HighRR TFR Hands-On: χ^2 stuff – Episode 1

Alessio Piucci

30 May 2017
HighRR TFR Hands-On

x^2 distribution

Given k degrees of freedom:
x^2 is the distribution of a sum of squares of k ind. random variables

$$Q = Z_1^2 + Z_2^2 + \dots + Z_k^2 \quad \rightarrow Q \sim x^2(k)$$

How to use the x^2 to compare data with models?

$$x^2 = (x_1 - \mu_1)^2/\sigma_1^2 + \dots + (x_k - \mu_k)^2/\sigma_k^2$$

$$\rightarrow \;=\; 1 \;+ \dots +\; 1 \;= k \;\rightarrow\; x^2 / k = 1$$

with 'correct' model

More info in Wiki / Fitting: chi2 minimisation

x^2 minimisation

- measurements (x_i, z_i)
- you want to fit a straight line $x = a + bz$

- best a’, b’ by minimising:

$$x^2 = (x_1 - a - bz_1)^2/\sigma_1^2 + \dots + (x_n - a - bz_n)^2/\sigma_n^2$$

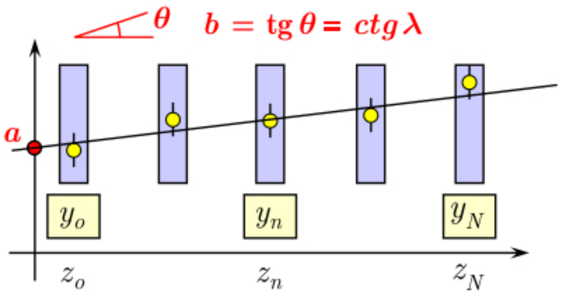
- the model can be generalised

More info in Wiki / Fitting: chi2 minimisation

x^2 minimisation

Straight Line Fit

- This is a well known problem
  - A reference frame
  - $N+1$ measuring detetectors at $z_0, \ldots, z_n, \ldots, z_N$
  - A particle crossing the detectors
  - $N+1$ coordinate measurements $y_0, \ldots, y_n, \ldots, y_N$
  - Each measurement affected by uncorrelated errors $\sigma_0, \ldots, \sigma_n, \ldots, \sigma_N$
- Find the best line $y = a + b\, z$ that fit the track
- The solution is found by minimizing the χ^2



$$\chi^2 = \sum_{n=0}^{N} \frac{(y_n - a - bz_n)^2}{\sigma_n^2}$$

$$a = (S_y S_{zz} - S_z S_{zy})/D$$

$$b = (S_1 S_{zy} - S_z S_y)/D$$

$$S_1 = \sum_{n=0}^{N} \frac{1}{\sigma_n^2} \qquad S_y = \sum_{n=0}^{N} \frac{y_n}{\sigma_n^2}$$

$$S_z = \sum_{n=0}^{N} \frac{z_n}{\sigma_n^2} \qquad S_{yz} = \sum_{n=0}^{N} \frac{y_n z_n}{\sigma_n^2}$$

$$S_{zz} = \sum_{n=0}^{N} \frac{z_n^2}{\sigma_n^2} \qquad D = S_1 S_{zz} - S_z S_z$$

- The covariance matrix (at $z = 0$) is

$$\begin{pmatrix} \sigma_a^2 & c_{ab} \\ c_{ab} & \sigma_b^2 \end{pmatrix} = \frac{1}{D} \begin{pmatrix} S_{zz} & -S_z \\ -S_z & S_1 \end{pmatrix}$$

Depends only on σ, z_n and N

An Introduction to Charged Particles Tracking - Francesco Ragusa 13

χ^2 minimisation

Straight Line Fit: Matrix Formalism

- It is useful to restate the problem using a matrix formalism [4:Avery 1991]
- This is useful because:
  - It is more compact
  - It is easily extensible to other linear problems
  - It is more useful to formulate an iterative procedure
- With the same assumption as before
  - The linear model is given by $\mathbf{f} = \mathbf{A}\mathbf{p}$

$$\mathbf{f} = \begin{pmatrix} f_0 \\ \dots \\ f_N \end{pmatrix} = \begin{pmatrix} a + bz_0 \\ \\ a + bz_N \end{pmatrix} = \begin{pmatrix} 1 & z_0 \\ 1 & \dots \\ 1 & z_N \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} = \mathbf{A}\mathbf{p}$$

  - Measurements and errors are

$$\mathbf{Y} = \begin{pmatrix} y_0 \\ \dots \\ y_N \end{pmatrix} \qquad (\mathbf{V})_{ij} = \left\langle (y_i - \langle y_i \rangle)(y_j - \langle y_j \rangle) \right\rangle$$

$$(\mathbf{V})_{ij} = \sigma_i^2 \delta_{ij} \quad \boxed{\text{if uncorrelated}}$$

- The χ^2 can be written as

$$\chi^2 = (\mathbf{Y} - \mathbf{A}\mathbf{p})^T \mathbf{W} (\mathbf{Y} - \mathbf{A}\mathbf{p}) \qquad \mathbf{W} = \mathbf{V}^{-1}$$

- The minimum χ^2 is obtained by

$$\boxed{\tilde{\mathbf{p}} = \left(\mathbf{A}^T\mathbf{W}\mathbf{A}\right)^{-1} \mathbf{A}^T\mathbf{W}\mathbf{Y}}$$

- The covariance matrix of the parameters is obtained from covariance matrix V of the measurements

$$\boxed{\mathbf{V}_\mathbf{P} = \left(\mathbf{A}^T\mathbf{V}^{-1}\mathbf{A}\right)^{-1}}$$

- Please notice (N+1 measurements, M parameters)
  - Dimensions $\mathbf{A} = (N+1) \times M$
  - Dimensions $\mathbf{V} = (N+1) \times (N+1)$
  - Dimensions $\mathbf{A}^T\mathbf{W}\mathbf{A} = M \times M$
  - Dimensions $\mathbf{A}^T\mathbf{W} = M \times (N+1)$
  - Dimensions $\mathbf{V}_\mathbf{p} = M \times M$

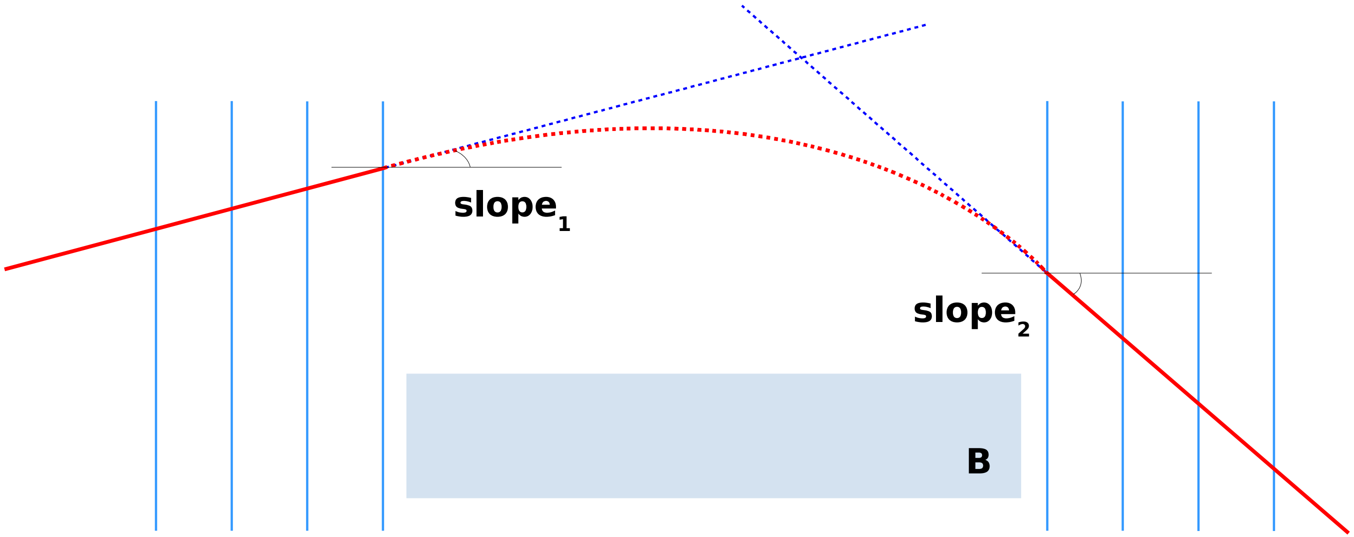
An Introduction to Charged Particles Tracking – Francesco Ragusa 14

Momentum measurement

Δslope method

- uniform, constant B

- tracklets after and before bending region



Momentum measurement

Momentum estimate

The momentum of a T seed can be estimated assuming that the particle originated from the interaction point. This method, known as the p-kick method, is based on the idea that the effect of the field can be described by an instant kick of the momentum vector in the centre of the magnet. In general, the actual momentum kick, $\Delta\vec{p}$, depends on the integrated magnetic field along the particle's trajectory:

$$\Delta\vec{p} = q\int \mathrm{d}\vec{l} \times \vec{B} \quad . \tag{6.48}$$

The main component, Δp_x, provides the highest precision on the momentum. In terms of the track parameters this relation becomes:

$$\Delta p_x = p_{x,f} - p_{x,i} = p\left(\frac{t_{x,f}}{\sqrt{1+t_{x,f}^2+t_{y,f}^2}} - \frac{t_{x,i}}{\sqrt{1+t_{x,i}^2+t_{y,i}^2}}\right) = q\int \left|\mathrm{d}\vec{l} \times \vec{B}\right|_x \quad , \tag{6.49}$$

CERN-THESIS-2005-040

χ^2 minimisation - Tasks

1) implement a 1D chi2 fit for straight lines
in the TFRChiSquaredFit class

- do not take in account multiple scattering in the covariance

Input: clusters of a reconstructed track
Output: estimation of a, b parameters

2) fit the complete track with two straight tracklets

- put the detector layers out of B

3) measure the momentum with delta slope method

A few suggestions

```
double values[9] = {0., 1., 2., 3., 4., 5., 6., 7., 8.};

//fill the matrix
TMatrixD m1(3, 3, values);

//fill a single element
m1(2, 1) = 5.;

//print the matrix, for debug
m1.Print();

//invert the matrix
TMatrixD m_inv(TMatrixD::kInverted, m1);

//transpose the matrix
TMatrixD mt(TMatrixD::kTransposed, m1);

//some crazy operations
TMatrixD m_new = m * m1 + mt;
```

$$\begin{matrix} 0 & 1 & 2 \\ 3 & 4 & 5 \\ 6 & 7 & 8 \end{matrix}$$

$$\begin{array}{cccc} 0 & 1 & 2 & i = 0 \\ 3 & 4 & 5 & i = 1 \\ 6 & \color{red}{5} & 8 & i = 2 \\ j = 0 & j = 1 & j = 2 & \end{array}$$