

Python講習会(応用編)



あらすじ

1. 自己紹介
2. Pythonでの開発環境
3. IPython magicの紹介(以下はJupyterで)
4. 高速化Tips
5. ライブラリ紹介

あらすじ

1. 自己紹介
2. Pythonでの開発環境
3. IPython magicの紹介(以下はJupyterで)
4. 高速化Tips
5. ライブラリ紹介

今までのプログラミング経歴

中学：Javaをかじる

高校：C#やC,C++などをかじる(国名タイピングゲームなど)

大学：CとGnuplotでシミュレーション

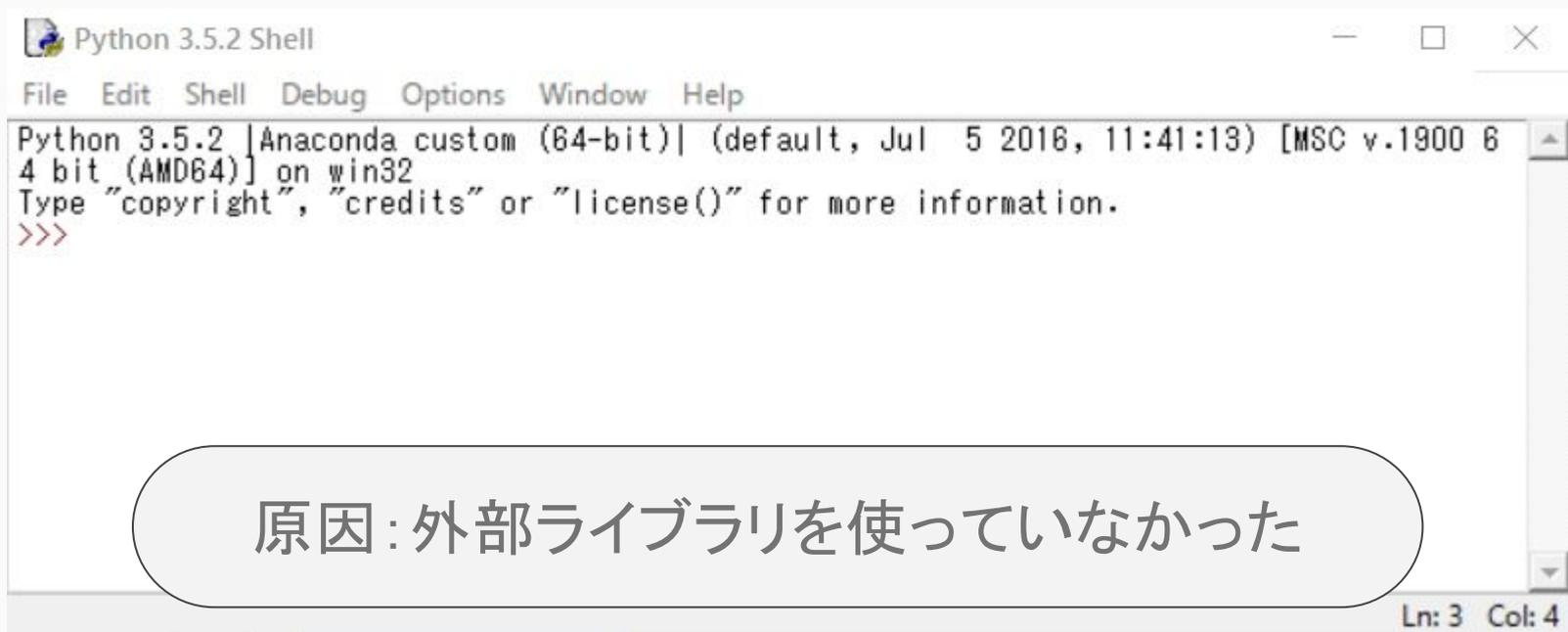
Node.jsを使ったWebサーバ

WebページでのJavascript、HTML

Pythonとの出会い



最初はあまりピンとこず



```
Python 3.5.2 Shell
File Edit Shell Debug Options Window Help
Python 3.5.2 [Anaconda custom (64-bit)] (default, Jul 5 2016, 11:41:13) [MSC v.1900 6
4 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> print()
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: print() takes exactly 1 argument (0 given)
```

原因: 外部ライブラリを使っていなかった

Ln: 3 Col: 4

2年前 : Numpy、Matplotlibの存在を知る

Numpy+Matplotlibを組み合わせることで
単一プログラム内で以下を同時に出来る

- シミュレーション
- 可視化



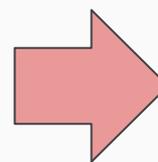
急激にハマリだす

～現在：Pythonの布教

研究でも趣味でもメインの言語に。

以下を全てPythonで完結できる

- Webページからのデータ収集自動化
- データを使いやすい形にする
- シミュレーションやデータ分析
- 可視化



得た知識を
共有したい！

あらすじ

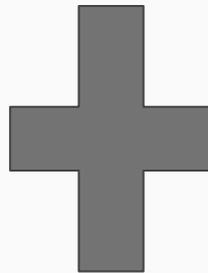
1. 自己紹介
2. Pythonでの開発環境
3. IPython magicの紹介(以下はJupyterで)
4. 高速化Tips
5. ライブラリ紹介

Pythonを使うための開発環境

そもそも開発環境とは・・・

コードを書き、実行する環境

例えば...



エディタ

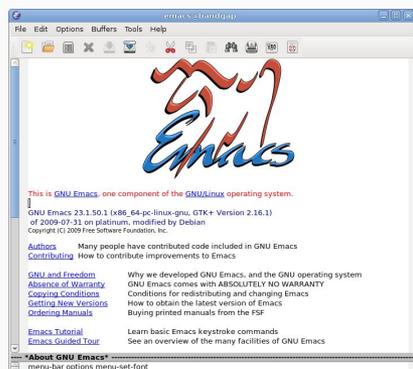
実行

Pythonを書けるエディタ



Sublime Text

自分にあったものを使いましょう



ATOM



etc.

Pythonでの実行環境

1. Pythonコンソール

```
C:¥>python
Python 3.5.2 |Anaconda custom (64-bit)| (default, Jul  5 2016, 11:41:13) [MSC v.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> print("hello, world")
hello, world
>>>
```

- 最も標準の実行環境
- 対話式（一行ずつ打ち込む）
- ファイルを引数として読み込むと上から順に実行
- 補完機能がないため（対話的には）使いづらい

Pythonでの実行環境

2. IPythonコンソール

```
C:\>ipython
Python 3.5.2 |Anaconda custom (64-bit)| (default, Jul 5 2016, 11:41:13) [MSC v.1900 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 5.1.0 -- 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]: print("helo, world")
helo, world

In [2]:
```

Pythonでの実行環境

2. IPythonコンソール

特徴

- Pythonコンソールを拡張している
- 強力な補完機能
- オブジェクトヘルプを容易に表示できる
- IPython magicコマンド

Pythonコンソールの**上位互換**

Pythonでの実行環境

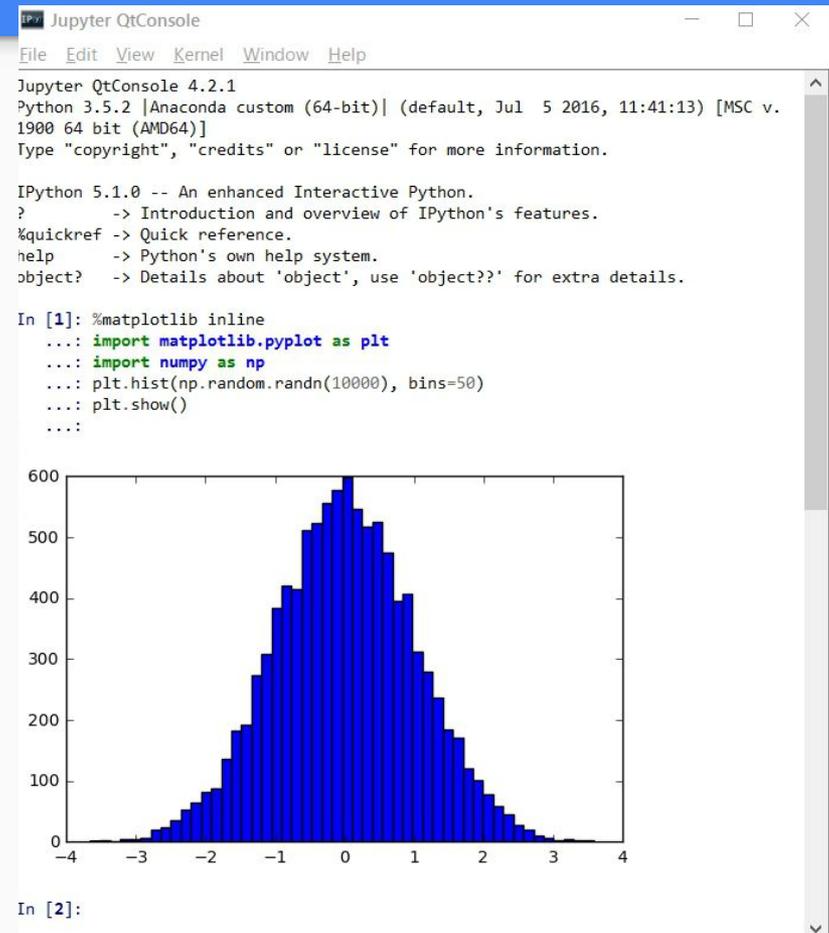
3. Jupyter Qtコンソール

もともと「IPython Qt Console」
と呼ばれていた。

> jupyter qtconsole

で実行可能

2. のIPythonコンソールを「Qt」
というGUIツールキットで表示し
ている



もっと強力な統合開発環境 (IDE)

IDE ≡ 「**エディタ**」+「**実行環境**」

- Spyder
- PyCharm
- Rodeo
- Python tools for Visual Studio
- Eclipse with PyDev

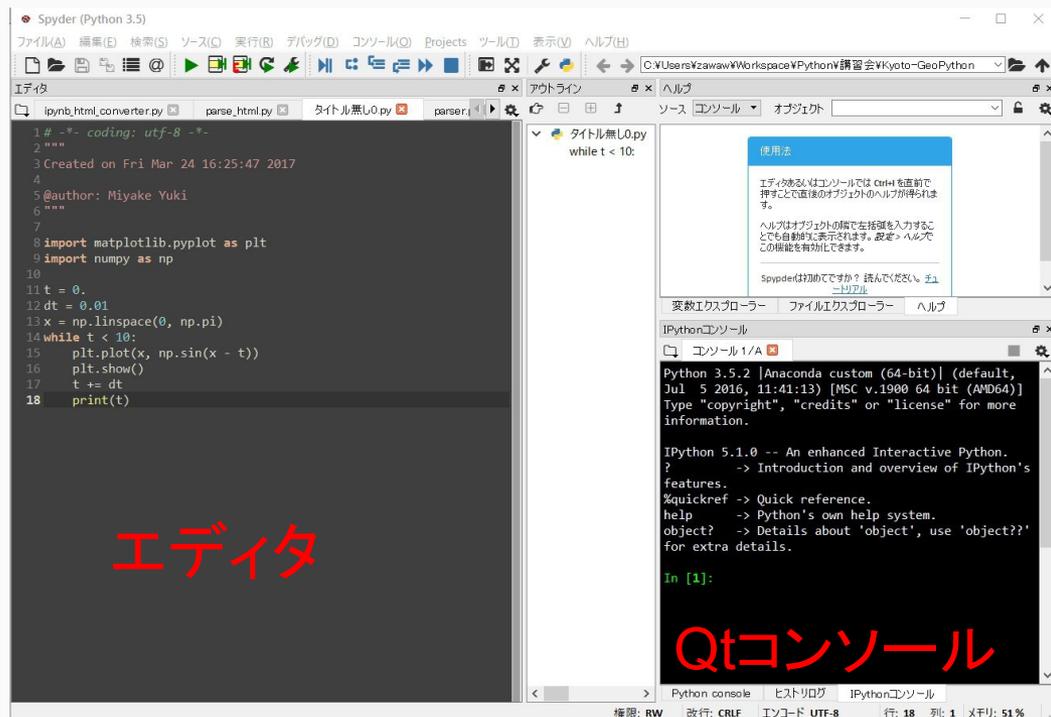
Spyder



- Qtコンソールで実行
- セル機能を持つ
- IDEの中では比較的軽量

インストール方法

>conda install spyder





- 強力な補完機能を持つIDE
- 高機能版は有料。ただし科学技術計算は無料版で十分。
- 選択部分をIPythonで実行できる
- やや重い？

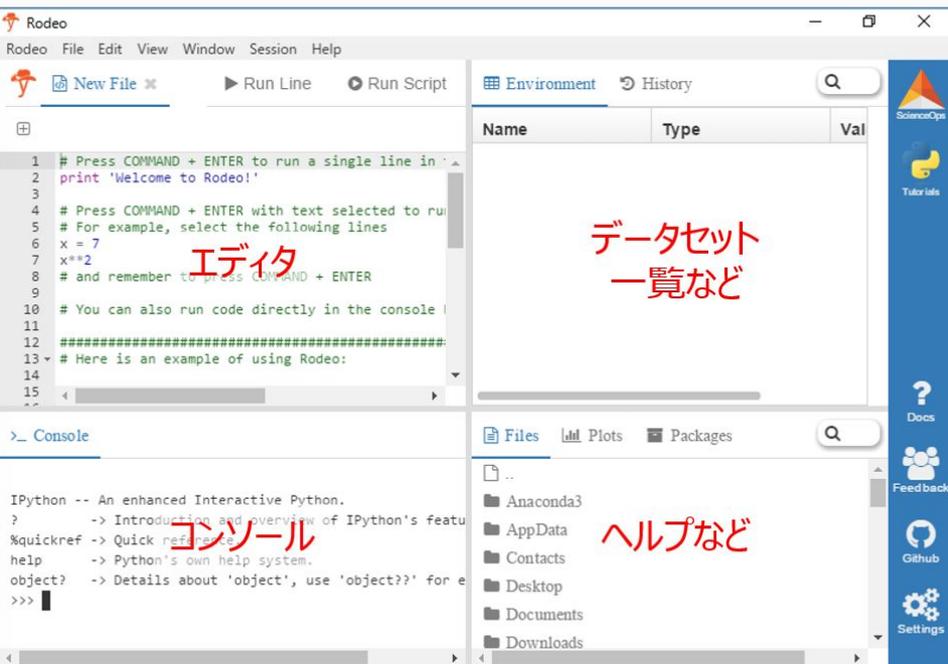
R使いの方へ



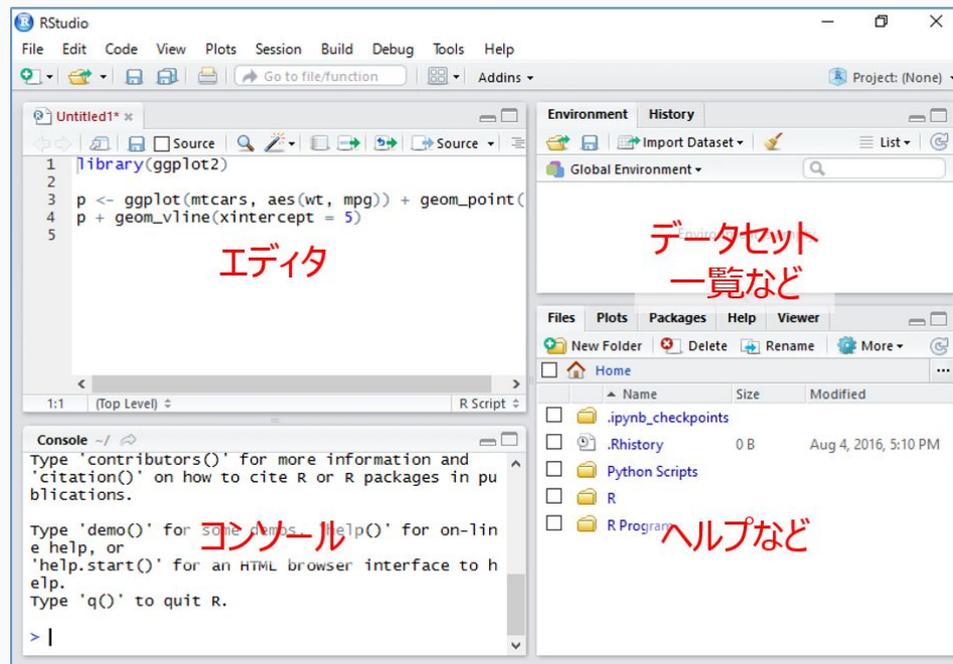
<http://pythondatascience.plavox.info/python%E3%81%AE%E9%96%8B%E7%99%BA%E7%92%B0%E5%A2%83/rodeo/> より

RのIDE「RStudio」に似たインターフェイス

Rodeo



RStudio





Jupyter Notebook

ブラウザ(!)がインターフェイス

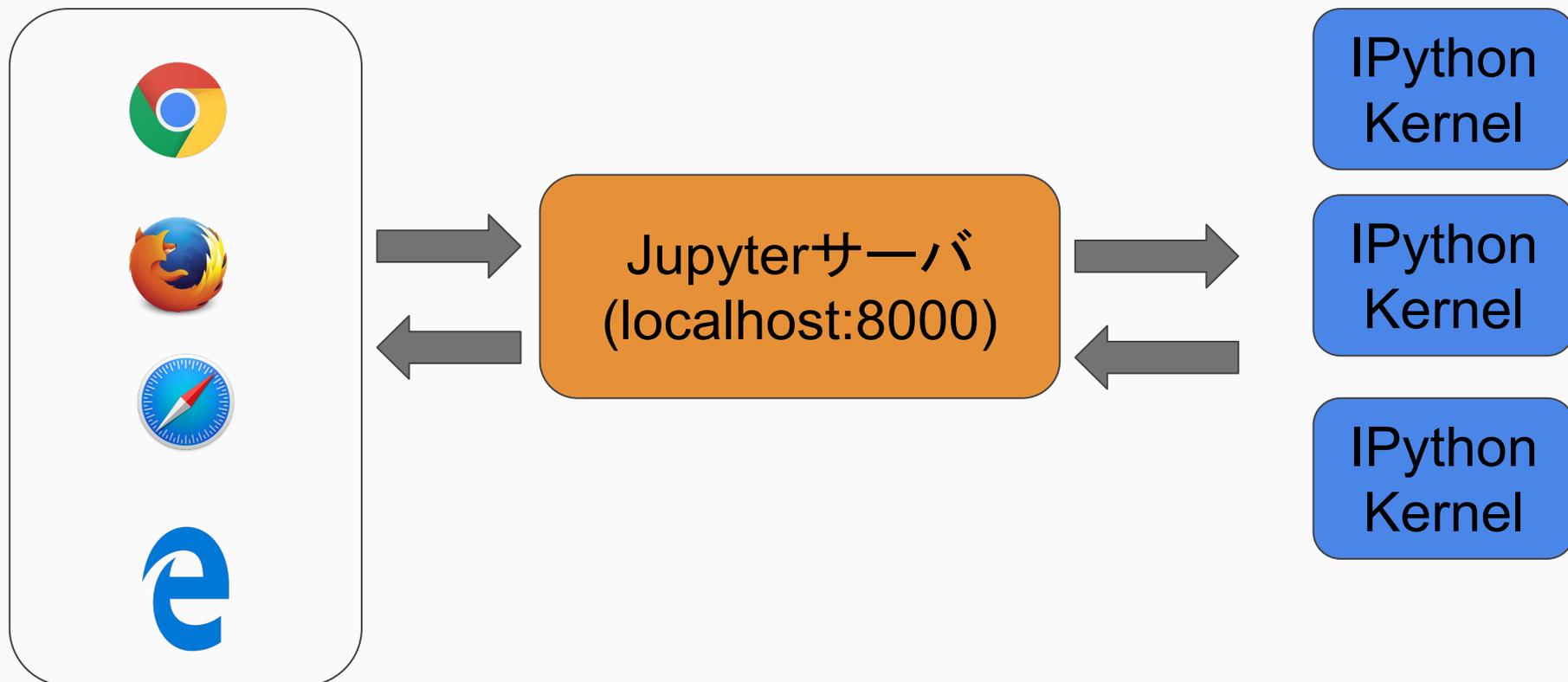
ローカルにサーバーを立てることで実現している

Python以外の言語にも対応 (Ruby, Julia, R, bash,...)

- インストール方法 :> `conda install jupyter`
- 実行 > `jupyter notebook`

Jupyter Notebookの仕組み

ブラウザ



とりあえず何使ったらいい？（一例）

- 迷ったらSpyderを使ってみることをオススメ
- Jupyter Notebookはとても便利
- GUIが開けない環境なら
 - IPythonコンソール＋エディタ