9/12/00

Intermediate Physics Lab, Physics 440
Data Plotting and Fitting Exercise

For many research projects, you will have to plot data on various types of graph, and find and plot theoretical models to compare with that data. Powerful computer packages such as Mathematica, Matlab, and Origin are now available and commonly used in research labs for these purposes. The purpose of this exercise is to get you to learn the rudiments of one of these packages, if you don't already know them.

The assignment is divided into stages of increasing difficulty and sophistication. **We want you to use a full-featured *scientific* data-handling package: Origin or Mathematica.** You may already know how to do some parts of the assignment using software like Excel, Deltagraph, Sigmaplot… We want you to use Origin or Mathematica so you can start to learn about one of these packages. All stages use the "dummy data" that is listed at the end of this document. You can type this data into Origin or Mathematica or you can import it without typing in various ways.

What to hand in. We don't need a lot of writing, but we *do* need at least the following:

Your name and the date
Actual plots, well-labeled to show what is what
Your answers to the questions below

Stage one: Simple plot and straight-line fit. Plot the z data set versus the x data set on linear axes, using points or other symbols (*not* a curve through the data points). Be sure the axes are labeled.

Then, determine the best straight-line fit to the data and produce a plot showing both the data and the fit.

Stage two: Semi-log and log-log plots and what they tell us. Plot the u data set versus the x data set. Make one, the other, or both axes logarithmic- do any of these turn the plot into a straight line? If so, what does that tell us about the data (what kind of function? what power?). Repeat with the v data set plotted versus x.

Stage three: Plot data with error bars. Plot the y data set versus the x data set, using the e data set as uncertainties on y. That is, each data point should show a symbol at (x_i, y_i) along with an error bar extending by plus and minus e_i in the y direction.

Stage four: Linear least-squares fit. Find the best fit of a quadratic function to the y versus x data. That is, find the values of a, b, c such that $y=ax^2+bx+c$ best describes the data. This will normally involve minimizing the sum of the squares of the differences between the fit and the data, hence the name "least squares". Plot the fit and the data (with error bars) on the same graph, and print out the values of a, b, c. Usually it works best to plot the data using points or other symbols, and the fit as a smooth curve.

Stage five: Nonlinear least-squares fits. Find and plot the best fit of the y versus x data to a gaussian function:

$$y = a \exp(-(x-c)^2 / 2\sigma^2)$$

with a, c, and σ to be determined. Do the same thing for a cosine function:

$$y = a \cos(\omega(x-c)).$$

Note: This stage is "harder" (requires a more sophisticated fitting program) than stage four, because the gaussian and cosine functions are not simply linear combinations of given functions.

Here are the data sets:

x	z	u	v	y	e
0.015	0.921	0.212	1.381	0.015	0.07
0.172	1.472	0.457	1.328	0.21	0.109
0.329	1.527	0.598	1.731	0.44	0.187
0.486	1.921	0.703	1.987	0.537	0.14
0.643	2.792	0.815	1.964	0.764	0.235
0.8	3.181	0.909	2.644	0.753	0.105
0.958	3.869	0.998	2.864	0.96	0.212
1.115	3.955	1.06	3.262	0.959	0.131
1.272	4.081	1.139	4.042	0.974	0.088
1.429	5.127	1.21	4.448	1.019	0.099
1.586	5.133	1.262	5.119	1.198	0.268
1.743	5.906	1.323	6.138	1.009	0.094
1.9	5.709	1.391	6.914	0.948	0.072
2.057	6.447	1.442	8.315	0.99	0.176
2.214	7.23	1.504	9.524	0.92	0.19
2.371	7.951	1.542	10.809	0.73	0.103
2.528	8.07	1.603	12.953	0.666	0.16
2.685	8.8	1.642	14.915	0.452	0.081
2.843	8.986	1.698	17.173	0.451	0.227
3	9.743	1.743	20.364	0.245	0.174