

SAGE: Wprowadzenie do pakietu i integracja z R

Andrzej Giniewicz

`a.giniewicz@gmail.com`

Pierwszy Wrocławski Zlot Użytkowników R
6 września 2008

O czym chcę opowiedzieć...

SAGE ogólnie

Co to takiego SAGE?

Historia pakietu

Założenia

Metody rozwoju pakietu

Wygląd

Możliwości pakietu

Statystyka w SAGE

Ukryty SciPy, przyczajony R

Sytuacja na chwilę obecną

Wypad w przyszłość

Demonstracja możliwości

Co to takiego SAGE?

SAGE jest **darmowym** pakietem matematycznym rozpowszechnianym na licencji GPL.

Co to takiego SAGE?

SAGE jest **darmowym** pakietem matematycznym rozpowszechnianym na licencji GPL.

Celem jakim przyświeca jego twórcom jest stworzenie rozsądnej i darmowej alternatywy dla pakietów takich jak Magma, Maple, Mathematica i Matlab, czyli „Sławnych M”.

Co to takiego SAGE?

SAGE jest **darmowym** pakietem matematycznym rozpowszechnianym na licencji GPL.

Celem jakim przyświeca jego twórcom jest stworzenie rozsądnej i darmowej alternatywy dla pakietów takich jak Magma, Maple, Mathematica i Matlab, czyli „Sławnych M”.

SAGE cechuje się bardzo charakterystyczną budową, jest on dystrybucją wielu darmowych pakietów z których każdy robi dobrze to do czego został stworzony, pozwala na wymianę wyników pomiędzy nimi i dodaje nowy, wspólny interfejs.

Co to takiego SAGE?

SAGE jest **darmowym** pakietem matematycznym rozpowszechnianym na licencji GPL.

Celem jakim przyświeca jego twórcom jest stworzenie rozsądnej i darmowej alternatywy dla pakietów takich jak Magma, Maple, Mathematica i Matlab, czyli „Sławnych M”.

SAGE cechuje się bardzo charakterystyczną budową, jest on dystrybucją wielu darmowych pakietów z których każdy robi dobrze to do czego został stworzony, pozwala na wymianę wyników pomiędzy nimi i dodaje nowy, wspólny interfejs.

Wszelkie informacje na temat pakietu i demo znaleźć można na jego stronie domowej, <http://sagemath.org/>. Notebook zawierający przykłady z tej prezentacji opublikowany jest pod adresem: <https://sage.math.washington.edu:10009/home/pub/7/>

Historia pakietu

- ▶ Rok 2004, dr William Stein po niemal 10 latach używania pakietu Magma decyduje iż nie podoba mu się to rozwiązanie ze względu na wysoką cenę i brak możliwości weryfikacji zastosowanych rozwiązań, zaczyna analizować możliwości...

Historia pakietu

- ▶ Rok 2004, dr William Stein po niemal 10 latach używania pakietu Magma decyduje iż nie podoba mu się to rozwiązanie ze względu na wysoką cenę i brak możliwości weryfikacji zastosowanych rozwiązań, zaczyna analizować możliwości...
- ▶ Styczeń 2005, pierwsza wersja pakietu SAGE 0.1, stworzona przez tylko jedną osobę

Historia pakietu

- ▶ Rok 2004, dr William Stein po niemal 10 latach używania pakietu Magma decyduje iż nie podoba mu się to rozwiązanie ze względu na wysoką cenę i brak możliwości weryfikacji zastosowanych rozwiązań, zaczyna analizować możliwości...
- ▶ Styczeń 2005, pierwsza wersja pakietu SAGE 0.1, stworzona przez tylko jedną osobę
- ▶ Luty 2006, wersja SAGE 1.0, pierwsze spotkanie SAGE Days, pojawia się interfejs graficzny będący jednocześnie interfejsem www, liczba twórców do tego momentu zwiększyła się do trzech

Historia pakietu

- ▶ Rok 2004, dr William Stein po niemal 10 latach używania pakietu Magma decyduje iż nie podoba mu się to rozwiązanie ze względu na wysoką cenę i brak możliwości weryfikacji zastosowanych rozwiązań, zaczyna analizować możliwości...
- ▶ Styczeń 2005, pierwsza wersja pakietu SAGE 0.1, stworzona przez tylko jedną osobę
- ▶ Luty 2006, wersja SAGE 1.0, pierwsze spotkanie SAGE Days, pojawia się interfejs graficzny będący jednocześnie interfejsem www, liczba twórców do tego momentu zwiększyła się do trzech
- ▶ Październik tego samego roku, kolejne spotkanie SAGE Days, społeczność zaczyna się interesować pakietem SAGE, co doprowadziło do...

Historia pakietu (c.d.)

- ▶ Obecnie pakiet SAGE jest w wersji 3.1.1, zawiera ponad 60 darmowych pakietów, w jego rozwoju pomagają ponad 110 osób z całego świata, prace nad nim fundowane są ze stypendiów i grantów Uniwersytetu w Waszyngtonie oraz z pomocy sponsorów, w tym znaczących jak Google lub Microsoft.

Historia pakietu (c.d.)

- ▶ Obecnie pakiet SAGE jest w wersji 3.1.1, zawiera ponad 60 darmowych pakietów, w jego rozwoju pomagają ponad 110 osób z całego świata, prace nad nim fundowane są ze stypendiów i grantów Uniwersytetu w Waszyngtonie oraz z pomocy sponsorów, w tym znaczących jak Google lub Microsoft.

Pakiet jest wciąż rozwijany w bardzo dużym tempie, pomimo iż nie jest pozbawiony błędów, to wszystkie które się pojawiają są bardzo szybko usuwane, nowe stabilne wersje pakietu ukazują się średnio dwa lub trzy razy w miesiącu i zwykle zawierają nie mniej niż sto zmian lub poprawek

Historia pakietu (c.d.)

- ▶ Obecnie pakiet SAGE jest w wersji 3.1.1, zawiera ponad 60 darmowych pakietów, w jego rozwoju pomagają ponad 110 osób z całego świata, prace nad nim fundowane są ze stypendiów i grantów Uniwersytetu w Waszyngtonie oraz z pomocy sponsorów, w tym znaczących jak Google lub Microsoft.

Pakiet jest wciąż rozwijany w bardzo dużym tempie, pomimo iż nie jest pozbawiony błędów, to wszystkie które się pojawiają są bardzo szybko usuwane, nowe stabilne wersje pakietu ukazują się średnio dwa lub trzy razy w miesiącu i zwykle zawierają nie mniej niż sto zmian lub poprawek

I wszystko wskazuje na to, że na tym się nie skończy. . .

Założenia

Pakietowi SAGE przyświeca kilka założeń, najważniejszym jest to by wykorzystać to co zostało zrobione dobrze, oficjalnym mottem projektu to *Building the car* co doskonale to opisuje.

Założenia

Pakietowi SAGE przyświeca kilka założeń, najważniejszym jest to by wykorzystać to co zostało zrobione dobrze, oficjalnym mottem projektu to *Building the car* co doskonale to opisuje.

Jeśli zatem gdzieś pojawia się wyspecjalizowany pakiet implementujący metody algebry liniowej nad ciałem skończonym $\mathbb{Z}/2\mathbb{Z}$ (pakiet „M4RI”) lub wielomianów nad pierścieniami Boole’a (pakiet „PolyBoRI”) i jeśli to co robi robi dobrze i szybko, to (jak dwa powyższe pakiety) jest włączany do pakietu SAGE. Zwykle implementacja dla jednej metody pojawia się w więcej niż jednym pakiecie, wtedy wybierana jest ta która działa najszybciej.

Założenia — składnia

Ważną i trafioną decyzją twórcy pakietu była rezygnacja z próby wymyślenia własnego języka jak to miało miejsce w przypadku wielu innych pakietów. Językiem którym posługujemy się w pakiecie SAGE jest Python, a w przypadku korzystania z konsoli lub interfejsu graficznego lekko zmodyfikowany Python w którym operator XOR został zmieniony po to by użyć go jako symbolu potęgowania zamiast dwóch gwiazdek.

Założenia — składnia

Ważną i trafioną decyzją twórcy pakietu była rezygnacja z próby wymyślenia własnego języka jak to miało miejsce w przypadku wielu innych pakietów. Językiem którym posługujemy się w pakiecie SAGE jest Python, a w przypadku korzystania z konsoli lub interfejsu graficznego lekko zmodyfikowany Python w którym operator XOR został zmieniony po to by użyć go jako symbolu potęgowania zamiast dwóch gwiazdek.

Oprócz nowej składni dla każdego z pakietów, SAGE pozwala skorzystać z ich oryginalnej składni z poziomu konsoli lub interfejsu, wykorzystanie wyników jest jednak wtedy trudniejsze choć nie niemożliwe (wyniki takie są w postaci napisów).

Założenia — składnia

Ważną i trafioną decyzją twórcy pakietu była rezygnacja z próby wymyślenia własnego języka jak to miało miejsce w przypadku wielu innych pakietów. Językiem którym posługujemy się w pakiecie SAGE jest Python, a w przypadku korzystania z konsoli lub interfejsu graficznego lekko zmodyfikowany Python w którym operator XOR został zmieniony po to by użyć go jako symbolu potęgowania zamiast dwóch gwiazdek.

Oprócz nowej składni dla każdego z pakietów, SAGE pozwala skorzystać z ich oryginalnej składni z poziomu konsoli lub interfejsu, wykorzystanie wyników jest jednak wtedy trudniejsze choć nie niemożliwe (wyniki takie są w postaci napisów).

Ciekawym dodatkiem jest też to, że SAGE integruje się również z płatnymi pakietami, mamy zatem możliwość przykładowo używania Matlab'a lub Mathematici z poziomu SAGE'a zupełnie jak w przypadku zawartych w nim darmowych pakietów.

Metody rozwoju pakietu

W projekcie tym tak jak i w każdym projekcie FLOSS ważna jest organizacja. Bezpośredni dostęp w trybie do zapisu do kodu który trafia do wydania ma jedynie kilka osób, bezpośrednich współpracowników twórcy pakietu, jednak każdy ma dostęp do wersji tylko do odczytu i może pomóc w rozwoju poprzez wysłanie patchy, początkowo na listę mailingową, po akceptacji choć jednego patcha można otrzymać dostęp do pozostałych narzędzi.

Metody rozwoju pakietu

W projekcie tym tak jak i w każdym projekcie FLOSS ważna jest organizacja. Bezpośredni dostęp w trybie do zapisu do kodu który trafia do wydania ma jedynie kilka osób, bezpośrednich współpracowników twórcy pakietu, jednak każdy ma dostęp do wersji tylko do odczytu i może pomóc w rozwoju poprzez wysłanie patchy, początkowo na listę mailingową, po akceptacji choć jednego patcha można otrzymać dostęp do pozostałych narzędzi. Zależnie od skomplikowania zadania patch który zostaje zamieszczony w raporcie oczekuje na jedną lub więcej recenzji innych użytkowników, jeśli zostanie on pozytywnie zaopiniowany trafi do jednego z następnych wydań. Ważnym elementem każdego patcha jest towarzysząca dokumentacja z przykładami które demonstrują użycie nowych narzędzi lub dowodzą naprawienia błędu i sprawdzane są przed każdym wydaniem.

Wygląd — konsola

Pakiet SAGE wyposażony jest w konsolę, wygląda ona tak jak większość narzędzi tego typu, jest jednak w rzeczywistości zintegrowaną konsolą dla wszystkich pakietów z SAGE które taką posiadają, spójrzmy na tę przykładową sesję.

```
[giniu@ginland ~]$ sage
```

```
-----  
| SAGE Version 3.1.2.alpha4, Release Date: 2008-09-02           |  
| Type notebook() for the GUI, and license() for information. |  
-----
```

```
sage: 2+2
```

```
4
```

```
sage: %r
```

```
--> Switching to R Interpreter <--  
,,
```

```
r: rnorm(3)
```

```
[1] 0.1464635 -2.3516542 -1.0098941
```

```
r: exit
```

```
--> Exiting back to SAGE <--
```

```
sage: exit
```

```
Exiting SAGE (CPU time 0m0.07s, Wall time 1m41.26s).
```

```
Exiting spawned R Interpreter process.
```

Wygląd — konsola (c.d.)

Jak widać udało nam się dostać do konsoli R, jest ona jednak nieco zmodyfikowana, przykładowo `plot` nie otwiera domyślnie nowego urządzenia graficznego. Dzieje się tak ze względu na sposób w jaki rozpowszechniany jest SAGE na różnych systemach operacyjnych. Jeśli chcemy możemy otworzyć nowe okno graficzne przykładowo używając `X11()` jeśli w danym momencie używamy Linux'a. Jeśli ktoś potrzebuje konsoli nie zmodyfikowanej, jest ona również dostępna, spójrzmy na znany wszystkim obrazek:

```
[giniu@ginland ~]$ sage --R
R version 2.6.1 (2007-11-26)
Copyright (C) 2007 The R Foundation for Statistical Computing
ISBN 3-900051-07-0
```

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

```
> rnorm(3)
[1] -0.88041283 -0.07919025  0.49146663
> q()
```


Wygląd — notebook

sage Notebook

admin | [Toggle](#) | [Home](#) | [Published](#) | [Log](#) | [Settings](#) | [Report a Problem](#) | [Help](#) | [Sign out](#)

Untitled

last edited on August 21, 2008 04:50 AM by admin

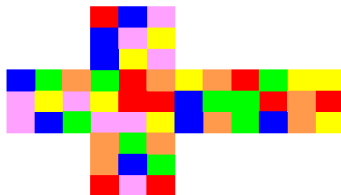
Save Save & quit Discard & quit

File... Action... Data... sage Typeset [Print](#) **Worksheet** Edit Text Undo Share Publish

```
C=RubiksCube().scramble()
ruchy=C.solve()
D=RubiksCube().move(ruchy)
print(ruchy)
print(D==C)
show(D)
D.show3d()
```

[evaluate](#)

```
L2 B L2 B' L2 B2 L2 F2 D2 L2 D R2 F' R' B' R F2 D' B' D'
True
```



sagemath.org

jsMath

Wygląd — podpowiedzi

sage Notebook admin | [Toggle](#) | [Home](#) | [Published](#) | [Log](#) | [Settings](#) | [Report a Problem](#) | [Help](#) | [Sign out](#)

Untitled
last edited on September 04, 2008 01:40 AM by admin

File... Action... Data... sage Typeset [Print](#) **Worksheet** Edit Text Undo Share Publish

evaluate

p|

pl_normalize	partitions_restricted	pow
pl_normalize_int	partitions_set	power
pl_normalize_llong	partitions_tuples	power_mod
plist	permutation_action	powerset
plist_int	permutations	prange
plist_llong	permutations_iterator	preparse
pAdicField	pg	preparser
pAdicRing	pi	pretty_print
pad_zeros	pickle_function	pretty_print_default
padic_printing	piecewise	previous_prime
pager	plot	previous_prime_power
parallel	plot3d	prime_divisors
parametric_plot	plot_vector_field	prime_factors
parametric_plot3d	plotkin_bound_asymp	prime_pi
parent	plotkin_upper_bound	prime_powers
paretovariate	pmem_malloc	prime_range
pari	point	prime_to_m_part
pari_gen	point2d	primes
parse_cremona_label	point3d	primes_first_n
partition	points	primitive_root
partition_associated	polar_plot	prod

jsMath

Wygląd — dokumentacja

```
primes_first_n?
```

[evaluate](#)

```
File:      /opt/sage/local/lib/python2.5/site-packages/sage/rings/arith.py
Type:     <type 'function'>
Definition: primes_first_n(n, leave_pari)
Docstring:
```

Return the first n primes.

INPUT:

```
leave_pari -- bool (default: False) if True the returned list
is a PARI list; this is *vastly* (10 times!)
faster since the time of prime_range is dominated
by conversion from PARI to SAGE integers.
However, PARI integers are much different than
SAGE integers.  If you use this option the lower
bound must be 2.
```

OUTPUT:

a list of the first n prime numbers.

EXAMPLES:

```
sage: primes_first_n(10)
[2, 3, 5, 7, 11, 13, 17, 19, 23, 29]
sage: len(primes_first_n(1000))
1000
```

This is very fast, because we leave the output as a PARI object:

```
sage: v = primes_first_n(10^6, leave_pari=True)
sage: len(v)
1000000
```

jsMath

Wygląd — kod funkcji

primes_first_n??

```
File: /opt/sage/local/lib/python2.5/site-packages/sage/rings/arith.py
Source Code (starting at line 579):
def primes_first_n(n, leave_pari=False):
    r"""
    Return the first $n$ primes.

    INPUT:
      leave_pari -- bool (default: False) if True the returned list
                  is a PARI list; this is *vastly* (10 times!)
                  faster since the time of prime_range is dominated
                  by conversion from PARI to SAGE integers.
                  However, PARI integers are much different than
                  SAGE integers. If you use this option the lower
                  bound must be 2.

    OUTPUT:
      a list of the first $n$ prime numbers.

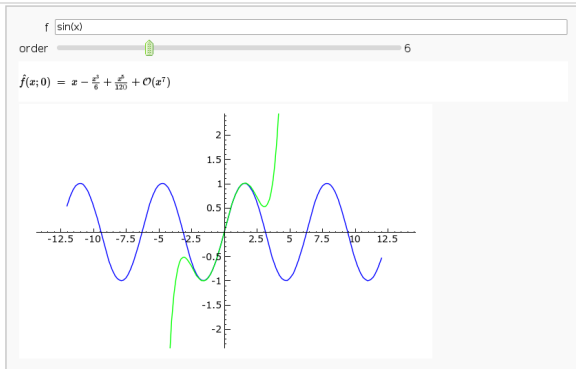
    EXAMPLES:
      sage: primes_first_n(10)
      [2, 3, 5, 7, 11, 13, 17, 19, 23, 29]
      sage: len(primes_first_n(1000))
      1000

    This is very fast, because we leave the output as a PARI object:
      sage: v = primes_first_n(10^6, leave_pari=True)
      sage: len(v)
      1000000
    """
    v = pari.prime_list(n)
    Z = integer_ring.ZZ
    if leave_pari:
        return v
    return [Z(p) for p in v] # this dominates runtime!
```

jmMath

Wygląd — „Interact”

```
var('x')
@interact
def _(f=input_box(default=sin(x)), order=(1..20)):
    maclaurin = f.taylor(x,0,order)
    funkcja = plot(f,-12,12)
    aproksymacja = plot(maclaurin,-12, 12, color='green')
    html( '$\hat{f}(x;0)\;=\;s+\mathcal{O}(x^{%s})$' % (latex(maclaurin),order+1) )
    show(funkcja + aproksymacja, ymin = -2, ymax = 2)
```



Wygląd — kod i dokumentacja R

```
r.replace??
```

A Function in the R interface (R source code):

```
function (x, list, values)
{
  x[list] <- values
  x
}
<environment: namespace:base>
```

```
r.replace?
```

```
Type:          <class 'sage.interfaces.r.RFunction'>
Definition:   r.replace( [noargspec] )
Docstring:
replace              package:base              R Documentation
```

Replace Values in a Vector

Description:

'replace' replaces the values in 'x' with indexes given in 'list' by those given in 'values'. If necessary, the values in 'values' are recycled.

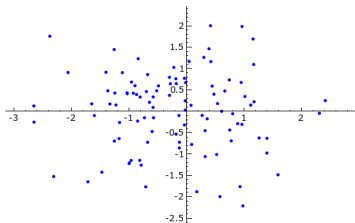
Usage:

```
replace(x, list, values)
```

jMath

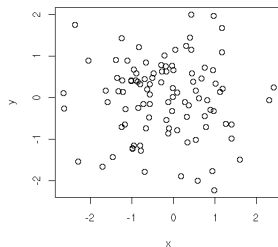
Wygląd — grafika R

```
x = sageobj(r.rnorm(100))
y = r.rnorm(100).sage_()
plot(points(zip(x,y)))
```



```
r.png()
r.plot(x,y,xlab='x',ylab='y')
r.dev_off()
```

```
null device
1
```



Możliwości pakietu

- ▶ Elastyczny interfejs użytkownika

Możliwości pakietu

- ▶ Elastyczny interfejs użytkownika
- ▶ Możliwość włączenia bezpośrednio kodu w Pythonie, Cythonie, Fortranie, testowo w Lispie a niedługo w Rubym. . .

Możliwości pakietu

- ▶ Elastyczny interfejs użytkownika
- ▶ Możliwość włączenia bezpośrednio kodu w Pythonie, Cythonie, Fortranie, testowo w Lispie a niedługo w Rubym. . .
- ▶ Generator rysunków dwu i trój wymiarowych pracujący na zasadzie łączenia obiektów graficznych

Możliwości pakietu

- ▶ Elastyczny interfejs użytkownika
- ▶ Możliwość włączenia bezpośrednio kodu w Pythonie, Cythonie, Fortranie, testowo w Lispie a niedługo w Rubym. . .
- ▶ Generator rysunków dwu i trój wymiarowych pracujący na zasadzie łączenia obiektów graficznych
- ▶ Integracja z \LaTeX 'em poprzez pakiet SAGETEX .

Możliwości pakietu

- ▶ Elastyczny interfejs użytkownika
- ▶ Możliwość włączenia bezpośrednio kodu w Pythonie, Cythonie, Fortranie, testowo w Lispie a niedługo w Rubym. . .
- ▶ Generator rysunków dwu i trój wymiarowych pracujący na zasadzie łączenia obiektów graficznych
- ▶ Integracja z \LaTeX 'em poprzez pakiet SAGETEX .
- ▶ Dostępne metody dla bardzo wielu dziedzin:
 - ▶ Analiza, algebra liniowa

Możliwości pakietu

- ▶ Elastyczny interfejs użytkownika
- ▶ Możliwość włączenia bezpośrednio kodu w Pythonie, Cythonie, Fortranie, testowo w Lispie a niedługo w Rubym. . .
- ▶ Generator rysunków dwu i trój wymiarowych pracujący na zasadzie łączenia obiektów graficznych
- ▶ Integracja z \LaTeX 'em poprzez pakiet \SAGETEX .
- ▶ Dostępne metody dla bardzo wielu dziedzin:
 - ▶ Analiza, algebra liniowa
 - ▶ Teoria grafów, teoria liczb, teoria kategorii, teoria grup, pierścienie, liczby p-adyczne. . .

Możliwości pakietu

- ▶ Elastyczny interfejs użytkownika
- ▶ Możliwość włączenia bezpośrednio kodu w Pythonie, Cythonie, Fortranie, testowo w Lispie a niedługo w Rubym. . .
- ▶ Generator rysunków dwu i trój wymiarowych pracujący na zasadzie łączenia obiektów graficznych
- ▶ Integracja z \LaTeX 'em poprzez pakiet \SAGETEX .
- ▶ Dostępne metody dla bardzo wielu dziedzin:
 - ▶ Analiza, algebra liniowa
 - ▶ Teoria grafów, teoria liczb, teoria kategorii, teoria grup, pierścienie, liczby p-adyczne. . .
 - ▶ Kryptografia, kombinatoryka, prawdopodobieństwo. . .

Możliwości pakietu

- ▶ Elastyczny interfejs użytkownika
- ▶ Możliwość włączenia bezpośrednio kodu w Pythonie, Cythonie, Fortranie, testowo w Lispie a niedługo w Rubym. . .
- ▶ Generator rysunków dwu i trój wymiarowych pracujący na zasadzie łączenia obiektów graficznych
- ▶ Integracja z \LaTeX 'em poprzez pakiet \SAGETEX .
- ▶ Dostępne metody dla bardzo wielu dziedzin:
 - ▶ Analiza, algebra liniowa
 - ▶ Teoria grafów, teoria liczb, teoria kategorii, teoria grup, pierścienie, liczby p-adyczne. . .
 - ▶ Kryptografia, kombinatoryka, prawdopodobieństwo. . .
 - ▶ Pierścienie wielomianów, pierścienie szeregów potęgowych, algebry, algebry kwaternionów, ciała skończone. . .

Możliwości pakietu

- ▶ Elastyczny interfejs użytkownika
- ▶ Możliwość włączenia bezpośrednio kodu w Pythonie, Cythonie, Fortranie, testowo w Lispie a niedługo w Rubym. . .
- ▶ Generator rysunków dwu i trój wymiarowych pracujący na zasadzie łączenia obiektów graficznych
- ▶ Integracja z \LaTeX 'em poprzez pakiet \SAGETEX .
- ▶ Dostępne metody dla bardzo wielu dziedzin:
 - ▶ Analiza, algebra liniowa
 - ▶ Teoria grafów, teoria liczb, teoria kategorii, teoria grup, pierścienie, liczby p-adyczne. . .
 - ▶ Kryptografia, kombinatoryka, prawdopodobieństwo. . .
 - ▶ Pierścienie wielomianów, pierścienie szeregów potęgowych, algebry, algebry kwaternionów, ciała skończone. . .
 - ▶ L-funkcje, krzywe i płaszczyzny eliptyczne, krzywe hipereliptyczne. . .

Możliwości pakietu

- ▶ Elastyczny interfejs użytkownika
- ▶ Możliwość włączenia bezpośrednio kodu w Pythonie, Cythonie, Fortranie, testowo w Lispie a niedługo w Rubym. . .
- ▶ Generator rysunków dwu i trój wymiarowych pracujący na zasadzie łączenia obiektów graficznych
- ▶ Integracja z \LaTeX 'em poprzez pakiet \SAGETEX .
- ▶ Dostępne metody dla bardzo wielu dziedzin:
 - ▶ Analiza, algebra liniowa
 - ▶ Teoria grafów, teoria liczb, teoria kategorii, teoria grup, pierścienie, liczby p-adyczne. . .
 - ▶ Kryptografia, kombinatoryka, prawdopodobieństwo. . .
 - ▶ Pierścienie wielomianów, pierścienie szeregów potęgowych, algebry, algebry kwaternionów, ciała skończone. . .
 - ▶ L-funkcje, krzywe i płaszczyzny eliptyczne, krzywe hipereliptyczne. . .
 - ▶ Od wersji 3.0 dzięki integracji z R również statystyka!

Ukryty SciPy, ...

To co było napisane w ostatnim punkcie ostatniego slajdu jest po części prawdą, do momentu wydania wersji 3.0 nie było oficjalnego wsparcia dla statystyki i integracja z R to zmieniła, jednak już wcześniej dało się dostać do prostych metod statystycznych z pakietu SciPy. W przestrzeni nazw `scipy.stats.*` znajduje się około 90 rozkładów, proste funkcje do statystyki opisowej, podstawowe testy związane z parametrem położenia i skali oraz metody uogólnionego modelu liniowego i anovy — nie znalazły się jednak w pakiecie SAGE celowo lecz przy okazji (SciPy jest używany przez inne metody pakietu).

Ukryty SciPy, ...

To co było napisane w ostatnim punkcie ostatniego slajdu jest po części prawdą, do momentu wydania wersji 3.0 nie było oficjalnego wsparcia dla statystyki i integracja z R to zmieniła, jednak już wcześniej dało się dostać do prostych metod statystycznych z pakietu SciPy. W przestrzeni nazw `scipy.stats.*` znajduje się około 90 rozkładów, proste funkcje do statystyki opisowej, podstawowe testy związane z parametrem położenia i skali oraz metody uogólnionego modelu liniowego i anovy — nie znalazły się jednak w pakiecie SAGE celowo lecz przy okazji (SciPy jest używany przez inne metody pakietu).

Dopiero w momencie wydania wersji 3.0 statystyka weszła na dobre do pakietu SAGE, choć prace nad wsparciem metod statystycznych nie zakończyły się na samej integracji z R...

... przyczajony R

Integracja nowej dziedziny do pakietu SAGE zwykle przebiega następująco:

- ▶ Wybór pakietów

... przyczajony R

Integracja nowej dziedziny do pakietu SAGE zwykle przebiega następująco:

- ▶ Wybór pakietów
- ▶ Stworzenie interfejsu PExpect dla wszystkich pakietów

... przyczajony R

Integracja nowej dziedziny do pakietu SAGE zwykle przebiega następująco:

- ▶ Wybór pakietów
- ▶ Stworzenie interfejsu PExpect dla wszystkich pakietów
- ▶ Stworzenie połączenia pakietu z Pythonem

... przyczajony R

Integracja nowej dziedziny do pakietu SAGE zwykle przebiega następująco:

- ▶ Wybór pakietów
- ▶ Stworzenie interfejsu PExpect dla wszystkich pakietów
- ▶ Stworzenie połączenia pakietu z Pythonem
- ▶ Natywny interfejs do wszystkiego czego brakuje użytkownikom przesiadającym się z pakietów komercyjnych

... przyczajony R

Integracja nowej dziedziny do pakietu SAGE zwykle przebiega następująco:

- ▶ Wybór pakietów
- ▶ Stworzenie interfejsu PExpect dla wszystkich pakietów
- ▶ Stworzenie połączenia pakietu z Pythonem
- ▶ Natywny interfejs do wszystkiego czego brakuje użytkownikom posiadającym się z pakietów komercyjnych

W przypadku statystyki w SAGE pierwsze trzy punkty tej listy są już wykonane, do połączenia R z Pythonem wykorzystano lekko zmodyfikowaną wersję RPy, również przygotowano interfejs PExpect dla pakietu R.

... przyuczajony R

Integracja nowej dziedziny do pakietu SAGE zwykle przebiega następująco:

- ▶ Wybór pakietów
- ▶ Stworzenie interfejsu PExpect dla wszystkich pakietów
- ▶ Stworzenie połączenia pakietu z Pythonem
- ▶ Natywny interfejs do wszystkiego czego brakuje użytkownikom przesiadającym się z pakietów komercyjnych

W przypadku statystyki w SAGE pierwsze trzy punkty tej listy są już wykonane, do połączenia R z Pythonem wykorzystano lekko zmodyfikowaną wersję RPy, również przygotowano interfejs PExpect dla pakietu R.

Interfejs natywny nie jest jednak jeszcze gotowy, głównie ze względu na małą bazę użytkowników zainteresowanych obecnie statystyką w SAGE zatem na małą znajomość ich potrzeb.

Sytuacja na chwilę obecną

Jeśli chcemy korzystać z pakietu R z poziomu SAGE'a możemy robić to dość skutecznie, należy jednak pamiętać o kilku zmianach:

- ▶ Kropki w nazwach funkcji zamienione są na podłogi

Sytuacja na chwilę obecną

Jeśli chcemy korzystać z pakietu R z poziomu SAGE'a możemy robić to dość skutecznie, należy jednak pamiętać o kilku zmianach:

- ▶ Kropki w nazwach funkcji zamienione są na podłogi
- ▶ Rysowanie w notebooku jest możliwe tylko poprzez urządzenie PNG zatem kod rysujący należy otoczyć w `r.png()`,
`r.dev_off()`

Sytuacja na chwilę obecną

Jeśli chcemy korzystać z pakietu R z poziomu SAGE'a możemy robić to dość skutecznie, należy jednak pamiętać o kilku zmianach:

- ▶ Kropki w nazwach funkcji zamienione są na podłogi
- ▶ Rysowanie w notebooku jest możliwe tylko poprzez urządzenie PNG zatem kod rysujący należy otoczyć w `r.png()`,
`r.dev_off()`
- ▶ Dowolne komendy otwierające nowe okno działają tylko z poziomu konsoli

Sytuacja na chwilę obecną

Jeśli chcemy korzystać z pakietu R z poziomu SAGE'a możemy robić to dość skutecznie, należy jednak pamiętać o kilku zmianach:

- ▶ Kropki w nazwach funkcji zamienione są na podłogi
- ▶ Rysowanie w notebooku jest możliwe tylko poprzez urządzenie PNG zatem kod rysujący należy otoczyć w `r.png()`, `r.dev_off()`
- ▶ Dowolne komendy otwierające nowe okno działają tylko z poziomu konsoli
- ▶ Przekazywanie napisów z poziomu interfejsu Pythona do interpretera R wymaga użycia podwójnych cudzysłowii (podwójnego w pojedynczym — `'"...'`).

Sytuacja na chwilę obecną

Jeśli chcemy korzystać z pakietu R z poziomu SAGE'a możemy robić to dość skutecznie, należy jednak pamiętać o kilku zmianach:

- ▶ Kropki w nazwach funkcji zamienione są na podłogi
- ▶ Rysowanie w notebooku jest możliwe tylko poprzez urządzenie PNG zatem kod rysujący należy otoczyć w `r.png()`, `r.dev_off()`
- ▶ Dowolne komendy otwierające nowe okno działają tylko z poziomu konsoli
- ▶ Przekazywanie napisów z poziomu interfejsu Pythona do interpretera R wymaga użycia podwójnych cudzysłowii (podwójnego w pojedynczym — `'"...'`).
- ▶ Nigdy nie należy polegać na nazwach zmiennych, zmienna `x` w interfejsie Pythona może być tłumaczona na `sage352` gdy prześlemy ją do konsoli R

Sytuacja na chwilę obecną (c.d.)

Warto zaznaczyć, że z poziomu konsoli lub gdy użytkownikowi jako który uruchamiany jest notebook dany odpowiednie uprawnienia do katalogu R'a możliwe jest instalowanie dodatkowych pakietów dokładnie tak samo jak w przypadku zwykłej instalacji R'a, natomiast z poziomu konsoli możliwe jest też zainstalowanie pakietów do własnego katalogu użytkownika (z oczywistych przyczyn opcja ta nie jest dostępna dla interfejsu www).

Wypad w przyszłość

Nie ma się co spierać, że R jest doskonałym pakietem do statystyki. Włączenie R'a do pakietu SAGE otwiera zupełnie nowe możliwości dla tych którzy zdecydują się na używanie R z poziomu SAGE'a tak samo jak otworzyło nowe możliwości dla użytkowników SAGE'a.

Wypad w przyszłość

Nie ma się co spierać, że R jest doskonałym pakietem do statystyki. Włączenie R'a do pakietu SAGE otwiera zupełnie nowe możliwości dla tych którzy zdecydują się na używanie R z poziomu SAGE'a tak samo jak otworzyło nowe możliwości dla użytkowników SAGE'a. Nie trudno wyobrazić sobie moduł statystyki symbolicznej pomocny przy pisaniu pracy teoretycznej. Już teraz za pomocą notebooka możemy stworzyć i publikować w sieci dynamiczne analizy podobne do tych jakie daje nam RPad lub inne interfejsy www dla R. Mamy dodatkowo możliwość wykorzystania w swoim programie dowolnego moduły Pythona jaki zainstalujemy na swoim komputerze — przykładowo istnieje wiele pakietów do obsługi baz danych czy pobierania plików z różnych sieci, warto wspomnieć też o pakiecie BioPython, który może być używany z poziomu SAGE'a już dziś jeśli tylko zainstalujemy go na swoim komputerze.

Wypad w przyszłość (c.d.)

Integracja arkusza kalkulacyjnego do wprowadzania i edycji danych takiego jak Google Spreadsheet to dzięki interfejsowi www tylko kwestia czasu (i zapotrzebowania).

Wypad w przyszłość (c.d.)

Integracja arkusza kalkulacyjnego do wprowadzania i edycji danych takiego jak Google Spreadsheet to dzięki interfejsowi www tylko kwestia czasu (i zapotrzebowania).

Już teraz dostępna jest też podstawowa wersja projektu DSage stworzonego z myślą o obliczeniach rozproszonych i równoległych, który jak tylko będzie gotowy prawdopodobnie w dość prosty sposób będzie mógł wzbogacić wszystkie pakiety które integruje SAGE o te możliwości.

Wypad w przyszłość (c.d.)

Integracja arkusza kalkulacyjnego do wprowadzania i edycji danych takiego jak Google Spreadsheet to dzięki interfejsowi www tylko kwestia czasu (i zapotrzebowania).

Już teraz dostępna jest też podstawowa wersja projektu DSage stworzonego z myślą o obliczeniach rozproszonych i równoległych, który jak tylko będzie gotowy prawdopodobnie w dość prosty sposób będzie mógł wzbogacić wszystkie pakiety które integruje SAGE o te możliwości.

Na koniec warto jeszcze raz podkreślić, że interfejs do R i statystyka w SAGE będą tak bardzo dopracowane i rozbudowane na ile będzie to potrzebne, ponieważ SAGE jest projektem w którym użytkownik jest najważniejszy.

Demonstracja możliwości

Pytania ?

Dziękuję za uwagę i życzę interesującej przygody z pakietem

