フィッシャーの正確検定とオッズ比

青木繁伸

フィッシャーの正確検定を行う場合，**Python** には `scipy.stats.fisher_exact()` がある。

また，本来は $m \times n$ 分割表に拡張したフィッシャーの正確検定を行う `FisherExact.fisher_exact()` もあるが，不具合があるようで 2×2 分割表の場合のみ使用できる。

ここでは 2×2 分割表においてフィッシャーの正確検定を行うプログラムを提示する。p 値を求めるときに，以下の 3 つの方法から選択することができる。

ピアソンの方法: 観察された四分表での χ^2 値を χ_0^2 とする。可能な全ての四分表に対する χ^2 値を求め，$\chi^2 \geqq \chi_0^2$ となる四分表の生起確率の和を p 値とする。

フィッシャーの方法: 観察された四分表の生起確率を P_0 とする。可能な全ての四分表の生起確率 P を求め，$P \leqq P_0$ となる四分表の生起確率の和を p 値とする

オッズ比に基づく方法: 観察された四分表でのオッズ比を OR_0 とする。可能な全ての四分表のオッズ比 OR を求め，$OR \geqq OR_0$ となる四分表の生起確率の和を p 値とする。

1 使用法

```
from Fisher import Fisher
Fisher(a, b, c, d, verbose=True)

from Fisher import OddsRatio
OddsRatio(a, b, c, d)
```

1.1 引数

a, b, c, d	四分表のマス目の値
verbose	True ならば必要最小限のプリント出力をする（デフォルトは True）

1.2 戻り値の名前

"alternative"	両側検定・片側検定の種別
"p(Pearson)"	ピアソンの方法による p 値
"p(Fisher)"	フィッシャーの方法による p 値
"p(Odds Ratio)"	オッズ比に基づく p 値
"chi square"	観察された分割表における χ^2 値
"odds ratio"	観察された分割表におけるオッズ比
"probability"	観察された分割表の生起確率
"tbl"	可能な分割表の統計量
"method"	検定手法名

2 使用例

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

a = Fisher(10, 13, 16, 61)
```

```
Fisher's Exact Test for Count Data
data: matrix(c(23, 13, 16, 61), 2)
alternative is 'two_sided'
p(Pearson) = 0.0353, p(Fisher) = 0.0551, p(Odds Ratio) = 0.0551
```

ピアソンの方法による p 値: 0.03529413313496793

フィッシャーの方法による p 値: 0.055089906063589475

オッズ比に基づく p 値: 0.055089906063589475

以下は，可能な分割表それぞれにおける，χ^2 値，オッズ比，生起確率の表である。mk の列に * が表示されているのが観察された分割表である。

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

	a	mk	chisq	oddsratio	prob
0	0		10.494910	24.184466	3.317552e-04
1	1		7.278386	10.576923	3.815185e-03
2	2		4.648818	4.754717	1.979577e-02
3	3		2.606207	2.839506	6.158685e-02
4	4		1.150553	1.900000	1.287725e-01
5	5		0.281857	1.350000	1.922390e-01
6	6		0.000117	1.005882	2.124746e-01
7	7		0.305335	1.335526	1.779344e-01
8	8		1.197510	1.748148	1.146018e-01
9	9		2.676642	2.268908	5.730091e-02
10	10	*	4.742731	2.932692	2.235675e-02
11	11		7.395777	3.788889	6.818481e-03
12	12		10.635780	4.909091	1.623448e-03
13	13		14.462741	6.400000	3.004940e-04
14	14		18.876658	8.425926	4.292771e-05
15	15		23.877533	11.250000	4.683023e-06
16	16		29.465364	15.314286	3.844272e-07
17	17		35.640153	21.407407	2.327847e-08
18	18		42.401899	31.050000	1.012107e-09
19	19		49.750602	47.500000	3.043931e-11
20	20		57.686262	78.888889	6.002118e-13
21	21		66.208879	151.200000	7.145379e-15
22	22		75.318454	401.500000	4.449177e-17
23	23		85.014985	1000.428571	1.045635e-19

```
a = Fisher(10, 13, 16, 61, alternative="two_sided")
```

```
Fisher's Exact Test for Count Data
data: matrix(c(23, 13, 16, 61), 2)
alternative is 'two_sided'
p(Pearson) = 0.0353, p(Fisher) = 0.0551, p(Odds Ratio) = 0.0551
```

```
a = Fisher(10, 13, 16, 61, alternative="less")
```

```
Fisher's Exact Test for Count Data
data: matrix(c(23, 13, 16, 61), 2)
alternative is 'less'
p(Pearson) = 0.9912, p(Fisher) = 0.9912, p(Odds Ratio) = 0.9912
```

```
a = Fisher(10, 13, 16, 61, alternative="greater")
```

```
Fisher's Exact Test for Count Data
data: matrix(c(23, 13, 16, 61), 2)
alternative is 'greater'
p(Pearson) = 0.0311, p(Fisher) = 0.0311, p(Odds Ratio) = 0.0311
```

scipy.stats.fisher_exact が返すのは，オッズ比と Fisher の定義による p 値である。

```
from scipy.stats import fisher_exact
fisher_exact([[43, 20], [13, 19]])
```

```
(3.1423076923076922, 0.014774409770060257)
```

```
a = Fisher(43, 20, 13, 19)
```

```
Fisher's Exact Test for Count Data
data: matrix(c(56, 20, 13, 19), 2)
alternative is 'two_sided'
p(Pearson) = 0.0148, p(Fisher) = 0.0148, p(Odds Ratio) = 0.0148
```

```
import sys
sys.path.append("statlib")
from Fisher import OddsRatio

print(a["odds ratio"])
```

```
3.1423076923076922
```

```
print(OddsRatio(43, 20, 13, 19)) # 3.1423076923076922
```

```
3.1423076923076922
```

両側検定・片側検定の種別ごとの結果

```
a = 10; b = 13; c = 16; d = 61
ans = Fisher(a, b, c, d, alternative="two_sided")
print(fisher_exact([[a, b], [c, d]], alternative="two-sided")[1])
```

```
Fisher's Exact Test for Count Data
data: matrix(c(23, 13, 16, 61), 2)
alternative is 'two_sided'
p(Pearson) = 0.0353, p(Fisher) = 0.0551, p(Odds Ratio) = 0.0551
0.05508990606358398
```

```
ans = Fisher(a, b, c, d, alternative="less")
print(fisher_exact([[a, b], [c, d]], alternative="less")[1])
```

```
Fisher's Exact Test for Count Data
data: matrix(c(23, 13, 16, 61), 2)
alternative is 'less'
p(Pearson) = 0.9912, p(Fisher) = 0.9912, p(Odds Ratio) = 0.9912
0.9912095576899763
```

```
ans = Fisher(a, b, c, d, alternative="greater")
print(fisher_exact([[a, b], [c, d]], alternative="greater")[1])
```

```
Fisher's Exact Test for Count Data
data: matrix(c(23, 13, 16, 61), 2)
alternative is 'greater'
p(Pearson) = 0.0311, p(Fisher) = 0.0311, p(Odds Ratio) = 0.0311
0.031147192560984603
```

FisherExact.fisher_exact() は 2×2 より大きな分割表の場合にはエラーになる。2×2 の分割表の場合には，対立仮説は 3 種から選べる（デフォルトは alternative="tow-sided"）。

```
from FisherExact import fisher_exact as fisher
fisher([[a, b], [c, d]])
```

```
0.05508990606358398
```

```
fisher([[a, b], [c, d]], alternative="less")
```

```
0.9912095576899763
```

```
fisher([[a, b], [c, d]], alternative="greater")
```

```
0.031147192560984603
```