

ケンドールの τ_b

青木繁伸

2020年3月17日

1 目的

二重クロス集計表としてまとめられた二変数について、ケンドールの τ_b を計算する。

2 使用法

```
import sys
sys.path.append("statlib")
from xtest import Kendall_tau_b
Kendall_tau_b(f, verbose=False)
```

2.1 引数

<code>f</code>	クロス集計表 (二次元配列または二重リスト)
<code>verbose</code>	必要最小限のプリント出力をする

2.2 戻り値の名前

<code>"taub"</code>	τ_b
<code>"ase1"</code>	漸近標準誤差
<code>"ase0"</code>	漸近標準誤差
<code>"Z"</code>	標準化得点
<code>"pvalue"</code>	p 値
<code>"Z2"</code>	標準化得点 2
<code>"pvalue2"</code>	p 値 2

3 使用例

```
d = [[0, 3, 1, 1],
      [1, 4, 4, 0],
      [1, 3, 0, 1],
```

```
[1, 0, 1, 0]]

import sys
sys.path.append("statlib")
from xtest import Kendall_tau_b

a = Kendall_tau_b(d, verbose=True)
```

```
tau b = -0.18065
ase1 = 0.20432
ase0 = 0.20678
Z value = -0.87363
p value = 0.38232
Z value 2 = -1.14558
p value 2 = 0.25197
```

```
print(a["taub"])
```

```
-0.18065152032833257
```

上の例の d は、以下に示す x と y のクロス集計表である。

x と y のケンドールの順位相関係数はケンドールの τ_b ということである。

`statlib.multi` の `cor()` で計算してみる。

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

```
import sys
sys.path.append("statlib")
from multi import cor

b = cor(x, y, method="Kendall")
```

```
Kendall's correlation coefficient = -0.181
```

```
print(b["r"])
```

```
-0.1806515203283326
```

`statlib.xtest` の `cor_test()` で計算してみる。

```
from xtest import cor_test
c = cor_test(x, y, method="Kendall", verbose=False)
print("tau_b = ", c["estimate"], " p value =", c["pvalue"])
```

```
tau_b = -0.18065152032833257 p value = 0.3507034029433831
```

Python では `scipy.stats.kendalltau` である。

```
from scipy.stats import kendalltau
kendalltau(x, y)
```

KendalltauResult(correlation=-0.18065152032833257, pvalue=0.3507034029433831)