

独立 2 標本の並べ替え検定

青木繁伸

1 目的

独立 2 標本データに対する並べ替え検定 (permutation test) を行う。

2 使用法

```
from permutation_test import permutation_test
permutation_test(x, y, FUNC, verbose=True)
```

2.1 引数

<code>x</code>	第 1 群のデータ
<code>y</code>	第 2 群のデータ
<code>FUNC</code>	検定統計量を返す任意の検定関数。既存の関数でも自作関数でもよいが、検定統計量を戻り値で返すもの。
<code>verbose</code>	必要最小限のプリント出力をする

2.2 戻り値

p 値

3 使用例

```
x = [2, 1, 3, 2, 5]
y = [4, 3, 2, 5, 4]

# van der Waerden 検定
import scipy as sp
from scipy.stats import norm, rankdata

def vdw2_test(x, y): # 第二群のデータベクトル
    n1 = len(x) # 第一群のサンプルサイズ
    z = sp.hstack((x, y)) # データベクトルを一つにまとめる
    n = len(z) # 合計したサンプルサイズ
    S = sum(norm.ppf((rankdata(z)[:n1])/(n+1))) # 第一群のデータに対する
    正規化得点の合計
    return S

import sys
```

```
sys.path.append("statlib")
from permutation_test import permutation_test

a = permutation_test(x, y, vdw2_test)
```

p value = 0.3413

```
import scipy as sp

def t_test(x, y):
    x = sp.ravel(x)
    y = sp.ravel(y)
    n_a = len(x)
    n_b = len(y)
    mean_a = sp.mean(x)
    mean_b = sp.mean(y)
    U_a = sp.var(x, ddof=1)
    U_b = sp.var(y, ddof=1)
    return (mean_a - mean_b) / sp.sqrt(U_a/n_a + U_b/n_b)

a = permutation_test(x, y, t_test)
```

p value = 0.373