Data Analysis

Problems for the lecture at the Bad Honnef Summer School on "Efficient Algorithms in Computational Physics", September 10-14, 2012.

Which software to use?

To read in data from files, select the lines of interest, and split the lines up into different fields, perl and python are more flexible than C, so you should be familiar with one of them.

It is necessary to plot results, and two common ways of doing them are the numplotlib package of python, and gnuplot. You should therefore be familiar with one of these, or another graphics package.

To do simple, straight-line fits it is easy to write your own code in C, perl, or python (or fortran 90). To do more complicated fits it is useful to use a packaged routine, such as provided in the scipy package of python, or in gnuplot. Alternatively, you could be brave and use the Numerical Recipes routines.

Altogether, I recommend that you are familiar with either

- python, including the scipy and numplotlib packages, or
- perl and gnuplot.

Sample solutions will be made available at the end of the course.

1. Firstly, just a toy problem to be done with pencil and paper, to make sure that you have the square roots and the factor of "-1" correct. I have 9 data points, assumed to be statistically independent. These have values $-2, -1, -1, 0, 0, 0, 1, 1, 2$. Estimate the mean and the error in the mean.
   Ans: Mean is 0, error bar is $1/\sqrt{6}$.

2. Consider the 1000 data points at
   `http://physics.ucsc.edu/~peter/bad-honnef/data.HW2`. I would like information on the *shape* of the distribution, as characterized by the kurtosis $\langle x^4\rangle/\langle x^2\rangle^2$. Using jackknife methods discussed in the handout, estimate the kurtosis including error bar.
   From this result, guess what is the shape of the distribution.

3. Consider the data in `http://physics.ucsc.edu/~peter/bad-honnef/data.HW3`. The first column is x, the second is y and the third is the error in y. There are 200 points.

   (a) Perform a least squares straight line fit, $y = a_0 + a_1 x$ using your own code (I suggest in C, perl or python) to implement the equations for a straight-line fit in the handout. You should determine the parameters a_0 and a_1, as well as the error bars in these quantities and the value of χ^2 per degree of freedom.

   (b) Compare your results for part 3a with what you find from a packaged fitting routine, for example from the scipy package of python or from gnuplot. (If you use gnuplot, note the subtlety of extracting the error bars correctly discussed in Appendix D of the handout.)

4. In some recent work we obtained results for a characteristic temperature $T_L^\star$ for different system sizes L. It is expected that $T_L^\star$ varies as

$$T_L^\star = T_c + \frac{A}{L^\omega} \tag{1}$$

where T_c is the transition temperature and ω is a "correction to scaling" exponent. The data is given in `http://physics.ucsc.edu/~peter/bad-honnef/data.HW4`.

(a) Estimate the values of T_c, A and ω, along with error bars in these quantities, the value of χ^2 per degree of freedom, and the goodness of fit parameter Q. Equation (1) is a non-linear model, and I suggest you use a packaged routine either in the scipy package of python, or in gnuplot.

(b) You will see that the parameters T_c and ω are not well determined and the error bars in these quantities, obtained from the *local* curvature of χ^2 at the minimum, are not sensible. Obtain a sensible confidence interval for T_c by determining χ^2 as a function of T_c (allowing the other two parameters to vary to minimize χ^2), and seeing where $\Delta\chi^2 \leq 1$. This procedure is discussed in the handout. A plot of χ^2 against T_c would be nice.