Gaussian profile estimation in one dimension

Nathan Hagen*, Matthew Kupinski, and Eustace L. Dereniak

College of Optical Sciences, University of Arizona, Tucson, Arizona 85721, USA

*Corresponding author: nhagen@optics.arizona.edu

Received 23 February 2007; revised 10 May 2007; accepted 11 May 2007;
posted 21 May 2007 (Doc. ID 80302); published 23 July 2007

We present several new results on the classic problem of estimating Gaussian profile parameters from a set of noisy data, showing that an exact solution of the maximum likelihood equations exists for additive Gaussian-distributed noise. Using the exact solution makes it possible to obtain analytic formulas for the variances of the estimated parameters. Finally, we show that the classic formulation of the problem is actually biased, but that the bias can be eliminated by a straightforward algorithm. © 2007 Optical Society of America

OCIS codes: 030.6600, 120.5240, 120.3940, 300.3700.

Preview

The problem of estimating the parameters of a Gaussian-shaped function in a noisy data set (such as the one shown in Fig. 1) is a common one in many fields. For one-dimensional (1D) data sets, the most prominent use of Gaussian profile estimation is probably in spectroscopy, where it is used to accurately locate the wavelength of a Doppler-broadened emission or absorption line and to estimate the line strength [1–4]. The technique can also be found in applications of position sensing for line-objects which have been blurred by the imaging process [5,6]. For two-dimensional data sets, the two-dimensional (2D) version of the technique is even more widely used, in fields such as in astrometry, wave front sensing, Gaussian beam characterization, bio-imaging, and also in calibrations of volumetric sensors. Achieving the most accurate possible estimate of the desired parameters is of course an important goal in all of these applications.

In this paper we analyze the use of maximum-likelihood estimation (MLE) for a 1D Gaussian profile corrupted by additive Gaussian-distributed noise and discuss the benefits and drawbacks of this approach. The MLE equations for this problem, while not solvable analytically, can nevertheless be solved exactly by a straightforward nonlinear optimization procedure. Previous efforts have all used approximations in order to solve the MLE equations, but keeping to exact solutions allows us to use the Cramer–Rao Bound (CRB) to obtain the parameter variances. The approach is also sufficiently general to allow nonuniform sampling of the profile, thus allowing for bad pixels or missing data. Although the exact form of the parameter covariance matrix is complicated, applying some reasonable approximations can reduce them to surprisingly simple analytic expressions. The various models for the background underneath the Gaussian profile can have an appreciable effect here, and we extend the result to more advanced background models.

We also show that the typical formulation of the Gaussian profile estimation problem produces a biased statistical model, and that the bias error can be large for systems which insufficiently sample the profile. However, because the expression for the bias term is simple, we show a straightforward technique for removing the bias error. In the conclusion, we combine our results for the variances and the bias to discuss the problem of estimating the Gaussian profile position. This has been treated in numerical simulations of previous publications, but our results here do not agree with previously published research.

This paper discusses the Gaussian profile estimation problem only for the 1D case. The extension to 2D is given in a subsequent publication.

0003-6935/07/220001-10$15.00/0

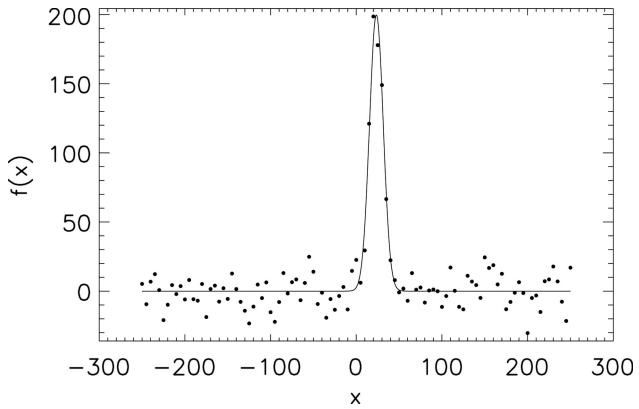


Fig. 1. Example data for a 1D Gaussian profile, generated from the signal ($A = 100.0$, $\bar{x} = 23.5$, $w = 8.0$, $\sigma = 10.0$, $\delta_x = 1$, and $M = 101$). The solid line represents the object $f(x)$ (true underlying Gaussian profile); the filled circles represent the measured data values, $\mathbf{g}$.

1. Introduction

One of the great advantages of MLE, as opposed to least-squares minimization, is that for a large number of measurements it provides the minimum-variance unbiased estimate of the parameter set–it is "efficient". If there is any prior information on the various parameters (such as the positivity of the Gaussian peak's amplitude and width) then it is well-known that Bayesian estimation techniques can provide better estimates. The disadvantage with Bayesian methods is that they result in more complex expressions both for the parameter estimators and for their variances, such that numerical rather than analytic solution techniques must be used.

An additional advantage of MLE is that the Cramer–Rao bound provides (again, asymptotically for the case of a large number M of measurements) not just a lower bound on the variance but the variance itself. Thus, it is unnecessary to locate the parameter variances by some numerical or Monte Carlo method; a straightforward technique is available which can provide analytical expressions for the variances of the fitted parameters.

We start with a zero-background model for the estimation problem (see Appendix A for the nonzero background model). The deterministic function we wish to measure as accurately as possible is what we will refer to as the "object" $f(x)$, the Gaussian profile

$$f(x) = Ae^{-(x-\bar{x})^2/2w^2}, \tag{1}$$

where A is the peak amplitude, $\bar{x}$ is the position of the peak (the center), and w the Gaussian width ($w = \mathrm{FWHM}/\sqrt{8 \ln 2} \approx \mathrm{FWHM}/2.35$). The quantity we wish to measure is a continuous function, but the detection process samples it to produce a discrete data vector $\bar{\mathbf{g}}$:

$$\bar{\mathbf{g}} = \mathcal{H}f(x).$$

Here $\mathcal{H}$ is an operator which maps a continuous function onto a discrete one. Typically, $\mathcal{H}$ is chosen such that

$$\begin{aligned}\bar{g}_m &= \delta_x \int_{-\infty}^{\infty} f(x)\delta(x - x_m)\mathrm{d}x \\ &= \delta_x A e^{-(x_m-\bar{x})^2/2w^2},\end{aligned} \tag{2}$$

where $\delta(\cdot)$ is the Dirac δ-function, m is the pixel index ($1 \le m \le M$), δ_x is the pixel width, and x_m gives the abscissa position for the center of pixel m. (In this paper, we assume that the pixel width coincides with the spacing between samples—i.e., 100% fill factor.) While we are not compelled to make this choice for $\mathcal{H}$, it produces a model which is simple both analytically and conceptually. The disadvantage of the choice is that it oversimplifies the measurement process and introduces a bias into the result. In Section 6 below we present a more realistic operator for $\mathcal{H}$ and discuss how to modify the problem to incorporate the necessary changes.

When the object $f(x)$ is detected, such as by an array detector, the measurement result $\mathbf{g}$ can be modeled as the sum of a noiseless discrete data vector, $\bar{\mathbf{g}}$, and a random perturbation $\mathbf{n}$

$$\mathbf{g} = \bar{\mathbf{g}} + \mathbf{n} = \mathcal{H}f(x) + \mathbf{n}, \tag{3}$$

where the noise vector $\mathbf{n}$ is a Gaussian-distributed random variable, such that

$$\mathrm{pr}(g_m) = \frac{1}{\sqrt{2\pi\sigma_m{}^2}} e^{-(g_m-\bar{g}_m)^2/2\sigma_m{}^2}, \tag{4}$$

and $\sigma_m{}^2$ is the variance of the noise at pixel m. This approach and nomenclature, adopted from the field of image science, [7] may appear as excessive mathematical machinery for the problem but greatly simplifies the bias discussion (Section 6).

There are several reasons why we choose to base our derivation on Gaussian statistics rather than Poisson statistics. While one reason is that the Gaussian distribution closely approximates the Poisson case for signals with more than a few digital counts, the primary reason is that the Gaussian model is more flexible, allowing the ability to work with postprocessed data. For example, applying pixel gain nonuniformity correction ("flat-fielding") to the data results in pixel values which are real-valued and not integer-valued, violating the assumption required for proper Poisson statistics. Applying background subtraction invariably produces a large number of negative-valued pixels in the data, which is of course problematic in Poisson statistics. In a naïve approach it is common to truncate the negative values by setting them to zero and then going on to perform the estimation. This obviously introduces a positive bias into the result. Another approach to the background problem is to incorporate parameters in the fit which model the background signal as well, making direct subtraction unnecessary.

This may or may not be a good choice, because the introduction of extra parameters can significantly worsen the accuracy of estimation for the remaining parameters. Also problematic is the fact that this approach reduces the stability of the optimization routine. It is usually best to avoid these issues by performing an independent measurement in which the background estimation can be done accurately.

The ML estimation process steps are:

1. Construct the likelihood function $L = \mathrm{pr}(\mathbf{g}\,|\,\boldsymbol{\theta})$, the probability measuring the data given a set of object parameters $\boldsymbol{\theta} = (A, \bar{x}, w)$.
2. Calculate the gradient of the log-likelihood function, often referred to as the "score": $\boldsymbol{\nabla}\ell = \frac{\partial}{\partial\theta_i}\ln L$.
3. Solve for the parameter set which maximizes the score.

The probability of measuring a single data value g_m is given by $\mathrm{pr}(g_m\,|\,\boldsymbol{\theta})$. Assuming independence among all measured data, the likelihood of the entire data set is the product of the individual measurements,

$$L = \mathrm{pr}(\mathbf{g}|\boldsymbol{\theta}) = \prod_{m=1}^{M}\left(\frac{1}{2\pi\sigma_m^{\,2}}\right)^{1/2}\exp\left[-\frac{(g_m-\bar{g}_m)^2}{2\sigma_m^{\,2}}\right] = \left[\prod(2\pi\sigma_m^{\,2})^{-1/2}\right]\exp\left(-\sum_{m=1}^{M}\frac{(g_m-\bar{g}_m)^2}{2\sigma_m^{\,2}}\right). \tag{5}$$

Taking the logarithm to obtain the log-likelihood,

$$\ell = \ln L = \left[-\frac{1}{2}\sum\ln(2\pi\sigma_m^{\,2})\right] - \sum_{m=1}^{M}\frac{(g_m-\bar{g}_m)^2}{2\sigma_m^{\,2}}.$$

Everything we want to estimate uses partial derivatives of ℓ with respect to the various parameters, so that we are free to drop any terms independent of the parameters, giving

$$\ell \sim -\sum_{m=1}^{M}\frac{1}{2\sigma_m^{\,2}}(g_m-\bar{g}_m)^2 \sim \sum\frac{1}{\sigma_m^{\,2}}(g_m\bar{g}_m - \frac{1}{2}\bar{g}_m^{\,2}).$$

The elements of the gradient then have the form

$$\frac{\partial\ell}{\partial\theta_i} = \sum\frac{1}{\sigma_m^{\,2}}(g_m-\bar{g}_m)\frac{\partial\bar{g}_m}{\partial\theta_i}. \tag{6}$$

2. Nonlinear Optimization by Newton's Method

To obtain the ML estimators $\hat{A}$, $\hat{\bar{x}}$, and $\hat{w}$ for the parameters A, $\bar{x}$, and w, we can set the appropriate components of the gradient to zero and solve. We quickly find, however, that we obtain a set of three coupled nonlinear equations, foiling access to an analytic solution. Previous approaches to this problem resorted to various approximation techniques to overcome this difficulty, but using a simple nonlinear optimization routine allows access to the exact solution, an advantage which afterwards makes using the Cramer–Rao bound credible. A disadvantage of the use of a nonlinear optimization routine is that it introduces the real possibility of finding false solutions (such as local rather than global maxima) or failing to find a solution at all when the algorithm diverges. Luckily, the solution space for maximum likelihood problems, the log-likelihood function, is generally rather well-behaved when the number of parameters is small and the number of data large, such that pathological behavior in the optimization algorithm is rare.

The most obvious choice for a nonlinear optimization method is Newton's method, both because it is possible to calculate the gradient vector and Hessian matrix for this problem and also for the reason that the Hessian matrix is itself intimately related to a later step in the analysis: calculating the Fisher information matrix. An additional advantage of selecting Newton's method is that it is very fast. A drawback is that it requires a reasonably close starting guess.

Locating a solution via Newton's method involves an iterative approach with the following procedure [8] (where a superscript indicates the iteration index):

1. Obtain initial guesses for the parameters $\boldsymbol{\theta}^{(0)} = (A^{(0)}, \bar{x}^{(0)}, w^{(0)})$
2. Calculate the gradient vector $\boldsymbol{\nabla}\ell^{(k)}$
3. Calculate the Hessian matrix $\mathbf{H}^{(k)}$
4. Check the algorithm stopping conditions
5. Update the parameter estimate with the equation

$$\boldsymbol{\theta}^{(k+1)} = \boldsymbol{\theta}^{(k)} - (\mathbf{H}^{(k)})^{-1}\boldsymbol{\nabla}\ell^{(k)}.$$

If it is necessary to avoid calculating the matrix inverse directly, then the update vector may also be obtained by an iterative solution technique such as QR-decomposition followed by back-substitution. This is a numerically stable method for obtaining a vector $\Delta\boldsymbol{\theta}$ solving the matrix equation $\mathbf{H}^{(k)}\Delta\boldsymbol{\theta} = -\nabla\ell^{(k)}$.

Calculating the elements of the gradient gives

$$\left.\begin{aligned}[\boldsymbol{\nabla}\ell]_1 &= \frac{\partial\ell}{\partial A} = \delta_x\sum E_m\gamma_x\\ [\boldsymbol{\nabla}\ell]_2 &= \frac{\partial\ell}{\partial\bar{x}} = \frac{\delta_x A}{w^2}\sum E_m\xi_m\gamma_x\\ [\boldsymbol{\nabla}\ell]_3 &= \frac{\partial\ell}{\partial w} = \frac{\delta_x A}{w^3}\sum E_m\xi_m^{\,2}\gamma_x\end{aligned}\right\}, \tag{7}$$

where we define $\xi_m = (x_m - \bar{x})$, $E_m = e^{-\xi_m^2/2w^2}$, and $\gamma_m = \frac{g_m-\bar{g}_m}{\sigma_m^{\,2}}$. All of the sums are taken over the pixel index, $1 \le m \le M$. The Hessian matrix for this problem has the form

$$\mathbf{H} \equiv \nabla_{\boldsymbol{\theta}}{}^2 \ell = \begin{pmatrix} \frac{\partial^2 \ell}{\partial A^2} & \frac{\partial^2 \ell}{\partial A \partial \bar{x}} & \frac{\partial^2 \ell}{\partial A \partial w} \\ \frac{\partial^2 \ell}{\partial \bar{x} \partial A} & \frac{\partial^2 \ell}{\partial \bar{x}^2} & \frac{\partial^2 \ell}{\partial \bar{x} \partial w} \\ \frac{\partial^2 \ell}{\partial w \partial A} & \frac{\partial^2 \ell}{\partial w \partial \bar{x}} & \frac{\partial^2 \ell}{\partial w^2} \end{pmatrix}.$$

Since **H** is a symmetric matrix, it has only six independent elements, and so it is only necessary to calculate the upper triangular portion of the matrix. Performing the partial derivative calculations gives for the elements of the matrix

$$\left.\begin{aligned} H_{11} &= -\delta_x{}^2 \sum \frac{1}{\sigma_m{}^2} E_m{}^2 \\ H_{12} &= \frac{\delta_x}{w^2} \sum E_m \xi_m \Gamma_m \\ H_{13} &= \frac{\delta_x}{w^3} \sum E_m \xi_m{}^2 \Gamma_m \\ H_{22} &= \frac{\delta_x A}{w^4} \sum E_m \left[\xi_m{}^2 \Gamma_m - w^2 \gamma_m \right] \\ H_{23} &= \frac{\delta_x A}{w^5} \sum E_m \left[\xi_m{}^3 \Gamma_m - 2 \xi_m w^2 \gamma_m \right] \\ H_{33} &= \frac{\delta_x A}{w^6} \sum E_m \left[\xi_m{}^4 \Gamma_m - 3 \xi_m{}^2 w^2 \gamma_m \right] \end{aligned}\right\}, \tag{8}$$

where $\Gamma_m = \frac{g_m - 2\bar{g}_m}{\sigma_m{}^2}$.

Assuming that the algorithm converges onto the global solution, inserting these formulas into Newton's method solves for the maximum likelihood estimates $\hat{A}$, $\hat{\bar{x}}$, and $\hat{w}$ to desired accuracy.

3. Calculating Parameter Variances

Once we have our ML estimates of the Gaussian function parameters, we would like to know how accurate they are. Previous approaches to this problem have used numerical methods to determine the parameter variances [1,4]. We show that not only is it possible to obtain the exact expression for the bound on the parameter covariance matrix, but with some reasonable approximations the variances can be reduced to simple analytical formulas. With these formulas it is straightforward to analyze the general behavior of the system.

Cramer–Rao bound theory states that the covariance matrix **K** of the parameters is bounded by the inverse of the Fisher information matrix **F** (Barrett and Myers p. 897):

$$\mathbf{K} \geq \mathbf{F}^{-1}.$$

For the majority of problems, the number of measurements M is sufficient that the bound is approximately reached, allowing the relation above to be replaced by an equality: $\mathbf{K} = \mathbf{F}^{-1}$. This is a convention we will continue to use in the expressions below, remembering that it holds only asymptotically.

The first step toward obtaining the parameter variances is to calculate the Fisher information matrix, which is conveniently just the element-by-element expectation of the Hessian matrix [9], with the expectation taken over the data:

$$F_{ij} = -\mathscr{E}\{H_{ij}\} = -\int \ell(\boldsymbol{\theta}) \left(\frac{\partial^2 \ell}{\partial \theta_i \partial \theta_j} \right) d^M g. \tag{9}$$

$\mathscr{E}\{\cdot\}$ indicates taking the expectation value of the argument, and ℓ is the log-likelihood of the data. Luckily, many of the Hessian matrix elements contain the factor $(g_m - \bar{g}_m)$, the expectation of which (taken over the data vector **g**) is zero. This greatly simplifies the expressions needed to form **F**:

$$\left.\begin{aligned} F_{11} &= \delta_x{}^2 \sum \frac{1}{\sigma_m{}^2} E_m{}^2 \\ F_{12} &= \frac{\delta_x{}^2 A}{w^2} \sum \frac{1}{\sigma_m{}^2} E_m{}^2 \xi_m \\ F_{13} &= \frac{\delta_x{}^2 A}{w^3} \sum \frac{1}{\sigma_m{}^2} E_m{}^2 \xi_m{}^2 \\ F_{22} &= \frac{\delta_x{}^2 A^2}{w^4} \sum \frac{1}{\sigma_m{}^2} E_m{}^2 \xi_m{}^2 \\ F_{23} &= \frac{\delta_x{}^2 A^2}{w^5} \sum \frac{1}{\sigma_m{}^2} E_m{}^2 \xi_m{}^3 \\ F_{33} &= \frac{\delta_x{}^2 A^2}{w^6} \sum \frac{1}{\sigma_m{}^2} E_m{}^2 \xi_m{}^4 \end{aligned}\right\}. \tag{10}$$

The covariance matrix **K** is given by the inverse of the above matrix, easily obtained by Cramer's rule. First calculating the determinant $D = \det(\mathbf{F})$, we can obtain the parameter variances themselves from the diagonal elements of **K**:

$$\mathrm{var}(\hat{A}) = K_{11} = \frac{F_{22} F_{33} - F_{23}{}^2}{D}, \tag{11}$$

$$\mathrm{var}(\hat{\bar{x}}) = K_{22} = \frac{F_{11} F_{33} - F_{13}{}^2}{D}, \tag{12}$$

$$\mathrm{var}(\hat{w}) = K_{33} = \frac{F_{11} F_{22} - F_{12}{}^2}{D}. \tag{13}$$

(Note that we have used the symmetry of the matrix **F** to replace F_{21}, F_{31} and F_{32} with their counterparts F_{12}, F_{13} and F_{23}.) The numerical values of the variance

expressions can be obtained by substituting Eq. (10) into Eqs. (11)–(13).

4. Parameter Variances in the Uniform Noise Case

Unfortunately, in the form given above the variances say little about the general behavior of the estimation and how the parameter variances might depend on system specifications. For this purpose we introduce some approximations which allow the exact variances to be reduced to a simple form. The necessary approximations we will use are:

1. Uniform noise: $\sigma_m = \sigma$ (a good approximation for detector-noise-limited systems).
2. Uniform sampling: the spacing between sample locations x_m is constant.
3. Complete sampling: the data region $x_{\min} - x_{\max}$ is sufficient such that the Gaussian peak $f(x)$ is approximately zero outside of the sampled region. This requirement, together with approximation #2, allows us to write $\sum {E_m}^2 {\xi_m}^p \approx 0$ when p is odd.
4. The profile is well-sampled: since each of the F_{ij} are sums of sampled Gaussian functions, we can approximate the sums with integrals and then replace the integrals with known analytic results.

With these approximations, we can write

$$\sum {E_m}^2 = \frac{1}{\delta_x}\sum e^{-\xi_m^2/w^2}\delta_x$$

$$\approx \frac{1}{\delta_x}\int_{-\infty}^{\infty} e^{-\xi^2/w^2} d\xi = \frac{\sqrt{\pi}w}{\delta_x}.$$

Similarly, we find that

$$\sum {E_m}^2 \xi_m \approx 0, \qquad \sum {E_m}^2 {\xi_m}^2 \approx \frac{\sqrt{\pi}w^3}{2\delta_x},$$

$$\sum {E_m}^2 {\xi_m}^3 \approx 0, \qquad \sum {E_m}^4 {\xi_m}^4 \approx \frac{3\sqrt{\pi}w^5}{4\delta_x},$$

so that $\mathbf{F}$ can be greatly simplified to

$$\mathbf{F} \approx \frac{\delta_x\sqrt{\pi}}{\sigma^2}\begin{pmatrix} w & 0 & \frac{A}{2} \\ 0 & \frac{A^2}{2w} & 0 \\ \frac{A}{2} & 0 & \frac{3A^2}{4w} \end{pmatrix}. \tag{14}$$

The corresponding covariance matrix is

$$\mathbf{K} \approx \frac{\sigma^2}{\sqrt{\pi}\delta_x}\begin{pmatrix} \frac{3}{2w} & 0 & \frac{-1}{A} \\ 0 & \frac{2w}{A^2} & 0 \\ \frac{-1}{A} & 0 & \frac{2w}{A^2} \end{pmatrix}, \tag{15}$$

so that

$$\mathrm{var}(\hat{A}) \approx \frac{3\sigma^2}{2w\delta_x\sqrt{\pi}},$$

$$\mathrm{var}(\hat{\bar{x}}) \approx \mathrm{var}(\hat{w}) \approx \frac{2\sigma^2 w}{A^2\delta_x\sqrt{\pi}}. \tag{16}$$

Some remarkable facts about these results are that the variances of $\hat{\bar{x}}$ and of $\hat{w}$ are equal and that the variance of $\hat{A}$ is independent of A. It is also satisfying to see that the parameter variances (16) have no dependence on the position of the Gaussian, $\bar{x}$. Figure 2 shows comparisons between the empirical parameter variances as determined by a Monte Carlo simulation and the parameter variances as determined by the approximate formulas (16).

We can also see that the variances of the parameters increase as the pixel widths shrink. This is a result of the fact that while decreasing the pixel widths increases the sampling rate (and thus improves the potential resolution of the estimation), it also divides a constant number of photons into smaller bins. These bins have uniform noise, so that by decreasing pixel size we are adding noise faster than we are improving the resolution.

Often we are interested in estimating the area U under the Gaussian curve, given by

$$U = \int_{-\infty}^{\infty} A e^{-(x-\bar{x})2/2w^2} \mathrm{d}x = Aw\sqrt{2\pi}. \tag{17}$$

If $\hat{A}$ and $\hat{w}$ are unbiased estimators of the respective parameters in the Gaussian function, then the unbiased estimator of U is formed by $\hat{U} = \hat{A}\hat{w}\sqrt{2\pi}$. As a linear function of unbiased parameters, the variance of $\hat{U}$ is given by the covariances of $\hat{A}$ and $\hat{w}$ together with the gradient of U with respect to the parameters [10],

$$\mathrm{var}(\hat{U}) = \frac{\partial U}{\partial \boldsymbol{\theta}^{\mathrm{T}}} \mathbf{K}_{\boldsymbol{\theta}} \frac{\partial U}{\partial \boldsymbol{\theta}}, \tag{18}$$

where $\frac{\partial U}{\partial \boldsymbol{\theta}} = \left(\frac{\partial U}{\partial A}, \frac{\partial U}{\partial \bar{x}}, \frac{\partial U}{\partial w}\right)^{\mathrm{T}}$. Therefore,

$$\begin{aligned} \mathrm{var}(\hat{U}) &= \begin{pmatrix} w\sqrt{2\pi} \\ 0 \\ A\sqrt{2\pi} \end{pmatrix}^{\mathrm{T}} \begin{pmatrix} K_{11} & K_{12} & K_{13} \\ K_{21} & K_{22} & K_{23} \\ K_{31} & K_{32} & K_{33} \end{pmatrix} \begin{pmatrix} w\sqrt{2\pi} \\ 0 \\ A\sqrt{2\pi} \end{pmatrix} \\ &= 2\pi(w^2K_{11} + 2Aw K_{13} + A^2K_{33}) \\ &= 3\sqrt{\pi}\sigma^2 w/\delta_x, \end{aligned} \tag{19}$$

where we have used $K_{31} = K_{13}$.

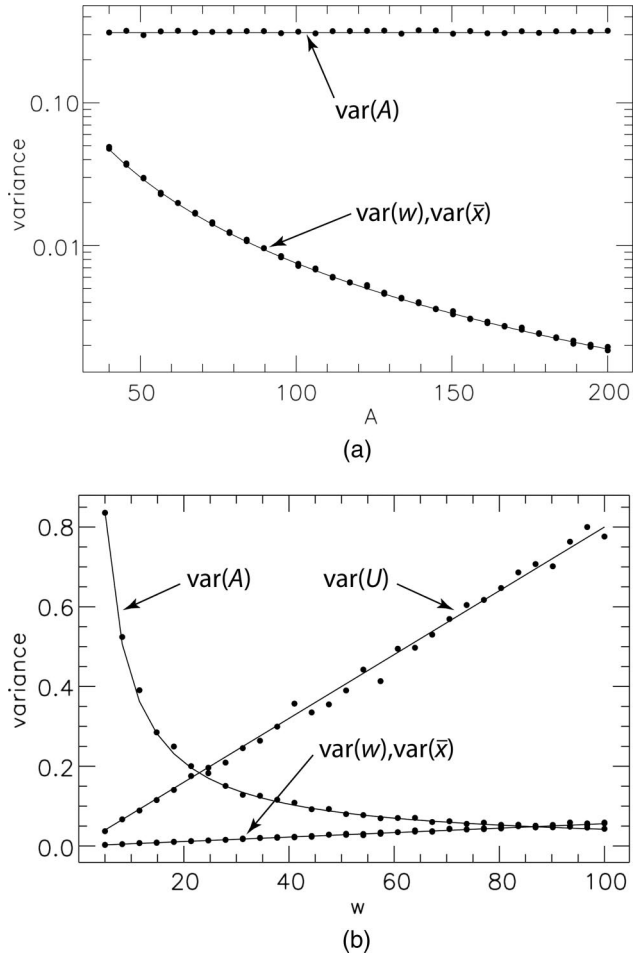


Fig. 2. Comparison between the parameter variances as measured by a Monte Carlo simulation (filled circles) and the approximate parameter variance formulas (16) (curves). The model function used for the Monte Carlo simulation is that of Fig. 1, with one of the model parameters allowed to vary, as shown in the figure abscissa. The data for $\mathrm{var}(\hat{U})$ has been rescaled to fit within the plotting range. (Note that the data here was simulated with δ-sampling, so that the bias discussed in Section 6 below is not present in this simulation.)

5. Parameter Variances in the Poisson-Noise-Dominated Case

If Gaussian statistics are being used for a Poisson-noise-dominated measurement, then it helps to avoid the approximation of uniform noise variance (i.e., ${\sigma_m}^2 \approx \sigma^2$) and instead equate the variance to the integrated value at a given pixel: ${\sigma_m}^2 \approx \delta_x A E_m$. The integrals approximating the sums (10) still have analytic solutions for this case, giving

$$\sum \frac{{E_m}^2}{{\sigma_m}^2} \approx \frac{\sqrt{2\pi}w}{A{\delta_x}^2},$$

$$\sum \frac{{E_m}^2 \xi_m}{{\sigma_m}^2} \approx 0,$$

$$\sum \frac{{E_m}^2 {\xi_m}^2}{{\sigma_m}^2} \approx \frac{\sqrt{2\pi}w^3}{A{\delta_x}^2},$$

$$\sum \frac{{E_m}^2 {\xi_m}^3}{{\sigma_m}^2} \approx 0,$$

$$\sum \frac{{E_m}^2 {\xi_m}^4}{{\sigma_m}^2} \approx \frac{3\sqrt{2\pi}w^5}{A{\delta_x}^2},$$

so that

$$\mathbf{F} \approx \sqrt{2\pi}\begin{pmatrix} \frac{w}{A} & 0 & 1 \\ 0 & \frac{A}{w} & 0 \\ 1 & 0 & \frac{3A}{w} \end{pmatrix}, \tag{20}$$

$$\mathbf{K} \approx \frac{1}{\sqrt{2\pi}}\begin{pmatrix} \frac{3A}{2w} & 0 & -\frac{1}{2} \\ 0 & \frac{w}{A} & 0 \\ -\frac{1}{2} & 0 & \frac{w}{2A} \end{pmatrix}, \tag{21}$$

$$\mathrm{var}(\hat{U}) \approx \sqrt{2\pi}Aw. \tag{22}$$

Note that in the Poisson-noise approximation $\mathrm{var}(\hat{\bar{x}}) \neq \mathrm{var}(\hat{w}) \neq \mathrm{var}(\hat{A})$ is a function of parameter A. Unlike the uniform-noise case, we find that the parameter variances are now independent of the pixel size δ_x. This reflects the fact that, in Poisson noise, as the pixel sizes shrink, the increased sampling rate is exactly offset by a decrease in both signal and noise in each pixel.

6. Bias in the Standard Model

Gaussian profile estimation is a classic problem, yet to our knowledge there has been no mention of bias in the model. The estimation process outlined so far in this paper minimizes the error between a discrete data set **g** and a discrete object model **f** even though the object is known to be continuous. Ignoring this difference is the source of the bias.

The approach up to this point has been to model the measured data **g** as coming from a system operator $\mathcal{H}$ which performs δ-sampling:

$$\bar{g}_m = \delta_x \int_{-\infty}^{\infty} f(x)\delta(x - x_m)\mathrm{d}x.$$

A more realistic model for an array detector, however, accounts for the physical dimension of the pixel over which the signal is integrated:

$$\bar{g}_m = \int_{-\infty}^{\infty} f(x)\,\mathrm{rect}\left(\frac{x - x_m}{\delta_x}\right)\mathrm{d}x, \tag{23}$$

where the rectangle function is defined as

$$\mathrm{rect}(x/L) = \begin{cases} 1 : |x| < L/2 \\ 0 : |x| > L/2 \end{cases}. \tag{24}$$

For pixel responses modeled as rect-functions (as shown in Fig. 3), the pixel value coincides with δ-sampling only if the continuous function being integrated is linear across the dimensions of each pixel. If the object contains quadratic or higher terms, rect-sampling and δ-sampling no longer coincide. The difference is what we refer to here loosely as the bias, $b(x)$. Ideally, we would like to know the parameter biases directly, and perform a correction by

$$\hat{\boldsymbol{\theta}}_{\mathrm{unbiased}} = \hat{\boldsymbol{\theta}}_{\mathrm{biased}} - \hat{\boldsymbol{\theta}}_{\mathrm{correction}}.$$

Unfortunately, the parameter correction terms appear to be unavailable in analytic form. Instead, we can provide a correction directly to $\hat{f}(x)$ in post-processing,

$$\hat{f}_{\mathrm{unbiased}}(x) = \hat{f}_{\mathrm{biased}}(x) - b(x). \tag{25}$$

The choice of using rect-functions to model the pixel responses is not compulsory. Gaussian functions are also a common choice, and will produce a different estimate of the bias. Without knowing the precise form of the pixel response function, however, the bias can only be approximated. Rect-functions are used here to illustrate the principle.

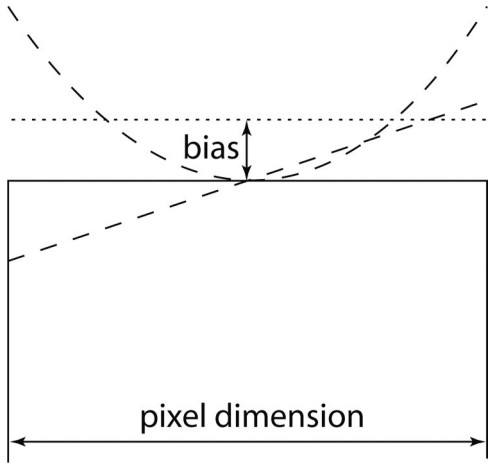


Fig. 3. Illustration of constant, linear, and quadratic functions across an idealized pixel. A δ-sampling measurement model selects the function value at the midpoint of the pixel whereas a rect-model selects the mean of the function across the pixel as the result. (The dotted line represents the mean of the parabolic function.)

The discussion of polynomial functions suggests using a Taylor expansion of the object $f(x)$ across the dimensions of each pixel, providing an explicit formula for the bias component. The Taylor series expansion of $f(x)$ at location x_m (the midpoint of pixel m) is given by

$$f(x) = \underbrace{[f(x)]_{x=x_m} + (x - x_m)[f'(x)]_{x=x_m}}_{\text{model}} + \underbrace{\frac{(x - x_m)^2}{2}[f'(x)]_{x=x_m} + \cdots}_{\text{bias error term, } b\langle x\rangle} \tag{26}$$

While it is possible to calculate the Taylor series terms to any desired order, the dominant error term is the leading (parabolic) term of the series. If we approximate the bias component using only this parabolic term, we need only calculate its mean value across the pixel dimensions to give $b(x)$. Using the second derivative of a Gaussian function,

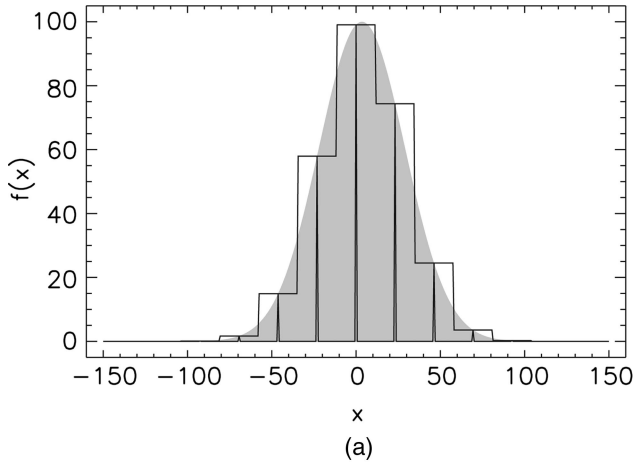
$$f'(x) = \frac{A}{w^2}\left[\frac{(x - \bar{x})^2}{w^2} - 1\right]e^{-(x-\bar{x})^2/2w^2}, \tag{27}$$

the mean of the parabolic term at pixel m gives the bias as

$$\begin{aligned} b(x_m) &\approx f'(x_m)\frac{1}{2\delta_x}\int_{x_m-\frac{1}{2}\delta_x}^{x_m+\frac{1}{2}\delta_x}(x - x_m)^2\mathrm{d}x \\ &= \frac{A\delta_x{}^2}{24w^2}\left[\frac{(x_m - \bar{x})^2}{w^2} - 1\right]e^{-(x_m-\bar{x})^2/2w^2}. \end{aligned} \tag{28}$$

The structure of this result appears reasonable: the bias increases with A and inversely with w, which both have the effect of narrowing the Gaussian. The bias also increases quadratically with the pixel width δ_x. Figure 4(b) gives an illustration of the bias curve for the Gaussian object of Fig. 1 but now sampled by only 13 pixels, so that $\delta_x \approx w$. For the example shown, the bias in $\hat{A}$ is more than 3% and the bias in $\hat{w}$ is 4.7%, whereas the $\hat{\bar{x}}$ estimator is unbiased in the quadratic approximation. Comparing this to the standard deviations of $\hat{A}$ and $\hat{w}$ shows that the bias term is important when the sampling becomes this small ($\delta_x \lesssim w$).

In order to use bias-corrected parameter estimates, we can first perform a biased estimation step. If we assume that the estimate produces a good approximation to $f(x)$ then we can calculate $b(x)$ and perform the estimation once more using "corrected" data, $\mathbf{g}_{\mathrm{corr}} = \mathbf{g} - \mathbf{b}$, where $\mathbf{b}$ is the δ-sampled bias. The usefulness of performing bias-correction is probably limited, however. Most data used for Gaussian-profile estimation is only approximately Gaussian, so that in most cases the modeling error is more likely to be larger than the bias discussed here.



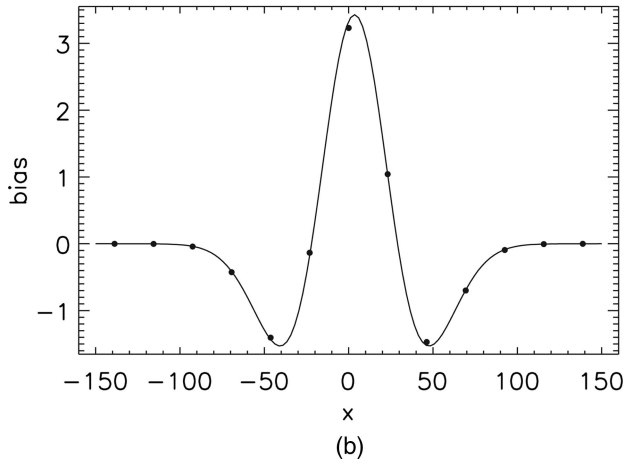


Fig. 4. (a) Illustration of the three functions involved in the bias discussion: (1) the underlying continuous function $f(x)$ (grey fill), rescaled by $1/\delta_x$ to match the dimensions of $\mathbf{g}$; (2) the piecewise continuous pixel model (stairstep), and (3) the discrete δ-sampled model (spikes). (b) The bias in $\hat{f}(x)$ resulting from using data measured from an array detector while estimating with a δ-sampled model. The curve is calculated from Eq. (28); the filled circles represent data obtained by measuring the bias directly in a simulation. The model $f(x)$ used for both (a) and (b) is that of Fig. 1 but measured with 13 pixels rather than 101.

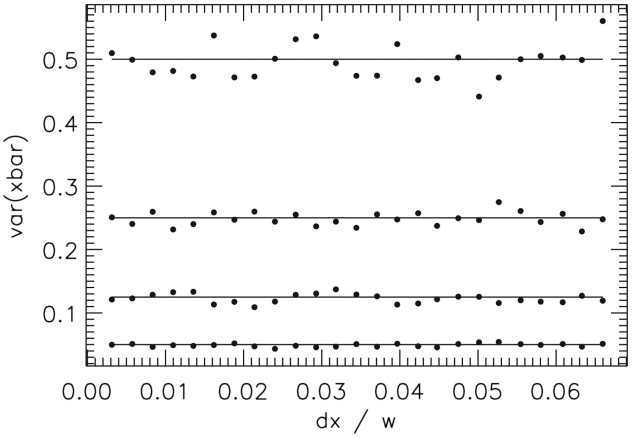


Fig. 5. Comparison of $\mathrm{var}(\hat{\hat{x}})$ determined from Eq. (7) with a Monte Carlo simulation of a Poisson-noise-dominated measurement. In descending order, from top to bottom, the lines and data points correspond to $U = 20\,000$, $10\,000$, 5000, and 2000 (in units of mean number of photoelectrons).

7. Conclusions

Two previous publications [1,4] which use MLE for 1D Gaussian profile estimation state an important result that there is an optimal choice for magnification of the Gaussian profile prior to sampling by a given detector array. If the optical system is modified such that the Gaussian width w is approximately $1 \sim 2$ times the pixel spacing, then Refs. 2, 5 both conclude that the variance of the position estimate $\hat{\hat{x}}$ will be minimized.

For the above situation, where the detector is unmodified while the optics are allowed to change magnification, A and w are not free parameters but are constrained to enforce a constant energy U via the relation $U = Aw\sqrt{2\pi}$. Plugging this into the equation for $\mathrm{var}(\hat{\hat{x}})$ in Eq. (16) for uniform-noise and Eq. (21) for Poisson noise gives

$$\text{Uniform:}\quad \mathrm{var}(\hat{\hat{x}}) = \frac{\sqrt{2}U\sigma^2}{\pi A^3 \delta_x},$$

$$\text{Poisson:}\quad \mathrm{var}(\hat{\hat{x}}) = \frac{U}{2\pi A^2}.$$

Combining these results with the model of the bias, we find that we can first decrease the pixel size δ_x to render the bias negligible, after which further decreasing the pixel size increases the variance in the uniform-noise case, or has no effect on the variance in the Poisson-noise case. (Figure 5 shows the simulation results of varying δ_x and measuring $\mathrm{var}(\hat{U})$ under the case of Poisson noise.) All other things being equal, it is usually desirable to allow the pixel widths to be as large as possible, which implies that $w \lesssim \delta_x \lesssim 2w$ is a reasonable choice, even for the Poisson-noise case. For this pixel size, the bias is kept small, but not so small as to be wasteful. Thus, our result is in qualitative agreement with that of Ireland [1] and Winick [4], but is in disagreement about the asymptotic behavior of the estimator as the pixel width is decreased in the presence of pure Poisson noise.

A reason why we find it implausible, for the case of Poisson noise, that there should be an increase in estimator variance with finer sampling is the following argument. If we compare two systems, one of which samples the Gaussian profile at twice the rate of the other (i.e., contains twice as many pixels, each of which are half the width of the other system), then we know that we can always bin the data from the finely-sampled system, by adding pairs of pixels together, to replicate data from the coarsely-sampled system. If the data noise is purely driven by Poisson statistics, then the coarsely-sampled data and the binned finely-sampled data will have identical statistics. A properly-designed estimation algorithm should be able to take advantage of this, and thus,

there should not be a penalty for increasing the sampling rate under Poisson noise.

The authors have posted a set of programs written in the IDL [11] language located at the url http://www.ittvis.com/codebank/index.asp (the file is gauss_mle1.pro under the heading Statistics) which perform the estimation algorithm discussed in this paper. Interested readers are encouraged to download, distribute, and use the code freely.

Appendix A. Extension to a Model with Nonzero Background

If the Gaussian profile is placed on a quadratic background, then we can model the measured profile (shown in Fig. 6) as

$$f(x) = \delta_x[b_0 + b_1 x + b_2 x^2 + Ae^{-(x-\bar{x})^2/2w^2}].$$

The gradient of the log likelihood equation becomes

$$[\nabla \ell]_1 = \frac{\partial \ell}{\partial b_0} = \delta_x \sum \gamma_m,$$

$$[\nabla \ell]_2 = \frac{\partial \ell}{\partial b_1} = \delta_x \sum x_m \gamma_m,$$

$$[\nabla \ell]_3 = \frac{\partial \ell}{\partial b_2} = \delta_x \sum {x_m}^2 \gamma_m,$$

$$[\nabla \ell]_4 = \frac{\partial \ell}{\partial A} = \delta_x \sum E_m \gamma_m,$$

$$[\nabla \ell]_5 = \frac{\partial \ell}{\partial \bar{x}} = \frac{\delta_x A}{w^2} \sum E_m \xi_m \gamma_m,$$

$$[\nabla \ell]_6 = \frac{\partial \ell}{\partial w} = \frac{\delta_x A}{w^3} \sum E_m {\xi_m}^2 \gamma_m,$$

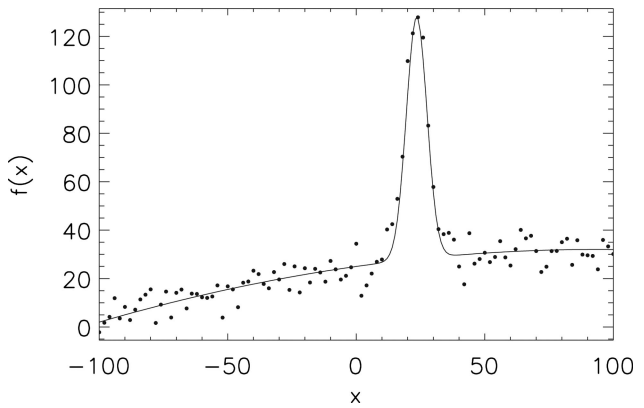


Fig. 6. Example data for a 1D Gaussian profile with a quadratic background, generated from the signal ($b_0 = 25.0$, $b_1 = 0.15$, $b_2 = -0.0008$, $A = 100.0$, $\bar{x} = 23.5$, $w = 4.0$, $\sigma = 5.0$, $\delta_x = 1$, and $M = 101$). The solid line represents the object $f(x)$ (true underlying Gaussian profile), the filled circles represent the measured data values, $\mathbf{g}$.

where, again, $\xi_m = (x_m - \bar{x})$, $E_m = e^{-\xi_m/2w^2}$, and $\gamma_m = \frac{g_m - \bar{g}_m}{{\sigma_m}^2}$. The Hessian matrix becomes

$$\mathbf{H} = \begin{pmatrix} \frac{\partial^2 \ell}{\partial {b_0}^2} & \frac{\partial^2 \ell}{\partial b_1 \partial b_0} & \frac{\partial^2 \ell}{\partial b_2 \partial b_0} & \frac{\partial^2 \ell}{\partial A \partial b_0} & \frac{\partial^2 \ell}{\partial \bar{x} \partial b_0} & \frac{\partial^2 \ell}{\partial w \partial b_0} \\ \cdot & \frac{\partial^2 \ell}{\partial {b_1}^2} & \frac{\partial^2 \ell}{\partial b_2 \partial b_1} & \frac{\partial^2 \ell}{\partial A \partial b_1} & \frac{\partial^2 \ell}{\partial \bar{x} \partial b_1} & \frac{\partial^2 \ell}{\partial w \partial b_1} \\ \cdot & \cdot & \frac{\partial^2 \ell}{\partial {b_2}^2} & \frac{\partial^2 \ell}{\partial A \partial b_2} & \frac{\partial^2 \ell}{\partial \bar{x} \partial b_2} & \frac{\partial^2 \ell}{\partial w \partial b_2} \\ \cdot & \cdot & \cdot & \frac{\partial^2 \ell}{\partial A^2} & \frac{\partial^2 \ell}{\partial \bar{x} \partial A} & \frac{\partial^2 \ell}{\partial w \partial A} \\ \cdot & \cdot & \cdot & \cdot & \frac{\partial^2 \ell}{\partial \bar{x}^2} & \frac{\partial^2 \ell}{\partial w \partial \bar{x}} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \frac{\partial^2 \ell}{\partial w^2} \end{pmatrix}$$

with elements

$$H_{11} = -{\delta_x}^2 \sum S_m,$$

$$H_{12} = -{\delta_x}^2 \sum S_m x_m,$$

$$H_{13} = -{\delta_x}^2 \sum S_m {x_m}^2,$$

$$H_{14} = -{\delta_x}^2 \sum S_m E_m,$$

$$H_{15} = -\frac{{\delta_x}^2 A}{w^2} \sum S_m E_m \xi_m,$$

$$H_{16} = -\frac{{\delta_x}^2 A}{w^3} \sum S_m E_m {\xi_m}^2,$$

$$H_{22} = -{\delta_x}^2 \sum S_m {x_m}^2,$$

$$H_{23} = -{\delta_x}^2 \sum S_m {x_m}^3,$$

$$H_{24} = -{\delta_x}^2 \sum S_m E_m x_m,$$

$$H_{25} = -\frac{{\delta_x}^2 A}{w^2} \sum S_m E_m x_m \xi_m,$$

$$H_{26} = -\frac{{\delta_x}^2 A}{w^3} \sum S_m E_m x_m {\xi_m}^2,$$

$$H_{33} = -{\delta_x}^2 \sum S_m {x_m}^4,$$

$$H_{34} = -{\delta_x}^2 \sum S_m E_m {x_m}^2,$$

$$H_{35} = -\frac{{\delta_x}^2 A}{w^2} \sum S_m E_m {x_m}^2 \xi_m,$$

$$H_{36} = -\frac{\delta_x{}^2 A}{w^3}\sum S_m E_m x_m{}^2 \xi_m{}^2,$$

$$H_{44} = -\delta_x{}^2 \sum S_m E_m{}^2,$$

$$H_{45} = \frac{\delta_x}{w^2}\sum E_m \xi_m \Gamma_m,$$

$$H_{46} = \frac{\delta_x}{w^3}\sum E_m \xi_m{}^2 \Gamma_m,$$

$$H_{55} = \frac{\delta_x A}{w^4}\sum E_m\left[\xi_m{}^2 \Gamma_m - w^2 \gamma_m\right],$$

$$H_{56} = \frac{\delta_x A}{w^5}\sum E_m\left[\xi_m{}^3 \Gamma_m - 2\xi_m w^2 \gamma_m\right],$$

$$H_{66} = \frac{\delta_x A}{w^6}\sum E_m\left[\xi_m{}^4 \Gamma_m - 3\xi_m{}^2 w^2 \gamma_m\right],$$

where $S_m = \frac{1}{\sigma_m{}^2}$ and $\Gamma_m = \frac{g_m - 2\bar{g}_m}{\sigma_m{}^2}$.

Since the first three rows of the Hessian matrix elements have no dependence on the measurements $\mathbf{g}$, the values of the Fisher information matrix $\mathbf{F}$ and $\mathbf{H}$ are the same for those elements. (The ":" notation below indicates all elements in the row.)

$$F_{1},: = -H_{1},:$$

$$F_{2},: = -H_{2},:$$

$$F_{3},: = -H_{3},:$$

$$F_{44} = -H_{44}$$

$$F_{45} = -\mathcal{E}\{H_{45}\} = \frac{\delta_x{}^2 A}{w^2}\sum S_m E_m{}^2 \xi_m,$$

$$F_{46} = -\mathcal{E}\{H_{46}\} = \frac{\delta_x{}^2 A}{w^3}\sum S_m E_m{}^2 \xi_m{}^2,$$

$$F_{55} = -\mathcal{E}\{H_{55}\} = \frac{\delta_x{}^2 A^2}{w^4}\sum S_m E_m{}^2 \xi_m{}^2,$$

$$F_{56} = -\mathcal{E}\{H_{56}\} = \frac{\delta_x{}^2 A^2}{w^5}\sum S_m E_m{}^2 \xi_m{}^3,$$

$$F_{66} = -\mathcal{E}\{H_{66}\} = \frac{\delta_x{}^2 A^2}{w^6}\sum S_m E_m{}^2 \xi_m{}^4.$$

From here it is possible to obtain $\mathbf{K} = \mathbf{F}^{-1}$ analytically, and to use approximations to simplify the resulting expressions, but the product of the effort is a thicket of lengthy equations. The parameter variances for the zero background case were shown in (16) to have no dependence on the position of the Gaussian, $\bar{x}$. While this remains true when we add a constant background term (b_0) to the model, it is not true if the background is allowed to be a linear or quadratic function of the abscissa. The expressions involve a frustrating dependence on $\bar{x}$, $x_{\min}$, and $x_{\max}$ that greatly complicate their behavior. Thus, in the nonzero background case, it is probably best to obtain $\mathbf{K}$ from $\mathbf{F}$ numerically and select the resulting diagonal elements as the parameter variances.

This work was supported in part by DOD contract DAAE07-02-C-L011. The authors thank Peter Jansson and William Dallas of the University of Arizona for helpful discussions.

References

1. J. Ireland, "Precision limits to emission-line profile measuring experiments," Astrophys. J. **620,** 1132–1139 (2005).
2. D. A. Landman, R. Roussel-Dupré, and G. Tanigawa, "On the statistical uncertainties associated with line profile fitting," Astrophys. J. **261,** 732–735 (1982).
3. D. D. Lenz and T. R. Ayres, "Errors associated with fitting Gaussian profiles to noisy emission-line spectra," Publ. Astron. Soc. Pac. **104,** 1104–1106 (1992).
4. K. A. Winick, "Cramér-Rao lower bounds on the performance of charge-coupled-device optical position estimators," J. Opt. Soc. Am. A **3,** 1809–1815 (1986).
5. N. Bobroff, "Position measurement with a resolution and noise-limited instrument," Rev. Sci. Instrum. **57,** 1152–1157 (1986).
6. B. E. A. Saleh, "Estimation of the location of an optical object with photodetectors limited by quantum noise," Appl. Opt. **13,** 1824–1827 (1974).
7. H. H. Barrett and K. Myers, *Foundations of Image Science* (Wiley, 2004).
8. J. Dennis, J. E. and R. B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations* (SIAM), 1996).
9. H. L. Van Trees, *Detection, Estimation and Modulation Theory, Part 1: Detection, Estimation, and Linear Modulation Theory* (Wiley, 1968).
10. M. G. Kendall and A. Stuart, *The Advanced Theory of Statistics,* 3rd ed. (Charles Griffin, 1969), Vol. 2.
11. The Interactive Data Language (IDL) software is developed by ITT Visual Information Systems, http://www.ittvis.com/index.asp.