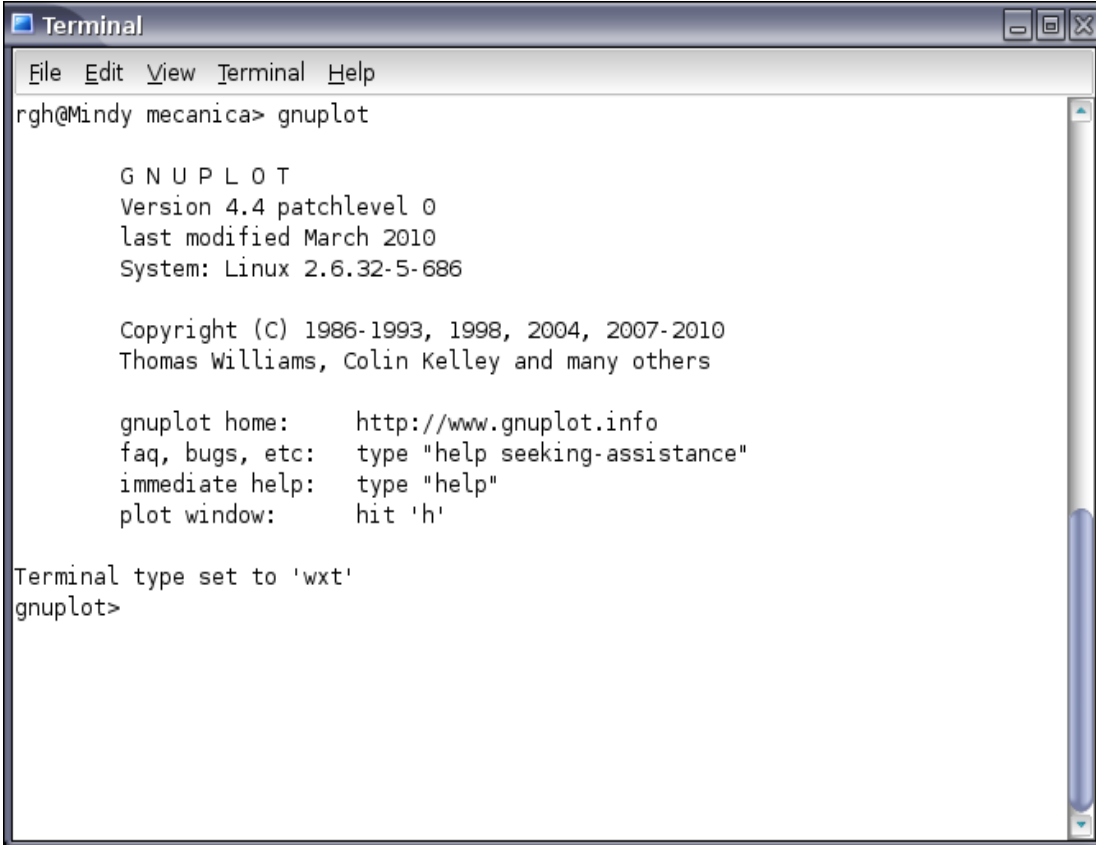


Breve Tutorial sobre el Graficador GNUPlot

Como iniciar el GNUPlot:

El programa para graficar datos científicos Gnuplot (standard en cualquier distribución Linux) se inicia desde cualquier ventana terminal escribiendo en el prompt “gnuplot”.

A screenshot of a terminal window titled "Terminal". The window has a menu bar with "File", "Edit", "View", "Terminal", and "Help". The terminal content shows the user typing "gnuplot" at the prompt "rgh@Mindy mecanica>". The output displays the GNUPlot version (4.4 patchlevel 0), the last modification date (March 2010), the system (Linux 2.6.32-5-686), and copyright information (1986-1993, 1998, 2004, 2007-2010 by Thomas Williams, Colin Kelley, etc.). It also lists help options: "gnuplot home: http://www.gnuplot.info", "faq, bugs, etc: type 'help seeking-assistance'", "immediate help: type 'help'", and "plot window: hit 'h'". At the bottom, it indicates "Terminal type set to 'wxt'" and shows the prompt "gnuplot>".

```
Terminal
File Edit View Terminal Help
rgh@Mindy mecanica> gnuplot

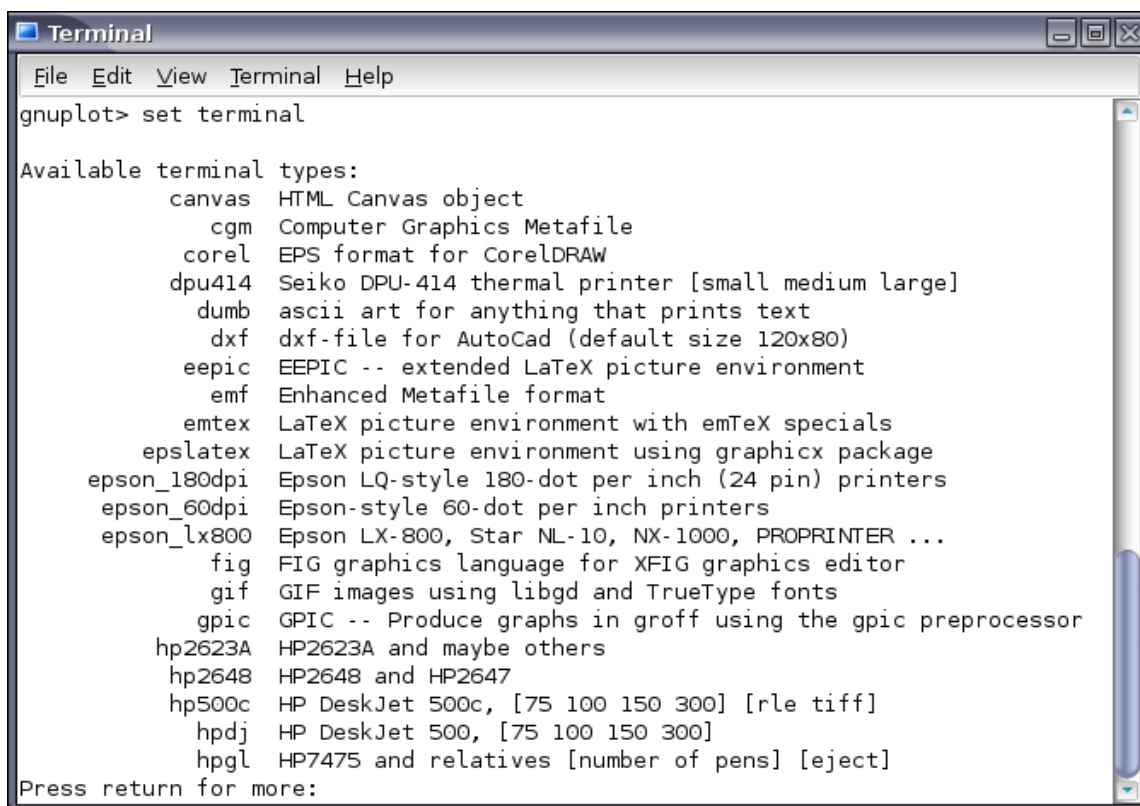
  G N U P L O T
  Version 4.4 patchlevel 0
  last modified March 2010
  System: Linux 2.6.32-5-686

  Copyright (C) 1986-1993, 1998, 2004, 2007-2010
  Thomas Williams, Colin Kelley and many others

  gnuplot home:      http://www.gnuplot.info
  faq, bugs, etc:   type "help seeking-assistance"
  immediate help:   type "help"
  plot window:      hit 'h'

Terminal type set to 'wxt'
gnuplot>
```

Al iniciarse el programa devuelve un prompt que indica que espera algún comando. Si se desea salir de Gnuplot se debe ingresar “exit”. Antes del prompt aparece una indicación de que tipo de terminal se esta usando, en nuestro caso una terminal “wxt”. Los diferentes tipos de terminales permiten generar salida a diferentes tipos de archivo. Para ver las terminales disponibles se debe escribir “set terminal”:



```
gnuplot> set terminal

Available terminal types:
  canvas HTML Canvas object
  cgm Computer Graphics Metafile
  corel EPS format for CorelDRAW
  dpu414 Seiko DPU-414 thermal printer [small medium large]
  dumb  ascii art for anything that prints text
  dxf   dxf-file for AutoCad (default size 120x80)
  eepic EEPIC -- extended LaTeX picture environment
  emf   Enhanced Metafile format
  emtex LaTeX picture environment with emTeX specials
  epslatex LaTeX picture environment using graphicx package
  epson_180dpi Epson LQ-style 180-dot per inch (24 pin) printers
  epson_60dpi  Epson-style 60-dot per inch printers
  epson_lx800 Epson LX-800, Star NL-10, NX-1000, PROPRINTER ...
  fig    FIG graphics language for XFIG graphics editor
  gif   GIF images using libgd and TrueType fonts
  gpic  GPIC -- Produce graphs in groff using the gpic preprocessor
  hp2623A HP2623A and maybe others
  hp2648 HP2648 and HP2647
  hp500c HP DeskJet 500c, [75 100 150 300] [rle tiff]
  hpdj  HP DeskJet 500, [75 100 150 300]
  hpgl  HP7475 and relatives [number of pens] [eject]

Press return for more:
```

El tipo de terminal inicial, wxt es una terminal que permite general gráficos en una ventana separada. La sintaxis es:

```
set term wxt {<n>}
  {size <width>,<height>}
  {{no}enhanced}
  {font <font>}
  {title "title"}
  {{no}persist}
  {{no}raise}
  {{no}ctrl}
  {close}
```

Este tipo de terminal permite múltiples ventanas con gráficos ya que el comando “set terminal wxt <n>” dirige la salida a la ventana número n. El título de la ventana usualmente es el número de ventana, pero también se puede cambiar utilizando el keyword “title”. Las ventanas de gráficos permanecerán abiertas hasta que se presione “q” con el cursor sobre la ventana, eligiendo “close” en el menú de la ventana, o ejecutando: “set terminal wxt <n> close”. Además, la terminal del tipo wxt permite interactuar con la ventana del gráfico utilizando el mouse y soporta un modo de texto extendido (“enhanced”) que permite introducir caracteres especiales.

Un comando para tener siempre en cuenta es “help” seguido del comando para el cual se quiere obtener información. Utilizando este comando es posible obtener una ayuda detallada incluso de keywords que modifican ciertos comandos. Por ejemplo, “help plot with” permite obtener los estilos permitidos en un gráfico y sus keywords modificadores.

Cómo graficar con Gnuplot utilizando los comandos plot y splot:

Los comandos “plot” y “splot” son los comandos básicos de Gnuplot. Permiten graficar funciones o datos de diversas maneras. El comando plot se utiliza para graficar funciones o datos bidimensionales y splot para tridimensionales.

La sintaxis a utilizar es:

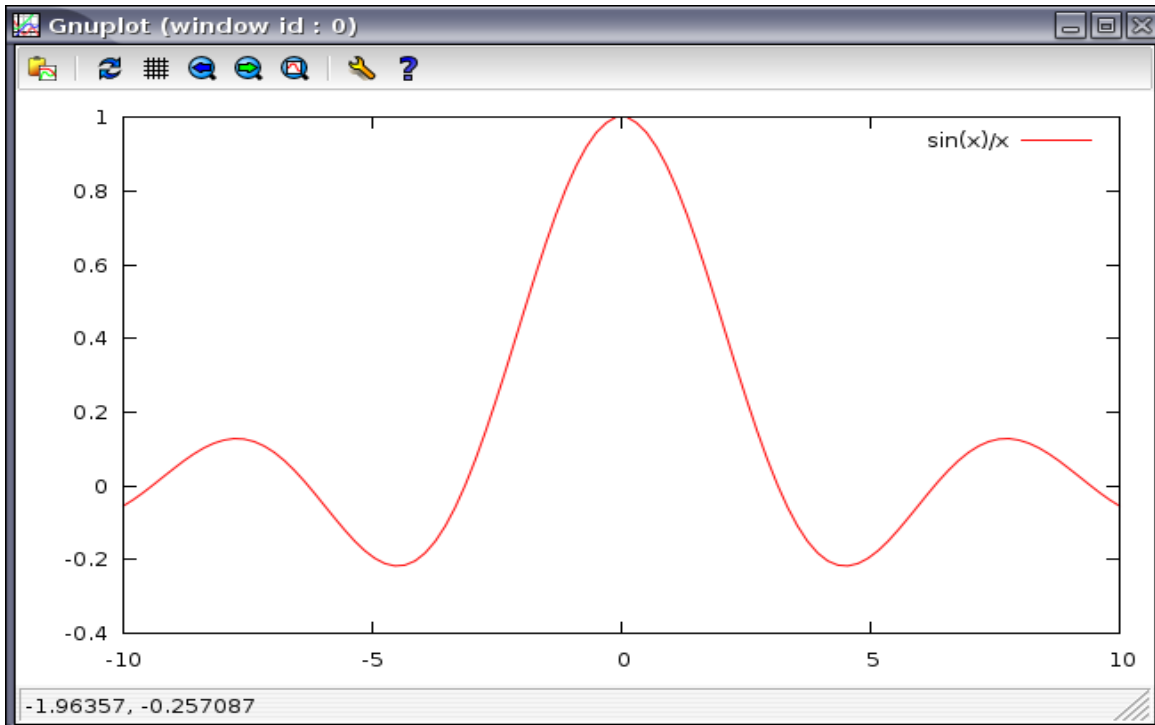
```
plot {<ranges>}
{<iteration>}
{<function> | {"<datafile>" {datafile-modifiers}}}}
{axes <axes>} {<title-spec>} {with <style>}
{, {definitions{,}} <function> ...}
```

donde se puede especificar para graficar una <function> o el nombre entre comillas de un archivo con datos. En la siguiente tabla se listan algunas de las funciones disponibles:

Función	Devuelve
abs(x)	absolute value of x, x
acos(x)	arc-cosine of x
asin(x)	arc-sine of x
atan(x)	arc-tangent of x
cos(x)	cosine of x, x is in radians.
cosh(x)	hyperbolic cosine of x, x is in radians
erf(x)	error function of x
exp(x)	exponential function of x, base e
inverf(x)	inverse error function of x
invnorm(x)	inverse normal distribution of x
log(x)	log of x, base e
log10(x)	log of x, base 10
norm(x)	normal Gaussian distribution function
rand(x)	pseudo-random number generator
sgn(x)	1 if x > 0, -1 if x < 0, 0 if x=0
sin(x)	sine of x, x is in radians
sinh(x)	hyperbolic sine of x, x is in radians
sqrt(x)	the square root of x
tan(x)	tangent of x, x is in radians
tanh(x)	hyperbolic tangent of x, x is in radians

Por ejemplo, para graficar la función $\sin(x)/x$:

```
Terminal
File Edit View Terminal Help
alumnos          mec-unsj-teo-1.pdf   tesp2-tema-2-2010.pdf
cartel-unsj.doc  mec-unsj-teo-2.pdf   tesp2-tema-3-2010.pdf
curso-doc-1.pdf  mec-unsj-teo-3.pdf   tesp2-tema-4-2010.pdf
curso-doc-2.pdf  mec-unsj-teo-4.pdf   tesp2-tema-5-2010.pdf
figuras          mec-unsj-teo-5.pdf   tesp2-tema-6-2010.pdf
gnuplot-tut.doc  mec-unsj-teo-6.pdf   tesp2-tema-7-2010.pdf
mec-unsj-prac-1.pdf mec-unsj-teo-7.pdf   textos
mec-unsj-prac-2.pdf papers              wi1.png
mec-unsj-prac-3.pdf programas           wi2.png
mec-unsj-prog.pdf tesp2-tema-1-2010.pdf
!
gnuplot> !ls
alumnos          mec-unsj-teo-1.pdf   tesp2-tema-2-2010.pdf
cartel-unsj.doc  mec-unsj-teo-2.pdf   tesp2-tema-3-2010.pdf
curso-doc-1.pdf  mec-unsj-teo-3.pdf   tesp2-tema-4-2010.pdf
curso-doc-2.pdf  mec-unsj-teo-4.pdf   tesp2-tema-5-2010.pdf
figuras          mec-unsj-teo-5.pdf   tesp2-tema-6-2010.pdf
gnuplot-tut.doc  mec-unsj-teo-6.pdf   tesp2-tema-7-2010.pdf
mec-unsj-prac-1.pdf mec-unsj-teo-7.pdf   textos
mec-unsj-prac-2.pdf papers              wi1.png
mec-unsj-prac-3.pdf programas           wi2.png
mec-unsj-prog.pdf tesp2-tema-1-2010.pdf
!
gnuplot> plot sin(x)/x
gnuplot>
```



Existen múltiples formas de modificar el gráfico creado incluyendo un título, nombre a los ejes, etc. Algunos de los comandos posibles son:

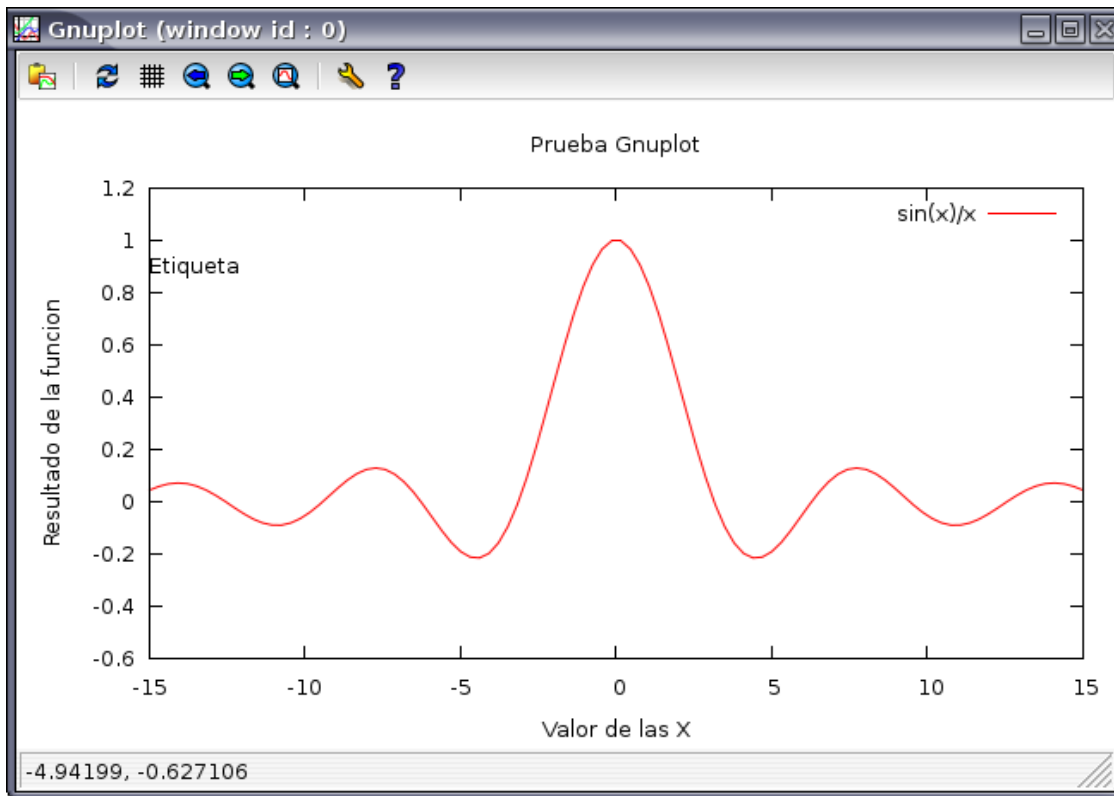
Create a title:	> set title "Prueba Gnuplot"
Put a label on the x-axis:	> set xlabel "Valor de las X"
Put a label on the y-axis:	> set ylabel "Resultado de la funcion"
Change the x-axis range:	> set xrange [-15:15]
Change the y-axis range:	> set yrange [-0.6:1.2]
Have Gnuplot determine ranges:	> set autoscale
Move the key:	> set key at 12,-0.5
Delete the key:	> unset key
Put a label on the plot:	> set label "Eitqueta" at -15,0.9
Remove all labels:	> unset label
Plot using log-axes:	> set logscale
Plot using log-axes on y-axis:	> unset logscale; set logscale y
Change the tic-marks:	> set xtics (-12,-6,0,6,12)
Return to the default tics:	> unset xtics; set xtics auto

Después de ingresar estos comandos se debe ejecutar el comando “replot” o graficar nuevamente con el comando “plot”:

```

Terminal
File Edit View Terminal Help
mec-unsj-prog.pdf      tesp2-tema-2-2010.pdf
mec-unsj-teo-1.pdf    tesp2-tema-3-2010.pdf
!
gnuplot> !ls
alumnos               mec-unsj-teo-2.pdf    tesp2-tema-4-2010.pdf
cartel-unsj.doc       mec-unsj-teo-3.pdf    tesp2-tema-5-2010.pdf
curso-doc-1.pdf       mec-unsj-teo-4.pdf    tesp2-tema-6-2010.pdf
curso-doc-2.pdf       mec-unsj-teo-5.pdf    tesp2-tema-7-2010.pdf
figuras               mec-unsj-teo-6.pdf    textos
gnuplot-tut.doc       mec-unsj-teo-7.pdf    wi1.png
mec-unsj-prac-1.pdf   papers                 wi2.png
mec-unsj-prac-2.pdf   programas              wi3a.png
mec-unsj-prac-3.pdf   tesp2-tema-1-2010.pdf wi3b.png
mec-unsj-prog.pdf     tesp2-tema-2-2010.pdf
mec-unsj-teo-1.pdf    tesp2-tema-3-2010.pdf
!
gnuplot> plot sin(x)/x
gnuplot> set title "Prueba Gnuplot"
gnuplot> set xlabel "Valor de las X"
gnuplot> set ylabel "Resultado de la funcion"
gnuplot> set xrange [-15:15]
gnuplot> set yrange [-0.6:1.2]
gnuplot> set label "Etiqueta" at -15,0.9
gnuplot> replot
gnuplot> █

```

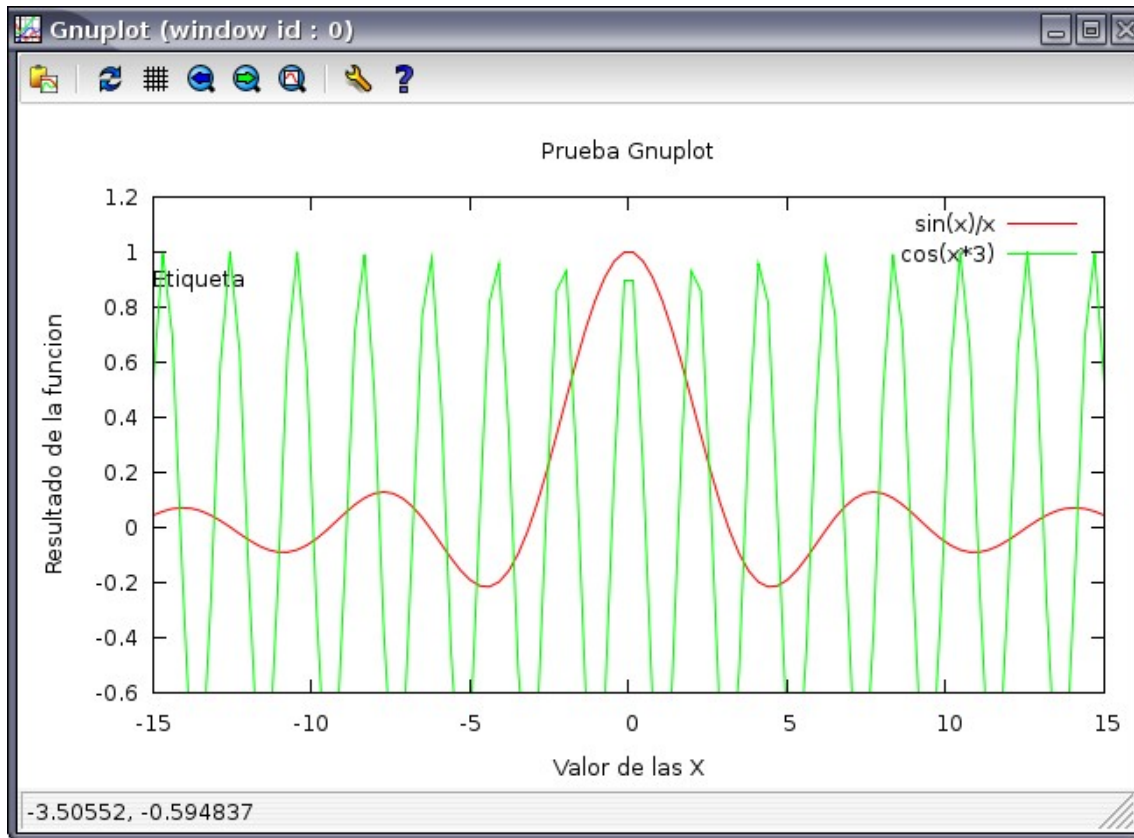


Si sobre este gráfico se quiere graficar otra función también se debe utilizar el comando “replot”:

```

Terminal
File Edit View Terminal Help
mec-unsj-teo-1.pdf  tesp2-tema-3-2010.pdf
!
gnuplot> !ls
alumnos          mec-unsj-teo-2.pdf  tesp2-tema-4-2010.pdf
cartel-unsj.doc  mec-unsj-teo-3.pdf  tesp2-tema-5-2010.pdf
curso-doc-1.pdf  mec-unsj-teo-4.pdf  tesp2-tema-6-2010.pdf
curso-doc-2.pdf  mec-unsj-teo-5.pdf  tesp2-tema-7-2010.pdf
figuras          mec-unsj-teo-6.pdf  textos
gnuplot-tut.doc  mec-unsj-teo-7.pdf  wi1.png
mec-unsj-prac-1.pdf  papers              wi2.png
mec-unsj-prac-2.pdf  programas           wi3a.png
mec-unsj-prac-3.pdf  tesp2-tema-1-2010.pdf  wi3b.png
mec-unsj-prog.pdf  tesp2-tema-2-2010.pdf
mec-unsj-teo-1.pdf  tesp2-tema-3-2010.pdf
!
gnuplot> plot sin(x)/x
gnuplot> set title "Prueba Gnuplot"
gnuplot> set xlabel "Valor de las X"
gnuplot> set ylabel "Resultado de la funcion"
gnuplot> set xrange [-15:15]
gnuplot> set yrange [-0.6:1.2]
gnuplot> set label "Etiqueta" at -15,0.9
gnuplot> replot
gnuplot> replot cos(x*3)
gnuplot>

```



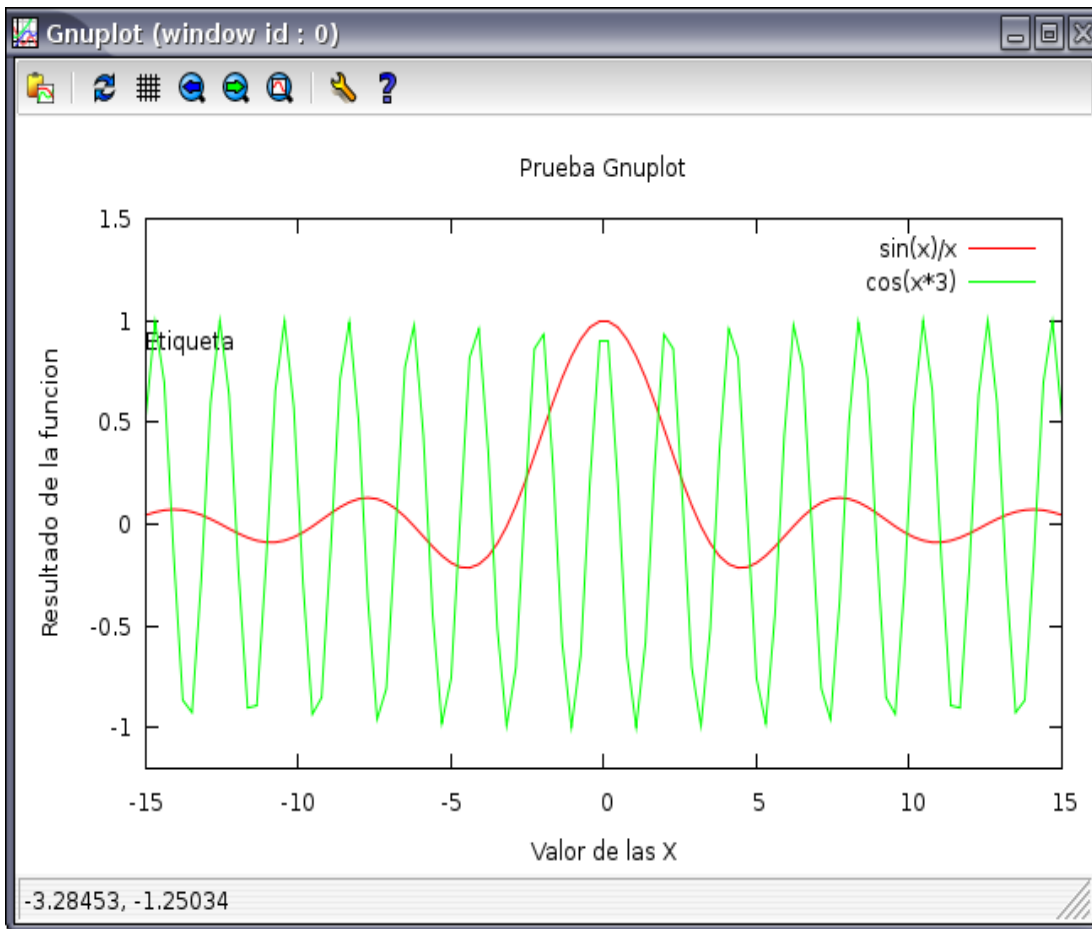
Otra opción es utilizar nuevamente el comando “plot” para graficar ambas funciones simultáneamente: “plot sin(x)/x , cos(x*3)”.

El gráfico anterior muestra que el rango en el eje Y no es el adecuado porque la función $\cos(x*3)$ toma valores menores a -0.6 y, además, la leyenda donde se indica con que color se grafica cada función quedo algo tapada al graficar esta función. Para resolver esto lo mejor es modificar el rango con el comando “set yrange [xx:xx]”, pero es posible también definir el rango en la línea del comando plot:

```

Terminal
File Edit View Terminal Help
!
gnuplot> !ls
alumnos          mec-unsj-teo-2.pdf      tesp2-tema-4-2010.pdf
cartel-unsj.doc  mec-unsj-teo-3.pdf      tesp2-tema-5-2010.pdf
curso-doc-1.pdf  mec-unsj-teo-4.pdf      tesp2-tema-6-2010.pdf
curso-doc-2.pdf  mec-unsj-teo-5.pdf      tesp2-tema-7-2010.pdf
figuras          mec-unsj-teo-6.pdf      textos
gnuplot-tut.doc  mec-unsj-teo-7.pdf      wi1.png
mec-unsj-prac-1.pdf papers                  wi2.png
mec-unsj-prac-2.pdf programas                wi3a.png
mec-unsj-prac-3.pdf tesp2-tema-1-2010.pdf   wi3b.png
mec-unsj-prog.pdf  tesp2-tema-2-2010.pdf
mec-unsj-teo-1.pdf  tesp2-tema-3-2010.pdf
!
gnuplot> plot sin(x)/x
gnuplot> set title "Prueba Gnuplot"
gnuplot> set xlabel "Valor de las X"
gnuplot> set ylabel "Resultado de la funcion"
gnuplot> set xrange [-15:15]
gnuplot> set yrange [-0.6:1.2]
gnuplot> set label "Etiqueta" at -15,0.9
gnuplot> replot
gnuplot> replot cos(x*3)
gnuplot> plot [-15:15][-1.2:1.5] sin(x)/x, cos(x*3)
gnuplot>

```



En esta oportunidad se incluyo el rango para ambos ejes en el comando “plot”. Si se desea ajustar solo uno de los ejes utilizando este método el rango del otro debe indicarse como “[]”.

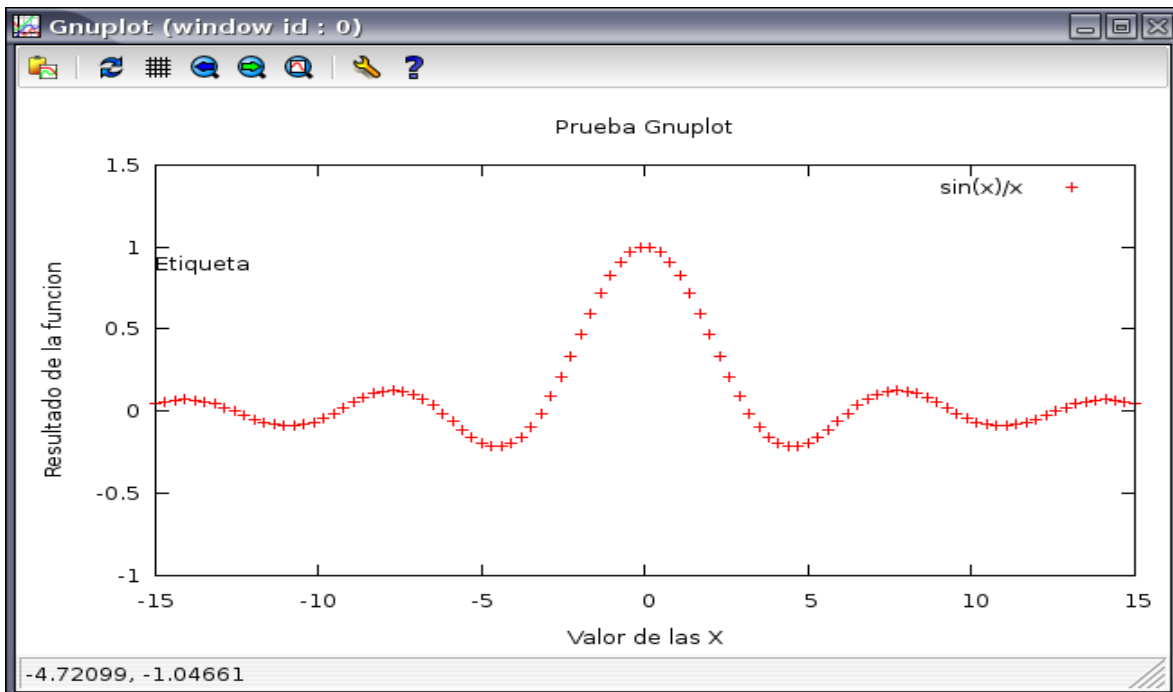
Hasta ahora los graficos se dibujan con líneas pero también son posibles otras opciones. Cada función a graficar puede incluir un keyword “with” que indica cómo se debe graficar la función y cuya sintaxis es:

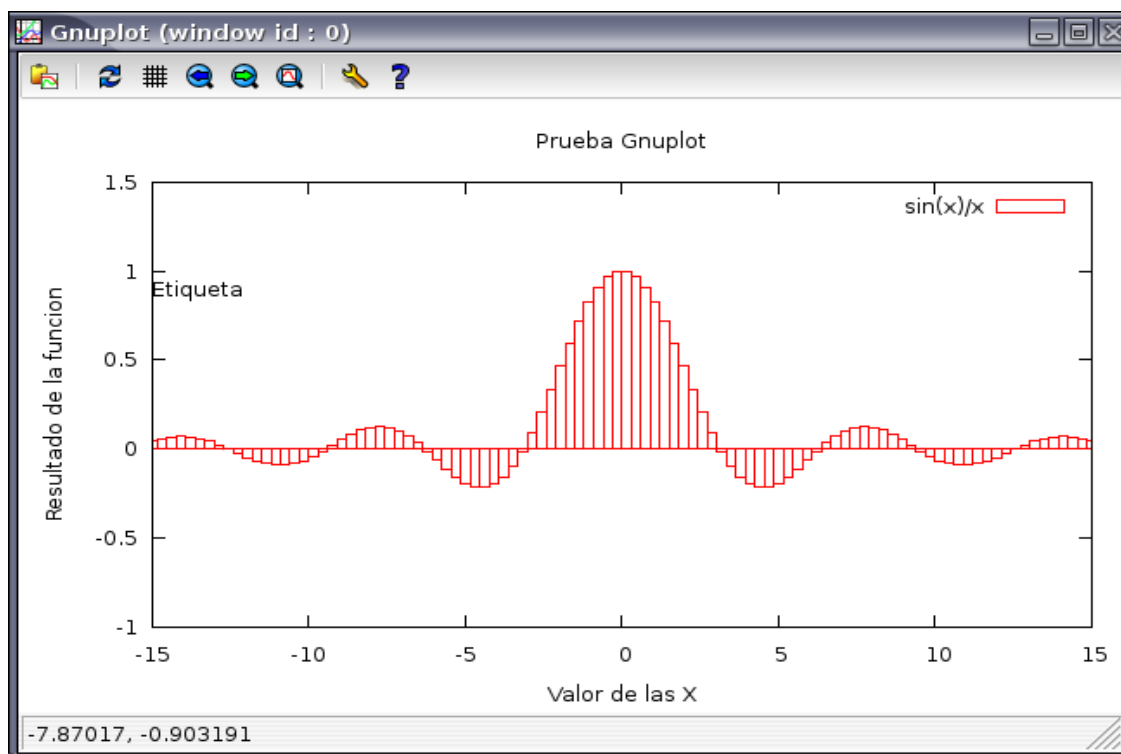
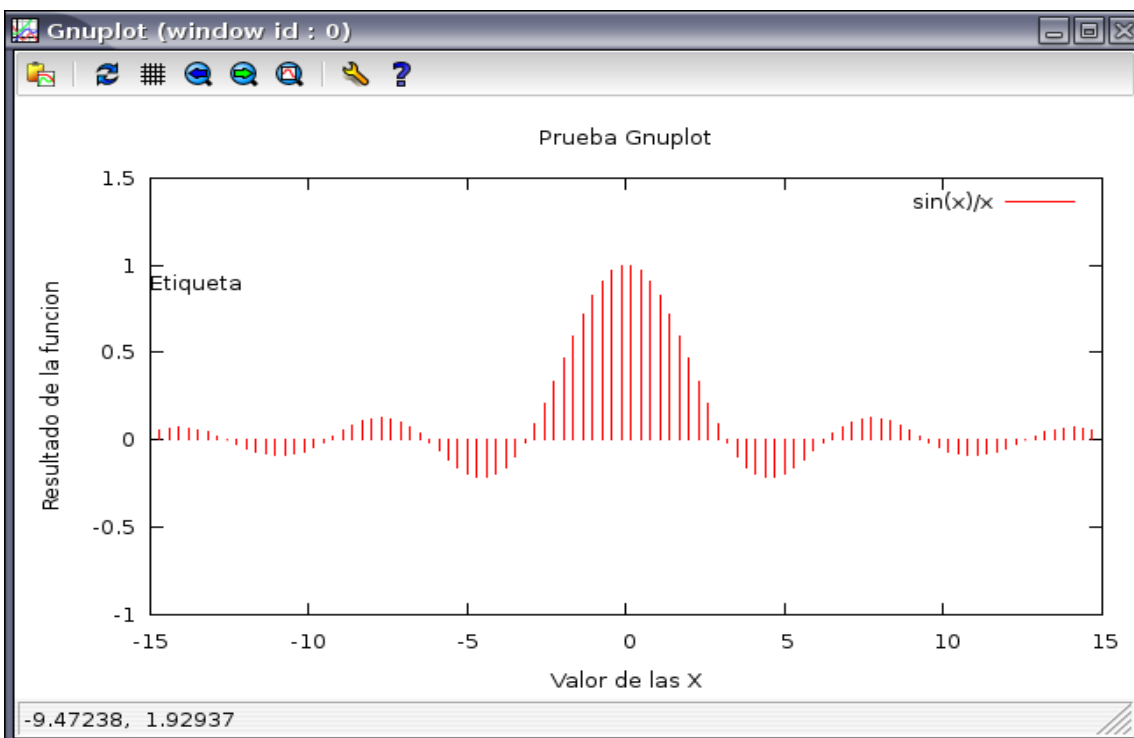
```
with <style> { {linestyle | ls <line_style>}
  | {{linetype | lt <line_type>}
  {linewidth | lw <line_width>}
  {linecolor | lc <colorespec>}
  {pointtype | pt <point_type>}
  {pointsize | ps <point_size>}
  {fill | fs <fillstyle>}
  {nohidden3d} {nocontours} {nosurface}
  {palette}}
}
```

donde <style> puede ser uno de los posibles estilos que se dividen en dos grupos. El primero incluye los estilos lines, points, linespoints, dots, impulses, labels, steps, fsteps, histeps, errorbars, errorlines, financebars, xerrorbar, xerrorlines, xyerrorbars, xyerrorlines, yerrorbars, yerrorlines, y vectors que corresponden a estilos que tienen asociadas líneas, puntos o textos. El segundo grupo incluye a los

estilos boxes, boxerrorbars, boxxyerrorbars, candlesticks, filledcurves, histograms, image, rgbimage, rgbalpha, circles, y pm3d que incluyen la posibilidad de rellenar áreas del gráfico. Por ejemplo, la función “sin(x)/x” graficada con diferentes estilos sería:

```
Terminal
File Edit View Terminal Help
curso-doc-1.pdf      mec-unsj-teo-3.pdf  tesp2-tema-4-2010.pdf  wi4b.png
curso-doc-2.pdf      mec-unsj-teo-4.pdf  tesp2-tema-5-2010.pdf  wi5a.png
figuras              mec-unsj-teo-5.pdf  tesp2-tema-6-2010.pdf  wi5b.png
gnuplot-tut.doc      mec-unsj-teo-6.pdf  tesp2-tema-7-2010.pdf  wi6a.png
mec-unsj-prac-1.pdf  mec-unsj-teo-7.pdf  textos                  wi6b.png
mec-unsj-prac-2.pdf  papers              wi1.png
mec-unsj-prac-3.pdf  programas           wi2.png
mec-unsj-prog.pdf    tesp2-tema-1-2010.pdf  wi3a.png
!
gnuplot> !ls
alumnos              mec-unsj-teo-1.pdf  tesp2-tema-2-2010.pdf  wi3b.png
cartel-unsj.doc      mec-unsj-teo-2.pdf  tesp2-tema-3-2010.pdf  wi4a.png
curso-doc-1.pdf      mec-unsj-teo-3.pdf  tesp2-tema-4-2010.pdf  wi4b.png
curso-doc-2.pdf      mec-unsj-teo-4.pdf  tesp2-tema-5-2010.pdf  wi5a.png
figuras              mec-unsj-teo-5.pdf  tesp2-tema-6-2010.pdf  wi5b.png
gnuplot-tut.doc      mec-unsj-teo-6.pdf  tesp2-tema-7-2010.pdf  wi6a.png
mec-unsj-prac-1.pdf  mec-unsj-teo-7.pdf  textos                  wi6b.png
mec-unsj-prac-2.pdf  papers              wi1.png
mec-unsj-prac-3.pdf  programas           wi2.png
mec-unsj-prog.pdf    tesp2-tema-1-2010.pdf  wi3a.png
!
gnuplot> plot [-15:15][-1:1.5] sin(x)/x with points
gnuplot> plot [-15:15][-1:1.5] sin(x)/x with impulses
gnuplot> plot [-15:15][-1:1.5] sin(x)/x with boxes
gnuplot>
```



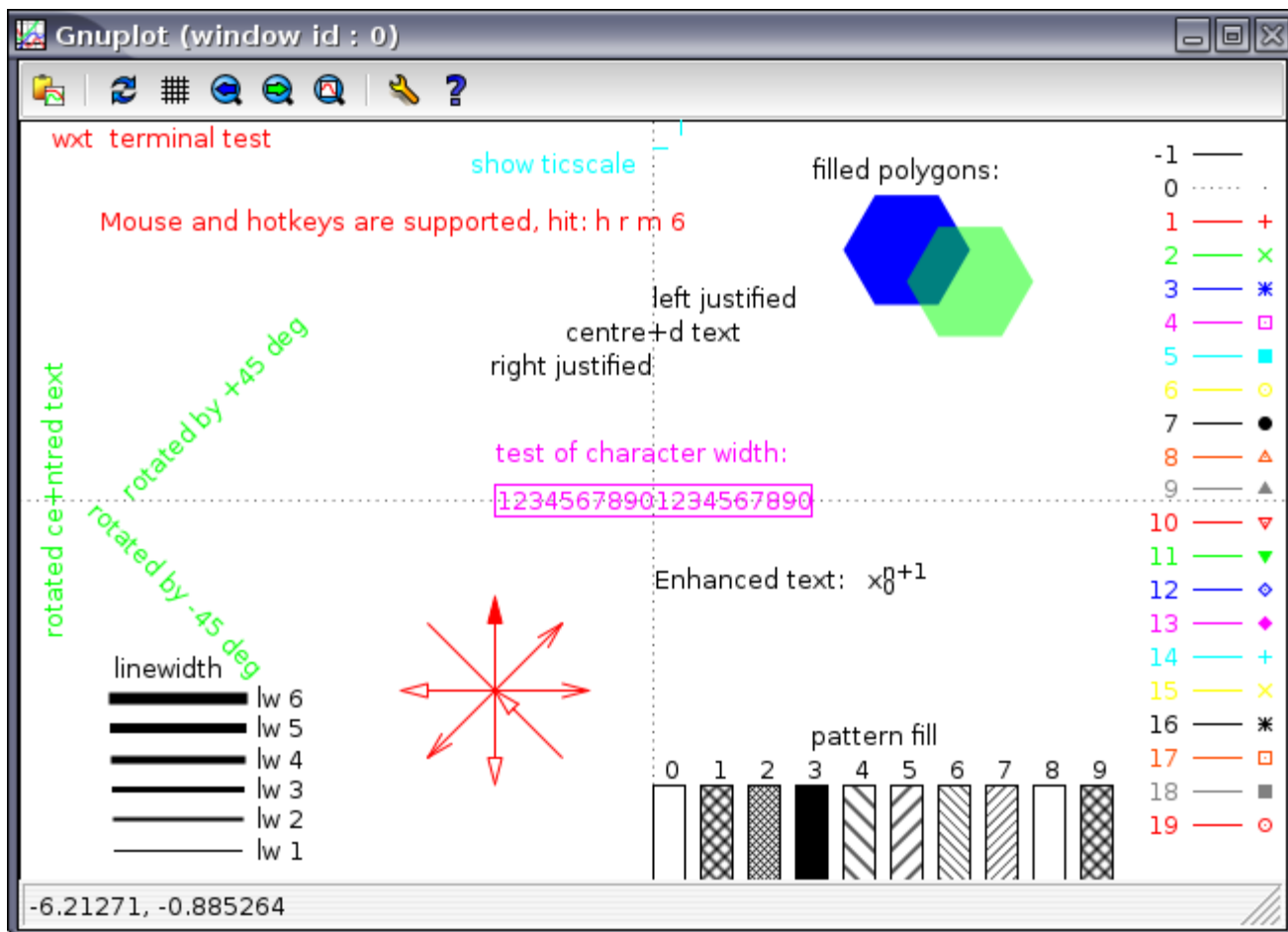


El keyword “with” también puede incluir información sobre el tipo de línea, color, tamaño, patrón de relleno (ver su sintaxis más arriba). Un resumen de las posibilidades se obtiene ingresando el comando “test” que muestra los tipos, tamaños, etc. disponibles en una ventana.

```

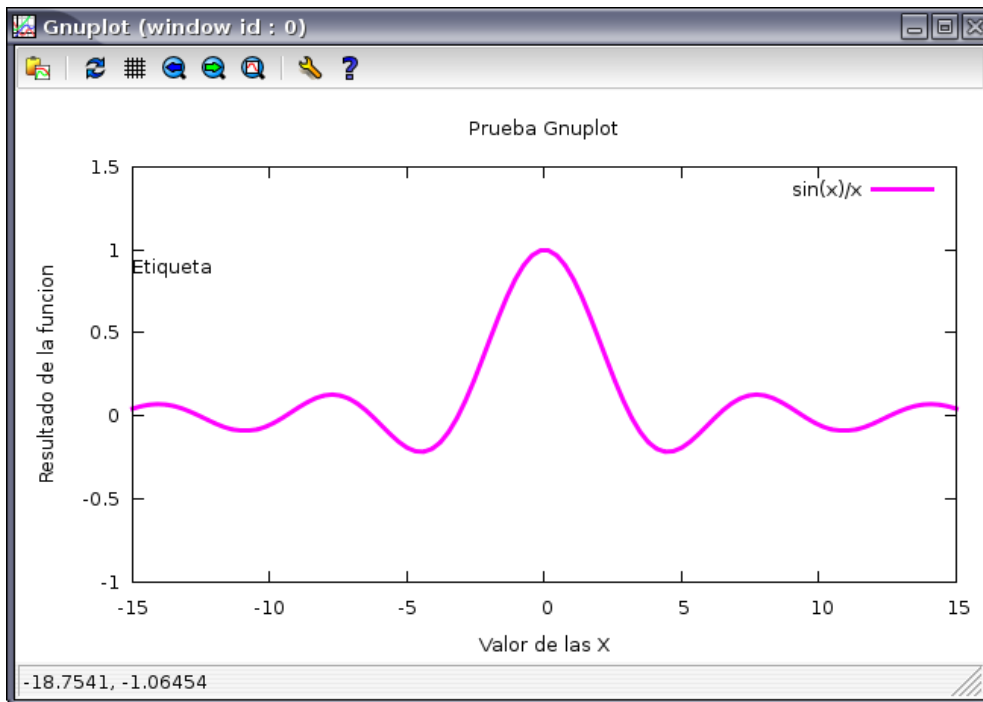
Terminal
File Edit View Terminal Help
mec-unsj-prac-1.pdf  mec-unsj-teo-7.pdf  textos              wi6b.png
mec-unsj-prac-2.pdf  papers              wi1.png
mec-unsj-prac-3.pdf  programas           wi2.png
mec-unsj-prog.pdf    tesp2-tema-1-2010.pdf  wi3a.png
!
gnuplot> !ls
alumnos              mec-unsj-teo-1.pdf  tesp2-tema-2-2010.pdf  wi3b.png
cartel-unsj.doc      mec-unsj-teo-2.pdf  tesp2-tema-3-2010.pdf  wi4a.png
curso-doc-1.pdf      mec-unsj-teo-3.pdf  tesp2-tema-4-2010.pdf  wi4b.png
curso-doc-2.pdf      mec-unsj-teo-4.pdf  tesp2-tema-5-2010.pdf  wi5a.png
figuras              mec-unsj-teo-5.pdf  tesp2-tema-6-2010.pdf  wi5b.png
gnuplot-tut.doc      mec-unsj-teo-6.pdf  tesp2-tema-7-2010.pdf  wi6a.png
mec-unsj-prac-1.pdf  mec-unsj-teo-7.pdf  textos              wi6b.png
mec-unsj-prac-2.pdf  papers              wi1.png
mec-unsj-prac-3.pdf  programas           wi2.png
mec-unsj-prog.pdf    tesp2-tema-1-2010.pdf  wi3a.png
!
gnuplot> plot [-15:15][-1:1.5] sin(x)/x with points
gnuplot> plot [-15:15][-1:1.5] sin(x)/x with impulses
gnuplot> plot [-15:15][-1:1.5] sin(x)/x with boxes
gnuplot>
gnuplot>
gnuplot>
gnuplot> test
gnuplot>

```

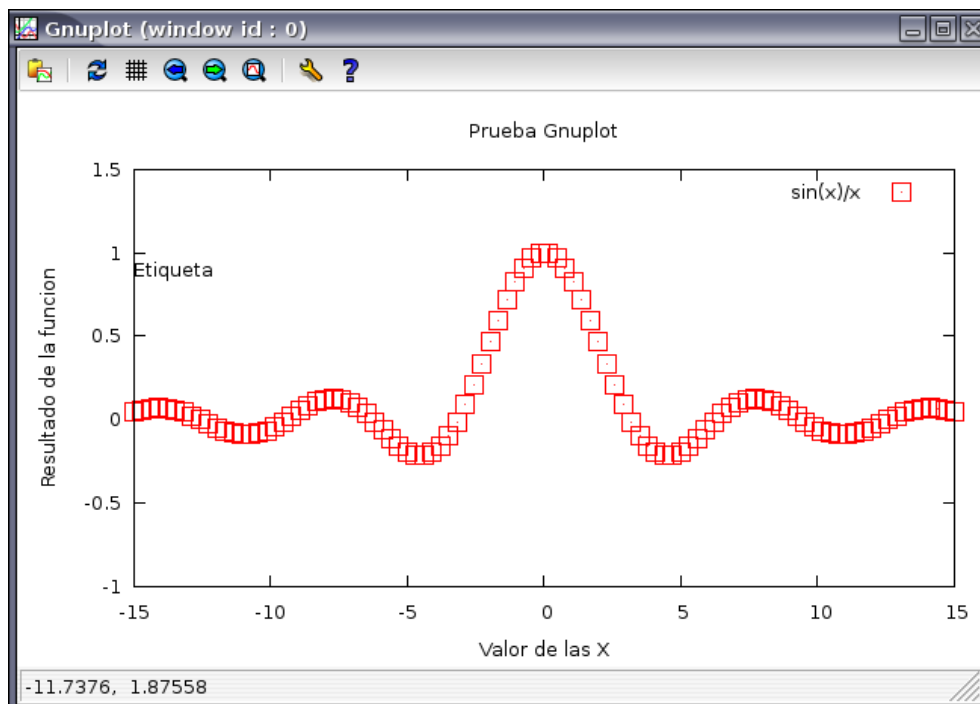


Por ejemplo, si queremos graficar la función con líneas un poco más gruesas y de color cian utilizamos

el comando “plot [-15:15][-1:1.5] sin(x)/x lw 3 lt 4”:



Si queremos graficar la función con puntos un poco más grandes que lo usual y representados con cuadrados usamos “plot [-15:15][-1:1.5] sin(x)/x w p pt 4 ps 2”:



Nótese que es posible abreviar los comandos si no hay ambigüedad en la escritura y que Gnuplot mantiene las leyendas de ejes, título, etc. hasta que no se borren con el comando “unset” (por ejemplo,

“unset title”) o cambien.

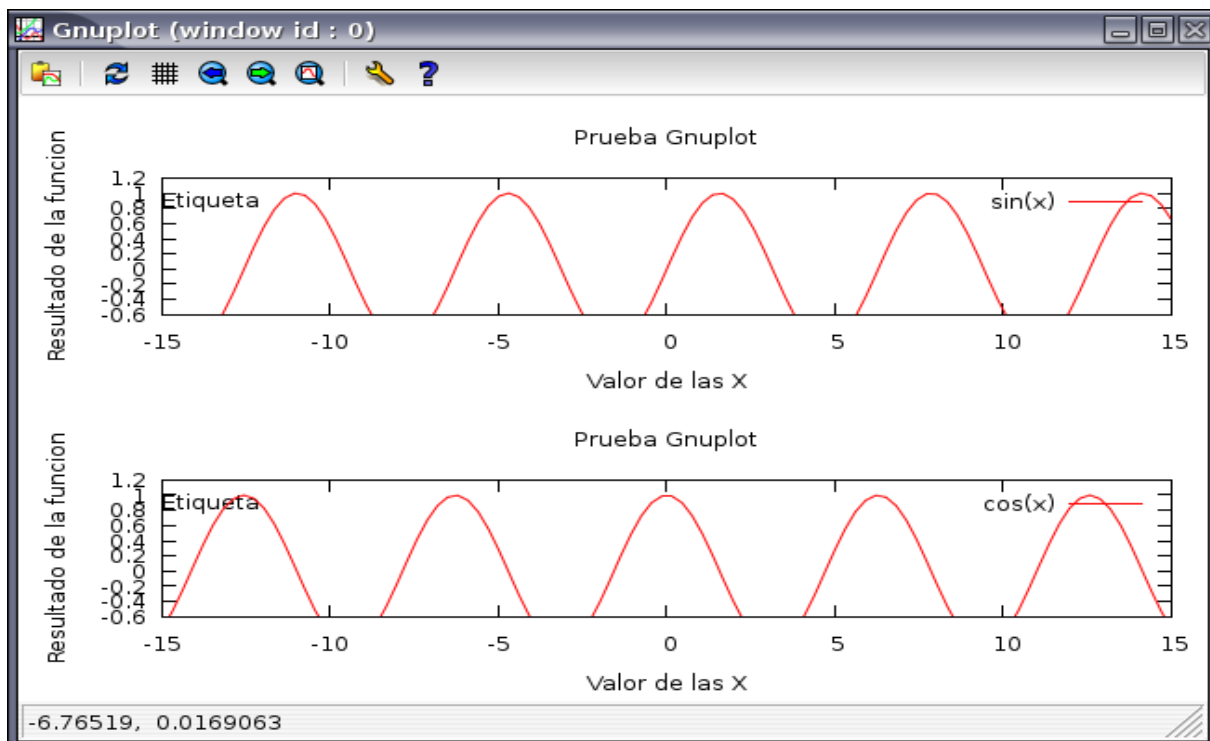
Gnuplot tiene la posibilidad de hacer gráficos múltiples en la misma ventana definiendo el modo “multiplot” y definiendo el área a utilizar. Por ejemplo, si queremos hacer un gráfico de dos funciones en la misma ventana:

```
Terminal
File Edit View Terminal Help
set size square 0.5,0.5

To make the graph twice as high as wide use:
set size ratio 2

See also
airfoil demo.

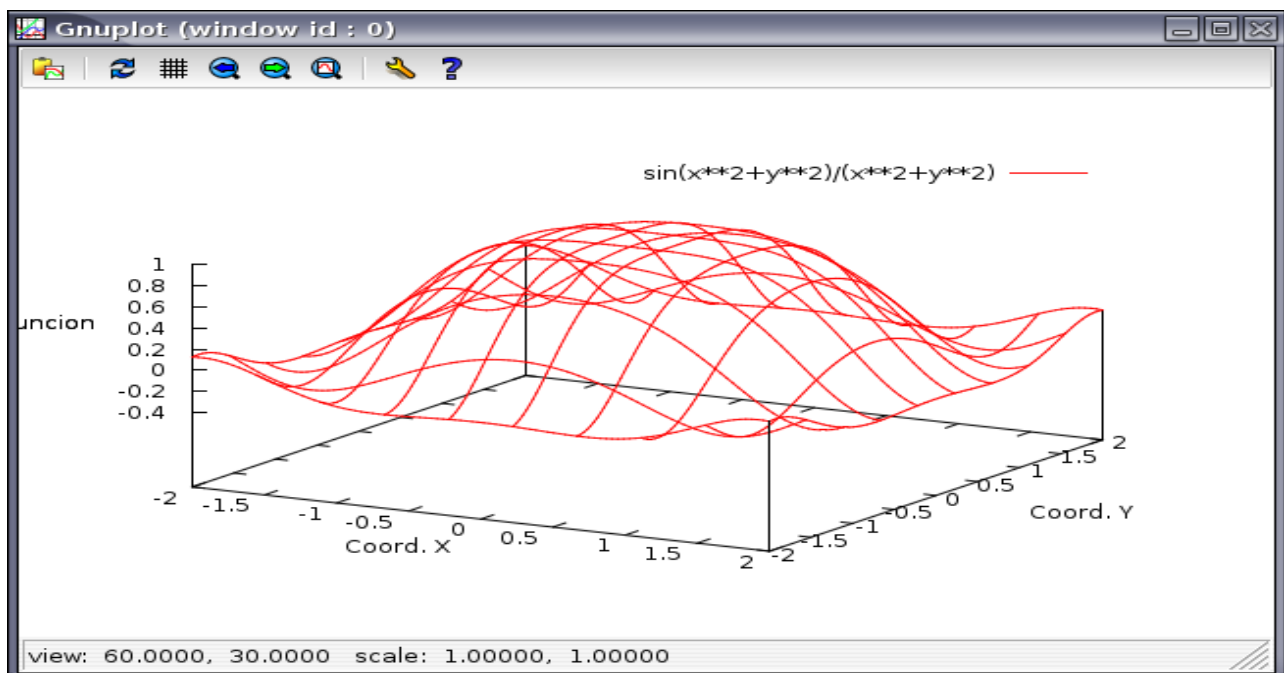
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot> set multiplot
multiplot> set size 1,0.5
multiplot> set origin 0.0,0.5
multiplot> plot sin(x)
multiplot> set origin 0.0,0.0
multiplot> plot cos(x)
multiplot> unset multiplot
gnuplot>
```



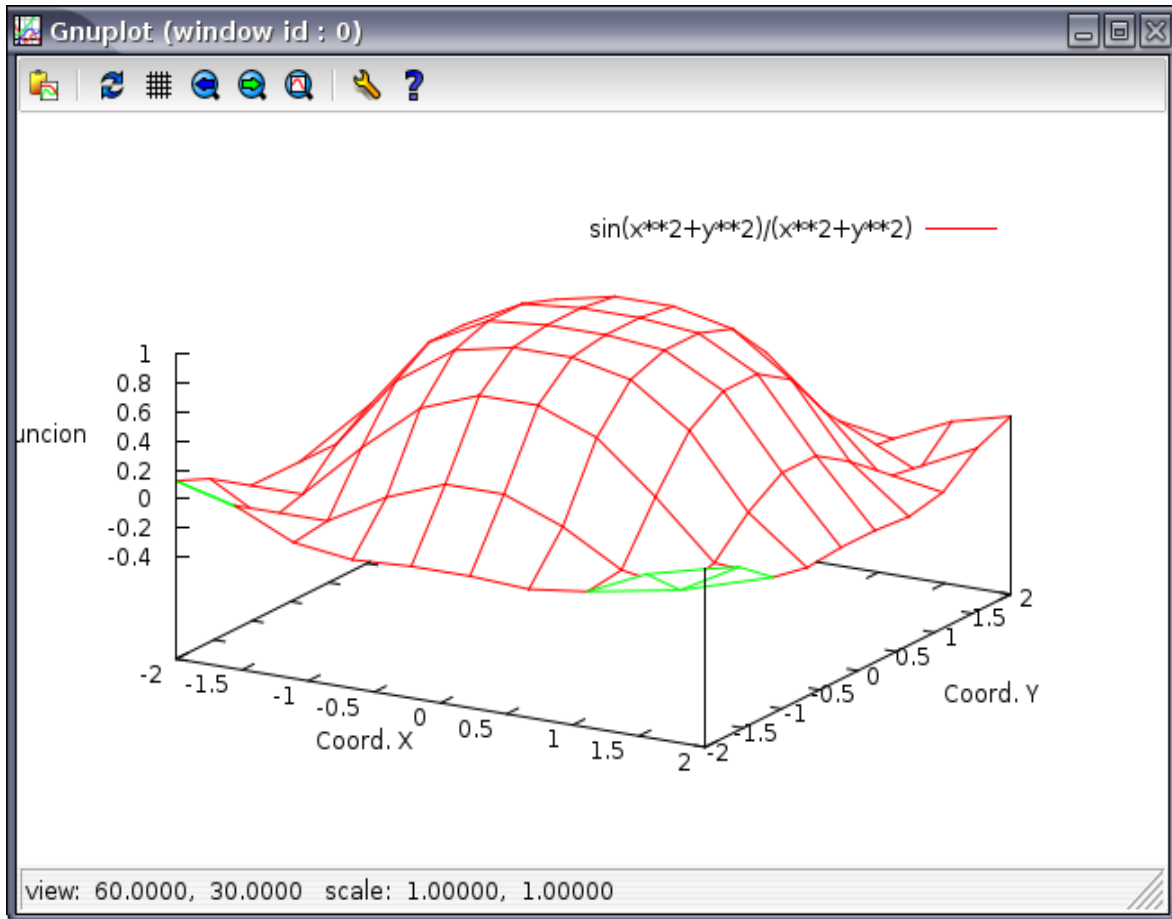
En este ejemplo, que el modo “multiplot” debe definirse al inicio y cerrarse al final (“unset multiplot”). A continuación se define el área a ocupar con el grafico mediante “set size xx,yy” que fija la extensión en ambas coordenadas (en el ejemplo, el gráfico tendrá una altura igual a la mitad del largo). Luego se fija donde está el origen del primer gráfico con “set origin xx,yy” y se grafica sin(x). Finalmente se repite para la función cos(x) cambiando el origen para que no se superpongan los gráficos.

Para graficar superficies se utiliza el comando “splot” que tiene una sintaxis similar a la de “plot” pero se debe tener en cuenta que existe un eje adicional. Por ejemplo, para graficar $\sin(x^2+y^2)/(x^2+y^2)$ es conveniente primero borrar las leyendas que utilizamos en los gráficos anteriores y definir rangos razonables para las coordenadas X e Y utilizando “set xrange” y “set yrange” (o incluir los nuevos rangos en el comando “splot”):

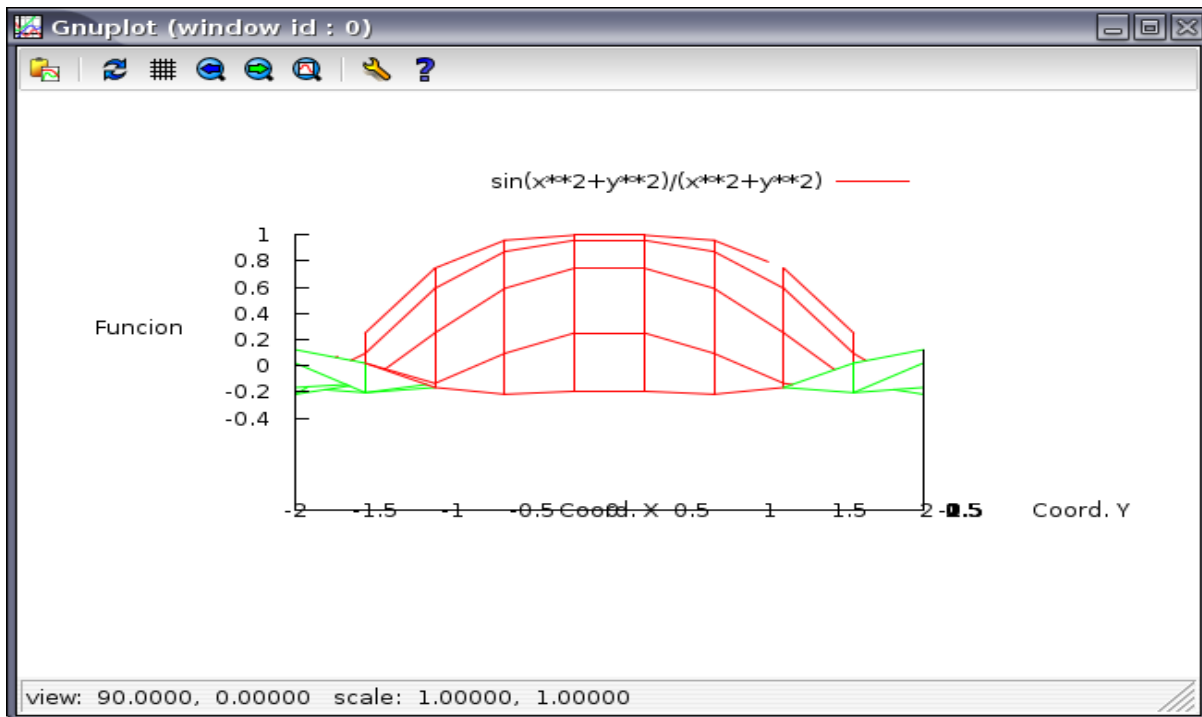
```
gnuplot>  
gnuplot>  
gnuplot>  
gnuplot>  
gnuplot>  
gnuplot>  
gnuplot>  
gnuplot>  
gnuplot>  
gnuplot>  
gnuplot>  
gnuplot>  
gnuplot>  
gnuplot>  
gnuplot>  
gnuplot>  
gnuplot>  
gnuplot> unset title  
gnuplot> set xlabel "Coord. X"  
gnuplot> set ylabel "Coord. Y"  
gnuplot> set zlabel "Funcion"  
gnuplot> set xrange [-2:2]  
gnuplot> set yrange [-2:2]  
gnuplot> splot sin(x**2+y**2)/(x**2+y**2)  
gnuplot>
```



En el gráfico se superponen las líneas que definen la superficie por delante y por detrás lo que puede generar alguna confusión. Para evitar esto se puede definir un modo para líneas ocultas mediante “set hidden3d” que oculta las líneas ocultas e identifica la parte superior e inferior de la superficie con diferentes colores:



El punto desde donde se esta observando la superficie graficada se define con “set view rotxx,rotzz” donde rotxx y rotzz son los ángulos de rotación alrededor de los ejes X y Z en grados. Inicialmente, el eje X coincide con el eje horizontal de la ventana, el eje Y con el vertical y el eje Z sale hacia afuera de la pantalla. Cuando uno usa el comando “splot” inicialmente grafica con un rotxx=60 grados y rotzz= 30 grados (ver esquina inferior izquierda de la ventana). Por ejemplo, si defino “set view 90,0” veo la superficie según una rotación de 90 grados alrededor del eje X (el eje X es el horizontal de la ventana, el Y se mete dentro de la ventana y el Z es el vertical de la ventana).

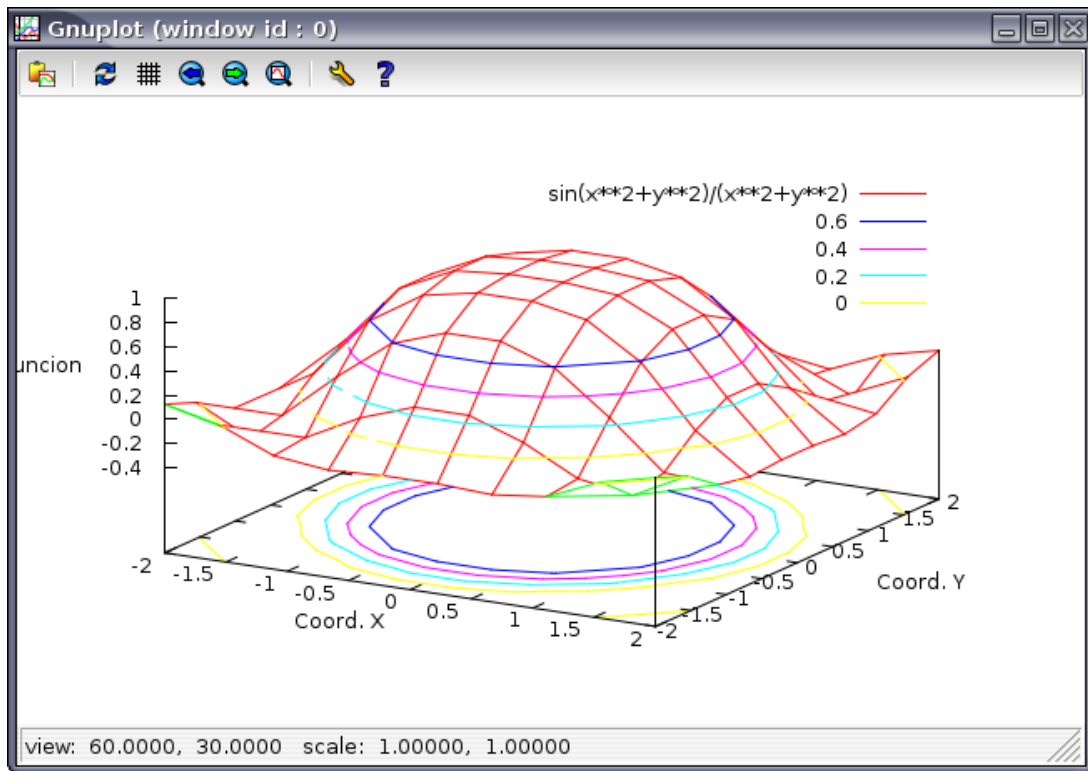


Si se necesita conocer en que puntos la superficie alcanza ciertos valores es útil definir valores de contorno que indiquen los puntos donde se logra ese valor. Para ello se dispone de “set contour” que permite marcar curvas de nivel en la superficie, en la base o en ambas, y de “set cntrparam” que permite definir el número de curvas de nivel a dibujar utilizando diferentes procedimientos. Por ejemplo, para volver al punto de vista original y definir curvas de nivel para los valores 0, 0.2, 0.4, y 0.6 en la superficie y en la base hacemos:

```

Terminal
File Edit View Terminal Help
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot> set view 60, 30
gnuplot> set cntrparam levels discrete 0,0.2,0.4,0.6
gnuplot> set contour both
gnuplot> splot sin(x**2+y**2)/(x**2+y**2)
gnuplot>

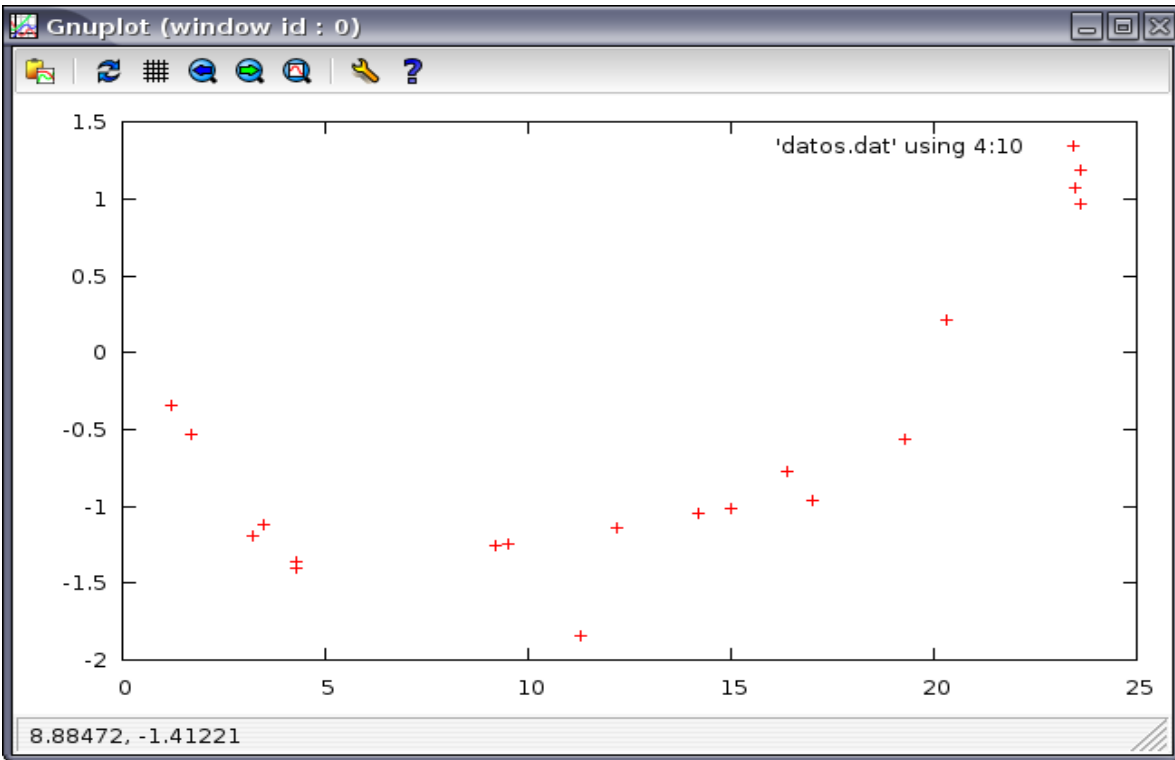
```

Cómo graficar datos desde un archivo:

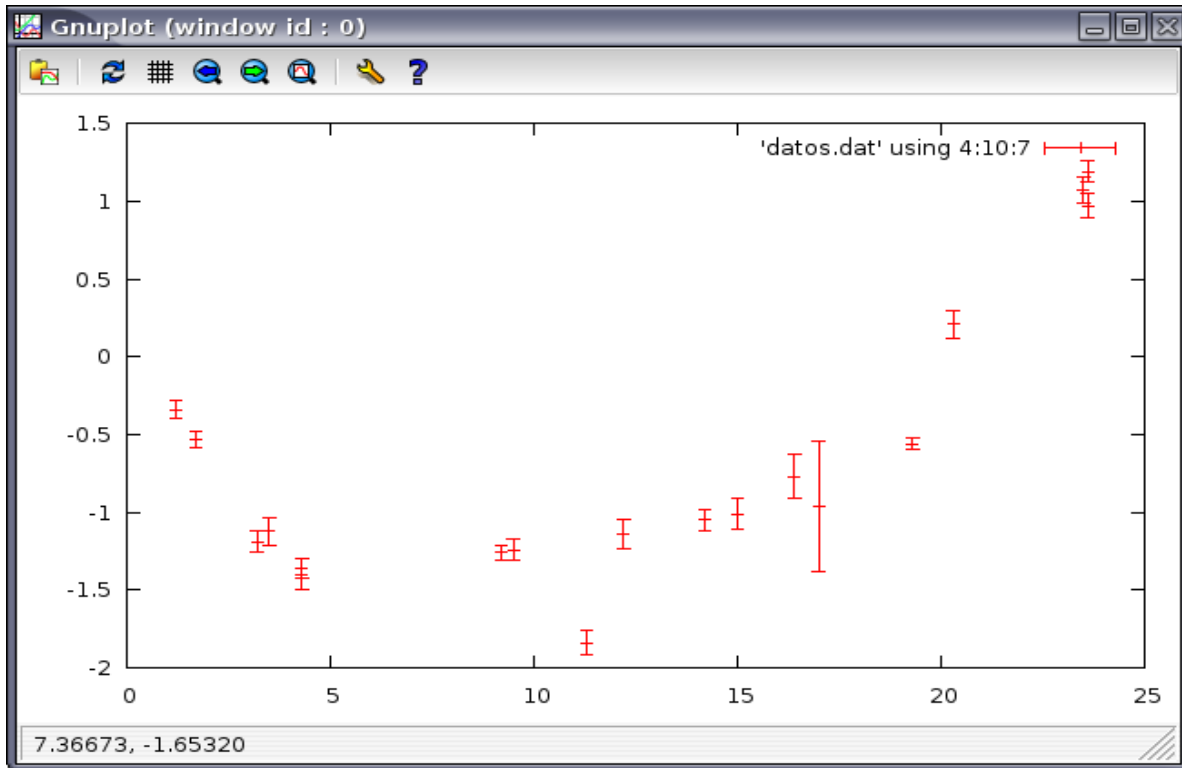
Además de graficar funciones, Gnuplot permite graficar datos guardados en archivos de texto comunes. Supongamos que tenemos un archivo llamado “datos.dat” que contiene mediciones de algunos objetos en un formato de varias columnas:

```
# Febrero 2011
# B, C y F
#
# Tholen Bus Lazzaro
# Ast Fecha exp ph th_s P% eP% th e_th Pr
#
#
# G C
1 2007Nov03 1080 4.3 122.9 1.40 0.10 124.8 2.1 -1.40 V
1 2007Nov03 1080 4.3 122.9 1.38 0.06 126.5 1.3 -1.36 V
1 2007Nov11 960 3.2 174.1 1.19 0.07 167.7 1.0 -1.19 V
1 2007Nov13 960 3.5 8.5 1.20 0.09 178.2 1.5 -1.12 V
#
# CF C
52 2007Jun15 480 9.2 73.5 1.29 0.05 79.2 1.0 -1.26 V
52 2007Jul10 480 1.7 44.8 0.53 0.05 43.2 2.7 -0.53 V
52 2007Jul12 480 1.2 25.8 0.34 0.06 29.6 5.1 -0.34 V
52 2007Aug13 480 9.5 86.8 1.26 0.07 82.0 1.5 -1.24 V
52 2008Nov04 960 14.2 59.9 1.06 0.07 65.0 2.0 -1.05 V
#
#
.....
```

Por otra parte, en la séptima columna (“eP%”) se indican los errores de la polarización, por lo cual sería interesante agregar esa información incluyendo barras de error para cada punto. Para ello debemos incluir el keyword “with” con el parámetro “errorbars”:

```
Terminal
File Edit View Terminal Help
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot> unset xlabel
gnuplot> unset ylabel
gnuplot> set autoscale
gnuplot> plot 'datos.dat' using 4:10
gnuplot> plot 'datos.dat' using 4:10:7 with errorbars
gnuplot>
```

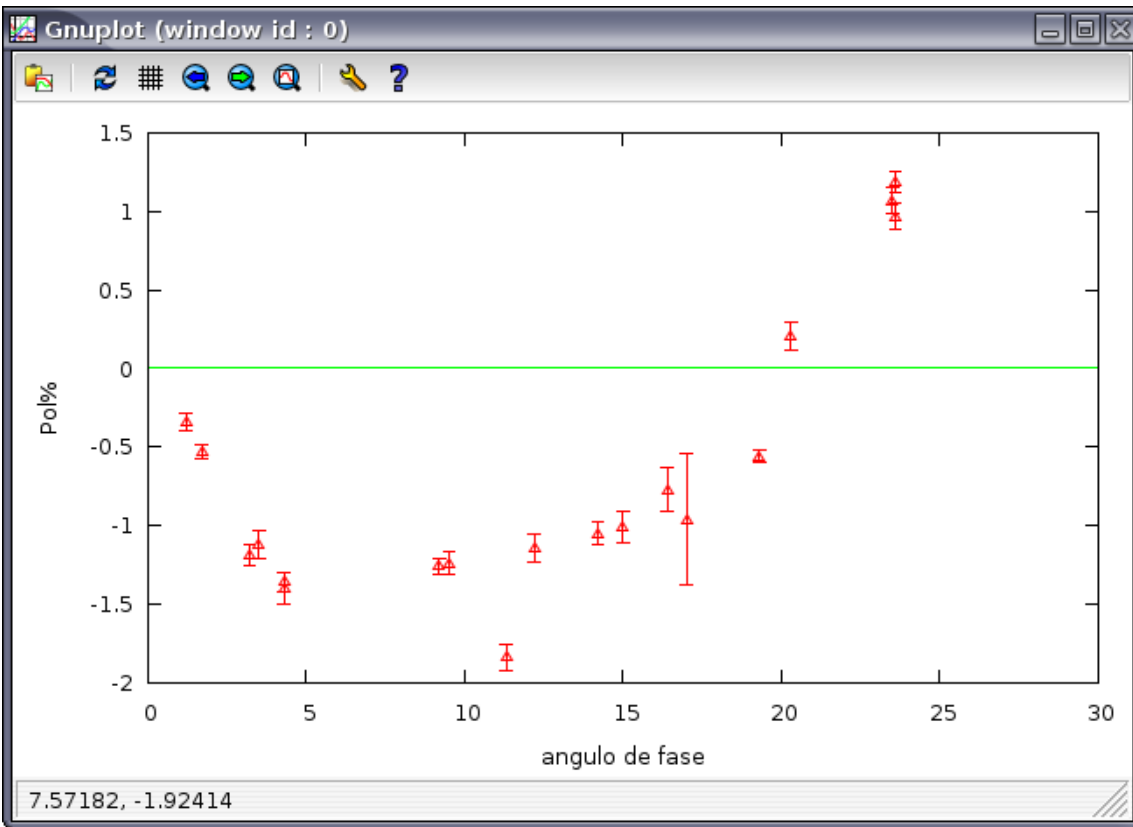


Para mejorar un poco el gráfico vamos a agregar leyendas en ambos ejes, sacaremos el indicador del rincón superior derecho, reemplazaremos los signos “+” por triángulos, asignaremos un rango de 0 a 30 grados para el ángulo de fase e incluiremos una línea que marque el valor cero de polarización reducida:

```

Terminal
File Edit View Terminal Help
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot> unset xlabel
gnuplot> unset ylabel
gnuplot> set autoscale
gnuplot> plot 'datos.dat' using 4:10
gnuplot> plot 'datos.dat' using 4:10:7 with errorbars
gnuplot> set xlabel "angulo de fase"
gnuplot> set ylabel "Pol%"
gnuplot> unset key
gnuplot> plot 'datos.dat' u 4:10:7 with e pt 8, 0.
gnuplot> plot [0:30][] 'datos.dat' u 4:10:7 with e pt 8, 0.
gnuplot>

```

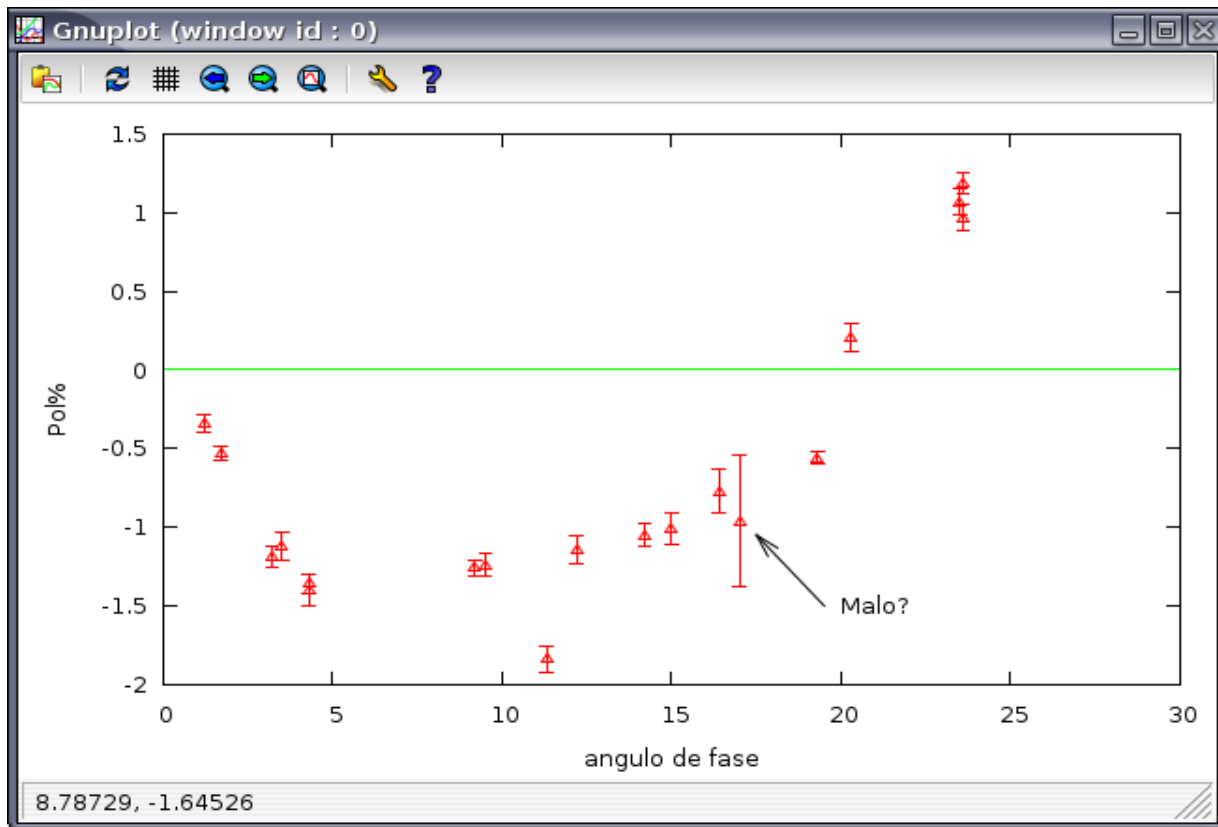


El punto que aparece en un ángulo de fase de 17 grados tiene una gran barra de error así que tal vez convendría marcarlo de algún modo. Para eso se puede definir etiquetas con “set label” y flechas con “set arrow”. Por ejemplo:

```

Terminal
File Edit View Terminal Help
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot> unset xlabel
gnuplot> unset ylabel
gnuplot> set autoscale
gnuplot> plot 'datos.dat' using 4:10
gnuplot> plot 'datos.dat' using 4:10:7 with errorbars
gnuplot> set xlabel "angulo de fase"
gnuplot> set ylabel "Pol%"
gnuplot> unset key
gnuplot> plot 'datos.dat' u 4:10:7 with e pt 8, 0.
gnuplot> plot [0:30][] 'datos.dat' u 4:10:7 with e pt 8, 0.
gnuplot> set label 1 "Malo?" at 20,-1.5
gnuplot> set arrow 1 from 19.5,-1.5 to 17.5,-1.05
gnuplot> plot [0:30][] 'datos.dat' u 4:10:7 with e pt 8, 0.
gnuplot>

```



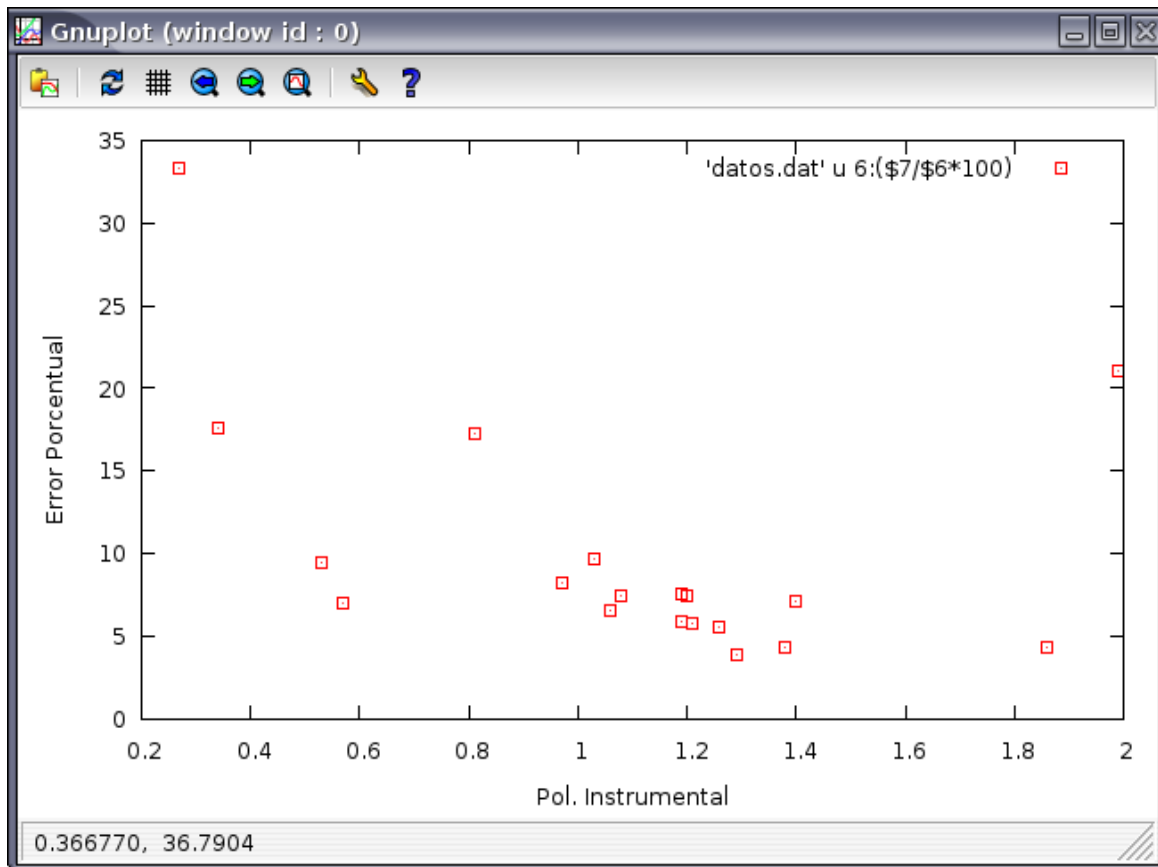
Tanto en “set label” como en “set arrow” el “1” identifica qué etiqueta o flecha es, lo que permite agregar otros elementos similares al gráfico indicándolos con números sucesivos, pero además me permite modificarlas sin necesidad de reescribir toda la instrucción nuevamente. Por ejemplo, si quiero cambiar la etiqueta “Malo?” por “Que es esto???” el comando “set label 1 “Que es esto???” es suficiente y no se requiere indicar de nuevo la posición.

También es posible para Gnuplot operar con las columnas de un archivo de datos. En las columnas 6 y 7 del archivo “datos.dat” aparecen los valores de polarización instrumental y el error. Si me interesa ver que error porcentual estoy cometiendo para una polarización dada debería graficar polarizacion vs. error/polarización, lo que implica que tengo que operar con las columnas del archivo. Para eso en los parámetros para el keyword “using” puedo aplicar cualquier función matemática para operar con columnas mediante el recurso de encerrar la expresión entre paréntesis y nombrar las columnas mediante la expresión “\$n” donde “n” es el número de columna:

```

Terminal
File Edit View Terminal Help
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot> unset xlabel
gnuplot> unset ylabel
gnuplot> set autoscale
gnuplot> plot 'datos.dat' using 4:10
gnuplot> plot 'datos.dat' using 4:10:7 with errorbars
gnuplot> set xlabel "angulo de fase"
gnuplot> set ylabel "Pol%"
gnuplot> unset key
gnuplot> plot 'datos.dat' u 4:10:7 with e pt 8, 0.
gnuplot> plot [0:30][] 'datos.dat' u 4:10:7 with e pt 8, 0.
gnuplot> set label 1 "Malo?" at 20,-1.5
gnuplot> set arrow 1 from 19.5,-1.5 to 17.5,-1.05
gnuplot> plot [0:30][] 'datos.dat' u 4:10:7 with e pt 8, 0.
gnuplot> set key
gnuplot> set xlabel "Pol. Instrumental"
gnuplot> set ylabel "Error Porcentual"
gnuplot> unset label 1
gnuplot> unset arrow 1
gnuplot> plot [0:30][] 'datos.dat' u 6:($7/$6*100) pt 4
gnuplot> plot 'datos.dat' u 6:($7/$6*100) pt 4
gnuplot>

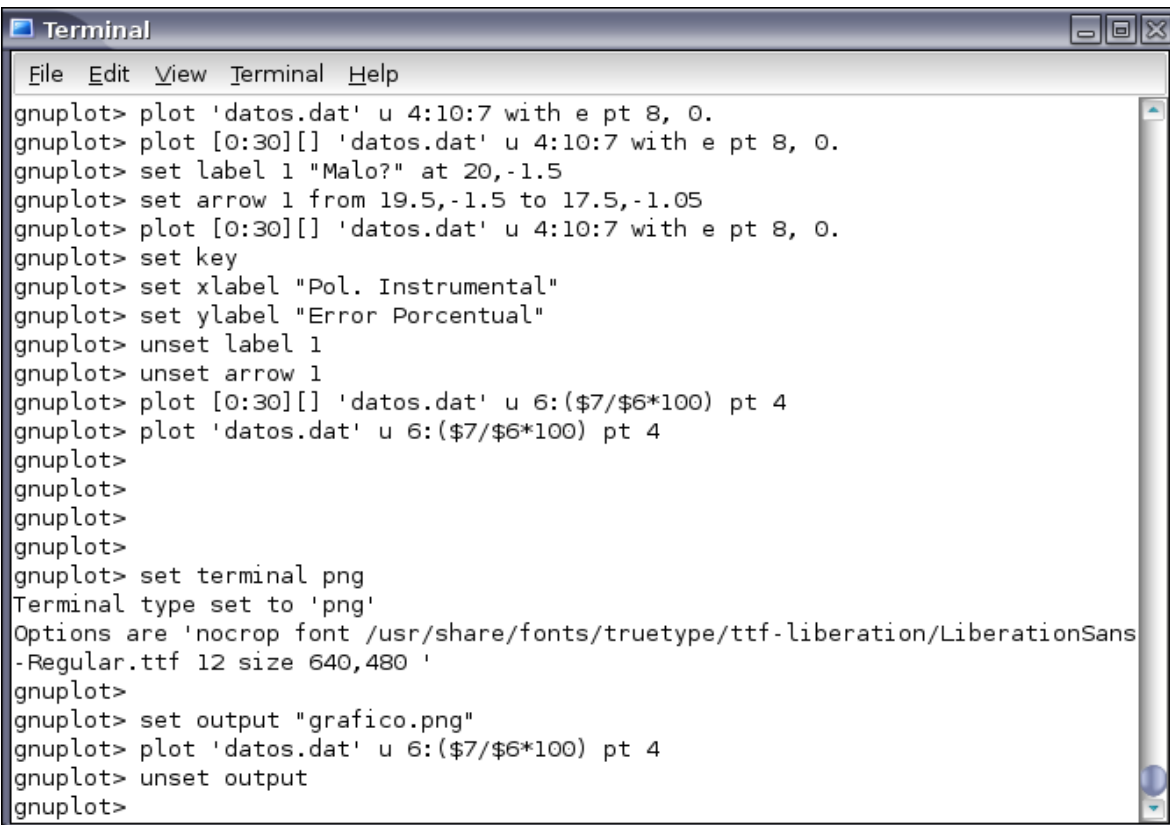
```



Cómo guardar los gráficos en diferentes formatos:

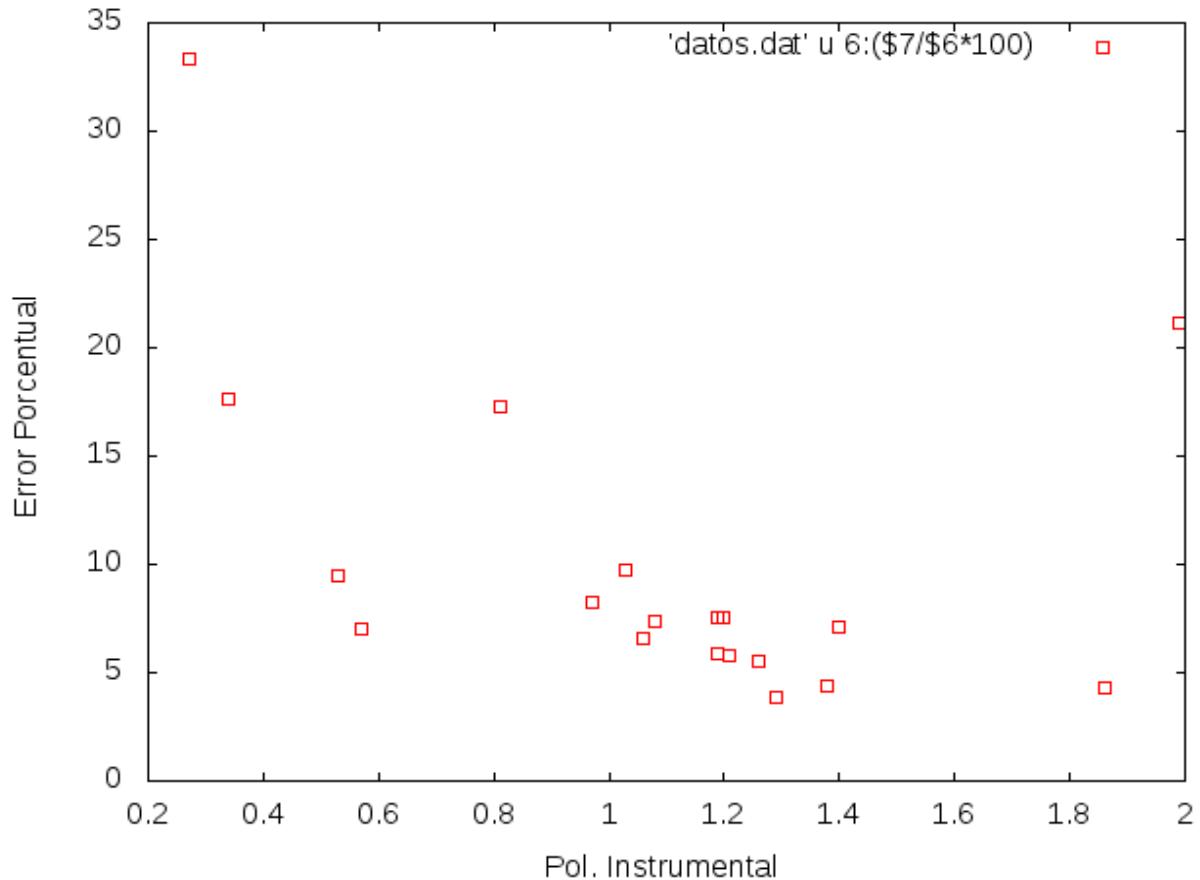
La posibilidad que tiene Gnuplot de definir terminales diferentes permite lograr salidas a gráficos con diferentes formatos. Los formatos más utilizados son el PNG (o JPEG) y el POSTSCRIPT.

Para crear el archivo con el gráfico se debe fijar el tipo de terminal que se desea con “set terminal <tipo de terminal>” y luego abrir un archivo de salida con “set output 'archivo.ext’”. Cada tipo de terminal tiene diferentes keywords que permiten definir el tipo de letra, el tamaño, si acepta caracteres especiales, etc. (consultar el help). Por ejemplo, para crear un archivo de nuestro último gráfico en formato PNG, hacemos:

A screenshot of a terminal window titled "Terminal" with a menu bar (File, Edit, View, Terminal, Help). The terminal displays a series of gnuplot commands and their outputs. The commands include plotting data from 'datos.dat', setting labels and arrows, and finally setting the terminal type to 'png' and the output file to 'grafico.png' before plotting again.

```
gnuplot> plot 'datos.dat' u 4:10:7 with e pt 8, 0.
gnuplot> plot [0:30][] 'datos.dat' u 4:10:7 with e pt 8, 0.
gnuplot> set label 1 "Malo?" at 20,-1.5
gnuplot> set arrow 1 from 19.5,-1.5 to 17.5,-1.05
gnuplot> plot [0:30][] 'datos.dat' u 4:10:7 with e pt 8, 0.
gnuplot> set key
gnuplot> set xlabel "Pol. Instrumental"
gnuplot> set ylabel "Error Porcentual"
gnuplot> unset label 1
gnuplot> unset arrow 1
gnuplot> plot [0:30][] 'datos.dat' u 6:($7/$6*100) pt 4
gnuplot> plot 'datos.dat' u 6:($7/$6*100) pt 4
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot> set terminal png
Terminal type set to 'png'
Options are 'nocrop font /usr/share/fonts/truetype/ttf-liberation/LiberationSans
-Regular.ttf 12 size 640,480 '
gnuplot>
gnuplot> set output "grafico.png"
gnuplot> plot 'datos.dat' u 6:($7/$6*100) pt 4
gnuplot> unset output
gnuplot>
```

y el gráfico obtenido en el archivo “grafico.png” es:



Hay que notar que si bien cerramos el archivo de salida con “unset output” la terminal válida sigue siendo PNG dado que no la modificamos nuevamente con “set terminal”. Esto quiere decir que si queremos ver los gráficos en una ventana debemos regresar a una terminal que así lo permita con, por ejemplo, “set terminal wxt”.

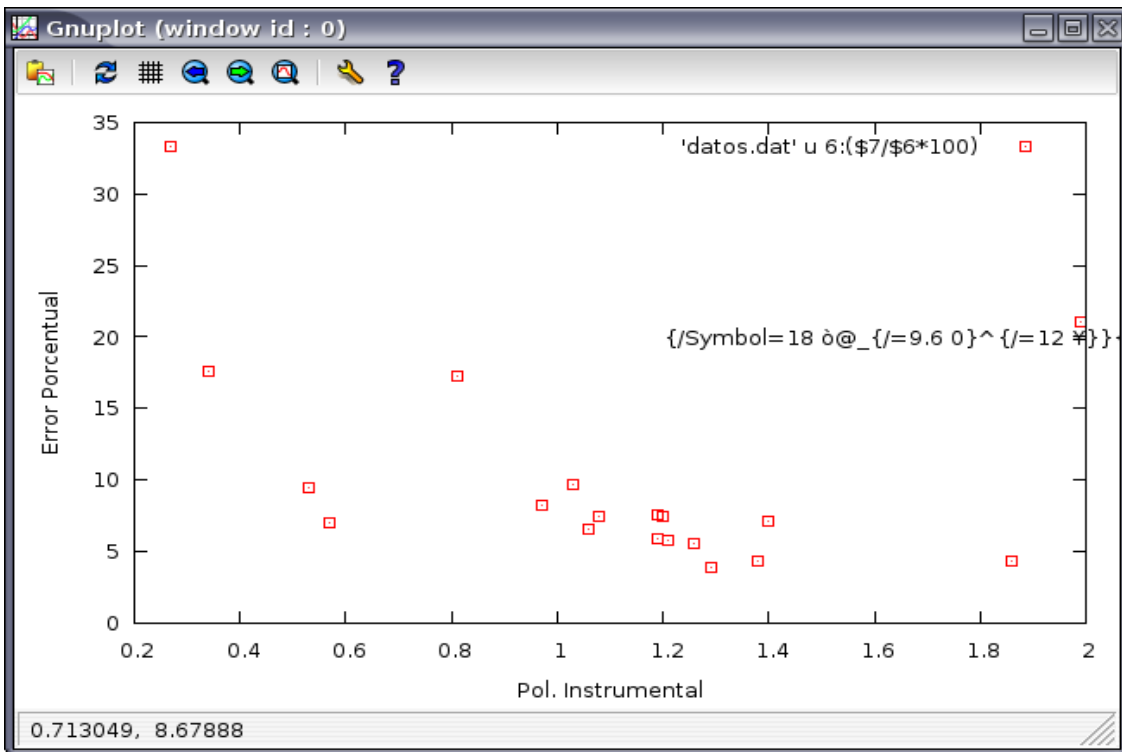
La ventaja de una terminal Postscript es que permite varias opciones muy útiles para crear gráficos de muy buena calidad utilizando letras y símbolos especiales. Por ejemplo, si al gráfico anterior le quiero agregar una etiqueta que contenga símbolos debo utilizar una codificación especial para que Gnuplot interprete que es lo que deseo hacer.

Primero, se debe asegurar que se utiliza la codificación correcta con “set encoding iso_8859_1”. Luego defino la etiqueta mediante “set label 2 '{/Symbol=18 \362@_}{/=9.6 0}^{/=12 \245}}{/Helvetica e^{ -{/Symbol m}^2/2} d}{/Symbol m = (p/2)^{1/2}}' at 1.2,20”. Si veo el gráfico en una terminal wxt la etiqueta no se verá muy bien:

```

Terminal
File Edit View Terminal Help
gnuplot> unset label 1
gnuplot> unset arrow 1
gnuplot> plot [0:30][] 'datos.dat' u 6:($7/$6*100) pt 4
gnuplot> plot 'datos.dat' u 6:($7/$6*100) pt 4
gnuplot>
gnuplot>
gnuplot>
gnuplot>
gnuplot> set terminal png
Terminal type set to 'png'
Options are 'nocrop font /usr/share/fonts/truetype/ttf-liberation/LiberationSans-
-Regular.ttf 12 size 640,480 '
gnuplot>
gnuplot> set output "grafico.png"
gnuplot> plot 'datos.dat' u 6:($7/$6*100) pt 4
gnuplot> unset output
gnuplot> set terminal wxt
Terminal type set to 'wxt'
Options are '0'
gnuplot> plot 'datos.dat' u 6:($7/$6*100) pt 4
gnuplot> set encoding iso_8859_1
gnuplot> set label 2 "{/Symbol=18 \362@_{/=9.6 0}^{\/=12 \245}}{/Helvetica e^{-{/
Symbol m}^2/2} d}{/Symbol m = (p/2)^{1/2}}" at 1.2,20
gnuplot> plot 'datos.dat' u 6:($7/$6*100) pt 4
gnuplot>

```

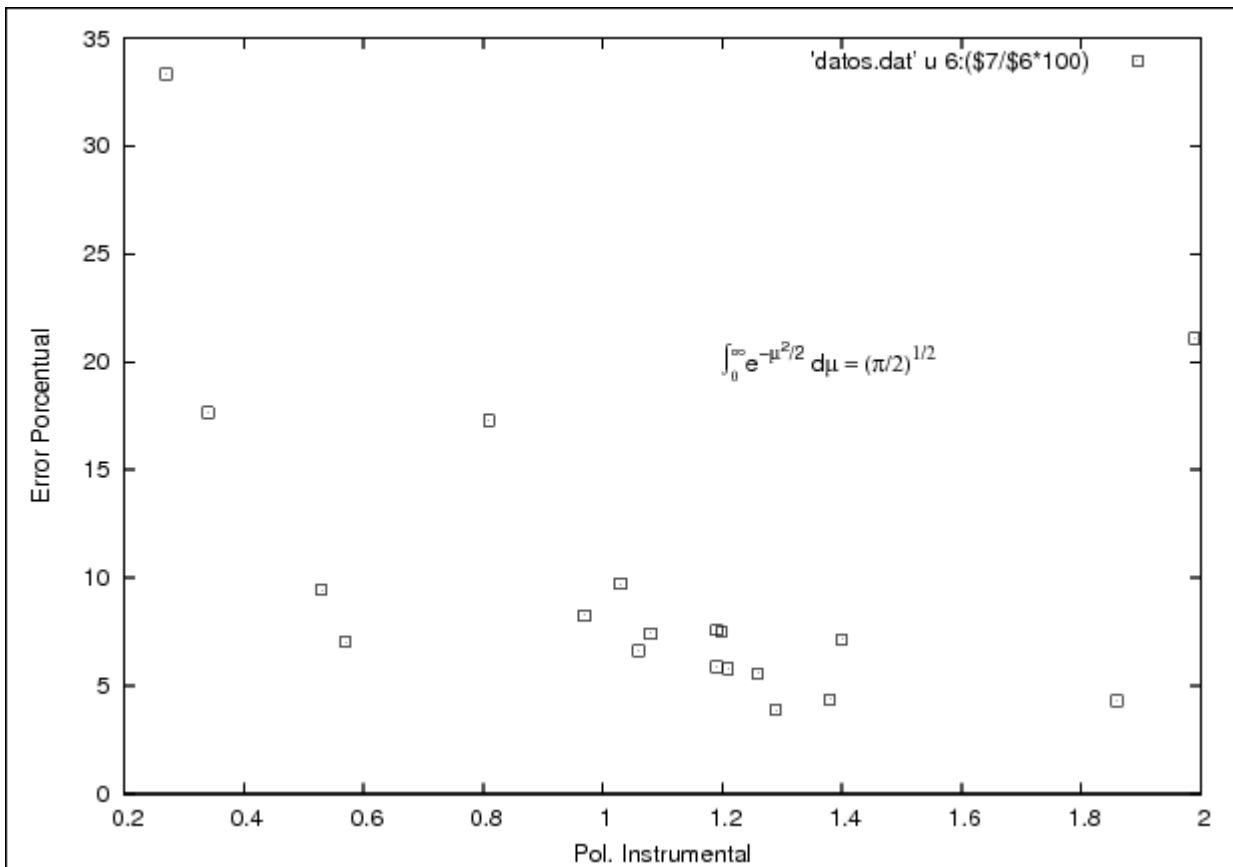


pero si utilizo una terminal Postscript mejorada obtengo:

```

Terminal
File Edit View Terminal Help
gnuplot>
gnuplot> set output "grafico.png"
gnuplot> plot 'datos.dat' u 6:($7/$6*100) pt 4
gnuplot> unset output
gnuplot> set terminal wxt
Terminal type set to 'wxt'
Options are '0'
gnuplot> plot 'datos.dat' u 6:($7/$6*100) pt 4
gnuplot> set encoding iso_8859_1
gnuplot> set label 2 "{/Symbol=18 \362@_{/=9.6 0}^{/=12 \245}}{/Helvetica e^{-{/
Symbol m}^2/2} d}{/Symbol m = (p/2)^{1/2}}}" at 1.2,20
gnuplot> plot 'datos.dat' u 6:($7/$6*100) pt 4
gnuplot>
gnuplot>
gnuplot> set terminal postscript enhanced
Terminal type set to 'postscript'
Options are 'landscape enhanced defaultplex \
leveldefault monochrome colortext \
dashed dashlength 1.0 linewidth 1.0 butt noclip \
palfuncparam 2000,0.003 \
"Helvetica" 14 '
gnuplot> set output "grafico.eps"
gnuplot> plot 'datos.dat' u 6:($7/$6*100) pt 4
gnuplot> unset output
gnuplot>

```



Si se quiere ver en una ventana el gráfico correcto es necesario utilizar otro tipo de terminal que no sea wxt. Usualmente se utiliza una terminal “X11” que se obtiene con “set terminal x11 enhanced” y que permite ver la etiqueta correctamente.

Otros comandos útiles:

Si observan algunos de los gráficos que hicimos verán que los valores indicados en los ejes no aparecen siempre con la misma cantidad de decimales o el mismo formato. Para corregir esto se puede definir el formato que se quiere utilizar en un eje dado mediante el comando “set format <eje> <formato>”, donde <eje> es el eje sobre el cual se aplicará (x, y, o z) y <formato> es un formato parecido al utilizado en Fortran que especifica el formato. Por ejemplo, “set format y “%.2f” “indica que los valores indicados en el eje Y aparecerán como reales con dos decimales. Para consultar los posibles formatos ver el help (“help format”).

Otro comando interesante es el comando “load”. Cuando uno tiene que hacer una serie de gráficos para un trabajo donde todos tienen el mismo formato, las mismas etiquetas, etc. y lo único que cambia es el conjunto de datos debería repetir una y otra vez la misma serie de comandos cambiando el nombre del archivo de datos y el archivo de salida del gráfico. Existe un comando en Gnuplot que permite ejecutar como instrucciones comandos incluidos en un archivo de texto que se lee con el comando “load”. Por ejemplo, si tenemos que graficar una y otra vez el mismo gráfico de ángulo de fase vs. polarización reducida para diferentes conjuntos de datos podemos hacer un archivo, al que llamaremos “instrucciones.gpl”, que contenga:

```
set size 0.8,0.8
set title “Fase vs. Polarizacion”
set xlabel “Fase (grados)”
set ylabel “Pol (%)”
set format y “%.1f”
set format x “%.0f”
set terminal postscript enhanced
set output “sale.eps”
plot [0:30] “datos.dat” u 4:10:7 w e pt 4, 0.
unset output
set terminal wxt
unset size
unset title
unset xlabel
unset ylabel
unset format y
unset format x
```

Luego de crear el archivo “instrucciones.gpl” se lo lee con “load “instrucciones.gpl” y se obtendrá el gráfico en el archivo de salida “sale.eps”. Cuando se tenga un nuevo conjunto de datos se reemplaza el nombre en el comando plot y se cambia el nombre del archivo de salida y se puede reutilizar con la garantía de que todos los gráficos serán iguales.