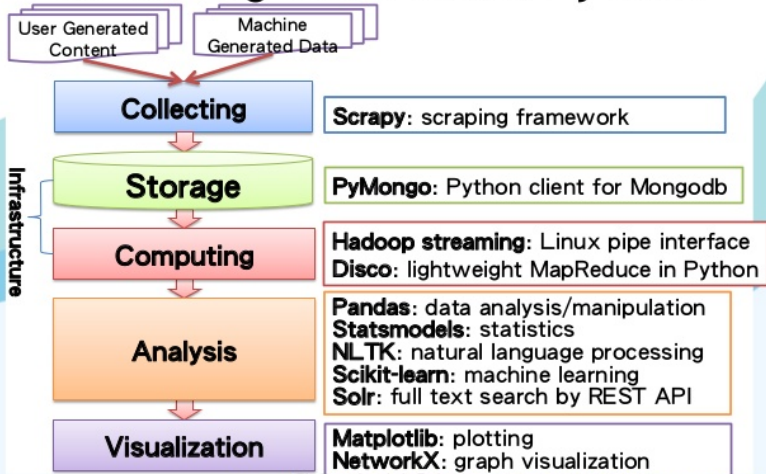


# Important Components of the Python Scientific Stack

# When Big Data meet Python



# Continuum Analytics Anaconda



Continuum Analytics

Welcome to Anaconda's doc...

docs.continuum.io/anaconda/

Anaconda 1.0 documentation »

next | modules | index

Table Of Contents

Welcome to Anaconda's documentation!

- Anaconda Pro Edition v. 1.0
- Anaconda Community Edition v. 1.0
  - Easy, Scalable Distributed Data Analysis in Python
- What's Included
- What's New?
- Package Documentation
- Packages included in Anaconda v.1.0
- Known Issues
- End User License Agreement
- Indices and tables

Next topic

Anaconda Install

This Page

Show Source

Quick search

Go

Enter search terms or a module, class or function name.

Welcome to Anaconda's documentation!

Contents

- Welcome to Anaconda's documentation!
  - Anaconda Pro Edition v. 1.0
  - Anaconda Community Edition v. 1.0
  - Package Documentation
  - Packages included in Anaconda v.1.0
  - Known Issues
  - End User License Agreement
  - Indices and tables

Anaconda Pro Edition v. 1.0

Anaconda Pro extends the easy to use Anaconda Community Edition with enterprise features that will enhance your ability to deal with large data-files, accelerate code that works on array-based data, configure execution environments for code, as well as provide access to cutting-edge algorithms and optimizations.

In addition to all the packages that come with Anaconda CE, Anaconda Pro includes

- IOPro — fast access to data-bases and text files
- NumbaPro — fast vectorization utilizing multiple cores and GPUs
- WiseRF — a fast, multi-core implementation of the Random Forest algorithm from Wise.IO
- Python Environments — the ability to create named "Python environments" to mix and match different versions of Python, NumPy, SciPy, etc. and easily switch between them

Purchase of Anaconda Pro gives you access to a year of free updates.

Download Anaconda Now!

# Continuum Analytics Anaconda

## **Anaconda**

- ▶ Anaconda, a free product of Continuum Analytics ([www.continuum.io](http://www.continuum.io)), is a virtually complete scientific stack (i.e. distribution) for Python.
- ▶ It includes both the core Python interpreter and standard libraries as well as most modules required for data analysis.

# Continuum Analytics Anaconda

## Anaconda

- ▶ Anaconda is free to use and modules for accelerating the performance of linear algebra on Intel processors using the **Math Kernel Library** (MKL) are available (free to academic users and for a small cost to non-academic users).
- ▶ Continuum Analytics also provides other high-performance modules for reading large data files or using the GPU to further accelerate performance for an additional, modest charge.

# Installing Anaconda

Most importantly, installation is extraordinarily easy on Windows, Linux and OS X. Anaconda is also simple to update to the latest version using

```
conda update conda  
conda update anaconda
```

# NumPy and SciPy

- ▶ **NumPy** provides a set of array and matrix data types which are essential for statistics and econometrics.
- ▶ **SciPy** contains a large number of routines needed for analysis of data. The most important include a wide range of random number generators, linear algebra routines and optimizers.
- ▶ Remark: SciPy depends on NumPy.
- ▶ More on them later.

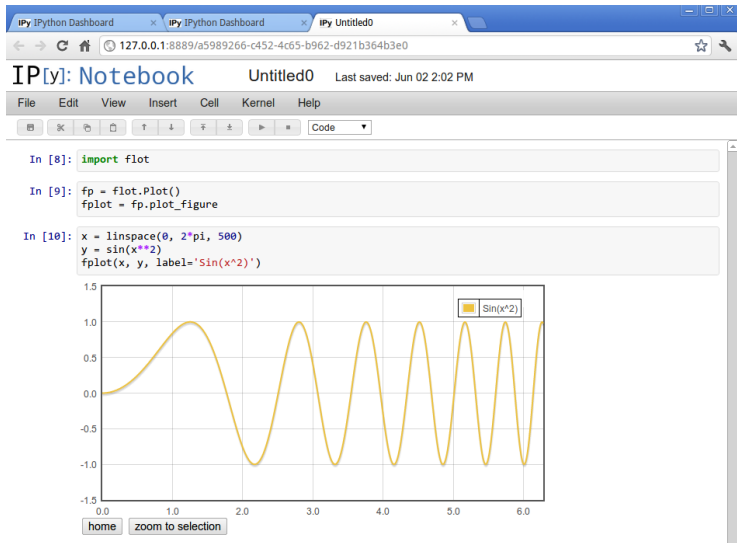
# IPython and IPython Notebooks

IPython provides an interactive Python environment which enhances productivity when developing code or performing interactive data analysis.

The IPython Notebook is a web-based interactive computational environment where you can combine code execution, text, mathematics, plots and rich media into a single document.



# IPython Notebook





## Evolved from the IPython Project

The language-agnostic parts of IPython are getting a new home in Project Jupyter

### IPython

- Interactive Python shell
- Python kernel for Jupyter
- Interactive Parallel Python

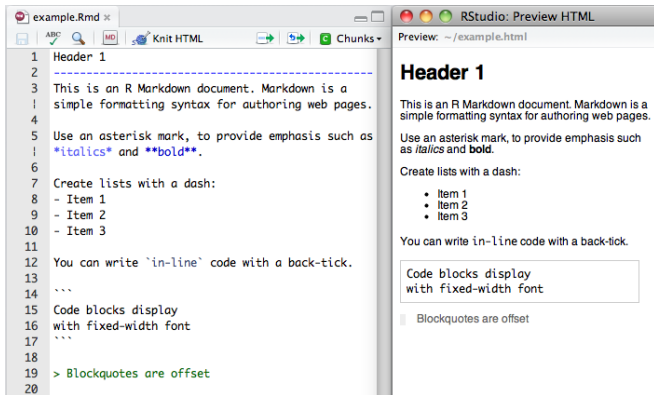
### Jupyter

- Rich REPL Protocol
- Notebook (format, environment, conversion)
- [JupyterHub](#) (multi-user notebook server)
- [More...](#)

# Markdown

Markdown is a text-to-HTML conversion tool for web writers.

Markdown allows you to write using an easy-to-read, easy-to-write plain text format, then convert it to structurally valid XHTML (or HTML).



The screenshot displays the RStudio interface with two panes. The left pane, titled 'example.Rmd', shows the source Markdown text. The right pane, titled 'RStudio: Preview HTML', shows the rendered HTML output.

**Source Markdown (example.Rmd):**

```
1 Header 1
2 -----
3 This is an R Markdown document. Markdown is a
4 | simple formatting syntax for authoring web pages.
5 Use an asterisk mark, to provide emphasis such as
6 | italics and bold.
7 Create lists with a dash:
8 - Item 1
9 - Item 2
10 - Item 3
11
12 You can write `in-line` code with a back-tick.
13 ```
14 Code blocks display
15 with fixed-width font
16 ```
17
18 > Blockquotes are offset
19
20
```

**Preview HTML (RStudio: Preview HTML):**

Preview: ~/example.html

## Header 1

This is an R Markdown document. Markdown is a simple formatting syntax for authoring web pages.

Use an asterisk mark, to provide emphasis such as *italics* and **bold**.

Create lists with a dash:

- Item 1
- Item 2
- Item 3

You can write in-line code with a back-tick.

```
Code blocks display
with fixed-width font
```

Blockquotes are offset

# matplotlib and seaborn

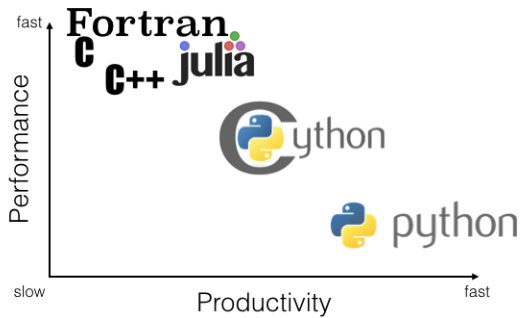
## Graphics Packages

- ▶ **matplotlib** provides a plotting environment for 2D plots, with limited support for 3D plotting.
- ▶ **seaborn** is a Python package that improves the default appearance of matplotlib plots without any additional code.

- ▶ *pandas* is a high-performance module that provides a comprehensive set of structures for working with data.
- ▶ *pandas* excels at handling structured data, such as data sets containing many variables, working with missing values and merging across multiple data sets.

# pandas

- ▶ While extremely useful, *pandas* is not an essential component of the Python scientific stack unlike NumPy, SciPy or matplotlib, and so while *pandas* doesn't make data analysis possible in Python, it makes it much easier.
- ▶ *pandas* also provides high-performance, robust methods for importing from and exporting to a wide range of formats.
- ▶ - example `read.csv()`





## Performance Modules : Cython and Numba

A number of modules are available to help with performance. These include Cython and Numba.

**Cython** Cython is a Python module which facilitates using a simple Python-derived creole to write functions that can be compiled to native (C code) Python extensions.

**Numba** Numba uses a method of just-in-time compilation to translate a subset of Python to native code using *Low-Level Virtual Machine* (LLVM).

# NumPy + Mamba = Numba

