4 Formulating the optimization problem

We assess how well a set of estimated parameters fits the data by returning to equations 1-5, plugging in our observations and estimates:

$$
\begin{aligned}
e_{1j} &= x_{1j} - \hat{x}_{1j} = x_{1j} - \sqrt{\hat{r}_1^2 - \hat{z}_j^2} && (6)\\
e_{2j} &= x_{2j} - \hat{x}_{2j} = x_{2j} - \sqrt{\hat{r}_2^2 - \hat{z}_j^2} && (7)\\
e_{3j} &= x_{3j} - \hat{x}_{3j} = x_{3j} - \sqrt{\hat{r}_3^2 - (\hat{z}_j - \hat{z}_o)^2} && (8)\\
e_{4j} &= x_{4j} - \hat{x}_{4j} = x_{4j} - \sqrt{\hat{r}_4^2 - (\hat{z}_j - \hat{z}_o)^2} && (9)\\
e_{10} &= x_{10} - \hat{x}_{10} = x_{10} - \hat{r}_1 && (10)
\end{aligned}
$$

We seek to maximize the likelihood

$$\prod_{\forall i,j} \frac{1}{\sqrt{2\pi}\sigma_{ij}} \exp\left(-\frac{e_{ij}^2}{2\sigma_{ij}^2}\right)$$

or, after taking logs and dropping constant terms[1], minimize the sum

$$L = \sum_{\forall i,j} \frac{e_{ij}^2}{\sigma_{ij}^2} \tag{11}$$

In the special case where all the $\sigma_{ij} = 1$ this reduces to simply minimizing $\sum e_{ij}^2$. We believe that, in the real world, the ϵ_{ij} are independent, as each results from a separate series of measurements on a different part of a particle photograph.

The maximum likelihood estimate is the global minimum of L, that is, the (usually unique) point in $m + k + 1$-dimensional space at which

$$\frac{\partial L}{\partial \hat{r}_i} = \frac{\partial L}{\partial \hat{z}_j} = 0 \tag{12}$$

for all i, j, and all the second derivatives are nonnegative.

Fortunately, all of equations 6-10 are easily differentiable, so it is easy to find the gradient of L to use the method of steepest descent, or the gradient and Hessian of L to use Newton's method. When a random variable is approximately normally distributed, the variance-covariance matrix of the estimates can be obtained by inverting the Hessian matrix of L at the MLE. [1]

[1] In the statistical literature, usually L refers to this formula after taking logs but *before* dropping all the constant terms, but we will have no need to return to the raw log-likelihood, so slightly abuse L for ease of reading.

The method of steepest descent is simplest to implement. However, it is known to suffer slow convergence when the minimum is in a long narrow diagonal "valley" – which almost certainly will be the case here: if both the radii and the vertical offset increase together, nearly the same circular cross-sections will be obtained. Newton's method obtains much faster convergence but requires inversion of a $m+k+1 \times m+k+1$ matrix at each iteration. If inversion of the Hessian is not feasible, an alternative is to use one of the "quasi-Newton methods," approximating the Hessian via the Broyden, DFP, or BFGS formula.[2]

To find the minimum of a one-variable function $f(x)$, Newton's method starts with an initial guess x_o, and improves that guess by examining the slope and curvature of f at the most recent guess:

$$x_{i+1} = x_i - \frac{f'(x)}{f''(x)}$$

In the multi-variable case, $f'(x)$ becomes the gradient (vector of partial derivatives with respect to each variable), $f''(x)$ becomes the Hessian (matrix of second partial derivatives with respect to each possible pair of variables), and "dividing by" $f''(x)$ means multiplying by the inverse of a matrix:

$$\begin{aligned} \vec{x}_{i+1} &= \vec{x}_i - \mathbf{H}^{-1}(x) \cdot \nabla f(x), \quad \text{where} \\ \nabla f(x) &= \left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, ..., \frac{\partial f}{\partial x_n} \right) \quad \text{and} \\ \mathbf{H} &= \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix} \end{aligned} \tag{13}$$

In the present application the vector $\vec{x}$ is the list of simultaneous estimates for all $m+k+1$ variables: $(\hat{r}_1, \hat{r}_2, ...\hat{r}_k, \hat{z}_o, \hat{z}_1, ...\hat{z}_m)$, and the partial derviatives are $\frac{\partial L}{\partial \hat{r}_1}, ...\frac{\partial L}{\partial \hat{z}_4}$, etc.

5 Implementation

For the current application, Newton's method was implemented in *Mathematica*, after confirming the inadequacy of a naive steepest-descent approach. *Mathematica* has built-in capacity for quickly inverting matrices of reasonable size, so it was not necessary to resort