CSC 599.70 Introduction to Data Science

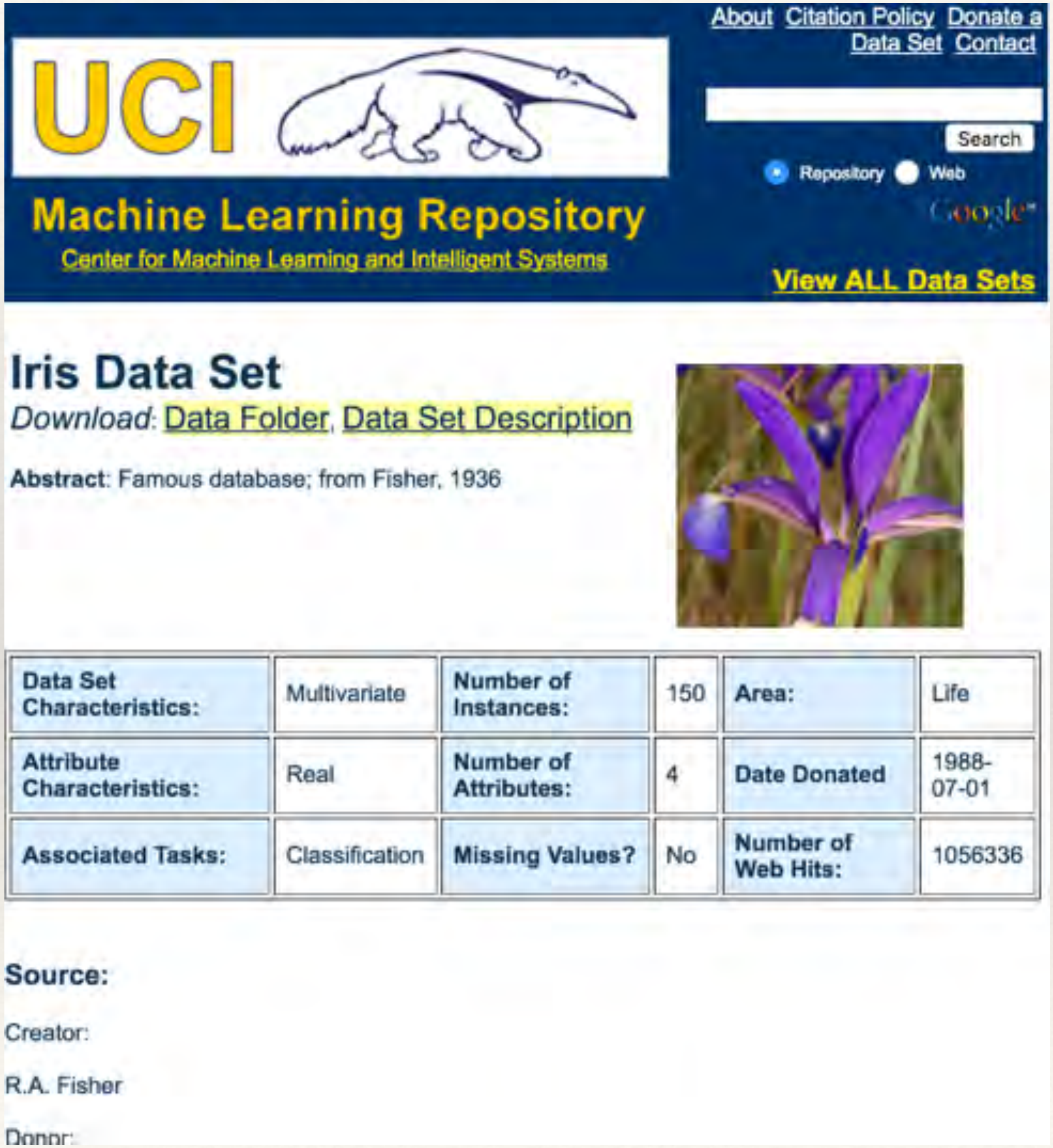
Introduction to DS Tools

grossberg@cs.ccny.cuny.edu

What can we use to do Data Science?

Excel?

Iris Data Set


About Citation Policy Donate a Data Set Contact
Search
● Repository ○ Web
Google
UCI
Machine Learning Repository
Center for Machine Learning and Intelligent Systems
View ALL Data Sets

Iris Data Set

Download: Data Folder, Data Set Description

Abstract: Famous database; from Fisher, 1936

Data Set Characteristics:	Multivariate	**Number of Instances:**	150	**Area:**	Life
Attribute Characteristics:	Real	**Number of Attributes:**	4	**Date Donated**	1988-07-01
Associated Tasks:	Classification	**Missing Values?**	No	**Number of Web Hits:**	1056336

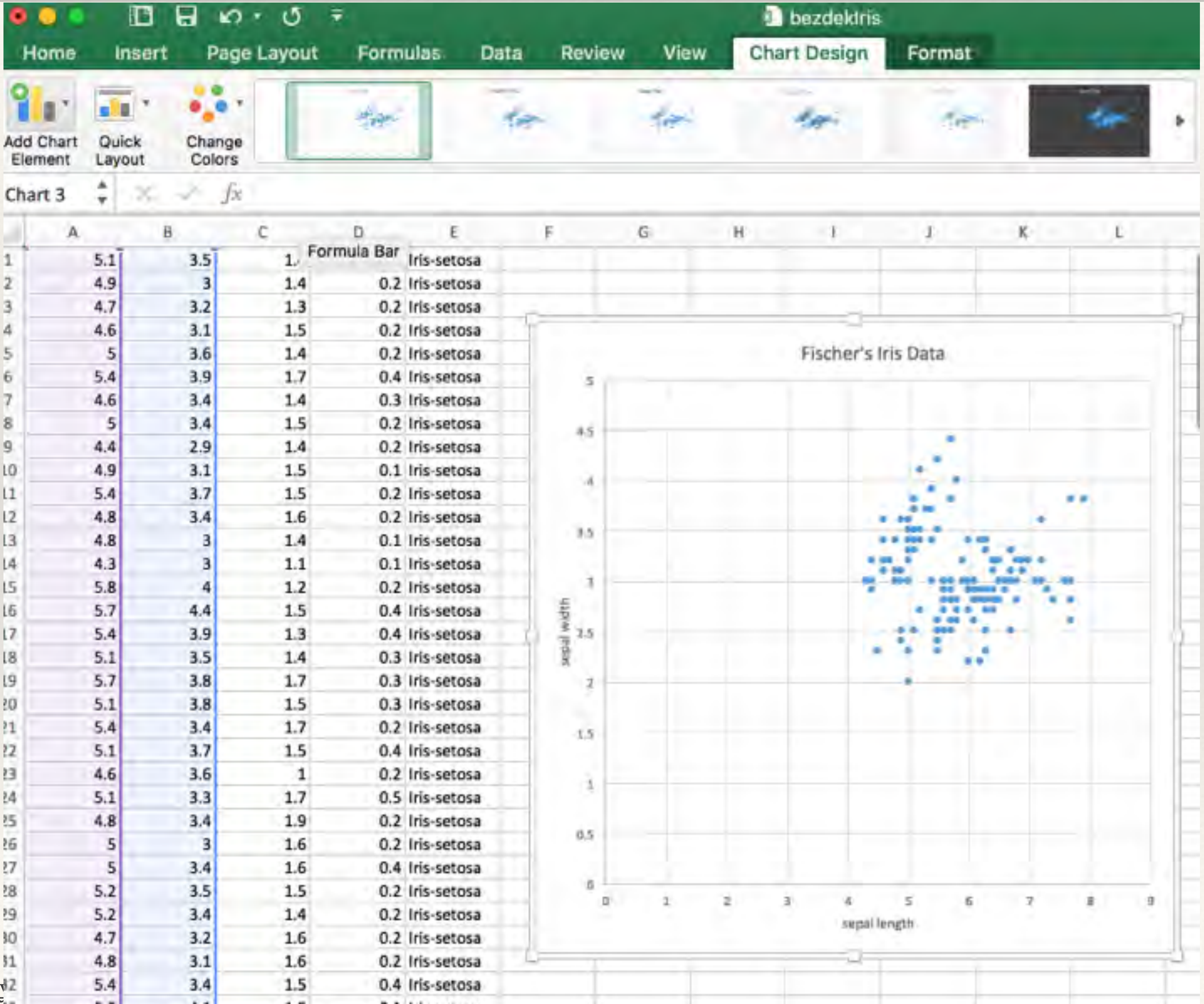
Source:

Creator:

R.A. Fisher

Donor:

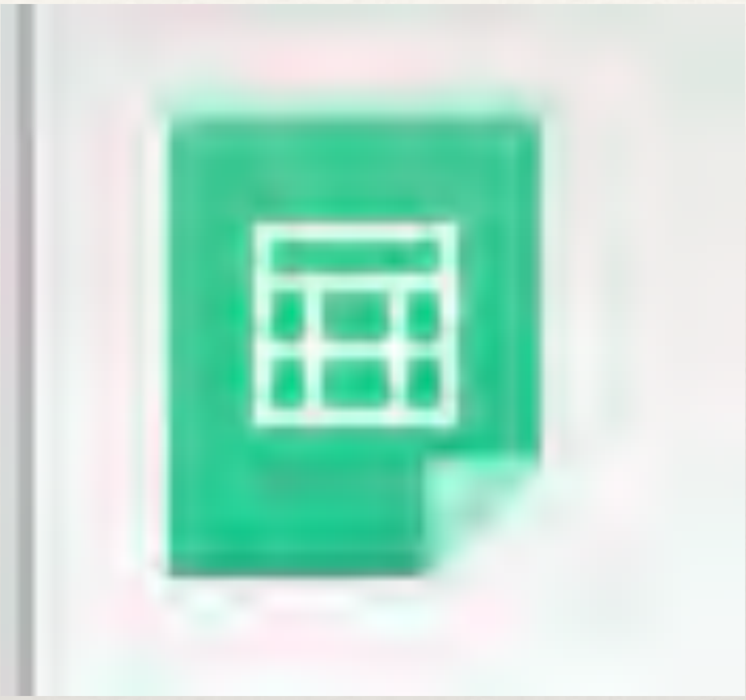
Basic Scatter Plots and Analysis



	A	B	C	D	E
1	5.1	3.5	1.		Iris-setosa
2	4.9	3	1.4	0.2	Iris-setosa
3	4.7	3.2	1.3	0.2	Iris-setosa
4	4.6	3.1	1.5	0.2	Iris-setosa
5	5	3.6	1.4	0.2	Iris-setosa
6	5.4	3.9	1.7	0.4	Iris-setosa
7	4.6	3.4	1.4	0.3	Iris-setosa
8	5	3.4	1.5	0.2	Iris-setosa
9	4.4	2.9	1.4	0.2	Iris-setosa
10	4.9	3.1	1.5	0.1	Iris-setosa
11	5.4	3.7	1.5	0.2	Iris-setosa
12	4.8	3.4	1.6	0.2	Iris-setosa
13	4.8	3	1.4	0.1	Iris-setosa
14	4.3	3	1.1	0.1	Iris-setosa
15	5.8	4	1.2	0.2	Iris-setosa
16	5.7	4.4	1.5	0.4	Iris-setosa
17	5.4	3.9	1.3	0.4	Iris-setosa
18	5.1	3.5	1.4	0.3	Iris-setosa
19	5.7	3.8	1.7	0.3	Iris-setosa
20	5.1	3.8	1.5	0.3	Iris-setosa
21	5.4	3.4	1.7	0.2	Iris-setosa
22	5.1	3.7	1.5	0.4	Iris-setosa
23	4.6	3.6	1	0.2	Iris-setosa
24	5.1	3.3	1.7	0.5	Iris-setosa
25	4.8	3.4	1.9	0.2	Iris-setosa
26	5	3	1.6	0.2	Iris-setosa
27	5	3.4	1.6	0.4	Iris-setosa
28	5.2	3.5	1.5	0.2	Iris-setosa
29	5.2	3.4	1.4	0.2	Iris-setosa
30	4.7	3.2	1.6	0.2	Iris-setosa
31	4.8	3.1	1.6	0.2	Iris-setosa
32	5.4	3.4	1.5	0.4	Iris-setosa

Quick look at “small data”

Google Sheet



LibreOffice Calc

Apple Numbers

Tableau

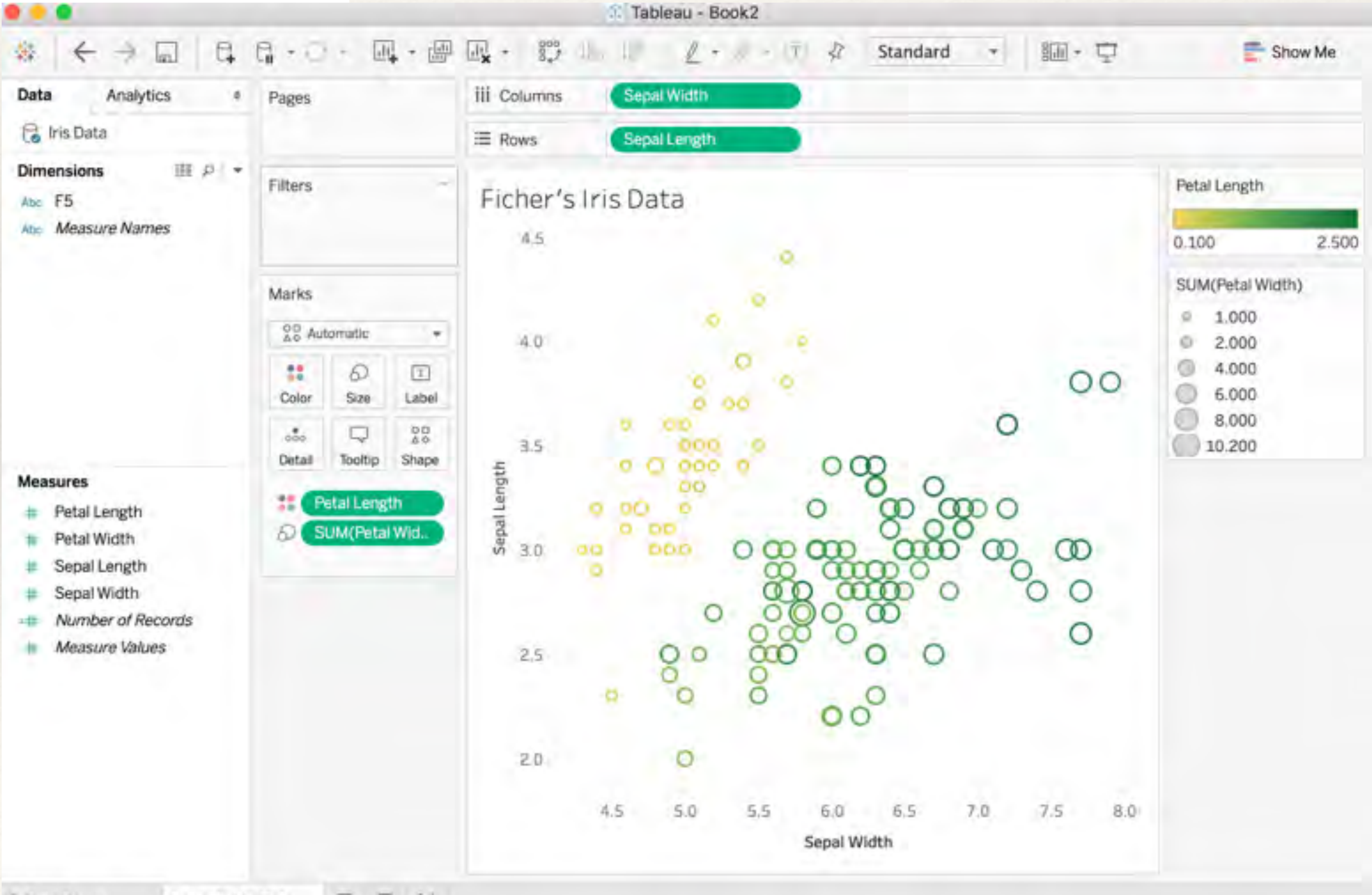


Tableau - Book2

Columns: Sepal Width

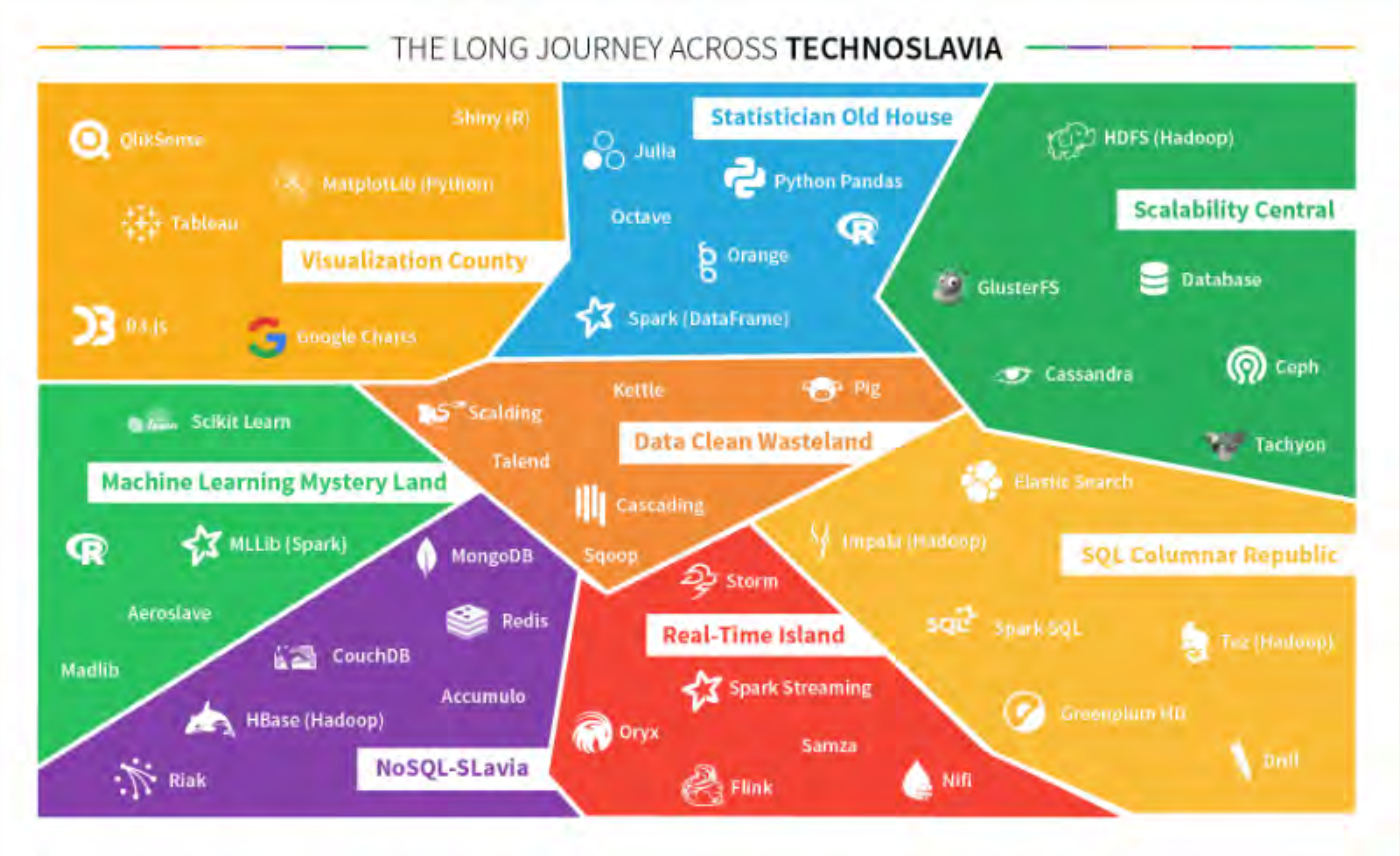
Rows: Sepal Length

Marks: Automatic

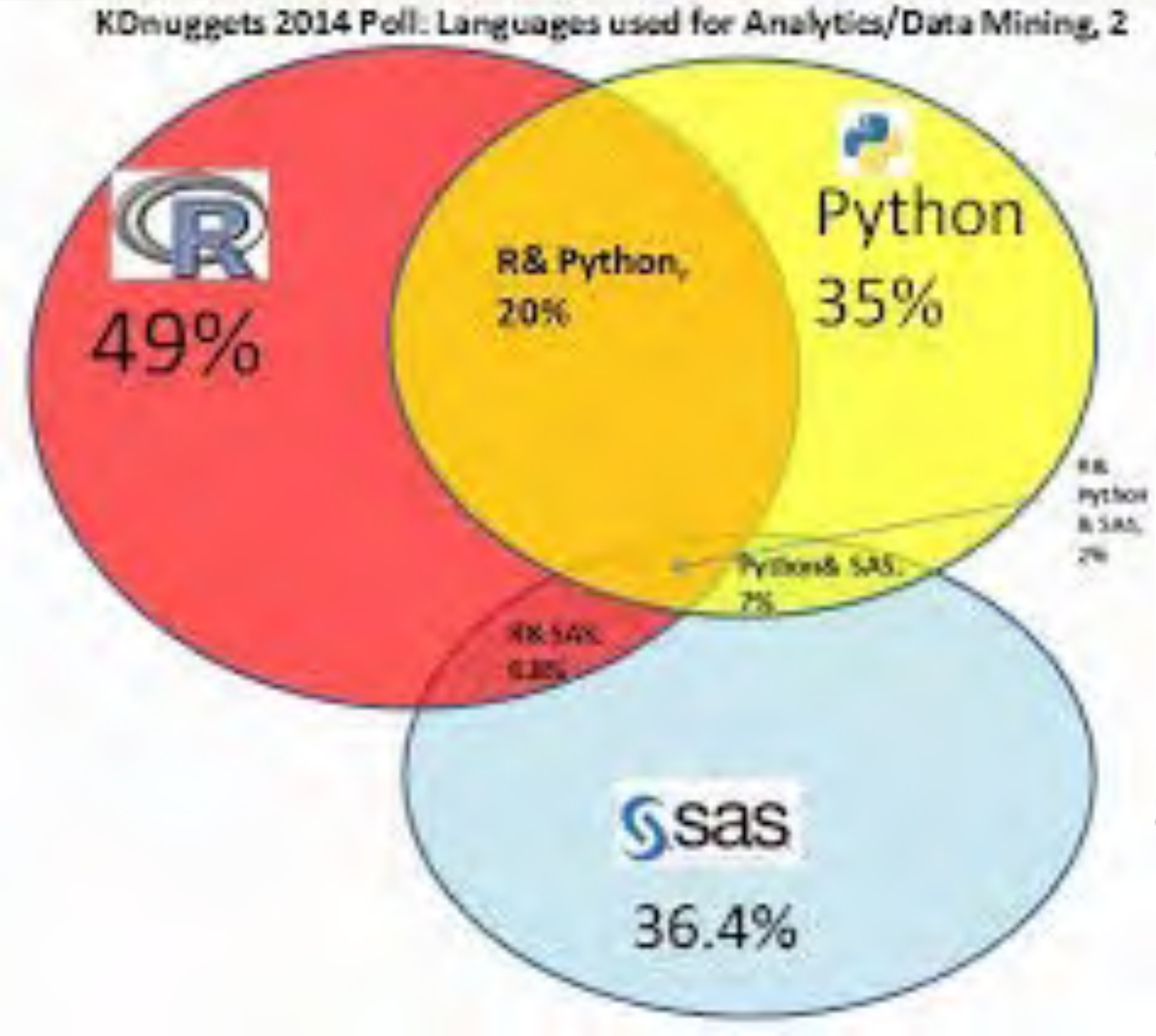
Color: Petal Length

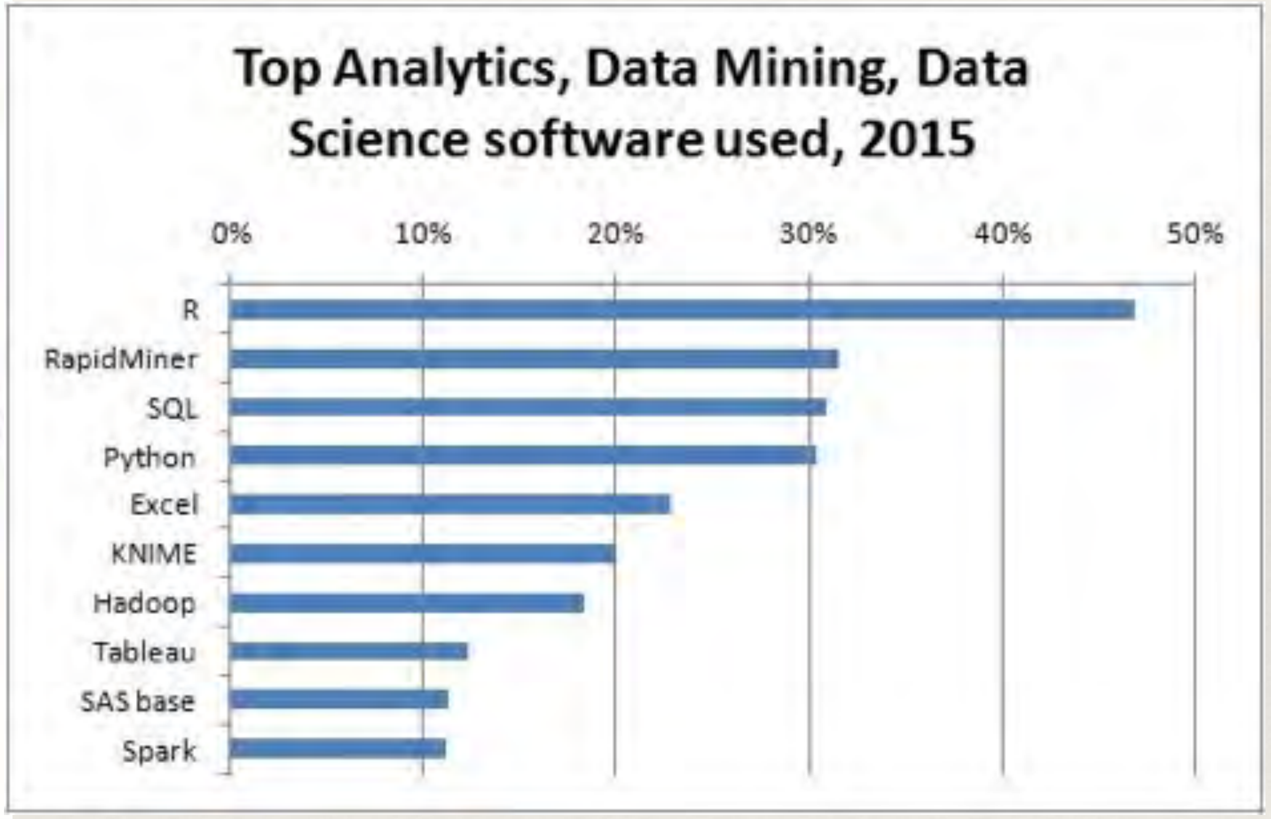
Size: SUM(Petal Wid..

Languages/Tools for Data Science

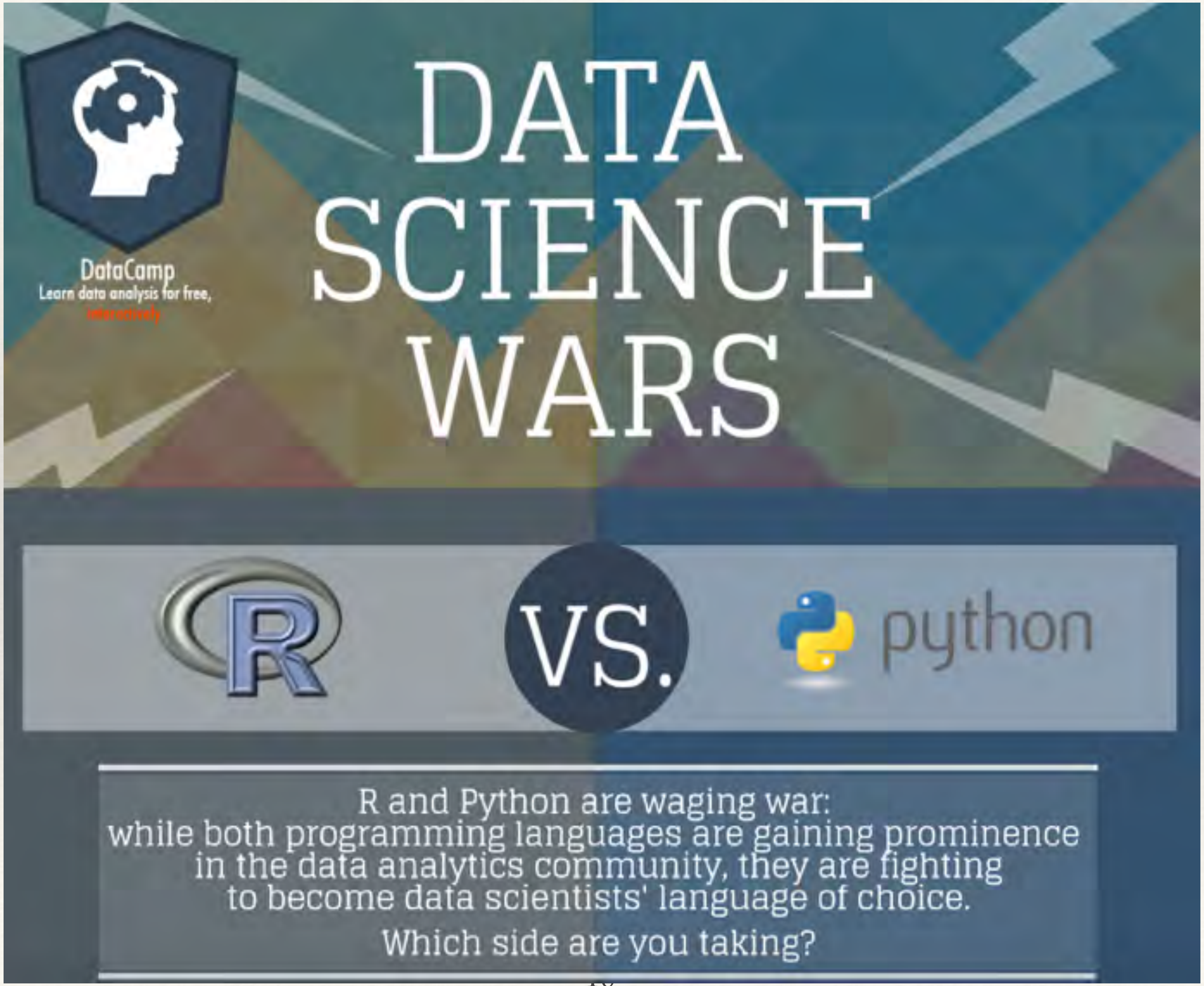


Programatic Analysis





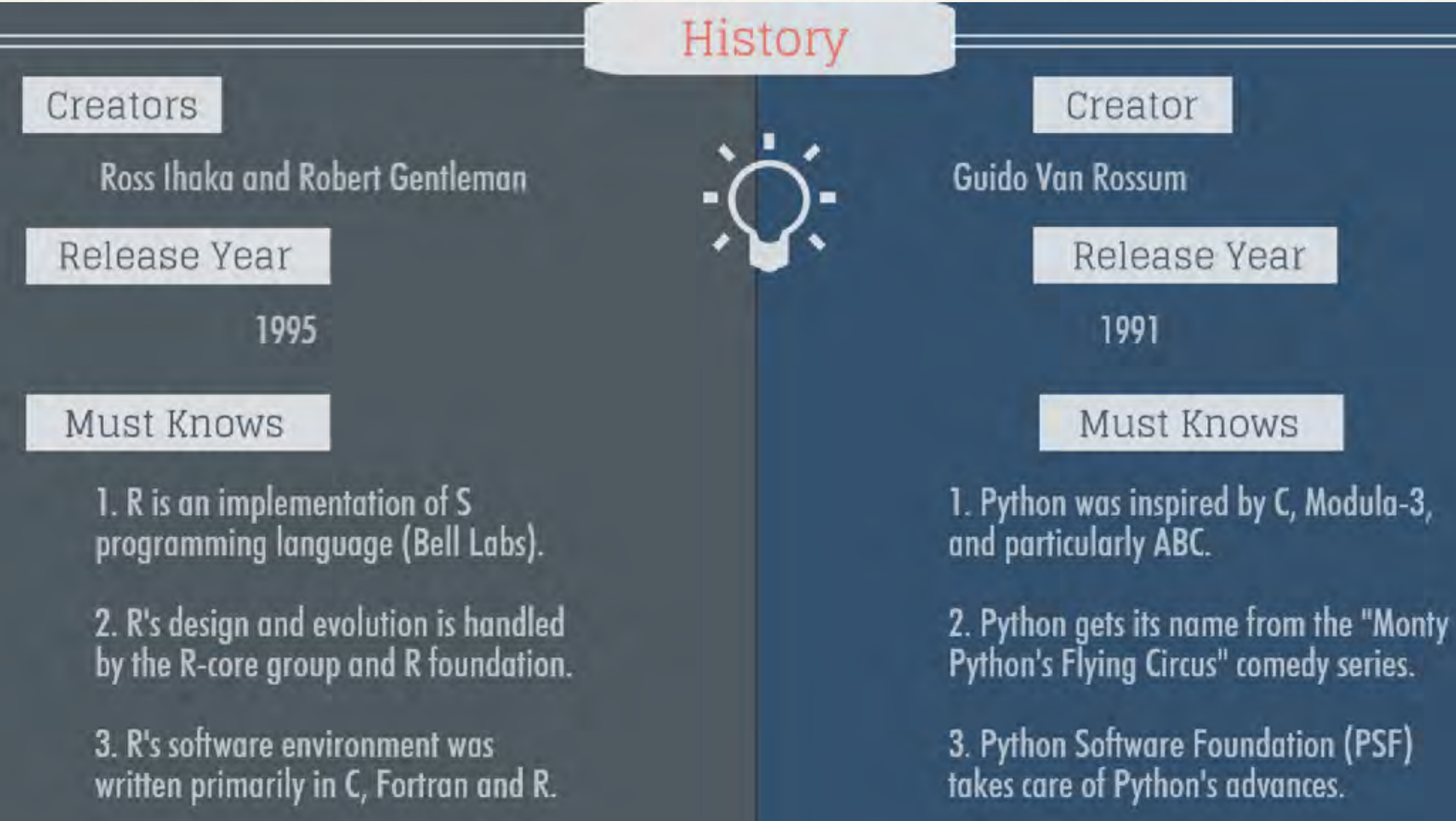
Python vs R: Data Science WAR



R and Python are waging war:
while both programming languages are gaining prominence in the data analytics community, they are fighting to become data scientists' language of choice.

Which side are you taking?

R vs Python



History

Creators

Ross Ihaka and Robert Gentleman

Release Year

1995

Must Knows

1. R is an implementation of S programming language (Bell Labs).

2. R's design and evolution is handled by the R-core group and R foundation.

3. R's software environment was written primarily in C, Fortran and R.

Creator

Guido Van Rossum

Release Year

1991

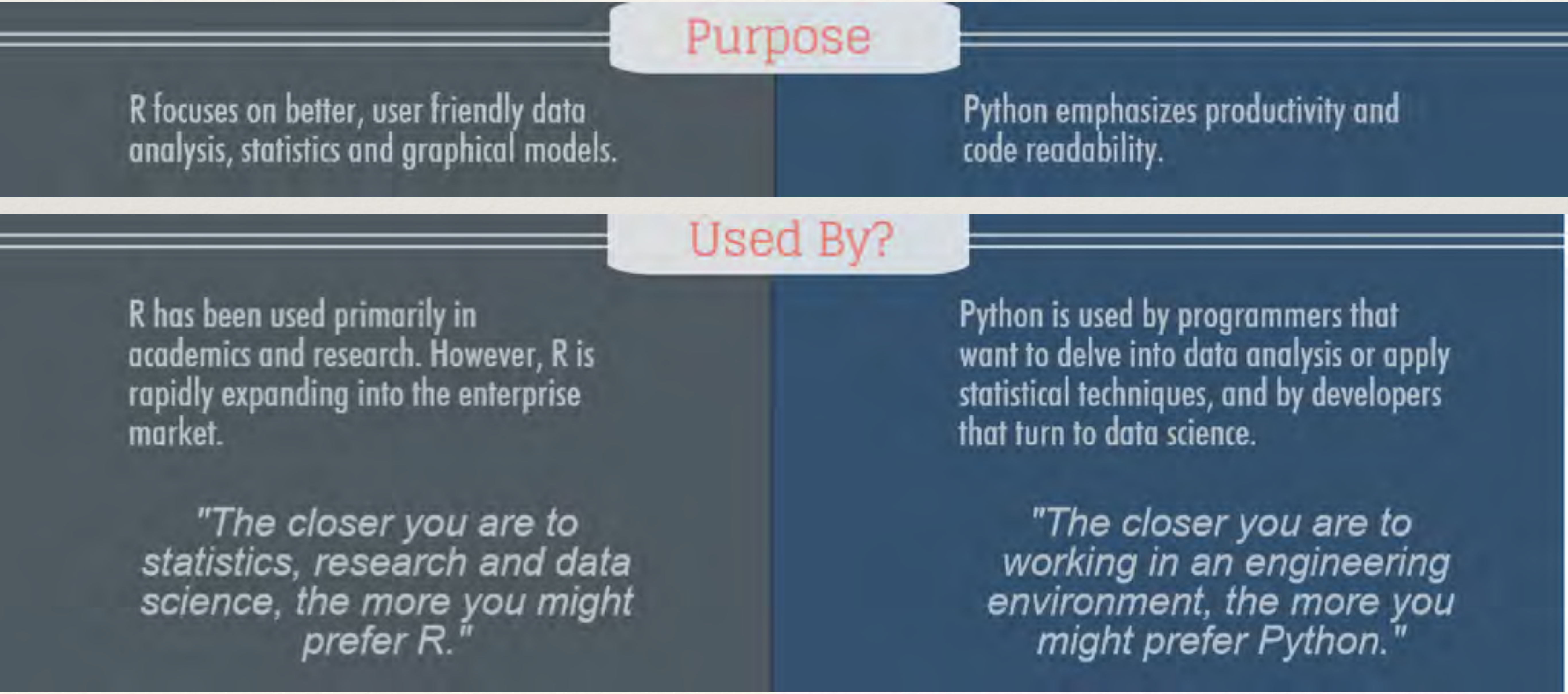
Must Knows

1. Python was inspired by C, Modula-3, and particularly ABC.

2. Python gets its name from the "Monty Python's Flying Circus" comedy series.

3. Python Software Foundation (PSF) takes care of Python's advances.

R vs Python



Purpose

R focuses on better, user friendly data analysis, statistics and graphical models.

Python emphasizes productivity and code readability.

Used By?

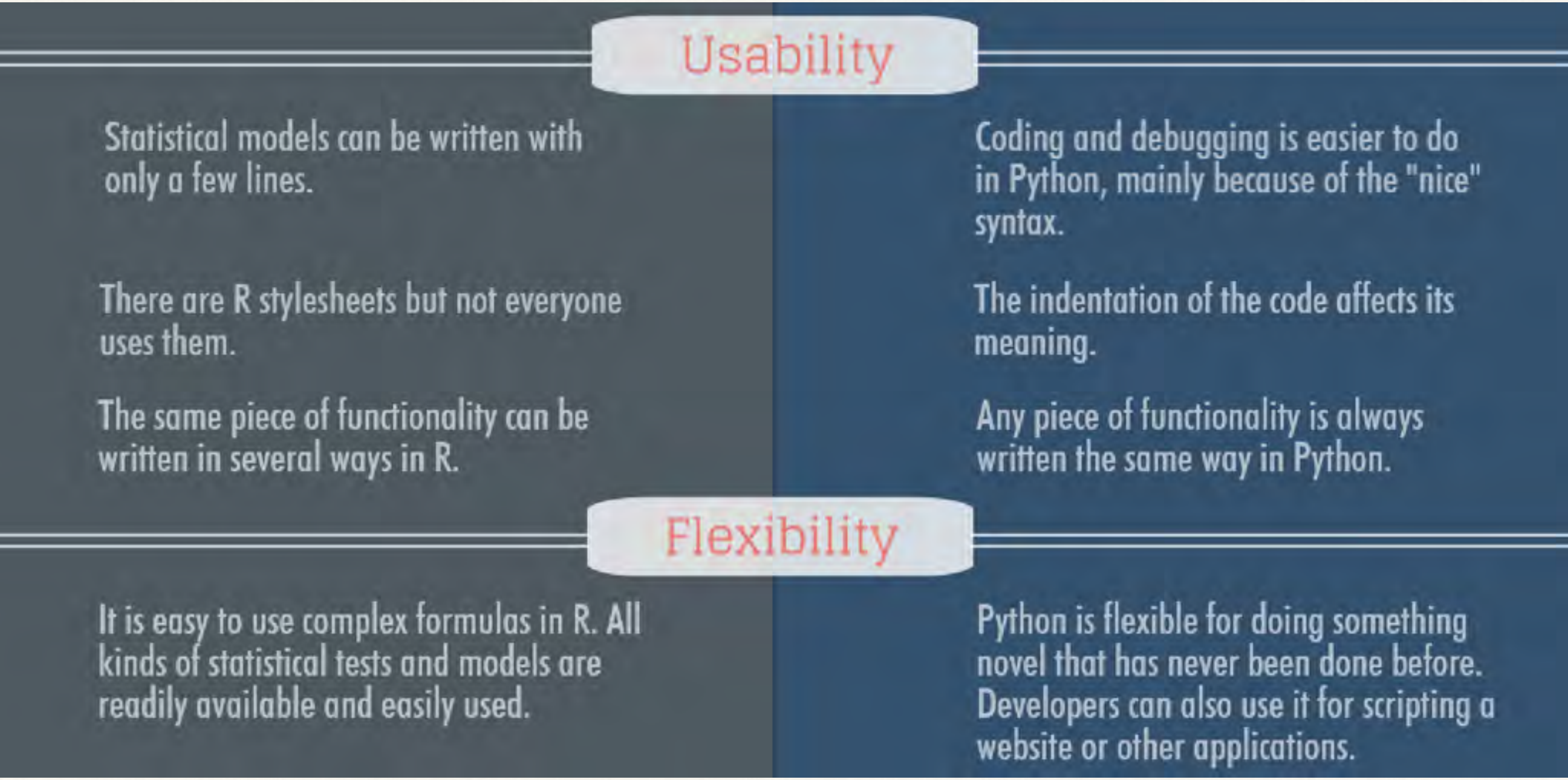
R has been used primarily in academics and research. However, R is rapidly expanding into the enterprise market.

"The closer you are to statistics, research and data science, the more you might prefer R."

Python is used by programmers that want to delve into data analysis or apply statistical techniques, and by developers that turn to data science.

"The closer you are to working in an engineering environment, the more you might prefer Python."

R vs Python

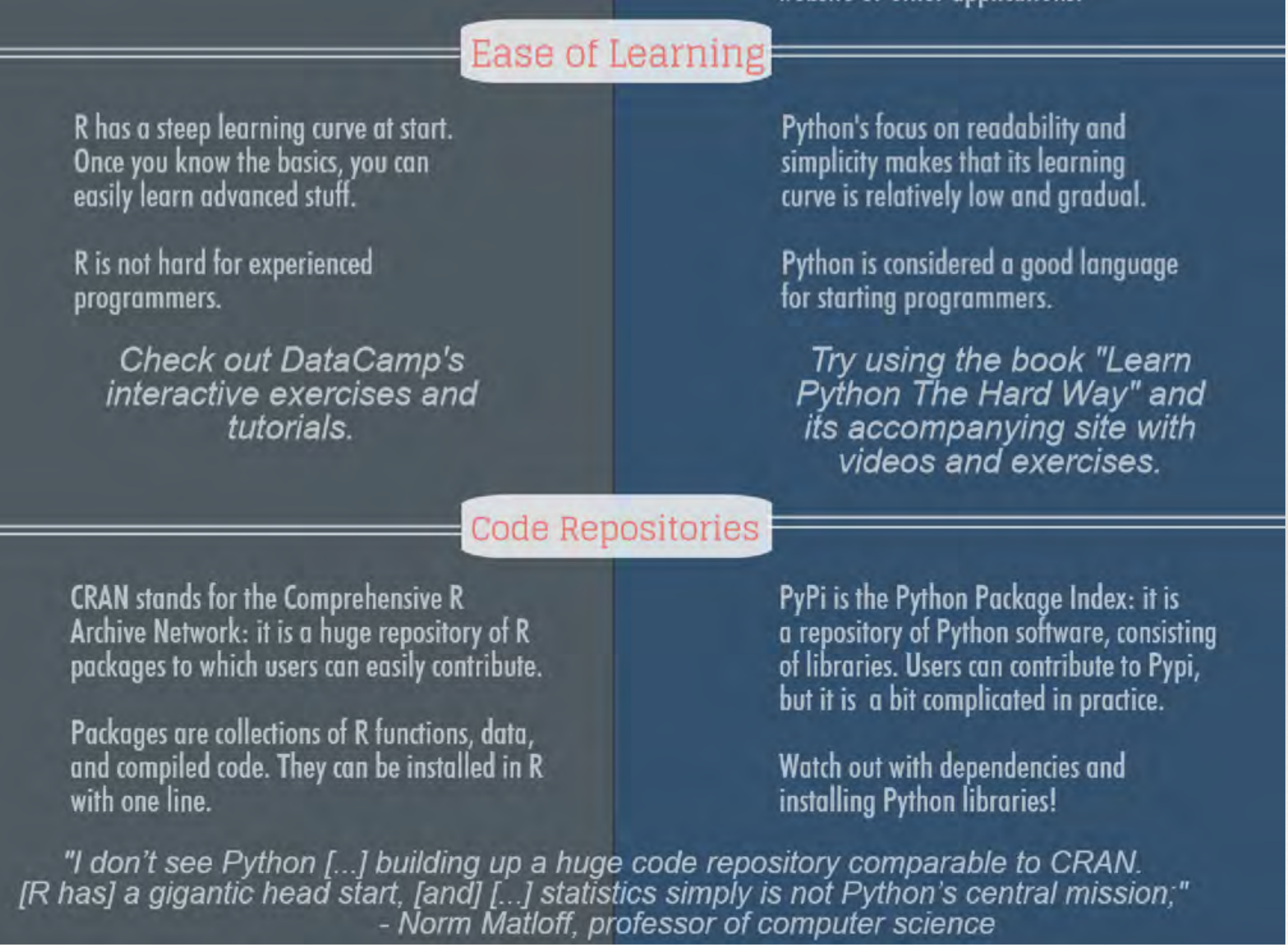


Usability

Statistical models can be written with only a few lines.	Coding and debugging is easier to do in Python, mainly because of the "nice" syntax.
There are R stylesheets but not everyone uses them.	The indentation of the code affects its meaning.
The same piece of functionality can be written in several ways in R.	Any piece of functionality is always written the same way in Python.

Flexibility

It is easy to use complex formulas in R. All kinds of statistical tests and models are readily available and easily used.	Python is flexible for doing something novel that has never been done before. Developers can also use it for scripting a website or other applications.



website or other applications.

Ease of Learning

R has a steep learning curve at start. Once you know the basics, you can easily learn advanced stuff.

R is not hard for experienced programmers.

Check out DataCamp's interactive exercises and tutorials.

Python's focus on readability and simplicity makes that its learning curve is relatively low and gradual.

Python is considered a good language for starting programmers.

Try using the book "Learn Python The Hard Way" and its accompanying site with videos and exercises.

Code Repositories

CRAN stands for the Comprehensive R Archive Network: it is a huge repository of R packages to which users can easily contribute.

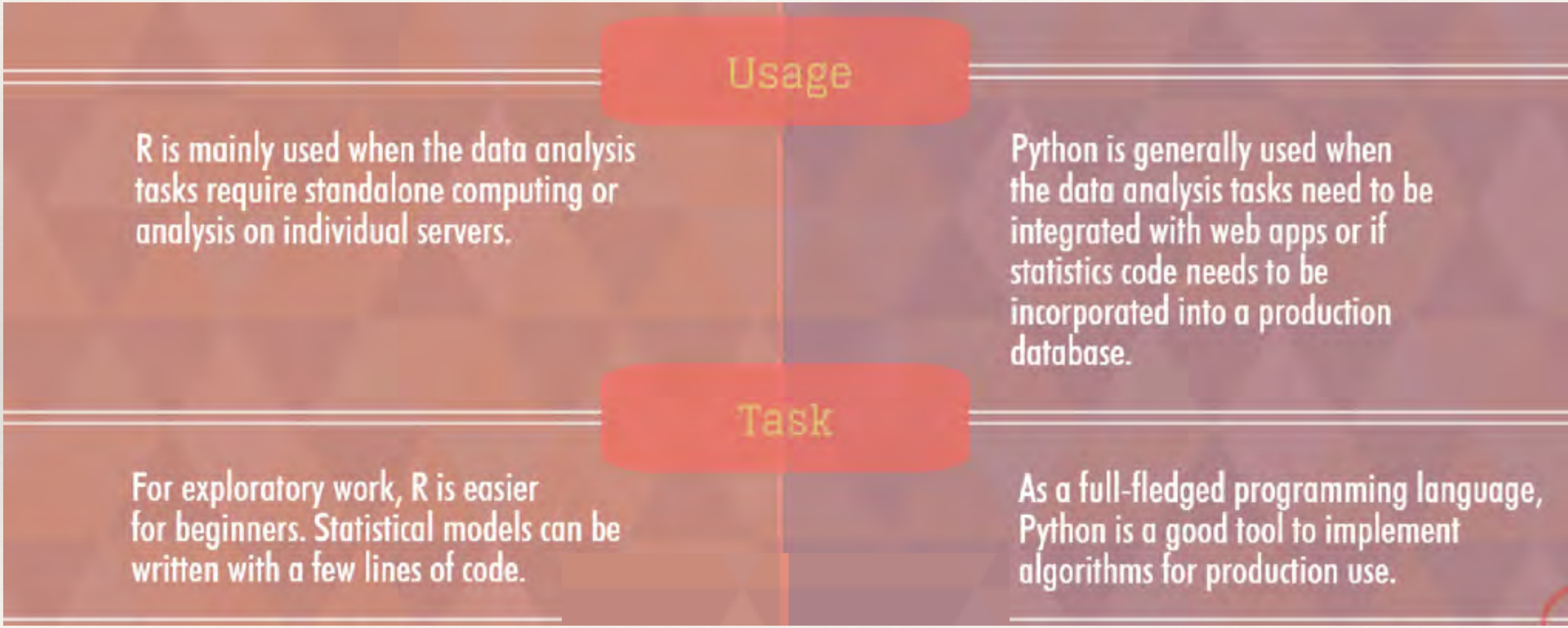
Packages are collections of R functions, data, and compiled code. They can be installed in R with one line.

PyPi is the Python Package Index: it is a repository of Python software, consisting of libraries. Users can contribute to Pypi, but it is a bit complicated in practice.

Watch out with dependencies and installing Python libraries!

"I don’t see Python [...] building up a huge code repository comparable to CRAN. [R has] a gigantic head start, [and] [...] statistics simply is not Python’s central mission;"
- Norm Matloff, professor of computer science

R vs. Python



Usage

R is mainly used when the data analysis tasks require standalone computing or analysis on individual servers.

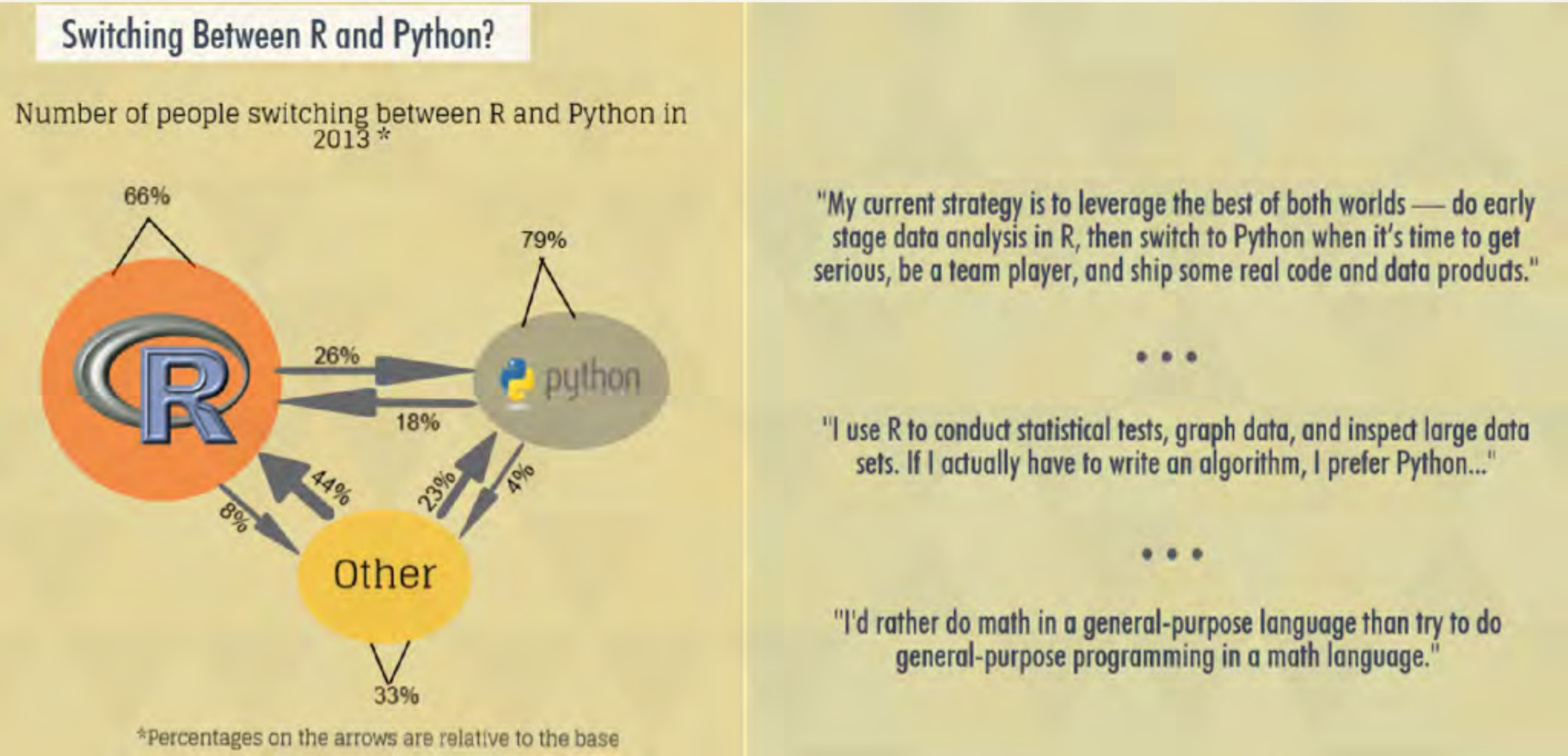
Python is generally used when the data analysis tasks need to be integrated with web apps or if statistics code needs to be incorporated into a production database.

Task

For exploratory work, R is easier for beginners. Statistical models can be written with a few lines of code.

As a full-fledged programming language, Python is a good tool to implement algorithms for production use.

Both Winners



Switching Between R and Python?

Number of people switching between R and Python in 2013 *

*Percentages on the arrows are relative to the base

"My current strategy is to leverage the best of both worlds — do early stage data analysis in R, then switch to Python when it's time to get serious, be a team player, and ship some real code and data products."

• • •

"I use R to conduct statistical tests, graph data, and inspect large data sets. If I actually have to write an algorithm, I prefer Python..."

• • •

"I'd rather do math in a general-purpose language than try to do general-purpose programming in a math language."

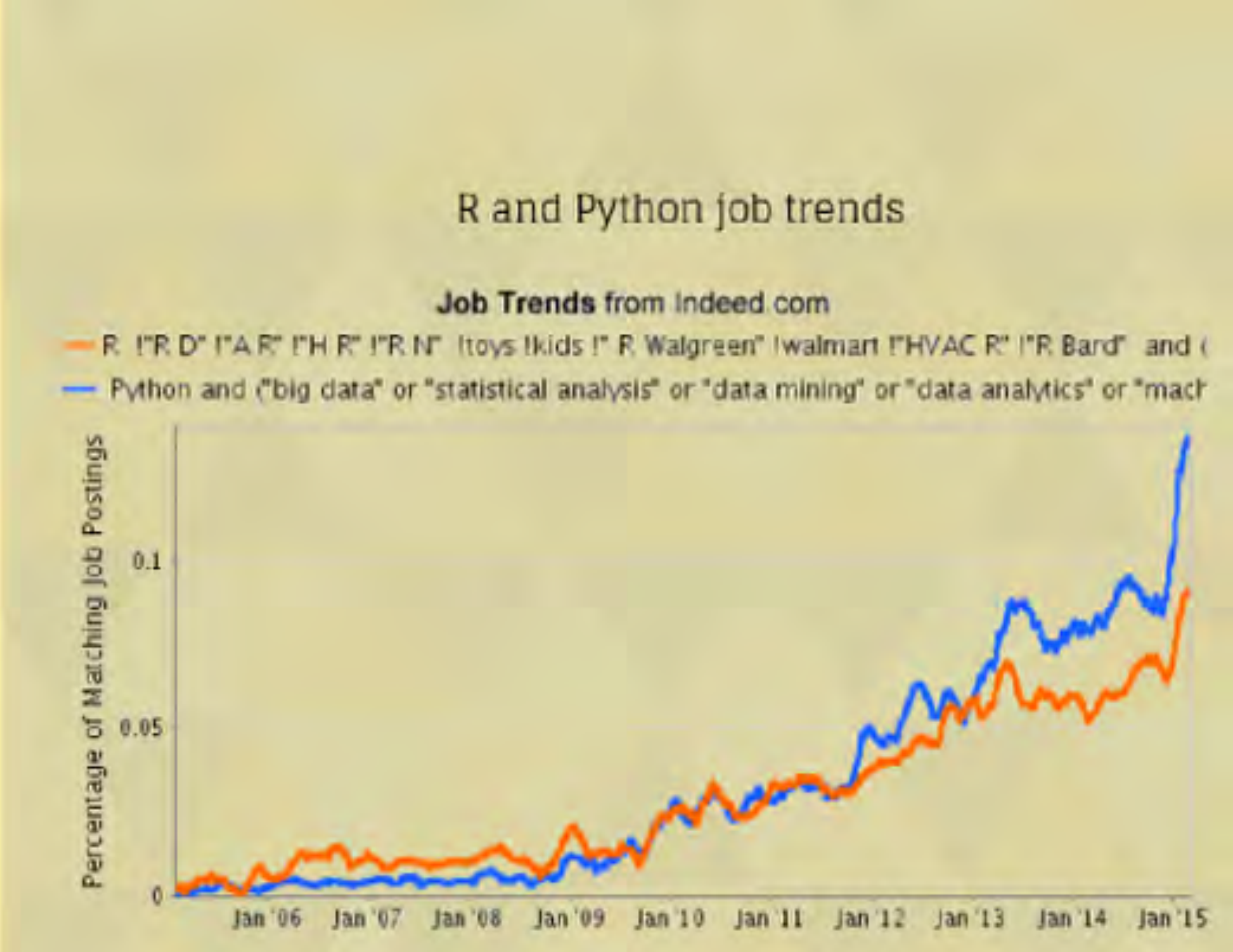
Jobs need both

Jobs and Salary?

O'Reilly 2014 Data Science Salary Survey

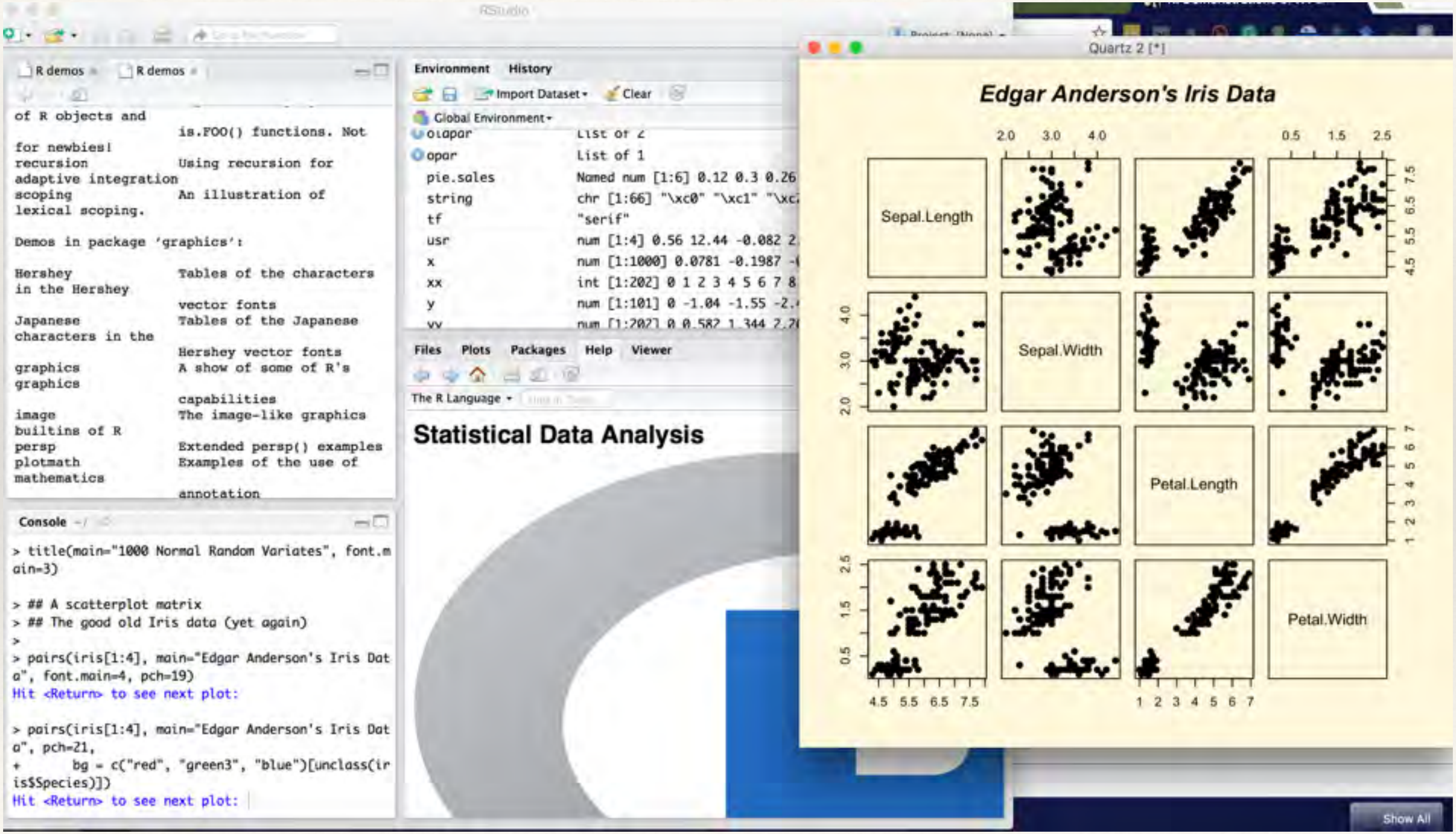
Average Annual Salaries In The Range Of:

Python = US$110, 000 to US$125,000 = R



R and Python job trends

R: IDE R-Studio

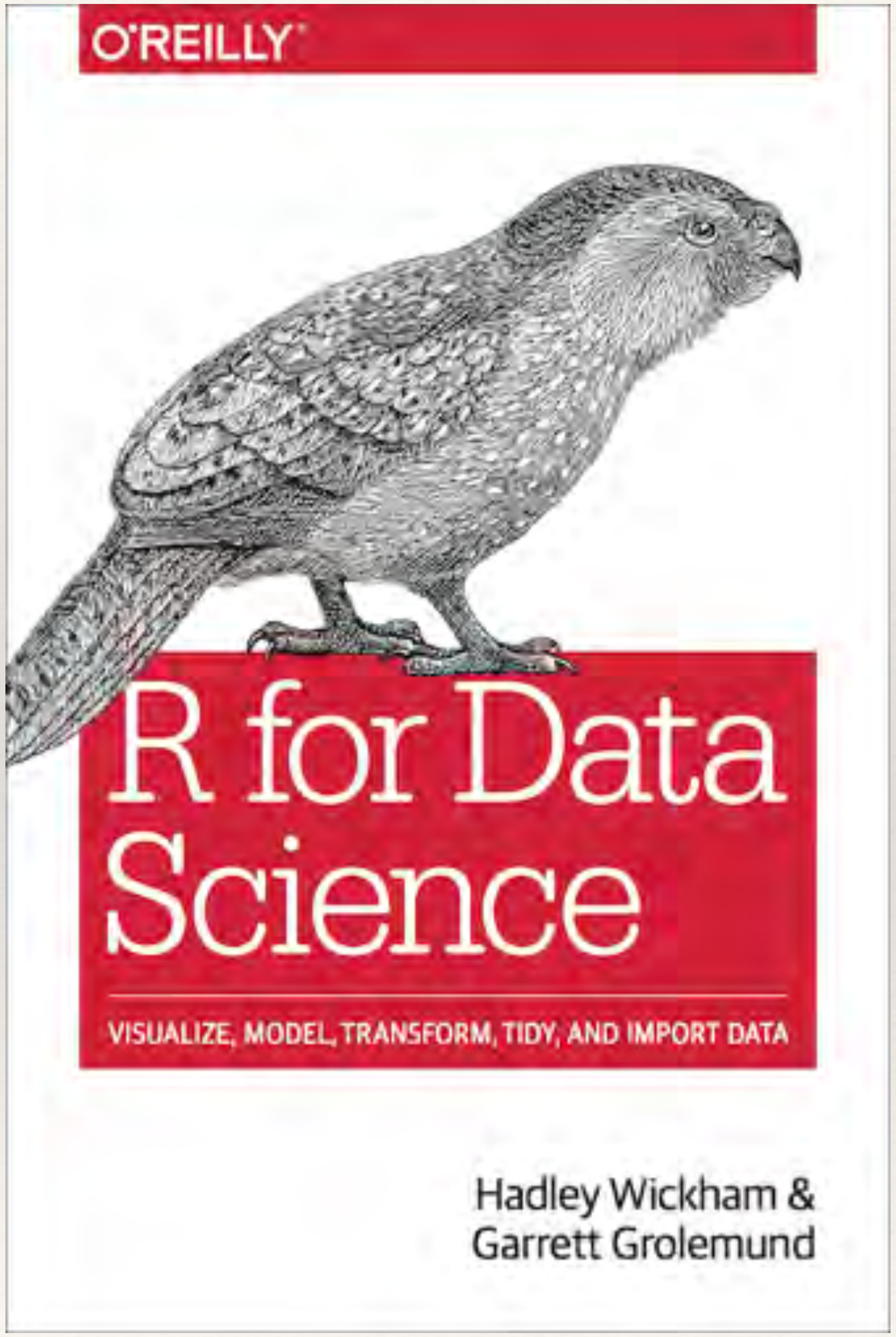


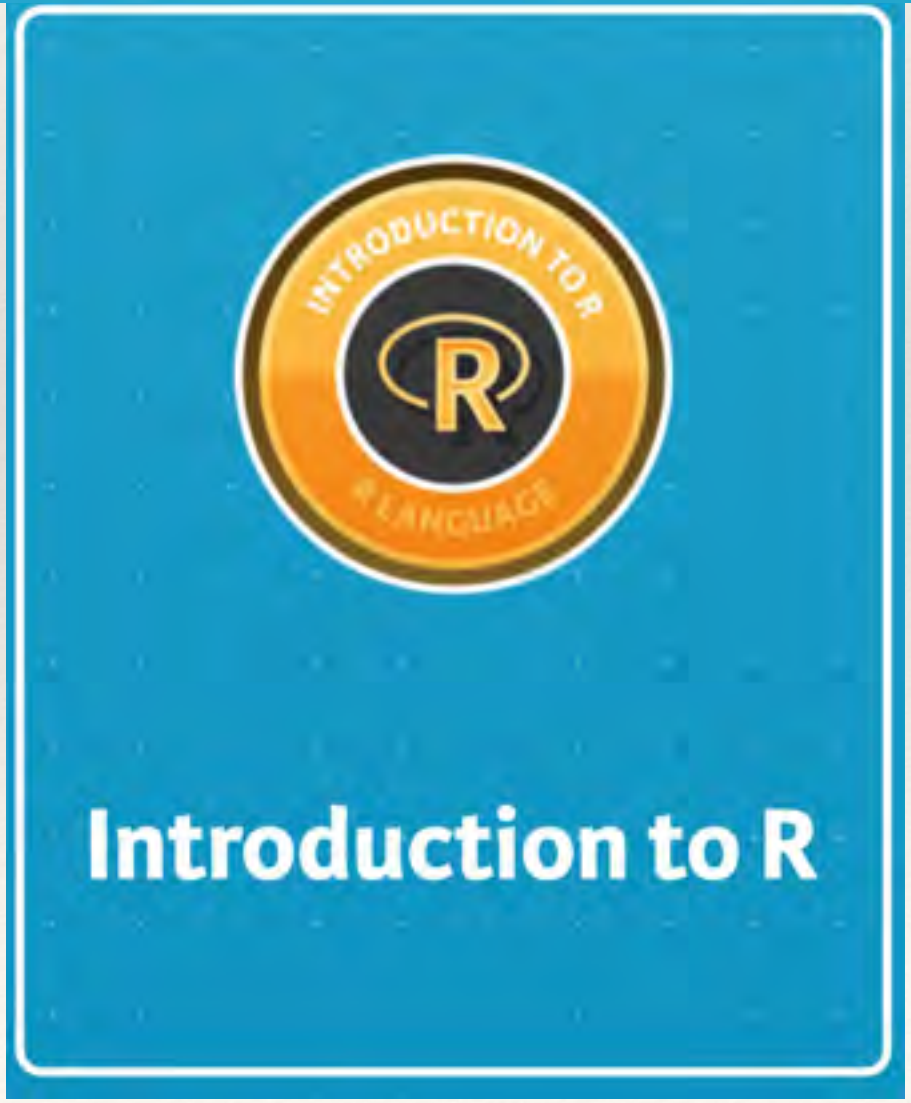
```
of R objects and
                            is.FOO() functions. Not
for newbies!
recursion                   Using recursion for
adaptive integration
scoping                     An illustration of
lexical scoping.

Demos in package 'graphics':

Hershey                     Tables of the characters
in the Hershey
                            vector fonts
Japanese                    Tables of the Japanese
characters in the
                            Hershey vector fonts
graphics                    A show of some of R's
graphics
                            capabilities
image                       The image-like graphics
builtins of R
persp                       Extended persp() examples
plotmath                    Examples of the use of
mathematics
                            annotation
```

```
oldpar       List of 2
opar         List of 1
pie.sales    Named num [1:6] 0.12 0.3 0.26
string       chr [1:66] "\xc0" "\xc1" "\xc
tf           "serif"
usr          num [1:4] 0.56 12.44 -0.082 2
x            num [1:1000] 0.0781 -0.1987 -
xx           int [1:202] 0 1 2 3 4 5 6 7 8
y            num [1:101] 0 -1.04 -1.55 -2.
yy           num [1:202] 0 0.582 1.344 2.7
```

```
> title(main="1000 Normal Random Variates", font.m
ain=3)

> ## A scatterplot matrix
> ## The good old Iris data (yet again)
>
> pairs(iris[1:4], main="Edgar Anderson's Iris Dat
a", font.main=4, pch=19)
Hit <Return> to see next plot:

> pairs(iris[1:4], main="Edgar Anderson's Iris Dat
a", pch=21,
+       bg = c("red", "green3", "blue")[unclass(ir
is$Species)])
Hit <Return> to see next plot:
```

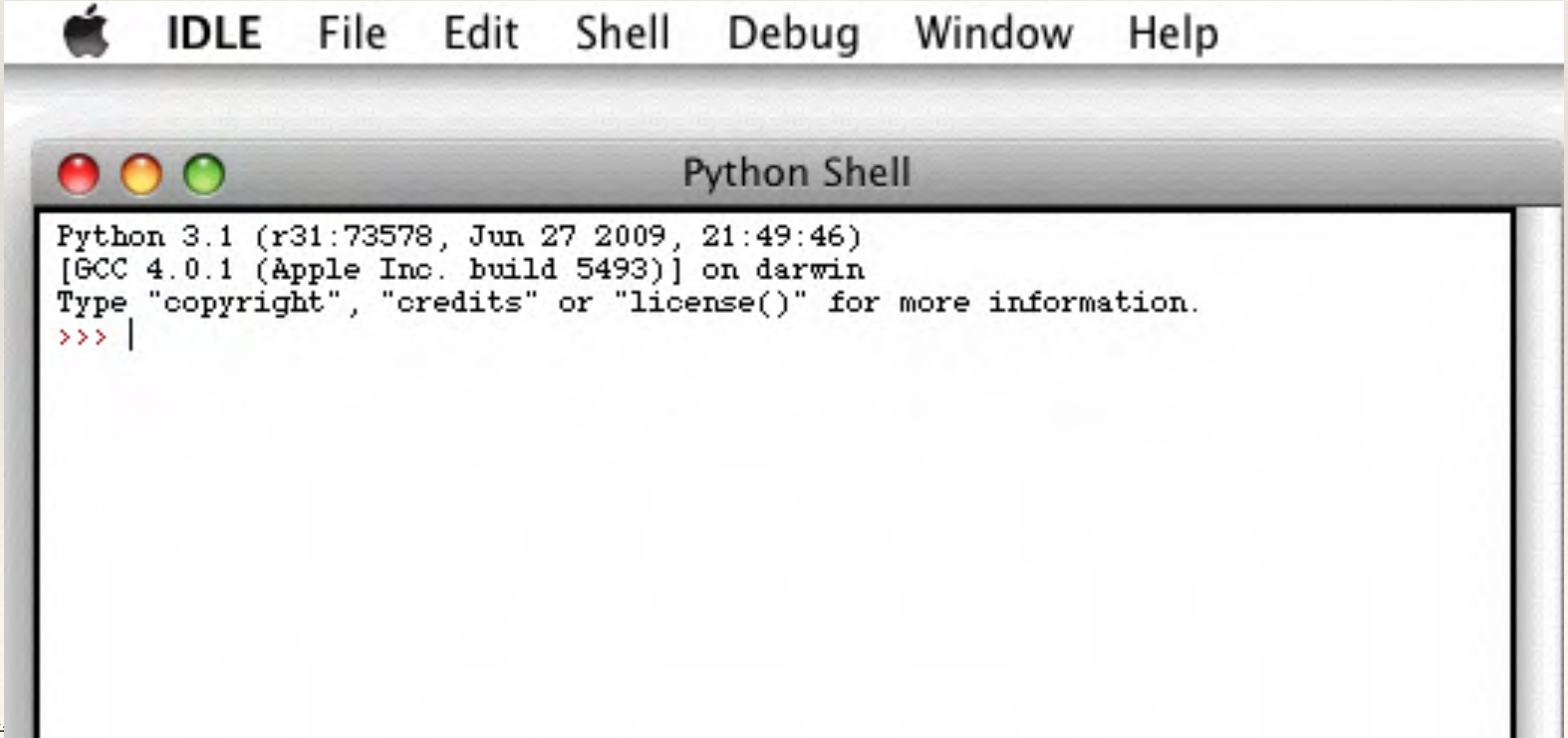
Some R resources





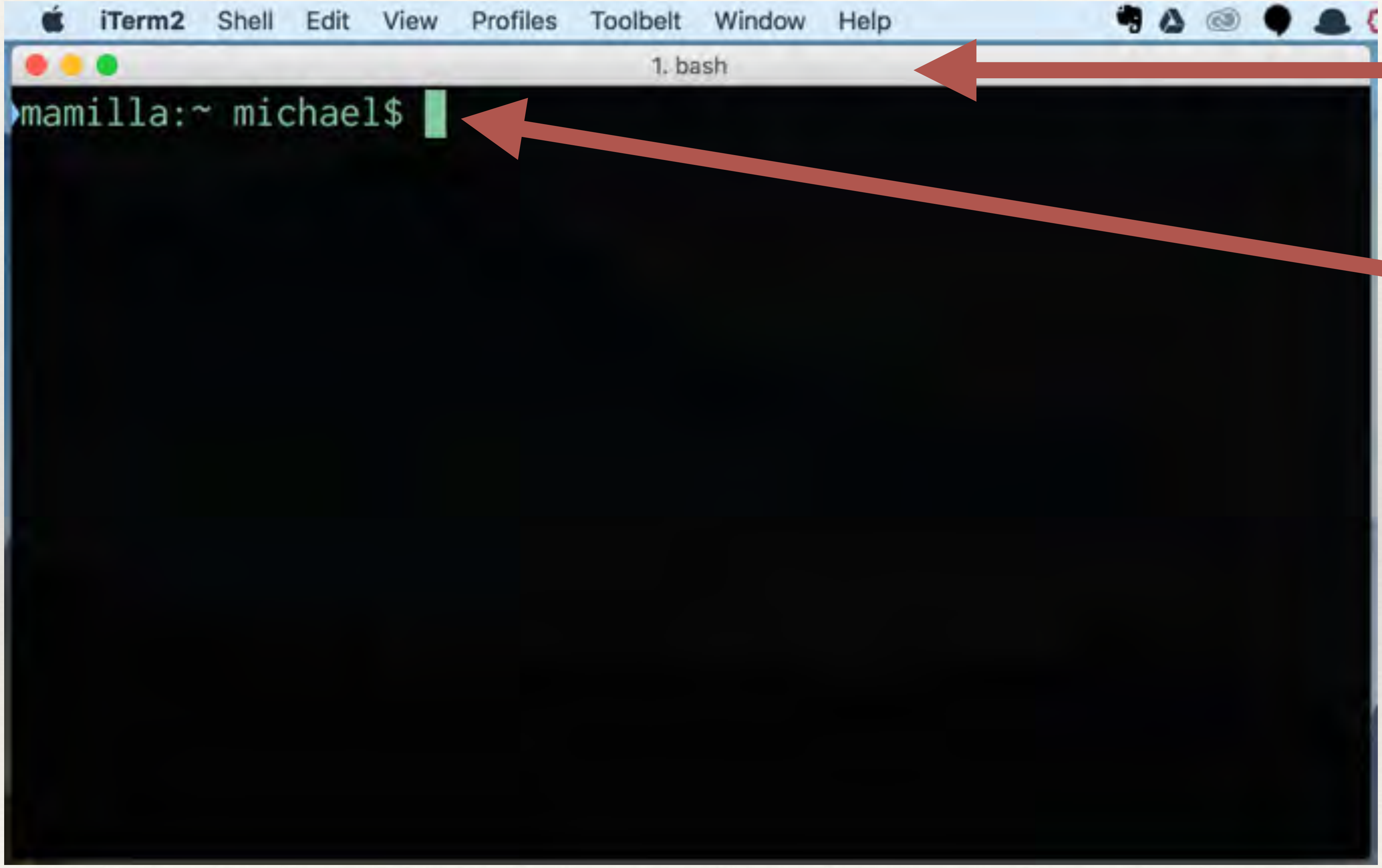
Python

Don't Use Python IDLE (it is not good)



```
Python 3.1 (r31:73578, Jun 27 2009, 21:49:46)
[GCC 4.0.1 (Apple Inc. build 5493)] on darwin
Type "copyright", "credits" or "license()" for more information.
>>> |
```

Unix Environment



```
mamilla:~ michael$ 
```

Terminal

Shell (Bash)

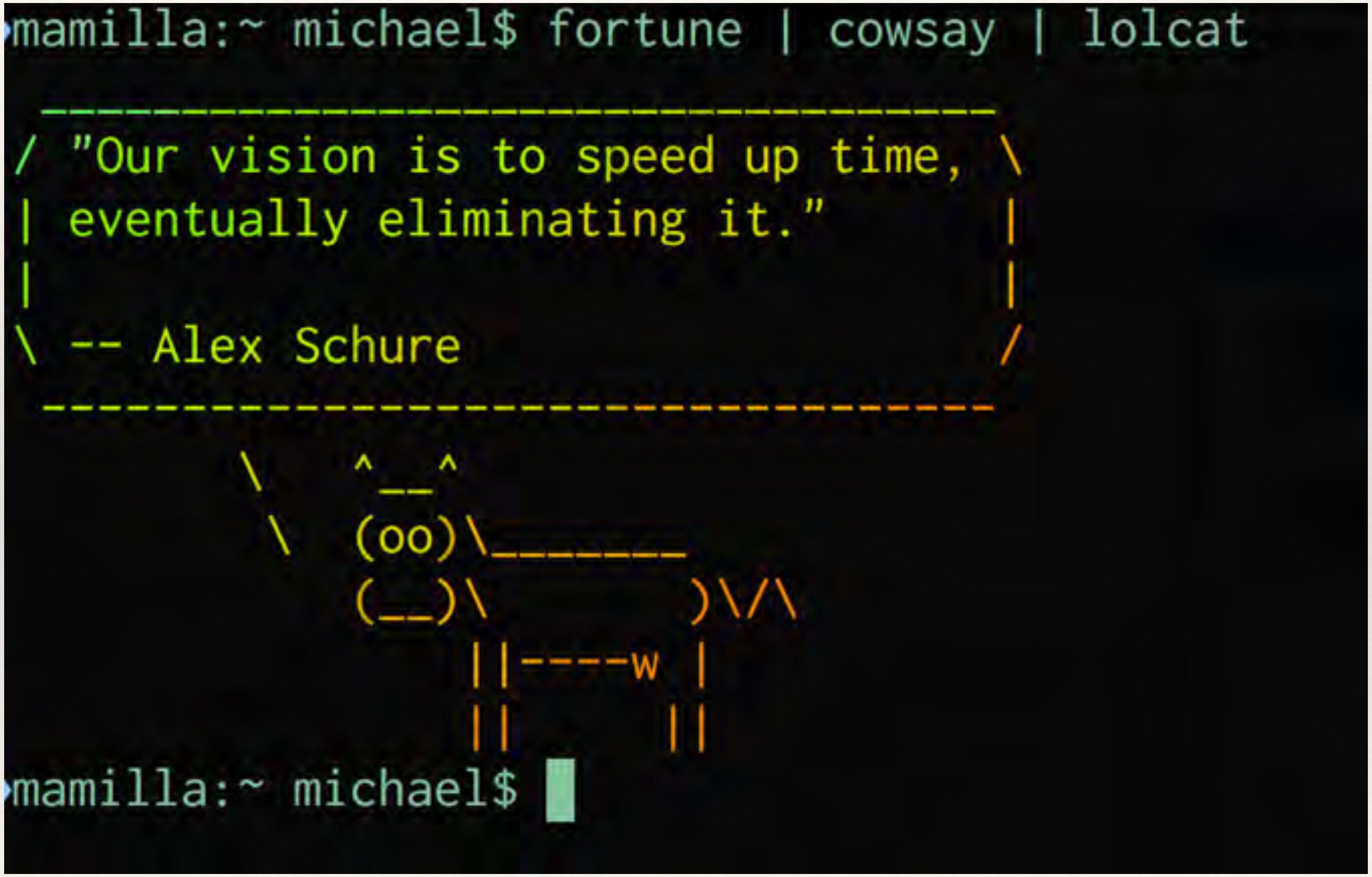
Some Must-Know Bash Commands

- ls -l -a
- find . -name “*.py”
- | [aka pipe]
- whereis
- cp
- rm
- mv

- cat
- touch
- mkdir
- > and >>
- less
- chmod
- vi [vim]

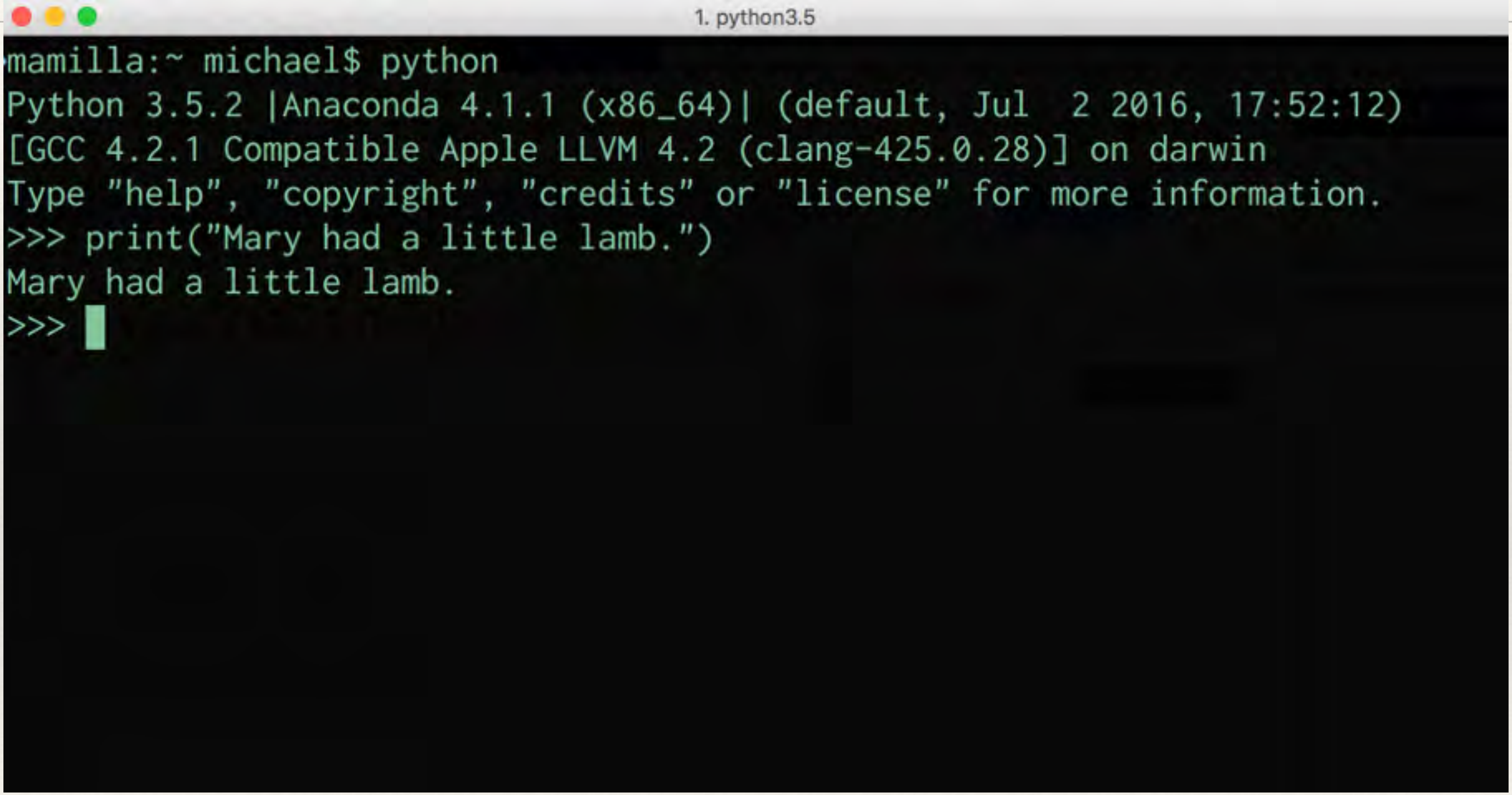
Totally Useless Bash Commands

- Cowsay
- locate
- fortune



```
mamilla:~ michael$ fortune | cowsay | lolcat
 __________________________________
/ "Our vision is to speed up time, \
| eventually eliminating it."      |
|                                  |
\ -- Alex Schure                   /
 ----------------------------------
        \   ^__^
         \  (oo)\_______
            (__)\       )\/\
                ||----w |
                ||     ||
mamilla:~ michael$
```

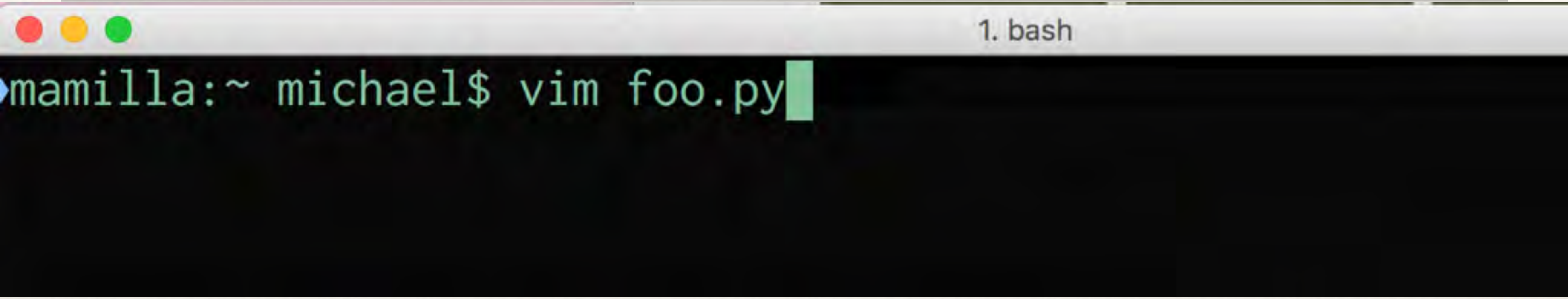
Python Shell (in Bash in Terminal)



```
mamilla:~ michael$ python
Python 3.5.2 |Anaconda 4.1.1 (x86_64)| (default, Jul  2 2016, 17:52:12)
[GCC 4.2.1 Compatible Apple LLVM 4.2 (clang-425.0.28)] on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>> print("Mary had a little lamb.")
Mary had a little lamb.
>>> 
```

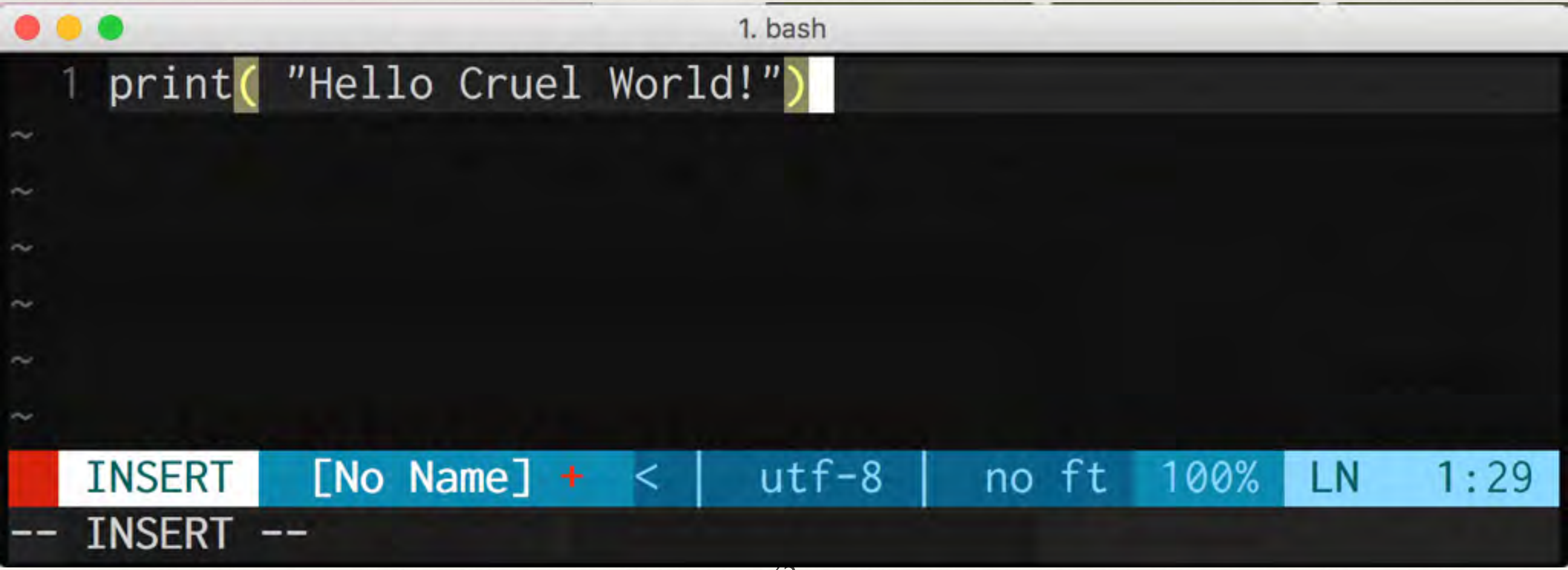
Not so useful. No history.

Create Edit Python File in vim (atom)



1. bash

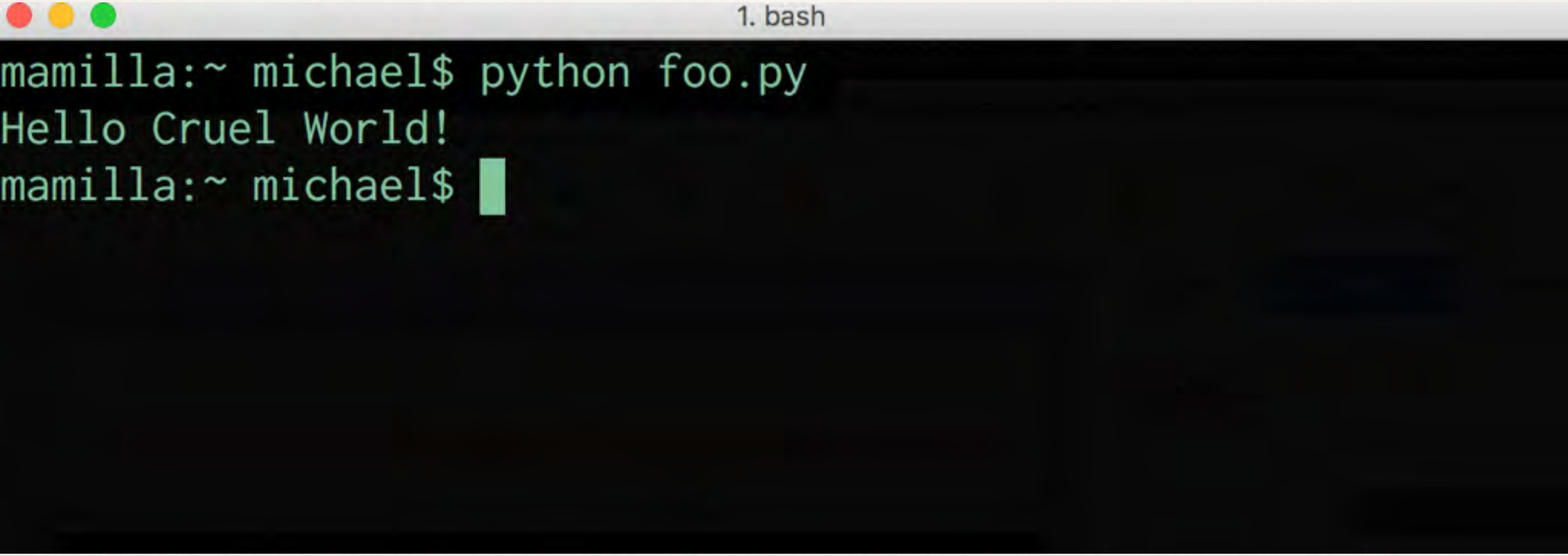
```
mamilla:~ michael$ vim foo.py
```



1. bash

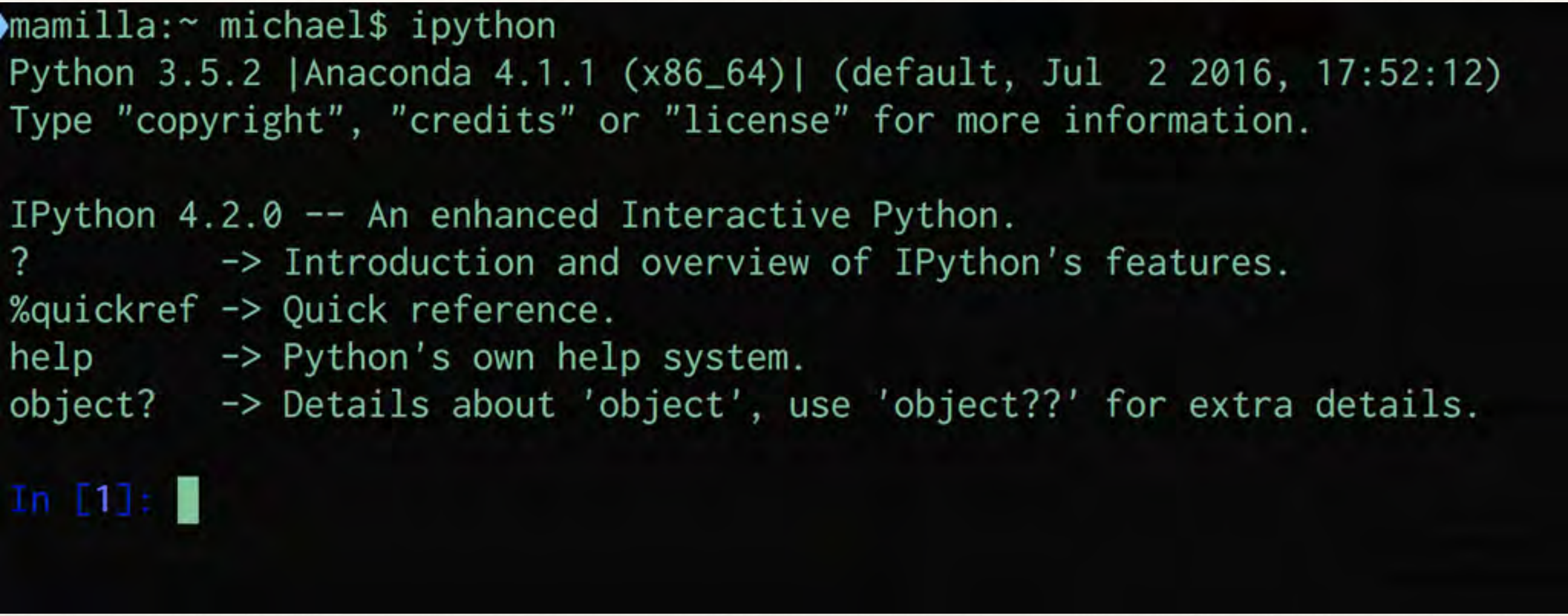
```
print( "Hello Cruel World!")
~
~
~
~
~
~
 INSERT   [No Name] + <  |  utf-8  |  no ft  100%  LN    1:29
-- INSERT --
```

run using “python”



```
mamilla:~ michael$ python foo.py
Hello Cruel World!
mamilla:~ michael$ 
```

Better Shell Ipython

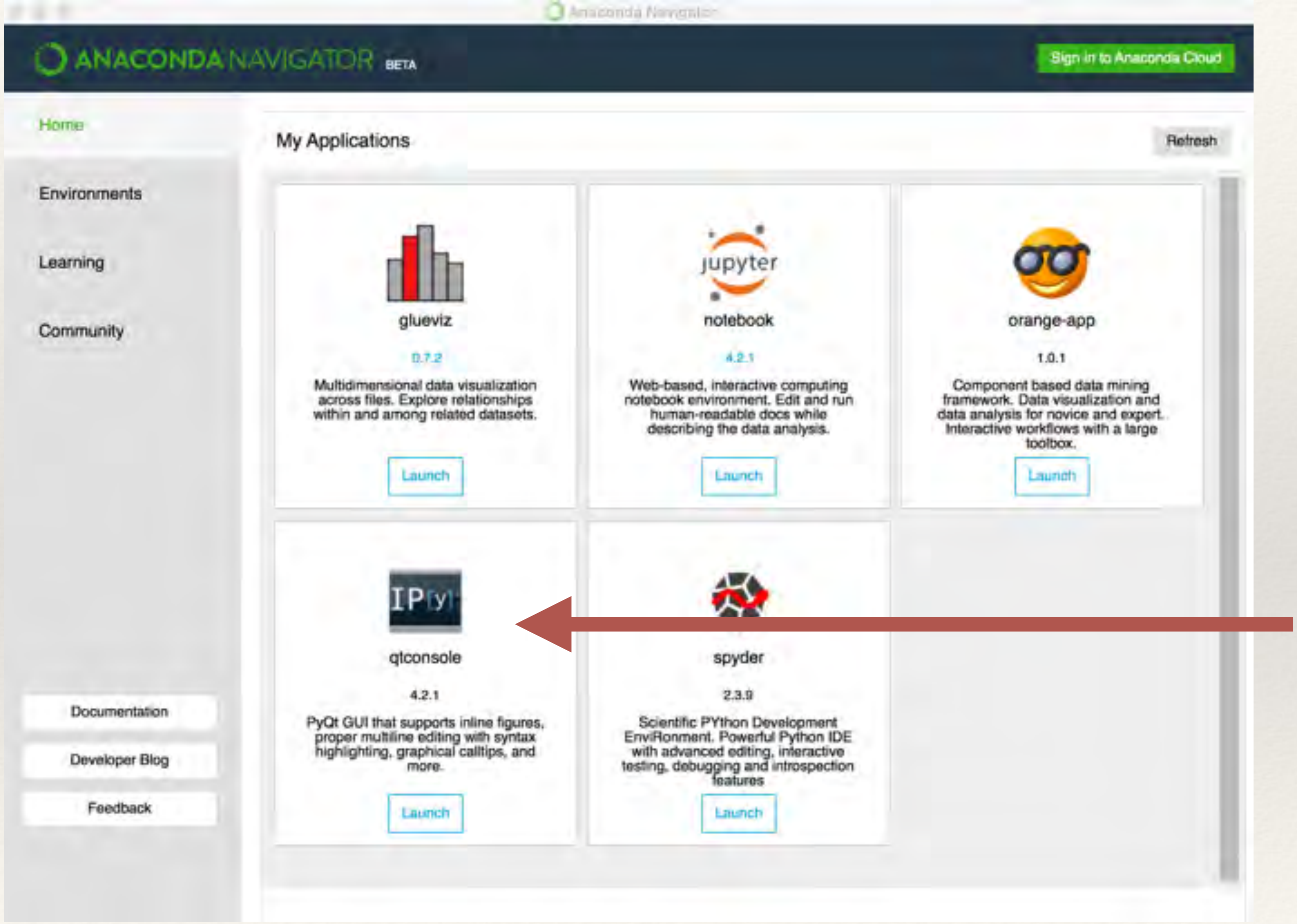


```
mamilla:~ michael$ ipython
Python 3.5.2 |Anaconda 4.1.1 (x86_64)| (default, Jul  2 2016, 17:52:12)
Type "copyright", "credits" or "license" for more information.

IPython 4.2.0 -- An enhanced Interactive Python.
?         -> Introduction and overview of IPython's features.
%quickref -> Quick reference.
help      -> Python's own help system.
object?   -> Details about 'object', use 'object??' for extra details.

In [1]:
```

Lots of features (including history)

Launch from terminal OR launch from anaconda

Ipython via Anaconda



Multidimensional data visualization across files. Explore relationships within and among related datasets.

Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.

Component based data mining framework. Data visualization and data analysis for novice and expert. Interactive workflows with a large toolbox.

PyQt GUI that supports inline figures, proper multiline editing with syntax highlighting, graphical calltips, and more.

Scientific PYthon Development EnviRonment. Powerful Python IDE with advanced editing, interactive testing, debugging and introspection features

Simple
Ipython
Shell

Basic Ipython

Access to shell:

- ls
- pwd
- !ls
- History:
  - [1] each command
  - control-p/-n

Help:

- print? [gets help]

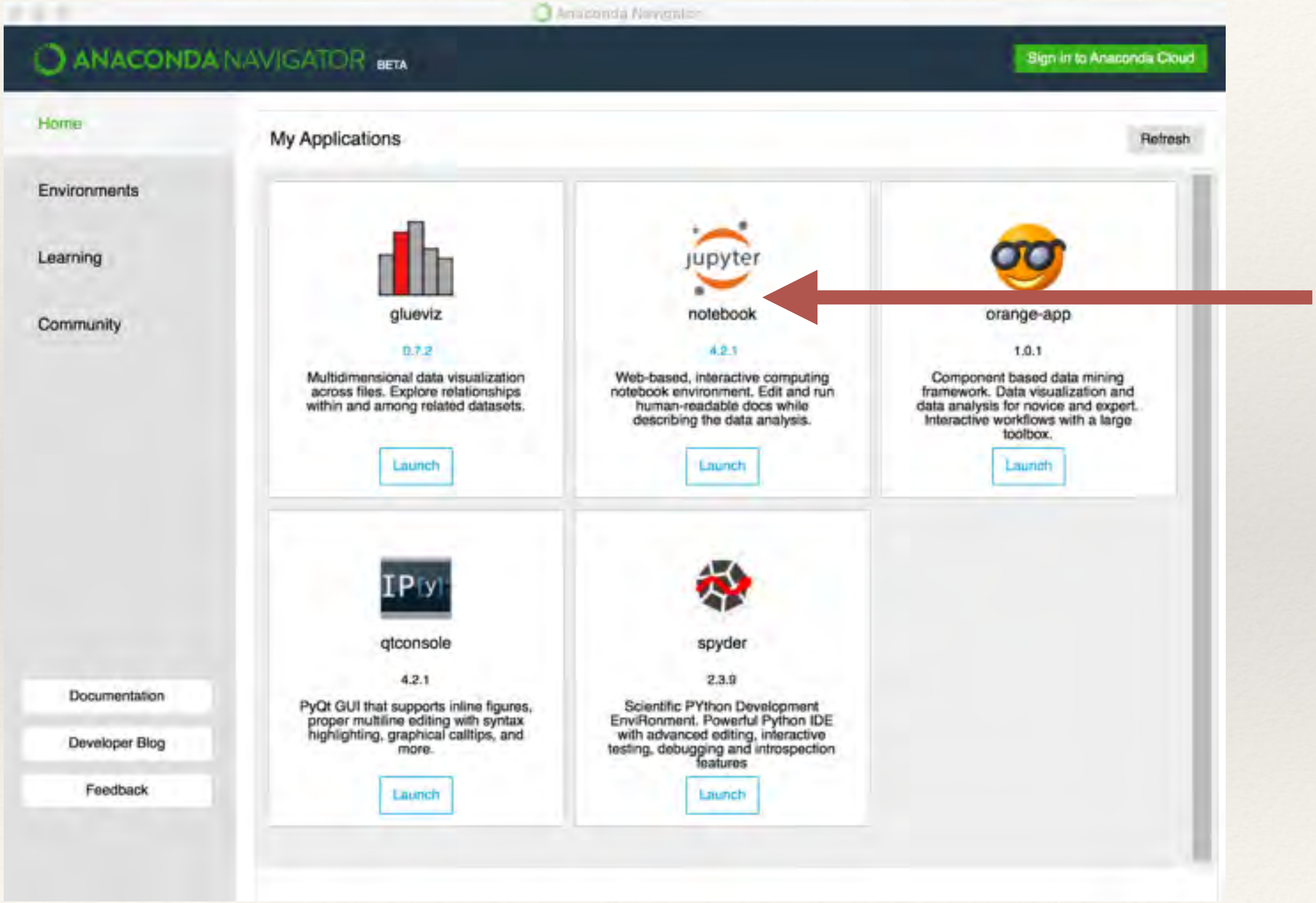
Magic:

- %lsmagic
- %automagic

Ipython

- Ipython a Python Shell (we'll talk about Jupyter later)

Jupyter Notebook: from anaconda



Multidimensional data visualization across files. Explore relationships within and among related datasets.

Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.

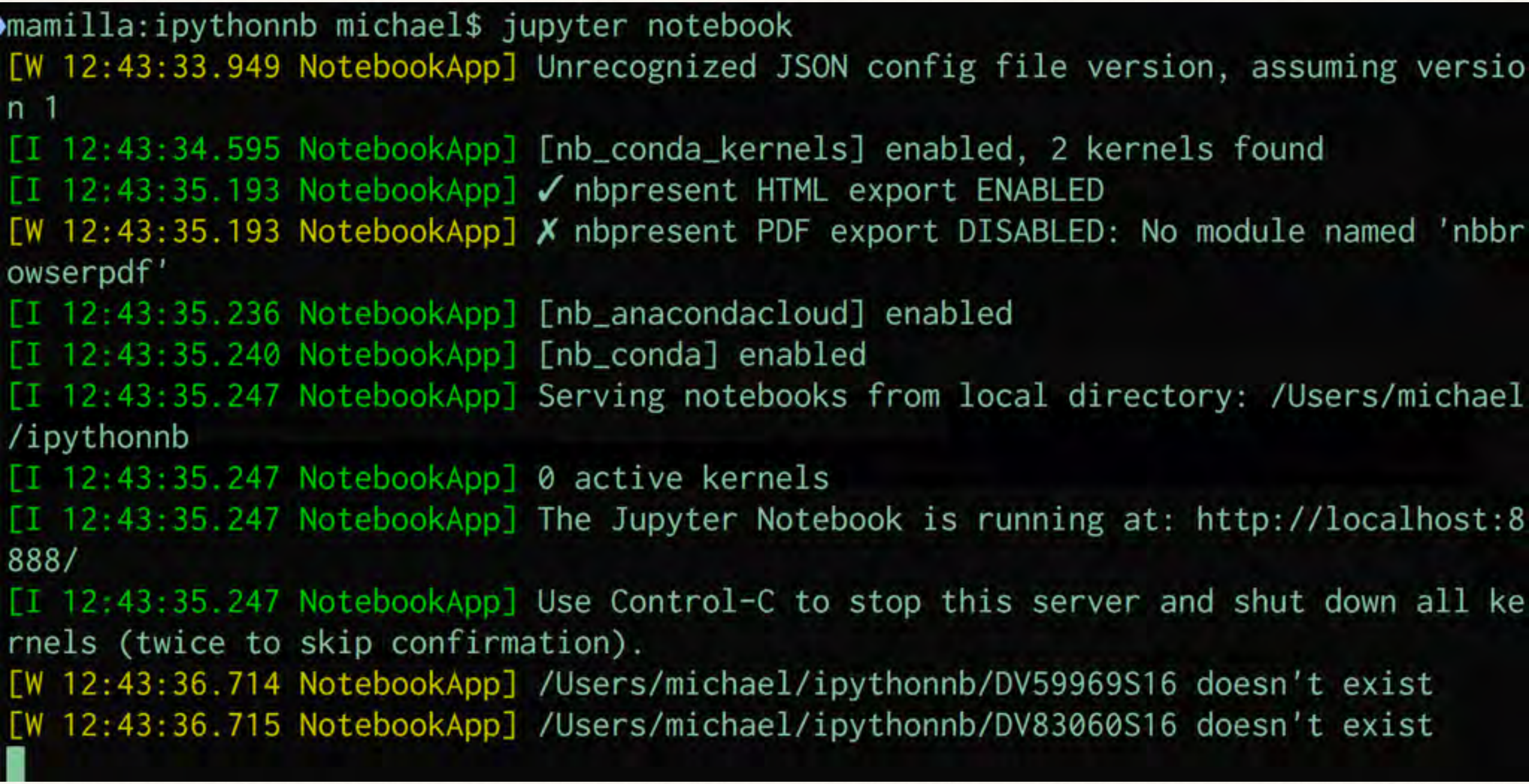
Component based data mining framework. Data visualization and data analysis for novice and expert. Interactive workflows with a large toolbox.

PyQt GUI that supports inline figures, proper multiline editing with syntax highlighting, graphical calltips, and more.

Scientific PYthon Development EnviRonment. Powerful Python IDE with advanced editing, interactive testing, debugging and introspection features

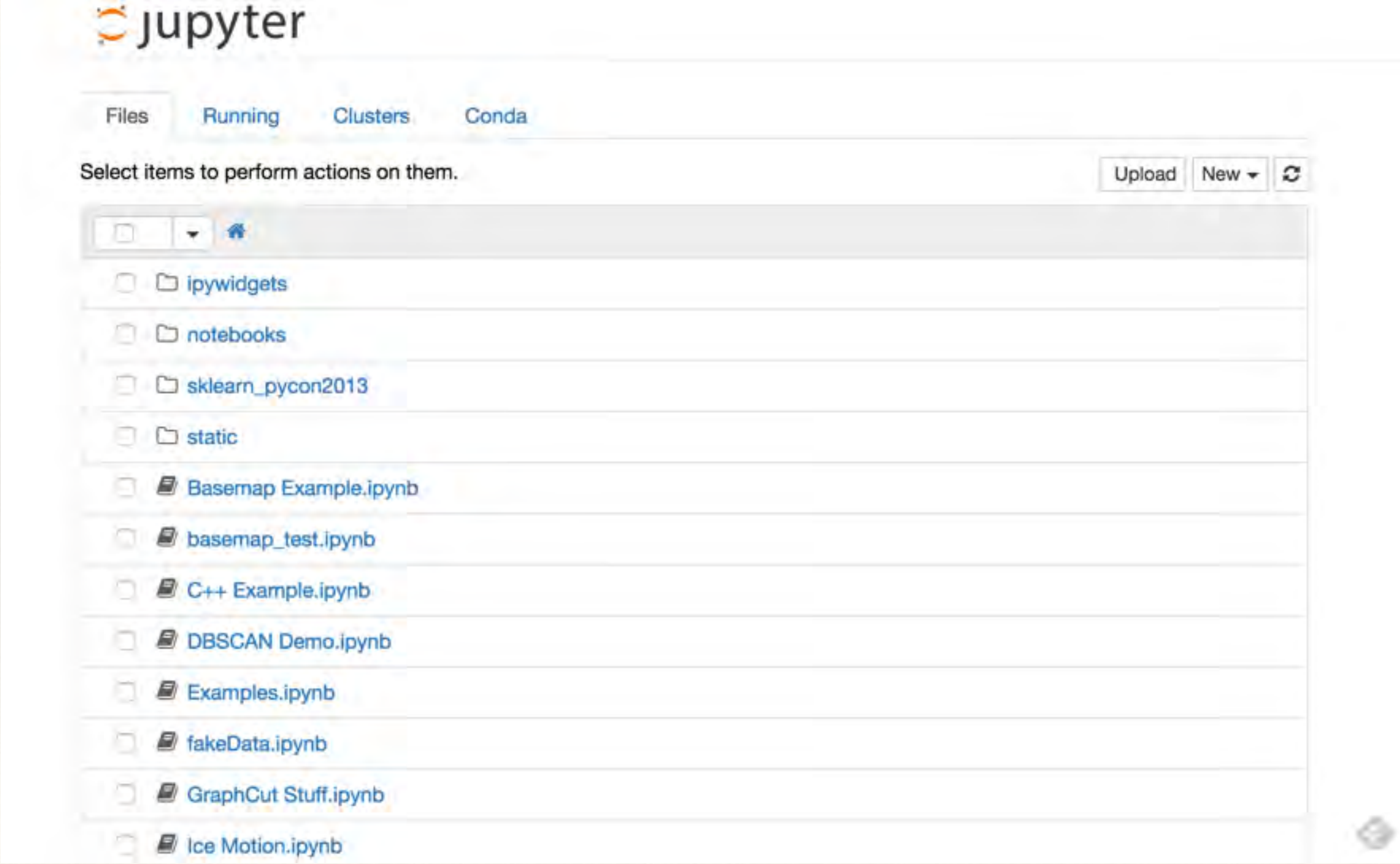
Run Notebook Server

Jupyter Notebook: from shell



```
mamilla:ipythonnb michael$ jupyter notebook
[W 12:43:33.949 NotebookApp] Unrecognized JSON config file version, assuming versio
n 1
[I 12:43:34.595 NotebookApp] [nb_conda_kernels] enabled, 2 kernels found
[I 12:43:35.193 NotebookApp] ✓ nbpresent HTML export ENABLED
[W 12:43:35.193 NotebookApp] ✗ nbpresent PDF export DISABLED: No module named 'nbbr
owserpdf'
[I 12:43:35.236 NotebookApp] [nb_anacondacloud] enabled
[I 12:43:35.240 NotebookApp] [nb_conda] enabled
[I 12:43:35.247 NotebookApp] Serving notebooks from local directory: /Users/michael
/ipythonnb
[I 12:43:35.247 NotebookApp] 0 active kernels
[I 12:43:35.247 NotebookApp] The Jupyter Notebook is running at: http://localhost:8
888/
[I 12:43:35.247 NotebookApp] Use Control-C to stop this server and shut down all ke
rnels (twice to skip confirmation).
[W 12:43:36.714 NotebookApp] /Users/michael/ipythonnb/DV59969S16 doesn't exist
[W 12:43:36.715 NotebookApp] /Users/michael/ipythonnb/DV83060S16 doesn't exist
```

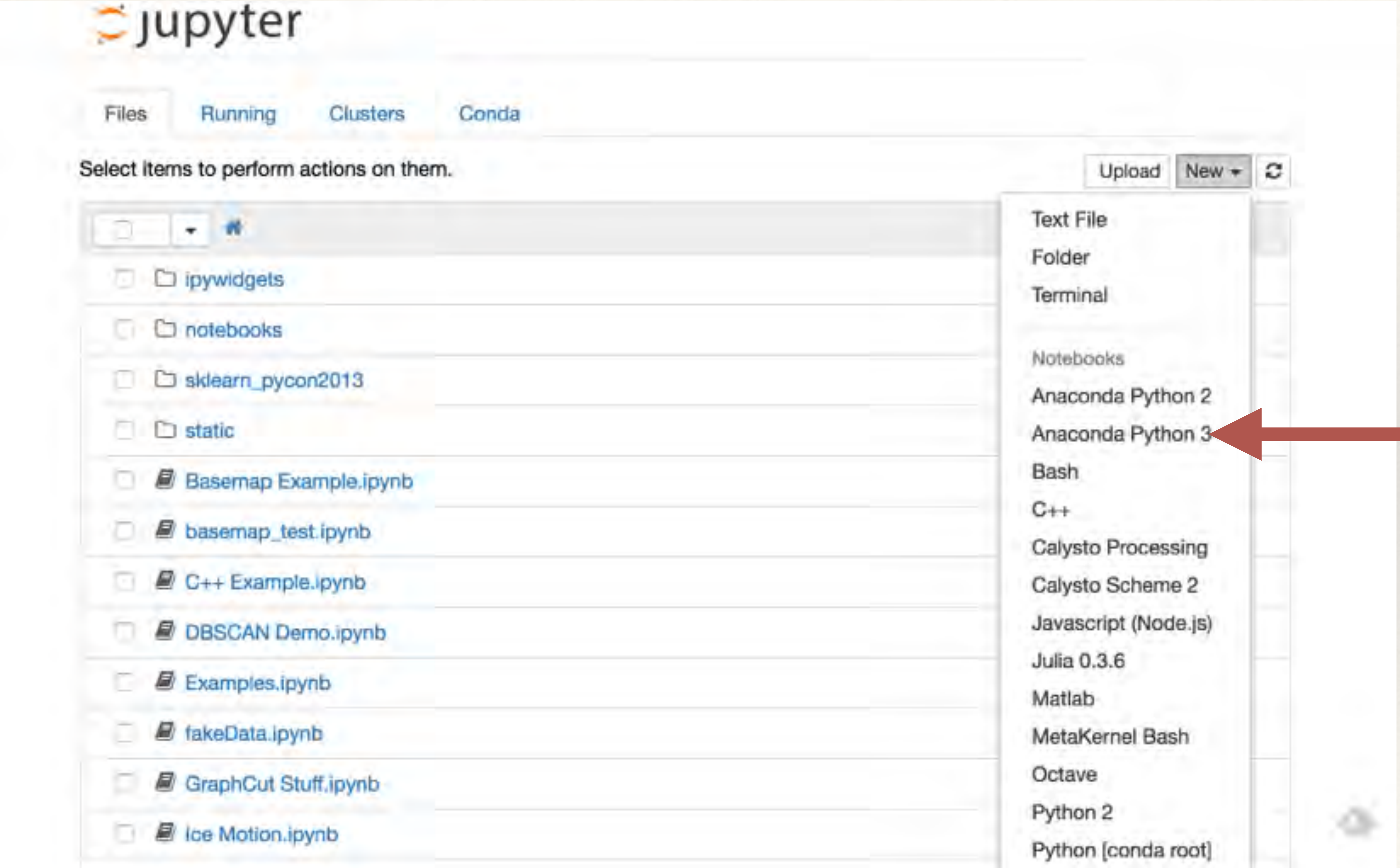
Jupyter Notebooks: Directory



Select items to perform actions on them.

- □ ipywidgets
- □ notebooks
- □ sklearn_pycon2013
- □ static
- □ Basemap Example.ipynb
- □ basemap_test.ipynb
- □ C++ Example.ipynb
- □ DBSCAN Demo.ipynb
- □ Examples.ipynb
- □ fakeData.ipynb
- □ GraphCut Stuff.ipynb
- □ Ice Motion.ipynb

Jupyter Notebooks: Launch

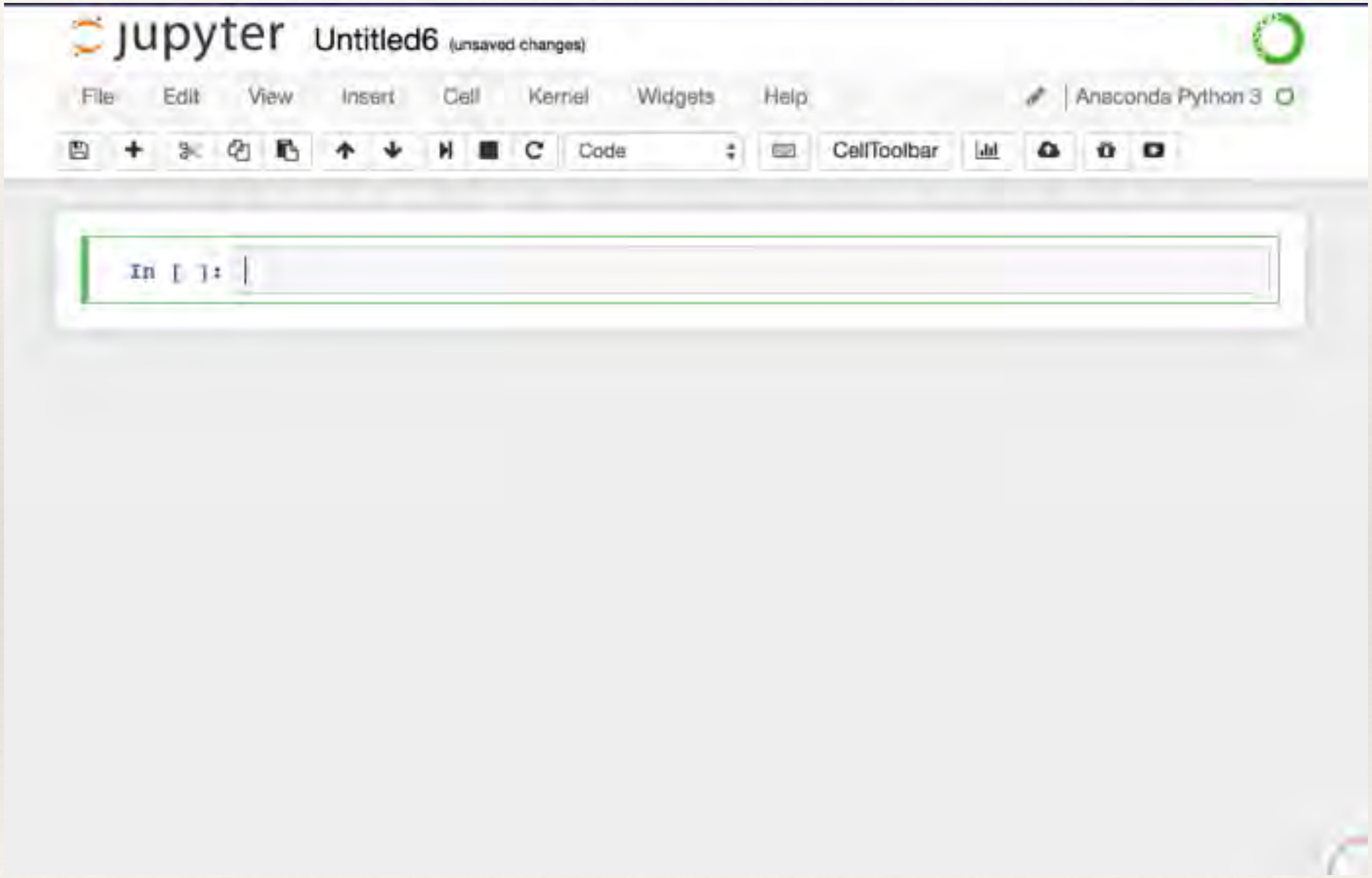


Select items to perform actions on them.

- ipywidgets
- notebooks
- sklearn_pycon2013
- static
- Basemap Example.ipynb
- basemap_test.ipynb
- C++ Example.ipynb
- DBSCAN Demo.ipynb
- Examples.ipynb
- fakeData.ipynb
- GraphCut Stuff.ipynb
- Ice Motion.ipynb

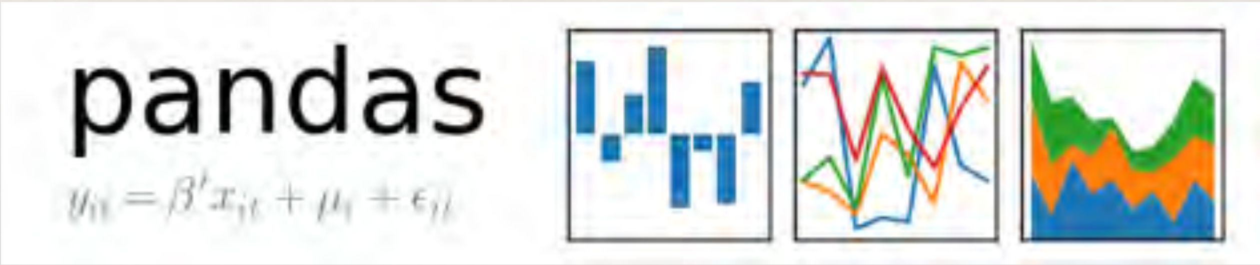
Start Ipython Notebook

Jupyter notebook with ipython kernel



Perfect for exploration experiments and tutorials

Basic DS Python Stack (know well)



Data Science From Scratch

- Chapter 2: Python Crash Course