Spreadsheet Addiction

Patrick Burns

Some people will think that the "addiction" in the title is over the top, or at least used metaphorically. It is used literally, and is not an exaggeration. Addiction is the persistent use of a substance where that use is detrimental to the user. It is not the substance that is the problem -- more limited use may be beneficial. It is the extent and circumstances of the use that determine if the behavior is addictive or not.

Spreadsheets are a wonderful invention. They are an excellent tool for what they are good at. The problem is that they are often stretched far beyond their home territory. The overuse of spreadsheets is only too common. The recent focus on operational risk may be one impetus for breaking some of the addiction.

I know there are many spreadsheets in financial companies that take all night to compute. These are complicated and commonly fail. When such spreadsheets are replaced by code more suited to the task, it is not unusual for the computation time to be cut to a few minutes and the process much easier to understand.

The technology acceptance model holds that there are two main factors that determine the uptake of a technology: the perceived usefulness and the perceived ease-of-use. Perception need not correspond to reality.

The perception of the ease-of-use of spreadsheets is to some extent an illusion. It is dead easy to get an answer from a spreadsheet, however, it is not necessarily easy to get the right answer. Thus the distorted view.

The difficulty of using alternatives to spreadsheets is overestimated by many people. Safety features can give the appearance of difficulty when in fact these are an aid.

The hard way looks easy, the easy way looks hard.

The remainder of this page is divided into the sections:

- Spreadsheet Computation
- The Treatment Center (Alternatives)
- If You Must Persist
- Specific Problems with Excel
- Additional Link

Spreadsheet Computation

The goal of computing is not to get an answer, but to get the correct answer. Often a wrong answer is much worse than no answer at all. There are a number of features of spreadsheets that present a challenge to error-free computing.

Ambiguity of Value and Formula

A key feature of spreadsheets is that a cell can be both a formula and a value. This is the great strength of spreadsheets. When something is made simple enough, it often becomes very powerful -- this is such a case.

While this double meaning of cells gives spreadsheets their appeal, it also has negative qualities. Primarily the problem is that some cells have hidden meaning. When you see a number in a cell, you don't know if that is a pure number or a number that is derived from a formula in the cell. While this distinction is usually immaterial, it can be critical.

The leading example is sorting. When rows are sorted, usually it is desired to sort the numbers as they are. However, it will be the formulas that are sorted. To humans the numbers are the primary thing, and the formulas are just something in the background. To the spreadsheet the formulas are primary, and the numbers are mere results.

Exercise: create a column of calls to a random function ("=rand()" in Excel). Now sort those cells.

This clash of viewpoints can end in tears. If the double meaning always made a difference, then errors on account of this would probably be quite rare. Since usually it doesn't matter, errors due to this can go unnoticed. (There is no guarantee that the errors will always be as obvious as they usually are from sorting.)

Data Extent

It is extremely common for data to be added to a spreadsheet after it has been created. The augmentation of data can go wrong, rendering a correct spreadsheet incorrect.

For example, the new data may not end up in the range of some formulas.

Data Checking

The evaluation of a formula either works (in which case a valid value is put into the cell), or it fails (resulting in an error value in the cell). Given the nature of spreadsheets (and humans), there is a tendency to favor few errors. Hence, for example, the convention of zero values for strings in numerical functions.

Good results depend on the user knowing that some of the cells are different than others and taking into account how they are treated. In practice this is hard to guarantee.

There should at least be a warning mechanism that can be inspected. The warning (in this example) would give a count of the unexpected types of data and state how they were treated.

Data Structure

When the data for an application naturally have the structure that spreadsheets impose, then correct results are likely. If the data have a more complex structure, more errors can be expected. Complex data (for example, statistical or mathematical structures) demand a convention for placement of the components. In practice the most common convention is higgledy-piggledy.

Command History

A history of the actions that were taken to achieve a result can be used to help verify the quality of the result. If the result is not right, the history can shed light on what went wrong. It seems largely impossible to get a serviceable history from the use of a spreadsheet (though some attempts are being made).

Computational Volatility

In general it doesn't make any difference how many times a formula is evaluated. Usually the only thing of interest is that the computation of the spreadsheet is complete. There are a few cases, though, where the number of computations is significant.

You may come to appreciate this if you try to sort random numbers (with their formulas intact). Applications involving random numbers are not suited to spreadsheets. In order to be able to verify results, random operations should be reproducible. It is extremely hard to assure reproducibility in spreadsheets.

Summary

When there is separation between functions and data, it is possible to refine the functions so that they work on all data of any size. In spreadsheets, where there is no such separation, it is easy for bugs to creep into the calculation on any use of the spreadsheet.

Spreadsheets are appropriate when both the data and the computations are simple. The chance of errors grows dramatically with the complexity of the spreadsheet.

The Treatment Center (Alternatives)

There are three major uses that spreadsheets are put to: as a database, as a computational engine, and as a graphics engine.

Database

Many organizations have at least begun to rationalize the use of spreadsheets as databases. Relational databases provide a more structured and secure environment in which to hold data. What may be called vectorized databases, such as Xenomorph and Kdb, are even more powerful -- though seldom would these replace spreadsheet operations. A variety of database programs exist, both commercial and open-source.

Computation

Perhaps the best alternative to spreadsheets for their computational function is the R language. R is a version of the S language that was first created at Bell Labs. Though this is often thought of as just being for statistics, that is not true. It was designed for computing with data -- precisely what spreadsheets are used for.

R has many advantages over spreadsheets when the computation or data structure is non-trivial. Some of its qualities are discussed in An Introduction to the S Language (which has a somewhat statistical orientation). 3.5 Reasons to Switch from Excel to R is a set of annotated slides from a talk.

R is open-source and freely available from The R Project. On Windows and several other platforms you can install and start to use R in just a few minutes. R runs on a wide variety of computers (and is remarkably similar on all of them). An introduction to using R is Some hints for the R beginner.

Many people will gasp in horror at the thought of using a programming language instead of a spreadsheet. The fact is that a spreadsheet is a programming language -- it is just one that you are used to. There will assuredly be a period of learning and frustration. However, that period may well be shorter and less traumatic than you imagine. The gains in productivity can be very large quickly.

Graphics

There is no reason why a spreadsheet can not produce good graphics. Such is not the case with Excel however -- more on this is given below. If the spreadsheet that you are using does not produce useful and pleasant-looking graphics, then you can use R for this task as well. The Charts & Graphs blog discusses moving from Excel to R.

If You Must Persist

If you must persist in using spreadsheets, then you should observe good spreadsheet practice and you should have a formal process to verify the quality of your spreadsheets.

Some sources to get you pointed in that direction are:

Spreadsheet Research

Stop the Subversive Spreadsheet which appears on the website of the European Spreadsheet Risks Interest Group (EuSpRIG).

Producers of spreadsheets could take some actions to help prevent bugs. For example, they could create visual cues that clearly distinguish pure values from values that depend on a formula, and add better visual cues for different forms of data.

Specific Problems with Excel

Currently the most commonly used spreadsheet is Microsoft Excel. It didn't get to be most popular because of its stunning superiority over its competitors. It was an accident of birth.

Actually Excel is known for its many faults, and the lack of response from Microsoft. Some (Americans) will remember the motto that the early Saturday Night Live created for the phone company -- "We don't care, we don't have to".

Below is a collection of problems with Excel and/or Works. Although it is probably common to assume that Works Spreadsheet is the same as Excel, that is not the case -- they apparently have different code underlying them. Some of these issues may apply to other spreadsheets as well.

Numerical Bugs

You might be forgiven for thinking that the world's largest software company could have stringent quality assurance for the numerical accuracy of the world's most heavily used spreadsheet.

Early versions of Excel 2007 when given the formula "= 850 * 77.1" returned 100,000. Sort of. Some times it acted like the real answer of 65,535. See the group discussion. The problem turned out to be how the answer was printed. It is conceivable how that bug got past quality assurance. However, another bug (discused below) regarding random nummbers suggests that the quality assurance process is severely lacking.

Writing ASCII Files

If you write a text file (csv or txt) from Excel or Works, then numbers will be written with a limited number of significant digits. Microsoft is aware that there is displeasure about this, but regards it as a "feature" -- it isn't clear what they think the up-side is of the feature. (Excel bugs don't exist, but Excel has a lot of features.)

The usual way to insure that all of the digits are retained in the file is to make sure that no numbers are written in scientific notation and the numbers are displayed in the spreadsheet with as many digits as you desire. (Yes, what is written to the file depends on what is displayed in the spreadsheet at the time, but it is not quite WYSIWYG.) Strings appear to be written out in full even if they are not fully displayed in the spreadsheet.

There is no telling how much damage has been done because precision has been lost, mostly unknowingly. There is a trick to get full precision, however -- turn the numbers into text. One way of doing this is to use the concatenate function with just one argument.

This loss of precision is especially troublesome since it limits the use of Excel as a staging area for data -- gathering data from one or more sources and writing the data out to be used by other programs. Spreadsheets are often a very handy tool for this sort of operation.

Propagation of Blanks

Once upon a time Excel considered any blank (or non-numeric) cell within the range of a numeric function to be zero. At some point this was changed so that blank cells were ignored entirely. This change was a good thing (apparently).

However, if a blank cell is referenced, then it is treated as zero in the reference cell. Exercise: in the first few rows of column A put some numbers but include at least one blank cell; below these numbers take the average of the cells; in cell B1 write the formula =A1; copy the formula down. You will get two different means.

It is not clear whether it is better to have proper behavior sometimes, or consistently wrong behavior. Apparently understanding what happens with blanks is a subject in itself.

Range Changes

Exercise: Put numbers in the first three rows of column A; in the fifth row of column A put the formula =SUM(A1:A3). You will get the sum of the three numbers. Now put a number in the fourth row of column A. In at least some versions of Excel the formula will magically change and you get the sum of the four numbers. There are a few things wrong here.

This is action at a distance. To the uninitiated it is as if a cosmic ray suddenly transformed the spreadsheet. That is, if they notice at all.

It is decidedly bad that the behavior of a very simple operation on a spreadsheet can not be predicted with certainty -- especially so for versions of the same spreadsheet product.

There is an option to turn off automatic range extension in the versions of Excel where it exists. At first blush this seems like a good thing, but really it only makes things worse. It means that it can bite you both ways. Ranges may be extended when you don't expect them to, and they may not be extended when you are expecting them to be. The ability to guarantee that the spreadsheet works as intended is closer to impossible.

Bad as this situation is, some sympathy for Microsoft should be extended here -- they are merely struggling to do their best in a tough medium. The reason to introduce automatic range extension is that a common bug in spreadsheets is for new data to be added to the sheet but not included in the summaries. (I've been nipped by this myself, and almost lost a couple days of pay from it.)

Typing Mistakes

Typing errors will occur when data are entered. Let's look at likely miskeying of the decimal point when entering numbers. (This assumes that the decimal place is marked by a period -- those who use a comma to indicate the decimal place may want to experiment for themselves.)

If a comma is typed instead of a period, (some?) versions of Excel consider the result to be a string (in which case its numerical value is 0), while (some?) versions of Works throw the comma away -- so what is intended to be 3.5 but typed as 3,5 becomes 35. Cool.

If a slash is typed instead of a period, then the result may become a date in which case its numerical value is probably large. Alternatively it can be a string -- giving a value of 0.

Excel does have the good quality that strings are left-justified and numbers are right-justified. This means that if the column is wide enough, then strings are distinguishable by sight from numbers. This is not good enough, however, as there have been cases of experienced users not spotting the difference even though they were looking and knew that something was wrong in the spreadsheet.

Unary Minus

Unary minus has peculiar properties.

Exercise: put the value 3 into cell A1; in another cell put the formula =-A1^2; in a third cell put the formula =-3^2. Most people will intuitively expect to get -9 as the answer in the latter two cells. Excel gives you 9 for these cells. That is, the precedence of unary minus is different than many other languages (including VBA which is often used with Excel).

Microsoft Works version 7.0 gives -9 for =-A1^2 and 9 for =-3^2. It gives -9 for =-(3)^2.

So now we have that -3^2 is different than -(3)^2. Virtually all languages would assert that in both cases there are two operators (unary minus and exponentiation) at work on a constant. One of the operators will take precedence (most likely exponentiation) and be performed first.

Since with the second formula Works gets the answer that implies that exponentiation was done first, it is not the operator precedence that is unusual. In order for Works to get the behavior that it does, it has to consider the minus as part of the constant in the first case. That is, it is not the operator precedence that is unusual, it is the parsing.

The behavior of Works can be explained by augmenting the precedence table with an additional operator. There is negation of a constant with precedence higher than exponentiation, and there is unary minus with precedence lower than exponentiation.

Circular References

While circular references are possible in other languages, their occurrence is particularly easy in spreadsheets. Exercise: Put numbers in the first three rows of column A, in cell A4 put "=sum(A1:A4)". Note that one of the numbers being summed is the cell where the formula is.

Excel and Works have different behavior for this. Excel says in a popup window that it can not do the computation, and sometimes (but not always) starts up a tool to help you trace the circularity. Although it says that it can not do the computation, it puts a zero in the cell. This is wrong -- zero is not a proper answer. It is quite curious that zero should be given since Excel displays error values for other problems. However, it really is only the display that is wrong since adding 20 to the circular reference (in a different cell) still results in zero being printed (that is, it can not be zero in the usual sense).

Works does even worse. It also displays a popup window with an error message, but that window seems much easier to ignore. The cell containing the circular reference gets a value, and that value changes on each computation of the spreadsheet.

Both Excel and Works have the good grace to warn of circular references when a spreadsheet is opened. However, Works doesn't give any hints about how to find the problem.

The Beginning Of Time

The world according to Microsoft was created on January first 1900 plus or minus a day. Dates before 1900 are not allowed.

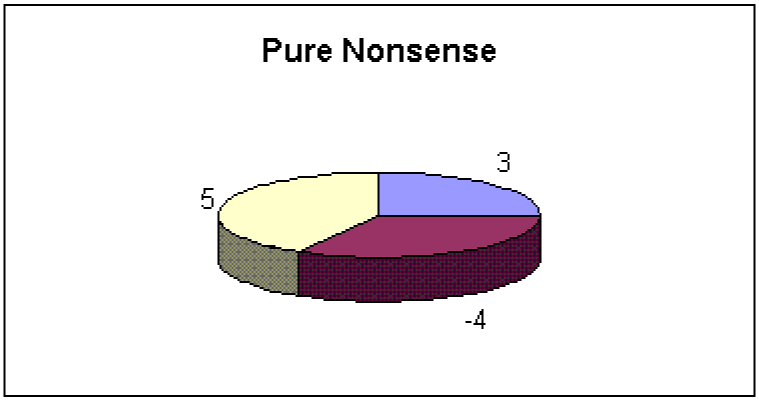
Microsoft made the year 1900 a leap year (even though in reality it is not) because another spreadsheet had mistakingly made the error. In order to maintain compatibility with the competing spreadsheet, they reproduced the bug. (They seem to have maintained the attitude that bugs should be replicated.)

There is more on dates below.

Bad Graphics

The quality of graphics that are produced by Excel is astoundingly amateurish. Graphics meant for presentation should be created elsewhere, for example in R.

Our entry for the ultimate bad graphic is "Pure Nonsense" presented here.



What's wrong? Let us count the ways.

First off, a pie chart is seldom the best graph to use. The way that information is coded in pie charts makes it hard for us (humans) to visually decode the information. See, for example, Frank Harrell's Statistical Graphics and its references. Another interesting site on graphics is the Gallery of Data Visualization.

Much, much worse is the three-dimensional effect. In most graphics a three-dimensional effect merely makes the graph harder to interpret. In this chart the three-dimensional effect entirely destroys the coding for the information. In a standard pie chart the information is encoded in the angles and areas. With the three-dimensional effect we don't know if it is the actual angle or the implied angle or the actual area or the implied area that holds the information. You really have to question the sanity of anyone who would think this is a good idea.

But wait, there's more ...

Notice that one of the numbers is negative. Pie charts only make sense if all of the numbers are positive. Negative numbers don't slow down Excel any -- it just goes ahead and gives you a graph with data it makes up for itself (it is taking the absolute value).

Poor Statistics

Serious problems with statistical functionality in Excel have been well-known within the statistics profession for some time. One of the better known expositions is Jonathan Cryer's 2001 JSM Talk. Even if Microsoft fixes all of the implementation problems identified in this, the final graphic (which is well worth the price of a few clicks) will stand.

Jeffrey Simonoff's Statistical analysis using Microsoft Excel contains an amazing collection of errors. To quote someone famous -- you can't make up stuff like this.

On the other hand, to the extent that Excel's poor statistical functionality has driven people to other programs, it has been a service. A spreadsheet is not a proper place to perform statistical analyses except in the very simplest of cases.

After years of inaction Microsoft made some changes to statistical functionality for Excel 2003. (Fixes for bugs in R are often available within a week of the report.) Microsoft's description of improvements indicates some of the things that were broken and some things that are still broken in Excel 2003. Another Microsoft support description has some more information.

While it seems that there is general agreement that Excel 2003 is better, it is probably not surprising that some of the fixes are less than perfect. B. D. McCullough and Berry Wilson in the journal article "On the accuracy of statistical procedures in Microsoft Excel 2003" cover a range of areas -- some fixes they find to be adequate, others not. Leo Knuesel's "On the accuracy of statistical distributions in Microsoft Excel 2003" shows improvements to statistical distributions but some tail probabilities that were exact in earlier versions of Excel are now rounded to zero. There were also missing value problems in the improved SLOPE, INTERCEPT and similar functions (Microsoft's description).

One of the things that was changed for Excel 2003 was the random number generator (Microsoft's description). They switched to a generator that passes the Diehard test suite for random number generators. However, they obviously didn't test to see if their implementation passed the test suite since the initial release would generate negative numbers. There are numerous discussions of this bug, for example this item from Woody's Watch. (McCullough and Wilson in "On the accuracy of statistical procedures in Microsoft Excel 2003" question the use of this generator in the first place.)

Improvements to the random number generator seem largely irrelevant to me. The old generator was probably perfectly adequate for the mundane uses of random numbers that are appropriate in spreadsheets. If an application needs to worry about the quality of the random number generator, then it shouldn't be performed in a spreadsheet anyway.

Unhelpful Help

Like much documentation elsewhere Microsoft documentation is often found to be inadequate. A common complaint about documentation is that it is incomprehensible. What is unique about Microsoft is its high proportion of documentation that is perfectly understandable, but wrong.

Look, for example, at the Microsoft Knowledge Base entry How to correct rounding errors in floating-point arithmetic. This page has "Symptoms", "Cause" and two methods of "Workaround". The Symptoms section clearly and concisely states the problem (well done). The statement in the Cause section is slightly (well, very) strange, but technically factual. The first method of workaround is good -- if you know that your answer has only two decimal places, then round the answer to two decimal places.

The second workaround -- use precision as displayed -- makes the problem worse. The problem is that floating point numbers are stored with a finite number of bits. Using the displayed precision effectively throws some of the bits away. How could that possibly be an improvement? (If all of the intermediate results have a known precision -- for example are all integer, then the proposal could help. The page now hints at this, and admits that it can compound the problem.)

The Other Unhelpful Help

At times Excel attempts to do what you want instead of what you say. When it doesn't get it right, then the results can be much worse than if it hadn't attempted to be clever. The automatic range extension, discussed above, is one example.

A story along similar lines was related by one correspondent. A multinational collaboration created a large spreadsheet that contained dates that had been agreed upon to use the European convention dd/mm/yyyy. Excel interpreted (wrongly) some of the dates to be in American format mm/dd/yyyy and the rest to be strings. The columns were not wide enough for these to be easily distinguished. Once the problem was found, their attempts to fix the problem within Excel only made things worse. It was finally fixed via a sophisticated tool-set outside of Excel.

There is a gene called SEP7. When Excel sees this, it assumes it is a date and turns it into a number representing the number of days since some particular day. Very helpful.

No Database of Bugs

It is virtually impossible to perform proper quality assurance when there is not a database of fixed and outstanding bugs. QA for spreadsheets is hard enough without this added handicap in Excel.

Excel ODBC DLL Problem

This is quite a technical problem. The ODBC DLL that reads Excel spreadsheets has some problems which means that it is not always possible to get data in its proper form when using this method. See R Help topic: Missing value representation in Excel.

Limited Dimensions

It is mind-boggling that a spreadsheet that limits the number of columns to 256 would ever have been taken seriously in the 21st century.

Additional Links

Systems Modelling Ltd.: Spreadsheet resources
http://www.sysmod.com/sslinks.htm

Systems Modelling Ltd.: PraxIS newsletter
http://www.sysmod.com/praxis/index.htm

Louise Pryor's list of topics in risk management, spreadsheet errors, software development, etc.
http://www.louisepryor.com/showThemes.do

Uncovering Effects of Programming Paradigms: Errors in Two Spreadsheet Systems by Markku Tukiainen
http://www.ppig.org/papers/12th-tukiainen.pdf

http://slack.ser.man.ac.uk/progs/stata/avoid_excel.html

Northview Biosciences: A practical approach to spreadsheet validations in the cGxP Environment (2002)
http://www.northviewlabs.com/Spreadsheet_Validation.htm

Use of Excel for Statistical Analysis by Neil Cox (2000). Note that many links elsewhere on the web are wrong for this.
http://www.agresearch.co.nz/Science/Statistics/exceluse1.htm

Fixing Statistical Errors in Spreadsheet Software: The Cases of Gnumeric and Excel by B. D. McCullough
http://www.csdassn.org/software_reports/gnumeric.pdf

Testing spreadsheets and other packages used in metrology by H. R. Cook, M. G. Cox, M. P. Dainton and P. M. Harris (1999)
http://www.npl.co.uk/ssfm/download/documents/cise27_99.pdf

Is it practical to use Excel for stats?
http://www.practicalstats.com/Pages/excelstats.html

Vanderbilt Medical Center TWiki entry on Excel Problems
http://biostat.mc.vanderbilt.edu/twiki/bin/view/Main/ExcelProblems

A specific instance of curing spreadsheet addiction: "The Harvest Rate Model for Klamath River Fall

Chinook Salmon, with Management Applications and Comments on Model Development and Documentation" (2001)
http://www.sefsc.noaa.gov/mprager/reports/Prager_Mohr_NAJFM_2001.pdf

On the Accuracy of Statistical Distributions in Microsoft Excel 97 by Leo Knuesel
http://www.stat.uni-muenchen.de/~knuesel/elv/excelacc.pdf

Statistical flaws in Excel by Hans Pottel
http://www.mis.coventry.ac.uk/~nhunt/pottel.pdf

Calculating Poisson confidence intervals in Excel by Iain Buchan (2004)
http://www.nwpho.org.uk/sadb/Poisson%20CI%20in%20spreadsheets.pdf

Software for uniform random number generation: Distinguishing the good from the bad by Pierre L'Ecuyer
http://www.iro.umontreal.ca/~lecuyer/myftp/papers/wsc01rng.pdf

Assessing the Reliability of Statistical Software: Part I by B. D. McCullough (1998)
http://www.amstat.org/publications/tas/mccull-1.pdf

Assessing the Reliability of Statistical Software: Part II by B. D. McCullough (1999)
http://www.amstat.org/publications/tas/mccull.pdf

Bloor Research on spreadsheet fraud (2005)
http://www.theregister.co.uk/2005/04/22/managing_spreadsheet_fraud/

End Users Shaping Effective Software (EUSES)
http://eecs.oregonstate.edu/EUSES/

Calculating Poisson confidence intervals in Excel

Verity Stob's Thirteen ways to loathe VB (2000)
http://www.ddj.com/documents/s=1503/ddj0001vs/jan00.htm

"On the Numerical Accuracy of Spreadsheets" by Almiron, et al. in the April 2010 edition of the Journal of Statistical Software:
http://www.jstatsoft.org/v34/i04/paper

Go to Burns Statistics Home.
Direct access to this page is
http://www.burns-stat.com/pages/Tutor/spreadsheet_addiction.html

First Version: 2005 January 02
Last Modified: 2010 September 29