

MA (主成分回帰)

青木繁伸

1 目的

Major Axis regression (主成分回帰) を行う。

2 使用法

```
from Major_Axis_regression import Major_Axis_regression
Major_Axis_regression(x, y, sig=0.95, verbose=True)
```

2.1 引数

x	データベクトル (リストでもよい)
y	データベクトル (リストでもよい)
sig	信頼率 [デフォルトは 0.95]
verbose	必要最小限のプリント出力をする

2.2 戻り値の名前

"intercept"	切片
"slope"	傾き
"intercept CL"	bootstlap を 2 以上にしたときに, 切片の信頼限界
"slope CL"	bootstlap を 2 以上にしたときに, 傾きの信頼限界

3 使用例

引数はリスト型で与えてもよい。

```
y = [61, 37, 65, 69, 54, 93, 87, 89, 100, 90, 97]
x = [14, 17, 24, 25, 27, 33, 34, 37, 40, 41, 42]

import sys
sys.path.append("statlib")
from Major_Axis_regression import Major_Axis_regression

a = Major_Axis_regression(x, y)
```

	Estimate	2.5%	97.5%
Intercept	6.656633	NaN	NaN
Slope	2.301728	1.598351	3.74939

```
x = [1,2,3,4,5]
y = [3,2,1,4,6]
a = Major_Axis_regression(x, y)
```

```
          Estimate  2.5%  97.5%
Intercept -0.832092   NaN   NaN
Slope      1.344031   NaN   NaN
```

```
print("slope =", a["slope"])
```

```
slope = 1.3440306508910553
```

```
print("intercept =", a["intercept"])
```

```
intercept = -0.8320919526731663
```

```
import scipy as sp

def simple_reg(x, y):
    mx = sp.mean(x)
    my = sp.mean(y)
    slope = sum((x-mx)*(y-my))/sum((x-mx)**2)
    intercept = my-slope*mx
    return intercept, slope
```

```
import matplotlib.pyplot as plt

intercept = a["intercept"]
slope = a["slope"]
x0 = sp.amin(x)
x1 = sp.amax(x)
x2 = sp.array([x0, x1])
y2 = intercept + slope * x2
intercept_reg, slope_reg = simple_reg(x, y)
y3 = intercept_reg + slope_reg * x2
plt.scatter(x, y, c="black", s=9)
plt.plot(x2, y2, label="MA regression", linewidth=0.5, color="red")
plt.plot(x2, y3, label="simple regression", linewidth=0.5, color="
    black")
plt.xlabel("x")
plt.ylabel("y")
plt.legend()
plt.show()
```

