Stata での複数仮説検定コマンドの概要
An overview of multiple hypothesis testing commands in Stata

著者：David McKenzie (Lead Economist, Development Research Group, World Bank)
2020 年 6 月 1 日
2020 年 6 月 8 日更新（rwolf の旧バージョンについての記載について修正を反映）

原文ブログ URL：
https://blogs.worldbank.org/impactevaluations/overview-multiple-hypothesis-testing-commands-stata

複数仮説検定を実施するための、ユーザー作成の Stata パッケージが増加している。私はこれらパッケージの長所と短所を見比べることが、どのパッケージがどの状況で機能するかを自分自身が思い出す際に役立ち、また同時に、他の人にとっても役立つ手段になるかもしれないと考え、比較しようと考えた。私はまず、これら様々なパッケージの作者らに対してコードを書いてくれたことを感謝することから始める。なぜならば、そのコードがある特定の実践例において求められる機能すべてを備えているわけではないという不満を私が持ったとしても、他のニーズにおいてはとても有用な場合があるからである。

我々が解決しようとしている問題は何か?

4 つの処置（*treat1, treat2, treat3, treat4*）からなる実験を実施し、様々なアウトカム（*y1, y2, y3, y4, y5*）への効果を調べることに関心がある場合を考えよう。ここで時点 t におけるアウトカム j について、以下の回帰分析を行う：

$$Y(j, t) = a + b1*treat1 + b2*treat2 + b3*treat3 + b4*treat4 + c1*y(j, 0) + d'X + e(j, t)$$

5 つのアウトカムと 4 つの処置で、 20 の仮説検定を行い、 20 の p 値が得られる。以下は最近私が行った実験（ここでのアウトカムは、企業の存続と 4 種類のビジネス・プラクティス指標）からの例であり、係数の下に p 値が示されている。

Example of Five Outcomes and Four Treatments, with p-values in parentheses

	Y1	Y2	Y3	Y4	Y5
Treat 1	0.022	0.043	0.083**	0.079***	0.032
	(0.516)	(0.258)	(0.031)	(0.001)	(0.178)
Treat 2	0.043	0.060	0.099***	0.083***	0.046**
	(0.168)	(0.109)	(0.006)	(0.001)	(0.048)
Treat 3	0.030	-0.006	-0.016	0.008	0.009
	(0.356)	(0.877)	(0.665)	(0.726)	(0.691)
Treat 4	0.042	0.093**	0.070*	0.044*	0.052**
	(0.179)	(0.014)	(0.052)	(0.064)	(0.024)
Sample Size	726	678	678	678	678

ここで、どの処置もいずれのアウトカムにも効果をもたらさず（すべての帰無仮説が正しい）、かつアウトカム同士が独立している場合を考える。この場合、仮説を 1 つずつ検定するだけだと、臨界値（critical value）0.05 を使用して 1 つ以上の仮説を誤って棄却する確率は、1-0.95^20=64%となる （臨界値 0.10 を使った場合は 88%となる）。この事実ゆえに、これらの誤った棄却の可能性を減らすため、複数の仮説を検定している（testing multiple hypotheses）という事実を調整する何らかの方法が欲しい。その方法こそが、以下の多様な手法が行うことである。

Anderson による「FDR シャープ化された q 値」のコード

この問題に対処する最も一般的な方法の 1 つは、Michael Anderson のコード[1]を使用して、False Discovery Rate（FDR）においてシャープ化された q 値（sharpened FDR q-values）を計算することである。FDR は、第 1 種の過誤の棄却（誤った棄却）の予想される割合である。Anderson はこの手順を脚注 2 のリンク先で説明している[2]。

このコードはとても使いやすい。p 値を保存し、それをデータとして Stata に読み込み、彼のコードを実行してシャープ化された q 値を得るだけである。私の例では、それらを以下の表に示す。

Example of Five Outcomes and Four Treatments, with sharpened q-values

	Y1	Y2	Y3	Y4	Y5
Treat 1	0.022	0.043	0.083**	0.079***	0.032
p-value	(0.516)	(0.258)	(0.031)	(0.001)	(0.178)
sharpened q-value	[0.381]	[0.255]	[0.091]	[0.011]	[0.179]
Treat 2	0.043	0.060	0.099***	0.083***	0.046**
p-value	(0.168)	(0.109)	(0.006)	(0.001)	(0.048)
sharpened q-value	[0.179]	[0.151]	[0.038]	[0.011]	[0.109]
Treat 3	0.030	-0.006	-0.016	0.008	0.009
p-value	(0.356)	(0.877)	(0.665)	(0.726)	(0.691)
sharpened q-value	[0.312]	[0.443]	[0.381]	[0.381]	[0.381]
Treat 4	0.042	0.093**	0.070*	0.044*	0.052**
p-value	(0.179)	(0.014)	(0.052)	(0.064)	(0.024)
sharpened q-value	[0.179]	[0.064]	[0.109]	[0.116]	[0.084]
Sample Size	726	678	678	678	678

いくつかの注意点が挙げられる：

・このアプローチの人気の主な理由は（その単純さに加えて）、元の p 値とシャープ化された q 値との比較に見られる。もし我々がボンフェローニ調整（Bonferroni adjustment）を適用する場合は、p 値にアウトカムの数（20）を掛けて、上限値を 1.000 に設定する。この場合、例えば、アウトカ

[1] 下記によりダウンロード可能：
http://are.berkeley.edu/~mlanderson/downloads/fdr_sharpened_qvalues.do.zip
[2] https://are.berkeley.edu/~mlanderson/pdf/Anderson%202008a.pdf

ム Y3 と処置 1 の p 値 0.031 が 0.62 に調整される。これと異なり、シャープ化された q 値は 0.091 である。つまり、この方法は他の多くの方法よりもはるかに大きな検出力（power）を持つ。
· この方法は p 値をインプットとして用いるので、各回帰分析に何を含めるかについての柔軟性を有する。例えば、クラスター誤差のある回帰分析とない回帰分析、コントロールのある回帰分析とない回帰分析などに応用が可能である。
·Anderson がコードの中で書いているように、多くの仮説が棄却された場合、シャープ化された q 値は調整されていない p 値よりも「小さく」なることがある。なぜならば、もし多くの棄却が真である場合、いくつかの偽の棄却も許容でき、偽の発見の割合（FDR）を低く保つことができるからである。ここでは、結果 Y1 と処置 1 に関してこの例が見られる。
· この方法の欠点は、 p 値間の相関を考慮しないことである。例えば、私の例では、ある処置によってビジネス・プラクティス Y2 が改善されれば、ビジネス・プラクティス Y3、Y4、Y5 も改善される可能性が高いと考えられるだろう。Anderson は、シミュレーションでは、この方法は正に相関する p 値でもうまく機能するように思われるが、p 値が負の相関を持つ場合には、より保守的なアプローチが必要であると指摘している。

mhtexp

このコードは John List, Azeem Shaikh and Yang Xu (2016) [3]によって書かれた手順を実装している。これは ssc install mhtexp とタイプすることで取得できる。

この手順は、すべての帰無仮説が正しいときに少なくとも 1 つの仮説を誤って棄却する第 1 種の過誤の確率（familywise type I error: FWER） をコントロールすることを目的としている。これは Romano and Wolf の研究に基づいており、ブートストラップ法を使用して、異なる検定の結合依存の構造（joint dependence structure）に関する情報を考慮する。つまり、p 値を相関させることが可能となる。

· このコマンドは、複数のアウトカムと複数の処置を対象とすることが可能だが、コントロール変数を含めることができず（従って、対象となるアウトカムのベースライン値や層化ランダム化の層（strata）固定効果をコントロールできない)、標準誤差のクラスタリングもできない。
· その上ではこのコマンドの利用は単純である。ここでは、*treatgroup* という変数を作成した。これは、対照群は 0、treat1 は 1 を、treat2 は 2、treat3 は 3、treat4 は 4 をとる変数である。この時のコマンドは以下の通り：

```
mhtexp y1 y2 y3 y4 y5, treatment (treatgroup) bootstrap (3000)
```

以下の表に、これらの FWER p 値を追加した。Bonferroni の p 値ほど大きくはないが、元の p 値やシャープ化された q 値よりもはるかに大きいことがわかる。元の p 値はコントロール変数を追加してもそれほど変化しないため、この場合はコントロール変数が含まれていないことがこの違

[3] https://doi.org/10.1007/s10683-018-09597-5

いの大きな理由ではない。代わりに、これは比較が多いときに FWER を使用することの問題を反映している。つまり、「あらゆる」第 1 種の過誤を避けるために、アウトカムや処置をより多く追

Example of Five Outcomes and Four Treatments, with mhtexp FWER p-values

	Y1	Y2	Y3	Y4	Y5
Treat 1	0.022	0.043	0.083**	0.079***	0.032
p-value	(0.516)	(0.258)	(0.031)	(0.001)	(0.178)
sharpened q-value	[0.381]	[0.255]	[0.091]	[0.011]	[0.179]
mhtexp FWER p-value	{0.950}	{0.827}	{0.233}	{0.012}	{0.678}
Treat 2	0.043	0.060	0.099***	0.083***	0.046**
p-value	(0.168)	(0.109)	(0.006)	(0.001)	(0.048)
sharpened q-value	[0.179]	[0.151]	[0.038]	[0.011]	[0.109]
mhtexp FWER p-value	{0.684}	{0.666}	{0.110}	{0.016}	{0.398}
Treat 3	0.030	-0.006	-0.016	0.008	0.009
p-value	(0.356)	(0.877)	(0.665)	(0.726)	(0.691)
sharpened q-value	[0.312]	[0.443]	[0.381]	[0.381]	[0.381]
mhtexp FWER p-value	{0.817}	{0.903}	{0.956}	{0.979}	{0.928}
Treat 4	0.042	0.093**	0.070*	0.044*	0.052**
p-value	(0.179)	(0.014)	(0.052)	(0.064)	(0.024)
sharpened q-value	[0.179]	[0.064]	[0.109]	[0.116]	[0.084]
mhtexp FWER p-value	{0.680}	{0.210}	{0.373}	{0.561}	{0.238}
Sample Size	726	678	678	678	678

加すればするほど、調整後の結果がますます厳しくなる。すなわち、検出力が低くなる。これとは対照的に、 FDR アプローチは、より高い検出力と引き換えに何らかの第 1 種の過誤を許容する。どちらがより適しているかは、特定の効果を検証する力と比較して、誤った棄却のコストがどれだけかかるかによって決まる。

rwolf

（私の最初のブログ投稿では、単一の処置の場合にしか対応していないこのコマンドの旧バージョンを使用していた点に注意。現在は、このコマンドで処置が複数の場合も対応できるようになっている。）

このコマンドは、「Romano-Wolf ステップダウン調整された p 値」を計算する。これは、FWER を制御し、ブートストラップ・リサンプリングにより p 値間の相関性を許容することが可能である。それゆえ、少しだけ異なるアルゴリズムではあるが、*mhtexp* と同じことを行うことを目的としており、Romano and Wolf 自身によって（Damien Clarke と共に）開発されたものである。このタイプを入手するには ssc install rwolf と入力すればよい。

· mhtexp とは対照的に、このコマンドではコントロール変数を回帰分析に含めることができるが、すべてのアウトカムに対して同じ（コントロール変数）にすることを求める。また、アウトカム

のベースライン値を含めるオプションもあり、アウトカムごとにこの値を変えることができるので、ANCOVA（ベースライン値が欠落していない場合。欠落したベースライン値については別のダミー変数が必要となる）を使うことができる。

· また、クラスター・ランダム化を許容し、層化ランダム化割当とクラスター・ランダム化割当の両方を考慮してブートストラップ・リサンプリングを行うことができる。

旧バージョンのコマンドでは処置が 1 つの場合にしか適用できなかったが、現在のコマンドでは複数の処置を扱うことができる。しかし、現在のところ、一度に実施できる複数検定補正は処置ごとである。つまり、treat1 について 5 つのアウトカムに対して検定する際の補正はするが、同時に他の処置も検定しているという事実について補正するわけではない。そのため、現在のところ、複数の処置における補正に使用することは勧めない。この問題は今後のアップデートで解決される予定であり、その際にはこの記事を修正するつもりである。

· 処置が 1 つの場合について説明するために、私は処置 1 と対照群のみを使用した場合について示し、上記の他の手法を 5 つの比較（5 つのアウトカムそれぞれに対する処置 1 の影響）だけに応用した場合と比較する。コマンドは次のようになる：

rwolf y1 y2 y3 y4 y5 if treatment

Comparing impacts on just treatment 1 outcomes using rwolf to other methods

	Y1	Y2	Y3	Y4	Y5
Treat 1	0.022	0.043	0.083**	0.079***	0.032
p-value	(0.516)	(0.258)	(0.031)	(0.001)	(0.178)
sharpened q-value	[0.422]	[0.240]	[0.067]	[0.006]	[0.217]
mhtexp FWER p-value	{0.518}	{0.470}	{0.078}	{0.002}	{0.391}
rwolf FWER p-value	<0.527>	<0.417>	<0.060>	<0.003>	<0.327>

予想通り、rwolf の p 値が mhtexp の値と非常に似ていることがわかる。

処置が複数の場合、次のようなコードを使用できる：
rwolf y1 y2 y3 y4 y5, indepvar (treat1 treat2 treat3 treat4) reps(1000) method(areg) abs(batchno) bl(_bl) strata(batchno) seed (456) r
しかし、上記の注意の通り、このコマンドは 1 つの処置の中だけでのみ調整され、処置間では調整されないため、現時点では処置が複数の場合の補正に使用することは勧めない。

なお、現在このコマンドでは、すべての回帰式で同じである限り、areg, ivreg などのコマンドを使用することもできる。

wyoung

Julian Reif によってプログラムされたこのコマンドは、「Westfall-Young ステップダウン調整された p 値」を計算する。これはまた、FWER を制御し、p 値間の相関を可能にする。この方法は Romano-Wolf 法の先駆けであり、アウトカム間の相関を許容するためにブートストラップ・リサンプリングを用いる。Romano and Wolf によると、Westfall-Young の手順では、「部分集合ピヴォタリティ（subset pivotality）」の仮定を追加的に求める。これは特定の状況では満たさない可能性があるため、Romano-Wolf の手順の方がより一般的であると指摘している。私自身は、どのような場合にそれが満たされなくなるかについての大まかな直観をあまり持っていない。

このコマンドを取得するには、ssc install wyoung と入力すればよい。

・このコマンドのよい点は、異なる推計式で異なるコントロール変数を使うことを許容する点である。例えば、アウトカムごとのベースライン変数をコントロールすることができる。また、クラスター・ランダム化を許容し、層化ランダム化割当とクラスター・ランダム化割当の両方を考慮してブートストラップ・リサンプリングを行うこともできる。

・大きな欠点としては、rwolf のように、処置が 1 つの場合しか扱えないことである。

従って、上記の Romano-Wolf の例と同様に、5 つのアウトカムに対して処置 1 についてこの調整を見て、他の方法と比較する。このコマンドは、異なる回帰分析を許容するために、やや長く見える。なお、当初すべての企業が生存しているため、y1（存続率） のベースラインのアウトカムはない。その上で、他のアウトカム b_y1 などのベースラインの値についてコントロールする。

wyoung, cmd("areg y1 treat1, r a(strata)" "areg y2 treat1 b_y2, r a(strata)" "areg y3 treat1 b_y3, r a(strata)" "areg y4 treat1 b_y4, r a(strata)" "areg y5 treat1 b_y5, r a(strata)" familyp(treat1) bootstraps(1000) seed(124)

この結果、mhtexp や rwolf と非常によく似た調整値が得られる。これは予想できることであり、なぜならばこれらの手法すべてが同様のアプローチを使用して FWER を制御しようとしているからである。

Comparing impacts on just treatment 1 outcomes using wyoung to other methods

	Y1	Y2	Y3	Y4	Y5
Treat 1	0.022	0.043	0.083**	0.079***	0.032
p-value	(0.516)	(0.258)	(0.031)	(0.001)	(0.178)
sharpened q-value	[0.422]	[0.240]	[0.067]	[0.006]	[0.217]
mhtexp FWER p-value	{0.518}	{0.470}	{0.078}	{0.002}	{0.391}
rwolf FWER p-value	<0.527>	<0.417>	<0.060>	<0.003>	<0.327>
wyoung FWER p-value	((0.521))	((0.435))	((0.083))	((0.003))	((0.369))

randcmd

これは Alwyn Young が書いた Stata コマンドで[4]、彼の最近の QJE 論文[5]に基づいてランダム化推論（randomization inference）p 値を計算する。これは上記のアプローチとは何か違うことをしている。複数検定のために個々の p 値を調整するのではなく、いずれの処置も効果がないという明確な仮説の結合検定を行い、次いで、Westfall-Young アプローチを用いてこれを推計式間で検定する。そこで、私の例では、どの処置もアウトカムに影響を及ぼさない、Y1 (p = 0.403)、Y2 (p = 0.045) 等々を検定し、さらに完全に無関係であるという帰無仮説（つまり、どの処置もいずれのアウトカムにも影響を及ぼさない）を検定する（ここでは p = 0.022）。このコマンドは、各推計式が異なるコントロール、異なるサンプル、クラスター化標準誤差を持つなどのことができるように非常に柔軟性がある。しかし、これは上記の他のアプローチとは異なる仮説を検定している。

Example of Five Outcomes and Four Treatments, with randcmd randomization-t p-values

	Y1	Y2	Y3	Y4	Y5	All equations
Treat 1	0.022	0.043	0.083**	0.079***	0.032	
p-value	(0.516)	(0.258)	(0.031)	(0.001)	(0.178)	
Treat 2	0.043	0.060	0.099***	0.083***	0.046**	
p-value	(0.168)	(0.109)	(0.006)	(0.001)	(0.048)	
Treat 3	0.030	-0.006	-0.016	0.008	0.009	
p-value	(0.356)	(0.877)	(0.665)	(0.726)	(0.691)	
Treat 4	0.042	0.093**	0.070*	0.044*	0.052**	
p-value	(0.179)	(0.014)	(0.052)	(0.064)	(0.024)	
Young Westfall-Young joint test	0.403	0.045	0.03	0.005	0.091	0.022
Sample Size	726	678	678	678	678	

全てまとめると

以下の表は、なぜ私が最も望ましい複数仮説検定コマンドの探索を続けるかを示している。どのコマンドも、私が望むすべてを実行するわけではない。少ない仮説の検定では、FWER を制御することが有用であり、mhtexp、rwolf、wyoung のうちいずれを選択するかは、私が処置を複数持っているかに依存し、異なるコントロール変数を含めたいかどうかに依存する。多くのアウトカムと処置がある場合、FDR を制御することが最善の方法であると思われ、Anderson の q 値のアプローチが私のスタンバイとなる。私は、それがアウトカム間の相関さえ考慮してくれればよいのにとだけ考える。Young による総合的な有意度についてのオムニバス検定は有用な補完となるが、別の問いに答えることとなる。

[4] https://ideas.repec.org/c/boc/bocode/s458774.html
[5] https://doi.org/10.1093/qje/qjy029

	Anderson q	*mhtexp*	*rwolf*	*wyoung*	*randcmd*
Testing concept	FDR	FWER	FWER	FWER	joint irrelevance
Allows for multiple treatments	Yes	Yes	No	No	Yes
Allows for different controls in different regressions	Yes	No	No	Yes	Yes
Allows for clustered randomization	Yes	No	Yes	Yes	Yes
Accounts for correlation across outcomes	No	Yes	Yes	Yes	Yes
Provides adjusted p-values for individual p-values	Yes	Yes	Yes	Yes	No

更新：新しいバージョンの rwolf は処置が複数の場合を扱うことができるが、現時点では推奨されない。

あなたのお気に入りのパッケージを忘れているだろうか？もちろん、ここでより多くのことができるバージョンをあなた自身がプログラムすることもできるが、みなが入手して使用するのが簡単な既製のパッケージこそが最も歓迎される。ぜひコメント欄で教えてほしい。

追加情報：mhtreg

他の人もこれらの限界に気付いていることがわかった。Andreas Steinmayr は、mhtexp コマンドを拡張して、mhtreg コマンドで異なる回帰式に異なるコントロール変数を含めることができるようにし、クラスター・ランダム化も許容できるようにした。

これを取得するには以下のようにタイプする：

net install mhtreg, from (https://sites.google.com/site/andreassteinmayr/mhtreg)

私はこれを試してみたが、確かに異なる回帰式で異なるコントロール変数を許容する。これは間違いなく mhtexp の改善である。ただし、注意すべきいくつかの限界がある。

· 処置が複数の場合のコードがやや書きにくい。各推計式の最初の説明変数の補正のみを行う。このため、処置が複数の場合の検定をするには、回帰式を繰り返し記述して、なおかつ処置が記述される順番を変える必要がある。
例）私の 20 の異なるアウトカムの例で検定するには、コードは次のようになる：

mhtreg (y1 treat1 treat2 treat3 treat4 i.strata, r) (y1 treat2 treat3 treat4 treat1 i.strata, r) (y1 treat3 treat4 treat1 treat2 i.strata, r) (y1 treat4 treat1 treat2 treat3 i.strata, r)

(y2 treat1 treat2 treat3 treat4 b_y2 i.strata, r) (y2 treat2 treat3 treat4 treat1 b_y2 i.strata, r) ....

· このコマンドはそれぞれの回帰式が実際に回帰式であるようにする必要がある。つまり、reg コマンドを使用しているようにする必要があり、areg や reghdfe のようなコマンドを使用しているようにはしない。または、ivreg や probits あるいはその他のことを行う場合には使用しない。そ

して、説明変数としてあらゆる固定効果を追加する必要があり、ゆえに ITT をする方がよく、TOT やその他の回帰ではない推計（non-regression estimation）はしない方がよい。この点、wyoung コマンドと対照的である。

もう一つ（他のコマンドにも当てはまりうる）注意すべき点は、デフォルトの 3000 ブートストラップ・レプリケーションであっても、補正された p 値があるランダム・シードと別のランダム・シードで少し異なる場合がある。これは特に、珍しいイベントを見ている場合など、非常に低い p 値の場合に問題になりうる。例えば、アウトカム Y4、処置 1、3 つの異なるシードで mhtreg を使用した結果、調整された p 値は 0.0003、0.0057 および 0.011 を得た。このため、調整値のシード選択による頑健性を得るためにはより多くのレプリケーションを指定する必要があるだろう。

以下は、mhtreg を使用した結果と、mhtexp およびシャープ化された q 値との比較の表である。結果は mhtexp と似ているが、その差はベースラインのコントロール変数が含まれることによる。

Example of Five Outcomes and Four Treatments, with mhtreg FWER p-values

	Y1	Y2	Y3	Y4	Y5
Treat 1	0.022	0.043	0.083**	0.079***	0.032
p-value	(0.516)	(0.258)	(0.031)	(0.001)	(0.178)
sharpened q-value	[0.381]	[0.255]	[0.091]	[0.011]	[0.179]
mhtexp FWER p-value	{0.950}	{0.827}	{0.233}	{0.012}	{0.678}
mhtreg FWER p-value	<0.959>	<0.842>	<0.280>	<0.0003>	<0.695>
Treat 2	0.043	0.060	0.099***	0.083***	0.046**
p-value	(0.168)	(0.109)	(0.006)	(0.001)	(0.048)
sharpened q-value	[0.179]	[0.151]	[0.038]	[0.011]	[0.109]
mhtexp FWER p-value	{0.684}	{0.666}	{0.110}	{0.016}	{0.398}
mhtreg FWER p-value	<0.699>	<0.686>	<0.116>	<0.016>	<0.421>
Treat 3	0.030	-0.006	-0.016	0.008	0.009
p-value	(0.356)	(0.877)	(0.665)	(0.726)	(0.691)
sharpened q-value	[0.312]	[0.443]	[0.381]	[0.381]	[0.381]
mhtexp FWER p-value	{0.817}	{0.903}	{0.956}	{0.979}	{0.928}
mhtreg FWER p-value	<0.851>	<0.935>	<0.919>	<0.945>	<0.939>
Treat 4	0.042	0.093**	0.070*	0.044*	0.052**
p-value	(0.179)	(0.014)	(0.052)	(0.064)	(0.024)
sharpened q-value	[0.179]	[0.064]	[0.109]	[0.116]	[0.084]
mhtexp FWER p-value	{0.680}	{0.210}	{0.373}	{0.561}	{0.238}
mhtreg FWER p-value	<0.683>	<0.118>	<0.335>	<0.433>	<0.221>
Sample Size	726	678	678	678	678

このため、比較の表に別の行と列も追加する必要があると思った。そこで更新された比較の表が下記である。

Summary of Stata Multiple Hypothesis Testing Commands

	Anderson q	*mhtexp*	*rwolf*	*wyoung*	*mhtreg*	*randcmd*
Testing concept	FDR	FWER	FWER	FWER	FWER	joint irrelevance
Allows for multiple treatments	Yes	Yes	No	No	Yes	Yes
Allows for different controls in different regressions	Yes	No	No	Yes	Yes	Yes
Allows for clustered randomization	Yes	No	Yes	Yes	Yes	Yes
Accounts for correlation across outcomes	No	Yes	Yes	Yes	Yes	Yes
Provides adjusted p-values for individual p-values	Yes	Yes	Yes	Yes	Yes	No
Allows for different commands (e.g. areg, ivreg)	Yes	No	No	Yes	No	Yes

更新：rwolf のアップデート版では、異なるコマンド（areg, ivreg）対して複数の処置を許容する（しかし、現時点では、複数のアウトカムについて 1 つの処置を調整するだけであり、処置間で複数の検定を調整するわけではないため、処置が複数の場合には私は勧めない)。各回帰式に異なるベースライン変数を含めることができるが、それ以外の場合はコントロール変数を同じにする必要がある。

翻訳：横尾英史（一橋大学）2020 年 6 月 19 日

翻訳に関する質問・コメントは下記にお願いします。

E-mail: hidefumi.yokoo <at> r.hit-u.ac.jp

原文ブログ URL（再掲）：
https://blogs.worldbank.org/impactevaluations/overview-multiple-hypothesis-testing-commands-stata
本翻訳ファイル掲載ページ URL：
http://www1.econ.hit-u.ac.jp/yokoo/j/essays.html