

# Chi-Square Minimization

John W. Fowler

12 March 2008

Given  $N$  data points  $y_i$  with uncertainties  $\sigma_i$ ,  $i = 1$  to  $N$ , and each data point associated with an abscissa  $x_i$ , we wish to model  $y$  as a function of  $x$  with a functional dependence of the form

$$y = \sum_{k=1}^M p_k f_k(x) \quad (1)$$

where the  $p_k$  are coefficients whose values must be found, and the  $f_k$  are  $M$  arbitrarily chosen *basis functions* that define the model. For example, if we wish to use a cubic polynomial model, then  $f_k(x) = x^{k-1}$ , and  $M=4$ . In order to solve for  $M$  model coefficients, we must have  $N \geq M$ , and for chi-square to be meaningful, we must have  $N > M$ . In typical cases,  $N \gg M$ . We have deliberately made the functional dependence linear with respect to the  $p_k$  coefficients. The basis functions may be anything at all as long as they can be evaluated at the abscissa values (hence they may not be explicit functions of  $p_k$ ); they may even be functions of both  $x$  and  $y$ , since the  $y_i$  are also known, but since the  $y_i$  are uncertain, any basis function that depends on them will also be uncertain, and in the chi-square minimization below, this will cause the  $p_k$  to appear in the denominators, making the system of equations that we have to solve nonlinear. Nonlinear systems can be solved via iterative methods, but the subject is far removed from the present scope, and so we will use only basis functions that do not depend on  $y$ . We also assume that the abscissa values are known exactly, since if they also have uncertainties, that too results in the  $p_k$  appearing in the denominators of the equations we have to solve, which would therefore again be nonlinear.

If the uncertainties in the data describe errors that may be approximated as zero-mean Gaussian random variables, then Gaussian estimation may be used to evaluate the coefficients  $p_k$ , and from this it follows that chi-square minimization is the optimal method to employ. For uncorrelated errors, chi-square is the sum of the ratios of the squared differences between the model and the observed values divided by the uncertainty variance:

$$\chi^2 = \sum_{i=1}^N \frac{\left( y_i - \sum_{k=1}^M p_k f_k(x_i) \right)^2}{\sigma_i^2} \quad (2)$$

The number of independent terms summed minus the number of parameters used in the fit is called the number of degrees of freedom of the  $\chi^2$  random variable,  $D_F$ , in this case  $N-M$ . Since  $\chi^2$  is a function of the random errors in the  $y_i$ , it is indeed a random variable itself, with its own probability density function and cumulative distribution. The exact forms of these two functions depend on  $D_F$ , but two useful features that all  $\chi^2$  random variables have in common is that the mean is  $D_F$  and the variance is  $2D_F$ . For the above expression to be a rigorously defined  $\chi^2$  random variable, the errors in the  $y_i$  must be truly Gaussian. Fortunately, in the vast majority of practical applications, treating errors as Gaussian even when that is not quite true is nevertheless an acceptable

approximation, especially when it enters only as a cost function to be minimized in some fitting process. One must use more care when interpreting the significance of an observed fluctuation (i.e., a sample of the random variable). Furthermore, biased estimates generally result when deviations from the Gaussian distribution include being asymmetric, so the quality of the approximation must be understood. For example, biases at the 0.1% level may or may not be acceptable.

The expression above assumes that the errors in the  $y_i$  are uncorrelated, i.e., that the expectation value of the product of the error in  $y_m$  and the error in  $y_n$ ,  $m \neq n$ , is zero. In fact, it often occurs that errors *are* correlated, and taking this into account involves using a more general form for  $\chi^2$ , but this slight complication will be postponed for the moment, and we will assume that the errors are uncorrelated. Ignoring significant correlations in fitting computations generally has a very small effect on the  $p_k$  solutions; the main impact is on the characterization of the fitting uncertainty, where it typically cannot be neglected. Inclusion of explicitly correlated data errors will be formulated further below.

In order to solve for the values of  $p_k$  that minimize  $\chi^2$ , we set the derivatives of the latter with respect to the former to zero:

$$\frac{\partial \chi^2}{\partial p_k} = -2 \sum_{i=1}^N \frac{\left( y_i - \sum_{j=1}^M p_j f_j(x_i) \right) f_k(x_i)}{\sigma_i^2} = 0 \quad (3)$$

This results in  $M$  equations in the  $M$  unknown coefficients  $p_k$ , forming a linear system that can be solved by standard methods. Each equation has the form

$$\begin{aligned} \sum_{i=1}^N \frac{y_i f_k(x_i)}{\sigma_i^2} &= \sum_{i=1}^N \frac{f_k(x_i) \sum_{j=1}^M p_j f_j(x_i)}{\sigma_i^2} \\ &= \sum_{j=1}^M p_j \sum_{i=1}^N \frac{f_k(x_i) f_j(x_i)}{\sigma_i^2} \end{aligned} \quad (4)$$

where we have interchanged the summation order of the right-hand side on the second line. With the definitions

$$\begin{aligned} b_k &\equiv \sum_{i=1}^N \frac{y_i f_k(x_i)}{\sigma_i^2} \\ a_{jk} &\equiv \sum_{i=1}^N \frac{f_j(x_i) f_k(x_i)}{\sigma_i^2} \end{aligned} \quad (5)$$

we obtain

$$b_k = \sum_{j=1}^M a_{jk} p_j \quad (6)$$

which can be written in vector-matrix form as

$$\vec{b} = A \vec{p} \quad (7)$$

where the matrix  $A$  has elements  $a_{jk}$  and is called the coefficient matrix, because it contains the coefficients of the equations forming the  $M \times M$  system of equations. Note that  $a_{jk} = a_{kj}$ , i.e.,  $A$  is a symmetric matrix. Equation 7 can be solved by various techniques, one of which is to employ the inverse of the  $A$  matrix: multiplying both sides from the left by  $A^{-1}$ , and dropping the identity matrix that results from  $A^{-1}A$ , yields the equation

$$\vec{p} = A^{-1} \vec{b} \quad (8)$$

If solutions for the  $p_k$  are all that is desired, then this is not the most computationally efficient way to proceed, but in fact, we will need the inverse matrix anyway, and so we might as well compute it first and use it to obtain the solution. The reason why we need it is that we need the error covariance matrix for the  $p_k$ , and this is just  $A^{-1}$  (a derivation will be given below). Fitting errors are typically significantly correlated, so when the model is used to compute a value of  $y$  for some chosen value of  $x$ , the uncertainty in that value of  $y$  should be computed using the full error covariance matrix:

$$\Omega = A^{-1} \equiv \begin{pmatrix} v_{11} & v_{12} & \cdots & v_{1M} \\ v_{21} & v_{22} & \cdots & v_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ v_{M1} & v_{M2} & \cdots & v_{MM} \end{pmatrix} \quad (9)$$

The diagonal elements  $v_{kk}$  are the uncertainty variances of the  $p_k$  solutions, also commonly denoted  $\sigma^2(p_k)$ . These elements are always positive in practical applications, i.e., uncertainties are always real and greater than zero in real life. Because  $A$  is symmetric, its inverse  $\Omega$  is also. The off-diagonal elements are the error covariances for the model coefficients, i.e. expectation values for the product of the errors in  $p_k$  and  $p_j$ ,  $k \neq j$ , and may be positive, negative, or zero. Using a circumflex to indicate true values, and denoting errors by  $\varepsilon$ , we can write the model equation as

$$\hat{y} + \varepsilon_y = \sum_{k=1}^M (\hat{p}_k + \varepsilon_{pk}) f_k(x) \quad (10)$$

Since

$$\hat{y} = \sum_{k=1}^M \hat{p}_k f_k(x) \quad (11)$$

we can subtract this from the previous equation to obtain

$$\varepsilon_y = \sum_{k=1}^M \varepsilon_{pk} f_k(x) \quad (12)$$

The uncertainty variance for  $y$  is the expectation value of the square of  $\varepsilon_y$ , where angle brackets denote expectation values:

$$\begin{aligned} \sigma_y^2 = \langle \varepsilon_y^2 \rangle &= \left\langle \left( \sum_{k=1}^M \varepsilon_{pk} f_k(x) \right)^2 \right\rangle \\ &= \sum_{k=1}^M \langle \varepsilon_{pk}^2 \rangle f_k^2(x) + 2 \sum_{k=1}^{M-1} \sum_{j=k+1}^M \langle \varepsilon_{pj} \varepsilon_{pk} \rangle f_j(x) f_k(x) \\ &= \sum_{k=1}^M \sigma^2(p_k) f_k^2(x) + 2 \sum_{k=1}^{M-1} \sum_{j=k+1}^M v_{jk} f_j(x) f_k(x) \end{aligned} \quad (13)$$

Once the  $p_k$  are known, equation 2 should be evaluated to verify that the value of  $\chi^2$  is reasonable, i.e., close to  $D_F$ . “Close” generally means within several standard deviations, where the value of the standard deviation is  $\sqrt{(2D_F)}$ . If this is not true, then the model is not trustworthy. Assuming error correlation was not inadvertently ignored, if  $\chi^2$  is too small, then the original data uncertainties may have been overestimated; if  $\chi^2$  is too large, then either the original data uncertainties were underestimated or the model formula is not a good representation of the data, or both.

It is important to appreciate that although the fitting uncertainties  $\sigma^2(p_k)$  may be small, this does not mean that the error in a value of  $y$  computed from the model will be small. The fitting uncertainties reflect only whether the resulting fit is the best possible fit to the given model, not whether that model itself is a good representation of the data. The fitting error covariance matrix  $\Omega$ , being the inverse of  $A$ , depends on the same parameters as  $A$ , and this dependence does not include any of the  $y_i$ .  $A$  and  $\Omega$  depend only on the  $a_{jk}$ , and as shown in equation 5, the  $a_{jk}$  depend only on the data uncertainties  $\sigma_i$  and the  $x_i$ . The data uncertainties produce some uncertainty in what the best model parameter values are to fit to the data, but the best fit may be simply the best of a set of very bad fits because the form of the model is poorly chosen. We have said that each  $y_i$  is associated with an  $x_i$ , and so these pairs can be plotted on a graph; we would normally choose a model that when plotted on the same graph would pass closely to the data points, and if there is structure in the pattern of the data points, we would want the model to show similar structure.

But imagine shuffling the relationship between the  $x_i$  and  $y_i$  to form very different pairs; these would form a very different pattern on the graph, and the fitting procedure would yield a very different set of solutions for the  $p_k$ , but the uncertainties of this new set of  $p_k$  would be exactly the same as before, because these uncertainties do not depend on the pairing between the  $x_i$  and  $y_i$ , simply because they do not depend on the  $y_i$  at all. The uncertainty in whether we have obtained the best possible fit to the shuffled data pairs given the form of the model is no greater than before, even though the fit itself yields a model that probably looks nothing like the new data points on the graph.

Clearly some check on whether the form of the model yields a good fit, not merely a fit with well determined parameters, is needed. This check is provided by the test on the value of  $\chi^2$  described above. The value of  $\chi^2$  will depend on whether we shuffled the data pairs, and it *does* depend on whether the model's shape follows the patterns of structure in the data.

If the original data uncertainties were well estimated, and if the form of the model is appropriate for the actual dependence of  $y$  on  $x$ , then the fluctuations of the  $y_i$  relative to the model should have statistical similarity to the original data uncertainties. The  $\chi^2$  test measures whether this is the case and should always be performed. If the value of  $\chi^2$  turns out to be large relative to  $D_F$ , then the uncertainties attached to the model should be inflated, or a better model should be used unless it can be shown that the original uncertainties were underestimated. One way to inflate the model uncertainty is simply to rescale  $\Omega$  by  $\chi^2/D_F$ . If nothing more than this can be done, then at least this much should be done, but unless resources simply do not permit, the reason for the failure of the model should be determined and fixed.

The convenient fact that the error covariance matrix  $\Omega$  is the inverse of the coefficient matrix  $A$  can be derived as follows. We write equation 8 in summation form, using the elements of  $\Omega$  explicitly, changing the dummy summation index from  $k$  to  $l$  and substituting the definition of  $b_l$ :

$$p_k = \sum_{l=1}^M v_{lk} b_l = \sum_{l=1}^M v_{lk} \sum_{i=1}^N \frac{y_i f_l(x_i)}{\sigma_i^2} \quad (14)$$

We can write this in the “true plus error” form, along with the “true” itself:

$$\begin{aligned} \hat{p}_k + \varepsilon_{pk} &= \sum_{l=1}^M v_{lk} \sum_{i=1}^N \frac{(\hat{y}_i + \varepsilon_i) f_l(x_i)}{\sigma_i^2} \\ \hat{p}_k &= \sum_{l=1}^M v_{lk} \sum_{i=1}^N \frac{\hat{y}_i f_l(x_i)}{\sigma_i^2} \end{aligned} \quad (15)$$

Subtracting the second line from the first gives

$$\varepsilon_{pk} = \sum_{l=1}^M v_{lk} \sum_{i=1}^N \frac{\varepsilon_i f_l(x_i)}{\sigma_i^2} \quad (16)$$

The elements of the error covariance matrix are  $\langle \varepsilon_{pj} \varepsilon_{pk} \rangle$ , so we write the product of the above equation with a similar one that uses indexes  $m, j$ , and  $n$  in place of  $l, k$ , and  $i$ , respectively, and take expectation values:

$$\begin{aligned} \langle \varepsilon_{pj} \varepsilon_{pk} \rangle &= \left\langle \left( \sum_{m=1}^M v_{mj} \sum_{n=1}^N \frac{\varepsilon_n f_m(x_n)}{\sigma_n^2} \right) \left( \sum_{l=1}^M v_{lk} \sum_{i=1}^N \frac{\varepsilon_i f_l(x_i)}{\sigma_i^2} \right) \right\rangle \\ &= \left\langle \sum_{m=1}^M \sum_{l=1}^M v_{mj} v_{lk} \sum_{i=1}^N \frac{\varepsilon_i^2 f_m(x_i) f_l(x_i)}{\sigma_i^4} \right\rangle \end{aligned} \quad (17)$$

The cross terms in the product of the summations over  $i$  and  $n$  on the first line have been dropped in the second, because the expectation values of the products  $\varepsilon_n \varepsilon_i$  will be zero. So we have

$$\begin{aligned}
\langle \varepsilon_{pj} \varepsilon_{pk} \rangle &= \sum_{m=1}^M \sum_{l=1}^M v_{mj} v_{lk} \sum_{i=1}^N \frac{\langle \varepsilon_i^2 \rangle f_m(x_i) f_l(x_i)}{\sigma_i^4} \\
&= \sum_{m=1}^M \sum_{l=1}^M v_{mj} v_{lk} \sum_{i=1}^N \frac{\sigma_i^2 f_m(x_i) f_l(x_i)}{\sigma_i^4} \\
&= \sum_{m=1}^M \sum_{l=1}^M v_{mj} v_{lk} \sum_{i=1}^N \frac{f_m(x_i) f_l(x_i)}{\sigma_i^2} \\
&= \sum_{m=1}^M \sum_{l=1}^M v_{mj} v_{lk} a_{ml} \\
&= \sum_{m=1}^M v_{mj} \sum_{l=1}^M v_{lk} a_{ml}
\end{aligned} \tag{18}$$

The second summation on the last line just yields the identity matrix, since the elements multiplied and summed belong to matrices that are inverses of each other; the result of the summation is therefore  $\delta_{km}$ , the Kronecker delta, whose value is 1 for  $k = m$  and 0 for  $k \neq m$ . This results in

$$\begin{aligned}
\langle \varepsilon_{pj} \varepsilon_{pk} \rangle &= \sum_{m=1}^M v_{mj} \delta_{km} \\
&= v_{kj} = v_{jk}
\end{aligned} \tag{19}$$

Now we consider the case for correlated errors in the original data, for which a data error covariance matrix  $\Omega_D$  must be provided instead of just the  $N$  uncertainties  $\sigma_i$ , the squares of which are now the diagonal elements of this data error covariance matrix, whose off-diagonal elements are no longer taken to be zero. The more general expression for  $\chi^2$  employs the following definitions.

$$\begin{aligned}
u_i &\equiv y_i - \sum_{k=1}^M p_k f_k(x_i) \\
W &\equiv \Omega_D^{-1} \equiv \begin{pmatrix} w_{11} & w_{12} & \cdots & w_{1N} \\ w_{21} & w_{22} & \cdots & w_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ w_{N1} & w_{N2} & \cdots & w_{NN} \end{pmatrix} \\
\chi^2 &= \sum_{i=1}^N \sum_{j=1}^N w_{ij} u_i u_j \\
&= \sum_{i=1}^N \sum_{j=1}^N w_{ij} \left( y_i - \sum_{k=1}^M p_k f_k(x_i) \right) \left( y_j - \sum_{k=1}^M p_k f_k(x_j) \right)
\end{aligned} \tag{20}$$

Again, we take the derivatives of  $\chi^2$  with respect to the  $p_k$ :

$$\frac{\partial \chi^2}{\partial p_k} = - \sum_{i=1}^N \sum_{j=1}^N w_{ij} \left( f_k(x_i) \left( y_j - \sum_{m=1}^M p_m f_m(x_j) \right) + f_k(x_j) \left( y_i - \sum_{m=1}^M p_m f_m(x_i) \right) \right) \quad (21)$$

where we have changed the dummy summation index  $k$  to  $m$  to distinguish it from the  $f_k$  factors produced by differentiating with respect to  $p_k$ . Setting this equal to zero and regrouping gives:

$$\sum_{i=1}^N \sum_{j=1}^N w_{ij} \left( y_j f_k(x_i) + y_i f_k(x_j) \right) = \sum_{i=1}^N \sum_{j=1}^N w_{ij} \left( f_k(x_i) \sum_{m=1}^M p_m f_m(x_j) + f_k(x_j) \sum_{m=1}^M p_m f_m(x_i) \right) \quad (22)$$

Interchanging the summation order on the right-hand side results in

$$\sum_{i=1}^N \sum_{j=1}^N w_{ij} \left( y_j f_k(x_i) + y_i f_k(x_j) \right) = \sum_{m=1}^M p_m \sum_{i=1}^N \sum_{j=1}^N w_{ij} \left( f_k(x_i) f_m(x_j) + f_k(x_j) f_m(x_i) \right) \quad (23)$$

With the following definitions

$$\begin{aligned} b_k &\equiv \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N w_{ij} \left( y_j f_k(x_i) + y_i f_k(x_j) \right) \\ a_{mk} &\equiv \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N w_{ij} \left( f_k(x_i) f_m(x_j) + f_k(x_j) f_m(x_i) \right) \end{aligned} \quad (24)$$

we obtain

$$b_k = \sum_{m=1}^M a_{mk} p_m \quad (25)$$

which is the same as equation 6 and can be solved in the same way. The inverse of the coefficient matrix is still the error covariance matrix for the model parameters  $p_k$ , but in this case, the derivation must retain the cross terms that were dropped in equation 17, since the expectation values  $\langle \varepsilon_n \varepsilon_i \rangle$  are no longer zero. Note that if we *do* apply these equations to the special case of uncorrelated data errors that we first considered, then  $w_{ij} = 0$  for  $i \neq j$ ,  $w_{ii} = 1/\sigma_i^2$ , and equation 24 reduces to equation 5; the coefficients of 1/2 on the summations in equation 24 are there for this purpose. They cancel out in equation 25 but are required to keep the relationship between the coefficient matrix and error covariance matrix.

One interesting special case is that for  $M = 1$  and  $f_1(x) = 1$ , i.e., the model “equation” is just a constant,  $p_1$ . Considering the case when data errors are uncorrelated, equation 5 becomes

$$\begin{aligned}
b_1 &= \sum_{i=1}^N \frac{y_i}{\sigma_i^2} \\
a_{11} &= \sum_{i=1}^N \frac{1}{\sigma_i^2}
\end{aligned}
\tag{26}$$

and equation 6 becomes

$$\sum_{i=1}^N \frac{y_i}{\sigma_i^2} = \sum_{i=1}^N \frac{p_1}{\sigma_i^2} = p_1 \sum_{i=1}^N \frac{1}{\sigma_i^2}
\tag{27}$$

Solving for  $p_1$  and using equation 9 to get its uncertainty,

$$p_1 = \frac{\sum_{i=1}^N \frac{y_i}{\sigma_i^2}}{\sum_{i=1}^N \frac{1}{\sigma_i^2}}
\tag{28}$$

$$\sigma^2(p_1) = \left( \sum_{i=1}^N \frac{1}{\sigma_i^2} \right)^{-1}$$

So the model is just the inverse-variance-weighted average of the data values  $y_i$ , and the uncertainty variance is the inverse of the coefficient “matrix”, which has been reduced to a scalar in this case. This is the familiar maximum-likelihood Gaussian refinement formula for averaging numbers that have uncertainties, and therefore chi-square minimization is seen to be a generalization of this for models with more elaborate functional dependence than a single scalar.