

Wizualizacja danych - Gnuplot

dr hab. Bożena Woźna-Szcześniak

Akademia im. Jan Długosza

bwozna@gmail.com

Laboratorium 2



O czym dziś będzie mowa

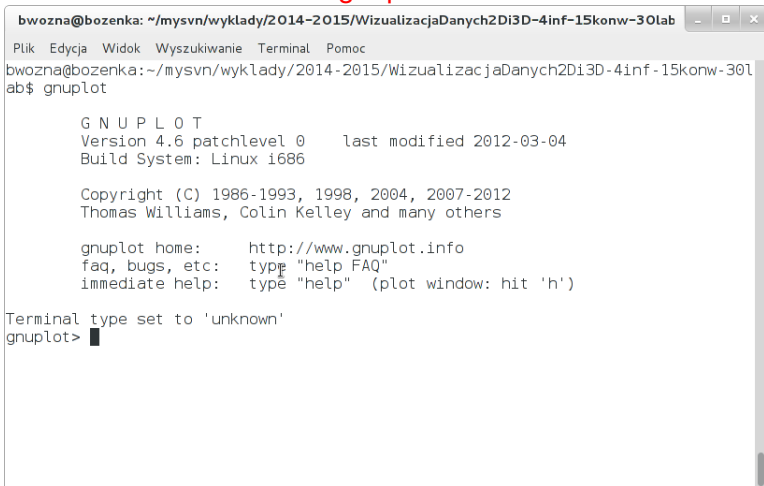
- Wywoływanie `gnuplot`a.
- Wykreślanie funkcji i danych.
- Zapisywanie i eksportowanie wykresów do typowych formatów plików graficznych.
- Obsługa wielu zestawów danych w jednym pliku.

- Najważniejsza komenda `gnuplot`a to polecenie `plot`.
- Komenda `plot` może być stosowany do wykreślenia zarówno funkcji (np., $\sin(x)$) jak i danych (zwykle zapisanych w pliku tekstowym).
- Komenda `plot` posiada wiele opcji i podpoleceń, za pomocą których można kontrolować wygląd wykresu oraz wyłuskiwać dane z pliku.

Wywoływanie gnuplota i pierwsze wykresy

- Gnuplota, jeśli jest zainstalowany, wywołujemy poleceniem

gnuplot



```
bwozna@bozenka: ~/mysvn/wyklady/2014-2015/WizualizacjaDanych2Di3D-4inf-15konw-30lab
Plik  Edycja  Widok  Wyszukiwanie  Terminal  Pomoc
bwozna@bozenka:~/mysvn/wyklady/2014-2015/WizualizacjaDanych2Di3D-4inf-15konw-30lab$ gnuplot

  G N U P L O T
  Version 4.6 patchlevel 0   last modified 2012-03-04
  Build System: Linux i686

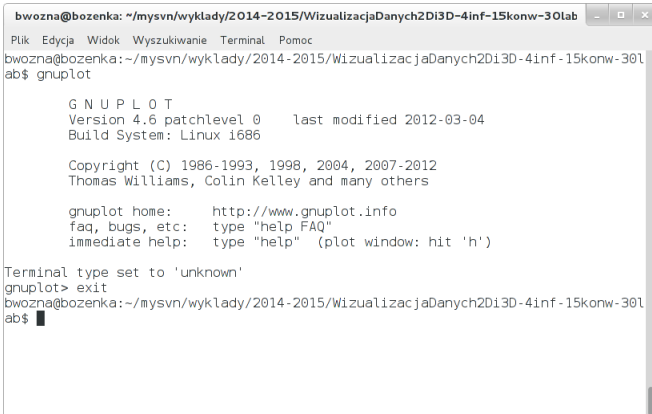
  Copyright (C) 1986-1993, 1998, 2004, 2007-2012
  Thomas Williams, Colin Kelley and many others

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

Terminal type set to 'unknown'
gnuplot> █
```

Wywoływanie gnuplota i pierwsze wykresy

- Polecenia wprowadzone po `gnuplot>` będą interpretowane jako polecenia gnuplota do czasu wydania polecenia `exit` lub `quit`, lub wprowadzenia znaku koniec-pliku (EOF), lub przez naciśnięcie `Control-D`.

A screenshot of a terminal window. The title bar reads "bwozna@bozenka: ~/mysvn/wyklady/2014-2015/WizualizacjaDanych2Di3D-4inf-15konw-30Lab". The terminal content shows the user typing "gnuplot" at the prompt "bwozna@bozenka:~/mysvn/wyklady/2014-2015/WizualizacjaDanych2Di3D-4inf-15konw-30Lab\$". The output displays the gnuplot version (4.6), build system (Linux i686), copyright information (1986-1993, 1998, 2004, 2007-2012), and home page (http://www.gnuplot.info). It also shows instructions for help and the terminal type set to 'unknown'. Finally, the user types "exit" and the prompt returns to the shell.

```
bwozna@bozenka: ~/mysvn/wyklady/2014-2015/WizualizacjaDanych2Di3D-4inf-15konw-30Lab
Plik Edycja Widok Wyszukiwanie Terminal Pomoc
bwozna@bozenka:~/mysvn/wyklady/2014-2015/WizualizacjaDanych2Di3D-4inf-15konw-30Lab$ gnuplot

  G N U P L O T
  Version 4.6 patchlevel 0      last modified 2012-03-04
  Build System: Linux i686

  Copyright (C) 1986-1993, 1998, 2004, 2007-2012
  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 set to 'unknown'
gnuplot> exit
bwozna@bozenka:~/mysvn/wyklady/2014-2015/WizualizacjaDanych2Di3D-4inf-15konw-30Lab$ █
```

Wywoływanie gnupłota i pierwsze wykresy

- Najprostszym polecenie kreślenia wykresów, które można wydać jest:

```
bwozna@bozenka: ~/mysvn/wyklady/2014-2015/WizualizacjaDanych2D130-4inf-15konw-30lab
Plik Edycja Widok Wyszukiwanie Terminal Pomoc
terminal type is unknown

gnuplot> show terminal

terminal type is unknown

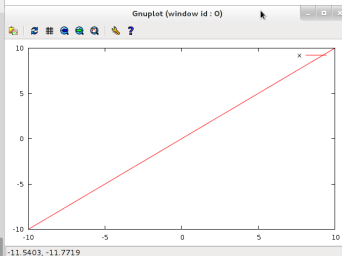
gnuplot> exit
bwozna@bozenka: ~/mysvn/wyklady/2014-2015/WizualizacjaDanych2D130-4inf-15konw-30lab$ gnuplot

G N U P L O T
Version 4.6 patchlevel 0 last modified 2012-03-04
Build System: Linux 1686

Copyright (C) 1986-1993, 1998, 2004, 2007-2012
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 set to 'wxt'
gnuplot> plot x
gnuplot>
```



Wywoływanie gnupłota i pierwsze wykresy

- Najprostszym polecenie kreślenia wykresów, które można wydać jest:

```
bwozna@bozenka: ~/mysvn/wyklady/2014-2015/WizualizacjaDanych2Di3D-4inf-15konw-30lab
Plik Edycja Widok Wyszukiwanie Terminal Pomoc

gnuplot> show terminal

terminal type is unknown

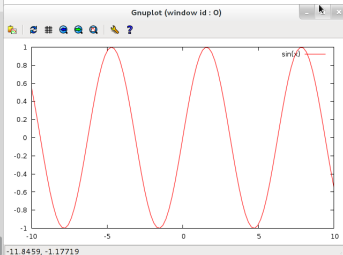
gnuplot> exit
bwozna@bozenka:~/mysvn/wyklady/2014-2015/WizualizacjaDanych2Di3D-4inf-15konw-30lab$ gnuplot

  G N U P L O T
  Version 4.6 patchlevel 0   last modified 2012-03-04
  Build System: Linux i686

  Copyright (C) 1986-1993, 1998, 2004, 2007-2012
  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 set to 'wxt'
gnuplot> plot x
gnuplot> plot sin(x)
gnuplot> █
```

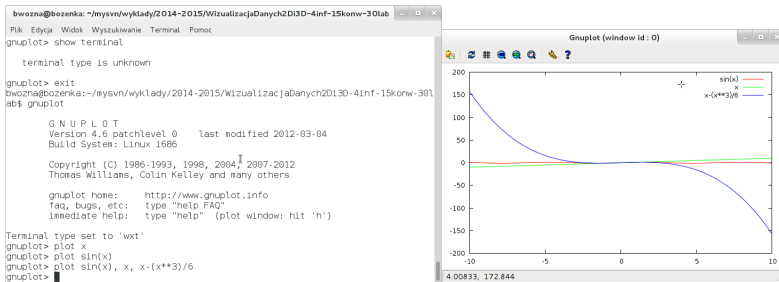


Wywoływanie gnuplota i pierwsze wykresy

- Proszę zwrócić uwagę, że `gnuplot` wybrał automatycznie domyślny zakres dla wartości x , tj. od -10 do $+10$, i obliczył zakres y zgodnie z wartościami kreślonej funkcji.

Wywoływanie gnupłota i pierwsze wykresy

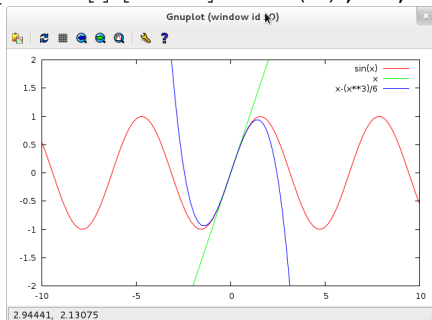
- Proszę zwrócić uwagę, że `gnuplot` wybrał automatycznie domyślny zakres dla wartości x , tj. od -10 do $+10$, i obliczył zakres y zgodnie z wartościami kreślonej funkcji.
- Założmy, że chcemy dodać więcej funkcji do wykreślenia wraz z sinus-em. Możemy to zrobić tak:



Wywoływanie gnupłota i pierwsze wykresy

- Zakres wartości y jest teraz bardzo duża w stosunku do poprzednich wykresów. Gnuplot automatycznie dostosowuje zakres y tak, aby dopasować wszystkie wartości funkcji.
- Zmiana zakresu jest możliwa, np:

```
gnuplot> plot [] [-2:2] sin(x), x, x-(x**3)/6
```



Wywoływanie gnupłota i pierwsze wykresy

- Zakres wartości danych podawany jest w nawiasach kwadratowych bezpośrednio po komendzie **plot**.
- Pierwsza para nawiasów definiuje zakres wartości dla x .
- Druga para nawiasów definiuje zakres wartości dla y .
- Pozostawienie pustych nawiasów powoduje, że `gnuplot` podstawia wartości domyślne, czyli:

```
plot [][] sin(x), x, x-(x**3)/6  
      jest równoważne  
plot sin(x), x, x-(x**3)/6
```

Wywoływanie gnuplota i pierwsze wykresy

- Składnia `gnuplot`-a dla wyrażeń matematycznych jest prosta i podobna do tych, które można znaleźć w innych językach programowania (Python, C, C++).

Operatory jednoargumentowe

Operator	Przykład	Typ argumentu	Opis
!	$!a$	int	logczne NIE
~	$\sim a$	int	dopełnienie
!	$a!$	int	silnia
-	$-a$	liczbowy	jednoargumentowy minus
+	$+a$	liczbowy	jednoargumentowy plus

Wywoływanie gnuplota i pierwsze wykresy

Operatory dwuargumentowe

Operator	Przykład	Typ argumentu	Opis
**	$a ** b$	liczbowy	potęgowanie
*	$a * b$	liczbowy	mnożenie
/	a / b	liczbowy	dzielenie
%	$a \% b$	int	dzielenie modulo
+	$a + b$	liczbowy	dodawanie
-	$a - b$	liczbowy	odejmowanie
<	$a < b$	liczbowy	mniejsze niż
<=	$a <= b$	liczbowy	mniejsze lub równe
>	$a > b$	liczbowy	większe niż
>=	$a >= b$	liczbowy	większe lub równe
==	$a == b$	liczbowy	równe
!=	$a != b$	liczbowy	różne

Wywoływanie gnuplota i pierwsze wykresy

Operatory dwuargumentowe

Operator	Przykład	Typ argumentu	Opis
&	$a \& b$	int	bitowe AND
^	$a \wedge b$	int	bitowe XOR
	$a b$	int	bitowe OR
&&	$a \& \& b$	int	logiczne AND
	$a b$	int	logiczne OR
.	$a . b$	string	konkatenacja łańcuchów
eq	$a \text{ eq } b$	string	równość łańcuchów

Operator trójargumentowe: $a ? b : c$

Obliczane jest wyrażenie a . Jeśli wartość a jest niezerowa, to obliczane jest wyrażenie b i zwracany jest jego wynik. W przeciwnym wypadku, obliczane jest wyrażenie c i zwracany jest jego wynik. **Czyli tak jak w C :) ...**

Wywoływanie gnuplota i pierwsze wykresy

Funkcje

Funkcja	Opis
$\text{sqrt}(x)$	Pierwiastek kwadratowy
$\text{exp}(x)$	Wykładnicza
$\text{log}(x)$	logarytm naturalny
$\text{log10}(x)$	logarytm przy podstawie 10

Wywoływanie gnuplota i pierwsze wykresy

Funkcje

Funkcja	Opis
$\sin(x)$, $\cos(x)$, $\tan(x)$ $\operatorname{asin}(x)$, $\operatorname{acos}(x)$, $\operatorname{atan}(x)$	funkcje trygonometryczne odwrotne
$\sinh(x)$, $\cosh(x)$, $\tanh(x)$ $\operatorname{asinh}(x)$, $\operatorname{acosh}(x)$, $\operatorname{atanh}(x)$	funkcje trygonometryczne funkcje hiperboliczne odwrotne
hline	funkcje hiperboliczne

Wywoływanie gnuplota i pierwsze wykresy

Funkcje

Funkcja	Opis
<i>abs(x)</i>	wartość bezwzględna
<i>floor(x)</i>	podłoga
<i>ceil(x)</i>	sufit
<i>int(x)</i>	część całkowita z x
<i>rand(x)</i>	generator liczb losowych
<i>sgn(x)</i>	funkcja znaku: -1 jeśli $x < 0$; 0 jeśli $x = 0$; 1 jeśli $x > 0$).

Kreślenie wykresów na podstawie danych z pliku

- Gnuplot odczytuje dane z plików tekstowych.
- Oczekiwane dane z pliku to dane liczbowe.
- Dane liczbowe przechowywane są w pliku, w kolumnach oddzielonych od siebie tzw. **białymi znakami**.
- Linie rozpoczynające się od znaku # są uważane za linie komentarzu i są ignorowane.
- Inne formaty pliku w specjalnych przypadkach też są dozwolone, ale o tym później

Kreślenie wykresów na podstawie danych z pliku

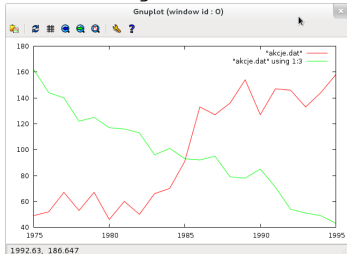
```
#akcje.dat
# Srednia cena akcji w dolarach za jedna akcje na rok
#ROK ABC XYZ
1975 49 162
1976 52 144
1977 67 140
1978 53 122
1979 67 125
1980 46 117
1981 60 116
1982 50 113
1983 66 96
1984 70 101
1985 91 93
1986 133 92
1987 127 95
1988 136 79
1989 154 78
1990 127 85
1991 147 71
1992 146 54
1993 133 51
1994 144 49
1995 158 43
```

Kreślenie wykresów na podstawie danych z pliku

Układ danych w pliku jest następujący:

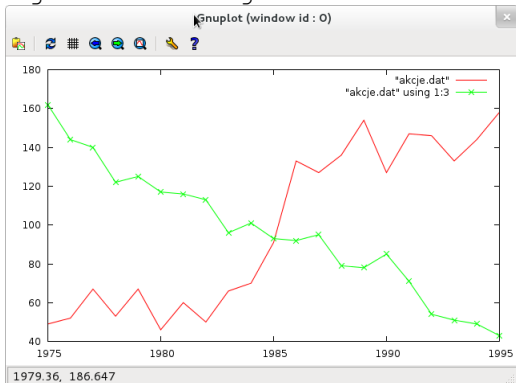
- wartość x jest w kolumnie 1
- wartość y jest w kolumnie 2
- dodatkowo wartości y odpowiadające każdej wartości x, są umieszczane w kolejnych kolumnach.
- nie zawsze tak być musi :) ...

```
plot "akcje.dat" using 1:2 with lines,  
      "akcje.dat" using 1:3 with lines
```



Kreślenie wykresów na podstawie danych z pliku

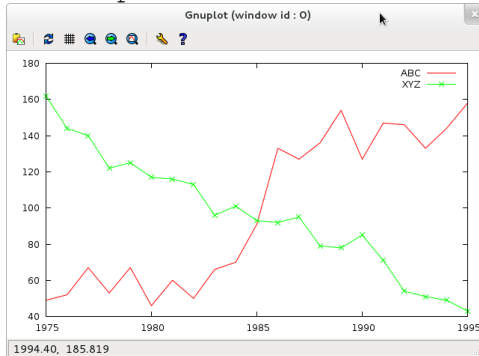
```
plot "akcje.dat" with lines,  
     "akcje.dat" using 1:3 with linespoints
```



Wykresy z automatycznym opisem danych. Ten automat powinno się jednak zmienić ...:)

Kreślenie wykresów na podstawie danych z pliku

```
plot "akcje.dat" using 1:2 title "ABC" with  
lines, "akcje.dat" using 1:3 title "XYZ"  
with linespoints
```

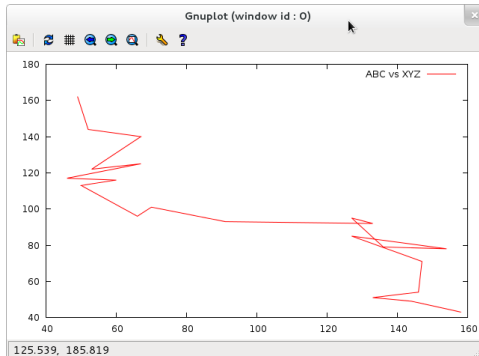


- Wykres bez automatycznego opisu danych. Polecenie **title**.
- **PAMIĘTAJ !** Komenda **title** występuje zawsze po komendzie **using**.

Kreślenie wykresów na podstawie danych z pliku

Nie zawsze x musi być w pierwszej kolumnie ...

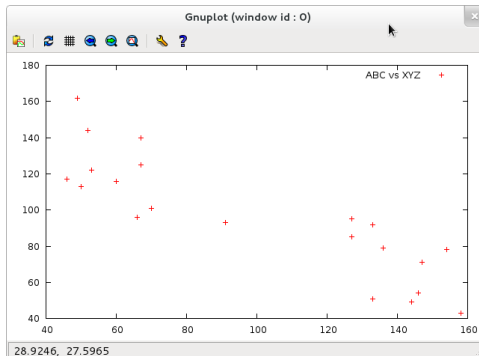
```
plot "akcje.dat" using 2:3 title "ABC vs XYZ"  
with lines
```



Kreślenie wykresów na podstawie danych z pliku

Nie zawsze x musi być w pierwszej kolumnie ...

```
plot "akcje.dat" using 2:3 title "ABC vs XYZ"  
with points
```



Kreślenie wykresów na podstawie danych z pliku

- Wykres jak powyższy to wykres punktowy i może pokazać zależności między dwoma zbiorami danych.
- W wykresie z poprzedniego slajdu widzimy wyraźną ujemną korelację: im akcje ABC miewają się lepiej, tym akcje XYZ miewają się gorzej.

Skróty i wartości domyślne

- Wszelkie polecenia, komendy lub opcje gnupłota mogą być zastąpione przez jednoznaczne skróty, np..

```
plot "akcje.dat" using 1:2 with lines,  
     "akcje.dat" using 1:3 with linespoints
```

można zastąpić przez

```
plot "akcje.dat" u 1:2 w l, "akcje.dat" u 1:3 w lp
```

lub też przez (domyślny plik, to ten ostatnio użyty):

```
plot "akcje.dat" u 1:2 w l, "" u 1:3 w lp
```

Uwaga ! Cudzysłowy, choć nie ma nazwy pliku, muszą pozostać.

Skróty i wartości domyślne

Jeszcze krócej (nie ma using, znaczy bierz **using 1:2**):

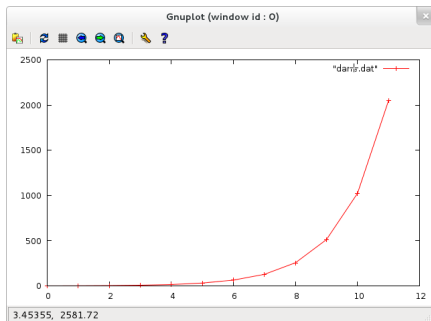
```
plot "akcje.dat" w 1, "" u 1:3 w 1p
```

Polecenie: Wpisz polecenia z poprzedniego i tego slajdu oraz porównaj wykresy !!!

Skróty i wartości domyślne

- Jeśli w pliku z danymi znajduje się tylko jedna kolumna, to gnuplot użyje tej kolumny dla wartości y, a za wartości x przyjmie liczby całkowite poczynając od zero.
- **Polecenie:** Sprawdź to na dowolnym przykładzie.

plot "dane.dat" w lp



Jak wygląda plik "dane.dat"?

Zapisywanie i eksportowanie

- Dwa sposoby, aby zapisać pracę gnuplota:
 - zapisujemy polecenia gnuplota do pliku tekstowego, np. `test.p`; można wtedy wygenerować wykres w dowolnym czasie i dowolnie wiele razy.
 - eksportujemy wygenerowany wykres do pliku w jednym z wielu obsługiwanych formatów plików graficznych; taki wykres można wydrukować lub umieścić go na stronach internetowych, w dokumentach tekstowych lub prezentacjach, ale nie można go już edytować.

Zapisywanie i ładowanie kommand

- Polecenia gnuplota mogą być zapisywane do pliku za pomocą następującego polecenia:

```
gnuplot> save "dane.p"
```

- Powyższa komenda pozwoli zachować bieżące wartości wszystkich opcji gnuplota wraz z ostatnio wykonaną komendą do określonego pliku.
- Wygenerowany plik może być później ponownie załadowany za pomocą komendy **load**.

```
gnuplot> load "dane.p"
```

- Efektem załadowania pliku jest wykonanie przez gnuplota wszystkich poleceń zawartych w tym pliku.
- Wygenerowany plik może być również załadowany za pomocą komendy **call**, ale on może znacznie więcej.

Eksportowanie wykresów

- Dla wykresu, który chcemy wygenerować, a następnie wyeksportować do pliku, musimy określić dwie rzeczy:
 - format wykresu (GIF, JPG, PNG, PDF, EPS i tak dalej)
 - urządzenie wyjściowe (plik lub ekran).
- W gnuplocie, ustawienia te wykonujemy za pomocą polecenia **set**.

```
# wybierz format pliku
set terminal png
# wybierz typ urządzenia wyjściowego
set output "mygraph.png"
```

Przepis na wygenerowanie pliku PNG

```
# wybierz wykres (np. tutaj funkcja kwadratowa)
gnuplot> plot x**2
# wybierz format pliku
gnuplot> set terminal png
# wybierz typ urządzenia wyjsciowego
gnuplot> set output "kwadratowa.png"
# powtorz ostatnio wykonana komende plot
gnuplot> replot
#przywroc pierwotne ustawienia terminala
#inaczej ciagle bedziesz drukowac do
#pliku "kwadratowa.png"
gnuplot> set terminal x11
#wyslij wynik na ekran
gnuplot> set output
```


Przydatny skrypt do eksportowania do pliku aktualnego wykresu

```
#zapisac biezace ustawienia terminala
set terminal push
# wybierz format pliku na PNG
set terminal png
#wybierz typ urzadzenia wyjsciwego zadanego parame
set output "$0"
# powtorz ostatnio wykonana komende plot
replot
#przywrocic wyjscie do trybu interaktywnego
set output
#odtworzenie terminala
set terminal pop
```

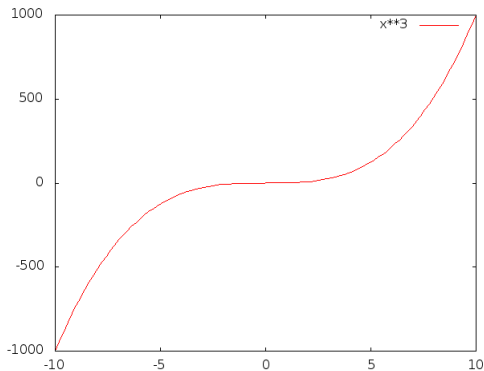
Przykład zastosowania

Przetestuj skrypt z poprzedniego slajdu:

```
gnuplot> plot x**3
```

```
gnuplot> call "skrypt.p" "szescian.png"
```

Uwaga ! Powyżej w call cudzysłowy są obowiązkowe.
Powinieneś otrzymać plik "szescian.png":



Wiele zestawów danych w jednym pliku

- **Problem:** Załóżmy, że pewien długo działający program wykonuje pewne kompleksowe obliczenia. Program ten okresowo podsumowuje wyniki osiągnięte do danego momentu i dołącza je do pliku wyjściowego, a następnie kontynuuje swoje obliczenia.
- Ważną kwestią jest to, że każdy wynik pośredni nie jest pojedynczą daną, ale całym zbiorem danych, obejmującym wiele wierszy (i, potencjalnie, wiele kolumn) w pliku z danymi.
- **Zadanie:** Chcemy wykorzystać gnuplota, aby zrozumieć zawartość pliku wynikowego.

Wiele zestawów danych w jednym pliku

```
# x: first column; corr(x): second col
```

```
# t=0
```

```
0 0.99
```

```
1 0.03
```

```
2 0.01
```

```
3 0.02
```

```
4 0.01
```

```
# t=1000
```

```
0 0.98
```

```
1 0.10
```

```
2 0.05
```

```
3 0.01
```

```
4 0.02
```

```
# t=2000
```

```
0 0.99
```

```
1 0.32
```

```
2 0.14
```

```
3 0.08
```

```
4 0.03
```

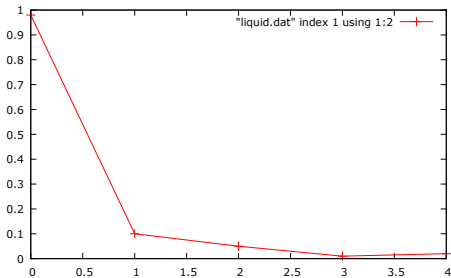
Wiele zestawów danych w jednym pliku

- Znaczenie **pustych linii** w pliku z danymi:
 - Dla gnuplota **pusta linia** w pliku z danymi oznacza brak ciągłości danych.
 - Dane powyżej i poniżej pustej linii będą traktowane jako należące do tego samego zestawu danych (i na wykresie będą przedstawione za pomocą tego samego stylu i koloru linii), ale nie będzie linii łączącej dane pochodzące od rekordu przed i po pustej linii.
- **Podwójne puste linie** są używane przez gnuplota do odróżnienia zestawów danych w pliku.
 - Każdy zestaw może być skierowany do komendy plot, jak gdyby był w osobnym pliku.
 - Wystarczy posłużyć się dyrektywą **index** polecenia **plot**.
 - Dyrektywa **index** następuje bezpośrednio po nazwie pliku z danymi i pobiera co najmniej jeden numeryczny argument określający zestaw danych z pliku, który należy wybrać.

Wiele zestawów danych w jednym pliku

- Aby wykreślić tylko funkcję korelacji dla $t = 1000$, można użyć polecenia:

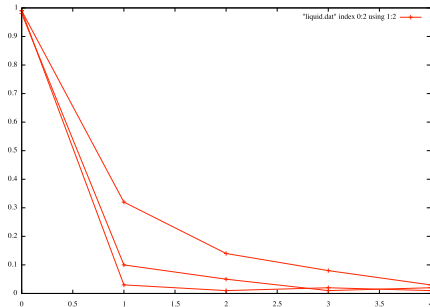
```
gnuplot> plot "liquid.dat" index 1 using 1:2  
w linespoints
```



Wiele zestawów danych w jednym pliku

- Aby wykreślić funkcję korelacji dla wszystkich t, można użyć polecenia:

```
gnuplot> plot "liquid.dat" index 0:2 using 1:2  
w linespoints
```



Wiele zestawów danych w jednym pliku

- Dyrektywa `index` może przyjąć do trzech argumentów, oddzielonych dwukropkiem:

```
index {int:start}[:{int:end}][:{int:step}]
```

- Jeśli zostanie podany tylko jeden argument, to wykreślony zostanie odpowiadający mu zestaw danych.
- Jeśli zostaną podane dwa argumenty, to są one traktowane jako wskaźnik na pierwszy i ostatni zestaw danych (włącznie).
- Jeśli zostaną podane trzy argumenty, to trzeci argument interpretowany jest jako tzw. krok.
- Tylko pierwszy argument jest obowiązkowy.

Wiele zestawów danych w jednym pliku

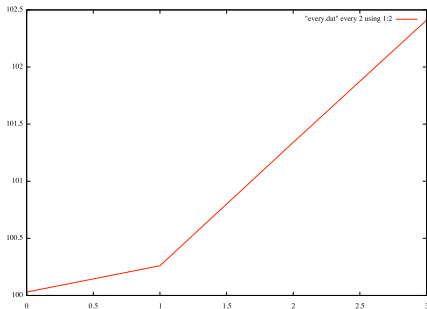
- Wyobraź sobie, że plik danych zawiera (w regularny sposób) mieszankę różnych rekordów tego samego zbioru danych, na przykład naprzemienne odczyty temperatury i ciśnienia:

```
# time - value
0    100.03    # temperatura
0     2.10     # cisnienie
1    100.26    # t
1     2.02     # c
2    101.34    # t
2     1.95     # c
3    102.41    # t
3     1.87     # c
```

Wiele zestawów danych w jednym pliku

- Jeśli chcemy wykreślić tylko temperaturę względem czasu, to możemy użyć dyrektywy **every**, aby pobrać tylko odpowiedni podzbiór wszystkich linii z danymi:

```
gnuplot> plot "every.dat" every 2 u 1:2 with lines
```



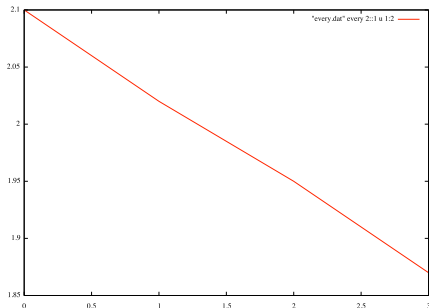
Wiele zestawów danych w jednym pliku

- Za pomocą dyrektywy **every** możemy kontrolować, w jaki sposób krok po kroku odczytujemy poszczególne linie w pliku z danymi. Składnia:

```
every {int:step} [::{int:start}[::{int:end}]]
```

Przykładowo:

```
plot "every.dat" every 2::1 u 1:2 with lines
```



Wiele zestawów danych w jednym pliku

- Pierwszym argumentem dyrektywy **every** jest przyrost, a następnie (opcjonalnie) podawany jest numer pierwszego i ostatniego wiersza.
- Numery linii są liczone od zera.
- Brak podwójnych dwukropków nie wygeneruje komunikatu o błędzie, ale doprowadzi do dziwnego i trudnego do przewidzenia zachowania.