

κ 統計量

青木繁伸

2020年3月17日

1 目的

κ 統計量を求める。

R の `vcd` パッケージに、`Kappa` 関数が用意されている。

2 使用法

```
import sys
sys.path.append("statlib")
from xtest import kappa_statistic
kappa_statistic(o, w=False, conflevel=0.95, verbose=True)
```

2.1 引数

<code>o</code>	集計表
<code>w</code>	重みつき κ を求めるときは重み行列
<code>conflevel</code>	信頼率
<code>verbose</code>	必要最小限のプリント出力をする (デフォルトは <code>True</code>)

2.2 戻り値の名前

<code>"kappa"</code>	κ 統計量
<code>"Z"</code>	Z 統計量
<code>"pvalue"</code>	p 値
<code>"confint"</code>	信頼区間
<code>"conflevel"</code>	信頼率
<code>"sigmakappa"</code>	σ_{κ}
<code>"sigmakappa0"</code>	$\sigma_{\kappa 0}$
<code>"o"</code>	集計表
<code>"w"</code>	重み
<code>"method"</code>	検定手法名

3 使用例

3.1 重みなしの場合

```
x = [[12, 6, 1],
      [ 3, 19, 4],
      [ 2, 5, 34]]

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

ans = kappa_statistic(x)
```

```
Kappa statistic
      kappa = 0.61525
          Z = 7.52025
      p value < 0.0001
      conf.int. = [0.47219, 0.75831]
      sigma_kappa = 0.07299
      sigma_kappa_0 = 0.08181
```

3.2 重みのある場合

```
w = [[0, 1, 3],
      [1, 0, 1],
      [3, 1, 0]]
ans = kappa_statistic(x, w)
```

```
Kappa statistic(weighted)
      kappa = 0.69326
          Z = 6.15628
      p value < 0.0001
      conf.int. = [0.55870, 0.82783]
      sigma_kappa = 0.06866
      sigma_kappa_0 = 0.11261
```

4 既存の Python 関数との比較

4.1 `sklearn.metrics.cohen_kappa_score()`

`sklearn.metrics.cohen_kappa_score()` は原データを対象にする。
使用例に示した集計表から原データを復元できるので、それを使って計算する。

```

a = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 2, 2, 0, 0, 0,
0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 2, 2, 2, 2, 2, 0, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2]
b = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2]

from sklearn.metrics import cohen_kappa_score
cohen_kappa_score(a, b)

```

0.6152535151256924

重み付きの κ は, `weights="linear"` か `weights="quadratic"` の選択しかできない (対角線から離れるにつれ $1, 2, 3, \dots$ となるか $1, 2^2, 3^2, \dots$ となるか)。少なくとも **scikit-learn 0.20.3** には `sample_weights` を指定できるとマニュアルには書いてあるが, 実装されていないようだ。

```

# weights="linear"

w = [[0, 1, 2],
      [1, 0, 1],
      [2, 1, 0]]

ans = kappa_statistic(x, w, verbose=False)
print(ans["kappa"])

```

0.6634050880626223

```
cohen_kappa_score(a, b, weights="linear")
```

0.6634050880626223

```

# weights="quadratic"

w = [[0, 1, 4],
      [1, 0, 1],
      [4, 1, 0]]

ans = kappa_statistic(x, w, verbose=False)
print(ans["kappa"])

```

0.713587921847247

```
cohen_kappa_score(a, b, weights="quadratic")
```

0.713587921847247

4.2 skll.metrics.kappa()

`skll.metrics.kappa()` は `weights="linear"` か `weights="quadratic"` の他に、任意の二次元配列で重みを指定できる。重み付きなしの場合がデフォルトである (`weights=None`)。

```
w = [[0, 1, 3],
      [1, 0, 1],
      [3, 1, 0]]

ans = kappa_statistic(x, w, verbose=False)
print(ans["kappa"])

import numpy as np
from skll.metrics import kappa

kappa(a, b, weights=np.array([[0, 1, 3], [1, 0, 1], [3, 1, 0]]))
```