

Chapter 8

Using Excel[®] to Do Statistical Analysis

Microsoft Excel[®] is a spreadsheet programme developed by Microsoft[®], which comes bundled with Microsoft Office[®]. The most recent version of Microsoft Office is Office 2013. Not only can Excel perform most basic spreadsheet commands, it contains a programming language called Visual Basic that can be used to create powerful and useful macros. Most, if not all, of the analysis presented in the previous chapters can be easily implemented in Excel. It will be assumed that the reader has a basic understanding of Excel, can write simple formulae (equations), understands what a column and row are, and can create basic graphs. Basic background information about Excel can be found from such sources as the *Excel for Dummies* Series (Harvey 2013). This chapter will examine in detail additional features, such as array functions, using Solver, and writing basic Excel macros. In order to clearly distinguish between the code required for Excel functions and text, all Excel commands and variables are shown in bold **Courier New**. Locations on the ribbon, menu paths, and keystrokes are shown in plain Courier New.

8.1 Ranges and Arrays in Excel

Ranges and *arrays* are how Excel refers to groups of columns and rows. The difference lies in how they are used. A *range* is a group of rows and columns in an Excel spreadsheet, while an *array* is a group of rows and columns used in an Excel function or code. In Excel, an array most closely approximates a matrix.

One useful property of Excel is the ability to write an equation in one cell and then drag it to other cells. When dragging such an equation, any references to a range will be changed as the cells are dragged, for example, the cell **A4** will become **A5** if the formula is dragged down one row and **B4** if it is dragged right by one column. This is called *relative* referencing, since it depends on the location. Although relative referencing is useful, it is not always desirable. Excel allows a cell reference to be made *absolute*, that is, it will not change its value as the cell is

	A	B	C	D	E	F	G
2		<i>y</i>	<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>	\hat{y}
3		5	1.00	1	1.00	1	5.8333333
4		6	1.00	1	1.00	-1	7.1666667
5		7	1.00	1	-1.00	1	4.0833333
6		8	1.00	1	-1.00	-1	5.4166667
7		2	1.00	-1	-1.00	1	2.8333333
8		3	1.00	-1	-1.00	-1	4.1666667
9		4	1.00	-1	1.00	1	4.5833333
10		5	1.00	-1	1.00	-1	5.9166667
11		2	1.00	1	-1.00	1	4.0833333
12		5	1.00	1	-1.00	-1	5.4166667
13		7	1.00	-1	1.00	1	4.5833333
14		8	1.00	-1	1.00	-1	5.9166667
15		3	1.00	0	0.00	0	5
16		$\hat{\beta}$	5	0.625	0.875	-0.66667	
17		$\pm\hat{\delta}$	1.308845	1.444926	1.444926	1.362289	

Fig. 8.1 Naming a range (Excel 2007)

dragged. This is done by placing a dollar sign $\$$ before the element that one wishes to freeze. There are three options (illustrated using cell **A2**):

1. *Absolute Rows and Columns ($\$A\2)*: in this case, the reference will always be to this cell.
2. *Absolute Row but Relative Column ($A\$2$)*: in this case, the row will stay the same, but the column can change.
3. *Absolute Column but Relative Row ($\$A2$)*: in this case, the column will stay the same, but the row can change.

Another useful feature in Excel is the ability to name a range. This means that rather than having to drag and select a very large range of cells, it can be conveniently referred to using the corresponding range name, for example, instead of **average(B4:B1000)** we could write **average(y)**, where **y** has been defined to equal **B4:B1000**. Naming a range is shown in Fig. 8.1 and consists of two steps: select the range and enter a name in the location shown. In order to make your life easier, make sure that the name selected is unique to the workbook and not just the worksheet.

Table 8.1 Excel array functions

Function	Description
mdeterm(array)	Determines the determinant of an array . The result will be a single scalar value.
minverse(array)	Determines the inverse of the $n \times n$ array . The result will be the same size as the initial array.
mmult(array1, array2)	Multiplies two arrays array1 and array2 together. If array1 has size $m \times n$, then array2 must have size $n \times p$. The result will have size $m \times p$.
transpose(array)	Transposes an array , that is, the rows and column are exchanged. If the array was originally $n \times m$, then the output will be $m \times n$.

8.2 Useful Excel Functions

This section will examine various Excel functions that can be used for solving statistical problems.

8.2.1 Array Functions in Excel

Array functions are Excel's equivalent to matrices. Arrays are defined as a range of cells that are treated together. When using array functions, the following steps must be followed:

1. Select the output range.
2. Enter the array formula into one of the cells in the selected array.
3. Once the formula has been entered, press **Ctrl + Shift + Enter** to register the formula as an array formula. Normally, one would simply press **Enter**.

A summary of the most common array functions is given in Table 8.1.

8.2.2 Statistical Functions in Excel

Table 8.2 lists some common statistical functions in Excel. Most of these functions as written only work on newer versions of Excel (2010 or newer). A detailed explanation of the functions and differences can be found in Sect. 2.4: Common Statistical Distributions.

Table 8.2 Excel statistical functions

Function	Description
average (range)	Determines the mean value of the numbers in range .
count (range)	Counts the number of nonempty cells in range .
f.inv (p, df1, df2)	Returns the critical value of the <i>F</i> -test for the given left probability p , the degrees of freedom in the numerator df1 , and the degrees of freedom in the denominator df2 .
norm.inv (p, mean, stdev)	Returns the critical value for the normal distribution with mean mean and standard deviation stdev given the left probability p .
norm.s.inv (p)	Returns the critical value for the standard normal distribution ($\mu = 0$ and $\sigma = 1$) given the left probability p .
rank (value, range, order)	Returns the order rank of value given the range . The option order determines in which manner the list is ordered: 0 implies descending order and 1 implies ascending order.
stdev (range)	Determines the sample standard deviation of the numbers in range .
sum (range)	Determines the sum of the numbers in range .
t.inv (p, df)	Returns the critical value of Student's <i>t</i> -test for the given left probability p and degrees of freedom df .

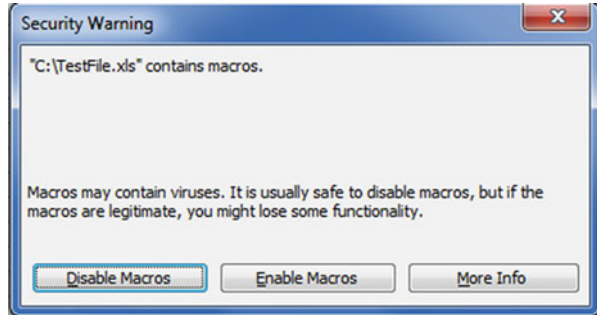
8.3 Excel Macros and Security

Macros are Excel's version of functions, or user-written code, that Excel can execute. The programming language used by Excel is called Visual Basic (VB).

In Excel 2003, code can be inserted by going to Tools → Macro → Visual Basic Editor (Alt + F11). In Excel 2007 or newer, code can be inserted by going to the View Ribbon and selecting the Macro icon and then View Macro. For both versions of Excel, in the window that appears, enter the name of the function that you desire to create (or edit) and press Create (Edit). If a new function is being created then, in the new window that opens, replace **Sub** with **Public Function**. This will allow the new code to be directly accessed from the spreadsheet by typing = **FunctionName (Required Parameters)**. Below, some sample code has been provided that implements the Michaelis–Menten equation.

```
Public Function MichaelisMenten(Concentration, vmax, KM) As
Double
' This function will contain a single line of code that ' implements
the Michaelis-Menten equation
MichaelisMenten = vmax * Concentration / (KM + Concentration)
End Function
```

Fig. 8.2 Warning when dealing with a file with a macro in Excel 2003



8.3.1 Security in Excel

Unfortunately, when a macro is designed, Excel has the tendency to be paranoid and think that it is always a nasty virus. Thus, the appropriate parameters should be set for security. The procedure in dealing with security in Excel depends on the version of Excel installed. The following sections explain the procedures for Excel 2003 (and older) and Excel 2007 (and newer).

8.3.1.1 Dealing with Security in Excel 2003 or Older

In Excel 2003, go to `Tools` → `Macros` → `Security`. A new window will appear. Select `Medium`. With this option, Excel will always ask you on opening the file whether you wish to activate the macros or not. When a file with macros is loaded, then the warning shown in Fig. 8.2 will appear when the file is initially loaded.

8.3.1.2 Dealing with Security in Excel 2007 or Newer

In Excel 2010 or newer, to set the security, go to the `File` menu, and select `Options`. In Excel 2007, go the `Start` button instead of the `File` menu and select `Excel Options`. Select `Trust Center` in the window that appears. After this, select `Trust Center Settings...` In the new window, go to `Macro Settings` and select the appropriate level of security you desire. A good choice is to select the option `Disable All Macros with Notification` because the macros will be disabled, but you will be notified of their existence. Press `OK` on all the open windows to save the changes. A file with a macro must be saved as an `.xlsm` file. In Excel 2007 or newer, when opening the file with a macro and the suggested above settings, a warning will appear. Clicking on the warning will in Excel 2010 or newer activate the macros (shown in Fig. 8.3), while in Excel 2007 another window will appear which allows the macros to be activated (shown in Fig. 8.4).

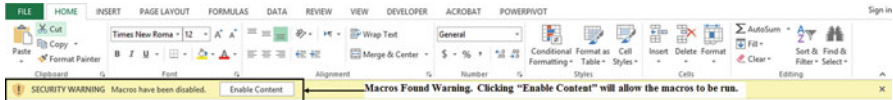


Fig. 8.3 Security warning when macros are present (Excel 2010 or newer)

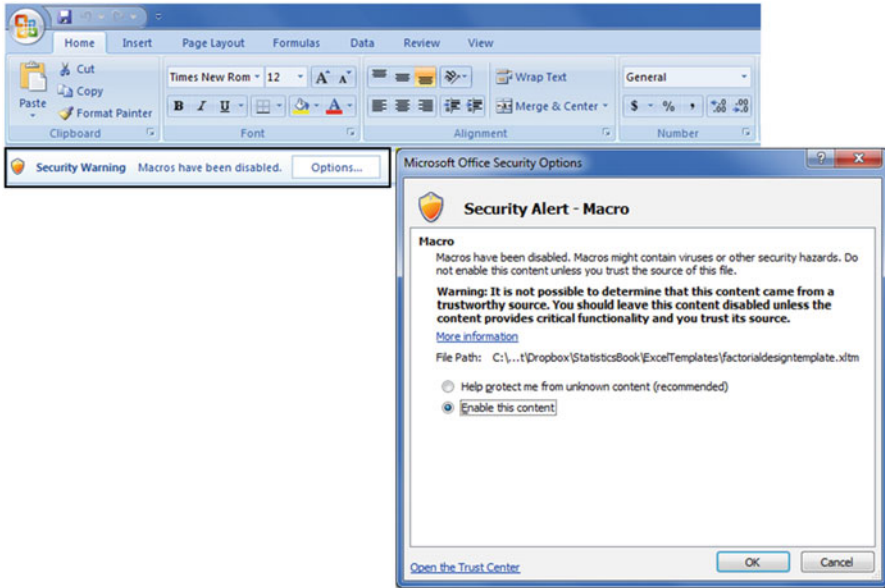


Fig. 8.4 Security warning when macros are present (Excel 2007). The inset shows the window that appears after clicking options

8.4 The Excel Solver Add-In

Solver is an Excel add-in that allows the user to iteratively solve systems of equations. Unfortunately, it is not installed by default on most computers.

8.4.1 Installing the Solver Add-In

In Excel 2003, in order to install the Solver add-in, go to **Tools** → **Add-in**. In the window that appears, which is shown in Fig. 8.6, select **Solver Add-in** and press **OK**. Excel will then try to install the add-in. This may require the original Excel CD or DVD.

In Excel 2007, in order to install the Solver add-in, go to the **Start** button and select **Excel Options** at the bottom of the menu that appears. In the new window

that appears, select `Add-ins`. Finally, go to the `Go...` button and a window similar to Fig. 8.6 should appear.

In Excel 2010 or 2013, in order to install the Solver add-in, go to the `File` menu and select `Options` at the bottom of the Menu that appears. In the new window that appears, select `Add-ins`. Finally, click the `Go...` button. The last two steps are shown in Fig. 8.5. A window similar to Fig. 8.6 should appear.

8.4.2 Using the Solver Add-In

In order to start Solver, in Excel 2007 or newer, locate the `Data` ribbon and go to the extreme right-hand side in the area marked `Analysis`. Solver should be there as shown in Fig. 8.7. In Excel 2003 or older, go to `Tools` → `Solver`.

Figure 8.8 shows the main Solver window that appears in Excel 2010 or newer. It is a must that the option `Make Unconstrained Variables Non-Negative` be *unchecked*, as it can lead to wrong results otherwise. The following sections are important for use in regression analysis:

1. *Objective Function Value*: this is the value of the objective function that is to be optimised.
2. *Type of Optimisation*: what type of optimisation is desired: maximisation (`Max`), minimisation (`Min`), or force the solver to obtain a particular value (`Value of`). For regression, the minimisation option should be used.
3. *Variables*: this is the range of the cells (variables) that the computer can vary to determine the solution. For regression, this would represent the cells where the parameter values have been entered.
4. *Constraints*: this box lists the constraints for the problem. In order to add a constraint, click on the “Add” button. The window shown in Fig. 8.9 should appear. Once the desired form of the constraint has been selected, click `Add` to add the constraint to the list of constraints. Selecting a constraint from the box and clicking `Change` will cause the same window to appear and the properties of the constraint can be changed. Finally, selecting a constraint and clicking `Delete` will remove the constraint.
5. *Solve*: clicking this button will start the solver. The solution of the problem may take some time. Solver will either state that a solution was found (Fig. 8.10 (left)) or that no solution was found (Fig. 8.10 (right)). In general, if a solution is found, select `Keep Solver Solution` and press `OK`; otherwise, select `Restore Original Values` and press `OK`. If the Solver fails to find a solution, an error message will be included. It can give a suggestion as to how to fix the problem. Three common things to check (in order of precedence) are that:
 - (a) The number of iterations was not exceeded.
 - (b) The Excel spreadsheet and Solver were properly configured, especially that Box 7 in Fig. 8.8 is *unchecked*; and

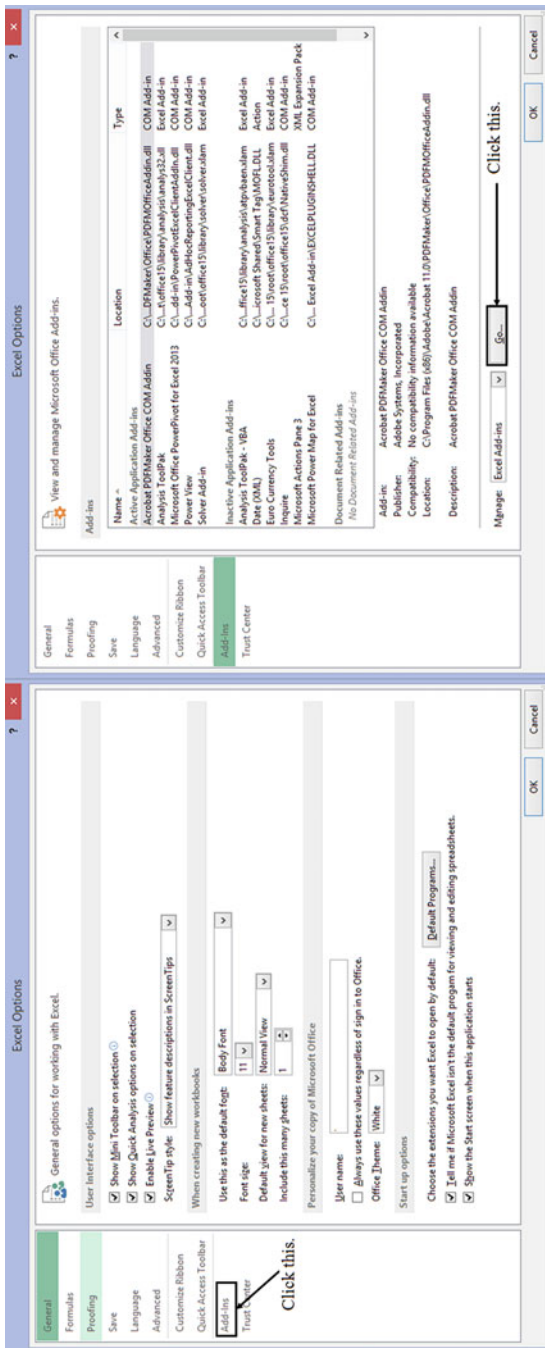


Fig. 8.5 Navigating to the Solver installation menu (Excel 2013)

Fig. 8.6 Installing Solver

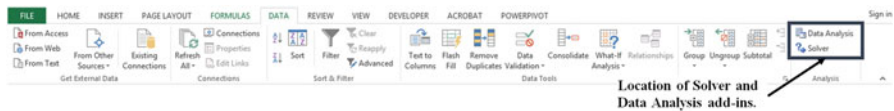
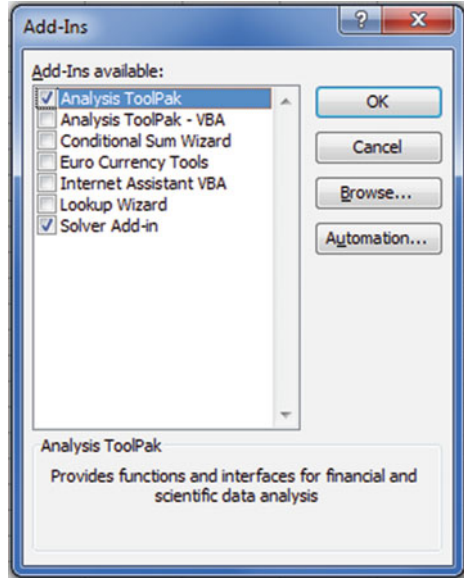


Fig. 8.7 Location of the Solver and Data Analysis add-ins (Excel 2013)

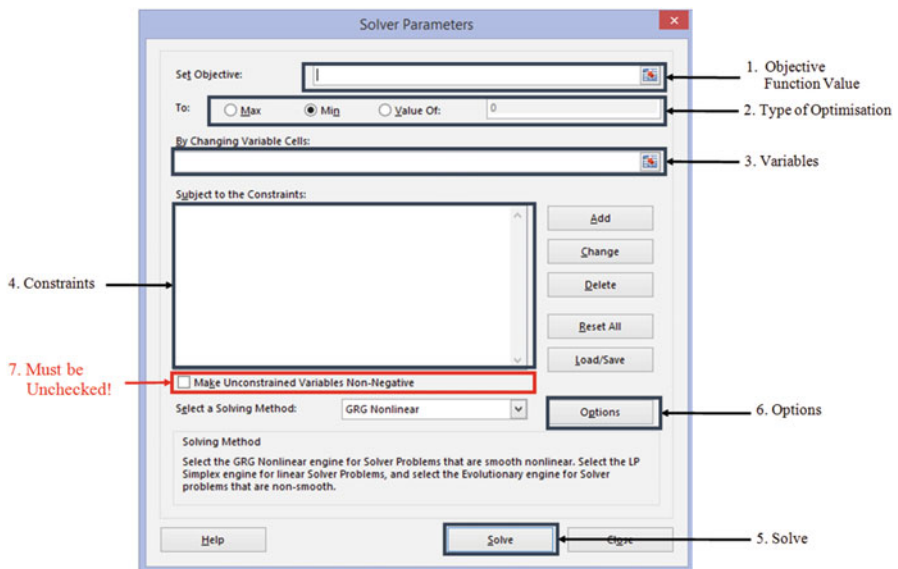


Fig. 8.8 Main Solver window (Excel 2010 or newer)

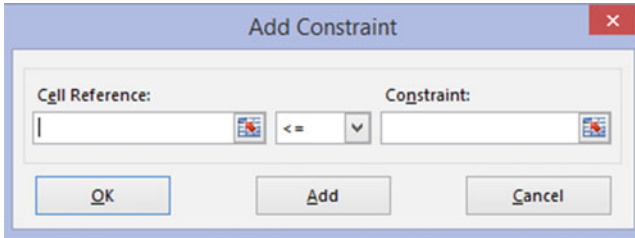


Fig. 8.9 Add constraint window

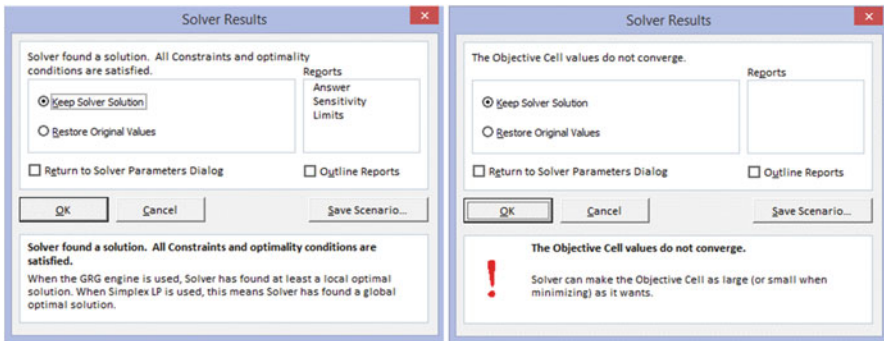


Fig. 8.10 (Left) Solver found a solution and (right) Solver failed to find a solution (one possible result)

- (c) To make sure that the appropriate solver method was selected. Changing the solver method from GRG nonlinear to evolutionary can be useful.
6. *Options*: clicking this button will bring up the window shown in Fig. 8.11. Each of the choices in this window can speed up or slow down the amount of time required to obtain a solution or even if a solution can be found. Each option will be discussed separately:
 - (a) *Max Time*: this represents the maximum amount of time that Solver will run in order to determine a solution. If the problem is large, then increasing this option can potentially allow Solver to find a solution.
 - (b) *Iterations*: this represents the maximum number of iterations that Solver will perform before it stops. If the initial guess is far from the solution, it may take many iterations before a solution is obtained. Thus, increasing the number of iterations can be a good idea.
 - (c) *Precision*: this represents the largest possible difference between the calculated value of the constraints and the specified value of the constraints. The smaller the number the longer it will take to find a solution.

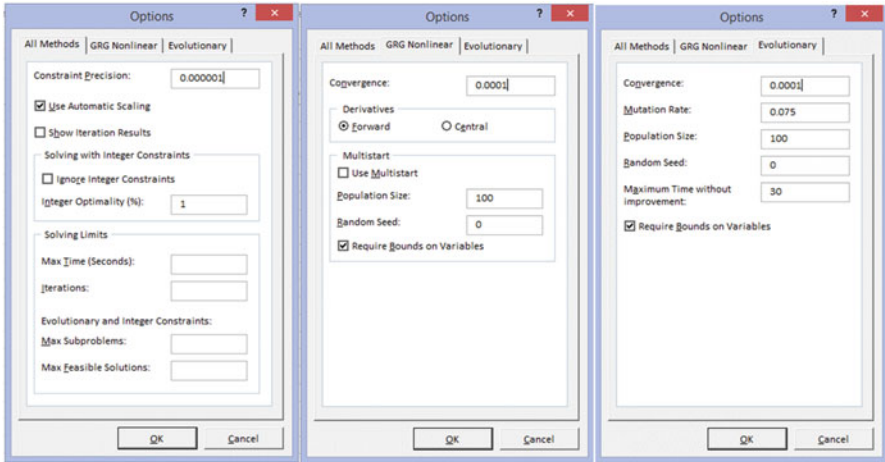


Fig. 8.11 Solver option window (Excel 2010 or newer)

- (d) *Tolerance*: this is similar to precision but is used for integer constraints. It represents the percentage by which the calculated values differ from the specified values.
- (e) *Convergence*: this is similar to precision but is used to compute the maximum allowable difference between 2 iterations of the parameters (or cells that can change). Since for most purposes, a relative value would be better, this entry should be changed whenever the parameters are expected to either be all very large numbers or very small numbers.
- (f) *Use Automatic Scaling*: this should always be selected as it minimises the effect the magnitude of the different variables can have on the solution. It is especially important if one of the variables ranges from 100 to 1,000, but the other variable ranges from 0.01 to 1.

The options in the other tabs are mostly irrelevant and should be left at their default values unless the problem at hand requires special treatment. However, the correct approach to take requires consulting an appropriate source on numerical methods.

In the older version of Excel (2007 or older), the solver window has the form shown in Fig. 8.12. The available information is the same except that it is arranged slightly differently. Also, the method that Excel uses to solve the nonlinear problem is based on a simple Newtonian search, so that the results can be more difficult to obtain or different from those obtained using the newer methods. The options are shown in Fig. 8.13.

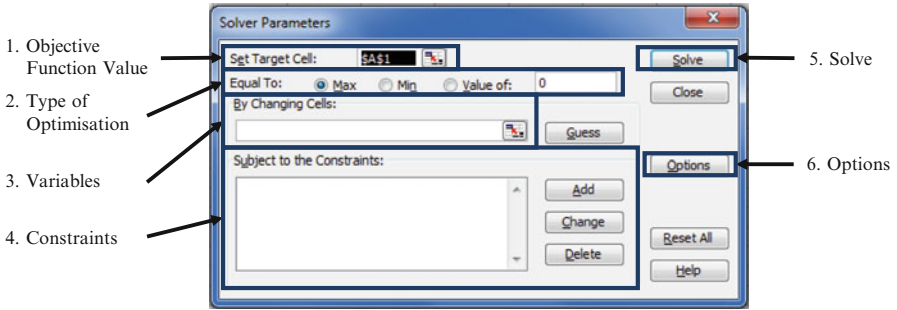
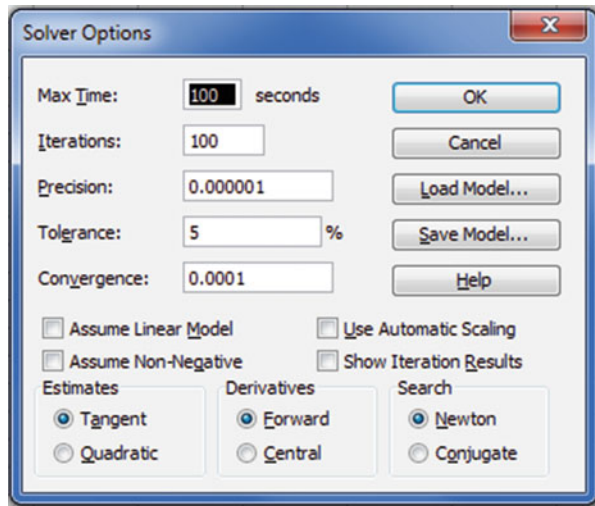


Fig. 8.12 Solver window (Excel 2007 or older)

Fig. 8.13 Solver options (Excel 2007 or older)



8.5 The Excel Data Analysis Add-In

The Data Analysis add-in in Excel is another very useful Excel add-in that can improve the ability to perform certain statistical tasks. It is installed using the same procedure as installing the Solver add-in (see Sect. 8.4.1: Installing the Solver Add-In). In order to start Solver, in Excel 2007 or newer, locate the **Data** ribbon and go to the extreme right-hand side in the area marked **Analysis**. Solver should be there as shown in Fig. 8.7. In Excel 2003 or older, go to **Tools** → **Data Analysis**.

The Data Analysis window is shown in Fig. 8.14. Although there are many different options, the main problem with the data analysis add-in is that the results are static and that any changes made in the original data set require the given programme to be rerun. Also, the display of information is not always the best. Nevertheless, for the purposes of this book, the only useful option is the *Fourier*

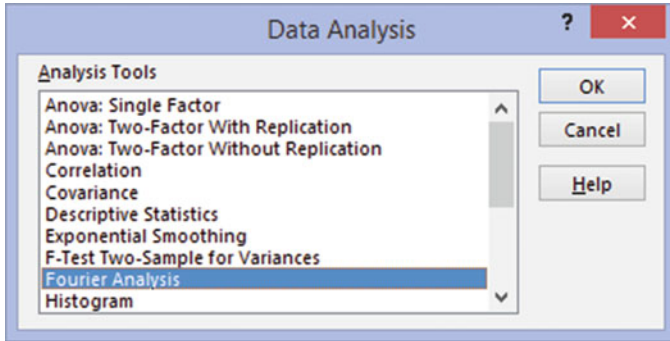


Fig. 8.14 Data Analysis window (Excel 2010 or newer)

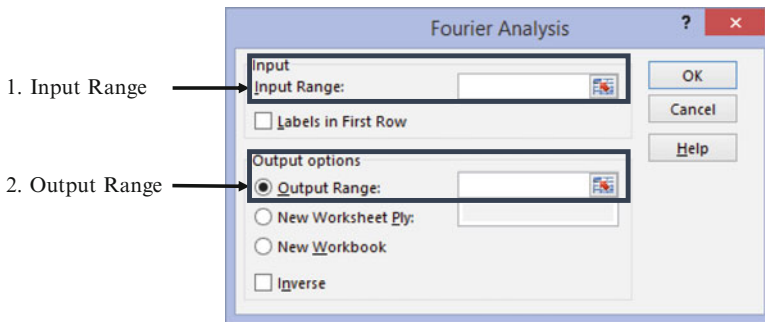


Fig. 8.15 Fourier analysis window (Excel 2010 or newer)

analysis option, which will compute, given a data set, the appropriate Fourier coefficients, which can then be used to create a periodogram for the data set. An Excel template file has been created to simplify the process (see Sect. 8.6.3: Periodogram Template).

Selecting the Fourier analysis option will give the window shown in Fig. 8.15. There are only two key areas to consider. First, the input range must have a length of 2^n , where $n \in \mathbb{N}$, that is, the length must be an integer power of 2. If the particular list is less than the desired value, then add extra zeros to the end of the list to make it an integer power of 2. The output range should have the same size and orientation as the input range, that is, if the input range is a column, then the output range should also be one, similarly for a row. Clicking OK will give the required Fourier coefficients.

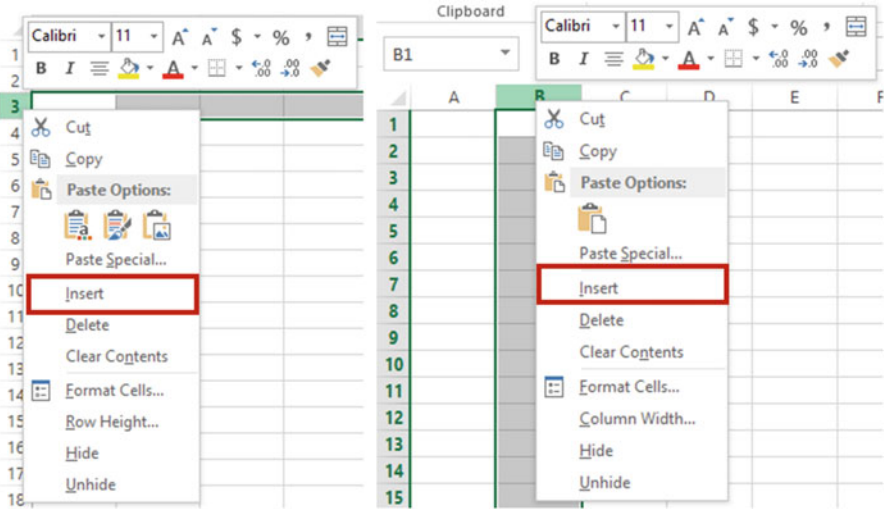


Fig. 8.16 (Left) Inserting a row and (right) column (Excel 2013)

8.6 Excel Templates

This section describes the Excel templates available from the book website. All templates have been tested on Excel 2013 and 2010. All should work on Excel 2007 as well. For the few that require special formulae, a separate Excel 2007 version has been included.

The following are some useful reminders when using the templates:

1. *Adding new rows*: new rows should be entered inside the thick-bordered area. This will automatically update all formulae to include the new row. This can be accomplished by right-clicking on the appropriate row name and selecting *Insert*. A row will be inserted above the selected row. See Fig. 8.16 for an example.
2. *Adding new columns*: new columns should be entered inside the double-bordered area. This will automatically update all formulae to include the new column. This can be accomplished by right-clicking on the appropriate column name and selecting *Insert*. A column will be inserted to the left of the selected column. See Fig. 8.16 for an example.
3. *Formulae*: most formulae can be dragged down or across to fill the new data. It is suggested that you drag from the first row (or column) down to the last to make sure that everything is properly aligned.
4. *Macros*: a few of the templates contain macros that allow for easier and better implementation of the given procedures. When macros are present, make sure that the security is appropriately set so that they can be used (see Sect. 8.3.1: Security).

	A	B	C	D	E	F	G
1	Data	Rank	Expected Value	Normalised Data		Data Summary Statistics	
2	-1.0625	4	-0.8994	-0.7151		mean	0.0525
3	-0.3125	8	-0.2670	-0.2341		$\hat{\sigma}$	1.5593
4	-0.0625	9	-0.1323	-0.0737			
5	2.6875	19	1.9379	1.6899			
6	-4.0625	1	-1.9379	-2.6391			
7	-1.0625	4	-0.8994	-0.7151			
8	1.9375	17	1.1190	1.2089			
9	2.1250	18	1.4122	1.3292			
10	1.0625	14	0.5549	0.6478			
11	-1.1875	3	-1.1190	-0.7952			
12	0.0625	10	0.0000	0.0064			
13	-1.4500	2	-1.4122	-0.9636			
14	1.0625	14	0.5549	0.6478			
15	-0.9375	6	-0.5549	-0.6349			
16	1.0625	14	0.5549	0.6478			
17	0.0625	10	0.0000	0.0064			
18	0.8560	13	0.4067	0.5153			
19	0.7856	12	0.2670	0.4702			
20	-0.5695	7	-0.4067	-0.3989			

← = (A1 - \$G\$2) / \$G\$3

↑ Data

↑ =rank(A1,Data,1)

↑ =normsinv((B1-0.5)/Count(Data))

Fig. 8.17 Normal probability plot data (the formulae given are those placed in the first row, and they would then be dragged down into each of the remaining rows)

8.6.1 Normal Probability Plot Template

Requirements: basic Excel installation.

Goal: create a normal probability plot in Excel that can be modified to deal with other distributions.

Filename: normplot.xlsx

Description: a screen shot of the template with an explanation of the formulae used is shown in Fig. 8.17. The resulting normal probability plot is shown in Fig. 8.18. The steps for creating a normal probability plot can be summarised as follows:

1. Place the original data in Column A.
2. Obtain the order of the data in Column A in Column B. You can use the rank function.
3. In Column C, enter = **normsinv((ColumnB1-0.5)/count(Column \$A))**. 0.5 is subtracted from the original ranked value in order to avoid asking the computer for the location for which the probability is 100% (it is +∞!).
4. In Column D, compute the Z-score for each of the data points, that is, subtract the mean and divide the resulting value by the standard deviation of the values in Column A.

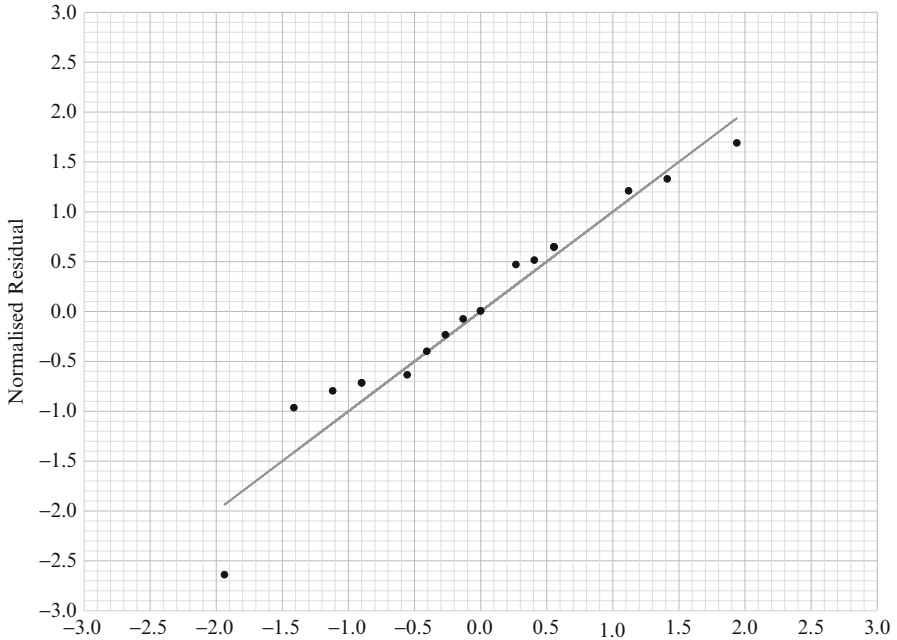


Fig. 8.18 Resulting normal probability plot

5. Plot a scatter plot of the data in Columns C and D.
6. The straight line can be added by plotting the data in Column C against itself.

Warnings: the axes of the plot are fixed to the range $[-3.0, 3.0]$. Should there be data outside this region, then it will be necessary to manually change the axis limits.

8.6.2 Box-and-Whisker Plot Template

Requirements: basic Excel installation.

Goal: create a box-and-whisker plot in Excel.

Filename: boxplot.xltx

Filename, Excel 2007 or Older: boxplot2007.xlt

Description: a screen shot of the box-and-whisker plot is shown in Fig. 8.19. The following steps can be used to create a box-and-whiskers plot in Excel from scratch:

1. Place the data in a column (or row) and call that range **data**.
2. Compute the minimum, first quartile, median, third quartile, and maximum values. This can be accomplished by using the following

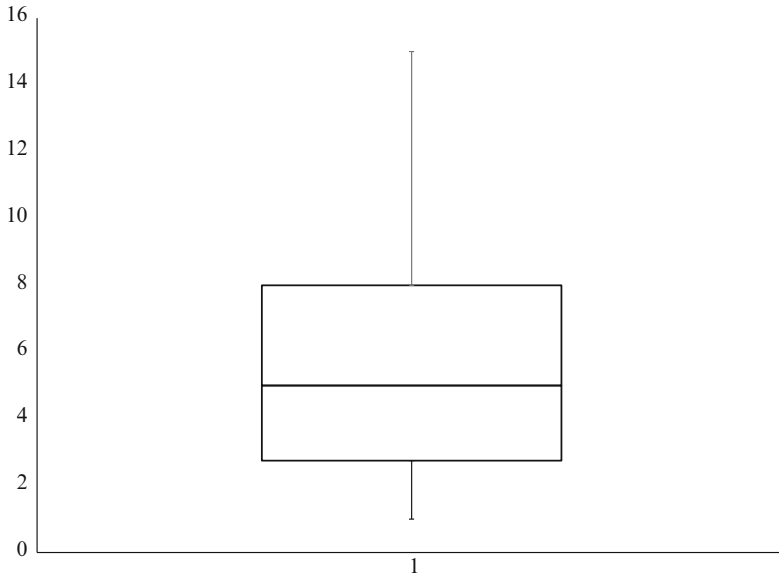


Fig. 8.19 Box-and-whisker plot in Excel

formulae: **min(data)**, **quartile.inc(data,1)**, **median(data)**, **quartile.inc(data,3)**, and **max(data)**. In Excel 2007 or older, one should replace **quartile.inc** by **quartile**.

3. Create a column containing the following values in the specified order: Q1 – minimum, Q1, median – Q1, Q3, and maximum – Q3. This will allow the box-and-whisker plot to be properly created in Excel.
4. Select the middle three items (Q1, median – Q1, and Q3) and create a stacked column graph. The steps required are shown in Fig. 8.20 for Excel 2013. The arrows provide the sequence of steps that should be followed to create the graph. The initial graph that is obtained now needs to be formatted to look like a box-and-whisker plot.
5. Select the bottom blue box and add a negative error bar, which is equal to Q1 – minimum. Set the positive error bar equal to zero. The steps required are shown in Fig. 8.21 for Excel 2013. Select the top grey box and add a positive error bar, equal to maximum – Q3. The negative error bar should be set to zero. The same procedure would be followed, *mutatis mutandis*.
6. Once again, select the blue box and set the fill option to `no fill` and the border option to `no line`. The steps required are shown in Fig. 8.22 for Excel 2013. Select the orange box and set its fill to `no fill`. Repeat for the grey box. This should now look like a box-and-whisker plot. Additional formatting can be performed to obtain the final version.



Fig. 8.20 Creating the initial graph for a box-and-whisker plot (Excel 2013). The arrows provide the sequence of events to follow

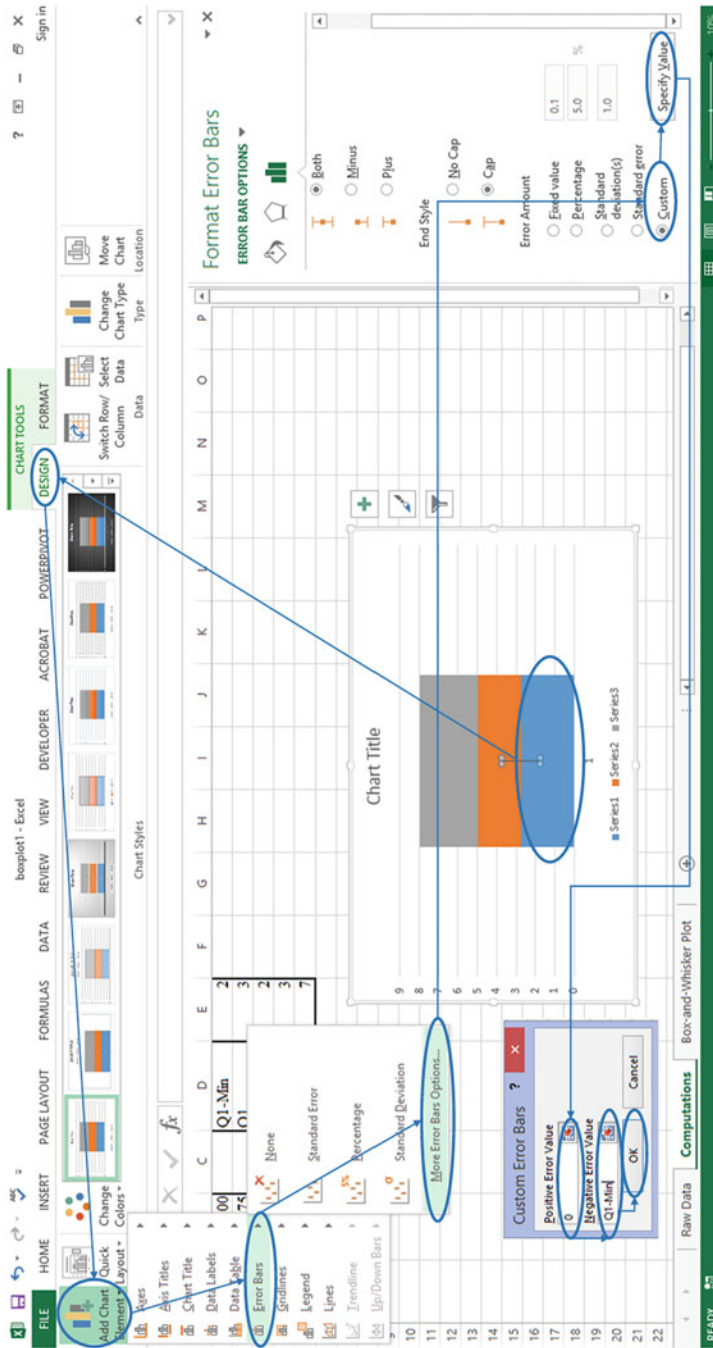


Fig. 8.21 Adding error bars (Excel 2013). The arrows provide the sequence of events to follow

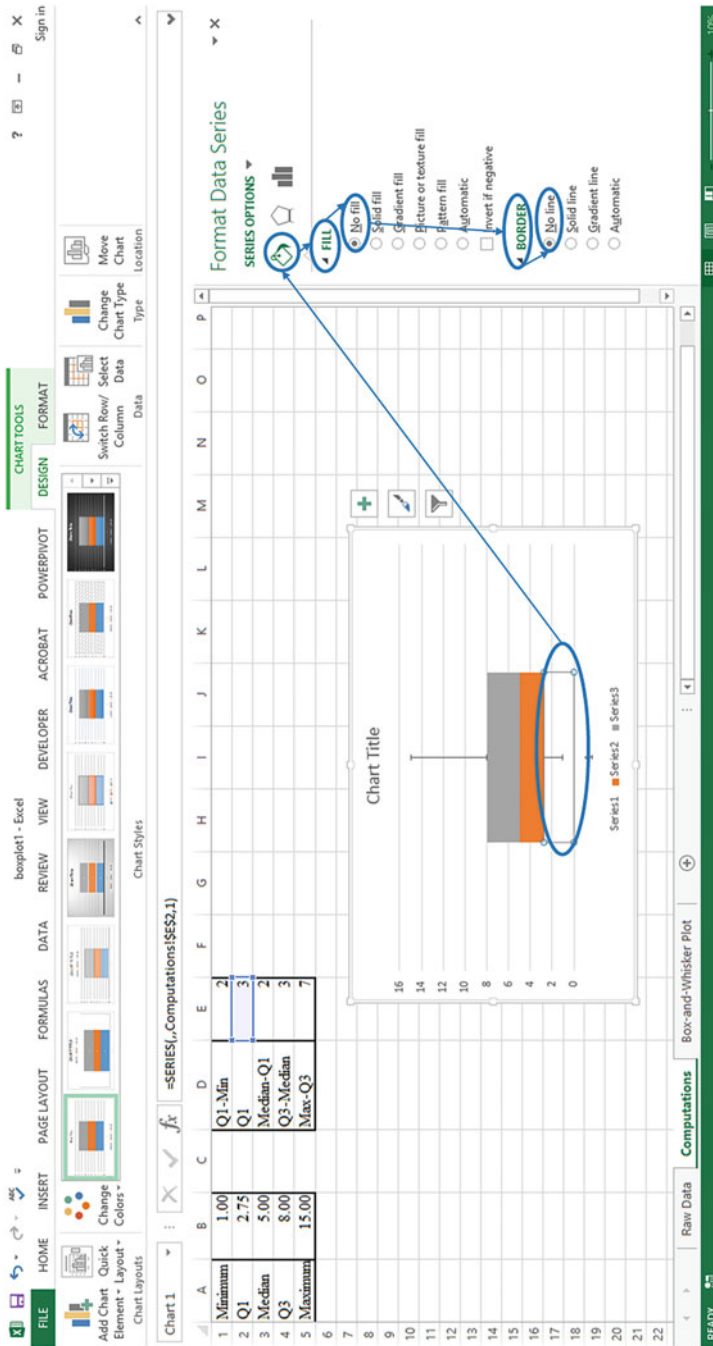


Fig. 8.22 Changing the fill and border options (Excel 2013). The arrows provide the sequence of events to follow

8.6.3 Periodogram Template

Requirements: basic Excel installation plus installing the Data Analysis add-in (see Sect. 8.5: the Excel Data Analysis for how to install it.)

Goal: create both the full and half periodograms in Excel.

Filename: periodogram.xlsx

Description: a screenshot of the template is shown in Fig. 8.23 with the resulting periodograms shown in Fig. 8.24. Note that every time new data are entered, it is necessary to rerun the Fourier analysis function in the Data Analysis add-in. The set-up of the Fourier transform window is shown as an inset in Fig. 8.23. Also, the number of data points must be a multiple of 2^n where n is an integer, that is, 2, 4, 8, 16, 64, 128, 456, ... If the data set of interest is not a multiple, then it is necessary to add extra zeros to the end of the list to make it so.

An explanation of the columns is as follows:

1. *Column A* contains a simple count of the sample number *starting from 1*.
2. *Column B* contains the values corresponding to each sample number. This column is called **data** and must be a multiple of 2^n , where n is an integer.
3. *Column C* contains the Fourier transform values as returned by the Fourier analysis function in Excel. The values are complex numbers and should not be changed.
4. *Column D* contains the magnitude of the values in Column C, that is, $=\text{abs}(C2)$. This column is used to construct the full periodogram.
5. *Column E* contains the frequency corresponding to each sample, that is, $=(A2-1)/\text{COUNT}(\text{data})$.
6. *Column F* contains the half periodogram frequencies, which is basically the first 2^{n-1} values from Column E with the remaining values set to #N/A, so that they will be ignored. The formula used is $=\text{IF}(A2-1 < \$I\$2, E2, \#N/A)$. It should be noted that cell $\$I\2 contains the centre point value.
7. *Column G* contains the half periodogram magnitudes, which is basically twice the corresponding value in Column D, up to the centre point value, after which the values are arbitrarily set to #N/A. This allows the half periodogram to be plotted for an arbitrary number of values. The formula used is $=\text{IF}(A2-1 < \$I\$2, D2*2, \text{"NaN"})$.
8. *Full Periodogram:* the full periodogram is created by plotting Column D as the y-axis and Column E as the x-axis.
9. *Half Periodogram:* the half periodogram is created by plotting Column G as the y-axis and Column F as the x-axis.

Warnings: the Fourier transform function must be rerun each time the data are changed. Furthermore, the data length must always be a multiple of 2^n , where n is an integer.

Figure117.118.xlsx - Excel

Sample	Value	FFT Value	Full Magnitude	Half Frequency	Half Magnitude	Half Frequency	Half Magnitude
1	0.9511	0.951056516295169	0.0037	0.0000	0.0000	0.0000	0.0074
2	0.8090	0.951056516295158	0.0053	0.0078	0.0078	0.0107	0.0181
3	0.4878	0.95105651629516	0.0090	0.0156	0.0156	0.0234	0.0295
4	0.3090	0.951056516295174	0.0148	0.0234	0.0234	0.0313	0.0493
5	0.0000	0.951056516295163	0.0247	0.0391	0.0391	0.0469	0.3666
6	-0.3090	0.951056516295184	0.0477	0.0547	0.0547	0.0625	0.2632
7	-0.4878	0.951056516295263	0.1833	0.0469	0.0469	0.0703	0.0679
8	-0.8090	0.951056516295058	0.1316	0.0524	0.0524	0.0781	0.0512
9	-0.9511	0.95105651629514	0.0524	0.0625	0.0625	0.0859	0.0416
10	-1.0000	0.951056516295134	0.0340	0.0703	0.0703	0.0938	0.0353
11	-0.9511	0.951056516295153	0.0256	0.0781	0.0781	0.1016	0.0308
12	-0.8090	0.951056516295127	0.0208	0.0859	0.0859	0.1094	0.0275
13	-0.4878	0.951056516295141	0.0177	0.0938	0.0938	0.1172	0.0249
14	-0.3090	0.951056516295133	0.0154	0.1016	0.1016	0.1250	0.0228
15	0.0000	0.951056516295137	0.0137	0.1094	0.1094	0.1328	0.0211
16	0.3090	0.951056516295151	0.0124	0.1172	0.1172	0.1406	0.0196
17	0.4878	0.951056516295134	0.0114	0.1250	0.1250	0.1484	0.0184
18	0.8090	0.951056516295128	0.0105	0.1328	0.1328	0.1563	0.0173
19	0.9511	0.951056516295161	0.0098	0.1406	0.1406		
20	1.0000	0.951056516295166	0.0092	0.1484	0.1484		
21	0.9511	0.951056516295148	0.0086	0.1563	0.1563		
22	0.8090	0.951056516295158	0.0080	0.1641	0.1641		

Centre point: 65
Data Length: 128
Power of 2: 128

Data Analysis dialog box options:

- Analysis Tools: Anova: Two-Factor Without Replication, Covariance, Descriptive Statistics, Exponential Smoothing, Forecast, Histogram, Moving Average, Random Number Generation
- Input Range: \$B\$2:\$B\$129
- Output Range: \$C\$2
- Labels in First Row:
- Output options: New Worksheet Ply, New Workbook

Fourier Analysis dialog box options:

- Input Range: \$B\$2:\$B\$129
- Output Range: \$C\$2
- Labels in First Row:
- Output options: New Worksheet Ply, New Workbook

Fig. 8.23 Periodogram template layout (Excel 2013). The inset shows how to initialise the Fourier analysis function

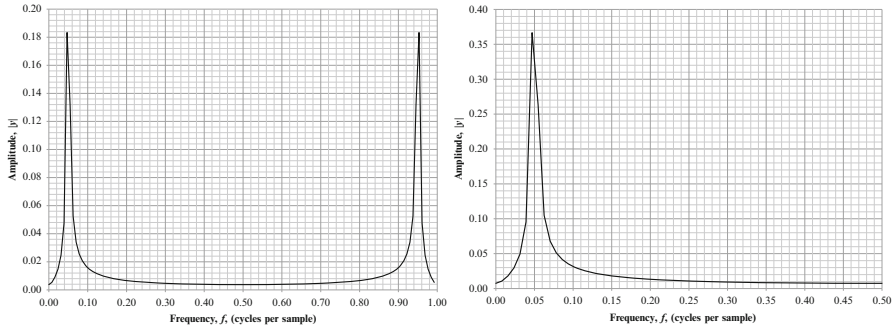


Fig. 8.24 Sample full and half periodograms

	A	B	C	D	E	F	G	H	I	J	K	L
1												
2		y	a	b	c	d	\hat{y}	residuals	r^2	Ordered	Z-Rank	Z-Score
3		5	1.00	1	1.00	1	5.8333333	0.83	0.69	7	0	0.461266
4		6	1.00	1	1.00	-1	7.1666667	1.17	1.36	10	0.61514	0.645772
5		7	1.00	1	-1.00	1	4.0833333	-2.92	8.51	1	-1.76883	-1.61443
6		8	1.00	1	-1.00	-1	5.4166667	-2.58	6.67	2	-1.19838	-1.42992
7		2	1.00	-1	-1.00	1	2.8333333	0.83	0.69	7	0	0.461266
8		3	1.00	-1	-1.00	-1	4.1666667	1.17	1.36	10	0.61514	0.645772
9		4	1.00	-1	1.00	1	4.5833333	0.58	0.34	6	-0.19403	0.322886
10		5	1.00	-1	1.00	-1	5.9166667	0.92	0.84	9	0.39573	0.507392
11		2	1.00	1	-1.00	1	4.0833333	2.08	4.34	13	1.76883	1.153164
12		5	1.00	1	-1.00	-1	5.4166667	0.42	0.17	5	-0.39573	0.230633
13		7	1.00	-1	1.00	1	4.5833333	-2.42	5.84	3	-0.86942	-1.33767
14		8	1.00	-1	1.00	-1	5.9166667	-2.08	4.34	4	-0.61514	-1.15316
15		3	1.00	0	0.00	0		5	2.00	12	1.19838	1.107037
16		β	5	0.625	0.875	-0.66667		sum(r^2)	39.17		$\mu_{residuals}$	8.88E-16
17		$\pm\delta$	1.308845	1.444926	1.444926	1.362289		σ_{model}^2	2.09		$\sigma_{residuals}^2$	1.806624
18								α	0.05		The normal probability plot is automatically created using the data provided in columns H, K, and L and displayed in the Normplot tab of this workbook.	
19								tinv(α)	2.262157			
20								m	13			
21								n	4			
22								k	3			
23								TSS	54			
24								ESS	39.16667			
25								SSR	14.83333			
26								R^2	0.274691			
27								F -test	1.13617			
28								F -critical, model	3.862548			
29								F -critical, factorial	5.117355			
30												

Fig. 8.25 Linear regression template

8.6.4 Linear Regression Template

Requirements: basic Excel installation plus ability to use macros.

Goal: perform linear regression in Excel in an easy and straightforward manner.

Filename: linearregression.xltm

Description: a screenshot of the plain template is shown in Fig. 8.25. The yellow blocks are where the required data are entered. The green block represents the row in which an array formula needs to be entered. The complete green row

should be selected and then the first cell highlighted. Finally, press **Ctrl+Shift+Enter** to copy the array formula to the entire green row. Adding additional parameters and data points will also require that the formulae be appropriately copied down. The spreadsheet automatically creates the normal probability plot for the residuals and plots of the residuals as a function of y and \hat{y} , as well as a time series plot of the residuals. Additional plots can be created by the user. An example of how to use the template is provided in Sect. 8.7.1: Linear Regression Example.

Warning: this template requires that the internal macros be enabled. Also, the array formulae need to be properly entered.

8.6.5 *Nonlinear Regression Template*

Requirements: basic Excel installation plus installation of Solver and the ability to use macros.

Goal: perform nonlinear regression in Excel in an easy and straightforward manner.

Filename: nonlinearregresion.xltm

Description: a screenshot of the plain template is shown in Fig. 8.26. The yellow blocks are where the required data are entered. Note that Solver needs to be used to obtain a solution to the problem. The configuration of Solver is shown as an inset in Fig. 8.26. The layout and formatting of the results are similar to the linear regression case. Two important differences are that the model and its Jacobian must be entered as a macro and that Solver must be used. The spreadsheet automatically creates the normal probability plot for the residuals and plots of the residuals as a function of y and \hat{y} , as well as a time series plot of the residuals. Additional plots can be created by the user. An example of how to use the template is provided in Sect. 8.7.2: Nonlinear Regression Example.

The template comes with four predefined functions for creating the model and the corresponding Jacobian. Each function takes the same inputs: the range corresponding to the parameters and the range corresponding to the inputs. The four functions are `model`, `dydb1`, `dydb2`, and `dydb3`. This approach is very similar to what MATLAB[®] requires and provides the most flexibility in defining the relevant functions.

Warning: this template requires that the internal macros be enabled and Solver installed.

8.6.6 *Factorial Design Analysis Template*

Requirements: basic Excel installation and appropriate macro security.

Goal: perform the analysis of a factorial design experiment in Excel in an easy and straightforward manner.

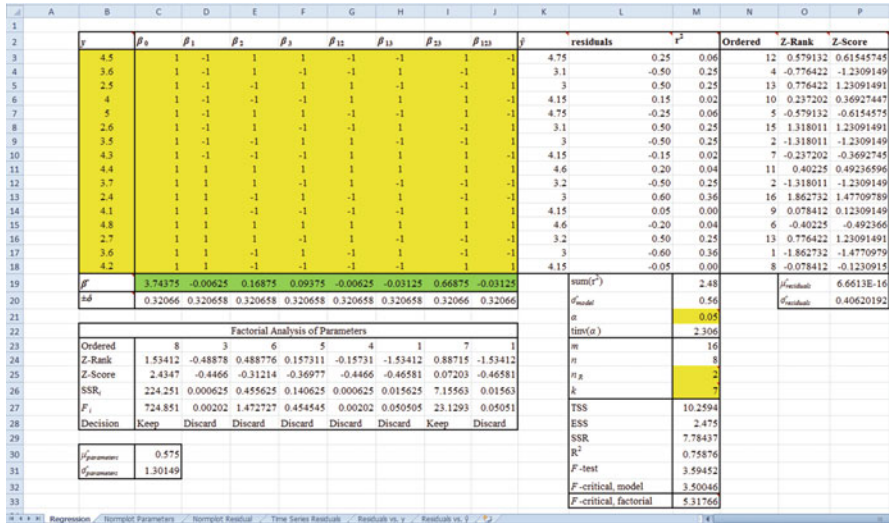


Fig. 8.27 Analysis of factorial experiments template

Filename: factorialdesigntemplate.xltn

Description: a screenshot of the plain template is shown in Fig. 8.27. The yellow blocks are where the required data are entered. The green block represents the row in which an array formula needs to be entered. The complete green row should be selected and then the first cell highlighted. Finally, press **Ctrl +Shift+Enter** to copy the array formula to the entire green row. Adding additional parameters and data points will also require that the formulae be appropriately copied down.

The spreadsheet automatically creates the normal probability plot for the parameters and residuals as well as plots of the residuals as a function of y and \hat{y} and a time series plot of the residuals. Additional plots can be created by the user. An example of how to use the template is provided in Sect. 8.7.3: Factorial Design Examples.

Warning: this template requires that the internal macros be enabled. Also, the array formulae need to be properly entered.

8.7 Excel Examples

This section presents three examples that show how to implement various forms of regression analysis in Excel. The topics considered are linear regression, nonlinear regression, and analysis of factorial design. All examples are based on real data obtained from experiments. The procedures use the appropriate templates for

Table 8.3 Fitting the virial equation (Excel example)

m_2 (millimol/kg solv)	m_3 (millimol/kg solv)	k_c (milliosm/kg solv)	Π (milliosm/kg solv)
0	0.0000	0	0
600	0.0390	1,052	1,314
1,268	0.0823	2,326	2,267
2,013	0.1307	3,879	3,712
2,852	0.1852	5,792	5,496
3,803	0.2469	8,170	8,035
4,889	0.3175	11,161	11,513

solving the problem faster. The final form of the spreadsheet including all required information is provided as a reference for the user.

8.7.1 Linear Regression Example in Excel

This example examines the problem of fitting a theoretical equation to experimental data in order to obtain the values of the different constants in the system. Detailed information about the problem can be found in Prickett et al. (2011); Elliott et al. (2007); Prickett et al. (2010); and Jochem and Körber (1987). Data provided courtesy of Dr. Richelle Prickett.

8.7.1.1 Problem Statement for Linear Regression Example

Consider the problem of obtaining the values of the parameters in a theoretical equation that describes the osmotic pressure of the sodium chloride (NaCl) salt and hydroxyethyl starch (HES, chemical formula $(C_6H_{10}O_5)_m(C_2H_5O)_n$). Based on the virial equation of state, it is assumed that the following equation can be used to describe the osmolality (Π) of such a mixture:

$$\Pi = B_3 m_3^2 + B_3 k_{diss} m_2 m_3 + C_3 m_3^3 + k_c \quad (8.1)$$

where B_3 and C_3 are the virial parameters to be determined, m_2 is the molality of NaCl in millimol/kg of solvent, m_3 is the molality of HES in millimol/kg of solvent, k_{diss} is the disassociation constant that is equal to 1.678, and k_c is a known constant that depends on the system being analysed. An experiment was run where the ratio of the mass of HES to the mass of NaCl was fixed to 0.5. The results obtained are shown in Table 8.3.

	A	B	C	D	E	F	G	H	I	J
1										
2		<i>y</i>	<i>B</i> ₃	<i>C</i> ₃	<i>f</i> ₃	residuals	<i>r</i> ²	Ordered	Z-Rank	Z-Score
3		0	0.00	0	0	0.00	0.00	5	0.366106	0.390353
4		262	39.27	5.93E-05	-28.03735	-290.04	84121.66	1	-1.46523	-2.180899
5		-59	175.12	0.000557	-102.0621	-43.06	1854.34	2	-0.79164	0.008597
6		-167	441.50	0.002233	-192.4215	-25.42	646.25	3	-0.36611	0.164986
7		-296	886.34	0.006352	-239.621	56.38	3178.59	7	1.465234	0.890167
8		-135	1575.64	0.015051	-131.135	3.86	14.94	6	0.791639	0.424617
9		352	2604.79	0.032006	342.05393	-9.95	98.92	4	0	0.302179
10		<i>β̂</i>	-0.832522	78441.55		sum(<i>r</i> ²)	89914.71		<i>μ</i> _{residuals}	-44.03186
11		<i>±δ</i>	0.637833	56795.37		<i>σ</i> _{model}	134.10		<i>σ</i> _{residuals}	112.8
12						<i>α</i>	0.05			
13						tinv(<i>α</i>)	2.570582			
14						<i>m</i>	7			
15						<i>n</i>	2			
16						<i>k</i>	2			
17						TSS	329494.9			
18						SSE	89914.71			
19						SSR	239580.2			
20						<i>R</i> ²	0.727113			
21						<i>F</i> -test	6.661317			
22						<i>F</i> -critical, model	5.786135			
23						<i>F</i> -critical, factorial	6.607891			

Fig. 8.28 Linear regression example: Data Analysis results

8.7.1.2 Problem Solution for Linear Regression Example

Before linear regression can be applied, the above equation must be rearranged so that all known constant information is on the left-hand side and all the unknown variables are on the right-hand side. Thus, the equation would be rewritten as

$$\Pi - k_c = B_3(m_3^2 + k_{diss}m_2m_3) + C_3m_3^3 \tag{8.2}$$

The required variables would be defined as

$$\begin{aligned} y &= \Pi - k_c \\ \vec{x} &= \langle m_3^2 + k_{diss}m_2m_3, m_3^3 \rangle \\ \vec{\beta} &= \langle B_3, C_3 \rangle^T \end{aligned} \tag{8.3}$$

Entering all the data in the Excel spreadsheet gives the results shown in Figs. 8.28 and 8.29.

Using the original data shows that the second point ($\Pi = 1,314$) is potentially an outlier, since its residual is extremely large. Thus, the row corresponding to this point (row 4 in the original layout) was deleted and the regression analysis was redone. The results are shown in Figs. 8.30 and 8.31. The results are much better as there are now no clear outliers and the data confidence intervals, especially for C_3 , are much smaller.

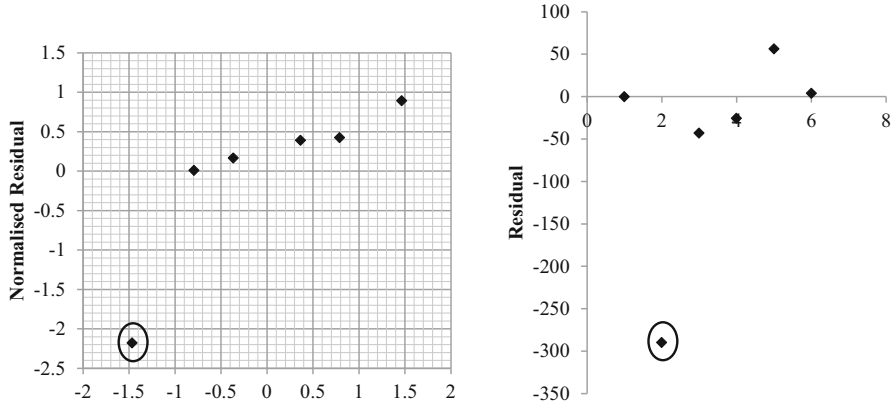


Fig. 8.29 (Left) Linear regression example: normal probability and (right) time series plots. The circled point is a potential outlier

	A	B	C	D	E	F	G	H	I	J
1										
2		y	B_3	C_3	\hat{f}	residuals	r^2	Ordered	Z-Rank	Z-Score
3		0	0.00	0	0	0.00	0.00	5	0.67449	0.231205
4		-59	175.12	0.000557	-106.3545	-47.35	2242.45	1	-1.38299	-1.24741
5		-167	441.50	0.002233	-200.7887	-33.79	1141.68	2	-0.67449	-0.82383
6		-296	886.34	0.006352	-250.8708	45.13	2036.65	6	1.38299	1.640339
7		-135	1575.64	0.015051	-139.9814	-4.98	24.81	3	-0.21043	0.075665
8		352	2604.79	0.032006	348.5676	-3.43	11.78	4	0.21043	0.124029
9		$\hat{\beta}$	-0.86648	81408.59		sum(r^2)	5457.37		$\hat{\mu}_{residuals}$	-7.40462
10		$\pm\delta$	0.190134	16929.1		σ_{model}^2	36.94		$\sigma_{residuals}^2$	32.02622
11						α	0.05			
12						tnv(α)	2.776445			
13						m	6			
14						n	2			
15						k	2			
16						TSS	245610.8			
17						SSE	5457.365			
18						SSR	240153.5			
19						R^2	0.97778			
20						F -test	88.01077			
21						F -critical, model	6.944272			
22						F -critical, factoria	7.708647			

Fig. 8.30 Linear regression example: Data Analysis results after removing the outlier

8.7.2 Nonlinear Regression Example in Excel

This example examines the problem of fitting a theoretical equation to experimental data in order to obtain the values of the different constants in the system. Unlike the previous case, nonlinear regression must be performed in order to obtain a result. Detailed information about the problem can be found in Ross-Rodriguez (2009). Data provided courtesy of Dr. Lisa Ross-Rodriguez.

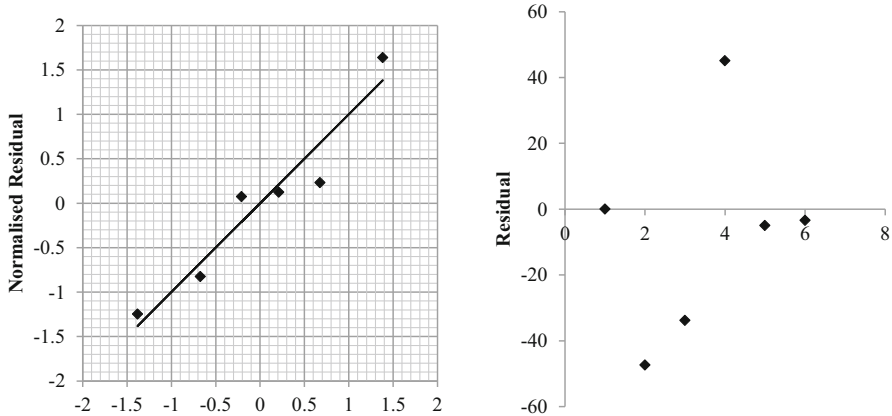


Fig. 8.31 Linear regression example: (left) normal probability and (right) time series plots after removing outliers

Table 8.4 Equilibrium cell volume data (Excel example)

V/V_0	Π
1.00034	0.29278
0.80465	0.57172
0.75358	0.85514
0.71548	1.13595
0.68588	1.43349
0.66600	1.72908
0.65913	2.02815
0.64004	2.32660
0.62661	2.66704

8.7.2.1 Problem Statement for Nonlinear Regression Example

Consider the problem of obtaining a relationship for the ratio between the equilibrium and isotonic cell volumes given the osmotic pressure. The theoretical relationship can be written as

$$\frac{V}{V_0} = (1 - b^*) \frac{-1 + \sqrt{1 + 4B\Pi_0}}{-1 + \sqrt{1 + 4B\Pi}} + b^* \tag{8.4}$$

where both B and b^* are the parameters to be determined and Π_0 is a known osmotic value. The experimental data are provided in Table 8.4. For this data set, Π_0 has a value of 0.293.

8.7.2.2 Problem Solution for Nonlinear Regression Example

Before we set up the problem in Excel, it is first necessary to compute some preliminary information. First, we need to obtain the derivatives of Eq. (8.4) with respect to the parameters, that is,

	A	B	C	D	E	F	G	H	I	J	K
1				Jacobian							
2		V/V_0	Π	$\partial y/\partial \beta_1$	$\partial y/\partial \beta_2$	β	residuals	r^2	Ordered	Z-Rank	Z-Score
3		1.0003	0.29278	0.00	-0.00001	1.0003	0.00	0.00	5	0	-0.056022
4		0.8047	0.57172	0.38	0.007262	0.8174	0.01	0.00	9	1.593219	1.924833
5		0.7536	0.85514	0.53	0.008435	0.7472	-0.01	0.00	1	-1.59322	-1.028294
6		0.7155	1.13595	0.61	0.008527	0.7095	-0.01	0.00	2	-0.96742	-0.975596
7		0.6859	1.43349	0.66	0.008329	0.6840	0.00	0.00	4	-0.28222	-0.335861
8		0.6660	1.72908	0.70	0.008053	0.6664	0.00	0.00	6	0.282216	0.015948
9		0.6591	2.02815	0.73	0.007763	0.6531	-0.01	0.00	3	-0.58946	-0.972668
10		0.6400	2.3266	0.75	0.007486	0.6429	0.00	0.00	7	0.589456	0.3941
11		0.6266	2.66704	0.77	0.007194	0.6336	0.01	0.00	8	0.967422	1.03356
12			β'	0.524581	2.408129		sum(r^2)	0.00		$\mu_{residuals}$	0.000291
13			$\pm \delta$	0.043624	3.616267		σ_{model}	0.01		$\sigma_{residuals}$	0.006449
14							α	0.05			
15							finv(α)	2.364624			
16							m	9			
17							n	2			
18							k	2			
19							TSS	0.109233			
20							SSE	0.000333			
21							SSR	0.1089			
22							R^2	0.996947			
23							F-test	1143.092			
24							F-critical, model	4.737414			
25							F-critical, factorial	5.591448			

Fig. 8.32 Nonlinear regression example: Excel spreadsheet results

$$\frac{d(V/V_0)}{db^*} = 1 - \frac{1 - \sqrt{1 + 4B\Pi_0}}{1 - \sqrt{1 + 4B\Pi}} \tag{8.5}$$

$$\frac{d(V/V_0)}{dB} = 2(1 - b^*) \left[\frac{\Pi_0}{\sqrt{1 + 4B\Pi_0}(-1 + \sqrt{1 + 4B\Pi})} - \frac{\Pi(-1 + \sqrt{1 + 4B\Pi_0})}{\sqrt{1 + 4B\Pi}(-1 + \sqrt{1 + 4B\Pi})^2} \right] \tag{8.6}$$

It can clearly be seen that this equation is nonlinear in the *parameters*. Thus, nonlinear regression using Solver will be performed. In order to obtain values for the parameter confidence intervals using Equation (198), the grand Jacobian will be calculated using the “best” estimated values of the parameters and the above derivatives.

The nonlinear regression Excel template used is set up identically to that of the linear regression template. The only difference is that now the estimated parameter values are not computed using a formula. Instead, they must be determined using Solver. Given the problem set-up, initial parameter estimates can be a bit of an issue, as the solution is sensitive to them. A recommended initial guess would be 0.5 for b^* and 2.5 for B . The macros are shown in Sect. 8.7.2.3: VB Macros.

The final results are shown in Fig. 8.32. Figure 8.33 shows the normal probability plot and a time series plot of the residuals. It is easy to note that the B parameter is not significant and its value could be zero. Given the overall good fit and the relative well-behaved nature of the residuals, this would suggest that potentially not enough data have been collected to make an appropriate estimate. This situation partly explains why the Solver can have issues with obtaining a good value for B . The residual plots are shown in Fig. 8.33. Overall the results are decent,

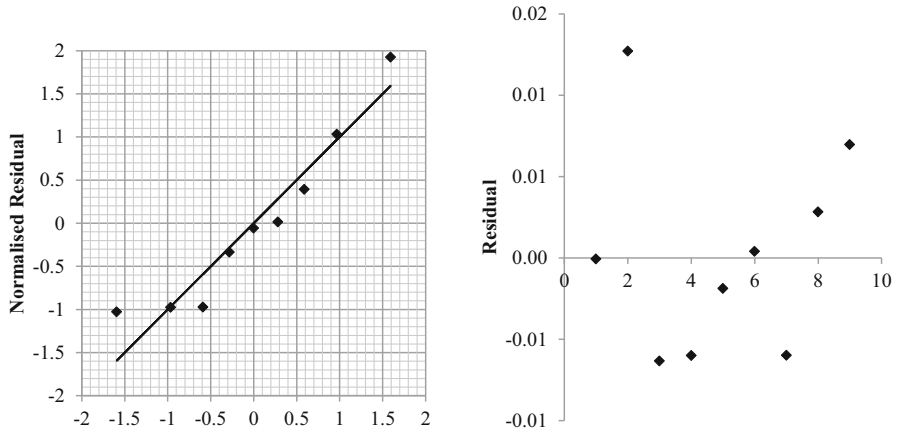


Fig. 8.33 Nonlinear regression example: (*left*) normal probability plot and (*right*) time series plot of the residuals

given the small sample. Since it has been assumed that the given equation holds, in order to obtain a better understanding of the data, additional experiments should be performed.

8.7.2.3 VB Macros

The macros required for performing the detailed regression analysis are shown here. There are three main macros: (1) to compute the model parameters (`model`), (2) to compute the derivative of the model with respect to b^* (`dydb1`), and (3) to compute the derivative of the model with respect to B (`dydb3`).

```
Public Function model(parameter, x)
    bs = parameter(1)
    B = parameter(2)
    model = (1 - bs) * (-1 + Sqr(1 + 4 * B * 0.293)) / (-1 + Sqr(1 + 4 * B * x(1)))
    + bs
End Function
```

```
Function dydb1(parameter As Range, x As Range)
    bs = parameter(1)
    B = parameter(2)
    dydb1 = 1 - (-1 + Sqr(1 + 4 * B * 0.293)) / (-1 + Sqr(1 + 4 * B * x(1)))
End Function
```



```

Function dydb3 (parameter As Range, x As Range)
bs = parameter (1)
B = parameter (2)
Pio = 0.293
so = Sqr (1 + 4 * B * Pio)
s = Sqr (1 + 4 * B * x (1))
dydb3 = 2 * (1 - bs) * (Pio / so / (s - 1) - x (1) * (so - 1) / s / (s - 1) ^ 2)
End Function

```

8.7.3 Factorial Design Examples Using Excel

This section presents the Excel spreadsheets for analysing some of the factorial design experiments presented in Chap. 4. The examples are all based on the factorial design template. The following examples have a corresponding Excel spreadsheet:

1. Figure 8.34: Example 4.2, Analysis of a Full Factorial Experiment, from Sect. 4.4.4, Projection;
2. Figure 8.35: Sect. 4.7.4, Detailed Mixed-Level Example; and
3. Figure 8.36: Sect. 4.8.2, Factorial Design with Centre Point Example.

8.8 Further Reading

The following are references that provide additional information about the topic:

1. *General Excel Help*:
 - (a) Harvey G (2013) Excel[®] 2013 All-in-One for dummies. Wiley, Hoboken
 - (b) Brillo J (2007) Excel for scientists and engineers: numerical methods. Wiley, Hoboken
 - (c) Schmuller J (2013) Statistical analysis with Excel[®] for dummies, 3rd edn. Wiley, Hoboken
2. *Linear Regression Data Set*:
 - (a) Elliott JA, Prickett RC, Elmoazzen HY, Porter KR, McGann LE (2007) A multisolute osmotic virial equation for solutions of interest in biology. J Phys Chem B 111:1775–1785
 - (b) Prickett RC, Elliott JA, McGann LE (2010) Application of the osmotic virial equation in cyrobiology. Cryobiology 2010:30–42

	A	B	C	D	E	F	G	H	I	J	K	L	M
1													
2		y	β_0	β_1	β_2	β_{12}	β_ϵ	y	residuals	r^2	Ordered	Z-Rank	Z-Score
3		39.3	1	-1	-1	1	1	39.3	0.00	0.00	4	-0.2822	0
4		40	1	-1	1	-1	1	40	0.00	0.00	4	-0.2822	0
5		40.9	1	1	-1	-1	1	40.9	0.00	0.00	4	-0.2822	0
6		41.5	1	1	1	1	1	41.5	0.00	0.00	4	-0.2822	0
7		40.3	1	0	0	0	-0.8	40.46	0.16	0.03	8	0.96742	1.09119
8		40.5	1	0	0	0	-0.8	40.46	-0.04	0.00	3	-0.5895	-0.2728
9		40.7	1	0	0	0	-0.8	40.46	-0.24	0.06	1	-1.5932	-1.63679
10		40.2	1	0	0	0	-0.8	40.46	0.26	0.07	9	1.59322	1.77319
11		40.6	1	0	0	0	-0.8	40.46	-0.14	0.02	2	-0.9674	-0.95479
12		β^*	-40.4444	0.775	0.325	-0.025	-0.01944	sum(r^2)	0.17		$\hat{\mu}_{residuals}$		0
13		$\pm\delta$	0.19191	0.287868	0.287868	0.287868	0.21456	σ_{model}^2	0.21		$\hat{\sigma}_{residuals}^2$	0.14663	
14		Factorial Analysis of Parameters						α	0.05				
15		Ordered	5	4	3	1	2	finv(α)	2.77645				
16		Z-Rank	1.28155	0.524401	0	-1.28155	-0.5244	m	9				
17		Z-Score	1.78856	-0.4187	-0.44374	-0.46321	-0.4629	n	5				
18		SSR _i	14721.8	2.4025	0.4225	0.0025	0.00272	n_R	2				
19		F_1	342367	55.87209	9.825581	0.05814	0.06331	k	4				
20		Decision	Keep	Keep	Keep	Discard	Discard	TSS	3.00222				
21								SSE	0.172				
22								SSR	2.83022				
23		$\hat{\mu}_{parameters}$	8.3					R^2	0.94271				
24		$\hat{\sigma}_{parameters}$	17.9723					F -test	16.4548				
25								F -critical, model	6.38823				
26								F -critical, factorial	7.70865				
27													

Fig. 8.36 Factorial design: combined factorial and centre point example

- (c) Prickett RC, Elliott JA, McGann LE (2011) Application of the multisolute osmotic virial equation to solutions containing electrolytes. J Phys Chem B 115:14531–14543
- (d) Jochem M, Körber C (1987) Extended phase diagrams for the ternary solutions H₂O – NaCl – glycerol and H₂O – NaCl – hydroxyethylstarch (HES) determined by DSC. Cryobiology 24:513–536

3. Nonlinear Regression Data Set:

- (a) Ross-Rodriguez LU (2009) Cellular osmotic properties and cellular responses to cooling. University of Alberta, Edmonton