

Fitting A Line: χ^2 with correlated data

Fitting a Line: χ^2 with covariance

- Previously, we discussed the basic χ^2 statistic and how it can be used to test whether a model fits data and, if so, to find confidence limits on the model parameters
 - Although there are some issues with χ^2 it is simple, quick and often defensible
 - One issue with χ^2 as it was presented in the last set of class notes is that it assumes that bins of data are *independent*
 - In other words, if you have a bin x_1 and a bin x_7 the basic χ^2 statistic assumes that these bins are not correlated, and have no knowledge of the patterns of data in the other
 - There are many examples in astronomy where this is not true (consider periodic light curves for exoplanet crossings, for instance)
-

Variance and covariance

- We are typically used to dealing with variances. We write variances as σ^2 , and think of σ as the standard deviation, without really attaching significance to the squared part of the variance term
 - If we have two bins, a bin x_1 and a bin x_7 we calculate and write the variances as $\sigma_1^2 = \sigma_1\sigma_1$ and $\sigma_7^2 = \sigma_7\sigma_7$
 - but the term σ_{17} is also meaningful. It is called the *covariance* and it characterizes how correlated are the data in bin x_1 and bin x_7 (i.e. whether it is reasonable to assume that they are independent)
 - If a matrix is constructed with row $1, 2 \dots n$ corresponding to $\sigma_1, \sigma_2 \dots \sigma_n$ and column $1, 2 \dots n$ corresponding to $\sigma_1, \sigma_2 \dots \sigma_n$ then the diagonal elements are the variances and the off-diagonal elements are the *covariances*
-

The covariance matrix

- If variance (for n samples of normally distributed data) is
 - $\sigma_i^2 = 1/(n-1) \sum_i (x_i - \mu_i)^2$
- Then the *covariance* matrix is
 - $C_{ij} = 1/(n-1) \sum_{i,j} (x_i - \mu_i) (x_j - \mu_j)$
 - where i, j are corresponding samples in different bins
- Which looks like the following matrix (for $n=5$)

$$\begin{array}{ccccc} \sigma_1^2 & \sigma_{12} & \sigma_{13} & \sigma_{14} & \sigma_{15} \\ \sigma_{21} & \sigma_2^2 & \sigma_{23} & \sigma_{24} & \sigma_{25} \\ \sigma_{31} & \sigma_{32} & \sigma_3^2 & \sigma_{34} & \sigma_{35} \\ \sigma_{41} & \sigma_{42} & \sigma_{43} & \sigma_4^2 & \sigma_{45} \\ \sigma_{51} & \sigma_{52} & \sigma_{53} & \sigma_{54} & \sigma_5^2 \end{array}$$

The correlation matrix

- If each term in the covariance matrix is divided by the two corresponding standard deviations, the resulting matrix is 1 along the diagonal elements and runs from -1 to 1 on the off-diagonal elements
 - For instance $\sigma_{11} = \sigma_1^2$ becomes $\sigma_1^2 / \sigma_1 \sigma_1$
 - σ_{12} becomes $\sigma_{12} / \sigma_1 \sigma_2$
 - σ_{41} becomes $\sigma_{41} / \sigma_4 \sigma_1$
 - This matrix, called the *correlation* matrix is a good indicator of whether the bins of data are independent
 - the correlation matrix values are 0 for independent bins, 1 for highly correlated bins and -1 for highly anti-correlated bins
-

χ^2 with covariance

- Just as we formulated the χ^2 statistic using variances (for normally distributed data) we can generalize the χ^2 statistic to incorporate the full covariance matrix:
 - $\chi^2 = \sum_i (O_i - E_i)^2 / \sigma_i^2$... Becomes ...
 - $\chi^2 = \sum_{i,j} (O_i - E_i) C_{ij}^{-1} (O_j - E_j)$
 - Where C_{ij}^{-1} is the *inverse* of the *covariance* matrix
 - matrix manipulation and inversion can be conducted with numpy's *matrix* module (see the links from the syllabus for some simple examples)
 - This version of χ^2 better describes data that are not independent. The hypothesis testing and confidence limit (etc.) tricks are the same as in the previous class notes
-

Python tasks

1. In my week13 directory is a file of (x,y) data called “line.data” Each of the 10 columns is an x bin of $0 < x < 1$, $1 < x < 2$, $2 < x < 3$, etc. Each of the 20 rows is a trial set of y measurements in that x-bin
 - Read in the file and determine the *covariance matrix* of the y measurements in each bin of x using *np.cov*
 - Should the resulting *covariance matrix* be a 10 x 10 matrix, or a 20 x 20 matrix? Why?
 - Use *np.var* to confirm that the diagonal elements of the *covariance matrix* are the *variances* of the data
 2. Determine the values and locations of the most anti-correlated and most correlated columns of data
 - *Ignoring* the perfectly correlated diagonal of the matrix, which data columns are the most correlated?
-

Python tasks

3. The data have been drawn from a straight line of the form $y = mx + b$. Using the full matrix formalism, determine the χ^2 statistic for a grid of m and b
- Find the minimum value of χ^2 and the corresponding values of m and b .
 - Do the best-fit values of m and b differ from the results from the previous lecture? Should they be?
-