

Statistical flaws in Excel

Hans Pottel

Innogenetics NV, Technologiepark 6, 9052 Zwijnaarde, Belgium

Introduction

In 1980, our ENBIS Newsletter Editor, Tony Greenfield, published a paper on ‘Statistical Computing for Business and Industry’. In that paper, he came to the conclusion that the programmable calculators used at that time were unknowingly threatening to inflict bad decisions on business, industry and society through their bland acceptance of incorrect machine-based calculations. Today, 23 years later, everybody will agree that there has been a revolution in computerscience, leading to very sophisticated computers and improved algorithms. The type of calculations Tony has been discussing in 1980 are nowadays very often done with the Commercial Off-the-Shelf (COTS) software package Microsoft Excel, which is very widespread, for various reasons:

- Its integration within the Microsoft Office suite
- The wide range of intrinsic functions available
- The convenience of its graphical user-interface
- Its general usability enabling results to be generated quickly

It is accepted that spreadsheets are a useful and popular tool for processing and presenting data. In fact, Microsoft Excel spreadsheets have become somewhat of a standard for data storage, at least for smaller data sets. This, along with the previously mentioned advantages and the fact that the program is often being packaged with new computers, which increases its easy availability, naturally encourages its use for statistical analysis. However, many statisticians find this unfortunate, since Excel is clearly not a statistical package. There is no doubt about that, and Excel has never claimed to be one. But one should face the facts that due to its easy availability many people, including professional statisticians, use Excel, even on a daily basis, for quick and easy statistical calculations. Therefore, it is important to know the flaws in Excel, which, unfortunately, still exist today. This text gives an overview of known statistical flaws in Excel, based on what could be found in the literature, the internet, and my own experience.

General remarks

Excel is clearly not an adequate statistics package because many statistical methods are simply not available. This lack of functionality makes it difficult to use it for more than computing summary statistics and simple linear regression and hypothesis testing.

Although each Microsoft Excel worksheet function is provided with a help-file that indicates the purpose of that function, including descriptions of the inputs, outputs and optional arguments required by the routine, no information about the nature of the numerical algorithms employed is generally provided or could be found. This is most unfortunate as it might help detect why numerical accuracy might be endangered or why – in some cases - a completely wrong result is obtained.

Another important remark is that although many people have voiced their concerns about the quality of Excel’s statistical computing, nothing has changed. Microsoft has never responded to comments on this issue. Consequently, the statistical flaws reported in Excel 97 worksheet functions and the Analysis Toolpak are still present in Excel 2000 and Excel XP. This, of course, is most unfortunate.

My overall assessment is that while Excel uses algorithms that are not robust and can lead to errors in extreme cases, the errors are very unlikely to arise in typical scientific data analysis. However, I would not advise data analysis in Excel if the final results could have a serious impact on business results, or on the health of patients. For students, it's my personal belief that the advantages of easy-to-use functions and tools counterbalance the need for extreme precision.

Numerical accuracy

Although the numerical accuracy is acceptable for most of Excel's built-in functions and for the tools in the Analysis Toolpak when applied to "easy" data sets, for "not-so-easy" data sets this may be no longer true.

The numerical performance of some of Excel's built-in functions can be poor, with results accurate to only a small number of significant figures for certain data sets. This can be caused by the use of a mathematical formula (as in the STDEV worksheet function) or a model parametrization (as in the LINEST and TREND worksheet functions) that exacerbates the natural ill-conditioning of the problem to be solved, i.e., leads to results that are not as accurate as those that would be returned by alternative stable algorithms. Alternatively, the poor performance can be a consequence of solving a problem that approximates the one intended to be solved (as in the LOGEST and GROWTH worksheet functions).

The numerical performance of Excel's mathematical and trigonometric functions is generally good. The exception is the inverse hyperbolic sine function, ASINH, for which the algorithm used is unstable for negative values of its argument.

For Excel's statistical distributions, the numerical performance of these functions exhibits systematic behaviour, with worsening accuracy at the tails of the distributions. Consequently, these functions should be used with care.

In many instances, the reported poor numerical performance of these functions can be avoided by appropriate pre-processing of the input data. For example, in the case of the STDEV worksheet function for the sample standard deviation of a data set, the accuracy loss can be avoided by subtracting the sample mean from all the values in the data set before applying the STDEV function. Mathematically, the standard deviations of the given and shifted data sets are identical, but numerically that of the latter can be determined more reliably.

Basic descriptive statistics


The most important flaw in basic statistical functions is the way Excel calculates the standard deviation and variance. The on-line help documentation for the STDEV worksheet function makes explicit reference to the formula employed by the function. This is in contrast to many of the other functions that provide no details about the numerical algorithms or formulae used.

$$s = \sqrt{\frac{n \sum x^2 - (\sum x)^2}{n(n-1)}}$$

Unfortunately, it is well known that this formula has the property that it suffers from subtractive cancellation for data sets for which the mean \bar{x} is large compared to the standard deviation s , i.e., for which the coefficient of variation s/\bar{x} is small. Furthermore, a floating-point error analysis of the above formula has shown that the number of incorrect significant figures in the results obtained from the formula is about twice that for the mathematically equivalent form

$$s = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n-1}}$$

I'll demonstrate this by an example. I programmed an alternative User Defined Function (UDF) (the UDF is programmed in Visual Basic for Applications, the Excel macro language) for the standard deviation, which I here called STDEV_HP. This function calculates the standard deviation, based on the second formula. The method of calculation is based on centering the individual data points around the mean. This algorithm is known to be much more numerically stable.



```
Function STDEV_HP(R As Range) As Double
    Dim i As Integer
    Dim n As Integer
    Dim Avg As Double
    'n = number of observations = number of cells in range R
    n = R.Cells.Count
    'calculate the average
    Avg = 0
    For i = 1 To n
        Avg = Avg + R.Cells(i).Value
    Next i
    Avg = Avg / n
    'calculate the standard deviation
    STDEV_HP = 0
    For i = 1 To n
        STDEV_HP = STDEV_HP + (R.Cells(i).Value - Avg) ^ 2
    Next i
    STDEV_HP = Sqr(STDEV_HP / (n - 1))
End Function
```

Example:

The data set used to demonstrate the difference in accuracy between Excel's built-in function STDEV and the new UDF STDEV_HP is:

Observation	X
1	10000000001
2	10000000002
3	10000000003
4	10000000004
5	10000000005
6	10000000006
7	10000000007
8	10000000008
9	10000000009
10	10000000010
AVG	10000000005.5
STDEV	0.000000000
STDEV_HP	3.027650354

In the example, it is clear that there is variation in the X-observations, but nevertheless Excel's built-in function STDEV gives ZERO as output. This is clearly wrong. The alternative UDF STDEV_HP gives 3.027650354 as output. As shown in the UDF, an easy

way to work around this flaw is by centering the data before calculating the standard deviation, in case it is expected that s/\bar{x} is small. For this example, after centering, I obtain

Obs	X-Avg
1	-4.5
2	-3.5
3	-2.5
4	-1.5
5	-0.5
6	0.5
7	1.5
8	2.5
9	3.5
10	4.5
STDEV	3.027650354

If Excel's built-in function STDEV is applied on the centered data, you will find exactly the same result as with my User Defined Function STDEV_HP.

Excel also comes with statistical routines in the Analysis Toolpak, an add-in found separately on the Office CD. You must install the Analysis Toolpak from the CD in order to get these routines on the Tools menu (at the bottom of the Tools menu, in the Data Analysis command). Applying the Analysis Toolpak tool "Descriptive Statistics" to the small data set of 10 observations, I obtained the following output:

X	
Mean	10000000005.5
Standard Error	0
Median	10000000005.5
Mode	#N/A
Standard Deviation	0
Sample Variance	0
Kurtosis	-1.2
Skewness	0
Range	9
Minimum	10000000001
Maximum	10000000010
Sum	100000000055
Count	10
Largest(1)	10000000010
Smallest(1)	10000000001
Confidence Level(95.0%)	0

Apparently, the Analysis Toolpak applies the same algorithm to calculate the standard deviation. As the sample variance, standard error and the confidence level (95.0%) are probably derived from this miscalculated standard deviation, they are wrong too. Again, if the data are centered before I apply "Descriptive Statistics" in the Analysis Toolpak, I obtain:

X	
Mean	0
Standard Error	0.957427108
Median	0
Mode	#N/A
Standard Deviation	3.027650354
Sample Variance	9.166666667
Kurtosis	-1.2
Skewness	0
Range	9
Minimum	-4.5
Maximum	4.5
Sum	0
Count	10
Largest(1)	4.5
Smallest(1)	-4.5
Confidence Level(95.0%)	2.165852240

The correct standard deviation is obtained now. As the variance, standard deviation, standard error and confidence level are invariant for this kind of transformation (centering the data around the mean), these results are correct for the original data set.

The functions in Excel STDEV, STDEVP, STDEVA, STDEVPA, VAR, VARP, VARA, VARPA all suffer from the same poor numerical accuracy. On the other hand, the functions KURT (Kurtosis) and SKEW (skewness) apply an algorithm on centered data and do not have this flaw.

Note that the confidence level is calculated using $z_{1-\alpha/2} = \text{NORMSINV}(0.975) = 1.96$ times the standard error, which might be valid if the population variance is known or for large sample sizes, but not for small samples, where $t_{\alpha/2, n-1} = \text{TINV}(0.05, n-1)$ should be used. Note that $1-\alpha/2 = 0.975$ has to be entered in the NORMSINV function, whereas the TINV function requires the value of α . Excel is quite inconsistent in the way these functions are used.

It has been seen many times that the Analysis Toolpak makes use of the worksheet functions for its numerical algorithms. Consequently, the Analysis Toolpak tools will have the same flaws as Excel's built-in functions.

Excel also has a strange way to calculate ranks and percentiles. Excel's built-in RANK function does not take into account 'tied' ranks. For example, in a series of measurements 100, 120, 120, 125 Excel gives two times rank 2 to the value of 120 and value 125 gets the rank 4. When 'tied' ranks are taken into account, the rank of 120 should be $(2 + 3)/2 = 2.5$ and the value of 125 should indeed get rank 4. Excel assigned the lowest of the two ranks to both observations, giving each a rank of 2. Because Excel doesn't consider 'tied' ranks it is impossible to calculate the correct non-parametric statistics from the obtained ranks. For this reason I developed a User Defined Function, called 'RANKING', which takes into account 'tied' ranks.



```
Function Ranking(V As Double, R As Range) As Double
    Dim No As Integer
    Ranking = Application.WorksheetFunction.Rank(V, R, 1)
    No = Application.WorksheetFunction.CountIf(R, V)
    Ranking = Ranking + (No - 1) / 2
End Function
```

The way Excel calculates percentiles is also not the way most statistical packages calculate them. In general, the differences are most obvious in small data sets. As an example, let's take the systolic blood pressures of 10 students sorted in ascending order: 120, 125, 125, 145, 145, 150, 150, 160, 170, 175. The lower quartile (or 25% percentile) as calculated with Excel's built-in function QUARTILE (or PERCENTILE) is 130 and the upper quartile is 157.5. A statistical package, however, will give 125 and 170 as lower and upper quartile, respectively. Apparently, Excel calculates the lower quartile $130 = 125 + (145-125)*0.25$ and the upper quartile as $157.5 = 150 + (160-150)*0.75$. This is an interpolation between the values below and above the 25% or 75% observation. Normally, the *pth* percentile is obtained by first calculating the rank $l = p(n+1)/100$, rounded to the nearest integer and then taking the value that corresponds to that rank. In case of lower and upper quartiles, the ranks are $0.25*(10+1) = 2.75 \Rightarrow 3$ and $0.75*(10+1) = 8.25 \Rightarrow 8$ which corresponds to 125 and 170 resp.

Correlation and regression

Regression on difficult data sets

Let's take back my first example and add a column for the dependent variable Y. Actually this example was presented by J. Simonoff in his paper entitled "Statistical analysis using Microsoft Excel". As shown before, with this kind of data, Excel has serious problems to calculate descriptive statistics. What about regressing Y against X?

Excel has different ways of doing linear regression: (a) using its built-in function LINEST, (b) using the Analysis Toolpak tool 'Regression' and (c) adding a trendline in an XY-scatter graph. Let me start making an XY-scatter plot and try to add a trendline:

X	Y
10000000001	1000000000.000
10000000002	1000000000.000
10000000003	1000000000.900
10000000004	1000000001.100
10000000005	1000000001.010
10000000006	1000000000.990
10000000007	1000000001.100
10000000008	1000000000.999
10000000009	1000000000.000
10000000010	1000000000.001

Apparently, Excel does not have a problem displaying these kind of data (see Figure 1). Now, by right-clicking the data points in the graph, and selecting Add Trendline (with options 'display R^2 and equation on the chart'), we obtain Figure 2. It is clear that Excel fails to add the correct straight line fit. The obtained line is very far away from the data. Excel even gives a negative R-square value. I also tried out every other mathematical function available via 'Add Trendline'. With the exception of 'Moving Average', all trendlines failed to fit the data, resulting in nonsense fit results and statistics.

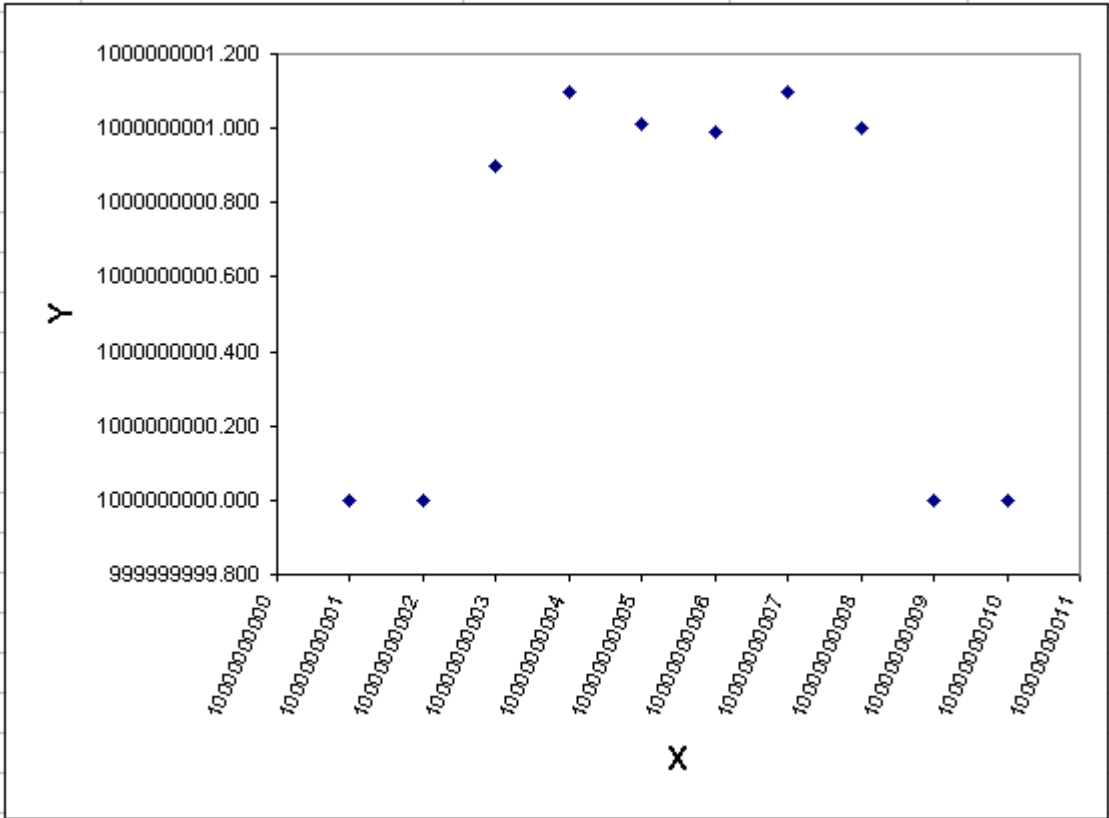


Figure 1: XY scatter graph for the J. Simonoff data set

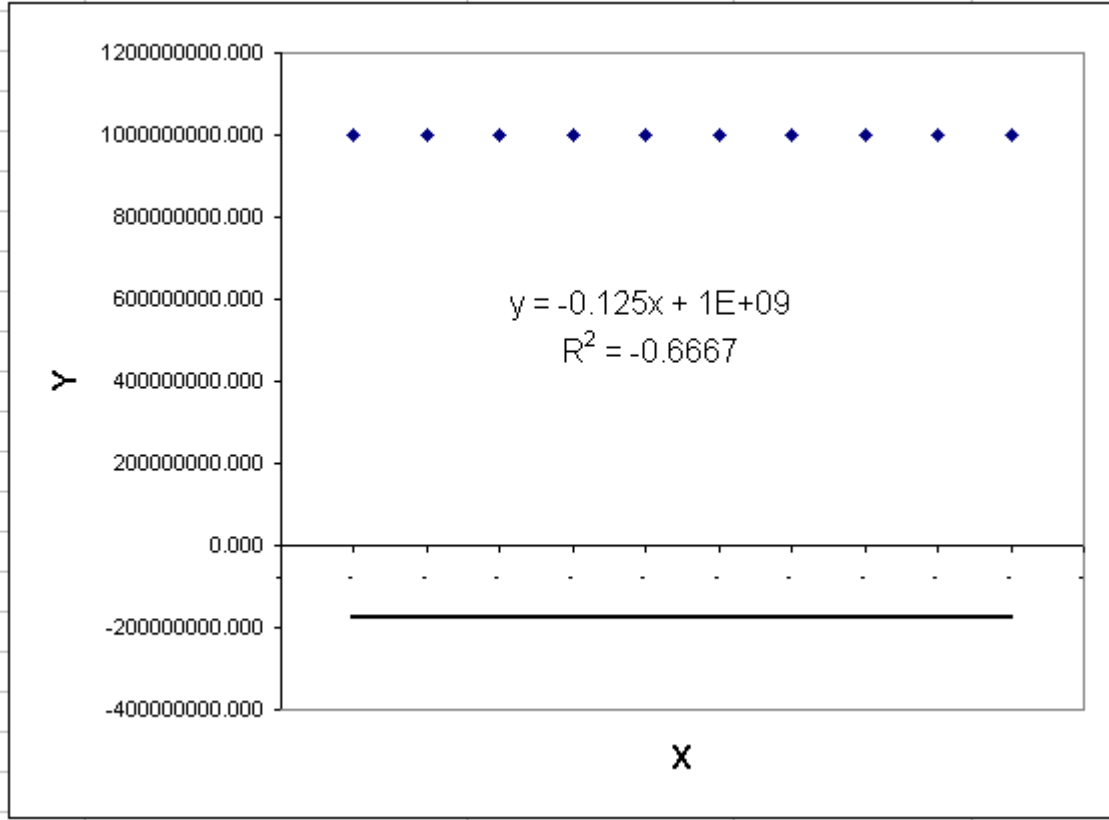


Figure 2: A trendline for the J. Simonoff example

The second way to do regression is by using the LINEST function. The LINEST function is actually an array function, which should be entered using 'CTRL+SHIFT+ENTER' to obtain the fit parameters plus statistics.

This is the output of LINEST for the example above:

-0.125	2250000001
0	0
-0.538274369	0.694331016
-2.799367289	8
-1.349562541	3.85676448

Note that in case of linear regression, the output of the LINEST functions corresponds to:

Slope	Intercept
Standard Error of Slope	Standard Error of Intercept
R-square	Standard Error of Y
F	df
SS(Regression)	SS(residual)

As you can see, the output is complete nonsense, with R-square, F, and SS(Regression) being negative. Standard errors of slope and intercept are zero, which is clearly wrong. Applying the Analysis Toolpak tool 'Regression' to the above example results in the following output:

SUMMARY OUTPUT

<i>Regression Statistics</i>	
Multiple R	65535
R Square	-0.538274369
Adjusted R Square	-0.730558665
Standard Error	0.694331016
Observations	10

ANOVA

	<i>df</i>	<i>SS</i>	<i>MS</i>	<i>F</i>	<i>Significance F</i>
Regression	1	-1.349562541	-1.3495625	-2.79936	#NUM!
Residual	8	3.85676448	0.482095		
Total	9	2.507201939			

	<i>Coefficients</i>	<i>Standard Error</i>	<i>t Stat</i>	<i>P-value</i>
Intercept	2250000001	0	65535	#NUM!
X Variable	-0.125	0	65535	#NUM!

As one can see, the same values are found with the Analysis Toolpak tool as with the LINEST worksheet function. Because a negative number is found for F and unrealistic values for t Stat, Excel is unable to calculate the corresponding p-values, resulting in the #NUM! Output.

Note that the slope is identical in the three cases (trendline, LINEST and the Analysis Toolpak), but the intercept and R-square are different when the 'Add Trendline' tool is used.

Excel also has different worksheet functions that are related to the linear regression calculation. These functions are SLOPE, INTERCEPT, TREND, etc. These functions give the same erroneous results and clearly they suffer from the application of numerically unstable algorithms.

Related to linear regression are the worksheet functions for correlation: CORREL and PEARSON and worksheet functions like STEYX. Here Excel is really surprising: CORREL gives the correct output, but PEARSON gives the result #DIV/0!. While they are actually the same, two different algorithms are used to calculate them. The worksheet function STEYX gives #N/A.

As with the calculation of the STDEV or VAR functions, the workaround is quite straightforward. By simply centering the data for X and Y around their respective means, the calculation becomes much more numerically stable and the results are correct (the negative value for the adjusted R-square is because of the very poor linear relationship between X and Y, but is correctly calculated from its definition).

SUMMARY OUTPUT

<i>Regression Statistics</i>	
Multiple R	0.016826509
R Square	0.000283131
Adjusted R Square	-0.124681477
Standard Error	0.559742359
Observations	10

Of course, due to the centering the obtained regression coefficients should be transformed back to obtain the true regression coefficients. The slope is unaffected by this transformation, but the intercept should be adjusted.

Below I have added some simple VBA code to calculate slope and intercept of a linear regression line, based on a numerically stable algorithm.

```

Sub Straight_Line_Fit()
Dim X_Values As Range
Dim Y_Values As Range
Dim Routput As Range
Dim avgx As Double, avgy As Double, SSxy As Double, SSxx As Double
Dim n As Integer, i As Integer
Dim FitSlope As Double
Dim FitIntercept As Double
Set X_Values = Application.InputBox("X Range = ", "Linear Fit", , , , , 8)
Set Y_Values = Application.InputBox("Y Range = ", "Linear Fit", , , , , 8)
Set Routput = Application.InputBox("Output Range = ", "Linear Fit", , , , , 8)
avgx = 0
avgy = 0
`number of observations
n = X_Values.Cells.Count
`averages
For i = 1 To n
    avgx = avgx + X_Values.Cells(i).Value / n

```

```

    avgy = avgy + Y_Values.Cells(i).Value / n
Next i
'sum of squares
SSxy = 0
SSxx = 0
For i = 1 To n
    SSxx = SSxx + (X_Values.Cells(i).Value - avgx) ^ 2
    SSxy = SSxy + (X_Values.Cells(i).Value - avgx) * (Y_Values.Cells(i).Value - avgy)
Next i
'slope
FitSlope = SSxy / SSxx
'intercept
FitIntercept = avgy - FitSlope * avgx
Routput.Offset(0, 0) = "Slope = "
Routput.Offset(0, 1) = FitSlope
Routput.Offset(1, 0) = "Intercept ="
Routput.Offset(1, 1) = FitIntercept
End Sub

```

Regression through the origin

Although Excel calculates the correct slope when regressing through the origin, the ANOVA table and adjusted R-square are not correct. Let me show you an example:

X	Y
3.5	24.4
4	32.1
4.5	37.1
5	40.4
5.5	43.3
6	51.4
6.5	61.9
7	66.1
7.5	77.2
8	79.2

Using the Analysis Toolpak 'Regression' tool and checking the 'Constant is Zero' checkbox, the following output is obtained:

SUMMARY OUTPUT

<i>Regression Statistics</i>	
Multiple R	0.952081354
R Square	0.906458905
Adjusted R Square	0.795347794
Standard Error	5.81849657
Observations	10

ANOVA

	<i>df</i>	<i>SS</i>	<i>MS</i>	<i>F</i>	<i>Significance F</i>
Regression	1	2952.6348	2952.63487	87.2143966	1.41108E-05
Residual	9	304.69412	33.8549023		
Total	10	3257.329			

	<i>Coefficients</i>	<i>Stand Err</i>	<i>t Stat</i>	<i>P-value</i>	<i>Lower 95%</i>	<i>Upper 95%</i>
Intercept	0	#N/A	#N/A	#N/A	#N/A	#N/A
X	9.130106762	0.3104578	29.4085247	2.96632E-10	8.427801825	9.83241169

In case of regression through the origin, the total sum of squares should not be calculated from $\sum_{i=1}^n (y_i - \bar{y})^2$ but from $\sum_{i=1}^n y_i^2$. Consequently, the total sum of squares of 3257.329 is wrong in the table above and should be replaced by the correct value of 29584.49. The correct ANOVA table then becomes:

ANOVA

	<i>df</i>	<i>SS</i>	<i>MS</i>	<i>F</i>	<i>Significance F</i>
Regression	1	29279.79588	29279.7958	864.861330	2.96632E-10
Residual	9	304.694121	33.8549023		
Total	10	29584.49			

Note that the p-value calculated from the ANOVA table and the p-value for the slope are now exactly the same, as it should be. Indeed, for simple linear regression the square of the value for t Stat for the slope should equal the value for F in the ANOVA table. The adjusted R-square can be calculated from the definition: $1 - n/(n-1) \times R^2 = 0.896065$.

Excel's normal probability plot

One of the output possibilities in the Analysis Toolpak's 'Regression' tool is the 'normal probability plot'. A probability plot of residuals is a standard way of judging the adequacy of the normality assumption in regression. Well, you might think that this plot in Excel is a normal probability plot of the residuals, but actually the ordered target values y_i are plotted versus $50(2i-1)/n$, which are the ordered percentiles. This has nothing to do with normality of residuals at all. It is simply a plot checking for uniformity of the target variable, which is of no interest in 'model adequacy checking'.

The multi-collinearity problem

Let me show you an example to demonstrate what can happen in case of multicollinearity. A physiologist wanted to investigate the relationship between the physical characteristics of preadolescent boys and their maximum oxygen uptake (measured in milliliters of oxygen per kilogram body weight). The data shown in the table were collected on a random sample of 10 preadolescent boys.

Maximal oxygen uptake	Age years	Height centimeters	Weight kilogram	Chest depth centimeters
1.54	8.4	132.0	29.1	14.4
1.74	8.7	135.5	29.7	14.5
1.32	8.9	127.7	28.4	14.0
1.50	9.9	131.1	28.8	14.2
1.46	9.0	130.0	25.9	13.6
1.35	7.7	127.6	27.6	13.9
1.53	7.3	129.9	29.0	14.0
1.71	9.9	138.1	33.6	14.6
1.27	9.3	126.6	27.7	13.9
1.50	8.1	131.8	30.8	14.5

Using the Analysis Toolpak 'Regression' tool the following output is obtained:

<i>Regression Statistics</i>	
Multiple R	0.983612406
R Square	0.967493366
Adjusted R Square	0.941488059
Standard Error	0.037209173
Observations	10

ANOVA

	<i>df</i>	<i>SS</i>	<i>MS</i>	<i>F</i>	<i>Significance F</i>
Regression	4	0.206037387	0.051509347	37.20369	0.000651321
Residual	5	0.006922613	0.001384523		
Total	9	0.21296			

	<i>Coefficients</i>	<i>Standard Error</i>	<i>t Stat</i>	<i>P-value</i>	<i>Lower 95%</i>	<i>Upper 95%</i>
Intercept	-4.774738582	0.862817732	-5.53389019	0.002643	-6.992678547	-2.556798617
Age	-0.035213868	0.015386301	-2.288650763	0.070769	-0.074765548	0.004337812
Height	0.0516366	0.006215219	8.308089712	0.000413	0.035659896	0.067613303
Weight	-0.023417087	0.013428354	-1.743853833	0.14164	-0.057935715	0.01110154
Chest depth	0.03448873	0.085238766	0.404613206	0.70249	-0.184624134	0.253601595

Let me now critically investigate this result by asking the following questions:

- Is the model adequate for predicting maximal oxygen uptake? Yes! From the ANOVA table one can see that $p = 0.00065$ (significance F) < 0.05 . R^2 is approximately 97%!
- Which variables are significant? Apparently, only the intercept and height are significant! Did you expect this? Didn't you expect that the greater a child's chest depth, the greater should be the maximal oxygen uptake? A strong non-significant p-value for chest depth is unexpected!
- It seems reasonable to think that the greater a child's weight, the greater should be his lung volume and the greater should be the maximal oxygen uptake? To be more specific: a positive coefficient for weight is expected!! A negative coefficient for weight is totally unexpected! It seems that common sense and statistics don't go together in this example!

What is happening here? Let me calculate the coefficient of correlation between each pair of independent variables! To do this use Data analysis \Rightarrow Correlation in Excel. Does this information ring a bell?

	<i>Age</i>	<i>Height</i>	<i>Weight</i>	<i>Chest depth</i>
Age	1			
Height	0.327482983	1		
Weight	0.230740335	0.789825204	1	
Chest depth	0.165752284	0.790945224	0.880960517	1

Apparently, there is a very high correlation between weight and chest depth, meaning that both variables are providing the same information to the data set. Also, weight and height, height and chest depth are strongly correlated. This causes the problem of multicollinearity. This data set cannot be fitted to the multivariate model because calculations become numerically unstable due to the high correlation between variables.

Although Excel correctly calculates the regression output, there is obviously something wrong here. However, there is no actual calculation problem. The fact is that there is no meaningful regression possible here, because the predictors are collinear. This means that no regression model can be fit using all predictors. The problem with Excel is – as compared to statistical packages – that it doesn't give a warning for such high collinearity. Statistical packages will correctly note the perfect collinearity among the predictors and drop one or more if necessary, allowing the regression to proceed, or report the problem and their inability to find a solution, while Excel will find a solution that is wrong. Excel does not compute collinearity measures (such as the Variance Inflation Factor) and consequently does not warn the user when collinearity is present and reports parameter estimates that may be nonsensical.

Data organization

Excel requires the X-variables to be in contiguous columns in order to input them to the regression procedure. This can be done with cut and paste, but is certainly annoying, especially if many regression models are to be built.

Hypothesis testing

As can be easily understood from the above discussion, all hypothesis tests in Excel that need the calculation of a standard deviation or a variance, will suffer from the poor numerical algorithms Excel uses. Let me take an example using two variables and perform (just to demonstrate the erroneous results) several hypothesis tests, such as t-tests and F-tests. Here is the data to demonstrate this (actually variable 2 = variable 1 plus 1):

	A	B
1	Variable 1	Variable 2
2	10000000001	10000000002
3	10000000002	10000000003
4	10000000003	10000000004
5	10000000004	10000000005
6	10000000005	10000000006
7	10000000006	10000000007
8	10000000007	10000000008
9	10000000008	10000000009
10	10000000009	10000000010
11	10000000010	10000000011


The t-test assuming equal variances from the Analysis Toolpak gives the following result:

t-Test: Two-Sample Assuming Equal Variances

	Variable 1	Variable 2
Mean	10000000005.5	10000000006.5
Variance	0	0
Observations	10	10
Pooled Variance	0	
Hypothesized Mean Difference	0	
df	18	

t Stat	65535
P(T<=t) one-tail	#NUM!
t Critical one-tail	1.734063062
P(T<=t) two-tail	#NUM!
t Critical two-tail	2.100923666

As can be seen from the table above, the variances equal zero, including the pooled variance. This results in an unrealistic value for t Stat of 65535. As a consequence, the p-value cannot be calculated. If Excel had used a better algorithm to calculate the variances, the result would have been correct.

	Note that if you apply Excel's built-in function TTEST on these data, you obtain =TTEST(A2:A11;B2:B11;2;2) = 0.4697, which is the correct result.
---	---

Applying the Analysis Toolpak's 't-test: two sample assuming unequal variances', one obtains:

t-Test: Two-Sample Assuming Unequal Variances

	<i>Variable 1</i>	<i>Variable 2</i>
Mean	10000000005.5	10000000006.5
Variance	0	0
Observations	10	10
Hypothesized Mean Difference	0	
df	0	
t Stat	65535	
P(T<=t) one-tail	#NUM!	
t Critical one-tail	#NUM!	
P(T<=t) two-tail	#NUM!	
t Critical two-tail	#NUM!	

Now apart from calculating the variances wrongly, Excel seems to be unable to even calculate the correct number of degrees of freedom. Again, the TTEST function gives the correct p-value. The degrees of freedom are calculated from the Welch-Satterthwaite approximation, which is based on the variances of both groups. So, it is not surprising that if the variances cannot be correctly calculated, this will also apply to the degrees of freedom.

Note also that even for an easy dataset, the Analysis Toolpak's 't-test: two sample assuming unequal variances', gives the incorrect p-value as it is based on the wrong number of degrees of freedom. The error here is that Excel rounds the obtained number of degrees of freedom from the Welch-Satterthwaite approximation to the nearest integer before it calculates the corresponding p-value. All statistical packages that I know use the exact non-integer number of the degrees of freedom obtained from the Welch-Satterthwaite approximation, and use an interpolation algorithm to calculate a more exact p-value. Remarkably the TTEST function, when applied for unequal variances, gives the correct p-value. Here, the Analysis Toolpak and Excel's built-in function do not use the same calculation algorithms.

The example above would be a bad example to use for the Analysis Toolpak's 't-test: paired two sample for means', as Variable 2 is simply Variable 1 plus 1 (the differences would be

the same for all observations, resulting in zero variance). Therefore, I modified the data slightly to give:

Variable 1	Variable 2
10000000001	10000000001
10000000002	10000000003
10000000003	10000000004
10000000004	10000000005
10000000005	10000000007
10000000006	10000000007
10000000007	10000000008
10000000008	10000000009
10000000009	10000000010
10000000010	10000000011

The output of the Analysis Toolpak's 't-test: paired two-sample for means' is:

t-Test: Paired Two Sample for Means

	Variable 1	Variable 2
Mean	10000000005.5	10000000006.5
Variance	0	0
Observations	10	10
Pearson Correlation	#DIV/0!	
Hypothesized Mean Difference	0	
df	9	
t Stat	-6.708203932	
P(T<=t) one-tail	4.3857E-05	
t Critical one-tail	1.833113856	
P(T<=t) two-tail	8.7714E-05	
t Critical two-tail	2.262158887	

Note that the Pearson correlation could not be calculated, resulting in #DIV/0!. The value of the variances are again zero. However, the value of t Stat and the p-values are correct. This is because the calculations are based on the differences, which become small numbers, and calculating the standard deviation will be correct again. Note that if the differences between paired observations had been very large, the results would probably have been wrong (the question then is whether you really need to do a paired test to see that there is a difference). The TTEST function also gives the correct result for a paired test.

Let me continue with the example and see what Excel generates if I want to do an F-test (for the equality of variances). The Analysis Toolpak's 'F-test: Two-sample for variances' test gives the following output:

F-Test Two-Sample for Variances

	Variable 1	Variable 2
Mean	10000000005.5	10000000006.5

Variance	0	0
Observations	10	10
df	9	9
F	65535	
P(F<=f) one-tail	#NULL!	
F Critical one-tail	0.314575033	

Clearly, this test suffers from the same problem: both the variances are zero and the value for F is unrealistic. Now Excel gives ‘#NULL!’ as the p-value.

Excel’s built-in function FTEST gives the following result: 0.867465. Excel’s on-line help says this function returns the one-tailed probability that the variances in Array 1 and Array 2 are not significantly different. This is clearly NOT correct as the value returned by the FTEST function is the two-tailed probability. This can easily be seen by calculating the F-value from the correct variances. These variances can be obtained on the centered data:

-4.5	-5.5
-3.5	-3.5
-2.5	-2.5
-1.5	-1.5
-0.5	0.5
0.5	0.5
1.5	1.5
2.5	2.5
3.5	3.5
4.5	4.5
Variance 1	Variance 2
9.166666667	10.277777778

The F-value thus becomes: $F = 10.2778 / 9.1667 = 1.1212$. Using FDIST to obtain the p-value one can find: $FDIST(1.1212, 9, 9) = 0.4337325$, which is exactly half of the value obtained by the FTEST function. The value obtained from FDIST is the one-tailed p-value. Taking 2 times the value obtained from FDIST is equal to the value obtained by FTEST, which is the two-tailed p-value.

Although the FTEST function returns the two-tailed p-value, (in contrast to what has been claimed in the online help), this value is correct. Apparently, Excel’s built-in function FTEST and the calculation in the Analysis Toolpak are not based on the same numerical algorithm.

Excel’s Analysis Toolpak algorithms for ANOVA (one-way, two-way) suffer from the same ill-balanced numerical algorithms. They calculate the wrong variances and as such the ANOVA tables are not correct.

Excel’s ZTEST function and the z-test in the Analysis Toolpak make use of the value for the population sigma, which has to be entered as such. Consequently these tools provide the correct p-values. Although two remarks should be made:

- 1) these two tools do not perform the same statistical test. The ZTEST function is the one-variable z-test, where the mean of one variable is tested against some prefixed population mean μ , with known value of σ . The ‘z-test two-sample for means’ of the

Analysis Toolpak compares two variables with known standard deviation against each other.

- 2) The ZTEST function returns the one-sided p-value, although Excel's help says it is the two-tailed p-value.

Missing data

Missing data can cause all kind of problems in Excel (at least in Excel '97 because apparently this has been 'solved' in Excel 2000 and XP, although 'solved' is not really the correct way to type it as only error messages are now displayed when there are missing values, and no result is given).

As an example, in Excel '97, I take the paired t-test function TTEST and the Analysis Toolpak's 't-test: paired two-sample for means'. The following data will be used to demonstrate the different outcomes of TTEST and the Analysis Toolpak's paired t-test.

Sample 1	Sample 2
3	2
4	
3	2
	3
2	3
4	3
4	3
3	4
2	3
4	2

The TTEST function returns 0.401508 for the p-value. The output of the Analysis Toolpak 'paired t-test' is:

t-Test: Paired Two Sample for Means

	<i>Sample 1</i>	<i>Sample 2</i>
Mean	3.222222222	2.777777778
Variance	0.694444444	0.444444444
Observations	9	9
Pearson Correlation	-0.18156826	
Hypothesized Mean Difference	0	
df	8	
t Stat	0.644658371	
P(T<=t) one-tail	0.268595733	
t Critical one-tail	1.85954832	
P(T<=t) two-tail	0.537191465	
t Critical two-tail	2.306005626	

The two-tailed p-value is clearly different from the one obtained with the TTEST function. Which one is right? Or are both wrong?

Manual calculation gives us the following results (assuming the data range is A1:C11):

Sample 1	Sample 2	Difference
3	2	1
4		
3	2	1
	3	
2	3	-1
4	3	1
4	3	1
3	4	-1
2	3	-1
4	2	2
	Average	0.375
	StDev	1.187734939

Note here that if you apply ‘=A2-B2’ in cell C2 to obtain the difference (and drag this formula to C11), Excel will assume empty cells as Zero and the result in cell C3 will be 4, and in cell C5 one will obtain -3. Be careful with this. It is clear that these cells should be empty as well. Excel’s built-in functions AVERAGE and STDEV do not have problems with empty cells and the correct result is obtained. From these values, using $t \text{ Stat} = \text{Average of Differences} / (\text{stdev}/\sqrt{n})$ where n is the number of pairs with non-missing data (here $n = 8$), one will find $t \text{ Stat} = 0.893$. Using TDIST($t \text{ Stat}$, 7, 2) gives 0.401508, which is exactly the same p-value as returned by Excel’s =TTEST(A2:A11;B2:B11;2;1). Apparently, the TTEST function knows how to deal with missing values, the Analysis Toolpak clearly not.

Chi-square test

Excel also has a function to perform a chi-square test, that is, CHITEST. This function requires the observed counts AND the expected counts. But here, you are supposed to calculate the expected counts yourself. If you have a sufficient statistical background and know how to do Excel calculations, you’ll be able to calculate them. If not, it seems to be your problem as Excel’s online help will definitely not tell you how.

General remarks about Excel’s statistical analysis tools

- Commonly used statistics and methods are simply not available within Excel. As stated earlier, Excel is clearly not a statistical package. It contains only a very small number of statistical tools, and even for the student, this will quickly turn out to be simply not enough.
- Paired t-tests, ANOVA, Regression and other Analysis Toolpak tools in Excel badly deal with missing values. However, this seems not to be the case for Excel’s built-in statistical functions.
- Excel requires the data in “tabular” form, rather than in “list” form, which encourages bad practice for data storage. This requires extra work if the data have been stored appropriately. Moreover, the data organization might differ according to the analysis, forcing you to organize your data in many ways if you want to do many analyses.

- Output is poorly organized, sometimes inadequately labeled (Excel uses strange names for the analysis it performs or to name output measures (e.g., significance F is the p-value)). The Analysis Toolpak never indicates the significance level used in the output.
- The numerical algorithms used are not optimal, especially when the variance or standard deviation is much smaller than the average of the variable. Standard deviation, variances and all Analysis Toolpak tools that rely on standard deviation calculations where large numbers with low variation are involved, will be unreliable. Remarkable here is the fact that Excel's built-in hypothesis test functions do not suffer from this unstable numerical algorithms. On the other hand, Excel's built-in regression functions (like LINEST, TREND, LOGEST, etc), which are also used by the Analysis Toolpak 'Regression' tool are based on poor numerical algorithms, mainly because the data are not centered around the mean in the calculations.
- Many analyses can only be done on one column at a time, making it inconvenient to do the same analysis on many columns.
- The Analysis Toolpak tools like ANOVA and Regression seem to be restricted to 16 columns.
- Excel's RANK function does not treat tied observations correctly. I defined a different User Defined Function, called RANKING, to be able to calculate non-parametric hypothesis testing. Perhaps it is fortunate that non-parametric tests are not available in Excel as they would probably rely on the RANK worksheet function.

Will these problems affect you?

If you are using Excel for simple data analysis, on relatively easy data sets, it is most unlikely you will have any problems. The impact of the poorer algorithms used by Excel will be more visible on relatively 'not so easy' data sets. If you are dealing with very large numbers, scaling and/or centering your numbers will solve the problem. Note that you should not use Excel's STDEV function to scale your data, in case of large numbers. In most cases, centering the data will suffice to generate the correct results.

Some of the difficulties mentioned in this text can be overcome by using a good third-party add-in. These add-ins will usually provide the user with many more statistical tools, including non-parametric hypothesis testing, which is completely absent in Excel.

References

Greenfield Tony and Siday Sean, Statistical computing for business and industry, The Statistician, 1980, vol. 29, no.1, p.33.

HR Cook, MG Cox, MP Dainton and PM Harris, Testing spreadsheets and other packages used in metrology. Testing the intrinsic functions of Excel., Report to the National Measurement System Policy Unit, Department of Trade & Industry, September 1999.
www.npl.co.uk/ssfm/download/documents/cise27_99.pdf

Goldwater Eva, Data Analysis Group, Academic Computing, University of Massachusetts, Using Excel for Statistical Data Analysis: Successes and Cautions, November 5, 1999.
www-unix.oit.umass.edu/~evagold/excel.html

Simonoff Jeffrey, Stern School of Business, New York University, Statistical Analysis Using Microsoft Excel 2000.
www.stern.nyu.edu/~jsimonof/classes/1305/pdf/excelreg.pdf

Simon Gary, ASSUME (Association of Statistics Specialists Using Microsoft Excel).
www.jiscmail.ac.uk

Cryer J., Problems using Microsoft Excel for statistics, Proceedings of the 2001 Joint Statistical Meetings.
www.cs.uiowa.edu/~jcryer/JSMTalk2001.pdf

Knüsel L. On the Accuracy of Statistical Distributions in Microsoft Excel 97, Computational Statistics and Data Analysis, 26, 375-377.
<http://www.stat.uni-muenchen.de/~knuesel/elv/excelacc.pdf>

Mc Cullough BD, Does Microsoft fix errors in Excel?, Proceedings of the 2001 Joint Statistical Meetings.

Cox Neil, Use of Excel for Statistical Analysis, AgResearch Ruakura, New Zealand, May 2000.
<http://www.agresearch.cri.nz/Science/Statistics/exceluse1.htm>