Statistical analysis of experimental data
Least-squares method (2)

Aleksander Filip Żarnecki

FACULTY OF PHYSICS
UNIVERSITY OF WARSAW

Lecture 09

December 7, 2023

Least-squares method (2)

1. Non-linear fit procedure
2. F-test
3. Constrained fit
4. Homework

Maximum Likelihood Method

see lectures 05 and 06

Let us consider N independent measurements of variable Y. Assuming measurement fluctuations are described by Gaussian pdf, the likelihood function is:

$$L = \prod_{i=1}^{N} G(y_i; \mu_i, \sigma_i) = \prod_{i=1}^{N} \frac{1}{\sigma_i \sqrt{2\pi}} \exp\left(-\frac{1}{2} \frac{(y_i - \mu_i)^2}{\sigma_i^2} \right)$$

Log-likelihood:

assuming σ_i are known

$$\ell = -\frac{1}{2} \sum \frac{(y_i - \mu_i)^2}{\sigma_i^2} + \text{const}$$

We can define

$$\chi^2 = -2\,\ell = -2 \ln L = \sum_{i=1}^{N} \frac{(y_i - \mu_i)^2}{\sigma_i^2}$$

Maximum of (log-)likelihood function corresponds to minimum of χ^2

χ^2 distribution

We conclude that distribution of χ^2 is described by Gamma pdf with:

$$k = \frac{N}{2} \quad \text{and} \quad \lambda = \frac{1}{2}$$

Properties of the χ^2 distribution (Gamma pdf: see lecture 03)

$$\langle \chi^2 \rangle = \frac{k}{\lambda} = N$$

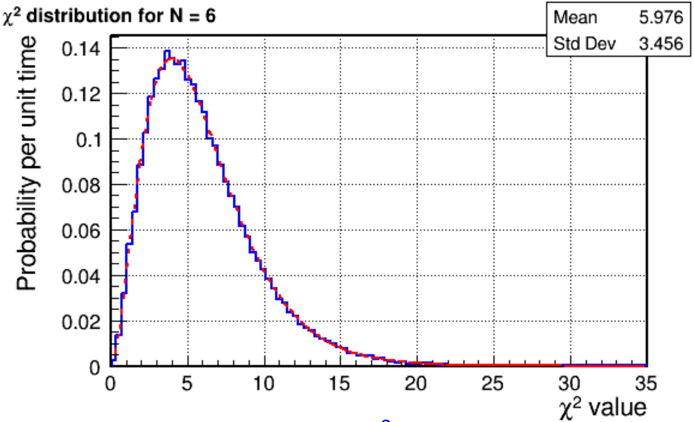
$$\mathbb{V}(\chi^2) = \frac{k}{\lambda^2} = 2N$$

$$\sqrt{\mathbb{V}(\chi^2)} = \sigma_{\chi^2} = \sqrt{2N}$$

For small N, value of χ^2 is a subject to large fluctuations...

χ^2 distribution

Results of the Monte Carlo sample generation (compared with predictions)



It is interesting to note that maximum position, $\chi^2_{max} = N - 2$ (!!!)

Reduced χ^2

When discussing consistency of large data samples it is often convenient to use reduced χ^2 value:

$$\chi^2_{red} \; = \; \frac{\chi^2}{N}$$

Distribution of χ^2_{red} is again described by Gamma pdf with

$$k = \frac{N}{2} \quad \text{and} \quad \lambda = \frac{N}{2}$$

Properties of the distribution:

$$\langle \chi^2_{red} \rangle \; = \; 1 \qquad\qquad \mathbb{V}(\chi^2_{red}) \; = \; \sigma^2_{\chi^2_{red}} \; = \; \frac{2}{N}$$

General case

We introduced χ^2 in a very general form:

$$\chi^2 = \sum_{i=1}^{N} \frac{(y_i - \mu_i)^2}{\sigma_i^2}$$

where different μ_i and σ_i are possible for each of N measurement y_i

It is quite often the case that values of μ_i depend on some controlled variables x_i and a smaller set of model parameters:

$$\mu_i = \mu(x_i; \mathbf{a})$$

we can then use the least-squares method to extract the best estimates of parameters **a** from the collected set of data points (x_i, y_i)

We can look for minimum of χ^2 using different numerical algorithms...

Linear case

The case which is particularly interesting is when the dependence is linear in parameters (!):

$$\mu(x; \mathbf{a}) \;=\; \sum_{k=1}^{M} a_k \, f_k(x)$$

where $f_k(x)$ is a set of functions with arbitrary analytical form.

One of the examples is the polynomial series:

$$f_k(x) \;=\; x^k \qquad \Rightarrow \quad \mu(x; \mathbf{a}) \;=\; \sum_{k=1}^{M} a_k \, x^{k-1}$$

but any set of functions can be used, if they are not linearly dependent.
Functions ortogonal on given set of points x_i should work best...

Parameter fit

Bonamente

We obtain a set of M equations for M parameters a_i:

$$\sum_{i=1}^{N} \frac{f_l(x_i)}{\sigma_i^2} \left(y_i - \sum_{k=1}^{M} a_k \; f_k(x) \right) \;=\; 0 \qquad l = 1 \ldots M$$

which can be rewritten as:

$$\sum_{k=1}^{M} \left(\sum_{i=1}^{N} \frac{f_l(x_i) \; f_k(x_i)}{\sigma_i^2} \right) a_k \;=\; \sum_{i=1}^{N} \frac{f_l(x_i) \; y_i}{\sigma_i^2}$$

or in the matrix form:

$$\mathbb{A} \cdot \mathbf{a} \;=\; \mathbf{b}$$

where $\mathbb{A}_{lk} = \sum_{i=1}^{N} \frac{f_l(x_i) \; f_k(x_i)}{\sigma_i^2}$ and $b_l = \sum_{i=1}^{N} \frac{f_l(x_i) \; y_i}{\sigma_i^2}$

Parameter fit

Solution of this set of equations can be obtained by inverting matrix $\mathbb{A}$

$$\mathbf{a} \;=\; \mathbb{A}^{-1} \cdot \mathbf{b}$$

This also gives us the estimate of parameter covariance matrix:

$$\mathbb{C}_{\mathbf{a}} \;=\; \left(- \frac{\partial^2 \ell}{\partial a_l \, \partial a_k} \right)^{-1} = \left(\frac{1}{2} \frac{\partial^2 \chi^2}{\partial a_l \, \partial a_k} \right)^{-1} = \mathbb{A}^{-1}$$

One can write

$$\mathbb{C}_{\mathbf{a}} \;=\; \left(\sum_{i=1}^{N} \frac{f_l(x_i)\; f_k(x_i)}{\sigma_i^2} \right)^{-1}$$

Expected uncertainties of the extracted parameter values depend on the choice of measurement points x_i but, surprisingly, do not depend on the actual results y_i

$\Rightarrow$ very useful when planning the experiment...

Polynomial fit example

For clarity of notation, it is convenient to change parameter numbering to $k, l = 0 \ldots M$ (for polynomial fit of order M, $M + 1$ parameters).

$$\mathbb{A}_{l\,k} = \sum_{i=1}^{N} \frac{x_i^{(l+k)}}{\sigma_i^2} \quad \text{and} \quad b_l = \sum_{i=1}^{N} \frac{x_i^l \; y_i}{\sigma_i^2}$$

For uniform uncertainties it is then:

$$\mathbb{A} = \frac{1}{\sigma^2} \sum_{i=1}^{N} \begin{pmatrix} 1 & x_i & \ldots & x_i^M \\ x_i & x_i^2 & \ldots & x_i^{M+1} \\ \vdots & & & \vdots \\ x_i^M & x_i^{M+1} & \ldots & x_i^{2M} \end{pmatrix} \qquad \mathbf{b} = \frac{1}{\sigma^2} \sum_{i=1}^{N} \begin{pmatrix} y_i \\ y_i \; x_i \\ \vdots \\ y_i \; x_i^M \end{pmatrix}$$

quite simple to implement...

Least-squares method (2)

1. Non-linear fit procedure
2. F-test
3. Constrained fit
4. Homework

Non-linear fit procedure

General case

We consider χ^2 in a general form:

$$\chi^2(\mathbf{a}) = \sum_{i=1}^{N} \frac{(y_i - \mu_i)^2}{\sigma_i^2}$$

where in general different σ_i are allowed for each of N measurements y_i.

General case

We consider χ^2 in a general form:

$$\chi^2(\mathbf{a}) = \sum_{i=1}^{N} \frac{(y_i - \mu_i)^2}{\sigma_i^2}$$

where in general different σ_i are allowed for each of N measurements y_i.

Values of μ_i depend on some controlled variables x_i and a set of model parameters $\mathbf{a}$:

$$\mu_i = \mu(x_i; \mathbf{a})$$

We look for the best estimate of **a**, which should correspond to the global minimum of $\chi^2(\mathbf{a})$ for given set of data points (x_i, y_i).

General case

We consider χ^2 in a general form:

$$\chi^2(\mathbf{a}) = \sum_{i=1}^{N} \frac{(y_i - \mu_i)^2}{\sigma_i^2}$$

where in general different σ_i are allowed for each of N measurements y_i.

Values of μ_i depend on some controlled variables x_i and a set of model parameters $\mathbf{a}$:

$$\mu_i = \mu(x_i; \mathbf{a})$$

We look for the best estimate of **a**, which should correspond to the global minimum of $\chi^2(\mathbf{a})$ for given set of data points (x_i, y_i).

For linear problem, this minimum can be found directly... (lecture 08)

General case

We consider χ^2 in a general form:

$$\chi^2(\mathbf{a}) = \sum_{i=1}^{N} \frac{(y_i - \mu_i)^2}{\sigma_i^2}$$

where in general different σ_i are allowed for each of N measurements y_i.

Values of μ_i depend on some controlled variables x_i and a set of model parameters **a**:

$$\mu_i = \mu(x_i; \mathbf{a})$$

We look for the best estimate of **a**, which should correspond to the global minimum of $\chi^2(\mathbf{a})$ for given set of data points (x_i, y_i).

For linear problem, this minimum can be found directly... (lecture 08)

For non-linear problems, we need to use iterative procedures...

Iterative procedure

(Brandt)

We start from some "initial guess" of parameter values $\mathbf{a_0}$.

Assuming small variations of the model parameters, $\mathbf{a} = \mathbf{a_0} + \delta\mathbf{a}$, we expand χ^2 in a series:

$$\chi^2(\mathbf{a}) \;=\; \chi^2(\mathbf{a_0}) - 2\,\mathbf{b}\cdot(\mathbf{a} - \mathbf{a_0}) + \ldots$$

where vector $\mathbf{b}$ is the negative gradient of χ^2:

$$\mathbf{b} \;=\; -\frac{1}{2}\,\nabla\,\chi^2(\mathbf{a_0}) \qquad\qquad b_j \;=\; -\frac{1}{2}\frac{\partial\chi^2}{\partial a_j} \;=\; \sum_{i=1}^{N}\frac{y_i - \mu_i}{\sigma_i^2}\cdot\frac{\partial\mu_i}{\partial a_j}$$

Iterative procedure (Brandt)

We start from some "initial guess" of parameter values $\mathbf{a_0}$.

Assuming small variations of the model parameters, $\mathbf{a} = \mathbf{a_0} + \delta\mathbf{a}$, we expand χ^2 in a series:

$$\chi^2(\mathbf{a}) \;=\; \chi^2(\mathbf{a_0}) - 2\,\mathbf{b}\cdot(\mathbf{a}-\mathbf{a_0}) + \ldots$$

where vector $\mathbf{b}$ is the negative gradient of χ^2:

$$\mathbf{b} \;=\; -\frac{1}{2}\,\nabla\,\chi^2(\mathbf{a_0}) \qquad\qquad b_j \;=\; -\frac{1}{2}\frac{\partial\chi^2}{\partial a_j} \;=\; \sum_{i=1}^{N}\frac{y_i-\mu_i}{\sigma_i^2}\cdot\frac{\partial\mu_i}{\partial a_j}$$

Vector $\mathbf{b}$ defines the direction of steepest χ^2 descent.
One of the possible procedures is to make a step in this direction:

$$\mathbf{a_1} \;=\; \mathbf{a_0} \,+\, \varepsilon\,\mathbf{b}$$

with small $\varepsilon > 0$ and then repeat the whole procedure...

Iterative procedure (Brandt)

We can try to be "smarter". Expanding χ^2 to quadratic term:

$$\chi^2(\mathbf{a}) \;=\; \chi^2(\mathbf{a_0}) - 2\,\mathbf{b}\cdot(\mathbf{a}-\mathbf{a_0}) + (\mathbf{a}-\mathbf{a_0})^\mathsf{T}\mathbb{A}(\mathbf{a}-\mathbf{a_0}) + \ldots$$

where $\mathbb{A}$ is the so called Hessian matrix of second derivatives:

$$\mathbb{A}_{jk} \;=\; \frac{1}{2}\,\frac{\partial^2\chi^2}{\partial a_j\,\partial a_k}\bigg|_{\mathbf{a}=\mathbf{a_0}}$$

Iterative procedure

(Brandt)

We can try to be "smarter". Expanding χ^2 to quadratic term:

$$\chi^2(\mathbf{a}) \;=\; \chi^2(\mathbf{a_0}) - 2\,\mathbf{b}\cdot(\mathbf{a}-\mathbf{a_0}) + (\mathbf{a}-\mathbf{a_0})^{\mathsf{T}}\mathbb{A}(\mathbf{a}-\mathbf{a_0}) + \ldots$$

where $\mathbb{A}$ is the so called Hessian matrix of second derivatives:

$$\mathbb{A}_{jk} \;=\; \frac{1}{2}\,\frac{\partial^2\chi^2}{\partial a_j\,\partial a_k}\bigg|_{\mathbf{a}=\mathbf{a_0}} \approx \sum_{i=1}^{N}\frac{1}{\sigma_i^2}\cdot\frac{\partial\mu_i}{\partial a_j}\cdot\frac{\partial\mu_i}{\partial a_k} \qquad \left(\text{neglecting } \frac{\partial^2\mu_i}{\partial a_j\,\partial a_k}\right)$$

Iterative procedure

(Brandt)

We can try to be "smarter". Expanding χ^2 to quadratic term:

$$\chi^2(\mathbf{a}) \;=\; \chi^2(\mathbf{a_0}) - 2\,\mathbf{b}\cdot(\mathbf{a}-\mathbf{a_0}) + (\mathbf{a}-\mathbf{a_0})^\mathsf{T}\mathbb{A}(\mathbf{a}-\mathbf{a_0}) + \ldots$$

where $\mathbb{A}$ is the so called Hessian matrix of second derivatives:

$$\mathbb{A}_{jk} \;=\; \frac{1}{2}\,\frac{\partial^2\chi^2}{\partial a_j\,\partial a_k}\bigg|_{\mathbf{a}=\mathbf{a_0}} \;\approx\; \sum_{i=1}^{N}\frac{1}{\sigma_i^2}\cdot\frac{\partial\mu_i}{\partial a_j}\cdot\frac{\partial\mu_i}{\partial a_k} \qquad \left(\text{neglecting } \frac{\partial^2\mu_i}{\partial a_j\,\partial a_k}\right)$$

In this approximation, we can calculate the expected position of the χ^2 minimum:

$$\nabla\,\chi^2(\mathbf{a}) \;=\; -2\,\mathbf{b} \;+\; 2\,\mathbb{A}\,(\mathbf{a}-\mathbf{a_0})$$

Iterative procedure

(Brandt)

We can try to be "smarter". Expanding χ^2 to quadratic term:

$$\chi^2(\mathbf{a}) \;=\; \chi^2(\mathbf{a_0}) - 2\,\mathbf{b}\cdot(\mathbf{a}-\mathbf{a_0}) + (\mathbf{a}-\mathbf{a_0})^{\mathsf{T}}\mathbb{A}(\mathbf{a}-\mathbf{a_0}) + \ldots$$

where $\mathbb{A}$ is the so called Hessian matrix of second derivatives:

$$\mathbb{A}_{jk} \;=\; \frac{1}{2}\,\frac{\partial^2\chi^2}{\partial a_j\,\partial a_k}\bigg|_{\mathbf{a}=\mathbf{a_0}} \;\approx\; \sum_{i=1}^{N}\frac{1}{\sigma_i^2}\cdot\frac{\partial\mu_i}{\partial a_j}\cdot\frac{\partial\mu_i}{\partial a_k} \qquad \left(\text{neglecting } \frac{\partial^2\mu_i}{\partial a_j\,\partial a_k}\right)$$

In this approximation, we can calculate the expected position of the χ^2 minimum:

$$\nabla\,\chi^2(\mathbf{a}) \;=\; -2\,\mathbf{b} \,+\, 2\,\mathbb{A}\,(\mathbf{a}-\mathbf{a_0}) \;=\; 0$$

$$\Rightarrow\;\; \mathbf{a_m} \;=\; \mathbf{a_0} \,+\, \mathbb{A}^{-1}\mathbf{b}$$

and we can try to "jump" directly to the minimum...

Iterative procedure (Brandt)

Efficiency of the iterative procedure depends strongly on the model and the data used, but also on the initial choice of model parameters $\mathbf{a_0}$.

Iterative procedure (Brandt)

Efficiency of the iterative procedure depends strongly on the model and the data used, but also on the initial choice of model parameters $\mathbf{a_0}$.

If our "initial guess" is close to the true (global) minimum, procedure based on the Hessian matrix inversion is very efficient and converges fast.
However, it can be unstable, if we start too far from the actual minimum...

Iterative procedure (Brandt)

Efficiency of the iterative procedure depends strongly on the model and the data used, but also on the initial choice of model parameters $\mathbf{a_0}$.

If our "initial guess" is close to the true (global) minimum, procedure based on the Hessian matrix inversion is very efficient and converges fast.
However, it can be unstable, if we start too far from the actual minimum...

Iterative procedure based on the gradient calculation is much slower, but it much more robust.
It finds the minimum even when starting from a parameter space point far away from it.

Iterative procedure (Brandt)

Efficiency of the iterative procedure depends strongly on the model and the data used, but also on the initial choice of model parameters $\mathbf{a_0}$.

If our "initial guess" is close to the true (global) minimum, procedure based on the Hessian matrix inversion is very efficient and converges fast.
However, it can be unstable, if we start too far from the actual minimum...

Iterative procedure based on the gradient calculation is much slower, but it much more robust.
It finds the minimum even when starting from a parameter space point far away from it.

There are many different numerical algorithms for solving this problem, and there is no universal "best choice"...

One of the main problems in minimization is that finding the minimum does not guarantee that it is a global minimum...

Marquardt Minimization (Brandt)

One of the popular approaches, combining the two previously discussed:

$$\mathbf{a_{i+1}} = \mathbf{a_i} + \lambda^{-1}\mathbf{b} \quad (\lambda \equiv 1/\varepsilon) \qquad \mathbf{a_{i+1}} = \mathbf{a_i} + \mathbb{A}^{-1}\mathbf{b}$$

Marquardt Minimization (Brandt)

One of the popular approaches, combining the two previously discussed:

$$\mathbf{a_{i+1}} \;=\; \mathbf{a_i} + (\mathbb{A} + \lambda \cdot \mathbb{I})^{-1}\,\mathbf{b}$$

where the value of λ determines the performance of the algorithm:

- for $\lambda \gg 1$ we make a small step along the gradient direction
  which corresponds to the gradient minimization with $\varepsilon \approx \frac{1}{\lambda}$
- for $\lambda \ll 1$ we try to "jump" directly to the minimum position
  Hessian matrix solution is reproduced for $\lambda \to 0$

Marquardt Minimization

(Brandt)

One of the popular approaches, combining the two previously discussed:

$$\mathbf{a_{i+1}} = \mathbf{a_i} + (\mathbb{A} + \lambda \cdot \mathbb{I})^{-1}\,\mathbf{b}$$

where the value of λ determines the performance of the algorithm:

- for $\lambda \gg 1$ we make a small step along the gradient direction
  which corresponds to the gradient minimization with $\varepsilon \approx \frac{1}{\lambda}$
- for $\lambda \ll 1$ we try to "jump" directly to the minimum position
  Hessian matrix solution is reproduced for $\lambda \to 0$

The key element proposed by D.W.Marquardt (1963) was to use variable λ parameter, adjusting its value to the results of the previous step...

Marquardt Minimization (Brandt)

The following algorithm can be used:

1. Take initial parameter values $\mathbf{a_0}$ resulting in χ^2 value of χ^2_0.
   Assume $\lambda_0 = 0.01$.

Marquardt Minimization (Brandt)

The following algorithm can be used:

1. Take initial parameter values $\mathbf{a_0}$ resulting in χ^2 value of χ^2_0.
   Assume $\lambda_0 = 0.01$.
2. Calculate matrix $\mathbb{A}$ and vector $\mathbf{b}$ for current $\mathbf{a_i}$.

Marquardt Minimization (Brandt)

The following algorithm can be used:

1. Take initial parameter values $\mathbf{a_0}$ resulting in χ^2 value of χ^2_0.
   Assume $\lambda_0 = 0.01$.
2. Calculate matrix $\mathbb{A}$ and vector $\mathbf{b}$ for current $\mathbf{a_i}$.
3. Calculate new parameter vector:
   $$\mathbf{a_{i+1}} = \mathbf{a_i} + \left(\mathbb{A} + \lambda_i \cdot \mathbb{I}\right)^{-1} \mathbf{b}$$
   and calculate the corresponding value of χ^2_{i+1}

Marquardt Minimization (Brandt)

The following algorithm can be used:

1. Take initial parameter values $\mathbf{a_0}$ resulting in χ^2 value of χ^2_0.
   Assume $\lambda_0 = 0.01$.
2. Calculate matrix $\mathbb{A}$ and vector $\mathbf{b}$ for current $\mathbf{a_i}$.
3. Calculate new parameter vector:
   $$\mathbf{a_{i+1}} = \mathbf{a_i} + \left(\mathbb{A} + \lambda_i \cdot \mathbb{I}\right)^{-1} \mathbf{b}$$
   and calculate the corresponding value of χ^2_{i+1}
4. Compare with the previous iteration:
   - If $\chi^2_{i+1} < \chi^2_i$:
     $\Rightarrow$ accept the new parameter set and decrease λ by factor 10
   - If $\chi^2_{i+1} \geq \chi^2_i$:
     $\Rightarrow$ keep old parameter values $(\mathbf{a_{i+1}} = \mathbf{a_i})$ and increase λ by factor 10

Marquardt Minimization (Brandt)

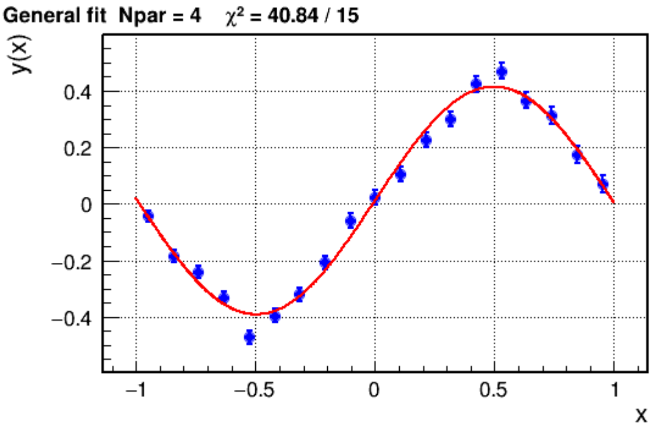
The following algorithm can be used:

1. Take initial parameter values $\mathbf{a_0}$ resulting in χ^2 value of χ^2_0.
   Assume $\lambda_0 = 0.01$.
2. Calculate matrix $\mathbb{A}$ and vector $\mathbf{b}$ for current $\mathbf{a_i}$.
3. Calculate new parameter vector:
   $$\mathbf{a_{i+1}} = \mathbf{a_i} + (\mathbb{A} + \lambda_i \cdot \mathbb{I})^{-1}\,\mathbf{b}$$
   and calculate the corresponding value of χ^2_{i+1}
4. Compare with the previous iteration:
   - If $\chi^2_{i+1} < \chi^2_i$:
     $\Rightarrow$ accept the new parameter set and decrease λ by factor 10
   - If $\chi^2_{i+1} \geq \chi^2_i$:
     $\Rightarrow$ keep old parameter values $(\mathbf{a_{i+1}} = \mathbf{a_i})$ and increase λ by factor 10
5. Iterate points 2–4 until required precision reached

Marquardt Minimization example

09_fit1.ipynb Open in Colab

Fitting Fourier series to example data set

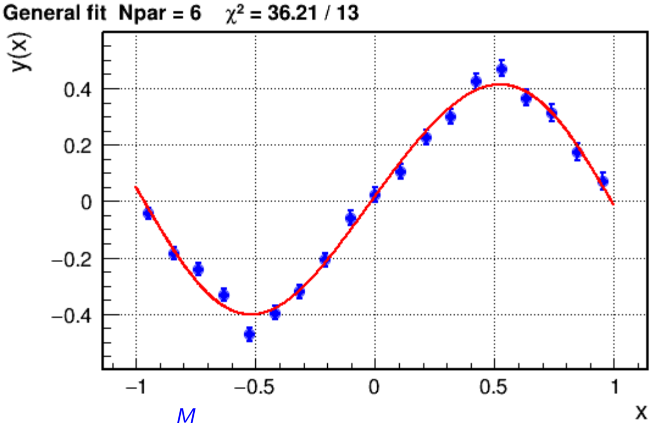


$$y(x) \quad = \quad a_1 + a_2 \sin(a_0\ x) + a_3 \cos(a_0\ x)$$

Marquardt Minimization example

09_fit1.ipynb Open in Colab

Fitting Fourier series to example data set

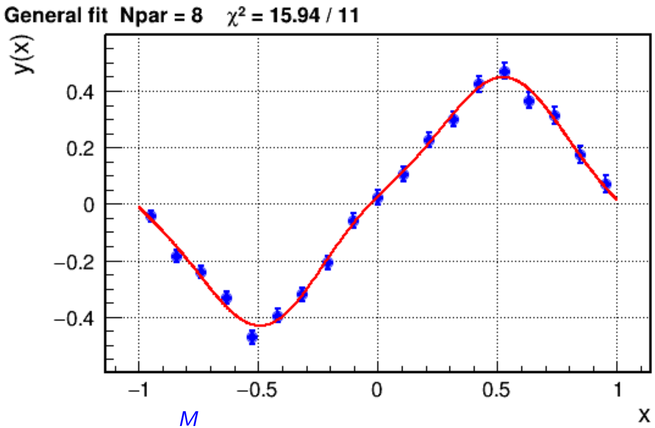


$$y(x) \quad = \quad a_1 + \sum_{n=1}^{M} a_{2n} \sin(n a_0 x) + a_{2n+1} \cos(n a_0 x) \qquad M = 2$$

Marquardt Minimization example

09_fit1.ipynb Open in Colab

Fitting Fourier series to example data set

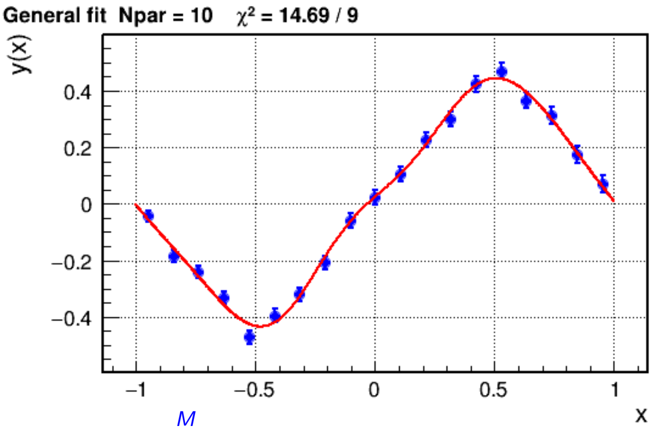


$$y(x) = a_1 + \sum_{n=1}^{M} a_{2n} \sin(n a_0 x) + a_{2n+1} \cos(n a_0 x) \qquad M = 3$$

Marquardt Minimization example

09_fit1.ipynb Open in Colab

Fitting Fourier series to example data set

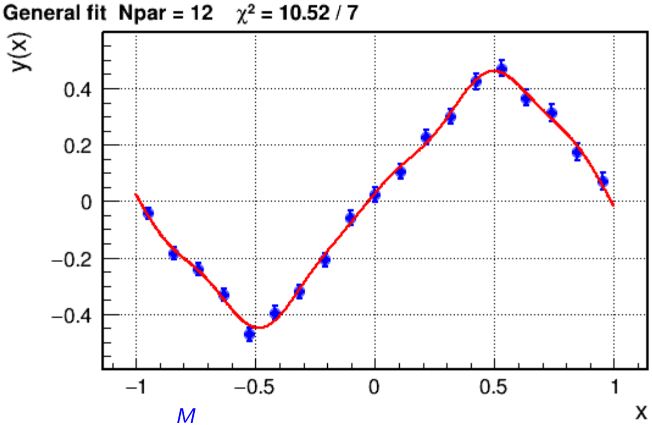


$$y(x) = a_1 + \sum_{n=1}^{M} a_{2n} \sin(n a_0 x) + a_{2n+1} \cos(n a_0 x) \qquad M = 4$$

Marquardt Minimization example

09_fit1.ipynb   Open in Colab

Fitting Fourier series to example data set

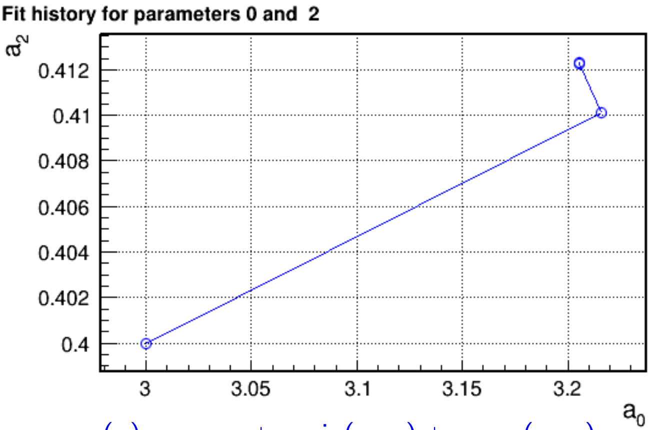


$$y(x) \quad = \quad a_1 + \sum_{n=1}^{M} a_{2n} \sin(n a_0 x) + a_{2n+1} \cos(n a_0 x) \qquad M = 5$$

Marquardt Minimization example

09_fit2.ipynb Open in Colab

Fit converges fast when the initial values are "correct" $(a_0 = 3)$

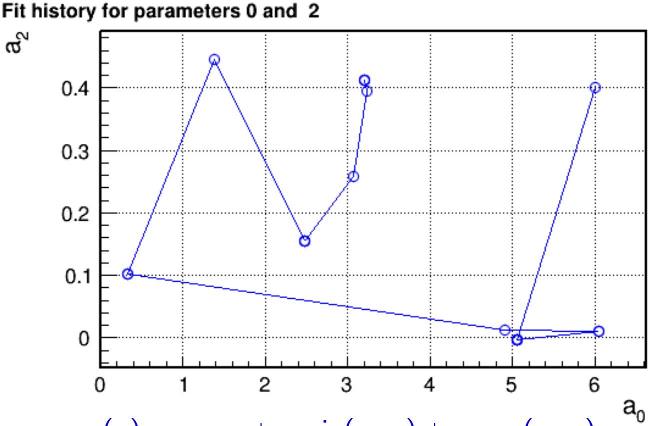


$$y(x) = a_1 + a_2 \sin(a_0\ x) + a_3 \cos(a_0\ x)$$

Marquardt Minimization example

09_fit2.ipynb Open in Colab

Much slower convergence when the initial values are “wrong” $(a_0 = 6)$

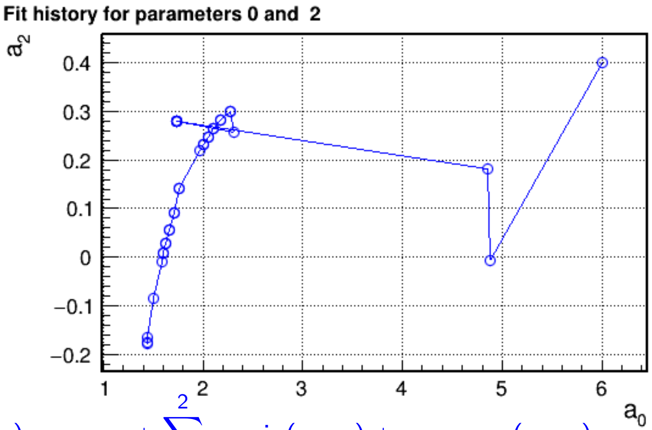


$$y(x) \quad = \quad a_1 + a_2 \sin(a_0\ x) + a_3 \cos(a_0\ x)$$

Marquardt Minimization example

09_fit2.ipynb Open in Colab

"Wrong" initial values can result in incorrect result $(a_0 = 6)$

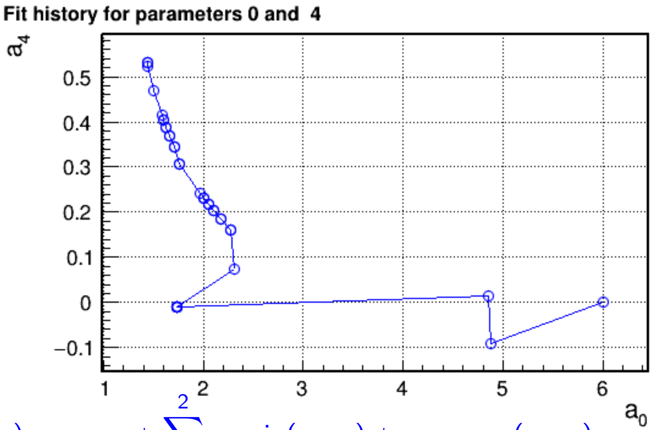


$$y(x) = a_1 + \sum_{n=1}^{2} a_{2n} \sin(n a_0 x) + a_{2n+1} \cos(n a_0 x)$$

Marquardt Minimization example

09_fit2.ipynb Open in Colab

"Wrong" initial values can result in incorrect result $(a_0 = 6)$

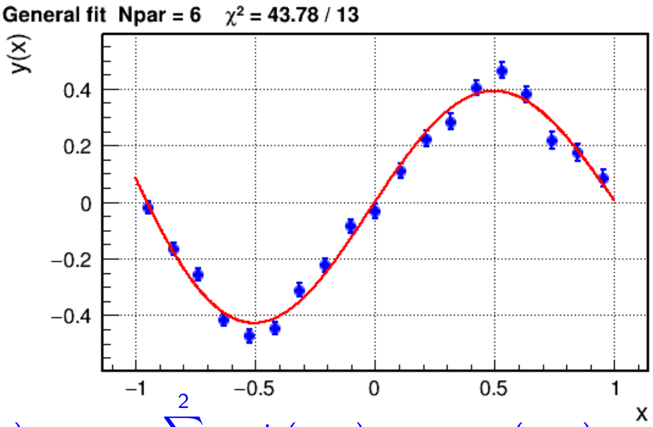


$$y(x) = a_1 + \sum_{n=1}^{2} a_{2n} \sin(n a_0 x) + a_{2n+1} \cos(n a_0 x)$$

Marquardt Minimization example

09_fit2.ipynb Open in Colab

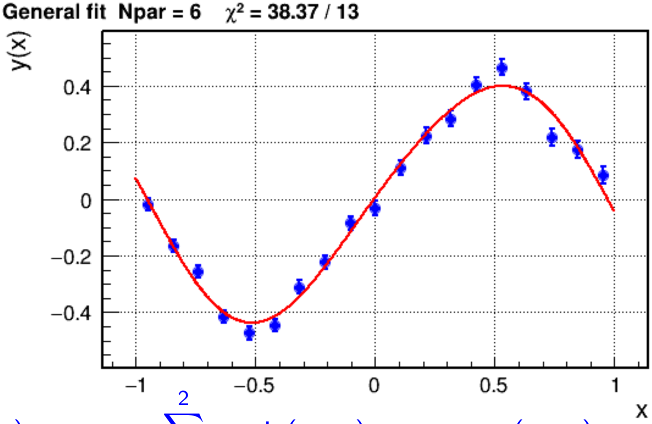
"Wrong" initial values can result in incorrect result $(a_0 = 6)$



$$y(x) = a_1 + \sum_{n=1}^{2} a_{2n} \sin(n a_0 x) + a_{2n+1} \cos(n a_0 x)$$

Marquardt Minimization example

Result of the same fit with "correct" initial values $(a_0 = 3)$



$$y(x) = a_1 + \sum_{n=1}^{2} a_{2n} \sin(n a_0 x) + a_{2n+1} \cos(n a_0 x)$$

Using derivatives

There are many different algorithms with multiple implementations (libraries and programs) for least-square fits of model parameters.

However, general algorithms need to be based on numerical calculation of model function derivatives.

Using dedicated code, with analytical derivative calculation (as in the presented examples) has advantages:

- number of calls to model function significantly reduced (speed)
- better precision of derivatives/numerical stability
- possibility to tune fit performance to particular needs to further improve fit efficiency

Non-linear fit procedure

Using derivatives

There are many different algorithms with multiple implementations (libraries and programs) for least-square fits of model parameters.

However, general algorithms need to be based on numerical calculation of model function derivatives.

Using dedicated code, with analytical derivative calculation (as in the presented examples) has advantages:

- number of calls to model function significantly reduced (speed)
- better precision of derivatives/numerical stability
- possibility to tune fit performance to particular needs to further improve fit efficiency

When large number of fits is to be performed to similar sets of data, one should consider implementing his/hers own fit procedure...

Least-squares method (2)

Comparison of variances

If the precision of the measurement is known, we can calculate the χ^2 value resulting from the fit (or arithmetic averaging in the simplest case) to verify the consistency of the procedure (measurement uncertainty estimate in particular).

However, we can also compare two independent series of measurements to check, if they are consistent. We can do it even, if our estimate of experimental uncertainties is not very reliable.

One can consider it as a procedure to compare two different estimates of the variance of the measurement, and check if they are compatible.

Comparison of variances

If the precision of the measurement is known, we can calculate the χ^2 value resulting from the fit (or arithmetic averaging in the simplest case) to verify the consistency of the procedure (measurement uncertainty estimate in particular).

However, we can also compare two independent series of measurements to check, if they are consistent. We can do it even, if our estimate of experimental uncertainties is not very reliable.

One can consider it as a procedure to compare two different estimates of the variance of the measurement, and check if they are compatible.

We define the random variable F as:

introduced by R.A.Fisher in 1924

$$F = \frac{\chi_1^2/N_1}{\chi_2^2/N_2}$$

where χ_1^2 and χ_2^2 are χ^2 values (or sample variances, if we set $\sigma \equiv 1$) of the two independent measurements with N_1 and N_2 degrees of freedom.

F variable distribution

Distribution of the Fisher's F variable is given by:

$$f(F; N_1, N_2) \;=\; A(N_1, N_2)\; F^{\frac{N_1}{2}-1} \left(1 + \frac{N_1}{N_2}\, F\right)^{-\frac{N_1+N_2}{2}}$$

F variable distribution

Distribution of the Fisher's F variable is given by:

$$f(F; N_1, N_2) = A(N_1, N_2) \; F^{\frac{N_1}{2}-1} \left(1 + \frac{N_1}{N_2} F\right)^{-\frac{N_1+N_2}{2}}$$

Expected (mean) value of F is slightly above one:

$$\langle F \rangle = \frac{N_2}{N_2 - 2} \qquad N_2 > 2$$

and the variance is:

$$\mathbb{V}(F) = \frac{2}{N_1} \cdot \frac{N_2^2 (N_1 + N_2 - 2)}{(N_2 - 2)^2 (N_2 - 4)}$$

F variable distribution

Distribution of the Fisher's F variable is given by:

$$f(F; N_1, N_2) \;=\; A(N_1, N_2)\; F^{\,\frac{N_1}{2}-1} \left(1 + \frac{N_1}{N_2}\,F\right)^{-\frac{N_1+N_2}{2}}$$

Expected (mean) value of F is slightly above one:

$$\langle F \rangle \;=\; \frac{N_2}{N_2 - 2} \qquad\qquad N_2 > 2$$

and the variance is:

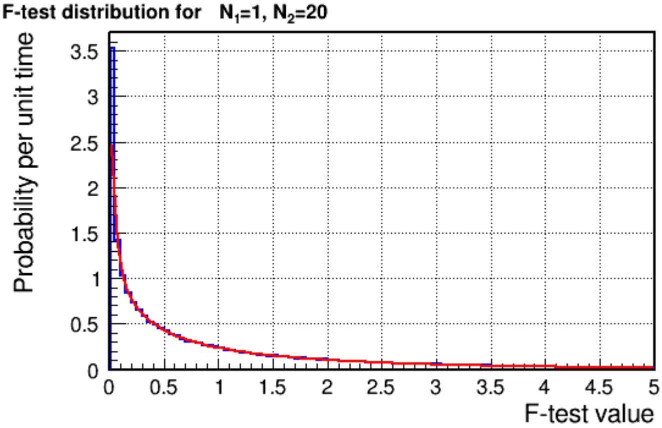
$$\mathbb{V}(F) \;\approx\; \frac{2}{N_1}$$

(for $N_2 \gg N_1$; ratio variance is deominated by the variance of the reduced χ^2 with N_1 d.o.f.)

F variable distribution

09_F-test.ipynb Open in Colab

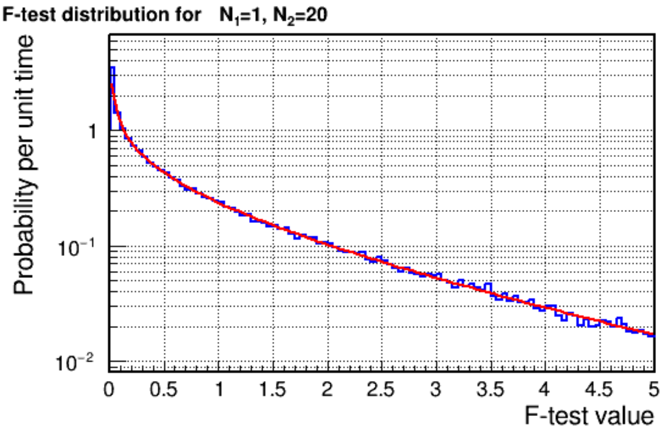
Example distributions of the Fisher's F variable



F variable distribution

09_F-test.ipynb Open in Colab

Example distributions of the Fisher's *F* variable

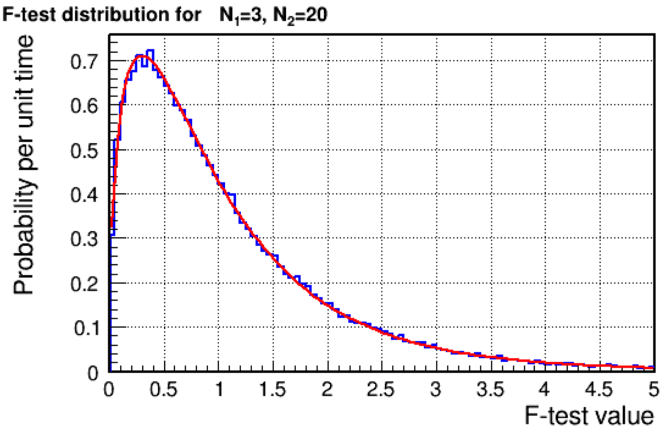


F variable distribution

09_F-test.ipynb

Open in Colab

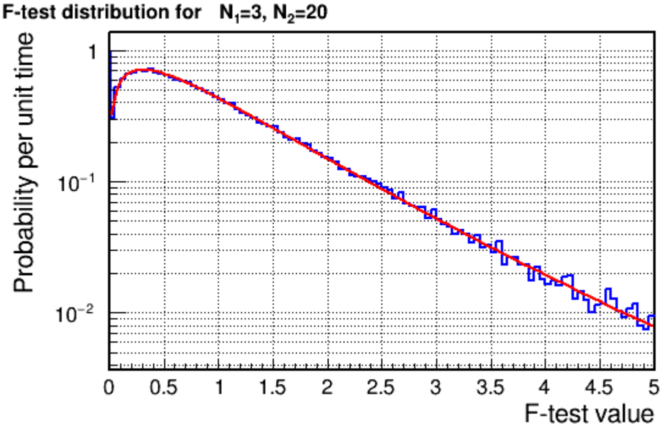
Example distributions of the Fisher's F variable



F variable distribution

09_F-test.ipynb Open in Colab

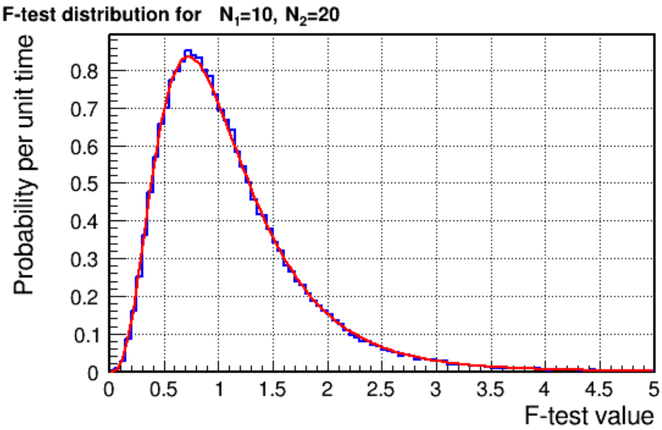
Example distributions of the Fisher's F variable



F variable distribution

09_F-test.ipynb Open in Colab

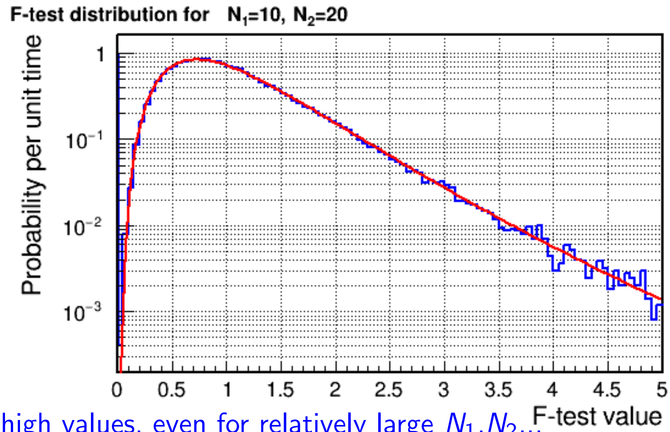
Example distributions of the Fisher's F variable



F variable distribution

09_F-test.ipynb Open in Colab

Example distributions of the Fisher's *F* variable

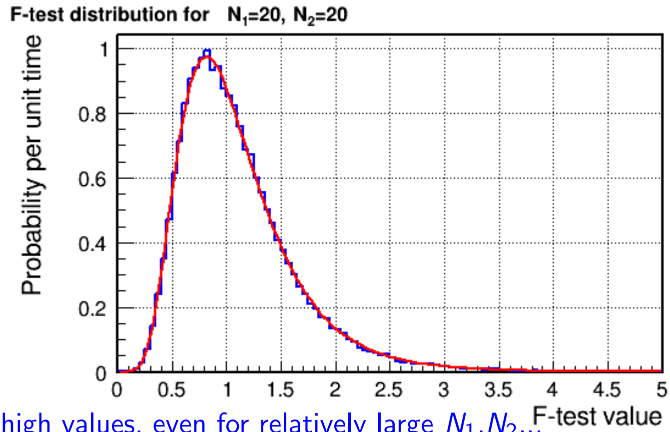


Significant tail of high values, even for relatively large N_1, N_2...

F variable distribution

09_F-test.ipynb Open in Colab

Example distributions of the Fisher's F variable

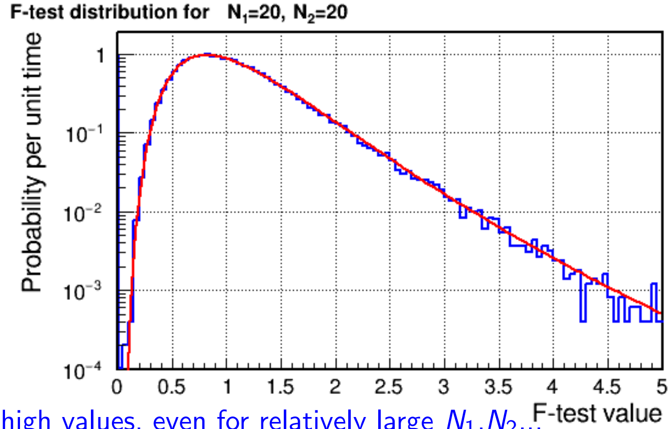


Significant tail of high values, even for relatively large N_1, N_2...

F variable distribution

09_F-test.ipynb Open in Colab

Example distributions of the Fisher's F variable



Significant tail of high values, even for relatively large N_1, N_2...

F-test

When the calculated value of F is large, $F \gg 1$, we can conclude that the two considered data sets are not consistent...

However, we need to be careful, as large fluctuations possible!

Conclusion can depend on the assumed confidence level for F, i.e. the maximum probability we allow for given (or higher) value to result from the statistical fluctuations in the (consistent) data sets.

F-test

When the calculated value of F is large, $F \gg 1$, we can conclude that the two considered data sets are not consistent...

However, we need to be careful, as large fluctuations possible!

Conclusion can depend on the assumed confidence level for F, i.e. the maximum probability we allow for given (or higher) value to result from the statistical fluctuations in the (consistent) data sets.

We can define critical value of F, F_{crit}, corresponding the the frequentist upper limit for given confidence level CL:

$$\int_0^{F_{crit}} dF \; f(F) \; = \; \mathrm{CL} \qquad\qquad \int_{F_{crit}}^{+\infty} dF \; f(F) \; = \; 1 - \mathrm{CL} \; = \; p$$

F-test (Brandt)

"Critical values" of F for small numbers of degrees of freedom N_1 and N_2

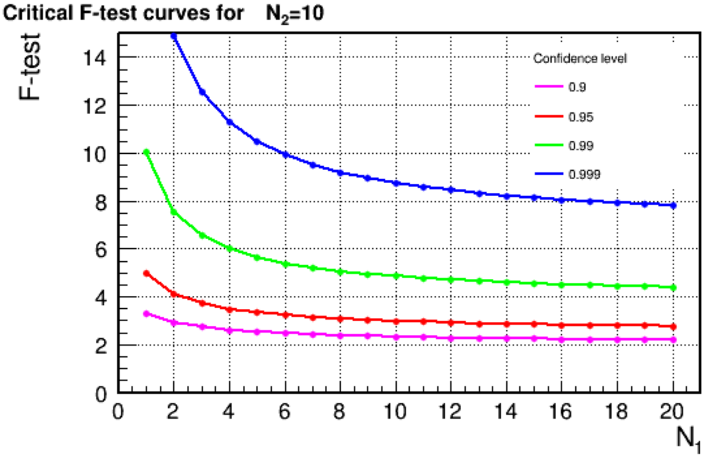
$$\mathrm{CL} = \quad 0.950 = P = \int_0^{F_P} f(F; f_1, f_2)\,\mathrm{d}F$$

N_2 f_2 \ N_1 f_1	1	2	3	4	5	6	7	8	9	10
1	161.4	199.5	215.7	224.6	230.2	234.0	236.8	238.9	240.5	241.9
2	18.51	19.00	19.16	19.25	19.30	19.33	19.35	19.37	19.38	19.40
3	10.13	9.552	9.277	9.117	9.013	8.941	8.887	8.845	8.812	8.786
4	7.709	6.944	6.591	6.388	6.256	6.163	6.094	6.041	5.999	5.964
5	6.608	5.786	5.409	5.192	5.050	4.950	4.876	4.818	4.772	4.735
6	5.987	5.143	4.757	4.534	4.387	4.284	4.207	4.147	4.099	4.060
7	5.591	4.737	4.347	4.120	3.972	3.866	3.787	3.726	3.677	3.637
8	5.318	4.459	4.066	3.838	3.687	3.581	3.500	3.438	3.388	3.347
9	5.117	4.256	3.863	3.633	3.482	3.374	3.293	3.230	3.179	3.137
10	4.965	4.103	3.708	3.478	3.326	3.217	3.135	3.072	3.020	2.978

F-test

09_F-limit.ipynb Open in Colab

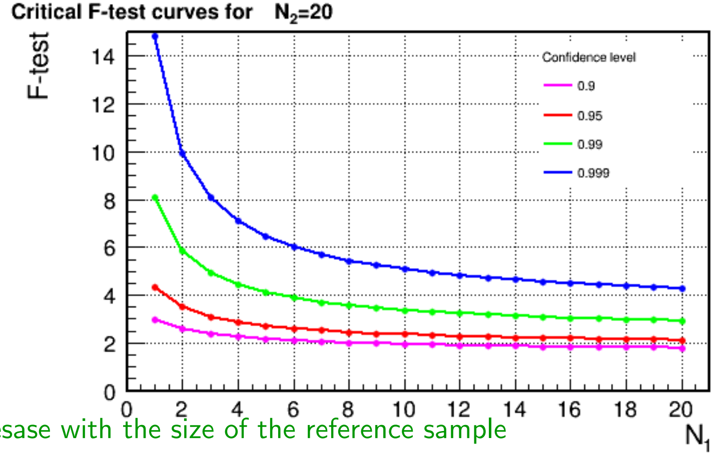
Plot of "critical values" of F for $N_2 = 10$



F-test

09_F-limit.ipynb Open in Colab

Plot of "critical values" of F for N_2 =20



Variations decresase with the size of the reference sample

F-test for fit model (Bonamente)

It turns out that the F variable can also be used to test our fit model.

Let us assume we have a model with m adjustable (free) parameters, which we apply to the set of N data points. Are all model parameters relevant?

F-test for fit model (Bonamente)

It turns out that the F variable can also be used to test our fit model.

Let us assume we have a model with m adjustable (free) parameters, which we apply to the set of N data points. Are all model parameters relevant?

We can consider "reduced" version of model with $m - \Delta m$ parameters.
It is clear that the resulting χ^2 value will be larger:

$$\chi^2_{(m-\Delta m)} \;=\; \chi^2_{(m)} + \Delta\chi^2$$

F-test for fit model (Bonamente)

It turns out that the F variable can also be used to test our fit model.

Let us assume we have a model with m adjustable (free) parameters, which we apply to the set of N data points. Are all model parameters relevant?

We can consider "reduced" version of model with $m - \Delta m$ parameters.
It is clear that the resulting χ^2 value will be larger:

$$\chi^2_{(m-\Delta m)} \;=\; \chi^2_{(m)} + \Delta\chi^2$$

If reduced model is equivalent to the "full" one, distribution of $\Delta\chi^2$ is given by the χ^2 distribution with Δm degrees of freedom. We can test this hypothesis by considering:

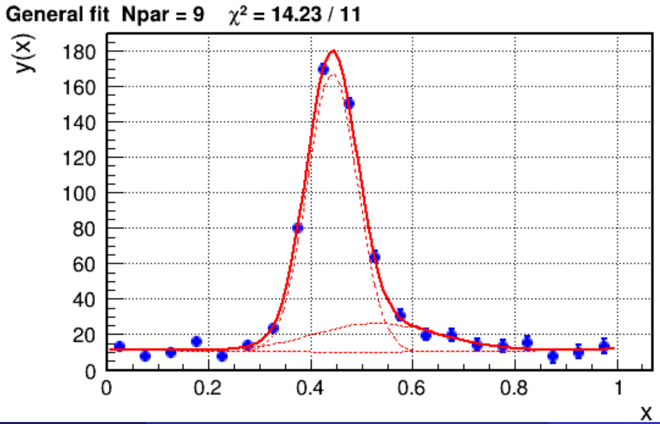
$$F \;=\; \frac{\Delta\chi^2/\Delta m}{\chi^2_{(m)}/(N-m)}$$

where we need to note that $\chi^2_{(m)}$ and $\Delta\chi^2$ are independent.

F-test for fit model

09_fit3.ipynb Open in Colab

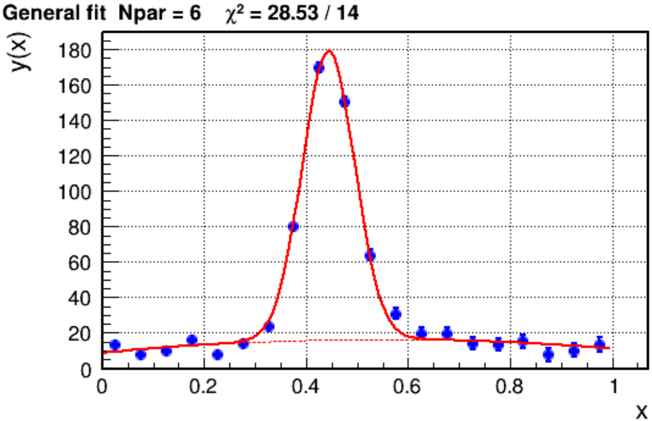
Example of F-test application. We can try to fit the data with polynomial background (3 parameters) and two Gaussian peaks (3+3 parameters).



F-test for fit model

09_fit3.ipynb Open in Colab

Example of F-test application. Is the second signal component really needed?!
Fit with one peak gives a reasonable description of the data as well.



F-test for fit model

Example of F-test application.

$$N = 20 \qquad m = 9$$

$$\chi^2_{(m)} \approx 14.23 \qquad \chi^2_{(m)}/(N-m) \approx 1.294$$

F-test for fit model

Example of F-test application.

$$N = 20 \qquad m = 9 \qquad \Delta m = 3$$

$$\chi^2_{(m)} \approx 14.23 \qquad \chi^2_{(m)}/(N-m) \approx 1.294$$

$$\Delta\chi^2 \approx 14.33 \qquad \Delta\chi^2/\Delta m \approx 4.777$$

F-test for fit model

Example of F-test application.

$$N = 20 \quad m = 9 \quad \Delta m = 3$$

$$\chi^2_{(m)} \approx 14.23 \qquad \chi^2_{(m)}/(N-m) \approx 1.294$$

$$\Delta\chi^2 \approx 14.33 \qquad \Delta\chi^2/\Delta m \approx 4.777$$

$$F \approx 3.69$$

$$p \approx 0.047$$

There is about 5% chance that the improvement in the fit result, when adding the second signal component, was only due to the statistical fluctuations in the data...

F-test for fit model

Example of F-test application.

$$N = 20 \qquad m = 9 \qquad \Delta m = 3$$

$$\chi^2_{(m)} \approx 14.23 \qquad \chi^2_{(m)}/(N-m) \approx 1.294$$

$$\Delta\chi^2 \approx 14.33 \qquad \Delta\chi^2/\Delta m \approx 4.777$$

$$F \approx 3.69$$

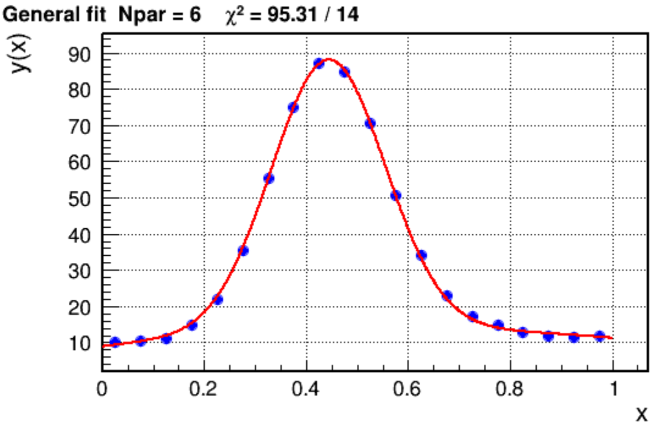
$$p \approx 0.047$$

There is about 5% chance that the improvement in the fit result, when adding the second signal component, was only due to the statistical fluctuations in the data...

Second component fit result misleading: $A_2 = 4.950 \pm 1.523$ (3.25σ)

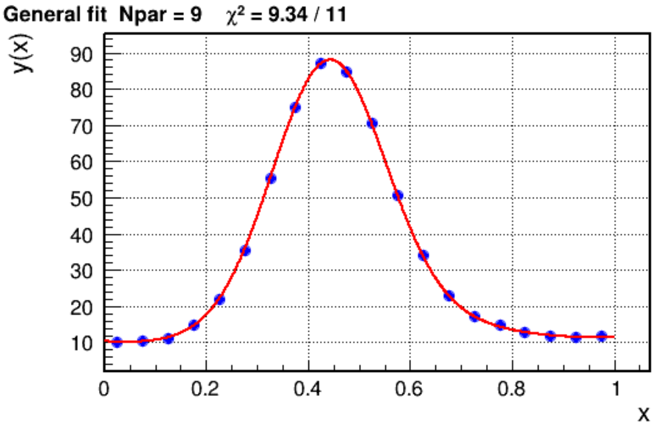
F-test for fit model

Example (2). Significantly reduced measurement errors. Fit with one signal component.



F-test for fit model

Example (2). Significantly reduced measurement errors. Fit with two signal components.



F-test for fit model

Example (2) of F-test application.

$$N = 20 \qquad m = 9 \qquad \Delta m = 3$$

$$\chi^2_{(m)} \approx 9.34 \qquad \chi^2_{(m)}/(N-m) \approx 0.849$$

$$\Delta\chi^2 \approx 85.97 \qquad \Delta\chi^2/\Delta m \approx 28.66$$

$$F \approx 33.75$$
$$p \approx 8 \cdot 10^{-6}$$

corresponding to about 4.3 σ deviation...

Can not yet be claimed as discovery, but significant enough to be shown...

Least-squares method (2)

Model constraints

Different quantities measured in our experiment can be related.

Relations, which are usually given in form of equations, can be due to theoretical predictions, model assumptions or experimental constraints.
For example:

- measurements of energy and momentum of the particle, are related by the invariant mass formula $E^2 = p^2 + m^2$
- contribution of different channels to the particle decays are related by the total branching ration of 100%

Model constraints

Different quantities measured in our experiment can be related.

Relations, which are usually given in form of equations, can be due to theoretical predictions, model assumptions or experimental constraints.

For example:

- measurements of energy and momentum of the particle, are related by the invariant mass formula $E^2 = p^2 + m^2$
- contribution of different channels to the particle decays are related by the total branching ration of 100%
- contribution of different components to the measured distribution can be constrained by the total normalization of the sample
- distributions can be constrained by the acceptance of the detector

Model constraints

Different quantities measured in our experiment can be related.

Relations, which are usually given in form of equations, can be due to theoretical predictions, model assumptions or experimental constraints.

For example:

- measurements of energy and momentum of the particle, are related by the invariant mass formula $E^2 = p^2 + m^2$
- contribution of different channels to the particle decays are related by the total branching ration of 100%
- contribution of different components to the measured distribution can be constrained by the total normalization of the sample
- distributions can be constrained by the acceptance of the detector
- we can make additional assumptions regarding eg. symmetry of the distribution or asymptotic behavior

Model constraints

We consider set of N measurement points (x_i, y_i), which can be compared to model predictions depending on parameters $\mathbf{a}$ in terms of the χ^2 value:

$$\chi^2(\mathbf{a}) \quad = \quad \sum_{i=1}^{N} \frac{(y_i - \mu(x_i, \mathbf{a}))^2}{\sigma_i^2}$$

Best estimate of $\mathbf{a}$ should correspond to the minimum of $\chi^2(\mathbf{a})$.

Model constraints

We consider set of N measurement points (x_i, y_i), which can be compared to model predictions depending on parameters **a** in terms of the χ^2 value:

$$\chi^2(\mathbf{a}) \;=\; \sum_{i=1}^{N} \frac{(y_i - \mu(x_i, \mathbf{a}))^2}{\sigma_i^2}$$

Best estimate of **a** should correspond to the minimum of $\chi^2(\mathbf{a})$.

However, we now need to look for this minimum taking additional constraints into account:

$$w_k(\mathbf{a}) \;=\; 0 \qquad\qquad k = 1 \ldots K$$

where number of constraints K should be lower than number of parameters M.

Model constraints

We consider set of N measurement points (x_i, y_i), which can be compared to model predictions depending on parameters $\mathbf{a}$ in terms of the χ^2 value:

$$\chi^2(\mathbf{a}) = \sum_{i=1}^{N} \frac{(y_i - \mu(x_i, \mathbf{a}))^2}{\sigma_i^2}$$

Best estimate of $\mathbf{a}$ should correspond to the minimum of $\chi^2(\mathbf{a})$.

However, we now need to look for this minimum taking additional constraints into account:

$$w_k(\mathbf{a}) = 0 \qquad k = 1 \ldots K$$

where number of constraints K should be lower than number of parameters M.

How can we find the best parameter values in this case?

Model reduction

The first approach is to reduce number of model parameters, using constraints to eliminate some of the independent model variables. $\Rightarrow$ We thus reduce the problem with M model parameters to problem with $M' = M - K$ independent parameters. (method of elements)

Model reduction

The first approach is to reduce number of model parameters, using constraints to eliminate some of the independent model variables. $\Rightarrow$ We thus reduce the problem with M model parameters to problem with $M' = M - K$ independent parameters. (method of elements)

Example

We would like to fit polynomial model to a series of measurements where the azimuthal angle $\theta \in [-\pi, +\pi]$ is the controlled variable:

$$\mu(x; \mathbf{a}) = \sum_{k=0}^{M-1} a_k \left(\frac{\theta}{\pi}\right)^k = \sum_k a_k \, x^k$$

where we introduced $x = \frac{\theta}{\pi}$ for simplicity, $w \in [-1, +1]$.

And we expect that the distribution should vanish for $\theta \to \pm\pi$:

$$\mu(-1; \mathbf{a}) = \mu(+1; \mathbf{a}) = 0 \qquad K = 2$$

Model reduction example

Our constraints give us direct relations between model parameters:

$$\mu(+1; \mathbf{a}) = \sum_k a_k \, (+1)^k = 0$$

$$\mu(-1; \mathbf{a}) = \sum_k a_k \, (-1)^k = 0$$

Model reduction example

Our constraints give us direct relations between model parameters:

$$\mu(+1;\mathbf{a}) = \sum_k a_k \, (+1)^k = 0$$

$$\mu(-1;\mathbf{a}) = \sum_k a_k \, (-1)^k = 0$$

We can add and subtract these two equations to obtain:

$$\sum_{k=0,2,4\ldots} a_k = 0 \qquad \text{and} \qquad \sum_{k=1,3,5\ldots} a_k = 0$$

$\Rightarrow$ we have independent constrains on even and odd parameters.

Model reduction example

Our constraints give us direct relations between model parameters:

$$\begin{aligned}\mu(+1;\mathbf{a}) &= \sum_k a_k\,(+1)^k = 0\\ \mu(-1;\mathbf{a}) &= \sum_k a_k\,(-1)^k = 0\end{aligned}$$

We can add and subtract these two equations to obtain:

$$\sum_{k=0,2,4\ldots} a_k = 0 \qquad \text{and} \qquad \sum_{k=1,3,5\ldots} a_k = 0$$

$\Rightarrow$ we have independent constrains on even and odd parameters.

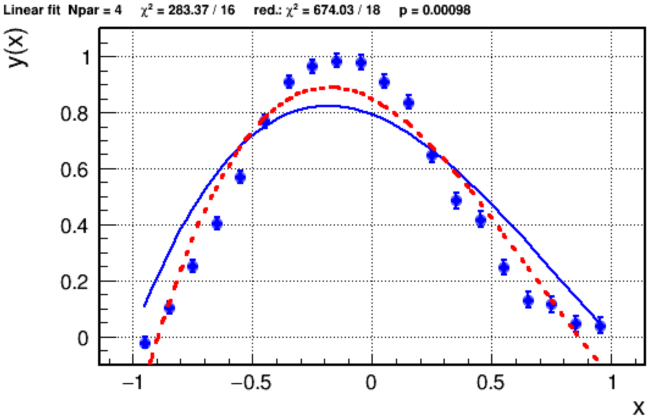
We can reduce number of parameters to $M' = M - 2$ by writing parameters for the two highest terms (for even M) as:

$$a_{M-2} = -\sum_{k=0,2,4\ldots}^{M-4} a_k \qquad \text{and} \qquad a_{M-1} = -\sum_{k=1,3,5\ldots}^{M-3} a_k$$

Model reduction example

09_reduce.ipynb Open in Colab

Example polynomial fit without and with model constraints $\mu(-1) = \mu(+1) = 0$

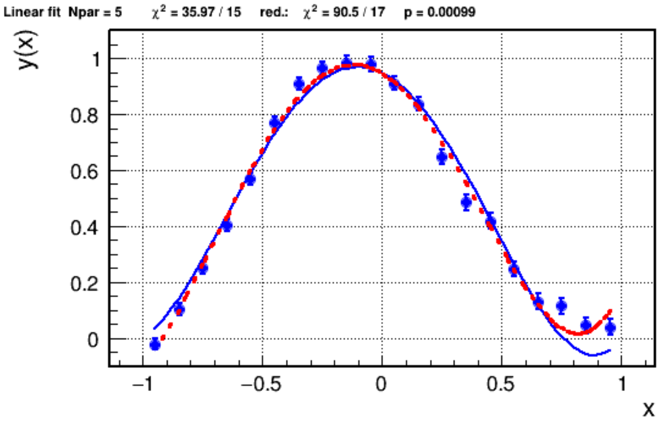


3^{rd} order polynomial not describing data at all

Model reduction example

09_reduce.ipynb Open in Colab

Example polynomial fit without and with model constraints $\mu(-1) = \mu(+1) = 0$

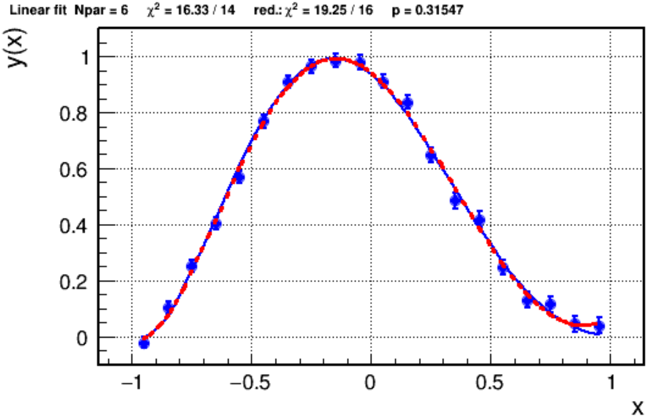


4^{th} order polynomial fit still not satisfactory...

Model reduction example

09_reduce.ipynb Open in Colab

Example polynomial fit without and with model constraints $\mu(-1) = \mu(+1) = 0$

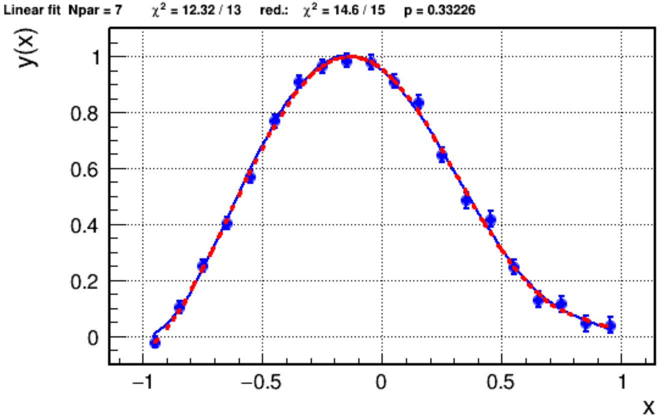


Reasonable description of the data.

Model reduction example

09_reduce.ipynb Open in Colab

Example polynomial fit without and with model constraints $\mu(-1) = \mu(+1) = 0$

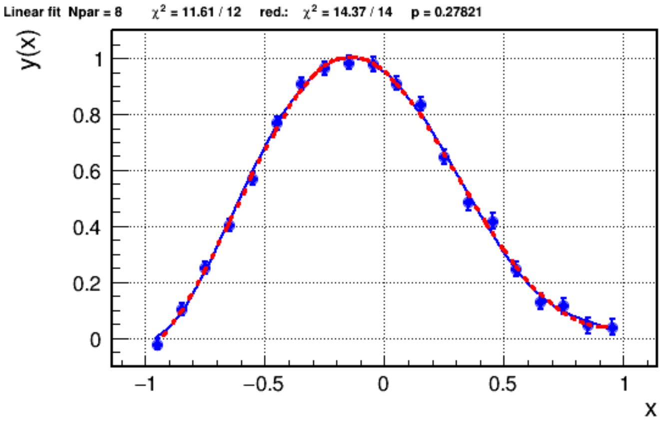


Even better description of the data. Constraints hardly affect the fit...

Model reduction example

09_reduce.ipynb Open in Colab

Example polynomial fit without and with model constraints $\mu(-1) = \mu(+1) = 0$



Even better description of the data. Constraints hardly affect the fit...

Model reduction test

for linear constraints !

We can use F-test to verify, if data are consistent with constraints.

We follow the procedure described previously: we have a model with m free parameters and Δm constraints, which we test on the set of N data points

Constrained ("reduced") model has $m - \Delta m$ parameters and results in higher χ^2:

$$\Delta\chi^2 \;=\; \chi^2_{\text{reduced}} - \chi^2_{\text{full}} \;\geq\; 0$$

To test if the reduced model is equivalent to the "full" one, we consider:

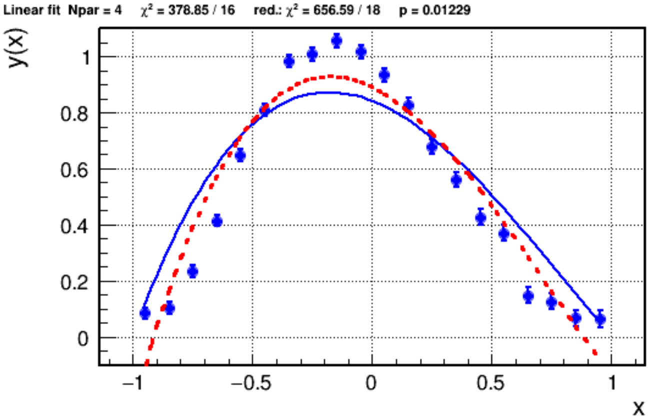
$$F = \frac{\Delta\chi^2/\Delta m}{\chi^2_{\text{full}}/(N-m)} \qquad \text{and} \qquad p \;=\; \int_F^{+\infty} dF' \, f(F')$$

p value (indicated in the plots) describe the probability of given χ^2 change (when imposing constraints) due to statistical fluctuations only.

Model reduction example (2)

(generated with $\mu(-1) = \mu(+1) = 0.05$)

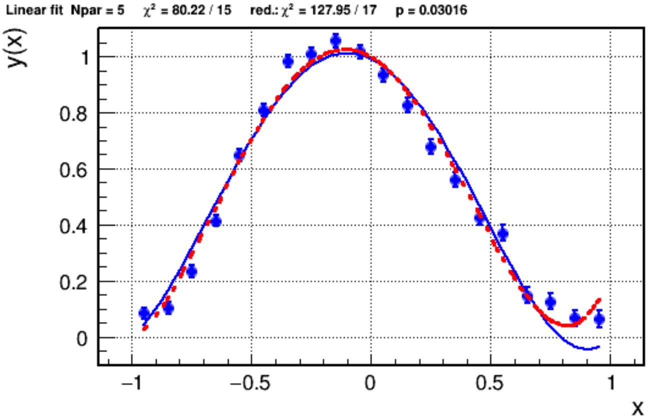
Example polynomial fit without and with model constraints, to inconsistent data set



3[rd] order polynomial not describing data at all

Model reduction example (2)

Example polynomial fit without and with model constraints, to inconsistent data set



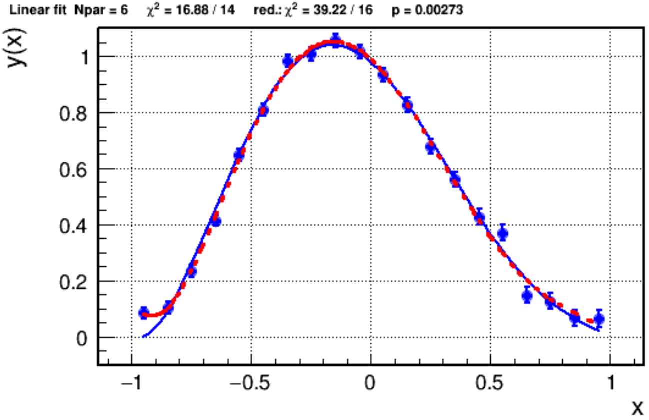
4^{th} order polynomial fit still not satisfactory...

Constrained fit

Model reduction example (2)

09_reduce2.ipynb Open in Colab

Example polynomial fit without and with model constraints, to inconsistent data set



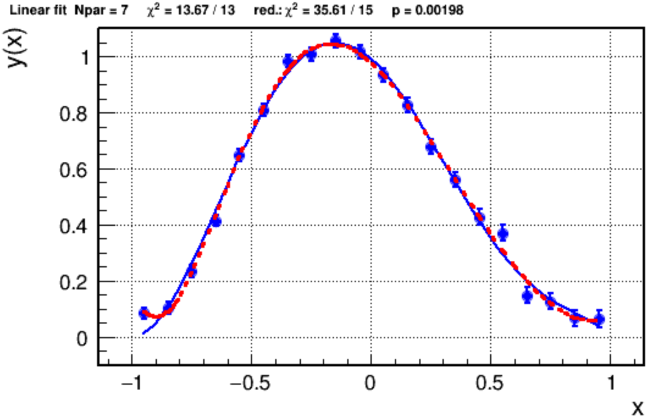
Reasonable description of the data only without constraints...

Constrained fit

Model reduction example (2)

09_reduce2.ipynb Open in Colab

Example polynomial fit without and with model constraints, to inconsistent data set

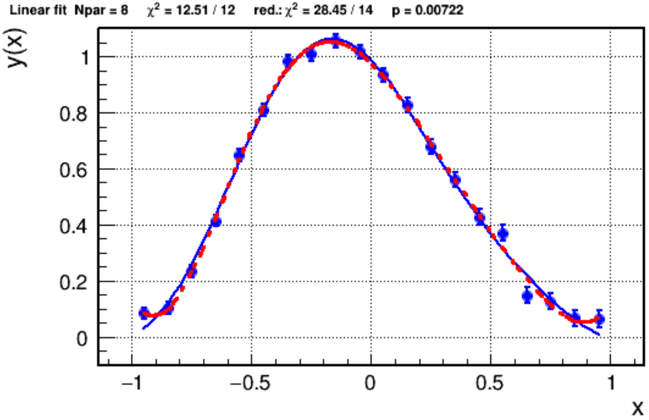


p value indicates that data is not consistent with constraints...

Model reduction example (2)

09_reduce2.ipynb Open in Colab

Example polynomial fit without and with model constraints, to inconsistent data set



p value indicates that data is not consistent with constraints...

Model reduction

After fitting the reduced model, constraints can be used to extract values of eliminated parameters and their uncertainties can be calculated from standard error propagation...

Model reduction

After fitting the reduced model, constraints can be used to extract values of eliminated parameters and their uncertainties can be calculated from standard error propagation...

When trying to reduce the number of parameters, one should consider redefining the parameter basis in such a way that constraints depend only on selected parameters.
Can be essential for numerical stability...

Model reduction

After fitting the reduced model, constraints can be used to extract values of eliminated parameters and their uncertainties can be calculated from standard error propagation...

When trying to reduce the number of parameters, one should consider redefining the parameter basis in such a way that constraints depend only on selected parameters.
Can be essential for numerical stability...

However, elimination of variables is not always possible or can result in problems with minimization / numerical instabilities.

Also, when iterative fit procedure is to be used, initial parameter values should be "guessed", but imposing constraints "by hand" can be difficult.

$\Rightarrow$ while model reduction is a preferred solution, resulting in higher fit efficiency in most cases, we need an alternative...

Method of Lagrange Multipliers (Behnke)

The method, invented by J.L.Lagrange in 1788, applies to general minimization problem with additional constraints imposed.

Problem of finding minimum of χ^2 $(\mathbf{a})$ with constraints $w_k(\mathbf{a}) = 0$ is equivalent to finding a stationary point (point with all first derivatives at zero) of the Lagrange function:

$$\mathcal{L}(\mathbf{a}, \boldsymbol{\lambda}) \;=\; \chi^2(\mathbf{a}) + \sum_k 2\lambda_k w_k(\mathbf{a})$$

where we introduce additional K parameters λ_k - Lagrange multipliers

Method of Lagrange Multipliers

(Behnke)

The method, invented by J.L.Lagrange in 1788, applies to general minimization problem with additional constraints imposed.

Problem of finding minimum of $\chi^2\,(\mathbf{a})$ with constraints $w_k(\mathbf{a}) = 0$ is equivalent to finding a stationary point (point with all first derivatives at zero) of the Lagrange function:

$$\mathcal{L}(\mathbf{a}, \boldsymbol{\lambda}) \;=\; \chi^2(\mathbf{a}) + \sum_k 2\lambda_k w_k(\mathbf{a})$$

where we introduce additional K parameters λ_k - Lagrange multipliers

Our problem is now reduced to finding parameters $\mathbf{a}$ and $\boldsymbol{\lambda}$ fulfilling

$$\frac{\partial \mathcal{L}}{\partial a_j} \;=\; 0 \qquad \text{and} \qquad \frac{\partial \mathcal{L}}{\partial \lambda_k} \;=\; 0$$

(without any additional constraints)

Method of Lagrange Multipliers

Writing the full formula for $\mathcal{L}$:

$\mu_i = \mu(x_i; \mathbf{a})$

$$\mathcal{L}(\mathbf{a}, \boldsymbol{\lambda}) = \sum_{i=1}^{N} \frac{(y_i - \mu_i)^2}{\sigma_i^2} + \sum_{k=1}^{K} 2\, \lambda_k \, w_k(\mathbf{a})$$

We obtain a set of $M + K$ equations for the same number of parameters:

$$\frac{\partial \mathcal{L}}{\partial a_j} = -2 \sum_{i=1}^{N} \frac{(y_i - \mu_i)}{\sigma_i^2} \frac{\partial \mu_i}{\partial a_j} + 2 \sum_{k=1}^{K} \lambda_k \frac{\partial w_k}{\partial a_j} \qquad j = 1 \ldots M$$

Method of Lagrange Multipliers

Writing the full formula for $\mathcal{L}$: $\mu_i = \mu(x_i; \mathbf{a})$

$$\mathcal{L}(\mathbf{a}, \boldsymbol{\lambda}) = \sum_{i=1}^{N} \frac{(y_i - \mu_i)^2}{\sigma_i^2} + \sum_{k=1}^{K} 2\,\lambda_k\, w_k(\mathbf{a})$$

We obtain a set of $M + K$ equations for the same number of parameters:

$$\frac{\partial \mathcal{L}}{\partial a_j} = -2 \sum_{i=1}^{N} \frac{(y_i - \mu_i)}{\sigma_i^2} \frac{\partial \mu_i}{\partial a_j} + 2 \sum_{k=1}^{K} \lambda_k \frac{\partial w_k}{\partial a_j} \qquad j = 1 \ldots M$$

$$\frac{\partial \mathcal{L}}{\partial \lambda_k} = 2\, w_k(\mathbf{a}) = 0 \qquad k = 1 \ldots K$$

Adding constraints modifies the resulting system of equation for **a**...

Method of Lagrange Multipliers

Let us consider the linear problem (linear in **a**):

$$\mu(x;\mathbf{a}) = \sum_{j=1}^{M} a_j \, f_j(x) \qquad \text{and} \qquad w_k(\mathbf{a}) = \sum_{j=1}^{M} d_{k,\,j} \, a_j - c_k$$

Our set of equations can be now presented as:

$$\sum_{j=1}^{M} \left(\sum_{i=1}^{N} \frac{f_l(x_i)\, f_k(x_i)}{\sigma_i^2} \right) a_j + \sum_{k=1}^{K} d_{k,\,l}\, \lambda_k = \sum_{i=1}^{N} \frac{f_l(x_i)\, y_i}{\sigma_i^2} \qquad l = 1 \ldots M$$

$$\sum_{j=1}^{M} d_{k,\,j}\, a_j = c_k \qquad k = 1 \ldots K$$

Method of Lagrange Multipliers

We can write these equations the matrix form:

$$\underbrace{\left(\begin{array}{c|c} \mathbb{A} & \mathbb{D} \\ \hline \mathbb{D}^{\mathsf{T}} & 0 \end{array}\right)}_{\tilde{\mathbb{A}}} \cdot \left(\begin{array}{c} \mathbf{a} \\ \hline \boldsymbol{\lambda} \end{array}\right) = \left(\begin{array}{c} \mathbf{b} \\ \hline \mathbf{c} \end{array}\right)$$

where: $\mathbb{A}_{jk} = \sum_{i=1}^{N} \frac{f_j(x_i)\, f_k(x_i)}{\sigma_i^2}$, $\mathbb{D}_{jk} = d_{k,\,j}$ and $b_j = \sum_{i=1}^{N} \frac{f_j(x_i)\, y_i}{\sigma_i^2}$

and the problem can be solved by inverting matrix $\tilde{\mathbb{A}}$.

Method of Lagrange Multipliers

We can write these equations the matrix form:

$$\underbrace{\left(\begin{array}{c|c} \mathbb{A} & \mathbb{D} \\ \hline \mathbb{D}^\mathsf{T} & 0 \end{array}\right)}_{\tilde{\mathbb{A}}} \cdot \left(\begin{array}{c} \mathbf{a} \\ \hline \boldsymbol{\lambda} \end{array}\right) = \left(\begin{array}{c} \mathbf{b} \\ \hline \mathbf{c} \end{array}\right)$$

where: $\mathbb{A}_{jk} = \sum_{i=1}^{N} \frac{f_j(x_i)\, f_k(x_i)}{\sigma_i^2}$, $\mathbb{D}_{jk} = d_{k,j}$ and $b_j = \sum_{i=1}^{N} \frac{f_j(x_i)\, y_i}{\sigma_i^2}$

and the problem can be solved by inverting matrix $\tilde{\mathbb{A}}$.

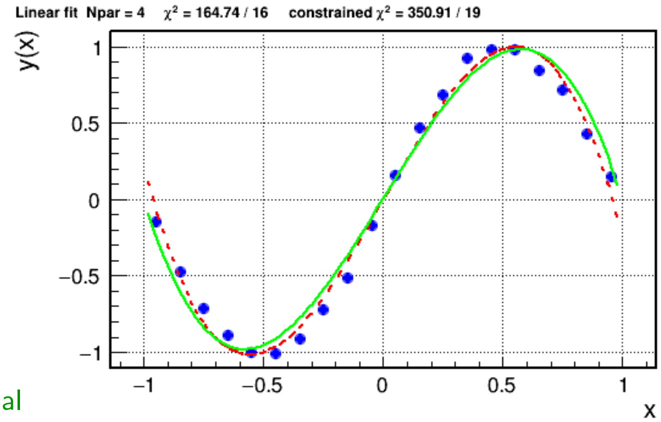
Covariance matrix for **a** can be extracted as: (seems to work for linear problems)

$$(\mathbb{C}_\mathbf{a})_{ij} = (\tilde{\mathbb{A}}^{-1})_{ij} \qquad i,j = 1 \ldots M$$

Lagrange Multipliers example

09_const.ipynb Open in Colab

Fitting $\sin(\pi x)$ with polynomial. Constraints: $\mu(-1) = \mu(0) = \mu(+1) = 0$.

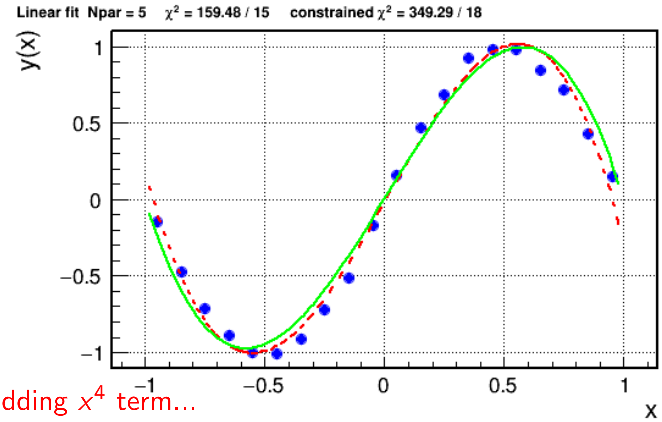


3^{rd} order polynomial

Lagrange Multipliers example

09_const.ipynb Open in Colab

Fitting $\sin(\pi x)$ with polynomial. Constraints: $\mu(-1) = \mu(0) = \mu(+1) = 0$.

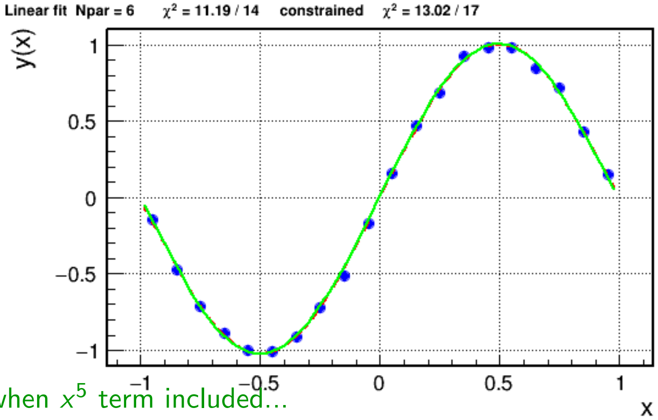


No improvement adding x^4 term...

Lagrange Multipliers example

09_const.ipynb Open in Colab

Fitting $\sin(\pi x)$ with polynomial. Constraints: $\mu(-1) = \mu(0) = \mu(+1) = 0$.



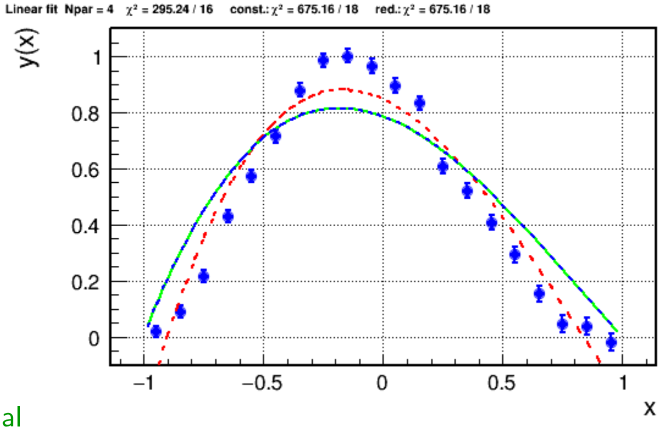
Good description when x^5 term included...

Lagrange Multipliers example (2)

09_const2.ipynb Open in Colab

Comparing two approaches to polynomial fit with constraints.

Fit approach: unconstrained Lagrange multipliers model reduction



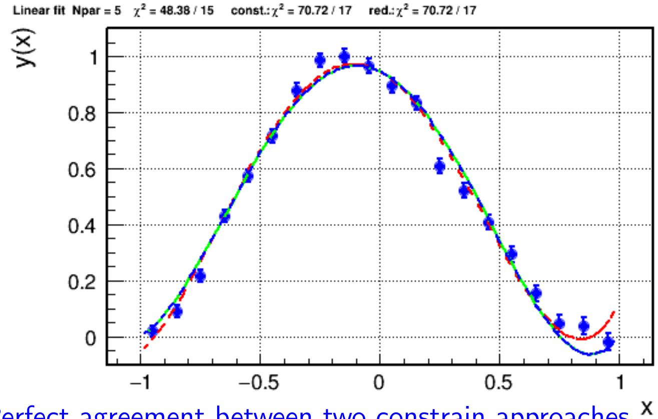
3^{rd} order polynomial

Lagrange Multipliers example (2)

09_const2.ipynb Open in Colab

Comparing two approaches to polynomial fit with constraints.

Fit approach: unconstrained Lagrange multipliers model reduction



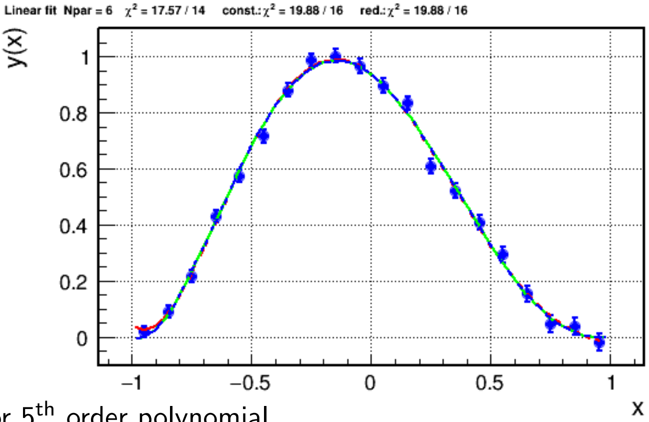
With x^4 term Perfect agreement between two constrain approaches

Lagrange Multipliers example (2)

09_const2.ipynb   Open in Colab

Comparing two approaches to polynomial fit with constraints.

Fit approach: unconstrained   Lagrange multipliers   model reduction



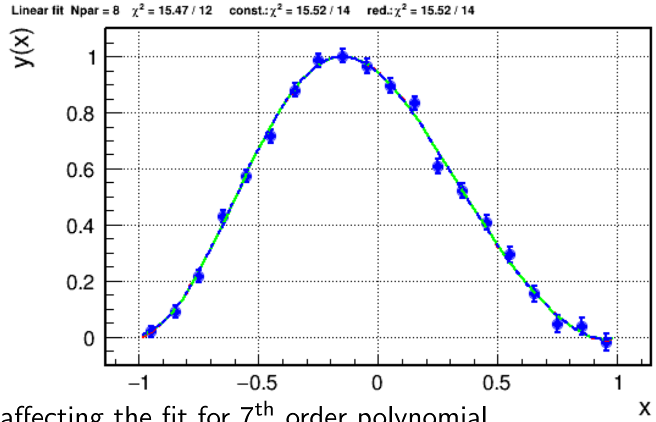
Good description for 5[th] order polynomial...

Lagrange Multipliers example (2)

09_const2.ipynb Open in Colab

Comparing two approaches to polynomial fit with constraints.

Fit approach: unconstrained Lagrange multipliers model reduction



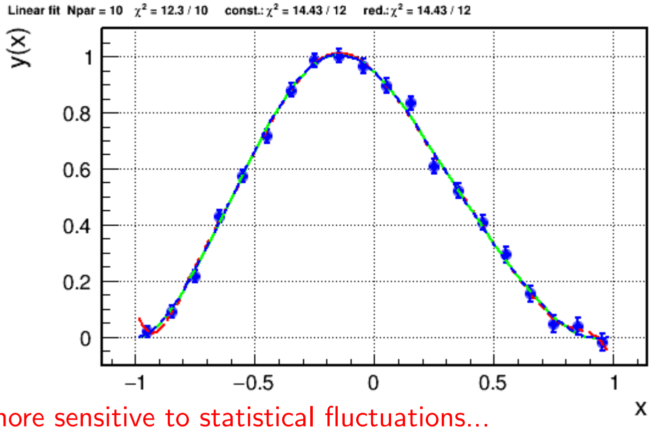
Constraints hardly affecting the fit for 7^{th} order polynomial...

Lagrange Multipliers example (2)

09_const2.ipynb Open in Colab

Comparing two approaches to polynomial fit with constraints.

Fit approach: unconstrained   Lagrange multipliers   model reduction



Unconstrained fit more sensitive to statistical fluctuations...

Least-squares method (2)

Homework

Solutions to be uploaded by December 21.

Consider outcome of the experiment, as illustrated in the figure.

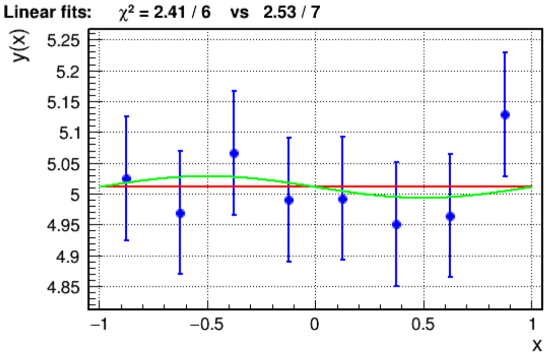
$N = 8$ measurements generated from uniform background model ($\mu = 5$, $\sigma = 0.1$)

Results can be compared to background only model and to background+signal:

$$\mu_b = A \quad \text{and} \quad \mu_{b+s} = B + C \cdot \sin(\pi x)$$

- Estimate probability of fitted C value deviating from zero by more than 3σ
- Verify estimate with simulation

Hint: $\mathbb{A}$ and $\mathbb{C}_{\mathbf{a}}$ do not depend on y_i



Homework

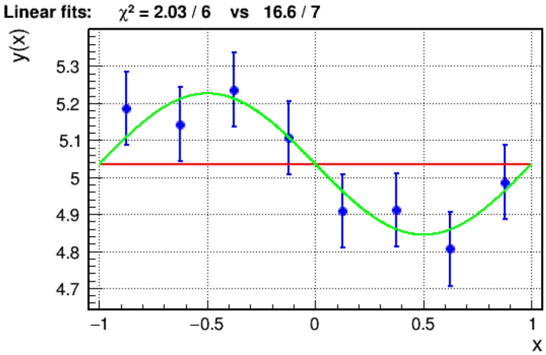
Solutions to be uploaded by December 21.

Consider outcome of the experiment, as illustrated in the figure.
$N = 8$ measurements generated from uniform background model ($\mu = 5$, $\sigma = 0.1$)

Results can be compared to background only model and to background+signal:

$$\mu_b \; = \; A \qquad \text{and} \qquad \mu_{b+s} \; = \; B + C \cdot \sin(\pi x)$$

- Estimate probability of fitted C value deviating from zero by more than 3σ
- Verify estimate with simulation



Hint: $\mathbb{A}$ and $\mathbb{C}_{\mathbf{a}}$ do not depend on y_i

Homework

Solutions to be uploaded by December 21.

Consider outcome of the experiment, as illustrated in the figure.

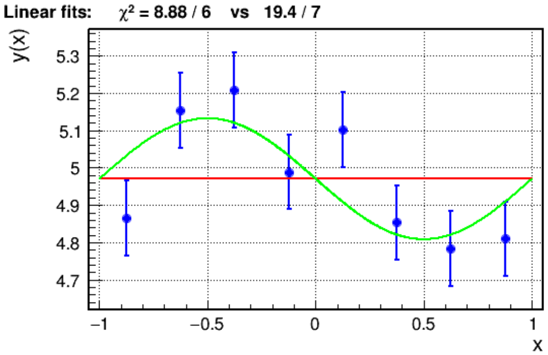
$N = 8$ measurements generated from uniform background model ($\mu = 5$, $\sigma = 0.1$)

Results can be compared to background only model and to background+signal:

$$\mu_b = A \quad \text{and} \quad \mu_{b+s} = B + C \cdot \sin(\pi x)$$

- Estimate probability of fitted C value deviating from zero by more than 3σ
- Verify estimate with simulation

Hint: $\mathbb{A}$ and $\mathbb{C}_{\mathbf{a}}$ do not depend on y_i



Homework

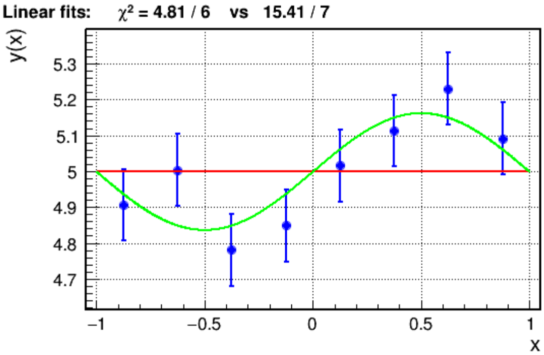
Solutions to be uploaded by December 21.

Consider outcome of the experiment, as illustrated in the figure.
$N = 8$ measurements generated from uniform background model ($\mu = 5$, $\sigma = 0.1$)

Results can be compared to background only model and to background+signal:

$$\mu_b = A \quad \text{and} \quad \mu_{b+s} = B + C \cdot \sin(\pi x)$$

- Estimate probability of fitted C value deviating from zero by more than 3σ
- Verify estimate with simulation



Hint: $\mathbb{A}$ and $\mathbb{C}_{\mathbf{a}}$ do not depend on y_i

Homework

Solutions to be uploaded by December 21.

Consider outcome of the experiment, as illustrated in the figure.

$N = 8$ measurements generated from uniform background model ($\mu = 5$, $\sigma = 0.1$)

Results can be compared to background only model and to background+signal:

$$\mu_b = A \quad \text{and} \quad \mu_{b+s} = B + C \cdot \sin(\pi x)$$

- Estimate probability of fitted C value deviating from zero by more than 3σ
- Verify estimate with simulation

Hint: $\mathbb{A}$ and $\mathbb{C}_{\mathbf{a}}$ do not depend on y_i