SDS 332/392

gnuplot

Original content courtesy of:

Dr. Karl W. Schulz, TACC
Dr. Bill Barth, TACC

SCIENTIFIC SIMULATION

- Running simulation codes is only part of the battle
- Pre- and post-processing may account for a large fraction of the total time to solution
- model creation and mesh generation
- data integration
- visualization of results
- derived quantities

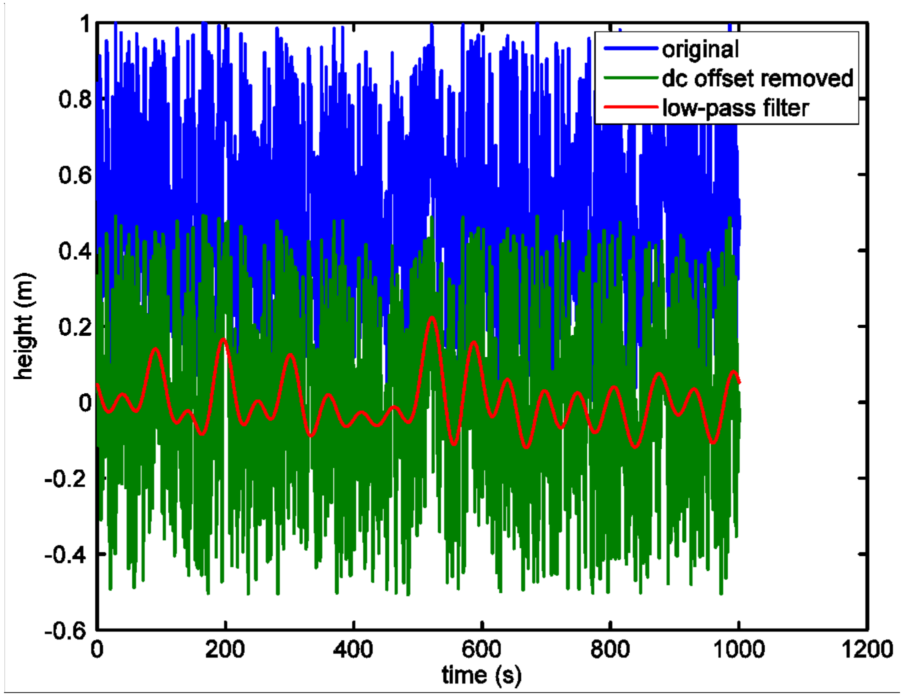
POST-PROCESSING

- Simulation results have to be interpreted
  - are they “right”?
    - are the errors acceptable?
    - does the model match the physics?
  - meaning needs to be extracted
- Plot pictures of the results
- Derive quantities of interest
  - average temperature
  - maximum temperature

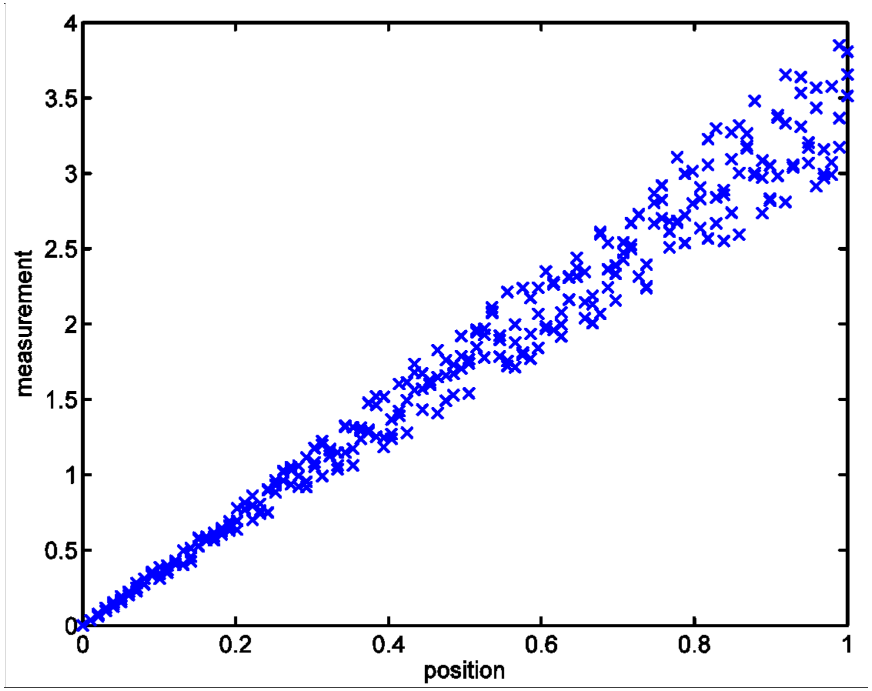
Plotting Results

- 1-D
- function graphing
- scatter plots
- 2-D
- contour/isoline plots
- surface plots
- pseudo color plots
- vector arrows
- 3-D
- isosurface plots
- slices with 2-D plots
- volume rendering

1-D

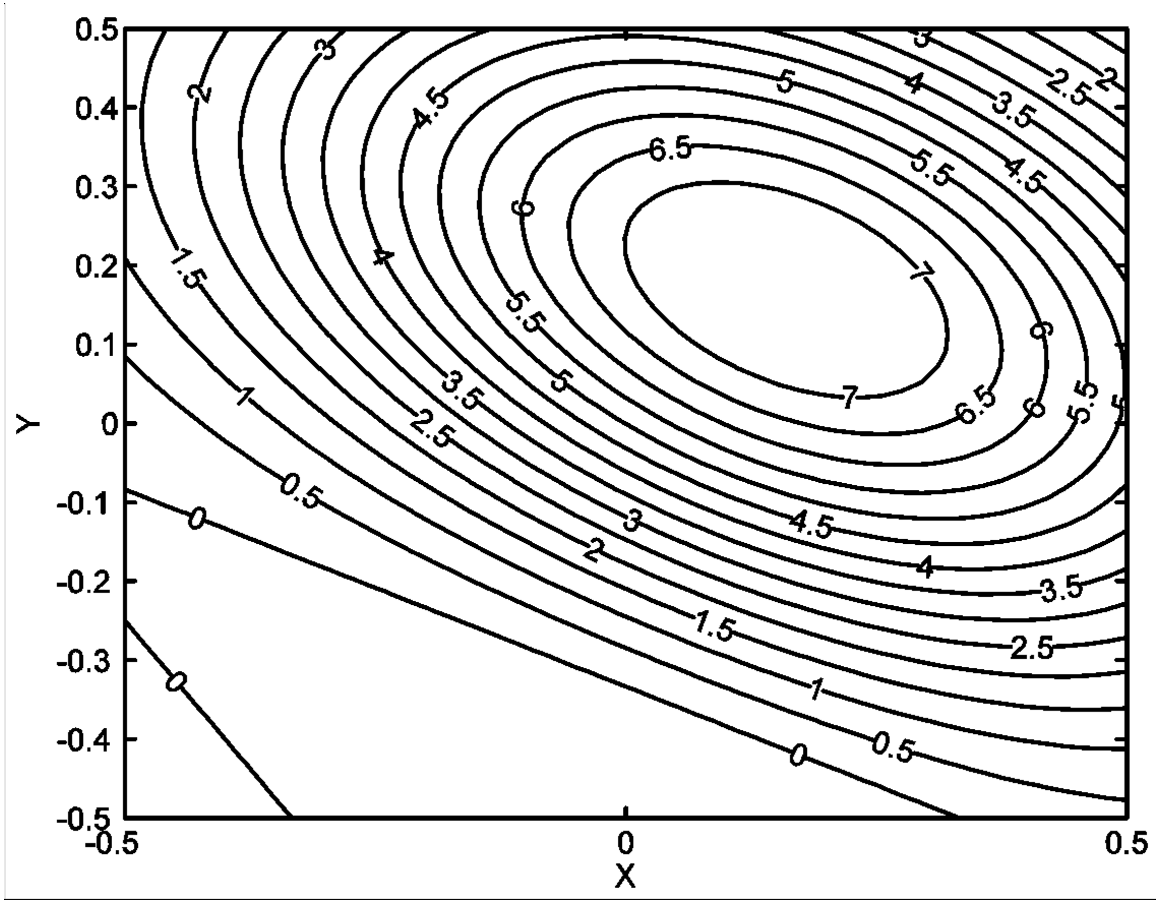


Line Plot



Scatter Plot

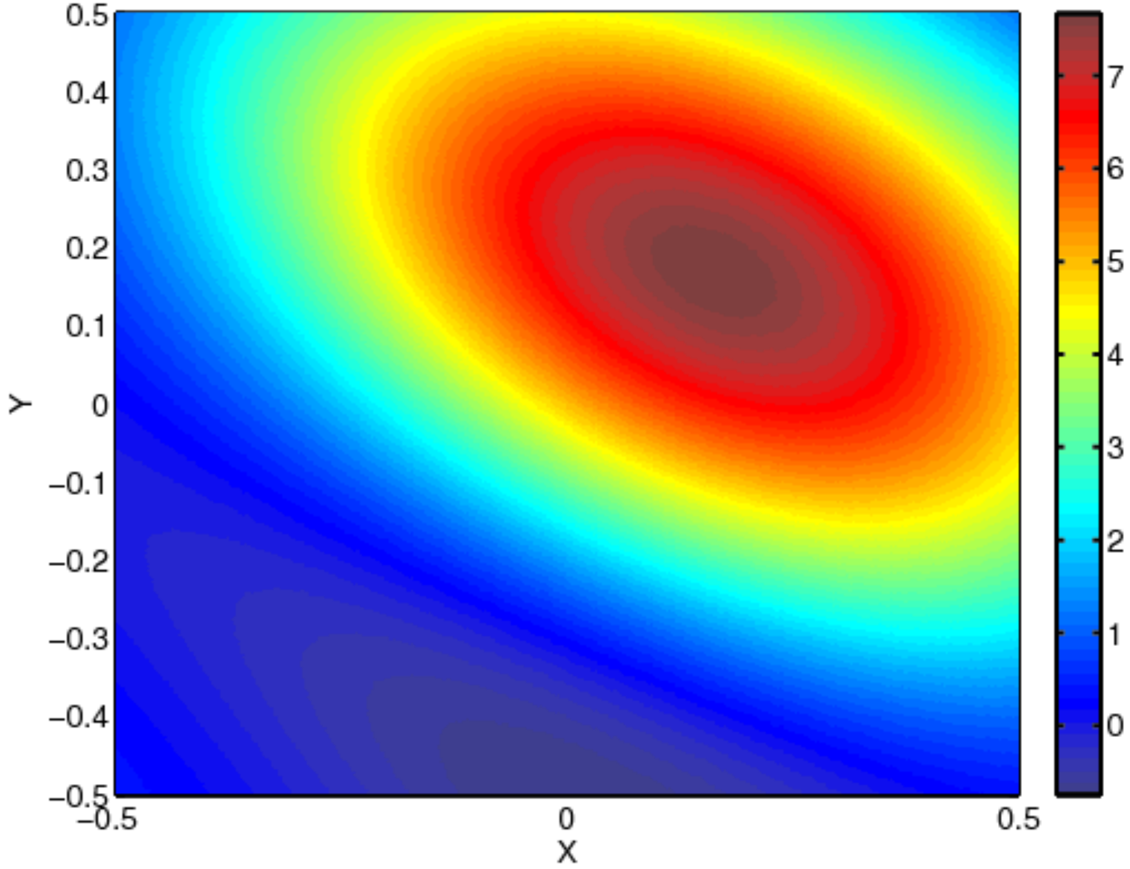
2-D—CONTOUR PLOTS



$c = f(x, y)$

- Draw a curve for each contour level

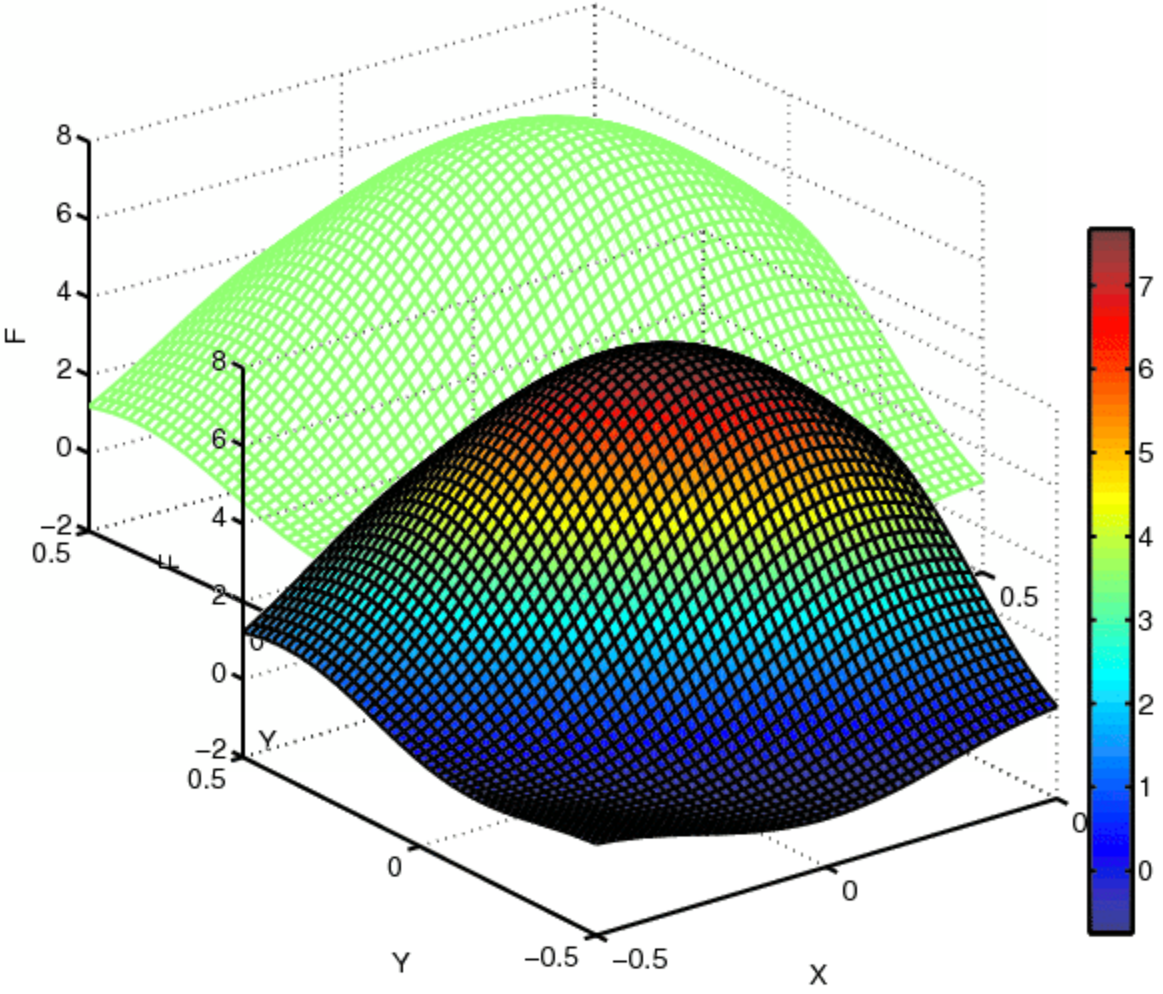
2-D—Pseudocolor Plots



c □ $f(x, y)$

- Map function values to colors
- Plot a blob of color at each data point

2-D—Surface/Function Plot

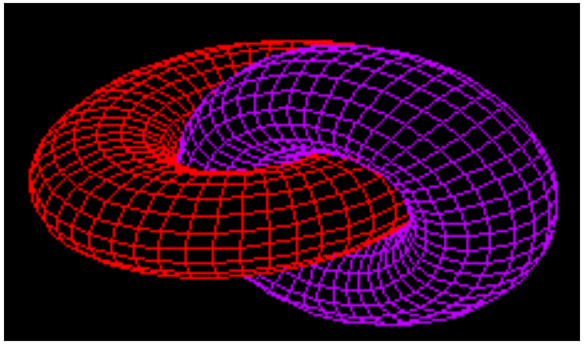


$z = f(x, y)$

- Function value is height
- May use pseudocolor on the plotted surface
- Needs 3-D projection

WHAT IS GNUPLOT?

- What is it?
- Gnuplot is a portable command-line driven, interactive data and function plotting utility for UNIX, Linux, Mac, and others
- Supports many types of plots in either two or three dimensions
- Can be used interactively or in a batch mode of operation
- Software is copyrighted but freely distributeable
- Includes a simple programming-style interface which is intuitive for those with a math/science background



- What it is not.
- gnuplot is not a fancy GUI for high-end visualization, but it can be extremely handy for post-processing data as it is generated on HPC resources

GNUPLOT BASICS

- gnuplot is generally used to visualize data from text files in columnar format

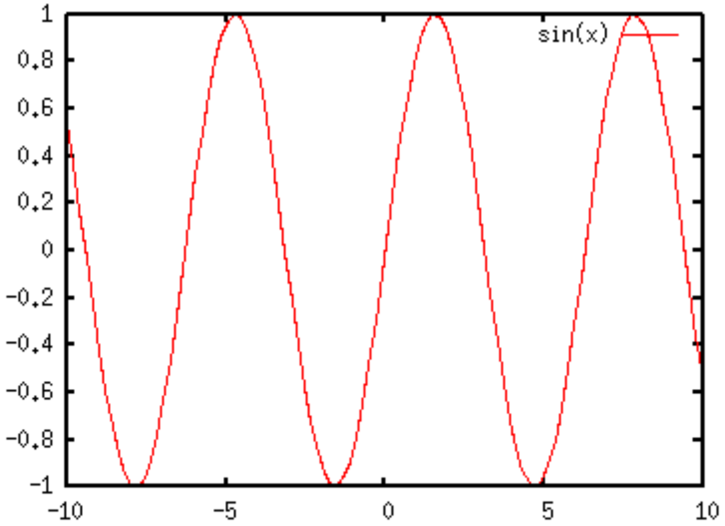
- Also includes support for plotting basic mathematical functions

- Numerous output formats are available (eg. png, gif, eps, pbm, etc)

- To use after installing, just type "**gnuplot**" and you will be placed into an interactive prompt:

```
   > gnuplot
G N U P L O T
Version 4.0 patchlevel 0
last modified Thu Apr 15 14:44:22 CEST 2004
System: Linux 2.6.12.6_TACC-lustre-perfctr-1.4

Copyright (C) 1986 - 1993, 1998, 2004
Thomas Williams, Colin Kelley and many others
   Terminal type set to 'x11'
   gnuplot> plot sin(x)
```



GNUPLOT BASICS

- Frequently Used Commands:
- plot: primary command for plotting functions and data in 2D functions
- splot: primary command for 3D plotting
- replot: repeats the last `plot` or `splot` command
- set: the set command can be used to set lots of options controlling all aspects of a particular plot; however, an updated screen is not until a `plot`, `splot`, or `replot` command is given. Example set options are:
- set grid (draws grid lines)
- set term (determines which kind of graphics device to plot to)
- set output (redirects the display to a specified file or device)
- set xlabel (identify text for x-axis)
- unset: options set via the `set` command can be returned to default values us
- help: interface into extensive on-line help;

GNUPLOT BASICS

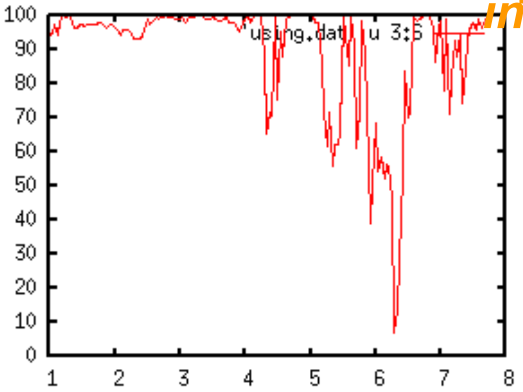
- For 2D line plotting, you identify which columns to plot against each other with the “using” option of plot. Consider the text file below with 6 columns of data

- To plot columns 3 and 6, issue the following:

```
gnuplot> plot ‘using.dat’ using 3:6 w lines
```

input filename

plot with lines



```
# Example Input File
#
891101 00   1.00  14   8.6  94.0
891101 01   1.04  10  17.5  94.4
891101 02   1.08   9   9.7  97.1
.
.
```

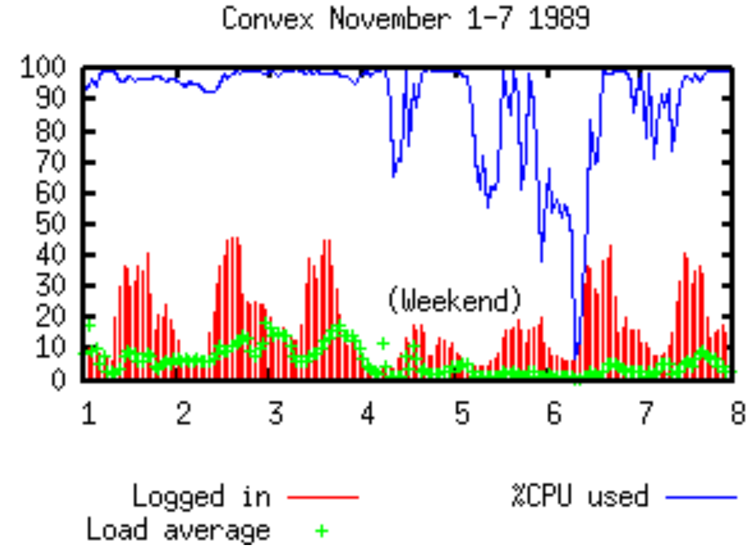
Comment: gnuplot ignores the line

GNUPLOT - LINE PLOTS

- It is easy to plot multiple series simultaneously; consider a more advanced example:

```
set xrange [1:8]
set title "Convex November 1-7 1989"
set key below
set label "(Weekend)" at 5,25 center
plot 'using.dat' using 3:4 title "Logged in" with impulses,\
     'using.dat' using 3:5 t "Load average" with points,\
     'using.dat' using 3:6 t "%CPU used" with lines
```

3 Series are plotted with different styles (impulses, points, and lines)

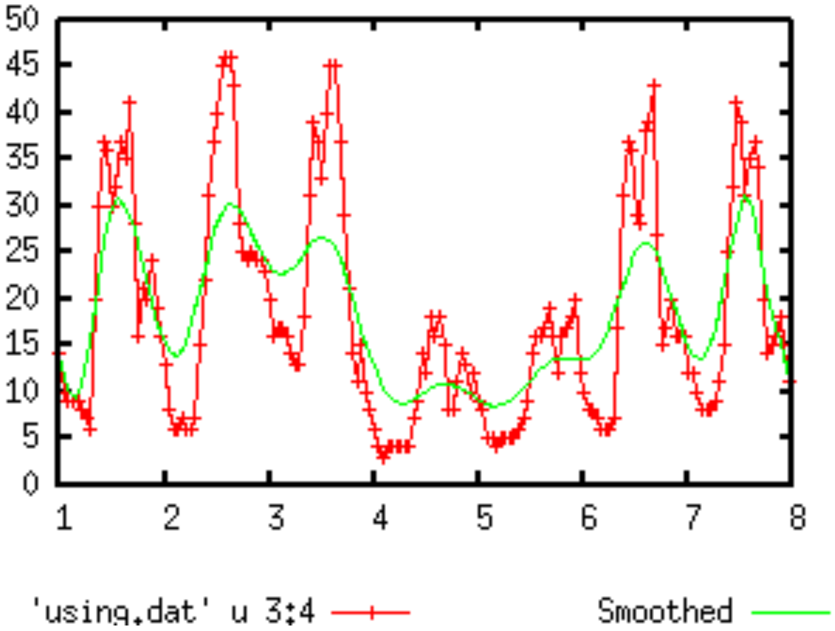


GNUPLOT - LINE PLOTS

- You can also use smoothing options on input data

- Available options are:
    - acsplines
    - bezier
    - csplines
    - frequency
    - sbezier
    - unique

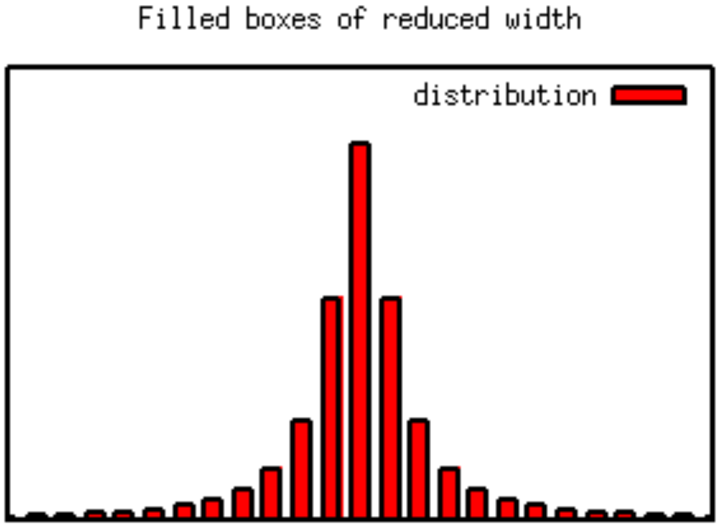
```
# Example line plot with smoothing
set key below
plot 'using.dat' u 3:4 w linesp,'using.dat' u 3:4 \
       smooth bezier t 'Smoothed' w l
```



GNUPLOT - HISTOGRAMS

```
set boxwidth 0.5 absolute
set style fill solid 1.000000 border -1
set samples 25, 25
set noxtics
set noytics
set title "Filled boxes of reduced width"
set yrange [ 0.00000 : 120.000 ]
plot [-10:10] 100/(1.0+x*x) title 'distribution' with boxes
```

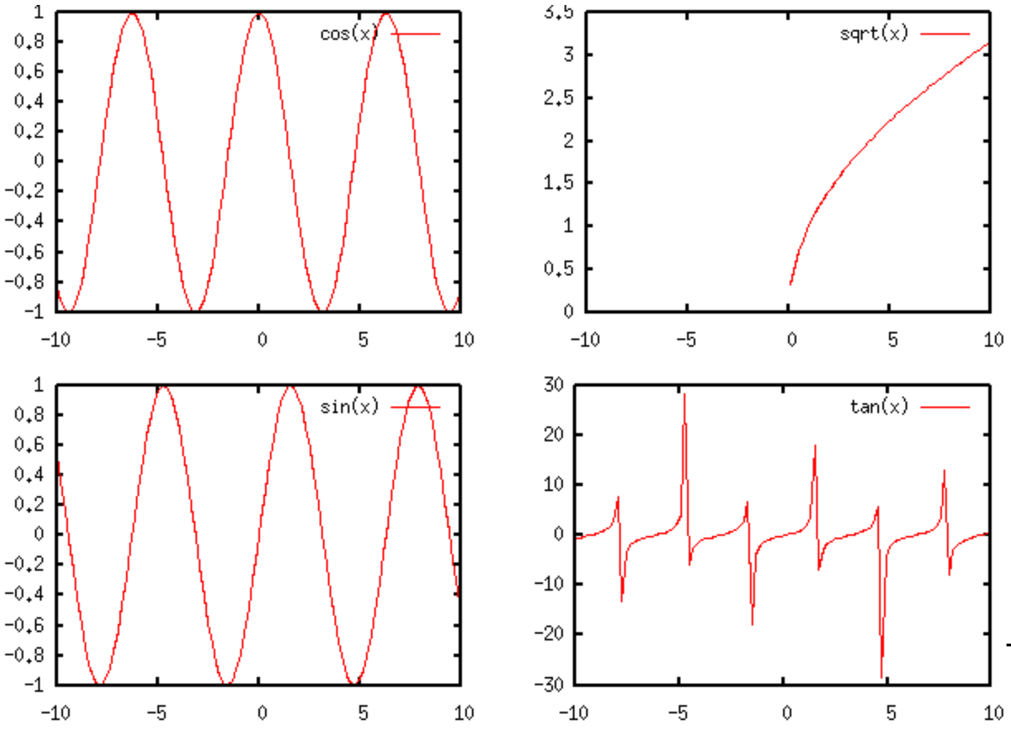
Example Histogram Plot



GNUPLOT - MULTIPLOT MODE

```
# Make 4 subplots
set multiplot
set size 0.5,0.5
set origin 0.,0.0  ; plot sin(x)
set origin 0.,0.5  ; plot cos(x)
set origin 0.5,0   ; plot tan(x)
set origin 0.5,0.5 ; plot sqrt(x)
```

Example Setup of SubPlots



GNUPLOT - PARAMETRIC PLOTS

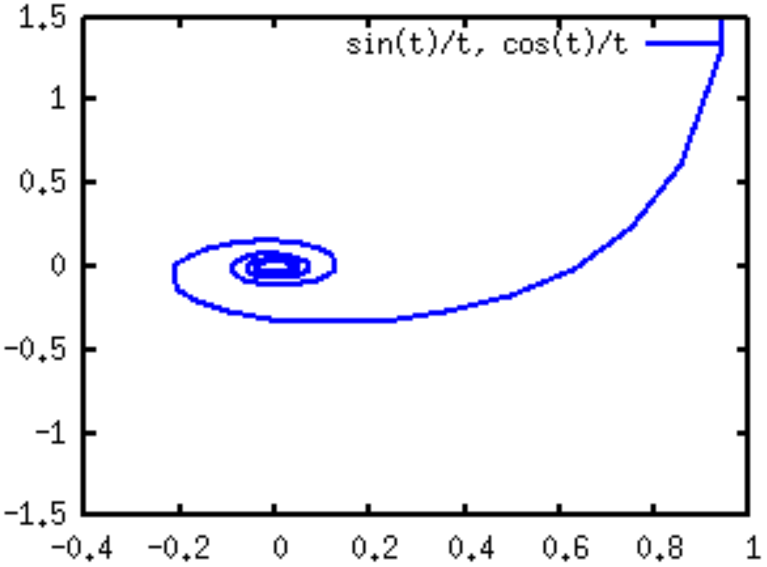
- Parametric functions can be plotted using an arbitrary dummy variable

- Use the "set dummy" command to define parametri variables

```
# Example parametric plot
set dummy t,y
set parametric
set trange [ 0.0001 : 31.4159 ]
set yrange [ -1.5 : 1.5 ]
plot sin(t)/t,cos(t)/t w l lt 3 lw 2
```

linetype of 3 = blue

linewidth of 2 draws a thicker line

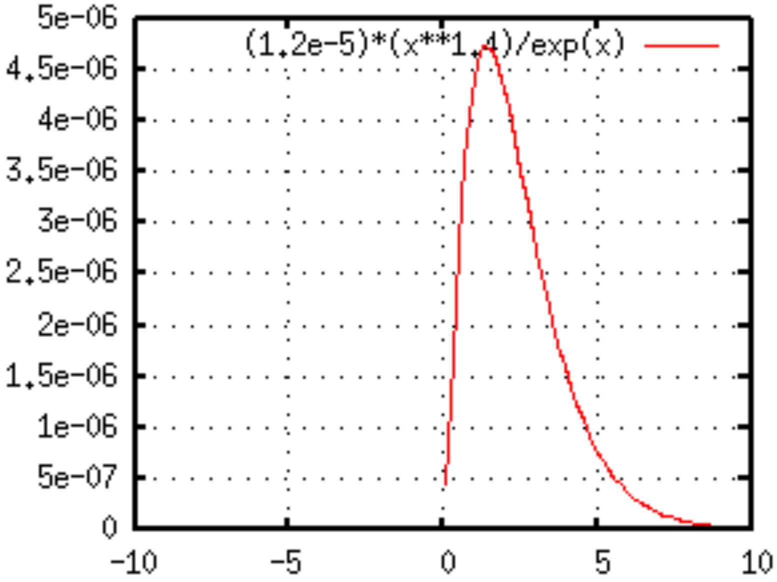


gnuplot - Functions

- **Math functions are specified in a Fortran style format**

```
# Plot a more complicated function
set grid
plot (1.2e-5)*(x**1.4)/exp(x)
```

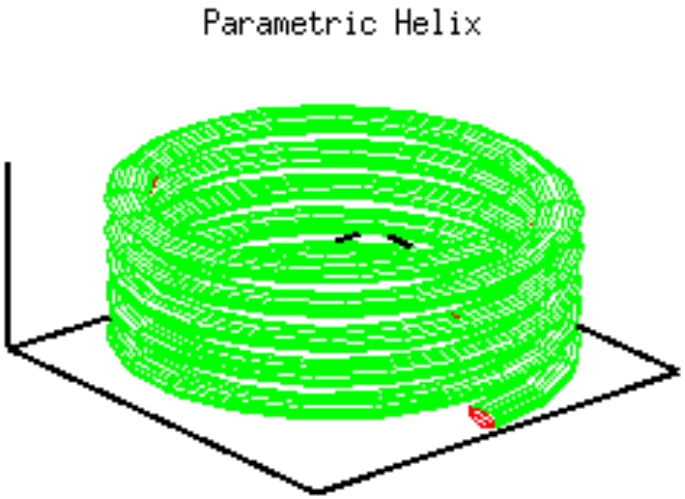
- **Many intrinsic functions are available (see 'help functions' to list)**
- **abs()**
- **sinh()**
- **log()**
- **rand()**
- **acos()**
- **sqrt()**



gnuplot - splot

```
set dummy u,v
set parametric
set view 45, 50, 1, 1
set isosamples 100, 20
set nokey; unset xtics; unset ytics; unset ztics
set hidden3d offset 1 trianglepattern 3 undefined 1 altdiagonal bentover
set ticslevel 0
set title "Parametric Helix"
set urange [ 0.00000 : 31.4159 ]
set vrange [ 0.00000 : 6.28319 ]
set zrange [ * : * ]
splot (1-0.1*cos(v))*cos(u),(1-0.1*cos(v))*sin(u),0.1*(sin(v)+u/1.7-10)
```

splot **is used to make 3D-style plots**



GNUPLOT - OUTPUT TIPS

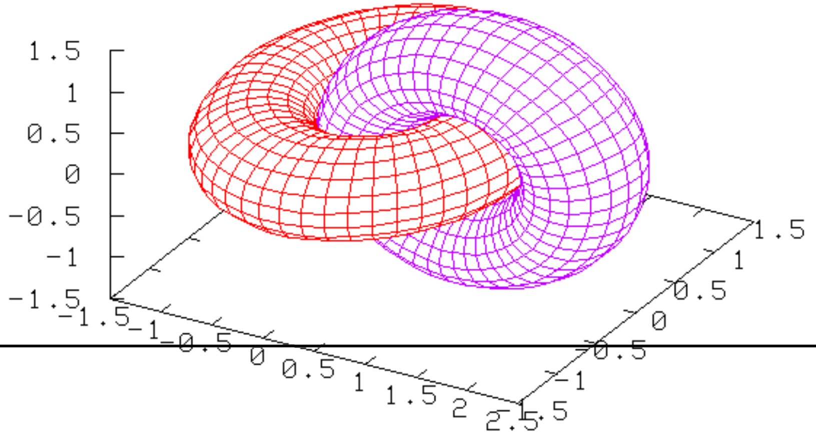
- To include your plots in papers or presentations, it is useful to save plots directly to a file

- Common formats are png or eps (eps is convenient for *latex* typesetting)

- Use the “terminal” and “output” options to control plotting format and filename

- The “set size” command is also useful for scaling down plots

```
set dummy u,v ; set nokey
set parametric
set view 50, 30, 1, 1
set isosamples 50, 20
set hidden3d
set ticslevel 0
set urange [ -3.14159 : 3.14159 ]
set vrange [ -3.14159 : 3.14159 ]
set zrange [ * : * ]
set term png transparent medium
set output 'test1.png'
splot cos(u)+.5*cos(u)*cos(v),sin(u)+.5*sin(u)*cos(v), \
     .5*sin(v) w l, 1+cos(u)+.5*cos(u)*cos(v), \
     .5*sin(v),sin(u)+.5*sin(u)*cos(v) w l
```



References/Acknowledgements

- Gnuplot Homepage: http://www.gnuplot.info/
- Gnuplot Short Course http://www.me.umn.edu/~langlais/UMN/gnuplot-shortcourse/gnuplot.pdf