

# Systematic underestimation of uncertainties by widespread neutron-scattering data-reduction software

Simon Heybrock\*, Jan-Lukas Wynen and Neil Vaytet

European Spallation Source ERIC, P.O. Box 176, SE-221 00, Lund, Sweden

E-mails: [simon.heybrock@ess.eu](mailto:simon.heybrock@ess.eu), [jan-lukas.wynen@ess.eu](mailto:jan-lukas.wynen@ess.eu), [neil.vaytet@ess.eu](mailto:neil.vaytet@ess.eu)

**Abstract.** Data-reduction software used at neutron-scattering facilities around the world, Mantid and Scipp, ignore correlations when propagating uncertainties in arithmetic operations. Normalization terms applied during data-reduction frequently have a lower dimensionality than the quantities being normalized. We show how the lower dimensionality introduces correlations, which the software does not take into account in subsequent data-reduction steps such as histogramming, summation, or fitting. As a consequence, any uncertainties in the normalization terms are strongly suppressed and thus effectively ignored. This can lead to erroneous attribution of significance to deviations that are actually pure noise, or to overestimation of significance in final data-reduction results that are used for further data analysis. We analyze this flaw for a number of different cases as they occur in practice. For the two concrete experiments that are comprised in these case studies the underestimation turns out to be of negligible size. There is however no reason to assume that this generalizes to other measurements at the same or at different neutron-scattering beamlines. We describe and implement a potential solution that yields not only corrected error estimates but also the full variance-covariance matrix of the reduced result with minor additional computational cost.

Keywords: Software, neutron data-reduction

## 1. Introduction

The number of neutron counts measured in a detector element during a neutron-scattering experiment follows a Poisson distribution. In practice this is typically handled with a Gaussian approximation, setting the standard-deviation  $\sigma = \sqrt{\text{counts}}$ , which is a good approximation for a large number of counts. After acquisition of the raw detector counts, *data-reduction* is performed with the goal to compute, e.g., a momentum-transfer dependent elastic differential cross section  $P(Q)$ . An important part of the result are the statistical uncertainties of  $P(Q)$ . Software targeted at this data-reduction step such as Mantid [1] and Scipp<sup>1</sup> [12,23] handle propagation of uncertainty automatically [7]. Error propagation in *event-mode* [19,20] is also supported and allows for delaying lossy histogramming and rebinning procedures. The software's implicit or explicit promise to the user – usually a scientist – is that the automatic propagation of uncertainties will ensure they can rely on the statistical significance of their result. As a concrete example, they may rely on the significance of a fit to a peak in  $P(Q)$ , which is linked to the size of the error bars of  $P(Q)$ .

One caveat in both Mantid and Scipp is that correlations are not taken into account in the propagation of uncertainties. This is generally not considered a problem since the raw data are assumed to be uncorrelated. However, as we will show, ubiquitous operations in neutron-scattering data-reduction introduce such correlations in intermediate computation steps but subsequently fail to take these into account. This leads to a systematic underestimation

---

\*Corresponding author. E-mail: [simon.heybrock@ess.eu](mailto:simon.heybrock@ess.eu).

<sup>1</sup>Disclaimer: The authors of the present paper are the principal authors of Scipp and have previously contributed to Mantid for several years.

of uncertainties in final results such as  $P(Q)$ . The issue we describe is not hinging on any neutron-scattering-specific aspect and would therefore pertain to any code or software implementing similar step-wise propagation of uncertainties. The main examples in our case are (1) normalization of counts measured in detector pixels to, e.g., a neutron monitor spectrum, (2) normalization in event-mode, where neutron event weights are scaled with the value in the corresponding bin of the normalization term histogram,<sup>2</sup> and (3) a combination of (1) and (2). We will show that this is not simply an issue of an inaccurate approximation. Whether this underestimation is of relevant size depends on the experiment design, in particular on whether all normalization terms have sufficient statistics.

The remainder of this work is organized as follows. We describe the problem in Section 2. In Section 3 we analyze a couple of concrete examples, in particular for small-angle neutron scattering (SANS) and powder diffraction. Furthermore, we derive full expressions for the variance-covariance matrices of the reduced data, which turn out to be simple and computationally tractable. We confirm the correctness of the improved estimates for the uncertainties using a bootstrapping [9] approach. The results are discussed in Section 4 and we draw conclusions in Section 5.

## 2. Problem description

For neutron-scattering data-reduction the most relevant operations that require handling of uncertainties are addition, multiplication, and division. Normalization operations can be implemented as division or as multiplication with the inverse. The Taylor series expansion (truncated after first order) for uncertainty propagation [3] gives

$$\text{var}(x + y) = \text{var}(x) + \text{var}(y) + 2 \text{cov}(x, y), \quad (1)$$

$$\text{var}(x - y) = \text{var}(x) + \text{var}(y) - 2 \text{cov}(x, y), \quad (2)$$

$$\text{var}(x \cdot y) \approx (xy)^2 \left[ \frac{\text{var}(x)}{x^2} + \frac{\text{var}(y)}{y^2} + 2 \frac{\text{cov}(x, y)}{xy} \right], \quad (3)$$

$$\text{var}\left(\frac{x}{y}\right) \approx \left(\frac{x}{y}\right)^2 \left[ \frac{\text{var}(x)}{x^2} + \frac{\text{var}(y)}{y^2} - 2 \frac{\text{cov}(x, y)}{xy} \right], \quad (4)$$

where  $\text{var}$  and  $\text{cov}$  denote the variance and covariance, respectively. The variance is given by the square of the standard deviation, e.g.,  $\text{var}(x) = \sigma_x^2$ . The uncertainty propagation in both Mantid and Scipp (current-software, abbreviated CS from here on) is based on this expansion. For the raw input data such as neutron counts measured in a detector pixel or in a neutron monitor, CS always assumes that the inputs are uncorrelated, i.e., that any covariance term  $\text{cov}(x, y) = 0$ .<sup>3</sup> However, correlations are introduced in operations that *broadcast* [5], i.e., given an array  $\mathbf{a} = [a_1, \dots, a_n]$  and  $\mathbf{ba} = [a_1b, \dots, a_nb]$  with a scalar  $b$ , the elements of the array  $\mathbf{ba}$  are correlated with

$$\text{cov}(a_ib, a_jb) = a_ia_j \text{var}(b). \quad (5)$$

Depending on the operation,  $i$  and  $j$  might label detector pixels, events within a bin, or both. Subsequent operations frequently include *summation* over  $i$ , i.e., the sum of all elements of  $\mathbf{ba}$  (or subsets of elements), a *fit* to an  $i$ -dependent function, or in general a computation of any function  $f(\mathbf{a}, b)$ . The CS implementations ignore covariance terms as the one given in Eq. (5). This is obviously incorrect, but in widespread use. The remainder of this manuscript offers a more detailed analysis of the consequences in a number of cases occurring in practice, but adds no further fundamental insights beyond the above statement.

<sup>2</sup>The normalization term could be, e.g., a neutron monitor spectrum or a reduced vanadium spectrum.

<sup>3</sup>Whether this assumption is always correct is not the topic of this manuscript.

The most straightforward example of the incorrect treatment of correlations is the computation of a histogram of event data after normalization. As described in [19], CS event-mode normalization first histograms the normalization term. Then, for every event of the data to normalize, the corresponding bin in the normalization histogram is identified and the event weight is scaled with the inverse of the value of that bin. We consider a single bin with  $n$  events, their weights  $a_i$ , and a bin-specific (but not event-specific) normalization term  $b$ . The value of the histogram in the bin is

$$f(\mathbf{a}, b) = \sum_{i=1}^n \frac{a_i}{b}. \quad (6)$$

Using Eq. (4), and subsequently Eqs (1) and (5), we obtain<sup>4</sup>

$$\text{var}(f(\mathbf{a}, b)) = \text{var}\left(\sum_{i=1}^n \frac{a_i}{b}\right) \approx \sum_{i=1}^n \left(\frac{a_i}{b}\right)^2 \left[\frac{\text{var}(a_i)}{a_i^2} + \frac{\text{var}(b)}{b^2}\right] + \sum_{\substack{i,j=1 \\ i \neq j}}^n \text{cov}\left(\frac{a_i}{b}, \frac{a_j}{b}\right) \quad (7)$$

$$= \frac{1}{b^2} \sum_{i=1}^n \text{var}(a_i) + \frac{\text{var}(b)}{b^4} \sum_{i=1}^n a_i^2 + \sum_{\substack{i,j=1 \\ i \neq j}}^n a_i a_j \text{cov}\left(\frac{1}{b}, \frac{1}{b}\right) \quad (8)$$

$$\approx \frac{1}{b^2} \sum_{i=1}^n \text{var}(a_i) + \frac{\text{var}(b)}{b^4} \sum_{i=1}^n a_i^2 + \frac{\text{var}(b)}{b^4} \sum_{\substack{i,j=1 \\ i \neq j}}^n a_i a_j \quad (9)$$

$$= \frac{1}{b^2} \sum_{i=1}^n \text{var}(a_i) + \frac{\text{var}(b)}{b^4} \sum_{i=1}^n \sum_{j=1}^n a_i a_j. \quad (10)$$

The key aspect of this result is the double sum multiplying the  $\text{var}(b)$  term. By contrast, without taking the covariance term into account, CS computes

$$\text{var}_{\text{CS}}\left(\sum_{i=1}^n \frac{a_i}{b}\right) = \frac{1}{b^2} \sum_{i=1}^n \text{var}(a_i) + \frac{\text{var}(b)}{b^4} \sum_i a_i^2 \not\approx \text{var}\left(\sum_{i=1}^n \frac{a_i}{b}\right), \quad (11)$$

with only a single sum in the  $\text{var}(b)$  term. Essentially CS is replacing the square of the sum over  $a_i$  by the sum over the squares. If all  $a_i$  are of similar magnitude then ignoring correlations is thus suppressing the contribution of  $b$ 's uncertainty  $\text{var}(b)$  (the normalization term's uncertainty) by a factor of  $1/n$ . In many typical applications  $n \gg 1$ . For such  $n$  we almost always have  $a_j^2 \ll \sum_{i=1}^n a_i a_j$  and the CS expression Eq. (11) would thus be valid only for

$$\frac{\text{var}(b)}{b^4} \sum_{\substack{i,j=1 \\ i \neq j}}^n a_i a_j \ll \frac{1}{b^2} \sum_{i=1}^n \text{var}(a_i). \quad (12)$$

---

<sup>4</sup>Here we chose to compute the same result in two steps using correlations of summands since it aids in highlighting the difference to the CS approach. A direct and likely simpler way to compute this is using  $\Sigma^f = J \Sigma^{ab} J^T = J \cdot \text{diag}(\text{var}(a_1), \dots, \text{var}(a_n), \text{var}(b)) \cdot J^T$  with Jacobian matrix  $J = \left[ \frac{\partial f}{\partial a_1} \dots \frac{\partial f}{\partial a_n} \frac{\partial f}{\partial b} \right] = \left[ \frac{1}{b} \dots \frac{1}{b} -\frac{1}{b^2} \sum_i a_i \right]$ . The diagonal of  $\Sigma^f$  yields the variances  $\text{var}(f(\mathbf{a}, b))$ .

From this we obtain the condition

$$\frac{\text{var}(b)}{b^2} \ll \frac{\sum_{i=1}^n \text{var}(a_i)}{\sum_{\substack{i,j=1 \\ i \neq j}}^n a_i a_j} \approx \frac{1}{n-1} \frac{\text{var}(a)}{a^2}, \quad (13)$$

where the approximation is valid for  $a_i \approx a$  and  $\text{var}(a_i) \approx \text{var}(a)$ . However, fulfilling Eq. (13) implies that the second terms in Eqs (10) and (11) are negligible. That is, the CS approach is valid only when the term it approximates is effectively zero. Computing the second term in Eq. (11) is thus simply a waste of computational resources and energy, more than doubling the number of required floating point operations.<sup>5</sup> Hereafter the *zero-variance approximation* (V0) will denote an approach where the variances of the normalization terms are strictly set to 0.

Concretely, for the example of histogramming event data, all neutrons typically have the same initial weight  $a_i = 1$  and  $\text{var}(a_i) = 1$ . Then the number of events  $n$  is equal to the number of counts  $C$  and Eq. (10) turns into

$$\text{var}\left(\sum_{i=1}^C \frac{a_i}{b}\right) \approx \frac{C}{b^2} + \frac{C^2 \text{var}(b)}{b^4}. \quad (14)$$

The second term is negligible for

$$\frac{C^2 \text{var}(b)b^2}{b^4 C} = \frac{C \text{var}(b)}{b^2} \ll 1. \quad (15)$$

Often the normalization term itself originates from a histogram of neutron event counts. Then  $b = N$ ,  $\text{var}(b) = \text{var}(N) = N$  and we have

$$\frac{C \text{var}(N)}{N^2} = \frac{CN}{N^2} = \frac{C}{N} = \alpha \ll 1. \quad (16)$$

This equation states that both the CS and V0 approximations are valid only if the number of detector counts  $C$  is (much) lower than the number of normalization counts  $b$ . The ratio of the two,  $\alpha$ , may be used as a generic indicator of the validity of the approximation.

To illustrate this effect, we show in Fig. 1 how the relative uncertainty for a histogram of normalized event data scales with the number of events. Both the CS implementation ( $\times$ ) and the V0 approximation ( $+$ ) of event-mode normalization lead to a relative error of the histogrammed result that goes to 0 with increasing event count. Importantly, CS and V0 yield near-identical results for all values of  $C$ . In contrast, normalization after histogramming ( $\bullet$ ) or correct handling of correlations yields a relative error limited by the relative error of the normalization factor.

In summary, the CS approach is thus either wasteful (since it is equivalent to V0 with additional compute overhead), or incorrect. In the next section we analyze more complex cases, namely normalization in SANS and powder diffraction workflows. These involve a transformation of coordinates<sup>6</sup> after the normalization step but before a histogramming/summing step. While the problem of the CS approach is conceptually the same, the concrete effects are thus slightly harder to visualize and grasp.

<sup>5</sup>Multiplication  $ab$ : 3 multiplications ( $b$  without variance) vs. 5 multiplications and 1 addition ( $b$  with variance); Division  $a/b$ : 2 divisions and 1 multiplication ( $b$  without variance) vs. 3 divisions, 3 multiplications, and 1 addition ( $b$  with variance); Numbers may vary based on implementation and when standard-deviations instead of variances are stored, as is the case in parts of Mantid.

<sup>6</sup>In Mantid such coordinate transformations are referred to as unit conversion.

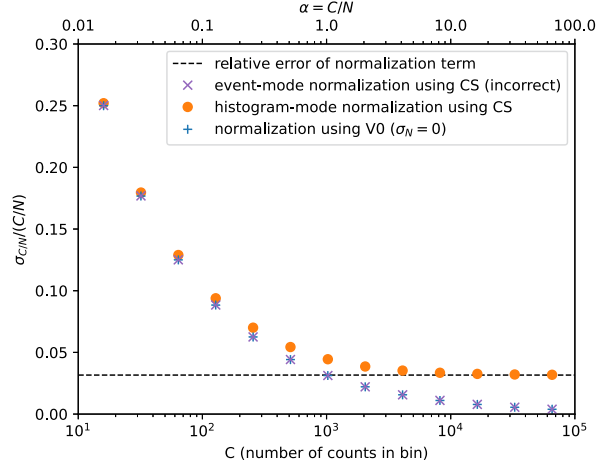


Fig. 1. Relative error  $\sigma_{C/N}/(C/N)$  of a single histogram bin computed by CS for normalization to a fixed number of  $N = 1000$  counts. The relative error of the normalization term,  $\sigma_N = \sqrt{N}/N \approx 0.032$ , is shown as a dashed line and should represent the limit and lower bound of the relative error for large number  $C$  of events in the bin. We show ( $\times$ ) relative error obtained from histogramming after normalization (incorrect), ( $+$ ) relative error obtained when forcing  $\sigma_N = 0$ , i.e., when ignoring uncertainties of the normalization term, and ( $\bullet$ ) relative error obtained from normalizing the histogrammed data (correct). In the V0 case the result is independent of the order of normalization and histogramming. The two CS implementations, Mantid and Scipp, yield results identical within floating-point rounding errors, so we are not plotting results separately.

### 3. Case studies

#### 3.1. SANS

In a time-of-flight SANS experiment, the differential scattering cross section  $S(Q)$  is defined as [11]

$$S(Q) = \frac{\partial \Sigma Q}{\partial \Omega} = \frac{A_H \sum_{i,j,\lambda \in Q} C(i, j, \lambda)}{A_M t \sum_{i,j,\lambda \in Q} M_{\text{sample}}^{\text{incident}}(\lambda) T(\lambda) D(\lambda) \Omega(i, j)}, \quad (17)$$

$$T(\lambda) = \frac{M_{\text{sample}}^{\text{transmission}}}{M_{\text{sample}}^{\text{incident}}} \frac{M_{\text{direct}}^{\text{incident}}}{M_{\text{direct}}^{\text{transmission}}}. \quad (18)$$

Here  $A_H$  is the area of a mask (which avoids saturating the detector) placed between the monitor of area  $A_M$  and the main detector,  $\Omega$  is the detector solid angle,  $C$  is the count rate on the main detector, which depends on the pixel indices  $i, j$  and the wavelength  $\lambda$ ,  $t$  is the sample thickness,  $M$  represents the incident or transmission monitor count rate for the sample run or the direct run, and  $T$  is known as the transmission fraction, and finally,  $D$  is the direct beam function.<sup>7</sup>  $\sum_{i,j,\lambda \in Q}$  denotes a sum over all pixels and wavelengths that contribute to the desired  $Q$ -bin. Note that  $M_{\text{sample}}^{\text{incident}}$  cancels with the corresponding divisor in  $T$ .  $S(Q)$  is computed for a sample run and a background run, and the background-subtracted  $P(Q) = S(Q) - S_{\text{background}}(Q)$  is then used for further analysis.

In the SANS case the CS calculation of the normalization term itself (rather than the normalization of the detector counts) in Eq. (17) suffers from the issue described in Section 2. In contrast to the example from Fig. 1, the problem here is not caused by event-mode normalization, but by a different type of broadcast operation. Since the  $\lambda$ -dependent factors in the denominator do not depend on  $i$  and  $j$ , and the solid angle  $\Omega(i, j)$  does not depend

<sup>7</sup>The direct-beam function allows to cross-normalize the incident spectrum to that of the empty beam (without sample) seen on the main detector.

on  $\lambda$ , multiplying them requires a broadcast in both  $\lambda$  and  $(i, j)$ .<sup>8</sup> The CS implementation then sums the product over  $\lambda, i, j$ , ignoring correlations. Finally, the resulting uncertainties are combined with the uncertainties from the numerator.

Note that there are a number of other contributions to the uncertainties with similar issues, which we ignored in this analysis. Our goal here is to provide an example for illustration of the problems in CS, not a complete analysis of the entire SANS data reduction. See the discussion in Section 4 for more background.

To illustrate the issue we study the data-reduction workflow for an experiment on non-ionic surfactants that was carried out at the Sans2d instrument [22] at the ISIS facility at the STFC Rutherford Appleton Laboratory.<sup>9</sup> The SANS reduction workflow we used can be found online at [21]. A key factor in determining whether the CS implementation is adequate is the ratio of total detector counts to total monitor counts<sup>10</sup>

$$\alpha = \frac{\sum_{i,j,\lambda} C(i, j, \lambda)}{\sum_{\lambda} M_{\text{sample}}^{\text{transmission}}(\lambda)}. \quad (19)$$

This can vary in practice based on the instrument design, in particular the monitor efficiency. A larger sample volume or a sample that is a better scatterer will also lead to a larger  $\alpha$ . For example, a  $10\times$  larger sample would scatter approximately  $10\times$  more neutrons and yield correspondingly higher detector counts while the monitor counts would be approximately unchanged. To highlight the influence of  $\alpha$ , we simulated  $\alpha > \alpha_{\text{actual}}$  by drawing, for an actual monitor bin with  $M(\lambda)$  counts, from a Poisson distribution  $f(k; \Lambda)$  with  $\Lambda = \frac{\alpha_{\text{actual}}}{\alpha} M(\lambda)$  and reran the reduction workflow. In Fig. 2 we compare the uncertainties for  $S(Q)$  computed by CS and an improved analytical approach (which will be described below) for three values of  $\alpha$ . The source code for reproducing these results can be found in the supplementary material [13]. In this concrete experiment  $\alpha_{\text{actual}} = 0.27$  and the difference between the CS result and corrected result is negligible. For larger  $\alpha$  we observe a clear difference.

As a verification of these results, we independently computed the uncertainties by using a bootstrap [9] method – a sampling technique that can be used to estimate the distribution of a statistic.<sup>11</sup> It involves repeatedly sampling from a population with replacement, and then calculating the statistic of interest for each sample. The resulting distribution of statistics can then be used to calculate a confidence interval for the population parameter. This approach is useful because it allows us to make inferences about a population even when we do not have a theoretical distribution for the statistic of interest. The biggest advantage of this approach is its simplicity, but it comes at a high cost – in our case we need to repeat the entire data-reduction for every bootstrap sample, i.e., hundreds or thousands of times. The bootstrapping approach is described in some more detail in Appendix A. We can see in Fig. 3 that there is an excellent agreement between the bootstrap results and our analytical derivation (see Appendix A, in particular Fig. 8 which shows the same result for  $R = 1024$  together with additional values of  $R$ ).<sup>12</sup> The source code for bootstrapping is also included in the supplementary material [13].

The correct (first order) expression for the uncertainties for this reduction workflow can be derived in a straightforward manner. We adopt a simpler and more generic notation equivalent to Eq. (17) as follows. We define

$$c_Q = A_H \sum_{i,j,\lambda \subset Q} C(i, j, \lambda), \quad (20)$$

$$p_k = \Omega(i, j) \quad \text{with } k = (i, j), \quad (21)$$

$$w_{\lambda} = A_M t M_{\text{sample}}^{\text{incident}}(\lambda) T(\lambda) D(\lambda). \quad (22)$$

<sup>8</sup>Here we assumed quasi-elastic scattering. A correction factor for inelastic-scattering [14] would depend on  $i, j$ , and  $\lambda$ .

<sup>9</sup>The data has been published in [17], where all the details on the experimental setup can be found.

<sup>10</sup>The monitors from the direct run,  $M_{\text{direct}}^{\text{incident}}$  and  $M_{\text{direct}}^{\text{transmission}}$  also play a role, but we consider them as a given invariant for this analysis.

<sup>11</sup>For related methods such as Markov-chain Monte-Carlo uncertainty estimates see, e.g., Ref. [2] and [18].

<sup>12</sup>Including background subtraction in our bootstrap calculations, i.e., computing  $P(Q)$  instead of just  $S(Q)$ , yielded very similar results for the variances as well as the correlations.

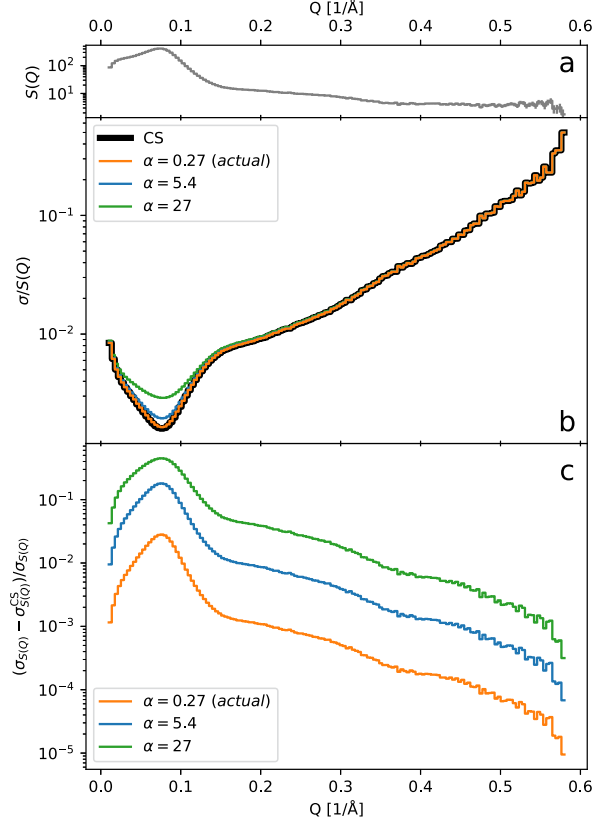


Fig. 2. (a) The scattering differential cross-section  $S(Q)$  computed by the SANS workflow. (b) Relative uncertainty of the cross section  $\Sigma^S$  for the SANS workflow. The thick black curve represents the results obtained with CS, while the coloured lines are the analytical calculations with varying  $\alpha$  coefficients. The results obtained with CS are virtually identical for all values of  $\alpha$ . (c) Relative difference in standard deviations between CS and the analytical calculation.

In vector notation this yields the input parameter vectors  $\mathbf{c} = (c_{Q_1}, \dots, c_{Q_C})^T$  (the detector counts binned into  $Q$ -bins of width  $\Delta Q$ ),<sup>13</sup>  $\mathbf{p} = (p_1, \dots, p_P)^T$  (the pixel-dependent scaling factors), and  $\mathbf{w} = (w_{\lambda_1}, \dots, w_{\lambda_W})^T$  (the wavelength-dependent scaling factors such as monitor counts), where  $C$ ,  $P$ , and  $W$  denote the number of  $Q$ -bins, pixels, and wavelength-bins, respectively. Typically we assume that the respective variance-covariance matrices  $\Sigma^c$ ,  $\Sigma^p$ , and  $\Sigma^w$  are diagonal, e.g.,  $\Sigma^c = \text{diag}(\text{var}(c_1), \dots, \text{var}(c_{Q_C}))$ , but the following derivation is valid even if they are not. Aside from constant factors, Eq. (17) is then equivalent to

$$S_Q = \frac{c_Q}{\mathbf{p}^T \mathcal{H}_Q \mathbf{w}} = c_Q n_Q, \quad \text{where } (\mathcal{H}_Q)_{i\lambda} = \begin{cases} 1 & \text{for } Q_{i\lambda} \in [Q, Q + \Delta Q) \\ 0 & \text{otherwise} \end{cases}. \quad (23)$$

$S_Q$  are the components of the discrete differential scattering cross section  $\mathbf{S}$ . The matrices  $\mathcal{H}_Q$  encode the histogramming procedure, equivalent to the  $\sum_{i,j,\lambda \in Q}$  sum in Eq. (17).<sup>14</sup> We use the usual first-order truncated Taylor

<sup>13</sup>We use bins of constant width to simplify the notation, but the following results are equally valid for other choices of bin widths.

<sup>14</sup>Note that in software histogramming is not actually implemented via a matrix multiplication as this would be highly inefficient.

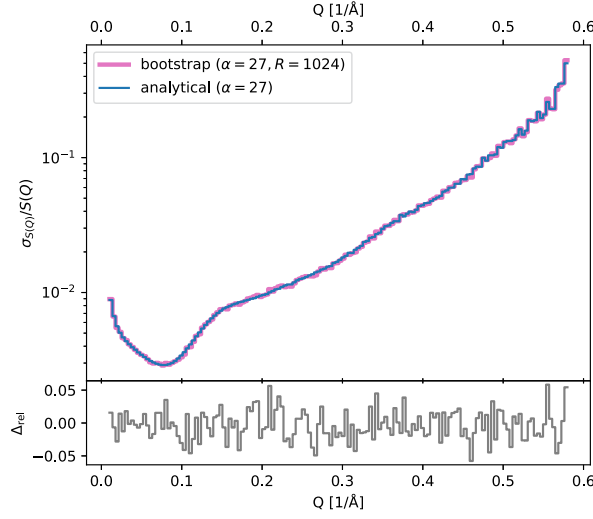


Fig. 3. Comparison of the relative standard deviation for  $S(Q)$  as computed by our improved analytical approach (blue) and the bootstrap approach (pink) at  $\alpha = 100 \cdot \alpha_{\text{actual}} = 27$ . The small bottom panel shows the relative difference between the two.  $R$  is the number of bootstrap replicas, which influences the amount of noise on the result (see Appendix A).

expansion to compute the variance-covariance matrix  $\Sigma^S$  of  $S$  from the variance-covariance matrices of the inputs,

$$\Sigma^S \approx \mathbf{J} \begin{bmatrix} \Sigma^c & & \\ & \Sigma^p & \\ & & \Sigma^w \end{bmatrix} \mathbf{J}^T, \quad \text{with the Jacobian matrix } \mathbf{J} = [\nabla_c^T S \quad \nabla_p^T S \quad \nabla_w^T S], \quad (24)$$

where  $T$  denotes the matrix-transpose and the rows of the three submatrices are given by

$$\nabla_c S_Q = n_Q \hat{e}_Q, \quad (25)$$

$$\nabla_p S_Q = -c_Q n_Q^2 \mathcal{H}_Q \mathbf{w}, \quad (26)$$

$$\nabla_w S_Q = -c_Q n_Q^2 \mathcal{H}_Q^T \mathbf{p}. \quad (27)$$

$\hat{e}_Q$  denotes the unit vector for direction  $Q$ . This yields for the elements of  $\Sigma^S$

$$\Sigma_{QQ'}^S \approx n_Q n_{Q'} \Sigma_{QQ'}^c + c_Q c_{Q'} n_Q^2 n_{Q'}^2 [\mathbf{w}^T \mathcal{H}_Q^T \Sigma^p \mathcal{H}_{Q'} \mathbf{w} + \mathbf{p}^T \mathcal{H}_Q \Sigma^w \mathcal{H}_{Q'}^T \mathbf{p}]. \quad (28)$$

In current practice, the uncertainties of the pixel-dependent term (such as the solid angle of detector pixels) are typically ignored, i.e.,  $\Sigma^p = \mathbf{0}$ . Then

$$\Sigma_{QQ'}^S \approx n_Q n_{Q'} \Sigma_{QQ'}^c + c_Q c_{Q'} n_Q^2 n_{Q'}^2 \mathbf{p}^T \mathcal{H}_Q \Sigma^w \mathcal{H}_{Q'}^T \mathbf{p}. \quad (29)$$

The  $\Sigma^w$  term involves histogramming the pixel-dependent  $\mathbf{p}$  into a matrix of  $(Q, \lambda)$ -bins. This is cheap to compute, in particular since for typical SANS experiments we have only moderate bin counts,  $N_Q = \mathcal{O}(100)$  and  $N_\lambda = \mathcal{O}(100)$ . The remaining operations for this term then involve row (or column) scaling as well as matrix-matrix multiplications of the matrices computed by the histogramming procedure. We refer to Section 4.3 for a more detailed look into the performance impact of the improved analytical approach.

The diagonal component of this result is included in Fig. 2. In addition, we show the resulting Pearson correlation matrix  $\text{corr}(S_Q, S_{Q'}) = \Sigma_{QQ'}^S / \sqrt{\text{var}(S_Q) \text{var}(S_{Q'})}$  in Fig. 4. This figure also includes the equivalent results



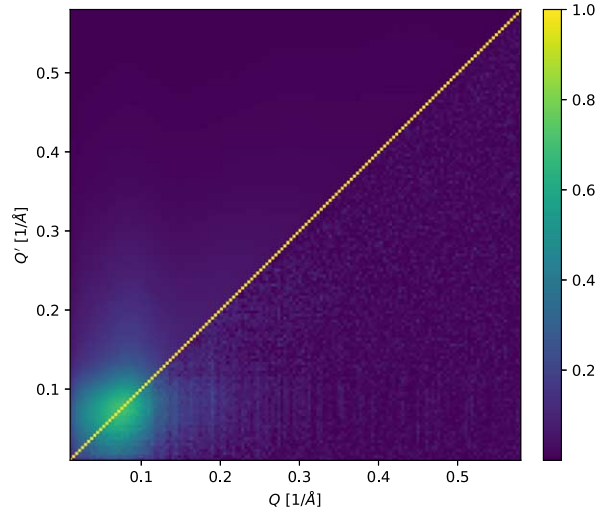


Fig. 4. Correlation matrices for the time-of-flight SANS-reduction workflow using  $\alpha = 100 \cdot \alpha_{\text{actual}} = 27$ . As the correlation matrix is symmetric under  $Q \leftrightarrow Q'$ , we can show in the same figure the analytical result (above the diagonal) and the bootstrap result (below the diagonal,  $R = 1024$  replicas). See Appendix B for a version of the figure with a logarithmic color scale.

computed using the bootstrapping approach and – as previously shown for the variances – there is excellent agreement for the covariances or correlations. Overall, there clearly are correlations and these should be taken into account when fitting a model to the data.<sup>15</sup>

### 3.2. Powder diffraction

We consider a simplified model of a time-of-flight powder-diffraction reduction workflow based on the workflow used at ISIS [15]. Our model workflow consists of the following steps: (1) compute wavelength coordinates for detector data and monitor data, (2) compute wavelength histograms, (3) normalize the detector data by the monitor, (4) compute  $d$ -spacing coordinate for detector data, and (5) compute  $d$ -spacing histogram, combining data from all (or a user-defined subset of) detector pixels. This yields the normalized  $d$ -spacing spectrum  $S_d$  (for the sample) or  $V_d$  (for a vanadium run). We will consider  $V_d$  and the normalized probability distribution  $P_d = S_d/V_d$  in the next section. Note that in contrast to the ISIS powder reduction workflow, the SNS workflow [24] does generally not use a monitor normalization. For our analysis we use data from the Wish beamline [6] at ISIS.<sup>16</sup>

In Fig. 5 we compare the uncertainties computed by CS and by an improved analytical approach which will be described below. Source code for reproducing these results can be found in the supplementary material [13]. As described in Section 3.1, we study the influence of the ratio of total detector to monitor counts,  $\alpha$  (defined equivalently to Eq. (19)), influenced, e.g., by the sample volume. For  $\alpha_{\text{actual}} = 2.2$  as in this example, the relative underestimation by CS is below the 10% level. The discrepancy is largest for low  $d$  and at diffraction peaks. For larger  $\alpha$  we observe a much more significant difference.

For the simplified model workflow we can derive the correct (first order) expression for the uncertainties in a straightforward manner. This also provides the covariances or correlations between different  $d$ -spacing values. We adopt the following notation. Let  $i \in [1, \dots, N_{\text{total}}]$ . For an event-mode workflow  $i$  labels events with  $N_{\text{total}} = N_{\text{event}}$ , irrespective of the detector pixel that recorded an event. Then the tuple  $(c_i, d_i, \lambda_i)$  represents the counts (usually 1),  $d$ -spacing, and wavelength for the  $i$ -th event. If not in event-mode,  $i$  labels bins across all detector

<sup>15</sup>For example, `scipy.optimize.curve_fit` accepts the covariance matrix as input via the `sigma` parameter. A full discussion for other approaches such as Bayesian statistics is beyond the scope of this paper.

<sup>16</sup>The sample is NaCaAlF measured in Wish run number 52889 and the vanadium was measured in run number 52920.

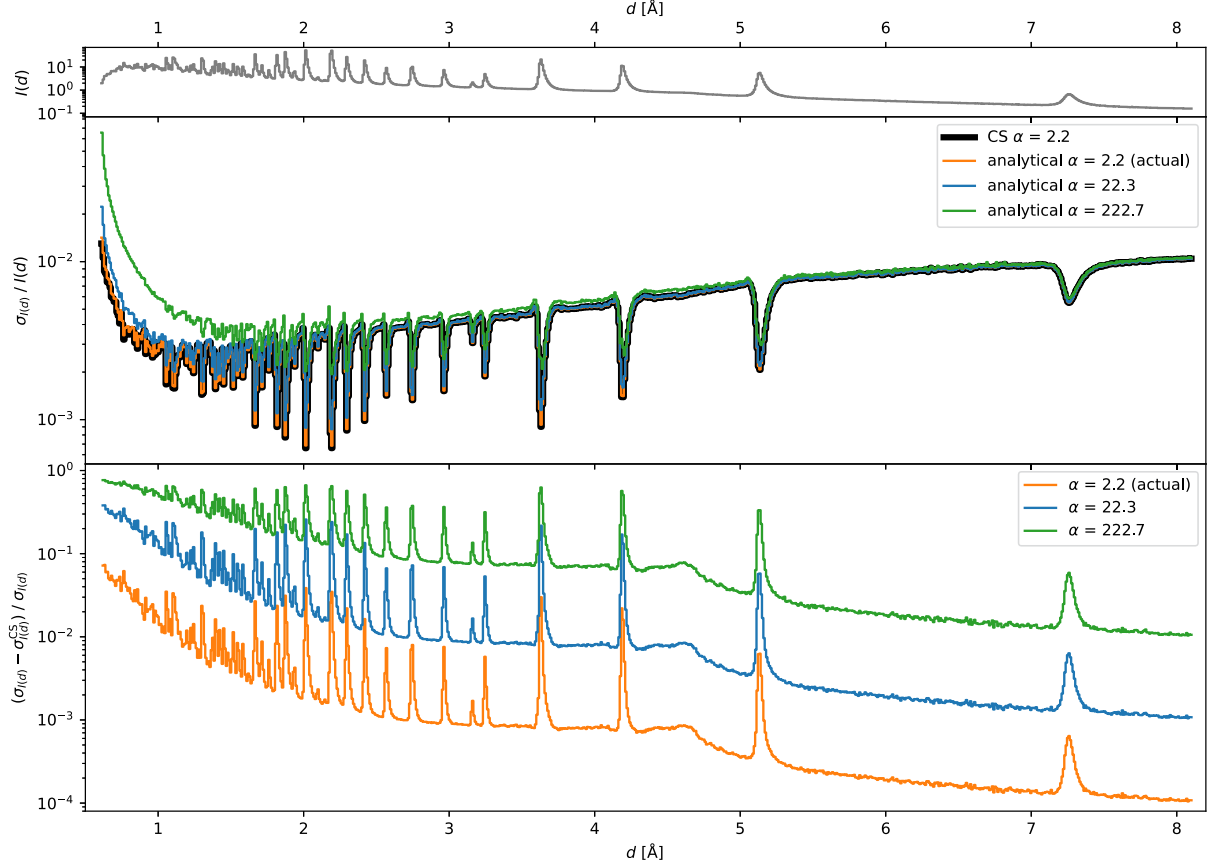


Fig. 5. Top:  $S(d)$ . Middle: relative standard deviation for  $S(d)$  as computed by CS and our improved analytical approach. Bottom: relative difference between standard deviations as computed by CS and our analytical approach. We show results for three different values of  $\alpha$ . As in Section 3.1, we simulated  $\alpha > \alpha_{\text{actual}}$  by drawing, for an actual monitor bin with  $w_\lambda$  counts, from a Poisson distribution  $f(k; \Lambda)$  with  $\Lambda = \frac{\alpha_{\text{actual}}}{\alpha} w_\lambda$  and reran the reduction workflow.

pixels, with  $N_{\text{total}} = N_\lambda N_{\text{pixel}}$  where  $N_\lambda$  is the number of wavelength bins (per pixel) and  $N_{\text{pixel}}$  is the number of detector pixels. Then the tuple  $(c_i, d_i, \lambda_i)$  represents the counts,  $d$ -spacing, and wavelength for the  $i$ -th bin. We define input parameter vectors  $\mathbf{c} = (c_1, \dots, c_{N_{\text{total}}})^T$  and  $\mathbf{w} = (w_{\lambda_1}, \dots, w_{\lambda_w})^T$  (the wavelength-dependent scaling factors such as monitor counts). Typically we assume that the respective variance-covariance matrices  $\Sigma^{\mathbf{c}}$  and  $\Sigma^{\mathbf{w}}$  are diagonal, e.g.,  $\Sigma^{\mathbf{w}} = \text{diag}(\text{var}(w_{\lambda_1}), \dots, \text{var}(w_{\lambda_w}))$ , but the following derivation is valid even if they are not. The model workflow computes  $S_d$  from detector counts  $\mathbf{c}$  and monitor counts  $\mathbf{w}$ ,

$$S_d = \mathbf{n}^T \mathcal{H}_d \mathbf{c}, \quad (30)$$

where  $\mathbf{n}$  is the element-wise inverse of  $\mathbf{w}$ , i.e.,  $\mathbf{n} = (1/w_{\lambda_1}, \dots, 1/w_{\lambda_w})^T$ . The  $\mathcal{H}_d$  matrix encodes the histogramming procedure with  $d$ -spacing bin size  $\Delta d$  and wavelength bin size  $\Delta \lambda$ ,<sup>17</sup> defined as

$$(\mathcal{H}_d)_{\lambda i} = \begin{cases} 1 & \text{for } d_i \in [d, d + \Delta d), \lambda_i \in [\lambda, \lambda + \Delta \lambda) \\ 0 & \text{otherwise.} \end{cases} \quad (31)$$

<sup>17</sup>We use bins of constant width for simplifying the notation, but the following results are also valid for other choices of bin widths.

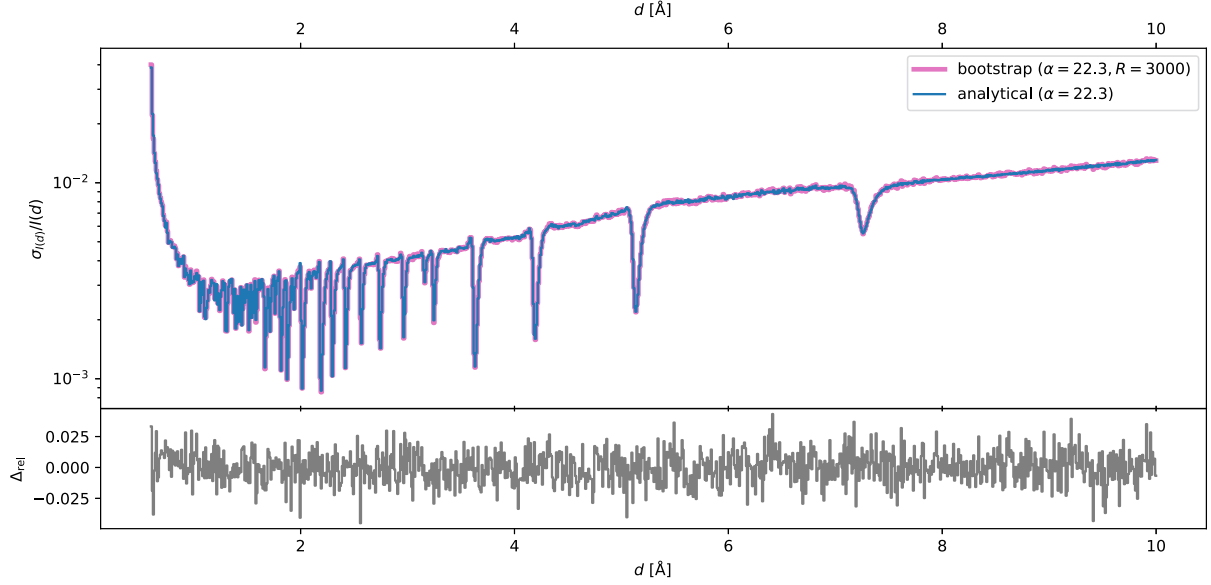


Fig. 6. Comparison of the relative standard deviation for  $S(d)$  as computed by our improved analytics approach and the bootstrap approach at  $\alpha = 10 \cdot \alpha_{\text{actual}} = 22.3$ . The small bottom panel shows the relative difference between the two.  $R$  is the number of bootstrap replicas, which influences the amount of noise on the result.

Note that the definition of  $\mathcal{H}_d$  differs from that of  $\mathcal{H}_Q$  in Eq. (23) – in the latter, the column index  $i$  labels pixels whereas in the former it labels events or bins. We use the usual first-order truncated Taylor expansion to compute the variance-covariance matrix  $\Sigma^S$  of  $S$  from the variance-covariances matrices of the inputs,

$$\Sigma^S \approx \mathbf{J} \begin{bmatrix} \Sigma^c \\ \Sigma^w \end{bmatrix} \mathbf{J}^T, \quad \text{with the Jacobian matrix } \mathbf{J} = \begin{bmatrix} \nabla_c^T S & \nabla_w^T S \end{bmatrix}. \quad (32)$$

The rows of these matrices compute as

$$\nabla_c S_d = \mathcal{H}_d^T \mathbf{n}, \quad (33)$$

$$\nabla_w S_d = -[\text{diag}(\mathbf{n})]^2 \mathcal{H}_d \mathbf{c} = -\mathbf{N}^2 \mathcal{H}_d \mathbf{c}. \quad (34)$$

The matrix  $\mathbf{N}$  is defined as  $\text{diag}(\mathbf{n})$ . Then the elements of  $\Sigma^S$  compute as

$$\Sigma_{dd'}^S \approx \mathbf{n}^T \mathcal{H}_d \Sigma^c \mathcal{H}_{d'}^T \mathbf{n} + \mathbf{c}^T \mathcal{H}_d^T \mathbf{N}^2 \Sigma^w \mathbf{N}^2 \mathcal{H}_{d'} \mathbf{c}. \quad (35)$$

For a diagonal  $\Sigma^c$  (the common case) the first term is also diagonal, i.e., this term does not contribute to the covariances. Notably, Eq. (35) is valid in event mode as well as for histogrammed data.<sup>18</sup> We will show in Section 4.3 that the computation of the variance-covariance matrix is not significantly more expensive than the computation of the cross section or the CS approach in general.

In Fig. 6 we compare the relative errors computed using the analytical approach and bootstrap (see Appendix A, in particular Fig. 9 which shows the same result for  $R = 3000$  together with additional values of  $R$ ). Additionally, Fig. 7 shows the correlation matrices computed by both approaches. The correlations are clear and long ranged for the displayed example with artificially reduced monitor counts which simulates, e.g., a  $10\times$  larger sample. When we use the actual monitor counts, the correlations are equally present, albeit weaker.

<sup>18</sup>For histogrammed data the expression may need to be adapted depending on the rebinning approach that is being used.

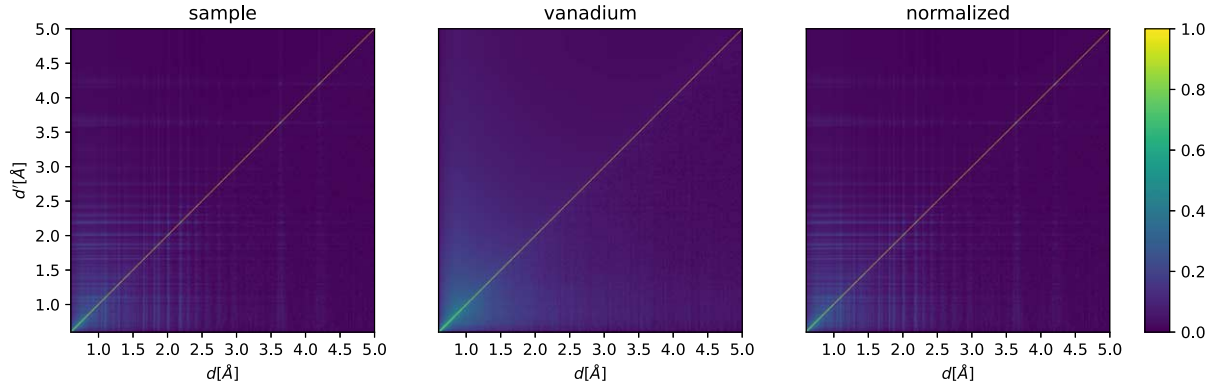


Fig. 7. Correlation matrices for a powder-reduction workflow. We show the result for the sample  $S$  (left,  $\alpha = 10 \cdot \alpha_{\text{actual}}^{\text{sample}} = 22.3$ ), vanadium  $V$  (center,  $\alpha = 10 \cdot \alpha_{\text{actual}}^{\text{vanadium}} = 47.0$ ), and the normalized  $P = S/V$  (right). As the correlation matrices  $\Sigma^S$ ,  $\Sigma^V$ , and  $\Sigma^P$  are symmetric under  $d \leftrightarrow d'$ , we can show in the same figure the analytical result (above the diagonal) and the bootstrap result (below the diagonal). For improved readability, the plots have a limited  $d$ -spacing range as there are few further features for higher  $d$ . See Appendix B for a version of the figure with a logarithmic color scale.

### 3.3. Vanadium normalization

For scattering off a vanadium sample, we define  $V_d$  equivalently to  $S_d$  for the sample under study above. The normalized probability distribution  $\mathbf{P}$  is then given by

$$P_d = \frac{S_d}{V_d}. \quad (36)$$

As  $S_d$  and  $V_d$  are assumed to be uncorrelated we obtain

$$\Sigma_{dd'}^P = \frac{\Sigma_{dd'}^S}{V_d V_{d'}} + \frac{S_d S_{d'} \Sigma_{dd'}^V}{V_d^2 V_{d'}^2}. \quad (37)$$

We can thus straightforwardly compute the variance-covariance matrix of the normalized  $d$ -spacing spectrum from the known  $\Sigma^S$  and  $\Sigma^V$ . The resulting correlations are displayed in the rightmost panel in Fig. 7.

### 3.4. The effect of smoothing

Smoothing is routinely used to reduce the effect of noise in normalization terms such as a neutron monitor spectrum or a reduced vanadium spectrum. This reduces the size of the variances of the normalization term,<sup>19</sup> and thus the relative importance of these uncertainties when combined with the variances of the detector counts. However, the smoothing procedure introduces additional correlations between bins of the normalization term which counterbalance the effect. Smoothing can be described as a convolution with a kernel function  $\omega$ . We repeat the derivation for powder diffraction from Section 3.2 with a smoothed wavelength-dependent monitor normalization term  $\mathbf{w}$ . Define

$$\tilde{\mathbf{w}} = \omega * \mathbf{w}, \quad (38)$$

$$\tilde{w}_n = (\omega * w)_n = \sum_m \omega_{n-m} w_m, \quad (39)$$

<sup>19</sup>Note that, e.g., Mantid's FFTSmooth (version 2) algorithm sets uncertainties directly to 0. This behavior appears to be undocumented.

where  $*$  denotes the convolution. Then

$$\Sigma^{\tilde{w}} \approx \mathbf{J}^{w\tilde{w}} \Sigma^w \mathbf{J}^{w\tilde{w}\text{T}}, \quad (40)$$

$$\mathbf{J}_{nn'}^{w\tilde{w}} = \frac{\partial \tilde{w}_n}{\partial w_{n'}} = \frac{\partial (\omega * w)_n}{\partial w_{n'}} = \sum_m \omega_{n-m} \delta_{mn'} = \omega_{n-n'}. \quad (41)$$

That is, the Jacobian is a diagonal-constant matrix, a Toeplitz matrix,  $\Omega$  with elements  $\Omega_{nn'} = \omega_{n-n'}$ . As an example, for a Gaussian filter this matrix would be symmetric, and for a typical discrete approximation with small finite support it would also be band-diagonal. Equivalently to Eq. (35) we then obtain

$$\Sigma_{dd'}^{\tilde{S}} = \tilde{\mathbf{n}}^{\text{T}} \mathcal{H}_d \Sigma^c \mathcal{H}_d^{\text{T}} \tilde{\mathbf{n}} + \mathbf{c}^{\text{T}} \mathcal{H}_d^{\text{T}} \tilde{N}^2 \Omega \Sigma^w \Omega^{\text{T}} \tilde{N}^2 \mathcal{H}_d' \mathbf{c}. \quad (42)$$

To compute the second term we may thus smooth the histogrammed and scaled counts  $\tilde{N}^2 \mathcal{H}_d' \mathbf{c}$ , which should result in more long-ranged correlations. Just like for the unsmoothed case this is computationally feasible in practice. For the Wish data studied in Section 3.2 we did not observe a significant impact of moderate monitor smoothing on the final variances or correlation matrices. We conjecture that the effect of smoothing may be of secondary importance for powder diffraction – after the coordinate transformation from wavelength to  $d$ -spacing, every monitor bin contributes to many  $d$ -spacing bins, even without smoothing. It is not clear if the same would also hold true, e.g., for single-crystal data. In this case only few detector pixels and wavelengths may contribute to a sharp Bragg peak. Thus, without smoothing each monitor bin might contribute only to a single peak, but smoothing could introduce correlations to other peaks.

## 4. Discussion

### 4.1. Current software: Scope and impact of incorrect uncertainty handling

We have described and demonstrated a systematic problem with the treatment of the statistical uncertainties of normalizations terms in Mantid and Scipp. This constitutes the most important result of this work.

Affected software includes Mantid (all versions at the time of writing, at least up to Mantid v6.5) and Scipp (all versions up to Scipp v22.11, inclusively) but the issue may also be present in other similar data-reduction software. Mantid is used at the ISIS Neutron and Muon Source at the STFC Rutherford Appleton Laboratory (<http://www.isis.stfc.ac.uk/>), the Spallation Neutron Source (SNS) and the High Flux Isotope Reactor (HFIR) at the Oak Ridge National Laboratory (<http://neutrons.ornl.gov/>), the Institut Laue-Langevin (ILL, <https://www.ill.eu/>), the China Spallation Neutron Source (CSNS, <http://english.ihep.cas.cn/csns/>), the European Spallation Source ERIC (ESS, <http://europespsallationsource.se/>), the Forschungs-Neutronenquelle Heinz Maier-Leibnitz (FRM II, <https://www.frm2.tum.de/en/frm2/home/>), and SINQ at the Paul Scherrer Institute (<https://www.psi.ch/en/sinq>). Scipp is used at the ESS. All experimental techniques, including powder-diffraction, small-angle scattering, reflectometry, single-crystal diffraction, spectroscopy, and imaging can be affected. Many data-reduction workflows are affected multiple times. For example, in powder diffraction we may perform normalization of sample data to monitor (problem from broadcast over pixels and event-mode), normalization of vanadium data to monitor (problem from broadcast over pixels and event-mode), normalization of reduced data to vanadium (problem from event-mode). Even this relatively simple example includes five distinct instances of the problem.

To our knowledge little or no impact of the issue described in this manuscript has been reported in practice. There are at least two likely explanations. Firstly, variations of Eq. (13) may hold in many cases. That is, in practice the normalization terms often have very good statistics – as we have seen for the concrete SANS and powder diffraction examples in Section 3.1 and Section 3.2, respectively. This can be both due to practicality (it is relatively simple

to measure many vanadium counts, or have a monitor with a high count rate) and due experience (instrument scientists know how long to measure normalization terms for good results). Secondly, coordinate transformations, as well as preprocessing steps of the normalization terms such as smoothing, lead to correlations that spread over a wider range. These correlations might affect mostly non-local aspects, which could be harder to discern when, e.g., fitting a physical model to the reduced result.<sup>20</sup>

In our opinion the above can however not justify dismissal of the issue. The current state-of-the-art software gives a false sense of security (of having handled uncertainties correctly) to the scientist. While we managed to compute corrected analytical results for variances and covariances in the model workflows used in this manuscript, this cannot be directly substituted in CS implementations for the operation-by-operation error-propagation mechanism. At the very least our software must thus avoid giving a false sense of security and should therefore reject normalization terms that have uncertainties. This will force the scientist to consider whether the normalization uncertainties fulfill Eq. (13). Based on this they can then either (A) confidently drop this term or (B) increase the statistics by measuring the normalization terms better.

A complete analysis for any particular technique such as SANS or powder-diffraction is beyond the scope of this manuscript. The workflows discussed in Section 3 are therefore not necessarily capturing all potential sources of uncertainties, but are meant to illustrate the key aspects. To give an idea of what a more complete analysis might involve, consider the SANS case. Our analysis in Section 3.1 covered the effects of the variances of the monitors. There are however a number of other contributions: (1) Before computation of wavelengths and  $Q$ , the beam center is determined. The uncertainty on the beam center leads to a correlation of all computed wavelength and  $Q$  values. (2) For the computation of the transmission fraction, each monitor is background-subtracted: The monitor background is computed as the mean over a user-defined wavelength range and then subtracted from the monitor. This broadcasts the mean value to all monitor bins, leading to correlations. (3) The direct beam function  $D$  itself comes with uncertainties which were computed using CS and may thus be underestimated in themselves.

Intriguingly, the SANS data reduction implemented in Scipp prior to this work (and used for this analysis) actually ignored the uncertainties of  $D$ . This provides a worrisome glimpse of the potentially cascading negative effects of the CS implementation: An interpolation is required to map  $D$  to the wavelength-binning of the monitors, and it was unclear how uncertainties should be handled during this interpolation. During development of the workflow it was determined empirically that setting  $D$ 's uncertainties to zero had no influence on the uncertainties of the final result. Looking back, we can now tell that this empiric evidence was based on the incorrect handling of uncertainties – since the broadcast of the pixel-independent  $D$  to all pixels suppresses its contribution to the final uncertainties. Thus the decision to ignore them needs to be revisited. Preliminary results indicate that correct handling of the uncertainties of  $D$  may noticeably impact the final uncertainties.

#### 4.2. Improved analytical approach

Aside from highlighting the systematic problem, we have succeeded in Section 3 in deriving expressions for variance-covariance matrices of the data-reduction results. As we will show in the discussion in Section 4.3, there is likely no major difference in terms of computational cost compared to the flawed approach currently implemented in Mantid and Scipp, even though it provides access to not just the variances but also the covariances of the final result. We conjecture that our approach may be feasible also for complete and realistic data-reduction workflows. We plan to investigate this in more detail in future work, in particular to extend this to other neutron-scattering techniques and to include corrections such as absorption corrections and background subtractions.

---

<sup>20</sup>We do not have enough background knowledge about concrete subsequent data analysis methods to tell when and where such non-local properties might be irrelevant.

A fundamental difference – and possibly the main challenge – is that unlike in the CS implementation, handling and computation of variances is difficult to automate. While Python packages such as JAX [4] and Uncertainties [16] implement automatic differentiation approaches, it is unlikely that this is directly applicable for the components of a neutron-scattering data-reduction workflow.<sup>21</sup> That is, we would need to either rely on fixed workflows with previously derived expressions for the variance-covariance matrix, or the user must derive these on their own. If a user does derive such an expression, programming skills are required to implement it. In either case, workflows that handle error propagation may become less interactive.

On the positive side, our new approach can produce data-reduction results including covariances. This may be beneficial for subsequent analysis and model fitting procedures. Furthermore, Scipp has proven flexible enough to allow for computation of the expressions for the variance-covariance matrix with very concise code and adequate performance. This further increases our confidence that the described approach may be a viable solution for handling data from day-to-day instrument operations.

### 4.3. Computational cost of the improved analytical approach

In Sections 3.1 and 3.2 we described an improved analytical approach for computing uncertainties and correlations for the result of the data-reduction. The computation of the variance-covariance matrices turns out to be not significantly more expensive than the computation of the cross sections. Despite yielding the full correlation matrix the overall cost is thus comparable to that of the CS implementation. This holds true even for event-mode data-reduction. To understand this, one important aspect to realize is that – in contrast to CS – it is no longer necessary to carry the uncertainties of the detector counts in intermediate steps, since for the raw input data the variances are equal to the measured counts. A full analysis of the performance characteristics is beyond the scope of this paper. We briefly touch on the main aspects and proceed to report concrete timings for an example below.

Firstly, considering the normalization operation we observe the following. In addition to the normalized counts,  $x/y$ , CS computes the variance as

$$\text{var}\left(\frac{x}{y}\right) = \frac{\text{var}(x) + \text{var}(y)\frac{x^2}{y^2}}{y^2}. \quad (43)$$

This is reordered compared to Eq. (4) to avoid infinity in limit cases such as  $x = 0$ . These operations for the variances are omitted for the improved approach. While this saves a lot of floating-point operations, in practice the normalization is typically bound by memory bandwidth on modern computer architectures. Cutting the required memory in half by not storing variances at this point thus has likely a more significant impact. Secondly, there are in principle savings in both floating-point operations as well as memory bandwidth from not having to handle variances during the histogramming step. In practice however, histogramming is strongly influenced by CPU cache effects and various operations with non-linear scaling in either the number of events or the number of bins. Details depend heavily on the concrete implementation. See the excellent Ref. [8] for more background on CPU architecture, memory, and their influence on performance. These saving are counterbalanced by additional computations for, e.g., Eq. (35). This involves computation of the  $\mathcal{H}_d c$  factors, which are obtained by computing a 2-D  $(d, \lambda)$ -histogram. The sum over the product of resulting factors then corresponds to a medium-sized matrix-matrix product, which takes a couple of seconds for realistic wavelength and  $d$ -spacing bin counts.<sup>22</sup>

We illustrate this on the example of the Wish powder-diffraction data-reduction. The input data has a file size of 930 MByte with 194560 detector pixels. In Table 1 we report timings for the relevant part of data-reduction: normalization, computation of  $d$ -spacing, and reduction (histogramming) using the multi-threaded Scipp implementation running on all 24 hardware threads of the test system. Times for loading data and computation of  $\lambda$  are

<sup>21</sup>Even if automatic differentiation could handle operations such as histogramming, the Uncertainties Python package stores arrays with uncertainties as arrays of values with uncertainties. The very high memory and compute overhead makes this unsuitable for our needs.

<sup>22</sup>Timings measured for NumPy [10] backed by a multi-threaded linear-algebra library.



Table 1

Timings for the relevant part of the Wish histogram-mode data-reduction.  $N_\lambda$  is the number of wavelength bins and  $N_d$  the number of  $d$ -spacing bins. Both columns reports timings for a Scipp-based implementation, but the mechanism for propagating uncertainties is used only for the CS column. Measured on an Intel Xeon W-2265, 19.25 MByte cache, 3.50 GHz, 12 cores (24 threads). Reported values are rounded and averaged between three runs

$N_\lambda, N_d$	CS [s]	ours [s]
1000	1.8	2.3
2000	3.3	4.5
4000	6.3	10.9

not included. We observe that our improved approach – which includes computation of the variance-covariance matrix – is less than twice as costly as the CS approach, taking between 2.3 s and 10.9 s depending on the choice of parameters. The additional cost ranges between 0.5 s and 4.6 s. For comparison, note that loading a Wish raw file using Mantid takes about 12 s on the same machine. That is, the overall impact on data-reduction performance is likely minor. The impact could be reduced further by optimizing the code. For example, Scipp does currently not have a direct 2-D histogramming algorithm but instead used a less efficient two-step approach.

## 5. Conclusion

In this paper we have shown a systematic problem with the treatment of the statistical uncertainties of normalizations terms in Mantid and Scipp. We have shared our findings with the Mantid development team. In the Scipp project, we have disabled propagation of uncertainties if one of the operands is broadcast in an operation with the release of Scipp v23.01. Support for event-mode normalization operations with normalization terms that have variances is also disabled. In both cases an exception is raised instead, i.e., the operation will fail, forcing the user to address the problem correctly. We had initially considered simply displaying a warning to the user when an operation introduces correlations. However, experience shows that warnings are almost always ignored and we now do not consider such an approach as a solution.

Aside from highlighting the systematic problem, we have succeeded in deriving expressions for variance-covariance matrices of the data-reduction results. Computationally the new approach would be feasible, but whether it is applicable and useful in practice remains to be investigated.

## Acknowledgements

We kindly thank the authors of Ref. [17] for allowing us to use their data for the SANS example. We thank Pascal Manuel for sharing Wish data, used in our powder-diffraction example. We thank Evan Berkowitz and Thomas Luu for valuable discussions about bootstrap and correlations. We thank our coworkers Celine Durniak, Torben Nielsen, and Wojciech Potrzebowski for helpful discussions and comments that helped to improve the clarity of the manuscript.

## Appendix A. Bootstrap

This section gives a brief overview of bootstrapping as used in this work. For a more complete explanation we refer to Ref. [9]. Given a measured distribution of random variables  $X$ , we aim to compute some parameter  $\theta$ . We estimate  $\theta$  from a sample  $X$  using an estimator  $s$ ,  $\hat{\theta} = s(X)$ .



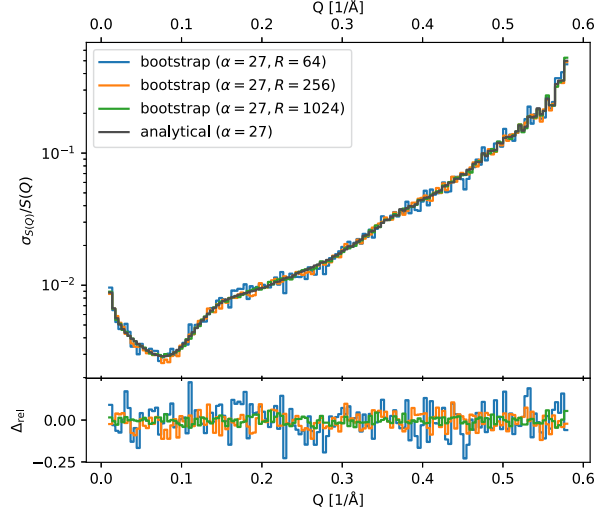


Fig. 8. Convergence of bootstrap to the analytical result for the relative standard deviation for SANS  $S(Q)$ . The bottom panels shows the relative difference between each bootstrap result and the analytical result.  $R$  is the number of bootstrap replicas.

In this work,  $\mathbf{X}$  typically corresponds to a neutron count rate per pixel, per wavelength bin, as well as other input parameters. That is,  $\mathbf{X}$  includes, e.g., both measurements at detectors  $D$  and monitors  $M$  such that, e.g.,  $\mathbf{X} \equiv [X_{i_1}, \dots, X_{i_n}] = [D_{\lambda_1 p_1}, \dots, D_{\lambda_m p_l}, M_{\lambda_1}, \dots, M_{\lambda_m}]$ .  $D$  is indexed by wavelength  $\lambda$  and pixel  $p$  while  $M$  is only indexed by the wavelength.  $\theta$  can be the scattering intensity as a function of  $d$ -spacing or  $Q$ . The estimator  $s$  corresponds to the data-reduction procedure that yields the intensity for that  $d$  or  $Q$ .

We assume all  $X_i$  to be independently normally distributed<sup>23</sup> with mean  $\mu_i$  and standard deviation  $\sigma_i$ , i.e.  $X_i = \mathcal{N}(\mu_i, \sigma_i)$ . Bootstrapping proceeds by sampling from  $X_i$  such that a single bootstrap sample is

$$\mathbf{x}^* = [x_{i_1}^*, \dots, x_{i_n}^*], \quad \text{where } x_i^* \sim X_i = \mathcal{N}(\mu_i, \sigma_i). \quad (44)$$

The desired parameter is then computed on this sample, yielding a bootstrap replica  $\hat{\theta}^* = s(\mathbf{x}^*)$ . Finally, the average and variance of  $\hat{\theta}$  are computed from  $R$  replicas as

$$\mu(\hat{\theta}^*) = \frac{1}{R} \sum_{r=1}^R s(\mathbf{x}_r^*), \quad (45)$$

$$\text{var}(\hat{\theta}^*) = \frac{1}{R-1} \sum_{r=1}^R (s(\mathbf{x}_r^*) - \mu(\hat{\theta}^*))^2. \quad (46)$$

In addition, the covariance of two parameters  $\text{cov}(\theta, \eta)$  can similarly be estimated as

$$\text{cov}(\hat{\theta}^*, \hat{\eta}^*) = \frac{1}{R-1} \sum_{r=1}^R (s_\theta(\mathbf{x}_r^*) - \mu(\hat{\theta}^*))(s_\eta(\mathbf{x}_r^*) - \mu(\hat{\eta}^*)). \quad (47)$$

As an example, Figs 8 and 9 show the relative standard deviation of  $S(Q)$  (SANS) and  $S(d)$  (powder diffraction) computed by bootstrap with different values of  $R$ . The figures demonstrate that the bootstrap calculations

<sup>23</sup>This assumption also underlies the regular propagation of uncertainties.

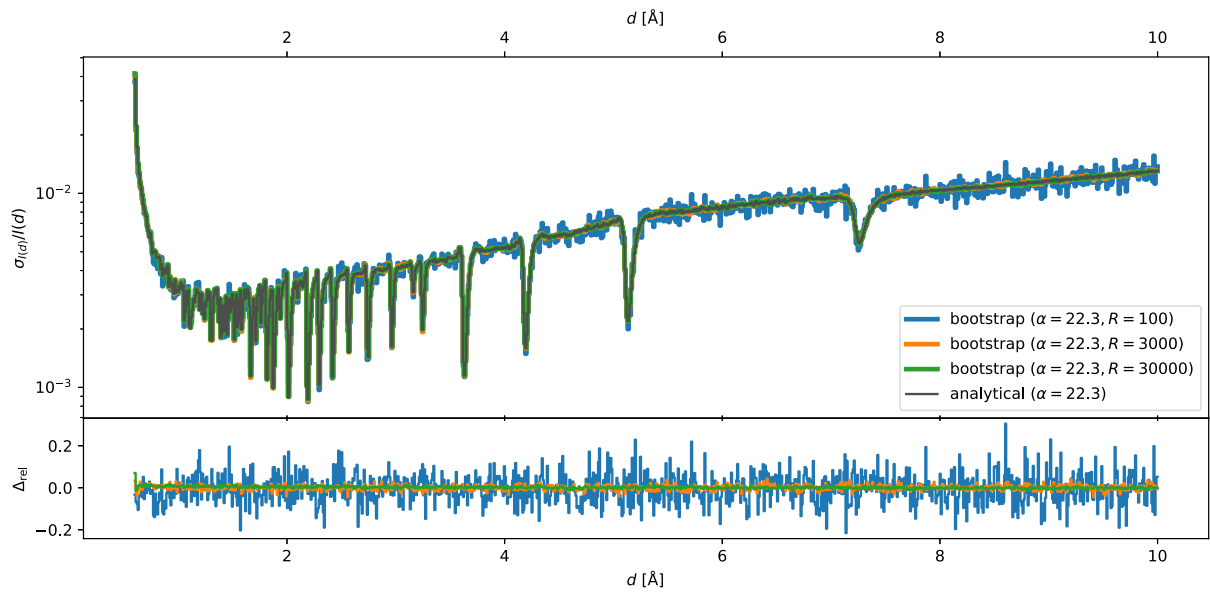


Fig. 9. Convergence of bootstrap to the analytical result for the relative standard deviation for powder diffraction  $S(d)$ . The bottom panels shows the relative difference between each bootstrap result and the analytical result.  $R$  is the number of bootstrap replicas.

converge to the analytical result. Our bootstrap code is available as a Python package at <https://scipp.github.io/scippuncertainty/>.

## Appendix B. Figures for correlation matrices

For completeness, we show in this section the figures of the correlation matrices from the SANS and powder diffraction workflows using a logarithmic color scale.

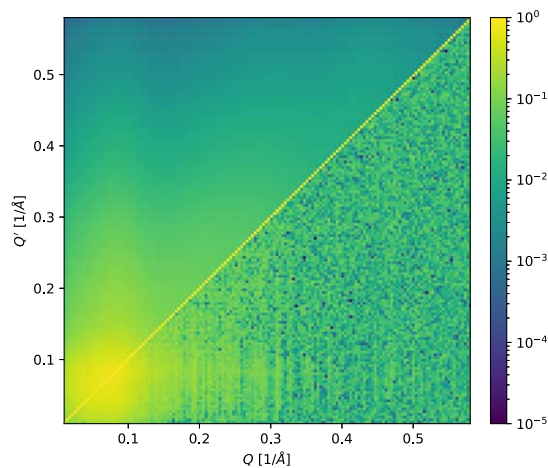


Fig. 10. Correlation matrices for the time-of-flight SANS-reduction workflow using  $\alpha = 100 \cdot \alpha_{\text{actual}} = 27$ . As the correlation matrix is symmetric under  $Q \leftrightarrow Q'$ , we can show in the same figure the analytical result (above the diagonal) and the bootstrap result (below the diagonal,  $R = 1024$  replicas). Analog of Fig. 4 with a logarithmic color scale.

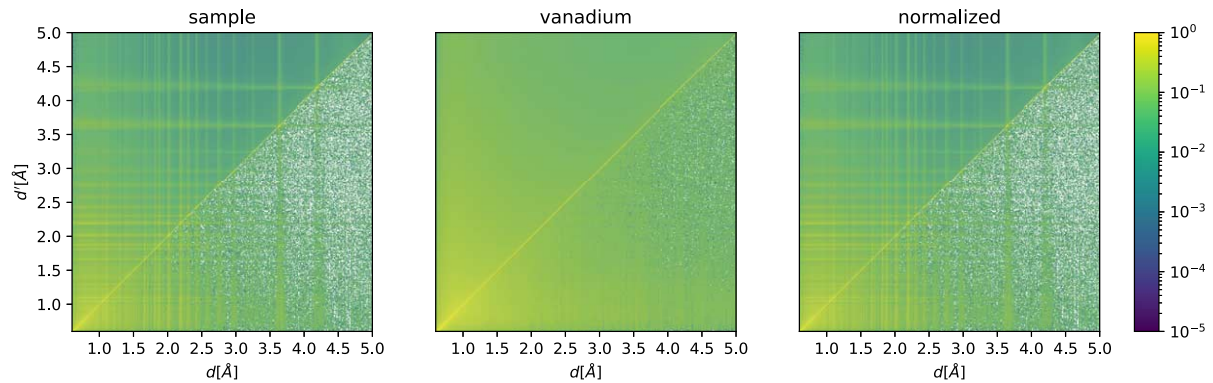


Fig. 11. Correlation matrices for a powder-reduction workflow. We show the result for the sample  $S$  (left,  $\alpha = 10 \cdot \alpha_{\text{actual}}^{\text{sample}} = 22.3$ ), vanadium  $V$  (center,  $\alpha = 10 \cdot \alpha_{\text{actual}}^{\text{vanadium}} = 47.0$ ), and the normalized  $P = S/V$  (right). As the correlation matrices  $\Sigma^S$ ,  $\Sigma^V$ , and  $\Sigma^P$  are symmetric under  $d \leftrightarrow d'$ , we can show in the same figure the analytical result (above the diagonal) and the bootstrap result (below the diagonal). For improved readability, the plots have a limited  $d$ -spacing range as there are few further features for higher  $d$ . Analog of Fig. 7 with a logarithmic color scale.

## References

- [1] O. Arnold, J.C. Bilheux, J.M. Borreguero, A. Buts, S.I. Campbell, L. Chapon, M. Doucet, N. Draper, R.F. Leal, M.A. Gigg, V.E. Lynch, A. Markvardsen, D.J. Mikkelson, R.L. Mikkelson, R. Miller, K. Palmen, P. Parker, G. Passos, T.G. Perring, P.F. Peterson, S. Ren, M.A. Reuter, A.T. Savici, J.W. Taylor, R.J. Taylor, R. Tolchenov, W. Zhou and J. Zikovsky, Mantid – data analysis and visualization package for neutron scattering and  $\mu$ SR experiments, *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* **764** (2014), 156–166. doi:[10.1016/j.nima.2014.07.029](https://doi.org/10.1016/j.nima.2014.07.029).
- [2] B.A. Berg, *Introduction to Markov Chain Monte Carlo Simulations and Their Statistical Analysis*, 2004. doi:[10.48550/ARXIV.COND-MAT/0410490](https://doi.org/10.48550/ARXIV.COND-MAT/0410490).
- [3] P.R. Bevington and D.K. Robinson, *Data Reduction and Error Analysis for the Physical Sciences*, McGraw-Hill, Boston, 2003.
- [4] J. Bradbury, R. Frostig, P. Hawkins, M.J. Johnson, C. Leary, D. Maclaurin, G. Necula, A. Paszke, J. VanderPlas, S. Wanderman-Milne and Q. Zhang, JAX: Composable transformations of Python+NumPy programs, 2018, <http://github.com/google/jax>.
- [5] Broadcasting in NumPy, <https://numpy.org/doc/stable/user/basics.broadcasting.html>.
- [6] L. Chapon, P. Manuel, P. Radaelli, C. Benson, L. Perrott, S. Ansell, N. Rhodes, D. Raspino, D.M. Duxbury, E. Spill and J. Norris, Wish: The new powder and single crystal magnetic diffractometer on the second target station, *Neutron News* **22** (2011), 22–25. doi:[10.1080/10448632.2011.569650](https://doi.org/10.1080/10448632.2011.569650).
- [7] Documentation of error propagation in Mantid, 2022, <https://docs.mantidproject.org/nightly/concepts/ErrorPropagation.html>.
- [8] U. Drepper, What Every Programmer Should Know About Memory, 2007, <https://people.freebsd.org/~lstewart/articles/cpumemory.pdf>.
- [9] B. Efron and R.J. Tibshirani, *An Introduction to the Bootstrap*, Chapman and Hall/CRC, 1994. doi:[10.1201/9780429246593](https://doi.org/10.1201/9780429246593).
- [10] C.R. Harris, K.J. Millman, S.J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N.J. Smith, R. Kern, M. Picus, S. Hoyer, M.H. van Kerkwijk, M. Brett, A. Haldane, J.F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke and T.E. Oliphant, Array programming with NumPy, *Nature* **585**(7825) (2020), 357–362. doi:[10.1038/s41586-020-2649-2](https://doi.org/10.1038/s41586-020-2649-2).
- [11] R.K. Heenan, J. Penfold and S.M. King, SANS at pulsed neutron sources: Present and future prospects, *Journal of Applied Crystallography* **30**(6) (1997), 1140–1147. doi:[10.1107/S0021889897002173](https://doi.org/10.1107/S0021889897002173).
- [12] S. Heybrock, O. Arnold, I. Gudich, D. Nixon and N. Vaytet, Scipp: Scientific data handling with labeled multi-dimensional arrays for C++ and Python, *Journal of Neutron Research* **22**(2–3) (2020), 169–181. doi:[10.3233/JNR-190131](https://doi.org/10.3233/JNR-190131).
- [13] S. Heybrock, N. Vaytet and J.-L. Wynen, Systematic underestimation of uncertainties by widespread neutron-scattering data-reduction software – Supplementary Material, Zenodo, 2023. doi:[10.5281/zenodo.7708115](https://doi.org/10.5281/zenodo.7708115).
- [14] R. Hjelm, The resolution of TOF low-Q diffractometers: Instrumental, data acquisition and reduction factors, *Journal of Applied Crystallography* **21** (1988), 618–628. doi:[10.1107/S0021889888005229](https://doi.org/10.1107/S0021889888005229).
- [15] ISIS Powder Diffraction Scripts Workflow Diagrams, <https://docs.mantidproject.org/v6.5.0/techniques/ISISPowder-Workflows-v1.html>.
- [16] O.E. Lebigot, Uncertainties: A Python package for calculations with uncertainties, 2023, <http://pythonhosted.org/uncertainties/>.
- [17] I. Manasi, M.R. Andalibi, R.S. Atri, J. Hooton, S.E.M. King and K.J. Edler, Self-assembly of ionic and non-ionic surfactants in type IV cerium nitrate and urea based deep eutectic solvent, *The Journal of Chemical Physics* **155**(8) (2021), 084902. doi:[10.1063/5.0059238](https://doi.org/10.1063/5.0059238).

- [18] C.E. Papadopoulos and H. Yeung, Uncertainty estimation and Monte Carlo simulation method, *Flow Measurement and Instrumentation* **12**(4) (2001), 291–298. doi:[10.1016/S0955-5986\(01\)00015-2](https://doi.org/10.1016/S0955-5986(01)00015-2).
- [19] P.F. Peterson, S.I. Campbell, M.A. Reuter, R.J. Taylor and J. Zikovsky, Event-based processing of neutron scattering data, *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* **803** (2015), 24–28. doi:[10.1016/j.nima.2015.09.016](https://doi.org/10.1016/j.nima.2015.09.016).
- [20] P.F. Peterson, D. Olds, A.T. Savici and W. Zhou, Advances in utilizing event based data structures for neutron scattering experiments, *Review of Scientific Instruments* **89**(9) (2018), 093001. doi:[10.1063/1.5034782](https://doi.org/10.1063/1.5034782).
- [21] Sans2d: I(Q) workflow for a single run (sample), [https://scipp.github.io/ess/instruments/loki/sans2d\\_to\\_I\\_of\\_Q.html](https://scipp.github.io/ess/instruments/loki/sans2d_to_I_of_Q.html).
- [22] Sans2d: Time-of-flight Small-Angle Neutron Scattering instrument, <https://www.isis.stfc.ac.uk/Pages/Sans2d.aspx>.
- [23] Scipp Documentation, 2022, <https://scipp.github.io/>.
- [24] SNSPowderReduction algorithm documentation, <https://docs.mantidproject.org/v6.5.0/algorithms/SNSPowderReduction-v1.html>.