SANDIA REPORT

SAND2015-0891
Unlimited Release
Printed February 2015

Formulas for robust, parallel computation of arbitrary-order, arbitrary-variate, statistical moments with arbitrary weights and compounding

Philippe Pébay, Timothy Terriberry, Hemanth Kolla, and Janine Bennett

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Sandia National Laboratories

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

Printed in the United States of America. This report has been reproduced directly from the best available copy.

Available to DOE and DOE contractors from
U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone:	(865) 576-8401
Facsimile:	(865) 576-5728
E-Mail:	reports@adonis.osti.gov
Online ordering:	http://www.osti.gov/bridge

Available to the public from
U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Rd
Springfield, VA 22161

Telephone:	(800) 553-6847
Facsimile:	(703) 605-6900
E-Mail:	orders@ntis.fedworld.gov
Online ordering:	http://www.ntis.gov/help/ordermethods.asp?loc=7-4-0#online

SAND2015-0891
Unlimited Release
Printed February 2015

Formulas for robust, parallel computation of arbitrary-order, arbitrary-variate, statistical moments with arbitrary weights and compounding

Philippe Pébay
Sandia National Laboratories
M.S. 9159, P.O. Box 969
Livermore, CA 94551, U.S.A.
pppebay@sandia.gov

Timothy Terriberry
The Xiph.Org Foundation
2521 S. Oxford St.
Arlington, VA 22206
tterribe@xiph.org

Hemanth Kolla
Sandia National Laboratories
M.S. 9158, P.O. Box 969
Livermore, CA 94551, U.S.A.
hnkolla@sandia.gov

Janine Bennett
Sandia National Laboratories
M.S. 9152, P.O. Box 969
Livermore, CA 94551, U.S.A.
jcbenne@sandia.gov

Abstract

Formulas for incremental or parallel computation of second order central moments have long been known, and recent extensions of these formulas to univariate and multivariate moments of aribtrary order have been developed. Such formulas are of key importance in scenarios where incremental results are required and in parallel and distributed systems where communication costs are high. We survey these recent results, and recall the first generalizations which we had obtained in [P08]. We then improve these arbitrary-order, numerically stable one-pass formulas to arbitrary-variate formulas which we further extend to arbitrary weights and compound variants. We also develop a generalized correction factor for standard two-pass algorithms that enables the maintenance of accuracy over nearly the full representable range of the input, avoiding the need for extended-precision arithmetic.

This page intentionally left blank

Contents

This page intentionally left blank

1 Introduction

Central moments, including the variance, and derived quantities like skewness and kurtosis, are some of the most widely used tools in descriptive statistics. However, standard approaches for computing them, reviewed in Section 2, either require two passes over the data, or are grossly inaccurate for data that is not contained within a very limited range. This poses a problem in streaming settings where incremental results are needed after each new value is observed, and for very large datasets, which may not fit in available memory, and increasingly are distributed over a number of hosts. The prevalence of large, distributed data sets has lead to the recent development of new statistical packages to analyze them [WBS08, WTP$^+$08, BGP$^+$09, PTB10, PTBM11, Edd10, SME$^+$09, Sta10]. In this setting the cost of distributed memory access is so large that two-pass algorithms become entirely impractical. Even a single machine increasingly performs large parallel computations on a Graphics Processing Unit (GPU), where memory bandwidth is a significant bottleneck. Using two passes doubles the execution time, and using double precision arithmetic doubles it again, almost irrespective of the number of arithmetic operations performed in each pass.

For the second central moment (the variance), accurate, one-pass, incremental approaches have long been known [Wel62, Nee66, Wes79]. Chan et al. generalized them into a "pairwise algorithm" [CGL79], which computes the variance of a set by partitioning it into two subsets, computing their second order statistics recursively, and then combining them with an updating rule to obtain the second order statistics of the whole set. Constraining the second set to be a singleton yields an efficient incremental (on-line) algorithm. Alternatively, using subsets of roughly equal size yields a highly parallel algorithm. The latter also ensures intermediate terms will be commensurate, increasing accuracy by preventing destructive underflow. Incremental formulas for cumulants up to fourth order have been proposed for a zero-mean process [AB95, DF98]. Ensuring a zero-mean process involves removing a mean estimated from the data [DF98], which traditionally requires a two-pass algorithm, eliminating the benefit of a recursive update formula.

In [BGP$^+$09] Chan et al.'s variance calculation approach was generalized to moments of arbitrary order and formulas for incremental and pairwise algorithms were provided. These formulas are particularly useful as a number of applications of higher order moments require on-line updates or parallel processing. For instance, many communications applications use both univariate [Men91] and multivariate [NM93] moments up to fourth order—or cumulants, which are frequently computed from the central moments. These include blind deconvolution [SW90], blind source separation [Tug97], direction finding [PF91], and speech detection [NGM01], all of which can benefit from on-line updates to adapt to changing channel conditions and minimize delay. Image processing also makes frequent use of higher-order moments for modeling non-linear distortions, with applications in deblurring [XC96, IHMM98, WHS06], noise removal [KLB97], gamma correction and radial distortion estimation [FP01], and steganalysis [LF02]. Skewness and kurtosis are also commonly used in financial modeling [Sam70, HS00], where datasets are so large that distributed processing is required. Although examples are less common, moments up to sixth order can aid chromatic dispersion compensation in long distance fiber-optic lines [KHSS05], and eighth-order moments provide a means to identify cell phone modulation schemes [PM08], to name a few. In this paper we further expand these formulas to a variety of other extensions: weighted moments, forgetting schemes, and compound moments. Compound moments have important applications for turbulent flow analysis [Jon93] and we demonstrate their application in this setting.

As the order of the moment increases, even the venerable two-pass algorithm may be inaccurate, as the numerical error for evaluating polynomials around the mean grows exponentially with the degree. When communication costs are the bottleneck, doubling the working precision doubles the computation time. Alternatives, such as compensation algorithms for summation [ORO05] and polynomial evaluation [LL07], require twice as much storage for intermediate values. This is not an issue when computations are performed locally, but for distributed computations this is just as costly as doubling the working precision. A well-known correction factor, attributed to Åke Björk [CGL83] though also proposed by Neely [Nee66], greatly improves the accuracy of the two-pass algorithm when computing the variance. In this paper we generalize this correction factor to moments of arbitrary order. Our scheme transmits only one additional value in the second pass, but can correct for the error in moments of all orders, providing increased accuracy for higher order moments at a fraction of the cost of generic compensation schemes.

2 Background: Computing Statistical Moments

We begin with a brief notational preamble, after which we directly formulate the main difficulties which arise when computing statistical moments using floating point representations.

2.1 Statement of the Problem

For p a non-negative integer and using $E[\cdot]$ to denote the expectation, the p-th central moment of a (univariate, real) random variable X is defined as

$$\mu_p \triangleq E\left[(X - E[X])^p\right] , \tag{2.1}$$

when the expectations exist[1]. For a finite population of n equiprobable values in a multiset $S = \{x_i\}_{i=1}^n$, this reduces to

$$\mu_p = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^p \tag{2.2}$$

where

$$\bar{x} \triangleq \frac{1}{n}\sum_{i=1}^{n} x_i \tag{2.3}$$

is the mean. The first central moment is exactly zero, and the second central moment is the *variance*, $\sigma^2 \triangleq \mu_2$. For this paper, we only consider the statistics of finite populations taken in their entirety, i.e., not sampled, to avoid issues of estimation bias. If S is instead just a finite sample of an infinite population, one may obtain unbiased estimates of the moments of the whole population [Hal46][2]. However, unbiased estimates of the moments do not, in general, lead to unbiased estimates of the derived quantities, such as standard deviation, skweness, and kurtosis.

The standard *two-pass algorithm* explicitly computes μ_p using (2.3) followed by (2.2). The two-pass algorithm is numerically stable even when $\bar{x}$ is large and μ_p is small. Its stability for $p = 2$ can be further improved by applying a well-known correction factor [Nee66, CGL83]:

$$\mu_2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2 - {\mu_1}^2 \tag{2.4}$$

$$= \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2 - \left(\frac{1}{n}\sum_{i=1}^{n} x_i - \bar{x}\right)^2 . \tag{2.5}$$

---

[1] Some random variables, such as those with a Cauchy distribution, do not have an expectation.

[2] For $p = 2$ and $p = 3$, the unbiased estimators are $\frac{n}{n-1}\mu_2$ and $\frac{n^2}{(n-1)(n-2)}\mu_3$, respectively, with μ_p computed over the sample as in (2.2). However, for $p = 4$ the unbiased estimator is [DR99]

$$\frac{(n-1)(n^2-3n+3)}{n^3}\mu_4 + \frac{3(2n-3)(n-1)}{n^3}{\mu_2}^2 .$$

Samples drawn (without replacement) from a finite population require additional corrections.

By definition, $\mu_1 = 0$ when evaluated with exact arithmetic, but Chan et al. [CL78] show that when computing both μ_1 and μ_2 with inexact arithmetic using the two-pass algorithm, the rounding error introduced into μ_1 cancels much of the rounding error introduced into μ_2. The *corrected two-pass algorithm* still only requires two passes, since μ_1 and μ_2 can be computed simultaneously.

However, the two-pass approach is inadequate for large or distributed data sets, where making two complete passes through the data is extremely expensive. It is also unsuitable when one needs a new estimate of μ_p each time a new x value is obtained. The obvious method of obtaining a one-pass calculation, what Chan et al. call the *textbook algorithm* [CGL83] for the variance, is to expand the product $(x-\bar{x})^p$ into explicit powers of x and $\bar{x}$. Using the binomial theorem, this is easy to generalize to arbitrary order:

$$\mu_p = \sum_{k=0}^{p} \binom{p}{k} \left(\frac{1}{n} \sum_{i=1}^{n} x_i^{p-k} \right) \left(-\bar{x} \right)^k . \tag{2.6}$$

The inner sums, including that for $\bar{x}$, can be updated incrementally or computed in parallel, and the outer sum requires negligible additional work, since p is typically small. However, even for $p = 2$, this expression quickly becomes grossly inaccurate. The alternating signs on each term cause destructive cancellation, and few, if any, significant digits are retained. The results may even be negative when p is even, which is clearly nonsensical for it violates the Cauchy-Schwarz inequality.

Example 2.1. Consider the values $x_1 = 1$ and $x_2 = x_3 = x_4 = 1 + 10^{-13}$. Their respective double-precision (64 bits) IEEE-754 floating point hexadecimal representations [iee85] are

$$r(x_1) = \texttt{3ff0000000000000}$$

and

$$r(x_2) = r(x_3) = r(x_4) = \texttt{3ff00000000001c2} .$$

Subsequently, one obtains the following representations for the mean:

$$r(\mu) = r\left(\frac{1}{4}\right)\left(\sum_{i=1}^{4} r(x_i)\right) = \texttt{3ff0000000000152} ,$$

and the mean of the squares:

$$r\left(\frac{1}{4}\right)\left(\sum_{i=1}^{4} r(x_i)^2\right) = \texttt{3ff00000000002a3} .$$

Thus the textbook algorithm yields the following value for the variance:

$$r\left(\frac{1}{4}\right)\left(\sum_{i=1}^{4} r(x_i)^2\right) - r(\mu)^2 = \texttt{bcb0000000000000} ,$$

which represents a negative number:

$$r(-2.220446049250313 \times 10^{-16}) = \texttt{bcb0000000000000}$$

thus establishing that the textbook algorithm can yield negative variances even with small data sets. This problem is therefore not limited to large statistical calculations, but it becomes potentially worse as the size of the sample set increases.

2.2 Numerically stable one-pass algorithms

Much better one-pass algorithms for computing the variance have long been known [Wel62, Nee66, Wes79, CGL79]. Chan et al. summarize them using a generic set of recurrence formulas [CGL79]. Partition S into multisets $\mathscr{A}$ and $\mathscr{B}$ of size $n_{\mathscr{A}}$ and $n_{\mathscr{B}}$ and define $\mu_{p,\mathscr{A}}$, $\mu_{p,\mathscr{B}}$, $\bar{x}_{\mathscr{A}}$, and $\bar{x}_{\mathscr{B}}$ to be the corresponding statistic computed over each partition. Then let

$$\delta_{\mathscr{B},\mathscr{A}} \triangleq \bar{x}_{\mathscr{B}} - \bar{x}_{\mathscr{A}} \ , \tag{2.7}$$

$$M_p \triangleq n\mu_p \ , \tag{2.8}$$

and again give $M_p^{\mathscr{A}}$ and $M_p^{\mathscr{B}}$ an equivalent definition restricted to each partition. We will find it more convenient to work with these M_p quantities, rather than μ_p, though either may be readily obtained from the other. Now

$$\bar{x} = \bar{x}_{\mathscr{A}} + \frac{n_{\mathscr{B}}}{n}\delta_{\mathscr{B},\mathscr{A}} \ , \tag{2.9}$$

$$M_2 = M_2^{\mathscr{A}} + M_2^{\mathscr{B}} + \frac{n_{\mathscr{A}} n_{\mathscr{B}}}{n}\delta_{\mathscr{B},\mathscr{A}}^2 \ . \tag{2.10}$$

A number of algorithms can be derived from these simple recurrences. Letting $n_{\mathscr{B}} = 1$ so that $\mathscr{B} = \{y\}$ a singleton yields the incremental update formulas of [Wes79]:

$$\bar{x} = \bar{x}_{\mathscr{A}} + \frac{y - \bar{x}_{\mathscr{A}}}{n} \ , \tag{2.11}$$

$$M_2 = M_2^{\mathscr{A}} + \frac{n-1}{n}(y - \bar{x}_{\mathscr{A}})^2 \ . \tag{2.12}$$

On the other hand, letting $n_{\mathscr{A}} = n_{\mathscr{B}} = n/2$ (assuming n is even) gives a recursive pairwise algorithm:

$$\bar{x} = \bar{x}_{\mathscr{A}} + \frac{1}{2}\delta_{\mathscr{B},\mathscr{A}} \ , \tag{2.13}$$

$$M_2 = M_2^{\mathscr{A}} + M_2^{\mathscr{B}} + \frac{n}{4}\delta_{\mathscr{B},\mathscr{A}}^2 \ . \tag{2.14}$$

While easily parallelizable, the pairwise algorithm can also reduce destructive underflow on a uniprocessor, since it ensures that the terms in the update formulas are approximately commensurate when the data is i.i.d. By contrast, when n is large, the terms corresponding to the $\mathscr{B}$ partition in (2.11) and (2.12) are very small, affecting only a few of the least significant digits of $\bar{x}$ and M_2. Both algorithms perform the same number of updates, but the pairwise algorithm requires $O(\log n)$ additional storage. The same pairwise strategy can be applied with similar benefits to *all* summation formulas, including (2.2), (2.3), and (2.6).

3 Arbitrary-Order Update Formulas

3.1 Univariate Formulas

We begin by generalizing all pairwise and update formulas to arbitrary order and arbitrary set decomposition.

Proposition 3.1. *For any integer $p \geq 2$,*

$$\begin{aligned} M_p = M_p^{\mathscr{A}} + M_p^{\mathscr{B}} + n_{\mathscr{A}} \left(\frac{-n_{\mathscr{B}}}{n} \delta_{\mathscr{B},\mathscr{A}} \right)^p + n_{\mathscr{B}} \left(\frac{n_{\mathscr{A}}}{n} \delta_{\mathscr{B},\mathscr{A}} \right)^p \\ + \sum_{k=1}^{p-2} \binom{p}{k} \delta_{\mathscr{B},\mathscr{A}}^k \left[M_{p-k}^{\mathscr{A}} \left(\frac{-n_{\mathscr{B}}}{n} \right)^k + M_{p-k}^{\mathscr{B}} \left(\frac{n_{\mathscr{A}}}{n} \right)^k \right]. \end{aligned} \tag{3.1}$$

Proof. By the definition of M_p, and because $\{\mathscr{A}, \mathscr{B}\}$ is a partition of S, one has

$$\begin{aligned} M_p &= \sum_{i=1}^{n} (x_i - \bar{x})^p && (3.2) \\ &= \sum_{i=1}^{n_{\mathscr{A}}} (x_i - \bar{x})^p + \sum_{i=n_{\mathscr{A}}+1}^{n} (x_i - \bar{x})^p && (3.3) \\ &= \sum_{i=1}^{n_{\mathscr{A}}} \left(x_i - \frac{n_{\mathscr{A}} \bar{x}_{\mathscr{A}} + n_{\mathscr{B}} \bar{x}_{\mathscr{B}}}{n} \right)^p + \sum_{i=n_{\mathscr{A}}+1}^{n} \left(x_i - \frac{n_{\mathscr{A}} \bar{x}_{\mathscr{A}} + n_{\mathscr{B}} \bar{x}_{\mathscr{B}}}{n} \right)^p && (3.4) \\ &= \sum_{i=1}^{n_{\mathscr{A}}} \left(x_i - \bar{x}_{\mathscr{A}} - \frac{n_{\mathscr{B}}}{n} \delta_{\mathscr{B},\mathscr{A}} \right)^p + \sum_{i=n_{\mathscr{A}}+1}^{n} \left(x_i - \bar{x}_{\mathscr{B}} + \frac{n_{\mathscr{A}}}{n} \delta_{\mathscr{B},\mathscr{A}} \right)^p && (3.5) \\ &= \sum_{k=0}^{p} \binom{p}{k} \left(\frac{\delta_{\mathscr{B},\mathscr{A}}}{n} \right)^k \left[(-1)^k M_{p-k}^{\mathscr{A}} n_{\mathscr{B}}^k + M_{p-k}^{\mathscr{B}} n_{\mathscr{A}}^k \right], && (3.6) \end{aligned}$$

thanks to the commutativity of summation over finite sets, which allows us to swap $\sum_{k=0}^{p}$ with $\sum_{i=1}^{n_{\mathscr{A}}}$ and $\sum_{i=n_{\mathscr{A}}+1}^{n}$. Now, a few simplifications are in order: first, the $k = 0$ term of the above summation is simply $M_p^{\mathscr{A}} + M_p^{\mathscr{B}}$; second, by definition, both $M_1^{\mathscr{A}}$ and $M_1^{\mathscr{B}}$ are zero, eliminating the $k = p-1$ term; last, $M_0^{\mathscr{A}} = n_{\mathscr{A}}$ and $M_0^{\mathscr{B}} = n_{\mathscr{B}}$, eliminating the need to compute these values separately for use in the $k = p$ term. Applying these three simplifications to (3.5) yields (3.1). □

The final result M_p requires $M_q^{\mathscr{A}}$ and $M_q^{\mathscr{B}}$ for each $q \in \{2 \dots p\}$, instead of just for $q = p$. Thus the update formula performs $O(p^2)$ arithmetic operations per element, compared to the $O(1)$ operations the two-pass algorithm required if only M_p is actually needed. A small improvement in accuracy may be obtained by evaluating (3.1) as the sum of two polynomials in $\frac{-n_{\mathscr{B}}}{n}\delta_{\mathscr{B},\mathscr{A}}$ and $\frac{n_{\mathscr{A}}}{n}\delta_{\mathscr{B},\mathscr{A}}$ using Horner's rule. When local computations are cheap, one could even use the compensated Horner scheme, which often provides exactly rounded results [LL07], but this does not prevent the accumulation of rounding errors in recursive applications of (3.1).

Corollary 3.2. *In the case where $\mathscr{B}$ is reduced to a singleton $\{y\}$, Proposition 3.1 reduces to the incremental update formula for $S = \mathscr{A} \cup \{y\}$ as follows*

$$\begin{aligned} M_p = M_p^{\mathscr{A}} + \left[\frac{n-1}{(-n)^p} + \left(\frac{n-1}{n}\right)^p\right]\delta_{\mathscr{B},\mathscr{A}}^p \\ + \sum_{k=1}^{p-2}\binom{p}{k} M_{p-k}^{\mathscr{A}}\left(\frac{-\delta_{\mathscr{B},\mathscr{A}}}{n}\right)^k. \end{aligned} \tag{3.7}$$

Proof. Corollary 3.2 is an immediate specialization of Proposition 3.1 obtained when $n_{\mathscr{A}} = n-1$ and $n_{\mathscr{B}} = 1$. In this case, each $M_p^{\mathscr{B}}$ vanishes since $\bar{x}_{\mathscr{B}} = y$, and thus (3.1) immediately simplifies to (3.7). □

Remark 3.1. By noticing that

$$\frac{n-1}{(-n)^2} + \left(\frac{n-1}{n}\right)^2 = \frac{n^2-n}{n^2} = \frac{n-1}{n} \tag{3.8}$$

and taking $p = 2$, one directly retrieves (2.12) from Corollary 3.2.

We provide implementations of univariate incremental and pairwise update formulas in the open-source Visualization Tool Kit (VTK), respectively in the `Learn()` and `Aggregate()` methods of the dexcriptive statistics class `vtkDescriptiveStatistics`.

3.2 Multivariate Formulas

We continue by generalizing the univariate results to arbitrary multivariate moments (co-moments). These are of interest, in particular, for Pearson correlation analysis, which we wish to conduct on large-scale, distributed data sets. Higher order co-moments such as co-skewness and co-kurtosis also have financial modeling applications [HSX04].

Extending the notation of previous sections, let $S = \{x_i\}_{i=1}^n \subset \mathbb{R}^d$, where $x_i = (x_{i,1}, \ldots, x_{i,d}) \in \mathbb{R}^d$ is a d-dimensional vector. Now let $\alpha = (\alpha_1, \ldots, \alpha_d)$ and $\beta = (\beta_1, \ldots, \beta_d)$ be *multi-indices* of non-negative integers so that

$$\alpha \le \beta \iff \alpha_j \le \beta_j \; \forall j \in \{1 \ldots d\} \,, \tag{3.9}$$

$$|\alpha| \triangleq \sum_{j=1}^{d} \alpha_j \,, \tag{3.10}$$

$$\binom{\alpha}{\beta} \triangleq \prod_{j=1}^{d} \binom{\alpha_j}{\beta_j} \,, \tag{3.11}$$

$$x_i^{\alpha} \triangleq \prod_{j=1}^{d} x_{i,j}^{\alpha_j} \,, \tag{3.12}$$

and define the central co-moment of order α of a finite population $S = \{x_i\}$ to be

$$\mu_\alpha \triangleq \frac{1}{n} \sum_{i=1}^{n} (x_i - \bar{x})^{\alpha} \,. \tag{3.13}$$

Under this definition the usual covariance is obtained with $\alpha = (1,1)$. All the terms $\bar{x}$, $\bar{x}_{\mathscr{A}}$, $\bar{x}_{\mathscr{B}}$, $\delta_{\mathscr{B},\mathscr{A}}$, $\mu_{\alpha,\mathscr{A}}$, $\mu_{\alpha,\mathscr{B}}$, M_α, $M_\alpha^{\mathscr{A}}$, $M_\alpha^{\mathscr{B}}$ are defined exactly as in the univariate case, with α replacing the univariate order p. Similarly, we define

$$M_\alpha \triangleq \sum_{i=1}^{n} (x_i - \bar{x})^{\alpha} \tag{3.14}$$

for the multi-index α.

Proposition 3.3. *The recursive update formula for M_α is:*

$$M_\alpha = \sum_{\beta \le \alpha} \binom{\alpha}{\beta} \delta_{\mathscr{B},\mathscr{A}}^{\beta} \left[\left(-\frac{n_{\mathscr{B}}}{n}\right)^{|\beta|} M_{\alpha-\beta}^{\mathscr{A}} + \left(\frac{n_{\mathscr{A}}}{n}\right)^{|\beta|} M_{\alpha-\beta}^{\mathscr{B}} \right]. \tag{3.15}$$

Proof. Following the proof of Proposition 3.1,

$$\begin{aligned} M_\alpha = &\sum_{i=1}^{n_{\mathscr{A}}} \left(x_i - \bar{x}_{\mathscr{A}} - \frac{n_{\mathscr{B}}}{n} \delta_{\mathscr{B},\mathscr{A}} \right)^{\alpha} \\ &+ \sum_{i=n_{\mathscr{A}}+1}^{n} \left(x_i - \bar{x}_{\mathscr{B}} + \frac{n_{\mathscr{A}}}{n} \delta_{\mathscr{B},\mathscr{A}} \right)^{\alpha} \,. \end{aligned} \tag{3.16}$$

Expanding out the multi-index products and applying the binomial theorem yields

$$\begin{aligned} M_\alpha = &\sum_{i=1}^{n_{\mathscr{A}}} \prod_{j=1}^{d} \sum_{k=0}^{\alpha_j} \binom{\alpha_j}{k} \left(x_{i,j} - \bar{x}_{j,\mathscr{A}}\right)^{\alpha_j - k} \left(-\frac{n_{\mathscr{B}}}{n} \delta_{\mathscr{B},\mathscr{A}\,j}\right)^{k} \\ &+ \sum_{i=n_{\mathscr{A}}+1}^{n} \prod_{j=1}^{d} \sum_{k=0}^{\alpha_j} \binom{\alpha_j}{k} \left(x_{i,j} - \bar{x}_{j,\mathscr{B}}\right)^{\alpha_j - k} \left(\frac{n_{\mathscr{A}}}{n} \delta_{\mathscr{B},\mathscr{A}\,j}\right)^{k}. \end{aligned} \tag{3.17}$$

Distributing the inner sums over the products, this simplifies to

$$M_\alpha = \sum_{i=1}^{n_\mathscr{A}} \sum_{\beta \le \alpha} \binom{\alpha}{\beta} \left(x_i - \bar{x}_\mathscr{A}\right)^{\alpha-\beta} \left(-\frac{n_\mathscr{B}}{n}\delta_{\mathscr{B},\mathscr{A}}\right)^\beta + \sum_{i=n_\mathscr{A}+1}^{n} \sum_{\beta \le \alpha} \binom{\alpha}{\beta} \left(x_i - \bar{x}_\mathscr{B}\right)^{\alpha-\beta} \left(\frac{n_\mathscr{A}}{n}\delta_{\mathscr{B},\mathscr{A}}\right)^\beta . \tag{3.18}$$

Once again, the commutativity of summation over finite sets allows us to swap $\sum_{\beta\le\alpha}$ with $\sum_{i=1}^{n_\mathscr{A}}$ and $\sum_{i=n_\mathscr{A}+1}^{n}$, and rearranging terms produces (3.15). □

A quick check verifies that (3.15) reduces to (3.5) when $d = 1$ and $\alpha = (p)$. The same simplifications made in the univariate case still apply when actually computing the full (3.15), though they do not simplify the notation. That is, the $\beta = \mathbf{0}$ term is simply $M_\alpha^\mathscr{A} + M_\alpha^\mathscr{B}$, and every term where $\exists j \in \{1\dots d\} \ni \alpha_j - \beta_j = 1$ vanishes. Similarly, when $\beta = \alpha$, $M_\mathbf{0}^\mathscr{A} = n_\mathscr{A}$ and $M_\mathbf{0}^\mathscr{B} = n_\mathscr{B}$. Applying these simplifications to (3.15) with $\alpha = (1,1)$ produces

$$M_{(1,1)} = M_{(1,1)}^\mathscr{A} + M_{(1,1)}^\mathscr{B} + \left[n_\mathscr{A}\left(-\frac{n_\mathscr{B}}{n}\right)^2 + n_\mathscr{B}\left(\frac{n_\mathscr{A}}{n}\right)^2\right]\delta_{\mathscr{B},\mathscr{A}}^{(1,1)} , \tag{3.19}$$

$$= M_{(1,1)}^\mathscr{A} + M_{(1,1)}^\mathscr{B} + \frac{n_\mathscr{A} n_\mathscr{B}}{n}(\bar{x}_{1,\mathscr{B}} - \bar{x}_{1,\mathscr{A}})(\bar{x}_{2,\mathscr{B}} - \bar{x}_{2,\mathscr{A}}) . \tag{3.20}$$

When $\mathscr{B}$ is reduced to a singleton $\{(y_1, y_2)\}$, this is equivalent to the incremental covariance update formula derived by Neely [Nee66]:

$$M_{(1,1)} = M_{(1,1)}^\mathscr{A} + M_{(1,1)}^\mathscr{B} + \frac{n-1}{n}(y_1 - \bar{x}_{1,\mathscr{A}})(y_2 - \bar{x}_{2,\mathscr{A}}) . \tag{3.21}$$

We provide implementations of bivariate incremental and pairwise update formulas in the open-source Visualization Tool Kit (VTK), respectively in the `Learn()` and `Aggregate()` methods of the correlative statistics `vtkCorrelativeStatistics` class.

3.3 Weighted Formulas

Consider a quantity $\widetilde{x}_W$ defined as a weighted arithmetic mean with respect to a given set of weights W. Replacing the set sizes n, $n_\mathscr{A}$, and $n_\mathscr{B}$ with sums of non-negative weights $\{w_i\}_{1\le i\le N}$,

$$W \triangleq \sum_{i=1}^{n} w_i\,,\;\; W_\mathscr{A} \triangleq \sum_{i=1}^{n_\mathscr{A}} w_i\,,\;\; W_\mathscr{B} \triangleq \sum_{i=n_\mathscr{A}+1}^{n} w_i\,, \tag{3.22}$$

and the other sums with weighted sums,

$$\widetilde{x}_W \triangleq \frac{1}{W}\sum_{i=1}^{n} w_i x_i \ , \tag{3.23}$$

$$M_p \triangleq \sum_{i=1}^{n} w_i (x_i - \widetilde{x}_W)^p \ , \tag{3.24}$$

leads directly to weighted versions of our main results:

$$\widetilde{x}_W = \widetilde{x}_{W,\mathscr{A}} + \frac{W_\mathscr{B}}{W}\delta_{\mathscr{B},\mathscr{A}} \ , \tag{3.25}$$

$$\begin{aligned} M_p = M_p^\mathscr{A} + M_p^\mathscr{B} + W_\mathscr{A}\Big(\frac{-W_\mathscr{B}}{W}\delta_{\mathscr{B},\mathscr{A}}\Big)^p + W_\mathscr{B}\Big(\frac{W_\mathscr{A}}{W}\delta_{\mathscr{B},\mathscr{A}}\Big)^p \\ + \sum_{k=1}^{p-2}\binom{p}{k}\delta_{\mathscr{B},\mathscr{A}}^k\left[M_{p-k}^\mathscr{A}\Big(\frac{-W_\mathscr{B}}{W}\Big)^k + M_{p-k}^\mathscr{B}\Big(\frac{W_\mathscr{A}}{W}\Big)^k\right], \end{aligned} \tag{3.26}$$

$$M_\alpha = \sum_{\beta\le\alpha}\binom{\alpha}{\beta}\delta_{\mathscr{B},\mathscr{A}}^\beta\left[M_{\alpha-\beta}^\mathscr{A}\Big(-\frac{W_\mathscr{B}}{W}\Big)^{|\beta|} + M_{\alpha-\beta}^\mathscr{B}\Big(\frac{W_\mathscr{A}}{W}\Big)^{|\beta|}\right]. \tag{3.27}$$

These formulas may be used to derive adaptive estimators for non-stationary signals by setting $W_\mathscr{B} = \frac{1}{\eta}W_\mathscr{A}$, where $\frac{1}{\eta}$ with $\eta > 0$ is a *forgetting factor*, similar to that proposed by Demélé and Favier [DF98]. This holds the relative importance of the most recent sample constant, while that of past samples decays exponentially. Other adaptive schemes are possible.

Remark 3.2. In addition, the proofs of Propositions 3.1 and 3.3 remain equally valid if the sums over S are replaced with integrals, since the other sums are finite. Thus, one can use (3.26) and (3.27) to compute moments of mixture distributions given the moments of each independent mixture element. Such moments can indicate goodness of fit or even be used to estimate the mixture parameters themselves via the method of moments [Pea94].

3.4 Formulas for Compound Moments

A special case is that of compound moments of the type

$$\mu_p = \frac{1}{n}\sum_{i=1}^{n}(x_i - \widetilde{x}_W)^p \ , \tag{3.28}$$

where $\widetilde{x}_W$ is the weighted mean defined by (3.23). Such compound moments are often used in moment-closure modelling methods of turbulent flows [Jon93]. Correspondingly, we define the quantity $\widetilde{M}_p$ by replacing $\bar{x}$ with $\widetilde{x}_W$ in (3.2) and, when (3.25) is applied, it expands to

$$\widetilde{M}_p = \sum_{i=1}^{n_{\mathscr{A}}} (x_i - \widetilde{x}_W)^p + \sum_{i=n_{\mathscr{A}}+1}^{n} (x_i - \widetilde{x}_W)^p \tag{3.29}$$

$$= \sum_{i=1}^{n_{\mathscr{A}}} \left(x_i - \widetilde{x}_{W,\mathscr{A}} - \frac{W_{\mathscr{B}}}{W} \delta_{\mathscr{B},\mathscr{A}} \right)^p + \sum_{i=n_{\mathscr{A}}+1}^{n} \left(x_i - \widetilde{x}_{W,\mathscr{B}} + \frac{W_{\mathscr{A}}}{W} \delta_{\mathscr{B},\mathscr{A}} \right)^p \tag{3.30}$$

$$= \sum_{k=0}^{p} \binom{p}{k} \delta_{\mathscr{B},\mathscr{A}}^k \left[\widetilde{M}_{p-k}^{\mathscr{A}} \left(\frac{-W_{\mathscr{B}}}{W} \right)^k + \widetilde{M}_{p-k}^{\mathscr{B}} \left(\frac{W_{\mathscr{A}}}{W} \right)^k \right] \tag{3.31}$$

where $\widetilde{\delta}_{\mathscr{B},\mathscr{A}}$ is the compound counterpart of $\delta_{\mathscr{B},\mathscr{A}}$, defined as

$$\widetilde{\delta}_{\mathscr{B},\mathscr{A}} \triangleq \widetilde{x}_{W,\mathscr{B}} - \widetilde{x}_{W,\mathscr{A}} \ . \tag{3.32}$$

The $k = 0$ term simplifies to $\widetilde{M}_p^{\mathscr{A}} + \widetilde{M}_p^{\mathscr{B}}$, while the $k = p$ term assumes the slightly different form of

$$n_{\mathscr{A}} \left(-\frac{W_{\mathscr{B}}}{W} \widetilde{\delta}_{\mathscr{B},\mathscr{A}} \right)^p + n_{\mathscr{B}} \left(\frac{W_{\mathscr{A}}}{W} \widetilde{\delta}_{\mathscr{B},\mathscr{A}} \right)^p, \tag{3.33}$$

since $\widetilde{M}_0^{\mathscr{A}} = n_{\mathscr{A}}$ and $\widetilde{M}_0^{\mathscr{B}} = n_{\mathscr{B}}$. However, the $k = p-1$ term is non-zero and expands to

$$p\widetilde{M}_1^{\mathscr{A}} \left(-\frac{W_{\mathscr{B}}}{W} \widetilde{\delta}_{\mathscr{B},\mathscr{A}} \right)^{p-1} + p\widetilde{M}_1^{\mathscr{B}} \left(\frac{W_{\mathscr{A}}}{W} \widetilde{\delta}_{\mathscr{B},\mathscr{A}} \right)^{p-1}, \tag{3.34}$$

where

$$\widetilde{M}_1^{\mathscr{A}} = n_{\mathscr{A}} \left(\bar{x}_{\mathscr{A}} - \widetilde{x}_{W,\mathscr{A}} \right) \tag{3.35}$$

and

$$\widetilde{M}_1^{\mathscr{B}} = n_{\mathscr{B}} \left(\bar{x}_{\mathscr{B}} - \widetilde{x}_{W,\mathscr{B}} \right). \tag{3.36}$$

The resulting expansion is thus

$$\begin{aligned} \widetilde{M}_p = {} & \widetilde{M}_p^{\mathscr{A}} + \widetilde{M}_p^{\mathscr{B}} \\ & + p\widetilde{M}_1^{\mathscr{A}} \left(\frac{-W_{\mathscr{B}}}{W} \widetilde{\delta}_{\mathscr{B},\mathscr{A}} \right)^{p-1} + p\widetilde{M}_1^{\mathscr{B}} \left(\frac{W_{\mathscr{A}}}{W} \widetilde{\delta}_{\mathscr{B},\mathscr{A}} \right)^{p-1} \\ & + n_{\mathscr{A}} \left(\frac{-W_{\mathscr{B}}}{W} \widetilde{\delta}_{\mathscr{B},\mathscr{A}} \right)^p + n_{\mathscr{B}} \left(\frac{W_{\mathscr{A}}}{W} \widetilde{\delta}_{\mathscr{B},\mathscr{A}} \right)^p \\ & + \sum_{k=1}^{p-2} \binom{p}{k} \widetilde{\delta}_{\mathscr{B},\mathscr{A}}^k \left[\widetilde{M}_{p-k}^{\mathscr{A}} \left(\frac{-W_{\mathscr{B}}}{W} \right)^k + \widetilde{M}_{p-k}^{\mathscr{B}} \left(\frac{W_{\mathscr{A}}}{W} \right)^k \right] \end{aligned} \tag{3.37}$$

which can be slightly simplified as follows

$$
\begin{aligned}
\widetilde{M}_p = \widetilde{M}_p^{\mathscr{A}} + \widetilde{M}_p^{\mathscr{B}} \\
&+ \left(p\widetilde{M}_1^{\mathscr{A}} - n_{\mathscr{A}} \frac{W_{\mathscr{B}}}{W} \widetilde{\delta}_{\mathscr{B},\mathscr{A}}\right)\left(\frac{-W_{\mathscr{B}}}{W}\widetilde{\delta}_{\mathscr{B},\mathscr{A}}\right)^{p-1} \\
&+ \left(p\widetilde{M}_1^{\mathscr{B}} + n_{\mathscr{B}} \frac{W_{\mathscr{A}}}{W} \widetilde{\delta}_{\mathscr{B},\mathscr{A}}\right)\left(\frac{-W_{\mathscr{A}}}{W}\widetilde{\delta}_{\mathscr{B},\mathscr{A}}\right)^{p-1} \\
&+ \sum_{k=1}^{p-2}\binom{p}{k}\widetilde{\delta}_{\mathscr{B},\mathscr{A}}^{k}\left[\widetilde{M}_{p-k}^{\mathscr{A}}\left(\frac{-W_{\mathscr{B}}}{W}\right)^{k} + \widetilde{M}_{p-k}^{\mathscr{B}}\left(\frac{W_{\mathscr{A}}}{W}\right)^{k}\right].
\end{aligned}
\tag{3.38}
$$

3.5 Two-pass Correction

Chan et al. note that the correction factor for $p = 2$ in (2.5) is equivalent to computing a *trial mean* with (2.3), shifting the data by this value, and then applying the textbook algorithm [CGL83].

Applying the same strategy to the arbitrary-order formula (2.6) yields

$$
\mu_p = \sum_{k=0}^{p}\binom{p}{k}\left(\sum_{i=1}^{n}(x_i - \bar{x})^{p-k}\right)\left(-\frac{1}{n}\sum_{i=1}^{n}x_i - \bar{x}\right)^{k}. \tag{3.39}
$$

Since central moments are defined relative to the mean, subtracting a constant from the data does not affect the result, but when that constant is the mean (or a close approximation), it can have a large effect on accuracy. Like the $p = 2$ case, there is no destructive cancellation since the correction terms are much smaller than the $k = 0$ term.

Again, this is a two-pass algorithm. The first pass computes $\bar{x}$, and the second pass computes all of the inner sums in parallel. Compared to using extended-precision arithmetic, it is inexpensive computationally, requiring only two additions per value plus a small, constant amount of work at the end. However, its real advantage over typical compensation schemes is that it can correct for the error in the moments of all orders while only transmitting a single correction term.

References

[AB95] Pierre-Olivier Amblard and Jean-Marc Brossier. Adaptive estimation of the fourth-order cumulant of a white stochastic process. *Signal Processig*, 42(1):37–43, February 1995.

[BGP+09] Janine Bennett, Ray W. Grout, Philippe P. Pébay, Diana C. Roe, and David C. Thompson. Numerically stable, single-pass, parallel statistics algorithms. In *CLUSTER*, pages 1–8. IEEE, 2009.

[CGL79] Tony F. Chan, Gene H. Golub, and Randall J. LeVeque. Updating formulae and a pairwise algorithm for computing sample variances. Technical Report STAN-CS-79-773, Stanford University, Department of Computer Science, November 1979.

[CGL83] Tony F. Chan, Gene H. Golub, and Randall J. LeVeque. Algorithms for computing the sample variance: Analysis and recommendations. *The American Statistician*, 37(3):242–247, August 1983.

[CL78] Tony F. Chan and John G. Lewis. Rounding error analysis of algorithms for computing means and standard deviations. Technical Report Tech. Rep. 289, Dept. of Mathematical Sciences, The Johns Hopkins U., Baltimore, MD, April 1978.

[DF98] Doulaye Dembélé and Gérard Favier. Recursive estimation of fourth-order cumulants with application to identification. *Signal Processing*, 68(2):127–139, July 1998.

[DR99] Yadolah Dodge and Valentin Rousson. The complications of the fourth central moment. *The American Statistician*, 53(3):267–269, August 1999.

[Edd10] Dirk Eddelbuettel. High-performance and parallel computing with R, 2010. Version 2010-09-16 downloaded 2010-11-15.

[FP01] Hany Farid and Alin C. Popescu. Blind removal of image non-linearities. In *Proc. of the 8th International Conference on Computer Vision (ICCV'01)*, volume 1, pages 76–81, Vancouver, BC, July 2001.

[Hal46] Paul R. Halmos. The theory of unbiased estimation. *The Annals of Mathematical Statistics*, 17(1):34–43, March 1946.

[HS00] Campbell R. Harvey and Akhtar Siddique. Conditional skewness in asset pricing tests. *The Journal of Finance*, 55(3):1263–1295, June 2000.

[HSX04] Daniel Chi-Hsiou Hung, Mark Shackleton, and Xinzhong Xu. CAPM, higher co-moment and factor models of UK stock returns. *Journal of Business Finance & Accounting*, 31(1-2):87–112, Jan./Mar. 2004.

[iee85] IEEE standard for binary floating-point arithmetic. *ANSI/IEEE Std 754-1985*, August 1985.

[IHMM98] Mostafa A. Ibrahim, Abdel-Wahab F. Hussein, Samia A. Mashali, and Ahmed H. Mohamed. A blind image restoration system using higher-order statistics and Radon transform. In *Proc. of the 5th International Conference on Electronics, Circuits and Systems (ICECS'98)*, volume 3, pages 523–530, Lisboa, Portugal, September 1998.

[Jon93] W. P. Jones. Turbulence modelling and numerical solution methods for variable density and combusting flows. In P. A. Libby and F. A. Williams, editors, *Turbulent reacting flows*, pages 309–374. Academic Press, London, 1993.

[KHSS05] N. Kikuchi, S. Hayase, K. Sekine, and S. Sasaki. Performance of chromatic dispersion monitoring using statistical moments of asynchronously sampled waveform histograms. *Photonics Technology Letters*, 17:1103–1105, May 2005.

[KLB97] Richard P. Kleihorst, Reginald L. Lagendijk, and Jan Biemond. An adaptive order-statistic noise filter for gamma-corrected image sequences. *IEEE Transactions on Image Processing*, 6(10):1442–1446, October 1997.

[LF02] Siwei Lyu and Hany Farid. Detecting hidden messages using higher-order statistics and support vector machines. In *Proc. of the 5th International Workshop on Information Hiding (IH'02)*, volume 2578 of *Lecture Notes in Computer Science*, pages 340–354, Noordwijkerhout, The Netherlands, October 2002. Springer-Verlag.

[LL07] Philippe Langlois and Nicolas Louvet. How to ensure a faithful polynomial evaluation with the compensated horner algorithm. In *Proc. 18th Symposium on Computer Arithmetic (ARITH'07)*, pages 141–149, Montpellier, France, June 2007.

[Men91] Jerry M. Mendel. Tutorial on higher-order statistics (spectra) in signal processing and system theory: Theoretical results and some applications. *Proceedings of the IEEE*, 79(3):278–305, March 1991.

[Nee66] Peter M. Neely. Comparison of several algorithms for computation of means, standard deviations and correlation coefficients. *Communications of the ACM*, 9(7):496–499, July 1966.

[NGM01] Elias Nemer, Rafik Goubran, and Samy Mahmoud. Robust voice activity detection using higher-order statistics in the LPC residual domain. *IEEE Transactions on Speech and Audio Processing*, 9(3):217–231, March 2001.

[NM93] Chrysostomos L. Nikias and Jerry M. Mendel. Signal processing with higher-order spectra. *IEEE Signal Processing Magazine*, 10(3):10–37, July 1993.

[ORO05] Takeshi Ogita, Siegfried M. Rump, and Shin'ichi Oishi. Accurate sum and dot product. *SIAM Journal on Scientific Computing*, 26(6):1955–1988, November 2005.

[Ṕ08] P. Pébay. Formulas for robust, one-pass parallel computation of covariances and arbitrary-order statistical moments. Sandia Report SAND2008-6212, Sandia National Laboratories, September 2008.

[Pea94] Karl Pearson. Contributions to the mathematical theory of evolution. *Philosophical Transactions of the Royal Society of London, Series A*, 185:71–110, January 1894.

[PF91] Boaz Porat and Benjamin Friedlander. Direction finding algorithms based on high-order statistics. *IEEE Transactions on Signal Processing*, 39(9):2016–2024, September 1991.

[PM08] P. Prakasam and M. Madheswaran. M-ary shift keying modulation scheme identification algorithm using wavelet transform and higher order statistical moment. *Journal of Applied Science*, 8(1):112–119, 2008.

[PTB10] Philippe Pébay, David C. Thompson, and Janine Bennett. Computing contingency statistics in parallel: Design trade-offs and limiting cases. In *Proceedings of the 2010 IEEE International Conference on Cluster Computing (12th CLUSTER'10)*, pages 156–165, Heraklion, Crete, Greece, September 2010. IEEE Computer Society.

[PTBM11] Philippe Pébay, David C. Thompson, Janine Bennett, and Ajith Mascarenhas. Design and performance of a scalable, parallel statistics toolkit. In *Proc. Workshop on Parallel & Distributed Scientific & Engineering Computing (12th PDSEC'11) Workshop Proceedings, 25th IEEE International Symposium on Parallel and Distributed Processing (25th IPDPS'11)*, pages 1475–1484, Anchorage, Alaska, USA, May 2011. IEEE Computer Society.

[Sam70] Paul A. Samuelson. The fundamental approximation theorem of portfolio analysis in terms of means, variances and higher moments. *The Review of Economic Studies*, 37(4):537–542, October 1970.

[SME$^+$09] M. Schmidberger, M. Morgan, D. Eddelbuettel, H. Yu, L. Tierney, and U. Mansmann. State-of-the-art in parallel computing with R. Technical Report 47, Department of Statistics, University of Munich, 2009.

[Sta10] Stata. Stata/MP performance report. Technical report, StataCorp LP, June 2010. version 2.0.0.

[SW90] Ofir Shalvi and Ehud Weinstein. New criteria for blind deconvolution of nonminimum phase systems (channels). *IEEE Transactions on Information Theory*, 36(2):312–321, March 1990.

[Tug97] Jitendra K. Tugnait. Identification and deconvolution of multichannel linear non-Gaussian processes using higher order statistics and inverse filter criteria. *IEEE Transactions on Signal Processing*, 45(3):658–672, March 1997.

[WBS08] B. Wylie, J. Baumes, and T. Shead. Titan informatics toolkit. In *IEEE Visualization Tutorial*, Columbus, OH, October 2008.

[Wel62] B. P. Welford. Note on a method for calculating corrected sums of squares and products. *Technometrics*, 4(3):419–420, 1962.

[Wes79] D. H. D. West. Updating mean and variance estimates: An improved method. *Communications of the ACM*, 22(9):532–535, September 1979.

[WHS06] Hongzhi Wang, Jinming Huo, and Chunpeng Song. A blind image restoration algorithm based on cumulants. In *Proc. of the 8^{th} International Conference on Signal Processing (ICSP'06)*, Beijing, November 2006.

[WTP+08] M. H. Wong, D. C. Thompson, P. Pébay, J. R. Mayo, A. C. Gentile, B. J. Debusschere, and J. M. Brandt. OVIS-2: A robust distributed architecture for scalable RAS. In *Proc. 22^{nd} IEEE International Parallel & Distributed Processing Symposium*, Miami, FL, April 2008.

[XC96] You Xu and Greg Crebbin. Image blur identification by using higher order statistic techniques. In *Proc. of the 3^{rd} International Conference on Image Processing (ICIP'96)*, volume 3, pages 77–80, Lausanne, Switzerland, September 1996.

DISTRIBUTION:

2	MS 9018	Central Technical Files, 8944
1	MS 0899	Technical Library, 9536

Sandia National Laboratories