

strona domowa gnuplot: <http://www.gnuplot.info/>

<https://ufkapano.github.io/gnuplot/index.html>

GNU PLOT: OPIS

Gnuplot jest programem do tworzenia wykresów funkcji i wizualizacji danych. Działa na systemach Linux, MS Windows i innych platformach.

Możliwe są dwa tryby pracy z gnuplotem: interaktywny i skryptowy.

TRYB INTERAKTYWNY

Po instalacji programu, uruchomiamy go przez: wgnuplot.exe lub gnuplot.exe

```
G N U P L O T
Version 5.4 patchlevel 1    last modified 2020-12-01

Copyright (C) 1986-1993, 1998, 2004, 2007-2020
Thomas Williams, Colin Kelley and many others

gnuplot home:      http://www.gnuplot.info
faq, bugs, etc:    type "help FAQ"
immediate help:    type "help" (plot window: hit 'h')
```

Terminal type is now 'qt'
gnuplot>

NAJWAŻNIEJSZE POLECENIA: plot (lub p), splot (sp), fit, print

ZMIENNE: nazwy domyślnych zmiennych niezależnych to x, y, z.

FUNKCJE WBUDOWANE (przykładowe):

```
+-----+-----+
| Funkcja   | Zwraca   |
+-----+-----+
| abs(x)   | wartość bezwzględna x, |x| |
| cos(x), sin(x), tan(x) | x jest w radianach |
| exp(x)   | funkcja eksponencjalna o podstawie e |
| log(x)   | logarytm o podstawie e |
| log10(x) | logarytm o podstawie 10 |
| norm(x) | funkcja rozkładu Gaussa |
| sqrt(x) | pierwiastek kwadratowy z x |
| int(x)  | część całkowita x |
| ceil(x) | najmniejszy int, ale nie mniejszy niż x |
| floor(x) | największy int, ale nie większy niż x |
+-----+-----+
```

Jak to działa:

```
gnuplot> plot sin(x)           # wykres funkcji 2D
gnuplot> plot sin(x), cos(x)   # dwie funkcje
gnuplot> set samples 1000      # zwiększenie próbkowania zmiennej x (gładszy linia)
```

TWORZENIE WŁASNYCH FUNKCJI

Możemy określić funkcję wprost w poleceniu plot:

```
gnuplot> plot 2*x+1
```

```
gnuplot> plot (x+1)**2
```

```
# Albo zdefiniować ją wcześniej; np. wielomian jednej zmiennej:
```

```
gnuplot> f(x) = x**6 -4*x**4 -x**2 +4
```

```
gnuplot> p f(x)
```

```
# Funkcja z parametrem.
```

```
# Parametr musi być określony zanim wykona się plot.
```

```
# Wartość parametru można zmieniać w instrukcji plot.
```

```
gnuplot> f(x) = sin(a*x)
```

```
gnuplot> a = 1
```

```
gnuplot> plot f(x)
```

```
albo:
```

```
gnuplot> plot a=0.1, f(x)
```

```
lub jeśli chcemy narysować funkcję dla wielu różnych parametrów:
```

```
gnuplot> plot a=0.1, g(x), a=0.2, g(x), a=0.3, g(x)
```

KONFIGURACJA WYKRESÓW

Parametry rysunku określamy za pomocą polecenia set. Powrót do domyślnych ustawień osiąga się za pomocą polecenia reset lub unset.

RANGES

Domyślny zakres x to [-10:10], y jest skalowany do danych. Zakresy można ustawiać w poleceniu plot lub osobno (set xrange, set yrange).

```
set xrange [-12:12]
```

```
set yrange [-3:3]
```

```
plot sin(x)
```

```
albo:
```

```
plot [-5:5] sin(x)
```

```
# zakres y domyślny
```

```
plot [] [-2:2] sin(x)
```

```
# zakres x domyślny
```

```
set autoscale
```

```
# powrót do ustawień domyślnych
```

```
set logscale y 10
```

```
# ustawienie osi y w tryb logarytmiczny (log10)
```

```
unset logscale
```

```
# powrót do trybu liniowego
```

GRID, BORDER, TICS

```
set grid
```

```
# włączenie siatki pomocniczej
```

```
unset grid
```

```
set border
```

```
# domyślnie jest ramka
```

```
unset border
```

```
# wyłącza ramkę
```

```
set tics out # domyślnie jest "in"
set xtics 0,.5,10 # od 0 do 10 co .5
set xtics auto # ustawienia domyślne
unset xtics # wiadomo, podobnie ytics
set tics font "Verdana,15" # przykładowe ustawienie typu i rozmiaru fontu
```

TITLE

Parametr title występuje w kilku różnych kontekstach, może odnosić się do różnych części rysunku.

```
gnuplot> set title "Wykres f(x)"
gnuplot> p a=0.1, f(x), a=0.2, f(x), a=0.3, f(x)

gnuplot> unset title # wyłączenie tytułu
```

```
set title "Funkcja sinus" # tytuł rysunku
set key title "Legenda" # tytuł do legendy
plot sin(x) title "sinus" # oznaczenie w legendzie
```

KEY

Pojedyncze wpisy w legendzie mogą być wyłączone w poleceniu plot przez notitle lub title "".

```
unset key # bez legendy

# set key
# {inside | outside}
# {lmargin | rmargin | tmargin | bmargin}
# {above | over | below | under} # też dozwolone
# {at <position>} # at 1,2,3.4
# {left | right | center}
# {top | bottom | center}
# {title "<text>"} {{no}enhanced}
# {{no}box { {linestyle | ls <line_style>}}
...
set key at 1,2 # legenda w punkcie (1,2)
set key outside # legenda na prawo od rysunku
set key below # legenda pod rysunkiem
set key box # domyślnie jest nobox
set key box lt 3 # ramka z linetype 3
set key right bottom
set key left top
set key right bottom title "Legenda" box # tytuł do legendy
set key default # powrót do domyślnych ustawień

set key font "Verdana,15" # przykładowe ustawienie typu i rozmiaru fontu
```

OPISY OSI

```
# Etykiety osi.
set xlabel "x1"
set ylabel "y1"

set xlabel "długość fali [nm]" font "Verdana,15" offset 0,-1 # przykładowe ustawienie tytułu
# osi x wraz z wyborem typu i
# rozmiaru fontu oraz wskazaniem
# przesunięcia tytułu względem osi
# (tu odsunięcie od osi)
```

ETYKIETY

Etykiety mają przyporządkowany znacznik (tag), czyli liczbę całkowitą. Domyślnie jest ustawiana najniższa niewykorzystana liczba. Dzięki znacznikowi możemy zdefiniować parametry etykiety (set label) lub usunąć etykietę (unset label).

```
gnuplot> help label
```

```
# Składnia:
# set label {<tag>}
# {"<label text>"}
# {at <position>}          # at x,y lub at x,y,z
# {left | center | right}  # domyślnie flush left
# {norotate | rotate {by <degrees>}}
# {font "<name>{,<size>"} }
# {noenhanced}             # wyłączenie trybu enhanced, gdy etykieta ma _ lub ^
# {front | back}           # przy front dane nie zakryją etykiety
# {textcolor <colourspec>}

set label 1 "Napis 1" at 1,10 center    # domyślnie left
set label 1 left                        # zmiana center na left

set label 2 "Napis 2" at 1,15 right
set label 2 at 1,17                    # będzie w nowym miejscu
set label 2 rotate by 90

unset label 2                          # wyłączenie etykiety nr 2
unset label                            # wyłączenie wszystkich etykiet
```

STYLE

Globalne ustawienia można przesłonić ustawieniem w plot.

```
# Globalne ustawienie stylu.
# Dostępne style: points (p), lines (l), linespoints (lp), impulses (i), dots (d),
# steps, fsteps, histeps,
# boxes

plot cos(x) w p, sin(x) w l, sin(2*x) w lp

# linetype 2 | lt 2          # drugi kolor domyślny linii; uwaga linetype określa też domyślny
                           # kształt symbolu, jeśli rysujemy używając points lub linespoints;
                           # polecenie linetype pozwala zdefiniować globalnie styl linii
                           # (kolor, grubość, kształt symbolu,...) używanych do wykresów
                           # (szczegóły zobacz w help)

# lista ustawień domyślnych:
# lt 1 (red)
# lt 2 (green)
# lt 3 (blue)
# lt 4 (magenta)
# lt 5 (cyan)
# lt 6 (sienna)
# lt 7 (orange)
# lt 8 (coral)

# linecolor 1 | lc 1        # kolor linii, tu pierwszy kolor domyślny;
```

oprócz kolorów z tabeli domyślnych, można wybierać też kolory po nazwach, np: lc rgb 'green'; listę kolorów z nazwami można znaleźć w internecie

```
# linewidth 2 | lw 2          # grubość linii, tu dwa razy szersza niż domyślna (multiplier)

p sin(x) w boxes lt 3 lw 3

# pointtype 3 | pt 3          # kształt punktów, tu trzeci kształt domyślny punktu
# pointsize 0.5 | ps 0.5      # rozmiar punktów, tu o połowę mniejsze niż domyślne (multiplier)

p sin(x) w p pt 2 lt 3 lw 3
p sin(x) w p pt 8 ps 2 lt 3 lw 3

set linewidth 2              # ustawienia globalne, dla wszystkich poleceń plot użytych później
set pointsize 3              #

set border lc "black" lw 2    # ustawienie cech ramki (osi) wykresu, tu kolor czarny oraz
                              # podwojona grubość

set margins 12, 3, 5, 1      # ustawienie marginesu - odległości między wykresem a krawędzią
                              # okna graficznego
```

SIZE

Polecenie przydaje się przy trybie multiplot.

```
# set size {{no}square | ratio r | noratio} {xscale,yscale}
# square - synonim do "ratio 1"
# ratio r - podaje się stosunek długości osi y do x

set size square 0.5,0.5      # kwadratowy, pomniejszony
set size ratio 2             # dwa razy wyższy niż szerszy
```

PRINT, QUIT, HELP

```
gnuplot> print pi, 3.5*sin(0.1)      # jak kalkulator
3.14159265358979 0.0998334166468282

+,-,*,**,sqrt(), ...             # operatory

gnuplot> help                       # ogólny help
gnuplot> help plot                   # help na dany temat

gnuplot> quit                        # lub q, Ctrl+D, exit, zakończenie pracy
```

Polecenia:

```
clear          # wyczyszczenie okna graficznego
reset          # przywrócenie wszystkich opcji ustawionych poleceniem set do wartości domyślnych
undefine a b   # wyczyszczenie zmiennych a i b zdefiniowanych przez użytkownika
```

WYKRESY Z DANYCH ZEWNĘTRZNYCH

Plik z danymi powinien być w określonym formacie. Kolumny powinny być oddzielone białymi znakami (spacje lub tabulacje). Komentarz w pliku z danymi jest oznaczany przez znak #, wtedy gnuplot ignoruje pozostałą część wiersza.

```
# Dane z jednego pliku.
```

```
gnuplot> cd 'katalog roboczy'  
gnuplot> p 'dane.txt' u 1:2 w lp
```

```
unset key  
set grid
```

```
# Jeśli oś X zawiera czas, należy to wskazać przed użyciem plot.
```

```
set xdata time  
set timefmt "%H:%M:%S"  
set format x "%H:%M:%S"      #Inne format czasu, np.: "%H:%M" (godziny i minuty) lub  
                             "%Y-%m-%dT%H:%M" (rok, miesiąc, dzień, godzina, minuta).  
                             Format deklarowany musi być zgodny z formatem w pliku z danymi.  
                             Dokładny opis: gnuplot.sourceforge.net/docs\_4.2/node274.html
```

```
set xtics rotate by 20 right  #Polecenie opcjonalne. Obraca opis osi o zadany kąt. Przydatne w  
                             przypadku, gdy opisy są za długie i nachodzą na siebie.
```

```
set xlabel "Czas"  
set ylabel "Liczba gwiazd"  
set title 'Pomiary'
```

```
p 'plot_dane.txt' u 1:3:5 w ye  
# na wykresie dodane zostały błędy w osi Y
```

```
reset
```

```
# Jeśli chcemy wrócić do wykresów, na których oś X nie jest czasem, należy podać:
```

```
unset xdata  
unset format x
```

UWZGLĘDNIANIE BŁĘDÓW

```
# Rysujemy błędy Y. Należy podać 3 kolumny.  
# Gnuplot spodziewa się (x, y, ydelta).
```

```
gnuplot> p 'dane.txt' u 1:2:4 w yerrorbars      # w ye  
gnuplot> p 'dane.txt' u 1:2:(abs($2-$3)) w yerrorbars  #UWAGA: to dla przypadku, gdy kol. 2 i 3  
                                                         zawiera zakres skrajnych wartości y:  
                                                         y_min i y_max  
gnuplot> p 'dane.txt' u 1:2:(0.1) w yerrorbars      #UWAGA: to dla przypadku, gdy plik z  
                                                         danymi nie zawiera błędów i chcemy je  
                                                         podać przy tworzeniu wykresu (tu: 0.1)
```

```
# Rysujemy błędy X. Należy podać 3 kolumny.  
# Gnuplot spodziewa się (x, y, xdelta).
```

```
gnuplot> p 'dane.txt' u 1:2:(60) w xerrorbars  # w xe, 60 - oznacza 60 sekund
```

```
# Rysujemy błędy X i Y. Należy podać 4 kolumny.
```

```
# Gnuplot spodziewa się (x, y, xdelta, ydelta).
```

```
gnuplot> p 'dane.txt' u 2:3:4:5 w xyerrorbars # w xye #UWAGA: tu oś X nie jest czasem
```

POLECENIE FIT

Polecenie fit umożliwia dopasowanie zdefiniowanej przez nas funkcji do zbioru danych pomiarowych. Dopasowanie opiera się na implementacji algorytmu najmniejszych kwadratów Marquardt-Levenberg.

Aby wspomóc algorytm dopasowujący można podać parametry początkowe, z których algorytm wystartuje w procesie optymalizacji. Domyślna wartość startowa każdego parametru to 1. Jeżeli chcemy określić dokładność optymalizacji, to można to zrobić poprzez ustawienie parametru FIT_LIMIT (zmiana wyliczanej wartości pomiędzy dwoma iteracjami) lub FIT_MAXITER (maksymalna liczba iteracji).

Podanie zakresów pozwala ograniczyć zakres danych, które będą brane pod uwagę przy dopasowaniu. Standardowo są brane pod uwagę dwie kolumny danych [using 1:2]. Jeżeli podamy trzy kolumny (using 1:2:3), to trzecia kolumna interpretowana jest jako odchylenie standardowe s, które użyte jest jako waga punktu $1/s^{**2}$.

Wartości parametrów fitowanych powinny być zbliżone co do rzędu wielkości, aby fitowanie szybciej zbieżne.

```
gnuplot> help fit
```

```
# Składnia:
# fit {ranges} expression datafile {datafile-modifiers} via var1{,var2,...}

# ranges - [{min}:{max}]
# expression - zwykle funkcja wcześniej zdefiniowana przez użytkownika
# datafile - nazwa pliku z danymi, do których dopasowywana jest funkcja
# datafile-modifiers - można użyć using, jak w plot
# via - precyzuje które parametry mają być fitowane (var1, var2, ...)
# Można fitować najwyżej pięć parametrów.
```

Jeśli dopasowanie robimy do danych, z których jedna zmienna (X) jest czasem, to kolumnę zawierającą czas należy rozbić na osobne HH, MM (dni, sekundy, itd.). Przed dopasowaniem musimy czas wyrazić w jednej jednostce, np. w godzinach i ułamkach godziny (HH.hh) lub w dniach i ułamkach dni (DD.dd), itd.

Plik dane_fit.txt:

#Godzina	minuta	jasność	liczba	błąd jas.	błąd liczby
18	00	15.4	50	0.2	50
18	05	16.3	120	0.2	50
18	10	17.2	190	0.2	50
18	15	18.1	310	0.2	50
18	20	18.8	380	0.2	50
18	25	19.5	450	0.2	50
18	30	20.0	560	0.2	50
18	35	20.3	670	0.2	50
18	40	20.7	730	0.2	50
18	45	21.0	780	0.2	50
18	50	21.2	930	0.2	50
18	55	21.3	1000	0.2	50
19	00	21.3	1090	0.2	50

```
gnuplot> unset key
```

```
gnuplot> p 'dane_fit.txt' u ($1+$2/60.):($4):(1./60.):($6) w xye
```

Wyjaśnienie powyższego przykładu: oś X tworzona jest z kolumny 1 i 2, przy czym minuty są zamieniane na ułamki godziny (kol.1 + kol.2/60.). Znak \$ wskazuje, że chcemy użyć kolumny o danym numerze. Oś Y to kolumna 4 (można podać (\$4) lub 4). Niepewność pomiaru czasu podana jest jako (1/60.) część godziny, czyli 1 min. Niepewność pomiaru liczby gwiazd podana jest jako kolumna 6 znajdująca się w pliku 'dane_fit.txt'.

Funkcję, którą chcemy dopasować musimy wcześniej zdefiniować:

```
gnuplot> f(x)=a*x+b
```

```
gnuplot> fit f(x) 'dane_fit.txt' u ($1+$2/60.):($4) via a,b #polecenie dopasowania
```

iter	chisq	delta/lim	lambda	a	b
0	5.0470847500e+06	0.00e+00	1.32e+01	1.000000e+00	1.000000e+00
1	4.5691113363e+05	-1.00e+06	1.32e+00	3.719584e+01	2.807391e+00
2	4.4355943927e+05	-3.01e+03	1.32e-01	3.990663e+01	-1.017993e+01
3	3.8909289419e+05	-1.40e+04	1.32e-02	1.055701e+02	-1.236040e+03
4	9.4034750854e+03	-4.04e+06	1.32e-03	9.449675e+02	-1.690687e+04
5	3.0955690675e+03	-2.04e+05	1.32e-04	1.067820e+03	-1.920042e+04
6	3.0955555556e+03	-4.36e-01	1.32e-05	1.068000e+03	-1.920378e+04

iter	chisq	delta/lim	lambda	a	b
------	-------	-----------	--------	---	---

After 6 iterations the fit converged.
final sum of squares of residuals : 3095.56
rel. change during last iteration : -4.36496e-06

degrees of freedom (FIT_NDF) : 7
rms of residuals (FIT_STDFIT) = sqrt(WSSR/ndf) : 21.0291
variance of residuals (reduced chisquare) = WSSR/ndf : 442.222

Final set of parameters	Asymptotic Standard Error
=====	=====
a = 1068	+/- 32.58 (3.05%)
b = -19203.8	+/- 608.2 (3.167%)

correlation matrix of the fit parameters:

	a	b
a	1.000	
b	-1.000	1.000

Końcowym rezultatem dopasowania jest zestaw wartości dopasowanych parametrów „Final set of parameters” (tu: a i b) wraz z ich niepewnościami „Asymptotic Standard Error”.
UWAGA. Jeśli wykonujemy kolejne dopasowanie z tymi samymi parametrami (tu: a, b), to procedura fit zaczyna działanie od wartości, które były wyliczone w poprzednim dopasowaniu (w tym przypadku: 1068 i -19203.8), co może prowadzić do błędnego wyniku. Aby tego uniknąć należy usunąć dopasowane parametry: undefine a b

```
gnuplot> p 'dane_fit.txt' u ($1+$2/60.):($4):(1./60.):($6) w xye, f(x) #rysowanie danych wraz z dopasowaną funkcją.
```

```
gnuplot> p 'dane_fit.txt' u ($1+$2/60.):($4-f($1+$2/60.)) w p, 0 #rysowanie różnic (w osi Y) między danymi a dopasowaną funkcją, plus linia dla wartości zero; f($1+$2/60.) to wartości dopasowanej funkcji dla kolejnych wartości x w danych
```

Możemy dostać (i użyć) wartość dopasowanej funkcji dla dowolnej wartości x (wyrażonej w jednostkach użytych przy dopasowaniu f, tu: godziny), np.:

```
gnuplot> print f(18.8)  
874.62222187686
```

TRYB SKRYPTOWY/WSADOWY

Polecenia Gnuplota można zapisać w pliku tekstowych z rozszerzeniem .gnu, .gnp, .gp, lub innym. Plik taki jest skrypsem, który po wywołaniu w Gnuplot spowoduje wykonanie wszystkich zapisanych w nim poleceń. Wywołanie skryptu:

```
gnuplot> load 'nazwa_pliku'
```

Skrypty ułatwiają pracę przy wykonywaniu wykresów - jednym poleceniem load możemy w dowolnej chwili uzyskać gotowy wykres.

Przykładowy zawartość skryptu, który rysuje dwa wykresy w jednym oknie (polecenie multiplot). Poniższe linie należy przekopiować do pliku o nazwie np. skrypt.gnu i wywołać go z linii poleceń w Gnuplot.

```
# skrypt rysujący dwa wykresy #
# ustawienie parametrów wykresów #
unset key
set grid
set multiplot
set size 0.85,0.50
set origin 0.1,0.01
set xdata time
set timefmt "%Y-%m-%dT%H:%M"
set format x "%Y-%m-%dT%H:%M"
set xtics rotate by 20 right
set xlabel "czas"
set xrange ["2015-01-02T18:10":"2015-01-02T19:10"]
# wykonanie wykresu 1 #
set ylabel "liczba gwiazd"
set title 'Pomiary tubą'
p 'dane.txt' u 1:3:5 w ye
# wykonanie wykresu 2 #
set origin 0.1,0.5
set ylabel "jasność nieba [mpsas]"
set title 'pomiaru SQM'
p 'dane.txt' u 1:2:4 w ye
# przywrócenie wartości domyślnych #
unset multiplot
reset
```

Uwagi do okna graficznego gnuplot

Okno otwiera się automatycznie po użyciu komendy plot. Okno umożliwia:

- skopiowanie wykresu do pamięci podręcznej
- wydruk lub zapisanie wykresu na dysk w formacie .pdf, .svg lub .png
- ponowne narysowanie poprzedniego wykresu - powtórzenie ostatniej komendy plot (replot)
- włącznie/wyłączenie siatki (grid)
- automatyczne dostosowanie zakresu na osiach do zakresu danych (autoscale). Przydaje się np. po przypadkowym przesunięciu zakresu spowodowanym przekręceniem kółka myszy.
- i inne

Rozmiar okna można zmienić ręcznie (skalowanie myszą).