

# Introduction to KaleidaGraph

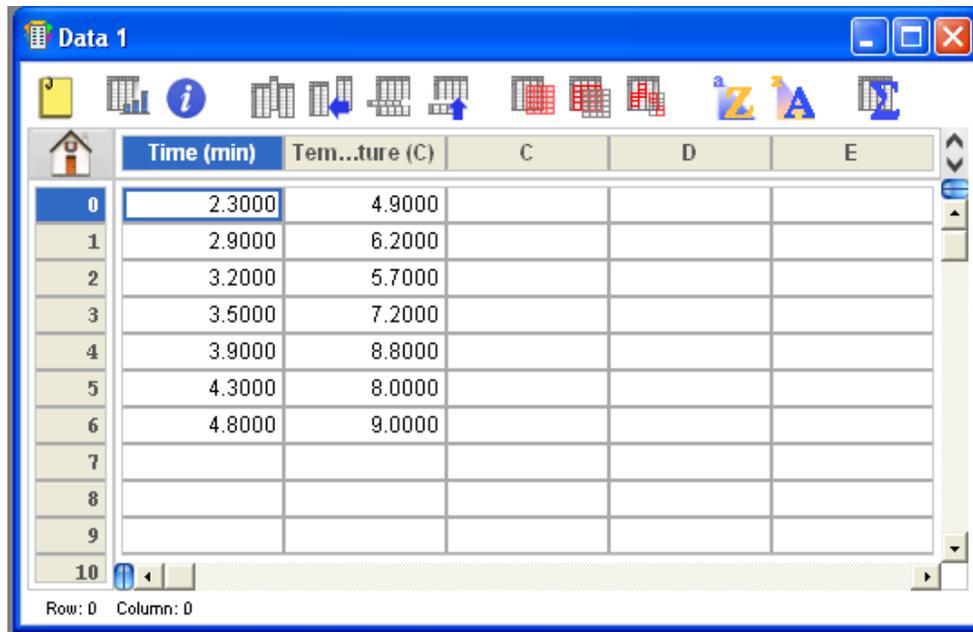
These notes provide an introduction to the software package *Kaleidagraph*. For more details, download the Quick Start Guide from the *Kaleidagraph* web site. [www.synergy.com]

The software includes an extensive **Help** section, which also contains a tutorial for using the software.

## 1. Fitting data to a Straight Line Graph

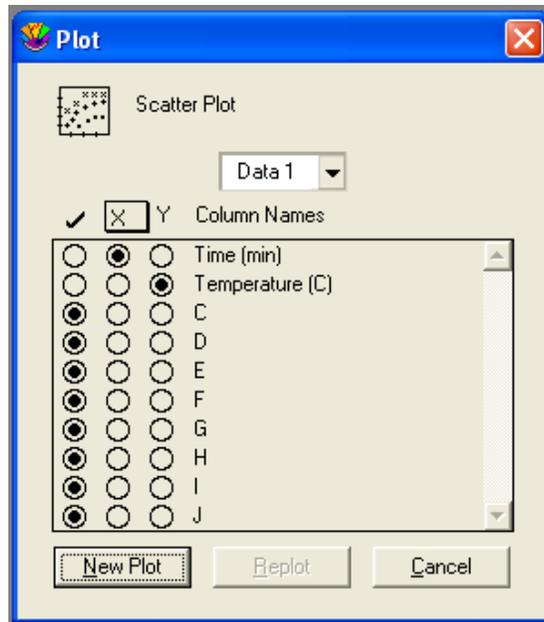
Start *KaleidaGraph* by double-clicking the icon or choosing *KaleidaGraph* from Windows **Start** Menu.

An empty data window will be displayed. Enter each value and then press **Enter**, **Return**, or **Down Arrow** to move to the next cell. You can change the column titles by double-clicking, and replace **A** and **B**, etc., with labels for your experiment such as **Time** and **Temperature**.



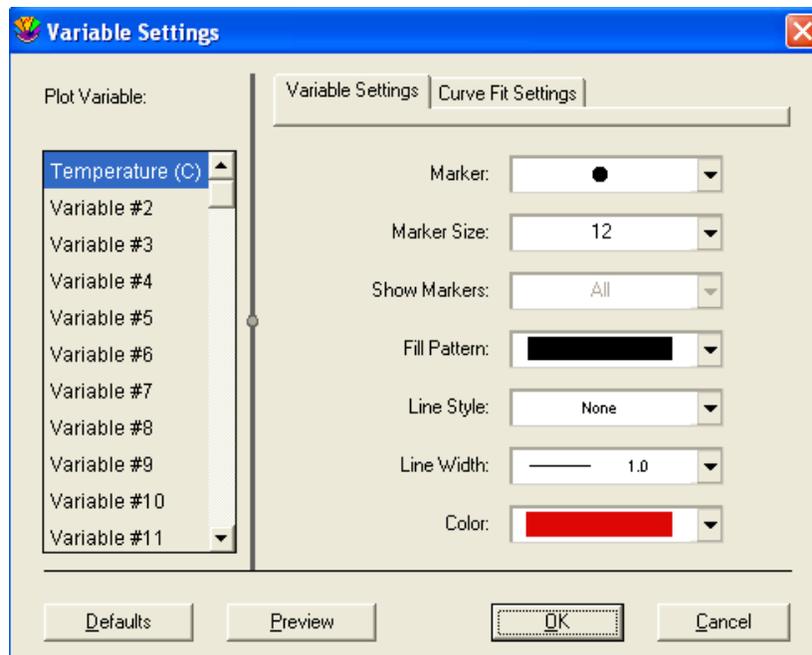
	Time (min)	Tem...ture (C)	C	D	E
0	2.3000	4.9000			
1	2.9000	6.2000			
2	3.2000	5.7000			
3	3.5000	7.2000			
4	3.9000	8.8000			
5	4.3000	8.0000			
6	4.8000	9.0000			
7					
8					
9					
10					

- Choose **Gallery > Linear > Scatter**. This will display the Variable Selection dialogue.
- Select the X variable and the Y variable by clicking the appropriate radio buttons.



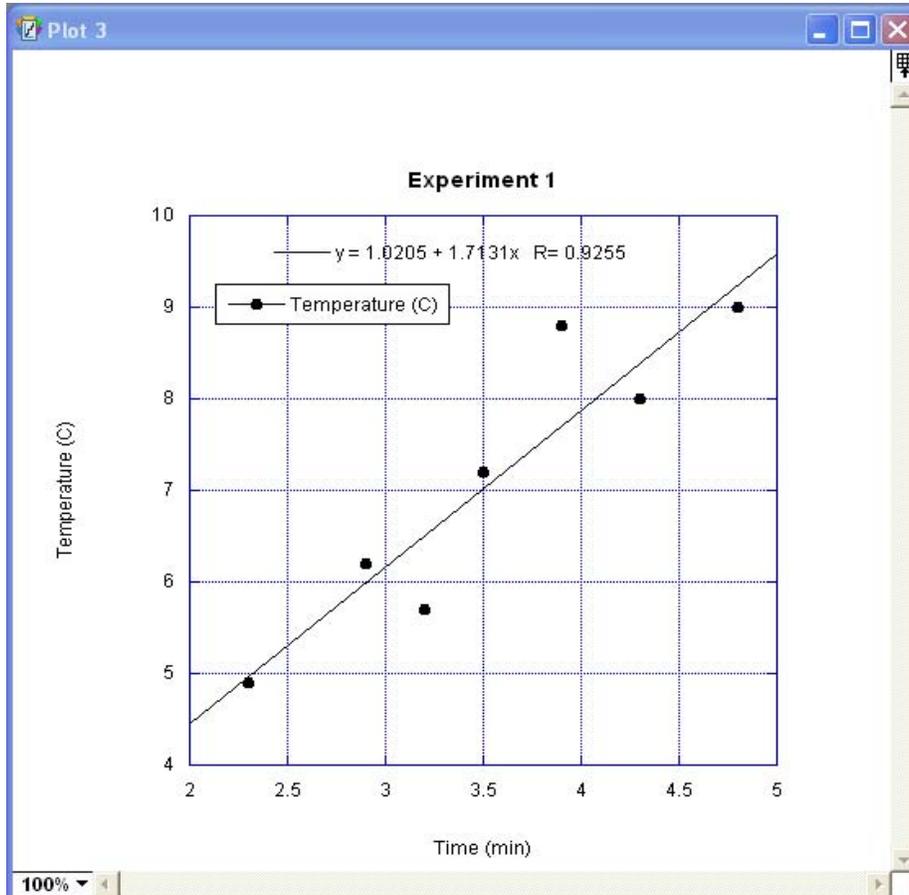
- Click **New Plot**. By default, the X variable is plotted on the horizontal axis (independent variable) and the Y variable is plotted on the vertical axis (dependent variable). The default title of the plot is taken from the name of the data window (Data 1 here, but you can change this). The X and Y axis titles are taken from the column titles of the variables being plotted. The Y variable title is also used in the legend.

The appearance of the graph can be modified using **Plot > Variable Settings**. Choose from **Marker** or **Marker Size**, etc. Click **OK** and the plot will be redrawn to reflect the changes that have been made.



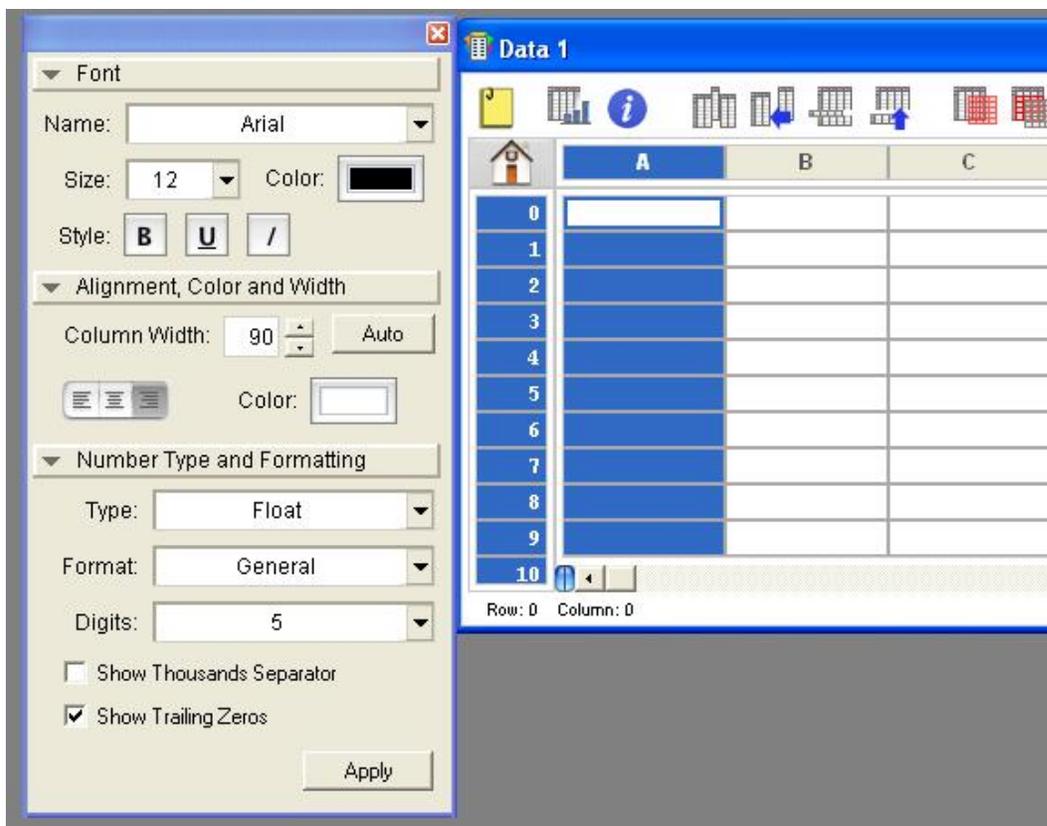
• Choose **Curve Fit >Linear**. This will display a dialogue to select which variables are fitted using the method of Least Squares. Select a variable to be fitted by clicking its check box.

The line of best fit will be drawn and the equation of the line will be displayed. You can extend this line to the axes by choosing **Curve Fit > Curve Fit Options** and selecting **Extrapolate Fit to Axis Limits**. The position of the equation can be changed using the Selection Arrow. You can use the same technique to move the legend. Pick an appropriate title for the graph, and your final plot might look like this:



**Caution:**

There is a possibility that Kaleidagraph will interpret your raw data as text, instead of the numbers you wish to plot. This may result in Kaleidagraph being unable to plot your data. To check that your data is in the correct format, right-click on the column header and select "Column Formatting" to display the following window:



**In this example, Number Type is set to “Float”, which is what we want. If it reads “Text”, simply use the pull-down menu to put it back to “Float”, or one of the other choices, if desired.**

## **2. Regression Statistics**

When fitting data to a straight line,  $y=mx+b$ , we need to know the error in the slope and intercept in order to determine the overall uncertainty in the result from a laboratory experiment.

Select **Curve Fit > General > Fit1 > OK**

This assumes that the function is a straight line. Click the **Define** button. For a straight line you should see something like:

$$M1 + m2 * M0; m1=1, m2=1$$

This tells *Kaleidagraph* to find the optimal values for the slope and intercept, starting with values of 1, adjusting these until the best fit is found. For the above graph, the following is generated:

y = m1 + m2 * M0		
	Value	Error
m1	1.0205	1.1423
m2	1.7131	0.31352
Chisq	2.1414	NA
R	0.9255	NA

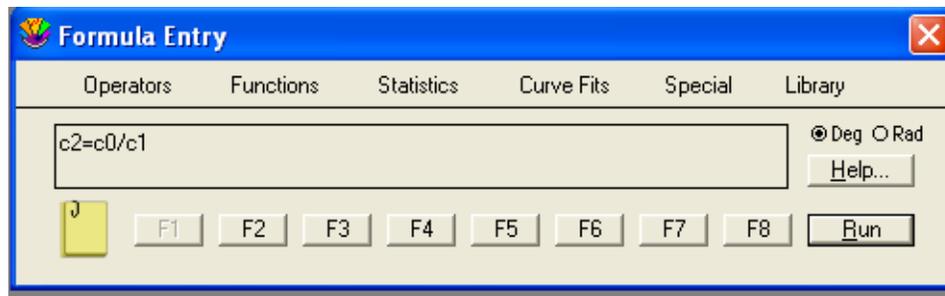
This says that the value of the slope is  $1.7 \pm 0.3$ , while the intercept is  $1.0 \pm 1.1$  (units will be determined from your experiment).

## 2. Editing your data

Sometimes you will want to edit your raw data before plotting. For example, you might want to plot the resistance of a component when your raw data is voltage and current. The data table can be edited just like columns of numbers in a spreadsheet. You can add, multiply, divide, etc., to generate a new column of numbers. In *Kaleidagraph* the columns are labeled c0, c1, c2, and so on. If your data consists of pairs of (x,y) values only, these will most likely be in columns c0 and c1.

Use the **Formula Entry** box. If the box is not visible on the screen select **Window > Formula Entry**, or use the key sequence **Ctrl-F**.

If c0 contains data for the voltage across an electrical component and c1 is the corresponding current, you can create a new column, c2, containing the resistance by typing  $c2=c0/c1$  into the Formula entry box, and click **Run**.



Most mathematical functions are included, such as  $\sin()$  and  $\log()$ . So, to generate a new column containing the natural logarithm of a previous column, type  $c4=\ln(c2)$  into the Formula entry box. For  $\log$  (base 10) use the  $\log()$  function.

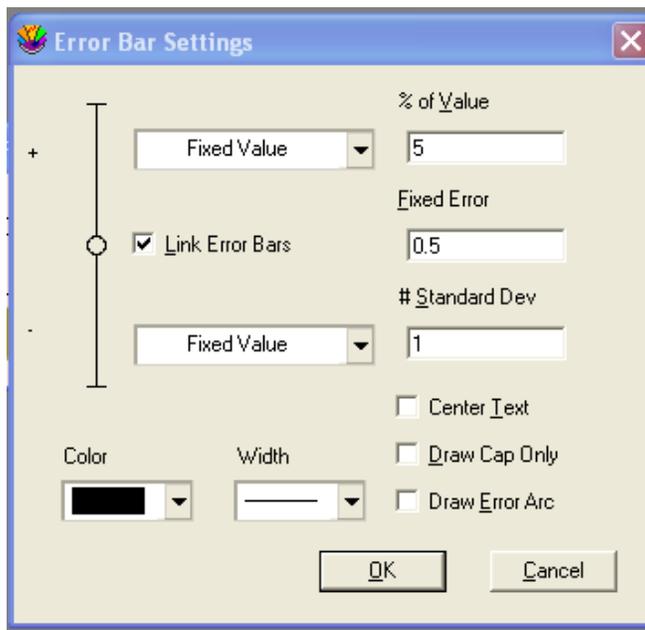
### Removing unwanted data

Data points can be removed from the graph using the MASK feature. Simply use the mouse to highlight the unwanted data, and click on the Mask icon, then **Plot Update**. The

graph will be redrawn with the points removed. Also the equation for the line of best fit will be recalculated for the new data set.

### Error Bars

Select **Plot > Error Bars** and choose Xerr and Yerr as appropriate. Click **Plot** to draw the error bars as a percentage of the value, a fixed value, or numbers read from another data column. The error bar dialogue box will look like this:



Error bars are usually symmetrical, so ensure that the **Link Error Bars** box is checked (✓)

## 4. Fitting Other Mathematical Functions

**Curve Fit** allows you to fit other common mathematical functions to your data. These include:

**Polynomial:** One of the most common functions you will use in the laboratory is a quadratic function, of the form  $ax^2 + bx + c$ . This is a second order polynomial, so to fit this function, select **Curve Fit > Polynomial** and choose Polynomial order of 2

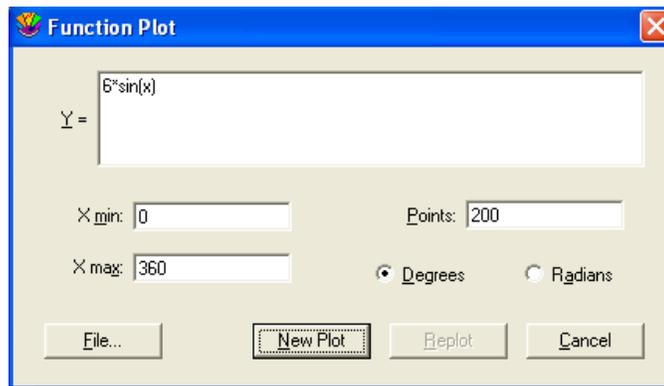
**Power Law Fit :** Data which obeys a simple power law can be described by a mathematical function of the form  $y = Ax^n$ . **Curve Fit > Power** will give values for A and n. Alternately, taking the logarithm of each side gives  $\log(y) = \log(A) + n \log(x)$  and a plot of  $\log(y)$  versus  $\log(x)$  will give a straight line of slope n.

**Exponential Functions :** Exponential functions occur commonly in physics. Use **Curve Fit > Exponential**

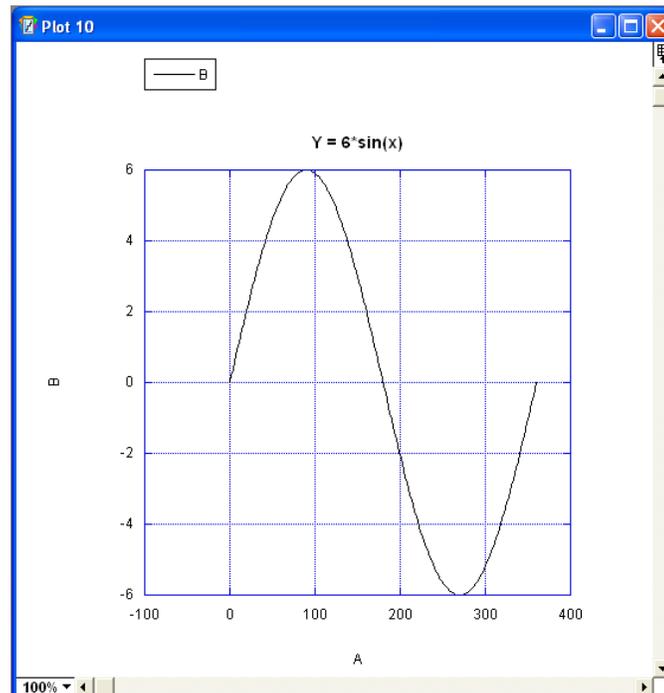
You can define your own functions too. See the *Kaleidgraph* guide for further details.

## 5. Plotting Functions

Choose **Gallery > Function** to display the Function Plot dialog. Enter the function you want to plot. Enter values for the **Xmin** and **Xmax** and click **New Plot** to plot the function. The number of points defaults to 200, but can be changed as required.



Use **Plot > Variable Settings** to control the line style, markers, and all other components of the plot. Normally this will just be a line plot; the *Show Markers* box will display *None*. The figure below is a plot of the function  $y=6 \sin(x)$  between  $x=0$  and  $x=360^\circ$ .



If you want to see the data generated by the function, choose **Plot > Extract Data**. Here is a sample of data from the above graph:

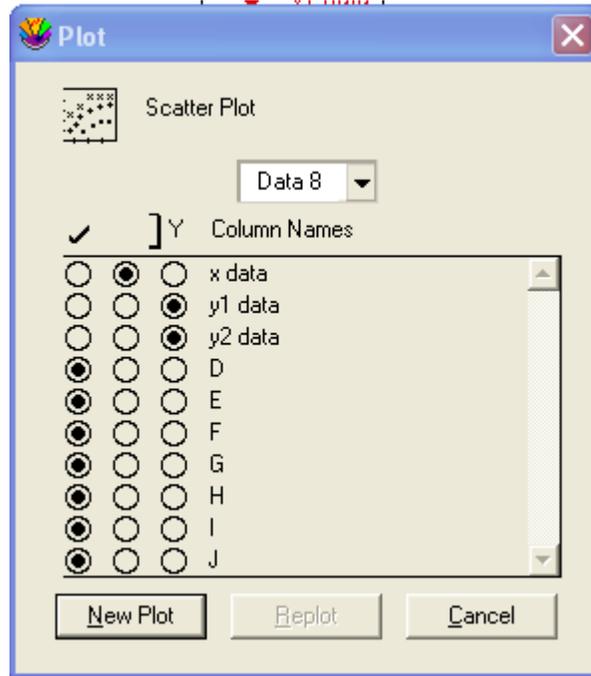
	A	B
0	1.0000	0.10471
1	2.8040	0.29352
2	4.6080	0.48203
3	6.4121	0.67007
4	8.2161	0.85744
5	10.020	1.0440
6	11.824	1.2294
7	13.628	1.4137
8	15.432	1.5966
9	17.236	1.7779

## 6. Plotting Multiple Curves on the same Axes

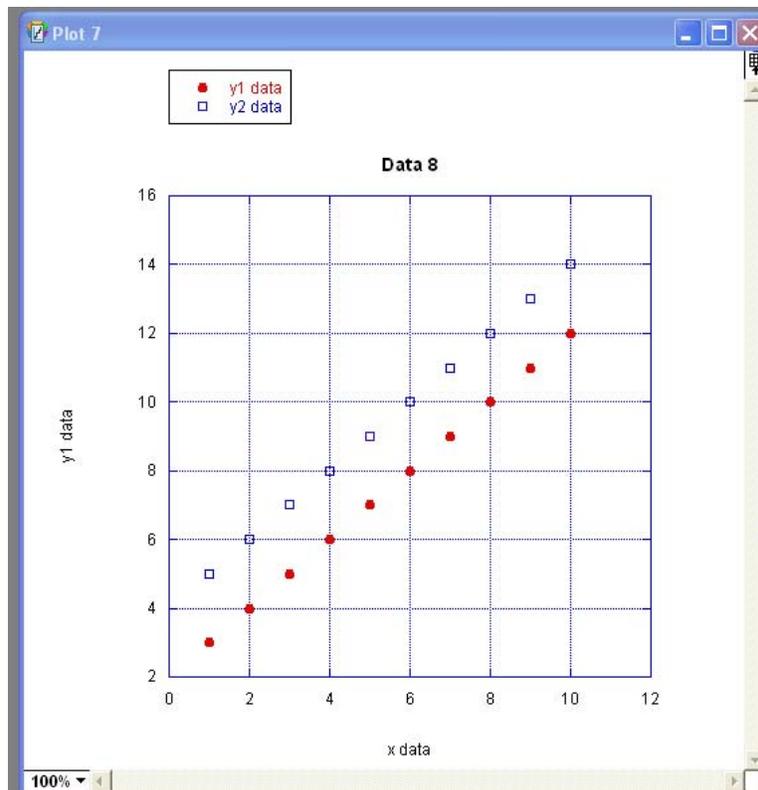
Consider the following data with common x-values, but two different y-values.

	x data	y1 data	y2 data	D	E
0	1.0000	3.0000	5.0000		
1	2.0000	4.0000	6.0000		
2	3.0000	5.0000	7.0000		
3	4.0000	6.0000	8.0000		
4	5.0000	7.0000	9.0000		
5	6.0000	8.0000	10.000		
6	7.0000	9.0000	11.000		
7	8.0000	10.000	12.000		
8	9.0000	11.000	13.000		
9	10.000	12.000	14.000		

Choose the kind of graph you want (for example, **Gallery > Linear > Scatter.**) Select the X variable and the Y variables by clicking the appropriate buttons.



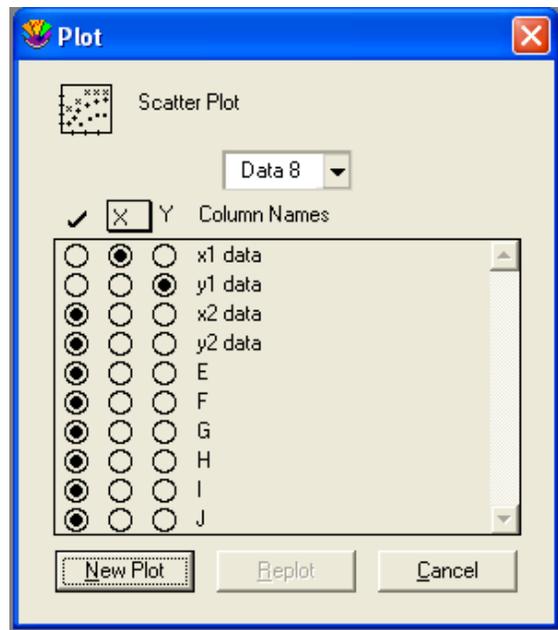
When you have chosen data sets, select **New Plot** to display the graph:-



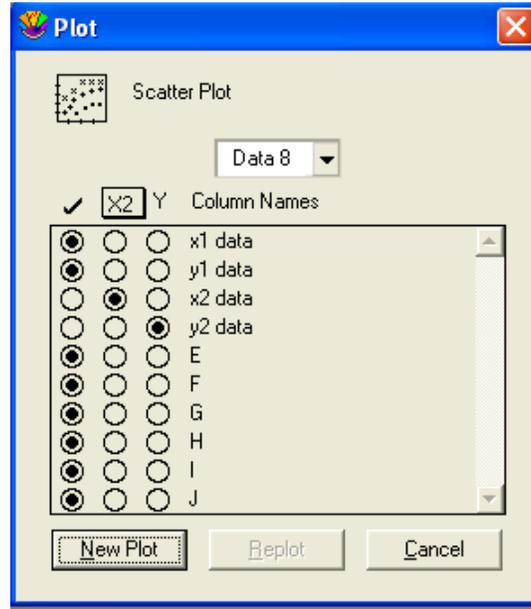
When the data set contains two or more sets of numbers (different pairs of x and y values) such as the following:

	x1 data	y1 data	x2 data	y2 data	E
0	1.0000	5.0000	1.5000	1.0000	
1	2.0000	6.0000	2.5000	2.0000	
2	3.0000	7.0000	3.5000	3.0000	
3	4.0000	8.0000	4.5000	4.0000	
4	5.0000	9.0000	5.5000	5.0000	
5	6.0000	10.000	6.5000	6.0000	
6	7.0000	11.000	7.5000	6.5000	
7	8.0000	12.000	8.5000	6.0000	
8	9.0000	13.000	9.5000	5.0000	
9	10.000	14.000	10.500	4.0000	
10			11.500	3.0000	
11			12.500	2.0000	
12			14.000	1.0000	
13					
14					

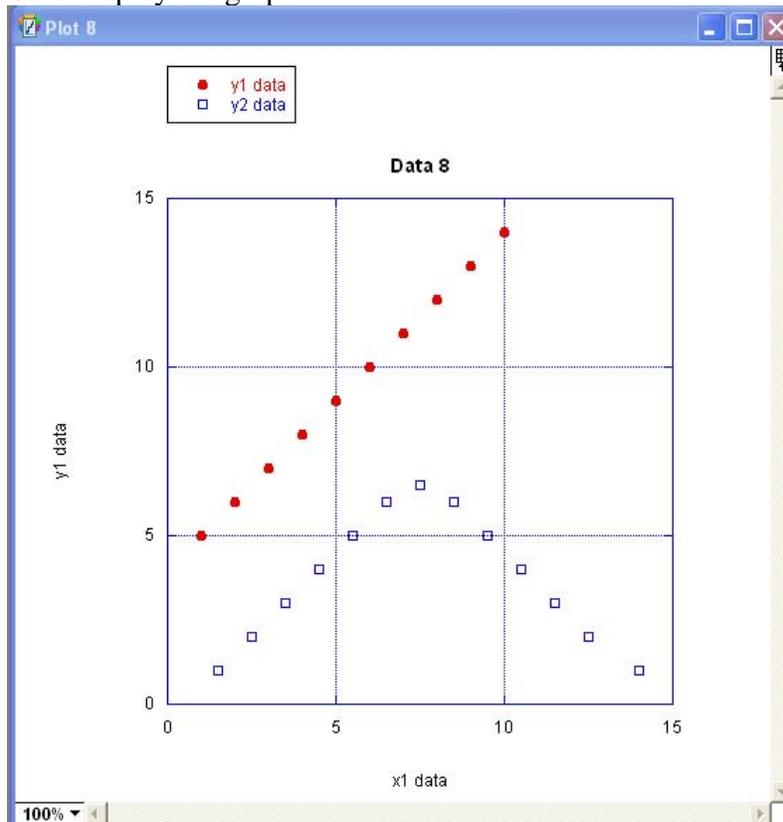
The data sets do not need to contain the same number of points. Choose the kind of graph you want (for example, **Gallery > Linear > Scatter.**) Select the X and Y variables for the first set:



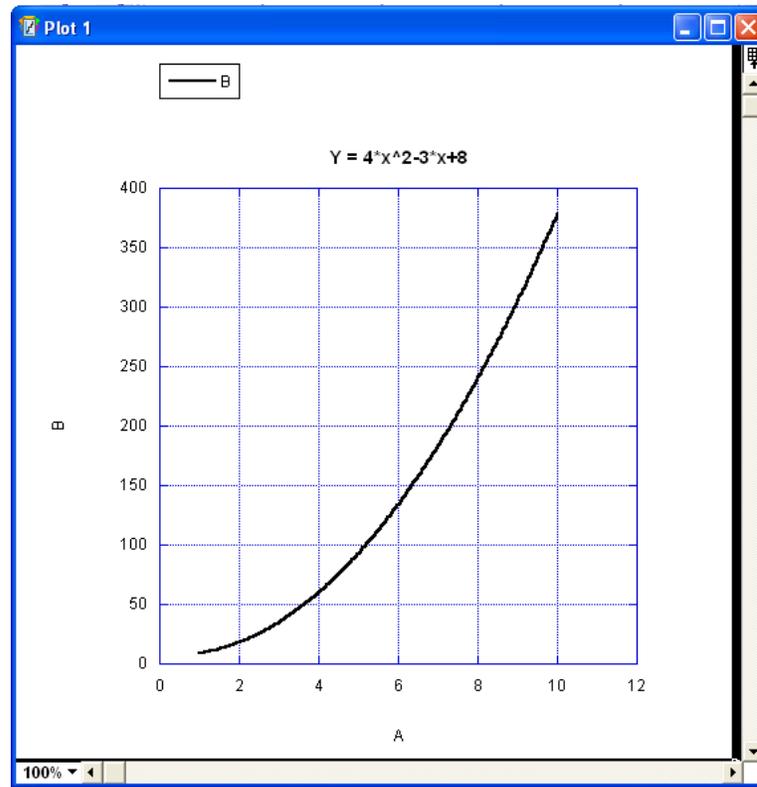
Then select the X and Y variables for the second set by choosing 'X2' from the drop-down menu. (You can select up to nine items from this list.)



Select **New Plot** to display the graph:-



Finally, to plot the data from two or more different data sets, simply select 'Data *nn*' from the plot window. This feature is particularly useful if you want to plot more functions and/or data on the same axes. For example, here is a plot of the function  $y = 4x^2 - 3x + 8$



And here is some data to be plotted on the same axes:

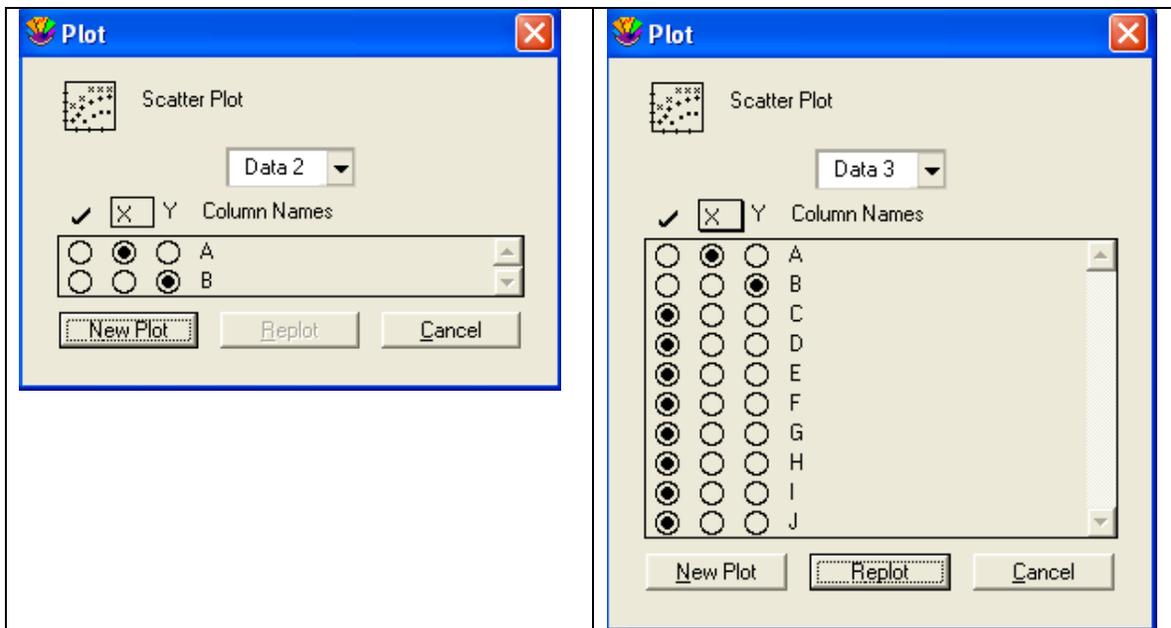
The data window displays a table with columns A, B, C, D, and E. The data points are:

	A	B	C	D	E
0	1.0000	50.000			
1	2.0000	75.000			
2	3.0000	100.00			
3	4.0000	125.00			
4	5.0000	150.00			
5	6.0000	200.00			
6	7.0000	250.00			
7	8.0000	300.00			
8	9.0000	375.00			
9	10.000	450.00			

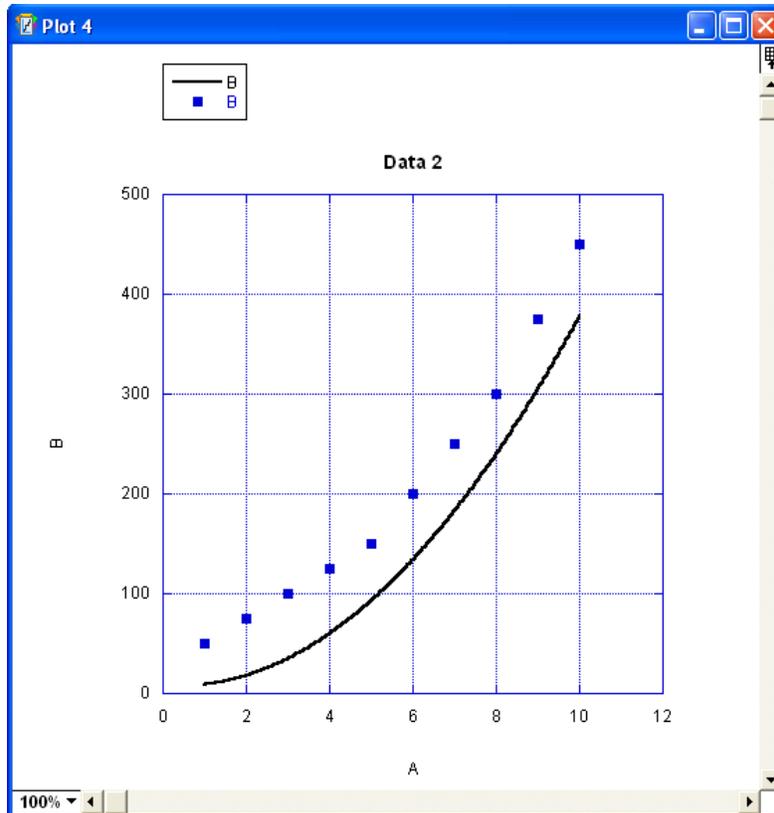
To combine the two plots, we can extract the data from the function plot using **Plot > Extract Data**. Here is a sample of the data.

	A	B
0	1.0000	9.0000
1	1.0452	9.2343
2	1.0905	9.4850
3	1.1357	9.7520
4	1.1809	10.035
5	1.2261	10.335
6	1.2714	10.651
7	1.3166	10.984
8	1.3618	11.333
9	1.4070	11.698
10		

Note that there are now two data sets – Data 2 and Data 3. Select these from the pull-down menu in the Plot window:



Then select **New Plot** or **Replot** to display the graph. Finally use **Plot -> Variable Settings** to adjust marker style or line type as desired. The combined plot is shown below:



## 7. Weighted Fit

When you include error bars on your graph, you expect data points with small errors to be more believable than data points with large errors. Thus you should expect a curve fit to follow the small error bars more closely than the larger ones.

This is known as a **weighted fit**. The weight depends on the error associated with each data point and is proportional to  $1/(\text{error}^2)$ . Thus points with small error bars have a larger weight than points with large error bars.

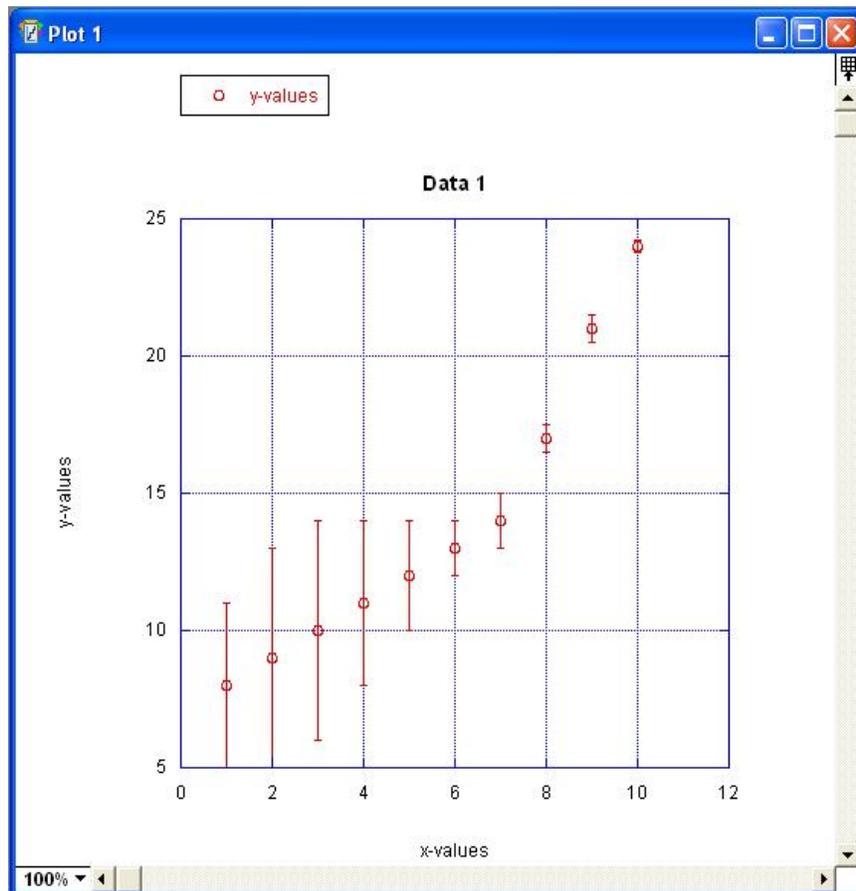
Consider the following data:

Data 1

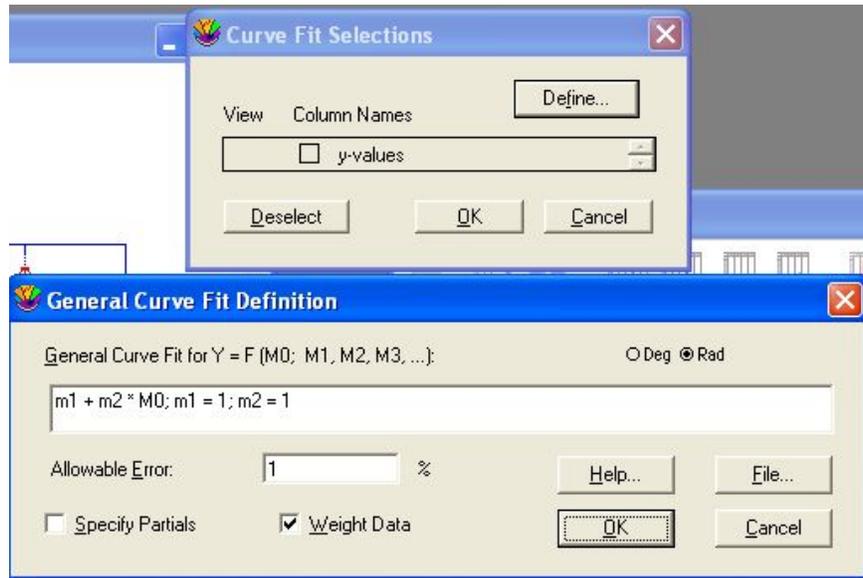
	x-values	y-values	error in y	D	E
0	1.0000	8.0000	3.0000		
1	2.0000	9.0000	4.0000		
2	3.0000	10.0000	4.0000		
3	4.0000	11.0000	3.0000		
4	5.0000	12.0000	2.0000		
5	6.0000	13.0000	1.0000		
6	7.0000	14.0000	1.0000		
7	8.0000	17.0000	0.50000		
8	9.0000	21.0000	0.50000		
9	10.0000	24.0000	0.20000		
10					

Row: 0 Column: 0

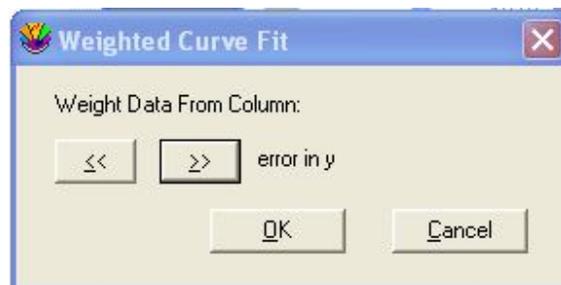
Plotting the data gives the graph shown overleaf. The error bars are drawn using the values in the “error in y” column.



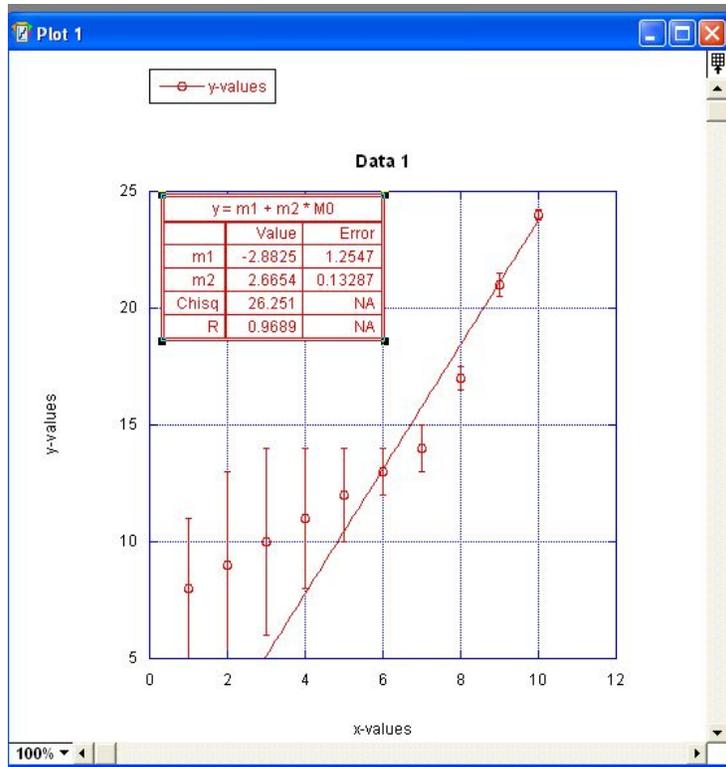
To obtain a weighted fit, select *Curve Fit* → *General*, followed by one of the built-in functions, or a function that you have defined yourself. In the *Curve Fit Selections* box click *Define*. Then in the *General Curve Fit Definition* box, check the *Weight Data* box ( ).



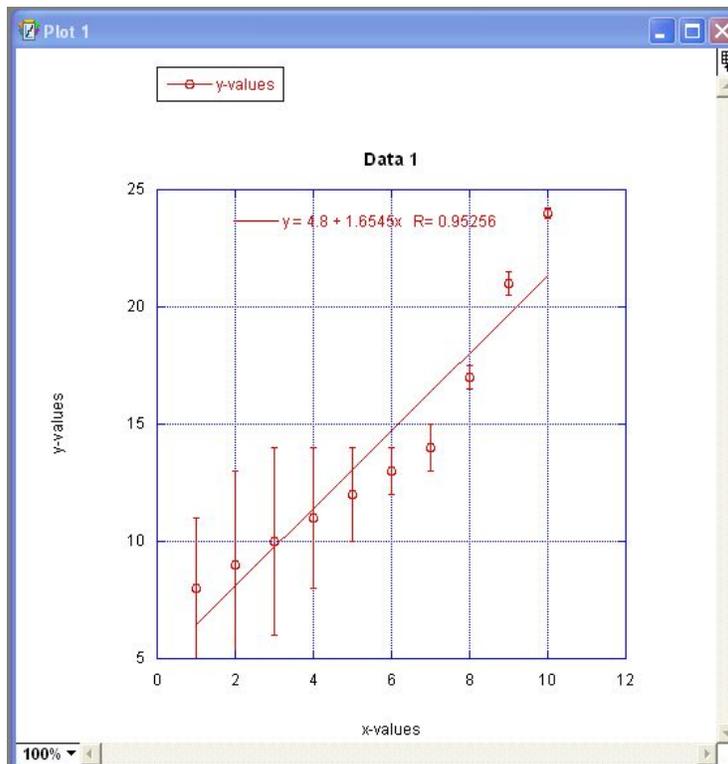
Finally, specify the column which should be used to obtain the weights.



The weighted plot is shown below. Note that the points with the largest error bars have been almost ignored by the straight line. (Of course, you can fit any function to your data; the straight line is used here as an example.)



In comparison, here's the unweighted plot.



Edited October 27, 2011