namaste
Incredible India
What India Dreams

India must master Western science and yet preserve its Culture and Heritage.
City of Pune.

Population: 6 million.
Oxford of the East.
Sameer Deshmukh

github.com/v0dro

@v0dro
Dr. Gopal Deshmukh  
Dr. Hemchandra Deshmukh  
Dr. Satish Deshmukh  
Sameer Desmukh
Sameer

www.soundcloud.com/catkamikazee
emerging technology trust
Pune Ruby Users Group

@punerb
@punerb
www.punerb.org

@deccanrubyconf
www.deccanrubyconf.org
Scientific Computing In Ruby
iruby notebook
Browser based Ruby REPL for interactive computing.
Runs in your browser

Input cell – accepts Ruby code

Output cell – can render HTML/CSS/JS

Renders:
```
In [1]: require 'matrix'
Matrix[[1,2,3], [1,2,3], [1,2,3]]

Out[1]:
\[
\begin{pmatrix}
1 & 2 & 3 \\
1 & 2 & 3 \\
1 & 2 & 3 \\
\end{pmatrix}
\]
```
The Everything Form
Marvel at the strange and varied inputs!

- Date
- File: Choose File
- Username
- Password
- Textarea
- Radio: 1 2 3 4 5 6 7 8 9 10
- Animals: Fish, Cat, Dog
- Color: blue

[ ]:
nmatrix
An n-dimensional array object.

Interface Ruby with high speed C libraries.
require 'nmatrix'

n = NMatrix.new(
  [2,2],
  [1,2,3,4],
  dtype: :float32,
  stype: :dense
)

n[0,1] #=> 2.0
Data Types

:int8         :float32
:int16        :float64
:int32        :complex64
:int64        :complex128
<table>
<thead>
<tr>
<th>Storage types</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dense</td>
<td>Dense matrix.</td>
</tr>
<tr>
<td>List</td>
<td>Sparse matrix type storing data as a linked list.</td>
</tr>
<tr>
<td>Yale</td>
<td>Sparse type storing data in the 'New Yale' format.</td>
</tr>
</tbody>
</table>
NMatrix C API
nyaplot
Interactive plotting tool for Rubyists.
interactive HTML and JavaScript plots that can be displayed in your browser.
Mapnya
Map visualizations with inbuilt country charts.

Nyaplot3D
Three Dimensional interactive plots.

Bionya
Biology plots for visualizing relationships of genes.
daru
(Data Analysis in Ruby)
daru
(=)
darù
= =
sake
alcohol
library for
analysis, cleaning, manipulation and
visualization
of data.
Data indexing

Read/write many data sources

Works well with 'wild' data

Ephemeral statistics functions
Acts as glue between other SciRuby libraries.
Daru::Vector

Heterogenous Array that can be indexed on any Ruby object.
Daru::DataFrame

2D spreadsheet like data structure indexed by rows or columns.

<table>
<thead>
<tr>
<th>Label(0)</th>
<th>Col0</th>
<th>Col1</th>
<th>Col2</th>
<th>Col(n-1)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Label(1)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Label(2)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Label(n-1)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
New Ideas for better Ruby
“Any sufficiently advanced technology is indistinguishable from magic.”

- Arthur C. Clarke
Writing C extensions

- FFI gem.
- Rice.
- SWIG.
- Writing C bindings manually.
Rubyist!
Write me a C extension!
def factorial n
    n > 1 ? n*factorial(n-1) : 1
end
unsigned long long int
calc_factorial(unsigned long long int n)
{
    return (n > 1 ? n * calc_factorial(n - 1) : 1);
}

static VALUE
cfactorial(VALUE self, VALUE n)
{
    return ULL2FIX(calc_factorial(NUM2ULL(n)));
}
void Init_factorial()
{
    VALUE cFact = rb_define_class("Fact",
    rb_cObject);

    rb_define_method(cFact, "factorial",
    cfactorial, 1);
}
a = Fact.new

a.factorial(8000)
Big Problems

- **Difficult and irritating** to write.
- Time consuming to **debug**.
- Tough to **trace memory leaks**.
- **Change mindset** from high level to low level language.
- **Need to care about small things.**

rubex - A Crystal-inspired language for writing Ruby extensions. — Edit
Rubex is a Crystal-inspired superset of Ruby that compiles to C.
class Fact
    def factorial(unsigned long long int n)
        n > 1 ? n*factorial(n-1) : 1
    end
end
end
# Create a C static array and return a Ruby Array

def adder(n)
    a = StaticArray(i32, n)
    i32 i = 0
    i32 sum = 0

    a.each(n) { a[i] = i*5 }
    for 0 <= i < n do
        sum += a[i]
    end

    sum
end
Received the
Ruby Association Grant 2016
for development of Rubex
https://github.com/v0dro/rubex
Scientific Computing on JRuby
NMatrix and NArray are a linear algebra libraries for Ruby similar to numpy.
NMatrix

C/C++ core

CRuby interpreter

Numo::NArray

C core

CRuby interpreter
JRuby backend for the NMatrix Ruby API – Sci. Computing on JVM.
Uses Apache Commons Math library for storage and operations on internal Java arrays.

Allows interfacing JRuby libraries with jBLAS for performance.
https://github.com/prasunananand/nmatrix/tree/jruby_port
Symbolic Computation in Ruby with symengine.rb
\[(x - y) \times (x ** \frac{y}{z})\]
require 'symengine'

x = SymEngine::Symbol.new("x")
y = SymEngine::Symbol.new("y")
z = SymEngine::Symbol.new("z")

f = (x - y) * (x ** y / z)
f.expand.to_s
  # x**(1 + y)/z - x**y*y/z

f == - (x**y*y/z) + (x**y*x/z)
  # true
https://github.com/symengine/symengine.rb
Ruby in Space
NASA SPICE
Ruby wrapper spice_rub
require 'spice_rub'

k_pool = SpiceRub::KernelPool.instance
k_pool.load_folder("spec/data/kernels")

epoch = SpiceRub::Time.now
moon = SpiceRub::Body.new(:moon)
earth = SpiceRub::Body.now(:earth)

earth.position_at(epoch)
moon.distance_from(:earth, epoch)
# 395791.1464913574 (Km)
https://github.com/gau27/spice_rub
Cool SciRuby Stickers
Thank You
Ruby World Conf!
Any questions?