



Evolved from the IPython Project

Machine Learning at Scale: R and Python Notebooks for Data Science via Jupyter

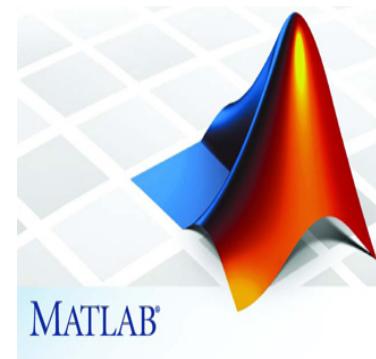
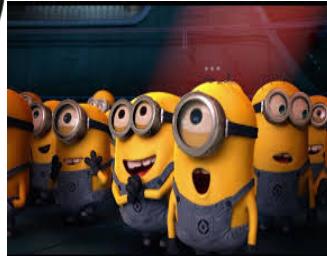
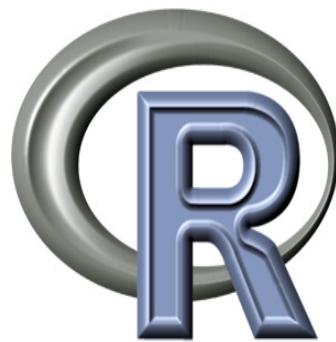


James G. Shanahan^{1,2}

¹*Church and Duncan Group Inc.*, ²*iSchool UC Berkeley, CA,*
EMAIL: James_DOT_Shanahan_AT_gmail_DOT_com

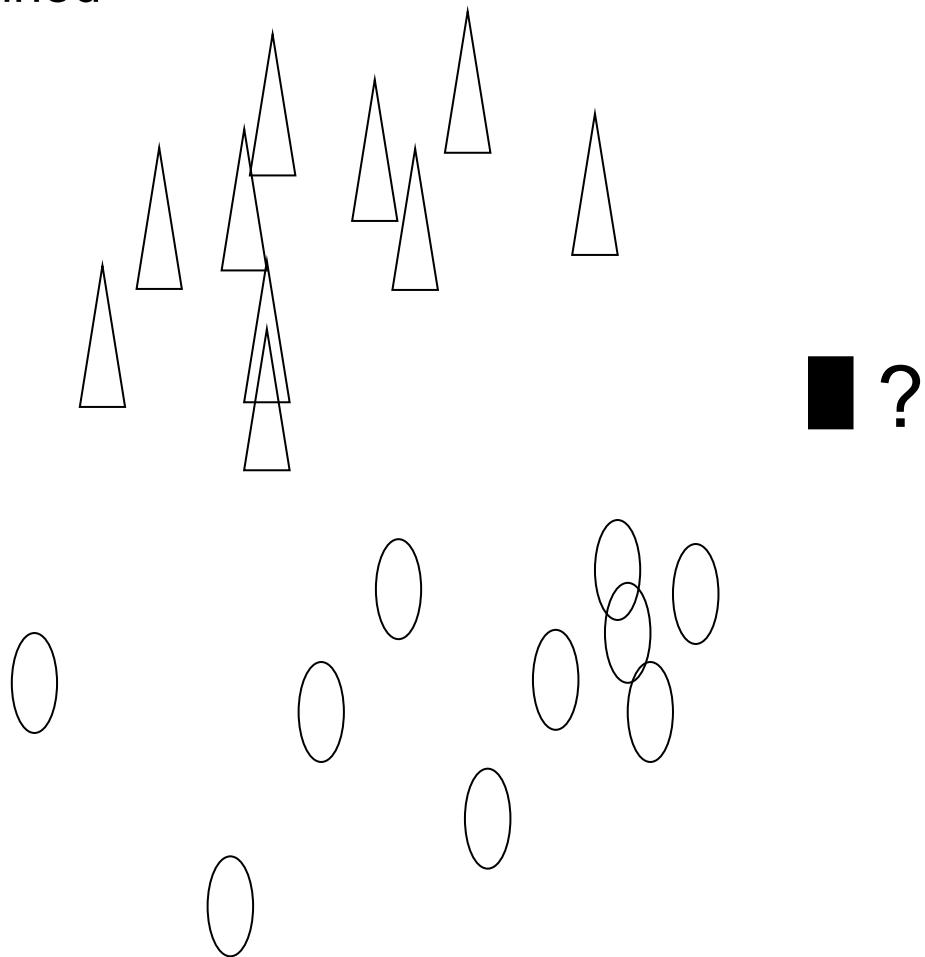
June 07, 2016

Python and R are trending for Data Science



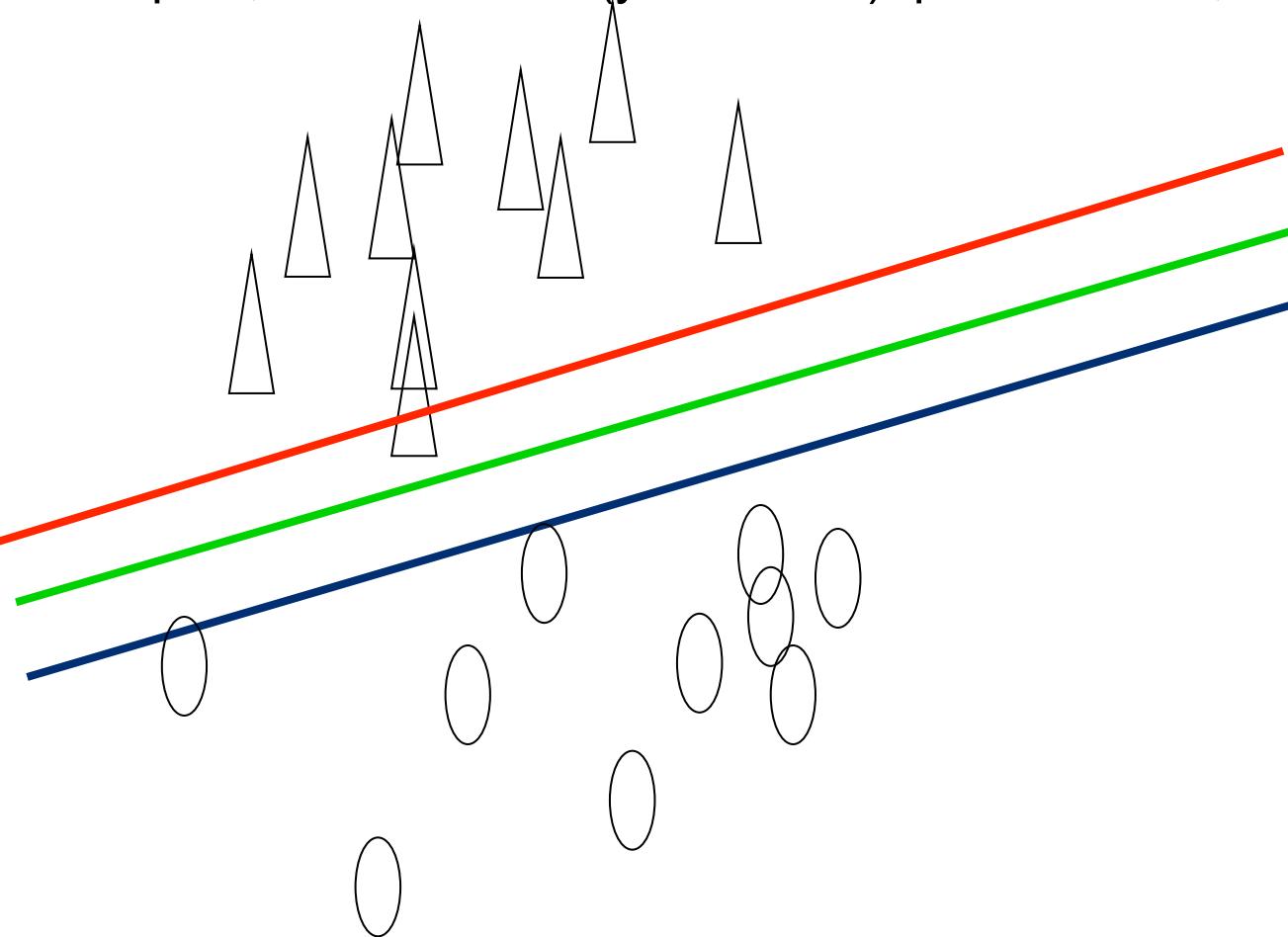
Example: Train a linear classifier

Problem explained



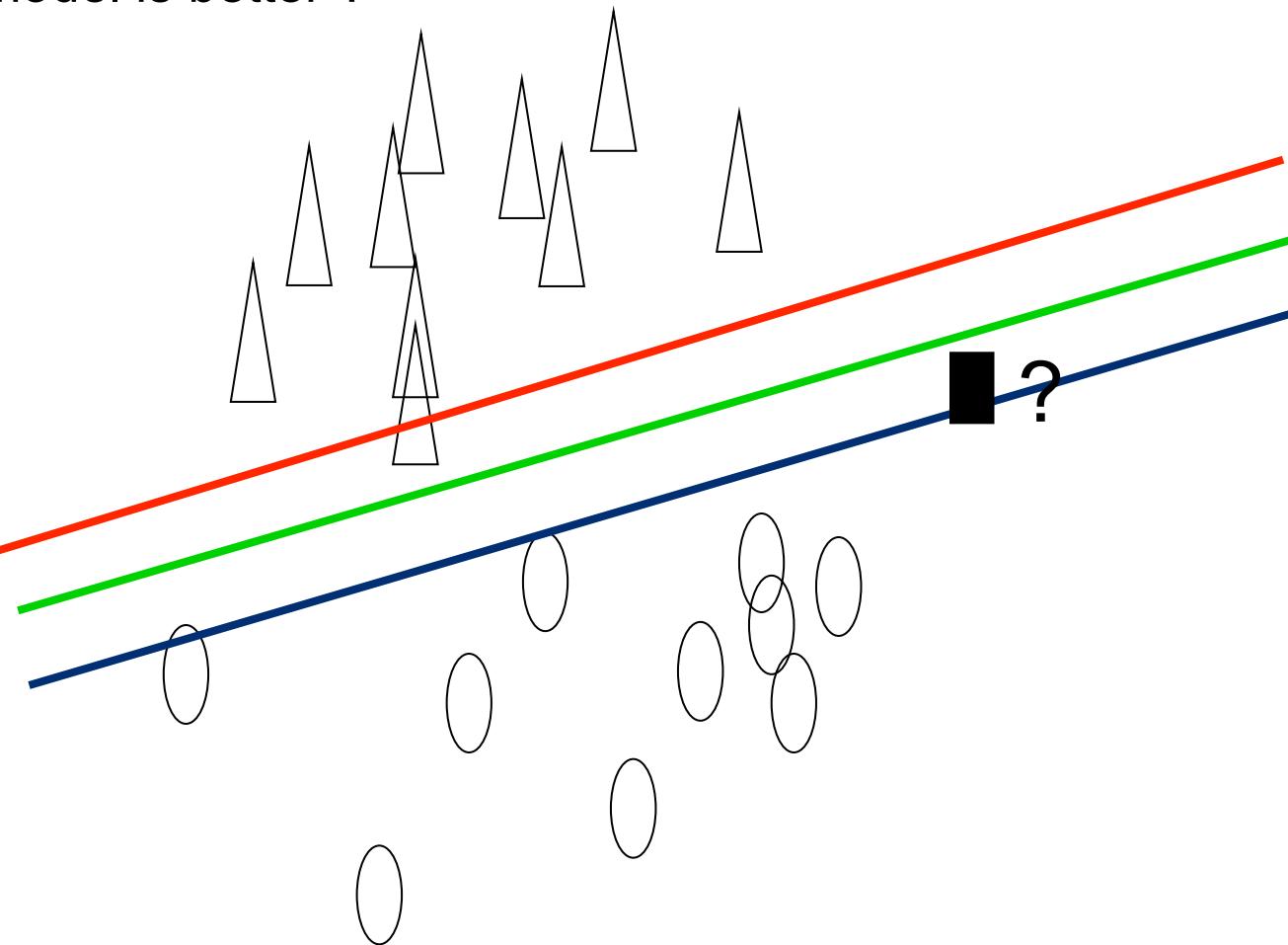
Example: Train a linear classifier

Based on samples, train a model ($y = x^*a + b$) parameter: a, b



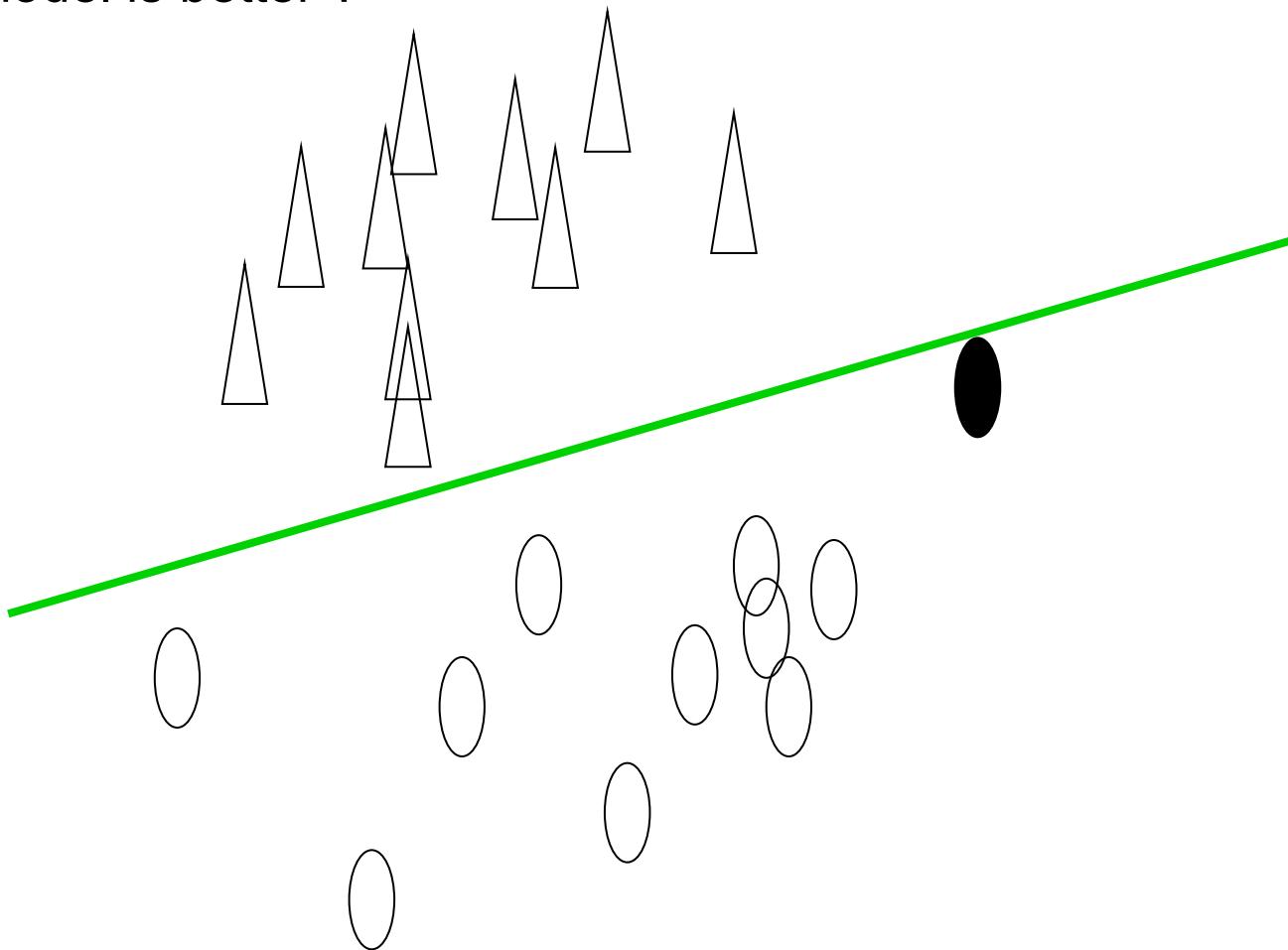
Example: Train an linear classifier

Which model is better ?

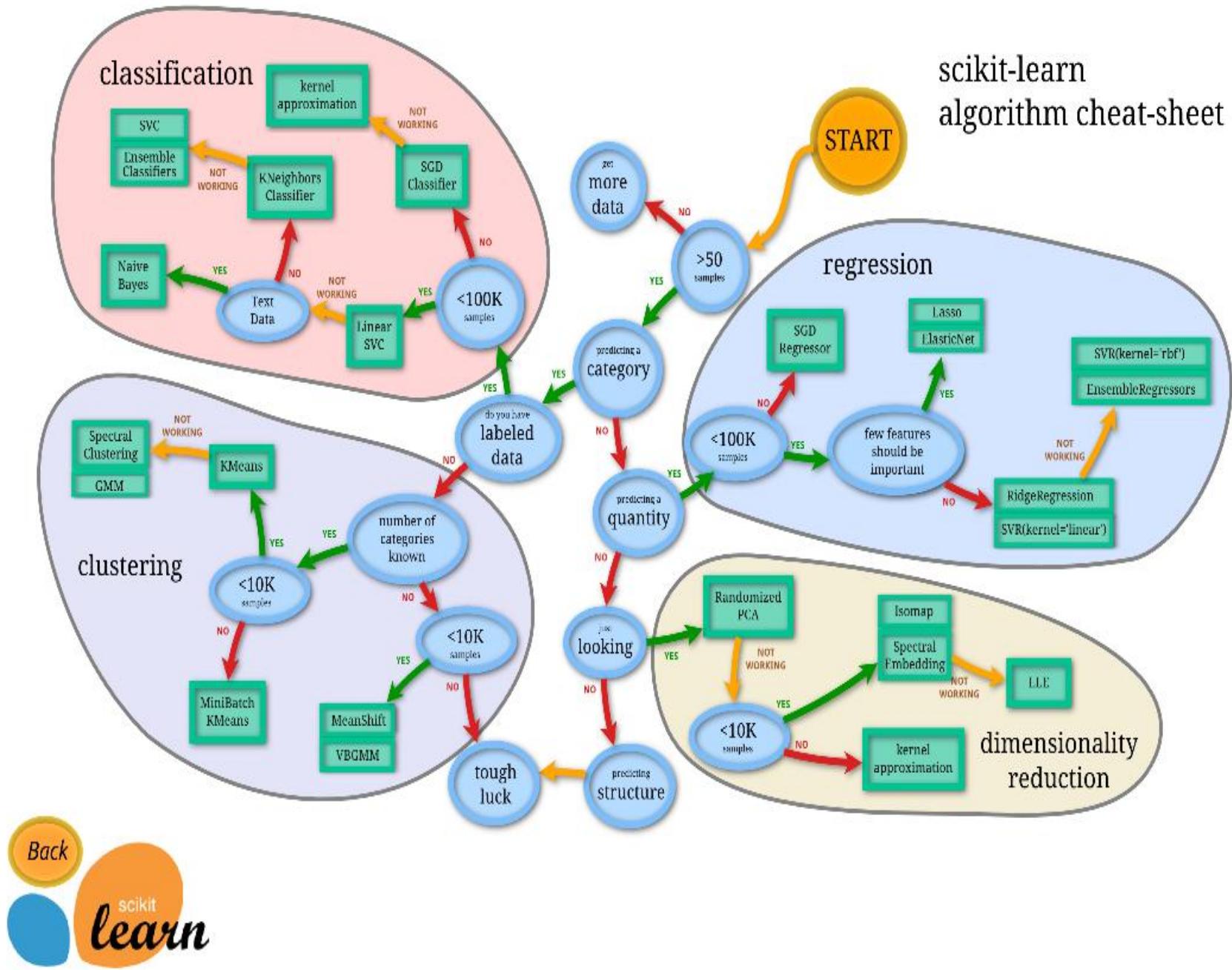


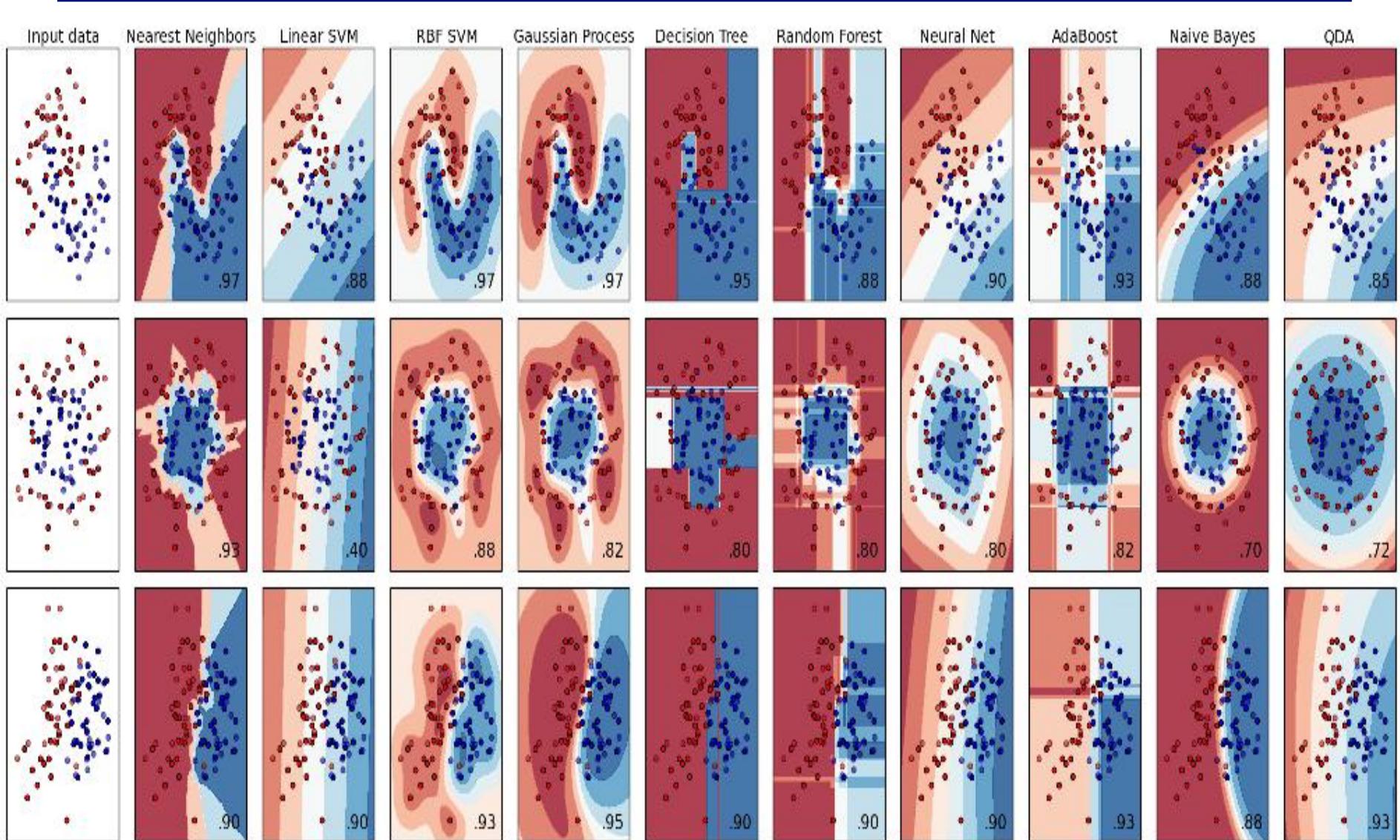
Example: Train a linear classifier

Which model is better ?



scikit-learn algorithm cheat-sheet





Jupyter's notebook enables R and Python notebooks

- **Jupyter's notebook interface enables mixing executable code with narrative text, equations, interactive visualizations**
 - Jupyter's notebook user interface enables mixing executable code with narrative text, equations, interactive visualizations, and images to enhance team collaboration and advance the state of reproducible research and training.
- **Jupyter began with Python and now has kernels for 50 different languages, including R**
- **The IRKernel is the native R kernel for Jupyter.**

PRP First Application: Distributed IPython/Jupyter Notebooks: Cross-Platform, Browser-Based Application Interleaves Code, Text, & Images

IJulia
IHaskell
IFSharp
IRuby
IGo
IScala
IMathics
Ialdor
LuaJIT/Torch
Lua Kernel

IRKernel (for the R language)

IErlang
IOCaml
IForth
IPerl
IPerl6
loctave
Calico Project

- kernels implemented in Mono,
including Java, IronPython, Boo, Logo,
BASIC, and many others

Python and R



Evolved from the IPython Project

IScilab
IMatlab
ICSharp
Bash
Clojure Kernel
Hy Kernel
Redis Kernel
jove, a kernel for io.js
IJavascript
Calysto Scheme
Calysto Processing
idl_kernel
Mochi Kernel
Lua (used in Splash)
Spark Kernel
Skulpt Python Kernel
MetaKernel Bash
MetaKernel Python
Brython Kernel
IVisual VPython Kernel

Source: John Graham, QI: 10/2015

Jupyter's notebook enables R and Python notebooks

- **To Install conda package manager to install and organize project dependencies**
 - Data scientists, researchers, and analysts use the conda package manager to install and organize project dependencies. With conda they can easily build and share metapackages, which are downloadable bundles of packages. Conda works with Linux, OS X, and Windows, and is language agnostic, so we can use it with any programming language and with projects that depend on multiple languages.
- **See slides below for installation instructions**

Jupyter Demo

The screenshot shows a Jupyter Notebook interface with the title "Jupyter EDA-JackPot-2016-06-09-Notebook". The notebook has a sidebar titled "Jackpot EDA Notebook" containing a table of contents and a link to share it in read-only mode. Two code cells are visible:

```
In [ ]: #!/usr/bin/Rscript
# source("/Users/jshanahan/Documents/workspace/HeritageHealthPrize/mainDriver.R")
#
# If you have any questions please contact
# Owen
# james.shanahan_AT_gmail.com
#
## Some code to setup data and do EDA
# 1. load the data into R and build some input features
# 2. Build some models
#
```

```
In [53]: library(dplyr)
#####
# STEP 20 Load data
# put into dplyr super efficient dataframe
loadData = function(config) {

  baseDir = config$baseDir
  nrows = config$nrows
  df <- read.csv(paste(baseDir, "Jan2016.csv", sep=","), sep = ",", header = TRUE,nrows = nrows)
  df2 <- read.csv(paste(baseDir, "Feb2016.csv", sep=","), sep = ",", header = TRUE,nrows = nrows)
  df3 <- read.csv(paste(baseDir, "Mar2016.csv", sep=","), sep = ",", header = TRUE,nrows = nrows)
  df4 <- read.csv(paste(baseDir, "Apr2016.csv", sep=","), sep = ",", header = TRUE,nrows = nrows)
  df5 <- read.csv(paste(baseDir, "May2016.csv", sep=","), sep = ",", header = TRUE,nrows = nrows)

  #combine csvs
  jpre results = rbind(df, df2, df3, df4, df5)

  #Convert dates and times so we can SORT etc..
  jpre results$startDate = as.Date(jpre results$startDate, format = "%m/%d/%Y")
  jpre results$endDate = as.Date(jpre results$endDate, format = "%m/%d/%Y")
  jpre results$signDate = as.Date(jpre results$signDate, format = "%m/%d/%Y")
  jpre results$startTime = as.POSIXct(jpre results$startTime, format = "%T")
  jpre results$endTime = as.POSIXct(jpre results$endTime, format = "%T")
  jpre results$signTime = as.POSIXct(jpre results$signTime, format = "%T")
```

INSTALL: R and Jupyter Notebooks

- **STEP 1: install Jupyter Notebook**
 - Install Anaconda (Python, Jupyter)
 - Follow slides included below
 - <http://jupyter.readthedocs.io/en/latest/install.html>
- **STEP 2: Install R Kernel**
 - Follow slides included below
 - <https://www.continuum.io/blog/developer/jupyter-and-conda-r>

INSTALL: R and Jupyter Notebooks

- **STEP 1: install Jupyter**
 - Install Anaconda (Python, Jupyter)
 - <http://jupyter.readthedocs.io/en/latest/install.html>
- **STEP 2: Install R Kernel**
 - <https://www.continuum.io/blog/developer/jupyter-and-conda-r>

Anaconda

CONTINUUM[®]
ANALYTICS

ANACONDA | COMMUNITY | SERVICES | SOLUTIONS | ABOUT | RESOURCES

LOG IN SUPPORT CONTACT



**ANACONDA GIVES
SUPERPOWERS TO
PEOPLE WHO CHANGE
THE WORLD**

Modern open source analytics platform powered by Python

DOWNLOAD FOR FREE

ANACONDA NOW AVAILABLE FOR CLOUDERA CDH

WHY YOU'LL LOVE ANACONDA

Making it easy to install, intuitive to discover, quick to analyze, simple to collaborate, and accessible to all.

**Committed to Open
Source. Now and
forever.**

**Tested and certified
packages to cover
your back.**

**Explore and visualize
complex data easily.**

**All the analytics you
ever wanted and
more.**

ANACONDA®

Application	Jupyter/IPython Notebook			Anaconda	
Analytics	pandas, NumPy, SciPy, Numba, NLTK, scikit-learn, scikit-image, and more from Anaconda ...				
Parallel Computation	Dask	Spark	Hive / Impala		
Data and Resource Management	HDFS, YARN, SGE, Slurm or other distributed systems				
Server	Bare-metal or Cloud-based Cluster				

Download Anaconda

CONTINUUM[®]
ANALYTICS

LOG IN SUPPORT CONTACT

ANACONDA | COMMUNITY | SERVICES | SOLUTIONS | ABOUT | RESOURCES

ANACONDA ▶ DOWNLOAD

DOWNLOAD ANACONDA NOW!

Jump to: [Windows](#) | [OSX](#) | [Linux](#)

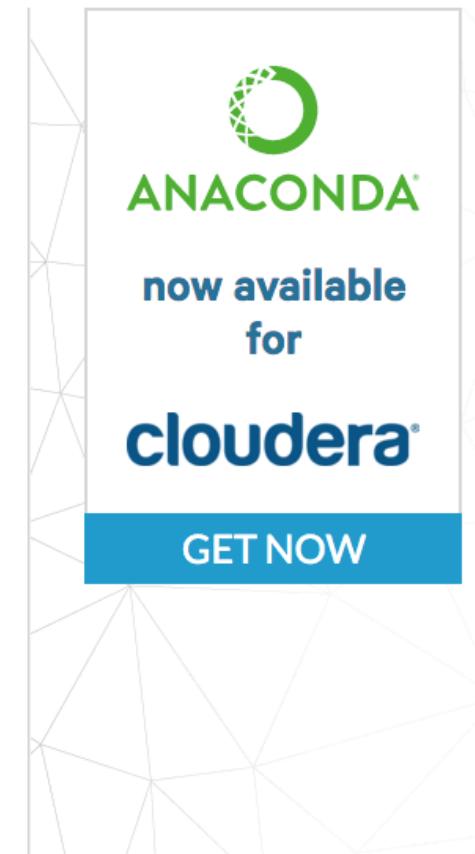
Get Superpowers with Anaconda

Anaconda is a completely free Python distribution (including for commercial use and redistribution). It includes more than 400 of the most popular Python packages for science, math, engineering, and data analysis. See the packages included with Anaconda and [the Anaconda changelog](#).

Which version should I download and install?

Because Anaconda includes installers for Python 2.7 and 3.5, either is fine. Using either version, you can use Python 3.4 with the conda command. You can create a 3.5 environment with the conda command if you've downloaded 2.7 – and vice versa.

If you don't have time or disk space for the entire distribution, try [Miniconda](#), which contains only conda and Python. Then [install](#) just the individual packages you want through the conda command.



Anaconda for Windows

PYTHON 2.7	PYTHON 3.5
WINDOWS 64-BIT GRAPHICAL INSTALLER 335M	WINDOWS 64-BIT GRAPHICAL INSTALLER 345M
Windows 32-bit Graphical Installer 281M	Windows 32-bit Graphical Installer 283M
Behind a firewall? Use these zipped Windows installers .	

Windows

Windows Anaconda Installation

1. Download the graphical installer.
2. Double-click the .exe file to install Anaconda and follow the instructions on the screen.
3. Optional: **Verify data integrity with MD5**.

Anaconda for OS X

PYTHON 2.7	PYTHON 3.5
MAC OS X 64-BIT GRAPHICAL INSTALLER 339M (OS X 10.7 or higher)	MAC OS X 64-BIT GRAPHICAL INSTALLER 342M (OS X 10.7 or higher)
Mac OS X 64-bit Command-Line installer 290M (OS X 10.7 or higher)	Mac OS X 64-bit Command-Line installer 293M (OS X 10.7 or higher)

OS X

OS X Anaconda Installation

Choose either the graphical installer or the command line installer for OS X.

Graphical Installer:

1. Download the graphical installer.
2. Double-click the downloaded .pkg file and follow the instructions.

Large

Anaconda Install

Download python+Notebook

Contents

- Anaconda Install
 - OS X Install
 - OS X Uninstall
 - Linux Install
 - Linux Uninstall
 - Windows Install
 - Windows Uninstall
 - Updating from older Anaconda versions
 - What's next?

<http://docs.continuum.io/anaconda/install>

OS X Install

Download the Anaconda installer and double click it.

Download the [Anaconda installer](#) and double click it.

NOTE: You may see a screen that says "You cannot install Anaconda in this location. The Anaconda

300 Meg: Takes 5-10 minutes via a hotel WiFi



Anaconda Installer

engineering, data analysis.



The screenshot shows the 'Welcome to the Anaconda Installer' screen. At the top right is the Anaconda logo. Below it, the text reads: 'Anaconda is a Python distribution for Scientific, Engineering, and Business Intelligence Data Management.' To the left is a large green circular graphic with a white 'yin-yang' style symbol inside. On the right side of the screen are two buttons: 'Go Back' and 'Continue'. In the bottom right corner of the slide, there is a yellow box containing the text: 'Based Environment for Data Analysis on your Servers'.

CHOOSE YOUR INSTALLER:

- Mac OS X – 64-Bit Python 2.7 Graphical Installer
Size: 275M (OS X 10.7 or higher)
- OTHER DOWNLOADS
- Mac OS X – 64-Bit Python 2.7 Command-Line installer
Size: 241M (OS X 10.7 or higher)

INSTALLATION
After download the .pkg file screen.
COMMAND

Introduction

- Read Me
- License
- Destination Select
- Installation Type
- Installation
- Summary

Takes 5-10 minutes via a hotel WiFi

```
conda update --prefix /Users/jshanahan/anaconda anaconda
```

Download Anaconda Python 2.7

Anaconda for OS X

PYTHON 2.7	PYTHON 3.5
<p>Mac OS X 64-bit Graphical Installer</p> <p>274M (OS X 10.7 or higher)</p>	<p>Mac OS X 64-bit Graphical Installer</p> <p>267M (OS X 10.7 or higher)</p>
<p>Mac OS X 64-bit Command-Line installer</p> <p>239M (OS X 10.7 or higher)</p>	<p>Mac OS X 64-bit Command-Line installer</p> <p>233M (OS X 10.7 or higher)</p>

OS X Anaconda Installation

Choose either the graphical installer or the command line installer for OS X.

Graphical Installer:

1. Download the graphical installer.
2. Double-click the downloaded .pkg file and follow the instructions.

OS X Anaconda Installation

OS X Anaconda Installation

Choose either the graphical installer or the command line installer for OSX.

Graphical Installer:

1. Download the graphical installer.
2. Double-click the downloaded .pkg file and follow the instructions.

Command Line Installer:

1. Download the command line installer.
2. In your terminal window, type one of the below and follow the instructions:

Python 2.7:

```
bash Anaconda2-2.5.0-MacOSX-x86_64.sh
```

Python 3.5:

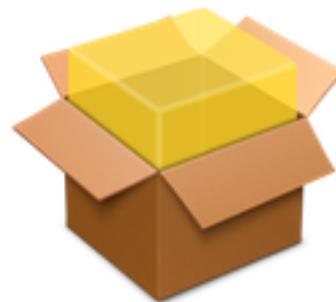
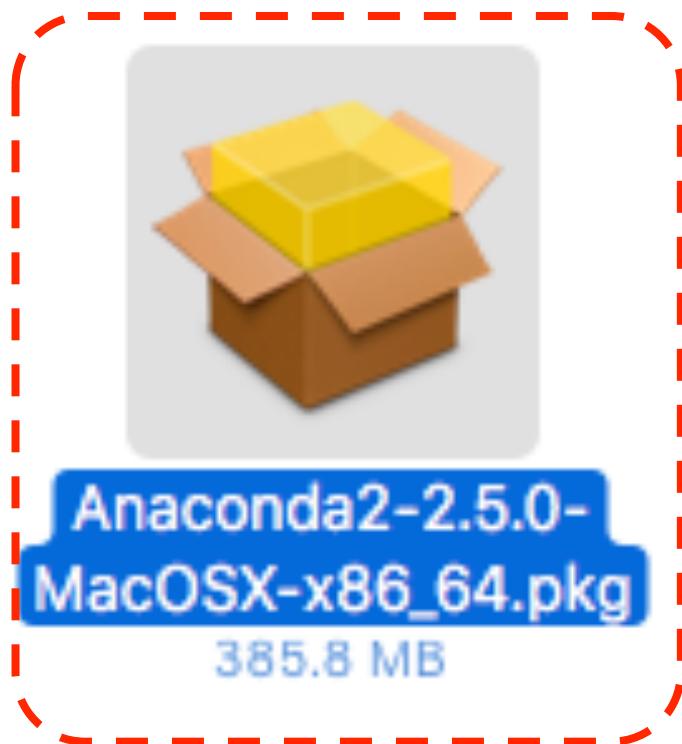
```
bash Anaconda3-2.5.0-MacOSX-x86_64.sh
```

NOTE: Include the "bash" command even if you are not using the bash shell.

3. Optional: Verify data integrity with MD5.

OS X Anaconda Installation

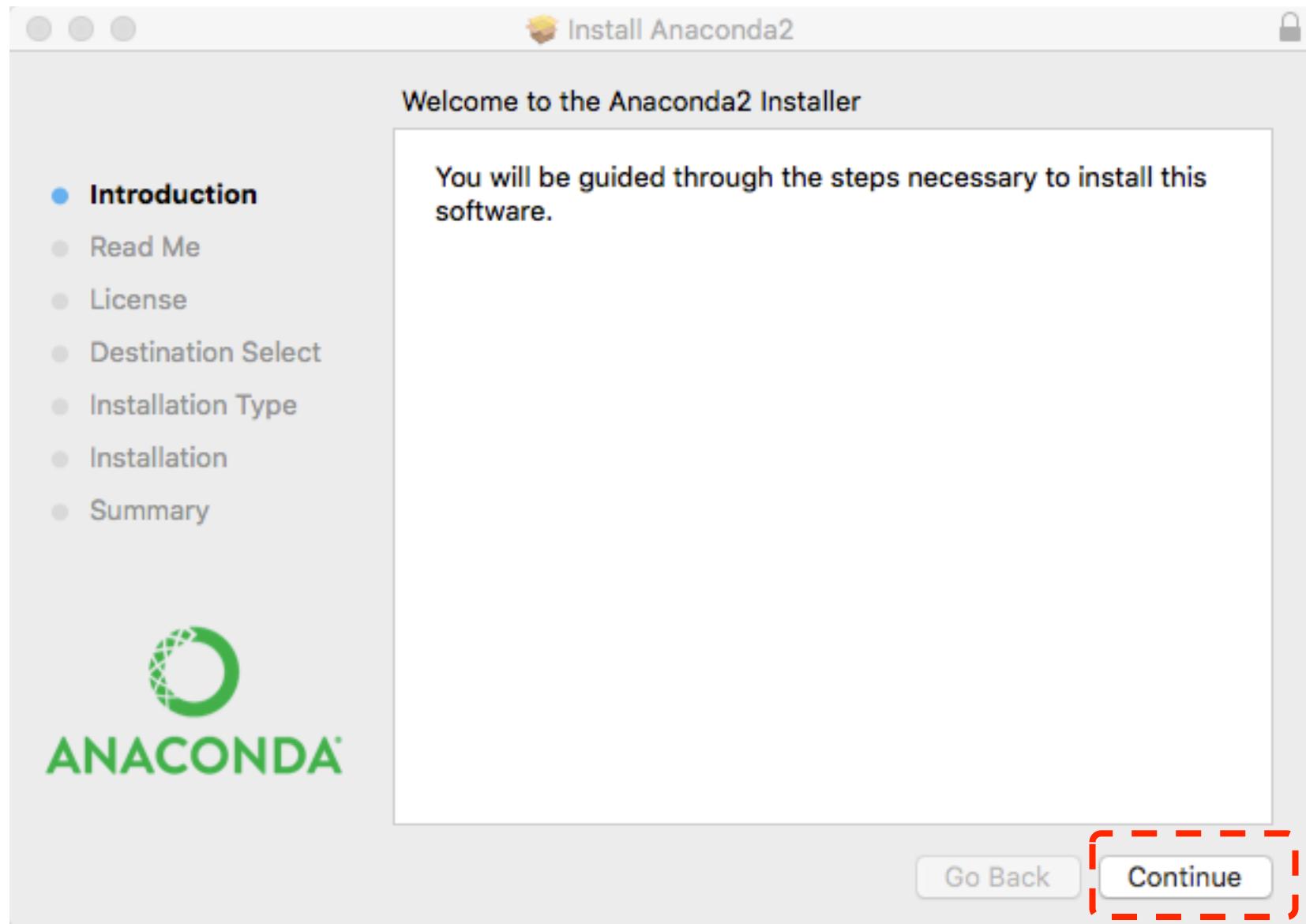
Anaconda2-2.5.0-MacOSX-x86_64.pkg



Anaconda2-2.5.0-MacOSX-x86_64.pkg

Installer package - 385.8 MB

OS X Anaconda Installation



OS X Anaconda Installation

The screenshot shows the 'Install Anaconda2' window on OS X. The title bar has a yellow icon and the text 'Install Anaconda2'. On the left is a sidebar with the following options:

- Introduction
- Read Me**
- License
- Destination Select
- Installation Type
- Installation
- Summary

The main content area is titled 'Important Information' and contains the following text:

Anaconda is a modern open source analytics platform powered by Python. See <https://www.continuum.io/downloads/>.

By default, this installer modifies your bash profile to put Anaconda in your PATH. To disable this, choose "Customize" at the "Installation Type" phase, and disable the "Modify PATH" option. If you do not do this, you will need to add ~/anaconda/bin to your PATH manually to run the commands, or run all anaconda commands explicitly from that path.

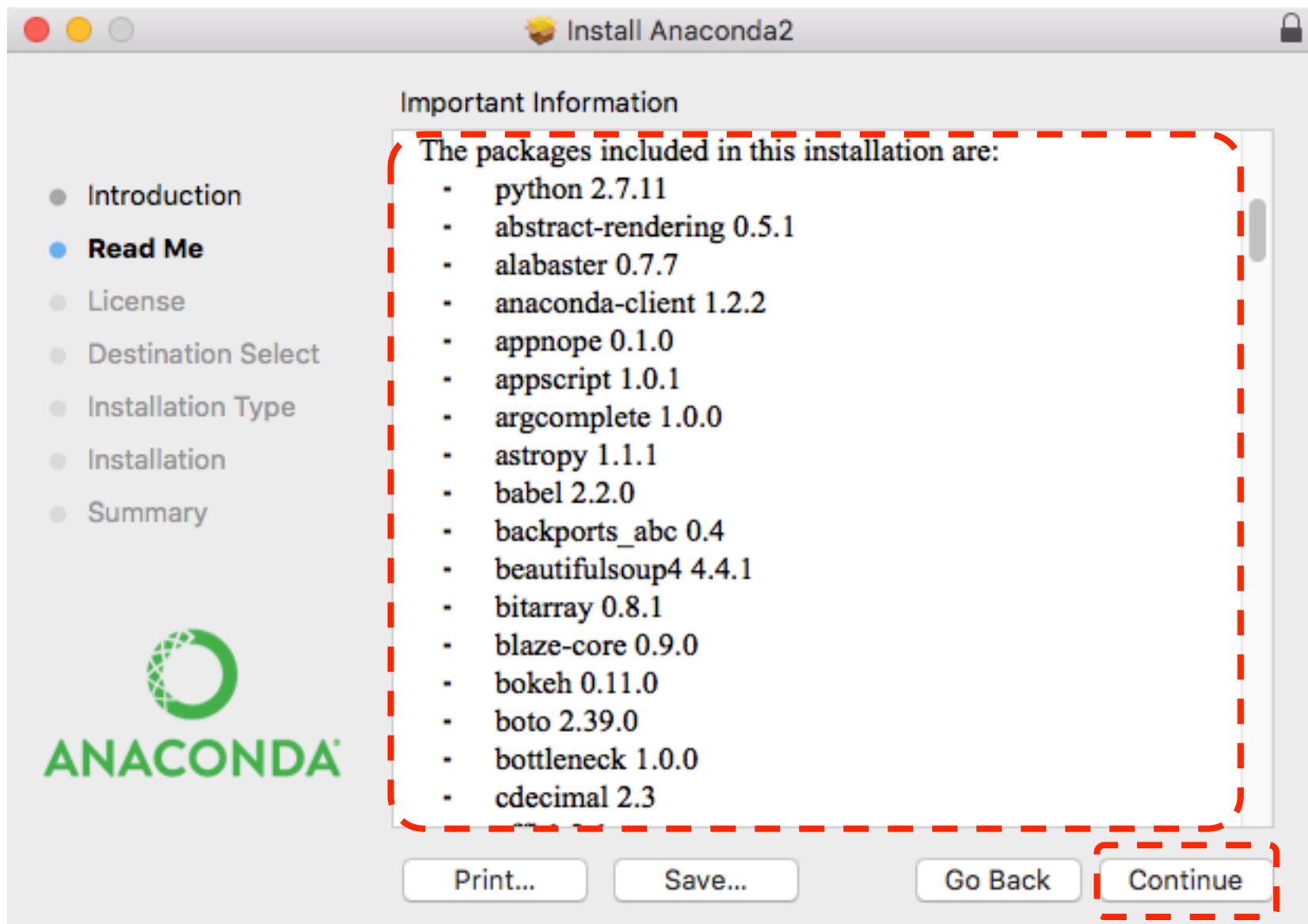
To install to a different location, select "Change Install Location..." at the "Installation Type" phase, the choose "Install on a specific disk...", choose the disk you wish to install on, and click "Choose Folder...". The "Install for me only" option will install anaconda to the default location, ~/anaconda.

The packages included in this installation are:

- python 2.7.11

At the bottom are four buttons: Print..., Save..., Go Back, and Continue.

OS X Anaconda Installation



Install Anaconda 2:

165 packages included

1	- python 2.7.11	59	- jupyter 1.0.0	128	- scikit-learn 0.17
2	- abstract-rendering 0.5.1	60	- jupyter_client 4.1.1	129	- scipy 0.17.0
3	- alabaster 0.7.7	61	- jupyter_console 4.1.0	130	- setuptools 19.6.2
4	- anaconda-client 1.2.2	62	- jupyter_core 4.0.6	131	- simplegeneric 0.8.1
5	- appnope 0.1.0	63	- launcher 1.0.0	132	- singledispatch 3.4.0.3
6	- appscript 1.0.1	64	- libdynd 0.7.1	133	- sip 4.16.9
7	- argcomplete 1.0.0	65	- libpng 1.6.17	134	- six 1.10.0
8	- astropy 1.1.1	66	- libtiff 4.0.6	135	- snowballstemmer 1.2.1
9	- babel 2.2.0	67	- libxml2 2.9.2	136	- sockjs-tornado 1.0.1
10	- backports_abc 0.4	68	- libxslt 1.1.28	137	- sphinx 1.3.5
11	- beautifulsoup4 4.4.1	69	- llvmlite 0.8.0	138	- sphinx_rtd_theme 0.1.9
12	- bitarray 0.8.1	70	- lxml 3.5.0	139	- spyder 2.3.8
13	- blaze-core 0.9.0	71	- markupsafe 0.23	140	- spyder-app 2.3.8
14	- bokeh 0.11.0	72	- matplotlib 1.5.1	141	- sqlalchemy 1.0.11
15	- boto 2.39.0	73	- mistune 0.7.1	142	- sqlite 3.9.2
16	- bottleneck 1.0.0	74	- mkl 11.3.1	143	- ssl_match_hostname 3.4.0.2
17	- cdecimal 2.3	75	- mkl-service 1.1.2	144	- statsmodels 0.6.1
18	- cffi 1.2.1	76	- multipledispatch 0.4.8	145	- sympy 0.7.6.1
19	- clyent 1.2.0	77	- nbconvert 4.1.0	146	- terminado 0.5
20	- colorama 0.3.6	78	- nbformat 4.0.1	147	- tk 8.5.18
21	- configobj 5.0.6	79	- networkx 1.11	148	- toolz 0.7.4
22	- cryptography 1.0.2	80	- nltk 3.1	149	- tornado 4.3
23	- curl 7.45.0	81	- node-webkit 0.10.1	150	- traitlets 4.1.0
24	- cycler 0.9.0	82	- nose 1.3.7	151	- unicodecsv 0.14.1
25	- cython 0.23.4	83	- notebook 4.1.0	152	- werkzeug 0.11.3
26	- cytoolz 0.7.5	84	- numba 0.23.1	153	- wheel 0.26.0
27	- datashape 0.5.0	85	- numexpr 2.4.6	154	- xlrd 0.9.4
28	- decorator 4.0.6	86	- numpy 1.10.4	155	- xlsxwriter 0.8.4
29	- docutils 0.12	87	- odo 0.4.0	156	- xlwings 0.6.4
30	- dynd-python 0.7.1	88	- openpyxl 2.3.2	157	- xlwt 1.0.0
		89	- openssl 1.0.2f	158	- xz 5.0.5
		90	- pandas 0.17.1	159	- yaml 0.1.6
		91	- path.py 8.1.2	160	- zeromq 4.1.3
		92	- patsy 0.4.0	161	- zlib 1.2.8
		93	- pep8 1.7.0	162	- anaconda 2.5.0
		94	- pexpect 3.3	163	- conda 3.19.1
		95	- pickleshare 0.5	164	- conda-build 1.19.0
				165	- conda-env 2.4.5

OS X Anaconda Installation

The screenshot shows a software license agreement window titled "Install Anaconda2". On the left, a sidebar lists steps: Introduction, Read Me, License (which is selected), Destination Select, Installation Type, Installation, and Summary. The main content area is titled "Software License Agreement" and contains the Anaconda License text. It states copyright information for Continuum Analytics, Inc., rights reserved under the 3-clause BSD License, and redistribution conditions. Three bullet points detail the requirements for redistributions: retaining copyright notice, reproducing conditions and disclaimer in documentation, and not using the name for endorsement or promotion without permission. At the bottom, a red box highlights the "Continue" button.

Install Anaconda2

Software License Agreement

=====

Anaconda License

=====

Copyright 2016, Continuum Analytics, Inc.

All rights reserved under the 3-clause BSD License:

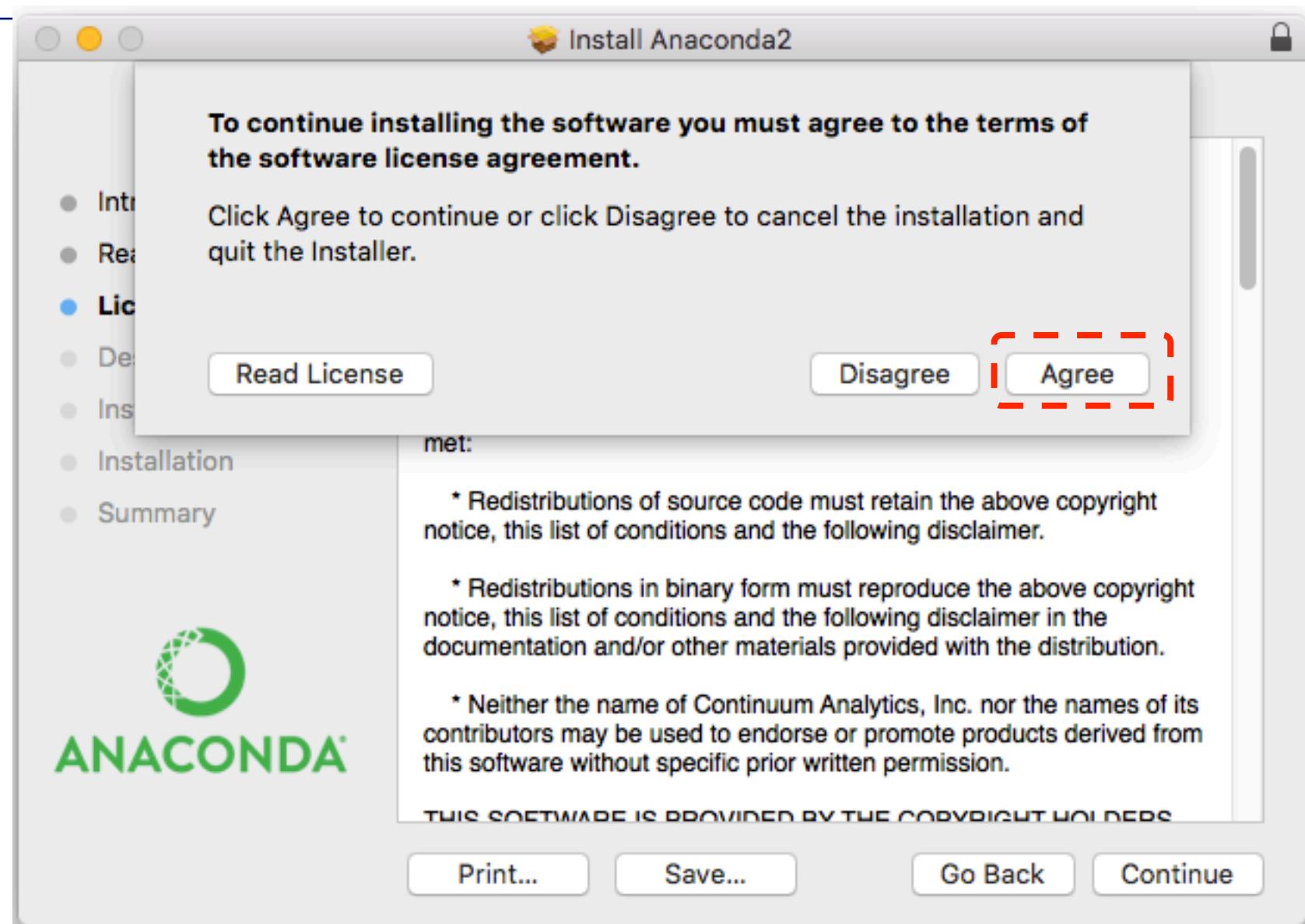
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of Continuum Analytics, Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

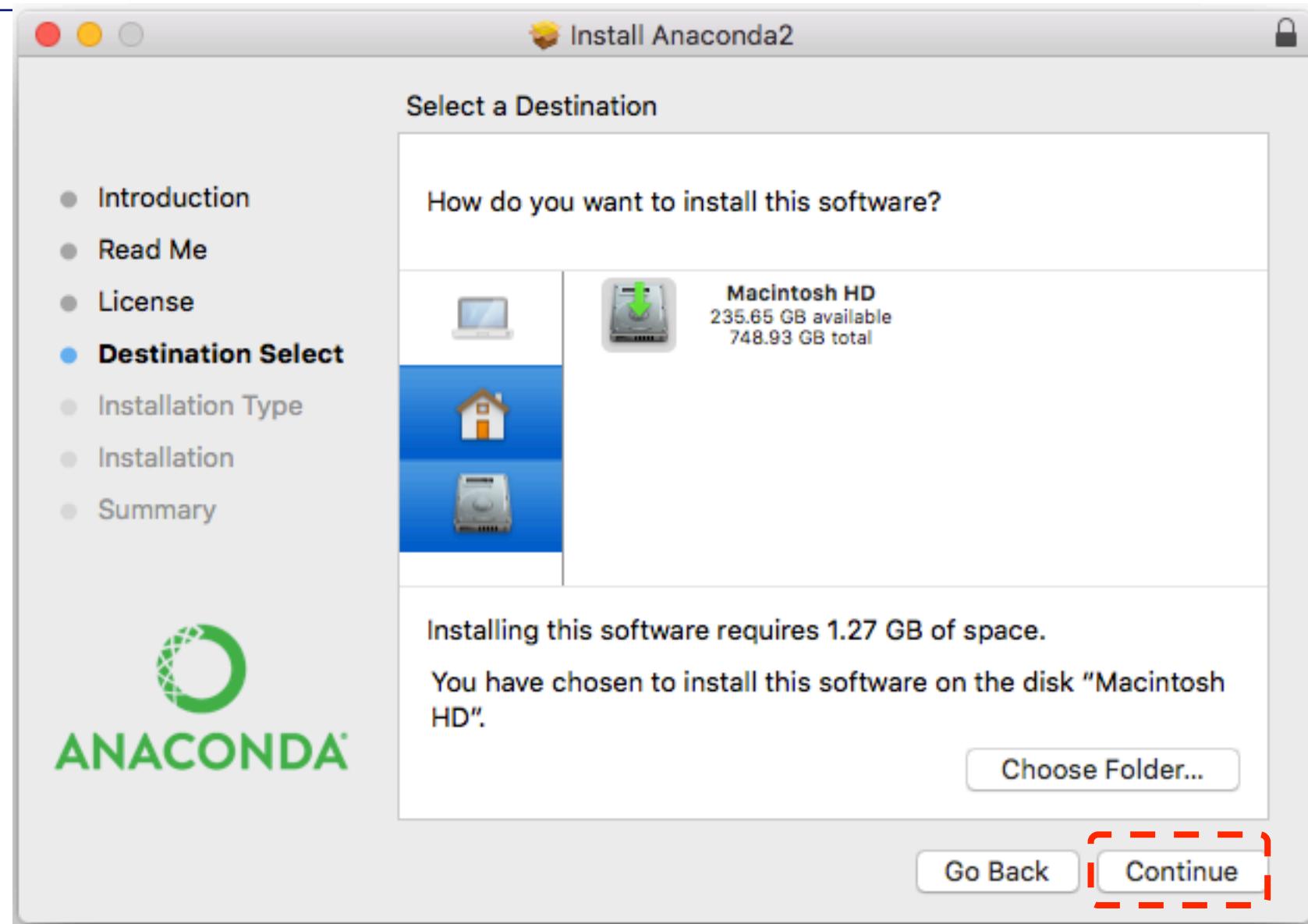
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS

Print... Save... Go Back Continue

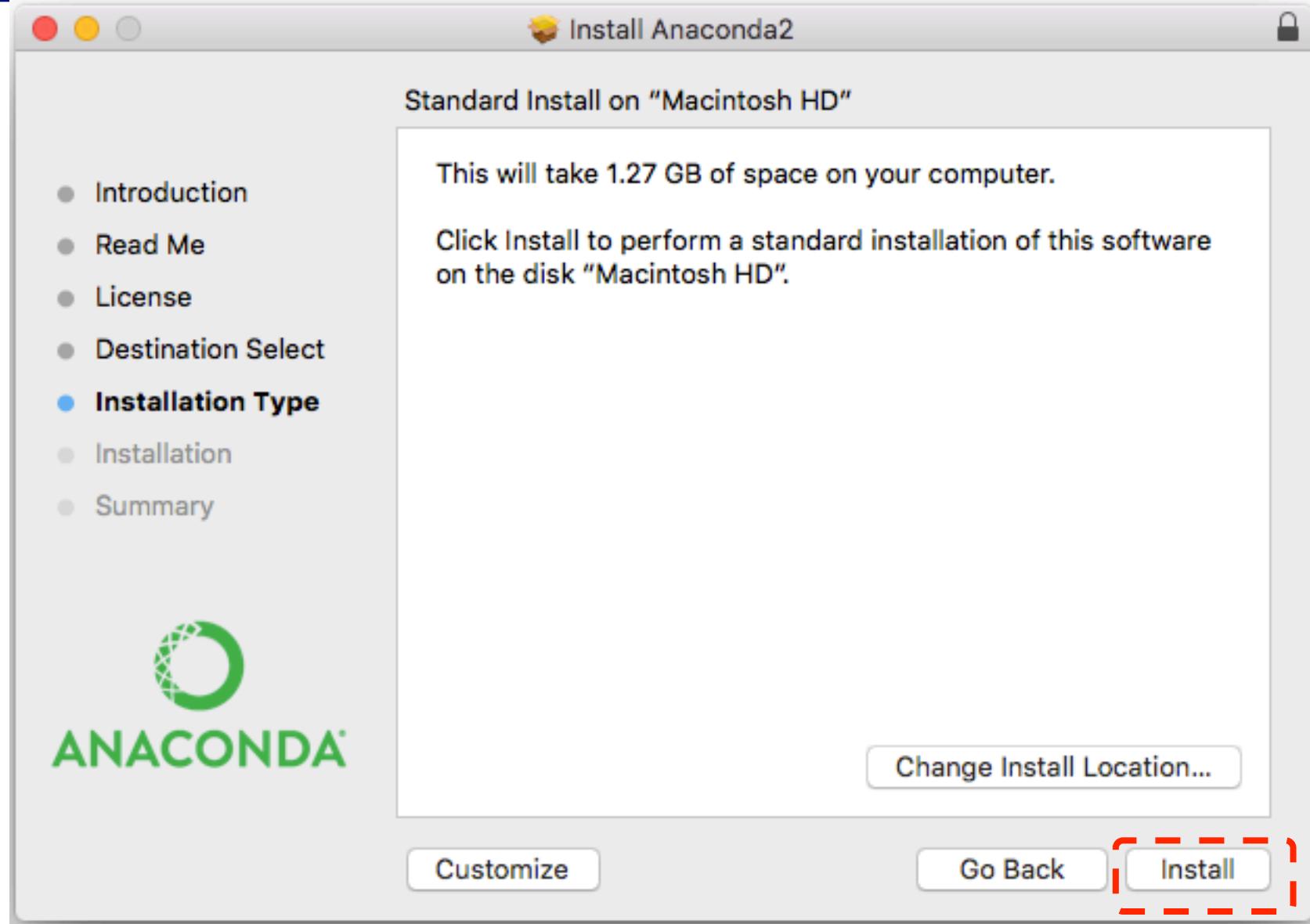
OS X Anaconda Installation



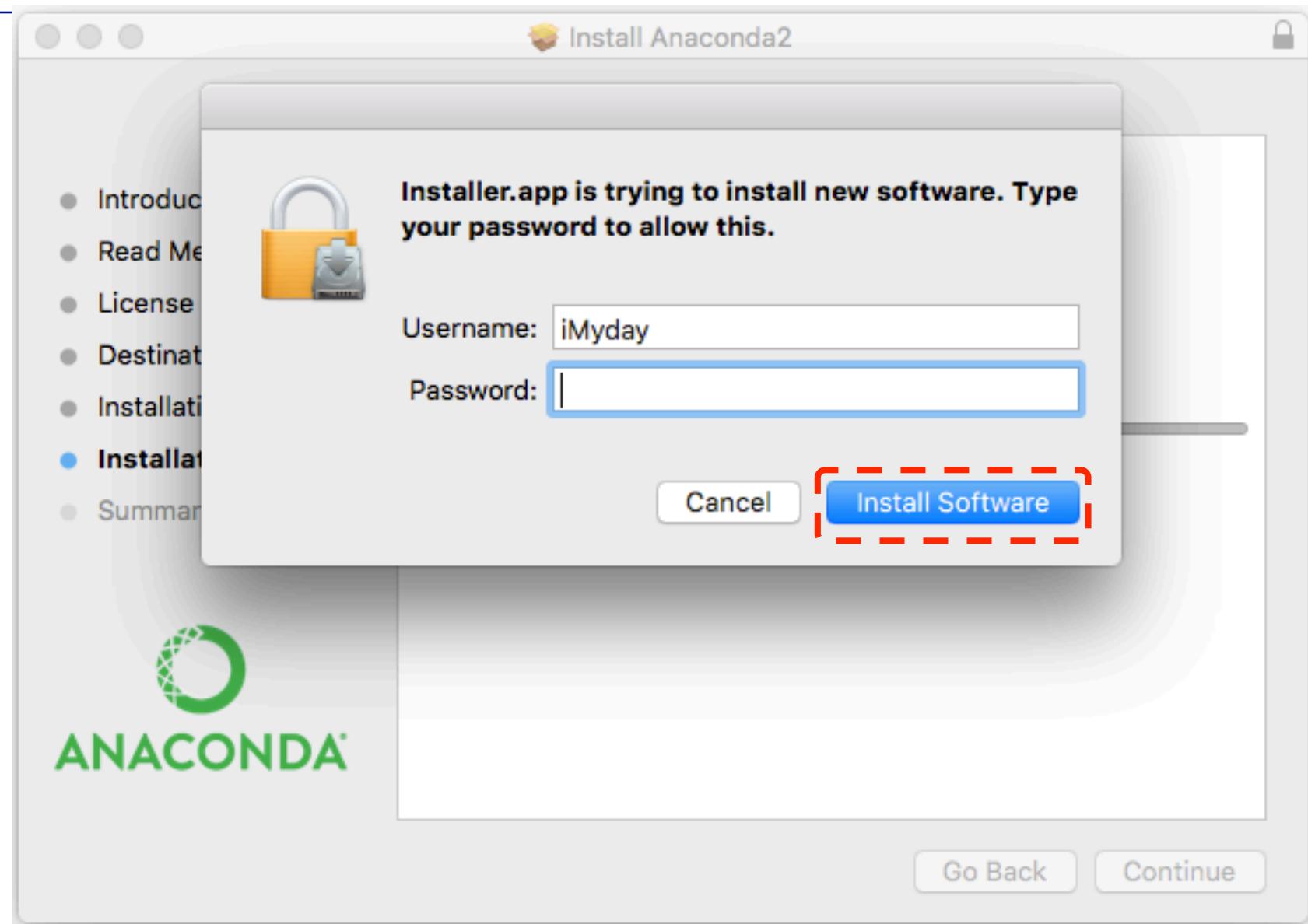
OS X Anaconda Installation



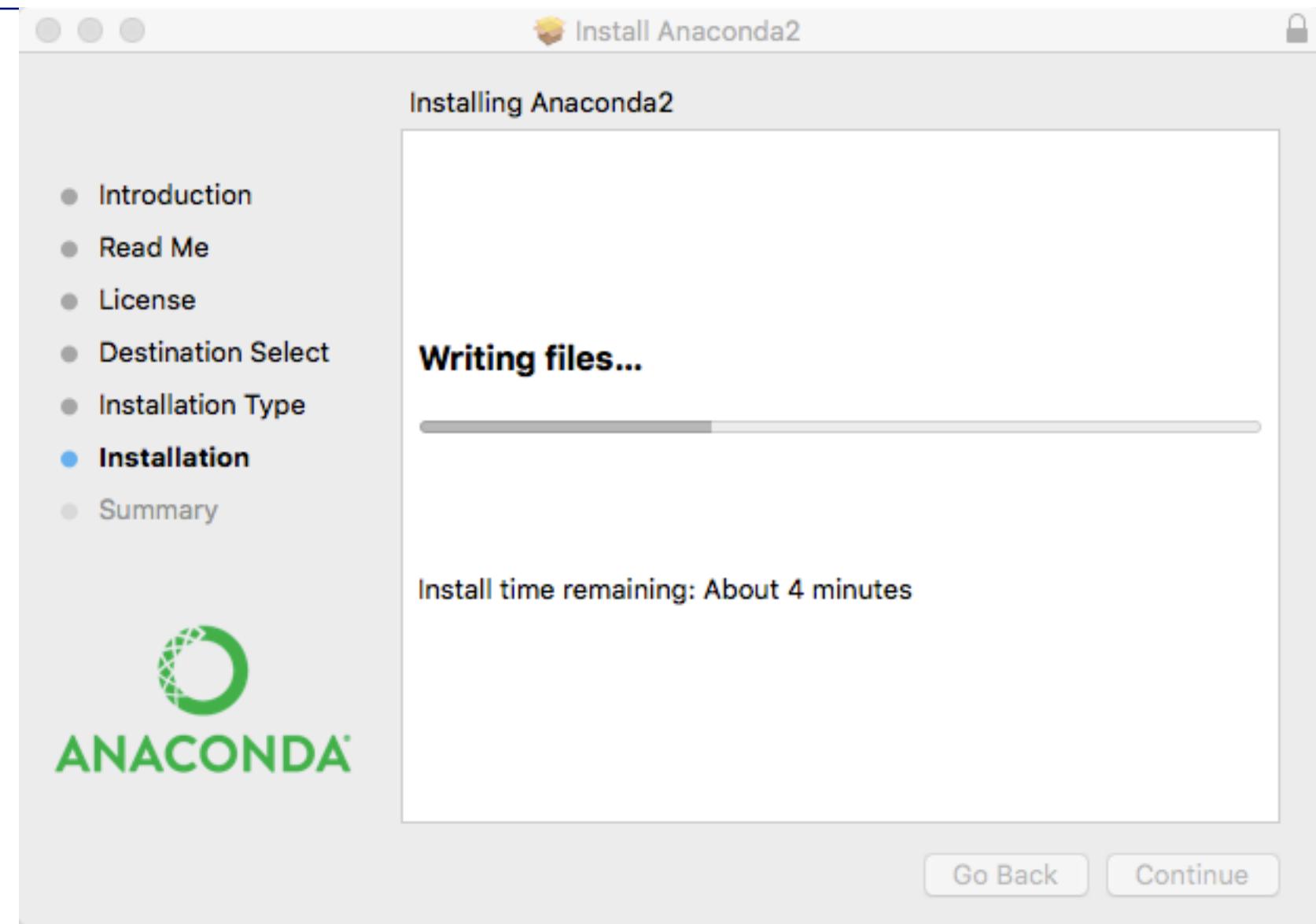
OS X Anaconda Installation



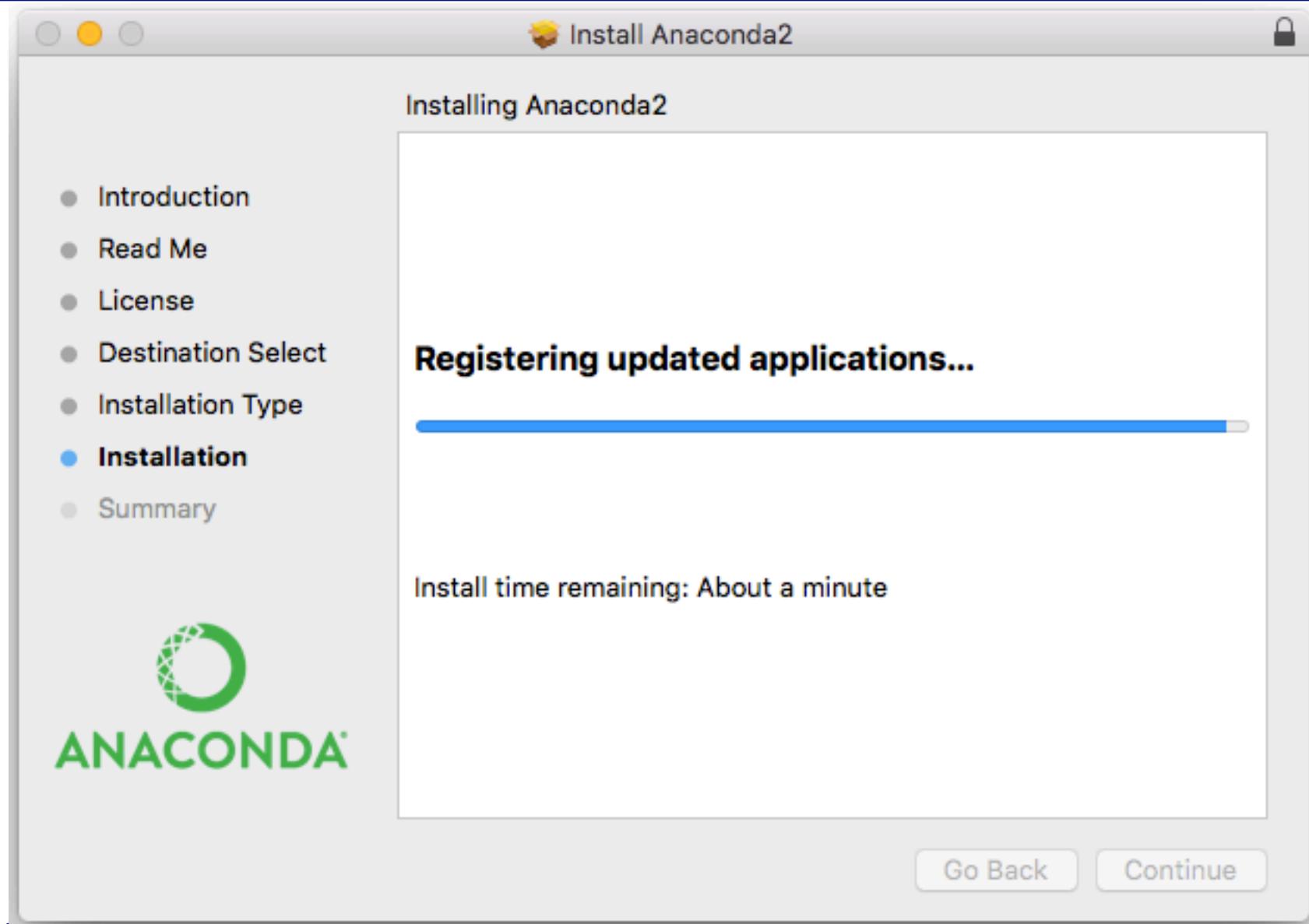
OS X Anaconda Installation



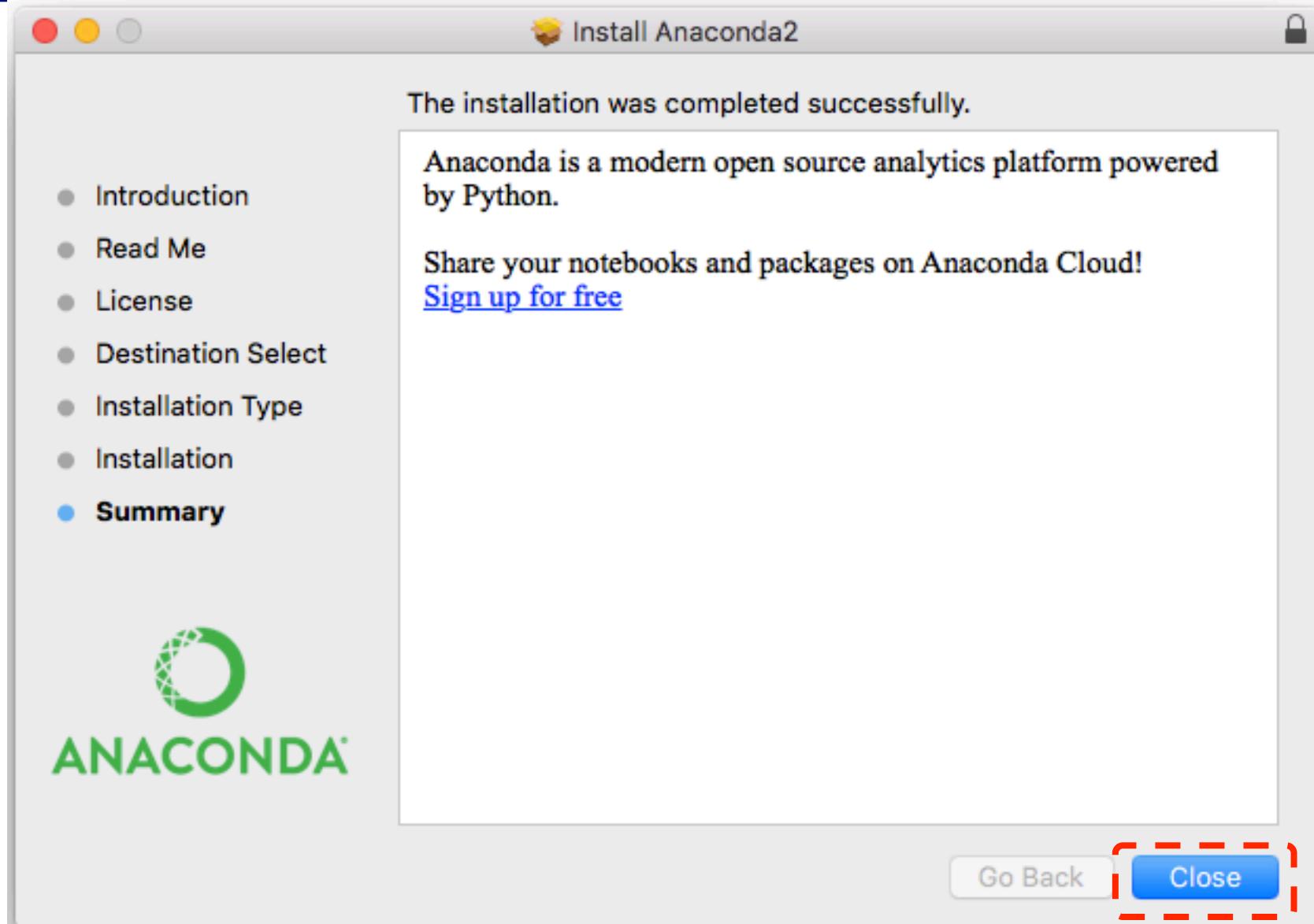
OS X Anaconda Installation



OS X Anaconda Installation



OS X Anaconda Installation



Previously installed: UPDATE ONLY

conda update --prefix /Users/jshanahan/anaconda anaconda

```
boto-2.38.0-py 100% [#####
 conda-env-2.4. 100% [#####
 cython-0.22.1- 100% [#####
 cytoolz-0.7.3- 100% [#####
 decorator-3.4. 100% [#####
 greenlet-0.4.7 100% [#####
 idna-2.0-py27_ 100% [#####
 ipaddress-1.0. 100% [#####
 llvmlite-0.5.0 100% [#####
 lxml-3.4.4-py2 100% [#####
 mistune-0.5.1- 100% [#####
 nose-1.3.7-py2 100% [#####
 astropy-1.0.3- 100% [#####
 bcolz-0.9.0-np 100% [#####
 bottleneck-1.0 100% [#####
 numba-0.19.1-n 100% [#####
 numexpr-2.4.3- 100% [#####
 blz-0.6.2-np19 100% [#####
 pillow-2.8.2-p 100% [#####
 ply-3.6-py27_0 100% [#####
 py-1.4.27-py27 100% [#####
 pycparser-2.14 100% [#####
 cffi-1.1.0-py2 100% [#####
 pycurl-7.19.5. 100% [#####
 pyflakes-0.9.2 100% [#####
 pytest-2.7.1-p 100% [#####
 python-2.7.10- 100% [#####
 python.app-1.2 100% [#####
 pytz-2015.4-py 100% [#####
 nvmem1-3 11-nv 100% [#####
] Time: 0:00:00 90.07 kB/s
] Time: 0:00:06 387.59 kB/s
] Time: 0:00:01 202.66 kB/s
] Time: 0:00:00 1.17 MB/s
] Time: 0:00:00 55.86 kB/s
] Time: 0:00:00 145.75 kB/s
] Time: 0:00:00 91.65 kB/s
] Time: 0:00:14 435.04 kB/s
] Time: 0:00:04 220.90 kB/s
] Time: 0:00:01 167.80 kB/s
] Time: 0:00:01 175.91 kB/s
] Time: 0:00:14 369.05 kB/s
] Time: 0:00:01 245.33 kB/s
] Time: 0:00:01 167.26 kB/s
] Time: 0:00:04 230.86 kB/s
] Time: 0:00:00 128.90 kB/s
] Time: 0:00:01 221.19 kB/s
] Time: 0:00:01 269.09 kB/s
] Time: 0:00:00 124.84 kB/s
] Time: 0:00:00 141.94 kB/s
] Time: 0:00:01 145.17 kB/s
] Time: 0:00:01 158.43 kB/s
] Time: 0:00:00 83.32 kB/s
] Time: 0:00:00 99.50 kB/s
] Time: 0:00:01 165.68 kB/s
] Time: 0:00:28 412.67 kB/s
] Time: 0:00:00 3.96 MB/s
] Time: 0:00:01 163.19 kB/s
] Time: 0:00:01 141.70 kB/s
```

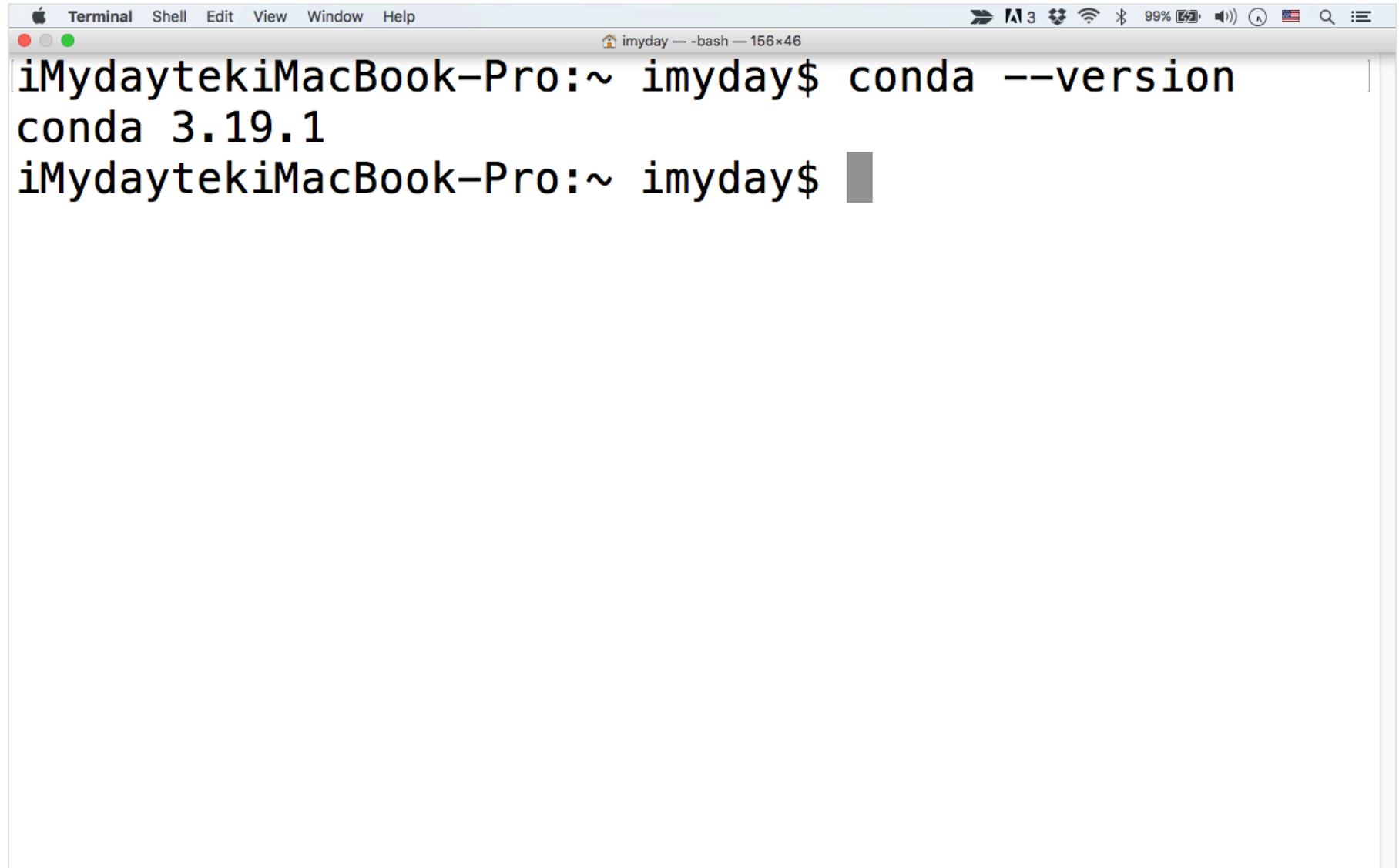
conda list

```
iMydaytekiMacBook-Pro:~ imyday$ conda list
# packages in environment at //anaconda:
#
abstract-rendering      0.5.1          np110py27_0
alabaster                0.7.7          py27_0
anaconda                 2.5.0          np110py27_0
anaconda-client           1.2.2          py27_0
appnope                  0.1.0          py27_0
appscript                 1.0.1          py27_0
argcomplete               1.0.0          py27_1
astropy                  1.1.1          np110py27_0
babel                    2.2.0          py27_0
backports-abc             0.4            <pip>
backports.ssl-match-hostname 3.4.0.2        <pip>
backports_abc              0.4          py27_0
beautifulsoup4             4.4.1          py27_0
bitarray                  0.8.1          py27_0
blaze                     0.9.0          <pip>
blaze-core                 0.9.0          py27_0
bokeh                     0.11.0         py27_0
boto                      2.39.0         py27_0
bottleneck                1.0.0          np110py27_0
cdecimal                  2.3            py27_0
cffi                     1.2.1          py27_0
```

conda --version

```
iMyday — -bash — 80x24
sqlite          3.9.2           0
ssl_match_hostname 3.4.0.2      py27_0
statsmodels     0.6.1      np110py27_0
sympy           0.7.6.1      py27_0
tables            3.2.2      <pip>
terminado        0.5       py27_1
tk                8.5.18          0
toolz             0.7.4      py27_0
tornado           4.3       py27_0
traitlets         4.1.0      py27_0
unicodecsv       0.14.1      py27_0
werkzeug          0.11.3      py27_0
wheel              0.26.0      py27_1
xlrd               0.9.4      py27_0
xlsxwriter        0.8.4      py27_0
xlwings            0.6.4      py27_0
xlwt               1.0.0      py27_0
xz                 5.0.5           0
yaml               0.1.6           0
zeromq            4.1.3           0
zlib               1.2.8           0
[iMydaytekiMacBook-Pro:~ imyday$ conda --version
conda 3.19.1
iMydaytekiMacBook-Pro:~ imyday$ ]
```

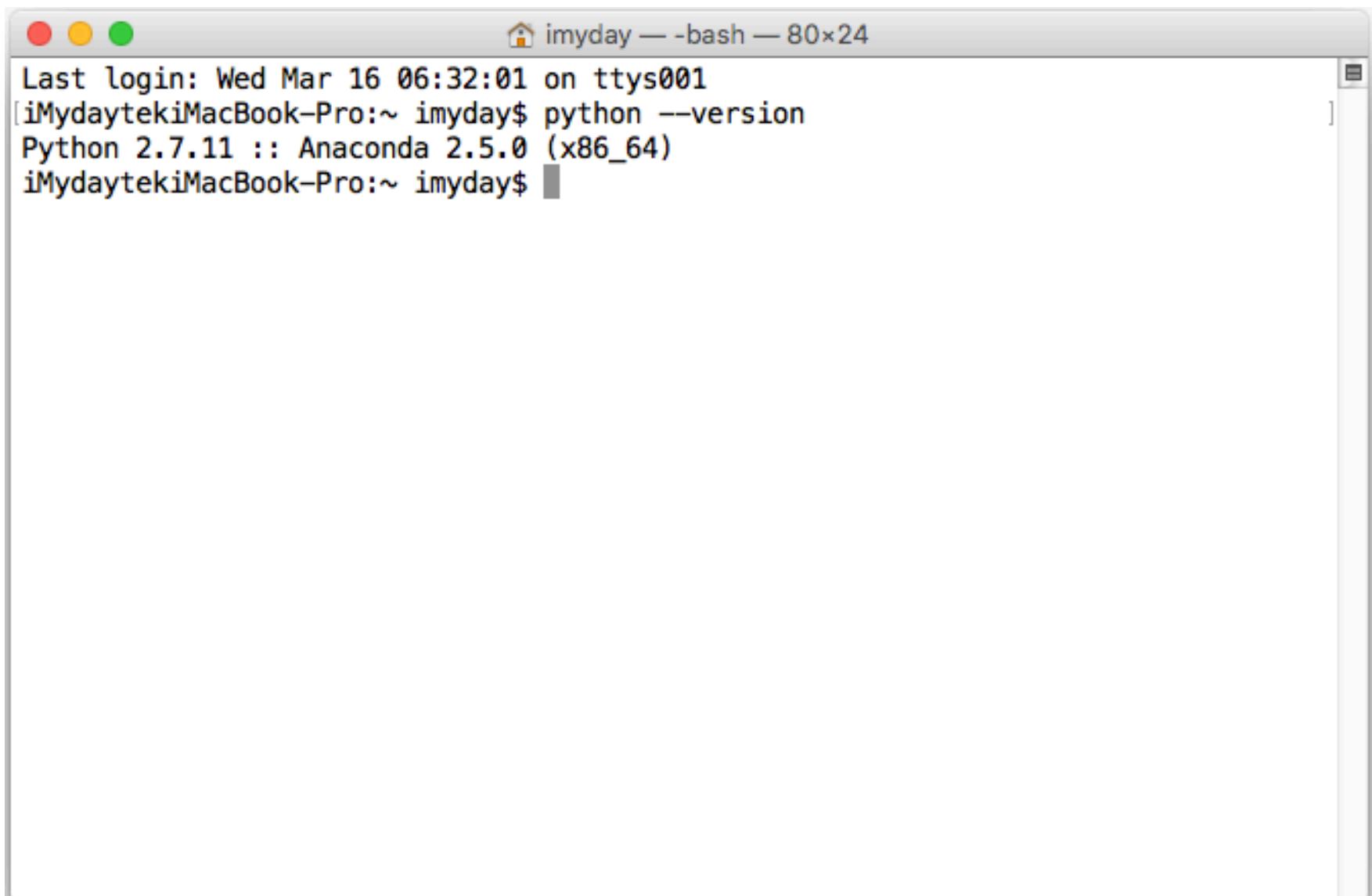
conda --version



A screenshot of a Mac OS X Terminal window. The window title bar shows "imyday — bash — 156x46". The menu bar includes "Terminal", "Shell", "Edit", "View", "Window", and "Help". The status bar at the top right shows battery level (99%), signal strength, and other system icons. The main terminal area displays the command "conda --version" and its output: "conda 3.19.1". The cursor is visible as a vertical bar on the right side of the terminal window.

```
iMydaytekiMacBook-Pro:~ imyday$ conda --version
conda 3.19.1
iMydaytekiMacBook-Pro:~ imyday$
```

python --version

A screenshot of a Mac OS X terminal window titled "imyday — bash — 80x24". The window shows the command "python --version" being run and its output. The output indicates a Python 2.7.11 installation from Anaconda 2.5.0 for an x86_64 architecture.

```
Last login: Wed Mar 16 06:32:01 on ttys001
[iMydaytekiMacBook-Pro:~ imyday$ python --version
Python 2.7.11 :: Anaconda 2.5.0 (x86_64)
iMydaytekiMacBook-Pro:~ imyday$ ]
```

ipython notebook

```
iMydaytekiMacBook-Pro:~ imyday$ ipython notebook
[I 14:26:49.944 NotebookApp] Serving notebooks from local directory: /Users/imyday
[I 14:26:49.944 NotebookApp] 0 active kernels
[I 14:26:49.944 NotebookApp] The Jupyter Notebook is running at: http://localhost:8888/
[I 14:26:49.944 NotebookApp] Use Control-C to stop this server and shut down all
kernels (twice to skip confirmation).
[W 14:26:56.639 NotebookApp] 404 GET /api/kernels/a87ab95b-6d6e-44d3-aaa7-c1901c
960677/channels?session_id=265FB16817FB4AB79202F6D3C3BDB0E6 (::1): Kernel does n
ot exist: a87ab95b-6d6e-44d3-aaa7-c1901c960677
[W 14:26:56.663 NotebookApp] 404 GET /api/kernels/a87ab95b-6d6e-44d3-aaa7-c1901c
960677/channels?session_id=265FB16817FB4AB79202F6D3C3BDB0E6 (::1) 95.43ms refere
r=None
[W 14:26:56.681 NotebookApp] 404 GET /api/kernels/b7fae9a6-d77b-4ead-832c-c070b1
8d642b/channels?session_id=EF4C761633E541C88568CDBCDE1091B7 (::1): Kernel does n
ot exist: b7fae9a6-d77b-4ead-832c-c070b18d642b
[W 14:26:56.683 NotebookApp] 404 GET /api/kernels/b7fae9a6-d77b-4ead-832c-c070b1
8d642b/channels?session_id=EF4C761633E541C88568CDBCDE1091B7 (::1) 6.62ms referer
=None
[W 14:27:29.595 NotebookApp] 404 GET /api/kernels/a87ab95b-6d6e-44d3-aaa7-c1901c
960677/channels?session_id=265FB16817FB4AB79202F6D3C3BDB0E6 (::1): Kernel does n
ot exist: a87ab95b-6d6e-44d3-aaa7-c1901c960677
[W 14:27:29.631 NotebookApp] 404 GET /api/kernels/a87ab95b-6d6e-44d3-aaa7-c1901c
```

conda search python

```
iMydaytekiMacBook-Pro:~ imyday$ conda search python
Using Anaconda Cloud api site https://api.anaconda.org
Fetching package metadata: ....
biopython          1.60          np17py27_0  defaults
                   1.60          np17py26_0  defaults
                                         py27_0  defaults
                                         py26_0  defaults
                                         py27_1  defaults
                                         py26_1  defaults
                                         py27_0  defaults
                                         py26_0  defaults
                                         py33_1  defaults
                                         py27_1  defaults
                                         py26_1  defaults
                                         py34_2  defaults
                                         py33_2  defaults
                                         py27_2  defaults
                                         py26_2  defaults
                                         py34_3  defaults
                                         py33_3  defaults
                                         py27_3  defaults
                                         py26_3  defaults
                                         py35_4  defaults
                                         py34_4  defaults
                                         py33_4  defaults
                                         py27_4  defaults
                                         py26_4  defaults
                                         py27_0  defaults
wxpython           3.0           np17py27_0  defaults
iMydaytekiMacBook-Pro:~ imyday$
```

INSTALL: R and Jupyter Notebooks

- **STEP 1: install Jupyter**
 - Install Anaconda (Python, Jupyter)
 - <http://jupyter.readthedocs.io/en/latest/install.html>
- **STEP 2: Install R Kernel**
 - <https://www.continuum.io/blog/developer/jupyter-and-conda-r>

“R Essentials” setup

The Anaconda team has created an “[R Essentials](#)” bundle with the IRKernel and over 80 of the most used R packages for data science, including `dplyr`, `shiny`, `ggplot2`, `tidyverse`, `caret` and `nnet`.

Downloading “R Essentials” requires conda. [Miniconda](#) includes conda, Python, and a few other necessary packages, while [Anaconda](#) includes all this and over 200 of the most popular [Python packages](#) for science, math, engineering, and data analysis. Users may install all of Anaconda at once, or they may install Miniconda at first and then use conda to install any other packages they need, including any of the packages in Anaconda.

Once you have conda, you may install “R Essentials” into the current environment:

**On MAC or Windows got the shell/CMD and type
*conda install -c r r-essentials***

or create a new environment just for “R essentials”:

```
conda create -n my-r-env -c r r-essentials
```

Bash

Might take 10-20 minutes

Jupyter

Jupyter provides a great notebook interface to write your analysis and share it with your peers. Open a shell and run this command to start the Jupyter notebook interface in your browser:

```
jupyter notebook
```

Bash

Install IRKernel for Jupyter

- The Anaconda team has created an “R Essentials” bundle with the IRKernel and over 80 of the most used R packages for data science, including dplyr, shiny, ggplot2, tidyr, caret and nnet.

On WINDOWS CMD Shell type
conda install -c r r-essentials

Might take 10-20 minutes

```
conda install -c r r-essentials

r-sparsem:          1.7-r3.3.0_0
r-spatial:          7.3_11-r3.3.0_0
r-stringi:          1.0_1-r3.3.0_0
r-stringr:          1.0.0-r3.3.0_0
r-survival:         2.39_4-r3.3.0_0
r-tidyr:            0.4.1-r3.3.0_0
r-ttr:               0.23_1-r3.3.0_0
r-uuid:              0.1_2-r3.3.0_0
r-xtable:            1.8_2-r3.3.0_0
r-xts:               0.9_7-r3.3.0_2
r-yaml:              2.1.13-r3.3.0_2
r-zoo:               1.7_13-r3.3.0_0

The following packages will be UPDATED:

  conda:           4.0.5-py27_0 --> 4.0.8-py27_0

Proceed ([y]/n)? y

Fetching packages ...
msys2-conda-ep 100% [#####] Time: 0:00:00 24.22 kB/s
m2w64-c-ares-1 100% [#####] Time: 0:00:00 207.52 kB/s
m2w64-expat-2. 100% [#####] Time: 0:00:00 232.56 kB/s
m2w64-gmp-6.1. 100% [#####] Time: 0:00:02 249.70 kB/s
m2w64-gsl-2.1- 38% [#####] Time: 0:00:11 83.96 kB/s
```

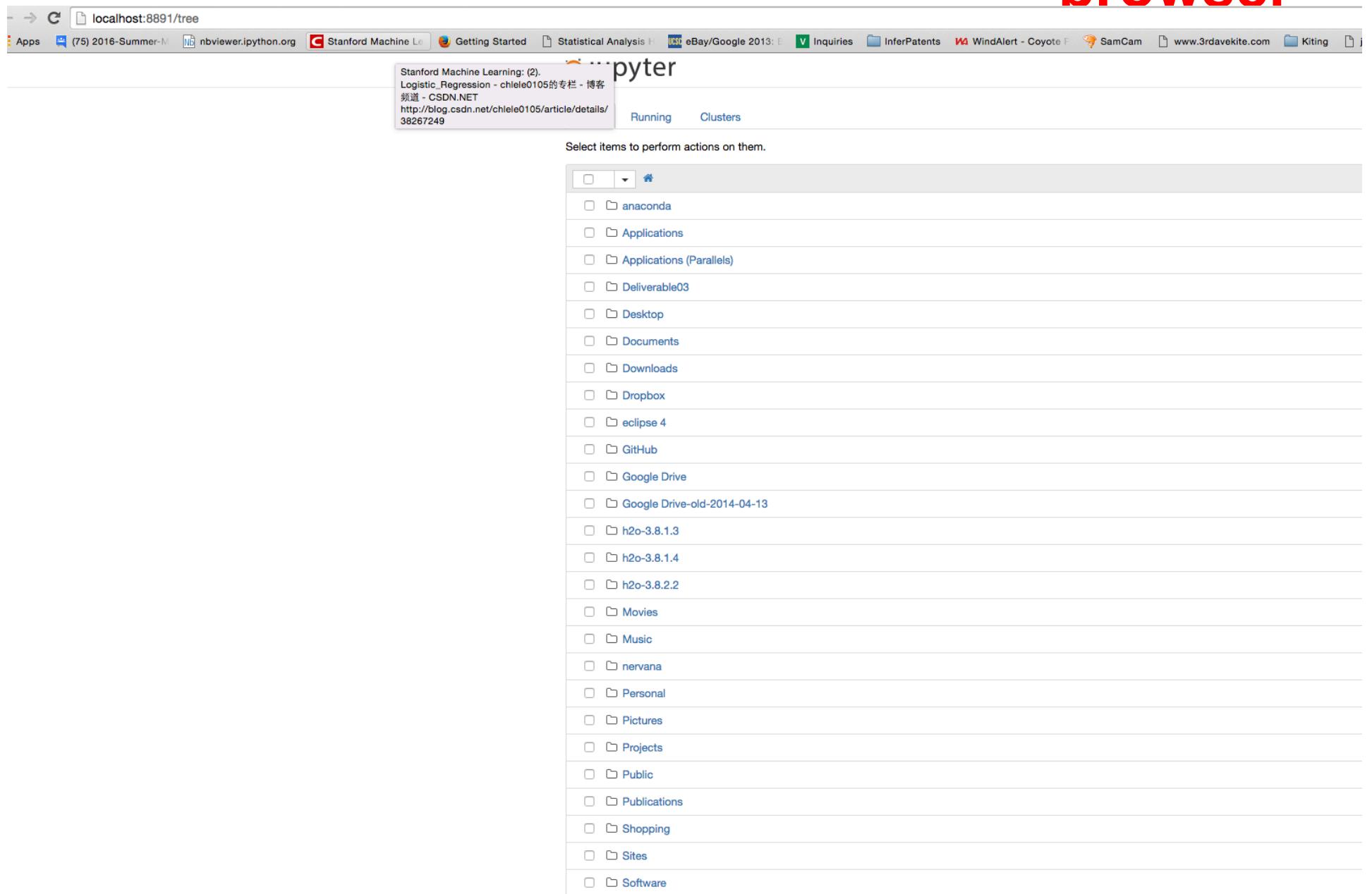
To start the Notebook on a MAC/Windows

- **To start the Jupyter server on the MAC**
 - See next slide
 - /Users/jshanahan/anaconda/bin/ipython notebook&
- **To start the Jupyter server on WINDOWS**
 - See Slides below

To start the Notebook on a MAC

```
Large : bash ... bash ... hadoop@ip-172-31-26-92:~ bash - 98x38
jshanahan - hadoop@ip-172-31-26-92:~ bash - 98x38
JAMES-SHANAHANS-Desktop-Pro-2:Lectures-UC-Berkeley-ML-Class-2015 jshanahan$ JAMES-SHANAHANS-Desktop-Pro-2:Lectures-UC-Berkeley-ML-Class-2015 jshanahan$ JAMES-SHANAHANS-Desktop-Pro-2:Lectures-UC-Berkeley-ML-Class-2015 jshanahan$ JAMES-SHANAHANS-Desktop-Pro-2:Lectures-UC-Berkeley-ML-Class-2015 jshanahan$ cd ~/Google\ Drive JAMES-SHANAHANS-Desktop-Pro-2:Google Drive jshanahan$ cd R R-SQL Stuff.gdoc ResearchIdeas.gdoc R-base+RedHat Notes.gdoc Restuarants.gdoc RStuff/ Resume/ RV- Sherman Island parking.gsheet ReviewForm-KDD-2016_review.rtf References.gdoc Ruben slides.pptx RegistrationReports.gsheet Rug and underlay rug for office.gdoc Reimbursement Request.gdoc JAMES-SHANAHANS-Desktop-Pro-2:Google Drive jshanahan$ cd RStuff/ JAMES-SHANAHANS-Desktop-Pro-2:RStuff jshanahan$ ls Data Docs Icon? Src JAMES-SHANAHANS-Desktop-Pro-2:RStuff jshanahan$ pwd /Users/jshanahan/Google Drive/RStuff JAMES-SHANAHANS-Desktop-Pro-2:RStuff jshanahan$ ls Data Docs Icon? Src JAMES-SHANAHANS-Desktop-Pro-2:RStuff jshanahan$ pwd /Users/jshanahan/Google Drive/RStuff JAMES-SHANAHANS-Desktop-Pro-2:RStuff jshanahan$ cd ../../ JAMES-SHANAHANS-Desktop-Pro-2:~ jshanahan$ pwd /Users/jshanahan JAMES-SHANAHANS-Desktop-Pro-2:~ jshanahan$ /Users/jshanahan/anaconda/bin/ipython notebook& [1] 3392 JAMES-SHANAHANS-Desktop-Pro-2:~ jshanahan$ [I 15:36:40.008 NotebookApp] The port 8888 is already in use, trying another random port. [I 15:36:40.009 NotebookApp] The port 8889 is already in use, trying another random port. [I 15:36:40.010 NotebookApp] The port 8890 is already in use, trying another random port. [I 15:36:40.024 NotebookApp] Serving notebooks from local directory: /Users/jshanahan [I 15:36:40.025 NotebookApp] 0 active kernels [I 15:36:40.025 NotebookApp] The IPython Notebook is running at: http://localhost:8891/
```

Opens a notebook window in your web browser



To start Jupyter Notebook on Windows

3. Running the Jupyter Notebook ↴

3.1. Launching Jupyter Notebook App

The [Jupyter Notebook App](#) can be launched by clicking on the *Jupyter Notebook* icon installed by Anaconda in the start menu (Windows) or by typing in a terminal (*cmd* on Windows):

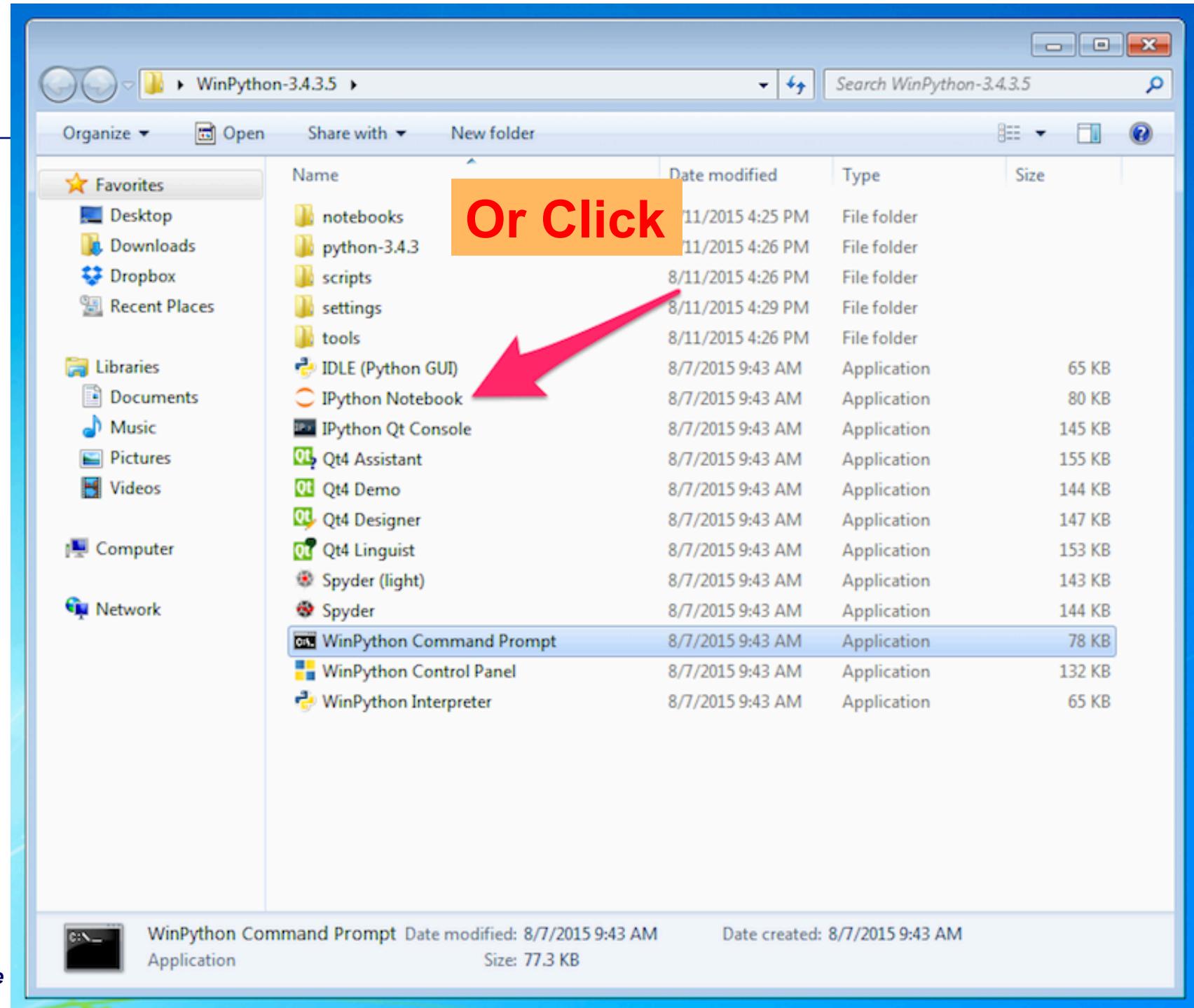
Either

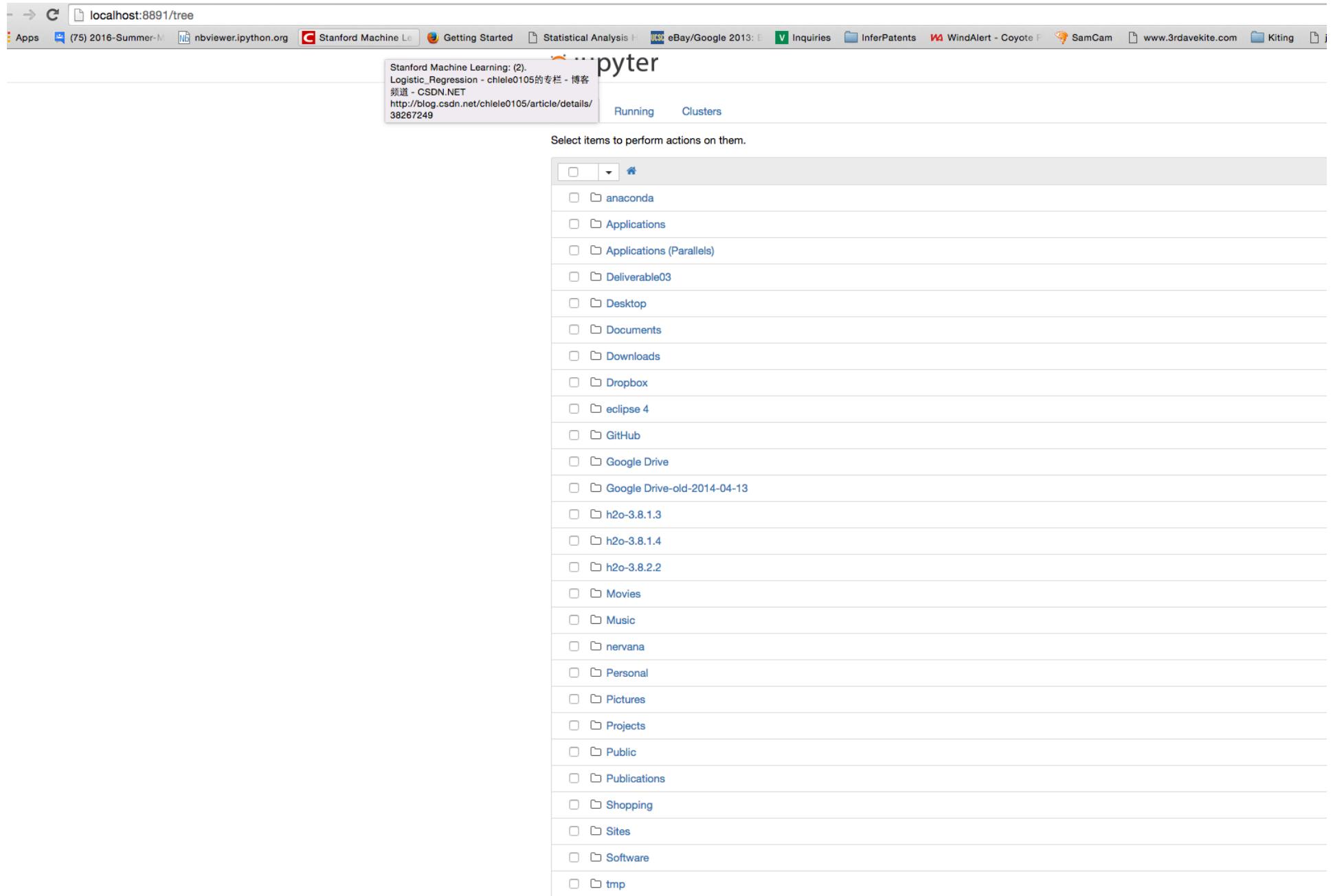
```
jupyter notebook
```

This will launch a new browser window (or a new tab) showing the [Notebook Dashboard](#), a sort of control panel that allows (among other things) to select which notebook to open.

When started, the [Jupyter Notebook App](#) can access only files within its start-up folder (including any sub-folder). If you store the notebook documents in a subfolder of your user folder no configuration is necessary. Otherwise, you need to choose a folder which will contain all the notebooks and set this as the [Jupyter Notebook App](#) start-up folder.

See below for platform-specific instructions on how to start [Jupyter Notebook App](#) in a specific folder.





Jupyter

Jupyter provides a great notebook interface to write your analysis and share it with your peers. Open a shell and run this command to start the Jupyter notebook interface in your browser:

```
jupyter notebook
```

Bash

Start a new R notebook:



You can immediately write and run R code in the notebook cells.

An R notebook example

Now you can:

- import the data wrangling R package, dplyr:

```
In [1]: library(dplyr)
```

- import the data wrangling R package, dplyr:

```
In [1]: library(dplyr)
```

S

- explore one of the available datasets, such as the `iris`:

```
In [2]: iris
```

Out[2]:

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3	1.4	0.2	setosa
...					

S

- calculate the average sepal width by species:

```
In [3]: iris %>%
  group_by(Species) %>%
  summarise(Sepal.Width.Avg = mean(Sepal.Width)) %>%
  arrange(Sepal.Width.Avg)
```

Out [3]:

	Species	Sepal.Width.Avg
1	versicolor	2.77
2	virginica	2.974
3	setosa	3.428

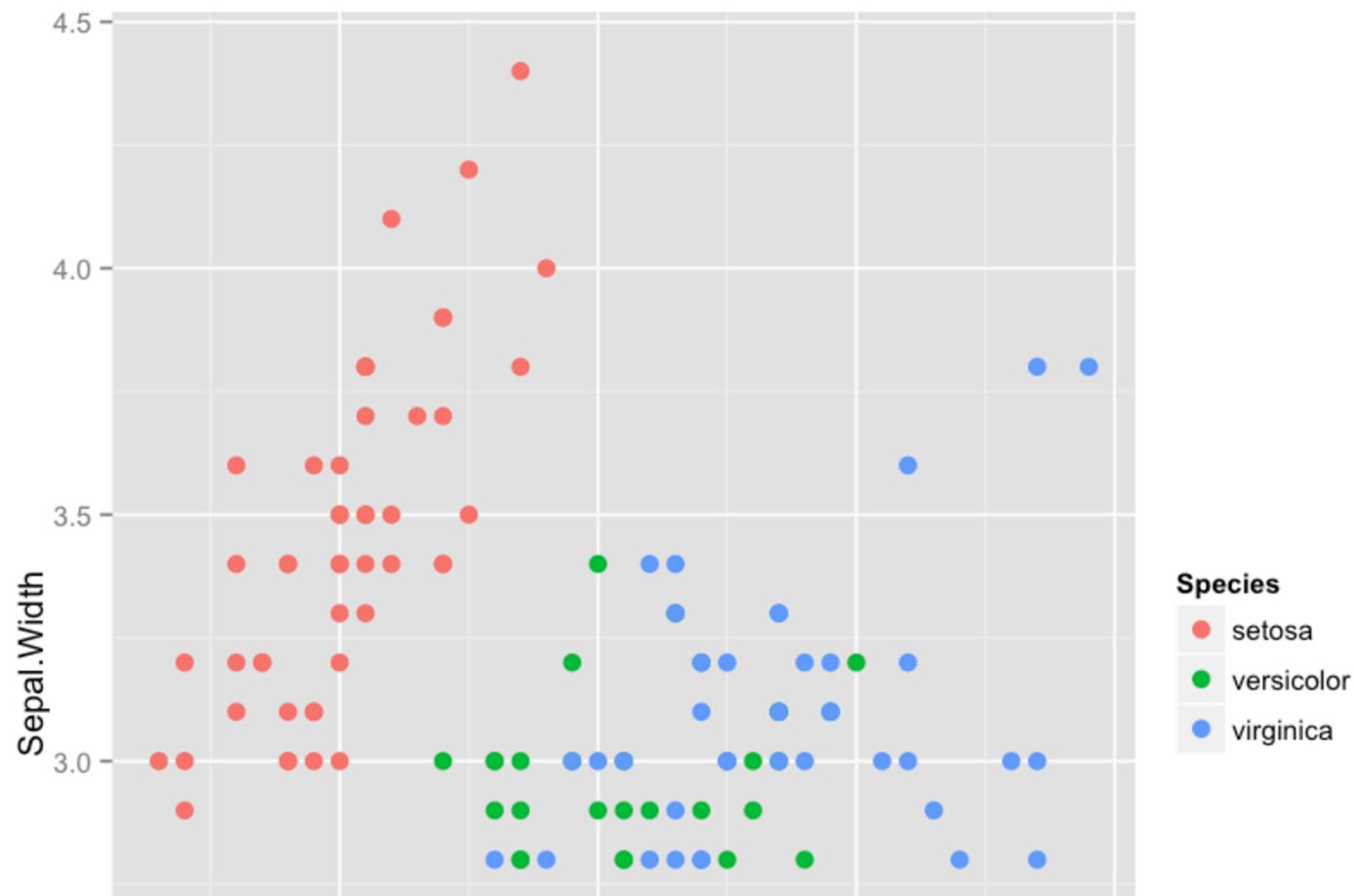
```
In [4]: library(ggplot2)
```

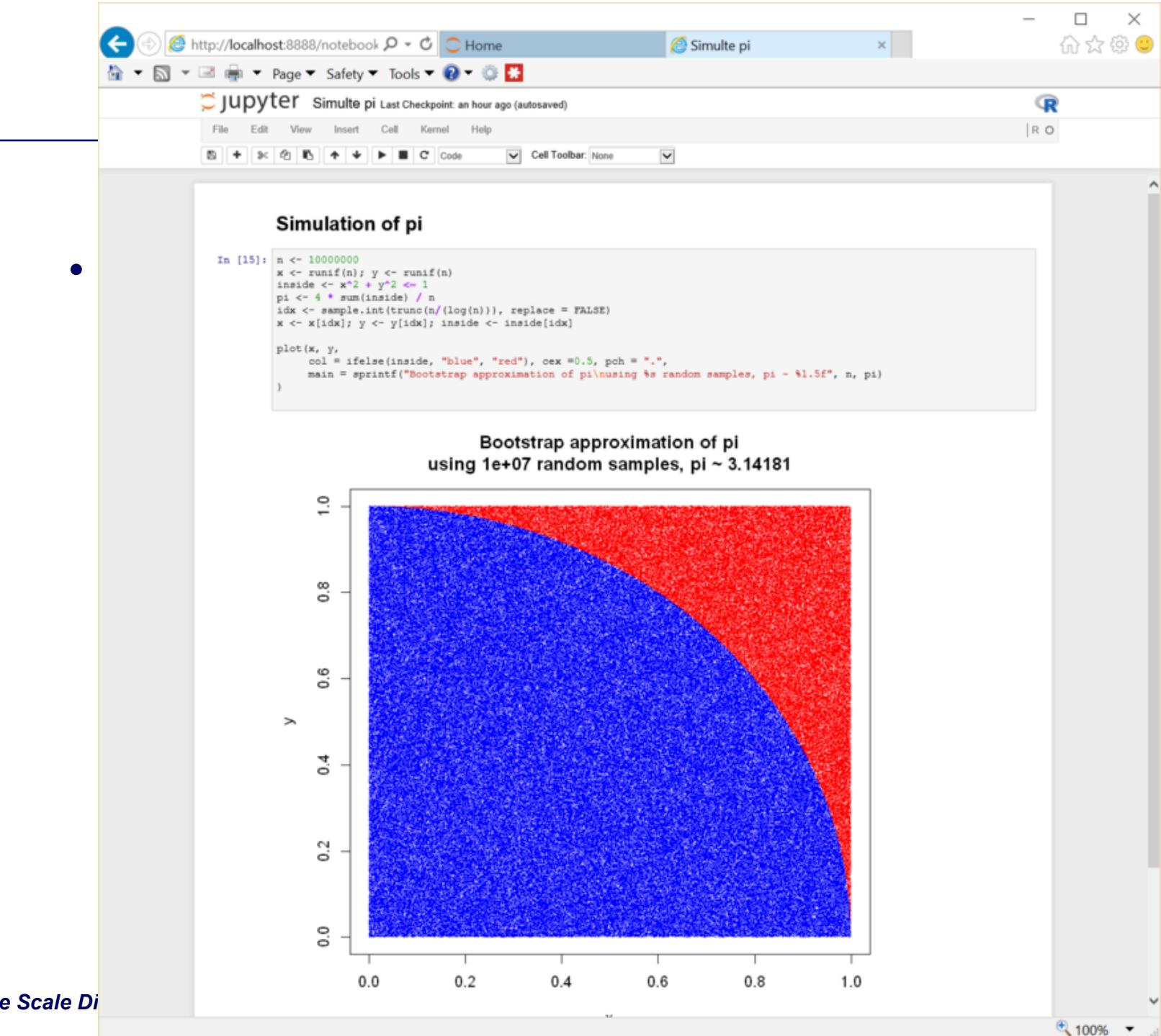
S

- plot the Sepal.Width vs. Sepal.Length

```
In [5]: ggplot(data=iris, aes(x=Sepal.Length, y=Sepal.Width, color=Species))
```

S





localhost:8888/notebooks/Jupyter%20and%20conda%20for%20R.ipynb

Jupyter Jupyter and conda for R Last Checkpoint: an hour ago (autosaved)

File Edit View Insert Cell Kernel Help

Cell Toolbar: None
Edit Metadata
Raw Cell Format
Slideshow

Jupyter and conda for R

In this notebook, we discuss how data scientists working in R can use Jupyter, the IRKernel ("Interactive R Kernel") and conda in their workflows.

[Jupyter](#), previously known as IPython, has seen a great adoption among data scientists, researchers and analysts. It has improved their workflow by providing a notebook user interface that allows them to mix executable code with narrative text, equations, interactive visualizations and images. It has advanced the state of reproducible research and training, and brought a tool for collaboration to teams. Though it started serving the Python community, it is now available in 50 different language through the different Jupyter kernels. The [IRKernel](#) is the native R kernel for Jupyter, which allows R users to benefit from all the notebook's functionality in their projects.

[Conda](#) is a package manager that data scientists, researchers, and analysts use to install and organize project dependencies. It allows them to easily build and share downloadable bundles of packages, also called metapackages, with their peers. Conda works with Linux, OS X, and Windows, and is language agnostic, which allows us to use it with any programming language or even multi-language projects.

We will explore how conda and Jupyter make it really easy to start a Data Science Project [Convert R notebooks to slides step 1](#)

Organize the cells into slides and subslides:

jupyter Jupyter and conda for R Last Checkpoint: a few seconds ago (autosaved)

File Edit View Insert Cell Kernel Help

Cell Toolbar: Slideshow

An R notebook example

Now you can:

- import the data wrangling R package, dplyr:

In [22]:

```
library(dplyr)
```

localhost:8888/notebooks/Jupyter%20and%20conda%20for%20R.ipynb

Jupyter Jupyter and conda for R Last Checkpoint: an hour ago (autosaved)

File Edit View Insert Cell Kernel Help

Cell Toolbar: None Edit Metadata Raw Cell Format Slideshow

Jupyter and conda for R

In this notebook, we discuss how data scientists working in R can use Jupyter, the IRKernel ("Interactive R Kernel") and conda in their workflows.

[Jupyter](#), previously known as IPython, has seen a great adoption among data scientists, researchers and analysts. It has improved their workflow by providing a notebook user interface that allows them to mix executable code with narrative text, equations, interactive visualizations and images. It has advanced the state of reproducible research and training, and brought a tool for collaboration to teams. Though it started serving the Python community, it is now available in 50 different language through the different Jupyter kernels. The [IRKernel](#) is the native R kernel for Jupyter, which allows R users to benefit from all the notebook's functionality in their projects.

[Conda](#) is a package manager that data scientists, researchers, and analysts use to install and organize project dependencies. It allows them to easily build and share downloadable bundles of packages, also called metapackages, with their peers. Conda works with Linux, OS X, and Windows, and is language agnostic, which allows us to use it with any programming language or even multi-language projects.

We will explore how conda and Jupyter make it really easy to start a Data Science Project [Convert R notebooks to slides step 1](#)

Organize the cells into slides and subslides:

jupyter Jupyter and conda for R Last Checkpoint: a few seconds ago (autosaved)

File Edit View Insert Cell Kernel Help

Cell Toolbar: Slideshow

An R notebook example

Now you can:

- import the data wrangling R package, dplyr:

In [22]:

```
library(dplyr)
```

The screenshot shows a Jupyter R notebook interface. At the top, there's a toolbar with File, Edit, View, Insert, Cell, Kernel, Help, and a Cell Toolbar dropdown set to "Markdown". Below the toolbar, there's a "Slide Type" dropdown set to "Slide". The main area contains a slide titled "An R notebook example". A sub-slide follows, with the text "Now you can:" and a bullet point: "• import the data wrangling R package, dplyr:". Below this is an input cell labeled "In [22]:" containing the command "library(dplyr)". Another sub-slide follows, with the text "• explore one of the available datasets, like the iris:". Below this is another input cell labeled "In [23]:" containing the command "iris". The output cell "Out[23]:" displays a table with the following data:

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3	1.4	0.2	setosa

And convert:

```
jupyter nbconvert my_r_notebook.ipynb --to slides --post serve
```

Bash

This opens a browser showing the slidedeck:

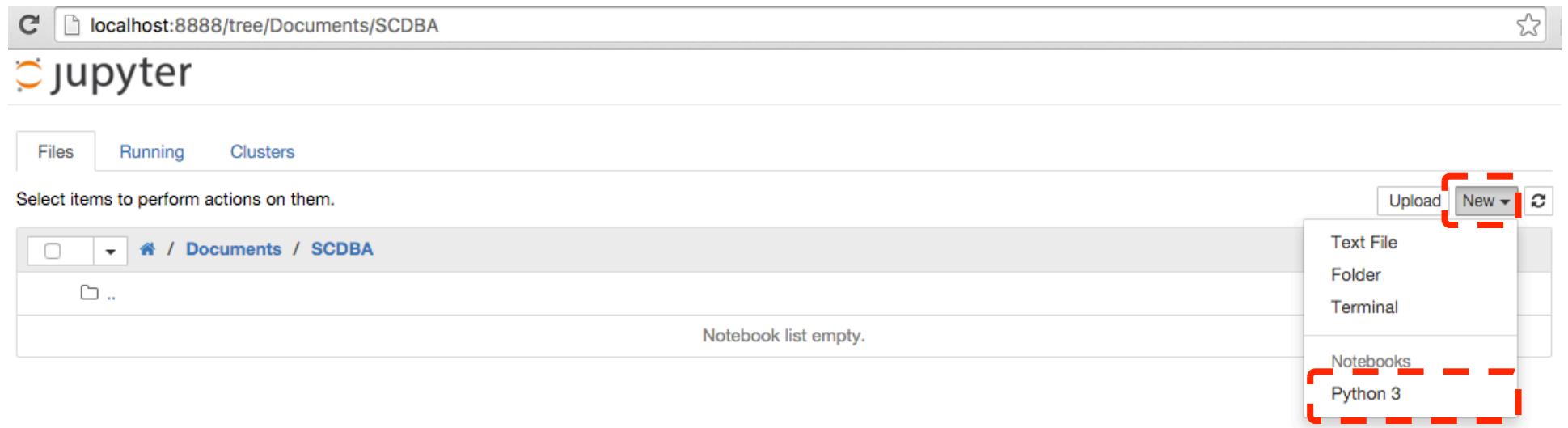
ipython notebook

ipython notebook

```
iMydaytekiMacBook-Pro:~ imyday$ ipython notebook
[I 14:26:49.944 NotebookApp] Serving notebooks from local directory: /Users/imyday
[I 14:26:49.944 NotebookApp] 0 active kernels
[I 14:26:49.944 NotebookApp] The Jupyter Notebook is running at: http://localhost:8888/
[I 14:26:49.944 NotebookApp] Use Control-C to stop this server and shut down all
kernels (twice to skip confirmation).
[W 14:26:56.639 NotebookApp] 404 GET /api/kernels/a87ab95b-6d6e-44d3-aaa7-c1901c
960677/channels?session_id=265FB16817FB4AB79202F6D3C3BDB0E6 (::1): Kernel does n
ot exist: a87ab95b-6d6e-44d3-aaa7-c1901c960677
[W 14:26:56.663 NotebookApp] 404 GET /api/kernels/a87ab95b-6d6e-44d3-aaa7-c1901c
960677/channels?session_id=265FB16817FB4AB79202F6D3C3BDB0E6 (::1) 95.43ms refere
r=None
[W 14:26:56.681 NotebookApp] 404 GET /api/kernels/b7fae9a6-d77b-4ead-832c-c070b1
8d642b/channels?session_id=EF4C761633E541C88568CDBCDE1091B7 (::1): Kernel does n
ot exist: b7fae9a6-d77b-4ead-832c-c070b18d642b
[W 14:26:56.683 NotebookApp] 404 GET /api/kernels/b7fae9a6-d77b-4ead-832c-c070b1
8d642b/channels?session_id=EF4C761633E541C88568CDBCDE1091B7 (::1) 6.62ms referer
=None
[W 14:27:29.595 NotebookApp] 404 GET /api/kernels/a87ab95b-6d6e-44d3-aaa7-c1901c
960677/channels?session_id=265FB16817FB4AB79202F6D3C3BDB0E6 (::1): Kernel does n
ot exist: a87ab95b-6d6e-44d3-aaa7-c1901c960677
[W 14:27:29.631 NotebookApp] 404 GET /api/kernels/a87ab95b-6d6e-44d3-aaa7-c1901c
```

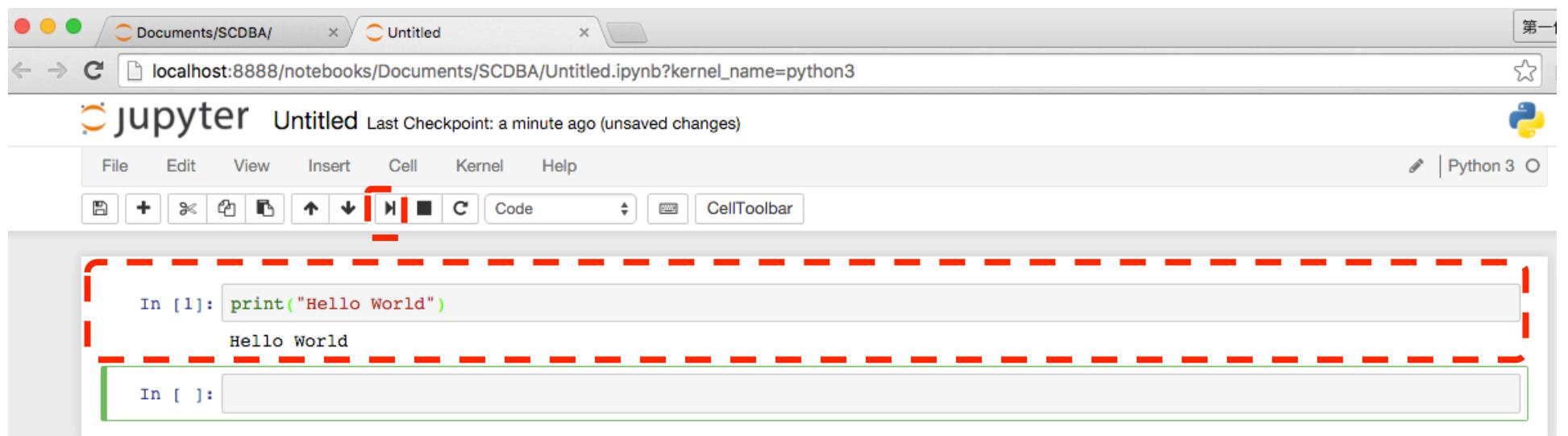
jupyter notebook

Python 3



jupyter notebook

Python 3



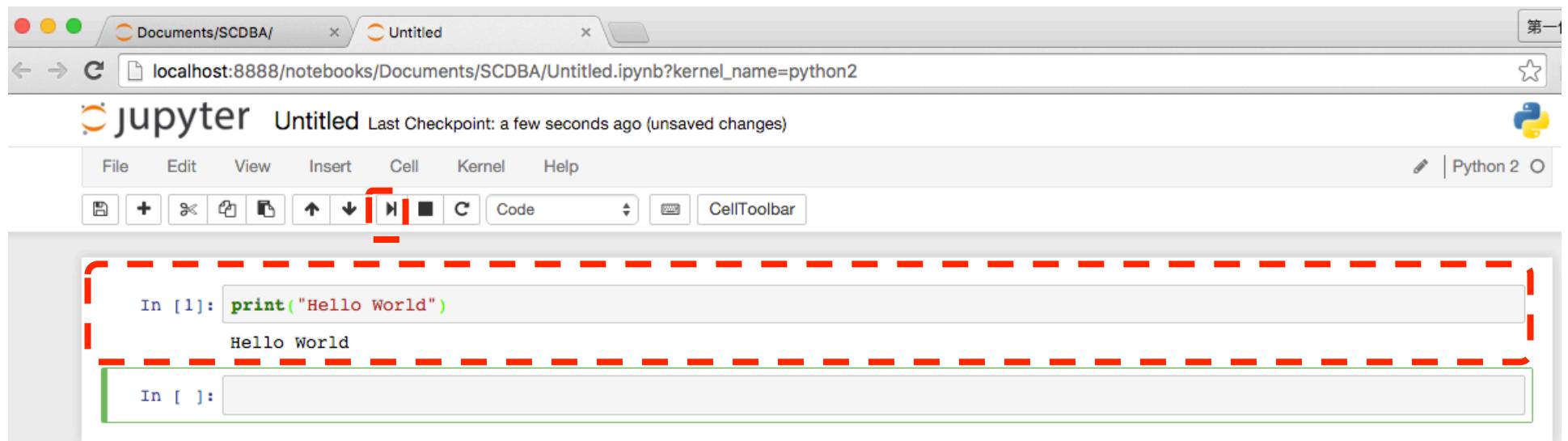
jupyter notebook

Python 2



jupyter notebook

Python 2



ipython notebook

jupyter notebook

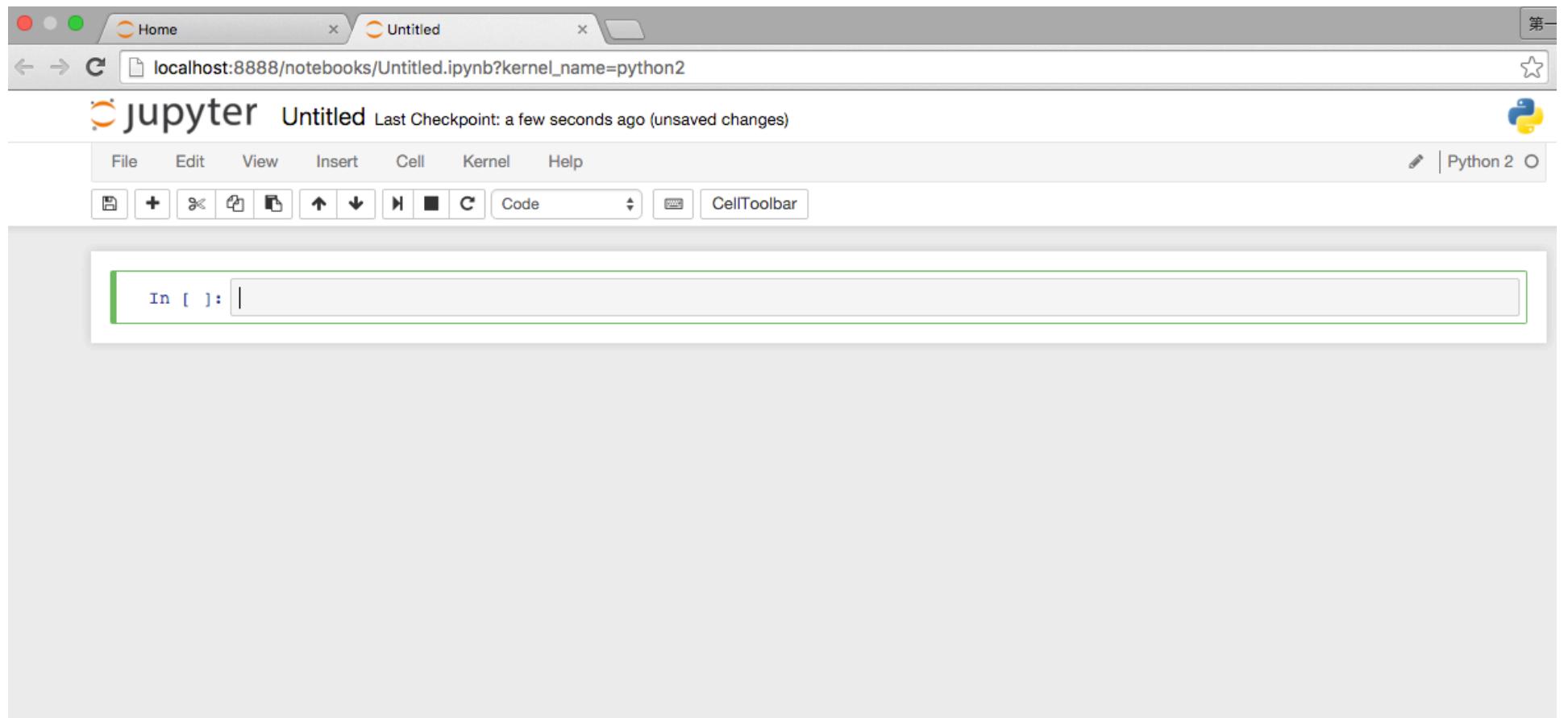
A screenshot of a web browser displaying a Jupyter Notebook interface. The address bar shows 'localhost:8888/tree'. The title bar has a logo and the word 'jupyter'. Below the title bar, there are tabs for 'Files' (which is selected), 'Running', and 'Clusters'. A message 'Select items to perform actions on them.' is displayed above a file tree. The file tree lists several directories: 'AndroidStudioProjects', 'app', 'Applications', 'AppsPro', 'bin', 'Desktop', 'Development', 'Documents', 'Downloads', 'Dropbox', 'imtkuapp5', 'jEdit', 'man', 'Movies', 'Music', 'OneDrive', and 'Pictures'. On the right side of the interface, there are buttons for 'Upload', 'New', and a refresh icon.

jupyter notebook

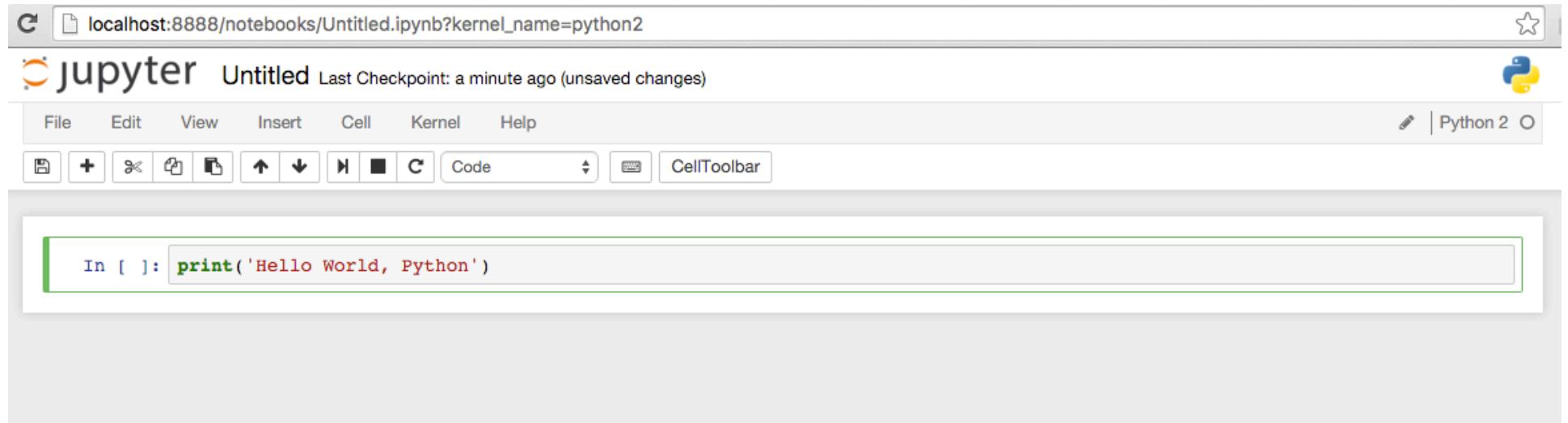
The screenshot shows a Jupyter Notebook interface running in a web browser. The title bar indicates the URL is `localhost:8888/tree`. The main area is titled "jupyter" and contains a sidebar with "Files", "Running", and "Clusters" tabs. A file tree view lists various local directories and files. On the right, there's a "New" button dropdown menu with options: "Text File", "Folder", "Terminal", "Notebooks", and "Python 2". The "Python 2" option is currently selected. Below the dropdown is a button labeled "Create a new notebook with Python 2". At the bottom of the screen, there's a footer bar with the URL "localhost:8888/tree#Large Scale Distributed Data Science using Jupyter Notebooks © James G. Shanahan" and a contact email "CONTACT.JAMES.SHANAHAN @ GMAIL.COM".

localhost:8888/tree#Large Scale Distributed Data Science using Jupyter Notebooks © James G. Shanahan CONTACT.JAMES.SHANAHAN @ GMAIL.COM

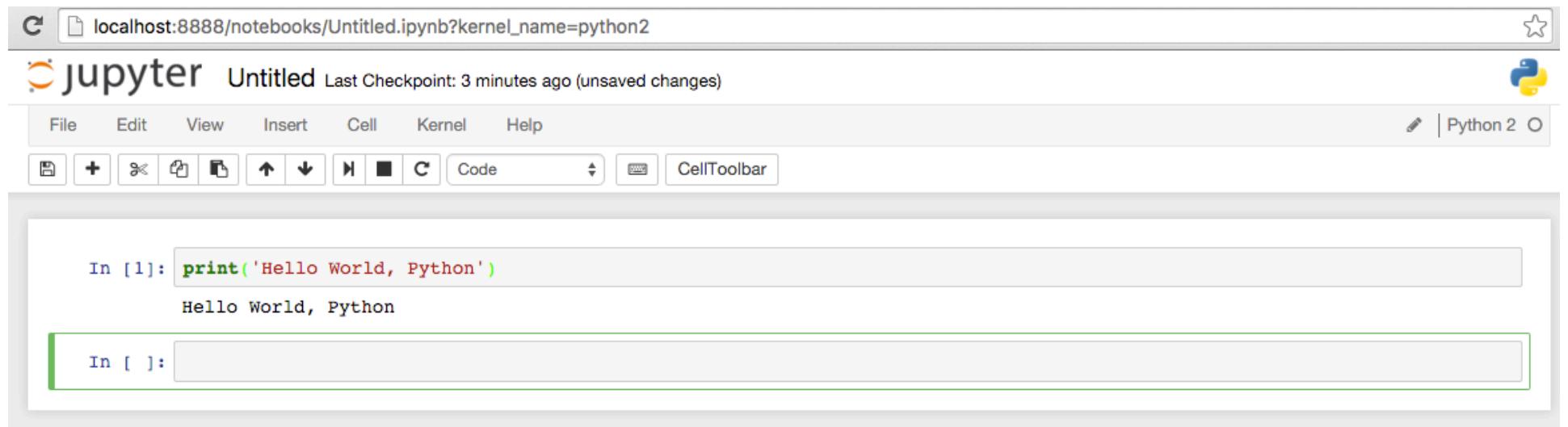
jupyter notebook



```
print('Hello World, Python')
```



```
print('Hello World, Python')
```



A screenshot of a Jupyter Notebook interface. The title bar shows the URL `localhost:8888/notebooks/Untitled.ipynb?kernel_name=python2`. The main window displays a single code cell:

```
In [1]: print('Hello World, Python')
Hello World, Python
```

The cell has a green border, indicating it is currently selected or being edited.

Rstudio Versus R in Jupyter

rmp · 7 months ago

the Jupyter notebook seems to be rather ignored by the larger R community. Any idea why that might be? Maybe its because Rmarkdown has been around for quite some time now and offers more options to write full reports and have more control over the output.

^ | v · Reply · Share ›

 **anarcho-chossid** → rmp · 6 months ago

Because Jupyter still doesn't provide the basic functionality in terms of working with code that RStudio does. It doesn't even do all of the syntax highlighting. (When I asked them for a way to change syntax highlighting for IRKernel, their response was basically "whatever".)

Also, while working in Jupyter is ok for quick prototyping, most of my time is spent building actual code for a pipeline. And RStudio is best/better for that.

Not to mention integration with Shiny.

^ | v · Reply · Share ›

 **Joris Meys** → rmp · 6 months ago

To my knowledge, big part of the R community isn't even fully aware of the capabilities of Jupyter. I also found out rather recently.

The reason for me to stay with RStudio for now, is simply because I use a lot more packages than the one mentioned there (including the Bioconductor set), and having them available is far less hassle in RStudio compared to jupyter.

^ | v · Reply · Share ›

-
- That's it
 - See you all on Monday, June 20 at 9AM