

# 1 このファイルは何?

Pythonのnumpyとmatplotlibの導入の仕方を記述したファイルです.

`pip` コマンドを使っていますが, `pip` について知りたい場合は同階層の他のファイルに大雑把にまとめたのでそちらをご覧ください.

## 2 numpy

### numpyとは

pythonの算術系のライブラリです.

行列計算とか, 近似式とか, まあいろいろできます.

ちなみに読み方は「なむばい」です.

### numpyの導入

コマンドプロンプトで以下のコマンドを入力します.

何処の階層にいても大丈夫です.

```
$ pip install numpy
```

いろいろ文字が出てきて終了したら, pythonでインポートできるか試しましょう.

```
C:\>python
Python 3.7.3 (v3.7.3:ef4ec6ed12, Mar 25 2019, 22:22:05) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import numpy as np
>>> np.pi
3.141592653589793
>>> exit()
C:\>
```

## 3 matplotlib

### matplotlibとは

pythonでグラフを描画するといえばこのパッケージ.

結構いろいろできます。

MATLABをリスペクトして作っているらしいです。名前も似てるもんね。

読み方は「まっとぷろっとりぶ」です。「mpl」と略すことも？

(ちなみに自分は「まっとぷろっとらいぶらり」と呼んでます。)

## matplotlibの導入

```
$ pip install matplotlib
```

確認方法は `pip list` でも別にいいんですが、GUIがきちんと機能するか微妙なので、以下のプログラムを任意のファイルにコピーして実行してみてください。

```
import numpy as np
from matplotlib import pyplot as plt

def main():

    fig = plt.figure()
    ax = fig.add_subplot(111)

    # あるべき曲線
    x_lim = [0, 3]
    x = np.linspace(*x_lim, 300)
    y = 2*(x**2) + 4*x + 3
    ax.plot(x, y, label="org", color="C0")

    # 実験値の描画
    y += np.random.normal(1, 5, len(x)) + np.random.normal(-1, 2, len(x))
    ax.plot(x, y, label="exp", alpha=0.4, color="C2")

    # 最小二乗法の描画
    p = np.polyfit(x, y, 2)
    y = p[0]*(x**2) + p[1]*x + p[2]
    ax.plot(x, y, label="LSM", color="C1")

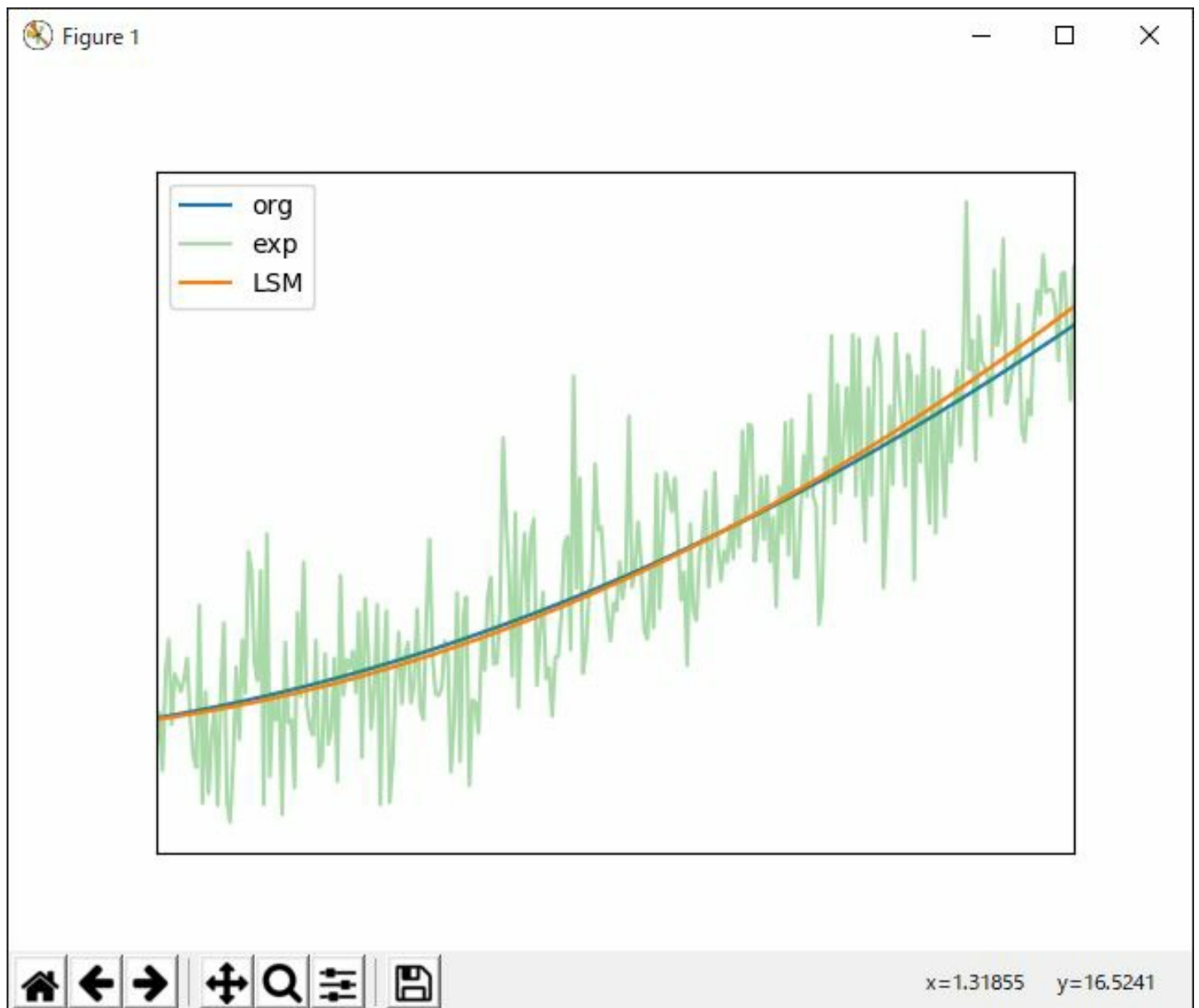
    # 図の設定
    ax.set_xticks([])
    ax.set_yticks([])
    ax.set_xlim(x_lim)
    ax.legend()
```

```
# 図の表示  
plt.show()
```

```
if __name__ == '__main__':  
    main()
```

以下のように実行すれば, グラフが描画されるはずです.

```
$ python [ファイル名]
```



## 4 mplの日本語

デフォルトのフォントが **DejaVu Sans** という日本語非対応のフォントなので, 素直に日本語を出力使用とすると, 文字化けして

しまいます。

そのため、以下のようなコマンドで自身のPCにあるフォントの一覧を出し、気に入ったものを設定してください。

もし、フォントの雰囲気を見たい場合は、 **コントロールパネル → デスクトップのカスタマイズ → フォント** で一覧を出すことができます。

```
fonts = set([fm.FontProperties(fname=font).get_name() for font in fm.findSystemFonts()])
fonts = sorted(fonts)
for f in fonts:
    print(f)
```

気に入ったフォントを見つけ次第、以下のようなコマンドを各プログラム内に埋め込めば、日本語が出力されるようになります。

```
import matplotlib as mpl

font = {"family": "フォント名(英語)"}
mpl.rc('font', **font)
```

例

```
import numpy as np
from matplotlib import pyplot as plt
import matplotlib as mpl

def main():
    font = {"family": "Yu Gothic"}
    mpl.rc('font', **font)

    fig = plt.figure()
    ax = fig.add_subplot(111)

    # あるべき曲線
    x_lim = [0, 3]
    x = np.linspace(*x_lim, 300)
    y = 2*(x**2) + 4*x + 3
    ax.plot(x, y, label="本来", color="C0")

    # 実験値の描画
    y += np.random.normal(1, 5, len(x)) + np.random.normal(-1, 2, len(x))
    ax.plot(x, y, label="実験値", alpha=0.4, color="C2")

    # 最小二乗法の描画
    p = np.polyfit(x, y, 2)
```

```

y = p[0]*(x**2) + p[1]*x + p[2]
ax.plot(x, y, label="最小二乗法", color="C1")

# 図の設定
ax.set_xticks([])
ax.set_yticks([])
ax.set_xlim(x_lim)
ax.legend()

# 図の表示
plt.show()

if __name__ == '__main__':
    main()

```

