Least squares methods

Volker Blobel – University of Hamburg

October 2003

1. The least squares principle
2. Linear least squares
3. Properties of least squares estimates
4. Independent data
5. χ^2 minimization
6. Orthogonal polynomials
7. Example: Large number of parameters
8. Nonlinear least squares
9. Least squares with constraints
10. Weighted least squares

The least squares principle

A model with parameters is assumed to describe the data.

Principle of parameter estimation: minimize sum of squares of deviations Δy_i between model and data!

Solution: derivatives of S w.r.t. parameters = zero!

Different forms: sum of squared deviations, weighted sum of squared deviations, sum of squared deviations weighted with inverse covariance matrix:

$$S = \sum_{i=1}^{n} \Delta y_i^2 \qquad S = \sum_{i=1}^{n} \left(\frac{\Delta y_i}{\sigma_i} \right)^2 \qquad S = \Delta \boldsymbol{y}^T \, \boldsymbol{V}^{-1} \, \Delta \boldsymbol{y}$$

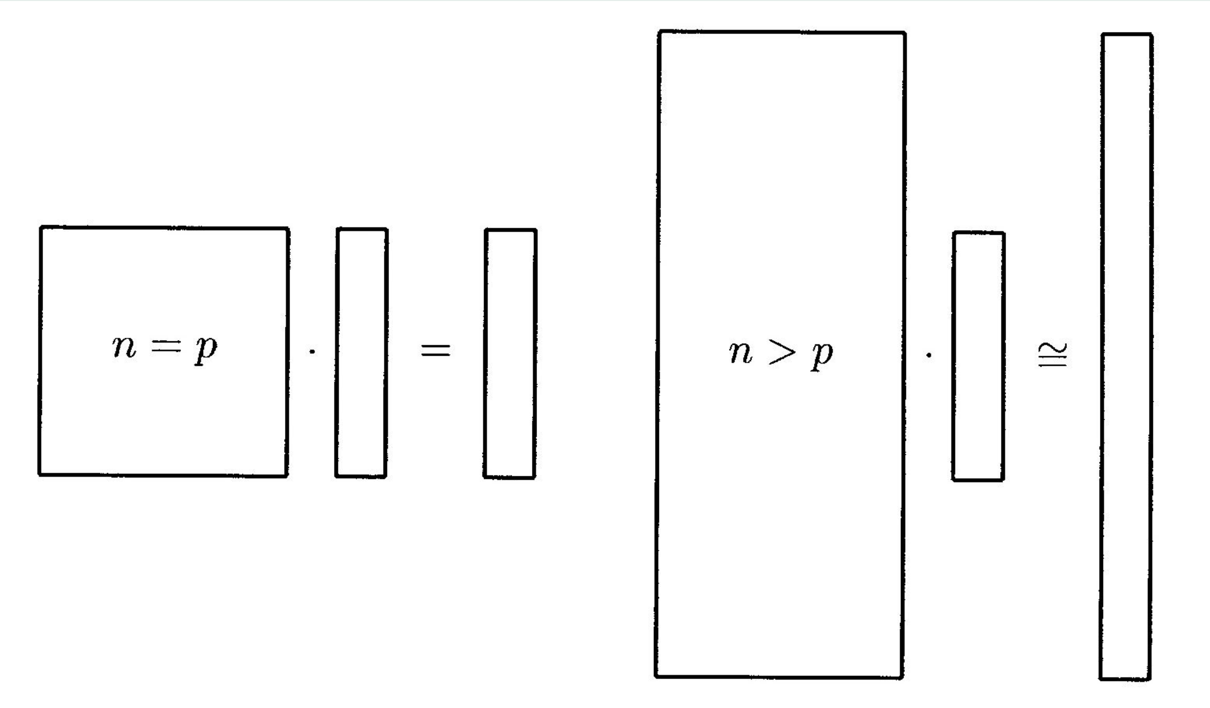
Example: mean value of n measured values y_i:

$$S = \sum_{i=1}^{n} (y - y_i)^2 = \text{ minimum} \qquad \hat{y} = \sum_{i=1}^{n} y_i / n$$

Systems of linear equations

$$\boldsymbol{A} \cdot \boldsymbol{a} = \boldsymbol{y} \qquad\qquad \boldsymbol{A} \cdot \boldsymbol{a} \cong \boldsymbol{y}$$

with n elements of the measured vector $\boldsymbol{y}$ and p elements of the parameter vector $\boldsymbol{a}$.



Linear least squares

The model of Linear Least Squares: $\boldsymbol{y} = \boldsymbol{A}\,\boldsymbol{a}$

$$\begin{aligned}
\boldsymbol{y} &= \text{vector of measured data } (n \text{ elements}) \\
\boldsymbol{A} &= \text{matrix (fixed)} \\
\boldsymbol{a} &= \text{vector of parameters } (p \text{ elements}) \\
\boldsymbol{r} &= \boldsymbol{y} - \boldsymbol{A}\boldsymbol{a} = \text{vector of residuals} \\
\boldsymbol{V}[\boldsymbol{y}] &= \text{covariance matrix of the data} \\
\boldsymbol{W} &= \boldsymbol{V}[\boldsymbol{y}]^{-1} \;\; \text{weight matrix}
\end{aligned}$$

Least Squares Principle: minimize the expression

$$S(\boldsymbol{a}) = \boldsymbol{r}^T \boldsymbol{W} \boldsymbol{r} = (\boldsymbol{y} - \boldsymbol{A}\boldsymbol{a})^T \; W \; (\boldsymbol{y} - \boldsymbol{A}\boldsymbol{a})$$

with respect to $\boldsymbol{a}$.

Least Squares solution

Derivatives of expression $S(\boldsymbol{a})$:

$$\begin{aligned}
\frac{1}{2}\frac{\partial S}{\partial \boldsymbol{a}} &= \left(-\boldsymbol{A}^T\boldsymbol{W}\boldsymbol{y} + \left(\boldsymbol{A}^T\boldsymbol{W}\boldsymbol{A}\right)\boldsymbol{a}\right) \\
\frac{1}{2}\frac{\partial^2 S}{\partial \boldsymbol{a}^2} &= \left(\boldsymbol{A}^T\boldsymbol{W}\boldsymbol{A}\right) \qquad = \text{constant}
\end{aligned}$$

Solution (from $\partial S/\partial \boldsymbol{a} = 0$)

$$-\boldsymbol{A}^T\boldsymbol{W}\boldsymbol{y} + \left(\boldsymbol{A}^T\boldsymbol{W}\boldsymbol{A}\right)\boldsymbol{a} = 0$$

is linear transformation of the data vector $\boldsymbol{y}$:

$$\hat{\boldsymbol{a}} = \left(\boldsymbol{A}^T\boldsymbol{W}\boldsymbol{A}\right)^{-1}\ \boldsymbol{A}^T\boldsymbol{W}\,\boldsymbol{y} \qquad = \boldsymbol{B}\boldsymbol{y}$$

Covariance matrix of $\boldsymbol{a}$ by "error" propagation ($\boldsymbol{V}[\boldsymbol{y}] = \boldsymbol{W}^{-1}$):

$$\begin{aligned}
\boldsymbol{V}[\hat{\boldsymbol{a}}] &= \boldsymbol{B}\,\boldsymbol{V}[\boldsymbol{y}]\,\boldsymbol{B}^T = \left(\boldsymbol{A}^T\boldsymbol{W}\boldsymbol{A}\right)^{-1}\boldsymbol{A}^T\boldsymbol{W}\ \boldsymbol{W}^{-1}\ \boldsymbol{W}\boldsymbol{A}\left(\boldsymbol{A}^T\boldsymbol{W}\boldsymbol{A}\right)^{-1} \\
&= \left(\boldsymbol{A}^T\boldsymbol{W}\boldsymbol{A}\right)^{-1} \qquad = \text{inverse of second derivate of } S
\end{aligned}$$

Properties of solution

General scheme

Starting from **Principles** properties of the solution are derived, which are valid under certain conditions.

Conditions:

- Data are unbiased: $E[\boldsymbol{y}] = \boldsymbol{A}\,\boldsymbol{a}_{\text{true}}$ $\quad$ ($\boldsymbol{a}_{\text{true}}$ = true parameter vector)
- Covariance matrix $\boldsymbol{V}[\boldsymbol{y}]$ is known (correct) and finite

Properties:

- Estimated parameters are unbiased:

$$E[\hat{\boldsymbol{a}}] = \left(\boldsymbol{A}^T\boldsymbol{W}\boldsymbol{A}\right)^{-1}\,\boldsymbol{A}^T\boldsymbol{W}\,E[\boldsymbol{y}] = \boldsymbol{a}_{\text{true}}$$

- In the class of unbiased estimates $\boldsymbol{a}^*$, which are linear in the data, the Least Squares estimates $\hat{\boldsymbol{a}}$ have the smallest variance (Gauß-Markoff theorem)

Properties are not valid, if conditions violated.

Simplification for independent (=uncorrelated) data

…assuming same variance σ^2 for all data.

Covariance matrix and weight matrix are diagonal:

$$\boldsymbol{V}(\boldsymbol{y}) = \sigma^2 \boldsymbol{I}_n \qquad\qquad \boldsymbol{W} = \frac{1}{\sigma^2}\boldsymbol{I}_n$$

($\boldsymbol{I}_n$ is n-by-n unit matrix).

$$\begin{array}{rrcl} \text{solution} & \hat{\boldsymbol{a}} & = & \boldsymbol{C}^{-1}\boldsymbol{A}^T\boldsymbol{y} \qquad \text{with} \quad \boldsymbol{C} = \boldsymbol{A}^T\boldsymbol{A} \\ \text{covariance matrix} & \boldsymbol{V}(\hat{\boldsymbol{a}}) & = & \sigma^2\boldsymbol{C}^{-1} \end{array}$$

Note: the solution $\hat{\boldsymbol{a}}$ does not depend on σ^2, but the covariance matrix is proportional to σ^2.

Properties of least square estimates

Basic assumptions on the properties of the data:

1. the data are unbiased: $E\left[\boldsymbol{y}\right] = \boldsymbol{A}\boldsymbol{a}_{\text{true}}$ or $E\left[\boldsymbol{y} - \boldsymbol{A}\boldsymbol{a}_{\text{true}}\right] = 0$
2. the variances are all the same: $\boldsymbol{V}\left[\boldsymbol{y} - \boldsymbol{A}\boldsymbol{a}_{\text{true}}\right] = \sigma^2 \boldsymbol{I}_n$

(i.e. special case of independent data of same precision is assumed).

No assumption is made on the *distribution* of the residuals (i.e. a Gaussian distribution is not required!)

Least squares estimates:

$$\hat{\boldsymbol{a}} = \boldsymbol{C}^{-1}\boldsymbol{A}^T\boldsymbol{y} \quad \text{with} \quad \boldsymbol{C} = \boldsymbol{A}^T\boldsymbol{A} \qquad\qquad \boldsymbol{V}\left[\hat{\boldsymbol{a}}\right] = \sigma^2\,\boldsymbol{C}^{-1}$$

First property: Least square estimates are unbiased.

Proof:

$$E\left[\hat{\boldsymbol{a}}\right] = \boldsymbol{C}^{-1}\boldsymbol{A}^T E\left[\boldsymbol{y}\right] = \boldsymbol{C}^{-1}\boldsymbol{A}^T\boldsymbol{A}\,\boldsymbol{a}_{\text{true}} = \boldsymbol{a}_{\text{true}}$$

Gauß-Markoff Theorem

Consider class of linear estimates $\boldsymbol{a}^* = \boldsymbol{U}\boldsymbol{y}$, which are unbiased:

$$E\left[\boldsymbol{a}^*\right] = \boldsymbol{U} E\left[\boldsymbol{y}\right] = \underbrace{\boldsymbol{U}\boldsymbol{A}}_{=\boldsymbol{I}_p} \boldsymbol{a}_{\text{true}} = \boldsymbol{a}_{\text{true}} \qquad \boldsymbol{V}\left[\boldsymbol{a}^*\right] = \sigma^2 \boldsymbol{U}\boldsymbol{U}^T$$

Case of least squares: $\boldsymbol{U}_{LS} = \boldsymbol{C}^{-1}\boldsymbol{A}^T$ with $\boldsymbol{V}\left[\hat{\boldsymbol{a}}\right] = \sigma^2\, \boldsymbol{C}^{-1}$.

Theorem: The least square estimate $\hat{\boldsymbol{a}}$ has the property

$$\boldsymbol{V}\left[\boldsymbol{a}^*\right]_{jj} \geq \boldsymbol{V}\left[\hat{\boldsymbol{a}}\right]_{jj} \quad \text{for all } j\,,$$

i. e., the least squares estimate has the smallest possible error.

Proof: product $\boldsymbol{U}\boldsymbol{U}^T$ can be written in the form

$$\begin{aligned} \boldsymbol{U}\boldsymbol{U}^T &= \boldsymbol{C}^{-1} + (\boldsymbol{U} - \boldsymbol{C}^{-1}\boldsymbol{A}^T)(\boldsymbol{U} - \boldsymbol{C}^{-1}\boldsymbol{A}^T)^T \\ &= \boldsymbol{C}^{-1} + \boldsymbol{U}\boldsymbol{U}^T - \boldsymbol{U}\boldsymbol{A}\boldsymbol{C}^{-1} - \boldsymbol{C}^{-1}\boldsymbol{A}^T\boldsymbol{U}^T + \boldsymbol{C}^{-1}\boldsymbol{A}^T\boldsymbol{A}\boldsymbol{C}^{-1} \end{aligned}$$

For the covariance matrix follows:

$$\boldsymbol{V}\left[\boldsymbol{a}^*\right] = \boldsymbol{V}\left[\hat{\boldsymbol{a}}\right] + \sigma^2(\boldsymbol{U} - \boldsymbol{C}^{-1}\boldsymbol{A}^T)(\boldsymbol{U} - \boldsymbol{C}^{-1}\boldsymbol{A}^T)^T$$

Product on the right has diagonal elements ≥ 0 (→ Theorem).

Sum of squares of residuals

Third property: The expectation of the sum of squares of the residuals is $\hat{S} = \sigma^2(n-p)$.

Definition of $\hat{S}$ in terms of the fitted vector $\hat{\boldsymbol{a}}$:

$$\hat{S} = (\boldsymbol{y} - \boldsymbol{A}\hat{\boldsymbol{a}})^T(\boldsymbol{y} - \boldsymbol{A}\hat{\boldsymbol{a}}) = \boldsymbol{y}^T\boldsymbol{y} - \boldsymbol{y}^T\boldsymbol{A}\hat{\boldsymbol{a}}$$

This equation is rewritten in terms of $\boldsymbol{a}_{\text{true}}$ (instead of $\hat{\boldsymbol{a}}$) using the matrix $\boldsymbol{U} = \boldsymbol{I}_n - \boldsymbol{A}\boldsymbol{C}^{-1}\boldsymbol{A}^T$ and the vector $\boldsymbol{z} = \boldsymbol{y} - \boldsymbol{A}\boldsymbol{a}_{\text{true}}$.

$$\hat{S} = (\boldsymbol{y} - \boldsymbol{A}\boldsymbol{a}_{\text{true}})^T\boldsymbol{U}(\boldsymbol{y} - \boldsymbol{A}\boldsymbol{a}_{\text{true}}) = \boldsymbol{z}^T\boldsymbol{U}\boldsymbol{z}$$

(check the agreement with $\hat{S}$ above by multiplication).

Properties of $\boldsymbol{z}$: $E[\boldsymbol{z}] = 0$ and covariance matrix

$$\boldsymbol{V}[\boldsymbol{z}] = \sigma^2\boldsymbol{I}_n \quad \text{i.e.} \quad V[z_i] = E\left[z_i^2\right] = \sigma^2 \quad \text{and} \quad E[z_i z_k] = 0\ .$$

$$\hat{S} = \sum_{i=1}^{n} \sum_{k=1}^{n} U_{ik} \; z_i \; z_k \qquad E\left[\hat{S}\right] = \sum_{i=1}^{n} U_{ii} \; E\left[z_i^2\right] = \sigma^2 \sum_{i=1}^{n} U_{ii} = \sigma^2 \; \mathrm{trace}(\boldsymbol{U})$$

(the trace of a square matrix is the sum of the diagonal elements). Calculation of the trace of $\boldsymbol{U}$:

$$\begin{aligned} \mathrm{trace}(\boldsymbol{U}) &= \mathrm{trace}(\boldsymbol{I}_n - \boldsymbol{A}\boldsymbol{C}^{-1}\boldsymbol{A}^T) = \mathrm{trace}(\boldsymbol{I}_n) - \mathrm{trace}(\boldsymbol{A}\boldsymbol{C}^{-1}\boldsymbol{A}^T) \\ &= \mathrm{trace}(\boldsymbol{I}_n) - \mathrm{trace}(\boldsymbol{C}^{-1}\boldsymbol{A}^T\boldsymbol{A}) \\ &= \mathrm{trace}(\boldsymbol{I}_n) - \mathrm{trace}(\boldsymbol{I}_p) = n - p. \qquad \longrightarrow \; \text{Proof} \end{aligned}$$

Application: estimate data variance (for $n \gg p$) by $\widehat{\sigma^2} = \hat{S}/(n-p)$

Special case of Gaussian distributed measurement errors:

$\hat{S}/\sigma^2$ distributed according to the χ^2_{n-p} distribution

to be used for goodness-of-fit test.

Summary of properties

Distribution-free properties of least squares estimates in linear problems:

1. Least square estimates are unbiased.
2. The least square estimate $\hat{\boldsymbol{a}}$ has the property

$$\boldsymbol{V}\left[\boldsymbol{a}^*\right]_{jj} \geq \boldsymbol{V}\left[\hat{\boldsymbol{a}}\right]_{jj} \quad \text{for all } j\,,$$

   i. e., the least squares estimate has the smallest possible error. (Gauß-Markoff Theorem)

3. The expectation of the sum of squares of the residuals is $\hat{S} = \sigma^2(n-p)$.

Valid under the condition that the data are unbiased!

Independent data

Often the direct measurements, which are input to a least squares problem, are *independent*, i.e. the covariance matrix $\boldsymbol{V}(\boldsymbol{y})$ and the weight matrix $\boldsymbol{W}$ are *diagonal*.

This property, which is assumed here, simplifies the computation of the matrix products

$$\boldsymbol{C} = \boldsymbol{A}^T \boldsymbol{W} \boldsymbol{A} \qquad \text{and} \qquad \boldsymbol{b} = \boldsymbol{A}^T \boldsymbol{W} \boldsymbol{y}$$

which are necessary for the solution

$$\hat{\boldsymbol{a}} = \boldsymbol{C}^{-1} \boldsymbol{b} \qquad\qquad \boldsymbol{V}(\hat{\boldsymbol{a}}) = \boldsymbol{C}^{-1}$$

Note: the parameters $\boldsymbol{a}$ will be correlated through the model $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{a}$ and the covariance matrix $\boldsymbol{V}(\hat{\boldsymbol{a}})$ will be non-diagonal.

Normal equations for independent data

The diagonal elements of the weight matrix $\boldsymbol{W}$ are denoted by w_i, with $w_i = 1/\sigma_i^2$. Each data value y_i with its weight w_i makes an independent contribution to the final matrix products. Calling the i-th row $\boldsymbol{A}_i$, with

$$i\text{-th row of } \boldsymbol{A} \qquad \boldsymbol{A}_i = (d_1,\, d_2,\, \ldots,\, d_p) \qquad y = d_1 a_1 + d_2 a_2 + \ldots + d_p a_p$$

the contributions of this row to $\boldsymbol{C}$ and $\boldsymbol{b}$ can be written as the $p \times p$-matrix $w_i \boldsymbol{A}_i^T \cdot \boldsymbol{A}_i$ and the p-vector $w_i \boldsymbol{A}_i^T \cdot y_i$. The contributions of a single row are:

$$\begin{array}{c|cccc} & d_1 & d_2 & \ldots & d_p \\ \hline d_1 & w_i d_1^2 & w_i d_1 d_2 & \ldots & w_i d_1 d_p \\ d_2 & & w_i d_2^2 & \ldots & w_i d_2 d_p \\ \ldots & & & \ldots & \ldots \\ d_p & & & & w_i d_p^2 \end{array} \qquad \begin{array}{c|c} & y_i \\ \hline d_1 & w_i d_1 y_i \\ d_2 & w_i d_2 y_i \\ \ldots & \ldots \\ d_p & w_i d_p y_i \end{array} \;,$$

where the symmetric elements in the lower half are not shown.

Straight line fit

Example: track fit of y (measured) vs. abscissa x

$$y_i = a_0 + a_1 \cdot x_i$$

Matrix $\boldsymbol{A}$ and parameter vector $\boldsymbol{a}$

$$\boldsymbol{A} = \begin{pmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_n \end{pmatrix} \qquad \boldsymbol{a} = \begin{pmatrix} a_0 \\ a_1 \end{pmatrix}$$

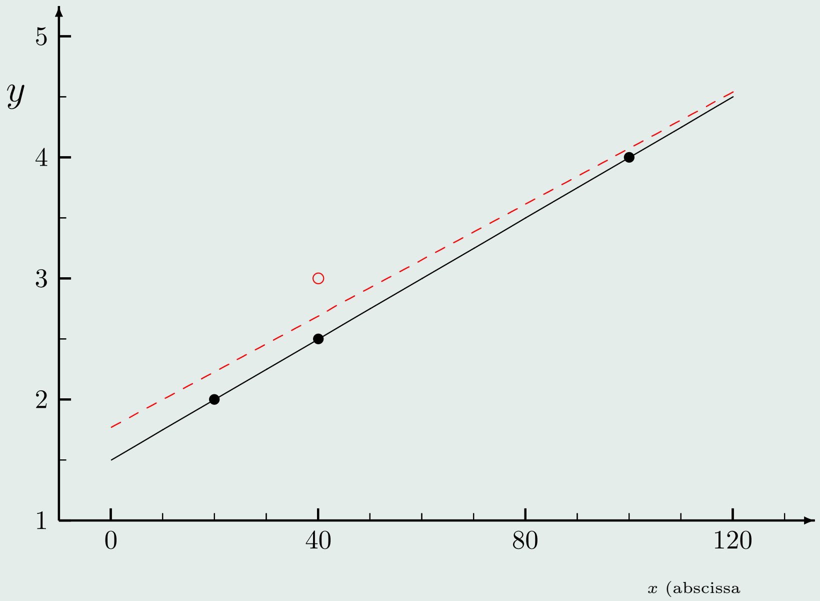
Weight matrix is diagonal (*independent* measurements):

$$\boldsymbol{C} = \boldsymbol{A}^T \boldsymbol{W} \boldsymbol{A} = \begin{pmatrix} \sum w_i & \sum w_i x_i \\ \sum w_i x_i & \sum w_i x_i^2 \end{pmatrix} \qquad \boldsymbol{b} = \boldsymbol{A}^T \boldsymbol{W} \boldsymbol{y} = \begin{pmatrix} \sum w_i y_i \\ \sum w_i x_i y_i \end{pmatrix}$$

If one measured y_i-value is shifted (biased), then

- parameters biased, and
- χ^2-value very high.

Straight line fit



The full line is a straight line fit to three well aligned data points (black dots).

The dashed curve is the straight line fit, if the middle point is ”badly aligned” (circle).

Recipe for robust least square fit

Assume estimate for the standard error of y_i (or of r_i) to be s_i.
Do least square fit on observations y_i, yielding fitted values $\hat{y}_i$, and residuals $r_i = y_i - \hat{y}_i$.

- "Clean" the data by pulling outliers towards their fitted values: winsorize the observations y_i and replace them by pseudo-observations y_i^*:

  $$\begin{aligned} y_i^* &= y_i \, , & \text{if} \quad |r_i| \le c\, s_i \ , \\ &= \hat{y}_i - c\, s_i \, , & \text{if} \quad r_i < -c\, s_i \ , \\ &= \hat{y}_i + c\, s_i \, , & \text{if} \quad r_i > +c\, s_i \ . \end{aligned}$$

  The factor c regulates the amount of robustness, a goid choice is $c = 1.5$.

- Refit iteratively: the pseudo-observations y_i^* are used to calculate new parameters and new fitted values $\hat{y}_i$.

Robust least square fit

Classical estimate of s_i, if all observations are equally accurate:

$$s^2 = \frac{1}{n-p} \sum_{i=1}^{n} r_i^2$$

($(n-p)$ = number of observations minus the number of parameters). Estimate standard error of residual s_i by $s_i = \sqrt{1-H_{ii}} s$, where H_{ii} is the diagonal element of $\boldsymbol{H} = \boldsymbol{X}\left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T$.

Using modified (reduced) residuals $r_i^* = y_i^* - \hat{y}_i$ instead of r_i, the (bias corrected) estimate of s^2 is

$$s^2 = \left(\frac{n}{m}\right)^2 \frac{1}{n-p} \sum_{i=1}^{n} r_i^{*2}$$

where m is the number of unmodified observations ($y_i^* = y_i$).

Least squares and Maximum Likelihood method

Example: straight line fit of y (measured data) vs. abscissa x

$$y_i = a_0 + a_1 \cdot x_i \; .$$

In the Maximum Likelihood method, assuming a Gaussian distribution of the data:

$$p(x_i | a_0, \, a_1) = \frac{1}{\sqrt{2\pi}\sigma_i} \exp\left(- \frac{(y_i - a_0 - a_1 x_i)^2}{2\sigma_i^2} \right) \; ,$$

the Likelihood function is

$$\mathcal{L}(a_0, \, a_1) = p(x_1 | a_0, \, a_1) \cdot p(x_2 | a_0, \, a_1) \cdots p(x_n | a_0, \, a_1) = \prod_{i=1}^{n} p(x_i | a_0, \, a_1) \; .$$

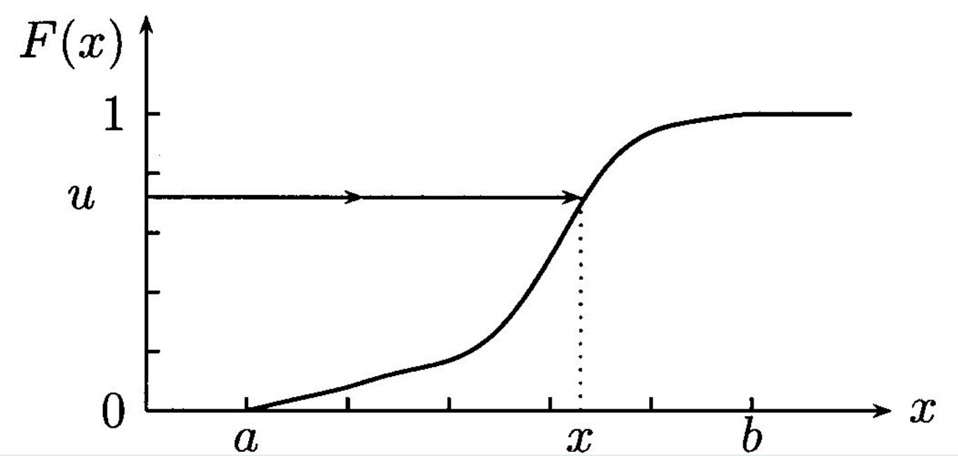
Maximizing the $\mathcal{L}(a_0, \, a_1)$ w.r.t. $a_0, \, a_1$ is equivalent to minimizing

$$-2 \ln \mathcal{L}(a_0, \, a_1) = \sum_{i=1}^{n} \frac{(y_i - a_0 - a_1 x_i)^2}{\sigma_i^2} + \text{const.}$$

Relation between χ^2 and probability

Assume x follows the density $f(x)$. The cumulative probability $F(x)$ is defined as integral:

$$\int_{-\infty}^{x} f(x')\,\mathrm{d}x' = F(x) = u.$$



If the random variable x is transformed to the random variable u, then the random variable u (and also $1-u$) will follow the uniform distribution $U(0,1)$.

For the χ^2 distribution: probability $P = 1 - F_n(\chi^2)$ should follow a uniform distribution ($n =$ number of degrees of freedom).

χ^2 minimisation

Confusion in terminology: A popular method for parameter estimation is χ^2 minimisation $\chi^2 = \boldsymbol{\Delta}^T \boldsymbol{V}^{-1} \boldsymbol{\Delta}$ – is this identical to least squares?
The minimum value of the objective function in Least Squares follows often (not always) a χ^2 distribution.
In contrast to the well-defined standard methods

- in χ^2 minimisation a variety of different non-standard concepts is used,
- often apparently motivated by serious problems to handle the experimental data in a consistent way;
- especially for the error estimation there are non-standard concepts and methods.

From publications:

> To determine these parameters one must minimize a χ^2 which compares the measured values . . . to the calculated ones . . .
>
> Our analysis is based on an effective global chi-squared function that measures the quality of the fit between theory and experiment . . .

Two examples are given, which demonstrate that χ^2 minimisation can give **biased results**:

- Calorimeter calibration
- Averaging data with common normalisation error

Calorimeters for energy measurements in a particle detector require a calibration, usually based on test beam data (measured cell energies y_{ik}) with known energy E. A common method [?, ?, ?, ?, ?, ?, ?, ?] based on the χ^2 minimisation of

$$\chi^2 = \frac{1}{N} \sum_{k=1}^{N} \left(a_1 y_{1,k} + a_2 y_{2,k} + \ldots + a_n y_{n,k} - E \right)^2$$

for the determination of the a_j can produce biased results, as pointed out by D. Lincoln et al. [?].

If there would be one cell only, one would have data y_k with standard deviation σ, with a mean value of $\overline{y} = \sum_k y_k / N$, and the intended result is simply $a = E/\overline{y}$

A one-cell version of the above χ^2 definition is

$$\chi^2 = \frac{1}{N} \sum_{k=1}^{N} \left(a \cdot y_k - E \right)^2$$

and minimizing this χ^2 has the biased result

$$a = \frac{E \cdot \overline{y}}{\left(\sum_k y_k^2 \right) / N} = \frac{E \cdot \overline{y}}{\overline{y}^2 + \sigma^2} \qquad \neq E/\overline{y}$$

. . . contnd.

The bias mimics a non-linear response of the calorimeter.
A known bias in fitted parameters is easily corrected for.

Example: for a hadronic calorimeter one may have

Energy resolution $\dfrac{\sigma}{E} = \dfrac{0.7}{\sqrt{E}}$ which will result in a biased ratio $= \dfrac{E}{E + 0.7^2}$

(at $E = 10$ GeV the resolution is 22 % and the bias is 5 %).

There would be no bias, if the inverse constant a_{inv} would have been determined from

$$\chi^2 = \frac{1}{N} \sum_{k=1}^{N} \left(y_k - a_{\text{inv}} E\right)^2$$

General principle: In a χ^2 expression the measured values y_k should not be modified; instead the expectation has to take into account all known effects.

There are N data x_k with different standard deviations σ_k and a common relative normalisation error of ε. Apparently the mean value $\overline{y}$ can not be affected by the normalisation error, but its standard deviation is.
One method is to use the full covariance matrix for the correlated data, e.g. in the case $N = 2$:

$$\boldsymbol{V}_a = \begin{pmatrix} \sigma_1^2 & 0 \\ 0 & \sigma_2^2 \end{pmatrix} + \varepsilon^2 \cdot \begin{pmatrix} y_1^2 & y_1 y_2 \\ y_1 y_2 & y_2^2 \end{pmatrix} = \begin{pmatrix} \sigma_1^2 + \varepsilon^2 y_1^2 & \varepsilon^2 y_1 y_2 \\ \varepsilon^2 y_1 y_2 & \sigma_2^2 + \varepsilon^2 y_2^2 \end{pmatrix}$$

and minimising

$$\chi^2 = \boldsymbol{\Delta}^T \boldsymbol{V}^{-1} \boldsymbol{\Delta} \qquad \text{with} \quad \boldsymbol{\Delta} = \begin{pmatrix} y_1 - \overline{y} \\ y_2 - \overline{y} \end{pmatrix}$$

Example (from [**?**]): Data are
$y_1 = 8.0 \pm 2\%$ and $y_2 = 8.5 \pm 2\%$, with a common (relative) normalisation error of $\varepsilon = 10\%$.
The mean value resulting from χ^2 minimisation is:

$$7.87 \pm 0.81 \qquad \text{i.e.} \;\; < y_1 \text{ and } \; < y_2$$

- this is apparently wrong.

> . . . that including normalisation errors in the correlation matrix will produce a fit which is biased towards smaller values . . . [**?**]

> . . . the effect is a direct consequence of the hypothesis to estimate the empirical covariance matrix, namely the linearisation on which the usual error propagation relies. [**?**, **?**]

Origin of the apparent problem . . . the used covariance matrix!

The contribution to $\boldsymbol{V}$ from the normalisation error was calculated from the measured values, which were different; the result is a covariance ellipse with axis different from 45° and this produces a biased mean value.

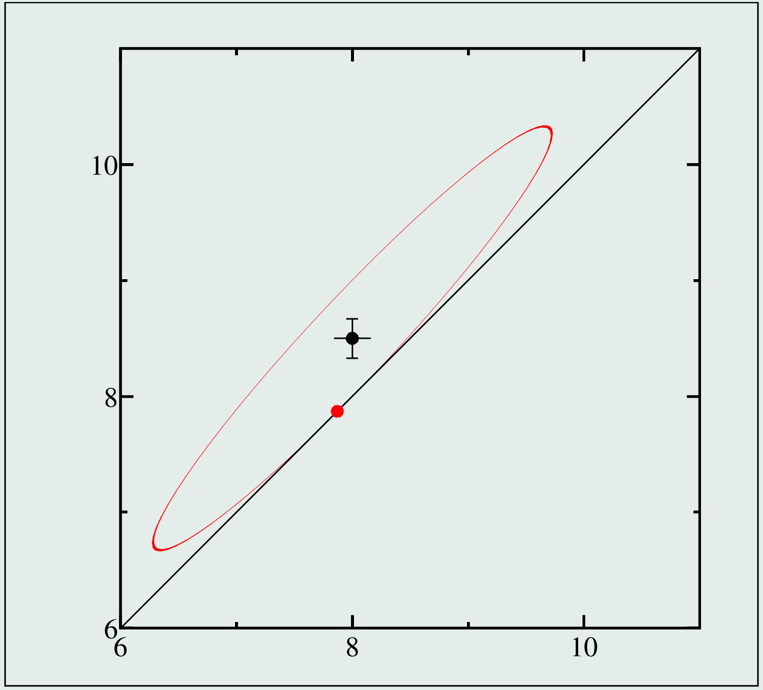
The correct model is: y_1 and y_2 have the same true value, then the normalisation errors $\varepsilon \cdot$value are identical, with

$$\boldsymbol{V}_b = \begin{pmatrix} \sigma_1^2 & 0 \\ 0 & \sigma_2^2 \end{pmatrix} + \varepsilon^2 \cdot \begin{pmatrix} \overline{y}^2 & \overline{y}^2 \\ \overline{y}^2 & \overline{y}^2 \end{pmatrix} = \begin{pmatrix} \sigma_1^2 + \varepsilon^2 \overline{y}^2 & \varepsilon^2 \overline{y}^2 \\ \varepsilon^2 \overline{y} & \sigma_2^2 + \varepsilon^2 \overline{y}^2 \end{pmatrix}$$

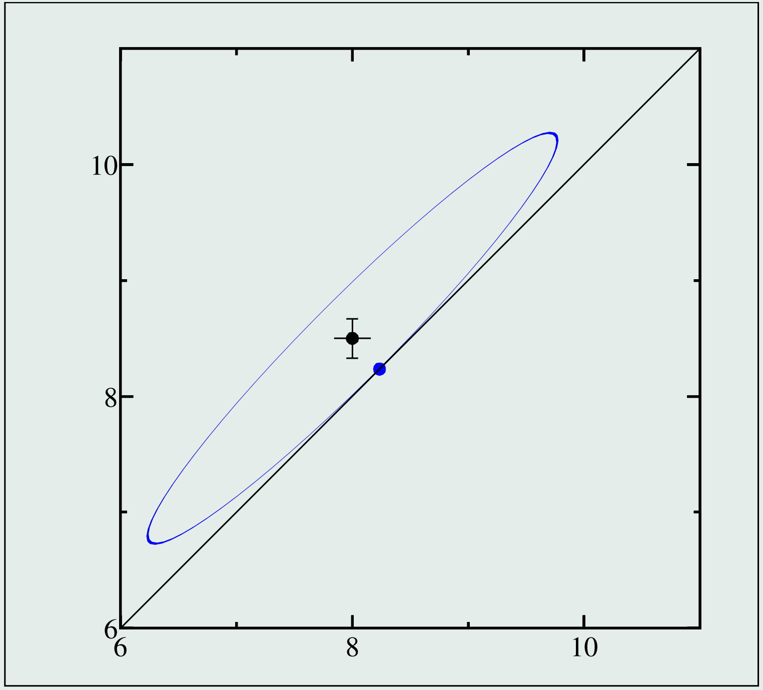
i.e. the covariance matrix depends on the resulting parameter.

Ellipses

Covariance ellipse for $\boldsymbol{V}_a$



Covariance ellipse for $\boldsymbol{V}_b$



Axis of ellipse is tilted w.r.t. the diagonal and ellipse touches the diagonal at a biased point.

Axis of the ellipse is ≈ 45° and ellipse touches the diagonal at the correct point.

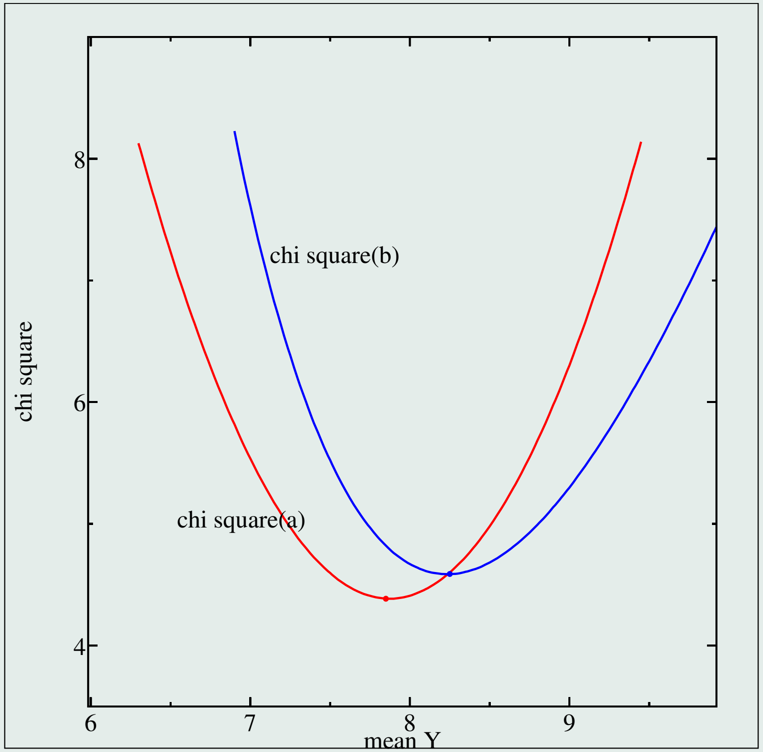
The result may depend critically on certain details of the model implementation.

The method with one additional parameter . . .

Another method often used is to define

$$\chi_a^2 = \sum_k \frac{(f \cdot y_k - \overline{y})^2}{\sigma_k^2} + \frac{(f-1)^2}{\varepsilon^2} ,$$

which will also produce a biased result.



The χ^2 definition for this problem

$$\chi_b^2 = \sum_k \frac{(y_k - f \cdot \overline{y})^2}{\sigma_k^2} + \frac{(f-1)^2}{\varepsilon^2}$$

will give the correct result (data unchanged and fitted value according to the model), as seen by blue curve.

Standard methods

Standard statistical methods for parameter determination are

- Method of Least Squares $S(\boldsymbol{a})$ • χ^2 minimisation is equivalent: $\chi^2 \equiv S(\boldsymbol{a})$
- Maximum Likelihood method $F(\boldsymbol{a})$
  …improves the parameter estimation if the detailed probability density is known.

Least squares and Maximum Likelihood can be combined, e.g

$$F_{\text{total}}(\boldsymbol{a}) = \frac{1}{2} S(\boldsymbol{a}) + F_{\text{special}}(\boldsymbol{a})$$

Doubts about justification of χ^2 minimisation from publications:

> The justification for using least squares lies in the assumption that the measurement errors are Gaussian distributed. [**?**]

> However it is doubtful that Gaussian errors are realistic.

> A bad χ^2 …Finally the data may very well not be Gaussian distributed.

The standard linear least squares method

The model of Linear Least Squares: $\boldsymbol{y} = \boldsymbol{A}\,\boldsymbol{a}$

$\boldsymbol{y} =$ measured data $\quad \boldsymbol{A} =$ matrix (fixed) $\quad \boldsymbol{a} =$ parameters $\quad \boldsymbol{V}_y =$ covariance matrix of $\boldsymbol{y}$

Least Squares Principle: minimize the expression $\quad (\boldsymbol{W} = \boldsymbol{V}_y^{-1})$

$$S(\boldsymbol{a}) = (\boldsymbol{y} - \boldsymbol{A}\boldsymbol{a})^T\ \boldsymbol{W}\ (\boldsymbol{y} - \boldsymbol{A}\boldsymbol{a}) \qquad \text{or} \quad F(\boldsymbol{a}) = \frac{1}{2}S(\boldsymbol{a})$$

with respect to $\boldsymbol{a}$.

Derivatives of expression $F(\boldsymbol{a})$:

$$\boldsymbol{g} = \frac{\partial F}{\partial \boldsymbol{a}} = -\boldsymbol{A}^T\boldsymbol{W}\boldsymbol{y} + \left(\boldsymbol{A}^T\boldsymbol{W}\boldsymbol{A}\right)\boldsymbol{a}$$

$$\boldsymbol{H} = \frac{\partial^2 F}{\partial a_j a_k} = \left(\boldsymbol{A}^T\boldsymbol{W}\boldsymbol{A}\right) \qquad = \text{constant}$$

Solution (from $\partial F/\partial \boldsymbol{a} = 0$) is linear transformation of the data vector $\boldsymbol{y}$:

$$\hat{\boldsymbol{a}} = \left[\left(\boldsymbol{A}^T\boldsymbol{W}\boldsymbol{A}\right)^{-1}\ \boldsymbol{A}^T\boldsymbol{W}\right]\boldsymbol{y} \qquad = \boldsymbol{B}\,\boldsymbol{y}$$

Covariance matrix of $\boldsymbol{a}$ by "error" propagation

$$\boldsymbol{V}[\hat{\boldsymbol{a}}] = \boldsymbol{B}\,\boldsymbol{V}[\boldsymbol{y}]\,\boldsymbol{B}^T = \left(\boldsymbol{A}^T\boldsymbol{W}\boldsymbol{A}\right)^{-1} \qquad = \text{inverse of } \boldsymbol{H}$$

Properties of the solution

Starting from **Principles** properties of the solution are derived, which are valid under certain conditions:

- Data are unbiased: $E[\boldsymbol{y}] = \boldsymbol{A}\,\bar{\boldsymbol{a}}$ $\qquad$ ($\bar{\boldsymbol{a}} =$ true parameter vector)
- Covariance matrix $\boldsymbol{V}_y$ of the data is known (and correct).

Distribution-free properties of least squares estimates in linear problems are:

- Estimated parameters are unbiased:

$$E[\hat{\boldsymbol{a}}] = \left(\boldsymbol{A}^T\boldsymbol{W}\boldsymbol{A}\right)^{-1}\,\boldsymbol{A}^T\boldsymbol{W}\,E[\boldsymbol{y}] = \bar{\boldsymbol{a}}$$

- In the class of unbiased estimates, which are linear in the data, the Least Squares estimates $\hat{\boldsymbol{a}}$ have the smallest variance (Gauß-Markoff theorem).
- The expectation of the sum of squares of the residuals is $\hat{S} = (n-p)$.

Special case of Gaussian distributed measurement errors:

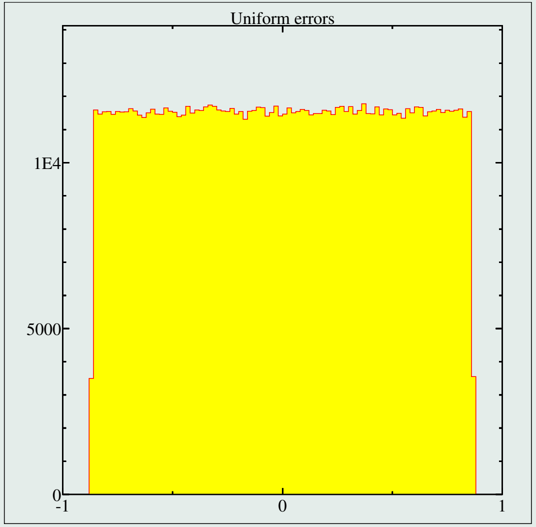
$$\hat{S}/\sigma^2 \quad \text{distributed according to the} \quad \chi^2_{n-p} \quad \text{distribution}$$

to be used for goodness-of-fit test. Properties are not valid, if conditions violated.

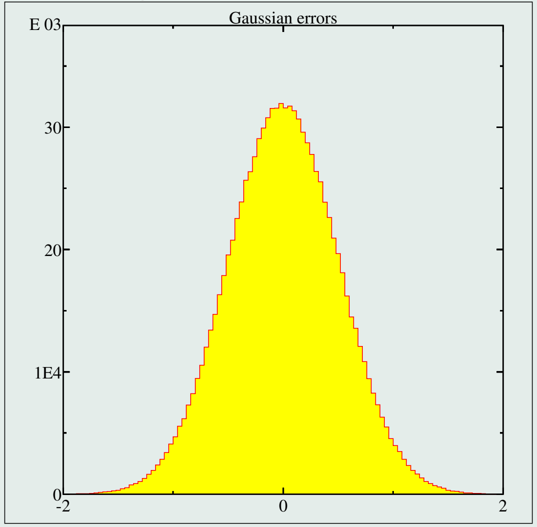
Test of non-Gaussian data

MC test of least squares fit of 20 data points to straight line (two parameters), generated with data errors from different distributions, but always mean = 0 and same standard deviation $\sigma = 0.5$.

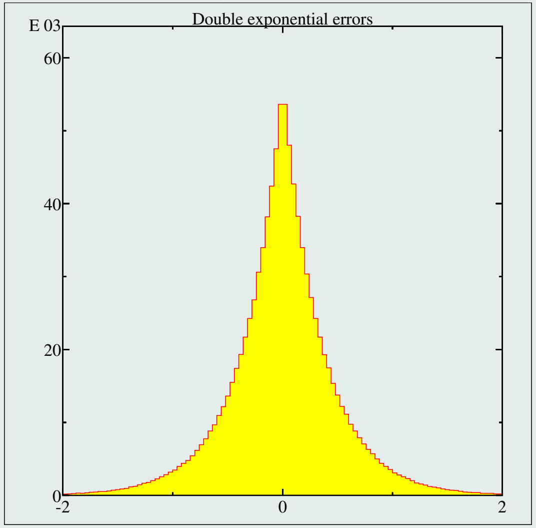
uniform errors



Gaussian errors

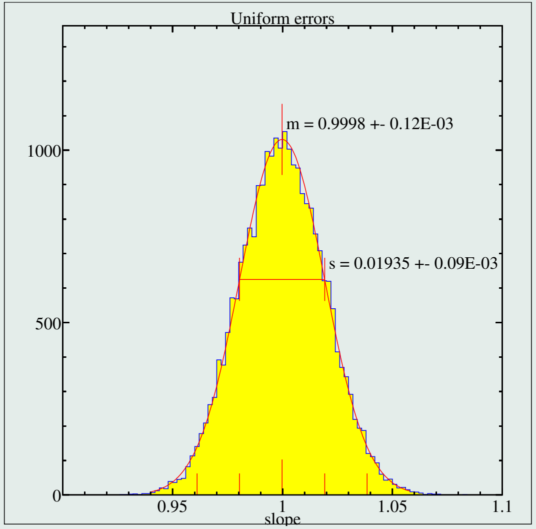


double exponential errors



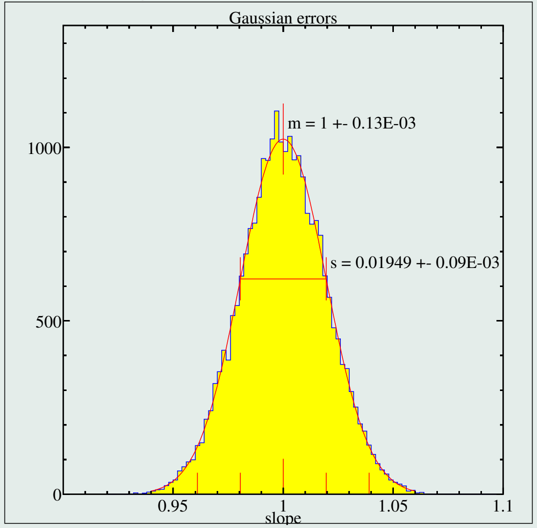
Results for slope parameters 25000 entries

uniform errors



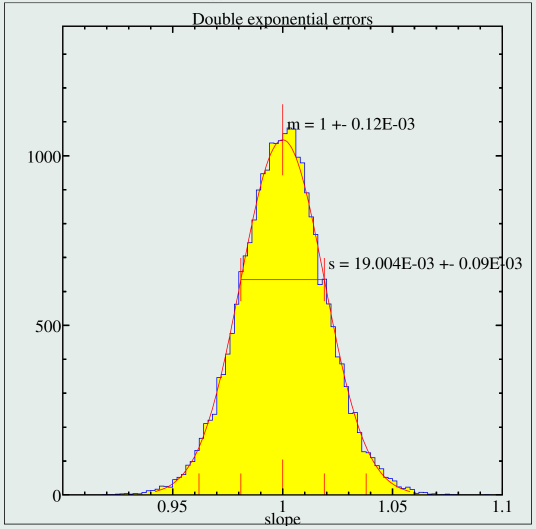
$\sigma = 0.0194$

Gaussian errors



$\sigma = 0.0195$

double exponential errors



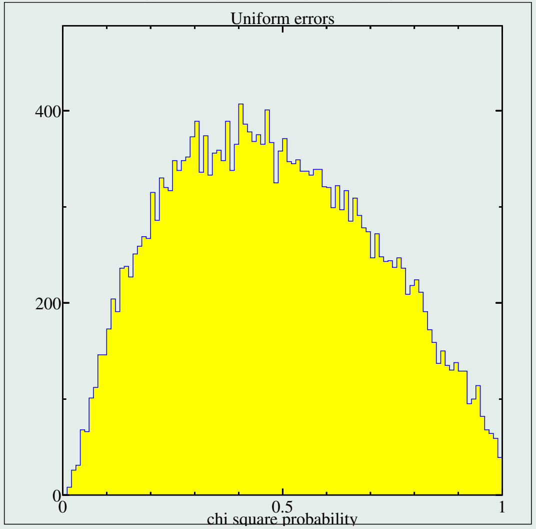
$\sigma = 0.0190$

- All parameter distributions are Gaussian, and of the width, expected from the standard error calculation.
- This is valid for both fitted parameters.

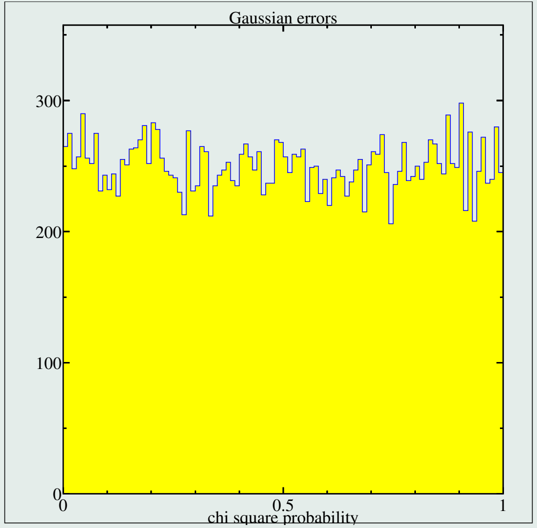
χ^2 and χ^2-probability **25000 entries**

- Mean χ^2-values are all equal to $n_{\mathrm{df}} = 20 - 2 = 18$, as expected, but
- χ^2-probabilities have different distributions, as expected.

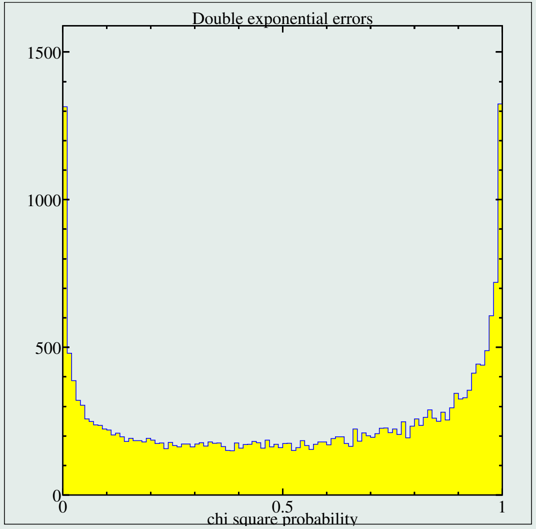
uniform errors



Gaussian errors



double exponential errors



Conclusion: Least squares works fine and as expected, also for non-Gaussian data, if . . . and only if

- data are unbiased and covariance matrix is complete and correct.

Orthogonal polynomials

If data interpolation required, but no parametrization known: fit of normal general p-th polynomial to the data

$$f(x_i) \cong y_i \qquad \text{with} \quad f(x) = \sum_{j=0}^{p} a_j \, x^j$$

Generalization of straight-line fit to general p-th order polynomial straightforward: matrix $\boldsymbol{A}^T \boldsymbol{W} \boldsymbol{A}$ contains sum of all powers of x_i up to x_i^{2p}.

Disadvantage:

- fit numerically unstable
- determination of optimal order p difficult
- value of coefficients a_j depend on highest exponent p

Better: fit of orthogonal polynomial.

Straight line fit

Parametrization

$$f(x) = a_0 + a_1 x \quad \text{replaced by} \quad f(x) = a_0 \cdot p_0(x) + a_1 \cdot p_1(x)$$

The orthogonal polynomials $p_0(x)$ and $p_1(x)$ are defined by

$$\begin{aligned} p_0(x) &= b_{00} \\ p_1(x) &= b_{10} + b_{11} x = b_{11} \left(x - \langle x \rangle \right) \quad . \end{aligned}$$

with coefficients b_{00} and b_{11}

$$b_{00} = \left(\sum_{i=1}^{n} w_i \right)^{-1/2} \qquad b_{11} = \left(\sum_{i=1}^{n} w_i \left(x_i - \langle x \rangle \right)^2 \right)^{-1/2}$$

with the coefficient $b_{10} = -\langle x \rangle \, b_{11}$.

Advantage of new ansatz

$$\boldsymbol{A}^T\boldsymbol{W}\boldsymbol{A} = \begin{pmatrix} \sum w_i p_0^2(x_i) & \sum w_i p_0(x_i) p_1(x_i) \\ \sum w_i p_0(x_i) p_1(x_i) & \sum w_i p_1^2(x_i) \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

$$\boldsymbol{A}^T\boldsymbol{W}\boldsymbol{y} = \begin{pmatrix} \sum w_i p_0(x_i) y_i \\ \sum w_i p_1(x_i) y_i \end{pmatrix} = \begin{pmatrix} a_0 \\ a_1 \end{pmatrix}$$

i.e. covariance matrix is unit matrix and parameters $\boldsymbol{a}$ are calculated by sums (no matrix inversion).

General orthogonal polynomial:

$$f(x) = \sum_{j=0}^{p} a_j p_j(x) \qquad \text{with} \quad \boldsymbol{A}^T\boldsymbol{W}\boldsymbol{A} = \quad \text{unit matrix}$$

. . . means construction of orthogonal polynomial from (for) the data!

Recurrence relation for $p_j(x)$

Construction of higher order polynomial $p_j(x)$ by recurrence relation:

$$\boxed{\gamma p_j(x) = (x_i - \alpha)\, p_{j-1}(x) - \beta p_{j-2}(x)}$$

from the two previous functions $p_{j-1}(x)$ and $p_{j-2}(x)$ with parameters α and β defined by

$$\alpha = \sum_{i=1}^{n} w_i x_i p_{j-1}^2(x_i) \qquad \beta = \sum_{i=1}^{n} w_i x_i p_{j-1}(x_i) p_{j-2}(x_i)$$

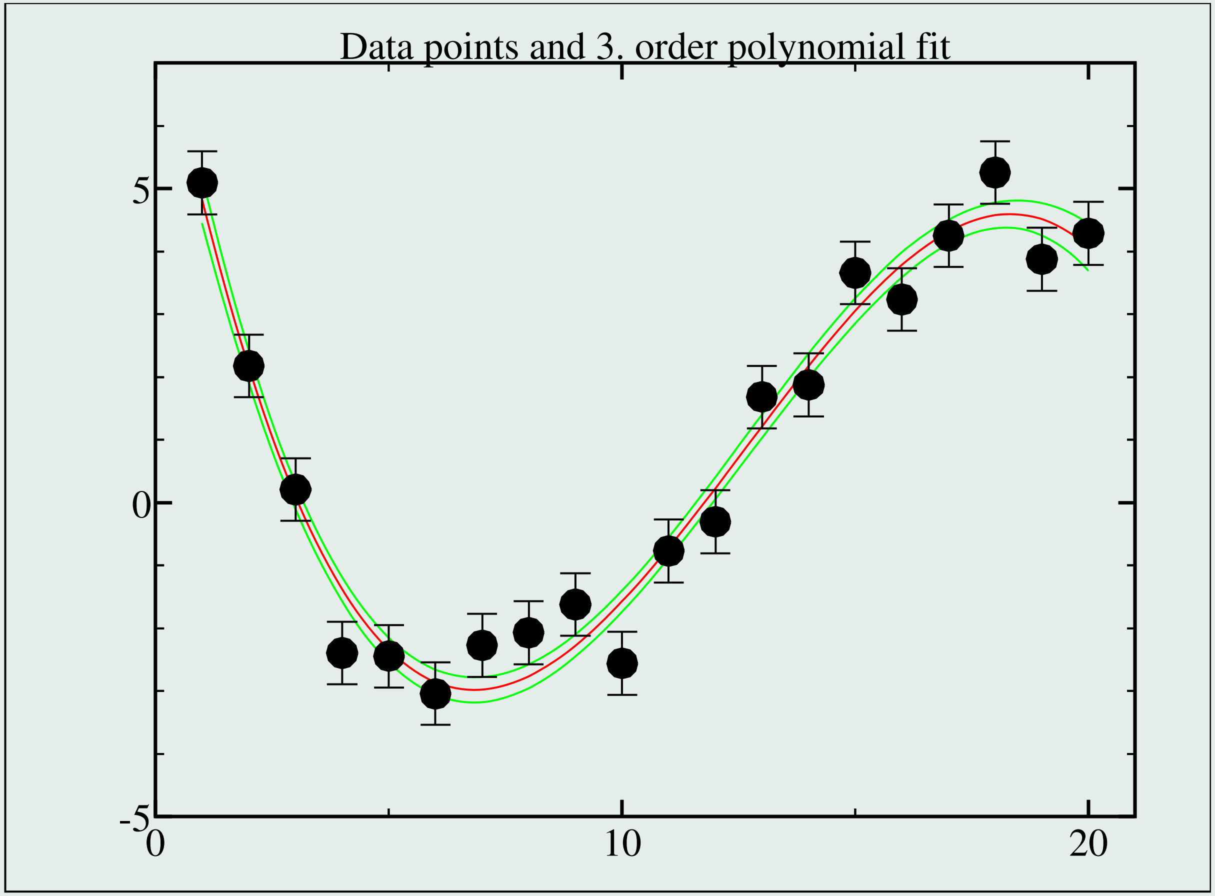
Normalization factor γ given by

$$\gamma^2 = \sum_{i=1}^{n} w_i \left[(x_i - \alpha)\, f_{j-1}\left(x_i\right) - \beta f_{j-2}\left(x_i\right) \right]^2$$

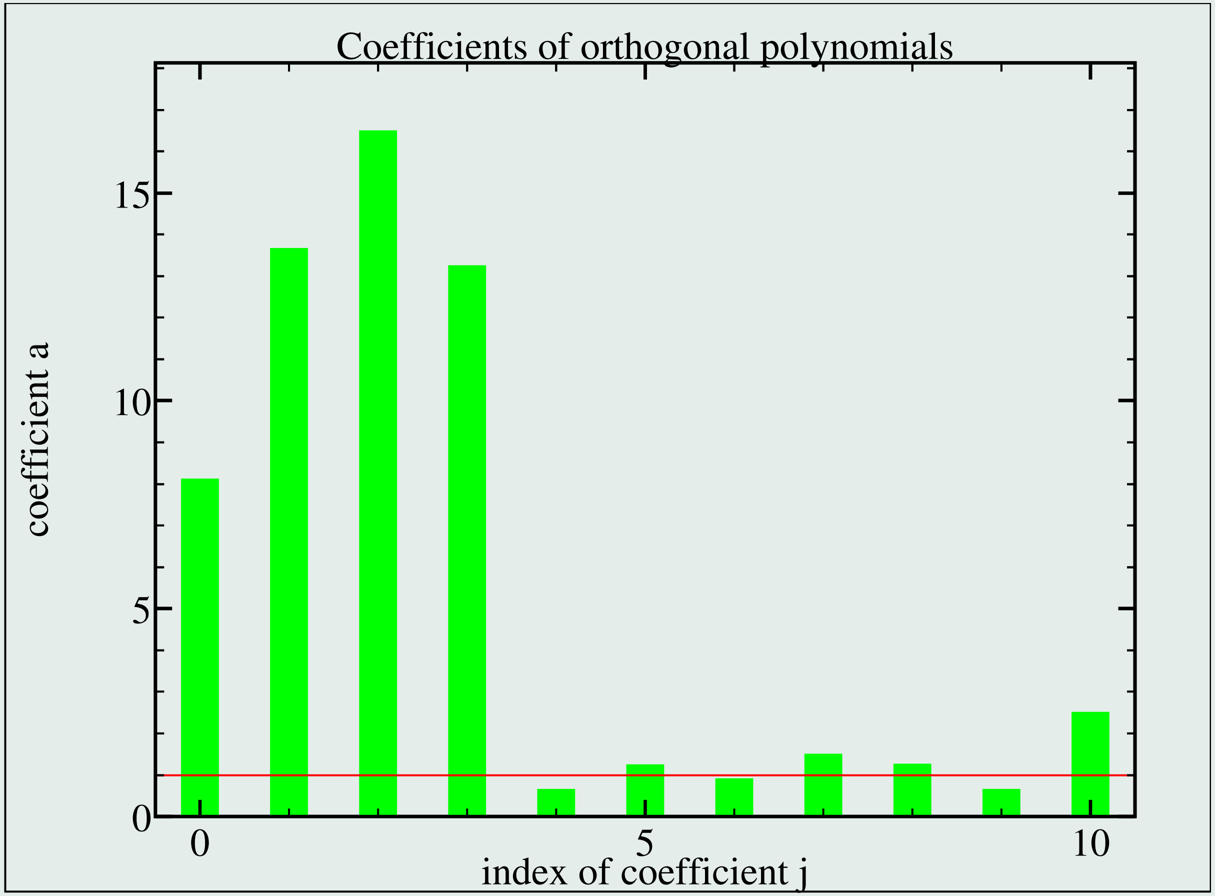
Parameters $\widehat{a}_j$ are determined from data by

$$\widehat{a}_j = \sum_{i=1}^{n} w_i p_j(x_i)\; y_i$$

Third order polynomial fit



Spectrum of coefficients

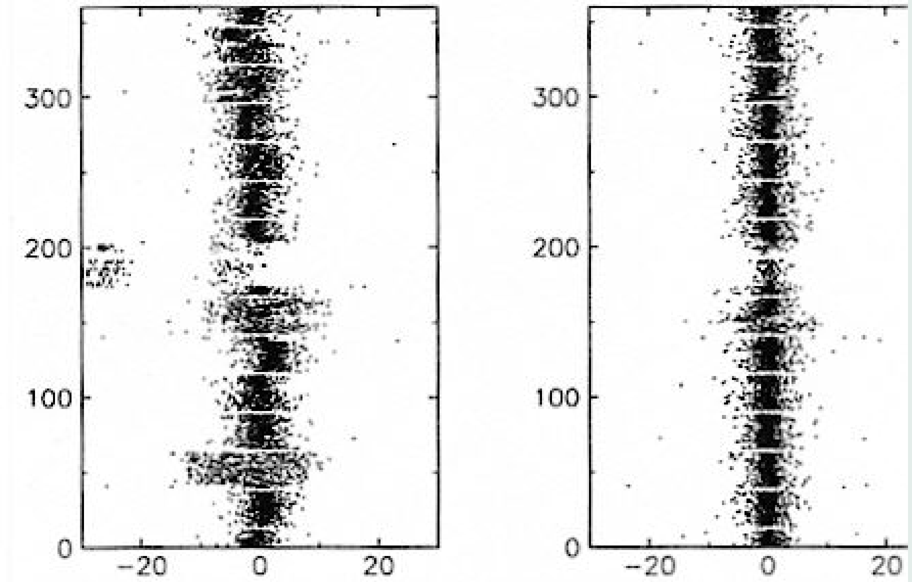


Example: High precision aligment of a track detector

- Motivation for alignment
- Solution by partitioning
- Simultaneous fit of global and local parameters
- Example: MC
- Example: Alignment of a central track detectors (H1)

(a) Motivation for alignment

Example: Residuals of track fit versus φ for a silicon tracker before … and after alignment:



Aim:

- smaller residuals!
- more accurate track parameters!!!

A simple method

An approximate alignment method:

- perform least squares fits on the e.g.track data *ignoring* initially the alignment parameters
- the *deviations* between the fitted and measured data, the *residuals*, are used to determine the alignment parameters afterwards.

The result of applying the alignment parameters on e.g. fits on the track data

- improved track residuals
- same track parameter values as before.

…because least squares alignment fits were based on a wrong track model.
The alignment problem is more general: in addition to pure alignment parameters there are drift velocities, variations of drift velocities, Lorentz angle …, which can not determined with residuals.

A better method?

The calculation of space coordinates from electronic signals requires alignment and many other parameters.

A better idea: perform a simultaneous least squares fit to the parameters of e.g. 20 000 tracks (each track has 50 hits and is described by three parameters) and all (e.g. 1000) alignment parameters i.e. solve normal equations $\boldsymbol{C}\boldsymbol{a} = \boldsymbol{b}$
Add all necessary constraints e.g. zero average displacement and rotation of the detector.

Now the track model is correct, but …

- $3\times$ 20 000 + 1000 parameters = 61 000 parameters
- 20 000 $\times$50 hits = 1 Mio hits

Can such a simultanenous fit be performed on a standard PC?

Constraints

Explicit relations between the parameters have to be taken into account e.g. zero average displacement and zero rotation of the whole detector.

Case of a single linear constraint between parameters:

$$\boldsymbol{f}^T \cdot \boldsymbol{a} = f_0$$

Modification of system of equations by Lagrange multiplier (λ) method:
add equation $\lambda\left(\boldsymbol{f}^T \cdot \boldsymbol{a} - f_0\right)$ to system of equations

$$\left(\begin{array}{c|c} \boldsymbol{C} & \boldsymbol{f} \\ \hline \boldsymbol{f}^T & \boldsymbol{0} \end{array}\right) \left(\begin{array}{c} \boldsymbol{a} \\ \hline \lambda \end{array}\right) = \left(\begin{array}{c} \boldsymbol{b} \\ \hline f_0 \end{array}\right) ,$$

Each constraint adds another parameter and equation to the whole system.

(b) Solution by partitioning

Structure of the matrix equation: $\boldsymbol{C}\boldsymbol{a} = \boldsymbol{b}$

$$\text{or} \qquad \left(\begin{array}{c|c} \boldsymbol{C}_{11} & \boldsymbol{C}_{12} \\ \hline \boldsymbol{C}_{21} & \boldsymbol{C}_{22} \end{array} \right) \left(\begin{array}{c} \boldsymbol{a}_1 \\ \hline \boldsymbol{a}_2 \end{array} \right) = \left(\begin{array}{c} \boldsymbol{b}_1 \\ \hline \boldsymbol{b}_2 \end{array} \right)$$

where the submatrix $\boldsymbol{C}_{11}$ is a p-by-p square matrix and the submatrix $\boldsymbol{C}_{22}$ is a q-by-q square matrix, with $p + q = n$.

If the sub-vector $\boldsymbol{a}_1$ would not exist:

$$\boldsymbol{C}_{22}\, \boldsymbol{a}_2^* = \boldsymbol{b}_2 \qquad\qquad \boldsymbol{a}_2^* = \boldsymbol{C}_{22}^{-1}\, \boldsymbol{b}_2$$

The solution $\boldsymbol{a}_2^*$ requires only inversion of $\boldsymbol{C}_{22}$ and is called the *local* solution.

Solution by partitioning contnd.

Submatrix $\boldsymbol{B}$ of the complete inverse matrix correponding to the upper left part $\boldsymbol{C}_{11}$ is easy to calculate afterwards:

$$\boldsymbol{B} = \left(\boldsymbol{C}_{11} - \boldsymbol{C}_{12}\boldsymbol{C}_{22}^{-1}\boldsymbol{C}_{12}^T\right)^{-1}$$

Complete inverse matrix equation in terms of $\boldsymbol{B}$:

$$\left(\begin{array}{c} \boldsymbol{a}_1 \\ \hline \boldsymbol{a}_2 \end{array}\right) = \left(\begin{array}{c|c} \boldsymbol{B} & -\boldsymbol{B}\boldsymbol{C}_{12}\boldsymbol{C}_{22}^{-1} \\ \hline -\boldsymbol{C}_{22}^{-1}\boldsymbol{C}_{12}^T\boldsymbol{B} & \boldsymbol{C}_{22}^{-1} - \boldsymbol{C}_{22}^{-1}\boldsymbol{C}_{12}^T\boldsymbol{B}\boldsymbol{C}_{12}\boldsymbol{C}_{22}^{-1} \end{array}\right) \left(\begin{array}{c} \boldsymbol{b}_1 \\ \hline \boldsymbol{b}_2 \end{array}\right)$$

Solution for subvector: $\boldsymbol{a}_1 = \boldsymbol{B}\boldsymbol{b}_1 - \boldsymbol{B}\boldsymbol{C}_{12}\underbrace{\boldsymbol{C}_{22}^{-1}\boldsymbol{b}_2}_{\boldsymbol{a}_2^*} = \boldsymbol{B}\left(\boldsymbol{b}_1 - \boldsymbol{C}_{12}\boldsymbol{a}_2^*\right)$

(c) Simultaneous fit of global and local parameters

Local parameters α_j are e.g. direct track parameters (belonging to a single track). For measurement z_k with (known) constant factors δ_j:

$$z_k = \alpha_1 \cdot \delta_{1k} + \alpha_2 \cdot \delta_{2k} + \ldots \alpha_\nu \cdot \delta_{\nu k} = \sum_{j=1}^{\nu} \alpha_j \cdot \delta_{jk}$$

Normal equation of linear least squares:

$$\boldsymbol{\Gamma}\,\boldsymbol{\alpha} = \boldsymbol{\beta}\,, \qquad \text{solved by} \quad \boldsymbol{\alpha} = \boldsymbol{\Gamma}^{-1}\boldsymbol{\beta}$$

Global parameters are e.g. alignment parameters and contribute to all the measurements. They represent corrections to ideal (design) values.

$$z = \underbrace{a_1 \cdot d_1 + a_2 \cdot d_2 + \ldots a_n \cdot d_n}_{\text{global parameters}} + \underbrace{\alpha_1 \cdot \delta_1 + \alpha_2 \cdot \delta_2 + \ldots \alpha_\nu \cdot \delta_\nu}_{\text{local parameters}}$$

Complete matrix equation for global and local parameters . . .

. . . has huge matrices: n-by-n matrices $\boldsymbol{C}$ for n global parameters + N of small ν-by-ν matrices $\boldsymbol{\Gamma}$.

$$\left(\begin{array}{c||c|c|c} \sum \boldsymbol{C}_i & \cdots & \boldsymbol{G}_i & \cdots \\ \hline\hline \vdots & \ddots & 0 & 0 \\ \hline \boldsymbol{G}_i^T & 0 & \boldsymbol{\Gamma}_i & 0 \\ \hline \vdots & 0 & 0 & \ddots \end{array}\right) \cdot \left(\begin{array}{c} \boldsymbol{a} \\ \hline\hline \vdots \\ \hline \boldsymbol{\alpha}_i \\ \hline \vdots \end{array}\right) = . \left(\begin{array}{c} \sum \boldsymbol{b}_i \\ \hline\hline \vdots \\ \hline \boldsymbol{\beta}_i \\ \hline \vdots \end{array}\right)$$

Example: 3.7×10^9 matrix elements for $n = 1000$, $\nu = 3$ and $N = 20\ 000$.

How to solve this linear system of equations?

The matrix of normal equations has a special structure with many vanishing sub-matrices. Only the global parameters $\boldsymbol{a}$ have to be calculated.

Use N times the partitioning method for the N sets of local parameters:

$$\boldsymbol{\alpha}_i^* = \boldsymbol{\Gamma}_i^{-1} \boldsymbol{\beta}_i$$

Finally the n normal equations

$$\left(\quad \boldsymbol{C}' \quad \right) \left(\boldsymbol{a} \right) = \left(\boldsymbol{b}' \right)$$

are obtained with modified matrix and vector:

$$\boldsymbol{C}' = \sum_i \boldsymbol{C}_i - \sum_i \boldsymbol{G}_i \boldsymbol{\Gamma}_i^{-1} \boldsymbol{G}_i^T \qquad \boldsymbol{b}' = \sum_i \boldsymbol{b}_i - \sum_i \boldsymbol{G}_i \left(\boldsymbol{\Gamma}_i^{-1} \boldsymbol{\beta}_i \right)$$

The solution $\boldsymbol{a}$ is the complete solution.

Solution of final system

The solution requires no iterations. Even the case $n = 5000$ will not take much time (but: use special solution method, where undetermined parameters are set to zero).

Iterations may be necessary for other reasons, namely

- the equations depend *non-linearly* on the global parameters; the equations have to be linearized;
- the data contain outliers, which have to be removed in a sequence of cuts, becoming narrower during the iteration;
- the accuracy of the data is not known before, and has to be determined from the data (after the alignment).

Method realized in the program **Millepede**. It provides a set of subroutines for the mathematical methods and allows to adapt the method to a particular problem with introduction of linear constraints.

(d) Monte Carlo simulation

- Track detector with ten planes is simulated by Monte Carlo methods.
- Straight tracks hit the ten planes and allow to measure one coordinate z per plane transverse to the axis of the detector, with an efficiency around 90 %.
- Each plane may be displaced perpendicular to the detector axis.

The parametrization of the track in terms of the local parameters is

$$z = \alpha_1 + \alpha_2 \, x$$

Three methods are used:

- Millepede with constraints
- Millepede without constraints
- Simple residual method

Result of Millepede in alignment simulation

Δ simulated shift of detector plane

Δ_f Millepede result with constraints: no overall rotation

Δ_n Millepede result without constraints

Δ_r result of simple residual method

σ [cm]	Δ [cm]	Δ_f	Δ_n	Δ_r
0.0020	0.0000	0.0000 ± 0.0000	0.0000 ± 0.0000	-0.0274
0.0020	-0.0400	-0.0399 ± 0.0001	-0.0407 ± 0.0012	-0.0636
0.0300	0.1500	0.1495 ± 0.0009	0.1465 ± 0.0047	0.1269
0.0300	0.0300	0.0293 ± 0.0009	0.0252 ± 0.0062	0.0099
0.0300	-0.0750	-0.0755 ± 0.0012	-0.0806 ± 0.0078	-0.0921
0.0300	0.0450	0.0483 ± 0.0010	0.0423 ± 0.0093	0.0346
0.0300	0.0350	0.0342 ± 0.0010	0.0271 ± 0.0108	0.0216
0.0300	-0.0800	-0.0814 ± 0.0010	-0.0895 ± 0.0123	-0.0954
0.0300	0.0900	0.0902 ± 0.0011	0.0811 ± 0.0139	0.0840
0.0300	-0.0500	-0.0494 ± 0.0011	-0.0595 ± 0.0154	-0.0513

(e) Alignment of a track detector

Alignment in the $r\varphi$-plane (perpendicular to the beam) of a 56-plane drift chamber and a 2-plane silicon vertex detector in the H1 detector

Components:

- drift chamber has a length (z) of about 2 m, and extending from 20.3 cm to 84.4 cm in radius r – resolution $\sigma = 150\mu$m
- silicon vertex detector with two planes around the beampipe – resolution $\sigma = 15\mu$m

Alignment fit ...

. . . using ten thousands of track from *ep*-data, from cosmics with and without B-field.

Large number of correction parameters with small values:

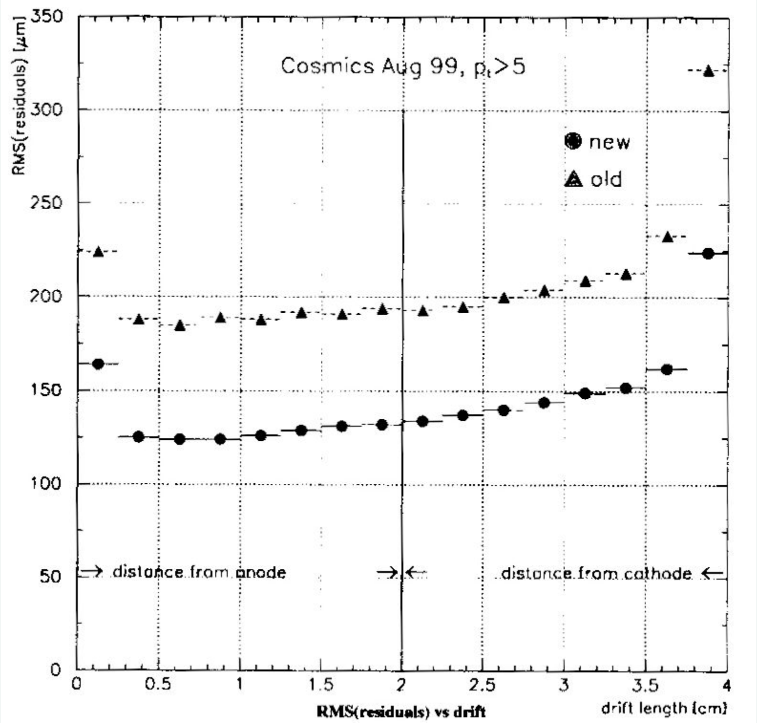
- geometrical shifts
- drift velocity and (relative) velocity corrections
- Lorentz angle corrections
- time corrections
- special correction for distortions of E field due to bad HV
- drift velocity change by electron beam current
- . . .

Table of parameters and precision

row.	number	parameter	σ	unit
1	2	Δx	1	μm
2	2	$\Delta x/\Delta z_r$	2	μm
3	2	Δy	1	μm
4	2	$\Delta y/\Delta z_r$	2	μm
5	2	$\Delta\varphi$	10	μrad
6	2	$\Delta\varphi/\Delta z_r$	10	μrad
7	2	$\Delta\alpha_{\text{Lor}}$	100	μrad
8	2	$\Delta v_{\text{drift}-}/v_{\text{drift}}$	10^{-5}	
9	2	$\Delta v_{\text{drift}+}/v_{\text{drift}}$	10^{-5}	
10	2	$\Delta T_0 \times v_{\text{drift}}$	< 1	μm
11	2	wire staggering in wire plane	few	μm
12	2	wire staggering perp wire plane	few	μm
13	2	sagging in wire plane	few	μm
14	2	sagging perp. wire plane	few	μm
15	180	$\Delta v_{\text{drift}}/v_{\text{drift}}$ per cell half	few	10^{-4}
16	112	$\Delta v_{\text{drift}}/v_{\text{drift}}$ per layer half	few	10^{-4}
17	330	$\Delta T_0 \times v_{\text{drift}}$ per group	10	μm
18	56	wire position in driftdir. per layer	10	μm
19	56	$\Delta T_0 \times v_{\text{drift}}$ per layer	10	μm
20	56	wire pos. perp. driftdir. per layer	few 10	μm
21	112	$\Delta v_{\text{drift}}/v_{\text{drift}}$ for $I_e/50$ mA	few 10^{-4}	
22	90	$\Delta v_{\text{drift}}/v_{\text{drift}}$ per layer	few 10^{-4}	
23	90	Δy_W per layer	few 10	μm
24	90	Δy_W per layer^2	few 10	μm
25	64	Δ in ladder	few	μm
26	64	Δ perp. ladder	few	μm
27	64	rel. Δ in ladder (z_r)	few	μm
28	64	rel. Δ perp. ladder (z_r)	10	μm
29	64	rel. Δ perp. ladder (φ)	few	μm

Improvement of drift chamber hit resolution

Mean track residuals from the pure drift chamber fit as function of the drift length – before and after improved alignment



→ improvement from $\sigma = 180\ \mu\text{m}$ to $\sigma = 125\ \mu\text{m}$

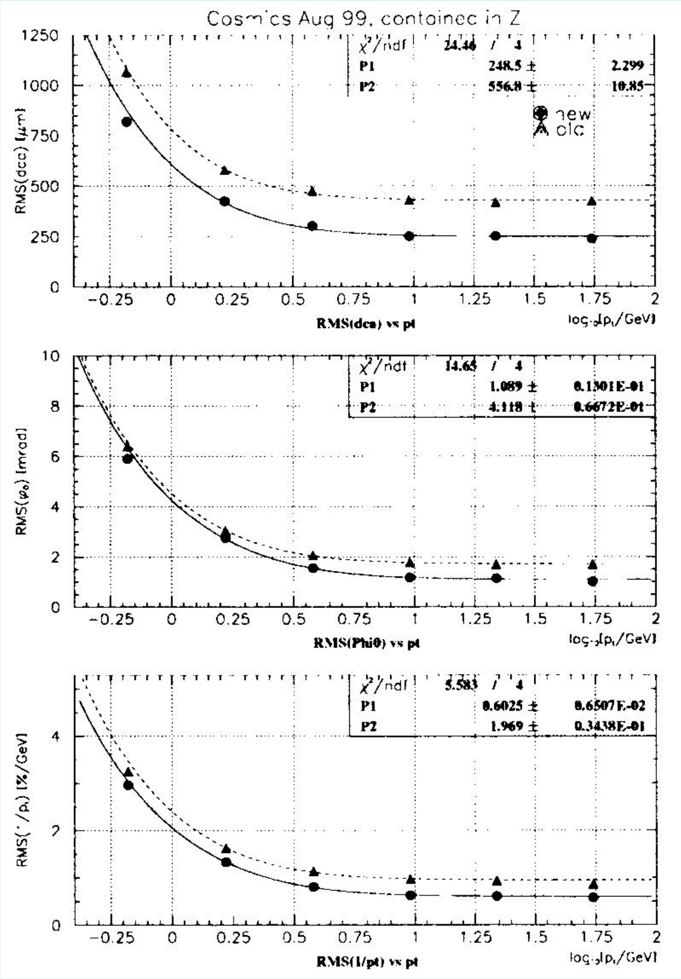
Improvement of track parameter accuracy

Standard deviation of track parameters as a function of $\log_{10} p_t/\text{GeV}$:

top distance of closest approach to center

middle azimuthal angle

bottom inverse transverse momentum



Difference of two parts of cosmic muon track

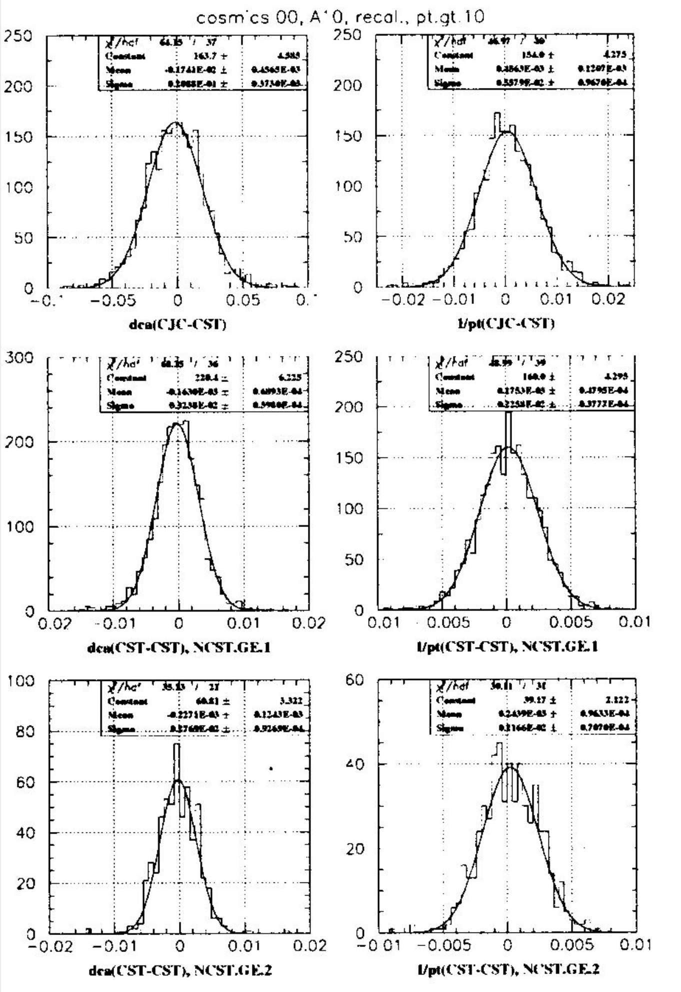
Residual distribution, calculated from two parts of cosmic muons

- for d_{ca} (distance of closeest approach to center) [left]
- for $1/p_t$ (inverse transverse momentum) [right]

top fit without silicon tracker (drift chamber resolution)

middle with at least 1 silicon tracker hit (accuracy due to silicon tracker)

bottom with at least 2 silicon tracker hits



Summary

Method allows accurate determination of detector parameters and can even run on a PC.

Experience shows:

- it is important to use different (all available) data sets
- it is essential to make a simultaneous alignment of all track detector components, which are also used for the measurement of tracks
- independent internal alignment of single track detector components may not lead to a good overall result
- introduce constraints or fix certain global parameters to get stable results

A warning: do not print the whole covariance matrix of the detector parameters!

A program description for **Millepede** and the code is available via `http:/www.desy.de/~blobel/`.

Nonlinear least squares

In practice quite often the function $f(x, \boldsymbol{a})$ for the expectation of the measured values y depends nonlinearly on the parameters a_j.

Example: $f(x, \boldsymbol{a}) = a_1 \cdot \exp(a_2 x)$ both 1. derivatives non-constant)

$$(a+\varepsilon)^2 \approx a^2 + 2a\varepsilon \qquad (a+\varepsilon)^n \approx a^n + na^{n-1}\varepsilon$$

$$\frac{1}{a+\varepsilon} \approx \frac{1}{a} - \frac{\varepsilon}{a^2} \qquad \sqrt{a+\varepsilon} \approx \sqrt{a} + \frac{\varepsilon}{2\sqrt{a}}$$

$$e^{a+\varepsilon} \approx e^a\left(1+\varepsilon\right) \qquad \ln\left(a+\varepsilon\right) \approx \ln a + \frac{\varepsilon}{a}$$

$$\sin\left(a+\varepsilon\right) \approx \sin a + \varepsilon \cos a \qquad \cos\left(a+\varepsilon\right) \approx \cos a - \varepsilon \sin a$$

$$\tan\left(a+\varepsilon\right) \approx \tan a + \frac{\varepsilon}{\cos^2 a}$$

Linearization formulas, showing the *linear* change of a function for a small change of the argument.

Linearization

Linearization of the function $f(x, \boldsymbol{a})$ requires reasonable starting values for the parameters $\boldsymbol{a}$, these are denoted by $\boldsymbol{a}^*$. The function $f(x, \boldsymbol{a})$ is replaced by a (linear) Taylor expansion,

$$f(x, a) \approx f(x, \boldsymbol{a}^*) + \sum_{j=1}^{p} \frac{\partial f}{\partial a_j} (a_j - a_j^*),$$

where the partial derivatives are taken at $\boldsymbol{a}^*$,

$$\boldsymbol{r} = \boldsymbol{y} - \boldsymbol{A \Delta a} - \boldsymbol{f}$$

$$\boldsymbol{A} = \begin{pmatrix} \partial f(x_1)/\partial a_1 & \partial f(x_1)/\partial a_2 & \ldots & \partial f(x_1)/\partial a_p \\ \partial f(x_2)/\partial a_1 & \partial f(x_2)/\partial a_2 & \ldots & \partial f(x_2)/\partial a_p \\ & & & \\ \ldots & & & \\ \partial f(x_n)/\partial a_1 & \partial f(x_n)/\partial a_2 & & \partial f(x_n)/\partial a_p \end{pmatrix}$$

Solution of one iteration

From the linearized problem

$$S = \boldsymbol{r}^T \boldsymbol{W} \boldsymbol{r} = (\boldsymbol{y} - \boldsymbol{A}\Delta\boldsymbol{a} - \boldsymbol{f})^T \boldsymbol{W} (\boldsymbol{y} - \boldsymbol{A}\Delta\boldsymbol{a} - \boldsymbol{f}-) = \min$$

the normal equations

$$(\boldsymbol{A}^T \boldsymbol{W} \boldsymbol{A}) \boldsymbol{\Delta a} = \boldsymbol{A}^T \boldsymbol{W} (\boldsymbol{y} - \boldsymbol{f})$$

follow,

which are solved by

$$\boldsymbol{\Delta a} = (\boldsymbol{A}^T \boldsymbol{W} \boldsymbol{A})^{-1} \left[\boldsymbol{A}^T \boldsymbol{W} (\boldsymbol{y} - \boldsymbol{f}) \right] .$$

Converging?

$$S(\boldsymbol{a}^* + \Delta\boldsymbol{a}) \leq S(\boldsymbol{a}^*) \qquad \text{is required}$$

It can be shown, that there is a certain λ, such that

$$S(\boldsymbol{a}^* + \tau\boldsymbol{\Delta a}) \quad ,$$

considered as a function of τ, is a monotonically decreasing function for $0 \leq \tau \leq \lambda$, in particular

$$S(\boldsymbol{a}^* + \lambda\Delta\boldsymbol{a}) < S(\boldsymbol{a}^*) \quad .$$

$$\left.\frac{\partial S}{\partial \tau}\right|_{\tau=0} = -2\,\Delta\boldsymbol{a}^T \left(\boldsymbol{A}^T\boldsymbol{W}\boldsymbol{A}\right)\Delta\boldsymbol{a} < 0\ ,,$$

(if $\boldsymbol{A}^T\boldsymbol{W}\boldsymbol{A}$ is positive definite), because the expression in parentheses is a quadratic form. From continuity, there exists a $\lambda > 0$ satisfying

$$\frac{\partial S}{\partial \tau} < 0 \quad \text{for} \quad 0 \leq \tau \leq \lambda$$

Divergent iteration what to do?

A simple method in cases, where after solution of the linearized problem the new value of S, $S(\boldsymbol{a}^* + \Delta \boldsymbol{a})$ is larger than $S(\boldsymbol{a}^*)$:

$\Delta \boldsymbol{a}$ is reduced by some factor, say 1/2, and the test is repeated, until a value smaller than $S(\boldsymbol{a}^*)$ is reached.

A better (more stable and faster) method: use optimal numerical method to minimize the one-dimensional function

$$f(\tau) = S(\boldsymbol{a}^* + \tau \cdot \Delta \boldsymbol{a})$$

w.r.t. the argument τ.

(see chapter on Function minimization.)

Recognition of convergence

The scalar product of the vectors $\Delta\boldsymbol{a}$ and $\left[\boldsymbol{A}^T\boldsymbol{W}(\boldsymbol{y}-\boldsymbol{f})\right]$,

$$\Delta S = \Delta\boldsymbol{a}^T \cdot \left[\boldsymbol{A}^T\boldsymbol{W}(\boldsymbol{y}-\boldsymbol{f})\right]$$

measures (in the linear case) the distance in S to the minimum (called EDM = estimated distance to the minimum in MINUIT).

For $\Delta S < 1$ the step $\Delta\boldsymbol{a}$ is within *one* standard deviation from the minimum; the linear approximation is usually good for this small distance, and the convergence to the minimum is fast (doubling correct digits in one iteration).

Method: assume convergence to be reached after

$$\Delta S < 0.01 \quad \text{and actual function change} < 0.01$$

(valid, independent of number of parameters).

Least squares with constraints

Linear or non-linear (equality) constraints:

$$\boxed{f_k(\boldsymbol{a}_{\text{true}}, \boldsymbol{y}_{\text{true}}) = 0 \qquad\qquad k = 1, 2, \ldots m}$$

with

$$\begin{aligned} \boldsymbol{y} &= n\text{-vector of measured data} \\ \boldsymbol{V}[\boldsymbol{y}] &= \text{covariance matrix of the data} \\ \boldsymbol{W} &= \boldsymbol{V}[\boldsymbol{y}]^{-1} \quad \text{weight matrix} \\ \boldsymbol{a} &= p\text{-vector of parameters, unmeasured} \end{aligned}$$

Least squares principle: minimize with respect to $\Delta\boldsymbol{y}$ and $\boldsymbol{a}$

$$S(\boldsymbol{a}, \boldsymbol{\Delta y}) = \Delta\boldsymbol{y}^T \boldsymbol{W} \Delta\boldsymbol{y}$$

under the conditions

$$f_k(\boldsymbol{a}, \boldsymbol{y} + \Delta\boldsymbol{y}) = 0 \quad k = 1, 2, \ldots m$$

Example: π^0 decay

Assume decay

$$\pi^0 \quad \longrightarrow \quad \gamma_1 \, \gamma_2$$

with both photons measured in a calorimeter.

Simple method: use 4-vector $\quad P_{\pi^0} = P_1 + P_2 \quad$ for the π^0 to look for resonances of the π^0 with other particles.

Better method: use constraint

$$(P_1 + P_2)^2 = m^2_{\pi^0} = 0.135^2 \text{ GeV/c}^2$$

to improve and check the calorimeter measurement:

- 6 measured values $(E_1,\ \theta_1, \varphi_1,\ E_2,\ \theta_2, \varphi_2)$
- no unknown parameter
- 1 constraint (= one degree of freedom)

Kinematical constraints

Example: reconstruction of kinematic variables is often overconstrained – kinematical fit possible.

A different analysis style is standard at Desy:

- measurement errors are ignored,
- several formulas are used, based on subsets of data (no fit),
- comparison of alternatives (electron method, hadron method, double angle method, Σ method) by MC.

$$\text{e.g.} \qquad y_{\text{el}} = 1 - \frac{E'_e}{2E_e}\left(1 - \cos\theta'_e\right) \qquad\qquad y_{\text{JB}} = \frac{\delta_h}{2E_e}$$

$$y_{\text{DA}} = \frac{\sin\theta'_e\left(1 - \cos\gamma_h\right)}{\sin\theta'_e + \sin\gamma_h - \sin\left(\theta'_e + \gamma_h\right)} \qquad y_\Sigma = \frac{\delta_h}{\delta}$$

Should there be an optimal method, which uses all measured data?

Method of Lagrange multipliers

Additional m parameters λ_k, the Lagrange multipliers, are introduced, one for each equation and a new function is defined:

$$L(\boldsymbol{a}, \Delta\boldsymbol{y}) = S(\boldsymbol{a}, \Delta\boldsymbol{y}) + 2\sum_{k=1}^{m} \lambda_k f_k(\boldsymbol{a}, \boldsymbol{y} + \Delta\boldsymbol{y})$$

The necessary condition for a local extremum of this function with respect to all parameters $\Delta\boldsymbol{y}$, $\boldsymbol{a}$ and λ is equivalent to the condition of a minimum of $S(\boldsymbol{a}, \Delta\boldsymbol{y})$ under conditions $f_k(\boldsymbol{a}, \boldsymbol{y} + \Delta\boldsymbol{y}) = 0$.

Number of unknowns

n	measured parameters and corrections Δy_i
p	parameters a_j
m	constraints, multipliers λ_k
$n + p + m$	parameters in total

Linearization

Non-linear conditions $f_k(y + \Delta y) = 0$ have to be linearized. The linear equation has to be written in terms of the correction Δy

$$\begin{aligned} f_k\left(y + \Delta y^* + (\Delta y - \Delta y^*)\right) &= 0 \\ f_k\big|_{y+\Delta y^*} + (\Delta y - \Delta y^*) \left.\frac{\partial f}{\partial y}\right|_{y+\Delta y^*} &= 0 \\ \left(\frac{\partial f^*}{\partial y}\right) \Delta y &= \left(\frac{\partial f^*}{\partial y}\right) \Delta y^* - f_k^* \end{aligned}$$

Previous correction is Δy^*. Function f_k^* and derivative $\partial^*/\partial y$ is evaluated at $y + \Delta y^*$, where Δy^* is previous correction.

Linearization of f_k w.r.t. the parameters $\boldsymbol{a}$ in the same way, i.e. introducing $\boldsymbol{\Delta a}$ and $\boldsymbol{\Delta a}^*$.

→ same treatment of measured ($\boldsymbol{y}$) and unmeasured ($\boldsymbol{a}$) parameters.

Linearization – vector notation

$$\boldsymbol{f} + \boldsymbol{A}(\Delta\boldsymbol{a} - \Delta\boldsymbol{a}^*) + \boldsymbol{B}(\Delta\boldsymbol{y} - \Delta\boldsymbol{y}^*) = 0$$

or

$$\boxed{\boldsymbol{A}\Delta\boldsymbol{a} + \boldsymbol{B}\Delta\boldsymbol{y} = \boldsymbol{c}} \qquad \text{with} \qquad \boldsymbol{c} = \boldsymbol{A}\Delta\boldsymbol{a}^* + \boldsymbol{B}\Delta\boldsymbol{y}^* - \boldsymbol{f}$$

$$\boldsymbol{A} = \begin{pmatrix} \partial f_1/\partial a_1 & \partial f_1/\partial a_2 & \ldots & \partial f_1/\partial a_p \\ \partial f_2/\partial a_1 & \partial f_2/\partial a_2 & \ldots & \partial f_2/\partial a_p \\ \\ \ldots & & & \\ \partial f_m/\partial a_1 & \partial f_m/\partial a_2 & \ldots & \partial f_m/\partial a_p \end{pmatrix} \qquad \boldsymbol{f} = \begin{pmatrix} f_1(\boldsymbol{a}^*, \boldsymbol{y}^*) \\ f_2(\boldsymbol{a}^*, \boldsymbol{y}^*) \\ \\ \ldots \\ f_m(\boldsymbol{a}^*, \boldsymbol{y}^*) \end{pmatrix}$$

$$\boldsymbol{B} = \begin{pmatrix} \partial f_1/\partial y_1 & \partial f_1/\partial y_2 & \ldots & \partial f_1/\partial y_n \\ \partial f_2/\partial y_1 & \partial f_2/\partial y_2 & \ldots & \partial f_2/\partial y_n \\ \ldots & & & \\ \partial f_m/\partial y_1 & \partial f_m/\partial y_2 & \ldots & \partial f_m/\partial y_n \end{pmatrix}$$

. . . contnd.

Function L with linearization of constraints:

$$L = \Delta\boldsymbol{y}^T\boldsymbol{W}\Delta\boldsymbol{y} + 2\lambda^T(\boldsymbol{A}\Delta\boldsymbol{a} + \boldsymbol{B}\Delta\boldsymbol{y} - \boldsymbol{c})$$

The necessary conditions for an extremum are obtained by differentiation:

$$\boxed{\begin{array}{rcccrcl} \boldsymbol{W}\Delta\boldsymbol{y} & & & + & \boldsymbol{B}^T\lambda & = & 0 \\ & & & & \boldsymbol{A}^T\lambda & = & 0 \\ \boldsymbol{B}\Delta\boldsymbol{y} & + & \boldsymbol{A}\Delta\boldsymbol{a} & & & = & \boldsymbol{c} \end{array}}$$

This system of coupled matrix equations (in total $n + p + m$ equations) has to be solved for the unknowns $\Delta\boldsymbol{y}$, $\Delta\boldsymbol{a}$ and λ.

Case of no unknowns

$$\boxed{\begin{array}{lcl} \boldsymbol{W}\Delta\boldsymbol{y} + \boldsymbol{B}^T\lambda & = & 0 \\ \boldsymbol{B}\Delta\boldsymbol{y} & = & \boldsymbol{c} \end{array}}$$

Solution: multiply first equation with $\boldsymbol{W}^{-1}$

$$\begin{array}{rl} \ldots\text{from the left} & \Delta\boldsymbol{y} = -\boldsymbol{W}^{-1}\boldsymbol{B}^T\lambda \qquad (*) \\ \text{insert into second equation} & \left(\boldsymbol{B}\boldsymbol{W}^{-1}\boldsymbol{B}^T\right)\lambda = -\boldsymbol{c} \\ \text{solve for } \boldsymbol{\lambda} \text{ by} & \lambda = -\boldsymbol{W_B}\,\boldsymbol{c} \quad \text{with} \quad \boldsymbol{W}_B = \left(\boldsymbol{B}\boldsymbol{W}^{-1}\boldsymbol{B}^T\right)^{-1} \\ \text{insert into equation } (*) & \Delta\boldsymbol{y} = \left(\boldsymbol{W}^{-1}\boldsymbol{B}^T\boldsymbol{W}_B\right)\,\boldsymbol{c} \end{array}$$

For *linear* problems with constant $\boldsymbol{B}$ this is the solution; for *non-linear* problems iterations are necessary.

$$\hat{\boldsymbol{y}} = \boldsymbol{y} + \Delta\boldsymbol{y} \qquad \boldsymbol{V}(\hat{\boldsymbol{y}}) = \boldsymbol{W}^{-1} - \boldsymbol{W}^{-1}\left(\boldsymbol{B}^T\boldsymbol{W}_B\boldsymbol{B}\right)\boldsymbol{W}^{-1}$$

with covariance matrix by error propagation.

Pulls = normalized corrections

On average all corrections Δy_i have the same magnitude w.r.t. their standard deviation. A check of systematic deviations is possible by looking at distributions of pulls (= normalized corrections).

Covariance matrix of the corrections $\Delta \boldsymbol{y}$:

$$\boldsymbol{V}(\Delta \boldsymbol{y}) = \boldsymbol{W}^{-1} \left(\boldsymbol{B}^T \boldsymbol{W}_B \boldsymbol{B} \right) \boldsymbol{W}^{-1} = \boldsymbol{V}(\boldsymbol{y}) - \boldsymbol{V}(\hat{\boldsymbol{y}})$$

Normalized *pull values* are obtained by

$$p_i = \frac{\Delta y_i}{\sqrt{V(\boldsymbol{y})_{ii} - V(\hat{\boldsymbol{y}})_{ii}}}$$

The pulls should follow the standardized Gaussian distribution $N(0,1)$, if the measured data are normally distributed and the conditions are linear.

The general case

System of equations to be solved in the general case:

$$\begin{pmatrix} \boldsymbol{W} & 0 & \boldsymbol{B}^T \\ 0 & 0 & \boldsymbol{A}^T \\ \boldsymbol{B} & \boldsymbol{A} & 0 \end{pmatrix} \begin{pmatrix} \Delta \boldsymbol{y} \\ \Delta \boldsymbol{a} \\ \lambda \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \boldsymbol{c} \end{pmatrix}$$

What is the difference between *measured* and *unmeasured* parameters?
Elements of weight matrix w.r.t. to unmeasured parameters is zero!

Inverse of partitioned matrix:

$$\begin{pmatrix} \boldsymbol{W} & 0 & \boldsymbol{B}^T \\ 0 & 0 & \boldsymbol{A}^T \\ \boldsymbol{B} & \boldsymbol{A} & 0 \end{pmatrix}^{-1} = \begin{pmatrix} \boldsymbol{C}_{11} & \boldsymbol{C}_{21}^T & \boldsymbol{C}_{31}^T \\ \boldsymbol{C}_{21} & \boldsymbol{C}_{22} & \boldsymbol{C}_{32}^T \\ \boldsymbol{C}_{31} & \boldsymbol{C}_{32} & \boldsymbol{C}_{33} \end{pmatrix}$$

Solution

Elements of the inverse matrix:

$$\begin{aligned}
\boldsymbol{C}_{11} &= \boldsymbol{W}^{-1} - \boldsymbol{W}^{-1}\boldsymbol{B}^T\boldsymbol{W}_B\boldsymbol{B}\boldsymbol{W}^{-1} \\
&\quad + \boldsymbol{W}^{-1}\boldsymbol{B}^T\boldsymbol{W}_B\boldsymbol{A}\boldsymbol{W}_A^{-1}\boldsymbol{A}^T\boldsymbol{W}_B\boldsymbol{B}\boldsymbol{W}^{-1} \\
\boldsymbol{C}_{21} &= -\boldsymbol{W}_A^{-1}\boldsymbol{A}^T\boldsymbol{W}_B\boldsymbol{B}\boldsymbol{W}^{-1} \\
\boldsymbol{C}_{22} &= \boldsymbol{W}_A^{-1} \\
\boldsymbol{C}_{31} &= \boldsymbol{W}_B\boldsymbol{B}\boldsymbol{W}^{-1} - \boldsymbol{W}_B\boldsymbol{A}\boldsymbol{W}_A^{-1}\boldsymbol{A}^T\boldsymbol{W}_B\boldsymbol{B}\boldsymbol{W}^{-1} \\
\boldsymbol{C}_{32} &= \boldsymbol{W}_B\boldsymbol{A}\boldsymbol{W}_A^{-1} \\
\boldsymbol{C}_{33} &= -\boldsymbol{W}_B + \boldsymbol{W}_B\boldsymbol{A}\boldsymbol{W}_A^{-1}\boldsymbol{A}^T\boldsymbol{W}_B
\end{aligned}$$

with abbreviations

$$\begin{aligned}
\boldsymbol{W}_B &= (\boldsymbol{B}\boldsymbol{W}^{-1}\boldsymbol{B}^T)^{-1} \\
\boldsymbol{W}_A^{-1} &= (\boldsymbol{A}^T\boldsymbol{W}_B\boldsymbol{A})^{-1}
\end{aligned}$$

Note: the weight matrix $\boldsymbol{W}$ appears only as inverse: $\boldsymbol{W}^{-1} = V[\boldsymbol{y}]$ = covariance matrix of the measured data.

Corrections and covariance matrix

Corrections $\Delta\boldsymbol{y}$, $\Delta\boldsymbol{a}$ and the Lagrangian multipliers λ are obtained by multiplication:

$$\begin{aligned}
\Delta\boldsymbol{y} &= \boldsymbol{C}_{31}^T\,\boldsymbol{c} = (\boldsymbol{W}^{-1}\boldsymbol{B}^T\boldsymbol{W}_B - \boldsymbol{W}^{-1}\boldsymbol{B}^T\boldsymbol{W}_B\boldsymbol{A}\boldsymbol{W}_A^{-1}\boldsymbol{A}\boldsymbol{W}_B)\,\boldsymbol{c}\\
\Delta\boldsymbol{a} &= \boldsymbol{C}_{32}^T\,\boldsymbol{c} = \boldsymbol{W}_A^{-1}\boldsymbol{A}^T\boldsymbol{W}_B\,\boldsymbol{c}\\
\lambda &= \boldsymbol{C}_{33}\,\boldsymbol{c} = (-\boldsymbol{W}_B + \boldsymbol{W}_B\boldsymbol{A}\boldsymbol{W}_A^{-1}\boldsymbol{A}^T\boldsymbol{W}_B)\,\boldsymbol{c}
\end{aligned}$$

Iteration: convergence reached if change ΔS small and e.g.

$$F = \sum_k |f_k(\boldsymbol{a}+\Delta\boldsymbol{a},\boldsymbol{y}+\Delta\boldsymbol{y})| < \epsilon \quad ,$$

(requires same *scale* for all conditions).

Covariance matrix of the combined vector $\hat{\boldsymbol{y}}$, $\hat{\boldsymbol{a}}$ is

$$\boldsymbol{V}\begin{pmatrix}\hat{\boldsymbol{y}}\\ \hat{\boldsymbol{a}}\end{pmatrix} = \begin{pmatrix}\boldsymbol{C}_{11} & \boldsymbol{C}_{21}^T\\ \boldsymbol{C}_{21} & \boldsymbol{C}_{22}\end{pmatrix}$$

Checks

- Check pulls: if they follow $N(0,1)$, the input covariance matrix is probably correct. Otherwise *all* pulls are distorted, even if only one element of the input covariance matrix is incorrect.
- Check value of squares of deviations:

$$E(\hat{S}) = (m - p)$$

Note: the model may be incorrect!

On average S is 1.0 per degree of freedom, independent of type of the distribution, and

$$\hat{S} \quad \text{distributed} \quad \chi^2_{n-p}$$

if and only if the measured data are Gaussian distributed and the conditions are linear.

How to do such a fit in practice?

Use APLCON (”apply constraints”):

- all matrix operations hidden from the user
- matrices of derivatives calculated numerically from the constraints
- specific user code reduced to the minimum
- example: only 11 statements for π^0 fit

A program description for **APLCON** and the code is available via `http:/www.desy.de/~blobel/`.

The program code for π^0 fit, using APLCON

```
*     6 parameters and 1 constraint equations
      CALL SIMNIT(6,1,IDEB,1.E-5)
*     add four vectors
   20 EN = X(1) + X(4)
      PX = X(1)*SIN(X(2))*COS(X(3)) + X(4)*SIN(X(5))*COS(X(6))
      PY = X(1)*SIN(X(2))*SIN(X(3)) + X(4)*SIN(X(5))*SIN(X(6))
      PZ = X(1)*COS(X(2))           + X(4)*COS(X(5))
*     mass squared by square of four vector
      FM2  = EN**2 - PX**2 - PY**2 - PZ**2
*     constraint = calculated mass**2 - pi zero mass**2
      F(1) = FM2 - 0.135**2
*     call fit program
      CALL APLCON(X,VX,F,IREP)
      IF(IREP.LT.0) GOTO 20
      PMF=SQRT(PX**2+PY**2+PZ**2)
      XMF=SQRT(FM2)
```

Weighted least squares

The model of Linear Least Squares: $\boldsymbol{y} = \boldsymbol{A}\,\boldsymbol{a}$

$$\begin{aligned}
\boldsymbol{y} &= \text{vector of measured data} \\
\boldsymbol{A} &= \text{matrix (fixed)} \\
\boldsymbol{a} &= \text{vector of parameters} \\
\boldsymbol{r} &= \boldsymbol{y} - \boldsymbol{A}\boldsymbol{a} = \text{vector of residuals} \\
\boldsymbol{V}[\boldsymbol{y}] &= \text{covariance matrix of the data} \\
\boldsymbol{W} &= \boldsymbol{V}[\boldsymbol{y}]^{-1} \quad \text{weight matrix}
\end{aligned}$$

Least Squares Principle: minimize the expression

$$S(\boldsymbol{a}) = \boldsymbol{r}^T \boldsymbol{W} \boldsymbol{r} = (\boldsymbol{y} - \boldsymbol{A}\boldsymbol{a})^T \; W \; (\boldsymbol{y} - \boldsymbol{A}\boldsymbol{a})$$

with respect to $\boldsymbol{a}$.

Least Squares solution

Derivatives of expression $S(\boldsymbol{a})$:

$$\begin{aligned} \frac{\partial S}{\partial \boldsymbol{a}} &= 2\left(-\boldsymbol{A}^T \boldsymbol{W} \boldsymbol{y} + \left(\boldsymbol{A}^T \boldsymbol{W} \boldsymbol{A}\right) \boldsymbol{a}\right) \\ \frac{\partial^2 S}{\partial \boldsymbol{a}^2} &= \left(\boldsymbol{A}^T \boldsymbol{W} \boldsymbol{A}\right) \qquad = \text{constant} \end{aligned}$$

Solution (from $\partial S / \partial \boldsymbol{a} = 0$) is linear transf. of the data vector $\boldsymbol{y}$:

$$\hat{\boldsymbol{a}} = \left(\boldsymbol{A}^T \boldsymbol{W} \boldsymbol{A}\right)^{-1} \boldsymbol{A}^T \boldsymbol{W}\, \boldsymbol{y} \qquad = \boldsymbol{B} \boldsymbol{y}$$

Covariance matrix of $\boldsymbol{a}$ by "error" propagation ($\boldsymbol{V}[\boldsymbol{y}] = \boldsymbol{W}^{-1}$):

$$\boldsymbol{V}[\hat{\boldsymbol{a}}] = \boldsymbol{B}\, \boldsymbol{V}[\boldsymbol{y}]\, \boldsymbol{B}^T = \left(\boldsymbol{A}^T \boldsymbol{W} \boldsymbol{A}\right)^{-1}$$

(identical to inverse of second derivate of S).

Properties of solution

General scheme

Starting from **Principles** properties of the solution are derived, which are valid under certain conditions.

Conditions:

- Data are unbiased: $E[\boldsymbol{y}] = \boldsymbol{A}\,\bar{\boldsymbol{a}}$ $\quad$ ($\bar{\boldsymbol{a}}$ = true parameter vector
- Covariance matrix $\boldsymbol{V}[\boldsymbol{y}]$ is known (correct) and finite

Properties:

- Estimated parameters are unbiased: $E[\boldsymbol{a}] = \hat{\boldsymbol{a}}$

$$E[\hat{\boldsymbol{a}}] = \left(\boldsymbol{A}^T\boldsymbol{W}\boldsymbol{A}\right)^{-1}\ \boldsymbol{A}^T\boldsymbol{W}\,E[\boldsymbol{y}] = \bar{\boldsymbol{a}}$$

- In the class of unbiased estimates $\boldsymbol{a}^*$, which are linear in the data, the Least Squares estimates $\hat{\boldsymbol{a}}$ have the smallest variance (Gauß-Markoff theorem)

Properties are not valid, if conditions violated.

Straight line fit

Example: track fit of y (measured) vs. abscissa x

$$y_i = a_0 + a_1 \cdot x_i$$

Matrix $\boldsymbol{A}$ and parameter vector $\boldsymbol{a}$

$$\boldsymbol{A} = \begin{pmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_n \end{pmatrix} \qquad \boldsymbol{a} = \begin{pmatrix} a_0 \\ a_1 \end{pmatrix}$$

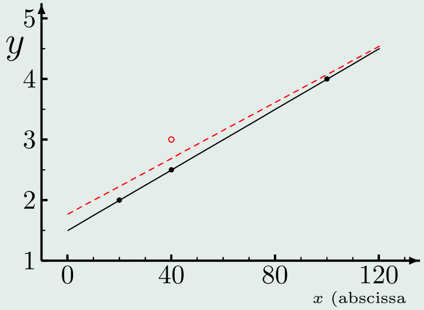
Weight matrix is diagonal (*independent* measurements):

$$\boldsymbol{A}^T \boldsymbol{W} \boldsymbol{A} = \begin{pmatrix} \sum w_i & \sum w_i x_i \\ \sum w_i x_i & \sum w_i x_i^2 \end{pmatrix} \qquad \boldsymbol{A}^T \boldsymbol{W} \boldsymbol{y} = \begin{pmatrix} \sum w_i y_i \\ \sum w_i x_i y_i \end{pmatrix}$$

If one measured y_i-value is shifted (biased), then

- parameters biased, and
- χ^2-value very high.

Straight line fit



The full line is a straight line fit to three well aligned data points (black dots).

The dashed curve is the straight line fit, if the middle point is "badly aligned" (circle).

Normal equations for uncorrelated data

Usually the direct measurements, which are input to a least squares problem, are *uncorrelated*, i.e. the covariance matrix $\boldsymbol{V}(\boldsymbol{y})$ and the weight matrix $\boldsymbol{W}$ are *diagonal.* This property, which is assumed here, simplifies the computation of the matrix products

$$\boldsymbol{C} = \boldsymbol{A}^T \boldsymbol{W} \boldsymbol{A} \qquad \text{and} \qquad \boldsymbol{b} = \boldsymbol{A}^T \boldsymbol{W} \boldsymbol{y}$$

which are necessary for the solution

$$\hat{\boldsymbol{a}} = \boldsymbol{C}^{-1} \boldsymbol{b}$$

Normal equations for uncorrelated data contnd.

The diagonal elements of the weight matrix $\boldsymbol{W}$ are denoted by w_i, with $w_i = 1/\sigma_i^2$. Each data value y_i with its weight w_i makes an independent contribution to the final matrix products. Calling the i-th row $\boldsymbol{A}_i$, with

$$i\text{-th row of } \boldsymbol{A} \qquad\qquad \boldsymbol{A}_i = (d_1,\, d_2,\, \ldots,\, d_p)$$

the contributions of this row to $\boldsymbol{C}$ and $\boldsymbol{b}$ can be written as the $p \times p$-matrix $w_i \boldsymbol{A}_i^T \cdot \boldsymbol{A}_i$ and the p-vector $w_i \boldsymbol{A}_i^T \cdot y_i$. The contributions of a single row are:

$$\begin{array}{c|cccc} & d_1 & d_2 & \ldots & d_p \\ \hline d_1 & w_i d_1^2 & w_i d_1 d_2 & \ldots & w_i d_1 d_p \\ d_2 & & w_i d_2^2 & \ldots & w_i d_2 d_p \\ \ldots & & & \ldots & \ldots \\ d_p & & & & w_i d_p^2 \end{array} \qquad\qquad \begin{array}{c|c} & y_i \\ \hline d_1 & w_i d_1 z_i \\ d_2 & w_i d_2 z_i \\ \ldots & \ldots \\ d_p & w_i d_p z_i \end{array} \;,$$

where the symmetric elements in the lower half are not shown.

Robust least square fit

Least square fit on observations y_i, yielding fitted values $\hat{y}_i$, and residuals $r_i = y_i - \hat{y}_i$. Estimate for the standard error of y_i (or of the r_i) is s_i.

- "Clean" the data by pulling outliers towards their fitted values: winsorize the observations y_i and replace them by pseudo-observations y_i^*:

$$\begin{aligned} y_i^* &= y_i \, , & \text{if} \quad & |r_i| \le c\, s_i \, , \\ &= \hat{y}_i - c\, s_i \, , & \text{if} \quad & r_i < -c\, s_i \, , \\ &= \hat{y}_i + c\, s_i \, , & \text{if} \quad & r_i > +c\, s_i \, . \end{aligned}$$

  The factor c regulates the amount of robustness, a goid choice is $c = 1.5$.

- Refit iteratively: the pseudo-observations y_i^* are used to calculate new parameters and new fitted values $\hat{y}_i$.

Robust least square fit

Classical estimate of s_i, if all observations are equally accurate:

$$s^2 = \frac{1}{n-p} \sum_{i=1}^{n} r_i^2$$

($(n-p)$ = number of observations minus the number of parameters). Estimate standard error of residual s_i by $s_i = \sqrt{1 - H_{ii}} s$, where H_{ii} is the diagonal element of $\boldsymbol{H} = \boldsymbol{X} \left(\boldsymbol{X}^T \boldsymbol{X} \right)^{-1} \boldsymbol{X}^T$.

Using modified (reduced) residuals $r_i^* = y_i^* - \hat{y}_i$ instead of r_i, the (bias corrected) estimate of s^2 is

$$s^2 = \left(\frac{n}{m} \right)^2 \frac{1}{n-p} \sum_{i=1}^{n} r_i^{*2}$$

where m is the number of unmodified observations ($y_i^* = y_i$).

Least squares methods