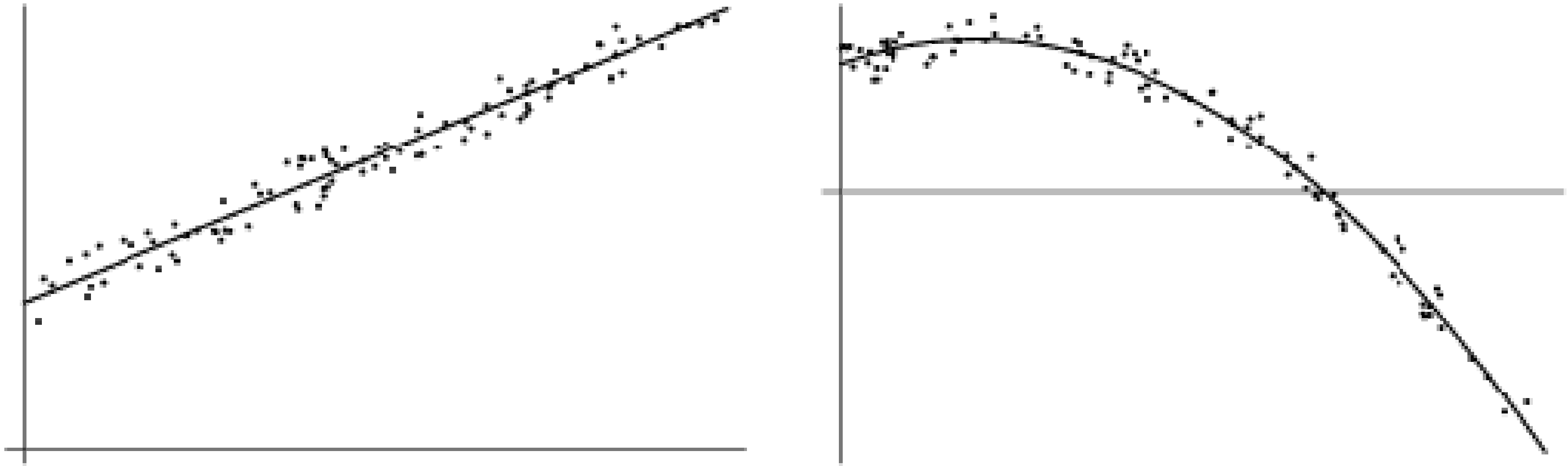
Some things they don't tell you about least squares fitting



"A mathematical procedure for finding the best-fitting curve to a given set of points by minimizing the sum of the squares of the offsets ("the residuals") of the points from the curve."
Mathworld

Luis Valcárcel, **McGill University**

October 19, 2005

HEP Graduate Student Meeting

Overview

- Linear Least Squares Fitting (review)
- Non Linear Least Squares Fitting
- Why do we minimize the chi-square?
  - Connection with Maximum Likelihood principle
  - Vertical vs Perpendicular offsets
  - Robust estimation
- What about errors in the inputs?
  - Weighting errors in y
  - What to do with the errors in x?
- What about errors in the outputs?
  - How to calculate them?
  - How to interpret them?
- Program comparisons

Linear Least Squares Fitting (review)

Line Fitting

$$R^2(a, b) \equiv \sum_{i=1}^{n} [y_i - (a + b\, x_i)]^2$$

$$\frac{\partial (R^2)}{\partial a} = -2 \sum_{i=1}^{n} [y_i - (a + b\, x_i)] = 0$$

$$\frac{\partial (R^2)}{\partial b} = -2 \sum_{i=1}^{n} [y_i - (a + b\, x_i)]\, x_i = 0.$$

$$\begin{bmatrix} n & \sum_{i=1}^{n} x_i \\ \sum_{i=1}^{n} x_i & \sum_{i=1}^{n} x_i^2 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{n} y_i \\ \sum_{i=1}^{n} x_i\, y_i \end{bmatrix},$$

$$a = \frac{\overline{y}\,(\sum_{i=1}^{n} x_i^2) - \overline{x}\, \sum_{i=1}^{n} x_i\, y_i}{\sum_{i=1}^{n} x_i^2 - n\, \overline{x}^2}$$

$$b = \frac{(\sum_{i=1}^{n} x_i\, y_i) - n\, \overline{x}\, \overline{y}}{\sum_{i=1}^{n} x_i^2 - n\, \overline{x}^2}$$

- Exact solution
- Implemented in scientific calculators
- Can even easily get the errors on the parameters

Polynomial Fitting

$$R^2 \equiv \sum_{i=1}^{n} [y_i - (a_0 + a_1 x_i + \ldots + a_k x_i^k)]^2.$$

$$\frac{\partial (R^2)}{\partial a_k} = -2 \sum_{i=1}^{n} [y - (a_0 + a_1 x + \ldots + a_k x^k)] x^k = 0.$$

$$\begin{bmatrix} n & \sum_{i=1}^{n} x_i & \cdots & \sum_{i=1}^{n} x_i^k \\ \sum_{i=1}^{n} x_i & \sum_{i=1}^{n} x_i^2 & \cdots & \sum_{i=1}^{n} x_i^{k+1} \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{i=1}^{n} x_i^k & \sum_{i=1}^{n} x_i^{k+1} & \cdots & \sum_{i=1}^{n} x_i^{2k} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_k \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{n} y_i \\ \sum_{i=1}^{n} x_i y_i \\ \vdots \\ \sum_{i=1}^{n} x_i^k y \end{bmatrix}.$$

- Really just a generalization of the previous case
- Exact solution
- Just big matrices

General Linear Fitting

$$y(x) = \sum_{k=1}^{M} a_k X_k(x)$$

$X_1(x), \ldots, X_M(x)$ are arbitrary fixed functions of x (can be nonlinear), called the *basis functions*

$$\chi^2 = \sum_{i=1}^{N} \left[\frac{y_i - \sum_{k=1}^{M} a_k X_k(x_i)}{\sigma_i} \right]^2$$

$$0 = \sum_{i=1}^{N} \frac{1}{\sigma_i^2} \left[y_i - \sum_{j=1}^{M} a_j X_j(x_i) \right] X_k(x_i) \qquad k = 1, \ldots, M$$

normal equations of the least squares problem

Can be put in matrix form and solved

Exponential Fitting

$$y = A\,e^{B\,x},$$

$$\ln y = \ln A + B\,x.$$ Linearize the equation and apply the fit to a straig

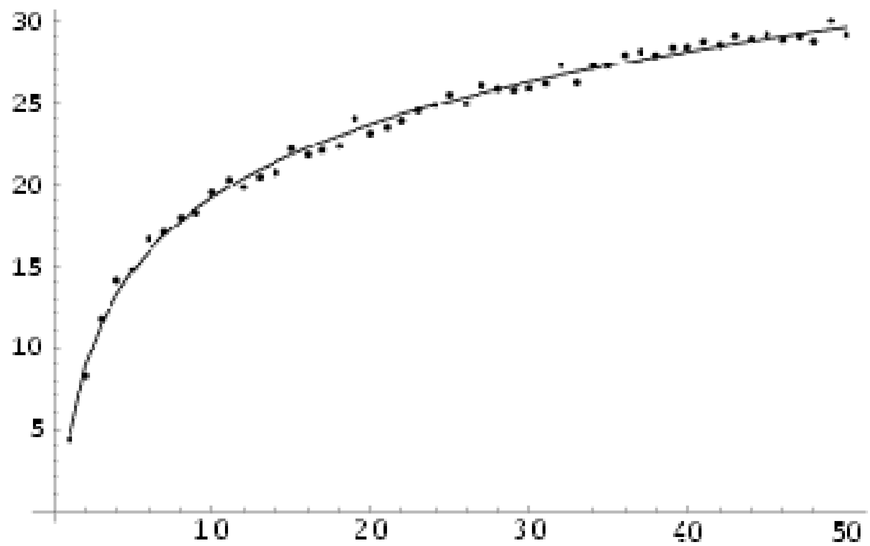
$$A \equiv \exp(a)$$

$$B \equiv b$$

$$a = \frac{\sum_{i=1}^{n} \ln y_i \sum_{i=1}^{n} x_i^2 - \sum_{i=1}^{n} x_i \sum_{i=1}^{n} x_i \ln y_i}{n \sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i\right)^2}$$

$$b = \frac{n \sum_{i=1}^{n} x_i \ln y_i - \sum_{i=1}^{n} x_i \sum \ln y_i}{n \sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i\right)^2},$$

Logarithmic Fitting



$$y = a + b \ln x,$$

$$b = \frac{n \sum_{i=1}^{n} (y_i \ln x_i) - \sum_{i=1}^{n} y_i \sum_{i=1}^{n} \ln x_i}{n \sum_{i=1}^{n} (\ln x_i)^2 - \left(\sum_{i=1}^{n} \ln x_i\right)^2}$$

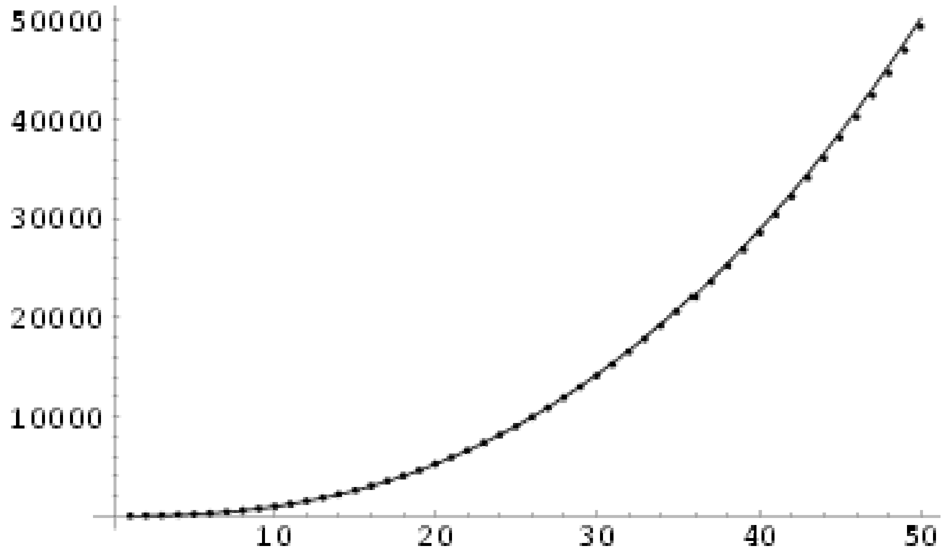
$$a = \frac{\sum_{i=1}^{n} y_i - b \sum_{i=1}^{n} (\ln x_i)}{n}.$$

Power Law Fitting

$$y = A\,x^B,$$

$$A \equiv e^a$$

$$B \equiv b$$



$$b = \frac{n \sum_{i=1}^{n} (\ln x_i \ln y_i) - \sum_{i=1}^{n} (\ln x_i) \sum_{i=1}^{n} (\ln y_i)}{n \sum_{i=1}^{n} (\ln x_i)^2 - (\sum_{i=1}^{n} \ln x_i)^2}$$

$$a = \frac{\sum_{i=1}^{n} (\ln y_i) - b \sum_{i=1}^{n} (\ln x_i)}{n},$$

Summary of Linear least squares fitting

- "The *linear* least squares fitting technique is the simplest and most commonly applied form of linear regression and provides a solution to the problem of finding the best fitting *straight* line through a set of points. In fact, if the functional relationship between the two quantities being graphed is known to within additive or multiplicative constants, it is common practice to transform the data in such a way that the resulting line *is* a straight line. For this reason, standard forms for exponential, logarithmic, and power laws are often explicitly computed. The formulas for linear least squares fitting were independently derived by Gauss and Legendre." Mathworld

Non Linear Least Squares Fitting

Non linear fitting

- "For nonlinear least squares fitting to a number of unknown parameters, linear least squares fitting may be applied iteratively to a linearized form of the function until convergence is achieved. However, it is often also possible to linearize a nonlinear function at the outset and still use linear methods for determining fit parameters without resorting to iterative procedures. This approach does commonly violate the implicit assumption that the distribution of errors is normal, but often still gives acceptable results using normal equations, a pseudoinverse, etc. Depending on the type of fit and initial parameters chosen, the nonlinear fit may have good or poor convergence properties." Mathworld.
- "We use the same approach as in previous sections, namely to define a $\chi 2$ merit function and determine best-fit parameters by its minimization. With nonlinear dependences, however, the minimization must proceed iteratively. Given trial values for the parameters, we develop a procedure that improves the trial solution. The procedure is then repeated until $\chi 2$ stops (or effectively stops) decreasing." Numerical Recipes

$$\chi^2 = \sum_i (y_i - y(x_i))^2$$

$$\frac{\partial \chi^2}{\partial a_j} = -2\sum_i \left\{ (y_i - y(x_i)) \frac{\partial y(x_i)^2}{\partial a_j} \right\} = 0$$

- Treat chi-squared as a continuous fct of the m parameters and search the m-dimensional space for the appropriate minimum value of chi-squared
- Apply to the m equations approximation methods for finding roots of coupled, nonlinear equations
- Use a combination of both methods

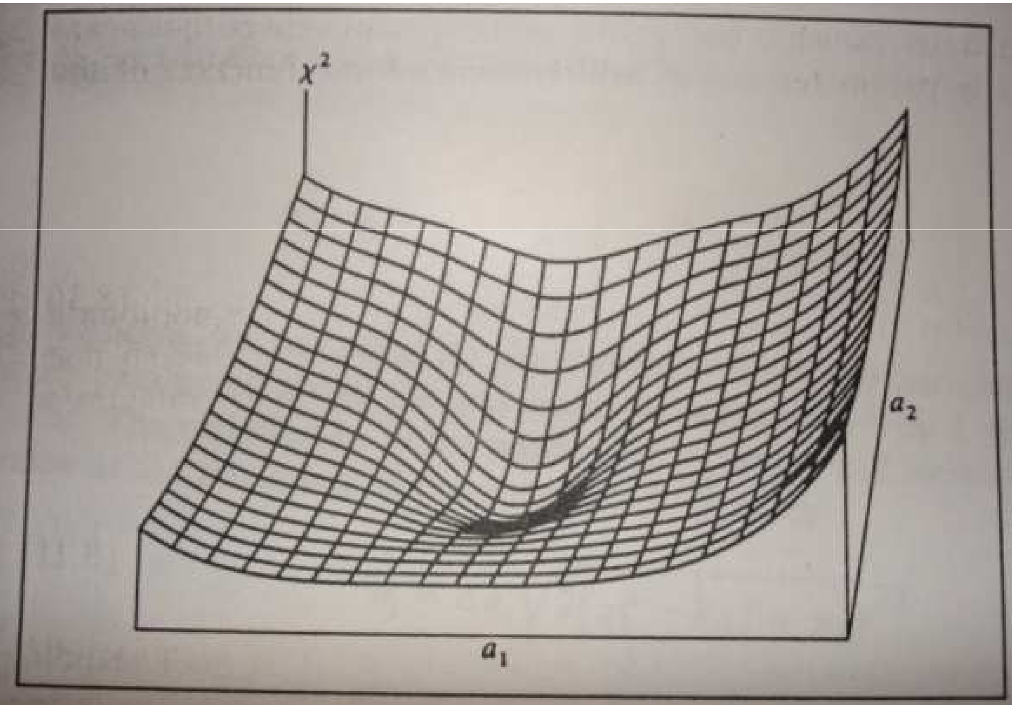


FIGURE 8.2
Chi-square hypersurface as a function of two parameters.

•Grid Search: Vary each parameter in turn, minimizing chi-squared wrt each parameter independently. Many successive iterations are required to locate the minimum of chi-squared unless the parameters are independent.

•Gradient Search: Vary all parameters simultaneously, adjusting relative magnitudes of the variations so that the direction of propagation in parameter space is along the direction of steepest descent of chi-squared

•Expansion Methods: Find an approximate analytical function that describes the chi-squared hypersurface and use this function to locate the minimum, with methods developed for linear least-squares fitting. Number of computed points is less, but computations are considerably more complicated.

•Marquardt Method: Gradient-Expansion combination

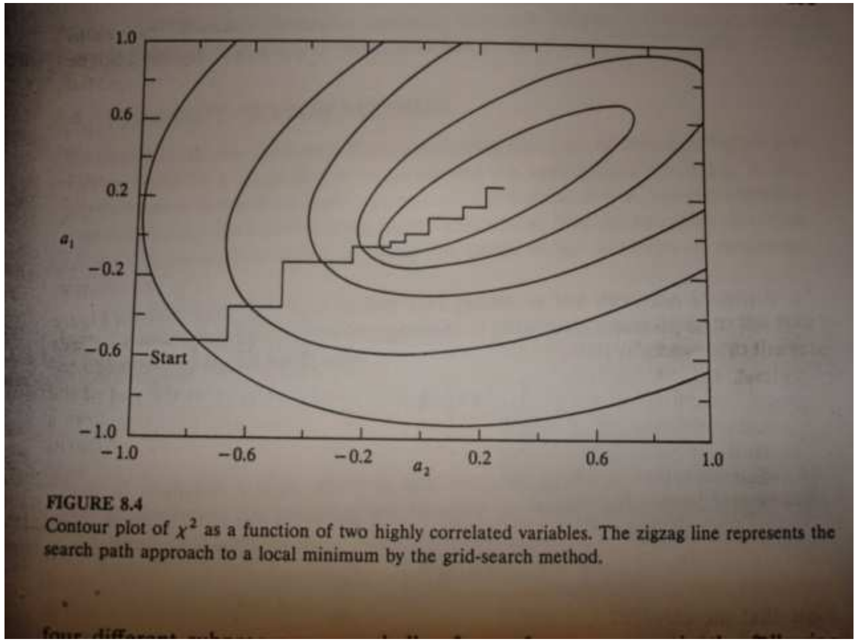


FIGURE 8.4
Contour plot of χ^2 as a function of two highly correlated variables. The zigzag line represents the search path approach to a local minimum by the grid-search method.

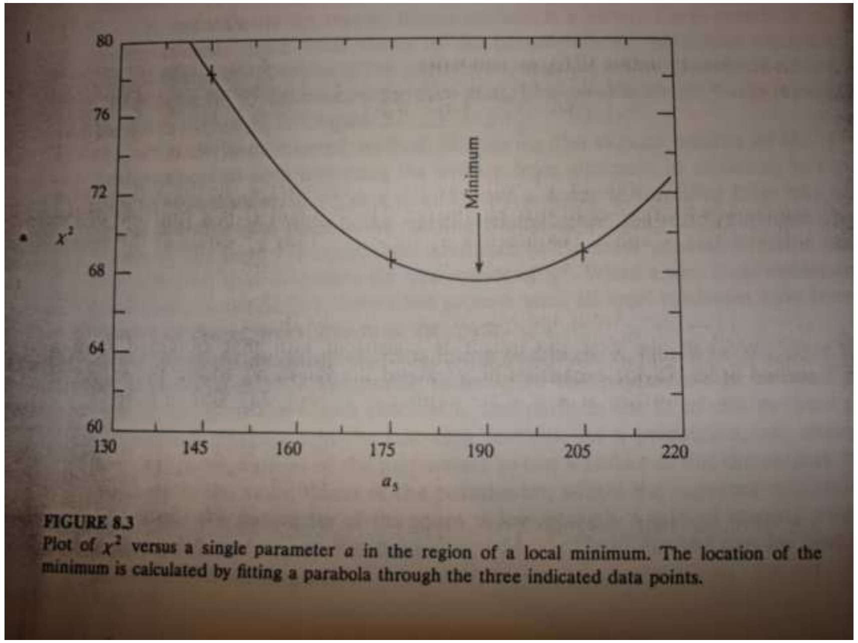


FIGURE 8.3
Plot of χ^2 versus a single parameter a in the region of a local minimum. The location of the minimum is calculated by fitting a parabola through the three indicated data points.

From Bevington and Robinson

MINUIT

- “What Minuit is intended to do.

  Minuit is conceived as a tool to find the minimum value of a multi-parameter function and analyze the shape of the function around the minimum. The principal application is foreseen for statistical analysis, working on chisquare or log-likelihood functions, to compute the best-fit parameter values and uncertainties, including correlations between the parameters. It is especially suited to handle difficult problems, including those which may require guidance in order to find the correct solution.

- What Minuit is not intended to do

  Although Minuit will of course solve easy problems faster than complicated ones, it is not intended for the repeated solution of identically parametrized problems (such as track fitting in a detector) where a specialized program will in general be much more efficient. ”,

MINUIT documentation

- Careful with error estimation using MINUIT: Read their documentation.
- Also see “How to perform a linear fit” in ROOT documentation

Why do we minimize the chi-square?

Other minimization schemes

- Merit function:= fct that measures the agreement between data and the fitting model for a particular choice of the parameters. By convention, this is small when agreement is good.

- MinMax problem: $\max_i \{|y_i - (ax_i + b)|\}$
  Requires advanced techniques

- Absolute deviation: $\sum_i |y_i - (ax_i + b)|$
  absolute value fct not differentiable at zero! "although the *unsquared* sum of distances might seem a more appropriate quantity to minimize, use of the absolute value results in discontinuous derivatives which cannot be treated analytically. " Mathworld

- Least squares: $\sum_i [y_i - (ax_i + b)]^2$
  Most convenient. "This allows the merit fct to be treated as a continuous differentiable quantity. However, because squares of the offsets are used, outlying points can have a disproportionate effect on the fit, a property which may or may not be desirable depending on the problem at hand." Mathworld.

- Least median squares, Maple

Connection with Maximum Likelihood principle

•“*Given a particular set of parameters*, what is the probability that this data set could have occurred?”
•Intuition tells us that the data set should not be too improbable for the correct choice of parameters.
•Identify the probability of the data given the parameters (which is a mathematically computable number), as the *likelihood* of the parameters given the data.
•Find those values that *maximize* the likelihood
•least-squares fitting *is* a maximum likelihood estimation of the fitted parameters *if* the measurement errors are independent and normally distributed with constant standard deviation

probability of the data set is the product of the probabilities of each point,

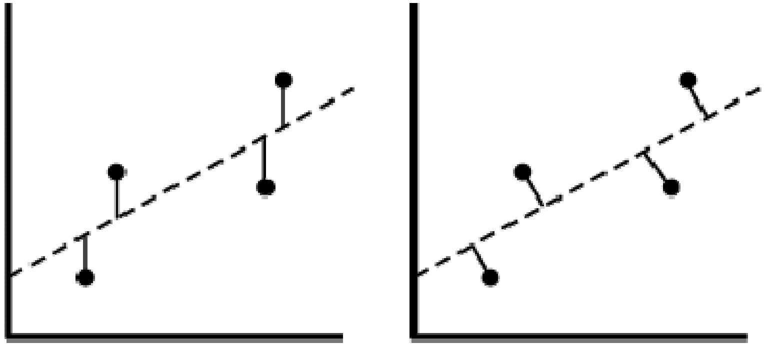
$$P \propto \prod_{i=1}^{N} \left\{ \exp\left[-\frac{1}{2} \left(\frac{y_i - y(x_i)}{\sigma} \right)^2 \right] \Delta y \right\}$$

$$\left[\sum_{i=1}^{N} \frac{[y_i - y(x_i)]^2}{2\sigma^2} \right] - N \log \Delta y$$

$$\text{minimize over } a_1 \dots a_M : \qquad \sum_{i=1}^{N} [y_i - y(x_i; a_1 \dots a_M)]^2$$

$$\chi^2 \equiv \sum_{i=1}^{N} \left(\frac{y_i - y(x_i; a_1 \dots a_M)}{\sigma_i} \right)^2$$

Vertical vs Perpendicular offsets



$$R_\perp^2 \equiv \sum_{i=1}^{n} \frac{[y_i - (a + b\,x_i)]^2}{1 + b^2}$$

$$\frac{\partial R_\perp^2}{\partial a} = \frac{2}{1+b^2} \sum_{i=1}^{n} [y_i - (a + b\,x_i)]\,(-1) = 0$$

$$\frac{\partial R_\perp^2}{\partial b} = \frac{2}{1+b^2} \sum_{i=1}^{n} [y_i - (a + b\,x_i)]\,(-x_i) + \sum_{i=1}^{n} \frac{[y_i - (a + b\,x_i)]^2\,(-1)\,(2\,b)}{(1+b^2)^2} = 0.$$

$$a = \frac{\sum_{i=1}^{n} y_i - b \sum_{i=1}^{n} x_i}{n} = \bar{y} - b\,\bar{x},$$

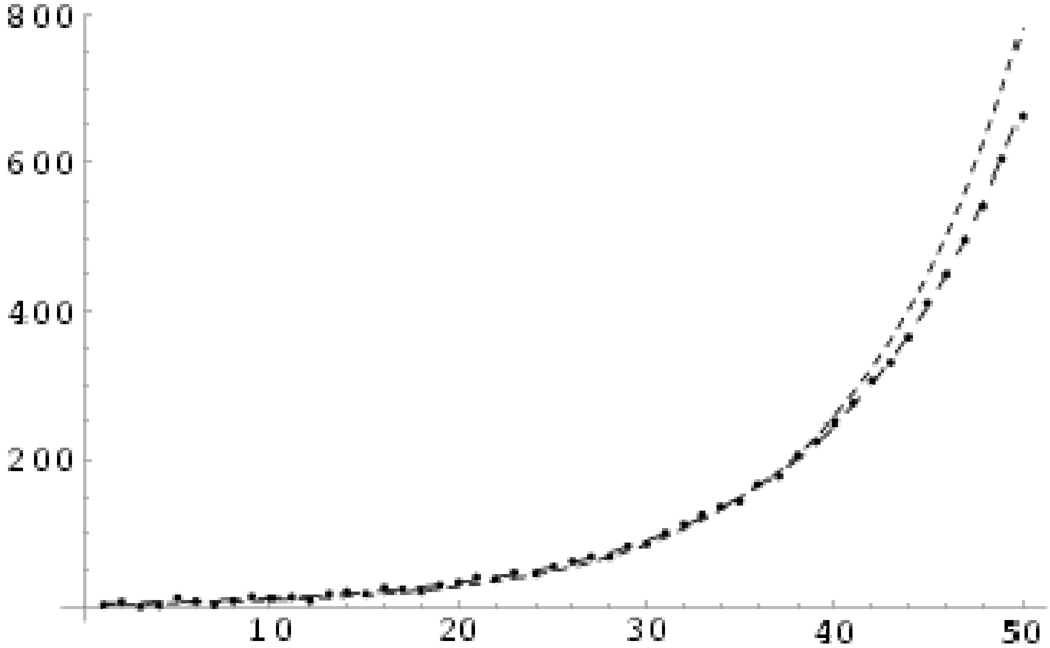
$$b^2 + \frac{\sum_{i=1}^{n} y_i^2 - \sum_{i=1}^{n} x_i^2 + \frac{1}{n}\left[\left(\sum_{i=1}^{n} x_i\right)^2 - \left(\sum_{i=1}^{n} y_i\right)^2\right]}{\frac{1}{n}\sum_{i=1}^{n} x_i \sum_{i=1}^{n} y_i - \sum_{i=1}^{n} x_i\,y_i}\,b - 1 = 0.$$

- "In practice, the *vertical* offsets from a line (polynomial, surface, hyperplane, etc.) are almost always minimized instead of the perpendicular offsets.
- This provides a much simpler analytic form for the fitting parameters.
- Minimizing R^2_{perp} for a second- or higher-order polynomial leads to polynomial equations having *higher* order, so this formulation cannot be extended.
- In any case, for a reasonable number of noisy data points, the difference between vertical and perpendicular fits is quite small." Mathworld

Exponential Fitting Revisited

- Linearizing the equation like we did previously gives too much weight to small y values
- This is not the least squares approximation of the original problem
- Better to minimize another function or to treat the exact leas linearly

$$\sum_{i=1}^{n} y_i \left(\ln y_i - a - b\, x_i\right)^2.$$



$$a = \frac{\sum_{i=1}^{n}\left(x_i^2\, y_i\right)\sum_{i=1}^{n}\left(y_i \ln y_i\right) - \sum_{i=1}^{n}\left(x_i\, y_i\right)\sum_{i=1}^{n}\left(x_i\, y_i \ln y_i\right)}{\sum_{i=1}^{n} y_i \sum_{i=1}^{n}\left(x_i^2\, y_i\right) - \left(\sum_{i=1}^{n} x_i\, y_i\right)^2}$$

$$b = \frac{\sum_{i=1}^{n} y_i \sum_{i=1}^{n}\left(x_i\, y_i \ln y_i\right) - \sum_{i=1}^{n}\left(x_i\, y_i\right)\sum_{i=1}^{n}\left(y_i \ln y_i\right)}{\sum_{i=1}^{n} y_i \sum_{i=1}^{n}\left(x_i^2\, y_i\right) - \left(\sum_{i=1}^{n} x_i\, y_i\right)^2}.$$

Robust Estimation

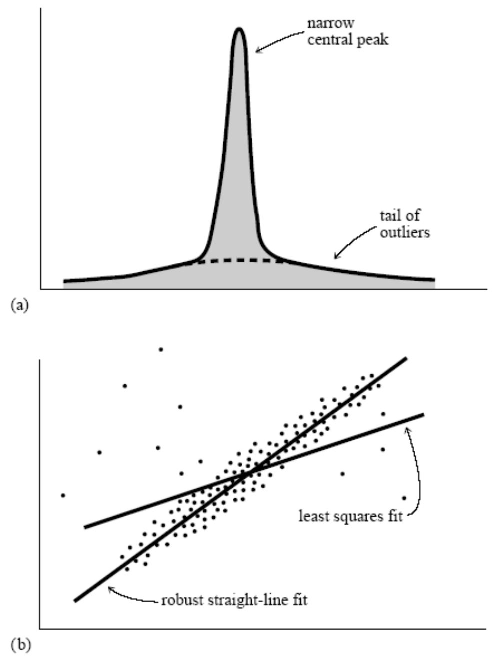


Figure 15.7.1. Examples where robust statistical methods are desirable: (a) A one-dimensional distribution with a tail of outliers; statistical fluctuations in these outliers can prevent accurate determination of the position of the central peak. (b) A distribution in two dimensions fitted to a straight line; non-robust techniques such as least-squares fitting can have undesired sensitivity to outlying points.

- "Insensitive to small departures from the idealized assumptions for which the estimator is optimized."
- Fractionally large departures for a small number of data points
- Can occur when measurement errors are not normally distributed
- General idea is that the weight given individual points should first increase with deviation, then decrease
- decide which estimate you want, that is, ρ
- Ex: if the errors are distributed as a *double* or *two-sided exponential*, namely

$$P = \prod_{i=1}^{N} \{\exp\left[-\rho(y_i, y\{x_i; \mathbf{a}\})\right] \Delta y\}$$

minimize over $\mathbf{a}$ $\quad \displaystyle\sum_{i=1}^{N} \rho\left(\frac{y_i - y(x_i; \mathbf{a})}{\sigma_i}\right)$

$$\text{Prob}\,\{y_i - y(x_i)\} \sim \exp\left(-\left|\frac{y_i - y(x_i)}{\sigma_i}\right|\right)$$

$$y(x; a, b) = a + bx$$

$$\sum_{i=1}^{N} |y_i - a - bx_i|$$

$$a = \text{median}\,\{y_i - bx_i\}$$

$$0 = \sum_{i=1}^{N} x_i \,\text{sgn}(y_i - a - bx_i)$$

What about errors in the inputs?

Weighting errors in y

- If the uncertainties are known, weight the distances with them

$$\chi^2 \equiv \sum_{i=1}^{N} \left(\frac{y_i - y(x_i; a_1 \dots a_M)}{\sigma_i} \right)^2$$

- What if the uncertainties are unknown? Use the chi-square to estimate them. But then, can't use the chi-square to estimate goodness of fit

$$\sigma^2 = \sum_{i=1}^{N} [y_i - y(x_i)]^2 / (N - M)$$

What to do with the errors in x?

- Trick: switch the x- and y-axis when the x errors are bigger than the y errors.
- Numerical Recipes:

$$\chi^2(a,b) = \sum_{i=1}^{N} \frac{(y_i - a - bx_i)^2}{\sigma_{y\,i}^2 + b^2\sigma_{x\,i}^2}$$

$$\mathrm{Var}(y_i - a - bx_i) = \mathrm{Var}(y_i) + b^2\mathrm{Var}(x_i) = \sigma_{y\,i}^2 + b^2\sigma_{x\,i}^2 \equiv 1/w_i$$

$$a = \left[\sum_i w_i(y_i - bx_i)\right] \Big/ \sum_i w_i$$

$\partial\chi 2/\partial a = 0$, is still linear
$\partial\chi 2/\partial b = 0$ is nonlinear.

- ROOT

$$\chi^2 = \frac{(y - f(x))^2}{\sigma_y^2 + ((f(x+\sigma_x) - f(x-\sigma_x))/2)^2}$$

- LSM program

$$\sum_i \left[\frac{(x_i - x_{oi})^2}{\sigma_{x_i}{}^2} + \frac{(y_i - y_{oi})^2}{\sigma_{y_i}{}^2}\right]$$

What about errors in the outputs?

How to calculate them?

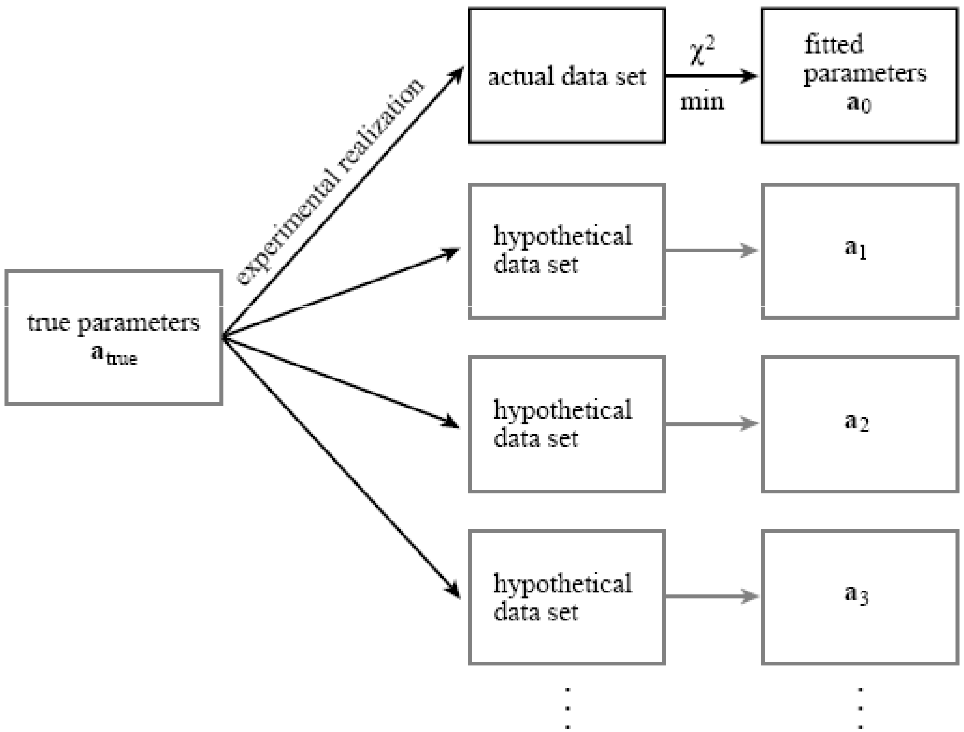


Figure 15.6.1. A statistical universe of data sets from an underlying model. True parameters $\mathbf{a}_{true}$ are realized in a data set, from which fitted (observed) parameters $\mathbf{a}_0$ are obtained. If the experiment were repeated many times, new data sets and new values of the fitted parameters would be obtained.

$\mathbf{a}_{(i)} - \mathbf{a}_{true}$: If we knew *this* distribution, we would know everything that there is to know about the quantitative uncertainties in our experimental measurement $\mathbf{a}_{(0)}$. So the name of the game is to find some way of estimating or approximating this probability distribution without knowing $\mathbf{a}_{true}$ and without having available to us an infinite universe of hypothetical data sets.

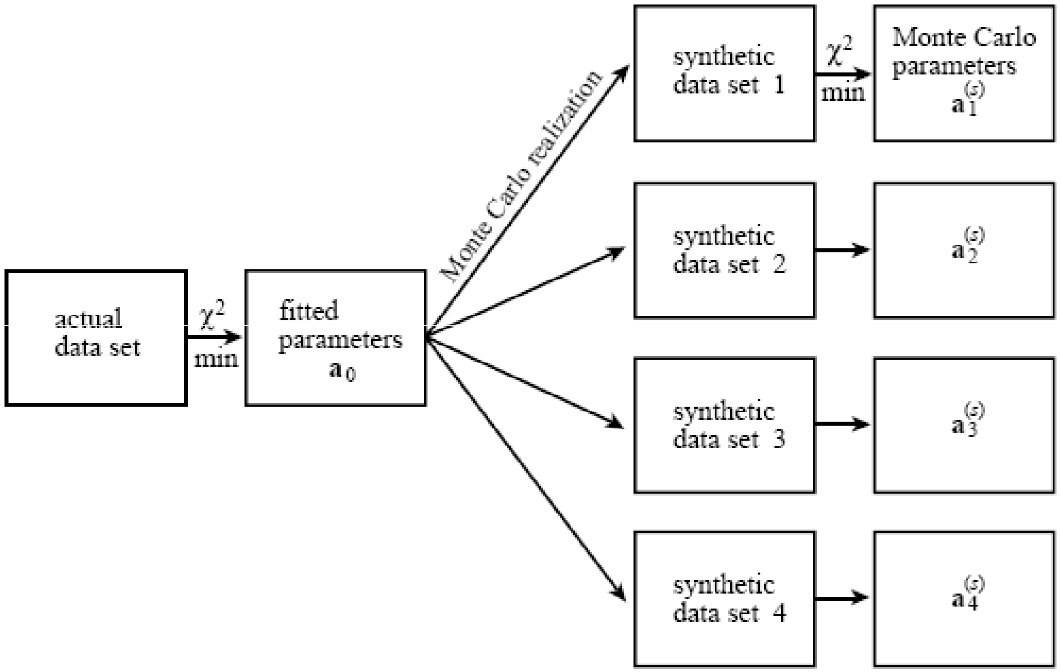


Figure 15.6.2. Monte Carlo simulation of an experiment. The fitted parameters from an actual experiment are used as surrogates for the true parameters. Computer-generated random numbers are used to simulate many synthetic data sets. Each of these is analyzed to obtain its fitted parameters. The distribution of these fitted parameters around the (known) surrogate true parameters is thus studied.

Let us *assume* — that the shape of the probability distribution $\mathbf{a}_{(i)} - \mathbf{a}_{(0)}$ in the fictitious world is the same, or very nearly the same, as the shape of the probability distribution $\mathbf{a}_{(i)} - \mathbf{a}_{\text{true}}$ in the real world.

How to interpret them?

- "Rather than present all details of the probability distribution of errors in parameter estimation, it is common practice to summarize the distribution in the form of *confidence limits*.
- A *confidence region* (or *confidence interval*) is just a region of that *M*-dimensional space (hopefully a small region) that contains a certain (hopefully large) percentage of the total probability distribution.
- The experimenter, get to pick both the *confidence level* (99 percent in the above example), and the shape of the confidence region. The only requirement is that your region does include the stated percentage of probability.", Numerical Recipes

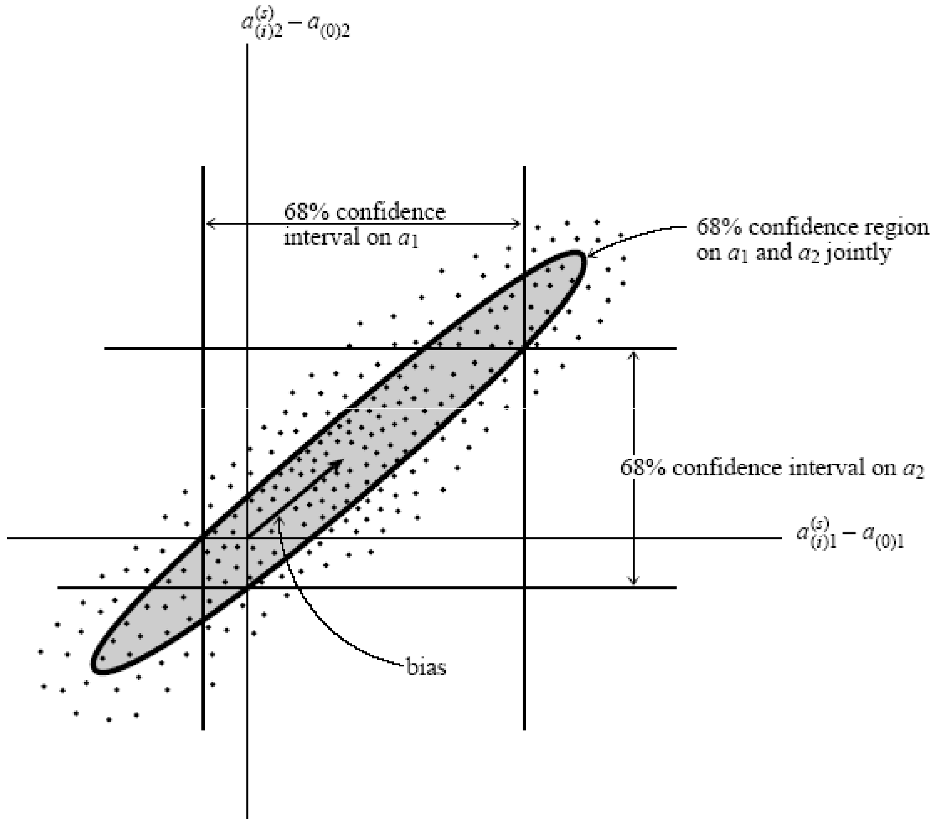


Figure 15.6.3. Confidence intervals in 1 and 2 dimensions. The same fraction of measured points (here 68%) lies (i) between the two vertical lines, (ii) between the two horizontal lines, (iii) within the ellipse.

- "When the method used to estimate the parameters $\mathsf{a}_{(0)}$ is chi-square minimization then there is a natural choice for the shape of confidence intervals.
- The region within which $\chi 2$ increases by no more than a set amount $\Delta\chi 2$ defines some *M*–dimensional confidence region around $\mathsf{a}(0)$", Numerical Recipes

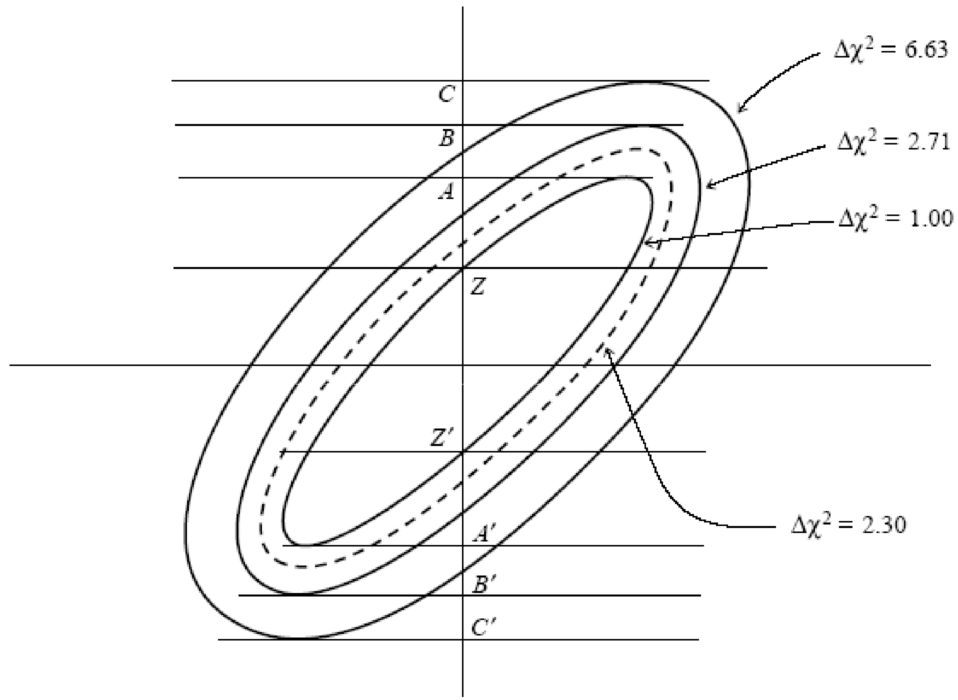


Figure 15.6.4. Confidence region ellipses corresponding to values of chi-square larger than the fitted minimum. The solid curves, with $\Delta\chi^2 = 1.00, 2.71, 6.63$ project onto one-dimensional intervals AA', BB', CC'. These intervals — not the ellipses themselves — contain 68.3%, 90%, and 99% of normally distributed data. The ellipse that contains 68.3% of normally distributed data is shown dashed, and has $\Delta\chi^2 = 2.30$. For additional numerical values, see accompanying table.

The formal covariance matrix that comes out of a $\chi 2$ minimization has a clear quantitative interpretation only if (or to the extent that) the measurement errors actually are normally distributed. In the case of *non*normal errors, you are "allowed"

- to fit for parameters by minimizing $\chi 2$
- to use a contour of constant $\Delta\chi 2$ as the boundary of your confidence region
- to use Monte Carlo simulation or detailed analytic calculation in determining *which* contour $\Delta\chi 2$ is the correct one for your desired confidence level
- to give the covariance matrix C_{ij} as the "formal covariance matrix of the fit."

You are *not* allowed

- to use formulas that we now give for the case of normal errors, which establish quantitative relationships among $\Delta\chi 2$, C_{ij}, and the confidence level.

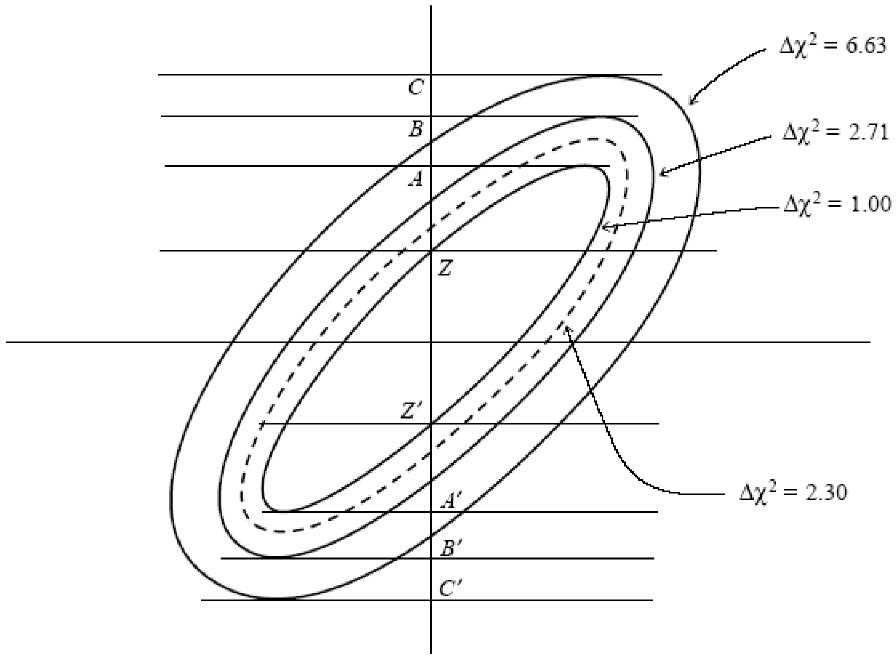


Figure 15.6.4. Confidence region ellipses corresponding to values of chi-square larger than the fitted minimum. The solid curves, with $\Delta\chi^2 = 1.00, 2.71, 6.63$ project onto one-dimensional intervals AA', BB', CC'. These intervals — not the ellipses themselves — contain 68.3%, 90%, and 99% of normally distributed data. The ellipse that contains 68.3% of normally distributed data is shown dashed, and has $\Delta\chi^2 = 2.30$. For additional numerical values, see accompanying table.

$\Delta\chi^2$ as a Function of Confidence Level and Degrees of Freedom

	ν					
p	1	2	3	4	5	6
68.3%	1.00	2.30	3.53	4.72	5.89	7.04
90%	2.71	4.61	6.25	7.78	9.24	10.6
95.4%	4.00	6.17	8.02	9.70	11.3	12.8
99%	6.63	9.21	11.3	13.3	15.1	16.8
99.73%	9.00	11.8	14.2	16.3	18.2	20.1
99.99%	15.1	18.4	21.1	23.5	25.7	27.8

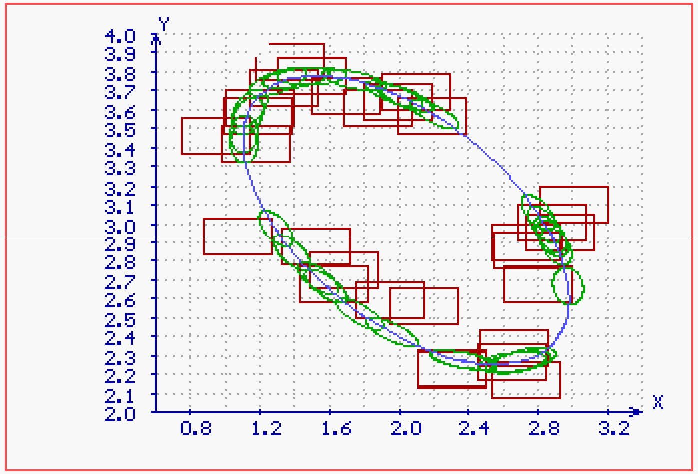
$$\delta a_1 = \pm\sqrt{\Delta\chi^2_\nu}\sqrt{C_{11}}$$

Program comparisons

- Maple
- Matlab & MFIT
- Root
- Origin
- LSM
- Kaleidagraph
- Excel

Things I didn't talk about

- Testing the fit
- Singular Value Decomposition for the general linear least squares fitting
- Covariance matrix
- Maximum likelihood method
- The method of Moments
- Is there a package that uses perpendicular offsets and uses errors in all dimensions?
- Fit to non fcts.



Interesting readings and references

- McGill University, *Lab Notes*
- Burden and Faires, *Numerical Analysis*
- Eric W. Weisstein. "*Least Squares Fitting*." From *Math World--*-A Wolfram Web Resource. http://mathworld.wolfram.com/LeastSquaresFitting.html
- Taylor, *An Introduction to Error Analysis The Study of Uncertainties in Physical Measurements*
- Bevington and Robinson, *Data Reduction and Error Analysis for the Physical Sciences*
- Frodesen, Skjeggestad and Tøfte, *Probability and Statistics in Particle Physics*
- *Numerical Recipes in C: The Art of Scientific Computing*
- Least Squares Method Curve Fitting Program, http://www.prz.rzeszow.pl/~janand/
- MINUIT Documentation, http://wwwasdoc.web.cern.ch/wwwasdoc/minuit/minmain.html
- ROOT Documentation, http://root.cern.ch/