Introduction to Data Fitting Ideas

David Litster — Department of Physics

MIT

September 12, 2011

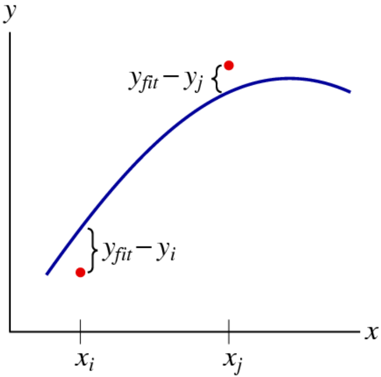
Why Fit Data?

Often when you have some measured data points $y_i(x_i)$ corresponding to independent variables x_i you would like to find a mathematical function $y_{\text{fit}}(x)$ that represents them. If there is a theoretical model that you think may explain your data then it will provide such a mathematical function.

- The model will have some parameters whose numerical values are unknown and can be adjusted.
- If the data are represented well by the model, then the values of the parameters in the model can be found from the fit.
- If the fit is good and the parameter values are physically reasonable, you might decide the model is correct.
- How do you decide if the fit is good?

Goodness of Fit

The goodness of fit is a measure of the distance between the data points and the function that is supposed to fit them.



$$TSE = \sum_{i=1}^{N} [\, y_{\text{fit}}(x_i) - y(x_i)]^2$$

where *TSE* is the total squared error between the data points and the equation that is supposed to fit them.

If the fit is to a straight line, $y(x_i) = A + Bx_i$, the parameters A and B are chosen to minimize a properly normalized *TSE*.

Goodness of Fit χ^2

Not all data points may have the same error (uncertainty) and one might expect more adjustable parameters would give a better fit. A quantity χ^2 is defined to take this into account. In normalized form [Bevington calls this χ^2_ν] this is

$$\chi^2_v = \frac{1}{(N - N_{\text{params}})} \sum_{i=1}^{N} \frac{[\,y_{\text{fit}}(x_i) - y(x_i)]^2}{\sigma_i^2}$$

where σ_i^2 is the variance (expected squared error) of the data point $y(x_i)$, N is the number of data points in the fit, and N_{params} is the number of adjustable parameters in the fit function.

Goodness of Fit and χ^2

The normalizing factor$(N - N_{\text{params}})$ is commonly used because one expects that if the number of adjustable parameters equals the number of data points a perfect fit should be possible.

Thus χ^2_ν is the ratio of the actual *TSE* obtained in the fit to the expected *TSE*. If the fit is good, we expect $\chi^2_\nu \simeq 1$.

Of course σ_i^2 is just the variance $\langle (x_i - \langle x_i \rangle)^2 \rangle$ for data point $y(x_i)$; that means if the measurement of the N data points is repeated a somewhat different χ^2_ν would be obtained each time even if the values σ_i^2 were known precisely—which they are usually not. To understand what χ^2_ν tells us about the fit, we need to know more about how it might change for different measurements of the same data points. Bevington, Chapter 10 has more detail, and in an appendix there is a table of the cumulative probability $P_\chi(\chi^2,\, \nu)$ for χ^2_ν.

Goodness of Fit and χ^2 (cont'd)

The value of $P_\chi(\chi^2, \nu)$ depends somewhat on ν, but for a few parameters, here are some useful rules of thumb.

- A good fit should have $\chi^2_\nu \simeq 1$; if you find that χ^2_ν is significantly less than 1, you have probably overestimated the uncertainties σ_j.
- The chance that $\chi^2_\nu > 2$ is 0.1 or less, so it is generally accepted that a fit to one function is statistically better than a fit to another function whose χ^2_ν is twice as large.
- The standard deviation of a fitting parameter A, σ_{A}, for example, is the change in parameter A (while optimizing all other parameters) that increases the *TSE* by about $N\langle\sigma_j\rangle$ or increases χ^2_ν by about $1/N$ from the optimum value. Most fitting algorithms compute this and Bevington explains how to calculate it for various fits in Chapters 8, 9, and 11.

Some Practical Considerations

- The fitting is often easier if all σ_i are the same, and hence equal to a quantity σ.
- The fitting is often easier if the data points are equally spaced in the independent variable x.
- Fitting to a polynomial [Bevington Chapter 8] and functions which depend linearly on the fitting parameters [Bevington Chapter 9] are relatively easy.
- Non-linear least squares fitting (i.e., to an arbitrary function) will usually be most useful to us. It is discussed in Bevington Chapter 11 and the commonly used Lev-Marquardt algorithm is available in our *Matlab* package.

Estimating The Errors

It is clear that knowing the values σ_i^2 for your measurements will be important for quantitative analysis of measurements that you make. Uncertainty in your measurements can result in two main ways.

1. Systematic errors (e.g., miscalibrated meter); these are usually the same for all measurements and are often difficult to recognize or find.
2. Random errors (e.g., Poisson counting statistics). These can usually be reduced by longer measurement times and are easier to quantify.

Independent measurements are those that do not depend on previous measurements or history. Repeating independent measurements can often be used to estimate uncertainties; it will not help with systematic errors.

Repeating a Measurement

If we make N independent measurements of a quantity y, the mean will be

$$\langle y \rangle = \frac{1}{N} \sum_{i=1}^{N} y_i$$

and the variance will be

$$\sigma_y^2 = \langle (y - \langle y \rangle)^2 \rangle = \frac{1}{N-1} \sum_{i=1}^{N} (y_i - \langle y \rangle)^2$$

The $(N-1)$ is because one measurement was needed to find $\langle y \rangle$ and so only $N-1$ independent measurements were made.

Repeating a Measurement (cont'd)

[Bevington, Chapter 2]

The uncertainty in the mean is:

$$\langle y \rangle = \frac{1}{N} \sum_{i=1}^{N} y_i \pm \frac{\sqrt{\sigma_y^2}}{\sqrt{N}} \text{ (improves with } N\text{)}$$

If uncertainty σ_i was different for each measurement, this has a more complicated form

$$\langle y \rangle = \frac{\sum_{i=1}^{N} y_i/\sigma_i^2}{\sum_{i=1}^{N} 1/\sigma_i^2} \pm \sqrt{\frac{1}{\sum_{i=1}^{N} 1/\sigma_i^2}}$$

Adding Errors

The total uncertainty in a measurement can be the cumulative result of several sources.

For statistically independent errors, add the variances:

$$\sigma_{\text{total}}^2 = \sigma_1^2 + \sigma_2^2 + \sigma_3^2 + \cdots$$

For correlated errors, add the standard deviations:

$$\sigma_{\text{total}}^2 = (\sigma_1 + \sigma_2 + \sigma_3 + \cdots)^2$$

Propagating Errors

[Bevington, Chapter 4]

This arises when you want to know the uncertainty in a quantity that is a function of other quantities that also are uncertain. You should read this chapter, but it is mostly common sense combined with what you know from calculus (Taylor's series and partial derivatives). It matters if the uncertainties in the quantities that contribute are correlated (the previous slide being an extreme example).

Suppose variables $u \pm \sigma_u$ and $v \pm \sigma_v$ and we have a quantity $x = f(u, v)$. Then:

$$\sigma_x^2 = \sigma_u^2 \left(\frac{\partial x}{\partial u} \right)^2 + \sigma_v^2 \left(\frac{\partial x}{\partial v} \right)^2 + 2\sigma_{uv}^2 \left(\frac{\partial x}{\partial u} \right) \left(\frac{\partial x}{\partial v} \right) + \cdots$$

where

$$\sigma_{uv}^2 = \langle (u - \langle u \rangle)(c - \langle v \rangle) \rangle$$

Propagating Errors (examples)

$$\mathbf{x = u \pm v} \qquad \sigma_x^2 = \sigma_u^2 + \sigma_v^2 + 2\sigma_{uv}^2$$

$$\mathbf{x = uv} \text{ or } \mathbf{u/v} \qquad \frac{\sigma_x^2}{x^2} = \frac{\sigma_u^2}{u^2} + \frac{\sigma_v^2}{v^2} + 2\frac{\sigma_{uv}^2}{uv}$$

$$\mathbf{x = a\,u^p} \qquad \frac{\partial x}{\partial u} = p\,\frac{x}{u}\text{, so } \frac{\sigma_x}{x} = p\,\frac{\sigma_u}{u}$$

Sometimes the function you want is a function of the parameters obtained from fitting data and the algorithm can give you the correlation or covariance you need; see Bevington, Chapter 9.
Note: if the function you are trying to fit has parameters that are significantly correlated it can make the fit very slow to converge and the parameter uncertainties very large.

Fitting Example: Human Vision

In 1942, S. Hecht, S. Shlaer, and M. Pirenne (HSP) published an article [1] in which they investigated the detectability of light at the threshold of human vision. This classic paper in biophysics is discussed in detail in the excellent textbook *Physics With Illustrative Examples From Medicine and Biology: STATISTICAL PHYSICS* by George B. Benedek and Felix M. H. Villars [Springer-Verlag, 2000]. This discussion is taken from that book.

HSP found that the eye integrates for about 0.1 s and the response is determined by the number of photons that arrive in that time. They applied 1 ms flashes of light to a dark-adapted carrot-fed eye so that the light would fall on the most sensitive part of the retina, and measured the probability the light would be detected as a function of the number of photons incident on the cornea.

[1] "Energy Quanta and Vision", S. Hecht, S. Shlaer, and M. Pirenne, *Journal of General Physiology* **25**, 819 (1942)

Fitting Example: Human Vision (cont'd)

HSP measured the probability light would be seen as a function of n, the number of photons incident on the cornea. Their data are given in the table at the right.

n	$p(n)$
37	0.00
59	0.12
93	0.44
149	0.94
239	1.00

Their model had two assumptions; first that only $c\,n$ of the n photons will reach the retina to excite a conformational change in rhodopsin.

Second, it assumed that the number of rhodopsin molecules excited for a given number of photons obeyed Poisson statistics and that the light would be seen if at least m_0 molecules were excited.

Fitting Example: the Function

The model has two adjustable parameters: the attenuation c and the threshold for detection m_0.

$$p(n) = \sum_{m=m_0}^{\infty} \frac{(cn)^m}{m!}\, e^{-cn} = 1 - \sum_{m=0}^{m_0-1} \frac{(cn)^m}{m!}\, e^{-cn}\,.$$

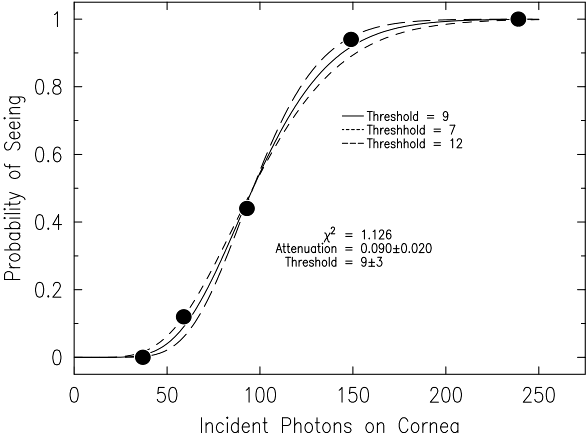
The fitting is slightly tricky, as m_0 must be an integer.

I wrote a program which found the c to minimize χ^2_ν for given values of m_0. Fixing $\sigma = 0.02$ for each $p(n)$, the program gave the results at right.

m_0	c	χ^2_ν
7	0.0694 ± 0.0016	2.17
9	0.0901 ± 0.0014	1.13
12	0.1212 ± 0.0026	2.35

Fitting Example: Results

The model seems good, and it is interesting to learn that the human eye can detect as few as 10 photons.



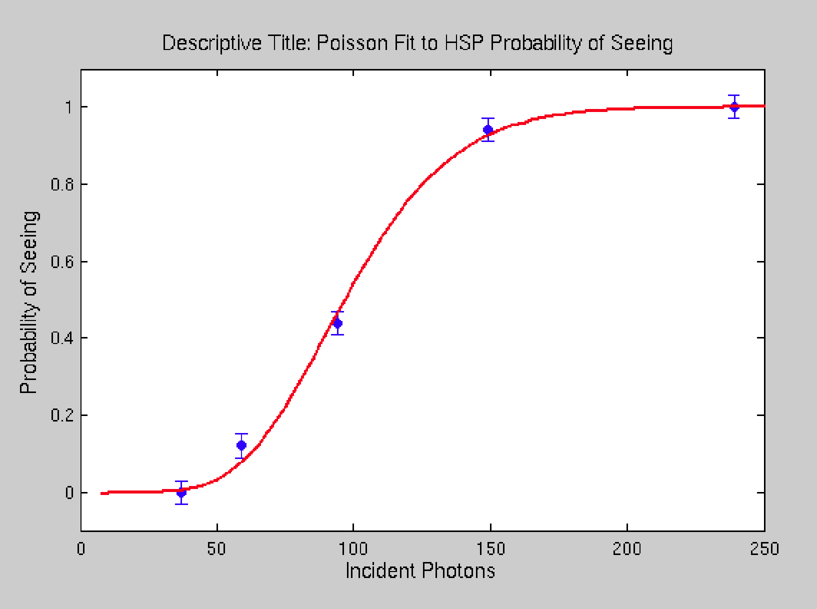
Fitting Example: Matlab Results

I tried the same fit using *matlab* and got essentially the same results. They differed slightly because I used $\sigma = 0.03$ instead of 0.02. My *matlab* script `fithsp.m` gave a reduced χ^2_ν of 0.99 and the parameters $c = 0.1000 \pm 0.0005$ and $m_0 = 10.0 \pm 1.8$ (in the *matlab* fit, m_0 was not restricted to be an integer).

If you want to try my calculation using *matlab*, you can get a copy of the data and the scripts I used from the HANDOUTS link on the 8.13 web page; get the file `HSP.zip`.

The *matlab* plot of the fit is on the next slide.

Fitting Example: Plot of Matlab Results


Descriptive Title: Poisson Fit to HSP Probability of Seeing
Probability of Seeing
Incident Photons