sample the function more frequently.
replot # it should look a lot better now
help    #get syntax inside gnuplot
exit # and we're done!
```