GnuPlot

Table des matières

1 Introduction

```
# apt-get install gnuplot gnuplot-doc
```

Sous emacs : `M-x info` puis `m gnuplot`

2 Exemple 1

```
$ x=0; while [ $x -lt 10 ]; do y=$(echo "e($x)" | bc -l); echo -e "$x\t$y"; x=$[$x+1]; done > exp.da

$ gnuplot
> plot "exp.dat" with line,"exp.dat" with impulses
> ^D

$ gnuplot << EOF > test.png
set terminal png
plot "exp.dat" with line,"exp.dat" with impulses
EOF

$ qiv test.png
```

3 Exemple 2

- Données (sous bash)

  ```
  min=-5
  max=5
  step=0.1

  # cos
  x=$min; \
  while [ $(echo "$x<=$max" | bc) -eq 1 ]; do \
   y=$(echo "c($x)" | bc -l); \
   echo -e "$x\t$y"; \
   x=$(echo $x+$step | bc); \
  done > cos.dat

  # sin
  x=$min; \
  ```

```
while [ $(echo "$x<=$max" | bc) -eq 1 ]; do \
 y=$(echo "s($x)" | bc -l); \
 echo -e "$x\t$y"; \
 x=$(echo $x+$step | bc); \
done > sin.dat

# arctan
x=$min; \
while [ $(echo "$x<=$max" | bc) -eq 1 ]; do \
 y=$(echo "a($x)" | bc -l); \
 echo -e "$x\t$y"; \
 x=$(echo $x+$step | bc); \
done > arc.dat
```

- Simple tracé

```
$ gnuplot << EOF
set terminal png
set output 'test.png'
plot "cos.dat" with line, \
     "sin.dat" with line, \
     "arc.dat" with line
EOF

$ qiv test.png
```

- Tracé quelque-peu amélioré

```
$ gnuplot << EOF
set output 'test.png'
set terminal png size 1000,500
set title "trigonometrie\ncos, sin et arctan";
set xlabel "degres";
set ylabel "amplitude";
set grid
plot "cos.dat" with line, \
     "sin.dat" with line, \
     "arc.dat" with line
EOF

$ qiv test.png
```

4 Approximation (fit)

- *test.dat*

```
0 0
1 2
2 4
3 8
4 16
5 32
6 64
7 128
```

- *test.plot*

```
set terminal png
set output 'test.png'
set xrange [0:10]
set yrange [0:100]

f(x) = a1 + b1*x + c1*x**2
fit [0:5346] f(x) "test.dat" via a1,b1,c1

g(x) = a2 * exp (x*b2)
fit [0:5346] g(x) "test.dat" via a2,b2

plot "test.dat", f(x), g(x)
```

- Tracé de 2 courbes : polynomiale et exponentielle

```
$ gnuplot test.plot
$ qiv test.png
```