The Stata Journal publishes reviewed papers together with shorter notes or comments, regular columns, book reviews, and other material of interest to Stata users. Examples of the types of papers include 1) expository papers that link the use of Stata commands or programs to associated principles, such as those that will serve as tutorials for users first encountering a new field of statistics or a major new technique; 2) papers that go “beyond the Stata manual” in explaining key features or uses of Stata that are of interest to intermediate or advanced users of Stata; 3) papers that discuss new commands or Stata programs of interest either to a wide spectrum of users (e.g., in data management or graphics) or to some large segment of Stata users (e.g., in survey statistics, survival analysis, panel analysis, or limited dependent variable modeling); 4) papers analyzing the statistical properties of new or existing estimators and tests in Stata; 5) papers that could be of interest or usefulness to researchers, especially in fields that are of practical importance but are not often included in texts or other journals, such as the use of Stata in managing datasets, especially large datasets, with advice from hard-won experience; and 6) papers of interest to those who teach, including Stata with topics such as extended examples of techniques and interpretation of results, simulations of statistical concepts, and overviews of subject areas.

The Stata Journal is indexed and abstracted by CompuMath Citation Index, Current Contents/Social and Behavioral Sciences, RePEc: Research Papers in Economics, Science Citation Index Expanded (also known as SciSearch), Scopus, and Social Sciences Citation Index.

For more information on the Stata Journal, including information for authors, see the webpage

http://www.stata-journal.com
Subscriptions are available from StataCorp, 4905 Lakeway Drive, College Station, Texas 77845, telephone 979-696-4600 or 800-782-8272, fax 979-696-4601, or online at http://www.stata.com/bookstore/sj.html

Subscription rates listed below include both a printed and an electronic copy unless otherwise mentioned.

<table>
<thead>
<tr>
<th></th>
<th>U.S. and Canada</th>
<th>Elsewhere</th>
</tr>
</thead>
<tbody>
<tr>
<td>Printed &amp; electronic</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1-year subscription</td>
<td>$124</td>
<td>$154</td>
</tr>
<tr>
<td>2-year subscription</td>
<td>$224</td>
<td>$284</td>
</tr>
<tr>
<td>3-year subscription</td>
<td>$310</td>
<td>$400</td>
</tr>
<tr>
<td>1-year student subscription</td>
<td>$ 89</td>
<td>$119</td>
</tr>
<tr>
<td>1-year institutional subscription</td>
<td>$375</td>
<td>$405</td>
</tr>
<tr>
<td>2-year institutional subscription</td>
<td>$679</td>
<td>$739</td>
</tr>
<tr>
<td>3-year institutional subscription</td>
<td>$935</td>
<td>$1,025</td>
</tr>
<tr>
<td>Electronic only</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1-year subscription</td>
<td>$ 89</td>
<td>$ 89</td>
</tr>
<tr>
<td>2-year subscription</td>
<td>$162</td>
<td>$162</td>
</tr>
<tr>
<td>3-year subscription</td>
<td>$229</td>
<td>$229</td>
</tr>
<tr>
<td>1-year student subscription</td>
<td>$ 62</td>
<td>$ 62</td>
</tr>
</tbody>
</table>

Back issues of the Stata Journal may be ordered online at http://www.stata.com/bookstore/sjj.html

Individual articles three or more years old may be accessed online without charge. More recent articles may be ordered online.

http://www.stata-journal.com/archives.html

The Stata Journal is published quarterly by the Stata Press, College Station, Texas, USA.

Address changes should be sent to the Stata Journal, StataCorp, 4905 Lakeway Drive, College Station, TX 77845, USA, or emailed to sj@stata.com.

Copyright © 2017 by StataCorp LLC

Copyright Statement: The Stata Journal and the contents of the supporting files (programs, datasets, and help files) are copyright © by StataCorp LLC. The contents of the supporting files (programs, datasets, and help files) may be copied or reproduced by any means whatsoever, in whole or in part, as long as any copy or reproduction includes attribution to both (1) the author and (2) the Stata Journal.

The articles appearing in the Stata Journal may be copied or reproduced as printed copies, in whole or in part, as long as any copy or reproduction includes attribution to both (1) the author and (2) the Stata Journal.

Written permission must be obtained from StataCorp if you wish to make electronic copies of the insertions. This precludes placing electronic copies of the Stata Journal, in whole or in part, on publicly accessible websites, file servers, or other locations where the copy may be accessed by anyone other than the subscriber.

Users of any of the software, ideas, data, or other materials published in the Stata Journal or the supporting files understand that such use is made without warranty of any kind, by either the Stata Journal, the author, or StataCorp. In particular, there is no warranty of fitness of purpose or merchantability, nor for special, incidental, or consequential damages such as loss of profits. The purpose of the Stata Journal is to promote free communication among Stata users.

The Stata Journal, electronic version (ISSN 1536-8734) is a publication of Stata Press. Stata, Stata Press, Mata, and NetCourse are registered trademarks of StataCorp LLC.
The Stata Journal

Articles and Columns

Announcement of the Stata Journal Editors‘ Prize 2017 ......................... 1
Creating HTML or Markdown documents from within Stata using webdoc..... 3
Fitting endogenous stochastic frontier models in Stata........ M. U. Karakaplan 39
Commands for testing conditional moment inequalities and equalities........ 56
A comprehensive set of postestimation measures to enrich interrupted time-series analysis.......................................................A. Linden 73
Within- and between-cluster effects in generalized linear mixed models: A discussion of approaches and the xthybrid command .... R. Schunck and F. Perales 89
Fitting the errors-in-variables model using high-order cumulants and moments... T. Erickson, R. Parham, and T. M. Whited 116
Capturing a Stata dataset and releasing it into REDCap ......................... S. T. Lirette, S. R. Seals, C. Blackshear, and W. May 130
Spatial panel-data models using Stata................................................. F. Belotti, G. Hughes, and A. Piano Mortari 139
The estimation and modeling of cause-specific cumulative incidence functions using time-dependent weights.................................P. C. Lambert 181
biasplot: A package to effective plots to assess bias and precision in method comparison studies........ P. Taffé, M. Peng, V. Stagg, and T. Williamson 208
Estimation of unit values in household expenditure surveys without quantity information.........................................................M. Menon, F. Perali, and N. Tommasi 222
Generalized maximum entropy estimation of linear models.................... P. Corral, D. Kuehn, and E. Jabir 240

Software Updates

250
Announcement of the Stata Journal Editors’ Prize 2017

The editors of the *Stata Journal* are pleased to invite nominations for their 2017 prize in accordance with the following rules. Nominations should be sent as private email to editors@stata-journal.com by July 31, 2017.

1. The *Stata Journal* Editors’ Prize is awarded annually to one or more authors of a specified paper or papers published in the *Stata Journal* in the previous three years.

2. The prize will consist of a framed certificate and an honorarium of U.S. $1,000, courtesy of the publisher of the *Stata Journal*. The prize may be awarded in person at a Stata Conference or Stata Users Group meeting of the recipient’s or recipients’ choice or as otherwise arranged.

3. Nominations for the prize in a given year will be requested in the *Stata Journal* in the first issue of each year and simultaneously through announcements on the *Stata Journal* website and on Statalist. Nominations should be sent to the editors by private email to editors@stata-journal.com by July 31 in that year. The recipient(s) will be announced in the *Stata Journal* in the last issue of each year and simultaneously through announcements on the *Stata Journal* website and on Statalist.

4. Nominations should name the author(s) and one or more papers published in the *Stata Journal* in the previous three years and explain why the work concerned is worthy of the prize. The precise time limits will be the annual volumes of the *Stata Journal*, so that, for example, the prize for 2017 will be for work published in the annual volumes for 2014, 2015, or 2016. The rationale might include originality, depth, elegance, or unifying power of work; usefulness in cracking key problems or allowing important new methodologies to be widely implemented; and clarity or expository excellence of the work. Comments on the excellence of the software will also be appropriate when software was published with the paper(s). Nominations might include evidence of citations or downloads or of impact either within or outside the community of Stata users. These suggestions are indicative rather than exclusive, and any special or unusual merits of the work concerned may naturally be mentioned. Nominations may also mention, when relevant, any body of linked work published in other journals or previously in the *Stata Journal* or *Stata Technical Bulletin*. Work on any or all of statistical analysis, data management, statistical graphics, and Stata or Mata programming may be nominated.

© 2017 StataCorp LLC
5. Nominations will be considered confidential both before and after award of the prize. Neither anonymous nor public nominations will be accepted. Authors may not nominate themselves and so doing will exclude those authors from consideration. The editors of the *Stata Journal* may not be nominated. Employees of StataCorp may not be nominated. Such exclusions apply to any person with such status at any time between January 1 of the year in question and the announcement of the prize. The associate editors of the *Stata Journal* may be nominated.

6. The recipient(s) of the award will be selected by the editors of the *Stata Journal*, who reserve the right to take advice in confidence from appropriate persons, subject to such persons not having been nominated themselves. The editors’ decision is final and not open to discussion.


H. Joseph Newton and Nicholas J. Cox
Editors, *Stata Journal*
Creating HTML or Markdown documents from within Stata using webdoc

Ben Jann
University of Bern
Bern, Switzerland
ben.jann@soz.unibe.ch

Abstract. In this article, I discuss the use of webdoc for creating HTML or Markdown documents from within Stata. The webdoc command provides a way to embed HTML or Markdown code directly in a do-file and automate the integration of results from Stata in the final document. The command can be used, for example, to create a webpage documenting your data analysis, including all Stata output and graphs. More generally, the command can be used to create and maintain a website that contains results computed by Stata.

Keywords: pr0065, webdoc, HTML, Markdown, weaving, Stata output, Stata log, reproducible research

1 Introduction

webdoc is a command to process a do-file that contains Stata commands as well as pieces of HTML or Markdown code. A primary use of webdoc is to produce an HTML document that displays literal Stata output as it is shown in Stata’s Results window. However, webdoc can be seen as a general tool for generating HTML documents that combine text sections and results from statistical analysis.

Several other user commands are available to support the production of HTML documents in Stata. For example, you can

- translate Stata output or Stata Markup and Control Language files to HTML format using commands such as log2html by Baum, Cox, and Rising (2001) or hlp2html by Jeanty (2010);
- create HTML documents from within Stata with tools such as the ht package by Quintó et al. (2012) or htmutil by Newson (2015);
- export tables or matrices into an HTML file with commands such as listtex by Newson (2001), matprint by Bruun (2016b), tabout by Watson (2004), or esttab by Jann (2007);
- and weave commands and HTML or Markdown code in a single do-file using tools such as log2markup by Bruun (2016a), weaver and markdoc by Haghish (2014a,b), or weave by Rodríguez (2015).
The `webdoc` command covers much of the functionality of these packages. Like `log2html`, it converts Stata output to HTML format (relying, in part, on the undocumented `log html` command); like the `ht` package or `htmlutil` command, it allows you to work on an HTML file from within Stata; like `log2markup` or `markdown`, it allows you to integrate HTML or Markdown code in a do-file. Furthermore, although `webdoc` does not provide specific tools for producing tables, results from special-purpose programs such as `listtex` can easily be integrated. A major difference from other weaving programs, such as `log2markup` or `markdown`, is that `webdoc` preprocesses the do-file. This provides some advantages, such as the ability to update the HTML output document without having to rerun all Stata commands. However, it also means that `webdoc` cannot be used interactively.

Below I will discuss the features of `webdoc` and provide examples of its usage (for further examples, see [http://repec.sowi.unibe.ch/stata/webdoc/](http://repec.sowi.unibe.ch/stata/webdoc/)). `webdoc` has a similar architecture, functionality, and user interface as `texdoc`, a command for producing LaTeX documents ([Jann 2016](#)). If you are familiar with `texdoc`, you will find `webdoc` easy to use. Additionally, knowledge of HTML and CSS will be beneficial. A good source for detailed information on HTML and CSS is [http://www.w3schools.com/](http://www.w3schools.com/). For information on Markdown, consult [http://daringfireball.net/projects/markdown/](http://daringfireball.net/projects/markdown/).

## 2 The `webdoc` command

### 2.1 Processing a do-file by `webdoc do`

The basic procedure is to write a do-file including Stata commands and sections of HTML code, then process the do-file by typing `webdoc do`. The command will create the HTML source file, which can then be viewed in a browser. One can also use Markdown code instead of HTML. In this case, a Markdown converter has to process the source document before it can be viewed in the browser. The syntax of `webdoc do` is

```
webdoc do filename [ arguments ] [ , options ]
```

where `filename` is the name of the do-file to be processed (as usual, include the file name in double quotes if it contains spaces) and `arguments` are optional arguments passed through to the do-file (as local macros 1, 2, 3, and so on; see `R` do). `options` are the following:

- `[no] init[ (docname) ]` initializes the output document. If the processed do-file contains an initialization command (that is, if the do-file contains `webdoc init docname`; see section 2.2) or if the output document is already open (that is, in a nested application of `webdoc do`), the default for `webdoc do` is not to initialize the output document. Otherwise, `webdoc do` will automatically initialize the output document in the folder of the do-file using `basename.html` (or, if the `md` option is specified, `basename.md`) as the name for the document, where `basename` is the name of the do-file without a suffix. Use the `init` option to override these defaults as follows:
noinit will deactivate automatic initialization; init will enforce automatic initialization; init(docname) will enforce initialization using docname as the name for the document (docname may include an absolute or relative path; the base folder is the current working directory or the folder of the do-file, depending on whether the cd option is specified).

init_options are options to be passed through to webdoc init. See section 2.2 for details on available options.

nostop allows continuing execution even if an error occurs. Use the nostop option if you want to make sure webdoc do runs the do-file all the way to the end, even if some of the commands return an error. Usage of this option is not recommended. Use the nostop option with webdoc stlog using if you want to log output from a command that returns an error (see section 2.5).

cd changes the working directory to the directory of the specified do-file for processing the do-file and restores the current working directory after termination. The default is not to change the working directory.

webdoc do can be nested. That is, webdoc do can be applied in a do-file that is processed by webdoc do. Options specified with a nested call to webdoc do will be applied only to the nested do-file. This is also true for applications of webdoc init or webdoc close within the nested do-file: after you terminate a nested do-file, all preexisting webdoc settings will be restored. For example, if you use the init() option or webdoc init to change the output document in the nested do-file, webdoc closes the new output document and switches back to the previous one when exiting the nested do-file (similarly, if you use webdoc close in the nested do-file, the document will be reopened after termination).

### 2.2 Initializing the output document

Within a do-file, use webdoc init to initialize the HTML or Markdown output document (alternatively, if the do-file does not contain an initialization command, webdoc do will automatically call webdoc init; see the init() option in section 2.1). The syntax of webdoc init is

```
webdoc init [ docname ] [, init_options]
```

`docname` is the name of the HTML or Markdown target file, possibly including a path. You may also apply webdoc init without docname in later parts of the do-file to change settings. init_options are the following:

- **replace** allows you to overwrite an existing output document.
- **append** appends results to an existing output document.
- **md** specifies that .md instead of .html be used as the default suffix for the output document.
Creating HTML or Markdown documents using webdoc

header[ (header_opts) ] causes an HTML header (and a footer) to be added to the output document. header_opts are as follows.

width(width) sets the maximum width of the HTML page, where width is a width specification in CSS units (see http://www.w3schools.com/cssref/css_units.asp), such as 800px or 50em. If you use the btheme() option, an alternative approach is to include the body of your page in a container. For example, type <div class="container-fluid" style="max-width:800px"> on the first line and </div> on the last line.

nofooter omits the footer. This is useful if you want to append more material to the same document later.

title(str) provides a title for the metadata of the page. The default is to use the name of the document as the title.

author(str), date(str), description(str), and keywords(str) provide author information, a date, a description, and a (comma-separated) list of keywords to be included in the metadata of the page.

language(str) specifies the document’s language. str is an HTML language specification (see https://www.w3.org/International/articles/language-tags/). The default is language(en).

charset(str) specifies the document’s character encoding. str is an HTML charset specification (see http://www.w3schools.com/html/html_charset.asp). The default depends on the Stata version. If you are using Stata 13 or older, the default is charset(iso-8859-1) (Windows and Unix) or charset(mac) (Mac OS X). If you are using Stata 14 or newer, the default is charset=utf-8.

btheme[ (spec) ] puts a Bootstrap CSS (see http://getbootstrap.com) in the header. spec is

[ theme ] [, jscript selfcontained ]

where theme is equal to default (for the default Bootstrap CSS) or equal to the name (in lowercase letters) of a Bootswatch theme (such as cerulean or cosmo; see http://bootswatch.com or https://www.bootstrapcdn.com/bootswatch/ for the list of available themes). If theme is omitted, the default Bootstrap CSS is used. In addition to the Bootstrap CSS, webdoc will append a few additional CSS definitions to sightly modify the display of images and code. Furthermore, if you use the btheme() option, you should consider specifying a maximum page width using the width() option or including the body of your page in a container, for example, typing <div class="container-fluid" style="max-width:800px"> on the first line and </div> on the last line. For more information on Bootstrap, see http://getbootstrap.com.

By default, webdoc does not load Bootstrap’s JavaScript plugins. Specify the jscript suboption if you want to use Bootstrap elements that require JavaScript.
webdoc will then add code at the end of the document to load the relevant plugins (also see http://getbootstrap.com/getting-started/#template).

Unless the selfcontained suboption is specified, webdoc includes the Bootstrap CSS and JavaScript plugins using links pointing to the minified files at https://www.bootstrapcdn.com. Specify selfcontained to copy the nonminified versions of the files into your document (this will increase the file size of your document by about 150 KB or, if jscript is specified, by about 500 KB). For larger projects, it may make sense to provide a copy of the CSS and JavaScript files at your website and include them in your HTML pages using local links.

If the bstheme option is omitted, a minimum set of CSS definitions resulting in a plain look will be included in the header of the document.

include(filename) adds the contents of filename to the HTML header. The contents of filename will be included within the <head> tag after the definitions requested by the bstheme() option.

stscheme(stscheme_options) specifies the look of the Stata output sections. This has an effect on sections containing Stata output only, not on sections containing Stata code. That is, sections created by the cmdlog option (see below) will not be affected by stscheme(). Note that, currently, webdoc does not tag errors and links in the Stata logs, so these elements will appear as regular output. stscheme_options are the following:

standard, studio, classic, desert, mountain, ocean, or simple selects one of Stata’s built-in color schemes (see the preferences dialog of Stata’s Results window; you can right-click on the Results window to open the dialog box).

bg(color), fg(color), rfg(color), cfg(color), rbf, and cbf affect the appearances of the different elements in the Stata output, where color is a CSS color specification (see http://www.w3schools.com/colors/default.asp). These options override the corresponding settings from the built-in schemes. bg() specifies the background color, fg() the default foreground color (that is, the color of standard output), rfg() the color of results (typically the numbers in the output), and cfg() the color of input (the commands). Furthermore, use rbf and cbf to request bold font for results and input/commands, respectively.

lcom italicizes and shades comments in the Stata output.

[no]logall specifies whether to include the output of all Stata commands in the output document. The default is nologall; that is, include the output selected only by webdoc stlog (see section 2.5). Specify logall if you want to log all output. When logall is specified, webdoc do will insert appropriate webdoc stlog and webdoc stlog close commands automatically at each /*** ***/ block and at each webdoc command (but not at webdoc stlog oom and webdoc stlog cnp). Empty lines (or lines that contain white space only) at the beginning and end of each command section will be skipped.

stlog_options sets the default behavior of webdoc stlog. See section 2.5 for details.
Creating HTML or Markdown documents using webdoc

\texttt{gropts(graph_options)} specifies default options to be passed through to \texttt{webdoc graph}. See section 2.6 for details. Updating \texttt{gropts()} in repeated calls to \texttt{webdoc init} will replace the option as a whole.

\texttt{[no]logdir\{\texttt{path}\}} specifies where to store the Stata output log files. The default is \texttt{nologdir}, in which case the log files are stored in the same directory as the output document, using the name of the output document as a prefix for the names of the log files; also see the \texttt{prefix()} option below. The \texttt{logdir} option without an argument causes the log files to be stored in a subdirectory with the same name as the output document. The \texttt{logdir(path)} option causes the log files to be stored in subdirectory \texttt{path}, where \texttt{path} is a relative path starting from the folder of the output document.

\texttt{grdir(path)} specifies an alternative subdirectory to be used by \texttt{webdoc graph} for storing the graph files, where \texttt{path} is a relative path starting from the folder of the output document. The default is to store the graphs in the same directory as the log files.

\texttt{dodir(path)} specifies an alternative subdirectory to be used by \texttt{webdoc stlog} for storing the do-files requested by the \texttt{dosave} option (see below), where \texttt{path} is a relative path starting from the folder of the output document. The default is to store the do-files in the same directory as the log files.

\texttt{[no]prefix\{\texttt{prefix}\}} specifies the prefix for the automatic names of the Stata output log files and graphs. The names are constructed as \texttt{prefix\#}, where \texttt{\#} is a counter (1, 2, 3, etc.). The \texttt{noprefix} option omits the prefix; the \texttt{prefix} option without an argument causes \texttt{basename} to be used as the prefix, where \texttt{basename} is the name of the output document without a path; the \texttt{prefix(prefix)} option causes \texttt{prefix} to be used as the prefix. The default prefix is empty if \texttt{logdir} or \texttt{logdir(path)} is specified; otherwise, the default prefix is equal to \texttt{basename} (note that reinitializing \texttt{logdir} may reset the prefix). Furthermore, the prefix will be ignored if a custom \texttt{name} is provided when calling \texttt{webdoc stlog} (see section 2.5). The suffix of the physical log files on disk is always \texttt{.log}.

\texttt{[no]stpath\{\texttt{path}\}} specifies how the path for linking files in the output document is to be constructed. \texttt{stpath()} has no effect on where the log files and graphs are stored in the file system. If \texttt{stpath} is specified without an argument, then the path of the output document (to be precise, the path specified in \texttt{docname} when initializing the output document) is added to the include-path. Alternatively, specify \texttt{stpath(path)} to add a custom path. The default is \texttt{nostpath}.

2.3 Including HTML or Markdown code

After initializing the output document, use

\texttt{/***/ text /***/}
B. Jann

to include a section of HTML or Markdown code. text can contain any text, including multiple lines and paragraphs. The opening tag of an HTML or a Markdown section, /***, must be at the beginning of a line (possibly preceded by white space) and must be followed by at least one blank or a line break; the closing tag, ***/, must be at the end of a line (possibly followed by white space) and must also be preceded by at least one blank or a line break. The provided text will be passed through to the output document as is, that is, without expanding Stata macros (although see section 3.10). However, you can use the webdoc substitute command to define a set of substitutions that will be applied to the text. The syntax of webdoc substitute is

```
webdoc substitute [from to [from to ...]] [, add]
```

The substitutions defined by webdoc substitute will be applied to all subsequent /***/ ***/ blocks until a new set of substitutions is defined or until the substitutions are turned off by calling webdoc substitute without arguments. To extend an existing set of substitution definitions, specify webdoc substitute with the add option.

A single line of HTML or Markdown code can also be written to the document using the following:

```
webdoc write textline
```
or
```
webdoc put textline
```

Stata macros in textline will be expanded before writing the line to the output document. The difference between webdoc write and webdoc put is that webdoc put includes a new-line character at the end of the line, whereas webdoc write omits the new-line character so more text can be added to the same line. Furthermore, to copy the contents of an external file to the output document, type

```
webdoc append filename [, substitute(from to [from to ...]) drop(numlist)]
```

where filename is the name (and path) of the file to be added. The contents of filename will be copied into the output document as is, at the position where webdoc append is specified. If substitute() is specified, all occurrences of from will be replaced by to. Include from and to in double quotes if they contain spaces. For example, to replace “@title” by “My Title” and “@author” by “My Name”, you could type substitute(@title "My Title" @author "My Name"). The drop() option causes the specified lines to be omitted when copying the file.
2.4 Adding a table of contents

An automatic table of contents (TOC) from the headings in the document can be generated by webdoc toc. The syntax of webdoc toc is

```
webdoc toc [levels [offset]] [, toc_options]
```

webdoc toc collects the HTML headings found in subsequent `/*** */*/*` blocks and constructs a corresponding TOC (using `<ul>` lists). The TOC will be inserted into the output document at the position where webdoc toc appears. The `levels` argument specifies the desired number of levels to be considered. For example, `webdoc toc 3` will create a TOC with three levels from `<h1>` to `<h3>`. Furthermore, use the `offset` argument to shift the highest level to be taken into account. For example, `webdoc toc 3 1` will use `<h2>`, `<h3>`, and `<h4>`; `webdoc toc 2 4` will use `<h5>` and `<h6>`. `offset` must be an integer between 0 and 5; the default is 0. `levels` must be an integer between 1 and $6 - offset$; the default is 3. `toc_options` are the following:

- `numbered` specifies that section numbers be added to the headings and entries in the TOC. The numbers added to the headings will be tagged by `<span class="heading-secnum">`; the numbers in the TOC will be tagged by `<span class="toc-secnum">`.
- `md` specifies that Markdown headings be taken into account. By default, HTML headings only, that is, lines starting with `<h1>` to `<h6>`, are collected. If `md` is specified, lines starting with `#` to `#####` are also treated as headings. In any case, a heading will be detected only if it starts at the beginning of the line (save white space in the case of HTML tags). When webdoc constructs an entry in the TOC, only the text that follows on the same line will be taken into account.

2.5 Including Stata output

If the `logall` option is specified with `webdoc do` or `webdoc init`, output from all Stata commands will automatically be added to the HTML document. Alternatively, select the output to be included using the `webdoc stlog` command. The syntax of `webdoc stlog` is

```
webdoc stlog [name] [, stlog_options]
commands ...
webdoc stlog close
```

where `webdoc stlog` opens the log, `commands` are the Stata commands to be logged, and `webdoc stlog close` closes the log. `name` is the name to be used for the log file (possibly including a relative path). If `name` is omitted, an automatic name is used (see the `prefix()` option in section 2.2 for details). Alternatively, you may type
webdoc stlog [name] using dofile [, stlog_options]

where dofile is the name (and path) of an external do-file that contains the Stata commands to be logged. Furthermore, to include just the output of a single command (without input), you can type

webdoc stlog [name] [, stlog_options] : command

(note that webdoc stlog close is not needed after the using form or the colon form of webdoc stlog). stlog_options are the following:

\textbf{linesize(#)} sets the line width (number of characters) to be used in the output log. # must be an integer between 40 and 255. The default is to use the current set linesize setting; see \textit{[R] log}.

\textbf{[no]} do specifies whether to run the Stata commands. The default is do; that is, run the commands. Type nodo to skip the commands and not write a new log file. nodo is useful if the Stata commands have been run before and did not change. For example, specify nodo if the Stata output is complete and you want to work on the text without having to rerun the Stata commands. Be aware that the automatic names of Stata output sections change if the order of Stata output sections changes. That is, nodo should be used only if the order did not change or if a fixed name was assigned to the Stata output section. An exception is if nodo is used together with the cmdlog option (see below). In this case, the log file will always be re-created (because running the commands is not necessary to re-create the log file).

\textbf{[no]} log specifies whether the Stata output is to be logged and included in the output document. The default is log; that is, log and include the Stata output. If you type nolog, the commands will be run without logging. nolog does not appear to be particularly useful, because you could simply include the corresponding Stata commands in the do-file without using webdoc stlog. However, nolog may be helpful in combination with the nodo option. It provides a way to include unlogged commands in the do-file that will not be executed if nodo is specified. Furthermore, nolog can be used to deselect output if the logall option has been specified.

\textbf{[no]} cmdlog specifies whether to print a plain copy of the Stata code instead of using a Stata output log. The default is nocmdlog; that is, include a Stata output log. If you type cmdlog, only a copy of the commands without output will be included (note that the commands will still be executed; add the nodo option if you want to skip running the commands). cmdlog is similar to nooutput (see below). A difference is that nooutput prints ". " at the beginning of each command, whereas cmdlog displays a plain copy of the commands. Furthermore, cmdlog can be combined with nodo to include a copy of the commands without executing the commands. Tag \texttt{<pre class="stcmd">}<code> will be used to start a cmdlog section in the output document. Other Stata output sections will be started by \texttt{<pre class="stlog">}<samp>. cmdlog is not allowed with the colon form of webdoc stlog.
Creating HTML or Markdown documents using webdoc

[no] dosave specifies whether to store a copy of the commands in an external do-file. The default is nodosave; that is, do not store a do-file. The name of the Stata output section is used as the name for the do-file (with suffix .do). The do-files will be stored in the same location as the log files unless an alternative location is specified using the dodir() option. All webdoc commands will be stripped from the do-file.

[no] output specifies whether to suppress command output in the log. The default is output; that is, display the output. If nooutput is specified, set output inform is applied before running the commands and, after closing the log, set output proc is applied to turn output back on (see [P] quietly). nooutput has no effect if cmdlog is specified. Furthermore, nooutput has no effect if specified with the using form or the colon form of webdoc stlog.

[no] matastrip specifies whether to strip the Mata opening and ending commands from the Stata output. The default is nomatastrip; that is, retain the Mata opening and ending commands. If you type matastrip, the mata or mata: command invoking Mata and the subsequent end command exiting Mata will be removed from the log. matastrip has an effect only if the Mata opening command is the first command in the output section.

[no] cmdstrip specifies whether to strip command lines (input) from the Stata output. The default is nocmdstrip; that is, retain the command lines. Specify cmdstrip to delete the command lines. Specifically, all lines starting with ".." (or "::" in Mata) and subsequent lines starting with "->" will be removed. cmdstrip has no effect if cmdlog is specified.

[no] lbstrip specifies whether to strip line-break comments from command lines in the Stata output. The default is noblbstrip; that is, do not strip the line-break comments. Specify lbstrip to delete the line-break comments. Specifically, "//" at the end of lines starting with ".." or of subsequent lines starting with "->" will be removed.

[no] gtstrip specifies whether to strip continuation symbols from command lines in the Stata output. The default is nogtstrip; that is, do not strip the continuation symbols. Specify gtstrip to delete the continuation symbols. Specifically, " " at the beginning of command lines that were broken by a line-break comment will be replaced by white space. gtstrip has no effect if cmdlog is specified.

[no] ltrim specifies whether to remove indentation of commands (that is, whether to remove white space on the left of commands) before running the commands and creating the log. The default is ltrim; that is, remove indentation. The amount of white space to be removed is determined by the minimum indentation in the block of commands. ltrim has no effect on commands called from an external do-file by webdoc stlog using.
mark(strlist) adds the <mark> tag to all occurrences of the specified strings, where strlist is

\[
{\text{string [ string ...]}}
\]

Enclose string in double quotes if it contains blanks; use compound double quotes if it contains double quotes.

tag(matchlist) applies custom tags to all occurrences of the specified strings, where matchlist is

\[
{\text{strlist = begin end [ strlist = begin end ...]}}
\]

and strlist is

\[
{\text{string [ string ...]}}
\]

strlist specifies the strings to be tagged, begin specifies the start tag, and end specifies the end tag. Enclose an element in double quotes if it contains blanks; use compound double quotes if the element contains double quotes.

[no]plain specifies whether to omit markup in the log file. The default is noplain; that is, annotate the log file with HTML tags. In particular, input (commands) will be tagged using <span class="stinp">, results will be tagged using <span class="stres">, and comments will be tagged using <span class="stcmt" > (if cmdlog is specified, only comments will be tagged). Specify plain to omit the HTML tags.

[no]raw specifies whether to omit markup in the log file and retain special characters. The default is noraw; that is, annotate the log file with HTML tags (see the plain option above) and replace characters <, >, and & by their HTML equivalents &lt;, &gt;, and &amp;. Specify raw to omit the HTML tags and retain the special characters.

[no]custom specifies whether to use custom code to include the log file in the output document. The default is nocustom; that is, use standard code to include the log. Specify custom if you want to skip the standard code, and be careful including the log yourself.

[no]keep specifies whether the external log file will be kept. The default is keep; that is, keep the log file so nodo can be applied later. Type nokeep if you want to erase the external log file.

[no]certify specifies whether to compare the current results with the previous version of the log file (if a previous version exists). The default is nocertify. Specify certify if you want to confirm that the output did not change. In case of a difference, webdoc will stop execution and display an error message. certify has no effect if nolog or cmdlog is specified or if a help file is processed (see the sthlp option below).

[no]sthlp([subst]) specifies whether to treat the provided file as a Stata help file. This is allowed only with webdoc stlog using. By default, files with an .hlp or .sthl suffix are treated as help files; all other files are treated as do-files. Type
Creating HTML or Markdown documents using webdoc

nosthlp or sthlp to override these defaults. Files treated as help files are translated by undocumented log webhtml (or, if plain or raw is specified, by translate with the smcl2log translator) and are not submitted to Stata for execution. Unless plain or raw is specified, text markup and help links are preserved. Internal help links (that is, links pointing to the processed help file) will be converted to appropriate internal links in the output document; other help links will be converted to links pointing to the corresponding help file at http://www.stata.com/. In addition, you may provide custom substitutions in sthlp(subst), where subst is a space-separated list (from to [from to ... ]). The custom substitutions will be applied before converting the internal links and the stata.com links (unless plain or raw is specified, in which case no substitutions will be applied). The help links written by log webhtml are constructed as <a href="/help.cgi?...">. Hence, you could, for example, type sthlp(/help.cgi?mycommand mycommand.html) to convert the help links for mycommand to links pointing to the local page mycommand.html.

The nolog, cmdlog, and dosave options are not allowed in help-file mode. Furthermore, contents options such as nooutput, cmdstrip, or matastrip will have no effect. However, you may use nodo to prevent reprocessing the help file or custom to use custom-inclusion code. By default, the included help file will be wrapped by a <pre class="sthlp"> tag.

nostop allows continuing execution even if an error occurs. Use the nostop option if you want to log output from a command that returns an error. The nostop option is allowed only with webdoc stlog using.

Furthermore, among the commands to be logged, you can use

webdoc stlog com command

to suppress the output of a specific command and display an “output omitted” message instead, or you can type

webdoc stlog quietly command

to suppress the output of a command without inserting an “output omitted” message, and you can type

webdoc stlog cnp

to insert a “continued on the next page” message and a page break (page breaks are ignored in the screen display of an HTML page, but they affect the print version of the page). The “output omitted” message is produced by

<\span class="stoom">(output omitted)</span>
The class attribute is set so you can use stylesheets to affect the appearance of these messages. For example, including

```html
<style type="text/css">
.stoom, .stcnp { font-style: italic; }
@media screen { .stcnp { display: none; } }
</style>
```

in the header of the output document will use italics for the messages and suppress the “continued on next page” message in screen display (where page breaks have no effect).

Within or after a Stata output section, you can use the `webdoc local` command to define local macros that will be backed up on disk. This is useful if you want to include specific results in your text and want to ensure that the results will be available in later runs when suppressing the Stata commands using the `nodo` option. The syntax of `webdoc local` is

```
webdoc local name definition
```

where possible definitions follow the same syntax as Stata’s `local` command; see [P] macro. The locals will be backed up in a library that has the same name as the Stata output section (using file suffix `.stloc`). Each output section has its own library, so the names of the locals can be reused between sections.

The defined locals will be expanded in subsequent `/*** ***/**` blocks up until the next `webdoc stlog` command. Alternatively, you can write the locals to your document using `webdoc put` or `webdoc write`. See the example in section 3.10 below.

### 2.6 Including graphs

`webdoc graph` exports the current graph and includes the appropriate code in the output document to display the graph. `webdoc graph` can be specified within a `webdoc stlog` section or directly after `webdoc stlog` close. If `webdoc graph` is specified within a `webdoc stlog` section, the graph is included in the output document before the Stata output; if `webdoc graph` is specified after `webdoc stlog` close, the graph is included after the Stata output. Furthermore, if `webdoc graph` is used outside a `webdoc stlog` section while `logall` is on, the graph will be placed at the position in the output where the `webdoc graph` command occurs. In general, if `nodo` is on, no graph will be exported, and only the include code will be written to the output document. The syntax of `webdoc graph` is

```
webdoc graph [name] [, graph_options]
```
Creating HTML or Markdown documents using webdoc

`name` specifies the name to be used for the graph. If `name` is omitted, the name of the `webdoc stlog` section is used to name the graph (possibly suffixed by a counter if the `webdoc stlog` section contains more than one `webdoc graph` command). `graph_options` are the following:

`as(fileformats)` sets the output formats. The default is `as(png)`. See [G-2] graph export for available formats. A further, currently undocumented format available since Stata 14 is `as(svg)` (scalable vector graphics). Multiple formats may be specified as in, for example, `as(png pdf)`, in which case `webdoc graph` will create multiple graph files. The first format will be used for the image in the output document.

`name(name)` specifies the name of the Graph window to be exported. The default is to export the topmost graph.

`width(#)` specifies the physical width of the graph in pixels. The default is `width(500)` unless `height()` is specified. If `height()` is specified, the appropriate width is determined from the graph's aspect ratio. `width()` has an effect only if the output format is `.png` or `.tiff`.

`height(#)` specifies the physical height of the graph (in pixels). The default height is determined from the graph's aspect ratio. `height()` has an effect only if the output format is `.png` or `.tiff`.

`override_options` modifies how the graph is converted. See [G-2] graph export for details.

`alt(string)` provides an alternative text for the image to be added to the `<img>` tag using the “alt” attribute. The default is to use the name of the graph as alternative text. The `alt()` option has no effect if embedding an SVG using the `hardcode` option.

`title(string)` provides a “tooltip” title for the image to be added to the `<img>` tag using the “title” attribute.

`attributes(args)` adds further attribute definitions to the `<img>` tag. For example, to set the display width of the graph to 50%, type `attributes(width="50\%")`.

`[no]link(fileformat)` specifies whether to add a link to the image pointing to the graph file. Clicking on the image in the browser will then open the graph file. The default is `link`; that is, add a link, unless `hardcode` is specified (see below), in which case `nolink` is the default. The `fileformat` argument may be used to select the file for the link if multiple output formats have been requested by the `as()` option. For example, specifying `link(pdf)` together with `as(svg pdf)` will display the SVG image and use the PDF for the link. The default is to use the first format for both the image and the link.
\[ \text{figure}(id) \] specifies whether to enclose the image in a \text{<figure>} environment. The default is \text{figure}; that is, use the figure tag. Type \text{nofiregure} to omit the figure tag. To add a custom ID to the figure tag, type \text{figure(id)}. If \text{id} is omitted, \text{webdoc} will add an automatic ID (constructed as \text{fig-name}, where \text{name} is the base name of the graph).

\text{caption(string)} provides a caption for the figure using the \text{<figcaption>} tag.

\text{cabove} or \text{cbelow} specifies whether the caption is printed above or below the figure. Only one of \text{cabove} and \text{cbelow} is allowed. \text{cbelow} is the default.

\[ \text{no} \text{hardcode} \] specifies whether to embed the graph source in the output document. This is supported only for \text{.png} and \text{.svg}. For \text{.png}, the graph file will be embedded using Base64 encoding. For \text{.svg}, the SVG code will be copied into the output document. The default is \text{nohardcode}; that is, include the graph using a link to the external graph file.

\[ \text{no} \text{keep} \] specifies whether the external graph file (and its Base64 variant) will be kept. This is relevant only if \text{hardcode} has been specified. The default is \text{keep}; that is, keep the graph files so \text{nodo} can be applied later. Type \text{nokepp} if you want to erase the external graph files.

\[ \text{no} \text{custom} \] specifies whether to use custom code to include the graph in the output document. The default is \text{nocustom}, in which case \text{webdoc graph} writes code to the output document to include the graph. Specify \text{custom} if you want to skip the standard code, and be careful including the graph yourself.

### 2.7 Changing the HTML settings for Stata output and graphs

Parts of the HTML code written by \text{webdoc} can be customized by the \text{webdoc set} command. The syntax of \text{webdoc set} is

\text{webdoc set } [ \text{setname } [ \text{definition} ] ]

where \text{setname} is the name of the element you want to change. To restore the default settings for all elements, type \text{webdoc set} without an argument. \text{webdoc set} has an effect only if applied within a do-file processed by \text{webdoc do}. Furthermore, all settings will be removed when \text{webdoc do} terminates. The elements you can modify, and their default definitions, are in table 1.
Table 1. HTML settings that can be changed by `webdoc set`  

<table>
<thead>
<tr>
<th>Description</th>
<th><code>setname</code></th>
<th>Default definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stata output section</td>
<td><code>stlog</code></td>
<td><code>&lt;pre id=&quot;id&quot; class=&quot;stlog&quot;&gt;&lt;samp&gt;</code></td>
</tr>
<tr>
<td>Stata code section</td>
<td><code>stcmd</code></td>
<td><code>&lt;pre id=&quot;id&quot; class=&quot;stcmd&quot;&gt;&lt;code&gt;</code></td>
</tr>
<tr>
<td>Stata help section</td>
<td><code>sthlp</code></td>
<td><code>&lt;pre id=&quot;id&quot; class=&quot;sthlp&quot;&gt;</code></td>
</tr>
<tr>
<td>Stata input tag</td>
<td><code>stinp</code></td>
<td><code>&lt;span class=&quot;stinp&quot;&gt;</code></td>
</tr>
<tr>
<td>Stata result tag</td>
<td><code>stres</code></td>
<td><code>&lt;span class=&quot;stres&quot;&gt;</code></td>
</tr>
<tr>
<td>Stata comment tag</td>
<td><code>stcmt</code></td>
<td><code>&lt;span class=&quot;stcmt&quot;&gt;</code></td>
</tr>
<tr>
<td>Output-omitted tag</td>
<td><code>stoom</code></td>
<td><code>&lt;span class=&quot;stoom&quot;&gt;(output omitted)&lt;/span&gt;</code></td>
</tr>
<tr>
<td>Continued on next page tag</td>
<td><code>stcnp</code></td>
<td><code>&lt;br&gt;(continued on next page)&lt;br&gt;</code></td>
</tr>
<tr>
<td>Figure tag</td>
<td><code>figure</code></td>
<td><code>&lt;figure id=&quot;\macval(id)&quot;&gt;</code></td>
</tr>
<tr>
<td>Figure caption</td>
<td><code>fcap</code></td>
<td><code>&lt;figcaption&gt;</code></td>
</tr>
<tr>
<td>Figure link tag</td>
<td><code>flink</code></td>
<td><code>&lt;a href=&quot;\webname\suffix&quot;&gt;</code></td>
</tr>
<tr>
<td>Image tag</td>
<td><code>img</code></td>
<td><code>&lt;img alt=&quot;\macval(alt)&quot; src=&quot;\macval(title)&quot;/&gt;</code></td>
</tr>
<tr>
<td>Embedded SVG</td>
<td><code>svg</code></td>
<td><code>&lt;span class=&quot;macval(attributes)&quot;&gt;</code></td>
</tr>
</tbody>
</table>

Names without an underscore refer to opening tags (or opening and closing tags), and names with an underscore refer to closing tags. As illustrated by the default settings, some of the elements make use of local macros, with a leading backslash for delayed expansion. An interesting additional macro that can be used in `stlog/_stlog` and `stcmd/_stcmd` is `\doname`, containing the name of the do-file that is generated if the `dosave` option has been specified. For example, to provide a download link for the do-file in the upper right corner of each output section, you could type the following:

```bash
webdoc set stlog <pre id="id" class="stlog">/*
   */style="position:relative;"<a href="\doname"/>
   */style="position:absolute;top:5px;right:5px">[code]<a>
</samp>
```

SVG images embedded in the output document using the `hardcode` option will be tagged by `svg/_svg`. For all other graphs, `img/_img` will be used.
2.8 Closing the output document and exiting the do-file

The syntax to stop writing to the output document is

\[ \text{webdoc close} \]

\[ \text{webdoc do} \] closes the output document automatically at the end of the do-file, so \text{webdoc close} is usually not needed.

To cause \text{webdoc do} to exit a do-file, type

\[ // \text{webdoc exit} \]

(without anything else on the same line). \text{webdoc do} will only read the do-file up to this line.

2.9 Stripping webdoc commands from a do-file

To clear a do-file from all \text{webdoc} commands, use

\[ \text{webdoc strip filename newname [ , replace append]} \]

where \text{filename} is the name of the do-file to be stripped and \text{newname} is the name of the file to be written to. The \text{replace} option replaces an existing file; the \text{append} option appends the results to an existing file. \text{webdoc strip} removes all \text{/*** ***/} blocks and all \text{webdoc} commands from the do-file.

2.10 Stored results

\text{webdoc init} clears \text{s()}, and \text{webdoc close} stores the following in \text{s()}:

Macros

\begin{verbatim}
  s(docname)  name of output document
  s(basename) base name of output document
  s(path)     (absolute) path of output document
  s(md)       md or empty
  s(logall)   logall or empty
  s(linesize) specified line width or empty
  s(nodo)     nodo or empty
  s(nolog)    nolog or empty
  s(cmdlog)   cmdlog or empty
  s(dosave)   dosave or empty
  s(plain)    plain or empty
  s(raw)      raw or empty
  s(nouput)   nouput or empty
  s(matastrip) matastrip or empty
  s(cmdstrip) cmdstrip or empty
  s(lbstrip)  lbstrip or empty
  s(gtstrip)  gtstrip or empty
  s(noltrin)  noltrin or empty
  s(gtstrip)  contents of \text{mark()} option
  s(tag)      contents of \text{tag()} option
  s(custom)   custom or empty
  s(nok)      nokeep or empty
  s(certify)  certify or empty
  s(gropts)   default graph export options
  s(logdir)   subdirectory used for Stata log files
  s(grdir)    subdirectory used for graphs
  s(cmdstrip) output document
  s(prefix)   prefix for automatic names
  s(stpath)   include-path to be used in the output document
\end{verbatim}
Creating HTML or Markdown documents using webdoc

webdoc stlog close and webdoc stlog using store the following in $s()$:

Macros

- `$s(name)$`: name of the Stata output log
- `$s(name0)$`: name of the Stata output log without `logdir()` path
- `$s(filename)$`: name of log file on disk (including path and suffix)
- `$s(filename0)$`: name of log file on disk without suffix
- `$s(webname)$`: name of log file with include path for use in output document
- `$s(webname0)$`: name of log file with include path for use in output document without suffix
- `$s(id)$`: ID of the log in the output document
- `$s(doname)$`: name (and include-path) of do-file
- `$s(linesize)$`: line width used for the output log
- `$s(indent)$`: size of indentation
- `$s(mark)$`: contents of `mark()` option
- `$s(tag)$`: contents of `tag()` option
- `$s(nokeep)$`: nokeep or empty
- `$s(certify)$`: certify or empty
- `$s(noltrim)$`: noltrim or empty
- `$s(dosave)$`: dosave or empty
- `$s(log)$`: nolog or empty
- `$s(plain)$`: plain or empty
- `$s(raw)$`: raw or empty
- `$s(cmdlog)$`: cmdlog or empty
- `$s(cmdstrip)$`: cmdstrip or empty
- `$s(matastrip)$`: matastrip or empty
- `$s(lbstrip)$`: lbstrip or empty
- `$s(gtstrip)$`: gtstrip or empty
- `$s(nooutput)$`: noutput or empty
- `$s(cmdlog)$`: cmdlog or empty
- `$s(filename)$`: name of log file on disk (including path and suffix)
- `$s(filename0)$`: name of log file on disk without suffix
- `$s(nolog)$`: nolog or empty
- `$s(nodoc)$`: nodo or empty
- `$s(coeffs)$`: coeffs or empty
- `$s(custom)$`: custom or empty
- `$s(linesize)$`: line width used for the output log
- `$s(indent)$`: size of indentation

3 Examples

3.1 Basic usage

A simple do-file using `webdoc` might look as follows:

```plaintext
webdoc init example1, replace logall plain

/****
<html>
<head><title>Example 1</title></head>
<body>
<h2>Exercise 1</h2>
<p>Open the 1978 automobile data and run a regression of price on mileage using the <code>regress</code> command.</p>

sysuse auto
regress price mpg

</body>
</html>

/****
```

The `logall` option has been specified, so all Stata output is included in the HTML document. (In addition, the `plain` option has been specified to omit HTML tags from the Stata output, so the display of the HTML file below fits the page.) To process the file, type

```
.webdoc do example1.do
```
This will create file `example1.html` with the following contents:

```html
<html>
<head><title>Example 1</title></head>
<body>
<h2>Exercise 1</h2>
<p>Open the 1978 automobile data and run a regression of price on mileage using the `regress` command.</p>
<pre id="stlog-1" class="stlog">. sysuse auto
(1978 Automobile Data)
. regress price mpg

Source | SS df MS Number of obs = 74
---------|--------|-------|------------------|
Model | 139449474 1 139449474 Prob &gt; F = 0.0000
Residual | 495615923 72 6883554.48 R-squared = 0.2196
---------|--------|-------|------------------|
Total | 635065396 73 8699525.97 Root MSE = 2623.7

------------------------------------------------------------------------------
price | Coef. Std. Err. t P>|t| [95% Conf. Interval]
-------------|----------|--------|----------|---------|------------------|-------------|
mpg | -238.8943 53.07669 -4.50 0.000 -344.7008 -133.0879
_cons | 11253.06 1170.813 9.61 0.000 8919.088 13587.03

------------------------------------------------------------------------------
</pre></body>
</html>
```

Displaying the file in a browser would look like what is shown in figure 1.
Exercise 1

Open the 1978 automobile data and run a regression of price on mileage using the `regress` command.

```stata
. use auto
(1978 Automobile Data)
. regress price mpg
```

<table>
<thead>
<tr>
<th>Source</th>
<th>SS</th>
<th>df</th>
<th>MS</th>
<th>Number of obs</th>
<th>F(1, 72)</th>
<th>Prob &gt; F</th>
<th>R-squared</th>
<th>Adj R-squared</th>
<th>Root MSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
<td>139449474</td>
<td>1</td>
<td>139449474</td>
<td>74</td>
<td>20.26</td>
<td>0.0000</td>
<td>0.2195</td>
<td>0.2087</td>
<td>2623.7</td>
</tr>
<tr>
<td>Residual</td>
<td>6803594.48</td>
<td>72</td>
<td>8669525.97</td>
<td>74</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>635065396</td>
<td>73</td>
<td>8669525.97</td>
<td>74</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| price | Coef. Std. Err.   t | P>|t| | [95% Conf. Interval] |
|-------|-------------------|------|-----------------|---------------------|
| _cons | 11253.06 | 1170.813 | 9.61 | 0.000  | 8919.088 - 13587.03 |

Figure 1. Output file `example1.html` displayed in a browser

3.2 Using Markdown

For simplified typing, you could also omit the HTML tags and use Markdown instead. An example do-file might look as follows:

```stata
webdoc init example1, replace logall plain md

begin example1-md.do

### Exercise 1

Open the 1978 automobile data and run a regression of price on mileage using the `regress` command.

```stata
sysuse auto
regress price mpg
```

end example1-md.do
```

This will create a file, `example1.md`, that can then be converted to HTML using a Markdown converter. For example, if you have Pandoc (see http://pandoc.org/), you could type

```bash
shell pandoc example1.md -s -o example1.html
```

to create an HTML file from `example1.md`. The `-s` option has been specified, so Pandoc produces a standalone HTML file including a header and footer. The resulting file will look about the same as the file shown in figure 1.
3.3 Changing the look of the HTML file

Use stylesheet specifications in the header of the HTML file to change the look of the document in the browser. For example, using the following header definition would create a file that displays like what is shown in figure 2:

```html
<!DOCTYPE html>
<html>
<head>
<title>Example 2</title>
<style>
body {
  font-family: sans-serif; padding: 0 15px; max-width: 700px;
}

code {
  background-color: #f2f2f2; border-radius: 3px; padding: 3px;
}

pre {
  background-color: #f2f2f2;
  border-radius: 3px; padding: 12px;
}

pre code {
  background: transparent; padding: 0;
}
</style>
</head>
<body>
<h2>Exercise 1</h2>
<p>Open the 1978 automobile data and run a regression of price on mileage using the <code>regress</code> command.</p>

```
Exercise 1

Open the 1978 automobile data and run a regression of price on mileage using the `regress` command.

```
. sysuse auto
   (1978 Automobile Data)
. regress price mpg
```

```
<table>
<thead>
<tr>
<th>Source</th>
<th>df</th>
<th>MS</th>
<th>Number of obs</th>
<th>F(1,72)</th>
<th>Prob &gt; F</th>
<th>Adjusted R-squared</th>
<th>Root MSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
<td>1</td>
<td>139469476</td>
<td>139469476</td>
<td>20.26</td>
<td>0.0000</td>
<td>0.2196</td>
<td>0.0287</td>
</tr>
<tr>
<td>Residual</td>
<td>72</td>
<td>6993554.40</td>
<td>6993554.40</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>73</td>
<td>6599525.57</td>
<td>6599525.57</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| price | Coef. | Std. Err. | t | P>|t| | [95% Conf. Interval] |
|-------|-------|-----------|---|-----|---------------------|---------|
| mpg   | -230.8943 | 53.67669 | -4.50 | 0.000 | -344.7000 | -133.0879 |
| _cons | 11253.96  | 1176.813 | 8.61  | 0.000 | 8919.088  | 13587.03 |
```

Figure 2. Output file `example2.html` displayed in a browser

If you do not want to put together your own header (and footer), you can use the `header` option of `webdoc init` to generate an automatic header, as in the following example:

```
begin example3.do

webdoc init example3, replace logall
    header(title(Example 3) width(700px) stscheme(classic))
/** *
<h2>Exercise 1</h2>
<p>Open the 1978 automobile data and run a regression of price on mileage using the `<code>regress</code>` command.</p>
***/
sysuse auto
  (1978 Automobile Data)
regress price mpg

end example3.do
```

In the example, `title()` specifies the text for the `<title>` tag in the document header, `width()` sets the maximum page width, and `stscheme(classic)` selects the “Classic” color scheme for the Stata output (see figure 3).
Exercise 1

Open the 1978 automobile data and run a regression of price on mileage using the `regress` command.

```
. sysuse auto
(1978 Automobile Data)
. regress price mpg

| Source | SS    | df | MS    | Number of obs = 74
|--------|-------|----|-------|---------------------
| Model  | 139449474 | 1 | 139449474 | Prob > F = 0.0009
| Residual | 495615923 | 72 | 6883551.48 | R-squared = 0.2196
| Total  | 635065396 | 73 | 8699525.97 | Root MSE = 2623.7

|            | Coef. | Std. Err. | t     | P>|t| | 95% Conf. Interval |
|------------|-------|-----------|-------|-----|-------------------|
| price      | -229.8963 | 53.07669 | -4.31 | 0.000 | -344.7008 | -113.0879 |
| _cons      | 11263.06 | 1170.813 | 9.61  | 0.000 | 8919.088 | 13587.03 |
```

By default, if the `header()` option is specified, `webdoc` writes a minimal header so the page displays well on computer screens and mobile devices. Alternatively, you can use the `bstheme()` suboption to include a Bootstrap CSS file (see http://getbootstrap.com/).

For example, the code below includes the “United” theme from http://bootswatch.com/ and picks Stata’s “Desert” scheme for the output (see figure 4 for the result).

```bash
begin example4.do

webdoc init example4, replace logall ///
   header(title(Example 4) width(700px) stscheme(desert) bstheme(united))

/***
  <h2>Exercise 1</h2>
  <p>Open the 1978 automobile data and run a regression of price on mileage using the ```regress``` command.</p>
  ***/

  sysuse auto
  regress price mpg

end example4.do
```

Figure 3. Output file `example3.html` displayed in a browser
Exercise 1

Open the 1978 automobile data and run a regression of price on mileage using the \textit{regress} command.

\begin{verbatim}
. sysuse auto
(1978 Automobile Data)
. regress price mpg

\begin{tabular}{lccccc}
Source & SS & df & MS & Number of obs & = & 74 \\
Model & 139449474 & 1 & 139449474 & F(1, 72) & = & 20.26 \\
Residual & 495615923 & 72 & 6803554.68 & Prob > F & = & 0.0000 \\
Total & 635085396 & 73 & 8699525.93 & R-squared & = & 0.2196 \\
Adj R-squared & 0.2087 & \\
\end{tabular}

\begin{tabular}{lcccccc}
price & Coef. & Std. Err. & t & P>|t| & [95\% Conf. Interval] \\
mpg & -238.8943 & 58.87669 & -4.50 & 0.000 & -344.7008 & -133.0879 \\
_cons & 11253.86 & 1178.813 & 9.61 & 0.000 & 8019.088 & 14587.63 \\
\end{tabular}
\end{verbatim}

Figure 4. Output file \texttt{example4.html} displayed in a browser

\subsection{3.4 Contents of output sections}

In the examples above, the \texttt{logall} option was specified to create output sections from all Stata commands in the do-file. Alternatively, or in addition, you can use the \texttt{webdoc stlog} command to select the output to be included. For example, if the \texttt{logall} option has been specified, you could type

\begin{verbatim}
webdoc stlog, nolog
commands
webdoc stlog close
\end{verbatim}

to skip creating an output section from \texttt{commands}. Furthermore, the \texttt{webdoc stlog} command is useful if you want to apply different options to specific output sections. The following example illustrates some of the available options (see figure 5 for the result). Note that all options can also be specified with \texttt{webdoc do} or \texttt{webdoc init} to set the default behavior. Furthermore, you can apply \texttt{webdoc init} repeatedly within a do-file (without specifying an output document) to change the defaults between different parts of the do-file.

\begin{verbatim}
begin example5.do
webdoc init example5, replace logall ///
header(title(Example 5) width(700px) stscheme(studio) bstheme)
/***/
\end{verbatim}
Options of webdoc stlog

- **Default**: input (commands) and output
  ```
  webdoc stlog
display as txt "sqrt(2) = " /// this is a comment
  as res sqrt(2)
  webdoc stlog close
  ```

- **cmdstrip**: output without input
  ```
  webdoc stlog, cmdstrip
display as txt "sqrt(2) = " /// this is a comment
  as res sqrt(2)
  webdoc stlog close
  ```

- **nooutput**: input without output
  ```
  webdoc stlog, nooutput
display as txt "sqrt(2) = " /// this is a comment
  as res sqrt(2)
  webdoc stlog close
  ```

- **lbstrip** and **gtstrip**: remove line-break comments and continuation symbols
  ```
  webdoc stlog, lbstrip gtstrip
display as txt "sqrt(2) = " /// this is a comment
  as res sqrt(2)
  webdoc stlog close
  ```

- **cmdlog**: display code instead of results
  ```
  webdoc stlog, cmdlog
display as txt "sqrt(2) = " /// this is a comment
  as res sqrt(2)
  webdoc stlog close
  ```

- **matastrip**: remove Mata begin and end commands
  ```
  webdoc stlog, matastrip
  mata:
sqrt(2)
end
webdoc stlog close
  ```

---

end example5.do
Creating HTML or Markdown documents using webdoc

Options of webdoc stlog

- Default: input (commands) and output

```
. display as txt "sqrt(2) = " /// this is a comment
> as res sqrt(2)
sqrt(2) = 1.4142136
```

- cmdstrip: output without input

```
sqrt(2) = 1.4142136
```

- mnoout: input without output

```
. display as txt "sqrt(2) = " /// this is a comment
> as res sqrt(2)
```

- \bstrip and gstrip: remove line-break comments and continuation symbols

```
. display as txt "sqrt(2) = "
> as res sqrt(2)
sqrt(2) = 1.4142136
```

- cmdlog: display code instead of results

```
display as txt "sqrt(2) = " /// this is a comment
> as res sqrt(2)
```

- matastrip: remove Mata begin and end commands

```
: sqrt(2)
1  .414213562
```

Figure 5. Output file example5.html displayed in a browser

Note that webdoc stlog distinguishes between Stata output and Stata code. By default, webdoc stlog displays Stata output, tagged by `<pre class="stlog">`<samp>`. However, if the cmdlog option is specified, webdoc stlog displays Stata code, tagged by `<pre class="stcmd">`<code>`. The color scheme chosen in `header(stscheme())` applies only to sections of Stata output, not to code. Code is displayed using standard settings, with shaded comments.\(^1\)

### 3.5 Generating do-files from output sections

webdoc stlog has a dosave option that stores a do-file from the commands in the logged output section. This is useful if you want to provide the commands in a downloadable file. Here is a somewhat advanced example in which a Code button (with an arrow icon from http://glyphicons.com/) is placed in the upper right corner of the Stata output box (see figure 6):

\(^1\) To omit the shading of comments in code display, you can specify the plain option with webdoc stlog. To apply comment shading in output display, add the lcom option in `header(stscheme())`.\)
Exercise 1

Open the 1978 automobile data and run a regression of price on mileage using the *regress* command.

```stata
. sysuse auto
(1978 Automobile Data)

. regress price mpg

Source | SS    df  MS         Number of obs =  74
-------+-------------------------------------------------------------
Model   | 139449474   1 139449474  Prob > F      =  0.0000
Residual| 635863996  72 8883554.48  R-squared     =  0.2196
         |                     Adj R-squared =  0.2087
Total   | 635863996  73 8699525.97  Root MSE     =  2523.7

price | Coef.  Std. Err.      t    P>|t|     [95% Conf. Interval]
-------+------------------------------------------
     mpg| -238.8943      53.07669  -4.50   0.000    -344.7008   -133.0879
 cons  |  11253.85     1170.813   9.61   0.000     8919.088    13587.03
```

Figure 6. Output file *example6.html* displayed in a browser

If the user clicks on the Code button, a file containing the Stata commands opens. The *webdoc put* command is used here to write the necessary code to generate the button (an alternative would be to use *webdoc set*; see section 2.7). The Stata output box is included in a <div style="position:relative"> tag so the button can be positioned relative to the box. For information on the code generating the button, see [http://getbootstrap.com/css/#buttons]; for the code to display the arrow icon, see [http://getbootstrap.com/components/#glyphicons].
Creating HTML or Markdown documents using webdoc

`webdoc stlog close` returns the name and relative path of the do-file in `s(doname)`, from where it can be provided to `webdoc put`. By default, the do-file is placed in the same folder as the output document. Specify `dodir(path)` with `webdoc do` or `webdoc init` to request a different location. Furthermore, if you want the do-file to have a specific name, specify a name with `webdoc stlog`. For example, type

```
.webdoc stlog exercise1, dosave
```

to use the name `exercise1.do` for the do-file (the suffix will always be `.do`).

### 3.6 The nodo option

An indispensable option for larger projects is the `nodo` option. This option allows you to recompile your document without rerunning the Stata commands. `webdoc` keeps the log files from previous runs, so rerunning the Stata commands would be a waste of time if the Stata commands did not change. Therefore, once the commands in a Stata output section are set, type the following:

```
webdoc stlog, nodo
```

To apply `nodo` to all Stata output sections in the document, specify `nodo` with `webdoc init` or `webdoc do`. To turn the commands back on in a specific section, type the following:

```
webdoc stlog, do
```

Note that you can also turn commands on and off between different parts of the document by applying the `webdoc init` command with the `do` or `nodo` option repeatedly within the do-file.

Be aware that `webdoc` uses consecutive numbers to name the log files of the output sections. Thus the name for a specific section will change if other (unnamed) sections are added or deleted in preceding parts of the document. In this case, you may have to rerun all output sections.\(^2\) Hence, if a specific Stata output section contains time-consuming commands, it is always a good idea to assign a fixed name (that is, type `webdoc stlog name`).

### 3.7 Graphs

To include a graph in the output document, simply type `webdoc graph` after the graph has been created. `webdoc graph` will store the graph on disk and place an appropriate `<img>` in the output document to display the graph. By default, a `.png` image with a width of 500 pixels is produced. There are various options to change how the graph is exported and how it is integrated into the output document. The following example

\(^2\) An exception are `cmdlog` output sections (see section 3.4 above), because the log files of these sections will always be updated regardless if `nodo` is specified or not.
sets the physical width of the graph to 1,000 pixels, sets the display width to 100%, provides a caption for the graph, and also sets a tooltip title.

begin example7.do
webdoc init example7, replace logall header(title(Example 7) width(700px))
/***
<h2>Exercise 1</h2>
<p>Open the 1978 automobile data and draw a scatterplot of price against mileage using the <code>twoway</code> command and include a linear fit.</p>
***/
sysuse auto
twoway (scatter price mpg) (lfit price mpg)
webdoc graph, caption(Figure 1: Twoway plot of price by mileage) cabove ///
width(1000) title(price by mpg) attributes(width="100%")
end example7.do

Exercise 1

Open the 1978 automobile data and draw a scatterplot of price against mileage using the <code>twoway</code> command and include a linear fit.

```
. sysuse auto
(1978 Automobile Data)
. twoway (scatter price mpg) (lfit price mpg)
```

![Figure 1: Twoway plot of price by mileage](example7.html)

Figure 7. Output file example7.html displayed in a browser
Figure 7 displays the resulting file as it looks in a browser. If the user moves the pointer to the graph, a tooltip containing price by mpg will be shown. Furthermore, if the user clicks on the graph, the graph file will be opened. Note that \texttt{webdoc graph} automatically creates a name for the graph (based on the name of the relevant Stata output section). If you want your graph to have a specific name, you can type \texttt{webdoc graph name}.

If, as in the example above, the \texttt{logall} option is specified, \texttt{webdoc} will stop the Stata output section at the position of the \texttt{webdoc graph} command, insert the graph, and then continue with a new output section. If you want to display a graph that has been produced within an explicit \texttt{webdoc stlog} section, it is usually better to call \texttt{webdoc graph} after the section has been closed. That is, type the following:

\begin{verbatim}
webdoc stlog
    sysuse auto
    twoway (scatter price mpg) (lfit price mpg)
webdoc stlog close
webdoc graph
\end{verbatim}

Typing \texttt{webdoc graph} within a \texttt{webdoc stlog} section is allowed, but it will cause the graph to be included in the HTML document before the output box.

The default for \texttt{webdoc graph} is to provide the image source in the form of a link to the external graph file. However, you can also specify the \texttt{hardcode} option to directly embed the image in the HTML document (this only works if the requested graph format is \texttt{.png} or \texttt{.svg}). The \texttt{hardcode} option is useful if you want to share your HTML file without having to copy around multiple files. Another use might be if you want to embed a low-resolution graph in the HTML document and, at the same time, provide a link to an external high-resolution graph file. This could be achieved by typing the following:

\begin{verbatim}
webdoc graph, hardcode width(200) link
webdoc graph, custom width(1000)
\end{verbatim}

The first \texttt{webdoc graph} command embeds a low-resolution graph (200 pixels wide) in the HTML document and also includes a link to the external graph file. The second \texttt{webdoc graph} command overwrites the external graph file with a high-resolution variant (1,000 pixels wide) but does not include any code in the HTML document (because of \texttt{custom}). If the user clicks on the image in the browser, the high-resolution graph will be opened.

### 3.8 Tables

\texttt{webdoc} does not provide specific tools for producing tables. However, you can use other programs such as \texttt{listtex} by Newson (2001) or \texttt{esttab} by Jann (2007) to write a table in HTML format and then add the result to your HTML document using \texttt{webdoc append}. Below is an example based on \texttt{esttab} (see figure 8 for the result). The procedure for \texttt{listtex} or other commands would be similar.
begin example8.do

webdoc init example8, replace header(title(Example 8) width(700px))

/***
<h2>Explaining price</h2>
<p>Table 1 shows two regression models explaining the price of cars.</p>
***/

webdoc stlog, nolog
sysuse auto
regress price mpg weight
estimates store m1
regress price mpg weight foreign turn
estimates store m2
esttab m1 m2 using example8_tab1.html, replace label wide nomtitle
//nostar b(2) align(right) width(500) title(Table 1: A regression table)
webdoc stlog close
webdoc append example8_tab1.html

end example8.do

Explaining price

Table 1 shows two regression models explaining the price of cars.

<table>
<thead>
<tr>
<th></th>
<th>(1)</th>
<th>(2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mileage [mpg]</td>
<td>-48.51</td>
<td>-0.47</td>
</tr>
<tr>
<td>Weight [lbs.]</td>
<td>1.75</td>
<td>4.28</td>
</tr>
<tr>
<td>Car type</td>
<td></td>
<td>3221.42</td>
</tr>
<tr>
<td>Turn Circle [ft]</td>
<td>-229.21</td>
<td>-2.01</td>
</tr>
<tr>
<td>Constant</td>
<td>1946.07</td>
<td>1368.20</td>
</tr>
</tbody>
</table>

Observations 74 74

$ t $-statistics in parentheses

Figure 8. Output file example8.html displayed in a browser

3.9 TOC

To generate a clickable TOC from the headings in your HTML document, you can use the webdoc toc command. Simply include the webdoc toc command at the position in the file where you want the TOC to appear. All relevant headings from this position on will be collected to construct the TOC. By default, webdoc toc collects three levels of headings, from <h1> to <h3>. To collect, say, four levels from <h2> to <h5>, you
could type \texttt{webdoc toc 4 1}. The first number specifies the number of desired levels, and the second specifies the offset (that is, how many upper levels to skip). To add automatic section numbers to the headings and the entries in the TOC, you can specify the \texttt{numbered} option. The numbers will be tagged (as class \texttt{toc-secnum} in the TOC and as class \texttt{heading-secnum} in the headings), so they can be styled by CSS. Likewise, use CSS definitions for the \texttt{<ul>} tag to affect the look of the TOC. To prevent the definitions from being applied to other instances of \texttt{<ul>} in the document, it is a good idea to wrap the TOC in its own class or include it in a \texttt{<nav>} tag and make the definitions conditional on that. A somewhat advanced example is as follows (for the result, see figure 9):

```example9.do
webdoc init example9, replace header(title(Example 9) width(700px) bstheme)

/***
stylesheet
.toc ul { padding-left:0; list-style:none; font-weight:bold; }
.toc ul li { font-weight:normal; }
.toc-secnum, .heading-secnum { float:left; min-width:45px; }
*/
/***
<h1>The title</h1>
<p>Some leading text.</p>
<h4>Contents</h4>
<div class="toc">

webdoc toc 3 1, numbered

/***/
</div>

<h2>A first section</h2>
<p>Some text.</p>

</h2>

<h2>A second section</h2>
<p>Some text.</p>
<h3>A first subsection to the second section</h3>
<p>Some text.</p>
<h3>A second subsection to the second section</h3>
<p>Some text.</p>
<h4>A first subsection to the second subsection of the second section</h4>
<p>Some text.</p>

</h2>

<h2>A final section</h2>
<p>Some text.</p>

```
3.10 Dynamic text

If you want to add results from a Stata output section to the text body, one approach is to store the results as local macros and then insert the contents of these locals at appropriate places in the text body using `webdoc put` or `webdoc write`. However, these locals will no longer be available in later runs once the `nodo` option is applied. A solution to this problem is the `webdoc local` command, which can be applied within or after a Stata output section. The command can be used just like Stata’s regular `local` command, but it maintains a backup of the locals on disk and restores them if
Creating HTML or Markdown documents using webdoc

needed. Furthermore, the local macros defined by `webdoc local` will be expanded in subsequent `/*** ****/` blocks (up until the next `webdoc stlog` command, which causes the macro library to be reset). An example is as follows (see figure 10 for the compiled result):

begin example10.do
webdoc init example10, replace
webdoc stlog
    sysuse auto, clear
    regress price weight
webdoc stlog close
webdoc local b = strofreal(_b[weight], "%9.3f")
webdoc local se = strofreal(_se[weight], "%9.3f")
/***
<p> As can be seen in the output above, the estimate for the effect of weight on price is equal to 'b' (with a standard error of 'se').</p>
***/
end example10.do

Figure 10. Output file `example10.html` displayed in a browser

Alternatively, you may use `webdoc write` or `webdoc put` to write the locals to the output document. That is, you could also type the following:

webdoc put <p> As can be seen in the output above, the estimate for the effect of weight on price is equal to 'b' (with a standard error of 'se').</p>
There is a slight difference between the two approaches: expansion in /*** ***/ blocks is based on the locals as stored on disk; webdoc write and webdoc put use the current values of the locals.

4 Limitations

In general, you can work on a do-file containing webdoc commands in the same way as you would work on another do-file. For example, if you submit the do-file to Stata without applying webdoc do, Stata will process the do-file like any other do-file; the /*** ***/ blocks containing HTML code will be ignored, and the webdoc commands will do nothing. However, there are some limitations and technical issues that should be kept in mind when working with webdoc:

- The $ character is used for global macro expansion in Stata. If you use webdoc write or webdoc put to write text containing $, type \$ instead of $.

- webdoc do provides only limited support for the semicolon command delimiter (see [P] #delimit). For example, do not use semicolons to delimit webdoc commands. However, the semicolon command delimiter should work as expected if it is turned on and off between /*** ***/ blocks and between webdoc commands.

- webdoc commands should always start on a new line with webdoc being the first (noncomment) word on the line. For example, do not type . quietly webdoc ...

  or something similar.

- webdoc stlog cannot be nested. Furthermore, do not use webdoc do or webdoc init within a webdoc stlog section.

- When processing a do-file, webdoc do does not parse the contents of a do-file that is called from the main do-file using the do command (see [R] do). Thus, for example, /*** ***/ blocks in such a file will be ignored. Use webdoc do instead of do to include such a do-file.

- webdoc tries to create missing subdirectories using Mata’s mkdir() function; see [M-5] chdir(). Usually, this works only if all intermediate directories leading to the target subdirectory already exist. If mkdir() fails, you will need to create the required directories manually prior to running webdoc.

5 References

Creating HTML or Markdown documents using webdoc


——. 2014b. weaver: Stata module to produce dynamic reports in HTML, \LaTeX{}, and PDF. Statistical Software Components S457878, Department of Economics, Boston College. https://ideas.repec.org/c/boc/bocode/s457878.html.


About the author

Ben Jann is a professor of sociology at the University of Bern, Switzerland. His research interests include social science methodology, statistics, social stratification, and labor market sociology. Recent publications include articles in *Sociological Methodology, Sociological Methods and Research*, the *Stata Journal, Public Opinion Quarterly*, and the *American Sociological Review*. 
Fitting endogenous stochastic frontier models in Stata

Mustafa U. Karakaplan
Georgetown University
Department of Economics
Washington, DC
mukarakaplan@yahoo.com

Abstract. In this article, I introduce sfkk, a new command for fitting endogenous stochastic frontier models. sfkk provides estimators for the parameters of a linear model with a disturbance assumed to be a mixture of two components: a measure of inefficiency that is strictly nonnegative and a two-sided error term from a symmetric distribution. sfkk can handle endogenous variables in the frontier or the inefficiency, and the sfkk estimates outperform the standard frontier estimates that ignore endogeneity.

Keywords: st0466, sfkk, endogeneity, endogenous stochastic frontier models, production frontier, cost frontier, endogenous inefficiency

1 Introduction

Stochastic frontier models constitute a popular subfield of econometrics. They were introduced by Aigner, Lovell, and Schmidt (1977) and Meeusen and van den Broeck (1977) and further developed by many other researchers. Kumbhakar and Lovell (2000) provide an extended review of stochastic frontier models, and the literature has many empirical examples from various fields such as agriculture, aviation, banking, education, energy, and health.

Standard estimators of the stochastic frontier models estimate the parameters of a linear model with a disturbance composed of two components: a measure of inefficiency that is strictly nonnegative and a two-sided error term with a symmetric distribution. The frontier command provides estimators and options to fit these models. However, these standard estimators do not handle endogeneity in the model, which would exist if the determinants of the frontier or inefficiency are correlated to the two-sided error term.

Stata researchers provide a few commands that solve similar econometric issues. For example, Petrin, Poi, and Levinsohn (2004) introduced the levpet command, which estimates production functions with intermediate inputs as proxies to control for unobservable productivity shocks using the methodology of Levinsohn and Petrin (2003). Yasar, Raciborski, and Poi (2008) introduced the oreg command, which estimates production functions with selection bias or simultaneity by implementing the three-stage algorithm of Olley and Pakes (1996). While these methodologies and commands are useful for analyzing certain economic scenarios, Mutter et al. (2013) emphasize that a
2 The estimator

Karakaplan and Kutlu (2013) consider a stochastic frontier model with endogenous explanatory variables in the frontier and inefficiency functions and present the following estimator, which outperforms standard estimators that ignore the endogeneity in the model,

\[
\ln L(\theta) = \ln L_{y|x}(\theta) + \ln L_{x}(\theta)
\]

\[
\ln L_{y|x}(\theta) = \sum_{i=1}^{n} \left\{ \ln 2 - \frac{1}{2} \ln \sigma_i^2 + \ln \phi \left( \frac{\varepsilon_i}{\sigma_i} \right) + \ln \Phi \left( -\frac{s\lambda_i\varepsilon_i}{\sigma_i} \right) \right\}
\]

\[
\ln L_{x}(\theta) = \sum_{i=1}^{n} \left\{ \ln \left( 2 - \ln \frac{\epsilon_i}{\Omega_i} \right) - \frac{\epsilon_i^2}{\Omega_i} \right\}
\]

where \( \theta = (\beta', \eta', \varphi', \delta')' \) is the vector of coefficients; \( y = (y_1, y_2, \ldots, y_n)' \) is the vector of dependent variables; \( x = (x_1', x_2', \ldots, x_n')' \) is the matrix of endogenous variables in the model; \( \phi \) and \( \Phi \) denote the standard normal probability density function and the cumulative distribution function, respectively; \( s = 1 \) (or \( s = -1 \) for production functions); \( y_i = x_i' \beta + v_i + s u_i - x_i' \beta + (\sigma_{wi}/\sigma_{cw}) \eta'(x_i - Z_i \delta) + w_i - sw_i \) is the logarithm of expenditure (or output for production functions) of the ith producer; \( x_{1i} \) is a vector of exogenous and endogenous variables; \( x_i = Z_i \delta + \epsilon_i \) is a \( p \times 1 \) vector of all endogenous variables (excluding \( y_i \)); \( Z_i = I_p \otimes z_i' \) and \( z_i \) is a \( q \times 1 \) vector of all exogenous variables; \( v_i \) and \( \epsilon_i \) are two-sided error terms; \( u_i = \sigma_u(x_{2i}; \varphi_u)u_{i}^{*} \geq 0 \) is the one-sided error term capturing the inefficiency; \( x_{2i} \) is a vector of exogenous and endogenous variables; \( u_{i}^{*} \sim N^+(0, 1) \) is a producer-specific random component; \( \sigma_{uw}^{2} = \exp(x_{3i}'\varphi_u) \); \( w_{i} = \sigma_{wi} \sqrt{1 - \rho^2} \eta_i \); \( \sigma_{wi} = \sigma_{cw} \sigma_{uw} (.; \varphi_w) \); \( \sigma_{uw}^{2} = \exp(x_{3i}'\varphi_u) \); \( \sigma_{cw} > 0 \) is a function of the constant term; \( \sigma_{cw}^{2} = \exp(\varphi_{cw}) \), where \( \varphi_{cw} \) is the coefficient of the constant.
term for $x_{3i}^{\prime} \varphi w_i \sim N(0, 1)$; $x_{3i}$ is a vector of exogenous and endogenous variables that can share the same variables with $x_{1i}$ and $x_{2i}$; $\Omega$ is the variance–covariance matrix of $\epsilon_i$; $\sigma_{vi}^2$ is the variance of $v_i$; and $\rho$ is the vector representing the correlation between $\tilde{\epsilon}_i$ and $v_i$. The details about the assumptions and how the estimator is derived are presented in Karakaplan and Kutlu (2013).

Moreover, Karakaplan and Kutlu (2013) provide the following formula to predict the efficiency, $\text{EFF}_i = \exp(-u_i)$:

$$
E \{ \exp(-su_i) | e_i \}^s = \left\{ \frac{1 - \Phi(s \sigma_{\epsilon_i}^2 / \sigma_{\epsilon_i}^2)}{1 - \Phi(-\mu_{\epsilon_i}^* / \sigma_{\epsilon_i}^2)} \exp \left( -s \mu_{\epsilon_i}^* + \frac{1}{2} \sigma_{\epsilon_i}^2 \right) \right\}^s
$$

Finally, a test for endogeneity is proposed by Karakaplan and Kutlu (2013). In this test, the joint significance of the components of the $\eta$ term is checked. If the joint significance of the components is rejected, then correction for endogeneity is not necessary, and the model can be fit by traditional frontier models. However, if the components of the $\eta$ term are jointly significant, then there is endogeneity in the model, and a correction through (1) would be necessary.

3 The sfkk command

Gould, Pitblado, and Poi (2010) provide an excellent guideline for researchers who need to compute maximum likelihood estimators that are not available as prepackaged routines. Following their suggestions and using Stata’s powerful ml tools, I programmed the sfkk command, which can estimate (1) and (1). The sfkk package includes three files: sfkk.ado, sfkk.ml.ado, and sfkk.sthlp. sfkk.ado provides the main estimation syntax that users access through running sfkk. sfkk.ml.ado includes the evaluator subroutines that sfkk calls for the actual estimation of the parameters. The default subroutine in sfkk.ml.ado is a method-d0 evaluator that calculates the overall log likelihood. sfkk.ml.ado has another subroutine that is a method-lf0 evaluator called by sfkk if the fast(#) option is specified. The method-lf0 evaluator speeds up the regression because it does not compute any derivatives. The fast(#) option combines this evaluator with a tolerance-based methodology to complete the estimation faster. The postestimation routines, such as predicting the efficiency, testing the endogeneity, and documenting the results, are handled in sfkk.ado. Finally, sfkk.sthlp is the sfkk command’s help file, which provides an extended version of the subsections below with further details, such as option abbreviations, stored results, and examples with clickable features.
Fitting endogenous stochastic frontier models in Stata

3.1 Syntax

Estimation syntax

sfkk depvar [ indepvars ] [ if ] [ in ] [ weight ] [, noconstant production cost endogenous( endovarlist ) instruments( ivarlist ) exogenous( exovarlist ) leaveout( lovarlist ) uhet( uvarlist [, noconstant ] ) whet( wvarlist ) initial( matname ) delve fast( # ) difficult technique( algorithm_spec ) iterate( # ) mlmodel( model_options ) mlmax( maximize_options ) header timer beep compare efficiency( effvar [, replace ] ) test nicely mldisplay( display_options ) ]

Version syntax

sfkk, version

Replay syntax

sfkk [, level( # )]

3.2 Options for the estimation syntax

Frontier

noconstant suppresses the constant term (intercept) in the frontier.

production specifies that the model to be fit is a production frontier model. The default is production.

cost specifies that the model to be fit is a cost frontier model. The default is production.

Equations

endogenous( endovarlist ) specifies that the variables in endovarlist be treated as endogenous. By default, sfkk assumes the model is exogenous.

instruments( ivarlist ) specifies that the variables in ivarlist be used as instrumental variables to handle endogeneity. By default, sfkk assumes the model is exogenous.

exogenous( exovarlist ) specifies that exovarlist is the complete list of included exogenous variables. The default for the complete list of included exogenous variables is indepvars + uvarlist + wvarlist. Depending on the model, exovarlist can be different from indepvars + uvarlist + wvarlist. For an illustration, please see the sfkk help file. exogenous() cannot be used with leaveout(). The exogenous() option is seldom used and can safely be omitted.
leaveout(lovarlist) specifies that the variables in lovarlist be removed from the default list of included exogenous variables, which is indepvars + wvarlist + uvarlist. Depending on the model, some variables, such as functions of some included exogenous variables, can be left out of the complete list of included exogenous variables. For an illustration, please see the sfkk help file. leaveout() cannot be used with exogenous(). The leaveout() option is seldom used and can safely be omitted.

uhet(uvarlist[, noconstant]) specifies the inefficiency component be heteroskedastic, with the variance function depending on a linear combination of uvarlist. Specifying noconstant suppresses the constant term from the variance function.

whet(wvarlist) specifies that the idiosyncratic error component be heteroskedastic, with the variance function depending on a linear combination of wvarlist.

Regression

initial(matname) specifies that matname is the initial value matrix.

delve provides a regression-based methodology to search for better initial values. The default is to use ml search. delve is often successful in finding better initial values. Using delve is recommended.

fast(#) provides a tolerance-based methodology to complete the regression faster. # can be specified to take any value larger than 0. The regression completes faster with larger values of #, but larger values of # result in less accurate findings. Experimenting with various values of # is suggested because different values of # work better with different models. Using fast() is recommended to explore the direction of the maximization problem faster. However, to improve the accuracy of the findings, one should avoid using fast() once the model is decided and specification is finalized.

difficult specifies that the likelihood function is likely to be difficult to maximize because of nonconcave regions. When the message “not concave” appears repeatedly, ml’s standard stepping algorithm may not be working well. difficult specifies that a different stepping algorithm be used in nonconcave regions. There is no guarantee that difficult will work better than the default; sometimes it is better and sometimes it is worse. The difficult option should be used only when the default stepper declares convergence and the last iteration is “not concave” or when the default stepper is repeatedly issuing “not concave” messages and producing only tiny improvements in the log likelihood.

 technique(algorithm_spec) specifies how the likelihood function is to be maximized. The following algorithms are allowed. For details, see Gould, Pitblado, and Poi (2010).

 technique(nr) specifies Stata’s modified Newton–Raphson algorithm.

 technique(bhhh) specifies the Berndt–Hall–Hall–Hausman algorithm, which is allowed only with the fast() option.
Fitting endogenous stochastic frontier models in Stata

technique(dfp) specifies the Davidon–Fletcher–Powell algorithm.

technique(bfgs) specifies the Broyden–Fletcher–Goldfarb–Shanno (BFGS) algorithm. The default is technique(bfgs).

Switching between algorithms is possible by specifying more than one algorithm in the technique() option. By default, an algorithm is used for five iterations before switching to the next algorithm. To specify a different number of iterations, include the number after the technique in the option. For example, specifying technique(bfgs 10 nr 1000) requests that sfkk perform 10 iterations with the BFGS algorithm, followed by 1,000 iterations with the Newton–Raphson algorithm, followed by 10 more iterations with the BFGS algorithm, and so on. The process continues until the convergence or maximum number of iterations is reached.

iterate(#) specifies the maximum number of iterations. When the number of iterations equals #, the optimizer stops and presents the current results. If the convergence gets declared before this threshold is reached, the optimizer stops and presents the optimized results. The default is iterate(16000), which is the current value of maxiter.

mlmodel(model_options) controls the ml model options; it is seldom used.

mlmax(maximize_options) controls the ml max options; it is seldom used.

Reporting

header displays a summary of the model constraints in the beginning of the regression.

header provides a way to check the model specifications quickly while the estimation is running or provides a guide to distinguish different regression results that are kept in a single log file.

timer displays the total elapsed time sfkk took to complete. The total elapsed time is measured from the moment the command is entered to the moment the reporting of all findings is completed.

beep produces a beep when sfkk reports all findings. beep is useful for multitasking.

compare fits the specified model with the exogeneity assumption and displays the regression results after displaying the endogenous model regression results.

efficiency(effvar[, replace]) generates the production or cost efficiency variable effvar_EN once the estimation is completed and displays its summary statistics in detail. The option automatically extends any specified variable name effvar with _EN. If the compare option is specified, efficiency() also generates effvar_EX, the production or cost efficiency variable of the exogenous model, and displays its summary statistics. Specifying replace replaces the contents of the existing effvar_EN and effvar_EX with the new efficiency values from the current model.

test provides a method to test the endogeneity in the model. It tests the joint significance of the components of the eta term and reports the findings after displaying
the regression results. For more information about test, see Karakaplan and Kutlu (2013).

nicely displays the regression results in a single table. nicely requires estout, a user-written command by Jann (2005), to format some parts of the table, and the sfkk table style resembles that of Karakaplan and Kutlu (2013). The nicely option checks whether the estout package is installed on Stata, and if not, the nicely option installs the package. If the compare option is specified, nicely displays the exogenous and endogenous models with their corresponding equations and statistics side by side in a single table for easy comparison. nicely estimates the production or cost efficiency and tests endogeneity and reports them in the table even if the efficiency(effvar) or test option is not specified.

mldisplay(display_options) controls the ml display options; it is seldom used.

3.3 Options for the version and replay syntax

version displays the version of sfkk installed on Stata and the program author information. This option can be used only in the version syntax.

level(#) specifies the confidence level, as a percentage, for confidence intervals. The default is level(95) or as set by set level. This option can be used in the replay syntax or in mldisplay(display_options).

4 Examples

In this section, I illustrate sfkk in three different examples. The first two examples analyze randomly generated datasets in a cost setting and in a production setting. These two datasets are for illustrative purposes, and the results do not represent a specific industry. The last example, however, examines a stochastic cost frontier model with a real dataset that come from the U.S. K–12 education sector. Eta endogeneity test results indicate that all models in the examples suffer from endogeneity problems. Correcting the endogeneity through sfkk results in substantially different coefficient estimates and efficiency scores.

4.1 Endogenous stochastic cost frontier example

The first example uses a cross-sectional dataset with 750 observations and fits a cost model in which one of the frontier variables (z1) and a variable determining cost inefficiency (z2) are endogenous. Two instrumental variables are used (iv1 and iv2) to handle the endogeneity. The header option summarizes the model specification.
Fitting endogenous stochastic frontier models in Stata

```
. use http://www.mukarakaplan.com/files/sfkkcost.dta
. sfkk y x1 x2 x3 z1, cost u(z2) en(z1 z2) i(iv1 iv2) header delve compare 
> nicely timer

17 Nov 2016 15:22:46
ENDOGENOUS STOCHASTIC COST FRONTIER MODEL (Model EN)
Dependent Variable: y
Frontier Variable(s): Constant x1 x2 x3 z1
U Variable(s): Constant z2
W Variable(s): Constant
Endogenous Variable(s): z1 z2
Excluded Instrument(s): iv1 iv2
Exogenous Variable(s): iv1 iv2 x1 x2 x3

Delving into the problem...
initial: log likelihood = -1286.915
rescale: log likelihood = -1286.915
rescale eq: log likelihood = -121.43367
Iteration 0: log likelihood = -121.43367
Iteration 1: log likelihood = -63.571795 (backed up)
Iteration 2: log likelihood = -60.637741 (backed up)
Iteration 3: log likelihood = -16.951509 (backed up)
Iteration 4: log likelihood = 15.209068 (backed up)
Iteration 5: log likelihood = 742.89103 (backed up)
(output omitted)
Iteration 52: log likelihood = 1604.6298
Iteration 53: log likelihood = 1604.6298

Analyzing the exogenous comparison model (Model EX)...
initial: log likelihood = -1143.5903
alternative: log likelihood = -633.5094
rescale: log likelihood = -633.5094
rescale eq: log likelihood = -358.6175
initial: log likelihood = -358.6175
rescale: log likelihood = -358.6175
rescale eq: log likelihood = 21.18479
Iteration 0: log likelihood = 21.18479
Iteration 1: log likelihood = 40.199285 (backed up)
(output omitted)
Iteration 24: log likelihood = 886.56289

Table: Estimation Results

<table>
<thead>
<tr>
<th></th>
<th>Model EX</th>
<th>Model EN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dep.var: y</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Constant</td>
<td>0.529*** (0.030)</td>
<td>0.225** (0.081)</td>
</tr>
<tr>
<td>x1</td>
<td>-0.084*** (0.016)</td>
<td>-0.042* (0.020)</td>
</tr>
<tr>
<td>x2</td>
<td>0.067** (0.025)</td>
<td>0.112*** (0.032)</td>
</tr>
<tr>
<td>x3</td>
<td>0.058* (0.029)</td>
<td>0.354*** (0.071)</td>
</tr>
<tr>
<td>z1</td>
<td>0.010 (0.021)</td>
<td>0.424*** (0.087)</td>
</tr>
</tbody>
</table>
```
### 4.2 Endogenous stochastic production frontier example

The second example uses a cross-sectional dataset with 500 observations and fits a production model in which one of the frontier variables ($z_1$) and a variable determining production inefficiency ($z_2$) are endogenous. Two instrumental variables are used ($iv_1$ and $iv_2$) to handle the endogeneity. Notice that in this example, the `compare` and `nicely` options are not specified; instead, the `efficiency()` and `test` options are specified. So the results are presented in raw format with prediction equations but no exogenous comparison model.
Fitting endogenous stochastic frontier models in Stata

```
. use http://www.mukarakaplan.com/files/sfkkprod.dta, clear
. sfkk y x1 x2 z1, prod u(z2) en(z1 z2) i(iv1 iv2) delve header eff(pef) test

17 Nov 2016 15:23:43

ENDOGENOUS STOCHASTIC PRODUCTION FRONTIER MODEL (Model EN)

Dependent Variable: y
Frontier Variable(s): Constant x1 x2 z1
U Variable(s): Constant z2
W Variable(s): Constant
Endogenous Variable(s): z1 z2
Excluded Instrument(s): iv1 iv2
Exogenous Variable(s): iv1 iv2 x1 x2

Delving into the problem...

initial: log likelihood = -915.26945
rescale: log likelihood = -674.50035
rescale eq: log likelihood = -199.47821
Iteration 0: log likelihood = -199.47821
Iteration 1: log likelihood = -190.32271 (backed up)
Iteration 2: log likelihood = -183.28355 (backed up)
Iteration 3: log likelihood = -164.08851 (backed up)
Iteration 4: log likelihood = -164.08851 (backed up)
Iteration 5: log likelihood = -14.005305 (backed up)
Iteration 6: log likelihood = 233.97196 (backed up)
(output omitted)
Iteration 47: log likelihood = 713.98035
Iteration 48: log likelihood = 713.98036

Endogenous stochastic prod frontier model with normal/half-normal specification

Number of obs = 500
Wald chi2(3) = 129.50
Log likelihood = 713.98036 Prob > chi2 = 0.0000

Coef. Std. Err. z P>|z| [95% Conf. Interval]

frontier_y
  x1 .1860659 .0314463 5.92 0.000 .1244324 .2476995
  x2 .1322843 .0326762 4.05 0.000 .0682402 .1963284
  z1 -.7470345 .1112763 -6.71 0.000 -.965132 -.528937
  _cons .6314415 .0320355 19.71 0.000 .5686531 .69423

ivr1_z1
  iv1 .6650172 .0891591 7.46 0.000 .4902686 .8397658
  iv2 .1354525 .0392074 3.45 0.001 .0586074 .2122976
  x1 -.0764425 .0370097 -2.07 0.039 -.1489801 -.0039049
  x2 .2075451 .0390051 5.32 0.000 .1310965 .2839937
  _cons -.0069407 .0351824 -0.20 0.844 -.0758969 .0620155

eta1_z1
  _cons .4570567 .1139459 4.01 0.000 .2337257 .6803866

```
In this example, the eta endogeneity test result rejects the null hypothesis at the 0.1% level, which means that a correction for endogeneity in the model is needed. The coefficient of \( z_1 \) in the frontier is negative and significant. Moreover, the coefficient of \( z_2 \) in the inefficiency term is positive and significant. If the \texttt{compare} option was specified, the results from an exogenous comparison model would show that the coefficient of \( z_2 \) is negative and larger in absolute terms if its endogeneity is not handled. This conclusion would also be reflected in production efficiency estimates. The mean production efficiency is 0.915 in Model EN, whereas the same statistic is 0.982 in Model EX, which is not displayed here. So producers are not as efficient in production as they would appear in a standard frontier model that ignores endogeneity. The \texttt{efficiency()} option saves the efficiency scores from Model EN as a variable, and when the \texttt{compare} option is specified, \texttt{efficiency()} would also save the efficiency scores from Model EX as a variable. Having these two variables would enable a graphical comparison of the models as shown below.
Fitting endogenous stochastic frontier models in Stata

```stata
. capture sfkk y x1 x2 z1, prod u(z2) en(z1 z2) i(iv1 iv2) delve
efficiency(pef, replace) compare
. histogram pef_EX if pef_EX>0.8, w(0.01) freq xtitle("Production efficiency")
  xtick(0.8(0.05)1) xlabel(0.8(0.05)1) ytick(0(25)300) ylabel(0(50)300)
  ytitle("Number of producers") color(gs8) lcolor(gs4) title("Model EX")
  graphregion(color(gs14))
  (bin=19, start=.81446136, width=.01)
```

Figure 1. Graphical representation—Model EX
4.3 Example from the U.S. K–12 education sector

For this example, I use the main data from the National Center for Education Statistics and New York State Education Department. The cross-sectional dataset consists of 635 traditional public school districts in the 2011–2012 school year. Following the models in Gronberg et al. (2015) and Karakaplan and Kutlu (2015), I set the dependent variable as the natural logarithm of actual current operating expenditures per pupil \( (\text{expend}) \). Cost frontier variables include district enrollment \( (\text{enroll}) \) and the square of district enrollment \( (\text{enroll}^2) \) as the output quantity variables, an index of district-level academic performance \( (\text{scores}) \) as the output quality variable, an index of input prices \( (\text{prices}) \) based on the derived prices of instructional material and wages of education personnel,\(^1\) and an index of district-level student characteristics \( (\text{body}) \) that measures the effects of environmental factors such as the percentage of special education students. Cost inefficiency is modeled with a Herfindahl–Hirschman index \( (\text{hhi}) \) of education market concentration ranging between zero and one, with one indicating a monopoly setting. I control for the endogeneity of \( \text{scores} \) and \( \text{hhi} \) by using the number of small

---

\(^1\) The comparable wage index was originally produced by Taylor and Fowler (2006) and is regularly updated at http://bush.tamu.edu/research/faculty/Taylor_CWI/.
Fitting endogenous stochastic frontier models in Stata

streams in a county (streams) and the unemployment rate in a county (unemp) as the instrumental variables.  

In this example, I specify the compare, nicely, and header options in the command line. Model EX represents the model that does not control for the endogeneity in the model (comparable with a standard frontier command estimation), and Model EN represents the model that handles the endogeneity. The coefficient associated with scores is expected to be positive and increasing costs. The coefficient associated with hhi is also expected to be positive and increasing cost inefficiency, because authorities in more concentrated markets may be less careful about how they spend their resources. These two coefficients are expected to be downward biased in Model EX. Looking at the results in the output, we see that individual eta terms of hhi and scores are both significant at the 0.1% level and that the eta endogeneity test result shows that correction for endogeneity is needed. As illustrated in the table, the coefficient of scores is positive and significant, and the coefficient of hhi is positive but not significant in Model EX. In Model EN, these two coefficients are substantially larger and significant. Because there are differences in the magnitudes and significance of the coefficients in Model EX and Model EN, controlling for the endogeneity in the model is important. The differences in the mean and median cost efficiencies of Model EX and Model EN indicate this importance as well.

. use http://www.mukarakaplan.com/files/sfkkedu.dta, clear
. sfkk expend enroll enroll2 scores prices body, cost uhet(hhi)
> endogenous(scores hhi) instruments(streams unemp) delve compare nicely
> header beep timer technique(dfp 25 bfgs 25)
17 Nov 2016 15:24:32
ENDOGENOUS STOCHASTIC COST FRONTIER MODEL (Model EN)
Dependent Variable: expend
Frontier Variable(s): Constant enroll enroll2 scores prices body
U Variable(s): Constant hhi
W Variable(s): Constant
Endogenous Variable(s): scores hhi
Excluded Instrument(s): streams unemp
Exogenous Variable(s): streams unemp enroll enroll2 prices body

Delving into the problem...
initial: log likelihood = -29952.478
rescale: log likelihood = -23152.856
rescale eq: log likelihood = -1527.7189
(setting technique to dfp)
Iteration 0: log likelihood = -1527.7189
Iteration 1: log likelihood = -978.04977 (backed up)
Iteration 2: log likelihood = -956.73029 (backed up)
(output omitted)
Iteration 96: log likelihood = 949.71336
Iteration 97: log likelihood = 949.71337

2. The topographical data comes from the U.S. Geological Survey Geographic Names Information System. The unemployment rates data come from New York State’s Department of Labor.
Analyzing the exogenous comparison model (Model EX)...

initial: log likelihood = -12515.72
alternative: log likelihood = -6631.5553
rescale: log likelihood = -1606.6784
rescale eq: log likelihood = -1351.0361
initial: log likelihood = -1351.0361
rescale: log likelihood = -1351.0361
rescale eq: log likelihood = -1351.0361
Iteration 0: log likelihood = -1351.0361
Iteration 1: log likelihood = -1329.4522 (backed up)
(output omitted)
Iteration 28: log likelihood = 296.36882

Table: Estimation Results

<table>
<thead>
<tr>
<th></th>
<th>Model EX</th>
<th>Model EN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dep.var: expend</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Constant</td>
<td>10.880*** (0.305)</td>
<td>11.531*** (0.522)</td>
</tr>
<tr>
<td>enroll</td>
<td>-0.531*** (0.080)</td>
<td>-0.823*** (0.156)</td>
</tr>
<tr>
<td>enroll2</td>
<td>0.031*** (0.005)</td>
<td>0.048*** (0.010)</td>
</tr>
<tr>
<td>scores</td>
<td>0.031*** (0.046)</td>
<td>0.048*** (0.347)</td>
</tr>
<tr>
<td>prices</td>
<td>0.656*** (0.042)</td>
<td>0.304** (0.114)</td>
</tr>
<tr>
<td>body</td>
<td>3.432*** (0.294)</td>
<td>6.089*** (0.795)</td>
</tr>
<tr>
<td>Dep.var: ln(σ²_u)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Constant</td>
<td>-3.685*** (0.288)</td>
<td>-6.939*** (1.073)</td>
</tr>
<tr>
<td>hhi</td>
<td>0.373 (0.434)</td>
<td>5.692*** (1.349)</td>
</tr>
<tr>
<td>Dep.var: ln(σ²_v)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Constant</td>
<td>-4.324*** (0.167)</td>
<td></td>
</tr>
<tr>
<td>Dep.var: ln(σ²_w)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Constant</td>
<td>-3.989*** (0.084)</td>
<td></td>
</tr>
<tr>
<td>eta1 (scores)</td>
<td>-1.410*** (0.349)</td>
<td></td>
</tr>
<tr>
<td>eta2 (hhi)</td>
<td>-0.263*** (0.060)</td>
<td></td>
</tr>
<tr>
<td>eta Endogeneity Test</td>
<td>X²=34.64 p=0.000</td>
<td></td>
</tr>
</tbody>
</table>

Observations 635 635
Log Likelihood 296.37 949.71
Mean Cost Efficiency 0.8756 0.9394
Median Cost Efficiency 0.8869 0.9574

Notes: Standard errors are in parentheses. Asterisks indicate significance at the 0.1% (***), 1% (**) and 5% (*) levels.
(output omitted)
Completed in 0 hour(s), 0 minute(s) and 47 second(s).

5 Conclusion

In this article, I follow the recent advances in the estimation of endogenous stochastic frontier models presented by Karakaplan and Kutlu (2013) and offer sfkk, a new
command to estimate such models in Stata. sfkk can handle endogenous variables in the frontier or the inefficiency, and examples show that sfkk estimates outperform the standard frontier estimates that ignore endogeneity. sfkk provides many options that can become handy for researchers from different fields such as agriculture, aviation, banking, education, energy, and health.

6 Acknowledgments

I thank Levent Kutlu, Isabel Canette, Ben Jann, and Kit Baum for their amazing support.

7 References


**About the author**

Mustafa U. Karakaplan has a PhD in Economics from Texas A&M University.
Commands for testing conditional moment inequalities and equalities

Donald W. K. Andrews
Yale University
New Haven, CT
donald.andrews@yale.edu

Wooyoung Kim
University of Wisconsin–Madison
Madison, WI
wkim68@wisc.edu

Xiaoxia Shi
University of Wisconsin–Madison
Madison, WI
xshi@ssc.wisc.edu

Abstract. In this article, we present two commands (\texttt{cmi\_test} and \texttt{cmi\_interval}) to implement the testing and inference methods for conditional moment inequality or equality models proposed in Andrews and Shi (2013, \textit{Econometrica} 81: 609–666). The \texttt{cmi\_test} command tests the validity of a finite number of conditional moment equalities or inequalities. This test returns the value of the test statistic, the critical values at significance levels 1%, 5%, and 10%, and the \textit{p}-value. The \texttt{cmi\_interval} command returns the confidence interval for a one-dimensional parameter defined by intersection bounds. We obtain this confidence interval by inverting \texttt{cmi\_test}. All procedures implemented are uniformly asymptotically valid under appropriate conditions (specified in Andrews and Shi [2013]).

Keywords: st0467, \texttt{cmi\_test}, \texttt{cmi\_interval}, conditional moment inequalities and equalities, confidence interval, uniformly asymptotically valid test

1 Introduction

This article provides a brief introduction to conditional moment inequality or equality testing and describes the new \texttt{cmi\_test} and \texttt{cmi\_interval} commands. The \texttt{cmi\_test} command implements the testing procedure proposed in Andrews and Shi (2013) for general moment inequality models, with a finite number of conditional moment restrictions and a finite-dimensional parameter. The \texttt{cmi\_interval} command returns confidence intervals for a one-dimensional parameter bounded above or below by a finite number of conditional moments by inverting the testing procedure proposed in Andrews and Shi (2013).

The package we describe is not intended for computing confidence intervals for $\theta$, unless the setting is the one associated with \texttt{cmi\_interval}. Computing confidence intervals in a general setting requires numerically sketching out the set of $\theta$ values for which \texttt{cmi\_test} returns an acceptance. Simple grid-search algorithms for this task become exponentially more costly as the dimension of $\theta$ increases. Other commonly used statistical software packages offer more efficient algorithms, but we are not aware of their implementation in Stata.

© 2017 StataCorp LLC

st0467
Null hypotheses arise frequently in econometrics in the form of conditional moment inequalities or equalities, for example, when testing the sign of the conditional average treatment effect, when certain incomplete models lead to conditional moment inequality or equality restrictions on parameters, and when testing the fundamental assumptions for the local average treatment-effect estimator (see Mourifie and Wan [2014]). Andrews and Shi (2013) first transform the conditional moment inequalities or equalities into a large number of unconditional moment inequalities or equalities, then construct a test statistic based on these unconditional moment inequalities or equalities. The resulting test is uniformly asymptotically valid and consistent against all fixed alternatives. Chernozhukov, Lee, and Rosen (2013) and Lee, Song, and Whang (2013) propose two main alternatives to the Andrews and Shi (2013) test, both based on nonparametric estimators of the conditional moment inequalities or equalities. All three tests are consistent and do not dominate one another in terms of power. In practice, one may choose a test based on computational feasibility or implement more than one test for more robust conclusions.

The commands we describe offer a rich set of options to allow the user to fine tune the procedure. However, in most applications, the default options—as recommended in Andrews and Shi (2013)—work well without much user input.

We use the following notation throughout this article: \( \lfloor a \rfloor \) denotes the largest integer less than or equal to \( a \), and \( \lceil a \rceil \) denotes the smallest integer larger than or equal to \( a \).

\section{Framework}

\subsection{Parameter inference based on conditional moment inequalities and equalities}

Consider an independent and identically distributed sample \( \{W_i\}_{i=1}^n \). Let \( X_i \) be a vector of instrumental variables, which is a subvector of \( W_i \). Conditional moment inequality and equality models are

\begin{align}
E\{m_j(W_i, \theta_0)|X_i\} &\geq 0 \text{ for } j = 1, \ldots, p \\
E\{m_j(W_i, \theta_0)|X_i\} &= 0 \text{ for } j = p + 1, \ldots, k, \text{ almost surely}
\end{align}

(1)

where \( p \) and \( k \) are two nonnegative integers so that \( k \geq p \) and \( m(\cdot, \theta_0) := \{m_1(\cdot, \theta_0), \ldots, m_k(\cdot, \theta_0)\}' \) is a vector of moment functions of the observables that are known up to the parameter \( \theta_0 \). The set \( \Theta \subseteq R^{d_\theta} \) denotes the parameter space for \( \theta_0 \). The moment functions need not depend on some elements of \( W_i \), which makes those elements excluded variables. The conditional moment inequality model arises in many modeling contexts. We give an example later, and more examples are in Andrews and Shi (2013).

In a conditional moment inequality model, the parameter \( \theta_0 \) may or may not be point identified. Thus a consistent point estimator for \( \theta_0 \) may or may not exist, and typical t
Testing conditional moment inequalities and equalities

test-based confidence intervals do not apply. However, one can still test hypotheses on
the parameter, such as

\[ H_0 : \theta_0 = \theta \]  

(2)

for a given value \( \theta \). Andrews and Shi (2013) propose—and the \texttt{cmi.test} command
implements—a test of the above hypothesis. Testing this hypothesis amounts to testing
(1), with \( \theta_0 \) replaced by \( \theta \).

The test is

\[ \phi_n(\theta) = 1\{T_n(\theta) > c_n(\theta, 1 - \alpha)\} \]  

(3)

where \( T_n(\theta) \) is a test statistic, \( c_n(\theta, 1 - \alpha) \) is a simulated critical value, and \( \alpha \) is the
nominal level of the test.

We can then invert the test to construct a confidence set (CS) for \( \theta_0 \). The CS is
defined as

\[ CS_n(1 - \alpha) = \{\theta \in \Theta : \phi_n(\theta) = 0\} \]

The standard way to compute this CS is to consider many grid points in \( \Theta \), compute
\( \phi_n(\theta) \) at each grid point, and collect the values for which \( \phi_n(\theta) = 0 \).

In some cases, it is of interest to test a null hypothesis of the form

\[ E\{m_j(W_i)|X_i\} \geq 0 \text{ for } j = 1, \ldots, p \]

\[ E\{m_j(W_i)|X_i\} = 0 \text{ for } j = p + 1, \ldots, k, \text{ almost surely} \]  

(4)

which does not depend on a parameter \( \theta \), where \( \mathbf{m}(\cdot) := \{m_1(\cdot), \ldots, m_k(\cdot)\}' \) is a vector
of known functions of the observables and \( W_i, X_i, k, p \) are as above. For example,
this arises when one is interested in the sign of a conditional average treatment
effect or the shape of a dose–response function, as discussed in examples 2.1 and 2.2 in
Lee, Song, and Whang (2013). Testing the hypothesis in (4) is the same as testing (2)
in the model in (1). One just replaces \( m(\cdot, \theta) \) with \( m(\cdot) \); consequently, the test in (3)
does not depend on \( \theta \).

Now, we briefly describe a conditional average treatment-effect example of the testing
problem in (4). Let \( D \) be a binary treatment variable, which equals 1 if treated and
0 if untreated. Let \( Y \) be the outcome variable. In the potential-outcome notation,
\( Y = DY(1) + (1 - D)Y(0) \), where \( Y(1) \) is the treated outcome observable only if \( D = 1 \)
and \( Y(0) \) is the untreated outcome observable only if \( D = 0 \). Let \( \mathbf{X} \) be a vector
of covariates. Suppose that \( D \) is randomly assigned, with each individual receiving
treatment with a known probability \( p \). We can then express the average conditional
treatment effect, given \( \mathbf{X} \), as follows:

\[ E\{Y(1) - Y(0)|\mathbf{X}\} = E\left\{\frac{DY}{p} - \frac{(1 - D)Y}{1 - p}\right|\mathbf{X}\} \]

2. You can combine \texttt{cmi.test} with any grid-search algorithm to complete this task. Usually, this
grid search is computationally costly when the dimension of the parameter space is large. One
way to circumvent the computational burden is applying a response surface algorithm for global
optimization introduced by Kaido, Molinari, and Stoye (2016). You can implement this algo-
rithm using a MATLAB toolbox called “DACE”, which is publicly available. So far, we are not
aware of whether this algorithm can be used with Stata commands. For details of “DACE”, see
http://www2.imm.dtu.dk/projects/dace/.
Suppose the researcher wants to test whether the average treatment effect is negative for individuals at all $X$ values. In the framework above, this problem can be written as testing the null hypothesis

$$H_0: E\{m_j(W)|X\} \geq 0$$

where $j = 1$, $W = (Y, D, X)$, and $m_1(W) = -\{(DY)/p\} + [(1 - D)Y]/(1 - p)$.

### 2.2 Confidence intervals based on intersection bounds

The `cmi_interval` command computes the CS for the special, but popular case that the parameter $\theta_0$ is one dimensional, and the moment inequalities provide intersection bounds for this parameter, so the CS of $\theta_0$ is an interval. The `cmi_interval` command combines a one-dimensional grid-search algorithm with `cmi_test` to compute this interval. Specifically, the command applies when the conditional moment inequality model is

$$E\{\rho_{u,j}(W_i) - \theta_0|X_i\} \geq 0 \text{ for } j = 1, \ldots, k_u$$

$$E\{\theta_0 - \rho_{\ell,j}(W_i)|X_i\} \geq 0 \text{ for } j = 1, \ldots, k_\ell$$

where $\theta_0$ is a real-valued parameter and $\rho_{u,j}(\cdot)$ and $\rho_{\ell,j}(\cdot)$ are known functions of the observables. The upper bounds for $\theta_0$ are $E\{\rho_{u,1}(W_i)|X_i\}, \ldots, E\{\rho_{u,k_u}(W_i)|X_i\}$, and the lower bounds are $E\{\rho_{\ell,1}(W_i)|X_i\}, \ldots, E\{\rho_{\ell,k_\ell}(W_i)|X_i\}$. It is easy to see that (5) is a special case of (1), with $p = k_u + k_\ell$, $k = p$, and

$$m_j(W_i, \theta_0) = \begin{cases} 
\rho_{u,j}(W_i) - \theta_0 & \text{for } j = 1, \ldots, k_u \\
\theta_0 - \rho_{\ell,j-k_u}(W_i) & \text{for } j = k_u + 1, \ldots, k_u + k_\ell
\end{cases}$$

The `cmi_interval` command allows one or more upper bounds, $\rho_{u,j}(W_i)$, to be identical to some lower bounds, $\rho_{\ell,j'}(W_i)$. We use a censored data example like that in Andrews and Shi (2014) to illustrate the model in (5). Let $D$ be a binary variable indicating data censorship and $X$ be a covariate vector. Let $Y^*$ be a variable subject to censoring; that is, we observe it only when $D = 1$. Let $\theta_0$ denote the conditional cumulative distribution function (c.d.f.) of $Y^*$ given $X$ evaluated at a certain point, $y_0$. Then, $\theta_0$ is bounded by the inequalities in (5) with $k_u = k_\ell = 1$, and

$$\rho_{u,1}(W) = 1\{Y \leq y_0, D = 1\} + 1\{D = 0\}$$

$$\rho_{\ell,1}(W) = 1\{Y_i \leq y_0, D_i = 1\}$$

We illustrate the implementation of both commands using this example in section 6 below.

### 3 Detailed procedures

In this section, we describe the detailed procedures from Andrews and Shi (2013) that the commands implement. Section 3.1 summarizes the steps in section 9 of Andrews and
Testing conditional moment inequalities and equalities

Shi (2013), and section 3.2 describes the algorithm to compute the confidence interval for the intersection bound model in (5).

3.1 Basic testing procedure

Test statistics

Now we describe the testing procedure that cmi.test implements. Although we focus on testing the hypothesis in (2) for the model in (1), we can also apply the procedure to the hypothesis in (4). Following Andrews and Shi (2013), we transform the conditional moment restrictions in (1) into unconditional moment restrictions before using them to construct the test statistic. The instrumental functions are functions of the instrumental variables, $X_i$. The ones we use are countable hypercubes on standardized $X_i$. We define the standardized $X_i$ variables first. The standardized $X_i$, denoted $X_i^o$, is

$$X_i^o = \Phi \left\{ \Sigma_{X,n}^{-1/2} \left( X_i - X_n \right) \right\}$$

where $X_n = n^{-1} \sum_{i=1}^n X_i \in \mathbb{R}^d$, $\Sigma_{X,n} = n^{-1} \sum_{i=1}^n (X_i - X_n)(X_i - X_n)'$, and $\Phi(x) = \{\Phi(x_1), \ldots, \Phi(x_d)\}'$. The function $\Phi(\cdot)$ denotes the standard normal c.d.f. and $x = (x_1, \ldots, x_d)'$.

The instrumental functions are

$$g_{a,r}(X_i^o) = 1\{X_i^o \in \times_{u=1}^{d_x} [(a_u - 1)/(2r), a_u/(2r)] \}$$

where $a = (a_1, \ldots, a_d)' \in \{1, 2, \ldots, 2r\}^d_x$ and $r = 1, 2, 3, \ldots$. In the implementation, we consider only $r = 1, 2, \ldots, r_{1,n}$ for a positive integer $r_{1,n}$. The cmi.test command uses $\lfloor n^{1/(2d_x)}/2 \rfloor$ as $r_{1,n}$ by default and allows the user to opt for a different positive integer.

Next, we compute the sample average of the unconditional moment functions for each $j = 1, \ldots, k$ and each $(a, r)$ described above. For notational simplicity, in the discussion below, we suppress the possible dependence of $m_j(W_i, \theta)$ on $\theta$ throughout. We have

$$m_{n,j}(g_{a,r}) = n^{-1} \sum_{i=1}^n m_j(W_i) g_{a,r}(X_i^o)$$

We also compute the sample variance, $\hat{\sigma}_{n,j}^2(g_{a,r})$, of $m_j(W_i) g_{a,r}(X_i^o)$. Because $\hat{\sigma}_{n,j}^2(g_{a,r})$ could be zero for some $(a, r)$, we also compute the variance, $\bar{\sigma}_{n,j}^2$, of the conditional moment function, $m_j(W_i)$, to regularize $\hat{\sigma}_{n,j}^2(g_{a,r})$. We use the regularized variance

$$\bar{\sigma}_{n,j}^2(g_{a,r}) = \hat{\sigma}_{n,j}^2(g_{a,r}) + \varepsilon \bar{\sigma}_{n,j}^2$$

in the test statistic. The regularization parameter $\varepsilon$ is 0.05 in cmi.test by default, and the user is allowed to set it to a different small positive number by specifying the epsilon() option. We then construct the test statistic that combines the information in all of these sample moments. After constructing the test statistic, we construct the
critical value $c_n(1 - \alpha)$. There are two versions of the critical value. One is based on
the asymptotic approximation, and the other is based on the bootstrap. The command
implements the former by default. The bootstrap version can be activated by selecting
the \texttt{boot} option. The test statistic and critical values are described next.

By default, \texttt{cmi\_test} uses summation (\texttt{sum}) to aggregate over $j$ for each $(a, r)$ and
uses Cramér–von Mises-type aggregation over $(a, r)$, which yields the following test
statistic,

$$T_n = n \sum_{r=1}^{r_1} \sum_{a \in \{1, \ldots, 2r \}^{d_z}} \left( \sum_{j=1}^{p} \left[ \frac{\sigma_{n,j}(g_{a,r})}{\sigma_{n,j}(g_{a,r})} \right]^2 - \sum_{j=p+1}^{k} \left[ \frac{\sigma_{n,j}(g_{a,r})}{\sigma_{n,j}(g_{a,r})} \right]^2 \right) \frac{1}{(r^2 + 100)(2r)^{d_z}}$$

(9)

where the negative part function $[x]_+ = \max\{0, -x\}$. By specifying the \texttt{sfunc()} and \texttt{ks}
options, \texttt{cmi\_test} allows the user to choose from the Cramér–von Mises max statistic,
the Kolmogorov–Smirnov sum statistic, or the Kolmogorov max statistic.

Choosing max instead of sum replaces the expression in the large brackets in the numerator with

$$\max \left\{ \max_{j=1,2,\ldots,p} \left[ \frac{\sigma_{n,j}(g_{a,r})}{\sigma_{n,j}(g_{a,r})} \right]^2, \max_{j=p+1,p+2,\ldots,k} \left[ \frac{\sigma_{n,j}(g_{a,r})}{\sigma_{n,j}(g_{a,r})} \right]^2 \right\}$$

Choosing the Kolmogorov–Smirnov sum statistic instead of the Cramér–von Mises max statistic replaces $\sum_{r=1}^{r_1} \sum_{a \in \{1, \ldots, 2r \}^{d_z}} / \{(r^2 + 100)(2r)^{d_z}\}$ in (9) with

$$\max_{(a,r):a \in \{1, \ldots, 2r \}^{d_z},r=1,\ldots,r_1,n}$$

\textbf{Asymptotic critical values}

The asymptotic approximation version of the critical value is a simulated quantile of a
statistic (denoted by $T_{n,\text{asy}}$), defined the same as $T_n$, except with $\sigma_{n,j}(g_{a,r})$ replaced by

$$n^{-1/2} \left\{ \nu_{n,j}(g_{a,r}) + \varphi_{n,j}(g_{a,r}) \right\}$$

where $\{\nu_{n,j}(g_{a,r})\}_{j,a,r}$ is a Gaussian random vector that approximates the distribution of
$n^{-1/2} \left\{ \mu_{n,j}(g_{a,r}) - E\{m_j(W_i)g_{a,r}(X_i^a)\} \right\}_{j,a,r}$ and $\varphi_{n,j}(g_{a,r})$ is the generalized moment
selection (GMS) function that approximates $n^{-1/2} E\{m_j(W_i)g_{a,r}(X_i^a)\}$ and selects the
binding moment restrictions.

Specifically, the command simultaneously draws the $k \sum_{r=1}^{r_1}(2r)^{d_z}$ dimensional vec-
tor $\{\nu_{n,j}(g_{a,r})\}_{j=1,\ldots,k,a \in \{1, \ldots, 2r \}^{d_z},r=1,\ldots,r_1,n}$ from a multivariate normal distribution.
The multivariate normal distribution has mean zero, and its variance–covariance ma-
trix is the empirical variance–covariance matrix of

$$\{m_j(W_i)g_{a,r}(X_i^a)\}_{j=1,\ldots,k,a \in \{1, \ldots, 2r \}^{d_z},r=1,\ldots,r_1,n}$$

3. The commands do not incorporate the quasilikelihood-ratio statistic discussed in Andrews and Shi
(2013), because that statistic requires carrying out a quadratic optimization operation many times.
We are not aware of a fast quadratic optimization routine in Stata.
Many draws are taken, and each is used to compute a draw of $T_n^{\text{Asy}}$. The command then computes the empirical $1 - \alpha$ quantile of the sample of $T_n^{\text{Asy}}$ values obtained. This quantile is $c_n(1 - \alpha)$. By default, the number of draws is set to 5,001, and the seed of the random-number generator is set to 10,000. You can change these by specifying the \texttt{rep()} and \texttt{seed()} options.

The GMS function, $\varphi_{n,j}(g_{a,r})$, is

$$\varphi_{n,j}(g_{a,r}) = \begin{cases} \hat{\sigma}_{n,j}B_n & \text{if } \kappa_n^{-1}n^{1/2}\hat{m}_{n,j}(g_{a,r})/\hat{\sigma}_{n,j}(g_{a,r}) > 1 \\ 0 & \text{otherwise} \end{cases}$$

where $B_n$ and $\kappa_n$ are two user-chosen tuning parameters that, in the asymptotic thought experiment, should satisfy $\kappa_n \to \infty$, $\kappa_n/n^{1/2} \to 0$, $B_n \to \infty$, and $B_n/\kappa_n \to 0$ as $n \to \infty$. By default, the command uses the recommended choices from Andrews and Shi (2013): $\kappa_n = \sqrt{0.3}\log n$ and $B_n = \sqrt{0.4}\log n/\log \log n$.

**Bootstrap critical values**

The bootstrap version of the critical value is a simulated quantile of a statistic (denoted by $T_n^{\text{Boot}}$) defined in the same way as $T_n$, except with $\hat{m}_{n,j}(g_{a,r})/\hat{\sigma}_{n,j}(g_{a,r})$ replaced by

$$n^{-1/2}\left\{ \nu_{n,j}^{\text{Boot}}(g_{a,r}) + \varphi_{n,j}(g_{a,r}) \right\} / \hat{\sigma}_{n,j}^{\text{Boot}}(g_{a,r})$$

$\{\nu_{n,j}^{\text{Boot}}(g_{a,r})\}_{j,a,r}$ is a bootstrap approximation of $(n^{1/2}|\hat{m}_{n,j}(g_{a,r}) - E\{m_j(W_{i1})g_{a,r}(X_{i1}^*)\}|)_{j,a,r}$, $\varphi_{n,j}(g_{a,r})$ is the GMS function described above, and $\hat{\sigma}_{n,j}^{\text{Boot}}(g_{a,r})$ is a bootstrap version of $\hat{\sigma}_{n,j}(g_{a,r})$.

Specifically, the command first randomly draws $n$ observations with replacement from the sample $\{W_{i1}\}_{i=1}^n$. These $n$ observations, denoted $\{W_{i1}^*\}_{i=1}^n$, form a bootstrap sample, which is used to compute one draw of $\nu_{n,j}^{\text{Boot}}(g_{a,r})$ and $\hat{\sigma}_{n,j}^{\text{Boot}}(g_{a,r})$. The draw of $\nu_{n,j}^{\text{Boot}}(g_{a,r})$ is

$$n^{1/2}\left\{ \hat{m}_{n,j}(g_{a,r}) - \hat{m}_{n,j}(g_{a,r}) \right\}$$

where we compute $\hat{m}_{n,j}(g_{a,r})$ using the same procedure as that for $\hat{m}_{n,j}(g_{a,r})$, except with $\{W_{i1}^*\}$ (and its subvector $\{X_{i1}^*\}$) replaced by $\{W_{i1}\}$ (and its subvector $\{X_{i1}\}$). We compute the draw of $\hat{\sigma}_{n,j}^{\text{Boot}}(g_{a,r})$ using the same procedure as that for $\hat{\sigma}_{n,j}(g_{a,r})$, except with $\{W_{i1}^*\}$ (and its subvector $\{X_{i1}^*\}$) replacing $\{W_{i1}\}$ (and its subvector $\{X_{i1}\}$). We then use these to compute one draw of $T_n^{\text{Boot}}$. By repeating the process, we take many $T_n^{\text{Boot}}$ draws, with $c_n(1 - \alpha)$ defined as the $1 - \alpha$ empirical quantile of these draws. By default, the number of draws is set to 5,001, and the seed of the random-number generator is set to 10,000. You can change these by specifying the \texttt{rep()} and \texttt{seed()} options.

### 3.2 Confidence interval construction for intersection bound models

In this section, we describe the algorithm to construct a confidence interval for the one-dimensional parameter in the model in (5).
One-sided bound

If either $k_u$ or $k_l$ is zero, the model gives a one-sided bound for the parameter. In this case, the command uses the following algorithm consisting of iterative steps, where step $(-1)$ is the preparation step, and for $i \geq 0$, step $(i)$ finds the confidence interval bounds for $\theta_0$ up to the $i$th digit after the decimal point.

**Step (-1).** First, we set a preliminary lower (upper) bound of the confidence interval:

$$\hat{\theta}_{\text{lb,pre}} = \min_{1 \leq j \leq k_l} \min_{i \leq n} \rho_{l,j}(W_i)$$

$$\hat{\theta}_{\text{ub,pre}} = \max_{1 \leq j \leq k_u} \max_{i \leq n} \rho_{u,j}(W_i)$$

In addition, we define two auxiliary statistics:

$$\hat{\theta}_{\text{lb,bound}} = \max_{1 \leq j \leq k_l} \max_{i \leq n} \rho_{l,j}(W_i)$$

$$\hat{\theta}_{\text{ub,bound}} = \min_{1 \leq j \leq k_u} \min_{i \leq n} \rho_{u,j}(W_i)$$

Note that $\hat{\theta}_{\text{lb,pre}} (\hat{\theta}_{\text{ub,pre}})$ is a preliminary conservative lower (upper) bound for the confidence interval. Meanwhile, $\hat{\theta}_{\text{lb,bound}} (\hat{\theta}_{\text{ub,bound}})$ is trivially contained in the one-sided confidence interval and thus is greater (smaller) than the lower (upper) bound. The following steps take advantage of these conservative bounds.

We explain the method for deriving the lower bound here. The upper bound method is analogous.

**Step (0).** If the distance between $\lfloor \hat{\theta}_{\text{lb,pre}} \rfloor$ and $\lceil \hat{\theta}_{\text{lb,bound}} \rceil$ is 1, skip the current step, let $\hat{\theta}_{\text{lb,0}} = \lfloor \hat{\theta}_{\text{lb,pre}} \rfloor$, and move to the next step. Otherwise, consider grid points on $[\lfloor \hat{\theta}_{\text{lb,pre}} \rfloor, \lceil \hat{\theta}_{\text{lb,bound}} \rceil]$ with distance between adjacent grid points being $d_0 = \lceil \max\{(\hat{\theta}_{\text{lb,bound}}) - (\hat{\theta}_{\text{lb,pre}})/20, 1\} \rceil$. Apply *cmi_test* for $\theta_0$ being each of these grid points. Record the largest grid point rejected by the test as $\hat{\theta}_{\text{lb,0}}$, and consider grid points on $[\hat{\theta}_{\text{lb,0}}, \hat{\theta}_{\text{lb,0}} + d_0]$ with the updated spacing between grids, $d_1 = \lceil \max(d_0/2, 1) \rceil$. Repeat until the distance equals 1. Record the smallest $\theta_0$ value not rejected and subtract 1. Let the resulting number be $\hat{\theta}_{\text{lb,0}}$.

**Step (1).** Apply *cmi_test* for $\theta_0$ being each of the points $\hat{\theta}_{\text{lb,0}}, \hat{\theta}_{\text{lb,0}} + 0.1, \ldots, \hat{\theta}_{\text{lb,0}} + 0.9$. Record the smallest point not rejected and subtract 0.1. Let the resulting number be $\hat{\theta}_{\text{lb,1}}$.

...  

**Step (i+1).** Apply *cmi_test* for $\theta_0$ being each of the points $\hat{\theta}_{\text{lb,i}}, \hat{\theta}_{\text{lb,i}} + 10^{-(i+1)}, \ldots, \hat{\theta}_{\text{lb,i}} + 9 \times 10^{-(i+1)}$. Record the smallest point not rejected and subtract $10^{-(i+1)}$. Let the resulting number be $\hat{\theta}_{\text{lb,i+1}}$.  


Testing conditional moment inequalities and equalities

By default, the command iterates this algorithm up to the thousandth place [that is, step (3)]. One can choose the number of iterations (that is, the accuracy of the confidence interval) by specifying the deci() option.

Two-sided bound

When \( k_L > 0 \) and \( k_u > 0 \), the model gives two-sided bounds for the parameter \( \theta_0 \). In this case, we first obtain two one-sided confidence intervals, each of confidence level \( 1 - \alpha/2 \). The two one-sided confidence intervals separately use the upper-bound and the lower-bound moment inequalities. The algorithm then forms a preliminary two-sided confidence interval (of nominal level \( 1 - \alpha \)) by intersecting the two one-sided bounds. If the two one-sided bounds do not intersect, cmi_interval terminates and returns an empty set. This implies that the model is rejected at the specified confidence level \( \alpha \) (the default is 95%).

Let \( \hat{\theta}_{lb,-1} \) and \( \hat{\theta}_{ub,-1} \) be the lower and upper bounds of the crude interval just specified. We then obtain the Andrews and Shi (2013) confidence interval by applying the following algorithm.

Step (0). Check the length of the crude interval. If it is less than 2, then skip step (0), let \( \hat{\theta}_{lb,0} = \hat{\theta}_{lb,-1}, \hat{\theta}_{ub,0} = \hat{\theta}_{ub,-1} \), and move to the next step. Otherwise, set \( d_0 = \lfloor \max\{ (\hat{\theta}_{ub,-1} - \hat{\theta}_{lb,-1}) / 20, 1 \} \rfloor \), and apply cmi_test using all inequalities for each of the evenly spaced grid points (including the endpoints) with spacing \( d_0 \) on \( [\hat{\theta}_{lb,-1}, \hat{\theta}_{ub,-1}] \).

Case 1: If at least one grid point is not rejected, let \( \theta_{lb,d_0} \) and \( \theta_{ub,d_0} \) denote the smallest and the largest nonrejected points, respectively.

Case 2: If all points are rejected, find the grid point with the largest \( p \)-value (denoted by \( \theta_{high,d_0} \)) and let \( \theta_{lb,d_0} = \theta_{high,d_0} \).

For both cases, let \( d_1 = \lfloor \max(d_0/2, 1) \rfloor \). Consider evenly spaced grid points (including endpoints) with spacing \( d_1 \) on \( [\theta_{lb,d_0} - d_0, \theta_{lb,d_0}] \) and also those on \( [\theta_{ub,d_0}, \theta_{ub,d_0} + d_0] \). Apply cmi_test using all inequalities for each of these grids. Repeat the checks in case 1 and case 2 above, and define \( \theta_{lb,d_1} \) and \( \theta_{ub,d_1} \) analogously to \( \theta_{lb,d_0} \) and \( \theta_{ub,d_0} \), respectively. Iterate this step until \( d_j = 1 \), then let \( \theta_{lb,0}, \theta_{ub,0} = [\theta_{lb,d_j} - d_j, \theta_{ub,d_j} + d_j] \). This interval is the Andrews and Shi (2013) confidence interval accurate up to the integer level. If you desire higher accuracy, move on to the next step.

... Step (i+1). If \( \hat{\theta}_{lb,i+1} - \hat{\theta}_{lb,i} \leq 2 \times 10^{-i+1} \), let \( \hat{\theta}_{lb,i+1} = \hat{\theta}_{lb,i}, \hat{\theta}_{ub,i+1} = \hat{\theta}_{ub,i} \) and move to the next step. Otherwise, consider evenly spaced grid points with spacing \( 10^{-i+1} \) on the intervals \( [\hat{\theta}_{lb}, \hat{\theta}_{lb} + 10^{-i}] \) and \( [\hat{\theta}_{ub} - 10^{-i}, \hat{\theta}_{ub}] \) (including endpoints). Apply cmi_test for \( \theta_0 \) being each of these grid points.
Case 1: If at least one point is not rejected, let $\theta_{lb,j}$ and $\theta_{ub,j}$ denote the smallest and the largest such point, respectively.

Case 2: If all points are rejected, find the point with the largest $p$-value (denoted by $\theta_{high,i}$), and let $\theta_{lb,i+1} = \theta_{ub,i+1} = \theta_{high,i+1}$. Let $[\theta_{lb,i+1}, \theta_{ub,i+1}] = [\theta_{lb,i+1} - 10^{-i-1}, \theta_{ub,i+1} + 10^{-i-1}]$. This interval is the Andrews and Shi (2013) confidence interval with accuracy up to $10^{-i-1}$. If you desire higher accuracy, move on to the next step.

Iterate until the desired accuracy is reached. cmi.interval iterates this algorithm up to the thousandth place by default. The user can set the accuracy level differently with deci().

Remark. If the confidence interval is narrower than the smallest grid, for example, $10^{-k}$ ($10^{-3}$ in the default setup), cmi.interval finds a grid point with the highest $p$-value, $\hat{\theta}_p$, and returns $[\hat{\theta}_p - 10^{-k}, \hat{\theta}_p + 10^{-k}]$ as the confidence interval. One may adjust the last digit of the confidence interval with deci() or by rescaling $m_{u,j}(W_i)$ and $m_{l,j}(W_i)$ by multiplying all of them by an appropriate power of 10 to get a more accurate confidence interval.

4 The cmi.test command

4.1 Syntax

The syntax of cmi.test is as follows:

```
  cmi.test ([cmi_vars]) ([cme_vars]) indepvars [if] [in] [ , rnum(#) hd boot ks sfunc(#) epsilon(real) kap(real) bn(real) rep(#) seed(#) simul]
```

4.2 Description

cmi.test implements the test described in section 3.1 for the hypothesis in (2) and the model in (1) [or the hypothesis in (4)]. To use this command, one first generates variables that equal $m_1(W_i, \theta), \ldots, m_k(W_i, \theta)$ for observations $i = 1, \ldots, n$ [or $m_1(W_i), \ldots, m_k(W_i)$ for observations $i = 1, \ldots, n$]. The first $p$ are cmi_vars, and the next $k - p$ are cme_vars. The command allows cmi_vars or cme_vars to be empty. The variables in $X_i$ are indepvars.

As described in section 3.1, cmi.test uses countable hypercubes as the collection of instrumental functions. They are constructed according to (8) above by default. That choice is fine when the number of indepvars is three or fewer. When the dimension of indepvars is greater than three, the number of cubes may be too large, resulting in long computation time. The command allows an alternative method for high-dimensional independent variables. The user can select the hd option to choose this method. This option implements the method described in the last paragraph of section 9 of Andrews and Shi (2013).
4.3 Options

\texttt{rnum(\#)} sets a scalar indicating the minimum side-edge lengths. The default is the smallest integer greater than \(n^{d_x/2}/2\), where \(d_x\) is the dimension of \texttt{indepvars}.

\texttt{hd} uses an alternative method for high-dimensional independent variables. This option is designed for three or more covariates; see the previous subsection for details.

\texttt{boot} lets the user turn on the bootstrap option. If the user does not specify this option, the command computes the critical value based on a Gaussian asymptotic approximation.

\texttt{ks} uses the Kolmogorov–Smirnov-type statistic. The default is the Cramér–von Mises-type statistic.

\texttt{sfunc(\#)} sets the function \(S\) to specify the form of the test statistic. \texttt{sfunc(1)} yields the modified method of moments or sum function, and \texttt{sfunc(3)} yields the max function. The default is \texttt{sfunc(1)}.

\texttt{epsilon(real)} sets the regularization parameter \(\varepsilon\) for the sample variances. The default is \texttt{epsilon(0.05)}.

\texttt{kap(real)} and \texttt{bn(real)} are two tuning parameters in the data-dependent GMS function \(\varphi_n(g_{a,r})\). The default for the former is \((0.3 \log n)^{3/2}\) and for the latter is \((0.4 \log n)/(\log \log n))^{3/2}\).

\texttt{rep(\#)} sets the number of repetitions for the critical value simulations. The default is \texttt{rep(5001)}.

\texttt{seed(\#)} sets the random-number seed for the critical value simulations. The default is \texttt{seed(10000)}.

\texttt{simul} lets the user choose to leave the seed number for the critical value simulations unset. Use this option when the command is inside a Monte Carlo simulation loop to not interfere with the random-number generation process set for the Monte Carlo simulation exercise.
4.4 Stored results

cmi_test stores the following in r():

Scalars
- \( r(N) \): number of observations
- \( r(\text{stat}) \): test statistic
- \( r(pval) \): p-value
- \( r(cv01) \): critical value for the 1% significance level
- \( r(cv05) \): critical value for the 5% significance level
- \( r(cv10) \): critical value for the 10% significance level
- \( r(kappa) \): tuning parameter \( \kappa_n \)
- \( r(B) \): tuning parameter \( B_n \)
- \( r(\epsilon) \): tuning parameter \( \epsilon \)
- \( r(\text{rep}_cv) \): repetitions for critical values
- \( r(\text{obs}) \): average number of observations in the smallest cubes
- \( r(r_n) \): index for minimum side-edge lengths
- \( r(ncube) \): number of cubes

Macros
- \( r(\text{cmd}) \): cmi_test
- \( r(\text{title}) \): title in output
- \( r(\text{m}_\text{ineq}) \): varlist for conditional moment inequalities, if any
- \( r(\text{m}_\text{eq}) \): varlist for conditional moment equalities, if any
- \( r(\text{x}) \): varlist for instrumental variables

5 The cmi_interval command

5.1 Syntax

The syntax of cmi_interval is as follows:

\[
\text{cmi_interval} \ (\text{[lower_bound_vars]} \ (\text{[upper_bound_vars]} \ \text{indepvars} \ [\text{if}] \ [\text{in}] \\
[\text{, level(\text{real})} \ \text{deci(\#)} \ \text{rnum(\#)} \ \text{hd boot ks sfunc(\#)} \ \text{epsilon(\text{real})} \\
\text{kap(\text{real})} \ \text{bn(\text{real})} \ \text{rep(\#)} \ \text{seed(\#)} \ \text{simul]} \n\]

5.2 Description

cmi_interval constructs the confidence interval for the parameter in (5) by inverting cmi_test. The \( \text{upper_bound-vars} \) are \( \rho_{u,1}(W_i), \ldots, \rho_{u,k_u}(W_i) \). The \( \text{lower_bound-vars} \) are \( \rho_{l,1}(W_i), \ldots, \rho_{l,k_l}(W_i) \). The \( \text{indepvars} \) are the elements of \( X_i \).

5.3 Options

cmi_interval accepts all the options that cmi_test does. Two additional options are available to cmi_interval, which are the following:

- \( \text{level(\text{real})} \): sets the confidence level \( 1 - \alpha \), where \( 1 - \alpha \) is the nominal confidence level. The default is \( \text{level}(0.95) \).
deci(\#) sets the accuracy of the confidence interval bounds as measured by the number of digits after the decimal point.

5.4 Stored results

cmi_interval stores the following in r():

Scalars
- r(N): number of observations
- r(lbound): estimated lower bound (if any)
- r(ubound): estimated upper bound (if any)
- r(level): confidence level
- r(ncube): number of cubes
- r(kappa): tuning parameter \( \kappa \)
- r(epsilon): tuning parameter \( \epsilon \)
- r(rep_cv): repetitions for critical values
- r(a_obs): average number of observations in the smallest cubes
- r(r_n): index for minimum side-edge lengths
- r(B): tuning parameter \( B \)

Macros
- r(cmd): cmi_interval
- r(title): title in output
- r(lbvar): varlist for conditional moment inequalities for the lower bound, if any
- r(ubvar): varlist for conditional moment inequalities for the upper bound, if any
- r(x): varlist for instrumental variables

6 Examples

In this section, we provide an example of estimating a conditional distribution with censored data, which was introduced earlier in section 2.1. We use the data for male employees who are not self-employed from the 15th round (year 2011) of the National Longitudinal Survey of Youth 1997. From that dataset, we take the log hourly dollar wage (\( Y \)), the dummy for college enrollment (\( D \)), the year of education of father (\( X_1 \)), and the year of education of mother (\( X_2 \)). The number of observations is 2,054.

Let \( Y_i^* \) be the natural logarithm of the potential wage after college enrollment. This variable is observed only for those who actually enrolled in a college. Suppose that the parameter of interest is \( \theta_0 \equiv F_{Y_*}(y_0) \), that is, the c.d.f. \( Y_i^* \) evaluated at \( y_0 \). The parameter \( \theta_0 \) is then bounded by (5), with the bounding moment functions defined in (6) and (7). See Andrews and Shi (2014) for details.

For the rest of the example, define \( \theta_0 = F_{Y_*}\{\log(20)\} \). In other words, \( \theta_0 \) is the percentage of the subpopulation (currently working, not self-employed male) whose expected hourly wage is lower than \$20 if he had enrolled in a college. We create two variables defined by \( 1\{Y_i \leq y_0, D_i = 1\} \) and \( 1\{Y_i \leq y_0, D_i = 1\} + 1\{D_i = 0\} \):

\[ .
\text{use cmitest}
\]
\[ .
\text{local y0 = log(20)}
\]
\[ .
\text{generate lbound = (Y < `y0´) * D}
\]
\[ .
\text{generate ubound = (Y < `y0´) * D + 1 - D}
\]


6.1 \texttt{cmi\_test}

Suppose that the research question is whether 0.5 is the value of $\theta_0$. That is, we would like to test

$$H_0: F_Y \cdot \{\log(20)\} = 0.5 \quad (10)$$

Then, the researcher creates two conditional moment inequalities [(6) and (7)] by using the following commands:

```plaintext
. local theta0 = 0.5
. generate CMI1 = `theta0` - lbound
. generate CMI2 = ubound - `theta0`
```

\texttt{cmi\_test} results are

```plaintext
. cmi_test (CMI1 CMI2) ( ) X1 X2
Conditional Moment Inequalities Test
Number of obs : 2054

<Variables>
Conditional Moment Inequalities : CMI1 CMI2
No Conditional Moment Equality
Instruments : X1 X2

<Methods>
Countable Hyper Cubes
Asymptotic Critical Value
Cramer-von Mises-type statistic / Sum function

<Results>
Test Statistic : 0.0331
Critical Value (1%) : 0.2347
(5%) : 0.1698
(10%) : 0.1446
p-value : 0.9882
```

```plaintext
. cmi_test (CMI1 CMI2) ( ) X1 X2, ks
Conditional Moment Inequalities Test
Number of obs : 2054

<Variables>
Conditional Moment Inequalities : CMI1 CMI2
No Conditional Moment Equality
Instruments : X1 X2

<Methods>
Countable Hyper Cubes
Asymptotic Critical Value
Kolmogorov-Smirnov-type statistic / Sum function

<Results>
Test Statistic : 0.8413
Critical Value (1%) : 5.7807
(5%) : 4.1274
(10%) : 3.3612
p-value : 0.9438
```
Testing conditional moment inequalities and equalities

```
. cmi_test (CMI1 CMI2) ( ) X1 X2, sfunc(3) boot
Conditional Moment Inequalities Test
Number of obs : 2054

<Variables>
Conditional Moment Inequalities : CMI1 CMI2
No Conditional Moment Equality
Instruments : X1 X2

<Methods>
Countable Hyper Cubes
Bootstrap Critical Value
Cramer-von Mises-type statistic / Max function

<Results>
Test Statistic : 0.0331
Critical Value (1%) : 0.2687
(5%) : 0.1840
(10%) : 0.1554
p-value : 0.9960
```

The first result shows the `cmi_test` outcome with default options. The second result uses the Kolmogorov–Smirnov-type statistic. The last result uses the max function in the test statistic and uses the bootstrapped critical value. All three versions of the test yield high p-values, indicating that 0.5 is not rejected even at significance level 10%.

Note that the example given here is for an inequalities-only model. If a model contains conditional moment equalities, you should position variables representing those equalities in the second parenthesis of the syntax.

### 6.2 cmi_interval

Now we compute a confidence interval for \( \theta_0 \). In this example, the `lbound` and `ubound` variables represent `lower_bound_vars` and `upper_bound_vars`, respectively. `cmi_interval` returns the following results:

```
. cmi_interval (lbound) (ubound) X1 X2
Conditional Moment Inequalities Interval
Number of obs : 2054

<Variables>
Variables for the Lower Bound : lbound
Variables for the Upper Bound : ubound
Instruments : X1 X2

<Methods>
Countable Hyper Cubes
Asymptotic Critical Value
Cramer-von Mises-type statistic / Sum function

<Results>
.95 confidence interval is:
( 0.413 , 0.621 )
```
The first case uses the default options and yields the 95% confidence interval: [0.413, 0.621]. The second case uses the max function for the test statistic and yields almost the same result as the first case.

In the third case, we omit ubound. This case illustrates how to construct a one-sided CS. Suppose only the lower bounds for the parameter exist [that is, \(k_u = 0\) in (5)]. When we empty the second bracket of the syntax, the command gives a one-sided confidence interval. The third case also activates the level(0.9) option. Thus the resulting confidence level is 90%. It also activates deci(2), yielding results with accuracy up to the second digit.

7 References


Testing conditional moment inequalities and equalities


About the authors

Donald W. K. Andrews is the T. C. Koopmans Professor of Economics and Statistics at Yale University.

Wooyoung Kim is a graduate student at the University of Wisconsin–Madison.

Xiaoxia Shi is an assistant professor at the University of Wisconsin–Madison.
A comprehensive set of postestimation measures to enrich interrupted time-series analysis

Ariel Linden
Linden Consulting Group, LLC
Ann Arbor, MI
alinden@lindenconsulting.org

Abstract. While the primary goal of interrupted time-series analysis (ITSA) is to evaluate whether there is a change in the level or trend of an outcome following an interruption (for example, policy change, intervention initiation), a series of additional measures may be relevant to the analysis. In this article, I seek to fill a gap in the ITSA literature by describing a comprehensive set of measures that can be computed following ITSA models, including those that fulfill the primary goal and those that provide supplementary information about trends. These measures can be calculated using the itsa command; this article therefore serves as a complement to “Conducting interrupted time-series analysis for single and multiple group comparisons” (Linden, 2015, Stata Journal 15: 480–500), which introduced the itsa command. Specific ITSA postestimation measures described in this article include individual trend lines, comparisons between multiple interventions, and comparisons with a counterfactual.

Keywords: st0389_3, itsa, interrupted time-series analysis, quasiexperimental designs, causal inference, counterfactual

1 Introduction

Interrupted time-series analysis (ITSA) is a suitable evaluation approach when a single unit is being studied (that is, individual, city, state, country), when the outcome variable is serially ordered as a time series, and when multiple observations are captured in both the preintervention and postintervention periods (Linden and Adams 2011; Linden 2015). The study design is called an interrupted time series because the intervention is expected to “interrupt” the level or trend of the time series, subsequent to its introduction (Campbell and Stanley 1966; Shadish, Cook, and Campbell 2002). ITSA has strong internal validity, even in the absence of a comparison group, primarily because of its control over the effects of regression to the mean (Campbell and Stanley 1966). When the treatment group’s outcomes can also be contrasted with those of one or more comparison groups, the internal validity is further enhanced by allowing the researcher to potentially control for confounding omitted variables (Linden 2015). Additionally, ITSA has strong external validity when the unit of measure is at the population level, or when the results can be generalized to other units, treatments, or settings (Shadish, Cook, and Campbell 2002; Linden, Adams, and Roberts 2004).
A comprehensive set of postestimation measures

Linden (2015) introduced the itsa command, which estimates the effect of an intervention on an outcome variable for a single treatment group or when compared with one or more control groups. Its options allow the user to estimate treatment effects for multiple interventions and control for autocorrelated disturbances.

While itsa provides many measures of primary interest, both as part of model estimation and postestimation, investigators may find a number of additional measures helpful. The existing ITSA literature lacks guidance on computing such measures and why they may be useful. Thus, in this article, I will provide calculations for a comprehensive set of measures that an investigator may want to pursue for a study using the ITSA framework.

2 Method and formulas

Other literature has provided both a comprehensive description of the ITSA design and methodological guidance in its implementation (see Box and Tiao [1975], Glass, Willson, and Gottman [1975], and McDowall et al. [1980] for using autoregressive integrated moving-average (ARIMA) models; and see Crosbie [1993], Gottman [1981], Linden and Adams [2011], Linden [2015], McKnight, McKean, and Huitema [2000], Simonton [1977a], and Velicer and McDonald [1991] for using ordinary least-squares (OLS) regression-based models).

itsa relies on OLS rather than ARIMA models because the former is often more flexible and broadly applicable in an interrupted time-series context (Box et al. 2016; Velicer and Harrop 1983). Additionally, investigators trained in conventional statistical methods are generally more familiar with the OLS framework than the more complex ARIMA framework. Below I briefly describe the models for single-group and multiple-group comparisons. See Linden and Adams (2011) and Linden (2015) for a more comprehensive discussion.

2.1 The single-group analysis

When there is only one group under study (no comparison groups), and only a single treatment period, the standard ITSA regression model assumes the following form (Huitema and McKean 2000; Linden and Adams 2011; Linden 2015; Simonton 1977a; Simonton 1977b):

\[ Y_t = \beta_0 + \beta_1 T_t + \beta_2 X_t + \beta_3 X_t T_t + \epsilon_t \]  

(1)

\( Y_t \) is the aggregated outcome variable measured at each equally spaced time point \( t \), \( T_t \) is the time since the start of the study, \( X_t \) is a dummy (indicator) variable representing the intervention (preintervention periods 0, otherwise 1), and \( X_t T_t \) is an interaction term. For a single-group study, \( \beta_0 \) represents the intercept or starting level of the outcome variable. \( \beta_1 \) is the slope or trend of the outcome variable until the introduction of the intervention. \( \beta_2 \) represents the change in the level of the outcome that occurs in the period immediately following the introduction of the intervention. \( \beta_3 \) represents the
difference between preintervention and postintervention slopes of the outcome. Thus we look for significant $p$-values in $\beta_2$ to indicate an immediate treatment effect or in $\beta_3$ to indicate a treatment effect over time (Linden and Adams 2011; Linden 2015).

In the model for a single treatment group exposed to a single intervention model, there are three additional measures of potential interest: the trend of the time series prior to introduction of the intervention, the trend of the time series after introduction of the time series, and the difference between the preintervention and postintervention trends. Two of these measures are part of the regression output: the preintervention trend, $\beta_1$, and the difference between the preintervention and postintervention trends, $\beta_3$. Thus, to measure all trend components, one needs to calculate the postintervention trend, $\beta_1 + \beta_3$.

For a second intervention, (1) is modified accordingly:

$$Y_t = \beta_0 + \beta_1 T_t + \beta_2 X_{1t} + \beta_3 X_{1t} T_{1t} + \beta_4 X_{2t} + \beta_5 X_{2t} T_{2t} + \epsilon_t$$

(2)

Here $X_{2t}$ and $X_{2t} T_{2t}$ are variables representing the additional treatment period in the study. Accordingly, $\beta_4$ represents the change in the level of the outcome that occurs in the period immediately following the introduction of the second intervention, and $\beta_5$ represents the difference between the first-intervention and second-intervention slopes of the outcome. We add corresponding $X_n$ and $X_n T_n$ variables to the model for each additional intervention included.

In the model for a single treatment group exposed to two sequentially administered interventions, there are six measures of interest: the trends in each of the three periods (preintervention and the two intervention periods) and the differences between each period’s trends (preintervention versus the first intervention, preintervention versus the second intervention, first intervention versus the second intervention). Three of these measures are provided in the regression output: the preintervention trend, $\beta_1$; the difference between the preintervention trend and the first-intervention trend, $\beta_3$; and the difference between the first- and second-intervention trends, $\beta_5$. Thus the three remaining measures of interest that require calculation are the first-intervention period trend, $\beta_1 + \beta_3$; the second-intervention period trend, $\beta_1 + \beta_3 + \beta_5$; and the difference between the second-intervention period trend and the preintervention trend, $\beta_3$.

When there is either a single treatment period or two treatment periods, the investigator may be interested in determining the counterfactual outcome, that is, the outcome at time-point $T$ had the intervention not been introduced. As an example, assume we have a dataset containing 10 observation periods, of which the first 5 are the preintervention period and the last 5 represent the intervention period (thus 1 intervention was introduced). We calculate the predicted outcome for the actual intervention in the last (10th) observation period (although the investigator may choose any time period for comparison) using the ITSA model’s parameter coefficients: $\beta_0 + (\beta_1 \times 10) + (\beta_2 \times 1) + (\beta_3 \times 4)$ (see Linden [2015] for details of the data variables corresponding to the model parameters). We calculate the counterfactual outcome for the last observation period as $\beta_0 + (\beta_1 \times 10)$. Thus, in this example, the difference in the outcome at the 10th observation period between the treatment and the counterfactual is $(\beta_2 \times 1) + (\beta_3 \times 4)$. 
A comprehensive set of postestimation measures

With additional interventions in the study, this last equation is expanded to include the corresponding $X_n$ and $X_nT_n$ parameters.

### 2.2 The multiple-group analysis

When one or more control groups are available for comparison and there is only one intervention, the regression model in (1) is expanded to include four additional terms ($\beta_4$ to $\beta_7$) (Linden and Adams 2011; Linden 2015; Simonton 1977a; Simonton 1977b):

$$Y_t = \beta_0 + \beta_1 T_t + \beta_2 X_t + \beta_3 X_t T_t + \beta_4 Z + \beta_5 Z T_t + \beta_6 Z X_t + \beta_7 Z X_t T_t + \epsilon_t$$  \hspace{1cm} (3)

Here $Z$ is a dummy variable denoting the cohort assignment (treatment or control), and $Z T_t$, $Z X_t$, and $Z X_t T_t$ are all interaction terms among previously described variables. The coefficients $\beta_0$ to $\beta_3$ represent the control group, and the coefficients $\beta_4$ to $\beta_7$ represent values of the treatment group. More specifically, $\beta_4$ represents the difference in the level (intercept) of the outcome variable between treatment and controls prior to the intervention, $\beta_5$ represents the difference in the slope (trend) of the outcome variable between treatment and controls prior to the intervention, $\beta_6$ indicates the difference between treatment and control groups in the level of the outcome variable immediately following introduction of the intervention, and $\beta_7$ represents the difference between treatment and control groups in the slope (trend) of the outcome variable after initiation of the intervention compared with the preintervention (Linden and Adams 2011; Linden 2015).

In this model, there are nine measures of interest: the preintervention and postintervention trends for the control group, the preintervention and postintervention trends for the treatment group, the differences in trends between groups in each period, and the differences between groups in their preintervention and postintervention trend differences (difference in differences of trends). The regression output provides four of these measures: the preintervention trend for the control group, $\beta_1$; the difference between the treatment and control group in their preintervention trends, $\beta_5$; the difference in the control group’s preintervention and postintervention trends, $\beta_3$; and the difference in pre-post trends between the treatment and control group, $\beta_7$ (which may be considered a difference-in-differences of slopes). The five remaining measures of interest to calculate are the preintervention trend for the treatment group, $\beta_5 + \beta_1$; the control group’s trend in the postintervention period, $\beta_1 + \beta_3$; the treatment group’s trend in the postintervention period, $\beta_1 + \beta_3 + \beta_5 + \beta_7$; the difference between the treatment and control group’s trends in the postintervention period, $\beta_5 + \beta_7$; and the difference in the treatment group’s preintervention to postintervention trend, $\beta_3 + \beta_7$.

In the case of a second intervention, (2) is modified accordingly:

$$Y_t = \beta_0 + \beta_1 T_t + \beta_2 X_{1t} + \beta_3 X_{1t} T_{1t} + \beta_4 Z + \beta_5 Z T_{1t} + \beta_6 Z X_{1t} + \beta_7 Z X_{1t} T_{1t} + \beta_8 X_{2t} + \beta_9 X_{2t} T_{2t} + \beta_{10} Z X_{2t} + \beta_{11} Z X_{2t} T_{2t} + \epsilon_t$$  \hspace{1cm} (4)

Here $X_{2t}$, $X_{2t} T_{2t}$, $Z X_{2t}$, and $Z X_{2t} T_{2t}$ are variables representing the additional treatment period in the study, and their connotation is interpreted as described in (2).
In this model containing 2 interventions, there are 18 total measures of interest: the preintervention, first-intervention, and second-intervention trends for the treatment group and the control group; the differences between groups in their trends in each of these periods, the differences between each period’s trends for the treatment group and control group (preintervention versus the first intervention, preintervention versus the second intervention, first intervention versus the second intervention), and the contrast between groups for each of these periodic comparisons. The regression output provides six of these measures: the preintervention trend for the control group, \( \beta_1 \); the difference between the treatment and control group’s preintervention trends, \( \beta_5 \); the control group’s difference in preintervention to first-intervention trends, \( \beta_3 \); the difference between the treatment and control groups’ differences in their preintervention to first-intervention trends, \( \beta_7 \); the change in the control group’s first-intervention to second-intervention trends, \( \beta_9 \); and the difference between the treatment and control groups’ differences in their first- to second-intervention trends, \( \beta_{11} \).

The 12 remaining measures of interest for calculation are the treatment group’s preintervention trend, \( \beta_5 + \beta_1 \); the control group’s first-intervention period trend, \( \beta_1 + \beta_3 \); the treatment group’s first-intervention period trend, \( \beta_1 + \beta_3 + \beta_5 + \beta_7 \); the difference between the groups in the first-intervention period, \( \beta_5 + \beta_7 \); the control group’s trend in the second-intervention period, \( \beta_1 + \beta_3 + \beta_9 \); the treatment group’s trend in the second-intervention period, \( \beta_1 + \beta_3 + \beta_5 + \beta_7 + \beta_9 + \beta_{11} \); the difference between the groups in the second-intervention period, \( \beta_7 + \beta_9 + \beta_{11} \); the treatment group’s difference in trends between the preintervention and first-intervention period, \( \beta_3 + \beta_7 \); the treatment group’s difference in trends between the first- and second-intervention periods, \( \beta_9 + \beta_{11} \); the control group’s difference in trends between the preintervention and second-intervention period, \( \beta_3 + \beta_9 \); the treatment group’s difference in trends between the preintervention and second-intervention period, \( \beta_3 + \beta_5 + \beta_7 + \beta_9 + \beta_{11} \); and the difference between the treatment and control group in comparing the differences in trends of the preintervention and second-intervention period, \( \beta_5 + \beta_7 + \beta_{11} \).

In contrast to the single-group ITSA, where the counterfactual estimates are based solely on the treatment group’s preintervention trend estimates carried forward, in a multiple-group ITSA, the control group serves as the counterfactual to the treatment group for all estimates.

3 Stata implementation of ITSA measures of interest

Stata can readily estimate all the measures of interest described in section 2 after running \texttt{itsa} using the \texttt{lincom} command, which provides point estimates as well as other important metrics, such as \( p \)-values and confidence intervals (CI). Table 1 provides a crosswalk between each measure of interest and the respective \texttt{lincom} specification after \texttt{itsa}. 
Table 1. Crosswalk between model parameters and estimates following *itsa*, for all measures of interest

<table>
<thead>
<tr>
<th>Measure of interest</th>
<th>Model parameters</th>
<th><em>lincom</em> specification (following <em>itsa</em>)</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single group—Single intervention (1)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Preintervention trend</td>
<td>$\beta_1$</td>
<td><code>lincom .b[.t]</code></td>
<td>In regression output</td>
</tr>
<tr>
<td>Postintervention trend</td>
<td>$\beta_1 + \beta_3$</td>
<td><code>lincom .b[.t] + .b[,x_t(trp)]</code></td>
<td><em>itsa posttrend</em> option</td>
</tr>
<tr>
<td>Difference preintervention versus postintervention trend</td>
<td>$\beta_3$</td>
<td><code>lincom .b[,x_t(trp)]</code></td>
<td>In regression output</td>
</tr>
<tr>
<td>Single group—Comparison to counterfactual</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Counterfactual value at time-point $T$</td>
<td>$\beta_0 + \beta_1 \times T$</td>
<td><code>lincom .b[.cons] + .b[.t] * T</code></td>
<td></td>
</tr>
<tr>
<td>Adjusted value for treatment at time-point $T$</td>
<td>$\beta_0 + (\beta_1 \times T) + (\beta_2 \times X)$</td>
<td><code>lincom .b[.cons] + (.b[.t] * T)</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$+ (\beta_3 \times XT)$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Difference predicted versus counterfactual at time-point $T$</td>
<td>$(\beta_2 \times X) + (\beta_3 \times XT)$</td>
<td><code>lincom (.b[x(trp)] * X)</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>$+ (.b[,x_t(trp)] * XT)$</td>
<td></td>
</tr>
<tr>
<td>Single group—Multiple interventions (1)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Preintervention trend</td>
<td>$\beta_1$</td>
<td><code>lincom .b[.t]</code></td>
<td>In regression output</td>
</tr>
<tr>
<td>First-intervention trend</td>
<td>$\beta_1 + \beta_3$</td>
<td><code>lincom .b[.t] + .b[,x_t(trp)]</code></td>
<td><em>itsa posttrend</em> option</td>
</tr>
<tr>
<td>Second-intervention trend</td>
<td>$\beta_1 + \beta_3 + \beta_5$</td>
<td><code>lincom .b[.t] + .b[,x_t(trp)]</code></td>
<td><em>itsa posttrend</em> option</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$+ .b[,x_t(trp2)]$</td>
<td></td>
</tr>
<tr>
<td>Difference preintervention versus first-intervention trend</td>
<td>$\beta_3$</td>
<td><code>lincom .b[,x_t(trp)]</code></td>
<td>In regression output</td>
</tr>
<tr>
<td>Difference preintervention versus second-intervention trend</td>
<td>$\beta_3 + \beta_5$</td>
<td><code>lincom .b[,x_t(trp)] + .b[,x_t(trp2)]</code></td>
<td></td>
</tr>
<tr>
<td>Difference first- versus second-intervention trend</td>
<td>$\beta_5$</td>
<td><code>lincom .b[,x_t(trp2)]</code></td>
<td>In regression output</td>
</tr>
</tbody>
</table>

*Continued on next page*
### Measure of interest | Model parameters | \( \text{lincom specification (following itsa)} \) | Notes |
---|---|---|---|
**Multiple groups—Single intervention (2)**

| Preintervention trend: control | \( \beta_1 \) | \( \text{lincom} \ .b[.t] \) | In regression output |
| Preintervention trend: treatment | \( \beta_5 + \beta_1 \) | \( \text{lincom} \ .b[z.t] + \_b[.t] \) | |
| Difference preintervention: | \( \beta_5 \) | \( \text{lincom} \ .b[z.t] \) | In regression output |
| treatment versus control | | | |
| Postintervention trend: control | \( \beta_1 + \beta_3 \) | \( \text{lincom} \ .b[.t] + \_b[x.t(trp)] \) | \text{itsa posttrend option} |
| Postintervention trend: treatment | \( \beta_1 + \beta_3 + \beta_5 + \beta_7 \) | \( \text{lincom} \ .b[.t] + \_b[z.t] + \_b[x.t(trp)] \) | \text{itsa posttrend option} |
| | | + \_b[z.x.t(trp)] | |
| Difference postintervention: | \( \beta_5 + \beta_7 \) | \( \text{lincom} \ .b[z.t] + \_b[z.x.t(trp)] \) | \text{itsa posttrend option} |
| treatment versus control | | | |
| Difference preintervention versus | \( \beta_3 \) | \( \text{lincom} \ .b[x.t(trp)] \) | In regression output |
| postintervention: control | | | |
| Difference preintervention versus | \( \beta_3 + \beta_7 \) | \( \text{lincom} \ .b[z.t(trp)] + \_b[z.x.t(trp)] \) | |
| postintervention: treatment | | | |
| Difference preintervention versus | \( \beta_7 \) | \( \text{lincom} \ .b[x.z.t(trp)] \) | In regression output |
| postintervention: treatment | | | |
| versus control | | | |

**Multiple groups—Multiple interventions (2.2)**

| Preintervention trend: control | \( \beta_1 \) | \( \text{lincom} \ .b[.t] \) | In regression output |
| Preintervention trend: treatment | \( \beta_5 + \beta_1 \) | \( \text{lincom} \ .b[z.t] + \_b[.t] \) | |
| Difference preintervention: | \( \beta_5 \) | \( \text{lincom} \ .b[z.t] \) | In regression output |
| treatment versus control | | | |
| First-intervention trend: control | \( \beta_1 + \beta_3 \) | \( \text{lincom} \ .b[.t] + \_b[x.t(trp1)] \) | \text{itsa posttrend option} |
| First-intervention trend: treatment | \( \beta_1 + \beta_3 + \beta_5 + \beta_7 \) | \( \text{lincom} \ .b[.t] + \_b[z.t] + \_b[x.t(trp)] \) | \text{itsa posttrend option} |
| | | + \_b[z.x.t(trp)] + \_b[z.x.t(trp)] | |

*Continued on next page*
<table>
<thead>
<tr>
<th>Measure of interest</th>
<th>Model parameters</th>
<th>lincom specification (following itsa)</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Difference first intervention: treatment versus control</td>
<td>$\beta_5 + \beta_7$</td>
<td>lincom $.<em>b[z</em>{xt}] + <em>b[z</em>{xt}(trp1)]$</td>
<td>itsa posttrend option</td>
</tr>
<tr>
<td>Second-intervention trend: control</td>
<td>$\beta_1 + \beta_3 + \beta_9$</td>
<td>lincom $._b[t] + <em>b[x</em>{zt}(trp1)]$</td>
<td>itsa posttrend option</td>
</tr>
<tr>
<td>Second-intervention trend: treatment</td>
<td>$\beta_1 + \beta_3 + \beta_5 + \beta_7 + \beta_9 + \beta_{11}$</td>
<td>lincom $._b[t] + <em>b[x</em>{zt}] + <em>b[x</em>{xt}(trp1)] + <em>b[x</em>{xt}(trp2)]$</td>
<td>itsa posttrend option</td>
</tr>
<tr>
<td>Difference second intervention: treatment versus control</td>
<td>$\beta_5 + \beta_7 + \beta_{11}$</td>
<td>lincom $.<em>b[z</em>{xt}] + <em>b[z</em>{xt}(trp1)] + <em>b[z</em>{xt}(trp2)]$</td>
<td>itsa posttrend option</td>
</tr>
<tr>
<td>Difference preintervention versus first intervention: control</td>
<td>$\beta_3$</td>
<td>lincom $.<em>b[x</em>{zt}(trp1)]$</td>
<td>In regression output</td>
</tr>
<tr>
<td>Difference preintervention versus first intervention: treatment</td>
<td>$\beta_3 + \beta_7$</td>
<td>lincom $.<em>b[x</em>{zt}(trp1)] + <em>b[x</em>{zt}(trp2)]$</td>
<td>In regression output</td>
</tr>
<tr>
<td>Difference preintervention versus first intervention: treatment versus control</td>
<td>$\beta_7$</td>
<td>lincom $.<em>b[x</em>{zt}(trp1)]$</td>
<td>In regression output</td>
</tr>
<tr>
<td>Difference first versus second intervention: control</td>
<td>$\beta_9$</td>
<td>lincom $.<em>b[x</em>{zt}(trp2)]$</td>
<td>In regression output</td>
</tr>
<tr>
<td>Difference first versus second intervention: treatment</td>
<td>$\beta_9 + \beta_{11}$</td>
<td>lincom $.<em>b[x</em>{zt}(trp2)] + <em>b[x</em>{zt}(trp2)]$</td>
<td>In regression output</td>
</tr>
<tr>
<td>Difference first versus second intervention: treatment versus control</td>
<td>$\beta_{11}$</td>
<td>lincom $.<em>b[x</em>{zt}(trp2)]$</td>
<td>In regression output</td>
</tr>
<tr>
<td>Difference preintervention versus second intervention: control</td>
<td>$\beta_3 + \beta_9$</td>
<td>lincom $.<em>b[x</em>{zt}(trp1)] + <em>b[x</em>{zt}(trp2)]$</td>
<td>In regression output</td>
</tr>
</tbody>
</table>

Continued on next page
<table>
<thead>
<tr>
<th>Measure of interest</th>
<th>Model parameters</th>
<th>\texttt{lincom} specification (following \texttt{itsa})</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Difference preintervention</td>
<td>$\beta_3 + \beta_5 + \beta_7 + \beta_9$</td>
<td>\texttt{lincom } \texttt{.b[z.x.t(trp1)] + .b[z.x.t]}</td>
<td></td>
</tr>
<tr>
<td>versus second intervention:</td>
<td>$+ \beta_{11}$</td>
<td>$+ .b[z.x.t(trp1)] + .b[z.x.t(trp2)]$</td>
<td></td>
</tr>
<tr>
<td>treatment</td>
<td></td>
<td>$+ .b[z.x.t(trp2)]$</td>
<td></td>
</tr>
<tr>
<td>Difference preintervention</td>
<td>$\beta_5 + \beta_7 + \beta_{11}$</td>
<td>\texttt{lincom } \texttt{.b[z.t]} + \texttt{.b[z.x.t(trp1)]}</td>
<td></td>
</tr>
<tr>
<td>versus control</td>
<td></td>
<td>$+ .b[z.x.t(trp2)]$</td>
<td></td>
</tr>
</tbody>
</table>

Note: \textit{trp} = treatment period
4 Example

To demonstrate how the measures of interest are calculated following the `itsa` command, we use the example data presented in Linden (2015). Briefly, in 1988, California passed the voter-initiative Proposition 99, which was a widespread effort to reduce smoking rates by raising the cigarette excise tax by 25 cents per pack and to fund antismoking campaigns and other related activities throughout the state (for a comprehensive discussion of this initiative, see Abadie, Diamond, and Hainmueller (2010)). The dependent variable for our analyses is the per capita cigarette sales (in packs), for the years 1970 to 2000. California’s actual intervention period commenced in 1989, and we use 38 states with no such intervention in place during those years as controls.

For illustration, we derive the various measures of interest using the most complex scenario: the multiple-group, multiple-intervention analysis (2.2). In this example, we use `itsa` to assess the impact of Proposition 99 in reducing California’s per capita cigarette sales (in packs), compared with that of the other 38 states in the data file. However, for exposition, we add a fictitious intervention to the cigarette sales data, starting in 1982 (in addition to the actual intervention in 1989).

First, we load the data and declare the dataset as panel:

```
. use cigsales
. tsset state year
    panel variable: state (strongly balanced)
    time variable: year, 1970 to 2000
    delta: 1 unit
```

Next, we specify a multiple-group ITSA with California (state number 3 in the study) as the treatment group and, by default, all other 38 states as controls. Additionally, we specify 1982 and 1989 as the start of the interventions and request postintervention trend estimates and a figure. We fit the model using `newey` with one lag:

```bash
```

**A. Linden**

```plaintext
. itsa cigsale, treat(3) trperiod(1982 1989) lag(1) posttrend figure
panel variable: state (strongly balanced)
time variable: year, 1970 to 2000
delta: 1 unit

Regression with Newey-West standard errors
Number of obs = 1,209
maximum lag: 1
F( 11, 1197) = 405.05
Prob > F = 0.0000

<table>
<thead>
<tr>
<th></th>
<th>Newey-West</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Coef. Std. Err.   t   P&gt;</td>
</tr>
<tr>
<td>_t</td>
<td>1.592952         .5958503   2.67  0.008       0.4239245  2.761979</td>
</tr>
<tr>
<td>_z</td>
<td>-.6469976        4.577683   -0.14  0.888      -9.628173   8.334177</td>
</tr>
<tr>
<td>_z_t</td>
<td>-1.796798        .6679163   -2.69  0.007       -3.107215  -.486381</td>
</tr>
<tr>
<td>_x1982</td>
<td>-10.24134        5.43969    -1.88  0.060       -20.91373  .4310467</td>
</tr>
<tr>
<td>_x_t1982</td>
<td>-5.150358        1.95146    -2.69  0.007       -8.475172  -1.825543</td>
</tr>
<tr>
<td>_x1982</td>
<td>2.20087          6.02712   0.37  0.715       -9.624026  14.02576</td>
</tr>
<tr>
<td>_x_t1982</td>
<td>1.491346         1.243095   1.20  0.230       -.947542   3.930234</td>
</tr>
<tr>
<td>_x1989</td>
<td>1.049573         4.753687   -0.21  0.834       -11.27506  13.377912</td>
</tr>
<tr>
<td>_x_t1989</td>
<td>2.486127         1.108565   2.24  0.025       .3111804   4.661074</td>
</tr>
<tr>
<td>_x1989</td>
<td>-7.336589        5.285895  -1.39  0.166      -17.70724  3.034061</td>
</tr>
<tr>
<td>_x_t1989</td>
<td>-1.917395        1.144526  -1.68  0.094      -4.162896  .3281052</td>
</tr>
<tr>
<td>_cons</td>
<td>126.6765         4.206786   29.87  0.000     117.423     133.93</td>
</tr>
</tbody>
</table>

Comparison of Linear Postintervention Trends: 1982

Treated : _b[_t] + _b[_z_t] + _b[_x_t1982] + _b[_z_x_t1982]
Controls : _b[_t] + _b[_x_t1982]
Difference : _b[_z_t] + _b[_z_x_t1982]

<table>
<thead>
<tr>
<th></th>
<th>Linear Trend</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Coeff       Std. Err.   t   P&gt;</td>
</tr>
<tr>
<td>Treated</td>
<td>-3.8429      0.2019       -19.0309  0.0000     -4.2390    -3.4467</td>
</tr>
<tr>
<td>Controls</td>
<td>-3.5374      0.9930       -3.5622    0.0004     -5.4857   -1.5891</td>
</tr>
<tr>
<td>Difference</td>
<td>-0.3055      1.0134       -0.3014    0.7631     -2.2936   1.6827</td>
</tr>
</tbody>
</table>

Comparison of Linear Postintervention Trends: 1989

Treated : _b[_t] + _b[_z_t] + _b[_x_t1982] + _b[_z_x_t1982] + _b[_x_t1989] + _b[_z_x_t1989]
Controls : _b[_t] + _b[_x_t1982] + _b[_x_t1989]
Difference : _b[_z_t] + _b[_z_x_t1982] + _b[_z_x_t1989]

<table>
<thead>
<tr>
<th></th>
<th>Linear Trend</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Coeff       Std. Err.   t   P&gt;</td>
</tr>
<tr>
<td>Treated</td>
<td>-3.2741      0.2521       -12.9865   0.0000     -3.7688   -2.7795</td>
</tr>
<tr>
<td>Controls</td>
<td>-1.0513      0.4259       -2.4686    0.0137     -1.8868  -0.2158</td>
</tr>
<tr>
<td>Difference</td>
<td>-2.2228      0.4949       -4.4916    0.0000     -3.1938  -1.2519</td>
</tr>
</tbody>
</table>
```
A comprehensive set of postestimation measures

Figure 1. Multiple-group ITSA with 2 intervention periods, using Newey–West standard errors and 1 lag; all 38 “nontreated” states are used for comparison.

As part of the regression output or the posttrend option, itsa provides 12 of the measures of interest. However, six important measures are not inherently calculated (see table 1). Table 2 provides a composite of all measures (both provided by itsa and calculated using lincom) following estimation of the ITSA model above.
Table 2. *lincom* estimates for all measures of interest following *itsa* estimation of a multiple-group, multiple-intervention design

<table>
<thead>
<tr>
<th>Measure of interest</th>
<th><em>lincom</em> specification (following <em>itsa</em>)</th>
<th>Point estimate [95% CI]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preintervention trend: control</td>
<td><em>lincom</em> <em>b</em>[z.t]</td>
<td>1.593 [0.424, 2.762]</td>
</tr>
<tr>
<td>Preintervention trend: treatment</td>
<td><em>lincom</em> <em>b</em> [z.t] + <em>b</em> [t]</td>
<td>−0.204 [−0.796, 0.388]</td>
</tr>
<tr>
<td>Difference preintervention: treatment versus control</td>
<td><em>lincom</em> <em>b</em> [z.t]</td>
<td>−1.797 [−3.107, −0.486]</td>
</tr>
<tr>
<td>First-intervention trend: control</td>
<td><em>lincom</em> <em>b</em> [z.t] + <em>b</em> [x_t1982] + <em>b</em> [x_t1982]</td>
<td>−3.537 [−5.486, −1.589]</td>
</tr>
<tr>
<td>Difference first intervention: treatment versus control</td>
<td><em>lincom</em> <em>b</em> [z.t] + <em>b</em> [x_t1982] + <em>b</em> [x_t1982]</td>
<td>−0.305 [−2.293, 1.683]</td>
</tr>
<tr>
<td>Second-intervention trend: control</td>
<td><em>lincom</em> <em>b</em> [z.t] + <em>b</em> [x_t1982] + <em>b</em> [x_t1982]</td>
<td>−1.051 [−1.887, −0.216]</td>
</tr>
<tr>
<td>Difference preintervention versus first intervention: control</td>
<td><em>lincom</em> <em>b</em> [x_t1982]</td>
<td>−5.130 [−7.475, −2.786]</td>
</tr>
<tr>
<td>Difference preintervention versus first intervention: treatment versus control</td>
<td><em>lincom</em> <em>b</em> [x_t1982] + <em>b</em> [x_t1982]</td>
<td>1.491 [−0.947, 3.390]</td>
</tr>
<tr>
<td>Difference first versus second intervention: control</td>
<td><em>lincom</em> <em>b</em> [x_t1989]</td>
<td>2.486 [0.311, 4.661]</td>
</tr>
<tr>
<td>Difference first versus second intervention: treatment</td>
<td><em>lincom</em> <em>b</em> [x_t1989] + <em>b</em> [x_t1989]</td>
<td>0.569 [0.010, 1.127]</td>
</tr>
</tbody>
</table>
A comprehensive set of postestimation measures

It is helpful to visually inspect the figure to interpret the related estimates provided in table 2. For example, we see in table 2 that the preintervention trend in annual cigarette sales per capita for the control states appears to be increasing, while the preintervention trend in sales for California appears to be decreasing. Table 2 confirms this by revealing that the control states had an annual increase in cigarette sales of 1.593 (95% CI: [0.424, 2.762]) packs per capita between 1970 and 1982, while California had an annual decrease in cigarette sales of −3.390 (95% CI: [−5.802, −0.978]) packs per capita over the same period. The difference in trends between California and the control states for the preintervention period was thus −1.797 (95% CI: [−3.107, −0.486]). All other estimates are interpreted accordingly.

5 Discussion

Interrupted time-series designs offer a reasonably robust quasi-experimental approach for evaluating treatment effects when studying a single unit, the outcome variable is serially ordered as a time series, and multiple observations are captured in both the preintervention and postintervention periods. ITSA is gaining in popularity across many disciplines and was recently added as an acceptable primary research design for inclusion in systematic reviews of the literature (Cochrane Effective Practice and Organisation of Care [EPOC] Group 2015).

This article fills a gap in the current ITSA literature by providing the calculations and Stata code for a comprehensive set of measures of interest that investigators may require after ITSA estimation using the itsa command or after a longitudinal model when multiple treatment units are available (Linden and Adams 2010). Further, understanding how to derive measures that capture the components of ITSA may serve as a helpful framework for understanding similar components of more complex models, such as arima, and for exploring alternative estimation using bootstrapping techniques (Politis 2003).

6 References


Cochrane Effective Practice and Organisation of Care (EPOC) Group. 2015. EPOC-specific resources for review authors. Oslo: Norwegian Knowledge Centre for the Health Services. http://epoc.cochrane.org/epoc-specific-resources-review-authors.


A comprehensive set of postestimation measures


About the author

Ariel Linden is a health services researcher specializing in the evaluation of healthcare interventions. He is both an independent consultant and a research scientist in the division of general medicine at the University of Michigan’s medical school.
Within- and between-cluster effects in generalized linear mixed models: A discussion of approaches and the xthybrid command

Reinhard Schunck  
GESIS—Leibniz-Institute for the Social Sciences  
Cologne, Germany  
reinhard.schunck@gesis.de

Francisco Perales  
Institute for Social Science Research  
University of Queensland  
Brisbane, Australia  
f.perales@uq.edu.au

Abstract. One typically analyzes clustered data using random- or fixed-effects models. Fixed-effects models allow consistent estimation of the effects of level-one variables, even if there is unobserved heterogeneity at level two. However, these models cannot estimate the effects of level-two variables. Hybrid and correlated random-effects models are flexible modeling specifications that separate within- and between-cluster effects and allow for both consistent estimation of level-one effects and inclusion of level-two variables. In this article, we elaborate on the separation of within- and between-cluster effects in generalized linear mixed models. These models present a unifying framework for an entire class of models whose response variables follow a distribution from the exponential family (for example, linear, logit, probit, ordered probit and logit, Poisson, and negative binomial models). We introduce the user-written command xthybrid, a shell for the meglm command. xthybrid can fit a variety of hybrid and correlated random-effects models.

Keywords: st0468, xthybrid, correlated random effects, fixed effects, generalized linear mixed models, hybrid model, meglm, Mundlak model, random effects

1 Introduction

Researchers undertaking multilevel and panel analysis of hierarchically clustered data often face a difficult decision between random- and fixed-effects models. Random-effects models allow researchers to estimate the effect of cluster-invariant variables (that is, level-two variables) on the outcome variable but impose the assumption that the random effects (for example, the level-two error) are uncorrelated with the observed covariates. If this assumption is violated, the model coefficients are biased. Fixed-effects models, on the other hand, do not require this assumption—and can provide unbiased estimates of the level-one variables, even if there is unobserved heterogeneity.
Within- and between-cluster effects in GLMM

at the cluster level. However, fixed-effects model estimation relies only on within-cluster variation in the explanatory and outcomes variables; thus these models cannot provide effect estimates for the level-two variables.¹

More flexible modeling specifications provide fixed-effects estimates (or estimates that are close to these) for level-one variables and allow inclusion of level-two variables, most notably the hybrid (Allison 2009) and correlated random-effects models (Wooldridge 2010). The latter is also known as the Mundlak model (Baltagi 2006; Mundlak 1978). These estimation strategies differentiate within- and between-cluster effects and combine the strengths of random- and fixed-effects models. In the linear case, they yield estimates of the level-one covariates that are unbiased by cluster-level unobserved heterogeneity, while allowing for level-two cluster-invariant covariates (Allison 2009; Mundlak 1978; Neuhaus and Kalbfleisch 1998; Rabe-Hesketh and Skrondal 2012; Raudenbush 1989; Schunck 2013; Snijders and Berkhof 2008).

Schunck (2013) described using these models for continuous outcome variables, providing a theoretical overview and a practical application in Stata. This article discusses the applicability of hybrid and correlated random-effects models within the umbrella of generalized linear mixed models (GLMM) (Brumback et al. 2010). In doing so, we show how the decomposition of within- and between-cluster effects can be extended to GLMM, which comprise popular models for binary, ordered, and count outcomes (Neuhaus and Kalbfleisch 1998; Neuhaus and McCulloch 2006; Brumback et al. 2010). Importantly, such decomposition can approximate fixed-effects estimates for specifications in which a fixed-effects estimator is not available or implemented (Neuhaus and McCulloch 2006).

In the remainder of the article, we first elaborate on the separation of within- and between-cluster effects in a GLMM framework, then present a user-written command, xthybrid, that builds on Stata’s meglm command and can fit hybrid and correlated random-effects models.

¹ The terms “fixed effects” and “random effects” are not used consistently across disciplines and literature. In the multilevel model literature, the term “fixed effects” denotes a model’s regression coefficients, whereas the term “random effects” refers to a model’s random intercepts and slopes. In this article, random-effects models refer to models for clustered data that have both random effects and fixed effects (also known as multilevel models, hierarchical models, and mixed models). In this context, a fixed-effects model refers to a model that includes only fixed effects, which is typically a pooled or cross-sectional model that does not consider that the data may be clustered. In econometric literature, however, the term “fixed effects model” refers to a model for clustered data that allows for arbitrary dependence between the unobserved effects and the covariates (Wooldridge 2010, 286). The name “fixed-effects model” emerged because these models treat the unobserved cluster-level effects as fixed rather than random (McCulloch, Searle, and Neuhaus 2008). Whether these effects are random or nonrandom is of concern to us. Modern econometrics assumes they are random (Wooldridge 2010, 286). In this article, we adopt the econometric terminology for fixed-effects models.
2 GLMM

Generalized linear models (GLM) constitute a unifying framework for an entire class of models whose response variables follow a distribution from the exponential family. This includes many popular models such as the standard linear model, models for binary responses (for example, logit and probit models), models for ordinal responses (for example, ordered probit and logit models), and models for count responses (for example, Poisson and negative binomial models). In this section, we will provide a short overview of GLM and GLMM (for details, see McCulloch, Searle, and Neuhaus [2008], Skrondal and Rabe-Hesketh [2003], and Rabe-Hesketh and Skrondal [2012]).

Let \( \mu_j \) be the expected response of \( y_j \) given the covariates \( \{ \mu_j = E(y_j|x_j) \} \). Then, a GLM with \( y_j \) as the response variable and \( x_j \) as a covariate is defined as

\[
g\{E(y_j|x_j)\} = g(\mu_j) = \beta x_j
\]

\( g(\cdot) \) is the so-called link function, which transforms the mean \( \mu_j \) so that it can be linearly related to the predictors. The link function therefore defines the functional relationship between the predictors and the response variable (McCullagh and Nelder 1989; McCulloch, Searle, and Neuhaus 2008; Rabe-Hesketh, Skrondal, and Pickles 2004; Skrondal and Rabe-Hesketh 2003). Specifying a GLM also requires choosing a conditional distribution for the response variable from the exponential family of distributions. Different permutations of link functions and distributions result in different models (see table 1).

GLMs can be extended to include random effects and are thus suited for analyzing clustered data, such as multilevel and panel data. These models are known as generalized linear mixed models (GLMM). Consider a situation where we have data with two hierarchical levels. Let \( i \) denote level two (for example, schools) and \( j \) denote level one (for example, students). \( y_{ij} \) is the response (dependent) variable, \( x_{ij} \) is a level-one variable that varies within and between clusters, \( c_i \) is a level-two variable that varies only between clusters, and \( u_i \) is the random intercept. A GLMM is specified as

\[
g\{E(y_{ij}|x_{ij}, c_i, u_i)\} = g(\mu_{ij}) = \beta x_{ij} + \gamma c_i + u_i
\]

(1)

The “mixing” outlined above becomes obvious: this model “mixes” a fixed part (the fixed coefficients \( \beta \) and \( \gamma \)) and a random part (the random intercept \( u_i \)). To relax the assumption that the effects of level-one covariates are the same across all clusters, we can include random slopes as follows:

\[
g(\mu_{ij}) = (\beta + u_{i2})x_{ij} + \gamma c_i + u_{i1}
\]

In Stata, the available link functions for GLMM comprise identity, logit, probit, log, and complementary log-log. The available distributions from the exponential family of distributions comprise the normal (Gaussian), Bernoulli, binomial, gamma, negative binomial, ordinal, and Poisson distributions (see table 1). Other link functions and distributions are theoretically possible.
Within- and between-cluster effects in GLMM

Table 1. Possible combinations of link functions and distributions in \texttt{xthybrid}

<table>
<thead>
<tr>
<th>Distribution</th>
<th>Identity</th>
<th>Log</th>
<th>Logit</th>
<th>Probit</th>
<th>Cloglog</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gaussian</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bernoulli</td>
<td></td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>Binomial</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Gamma</td>
<td></td>
<td>x</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Negative binomial</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ordinal</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Poisson</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Using the identity link $g(\mu_{ij}) = \mu_{ij}$ and the Gaussian distribution for $y_{ij}$ yields a linear random-intercept model,

$$\mu_{ij} = \beta x_{ij} + \gamma c_i + u_i$$  

(2)

where the conditional distribution of $y_{ij}$ is $y_{ij}|x_{ij}, c_i, u_i \sim N(\mu_{ij}, \sigma^2)$. If the outcome variable is binary, the expected value for $y_{ij}$ is the probability that $y_{ij} = 1$; that is, $\mu_{ij} = \Pr(y_{ij} = 1|x_{ij}, c_i, u_i)$. Combining a Bernoulli or a binomial distribution for the response variable with a probit link results in the random-intercept probit model

$$\Phi^{-1}(\mu_{ij}) = \beta x_{ij} + \gamma c_i + u_i$$

where $\Phi(\cdot)^{-1}$ is the inverse function of the standard normal cumulative distribution. Here the conditional distribution of $y_{ij}$ is $y_{ij}|x_{ij}, c_i, u_i \sim B(1, \pi_{ij})$. We specify a random-effects logit model by choosing the logit link:

$$\logit(\mu_{ij}) = \beta x_{ij} + \gamma c_i + u_i$$

This rationale can extend to other, more complex models, for example, models for ordered and count outcomes (for an overview, see McCulloch, Searle, and Neuhaus [2008]; Skrondal and Rabe-Hesketh [2003, 2004]).

3 Level-two confounders (unobserved heterogeneity)

The standard assumptions in GLMM (that is, in multilevel models) are that i) the level-two error is a normally distributed random variable—that is, $u_i \sim N(0, \sigma^2_u)$; and ii) the level-two error is uncorrelated with the covariates—that is, $E(u_i|x_{ij}, c_i) = 0$. The latter assumption is of particular importance and means that there are no omitted level-two confounders; that is, there is no unobserved heterogeneity at level two. If the level-two error is correlated with the covariates so that $E(u_i|x_{ij}, c_i) \neq 0$, then the effect estimates in a random-effects model, for example, as in (1), will be biased. This situation emerges if we omit a confounding variable at level two from the model.
Alternatively, we can treat unobserved level-two effects as fixed effects. In linear models, a simple way of accomplishing this is to directly estimate the fixed effects by including a dummy variable for each of the clusters:

$$y_{ij} = \beta_{LSDV} x_{ij} + \sum_{i=1}^{i} \beta_i k_i + \epsilon_{ij}$$

This model is the least-squares dummy variable (LSDV) estimator (Wooldridge 2010). It fits $i$ intercepts—one for each cluster—represented by the variables $k_i$. This approach provides consistent estimation of the level-one covariates without the assumptions of the random-effects model that the cluster-specific intercepts are random variables and uncorrelated with the covariates. Instead, they are explicitly included in the model and estimated as fixed effects. Thus the estimated level-one effects will be unbiased by level-two unobserved heterogeneity, because there is none anymore. Note that estimates may still be biased because of unobserved heterogeneity at level one. The models still assume that $E(\epsilon_{ij} | x_{ij}, k_i) = 0$. $\epsilon_{ij}$ is the level-one error, which we will treat as white noise for the remainder of this article.

A disadvantage of the LSDV model is that we cannot retrieve the effect of level-two variables ($c_i$). Because the model incorporates dummy variables capturing the overall cluster effects, we cannot identify the effects of cluster-level covariates because of collinearity. Additionally, estimating each cluster effect is impractical when there is a large number of clusters. Furthermore, using maximum likelihood to fit these models leads to inconsistent parameter estimates. This is because of the incidental parameters problem; the number of fixed effects parameters (or “nuisance parameters”) increases with sample size, which leads to inconsistent estimates when using maximum likelihood estimation (Andersen 1970; Chamberlain 1980; Wooldridge 2010). This makes it infeasible to use this approach for models estimated using maximum likelihood—including GLMM.

An alternative approach to fitting a fixed-effects (FE) model in the linear case is demeaning the explanatory and outcome variables. We do this by subtracting the between model

$$\bar{y}_i = \beta \bar{x}_i + \gamma c_i + u_i + \tau_i$$

from the random-effects model

$$y_{ij} = \beta x_{ij} + \gamma c_i + u_i + \epsilon_{ij}$$

Note that (4) is equivalent to (2.2). Because $\tau_i = c_i$ and $\bar{u}_i = u_i$, the subtraction leads to

$$(y_{ij} - \bar{y}_i) = \beta_{FE} (x_{ij} - \bar{x}_i) + (\epsilon_{ij} - \tau_i)$$

This technique, also called the “within transformation”, averages out all elements in (5) that do not vary within clusters, including the level-two error term, $u_i$. Thus this FE model does not require any assumptions on the distribution of $u_i$ or its correlation with the covariates. The estimated effects of $\beta$ in the LSDV model in (4) and the
Within- and between-cluster effects in GLMM

FE model in (5) are identical ($\beta_{LSDV} = \beta_{FE}$). Thus the fixed-effects model in its demeaned form (5) also provides estimates of level-one covariates, which are unbiased by unobserved heterogeneity at level two. Because any level-two characteristic—observed or unobserved—is removed from the equation, we cannot retrieve the effects of level-two covariates, just as in the LSDV model (4).

The above refers only to the linear case. For some GLMM that are estimated with maximum likelihood (for example, logit models), we can compute a similar fixed-effects estimator using a conditional likelihood approach (Chamberlain 1980; McCulloch, Searle, and Neuhaus 2008; Wooldridge 2010). The conditional likelihood approach uses a sufficient statistic to remove the level-two error from the equation (Chamberlain 1980; McCulloch, Searle, and Neuhaus 2008; Wooldridge 2010). The conditional likelihood approach can thus forgo any assumptions on the level-two error. Therefore, conditional likelihood models are also referred to as fixed-effects models. However, conditional likelihood approaches are unavailable for most GLMM, so this approach is not always a viable alternative (McCulloch, Searle, and Neuhaus 2008; Wooldridge 2010).

Altogether, fixed-effects models provide less biased estimates of the level-one covariates than random-effects models, but unlike random-effects models, they fail to retrieve the effect estimates of level-two variables. Depending on the nature of the research question, disregarding between-cluster variation can even be seen as an advantage. For example, this is sometimes regarded as focusing on the informative cases (Halaby 2004; Wooldridge 2010, 621f.). This holds in particular for longitudinal research investigating how change in an explanatory variable, $x_{ij}$, is associated with change in an outcome variable $y_{ij}$. However, failure to retrieve the effect estimates of level-two variables is a major problem in multilevel analysis, where the interest often lies in these effects, for example, how the characteristics of neighborhoods, schools, workplaces, or geographical areas influence individuals’ outcomes (Sampson 2003; Sampson, Raudenbush, and Earls 1997). Because the fixed-effects approach discards all contextual (level-two) information, some argue that it is generally less preferable than the random-effects approach for multilevel analysis (Bell and Jones 2015).

4 Within- and between-decomposition

An alternative to both random and fixed-effects models within the framework of GLMM is models that separate within- and between-cluster effects (Neuhaus and McCulloch 2006; Schunck 2013), such as the hybrid model (Allison 2009) and the related correlated random-effects model (Cameron and Trivedi 2005; Wooldridge 2010).

The hybrid model (Allison 2009) splits within- and between-cluster effects for the level-one covariates:

$$g(\mu_{ij}) = \beta_W(x_{ij} - \bar{x}_i) + \beta_B \bar{x}_i + \gamma c_i + u_i$$

(6)

2. To be precise, both (4) and (5) are fixed-effects models. Thus we could label both estimators of $\beta$ with the subscript FE.
This is accomplished by including both the deviation from the cluster-specific mean \((x_{ij} - \bar{x}_i)\) and the cluster-specific mean \(\bar{x}_i\) among the model covariates. \(\beta_W\) gives the within-cluster effect, and \(\beta_B\) gives the between-cluster effect.

The correlated random-effects model (Wooldridge 2010), sometimes called the Mundlak (1978) model, is mathematically equivalent to the hybrid model. However, in contrast to (6), it includes the level-one variable \((x_{ij})\) in its undemeaned form:

\[
g(\mu_{ij}) = \beta_W x_{ij} + \tau \bar{x}_i + \gamma c_i + v_i
\]  

In contrast to the standard random-intercept model, it introduces the assumption that the level-two error \(u_i = \tau \bar{x}_i + v_i\) and \(v_i \sim N(0, \sigma^2_v)\). This means that the level-two error can depend on \(x_{ij}\) through its cluster means. The inclusion of \(\bar{x}_i\) picks up any correlation between this variable and the unobserved random effect. In this model, \(\tau = \beta_B - \beta_W\). The relationship between the two model specifications is apparent when rewriting (6) as

\[
g(\mu_{ij}) = \beta_W x_{ij} + (\beta_B - \beta_W) \bar{x}_i + \gamma c_i + u_i
\]

### 4.1 Within-cluster and between-cluster effects

The basic idea behind the hybrid and correlated random-effects models is to restrict the dependency between \(u_i\) and the level-one covariates. We can accomplish this by assuming that \(u_i\) depends on the mean values of the level-one covariates \((\bar{x}_i)\). We estimate the within-cluster effect, \(\beta_W\), using only within-cluster variation. It assesses how on average a within-cluster change in \(x_{ij}\) is associated with a within-cluster change in \(y_{ij}\). In the linear case (that is, with the identity link and the Gaussian distribution), the within-cluster effects fit by both the hybrid and correlated random-effects models are identical to the fixed-effects estimates, so that \(\beta_W = \beta_{FE}\) (Goetgeluk and Vansteelandt 2008; Hsiao 2003; Mundlak 1978).

The between-cluster effect, \(\beta_B\), assesses how a change in \(\bar{x}_i\) is associated with a change in \(\bar{y}_i\). It is estimated using only between-cluster variation. Different research traditions have different views as to whether these effects are informative. In multilevel research, the interest often lies on the level-two variables, and between-cluster effects of the level-one variables are level-two effects. The substantive interpretation of the cluster means of the level-one variables is different from the substantive interpretation of the level-one variables (Snijders and Berkhof 2008, 146). However, one should keep in mind that if the random-effects assumption is violated (that is, there is unobserved heterogeneity at level two), the between-cluster effect is biased. In panel-data analysis, the interest lies chiefly on the within-cluster effects, so it is imperative to obtain estimates of these effects, that are robust to unobserved heterogeneity at level two (Allison 2009; Halaby 2004; Wooldridge 2010). From this perspective, it is questionable whether the between-cluster effects are of substantial interest at all.

Regardless of whether one is interested in interpreting the between-cluster effect, one should include the means of the level-one variables as controls for the other level-two variables. If one includes other level-two variables, for example, \(c_i\), without controlling
Within- and between-cluster effects in GLMM

for $\pi_i$, then the estimated effect of $c_i$ is not adjusted for between-cluster differences in $x_{ij}$. Note that consistent estimation of the effects of level-two variables still rests on the assumption that there is no correlation between these and the level-two error $\{E(u_i|c_i) = 0\}$.

In both multilevel and panel-data models, it is helpful to compare between- and within-cluster effects for pragmatic reasons (Allison 2009; Schunck 2013). The random-effects model (1) assumes that both effects are the same and uses a weighted average of within- and between-cluster variation in estimation. However, if this assumption does not hold, the estimates of the random-effects model are biased. The comparison of within- and between-cluster effects is in fact a regression-based alternative to the Hausman specification test (Baltagi 2013, 76–77). In the hybrid model (6), one can test whether $\beta_W = \beta_B$ using a Wald test. In the correlated random-effects model (7), one can test whether $\tau = 0$. Note that both tests are mathematically equivalent and yield the same test statistics. This is because $\tau = \beta_B - \beta_W$. If the between-cluster effect $\beta_B$ and the within-cluster effect $\beta_W$ are not statistically significantly different from each other (which implies that $\tau = 0$), this suggests that $\beta_W = \beta_B = \beta$. In this case, (6) and (7) simplify to (1), the standard random-intercept model. Substantively, this means that the random-effects model’s assumption of a zero correlation between the level-two error and the level-one covariates holds. In contrast to the Hausman test, this test can also be used when we estimate (cluster) robust standard errors (SEs). Furthermore, it also works if the difference of the covariance matrices in the Hausman test is not positive definite. An additional advantage of this test is that it works at the level of individual variables. Thus one can use the more efficient random-effects estimate for those variables for which within and between effects do not differ significantly and retain both within and between effects for those variables for which they are significantly different.

4.2 Within-cluster effects in nonlinear models and nonlinear dependencies

Equivalence between effects estimates from standard fixed-effects models and effects estimates from hybrid or correlated random-effects models holds only in the linear case. In nonlinear models, the estimated within-cluster effects are typically similar, though not identical to the fixed-effects estimates, that is, the conditional likelihood estimates. In contrast to conditional likelihood approaches, which use a sufficient statistic to condition the cluster-level effects away (McCulloch, Searle, and Neuhaus 2008, 295), hybrid and correlated random-effects models take a parametric approach to the unobserved heterogeneity problem at level two (Wooldridge 2010, 286), placing certain restrictions on the conditional distribution of such heterogeneity, given the level-one covariates. While the conditional likelihood approach does not require any distributional assumption on the level-two error and its correlation with the covariates, the hybrid and correlated random-effects models assume that $u_i$ depends on the mean values of $x_{ij}$.

Differences in effect estimates of nonlinear hybrid and correlated random-effects models and standard fixed-effects models can emerge because of a violation of this
assumption, that is, if the level-two error and the independent variables at level one are not completely but only linearly uncorrelated. Failure to meet this assumption may result in biased estimates (Brumback et al. 2010; Brumback, Dailey, and Zheng 2012). However, note that this assumption is still less restrictive than the assumption of complete independence between the level-two error and the level-one variables of (1).

If we implement a conditional likelihood approach for a model belonging to the family of GLMM, we can easily compare the estimates from (6) and (7) against actual fixed-effects estimates. For example, this is possible for logit models. Unfortunately, conditional likelihood approaches are not available for many GLMM. Differences between fixed-effects estimates and within-cluster estimates from hybrid and correlated random-effects models may suggest that $u_i$ depends on $x_{ij}$ through functional forms other than the cluster means of the cluster-varying covariates (Brumback et al. 2010, 1652). Given enough observations within clusters, one can explicitly model other dependencies. For instance, one can do so by adding polynomial functions of the cluster means of the level-one covariates to the model (Allison 2014). A correlated random-effects model with additional quadratic and cubic terms is given by

$$g(\mu_{ij}) = \beta_W x_{ij} + \tau \bar{x}_i + \delta \bar{x}_i^2 + \eta \bar{x}_i^3 + \gamma c_i + v_i \quad (8)$$

Here we assume that $u_i = \tau \bar{x}_i + \delta \bar{x}_i^2 + \eta \bar{x}_i^3 + v_i$ and $v_i \sim N(0, \sigma^2_v)$. Just as in (7), a test of $\tau = 0, \delta = 0,$ and $\eta = 0$ can serve as inference regarding dependencies between $u_i$ and $x_{ij}$. If the estimates of $\delta$ and $\eta$ are not statistically significant, we can take this as evidence that the assumption of the correlated random-effects model is not violated. The equivalent hybrid model is

$$g(\mu_{ij}) = \beta_W (\bar{x}_i - x_{ij}) + \beta_B x_{ij} + \delta \bar{x}_i^2 + \eta \bar{x}_i^3 + \gamma c_i + v_i \quad (9)$$

Note that the estimated effects of the nonlinear dependencies are the same for both (8) and (9). In the hybrid model, however, inference regarding dependencies between $u_i$ and $x_{ij}$ requires a test of $\beta_W = \beta_B, \delta = 0,$ and $\eta = 0$. Thus the only difference between a correlated random-effects model with nonlinear dependencies and a hybrid model with nonlinear dependencies lies in $\tau$.

One may be tempted to compare the estimated within-cluster effects ($\beta_W$) from (6) or (7) with the effect estimates obtained by analogous models, including nonlinear dependencies. If $\beta_W$ does not differ substantially when we account for additional nonlinear dependencies, we could take this as evidence that the assumption of the hybrid and correlated random-effects models holds. However, such a comparison is complicated by the fact that estimates in nonlinear models with fixed error variance at level one are not directly comparable because of the “rescaling problem” (Allison 1999; Kohler, Karlson, and Holm 2011). This extends to comparisons of $\beta_W$ from (6) and $\beta_W$ from (9) and of $\beta_W$ from (7) and $\beta_W$ from (8). Including additional nonlinear dependencies will affect the estimate of $\beta_W$, even if these are orthogonal to $x_{ij}$ because of the rescaling problem. Consequently, $\beta_W$ from (6) cannot be identical to $\beta_W$ from

---

3. See also Chamberlain (1982).
Within- and between-cluster effects in GLMM

(9), and the same applies to the $\beta_W$ estimates from (7) and (8). Thus we should exert caution when comparing estimates from nonlinear models with different functional forms of the dependency between $u_i$ and $x_{ij}$. A strict comparison of coefficients is not possible (Allison 1999; Kohler, Karlson, and Holm 2011). Instead, one should carefully inspect if the additional nonlinear dependencies are statistically significant or use a likelihood-ratio (LR) test to decide whether to remove or retain them.

4.3 Random slopes

Just like standard GLMMs, these hybrid and correlated random-effects models can include random slopes that allow the (within) effects of the level-one variables to vary between clusters. A hybrid model with a random slope on $\beta_W$ is

$$g(\mu_{ij}) = (\beta_W + u_{2i})(x_{ij} - \bar{x}_i) + \beta_B \bar{x}_i + \gamma c_i + u_{1i}$$

What is the advantage of using a hybrid model with random slopes over a standard random-slope model? Again, in the standard random-effects model, we assume the random effects (including the random slope) to be uncorrelated with any unobserved characteristics at level two. Using a hybrid model relaxes this assumption. One can also specify correlated random-effects models with random slopes. However, while correlated random-effects and hybrid models without random slopes produce equivalent results (both in terms of fixed coefficients and variance components), this is not the case if random slopes are present (Kreft, de Leeuw, and Aiken 1995).

5 Hybrid and correlated random-effects models in Stata: The xthybrid command

Correlated random-effects and hybrid models have been increasingly discussed in the methodological literature (Allison 2009; Bell and Jones 2015; Neuhaus and Kalbfeisch 1998; Neuhaus and McCulloch 2006; Rabe-Hesketh and Skrondal 2012; Schunck 2013). However, their use in the wider scientific community is not widespread. The xthybrid command in this article simplifies the specification of hybrid as well as correlated random-effects models. It builds on the existing mundlak command (Perales 2013). However, unlike mundlak, it allows for all models belonging to the class of GLMM.

5.1 Syntax

xthybrid relies on Stata’s meglm command to estimate hybrid and correlated random-effects versions of any two-level specification that can be fit with meglm. The syntax for xthybrid is
xthybrid depvar indepvars [if] [in], clusterid(varname) [family(type)
link(type) cre nonlinearities(type) randomslope(varlist) use(varlist)
percentage(#) test full stats(list) se t p star vce(vcetype) iterations
meglmoptions(list)]

5.2 Options

clusterid(varname) specifies the cluster or grouping variable. clusterid() is re-
quired.

family(type) specifies the distribution of the outcome variable. type may be gaussian,
bernoulli, binomial, gamma, nbinomial, ordinal, or poisson. The default is
family(gaussian).

link(type) specifies the link function. type may be identity, log, logit, probit, or
cloglog. The default is link(identity).

cre requests a correlated random-effects model instead of a hybrid model.

nonlinearities(type) adds polynomial functions of the cluster means to the model.

randomslope(varlist) requests random slopes on the random-effect and within-group
coefficients of selected variables.

use(varlist) splits between- and within-cluster effects only for selected explanatory vari-
ables.

percentage(#) sets the minimum percent within-cluster variance for explanatory vari-
ables to be considered cluster varying.

test presents test results of the random-effects assumption for separate model variables.

full prints the full model output (meglm).

stats(list) allows users to select which model summary statistics are reported.

se requests SEs for the parameters on model variables.

t requests t-values for the parameters on model variables.

p requests p-values for the parameters on model variables.

star requests stars to denote statistically significant parameters on model variables.

vce(vcetype) specifies the type of SE to be reported. vcetype may be oim, robust, or
cluster clustervar.

iterations requests that the command be executed noisily.

meglmoptions(list) enables the user to request options from the meglm command.
5.3 Stored results

`xthybrid` returns in `e()` the same results as `meglm` because `xthybrid` relies on `meglm`. See [ME] `meglm` for details.

5.4 Applications

We now illustrate the `xthybrid` command through practical examples. We use Stata’s `nlswork.dta`, which contains unbalanced panel data on a sample of 4,711 young, working, American women, observed up to 15 times between 1968 and 1988. The data are hence nested so that the level-one units are person-year observations and the level-two units are individuals. The cluster variable is `idcode`. Suppose that our interest is on the relationships between several socioeconomic factors (`age`, `msp`, and `race`) and the number of weekly hours worked (`hours`). Two of these factors are level-one variables that vary both within and between clusters (`age` and `msp`), whereas one is a level-two variable that varies only between clusters (`race`).

We first open the dataset, describe its contents, and create dummy variables out of the `race` variable:

```
. webuse nlswork
(National Longitudinal Survey. Young Women 14-26 years of age in 1968)
. describe idcode hours age msp race
    storage  display value
  variable name   type format label          variable label
    idcode        int  %8.0g   NLS ID
    hours        int  %8.0g   usual hours worked
    age          byte  %8.0g   age in current year
    msp          byte  %8.0g   1 if married, spouse present
    race         byte  %8.0g   racelbl race
. generate black = race==2 if race!=.
. generate other = race==3 if race!=.
```

We can estimate the relationships between the variables of interest as a linear hybrid model using the `xthybrid` command. The `se` option requests SEs to be displayed below model coefficients, while the `test` option requests separate tests of the random-effects assumption (\( \tau = 0 \) or \( \beta_W = \beta_B \) depending on the specification) for individual regressors.

```
. xthybrid hours age msp black other, clusterid(idcode) se test
The variable `black` does not vary sufficiently within clusters and will not be used to create additional regressors.
[0% of the total variance in `black` is within clusters]
The variable `other` does not vary sufficiently within clusters and will not be used to create additional regressors.
[0% of the total variance in `other` is within clusters]
```

<table>
<thead>
<tr>
<th>Variable</th>
<th>model</th>
</tr>
</thead>
<tbody>
<tr>
<td>hours</td>
<td></td>
</tr>
<tr>
<td>( R_{\text{black}} )</td>
<td>0.5470</td>
</tr>
<tr>
<td>( R_{\text{other}} )</td>
<td>-0.0404</td>
</tr>
<tr>
<td>( W_{\text{age}} )</td>
<td>-0.0236</td>
</tr>
<tr>
<td>( W_{\text{msp}} )</td>
<td>-1.1661</td>
</tr>
<tr>
<td>( B_{\text{age}} )</td>
<td>0.0552</td>
</tr>
<tr>
<td>( B_{\text{msp}} )</td>
<td>-3.3647</td>
</tr>
<tr>
<td>_cons</td>
<td>36.5595</td>
</tr>
</tbody>
</table>

\[ \text{var(\_cons[idcode])} \]
\[ \_\_cons \]

\[ \text{var(e.hours)} \]
\[ \_\_cons \]

<table>
<thead>
<tr>
<th>Statistics</th>
<th>value</th>
</tr>
</thead>
<tbody>
<tr>
<td>ll</td>
<td>-1.032e+05</td>
</tr>
<tr>
<td>chi2</td>
<td>259.5143</td>
</tr>
<tr>
<td>p</td>
<td>0.0000</td>
</tr>
<tr>
<td>aic</td>
<td>2.065e+05</td>
</tr>
<tr>
<td>bic</td>
<td>2.066e+05</td>
</tr>
</tbody>
</table>

legend: b/se

Level 1: 28428 units. Level 2: 4709 units.

Tests of the random effects assumption:
\[ _b[B_{\text{age}}] = _b[W_{\text{age}}]; \text{p-value: 0.0004} \]
\[ _b[B_{\text{msp}}] = _b[W_{\text{msp}}]; \text{p-value: 0.0000} \]

In the \texttt{xthybrid} output, variables with the \( W \) prefix denote within-cluster effects, variables with the \( B \) prefix denote between-cluster effects, and variables with the \( R \) prefix are those for which their effects are estimated the same as those in a standard random-effects model.\(^4\)

As expected, \texttt{xthybrid} estimates two separate effects for the level-one variables \texttt{age} and \texttt{msp}. The coefficients \( W_{\text{age}} \) (−0.024) and \( W_{\text{msp}} \) (−1.166) give the within-cluster effects. Within-cluster increases in \texttt{age} are associated with a within-cluster decrease in \texttt{hours}, as are within-cluster increases in \texttt{msp}. That is, women in this sample work fewer hours in those years in which they are younger and unmarried relative to those years in which they are older and married, all else being equal.

\(^4\) \texttt{xthybrid} generates these variables in the background during operation. Users should be aware—if variables with the same name already exist in the active dataset, \texttt{xthybrid} will issue an error.
The coefficients on the level-one variables $B_{age}$ and $B_{msp}$ give their between-cluster effects. For $B_{age}$, the estimated coefficient (0.055) indicates that a between-individual one-year increase in age is associated with a small increase in work hours, suggesting that women from younger cohorts work more hours. For $B_{msp}$, the estimated coefficient ($-3.365$) indicates that on average, women who are never married in the data work about three hours less than women who are always married, all other things being equal.

The within-cluster effects are statistically different from the between-cluster effects, as can be seen from the small $p$-values in the formal tests of the random-effects assumption of orthogonality between the observables and the unobservables ($b[B_{age}] = b[W_{age}]$ p-value: 0.0004 and $b[B_{msp}] = b[W_{msp}]$ p-value: 0.0000). This constitutes evidence in favor of rejecting such an assumption as well as using a standard random-effects model.

An analogous correlated random-effects model can be estimated using `xthybrid` by adding the `cre` option (the results are presented in table 2):

```
.xthybrid hours age msp black other, cluster(idcode) se test cre
(output omitted)
```

For the sake of comparison, estimates from analogous, standard, random-effects, and fixed-effects models (estimated using `xtreg`) are also presented in table 2. Such models are fit as follows:

```
5. xtreg hours age msp black other, i(idcode) fe
(output omitted)
.xtreg hours age msp, i(idcode) fe
(output omitted)
```

---

5. Note that we include the level-two variables `black` and `other` in the fixed-effects model although they are omitted in the estimation. This ensures that the fixed-effects model uses the same sample as the random-effects model. There are, obviously, more elegant ways to define the analysis sample (see, for example, Schunck [2013]).
Table 2. Coefficients from linear models (identity link and Gaussian distribution)

<table>
<thead>
<tr>
<th></th>
<th>(1) Hybrid model</th>
<th>(2) Correlated random-effects model</th>
<th>(3) Random-effects model</th>
<th>(4) Fixed-effects model</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>W_age</strong></td>
<td>$-0.024^*$</td>
<td>$-0.024^*$</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(0.010)</td>
<td>(0.010)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>W_msp</strong></td>
<td>$-1.166^{***}$</td>
<td>$-1.166^{***}$</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(0.153)</td>
<td>(0.153)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>B_age</strong></td>
<td>0.055**</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(0.020)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>B_msp</strong></td>
<td>$-3.365^{***}$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(0.269)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>D_age</strong></td>
<td></td>
<td>0.079^{***}</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0.022)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>D_msp</strong></td>
<td></td>
<td>$-2.199^{***}$</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0.309)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>R_black</strong></td>
<td>0.547*</td>
<td>0.547*</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(0.228)</td>
<td>(0.228)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>R_other</strong></td>
<td>$-0.040$</td>
<td>$-0.040$</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(0.940)</td>
<td>(0.940)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>age</strong></td>
<td></td>
<td>$-0.010$</td>
<td>$-0.024^*$</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0.009)</td>
<td>(0.010)</td>
<td></td>
</tr>
<tr>
<td><strong>msp</strong></td>
<td></td>
<td>$-1.627^{***}$</td>
<td>$-1.166^{***}$</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0.134)</td>
<td>(0.152)</td>
<td></td>
</tr>
<tr>
<td><strong>black</strong></td>
<td></td>
<td>0.860^{***}</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0.243)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>other</strong></td>
<td></td>
<td>$-0.035$</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>(1.021)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>cons</strong></td>
<td>36.560^{***}</td>
<td>36.560^{***}</td>
<td>37.303^{***}</td>
<td>37.944^{***}</td>
</tr>
<tr>
<td></td>
<td>(0.596)</td>
<td>(0.596)</td>
<td>(0.286)</td>
<td>(0.284)</td>
</tr>
<tr>
<td>N (Level 2)</td>
<td>4709</td>
<td>4709</td>
<td>4709</td>
<td>4709</td>
</tr>
<tr>
<td>N (Level 1)</td>
<td>28428</td>
<td>28428</td>
<td>28428</td>
<td>28428</td>
</tr>
</tbody>
</table>

Standard errors in parentheses
* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

In the correlated random-effects model, the coefficients $W_{age}$ ($-0.024$) and $W_{msp}$ ($-1.166$) give the within-cluster effects on the *age* and *msp* and are identical to those fit in the hybrid model. In these linear models, the within-cluster effects fit by both the hybrid and correlated random-effects models are the same as those fit by a standard fixed-effects model.
The coefficients $D_{\text{age}}$ (0.079) and $D_{\text{msp}}$ (−2.199) give the difference between the between- and within-cluster effects. For example, using the estimated between- and within-cluster effects for the variable age from the hybrid model (that is, $B_{\text{age}}$ and $W_{\text{age}}$) and the $D_{\text{age}}$ coefficient in the correlated random-effects model, we see that $0.055 - (−0.024) = 0.079$.

In the correlated random-effects model, the coefficients on the cluster-invariant variables $R_{\text{black}}$ (0.547) and $R_{\text{other}}$ (−0.040) are estimated like those in a standard random-effects regression model and are identical to those in the hybrid model. For these to be unbiased, the random-effects assumption of orthogonality between observables and unobservables at level two must still hold. Note that these coefficients are not identical to those in the standard random-effects model. The inclusion of the cluster-mean variables accounts for additional sources of between-cluster variation, which affects the estimated effects of these level-two variables (Schunck 2013, 71).

We can also use xthybrid to fit a model with random slopes for level-one variables. For instance, if we wanted to allow the slope of age to vary across clusters, we would specify

```
. xthybrid hours age msp black other, cluster(idcode) se randomslope(age)
> iterations
The variable 'black' does not vary sufficiently within clusters and will not be used to create additional regressors.
> 0% of the total variance in 'black' is within clusters
The variable 'other' does not vary sufficiently within clusters and will not be used to create additional regressors.
> 0% of the total variance in 'other' is within clusters
Fitting fixed-effects model:
Iteration 0:  log likelihood = -105142.73
Iteration 1:  log likelihood = -105142.73
Refining starting values:
Grid node 0:  log likelihood = -104177.16
Fitting full model:
Iteration 0:  log likelihood = -104177.16  (not concave)
Iteration 1:  log likelihood = -104034.22
Iteration 2:  log likelihood = -102461.4
Iteration 3:  log likelihood = -102325.48
Iteration 4:  log likelihood = -102298.87
Iteration 5:  log likelihood = -102298.83
Iteration 6:  log likelihood = -102298.83
```
**Mixed-effects GLM**

<table>
<thead>
<tr>
<th>Family:</th>
<th>Gaussian</th>
</tr>
</thead>
<tbody>
<tr>
<td>Link:</td>
<td>identity</td>
</tr>
<tr>
<td>Group variable:</td>
<td>idcode</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Number of obs</th>
<th>28,428</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number groups</td>
<td>4,709</td>
</tr>
</tbody>
</table>

**Obs per group:**

- min = 1
- avg = 6.0
- max = 15

**Integration method:** mvaghermite

| Integration pts. | 7 |

**Log likelihood** = -102298.83

<table>
<thead>
<tr>
<th>Wald chi2(6)</th>
<th>201.16</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prob &gt; chi2</td>
<td>0.0000</td>
</tr>
</tbody>
</table>

| hours       | Coef.  | Std. Err. | z     | P>|z|   | [95% Conf. Interval] |
|-------------|--------|-----------|-------|-------|---------------------|
| R_black     | 0.517  | 0.231     | 2.24  | 0.025 | 0.065, 0.970        |
| R_other     | -0.140 | 0.949     | -0.15 | 0.882 | -2.000, 1.718       |
| W_age       | -0.014 | 0.016     | -0.87 | 0.385 | -0.046, 0.018       |
| W_msp       | -0.607 | 0.159     | -3.82 | 0.000 | -0.918, -0.296      |
| B_age       | -3.364 | 0.271     | -12.43| 0.000 | -3.894, -2.833      |
| _cons       | 36.456 | 0.594     | 61.33 | 0.000 | 35.3, 37.62         |

<table>
<thead>
<tr>
<th>idcode</th>
<th>var(W_age)</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>var(_cons)</td>
<td>0.497</td>
<td>0.023</td>
<td>0.453</td>
<td>0.544</td>
<td></td>
</tr>
<tr>
<td>var(e.hours)</td>
<td>55.187</td>
<td>0.557</td>
<td>54.105</td>
<td>56.290</td>
<td></td>
</tr>
</tbody>
</table>

**LR test vs. linear model:**

- LR χ²(2) = 5687.80
- Prob > χ² = 0.0000

Note: LR test is conservative and provided only for reference.

An LR test, which compares the restricted model (the model without the random slope) with the unrestricted model (the model with the random slope), indicates that the model with a random slope for age fits the data better (LR χ²(1) = 1861.97).
Within- and between-cluster effects in GLMM

Because `xthybrid` relies on the Stata `meglm` command, it can easily handle nonlinear models. To illustrate this, we transform our continuous outcome variable (`hours`) into a dummy variable (`full_time`) where 1 indicates that the respondent works full-time (> 30 hours) and 0 indicates that the respondent works part-time (≤ 30 hours). We then fit a model using the logit link and the binomial distribution for the outcome variable, which yields a logistic regression (table 3):

```
. generate full_time = hours > 30 if hours!=.
. xthybrid full_time age msp black other, clusterid(idcode) family(binomial)
   > link(logit) se
   (output omitted)
```

If we wanted a correlated random-effects logit model, we would instead specify

```
. xthybrid full_time age msp black other, clusterid(idcode) family(binomial)
   > link(logit) se cre
   (output omitted)
```

The results show that also in nonlinear specifications, the estimated within-cluster effects on the level-one and level-two variables are the same in the hybrid model (column 1) and the correlated random-effects model (column 2). The corresponding standard random-intercept and fixed-effects logit models are specified as (for results, see table 3)

```
. xtlogit full_time age msp black other, i(idcode) re
   (output omitted)
. xtlogit full_time age msp black other, i(idcode) fe
   (output omitted)
```

However, as explained before, within-cluster effects from hybrid and correlated random-effects models in these nonlinear specifications are not identical to those from a standard fixed-effects model, though they are very similar. For instance, the estimated within effect for `age` from the hybrid model and the correlated random-effects model is \(-0.020\), whereas in the fixed-effects model this effect is \(-0.019\).

Note that the number of cases differs for the fixed-effects model and the other models. This is because the fixed-effects logit model discards all clusters that have no within-cluster variation, because they do not contribute information about the parameters being estimated (these clusters’ contributions to the log likelihood is zero). However, although the number of cases is higher in the hybrid and the correlated random-effects models, these models are not more efficient when it comes to estimating the within-cluster effects. Here, too, we can use only those cases that have within variation in the dependent variable and the covariates. However, we use all cases to estimate the between-cluster effects, that is, the effects of level-two covariates, and the variance components.
Table 3. Coefficients from logit models (logit link and binomial distribution)

<table>
<thead>
<tr>
<th></th>
<th>(1) Hybrid model</th>
<th>(2) Correlated random-effects model</th>
<th>(3) Random-effects model</th>
<th>(4) Fixed-effects model</th>
</tr>
</thead>
<tbody>
<tr>
<td>W_age</td>
<td>$-0.020^{***}$</td>
<td>$-0.020^{***}$</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(0.003)</td>
<td>(0.003)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>W_msp</td>
<td>$-0.438^{***}$</td>
<td>$-0.438^{***}$</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(0.058)</td>
<td>(0.058)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>B_age</td>
<td>$-0.004$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(0.007)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B_msp</td>
<td>$-1.030^{***}$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(0.093)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D_age</td>
<td></td>
<td>$0.016^{*}$</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0.008)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>D_msp</td>
<td></td>
<td>$-0.592^{***}$</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0.109)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>R_black</td>
<td>$0.453^{***}$</td>
<td>$0.453^{***}$</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(0.079)</td>
<td>(0.079)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>R_other</td>
<td>0.188</td>
<td>0.188</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(0.319)</td>
<td>(0.319)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>age</td>
<td>$-0.017^{***}$</td>
<td>$-0.019^{***}$</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(0.003)</td>
<td>(0.003)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>msp</td>
<td></td>
<td>$-0.602^{***}$</td>
<td>$-0.427^{***}$</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0.050)</td>
<td>(0.057)</td>
<td></td>
</tr>
<tr>
<td>black</td>
<td></td>
<td>0.542^{***}</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0.078)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>other</td>
<td>0.207</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(0.320)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>_cons</td>
<td>2.637^{***}</td>
<td>2.637^{***}</td>
<td>2.730^{***}</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(0.204)</td>
<td>(0.204)</td>
<td>(0.102)</td>
<td></td>
</tr>
<tr>
<td>N (Level 1)</td>
<td>28428</td>
<td>28428</td>
<td>28428</td>
<td>15036</td>
</tr>
<tr>
<td>N (Level 2)</td>
<td>4709</td>
<td>4709</td>
<td>4709</td>
<td>2042</td>
</tr>
</tbody>
</table>

Standard errors in parentheses

* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

We can use the `xthybrid` command to fit hybrid and correlated random-effects models using specifications for which a conditional likelihood (fixed-effects) estimator does not exist or has not been implemented in Stata. The different columns in table 4
show the results of hybrid models fit using \texttt{xthybrid} for three specifications for which a fixed-effects approach is not readily available:\textsuperscript{6}

- a probit model of the binary variable capturing full-time work (column 1)
- an ordered logit model of an ordinal variable capturing work hours (column 2)
- a negative binomial model of weeks spent in unemployment (column 3)

Table 4. Coefficients from hybrid models

<table>
<thead>
<tr>
<th></th>
<th>Probit</th>
<th>Ordered logit</th>
<th>Negative binomial</th>
</tr>
</thead>
<tbody>
<tr>
<td>( W_{\text{age}} )</td>
<td>-0.009***</td>
<td>-0.004</td>
<td>0.002</td>
</tr>
<tr>
<td></td>
<td>(0.002)</td>
<td>(0.003)</td>
<td>(0.005)</td>
</tr>
<tr>
<td>( W_{\text{msp}} )</td>
<td>-0.234***</td>
<td>-0.320***</td>
<td>-0.240**</td>
</tr>
<tr>
<td></td>
<td>(0.032)</td>
<td>(0.048)</td>
<td>(0.078)</td>
</tr>
<tr>
<td>( B_{\text{age}} )</td>
<td>-0.002</td>
<td>0.028***</td>
<td>-0.004</td>
</tr>
<tr>
<td></td>
<td>(0.004)</td>
<td>(0.006)</td>
<td>(0.006)</td>
</tr>
<tr>
<td>( B_{\text{msp}} )</td>
<td>-0.572***</td>
<td>-0.824***</td>
<td>-0.494***</td>
</tr>
<tr>
<td></td>
<td>(0.052)</td>
<td>(0.076)</td>
<td>(0.073)</td>
</tr>
<tr>
<td>( R_{\text{black}} )</td>
<td>0.248***</td>
<td>-0.077</td>
<td>0.559***</td>
</tr>
<tr>
<td></td>
<td>(0.044)</td>
<td>(0.064)</td>
<td>(0.062)</td>
</tr>
<tr>
<td>( R_{\text{other}} )</td>
<td>0.109</td>
<td>-0.155</td>
<td>0.380</td>
</tr>
<tr>
<td></td>
<td>(0.179)</td>
<td>(0.261)</td>
<td>(0.250)</td>
</tr>
<tr>
<td>_cons</td>
<td>1.484***</td>
<td>1.082***</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(0.114)</td>
<td>(0.164)</td>
<td></td>
</tr>
<tr>
<td>cut1</td>
<td>-2.458***</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(0.170)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>cut2</td>
<td>3.013***</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(0.171)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>cut3</td>
<td>6.719***</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(0.193)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>cut4</td>
<td>10.302***</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(0.530)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \ln \alpha )</td>
<td>2.472***</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(0.024)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( N ) (Level 1)</td>
<td>28428</td>
<td>28428</td>
<td>22794</td>
</tr>
<tr>
<td>( N ) (Level 2)</td>
<td>4709</td>
<td>4709</td>
<td>4709</td>
</tr>
</tbody>
</table>

Standard errors in parentheses

\* \( p < 0.05 \), \** \( p < 0.01 \), \*** \( p < 0.001 \)

\textsuperscript{6} Note that a conditional likelihood approach for the negative binomial model is implemented in Stata, but this method does not control for unobserved heterogeneity at the cluster level (Allison and Waterman 2002; Green 2007; Guimarães 2008).
The requisite syntax to fit these models is

\[
\text{. recode hours 0/20=1 21/40=2 41/60=3 61/100=4 101/200=5, gen(hour_categories)  
(28454 differences between hours and hour_categories)}
\]

\[
\text{. xthybrid full_time age msp black other, clusterid(idcode) family(binomial)  
> link(probit) se  
(output omitted)}
\]

\[
\text{. xthybrid hour_categories age msp black other, clusterid(idcode) family(ordinal)  
> link(logit) se  
(output omitted)}
\]

\[
\text{. xthybrid wks_ue age msp black other, clusterid(idcode) family(nbinomial)  
> link(log) se  
(output omitted)}
\]

If a conditional likelihood estimator is available, as with the logit model, it is easy to judge whether the within effect in the GLMM corresponds to the fixed-effects estimate. If it does not, this may be an indication that \( u_i \) depends on \( x_{ij} \) through other functions of the cluster means of the level-one variables. If a conditional likelihood estimator is not available, an LR test can assess whether including other functions of \( x_{ij} \) is sensible. As an example, the following syntax requests a probit hybrid model with quadratic polynomials of the cluster means:

\[
\text{. xthybrid full_time age msp black other, clusterid(idcode) family(binomial)  
> link(probit) se test nonlinearities(quadratic)  
The variable ‘black’ does not vary sufficiently within clusters and will not be used to create additional regressors.  
[0% of the total variance in ‘black’ is within clusters]}
\]

\[
\text{The variable ‘other’ does not vary sufficiently within clusters and will not be used to create additional regressors.  
[0% of the total variance in ‘other’ is within clusters]}
\]

<table>
<thead>
<tr>
<th>Variable</th>
<th>model</th>
</tr>
</thead>
<tbody>
<tr>
<td>full_time</td>
<td>0.2474</td>
</tr>
<tr>
<td>R_black</td>
<td>0.0436</td>
</tr>
<tr>
<td>R_other</td>
<td>0.1282</td>
</tr>
<tr>
<td>W_age</td>
<td>-0.0093</td>
</tr>
<tr>
<td>W_msp</td>
<td>-0.2390</td>
</tr>
<tr>
<td>B_age</td>
<td>0.0761</td>
</tr>
<tr>
<td>B_age_2</td>
<td>-0.0013</td>
</tr>
<tr>
<td>B_msp</td>
<td>0.2753</td>
</tr>
<tr>
<td>B_msp_2</td>
<td>-0.8287</td>
</tr>
<tr>
<td>_cons</td>
<td>0.2231</td>
</tr>
<tr>
<td></td>
<td>0.4845</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>var(_cons[idcode])</th>
<th>_cons</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.8631</td>
</tr>
<tr>
<td></td>
<td>0.0418</td>
</tr>
</tbody>
</table>

Statistics

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>ll</td>
<td>-1.203e+04</td>
</tr>
<tr>
<td>chi2</td>
<td>316.8241</td>
</tr>
<tr>
<td>p</td>
<td>0.0000</td>
</tr>
<tr>
<td>aic</td>
<td>24082.5895</td>
</tr>
<tr>
<td>bic</td>
<td>24165.1408</td>
</tr>
</tbody>
</table>

Legend: b/se

Tests of the random effects assumption:

- \( _b[B_{age}^2] = _b[W_{age}]; p\)-value: 0.0108
- \( _b[B_{msp}^2] = _b[W_{msp}]; p\)-value: 0.0125

The quadratic terms of the cluster means for the variables \( msp \) and \( age \) are statistically significant (\( B_{age}^2 = -0.001, SE = 0.001; B_{msp}^2 = -0.829, SE = 0.188 \)), but their inclusion has negligible effects (from a substantial significance standpoint) on the estimated within-cluster effects. However, an LR test suggests that the model with the quadratic terms of the cluster means fits the data better than a model without. Including a cubic function does not further improve the model fit:

```stata
. quietly xthybrid full_time age msp black other, clusterid(idcode) family(binomial) link(probit) se test
. estimates store model3
```

Note that a Wald test of joint insignificance of the quadratic and the cubic terms comes to the same conclusion. This is not surprising, considering that the LR test and the Wald test are asymptotically equivalent (Johnston and DiNardo 1997, 150).
In this case, we would choose the model that includes quadratic terms of the cluster means. Finally, as with the linear model, \texttt{xthybrid} can fit nonlinear models with random slopes. For instance, a hybrid GLMM with a logit link and binomial distribution (a hybrid logit model) with a random slope on \texttt{age} is specified as

\begin{verbatim}
. xthybrid full_time age msp black other, clusterid(idcode) family(binomial) link(logit) se randomslope(age)
\end{verbatim}

Note that we fit GLMMs with maximum likelihood estimation (Skrondal and Rabe-Hesketh 2004; McCulloch, Searle, and Neuhaus 2008; Wooldridge 2010). This applies also to models fit using Stata’s \texttt{meglm} command, on which the \texttt{xthybrid} command is based. Stata uses numerical integration to calculate and maximize the likelihood. Because these methods are computationally intensive, the models may take some time to converge. It is therefore sensible to start by fitting simple models, with few random terms.

Maximum likelihood estimates are consistent, asymptotically normally distributed, and asymptotically efficient (McCulloch and Neuhaus 2013; Wooldridge 2010). However, maximum-likelihood estimation does not perform well with small samples, often providing biased estimates (Neuhaus and McCulloch 2006, 79). The minimum number of observations required for robust maximum likelihood estimation of GLMMs depends on the model specification (for discussions on sample sizes in mixed models, see Maas and Hox [2005]; Moineddin, Matheson, and Glazier [2007]; Schunck [2016]).

6 Conclusion

We have discussed the rationale behind hybrid and correlated random-effects models (Allison 2009; Schunck 2013; Wooldridge 2010) for clustered data, focusing on how we can adapt these to specifications that fall under the umbrella of GLMM (McCullagh and Nelder 1989; McCulloch, Searle, and Neuhaus 2008; Rabe-Hesketh, Skrondal, and Pickles 2004; Skrondal and Rabe-Hesketh 2003). We introduced the user-written \texttt{xthybrid} command as an accessible and flexible tool to fit these models using Stata.
Within- and between-cluster effects in GLMM

When a researcher’s main goal is to account for unobserved heterogeneity at the cluster level, fixed effects (for example, conditional likelihood) approaches are the best choice. However, these approaches are not feasible for some models, including probit models, ordered logit or probit models, and certain count models (Wooldridge 2010). In these situations, hybrid and correlated random-effects models are a good solution to obtain proximate fixed-effects estimates (Cameron and Trivedi 2005, 719; Neuhaus and McCulloch 2006, 886).

One limitation of nonlinear hybrid and correlated random-effects models is that the within-cluster effect estimates are not identical to those from standard fixed-effects models, which may lead to bias (Brumback et al. 2010). When possible, we advise comparisons of the estimated within-cluster effects from hybrid and correlated random-effects models and standard fixed-effects models. When not, including nonlinear dependencies in the model can be insightful as to whether the correlated random-effects assumption is violated. Despite this shortcoming relative to traditional conditional likelihood approaches, estimates from correlated random-effects or hybrid models are still preferable to those from standard random-effects models that impose very strict assumptions on the distributions of the unobserved effects and their correlation with the covariates. Correlated random-effects and hybrid models can also be more useful than standard fixed-effects models, because the latter fail to retrieve the effects on the outcome variable of level-two variables. Moreover, correlated random-effects and hybrid models can include random slopes; thus they can relax the assumption of effect homogeneity of level-one variables across clusters. Overall, decomposing within and between effects in GLMM is a flexible alternative to standard random and fixed-effects models.

7 Acknowledgments
The authors thank Marco Giesselmann, Philipp Lersch, and an anonymous reviewer for their constructive and helpful feedback and Yangtao Huang for assistance testing the \texttt{xt}hybrid command. This research was supported by the Australian Research Council Centre of Excellence for Children and Families over the Life Course (project number CE140100027).

8 References


Andersen, E. B. 1970. Asymptotic properties of conditional maximum-likelihood esti-

Baltagi, B. H., ed. 2006. *Panel Data Econometrics: Theoretical Contributions and
Empirical Applications.* Amsterdam: Elsevier.


Bell, A., and K. Jones. 2015. Explaining fixed effects: Random effects modeling of
time-series cross-sectional and panel data. *Political Science Research and Methods* 3:
133–153.

Adjusting for confounding by cluster using generalized linear mixed models. *Statistics &

neighborhood using a proportional odds model and complex survey data. *American

Cameron, A. C., and P. K. Trivedi. 2005. *Microeconometrics: Methods and Applica-
tions.* New York: Cambridge University Press.

Studies* 47: 225–238.

———. 1982. Multivariate regression models for panel data. *Journal of Econometrics*
18: 5–46.


Green, W. 2007. Functional form and heterogeneity in models for count data. *Founda-


Press.

Hill.

Kohler, U., K. B. Karlson, and A. Holm. 2011. Comparing coefficients of nested non-
Within- and between-cluster effects in GLMM


Schunck, R. 2013. Within and between estimates in random-effects models: Advantages

———. 2016. Cluster size and aggregated level 2 variables in multilevel models: A

and mixed models in epidemiology: Repeated measures, measurement error and

Structural Equation Models*. Boca Raton, FL: Chapman & Hall/CRC.


About the authors

Reinhard Schunck is the head of GESIS Training at GESIS—Leibniz-Institute for the Social
Sciences in Cologne, Germany. His research interests span multilevel and longitudinal data
analysis. He works primarily in the field of social stratification and inequality.

Francisco (Paco) Perales is a research fellow at the Life Course Centre at the Institute for Social
Science Research (The University of Queensland). His research relies heavily on the analysis
of large-scale panel surveys. His current interests include the impact of life-course transitions
on gender-based socioeconomic inequality, differences in life outcomes by sexual identity, and
the effect of early life-course family structure on children’s development.
Fitting the errors-in-variables model using high-order cumulants and moments

Timothy Erickson
Bureau of Labor Statistics
Washington, DC
Erickson.Timothy@bls.gov

Robert Parham
University of Rochester
Rochester, NY
robert.parham@simon.rochester.edu

Toni M. Whited
University of Michigan
Ann Arbor, MI
twhited@umich.edu

Abstract. In this article, we consider a multiple mismeasured regressor errors-in-variables model. We present \texttt{xtewreg}, a command for using two-step generalized method of moments and minimum distance estimators that exploit overidentifying information contained in high-order cumulants or moments of the data. The command supports cumulant or moment estimation, internal support for the bootstrap with moment condition recentering, an arbitrary number of mismeasured regressors and perfectly measured regressors, and cumulants or moments up to an arbitrary degree. We also demonstrate how to use the estimators in the context of a corporate leverage regression.

Keywords: st0469, xtewreg, errors-in-variables, high-order moments, high-order cumulants

1 Introduction

We present the command \texttt{xtewreg} for implementing the estimators in Erickson and Whited (2000, 2002, 2012) and Erickson, Jiang, and Whited (2014) for the classical errors-in-variables (EIV) model. The model is of a multiple linear regression, in which we can measure any number of the explanatory variables with an additive error. The estimators produce consistent regression slope estimates by exploiting information contained in the third- and higher-order cumulants or moments of the data. Such estimators are of interest because ordinary least squares (OLS) is inconsistent when error-laden measurements or proxy variables replace the independent variables of a linear regression. Typically, researchers address this issue by finding additional observable variables that can serve as instruments—but in many situations, no such variables are available. Therefore, consistent estimators based on the original, unaugmented set of observable variables are potentially quite valuable.

This article proceeds as follows: In section 2, we start with a description of the EIV model and the moment and cumulant estimators. In section 3, we describe the \texttt{xtewreg} command. In section 4, we demonstrate the \texttt{xtewreg} command.
2 Background

This section draws from Erickson and Whited (2002) and Erickson, Jiang, and Whited (2014) to sketch the EIV model and the high-order moment and cumulant estimators. For details, see those articles.

2.1 Notation and model

Let \((y_i, x_i, z_i)\), with \(i = 1, \ldots, n\), be a sequence of observable vectors, where \(x_i \equiv (x_{i1}, \ldots, x_{iJ})\) and \(z_i \equiv (1, z_{i1}, \ldots, z_{iM})\). Let \((u_i, \varepsilon_i, \chi_i)\) be a sequence of unobservable vectors, where \(\chi_i \equiv (\chi_{i1}, \ldots, \chi_{iJ})\) and \(\varepsilon_i \equiv (\varepsilon_{i1}, \ldots, \varepsilon_{iJ})\). We consider a multiple-regressor version of the classical EIV model, where \((y_i, x_i, z_i)\) is related to \((u_i, \varepsilon_i, \chi_i)\) and unknown parameters \(\alpha \equiv (\alpha_0, \alpha_1, \ldots, \alpha_M)'\) and \(\beta \equiv (\beta_1, \ldots, \beta_J)'\) according to

\[ y_i = z_i\alpha + \chi_i\beta + u_i \tag{1} \]

\[ x_i = \chi_i + \varepsilon_i \tag{2} \]

Equation (1) is a linear regression model containing \(J\) regressors \(\chi_i\) that are imperfectly measured by \(x_i\) according to (2) and containing \(M\) perfectly measured regressors \(z_i\). We must assume the unit slopes and zero-valued intercepts in (2) to identify the parameters in (1). We assume the variables in (1) and (2) satisfy the following assumptions:

i) \((u_i, \varepsilon_i, \chi_i, z_i), i = 1, \ldots, n,\) is an independent and identically distributed sequence;

ii) \(u_i\) and the elements of \(\varepsilon_i, \chi_i,\) and \(z_i\) have finite moments of every order;

iii) \((u_i, \varepsilon_i)\) is independent of \((\chi_i, z_i),\) and the individual elements in \((u_i, \varepsilon_i)\) are independent of each other;

iv) \(E(u_i) = 0\) and \(E(\varepsilon_i) = 0;\)

v) \(E\{(\chi_i, z_i)'(\chi_i, z_i)\}\) is positive definite.

Before sketching the estimators, we partial out the perfectly measured variables and rewrite the model in terms of population residuals. The \(1 \times J\) residual from the population linear regression of \(x_i\) on \(z_i\) is \(x_i - z_i\mu_x,\) where

\[ \mu_x \equiv \{E(z_i'z_i)\}^{-1} E(z_i'x_i) \]

The corresponding \(1 \times J\) residual from the population linear regression of \(\chi_i\) on \(z_i\) is

\[ \eta_i \equiv \chi_i - z_i\mu_x \]

where \(\mu_x\) appears because (2) and the independence of \(\varepsilon_i\) and \(z_i\) imply

\[ \mu_x = \{E(z_i'z_i)\}^{-1} E(z_i'\chi_i + \varepsilon_i) = \{E(z_i'z_i)\}^{-1} E(z_i'\chi_i) \]

Subtracting \(z_i\mu_x\) from both sides of (2) gives

\[ x_i - z_i\mu_x = \eta_i + \varepsilon_i \tag{3} \]
Similarly, the residual from the population linear regression of \( y_i \) on \( z_i \) is \( y_i - z_i \mu_y \), where \( \mu_y = (E(z'_iz_i))^{-1}E(z'_iy_i) \). Equation (1) and the independence of \( u_i \) and \( z_i \) imply
\[
\mu_y = (E(z'_iz_i))^{-1}E(z'_i(z_i\alpha + \chi_i\beta + u_i)) = \alpha + \mu_x\beta
\]
Therefore, subtracting \( z_i \mu_y \) from both sides of (1) gives
\[
y_i - z_i \mu_y = \eta_i\beta + u_i \tag{5}
\]

### 2.2 Estimators

Both the cumulant and moment estimators are based on a two-step approach to estimation, where the first step is to substitute the least-squares estimates
\[
\hat{\mu}_x = \left(\sum_{i=1}^{n} z'_iz_i\right)^{-1} \sum_{i=1}^{n} z'_ix_i
\]
\[
\hat{\mu}_y = \left(\sum_{i=1}^{n} z'_iz_i\right)^{-1} \sum_{i=1}^{n} z'_iy_i
\]
into (3) and (5), and the second step is to estimate \( \beta \) using sample cumulants or moments of \( y_i - z_i \hat{\mu}_y \) and \( x_i - z_i \hat{\mu}_x \).

Regarding the practical implementation of this step, the researcher should classify all possibly mismeasured variables as belonging to vector \( \chi_i \) rather than vector \( z_i \). Correct classification is important, even if one or more of the mismeasured variables is not of primary economic interest. If you classify any mismeasured regressor as perfectly measured, then the OLS estimates—\( \hat{\mu}_x \) and \( \hat{\mu}_y \)—will be biased. In this case, (3) and (5) will be misspecified.

### Moments

The high-order moment estimators are based on moment conditions derived from (3) and (5) by taking powers of these two equations, multiplying the results together, and then taking expectations of both sides. The resulting equations express observable higher-order moments and cross-moments of the data as nonlinear functions of \( \beta \) and moments of unobservable variables. The equations treat these latter moments as parameters. The general form for these moment equations is given by
\[
E \left\{ (y_i - z_i \mu_y)^{r_0} \prod_{j=1}^{J} (x_{ij} - z_i \mu_{xj})^{r_j} \right\} = \sum_{v \in V} \sum_{m \in M} a_{v,m} \left( \prod_{j=1}^{J} \beta_j^{n_{v,j}} \right) \left[ \prod_{j=1}^{J} E \left\{ \eta^{(r_j + m_j)}_{ij} \right\} \right] \left[ \prod_{j=1}^{J} E \left\{ \varepsilon^{(r_j - m_j)}_{ij} \right\} \right] E(u_i^{r_0})
\]
where \( v \equiv (v_0, v_1, \ldots, v_J) \) and \( m \equiv (m_1, \ldots, m_J) \) are vectors of nonnegative integers, \( V \equiv (\sum_{j=0}^{J} v_j = r_0) \), \( M \equiv (m_j : \sum_{j=0}^{J} m_j \leq \sum_{j=0}^{J} r_j, m_j \leq r_j, j = 1, \ldots, J) \), and

\[
av_{v,m} \equiv \frac{r_0!}{v_0! v_1! \cdots v_J!} \prod_{j=1}^{J} \frac{r_j!}{m_j! (r_j - m_j)!}.
\]

We can then construct a generalized method of moments (GMM) estimator using subsets of these moment conditions, where the weight matrix is simply the covariance matrix of the observable moments on the left-hand side of (6), adjusted to account for the sampling variation in the estimates of \( \mu_x \) and \( \mu_y \). As explained in more detail in Erickson and Whited (2002), it is natural to consider sets of equations based on moments up to a certain order \( N = r_0 + r_1 + \cdots + r_J \), so the \texttt{xtewreg} command considers sets of moment equations based on moments of order 3, 4, 5, and so on.

We now describe a simple example of (6) that we can use to construct an estimator. We consider the case of a single mismeasured regressor, so \( J = 1 \). First, we square (5), multiply the result by (3), and take expectations of both sides, obtaining

\[
E \{(y_i - z_i \mu_y)^2 (x_i - z_i \mu_x)\} = \beta^2 E (\eta_i^3)
\]

Similarly, if we square (3), multiply the result by (5), and take expectations, we obtain

\[
E \{(y_i - z_i \mu_y) (x_i - z_i \mu_x)^2\} = \beta E (\eta_i^3)
\]

If \( \beta \neq 0 \) and \( E (\eta_i^3) \neq 0 \), then dividing (7) by (8) produces a consistent estimator for \( \beta \):

\[
\beta = \frac{\beta^2 E (\eta_i^3)}{\beta E (\eta_i^3)} = \frac{E \{(y_i - z_i \mu_y)^2 (x_i - z_i \mu_x)\}}{E \{(y_i - z_i \mu_y) (x_i - z_i \mu_x)^2\}}
\]

We can derive an estimator from (9) by replacing the population moments by sample moments.

**Cumulants**

As shown in Erickson, Jiang, and Whited (2014), the cumulant estimators are asymptotically equivalent to the moment estimators, but they have a convenient closed form. The following outline of the estimators draws from Erickson, Jiang, and Whited (2002). Let \( K(s_0, s_1, \ldots, s_J) \) be the cumulant of order \( s_0 \) in \( y_i - z_i \mu_y \) and \( s_j \) in \( x_{ij} - z_i \mu_x \). The cumulant estimators are based on the result from Geary (1942)—that is, for any \((s_0, s_1, \ldots, s_J)\) containing two or more positive elements, the following relationship between cumulants holds:

\[
K(s_0 + 1, s_1, \ldots, s_J) = \beta_1 K(s_0, s_1 + 1, \ldots, s_J) + \cdots + \beta_J K(s_0, s_1, \ldots, s_J + 1)
\]
High-order cumulants and moments

An infinite amount of equations are given by (10), one for each admissible vector \((s_0, s_1, \ldots, s_J)\). Let

\[ K_y = K_x \beta \]

denote a system of \(M\) equations of the form (10). If \(M = J\) and \(\det K_x \neq 0\), then it is possible to solve for \(\beta\).

We consider possibly overidentified estimators for \(\beta\), so \(M \geq J\). Let \(\hat{K}_y\) and \(\hat{K}_x\) be consistent estimates of \(K_y\) and \(K_x\), and let \(\hat{W}\) be a symmetric positive-definite matrix. The estimator \(\hat{\beta}\) solves

\[ \hat{\beta} = \arg\min_{b \in \mathbb{R}^J} \left( \hat{K}_y - \hat{K}_x b \right)^T \hat{W} \left( \hat{K}_y - \hat{K}_x b \right) \]  

(11)

Because \(\hat{K}_y - \hat{K}_x b\) is linear in \(b\), (11) has the solution

\[ \hat{\beta} = \left( \hat{K}_x^T \hat{W} \hat{K}_x \right)^{-1} \hat{K}_x^T \hat{W} \hat{K}_y \]

whenever \(\hat{K}_x\) has full column rank. As in the case of the moment estimators, we consider estimators based on sets of cumulant equations up to a certain integer order, \(N = s_0 + s_1 + \cdots + s_J\).

2.3 Identifying assumptions

Both the cumulant and moment estimators obtain identification from the third- and higher-order moments or cumulants of the regression variables. In particular, as shown in Erickson and Whited (2002), identification requires that the distribution of \(\eta\) satisfy

\[ E\{(\eta_i c)^3\} \neq 0 \]

for every vector of constants \(c = (c_1, \ldots, c_J)\) having at least one nonzero element. For practical problems, this requirement boils down to having nonnormally distributed mismeasured regressors. An example of this requirement can be seen intuitively in (9), which contains the third moment of \(\eta_i\) in the denominator. Without a skewed distribution, this particular third-order moment estimator is undefined. The assumption of nonnormality clearly limits the applicability of these estimators. For instance, asset returns are often approximately normally distributed, and many aggregate variables are often approximately lognormally distributed and typically expressed as natural logarithms. In both of these cases, the cumulant or moment estimators are unlikely to be useful. However, in many microeconometric settings, especially those in corporate finance and accounting, many regression variables are plausibly nonnormally distributed.
2.4 Other estimates and test statistics

Both the moment and the cumulant estimators can produce estimates of the coefficients on the perfectly measured regressors $\alpha$, which can be recovered from identity (4). The estimators can also produce estimates of the population coefficient of determination for (1), which can be written as

$$\rho^2 = \frac{\mu_y' \text{var}(z_i) \mu_y + \beta' \text{var}(\eta_i) \beta}{\mu_y' \text{var}(z_i) \mu_y + \beta' \text{var}(\eta_i) \beta + E(u_i^2)}$$

Similarly, the estimators can produce an estimate of the population coefficients of determination for (2):

$$\tau_j^2 = \frac{\mu_{xj}' \text{var}(z_i) \mu_{xj} + \text{var}(\eta_{ij})}{\mu_{xj}' \text{var}(z_i) \mu_{xj} + \text{var}(\eta_{ij}) + E(\varepsilon_{ij}^2)}$$

In (12), the $j$ subscript refers to the $j$th mismeasured regressor. The standard errors for $\alpha$, $\rho^2$, and $\tau^2$ are calculated by stacking the influence functions for their various components to obtain the covariance matrix of these components and then using the delta method.

Finally, except for the case of the third-order moment estimator with one mismeasured regressor, all estimators included in xtewreg are overidentified. The standard Hansen–Sargan test statistic accompanies both the cumulant and the moment estimators for the overidentifying restrictions.

3 The xtewreg command

3.1 Syntax

```
xtewreg depvar misindepvars [indepvars] [if] [in], maxdeg(#) [mismeasured(#) method(string) panmethod(string) bxint(numlist) centmom(string) hacons nocons noprn]
```

Here, misindepvars are independent variables assumed to be mismeasured, and indepvars are independent variables assumed to be perfectly measured. For more than one mismeasured variable, use the mismeasured() option to specify the number of mismeasured independent variables.

3.2 Options

maxdeg(#) sets the highest order of cumulants or moments to use. The minimum value is 3, which corresponds to an exactly identified Geary (1942) estimator. Very high values (above 8) are not advised, because the computational time for these models increases sharply with maxdeg(). xtewreg does not provide a default value
for \texttt{maxdeg()}; this empirical choice is left to the researcher. Generally speaking, the more data one has, the higher order moment or cumulant one can use. A reasonable starting value for applied work is \texttt{maxdeg(5)}, but the sensitivity of the estimates to different values of \texttt{maxdeg()} should be explored on a case-by-case basis. \texttt{maxdeg()} is required.

\texttt{mismeasured(\#)} declares the number of mismeasured independent variables in the model. The default is \texttt{mismeasured(1)}. \texttt{xtewreg} uses this value to distinguish between \texttt{misindepvars} and \texttt{indepvars}. For example, if \texttt{mismeasured(2)} is specified, then the first two listed variables are taken to be \texttt{misindepvars}, and the rest are taken to be \texttt{indepvars}.

\texttt{method(string)} specifies whether to use high-order cumulants (\texttt{cml}, the default) or high-order moments (\texttt{mom}). While \texttt{xtewreg} supports both of these, using high-order moment estimators is not advised. This is because the high-order moment estimators require a numerical minimization procedure when computing the GMM objective function, whereas the cumulant-based estimators are linear with a closed-form solution.

\texttt{panmethod(string)} specifies whether to perform panel estimation by using a clustered weight matrix used for the cumulant or moment estimators (\texttt{cls}, the default) or by combining cross-sections using a minimum distance estimator (\texttt{cmd}). While \texttt{xtewreg} supports panel data using either of these, classical minimum distance can entail long computation time for panels with a large time dimension.

\texttt{bxint(numlist)} specifies a list of starting values for the coefficients on \texttt{misindepvars}. This option requires setting \texttt{method(mom)}. The high-order moment estimators require numerical minimization of a nonlinear objective function and thus require starting values. The default is to use both the \texttt{OLS} coefficients and the coefficients from \texttt{maxdeg()} as possible starting values. If there are $J$ \texttt{misindepvars} and one wishes to provide $K$ sets of possible starting values, then \texttt{numlist} should be of order $J \times K$.

\texttt{centmom(string)} specifies a directive supporting the centering of the moment conditions for bootstrap computation of $t$ test and overidentification test critical values. \texttt{string} may be \texttt{set}, \texttt{use}, or \texttt{reset}. \texttt{centmom(set)} saves the value of the moment conditions for the entire sample and should be used before using the \texttt{bootstrap} command. \texttt{centmom(use)} should be specified when using \texttt{bootstrap} with \texttt{xtewreg}. \texttt{centmom(reset)} resets the value of saved moment conditions and is rarely used.

\texttt{hascons} indicates that \texttt{indepvar} already contains a constant variable, so the estimation procedure should not add a constant.

\texttt{nocons} specifies that a constant not be added by the estimation procedure. When this option is used, the researcher should verify that all variables included in the estimation have mean 0, or regression results will be inconsistent.

\texttt{noprn} suppresses the printing of the results table.
3.3 Stored results

`xtewreg` stores the following in `e()`

Scalars

- `e(N)` : number of observations
- `e(rho)` : estimate of $\rho^2$
- `e(SErho)` : standard error for $\rho^2$
- `e(Jstat)` : Sargan–Hansen $J$ statistic for overidentifying restrictions
- `e(Jval)` : $p$-value for $e(Jstat)$
- `e(dfree)` : degrees of freedom for $e(Jstat)$
- `e(obj)` : minimized value of the GMM objective function

Macros

- `e(bxint)` : `numlist` of initial guesses for $\beta$
- `e(method)` : method used for estimation (`cml` or `mom`)
- `e(panmethod)` : panel method used for estimation (`cmd` or `cls`)

Matrices

- `e(b)` : regression coefficients
- `e(V)` : variance–covariance matrix of the estimators
- `e(se(r))` : standard errors for `e(b)`
- `e(tau)` : estimates of $\tau^2$, the proxy accuracy indices
- `e(SEtau)` : standard errors for $\tau^2$
- `e(vchrotau)` : variance–covariance matrix for $\rho^2$ and all of the $\tau^2$
- `e(w)` : weighting matrix used for estimation in the GMM estimation

Additionally, `xtewreg` sets two global Mata variables:

- `EWSAVEDprb` holds the problem structure (that is, the Symbolic estimation equations) for a given number of mismeasured independent variables $J$ and a given maximum cumulant or moment degree $M$. Generating these equations is computationally intensive, and `xtewreg` saves the last estimated problem structure to optimize repeated estimations of the same problem structure, for example, when using bootstrap. Calling `xtewreg` with a problem structure different from the one last used causes it to print the message “Problem structure different from last executed. Rebuilding problem.” `xtewreg` will rebuild and save the new Symbolic estimation equations.

- `EWSAVEDfCent` holds the centered moment conditions generated by specifying `centmom(set)` and used when specifying `centmom(use)`.

3.4 The Symbolic class

To implement a moment system of arbitrary degree $M$ and with arbitrarily many mismeasured variables $J$, `xtewreg` needs to be able to construct a large set of equations of the type described by the general form in (6). We then need to evaluate these equations for the data provided to calculate the moments and cumulants. To construct these equations, we implement a symbolic algebra class in Mata, `Symbolic`, which supports the complete algebra over the polynomial ring with arbitrarily many indeterminates and with coefficients from the real field. The class is similar in capabilities to Stata’s `polyeval()` function, with two important differences: it is a stateful Mata class, which
allows superior encapsulation, and it supports arbitrarily many indeterminates (for example, polynomials of the form $\alpha_1 x_1 + \alpha_2 x_2 + \alpha_3 x_1 x_2^2$), whereas \texttt{polyeval()} supports only a single indeterminate (that is, $x$ is a scalar, not a vector). Further discussion of the \texttt{Symbolic} class is outside the scope of this article, but a stand-alone version of the \texttt{Symbolic} class is available from the authors upon request.

The \texttt{EwSAVEDprb} problem structure contains the set of \texttt{Symbolic} equations corresponding to the current degree $M$ and mismeasured variable count $J$, and these equations can then be reevaluated given a set of data. This way, the problem structure is constructed only once and can then be evaluated multiple times given different data.

4 Investment and leverage example

This distribution includes a firm leverage dataset from Compustat. It contains over 121,000 firm-year observations for approximately 11,000 firms. We include the following variables, defined in terms of Compustat mnemonic variable names:

- \texttt{gvkey} — The Compustat unique firm identifier.
- \texttt{fyear} — The firm fiscal year.
- \texttt{lever} — Firm leverage, defined as (total long-term debt in current liabilities)/total assets.
- \texttt{mtb} — Firm market-to-book ratio, where the numerator is total assets plus price close annual times common shares outstanding minus total common equity minus deferred taxes balance sheet, and the denominator is total assets.
- \texttt{tangib} — Fixed assets, defined as total property, plant, and equipment (net)/total assets.
- \texttt{logsales} — The natural log of firm sales (sales—turnover net).
- \texttt{oi} — Firm operating income, defined as operating income before depreciation/total assets.

All cash items are provided in terms of deviation from firm mean and year mean.
First, we provide a summary of `epw.dta`:

```stata
use epw
taxset gvkey
  panel variable: gvkey (unbalanced)
summarize fyear gvkey lever mtb tangib logsales oi
```
```
<table>
<thead>
<tr>
<th>Variable</th>
<th>Obs</th>
<th>Mean</th>
<th>Std. Dev.</th>
<th>Min</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>gvkey</td>
<td>121,733</td>
<td>21855.43</td>
<td>35329.08</td>
<td>1000</td>
<td>287462</td>
</tr>
<tr>
<td>lever</td>
<td>121,733</td>
<td>-1.45e-10</td>
<td>0.1482152</td>
<td>-.9990212</td>
<td>.9677935</td>
</tr>
<tr>
<td>mtb</td>
<td>121,733</td>
<td>1.15e-10</td>
<td>0.6669923</td>
<td>-9.285594</td>
<td>12.85808</td>
</tr>
<tr>
<td>tangib</td>
<td>121,733</td>
<td>-8.37e-11</td>
<td>0.1167422</td>
<td>-.8033313</td>
<td>.7093889</td>
</tr>
<tr>
<td>logsales</td>
<td>121,733</td>
<td>-4.75e-10</td>
<td>.5162762</td>
<td>-6.318236</td>
<td>4.589146</td>
</tr>
<tr>
<td>oi</td>
<td>121,733</td>
<td>5.28e-12</td>
<td>.0930696</td>
<td>-1.030859</td>
<td>.6828895</td>
</tr>
</tbody>
</table>
```

We begin by using an OLS regression to estimate the relationship between leverage, market-to-book, and tangibility. The market-to-book ratio is a proxy for firm growth opportunities, and the ratio of fixed to total assets is a proxy for asset tangibility. We cluster standard errors at the firm level and use the `nocons` option in the regression because the dependent variable was demeaned:

```
regress lever mtb tangib, vce(cluster gvkey) nocons
```

```
Linear regression
Number of obs = 121,733
F(2, 10795) = 497.72
Prob > F = 0.0000
R-squared = 0.0390
Root MSE = .14529

(Std. Err. adjusted for 10,796 clusters in gvkey)

| Coef.  | Std. Err. | t      | P>|t|  | [95% Conf. Interval] |
|--------|-----------|--------|------|----------------------|
| lever  | -.0242871 | .0011228 | -21.63 | 0.000   | -.0264881 to -.0220862 |
| mtb    | .2049681  | .0098809 | 20.74  | 0.000   | .1855996 to .2243365 |
| tangib | .115e-10  | .0014767 | -8.91  | 0.000   | -.0189939 to .0451727 |
```

Compare these results with those of `xtewreg`, assuming both regressors are measured with error (as indicated by the `mismeasured(2)` option):

```
xtewreg lever mtb tangib, maxdeg(5) mismeasured(2) nocons
```

```
5(2) EIV results
N = 121733
Rho^2 = 0.171
(Sargan-Hansen J statistic: 210.285 (p=0.000, d=20)

| Coef.  | Std. Err. | z      | P>|z|  | [95% Conf. Interval] |
|--------|-----------|--------|------|----------------------|
| mtb    | -.0339095 | .0037329 | -9.08  | 0.000   | -.0412259 to -.0265932 |
| tangib | 1.185099  | .0373822 | 31.70  | 0.000   | 1.111831 to 1.258367  |
```

Tau1^2: 0.570 (0.083)
Tau2^2: 0.172 (0.010)
Sargan-Hansen J statistic: 210.285  

The coefficient on tangibility rises by a factor of six, and the coefficient of determination ($\rho^2$) for the model rises considerably. These are explained by the estimates of the errors in market-to-book and tangibility, measured by the $\tau_1^2$ and $\tau_2^2$ coefficients. These errors, when ignored in OLS, lead to attenuation bias. Note that the estimation uses cumulants and a clustered weighting matrix (the defaults), and we set maxdeg(5) for an estimator based on cumulants up to fifth order.

Next, we add several perfectly measured controls and fit the model using an OLS regression. We again cluster standard errors at the firm level.

```
regress lever mtb tangib logsales oi, vce(cluster gvkey) nocons
```

| Coef. | Std. Err. | t | P>|t| | [95% Conf. Interval] |
|-------|-----------|---|-----|------------------------|
| mtb   | -.0149454 | .0011126 | -13.43 | .000 | -.0171263 | -.0127645 |
| tangib| .1991992  | .0099499 | 20.02  | .000 | .1796956 | .2187028  |
| logsales | .0394179  | .0092385 | 15.76  | .000 | .0345138 | .0443221  |
| oi    | -.2411662 | .0092385 | -26.10 | .000 | -.2592753 | -.2230571 |

(Std. Err. adjusted for 10,796 clusters in gvkey)

Compare these results with those of xtewreg, assuming again that mtb and tangib are measured with error:

```
xtewreg lever mtb tangib logsales oi, maxdeg(5) mismeasured(2) nocons
```

| Coef. | Std. Err. | z  | P>|z| | [95% Conf. Interval] |
|-------|-----------|----|-----|------------------------|
| mtb   | -.0318794 | .0044092 | -7.23 | .000 | -.0405212 | -.0232376 |
| tangib| 1.207097  | .0378421 | 31.90 | .000 | 1.132928  | 1.281266  |
| logsales | .0579218  | .0025019 | 15.76 | .000 | .0507467 | .0650976  |
| oi    | -.0566342 | .0092385 | -6.10 | .000 | -.0879328 | -.0253356 |

$\tau_1^2$: 0.478 (0.080)  
$\tau_2^2$: 0.186 (0.010)  
Sargan-Hansen J statistic: 245.977 (p=0.000, d=20)

The $J$ statistic for the test of overidentifying restrictions is quite large. This result indicates a violation of one of the conditions in the assumptions in section 2.1, with the likely culprit being a regression error $u_i$ that is independent of the regressors $x_i$ and $z_i$. The leverage regression we have chosen as an example, although widely used, likely suffers from problems of omitted variables.
Repeating the estimation with `maxdeg(8)`—therefore using all moment conditions up to degree eight—yields the following:

```
. xtewreg lever mtb tangib logsales oi, maxdeg(8) mismeasured(2) nocons
Problem structure different from last executed. Rebuilding problem.
8(2) EIV results
N = 121733
Rho^2 = 0.204 (0.008)
```

```
|        | Coef.  | Std. Err. | z     | P>|z|  | [95% Conf. Interval] |
|--------|--------|-----------|-------|-----|---------------------|
| lever  |        |           |       |     |                     |
| mtb    | -0.0241135 | 0.0008971 | -26.88 | 0.000 | -0.0258717 -0.0223552 |
| tangib | 1.264688 | 0.0079716 | 158.65 | 0.000 | 1.249064 1.280312   |
| logsales | 0.0599253 | 0.0079716 | 158.65 | 0.000 | 0.0527266 0.0671239 |
| oi     | -0.061634 | 0.0119372 | -5.16  | 0.000 | -0.0850305 -0.0382375 |
```

```
Tau^2: 0.611 (0.066)
 Tau^2: 0.179 (0.008)
Sargan-Hansen J statistic: 1289.990 (p=0.000, d=96)
```

Note the message printed by `xtewreg` regarding rebuilding the problem; the parameters of the problem are different from those used during the latest call to `xtewreg`.

### 4.1 Using bootstrap with xtewreg

To calculate the bootstrapped critical values for the test statistics, we need to recenter the moment conditions for every bootstrap iteration (see Hall and Horowitz [1996] for details). To do so, we first execute `xtewreg` on the entire dataset while specifying `centmom(set)`.

```
. bootstrap t_mtb=(_b[mtb]/el(e(serr),1,1)) t_tangib=(_b[tangib]/el(e(serr),2,1))
   > t_logsales=(_b[logsales]/el(e(serr),3,1))
   > t_oi=(_b[oi]/el(e(serr),4,1)), rep(100) seed(1337) cluster(gvkey) notable:
   > xtewreg lever mtb tangib logsales oi, maxdeg(5) mismeasured(2) centmom(use)
   > nocons
   (running xtewreg on estimation sample)
```

```
Bootstrap replications (100)
1 2 3 4 5
.................................................. 50
.................................................. 100
Bootstrap results
Number of obs = 121,733
Replications = 100
```

```
command: xtewreg lever mtb tangib logsales oi, maxdeg(5) mismeasured(2)
```

```
> centmom(use)
```

```
t_mtb: _b[mtb]/el(e(serr),1,1)
t_tangib: _b[tangib]/el(e(serr),2,1)
t_logsales: _b[logsales]/el(e(serr),3,1)
t_oi: _b[oi]/el(e(serr),4,1)
```
High-order cumulants and moments

```
. estat bootstrap, p
Bootstrap results                                      Number of obs = 121,733
Replications = 100
command: xtewreg lever mtb tangib logsales oi, maxdeg(5) mismeasured(2)
> centmom(use) nocons
    t_mtb: _b[mtb]/el(e(serr),1,1)
    t_tangib: _b[tangib]/el(e(serr),2,1)
    t_logsales: _b[logsales]/el(e(serr),3,1)
    t_oi: _b[oi]/el(e(serr),4,1)
(Replications based on 10,796 clusters in gvkey)

                          Observed Bootstrap
                          Coef.  Bias  Std. Err.  [95% Conf. Interval]
    t_mtb   -7.2302642  1.787705 .96365039  -7.276362  -3.500391 (P)
    t_logsales  15.821917 -4.443195 .90578445   9.769751  13.05211 (P)
    t_oi   -3.5465206  .6906451 .71034705  -4.210805  -1.481791 (P)
```

(P) percentile confidence interval

We use the bootstrap to calculate the critical value for the $t$ statistic, because it is asymptotically pivotal (see Horowitz [2001] for details). Furthermore, we use the percentile method to derive confidence intervals and $p$-values (by `estat bootstrap, p` after executing `bootstrap`).

5 References


About the authors

Timothy Erickson is a research economist at the Bureau of Labor Statistics in the Division of Price and Index Number Research.

Robert Parham is a PhD student at the University of Rochester.

Toni M. Whited is the Dale L. Dykema Professor of Business Administration at the University of Michigan and a research associate at the National Bureau of Economic Research.
Capturing a Stata dataset and releasing it into REDCap

Seth T. Lirette
Department of Data Science
University of Mississippi Medical Center
Jackson, MS
and
Department of Biostatistics
University of Alabama at Birmingham
Birmingham, AL
slirette2@umc.edu

Samantha R. Seals
Department of Mathematics and Statistics
University of West Florida
Pensacola, FL

Chad Blackshear
Department of Data Science
University of Mississippi Medical Center
Jackson, MS

Warren May
School of Health Related Professions
University of Mississippi Medical Center
Jackson, MS

Abstract. With technology advances, researchers can now capture data using web-based applications. One such application, Research Electronic Data Capture (REDCap), allows for data entry from any computer with an Internet connection. As the use of REDCap has increased in popularity, we have observed the need to easily create data dictionaries and data collection instruments for REDCap. The command presented in this article, redcapture, demonstrates one method to create a REDCap-ready data dictionary using a loaded Stata dataset, illustrated by examples of starting from an existing dataset or completely starting from scratch.

Keywords: dm0091, redcapture, reproducibility, REDCap, web based, data entry

1 Introduction

Research Electronic Data Capture (REDCap) is a secure, web-based application designed to support data capture for research studies. It provides 1) an intuitive interface for validated data entry; 2) audit trails for tracking data manipulation and export procedures; 3) automated export procedures for seamless data downloads to common statistical packages; and 4) procedures for importing data from external sources (Harris et al. 2009). While developed at Vanderbilt, REDCap is hosted at the individual institution level; that is, all data collected using REDCap are stored on a local server.

© 2017 StataCorp LLC dm0091
REDCap has many desirable properties for researchers interested in electronic data collection. Because the data are stored on a centrally managed server at the local host institution, researchers do not have to worry about losing the data file (for example, an Excel file maintained on a USB flash drive), and the security of the data can be more easily ensured. Clinicians have begun to recognize the need for securing patient data (Anand and Spalding 2015; Niehaus, Boimbo, and Akuthota 2015; and Morinville et al. 2014). Further, REDCap has double data-entry functionality, including a data comparison tool that allows users to select the correct data-entry mode for the type of study using article forms or direct entry. Other useful features of REDCap are its ease of use and intuitive interface. With minimal training, staff can enter data into the system consistently. For relatively simple data-entry forms, REDCap offers a form development platform that is straightforward and easy to use without any knowledge of database programming. Certain functions, such as audit trails, are automatically activated when the project’s database is put into production.

In addition to data collected within an institution, REDCap allows data capture among multiple sites using a common web-based interface. As an example, consider a multicenter clinical trial where all sites enter data into the same database through REDCap. When the statistician needs access to the data to create the data safety monitoring board report, the data can be exported for immediate use in several softwares: Excel, SPSS, SAS, R, and Stata. These exported data are formatted specifically for the statistical software of choice and save the statistician from extensive data management. Other useful features include a data import tool, a randomization tool, and a longitudinal study planner with a calendar function to inform the data collector of the progress of each individual subject.

Because REDCap is becoming more popular, we discovered a need to easily create a REDCap instrument from an existing dataset. Researchers either want to import their existing data into REDCap or want to collect the same data as a previous project. Outside of basic single-study clinical research, longitudinal cohort studies may be interested in collecting future data in REDCap. In that case, studies often reuse questionnaires across visits, so it may be helpful for statisticians to be able to use existing data to create a data dictionary to upload to REDCap. For example, most large epidemiologic studies use standard instruments at each clinic exam to take blood pressure and medical health history. If another clinic visit is funded, these, among other, instruments will be repeated and REDCap is an attractive choice in terms of ease of use, security, accessibility, and cost for the continuation.

Some practical published examples are warranted at this point. Shared libraries are very efficient, both for multisite and multivisit studies and for sharing across studies. REDCap shines here (Obeid et al. 2013). REDCap offers low-cost functionality in developing countries (Tuti et al. 2016). Finally, studies have demonstrated the ease-of-use capabilities of REDCap (Pang et al. 2014; Franklin, Guidry, and Brinkley 2011).

Behind any good study is a well-designed data structure. REDCap uses the data dictionary to allow professional databases to be built with minimal programming but accounts for different variable types and limiters, variable and value labels, skip patterns,
and on-screen appearance of the data-entry screens. If starting a new project, one can build the dictionary within REDCap, but REDCap also offers one the ability to build the data dictionary externally and upload it to create the necessary forms. Further, REDCap has built-in functionality for downloading existing data dictionaries once they are created. However, to upload data from another data source, one must first create the REDCap data dictionary.

By using existing data to create a data dictionary for REDCap, statisticians can create the framework for the REDCap instrument by first uploading the data dictionary. This can then be used to either upload existing data or, in our case, easily generate forms for our new study visit based on existing data currently stored in Stata data files. These goals are easily accomplished using redcapture.

2 The redcapture command

The essential starting block for a new project in REDCap is a data dictionary. The redcapture command seamlessly automates this process by using the loaded Stata dataset instead of the point-and-click interface of the web-based REDCap. In essence, this data dictionary is merely a CSV file that contains a certain set of column headers and formatting rules. The user can then simply load the CSV file into REDCap.

The heart of redcapture is built around the text(), dropdown(), and radio() options. At least one of these must be specified for redcapture to run, and anything listed in the main varlist must be listed in one of these three options’ varlists. We do note that check boxes are also available in the web version of REDCap; however, we are attune to the headaches and inconsistencies involved with check-box answers and decided to omit them as an option in redcapture. Also note that for the variables declared to be drop downs or radios (that is, categorical in nature), redcapture searches the dataset and compiles the form based solely upon each unique label of the categorical variables. Therefore, first, drop-down or radio variables need to be numerical with value labels attached. Second, all values the user wishes to include in the data dictionary need to be defined in the value label, although that actual value does not need to be present in the loaded data.

REDCap also offers the ability to require validation for variables entered as text fields, and redcapture provides this option as well. The user simply declares validation variables in the validate() option. This option requires the declared variables to also be declared as text variables in the text() option. In addition to validating the variables, REDCap and redcapture offer the option of providing required minimums and maximums to the validated variables, done in redcapture through the validmin() and validmax() options. If the user wishes to omit validation minimums and maximums for any or all the validation variables, none should be entered into the corresponding location in the validmin() and validmax() options. REDCap version 6.10.1 has 24 different validation types. Each is listed in table 1 with an example of a minimum and maximum. Blanks indicate minimums and maximums are not legitimate for that particular validation type. Examples will follow.
As one last option, questions with the same responses can be grouped together in REDCap using the matrix functionality. An example would be 25 survey questions each with a Likert-type response. `redcapture` allows for up to 10 different matrices.

<table>
<thead>
<tr>
<th><code>validtypes</code></th>
<th>Example min</th>
<th>Example max</th>
</tr>
</thead>
<tbody>
<tr>
<td>date_dmy</td>
<td>01/01/1900</td>
<td>12/31/3000</td>
</tr>
<tr>
<td>date_mdy</td>
<td>01/01/1900</td>
<td>12/31/3000</td>
</tr>
<tr>
<td>date_ymd</td>
<td>01/01/1900</td>
<td>12/31/3000</td>
</tr>
<tr>
<td>datetime_dmy</td>
<td>01/01/1900 00:01</td>
<td>12/31/3000 23:59</td>
</tr>
<tr>
<td>datetime_mdy</td>
<td>01/01/1900 00:01</td>
<td>12/31/3000 23:59</td>
</tr>
<tr>
<td>datetime_ymd</td>
<td>01/01/1900 00:01</td>
<td>12/31/3000 23:59</td>
</tr>
<tr>
<td>datetime_seconds_dmy</td>
<td>01/01/1900 00:00:01</td>
<td>12/31/3000 23:59:59</td>
</tr>
<tr>
<td>datetime_seconds_mdy</td>
<td>01/01/1900 00:00:01</td>
<td>12/31/3000 23:59:59</td>
</tr>
<tr>
<td>datetime_seconds_ymd</td>
<td>01/01/1900 00:00:01</td>
<td>12/31/3000 23:59:59</td>
</tr>
<tr>
<td>email</td>
<td></td>
<td></td>
</tr>
<tr>
<td>integer</td>
<td>1</td>
<td>100</td>
</tr>
<tr>
<td>alpha_only</td>
<td></td>
<td></td>
</tr>
<tr>
<td>mrn_10d</td>
<td></td>
<td></td>
</tr>
<tr>
<td>number</td>
<td>1</td>
<td>100</td>
</tr>
<tr>
<td>number_1dp</td>
<td>0</td>
<td>100.1</td>
</tr>
<tr>
<td>number_2dp</td>
<td>0</td>
<td>10</td>
</tr>
<tr>
<td>number_3dp</td>
<td>0</td>
<td>2.012</td>
</tr>
<tr>
<td>number_4dp</td>
<td>1.3132</td>
<td>5.3216</td>
</tr>
<tr>
<td>phone</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ssn</td>
<td></td>
<td></td>
</tr>
<tr>
<td>time</td>
<td>01:23</td>
<td>14:55</td>
</tr>
<tr>
<td>time_mm_ss</td>
<td>01:23:01</td>
<td>59:59:00</td>
</tr>
<tr>
<td>vmrn</td>
<td></td>
<td></td>
</tr>
<tr>
<td>zipcode</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### 2.1 Syntax

```
redcapture varlist, file(string) form(string) [text(varlist) dropdown(varlist) radio(varlist) header(string) validate(varlist) validtype(validtypes)
validmin(minlist) validmax(maxlist) matrix1(varlist) matrix2(varlist) matrix3(varlist) matrix4(varlist) matrix5(varlist) matrix6(varlist) matrix7(varlist) matrix8(varlist) matrix9(varlist) matrix10(varlist) ]
```
2.2 Options

`file(string)` is required and indicates the filename for the CSV file that is output to the active directory. Subdirectories are also supported. This file must be closed before running `redcapture`.

`form(string)` is required and indicates the form name for later use when loading the instrument into REDCap. This is a REDCap requirement.

Note: At least one of the options `text()`, `dropdown()`, or `radio()` is required.

`text(varlist)` contains a list of all variables the user wishes to be input as text fields.

`dropdown(varlist)` contains a list of all variables the user wishes to be input as drop-down lists.

`radio(varlist)` contains a list of all variables the user wishes to be input as radio buttons.

`header(string)` gives a header or title to the REDCap instrument.

`validate(varlist)` contains a list of all variables the user wants to require validation. Only text fields can be validated. Drop downs and radio buttons are inherently validated by only offering valid choices. This option can be left blank, but validating text fields is usually a good idea to ensure quality data.

`validtype(validtypes)` is required for each variable declared in the `validate()` option and must be listed in the same order as the `varlist` in the `validate()` option. REDCap currently has 24 different validation types. The full list is in table 1 above.

`validmin(minlist)` contains the minimum values allowed for the validation variables and must be listed in the same order as the `varlist` in the `validate()` option. Only `validtypes` dates, datetimes, integers, numbers, and times use minimum and maximum values. The user is not required to set minimums and maximums and must type `none` where this is desired. `none` must be typed for `validtypes` that are not dates, datetimes, integers, numbers, or times. Dates must be in mm/dd/yyyy format and times in hh:mm:ss format. More details are below.

`validmax(maxlist)` contains the maximum values allowed for the validation variables and must be listed in the same order as the `varlist` in the `validate()` option. Only `validtypes` dates, datetimes, integers, numbers, and times use minimum and maximum values. The user is not required to set minimums and maximums and must type `none` where this is desired. `none` must be typed for `validtypes` that are not dates, datetimes, integers, numbers, or times. Dates must be in mm/dd/yyyy format and times in hh:mm:ss format. More details are below.

`matrix#(varlist)` specifies variables the user wishes to cluster. REDCap allows multiple variables to be combined into a single “data matrix”. This is most useful for things like combining multiple variables in a Likert scale format into a single cluster in the instrument. `redcapture` allows up to 10 of these matrices.
3 Examples

. *Setting up the data
. clear
. input str9 id consented age race sex str10 bdate sbp dbp happy1 happy2 happy3
> str4 comment
> id consented age race sex bdate
> sbp dbp happy1 happy2 happy3 comment
  1. "1" 1 54 1 1 "1953-06-21" 110 80 2 2 1 "none"
  2. "2" 1 64 2 2 "1943-05-19" 140 90 3 4 2 "none"
  3. "3" 0 85 3 2 "1929-04-03" 160 85 1 5 3 "none"
  4. "4" 0 80 1 1 "1934-10-23" 120 95 4 3 4 "none"
  5. "5" 1 60 2 2 "1947-09-18" 115 70 5 1 5 "none"
. end

. *Fixing the birth date
. generate bdate2=date(bdate,"YMD")
. format bdate2 %td
. drop bdate
. rename bdate2 bdate
. order bdate, after(sex)

. *Variable labels
. label variable id "Participant ID"
. label variable consented "Is a consent document on file?"
. label variable age "How old were you on your last birthday?"
. label variable race "What is your race?"
. label variable sex "What is your sex?"
. label variable bdate "What is your date of birth?"
. label variable sbp "What was your last known systolic blood pressure?"
. label variable dbp "What was your last known diastolic blood pressure?"
. label variable happy1 "The staff greeted me in a professional and
courteous manner."
. label variable happy2 "The waiting time to see a doctor was satisfactory."
. label variable happy3 "I would return to this hospital."
. label variable comment "Comments"

. *Value Labels
. label define ynlab 0 "No" 1 "Yes", replace
. label define rlab 1 "Caucasian" 2 "African American" 3 "Other", replace
. label define llab 1 "Strongly Agree" 2 "Agree" 3 "Neither Agree nor Disagree" 4 "Disagree" 5 "Strongly Disagree", replace

This first example is a case where a researcher is starting a study from scratch. He or she desires to include variables for an id variable, whether the participant consented, age, race, sex, birth date, systolic blood pressure, diastolic blood pressure, three Likert-scale variables relating to satisfaction, and an open comment field. The above shows
how the researcher can initiate such a dataset completely within a Stata do-file. Notice that variables the researcher will later declare to be drop downs and radios (\texttt{consented, race, happy1, happy2, and happy3}) have all values he or she wishes to make available. The rest of the setup regards attaching variable labels and value labels.

\begin{verbatim}
. redcapture *, file(example) form(example_form) header(Example)
> text(id age sex bdate sbp dbp comment)
> dropdown(consented race)
> radio(happy1 happy2 happy3)
> validate(id bdate dbp comment)
> validtype(ssn date_ymd integer alpha_only)
> validmin(none 1/1/1900 20 none)
> validmax(none 12/31/2014 200 none)
> matrix1(happy1 happy2 happy3)
\end{verbatim}

REDcap data dictionary \texttt{example.csv} has been created in the active directory.

Next comes the \texttt{redcapture} call. We wish to include all variables, with \texttt{consented} and \texttt{race} chosen from drop-down menus, \texttt{happy1}, \texttt{happy2}, and \texttt{happy3} being radio buttons collected into a matrix, and the rest typed in as text fields. We have chosen to validate \texttt{id} as a social security number (not a good practice), \texttt{bdate} as a \texttt{date_ymd} date with a minimum of 1/1/1900 and maximum of 12/31/2014, \texttt{dbp} as an integer with a minimum of 20 and maximum of 200, and \texttt{comment} as a free-text field. We have chosen not to validate \texttt{age}, \texttt{sex}, and \texttt{sbp} (also not a good practice). Once this runs, we get a CSV file whose first few columns are displayed as a screenshot in figure 1. This file can then be uploaded into REDCap.

\begin{table}[h]
\centering
\begin{tabular}{ |c|c|c|c|c| }
\hline
Variable & FormName & Section & FieldType & FieldLabel & Choices/Calculations/Slider \\
\hline
id & example_example & & text & Participant ID & \\
\hline
consented & example_example & & dropdown & is a consent document on file? & 0, No | 1, Yes \\
\hline
age & example_example & & text & How old were you on your last birthday? & \\
\hline
race & example_example & & dropdown & What is your race? & 1, Caucasian | 2, African \\
\hline
sex & example_example & & text & What is your sex? & \\
\hline
bdate & example_example & & text & What is your date of birth? & \\
\hline
sbp & example_example & & text & What was your last known systolic blood pressure? & \\
\hline
dbp & example_example & & text & What was your last known diastolic blood pressure? & \\
\hline
happy1 & example_example & & radio & The staff greeted me in a professional and courteous manner | Strongly Agree | Agree \\
\hline
happy2 & example_example & & radio & The waiting time to see a doctor was satisfactory | Strongly Agree | Agree \\
\hline
happy3 & example_example & & radio & I would return to this hospital | Strongly Agree | Agree \\
\hline
comment & example_example & & text & Comments & \\
\hline
\end{tabular}
\caption{Screenshot of CSV file for example dataset}
\end{table}
Suppose we would like to create a new REDCap data dictionary from an already existing dataset, such as the extract of the U.S. National Longitudinal Survey for employed women in 1988 (a Stata installed dataset). We simply load the dataset, set file and form names, declare radio, drop-down, and text variables, and set all text variables validation types, minimums, and maximums (a good practice). Upload the data dictionary created by redcapture, and we are now ready to start data collection for our study.

```
. sysuse nlsw88, clear
(NLSW, 1988 extract)
. redcapture * , file(nlsw_instrument) form(nlsw) header("NLSW Instrument")
  > radio(race married collgrad smsa)
  > dropdown(industry occupation union)
  > text(idcode age never_married grade south c_city wage hours ttl_exp tenure)
  > validate(age never_married grade south c_city wage hours ttl_exp tenure)
  > validtype(integer integer integer integer integer number integer
  >       number_4dp number_2dp)
  > validmin(20 0 0 0 0 0 0 0)
  > validmax(none 1 1 1 none 80 30 none)
REDcap data dictionary nlsw_instrument.csv has been created in the active
  > directory
```

### 4 References


**About the authors**

Seth Lirette is a biostatistician at the University of Mississippi Medical Center and a PhD student at the University of Alabama at Birmingham.

Samantha Seals is a visiting assistant professor of statistics at the University of West Florida.

Chad Blackshear is a biostatistician at the University of Mississippi Medical Center.

Warren May is a professor of biostatistics at the University of Mississippi Medical Center.
Spatial panel-data models using Stata

Federico Belotti
Centre for Economic and International Studies
University of Rome Tor Vergata
Rome, Italy
federico.belotti@uniroma2.it

Gordon Hughes
University of Edinburgh
Edinburgh, UK
g.a.hughes@ed.ac.uk

Andrea Piano Mortari
Centre for Economic and International Studies
University of Rome Tor Vergata
Rome, Italy
andrea.piano.mortari@uniroma2.it

Abstract. *xsmle* is a new user-written command for spatial analysis. We consider the quasi–maximum likelihood estimation of a wide set of both fixed- and random-effects spatial models for balanced panel data. *xsmle* allows users to handle unbalanced panels using its full compatibility with the *mi* suite of commands, use spatial weight matrices in the form of both Stata matrices and *spmat* objects, compute direct, indirect, and total marginal effects and related standard errors for linear (in variables) specifications, and exploit a wide range of postestimation features, including the panel-data case predictors of Kelejian and Prucha (2007, *Regional Science and Urban Economics* 37: 363–374). Moreover, *xsmle* allows the use of *margins* to compute total marginal effects in the presence of nonlinear specifications obtained using *factor variables*. In this article, we describe the command and all of its functionalities using simulated and real data.

Keywords: st0470, xsmle, spatial analysis, spatial autocorrelation model, spatial autoregressive model, spatial Durbin model, spatial error model, generalized spatial panel random-effects model, panel data, maximum likelihood estimation

1 Introduction

It is widely recognized that sample data collected from geographically close entities are not independent but spatially correlated, which means observations of closer units tend to be more similar than observations of further units (Tobler 1970).\(^1\) Spatial clustering, or geographic-based correlation, is often observed for economic and sociodemographic variables such as unemployment, crime rates, house prices, per-capita health expenditures, and so on (Ollé 2003, 2006; Moscone and Knapp 2005; Revelli 2005; Kostov 2009; Elhorst and Fréret 2009; Elhorst, Piras, and Arbia 2010; Moscone, Tosetti, and Vittadini 2012). Theoretical models usually recognize the existence of spatial spillovers,\(^1\) Note that nonspatial structured dependence may also be observed. In these cases, measures of geographical proximity are replaced by measures of similarity, allowing one to investigate peer effects through social or industrial networks (LeSage and Pace 2009; Bramouillé, Djebbari, and Fortin 2009).
which decline as distance between units increases; empirically, spatial panel-data models have become a popular tool for measuring such spillovers.

As far as we know, while both R and MATLAB offer a large suite of functions to estimate spatial panel-data models (Millo and Piras 2012; LeSage and Pace 2009)—with the notable exception represented by the accompanying code of Kapoor, Kelejian, and Prucha (2007)—Stata’s capabilities include a wide set of commands designed to deal only with cross-sectional data (Drukker et al. 2013; Drukker, Prucha, and Raciborski 2013a,b). We developed the \texttt{xsmle} command to estimate a wide range of spatial panel-data models using Stata. In particular, \texttt{xsmle} allows users to estimate both fixed-effects (FE) and random-effects (RE) spatial autoregressive (SAR) models, spatial Durbin models (SDMs), spatial error models (SEMs), FE spatial autocorrelation (SAC) models, and generalized spatial RE (GSPRE) models. For spatial autoregressive (SAR) and SDMs with FE, \texttt{xsmle} also allows a dynamic specification by implementing the bias-corrected maximum likelihood approach described in Yu, de Jong, and Lee (2008). Among other interesting features, \texttt{xsmle} allows users to i) use spatial weight matrices created through the \texttt{spmat} command of Drukker et al. (2013); ii) compute direct, indirect, and total marginal effects; iii) compute both clustered and Driscoll–Kraay standard errors; iv) test whether an FE or RE model is appropriate using a robust Hausman test; v) and exploit a wide range of predictors, extending to the panel-data case estimators of Kelejian and Prucha (2007).

The rest of this article is organized as follows. In section 2, we present a brief review of spatial panel-data models that can be estimated with \texttt{xsmle}. Section 3 documents \texttt{xsmle} syntax and its main options, while section 4 illustrates its main features using simulated and real datasets. The last section concludes.

\section{Spatial panel-data models}

Here we summarize spatial panel-data models, focusing on those that can be estimated with \texttt{xsmle}. Note that \texttt{xsmle} is primarily designed to deal with balanced panel data in which \(n\) units are observed for exactly \(T\) periods. We will turn to unbalanced panels in section 4.1, where we show how to handle these by exploiting the official Stata \texttt{mi} suite of commands.

In what follows, we denote the \(n \times 1\) column vector of the dependent variable with \(y_t\) and the \(n \times k\) matrix of regressors with \(X_t\), where \(t = 1, \ldots, T\) indicating time periods. For each cross-section, \(W\) is the \(n \times n\) matrix describing the spatial arrangement of the \(n\) units, and each entry \(w_{ij} \in W\) represents the spatial weight associated to units \(i\).
and $j$. To exclude self-neighbors, the diagonal elements $w_{ii}$ are conventionally set equal to zero. Note that `xsmle` allows the use of two different formats for the weight matrix; that is, $W$ can be a Stata matrix or an `spmat` object. This allows the user to leverage the capabilities of other Stata commands that allow the creation and management of weight matrices, such as `spmat`, `spatwmat` (Pisati 2001), or `spwmatrix` (Jeanty 2010). Furthermore, `xsmle` automatically takes care of the longitudinal nature of the data. Hence, users need to provide only the cross-sectional $n \times n$ weight matrix to fit a specific model.

`xsmle` allows users to fit the following models:

**SAR model.** The basic equation for the SAR model is

$$y_t = \rho Wy_t + X_t\beta + \mu + \epsilon_t \quad t = 1 \ldots, T$$

It is assumed that $\mu \sim N(0, \sigma^2_\mu)$ in the RE case, while $\mu$ is a vector of parameters to be estimated in the FE variant. The standard assumptions—that $\epsilon_{it} \sim N(0, \sigma^2_\epsilon)$ and $E(\epsilon_{it}\epsilon_{js}) = 0$ for $i \neq j$ or $t \neq s$—apply in this case.

**SDM.** This model is a generalization of the SAR model, which also includes spatially weighted independent variables as explanatory variables,

$$y_t = \rho Wy_t + X_t\beta + WZ_t\theta + \mu + \epsilon_t$$

where $M$ is a matrix of spatial weights that may or may not be equal to $W$. This model can be further generalized by using $Z_t \neq X_t$.

**SAC model.** This model (alternatively referred to as the SAR with spatially autocorrelated errors, SAC) extends the SAR model by allowing for a spatially autocorrelated error,

$$y_t = \rho Wy_t + X_t\beta + \mu + \nu_t$$

where $M$ is a matrix of spatial weights that may or may not be equal to $W$. The literature focuses on the FE variant of this specification because the RE variant can be written as a special case of the SAR specification.

---

2. Two sources of locational information are generally exploited. First, the location in Cartesian space (for example, latitude and longitude) is used to compute distances among units. Second, the knowledge of the size and shape of observational units allows the definition of contiguity measures. For example, one can determine which units are neighbors in the sense that they share common borders. Thus the former source points toward the construction of spatial distance matrices, while the latter is used to build spatial contiguity matrices. Note that the aforementioned sources of locational information are not necessarily different. For instance, a spatial contiguity matrix can be constructed by defining units as contiguous when they lie within a certain distance; on the other hand, by computing the coordinates of the centroid of each observational unit, one can obtain approximated spatial distance matrices using the distances between centroids. More details are available in LeSage and Pace (2009).
**Spatial panel-data models using Stata**

SEM. The SEM focuses on SAC in the error term, as in

\[
\begin{align*}
y_t &= X_t \beta + \mu + \nu_t \\
\nu_t &= \lambda M \nu_t + \epsilon_t
\end{align*}
\]

This is a special case of the SAC model, but it is also a special case of the SDM.

GSPRE. This model can be represented as

\[
\begin{align*}
y_t &= X_t \beta + \mu + \nu_t \\
\nu_t &= \lambda M \nu_t + \epsilon_t \\
\mu &= \phi W \mu + \eta
\end{align*}
\]

This is a generalization of the SEM, in which the panel effects, represented by the vector \( \mu \), are spatially correlated. The vectors \( \mu \) and \( \epsilon_t \) are assumed to be independently normally distributed errors, so the model is necessarily an RE specification with \( \mu = (I - \phi W)^{-1} \eta \) and \( \nu_t = (I - \lambda W)^{-1} \epsilon_t \). There are various special cases of the general specification, with (a) \( \lambda \neq \phi \neq 0 \), (b) \( \lambda = 0 \), (c) \( \phi = 0 \), (d) \( \lambda = \phi \).

In addition to the distinction between the FE and RE, there is a separate distinction between static and dynamic specifications. The aforementioned models are all static in that they involve contemporaneous values of the dependent and independent variables. **xsmle** also allows the estimation of SAR and SDM models, such as

\[
y_t = \tau y_{t-1} + \psi W y_{t-1} + \rho W y_t + X_t \beta + \mu + \epsilon_t
\]

where the lagged (in time) dependent variable or the lagged (in both time and space) dependent variable can be included in the specification.

### 2.1 Estimation

Various methods of fitting spatial panel models have been proposed. Broadly, they fall into two categories: i) generalized method of moments and ii) quasi–maximum likelihood (QML) estimators. All models that can be fit using **xsmle** fall into the second category. A synopsis guide with all estimable models and their features is reported in table 1.\(^3\) The gain from programming gradients is large, so \( v1 \) evaluators are used for all but one of the specifications. The exception is the RE SEM, whose likelihood function involves a transformation using the Cholesky factors of a rather complicated matrix containing the parameters to be estimated, so the matrix differentiation is extremely messy.

---

\(^3\) Elhorst (2010a) suggests that the computation time required to carry out full maximum likelihood estimation can be reduced by transforming variables in a way that permits the likelihood function to be concentrated so the estimation can be carried out in two steps. In translating his routines to Mata, we found that using a concentrated likelihood tended to increase both the number of iterations and the time required to fit the models.
Table 1. A summary of \texttt{xsmle} estimation capabilities

<table>
<thead>
<tr>
<th>Model</th>
<th>Estimation method</th>
<th>Time FE</th>
<th>Individual FE</th>
<th>Random effects</th>
<th>\texttt{wmatrix()}</th>
<th>\texttt{ematrix()}</th>
<th>\texttt{dmatrix()}</th>
<th>Dynamic</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAR</td>
<td>QML</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>SEM</td>
<td>QML</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SAC</td>
<td>QML</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SDM</td>
<td>QML</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>GSPRE</td>
<td>QML</td>
<td></td>
<td></td>
<td></td>
<td>x</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Spatial panel-data models using Stata

For dynamic models, that is, those including a time-lagged dependent variable, a time and space-lagged dependent variable, or both, `xsmle` implements only the FE variant of the SAR and SDM models using the bias-corrected QML approach described by Yu, de Jong, and Lee (2008), which is consistent when both $n \to \infty$ and $T \to \infty$. The command starts by constructing maximum likelihood estimates, treating the aforementioned lagged variables as exogenous regressors. Bias corrections are then computed for each of the coefficients and used to adjust the initial maximum likelihood estimates.

For each model, the default asymptotic variance–covariance (VC) matrix of the coefficients is obtained from the observed information matrix.\textsuperscript{4} Angrist and Pischke (2009) emphasize the potential dangers of this approach for datasets for which there may be unknown serial correlation in the errors within each panel unit. To our knowledge, there are no established methods of computing robust standard errors for spatial panel-data models. Mimicking the derivation of robust standard errors for nonspatial panel models, `xsmle` implements two different approaches: i) one-way clustered standard errors and ii) Driscoll and Kraay (1998) standard errors. As in other panel-data official Stata commands, specifying `vce(robust)` is equivalent here to specifying `vce(cluster panelvar)`, where `panelvar` is the variable that identifies the panels.

As for the Driscoll–Kraay standard errors, the `xsmle` implementation is based on Hoechle's (2007) `xtscc` command. The Driscoll–Kraay approach provides a specific variant of the Newey–West robust covariance estimator computed using the Bartlett kernel and a time series of scores’ cross-sectional averages.\textsuperscript{5}

In our test runs, the differences between the asymptotic and robust standard errors are usually small, but we have not focused on cases with small values of $n$ and $T$. In principle, it would be useful to include a bootstrap estimator for the VC matrix. Unfortunately, there is a major barrier to applying standard bootstrap methods in this case. The crucial assumption for resampling is that the errors for the observations or units from which each sample is drawn should be independent. For panel or clustered data, this means the resampling is based on panel units or clusters. For spatial panels, our base model assumes the observations for different panel units are correlated over space for any given period $t$. It follows that resampling based on panel units cannot be reconciled with the hypothesis of spatial interactions in the relationships of interest. As an alternative, we could use time periods as the resampling unit, but this will be valid only if there is no serial correlation within panels. Further, for many applications of spatial panel estimation, the value of $T$ is considerably smaller than $n$, so large sample assumptions of bootstrap statistics do not apply. Statisticians have developed bootstrap methods for spatial data but at the cost of imposing substantial restrictions on the extent of spatial interactions that can be examined. The methods have tended to focus on regular lattices, but they can be applied to spatial data for fairly small economic units such as counties and labor market areas.

---

\textsuperscript{4} A variant obtained from the outer product of the gradients is also available by specifying `vce(opg)`.

\textsuperscript{5} The bandwidth for the kernel is specified with a default value of $\text{floor}\{4(T/100)^{2/9}\}$ if no value is specified.
Direct, indirect, and total marginal effects

Because spatial regression models exploit the complicated dependence structure between units, the effect of an explanatory variable's change for a specific unit will affect the unit itself and, potentially, all other units indirectly. This implies the existence of direct, indirect, and total marginal effects. With the exception of the SEM and the GSPRE models, and only if the effects option is specified, these effects are computed using the formulas reported in table 2. The command automatically distinguishes between short- and long-run marginal effects when a dynamic spatial model is fit.
**Table 2. Direct, indirect, and total effects**

<table>
<thead>
<tr>
<th>Type of model</th>
<th>Short-term direct effect</th>
<th>Short-term indirect effect</th>
<th>Long-term direct effect</th>
<th>Long-term indirect effect</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Static</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SDM</td>
<td>none</td>
<td>none</td>
<td>((I - \rho W)^{-1} \times (\beta_k I + \theta_k W))^{\bar{d}}</td>
<td>((I - \rho W)^{-1} \times (\beta_k I + \theta_k W))^{\bar{r}}sum</td>
</tr>
<tr>
<td>SEM</td>
<td>none</td>
<td>none</td>
<td>(\beta_k)</td>
<td>none</td>
</tr>
<tr>
<td>SAR</td>
<td>none</td>
<td>none</td>
<td>((I - \rho W)^{-1} \times (\beta_k I))^{\bar{d}}</td>
<td>((I - \rho W)^{-1} \times (\beta_k I))^{\bar{r}}sum</td>
</tr>
<tr>
<td>SAC</td>
<td>none</td>
<td>none</td>
<td>((I - \rho W)^{-1} \times (\beta_k I))^{\bar{d}}</td>
<td>((I - \rho W)^{-1} \times (\beta_k I))^{\bar{r}}sum</td>
</tr>
<tr>
<td><strong>Dynamic</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SDM</td>
<td>((I - \rho W)^{-1}) \times (\beta_k I + \theta_k W))^{\bar{d}}</td>
<td>((I - \rho W)^{-1}) \times (\beta_k I + \theta_k W))^{\bar{r}}sum</td>
<td>((1 - \tau)I - (\rho + \psi)W)^{-1} \times (\beta_k I + \theta_k W))^{\bar{d}}</td>
<td>((1 - \tau)I - (\rho + \psi)W)^{-1} \times (\beta_k I + \theta_k W))^{\bar{r}}sum</td>
</tr>
<tr>
<td>SAR</td>
<td>((I - \rho W)^{-1}) \times (\beta_k I))^{\bar{d}}</td>
<td>((I - \rho W)^{-1}) \times (\beta_k I))^{\bar{r}}sum</td>
<td>((1 - \tau)I - (\rho + \psi)W)^{-1} \times (\beta_k I))^{\bar{d}}</td>
<td>((1 - \tau)I - (\rho + \psi)W)^{-1} \times (\beta_k I))^{\bar{r}}sum</td>
</tr>
</tbody>
</table>

Source: Adapted from Elhorst (2014). Note: The superscript \(\bar{d}\) denotes the operator that calculates the mean diagonal element of a matrix, and the superscript \(\bar{r}\)\ sum denotes the operator that calculates the mean row sum of the nondiagonal elements.
Note that the analytical results reported in table 2 are valid only for linear (in variables) specifications. Thus, by default, a “factor variables” specification will block the computation of these effects. Nonetheless, in these cases, \texttt{xsmle} allows for the use of \texttt{margins} to at least compute total marginal effects. As described in section 4.1, \texttt{xsmle} also computes the standard errors of marginal effects using Monte Carlo simulation (the default) or the Delta method.

**Robust Hausman test**

A classical question in panel-data empirical analyses refers to the choice between FE and RE variants (when both can be estimated). An answer to this question can be given using the Hausman (1978) statistic,

$$\hat{\xi} = \hat{\delta}' \hat{V}_0^{-1} \hat{\delta}$$

(1)

where $\hat{\delta} = (\hat{\beta}_{FE} - \hat{\beta}_{RE})$ is the difference between the FE and RE estimates and $\hat{V}_0$ is an estimate of the VC matrix of $\hat{\delta}$. The asymptotic distribution of (1) under the null hypothesis of no systematic difference between the two sets of estimates is $\chi^2$ with $c$ degrees of freedom, with $c$ being usually the size of the estimated parameter vector. This test can be easily implemented in Stata using the official \texttt{hausman} command. However, one of the common issues with spatial panel-data models is that the Hausman specification test often fails to meet its asymptotic assumptions, especially in small samples. This is because, under the alternative hypothesis, $\hat{V}_0 = \hat{V}_{FE} + \hat{V}_{RE} - 2\text{Cov}(\hat{\beta}_{FE}, \hat{\beta}_{RE})$. In particular, \texttt{xsmle} estimates $\hat{V}_0$ through $D\hat{W}_0D'$, where $D = (I_c, -I_c)$ and $I_c$ denotes the identity matrix of size $c$. The joint VC matrix, $\hat{W}_0$, is consistently estimated using the following sandwich formula,

$$\hat{W}_0 = \begin{pmatrix} H_{FE} & O \\ O & H_{RE} \end{pmatrix}^{-1} \begin{pmatrix} S_{FE,FE} & S_{FE,RE} \\ S_{RE,FE} & S_{RE,RE} \end{pmatrix} \begin{pmatrix} H_{FE} & O \\ O & H_{RE} \end{pmatrix}^{-1}$$

with

$$\hat{H}_p = -\frac{1}{n} \sum_{i=1}^{n} \frac{\partial^2 L_{pi}}{\partial \beta \partial \beta'} (\hat{\beta}_p), \quad p = FE, RE$$

$$\hat{S}_{pq} = \frac{1}{n} \sum_{i=1}^{n} \frac{\partial L_{pi}}{\partial \delta} \frac{\partial L_{qi}}{\partial \beta'} (\hat{\beta}_q), \quad p, q = FE, RE$$

where $\hat{H}_{FE}^{-1}(S_{FE,FE})\hat{H}_{FE}^{-1} \text{ and } \hat{H}_{RE}^{-1}(S_{RE,RE})\hat{H}_{RE}^{-1}$ are the cluster-robust VC matrices of $\hat{\beta}_{FE}$ and $\hat{\beta}_{RE}$, where the cluster is represented by the panel unit. Note that the \texttt{hausman} option is allowed only for static models.

---

6. We thank an anonymous referee for bringing this point to our attention. Like other Stata estimation commands, \texttt{xsmle} cannot recognize nonlinear specifications not based on factor variables, for example, user-defined second-order terms or interactions.
The \textit{xsmle} command is written using the \texttt{optimize()} suite of functions and the optimization engine used by \textit{ml}. It shares the same features of all Stata estimation commands. Stata 10.1 is the earliest version that can run \textit{xsmle}. Only analytic weights (\texttt{aweights}) are allowed, but the declared \textit{weights} variable must be constant within each unit of the panel. \textit{xsmle} supports the \texttt{mi} prefix but does not support the \texttt{svy} prefix. Factor variables are allowed if Stata 11 (or later) is used to run the command.

One major prerequisite for using the command concerns the construction of the $n \times n$ matrix of spatial weights. This matrix can be a Stata matrix or \texttt{spmat} object, and it may follow any spatial weighting scheme, though it is usual to normalize spatial weights so either the row or column sums are equal to one. \textit{xsmle} does not allow the use of time-varying weight matrices. This means that the weights matrix is forced to be the same for each cross-section, and \textit{xsmle} will automatically replicate it for all time periods. This could be a limitation, especially in long panels, so a possible extension to \textit{xsmle} may provide the option to read multiple (time-varying) weight matrices. Note that the maximum dimension of a single Stata matrix depends on Stata’s flavor: 40 $\times$ 40 (Small), 800 $\times$ 800 (IC), and 11000 $\times$ 11000 (SE or MP). To overcome this limitation, one must specify bigger matrices as \texttt{spmat} objects. A second requirement for \textit{xsmle} is that the data must be \texttt{tsset} or \texttt{xtset} by the \textit{panel} and \textit{time} variables before the command is executed.

The basic \textit{xsmle} syntax is the following:

\begin{verbatim}
xsmle depvar [ indepvars ] [ if ] [ in ] [ weight ] [ , options ]
\end{verbatim}

The default is the \textit{RE SAR} model. A description of the main estimation and postestimation options is provided below. A full description of all available options is provided in the \textit{xsmle} help file.

### 3.1 Main options for xsmle

**Options common to all models**

\texttt{model(name)} specifies the spatial model to be fit. \textit{name} may be \texttt{sar} for the SAR model, \texttt{sdm} for the SDM, \texttt{sac} for the SAR with spatially autocorrelated errors model, \texttt{sem} for the SEM, or \texttt{gspre} for the GSPRE model. The default is \texttt{model(sar)}.

---

7. It is not assumed that $W$ is symmetric, but $(I - \rho W)$ must be nonsingular. This implies conditions on the eigenvalues of $W$ discussed extensively in the literature (for example, see LeSage and Pace [2009, chap. 3]).

8. This avoids the necessity of adding syntax to specify the panel and time variables. However, there is a corollary that should be noted. The natural way of organizing spatial panel data for estimation purposes is to stack each panel unit for period $t = 1$ followed by panel units for $t = 2$, and so on. Thus \textit{xsmle} internally sorts the dataset by time and panel unit but restores the original sorting on exit.
`vce(vcetype)` specifies how to estimate the VC matrix corresponding to the parameter estimates. The standard errors reported in the estimation results table are the square root of the variances (diagonal elements) of the VC estimator. `vcetype` may be one of the following:

- `oim` uses the observed information matrix.
- `opg` uses the sum of the outer product of the gradients.
- `robust` is the synonym for clustered sandwich estimator, where `clustvar` is the `panelvar`.
- `cluster clustvar` specifies the clustered sandwich estimator.
- `dkraay #` specifies the Driscoll–Kraay robust estimator. `#` is the maximum lag used in the calculation.
- `robust` is the synonym for `vce(cluster panelvar)`.
- `cluster(clustvar)` is the synonym for `vce(cluster clustvar)`.
- `constraints(constraints)`; see [R] estimation options.
- `level(#)` sets the confidence level for confidence intervals; the default is `level(95)`.
- `postscore` saves observation-by-observation scores in the estimation results list.
- `posthessian` saves the Hessian corresponding to the full set of coefficients in the estimation results list.
- `display_options`: vsquish, baselevels, allbaselevels; see [R] estimation options.
- `maximize_options`: difficult, technique(algorithm_spec), iterate(#), [no] log, from(init_specs), tolerance(#), itolerance(#), nrtolerance(#), nonrtolerance; see [R] maximize. These options are seldom used.

**Options for the SAR model**

`wmatrix(name)` specifies the weight matrix for the SAR term. `name` can be a Stata matrix or an `spmat` object. This matrix can be standardized or not. `wmatrix()` is required.

- `re` uses the random-effects estimator; `re` is the default.
- `fe` uses the fixed-effects estimator.
- `type(type_option[, leeyu])` specifies fixed-effects type. `type_option` may be `ind` for individual-fixed effects, `time` for time-fixed effects, or `both` for time- and individual-fixed effects. The `leeyu` suboption transforms the data according to Lee and Yu (2010).
- `dlag(dlag)` defines the structure of the spatiotemporal model. When `dlag` is equal to 1, only the time-lagged dependent variable is included; when `dlag` is equal to 2, only
the space-time-lagged dependent variable is included; when \( dlag \) is equal to 3, both the time-lagged and space-time-lagged dependent variables are included.

`noconstant` suppresses the constant term in the model. It is used only for the `re` estimator.

`effects` computes direct, indirect, and total effects and adds them to \( \mathbf{e}(b) \).

`vceffects(vce_type[, nsim(#)])` sets how the standard errors for the direct, indirect, and total effects are computed. `vce_type` may be `dm` for delta method standard errors, `sim[, nsim(#)]` for Monte Carlo standard errors, where `nsim(#)` sets the number of simulations for the LeSage and Pace (2009) procedure, or `none` for no standard errors.

`hausman` performs the robust Hausman test, automatically detecting the alternative estimator. The test is computed estimating the VC matrix of the difference between `fe` and `re` estimators as in White (1982). It is allowed only for static models.

**Options for the SDM model**

`wmatrix(name)` specifies the weight matrix for the SAR term. `name` can be a Stata matrix or an `spmat` object. This matrix can be standardized or not. `wmatrix()` is required.

`dmatrix(name)` specifies the weight matrix for the spatially lagged regressors; the default is to use the matrix specified in `wmat(name)`. `name` can be a Stata matrix or an `spmat` object. This matrix can be standardized or not.

`durbin(varlist)` specifies the regressors that have to be spatially lagged; the default is to lag all independent variables in `varlist`.

`re` uses the random-effects estimator; `re` is the default.

`fe` uses the fixed-effects estimator.

`type(type_option[, leeyu])` specifies fixed-effects type. `type_option` may be `ind` for individual-fixed effects, `time` for time-fixed effects, or `both` for time- and individual-fixed effects. The `leeyu` suboption transforms the data according to Lee and Yu (2010).

`dlag(dlag)` defines the structure of the spatiotemporal model. When `dlag` is equal to 1, only the time-lagged dependent variable is included; when `dlag` is equal to 2, only the space-time-lagged dependent variable is included; when `dlag` is equal to 3, both time-lagged and space-time-lagged dependent variables are included.

`noconstant` suppresses the constant term in the model. It is used only for the `re` estimator.

`effects` computes direct, indirect, and total effects and adds them to \( \mathbf{e}(b) \).
\texttt{vceffects}([\texttt{vcee\_type}], \texttt{nsim(#)}) sets how the standard errors for the direct, indirect, and total effects are computed. \texttt{vcee\_type} may be \texttt{dm} for delta method standard errors, \texttt{sim[, nsim(#)]} for Monte Carlo standard errors, where \texttt{nsim(#)} sets the number of simulations for the LeSage and Pace (2009) procedure, or \texttt{none} for no standard errors.

\texttt{hausman} performs the robust Hausman test, automatically detecting the alternative estimator. The test is computed estimating the \texttt{VC} matrix of the difference between \texttt{fe} and \texttt{re} estimators as in White (1982). It is allowed only for static models.

**Options for the SAC model**

\texttt{wmatrix(name)} specifies the weight matrix for the SAR term. \texttt{name} can be a Stata matrix or an \texttt{spmat} object. This matrix can be standardized or not. \texttt{wmatrix()} is required.

\texttt{ematrix(name)} specifies the weight matrix for the SAC error term. \texttt{name} can be a Stata matrix or an \texttt{spmat} object. This matrix can be standardized or not. \texttt{ematrix()} is required.

\texttt{fe} uses the fixed-effects estimator.

\texttt{type(type\_option[, leeyu])} specifies fixed-effects type. \texttt{type\_option} may be \texttt{ind} for individual-fixed effects, \texttt{time} for time-fixed effects, or \texttt{both} for time- and individual-fixed effects. The \texttt{leeyu} suboption transforms the data according to Lee and Yu (2010).

\texttt{effects} computes direct, indirect, and total effects and adds them to \texttt{e(b)}.

\texttt{vceffects}([\texttt{vcee\_type}], \texttt{nsim(#)}) sets how the standard errors for the direct, indirect, and total effects are computed. \texttt{vcee\_type} may be \texttt{dm} for delta method standard errors, \texttt{sim[, nsim(#)]} for Monte Carlo standard errors, where \texttt{nsim(#)} sets the number of simulations for the LeSage and Pace (2009) procedure, or \texttt{none} for no standard errors.

**Options for the SEM model**

\texttt{ematrix(name)} specifies the weight matrix for the SAC error term. \texttt{name} can be a Stata matrix or an \texttt{spmat} object. This matrix can be standardized or not. \texttt{ematrix()} is required.

\texttt{re} uses the random-effects estimator; \texttt{re} is the default.

\texttt{fe} uses the fixed-effects estimator.

\texttt{type(type\_option[, leeyu])} specifies fixed-effects type. \texttt{type\_option} may be \texttt{ind} for individual-fixed effects, \texttt{time} for time-fixed effects, or \texttt{both} for time- and individual-fixed effects. The \texttt{leeyu} suboption transforms the data according to Lee and Yu (2010).
**Spatial panel-data models using Stata**

`noconstant` suppresses the constant term in the model. It is used only for the `re` estimator.

`hausman` performs the robust Hausman test, automatically detecting the alternative estimator. The test is computed estimating the VC matrix of the difference between `fe` and `re` estimators as in White (1982). It is allowed only for static models.

### Options for the GSPRE model

- `wmatrix(name)` specifies the weight matrix for the SAC RE. `name` can be a Stata matrix or an `spmat` object. This matrix can be standardized or not. `wmatrix()` is required.

- `ematrix(name)` specifies the weight matrix for the SAC error term. `name` can be a Stata matrix or an `spmat` object. This matrix can be standardized or not.

- `re` uses the random-effects estimator.

- `error(error_options)` defines the random-effect error structure with `error_options = 1,...,4`. In particular, `error(1)` (the default) for $\phi \neq \lambda \neq 0$, `error(2)` for $\phi \neq 0$ and $\lambda = 0$, `error(3)` for $\phi = 0$ and $\lambda \neq 0$ (SEM model), and `error(4)` for $\phi = \lambda$.

- `noconstant` suppresses the constant term in the model. It is used only for the `re` estimator.

### 3.2 Postestimation command after xsmle

After an `xsmle` estimation, the `predict` command can be used to compute predicted values. Moreover, `predict` allows postestimation of FE or RE. The methods implemented in this command are the panel-data extension of those available in Kelejian and Prucha (2007) and Drukker, Prucha, and Raciborski (2013b). See section 4.1 for details.

#### Syntax for predict

The syntax of the command is the following:

```
predict [type] newvar [if] [in] [, rform full limited naive xb a noie]
```

#### Options for predict

- `rform`, the default, calculates predicted values from the reduced-form equation, $y_{it} = (I_n - \rho \mathbf{W})^{-1}(x_{it}\beta + \alpha_i)$.

- `full` calculates predicted values based on the full information set. This option is available only with `model(sac)`.
limited calculates predicted values based on the limited information set. This option is available only with model(sac).

naive calculates predicted values based on the observed values of \( y_{it} = \rho W y_{it} + x_{it}\beta + \alpha_i \).

xb calculates the linear prediction including the FE or RE \( x_{it}\beta + \alpha_i \).

a estimates \( \alpha_i \), the FE, or RE. With FE models, this statistic is allowed only with type(ind).

noie excludes the estimated \( \alpha_i \), the FE, or RE from the prediction.

4 Examples

4.1 Simulated data

In this section, we use simulated data to illustrate the xsmle command’s estimation capabilities, focusing on model selection, prediction, and estimation in the presence of missing data. In particular, we consider the following FE SDM model,

\[
y_{it} = 0.3 \sum_{j=1}^{n} w_{ij} y_{jt} + 0.5x_{1it} - 0.3x_{2it} - 0.2x_{3it} + 0.3 \sum_{j=1}^{n} w_{ij}x_{1it} \\
+ 0.6 \sum_{j=1}^{n} w_{ij}x_{2it} + 0.9 \sum_{j=1}^{n} w_{ij}x_{3it} + \mu_i + \epsilon_{it}
\]

where the nuisance parameters (\( \mu_i \)) are drawn from an independent and identically distributed (i.i.d.) standard Gaussian random variable. To allow for dependence between the unit-specific effects and the regressors, we generate the latter as follows,

\[
x_{kit} = 0.4\mu_i + (1 - 0.4^2)^{1/2} z_{kit}
\]

where \( z_{kit} \) is standard Gaussian with \( k = 1, 2, 3 \). The sample size is set to 940 (\( n = 188 \) and \( T = 5 \)) observations.

Let us begin by importing a first-order spatial contiguity matrix of the Italian local health authorities using the spmat command:

```stata
use ASL_contiguity_mat_ns.dta
spmat dta W W*, replace
```

The spmat dta command allows users to store an spmat object called \( W \) in the Stata memory. Notice that, to fit a model using xsmle, one must use the spatial weight matrix as a Stata matrix or an spmat object. The following spmat entry allows users to easily summarize the \( W \) object:

---

9. We report the code used for each example in the sj_examples_simdata.do accompanying file.
10. The chosen cross-sectional dimension (\( n = 188 \)) depends on the dimension of the used weight matrix, a contiguity matrix of the Italian local health authorities.
**Spatial panel-data models using Stata**

```
.spmat summarize W, links
Summary of spatial-weighting object W

<table>
<thead>
<tr>
<th>Matrix</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dimensions</td>
<td>188 x 188</td>
</tr>
<tr>
<td>Stored as</td>
<td>188 x 188</td>
</tr>
<tr>
<td>Links</td>
<td></td>
</tr>
<tr>
<td>total</td>
<td>906</td>
</tr>
<tr>
<td>min</td>
<td>1</td>
</tr>
<tr>
<td>mean</td>
<td>4.819149</td>
</tr>
<tr>
<td>max</td>
<td>13</td>
</tr>
</tbody>
</table>
```

As can be seen, the imported spatial matrix consists of 188 cross-sectional units with at least 1 neighbor, with about 4.8 contiguous units on average. Because `xsmle` does not make this transformation automatically, the next step consists in the row-normalization of the W object. This can easily be performed using the following:

```
.spmat dta W W*, replace normalize(row)
```

In particular, the syntax for fitting an FE SDM is

```
.xtset id t
    panel variable: id (strongly balanced)
    time variable: t, 1 to 5
    delta: 1 unit
.xsmle y x1 x2 x3, wmat(W) model(sdm) fe type(ind) nolog
```

Warning: All regressors will be spatially lagged

<table>
<thead>
<tr>
<th>SDM with spatial fixed-effects</th>
<th>Number of obs = 940</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group variable: id</td>
<td>Number of groups = 188</td>
</tr>
<tr>
<td>Time variable: t</td>
<td>Panel length = 5</td>
</tr>
<tr>
<td>R-sq: within = 0.3852</td>
<td></td>
</tr>
<tr>
<td>between = 0.3705</td>
<td></td>
</tr>
<tr>
<td>overall = 0.3635</td>
<td></td>
</tr>
<tr>
<td>Mean of fixed-effects = 0.0314</td>
<td></td>
</tr>
<tr>
<td>Log-likelihood = -1204.1194</td>
<td></td>
</tr>
</tbody>
</table>

| y          | Coef. | Std. Err. | z    | P>|z| | [95% Conf. Interval] |
|------------|-------|-----------|------|------|---------------------|
| Main       |       |           |      |      |                     |
| x1         | .5456416 | .034473 | 15.83 | 0.000 | .4780758 .6132075   |
| x2         | -.2798453 | .0356246 | -7.86 | 0.000 | -.3496683 -.2100224 |
| x3         | -.1896873 | .0356751 | -5.32 | 0.000 | -.2596093 -.1197654 |
| Wx         |       |           |      |      |                     |
| x1         | .3093964 | .0716979 | 4.32 | 0.000 | .16887 .4499207    |
| x2         | .5063665 | .0356246 | -7.86 | 0.000 | -.3496683 -.2100224 |
| x3         | .9072591 | .0748364 | 12.12 | 0.000 | .7605825 1.053936   |
| Spatial    |       |           |      |      |                     |
| rho        | .2274947 | .0425135 | 5.35 | 0.000 | .1441699 .3108196  |
| Variance   |       |           |      |      |                     |
| sigma2_e   | .7500305 | .0347637 | 21.58 | 0.000 | .6818948 .8181661  |
```
When the `fe` option is specified, `xsmle` fits a model with a unit-specific FE. This means that, in the example above, we might omit the `type(ind)` option.\footnote{The `nolog` option is seldom used and allows users to omit the display of the log-likelihood function iteration log. `xsmle` allows users to use all `maximize` options available for `ml` estimation commands (see `help maximize`) plus the additional `postscore` and `posthessian` options, which report the score and the hessian as an `e()` matrix. Note that the usual limit for matrix dimension does apply in this case.} The latter allows users to specify alternative forms for the FE: `type(time)` allows for time FE, while `type(both)` specifies both time and unit FE. In the case of SDM, `xsmle` also allows users to specify a different set of spatially lagged explanatory variables through the `durbin(varlist)` option. As the warning message reports, the default is to lag all independent variables in `varlist`.

To simplify the task of producing publication-quality tables, `xsmle` reports labeled estimation results. The `Main` equation contains the $\beta$ vector, the $WX$ equation reports (only for SDM) the $\theta$ vector, the `Spatial` equation reports the spatial coefficients (in this case $\rho$), and the `Variance` equation reports ancillary parameters as the variance of the error ($\sigma^2$ in this case).\footnote{Notice that for models other than SDM, the ancillary equations will be different following the specific parametrization used.}

Even if we already know the FE SDM is correctly specified in this example, we might be interested in testing the appropriateness of a RE variant using the official Stata `hausman` command:
Spatial panel-data models using Stata

```
xsmle y x1 x2 x3, wmat(W) model(sdm) re type(ind) nolog
Warning: Option type(ind) will be ignored
Warning: All regressors will be spatially lagged
SDM with random-effects
Number of obs = 940
Group variable: id
Number of groups = 188
Time variable: t
Panel length = 5
R-sq: within = 0.3671
between = 0.5567
overall = 0.4429
Log-likelihood = -1461.5464

|       | Coef.  | Std. Err. | z    | P>|z|  | [95% Conf. Interval] |
|-------|--------|-----------|------|------|----------------------|
| Main  |        |           |      |      |                      |
| y     |        |           |      |      |                      |
| x1    | .6278704 | .0383441 | 16.37 | 0.000 | .5527173 - .7030236 |
| x2    | -.1595226 | .0402597 | -3.96 | 0.000 | -.2384301 - .0806151 |
| x3    | -.0807422 | .0400913 | -2.01 | 0.044 | -.1593197 - .0021648 |
| _cons | .0214849 | .0669073 | 0.32  | 0.748 | -.109651 .1526208   |
| Wx    |        |           |      |      |                      |
| x1    | .3042129 | .0784076 | 3.88  | 0.000 | .1505368 .4578889  |
| x2    | .5215032 | .0805461 | 6.47  | 0.000 | .3636356 .6793707  |
| x3    | .9631849 | .0813256 | 11.84 | 0.000 | .8037896 1.12258   |
| Spatial|        |           |      |      |                      |
| rho   | .2558274 | .040904  | 6.25  | 0.000 | .175657 .3359977   |
| Variance|        |           |      |      |                      |
| lgtheta| -.0751917 | .1284863 | -0.59 | 0.558 | -.3270202 .1766369 |
| sigma2_e| .9648846 | .0515123 | 18.73 | 0.000 | .8639224 1.065847  |

```
```
estimates store sdm_re
describe
```
```
hausman sdm_fe sdm_re, eq(1:1 2:2 3:3)
```
```
<table>
<thead>
<tr>
<th>(b)</th>
<th>(B)</th>
<th>(b-B)</th>
<th>sqrt(diag(V_b-V_B))</th>
</tr>
</thead>
<tbody>
<tr>
<td>(b)</td>
<td>(B)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>sdme</td>
<td>sdm_re</td>
<td>Difference</td>
<td>S.E.</td>
</tr>
<tr>
<td>comp1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>x1</td>
<td>.5456416</td>
<td>.6278704</td>
<td>-.0822288</td>
</tr>
<tr>
<td>x2</td>
<td>-.2798453</td>
<td>-.1595226</td>
<td>-.1203277</td>
</tr>
<tr>
<td>x3</td>
<td>-.1896873</td>
<td>-.0807422</td>
<td>-.1089451</td>
</tr>
<tr>
<td>comp2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>x1</td>
<td>.3093954</td>
<td>.3042129</td>
<td>.0051825</td>
</tr>
<tr>
<td>x2</td>
<td>.5063665</td>
<td>.5215032</td>
<td>-.0151366</td>
</tr>
<tr>
<td>x3</td>
<td>.9072691</td>
<td>.9631849</td>
<td>-.0559257</td>
</tr>
<tr>
<td>comp3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>rho</td>
<td>.2274947</td>
<td>.2558274</td>
<td>-.0283326</td>
</tr>
</tbody>
</table>
```
b = consistent under Ho and Ha; obtained from xsmle
B = inconsistent under Ha, efficient under Ho; obtained from xsmle

Test: Ho: difference in coefficients not systematic
\[ \chi^2(7) = (b-B)'[(V_b-V_B)^{-1}](b-B) \]
\[ = -75.83 \] \( \chi^2 < 0 \) \( \Rightarrow \) model fitted on these data fails to meet the asymptotic assumptions of the Hausman test; see suest for a generalized test.

In this example, the Hausman statistic fails to meet its asymptotic assumptions. This problem can be overcome by adding the `hausman` option to the estimation command:

```
.xsmle y x1 x2 x3, wmat(W) model(sdm) fe type(ind) hausman nolog
```

As expected, in this case, we strongly reject the null hypothesis, with a \( \chi^2 \) test statistic equal to 91.10 and a \( p \)-value lower than 1%. Note that, if specified, the `hausman` option automatically detects the alternative model, which in our example is the RE.

Another common task routinely undertaken by spatial practitioners is model selection. Following the strategy described in LeSage and Pace (2009) and Elhorst (2010b), investigators should start with the SDM as a general specification and test for the alternatives. That is, we fit an SDM but would like to know whether it is the best model for the data at hand. This kind of procedure can be easily implemented using `xsmle`. For
instance, one may be interested in testing for SAR or SEM specifications. Because the SDM may be easily derived starting from a SEM, one can easily show that if $\theta = 0$ and $\rho \neq 0$, the model is a SAR, while if $\theta = -\beta \rho$, the model is a SEM. After the estimation of the SDM, these tests can be performed by exploiting the `xsmle` “equation-labeled” vector of estimated coefficients and using the official Stata `test` and `testnl` commands as follows:

```
.test [Wx]x1 = [Wx]x2 = [Wx]x3 = 0
( 1) [Wx]x1 - [Wx]x2 = 0
( 2) [Wx]x1 - [Wx]x3 = 0
( 3) [Wx]x1 = 0
   chi2( 3) = 203.77
   Prob > chi2 = 0.0000
```

```
.testnl ([Wx]x1 = -[Spatial]rho*[Main]x1) ([Wx]x2 = -[Spatial]rho*[Main]x2)
 > ([Wx]x3 = -[Spatial]rho*[Main]x3)
(1) [Wx]x1 = -[Spatial]rho*[Main]x1
(2) [Wx]x2 = -[Spatial]rho*[Main]x2
(3) [Wx]x3 = -[Spatial]rho*[Main]x3
   chi2(3) = 193.70
   Prob > chi2 = 0.0000
```

Finally, because the SAC and SDM are nonnested, information criteria can be used to test whether the most appropriate model is the SAC using the following:

```
.estimates restore sdm_fe
(results sdm_fe are active now)
.estat ic
Akaike's information criterion and Bayesian information criterion

<table>
<thead>
<tr>
<th>Model</th>
<th>Obs</th>
<th>ll(null)</th>
<th>ll(model)</th>
<th>df</th>
<th>AIC</th>
<th>BIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>sdm_fe</td>
<td>940</td>
<td>-1204.119</td>
<td>2424.239</td>
<td>8</td>
<td>2463.006</td>
<td></td>
</tr>
</tbody>
</table>
```

Note: N=Obs used in calculating BIC; see [R] BIC note.
and

```
.xsmle y x1 x2 x3, wmat(W) emat(W) model(sac) fe type(ind) nolog
```

SAC with spatial fixed-effects

```
Number of obs = 940
Group variable: id
```

```
Number of groups = 188
Time variable: t
Panel length = 5
```

```
R-sq: within = 0.2208  
      between = 0.0007  
      overall = 0.0667
```

```
Mean of fixed-effects = 0.0831
Log-likelihood = -1290.9574
```

|   | Coef. | Std. Err. | z    | P>|z| | [95% Conf. Interval] |
|---|-------|-----------|------|-----|---------------------|
|   |       |           |      |     |                     |
| **Main** |     |           |      |     |                     |
| x1 | 0.4860935 | 0.0415495 | 11.70 | 0.000 | 0.4046579 0.5675291 |
| x2 | -0.3332588 | 0.0370124 | -9.00 | 0.000 | -0.4058019 -0.2607158 |
| x3 | -0.3039008 | 0.0371472 | -8.18 | 0.000 | -0.3767081 -0.2310936 |
| **Spatial** |     |           |      |     |                     |
| rho  | -0.134535 | 0.1106866 | -1.22 | 0.224 | -0.3514768 0.0824067 |
| lambda | 0.4760945 | 0.0877639 | 5.42 | 0.000 | 0.3040804 0.6481085 |
| **Variance** |     |           |      |     |                     |
| sigma2_e | 1.073918 | 0.0469018 | 22.90 | 0.000 | 0.9819918 1.165844 |

```
 estad ic
AKaike’s information criterion and Bayesian information criterion
```

<table>
<thead>
<tr>
<th>Model</th>
<th>Obs</th>
<th>ll(null)</th>
<th>ll(model)</th>
<th>df</th>
<th>AIC</th>
<th>BIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>.</td>
<td>940</td>
<td>-1290.957</td>
<td>6 2593.915</td>
<td>2622.99</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Note: N=Obs used in calculating BIC; see [R] BIC note.

In this case, all tests point toward an FE SDM. Finally, one may be interested in the postestimation of the FE or predicted values of the outcome variable. In section 4.1, we summarize the spatial predictors implemented in xsmle. They are the panel-data extension of the predictors discussed in Kelejian and Prucha (2007), which range from the suboptimal na¨ıve predictor to the efficient minimum mean square error (MSE) full-information predictor. Here we give some examples of the xsmle postestimation syntax. For instance, to postestimate the FE once an FE spatial model has been fit, we type

```
. estimates restore sdm_fe
```

Now, to immediately visualize the deviation between the true (simulated) and estimated $\mu_i$ values, we may plot them using

```
.twoway (kdensity alpha, lpattern(dot) lwidth(*2)) > (kdensity alphahat, lpattern(dash)), > legend(row(1) label(1 "True") label(2 "Estimated"))
```
The resulting plot is shown in figure 1. Similarly, we can obtain a reduced form and naive prediction of the outcome variable using (the resulting plot is shown in figure 2)

```
. predict yhat_rform
      (option rform assumed)
. predict yhat_naive, naive
. twoway (kdensity y, lpattern(dot) lwidth(*2))
> (kdensity yhat_rform, lpattern(dash))
> (kdensity yhat_naive), legend(row(1) label(1 "True")
> label(2 "Reduced form") label(3 "Naive"))
```
Postestimation

In this section, we briefly discuss the predictors available in \texttt{xsmle} and replicate the Kelejian and Prucha (2007) Monte Carlo study, extending it to the case of panel data. Let us consider the following SAR with a SAC errors model,

\begin{align}
  y_t &= \rho W y_t + X_t \beta + \mu + \nu_t \\
  \nu_t &= \lambda M \nu_t + \epsilon_t 
\end{align}

for which we use the same notation discussed in section 2. In this model, $y_{it}$ is determined as

\begin{align}
  y_{it} &= \rho w_i y_t + x_{it} \beta + \mu_i + \nu_{it} \\
  \nu_{it} &= \lambda m_i \nu_t + \epsilon_{it}
\end{align}

where, for $t = 1, \ldots, T$, $w_i$ and $m_i$ are the $i$th rows of $W$ and $M$, $x_{it}$ is the $i$th row of $X_t$, $\nu_{it}$ and $\epsilon_{it}$ are the $i$th elements of $\nu_t$ and $\epsilon_t$, $\mu_i$ is the $i$th element of $\mu$, and $w_i, y_t$ and $m_i, \nu_t$ denote the $i$th elements of the spatial lags $Wy_t$ and $M \nu_t$ with $w_i, y_t$ that does not include $y_{it}$. By making the same assumptions of Kelejian and Prucha (2007), we have (see Kelejian and Prucha [2007] for more details on model assumptions)

\begin{align}
  \nu_t &\sim \mathcal{N}(0, \sigma_{\nu}^2 \Sigma^{\nu_t}) \\
  y_t &\sim \mathcal{N}(\xi_t, \sigma_y^2 \Sigma^{y_t})
\end{align}
Spatial panel-data models using Stata

with

\[ \xi_t = (I - \rho W)^{-1}(X_t\beta + \mu) \]
\[ \Sigma^\nu_t = (I - \lambda M)^{-1}(I - \lambda M')^{-1} \]
\[ \Sigma^y_t = (I - \rho W)^{-1}\Sigma^\nu_t(I - \rho W')^{-1} \]

We consider three information sets,

\[ \Lambda_1 = \{X_t, W\} \]
\[ \Lambda_2 = \{X_t, W, w_t, y_t\} \]
\[ \Lambda_3 = \{X_t, W, y_{t-1}\}, \quad t = 1, \ldots, T \]

where \( \Lambda_3 \) is the full-information set containing all \( n - 1 \) observations on \( y_t \) and \( \Lambda_1 \) and \( \Lambda_2 \) are both subsets of \( \Lambda_3 \). We consider the following four predictors of \( y_{it} \) (denoted as \( y_{it}^{(p)} \) with \( p = 1, \ldots, 4 \)),\(^1\)

\[ y_{it}^{(1)} = E(y_{it}|\Lambda_1) = \rho w_{it} y_t + x_{it}\beta + \mu_i + \frac{\text{cov}(\nu_{it}, w_{it}, y_t)}{\text{var}(w_{it}, y_t)} \{w_{it}y_t - E(w_{it}y_t)\} \]
\[ y_{it}^{(2)} = E(y_{it}|\Lambda_2) = \rho w_{it} y_t + x_{it}\beta + \mu_i + \text{cov}(\nu_{it}, y_{t-1}) \{\text{VC}(y_{t-1})\}^{-1} \{y_{t-1} - E(y_{t-1})\} \]
\[ y_{it}^{(3)} = E(y_{it}|\Lambda_3) = \rho w_{it} y_t + x_{it}\beta + \mu_i \]

where

\[ E(w_{it}, y_t) = w_{it}(I - \rho W)^{-1}(X_t\beta + \mu) \]
\[ \text{var}(w_{it}, y_t) = \sigma_w^2 w_{it} \Sigma^y w_i' \]
\[ \text{cov}(\mu_i, w_{it}, y_t) = \sigma_\nu^2 \Sigma^\nu w_i' \]
\[ E(y_{t-1}) = S_{t-1}(I - \rho W)^{-1}(X_t\beta + \mu) \]
\[ \text{VC}(y_{t-1}) = \sigma^2 S_{t-1} \Sigma^y S_{t-1}' \]
\[ \text{cov}(\nu_{it}, y_{t-1}) = \sigma_\nu^2 \Sigma^\nu w_i' \]

In the above expressions, \( (I - \rho W)^{-1} \) and \( \Sigma^\nu \) denote the \( i \)th rows of \( (I - \rho W)^{-1} \) and \( \Sigma^\nu \), respectively, while \( S_{t-1} \) is the \( n \times n \) selector matrix identical to the \( n \times n \) identity matrix \( I \), except that the \( i \)th row of \( I \) is deleted.

\(^1\) \( p = 1 \) indicates the reduced-form predictor, \( p = 2 \) indicates the limited-information predictor, \( p = 3 \) indicates the full-information predictor, and \( p = 4 \) indicates the naïve predictor.
We now compare the above predictors in terms of predictive efficiencies, extending the Kelejian and Prucha (2007) Monte Carlo design to FE models like the one reported in (4)–(5). In particular, we consider the following FE SAC model,

\begin{align*}
y_{it} &= \rho \sum_{j=1}^{n} w_{ij} y_{jt} + 0.5x_{1it} + \mu_i + \nu_{it} \\
\nu_{it} &= \lambda \sum_{j=1}^{n} w_{ij} \nu_{jt} + \epsilon_{it}
\end{align*}

where the nuisance parameters, \( \mu_i \), are drawn from an i.i.d. standard Gaussian random variable, while the \( x_{1it} \) regressor is generated according to (3). The simulation is based on what Kelejian and Prucha (2007) describe as the “two ahead and two behind” weight matrix, in which each unit is directly related to the two units immediately after it and immediately before it in the ordering. The matrix is row normalized, and all of its nonzero elements are equal to 1/4. As in Kelejian and Prucha (2007), we report results for 25 combinations of \( \rho, \lambda = -0.9, -0.4, 0, 0.4, 0.9 \) and set \( \sigma^2 = 1 \). The sample size is set to 500 (\( n = 100 \) and \( T = 5 \)) observations. Note that when \( \rho = 0 \), results refer to the SEM.

Simulation results in terms of sample averages over \( i = 1, \ldots, 100 \) and \( t = 1, \ldots, 5 \) for MSE(\( \hat{y}_{it}^{(p)} \)) for \( p = 2, \ldots, 4 \) are given in table 3. As expected, even in the panel-data case, numerical results are fully consistent with the theoretical notions reported in Kelejian and Prucha (2007): the biased naive predictor is the worst, especially when \( \rho = \lambda = 0.9 \), while the full information predictor is always the best.

---

14. See Kelejian and Prucha (2007) for more details on the structure of this weight matrix. Clearly, the results reported here depend on the structure of this matrix.

15. Because the reduced-form predictor has by far the worst performance, we do not report its results.
Table 3. Simulation results (MSEs)

<table>
<thead>
<tr>
<th>ρ</th>
<th>λ</th>
<th>naive</th>
<th>limited</th>
<th>full</th>
<th>α</th>
</tr>
</thead>
<tbody>
<tr>
<td>−0.9</td>
<td>−0.9</td>
<td>1.441</td>
<td>0.488</td>
<td>0.308</td>
<td>0.398</td>
</tr>
<tr>
<td>−0.9</td>
<td>−0.4</td>
<td>0.931</td>
<td>0.547</td>
<td>0.464</td>
<td>0.263</td>
</tr>
<tr>
<td>−0.9</td>
<td>0</td>
<td>0.811</td>
<td>0.678</td>
<td>0.659</td>
<td>0.221</td>
</tr>
<tr>
<td>−0.9</td>
<td>0.4</td>
<td>0.940</td>
<td>0.932</td>
<td>0.913</td>
<td>0.244</td>
</tr>
<tr>
<td>−0.9</td>
<td>0.9</td>
<td>6.384</td>
<td>1.331</td>
<td>1.155</td>
<td>1.594</td>
</tr>
<tr>
<td>−0.4</td>
<td>−0.9</td>
<td>1.348</td>
<td>0.547</td>
<td>0.465</td>
<td>0.359</td>
</tr>
<tr>
<td>−0.4</td>
<td>−0.4</td>
<td>0.902</td>
<td>0.649</td>
<td>0.625</td>
<td>0.246</td>
</tr>
<tr>
<td>−0.4</td>
<td>0</td>
<td>0.809</td>
<td>0.764</td>
<td>0.761</td>
<td>0.224</td>
</tr>
<tr>
<td>−0.4</td>
<td>0.4</td>
<td>0.937</td>
<td>0.863</td>
<td>0.857</td>
<td>0.252</td>
</tr>
<tr>
<td>−0.4</td>
<td>0.9</td>
<td>6.362</td>
<td>0.856</td>
<td>0.849</td>
<td>1.586</td>
</tr>
<tr>
<td>0</td>
<td>−0.9</td>
<td>1.340</td>
<td>0.678</td>
<td>0.659</td>
<td>0.346</td>
</tr>
<tr>
<td>0</td>
<td>−0.4</td>
<td>0.887</td>
<td>0.764</td>
<td>0.761</td>
<td>0.240</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0.803</td>
<td>0.793</td>
<td>0.792</td>
<td>0.220</td>
</tr>
<tr>
<td>0</td>
<td>0.4</td>
<td>0.937</td>
<td>0.765</td>
<td>0.762</td>
<td>0.257</td>
</tr>
<tr>
<td>0</td>
<td>0.9</td>
<td>6.019</td>
<td>0.702</td>
<td>0.660</td>
<td>1.548</td>
</tr>
<tr>
<td>0.4</td>
<td>−0.9</td>
<td>1.337</td>
<td>0.931</td>
<td>0.913</td>
<td>0.341</td>
</tr>
<tr>
<td>0.4</td>
<td>−0.4</td>
<td>0.885</td>
<td>0.863</td>
<td>0.857</td>
<td>0.233</td>
</tr>
<tr>
<td>0.4</td>
<td>0</td>
<td>0.811</td>
<td>0.765</td>
<td>0.762</td>
<td>0.229</td>
</tr>
<tr>
<td>0.4</td>
<td>0.4</td>
<td>0.966</td>
<td>0.673</td>
<td>0.651</td>
<td>0.274</td>
</tr>
<tr>
<td>0.4</td>
<td>0.9</td>
<td>5.541</td>
<td>0.659</td>
<td>0.518</td>
<td>1.445</td>
</tr>
<tr>
<td>0.9</td>
<td>−0.9</td>
<td>1.340</td>
<td>1.334</td>
<td>1.156</td>
<td>0.339</td>
</tr>
<tr>
<td>0.9</td>
<td>−0.4</td>
<td>0.887</td>
<td>0.857</td>
<td>0.849</td>
<td>0.230</td>
</tr>
<tr>
<td>0.9</td>
<td>0</td>
<td>0.805</td>
<td>0.702</td>
<td>0.660</td>
<td>0.215</td>
</tr>
<tr>
<td>0.9</td>
<td>0.4</td>
<td>1.445</td>
<td>0.659</td>
<td>0.518</td>
<td>0.758</td>
</tr>
<tr>
<td>0.9</td>
<td>0.9</td>
<td>8.150</td>
<td>1.127</td>
<td>0.391</td>
<td>2.250</td>
</tr>
</tbody>
</table>

On marginal effects

As already mentioned in section 2.1, a peculiar feature of spatial regression models is the feedback process among spatially correlated units, which leads to the distinction between direct, indirect, and total marginal effects. To show how to compute these effects using \texttt{xsmle}, let us consider the data-generating process of the following dynamic FE SDM model,

\[
y_{it} = \tau y_{it-1} + \psi \sum_{j=1}^{n} w_{ij} y_{jt-1} + 0.2 \sum_{j=1}^{n} w_{ij} y_{jt} + 0.5 x_{1it} + 0.3 x_{2it} - 0.2 x_{3it} + 0.3 \sum_{j=1}^{n} w_{ij} x_{1it} + 0.6 \sum_{j=1}^{n} w_{ij} x_{2it} + 0.9 \sum_{j=1}^{n} w_{ij} x_{3it} + \mu_i + \epsilon_{it}
\]  

(6)
where, as for the data-generating process reported in (2), the nuisance parameters are drawn from an i.i.d. standard Gaussian random variable and the correlation between unit-specific effects and regressors is obtained according to (3). The sample size is set to 1,960 observations ($n = 196$ and $T = 10$) and $\tau = \psi = 0.3$.

As documented in section 3.1, xsmle allows the estimation of (6) by specifying the dlag(3) option. By adding the effects option, one can use xsmle to compute direct, indirect, and total effects:

```
xsmle y x1 x2 x3, wmat(Wspmat) model(sdm) fe dlag(3) effects nolog
```

```
Dynamic SDM with spatial fixed-effects
```

```
Group variable: id
Time variable: t
Panel length = 9
```

```
R-sq: within = 0.3876
between = 0.9108
overall = 0.8354
```

```
Mean of fixed-effects = 0.0708
```

```
Log-likelihood = -2396.3051
```

```
Main

|   | Coef.  | Std. Err. | z    | P>|z| | [95% Conf. Interval] |
|---|--------|-----------|------|------|---------------------|
|  y | .278483 | .0187886  | 14.82| 0.000| .2416579 .315308    |
| L1|        |           |      |      |                     |
| Wy| .3371464| .0312009  | 10.81| 0.000| .2759938 .3982989    |
| L1|        |           |      |      |                     |
| x1| .471855 | .0312009  | 10.81| 0.000| .420539 .523171     |
| x2| -.2774485| .0263341  | -10.54| 0.000| -.3290623 -.2258347 |
| x3| -.1814445| .02648751 | -6.75| 0.000| -.2341187 -.1287704 |
| Wx|        |           |      |      |                     |
| x1| .3501276| .0516946  | 6.77 | 0.000| .2488081 .4514471   |
| x2| .6557425| .0498047  | 11.15| 0.000| .4580572 .6534278   |
| x3| .9499813| .0503458  | 18.87| 0.000| .8513054 1.048657   |

Spatial

```
rho | .152564 | .0287441  | 5.31 | 0.000| .0962165 .2088915   |

Variance

```
sigma2_e | .9612217 | .0291937 | 32.93 | 0.000| .9040031 1.01844    |

SR_Direct

```
x1 | .4920234 | .0251053 | 19.60 | 0.000| .4428179 .541229    |
|x2 | -.2567458| .0253696 | -10.12| 0.000| -.3064693 -.2070222 |
|x3 | -.1435512| .0251039 | -5.72 | 0.000| -.1927539 -.0943484 |
```

---

16. We thank Jihai Yu for sharing his MATLAB code for creating the rook spatial weights matrix used in this example. The original code has been translated into Mata for our purposes (see the accompanying sj examples simdata.do file for details).

17. dlag(1) allows the estimation of (6), in which $\psi = 0$, while dlag(2) is the case in which $\tau = 0$. 
Spatial panel-data models using Stata

<table>
<thead>
<tr>
<th>SR_Indirect</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>x1</td>
<td>.4867733</td>
<td>.0582277</td>
<td>8.36</td>
<td>0.000</td>
</tr>
<tr>
<td>x2</td>
<td>.589261</td>
<td>.0604524</td>
<td>9.69</td>
<td>0.000</td>
</tr>
<tr>
<td>x3</td>
<td>1.052221</td>
<td>.0616699</td>
<td>17.06</td>
<td>0.000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SR_Total</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>x1</td>
<td>.9787967</td>
<td>.0683064</td>
<td>14.33</td>
<td>0.000</td>
</tr>
<tr>
<td>x2</td>
<td>.3291804</td>
<td>.061426</td>
<td>4.83</td>
<td>0.000</td>
</tr>
<tr>
<td>x3</td>
<td>1.052221</td>
<td>.0616699</td>
<td>17.06</td>
<td>0.000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>LR_Direct</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>x1</td>
<td>.8954026</td>
<td>.0604489</td>
<td>14.24</td>
<td>0.000</td>
</tr>
<tr>
<td>x2</td>
<td>-.266502</td>
<td>.044616</td>
<td>-5.77</td>
<td>0.000</td>
</tr>
<tr>
<td>x3</td>
<td>.038457</td>
<td>.0470276</td>
<td>0.82</td>
<td>0.414</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>LR_Indirect</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>x1</td>
<td>2.750811</td>
<td>.462418</td>
<td>6.16</td>
<td>0.000</td>
</tr>
<tr>
<td>x2</td>
<td>1.485876</td>
<td>.274933</td>
<td>5.41</td>
<td>0.000</td>
</tr>
<tr>
<td>x3</td>
<td>3.352583</td>
<td>.4749726</td>
<td>7.06</td>
<td>0.000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>LR_Total</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>x1</td>
<td>3.646213</td>
<td>.483053</td>
<td>7.55</td>
<td>0.000</td>
</tr>
<tr>
<td>x2</td>
<td>1.223974</td>
<td>.3028385</td>
<td>4.06</td>
<td>0.000</td>
</tr>
<tr>
<td>x3</td>
<td>3.391083</td>
<td>.505929</td>
<td>6.71</td>
<td>0.000</td>
</tr>
</tbody>
</table>

When the effects option is specified, the marginal effects will be both displayed and added to the estimated vector \( e(b) \). Given its dynamic nature, (6) implies both short- and long-run effects (see table 2). In these cases, short-run effects are reported under the three equations labeled \( \text{SR}_{\text{Direct}}, \text{SR}_{\text{Indirect}}, \) and \( \text{SR}_{\text{Total}}, \) while long-run effects are reported under \( \text{LR}_{\text{Direct}}, \text{LR}_{\text{Indirect}}, \) and \( \text{LR}_{\text{Total}}. \) Equivalently, short-run total effects can be obtained through \text{margins} using the following syntax:

```
. margins, dydx(*) predict(rform noie)
```

Average marginal effects Number of obs = 1,764
Model VCE : OIM
Expression : Reduced form prediction, predict(rform noie)
dy/dx w.r.t. : x1 x2 x3

| Delta-method          | dy/dx | Std. Err. | z     | P>|z|  | [95% Conf. Interval] |
|-----------------------|-------|-----------|-------|------|---------------------|
| x1                    | .9695528 | .061065  | 14.24 | 0.000 | .8364665 1.030439 |
| x2                    | .328914  | .0669413 | 4.91  | 0.000 | .1971888 .450594 |
| x3                    | .9068859  | .0712236 | 12.73 | 0.000 | .7672903 1.014481 |

18. Equation names follow Elhorst (2014) terminology on short- and long-run marginal effects.
To ensure `margins` works, we added the `noisexsmle` postestimation option through the `predict()` option of `margins`. As can be seen, the two procedures produce slightly different results. This is because `xsmle`, by default, uses the Monte Carlo procedure outlined in LeSage and Pace (2009). Hence, the point estimates (standard errors) are averages (standard deviations) over the (default) 500 Monte Carlo replications. The same point estimates can be obtained using `xsmle` with the `vceeffects(none)` option.

```stata
. quietly xsmle y x1 x2 x3, wmat(Wspmat) model(sdm) fe dlag(3) effects > vceeffects(none)
. estimates store dsdm_fe
. estout dsdm_fe, keep(SR_Total:) c(b)

<table>
<thead>
<tr>
<th></th>
<th>b</th>
</tr>
</thead>
<tbody>
<tr>
<td>SR_Total</td>
<td></td>
</tr>
<tr>
<td>x1</td>
<td>.9699528</td>
</tr>
<tr>
<td>x2</td>
<td>.3283914</td>
</tr>
<tr>
<td>x3</td>
<td>.9068859</td>
</tr>
</tbody>
</table>
```

Because the analytical formulas for direct, indirect, and total effects reported in Table 2 imply a linear (in variables) specification, `xsmle` suppresses the computation of these effects when factor variables are specified, as shown in the example below:

```stata
. xsmle y c.x1##c.x1 c.x1#c.x2 c.x2 c.x3, wmat(Wspmat) model(sdm) fe dlag(3) > effects nolog
Warning: All regressors will be spatially lagged
Warning: direct and indirect effects cannot be computed if factor variables > are specified option -effects- ignored. Notice that total effects > can be obtained using -margins-
```

Dynamic SDM with spatial fixed-effects           Number of obs =  1764
Group variable: id                             Number of groups =  196
Time variable: t                               Panel length =  9
R-sq: within = 0.3929                           Number of obs =  1764
          between = 0.9122
          overall = 0.8378

19. The `vceeffects(none)` option suppresses the computation of standard errors.
Spatial panel-data models using Stata

Mean of fixed-effects = 0.0395
Log-likelihood = -2389.1429

|       | Coef. | Std. Err. | z     | P>|z|  | [95% Conf. Interval] |
|-------|-------|-----------|-------|------|---------------------|
| Main  |       |           |       |      |                     |
| y     |       |           |       |      |                     |
| L1.   | 0.282856 | 0.0187596 | 15.08 | 0.000 | 0.2460878 - 0.3196242 |
| Wy    |       |           |       |      |                     |
| L1.   | 0.3418741 | 0.0311604 | 10.97 | 0.000 | 0.2808008 - 0.4029474 |
| x1    | 0.4766421 | 0.0261214 | 18.25 | 0.000 | 0.4254451 - 0.5278391 |
| c.x1#c.x1 | 0.0446299 | 0.017106 | 2.61 | 0.009 | 0.0111027 - 0.0781571 |
| c.x1#c.x2 | -0.0847021 | 0.0311604 | -2.61 | 0.009 | -0.1323737 - 0.0370306 |
| x2    | -0.2720875 | 0.0261214 | -10.34 | 0.000 | -0.3236725 - 0.2205025 |
| x3    | -0.1824236 | 0.0261214 | -6.81 | 0.000 | -0.2340128 - 0.1308343 |

Wx

|       | Coef. | Std. Err. | z     | P>|z|  | [95% Conf. Interval] |
|-------|-------|-----------|-------|------|---------------------|
| x1    | 0.3633111 | 0.0517657 | 7.02  | 0.000 | 0.2618521 - 0.4647701 |
| c.x1#c.x1 | 0.0073658 | 0.0907751 | 0.22  | 0.822 | -0.056635 - 0.0715667 |
| c.x1#c.x2 | -0.0608714 | 0.0472681 | -1.29 | 0.198 | -0.1534612 - 0.0318264 |
| x2    | 0.5585988 | 0.0496918 | 11.24 | 0.000 | 0.4612047 - 0.6559929 |
| x3    | 0.9484272 | 0.0501452 | 18.91 | 0.000 | 0.8501445 - 1.04671         |

Spatial

|       | Coef. | Std. Err. | z     | P>|z|  | [95% Conf. Interval] |
|-------|-------|-----------|-------|------|---------------------|
| rho   | 0.1509144 | 0.0287019 | 5.26  | 0.000 | 0.0946597 - 0.2071691 |

Variance

|       | Coef. | Std. Err. | z     | P>|z|  | [95% Conf. Interval] |
|-------|-------|-----------|-------|------|---------------------|
| sigma2_e | 0.9534039 | 0.0289549 | 32.93 | 0.000 | 0.8966533 - 1.010154 |

Nonetheless, when the specification includes factor variables, xsmle allows for the use of margins to compute total marginal effects:

```
. margins, dydx(x1 x2 x3) predict(rform noie)
Warning: cannot perform check for estimable functions.
```

Average marginal effects Number of obs = 1,764
Model VCE : OIM
Expression : Reduced form prediction, predict(rform noie)
dy/dx w.r.t. : x1 x2 x3

|       | dy/dx | Std. Err. | z     | P>|z|  | [95% Conf. Interval] |
|-------|-------|-----------|-------|------|---------------------|
| x1    | 0.9952611 | 0.0673388 | 14.56 | 0.000 | 0.8613 - 1.129222 |
| x2    | 0.3278821 | 0.0655554 | 4.93  | 0.000 | 0.1974359 - 0.4583284 |
| x3    | 0.9021512 | 0.0707735 | 12.75 | 0.000 | 0.7634377 - 1.040865 |
`xsmle` also offers the opportunity to compute standard errors using the Delta method through the `vceffects(dm)` option.\footnote{While using the Delta method ensures the results do not depend on stochastic variability, it is a more computationally intensive procedure.}

```
. xsmle y x1 x2 x3, wmat(Wspmat) model(sdm) fe dlag(3) effects vceeffects(dm)
> nolog
Warning: All regressors will be spatially lagged
Computing marginal effects standard errors using delta-method...
Dynamic SDM with spatial fixed-effects          Number of obs =    1764
Group variable: id                            Number of groups =   196
Time variable: t                              Panel length =       9
R-sq: within = 0.3876                       between = 0.9108
       overall = 0.8354                     Mean of fixed-effects = 0.0708
Log-likelihood = -2396.3051

|          | Coef.  | Std. Err. |     z  |   P>|z| |     [95% Conf. Interval] |
|----------|--------|-----------|--------|--------|--------------------------|
| Main     |        |           |        |        |                          |
| y        |        |           |        |        |                          |
| L1.      | .278483| .0187886  | 14.82  | 0.000  | .2416579                 | .315308     |
| Wy       |        |           |        |        |                          |
| L1.      | .3371464| .0312009 | 10.81  | 0.000  | .2759938                 | .3982989   |
| x1       | .471855| .0261821  | 18.02  | 0.000  | .420539                  | .523171    |
| x2       | -.2774485| .0263341 | -10.54 | 0.000  | -.3290623                | -.2258347  |
| x3       | -.1814445| .0268751 | -6.75  | 0.000  | -.2341187                | -.1287704  |
| Wx       |        |           |        |        |                          |
| x1       | .3501276| .0516946 | 6.77   | 0.000  | .2488081                 | .4514471   |
| x2       | .6557425| .0498404 | 11.15  | 0.000  | .5680572                 | .7534278   |
| x3       | .9499813| .0503458 | 18.87  | 0.000  | .8513054                 | 1.048657   |
| Spatial  |        |           |        |        |                          |
| rho      | .152554| .0287441  | 5.31   | 0.000  | .0962165                 | .2088915   |
| Variance |        |           |        |        |                          |
| sigma2_e | .9612217| .0291937 | 32.93  | 0.000  | .9040031                 | 1.018444   |
| SR_Direct|        |           |        |        |                          |
| x1       | .4889378| .0262506 | 18.63  | 0.000  | .4374875                 | .5403881   |
| x2       | -.256507| .0262086 | -9.79  | 0.000  | -.3080385                | -.2053026  |
| x3       | -.144119| .02613   | -5.52  | 0.000  | -.1953328                | -.0929052  |
| SR_Indirect|       |           |        |        |                          |
| x1       | .481015 | .058855  | 8.21   | 0.000  | .3661905                 | .5958395   |
| x2       | .585062 | .0580324 | 10.08  | 0.000  | .4713206                 | .6988034   |
| x3       | 1.051005| .0630431 | 16.67  | 0.000  | .9274428                 | 1.174567   |
### Spatial panel-data models using Stata

<table>
<thead>
<tr>
<th>SR_Total</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>x1</td>
<td>.9699528</td>
<td>.0681065</td>
<td>14.24</td>
<td>0.000</td>
<td>.8364665</td>
<td>1.103439</td>
</tr>
<tr>
<td>x2</td>
<td>.3283914</td>
<td>.0699413</td>
<td>4.91</td>
<td>0.000</td>
<td>.1971888</td>
<td>.459594</td>
</tr>
<tr>
<td>x3</td>
<td>.9068859</td>
<td>.0712236</td>
<td>12.73</td>
<td>0.000</td>
<td>.7672903</td>
<td>1.046481</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>LR_Direct</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>x1</td>
<td>.8860048</td>
<td>.0680006</td>
<td>13.03</td>
<td>0.000</td>
<td>.7527261</td>
<td>1.019283</td>
</tr>
<tr>
<td>x2</td>
<td>-.25736</td>
<td>.0472579</td>
<td>-5.45</td>
<td>0.000</td>
<td>-.3499839</td>
<td>-.1647362</td>
</tr>
<tr>
<td>x3</td>
<td>.0347029</td>
<td>.0566012</td>
<td>0.61</td>
<td>0.540</td>
<td>-.0762334</td>
<td>.1456392</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>LR_Indirect</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>x1</td>
<td>2.659826</td>
<td>.5336009</td>
<td>4.80</td>
<td>0.000</td>
<td>1.574789</td>
<td>3.744864</td>
</tr>
<tr>
<td>x2</td>
<td>1.457852</td>
<td>.284823</td>
<td>5.12</td>
<td>0.000</td>
<td>.9002767</td>
<td>2.015427</td>
</tr>
<tr>
<td>x3</td>
<td>3.280576</td>
<td>.5347181</td>
<td>6.14</td>
<td>0.000</td>
<td>2.232548</td>
<td>4.328604</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>LR_Total</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>x1</td>
<td>3.545831</td>
<td>.6090295</td>
<td>5.82</td>
<td>0.000</td>
<td>2.352155</td>
<td>4.793507</td>
</tr>
<tr>
<td>x2</td>
<td>1.200492</td>
<td>.3154343</td>
<td>3.81</td>
<td>0.000</td>
<td>.582252</td>
<td>1.818732</td>
</tr>
<tr>
<td>x3</td>
<td>3.315279</td>
<td>.5776118</td>
<td>5.74</td>
<td>0.000</td>
<td>2.185181</td>
<td>4.447377</td>
</tr>
</tbody>
</table>

#### Unbalanced panels

Missing data can pose major problems when fitting econometric models because it is unlikely that missing values are missing completely at random. Most important here is that `xsmle` generally cannot handle unbalanced panels. A strategy to address this issue without relying on more complex econometric approaches is by multiple imputation, that is, the process of replacing missing values by multiple sets of plausible values. This section provides a simple example in which `xsmle` is used together with `mi`, Stata’s suite of commands dealing with multiple data imputation, to overcome the hurdle. Let us consider the same data-generating process reported in (2). The following syntax allows users to randomly assign 5% missing values to the $x_{1t}$ covariate:

```
. set seed 12345
. replace x1 = . if uniform()<0.05
   (49 real changes made, 49 to missing)
```

The first step is to declare the dataset as an `mi` dataset using `mi set`. Data must be `mi set` before other `mi` commands can be used. In this example, we choose the `wide` style. The second step is to register (declare) the variables with missing values using the `mi register` command:

```
. mi set wide
. mi register imputed x1
```

---

21. As usual, a good practice to obtain reproducible results is to set the seed of Stata’s pseudorandom number generator using the command `set seed #`, where # is any number between 0 and $2^{31} – 1$. 
We then use *mi impute regress* to fill in $x_1$’s missing values using the linear regression method with the $z$ covariate as the predictor.\(^{22}\) The `add(50)` option specifies the number of imputations to be added (currently, the total number of imputations cannot exceed 1,000).

\[
\text{. mi impute regress } x_1 = z, \text{ add(50) rseed(12345)}
\]

<table>
<thead>
<tr>
<th>Variable</th>
<th>Complete</th>
<th>Incomplete</th>
<th>Imputed</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_1$</td>
<td>891</td>
<td>49</td>
<td>49</td>
<td>940</td>
</tr>
</tbody>
</table>

(complete + incomplete = total; imputed is the minimum across $m$ of the number of filled-in observations.)

After *mi impute* has been executed, 50 new variables $\#_1x_1$ (with $\# = 1, \ldots, 50$) are added to the dataset, each representing an imputed version of $x_1$. Finally, we type

\[
\text{. mi estimate, dots post: xsmle y x1 x2 x3, wmat(W) model(sdm) fe type(ind) nolog}
\]

\[
\text{Imputations (50):} \\
\ldots
done
\]

| y | Coef. Std. Err. t P>|t| [95% Conf. Interval] |
|---|-----------------|---|-----------------|-----------------|---|
| Main | | | | | |
| $x_1$ | 0.509667 | 0.0367065 | 13.88 | 0.000 | .4377079 | .581626 |
| $x_2$ | -.2737751 | .0363977 | -7.52 | 0.000 | -.3451134 | -.2024368 |
| $x_3$ | -.1947675 | .036523 | -5.33 | 0.000 | -.2663518 | -.1231832 |
| Wx | | | | | |
| $x_1$ | 0.2788079 | .0769524 | 3.62 | 0.000 | .1279211 | .4296947 |
| $x_2$ | .5316003 | .0779399 | 6.82 | 0.000 | .3783891 | .6843615 |
| $x_3$ | .8991836 | .0768688 | 11.70 | 0.000 | .748522 | 1.049845 |
| Spatial | | | | | |
| rho | 0.2471005 | .042971 | 5.75 | 0.000 | .1628754 | .3313267 |
| Variance | | | | | |
| $\sigma^2_e$ | .7751222 | .0364928 | 21.24 | 0.000 | .7035961 | .8466484 |

\(22\) See *help mi impute* for details on the available imputation methods. The $z$ covariate is a standard Gaussian random variable specifically designed to be correlated with $x_1$. See the code reported in the *sj_examples_simdata.do* file for details.
to exploit \texttt{xsmle} to fit the FE SDM using the 50 imputed versions of the \texttt{x1} variable. In this way, both the coefficients and standard errors will be adjusted for the between-imputations variability according to the combination rules given in Rubin (1987). We replicated the same exercise by assigning (at random) a higher percentage (10\% and 20\%) of missing values to the \texttt{x1} covariate. To offer an example in which the multiple imputation strategy directly affects the \( \rho \) parameter value, we used the same strategy, assigning 5\%, 10\%, and 20\% missing values to the dependent variable.\textsuperscript{23}

The upper panel of table 4 reports the results for the case in which \texttt{x1} is the missing variable. As expected, the bias affecting the \( \beta \) parameter increases when the number of imputed values grows. The same is true for the \( \rho \) parameter when the missing values are in the dependent variable (lower panel of table 4). Note that even if these are not the results of a Monte Carlo simulation, the effect of missing values is seemingly stronger on \( \rho \) than \( \beta \).

\begin{table}[h]
\centering
\begin{tabular}{lcccc}
\hline
 & No missing & 5\% missing & 10\% missing & 20\% missing \\
\hline
\( \beta_1 \) & 0.546 & 0.510 & 0.471 & 0.425 \\
 & (0.034) & (0.037) & (0.040) & (0.043) \\
\hline
\hline
 & No missing & 5\% missing & 10\% missing & 20\% missing \\
\hline
\( \rho \) & 0.227 & 0.192 & 0.171 & 0.103 \\
 & (0.043) & (0.047) & (0.053) & (0.060) \\
\hline
\end{tabular}
\caption{Summary of estimation results by \% of missing values\textsuperscript{†}}
\end{table}

\textsuperscript{†} Standard errors in parentheses. True values: \( \beta_1 = 0.5 \), \( \rho = 0.3 \).

\subsection{4.2 Real data}

As an example of using spatial panel models with real data, we use a dataset on electricity usage at the state level in the United States. The data cover the 48 states in the continental United States plus the District of Columbia for the period 1990–2010. The data are drawn from the Electric Power Annual compiled by the Department of Energy’s Energy Information Agency together with general economic, demographic, and weather information from other U.S. statistical agencies, including the Bureau of Labor Statistics and the Census Bureau.\textsuperscript{24}

\textsuperscript{23} Interested readers can find the related Stata code in the accompanying \texttt{sj_examples_smdat.do} file.
\textsuperscript{24} Interested readers can find the Stata code and data used for this application in the accompanying \texttt{sj_empirical_application.do}, \texttt{wstate_rook.spda}, and \texttt{state_spatial.dta} files.
The analysis focuses on the response of residential electricity demand to prices and weather or climate conditions. The spatial dimension arises in at least two ways:

- Relative prices in neighboring states may influence decisions about the location of economic activities and subsequently of the residence. Electricity prices in California are high in comparison with prices in the Northwest and parts of the Midwest, but one would expect that location decisions and thus electricity demand will be more strongly influenced by prices in the Northwest than in the Midwest. In modeling terms, this behavior may be manifested as a significant coefficient on spatially weighted prices or on the spatially lagged dependent variable.

- Both weather and climate variables may serve as proxies for short- and long-term regional influences on the location of economic activity, the energy efficiency of buildings, and other determinants of electricity use. Given the physical capital stock, annual variations in weather will affect electricity demand for air conditioning or heating. Hence, it is interesting to determine whether local or regional weather variables have a statistically distinct influence on electricity demand.

Note that the logic suggesting a role for spatial influences on electricity demand in each state does not imply direct spatial interactions for the dependent variable, as in cases where it is argued that policy decisions in one state—for example, taxes on property—are influenced by decisions made by neighboring states. Instead, the arguments reflect a combination of omitted variables that may be spatially correlated plus the spatially distributed influence of variables that would be included in any model of electricity demand.

Tables 5–7 summarize the results obtained when FE models are used to examine residential demand for electricity using the log of residential consumption per person as the dependent variable. Elhorst and others argue that FE models are more appropriate for such data because the sample represents the complete population of U.S. continental states rather than a random sample drawn from that population. This claim is supported by the evidence given in the last two lines of Table 5, where all static RE specifications are strongly rejected by the Hausman test. The models do not provide a comprehensive analysis of factors that may influence demand, but they have been refined to focus on key variables that explain changes in electricity demand over the last two decades. For residential consumption, the large differences between the within and between $R^2$ statistics, other than for the models that include the lagged (in time) dependent variable, confirm the importance of state FE associated with variables that are not included in the analysis or that cannot be identified in this specification. Nonetheless, the within $R^2$ statistics, at least equal to 0.82, show that the models can account for a large proportion of variation over time in electricity consumption for residential usage by state. Weather variables, both heating and cooling degree days, have an important influence on residential consumption, and so does the size of the housing stock.

We test for alternative measures of income; the best indicator seems to be personal disposable income adjusted for differences in the cost of living across states (using the ACCRA cost of living index) and for changes in the CPI over time.
Table 5. FE models for residential electricity demand

<table>
<thead>
<tr>
<th></th>
<th>FE</th>
<th>SAR</th>
<th>dynamic SAR</th>
<th>SDM</th>
<th>dynamic SDM</th>
<th>SEM</th>
<th>SAC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Real personal income</td>
<td>0.391***</td>
<td>0.203***</td>
<td>0.033</td>
<td>0.212***</td>
<td>0.039</td>
<td>0.375***</td>
<td>0.207***</td>
</tr>
<tr>
<td>Real average residential price</td>
<td>-0.235***</td>
<td>-0.239***</td>
<td>-0.144***</td>
<td>-0.293***</td>
<td>-0.165***</td>
<td>-0.271***</td>
<td>-0.241***</td>
</tr>
<tr>
<td>Housing units</td>
<td>1.019***</td>
<td>0.747***</td>
<td>0.150*</td>
<td>0.629***</td>
<td>0.128*</td>
<td>0.818***</td>
<td>0.748***</td>
</tr>
<tr>
<td>Cooling degree days</td>
<td>0.075***</td>
<td>0.057***</td>
<td>0.073***</td>
<td>0.067***</td>
<td>0.072***</td>
<td>0.071***</td>
<td>0.057***</td>
</tr>
<tr>
<td>Heating degree days</td>
<td>0.188***</td>
<td>0.140***</td>
<td>0.146***</td>
<td>0.131***</td>
<td>0.140***</td>
<td>0.156***</td>
<td>0.141***</td>
</tr>
<tr>
<td>L. Residential electricity consumption</td>
<td>0.554***</td>
<td>0.534***</td>
<td>0.534***</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Real average total price</td>
<td>0.367***</td>
<td>0.261***</td>
<td>0.416***</td>
<td>0.284***</td>
<td>0.390***</td>
<td>0.359***</td>
<td>0.018</td>
</tr>
<tr>
<td>Log likelihood</td>
<td>2029.91</td>
<td>2108.54</td>
<td>2315.67</td>
<td>2144.51</td>
<td>2323.58</td>
<td>2071.08</td>
<td>2108.58</td>
</tr>
<tr>
<td>Observations</td>
<td>1078</td>
<td>1078</td>
<td>1029</td>
<td>1078</td>
<td>1029</td>
<td>1078</td>
<td>1078</td>
</tr>
<tr>
<td>$R^2_b$</td>
<td>0.82</td>
<td>0.82</td>
<td>0.89</td>
<td>0.85</td>
<td>0.89</td>
<td>0.82</td>
<td>0.82</td>
</tr>
<tr>
<td>$R^2_w$</td>
<td>0.12</td>
<td>0.14</td>
<td>0.91</td>
<td>0.16</td>
<td>0.91</td>
<td>0.21</td>
<td>0.14</td>
</tr>
<tr>
<td>$R^2$</td>
<td>0.18</td>
<td>0.20</td>
<td>0.89</td>
<td>0.22</td>
<td>0.90</td>
<td>0.26</td>
<td>0.20</td>
</tr>
<tr>
<td>Hausman $\chi^2$</td>
<td>28.24</td>
<td>27.33</td>
<td>33.19</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hausman $p$-value</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Significance levels: * $p < 10\%$, ** $p < 5\%$, and *** $p < 1\%$
Table 6. Residential electricity demand—test for model selection

<table>
<thead>
<tr>
<th>Model Comparison</th>
<th>$\chi^2$</th>
<th>p-value</th>
<th>Akaike’s information criterion</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAR versus dynamic SAR</td>
<td>414.26</td>
<td>0.000</td>
<td>-</td>
</tr>
<tr>
<td>SDM versus dynamic SDM</td>
<td>358.14</td>
<td>0.000</td>
<td>-</td>
</tr>
<tr>
<td>dynamic SAR versus dynamic SDM</td>
<td>15.82</td>
<td>0.000</td>
<td>-</td>
</tr>
<tr>
<td>SEM versus dynamic SDM</td>
<td>505.00</td>
<td>0.000</td>
<td>-</td>
</tr>
<tr>
<td>SAC</td>
<td></td>
<td></td>
<td>$-4201.0$</td>
</tr>
<tr>
<td>dynamic SDM</td>
<td></td>
<td></td>
<td>$-4629.0$</td>
</tr>
</tbody>
</table>
### Table 7. Direct, indirect, and total effects—residential electricity demand

<table>
<thead>
<tr>
<th></th>
<th>SAR</th>
<th>dynamic</th>
<th>SAR</th>
<th>dynamic</th>
<th>SAC</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Long-run direct effects</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Real personal income</td>
<td>0.210***</td>
<td>0.082</td>
<td>0.223***</td>
<td>0.096</td>
<td>0.214***</td>
</tr>
<tr>
<td>Real average residential price</td>
<td>−0.247***</td>
<td>−0.359***</td>
<td>−0.307***</td>
<td>−0.401***</td>
<td>−0.249***</td>
</tr>
<tr>
<td>Housing units</td>
<td>0.775***</td>
<td>0.375**</td>
<td>0.660***</td>
<td>0.312*</td>
<td>0.775***</td>
</tr>
<tr>
<td>Cooling degree days</td>
<td>0.059***</td>
<td>0.182***</td>
<td>0.059***</td>
<td>0.174***</td>
<td>0.059***</td>
</tr>
<tr>
<td>Heating degree days</td>
<td>0.145***</td>
<td>0.366***</td>
<td>0.138***</td>
<td>0.339***</td>
<td>0.146***</td>
</tr>
<tr>
<td>Real average total price</td>
<td>0.020***</td>
<td>0.026*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Long-run indirect effects</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Real personal income</td>
<td>0.110***</td>
<td>0.095</td>
<td>0.141***</td>
<td>0.121*</td>
<td>0.109***</td>
</tr>
<tr>
<td>Real average residential price</td>
<td>−0.130***</td>
<td>−0.417**</td>
<td>−0.194***</td>
<td>−0.508**</td>
<td>−0.126**</td>
</tr>
<tr>
<td>Housing units</td>
<td>0.406***</td>
<td>0.435*</td>
<td>0.417***</td>
<td>0.394</td>
<td>0.393*</td>
</tr>
<tr>
<td>Cooling degree days</td>
<td>0.031***</td>
<td>0.211***</td>
<td>0.038***</td>
<td>0.221***</td>
<td>0.030*</td>
</tr>
<tr>
<td>Heating degree days</td>
<td>0.076***</td>
<td>0.424*</td>
<td>0.087***</td>
<td>0.429*</td>
<td>0.074*</td>
</tr>
<tr>
<td>Real average total price</td>
<td>0.262***</td>
<td>0.285*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Long-run total effects</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Real personal income</td>
<td>0.321***</td>
<td>0.177</td>
<td>0.364***</td>
<td>0.217</td>
<td>0.323***</td>
</tr>
<tr>
<td>Real average residential price</td>
<td>−0.377***</td>
<td>−0.777***</td>
<td>−0.502***</td>
<td>−0.909***</td>
<td>−0.375***</td>
</tr>
<tr>
<td>Housing units</td>
<td>1.180***</td>
<td>0.809**</td>
<td>1.077***</td>
<td>0.706*</td>
<td>1.168***</td>
</tr>
<tr>
<td>Cooling degree days</td>
<td>0.090***</td>
<td>0.393***</td>
<td>0.097***</td>
<td>0.395***</td>
<td>0.089***</td>
</tr>
<tr>
<td>Heating degree days</td>
<td>0.222***</td>
<td>0.790***</td>
<td>0.225***</td>
<td>0.768***</td>
<td>0.229***</td>
</tr>
<tr>
<td>Real average total price</td>
<td>0.282***</td>
<td>0.311*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Short-run direct effects</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Real personal income</td>
<td>0.033</td>
<td>0.040</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Real average residential price</td>
<td>−0.146***</td>
<td>−0.169***</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Housing units</td>
<td>0.152*</td>
<td>0.131*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cooling degree days</td>
<td>0.074***</td>
<td>0.073***</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Heating degree days</td>
<td>0.149***</td>
<td>0.142***</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Real average total price</td>
<td>0.004*</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Short-run indirect effects</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Real personal income</td>
<td>0.011</td>
<td>0.015</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Real average residential price</td>
<td>−0.048***</td>
<td>−0.062***</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Housing units</td>
<td>0.050*</td>
<td>0.048</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cooling degree days</td>
<td>0.024***</td>
<td>0.027***</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Heating degree days</td>
<td>0.049***</td>
<td>0.052***</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Real average total price</td>
<td>0.075*</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Short-run total effects</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Real personal income</td>
<td>0.044</td>
<td>0.055</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Real average residential price</td>
<td>−0.194***</td>
<td>−0.231***</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Housing units</td>
<td>0.203*</td>
<td>0.179</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cooling degree days</td>
<td>0.098***</td>
<td>0.100***</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Heating degree days</td>
<td>0.198***</td>
<td>0.195***</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Real average total price</td>
<td>0.079*</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Significance levels: * \( p < 10\% \), ** \( p < 5\% \), and *** \( p < 1\% \)
Table 5 shows strong spatial interactions in residential consumption. The coefficients of the spatially lagged dependent variable ($\rho$) and of the spatially weighted price are highly significant and appear to have quite separate influences on consumption. Spatial Durbin variables with coefficients not significantly different from zero have been dropped from the model. The results in table 6 reinforce recommendations from LeSage and Pace (2009) and Elhorst (2010a) that investigators should start with the SDM as a general specification and test for the exclusion of variables for nested models using likelihood-ratio tests; for the SAC model, we adopted the modified Akaike’s information criterion as in Burnham and Anderson (2004). The positive $\rho$ coefficient is consistent with omitted regional factors that vary over time and affect residential consumption. The positive coefficient on the spatially weighted average price in neighboring states indicates a clear displacement effect by which an increase in electricity prices in one state encourages a shift in demand from that state to neighboring states. That is an important constraint on the impact of state programs to promote renewable energy or reduce CO2 emissions. The coefficients on the lagged dependent variable are both highly significant in columns (3) and (5) of table 5. The series within panel time are too short to carry out reliable tests, but the coefficient on the lagged dependent variable is so far from one that it is unlikely the equations have a unit root. The $\lambda$ coefficient in the SEM column, (6), is highly significant, but this specification is dominated by the SAC model in column (7). The inclusion of the spatially lagged dependent variables reduces the estimate of $\lambda$ from 0.39 to about 0.02 so that it is no longer significantly different from 0. Overall, the results in table 5 together with the test reported in table 6 suggest that the dynamic SDM (column 5) provides the best specification.

One of the reasons for studying such models is to fit the price elasticities of demand. In the nonspatial specification, the elasticity is simply the coefficient of the log price. As discussed in section 2, the marginal effect of price on electricity demand may differ across states because of spatial interactions. The key difference between the direct and total impacts is that the direct impact measures the impact of a unit change in variable $x_k$ in state $i$ on demand in state $i$ averaged over all states. In contrast, the total impact measures the impact of the same unit change in variable $x_k$ in all states on demand in state $i$, again averaged over all states. xsmle displays values for the direct, indirect, and total impact of changes in each of the independent variables. Unlike the values reported in table 5, table 7 reports elasticities accounting for spatial feedback. Moreover, for the SAR and SDM dynamic specifications, table 7 also distinguishes between short- and long-run marginal effects. Note that marginal effects in static models have been labeled as long run, but they should be compared with short-run effects from dynamic models (see table 2). These additional results are consistent across all spatial specifications, with the controls being significant and with the expected signs. The inclusion of the time-lagged dependent variable makes the coefficient for the real personal income not significant anymore and greatly reduces the elasticity of residential consumption with respect to the other controls.
5 Conclusions

In this article, we described the new \texttt{xsmle} command, which can be used to fit an extensive array of spatial models for panel data. \texttt{xsmle} supports weight matrices in the form of both Stata matrices and \texttt{spmat} objects, allows the computation of direct, indirect, and total effects and related standard errors, and provides various postestimation features for obtaining predictions, including the use of \texttt{margins}. Furthermore, \texttt{xsmle} is fully compatible with the \texttt{mi} Stata suite of commands. We used simulated data to illustrate \texttt{xsmle} estimation capabilities, focusing on model selection, prediction, and estimation in the presence of missing data, and provided an empirical application based on electricity usage data at the state level in the United States.

6 Acknowledgments

We would like to thank, in particular, Paul Elhorst and Michael Pfaffermayr for permission to use their MATLAB code. For most routines, we have modified or extended the way that the routines operate, so they should not be held responsible for any errors that may be in our code.

7 References


F. Belotti, G. Hughes, and A. Piano Mortari


Spatial panel-data models using Stata


About the authors

Federico Belotti is an assistant professor of econometrics at the Department of the Economics and Finance of the University of Rome Tor Vergata and a fellow of the Centre for Economics and International Studies (University of Rome Tor Vergata).

Gordon Hughes is a professor of economics at the School of Economics of the University of Edinburgh.

Andrea Piano Mortari is a researcher at the National Research Council, Institute for Research on Population and Social Policies, and a fellow of the Centre for Economics and International Studies (University of Rome Tor Vergata).
The estimation and modeling of cause-specific cumulative incidence functions using time-dependent weights

Paul C. Lambert
Department of Health Sciences
University of Leicester
Leicester, UK
and
Medical Epidemiology and Biostatistics
Karolinska Institutet
Stockholm, Sweden
paul.lambert@le.ac.uk

Abstract. Competing risks occur in survival analysis when an individual is at risk of more than one type of event and one event’s occurrence precludes another’s. The cause-specific cumulative incidence function (CIF) is a measure of interest with competing-risks data. It gives the absolute (or crude) risk of having the event by time $t$, accounting for the fact that it is impossible to have the event if a competing event occurs first. The user-written command stcompet calculates nonparametric estimates of the cause-specific CIF, and the official Stata command stcrreg fits the Fine and Gray (1999, Journal of the American Statistical Association 94: 496–509) model for competing-risks data. Geskus (2011, Biometrics 67: 39–49) has recently shown that standard software can estimate some of the key measures in competing risks by restructuring the data and incorporating weights. This has a number of advantages because any tools developed for standard survival analysis can then be used to analyze competing-risks data. In this article, I describe the stcrprep command, which restructures the data and calculates the appropriate weights. After one uses stcrprep, a number of standard Stata survival analysis commands can then be used to analyze competing risks. For example, sts graph, failure will give a plot of the cause-specific CIF, and stcox will fit the Fine and Gray (1999) proportional subhazards model. Using stcrprep together with stcox is computationally much more efficient than using stcrreg. In addition, stcrprep opens up new opportunities for competing-risk models. I illustrate this by fitting flexible parametric survival models to the expanded data to directly model the cause-specific CIF.

Keywords: st0471, stcrprep, survival analysis, competing risks, time-dependent effects

1 Introduction

Competing risks occur in survival analysis when a subject is at risk of more than one type of event. A classic example is consideration of different causes of death. If a
Competing risks

subject dies of one particular cause, he or she is no longer at risk of death from any other cause. Interest may lie in the cause-specific hazard rate, which can be estimated using standard survival techniques by treating any competing events as censored observations at the time of the competing event. An alternative measure is the cause-specific cumulative incidence function (CIF), which gives an estimate of absolute or crude risk of death, accounting for the possibility that individuals may die of other causes (Putter, Fiocco, and Geskus 2007).

The user-written \texttt{stcompet} command, available from the Statistical Software Components (Coviello and Boggess 2004), can estimate a nonparametric estimate of the cause-specific CIF (Kalbfleisch and Prentice 2002). Regression models of the cause-specific CIF are usually based around the subdistribution hazard, which is the hazard function corresponding to the cause-specific CIF. Within Stata, the \texttt{stcrreg} command implements the approach of Fine and Gray (1999). This uses a weighting procedure where individuals with a competing event remain at risk, with a weight that depends on the censoring distribution for the study population as a whole.

Geskus (2011) proposed an alternative way to estimate the cause-specific CIF using weighted versions of standard estimators. A corresponding R command, \texttt{crprep}, which is part of the \texttt{mstate} package, restructures the data and calculates the appropriate weights (de Wreede, Fiocco, and Putter 2011). In this article, I describe the \texttt{stcrprep} command, which has similar functionality to \texttt{crprep}—with further extensions to allow parametric models for the cause-specific CIF to be fit. After using \texttt{stcrprep}, you can plot the cause-specific CIF using \texttt{sts graph} and fit a Fine and Gray (1999) model using \texttt{stcox}. An advantage of this approach is that some of the methods developed for the Cox model can be used for models on the subdistribution hazard scale. For example, testing and visualization of the proportional subdistribution hazards assumption can use Schoenfeld residuals using \texttt{estat phtest}.

Parametric models have a number of advantages over Cox models when modeling cause-specific hazard functions—if the parametric form is appropriate. Flexible parametric survival models use restricted cubic splines to estimate the underlying hazard and survival functions, which enables virtually any shaped hazard to be captured (Royston and Parmar 2002). These models are implemented in Stata with the \texttt{stpm2} command, available from the Statistical Software Components (Lambert and Royston 2009; Royston and Lambert 2011). In this article, I will show that by restructuring the data and calculation of appropriate weights, these models can directly estimate and model CIFs.

The remainder of this article is structured as follows. Section 2 describes some competing-risks theory and the Fine and Gray (1999) model for the cause-specific CIF. Section 3 describes the syntax of the \texttt{stcrprep} command. Section 4 gives some examples of \texttt{stcrprep} for nonparametric estimation of the cause-specific CIF and fitting the Fine and Gray (1999) model. Section 5 describes how the approach extends to parametric models, with section 6 giving some examples. Finally, section 7 discusses the approach and briefly describes some possible extensions.
2 Methods

In competing risks, a patient is at risk from $K$ different causes. The cause-specific hazard function for cause $k$, $h_k(t)$, is defined as

$$h_k(t) = \lim_{\Delta t \to 0} \frac{P(t \leq T < t + \Delta t, \text{cause} = k | T \geq t)}{\Delta t}$$

When the competing events are deaths from different causes, these can be thought of as the mortality rate for cause $k$, conditional on survival to time $t$. To be at risk at time $t$, an individual cannot have died of the cause of interest or any of the competing causes of death.

One can transform the cause-specific hazard function to a survival function, $S_k(t)$, as follows:

$$S_k(t) = \exp \left\{ - \int_0^t h_k(u) \, du \right\}$$

(1)

If the competing events are assumed to be independent (conditional on covariates), then $1 - S_k(t)$ can be interpreted as the probability of having event $k$ by time $t$ in a hypothetical world where it is not possible to have any of the competing events. However, in many situations, the assumption of independence will not be reasonable, and the estimates obtained through (1) are not interpretable as probabilities. Even if independence is reasonable, it may be of more interest to calculate probabilities in the real world rather than in this hypothetical world. In such situations, it is of more interest to estimate the cause-specific cumulative incidence function, $F_k(t)$, defined as

$$F_k(t) = \int_0^t h_k(s) \exp \left\{ - \int_0^s \sum_{k=1}^K h_k(u) \, du \right\} \, ds$$

(2)

This gives the probability of having event $k$ as a function of time, accounting for the fact that subjects may have one of the competing events first. It depends on the underlying hazard rates for all $K$ causes. When one estimates cause-specific CIFs, the competing events do not have to be independent.

A nonparametric estimate of the cause-specific cumulative incidence function for cause $k$ is (Kalbfleisch and Prentice 2002)

$$\widehat{F}_k(t) = \sum_{j|t_j \leq t} \widehat{S}(t_{j-1}) \frac{d_{kj}}{n_j}$$

where $\widehat{S}(t_{j-1})$ is the Kaplan–Meier estimate of the all-cause survival function at time $t_{j-1}$, $d_{kj}$ is the number of deaths due to cause $k$ at time $t_j$, and $n_j$ is the number at risk.

A mathematically equivalent estimate of the CIF is the product limit estimate

$$\left( \prod_{j|t_j \leq t} \left( 1 - \frac{d_{kj}}{n_j^*} \right) \right)^{PL}$$
where \( n^*_j \) is the observed number at risk, \( n_j \), at time \( t_j \)—augmented by a weighted sum of individuals who had a competing event. The weights for each individual for a competing event are obtainable through estimation of the censoring distribution, \( S_c(t) \). The weight for individual \( i \) at time \( t_j \) is

\[
w_{ij} = \begin{cases} 
1 & \text{if still at risk at } t_j \\
0 & \text{if censored before } t_j \\
\frac{\hat{S}_c(t_i)}{\hat{S}_c(t_j)} & \text{if had competing event at } t_l < t_j
\end{cases}
\]

(3)

giving \( n^*_j = \sum_{i=1}^{N} w_{ij} \). The weight for those that experience a competing event is the probability of not being censored by time \( t_j \), given that they had a competing event at time \( t_l \). The censoring distribution, \( S_c(t) \), is usually calculated using the Kaplan–Meier method. The weights are time dependent and need to be calculated at each event time subsequent to the competing event.

### 2.1 Models for the cumulative incidence function

Although models that estimate the cause-specific hazards for each of the competing causes can estimate the cause-specific cumulative incidence function using (2), they provide no direct estimate of covariate effects on the cause-specific cumulative incidence function. Fine and Gray (1999) proposed an alternative model for the subhazard function, defining the subhazard as

\[
h^*_k(t) = -\frac{d \ln(1 - F_k(t))}{\Delta t} = \lim_{\Delta t \to 0} \frac{P\{t \leq T < t+\Delta t, cause = k \} | T \geq t \text{ or } T \leq t \text{ and cause} \neq k\}}{\Delta t}
\]

The subhazard rate has quite an awkward interpretation because it is the event rate at time \( t \) for those who have not had the event of interest, but may have had a competing event. If the events are different causes of death, it is the event rate where people who have died from competing causes are still considered at risk of the event of interest.

An advantage of the subhazard function is that one can derive the cause-specific CIF for cause \( k \) by the usual transformation from hazard to survival function:

\[
F_k(t) = 1 - \exp \left\{ \int_0^t h^*_k(u)du \right\}
\]

Fine and Gray (1999) defined the proportional subhazards model as

\[
h^*_k(t) = h^*_k,0(t) \exp(x\beta)
\]

This is the same form as a Cox proportional hazards model, but the hazard function has been replaced by the subhazard function. The \( \beta \) coefficients give log subhazard-ratios. The Fine and Gray (1999) model is similar to the Cox model in that the baseline
subhazard is not directly estimated. Estimation of parameters for a proportional sub-
hazards model is more complex than the Cox model because those that experience
competing events are kept in the risk set—but have time-dependent weights after their
event time is incorporated into the partial-likelihood, with the weights being a function
of the censoring distribution (Fine and Gray 1999). The weights are necessary because
they account for the fact that the probability that these observations could have been
censored increases with follow-up time. Estimation is done by maximizing the weighted
partial likelihood,

$$
\log L = \sum_{i=1}^{n} d_i \left[ x_i \beta - \log \left\{ \sum_{j \in R_i} w_{ij} \exp(x_i \beta) \right\} \right]
$$

where \( d_i \) is the event indicator and \( R_i \) is the set of observations, \( j \), that are at risk
at time \( t_i \). Note that (4) is the same as the partial likelihood for the Cox model with
the addition of the weights, \( w_{ij} \). These weights are calculated internally when using
the `stcrreg` command to fit the Fine and Gray (1999) model. Using the `stcrprep`
command expands the data and calculates the relevant weights. These weights are the
same as those given in (3), so `stcox` can fit the model.

3 The stcrprep command

3.1 Syntax

```
stcrprep [if] [in], events(varname) [byg(varlist) byh(varlist) censvalue(#) epsilon(#) keep(varlist) noshorten trans(numlist) wtstpm2 censcov(varlist) censdf(#) censvc(varlist) censvcdf(#)]
```

`stcrprep` is an `st` command, and the data must be `stset` before using it. All events
must be defined in the `failure` option of `stset`. The `id` option of `stset` is required.
The Kaplan–Meier method is used to estimate the censoring distribution unless you
specify the `wtstpm2` option.

3.2 Options

`events(varname)` specifies the variable that defines the different events. By default,
censored observations have the value 0. You can change this with the `censvalue()`
option. `events()` is required.

`byg(varlist)` calculates censoring weights separately by `varlist`.

`byh(varlist)` calculates left-truncation weights separately by `varlist`. 
\begin{quote}
\textbf{Competing risks}

\begin{itemize}
\item \texttt{censvalue(\#)} denotes a censored observation in the variable given in the \texttt{events()} option. The default is \texttt{censvalue(0)}.
\item \texttt{epsilon(\#)} specifies the value added to survival time when calculating probability of censoring to ensure that events occur before censoring. The default is \texttt{epsilon(0.000001)}.
\item \texttt{keep(varlist)} specifies names of variables to keep in the expanded dataset. This is generally a list of the variables that you want to include in the analysis.
\item \texttt{noshorten} specifies not to collapse over rows with equal weights.
\item \texttt{trans(numlist)} lists the transitions of interest, that is, the events contained in the \texttt{events()} option that you want to have as events of interest in the analysis. By default, all events are included. If only one event is of interest and the data are large, use this option to create a dataset only for the specific event of interest.
\end{itemize}

The following options fit a parametric model using \texttt{stpm2} for the censoring distribution.

\begin{itemize}
\item \texttt{wtstpm2} requests that the censoring distribution be estimated by fitting a flexible parametric survival model using \texttt{stpm2}.
\item \texttt{censcov(varlist)} lists covariates to include in the model for the censoring distribution.
\item \texttt{censdf(\#)} gives the degrees of freedom used for the baseline when using \texttt{stpm2} to obtain the censoring distribution. The default is \texttt{censdf(5)}.
\item \texttt{censtvc(varlist)} gives any variables to be included as time-dependent effects when using \texttt{stpm2} to estimate the censoring distribution.
\item \texttt{censtvcdf(\#)} gives the degrees of freedom used for any time-dependent effects when using \texttt{stpm2} to obtain the censoring distribution. The default is \texttt{censtvcdf(3)}.
\end{itemize}

\section{Examples 1}

\subsection{European Blood and Marrow Transplantation data}

To demonstrate the methods, we use data for 1,977 patients from the European Blood and Marrow Transplantation (EBMT) registry who received an allogeneic bone marrow transplantation. We measure time in days from transplantation to either relapse or death. There is only one covariate of interest, the EBMT risk score, categorized into three groups (low, medium, and high risk). The data are available as part of the \texttt{mstate} R package (de Wreede, Fiocco, and Putter 2011).

The estimated cause-specific CIFs for relapse and for death by risk group have been estimated with \texttt{stcompet} and are shown in figure 1. The more severe the risk group, the higher the probability is of both relapse and death.
Figure 1. Nonparametric estimates of the cause-specific CIF for relapse and death using \texttt{stcompet}

4.2 Using \texttt{stcrreg}

We will first apply the Fine and Gray (1999) model, implemented in \texttt{stcrreg}, so subsequent model results can be compared. The output below shows the model where relapse is the outcome of interest.
Competing risks

```
. stset time, failure(status==1) scale(365.25) id(patid) noshow
        id: patid
    failure event: status == 1
  obs. time interval: (time[_n-1], time]
     exit on or before: failure
   t for analysis: time/365.25

1977    total observations
     0   exclusions

1977  observations remaining, representing
1977  subjects
456  failures in single-failure-per-subject data
3796.057  total analysis time at risk and under observation
       at risk from t = 0
    earliest observed entry t = 0
    last observed exit t = 8.454483

. stcrreg i.score, compete(status==2) nolog noshow
  Competing-risks regression
          No. of obs = 1,977
        No. of subjects = 1,977
  Failure event : status == 1
             No. failed = 456
        No. competing = 836
    No. censored = 685
     Wald chi2(2) = 9.87
   Prob > chi2 = 0.0072
     (Std. Err. adjusted for 1,977 clusters in patid)

     _t  SHR     Std. Err.     z     P>|z|    [95% Conf. Interval]
        score
    2   1.271221  .1554323  1.96  0.050   1.000333  1.615465
    3   1.769853  .3238535  3.12  0.002  1.236465  2.533337
```

We estimate the subhazard ratios (with low risk as the reference group) to be 1.27 (95% confidence interval [1.00 to 1.62]) and 1.77 [1.24 to 2.53] for the medium- and high-risk groups, respectively. That these subhazard ratios are greater than one is to be expected, given the nonparametric estimates of the cause-specific CIF having higher values for the higher risk groups (figure 1). The corresponding estimated subhazard ratios for death are 1.77 [1.42 to 2.21] and 2.67 [1.97 to 3.62] (output not shown).

4.3 Using stcrprep

We will now expand the data with `stcrprep` and calculate the appropriate weights, so that we can use “standard” survival methods to estimate and model the cause-specific CIF.
The data first need to be \texttt{stset}, with all events defined as failures.

\begin{verbatim}
. stset time, failure(status==1,2) scale(365.25) id(patid)
    id: patid
    failure event: status == 1 2
    obs. time interval: (time[_n-1], time]
    exit on or before: failure
    t for analysis: time/365.25

1977  total observations
  0  exclusions

1977  observations remaining, representing
1977  subjects
1141  failures in single-failure-per-subject data
3796.057  total analysis time at risk and under observation
          at risk from t = 0
          earliest observed entry t = 0
          last observed exit t = 8.454483
\end{verbatim}

The data for subject 17 (\texttt{patid==17}) will be compared before and after using \texttt{stcrprep}. This subject died after 2.29 years and, in the original dataset, has one row of data, as seen below:

\begin{verbatim}
. list patid status _t0 _t _d if patid==17, noobs

  patid  status  _t0   _t   _d
     17       died  0 2.2888433  1
\end{verbatim}

We then run \texttt{stcrprep} to expand the data.

\begin{verbatim}
. stcrprep, events(status) keep(score) trans(1 2) byg(score)
\end{verbatim}

The \texttt{events()} option requires the variable defining all possible events and the censored value. The \texttt{trans()} option gives the transitions of the events of interest; here we are interested in the transitions to both relapse (\texttt{status = 1}) and death (\texttt{status = 2}). This is actually the default but is shown here for clarity. The \texttt{keep()} option lists variables to retain in the expanded data; usually, any covariates to be later analyzed are included here. The \texttt{byg()} option requests the censoring distribution to be estimated separately for the given groups. Because we are first going to obtain a separate nonparametric estimate of the cause-specific CIF in each group, the \texttt{byg()} option will estimate the censoring distribution separately in each group. After we use \texttt{stcrprep}, the number of rows has increased from 1,977 to 70,262.
The expanded data for subject 17 are shown below.

```
. list failcode patid status tstart tstop weight_c weight_t if patid==17,
>   > sepby(failcode) noobs //linesize(78)
```

<table>
<thead>
<tr>
<th>failcode</th>
<th>patid</th>
<th>status</th>
<th>tstart</th>
<th>tstop</th>
<th>weight_c</th>
<th>weight_t</th>
</tr>
</thead>
<tbody>
<tr>
<td>relapse</td>
<td>17</td>
<td>died</td>
<td>0.0000</td>
<td>2.28884</td>
<td>1.00000</td>
<td>1</td>
</tr>
<tr>
<td>relapse</td>
<td>17</td>
<td>died</td>
<td>2.28884</td>
<td>2.31622</td>
<td>0.99000</td>
<td>1</td>
</tr>
<tr>
<td>relapse</td>
<td>17</td>
<td>died</td>
<td>2.31622</td>
<td>2.32717</td>
<td>0.98497</td>
<td>1</td>
</tr>
<tr>
<td>relapse</td>
<td>17</td>
<td>died</td>
<td>2.32717</td>
<td>2.36003</td>
<td>0.97992</td>
<td>1</td>
</tr>
<tr>
<td>relapse</td>
<td>17</td>
<td>died</td>
<td>2.36003</td>
<td>2.55441</td>
<td>0.91392</td>
<td>1</td>
</tr>
<tr>
<td>relapse</td>
<td>17</td>
<td>died</td>
<td>2.55441</td>
<td>2.65845</td>
<td>0.89843</td>
<td>1</td>
</tr>
<tr>
<td>relapse</td>
<td>17</td>
<td>died</td>
<td>2.65845</td>
<td>2.89938</td>
<td>0.85142</td>
<td>1</td>
</tr>
<tr>
<td>relapse</td>
<td>17</td>
<td>died</td>
<td>2.89938</td>
<td>3.02806</td>
<td>0.80937</td>
<td>1</td>
</tr>
<tr>
<td>relapse</td>
<td>17</td>
<td>died</td>
<td>3.02806</td>
<td>3.18960</td>
<td>0.78176</td>
<td>1</td>
</tr>
<tr>
<td>relapse</td>
<td>17</td>
<td>died</td>
<td>3.18960</td>
<td>3.26626</td>
<td>0.74878</td>
<td>1</td>
</tr>
<tr>
<td>relapse</td>
<td>17</td>
<td>died</td>
<td>3.26626</td>
<td>3.62765</td>
<td>0.65847</td>
<td>1</td>
</tr>
<tr>
<td>relapse</td>
<td>17</td>
<td>died</td>
<td>3.62765</td>
<td>3.89870</td>
<td>0.59519</td>
<td>1</td>
</tr>
<tr>
<td>relapse</td>
<td>17</td>
<td>died</td>
<td>3.89870</td>
<td>3.97536</td>
<td>0.57881</td>
<td>1</td>
</tr>
<tr>
<td>relapse</td>
<td>17</td>
<td>died</td>
<td>3.97536</td>
<td>4.10951</td>
<td>0.55124</td>
<td>1</td>
</tr>
<tr>
<td>relapse</td>
<td>17</td>
<td>died</td>
<td>4.10951</td>
<td>4.39425</td>
<td>0.51163</td>
<td>1</td>
</tr>
<tr>
<td>relapse</td>
<td>17</td>
<td>died</td>
<td>4.39425</td>
<td>4.50103</td>
<td>0.47714</td>
<td>1</td>
</tr>
<tr>
<td>relapse</td>
<td>17</td>
<td>died</td>
<td>4.50103</td>
<td>4.69815</td>
<td>0.45968</td>
<td>1</td>
</tr>
<tr>
<td>relapse</td>
<td>17</td>
<td>died</td>
<td>4.69815</td>
<td>5.08419</td>
<td>0.37101</td>
<td>1</td>
</tr>
<tr>
<td>relapse</td>
<td>17</td>
<td>died</td>
<td>5.08419</td>
<td>5.22656</td>
<td>0.32235</td>
<td>1</td>
</tr>
<tr>
<td>relapse</td>
<td>17</td>
<td>died</td>
<td>5.22656</td>
<td>5.33607</td>
<td>0.30995</td>
<td>1</td>
</tr>
<tr>
<td>relapse</td>
<td>17</td>
<td>died</td>
<td>5.33607</td>
<td>5.97673</td>
<td>0.22772</td>
<td>1</td>
</tr>
<tr>
<td>relapse</td>
<td>17</td>
<td>died</td>
<td>5.97673</td>
<td>6.27515</td>
<td>0.20170</td>
<td>1</td>
</tr>
<tr>
<td>died</td>
<td>17</td>
<td>died</td>
<td>0.0000</td>
<td>2.28884</td>
<td>1.00000</td>
<td>1</td>
</tr>
</tbody>
</table>
```

The rows have been divided based on the failure of the newly created `failcode` variable. We will use this variable to fit different models depending on the event of interest. The variables `patid` and `status` are the same as in the nonexpanded data. The variables `tstart` and `tstop` give the times an individual starts and stops being at risk. They change within an individual when their weight, defined by variable `weight_c`, changes value. The `weight_t` gives the weights when there is left-truncation. Because there is no left-truncation in these data, they take the value 1 for all subjects at all times.

When the event of interest is a relapse, `failcode==1`. Because the subject with `patid==17` died after 2.29 years (that is, had a competing event), he or she is initially at risk until this time and should receive a weight of 1 in the analysis. After death, he or she is still kept in the risk set, but the weight decreases. The decrease is based on the conditional probability of being censored, which is estimated using a nonparametric (Kaplan–Meier) estimate of the censoring distribution. The weights only change at times when there is a failure for the event of interest and the value of censoring distribution has changed.

When the event of interest is death, `failcode==2`. Because this patient experienced the event of interest, he or she requires only one row of data.
4.4 Plotting the cause-specific CIF using \texttt{sts graph}

We can use \texttt{sts graph} to plot the cause-specific CIF. We first need to \texttt{stset} the data using the information on the weights contained in the variable \texttt{weights.c} by specifying \texttt{iweight}.

\begin{verbatim}
. generate event = status == failcode
. stset tstop [iw=weight_c], failure(event) enter(tstart) noshow
  failure event:  event != 0 & event < .
  obs. time interval:  (0, tstop]
  enter on or after:  time tstart
  exit on or before:  failure
  weight:  [iweight=weight_c]

  70262  total observations
  0  exclusions

  70262  observations remaining, representing
  1141  failures in single-record/single-failure data
  13820.402  total analysis time at risk and under observation
    at risk from t = 0
    earliest observed entry t = 0
    last observed exit t = 8.454483
\end{verbatim}

We first create the variable \texttt{event}. This is defined as one if the event of interest occurs and zero otherwise. Because we have split time data, we need to give information on the start time (\texttt{tstart}) and stop time (\texttt{tstop}) of each row of data.

We use \texttt{sts graph} as usual but with the \texttt{failure} option—because we are interested in the probability of relapse as opposed to the probability of not having a relapse (which includes the probability of death). For example, the cause-specific CIF for relapse can be plotted as follows:

\begin{verbatim}
. sts graph if failcode==1, by(score) failure
  ytitle("Probability of relapse")
  xtitle("Years since transplantation")
  ylabel(0(0.1)0.5, angle(h) format(%3.1f))
  legend(order(1 "Low risk" 2 "Medium risk" 3 "High risk")
         cols(1) ring(0) pos(5))
  scheme(sj) name(cif_relapse, replace)
\end{verbatim}

The corresponding graph can be seen in figure 2.
This graph is the same as the graph in the left-hand panel of figure 1, except that the lines extend to the maximum censoring time in each group. Alternatively, \texttt{sts gen} can generate the cause-specific CIF, and we can plot this with appropriate \texttt{if} statements to control the maximum follow-up time for each line.

### 4.5 Testing for differences in the CIF using \texttt{sts test}

Gray (1988) developed a modified log-rank test for testing differences between cause-specific CIFs. We can perform a similar, but slightly different, approach by applying a weighted log-rank test with weights defined in (3), using \texttt{sts test}. Further work is required to evaluate the performance of this test under a range of different assumptions, but some simulation studies show that it appears to have good statistical properties (Mertsching 2013).

\begin{verbatim}
. stcox if failcode==1
Log-rank test for equality of survivor functions

<table>
<thead>
<tr>
<th></th>
<th>Events observed</th>
<th>Events expected</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low</td>
<td>79</td>
<td>99.64</td>
</tr>
<tr>
<td>Medium</td>
<td>328</td>
<td>324.33</td>
</tr>
<tr>
<td>High</td>
<td>49</td>
<td>32.04</td>
</tr>
<tr>
<td>Total</td>
<td>456</td>
<td>456.00</td>
</tr>
</tbody>
</table>

\text{chi2(2) = 13.37}
\text{Pr>chi2 = 0.0012}
\end{verbatim}

There is strong evidence of a difference in the probability of relapse between the groups.
4.6 Proportional subhazards model using stcox

When we previously used stcrprep, we calculated the weights separately in each risk group because we were interested in calculating the cause-specific CIF separately for each of these groups. We now reload the original data and run stcrprep again, but this time, we calculate the censoring distribution used to generate the weights for the cohort as a whole—that is, we do not use byg(). This is because we want to fit the same model as that fit by stcrreg, which does not calculate weights separately by subgroup.

```
. stset time, failure(status==1,2) scale(365.25) id(patid) noshow
   id: patid
  failure event: status == 1 2
   obs. time interval: (time[_n-1], time]
    exit on or before: failure
   t for analysis: time/365.25

1977  total observations
      0  exclusions

1977  total analysis time at risk and under observation
      3796.057  total analysis time at risk and under observation
        at risk from t = 0
        earliest observed entry t = 0
        last observed exit t = 8.454483

. stcrprep, events(status) keep(score) trans(1 2)
```

After defining the event indicator and using stset, we can fit a proportional subhazards model for relapse as follows:

```
. generate event = status == failcode
. stset tstop [iw=weight_c], failure(event) enter(tstart) noshow
     failure event: event != 0 & event < .
   obs. time interval: (0, tstop]
  enter on or after: time tstart
 exit on or before: failure
   weight: [iweight=weight_c]

127730  total observations
       0  exclusions

127730  total analysis time at risk and under observation
       14418.735  total analysis time at risk and under observation
         at risk from t = 0
        earliest observed entry t = 0
        last observed exit t = 8.454483
```
. stcox i.score if failcode == 1, nolog

Cox regression -- Breslow method for ties

No. of subjects = 72,880  Number of obs = 72,880
No. of failures = 456
Time at risk = 6026.27434

LR chi2(2) = 9.63  Log likelihood = -3333.3112
Prob > chi2 = 0.0081

|      | Haz. Ratio | Std. Err. |     z  |   P>|z|   | [95% Conf. Interval] |
|------|------------|-----------|-------|-------|----------------------|
| score|            |           |       |       |                      |
| 2    | 1.271235   | .1593392  | 1.91  | 0.056 | .9943389  1.628238   |
| 3    | 1.769899   | .3219273  | 3.14  | 0.002 | 1.239148  2.52798    |

The output gives the subhazard ratios for the medium- and high-risk groups. The parameter estimates are the same as those produced by stcrreg in section 4.2 to four decimal places. The standard errors are slightly different because we did not use a clustered sandwich estimator: Geskus (2011) showed that the sandwich estimator was asymptotically unbiased but less efficient than using the standard errors derived with the observed information matrix. If one uses pweights rather than iweights, along with using stset and the vce(cluster patid) option for the stcox command, then the standard errors are the same as stcrreg to four decimal places.

4.7 Time gains for large datasets

Fitting Fine and Gray (1999) models for large datasets is computationally intensive because each subject with a competing event has time-dependent weights. One of the advantages of using stcrprep is that after we expand the data and apply the weights, it can be much quicker to fit the models. As an example, the EBMT data were increased by a factor of 10 (that is, \( N = 19770 \)), and small random numbers were added to the event times to ensure there were no ties. Fitting the same model as above using stcrreg took just under 47 minutes on an Intel I7 @ 2.00GHz running Stata/MP2 13 on 64-bit Windows 7. Running stcrprep took just over 15 minutes, and fitting the model using stcox took 1.5 seconds. The key point here is that stcrprep needs to be run only once, making the process of model building and comparison much quicker.

4.8 Testing the proportional-subhazards assumption

One of the advantages of fitting proportional subhazards models after using stcrprep is that we can use some of the standard tools developed for the Cox model. For example, we can test the assumptions of proportional subhazards with the test developed using Schoenfeld residuals.
There is strong evidence of nonproportional subhazards for both the medium- and high-risk groups, confirmed by the Schoenfeld residual plots produced using the `estat phtest` command with the `plot()` option. Figure 3 shows the graph for the medium-risk group.

![Test of PH Assumption](image)

**Figure 3.** Assessing the proportional-subhazards assumption using Schoenfeld residuals

See section 6.2 for how nonproportional subhazards can be modeled parametrically.

### 4.9 Estimation within one model

If we are interested in the cause-specific CIF for both events, we can fit two separate models. Alternatively, we can estimate the same parameters from fitting one model, stratified by event type. The code for this is
Competing risks

```
.stcox ibn.failcode#i(2 3).score, strata(failcode) nolog
Stratified Cox regr. -- Breslow method for ties
No. of subjects = 100,732  Number of obs = 100,732
No. of failures = 1,141
Time at risk = 11077.05194
Log likelihood = -8337.8823  LR chi2(4) = 55.30
   Prob > chi2 = 0.0000

       _t  Haz. Ratio  Std. Err.     z  P>|z|  [95% Conf. Interval]
failcode#  score
    1  2  1.271235  .1593392  1.91  0.056  .9943389  1.625238
    1  3  1.769899  .3219273  3.14  0.002  1.239148  2.52798
    2  2  1.768168  .2002336  8.62  0.000  1.41622  2.207581
    2  3  2.671037  .4085735  6.42  0.000  1.97914  3.604819
```

Stratified by `failcode`

In this case, the estimates are identical to fitting two separate models. The reason for modeling in this way is that in more complex models, the effect of some covariates can be forced to be identical for different event types. This may be beneficial when one of the events is rare.

### 4.10 Fitting cause-specific hazards models

We can fit a cause-specific hazards model after using `stcrprep` rather than reloading the original data by restricting the analysis to the first row of data for each individual. For example, we can fit cause-specific Cox proportional hazards models for relapse as follows:

```
.bysort failcode patid (tstop): generate firstrow = _n == 1
.stcox i.score if failcode == 1 & firstrow, nolog
Cox regression -- Breslow method for ties
No. of subjects = 1,977  Number of obs = 1,977
No. of failures = 456
Time at risk = 3796.057495
Log likelihood = -3161.9912  LR chi2(2) = 33.00
   Prob > chi2 = 0.0000

       _t  Haz. Ratio  Std. Err.     z  P>|z|  [95% Conf. Interval]
score
   2  1.539886  .193302  3.44  0.001  1.20403  1.969428
   3  2.959512  .5413436  5.34  0.000  2.06786  4.235638
```

We can also fit stacked cause-specific hazards models similar to the model in section 4.9 by stratifying by `failcode`. 
There are a number of situations where fitting a parametric model for the cause-specific CIF is beneficial. We can apply the general ideas of Geskus (2011) regarding data expansion and weighting to parametric models. Previous parametric models for the cause-specific CIF have required the CIFs for all the \( K \) causes to be modeled simultaneously and have used more standard survival distributions such as the Weibull and Gompertz distributions (Jeong and Fine 2006, 2007). Here we extend the ideas of the nonparametric models to allow modeling of a cause-specific CIF by incorporating the censoring weights. For the nonparametric estimates and the semiparametric Fine and Gray (1999) model, the censoring distribution has to be evaluated at the event times of the event of interest. When one fits a parametric model for the CIF using weighted maximum likelihood, the censoring distribution is a continuous function of time. The contribution of the \( i \)th subject to the log likelihood is

\[
\ln L_i = d_{1i} \ln \{ h_1^s(t_i) \} - (1 - d_{2i}) H_1^s(t_i) - d_{2i} \int_0^{t_i} w(u) h_1^s(u) du
\]

where \( h_1^s(t) \) is the subhazard function and \( H_1^s(t) \) is the cumulative subhazard function of the event of interest. To implement this, we can approximate the integral in (5) by splitting the time scale after a competing event into a finite number of intervals, assuming the weight is constant within each interval. The number of intervals per subject, \( N_{ij} \), will vary—because the length of follow-up will depend on the time of their competing event. The likelihood becomes

\[
\ln L_i = d_{1i} \ln \{ h_1^s(t_i) \} - (1 - d_{2i}) H_1^s(t_i) - d_{2i} \sum_{j=1}^{N_{ij}} w_{ij} \{ H_1^s(t_{ij}) - H_1^s(t_{i(j-1)}) \}
\]

where \( t_i \) denotes the time to an event or censoring with binary event indicators \( d_{1i} \) and \( d_{2i} \) for the primary (event 1) and competing (event 2) events, and \( t_{ij} \) gives the \( j \)th follow-up time after individual \( i \) has had a competing event. The weights, \( w_{ij} \), give the conditional probability of censoring at the \( j \)th follow-up time for individual \( i \). We can use the Kaplan–Meier estimate, but instead, we use a flexible parametric survival model. This makes evaluating the censoring distribution relatively simple because there is an analytic expression for the conditional censoring distribution. For further details of the parametric approach, see Lambert, Wilkes, and Crowther (Forthcoming).

The advantage of this approach is that after restructuring the data by creating extra rows for those with a competing event and calculating the weights, we can use standard parametric survival models to estimate the cause-specific CIF as long as the software allows weighted likelihood estimates and delayed entry. In the following section, we use flexible parametric survival models with the \texttt{stpm2} command.
6 Examples 2

6.1 Parametric proportional subhazards model

In the following code, using the `wtstpm2` option with `stcrprep` requests the censoring distribution be estimated parametrically using a flexible parametric survival model, with the `censdf(4)` option requesting four degrees of freedom (five knots) to estimate the survival function of the censoring distribution. The `every(0.25)` option means that the weights for those with a competing event will be estimated every 0.25 years after their competing event.

```plaintext
. stset time, failure(status==1,2) scale(365.25) id(patid) noshow
   id: patid
   failure event: status == 1 2
   obs. time interval: (time[_n-1], time]
   exit on or before: failure
   t for analysis: time/365.25

1977 total observations
   0 exclusions

1977 observations remaining, representing
1977 subjects
1141 failures in single-failure-per-subject data
3796.057 total analysis time at risk and under observation
   at risk from t = 0
   earliest observed entry t = 0
   last observed exit t = 8.454483

. stcrprep, events(status) keep(score) trans(1 2) wtstpm2 censdf(4) every(0.25)
```

After using `stset` in the same way as in section 4.6, we can fit a parametric model to model the cause-specific CIF for relapse. Below we fit a flexible parametric survival model using the `stpm2` command. This means that a restricted cubic spline function is used to estimate the underlying subhazard function. We could also use other parametric models, such as those incorporated into `streg`.

```plaintext
. stpm2 i.score if failcode == 1, df(4) scale(hazard) nolog eform
   note: delayed entry models are being fitted
Log likelihood = -1678.9025   Number of obs  =  23,673

| exp(b) | Std. Err. | z  | P>|z| | [95% Conf. Interval] |
|--------|-----------|----|------|----------------------|
| xb     |           |    |      |                      |
| score  |           |    |      |                      |
| 2      | 1.270361  | .1592233 | 1.91 | 0.056 | .9936652 1.624105 |
| 3      | 1.769961  | .3219309 | 3.14 | 0.002 | 1.239203 2.528048 |
| _rcs1  | 1.427308  | .02829  | 17.95 | 0.000 | 1.372923 1.483846 |
| _rcs2  | 1.124042  | .0158208 | 8.31 | 0.000 | 1.093458 1.155482 |
| _rcs3  | 1.038305  | .0135277 | 2.89 | 0.004 | 1.012127 1.06516  |
| _rcs4  | .9690918  | .0078049 | -3.90 | 0.000 | .9539146 .9845105 |
| _cons  | .2090643  | .0235494 | -13.89 | 0.000 | .1676481 .250712 |
```

```plaintext
```
The subhazard ratios are very similar to those obtained when using a weighted Cox model in section 4.6, as are the associated standard errors. We estimated five additional parameters, giving a parametric form for the cause-specific CIF. We can obtain predictions of the cause-specific CIF using the `failure` option of the `predict` command for `stpm2`.

```stata
. range temptime 0 7.2 200
(39,027 missing values generated)
. predict CIF1, failure at(score 1) timevar(temptime) ci
. predict CIF2, failure at(score 2) timevar(temptime) ci
. predict CIF3, failure at(score 3) timevar(temptime) ci
```

Because the number of observations is now fairly large, we use the `timevar()` option of the `predict` command. We first create a new variable using `range` to create another variable, `temptime`, with 200 observations equally spaced between 0 and 7.2. Using `timevar()` in conjunction with the `at()` option enables us to plot predictions for 200 observations, rather than the default of 39,227 values of `t`. The `ci` option requests calculating confidence intervals with the names `newvar_lci` and `newvar_uci`. Figure 4 shows the baseline cause-specific CIF (score = 1) with a 95% confidence interval plotted in panel (a) with the cause-specific CIFs for all three groups plotted in panel (b).

![Figure 4](image)

Figure 4. Prediction of cause-specific CIFs for relapse after using `stpm2`. Panel (a) shows the baseline cause-specific CIF with a 95% confidence interval, and panel (b) shows the cause-specific CIFs for the three risk groups.
Various other predictions are available. For example, the following code calculates
the difference in cause-specific CIFs for relapse between the high- and the low-risk groups
with a 95% confidence interval.

```
predict CIF_diff, sdiff1(score 1) sdiff2(score 3) timevar(temptime) ci
```

Figure 5 shows the plotted predictions. This gives an estimate of the difference
in absolute risk of relapse between the two risk groups, accounting for the competing
event of death. The reciprocal of the difference in CIFs estimates the number needing
treatment, accounting for competing risks (Gouskova et al. 2014).

![Figure 5. Prediction of difference in cause-specific CIFs for relapse between high- and low-risk groups after using stpm2](attachment:image)

6.2 Nonproportional subhazards

One of the advantages of the flexible parametric survival approach is the ability to
model nonproportional hazards. These advantages carry over to the models fit after
using stcrprep to allow for nonproportional subhazards. We fit time-dependent effects
by adding interactions between the covariate of interest and some restricted cubic splines
to the model. For more details of fitting time-dependent effects for these types of models,
see Lambert and Royston (2009) and Royston and Lambert (2011). We can fit the time-
dependent effects through the tvce() and dfvc() options. Below is the code to fit a
nonproportional subhazards model for relapse (proportional subhazards for death was
a reasonable assumption).
The tvc() option of stpm2 does not allow factor variables, so we create dummy variables using the tab command. The dftvc(1) option requests that the time-dependent effects be a simple function of log(time). Having more degrees of freedom would model the time dependence using restricted cubic splines in the same way as for the baseline, but one degree of freedom is sufficient here. After fitting the model, we can obtain predictions of the cause-specific CIF in the same way as above.

```
ipredict CIF1_tvc, failure zeros timevar(temptime) ci
ipredict CIF2_tvc, failure at(score2 1) zeros timevar(temptime) ci
ipredict CIF3_tvc, failure at(score3 1) zeros timevar(temptime) ci
```

The zeros option forces all covariates to be zero, except those listed in the at() option. We can then plot the predictions as before. Figure 6 shows the predictions from both the proportional subhazards and nonproportional subhazards models, with the empirical cause-specific CIF shown for comparison (obtained using stcompet). The predicted values from the nonproportional subhazards model give a much closer fit to the empirical cause-specific CIFs.
6.3 Models on other scales

The models fit with stpm2 use the scale(hazard) option, which requests that models be fit on the log cumulative-subhazard scale. When one models cause-specific hazards, other options with stpm2 include proportional odds models (scale(odds)), probit models (scale(normal)), and the Aranda–Ordaz family of link function (scale(theta)). For more details on these models, see Royston and Lambert (2011). One can use all of these link functions for modeling the cause-specific CIF. We can illustrate this with a proportional odds model. This idea of using a logit link of the cause-specific CIF is similar to ideas published by Fine (1999), who used semiparametric transformation models, and by Klein and Andersen (2005), who used generalized estimating equations applied to pseudovalues to model the cause-specific CIF. The latter is implemented via the stpci command (Parner and Andersen 2010).

Logit links give a proportional odds model for the CIF and make the assumption that the odds of an event with cause $k$ occurring up to time $t$ (for any $t$) are proportional across covariate patterns. The model is

$$\frac{F_k(t|x)}{1 - F_k(t|x)} = \frac{F_{0k}(t)}{1 - F_{0k}(t)} \exp(x\beta)$$
An example of fitting a proportional odds model for the cause-specific CIF for relapse using \texttt{stpm2} with \texttt{scale(odds)} is

\begin{verbatim}
  . stpm2 i.score, scale(odds) df(4) nolog eform
  note: delayed entry models are being fitted
  Log likelihood = -4032.1604  Number of obs = 39,227

  exp(b) Std. Err.  z   P>|z|    [95% Conf. Interval]
  xb
    score
    2  1.657222  .1566403  5.34 0.000  1.376972  1.994509
    3  2.674231  .3717455  7.08 0.000  2.036447  3.511759
    _rcs1  1.371692  .0155809 27.82 0.000  1.341491  1.402572
    _rcs2  1.174037  .0109122 17.26 0.000  1.152843  1.195623
    _rcs3  1.031093  .0079927  3.95 0.000  1.015447  1.046678
    _rcs4 .9829622  .0047848 -3.53 0.000  .9736288  .9923851
    _cons .2555708  .0218471 -15.96 0.000  .2161462  .3021865
\end{verbatim}

The odds of recurrence up to time $t$ is 2.67 higher in the high-risk group when compared with the low-risk group. This is a proportional odds model, so the odds ratio is assumed to apply for any $t$. This assumption could be relaxed through the \texttt{tvc()} and \texttt{dftvc()} options in the same way as in section 6.2.

We add a new link function to \texttt{stpm2} using \texttt{scale(log)}, which uses a log-link function, $\log\{1 - S(t)\}$. When used to model the CIF, the parameter estimates are log relative-risks of the cause-specific CIF, which has the advantage of a simple interpretation. This link function has previously been used with the direct binomial approach to model the cause-specific CIF (Gerds, Scheike, and Andersen 2012). Use the link function with a degree of caution because it does not constrain the CIF to be below one. However, for many datasets, this will not lead to a problem in estimation.

The following code uses a log-link function (\texttt{scale(log)}):

\begin{verbatim}
  . stpm2 i.score, scale(log) df(4) nolog eform
  note: delayed entry models are being fitted
  Log likelihood = -4038.4053  Number of obs = 39,227

  exp(b) Std. Err.  z   P>|z|    [95% Conf. Interval]
  xb
    score
    2  1.421111  .1041615  4.79 0.000  1.230944  1.640686
    3  1.870372  .1791209  6.54 0.000  1.550281  2.286553
    _rcs1  1.284623  .0133339 28.39 0.000  1.262619  1.306727
    _rcs2  1.169748  .0101003 18.16 0.000  1.150119  1.190314
    _rcs3  1.018622  .0072396  2.60 0.009  1.004531  1.032712
    _rcs4 .9817386  .0040125 -4.51 0.000  .9736288  .9900484
    _cons .2067656  .0139752 -23.32 0.000  .1811115  .2360537
\end{verbatim}
The relative risk is 1.87 when comparing the high- and low-risk groups; that is, the probability of relapse is 87% higher in the high-risk group than in the low-risk group. As for the proportional odds model, it is assumed that this relative effect is the same over time; that is, the 87% increase is assumed to apply to the cumulative probability of an event for any $t$. We now relax this assumption through the `tvc()` and `dftvc()` options. We then use the `partpred` command (Lambert 2010) to generate partial prediction $rr$.

```
. stpm2 score2 score3, scale(log) df(4) nolog tvc(score2 score3) dftvc(1)
```

```
Log likelihood = -4027.9917 Number of obs = 39,227
```

| Coef. | Std. Err. | z    | P>|z|  | [95% Conf. Interval] |
|--------|-----------|------|------|----------------------------|
| $xb$   |           |      |      |                            |
| score2 | 0.3558933 | 0.0730771 | 4.87 | 0.000 | 0.2126649 - 0.4991217 |
| score3 | 0.6257833 | 0.0965325 | 6.48 | 0.000 | 0.436583 - 0.8149835 |
| _rcs1  | 0.363938  | 0.0274412 | 12.26 | 0.000 | 0.28261 - 0.3901777 |
| _rcs2  | 0.1591966 | 0.0086285 | 18.45 | 0.000 | 0.1422842 - 0.1761071 |
| _rcs3  | 0.192613  | 0.0072134 | 26.7  | 0.000 | 0.181232 - 0.3033993 |
| _rcs4  | -0.0174134| 0.0040943 | -4.25 | 0.000 | -0.025438 - 0.0099988 |
| _rcs_score21 | -0.0932564 | 0.0291333 | -3.20 | 0.001 | -0.1503556 - 0.0361551 |
| _rcs_score31 | -0.1515132 | 0.0337963 | -4.48 | 0.000 | -0.2177526 - 0.0852737 |
| _cons  | -1.581555 | 0.0671193 | -23.56 | 0.000 | -1.713106 - 1.450003 |

```
. partpred rr if score3==1 & failcode==1, for(score3 _rcs_score3*)
> ci(rr_lci rr_uci) eform
```

(36,492 missing values generated)

Figure 7 shows the plot for the time-dependent relative risk.

![Figure 7. Relative absolute risk predicted from flexible parametric survival model with log-link function](image-url)
Given that this is a ratio of a cumulative probability, the time-dependent relative risk is unlikely to change rapidly as \( t \) increases.

One can still estimate differences in cause-specific CIFs from models on other scales using \( \text{sdiff1()} \) and \( \text{sdiff2()} \) in the same way as in section 6.1, as well as obtain the time-dependent subhazard ratios using \( \text{hrnumerator} \) and \( \text{hrdenominator} \) and the subhazard differences using \( \text{hdiff1} \) and \( \text{hdiff2} \).

### 7 Conclusion

I have described the use of \texttt{stcrprep} to restructure survival data so standard survival analysis tools can estimate and model competing-risks data. The command has similar functionality to \texttt{crprep} in R but has some additional options that are useful for fitting parametric survival models and also useful in larger datasets.

One important advantage of using \texttt{stcrprep} followed by \texttt{stcox} rather than \texttt{stcrreg} to fit a Fine and Gray (1999) model is that it is faster. This is the case for fitting a single model, but the main advantage comes from fitting multiple models to the same dataset—for example, during a model-selection process—because the data expansion and generation of the weights need to be done only once. A second advantage is that some of the standard tools used in theory developed for the Cox model can be used for the competing-risks model—for example, the use of Schoenfeld residuals to assess the proportional-subhazards assumption.

One can also model the censoring distribution and thus allow the weights to depend on covariates. When the \texttt{byg()} option is specified, weights are based on Kaplan–Meier estimates. Modeling the censoring distribution is done through the \texttt{censcov()}, \texttt{censtvc()}, and \texttt{censtvcdf()} options. When using \texttt{stcrreg}, one cannot allow the censoring distribution to depend on covariates. Modeling the censoring distribution is unresolved, and a common censoring distribution is often assumed. Gerds, Scheike, and Andersen (2012) and Scheike, Zhang, and Gerds (2008) have advocated modeling the censoring distribution, but further research is required to understand in what situations one should do this. If one jointly models the cause-specific CIF of all causes under consideration, then one does not need to model the censoring distribution (Jeong and Fine 2007).

The \texttt{stcrprep} command allows a parametric model for the cause-specific CIF to be fit. Using the flexible parametric modeling framework with \texttt{stpm2} enables a flexible parametric version of the Fine and Gray (1999) model with the ability to directly obtain predictions of the cause-specific CIF both in and out of sample. These models also relax the proportional-subhazards assumption by fitting interactions with time or modeling on alternative scales by using different link functions.

In this article, I have not covered situations where there is left-truncation or delayed entry. Left-truncation subjects with a competing event need to be upweighted to represent those that could potentially enter the study after the competing event. Geskus (2011) used weights based on the distribution of entry times. This is currently imple-
mented when using a Kaplan–Meier-based estimate of the distribution of entry times but not for parametric models. This will hopefully be updated in a future release.

8 References


**About the author**

Paul C. Lambert is a professor of biostatistics at the University of Leicester, UK. He also works part-time at the Department of Medical Epidemiology and Biostatistics, Karolinska Institutet, Stockholm, Sweden. His main interest is in the development and application of methods in population-based cancer research.
biasplot: A package to effective plots to assess bias and precision in method comparison studies

Patrick Taffé
Institute for Social and Preventive Medicine
University of Lausanne
Lausanne, Switzerland
Patrick.Taffe@chuv.ch

Mingkai Peng
Department of Community Health Sciences
University of Calgary
Calgary, Canada
Mingkai.peng@ucalgary.ca

Vicki Stagg
Calgary Statistical Support
Calgary, Canada
Vicki@calgarystatisticsupport.com

Tyler Williamson
Department of Community Health Sciences
University of Calgary
Calgary, Canada
Tyler.williamson@ucalgary.ca

Abstract. Bland and Altman’s (1986, Lancet 327: 307–310) limits of agreement have been used in many clinical research settings to assess agreement between two methods of measuring a quantitative characteristic. However, when the variances of the measurement errors of the two methods differ, limits of agreement can be misleading. biasplot implements a new statistical methodology that Taffé (Forthcoming, Statistical Methods in Medical Research) recently developed to circumvent this issue and assess bias and precision of the two measurement methods (one is the reference standard, and the other is the new measurement method to be evaluated). biasplot produces three new plots introduced by Taffé: the “bias plot”, “precision plot”, and “comparison plot”. These help the investigator visually evaluate the performance of the new measurement method. In this article, we introduce the user-written command biasplot and present worked examples using simulated data included with the package. Note that the Taffé method assumes there are several measurements from the reference standard and possibly as few as one measurement from the new method for each individual.

Keywords: gr0068, biasplot, limits of agreement, differential bias, proportional bias, Bland–Altman’s plot, method comparison, measurement, empirical Bayes, BLUP
1 Introduction

Clinical researchers frequently use Bland and Altman’s (1986) limits of agreement (LoA) to evaluate the agreement between two methods for measuring quantitative characteristics. Often this is motivated by a new, perhaps less expensive or easier, method of measurement against an established reference standard. To evaluate the comparability of the methods, the investigator collects measurements—one or several—from each method for a set of subjects. The investigator then computes Bland and Altman’s LoA by adding and subtracting 1.96 times the estimated standard deviation to the mean differences. A scatterplot of the differences versus the means of the two variables with the LoA superimposed is then used to visually appraise the degree of agreement and quantify the magnitude. Further, the investigator adds a regression of the differences as a function of the means to the plot to indicate whether there is a bias and the direction of that bias (Bland and Altman 1999).

However, Bland and Altman’s plot may be misleading when the variances of the measurement error for each method differ from one another. When this is the case, the regression line may show an upward or a downward trend when there is no bias or a zero slope when there is a bias. The literature has previously shown this problem. However, to the best of our knowledge, no simple-to-use and effective plots that evaluate bias and precision have been presented as an alternative (Hopkins 2004; Krouwer 2008; Carstensen, Simpson, and Gurrin 2008; Ludbrook 2010b,a; Carstensen 2010).

However, the purpose of this article is not a careful review of the literature. Interested readers should look at Nawarathna and Choudhary (2015) and the references therein for a recent review of measurement error. Rather, we will present the implementation of Taffé’s (Forthcoming) method, which extends previously published methods to the setting of heteroskedastic measurement errors, particularly when heteroskedasticity is a function of the latent trait. We will perform the estimation in two steps, using an empirical Bayes approach to identify and quantify the amount of differential and proportional bias. Further, Taffé introduced two new plots, the “bias plot” and the “precision plot”, to aid in assessing the new measurement method. These plots are not afflicted with the same issues as LoA plots but are still easily interpreted. The Taffé method requires that several measurements be made with the reference standard for each individual (usually more than five) and possibly only one measurement with the new method. The Taffé method allows each individual to have a different number of repeated measurements by each method and is applicable in all circumstances with or without differential or proportional bias and when the measurement errors are either homoskedastic or heteroskedastic.

2 The measurement error model

2.1 Formulation of the model

For a full presentation of the methodological theory, see Taffé (Forthcoming). Below we present an abridged version of the methods. Consider the measurement error model
**Plots to assess bias and precision in method comparison studies**

\[ y_{1ij} = \alpha_1 + \beta_1 x_{ij} + \epsilon_{1ij} \quad \epsilon_{1ij}|x_{ij} \sim N \{0, \sigma^2_{\epsilon_1}(x_{ij};\theta_1)\} \]

\[ y_{2ij} = \alpha_2 + \beta_2 x_{ij} + \epsilon_{2ij} \quad \epsilon_{2ij}|x_{ij} \sim N \{0, \sigma^2_{\epsilon_2}(x_{ij};\theta_2)\} \]

\[ x_{ij} \sim f_x(\mu_x, \sigma^2_x) \]

where \( y_{1ij} \) is the \( j \)th replicate measurement by method 1 on individual \( i \), \( j = 1, \ldots, n_i \) and \( i = 1, \ldots, N \), \( y_{2ij} \) is obtained by method 2, \( x_{ij} \) is a latent variable with density \( f_x \) representing the true unknown trait, and \( \epsilon_{1ij} \) and \( \epsilon_{2ij} \) represent measurement errors by methods 1 and 2. We assume the variances of these errors, that is, \( \sigma^2_{\epsilon_1}(x_{ij};\theta_1) \) and \( \sigma^2_{\epsilon_2}(x_{ij};\theta_2) \), are heteroskedastic and increase with the level of the true latent trait, \( x_{ij} \), in a way to be precisely specified later, depending on the vectors of unknown parameters \( \theta_1 \) and \( \theta_2 \). For the reference method, \( \alpha_2 = 0 \) and \( \beta_2 = 1 \), whereas for method 1, we must estimate the differential \( \alpha_1 \) and proportional \( \beta_1 \) biases from the data. The mean value of the latent variable \( x_{ij} \) is \( \mu_x \), and its variance is \( \sigma^2_x \). When method 2 is the reference standard and method 1 the new method for evaluation, the model reduces to

\[ y_{1ij} = \alpha_1 + \beta_1 x_i + \epsilon_{1ij} \quad \epsilon_{1ij}|x_i \sim N \{0, \sigma^2_{\epsilon_1}(x_i;\theta_1)\} \]

\[ y_{2ij} = x_i + \epsilon_{2ij} \quad \epsilon_{2ij}|x_i \sim N \{0, \sigma^2_{\epsilon_2}(x_i;\theta_2)\} \]

\[ x_i \sim f_x(\mu_x, \sigma^2_x) \]

Note that this measurement error model is slightly different from the classical measurement error model; the heteroskedasticity depends on the latent trait and not on an observed average (Dunn 2004).

### 2.2 Estimation of the model

The estimation process has two steps:

**Estimation step 1**

Other methods treat \( x_i \) as a nuisance parameter and attempt to integrate it out from the joint likelihood function. The Taffé method fits the regression model for \( y_{2ij} \) using marginal maximum likelihood, allowing the variance of \( \epsilon_{2ij} \) to be different for each decile of the empirical distribution of \( y_{2i} \) (that is, the mean of the individual repeated measurements \( y_{2i} \) is used as a rough approximation to \( x_i \)). Then, following an empirical Bayes approach, we predict \( x_i \) from the mean of its posterior distribution (that is, the mean of the conditional distribution of \( x_i \) given the vector \( y_{2i} \) of observations for individual \( i \) by method 2), which is the best linear unbiased prediction (BLUP) for \( x_i \).

\[ \widehat{x}_i = E(x_i|y_{2i}) = \int x_i \frac{f_{y_{2i}}(y_{2i}|x_i)f_x(x_i)}{\int f_{y_{2i}}(y_{2i}|x_i)f_x(x_i)dx_i} dx_i \]
For the sake of notational convenience, we have suppressed the dependence of the density functions \( f_{y2} \) and \( f_x \) from their parameters, which have been estimated by maximum likelihood.

When \( f_x \) is the normal density, (2) is

\[
\hat{x}_i = \sigma_x^2 \mu' V_i^{-1} (y_{2i} - \mu \hat{\mu}_x) + \hat{\mu}_x
\]

where \( \mu \) is an \( n \) vector of ones and \( V_i = \sigma_x^2 \mu' + \text{diag} \{ \sigma_x^2 (x_i; \theta_2) \} \) is the variance–covariance matrix of \( y_{2i} \).

It is desirable to have a smooth estimate of the heteroskedasticity that does not depend on \( y_{2i} \) but rather on \( \hat{x}_i \), the BLUP for \( x_i \). Therefore, Taffé suggests an approach similar to that of Bland and Altman (1999) by regressing the absolute values of the residuals \( \hat{e}_{2ij} \) from the linear regression model \( y_{2ij} = \alpha^*_2 + \beta^*_2 \hat{x}_i + \epsilon^*_{2ij} \) on \( \hat{x}_i \) by ordinary-least squares (OLS) to create a smooth estimate of the heterogeneous variance:

\[
|\hat{e}_{2ij}| = \hat{\theta}_2^{(0)} + \hat{\theta}_2^{(1)} \hat{x}_i + v_{ij}
\]

Under the normality assumption, \( |\epsilon^*_{2ij}| \) follows a half-normal distribution with mean \( E(|\epsilon^*_{2ij}|) = \sigma_{\epsilon^*} (\hat{x}_i; \theta_2) \sqrt{2/\pi} \). Therefore, we obtain a smooth standard-deviation estimate as follows:

\[
\hat{\sigma}_{\epsilon^*} (\hat{x}_i; \theta_2) = \sqrt{\pi/2} \sqrt{\hat{\theta}_2^{(0)} + \hat{\theta}_2^{(1)} \hat{x}_i}
\]

Note that Taffé suggests that the form of the heterogeneity need not be a straight line; we may consider other heterogeneity structures, and a graphical representation of \( |\hat{e}_{2ij}| \) versus \( \hat{x}_i \) provides a good start to visually check the plausibility of the straight-line model. It may be useful to assess the fit using a scatterplot of \( y_{2ij} \) versus \( \hat{x}_i \), with the estimated regression line and the 95% prediction limits computed as \( \hat{\alpha}_2^* + \hat{\beta}_2^* \hat{x}_i \pm 2\hat{\sigma}_{\epsilon^*} (\hat{x}_i; \theta_2) \).

**Estimation step 2**

The second stage of the estimation process involves the estimation of the regression equation for \( y_{1ij} \) in (1) and estimation of the differential (\( \alpha_1 \)) and proportional (\( \beta_1 \)) biases by OLS after substituting the BLUP for \( \hat{x}_i \) for the true unmeasured trait, \( x_i \). We may then use the Wald test and 95% confidence intervals (CIs) for \( \alpha_1 \) and \( \beta_1 \) to formally assess these biases. As before, we can obtain a smooth estimate of the variance by using OLS to fit the model \( |\hat{e}_{1ij}| = \theta_1^{(0)} + \theta_1^{(1)} \hat{x}_i + \omega_{ij} \), where \( |\hat{e}_{1ij}| \) is the absolute value of the residuals \( \hat{e}_{1ij} \) from the linear regression model \( y_{1ij} = \alpha^*_1 + \beta^*_1 \hat{x}_i + \epsilon^*_{1ij} \). Then, based on the estimates \( \hat{\alpha}_1^* \) and \( \hat{\beta}_1^* \), the bias of the new method is estimated as

\[
\text{bias}_i = \hat{\alpha}_1^* + \hat{x}_i (\hat{\beta}_1^* - 1)
\]

To visually assess the degree of bias, we obtain the “bias plot”, after which the package is named, by graphing a scatterplot of \( y_{1ij} \) and \( y_{2ij} \) versus the BLUP for \( \hat{x}_i \).
along with the two regression lines, while adding a second scale on the right showing
the relationship between the estimated amount of bias and $\hat{\alpha}_1$.

Taffé shows, by simulation, that this methodology performs well and that the es-
timates of the differential $\alpha_1$ and proportional $\beta_1$ biases are reasonably unbiased and
consistent already for sample sizes of 100 persons, with 3 to 5 repeated measurements
per individual from the reference method and only 1 measurement from the new method.
However, to appropriately estimate the (heterogeneous) measurement error variances,
one should have 10 to 15 repeated measurements per individual from the reference
method and 1 or several measurements from the new method.

2.3 Recalibration of the new method

To remove the differential and proportional biases of the new method, we recalibrate it
by computing $y_{1ij}^* = (y_{1ij} - \hat{\alpha}_1)/\hat{\beta}_1$. The “comparison plot” allows us to visualize the
recalibration procedure.

Now that $y_{2ij}$ and $y_{1ij}^*$ are on the same scale, we can compare the variances of the
measurement errors to determine which method is more precise. Because we would like
to compare $y_{2ij}$ with $y_{1ij}^*$ (and not with $y_{1ij}$), we should recalculate a smooth estimate
of the measurement errors variance of $y_{1ij}^*$ by proceeding like before.

We can then compare the variances by making a scatterplot of the estimated stan-
dard deviations $\hat{\sigma}_{\epsilon_1}(\hat{x}_i; \theta_1)$ and $\hat{\sigma}_{\epsilon_2}(\hat{x}_i; \theta_2)$ versus $\hat{x}_i$, which we call the “precision plot”.
It is possible that after recalibration, the new method will turn out to be more precise
(locally or globally) than the reference standard.

3 The biasplot command

biasplot fits the measurement error model and provides estimates of the differential and
proportional biases. It also allows the computation of the extended version of Bland and
Altman’s LoA when the variances of measurement errors are possibly heteroskedastic.
We obtain the (extended) LoA, bias, precision, and comparison plots by specifying one
of the options: loa, bias, precision, or comp.

3.1 Syntax

The syntax for using biasplot is

```
biasplot [if] [in], idvar(varname) ynew(varname) yref(varname) [loa
bias precision comp pdfs]
```
Note that you must choose at least one of the options loa, bias, precision, or comp for the program to run and save the corresponding graphs to the current directory. Also, the recalibrated values (y1_corr) of the new measurement method will be added to the dataset after computing the bias plot.

### 3.2 Options

- **idvar(varname)** specifies the variable identifying the individual. **idvar()** is required.
- **ynew(varname)** specifies the new measurement method’s variable name. **ynew()** is required.
- **yref(varname)** specifies the reference standard method’s variable name. **yref()** is required.
- **loa** computes and graphs the (extended) LoA.
- **bias** graphs the bias plot.
- **precision** graphs the precision plot.
- **comp** graphs the comparison plot.
- **pdfs** saves the graphs in .pdf format (instead of Stata’s .gph format).

### 4 Examples

To illustrate the use of biasplot, we will consider three simulated datasets:

#### Simulated dataset 1

\[
y_{1i} = -4 + 1.2x_i + \epsilon_{1i} \quad \epsilon_{1i} | x_i \sim N\left(0, (1 + 0.1x_i)^2\right) \\
y_{2ij} = x_i + \epsilon_{2ij} \quad \epsilon_{2ij} | x_i \sim N\left(0, (2 + 0.2x_i)^2\right) \\
x_i \sim \text{Uniform}[10-40]
\]

where \(i = 1, \ldots, 100\) and the number of repeated measurements of individual \(i\) from the reference standard was \(n_{1i} = 1\) and \(n_{2i} \sim \text{Uniform}[10-15]\).

#### Simulated dataset 2

\[
y_{1i} = -4 + 1.2x_i + \epsilon_{1i} \quad \epsilon_{1i} | x_i \sim N\left(0, (1 + 0.1x_i)^2\right) \\
y_{2ij} = x_i + \epsilon_{2ij} \quad \epsilon_{2ij} | x_i \sim N\left(0, (2 + 0.2x_i)^2\right) \\
x_i \sim \text{Uniform}[10-40]
\]

where \(i = 1, \ldots, 100\) and the number of repeated measurements of individual \(i\) from the reference standard was \(n_{1i} \sim \text{Uniform}[1-5]\) and \(n_{2i} \sim \text{Uniform}[10-15]\).
Simulated dataset 3

\[ y_{1i} = 3 + 0.9x_i + \epsilon_{1i} \quad \epsilon_{1i} | x_i \sim N \left\{ 0, (2 + 0.06x_i)^2 \right\} \]

\[ y_{2ij} = x_i + \epsilon_{2ij} \quad \epsilon_{2ij} | x_i \sim N \left\{ 0, (1 + 0.01x_i)^2 \right\} \]

\[ x_i \sim \text{Uniform}[10-40] \]

where \( i = 1, \ldots, 100 \) and the number of repeated measurements of individual \( i \) from the reference standard was \( n_{1i} \sim \text{Uniform}[1-5] \) and \( n_{2i} \sim \text{Uniform}[10-15] \).

### 4.1 Dataset 1

In dataset 1, there are between 10 and 15 repeated measurements by the reference standard and only 1 by the new measurement method for each individual. The differential and proportional biases are \(-4\) and \(1.2\), respectively. The standard deviation of the measurement errors is heteroskedastic for both measurement methods and increases with the level of the underlying true latent trait. However, the dispersion of the reference standard is twice that of the new measurement method.
Figure 1. (a) Bland and Altman’s LoA plot; (b) bias plot showing the amount of bias of the new measurement method; (c) scatterplot illustrating that recalibration of the new measurement method (that is, $y_{1\text{corr}}$) was effective; and (d) precision plot showing the precision (that is, standard deviation of the measurement error) of each measurement method.

The LoA plot indicates a slight positive bias of the new measurement method for low values of the estimated latent trait level (that is, BLUP of $x$) and a negative bias for high values. On the contrary, the bias plot illustrates that the bias is negative for low values and positive for high values. The estimated differential bias is $-3.21\ 95\%\ CI = [-5.36, -1.06]$, and the estimated proportional bias is $1.19\ 95\%\ CI = [1.10, 1.28]$. These values are close, and the CIs cover the true values. The precision plot shows that after recalibration, the new measurement method is about twice as precise as the reference standard and that both measurement methods are more precise for lower values than for higher values of the true latent trait. One can see on the comparison plot that recalibration of the new measurement method was effective in removing bias.
4.2 Dataset 2

Dataset 2 is similar to dataset 1, except there are between one to five repeated measurements (instead of just one) by the new measurement method for each individual:

```
. use sample2.dta, clear
. biasplot, idvar(id) ynew(y1) yref(y2) loa
(0 observations deleted)
Bias and Precision Plots
Variables - Please check -
id Variable: id
New Method Y Variable: y1
Reference Method Y Variable: y2
Running ...
Generating Bland and Altman LoA Plot
Bland and Altman LoA Plot saved to current working directory
Please wait ...
diff_bias=-3.527, 95%CI=[-5.4047941;-1.6485342]
prop_bias=1.181, 95%CI=[1.1038746;1.2575773]
Bias Plot Omitted
Comparison Plot Omitted
Precision Plot Omitted
End of Commands
. biasplot, idvar(id) ynew(y1) yref(y2) bias
(output omitted)
. biasplot, idvar(id) ynew(y1) yref(y2) comp
(output omitted)
. biasplot, idvar(id) ynew(y1) yref(y2) precision
(output omitted)
```
Figure 2. (a) Bland and Altman’s LoA plot; (b) bias plot showing the amount of bias of the new measurement method; (c) scatterplot illustrating that the recalibration of the new measurement method (that is, $y_{1 \text{corr}}$) was effective; and (d) precision plot showing the precision (that is, standard deviation of the measurement error) of each measurement method.

Consistently with the results for dataset 1, the LoA plot indicates a slight positive bias of the new measurement method for low values of the estimated latent trait level (that is, BLUP of $x$) and a negative bias for high values. In contrast, the bias plot (correctly) illustrates that the bias is negative for low values and positive for high values. The estimated differential bias is $-3.53$ 95% CI = $[-5.13, -1.92]$, and the estimated proportional bias is $1.18$ 95% CI = $[1.12, 1.24]$. Note that these CIs are, as expected, more narrow than when we have only one measurement by the new method (that is, dataset 1). The conclusions for the precision and comparison plots are the same as for dataset 1.

## 4.3 Dataset 3

In dataset 3, there are between 10 to 15 repeated measurements by the reference standard and 1 to 5 repeated measurements by the new measurement method for each individual. The differential bias amounts to 3, and the proportional bias amounts
Plots to assess bias and precision in method comparison studies

to 0.9. The standard deviation of the measurement errors is heteroskedastic for both measurement methods and increases with the level of the underlying true latent trait. However, the dispersion of the reference standard is much lower than that of the new measurement method.

```
. use sample3.dta, clear
. biasplot, idvar(id) ynew(y1) yref(y2) loa
(0 observations deleted)
Bias and Precision Plots
Variables - Please check -
id Variable: id
New Method Y Variable: y1
Reference Method Y Variable: y2
Running ...
Generating Bland and Altman LoA Plot
Bland and Altman LoA Plot saved to current working directory
Please wait ...
diff_bias=2.714, 95%CI=[1.3859422;4.0417949]
prop_bias=.902, 95%CI=[.84553455;.95922297]
Bias Plot Omitted
Comparison Plot Omitted
Precision Plot Omitted
End of Commands
. biasplot, idvar(id) ynew(y1) yref(y2) bias
(output omitted)
. biasplot, idvar(id) ynew(y1) yref(y2) comp
(output omitted)
. biasplot, idvar(id) ynew(y1) yref(y2) precision
(output omitted)
```
Figure 3. (a) Bland and Altman’s LoA plot; (b) bias plot showing the amount of bias of the new measurement method; (c) scatterplot illustrating that the recalibration of the new measurement method (that is, $y_{1,\text{corr}}$) was effective; and (d) precision plot showing the precision (that is, standard deviation of the measurement error) of each measurement method.

The LoA plot does not indicate any bias from the new measurement method, whereas the bias plot illustrates that the bias is positive for low values and negative for high values. The estimated differential bias is $2.71$ 95% CI $=[1.37, 4.06]$, and the estimated proportional bias is $0.9$ 95% CI $=[0.85, 0.96]$. The precision plot shows that after recalibration, the new measurement method is clearly less precise than the reference standard. Note that the dispersion of measurement errors of the reference standard is almost constant throughout the whole range of the latent trait, whereas that of the new method is, in comparison, sharply increasing. Again, the recalibration of the new measurement method was very effective in removing bias, as illustrated by the comparison plot.

5 Discussion

Using simulated data where the relationship between the true latent trait and the two measurement methods is known, we have illustrated that biasplot was effective in
Plots to assess bias and precision in method comparison studies

removing existing bias of the new measurement method and in assessing the precision of the two measurement methods after recalibration. We have also illustrated that there are settings where Bland and Altman’s LoA methodology is misleading, whereas biasplot allows one to properly identify, quantify, and correct for any biases.

biasplot is widely applicable when one has repeated measurements from the reference standard or possibly as few as one measurement per individual from the new method to be evaluated. Also, biasplot is useful when measurement errors are either homoskedastic or heteroskedastic. When one’s focus is mainly on identifying and correcting for the bias, as few as three to five repeated measurements from the reference standard and only one from the new measurement method may be large enough to provide good point estimates of the proportional and differential biases and CIs with appropriate coverage rates. However, when one compares the precision of the two measurement methods, it is better to have at least 10 to 15 repeated measurements by the reference standard (and possibly only 1 from the new measurement method) to appropriately estimate the (heteroskedastic) standard deviations of the 2 measurement methods. Actually, it is important to have repeated measurements from the reference standard because our methodology relies essentially on the BLUP of $x_i$, whereas repeated measurements from the new method will increase precision of the estimated heteroskedastic relationship.

In summary, we have implemented in biasplot a new estimation procedure to assess bias and precision of a quantitative measurement method relative to the reference standard, which performs very well in many settings, particularly when several measurements from the reference standard and possibly only one from the new measurement method are available. This method enables measurement errors to be either homoskedastic or heteroskedastic and provides bias, precision, and comparison plots to allow the investigator to visually and clinically appraise the performance of the new method. These plots do not have the shortcomings of Bland and Altman’s LoA and are still in the spirit of the original article.

6 References


**About the authors**

Patrick Taffé is a biostatistician in the Institute for Social and Preventive Medicine at the University of Lausanne and a lecturer in the Faculty of Biology and Medicine at the University of Lausanne, Switzerland.

Mingkai Peng is a statistical associate in the Department of Community Health Sciences at the University of Calgary and a member of the O’Brien Institute for Public Health.

Vicki Stagg is the Senior Statistical Analyst with Calgary Statistical Support (http://calgarystatisticalsupport.com/).

Tyler Williamson is an assistant professor of biostatistics in the Department of Community Health Sciences at the University of Calgary and a member of the Alberta Children’s Hospital Research Institute and the O’Brien Institute for Public Health.
Estimation of unit values in household expenditure surveys without quantity information

Martina Menon
University of Verona
Department of Economics
Verona, Italy
martina.menon@univr.it

Federico Perali
University of Verona
Department of Economics and Centre for Household, Income, Labour and Demographic Economics
Verona, Italy
federico.perali@univr.it

Nicola Tommasi
University of Verona
Interdepartmental Centre of Economic Documentation
Verona, Italy
nicola.tommasi@univr.it

Abstract. In this article, we present the pseudounit command, which estimates pseudounit values in cross-sections of household expenditure surveys without quantity information. Household surveys traditionally record only expenditure information. The lack of information about quantities purchased precludes the possibility of deriving household-specific unit values. We use a theoretical result developed by Lewbel (1989, Review of Economic Studies 56: 311–316) to construct pseudounit values by first reproducing cross-sectional price variation, then adding this variability to the aggregate price indexes published by national statistical institutes. We illustrate the method with an example that uses a time series of cross-sections of Italian household budgets.

Keywords: st0472, pseudounit, unit values, cross-section prices, demand analysis

1 Introduction

This article presents the theory used to implement pseudounit, a command that estimates unit values in cross-sections of household expenditure surveys without quantity information, and describes how the command should be used. Empirical works on demand analysis generally rely on the assumption of price invariance across households, supported by the hypothesis that, in cross-sectional data, there are neither time nor
spatial variations in prices. According to this assumption, each family pays the same prices for homogeneous goods. Microdata with this characteristic allow researchers to estimate only Engel curves without accounting for price effects, which are crucial for both behavioral and welfare applications. Slesnick (1998, 2150) states that “the absence of price information in the surveys creates special problems for the measurement of social welfare, inequality and poverty. . . . Most empirical work links micro data with national price series on different types of goods[,] so cross sectional variation is ignored. Access to more disaggregate information on prices will enhance our ability to measure social welfare, although it remains to be seen whether fundamental conclusions concerning distributional issues will be affected”. In empirical works, such limitation is usually bypassed by analyzing time series of cross-sections where price information comes from aggregate time-series data. Plausible estimates of price effects require a sufficiently long series of cross-sections and, if possible, aggregate price indexes that vary by month and location, usually by region or province.

Household budget surveys of both developed and developing countries can be classified into two broad categories in increasing order of frequency of occurrence: 1) surveys of expenditure and quantities purchased and 2) surveys of expenditure data only. In the first case, where quantities and expenditure are both observed, cross-sectional prices are obtained as implicit prices, dividing expenditure by quantities, and are more properly referred to as “unit values”. When dealing with these surveys, one should remember that a proper use of unit values in econometric analyses must account for problems arising from the fact that unit values provide useful information about prices, but differ from market prices in many respects. The ratio between expenditure and quantities bought embed information about the choice of quality (Deaton 1987, 1988, 1989, 1990, 1998; Perali 2003). The level of the unit value of a composite good depends on the relative share of high-quality items and the composition of the aggregate good. Unit values can also be highly variable for supposedly homogeneous goods because the market offers many different grades and types.

On the other hand, when one deals with surveys that report only expenditure information, aggregate national price indexes are usually merged with household expenditure to obtain estimates of price elasticities. Unfortunately, this approach requires a long time series of cross-sectional data to estimate a demand system with sufficient price variation and relies on very restrictive assumptions (Frisch 1959), which often turn out to be rejected in empirical applications. Aggregate price indexes are generally highly correlated, may suffer from endogeneity problems (Lecocq and Robin 2015), and the estimated elasticities are often not coherent with the theory (Atella, Menon, and Perali 2004; Coondoo, Majumder, and Ray 2004; Dagsvik and Brubakk 1998; Lahatte et al. 1998). Thus surveys gathering exclusively expenditure data, such as the Italian household budget survey conducted by the National Statistical Institute (Italian Statistical Institute [ISTAT]) used in our example and the majority of existing household budget surveys, have limited applicability in modern demand and welfare analysis. Thus one should devise an appropriate procedure to compute pseudounit values using the information traditionally available in expenditure surveys, such as budget shares and demographic characteristics, which help reproduce the distribution of the unit-value
variability as closely as possible. The theoretical background for this undertaking is provided in a study by Lewbel (1989).

The remainder of the article is organized as follows: Section 2 presents the theory and method used to derive consumer price indexes and pseudounit values (PUVs) when the main objective is to implement demand analysis of household budget data without quantity information. Section 3 provides the syntax and options of the pseudounit command. Section 4 illustrates the application. Section 5 concludes.

2 The estimation of unit values in cross-section analysis of household budget surveys

We introduce a method that recovers unit values when only expenditure information is available using knowledge about aggregate price indexes available from national statistics.

First, as illustrated in section 2.1, we need to collect the consumer price indexes available from official statistics and associate them with each household in the survey. Then, to improve the precision of the estimated price elasticities as shown in Atella, Menon, and Perali (2004), we reproduce as best as we can the price variation of actual unit values, which could be obtained as the ratio between expenditure and quantities if quantity information were available in the survey. The estimation of PUVs is described in section 2.2 and 2.3.

2.1 Consumer price indexes

Eurostat adopts the classification of individual consumption by purpose (COICOP), which is a nomenclature developed by the United Nations Statistics Division to classify and analyze individual consumption expenditures incurred by households, nonprofit institutions serving households, and general government according to their purpose. National statistical institutes traditionally publish consumer price indexes per each COICOP category monthly, which are collected at the provincial level.

Let $P_{rim}^{ij}$ be the consumer price index for the $j$th of the $i$th COICOP group with $i = 1, \ldots, n$ collected monthly by national statistical institutes, $m = 1, \ldots, M$, per each territorial level $r = 1, \ldots, R$, such as a province or a region. These price indexes are the same for all households living in the same region and interviewed in the same month. If detailed price information is disaggregated either by territorial level or on a monthly basis, then we have only $P_i$, which is the same for all households. With this highly limited price information, demand analysis cannot be implemented, because the data matrix is not invertible and PUVs must be estimated.

1. The COICOP top-level aggregation encompasses 12 categories: food and nonalcoholic beverages; alcoholic beverages, tobacco and narcotics; clothing and footwear; housing, water, electricity, gas and other fuels; furniture; health; transportation; communication; recreation and culture; education; restaurants and hotels; and miscellaneous goods and services.
The next task is to match the monthly price index specific to each territorial unit, $P_{ij}^{rm}$, with all households living in province or region $r$ and interviewed at month $m$. Then, each $P_{ij}^{rm}$ is aggregated into a price index $P_{i}^{rm}$ for $i = 1, \ldots, n$ groups corresponding to the goods selected for the empirical demand analysis. The aggregation uses Laspeyres indexes

$$P_{i}^{rm} = \sum_{j=1}^{n_{i}} (P_{ij}^{w_{ij}})$$

where $j = 1, \ldots, n_{i}$, with $n_{i}$ being the number of goods within group $i$ and $w_{ij}$ being the weights provided by national statistical institutes for each item $j$ of group $i$. As an example, we may suppose that a budget is divided into $i = 1, 2$ groups such as food and nonfood and that the food subgroup is composed by $j = 1, 2, 3$ items such as cereals, meat, and other food.

So far we have described how to prepare the data matrix containing information about the available price indexes. Next, we present the background theory used to pursue the objective to reconstruct the cross-sectional variability of unit values.

### 2.2 Demographically varying PUVs: Theory

Lewbel (1989) proposes a method to estimate the cross-sectional variability of actual unit values by exploiting the demographic information included in generalized “within-group” equivalence scales or, more generally, demographic functions. For a group $i$ of goods, these are defined as the ratio of a subutility function of a reference household to the corresponding subutility function of a given household estimated without price variation in place of “between-group” price variation. The method relies on the assumption that the original utility function is homothetically separable and “within-group” subutility functions are Cobb–Douglas.

Consider a separable utility function $U\{u_{1}(q_{1}, d), \ldots, u_{n}(q_{n}, d)\}$ defined over the consumption of good $q_{i}$ and a set of demographic characteristics $d$, where $U(u_{1}, \ldots, u_{n})$ is the “between-group” utility function and $u_{i}(q_{i}, d)$ is the “within-group” subutility function, where $i = 1, \ldots, n$ denotes the aggregate commodity groups. Demographic characteristics, $d$, affect $U$ indirectly through the effects on the within-group subutility function. Define the group equivalence scale, $M_{i}(q_{i}, d)$, as

$$M_{i}(q_{i}, d) = \frac{u_{i}(q_{i}, d^{h})}{u_{i}(q_{i}, d)}$$

where $d^{h}$ describes the demographic profile of a reference household. Define a quantity index for group $i$ as $Q_{i} = u_{i}(q_{i}, d^{h})$, and rewrite the between-group utility function as

$$U(u_{1}, \ldots, u_{n}) = U\left(\frac{Q_{1}}{M_{1}}, \ldots, \frac{Q_{n}}{M_{n}}\right)$$

2. When not available, the subgroup budget shares can be used as the weight of aggregation.
3. If the interest is to build a time-series collection of cross-sections of household budget surveys, then in the base year, the indexes for all goods are equal to 100.
4. This section closely reproduces the procedure developed by Lewbel (1989).
Estimation of unit values

which is formally analogous to Barten’s (1964) technique to introduce demographic factors in the utility function. Define further the price index for group $i$ as $P_i = Y_i^h / Q_i$, where $Y_i^h$ is the expenditure on group $i$ by the reference household. To guarantee group demands are closed under unit scaling, we must apply a scaling factor $k_i$ to the quantity index, $Q_i$, that makes $P_i = 1$ for all $i$ when $p_{ij} = 1$ for all $i$ and $j$. This would occur, for example, in a base year when $p_{ij}$ are in index form. Thus $P_i = Y_i^h / k_i Q_i$. Barten’s utility structure implies the following share demands for each household with total expenditure $Y$:

$$W_i = H_i(P_1 M_1, \ldots, P_n M_n, Y)$$

This takes the form of $W_i^h = H_i^h(P_1, \ldots, P_n, Y^h)$ for the reference household with scales $M_i = 1$ for all $i$. The further assumption of homothetic separability admits two-stage budgeting (Deaton and Muellbauer 1980) and implies the existence of functions $V_i$, such that $P_i = V_i(p_i, d_i)$ is the price index of group $i$ for the reference household with demographics $d_i$, $p_i = (p_{i1}, \ldots, p_{in_i})$ is the vector of prices, where $n_i$ is the number of goods that compose group $i$. By analogy with the definition of group equivalence scales in utility space, it follows that

$$M_i = \frac{V_i(p_i, d)}{V_i(p_i, d^h)}$$

where $V_i(p_i, d) = M_i P_i$. Therefore, when demands are homothetically separable, each group scale depends only on relative prices within group $i$ and on $d$ as expected given that homothetic separability implies strong separability. Maximization of $u_i(q_i, d)$, subject to the expenditure $p_i q_i = Y_i$ of group $i$, gives the budget share for an individual good $w_{ij} = h_{ij}(p_i, d, Y_i)$. For homothetically separable demands, the budget shares do not depend on expenditure $w_{ij} = h_{ij}(p_i, d)$ and integrate back in a simple fashion to $V_i = M_i P_i$. This information can be used at the between-group level in place of price data to estimate $W_i = H_i(V_1, \ldots, V_n, Y)$. Under the assumption that the subgroup utility functions are Cobb–Douglas with parameters specified as “shifting” functions of demographic variables alone, we can specify the following relationship:

$$F_i(q_i, d) = k_i \prod_{j=1}^{n_i} q_{ij}^{m_{ij}(d)}$$

The shares $w_{ij} = (\partial \log V_i) / (\partial \log p_{ij})$ then correspond to the demographic functions

$$w_{ij} = h_{ij}(p_i, d) = m_{ij}(d)$$

with

$$\sum_{j=1}^{n_i} w_{ij}(d) = \sum_{j=1}^{n_i} m_{ij}(d) = 1$$

The implied price index is

$$V_i(p_i, d) = M_i P_i = \frac{1}{k_i} \prod_{j=1}^{n_i} \left( \frac{p_{ij}}{m_{ij}} \right)^{m_{ij}}$$
with
\[ k_i = \prod_{j=1}^{n_i} m_{ij} \left( d^h \right)^{-m_{ij}(d^h)} \]
where \( k_i \) is a scaling function depending only on the choice of the reference demographic levels.

Note\(^5\) that, assuming separable and homothetic preferences within groups and letting \( q_{ij} \) denote scaled units so that corresponding prices \( p_{ij} \) are unity in a base year,\(^6\) the group cost function for the reference household is
\[
\begin{aligned}
c_i \left( u_i, \tilde{p}_i, d^h \right) &= k_i u_i \left( \tilde{q}_i, d^h \right) \frac{b \left( \tilde{p}_i, d^h \right)}{k_i} = k_i Q_i P_i \\
\end{aligned}
\]
where \( b \left( \tilde{p}_i, d^h \right) \) is concave and linearly homogeneous in prices, and time subscripts are omitted for simplicity. To ensure the group price index is unity in the base year, we denote the scaling factor as \( k_i = b \left( \tilde{p}_i, d^h \right) \). Thus \( P_i = \{ b \left( \tilde{p}_i, d^h \right) \} / k_i = V_i \left( \tilde{p}_i, d^h \right) \), and the price per equivalent capita is
\[
\begin{aligned}
M_i P_i = \frac{b \left( \tilde{p}_i, d^h \right)}{k_i} \frac{b \left( \tilde{p}_i, d \right)}{k_i} = \frac{b \left( \tilde{p}_i, d \right)}{k_i} = V_i \left( \tilde{p}_i, d \right)
\end{aligned}
\]
When the subutility functions are Cobb–Douglas, \( b \left( \tilde{p}_i, d \right) = \prod_{j=1}^{n_i} \left( \tilde{p}_{ij}/m_{ij} \right)^{m_{ij}} \), and it is easy to see that the scaling factor is \( k_i = \prod_{j=1}^{n_i} \left( 1/m_{ij}^h \right)^{m_{ij}}, \) where the parameters \( m_{ij} = m_{ij} \left( d^h \right) \) and \( m_{ij}^h = m_{ij} \left( d^h \right) \).

Note that the Cobb–Douglas assumption places restrictions only at the within-group level while leaving the between-group demand equations free to be arbitrarily flexible. An approximation to \( M_i P_i = M_i \) can be obtained by using the observed within-group budget shares.\(^7\) These results support a simple procedure to estimate price variation in survey data without quantity information.

### 2.3 Demographically varying PUVs: Practice

Given this theoretical setup, we now describe how PUVs can be obtained in practice. The description corresponds to the implementation of the `pseudounit` command.

**Definition 1.** PUVs—\( \text{PUV}(\tilde{P}_D^i) \)
\[
\tilde{P}_D^i = M_i P_i = M_i = \frac{1}{k_i} \prod_{j=1}^{n_i} w_{ij}^{-w_{ij}}
\]
where \( k_i \) is the average of the subgroup expenditure for the \( i \)th group budget share.

---

5. We would like to thank an anonymous reviewer for suggesting we report how the expressions for \( M_i \), \( P_i \), and \( k_i \) are derived.

6. If \( p_{ijt} \) is the price in year \( t \) and \( p_{ij0} \) the price in base year 0, then \( \tilde{p}_{ijt} = p_{ijt}/p_{ij0} \) and \( \tilde{q}_{ijt} = q_{ijt}/p_{ij0} \).

7. In cross-sectional data where prices are not reported or unit values cannot be derived, it is common to assume that price indexes do not vary and are therefore equal to one.
The index \( \hat{P}_D \) summarizes the cross-section variabilities of prices that can be added to spatially varying price indexes to resemble unit values expressed in index form as follows. In general, this technique allows the recovery of the household-specific price variability that can be found in unit values. The PUV is an index that can be compared with actual unit values after normalization, choosing the value of a specific household as a numeraire.

**Definition 2.** PUVs in index form—PUV(\( \hat{P}_{DI} \))

\[
\hat{P}_{DI}^i = \hat{P}_D^i P_{rm}^i
\]

where \( P_{rm}^i \) are the group-specific price indexes derived in (1).

For PUVs in index form to look like actual unit values, they have to be transformed into levels. The transformation in nominal terms is fundamental to properly capture complementary and substitution effects as shown in Atella, Menon, and Perali (2004). Cross effects would otherwise be the expression of the differential speed of change of the good-specific price indexes through time only. Note that \( \hat{P}_D = M_t P_t \) holds for the base year only where \( \tilde{p}_{ij} = 1 \) for all regions and that time subscripts are omitted for simplicity. In subsequent time-periods,

\[
M_t P_t = \frac{1}{k_i} \left( \prod_{j=1}^{n_i} w_{ij} \right) \left( \prod_{j=1}^{n_i} \tilde{p}_{ij} \right) = \hat{P}_D \left( \prod_{j=1}^{n_i} \tilde{p}_{ij} \right)
\]

which is represented by the PUV in index form, \( \hat{P}_{DI}^i = \hat{P}_D^i P_{rm}^i \). Further, \( \hat{P}_{DI}^i \) will be an approximation, unless \( P_{rm}^i \) is equivalent to \( \prod_{j=1}^{n_i} \tilde{p}_{ij} \), which resembles a Stone price index.

**Definition 3.** PUVs in levels—PUV(\( \hat{P}_{DIL} \))

\[
\hat{P}_{DIL}^i = \hat{P}_D^i \bar{y}_i
\]

where \( \bar{y}_i \) is the average expenditure of group \( i \) in the base year.

Early experiments with PUVs with Italian household budget data (Perali 1999, 2000; Atella, Menon, and Perali 2004; Menon and Perali 2010) and Hoderlein and Mihaeleva (2008) and Berges, Pace Guerrero, and Echeverría (2012) for other datasets have provided comforting indications about the possibility of estimating regular preferences. Atella, Menon, and Perali (2004) describe the effects on the matrix of cross-price elasticities associated with several price definitions and find that the matrix of compensated elasticities is negative definite only if PUVs are used. Nominal PUVs, which more closely reproduce actual unit values, give a set of own- and cross-price effects that is more economically plausible. The derived demand systems are regular and suitable for sound welfare and tax analysis. The authors conclude that the adoption of PUVs does no harm because Lewbel’s method simply consists of adding cross-sectional price variability to
aggregate price data. Therefore, Lewbel’s method for constructing demographically varying prices is potentially of great practical utility.

Because goods may differ in quality from one household to another and because their associated unit values may both reflect these differences in quality, measurement errors, and endogenous expenditure information, the estimated unit values are likely to be correlated with the equation errors, and the resulting estimators will be both biased and inconsistent. The demand estimation technique should therefore account for price endogeneity by using instrumental-variable methods.

We now proceed with the description of the \texttt{pseudounit} command.

3 \hspace{1em} The pseudounit command

3.1 Syntax

The syntax of \texttt{pseudounit} is as follows:

\texttt{pseudounit expenditures [if] [in], generate(varname) pindex(varname)}

\[ \text{impvars(varlist)} \text{ seed(#)} \text{ add(#) coll_rule(mean|median) expby(varname)} \]

\[ \text{pdi(varname)} \text{ year(varname)} \text{ saving(filename [, replace])} \]

where \textit{expenditures} is the list of expenditure variables of interest. The list must be specified as follows: the group expenditure first, then all the subexpenditures of the group; the \texttt{pseudounit} command verifies whether the sum of all subexpenditures sum to the group expenditure and whether each expenditure has positive or zero value.

3.2 Options

\texttt{generate(varname)} specifies the name of the new variable that will be created with the unit values associated with the group expenditure. \texttt{generate()} is required.

\texttt{pindex(varname)} specifies the variable with the price index associated with the group expenditure. For the construction of this variable, see Atella, Menon, and Perali (2004). The index must have the same base year as \texttt{expby(varname)}. \texttt{pindex()} is required.

\texttt{impvars(varlist)} specifies the variables to be used for the imputation of the zero expenditure shares. The subexpenditures must be at least two. The imputation uses the \texttt{mi impute truncreg} command, where the dependent variable is the expenditure share and the independent variables are the variables specified in \texttt{impvars(varlist)}. Without the \texttt{impvars(varlist)} option, the PUV variable is not computed for observations with subexpenditures equal to zero. If the \texttt{year()} option is specified, \texttt{year(varname)} variables are also added to \texttt{impvars(varlist)}. If the imputation with \texttt{mi impute truncreg} fails, the command switches to \texttt{mi impute pmm} using a
number of \( k \) nearest neighbors equal to 5% of the positive observations of the within-
group shares. One can also use categorical variables with the appropriate syntax (see [u] 11.4.3 Factor variables). Because imputed shares must be positive, the program checks for negative imputed values and substitutes them with the value of one.

Because the procedure uses a product, this guarantees the subgroup expenditure
does not contribute to the group price for that specific household.

**seed(#)** sets the random-number seed. This option is used to reproduce results. The default is **seed(159753)**.

**add(#)** specifies the number of imputations to add to the **mi** data. The total number of imputations must be comprised between 5 and 1,000. The default is **add(20)**.

**coll_rule(mean|median)** specifies the rule used to collapse **mi** data. **mi imputation**
**truncreg** adds \( n \) replicas to the data with \( n \) imputations of the missing data, where \( n \) corresponds to the value reported in the **add(#)** option. Each dataset is identified with values of the variable **_mi_id**. These \( n \) imputations are then reduced to a single imputation by a data collapse that can be implemented with either the mean or the median. **pseudounit** executes the following command: **collapse (mean|median) share.var, by(_mi_id)**. The default statistic is the mean.

**expby(varname)** specifies the average group expenditure for the base year.\(^8\) Without the **expby(varname)** option, the variable in the **generate(varname)** option is equal to the PUV in index form (PUV(DI)).

**pdi(varname)** specifies the variable generated with PUV in index form (PUV(DI)).

**year(varname)** specifies the name of the year variable when estimating unit values for several years. This computes the average of the subgroup expenditure for the \( i \)th group budget share for each year. The **year()** option can be used when a time series of cross-sections is available, so it is possible to compute the mean expenditure shares \( w_{ij} \) by each year.

**saving(filename [, replace])** specifies the name of the disk file to be created or replaced. This option saves a kernel density graph of PUV in levels—PUV(\( \hat{P}_iDIL \)). If the **year(varname)** option is specified, no graph is produced. If **filename** is specified without an extension, .gph will be assumed.

### 4 The pseudounit command: Examples

To become familiar with the command, the user may be interested in the following examples using **pseudounit.cmd.dta**, which is provided with the package.

---

\(^8\) When using one cross-section only, we can choose a given month (for instance, January) as the base year for both the price indexes and the group expenditures.
4.1 Data

For our example, expenditure data come from a series of repeated cross-sectional national household budget surveys conducted yearly by the ISTAT. Within each cross-section, households are interviewed monthly at different times during the year. The ISTAT budget survey is representative at the regional level.

The samples of household budgets for 2007 and 2008 used in this example comprise more than 23,000 households per year. To reduce the estimation burden of the present application, we have drawn a random sample of 4,935 households for 2007 and 4,916 for 2008. Household expenditures in the provided dataset have been aggregated into six groups and then transformed in budget shares: food, clothing, housing, transport and communications, leisure and education, and health and other nonfood.

ISTAT collects information about consumer price indexes based on the consumption habits of the whole population available monthly for each of the 106 Italian provinces with the COICOP level of disaggregation. We have chosen January 1997 as the base year. Price indexes have been matched to the two samples, accounting for the period of the year when the household was interviewed. This means that households interviewed in March have been matched with prices collected in the same month. After determining the expenditure groups, we constructed the corresponding consumer price indexes starting from the COICOP categories available for territorial disaggregation and months that have been matched to all households living in the same region and interviewed in the same month.

Table 1 reports the descriptive statistics of the price index, $P_{rm}$, of the pseudounit procedure for the six groups of goods and services. If users already have price information from external sources organized as in table 1, they can call the pseudounit procedure without following the bottom-up approach outlined above.

Table 1. Descriptive statistics of $P_{rm}$ by year

<table>
<thead>
<tr>
<th>year</th>
<th>idx_aggr1</th>
<th>idx_aggr2</th>
<th>idx_aggr3</th>
<th>idx_aggr4</th>
<th>idx_aggr5</th>
<th>idx_aggr6</th>
</tr>
</thead>
<tbody>
<tr>
<td>2007</td>
<td>124.776</td>
<td>119.710</td>
<td>126.238</td>
<td>122.144</td>
<td>124.891</td>
<td>118.798</td>
</tr>
<tr>
<td></td>
<td>(4.192)</td>
<td>(5.426)</td>
<td>(2.751)</td>
<td>(2.310)</td>
<td>(2.677)</td>
<td>(2.916)</td>
</tr>
<tr>
<td>2008</td>
<td>124.832</td>
<td>119.721</td>
<td>126.183</td>
<td>122.097</td>
<td>124.846</td>
<td>118.733</td>
</tr>
<tr>
<td></td>
<td>(4.199)</td>
<td>(5.467)</td>
<td>(2.747)</td>
<td>(2.301)</td>
<td>(2.687)</td>
<td>(2.916)</td>
</tr>
<tr>
<td>Total</td>
<td>124.804</td>
<td>119.715</td>
<td>126.211</td>
<td>122.120</td>
<td>124.869</td>
<td>118.766</td>
</tr>
<tr>
<td></td>
<td>(4.195)</td>
<td>(5.446)</td>
<td>(2.749)</td>
<td>(2.306)</td>
<td>(2.682)</td>
<td>(2.916)</td>
</tr>
</tbody>
</table>

Note: Standard deviations are in parentheses.

Table 2 reports the levels of the average indexes, $P_{rm}$, by macroregion, selecting two households (HH1 and HH2) interviewed in time 1 or 2 in each macroregion to illustrate how the levels of price indexes may vary within each region by the time of the interview of the household.
Estimation of unit values

Table 2. Average levels of $P_{i}^{rm}$ by macroregion and households HH1 or HH2 interviewed in period 1 or 2

<table>
<thead>
<tr>
<th>Macro</th>
<th>idx_aggr1</th>
<th>idx_aggr2</th>
<th>idx_aggr3</th>
<th>idx_aggr4</th>
<th>idx_aggr5</th>
<th>idx_aggr6</th>
</tr>
</thead>
<tbody>
<tr>
<td>NW (HH1)</td>
<td>119.9</td>
<td>115.9</td>
<td>123.66</td>
<td>119.4</td>
<td>124.3</td>
<td>117.4</td>
</tr>
<tr>
<td>NW (HH2)</td>
<td>124.6</td>
<td>116.4</td>
<td>122.8</td>
<td>120.5</td>
<td>123.3</td>
<td>116.9</td>
</tr>
<tr>
<td>NE (HH1)</td>
<td>124.8</td>
<td>115.3</td>
<td>129.4</td>
<td>123.2</td>
<td>123.4</td>
<td>119.2</td>
</tr>
<tr>
<td>NE (HH2)</td>
<td>121.8</td>
<td>116.4</td>
<td>131.2</td>
<td>123.5</td>
<td>124.7</td>
<td>120.6</td>
</tr>
<tr>
<td>Centre (HH1)</td>
<td>125.6</td>
<td>123.2</td>
<td>121.0</td>
<td>121.4</td>
<td>130.8</td>
<td>116.1</td>
</tr>
<tr>
<td>Centre (HH2)</td>
<td>122.1</td>
<td>117.8</td>
<td>128.0</td>
<td>123.7</td>
<td>124.7</td>
<td>116.4</td>
</tr>
<tr>
<td>South (HH1)</td>
<td>123.9</td>
<td>109.5</td>
<td>123.4</td>
<td>113.9</td>
<td>114.2</td>
<td>117.6</td>
</tr>
<tr>
<td>South (HH2)</td>
<td>133.6</td>
<td>130.9</td>
<td>121.6</td>
<td>120.5</td>
<td>127.2</td>
<td>115.8</td>
</tr>
<tr>
<td>Islands (HH1)</td>
<td>122.5</td>
<td>116.6</td>
<td>125.1</td>
<td>123.3</td>
<td>124.53</td>
<td>121.0</td>
</tr>
<tr>
<td>Islands (HH2)</td>
<td>123.5</td>
<td>109.9</td>
<td>126.9</td>
<td>120.3</td>
<td>120.6</td>
<td>119.4</td>
</tr>
</tbody>
</table>

The composition of the group expenditures in our dataset is as follows:

Group expenditure 1: Food (ag6sp.1)

- Bread, cereals, and pasta (ag6sp.1.1)
- Meat, fish, and milk derivates (ag6sp.1.2)
- Fruit and vegetables (ag6sp.1.3)
- Fats and oils, sugar, alcoholic and nonalcoholic drinks and beverages, and tobacco (ag6sp.1.4)

Group expenditure 2: Clothing (ag6sp.2)

- Nonassignable clothing (ag6sp.2.1)
- Clothing and footwear: men (ag6sp.2.2)
- Clothing and footwear: women (ag6sp.2.3)
- Clothing and footwear: children (ag6sp.2.4)

Group expenditure 3: Housing (ag6sp.3)

- Rents and condominium fees (ag6sp.3.1)
- Water, energy, and heating (ag6sp.3.2)
- Home repairs and large electrical appliances (ag6sp.3.3)
- Small electrical appliances and flatware (ag6sp.3.4)
Group expenditure 4: Transport and communications (ag6sp_4)

- Private transportation (fuels and repairs) (ag6sp_4.1)
- Public transportation (ag6sp_4.2)
- Telephone (ag6sp_4.3)
- Purchase of means of transportation and telephone (ag6sp_4.4)

Group expenditure 5: Leisure and education (ag6sp_5)

- Education expenditures (ag6sp_5.1)
- Leisure (ag6sp_5.2)
- Computer, music, and television (ag6sp_5.3)
- Other (ag6sp_5.4)

Group expenditure 6: Health and other nonfood (ag6sp_6)

- Medical examinations and medicines (ag6sp_6.1)
- Insurance, expenditures for medical assistance, and other (ag6sp_6.2)

The dataset comprises price indexes associated with each expenditure (idx_aggr1–idx_aggr6) and mean expenditures evaluated at the base year (1997) for each of the six selected expenditure categories conditioned by region, number of household members, and month (mu_ag6sp1–mu_ag6sp6). Other variables are residential location, whether urban or rural (location), number of household components (nc), macroarea (ripgeo), age of the household head (etacf), education of the household head (titstucf), and the logarithm of the household total annual expenditure (lnx).

Note that in the base year, average expenditures are computed by region and month to preserve the maximum territorial and time variation.

4.2 Examples

We now implement the pseudounit command to estimate unit values for food, clothing, housing, and transport and communications to illustrate how to use the options available in the command.9

9. Our results are obtained using Stata 14. Possible marginal differences may be due to previous versions of Stata adopting a different pseudorandom-number generator.
Estimation of unit values

The first expenditure is food (\textit{ag6sp}_1) (subgroup expenditures are bread, cereals, and pasta [\textit{ag6sp}_1.1]; meat, fish, milk, and other protein [\textit{ag6sp}_1.2]; fruits and vegetables [\textit{ag6sp}_1.3]; and fats and oils, sugar, beverages, and tobacco [\textit{ag6sp}_1.4]).

The variables used for the imputation of zero expenditures are residential location (\textit{location}), macroarea (\textit{ripgeo}), number of household components (\textit{nc}), age of the household head (\textit{etacf}), education of the household head (\textit{titstucf}), and the logarithm of the total annual household expenditure (\textit{lnx}). The multiple imputation of the 0 expenditure shares generates 30 datasets that are then summarized using the mean as the default. The regional price index is \textit{idx}_{aggr1}\_\text{reg}, and the mean expenditure computed at the base year for food is \textit{mu}_{ag6sp}_1.

The variable associated with the unit values of the food expenditure is \textit{lwbp}\_\text{aggr1}.

In this case, there are no imputations because there are no zero share expenditures.

The second expenditure is clothing (subgroup expenditures are nonassignable clothing, clothing and footwear for men, clothing and footwear for women, and clothing and footwear for children).
**** EXPENDITURE ag6sp_2_3 ****
355 observations to impute
MULTIPLE IMPUTATION OVERVIEW
Method: truncreg regression
Limit: lower = 0
upper = 1
Total Observations: 4971
Complete observations: 4616
Missing observations: 355
Imputed observations: 355
0 values for expenditure ag6sp_2_3 converted to 1

**** EXPENDITURE ag6sp_2_4 ****
3149 observations to impute
MULTIPLE IMPUTATION OVERVIEW
Method: truncreg regression
Limit: lower = 0
upper = 1
Total Observations: 4971
Complete observations: 1822
Missing observations: 3149
Imputed observations: 3149
0 values for expenditure ag6sp_2_4 converted to 1

DESCRIPTIVES STATISTICS

<table>
<thead>
<tr>
<th>Variable</th>
<th>Obs.</th>
<th>Mean</th>
<th>Median</th>
<th>Std. Dev.</th>
<th>Min</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>PUV(D)</td>
<td>4971</td>
<td>1.0491</td>
<td>1.0514</td>
<td>0.0675</td>
<td>1.18</td>
<td>0.47</td>
</tr>
<tr>
<td>PUV(DI)</td>
<td>4971</td>
<td>1.2561</td>
<td>1.2592</td>
<td>0.1019</td>
<td>1.55</td>
<td>0.62</td>
</tr>
<tr>
<td>lwbp_aggr2</td>
<td>4971</td>
<td>174.42</td>
<td>177.34</td>
<td>67.20</td>
<td>555.83</td>
<td>24.84</td>
</tr>
</tbody>
</table>

Note: lwbp_aggr2 is Pseudo Unit Values in Levels PUV(DIL)

In this case, there are no imputations for subgroup expenditure ag6sp_2_1, but there are imputations for subgroup expenditures ag6sp_2_2, ag6sp_2_3, and ag6sp_2_4.

The third expenditure is housing (subgroup expenditures are rent and condo expenses; water, energy, and heating; home repairs and large electrical appliances; small electrical appliances and flatware). The lwbp_aggr3 variable is created for each year (year).

```
.pseudounit ag6sp_3 ag6sp_3_?,
> generate(lwbp_aggr3 pindex(idx_aggr3)
> impvars(location i.ripgeo nc etacf i.titstucf lnx)
> expby(mu_ag6sp_3) year(year)

**** EXPENDITURE ag6sp_3_3 ****
1668 observations to impute
MULTIPLE IMPUTATION OVERVIEW
Method: truncreg regression
Limit: lower = 0
upper = 1
Total Observations: 9859
Complete observations: 8191
Missing observations: 1668
Imputed observations: 1668
0 values for expenditure ag6sp_3_3 converted to 1
```
Estimation of unit values

DESCRIPTIVES STATISTICS

<table>
<thead>
<tr>
<th>Variable</th>
<th>Obs.</th>
<th>Mean</th>
<th>Median</th>
<th>Std. Dev.</th>
<th>Min</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>2007</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PUV(D)</td>
<td>4971</td>
<td>.9480497</td>
<td>.931581</td>
<td>.1956049</td>
<td>.4917437</td>
<td>1.509897</td>
</tr>
<tr>
<td>PUV(DI)</td>
<td>4971</td>
<td>1.196825</td>
<td>1.177358</td>
<td>.2483218</td>
<td>.6092809</td>
<td>1.940222</td>
</tr>
<tr>
<td>lwbp_aggr3</td>
<td>4971</td>
<td>729.4016</td>
<td>698.4557</td>
<td>248.5652</td>
<td>160.168</td>
<td>1894.283</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2008</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PUV(D)</td>
<td>4888</td>
<td>.9482172</td>
<td>.9302261</td>
<td>.1977599</td>
<td>.4697024</td>
<td>1.527056</td>
</tr>
<tr>
<td>PUV(DI)</td>
<td>4888</td>
<td>1.196471</td>
<td>1.173393</td>
<td>.2506707</td>
<td>.5757318</td>
<td>1.938348</td>
</tr>
<tr>
<td>lwbp_aggr3</td>
<td>4888</td>
<td>723.3252</td>
<td>692.9372</td>
<td>246.5752</td>
<td>187.4321</td>
<td>1780.587</td>
</tr>
</tbody>
</table>

Note: lwbp_aggr3 is Pseudo Unit Values in Levels PUV(DIL)

The fourth expenditure is transport and communications (subgroup expenditures are private and public transportation, telephone, and purchase of transportation means).

. pseudounit ag6sp_4 ag6sp_4_? if year==2008,
  > generate(lwbp_aggr4) pindex(idx_aggr4)
  > impvars(location i.ripgeo nc etacf i.titstucf lnx)
  > expby(mu_ag6sp_4) seed(889922) coll_rule(median) saving(kd_sp4, replace)

**** EXPENDITURE ag6sp_4_1 ****
907 observations to impute
MULTIPLE IMPUTATION OVERVIEW
Method: truncreg regression
Limit: lower = 0
  upper = 1
Total Observations: 4888
Complete observations: 3981
Missing observations: 907
Imputed observations: 907
0 values for expenditure ag6sp_4_1 converted to 1

**** EXPENDITURE ag6sp_4_2 ****
379 observations to impute
MULTIPLE IMPUTATION OVERVIEW
Method: truncreg regression
Limit: lower = 0
  upper = 1
Total Observations: 4888
Complete observations: 4509
Missing observations: 379
Imputed observations: 379
0 values for expenditure ag6sp_4_2 converted to 1

**** EXPENDITURE ag6sp_4_3 ****
60 observations to impute
MULTIPLE IMPUTATION OVERVIEW
Pay attention: imputation method switched to pmm
Method: pmm regression
Total Observations: 4888
Complete observations: 4828
Missing observations: 60
Imputed observations: 60
Number of k nearest neighbors: 241
0 values for expenditure ag6sp_4_3 converted to 1
DESCRIPTIVES STATISTICS

<table>
<thead>
<tr>
<th>Variable</th>
<th>Obs.</th>
<th>Mean</th>
<th>Median</th>
<th>Std. Dev.</th>
<th>Min</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>PUV(D)</td>
<td>4888</td>
<td>.8136841</td>
<td>.8143076</td>
<td>.1388408</td>
<td>.4111537</td>
<td>1.281451</td>
</tr>
<tr>
<td>PUV(DI)</td>
<td>4888</td>
<td>.9933574</td>
<td>.9960147</td>
<td>.169638</td>
<td>.5019528</td>
<td>1.561017</td>
</tr>
<tr>
<td>lwbp_aggr4</td>
<td>4888</td>
<td>360.8287</td>
<td>319.7612</td>
<td>223.4753</td>
<td>10.74425</td>
<td>2049.023</td>
</tr>
</tbody>
</table>

Note: lwbp_aggr4 is Pseudo Unit Values in Levels PUV(DIL)
(file kd_sp4.gph saved)

Figure 1. lwbp_aggr4 kernel density estimation

Note that the multiple-imputation procedure using the `truncreg` method for the subgroup expenditure `ag6sp_4_3` failed and that the program switched to the predictive mean matching method.

5 Conclusions

The main objective of the `pseudounit` command presented here is to make household budget surveys that collect only information about expenditures suitable for demand and welfare analysis. Thanks to the `pseudounit` command, the lack of information about quantities no longer precludes the possibility of deriving household-specific prices (unit values) and of estimating complete demand systems suitable for welfare analysis.

6 Acknowledgments

The authors wish to thank an anonymous referee and Lucia Echeverria for helpful comments and suggestions. Any errors and omissions are the sole responsibility of the authors.
7 References


**About the authors**

Martina Menon is an assistant professor of economics at the University of Verona in Italy. Her main research interests are economics of the family, consumption analysis, welfare analysis, and applied econometrics.

Federico Perali is a full professor of economics at the University of Verona in Italy. His main research interests are economics of the family, consumption analysis, welfare analysis, and applied econometrics.

Nicola Tommasi is a research assistant at the University of Verona in Italy. His main research interests are applied econometrics and statistical programming.
Generalized maximum entropy estimation of linear models

Paul Corral  
The World Bank  
Washington, DC  
PCorralRodas@worldbank.org

Daniel Kuehn  
Urban Institute  
Washington, DC  
DKuehn@urban.org

Ermengarde Jabir  
American University  
Washington, DC  
Ejabir@American.edu

Abstract. In this article, we describe the user-written `gmentropylinear` command, which implements the generalized maximum entropy estimation method for linear models. This is an information-theoretic procedure preferable to its maximum likelihood counterparts in many applications; it avoids making distributional assumptions, works well when the sample is small or covariates are highly correlated, and is more efficient than its maximum likelihood equivalent. We give a brief introduction to the generalized maximum entropy procedure, present the `gmentropylinear` command, and give an example using the command.

Keywords: st0473, gmentropylinear, generalized maximum entropy, maximum entropy, linear

1 Introduction

In this article, we introduce a user-written command, `gmentropylinear`, for fitting a linear model using generalized maximum entropy (GME) methods. GME estimation of the linear regression model provides an alternative to traditional estimation methods such as least-squares models or generalized linear models. Instead of minimizing squared residuals or maximizing a likelihood function defined by the researcher, maximum entropy methods select the linear model coefficients that are maximally informative, using an entropy measure of information content. Maximum entropy estimation selects the most conservative or noncommittal solution to the linear model. The GME linear model developed by Golan, Judge, and Miller (1996) builds on this maximum entropy principle by introducing stochastic moments into the optimization problem. GME provides a framework for fitting models that are robust to poor specification and to data that are partial or incomplete.

2 GME linear model

Using maximum entropy prevents the econometrician from imposing moment conditions that must be fulfilled on data that may be neither large nor well behaved.

The entropy measure used here is defined by Shannon (1948) as

\[ H(p) = - \sum p_i \ln p_i \]
where $p_i$ is the probability of observing outcome $i$. Any base for the logarithm will provide a viable entropy measure. Shannon (1948) used log base 2 for work with the communication of bits of information, although the natural logarithm is commonly used in econometric applications. When the full probabilistic distribution of a sample is known, the entropy measure equals zero. However, when the distribution is completely unknown, the probability distribution that imposes no priors on the data (but rather only on the information contained within the data) is the uniform distribution whose entropy measure equals the maximum value allowed, given by the entropy measure described above. Thus maximizing entropy based on the sample’s information content leads to selecting the least-informed distribution.

In GME, the probabilities entering the entropy measure are weights associated with a vector of supports for both the parameters of the linear model (the $\beta$‘s) and the error terms. The maximum entropy criterion from Jaynes (1957a,b) is used to select the set of probabilities, or weights, that is maximally informative but still consistent with the empirical data. Golan, Judge, and Miller’s (1996) generalization of Jaynes’s maximum entropy criterion to include stochastic moments of the data by additively including the entropy of the error term into the objective function is implemented here.

Linear regressions are commonly used in economics to model the relationship between a variable of interest and a set of explanatory variables. The GME approach loosens the assumptions of many alternative linear model estimation techniques, such as least squares, and imposes minimal distributional assumptions. Golan, Judge, and Perloff (1996) and Golan, Judge, and Miller (1996) have used Monte Carlo simulations to demonstrate that GME discrete choice and linear models provide more stable parameter estimates as collinearity of the covariates increases than their maximum likelihood or least-squares counterparts.

The linear GME model recovers probability distributions for the coefficients and the error terms. These probability distributions make use of the available sample information. The GME method for linear models is developed by Golan, Judge, and Miller (1996, 85), and the discussion below draws from that source.

To recover probability distributions, one must reparameterize the coefficients and the error terms. The generic version of the linear model to be fit is

$$Y = X\beta + \varepsilon$$

where $Y$ is a $(T \times 1)$ matrix and $X$ is a $(T \times K)$ matrix. The linear GME model reparameterizes both the $\beta$ and the $\varepsilon$ in the generic linear model as the expected values of a random variable defined on a probability distribution. Each coefficient in the GME framework has a bounded support space $z_k$, associated with the $k$th variable, which is symmetrically built around zero and weighted by the vector $p_k$ to reflect that the econometrician may not have prior knowledge to incorporate into the support space. Alternative support spaces not built around zero are allowable if, for example, the econometrician has prior knowledge of the value of $\beta$. This would be similar in principle to the generalized cross entropy approach, where the probability weights are informed by prior knowledge rather than the supports. Both $z_k$ and $p_k$ are $(M \times 1)$
matrices. These weights must be in probability form; as such, their sum is equal to one (a requirement that will enter the system through the constraints). The coefficients are thus

\[
\beta = Zp = \begin{bmatrix}
z'_1 & 0 & 0 \\
0 & z'_2 & 0 \\
\vdots & \vdots & \vdots \\
0 & 0 & z'_K
\end{bmatrix}
\begin{bmatrix}
p_1 \\
p_2 \\
\vdots \\
p_K
\end{bmatrix}
\]

where \( Z \) is a \((K \times KM)\) matrix and \( p \) is a weight vector of dimension \( KM \times 1 \). There are \( M \) supports for each coefficient, and each support is associated with a probability weight. The support space recommended by Golan, Judge, and Miller (1996) is symmetric around zero and can be widened as needed to ensure that the true value for \( \beta \) lies inside the support space. Consequently, the product of these matrices is \((K \times 1)\), and for any particular \( k \), the coefficient is equal to

\[
\beta_k = \sum_m z_{km} p_{km}
\]

Similarly, the error terms must also be parameterized as follows:

\[
\varepsilon \equiv Vw = \begin{bmatrix}
v'_1 & 0 & 0 \\
0 & v'_2 & 0 \\
\vdots & \vdots & \vdots \\
0 & 0 & v'_T
\end{bmatrix}
\begin{bmatrix}
w_1 \\
w_2 \\
\vdots \\
w_T
\end{bmatrix}
\]

Thus an individual observation’s error term is equal to

\[
\varepsilon_t = \sum_j w_{tj} v_j
\]

where \( w_{tj} \) is the set of proper probabilities for each \( t \) and \( w \) is the \((TJ \times 1)\) vectorization of \( w \). It is common to build the error support using the three-sigma rule \( v = (-3\hat{\sigma}_Y, 0, 3\hat{\sigma}_Y) \), where \( \hat{\sigma}_Y \) is the sample standard deviation for the dependent variable \( Y \) (Pukelsheim 1994). The linear model under the reparameterization done by Golan, Judge, and Miller (1996) becomes

\[
Y = XZp + Vw
\]

The entropy term is maximized subject to the requirements of the proper probability distributions for \( p_{km} \) and \( w_{tj} \) and the \( T \) information-moment constraints of the linear model (because \( Y \) is \((T \times 1)\), all \( T \) information moments, or data points, enter through the constraints). Therefore, the Lagrangian is

\[
\mathcal{L} = -p' \ln p - w' \ln w + \lambda' (XZp + Vw - Y) \\
+ \delta' \{1_K - (1_K \otimes 1_M) p\} \\
+ \gamma' \{1_T - (1_T \otimes 1_J) w\}
\]
where \(1_K\) is a \((K \times 1)\) vector of ones; the same holds for the other subscripts. The gradient of the Lagrangian is taken with respect to the Lagrangian parameters \((\lambda, \delta, \text{and } \gamma)\) and the probabilities \((p\text{ and } w)\). The solutions for \(p\) and \(w\) are

\[
\hat{p}_{km} = \frac{\exp\left(-z_{km} \sum_t \hat{\lambda}_t x_{tk}\right)}{\sum_m \exp\left(-z_{km} \sum_t \hat{\lambda}_t x_{tk}\right)} \equiv \exp\left(-z_{km} \sum_t \hat{\lambda}_t x_{tk}\right) / \Omega_k (\hat{\lambda}) \quad (3)
\]

and

\[
\hat{w}_{tj} = \frac{\exp\left(-\hat{\lambda}_t v_j\right)}{\sum_j \exp\left(-\hat{\lambda}_t v_j\right)} \equiv \exp\left(-\hat{\lambda}_t v_j\right) / \Psi_t (\hat{\lambda}) \quad (4)
\]

The \(\beta\)'s and \(\varepsilon\)'s can be recovered by substituting \(p\) and \(w\) into (1) and (2). However, a more efficient way to solve for the \(\beta\)'s is to solve the unconstrained dual formulation of the problem, which is a function of the \(\lambda\)'s. The resulting function is referred to as the minimal value function by Golan, Judge, and Miller (1996).

\[
\max_{p, w} H(p, w) = \min_\lambda \left\{ \sum_t y_t \lambda_t + \sum_k \ln \Omega_k (\lambda) + \sum_t \ln \Psi_t (\lambda) \right\}
\]

Minimizing with respect to \(\lambda\) solves the dual formulation and provides the optimal \(\lambda\)'s, which are subsequently used to solve for the optimal \(p\)'s and \(w\)'s [(3) and (4)]. These, in turn, are the weights associated with the support space, which generates the parameters and the errors. Although this approach appears more roundabout, it offers greater efficiencies because there are only \(T\lambda\)'s to estimate rather than a combination of \(KM\) \(p\)'s and \(TJ\) \(w\)'s. The gmentropylinear command optimizes the dual unconstrained model using the Newton–Raphson method, following the implementation in Cameron and Trivedi (2010). All analyses conducted by the authors have rapidly converged using this procedure.

### 2.1 Asymptotic variance of the GME estimator

Mittelhammer, Cardell, and Marsh (2013) develop the asymptotic theory and inference for the linear GME estimator.\(^1\) The authors state that the GME estimator is asymptotically and normally distributed with a variance–covariance matrix given by

\(^1\) See Mittelhammer, Judge, and Miller (2000) for a full derivation of the asymptotic properties of the model.
Generalized maximum entropy estimation of linear models

\[ \hat{\text{Var}}(\hat{\beta}) = \frac{\hat{\sigma}^2(\hat{\beta})}{\zeta^2(\hat{\beta})}(X'X)^{-1} \]

where

\[ \hat{\sigma}^2(\hat{\beta}) = \frac{1}{T} \sum_{t=1}^{T} \lambda_t^2 \]

and

\[ \zeta^2(\hat{\beta}) = \left[ \frac{1}{T} \sum_{t=1}^{T} \left\{ \sum_{j=1}^{J} v_j^2 w_{tj} - \left( \sum_{j=1}^{J} v_j w_{tj} \right)^2 \right\} \right]^{-1} \]

The variance of the estimators, because it is asymptotically normally distributed, can be used for hypothesis testing on the values of \( \beta \).

3 The gmentropylinear command

This command fits a linear model using the GME principle. The syntax reflects standard linear regression syntax in Stata, with one exception. Unlike the discrete choice version of the GME model, gmentropylogit (Corral and Terbish 2015), users must provide the parameter support space for the \( \beta \)'s in the gmentropylinear command. The support space for the error terms is set by default to a dimension of three, and it is equally and symmetrically built around zero using the three-sigma rule as suggested by Golan, Judge, and Miller (1996). However, the command allows the user to specify an alternative error support space in both dimension and values. The \( \beta \) coefficient support space is a \( K \times M \) matrix, where \( K \) is the number of covariates including the constant (estimation without the constant term is also allowed) and \( M \) is the number of supports. Although some packages for the GME linear model in other software build parameter support spaces around ordinary least-squares (OLS) estimates of the model, we recommend a support space that is symmetric around zero instead. When sample sizes are small, and a wide support space around zero is built, the GME estimator is usually a better estimator than the OLS (Mittelhammer, Cardell, and Marsh 2013). Golan, Judge, and Miller (1996, 109) note that although the GME solution is consistent, it is still likely to suffer from small-sample bias. Nevertheless, in simulations, it still has lower mean square errors than traditional methods. If the user has prior knowledge about the true \( \beta \)'s, a narrow support around this prior knowledge will provide a far better fit than its OLS counterpart, particularly in small sample sizes (Mittelhammer, Cardell, and Marsh 2013). Thus the GME’s quality is dependent upon the supports chosen. Nevertheless, wide supports built around zero usually outperform their OLS counterparts.

One of GME’s principal advantages is that it provides the solution least committed to potentially wrong specification assumptions (aside from linearity). Building the
support space from potentially mistaken prior OLS estimates will push the GME result toward the OLS result by adding information from the OLS specification into the entropy maximization problem. As Golan, Judge, and Miller (1996) note, a wider support space increases the impact of the data, while a tighter support space increases the impact of the support. The support space is generated simply by defining an appropriate matrix:

```
matrix matname = matrix_expression
```

Once a matrix for the \( \beta \) support has been specified, the user can proceed with the command:

```
gmentropylinear depvar [ indepvars ] [ if ] [ in ], support(matrix)
  [ sigmavalue(#) endpoint(#) lambda(string) wmat(string) residual(string) 
    nosigma nocons ]
```

The `support(matrix)` component of the command identifies the matrix defining the support space for the coefficients and must be provided by the user as a matrix. `support()` is required. The `gmentropylinear` command provides various options. Although the specification of the coefficient support is required, the support for the error terms is constructed by the command. However, the user can modify the number of support spaces and decide whether to use the three-sigma rule. The `endpoint(#)` option tells the `gmentropylinear` command how many supports are used in the estimation for the error term. The default for the error supports is to use the empirical three-sigma rule, with \( J = 3 \). Thus the default error space for each observation is \( v = (-3\hat{\sigma}_Y, 0, 3\hat{\sigma}_Y) \). The `sigmavalue()` option is used to specify the sigma endpoint; the default is `sigmavalue(3)`. If the user specifies `sigmavalue(4)`, then \( v = (-4\hat{\sigma}_Y, 0, 4\hat{\sigma}_Y) \). The use of the empirical \( \sigma \) can also be overridden with the `nosigma` option. Therefore, if the user specifies `sigmavalue(4)` and `nosigma`, then \( v = (-4, 0, 4) \). Finally, the user can adjust the number of supports \( J \). This is done with the `endpoint()` option; the default is `endpoint(3)`. Regardless of the number of supports specified, the command will always adjust this to be odd numbered.

Several options return estimates from the model, including `lambda()`, which returns the estimated \( \lambda \)'s, `wmat()`, which returns the estimated \( w \), and `residual()`, which returns the residuals. One can also suppress the constant term with the `nocons` option.

Besides the estimates of the coefficients and their standard errors, `gmentropylinear` provides the final entropy for the model as well as the normalized signal entropy:

\[
S(\hat{p}) = -\frac{\sum_k \sum_j p_{kj} \ln p_{kj}}{K \ln M}
\]

The normalized entropy for the noise parameters is also included as follows:

\[
S(\hat{w}) = -\frac{\sum_t \sum_j w_{tj} \ln w_{tj}}{T \ln J}
\]
Generalized maximum entropy estimation of linear models

The pseudo-$R^2$, measuring goodness of fit within the sample analyzed, follows from the following normalized entropy metric:

$$\text{Pseudo-}R^2 = 1 - S(\hat{\rho})$$

The normalized entropy measure scales the estimated entropy by the maximum possible entropy of a problem with the provided number of variables, elements of the support space, and observations. The maximum possible entropy occurs only when the distribution of probabilities over the $p_m$’s and the $w_j$’s is uniform and departs from this maximum entropy point only when the moment constraints are included (Soofi 1992). The data, which enter through the moment constraints, add information that reduces uncertainty and pushes the solution away from the uniform distribution. As its name suggests, the normalized entropy figure can be compared across analyses and ranges between zero and one.

4 Example

The following example uses the `gentropylinear` command to estimate the (logged) price of a car in Stata’s `auto.dta`. Price is estimated here as a function of miles per gallon, weight, and whether the car is foreign. First, the parameter support matrix (`support()`) is defined. This is a $(K \times M)$ matrix, where $K = 4$, because there are three coefficients and a constant term to estimate. In this case, $M = 5$, although other dimensions are admissible. Running the estimation without a constant is also possible with the `nocons` option.
. sysuse auto
  (1978 Automobile Data)
. generate lnprice=ln(price)
. matrix support=(-1,-.5,0,.5,1)
> (-5,-2.5,0,2.5,5)
. gmentropylinear lnprice mpg weight foreign, support(support) sigmavalue(3)
> endpoint(3) wmat(err) residual(error) lambda(lambda)
Iteration 1:  Entropy = 87.735061
Iteration 2:  Entropy = 82.891972
Iteration 3:  Entropy = 82.381972
Iteration 4:  Entropy = 82.3207142
Iteration 5:  Entropy = 82.3199698
Iteration 6:  Entropy = 82.3199698
Iteration 7:  Entropy = 82.3199698
Generalized Maximum Entropy (Linear) Number of obs = 74
Degrees of freedom = 3
Model Entropy = 82.3
Pseudo R2 = 0.2133
Signal entropy = 0.7867
Noise entropy = 0.9503

|        | Coef.  | Std. Err. | z     | P>|z|   | [95% Conf. Interval] |
|--------|--------|-----------|-------|-------|----------------------|
| mpg    | 0.0455501 | 0.0107623 | 4.23  | 0.000 | 0.0244564 0.0666437  |
| weight | 0.0008686 | 0.0000915 | 9.50  | 0.000 | 0.0006893 0.0010478  |
| foreign| 0.7168477 | 0.0991786 | 7.23  | 0.000 | 0.5224612 0.9112342  |
| _cons  | 4.824633  | 0.4896719 | 9.85  | 0.000 | 3.864894  5.784372  |

Finally, `gmentropylinear` stores the following in `e()`:

Scalars
- `e(N)`: number of observations
- `e(df_m)`: model degrees of freedom
- `e(entropy)`: final entropy for the model
- `e(int_entropy)`: initial entropy
- `e(pseudoR2)`: pseudo-$R^2$
- `e(sig entropy)`: normalized entropy for the signal
- `e(noise_entropy)`: normalized entropy for the noise

Macros
- `e(depvar)`: name of dependent variable
- `e(properties)`: `b V`

Matrices
- `e(b)`: coefficient vector
- `e(V)`: variance–covariance matrix of the estimators
- `e(esupport)`: error support space specified
- `e(betaprobs)`: coefficient parameter support space

Functions
- `e(sample)`: marks estimation sample
Generalized maximum entropy estimation of linear models

5 Conclusion

In this article, we described the user-written gmentropylinear command, which provides users with a GME alternative for fitting linear models. GME estimation is particularly advantageous when estimation is performed on small datasets or when the user is unsure about the appropriate model specification and would like to find the maximally noncommittal solution. The GME approach shrinks the joint entropy distance between the data being analyzed and an assumption of uniform priors where there is complete uncertainty about the underlying distribution. Implementing a dual unconstrained model allows for equal emphasis to be placed on the precision of the estimates as well as on prediction, where the estimated probabilities yield the distribution of the parameters of interest (the error and coefficient estimates) up to the $M - 1$ moment for the coefficient and the $J - 1$ moment for the error term (Golan, Judge, and Perloff 1996). This approach also benefits from being robust to collinearity. As with other robust estimators, discrete support spaces for the parameters (coefficients and errors) must be specified. The command outlined here generates GME estimates derived from user-specified parameter supports and error supports that are provided either by the user or by the defaults.

6 Acknowledgments

The code presented in this article was written with the support and encouragement of Amos Golan and is part of the examples used in his upcoming book on information theory. Additionally, we would like to thank Minh Nguyen for comments provided on the code. Finally, we thank the reviewer.

Any mistake or omission is the authors’ responsibility alone.

7 References

Cameron, A. C., and P. K. Trivedi. 2010. Microeconometrics Using Stata. Rev. ed. College Station, TX: Stata Press.


P. Corral, D. Kuehn, and E. Jabir 249


About the authors

Paul Corral is a data scientist in the Poverty and Equity Global Practice at the World Bank.

Daniel Kuehn is a research associate in the Income and Benefits Policy Center at the Urban Institute.

Ermengarde Jabir is an economics PhD candidate at American University.
Software Updates


The simulations were calculating a $p$-value based on “more extreme”. Now they use “at least as extreme”, as originally advertised. Effectively, I swapped “$>$” for “$\geq$” on lines 252 and 287 of `skilmack.ado`.