Safety in numbers: Minimum thresholding, Maximum bounds, and Little White Lies: The case of the mean and standard deviation

Ben Derrick, Lizzie Green, Kristian Kember, Felix Ritchie, Paul White

Corresponding Author Paul White (paul.white@uwe.ac.uk)

Abstract

Reporting the sample mean, sample standard deviation and sample size could in some cases lead to the unique identification of the underpinning sample. The likelihood of this reveal via direct enumeration of the possible search space decreases with increasing sample size and increasing domain size and the degree of obfuscation increases with sample size. An R routine, uwedragon, is presented to assist analysts in evaluating the risk of disclosure and to help publish useful information whilst minimising the degree of risk. The identification of unusually large observations (sample maximum) would also be on interest. The use of "reverse marching observations" is used to place bounds on estimated maximum values.

1. Background and Motivation

The thrust of this work is to determine whether an interested and sufficiently determined person can reverse engineer or glean additional sample information when presented with only sample size, sample mean and sample standard deviation.

In any practical situation, data is inevitably collected to a finite number of decimal places and as such is discrete i.e., in any finite interval there is a finite number of possible distinct data values. Without loss of generality the following is based on integer data as, for instance, data recorded to one decimal place, when multiplied by 10 would give integer values, and so on for data recorded to other degrees of precision. To motivate matters, initial attention will be restricted to highly discretised sample spaces such as a five-point scale with low samples sizes before extending to scales with a greater number of potential outcome values and larger sample sizes and in these instances the focus is on the ability to completely reverse engineer and discover the unknown sample based purely on sample size, sample mean and sample standard deviation. This body of work is considered in Section 3.

In other situations, complete and precise discovery of an entire sample might not be of interest, but discovery or identification of plausible ranges for maximum sample values may be of interest. Exact recovery of an unknown sample minimum or sample maximum might not be possible but a judicious use of statistical assumptions and statistical procedures may narrow a potential range for sample extrema. These considerations are given in Section 4.

In Section 5, we further consider how an analyst may report plausible means and plausible standard deviations which cannot be used to reverse engineer the original sample.

The following work will draw on a small number of known statistical results and some statistical procedures. It is convenient to establish notation, to recap statistical results and theorems to be used, and to give an overview of some concepts which will be used in gaining insight to extract additional information.

2. Notation and Known results for Mean and Standard Deviation

In the following, n will denote sample size, $\bar{x}$ will denote the sample mean, s^2 the sample variance with divisor $(n-1)$, and $s = \sqrt{s^2}$ the sample standard deviation. A population mean,

population variance and population standard deviation will be denoted by μ, σ^2, and σ respectively. The theorems and statistical results considered will be the Chebyshev Inequality, the Selberg Inequality [1], and the Samuelson Inequality [2]. We will also consider sample augmentation using the concept of a marching observation as given by Derrick et al, (2017) [3], sample deflation using a reverse marching observation, and techniques to provide protection against reverse engineering a sample which entail reporting some obscured information referred to as "little white lies". A brief overview of these will be given prior to their application.

2.1 Known Results

The Chebyshev's inequality (Bienaymé–Chebyshev inequality) was established by Irenee Jules Bienayme and Pafnuty Chebyshev between 1850 and 1870 and states that for finite mean μ and finite variance σ^2 the proportion of a distribution outside of the interval $\mu \pm h\,\sigma$ has to be less than $1/h^2$ where h is a positive number. For instance, if $h = \sqrt{2}$, then the Chebyshev inequality states that no more than half of a distribution can lie outside of the interval $(\mu - \sqrt{2}\sigma,\ \mu + \sqrt{2}\sigma)$. Put, another way, this example instance states that at least 50% of a distribution has to lie in the interval $(\mu - \sqrt{2}\sigma,\ \mu + \sqrt{2}\sigma)$.

The Chebyshev inequality is a general inequality and applies to all possible distributions and consequently, due to the general nature of this result, the intervals obtained from its application are, in general, wide so as to encapsulate all possible distributions. The Chebyshev intervals cannot be improved upon without making some distributional assumptions and as a consequence the width of the interval being large might limit practical utility. For arbitrary positive constants α, β, Selberg's inequality states that the proportion of a distribution which lies outside of the interval $[\mu - \alpha,\ \mu + \beta]$ is greater than or equal to either $\alpha^2/(\alpha^2 + \sigma^2)$ if $\alpha(\beta - \alpha) > 2\sigma^2$ or greater than or equal to $(4\alpha\beta - 4\sigma^2)/(\alpha + \beta)^2$ if $2\alpha\beta \geq \sigma^2 \geq \alpha(\beta - \alpha)$ or equal to 0 if $\sigma^2 > \alpha\beta$. The Chebyshev inequality is a special case of Selberg's inequality and is obtained by setting $\alpha = \beta$. Application of the Selberg inequality could potentially narrow the width of intervals compared to the intervals from the Chebyshev inequality.

In the context of sample Z values, Samuelson [2] has shown that a sample Z score is bounded by the value $\sqrt{n}\,s$. Hence, the values in a sample lie within the interval $(\bar{x} - \sqrt{n}\,s;\ \bar{x} + \sqrt{n}\,s)$ with probability 1.

Derrick et al [3], introduced the concept of a marching inserted observation to assess the impact of the inclusion of a new observation on the sensitivity of a statistic. In this method an observation is included in a sample and the statistic is updated; this additional observation is then removed from the sample its value altered and re-included into the sample. The alteration is done is a systematic manner e.g., starting at say 0, increments by 1, through to say 100. The starting point, increment and finish point are context dependent. In these regards, it is useful to know that if a new sample point x_{n+1} is introduced into a sample, then the new sample variance based on $n+1$ observations with n degrees of freedom, say s_{n+1}^2, is given by

$$ns_{n+1}^2 = (n-1)s_n^2 + (x_{n+1} - \bar{x}_{n+1})(x_{n+1} - \bar{x}_n)$$

where s_n^2 is the sample variance for the original n observations based on $(n-1)$ degrees of freedom, $\bar{x}_n$ is the sample mean based on n observations and $\bar{x}_{n+1}$ is the sample mean based on $n+1$ observations. By application of Brahmagupta's quadratic formula, it is readily derived that the inclusion of a new observation x_{n+1} will leave the sample variance unchanged (i.e., $s_{n+1}^2 = s_n^2$) if

$$x_{n+1} = \bar{x}_n \pm \sqrt{\frac{n}{n+1}}\, s$$

That is to say, if the newly introduced observation is greater than $\bar{x}_n + \sqrt{n/(n+1)}\, s$ or is smaller than $\bar{x}_n - \sqrt{n/(n+1)}\, s$ then the new variance and standard deviation will increase compared to the original variance.

In a similar manner, we may consider the conceptual removal of an observation in a systematic manner, referred to as a marching deleted observation. The underpinning mathematical formulae for the sample variance follow the same principles as given above except the removal might be the conceptual removal of an observation which is not in the original data set, and is therefore looking at the sensitivity of the sample variance to the removal of a hypothetical observation.

The above-mentioned inequalities based on the mean and standard deviation/variance will be used to help ascertain the maximum value in a data set when only n, mean and standard deviation are given. The use of marching observations will be shown to help refine lower and upper bounds on an assumption that the sample maximum, M, is further from the mean, that the sample minimum m i.e., assuming $(M - \bar{x}) > (\bar{x} - m)$.

3. Reverse Engineering based on sample size, mean and standard deviation

Often the mean and standard deviation are used as summary statistics for a group or cell in a descriptive table. If a cell size or group size (n) is equal to 2 then a known sample mean, $\bar{x}$, and known sample standard deviation, s , may be re-expressed as the two data points i.e., the two equations for mean and standard deviation may be solved to give the input data and a unique solution is obtained by virtue of having two equations in two unknowns, $(x_1,\ x_2) = \left(\bar{x} - s/\sqrt{2},\ \ \bar{x} + s/\sqrt{2}\right)$.

For the case of $n > 2$ simply knowing the mean and standard deviation is not sufficient to precisely uncover the original data set without using some external information. External information might include that the data under consideration is non-negative count data (i.e. 0, 1, 2, 3, 4, …. , etc). For instance, if n = 3, and the sample mean is 4, and sample standard deviation (divisor of variance being n -1) is equal to 2, and knowledge that the data is non-negative integers, then it is easily verified that the original data must comprise the values 2, 4, 6. In this case knowing the mean, standard deviation, sample size and knowledge of the domain would uniquely identify the underpinning sample.

For developmental purposes consider when the data under consideration has a known restricted scale. For instance, it is not uncommon to see a five-point Likert scale coded 1, 2, 3, 4, 5 being summarised with mean and standard deviation, where the coded values of 1, 2, 3, 4 and 5 are used as scores. By way of example, consider a 5-point Likert Scale for n = 5 with mean = 2.40 and standard deviation = 1.14. With this configuration it is relatively easy to scrutinise the total search space and discover that the underpinning data has to have the values 1, 2, 2, 3, 4; no other combination of the numbers 1, 2, 3, 4, 5 would give the quoted mean and standard deviation; there is a unique identification. Likewise, under the same conditions, for n = 6, a mean of 3.50 and a standard deviation of 1.049 would uniquely identify the raw data as 2, 3, 3, 4, 4, 5. For the same 5-point Likert scale, there is a potential for non-unique solutions being discovered particularly with increasing sample size and the use of rounded means and rounded standard deviations. For instance, for n = 7, a mean of 2.7 and a standard deviation of 1.38, would give two candidate data sets of either {1, 1, 2, 3, 4, 4, 4} or {1, 1, 3, 3, 3, 3, 5}, but no other solution.

For 5-point Likert-like data, Table 1 summarises the total number of possible different sample configurations for sample sizes n= 3, 4, 5, … 10, 11, 12 and the number of these samples which can be uniquely identified through knowing the mean and standard deviation

when reported with full precision, and when mean and SD are reported to two decimal places, and when reported to 1 decimal place. Tables 2, 3, 4, 5, and 6 provide the same information for a 7-point scale, 9-point scale, 10-point scale, 11-point scale and 12-point scale respectively. Note that these tables summarise those situations where there is a unique one-to-one correspondence between $(\bar{x}, s, n)$ and a sample configuration leading to $(\bar{x}, s, n)$ uniquely identifying the sample which gives rise to $(\bar{x},\ s)$.

Tables 1 to Table 6 give the number of unique solutions; Panel A of Figure 1 summarises the percentage of times $(\bar{x}, s, n)$ lead to the unique uncovering of the underlying sample for a scale comprising k = 10, 11, 12.

There are some key points to note from these tables and from Figure 1. Firstly, for any given sample of size n with a domain based on scale comprising k possible distinct outcomes then the total number of possible samples (Column 2 in tables 1 to 6) may be derived from combinatorics and shown to be equal to $(n + k - 1)!/(k!\,(n - 1)!)$.

In general, the number of samples which can be uniquely identified generally increases with sample size (see column 3) but the *percentage* of uniquely identifiable samples out of the total search space decreases with increasing sample size; for sample size of n = 10 and domain space k = 10, it starts to become a hunt for a needle-in-a-haystack and the size of haystack grows exponentially with increasing n and increasing k. Specifically, for n = 10, k = 10, the search space of 92,378 possible samples yields a 1% chance of uniquely identifying the sample given sample size, mean and standard deviation. The search space rises to 167,960 possible samples with n = 11, k = 10, with a 0.7% chance of reverse engineering and uniquely identifying the underpinning sample (see Table 5).

For n = 11, k = 11 (Table 5) the search space comprises some 352,716 potential samples of which 0.36 percent can uniquely be identified by enumerating all possibilities and comparing with the sample mean and sample standard deviation.

It is also evident that the degree of uniqueness diminishes with increased rounding. For instance, for n = 10, k = 11, if mean and standard deviation are rounded to one decimal place then only 0.2% of the total search space, comprising some 184,756 potential samples, would uniquely reveal the underpinning sample.

Table 1 Five-point scale and the number of unique solutions

Sample Size (n)	Number of Possible Samples	Number of unique solutions uncovered		
		No rounding	Rounded to 2dp	Rounded to 1dp
3	35	33	33	33
4	70	56	56	56
5	129	87	79	79
6	210	105	101	101
7	330	131	121	121
8	495	141	141	133
9	715	177	161	135
10	1001	205	181	157
11	1365	223	201	130
12	1820	243	221	149

Table 2 Seven-point scale and the number of unique solutions

Sample Size (n)	Number of Possible Samples	Number of unique solutions uncovered		
		No rounding	Rounded to 2dp	Rounded to 1dp
3	84	76	76	76
4	210	143	143	143
5	462	206	193	193
6	924	246	222	200
7	1716	295	253	203
8	3003	289	289	201
9	5005	405	325	215
10	8008	438	361	202
11	12376	493	397	198
12	18564	533	433	211

Table 3 Nine-point scale and the number of unique solutions

Sample Size (n)	Number of Possible Samples	Number of unique solutions uncovered		
		No rounding	Rounded to 2dp	Rounded to 1dp
3	165	145	145	145
4	495	271	271	271
5	1,287	396	327	286
6	3,003	440	364	279
7	6,435	527	399	306
8	12,870	449	449	284
9	24,310	693	499	270
10	43,758	701	549	275
11	75,582	821	599	246
12	125,970	837	649	261

Table 4 10-point scale and the number of unique solutions

Sample Size (n)	Number of Possible Samples	Number of unique solutions uncovered		
		No rounding	Rounded to 2dp	Rounded to 1dp
3	220	188	188	188
4	715	353	353	343
5	2,002	509	422	346
6	5,005	564	472	332
7	11,440	747	527	310
8	24,310	603	603	344
9	48,620	955	676	310
10	92,378	944	749	338
11	167,960	1134	822	286
12	293,930	1143	895	291

Table 5 11-point scale and the number of unique solutions

Sample Size (n)	Number of Possible Samples	Number of unique solutions uncovered		
		No rounding	Rounded to 2dp	Rounded to 1dp
3	286	238	238	238
4	1,001	443	443	419
5	3,003	592	496	386
6	8,008	530	654	369
7	19,448	830	580	342
8	43,758	794	652	355
9	92,378	1080	722	342
10	184,756	1044	749	363
11	352,716	1263	866	304
12	646,646	1232	938	311

Table 6 12-point scale and the number of unique solutions

Sample Size (n)	Number of Possible Samples	Number of unique solutions uncovered		
		No rounding	Rounded to 2dp	Rounded to 1dp
3	364	300	300	300
4	1,365	562	562	513
5	4,368	724	616	428
6	12,376	822	683	390
7	31,824	1067	756	388
8	75,582	852	852	404
9	167,960	1360	950	390
10	352,716	1390	1048	399
11	705,432	1589	1146	344
12	1,352,078	1579	1244	364

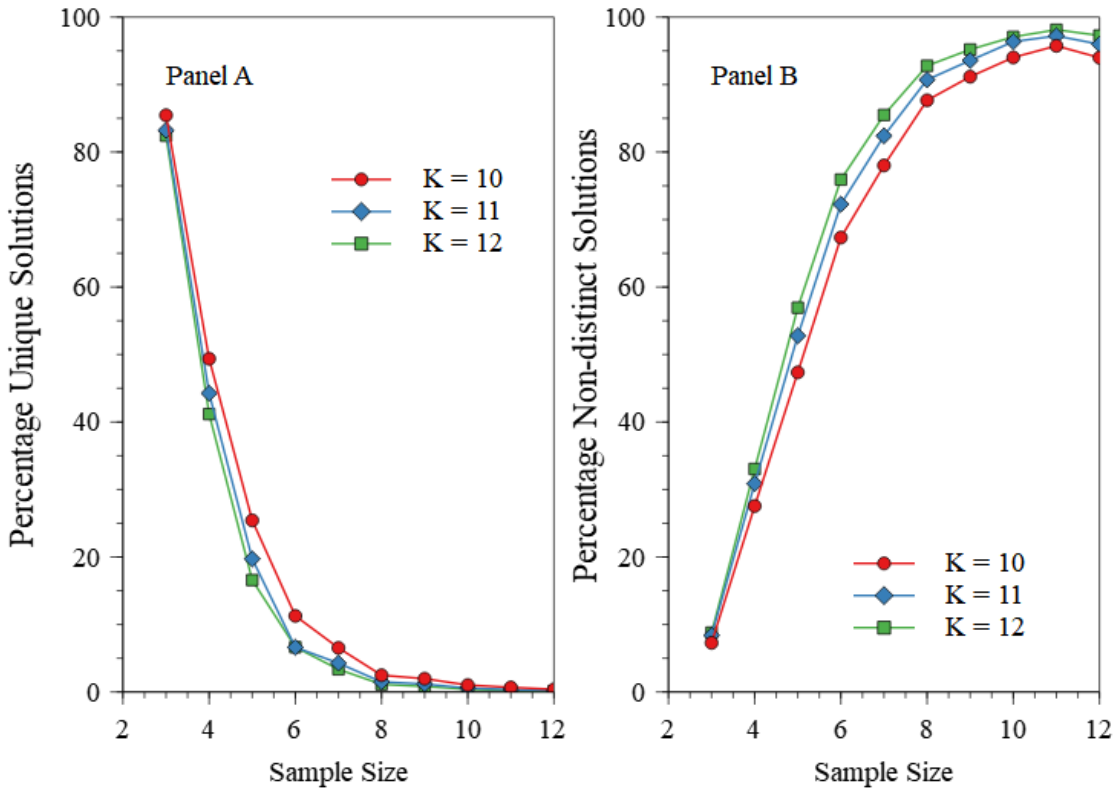

Figure 1. Panel A: Percentage of Samples which can be uniquely identified for a given $(\bar{x}, s, n)$ for domain size k = 9, 10, 11, 12. Panel B Percentage of non-distinct solutions

Figure 1, Panel A, summarises the percentage of the sample space which, for n = 3, 4, 5, 6, 7, 8, 9, 10, 11, 12 crossed with k = 10, 11, 12, permits the unique identification of the underpinning sample by exhaustively searching the underpinning sample space using full precision. When $n \geq 10$, coupled with $k \geq 10$, the sample space is large and the percentage of times the true underpinning sample can be discovered is less than 1%. Panel B gives a plot of the percentage of the sample space with non-distinct ordered pairs of $(\bar{x},\ s)$ as per Table 8 and Figure 2. Table 8 and Figure 2 are based purely on $k = 10$ and record the number of times $(\bar{x},\ s)$ occur once in the total sample space (i.e. uniquely identify the underpinning sample), the number of times $(\bar{x},\ s)$ occurs twice (i.e. two different samples having the same mean and standard deviation), the number of times $(\bar{x},\ s)$ occurs three times (triples), occurs four time (quadruples) through to occurring 11 times (undecuples). Thus, for instance, on a 10-point scale $(k = 10)$, the total search space for n = 9 contains 921 instances where three distinct samples give the same values for $(\bar{x},\ s)$ and the total search space for n = 12 has 168 sets of seven where each set of seven samples have precisely the same values for $(\bar{x},\ s)$. Inspection of Table 8 and Figure 2 indicates that the likelihood of large sets of possible samples having the same values for $(\bar{x},\ s)$ increases with increasing sample size.

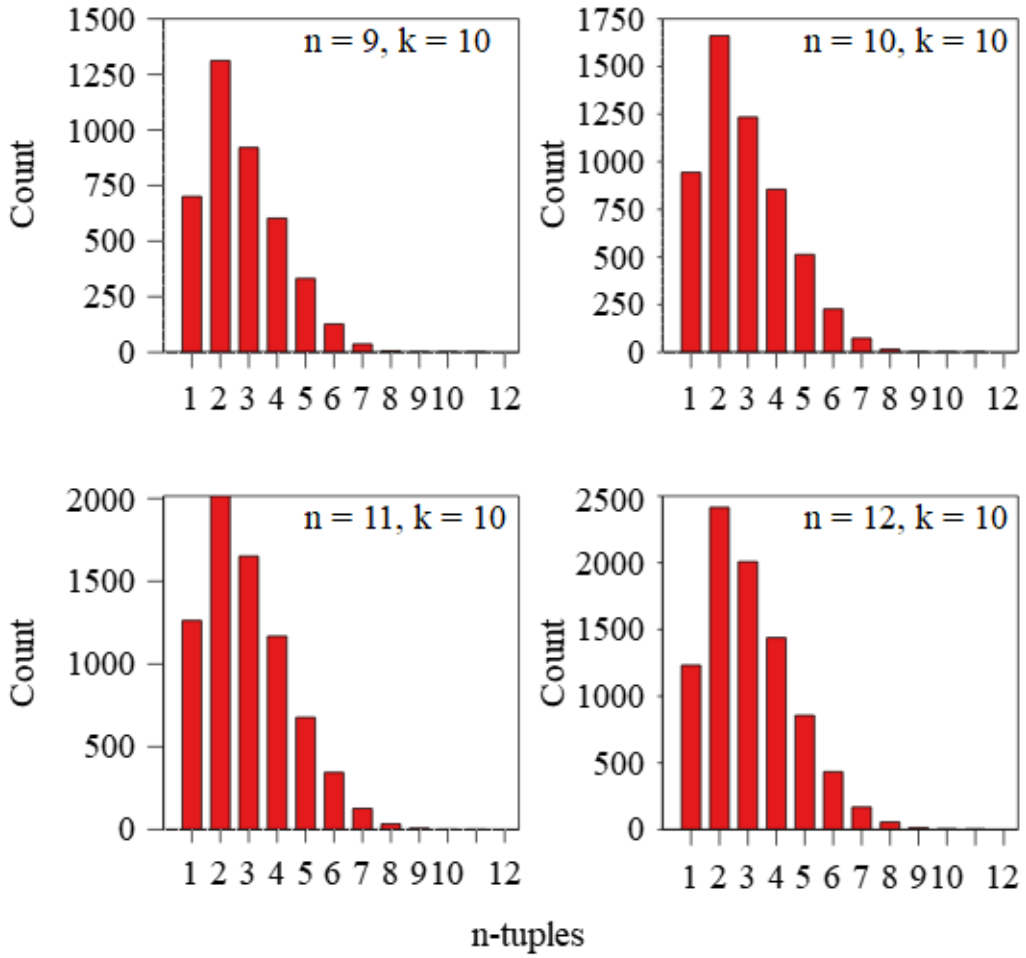

Figure 2 The count is the number of n-tuples (doubles, triples, quadruples, and so on) arising for a given $(\bar{x}, s)$ over all possible $(\bar{x}, s)$ for k = 10.

Table 7 Number of n-tuples (doubles, triples, quadruples, and so on) arising for a given $(\bar{x}, s)$ over all possible $(\bar{x}, s)$ for k = 10.

	Sample Size (n)			
	9	10	11	12
Double	1312	1661	2015	2415
Triple	921	1234	1652	2008
Quadruple	603	855	1168	1438
Quintuple	332	513	678	856
Sextuple	127	227	345	433
Septuple	37	74	127	168
Octuple	6	16	34	56
Nonuple	1	1	8	14
Decuple	1	0	0	5
Undecuple	0	0	0	3

Example: for n = 9 there are 1,312 pairs of samples with the same mean and standard deviation; for n = 9 there are 921 groups each of size three which have the same mean and standard deviation.

Accordingly, not only does the proportion of the search space containing unique sets diminish with increasing n and increasing k but the number of two-way ties, or three-way ties, or higher order ties, also increases with n, indicating the increased difficulty of narrowing down a small solution set.

4. Estimating the Maximum

The mean and sample standard deviation could potentially give information on the maximum value in a data set. For instance, consider a domain comprising non-negative count data, 0, 1, ,2, 3, 4, … where a sample of n= 9 has a mean of 2.111 and a standard deviation of 1.764. In this case the total sum of the observation would be 9*2.111 = 19. Hence an upper bound on the sample maximum, from this statistic alone would be 19. However, the contribution of a sample maximum value, M, to the variance would be $(M - \bar{x})^2$ which has to be less than or equal to $(n-1)s^2$. Accordingly an upper bound on the sample maximum value would be $\bar{x} + \sqrt{(n-1)}\, s$. For the example data, the upper maximum for M would therefore be sqrt(8)*1.764 + 2.111 = 7.1, and since M can only be an integer in the given scenario the search space for uncovering the data set would be 0, 1, ,2, 3, 4, 5, 6, 7. In fact, for the example given, three possible solutions can be derived viz, {1, 1, 1, 1, 2, 3, 4, 4, 4}, {1, 1, 1, 1, 3, 3, 3, 3, 5}, and {1, 1, 1, 2, 2, 2, 3, 4, 5}

4.1 Integer Samples

Suppose for n = 9 a sample with an integer domain is reported with mean = 105.89 and a standard deviation = 3.10. The Chebyshev inequality gives bounds on the proportion of a distribution exceeding the $\mu \pm k\sigma$ to be less than $1/k^2$. A cautious application of this general statement for the example situation, would be to set $1/k^2 = 1/n$. Hence, $k = 3$ and the Chebyshev inequality would identify a restricted search space to be the interval 96.588 to 115.189, which in terms of integers would be 97 to 115.

For the same example, the Selberg inequality, which can give tighter bounds than the Chebyshev inequality, would similarly restrict the search space from 96.09 to 114.99 (using alpha = 9.8, beta = 9.1) which in terms of integers would be 97 to 114 inclusive. Samuelson's inequality, states, with probability 1, the sample is contained in the $\bar{x} \pm \sqrt{(n-1)}\, s$ (as shown above). In this example case the Samuelson interval would be 97.12 to 114.66, which in terms of integers would be 98 to 114 reducing the search space to a 17-point scale. Note

that there is further economy as any solution could contain at most only one of the values 97, 98, 99, 113, 114 as their individual contribution to a solution would account for more than half of the known sample variance (and therefore including two of these values would create a standard deviation exceeding 3.10). Adopting the same logic, any solution could contain at most any two of 97, 98, 99, 100, 111, 112, 113, 114 as they each contribute to more than one third of the known sample variance. This additional knowledge could reduce computation time in discovering solutions by restricting the factorial space. Using the restricted sample space at total of 114 possible integer samples of size 9 would give a mean = 105.89, and sd = 3.10.

4.2 Sample maximum

In other cases, it might be of interest to determine a range of plausible values for the maximum value in a data set based purely on the mean and standard deviation. Suppose we consider the sample 42, 49, 50, 52, 52, 58, 58, 60, 61, 120. Clearly this sample includes one large outlier (120). However, suppose the sample is unknown and only the mean (60.2), standard deviation (21.811), and sample size ($n = 10$) is reported. We may wonder what would be the value of the outlier (maximum) making a further assumption $(M - \bar{x}) > (\bar{x} - m)$.

Samuelson's inequality would indicate that the maximum possible value would be 60.2 + 3*21.811 = 126.3 i.e., an upper bound of 126. Suppose we make a guess that the true maximum is the value 70. If we make this guess than we are saying that the mean of the other 9 values is 53.55 with a standard deviation of 22.844. However, we know that this guess cannot be right as it would mean the removal of an outlier has caused the variance (standard deviation) to increase. Accordingly, we may take the view that the maximum value has to exceed 70. Repeating the "guess" with 80 gives the same conclusion (i.e. variance is larger so 80 cannot be the sample maximum) but repeating with a value of 81 gives a lower variance i.e. if we assume that the removal of the maximum has to reduce the variance then the outlier must have a value of 81 or larger for this example. Hence, we know that the maximum value has to be in the range 81 to 126. However, if we guess the maximum is 126 then the removal of a value equal to 126 would give a negative variance (unfeasible); ditto with 123, 124, and 125. A value of 122 would give a positive variance. Hence based purely on the knowledge of a mean = 60.2, a standard deviation = 21.811, and a sample size n = 10 and an assumption that the variance must decrease on removal of an outlier, then we would know that the maximum value is at least 81 but could be as high as 122. If we have a **reverse** marching observation i.e. an observation being deleted as it marches through the data then we get the following plot of standard deviations which show the unknown maximum

is between 81 and 122 (see Fig 3). If we have marching observation being added then the standard deviation will only increase at the top end of the scale if values are between 84 to 122 suggesting the maximum must be at least 84 (see Figure 3).

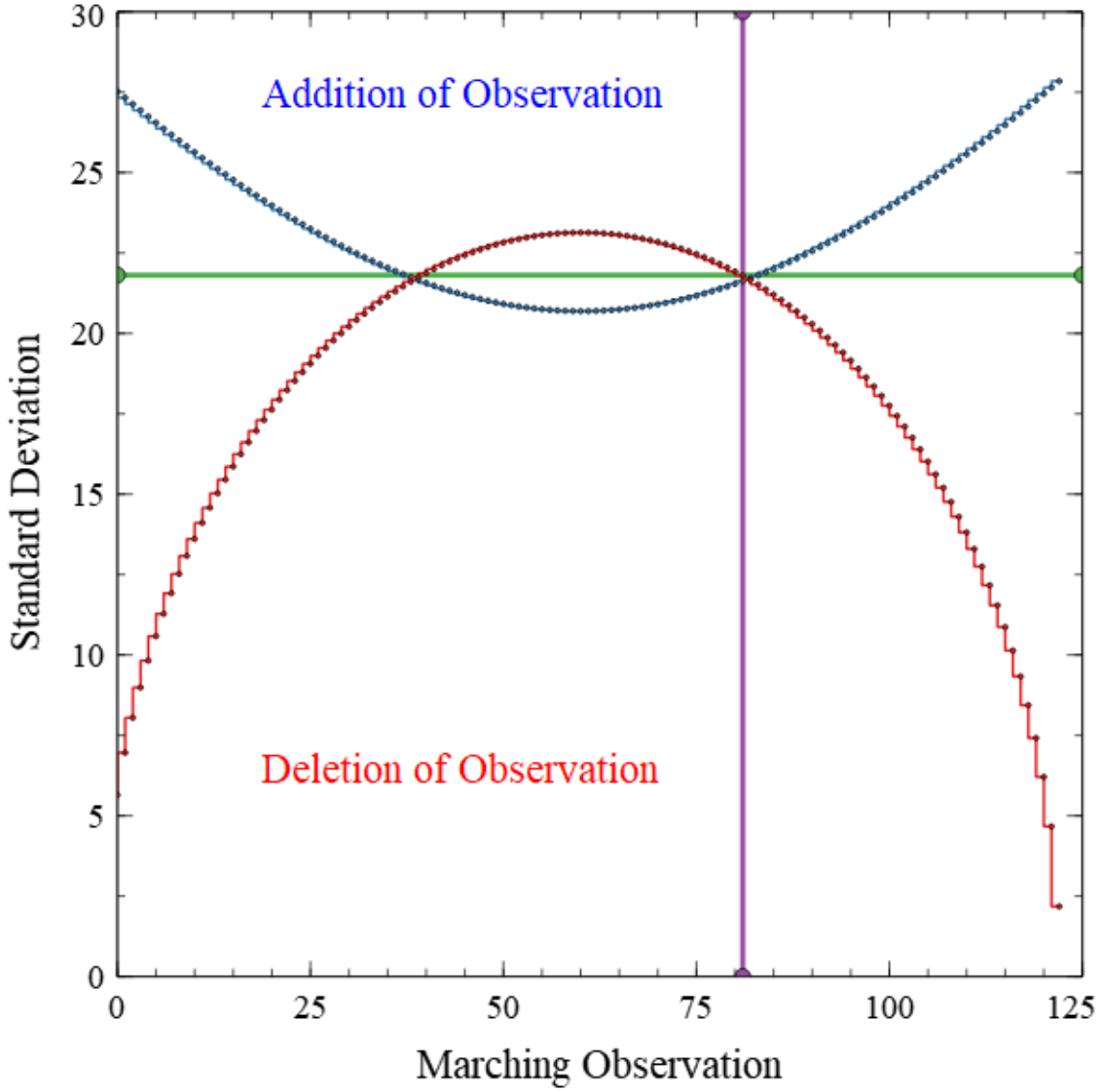

Figure 2

In general, the above bracket for the sample maximum cannot be improved upon as tight equalities apply under certain pathological situations. Additional assumptions would have to be made to tighten the bounds on the maximum. For instance, the marching observation indicates that removing an observation with a value between $\bar{x}$ and 80 would result in a new sample which (i) retains the maximum value (the outlier) but with (ii) a bigger variance

(standard deviation). If we were to assume that two values in the sample would be say, 50 and 52, then their removal would give a sample of 8 observations with mean = 62.25, s = 24.241 and by virtue of the use of the marching removal observation, the range of values for the sample maximum would be 85 to 122. If we extend this and assume that three observations would be close to the central values (e.g. 50, 52, 61) then the new mean and standard deviation would be mean = 62.43, s = 26.18 and the range for the maximum would be 87 to 121.

Of course, the above assumptions of guessing precise values are not fully warranted but we may implement a program such as (a) Randomly pick one number between $\bar{x}$ and 80; (b) remove from sample so as to create a new sample which retains the maximum; (c) calculate the standard deviation (d) estimate the minimum value for the maximum (e) Repeat. (f) Get probability distribution for the minimum maximum and make a probability statement about the lowest value for the maximum. This algorithm could be amended to randomly pick 2 numbers between Mean and 80 for deletion. Or the assumption could be extended to remove three observations. Refinement on the lower bound for the sample maximum could and would be made providing the assumption being made is correct.

5. Little White Lies

For small sample sizes an analyst considering reporting a mean and standard deviation could either not report these statistics (cell suppression), report with the risk of sample features being reversed engineered or provide some information or reduced information by providing either a modified mean or a modified standard deviation or modifications to both mean and standard deviation (little white lies). The R-package, uwedragon, (authored by BD and with R-code given in Appendix 1) will take a sample mean, sample standard deviation and sample size as inputs, along with the domain range, and will enumerate all possible feasible samples based on the information supplied. These outputs will assist the analyst in the decision making over the three options of cell suppression, full information, or little white lies. By illustration we will consider a small sample of size n = 7, with sample values 1, 1, 2, 3, 4, 4, 5 which give a mean = 2.857 and a standard deviation = 1.574. If the values mean = 2.857, standard deviation = 1.574 and n = 7 is entered into the R-package uwedragon then two possible solutions are returned, viz {1, 1, 2, 3, 4, 4, 5} and $\{1,\ 2,\ 2,\ 2,\ 3,\ 5,\ 5\}$.

5.1 Split-half sample. In a split-half sample the sample would be split into two non-overlapping samples of approximate. equal size. For one sample, sample A, the mean would

be calculated, and for the other sample, sample B, the standard deviation calculated. For example, Sample A values {1, 2, 4, 5} and Sample B values {1, 3, 4} give a mean of 3 and a standard deviation of 1.528. Based on these disguised values, the R-package uwedragon gives two possible solutions, {1, 1, 3, 3, 4, 4, 5} and {1, 2, 2, 3, 3, 5, 5} neither of which are the original sample values.

5.2 Bootstrap sample. In this approach a single bootstrap sample (i.e. sample n times with replacement) is used to calculate a mean (Sample A) and a second bootstrap sample (Sample B) is taken to calculate the standard deviation. Application of this approach in the example scenario gave Sample A values {1, 1, 1, 4, 4, 4, 5} and Sample B values {1, 1, 1, 2, 3, 4, 5} to give a modified or disguised mean of 2.587 and standard deviation 1.512. Entry of the disguised values into 'uwedragon' reveals "*no solutions found: data input is incorrect or mean and standard deviation disguised*". If mean and standard deviation are rounded to two decimal places then the same output is given. Further rounding to one decimal place gives four solutions, none of which are the original sample.

5.3 Variable sample sizes. In this approach a random number $(R1)$ between $n/2$ and n is generated. Either with or without replacement, a sample (Sample A) is generated. The process is repeated with a second random number $R2 \in (n/2, n)$ to generate a second sample, Sample B. Sample A is used for the calculation of a mean and Sample B is used for the calculation of a standard deviation. By way of example, sampling with replacement with $R1 = 6$, gives Sample A values {1, 1, 3, 3, 3, 3} with a mean of 2.333, and with $R2 = 5$, gives Sample B values {1, 1, 3, 4, 5} with a standard deviation of 1.789. Entry of the disguised values into 'uwedragon' reveals "*no solutions found: data input is incorrect or mean and standard deviation disguised*". Rounding the mean and standard deviation to either one or two decimal places gives the same conclusion.

Conclusions and Summary

For $n \leq 10$ and $k \leq 10$ it is relatively easy to demonstrate how to reverse engineer the sample space based on sample mean, sample standard deviation, and sample size. For $(n, k) = (11, 11)$ the search space is enumerable but large and the possibility of being able to uniquely identify the underpinning sample is less than 1%. This percentage decreases with the degree of rounding for the sample mean and sample standard deviation, and the extent

of the number of possible n-tuple solutions increases which, a priori, further decreases confidence in being able to make an educated guess about the true underpinning sample.

The R package, uwedragon, will enumerate a potential space providing $n \leq 14$ and $k \leq 14$. This software is intended to assist an analyst in being able to quickly ascertain whether reporting the mean, standard deviation and sample size, would leave the analyst at risk of the sample being reversed engineered. If there is such a risk, it may be prudent to supress some information such as only reporting the mean but not the reporting the standard deviation. Alternatively, it may be acceptable to provide some chance corrected information on the mean and standard deviation which does not permit the underpinning sample to be discovered. These "little white lies" are a compromise position and may be generated using the uwedragon software package.

It is recognised that some interested and determined people may want to us the mean and standard deviation to make an inference about a sample maximum. General long-established theory attributable to Chebyshev, Selberg and Samuelson could be used to place an upper bound on the maximum. However, the process of a conceptual deletion of an observation can improve and reduce this upper bound as shown by example. The same process may also be used to provide a lower bound i.e. to create an interval [lower bound, upper bound] in which the maximum will reside. This process therefore extends general theory in this area. Further assumptions on the number of observations close to the mean could be used to further refine and narrow the interval [lower bound, upper bound] providing the assumptions are true.

References

[1] Selberg HL. (1940) "Zwei Ungleichungen zur Ergänzung des Tchebycheffschen Lemmas" [Two Inequalities Supplementing the Tchebycheff Lemma]. *Skandinavisk Aktuarietidskrift (Scandinavian Actuarial Journal)* (in German). 121–125. doi:10.1080/03461238.1940.10404804

[2] Samuelson PA (1968) How deviant can you be? *Journal of the American Statistical Association*, Vol 63, 1522 – 1525.

[3] Derrick B, Broad A, Toher D and White P (2017) The impact of an extreme observation in a paired samples design, *Metodoloski Zvezki*, Vol 14, No 2, 1 - 11

Appendix

Code for R package uwedragon

Find individual sample values from the sample mean and standard deviation

For integer based scales, finds possible solutions for each value within a sample.
This is revealed upon providing sample size, minimum possible value, maximum possible value, mean, standard deviation (and optionally median).

@param n Sample size.

@param min_poss Minimum possible value. If sample minimum is disclosed, this can be inserted here, otherwise use the theoretical minimum. If there is no theoretical maximum 'Inf' can be inserted.

@param max_poss Maximum possible value. If sample maximum is disclosed, this can be inserted here, otherwise use the theoretical maximum. If there is no theoretical minimum '-Inf' can be inserted.

@param usermean Sample mean.

@param usersd Sample standard deviation, i.e. n-1 denominator.

@param meandp (optional, default=NULL) Number of decimal places mean is reported to, only required if including trailing zeroes.

@param sddp (optional, default=NULL) Number of decimal places standard deviation is reported to, only required if including trailing zeroes.

@param usermed (optional, default=NULL) Sample median.

@return Outputs possible combinations of original integer sample values.

```
solutions<- function(n, min_poss, max_poss, usermean, usersd, meandp=NULL,
sddp=NULL, usermed=NULL) {

#ensure valid data entry
  if (min_poss > max_poss)
    stop("Check data input. Minimum cannot be greater than Maximum")
  if ((is.null(n)) | (is.null(min_poss)) | (is.null(max_poss)) | (is.null(usermean)) |
(is.null(usersd)))
    stop("N, Minimum, Maximum, Mean and Standard Deviation all required")

#check Samuelson's inequality and adjust min / max if this reduces the range
  min_poss_s<- floor(usermean-((sqrt(n-1))*usersd))
  max_poss_s<- ceiling(((sqrt(n-1))*usersd) + usermean)
  if(min_poss_s> min_poss){
```

```
  min_poss_f<- min_poss_s
 }
  else{
   min_poss_f<- min_poss
   }
 if(max_poss_s< max_poss){
  max_poss_f<-max_poss_s
 }
  else{
   max_poss_f<-max_poss
   }
 k<-length(min_poss_f:max_poss_f)

#stop user if combinations too large for R to store in memory
 if (factorial(n+k-1) == Inf)
  stop("Sample size or potential range of values too large to compute solutions")
 if (factorial(n+k-1)/(factorial(k)*factorial(n-1))>65000000)
  stop("Sample size or potential range of values too large to compute solutions")

#apply number of decimal places as determined by user input (where meandp and sddp not
specified)
 decimalplaces <- function(x) {
  if (abs(x - round(x)) > .Machine$double.eps^0.5) {
   nchar(strsplit(sub('0+$', '', as.character(x)), ".", fixed = TRUE)[[1]][[2]])
  } else {
   return(0)
  }
 }

#generate all combinations
 samples<-gtools::combinations(k, n, min_poss_f:max_poss_f, repeats.allowed=TRUE)
 samples<-data.frame(samples)

#calculate summary statistics for all combinations and compare to users stated summary
statistics
 if (is.null(usermed)){
  all<-transform(samples, means=apply(samples, 1, mean))
  all2<-transform(all, sd=apply(samples, 1, sd))
   if (is.null(meandp) & is.null(sddp)) {
    all2$combine<-
paste0(round(all2$means,decimalplaces(usermean)),round(all2$sd,decimalplaces(usersd)))
     user<-
paste0(round(usermean,decimalplaces(usermean)),round(usersd,decimalplaces(usersd)))
     }
   else{
    if (is.null(meandp) | is.null(sddp))
     stop("If either specified, number of decimal places must be specified for both mean and
standard deviation")
    if (meandp > round(meandp) | meandp < round(meandp) | sddp > round(sddp) | sddp <
round(sddp)  )
```

```
        stop("Data entry incorrect. Number of decimal places must be integer")
      all2$combine<-paste0(round(all2$means,meandp),round(all2$sd,sddp))
      user<-paste0(round(usermean,meandp),round(usersd,sddp))
      }
    if(nrow(all[which(all2$combine == user ),c(1:n)])==0){
      warning("No solutions found: data input is incorrect or mean and standard deviation
disguised")
    }
    return(all[which(all2$combine == user ),c(1:n)])
  }
  #if including median
  else {
    all<-transform(samples, means=apply(samples, 1, mean))
    all2<-transform(all, sd=apply(samples, 1, sd))
    all2<-transform(all2, med=apply(samples, 1, median))
    if (is.null(meandp) & is.null(sddp)) {
      all2$combine<-
paste0(round(all2$means,decimalplaces(usermean)),round(all2$sd,decimalplaces(usersd)),ro
und(all2$med,decimalplaces(usermed)))
      user<-
paste0(round(usermean,decimalplaces(usermean)),round(usersd,decimalplaces(usersd)),roun
d(usermed,decimalplaces(usermed)))
      }
    else{
      if (is.null(meandp) | is.null(sddp))
        stop("If either specified, number of decimal places must be specified for both mean and
standard deviation")
      if (meandp > round(meandp) | meandp < round(meandp) | sddp > round(sddp) | sddp <
round(sddp)  )
        stop("Data entry incorrect. Number of decimal places must be integer")
      all2$combine<-
paste0(round(all2$means,meandp),round(all2$sd,sddp),round(all2$med,decimalplaces(userm
ed)))
      user<-paste0(round(usermean,meandp),round(usersd,sddp),
round(usermed,decimalplaces(usermed)))
    }
    if(nrow(all[which(all2$combine == user ),c(1:n)])==0){
      warning("No solutions found: data input is incorrect or mean and standard deviation
disguised")
    }
    return(all[which(all2$combine == user ),c(1:n)])
  }
}
```

######### White lies: Disguise the sample mean and standard deviation #########

Disguises the sample mean and standard deviation via a choice of methods.

@param usersample A vector of all individual sample values.

@param method Approach for disguising mean and standard deviation (default = 2).

@param meandp Number of decimal places mean to be reported to (default = 1).

@param sddp Number of decimal places standard deviation to be reported to (default = 1).

@return Outputs disguised mean and disguised standard deviation.

```
disguise<-function(usersample,method=2,meandp=1,sddp=1){
  n<-length(usersample)
  #### method 1 #####
  if (method == 1){
    ind<-sample(1:n,size=ceiling(n/2), replace = FALSE)
    split<-sort(ind)
    SampleA<-usersample[split]
    SampleB<-usersample[-split]
    SampleAmean<-mean(SampleA)
    SampleBsd<-sd(SampleB)
  }

  if (method == 2){
    #### method 2 #####
    SampleA<-sample(usersample, size =n, replace = TRUE)
    SampleAmean<-mean(SampleA)
    SampleB<-sample(usersample, size =n, replace = TRUE)
    SampleBsd<-sd(SampleB)
  }

  if (method == 3){
    ### method 3 ####
    rnd1<-round(runif(1,min=n/2,max=n),0)
    SampleA<-sample(usersample, size =rnd1, replace = TRUE)
    SampleAmean<-mean(SampleA)
    rnd2<-round(runif(1,min=n/2,max=n),0)
    SampleB<-sample(usersample, size =rnd2, replace = TRUE)
    SampleBsd<-sd(SampleB)
  }

  if (method == 4){
    ### method 4 ####
    rnd1<-round(runif(1,min=n/2,max=n),0)
    SampleA<-sample(usersample, size =rnd1, replace = FALSE)
    SampleAmean<-mean(SampleA)
```

```
    rnd2<-round(runif(1,min=n/2,max=n),0)
    SampleB<-sample(usersample, size =rnd2, replace = FALSE)
    SampleBsd<-sd(SampleB)
  }

#report mean and sd to number of decimal places specified by user
  print(paste0("mean = ",round(SampleAmean,meandp)))
  print(paste0("sd = ",round(SampleBsd,sddp)))
}
```