

MAN-HEP 93-1



15

**m**anchester  
particle physics  
MAN/HEP/93/1

# Fitting using finite Monte Carlo samples

*Roger Barlow and Christine Beeston*

Department of Physics  
Manchester University  
Manchester M13 9PL  
England

FERMILAB

JUN 0 1993

IBR

## Abstract

Analysis of results from HEP experiments often involves estimation of the composition of a sample of data, based on Monte Carlo simulations of the various sources. Data values (generally of more than one dimension) are binned, and because the numbers of data points in many bins are small, a  $\chi^2$  minimisation is inappropriate, so a maximum likelihood technique using Poisson statistics is often used. This note shows how to incorporate the fact that the Monte Carlo statistics used are finite and thus subject to statistical fluctuations.

## 1. The problem.

A common problem arises in the analysis of experimental data. There is a sample of real data, each member of which consists of a set of values  $\{x_r\}$  – for example, a set of  $Z$  decay events with inclusive leptons, for each of which there is a value of  $\{p^\ell, p_i^\ell, T, E_{vis}\}$ , or a set of measured particle tracks, each with  $\{p, \frac{dE}{dx}, \text{Cos}\theta\}$ . You know that these arise from a number of sources: the lepton events from direct  $b$  decays, cascade  $b$  decays,  $c$  decays, and background, the tracks from  $\pi$ ,  $K$ , and  $p$  hadrons. You wish to determine the proportions  $P_j$  of the different sources in the data.

There is no analytic form available for the distributions of these sources as functions of the  $\{x_r\}$ , only samples of data generated by Monte Carlo simulation. You therefore have to bin the data, dividing the multidimensional space spanned by the  $\{x_r\}$  values into  $n$  bins. This gives a set of numbers  $\{d_1, d_2 \dots d_n\}$ , where  $d_i$  is the number of events in the real data that fall into bin  $i$ . Let  $f_i(P_1, P_2 \dots P_m)$  be the predicted number of events in the bin, given by the strengths  $P_j$  and the numbers of Monte Carlo events  $a_{ji}$  from source  $j$  in bin  $i$ .

$$f_i = N_D \sum_{j=1}^m P_j a_{ji} / N_j \quad (1)$$

where  $N_D$  is the total number in the data sample, and  $N_j$  the total number in the MC sample for source  $j$ .

$$N_D = \sum_{i=1}^n d_i \quad N_j = \sum_{i=1}^n a_{ji} \quad (2)$$

The  $P_j$  are then the actual proportions and should sum to unity. It is convenient to incorporate these normalisation factors into the strength factors, writing  $p_j = N_D P_j / N_j$ , giving the equivalent form

$$f_i = \sum_{j=1}^m p_j a_{ji} \quad (3)$$

One approach is then to estimate the  $p_j$  by adjusting them to minimise

$$\chi^2 = \sum_i \frac{(d_i - f_i)^2}{d_i} \quad (4)$$

This  $\chi^2$  assumes that the distribution for  $d_i$  is Gaussian; it is of course Poisson, but the Gaussian is a good approximation to the Poisson at large numbers.

Unfortunately it often happens that many of the  $d_i$  are small. This is basically a consequence of the multidimensionality of the data. If each data point has (say) 4 elements, and each axis is split into a fairly modest 10 divisions, that gives  $10^4$  bins, and even for an apparently large sample of (say) 50,000 events there will be many bins for which  $d_i$  is so small that the  $\chi^2$  value as given in Equation 4 is inappropriate. Instead one can go back to the original Poisson distribution, and write down the probability for observing a particular  $d_i$  as

$$e^{-f_i} \frac{f_i^{d_i}}{d_i!} \quad (5)$$

and the estimates of the proportions  $p_j$  are found by maximising the total likelihood or, for convenience, its logarithm (remembering  $a^b = e^{b \ln a}$ , and omitting the constant factorials)

$$\ln \mathcal{L} = \sum_{i=1}^n d_i \ln f_i - f_i \quad (6)$$

This accounts correctly for the small numbers of data events in the bins, and is a technique in general use. It is often referred to as a "binned maximum likelihood" fit.

But this does not account for the fact that the Monte Carlo samples used may also be of finite size, leading to statistical fluctuations in the  $a_{ji}$ . In Equation 1 it can be seen that these are damped by a factor  $N_D/N_j$ , but we cannot hope that this is small. There is a general rule of thumb that the MC samples should be ten times larger than the data sample, so any effects of finite MC data size are relatively small. Unfortunately many MC programs are slow, and the production of enormous samples (and their storage and treatment) presents severe practical problems so that this rule cannot be followed.

So: disagreements between a particular  $d_i$  and  $f_i$  arise from incorrect  $p_j$ , from fluctuations in  $d_i$ , and from fluctuations in the  $a_{ji}$ . Binned maximum likelihood reckons with the first two sources, but not the third. In the  $\chi^2$  formalism of Equation 4 this can be dealt with by adjusting the error used in the denominator

$$\chi^2 = \sum_i \frac{(d_i - f_i)^2}{d_i + N_D^2 \sum_j a_{ji}/N_j^2} \quad (7)$$

but this still suffers from the incorrect Gaussian approximation. The problem is to find the equivalent treatment for the binned maximum likelihood method.

## 2. Methodology

The correct way to view the problem is as follows. For each source, in each bin, there is some (unknown) expected number of events  $A_{ji}$ . The prediction for the number of data events in a bin is not Equation 3 but

$$f_i = \sum_{j=1}^m p_j A_{ji} \quad (8)$$

From each  $A_{ji}$  the corresponding  $a_{ji}$  is generated by a distribution which is in fact binomial, but can be taken as Poisson if  $A_{ji} \ll N_j$  (which is indeed the case, as our problem is just that a large number of total events gives a small number in each bin.)

The total likelihood which is to be maximised is now the combined probability of the observed  $\{d_i\}$  and the observed  $\{a_{ji}\}$  and we want to maximise

$$\ln \mathcal{L} = \sum_{i=1}^n d_i \ln f_i - f_i + \sum_{i=1}^n \sum_{j=1}^m a_{ji} \ln A_{ji} - A_{ji} \quad (9)$$

The estimates for the  $p_j$  (which we want to know) and the  $A_{ji}$  (in which we're not really interested) are found by maximising this likelihood. This is the correct methodology to incorporate the MC statistics: unfortunately it consists of a maximisation problem in  $m \times (n + 1)$  unknowns. However, the problem can be made much more amenable.

### 3. The Solution

To find the maximum we differentiate Equation 9 (including Equation 8 for  $f_i$ ) and set the derivatives to zero. This gives two sets of equations, those for the differentials with respect to  $p_j$

$$\sum_{i=1}^n \frac{d_i A_{ji}}{f_i} - A_{ji} = 0 \quad \forall j \quad (10)$$

and those for the differentials with respect to  $A_{ji}$

$$\frac{d_i p_j}{f_i} - p_j + \frac{a_{ji}}{A_{ji}} - 1 = 0 \quad \forall i, j \quad (11)$$

These  $m \times (n + 1)$  simultaneous equations are nonlinear and coupled (remembering that the  $f_i$  that appear in them are functions of the  $p_j$  and the  $A_{ji}$ ). However they can be remarkably simplified. Equations 11 can be rewritten

$$1 - \frac{d_i}{f_i} = \frac{1}{p_j} \left( \frac{a_{ji}}{A_{ji}} - 1 \right) \quad \forall i, j \quad (12)$$

The left hand side depends on  $i$  only, so write it as  $t_i$ .

$$t_i = 1 - \frac{d_i}{f_i} \quad (13)$$

The right hand side then becomes

$$A_{ji} = \frac{a_{ji}}{1 + p_j t_i} \quad (14)$$

which is a great simplification: for a given set of  $p_j$ , the  $n \times m$  unknown quantities  $A_{ji}$  are given by the  $n$  unknown quantities  $t_i$ .

The  $t_i$  are given by Equation 13. If  $d_i$  is zero then  $t_i$  is 1: if not then

$$\frac{d_i}{1 - t_i} = f_i = \sum_j p_j A_{ji} = \sum_j \frac{p_j a_{ji}}{1 + p_j t_i} \quad (15)$$

If these  $n$  equations are satisfied, with Equation 14 used to define the  $A_{ji}$ , then all the  $m \times n$  Equations 11 are satisfied.

Thus the solution to a binned maximum likelihood fit with finite Monte Carlo statistics is to find the  $m$  variables  $p_j$  by solving the  $m$  Equations 10 iteratively (or, equivalently, finding the maximum of the likelihood as given by Equation 9 by numerical means). These are the same equations that would be solved in the naïve approach, (Equation 6) except that the  $A_{ji}$  occur in place of the  $a_{ji}$ . At every stage in this solution procedure, the appropriate  $A_{ji}$  are found by solving the  $n$  Equations 15.

Although there are  $n$  of these equations, to be solved numerically, this does not present a problem. They are not coupled: each is separate, i.e., the solution required for each bin

depends only on the event numbers in that bin. (This suggests that the problem could usefully be vectorised). Each equation clearly has one and only one solution in the 'allowed' region for  $t_i$ , i.e. the region where the  $A_{ji}$  are all positive, which lies between  $t = -1/p_{max}$  and  $t = 1$  ( $p_{max}$  being the largest of the  $p_j$ ).  $t = 0$  is a suitable place to start, and Newton's method readily gives a solution. Although this has to be done at every step of solving Equations 10, such steps usually consist of a set of small changes in the  $p_j$ , so only one step of Newton's method is generally sufficient for finding the new correct solution of Equations 15.

#### 4. Other points concerning the solution

Some nice points emerge from the algebra. The Equations 10 can be written more simply as

$$\sum_{i=1}^n t_i A_{ji} = 0 \quad \forall j \quad (16)$$

Also one can replace  $\frac{d_i}{f_i}$  by  $1 + (A_{ji} - a_{ji})/p_j A_{ji}$ , from Equation 12, and the equations then reduce to

$$\sum_i A_{ji} = \sum_i a_{ji} \quad \forall j \quad (17)$$

These are telling us that the estimates of the  $A_{ji}$  for some source will change the shape of the distribution from that of the MC data  $a_{ji}$ , but will not change the overall total number.

Equation 11 can be multiplied by  $A_{ji}$  and summed over  $j$  to give

$$\sum_j d_i - p_j A_{ji} + a_{ji} - A_{ji} = 0$$

summing over  $i$ , and using Equation 17, gives

$$\sum_i d_i = \sum_i \sum_j p_j a_{ji}$$

$$N_D = \sum_j p_j N_j \quad (18)$$

which nicely returns the normalisation, and makes clear the significance of the  $p_j$ . It is interesting that such an automatic normalisation does not occur in the  $\chi^2$  minimisation technique of Equation 4. If the different  $p_j$  are allowed to float independently they return a set of values for which the fitted number of events is generally less than the actual total number, as downward fluctuations have a smaller assigned error and are given higher weight.

## 5. Bins with no events from an MC source

*“Nothing will come of nothing. Speak again.”*  
— William Shakespeare: King Lear

Special considerations apply in a bin if the numbers of events  $a_{ki}$  generated by one or more of the source Monte Carlos are zero. (Note: for clarity, the index  $k$  will be used for sources for which this is the case.  $j$  is used for general sources, and  $i$  refers to bins.) For any such bin  $i$  and source  $k$ , Equation 11 gives

$$\frac{d_i}{f_i} - 1 = \frac{1}{p_k} \quad (19)$$

As this may have to be true for several different  $p_k$ , this obviously leads to inconsistency.

What has happened is seen from the likelihood function of Equation 9, considering the contribution from a particular bin as a function of one of the  $A_{ji}$ , over its allowed range from 0 to  $\infty$ . For large values,  $\ln \mathcal{L}$  tends to  $-\infty$  thanks to the  $-A_{ji}$  term. If  $a_{ji}$  is non-zero then  $\ln \mathcal{L}$  also tends to  $-\infty$  as  $A_{ji} \rightarrow 0$ , thanks to the  $a_{ji} \ln A_{ji}$  term. Given such behaviour at the two extremes, there must be a maximum in between, which is a turning point.

But if the  $a_{ki}$  factor is zero, then  $\mathcal{L}$  is finite at  $A_{ki} = 0$ . The slope may be positive or negative: the function may rise to a maximum before falling, or it may be monotonically decreasing. In this latter case the maximum  $\mathcal{L}$  occurs at  $A_{ki} = 0$ . This is not a turning point, and the Equations 19 do not apply.

Notice that if  $A_{ki}$  is zero in such a case, it is still given by Equation 14. This has no significance, but does make the programming simpler as this does not have to be considered as a special case.

If the Monte Carlo numbers in a bin are zero for several  $k$ , then (assuming the strengths  $p_k$  are all different) \* Equation 19 can be true for at most one of the  $k$ . This shows that FOR ANY  $i$  AT MOST ONE OF THE  $A_{ki}$  CORRESPONDING TO ZERO  $a_{ki}$  CAN BE NON-ZERO.

Such a non-zero  $A_{ki}$  will be required when  $d_i$  is larger than can be easily accommodated by the sources for which the MC numbers are non-zero – this qualitative statement will be quantified shortly. Making one  $A_{ki}$  non-zero eases the difference between  $d_i$  and  $f_i$  and increases the overall likelihood, despite incurring a penalty  $\exp(-A_{ki})$  from the ‘result’ that this expected number in fact gave zero observed MC events in the bin. It is clear that if several  $A_{ki}$  are available, the most effective one will be that for which  $p_k$  is greatest, as that provides the greatest easing of any  $d_i - f_i$  deficit for the smallest  $\exp(-A_{ki})$ . Thus the previous result becomes more specific: FOR ANY  $i$ , IF ANY OF THE  $A_{ki}$  CORRESPONDING TO ZERO  $a_{ki}$  ARE NON-ZERO, IT WILL BE THE ONE FOR WHICH  $p_k$  IS LARGEST. All others must be zero.

If such a non-zero  $A_{ki}$  does exist, then Equations 19 and 13 give

$$t_i = -1/p_k \quad (20)$$

---

\* If two or more  $p_k$  are exactly equal then there is an ambiguity as the likelihood depends only on the sum  $A_{k_1i} + A_{k_2i}$ . In the program we arbitrarily resolve this by making the two contributions equal.

For the other  $A_{ji}$  to be physically sensible (i.e. non-negative) requires, from Equation 14  $p_k > p_j$  for all  $j$  with non-zero  $a_{ji}$ . The previous condition thus extends: IF  $a_{ki}$  IS ZERO,  $A_{ki}$  IS NON ZERO ONLY IF  $p_k$  IS THE LARGEST OF ALL THE  $p_j$ .

If this is the case, then Equation 15 becomes

$$\frac{d_i}{1 + 1/p_k} = p_k A_{ki} + \sum_{j \neq k} \frac{p_j a_{ji}}{1 - p_j/p_k} \quad (21)$$

This gives the expression for the value of  $A_{ki}$  at which the likelihood is a maximum

$$A_{ki} = \frac{d_i}{1 + p_k} - \sum_{j \neq k} \frac{p_j a_{ji}}{p_k - p_j} \quad (22)$$

This expression may turn out to be negative – in which case it is meaningless, and the maximum is again given by  $A_{ki} = 0$ . (This is always the case if  $d_i = 0$ ).

In summary, special action is needed only if there is a zero in the number of entries for the MC source with the largest strength, and then only if Equation 22 is positive. In such a case,  $t_i$  is given by Equation 20 instead of 15. Equation 22 gives  $A_{ki}$ , and the other  $A_{ji}$  are given by Equation 14 as usual.

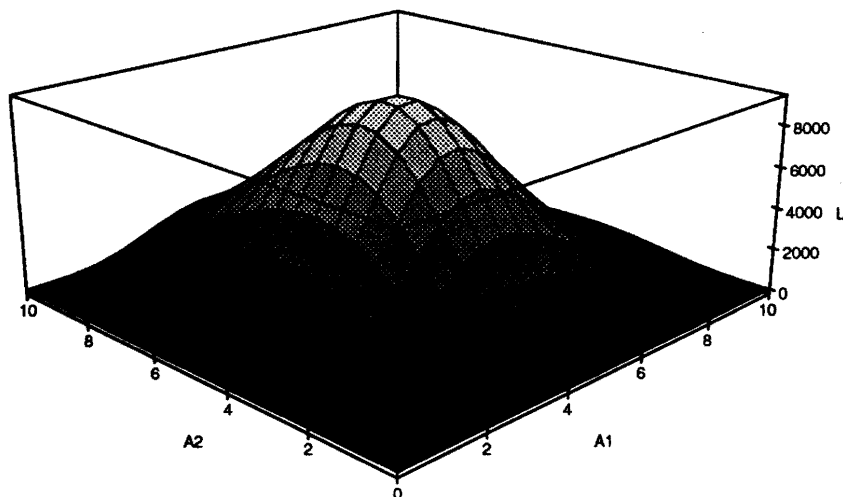


Figure 1: Likelihood as a function of  $A_1$  and  $A_2$  for a bin with 5 data events and 5 from each MC source

This is illustrated in Figures 1-3. Figure 1 shows the likelihood contours for a ‘typical’ bin, drawn using Mathematica [1]. There were 5 data events and 5 events from each of the 2 Monte Carlo sources, for which source 1, the major source, has strength  $p_1 = 0.8$  and source 2, the minor source, has strength  $p_2 = 0.2$ . Likelihood contours are shown as a function of the ideal MC numbers  $A_{1i}$  and  $A_{2i}$ . The most likely values are, of course, 5 events each, and the major source is more tightly constrained as it feels the effect of the data measurement more strongly.

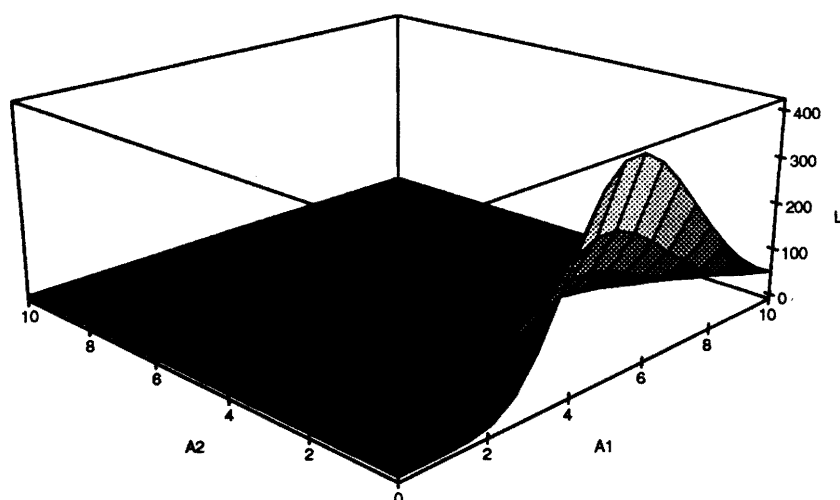


Figure 2: Likelihood as a function of  $A_1$  and  $A_2$  for a bin with 5 data events, 5 from Source 1 (which has  $p=0.8$ ) and 0 from Source 2 ( $p=0.2$ )

In Figure 2 we suppose that the minor source has zero MC events in the bin, the major source still has 5, and so has the data. Then the value of  $A_{2i}$  for which the likelihood is maximum is also zero; the deficit in the prediction is taken up easily by an upward increase in the number  $A_{1i}$  from the major source.

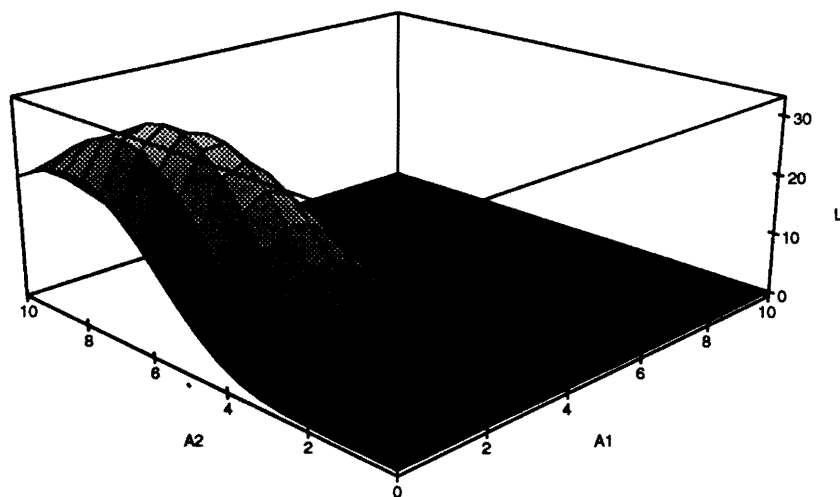


Figure 3: Likelihood as a function of  $A_1$  and  $A_2$  for a bin with 5 data events, 0 from Source 1 and 5 from Source 2

Figure 3 shows what happens if the major source has no MC events, whereas the data and the minor source have 5. Now the strain on the minor source is too great, and the highest likelihood occurs with a non-zero expected value for the major source :  $A_{1i}$  is non-zero even though  $a_{1i}$  is zero. This corresponds to Equation 22 being positive.



## 6. Weighted Events

In some problems it is necessary to apply weights to the Monte Carlo data before comparing it with the real data.

An example occurs when events are detected with some efficiency, which differs from bin to bin, and the form of which is known exactly. Rather than reject MC events on a random basis, it is more effective to include them all, weighted by the appropriate efficiency.

Another such instance arises if MC data has been generated according to one function, but another one is desired. For example, data on  $\{p, \frac{dE}{dx}, \cos\theta\}$  may have been generated using some form of the Bethe-Bloch Equation

$$\frac{dE}{dx} = F_0(p, \theta, m_j)$$

and with hindsight it is realised that some other form  $F_1(p, \theta, m_j)$  is more correct. This can be accommodated by weighting each bin by

$$w_{ji} = F_1/F_0$$

In such a case the predicted number of events in each bin is modified and Equation 8 becomes

$$f_i = \sum_{j=1}^m p_j w_{ji} A_{ji} \quad (23)$$

The likelihood function of Equation 9 is unchanged. The differentials of Equation 10 become

$$\sum_{i=1}^n \left( \frac{d_i}{f_i} - 1 \right) w_{ji} A_{ji} = 0 \quad \forall j \quad (24)$$

and the differentials with respect to the  $A_{ji}$  give the equivalents of Equations 14 and 15.

$$A_{ji} = \frac{a_{ji}}{1 + p_j w_{ji} t_i} \quad (25)$$

$$\frac{d_i}{1 - t_i} = f_i = \sum_j \frac{p_j w_{ji} a_{ji}}{1 + p_j w_{ji} t_i} \quad (26)$$

The solution of these 4 sets of equations proceeds as before. Notice that, as one would expect, if  $w_{ji}$  is the same for all  $i$ , then this merely amounts to a scaling of the Monte Carlo strength  $p_j$ .

An interesting application of this occurs if it is desired to combine two Monte Carlo sources in a known ratio: for example, Monte Carlo sources 1 and 2 could represent two different decay channels of a particular heavy particle: the ratio of the two decay modes is known but the overall number of such decays is not. This can be treated by solving the equations 10 subject to the constraint

$$k p_1 + (1 - k) p_2 = p_{1+2}$$

using the method of Lagrangian multipliers, or equivalently by replacing  $p_1$  and  $p_2$  by the combined strength  $p_{1+2}$  in the equations, applying a constant weight  $k$  to events from source 1 and  $(1 - k)$  to those from source 2.

So far this assumes that the weight is the same for all events from a given source in a given bin: the quantity  $w_{ji}$ . This may not be the case if either (a) the bin size is so large that the weight factor varies significantly within the bin or (b) the weight factor depends not only on the variable(s)  $x$  used in making the comparison but also on some other variable(s) – call it  $z$  – which is not binned and used in the comparison; perhaps it does not exist in the real data. In either case the weights of different events from the same source in the same bin can be different.

In such a case the Equations 23 – 26 still apply, with  $w_{ji}$  equal to the ideal average weight for source  $j$  in bin  $i$ . This may be a known quantity: more likely it has to be estimated using the actual weights attached to the MC data values themselves.

At this point one has to worry whether the discrepancy between the average actual weight and the true average weight should be included in the fitting procedure, estimation and errors. Now in practice this method of weighting only works satisfactorily if the weights do not differ very much. The variance of a sum of weights from Poisson sources is  $\sum_i w_i^2$  [2] and thus the proportional error on the bin contents  $\sqrt{\sum_i w_i^2} / \sum_i w_i$  is greater than the  $1/\sqrt{N}$  obtained from unweighted Poisson statistics, and this effect get worse as the spread of weights,  $\overline{w^2} - \bar{w}^2$ , gets larger. Fluctuations in a small number of events with large weights will swamp the information obtained from low weight events. Thus in any application the spread in weights for a source in a bin should be small, and this means that the resulting uncertainty in its value will also be small.

Some insight can be gained by noting that in the set of Equations 23– 26 the weights  $w_{ji}$  always appear together with the  $p_j$ . (Equation 24 can be multiplied by  $p_j$  to make this explicit). Thus if the weights are all too high by some factor the strengths will be low by exactly the same factor. So the error in the  $p_j$  estimates resulting from uncertainties in the weights is of the same order as that uncertainty, and in any application this should be small.

## 7. Implementation details for the new style fit

### 7.1 Possible Approaches

There are two possible methods of finding the desired values of  $p_j$  for which  $\ln \mathcal{L}$  is a maximum:

1: Find the point at which the differentials vanish, by solving equations 16. This can be done iteratively, using an equation-solving routine such as the NAGLIB routine C05PCF [3]. Given a set of starting values for the  $p_j$ , the user routine called by C05PCF solves Equations 15 for the  $t_i$  by using Newton's method, as outlined above. It then calculates the values of the quantities in equations 16, and the derivatives, including the effects of changes in the  $p_j$  that arise directly and those that arise through the solutions of Equations 15. These are then used to find an improved set of  $p_j$ , and the procedure iterates until a suitable solution is found.

2: Maximise  $\ln \mathcal{L}$  itself with respect to the  $P_j$ , using a package such as MINUIT [4]. For each set of values for  $p_j$ , equations 15 may be solved for the  $t_i$  as before; these give the  $A_{ji}$  and thus  $\ln \mathcal{L}$  may be calculated for this set of  $p_j$ , and a maximum found by standard techniques.

Both these approaches have been used successfully by the authors on test problems. The second is easier to apply, and is thus the one we provide for general use, in the belief it will be more convenient for potential users who have such an application (and also for the reasons given in the next two sections).

### 7.2 Fixing the $p_j$

In many cases one or more of the  $p_j$  are known *a priori*, and they must be assigned fixed values. If the solution is found by maximisation this is straightforward. If it is sought by solving the Equations 16 it is more complicated, as the user function called by C05PCF must be able to take this into account in the calculation of the differentials.

### 7.3 Error estimates

For every fit, an error estimate is almost certain to be required. One possible way would be to invert the second derivative matrix of  $\ln \mathcal{L}$ , to give the error matrix. Unfortunately this can be very complex and time consuming as  $\ln \mathcal{L}$  is a function of  $m \times (n + 1)$  variables - that is, the  $p_j$  and all the  $A_{ji}$ . A simpler method is to vary the  $p_j$  around the region of the solution and map out the variation of  $\ln \mathcal{L}$ , the errors being given by the variation in the  $p_j$  needed to decrease  $\ln \mathcal{L}$  by an amount depending on the number of free  $p_j$  and the required confidence level.

Such an error analysis combines naturally with a solution found by maximising the likelihood (type 2 above).

## 8. An Example

The following example illustrates the differences between results obtained using the 4 different methods:

- 1) Simple  $\chi^2$  minimisation (Equation 4).
- 2)  $\chi^2$  minimisation adjusted to include the effects of finite MC statistics (Equation 7).
- 3) The simple binned likelihood fit (Equation 6).
- 4) The new binned likelihood fit adjusted to include the effects of finite MC statistics (Equation 9).

The set of values considered was two dimensional, thus each measurement gave two values,  $x_1$  and  $x_2$ . The space considered was defined by the square  $0 \leq x_1 \leq 1; 0 \leq x_2 \leq 1$ .

Two Monte Carlo sources were considered. Events from the first source were produced with a distribution given by rising with  $x_1$  and  $x_2$ .

$$F_1(x_1, x_2) \propto \frac{x_1 + x_2}{2}$$

The distribution of the second source was one rising with  $x_1$  and falling with  $x_2$ .

$$F_2(x_1, x_2) \propto \frac{1 + x_1 - x_2}{2}$$

For each fit, 1000 'data' events were generated in the ratio 1:2, i.e.  $p_1 = 0.3333, p_2 = 0.6667$ . The data were binned in the 2 dimensions. The results of the 4 different fitting procedures for varying numbers of MC events per source, and varying numbers of bins per dimension were found. 500 attempts (using 500 sets of data and Monte Carlo data) were made for each of the cases summarised in the table.

For the  $\chi^2$  fits, the sum of the fractions was constrained to be 1. (If the two fractions are both free the results sum to less than 1, as discussed earlier). For the maximum likelihood fits this constraint was not applied, but it emerges automatically in the results.

	Number of MC events per source	Number of bins per dimension	Total number of bins	Average entries per MC source per bin
A	10000	5	25	400
B	1000	5	25	40
C	1000	10	100	10

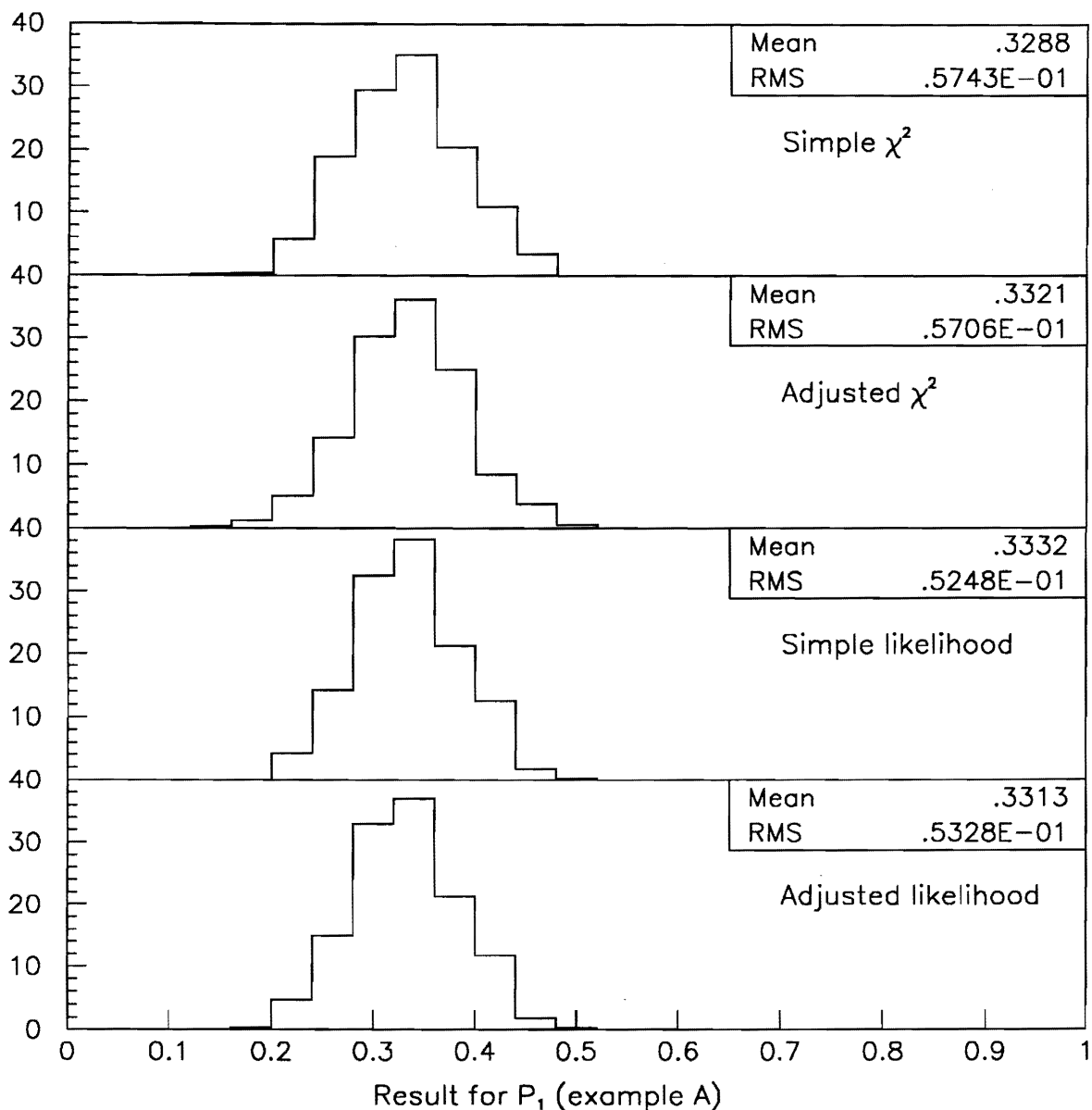


Figure 4: Results for example A: 1000 data events and 10000 events for each type of Monte Carlo in 25 bins

Figure 4 shows the  $p_1$  results for example A. All four methods give much the same performance, as would be expected since the number of MC events in each bin is satisfactorily large, as is the number of data events. No method shows a bias: the error on the mean for each of the 4 distributions is given by the RMS deviation divided by  $\sqrt{500}$ , and is thus of order 0.002 - 0.003. The mean results agree with the ideal 0.333 within such an error in all 4 cases.

The effect of the Monte Carlo statistics on the (RMS) width of the distribution can be seen to be negligible.

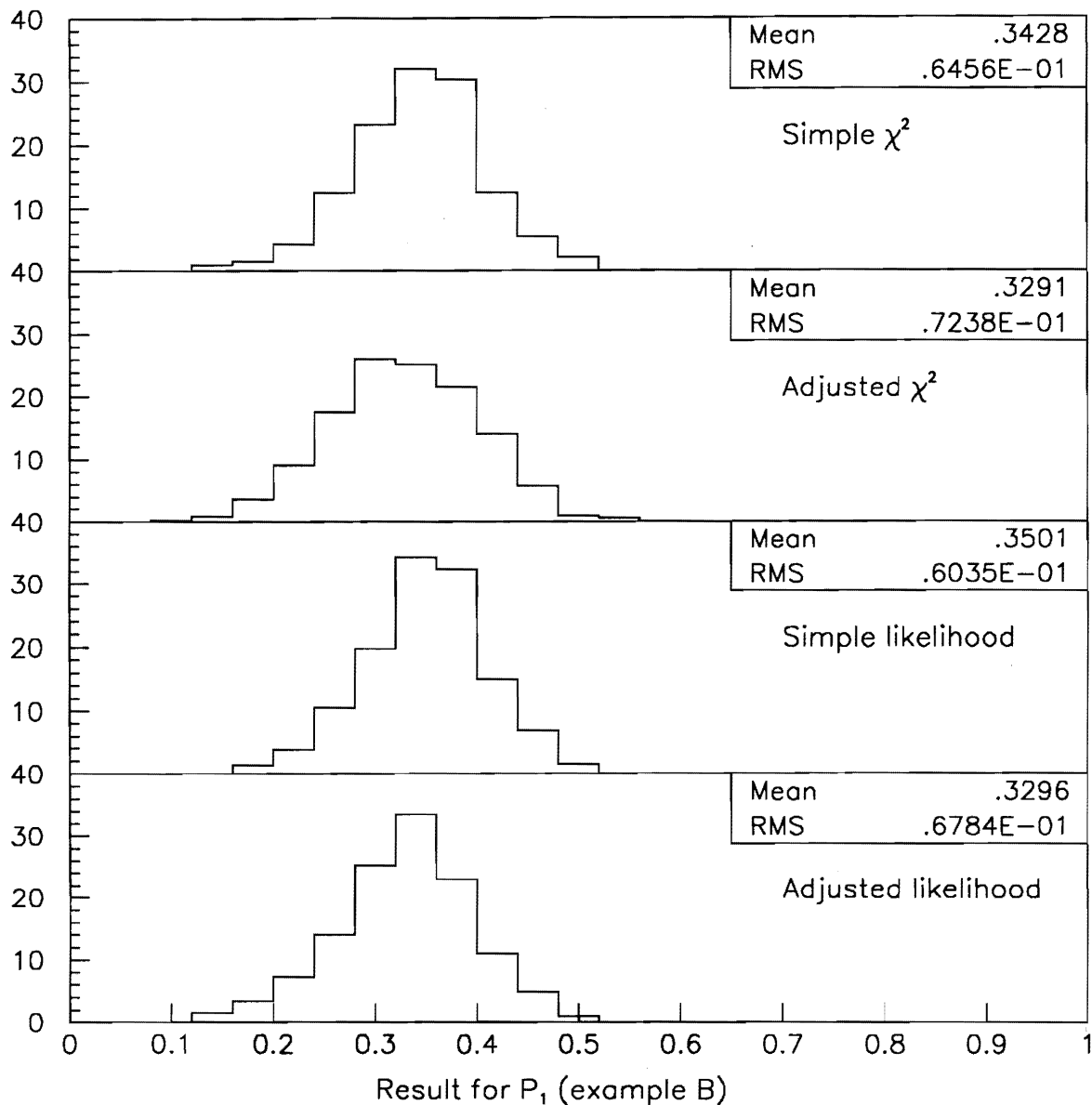


Figure 5: Results for example B: 1000 data events and 1000 events for each type of Monte Carlo in 25 bins

Figure 5 shows the results from example B, where the Monte Carlo statistics are reduced to the level of the data statistics.

The expected error on the position of the mean is of order  $0.06/\sqrt{500} \approx 0.003$ . It can be seen that the 4 mean values (from 0.329 to 0.350) are thus significantly different, and that the simple  $\chi^2$  and simple likelihood are meaningfully different from the ideal value of 0.3333, and are thus biased.

The width of the result distributions is larger for the adjusted  $\chi^2$  and  $\ln \mathcal{L}$  methods than for the simple methods. This is faithfully represented by the error estimates given by the fits. So it is important to include the effect of MC statistics in such cases in order to get a valid error estimate. (This will be discussed further in the next example when the trend is clearer).

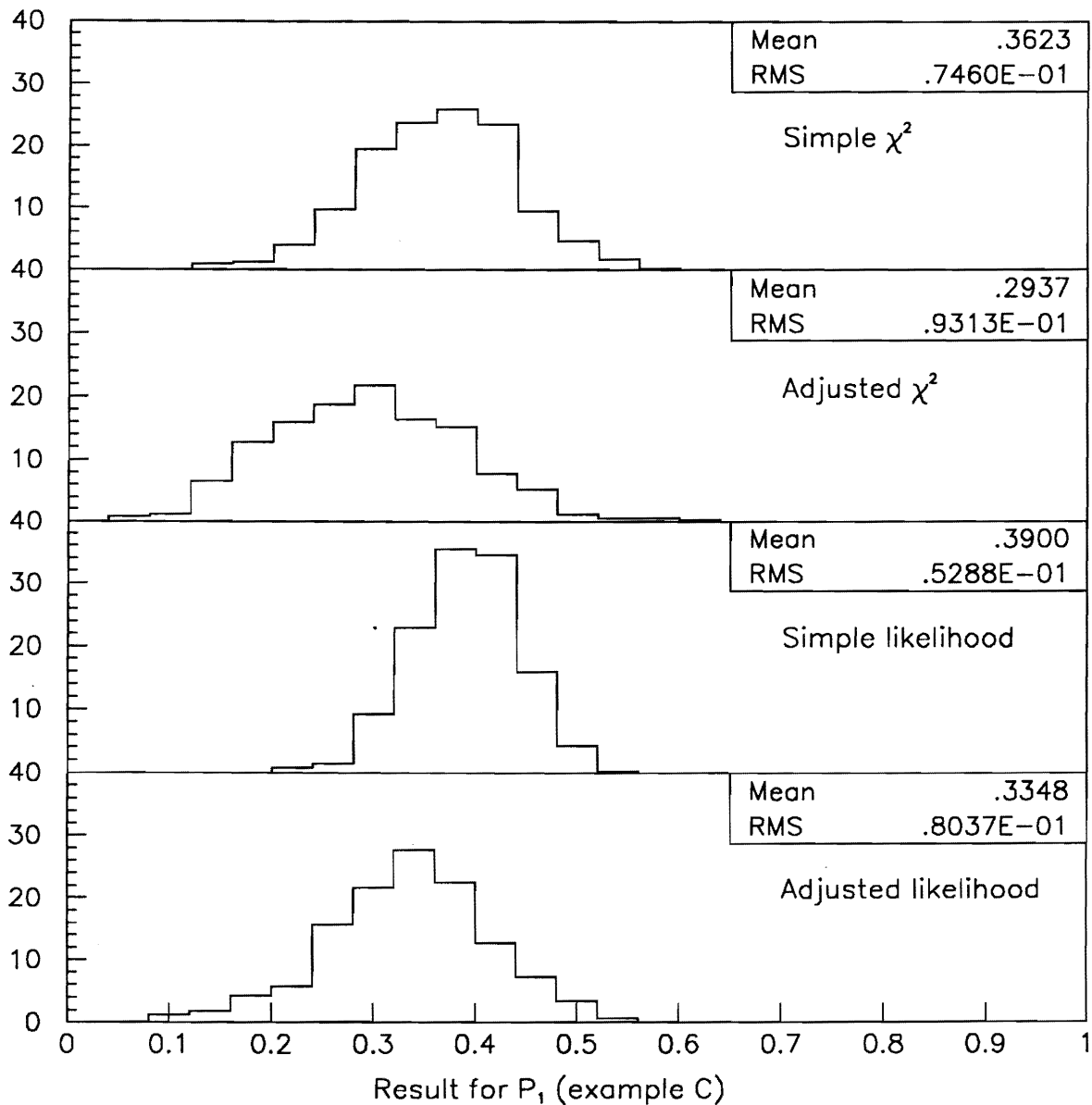


Figure 6: Results for example C: 1000 data events and 1000 events for each type of Monte Carlo in 100 bins

Finally Figure 6 shows the results of each method when the number of MC and data events per bin is reduced to 10 (on average). The difference in the results is now plain: all methods show a bias in the result except for the new  $\ln \mathcal{L}$  method.

The widths of the distributions (and hence the errors on the results) are also different. Although the results of the simple likelihood fit have a narrower distribution than the new adjusted likelihood, as is evidenced by the smaller rms deviation, this does not mean the method has a smaller error! The deviation about the observed mean 0.39 is smaller, but the deviation about the true mean of 0.33 is not! It might be thought that one could overcome this by using the results of the simple likelihood and then compensating for the bias of +0.06 by subtracting it. Unfortunately this bias figure can only be found by Monte