



**NAVAL
POSTGRADUATE
SCHOOL**

MONTEREY, CALIFORNIA

**The “Smarter Regression” Add-In for Linear
and Logistic
Regression in Excel**

by

Samuel E. Buttrey

July 2007

Approved for public release; distribution is unlimited.

Prepared for: Operations Research Department
Naval Postgraduate School
Monterey, CA 93943-5219

**NAVAL POSTGRADUATE SCHOOL
MONTEREY, CA 93943-5001**

VADM Daniel T. Oliver, USN (Ret.)
President

Leonard A. Ferrari
Provost

This report was prepared for and funded by the Operations Research Department,
Naval Postgraduate School, Monterey, CA 93943-5219.

Reproduction of all or part of this report is authorized.

This report was prepared by:

SAMUEL E. BUTTREY
Associate Professor of
Operations Research

Reviewed by:

SUSAN M. SANCHEZ
Associate Chairman for Research
Department of Operations Research

Released by:

JAMES N. EAGLE
Chairman
Department of Operations Research

DAN C. BOGER
Interim Associate Provost and
Dean of Research

REPORT DOCUMENTATION PAGE		<i>Form Approved OMB No. 0704-0188</i>	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington, DC 20503.			
1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE July 2007	3. REPORT TYPE AND DATES COVERED Technical Report	
4. TITLE AND SUBTITLE: The "Smarter Regression" Add-In for Linear and Logistic Regression in Excel		5. FUNDING NUMBERS	
6. AUTHOR(S) Samuel E. Buttrey		8. PERFORMING ORGANIZATION REPORT NUMBER NPS-OR-07-002	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000		10. SPONSORING / MONITORING AGENCY REPORT NUMBER	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)		11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.	
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.		12b. DISTRIBUTION CODE	
13. ABSTRACT (maximum 200 words) The widely-used Excel spreadsheet program has a linear regression routine, but it has a number of drawbacks: it does not handle categorical predictors; it requires to the user to generate columns for interactions; it cannot compute logistic regressions; and it is limited to 16 predictor columns. We have developed an Excel add-in that does both logistic and linear regression, handles categorical and interaction variables in an obvious way, and removes the 16-column limit. We also support nested model tests and, for linear regression, transformations of the response variable. Although we do not claim that Excel is the proper tool for data analysis, our tool can make both small, quick analyses and introductory statistics courses simpler and more complete.			
14. SUBJECT TERMS Linear Regression, Logistic Regression, Categorical Variables, Excel		15. NUMBER OF PAGES 17	
		16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UU

The “Smarter Regression” Add-In for Linear and Logistic Regression in Excel

Samuel E. Buttrey
Naval Postgraduate School

Abstract

The widely-used Excel spreadsheet program has a linear regression routine, but it has a number of drawbacks: it does not handle categorical predictors; it requires to the user to generate columns for interactions; it cannot compute logistic regressions; and it is limited to 16 predictor columns. We have developed an Excel add-in that does both logistic and linear regression, handles categorical and interaction variables in an obvious way, and removes the 16-column limit. We also support nested model tests and, for linear regression, transformations of the response variable. Although we do not claim that Excel is the proper tool for data analysis, our tool can make both small, quick analyses and introductory statistics courses simpler and more complete.

1. The Excel Spreadsheet Program and Regression

A large number of introductory statistics courses teach statistics through the popular spreadsheet program Excel (see, e.g., Levine, Berenson, and Stephan, 1998). Excel has a number of features that make it useful in this effort; in addition, it is easy to learn and highly interactive. However, the built-in statistical functions are few and, in some cases, unreliable (McCullough and Wilson, 1999). Functionality that is not integral to Excel can often be added by means of an “add-in,” which is Excel’s name for a library or package installed by the user.

Excel comes with some add-ins that are supplied with the program. Among these is one called “Analysis Toolpak,” which includes a linear regression routine. That routine has a number of drawbacks: the set of regressors must be contiguous on the sheet; there must be no more than 16 regressors; and all regressors must be numeric. This last means that students need to do their own coding of categorical (and interaction) variables, and, of course, those columns count against the maximum of 16.

The inadequacies of Excel for statistics in general have been widely discussed (see, e.g., McCullough and Wilson, 1999). It is not our intention to defend the use of Excel as a statistics package. However, we are interested in teaching statistics to students who use only Excel. It is our belief that an introductory course in statistics can be taught using only Excel; students interested in doing statistical analyses for real-life problems would then need to learn how to use a more reliable and robust package.

In order to teach an introduction to regression that could occupy only a couple of weeks' worth of class, we set out to find an Excel add-in that:

- performs both linear and logistic regression;
- makes it easy to transform the response variable in linear regression;
- handles categorical predictor variables;
- generates and includes two-term interactions as desired;
- makes it easy to exclude a regressor and refit;
- makes it easy to compare two regressions with the nested model F test for linear regression, or the χ^2 test for logistic regression; and
- is free of charge.

Because we were unable to find such an add-in, we set out to create one ourselves. Our design philosophy had two points: first, we wanted to use only Visual Basic, the built-in programming language of Excel (Walkenbach, 1999). We wanted to avoid the use of external Dynamic Linked Libraries (DLLs) since using them can cause installation issues. Second, we wanted to include only the most necessary features, since this tool is intended for instructing beginning students. Rather than add features to this add-in, we feel it wise to direct students with more sophisticated analysis needs to the appropriate software. We call our software the “Smarter Regression” add-in and refer it here as “SR.”

In the next section, we describe the user interface and show an example of it in use. Section 3 describes the computational details of the add-in, and Section 4 offers some conclusions and ideas for improvements. We will follow the convention of denoting matrices (even in Excel) by uppercase bold letters and vectors as lowercase bold letters.

2. The User Interface

2.1 Preparing the Data

In order to use SR, the user needs a rectangular array of data, as in many other regression routines. The first row of data should contain column labels; the second row contains indicators about the data in each column; and subsequent rows have data. Every element in the second row must contain a keyword denoting that column's purpose in the regression. Those keywords are:

- **Response:** labeling the response variable in a linear regression.
- **RespCat:** labeling the response variable in a logistic regression.
- **Numeric:** labeling numeric predictors.
- **Categorical (or Cat):** labeling categorical predictors.
- **Ignore:** labeling columns to be ignored.

Every column must contain exactly one of these case-sensitive labels. Of course, every regression problem must have exactly one column labeled either “Response” or “RespCat.” A categorical predictor variable must have fewer categories than the value of

a global variable (MAXCAT), currently set to 12; a categorical response variable must have exactly two different categorical labels.

We chose to use this label set-up for two reasons. One, of course, is the convenience (from our standpoint) of writing the add-in, but we also feel there is a benefit to the student in terms of simplicity. Rather than have to select a model interactively with each call to the add-in, he or she can make changes to the labels, then re-run the function call in its entirety. The drawback of this approach is that the presence of the second row makes the data's format slightly unusual, which can cause difficulties when the data needs to be exported to another system.

2.2 The “Smarter” Window

As an example, Figure 1 shows the window displayed when the SR add-in is invoked by the user. The sample data set behind the window is the well-known “stack loss” data (Brownlee, 1965), although for purposes of this exposition we have discretized the “Air Flow” column. Each column is labeled according to its use in the model: “Stack Loss” is the response, “Air Flow,” “Water Temp,” and “Acid Conc.” are the predictors, the first one being discrete; and “AF” is the “Air Flow” column, recoded into a two-level variable for use in our demonstration of logistic regression below. It is not used in this model, so its keyword is “Ignore.”

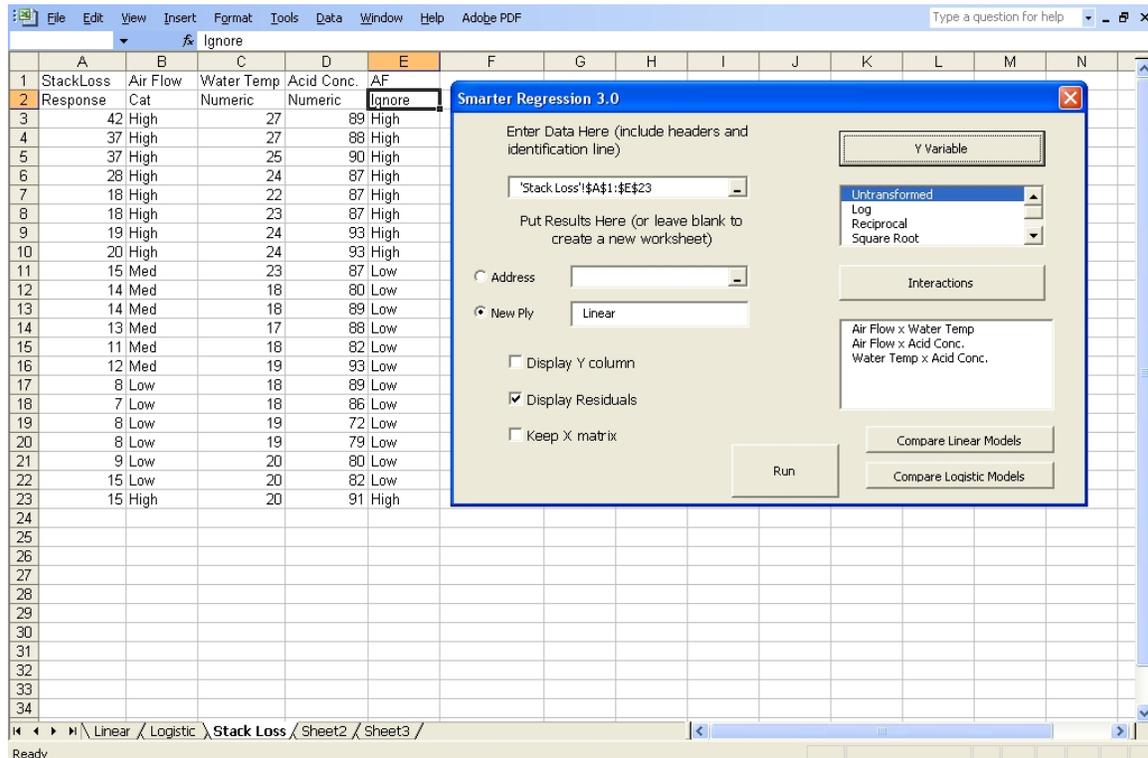


Figure 1: The “Smarter” Window

The area in the top left of the “Smarter” window holds the reference to the data, and is filled first. Then, when the user clicks anywhere else on the form, the “Y” and “Interactions” windows are filled in. (Details on that process are postponed until Section 3.2.) In Figure 1, the choices available to the user are visible. Since this is a linear regression, the “Y” box holds a list of available transformations. The user may select any of these; the default, of course, is no transformation. The “Interactions” box lists all pairs of predictor variables; the user may select any or none of these. For an ordinary no-transformation regression with no interactions, no action is necessary. The “Run” button then runs the regression and produces the results.

The remaining controls on the left side of the SR window give the user some options regarding the set of results displayed and their location. By default, the results are placed in a new sheet named “Linear,” any existing sheet by that name being deleted after a warning. The user can opt to place the results at some other location. In addition, he or she can elect to compute and keep the original Y values, the residuals, and the model matrix, all of which, it seems to us, can have educational value.

Figure 2 shows the “Linear” worksheet after the model in Figure 1 was executed. We have used treatment contrasts for the categorical variable “Air Flow,” and chosen whichever level appears first in the data—in this case, “High”—as the baseline. That baseline level is reported in the output, together with its implicit regression coefficient of 0. Our hope is that this display will make the students more aware of the way we typically treat categorical variables in regression models. We also round the regression coefficients. However, for technical reasons, we currently do not round or display baseline levels when there are more than 16 columns in the model matrix, nor for logistic regression.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	Coefficient	Estimate	Std. Error	t value	2-sided p		Predicted	Residual							
2	Intercept	-31.356	30.0018	-1.0451	0.3115		33.95383	8.046166							
3	Air Flow : High	0	(baseline)				34.0213	2.978696							
4	Air Flow : Med	0.6146	4.3321	0.1419	0.8889		28.6038	8.396201							
5	Air Flow : Low	-4.1742	4.8076	-0.8682	0.3981		26.16493	1.835073							
6	Water Temp	2.6413	0.6268	4.2136	0.0007		20.88236	-2.88236							
7	Acid Conc.	-0.0675	0.2791	-0.2417	0.8121		23.52364	-5.52364							
8	SE Est.	5.1143					25.76011	-6.76011							
9	n	21	Res DF	16			25.76011	-5.76011							
10	y Transform	None					24.13827	-9.13827							
11							11.40415	2.595847							
12							10.79692	3.203077							
13							8.223111	4.776889							
14							11.26921	-0.26921							
15							13.16833	-1.16833							
16							6.008114	1.991886							
17							6.210524	0.789476							
18							9.796387	-1.79639							
19							9.324097	-1.3241							
20							11.89791	-2.89791							
21							11.76297	3.237031							
22							15.32992	-0.32992							
23															
24															
25															
26															
27															
28															
29															
30															
31															
32															
33															
34															

Figure 2: Linear Regression Results

The logistic regression window looks almost exactly like the linear one, although in the current version of the add-in, the **X** matrix needs to be retained. Figure 3 shows the same data that is visible in Figure 1; notice, though, that here the column labels have been changed. In Figure 3, we have made the (new) variable “AF” into a response variable for logistic regression, and it carries the label “RespCat.” Notice also that we have excluded the “Stack Loss” column as a predictor. For this simple example, including that column leads to numerical instability (observe that every AF for which “Stack Loss” > 15 has the value “High,” every “AF” for which “Stack Loss” < 15 is “Low,” and there is one of each for “Stack Loss” = 15). A student who includes that term in the regression will see a pop-up message box with the phrase “Unable to compute SE’s: possible numeric problems.” This, too, may have educational value.

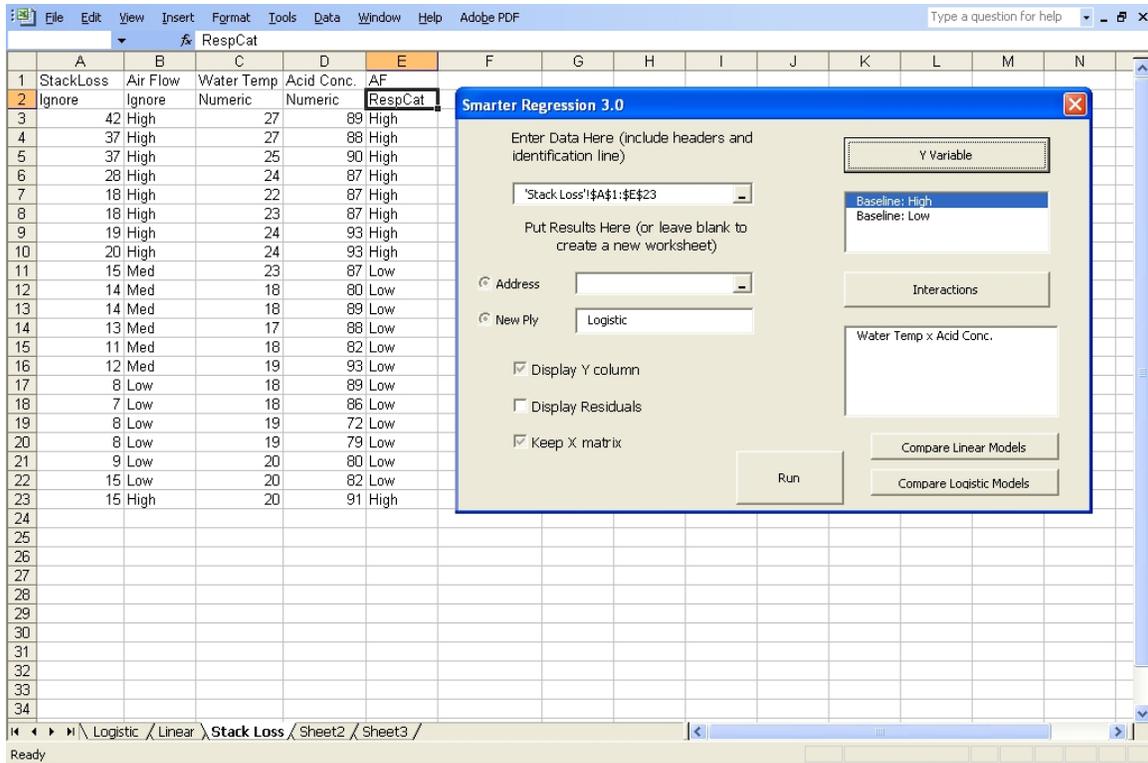


Figure 3: Smarter Regression Window for Logistic Regression

Figure 4 shows the output from the logistic regression. The X matrix is retained, and the deviance, rather than the residual sum of squares (or SSE), is displayed underneath the set of coefficients. Other than that, the two displays look very much alike, reinforcing the understanding that these two regression models are very similar. A few extra columns are visible to the right of the spreadsheet in Figure 4; these contain the computations needed to construct the log-likelihood and will not generally be of interest to the student.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	Name	Coef	SE	t stat	p-value	Y	Intercept	Water Temp	Acid Conc	Logit	Prob	Contrib	Total		
2	Intercept	64.49152	50.08464	1.287651	0.214178		0	1	27	89	-6.8926	0.001014	-0.00101	-3.91036	
3	Water Temp	-1.14711	0.641841	-1.78721	0.090755		0	1	27	88	-6.43853	0.001596	-0.0016		
4	Acid Conc.	-0.45407	0.449591	-1.00996	0.325897		0	1	25	90	-5.05246	0.006353	-0.00637		
5	Deviance						0	1	24	87	-2.54314	0.072889	-0.07568		
6	7.820714591						0	1	22	87	-0.24893	0.438088	-0.57641		
7							0	1	23	87	-1.39603	0.198446	-0.2212		
8							0	1	24	93	-5.26756	0.00513	-0.00514		
9							0	1	24	93	-5.26756	0.00513	-0.00514		
10							1	1	23	87	-1.39603	0.198446	-1.61724		
11							1	1	18	80	7.517989	0.999457	-0.00054		
12							1	1	18	89	3.431356	0.96867	-0.03183		
13							1	1	17	88	5.032533	0.99352	-0.0065		
14							1	1	18	82	6.609848	0.998655	-0.00135		
15							1	1	19	93	0.467969	0.614903	-0.48629		
16							1	1	18	89	3.431356	0.96867	-0.03183		
17							1	1	18	86	4.793567	0.991785	-0.00825		
18							1	1	19	72	10.00345	0.999955	-4.5E-05		
19							1	1	19	79	6.824953	0.998915	-0.00109		
20							1	1	20	80	5.223777	0.994642	-0.00537		
21							1	1	20	82	4.315636	0.986818	-0.01327		
22							0	1	20	91	0.229004	0.557002	-0.81419		
23															
24															
25															
26															
27															
28															
29															
30															
31															
32															
33															
34															

Figure 4: Example of Logistic Regression Output

2.3 Nested Model Tests

The other two windows in the system perform nested model F and χ^2 tests. In this example, we created a second sheet, named “Linear2,” by including the Air Flow/Water Temp. interaction in the regression (see the highlight in the interaction box). Then, after the computation was completed, pressing “Compare Linear Models” and typing in the names of the two sheets gave rise to Figure 5. In general, the user needs to enter the names of two different sheets on which linear regression results, for nested models, have been placed. It is up to him or her to ensure that the models are indeed nested, and that the Y variables have common transformations. In this case, the message reads “ F is 5.9 on 2 and 14, $p = 0.014$.”

The window for the χ^2 test for logistic regression is identical, except for the title.

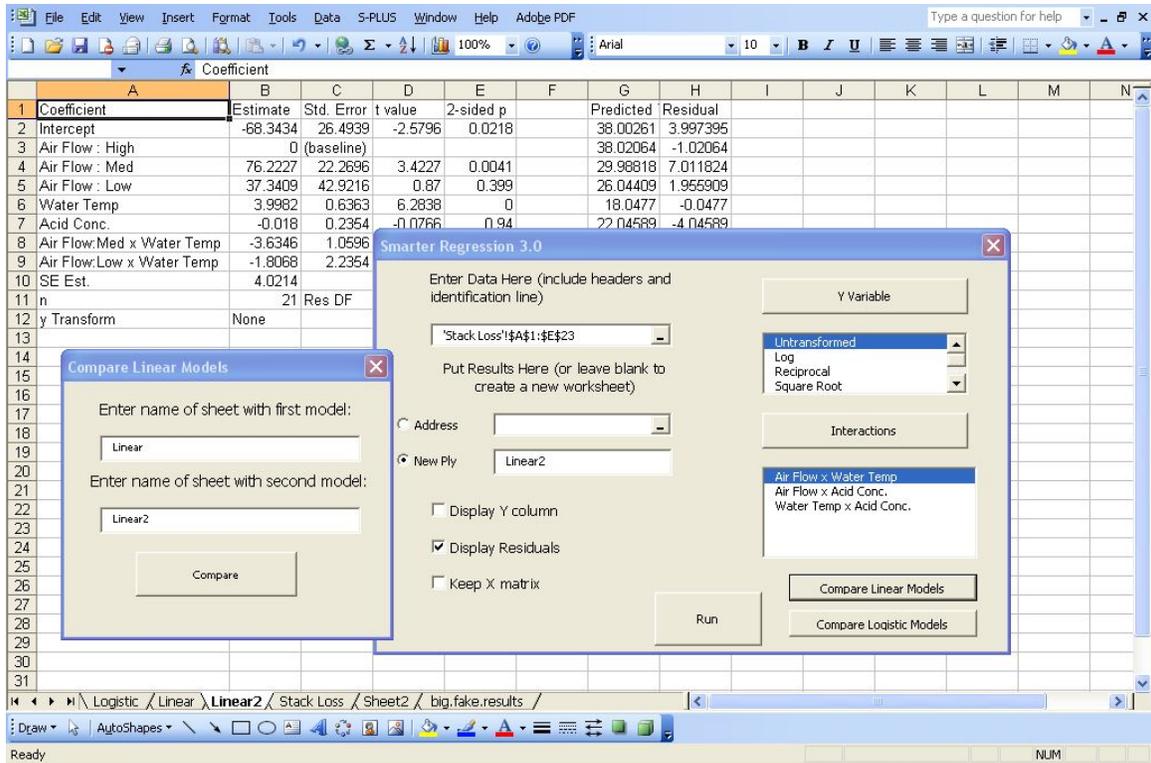


Figure 5: Nested Model Test Window

3. Computational Details

3.1 Obtaining and Installing the Add-In

The current version of the add-in can be retrieved from the author’s Web site, <http://web.nps.navy.mil/~buttrey/Software/Smarter>. The user should save the add-in to a known location on the disk and then open Excel, making sure that some worksheet (possibly a blank one) is open. From the Tools menu, the add-ins submenu should be selected. The user should then choose “Browse,” navigate to the directory holding the add-in, select that file and click on “Ok.” Users will need to ensure that Excel’s built-in Solver add-in is loaded before installing and using SR.

3.2 The Two Major Events

There are two major events that required some programming effort. The first is when the user has completed entering the range of the data, and moves elsewhere on the input window. At that point, the range’s “Exit” method is invoked. The code then establishes that there is exactly one response variable, and loads the relevant choices into the “Y” variable area (the set of transformations, if *Y* is declared to be numeric, or the two values of *Y*, if it is declared to be categorical). The program also counts the number of categories in each categorical variable, stores the set of labels, and loads the set of possible interactions into the interaction box.

The second major event, of course, is invoked when the “Run” button is pressed. There are actually three different approaches used, depending on the situation:

1. In a linear regression in which the number of columns in the \mathbf{X} matrix (after including the intercept, dummy variables, and interactions) does not exceed 16, we call Excel’s built-in `LINEST` function.
2. In a linear regression in which the \mathbf{X} matrix has 17 or more columns, we compute the results of the regression ourselves, using Excel’s matrix multiplication, dot product, and matrix inversion routines.
3. For every logistic regression, we compute the results ourselves, using Excel matrix tools and the Solver.

3.3 Computations

We make no effort to be efficient in our computations; in case 2, for example, we calculate the estimate of β as $(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$ directly, rather than by first computing the QR decomposition. We make no claim that our technique will be robust in the face of multicollinearity; our add-in is more a teaching tool than a tool for serious statistical research.

As mentioned, under case 1 above, we display the baseline values for categorical variables on a separate line. We have not implemented that for case 2 or for logistic regression.

The logistic regression computations require a little more explanation. We use Excel’s built-in Solver add-in to maximize the likelihood function as a function of the elements of \mathbf{b} , the estimate of β . We have used several steps to calculate the log-likelihood. First, we compute the logit for each observation based on the values in the current value of β . As a convenience to us, we have implemented this through Excel, by constructing a vector of `SumProduct()` calls and inserting it into the worksheet. This call computes the logits directly, as $\mathbf{X}\mathbf{b}$. (See the note in Section 3.8 regarding the drawbacks of Excel’s built-in `Mmult()` function.) Then we convert those logits to probabilities with another array-type spreadsheet formula. This formula uses the inverse of the logit transformation to compute the probabilities, setting very small and very large logits to correspond to probabilities 0 and 1 for numerical stability. Finally, a third formula computes the contribution to the log-likelihood for each observation, and a fourth formula is entered into a cell holding the total of those contributions.

We then tell the Solver to maximize that likelihood by choosing the elements of \mathbf{b} . The standard errors of those estimates can be computed from $\mathbf{X}^T\mathbf{W}\mathbf{X}^{-1}$, where \mathbf{W} is a diagonal matrix whose i^{th} entry is $\hat{p}_i(1-\hat{p}_i)$. We compute this by constructing the matrix $\mathbf{X}^* = \mathbf{W}^{1/2}\mathbf{X}$ and computing $\mathbf{X}^{*T}\mathbf{X}^*$ using Excel’s built-in matrix inversion routine.

3.4 Categorical Variables

We felt it was important to keep users from having to recode their data for use in regression. When a column is marked as categorical, then, we read and record the set of labels and construct a set of 0-1 dummy variables, using $k-1$ dummy variables for a categorical with k levels in the usual way. If the student elects to keep the \mathbf{X} matrix, we can show precisely how categorical variables are handled (in the treatment contrasts case). There is no provision for selecting the baseline level of a categorical predictor.

3.5 Interactions

The set of interactions is constructed and saved after the data reference area's "Exit" event is invoked. For every interaction selected by the user we add the proper number of columns to the \mathbf{X} matrix and fill them accordingly. We also construct reasonable labels for their coefficients and display them as needed. In Figure 4, we show the output from a logistic regression with an interaction term.

3.6 Nested Model Tests

The nested model tests rely on the sheets with results being formatted in exactly the right way. For the F test, we search for the string "SE Est." in column A. When that is found, we can locate the SSE and the number of degrees of freedom on that row, in columns B and D, respectively, and, of course, if these are found on both sheets then it is straightforward to compute the F statistic. The p -value can then be computed by a call to Excel's `FDist()` function.

In a similar way, the nested model χ^2 test is computed by locating the "Deviance" label and moving down one row. Since the label is immediately underneath the set of coefficients, the difference in the row numbers associated with the deviances is the same as the difference in degrees of freedom between the two models. The p -value is computed by Excel's `ChiDist()` function.

3.7 Computation Time

The SR add-in is fairly fast for the sort of linear regression that might be performed in Excel. For example, we constructed a $10,000 \times 10$ matrix of random numbers and an 11th column as a numeric response. The add-in computed the regression (using Excel's `Linest()` function) in under a second on a laptop running Windows XP on a 2.13 GHz processor. With 10,000 rows and a 20-column \mathbf{X} matrix, we have to compute the regression coefficients ourselves; that operation takes about five seconds on the same machine. As a final check, we computed a linear regression for a simulated data set with 60,000 rows (this being close to Excel's limit of 65,536), 25 numeric columns and three categorical columns with three levels each; that problem required about 40 seconds. In each case, the coefficients and standard errors agree with those produced by the commercial software S-Plus (Insightful, 2001) to within 6×10^{-9} .

Logistic regression is quite a bit slower, however, because Excel's Solver is not optimized for this problem. Our test example, with two predictors (plus an intercept) and 21 rows, took over five seconds on our laptop. However, the add-in is still useable for larger problems: a $1,000 \times 10$ simulated problem took only about 15 seconds. Again, the results are very close to those produced by S-Plus. The differences are somewhat bigger in this example: the largest difference was around 2×10^{-6} . This is presumably because the iterative algorithms used by Solver and by S-Plus have different stopping criteria.

3.8 Notes on Computation

Some versions of Excel do not allow matrix multiplications performed with the `Mmult()` function that produce more than 5,460 cells, nor matrix inversion with `Minverse()` of a matrix larger than 52×52 (Microsoft, 2004). The former restriction was onerous enough that we removed calls to `Mmult()` from the linear regression portion of the add-in, and replaced them with multiple calls to the dot product function `SumProduct()`. Since we were unwilling to write our own matrix inversion routine, we accept that our add-in will not handle \mathbf{X} matrices with more than 52 columns. For logistic regression we continue to use the `Mmult()` function, reasoning that no logistic regression chosen for demonstration purposes will require a matrix multiplication using more than 5,460 cells.

There are at least two different ways of entering results into workbook cells in Visual Basic. One is to establish an array in Visual Basic, fill up the elements of the array with results, and then place the array whole into the workbook. The other is to place a formula or an array formula into the workbook and then rely on Excel to compute the cell values. We have used both, choosing convenience over consistency.

4. Conclusion and Future Development Efforts

4.1 Conclusion

We have constructed an Excel add-in to make regression very much easier than in native Excel. Our add-in performs both linear and logistic regression and allows categorical and numeric predictors, as well as two-way interactions. Unlike Excel's native regression, we do not require all predictors be adjacent on the sheet, and we allow users to perform nested model tests to compare two regressions. It is also easy to transform the response variable in a linear regression.

4.2 Future Development

A few items will be cleared up in future releases. Although it is possible to place regression output somewhere other than a new sheet, it is less desirable from a user's standpoint, because the nested model test is then unavailable. Furthermore, there are fewer choices than the SR window seems to suggest: the \mathbf{X} matrix must be retained for logistic regression. However, if it is retained in a linear regression, it is saved on a sheet named "deleteme," which may not be optimal.

Right now the user needs to create any graphs of interest, for example a qqplot of the residuals or a graph of residuals versus predicted values. Excel's built-in regression procedure can produce some graphs (although its "Normal Probability Plot" option produces a normal probability plot of the responses, not the residuals, and so is essentially useless). We have tried to avoid the temptation to add features to our add-in, preferring to redirect our students to more robust tools, but if there were to be real users of the add-in, we might want to see some graphics capability added.

Bibliography

- Brownlee, K.A., 1965, *Statistical Theory and Methodology in Science and Engineering*, John Wiley and Sons, New York, NY, p. 454.
- Insightful Corp., 2001, *S-Plus 6 for Windows User's Guide*, Insightful Corporation, Seattle, WA.
- Levine, D., Berenson, M., and Stephan, D., 1998, *Statistics for Managers Using Microsoft Excel*, Prentice-Hall, Upper Saddle River, NJ.
- Microsoft Corp., 2004, Knowledge Base article 166342, <http://www.support.microsoft.com/?kbid=166342> (last accessed April 4, 2007).
- McCullough, B.D. and Wilson, B., 1999, "On the Accuracy of Statistical Procedures in Microsoft Excel 97," *Computational Statistics and Data Analysis*, 31, pp. 27-37.
- Walkenbach, J., 1999, *Microsoft Excel 2000 Power Programming With VBA*, IDG Books, Foster City, CA.

INITIAL DISTRIBUTION LIST

1. Research Office (Code 09).....1
Naval Postgraduate School
Monterey, CA 93943-5000
2. Dudley Knox Library (Code 013).....2
Naval Postgraduate School
Monterey, CA 93943-5002
3. Defense Technical Information Center2
8725 John J. Kingman Rd., STE 0944
Ft. Belvoir, VA 22060-6218
4. Richard Mastowski (Technical Editor).....2
Graduate School of Operational and Information Sciences (GSOIS)
Naval Postgraduate School
Monterey, CA 93943-5219
5. Associate Professor Samuel E. Buttrey/Code Sb.....1
Graduate School of Operational and Information Sciences
Operations Research Department
Naval Postgraduate School
Monterey, CA 93943-5219
6. Professor George Thomas.....1
Graduate School of Business and Public Policy
Naval Postgraduate School
Monterey, CA 93943-5219