OPEN-SOURCE EXCEL TOOLS FOR STATISTICAL METROLOGY FOR THE XVIII IMEKO WORLD CONGRESS

Hung-kung Liu, William F. Guthrie, Juan Soto

NIST, Gaithersburg, MD, USA, liu@nist.gov

Abstract: In this paper we introduce an approach for constructing statistical metrology tools with a Microsoft® Excel based interface that uses the open source statistical package R as the computational engine. These metrology tools will enable Excel users to access a larger array of specialized statistical tools for metrology. We will describe our general approach and demonstrate one tool - an automated uncertainty calculator. This uncertainty calculator provides users with a simple and flexible tool for estimating measurement uncertainty using the methods outlined in the ISO Guide to the Expression of Uncertainty in Measurement, popularly known as the ISO GUM. The uncertainty calculator is easy to use for scientists who are comfortable with spreadsheets, can handle a variety of typical correlation structures, and performs a validity check on the results by simulating the coverage probabilities actually attained by the uncertainty intervals.

Keywords: statistical metrology, uncertainty, Excel, R.

1. INTRODUCTION

Many scientists use Microsoft® Excel1 for data collection and statistical analysis. However, Excel has limited statistical functionality, and may not always work well when used for specialized metrology computations, such as the evaluation of measurement uncertainty. In contrast, the statistical package R [1] is a language and environment for statistical computing and graphics. R provides a wide variety of statistical (linear and nonlinear modeling, classical statistical tests, time-series analysis, classification, clustering, etc.) and graphical techniques, and is highly extensible. The S language developed at Bell Laboratories is often the vehicle of choice for research in statistical methodology, and R provides an open source route to participation in that activity. Thomas Baier’s R-(D) COM package and Erich Neuwirth’s Excel add-in R-Excel [2,3,4], combined with R, can be used to extend Excel’s statistical reach. Using their technology, we have developed scripts that harness the power of R to enable Excel users to use NIST recommended methodologies for statistical analysis. One such example is an automatic uncertainty calculator.

Reporting the uncertainty of a measurement result historically has been done using a variety of different approaches. In 1993, however, ISO recommended a standardized approach, in the Guide to the Expression of Uncertainty in Measurement, often called the GUM [5]. The GUM has since been accepted as the default approach among national metrology institutes for assessing and reporting measurement uncertainties. Although adoption of the GUM minimized a wide range of issues that affected uncertainty assessment in the past, computation of measurement uncertainty, especially with correlated variables or correlated uncertainties, remains a problem for many metrologists.

With that in mind, we developed an uncertainty calculator with an Excel-based interface that uses the R as the computational engine to provide scientists and metrologists extended computational capabilities for uncertainty analysis, including symbolic differentiation2, methods for handling correlated input variables, and methods for repeated use of standard uncertainty estimates. The calculator also includes a validity check on the results by simulating the coverage probabilities of the uncertainty intervals obtained. The Kragten spreadsheet [6], and some other uncertainty software tools, do not have these advanced capabilities.

In the remainder of this paper, we discuss the development and use of the uncertainty calculator as an example of the type of tools that we plan to develop as part of a suite of NIST Statistical Metrology Tools. In section 2, we discuss the steps we follow in writing R functions to implement the GUM uncertainty computations. Section 3 describes how the uncertainty spreadsheet fits into the general architecture of our toolkit. Section 4 demonstrates the use of the uncertainty calculator for the calibration of a...

1 Identification of commercial products in this document does not imply recommendation or endorsement by the National Institute of Standards and Technology, nor does it imply that the products identified are necessarily the best available for the purpose.

2 As explained by Dr. Gordon S. Novak of University of Texas, “A symbolic differentiation program finds the derivative of a given formula with respect to a specified variable, producing a new formula as its output. In general, symbolic mathematics programs manipulate formulas to produce new formulas, rather than performing numeric calculations based on formulas. In a sense, symbolic manipulation programs are more powerful: a formula provides answers to a class of problems, while a numeric answer applies only to a single problem.” (http://www.cs.utexas.edu/users/novak/asg-symdif.html)
load and Section 5 outlines the simulation used for the validation assessment of the expanded uncertainty.

2. KEY STEPS IN THE GUM METHOD

One of the key steps incorporated in the GUM is the use of a measurement equation that shows how all the input quantities are combined to obtain the desired measurement result. That is, the value of a measurand is related to input quantities denoted by $\mu_1, \mu_2, \ldots, \mu_n$ through a known functional relationship of the form

$$\theta = f(\mu_1, \mu_2, \ldots, \mu_n),$$

where the function $f$ is generally determined by physical laws relating the relevant quantities. In practice, the values of $\mu_j$ are not known exactly. However, mean results $\bar{X}_1, \bar{X}_2, \ldots, \bar{X}_n$ are available that estimate $\mu_1, \mu_2, \ldots, \mu_n$, respectively. The GUM recommends the same functional relationship that relates the value $\theta$ of the measurand to the input variables $\mu_1, \mu_2, \ldots, \mu_n$ be used to calculate $Y$ from $\bar{X}_1, \bar{X}_2, \ldots, \bar{X}_n$. That is, let $Y$ be the estimate of $\theta$ that is obtained from $Y = f(\bar{X}_1, \bar{X}_2, \ldots, \bar{X}_n)$. Consequently the uncertainty in the measurement result $Y$ must be calculated or estimated by propagating the uncertainties in the quantities $\bar{X}_1, \bar{X}_2, \ldots, \bar{X}_n$ used to compute $Y$.

The GUM also recognizes two primary methods, Type A and Type B, for the summarization of the data for each measurement equation input. The most commonly used Type A (or statistical) estimator for the input $\mu_i$ is the sample mean of $n_i$ observations:

$$\bar{X}_i = \frac{1}{n_i} \sum_{j=1}^{n_i} X_{ij},$$

(2)

The most commonly used method for summarizing data to obtain the Type A standard uncertainty of $\bar{X}_i$ is

$$u_{\bar{X}_i} = s_i / \sqrt{n_i},$$

where

$$s_i = \sqrt{\frac{1}{n_i-1} \sum_{j=1}^{n_i} (X_{ij} - \bar{X}_i)^2}$$

(3)

is the sample standard deviation with $v_{\bar{X}_i} = (n_i - 1)$ degrees of freedom.

Type B (or non-statistical) types of information that can be used to estimate an input include scientific judgment, manufacturer’s specifications, or other indirectly related or incompletely specified information.

The GUM procedure for finding the combined standard uncertainty of $Y$ is then to linearize the measurement equation and obtain the standard uncertainty of the linearized result. The first-order Taylor series for values of the function $f(\bar{X}_1, \ldots, \bar{X}_n)$ near $f(\mu_1, \ldots, \mu_n)$ is given by

$$f(\bar{X}_1, \ldots, \bar{X}_n) = f(\mu_1, \ldots, \mu_n)$$

$$+ \left[ \frac{\partial f}{\partial \bar{X}_1} \right] (\bar{X}_1 - \mu_1) + \cdots + \left[ \frac{\partial f}{\partial \bar{X}_n} \right] (\bar{X}_n - \mu_n)$$

(4)

where the quantities in the square brackets are the partial derivatives of the measurement equation with respect to the $i$th input to the measurement equation evaluated at $\bar{X}_i$. Treating the partial derivatives as constants and applying the rules for the propagation of uncertainty of linear measurement equations gives the approximate combined standard uncertainty of $f(\bar{X}_1, \ldots, \bar{X}_n)$

$$u_c = \sqrt{\left( \frac{\partial f}{\partial X_1} \right)^2 u_{X_1}^2 + \cdots + \left( \frac{\partial f}{\partial X_n} \right)^2 u_{X_n}^2}$$

$$+ 2 \rho_{X_1,X_2} \frac{\partial f}{\partial X_1} \frac{\partial f}{\partial X_2} u_{X_1} u_{X_2} + \cdots$$

(5)

where $\rho_{X_1,X_2}$ is the correlation between $\bar{X}_1$ and $\bar{X}_2$. Treating the partial derivatives as constants even though they are evaluated at the values of the sample means is not unreasonable because, to the 1st order of approximation, all derivatives are constants. The partial derivatives, called sensitivity coefficients, are typically denoted by $c_r$ to simplify the notation, thus

$$u_c = \sqrt{c_{\bar{X}_1}^2 u_{\bar{X}_1}^2 + \cdots + c_{\bar{X}_n}^2 u_{\bar{X}_n}^2 + 2 c_{\bar{X}_1} c_{\bar{X}_2} \rho_{X_1,X_2} \mu_n u_{\bar{X}_1} u_{\bar{X}_2} + \cdots}$$

(6)

Finally, to quantify the uncertainty of $u_c$, the GUM suggests computing the effective degrees of freedom of $u_c$ using the Welch-Satterthwaite formula

$$v_{eff} = \frac{u_c^4}{\bar{X}_1 u_{\bar{X}_1}^4 + \cdots + \bar{X}_n u_{\bar{X}_n}^4}.$$  

(7)

This is only appropriate, however, when all of the individual uncertainties in the denominator are estimated independently of one another.

After obtaining $f(\bar{X}_1, \ldots, \bar{X}_n)$, $u_c$, and $v_{eff}$, probabilistic statements about the measurand

$$P\left\{ -k \leq f(\bar{X}_1, \ldots, \bar{X}_n) - f(\mu_1, \ldots, \mu_n) \leq k \right\} = 0.95$$

(8)

can be solved using the appropriate constant, $k$, from Student’s $t$ distribution. Solving for $f(\mu_1, \ldots, \mu_n)$ gives the expanded uncertainty intervals.
The target coverage probability of these intervals is typically 0.95.

3. TOOL KIT ARCHITECTURE

The suite of NIST Statistical Metrology Tools is constructed using four building blocks: (a) The Microsoft® Excel front-end, (b) The Microsoft® Excel native programming language, Visual Basic for Applications (VBA), (c) the open source programming environment for statistical computation and graphics, and (d) R-(D) COM Server V1.35 (R-Excel Add-in) developed by Thomas Baier and Erich Neuwirth. Collectively, these four components are used to build and package computational metrology solutions. Fig. 1 illustrates the use of these different components for uncertainty analysis based on the GUM.

Fig. 1. Computational steps depicting GUM uncertainty analysis using the R-Excel spreadsheet.

To effectively understand how these components are interrelated we will briefly describe each of these components and how they operate in unison.

**Microsoft® Excel:** Widely available among desktops running Microsoft® Windows operating systems, this spreadsheet application enables users to store, process, analyze and graph their data. To provide greater access to alternative statistical methods and libraries, we have opted to: (a) construct tools using both R and R-Excel, and (b) integrate these tools into Excel.

**Visual Basic for Applications (VBA):** Using VBA subroutines we built up the GUI (graphical user interface) to create a hook for a Metrology menu. Currently, our prototype implementation is based on the GUM. However, we envision that the suite of metrology tools will support additional methods including: the GUM Supplement, Bootstrap uncertainty analysis, Bayes uncertainty analysis, and ANOVA. In addition, we wrote subroutines to replicate the GUM template, thereby enabling users to have multiple worksheets in a single workbook.

**R (D) COM Server (R-Excel Add-in):** Fortunately, Baier & Neuwirth [4] created a COM server and an Excel Add-in⁴ to facilitate the integration of R into Excel. Via this mechanism the client application (e.g., Excel) can communicate with R (the COM server). In fact, there are multiple ways to go about implementing such a connection. However, for our purposes we focused on *scratchpad mode*⁵. This mode made sense given the desire to have a solution that would update interactively with each change in a cell value. Additional information regarding COM servers may be found in [2, 4]. An Excel Add-in essentially incorporates a plug-in into the Excel environment, thereby exposing public methods defined in the add-in.

**Statistical Computing Environment:** Ultimately, data contained in the Excel spreadsheet are passed into R (the statistical computation engine), processed and then returned back to Excel.

4. LOAD CELL CALIBRATION EXAMPLE

We will focus on the calibration and use of a load cell to illustrate the features and operation of the GUM uncertainty spreadsheet. The GUM approach used here will contrast with the presentation of this example in Section 6.1 of the NIST/SEMATECH e-Handbook of Statistical Methods [7], where an approach based on the theory of least-squares regression is used. The GUM approach has the advantages that the computations are carried out in the familiar, standard framework promoted in the GUM [5] and that the computations are largely automated.

Fig. 2 shows the observed deflections of the cell when loaded with the known loads used in the collection of the calibration data. (Note, the loads used here have been scaled relative to the load data presented in [7] to reduce the effects of floating-point errors during computation.) Although it looks like a straight-line model would be an appropriate calibration function, as discussed in [7], a quadratic model is a more appropriate calibration model for this load cell.

![Fig. 2. Load cell calibration data with 20 load levels measured in duplicate (duplicates not visible on this scale).](image)

---

⁴ An Excel Add-in is a software application designed to extend Excel functionality.

⁵ Writing R Code directly in an Excel worksheet and transferring scalar, vector, and matrix variables between R and Excel.
Denoting the deflection of the cell \( D \) for an applied load, \( L \), the quadratic model is expressed
\[
D = \beta_0 + \beta_1 L + \beta_2 L^2 + \epsilon, \tag{10}
\]
with unknown parameters \( \beta \) and a random measurement error \( \epsilon \). Assuming the random errors are independent and approximately follow a Gaussian distribution with a mean of zero and an unknown standard deviation, \( \sigma \), the parameters can be estimated using ordinary least-squares regression.

After the fit of the model has been validated using graphical residual analysis and appropriate numerical methods [7], the model can be used to estimate an unknown load applied to the cell based on its observed deflection.

Table 1 gives the output from the fit of the quadratic model to the data.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Estimated Value</th>
<th>Standard Uncertainty</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \beta_0 )</td>
<td>6.73566E-04</td>
<td>1.07939E-04</td>
</tr>
<tr>
<td>( \beta_1 )</td>
<td>7.32059E-01</td>
<td>1.57817E-04</td>
</tr>
<tr>
<td>( \beta_2 )</td>
<td>-3.16082E-03</td>
<td>4.86653E-05</td>
</tr>
<tr>
<td>( \sigma )</td>
<td>2.05177E-04</td>
<td>37</td>
</tr>
</tbody>
</table>

Following [7], suppose an unknown load applied to the cell produces a deflection of \( D=1.239722 \). Then the observed value of \( D \) can be substituted into the calibration function and the load can be obtained by solving for \( L \) using the quadratic formula. Thus, in general, the measurement equation to estimate an unknown load using this load cell can be written
\[
L = -\frac{\beta_0 + \sqrt{\beta_1^2 - 4\beta_2(D - \beta_0)}}{2\beta_2}, \tag{11}
\]
where the estimates of each parameter based on the calibration data are differentiated from their actual unknown values by adding a “hat” over the symbol for each parameter. Although there are theoretically two possible solutions for \( L \) because we are using a quadratic calibration function, we can easily determine which solution is correct because one solution will fall within (or near) the range of the calibration data and one will be well outside the range of the data. The measurement equation (11) above shows the correct solution for this data.

Of course, for an estimated load value obtained using eq. (11) to have any meaning, an assessment of its uncertainty is also required. Based on the information given in Table 1, it looks like this should only require a fairly straight-forward application of the standard rules for the propagation of uncertainty as outlined in [5]. Although the measurement equation is somewhat complicated, we have estimates and associated standard uncertainties for each of the parameters in the calibration model and the standard uncertainty of an observed deflection is given by \( \sigma \), also obtained from the fit of the calibration model.

Unfortunately, however, two complications keep the appropriate uncertainty computations from being as simple as they might seem at first glance. The first is the fact that the estimated parameter values obtained from the fit of the model are correlated with one another. The second complication arises from the fact that the standard uncertainties of the estimated parameters and the observed deflection are all based on the residual standard deviation of the fit, \( \sigma \). Thus the uncertainty estimates for each parameter are also (perfectly) correlated with one another. Both of these complications affect the uncertainty computations and must be addressed to ensure that the uncertainty computed for an estimated load will be fit for purpose.

Fortunately these two complications can be overcome fairly easily using the GUM uncertainty spreadsheet since it can be used to account for the correlations between the input variables in the measurement equation and the uncertainty estimates. To take advantage of these features of the spreadsheet, we will need to be able to specify the correlations between the estimated parameters in the calibration output and specify what subset of the standard uncertainties are based on a common estimate. In general, determining the correlations between inputs to a measurement equation can be difficult, however, in this case, because the correlations result purely from the fit of the model to the data, we know what those correlations are [5,8] and they are often included in regression analysis output. For this data, the correlations between the parameter estimates are given in Table 2.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>( \beta_0 )</th>
<th>( \beta_1 )</th>
<th>( \beta_2 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \beta_0 )</td>
<td>1.000</td>
<td>-0.889</td>
<td>0.781</td>
</tr>
<tr>
<td>( \beta_1 )</td>
<td>-0.889</td>
<td>1.000</td>
<td>-0.971</td>
</tr>
<tr>
<td>( \beta_2 )</td>
<td>0.781</td>
<td>-0.971</td>
<td>1.000</td>
</tr>
</tbody>
</table>

As a further consequence of the mathematics underlying least-squares fitting, we know that the standard uncertainties of the parameter estimates are all based on the residual standard deviation from the least-squares fit. Because it also seems reasonable to assume that the operation of the load cell is same under regular testing conditions as it is during calibration, it also seems reasonable to estimate the standard uncertainty of the observed deflection, \( D=1.239722 \), using the residual standard deviation, \( \sigma \).

Fig. 3 shows the layout of the GUM uncertainty spreadsheet. On rows 2, 5, and 8, the user first simply enters the number of inputs in the measurement equation, the names of the inputs, and the measurement equation. The values for the load cell calibration have been entered here based on eq. (11). Then the estimated parameter values, their uncertainties, and degrees of freedom, as obtained from Table 1, are entered in rows 11 through 13 and the columns labeled with the appropriate input name. For each input the user can select whether the standard uncertainty is a Type A or Type B uncertainty in row 14 and can select the appropriate probability distribution for each input in row 15.

In cell A18 the user can select whether or not there are any common uncertainty estimates underlying the standard uncertainties of the inputs. If “Yes” is selected, then the
Table showing standard uncertainties and their contributions to the overall uncertainty. Note that the standard uncertainties share a common basis. For this example, all of the standard uncertainties are based on the residual standard deviation, so the variables listed in row 17 all share a common class, as denoted by the common value given for each variable in row 18.

As for cell A18, the user can indicate if any of the estimated input quantities, uncertainties, and other inputs are correlated. In the Monte Carlo study, we use the estimated input quantities, uncertainties, and other inputs as a coverage probability. A simulated coverage near the target value of 95% is an indication of the validity of the uncertainty interval. Finally, the value given in cell E31 is an indication of the validity of the uncertainty interval. The estimated load has a combined standard uncertainty of 2.922710\(g\), with a coverage probability of 95%. This coverage probability is the result of a simulation using the estimated input quantities, uncertainties, and other inputs as an input to the measurement equation.

To the number of digits reported, these values are quite close to the results given in [7], illustrating the similarity of these two approaches. For further comparison, if the correlations between the inputs and the uncertainty estimates had not been taken into account, the expanded uncertainty would have been \(U = 10.51 \times 10^{-4}\) g, almost twice the level of the appropriate value, clearly demonstrating the potential impact correlations can have on uncertainty assessment.

### 5. Validity of Uncertainty Intervals

As described in section 2, the GUM takes \(u(\bar{X})\), for \(i = 1,...,n\), as its input to produce an uncertainty interval \(I\). The true unknown frequentist coverage probability

\[
P_I(\{\mu_i, ..., \mu_n\} \in \bar{X})
\]

measures the validity of the GUM procedure. The unknown coverage probability can be estimated by the coverage frequency based on a Monte Carlo study of a large number of \(N\) replicates. We study the validity of the GUM uncertainty interval by checking how close this coverage frequency is to its target of 0.95.

In the Monte Carlo study, we use the estimated quantities \((\bar{X}, u(\bar{X}), N(\bar{X}))\) and the user-provided

---

5 This assessment of the validity is based on the assumption that the values observed for each input quantity are sufficiently near their associated true values, as described below, so that the simulation results based on the input estimates will be relevant to the physical situation.
correlation matrix as the true values to simulate \((X_i^*, u(X_i^*), \nu(X_i^*))\). In other words, we use \(\bar{X}\) as \(\mu\), and generate new values of \(\bar{X}\) based on the new artificially known measurand. We denote the simulated \(\bar{X}\) as \(X_i^*\). We apply the GUM approach to the simulated data \((X_i^*, u(X_i^*), \nu(X_i^*))\), for \(i = 1, ..., n\), to obtain an uncertainty interval \(I^*\). This process is repeated \(N\) times to yield \(N\) such \(I^*\)’s. The coverage probability is then estimated by the relative frequency of the coverage of \(f(\bar{X}_1, ..., \bar{X}_n)\) by these \(N\) intervals.

\[
\begin{align*}
X_1^*, ..., X_n^* \\
u(X_1^*), ..., \nu(X_n^*) \\
\text{Count} = 1, \text{if } f(\bar{X}_1, ..., \bar{X}_n) \in I^* \\
0, \text{otherwise}
\end{align*}
\]

Fig. 4. Schematic depiction of the simulation process.

Fig. 4 illustrates this simulation process. The coverage frequency equals \(\sum \text{Count} / N\).

For each input, depending on whether its uncertainty is Type A or Type B, we use different model to simulate \((X^*, u(X^*), \nu(X^*))\). We assume that Type A data comes from a normal population, and that Type B data can come from a uniform, triangular, or normal distribution. Fig. 5 and Fig. 6 specify, in detail, the parent distributions for the Type A and Type B inputs respectively.

Generate a random sample of size \(n\) from a normal distribution with mean \(\bar{X}\) and standard deviation \(\nu(\bar{X})\).

\[
\cdot X^* = \text{the sample mean (of the generated data)}
\cdot u(X^*) = (\text{the sample s.d.}) / \sqrt{n}
\cdot \nu(X^*) = \nu(\bar{X})
\]

Fig. 5. Illustration of the simulation of a Type A input.

6. CONCLUSIONS

In this paper we have discussed the development of statistical metrology tools with a Microsoft® Excel based interface using the open source statistical package R as the computational engine. This process was illustrated with an automated uncertainty calculator that incorporates several features (e.g. use of common uncertainty estimates and confirmation of coverage probability) not available in earlier uncertainty software. These statistical metrology tools should enable scientists, engineers and metrologists who use Excel to access a growing range of specialized statistical tools for metrology.

REFERENCES


http://www.ci.tuwien.ac.at/Conferences/DSC-2003/Proceedings/

http://www.ci.tuwien.ac.at/Conferences/DSC-2001/Proceedings/index.html

http://www.ci.tuwien.ac.at/Conferences/DSC-2003/Proceedings/


