Published in final edited form as:
Powder Diffr. 2018 ; 33: . doi:10.1017/S0885715618000726.

Poisson errors and adaptive rebinning in X-ray Powder Diffraction Data

Marcus H. Mendenhall
National Institute of Standards and Technology (NIST), 100 Bureau Drive, Gaithersburg, MD 20899 USA

Abstract

This work provides a short summary of techniques for formally-correct handling of statistical uncertainties in Poisson-statistics dominated data, with emphasis on X-ray powder diffraction patterns. Correct assignment of uncertainties for low counts is documented. Further, we describe a technique for adaptively rebinning such data sets to provide more uniform statistics across a pattern with a wide range of count rates, from a few (or no) counts in a background bin to on-peak regions with many counts. This permits better plotting of data and analysis of a smaller number of points in a fitting package, without significant degradation of the information content of the data set. Examples of the effect of this on a diffraction data set are given.

Keywords

powder diffraction; Poisson statistics; adaptive rebinning

1 Introduction

The x-ray diffraction community collects a great deal of data which consist of patterns with very sharp, intense peaks scattered over a background with a weak signal. The individual bins in these patterns consist of photon counts, and their statistical variation is well described by Poisson (counting) statistics. Such data sets may be collected either as a single, uniform scan of an instrument over the full angular range of interest, or as a set of shorter scans which cover the regions around the sharp peaks at high resolution, so that most of the data acquisition time is spent on ‘interesting’ regions. Hybrid scans which cover the peaks at high resolution, and the whole pattern at lower resolution, are particularly effective at reducing counting time while assuring precise peak information and a good understanding of the background. This paper presents a statistically rigorous set of procedures for manipulating such data sets, especially in the case in which the data involve very low counting rates, where the difference between exact Poisson statistics and the commonly-used Gaussian approximation is significant.

marcus.mendenhall@nist.gov. Official contribution of NIST; not subject to copyright in the United States.

2 Statistical background

For a Poisson-distributed variable which describes the counting of uncorrelated events at a fixed rate, the following well-known relations hold (Wikipedia, 2018) (where a quantity x in angle brackets $\langle x \rangle$ represents the mean value of the quantity):

$$P(\mu, N) = \frac{\mu^N}{N!} e^{-\mu} \quad (1)$$

$$\mathrm{Var(N,}\mu) \equiv \langle (N - \mu)^2 \rangle = \left\langle N^2 \right\rangle - \mu^2 = \mu, \quad (2)$$

where $P(\mu, N)$ is the Probability Distribution Function (PDF) of observing N events in some interval if the perfectly-known mean rate of events for this interval is μ, and Var(N, μ) is the expected variance of the number of events around this mean, which is also equal to the mean. The critical statement here is that μ is somehow known correctly, *a priori*. However, in a real measurement, all that is available is an observation of the number of counts N itself. The first issue to address is the determination of the relationship between an observed number of counts and an expected mean value μ. This has been addressed by Bayesian methods in papers such as Kirkpatrick and Young (2009), which conclude that, for an observation of N events in an interval, the most probable assignment of μ is $\mu = N + 1$, and that the variance from eq. 2 is also $N + 1$.

Another (more transparent) approach, which yields exactly the same result, is to directly consider the possibilities presented by an observation of N events. To accomplish this, we need tonote some properties of $P(\mu, N)$:

$$\Sigma_{i=0}^{\infty} P(\mu, i) = 1 \quad (3)$$

$$\int_0^{\infty} P(\mu, N)\, d\mu = 1 \quad (4)$$

$$\int_0^{\infty} \mu^m P(\mu, N)\, d\mu = \frac{(m + N)!}{N!} \quad (5)$$
$$= (N + 1) \times \ldots \times (N + m).$$

Equations 3 and 4 imply $P(\mu, N)$ is both a normalized PDF for the discrete variable N at fixed μ and for the continuous variable μ for a fixed N. Then, for a given number N of counts observed, we can consider these counts to have resulted from, with equal probability, a parent distribution with any possible value of μ. From this, we calculate the expectation value of μ and its variance. The assumption of equal probability is equivalent to a Bayesian approach having no prior information. Then,

$$\begin{aligned} \langle \mu \rangle &= \int_0^\infty \mu P(\mu,\, N)\, d\mu \,=\, N \,+\, 1 \\ \left\langle \mu^2 \right\rangle - \langle \mu \rangle^2 &= \int_0^\infty \mu^2 P(\mu, N)\, d\mu \,-\, (N+1)^2 \\ &= (N+2)(N+1) \,-\, (N+1)^2 \,=\, N+1. \end{aligned} \qquad (6)$$

3 Application to data

Equation 6, then, establishes that the variance of an observation of N counts is $N+1$. As pointed out in Kirkpatrick and Young (2009), this is commonly used *ad hoc* to eliminate divide-by-zero conditions in statistical analyses in which a weight of $1/\sigma^2$ is used in a fitting procedure, but it is formally correct to do this. Data from X-ray power diffraction experiments are often stored as ‘xye’ files, in which the first column is the detector angle, the second column is the counting rate, and the third column is the standard uncertainty on that counting rate. The previous section then yields rules for both creating and manipulating *xye* file data. First, in the creation of an *xye* entry, if one has N counts in a dwell time of τ, the columns would be set to

$$\begin{aligned} y &= N/\tau \\ e &= \sqrt{N \,+\, 1}/\tau\,. \end{aligned} \qquad (7)$$

If one is faced with already-created *xye* files, in which the more standard choice of $y = N/\tau$ and $e = \sqrt{N/\tau}$ has been made, it is possible to approximately convert them to this standard. The problem lies in the bins with zero counts, which therefore are recorded with zero error. This makes it impossible to directly compute the dwell time for the empty bins. Assuming the data were taken with constant or smoothly varying count times, though, one can use the dwell time τ' from a nearby non-empty bin with rate y' and error e', by computing $\tau' = y'/e'^2$ and then replacing the e value of the empty bin with $e_{\text{new}} = 1/\tau' = e'^2/y'$. The e column of non-zero bins will be replaced with

$$e_{\text{new}} = \frac{\sqrt{N+1}}{\tau} = \frac{\sqrt{\left(\frac{y}{e_{\text{orig}}}\right)^2 + 1}}{y/e_{\text{orig}}^2} = \sqrt{e_{\text{orig}}^2 + (e_{orig}^2/y)^2}. \qquad (8)$$

Note that the alternative of just dropping empty bins is statistically wrong; the empty bins have finite weight and contribute to any analysis.

After this, the more interesting question is how to combine bins in such data sets. The data are always treated as heteroskedastic, but the distributions are not really Gaussian. The usual method of computing the minimum-uncertainty weighted mean of two Gaussian-distributed quantities $y_1 \pm \sigma_1$ and $y_2 \pm \sigma_2$:

$$y = \frac{y_1/\sigma_1^2 + y_2/\sigma_2^2}{1/\sigma_1^2 + 1/\sigma_2^2}, \quad (9)$$

isn't really right, since these are Poisson variates, and not Gaussian. The correct solution to combine a set of M measurements recorded as y_j and e_j ($j = 1 \ldots M$) is to reconstruct the N_j and τ_j which are represented by the recorded values, and compute the total N and τ. This preserves the Poisson nature of the statistical distribution (since all that has been done is to regroup counts). One can solve equation 7 for each N and τ using an intermediate quantity α:

$$\begin{aligned} \alpha_j &\equiv \left(\frac{y_j}{e_j}\right)^2 = \frac{N_j^2}{N_j + 1} \quad (10) \\ N_j &= \frac{\alpha_j + \sqrt{\alpha_j^2 + 4\alpha_j}}{2} \\ \tau_j &= \frac{\sqrt{1 + N_j}}{e_j}. \end{aligned}$$

and the statistics of the M combined measurements are:

$$\begin{aligned} y &= \frac{\Sigma_{j=1}^{M} N_j}{\Sigma_{j=1}^{M} \tau_j} \\ e &= \frac{\sqrt{1 + \Sigma_{j=1}^{M} N_j}}{\Sigma_{j=1}^{M} \tau_j}. \end{aligned} \quad (11)$$

4 Adaptive Rebinning

Often, it is useful to take a data set which has regions with many bins with only a few counts, and accumulate the many low-count bins into a smaller number of bins with higher numbers of counts. This is not normally recommended for least-squares fitting procedures, assuming the weights are computed carefully, since any aggregation of data results in some loss of information. However, most of the aggregation is in regions with few counts, where there isn't much information in the first place, and it may result in a large speed increase due to the reduction in the number of bins to analyze. For the purposes of plotting data sets, and for presentation of results for distribution, rebinning can be very useful. If such rebinning is carried out in such a way as to assure a minimal statistical significance for each accumulated bin, rather than by just collecting fixed-width groups of bins together (which at least results in uncorrelated bins, but results in broadening of peaks in regions with plenty of counts), or (worse) by computing a running average (which produces correlated bins, most likely resulting in incorrect error estimates from fitting software), the resulting data set can preserve a great deal of information about the widths and positions of strong peaks, while creating points in the weak regions which have reduced y uncertainties at the expense of increased x uncertainties.

We present an algorithm here that rebins data from a set of *xye* values while reasonably preserving the shape of strong peaks, and strictly preserving statistics of the counts within bins and the first moments of peaks. It transforms an *xye* set into a new *xye* set. This set can be created from a single *xye* pattern, or from multiple patterns which have just been concatenated into a single array, and then sorted on x. There is no requirement on the uniqueness of x values, as long as they are non-decreasing. The notation below implicitly uses the conversions in eq. 10. Although we represent the calculation of α_j and N_j as pointwise operations, if one is working in a computer language which permits direct array operations, these can be computed for the entire xye array in advance of the iterative part of the algorithm. We assume a computer language which includes lists of objects in one form or another (python lists, c++ std::vector, etc.), and in which the first element of a list is indexed as element 0, and the the i^{th} element of list z is written as z_i. We use quantities in brackets {a, b, c, … } to represent a list of items. The pseudocode is written without the use of structured programming constructs, even though a ‘while’ loop is likely the real implementation of the steps from 3 through 13 in a modern computer language. The ’=’ sign is a comparison operator; the ’⟵’ is assignment.

The input to the algorithm is M points of *xye* data, referenced as x_j, y_j, e_j, α_j, and N_j (from eq. 10), and a minimum relative error ε for a bin to be considered sufficient. The tolerance in step 8 is just a small fraction of the typical bin spacing, so that combined data sets which may have very nearly equal x values don’t get similar bins split across outgoing channels. The algorithm runs as follows:

```
1.  create lists $sn \leftarrow \{0\}$, $s\tau \leftarrow \{0\}$, $sxn \leftarrow \{0\}$
2.  create data counter $j \leftarrow 0$ and current bin counter $k \leftarrow 0$
3.  if $j = M$: go to step 13
4.  $sn_k \leftarrow sn_k + N_j$
5.  $s\tau_k \leftarrow s\tau_k + \tau_j$
6.  $sxn_k \leftarrow sxn_k + x_j N_j$
7.  $j \leftarrow j + 1$
8.  if $j < M$ and $x_j - x_{j-1} <$ tolerance: go to step 3 (make sure nearly repeated $x$
    values all get summed into the same bin)
9.  if $sn_k + 1 < 1/\varepsilon^2$: go to step 3 to accumulate more data
10. append a 0 to lists $sn$, $s\tau$, and $sxn$ to start a new bin
```

11. $k \leftarrow k + 1$
12. go to step 3
13. eliminate any bins in all lists corresponding to bins for which $sn_j = 0$. This can really only happen on the final bin, if there are empty bins at the end of the incoming data sets.
14. compute $x' \leftarrow sxn/sn$ (operations on lists are carried out element-by-element).
15. compute $y' \leftarrow sn/s\tau$
16. compute $e' \leftarrow \sqrt{sn + 1}/s\tau$

These final lists are the new *xye* data set. It is worth noting, though, that this has thrown away one piece of statistical information. The new bins are unevenly spaced, and have an uncertainty on their x value, too, since they are aggregated from multiple original bins. A more complete version of this algorithm would generate 4 columns of output: x, x error, y, y error, and would include a summation of $x_j^2 Nj$ to allow computation of the second moment of x which would feed into the x error. The incompatibility of this with common pattern fitting algorithms makes it less easy to use, and the benefits seem mostly weak, so in most cases the algorithm in this this section suffices.

5 Sample results

Figure 1 shows the result of this type of operation on data sets collected from the NIST Parallel Beam Diffractometer (PBD) (Mendenhall *et al.*, 2016; Mendenhall *et al.*, 2017) equipped with a focussing mirror. The data consist of a coarse survey scan of diffraction of Cu radiation from a silicon powder (SRM660b, NIST 2010) sample (red ’+’ signs), and a very fine scan over the peak to get details of the peak shape (green circles). The blue crosses are the result of concatenating and sorting these two sets, and rebinning with a $\varepsilon = 2$ % relative tolerance. The following characteristics are evident: 1) on top of the peak, the rebinned channels are in 1:1 correspondence with the raw data, since statistics are sufficient there that each channel satisfies the $\varepsilon = 2$ % requirement; 2) as one moves down the side of the peak, the rebinned points move farther apart, since more channels are being aggregated to achieve the goal; 3) the variance of the blue crosses is much lower than the red (survey) data in the low-counts region, since the bins are highly aggregated. The total number of points in the source data sets, over the whole scan range (20 degrees to 140 degrees) is about 6000, but only 640 remain in the rebinned set, yet very little information has been lost.

The utility of this procedure for preparation of readable graphic representations of data becomes particularly clear when data are being presented on a logarithmic vertical scale. In this case, the noise in low-count areas, especially if there are channels with no counts, results in a nearly unreadable baseline. Figure 2 shows this effect, with data synthesized from those of figure 1 to simulate reduced counting statistics. The red crosses are widely scattered on the log scale, but the blue, rebinned result has a very easily determined level.

Although, in general, data rebinning is harmful to analyses of data such as least-squares fitting, it is worthwhile to quantify the actual effect of such binning on such fits. The complete data set, from $2\theta = 20°$ to $2\theta = 140°$, used as an example in figure 1 has been adaptively rebinned into sets with $\varepsilon = 10$ %, $\varepsilon = 5$ %, $\varepsilon = 2$ % and $\varepsilon = 1$ % tolerances. These sets were then fitted using the Fundamental Parameters Approach (FPA) (Cheary and Coelho, 1992; Cheary *et al.*, 2004; Mendenhall *et al.*, 2015) and a Pawley procedure (Pawley, 1980) using Topas5[1] (Bruker AXS, 2014) software. The fit parameters allowed to vary were the lattice parameter, the Lorentzian crystallite size broadening, and the apparent outgoing Soller slit width to fit the axial divergence, and are displayed in table 1. It is important to note that the same underlying data set is used in all cases, so the the differences between the fits should be much less than the statistical error bars if the rebinning is valid. Only the set reduced to 176 points (less than 3 % of the original size) is beginning to show changes to the fit that are statistically significant; in this set, many of the weaker peaks only have 2 or 3 points across the full width at half maximum. The fit times were the time for 200 iterations of the fitter, they vary significantly from run to run, and should only be taken as general guidance for speed. The difference between the almost-complete ($\varepsilon = 10$ % tolerance) and very sparse ($\varepsilon = 1$ % tolerance) data set is shown in figure 3.

6 Conclusion

A formal recognition of the differences between the errors associated with a Poisson distribution and those of a Gaussian distribution leads to some rules which allow manipulation of counting-statistics data sets in a manner that does not degrade the statistical information in them. In particular, the association of a variance of $N + 1$ with an observation of N counts allows uniform handling of statistics in systems that span the extremely-low pure-Poisson range up to the usual Gaussian limit. This allows simple aggregation of data from multiple sets, as well as adaptive adjustment of the size of counting bins to maintain statistical significance even in regions of very sparse counts. Although, in general, precision analysis of data sets should be carried out on minimally-preprocessed data, we demonstrate that using rebinning, within reason, does not perturb fitting results and can speed up fits due to the reduced number of data points. The data compression that results from adaptive rebinning may be very useful in building rapidly-searchable catalogs of patterns. This probably has its primary utility in patterns in which strong features are sparsely distributed over a largely featureless background, or data which are oversampled relative to the resolution required to describe the narrowest features.

References

Bruker AXS (2014) Topas v5, a component of DIFFRAC.SUITE, 2014 URL https://www.bruker.com/products/x-ray-diffraction-and-elemental-analysis/x-ray-diffraction/xrd-software.html.

Cheary RW and Coelho A (1992) A fundamental parameters approach to X-ray line-profile fitting. J. Appl. Cryst, 25 (2):0 109–121, 1992 10.1107/S0021889891010804.

Cheary RW, Coelho AA, and Cline JP (2004) Fundamental parameters line profile fitting in laboratory diffractometers. J. Res. NIST, 109:0 1–25, 2004 10.6028/jres.109.002.

[1] Certain commercial equipment, instruments, or materials are identified in this paper in order to specify the experimental procedure adequately. Such identification is not intended to imply recommendation or endorsement by the U.S. government, nor is it intended to imply that the materials or equipment identified are necessarily the best available for the purpose.

Kirkpatrick JM and Young BM (2009) Poisson statistical methods for the analysis of low-count gamma spectra. IEEE Transactions on Nuclear Science, 56 (3):0 1278–1282, 6 2009 10.1109/TNS. 2009.2020516.

Mendenhall MH, Mullen K, and Cline JP (2015) An implementation of the Fundamental Parameters Approach for analysis of X-ray powder diffraction line profiles. J. Res. NIST, 120:0 223–251, 10 2015 10.6028/jres.120.014.

Mendenhall MH, Henins A, Windover D, and Cline JP (2016) Characterization of a self-calibrating, high-precision, stacked-stage, vertical dual-axis goniometer. Metrologia, 53 (3):0 933–944, 4 201610.1088/0026-1394/53/3/933.

Mendenhall MH, Henins A, Hudson LT, Szabo CI, Windover D, and Cline JP (2017) High-precision measurement of the x-ray Cu Kα spectrum. Journal of Physics B: Atomic, Molecular and Optical Physics, 50 (11):0 115004, 6 201710.1088/1361-6455/aa6c4a.

NIST (2010) Standard Reference Material 660b online. https://www-s.nist.gov/srmors/view_detail.cfm?srm=660b.

Pawley GS (1980) EDINP, the Edinburgh powder profile refinement program. J. Appl. Cryst, 13 (6):0 630–633, 12 1980 10.1107/S0021889880012964.

Wikipedia (2018) Poisson Distribution *online*. https://en.wikipedia.org/wiki/Poisson_distribution.

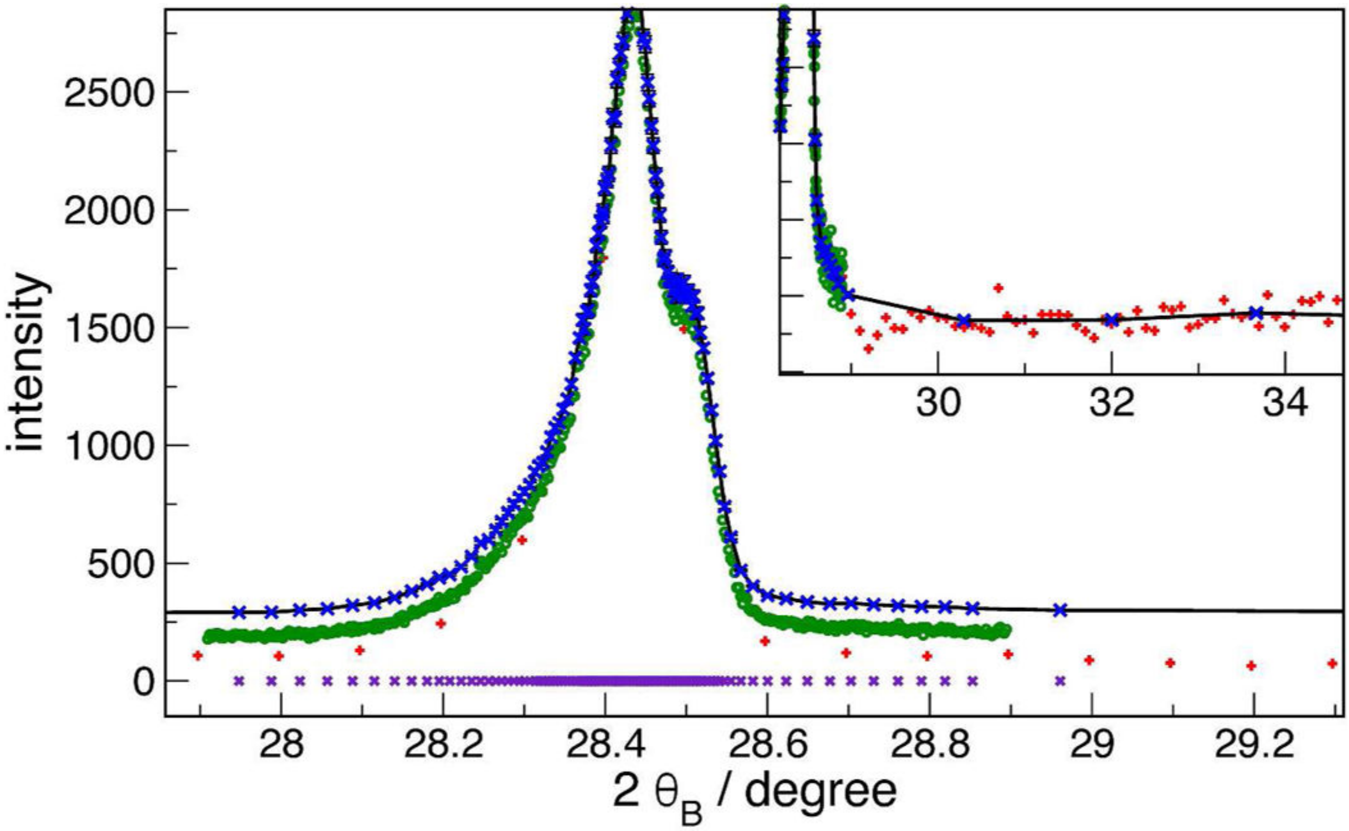

Figure 1:

Example of rebinned data from Cu$K\alpha$ diffraction from silicon powder. Green circles, high-resolution on-peak scan. Red ’+’, low-resolution survey scan. Blue crosses, rebinned combination showing variable bin spacing with $\varepsilon = 2$ %. The data sets are offset vertically for clarity. Violet crosses at the bottom are the rebinned set projected down to the x axis, to make it easier to see the adaptive point spacing. Inset shows a vertically expanded region where the count rate is very low.

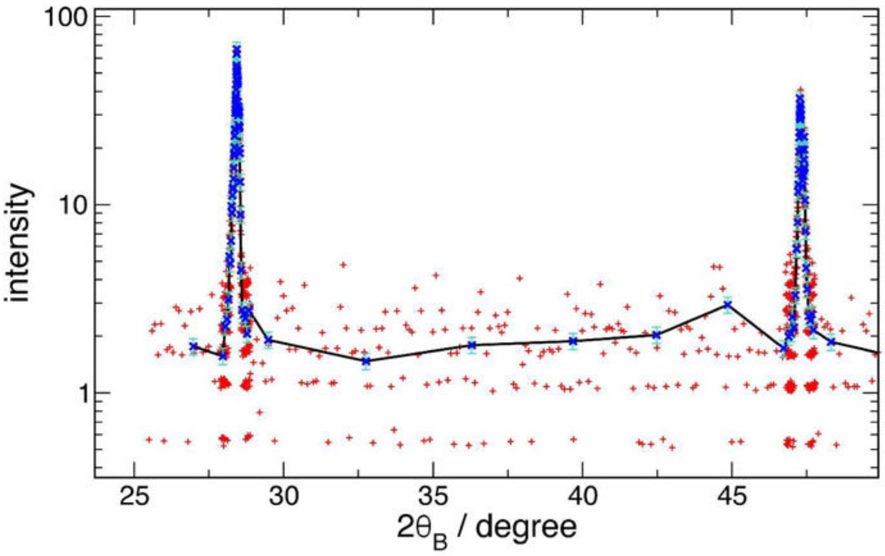

Figure 2:

Log-scale plotted data showing benefit of rebinning to readability of baseline below peaks. Red '+' are semi-synthetic data; blue 'x' with line are rebinned. Error bars are 1σ of the aggregated data.

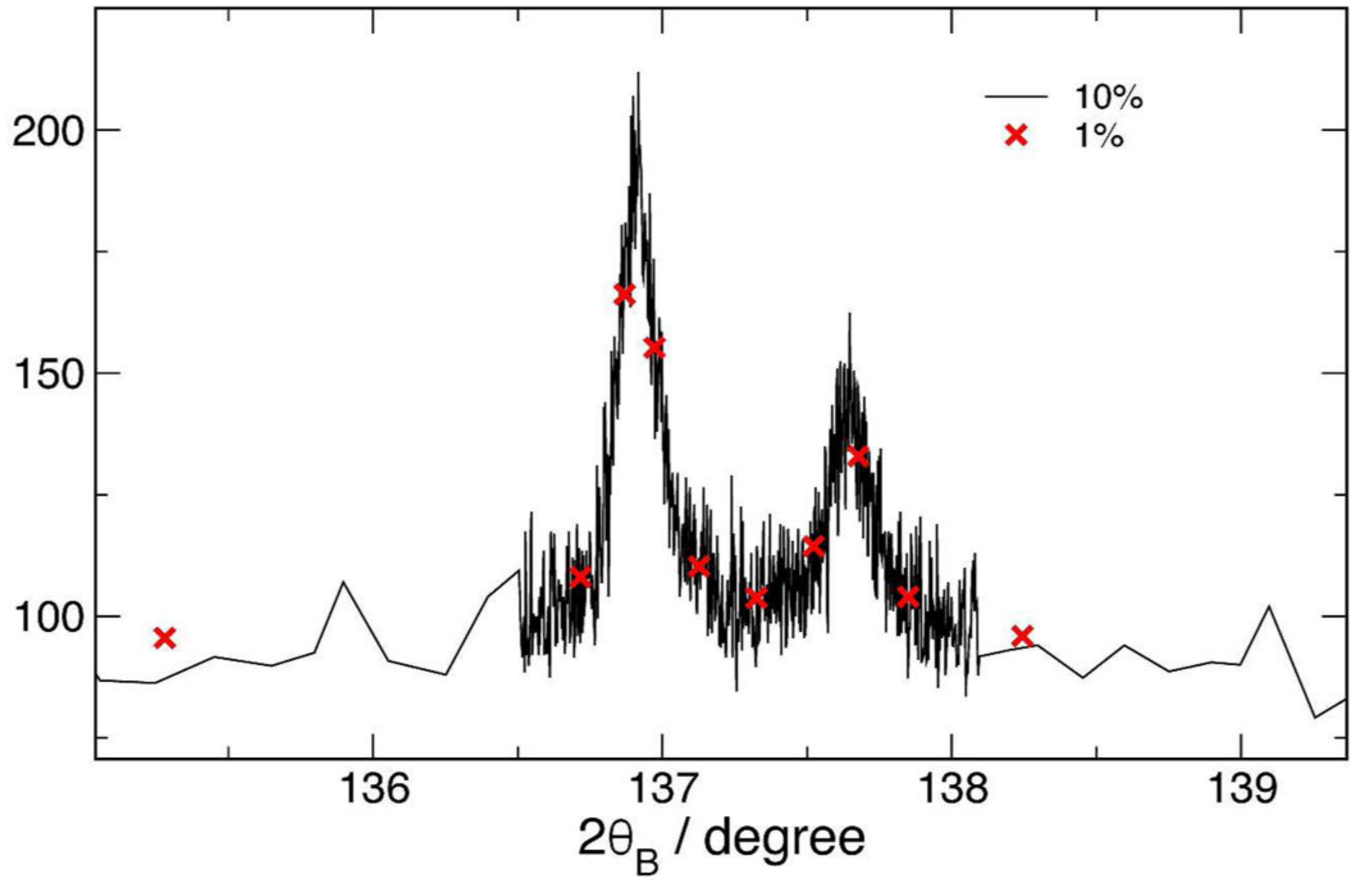

Figure 3:
Comparison of minimally aggregated data to highly aggregated data used for fits in table 1. This is a detail of a weak region of the entire angular range from 20 to 140 degrees. The data labels are the tolerance used in the rebinning.

Table 1:

Results of fitting data set with varying adaptive rebinning tolerance ε. Errors reported are pure statistical 1σ.

tolerance ε (%)	set size (points)	fit time (s)	lattice (pm)	size (nm)	Soller width (degree)
10	6686	16	543.1008 ± 0.002	568 ± 13	9.6 ± 0.2
5	2599	13	543.1015 ± 0.002	568 ± 13	9.5 ± 0.2
2	638	3	543.1013 ± 0.001	575 ± 13	9.5 ± 0.1
1	176	3	543.1048 ± 0.003	509 ± 20	9.3 ± 0.2