

INTEGRACJA PYTHONA Z R

Piotr Byzia

W Z U R



PLAN

- PO CO INTEGROWAĆ R Z INNYM ŚRODOWISKIEM.
- DLACZEGO PYTHON.
- JAK ZINTEGROWAĆ.
- WADY, ZALETY I OGRANICZENIA.
- PRZYKŁADY UŻYCIA.

PO CO INTEGROWAĆ?

- Większa kontrola nad **importem** danych (DB, XML, YAML, HDF(Hierarchical Data Format), ORM)
- **Obróbka** danych (m.in. regexp)
- **API** (w najgorszym wypadku wrappery w Pythonie)
- **Wizualizacja** (Matplotlib)
- **NumPy, SciPy**
- **Web frameworks** (Django, Pylons, Turbo Gears, Zope)
- Już **istniejący kod** w Pythonie
- **GUI** (Qt, GTK, Wx, Cocoa)
- **Jython, IronPython, C API, ...**

Finance Chart Demonstration



MATPLOTLIB

http://www.advsofteng.com/gallery_finance.html

DLACZEGO PYTHON?

- Z perspektywy naukowca
- Z perspektywy obecnego użytkownika R
(Matplotlib, Gnuplot, Processing, Prefuse, ManyEyes)

- Ruby: RsRuby <http://sciruby.codeforpeople.com/sr.cgi/RubyWithRlang>

- Perl: RSPerl <http://www.omegahat.org/RSPerl/>



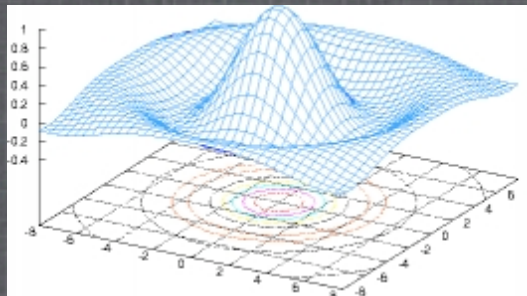
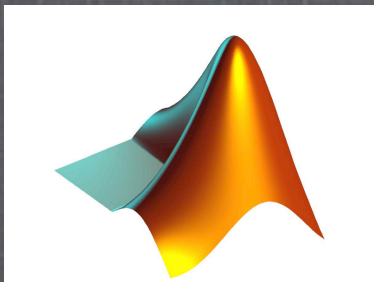
A W PRZECIWNYM KIERUNKU...

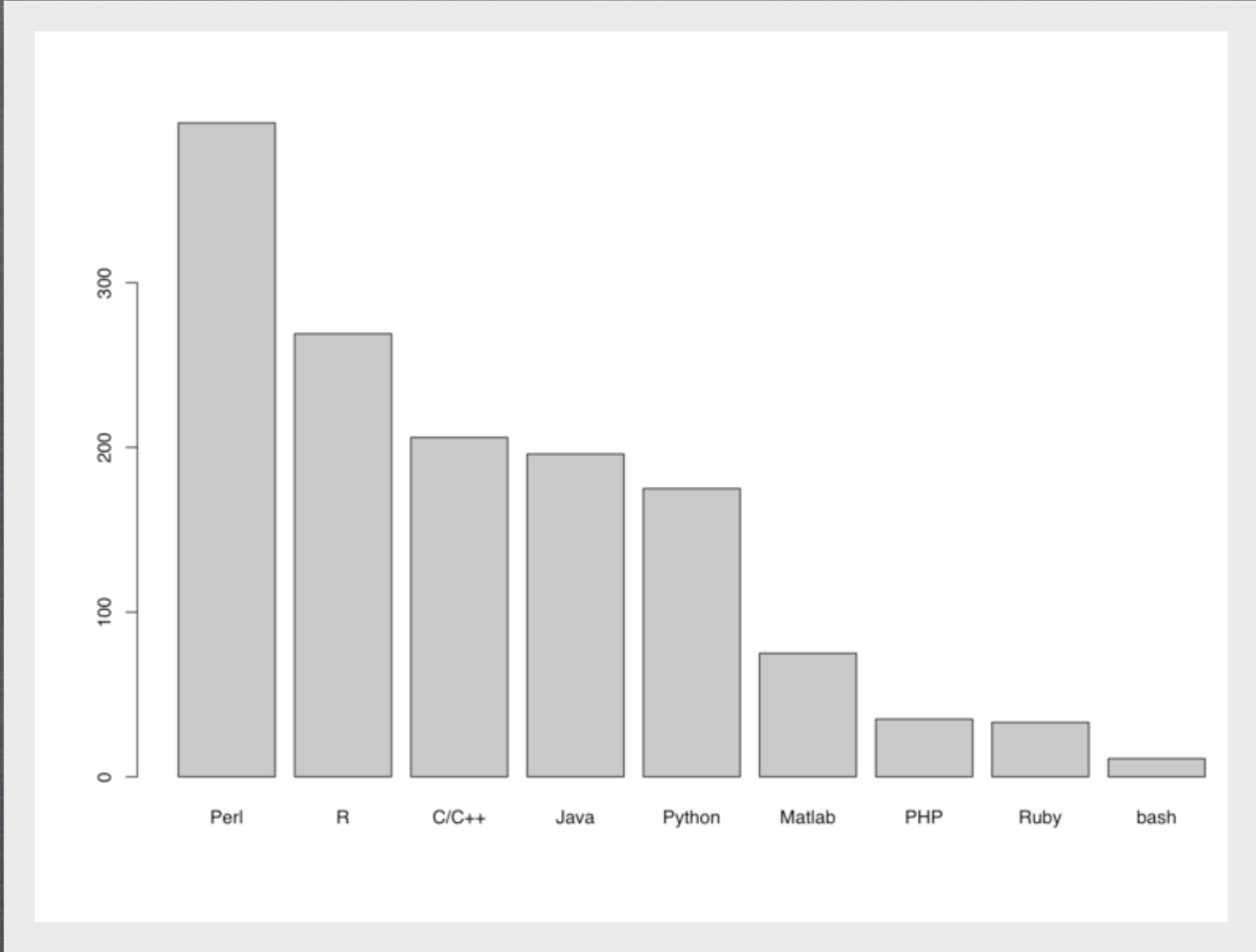
- Programiści Pythona mają SciPy.Stats



- R oferuje znacznie więcej

- Matlab, Octave, SAS, Statistica, SPSS





SUMMARY BARPLOT OF MOST POPULAR PROGRAMMING LANGUAGES

http://openwetware.org/wiki/Image:Most_Popular_Bioinformatics_Programming_Languages.png



[HTTP://OPENWETWARE.ORG/WIKI/
BIOGANG:PROJECTS/
BIOINFORMATICS_CAREER_SURVEY_2008](http://openwetware.org/wiki/BIOGANG:PROJECTS/BIOINFORMATICS_CAREER_SURVEY_2008)

*git clone git://github.com/michaelbarton/
bioinformatics-career-survey.git*

JAK ZINTEGROWAĆ?



1. RPy2 (<http://rpy.sourceforge.net/rpy2>)
2. RSPython (<http://www.omegahat.org/RSPython/>)



- RPy to pythonowy interfejs dla R
- RPy vs. RPy2 (CONVERSION vs. ten sam obiekt w pamięci dla R i dla Pythona)
- RPy zastąpione przez RPy2 (alpha 3)
- R działa jako embedded process w procesie Pythona (rpy2 to pythonowy front-end do biblioteki C zarządzanej przez R)
- Laurent Gautier (Bioconductor, Pfizer)



- Python 2.4 or higher, R-2.7.0 or higher
- Microsoft's Windows precompiled binaries
- Install from source
- <http://rpy.sourceforge.net/>



- **rpy2.rpy_classic**
Higher-level interface similar to the one in RPy-1.x. This is provided for compatibility reasons, as well as to facilitate the migration to RPy2.
- **rpy2.objects**
Higher-level interface, when ease-of-use matters most.
- **rpy2.rinterface**
Low-level interface to R, when speed and flexibility matter most. Here the programmer gets close(r) to R's C-level API.

PODSTAWY + RÓŻNICE

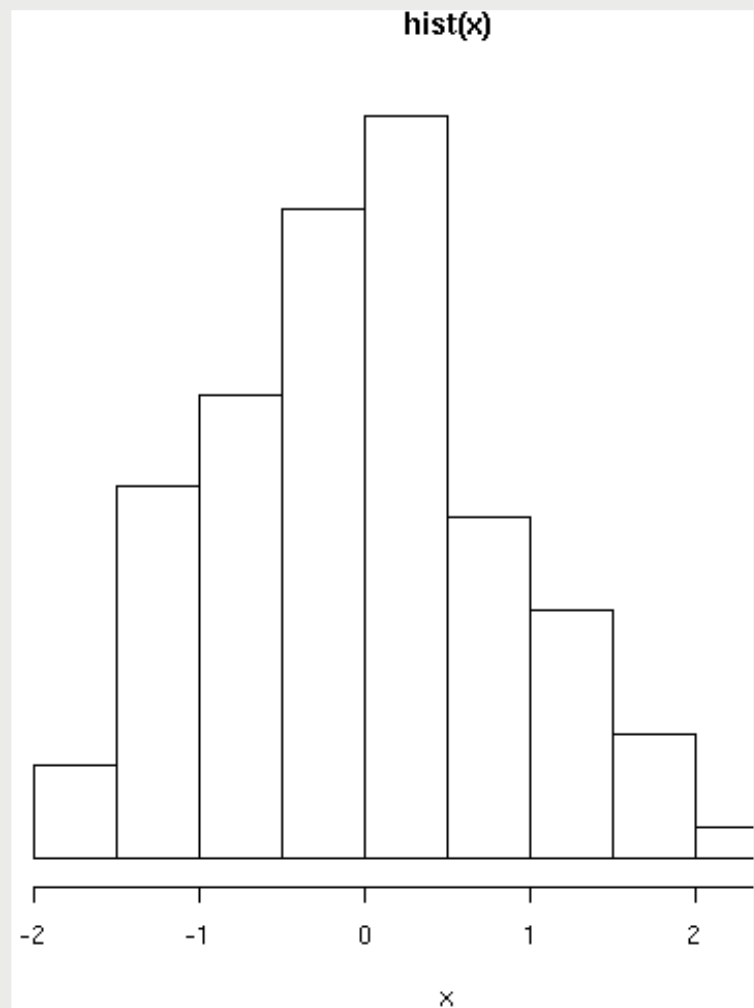
```
untitled
>>> import rpy2.robjects as robjects

>>> pi = robjects.r.pi
>>> pi
3.14159265358979

>>> robjects.r('1+2')
3

x = robjects.r.rnorm(100)
robjects.r('hist(%s, xlab="x", main="hist(x)")' % repr(x))

>>> y = robjects.r.seq(1, 10)
>>> y[0]
1
>>> y.r[0]
integer(0)
>>> y.r[-1]
2:10
>>> y.r + 1
2:11 # a w Pythonie operator '+' oznacza konkatencje
```



PRZYKŁAD

```
import rpy2.robjects as robjects
import array
```

```
r = robjects.r
```

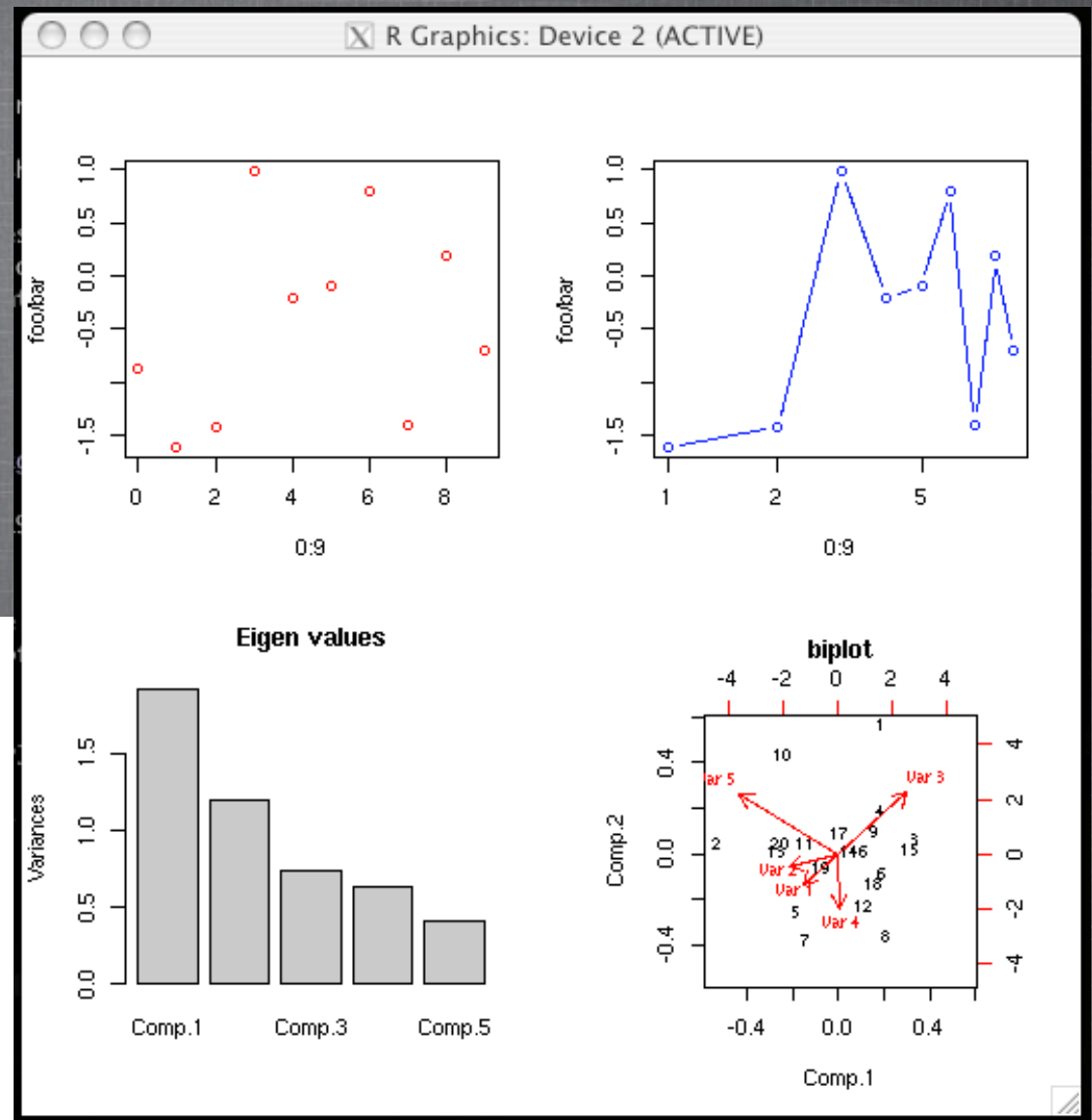
```
x = array.array('i', range(10))
y = r.rnorm(10)
```

```
r.X11()
```

```
r.par(mfrow=array.array('i', [2,2]))
r.plot(x, y, ylab="foo/bar", col="red")
```

```
kwargs = {'ylab':"foo/bar", 'type':"b", 'col':"blue", 'log':"x"}
r.plot(x, y, **kwargs)
```

```
m = r.matrix(r.rnorm(100), ncol=5)
pca = r.princomp(m)
r.plot(pca, main="Eigen values")
r.biplot(pca, main="biplot")
```



R VS. RPy2

The R code is

```
m <- matrix(rnorm(100), ncol=5)
pca <- princomp(m)
plot(pca, main="Eigen values")
biplot(pca, main="biplot")
```

The rpy2.objects code is

```
import rpy2.objects as robjects
r = robjects.r

m = r.matrix(r.rnorm(100), ncol=5)
pca = r.princomp(m)
r.plot(pca, main="Eigen values")
r.biplot(pca, main="biplot")
```

Similar due to inheritance



Similar due to... uh... other factors



R OBJECTS

- RVector
- RArray (dziedziczy po robjcts.RVector dodając atrybut dim)
- RMatrix (specjalny przypadek RArray)
- RDataFrame (eksperymentalny, SexpVector albo słownik, którego elementy będą kolumnami data.frame)
- Environments, Functions, Formulae, rpy2.rinterface

WADY, ZALETY I OGRANICZENIA

- Ten sam obiekt dla R i Pythona w przestrzeni nazw R
- Automatyczna konwersja: `ri2py()`, `py2ri()`
- Domyślna paczka Debiana
- RPy łatwiejszy od RSPythona
- RSPython to interfejs w obie strony
- RSPython 0.7.1 z 2005 roku

Pytania?

