

test/testparm - 線形仮説の検定 【評価版】

test コマンドは最も最近フィットされたモデルのパラメータに関する線形の仮説について Wald 検定を実行します。testparm コマンドは test と同様の機能を提供しますが、*varlist* を入力とする点が異なります。

- | | |
|------------------|------------|
| 1. 基本的な用例 | Example 1 |
| | Example 2 |
| | Example 3 |
| | Example 4 |
| | Example 5 |
| | Example 6 |
| | Example 7 |
| | Example 8 |
| | Example 9 |
| | Example 10 |
| 2. 複数式推定の場合の特殊構文 | |
| 3. 制約時の係数値 | |
| 4. 多重検定 | |

1. 基本的な用例

test は最も最近フィットされたモデルを対象とした線形の条件に関し、 F 検定、または χ^2 検定を実行します。任意の推定コマンド実行後に test コマンドを用いることができます。しかし最尤法の場合、test の実行結果は Wald 検定に基づくもので、推定された共分散行列にのみ依存したものとなります。計算的にはより負担の大きい尤度比検定を実行したい場合には lrtest コマンドを使用してください ([R] lrtest (mwp-181) 参照)。

test の構文には次の 4 種類があります。

```
test coeflist [ 構文 1 ]
test exp = exp [= ... ] [ 構文 2 ]
test [ eqno ] [: coeflist ] [ 構文 3 ]
test [ eqno = eqno [= ... ] ] [: coeflist ] [ 構文 4 ]
```

構文 2

```
test exp = exp [= ... ]
```

は任意の推定に対して使用できます。 *devar* を x_1, x_2, x_3 に対しフィットさせた場合、

```
test  $x_1 + x_2 = x_3$ 
```

というコマンドは x_1, x_2 に対する係数値の和が x_3 の係数値に等しいという条件をテストします。数式 *exp* はどんなに複雑なものであっても構いません。

等号を複数指定することもできます。例えば

```
test  $x_1 + x_2 = x_3 + x_4 = x_5 + x_6$ 
```

と指定した場合には 3 組の係数値の和が等しいことをテストします。

x_1 と指定した場合、それは x_1 に対する係数を指すものとして解釈されます。より明示的に

```
test  $\_b[x_1] + \_b[x_2] = \_b[x_3]$  あるいは test  $\_coef[x_1] + \_coef[x_2] = \_coef[x_3]$ 
```

等と指定しても構いません。一方、複数の方程式を用いたモデルの場合には、ある一つの独立変数に対し複数の係数が対応するため、より明示的な指定が必要となります。例えば

```
test [#2] $x_1 + [#2]x_2 = [#2]x_3$ 
```

と指定した場合には方程式 2 に関する条件をテストします。あるいは

```
test [ford] $x_1 + [ford]x_2 = [ford]x_3$ 
```

と指定した場合には、*ford* に対応した方程式に関する条件をテストします。数式をまたがる形で

```
test [ford] $x_1 + [ford]x_2 = [buick]x_3$ 
```

と指定することも可能です。

評価版では割愛しています。

▷ Example 1: 構文 1

Example データセット *census3.dta* には米国 50 州における国勢調査データが含まれています。ここでは各州における出生率 *brate* を次の変数によって説明する線形回帰モデルをフィットさせます。

変数	内容
<i>medage</i>	各州における年齢中央値
<i>region</i>	地域区分 (1 = Northeast, 2 = North Central, 3 = South, 4 = West)

まずデータセットをロードします。

```
. use http://www.stata-press.com/data/r14/census3.dta *1
(1980 Census data by state)
```

次に regress コマンドを用いてモデルのフィットを行います。なお、c.medage#c.medage は medage² を意味するわけですが、このような因子変数の用法については [U] 11.4.3 Factor variables (*mwp-028*) をご参照ください。

```
. regress brate medage c.medage#c.medage i.region, nofvlabel *2
```

. regress brate medage c.medage#c.medage i.region, nofvlabel						
Source	SS	df	MS	Number of obs	=	50
Model	38803.4208	5	7760.68416	F(5, 44)	=	100.63
Residual	3393.39921	44	77.1227094	Prob > F	=	0.0000
				R-squared	=	0.9196
				Adj R-squared	=	0.9104
Total	42196.82	49	861.159592	Root MSE	=	8.782

brate	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
medage	-109.0958	13.52452	-8.07	0.000	-136.3527	-81.83892
c.medage# c.medage	1.635209	.2290536	7.14	0.000	1.173582	2.096836
region						
2	15.00283	4.252067	3.53	0.001	6.433353	23.57231
3	7.366445	3.953335	1.86	0.069	-.6009775	15.33387
4	21.39679	4.650601	4.60	0.000	12.02412	30.76946
_cons	1947.611	199.8405	9.75	0.000	1544.859	2350.363

test コマンドを使うとフィットされたモデルを対象に種々の検定を行うことができます。例えば 3.region の係数値が 0 と言えるかどうかを検定してみます。

- Statistics ▷ Postestimation ▷ Tests, contrasts, and comparisons of parameter estimates
 - ▷ Linear tests of parameter estimates ▷ Launch と操作
- test ダイアログ: Main タブ: Specification 1: Test type: Coefficients are 0
 - Test these coefficients: 3.region

*1 メニュー操作 : File ▷ Example Datasets ▷ Stata 14 manual datasets と操作、Base Reference Manual [R] の test の項よりダウンロードする。

*2 メニュー操作 : Statistics ▷ Linear models and related ▷ Linear regression なお、nofvlabel というオプションは値ラベルによる表示を抑止するためのものです。

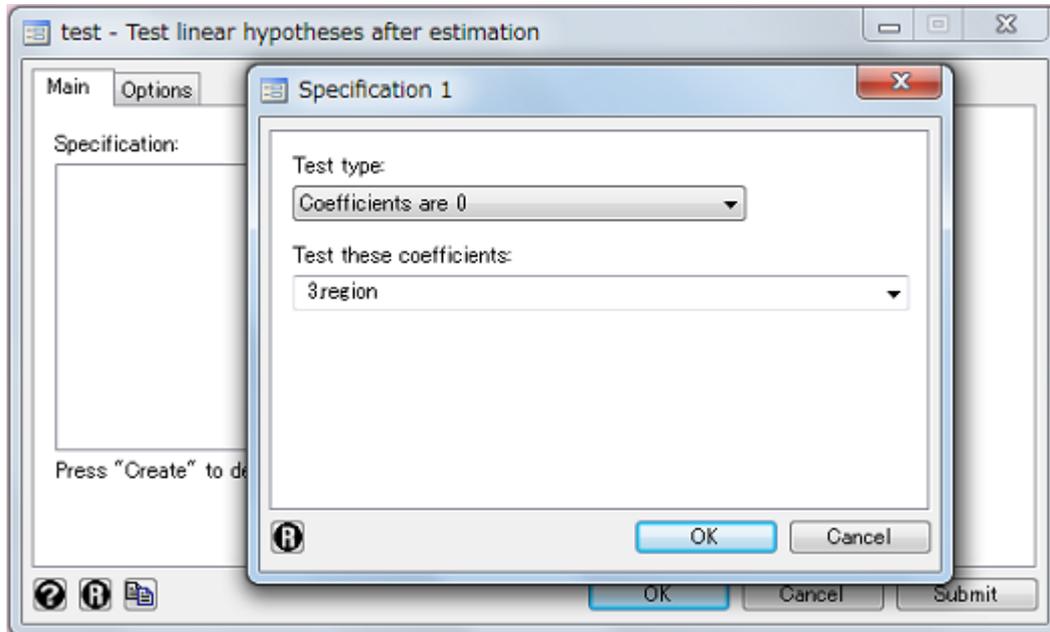


図1 test ダイアログ - Main タブ

```
. test (3.region)

( 1) 3.region = 0

      F( 1, 44) = 3.47
      Prob > F = 0.0691
```

F 統計量は $F(1, 44) = 3.47$ 、それに対する p 値は 0.0691 とレポートされています。従って有意水準を 10% とするならば 3.region の係数値を 0 とする帰無仮説を棄却できますが、5% の場合には棄却できないことになります。

この test コマンドによる検定結果は 3.region に対する regress からの出力と一致しています。regress からの出力では 3.region の係数値に関する t 統計量が 1.86、その p 値が 0.069 とレポートされているわけですが、分子の自由度が 1 のときの F 分布は t^2 分布と等価である点に注意してください。実際、 $1.86^2 \approx 3.47$ となります。 <

□ Technical note

最尤法に基づくものも含め、すべての推定コマンド実行後において、ある変数が 0 であるかの検定は推定コマンドの出力上の情報と同一となります。テストは推定された共分散行列を用いて同様に行われるもので、Wald 検定として知られています。推定コマンドからの出力が t 統計量ではなく z 統計量に基づくものであった場合、test は F 統計量ではなく χ^2 統計量を用いた結果をレポートします。 □

▷ Example 2: 構文 2

評価版では割愛しています。

▷ Example 3: 構文 2

評価版では割愛しています。

▷ Example 4

評価版では割愛しています。

▷ Example 5: 結合仮説の検定

評価版では割愛しています。

▷ Example 6

評価版では割愛しています。

▷ Example 7: varlist の指定

評価版では割愛しています。

▷ Example 8: 再表示

評価版では割愛しています。

▷ Example 9: 複数係数の等値性

評価版では割愛しています。

▷ Example 10

評価版では割愛しています。

2. 複数式推定の場合の特殊構文

評価版では割愛しています。

3. 制約時の係数値

評価版では割愛しています。

4. 多重検定

結合仮説 (joint hypothesis) の検定に際して、そのベースとなる自由度 1 の仮説について内容を確認したいときがあります。複数の条件の中でどれが問題だったのかを問うわけです。そのために用意されているのが `mtest`(multiple testing) オプションです。

セクション 2 では

```
. test [price=weight], common constant
```

と操作することによって次のような検定結果が得られたわけです。

```
. test [price=weight], common constant

( 1)  [price]foreign - [weight]foreign = 0
( 2)  [price]_cons - [weight]_cons = 0

           chi2( 2) =    51.23
           Prob > chi2 =    0.0000
```

この結合検定の結果である 0.0000 という p 値に対して、`foreign`, `_cons` の各々の貢献度を確認するには次のように操作します。

- Statistics ▸ Postestimation ▸ Tests, contrasts, and comparisons of parameter estimates
 - Linear tests of parameter estimates ▸ Launch と操作
- `test` ダイアログ: Main タブ: Specification 1: Test type: Coefficients equal between equations
 - Equation: price → Add
 - Equation: weight → Add
 - Test these coefficients: 空欄
- Options タブ: Test only variables common to all the equations: ✓
 - Include the constant in coefficients to be tested: ✓
 - Do multiple testing: No adjustment (デフォルト)

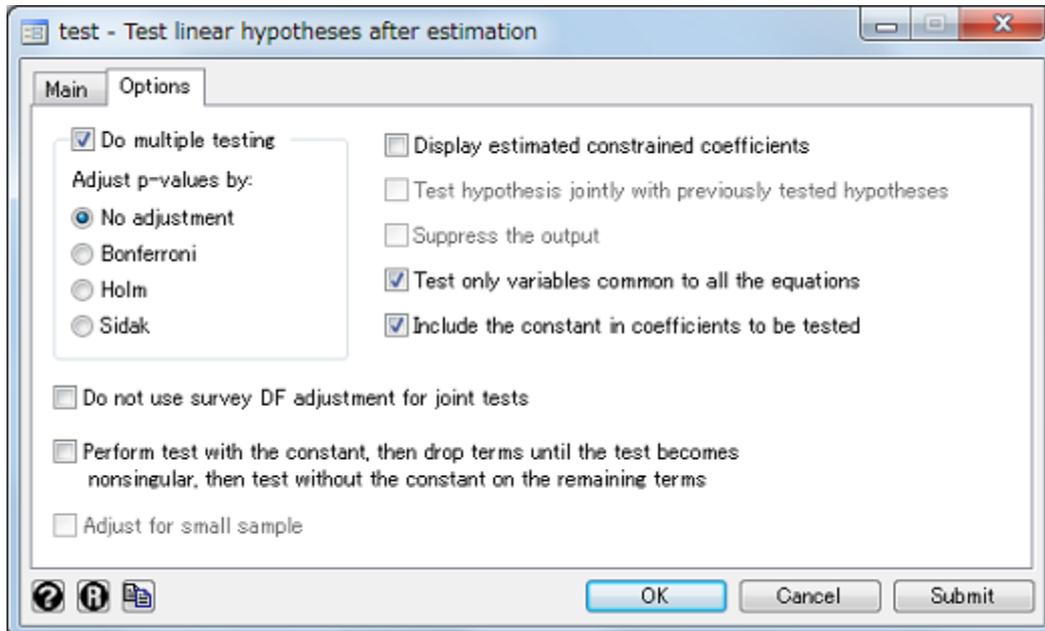


図7 test ダイアログ - Options タブ

```
. test ([price = weight]), mtest(noadjust) common constant

( 1) [price]foreign - [weight]foreign = 0
( 2) [price]_cons - [weight]_cons = 0
```

	chi2	df	p
(1)	23.07	1	0.0000 #
(2)	11.17	1	0.0008 #
all	51.23	2	0.0000

unadjusted p-values

この結果からすると、0.0000 という p 値に対しては双方共に貢献しているように見えます。ここに示されている自由度 1 のテスト結果は、それを単独に test したときに得られる結果と同一のものです。

```
. test [price = weight]: foreign
```

```
. test [price = weight]: foreign

( 1) [price]foreign - [weight]foreign = 0

      chi2( 1) =    23.07
      Prob > chi2 =    0.0000
```

```
. test [price = weight]: _cons
```

```
. test [price = weight]: _cons

( 1)  [price]_cons - [weight]_cons = 0

           chi2( 1) =   11.17
           Prob > chi2 =    0.0008
```

しかしこれらの単純な仮説を複数同時に吟味する際には注意が必要です。仮説の数が増えるほど、一つの仮説が偶然によって有意と判定されてしまう確率が高くなるからです。上記の操作では多重検定に伴う補正を行わなかったわけですが、`mtest` オプションでは Bonferroni, Šidák, Holm の補正を選択することもできます。そこで今度は Bonferroni の補正法を指定して `test` を実行してみます。

```
. test ([price = weight]), mtest(bonferroni) common constant
```

```
. test [price = weight]: _cons

( 1)  [price]_cons - [weight]_cons = 0

           chi2( 1) =   11.17
           Prob > chi2 =    0.0008

. test ([price = weight]), mtest(bonferroni) common constant

( 1)  [price]foreign - [weight]foreign = 0
( 2)  [price]_cons - [weight]_cons = 0
```

	chi2	df	p
(1)	23.07	1	0.0000 #
(2)	11.17	1	0.0017 #
all	51.23	2	0.0000

Bonferroni-adjusted p-values