

The SciDAVis Handbook

Ion Vasilief, Roger Gadiou, and Knut Franke

February 28, 2008

Contents

| | | |
|----------|---|-----------|
| 1 | Introduction | 1 |
| 1.1 | What is SciDAVis? | 1 |
| 1.2 | Command Line Parameters | 2 |
| 1.2.1 | Specify a File | 2 |
| 1.2.2 | Command Line Options | 3 |
| 1.3 | General Concepts and Terms | 3 |
| 1.3.1 | Tables | 6 |
| 1.3.2 | Matrix | 7 |
| 1.3.3 | Plot Window | 7 |
| 1.3.4 | Note | 8 |
| 1.3.5 | Log Window | 9 |
| 1.3.6 | The Project Explorer | 9 |
| 2 | Drawing plots with SciDAVis | 11 |
| 2.1 | 2D plots | 11 |
| 2.1.1 | 2D plot from data. | 11 |
| 2.1.2 | 2D plot from function. | 14 |
| 2.1.2.1 | Direct plot of a function. | 15 |
| 2.1.2.2 | Filling of a table with the values of a function. | 15 |
| 2.2 | 3D plots | 16 |
| 2.2.1 | Direct 3D plot from a function | 17 |
| 2.2.2 | 3D plot from a matrix | 19 |
| 2.3 | Multilayer Plots | 20 |
| 2.3.1 | Building a multilayer plot panel | 20 |
| 2.3.2 | Building a multilayer plot step by step | 21 |
| 3 | Command Reference | 24 |
| 3.1 | The File Menu | 24 |
| 3.2 | The Edit Menu | 30 |
| 3.3 | The View Menu | 30 |
| 3.4 | The Graph Menu | 31 |
| 3.5 | The Plot Menu | 32 |
| 3.6 | The Plot 3D menu | 39 |
| 3.7 | The Data Menu | 42 |

| | | |
|----------|--|-----------|
| 3.8 | The Analysis Menu | 43 |
| 3.8.1 | Commands for the analysis of data in tables | 43 |
| 3.8.2 | Commands for the analysis of curves in plots | 45 |
| 3.9 | The Table Menu | 49 |
| 3.10 | The Matrix Menu | 51 |
| 3.11 | The Format Menu | 51 |
| 3.12 | The Window Menu | 52 |
| 3.13 | Customization of 3D plots | 53 |
| 4 | The Toolbars | 55 |
| 4.1 | The Edit Toolbar | 55 |
| 4.2 | The File Toolbar | 55 |
| 4.3 | The Plot Toolbar. | 56 |
| 4.4 | The Table Toolbar. | 56 |
| 4.5 | The Plot 3D Toolbar. | 60 |
| 5 | The Dialogs | 62 |
| 5.1 | Add Error bars | 62 |
| 5.2 | Add Function | 63 |
| 5.3 | Add Layer | 65 |
| 5.4 | Add/Remove curves. | 65 |
| 5.5 | Add Text | 66 |
| 5.6 | Arrange Layers | 67 |
| 5.7 | Add Arrow | 68 |
| 5.8 | Column Options | 69 |
| 5.9 | Contour Curves Options | 70 |
| 5.10 | Custom Curves | 72 |
| 5.10.1 | Custom curves for lines and scatter plots | 73 |
| 5.10.2 | Custom curves for pie plots | 73 |
| 5.10.3 | Custom curves for box plots | 74 |
| 5.10.4 | Custom curves for pie histogram | 75 |
| 5.11 | Define surface plot | 77 |
| 5.12 | Export ASCII | 77 |
| 5.13 | Fast Fourier Transform | 78 |
| 5.14 | Integrate dialog | 79 |
| 5.15 | Non linear curve fit | 79 |
| 5.16 | General Plot Options | 81 |
| 5.17 | Plot Wizard | 84 |
| 5.18 | Project Explorer | 85 |
| 5.19 | Preferences Dialog | 86 |
| 5.20 | Printer-setup | 91 |
| 5.21 | Set Column Values | 92 |
| 5.22 | Set Dimensions... | 92 |
| 5.23 | ASCII Import options | 93 |
| 5.24 | Set Properties... | 94 |
| 5.25 | Set Values... | 94 |

| | | |
|----------|--|------------|
| 5.26 | Surface plot options | 95 |
| 5.27 | Text options | 97 |
| 6 | Analysis of data and curves | 99 |
| 6.1 | Fast Fourier Transform | 99 |
| 6.2 | Correlation | 100 |
| 6.3 | Convolution | 101 |
| 6.4 | Deconvolution | 101 |
| 6.5 | Non Linear Curve Fit | 101 |
| 6.6 | Fitting to specific curves | 102 |
| 6.6.1 | Fitting to a line | 102 |
| 6.6.2 | Fitting to a polynome | 103 |
| 6.6.3 | Fitting to a Boltzmann function | 104 |
| 6.6.4 | Fitting to a Gauss function | 105 |
| 6.6.5 | Fitting to a Lorentz function | 106 |
| 6.7 | Multi-Peaks fitting | 106 |
| 6.8 | Filtering of data curves | 107 |
| 6.8.1 | FFT low pass filter | 108 |
| 6.8.2 | FFT high pass filter | 109 |
| 6.8.3 | FFT band pass filter | 110 |
| 6.8.4 | FFT block band filter | 111 |
| 6.9 | Interpolation | 112 |
| 7 | Mathematical Expressions and Scripting | 114 |
| 7.1 | muParser | 114 |
| 7.2 | Python | 116 |
| 7.2.1 | Getting Started | 116 |
| 7.2.2 | Python Basics | 117 |
| 7.2.2.1 | Expressions | 117 |
| 7.2.2.2 | Statements | 117 |
| 7.2.3 | Evaluation Reloaded | 119 |
| 7.2.4 | Mathematical Functions | 119 |
| 7.2.5 | Accessing SciDAVis's functions from Python | 121 |
| 7.2.5.1 | Establishing contact | 121 |
| 7.2.5.2 | Working with Tables | 122 |
| 7.2.5.3 | Working with Matrices | 123 |
| 7.2.5.4 | Plotting and Working with Graphs | 123 |
| 7.2.5.5 | Fitting | 125 |
| 7.2.6 | The Initialization File | 127 |
| A | Appendix | 128 |
| A.1 | Credits and License | 128 |
| A.1.1 | GNU Free Documentation License | 128 |
| A.1.1.1 | Preamble | 128 |
| A.1.1.2 | Applicability And Definitions | 129 |
| A.1.1.3 | Verbatim Copying | 130 |

| | | |
|----------|---|------------|
| A.1.1.4 | Copying In Quantity | 130 |
| A.1.1.5 | Modifications | 131 |
| A.1.1.6 | Combining Documents | 132 |
| A.1.1.7 | Collections Of Documents | 132 |
| A.1.1.8 | Aggregation With Independent Works | 133 |
| A.1.1.9 | Translation | 133 |
| A.1.1.10 | Termination | 133 |
| A.1.1.11 | Future Revisions Of This License | 133 |
| A.2 | How to obtain SciDAVis | 134 |
| A.3 | Requirements | 134 |
| A.4 | Installation from binary packages | 135 |
| A.5 | Compilation and Installation from sources | 135 |
| B | Frequently asked questions | 136 |

List of Figures

| | | |
|------|--|----|
| 1.1 | A typical SciDAVis session | 4 |
| 1.2 | The SciDAVis table | 6 |
| 1.3 | The SciDAVis matrix | 7 |
| 1.4 | An example of SciDAVis 2D graph | 8 |
| 1.5 | The SciDAVis Note Window | 9 |
| 1.6 | The SciDAVis Log window | 9 |
| 1.7 | The SciDAVis Project Explorer | 10 |
| 2.1 | A simple 2D plot: the table. | 12 |
| 2.2 | A simple 2D plot: the default plot. | 13 |
| 2.3 | A simple 2D plot: the plot finished. | 13 |
| 2.4 | A 2D plot with two Y axis. | 14 |
| 2.5 | Direct plot of a function. | 15 |
| 2.6 | Function plot: filling of the X column. | 16 |
| 2.7 | Function plot: filling of the Y column. | 16 |
| 2.8 | Example of a 3D Plots. | 17 |
| 2.9 | Definition of a new surface 3D plot | 18 |
| 2.10 | The 3D surface plot created by default | 18 |
| 2.11 | The 3D surface plot after customizations | 19 |
| 3.1 | The Smooth -> Moving Window Average... dialog. | 46 |
| 3.2 | The Smooth -> Moving Window Average... dialog. | 46 |
| 3.3 | The FFT Filter -> Low Pass... dialog. | 46 |
| 3.4 | The FFT Filter -> High Pass... dialog. | 47 |
| 3.5 | The FFT Filter -> Band Pass... dialog. | 47 |
| 3.6 | The FFT Filter -> Band Block... dialog. | 48 |
| 3.7 | The Interpolate... dialog. | 48 |
| 4.1 | The SciDAVis Edit Toolbar | 55 |
| 4.2 | The SciDAVis File Toolbar | 55 |
| 4.3 | The SciDAVis Plot Toolbar | 56 |
| 4.4 | The SciDAVis Table Toolbar | 56 |
| 4.5 | The SciDAVis Plot 3D Toolbar | 60 |
| 5.1 | The Add Error Bars... dialog. | 62 |

| | | |
|------|---|----|
| 5.2 | A plot with X and Y Error Bars. | 63 |
| 5.3 | The Add Function... dialog box: cartesian coordinates. | 64 |
| 5.4 | The Add Function... dialog box: parametric coordinates. | 64 |
| 5.5 | The Add Function... dialog box: polar coordinates. | 65 |
| 5.6 | The Add Layer dialog box. | 65 |
| 5.7 | The Add/Remove Curves... dialog box. | 66 |
| 5.8 | The Add Text dialog box. | 67 |
| 5.9 | The Arrange Layers dialog | 67 |
| 5.10 | The <i>Arrow options</i> dialog: first tab | 68 |
| 5.11 | The <i>Arrow options</i> dialog: second tab | 68 |
| 5.12 | The <i>Geometry</i> dialog: third tab | 69 |
| 5.13 | The Column Options... dialog. | 69 |
| 5.14 | The Contour Options dialog. | 70 |
| 5.15 | The Custom Curves Dialog: Line formatting. | 72 |
| 5.16 | The Custom Curves Dialog: Plot Associations. | 72 |
| 5.17 | The Custom Curves Dialog: Line formatting. | 73 |
| 5.18 | The Custom Curves Dialog: Symbol formatting. | 73 |
| 5.19 | The Custom Curves Dialog for pies: pie segment formatting. | 74 |
| 5.20 | The Custom Curves Dialog for box: pattern formatting. | 74 |
| 5.21 | The Custom Curves Dialog for box: whiskers formatting. | 75 |
| 5.22 | The Custom Curves Dialog for box: percentile formatting. | 75 |
| 5.23 | The Custom Curves Dialog for histogram: pattern formatting. | 76 |
| 5.24 | The Custom Curves Dialog for histogram: spacing formatting. | 76 |
| 5.25 | The Custom Curves Dialog for histogram: data formatting. | 76 |
| 5.26 | The New -> New Surface 3D Plot dialog box. | 77 |
| 5.27 | Export of a selection in a table to an ASCII file. | 77 |
| 5.28 | The FFT... dialog box for a curve. | 78 |
| 5.29 | The FFT... dialog box for a table. | 78 |
| 5.30 | The Integrate... dialog box. | 79 |
| 5.31 | The first step of the Non Linear Curve Fit... dialog box. | 80 |
| 5.32 | The second step of the Non Linear Curve Fit... dialog box. | 81 |
| 5.33 | General plot options dialog: the scale tab. | 82 |
| 5.34 | General plot options dialog: the grid tab. | 83 |
| 5.35 | General plot options dialog: the axis tab. | 83 |
| 5.36 | General plot options dialog: General settings. | 84 |
| 5.37 | The plot wizard dialog box. | 85 |
| 5.38 | The project explorer panel. | 85 |
| 5.39 | The preferences dialog: general parameters for the application. | 86 |
| 5.40 | The preferences dialog: table options. | 88 |
| 5.41 | The preferences dialog: 2D plot options. | 88 |
| 5.42 | The preferences dialog: 3D plot options. | 90 |
| 5.43 | The preferences dialog: fitting options. | 91 |
| 5.44 | The Print dialog. | 91 |
| 5.45 | The Set Column Values... dialog. | 92 |
| 5.46 | The Set Dimensions... dialog for matrix. | 93 |
| 5.47 | The Set Import Options... dialog box. | 93 |

| | | |
|------|---|-----|
| 5.48 | The Set Properties... dialog for matrix. | 94 |
| 5.49 | The Set Values... dialog for matrix. | 94 |
| 5.50 | The surface plot options dialog box. | 95 |
| 5.51 | The surface plot options dialog box with tab 5: aspect ratio. | 97 |
| 5.52 | The text options dialog. | 97 |
| 6.1 | An example of a inverse FFT. | 100 |
| 6.2 | An example of a correlation between two sinus functions. | 101 |
| 6.3 | The results of the Non Linear Curve Fit... | 102 |
| 6.4 | The results of a Fit Linear | 103 |
| 6.5 | The results of a Fit Polynomial... , showing the initial data, the curve added to the plot, and the results in the log panel. | 104 |
| 6.6 | The results of a Fit Boltzmann (sigmoidal) | 105 |
| 6.7 | The results of a Fit Gaussian | 105 |
| 6.8 | The results of a Fit Lorentzian | 106 |
| 6.9 | The results of a Fit Multi-peak ->Gaussian... | 107 |
| 6.10 | Signal after a FFT low pass filter | 108 |
| 6.11 | Signal after a FFT high pass filter | 109 |
| 6.12 | Signal after a FFT band pass filter | 110 |
| 6.13 | Signal after a FFT block band filter | 111 |
| 6.14 | Comparison of the three methods of interpolation | 113 |

List of Tables

| | | |
|-----|--|-----|
| 4.1 | Edit toolbar commands. | 56 |
| 4.2 | File toolbar commands. | 57 |
| 4.3 | Plot toolbar commands | 58 |
| 4.4 | Table toolbar commands. | 59 |
| 4.5 | 3D Plot toolbar commands. | 61 |
| 7.1 | Supported Mathematical Operators | 114 |
| 7.2 | Mathematical Functions | 115 |
| 7.3 | Non-Mathematical Functions | 116 |
| 7.4 | Supported Mathematical Functions | 120 |

Abstract

This document is a handbook for using SciDAVis, a program for two- and three-dimensional graphical presentation of data sets and for data analysis.

This manual is organized in several chapters:

- The [first one](#) describes the main concepts and terms which are used in SciDAVis.
- The [second chapter](#) is a tutorial on how to obtain plots from different data sets. It is the one you need to read first to understand the basics of SciDAVis and to be able to draw plots.
- The three following chapters are descriptions of all the [commands](#), [buttons](#) and [dialogs](#) used in SciDAVis. These chapters are the reference manual of SciDAVis.
- The two following chapters describe more deeply some specific possibilities of SciDAVis, that is the [statistical and mathematical analysis](#) of data, and the [scripting](#).

Chapter 1

Introduction

1.1 What is SciDAVis?

SciDAVis stands for *Scientific Data Analysis and Visualization*. It is a free cross-platform program for two- and three-dimensional graphical presentation of data sets and for data analysis. The plots can be produced from data sets stored in [tables](#) or from analytical functions.

The SciDAVis project started as a fork of QtiPlot with the aim of introducing some changes in design as well as project structure. The QtiPlot development was initiated in 2004 by Ion Vasilief. He was the only programmer until May 2006 when Knut Franke and Tilman Hoener zu Siederdisen joined the project. Not much later, Roger Gadiou officially joined as the main documentation writer. In June 2007, insuperable disagreements among the developers lead to the fork and the creation of the SciDAVis project by Knut Franke and Tilman Hoener zu Siederdisen, soon followed by Roger Gadiou. The project is hosted at [Sourceforge](#).

SciDAVis aims to be a tool for analysis and graphical representation of data, allowing powerful mathematical treatment and visualization of scientific data while keeping a user-friendly graphical user interface. Another keypoint for the SciDAVis project is to be a multi-system software, it should work on Windows, Linux, and OS-X systems.

SciDAVis is a dynamic tool, the plots created from data sets and the spreadsheets owing the data are interconnected. When the spreadsheets are modified, all the objects in the depending plots (curves, axes scales, legends) are automatically updated. For example, deleting a spreadsheet or only some columns will automatically remove all the corresponding curves from the depending plots.

All settings of a complete set of tables, matrix and plots can be saved in project files, having the extension ".sciprj". These project files may be opened using the [command line](#), or using the [File menu](#), or by using the *Open project* icon from the [File toolbar](#).

The plots can be exported to several graphic formats such as JPEG or PNG and inserted as images in documents or presentations.

Data analysis operations (integration, interpolation, FFT, curve fitting, etc...) can be performed on the curves in a 2D plot via the Analysis menu. The results of all these

operations are also stored in the project files. They can be visualized at any moment using the [Results log command](#) and can be deleted from the project file via the [Clear Log Informations command](#).

When the application is launched, a new project file is created consisting of a grey main window (the workspace) which contains an empty table. In order to be operational, this workspace must be populated with tables storing data sets, either by creating empty tables first ([New -> New Table command](#)) and then filling them with data, or by importing ASCII files ([Import ASCII -> Single File... command](#)), which automatically creates new tables.

The user can easily navigate through the objects of a project file using the project explorer or the Windows menu. The project explorer also allows the user to perform various operations on the windows (tables and plots) in the workspace: hiding, minimizing, closing, renaming, printing, etc...

1.2 Command Line Parameters

1.2.1 Specify a File

When starting SciDAVis from the command prompt, you can supply the name of a project file:

```
SciDAVis file_name.sciprj
```

Other file format are also accepted: *.opj*, *.ogm*, *.ogw*, *.ogg* for Origin projects, and *.qti*, *qti.gz* for Qtiplot projects.

The name can also refer to an ASCII file:

```
SciDAVis ASCII_file_name
```

In this latter case a new "untitled" project will be created, containing a spreadsheet with the ASCII data in the file and a 2D plot of all columns as a function of the first column in the file. You must take care of the format of the ASCII file because it will be read with the current values of the [Set Import Options... command](#) dialog. These default values are:

- the default field separator is ; but it can be changed in the [Preferences... command](#) dialog,
- all lines are read,
- the first line is used to name the columns,
- the spaces at the end of the lines are not removed,
- the spaces are not simplified.

1.2.2 Command Line Options

Valid options are:

- -a or --about: show about dialog and exit
- -h or --help: show command line options
- -l=XX or --lang=XX: start SciDAVis in language XX ('en', 'fr', 'de', ...)
- -m or --manual: show SciDAVis manual in a standalone window
- -v or --version: print SciDAVis version and release date
- -x or --execute: execute the script file given as argument

1.3 General Concepts and Terms

Several plots and all the data related to these plots can be save in a *project* file, the project is therefore the main container of SciDAVis. The following screenshot gives an example of a typical session. This example shows the [log panel](#) at the top of the workspace, the [project explorer](#) at the bottom, a [table](#) and a [plot window](#) are shown while other are docked or hidden.

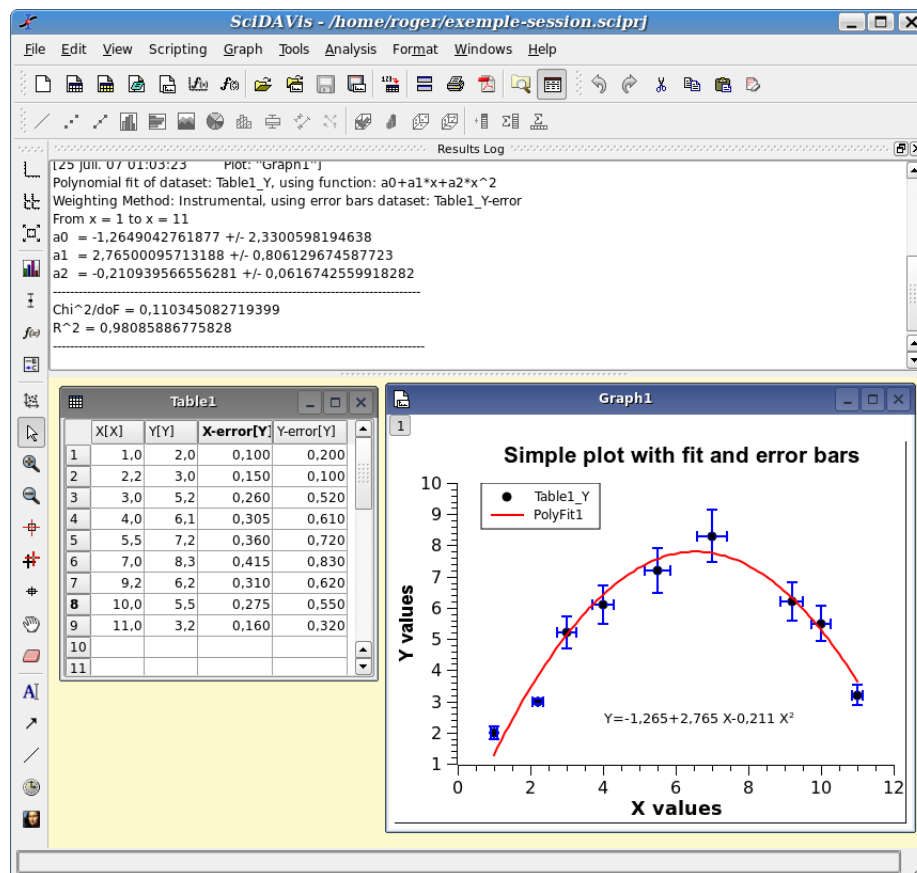


Figure 1.1: A typical SciDAVis session

There are numerous commands available in SciDAVis depending on the element which is selected. Therefore, the main menu bar changes when you select a particular element of the project. Moreover, you can access to the set of commands relevant of an element by activating the context menu with the right button of the mouse.

In a project, the containers which can be used are:

A Table A table is a spreadsheet which can be used to store the datas you are entering. It can also be used to do some calculations and statistical analysis of datas. In each table, columns can be labelled as X-values or Y-values for 2D-plotting, or Z-values if you plan to build a 3D-plot.

A table can be created by the [New -> New Table](#) command. Then there are several ways to fill the table with your data. If you want to read a table from an ASCII file, you can import the data from the file to a table with the [Import ASCII -> Single File...](#) command. You can also enter each value from the keyboard. The

last way to enter your data is to fill the table with the results of a mathematical function ([Set Column Values...](#) command from the [Table menu](#))

A Matrix A matrix is a special table which is used to store the data points for surface 3D plots. It contains Z-values and doesn't include any column or row which could be designed as X-values or Y-values. Nevertheless, you can specify the X-values and the Y-values with the [Set Dimensions...](#) command from the [Matrix menu](#).

A matrix can be created by the [New -> New Matrix](#) command. If you want to read a matrix from an ASCII file, you can import the data of the file to a table with the [Import ASCII -> Single File...](#) command and then convert this table to a matrix with the [Convert to Matrix](#) command. In the same way as for tables, you can also fill matrix with the results of a function $z=(i,j)$ in which i and j are row and column numbers ([Set Values...](#) command from the [Matrix menu](#))

A Graph A graph can contain one or several plots. Each of these plots is contained in a different *layer*, these layers can be arranged in many ways to build matrix of plots.

A new layer can be added to an existing graph with the [Add Layer](#) command from the [Graph menu](#). you can also remove an existing layer with the [Remove Layer](#) command, but if you remove a layer, the plot will be deleted. You can also copy a layer from one graph to another. You can also copy an existing graph into another, the window will be added as a new layer (see the section on [Multilayer Plots](#) for more details).

Plots can be created in several ways. You can select data in tables or matrix and build a plot, or create new plots from functions of one or two variables (see sections [2D plots](#) and [3D plots](#)).

A Note This window is a text container which can simply be used to insert comments into a project, but is really far more powerfull than that. It can be used as a calculator, for executing single commands and for writing scripts.

The Log Window This window is used to store the results of all the calculations which have been done. If this window is not visible, you can find it with the [Project Explorer](#) or with the [Results log](#) command.

The text in the log window is also saved in the project file, so that when you load a previously saved project, the results-log panel is re-filled with the results of the calculations.

The Project Explorer This window is used to list all the windows contained in a project. The [Project Explorer](#) gives a quick access to all elements of a project, hidden or visibles. It can be used to do some operations on the windows related to these items such as hiding a window, renaming windows, etc.

Since the version 0.8.5, a project file can include several independant projects. In this case, the containers of each project are stored in different folders.

1.3.1 Tables

The table is the main part of SciDAVis when working with data. For controlling and converting data the spreadsheet contains a highly customizable table: all colors and font preferences can be set using the [Preferences...](#) command of the [View menu](#). You can resize a table in terms of rows and columns using the [Table menu](#) with [Rows command](#) or [Columns command](#).

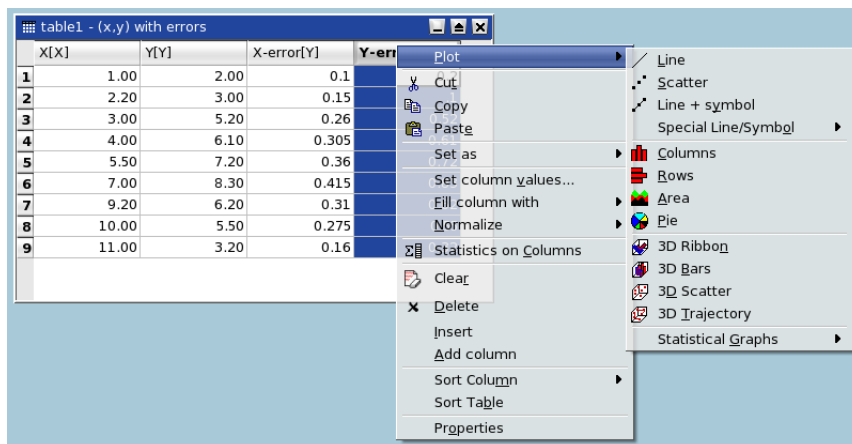


Figure 1.2: The SciDAVis table

Every column of the table has a label and can be assigned a format: numeric, text, date or time. In a spreadsheet, columns can have the following flags: X, Y, Z, X-error, Y-error or can be simple columns without any special flag. The X columns are abscissae columns while the Y columns are ordinates columns used when creating a 2D plot from data. The X-error and Y-error columns can be used in order to add error bars to 2D plots. These flags can be changed using the [Column options dialog](#). To reach this dialog you simply have to double-click on the column label or to use the [Column Options...](#) command of the [Table menu](#).

You can select all the columns of the spreadsheet (Ctrl+A) or only some of them by clicking on the column label while keeping the Ctrl key pressed, or by moving the mouse over the column label. This also allows you to deselect columns.

On the selected columns you can perform various operations: fill with data, normalize, sort, view statistics and finally make plots out of your data. All these functions can be reached by right clicking on the column label or by using the [Table menu](#).

Any other spreadsheet function: rename, duplicate, export, print, close can be reached via the context menu (right click anywhere in the table outside the column labels area).

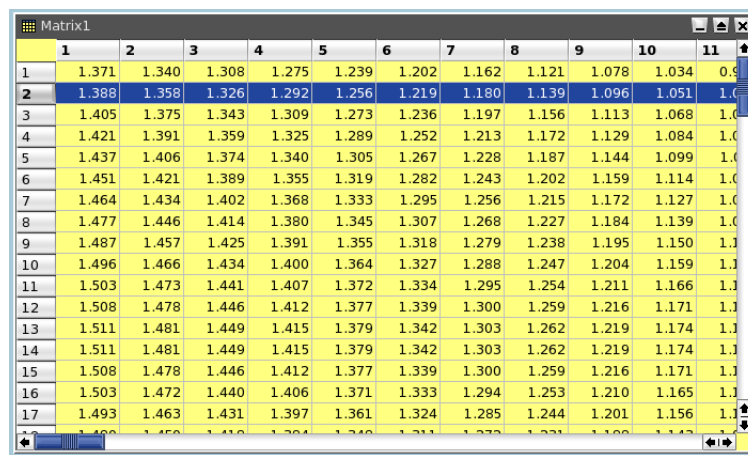
You can cut, copy and paste data between spreadsheets or between a spreadsheet and another application (Excel, Gnumeric, etc...).

You can import single or multiple ASCII files using the [Import ASCII command](#)

from the [File menu](#). The file import options can be changed via the [Set Import Options... command](#). Of course you can export the data from the spreadsheet to a text file using the [Export ASCII command](#).

1.3.2 Matrix

The matrix is a special table which is used for data which depends on two variables. This special table is used to store data for 3D-plots. The difference between a table and a matrix is that there is no special column nor special row for X or Y labels or values. Nevertheless, you can specify an X-scale and an Y-scale with the [Set Dimensions... command](#).



| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
|----|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| 1 | 1.371 | 1.340 | 1.308 | 1.275 | 1.239 | 1.202 | 1.162 | 1.121 | 1.078 | 1.034 | 0.991 |
| 2 | 1.388 | 1.358 | 1.326 | 1.292 | 1.256 | 1.219 | 1.180 | 1.139 | 1.096 | 1.051 | 1.006 |
| 3 | 1.405 | 1.375 | 1.343 | 1.309 | 1.273 | 1.236 | 1.197 | 1.156 | 1.113 | 1.068 | 1.023 |
| 4 | 1.421 | 1.391 | 1.359 | 1.325 | 1.289 | 1.252 | 1.213 | 1.172 | 1.129 | 1.084 | 1.039 |
| 5 | 1.437 | 1.406 | 1.374 | 1.340 | 1.305 | 1.267 | 1.228 | 1.187 | 1.144 | 1.099 | 1.054 |
| 6 | 1.451 | 1.421 | 1.389 | 1.355 | 1.319 | 1.282 | 1.243 | 1.202 | 1.159 | 1.114 | 1.069 |
| 7 | 1.464 | 1.434 | 1.402 | 1.368 | 1.333 | 1.295 | 1.256 | 1.215 | 1.172 | 1.127 | 1.082 |
| 8 | 1.477 | 1.446 | 1.414 | 1.380 | 1.345 | 1.307 | 1.268 | 1.227 | 1.184 | 1.139 | 1.094 |
| 9 | 1.487 | 1.457 | 1.425 | 1.391 | 1.355 | 1.318 | 1.279 | 1.238 | 1.195 | 1.150 | 1.105 |
| 10 | 1.496 | 1.466 | 1.434 | 1.400 | 1.364 | 1.327 | 1.288 | 1.247 | 1.204 | 1.159 | 1.114 |
| 11 | 1.503 | 1.473 | 1.441 | 1.407 | 1.372 | 1.334 | 1.295 | 1.254 | 1.211 | 1.166 | 1.121 |
| 12 | 1.508 | 1.478 | 1.446 | 1.412 | 1.377 | 1.339 | 1.300 | 1.259 | 1.216 | 1.171 | 1.126 |
| 13 | 1.511 | 1.481 | 1.449 | 1.415 | 1.379 | 1.342 | 1.303 | 1.262 | 1.219 | 1.174 | 1.129 |
| 14 | 1.511 | 1.481 | 1.449 | 1.415 | 1.379 | 1.342 | 1.303 | 1.262 | 1.219 | 1.174 | 1.129 |
| 15 | 1.508 | 1.478 | 1.446 | 1.412 | 1.377 | 1.339 | 1.300 | 1.259 | 1.216 | 1.171 | 1.126 |
| 16 | 1.503 | 1.472 | 1.440 | 1.406 | 1.371 | 1.333 | 1.294 | 1.253 | 1.210 | 1.165 | 1.120 |
| 17 | 1.493 | 1.463 | 1.431 | 1.397 | 1.361 | 1.324 | 1.285 | 1.244 | 1.201 | 1.156 | 1.111 |

Figure 1.3: The SciDAVis matrix

The values which are stored in a matrix can be obtained from a function of the form $z=(i,j)$ with the [Set Values... command](#), i and j being the column and row numbers. They can also be read from a file with the [Import ASCII command](#) which allows to read a file in a table, then the table can be converted to a matrix with the [Convert to Matrix command](#) of the [Matrix menu](#).

The operations which can be done on a matrix are limited changes in number presentation and matrix size. The data of a matrix can be used to build a 3D plot with the commands present in the [Plot3d menu](#) and in [3d plot toolbar](#).

1.3.3 Plot Window

The plot window is the one in which the graphic is plotted. It contains at least one layer, which is the main container of the plot window. Each new plot can be inserted in a new layer of this plot window, it has its own geometry and graphic properties (background color, frame, etc). The example presented below shows a graph with two layers which have different geometries.

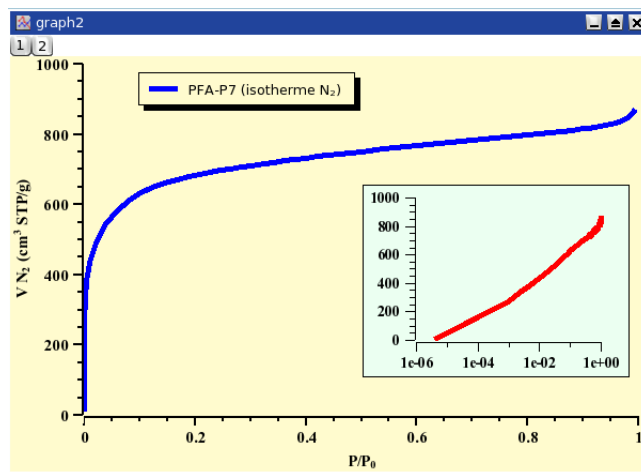




Figure 1.4: An example of SciDAVis 2D graph

Each layer can be activated by clicking on the corresponding gray button   in the top-left corner of the window.

The elements which can be accessed by a double click in a layer are:

- the graph itself: this will open the [Custom Curve Dialog](#). You can then add new curves to the plot, or change the way the curves are plotted.
- The axes or the axes labels: this will open the [General Plot Options Dialog](#). It is used to customize the axes, the numbers and labels of the axes, and the grid.
- Any other text item, including the legend of the plot: this will open the [Text Options Dialog](#) which allows to customize the font of the label and the frame in which it is drawn.

1.3.4 Note

A note can simply be used to insert text (comments, notes, etc) into a project, but is really far more powerful than that. It can be used as a calculator, for executing single commands and for writing scripts.

Evaluation of mathematical expressions and execution of code is done via a note's context menu, the Scripting menu or the convenient keyboard shortcuts.

For information on expression syntax, supported mathematical functions and how to write scripts, see [here](#).

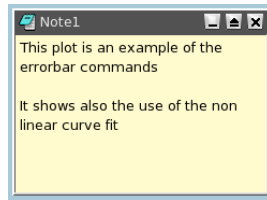


Figure 1.5: The SciDAVis Note Window

You can also change the text input method. *Simple Composing Input Method* is the standard method to enter text in QT applications. *Xim* is the X input method, it is the legacy system of the X window environment to support localized text input. The default choice is the second one, it allows to enter special characters and accents from your localised environment.

1.3.5 Log Window

This window keeps a history of all analysis which have been done in the project. This panel contains the results of all the correlations, fittings, etc.

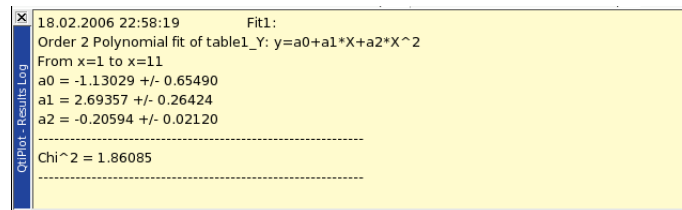



Figure 1.6: The SciDAVis Log window

1.3.6 The Project Explorer

The project explorer can be opened/closed using the [Project Explorer command](#) from the [View menu](#) or by clicking on the  in the [file toolbar](#).

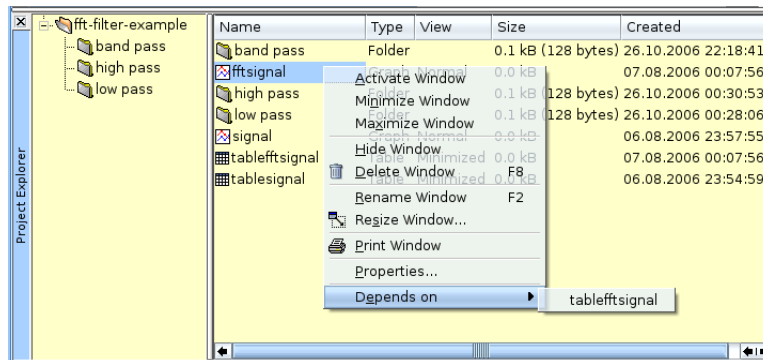


Figure 1.7: The SciDAVis Project Explorer

It gives an overview of the structure of a project and allows the user to perform various operations on the windows (tables and plots) in the workspace: hiding, minimizing, closing, renaming, printing, etc... These functions can be reached via the context menu, by right-clicking on an item in the explorer.

By double-clicking on an item, the corresponding window is shown maximized in the workspace, even if it was hidden before.

You can organize the different objects in folders. When selecting a folder, the default policy is that only the objects contained in it will be showed in the workspace window. You can also display all the objects in the subfolders if you change this policy with the "View Windows" command to "Windows in Active Folder and Subfolders".

Chapter 2

Drawing plots with SciDAVis

2.1 2D plots

A 2D plot is based on curves which are defined by Y values as functions of X values. There are two ways to obtain a 2D plot depending on the way the (X,Y) values are defined:



- You can have your (X,Y) values in a [table](#). You need to select at least one column as X values and one column as Y values. This is specified with the [Column Options... command](#). Then you can select the columns and use one command of the [Plot menu](#) to plot the data.
- If you want to plot a function, you don't need a table. You can use directly the [New -> New Function Plot command](#). This will open the corresponding [dialog box](#) and you will be able to define the mathematical expression of your function.
- The combined way is to define a [table](#), and then to fill in the table with the results of functions. This can be done with the [Set Column Values... command](#). Then you can select the columns and use one command of the [Plot menu](#) to plot the data.

SciDAVis will create a new plot window, and the plot will be inserted in a new layer

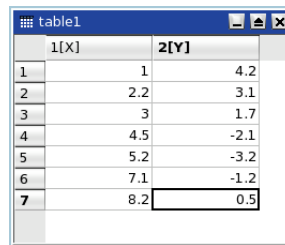
Once the plot is created, you can customize all the graphic items of the plot with the commands of the [Format Menu](#). You can add new items (text labels, lines or arrows, new legend, images) on the plot with the commands of the [Graph Menu](#).

2.1.1 2D plot from data.

The data must be stored in a [table](#). There are several possibilities to insert your (X,Y) values in the table: you can write them directly from the keyboard, or read them from a file. Here we will use the first solution, refer to the [Import ASCII command](#) to use the second one.


The first step is to create an empty project with the [New -> New Project command](#) from the [File menu](#), you can also use the key Ctrl-N or the  icon from the [File toolbar](#). Then create a new table with the [New -> New Table command](#) from the [File menu](#) or with the Ctrl-T or with the  icon from the [File toolbar](#).

At its creation, the table has two column (one for X and one for Y) and 32 rows. You can add rows and columns by selecting a row or a column and using the right button of the mouse, you can also modify the number of rows and columns with the [Rows command](#) and [Columns command](#) from the [Table menu](#). You enter your values, and obtain this table:



| | 1[X] | 2[Y] |
|---|------|------|
| 1 | 1 | 4.2 |
| 2 | 2.2 | 3.1 |
| 3 | 3 | 1.7 |
| 4 | 4.5 | -2.1 |
| 5 | 5.2 | -3.2 |
| 6 | 7.1 | -1.2 |
| 7 | 8.2 | 0.5 |

Figure 2.1: A simple 2D plot: the table.

The you have to select the two columns, and build your plot (here a simple 2D scatter) with the [Scatter command](#) from the context menu, or by clicking on the corresponding  icon from the [Plot toolbar](#) or with the [Scatter command](#) from the [Plot menu](#). A plot is created which uses the default options for all elements. You can customize these default options with the [preferences dialog](#). With the default options, you obtain the following plot:

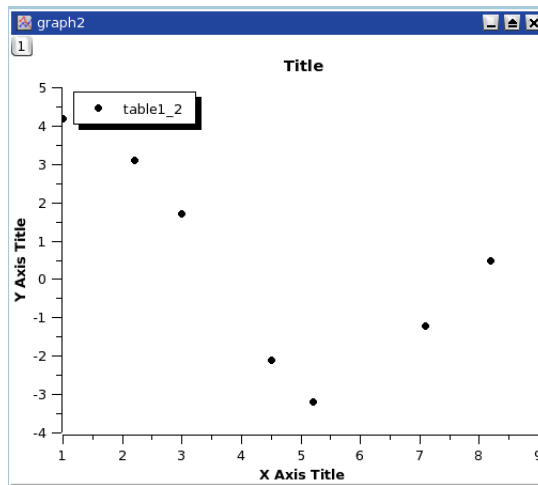


Figure 2.2: A simple 2D plot: the default plot.

You can then customize your plot. By double clicking on the points, you open the [Custom curves dialog](#) which is used to modify the symbols. Then a double-click on the axes opens the [general plot options dialog](#), and you can change the scales, the fonts for the axes labels, etc. You can also add grid lines on X or Y axes, etc. Finally, a double click on each text item (X title, Y title, plot title) allows to change the text and the presentation of these elements. The final plot is:

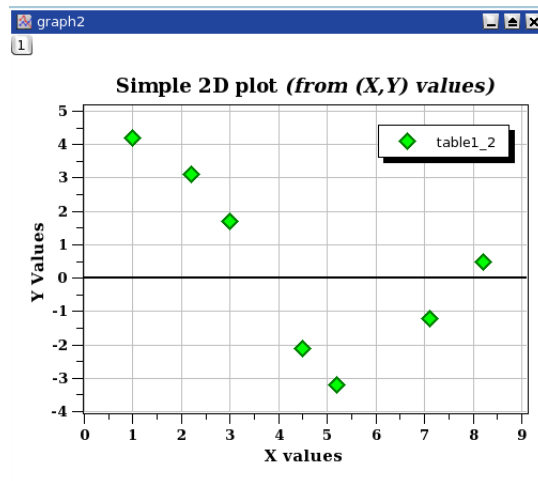



Figure 2.3: A simple 2D plot: the plot finished.

Finally, you have to save your project in a '.sciprj' file with the [Save Project com-](#)

mand from the [File menu](#) or with the Ctrl-S or with the  icon from the [File toolbar](#). Depending on your application, you can export your plot to a standard image file with the command [Export Graph -> Current](#) command from the [File menu](#) (or with the Alt-G keycode).

There are several types of plots which can be built from a table. They are presented in the [Plot menu](#)

It is possible to use up to four axes for the data:

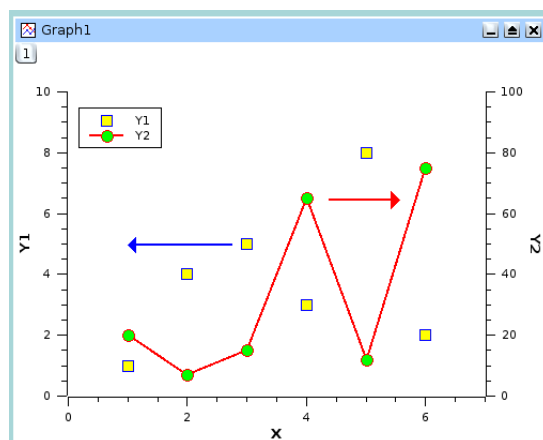
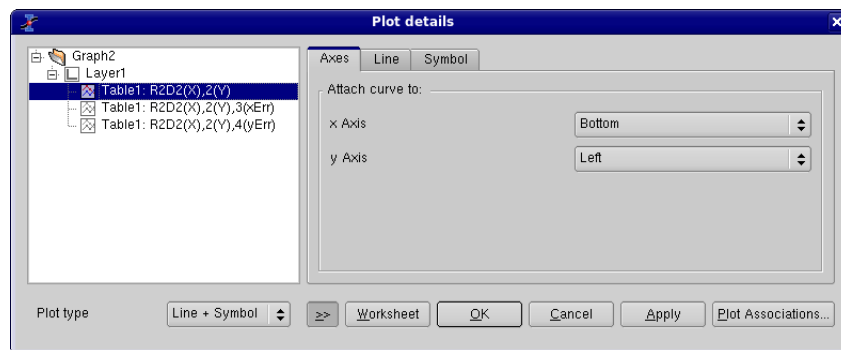


Figure 2.4: A 2D plot with two Y axis.


In addition to the customization which has been already described, the axes which are used for each curve were defined with the [Custom Curves](#) Dialog, and two arrows were added with the [Draw Arrow](#) command.



2.1.2 2D plot from function.

There are two ways to obtain such a plot: you can plot directly a function or fill a table with the values calculated from this function before doing a plot in the classical way.

2.1.2.1 Direct plot of a function.

If you just want to plot a function, you can use the [New -> New Function Plot](#) command from the [File menu](#) or with the [Ctrl-F](#) or with the  icon from the [File toolbar](#).

This command will open the [Add Function Curve](#) dialog. You can then enter the expression of your mathematical function, the X range used for the plot, and the number of points used in this X range. Beside classical $Y=f(X)$ functions, parametric and polar functions can be defined.

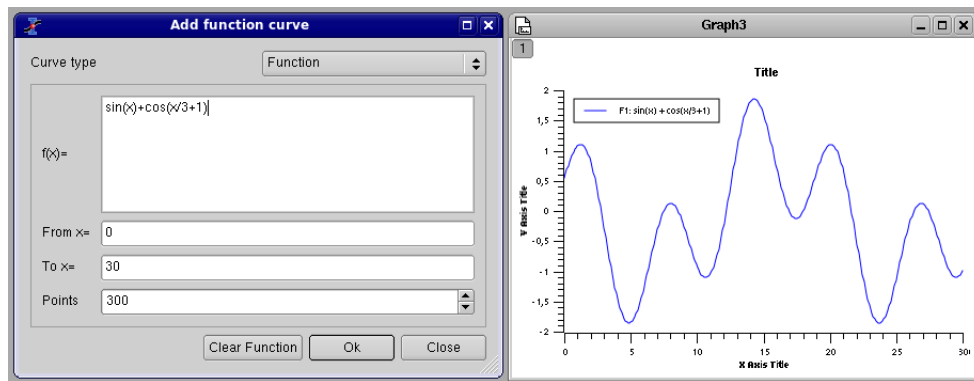


Figure 2.5: Direct plot of a function.

2.1.2.2 Filling of a table with the values of a function.

If you just want to work not only with the plot but also with the data, you can create a new table as explained in the [previous section](#). Then you can fill this table with the values of a function with the [Set Column Values...](#) command.

To obtain the same plot as in the previous example, you need to create a new table (key [Ctrl-T](#)), then select the first column and use the command [Set Column Values...](#) command from the context menu, or from the [Table menu](#). The row number symbol is i , so you can enter the function expression $i/10$ and use 300 rows.

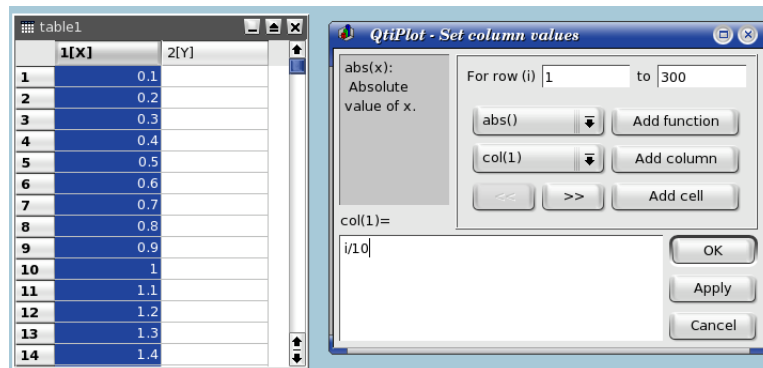


Figure 2.6: Function plot: filling of the X column.

The second step is to select the second column and use the same command. The expression is a function of the X values, that is the first column named *col(1)*.

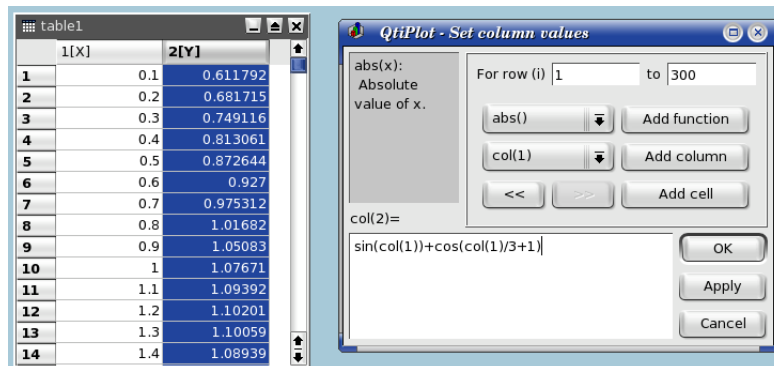


Figure 2.7: Function plot: filling of the Y column.

Once the table is ready, you just have to build the plot as explained in the previous section.

2.2 3D plots

3D plot are generated from data defined as $Z=f(X,Y)$. As for 2D plots, there are two ways to obtain a 3D plot depending on the way the (X,Y,Z) values are defined:

- You can have your Z values in a [matrix](#). SciDAVis will consider that all the data present in the matrix are Z values, and the X and Y values can be defined as a function of the columns and rows numbers.

The data in the matrix can be entered in several ways:

- one by one from the keyboard,
 - by reading an ascii file in a table and converting the table into a matrix,
 - by setting the values with a function.
- If you want to plot a function, you don't need a matrix. You can use directly the [New -> New Surface 3D Plot command](#). This will open the corresponding [dialog box](#) and you will be able to define the mathematical expression of your function.

There are several kinds of 3D plots which can be selected, see the [Plot3d menu](#) section of the [reference chapter](#) for a list of the available plots.

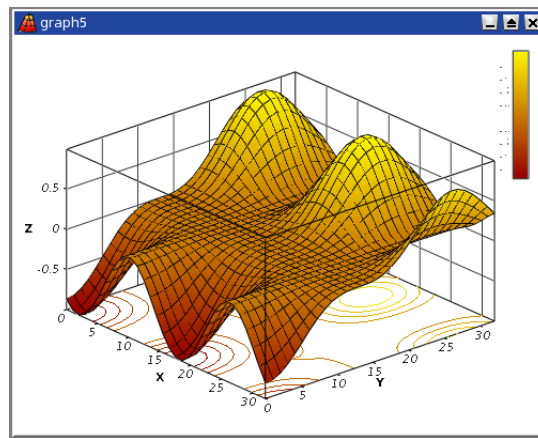


Figure 2.8: Example of a 3D Plots.

The 3D plots use OpenGL so you can easily rotate, scale and shift them with the mouse. Via the 3D plot settings dialog or via the Surface 3D Toolbar you can change all the predefined settings of a three dimensional plot: grids, scales, axes, title, legend and colors for the different elements.

There are several types of plots which can be built from a matrix. They are presented in the [Plot3d menu](#)

2.2.1 Direct 3D plot from a function

This is the simplest way to obtain a 3d plot. It is done with the [New -> New Surface 3D Plot command](#) from the [File menu](#) or directly with the Ctrl-Alt-Z. This will open the following [dialog box](#):

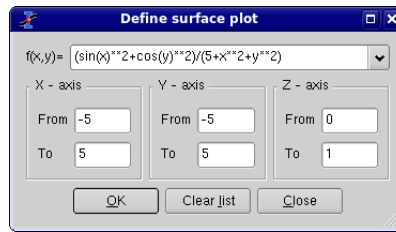


Figure 2.9: Definition of a new surface 3D plot

You can enter the function $z=f(x,y)$ and the ranges for X, Y and Z. Then SciDAVis will create a default 3d plot:

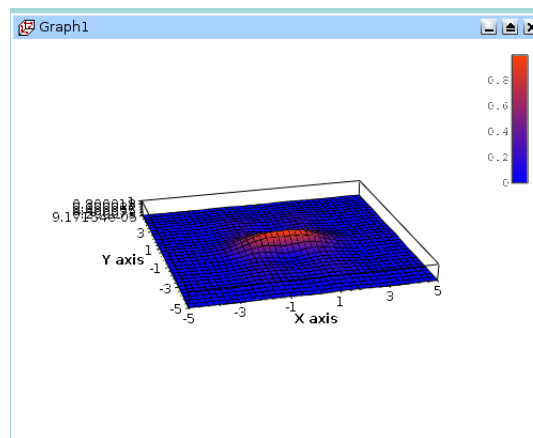


Figure 2.10: The 3D surface plot created by default

You can then customize this plot by opening the [Surface plot options dialog](#). You can modify the axis ranges and parameters, add a title, change the colors of the different items, and modify the aspect ratio of the plot. In addition, you can use the different commands of the [3d plot toolbar](#) to add grids on the walls or to modify the style of the plot. After some modifications, you can obtain the following plot:

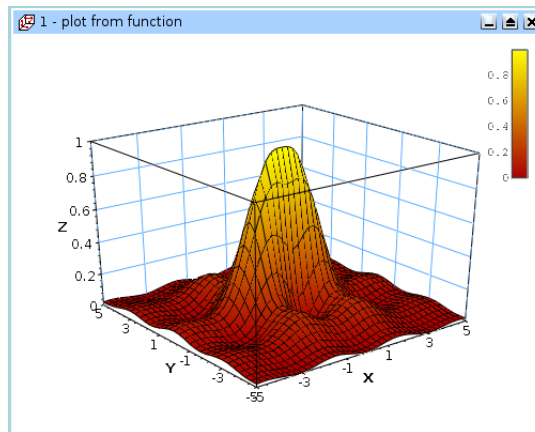


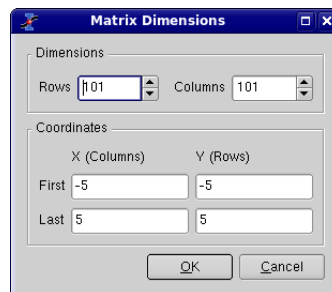
Figure 2.11: The 3D surface plot after customizations

If you want to modify the function itself, you can use the **surface...** command which can be activated from the context menu with a right click on the 3D plot. This will re-open the [define surface function dialog box](#).

2.2.2 3D plot from a matrix

The second way to obtain a 3D plot is to use a [matrix](#). Therefore, the first step is to fill the matrix. This can be done by defining a function.

The [New -> New Matrix command](#) create a default empty matrix with 32x32 cells. Then use the [Set Dimensions... command](#) to modify the number of rows and columns of the matrix. This [dialog box](#) is also used to define the X and Y ranges.



Then use the [Set Values... command](#) to fill the cells with numbers. The ranges of X and Y defined in the previous step are not known by this dialog box, then the function is defined with the row and column numbers (i and j) as entry parameters (see the section [set-values](#) for details).

The other way to obtain a matrix is to import an ASCII file into a table with the [Import ASCII -> Single File... command](#) from the [File menu](#). The table can then be

transformed in a matrix with the command [Convert to Matrix command](#) from the [Table menu](#).

You can then use this matrix to build a 3D plot with one of the command of the [Plot menu](#).

2.3 Multilayer Plots

The multilayer windows can contain multiple plots (layers) with different characteristics. Each layer has a corresponding button, which displays a number and is pressed when the layer is the currently active layer. There is only one active layer at a time, and the plot tools (zoom, cursors, drawing tools, delete and move points) can only operate on this layer. Each plot can be made active by clicking on it or on its corresponding button.

To arrange the layers use the [Arrange Layers dialog](#). You can add or remove layers with the [Add Layer command](#) and [Remove Layer command](#) or copy/paste layers from one multilayer window to another. All these functions can be reached via the [Graph menu](#), by using the [Plot toolbar](#) or via the context menu (right click in the multilayer window anywhere outside a plot area).

You can resize and move a layer using the Layer geometry dialog. You can also arrange and resize the plots by hand. A whole plot can be moved by drag and drop: click on the plot and keep the left mouse button pressed.

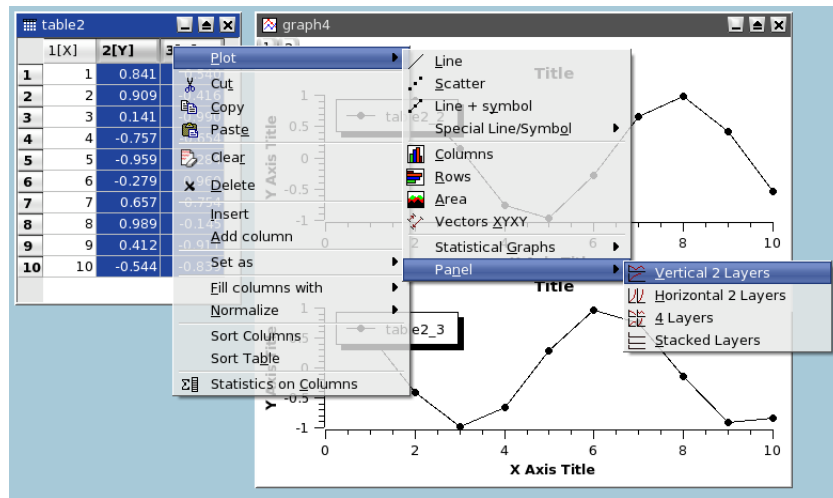
By keeping the Shift key pressed and dragging the border of a plot you can scale a plot as needed. When moving the mouse over the borders of a plot, you will see the corresponding arrows.

You can also use the mouse wheel in order to resize the layers: keeping the Ctrl key pressed and scrolling will resize the height of the plot canvas, while keeping the Alt key pressed and scrolling will resize its width. By keeping the Shift key pressed and scrolling you can resize the plot in both dimensions.

2.3.1 Building a multilayer plot panel

This is the simplest way to obtain a multilayer plot. It can be used if you want to build a panel of plots with a simple arrangement: 2 plot in a row or in a column, or 4 plots in 2 rows and 2 columns.

You can select two columns with Y-values in a table, and then use one of the **Panel** commands in the [Plot menu](#). SciDAVis will create a panel of plots in which the size of the different elements of each plot are synchronized.

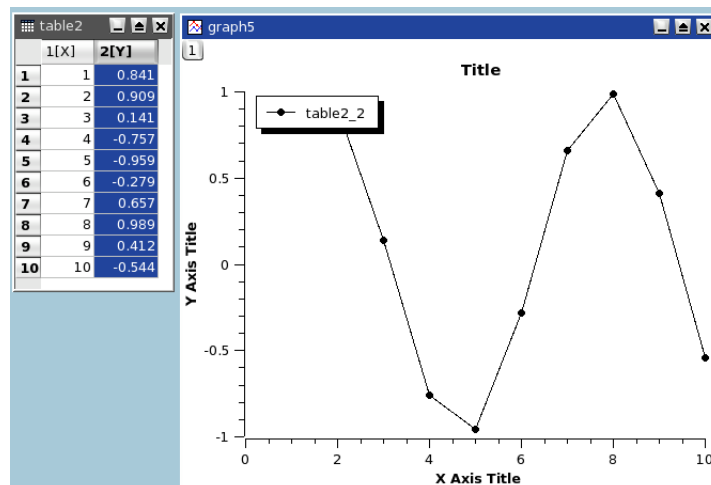


You can then customize the two plots, if you want to change the arrangement of the panel, you can use the [Arrange Layers](#) command from the [Graph](#) menu. It must be reminded in this case that each plot is in a layer with a surface which is the half or the quarter of the window surface area. So, if you want to share an element between the two plots (for example a text label), you need to add it in a new layer (see the [Add Text](#) command for more details).

2.3.2 Building a multilayer plot step by step

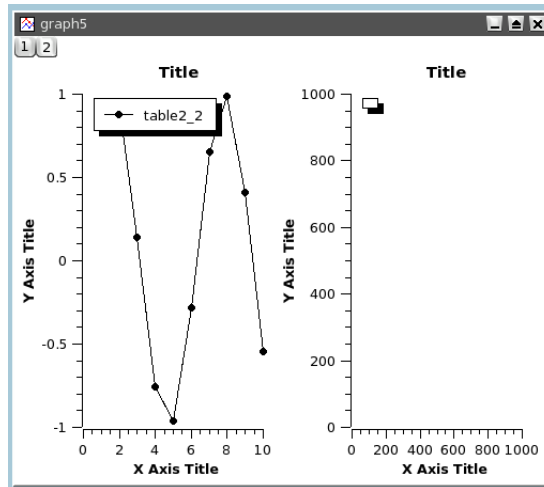
If you need to build a more complex multilayer plot, you can define it step by step.

The first step is to build your first plot, for example from two columns of a table. We obtain a standard plot window:

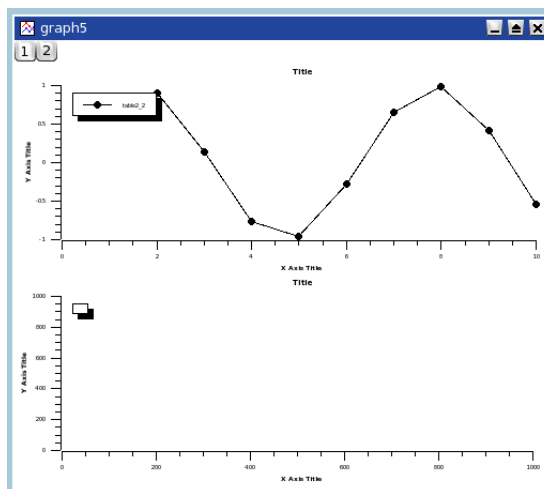


Then, select the plot window and use the [Add Layer](#) command from the [Graph](#) menu. This will activate the [Add Layer](#) dialog. If you choose "Guess" you will obtain

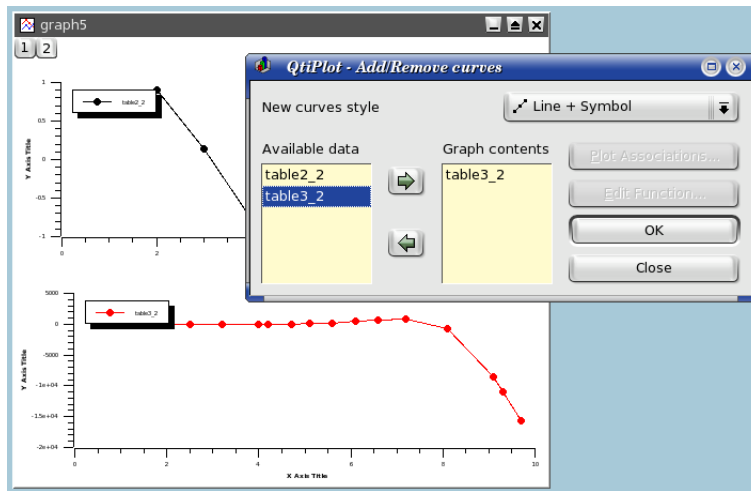
a panel with two columns, if you choose "corner" you will obtain two superposed layers, you can then modify these two layers.



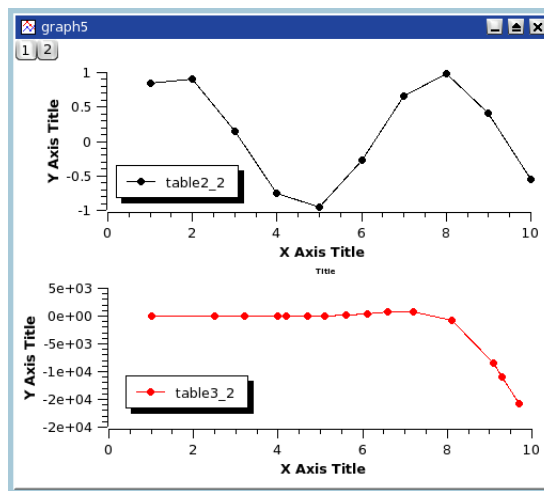
If you want to build a panel with two rows, you can use the [Arrange Layers command](#) to convert this panel.



Then select the second void plot and use the [Add/Remove Curves...](#) command to select the Y-values from one of the tables of the project.



After this, you can customize your plot. At the end, the modifications done on the axis or on the axis labels may have modified the geometry of the two plots. You can synchronize again the two plots by applying again the [Arrange Layers command](#).



Chapter 3

Command Reference

The active items in the menus depend on the active window in the project. If the active window is a spreadsheet, then all the items linked to table functions are enabled and the others are automatically disabled.

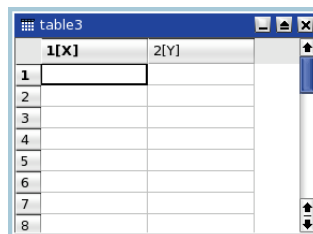
3.1 The File Menu

These commands can also be done by clicking on the *New Project* icon from the [File Toolbar](#)

File-> New ->

New -> New Project (Ctrl-N) Creates a new SciDAVis project file. If a project is open and saved, it will be closed. If a project is open is not saved, a dialog will be open to ask if the current project has to be saved.

New -> New Table (Ctrl-T) Creates a new spreadsheet into the project. This empty table will have 30 rows and 2 columns. This number of rows and columns can be changed with the [Rows command](#) and [Columns command](#) of the [Table menu](#).

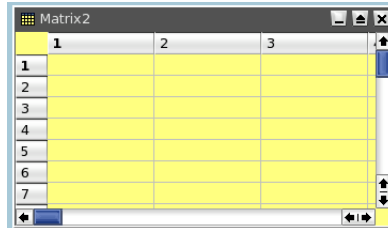


The screenshot shows a window titled 'table3' containing a spreadsheet. The spreadsheet has two columns: '1[X]' and '2[Y]'. The first column is highlighted. The rows are numbered 1 through 8 on the left side. The window has standard OS controls (minimize, maximize, close) in the top right corner.

| | 1[X] | 2[Y] |
|---|------|------|
| 1 | | |
| 2 | | |
| 3 | | |
| 4 | | |
| 5 | | |
| 6 | | |
| 7 | | |
| 8 | | |

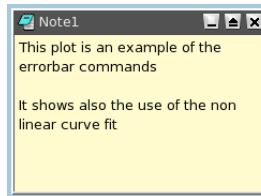
The properties of each column (format of numbers, width, etc) can be modified by the [Column Options...](#) command of the [Table menu](#). See the [table section](#) for more details.

New -> New Matrix Creates a new Matrix into the project. The empty matrix will have 32x32 cells, these dimensions can be changed by the [Set Dimensions...](#) command of the [Matrix menu](#)

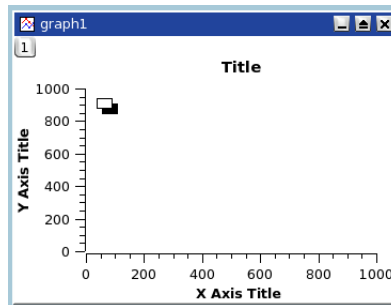


See the [matrix section](#) for more details.

New -> New Note Creates a new note window in the project. A note is a simple text window which can be used to add comments to the current project.



New -> New Graph Creates a new empty 2D plot in the project. This default graph is just a framework in which you can add curves with the [Add/Remove Curves...](#) command.



New -> New Function Plot (Ctrl-F) Opens a [dialog](#) allowing to create a plot by specifying an analytical function. See the [2D plot section](#) of the tutorial for a general overview of this function.

This function can be defined in cartesian, parametric or polar coordinates, see the [Add Function...](#) command for more details.

New -> New Surface 3D Plot (Ctrl-Alt-Z) Opens a [dialog](#) allowing to create a 3D plot by specifying an analytical function. Only cartesian coordinates are available. See the [3D plot section](#) of the tutorial for more detail on this function.

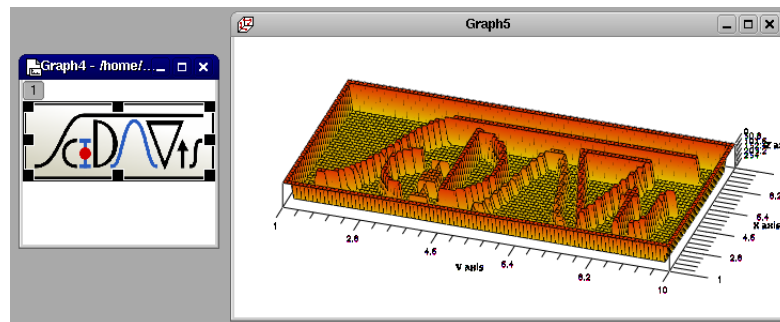
File -> Open (Ctrl-O) Opens an existing SciDAVis project file (default file extension .sciprj). If your project has been save in a compressed format, you must select the .sciprj.gz file format.

This command can also be used to open projects which have been built with the *Origin* software.

File-> Recent Projects Opens a list of the most recently used SciDAVis project files. You can open one of these files by selecting it from the list. If the file doesn't exist anymore an error message will pop-out and the file will be automatically deleted from the list.

File-> Open Image File This command loads an image file in a SciDAVis project. This image can be resized and then inserted in another 2D plot. It is in this case similar to the [Add Image command](#). This image can also be used to generate an intensity matrix (see the [Import Image... command](#)).

File-> Import Image... With this command, an image is loaded in the SciDAVis project and converted to an intensity matrix. For each pixel, an intensity between 0 and 255 is computed from the intensities of the three colors red, green and blue.



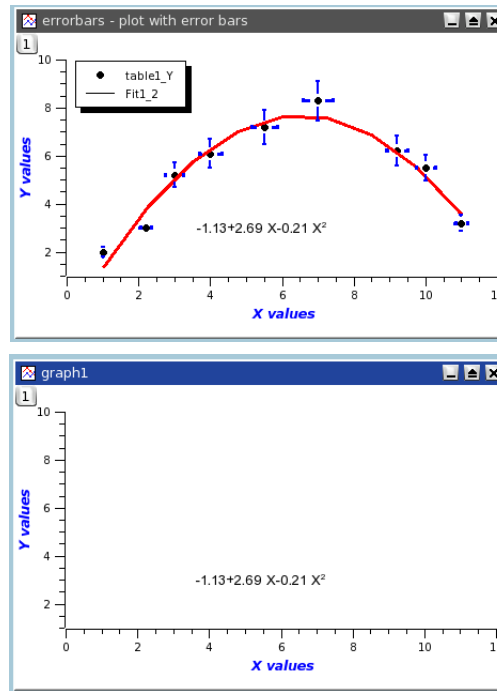
This example shows the 3D plot which has been drawn from the matrix obtained with the SciDAVis logo.

File-> Save Project (Ctrl-S) Saves the actual project. If the project hasn't been saved yet ("untitled" project), a dialog will open, allowing to save the project to a specific location. In a project file all settings and all plots are stored in ASCII format. If the project include large tables, it may be usefull to save the project in a compressed file format. The free zlib library is used to build files in gzip formats (.sciprj.gz).

File-> Save Project as... Saves the actual project under a file name different from the current one.

File -> Open Template Opens an existing template SciDAVis plot file (.qpt). This command will create a new empty plot with the same graphical parameters (window geometry, fonts, colors, etc).

The first figure is the initial plot saved as a template, and the second one is the empty plot created by the **Open Template**.

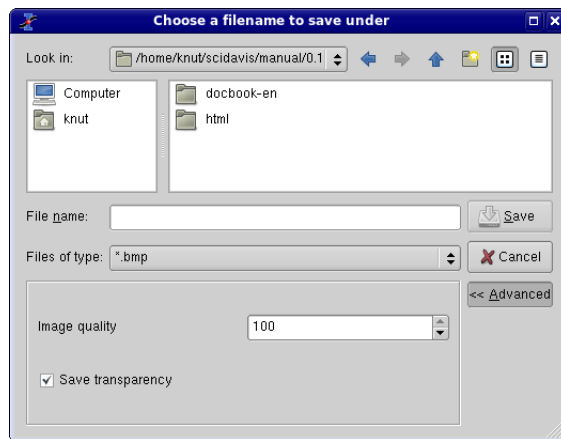


You just have to add curves with the [Add/Remove Curves...](#) command, but the style used to draw the curves is not kept in the template.

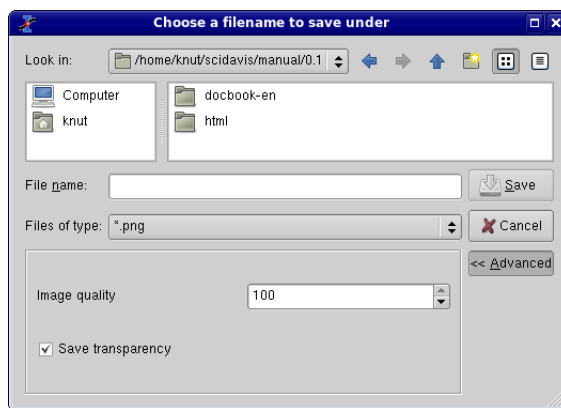
File -> Save as Template Save the active plot as a SciDAVis template file (.qpt). In this template, the graphical parameters of the plot, together with the text labels (axis, etc) are restored, but the style used to draw the curves and the scales are not saved.

File -> Export Graph The plot can be exported into several different image formats. You can define some parameters to customize your image file by checking the *show options* check box. Depending on the chosen image format, the available options are not the same.

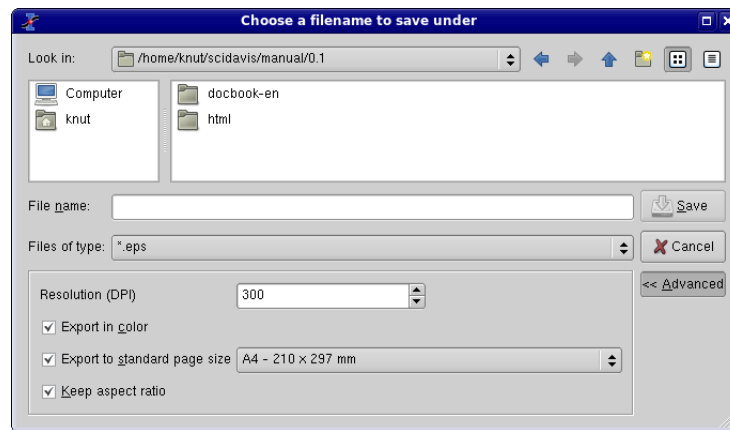
For *bmp*, *pbm*, *jpeg*, *xbm*, *pgm*, *ppm* image formats, the only option is the quality of the image, this parameter between 0 and 100% defines the compression ratio. The higher it is, the best the quality is but the larger the file is.



For *png* and *xpm*, you can choose to use a transparent background.



For *eps* file format, the option dialog is different. The parameters available are: the size of the paper which is used to draw the plot, and the orientation of the paper sheet. The plot aspect ratio will be adapted to the sheet size and orientation. In addition, you can define the resolution. The default value is 84. If you increase this parameter, the quality of the graphic elements will be better (but the overall size of the plot will be unchanged).



The last format which can be selected is the Scalable Vector Graphic format. With this format, the files can be modified in vector drawing software such as [Sodipodi](#).

Export Graph -> Current (Alt-G) Here you have the possibility to save the active plot under different image formats.

Export Graph -> All (Alt-X) Here you have the possibility to save all plots of the project under different image formats.

File-> Print (Ctrl-P) Prints the active plot. A print [dialog](#) is opened where you can select the printer, different paper sizes, etc.

File-> Print All Plots Prints all plots of the projects. A print dialog is opened where you can select the printer, different paper sizes, etc.

File -> Export ASCII Opens the [Export ASCII dialog](#) allowing to save the data from the active spreadsheet to an ASCII file.

File -> Import ASCII -> The options for the importation of ASCII datas are set by the [Import dialog](#) which is activated by the [Set Import Options...](#) command.

Import ASCII -> Single File... Imports a single ASCII file into the project by creating a new spreadsheet storing the data from the file.

Import ASCII -> Multiple Files... Imports several ASCII files into the project by creating a new spreadsheet for each file. The new spreadsheets are automatically arranged in a cascade style.

You can choose to put each data file in a separate table, or join all the data files in one table. Remind that if you choose this last solution, only one column can be selected as X values.

Set Import Options... Opens the [Import dialog](#) allowing to change import options like: the column separator or the number of first lines to be ignored.

File -> Quit (Alt-F4) Closes the application. You will be asked whether you want to save your last changes or not.

3.2 The Edit Menu

Edit -> Undo (Ctrl-Z) Restores the last modified table at the state it had after the last "Save Project" operation. Not available for plot windows.

Edit -> Redo (Ctrl-R) Restores the modifications in a table after a "Undo" operation. Function not available for plot windows.

Edit -> Cut Selection (Ctrl-X) Copies the current selection to the clipboard and deletes the selection. It currently works for spreadsheets and for 2D plots objects.

Edit -> Copy Selection (Ctrl-C) Copies the current selection to the clipboard. It currently works for spreadsheets and for 2D plots objects.

Edit -> Paste Selection (Ctrl-V) Pastes the content of the clipboard to the active window.

Edit -> Delete Selection () Clears the current selection. It currently works for spreadsheets and for 2D plots objects.

Edit -> Delete Fit Tables Each time you do a fit of your data with some mathematical model, a new table is created to put the results of the fit (i.e. the values computed by the model). These tables can be used to plot comparisons of experimental and fitted values.

If you have done several fitting tentatives, a number of unused table may be present in your project. This command allows to remove the results of all the different fits that you have tested.

Edit -> Clear Log Informations Deletes from the project file all the history information about the analysis operations performed by the user. The [log panel](#) is then empty.

3.3 The View Menu

View -> Plot Wizard (Ctrl-Alt-W) Opens the [Plot Wizard dialog](#).

View -> Project Explorer (Ctrl-E) Opens/Close the [Project Explorer](#), which gives an overview of the structure of a project and allows the user to perform various operations on the windows (tables and plots) in the workspace.

View -> Results log Opens/Close a [panel](#) displaying the historic of the data analysis operations performed by the user.

View -> Preferences... Opens the [Preferences dialog](#).

3.4 The Graph Menu

This menu is only active when a plot window is selected.

Graph -> Add/Remove Curves... (Alt-C) Opens the [Add/Remove Curves... dialog](#), allowing to easily add or remove curves from the active plot layer. This dialog can also be used to modify a curve which is already plotted by changing the columns which are used as X or Y values.

Graph -> Add Error Bars... (Ctrl-B) Opens the [Add Error Bars... dialog](#). You can add error bars on X and/or on Y values on an existing plot.

Graph -> Add Function... (Ctrl-Alt-F) Opens the [Add Function... dialog](#). This command allows to add a new curve on an existing plot.

Graph -> New Legend (Ctrl-L) Adds a new legend object to the active plot layer. You can have more than one legend on a plot. These legends can then be customized by double clicking on a given legend.

Graph -> Add Text (Alt-T) Opens a [dialog](#) allowing you to select whether the text is to be added to the active plot layer or on a new layer. The cursor changes to an edit text cursor. Next, you must click in the plot window to specify the position of the new text box. A text dialog will pop-up allowing you to type the new text to be displayed and all its properties (color, font, etc...)

Graph -> Draw Arrow (Ctrl-Alt-A) Changes the active layer operation mode to the drawing mode. You must click on the layer canvas in order to specify the starting point for the new arrow, and then click once more to specify its ending point. You can edit the new arrow using the Arrow dialog. You can switch back to the normal operating mode by clicking the "Pointer" icon in the Plot toolbar.

Graph -> Draw Line (Ctrl-Alt-L) Changes the active layer operation mode to the drawing mode. You must click on the layer canvas in order to specify the starting point for the new arrow, and then click once more to specify its ending point. You can edit the new arrow using the line dialog. You can switch back to the normal operating mode by clicking the "Pointer" icon in the Plot toolbar.

Graph -> Add Time Stamp (Ctrl-Alt-T) This command is used to add a special label in the current plot which contains the current date and time. The properties of this label can be customized like any other label that is added by the [Add Text command](#).

A timestamp label is not modified if the plot is modified, saved, etc.

Graph -> Add Image (Alt-I) Opens a file dialog allowing you to select an image to be added to the active plot layer. Only a link to the image file will be saved into the project file and not the image itself. The new image is added to the left-top corner of the layer and can be moved by drag-and-drop.


Graph -> Add Layer (Alt-L) Opens a [dialog](#) allowing you to select whether the new layer is to be added to the left-top corner of the plot window or to a best-guess position (based on a layer positioning algorithm in columns and rows).

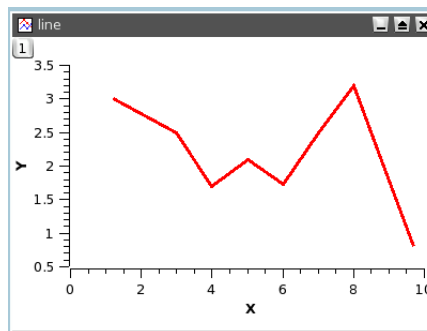
Graph -> Remove Layer (Alt-R) Deletes the active layer and prompts out a question dialog allowing you to choose whether the remaining layers should be automatically re-arranged or not.


Graph -> Arrange Layers (Alt-A) Opens the [Arrange layers dialog](#), allowing you to custom the layout of the active 2D plot window.

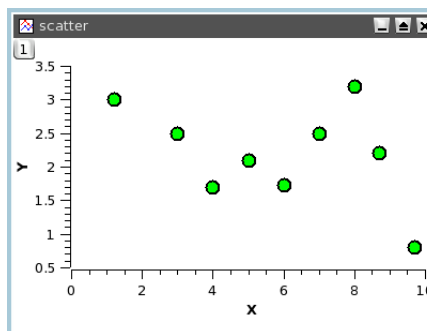
3.5 The Plot Menu


This menu is active only when a table is selected. These commands allow to plot the data selected in the active table.

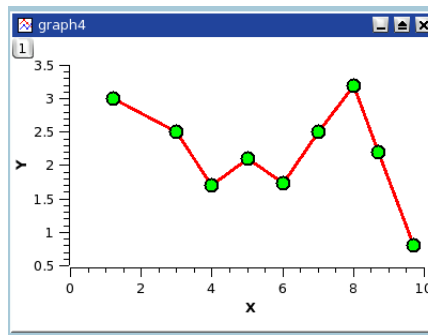
Line Plots the selected data columns in the active table window using the "Line" style. This command can also be activated by clicking on the  icon of the [Table toolbar](#). Once the plot is created, the drawing of the data series can be customized with the [Custom curves dialog](#).



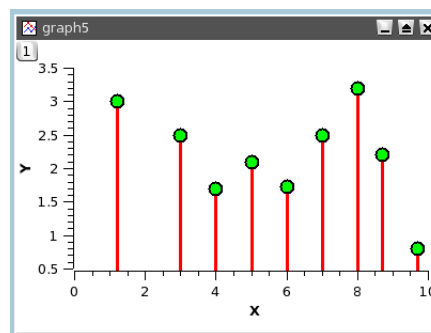
Scatter Plots the selected data columns in the active table window using the "Scatter" style. This command can also be activated by clicking on the  icon of the [Table toolbar](#). Once the plot is created, the drawing of the data series can be customized with the [Custom curves dialog](#).



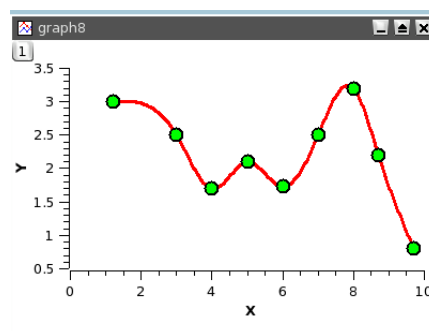
Line+Symbol Plots the selected data columns in the active table window using the "Line + Symbol" style. This command can also be activated by clicking on the  icon of the [Table toolbar](#). Once the plot is created, the drawing of the data series can be customized with the [Custom curves dialog](#).



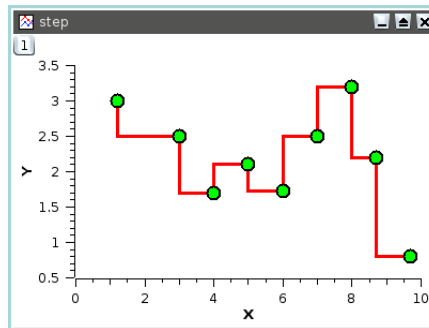
Special Line+Symbol -> Vertical Drop Lines Plots the selected data columns in the active table window using the "Vertical drop lines" style. Once the plot is created, the drawing of the data series can be customized with the [Custom curves dialog](#).



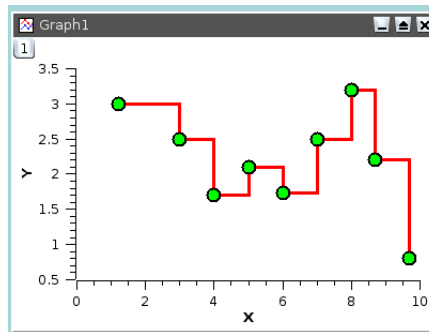
Spline Plots the selected data columns in the active table window using the "Spline" style. Once the plot is created, the drawing of the data series can be customized with the [Custom curves dialog](#).



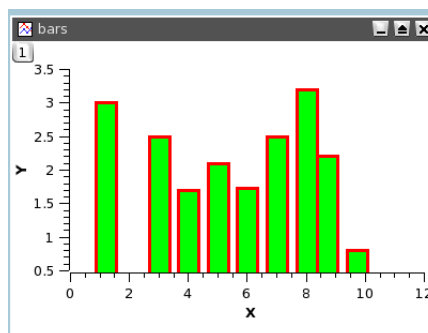
Vertical Steps Plots the selected data columns in the active table window using the "Vertical Steps" style. Once the plot is created, the drawing of the data series can be customized with the [Custom curves dialog](#).



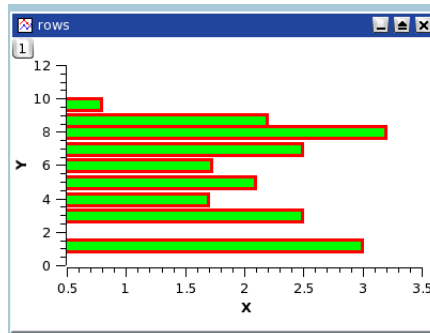
Horizontal Steps Plots the selected data columns in the active table window using the "Horizontal Steps" style. Once the plot is created, the drawing of the data series can be customized with the [Custom curves dialog](#).



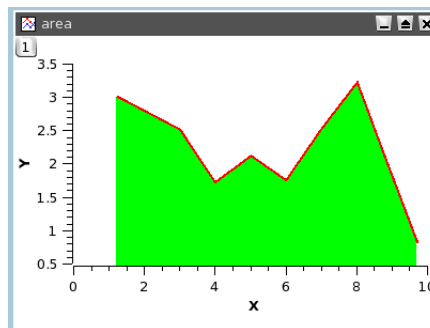
Columns Plots the selected data columns in the active table window using the "Columns" style, that is vertical bars.



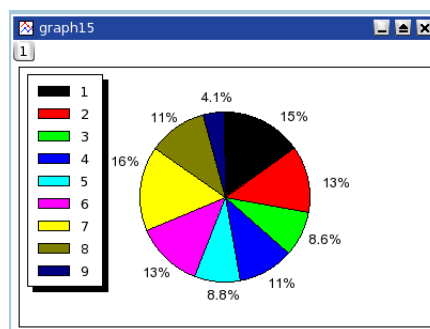
Rows Plots the selected data columns in the active table window using the "Rows" style.



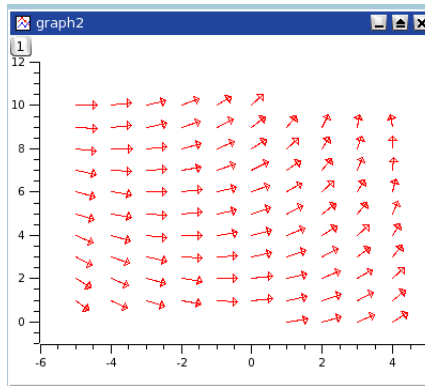
Area Plots the selected data columns in the active table window using the "Area" style.



Pie Creates a 2D Pie plot of the selected column in the active table window (only one column allowed).



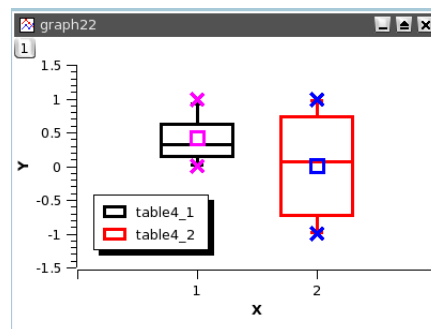
Vectors YYYY Creates a vectors plot of the selected column in the active table window. You must select four columns for this particular type of plot. The two first columns give the coordinates for the starting points of the vectors, the two last columns giving the information regarding the end points.



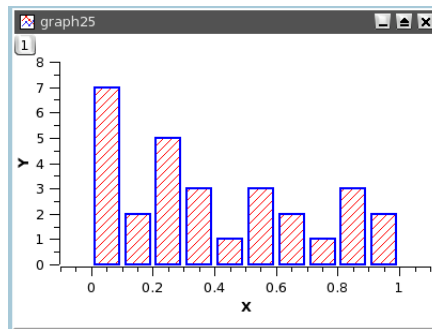
Vectors XYAM Creates a vectors plot of the selected column in the active table window. You must select four columns for this particular type of plot. The two first columns give the coordinates for the starting points of the vectors, the two last columns giving the angle (in radians) and the magnitude of the vectors.

Statistical Graphs -> Statistical plot will not give a direct drawing of the data selected in the table, but they will give a representation of the frequency distribution of the Y-values.

Statistical Graphs -> Box Plot Creates a box plot of the selected data columns in the active table window. This type of plot is used to give a graphical representation of the some classical parameters of the frequency distribution such as the mean of data, the min and max values, the position of the 95 and 5 percentiles, etc. The choice of the statistical parameters and the graphical parameters can be modified with the [Custom curves dialog](#).



Statistical Graphs -> Histogram Creates a frequency histograms of the selected data columns in the active table window. The default binning uses 10 steps between the max and the min of Y-values. This can be modified with the [Custom curves dialog](#).



With this command, a frequency distribution is computed from your data. If you want to draw an histogram directly from values, use the **Bars**.

Statistical Graphs -> Stacked Histogram Creates vertically stacked layers displaying the histograms of the selected data columns in the active table window (one histogram per layer) See the [Panel -> Vertical 2 Layers](#) command for more details.

Panel -> These commands can be used to obtain quickly some classical arrangements of multiple plot.

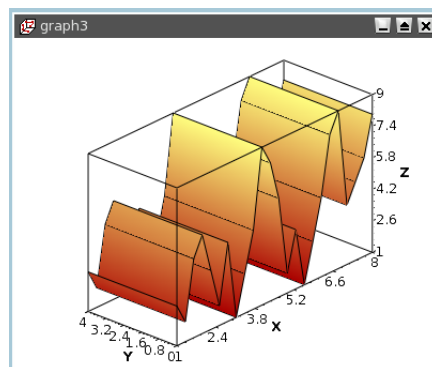
Panel -> Vertical 2 Layers Creates 2 vertically stacked layers displaying the selected data columns in the active table window (one curve per layer).

Panel -> Horizontal 2 Layers Creates 2 horizontally stacked layers displaying the selected data columns in the active table window (one curve per layer).

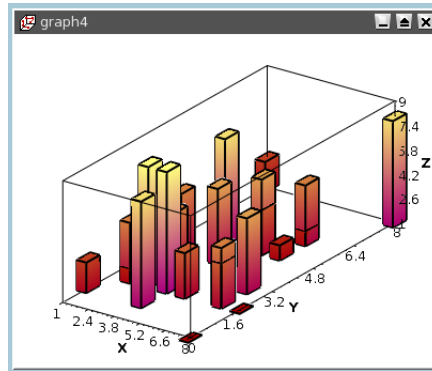
Panel -> 4 Layers Creates 4 layers on a 2x2 grid, displaying the selected data columns in the active table window (one curve per layer).

Panel -> Stacked Layers Creates vertically stacked layers displaying the selected data columns in the active table window (one curve per layer).

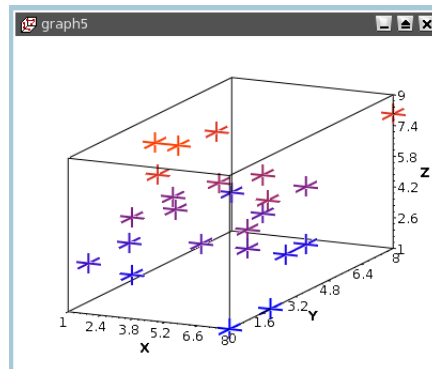
Data -> Plot 3D -> Plot 3D -> Ribbons Makes a 3D plot of the selected data column in the active table window (only one column allowed) using the "Ribbon" style.






Bars Makes a 3D plot of the selected data column in the active table window (only one column allowed) using the "3D Bars" style.

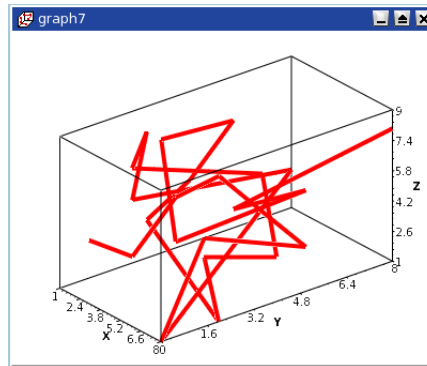


Scatter Makes a 3D plot of the selected data column in the active table window (only one column allowed) using the "3D Dots" style. The 3D point symbol style can be changed via the 3D Plots Settings dialog.



With scatter plots, you can choose the kind of graphic item which is used to plot the data points. The example above is done with cross hairs, but you can also select points or cones. This can be done either with the corresponding icons of the [3d plot toolbar](#) (respectively ,  and  for cross-hairs, dots and cones) or with the [custom-curves dialog](#).

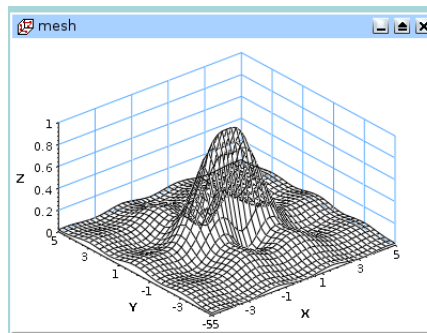
Plot 3D -> Trajectory Makes a 3D plot of the selected data column in the active table window (only one column allowed) using the "3D Line" style. The line width and color may be changed via the 3D Plots Settings dialog.



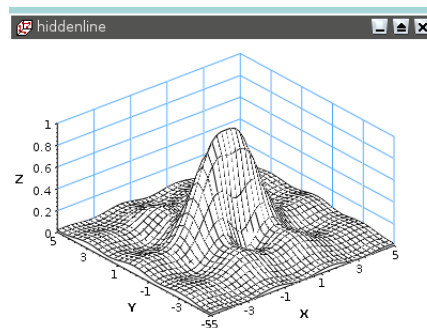
3.6 The Plot 3D menu

This menu is only active when a matrix is selected.

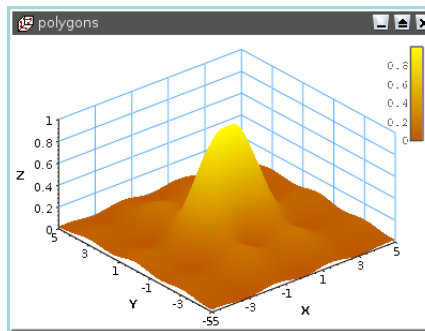
3D Wire Frame Makes a 3D plot of the selected matrix using the "3D mesh" style.



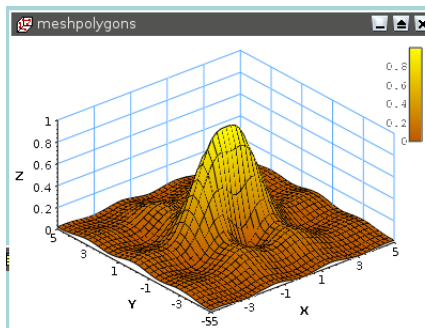
3D Hidden Lines Makes a 3D plot of the matrix using the "3D mesh" style with hidden lines.



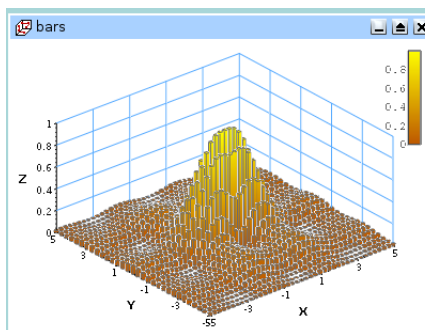
3D Polygons Makes a 3D plot of the matrix using the "3D polygons" style.



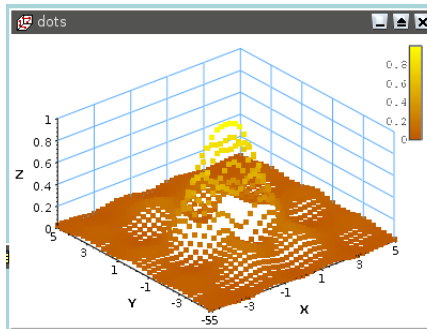
3D Wire Surface Makes a 3D plot of the matrix using the "3D polygons" style with the mesh drawn.



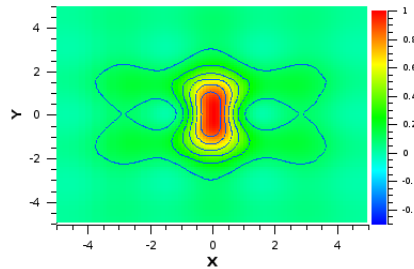
Bars Makes a 3D plot of the selected data column in the active table window (only one column allowed) using the "3D Bars" style.



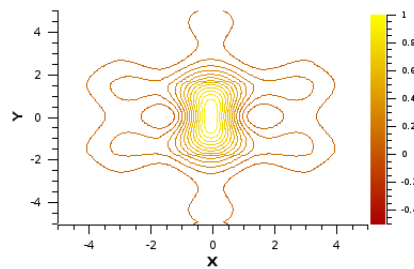
Scatter Makes a 3D plot of the selected data column in the active table window (only one column allowed) using the "3D Dots" style. The 3D point symbol style can be changed via the [3D Plots Settings dialog](#).



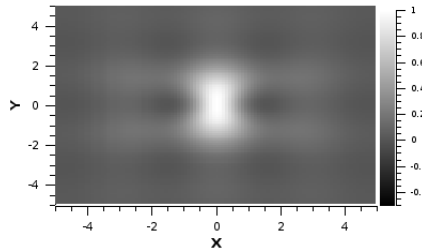
Contour+Color Fill Makes a color map plot of the data in the active matrix window. The contour lines and the colormap settings may be changed by clicking on the plotting area, this will activate the [Contour Options Dialog](#).



Countour Lines Makes a contour plot of the data in the active matrix window. The contour lines and the colormap settings may be changed by clicking on the plotting area, this will activate the [Contour Options Dialog](#).



Gray Scale Map Makes a gray map plot of the data in the active matrix window. The contour lines and the colormap settings may be changed by clicking on the plotting area, this will activate the [Contour Options Dialog](#).



3.7 The Data Menu

This menu is active only when a plot is selected.

Data -> Disable tools When you are using a command which modify the pointer such as the **Data Reader**, this command can be used to exit this special mode, and go back to the normal pointer behaviour.

Data -> Zoom in (Ctrl-+) Switches the active plot layer to the zoom mode. The mouse cursor shape changes to a magnifying lens only inside the active plot canvas. You can select a window in the current plot which will be used as the new plotting window.

Data -> Zoom out (Ctrl--) This command cancel the previous zooming, a history of the zoom is kept so that you can do multiple zoom out commands.

Data -> Rescale To Show All (Ctrl-Shift-R) Rescale the active plot layer after a zoom operation.

Data -> Data Reader (Ctrl-D) Shows a red cross cursor and opens the Data Display toolbar giving easy and fast access to the values of the data points. You can select data points by moving the cursor with the Left and Right arrow keys or faster by clicking on them with the mouse. You can navigate through the curves on the plot layer using the Up and Down arrow keys.

Data -> Select Data Range (Alt-S) Shows two rectangular cursors that can be used for selecting the data range when performing analysis operations. The mouse cursor shape changes to a rectangular target only inside the active plot canvas. The active cursor is red, the other is black. You can move the active cursor with the arrows keys while keeping the Ctrl key pressed or faster by clicking on a curve point. You can change the active cursor using the Left and Right arrow keys. You can navigate through the curves on the plot layer using the Up and Down arrow keys.

Data -> Screen Reader Opens the Data Display toolbar and changes the mouse cursor shape to a small cross target. By keeping the left button pressed and moving the mouse you can view the coordinates of the cursor with respect to the axes of the active plot layer.

Data -> Move Data points (Ctrl-Alt-M) Allows you to modify the position of data points in the active plot layer by simple drag-and-drop. It opens the Data Display toolbar, for a better visualisation of the new coordinates. The changes you make automatically modify the data into the corresponding tables and all the plots depending on those data sets.

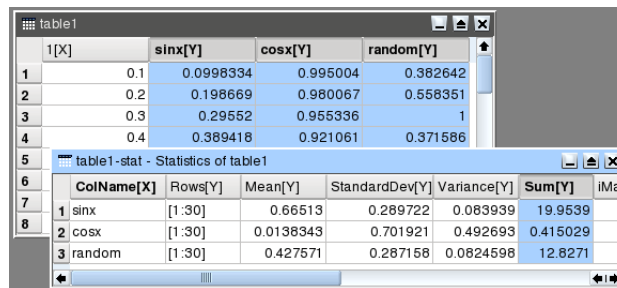
Data -> Remove Bad Data Points (Alt-B) Allows you to remove data points from the active plot layer by double-clicking on them. The coordinates of the points selected for removal are shown in the Data Display toolbar. The changes you make automatically modify the data into the corresponding tables and all the plots depending on those data sets.

3.8 The Analysis Menu

The commands which are available in this menu are not the same if a table or a plot is selected.

3.8.1 Commands for the analysis of data in tables

Statistics on Columns Creates a new table providing basic statistical information about the selected columns in the active table: average, variance, standard deviation, max value, etc...



The screenshot shows a window titled 'table1' containing a table with 5 columns: '1[X]', 'sinx[Y]', 'cosx[Y]', and 'random[Y]'. The first four columns have data rows 1 through 4. Below this table, a 'Statistics of table1' window is open, showing a table with 6 columns: 'ColName[X]', 'Rows[Y]', 'Mean[Y]', 'StandardDev[Y]', 'Variance[Y]', and 'Sum[Y]'. The 'Sum[Y]' column is highlighted in blue. The data in the statistics table is as follows:

| ColName[X] | Rows[Y] | Mean[Y] | StandardDev[Y] | Variance[Y] | Sum[Y] |
|------------|---------|-----------|----------------|-------------|----------|
| 1 sinx | [1:30] | 0.66513 | 0.289722 | 0.083939 | 19.9539 |
| 2 cosx | [1:30] | 0.0138343 | 0.701921 | 0.492693 | 0.415029 |
| 3 random | [1:30] | 0.427671 | 0.287168 | 0.0824598 | 12.8271 |

You can select several columns in one table, one line will be created for each column. You can't select columns in different tables to obtain one single table of statistics.

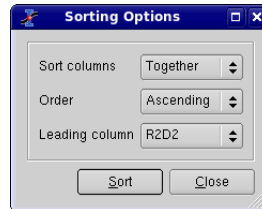
Statistics on Rows Creates a new table providing basic statistical information about the selected rows in the active table: average, variance, standard deviation, max value, etc...

See the [Statistics on Columns](#) command for more details.

Sort Column Sorts the columns selected. If more than one column is selected, you can sort them:

- separately: each column will be sorted in ascending or descending order

- together: the column selected as *leading column* will be sorted in ascending or descending order, and the others column selected will be sorted in order to keep the rows unchanged.



Sort Table This is the same command as [Sort Column](#) but it operate on all columns of the active table.

Normalize Normalizes the columns selected, that is modify the data in order to obtain a range of 0 to 1. All columns selected are normalized separately. This command doesn't create new normalized columns but replace the values of the selected columns.

Normalize -> Columns Normalizes the selected column.

Normalize -> Table Normalizes all the columns of the table, it is not a global normalization of all values of the table: each column is normalized separately.

FFT... Computes a direct or inverse Fast Fourier Transform. The parameters used can be set with the [FFT dialog](#). See the [fft section](#) of the [Analysis chapter](#) for more details.

Correlate Does a cross-correlation of the two columns which are selected. See the [correlate section](#) of the [Analysis chapter](#) for more details.

Convolute Does a convolution of the two columns which are selected. The first one being the response and the second the signal. See the [convolution section](#) of the [Analysis chapter](#) for more details.

Deconvolute Does a deconvolution of the two columns which are selected. The first one being the response and the second the signal. See the [deconvolution section](#) of the [Analysis chapter](#) for more details.

Non Linear Curve Fit... (Ctrl-Y) Opens the [Non-linear Fit](#) dialog, allowing you to choose the curve to fit, the algorithm and the tolerance, the number of iterations to be performed, and to type the analytical function to use, the names of the fitting parameters and their initial guessed values. See the [Non Linear Curve Fit section](#) of the [Analysis chapter](#) for more details.

3.8.2 Commands for the analysis of curves in plots

The following items are enabled only if the active window is a 2D Multilayer Plot Window. If the active plot layer contains more than one curve, and the Data Range Selectors are not enabled, a dialog window will pop-out allowing you to select the curve you want to analyse.

In most of the cases (except for integration), a new red curve is added to the active plot layer and a new table containing the data used to plot this curve is added to the workspace. Useful information about the operation performed will be showed in the Results Log display.

The commands [FFT...](#) and [Non Linear Curve Fit...](#) are presented in the [Table Analysis Menu](#).

Analysis -> Differentiate Creates a new plot displaying the resulting curve of the numerical differentiation. The computation of the derivative is done by centered finite differences.

This command creates a new table which contains one column for X-values and one column for derivatives of Y-values. It also creates a new plot of the derivative.

Analysis -> Integrate... Opens the [Integration dialog](#), allowing to choose the curve to integrate and the integration method.

This command can't be used to obtain a cumulative curve from a selected curve, it can only compute the integral of the data between two limits. The result is given in the [Log Panel](#).

Analysis -> Smooth .

Savitski-Golay This command performs a smoothing of the selected curve with the Savitzky-Golay method. The formula used to smooth the curve defined by the points $y_i=f(x_i)$ is:

$$z_i = \frac{1}{n} \sum_{j=i-n/2}^{j+i/2} f_j y_j$$

The f_i values are computed by fitting the data points to a polynome, they depend on the number of points used for the smoothing of the curve and the order of the polynome. Compared to the moving window average method, the advantage of this smoothing method is that the values of extrema are not truncated. The dialog allows to specify the curve which will be smoothed, the value of the order of the polynome, the number of data points used for the polynomial fit before and after each point and the color used to draw the smoothed curved. A new table will be created to store the data points x_i, z_i .

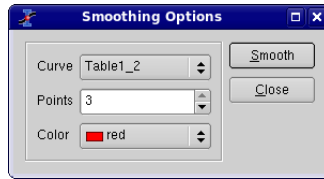


Figure 3.1: The **Smooth -> Moving Window Average...** dialog.

Moving Window Average... This command performs a smoothing of the selected curve with the moving window average method. The formula used to smooth the curve defined by the points $y_i=f(x_i)$ is:

$$z_i = \frac{1}{n} \sum_{j=i-n/2}^{j=i+n/2} y_j$$

The greater the number of points n , the smoother the resulting curve $z_i=f(x_i)$ is. The dialog allows to specify the curve which will be smoothed, the value of n and the color used to draw the smoothed curve. A new table will be created to store the data points x_i , z_i .

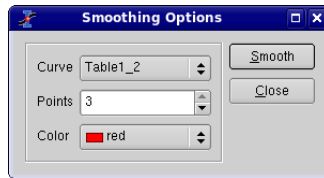


Figure 3.2: The **Smooth -> Moving Window Average...** dialog.

Analysis -> FFT Filter Low Pass... This command allows to filter the high frequencies of a signal. See the [filtering section](#) for more details. A dialog box will be opened in which you can select the curve to filter and the cut-off frequency of the filter.

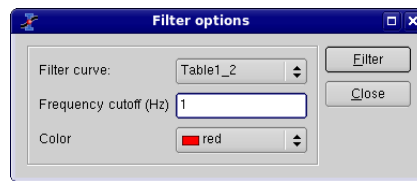


Figure 3.3: The **FFT Filter -> Low Pass...** dialog.

This command creates a new table with the filtered data, and a new curve will be added on the current plot.

High Pass... This command allows to filter the low frequencies of a signal. See the [filtering section](#) for more details. A dialog box will be opened in which you can select the curve to filter and the cut-off frequency of the filter.

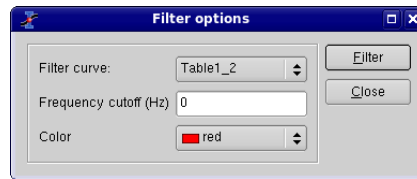


Figure 3.4: The **FFT Filter -> High Pass...** dialog.

This command creates a new table with the filtered data, and a new curve will be added on the current plot.

Band Pass... This command allows to filter the low and high frequencies of a signal. See the [filtering section](#) for more details. A dialog box will be opened in which you can select the curve to filter and the cut-off frequency of the filter.

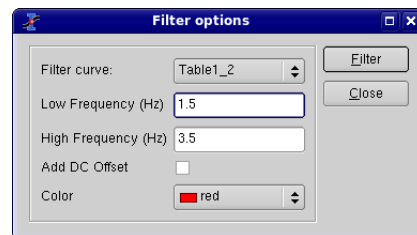


Figure 3.5: The **FFT Filter -> Band Pass...** dialog.

This command creates a new table with the filtered data, and a new curve will be added on the current plot.

Band Block... This command allows to keep the low and high frequencies of a signal. See the [filtering section](#) for more details. A dialog box will be opened in which you can select the curve to filter and the cut-off frequency of the filter.

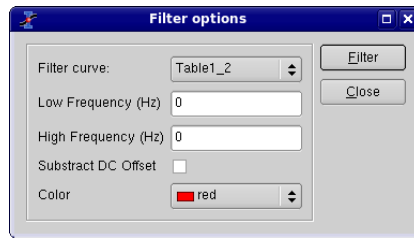


Figure 3.6: The **FFT Filter -> Band Block...** dialog.

This command creates a new table with the filtered data, and a new curve will be added on the current plot.

Analysis -> Interpolate... Performs an interpolation. The curve must have enough data points to compute the interpolated points, if not a warning message will be prompted out.

The methods available to perform the interpolation are *Linear* (the curve must contain at least 3 points), *Cubic Spline* (the curve you analyse must contain at least 4 points, if not a warning message will be prompted out, *Non-rounded Akime spline* (the curve you analyse must contain at least 5 points). See the [Analysis chapter](#) for a comparison of the differents methods.

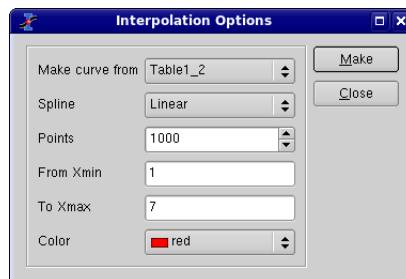


Figure 3.7: The **Interpolate...** dialog.

This command creates a new curve on the current plot, and a new table.

Analysis -> FFT... Performs a [forward or inverse FFT](#) transform of the selected curve. The parameters used can be set with the [FFT dialog](#).

The inverse FFT transform of a forward transform will result in a data set identical to that used for the forward transform.

Analysis -> Fit Linear Performs a [linear fit](#) of the selected curve. The results will be given in the [Log panel](#)

Analysis -> Fit Polynomial... Opens the Polynomial Fit dialog, allowing you to choose the curve to fit, the order of the polynomial function to use, the number of points of the resulting curve and the abscissae limits for the fit.

Analysis -> Fit Exponential Decay

First Order... Opens the Exponential Fit dialog, allowing you to choose the curve to fit and the initial guesses for the fit parameters.

Second Order... Opens a dialog, allowing you to choose the curve to fit and the initial guesses for the fit parameters.

Third Order... Opens a dialog, allowing you to choose the curve to fit and the initial guesses for the fit parameters.

Analysis -> Fit Exponential Growth... Performs an exponential growth fit of the selected curve.

Analysis -> Fit Lorentzian Performs a lorentzian fit of the selected curve. It can be used to obtain a correlation equation of a [bell shaped data set](#).

Analysis -> Fit Gaussian Performs a gaussian fit of the selected curve. It can be used to obtain a correlation equation of a [bell shaped data set](#).

Analysis -> Fit Boltzmann (sigmoidal) Performs a fit to a [boltzmann function](#) of the selected curve. It can be used to obtain a correlation equation of a [S shaped data set](#).

Analysis -> Fit Multi-peak ->Gaussian... Performs a fit to a [sum of N gaussian functions](#) of the selected curve.

Analysis -> Fit Multi-peak -> Lorentzian... Performs a fit to a [sum of N lorentz functions](#) of the selected curve.

3.9 The Table Menu

This menu is only active when a table is selected.

Set Column As These commands are used to define the kind of data which is stored in the different columns of a table.

Set Column As -> X Define the selected column as abscissae for the plots. You can define more than one column as X-values in a tables, they will be referenced as X1, X2, etc.

Set Column As -> Y In the case of 2D plots, this command defines the selected column as Y-values for the plots. In the case of 3D plots, Y columns can be used as the second abscissae.

Set Column As -> Z In the case of 3D plots, Z columns will be used as plotted values.

Set Column As -> X error Define the selected column for use as error bars width for abscissae.

Set Column As -> Y error Define the selected column for use as error bars for Y-values.

Set Column As -> None The selected column can be used in different ways in several plots (as X values, Y values, etc).

Column Options... This command is used to define the global parameters of each column such as numeric format, column name, etc. See the [corresponding dialog box section](#) for more details.

Set Column Values... This command is used to fill the selected column with the values resulting from a mathematical formula. See the [corresponding dialog box section](#) for more details.

Recalculate When you fill a column (named for example 'C1') with the results of a formula (by using the [Set Column Values... command](#)), the values of the column are calculated only once when you define the formula. If your formula depends on values of another column (name for example 'C2'), the values of 'C1' are not updated if you modify the values in 'C2'. This command is used to recalculate the values of the selected column.

Fill column with This command is used to fill the selected column with special values:

Fill Column With -> Row Numbers The filling is done with the number of the corresponding rows.

Fill Column With -> Random Numbers The filling is done with random values between 0 and 1.

Clear Removes all the values of the selected column

Add Column Adds a new column in the table. Whatever the selected column, the new one will be inserted at the right of the table after the left column.

Columns Allows to define the number of columns in the table. Be carefull if you decrease the number of columns in a table, a number of columns will be removed and the data will be lost.

Rows Allows to define the number of rows in the table. Be carefull if you decrease the number of rows in a table, a number of rows will be removed and the data will be lost.

Go to Row... Defines the active line in the selected table.

Convert to Matrix This command is used to convert a table into a matrix. It is mainly used to import data from files: the first step import data in a table, and the second one is the conversion of the table in a matrix.

3.10 The Matrix Menu

This menu is only active when a matrix is selected.

Set Properties... This command opens a [dialog window](#) which is used to specify some view parameters of the matrix (cell width, format of numbers).

Set Dimensions... This command opens a [dialog window](#) which is used to specify the size of a matrix. It can also be used to specify the X and Y ranges which will be used as axis ranges for a 3D-plot of the matrix data.

Set Values... This command opens a [dialog window](#) which is used to fill in a matrix with the result of a function $z=f(i,j)$ in which i and j stand for the row and column numbers.

Transpose Transpose the selected matrix.

Invert Inverse the selected matrix.

Determinant Compute the determinant of the selected matrix.

Convert to Spreadsheet Convert the selected matrix in a table.

3.11 The Format Menu

This menu is only active when a plot is selected.

Plot... In the case of a classical 2D plot, opens the [format plot dialog](#) with the general plot options tab selected. It allows to customize the line styles and colors of plot frame, etc.

In the case of a surface plot, this command opens the [surface plot options](#) with the general plot options tab selected. In this case the aspect ratio of the plot can also be modified.

Curves... Opens the [Custom Curves dialog](#). It allows to customize the line style and colors used to draw curves.

If the selected plot is a surface plot, this menu item is not showed.

Scales... Opens the [format plot dialog](#) with the scales tab selected. It allows to customize the ranges of the different axes. It must be reminded that any modification in the table or in the plotted curves will result in a reset of these scales to the default values.

In the case of a surface plot, this command opens the [surface plot options](#) with the scales options tab selected.

Axes... Opens the [format plot dialog](#) with the axes tab selected. It allows to customize the settings for the different axes such as the size and color of axes and ticks, the label of the axes, etc.

In the case of a surface plot, this command opens the [surface plot options](#) with the axis options tab selected.

Grid... Opens the [format plot dialog](#) with the grid tab selected. It allows to add and customize grid lines on the different axes.

If the selected plot is a surface plot, this menu item is not showed.

Title... Opens a [text options dialog](#), allowing you to modify the title of the plot and its properties (color, font, alignment).

In the case of a surface plot, this command opens the [surface plot options](#) with the title options tab selected.

3.12 The Window Menu

Additionally to the items listed bellow, this menu will also display a list with the first ten windows created in the workspace. These windows can be made active or can be shown if they are hidden, by selecting their name from the list. If your project contains more then ten windows, you must use the Project explorer in order to perform these operations.

Cascade Arranges the visible windows in the project in a cascading style.

Tile Tiles the visible windows in the project.

Next (F5) Makes the next visible window in the workspace stack the active window.

Previous (F6) Makes the previous visible window in the workspace stack the active window.

Rename Window Opens a dialog allowing to change the title of the active window.

Duplicate Clonates the active window.


Window Geometry... Opens a dialog allowing to change the size and the position of the active window. The size of the plot will be adapted to the new window size.

Hide Window Hides the active window. A hidden window can be made visible again via the Project explorer.


Close Window (Ctrl-W) Closes the active window. You will be prompted out a question dialog asking you to confirm the operation, if you checked this option in the Preferences dialog ("Confirmations" tab).

3.13 Customization of 3D plots


These commands are not available through any menu nor by any keyboard shortcut. They can be accessed through the [3D toolbar](#).

Frame This command can be accessed by a click on the 


Draws only the three axis on the active 3D plot.

Box This command can be accessed by a click on the 


Draws the three axis on the active 3D plot, and a box around it.

No axes This command can be accessed by a click on the 


Doesn't draw the three axis nor the box on the active 3D plot.

Front Grid This command can be accessed by a click on the 


Draws a grid on the front panel of the active 3D plot. The position of this grid is the plan defined by $y=y_{\min}$.

Back Grid This command can be accessed by a click on the 


Draws a grid on the back panel of the active 3D plot. The position of this grid is the plan defined by $y=y_{\max}$.

Left Grid This command can be accessed by a click on the 


Draws a grid on the left panel of the active 3D plot. The position of this grid is the plan defined by $x=x_{\min}$.

Right Grid This command can be accessed by a click on the 


Draws a grid on the right panel of the active 3D plot. The position of this grid is the plan defined by $x=x_{\max}$.

Ceiling Grid This command can be accessed by a click on the 


Draws a grid on the top panel of the active 3D plot. The position of this grid is the plan defined by $z=z_{\max}$.

Floor Grid This command can be accessed by a click on the 


Draws a grid on the floor panel of the active 3D plot. The position of this grid is the plan defined by $z=z_{\min}$.

Enable perspective This command can be accessed by a click on the 


Enables/Disables the 3D perspective mode.

Reset rotation This command can be accessed by a click on the 


Resets the rotation of the 3D plot to the default values.

Fit frame to window This command can be accessed by a click on the 


Finds the best layout of the 3D plot fitting the window size. It readjusts the length of the axis ticks to a default value.

Bars Style This command can be accessed by a click on the 


If the active 3D plot is a **3D histogram**, this command is used to modify the style of the bars.

Dots This command can be accessed by a click on the 


If the active 3D plot is a **3D scatter**, this command is used to modify the style of the data points to dots.

Cones This command can be accessed by a click on the 


If the active 3D plot is a **3D scatter**, this command is used to modify the style of the data points to cones. It is then possible to modify the drawing parameters of the cones by double clicking on the plotting area.

Cross Hairs This command can be accessed by a click on the 


If the active 3D plot is a **3D scatter**, this command is used to modify the style of the data points to cross-hairs. It is then possible to modify the drawing parameters of the crosses by double clicking on the plotting area.

3D Wire Frame This command can be accessed by a click on the 


If the active 3D plot is a 3D surface, this command is used to modify the style of the surface to a simple wireframe.

3D Hidden Lines This command can be accessed by a click on the 


If the active 3D plot is a 3D surface, this command is used to modify the style of the surface to a wireframe. A computation of the hidden line is done.

3D Polygons This command can be accessed by a click on the 


If the active 3D plot is a 3D surface, this command is used to modify the style of the surface to polygons.

3D Wire Surface This command can be accessed by a click on the 


If the active 3D plot is a 3D surface, this command is used to modify the style of the surface to polygons with a mesh.

Floor Data Projection This command can be accessed by a click on the 


If the active 3D plot is a 3D surface, this command is used to add a filled area projection of the surface on the floor of the plot.

Floor Isolines This command can be accessed by a click on the 

If the active 3D plot is a 3D surface, this command is used to add an isoline.

Empty Floor This command can be accessed by a click on the 

If the active 3D plot is a 3D surface, this command is used to remove any projection from the floor.

Animation This command can be accessed by a click on the 

Enables/disables animation.

Chapter 4

The Toolbars

All toolbars can be moved and docked to a more convenient location (left, right or bottom sides of the application window) or on the desktop (outside the main window) by drag-and-drop, using their left side handle. The toolbars are automatically enabled/disabled depending on the currently active window: for example if the current window is a table, the Table toolbar will be enabled and all the other toolbars will be automatically disabled.

The same approach is used for showing/hidding the toolbars: if there are no more visible tables in the workspace, the Table toolbar will be automatically hidden and will be shown again when the users adds a new table into the project. A toolbar can be manually shown/hidden by the user, at any time, by right-clicking on the main window menu area and checking/unchecking the corresponding box in the pop-up menu.

4.1 The Edit Toolbar



Figure 4.1: The SciDAVis Edit Toolbar

4.2 The File Toolbar

The *File Toolbar* allows to access commands mainly from the [File menu](#). Refer to this section for a more complete description of these commands.



Figure 4.2: The SciDAVis File Toolbar







| Icon | Command | Key | Description |
|---|--|--------|--|
|  | Undo command | Ctrl-Z | Undo the last command, this feature doesn't work for plot modifications. |
|  | Redo command | Ctrl-R | Redo the last command, this feature doesn't work for plot modifications. |
|  | Cut Selection command | Ctrl-X | Cut the current selection. |
|  | Copy Selection command | Ctrl-C | Copy the current selection. |
|  | Paste Selection command | Ctrl-V | Paste the current selection. |
|  | Delete Selection command | | Delete the current selection. |

Table 4.1: Edit toolbar commands.

4.3 The Plot Toolbar.

This toolbar is only active when a plot window is selected. It allows the quick access to the commands of the [Graph menu](#) and of the [Data menu](#) which are used for the modification of the plots and of the data points of the plots.



Figure 4.3: The SciDAVis Plot Toolbar

4.4 The Table Toolbar.

This toolbar allows a quick access to the commands of the [Plot Menu](#) used to plot the datas of the table.

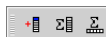


Figure 4.4: The SciDAVis Table Toolbar


















| Icon | Command | Key | Description |
|---|---|------------|--|
|  | New -> New Project command | Ctrl-N | Create a new project. |
|  | New -> New Table command | Ctrl-T | Create a new table. |
|  | New -> New Matrix command | | Create a new matrix. |
|  | New -> New Note command | | Create a new note window. |
|  | New -> New Graph command | Ctrl-G | Create a new empty 2D plot. |
|  | New -> New Function Plot command | Ctrl-F | Creates a new plot based on a function $Y=f(X)$. |
|  | New -> New Surface 3D Plot command | Ctrl-Alt-Z | Creates a new 3D plot based on a function $Z=f(X,Y)$. |
|  | Open command | Ctrl-O | Opens an existing SciDAVis project file. |
|  | Open Template command | | Opens an existing template SciDAVis project file. |
|  | Save Project command | Ctrl-S | Saves the current project. |
|  | Save as Template command | | Saves the current project as a template. |
|  | Import ASCII -> Single File... command | | Imports an ASCII file into one single table. |
|  | Import ASCII -> Multiple Files... command | | Imports an ASCII file into multiple tables. |
|  | Duplicate command | | Clonates the active window. |
|  | Print command | Ctrl-P | Print the active window. |
|  | Project Explorer command | Ctrl-E | Show or hide the project explorer. |
|  | Results log command | | Show or hide the results window. |

Table 4.2: File toolbar commands.




















| Icon | Command | Key | Description |
|---|------------------------|------------|---|
|  | Add Layer | Alt-L | Adds a new layer to the active plot window. |
|  | Arrange Layers | Alt-A | Arranges the different layers of the active plot window. |
|  | Add/Remove Curves... | Alt-C | Adds or removes curves to the active plot window. |
|  | Add Error Bars... | Ctrl-B | Adds error bars to a curve of the active plot window. |
|  | Add Function... | Ctrl-Alt-F | Adds a curve based on a function to the active plot window. |
|  | New Legend | Ctrl-L | Adds a new legend to the active plot window. |
|  | Disable tools | | Comes back to the normal pointer mode, this is useful when you have select other modes of the plot window such as the data reader . |
|  | Zoom in | Ctrl+ | Switches the active plot layer to the zoom mode. |
|  | Zoom out | Ctrl-- | Switches the active plot layer to the zoom mode. |
|  | Data Reader | Ctrl-D | Switches the data display mode. |
|  | Select Data Range | Alt-S | Switches the active plot to the Select Data Range mode. |
|  | Screen Reader | | Switches the active plot layer to the Screen Reader mode. |
|  | Move Data points | Ctrl-Alt-M | Allows to move data points on the active plot. |
|  | Remove Bad Data Points | Alt-B | Allows to remove data points on the active plot. |
|  | Add Text command | Alt-T | Add a new text element in the active plot. |
|  | Draw Line command | Ctrl-Alt-L | Add a new line on the active plot. |
|  | Draw Arrow command | Ctrl-Alt-A | Add a new arrow on the active plot. |
|  | Add Time Stamp command | Ctrl-Alt-T | Add a time/date label on the active plot. |
|  | Add Image command | Alt-I | Insert a new image in the active plot. |

Table 4.3: Plot toolbar commands

















| Icon | Command | Key | Description |
|---|---|-----|--|
|  | plot -> Line | | plot with the line style. |
|  | plot -> Scatter | | plot with the scatter style. |
|  | plot -> Line+Symbol | | plot with the line+symbol style. |
|  | plot -> Columns | | plot with the columns style. |
|  | plot -> Rows | | plot with the rows style. |
|  | plot -> Area | | plot with the area style. |
|  | plot -> Pie | | plot with the pie style. |
|  | plot -> Statistical Graphs -> Histogram | | plot with the histogram style. |
|  | plot -> Vectors XYYX | | plot with the vector style. |
|  | plot -> Plot 3D -> Ribbons | | plot with the 3D ribbons style. |
|  | plot -> Bars | | plot with the 3D bars style. |
|  | plot -> Scatter | | plot with the 3D scatter style. |
|  | plot -> Plot 3D -> Trajectory | | plot with the trajectory style. |
|  | plot -> Add Column | | add a new column to the table |
|  | plot -> Statistics on Columns | | compute statistical parameters on selected columns |
|  | plot -> Statistics on Rows | | compute statistical parameters on selected row |

Table 4.4: Table toolbar commands.

4.5 The Plot 3D Toolbar.



Figure 4.5: The SciDAVis Plot 3D Toolbar

















| Icon | Command | Key | Description |
|---|-------------------------------|-----|---|
|  | plot -> Frame | | Draw only the three axes. |
|  | plot -> Box | | Draw the three axes and the 3D box around the plot. |
|  | plot -> No axes | | Doesn't draw the axes nor the box. |
|  | plot -> Front Grid | | Draw a grid on the front panel. |
|  | plot -> Back Grid | | Draw a grid on the back panel. |
|  | plot -> Left Grid | | Draw a grid on the left panel. |
|  | plot -> Right Grid | | Draw a grid on the right panel. |
|  | plot -> Ceiling Grid | | Draw a grid on the top panel. |
|  | plot -> Floor Grid | | Draw a grid on the bottom panel. |
|  | plot -> Floor Grid | | Draw a grid on the bottom panel. |
|  | plot -> Enable perspective | | Enables/Disables the 3D perspective mode. |
|  | plot -> Reset rotation | | Resets the rotation of the 3D plot to the default values. |
|  | plot -> Fit frame to window | | Finds the best layout of the 3D plot fitting the window size. It readjusts the length of the axis ticks to a default value. |
|  | plot -> Bars Style | | Changes the styles of the bars. |
|  | plot -> Dots | | Draw the 3D scatter points with the dot style. |
|  | plot -> Cones | | Draw the 3D scatter points with the cone style. |
|  | plot -> Cross Hairs | | Draw the 3D scatter points with the cross-hairs style. |
|  | plot -> 3D Wire Frame | | Draw a surface with the wireframe style. |
|  | plot -> 3D Hidden Lines | | Draw a surface with the mesh style (with hidden lines). |
|  | plot -> 3D Polygons | | Draw a surface with the polygons style. |
|  | plot -> 3D Wire Surface | | Draw a surface with the mesh+polygons style. |
|  | plot -> Floor Data Projection | | Draw a projection of the plot on the floor. |
|  | plot -> Floor Isolines | | Draw an isolines projection on the floor. |
|  | plot -> Empty Floor | | Draw an empty floor. |
|  | plot -> Animation | | Enables/Disables animation. |

Table 4.5: 3D Plot toolbar commands.

Chapter 5

The Dialogs

5.1 Add Error bars

This dialog is activated by selecting the [Add Error Bars...](#) command from the [Graph menu](#).

This command is used to plot X and/or Y error bars around the data points.

It must be taken care that the "add" button add the errors bars, and so do the "OK" button. Then, you should close the dialog with cancel if you have clicked on the "add" button.

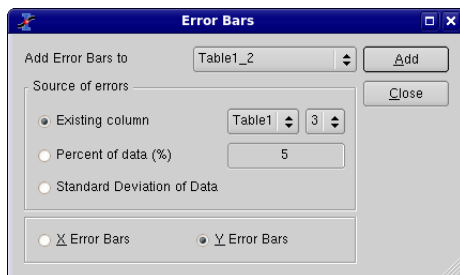


Figure 5.1: The **Add Error Bars...** dialog.

There are three ways to specify the size of the bar:

A column of the table In this case, the values of the selected column are used to compute the error bars. if V is the value of the data point, and E the value of the errorbar column, the size of the bars will be $V-E$ to $V+E$.

A percentage of the values if E is the percentage selected, the size of the bars will be $V(1-E/100)$ to $V(1+E/100)$. It must be noticed that, in addition to the errorbars on the plot, this command will create a new column in the active table with can

be used in the way as with the previous option. This column can be modified like any other one.

The standard deviation of the values the standard deviation of the values. This has a meaning only if the data are centered around an average value. Like with the previous option, a new column will be created in the active table.

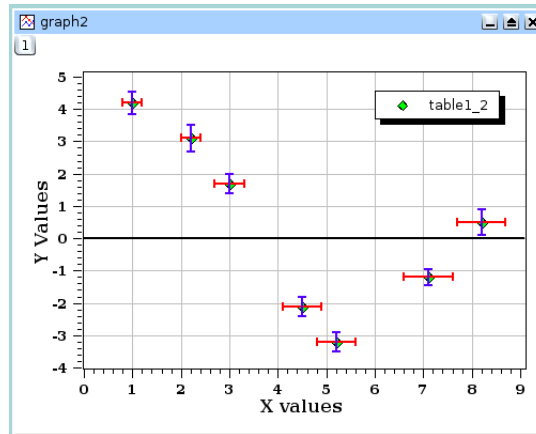


Figure 5.2: A plot with X and Y Error Bars.

5.2 Add Function

This dialog box is used to add a function curve to the active plot. The function can be built with the common operators: * + / - and ^ for the power. The intrinsic functions available are listed in the [appendix](#).

The most common way to define a function is the classical cartesian coordinate definition $y=f(x)$, this is the default option. The two following parameters allow to select the x range used for the plot, and the last one is used for the number of data points that are computed in the X-range.

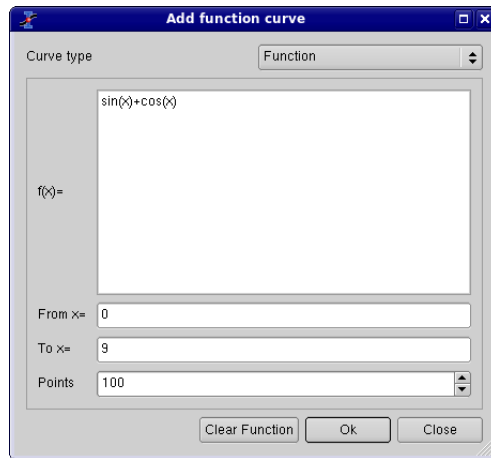


Figure 5.3: The **Add Function...** dialog box: cartesian coordinates.

The functions can also be defined in a parametric definition: if t is the parameter, the (x,y) data points are computed by $x=f(t)$ and $y=g(t)$.

The first parameter is the name of the parametric variable (here t) followed by the range, the definition of the two functions and the number of data points.

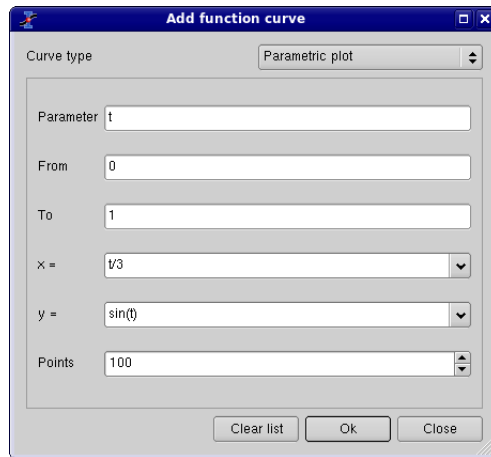


Figure 5.4: The **Add Function...** dialog box: parametric coordinates.

The last way is the polar definition of the function: if t is the parameter, the radius r and the angle θ are computed by $r=f(t)$ and $\theta=g(t)$. Then the (x,y) data points are computed by $x=r*\cos(\theta)$ and $y=r*\sin(\theta)$.

The first parameter is the name of the parametric variable (here t) followed by the

range, the definition of the two functions and the number of data points. The angle is defined in radians, and the constant value π can be used: it is possible to use 3π to define the parameter range.

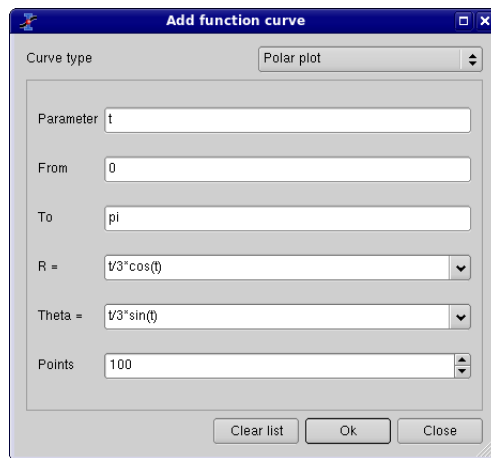


Figure 5.5: The **Add Function...** dialog box: polar coordinates.

5.3 Add Layer

This dialog is opened when you want to add a new layer on the active plot. If you select *Guess*, SciDAVis will divide the window in two columns and put the new layer on the right. If you choose *Top-Left Corner*, SciDAVis will create a new layer with the maximum possible size over the existing layer, this layer contains an empty plot. You can then modify the size and position of each layer by selecting it with the layer number buttons and selecting the *Layer Geometry* command from the context menu.



Figure 5.6: The **Add Layer** dialog box.

5.4 Add/Remove curves.

This dialog is activated by selecting the command [Add/Remove Curves...](#) from the [Graph Menu](#).

The left window shows the columns which are available for plotting in the different tables of the project, and the right window gives the list of the curves already plotted. In the case presented below, there are two tables in which the **Add/Remove Curves...** dialog box allows to select columns. If you use this dialog box to add a column, the X column will be the one define as X in the corresponding table.

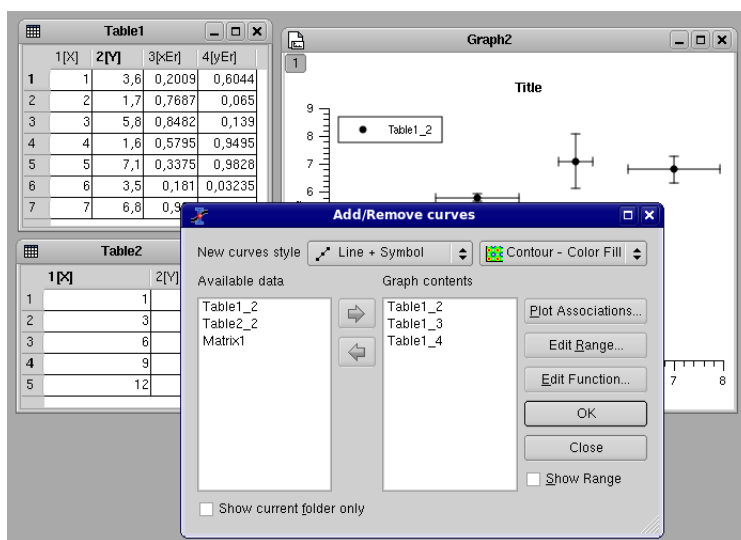


Figure 5.7: The **Add/Remove Curves...** dialog box.

In this dialog box, if you select one curve of the plot in the right window, you can change the columns used for X and Y with the *Plot Association* button. In any case, you can't mix the X values of one table with the Y values of another one. If you want to do this, you have to copy the columns in the same table.

If the curve selected is a function, you can modify it. Refer to the [Add Function... dialog box](#) for more details on functions editing.

5.5 Add Text

This dialog box is opened when you use the [Add Text command](#) command from the [Graph menu](#). If you choose the *On new layer* option, the text will be inserted as a new layer which has the size and the position of the text. You can then modify the size and position of this layer with the *layer Geometry* (see the [Add Layer](#) for details). Beware that in this case, all text which is not in the layer will be clipped, therefore, you need to modify the layer to modify the position of the text. If you choose the *On Active layer* option, the text will be inserted in the selected layer, and its position can be modified directly with the mouse inside this layer.

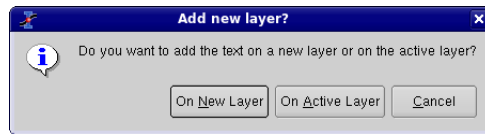


Figure 5.8: The **Add Text** dialog box.

5.6 Arrange Layers

This dialog is activated by selecting the command **Arrange Layers** from the [Graph Menu](#) or by the key code Alt-A.

It allows to modify the geometrical arrangement of the plots which are already present in the active window. You can also add new layers or remove existing ones.

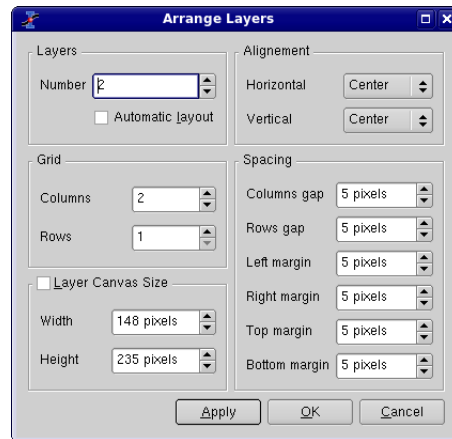


Figure 5.9: The **Arrange Layers** dialog

The *Arrange Layers* dialog is used to modify the geometrical arrangement of the plots. You can specify the numbers of rows and columns which will define a table of plots. As pointed out above, you can also add or remove layers with this dialog, using the "Number of Layers" box.

With the default setting, SciDAVis computes the size of the layers from the size of the window. If you check the *Layer Canvas Size*, you can set the size of the layers and SciDAVis will modify the size of the window.

The two right zones allow to set the alignment of the layers in the window, and the margins between the layer borders and the window limits.

If you do some modifications on your plot, the alignment of the different axis may not be conserved. You can exec again the **Arrange Layers** to re-arrange your plot.

5.7 Add Arrow

This dialog allows to modify a line or an arrow which has been created by the command **Draw Arrow** from the [Graph Menu](#) or with Ctrl-Alt-A. One can also open it with a double click on an arrow or a line, or by selecting an arrow or a line and selecting *Properties...* with the right button of the mouse.

The first tab allows to change the color, the line type and the line width. This last parameter is set in pixels. It is possible to define a default style for all the new lines by pressing the *Set Default* button.

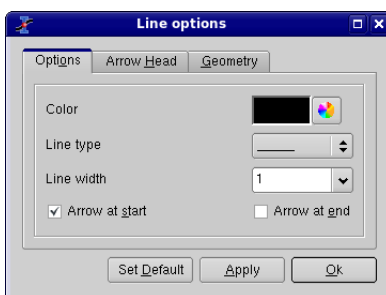


Figure 5.10: The *Arrow options* dialog: first tab

The *Arrow head* tab is used to modify the shape of the head of the arrow. The length is set in pixels and the angle is in degrees. It is also possible to define a default style for the arrow heads using the same *Set Default* button.

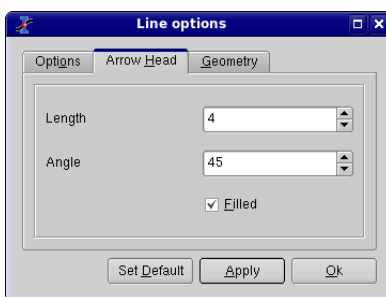


Figure 5.11: The *Arrow options* dialog: second tab

The *Geometry* tab allows to specify the start and end points of the line/arrow. The coordinates can be set as a function of the scales values displayed on the left axis (Y) and on the bottom axis (X) or in pixels, by choosing the desired method from the *Unit* pull-down list. The pixel coordinates are relative to the top-left corner of the layer which contains the line.

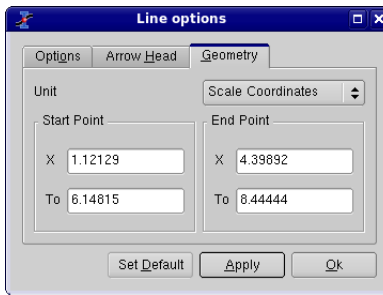


Figure 5.12: The *Geometry* dialog: third tab

5.8 Column Options

This dialog is activated by selecting the command [Column Options...](#) from the [Table Menu](#). At least one column must be selected.

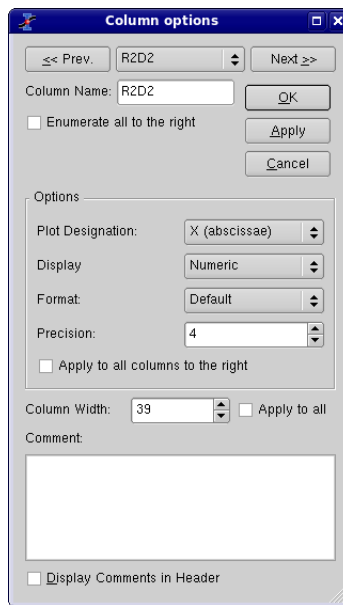


Figure 5.13: The **Column Options...** dialog.

The checkbox *Enumerate all to the right* can be used to build the name of all the columns which are at the right of the selected one. If the name of the selected column is "xyz", this column and the following ones will be renamed to "xyz1", "xyz2", and so on.

The buttons "<<" and ">>" are used to change the selected column. The highlighting of the column in the table behind the dialog box will change indicating that a new column was selected. The column to which the formatting commands are applied is the one whose name appear in the "Column Name" box.

The *Plot Designation* selector is used to define the columns which are used as X, Y or Z values or as error bars. In a table you can select several columns as X, in this case in the column label they will be indicated as [X1], [X2], etc... and their corresponding Y columns will be indicated as [Y1], [Y2], etc...

5.9 Contour Curves Options

This dialog is activated by clicking on a contour curve (or on the plotting area) when a 3D plot has been created from a matrix with one of the following commands of the **Plot3d** menu: **Contour+Color Fill** command, **Countour Lines** command or **Gray Scale** **Map** command.

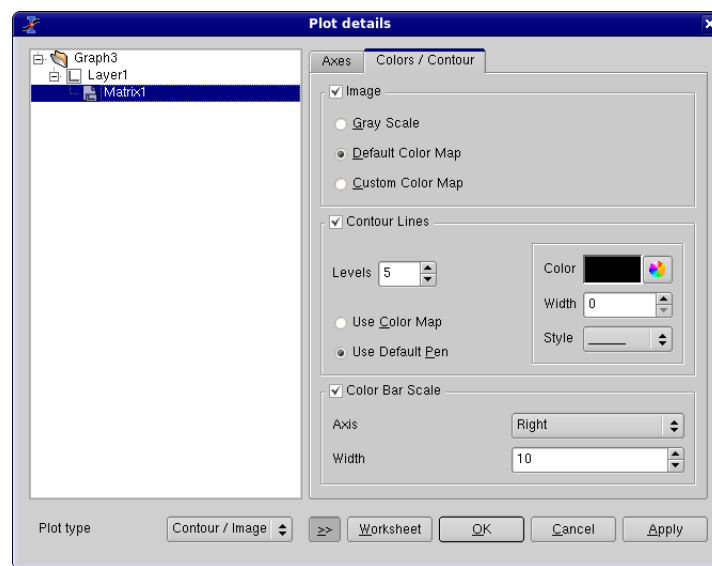
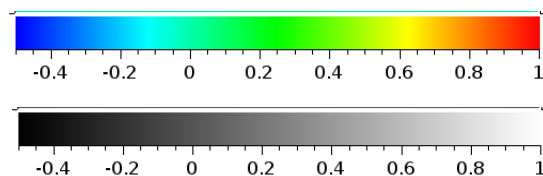
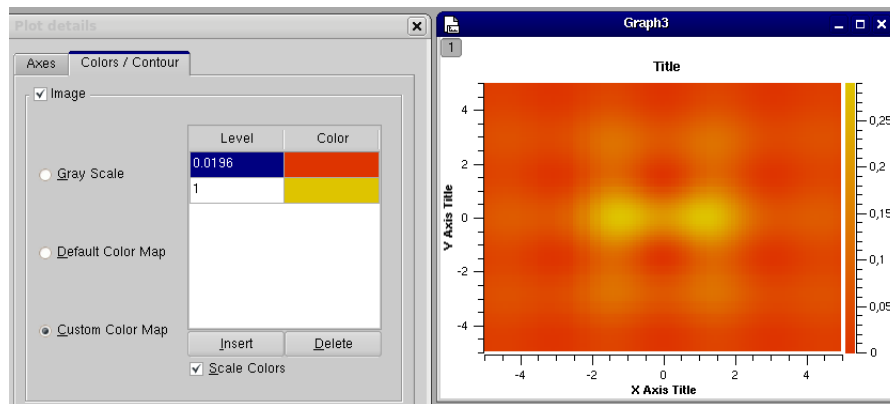


Figure 5.14: The Contour Options dialog.

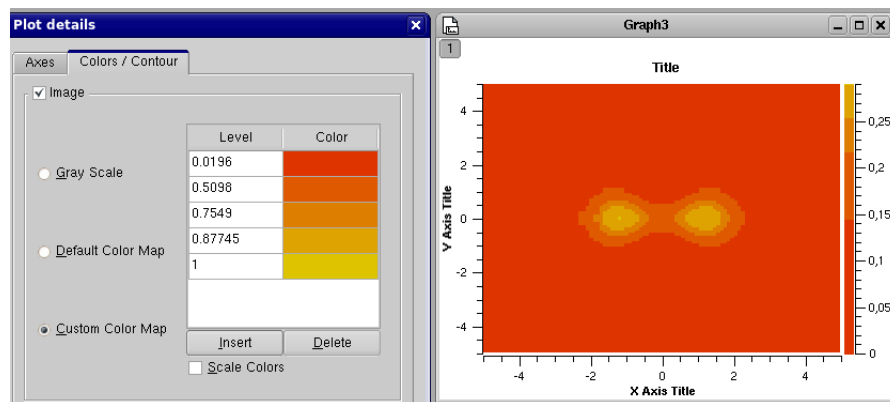
The first group of settings *Image* is checked if you want to have a color or gray level filling of the contour plot. The default gray and color maps are the following:



You can customize this colormap by checking the *Custom Colors* box. A table with a set of numbers (the Z levels) and the corresponding colors is presented. You can then add or delete new levels for the definition of the colormap, and modify the corresponding Z levels. You are not allowed to modify the first and last levels, which are set to the minimum and maximum Z values. Beware that this is only the definition of the colormap, it won't change the number of contour lines of your plot. An example of classical custom colormap is given here:



If you want to obtain discrete colors for each level, you must uncheck the *Scale Colors* checkbox. In this case you must define enough levels in your colormap.



The second group of settings is used to customize the contour lines. You can select the number of lines and their color. If you check *Use Default Pen*, the color of the line will follow the settings defined in the group at the left of the checkbox. If you check *Use Color Map*, the lines will be colored as a function of the Z levels following the colormap defined in the *image* setting group.

The last group of settings must be checked if you want to have a bar scale on your plot. You can then define its position and width.

5.10 Custom Curves

This dialog is activated by selecting the command **Plot...** from the **Format Menu**. It is also activated by a double click on the plot. If there are more than one layer in the window, SciDAVis will select the layer which contain the plot under the mouse pointer.

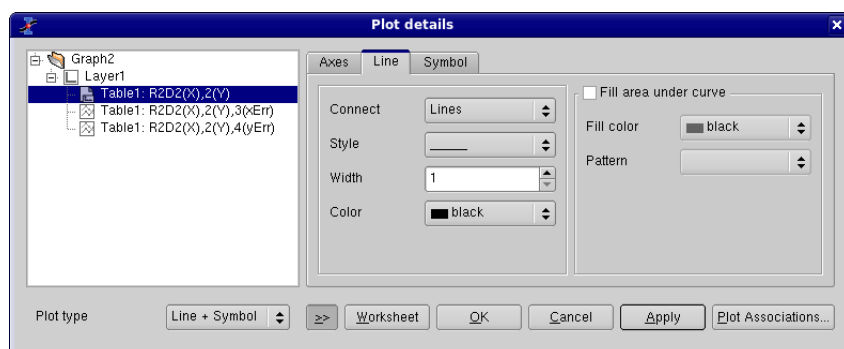


Figure 5.15: The Custom Curves Dialog: Line formatting.

The right part of the dialog box contains several tabs which depend on the kind of plot that you are using, they are described in the following subsections. The left part of the dialog window shows the curves which are plotted in the active layer. All the modifications will be done on the selected curve. You can change the columns which are used by clicking on the *Plot Associations...* button. This will open a dialog which can be used to select the columns of the table which are used as X and Y values.

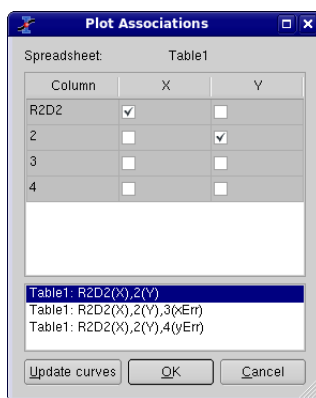


Figure 5.16: The Custom Curves Dialog: Plot Associations.

The button *Worksheet* can be used to access to the table which contains the columns selected.

5.10.1 Custom curves for lines and scatter plots

This dialog box is activated for plots drawn with *symbols*, *line+symbols*, *lines*, *vertical drop lines*, *steps* and *splines*.

The first tab of the right part of the dialog window allows to modify the style of the line (color, line style, thickness). The connect button allows to change the style which is used to draw the selected curve (steps, droplines, etc). See the [Plot menu](#) to see the different types of plot available.

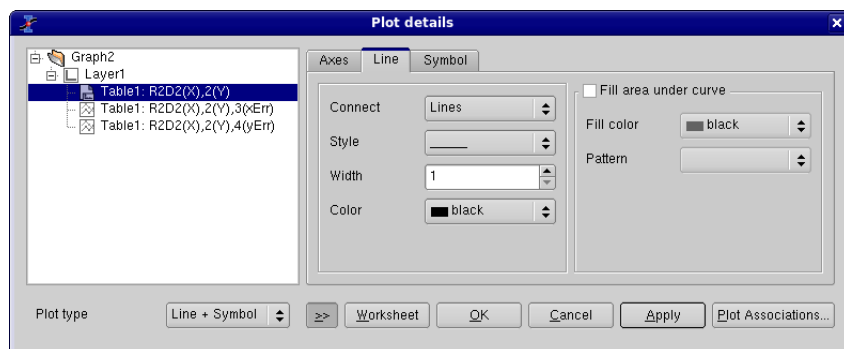


Figure 5.17: The Custom Curves Dialog: Line formatting.

A second tab can be activated to select the symbol, and to modify the size, the color and the filling color of the symbols.

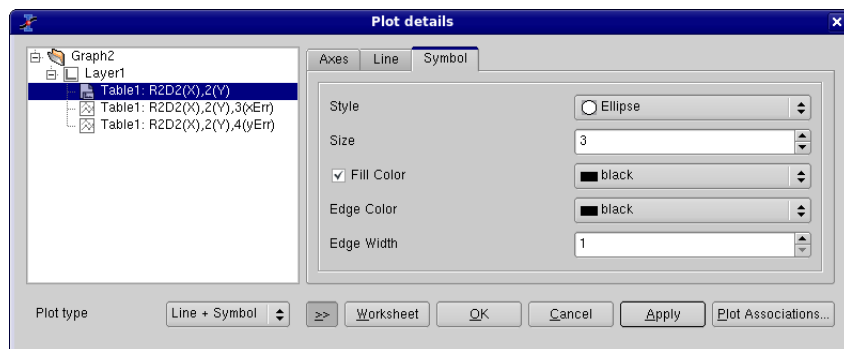


Figure 5.18: The Custom Curves Dialog: Symbol formatting.

5.10.2 Custom curves for pie plots

These commands are available for [pie plots](#). The first tab allows the customization of the pie segments. The left fields are used to modify the border which is drawn round each segment: color, type and width of line. The default is no border (line width = 0).

The right fields are used to define the filling of the plots. The color button defines the one used for the first segment, then the others segments will have colors which follow the order defined in the list. The default value for this field is black, so segment 2, 3, etc will be red, green, etc.

The pattern will be used for all segments of the pie, the default value is solid filling. The last field defines the size of the pie in pixels.

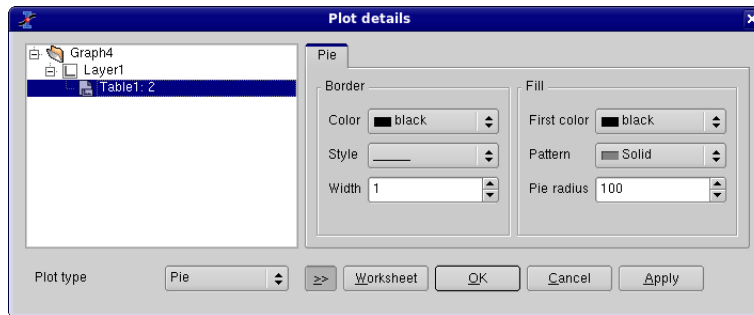


Figure 5.19: The Custom Curves Dialog for pies: pie segment formatting.

5.10.3 Custom curves for box plots

X

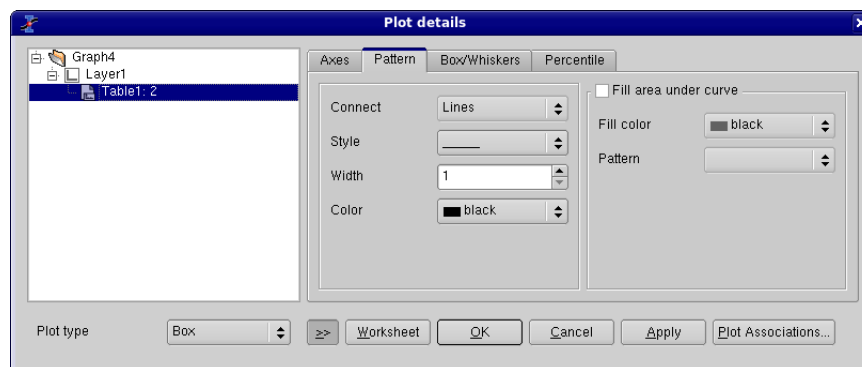


Figure 5.20: The Custom Curves Dialog for box: pattern formatting.

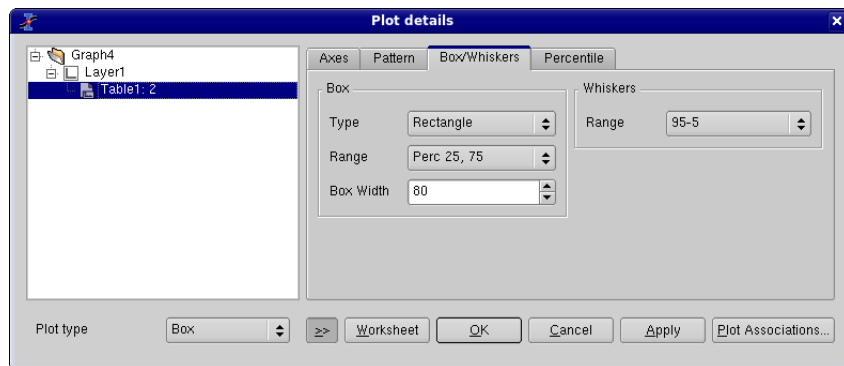


Figure 5.21: The Custom Curves Dialog for box: whiskers formatting.

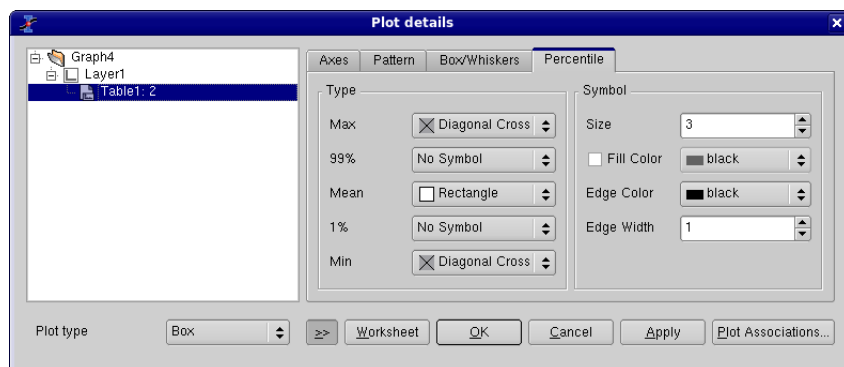


Figure 5.22: The Custom Curves Dialog for box: percentile formatting.

5.10.4 Custom curves for pie histogram

x

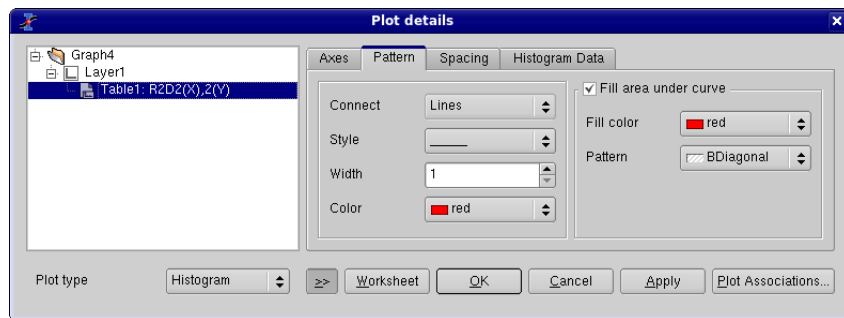


Figure 5.23: The Custom Curves Dialog for histogram: pattern formatting.

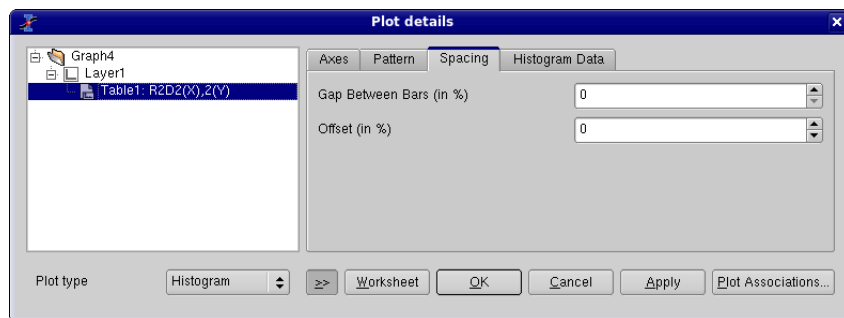


Figure 5.24: The Custom Curves Dialog for histogram: spacing formatting.

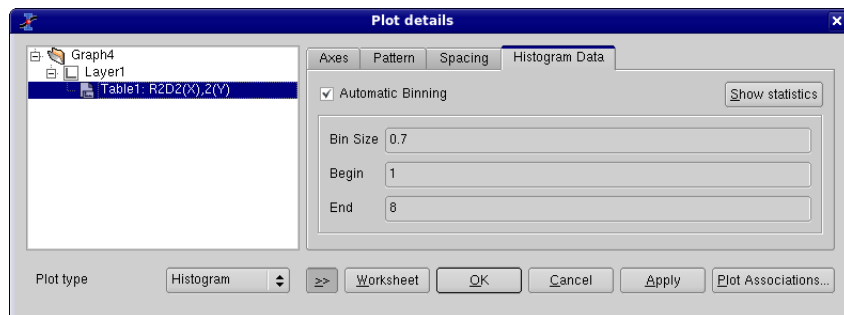


Figure 5.25: The Custom Curves Dialog for histogram: data formatting.

5.11 Define surface plot

This dialog is used when you enter the **New -> New Surface 3D Plot** command. It allows to create a new function of two variables. The only available coordinate system is the cartesian one: $z=f(x,y)$.

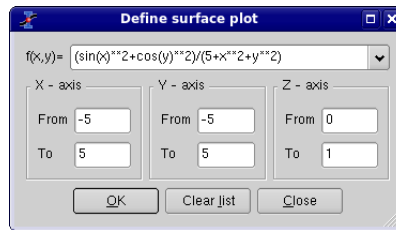


Figure 5.26: The **New -> New Surface 3D Plot** dialog box.

You can then enter the X, Y and Z scales.

5.12 Export ASCII

This dialog is activated by selecting the command **Export ASCII** from the **File Menu**. It is only active when a table is selected.

This command is used to export all or a part of the data of a project in an ASCII file.

In this example of export of a selection in a table to an ASCII file, TAB is used as a separator between the columns.

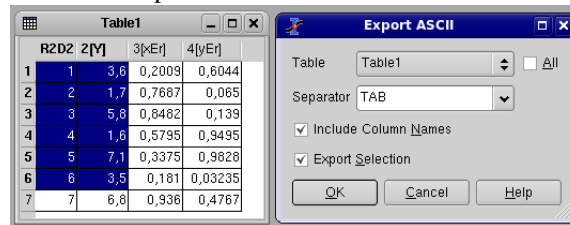


Figure 5.27: Export of a selection in a table to an ASCII file.

If the names of the columns have not been set, they will be set to C1,C2,... in the exported file. The formatting of the numbers is kept in the ASCII file, so you have to be careful to obtain a good enough precision in the ASCII file.

5.13 Fast Fourier Transform

The **FFT...** dialog box can be used either on a table or on a plot. It is used to compute a direct or inverse FFT. See the [FFT section](#) in the [Analysis chapter](#) for an example.

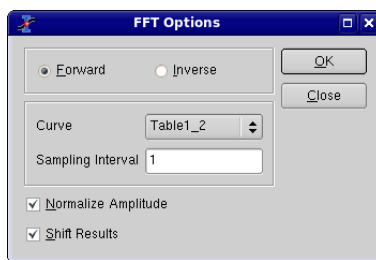


Figure 5.28: The **FFT...** dialog box for a curve.

SciDAVis will create a new plot window with the FFT amplitude curve, and a new table which contains the real part, the imaginary part, the amplitude, and the angle of the FFT.

If the *Normalize Amplitude* check box is on, the amplitude curve is normalized to 1. If the *Shift Results* check box is on, the frequencies are shifted in order to obtain a centered x-scale.

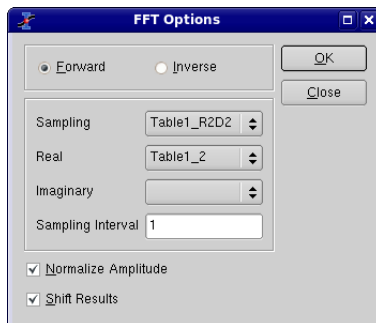


Figure 5.29: The **FFT...** dialog box for a table.

In the case of a table, you must select the sampling column (X-values) and two columns for Y-values. If they are complex numbers, the first column is the real part of Y-values and the second is the imaginary part. If Y-values are simple reals, you must select the same column for real and imaginary part.

By default, the *Sampling Interval* corresponds to the interval between X-values. Giving a smaller value makes no sense, but you can increase this value in order to sample less values

5.14 Integrate dialog

This dialog box is opened if you select the **Integrate...** command from the [Analysis menu](#). The first field is the curve that will be integrated. The second one is the order of the integration: the order 1 corresponds to the trapezoid rule, i.e. the curve is approximated by a straight line between 2 successive points. If you choose the order 2, three successive points are used and a second order polynome is used to approximate the curve. etc. If you have a large amount of points in your curve, the order 1 is enough.

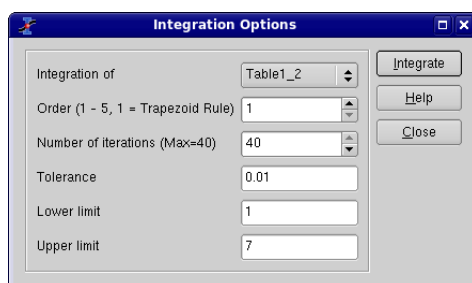


Figure 5.30: The **Integrate...** dialog box.

The result of the integration will be given in the [The Project Explorer](#).

5.15 Non linear curve fit

This dialog is activated by selecting the command [Non Linear Curve Fit...](#) from the [Analysis Menu](#). This command is active if a plot or a table window is selected. In the latter case, this command first creates a new plot window using the list of selected columns in the table.

This dialog is used to fit discrete data points with a mathematical function. The fitting is done by minimizing the least square difference between the data points and the Y values of the function.

Note: If the data points are modified, the fit is not re-calculated. Then, you need to remove the old fitted curve and to redo the fit with the same function and the new points.

The top of the dialog box is used to choose a function among the one which are already define. Four types of functions are available: the user defined functions which have been saved, the classical functions proposed by SciDAVis in the analysis menu, the simple elementary built-in functions, and external functions via pluggins.

To choose one of these functions, you just have to select it and to click on the checkbox under the selector.

If you want to define your own function, you can use the bottom half of the dialog box. You can write you own mathematical expression or add expressions obtained with

the function selector. Then you need to define the parameters which have to be fitted in a comma separated list.

This first step is used to define the function which will be used for the fitting

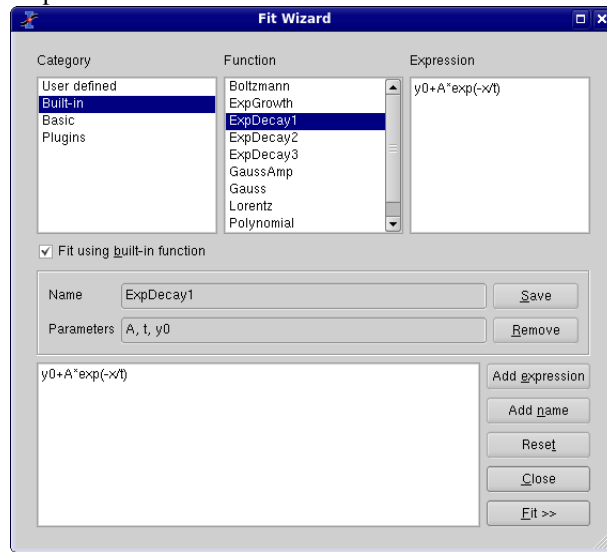


Figure 5.31: The first step of the **Non Linear Curve Fit...** dialog box.

The second step is to define the parameters for the fit. You have to give initial guess for the fitting parameters.

This second step is used to define the parameters of the fitting

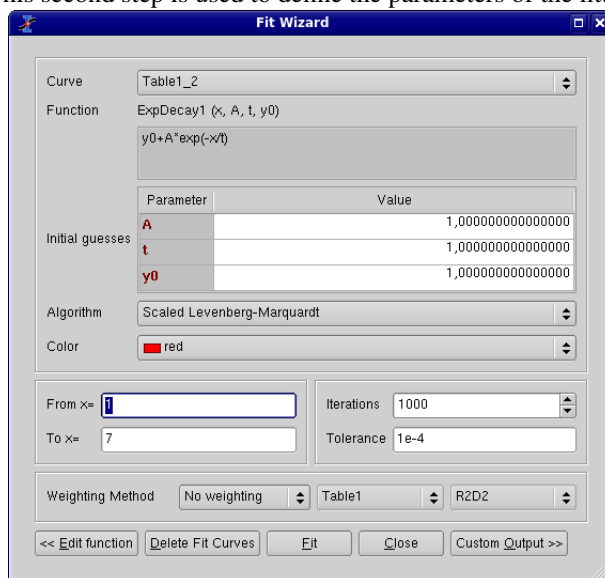


Figure 5.32: The second step of the **Non Linear Curve Fit...** dialog box.

In this second tab you can also choose a weighting method for your fit (the default is *No weighting*). The available weighting methods are:

1. *Instrumental*: the values of the associated error bars are used as weighting coefficients. You must add Y-error bars to the analysed curve before performing the fit.
2. *Statistical*: the weighting coefficients are calculated as the square-roots of each data point in the fitted curve.
3. *Arbitrary Dataset*: you have the possibility to set the weighting coefficients using an arbitrary data set. The column used for the weighting must have a number of rows equal to the number of points in the fitted curve.

After the fit, the log window is opened to show the results of the fitting process.

Depending on the settings in the *Custom Output* tab, a function curve (option *Uniform X Function*) or a new table (if you choose the option *Same X as Fitting Data*) will be created for each fit. The new table includes all the X and Y values used to compute and to plot the fitted function and is hidden by default, but it can be found and viewed with the [project explorer](#).

5.16 General Plot Options

The first tab is used to set the general scales used for the two or three axis.

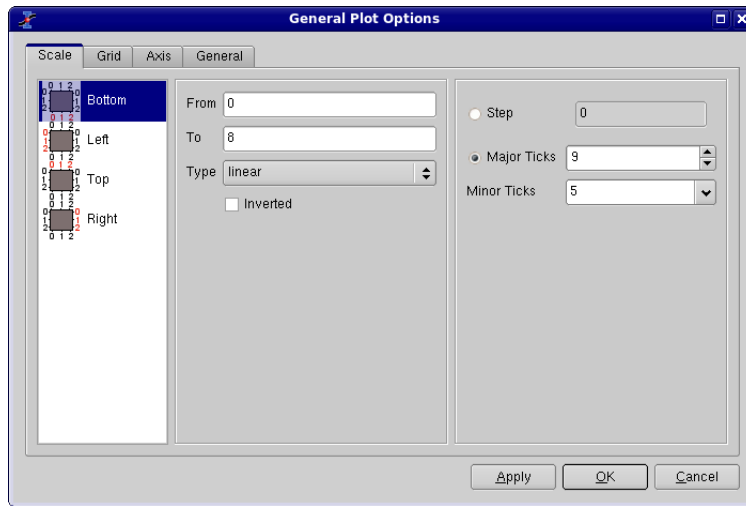


Figure 5.33: General plot options dialog: the scale tab.

In this tab, you can also set the number of ticks used for each axis. This can be done in two ways: you can set the number of labels which are used for the whole scale. Whatever the number you enter, SciDAVis will use a value which leads to a pretty plot: for example, if you enter 7 ticks for a 0..100 scale, SciDAVis will use 10 major ticks from 10 to 100. If you want to fix non classical values, you can select the step method.

The grid tab is used to draw grid lines on the plot. The frequency of the lines are related to the number of label and major ticks set with the *Scale* tab.

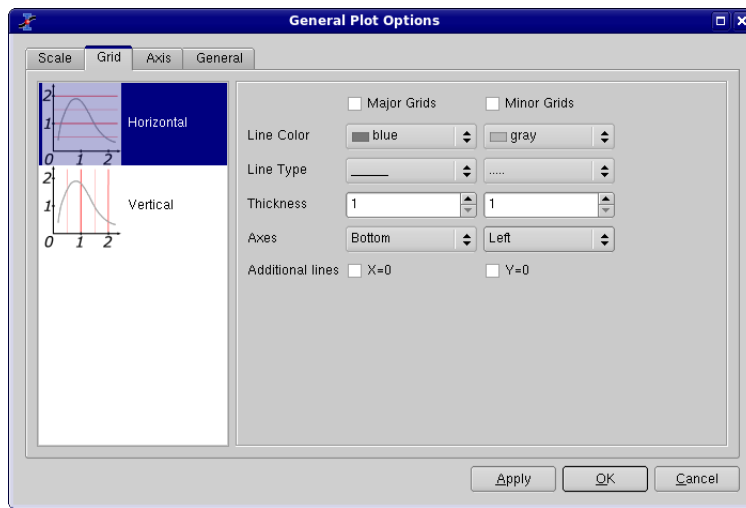


Figure 5.34: General plot options dialog: the grid tab.

The third tab is used to modify the setting of the different axis. You must select the axis that must be customized in the right window. The label of the axis can be modified in the title window, see the [text options dialog](#) section for more details.

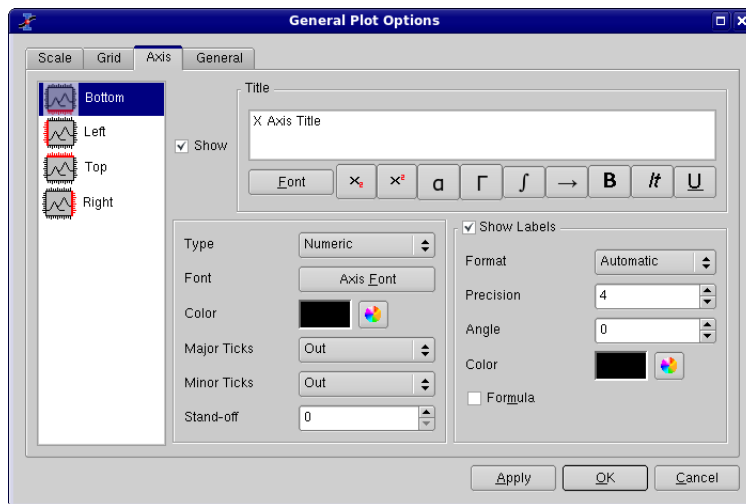


Figure 5.35: General plot options dialog: the axis tab.

The *General* settings tab is used to customize the global aspect of the plot. The canvas is the area defined by the axis, you can draw a box around this canvas and

define a background color for this canvas. The background area is the global drawing area, you can also define a color border and a background color for this area. The margin parameter controls the distance between the drawing area limit and the canvas. If you want to modify the margin between the window limits and the drawing area, you must modify the layer parameters (manually with the mouse or with the [arrange layers dialog](#)).

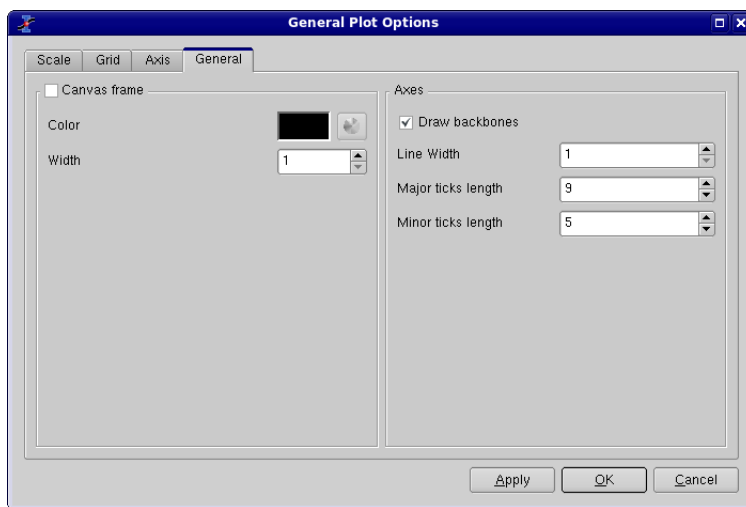


Figure 5.36: General plot options dialog: General settings.

The parameters in the *Axes* group allow to modify the linestyle of the axes and of the ticks.

5.17 Plot Wizard

This dialog is activated by selecting the command [Plot Wizard](#) from the [View Menu](#) or with the Ctrl-Alt-W key. This command is always active.

This dialog is used to build a new plot by selecting the columns in the tables available in the current project. At first, you have to select the table you want to use, and then click on *New curve* to create the curve. After that, you have to select at least one column for X and one for Y. You can also select one more column for X-errors or for Y-errors. The plot created will have the default style you defined using the [Preferences dialog](#) through the '2D Plots -> Curves' tab.

In this example, one curve is selected from the first table, and the other from the

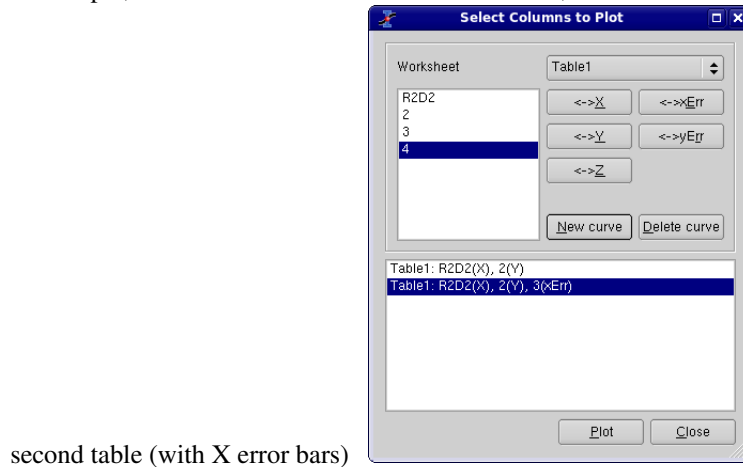


Figure 5.37: The plot wizard dialog box.

5.18 Project Explorer

The project explorer shows a list of all the windows, tables, matrices and folders which are included in the current project. It can be used to create new folders and windows, to find existing ones, to make hidden elements visible, to perform basic operations like: renaming, deleting, hiding, resizing, printing, etc... You can also use it in order to display the list of dependencies and properties of an element in the project.

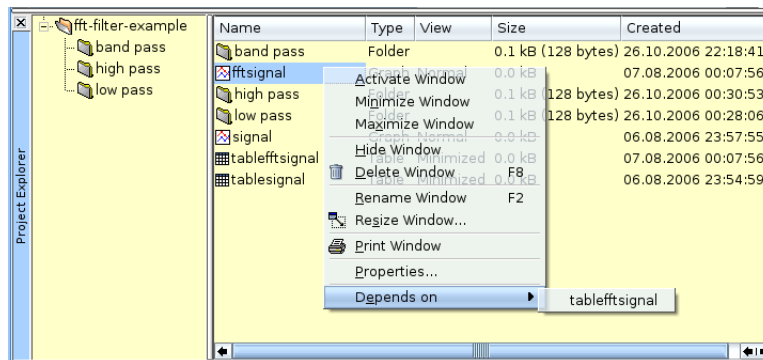


Figure 5.38: The project explorer panel.

5.19 Preferences Dialog

The preference dialog is used to customize the application. It has six different tabs. If you confirm your changes to the default behaviour of the application, the changes are saved and stored immediately.

The first icon can be selected to change the *General* options of the application. In the first tab: *Application*, the style is the general decoration used for the windows. It defines the aspect of the buttons and dialog boxes, as an example all screenshots presented in this manual have been done with the Keramik style available in KDE. The available styles are part of the Qt library. The font is the general font used for the GUI (menus, dialogs, etc), it doesn't apply to the plots. You can select the language of the application in the corresponding combo-box. All the available translations can be downloaded from the following address: <http://soft.proindependent.com/translations.html> and must be placed in a folder called *translations*, situated in the same location as the SciDAVis executable, in order to be loaded by the application.

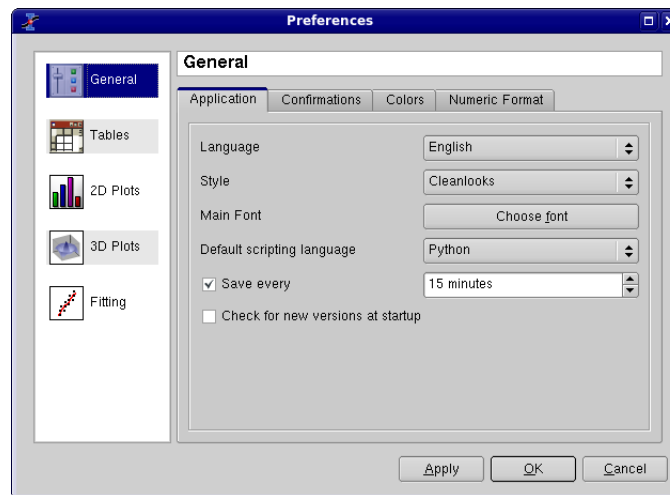
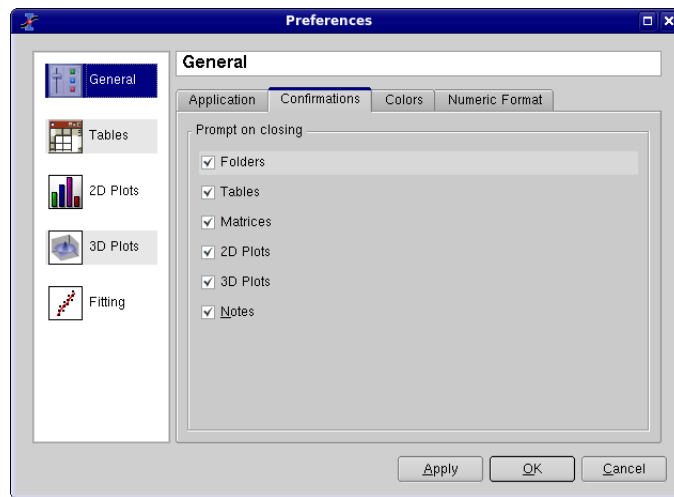
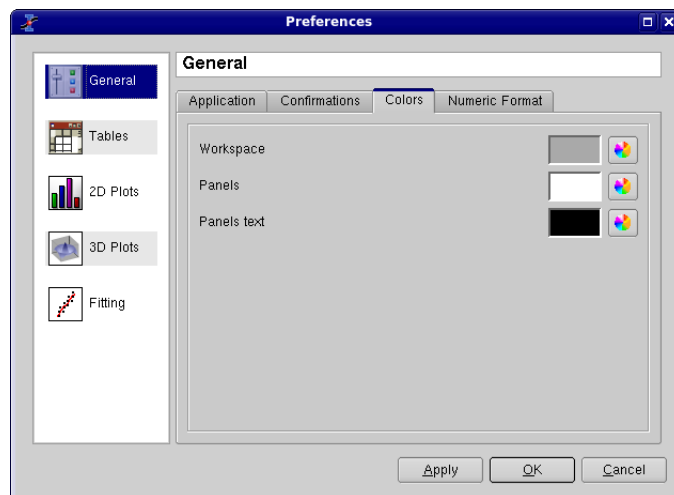


Figure 5.39: The preferences dialog: general parameters for the application.

The second tab of the *General* option set is used to disable the prompting on deleting of objects.



In this tab, you can change the default color for the workspace of the application. You can also choose the background color and the text color for panels. The panels are the [Log Window](#) and the [Project explorer](#).



The second set of option allows to customize the default aspect of [tables](#): background and text colors, and fonts for tables and labels.

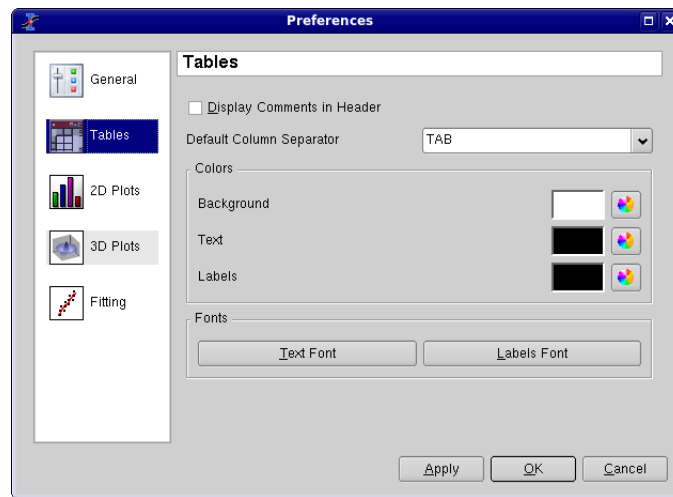


Figure 5.40: The preferences dialog: table options.

The second set of options is used to customize the default aspect of *2D plots*. The first tab is used to modify general options.

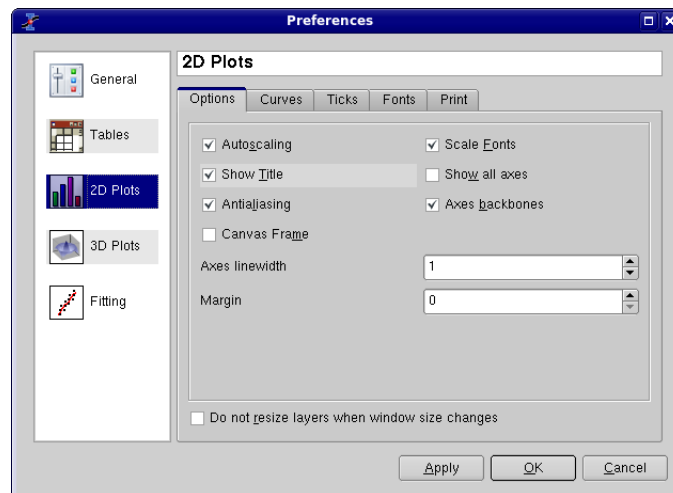
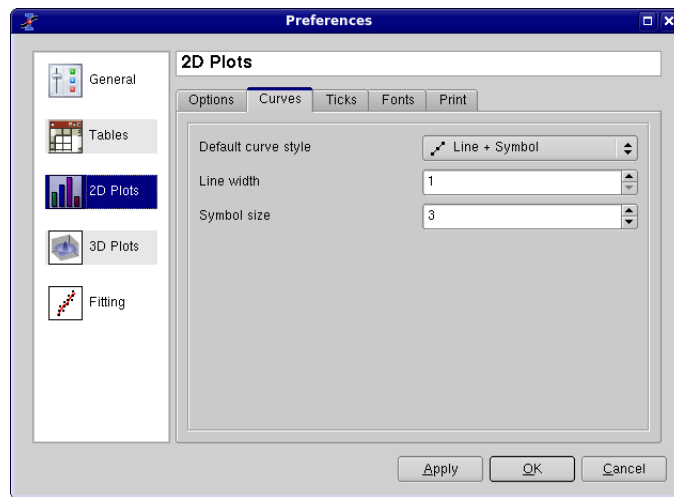
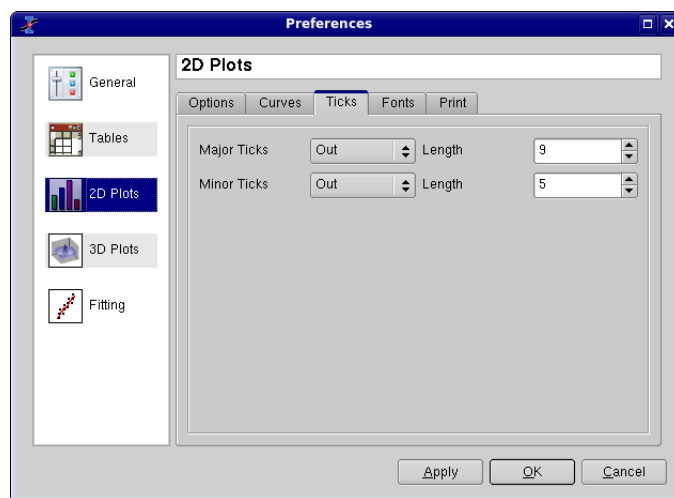


Figure 5.41: The preferences dialog: 2D plot options.

The second tab named *Curves* defines the default style used when you create a new plot.



The third tab named *Ticks* defines the default style for the ticks of the axes used when you create a new plot.



The fourth tab named *Fonts* defines the default style for the fonts used for the axes, used when you create a new plot.

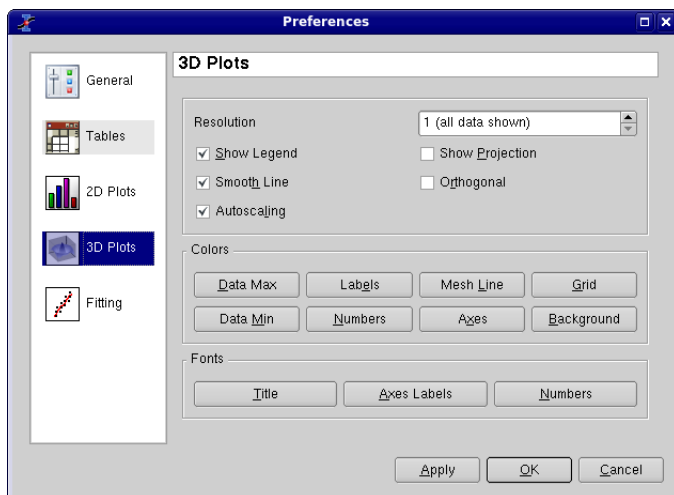
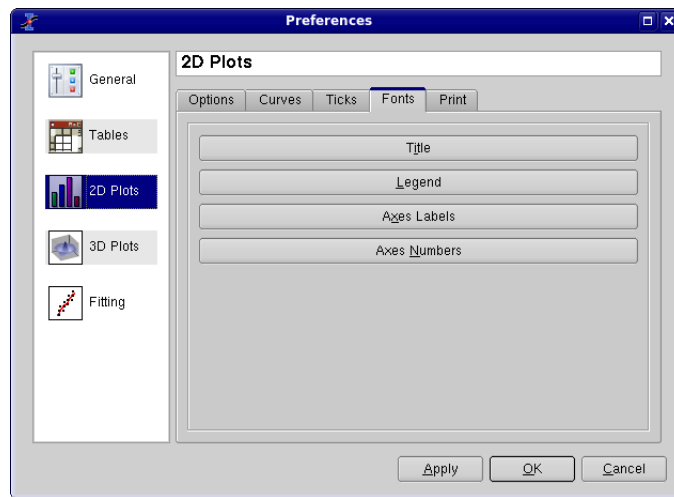


Figure 5.42: The preferences dialog: 3D plot options.

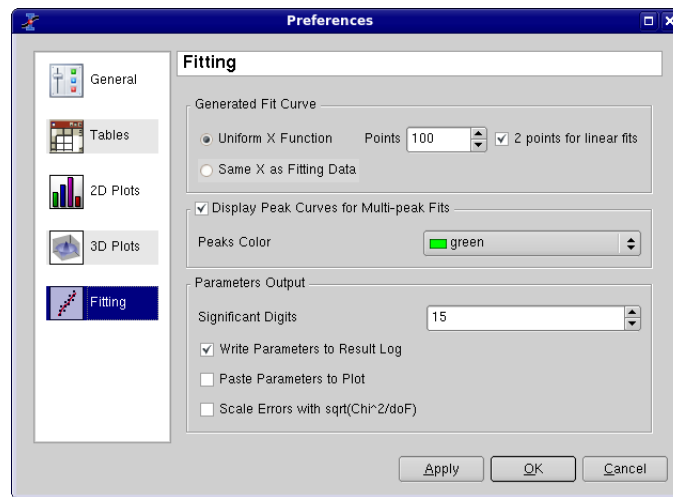


Figure 5.43: The preferences dialog: fitting options.

5.20 Printer-setup

This dialog box is opened by the [Print command](#) from the [File menu](#). It is used to print the selected window (plot or table) and its aspect depends on your operating system. The following screenshot shows this dialog on a Linux system using the KDE window manager.

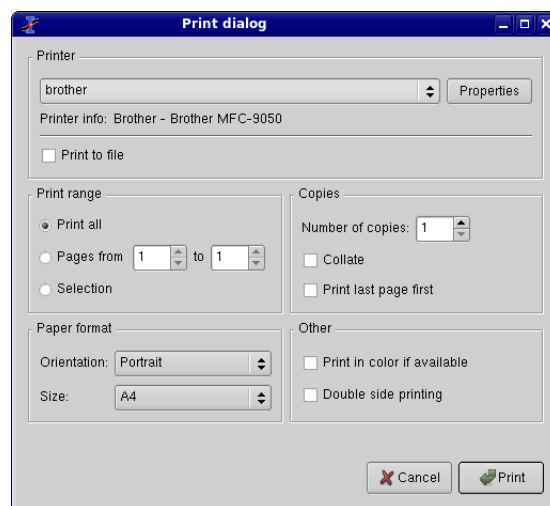


Figure 5.44: The **Print** dialog.

5.21 Set Column Values

This dialog is activated by the [Set Column Values...](#) command of the [Table menu](#). It allows to fill a column with the result of a function.

The available mathematical functions (assuming you are using the default scripting language, muParser) are listed in the [appendix](#). The special function *col(x)* can be used to access to the values of the column x, where x can be the column's number (as in *col(2)*) or its name in doublequotes (as in *col("time")*). You can also get values from other tables using the function *tablecol(t,c)*, where t is the table's name in doublequotes and c is the column's number or name in doublequotes (example: *tablecol("Table1","time")*).

The variables *i* and *j* can be used to access the current row and column numbers. Similarly, *sr* and *er* represent the selected start and end row, respectively.

Using Python as scripting language gives you even more possibilities, since you can not only use arbitrary Python code in the function body, but also access other objects within your project. For details, see [here](#).

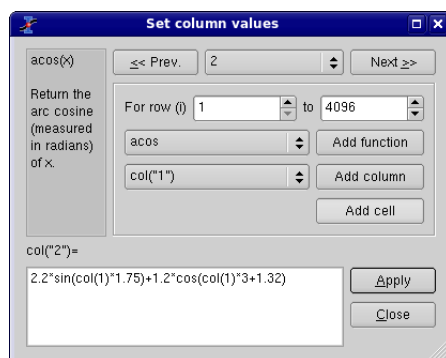


Figure 5.45: The **Set Column Values...** dialog.

If you make some changes in the table, the values are not computed again. You have to explicitly tell SciDAVis to recalculate individual cells or whole columns or rows by selecting "Recalculate" from their context menu or pressing Control+Return.

5.22 Set Dimensions...

This command is in the [Matrix menu](#). It allows to specify the number of rows and columns of a matrix

In this window, you can also define X-values and Y-values. These X and Y ranges will only be used by the 3D-plot, they are not known if you choose to define the content of the matrix with the [Set Values Dialog](#).

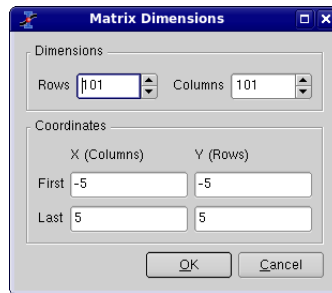


Figure 5.46: The **Set Dimensions...** dialog for matrix.

5.23 Set options for Importation of ASCII files.

This dialog is activated by selecting the command **Set Import Options...** from the **File Menu**.

This dialog is used to set the options which are used for the importation of ASCII data files by the commands **Import ASCII -> Single File...** and **Import ASCII -> Multiple Files...**

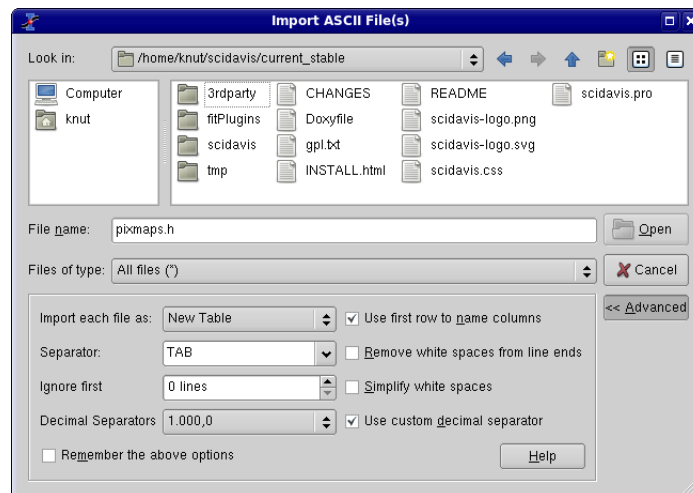


Figure 5.47: The **Set Import Options...** dialog box.

The first parameter is the separator which is used between the columns. The second allows to skip the n first lines of the file.

- If you choose to use the first line as column names, you must use the same separator between the column names and between the data columns.

- There is no grouping of separators, so if you use "SPACE", you must put only *one* separator between each column.

5.24 Set Properties...

This command is in the [Matrix menu](#). It allows to specify some global properties of the selected matrix such as the cell width (in pixels) and the format for numbers.

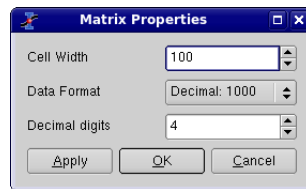


Figure 5.48: The **Set Properties...** dialog for matrix.

5.25 Set Values...

This command is in the [Matrix menu](#). It allows to fill in a matrix with the results of a function $z=f(i,j)$ in which i and j are the row and column numbers.

Even if you have defined X-values and Y-values with the [Set Dimensions...](#) command, you must use i and j as parameters for the function. In the example below, X and Y ranges have been defined as 101 values (i and j from 1 to 101) between -5 and +5. Therefore, the function uses as entries the parameters $x=(j-1)/100-5$ and $y=(i-1)/100-5$.

The functions can be written on several lines, and the intrinsic functions which are available are listed in the [appendix](#).

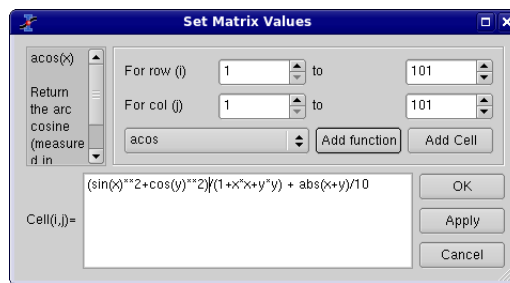


Figure 5.49: The **Set Values...** dialog for matrix.

5.26 Surface plot options

This dialog box is used to customize a 3D function plot which has been created by the [New -> New Surface 3D Plot command](#) from the [File menu](#). It is activated by a double click on the 3D plot.

The first tab is used to modify the X, Y and Z ranges. It allows also to specify the number of labels on the axis and the number of secondary ticks.

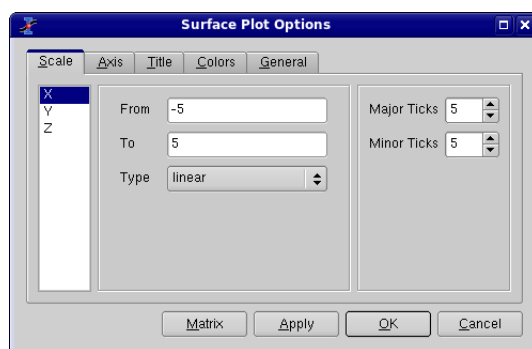
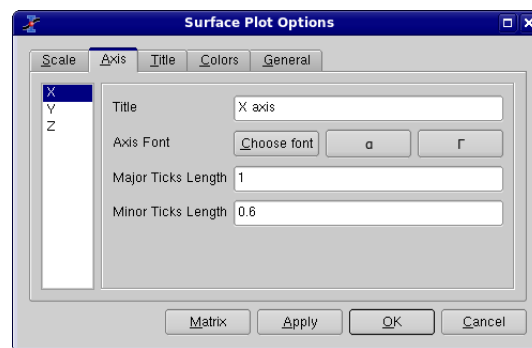
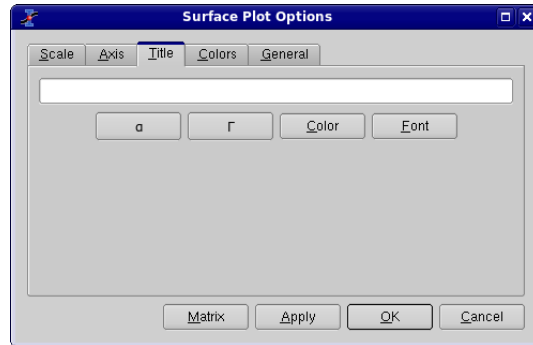


Figure 5.50: The surface plot options dialog box.

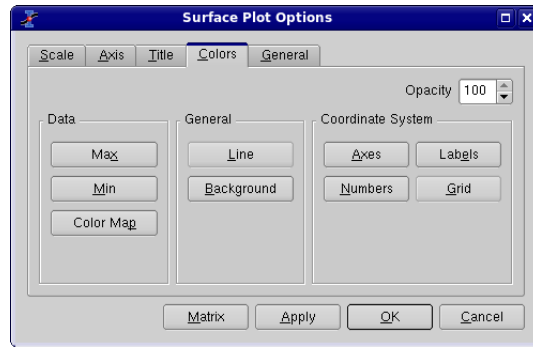
The second tab defines the main parameters of the three axis: the axis label and its font, and the length of the ticks. This length is defined in the same units as the range of the axis. If something is changed in the scales of the graph, the length of the ticks is re-calculated by SciDAVis. The font button allows to modify only the font used for the label, if you want to customize the font of the numbers used for the axis, you must use the fifth tab.



The third tab is used to define or modify the title of the plot. You can not add subscripts/superscripts, bold characters, etc in your title as you can do it for 2D plots.



The fourth tab allows to modify the colors used in the different elements of the plot.



The first set of two colors (data min and data max) defines the color scheme which is used to show the Z-values. They are the colors used for the minimum value of Z (Z_{\min}) and the maximum value of Z (Z_{\max}). We can define the colors by their Red, Green and Blue parameters: $[R,G,B]$. Then, a value Z will be represented by a color defined as a linear interpolation:

$$\begin{aligned} R &= R_{\min} + \frac{(R_{\max} - R_{\min})}{(Z_{\max} - Z_{\min})} (Z - Z_{\min}) \\ G &= G_{\min} + \frac{(G_{\max} - G_{\min})}{(Z_{\max} - Z_{\min})} (Z - Z_{\min}) \\ B &= B_{\min} + \frac{(B_{\max} - B_{\min})}{(Z_{\max} - Z_{\min})} (Z - Z_{\min}) \end{aligned}$$

The default colors for Z_{\min} and Z_{\max} are respectively blue ($[R,G,B] = [0,0,255]$) and red ($[R,G,B] = [255,0,0]$). This lead to the following color scheme:



Another classical color scheme can be built with $Z_{\min} = [160,32,32]$ and $Z_{\max} = [255,255,0]$ (yellow). It leads to:



Another way to define colors is to read a colormap from a file. The format of the file is simple: each line defines a color by red, green and blue values as integers between 0

and 255. The numbers should be separated by spaces. You can find several examples of colormaps on the [QwtPlot3D web site](#).

The last tab is used to define some global parameters and the aspect ratio of the plot. The default behaviour is to use the perspective to compute the 3D plot. If you choose to check the *Orthogonal* check box, the plot will use vertical Z axis whatever the view angle of the plot.

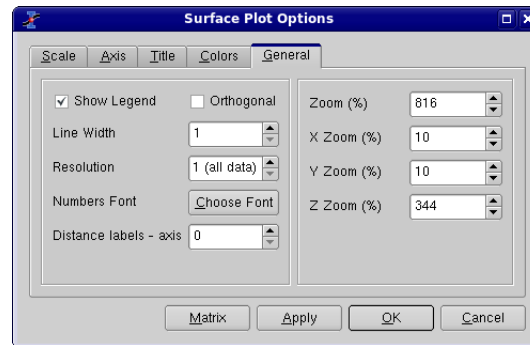


Figure 5.51: The surface plot options dialog box with tab 5: aspect ratio.

5.27 Text options

This dialog can be opened by several commands such as [Title... command](#) or when you double click on a text object in your plot. It allows to add/customize the text objects.

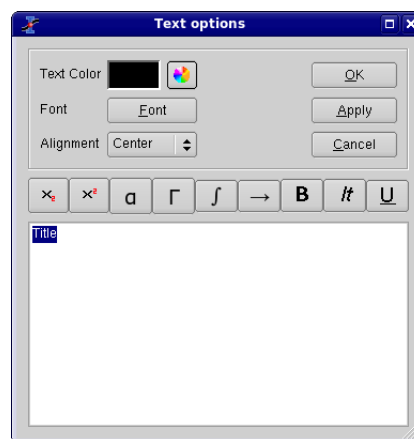





Figure 5.52: The text options dialog.


The *Color*, *Font* and *Alignment* commands allow the modification of the general

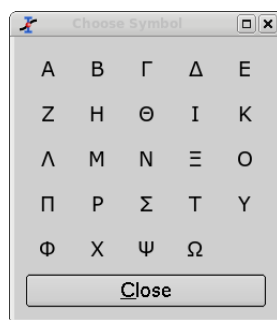
settings of the text label.




The text item can be modified in the text window. Several improvements can be added to the text:

- `_{text}` will draw the text as subscripts. You can insert this sequence by clicking on the .
- `^{text}` will draw the text as superscripts. You can insert this sequence by clicking on the .
- By clicking on the , you can open a new dialog which allows to select greek characters:



- By clicking on the , you can open a new dialog which allows to select various mathematical symbols:



- `text` will draw the text with bold characters. You can insert this sequence by clicking on the .
- `<i>text</i>` will draw the text with italic characters. You can insert this sequence by clicking on the .
- `<u>text</u>` will draw the text with underlined characters. You can insert this sequence by clicking on the .

Chapter 6

Analysis of data and curves

6.1 Fast Fourier Transform

This function can be accessed by the command [FFT....](#) It can be found in the [Analysis Menu](#) when a table or a plot is selected. The Fourier transform decomposes a signal in its elementary components by assuming that the signal $x(t)$ can be describe as a sum:

$$x(t) = \sum_n a_n \cos(\omega_n t + \psi_n)$$

EQUATION 6.1.1: Fourier equation

in which ω_n are the frequencies, a_n are the amplitudes of each frequency and ψ_n are the phase corresponding frequency. SciDAVis will compute these parameters and build a new plot of the amplitude as a function of the frequency.

FFT performed on a curve to extract the characteristic frequencies. The signal is on the bottom plot, while the amplitude-frequency plot is on the top layer. In this example, the amplitude curve has been normalized, and the frequencies have been shifted to obtain a centered x-scale.

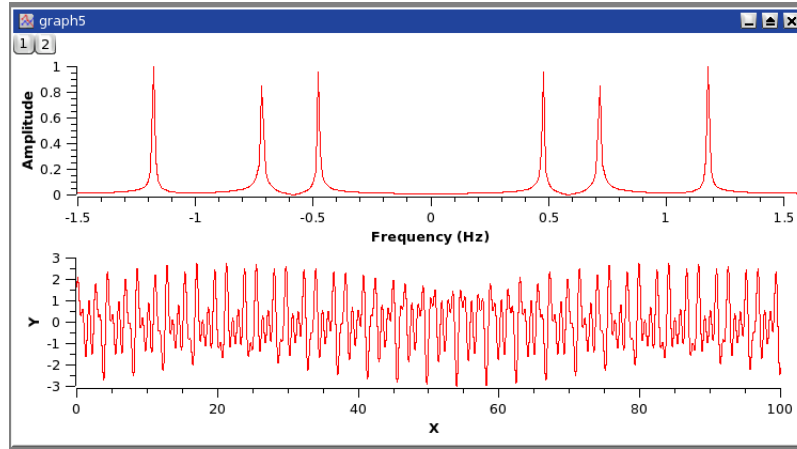


Figure 6.1: An example of a inverse FFT.

Some parameters of the FFT can be modified in the [FFT dialog](#).

6.2 Correlation

This function can be accessed by the command [Correlate](#). It can be found in the [Analysis Menu](#) when a table is selected. The correlation function, also known as the covariance function is used to test the similarity of two signals $x(t)$ and $y(t)$. It is computed by:

$$R(\tau) = \overline{(x(t) - \bar{x}) \cdot (y(t + \tau) - \bar{y})}$$

EQUATION 6.2.1: Covariance function of two signals $x(t)$ and $y(t)$

in which \bar{x} and \bar{y} are the mean values of the signals $x(t)$ and $y(t)$ respectively.

If the number of points is N , the function will be computed between $-N/2$ and $N/2$. The abscissae are therefore point numbers and not t values.

The first plot shows the two signals, the second one is the correlation function between the two signals which shows that there are correlations, and the third one is the Fourier transform which is done to extract the characteristic frequencies of the correlation function.

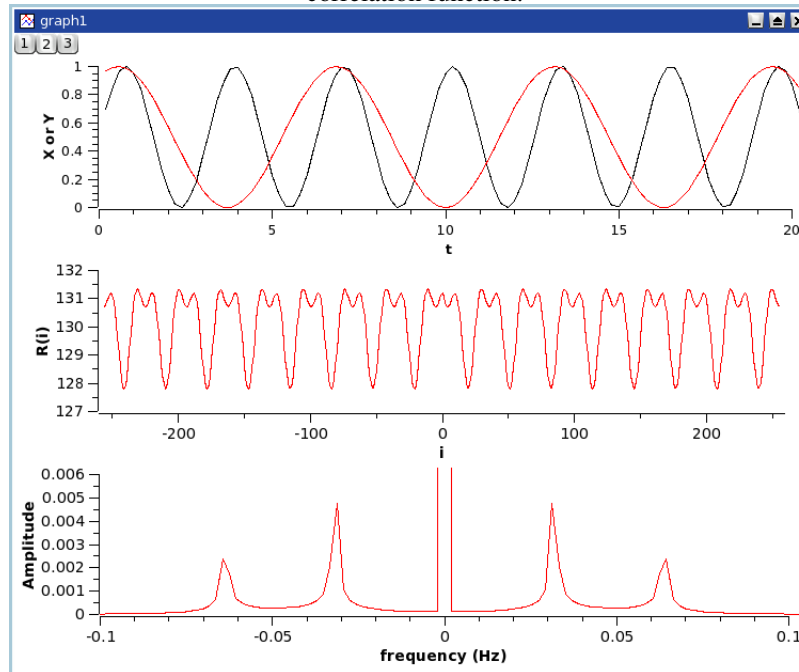


Figure 6.2: An example of a correlation between two sinus functions.

The correlation of a signal with itself can also be used in spectral analysis (it is then called autocorrelation or autocovariance function).

6.3 Convolution

.

6.4 Deconvolution

.

6.5 Non Linear Curve Fit

This function can be accessed by the command [Non Linear Curve Fit....](#) It can be found in the [Analysis Menu](#) when a table is selected.

The results are shown in the log window, the curve is plotted in the active window, and a table is created to store the fit.

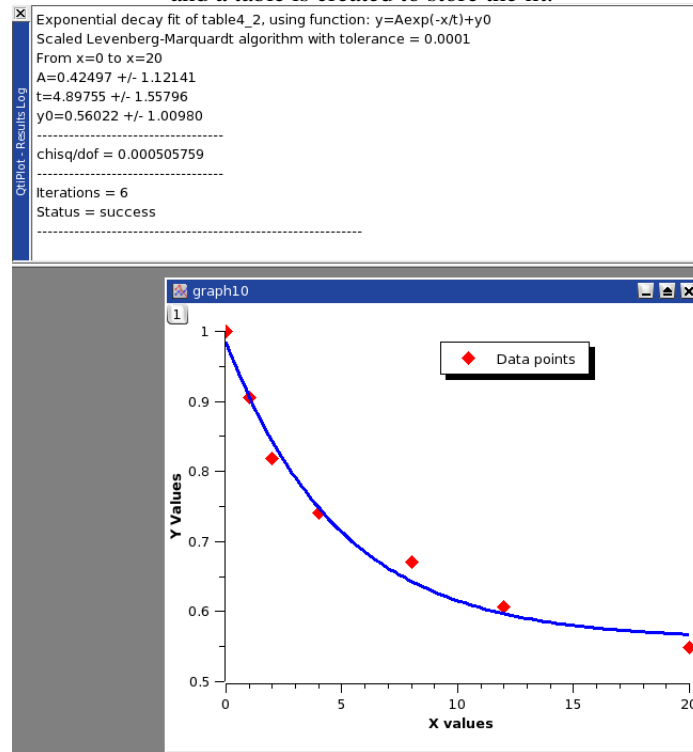


Figure 6.3: The results of the **Non Linear Curve Fit...**

6.6 Fitting to specific curves

SciDAVis include quick access to the most usefull functions for fitting.

6.6.1 Fitting to a line

This command is used to fit a curve which has a linear shape.

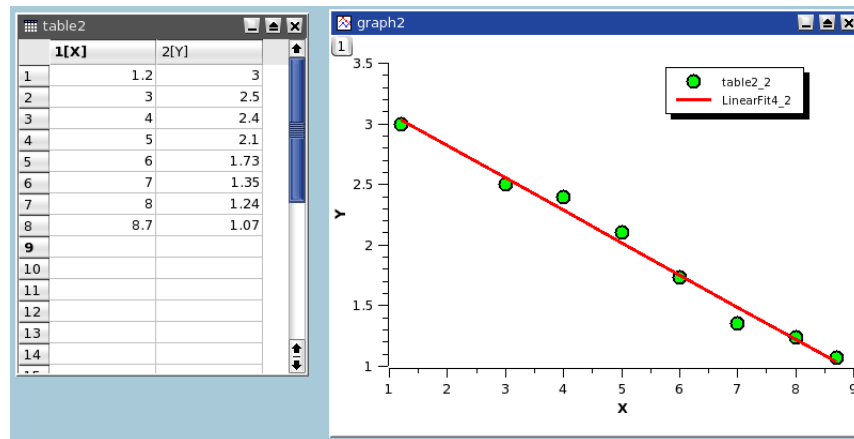


Figure 6.4: The results of a **Fit Linear**.

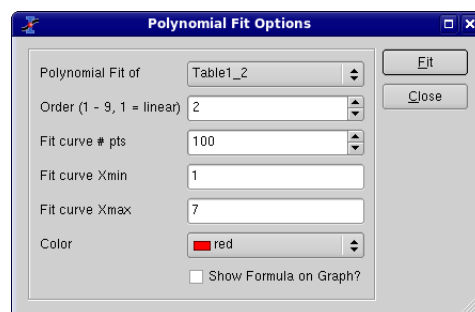
The results will be given in the [Log panel](#):

```

11.05.2006 22:29:50      LinearFit3:
Linear regression of table2_2: y=A*x+B
From x=1.2 to x=8.7
A = -0.266281 +/- 0.0126381
B = 3.35168 +/- 0.0742493
-----
sumsq = 0.0441584
Rsquare = 0.986665
  
```

6.6.2 Fitting to a polynome

This command is used to fit a curve which has a linear shape. The results will be given in the [Log panel](#)



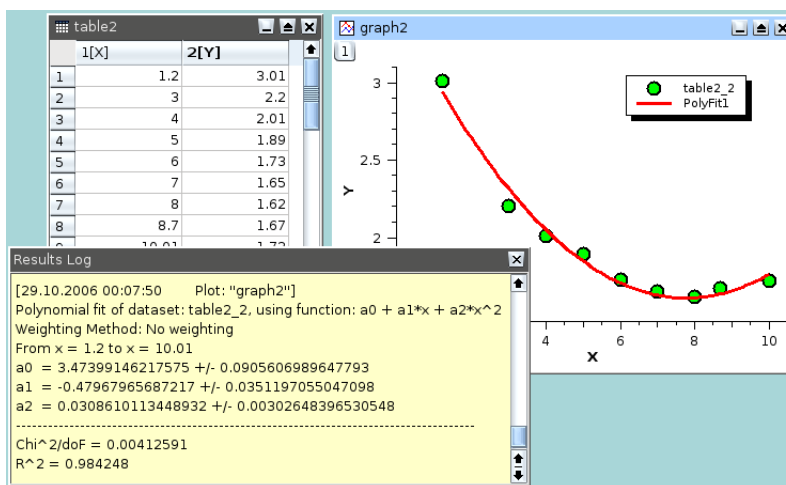


Figure 6.5: The results of a **Fit Polynomial...**, showing the initial data, the curve added to the plot, and the results in the log panel.

6.6.3 Fitting to a Boltzmann function

This command is used to fit a curve which has a sigmoidal shape. The function used is:

$$y = (A_2 - A_1) \left(1 + \exp \left(\frac{x - x_0}{dx} \right) \right)$$

EQUATION 6.6.1: Boltzmann equation

in which A_2 is the high Y limit, A_1 is the low Y limit, x_0 is the inflexion point and dx is the width.

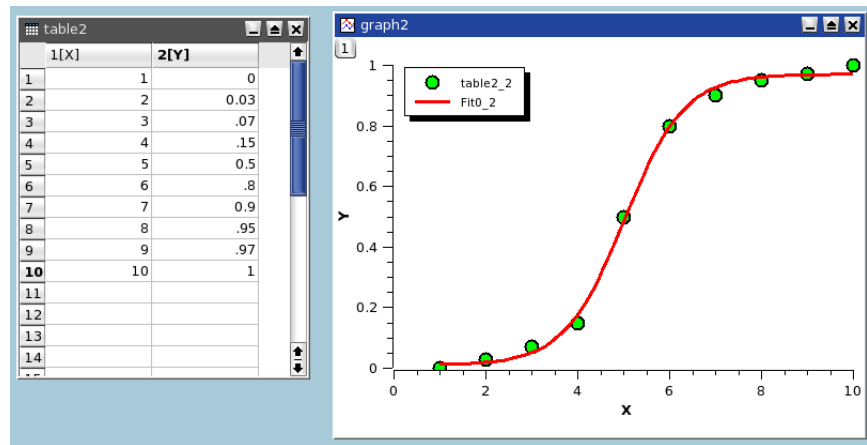


Figure 6.6: The results of a **Fit Boltzmann (sigmoidal)**.

6.6.4 Fitting to a Gauss function

This command is used to fit a curve which has a bell shape. The function used is:

$$y = y_0 + A \exp\left(\frac{-(x - x_c)^2}{2w^2}\right)$$

EQUATION 6.6.2: Gauss equation

in which A is the height, w is the width, x_c is the center and y_0 is the Y-values offset.

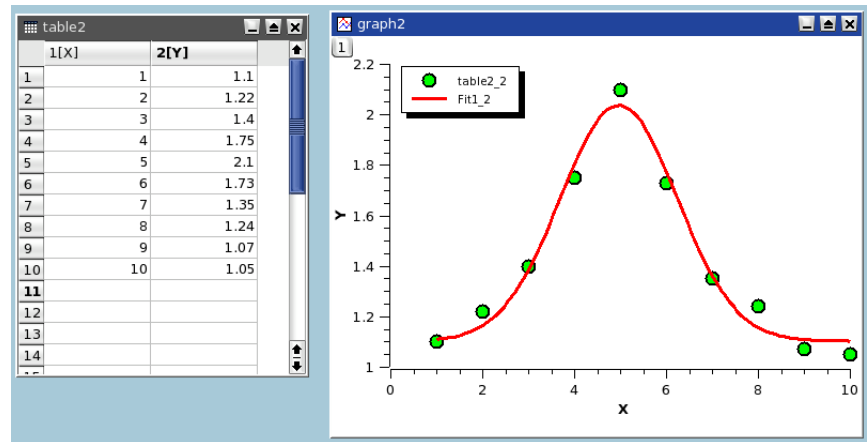


Figure 6.7: The results of a **Fit Gaussian**.

6.6.5 Fitting to a Lorentz function

This command is used to fit a curve which has a bell shape. The function used is:

$$y = y_0 + 2 \frac{A}{\pi} \frac{w}{4(x - x_c)^2 + w^2}$$

EQUATION 6.6.3: Lorentz equation

in which A is the area, w is the width, x_c is the center and y_0 is the Y-values offset.

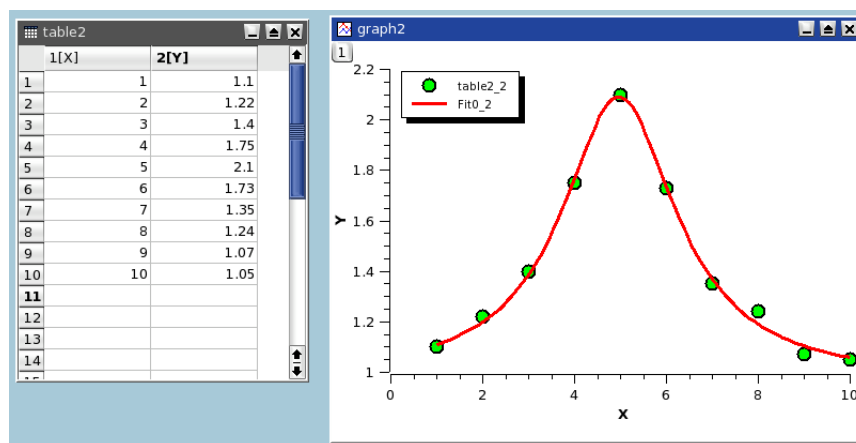


Figure 6.8: The results of a **Fit Lorentzian**.

6.7 Multi-Peaks fitting

This kind of fitting allows to fit your data points to a sum of N gaussian or lorentzian functions.

The first step is to specify the number of peaks. Then you must define the position of each peak on the curve. This is done by clicking on the plot, then validate your choice for each peak with the *ENTER* key.

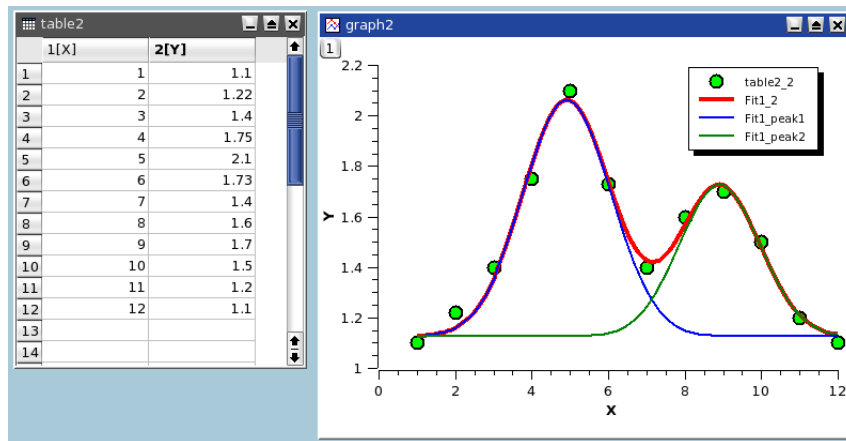
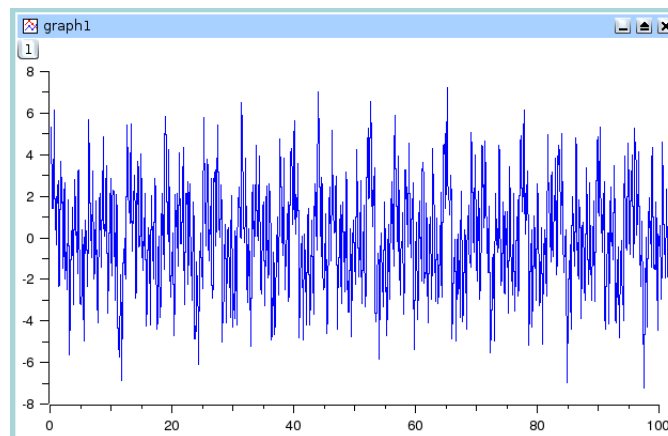


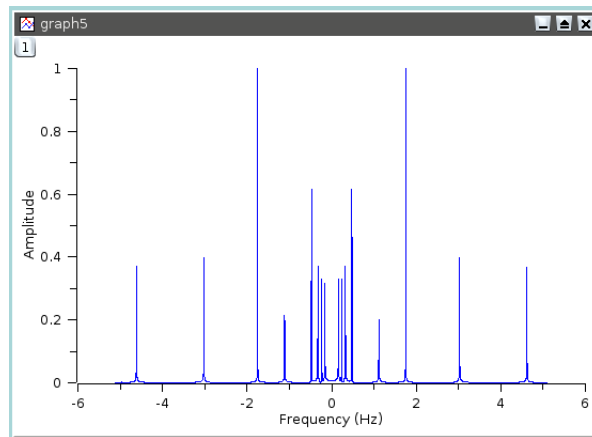
Figure 6.9: The results of a **Fit Multi-peak ->Gaussian....**

6.8 Filtering of data curves

In this section, it will be assumed that you have the following data curve:



This signal has a power spectrum with high and low frequencies. We can analyze this by doing a FFT on the data curve, this leads to the following figure:



The new sections will show the influence of the different filters on this data curve.

6.8.1 FFT low pass filter

This filter allows to cut the high frequencies of a signal. You just have to select the cut-off frequency of the filter. Let us assume that we want to keep the frequencies below 1 Hz, we will obtain:

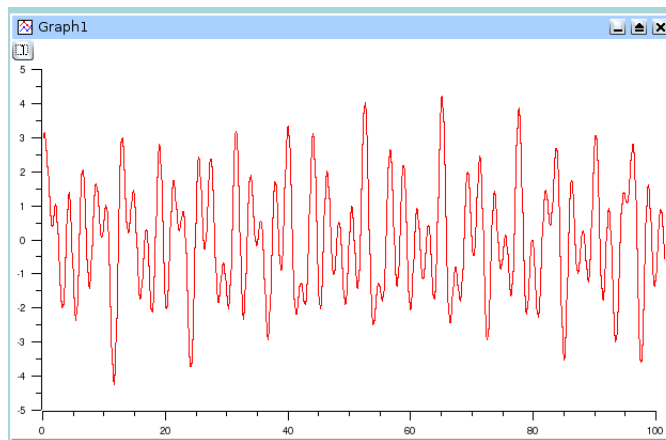
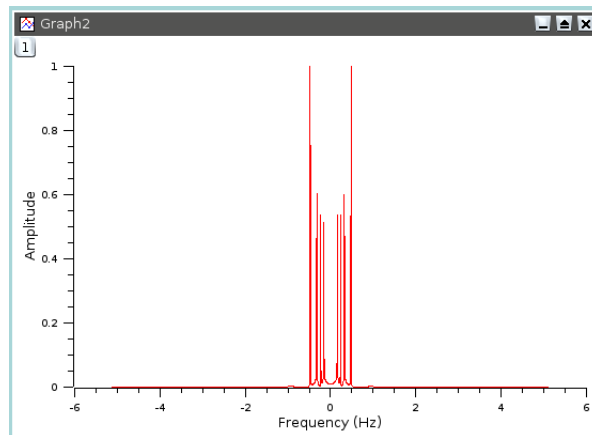


Figure 6.10: Signal after a FFT low pass filter

The power spectrum of this new signal shows that the frequencies below 1 Hz have been kept.



6.8.2 FFT high pass filter

This filter allows to cut the low frequencies of a signal. You just have to select the cut-off frequency of the filter. Let us assume that we want to keep the frequencies above 1 Hz, we will obtain:

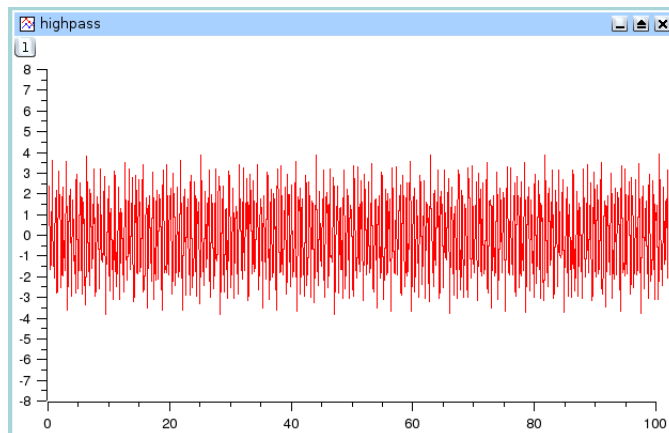
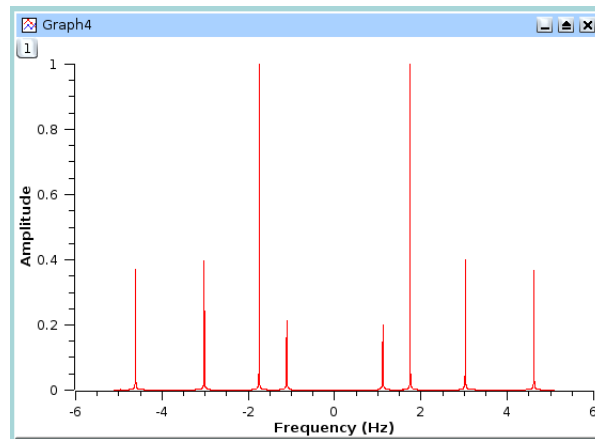


Figure 6.11: Signal after a FFT high pass filter

The power spectrum of this new signal shows that the frequencies above 1 Hz have been kept.



6.8.3 FFT band pass filter

This filter allows to cut the low and high frequencies of a signal. You just have to select the high and low cut-off frequencies of the filter. Let us assume that we want to keep the frequencies between 1.5 and 3.5 Hz, we will obtain:

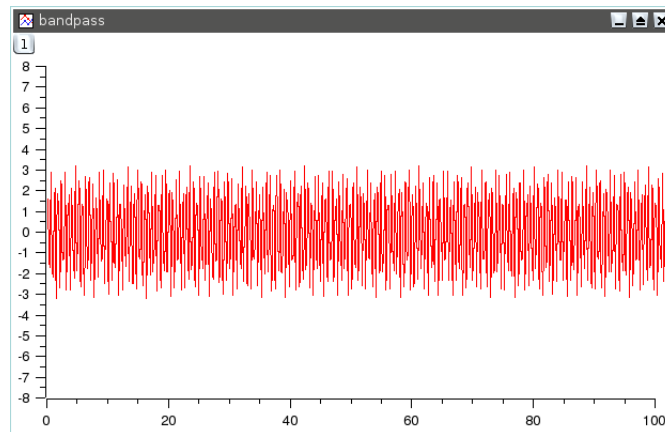
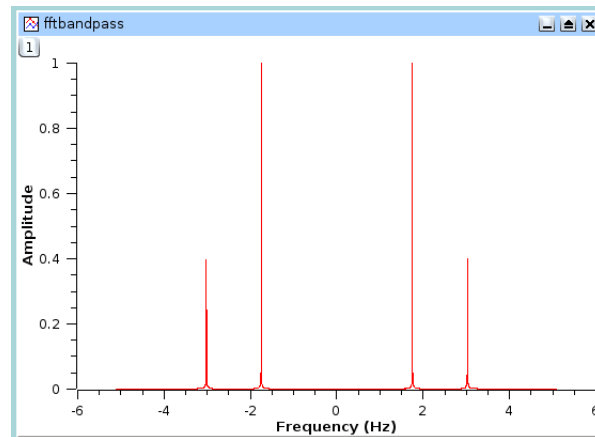


Figure 6.12: Signal after a FFT band pass filter

The power spectrum of this new signal shows that only the frequencies at 2 and 3 Hz have been kept.



6.8.4 FFT block band filter

This filter allows to keep the low and high frequencies of a signal. You just have to select the high and low cut-off frequencies of the filter. Let us assume that we want to remove the frequencies between 1.5 and 3.5 Hz, we will obtain:

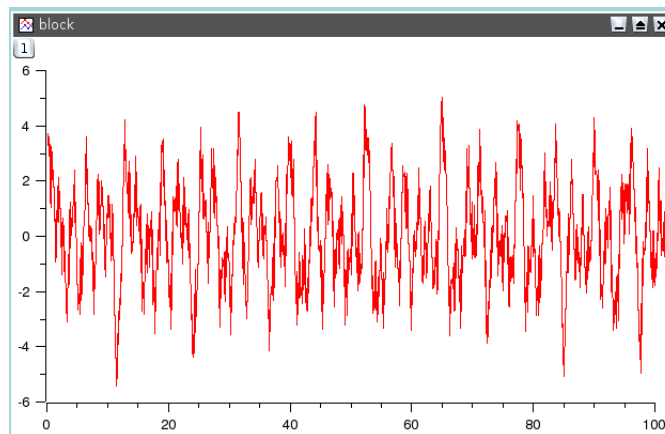
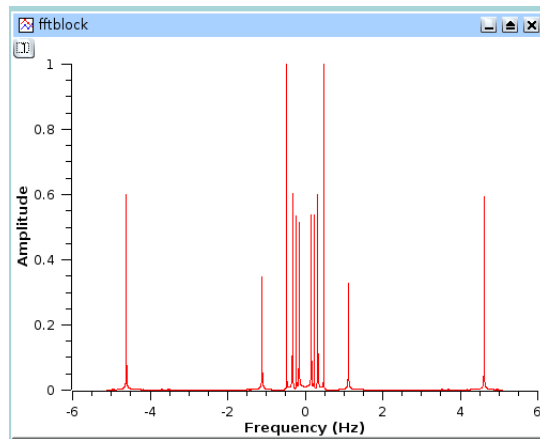


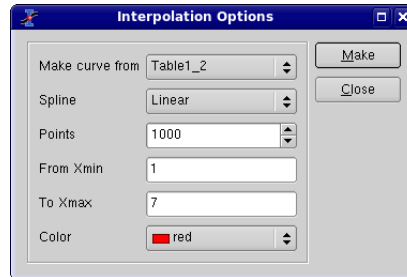
Figure 6.13: Signal after a FFT block band filter

The power spectrum of this new signal shows that only the frequencies below 1.5 Hz and above 3.5 Hz have been kept.



6.9 Interpolation

The interpolation command will create a new data curve with a high number of points by interpolation of your data. The dialog box allows to define this number of points (default value = 1000). Then the method used for interpolation, the interval of X-values and the color of the interpolated curve can be chosen. In addition to the new curve in the active plot, a new table will be created.



The simplest interpolation method is the linear method. In this case, a linear variation is used to compute the data points between two values. The cubic method will use the Cubic Splines method (in this case at least 4 points are needed). The last method *Akima* is a polynomial interpolation. You can refer to the corresponding section of the [GNU Scientific Library](#) for more details.

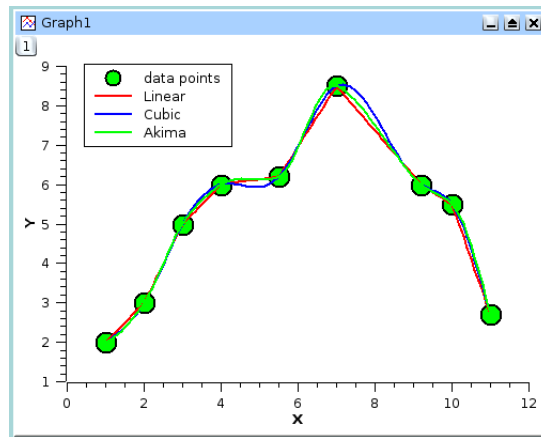


Figure 6.14: Comparison of the three methods of interpolation

Chapter 7

Mathematical Expressions and Scripting

SciDAVis supports different interpreters for evaluating mathematical expressions and for executing scripts.

7.1 muParser

The constants `_e=e=E` and `_pi=pi=PI=Pi` are defined, as well as the following operators and functions. muParser is purely a parser for mathematical expressions and as such does not support executing scripts.

| Name | Description |
|------|--|
| + | Addition |
| - | Substraction |
| * | Multiplication |
| / | Division |
| ^ | Exponentiation (raise a to the power of b) |
| and | logical and (returns 0 or 1) |
| or | logical or (returns 0 or 1) |
| xor | logical exclusive or (returns 0 or 1) |
| < | less then (returns 0 or 1) |
| <= | less then or equal (returns 0 or 1) |
| == | equal (returns 0 or 1) |
| >= | greater then or equal (returns 0 or 1) |
| > | greater then (returns 0 or 1) |
| != | not equal (returns 0 or 1) |

Table 7.1: Supported Mathematical Operators

| Name | Description |
|----------------------|---|
| abs(x) | absolute value of x |
| acos(x) | inverse cosinus |
| acosh(x) | inverse hyperbolic cosinus |
| asin(x) | inverse sinus |
| asinh(x) | inverse hyperbolic sinus |
| atan(x) | inverse tangent |
| atanh(x) | inverse hyperbolic tangent |
| avg(x1,x2,x3,...,x5) | average value, this command accept a list of arguments separated by commas |
| bessel_j0(x) | Regular cylindrical Bessel function of zeroth order, $J_0(x)$. |
| bessel_j1(x) | Regular cylindrical Bessel function of first order, $J_1(x)$. |
| bessel_jn(x) | Regular cylindrical Bessel function of n^{th} order, $J_n(x)$. |
| bessel_y0(x) | Regular cylindrical Bessel function of zeroth order, $Y_0(x)$ for $x>0$. |
| bessel_y1(x) | Regular cylindrical Bessel function of first order, $Y_1(x)$ for $x>0$. |
| bessel_yn(x) | Regular cylindrical Bessel function of n^{th} order, $Y_n(x)$ for $x>0$. |
| beta(a,b) | Computes the Beta Function, $B(a,b) = \frac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)}$ for $a > 0$ and $b > 0$. |
| cos(x) | cosinus of x |
| cosh(x) | hyperbolic cosinus of x |
| erf(x) | error function of x |
| erfc(x) | Complementary error function $\text{erfc}(x) = 1 - \text{erf}(x)$. |
| erfz(x) | The Gaussian probability density function $Z(x)$. |
| erfq(x) | The upper tail of the Gaussian probability function $Q(x)$. |
| exp(x) | Exponential function: e raised to the power of x. |
| gamma(x) | Computes the Gamma function, subject to x not being a negative integer |
| gammaln(x) | Computes the logarithm of the Gamma function, subject to x not a being negative integer. For $x<0$, $\log(\Gamma(x))$ is returned. |
| hazard(x) | Computes the hazard function for the normal distribution $h(x) = \text{erfz}(x)/\text{erfq}(x)$. |
| ln(x) | natural logarythm of x |
| log(x) | decimal logarythm of x |
| log2(x) | base 2 logarythm of x |
| min(x1,x2,x3,...,x5) | Minimum of the list of arguments |
| max(x1,x2,x3,...,x5) | Maximum of the list of arguments |
| rint(x) | Round to nearest integer. |
| sign(x) | Sign function: -1 if $x<0$; 1 if $x>0$. |
| sin(x) | sinus of x |
| sinh(x) | hyperbolic sinus of x |
| sqrt(x) | square root of x |
| tan(x) | tangent of x |
| tanh(x) | hyperbolic tangent of x |

Table 7.2: Mathematical Functions

| Name | Description |
|----------------------------|---|
| <code>cell(a,b)</code> | In the context of a matrix, returns the value at row a and column b. In the context of a table, returns the value at column a and row b (remember that tables use column logic). Everywhere else, this function is undefined. |
| <code>col(c)</code> | Only works in the context of a table. Returns the value at column c and row i (the current row) in the context table. c can either be the column's number, or its name in doublequotes. |
| <code>if(e1,e2,e3)</code> | If e1 is true, e2 is executed else e3 is executed. |
| <code>tablecol(t,c)</code> | Only works in the context of a table. Returns the value at column c and row i (the current row) in the table t. t is the table's name in doublequotes, c is either the column's number or its name in doublequotes. |

Table 7.3: Non-Mathematical Functions

7.2 Python

This module provides bindings to the **Python** programming language. Basic usage in the context of SciDAVis will be discussed below, but for more in-depth information on the language itself, please refer to its excellent [documentation](#).

7.2.1 Getting Started

Make sure your current project uses Python as its interpreter by selecting the menu point Scripting->Scripting Language and double-clicking on "Python" in the resulting dialog (if the dialog appears, but does not contain the "Python" item, your installation of SciDAVis has been compiled without Python support). Next, open a Notes window and enter `print "Hello World!"`. Position the cursor anywhere on this line and press Ctrl+J, or select "Execute" from the context menu. The string "Hello World!" should appear below the line you entered. Congratulations, you've made contact with Python! You can also enter more complex code fragments, such as function or class definitions, in which case you have to select the whole code block before executing it.

You can also use Notes windows as a handy calculator. Enter a mathematical expression on a line of its own - say, `5*sin(pi/2)`. Press Ctrl+Enter, or select "Evaluate" from the context menu. You are rewarded by a new line, stating (to general astonishment), that the result of evaluating your expression is `#> 5`. If you have **SciPy** and/or **PyGSL** installed, you will have immediate access to a huge number of interesting functions, browsable via the submenu "Functions" of the context menu. Pressing Shift+F1 while in this menu will give you a short description of the current function. The advantage of only executing/evaluating single lines or selections on request is that you can freely mix text and calculations within a Notes window.

Another particularly handy place for using Python code are column formulas. These work just like evaluating expressions in Note windows, with the additional advantage of some pre-defined variables: i (the row currently being computed), j (the column number), sr (start row), er (end row) and self (the table to which the column being

computed belongs; see below for what you can do with it).

If you are already familiar with Python, you might ask yourself at this point if you can use more complicated column formulas than just single expressions (for instance, if:/else: decisions based on the current row number). The answer is: yes, you can. For the benefit of those not familiar with Python, we will give a short introduction to the language before discussing how to do this.

7.2.2 Python Basics

7.2.2.1 Expressions

Mathematical expressions work largely as expected. However, there's one caveat, especially when switching from muParser: a^b does not mean "raise a to the power of b " but rather "bitwise exclusive or of a and b "; Python's power operator is `**`. Thus:

```
2^3 # read: 10 xor 11 = 01
#> 1
2**3
#> 8
```

7.2.2.2 Statements

One thing you have to know when working with Python is that indentation is very important. It is used for grouping (most other languages use either braces or keywords like `do . . . end` for this). For example, executing the following:

```
x=23
for i in (1,4,5):
    x=i**2
    print(x)
```

will do what you would expect: it prints out the numbers 1, 16 and 25; each on a line of its own. Deleting just a bit of space will change the functionality of your program:

```
x=23
for i in (1,4,5):
    x=i**2
print(x)
```

will print out only one number - no, not 23, but rather 25. This example was designed to also teach you something about variable scoping: There are no block-local variables in Python.

There are two different variable scopes to be aware of: local and global variables. Unless specified otherwise, variables are local to the context in which they were defined. Thus, the variable `x` can have three different values in, say, two different Note windows and a column formula. Global variables on the other hand can be accessed from everywhere within your project. A variable `x` is declared global by executing the statement `global x`. You have to do this before assigning a value to `x`, but you have

to do it only once within the project (no need to "import" the variable before using it). Note that there is a slight twist to these rules when you define your own functions.

The basic syntax for defining a function (for use within one particular note, for example) is

```
def answer():  
    return 42
```

If you want your function to be accessible from the rest of your project, you have to declare it global before the definition:

```
global answer  
def answer():  
    return 42
```

You can add your own function to SciDAVis's function list. We'll also provide a documentation string that will show up, for example, in the "set column values" dialog:

```
global answer  
def answer():  
    "Return the answer to the ultimate ↔  
    question about life, the ↔  
    universe and everything."  
    return 42  
sci.mathFunctions["answer"] = answer
```

If you want to remove a function from the list, execute

```
del sci.mathFunctions["answer"]
```

Note that functions have their own local scope. That means that if you enter a function definition in a Notes window, you will not be able to access (neither reading nor writing) Notes-local variables from within the function. However, you can access global variables as usual.

If-then-else decisions are entered as follows:

```
if x>23:  
    print(x)  
else:  
    print("The value is too small.")
```

You can do loops, too:

```
for i in range(1, 11):  
    print(i)
```

This will print out the numbers between 1 and 10 inclusively (the upper limit does not belong to the range, while the lower limit does).

7.2.3 Evaluation Reloaded

As we've already mentioned above, there's a bit more to evaluation than just expressions. Note that Python itself can indeed only evaluate expressions; the following describes a feature of SciDAVis added on top of Python in order to support more interesting column formulas.

If you use statements (e.g. variable assignments or control structures) in a formula, SciDAVis will assume it to be the body of a function. That is, the following code will not work:

```
a=1
a+a
```

The statement in the first line means that the formula cannot be evaluated as a single expression. Instead, the above code assigns every row in the column the return value of the following function:

```
def f():
    a=1
    a+a
```

However, since Python does not implicitly interpret expressions as something to return, this evaluates to nothing. The correct way to enter formulas with statements in them is to explicitly return something:

```
a=1
return a+a
```

There is a slight catch associated with this strategy. In a Notes window, SciDAVis will allow you to *evaluate* variable assignments like `ee=1.6021773e-19` without complaining - but this will not lead to the result presumably intended, i.e. introducing a shortcut for the elementary charge to be used within the notes window. What actually happens is this: SciDAVis evaluates the function

```
def f():
    ee=1.6021773e-19
```

As mentioned in the Python introduction above, the function `f` has its own variable scope and (unless it happens to be declared global) the variable `ee` will only be visible within this scope (instead of the Notes window's scope). The solution is simple: always *execute* things like assignments and function definitions, never *evaluate* them.

7.2.4 Mathematical Functions

Python comes with some basic mathematical functions that are automatically imported (if you use the [initialization file](#) shipped with SciDAVis). Along with them, the constants `e` (Euler's number) and `pi` (the one and only) are defined. Many, many more functions can be obtained by installing [SciPy](#) and/or [PyGSL](#).

| Name | Description |
|------------|---|
| acos(x) | inverse cosinus |
| asin(x) | inverse sinus |
| atan(x) | inverse tangent |
| atan2(y,x) | equivalent to atan(y/x), but more efficient |
| ceil(x) | ceiling; smallest integer greater or equal to x |
| cos(x) | cosinus of x |
| cosh(x) | hyperbolic cosinus of x |
| degrees(x) | convert angle from radians to degrees |
| exp(x) | Exponential function: e raised to the power of x. |
| fabs(x) | absolute value of x |
| floor(x) | largest integer smaller or equal to x |
| fmod(x,y) | remainder of integer division x/y |
| frexp(x) | Returns the tuple (mantissa,exponent) such that $x = \text{mantissa} * (2^{**}\text{exponent})$ where exponent is an integer and $0.5 \leq \text{abs(m)} < 1.0$ |
| hypot(x,y) | equivalent to $\sqrt{x^2 + y^2}$ |
| ldexp(x,y) | equivalent to $x * (2^{**}y)$ |
| log(x) | natural (base e) logarythm of x |
| log10(x) | decimal (base 10) logarythm of x |
| modf(x) | return fractional and integer part of x as a tuple |
| pow(x,y) | x to the power of y; equivalent to $x^{**}y$ |
| radians(x) | convert angle from degrees to radians |
| sin(x) | sinus of x |
| sinh(x) | hyperbolic sinus of x |
| sqrt(x) | square root of x |
| tan(x) | tangent of x |
| tanh(x) | hyperbolic tangent of x |

Table 7.4: Supported Mathematical Functions

7.2.5 Accessing SciDAVis's functions from Python

We will assume that you are using the [initialization file](#) shipped with SciDAVis.

7.2.5.1 Establishing contact

Accessing the objects in your project is straight-forward,

```
t = table("Table1")
m = matrix("Matrix1")
g = graph("Graph1")
n = note("Notes1")
```

as is creating new objects:

```
# create an empty table named "tony" ←
    with 5 rows and 2 columns:
t = newTable("tony", 5, 2)
# use defaults
t = newTable()
# create an empty matrix named "gina" ←
    " with 42 rows and 23 columns:
m = newMatrix("gina", 42, 23)
# use defaults
m = newMatrix()
# create an empty graph window
g = newGraph()
# create an empty note named "momo"
n = note("momo")
# use defaults
n = note()
```

New objects will always be added to the active folder. The functions table, matrix, graph and note will start searching in the active folder and, failing this, continue with a depth-first recursive search of the project's root folder. In order to access other folders, there are the functions

```
f = activeFolder()
# and
f = rootFolder()
```

which can be used to access subfolders and windows:

```
f2 = f.folders()[number]
f2 = f.folder(name, caseSensitive= ←
    True, partialMatch=False)
t = f.table(name, recursive=False)
m = f.matrix(name, recursive=False)
g = f.graph(name, recursive=False)
n = f.note(name, recursive=False)
```

If you supply `True` for the recursive argument, a depth-first recursive search of all subfolders will be performed and the first match returned.

Also, every piece of code is executed in the context of an object which you can access via the `self` variable. For example, entering `self.cell("t", i)` as a column formula is equivalent to the convenience function `col("t")`.

7.2.5.2 Working with Tables

We'll assume that you have assigned some table to the variable `t`. You can access its numeric cell values with

```
t.cell(col, row)
# and
t.setCell(col, row, value)
```

Whenever you have to specify a column, you can use either the column name (as a string) or the consecutive column number (starting with 1). Row numbers also start with 1, just as they are displayed. If you want to work with arbitrary texts or the textual representations of numeric values, you can use

```
t.text(col, row)
# and
t.setText(col, row, string)
```

The number of columns and rows is accessed via

```
t.numRows()
t.numCols()
t.setNumRows(number)
t.setNumCols(number)
```

Column names can be read and written with

```
t.colName(number)
t.setColName(col, newName)
```

Normalize a single or all columns:

```
t.normalize(col)
t.normalize()
```

Import values from file, using `sep` as separator char and ignoring `ignore` lines at the head of the file. The flags should be self-explanatory.

```
t.importASCII(file, sep="\t", ignore ↵
              =0, renameCols=False, ↵
              stripSpaces=True, simplifySpace= ↵
              False, newTable=False)
```

After having changed some table values from a script, you will likely want to update dependent Graphs:


```
t.notifyChanges()
```

As a simple example, let's set some column values without using the dialog.

```
t = table("table1")
for i in range(1, t.numRows()+1):
    t.setCell(1, i, i**2)
t.notifyChanges()
```

7.2.5.3 Working with Matrices

We'll assume that you have assigned some matrix to the variable `m`. Accessing cell values is very similar to `Table`, but since `Matrix` doesn't use column logic, row arguments are specified before columns and obviously you can't use column name.

```
m.cell(row, col)
m.setCell(row, col, value)
m.text(row, col)
m.setText(row, col, string)
```

Also like with tables, there's

```
m.numRows()
# and
m.numCols()
```

7.2.5.4 Plotting and Working with Graphs

If you want to create a new `Graph` window for some data in table `table1`, you can use the `plot` command:

```
g = plot(table, column, type)
```

`type` is a number between 0 and 10 and specifies the desired plot type:

0. Line
1. Symbols
2. Line and Symbols
3. Columns
4. Area
5. Pie
6. Vertical drop lines
7. Splines and Symbols

8. Vertical steps

9. Histogram

10. Rows

You can plot more than one column at once by giving a Python tuple (see the [Python Tutorial](#)) as an argument:

```
g = plot(table("table1"), (2,4,7), ←  
    2)
```

If you want to add a curve to an existing Graph window, you have to choose the destination layer. Usually,

```
l = g.activeLayer()
```

will do the trick, but you can also select a layer by its number:

```
l = g.layer(num)
```

You can then add or remove curves to or from this layer:

```
l.insertCurve(table, column, type=1)  
l.insertCurve(table, Xcolumn, ←  
    Ycolumn, type=1)  
l.removeCurve(curveName)  
l.removeCurve(curveNumber)  
l.deleteFitCurves()
```

In case you need the number of curves on a layer, you can get it with

```
l.numCurves()
```

Layers and whole Graphs can be printed and exported from within Python. Before you do this, you would probably want to change layer and axis titles as well as legend texts:

```
l.setTitle(title)  
l.setXTitle(Xtitle)  
l.setYTitle(Ytitle)  
l.setLegend(text)
```

You can also customize the scales of the different axes using:

```
l.setScale(int axis, double start, ←  
    double end, double step=0.0, int ←  
    majorTicks=5, int minorTicks=5, ←  
    int type=0, bool inverted=False ←  
    );
```

where axis is a number between 0 and 3 with the following signification:

0. Left axis

1. Right axis
2. Bottom axis
3. Top axis

`type` specifies the desired scale type:

0. Linear
1. Log10

and `step` defines the size of the interval between the major scale ticks. If not specified (default value is 0.0), the step size is calculated automatically. The other flags should be self-explanatory. Now, here is how you can export a layer

```
l.print()
l.exportToSVG(filename)
l.exportToEPS(filename)
l.exportImage(filename, filetype=" ←
    PNG", quality=100, transparent= ←
    False)
```

and a graph

```
g.print()
g.exportToSVG(filename)
g.exportToEPS(filename)
```

7.2.5.5 Fitting

Assuming you have a Graph named "graph1" with a curve entitled "table1_2" (on its active layer), a minimal Fit example would be:

```
f = GaussFit(graph("graph1"). ←
    activeLayer(), "table1_2")
f.guessInitialValues()
f.fit()
```

This creates a new `GaussFit` object on the curve, lets it guess the start parameters and does the fit. The following fit types are supported:

- `LinearFit(layer, curve)`
- `PolynomialFit(layer, curve, degree=2, legend=False)`
- `ExponentialFit(layer, curve, growth=False)`
- `TwoExpFit(layer, curve)`
- `ThreeExpFit(layer, curve)`
- `GaussFit(layer, curve)`

- GaussAmpFit(layer, curve)
- LorentzFit(layer, curve)
- SigmoidalFit(layer, curve)
- NonLinearFit(layer, curve)

```
f = NonLinearFit(layer, curve)
f.setParameters(name1, ...)
f.setFormula(formula_string)
```

- PluginFit(layer, curve)

```
f = PluginFit(layer, curve)
f.load(pluginName)
```

For each of these, you can optionally restrict the X range that will be used for the fit, like in

```
f = LinearFit(graph("graph1"). ←
    activeLayer(), "table1_2", 2, 7)
f.fit()
```

After creating the Fit object and before calling its fit() method, you can set a number of parameters that influence the fit:

```
f.setDataFromCurve(curve)      change data source
f.setDataFromCurve(curve, graph) change data source
f.setDataFromCurve(curve, from, to) change data source
f.setDataFromCurve(curve, from, to, graph) change data source
f.setInterval(from, to)        change data range
f.setInitialValue(number, value)
f.setInitialValues(value1, ...)
f.guessInitialValues()
f.setAlgorithm(algo) # algo = Fit.ScaledLevenbergMarquardt, Fit. ←
    UnscaledLevenbergMarquardt, Fit.NelderMeadSimplex
f.setWeightingData(method, colname) # method = Fit.NoWeighting, Fit. ←
    Instrumental, Fit.Statistical, Fit.Dataset
f.setTolerance(tolerance)
f.setOutputPrecision(precision)
f.setMaximumIterations(number)
f.scaleErrors(yes = True)
f.setColor(qt.QColor("green")) change the color of the result fit ←
    curve to green (default color is red)
```

After you've called fit(), you have a number of possibilities for extracting the results:

```
f.results()
f.errors()
```

```
f.chiSquare()  
f.rSquare()  
f.dataSize()  
f.numParameters()  
f.parametersTable("params")  
f.covarianceMatrix("cov")
```

7.2.6 The Initialization File

This file allows you to customize the Python environment, import modules and define functions and classes that will be available in all of your projects. The default initialization file shipped with SciDAVis imports Python's [standard math functions](#) as well as special functions from [SciPy](#) and [PyGSL](#)(if available). Also, it creates some handy shortcuts, like `table("table1")` for `sci.app.table("table1")`.

When activating Python support, SciDAVis searches the following places, executing the first file it can find:

1. `~/scidavsrc.py[c]`
2. `/etc/scidavsrc.py[c]`
3. `./scidavsrc.py[c]`

Files ending in `.pyc` are compiled versions of the `.py` source files and therefore load a bit faster. The compiled version will be used if the source file is older or nonexistent. Otherwise, SciDAVis will try to compile the source file (if you've got write permissions for the output file).

Appendix A

Appendix

A.1 Credits and License

SciDAVis

Program copyright:

2004-2007 Ion Vasilief ion_vasilief@yahoo.fr

2006-2007 Tilman Hoener zu Siederdisen thzs@gmx.net

2006-2007 Knut Franke Knut.Franke@gmx.de

Documentation copyright:

2004-2007 Ion Vasilief ion_vasilief@yahoo.fr

2006-2007 Roger Gadiou Roger.Gadiou@uha.fr

2006-2007 Knut Franke Knut.Franke@gmx.de

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.1 or any later version published by the Free Software Foundation; with no Invariant Sections, with no Front-Cover Texts, and with no Back-Cover Texts.

A.1.1 GNU Free Documentation License

Version 1.1, March 2000

Copyright (C) 2000 Free Software Foundation, Inc. 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

A.1.1.1 Preamble

The purpose of this License is to make a manual, textbook, or other written document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

A.1.1.2 Applicability And Definitions

This License applies to any manual or other work that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you".

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (For example, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, whose contents can be viewed and edited directly and straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup has been designed to thwart or discourage subsequent modification by readers is not Transparent. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML designed for human modification. Opaque formats include PostScript, PDF, proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD

and/or processing tools are not generally available, and the machine-generated HTML produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

A.1.1.3 Verbatim Copying

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

A.1.1.4 Copying In Quantity

If you publish printed copies of the Document numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a publicly-accessible computer-network location containing a complete Transparent copy of the Document, free of added material, which the general network-using public has access to download anonymously at no charge using public-standard network protocols. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

A.1.1.5 Modifications

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has less than five). State on the Title page the name of the publisher of the Modified Version, as the publisher. Preserve all the copyright notices of the Document.

Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

Include an unaltered copy of this License.

Preserve the section entitled "History", and its title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

In any section entitled "Acknowledgements" or "Dedications", preserve the section's title, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles. Delete any section entitled "Endorsements". Such a section may not be included in the Modified Version.

Do not retitle any existing section as "Endorsements" or to conflict in title with any Invariant Section.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

A.1.1.6 Combining Documents

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections entitled "History" in the various original documents, forming one section entitled "History"; likewise combine any sections entitled "Acknowledgements", and any sections entitled "Dedications". You must delete all sections entitled "Endorsements."

A.1.1.7 Collections Of Documents

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various

documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

A.1.1.8 Aggregation With Independent Works

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, does not as a whole count as a Modified Version of the Document, provided no compilation copyright is claimed for the compilation. Such a compilation is called an "aggregate", and this License does not apply to the other self-contained works thus compiled with the Document, on account of their being thus compiled, if they are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one quarter of the entire aggregate, the Document's Cover Texts may be placed on covers that surround only the Document within the aggregate. Otherwise they must appear on covers around the whole aggregate.

A.1.1.9 Translation

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License provided that you also include the original English version of this License. In case of a disagreement between the translation and the original English version of this License, the original English version will prevail.

A.1.1.10 Termination

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

A.1.1.11 Future Revisions Of This License

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to

the present version, but may differ in detail to address new problems or concerns. See <http://www.gnu.org/copyleft/>. Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by

A.2 How to obtain SciDAVis

The SciDAVis home page can be found at <http://sourceforge.net/project/scidavis>. Updates and news can be found there.

SciDAVis is distributed as a package with sources which have to be compiled. Compiled packages suitable for most Linux distributions and Windows can also be obtained. If you want to build SciDAVis from sources, don't forget first to download and install the [Qt](#), [QwtPlot3D](#), [Qwt](#), [liborigin](#), [zlib](#), [muParser](#), and [GSL](#) libraries.

A.3 Requirements

In order to successfully use SciDAVis, you need the following libraries:

Qt >= 4.2 SciDAVis uses the [Qt](#) toolkit. Version 4.2 or above is needed. It provides the necessary dynamic libraries to run the SciDAVis binaries and important tools to compile it.

Qwt You need to install the [Qwt library](#) version 5.0.0 or above. It is recommended to use the latest release. Qwt must be compiled with Qt 4.x.

muParser You also need the [muParser](#) library 1.28 or later.

liborigin The latest version of the [liborigin](#) is required for the import of Origin OPJ project files.

Qwtplot3d The 3D plots in SciDAVis make use of [qwtplot3d](#). The library and must be available in version 0.2.6 or higher.

GSL Furthermore, [the GNU Scientific Library \(GSL\)](#) (1.8 or later) must be installed on your system.

Zlib Additionally, [zlib](#) >=1.2.3 is required. Since this is also a requirement of Qt, installing Qt should fulfill this requirement on most systems.

Python scripting If you want to use Python expressions and scripts, make sure you have the following additional dependencies installed: [Python](#) 2.5, [SIP](#) 4.5.2 or 4.6, [PyQt](#) 4.2. Other versions as those indicated above may or may not work.

A.4 Installation from binary packages

SciDAVis distributes binaries which can be downloaded from [the SciDAVis page on Sourceforge](#). Refer to the installation instructions and requirements there. Typically, you need at least Qt and Python installed on your system.

A.5 Compilation and Installation from sources

To compile SciDAVis from sources, download the latest source distribution from [the SciDAVis page on Sourceforge](#). It comes as a zip or tar.gz archive containing detailed building instructions. If you are interested in the current development, you can also obtain a snapshot from the [SciDAVis SVN repository](#).

Appendix B

Frequently asked questions

Q: *How can I visualise data from a text file?*

A: Go to the [File menu](#) and select the [Import ASCII -> Single File... command](#) command. If the file is not imported correctly, change the columns separator using the [Set Import Options... command](#) options command from the [File menu](#). The default columns separator is the TAB.

Q: *How can I plot data from a table (worksheet)?*

A: Click on the table header to choose the columns to plot and then right click. Chose the 'Plot' option from the pop-up menu and then the type of plot you want. You can also use the plot assistant: press 'CTRL+ALT+W' keys to show it, or go to 'View' menu -> 'Plot wizard'.

Q: *How can I export a plot to an image format?*

A: Right click in the plot window and chose the 'Export' option.

Q: *Can I export transparent images?*

A: Yes, ".png" images have transparent background. See the [Export Graph -> Current command](#) command.

Q: *How can I export a text file?*

A: Go to the [File menu](#) and select the [Export ASCII command](#) command.

Q: *How can I choose a window using the project explorer?*

A: Double click on the window name will show the window maximized, even if it was hidden before.

Q: *How can I choose the data range from a plot curve, when doing a curve fit?*

A: Go to the [Data menu](#) and use the [Select Data Range command](#) command. Click in the plot window and use the 'Up' and 'Down' arrows keys to select the curve to analyse. Keeping 'CTRL' button and 'Left' or 'Right' arrow keys simultaneously pressed permit to move the selected cursor and consequently to modify the data range.

Q: *Can I fit a plot curve using my own function?*

A: Go to the [Analysis menu](#) and select the [Non Linear Curve Fit... command](#) command. Define the function (myFunction=...), enter the initial guesses for the parameters, separated by comas, choose the fitting range and the number of iterations and click 'OK'

Q: *How can I visualize a pixel line profile from an image?*

A: Right click on the image you want to analyse and select the option 'View pixel line profile' from the popup menu. A dialog window opens and allows you to select the number of pixels used for the analysis. Choose a value and click "OK". Then click on the image to select the start point and move your mouse to select an end point while keeping the left button pressed. When you release the left button a plot window appears, representing the pixel intensity versus pixel index.

Index

A

Arrows and Lines

Add an arrow/line, [68](#)

C

Curve analysis

Curve filtering, [107](#)

Band pass FFT, [110](#)

Block pass FFT, [111](#)

High pass FFT, [109](#)

Low pass FFT, [108](#)

Curve fitting

Boltzmann function, [104](#)

Gaussian function, [105](#)

line, [102](#)

Lorentz function, [106](#)

Multi peak, [106](#)

Non linear function, [79](#), [101](#)

Polynom, [103](#)

FFT, [78](#), [99](#)

Integration, [79](#)

interpolation, [112](#)

F

Filtering, *see* Curve analysis

M

Matrix, [5](#), [7](#)

Create a new matrix, [24](#)

Dimensions, [92](#)

Fill with a function, [94](#)

Import from file, [93](#)

Properties, [94](#)

Multilayers plot, [20](#)

Add a new layer, [65](#)

Organize the layers, [67](#)

O

Options, [86](#)

P

Plot, [5](#), [11](#)

Add a curve, [65](#)

Change default options, [87](#)

Create a new plot, [24](#)

Create from data, [11](#)

Create from function, [14](#)

Create with the assistant, [84](#)

Curves options, [72](#)

Error bars, [62](#)

Plot a function, [63](#)

Properties, [81](#)

Remove a curve, [65](#)

Printing, [91](#)

Project explorer, [85](#)

S

Scripting, [114](#)

MuParser, [114](#)

Python, [116](#)

Mathematical functions, [119](#)

Surface plot, [16](#)

Change default options, [87](#)

Coutour curves options, [70](#)

Create a new surface plot, [24](#)

Create from data, [19](#)

Create from function, [17](#), [77](#)

Options, [97](#)

Properties, [95](#)

T

Table, [4](#), [6](#)

Columns

Fill with values, [92](#)

Properties, [69](#)

- Create a new table, [24](#)
 - Export to file, [77](#)
 - Import from file, [93](#)
- Table analysis
 - Convolution, [101](#)
 - Correlation function, [100](#)
 - Deconvolution, [101](#)
- Text label
 - Add a text label, [66](#)
 - Properties, [97](#)