# Documentation for gnuPipe.h and gnuPipe.c

Steven Andrews, © 2003-2017

## Description

These routines allow plotting of data by piping to the free gnuplot program. They require that gnuplot is loaded and functional. To test whether it works enter "gnuplot" at a system prompt and see if it runs.

## Dependencies

Only the gnuplot software.

## History

| | |
|---|---|
| 10/29/01 | started |
| 1/2017 | updated |
| 11/30/22 | added variable arguments to gnuCommand and added gnuData3 |

## Functions

*File functions*

```
FILE *gnuWindow();
```
Opens a pipe to gnuplot, returning a pointer to the file. This pointer needs to be sent back to all of the other library routines. This also turns off the standard gnuplot legend. Returns `NULL` on failure (probably due to gnuplot not being installed).

```
void gnuClose(FILE *gnu);
```
Closes the pipe to gnuplot. Note that a pipe can only be opened once, so don't close it unless you're completely done with it.

*Direct gnuplot access*

```
void gnuCommand(FILE *gnu,char *s,...);
```
Sends a command string to gnuplot. Enter the string, or format string, in s and enter any variable values afterward. The formatted string can't have more than 1024 characters. Since it's not possible to get messages back from gnuplot, there is no way for a calling program to know if an error occured.

```
void gnuPrompt(FILE *gnu);
```
Displays a prompt to the standard error to allow the program user to enter commands directly to gnuplot. It terminates when the user enters a blank line.

*Set up functions*

```
void gnuSetScales(FILE *gnu,double xa,double xb,double ya,double yb);
```

Sets scales of the plot, where the *x* range is from `xa` to `xb` and the *y* range is from `ya` to `yb`. If `xa=xb` then autoscaling is used for `x`, and similarly for *y*.

*Data plotting*

```
void gnuData(FILE *gnu,double *x,double *y,int m,int col);
```
Plots sets of *x*,*y* data to gnuplot, where there is one column of `x` values, with `m` items and `col` columns of `y` data, with `m` values each.

```
void gnuData2(FILE *gnu,double *x,double *y,int *ct,int col);
```
Similar to `gnuData` except that there are `col` columns of `x` data to plot against the respective columns of `y` data. Each column has `ct[]` elements.

```
void gnuData3(FILE *gnu,double *x,double **y,int m,int col);
```
Same as gnuData, except that the y vector has separate rows and columns, rather than combined together.

**Examples of gnuplot functions**

```
plot 'file.txt'
plot 'file.txt' with lines
plot 'file.txt' with linespoints
plot 'file.txt' with points

plot sin(x) title 'Sine function', tan(x) title 'Tangent'
plot [t=-4,4] exp(-t**2), t**2 / 16

set arrow 1 from 0,0 to 10,1.1 nohead lc rgb 'red'
unset arrow 1
```