



*A package for simulation, curve fitting,
statistical analysis, and graph plotting.*

<https://simfit.uk>

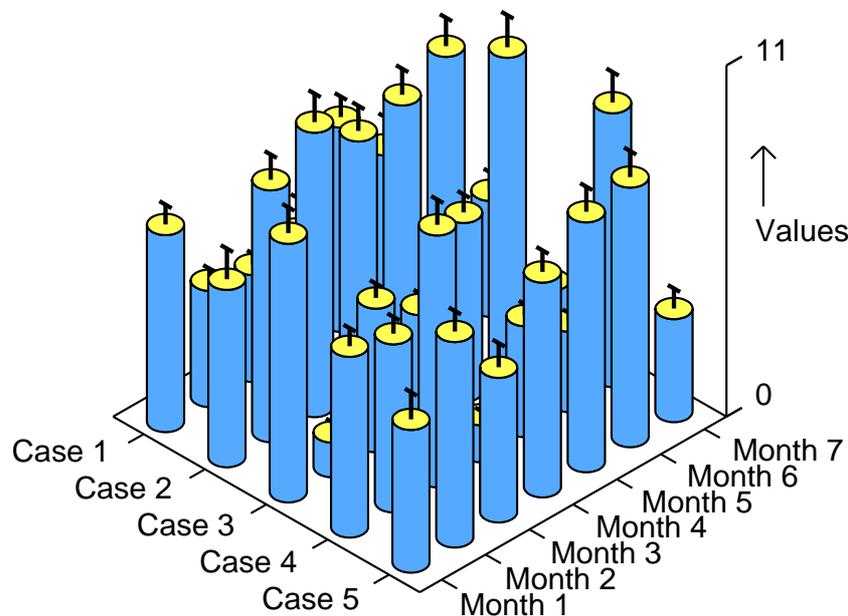
<https://simfit.org.uk>

<https://simfit.silverfrost.com>

SIMFIT, Microsoft Office, and LibreOffice.

Bill Bardsley, Francisco Burguillo, John Clegg, and Abdul Sattar

Simfit Cylinder Plot with Error Bars



Contents

1	The easy way to import data into SIMFIT	1
1.1	Creating a pie chart	1
1.2	Creating a bar chart	4
2	Definitions	5
2.1	Data tables	6
2.2	Labeled data tables	6
2.3	Missing values	6
2.4	SIMFIT data files	6
2.5	SIMFIT data files with labels	7
2.6	Clipboard data	7
2.7	Files exported from spreadsheet programs	7
3	Spreadsheet tables	8
4	Using the clipboard to transfer data into SIMFIT	10
4.1	Pasting data from the clipboard directly into SIMFIT	10
4.2	Converting data from the clipboard into a SIMFIT file	10
5	Using spreadsheet output files to transfer data into SIMFIT	11
5.1	Space-delimited text files (.txt)	11
5.2	Comma-delimited text files (.csv with standard Excel setup)	11
5.3	Semicolon-delimited text files (.csv with continental Excel setup)	11
5.4	Tab-delimited files (.txt)	11
5.5	Unicode (.txt)	11
5.6	Web documents (.xml, .html, .htm, .mht, .mhtml)	11
6	Exporting SIMFIT data files from Excel	13
6.1	Using simfit4.xls with Excel to create SIMFIT data files	13
6.2	Using simfit6.xls with Excel to create transformed SIMFIT data files	13
6.2.1	Invoking the simfit6.xls macro	13
6.2.2	Step 1: Open the simfit6.xls workbook	13
6.2.3	Step 2: Select the data table within the user's workbook	13
6.2.4	Step 3: Invoke the simfit6.xls macro	14
6.2.5	Using the simfit6.xls macro	14
6.2.6	Filling empty cells found in the data table	14
6.2.7	Validation Checks Completed	15
6.2.8	Performing transformations of the data table	15
6.2.9	Transposing the SIMFIT table	15
6.2.10	Inspecting and saving the modified worksheet	16
6.2.11	The History Log	16
6.2.12	Processing very large data tables	16
7	Importing SIMFIT results tables into documents and spreadsheets	17
7.1	SIMFIT results files	17
7.2	Preliminary analysis	17
7.3	The procedure	18
7.4	Example 1	20
7.5	Example 2	22
7.6	Example 3	23
7.7	Summary	25

8	Printing and importing <code>SimFIt</code> graphs into documents	26
8.1	Graphics hardcopy	26
8.1.1	Bitmaps and compressed bitmaps	26
8.1.2	Enhanced metafiles (.emf)	26
8.1.3	Scalable vector graphics (.svg)	26
8.2	PostScript graphics files (.eps)	26
8.2.1	Ghostscript generated files	27
8.2.2	Portable Document graphics files (.pdf)	27
8.2.3	Portable network graphics files (.png)	27
8.3	Using Encapsulated PostScript (.eps) files directly	27

1 The easy way to import data into SIMFIT

Often the easiest way to use data from from a spreadsheet program, word processor, or text editor in SIMFIT is to use the clipboard which will now be illustrated by two simple examples. However, for repeated analysis or where extra features are required, it is better to create a SIMFIT style data file, and this will be explained subsequently.

1.1 Creating a pie chart

This example illustrates how to plot a relatively small sample of positive numbers in pie chart format.

First of all open SIMFIT to obtain the menu displayed in figure 1 which will be referred to as the main SIMFIT menu.

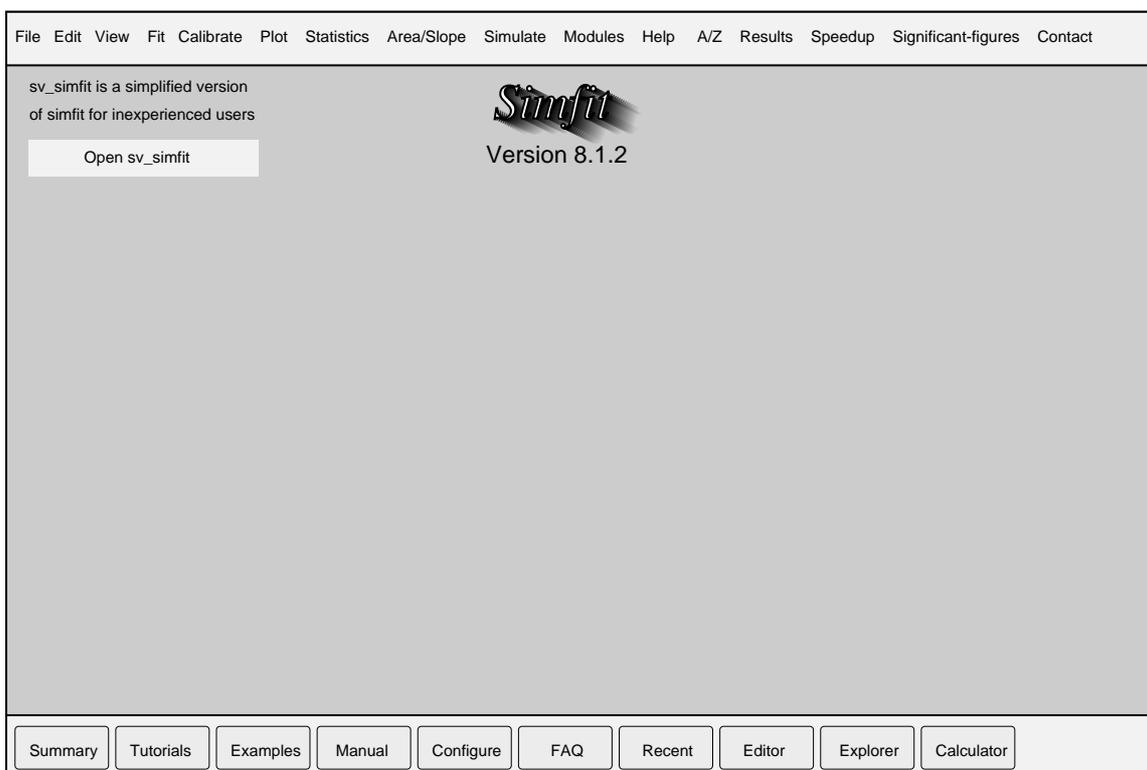


Figure 1: The main SIMFIT menu

Now click on the [Editor] button which will open your text editor, and type in a data vector consisting of a single column of nonnegative numbers as in

```
1
2
3
4
5.
```

Then select, that is highlight, the column and copy to the clipboard as follows.

1
2
3
4
5

This data vector has been stored for further use so you can now click on the [A/Z] button to get an alphabetical list of SIMFIT programs and select program **simplot**. Choose to create a pie chart and you will then reach the SIMFIT file open control shown in figure 2 where the [Paste] button will not be greyed out, showing that

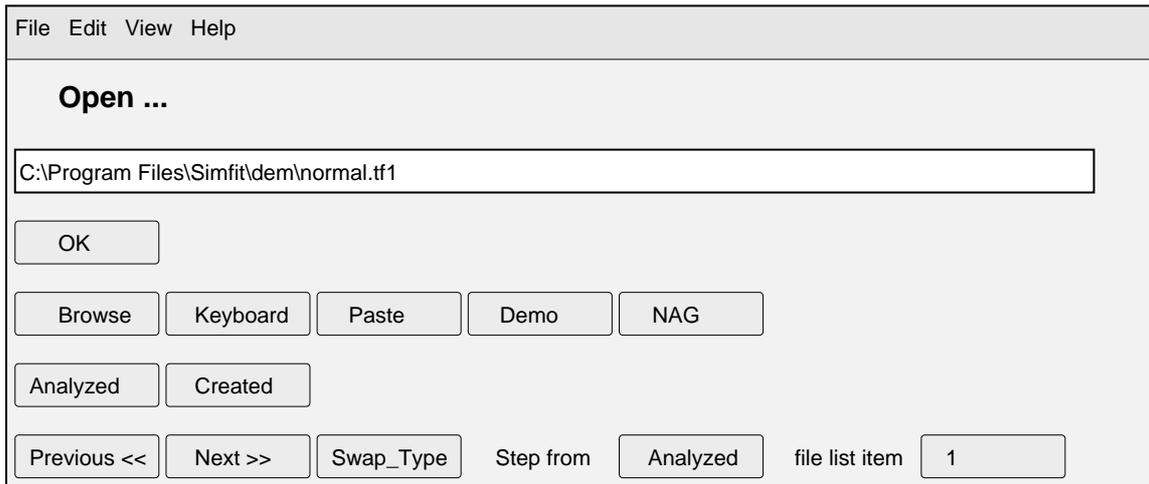


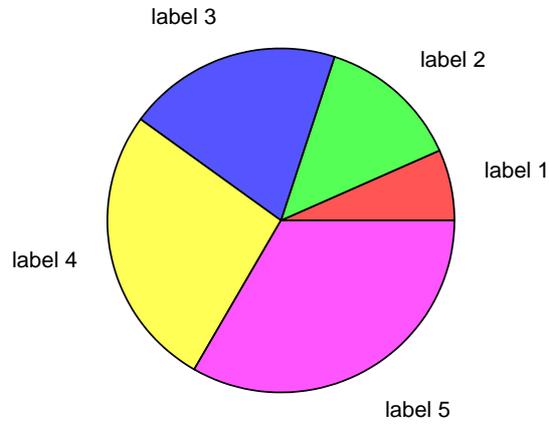
Figure 2: The SIMFIT file open control

SIMFIT has recognized the data vector that has been copied to the clipboard.

Note that, until you copy something else to the clipboard, this data set will remain ready for re-use by any SIMFIT procedure, and on pressing the [Paste] button SIMFIT gives you the option to create a temporary SIMFIT data file then proceed with further analysis.

First a default pie chart will be displayed which can then be edited if required, as in figure 3 where the SIMFIT segment fill styles are illustrated.

Temporary Simfit data file



Demonstrating Pie Chart Fill Styles

Data Vector $x = (1, 2, 3, 4, 5)$

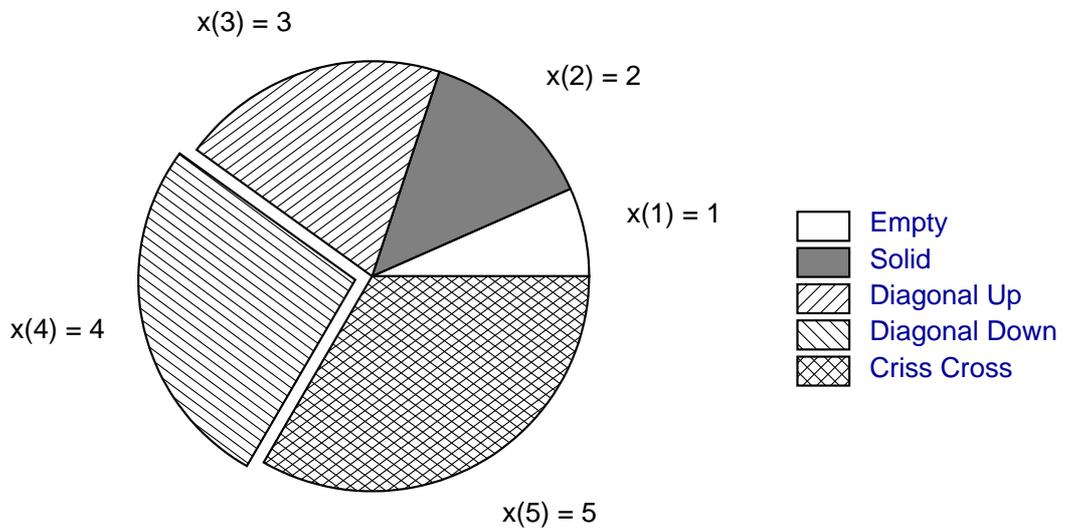


Figure 3: Creating a SIMFIT pie chart

1.2 Creating a bar chart

This example explains how to input a data matrix with row and column labels into SIMFIT in order to plot a simple bar chart, and also how to save the data matrix in SIMFIT data file format for re-use.

Type the following table into the Editor as for a pie chart, or highlight and copy to the clipboard from your spreadsheet. Note that each row and column has a label, also there is a dummy label in position 1,1 and labels must not contain spaces. For instance, you can use Time_of_Day as a label but not Time of Day.

name/variable	one	two	three
Fred	2	3	1
Mary	1	2	3
Jim	4	6	1
Jean	3	7	2
Anne	4	4	2
Kevin	5	1	3

After choosing the barchart option in program **simplot**, the clipboard data will be written to a temporary file called `clipboard_n.tmp` in your `ProgramData\simfit\you\res SIMFIT results` folder, accessible subsequently after pressing the main menu [Recent] button so you can save it for future re-use. The temporary SIMFIT file will look like this

```
Temporary Simfit data file
 6 3
2 3 1
1 2 3
4 6 1
3 7 2
4 4 2
5 1 3
 12
begin{labels}
Fred
Mary
Jim
Jean
Anne
Kevin
one
two
three
end{labels}
ccyyymmdd=20130703, hhmss=085959
```

indicating the title, number of rows and columns, the data matrix, then the number of extra lines followed by row then column labels, and finally the date. Figure 4 shows the resulting bar chart after preliminary then more extensive editing.

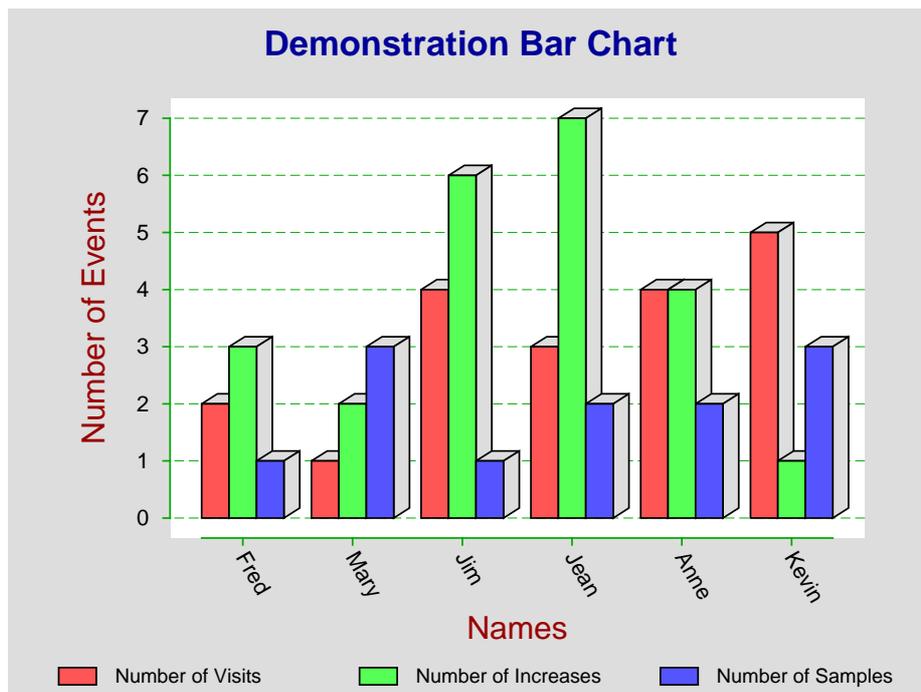
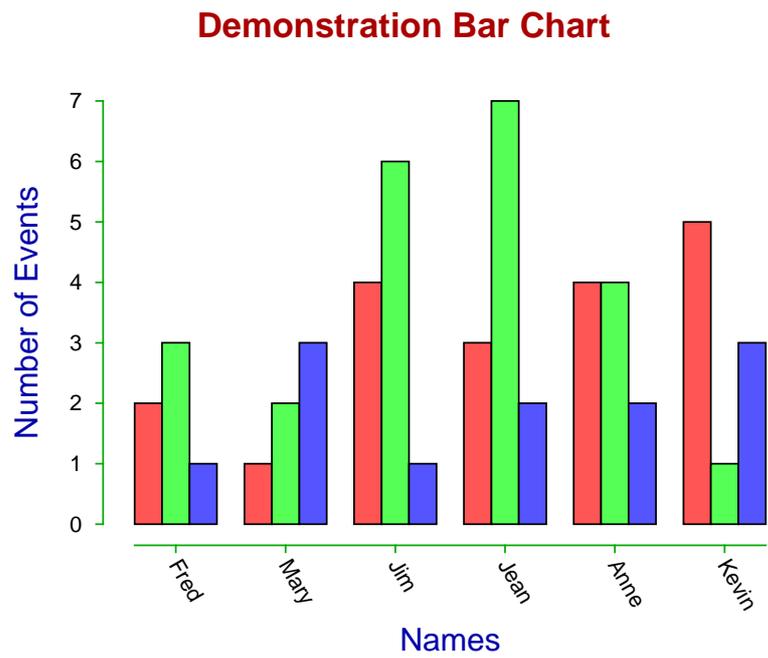


Figure 4: Creating a SIMFIT bar chart

2 Definitions

Here we summarize the definitions to be used in this document.

2.1 Data tables

A data table is a n by m rectangular set of cells which contain only numerical values, as in the following example of a 2 by 3 data matrix.

```
1.1  1.2  1.3
2.1  2.2  2.3
```

Here the the columns are space-delimited, i.e., separated by one or more spaces, but we can also have comma-delimited tables, with zero or more spaces in addition to the commas, as shown next.

```
1.1,  1.2,  1.3
2.1,  2.2,  2.3
```

In spreadsheet and word-processing programs, data tables are stored in tab-delimited format, which results in tables being displayed with cells outlined by lines, as follows.

1.1	1.2	1.3
2.1	2.2	2.3

2.2 Labeled data tables

A labeled data table employs row 1 for column labels and column 1 for row labels with a dummy label in cell(1,1) as illustrated below.

```
   R/C  Col-1  Col-2  Col-3
Row-1   1.1   1.2   1.3
Row-2   2.1   2.2   2.3
```

Here labels are shown as character strings, and the main table contains only numerical data. Note that there can be no spaces in labels of space-delimited tables, and underscores, hyphens, or similar must be used to avoid ambiguities. Spaces within labels also cause problems with importing SIMFIT results log files into Word and Excel.

2.3 Missing values

Missing values arise because data values are misplaced, not recorded, or discarded and this leads to a deficient table. The next example illustrates how a missing value can be indicated by an empty cell or a character variable in a data cell.

1.1		1.3
2.1	2.2	X

Any non-numerical characters can be used to indicate missing values, and pre-processing is required to replace empty cells by estimates derived in some way from the rest of the data set, before SIMFIT can process the data.

2.4 SIMFIT data files

SIMFIT data files can have an optional header with title on line 1, and number of rows and columns on line 2, then a data section table of numerical values, with no missing values, as follows.

```
Title for data
  2   3
1.1  1.2  1.3
2.1  2.2  2.3
```

2.5 SIMFIT data files with labels

For some purposes row and column labels may be appended to SIMFIT data files using a special symbolism as follows.

```
Title for data
  2,    3
1.1,  1.2,  1.3
2.1,  2.2,  2.3
7
begin{labels}
Row-1
Row-2
Col-1
Col-2
Col-3
end{labels}
```

The number 7 after the data table indicates that there are 7 extra lines appended, but this extra counter is optional. Note that the label in cell(1,1) is not recorded, and the labels are in sequential order with rows followed by columns.

Note that SIMFIT data files are created automatically by all of the various methods described in the following sections, so that the user need only create a data table or labeled data table before using a particular method.

2.6 Clipboard data

The only type of clipboard data that can be used by SIMFIT is where a data table or labeled data table has been selected and copied to the clipboard. If a space-delimited labeled table has been copied to the clipboard from a text editor like Notepad, there must be no spaces within labels, so use underscores.

2.7 Files exported from spreadsheet programs

The only types of files exported from spreadsheet programs that can be used by SIMFIT are data tables or labeled data tables exported in text, XML, or HTML format. Text files can be delimited by spaces, tabs, commas, or semicolons, but files exported in space-delimited text format must have no spaces within labels, so use underscores.

3 Spreadsheet tables

Data for analysis by SIMFIT would usually be held in a spreadsheet program, like Microsoft Office Excel or LibreOffice Calc, as in this example of an unselected table of multivariate statistical data from K.R. Gabriel in *Biometrika* 1971, **58**, 453–67.

Percent	Christian	Armenian	Jewish	Moslem	American	Shaafat	A-Tur	Silwan	Sur-Bahar
Toilet	98.2	97.2	97.3	96.9	97.6	94.4	90.2	94	70.5
Kitchen	78.8	81	65.6	73.3	91.4	88.7	82.2	84.2	55.1
Bath	14.4	17.6	6	9.6	56.2	69.5	31.8	19.5	10.7
Electricity	86.2	82.1	54.5	74.7	87.2	80.4	68.6	65.5	26.1
Water	32.9	30.3	21.1	26.9	80.1	74.3	46.3	36.2	9.8
Radio	73	70.4	53	60.5	81.2	78	67.9	64.8	57.1
TV set	4.6	6	1.5	3.4	12.7	23	5.6	2.7	1.3
Refrigerator	29.2	26.3	4.3	10.5	52.8	49.7	21.7	9.5	1.2

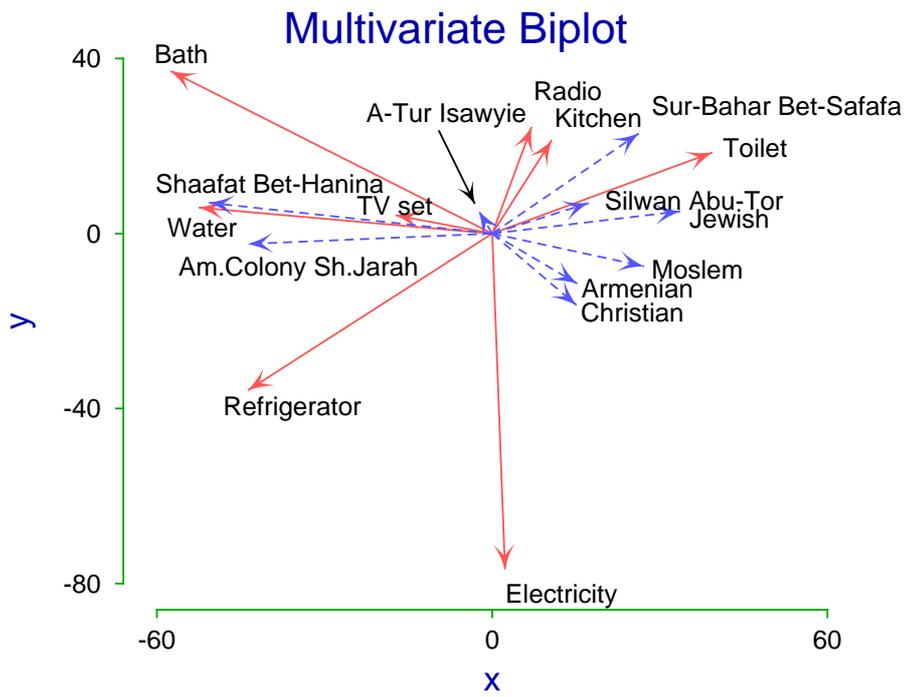
Such tables must be rectangular, with optional row and column labels, and all other cells filled with numerical data (missing data will be discussed later). Often it would only be necessary to select cells containing numerical values from such a table by highlighting as follows.

Percent	Christian	Armenian	Jewish	Moslem	American	Shaafat	A-Tur	Silwan	Sur-Bahar
Toilet	98.2	97.2	97.3	96.9	97.6	94.4	90.2	94	70.5
Kitchen	78.8	81	65.6	73.3	91.4	88.7	82.2	84.2	55.1
Bath	14.4	17.6	6	9.6	56.2	69.5	31.8	19.5	10.7
Electricity	86.2	82.1	54.5	74.7	87.2	80.4	68.6	65.5	26.1
Water	32.9	30.3	21.1	26.9	80.1	74.3	46.3	36.2	9.8
Radio	73	70.4	53	60.5	81.2	78	67.9	64.8	57.1
TV set	4.6	6	1.5	3.4	12.7	23	5.6	2.7	1.3
Refrigerator	29.2	26.3	4.3	10.5	52.8	49.7	21.7	9.5	1.2

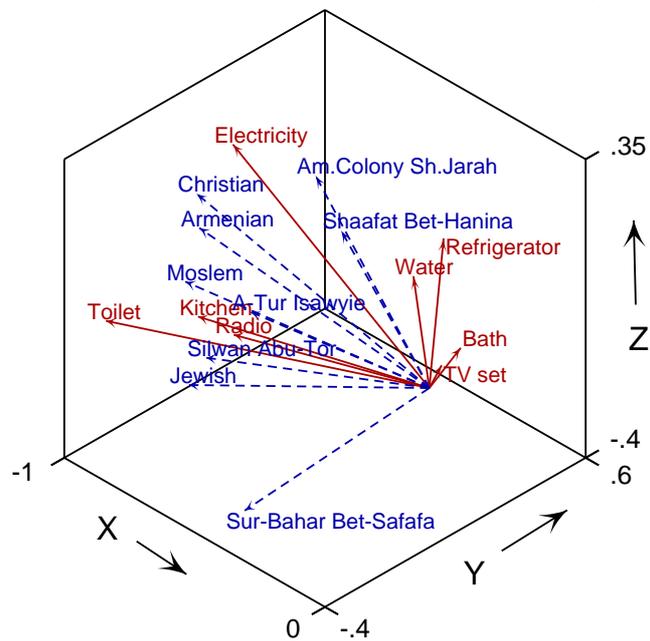
However, sometimes row and column labels could also be needed, when a labeled table with cells containing either labels or numerical values would be selected, as follows.

Percent	Christian	Armenian	Jewish	Moslem	American	Shaafat	A-Tur	Silwan	Sur-Bahar
Toilet	98.2	97.2	97.3	96.9	97.6	94.4	90.2	94	70.5
Kitchen	78.8	81	65.6	73.3	91.4	88.7	82.2	84.2	55.1
Bath	14.4	17.6	6	9.6	56.2	69.5	31.8	19.5	10.7
Electricity	86.2	82.1	54.5	74.7	87.2	80.4	68.6	65.5	26.1
Water	32.9	30.3	21.1	26.9	80.1	74.3	46.3	36.2	9.8
Radio	73	70.4	53	60.5	81.2	78	67.9	64.8	57.1
TV set	4.6	6	1.5	3.4	12.7	23	5.6	2.7	1.3
Refrigerator	29.2	26.3	4.3	10.5	52.8	49.7	21.7	9.5	1.2

As an example, consider the next figures, showing these data from the SIMFIT test file `houses.tf1` when displayed as biplots. The dummy label in cell(1,1) is not used.



Three Dimensional Multivariate Biplot



This document explains how to transfer data from such spreadsheet tables into SIMFIT.

4 Using the clipboard to transfer data into SIMFIT

Two formats can be used to transfer a tabular data set into SIMFIT for analysis.

- The cells contain only numerical data
- The cells also have associated row and column labels

If labels are present then the column labels (usually variables) must occupy the top row of cells, the row labels (usually cases) must occupy the first column of cells, and there must be a dummy row/column label in cell(1,1). Purely numerical tables are used for curve fitting and statistical analysis, where the identity of the observations is immaterial. Row and column labels can be added just to help identify the data retrospectively, but they are also used in multivariate statistics to identify plotting symbols or label dendrograms. If there are no empty cells, such tables can be imported directly into SIMFIT by a [Paste] button, or be transformed into SIMFIT data files using program **maksim**.

Cells can be delimited by spaces, commas, semicolons, or tabs, and there must be no empty cells. If semicolons or tabs are used as column separators, then using continental notation like 1,5 for 1.5 is allowed. Further, if a space-delimited table has been copied to the clipboard from a text editor like Notepad, there must be no spaces within labels.

4.1 Pasting data from the clipboard directly into SIMFIT

In Microsoft Excel, LibreOffice Calc, or almost any text editor (such as Notepad) or word processing program, the rectangular table required is selected then copied to the clipboard. The appropriate SIMFIT program is opened and, when a data set is requested, the [Paste] button on the file opening control is pressed. If the table is rectangular, with no missing values, one of the following options will be available.

- Data: use as a SIMFIT data file (no labels)
- Data: use as a SIMFIT data file (with labels)

If the option to use as a SIMFIT data file is chosen from this SIMFIT clipboard control, the data will be written to a temporary file then analyzed. The names of these temporary files are displayed so they can be saved retrospectively. The options will only be available if the data are consistent with the SIMFIT format, otherwise error messages will be displayed.

4.2 Converting data from the clipboard into a SIMFIT file

This involves the use of the SIMFIT program **maksim**. The clipboard table is input into **maksim** exactly as just described for any SIMFIT program, but then there are many options.

- Rows or columns selected from a check list can be suppressed or restored
- Sub-tables can be selected by properties
- Sub-tables can be Saved As ... SIMFIT data files.

The advantage of this technique is that, from a single table copied to the clipboard, numerous selected sub-tables can be archived, so that retrospective analysis of data sub-sets can be performed.

5 Using spreadsheet output files to transfer data into SIMFIT

SIMFIT will accept application files (created by any of the data preparation programs) containing rectangular data tables with or without labels, but with no missing values, and the exported files will be transformed interactively into temporary SIMFIT data files. Otherwise, the exported files can be transformed into SIMFIT data files using program **maksim**, in which case the options just described for clipboard data will also be available. Care is needed when using continental notation like 1,5 for 1.5 in text files with spaces or commas as delimiters, as mentioned below. Spreadsheets with missing values must be pre-processed, e.g., using the macro **simfit6.xls** which is in the C:\Program Files\simfit\doc installation folder.

5.1 Space-delimited text files (.txt)

These files must not have spaces to represent empty cells, or spaces within labels. For example, use TV_set instead of TV set. Using 1,5 for 1.5 is not acceptable with this file type, since commas are interpreted as column separators.

5.2 Comma-delimited text files (.csv with standard Excel setup)

If it is wished to use commas to separate thousands, i.e., triples of figures before the decimal point, then the cell must be quoted. That is, writing 1,234.56 as "1,234.56" to represent 1234.56. This has to be done explicitly in text editors like Notepad, but Excel and Calc will add the quotes. Note that, when the SIMFIT file is output, it will have 1234.56 not 1,234.56. Any comma pairs ,, will be taken to represent blank cells and be expanded to ,X,. Using 1,5 for 1.5 is not acceptable with this file type, since commas are reserved for use as column separators.

5.3 Semicolon-delimited text files (.csv with continental Excel setup)

Commas for thousands in numeric cells will still have to be quoted and semicolon pairs ;; will be taken to represent blank cells and be expanded to ;X;. Using 1,5 for 1.5 is acceptable with this file type, since semicolons are reserved for use as column separators.

5.4 Tab-delimited files (.txt)

Commas for thousands in numeric cells will still have to be quoted, and adjacent tabs will be interpreted as empty cells and be expanded to (tab)X(tab). Using 1,5 for 1.5 is acceptable with this file type, since tabs are reserved for use as column separators.

5.5 Unicode (.txt)

If SIMFIT program **maksim** fails to read Unicode files, then switch Notepad into ASCII text mode, read in the Unicode file, then export again as an ASCII text file, which will then be acceptable

5.6 Web documents (.xml, .html, .htm, .mht, .mhtml)

Using 1,5 for 1.5 is acceptable with these file types, since special techniques are used for column separators, but such files must have the file extension .xml, .html, .htm, .mht, or .mhtml.

Also note that XML files must follow this case-sensitive convention.

- The tag <Table indicates the start of a table and </Table> indicates the end.
- The tag <Row indicates the start of a row and </Row> indicates the end.
- The tag <Cell indicates the start of a new cell and </Cell> indicates the end.
- The following tokens are used:
 - ss:Index="2" to indicate an empty cell in column 1, i.e., the next cell is cell 2.
 - ss:Type="String" for a label, and
 - ss:Type="Number" for a numeric cell.

HTML files must use standard case-insensitive tags. Files exported from Microsoft Office Excel and Libre-Office Calc use these conventions.

6 Exporting SIMFIT data files from Excel

6.1 Using *simfit4.xls* with Excel to create SIMFIT data files

The *simfit4.xls* macro distributed with SIMFIT version 4 in the C:\Program Files\simfit\doc installation folder is a very simple program that remains useful, especially for exporting curve fitting files from Excel, for these reasons:

- it can gather together non-adjacent columns;
- it can check for nondecreasing values, e.g., an independent variable in column 1; and
- it can check for nonnegative values, e.g., positive weights in column 3.

However, it cannot deal with labels, missing values, or perform data transformations, so the more versatile *simfit6.xls* macro distributed with SIMFIT version 6 is now the preferred tool for creating export files in SIMFITdata file format from Excel. The *simfit6.xls* macro also validates the data, checking for basic errors that would cause SIMFIT to reject the data table.

6.2 Using *simfit6.xls* with Excel to create transformed SIMFIT data files

The *simfit6.xls* macro replicates some of the functionality of the *simfit4.xls* macro, and previous versions of the *simfit6.xls* macro, but offers additional features for filling empty cells in the data table, for performing transformations of the data table, and for transposing the data table when the SIMFIT export file is created.

In contrast with *simfit4.xls*, and previous versions of *simfit6.xls*, which did not modify the data table in the worksheet, *simfit6.xls* now applies its editing and transformations to the actual worksheet, and leaves this modified worksheet in place after the macro has terminated.

6.2.1 Invoking the *simfit6.xls* macro

The *simfit6.xls* workbook can be located in the C:\Program Files\simfit\doc folder, and in order to use the macro you may need to change the Excel macro security setting. Navigate Tools>Macro>Security from the Menu Bar and set the security level to Medium.

The *simfit6.xls* macro has been written and tested with the Microsoft Office XP Excel Version 2002, but should work satisfactorily with other versions from Excel 97 onwards.

6.2.2 Step 1: Open the *simfit6.xls* workbook

The *simfit6.xls* workbook acts purely as a carrier for the *simfit6* macro and contains no worksheets accessible to the user. The workbook must, however, be opened in the normal manner, in order to make the macro available. With the macro security level set to Medium, as described above, you will be warned that workbook contains macros. Click the Enable Macros button to complete the opening procedure. After the workbook has been opened, there is no visible sign of its presence, because it operates in the background.

6.2.3 Step 2: Select the data table within the user's workbook

Open the user's workbook containing the SIMFIT table and activate the worksheet which holds the table. The SIMFIT table (either labeled or unlabeled) can be located anywhere within the spreadsheet, i.e. it does not have to occupy Row 1 and Column 1 of the spreadsheet. Next select the SIMFIT table with the mouse in order to inform the macro of the position and dimensions of the table. Selection is carried out by left-clicking the mouse on the top left cell of the table and dragging (moving the mouse with the left key held down) to the bottom right cell of the SIMFIT table. [The above selection is called a single-area selection. The *simfit6* macro does not support multiple-area selections.]

6.2.4 Step 3: Invoke the *simfit6.xls* macro

The macro is invoked either by navigating Tools>Macro>Macros from the Menu Bar, or by Alt+F8 from the keyboard (meaning hold the Alt key down whilst pressing the F8 key). Click on *simfit6* in the list of macros, then click on the Run button, and respond to the various dialog boxes which the macro uses for communicating with the user. If *simfit6* does not appear in the list of macros, check that Step 1 has been carried out successfully.

6.2.5 Using the *simfit6.xls* macro

Since *simfit6* applies its transformations directly to the table which is selected in the active worksheet, it is necessary to take precautions to preserve the original contents of the worksheet in order to safeguard the user's data. The *simfit6* macro does this by offering the following two options to the user as soon as the macro commences:

- First, if the macro detects that the user's workbook has not been saved on disk, it offers the user the option of saving the workbook.
- The second option is to make a copy of the active worksheet within the user's workbook and to have the macro apply its transformations to this copy, leaving the original worksheet unaltered.

Next the macro ascertains from the user whether or not the selection contains a labeled table, after which it notifies the user of the data table's dimensions and obtains confirmation that a *SIMFIT* export file is to be created. Users are reminded that a *SIMFIT* labeled table must contain **both** row **and** column labels.

A *simfit6* data table must not contain any non-numeric cells or any cells containing formulas. Any non-numeric cells found in the data table are reported to the user, but their presence forces the macro to terminate, as *SIMFIT* does not accept non-numeric data items.

If any cell in the data table is found to contain a formula, this is likewise reported to the user and will force the macro to terminate. The user must replace all formulas in the data table by their numeric values before re-running the macro. [Tip: This is simply done by selecting the data table and navigating Edit>Copy, Edit>Paste Special>Values.]

6.2.6 Filling empty cells found in the data table

If any empty cells are found in the data table, the user's processing options are displayed in the Data Table contains empty cells dialog box. The options are as follows.

1. Report references of empty cells
2. Replace empty cells in rows by the row average
3. Replace empty cells in rows by by row-wise interpolation
4. Replace empty cells in columns by the column average
5. Replace empty cells in columns by by column-wise interpolation
6. Delete all rows containing an empty cell
7. Delete all columns containing an empty cell
8. Delete both rows and columns containing an empty cell

Row-wise interpolation acts as follows: an empty cell range in the middle of the row will be filled by the average of the neighboring cells adjacent to the range; and an empty cell range which extends to the beginning or end of the row will be filled by the neighboring adjacent cell to the range. Column-wise interpolation applies the same algorithm to columns.

Note that options 2 to 8 are applied to **all** rows (or columns) which contain empty cells, i.e. the user is not able to select a different option for each row or column. Note also that whereas options 2 to 8 will fill all empty cells in a data table, option 1 merely reports the empty cells and leaves the data table unchanged. In the latter case the macro will terminate after the reporting has been completed, as empty cells in the data table are not acceptable to SIMFIT.

6.2.7 Validation Checks Completed

If the data table passes the checks described above for non-numeric cells, formula cells and empty cells, the macro displays the Validation Checks Completed window.

- Perform transformations
- Create Simfit file

Clicking the Perform transformations option allows one or multiple transformations to be carried out on the data table as described below, after which the option to create the SIMFIT file will be offered.

If no transformations are required, click the Create Simfit file option to bypass the transformation process and commence creation of the SIMFIT file.

6.2.8 Performing transformations of the data table

Transformations can be selected via the Select Required Transformation dialog box, as follows.

1. Normalize columns to length 1
2. Normalize columns to standard deviation 1
3. Centralize columns to mean 0
4. Centralize columns to mean 0 and normalize columns to standard deviation 1
5. Normalize rows to length 1
6. Normalize rows to standard deviation 1
7. Centralize rows to mean 0
8. Centralize rows to mean 0 and normalize rows to standard deviation 1

Multiple transformations can be performed in sequence: each transformation changes the data table, and a subsequent transformation will be applied to the modified data table. Option 4 is an example of a sequence of Option 3 followed by Option 2, but is offered as a separate option because this combination is frequently used. It is the user's responsibility to ensure that any chosen sequence of transformations is viable from a statistical analysis standpoint.

The macro will report any normalization transformation for which the divisor evaluates to zero, as this would result in a division by zero error. However this occurrence is not treated as a catastrophic failure, as the transformation is declined and the data table is left unaltered.

6.2.9 Transposing the SIMFIT table

This facility is provided via the dialog box

Option to create Simfit File from the transposed Data Table.

Note that this option operates only on the exported file and does not alter the table which is contained within the worksheet. The possibilities are as follows.

1. Create Simfit file from the Data Table

2. Create Simfit file from the transposed Data Table

After this choice has been made the macro proceeds to create the export file with the SIMFIT-defined structure. At the end of this process, the macro will terminate after reporting successful creation of the SIMFIT file.

6.2.10 Inspecting and saving the modified worksheet

Because *simfit6.xls* makes successive changes to the actual worksheet, it can be quite useful to be able to inspect and/or save the modified worksheet. This can only be done when the macro has terminated, either on normal completion or by taking a termination option at an intermediate stage of processing, such as after filling empty cells, or after any transformation in a sequence, or before creating the final SIMFIT file.

If the user wishes to save the modified workbook it is advisable to do this using the Save As method with a different filename, otherwise the original file will be overwritten by the modified version. In order to resume processing, the user must either:

- Select the table in the modified worksheet and re-run the macro; or
- Re-load the original worksheet and re-run the macro, repeating the previous processing steps.

6.2.11 The History Log

Every time the *simfit6* macro is run, a new worksheet is inserted into the user's workbook to hold the History Log. Any History Log worksheet which is already present in the workbook (created by a previous macro run) is deleted. [Tip: if you wish to retain History Logs from previous macro runs, rename their worksheets (History Log 1, History Log 2, etc) before running the macro.]

The History Log records:

- The sequence of processes which have been applied by the macro. This is important as an audit trail of how the final data table was derived from the original version, especially if the process sequence has to be repeated at some future time.
- The processing time spent on the main iterative processes (to assist in estimating overall processing times for very large data tables).
- Any errors which were detected during processing.

In order to ensure that the History Log travels with the SIMFIT file, its contents are appended to the SIMFIT file as additional lines between the delimiters `begin{transformation history}` and `end{transformation history}`.

6.2.12 Processing very large data tables

The processing steps for very large data tables can run into several minutes. During these periods, the mouse pointer will be displayed as an hourglass icon, indicating that the user must wait for the current process to finish.

Timing tests on a 60000 rows by 50 columns matrix indicated that the longest processes were the copying of the active worksheet (t_1), the actual transformations (t_2), and the creation of the SIMFIT file (t_3), where the times were in the ratio

$$t_1 : t_2 : t_3 = 69 : 62 : 14.$$

The length of time for the copying of the active worksheet is the reason why this process has been made optional, but the user who chooses to omit this step must take alternative precautions to safeguard the original data.

7 Importing SIMFIT results tables into documents and spreadsheets

7.1 SIMFIT results files

Each time a data set is analyzed the results are written to a file called `f$result.txt`, and the current files are renamed so that the existing `f$result.txt` becomes `f$result.001` while `f$result.001` becomes `f$result.002` and so on. These can be viewed using the [Results] option from the main SIMFIT menu.

These SIMFIT results files are formatted so that the numbers displayed only contain the number of significant figures that are meaningful in context. For instance, probabilities will usually only have four digits after the decimal point, which indicates that it does not make any sense to consider any subsequent digits for purpose of statistical testing, and in any case probability estimates will not be accurate for more than about four digits. Furthermore, as experimental data are rarely more accurate than about three or four significant figures anyway, it may be wishful thinking to ever consider more than say six. From SIMFIT Version 8 users can choose the number of significant figures but the tables are formatted using a fixed font to line up column entities irrespective of absolute size. Many users do not want this in a thesis or published document. Naturally, these arguments do not apply to integers.

7.2 Preliminary analysis

As an example consider the following case with the title Table 1.

Table 1					
1-Way Analysis of Variance: 1 (Grand Mean 4.316E+01)					
Transformation:- x (untransformed data)					
Source	SSQ	NDOF	MSQ	F	p
Between Groups	2.19344E+03	4	5.48316E+02	5.61546E+01	0.0000
Residual	2.44130E+02	25	9.76520E+00
Total	2.43757E+03	29

Actually, most users would want to import such a table formatted as tabbed-text, html, xml, or L^AT_EX into documents such as a report, thesis, or publication looking something like Table 2.

Table 2					
1-Way Analysis of Variance: 1 (Grand Mean 43.16)					
Transformation:- x (untransformed data)					
Source	SSQ	NDOF	MSQ	F	p
Between Groups	2193.44	4	548.316	56.1546	0.0000
Residual	244.130	25	9.76520		
Total	2437.57	29			

Or even, for those with artistic leanings, possibly something like Table 3.

Table 3					
1-Way Analysis of Variance: Grand Mean 43.16					
Transformation: <i>x</i> (untransformed data)					
Source	SSQ	NDOF	MSQ	F	p
Between Groups	2193.44	4	548.316	56.1546	0.0000
Residual	244.130	25	9.76520		
Total	2437.57	29			

7.3 The procedure

This article explains the procedures required to export tables from from SIMFIT results files into forms suitable for inclusion into word processors, spreadsheet programs, website scripts, or even professional document preparation systems such as L^AT_EX. Also, decimal points can be replaced by commas as in continental notation if required.

It is important to realize that the [Results] option from the SIMFIT main menu gives access to all the currently saved results files.

- **Choosing a results file.**

Early versions of SIMFIT allowed users to name results files individually to avoid anything being lost. However, now that up to 100 results files are saved and users have the option [Results] from the main SIMFIT menus from which to view, print, save, edit, or export tables, this is no longer usually necessary. Clearly, if results are always required for retrospective use, regular back-up or saving will be necessary.

- **Extracting a table.**

From the [Extract tables] option view the file to make sure it is the one required then copy to the clipboard only the table required along with any associated header and trailer sections ... but nothing else.

- **Preliminary editing.**

Sometimes editing of the file is required to make sure that every row of the table has exactly the same number of columns. So note that, for extracting a table there can be no empty cells, and each cell must contain precisely one word. Any column titles must be edited so that they consist of one word, for instance changing Time of Day to Time_of_Day, or filling empty cells by three dots. Added underscores and sets of three contiguous dots can be removed when the final table is written to file. A pre-processing option is provided for editing before attempting to create a table.

Note that often tables have cells with added comments relating to goodness of fit or results of statistical testing, and these do not need to be underscored. There are also special tables with only two columns containing several words in some cells, and Example 1 later will make this clear.

- **Viewing the hashtag table.**

The algorithm attempts to identify cells in a table by inserting a hashtag between every column. If the algorithm succeeds there would be no need to view this hashtag table. However this option should be switched on until the process of the algorithm is understood, or if it fails and you need to see why.

The hashtag table is clarified in Example 1.

- **Headers and trailers.**

Frequently tables have header and trailer sections that are descriptive and not part of the table itself. As these can have strings of word and numbers that would confuse the algorithm checking that every row must have the same number of columns, they must be identified. This is done by using buttons on a window that allows the header and trailer lines to be highlighted. If this is not done the table creating algorithm will fail.

Selecting headers and trailers is clarified in Example 2.

- **Saving the table.**

For programs that produce Windows quality hardcopy the table should be saved as html or xml as these will import directly into word processors or spreadsheets. Tabbed-text is also available but is much less versatile than html or xml, and L^AT_EX is available for those up to it.

- **Fine tuning.**

Inevitably there will often be the need for dealing with details. For instance, users will sometimes want to replace alpha by α or chi-sqd by χ^2 and this can be done for html, xml, and L^AT_EX output but not

for tabbed-text. On the other hand html, xml, and L^AT_EX have reserved letters and these must be dealt with retrospectively.

For instance, consider the transformation of the following expressions which can be done in html, xml, and L^AT_EX but not in tabbed text.

Original	Transformed
>=	≥
=<	≤
alpha	α
beta	β
delta	δ
gamma	γ
lambda	λ
infinity	∞
P(chi-sqd >= TS) = 0.2037	$P(\chi^2 \geq TS) = 0.2037$

Where ambiguity could arise in L^AT_EX using underscores or similar special characters they will be replaced by question marks. So L^AT_EX users should search for ? characters to replace for the final table.

- **Padding with zeros.**

As the [Extract tables] option will never remove significant figures a problem arises if users wish to replace numbers in scientific notation by floating point representation. In order to extend the range over which this can be done, padding zeros can be introduced as illustrated in this next table.

Scientific notation	Padding	Floating point representation
1.234E-01	0	.1234
1.234E+00	0	1.234
1.234E+01	0	12.34
1.234E+02	0	123.4
1.234E-02	2	0.01234
1.234E-01	2	0.12340
1.234E+00	2	1.23400
1.234E+01	2	12.3400
1.234E+02	2	123.400
1.234E+03	2	1234.00
1.234E+04	2	12340.0
1.234E-04	4	0.0001234
...
1.234E+06	4	1234000.0

Evidently increasing the number of padding zeros increases the range over which transformation from scientific to floating point representation can be achieved, and the default is two which allows a wide range, but four ensures that a mixture of transformed and untransformed numbers will occupy the same width in the columns of a table.

However, there is here a problem because adding padding zeros could suggest all trailing zeros are meaningful. For instance, the number 1.2341213179 stored in the computer could be written as 1.234E+00 in SIMFIT output because the analysis in question only justifies accuracy or meaning for up to four significant figures. However, 1.23400 could be mistaken for indicating the internal representation with twelve significant figures. So users may wish to suppress trailing zeros in such cases, noting that this could result in numbers with different widths in a column. In any case, the transformation from scientific notation to floating point can be switched off.

Sometimes, for instance with numerical analysis where more significant figures are justified than with data analysis, eight significant figures are output, and some procedures can optionally allow more. In addition special DLLs dedicated for particular routines can be supplied for this purpose.

7.4 Example 1

This example shows the transformation of a special type of SIMFIT table containing just two columns separated by equals signs (i.e., =) and containing cells with multiple words. As long as the equals signs are perfectly lined up and there is no header or trailer section, this type of table can always be transformed. Here is the table with no header or trailer sections as extracted using the [Results] then [Extract tables] options from the main SIMFIT menu using the results file f\$result.txt following the exhaustive analysis of a vector process used to analyze data contained in the default test file normal.tf1.

Sample size	=	50
Minimum value	=	-2.20820E+00
Maximum value	=	1.61750E+00
Coefficient of skew	=	-1.66905E-02
Coefficient of kurtosis	=	-7.68395E-01
Lower Hinge (25th percentile)	=	-8.55015E-01
Median value (50th percentile)	=	-9.73615E-02
Upper Hinge (75th percentile)	=	7.85965E-01
Sample mean	=	-2.57897E-02
Sample standard deviation	=	1.00553E+00
Coefficient of variation (CV%)	=	> 100%
Standard error of the mean	=	1.42203E-01
Upper 2.5% t-value	=	2.00958E+00
Lower 95% con lim for mean	=	-3.11558E-01
Upper 95% con lim for mean	=	2.59978E-01
Sample variance	=	1.01109E+00
Lower 95% con lim for variance	=	7.05519E-01
Upper 95% con lim for variance	=	1.57006E+00
Shapiro-Wilks W statistic	=	9.62693E-01
Significance level for W	=	0.1153
Conclusion	=	Tentatively accept normality

This is the corresponding intermediate hashtag table.

Sample size	#	50
Minimum value	#	-2.20820E+00
Maximum value	#	1.61750E+00
Coefficient of skew	#	-1.66905E-02
Coefficient of kurtosis	#	-7.68395E-01
Lower Hinge (25th percentile)	#	-8.55015E-01
Median value (50th percentile)	#	-9.73615E-02
Upper Hinge (75th percentile)	#	7.85965E-01
Sample mean	#	-2.57897E-02
Sample standard deviation	#	1.00553E+00
Coefficient of variation (CV%)	#	> 100%
Standard error of the mean	#	1.42203E-01
Upper 2.5% t-value	#	2.00958E+00
Lower 95% con lim for mean	#	-3.11558E-01
Upper 95% con lim for mean	#	2.59978E-01
Sample variance	#	1.01109E+00
Lower 95% con lim for variance	#	7.05519E-01
Upper 95% con lim for variance	#	1.57006E+00
Shapiro-Wilks W statistic	#	9.62693E-01
Significance level for W	#	0.1153
Conclusion	#	Tentatively accept normality

The hashtag table is very useful for detecting the source of errors. The table making algorithm attempts to locate the position separating columns and writes a hashtag there. If every row has the same number of columns then every row will have the same number of hashtags and the algorithm has succeeded. Observing this hashtag table when the algorithm has failed will allow you identify then correct the error.

Here is the selected table as it would be written to the output file.

Sample size	50
Minimum value	-2.2082000
Maximum value	1.6175000
Coefficient of skew	-0.0166905
Coefficient of kurtosis	-0.7683950
Lower Hinge (25th percentile)	-0.8550150
Median value (50th percentile)	-0.0973615
Upper Hinge (75th percentile)	0.7859650
Sample mean	-0.0257897
Sample standard deviation	1.0055300
Coefficient of variation (CV%)	> 100%
Standard error of the mean	0.1422030
Upper 2.5% t-value	2.0095800
Lower 95% con lim for mean	-0.3115580
Upper 95% con lim for mean	0.2599780
Sample variance	1.0110900
Lower 95% con lim for variance	0.7055190
Upper 95% con lim for variance	1.5700600
Shapiro-Wilks W statistic	0.9626930
Significance level for W	0.1153
Conclusion	Tentatively accept normality

Here it is with a few minor cosmetic changes.

Exhaustive analysis of a vector	
Sample size	50
Minimum value	-2.2082000
Maximum value	1.6175000
Coefficient of skew	-0.0166905
Coefficient of kurtosis	-0.7683950
Lower Hinge (25th percentile)	-0.8550150
Median value (50th percentile)	-0.0973615
Upper Hinge (75th percentile)	0.7859650
Sample mean	-0.0257897
Sample standard deviation	1.0055300
Coefficient of variation (CV%)	> 100%
Standard error of the mean	0.1422030
Upper 2.5% t-value	2.0095800
Lower 95% confidence limit for mean	-0.3115580
Upper 95% confidence limit for mean	0.2599780
Sample variance	1.0110900
Lower 95% confidence limit for variance	0.7055190
Upper 95% confidence limit for variance	1.5700600
Shapiro-Wilks W statistic	0.9626930
Significance level for W	0.1153
Conclusion:	<i>Tentatively accept normality</i>

7.5 Example 2

From fitting a two-exponential model to data in the test file `exfit.tf4` using program `exfit` the following results can be extracted from the results file.

For best-fit 2-exponential function					
Parameter	Value	Std.Error	Lower95%cl	Upper95%cl	p
A(1)	8.5255E-01	6.7731E-02	7.1332E-01	9.9177E-01	0.0000
A(2)	1.1765E+00	7.4779E-02	1.0227E+00	1.3302E+00	0.0000
k(1)	6.7935E+00	8.5386E-01	5.0383E+00	8.5486E+00	0.0000
k(2)	1.1121E+00	5.1128E-02	1.0070E+00	1.2172E+00	0.0000
AUC	1.1834E+00	1.4714E-02	1.1531E+00	1.2136E+00	0.0000
AUC is the area under the curve from t = 0 to t = infinity					
Initial time point (A) = 3.5983E-02					
Final time point (B) = 1.6110E+00					
Area over range (A,B) = 9.3832E-01					
Average over range (A,B) = 5.9575E-01					

Now the file has an additional head and trailer section so, if the full table is selected, it will have to be highlighted as follows in the header and trailer selection control as shown next, where the header is colored magenta and the trailer colored cyan.

For best-fit 2-exponential function					
Parameter	Value	Std.Error	Lower95%cl	Upper95%cl	p
A(1)	8.5255E-01	6.7731E-02	7.1332E-01	9.9177E-01	0.0000
A(2)	1.1765E+00	7.4779E-02	1.0227E+00	1.3302E+00	0.0000
k(1)	6.7935E+00	8.5386E-01	5.0383E+00	8.5486E+00	0.0000
k(2)	1.1121E+00	5.1128E-02	1.0070E+00	1.2172E+00	0.0000
AUC	1.1834E+00	1.4714E-02	1.1531E+00	1.2136E+00	0.0000
AUC is the area under the curve from t = 0 to t = infinity					
Initial time point (A) = 3.5983E-02					
Final time point (B) = 1.6110E+00					
Area over range (A,B) = 9.3832E-01					
Average over range (A,B) = 5.9575E-01					

Using two padding zeros this leads to the following table.

For best-fit 2-exponential function					
Parameter	Value	Std.Error	Lower95%cl	Upper95%cl	p
A(1)	0.852550	0.067731	0.713320	0.991770	0.0000
A(2)	1.176500	0.074779	1.022700	1.330200	0.0000
k(1)	6.793500	0.853860	5.038300	8.548600	0.0000
k(2)	1.112100	0.051128	1.007000	1.217200	0.0000
AUC	1.183400	0.014714	1.153100	1.213600	0.0000
AUC is the area under the curve from t = 0 to t = ∞					
Initial time point (A) = 0.035983					
Final time point (B) = 1.611					
Area over range (A,B) = 0.93832					
Average over range (A,B) = 0.59575					

However, note that, with this example, three points emerge.

1. Numbers outside the main table will also be transformed into floating point numbers.
2. Equals signs lined up the trailer will not lead directly to secondary tabulation.
3. Some special words, like infinity, will be recognized.

So, because a certain amount of fine tuning will be required, the possibilities for handcrafting are endless. Here, for example, the header is enlarged by adding a formula, while the trailer is added in the form of a footnote to the main table.

For the best-fit 2-exponential function					
$f(t) = A_1 \exp(-k_1 t) + A_2 \exp(-k_2 t)$					
Parameter	Value	Std.Error	Lower95%cl	Upper95%cl	p
A_1	0.852550	0.067731	0.713320	0.991770	0.0000
A_2	1.176500	0.074779	1.022700	1.330200	0.0000
k_1	6.793500	0.853860	5.038300	8.548600	0.0000
k_2	1.112100	0.051128	1.007000	1.217200	0.0000
AUC	1.183400	0.014714	1.153100	1.213600	0.0000

$$\text{Area under the curve } AUC = \int_0^{\infty} \hat{f}(t) dt$$

$$\text{Initial time point (A)} = 0.035983$$

$$\text{Final time point (B)} = 1.611$$

$$\text{Area over range (A,B)} = 0.93832$$

$$\text{Average over range (A,B)} = 0.59575$$

7.6 Example 3

A special situation exists with symmetric matrices where just a lower or upper triangle is displayed, and also some other related situations. For instance, following on from the previous example we have the parameter correlation matrix expressed in the following form.

Parameter correlation matrix			
1.0000			
-0.8758	1.0000		
-0.5964	0.8996	1.0000	
-0.8480	0.9485	0.8200	1.0000

Clearly, the algorithm to count the number of columns per row in order to insert hashtags will fail because all the rows have different numbers of columns, unless editing is performed like this.

Parameter correlation matrix			
1.0000
-0.8758	1.0000
-0.5964	0.8996	1.0000	...
-0.8480	0.9485	0.8200	1.0000

Now transformation would be possible leading to a table such as the following.

Parameter correlation matrix			
1			
-0.8758	1		
-0.5964	0.8996	1	
-0.8480	0.9485	0.8200	1

Another example to consider is from correlation analysis which leads to r values in the strict upper triangle and significance levels in the strict lower triangle as shown next followed by the extracted table.

Pearson correlation results							
Upper triangle = r , Lower = corresponding two-tail p values							
.....	0.5295	0.2874	0.0662	0.1941	0.6255	-0.5876	0.3010
0.0766	0.3285	-0.0219	0.7930	0.5338	-0.4230	0.3006
0.3650	0.2971	-0.2833	0.2165	0.0264	0.2314	-0.0304
0.8381	1.0000	1.0000	0.2787	0.2837	-0.5238	-0.1166
0.5455	0.0021	0.4992	0.3804	0.2029	-0.1949	0.2144
0.0296	0.0738	0.9351	1.0000	0.5271	-0.4532	0.1360
1.0000	1.0000	0.4694	1.0000	1.0000	1.0000	-0.1696
0.3418	0.3424	1.0000	1.0000	0.5035	0.6735	1.0000
Test for absence of any significant correlations							
H0: correlation matrix is the identity matrix							
Determinant = 2.476E-03							
Test statistic (TS) = 4.501E+01							
Degrees of freedom = 28							
P(chi-sq >= TS) = 0.0220 Reject H0 at 5% sig.level							

Pearson correlation results

Upper triangle = r , Lower = corresponding two-tail p values

.....	0.5295	0.2874	0.0662	0.1941	0.6255	-0.5876	0.3010
0.0766	0.3285	-0.0219	0.7930	0.5338	-0.4230	0.3006
0.3650	0.2971	-0.2833	0.2165	0.0264	0.2314	-0.0304
0.8381	1.0000	1.0000	0.2787	-0.2837	-0.5238	-0.1166
0.5455	0.0021	0.4992	0.3804	0.2029	-0.1949	0.2144
0.0296	0.0738	0.9351	1.0000	0.5271	-0.4532	0.1360
1.0000	1.0000	0.4694	1.0000	1.0000	1.0000	-0.1696
0.3418	0.3424	1.0000	1.0000	0.5035	0.6735	1.0000

Test : for absence of any significant correlations

H_0 : correlation matrix is the identity matrix

Determinant = 0.002476

Test statistic(TS) = 45.01

Degrees of freedom = 28

$P(\chi^2 \geq TS) = 0.0220$ Reject H_0 at 5% significance level

Here the five dots (.....) denote that the diagonal elements have no meaning and this is just a convenient way to conserve space by having one matrix instead of two. Note that the option to blank out three dots (...) used as temporary column separators does not blank out groups with less three or more than three contiguous dots.

A summary of the options available and procedure to be used comes next.

7.7 Summary

- The first step is to select just one table from the results file.
- This can be followed by an optional pre-processing step to edit the table so that every row has exactly the same number of columns.
- Empty cells must be denoted by a three dot ellipsis (...) and cells containing multiple words must have them joined by underscores or similar.
- Three dot symbols to denote empty cells are deleted from the output file.
- The input table can have optional header and trailer sections if required, but these must be highlighted by the control to select headers and trailers.
- There is an option to transform scientific numbers into floating point format by specifying the number of padding zeros required. This option can be switched off.
- Numbers in the header and trailer will also be transformed in this way.
- If it is required, decimal points in floating point numbers can be replaced by commas.
- If transformation fails then the option for pre-processing should be switched on, and also the hashtag table should be requested. By viewing the hashtag table most errors can easily be diagnosed, then rectified by a re-run using pre-process editing.
- If tabbed-text output is selected the resulting file will have to be input into a spreadsheet program for formatting before importing into a word processing program.
- Both html and xml output can allow a certain number of further changes, like changing alpha into α , or adding cell borders.
- \LaTeX output will have question marks (?) inserted to replace forbidden character such as underscores which must be edited retrospectively depending the intention, e.g., linking words, or denoting subscripts.

Three further things should be emphasized.

1. Some tables have specialized features such as lined up equals signs that allow multiple words in a column and, as long as every row in the table has an equals sign in exactly the same position, this feature will be recognized.
2. Some *SIMFIT* results files output tables to the display without three dot separators (...) to create a more pleasing effect, but add them to the results files to assist the processing described in this document.
3. There are several widely used tables that can have empty cells and multi-word titles that the parsing routine will recognize and format automatically.

Finally, should you require further worked examples, you can browse the *SIMFIT* tutorials, or the document `w_examples.pdf`, where a large number of alternative display styles are demonstrated.

8 Printing and importing *SiMFiT* graphs into documents

SiMFiT will create high quality graphics files without any other software packages. However, GhostScript DLLs are installed with *SiMFiT* and not only does this greatly increase the quality of output available, but also many other advanced graphical options are then available. This is because the primary high level graphics format created by *SiMFiT* is the Encapsulated PostScript vector format, i.e., .eps files but, as not all applications can use these directly, you may have to create graphic image files in other formats. However, if you cannot benefit directly from the superior quality of such .eps files, you should never import *SiMFiT* graphics files into documents in .bmp, .pcx, .jpg, or .tif formats; only use .emf or svg exported from *SiMFiT* using the [Win] button, or better .png, or pdf exported using the [PS] button then generated from .eps using the GhostScript DLLs should be used.

8.1 Graphics hardcopy

Any graph displayed by *SiMFiT* can be used directly to drive printers using a high resolution bitmap. The resolution and size can be adjusted but this is only useful for a quick record.

If you have a PostScript printer, then hardcopy driven directly from the display will be of very high quality. If you do not have a PostScript printer, then install Ghostscript and Gsview and similar print quality can be obtained on any printer.

8.1.1 Bitmaps and compressed bitmaps

You should never choose to save .bmp, .jpg, .tif, .pcx from the display unless you have a good reason for being satisfied with such large, poor resolution files.

8.1.2 Enhanced metafiles (.emf)

Probably the easiest way to use *SiMFiT* graphs in Windows is to save Enhanced Metafiles, i.e., .emf files directly from *SiMFiT*. The quality of .emf files can be improved somewhat by configuring *SiMFiT* to use slightly thicker lines and bolder fonts than the default settings, which you can investigate. If you do not have a PostScript printer, and do not have *SiMFiT* configured to use Gsview, then this is the only course of action open to you, and you will miss out on a great many sophisticated *SiMFiT* plotting techniques.

A further undesirable feature of using .emf files is that they can too easily have their aspect ratio changed within Windows programs, leading to ugly fonts.

8.1.3 Scalable vector graphics (.svg)

The W3C recommended format for web line graphics is now the XML based SVG format, and many Windows applications can import these. As *SiMFiT* graphs are line drawings and do not need to save pixel information for the background or filling symbols, then .svg files are an alternative to .png, and have many of the advantages of .eps files.

8.2 PostScript graphics files (.eps)

The advantage of storing *SiMFiT* encapsulated PostScript graphics files is that they are very compact, can be printed at any resolution with any number of colors, and *SiMFiT* has many facilities for editing, re-sizing, rotating, overlaying, or making collages from such .eps files. A unique feature is that *SiMFiT* .eps files have a structured format, so they can easily be edited in a text editor, e.g., Notepad, to change sizes, fonts, line types symbols, colors, labels, etc., retrospectively.

8.2.1 Ghostscript generated files

As Ghostscript DLLs are now installed with *SIMF_T* you can supply .eps files for transformation into other graphics formats such as .bmp, .jpg, .tif, or especially .pdf, or .png.

8.2.2 Portable Document graphics files (.pdf)

If you use Adobe Acrobat, or can import Portable Document Format files, i.e., .pdf files generated from *SIMF_T* .eps files by Ghostscript, into your Windows application, then such true .pdf files are an excellent choice. However, beware of the fact that many applications simply embed bitmaps into PostScript or .pdf files, whereupon all the advantages are lost.

8.2.3 Portable network graphics files (.png)

Increasingly, the most versatile format for importing graphic image files into Windows programs, such as Microsoft Word and PowerPoint, or LibreOffice Writer or Impress, is the Portable Network Graphics format, as the compression used in .png files results in smaller files than .bmp or .jpg, and edge resolution is far superior. So the best way for users without PostScript printers to use *SIMF_T* graphs in Windows is to store graphs in .eps format, then create .png files at 72dpi for small applications, like the web, where resolution is not important, but at 300dpi or 600dpi if you wish the graph to be printed or displayed at high resolution.

The industry standard for scientific graphs is .png, or .svg as these are free from patent problems and are increasingly being accepted by all applications and all operating systems.

8.3 Using Encapsulated PostScript (.eps) files directly

The main advantage of working with *SIMF_T* .eps files is that they are small, compact, and device-independent and can be printed or displayed at any resolution and size with no pixellation or other losses of quality. In particular they can be used directly to create high-quality scientific documents like the *SIMF_T* manuals and tutorials using \LaTeX .

If you have a PostScript printer, you may be able to import *SIMF_T* .eps files into Word, but Word may add a low resolution preview, so that what you see in the display may not be what you get on printing. A Word document containing .eps files will print *SIMF_T* graphs at high resolution only on PostScript printers, on non-PostScript printers the resolution may be poor and the graph may not be printed correctly.

For this reason, the best way is to save .eps files then create .pdf, .png or .svg files, either when the .eps file is created or retrospectively. From version 8 this can be done at any time as the Ghostscript DLLs are loaded with the *SIMF_T* package and GSview is no longer required.

Even if *SIMF_T* does not need GSview this extremely valuable free program and the Ghostscript installation it requires can be downloaded from the *SIMF_T* website. An advantage of having access to the GSview package is that your archived .eps files can then be printed as professional quality stand alone graphs at high resolution on any printer.