LEAST-SQUARES AND CHI-SQUARE FOR THE BUDDING AFICIONADO: ART AND PRACTICE

14. FITTING WHEN MORE THAN ONE MEASURED PARAMETERS HAVE UNCERTAINTIES

We've mentioned that one of the essential assumptions of least squares is that the independent variables are known with high precision and the errors occur only in the measured data. Suppose you're fitting two variables, t and y, as in equation **??**. This essential assumption means that t is known with high precision and all the uncertainty is in y, and you are minimizing the squares of the residuals in the y-direction only. If *both* variables have uncertainties, then you have to be careful because the essential assumption is violated. If you go ahead with a standard least-squares fit when there are errors in both coordinates, the slope will be systematically too small.

Thanks to Jefferys (1980), the ML formulation of this problem is straightforward. Nevertheless, there is a lot of confusion on such fitting and not-inconsiderable propagation of myth. Before reviewing Jefferys' formulation, let's see two approaches:

1. Taylor §8.4 argues that you can account for x-variance $\sigma_{x_m}^2$ by increasing the y-variance by the usual error propagation, i.e. define an equivalent y-variance $\sigma_{y_m}^2(equiv) = [\sigma_{y_m}^2 + (a_1\sigma_{x_m})^2]$, where a_1 is the slope. This is equivalent to our results below.

2. Isobe et al. (1990, ApJ 364, 104) discuss the case incorrectly. Look in particular at their Section V, where they make 5 numbered recommendations. Two of these are incorrect:

   (a) Number 3 says, in essence, that if you have measurement errors in y but not in x, and want to predict x from y in some future dataset, that you should least-squares fit the x values (which have no errors) to the y. This is *flat wrong*. Again, it leads to a slope that is systematically too small. The proper procedure is to fit y to x in the standard way, which is consistent with the ML formulation and gives the right answer; then use the resulting parameters, whose errors you know about, to predict x from y in the future.

   (b) Number 4 says that if both x and y have errors, and your main focus is finding the true slope, you should use their "bisector" method. I won't explain this because this concept is wrong.

14.1. A preliminary: Why the slope is systematically small

Why is the derived slope systematically too small if you use the standard least-squares technique when both variables have errors? To see this, take a look back at equation **??**, where we explicitly write the normal equations for fitting a straight line of the form $As_m + Bt_m = y_m$. To focus the discussion and make it easy, replace that problem with a single-parameter solution for only the slope B, and use the usual variables (x, y) in place of (t, y). Then we are fitting the set of M equations

$$Bx_m = y_m \ . \tag{14.1a}$$

The set of two normal equations becomes just the single equation

$$B[x^2] = [xy] \ , \tag{14.1b}$$

or, writing out the sums explicitly,

$$B = \frac{\sum_{m=0}^{M-1} {x^*}_m y_m}{\sum_{m=0}^{M-1} {x^*}_m^2} \ . \tag{14.1c}$$

Here we use the star to designate the perfectly-known independent variable ${x^*}_m$. It is important to realize that the x_m that appear in this equation are the perfectly-known ones ${x^*}_m$; this is a fundamental tenet of least-squares fitting, which comes from the concept and principle of maximum likelihood ML.

Because B is defined by the ${x^*}_m$ and we are asking what happens when we use the imperfectly known x_m instead, let us reduce the problem to the essence and imagine that y_m is perfectly known, i.e. $y_m = {y^*}_m$; and that

$${x^*}_m = x_m - \delta x_m \ , \tag{14.2}$$

where δx_m is the observational error in point m. If we do standard least squares on this situation, then we (incorrectly) rewrite equation 14.1c to read

$$B = \frac{\sum_{m=0}^{M-1} x_m y_m}{\sum_{m=0}^{M-1} x_m^2} \ , \tag{14.3a}$$

that is, using x_m instead of ${x^*}_m$ (because we don't know what x^*_m is!). Substituting equation 14.2, and remembering that $y_m = {y^*}_m = B{x^*}_m$, we have

$$B = \frac{\sum_{m=0}^{M-1} y^*{}_m (x^*{}_m + \delta x_m)}{\sum_{m=0}^{M-1} (x^{*2}_m + 2x^*{}_m \delta x_m + \delta x_m^2)} \ . \tag{14.3b}$$

Now all terms having δx_m sum to zero because the errors are distributed symmetrically around zero. But the denominator contains δx_m^2. The denominator is irrevocably increased by this term, which decreases the derived value of B from its true value. Yes, this is only a second-order effect, but it matters—after all, χ^2 is a second-order quantity! Try some numerical experiments!

14.2. Jefferys' Method: Introduction

Elsewhere we have regarded y as the dependent variable with measured errors and x (and its multiple counterparts) as independent variables that have no measured errors. But sometimes *all* of the measured parameter's have uncertainties. In this case the distinction between "dependent" and "independent" variables disappears and we need to treat them symmetrically. Moreover, we can have an arbitrarily large number of variables. This means we need to generalize our notation—and our mathematical technique. Our treatment follows Jefferys (1980) and we will adopt his notation in some measure. To explain the method we will follow his example and present the general case and also show the application to a specific example.

Figure 14.1 compares a conventional fit to $y = a_0 + a_1 t$ with a proper fit when both variables have uncertainties. The right-hand panel is the conventional fit in which the measurement uncertainties in t are set equal to zero; the left-hand panel includes the uncertainties in t. Some important differences are apparent:

1. Errors on the left panel are completely specified by errorbars. On the right panel we could use vertical errorbars for y and horizontal ones for t, but these would be sufficiently only if the errors were uncorrelated (as they are for points 0, 2, and 4). However, when they are correlated we must use error *ellipses* that specify $\chi^2 = 1$.

2. For the left panel, the best-fit points have the same t values as the datapoints but different y values. For the right panel the best-fit values differ from the data values for *both* variables.

3. If you look carefully, you'll see that the fitted slope on the left is a bit smaller than that on the right. This systematic bias arises from ignoring the errors in t.

14.3. The Data Matrix and Vector

First, a word on notation. We distinguish between scalars, vectors, and matrices as follows: for Roman letters, vectors are **lower-case bold** and matrices are **UPPER-CASE BOLD**. For Greek

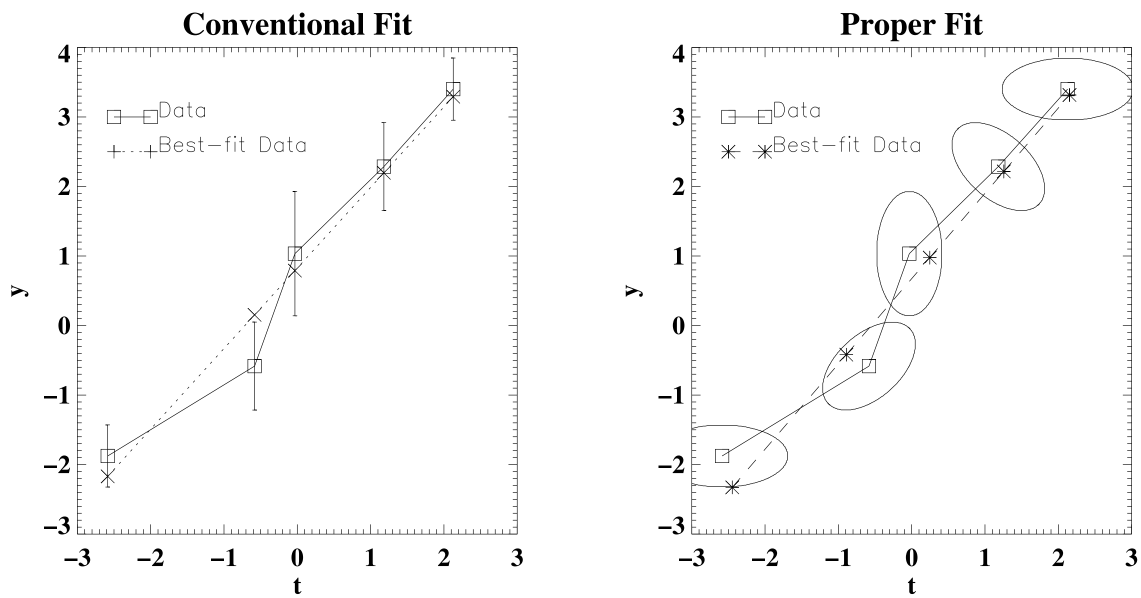

Fig. 14.1.— Comparing a conventional fit for to $y = a_0 + a_1 t$ (left panel) to a proper one when both measured variables have errors. On the right, the ellipses denote the correlated errors in (t, y); these are the generalization of the errorbars on the left. The right-hand slope is a bit steeper than the left-hand one.

letters, scalars are the letter itself (e.g. α), vectors are single-underlined (e.g. $\underline{\phi}$), and matrices are double-underlined (e.g. $\underline{\underline{\sigma}}$).

For the general case we have M experiments. In each experiment we measure J parameters. We want to combine these $(M \times J)$ measurements to derive N coefficients. We denote the matrix of measured parameters by $\mathbf{X}$, which is $(M \times J)$ [using conventional matrix notation, not IDL notation, in which in the vertical dimension (number of rows) is M and the horizontal dimension (number of columns) is J]. We will want to concatenate this matrix to a one-dimensional vector $\mathbf{x}$ of length (MJ). Specifically, the $M \times J$ datapoint matrix is

$$\mathbf{X} = \begin{bmatrix} x_{0,0} & x_{0,1} & x_{0,2} & \cdots & x_{0,J-1} \\ x_{1,0} & x_{1,1} & x_{1,2} & \cdots & x_{1,J-1} \\ x_{2,0} & x_{2,1} & x_{2,2} & \cdots & x_{2,J-1} \\ . & . & . & \cdots & . \\ . & . & . & \cdots & . \\ . & . & . & \cdots & . \\ x_{M-1,0} & x_{M-1,1} & x_{M-1,2} & \cdots & x_{M-1,J-1} \end{bmatrix} \tag{14.4a}$$

We don't use this big matrix in the solution. Instead, we turn it into a vector in which the first J elements are the data for $m = 0$, the second J elements are for $m = 1$, etc. So the vector has dimensions $(MJ) \times 1$, like this:

$$\mathbf{x} = \begin{bmatrix} x_{0,0} \\ x_{0,1} \\ x_{0,2} \\ . \\ . \\ . \\ x_{0,J-1} \\ x_{1,0} \\ x_{1,1} \\ x_{1,2} \\ . \\ . \\ . \\ x_{1,J-1} \\ x_{2,0} \\ x_{2,1} \\ x_{2,2} \\ . \\ . \\ . \\ x_{2,J-1} \\ \ldots \\ \ldots \\ \ldots \\ x_{M-1,0} \\ x_{M-1,1} \\ x_{M-1,2} \\ . \\ . \\ . \\ x_{M-1,J-1} \end{bmatrix} \tag{14.4b}$$

One important reason for writing the whole set of data as a vector instead of a matrix is to make it possible to write the covariance matrix for all measured data in the conventional form, as we now discuss.

14.4. The Data Covariance Matrix and Defining Chi-Square

The whole object of fitting is to minimize the chi-square. When all measured parameters have uncertainties, their uncertainties can be correlated. We have to generalize the definition of chi-square accordingly.

First, suppose that the observational errors in the datapoints are uncorrelated. Then the intrinsic variance of each datapoint is described by a single number. In our example, for uncorrelated observational errors we'd have the variances in the y values be $(\sigma_{y0}^2, \sigma_{y1}^2, \ldots)$, and similarly for the t values; this would give

$$\chi^2 = \sum_m \frac{\delta y_m^2}{\sigma_{ym}^2} + \frac{\delta t_m^2}{\sigma_{tm}^2} \tag{14.5}$$

However, it is common that errors are correlated. For example, if we were fitting y to a polynomial in t, then the errors in the various powers of t would certainly be correlated. More generally, then, the *covariances* among the different measured values are nonzero. These covariances are the off-diagonal terms in the covariance matrix. Thus, if we denote the covariance matrix for the measured $\mathbf{x}$ values by the $(MJ \times (MJ)$ matrix $\underline{\underline{\sigma}}$, then the most general case has $\underline{\underline{\sigma}}$ with no nonzero elements.

Less general, but much more common, is the situation shown in Figure 14.1. Here, the covariances among the J measured parameters nonzero are for a *particular* experiment m, but the covariances from one experiment m to another are zero; in other words, each experiment is completely independent of the others. In this less general but very common case, the covariance matrix looks like

$$\underline{\underline{\sigma}} = \begin{bmatrix} \underline{\underline{\sigma_0}} & \mathbf{0} & \mathbf{0} & \ldots \\ \mathbf{0} & \underline{\underline{\sigma_1}} & \mathbf{0} & \ldots \\ \mathbf{0} & \mathbf{0} & \underline{\underline{\sigma_2}} & \ldots \\ . & & & \\ . & & & \\ . & & & \end{bmatrix} \tag{14.6}$$

Here, each element (including the $\mathbf{0}$ elements) is itself a $J \times J$ matrix. For our specific example of §14.7, $J = 2$ so $\underline{\underline{\sigma_0}}$ is a covariance matrix of the form

$$\underline{\underline{\sigma_0}} = \begin{bmatrix} \sigma_{yy} & \sigma_{yt} \\ \sigma_{yt} & \sigma_{tt} \end{bmatrix} \tag{14.7}$$

Generally, the chi-square is given by (e.g. Cowan equation 2.39)

$$\chi^2 = \delta\mathbf{x^T} \cdot \underline{\underline{\sigma^{-1}}} \cdot \delta\mathbf{x} \tag{14.8}$$

14.5. Formulation of the Problem and its Solution with Lagrange Multipliers

We will be referring to various versions of the data parameters $\mathbf{x}$ and derived parameters $\mathbf{a}$: *measured*, *best-fit*, and (for the iterative solution) *guessed*. The subscript d denotes the set of *measured datapoints*, of which there are (JM). The subscript $*$ denotes the set of *best-fit* quantities; these parameters include not only the datapoints $\mathbf{x}$, but also the derived parameters $\mathbf{a}$. We will be doing an iterative fit using *guessed* values of both the data and derived parameters, represented by the subscript g.

We begin by writing *exact* equations for each measurement. The fitted vales, subscripted with stars, satisfy the *exact* equations of condition

$$\boldsymbol{f(x_*, a_*) = 0} \tag{14.9a}$$

This is an M-long vector of functions $\mathbf{f(x,a) = 0}$ (one row for each measurement). This set of M equations doesn't do us much good because we don't know the best-fit (starred) values. Consequently, for the datapoints we define the difference between the best-fit and measured data values

$$\delta\mathbf{x} = \mathbf{x_d} - \mathbf{x}_* \tag{14.9b}$$

This is the *negative* of Jefferys' definition of the corresponding quantity $\hat{\mathbf{v}}$ in his section II. With this, the equation 14.9a becomes

$$\mathbf{f}(\mathbf{x_d} - \delta\mathbf{x}, \mathbf{a}_*) = \mathbf{0} \ . \tag{14.9c}$$

Our goal is to solve these M equations for the (MJ) differences $\delta\mathbf{x}$ and the N parameters $\mathbf{a}_*$ and, simultaneously, minimize χ^2.

This is a classic minimization problem: we minimize χ^2 with respect to the $(MJ+N)$ values of $\delta\mathbf{x}$ and $\mathbf{a}$, subject to the M constraints of equation 14.9c. Such problems are solved using Lagrange multipliers. Here, the M Lagrange multipliers form the vector $\underline{\lambda}$. We define the Lagrangian $\mathcal{L}$ as

$$\mathcal{L} = \left[\frac{1}{2}\delta\mathbf{x^T} \cdot \underline{\underline{\sigma^{-1}}} \cdot \delta\mathbf{x}\right] + \left[\mathbf{f^T}(\mathbf{x_d} - \delta\mathbf{x}, \mathbf{a}) \cdot \underline{\lambda}\right] \ ; \tag{14.10}$$

the $\frac{1}{2}$ arises because, for a Gaussian pdf for the errors, the residuals are distributed as $e^{-\frac{\chi^2}{2}}$ (e.g. Cowan equation 2.28). We differentiate $\mathcal{L}$ with respect to each of the unknowns $\delta\mathbf{x}$ and $\mathbf{a}$. The solution provides $\mathbf{a}$, the vector of N derived coefficients (so $\mathbf{a}$ is defined as elsewhere in this tutorial) together with the MJ fitted datapoints. To proceed, we need the derivatives of $\mathbf{f}$ in equation 14.9a with respect to the vectors $\mathbf{x}$ and $\mathbf{a}$, as we now discuss.

14.6. The Derivative Matrices

For the analysis we will need the derivatives of $\mathbf{f}$ with respect to the vectors $\mathbf{x}$ and $\mathbf{a}$. The derivatives will always be evaluated at the *guessed* values $\mathbf{x_g}$ and $\mathbf{a_g}$. The derivative with respect to $\mathbf{x}$ is a $M \times MJ$ matrix and looks like this [We take the case $(M = 4, J = 3)$ for transparency; subscripts of x are in the order (m, j)]:

$$\left.\frac{\partial \mathbf{f}(\mathbf{x},\mathbf{a})}{\partial \mathbf{x}}\right|_{\mathbf{x_g},\mathbf{a_g}} = \begin{bmatrix} \frac{\partial \mathbf{f}}{\partial x_{0,0}} & \frac{\partial \mathbf{f}}{\partial x_{0,1}} & \frac{\partial \mathbf{f}}{\partial x_{0,2}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{\partial \mathbf{f}}{\partial x_{1,0}} & \frac{\partial \mathbf{f}}{\partial x_{1,1}} & \frac{\partial \mathbf{f}}{\partial x_{1,2}} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{\partial \mathbf{f}}{\partial x_{2,0}} & \frac{\partial \mathbf{f}}{\partial x_{2,1}} & \frac{\partial \mathbf{f}}{\partial x_{2,2}} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{\partial \mathbf{f}}{\partial x_{M-1,0}} & \frac{\partial \mathbf{f}}{\partial x_{M-1,1}} & \frac{\partial \mathbf{f}}{\partial x_{M-1,2}} \end{bmatrix} \tag{14.11}$$

where all the derivatives are evaluated at $(\mathbf{x_g}, \mathbf{a_g})$. Much easier is the derivative with respect to $\mathbf{a}$, which is a $M \times N$ matrix and looks like

$$\left.\frac{\partial \mathbf{f}(\mathbf{x},\mathbf{a})}{\partial \mathbf{a}}\right|_{\mathbf{x_g},\mathbf{a_g}} = \begin{bmatrix} \left.\frac{\partial \mathbf{f}}{\partial \mathbf{a_0}}\right|_{\mathbf{x_{g,0}},\mathbf{a_g}} & \left.\frac{\partial \mathbf{f}}{\partial \mathbf{a_1}}\right|_{\mathbf{x_{g,0}},\mathbf{a_g}} & \left.\frac{\partial \mathbf{f}}{\partial \mathbf{a_2}}\right|_{\mathbf{x_{g,0}},\mathbf{a_g}} & \cdots & \left.\frac{\partial \mathbf{f}}{\partial \mathbf{a_{N-1}}}\right|_{\mathbf{x_{g,0}},\mathbf{a_g}} \\ \left.\frac{\partial \mathbf{f}}{\partial \mathbf{a_0}}\right|_{\mathbf{x_{g,1}},\mathbf{a_g}} & \left.\frac{\partial \mathbf{f}}{\partial \mathbf{a_1}}\right|_{\mathbf{x_{g,1}},\mathbf{a_g}} & \left.\frac{\partial \mathbf{f}}{\partial \mathbf{a_2}}\right|_{\mathbf{x_{g,1}},\mathbf{a_g}} & \cdots & \left.\frac{\partial \mathbf{f}}{\partial \mathbf{a_{N-1}}}\right|_{\mathbf{x_{g,1}},\mathbf{a_g}} \\ \left.\frac{\partial \mathbf{f}}{\partial \mathbf{a_0}}\right|_{\mathbf{x_{g,2}},\mathbf{a_g}} & \left.\frac{\partial \mathbf{f}}{\partial \mathbf{a_1}}\right|_{\mathbf{x_{g,2}},\mathbf{a_g}} & \left.\frac{\partial \mathbf{f}}{\partial \mathbf{a_2}}\right|_{\mathbf{x_{g,2}},\mathbf{a_g}} & \cdots & \left.\frac{\partial \mathbf{f}}{\partial \mathbf{a_{N-1}}}\right|_{\mathbf{x_{g,2}},\mathbf{a_g}} \\ & & \cdot & & \\ & & \cdot & & \\ & & \cdot & & \end{bmatrix} \tag{14.12}$$

where again, all the derivatives are evaluated at $(\mathbf{x_g}, \mathbf{a_g})$.

14.7. The Specific Example

We illustrate the above with this specific example, for which we fit a first-order polynomial to (t, y) of the form $[y = a_0 + a_1 t]$; the coefficients are (a_0, a_1) and $[\mathbf{f}(\mathbf{x},\mathbf{a}) = y - a_0 - a_1 t]$. For this example, the matrix of measured parameters is

$$\mathbf{X} = \begin{bmatrix} y_{d0} & t_{d0} \\ y_{d1} & t_{d1} \\ y_{d2} & t_{d2} \\ . & \\ . & \\ . & \end{bmatrix} \tag{14.13}$$

and the concatenated vector version is

$$\mathbf{x} = \begin{bmatrix} y_{d0} \\ t_{d0} \\ y_{d1} \\ t_{d1} \\ y_{d2} \\ t_{d2} \\ . \\ . \\ . \end{bmatrix} \tag{14.14}$$

The exact vector-of-functions equation $\mathbf{f}(\mathbf{x}_*, \mathbf{a}_*) = \mathbf{0}$ uses the starred (best-fit) values, and is

$$\mathbf{f}(\mathbf{x}_*, \mathbf{a}_*) = \begin{bmatrix} y_{*0} - a_{*0} - a_{*1}t_{*0} \\ y_{*1} - a_{*0} - a_{*1}t_{*1} \\ . \\ . \\ . \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ . \\ . \\ . \end{bmatrix} \tag{14.15}$$

The derivative matrix of $\mathbf{f}$ with respect to the vector $\mathbf{x}$ is

$$\left.\frac{\partial \mathbf{f}(\mathbf{x}, \mathbf{a})}{\partial \mathbf{x}}\right|_{\mathbf{x_g}, \mathbf{a_g}} = \begin{bmatrix} 1 & -a_{g1} & 0 & 0 & 0 & 0 & \dots \\ 0 & 0 & 1 & -a_{g1} & 0 & 0 & \dots \\ 0 & 0 & 0 & 0 & 1 & -a_{g1} & \dots \\ & & & . & & & \\ & & & . & & & \\ & & & . & & & \end{bmatrix}, \tag{14.16}$$

always evaluated at the *guessed* values of the parameters (subscript g). The derivative matrix of $\mathbf{f}$ with respect to the vector $\mathbf{a}$ is

$$\left.\frac{\partial \mathbf{f}(\mathbf{x},\mathbf{a})}{\partial \mathbf{a}}\right|_{\mathbf{x_g},\mathbf{a_g}} = \begin{bmatrix} -1 & -t_{g0} \\ -1 & -t_{g1} \\ -1 & -t_{g2} \\ . & \\ . & \\ . & \end{bmatrix} , \tag{14.17}$$

again always evaluated at the *guessed* values of the parameters (subscript g).

14.8. The Solution to the Lagrangian: Two Matrix Equations

Jefferys does all this this[1] for us and, after some algebraic manipulation, provides the two following matrix equations (from the set of four in his equation 10):

$$\underline{\underline{\sigma}}^{\mathbf{-1}} \cdot \delta\mathbf{x} + \left.\frac{\partial \mathbf{f^T}(\mathbf{x},\mathbf{a})}{\partial \mathbf{x}}\right|_{\mathbf{x_d},\mathbf{a_*}} \cdot \underline{\lambda} = \mathbf{0} \tag{14.18a}$$

This single matrix equation embodies MJ individual equations. Here we write $\frac{\partial \mathbf{f^T}}{\partial \mathbf{x}}$ instead of $\frac{\partial \mathbf{f^T}}{\partial \delta\mathbf{x}}$ for clarity, and use the fact that they are the negative of each other.

$$\left.\frac{\partial \mathbf{f^T}(\mathbf{x},\mathbf{a})}{\partial \mathbf{a}}\right|_{\mathbf{x_d},\mathbf{a_*}} \cdot \underline{\lambda} = \mathbf{0} \tag{14.18b}$$

This matrix equation embodies N individual equations. These two matrix equations embody $MJ+N$ individual equations, which is equal to the number of unknowns, so we can solve for them! In both of these equations, the dimensions of the factors are (*Note the transposes!*):

$$\underline{\underline{\sigma}} : (MJ) \times (MJ) \tag{14.19a}$$

$$\delta\mathbf{x} : MJ \times 1 \tag{14.19b}$$

$$\mathbf{f^T} : 1 \times M \tag{14.19c}$$

$$\frac{\partial \mathbf{f^T}(\mathbf{x},\mathbf{a})}{\partial \mathbf{x}} : (MJ) \times M \tag{14.19d}$$

[1]And more: he includes the possibility for additional constraints, the second line in his equation (7). This introduces additional complications and we ignore this possibility in the interest of pedagogical simplification. For example, his set of equations (10) has four equations, not two as we write here.

$$\underline{\lambda} : M \times 1 \tag{14.19e}$$

$$\frac{\partial \mathbf{f^T}(\mathbf{x}, \mathbf{a})}{\partial \mathbf{a}} : N \times M \tag{14.19f}$$

Note that in 14.19d above the dimension is $(MJ) \times M$: M elements in $\mathbf{f}$, differentiated by (MJ) different variables $\mathbf{x}$; similarly for 14.19f above.

14.9. Solving Equations 14.18a and 14.18b Iteratively

Generally, these equations are nonlinear and we solve them iteratively using guessed solutions. We denote the guessed values with subscript g, so we have

$$\mathbf{a_g} = \text{guessed values for } \mathbf{a}_* \tag{14.20a}$$

$$\mathbf{x_g} = \text{guessed values for } \mathbf{x}_* \tag{14.20b}$$

ITERATION STEP 1: We define the difference between the measured data quantities and their currently-guessed counterparts

$$\mathbf{\Delta x_g} \equiv \mathbf{x_d} - \mathbf{x_g} \tag{14.20c}$$

Above, our $\mathbf{\Delta x_g}$ is the *negative* of Jefferys' $\hat{\mathbf{v}}$. The nonlinear fit solves for corrections to these guesses, which we denote by $\mathbf{\Delta x_{new}}$ (the negative of Jefferys' $\hat{\mathbf{v}}_{\mathbf{new}}$) and $\mathbf{\Delta a_{new}}$ (which is identical to Jefferys' $\hat{\underline{\delta}}$).

ITERATION STEP 2: We define the following:

1. The $M \times M$ weight matrix $\mathbf{W}$ is from Jefferys' equation (15):

$$\mathbf{W} \equiv \left[\left. \frac{\partial \mathbf{f}(\mathbf{x}, \mathbf{a})}{\partial \mathbf{x}} \right|_{\mathbf{x_g}, \mathbf{a_g}} \cdot \underline{\underline{\sigma}} \cdot \left. \frac{\partial \mathbf{f^T}(\mathbf{x}, \mathbf{a})}{\partial \mathbf{x}} \right|_{\mathbf{x_g}, \mathbf{a_g}} \right]^{-1} \tag{14.21a}$$

2. The $[0, 0]$ element of his equation (17), which is equivalent to our $\underline{\underline{\alpha}}$ (the curvature matrix) elsewhere in this document:

$$\underline{\underline{\alpha}} \equiv \left. \frac{\partial \mathbf{f^T}(\mathbf{x}, \mathbf{a})}{\partial \mathbf{a}} \right|_{\mathbf{x_g}, \mathbf{a_g}} \cdot \mathbf{W} \cdot \left. \frac{\partial \mathbf{f}(\mathbf{x}, \mathbf{a})}{\partial \mathbf{a}} \right|_{\mathbf{x_g}, \mathbf{a_g}} \tag{14.21b}$$

$\underline{\underline{\alpha}}$ is $N \times N$.

3. The modified equations of condition from his equation (18) (we have a + instead of his − because $\mathbf{\Delta x_g} = -\hat{\mathbf{v}}$)

$$\underline{\phi_{\mathbf{g}}} \equiv \mathbf{f}(\mathbf{x_g}, \mathbf{a_g}) + \left(\left. \frac{\partial \mathbf{f}(\mathbf{x}, \mathbf{a})}{\partial \mathbf{x}} \right|_{\mathbf{x_g}, \mathbf{a_g}} \right) \cdot \mathbf{\Delta x_g} \tag{14.21c}$$

$\underline{\phi_{\mathbf{g}}}$ is $M \times 1$.

ITERATION STEP 3: The solutions follow directly. The matrix equation for $\mathbf{\Delta a_{new}}$ is Jefferys equation (17)

$$\underline{\underline{\alpha}} \cdot \mathbf{\Delta a_{new}} = - \left. \frac{\partial \mathbf{f^T}(\mathbf{x}, \mathbf{a})}{\partial \mathbf{a}} \right|_{\mathbf{x_g}, \mathbf{a_g}} \cdot \mathbf{W} \cdot \underline{\phi_{\mathbf{g}}} \tag{14.22}$$

which is solved for $\mathbf{\Delta a_{new}}$ in the conventional way by premultiplying both sides by $\underline{\underline{\alpha}}^{-1}$. The matrix equation for the new, corrected $\mathbf{\Delta x_g}$ is Jefferys equation (19)

$$\mathbf{\Delta x_{new}} = -\underline{\underline{\sigma}} \cdot \left. \frac{\partial \mathbf{f^T}(\mathbf{x}, \mathbf{a})}{\partial \mathbf{x}} \right|_{\mathbf{x_g}, \mathbf{a_g}} \cdot \mathbf{W} \cdot \left(\underline{\phi_{\mathbf{g}}} + \left. \frac{\partial \mathbf{f}}{\partial \mathbf{a}} \right|_{\mathbf{x_g}, \mathbf{a_g}} \cdot \mathbf{\Delta a_{new}} \right) \tag{14.23}$$

ITERATION STEP 4: The previous two equations provide corrections to the values $\mathbf{x_d}$ and $\mathbf{a_g}$. One applies them (Jefferys equation 20):

$$\mathbf{a_{g,new}} = \mathbf{a_g} + \mathbf{\Delta a_{new}} \tag{14.24a}$$

$$\mathbf{x_{g,new}} = \mathbf{x_d} + \mathbf{\Delta x_{new}} \tag{14.24b}$$

Note that $\mathbf{\Delta x_{new}}$ is applied to $\mathbf{x_d}$, not to $\mathbf{x_g}$.

ITERATION STEP 5: Return to **Iteration Step 1**, using these new values $\mathbf{a_{g,new}}$ and $\mathbf{x_{g,new}}$ in place of $\mathbf{a_g}$ and $\mathbf{x_g}$. Iterate until convergence. Convergence occurs when all derived corrections $\mathbf{\Delta a_{new}}$ and $\mathbf{\Delta x_{new}}$ become negligible. [How do you define "negligible"...]

14.10. Taking all those derivatives!

Why not use numerical derivatives? It's easier with complicated functions!

14.11. The Initial Guess

Getting started is the hard part unless the problem is relatively easy. One way is to use conventional least squares to derive the initial guesses $\mathbf{a_g}$. To do this, you designate one particular variable as the dependent variable and all the others as the independent ones (for which you assume no errors). Do the conventional least squares solution for the parameters $\mathbf{a}$ and use these as the initial guesses. A good choice for the dependent variable is the one with the largest errors, if there is one; otherwise the choice is more-or-less arbitrary. For the initial guesses of the data parameters $\mathbf{x_g}$, just use the measured data values $\mathbf{x_d}$. If this scheme doesn't work, you're on your own!

14.12. The Covariance Matrix (and errors) of the Derived Parameters

The matrix $\underline{\underline{\alpha}}$, which is defined above in equation 14.21b, is the conventionally-defined curvature matrix and its inverse $\underline{\underline{\alpha^{-1}}}$ is the conventionally-defined covariance matrix. Because we have formulated this problem as a chi-squared one, the elements of this matrix give the covariance directly. *Thus, the errors in the derived parameters can be taken as the square-root of the diagonal elements of* $\underline{\underline{\alpha^{-1}}}$.

One usually wants to calculate the chi-squared of the solution. This involves not only the best-fit parameters $\mathbf{a}_*$ but also the best fit datapoints $\mathbf{x}_*$. To do this, use equation 14.8 using $\mathbf{\Delta x_g}$ in place of $\delta\mathbf{x}$. This is the same as Jefferys' equation (43).

The expectation value of χ^2 is the number of degrees of freedom. In one sense this is just like the usual least-squares solution: it's equal to the number of datapoints minus the number of derived parameters. Here the number of datapoints is not the number of experiments M; rather, it's JM. So the number of degrees of freedom is $JM - N$.