

Python勉強会@HACHINONE

第11章

プロットとクラス

お知らせ

Python勉強会@HACHINOHEでは、ジョン・V・グッターグ『Python言語によるプログラミングイントロダクション』近代科学社、2014年をみんなで勉強しています。

この本は自分で読んで考えて調べると力が付くように書かれています。

自分で読んで考えて調べる前に、このスライドを見るのは、いわば**ネタバレ**を聞かされるようなものでもったいないです。

是非、本を読んでからご覧ください。

PyLab

Python勉強会@HACHINOHE

- NumPy: Pythonの科学計算ライブラリ
- Matplotlib: PythonとNumPyのための描画ライブラリ
<https://matplotlib.org/>
- matplotlib.pyplot: Matplotlibのプロット・モジュール
- PyLab: matplotlib.pyplotとNumPyをまとめて読み込むモジュールで
MATLABユーザー向け。現在は非推奨

```
import pylab
```

PyLabで日本語を使用する設定

donec quis nunc

- 源ノ角ゴシックをインストール

- matplotlibrcの場所を探す

```
import matplotlib
matplotlib.matplotlib_fname()
```

- rc: run commands, run-control, running configuration
- Pythonのプログラムのフォルダにmatplotlibrcをコピー
- コピーしたmatplotlibrcのfont.sans-serifの先頭で源ノ角ゴシックを指定

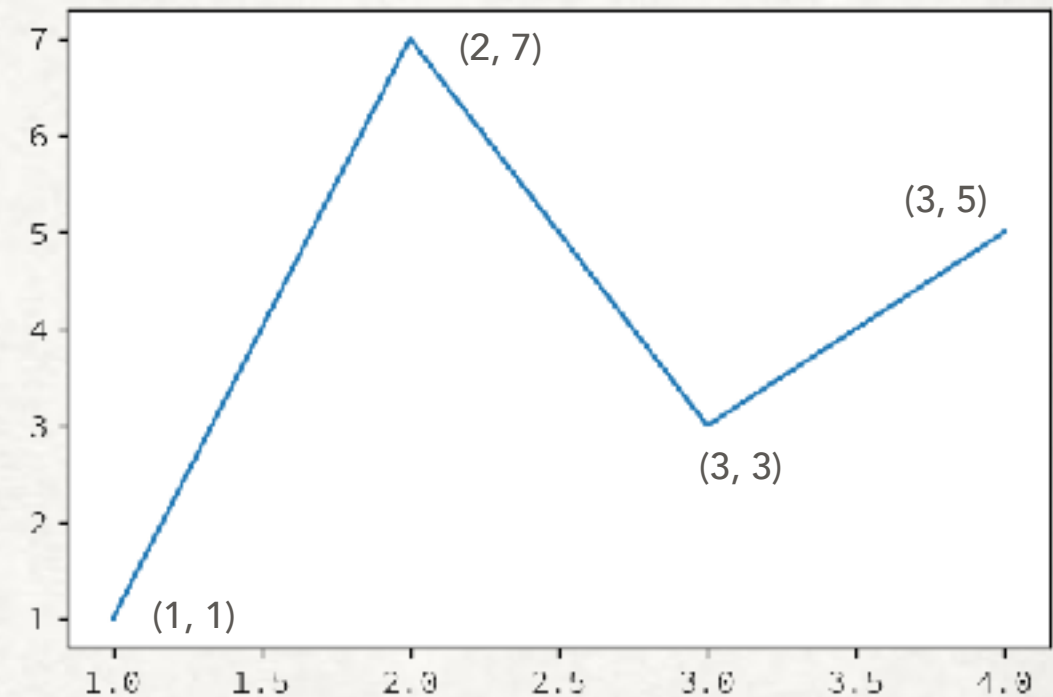
```
font.sans-serif      : Source Han Code JP, DejaVu Sans, ...
```


PyLabの例:1

Python勉強会@HACHINOHE

- p.159

```
# -*- coding: utf-8 -*-  
import pylab # pylabのインクルード  
pylab.figure(1) # figure 1の作成  
pylab.plot([1, 2, 3, 4], [1, 7, 3, 5]) # プロット  
#           xの値       yの値  
pylab.show() # 図を表示
```



PyLabの例:2

Python勉強会@HACHINOHE

- p.160-161

```
# -*- coding: utf-8 -*-
import pylab
pylab.figure(1)
pylab.plot([1, 2, 3, 4], [1, 7, 3, 5])
pylab.figure(2)
pylab.plot([1, 4, 2, 3], [5, 6, 7, 8])
pylab.savefig('Figure-Addie')
pylab.figure(1)
pylab.plot([5, 6, 10, 3])
pylab.savefig('Figure-Jane')
```

xはrange(len([5, 6, 10, 3]))

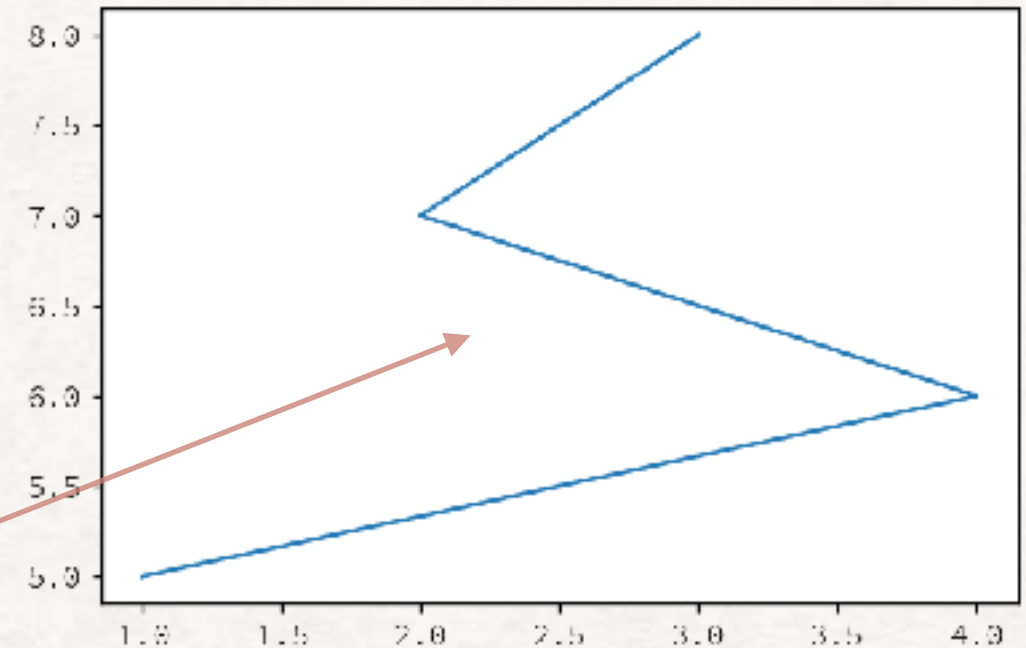


Figure-Addie

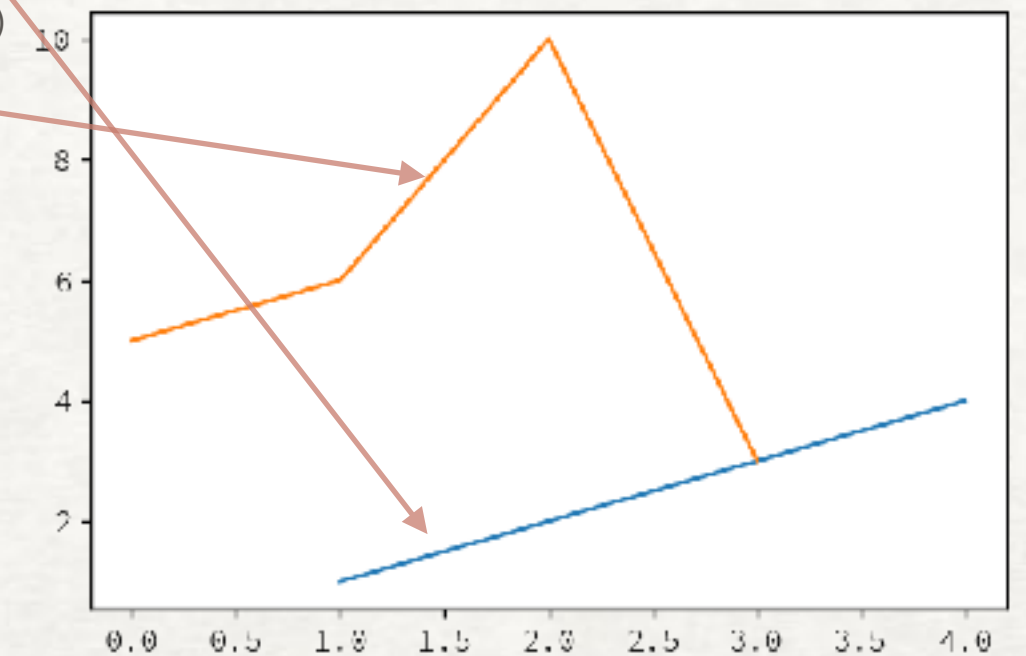


Figure-Jane

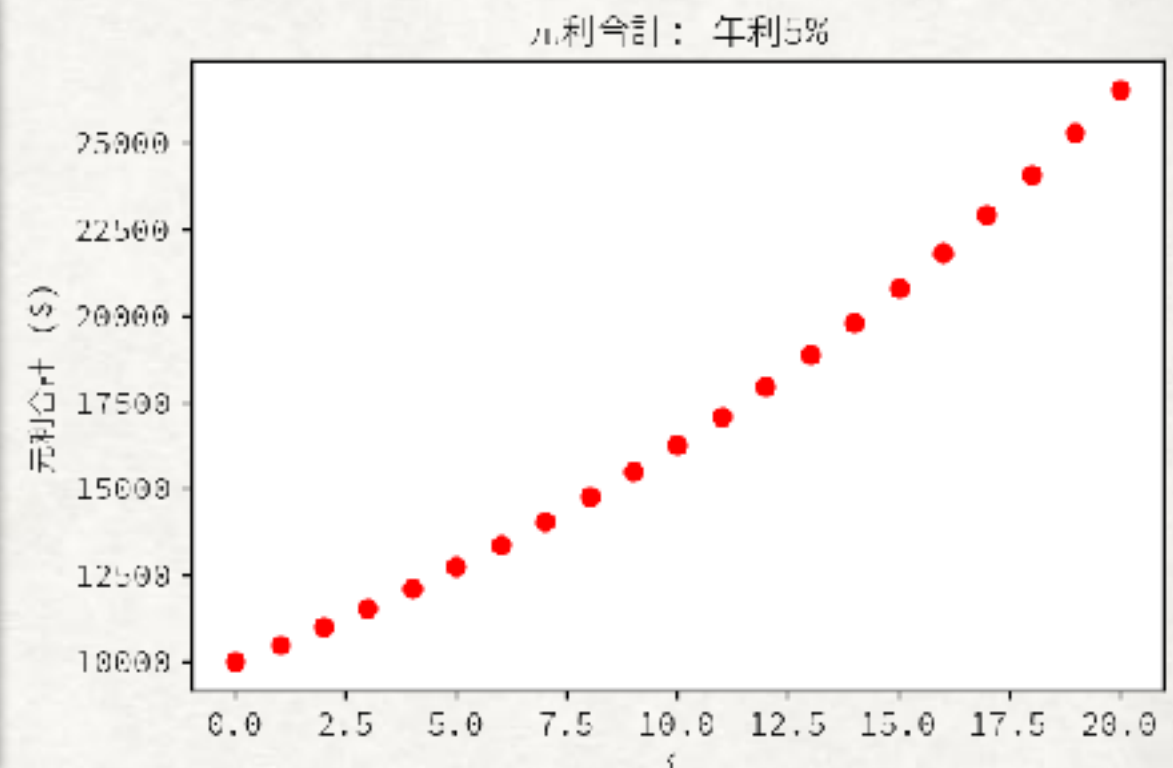
- figure(現行図)で「現行図」を指定

PyLabの例:3

Python勉強会@HACHINOHE

- p.161コード11.1

```
# -*- coding: utf-8 -*-
import pylab
principal = 10000 # 元本
ineterestRate = 0.05 # 利息
years = 20 # 年数
values = [] # 元利合計
for i in range(years + 1):
    values.append(principal)
    principal += principal * ineterestRate
pylab.plot(values, 'ro') # 赤丸
pylab.title(u'元利合計: 年利5%')
pylab.xlabel(u'年')
pylab.ylabel(u'元利合計 ($)')
pylab.show()
```



- 日本語表示

- 日本語を含んだ文字列は「u'文字列'」とする

PyLabの設定

Python勉強会@HACHINOHE

- デフォルト
- <https://matplotlib.org/users/customizing.html>

```
# 線の太さ
pylab.rcParams['lines.linewidth'] = 4
# タイトルの文字サイズ
pylab.rcParams['axes.titlesize'] = 20
# 軸のラベルの文字サイズ
pylab.rcParams['axes.labelwidth'] = 20
# x軸の目盛の文字サイズ
pylab.rcParams['xtick.labelsize'] = 16
# y軸の目盛の文字サイズ
pylab.rcParams['ytick.labelsize'] = 16
# x軸の目盛のサイズ
pylab.rcParams['xtick.major.size'] = 7
# y軸の目盛のサイズ
pylab.rcParams['ytick.major.size'] = 7
# マーカーのサイズ
pylab.rcParams['lines.markersize'] = 10
```


PyLabの凡例

Python勉強会@HACHINOHE

- `pylab.legend()`で指定

```
pylab.legend(loc = 'upper center', label = 'ラベル')
```

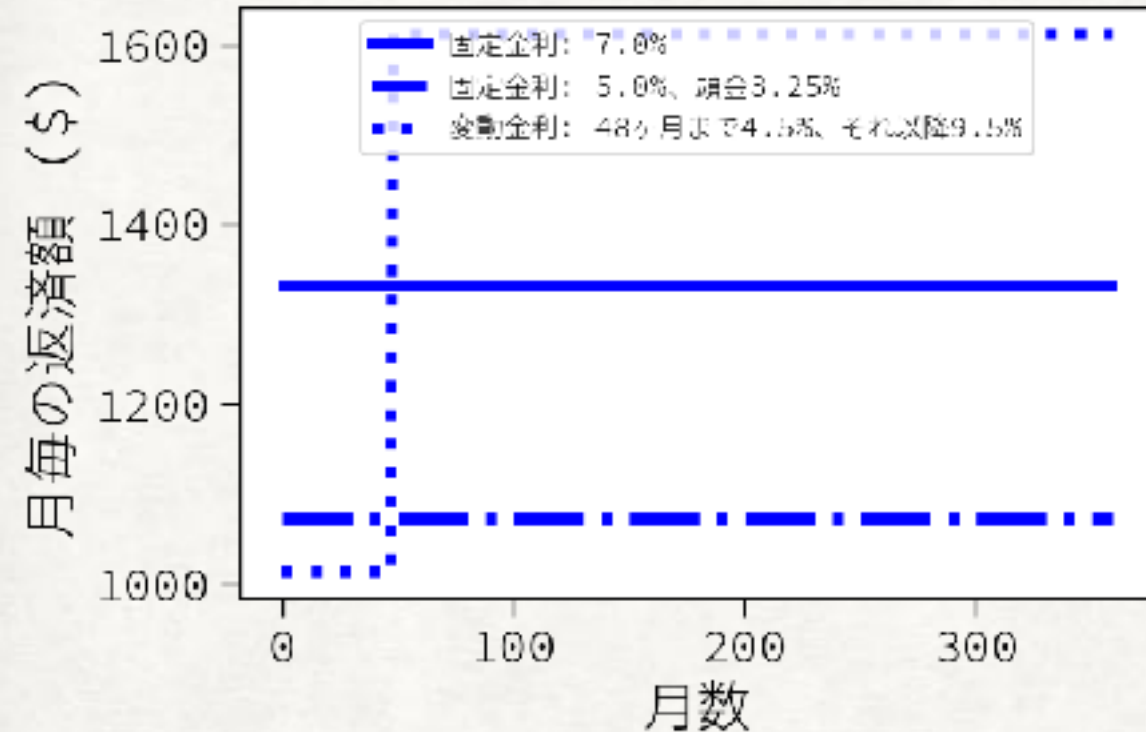
PyLab/NumPyの配列

Python勉強会@HACHINOHE

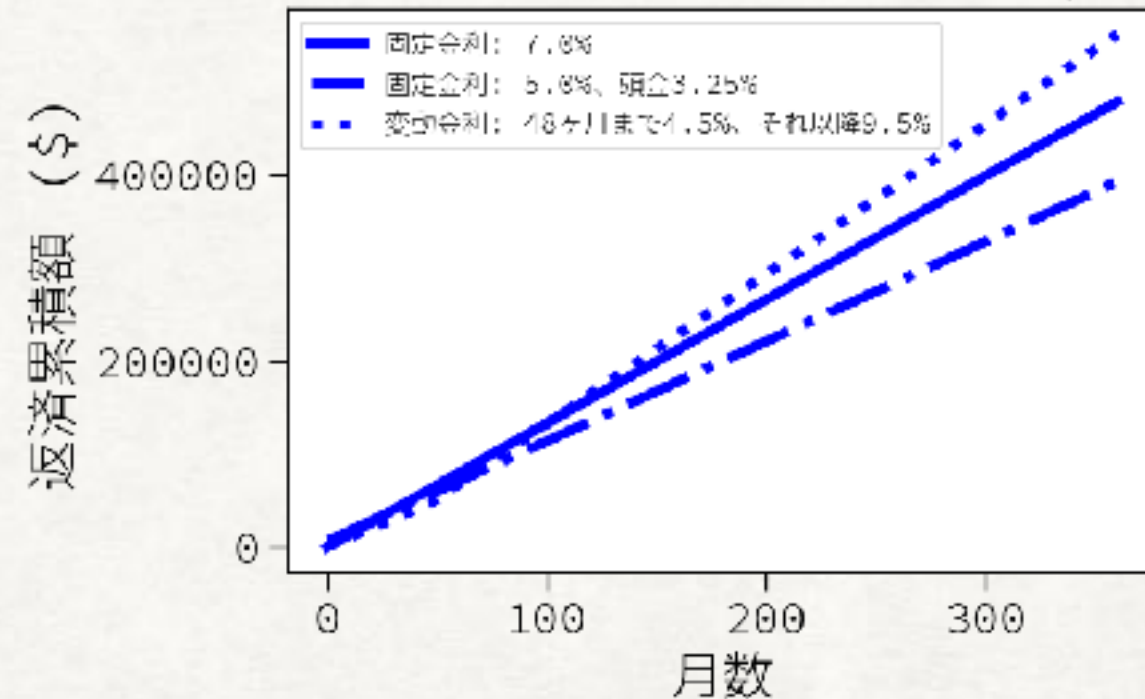
- PyLabの引数にリストで与えたデータは内部で配列に変換されている
- PyLabではNumPyの配列を使用できる

```
# -*- coding: utf-8 -*-
import pylab
a1 = pylab.array([1, 2, 4])
print 'a1 =', a1          # a1 = [1 2 4]
a2 = a1 * 2
print 'a2 =', a2          # a2 = [2 4 8]
print 'a1 + 3 =', a1 + 3  # a1 + 3 = [4 5 7]
print '3 - a1 =', 3 - a1  # 3 - a1 = [ 2  1 -1]
print 'a1 - a2 =', a1 - a2 # a1 - a2 = [-1 -2 -4]
print 'a1 * a2 =', a1 * a2 # a1 * a2 = [ 2  8 32]
```

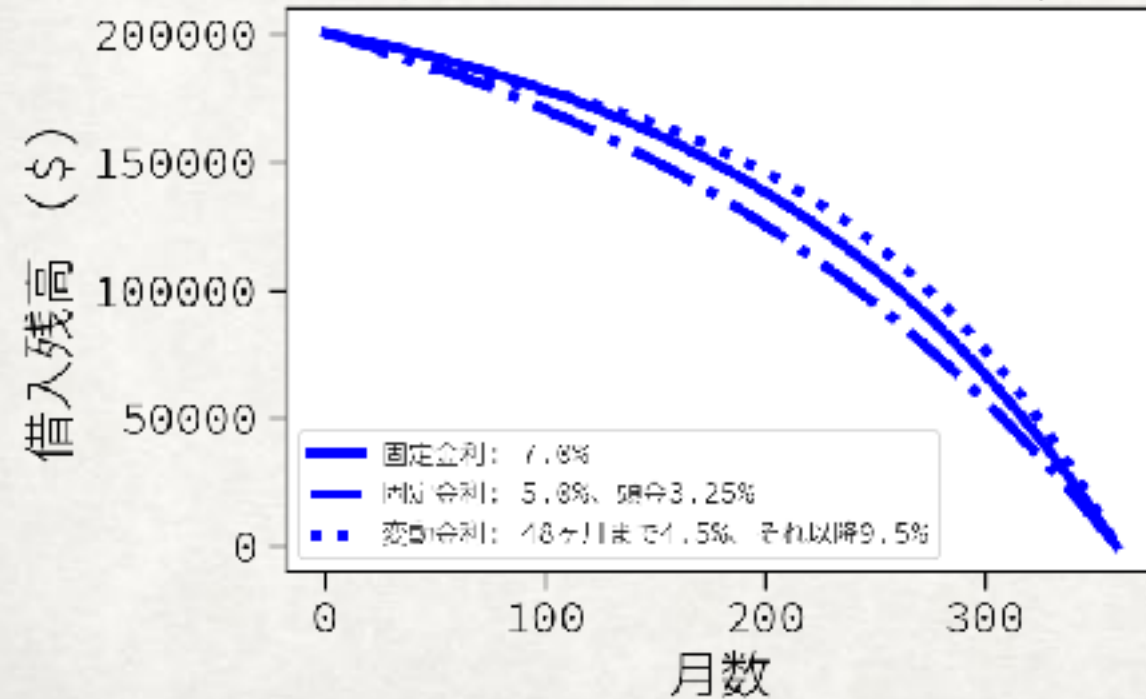
月毎の返済額： 借入金2000000\$



返済累積額： 借入金2000000\$



借入残高： 借入金2000000\$



総経費： 借入金2000000\$

