Dynamic Documents in Stata

Bill Rising

StataCorp LLC

2018 Canadian Stata Conference
Simon Fraser University
27 July 2018

The Good and Bad of Creating Documents

- Think of documents you've made in the past, good and bad
- Good:
    - Reused ideas from one project for another
    - Reused and polished lessons for teaching
- Bad:
    - Questions on methods for reaching particular numerical results
    - Updating analyses because of new or improved data
    - Producing repetitive reports

General Idea

- What gets done once often gets done twice
  - Similar projects
  - Updated datasets
  - Datasets arriving over time or from various sources
  - Teaching
  - Production work, such as dreaded monthly reports

- The second and later repetitions should not start from scratch

Dynamic Documents

- Needed: reproducible, reusable, and maintainable documents, aka dynamic documents
  - Documents should be reproducible at the push of a button
    - No manual intervention!
  - Documents should be reusable
  - Documents should be easily maintained and improved
    - This is especially necessary for teaching
- Both of these are easy for pure narratives
- Including computational results is trickier
- Making this nice for all collaborative parties is even trickier

STATA 15

Best Possible Process

- One underlying file for producing a final document, including both narrative and computation
  - If not a single document, a single folder with easily-related files
- The final document can be reliably reproduced from scratch
- Drafts of the final document can be passed around to all collaborators
  - Topic experts as well as statistical experts as well as writers
  - Those comfortable with programmerish work and those who are not
- The final document could be in a variety of forms

STaTa 15

What We'll See Here

- Several tools for producing dynamic documents
- Some way of deciding between complexity, completeness, and comprehension

Bare Necessities for Teaching

- Commands
- Results
- Graphs

Bare Necessities for Reports

- Results without commands
- Inline results
  - Results often show up within the narrative
- Invisible commands

Dream World

- Extremely readable documents
- Flexible formatting

Introduction
Dynamic Documents
Software Review
Conclusion

Overview
Concept
markstat
dyndoc
putdocx
putwrap

Included Software

- We will look at three and one half pieces of software
- Germán Rodríguez' `markstat` command
- Stata's official `dyndoc` command
- Stata's official `putdocx` command
- A wrapper to (possibly) make `putdocx` simpler, called `putwrap`

STATA 15

Introduction
Dynamic Documents
Software Review
Conclusion

Overview
Concept
markstat
dyndoc
putdocx
putwrap

Excluded Software

- The software below was covered in a similar talk in 2016:
  - texdoc for making documents which are like Stata Journal articles
    - Still relevant
  - Markdoc for creating general-purpose documents in many formats
  - StatWeave for making general-purpose documents
  - A suite for producing lessons with handouts

STaTa 15

Introduction
Dynamic Documents
Software Review
Conclusion

Overview
Concept
markstat
dyndoc
putdocx
putwrap

Terminology

- It will help to have some defined jargon here to refer to files

  - A *base* file gets processed by the software
  - The result of the processing is an *interim* file, if that file needs more processing
  - The document as it would be viewed will be called a *final* file

    - This is not final as in "final draft"

STaTa 15

Introduction
Dynamic Documents
Software Review
Conclusion

Overview
Concept
markstat
dyndoc
putdocx
putwrap

Working Through the Examples

- Much as something fully interactive would be nice, typing is dull
- We'll look at examples of files for each of the methods and then see if we can get them to turn into documents
- Most of the talk will be spent looking at these files
- When this talk is posted, all the example files will be in the file `repdoc.zip`

STATA 15

Introduction
Dynamic Documents
Software Review
Conclusion

Overview
Concept
markstat
dyndoc
putdocx
putwrap

A Sketch of What to Do

- Here is a basic outline of a small evaluation we'd like to do
    - This is in the data/shared/pseudo.txt file

- It has a few items of interest
    - Stata commands and output
    - Graphics
    - A table from tabout
    - An unnumbered list
    - Boldface, italics and fixed-width fonts

- We would like to realize this report (or something close to it) in different ways

STATA 15

Introduction
Dynamic Documents
Software Review
Conclusion

Overview
Concept
markstat
dyndoc
putdocx
putwrap

markstat Basics

- markstat was written and is maintained by Germán Rodríguez
- markstat is based on the **markdown** language
- markstat can produce most any document type you would like
- markstat can be used in either simple markdown mode or in a strict mode
- Narrative and code are in the same file

STATA 15

Introduction
Dynamic Documents
Software Review
Conclusion

Overview
Concept
markstat
dyndoc
putdocx
putwrap

markstat Process

- markstat processes a markdown file to produce the end document
- markstat produces many small files containing code and output
  - By default these get deleted, but they can be kept
- It is possible to regenerate the document without running the Stata commands
  - While dangerous in general, this is useful when fixing typos in the narrative
  - Germán credits taking this idea from Ben Jann's texdoc

Introduction
Dynamic Documents
Software Review
Conclusion

Overview
Concept
markstat
dyndoc
putdocx
putwrap

markstat Advantages

- Can be quite simple
  - Simplicity can lose some important features
- Can be made more complex
  - The added complexity reduces the readability of the base file
- Has the ability to include external files as the markdown gets processed
  - This is not possible in vanilla markdown

STATA 15

Introduction
Dynamic Documents
Software Review
Conclusion

Overview
Concept
markstat
dyndoc
putdocx
putwrap

`markstat` Disadvantages

- Markdown has some limitations
- Unfortunately, markdown doesn't have some hidden rarely-used constructions which allow extra complexity

STATA 15

Introduction
Dynamic Documents
Software Review
Conclusion

Overview
Concept
markstat
dyndoc
putdocx
putwrap

markstat Installation & Dependencies

- Getting markstat itself is simple
  - . ssc install markstat
- It does require another piece of Stata software
  - . ssc install whereis
- It also requires Pandoc (http://pandoc.org)
- If you want to use LaTeX, you need to install the package for your OS
  - You also need to get Stata's style file
  - Instructions for this are at the site (http://data.princeton.edu/stata/markdown)

STATA 15

Introduction
Dynamic Documents
Software Review
Conclusion

Overview
Concept
markstat
dyndoc
putdocx
putwrap

dyndoc Basics

- dyndoc is an official Stata command
- dyndoc uses markdown for its formatting language
- dyndoc makes web pages (HTML)
- Narrative and code are in the same file
- Rather than indentation or code fences, dyndoc use its own dyndoc tags

STaTa 15

Introduction
Dynamic Documents
Software Review
Conclusion

Overview
Concept
markstat
dyndoc
putdocx
putwrap

dyndoc Process

- dyndoc takes a markdown + Stata file and turns it into an html file
- There are no interim files

Introduction
Dynamic Documents
Software Review
Conclusion

Overview
Concept
markstat
dyndoc
putdocx
putwrap

dyndoc Advantages

- There are extra dyndoc tags which allow for conditional processing
    - This can be useful in dreadful monthly reports for calling out rare events
- Has the ability to include external files as the markdown gets processed
- It's built in to Stata

STaTa 15

Introduction
Dynamic Documents
Software Review
Conclusion

Overview
Concept
markstat
dyndoc
putdocx
putwrap

dyndoc Disadvantages

- The tags can look a bit cluttered
  - The clutter is not as bad when the file is viewed as a Stata do-file in your text editor

STATA 15

Introduction
Dynamic Documents
Software Review
Conclusion

Overview
Concept
markstat
dyndoc
putdocx
putwrap

dyndoc Dependencies

- None, of course

Introduction
Dynamic Documents
Software Review
Conclusion

Overview
Concept
markstat
dyndoc
putdocx
putwrap

putdocx Basics

- putdocx is an official Stata command
- putdocx makes docx documents
  - The documents are based on the open standard for docx
  - So... putdocx works best with Open Office and its relatives
  - putdocx also works well with Microsoft Office

Introduction
Dynamic Documents
Software Review
Conclusion

Overview
Concept
markstat
dyndoc
putdocx
putwrap

putdocx Process

- putdocx allows writing text, tables and graphs
- It does not write Stata commands or their output directly
  - It is made more for reports than for reporting on Stata
- It is always in Stata mode

STATA 15

Introduction
Dynamic Documents
Software Review
Conclusion

Overview
Concept
markstat
dyndoc
putdocx
putwrap

putdocx Advantages

- Easy to push out estimation tables
- Very flexible table generation
  - Can write line by line to update a table rather than needing to write one single massive command
- Has a lot of user interest, so there are a slew of community-contributed aids

Introduction
Dynamic Documents
Software Review
Conclusion

Overview
Concept
markstat
dyndoc
putdocx
putwrap

putdocx Disadvantages

- putdocx documents look like pure code
  - Tough on collaborators
- Changing small pieces can take some effort.

STATA 15

putdocx Dependencies

- None, of course

STaTa 15

Introduction
Dynamic Documents
Software Review
Conclusion

Overview
Concept
markstat
dyndoc
putdocx
putwrap

putwrap Basics

- putwrap attempts to allow putdocx to have a narrative mode and a Stata mode
- Otherwise it is putdocx

STATA 15

Introduction
Dynamic Documents
Software Review
Conclusion

Overview
Concept
markstat
dyndoc
putdocx
putwrap

putwrap Process

- By default, it is assumed that the do-file is in narrative mode (i.e. writing the document)

- To go into Stata mode, use `putdocx pause`

- To go back to narrative mode, use `putdocx resume`
  - Adding two subcommands to an official Stata command breaks all the rules for community-contributed software
  - Be forewarned!

- `putwrap` takes a file using these commands, and created a do-file which has all the requisite paragraph and text commands

STaTa 15

Introduction
Dynamic Documents
Software Review
Conclusion

Overview
Concept
markstat
dyndoc
putdocx
putwrap

putwrap Advantages

- It should make documents with long narrative sections easier to read

Introduction
Dynamic Documents
Software Review
Conclusion

Overview
Concept
markstat
dyndoc
putdocx
putwrap

putwrap Disadvantages

- If there are a lot of font changes, it is still necessary to break up the narrative
- It is not a clever program, so it can get fooled if lines start with special contructions (like inline macro expansions)

STaTa 15

Introduction
Dynamic Documents
Software Review
Conclusion

Overview
Concept
markstat
dyndoc
putdocx
putwrap

putwrap Dependencies

- Needs `putdocx`

Conclusion

- There are plenty of packages out there for making dynamic documents
- The quality of the packages has greatly increased in the past couple of years
- You should really give this a try