

春からはじめる Python環境の作り方

辻 真吾 (@tsjshg)

みんなのPython勉強会#11

2016.4.12

自己紹介

- ・ 辻 真吾 (つじ しんご) 1975年生まれ
- ・ 東京大学先端科学技術研究センター ゲノムサイエンス分野
 - ・ 3/31に切れた任期が1年繋がりました！
- ・ 専門 (だと思ふ分野)
 - ・ 分子生物学と数理工学、Pythonとデータ解析とWeb
- ・ プログラミング経験
 - ・ MSX2でBASIC, Logo (小学生)
 - ・ C, C++ (大学生)
 - ・ Java (IT系ベンチャー勤務時代、大学院博士課程)
 - ・ Python (10年ちょっと前からほとんどこれ)

「みんなのPython勉強会」参加者20名限定

4/30

(土)まで!

Udemy人気講座

【世界で2万人が受講】実践 Python データサイエンス

30%OFFクーポン プレゼント

4,200円

—6,000円—

データ解析の基本、可視化、統計、機械学習などデータサイエンスに関するあらゆる実践的なスキルがPythonで身に付く！

学習内容

- 104本のレクチャーと17.5時間のコンテンツ！
- Pythonを使った基本的なプログラミング
- IPython notebookを使ったPythonプログラミング
- numpyを使ったベクトルや行列の操作
- pandasを使ったデータ処理
- JSON、HTML、Excelシートなどのデータ形式をPythonで扱う
- matplotlib、seabornを使ったデータの可視化
- 応用範囲の広いデータ解析のスキル

こんな方におすすめ

- Pythonプログラミングだけでなく、データサイエンスやデータの可視化に興味がある方々
- Pythonの経験は問いませんし、プログラミング初心者でも大丈夫です。
- 急速な広がりを見せる、データサイエンスの世界を覗いてみたい方々



講師
Shingo Tsuji

① WEBサイトにアクセス

Udemy

検索

<https://www.udemy.com/python-jp/?couponCode=spc-35D-apr>



② 会員登録

- ▶お名前
 - ▶メールアドレス
 - ▶パスワード
- で登録完了!

③ 講座のページから下記クーポンを入力して購入

クーポンコード：**spc-35D-apr**

※①のURLから入っていただければ、クーポン価格はすでに適用されています。そのため、クーポンコードの入力は不要です。もし表示が定価になっている場合は、「クーポンを利用する」から上記のクーポンコードを入力の上、ご購入ください。

¥4,200 ¥6,000 30% オフ

このコースを受講する

クーポンを利用する
無料プレビューを開始する
その他のオプション▼

Udemy日本語版が動作する最低必須環境につきましては、下記サポートページを参照ください。
<https://support.udemy.com/customer/ja/portal/articles/1494508>
※機器により、一部の機能が正常に動作しない可能性があります。 ※これ以外の機器でも表示できる場合があります。

個人情報の取り扱いについて
サービス利用にあたりご登録いただく個人情報はUdemy, Inc.のプライバシーポリシーに則り保護されます。詳しくは、<https://www.udemy.com/terms/privacy/>をご覧ください。

本日は、Python環境をどう作るかのお話

CONTINUUM[®]
ANALYTICS

ANACONDA

COMMUNITY

SERVICES

SOLUTIONS

ABOUT

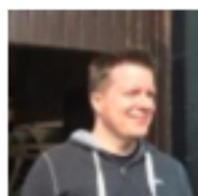
RESOURCES

 PYTHON PACKAGES AND ENVIRONMENTS WITH CONDA

DEVELOPER BLOG

Python Packages and Environments with conda

Posted Thursday, January 29, 2015



Bryan Van de Ven

If you had to pick one Python weakness...

For all the many strengths of Python, people often express frustration in finding, building, and installing third party packages. This pain can be especially acute with scientific and analytical libraries, which are often a mix of Python with compiled, platform-dependent C, C++, or Fortran code. One day, tools like [PyPy](#) and [Numba](#) may rescue us from this state of affairs, but data scientists working today need solutions—today. When we began building [Wakari](#)

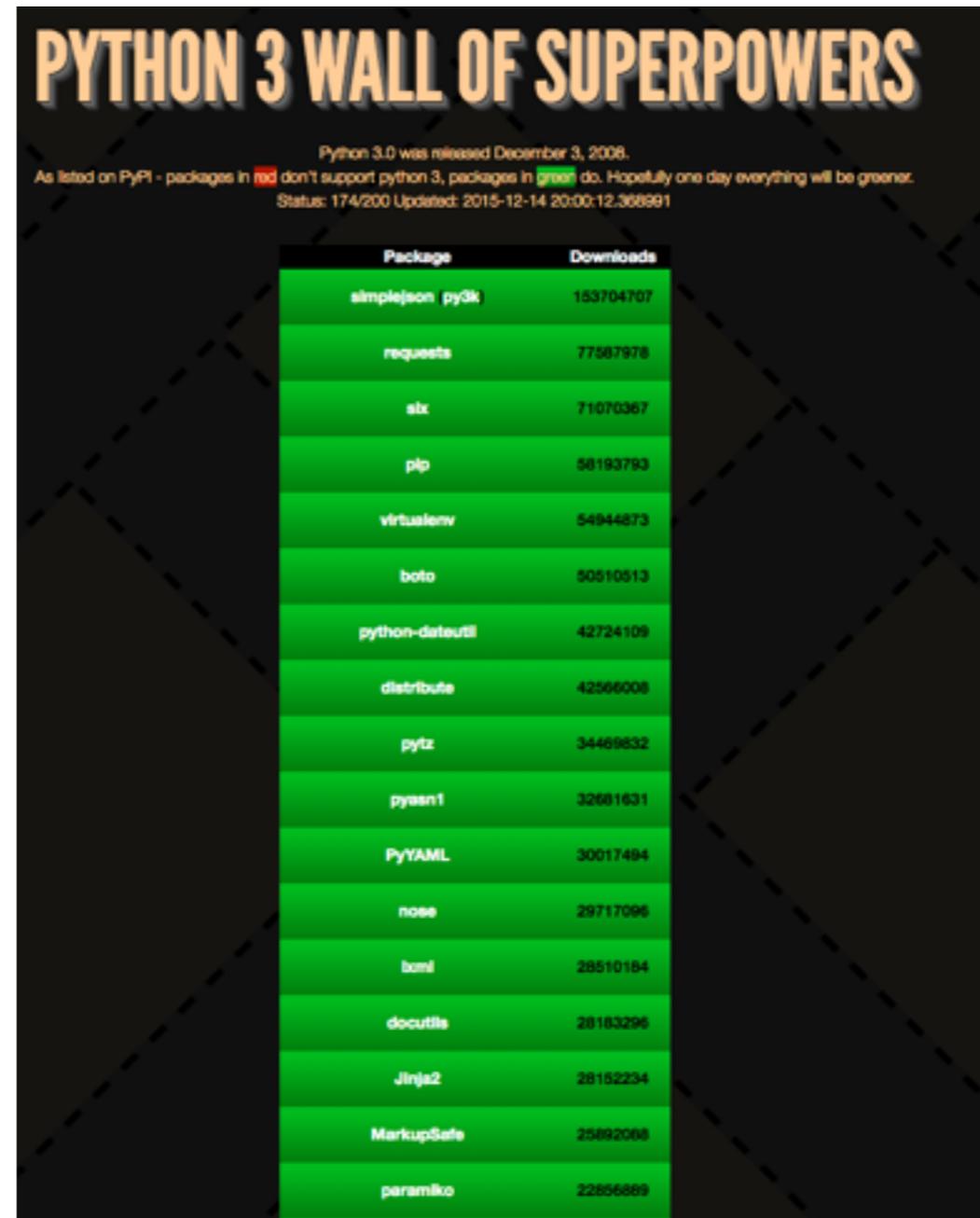
<https://www.continuum.io/content/python-packages-and-environments-conda>

おすすめ（結論）

- Linux, MacOSX, Windowsすべてのプラットフォームにおいて**Anaconda**
 - <https://www.continuum.io/downloads>
- データ解析をしない and/or 不必要なthird party packageいらないうなら、**Miniconda**
 - <http://conda.pydata.org/miniconda.html>

まずは基本

- ・ 2と3
- ・ 最新は2.7.11と3.5.1
- ・ まあ、3が良いような気がします
- ・ でも切り替えたいときはどうする？
- ・ 後ほど話題にします



実装の種類

- ・ 標準の実装 (CPython) www.python.org
- ・ Jython Javaによる実装 www.jython.org
- ・ IronPython .NET環境 ironpython.net
- ・ Cython 高速化の試み cython.org
- ・ 標準の実装を前提に話を進めます。

PACKAGE INDEX »

Browse packages
Package submission
List trove classifiers
List packages
RSS (latest 40 updates)
RSS (newest 40 packages)
Python 3 Packages
PyPI Tutorial
PyPI Security
PyPI Support
PyPI Bug Reports
PyPI Discussion
PyPI Developer Info

ABOUT »

NEWS »

DOCUMENTATION »

DOWNLOAD »

COMMUNITY »

FOUNDATION »

CORE DEVELOPMENT »

PyPI - the Python Package Index

The Python Package Index is a repository of software for the Python programming language. There are currently **78280** packages here. To contact the PyPI admins, please use the [Support](#) or [Bug reports](#) links.

Not Logged In

[Login](#)
[Register](#)
[Lost Login?](#)
Use [OpenID](#) 
[Login with Google](#) 

Status

[Nothing to report](#)

Get Packages

To use a package from this index either "pip install package" ([get pip](#)) or download, unpack and "python setup.py install" it.

Package Authors

Submit packages with "[python setup.py upload](#)". The index [hosts package docs](#). You may also use the [web form](#). You must [register](#). Testing? Use [testpypi](#).

Infrastructure

To interoperate with the index use the [JSON](#), [OAuth](#), [XML-RPC](#) or [HTTP](#) interfaces. Use [local mirroring or caching](#) to make installation more robust.

Updated	Package	Description
2016-04-10	s2scm 0.2.7	Compile Scheme to MIT Scratch
2016-04-10	django-svgselect 0.9.1	A Django form widget that uses SVG files in place of Select widgets.
2016-04-10	ongair-yowsup2 2.4.48.9	A WhatsApp python library
2016-04-10	psyplot_gui 0.0.1.dev1	Graphical user interface for the psyplot package
2016-04-10	zChainer 0.3.0	scikit-learn like interface and stacked autoencoder for chainer
2016-04-10	rowingdata 0.76.4	The rowingdata library to create colorful plots from CrewNerd, Painsled and other rowing data tools
2016-04-10	easy_tmpl 0.1.5	A CLI programm for operation with text templates
2016-04-10	media-manager 0.9.2	A personal media manager program
2016-04-10	gherkin-official 4.0.0	Gherkin parser (official, by Cucumber team)
2016-04-10	ravello-sdk 1.26	Python SDK for the Ravello API
2016-04-10	uncertainty_wrapper 0.2	Uncertainty wrapper using estimate Jacobian
2016-04-10	kpm 0.12.1	KPM cli

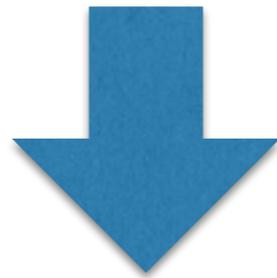
豊富な外部パッケージ

78,280 (2016年4月10日)

外部パッケージの追加方法

ファイルをダウンロードして解凍後

```
$ python setup.py install
```



lib/python3.5/site-packages
のようなディレクトリにインストールされる

実際はpipが便利

- ・ Python3.4から標準装備
- ・ PyPIから自動ダウンロード
- ・ 削除にも対応
- ・ 使い方
 - ・ `pip install -U django`
 - ・ `-U` or `—upgrade`で最新版を取得
 - ・ `pip freeze`
 - ・ いまの状態を表示

しかし・・・

- ・ CやFORTRANで書かれたライブラリがその場でコンパイルされることがある
 - ・ 開発環境を手軽に整えるのが難しいWindowsで問題になる
- ・ Unofficial Windows Binaries for Python Extension Packages
 - ・ その名の通り、バイナリを作って配布してくれているサイトがあったりする

condaが便利

- `$ conda install django`
- などのように外部パッケージのインストールに使える
- conda自体もPythonのパッケージ
- 開発元のContinuum Analytics社が色々面倒を見てくれていてすごいと思う

環境の使い分け

- ・ 2と3など、バージョンの使い分け
- ・ 同一バージョンを複数用意し、追加するパッケージの使い分け
- ・ 実は両方condaで対応可能

pyenv

- ・ 複数のバージョンのPythonを手軽に管理
- ・ pipなどを実行したときにそれぞれのバージョンの物が動くようになる
- ・ インストールがちょっと面倒
- ・ githubからもってくるか、Macならbrewコマンド

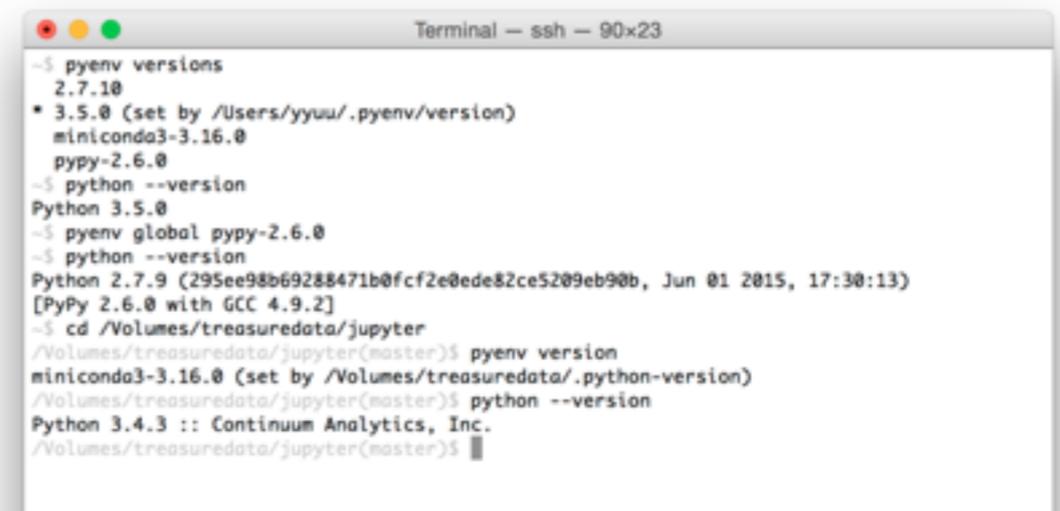
Simple Python Version Management: pyenv

gitter [join chat](#)

build [passing](#)

pyenv lets you easily switch between multiple versions of Python. It's simple, unobtrusive, and follows the UNIX single-purpose tools that do one thing well.

This project was forked from [rbenv](#) and [ruby-build](#), and modified for Python.



```
Terminal -- ssh -- 90x23
~$ pyenv versions
2.7.10
* 3.5.0 (set by /Users/yyuu/.pyenv/version)
  miniconda3-3.16.0
  pypy-2.6.0
~$ python --version
Python 3.5.0
~$ pyenv global pypy-2.6.0
~$ python --version
Python 2.7.9 (295ee98b69288471b0fcf2e0ede82ce5209eb90b, Jun 01 2015, 17:30:13)
[PyPy 2.6.0 with GCC 4.9.2]
~$ cd /Volumes/treasuredata/jupyter
~/Volumes/treasuredata/jupyter(master)$ pyenv version
miniconda3-3.16.0 (set by /Volumes/treasuredata/.python-version)
~/Volumes/treasuredata/jupyter(master)$ python --version
Python 3.4.3 :: Continuum Analytics, Inc.
~/Volumes/treasuredata/jupyter(master)$
```

pyvenv

- ・ 3.3から標準
- ・ 1つのPython環境に複数の仮想環境
- ・ インストールするパッケージを分けられる
- ・ 名前がややこしい
- ・ 同じことができる virtualenv もある

28.3. venv — 仮想環境の作成 (原文)

バージョン 3.3 で追加.

ソースコード: [Lib/venv](#)

`venv` モジュールは、軽量な“仮想環境”の作成のサポートを提供します。仮想環境には、仮想環境ごとの `site-packages` の仮想環境には、それ自身の Python バイナリ (様々な Python バージョンで環境を作成できます) があり、

Python仮想環境に関してより詳しくは [PEP 405](#) を参照してください。

28.3.1. 仮想環境の作成 (原文)

`pyvenv` スクリプトを実行することで `virtual environments` を作成出来ます

```
pyvenv /path/to/new/virtual/environment
```

このコマンドを実行すると、対象のディレクトリ (および必要なだけの存在していない親ディレクトリ) が作り出された Python のインストール場所を指します。このコマンドはまた、python バイナリのコピーを含む `bin` ディレクトリを作成します。さらに、`lib/pythonX.Y/site-packages` (Windows では `Lib\site-packages`) サブディレクトリを作成します。

参考: [Python Packaging User Guide: Creating and using virtual environments](#)

Windows では、関連する `PATH` および `PATHEXT` の設定をしていない場合、以下のように `pyvenv` スクリプトを実行する

両方condaでやる

- Python3.5のAnaconda (Miniconda) を入れて
- `conda create -n myenv`
 - 同じバージョンの別の環境
- `conda create -n mypy2env python=2`
 - 別のバージョンの環境
- `source activate myenv`で入って、`source deactivate`で抜ける

さらに

- condaとpipは共存可能
- condaに無くて、pipにあるパッケージはpipで入れればいい

まとめ

- Python+pip+conda
- Miniconda1つで全部揃う
- データサインスやるなら、
Anaconda
- Continuum Analytics社の回し者ではありませんが、すごいと思っています。



<https://www.continuum.io/>

ちなみに

- ・ Continuumと言うと、最近 Microsoftが発表した、WindowsスマホをPCの用に操作できる技術が同じ名前に



そんなMSも

- ubuntuがWindowsにのる！
- 結局全部Unix
- よかったと思います

Ubuntu on Windows – The Ubuntu Userspace for Windows Developers

By Dustin Kirkland on 30 March 2016



I'm in San Francisco this week, attending Microsoft's Build developer conference, as a sponsored guest of Microsoft





“コンピュータなんて役立たずさ。
ただ、答えを出すだけだる？”

– パブロ・ピカソ