Title

stata.com

putdocx intro — Introduction to generating Office Open XML (.docx) files

Description   Remarks and examples   References   Also see

Description

The `putdocx` suite of commands creates Office Open XML (`.docx`) documents that include text, formatted images, and tables of Stata estimation results and summary statistics. The following commands are used to create, format, add content to, and save `.docx` files that are compatible with Microsoft Word 2007 and later:

Create, save, and append .docx files (see [RPT] **putdocx begin**)

`putdocx begin`	Creates a `.docx` file for export
`putdocx describe`	Describes contents of the active `.docx` file
`putdocx save`	Saves and closes the `.docx` file
`putdocx clear`	Closes the `.docx` file without saving the changes
`putdocx append`	Appends the contents of multiple `.docx` files

Insert page breaks in a .docx file (see [RPT] **putdocx pagebreak**)

`putdocx pagebreak`	Adds a page break to the document
`putdocx sectionbreak`	Adds a new section to the document

Add paragraphs with text and images (see [RPT] **putdocx paragraph**)

`putdocx paragraph`	Adds a new paragraph to the active document
`putdocx text`	Adds text to the active paragraph
`putdocx textblock`	Adds a block of text to the active paragraph or to a new paragraph
`putdocx textfile`	Adds a block of preformatted text to a new paragraph with a predefined style
`putdocx image`	Appends an image to the active paragraph
`putdocx pagenumber`	Adds page numbers to a paragraph in a header or footer

Add tables to a .docx file (see [RPT] **putdocx table**)

`putdocx table`	Creates a new table in the `.docx` file containing estimation results, summary statistics, or data in memory

Add a table from a collection to a .docx file (see [RPT] **putdocx collect**)

`putdocx collect`	Adds a customized table created by `collect` or `table` to the `.docx` file

In this manual entry, we show you how to use the `putdocx` commands by walking you through a first example that creates a simple report as a `.docx` file. We also provide some suggestions for choosing the best workflow for creating your own `.docx` files.

Remarks and examples

Remarks are presented under the following headings:

Introduction
A first example
    Create a document
    Add a paragraph with text
    Add an image to a paragraph
    Add a table of estimation results
Automating a report
Workflow options for report building
    Create a complete document in Stata
    Create a document from Stata and Word
    Append files in Stata
    Append files in Word

Introduction

putdocx is a suite of commands used to write paragraphs, images, and tables to an Office Open XML (.docx) file. This allows you to create Word documents that include Stata results and graphs. putdocx generates files compatible with Microsoft Word 2007 and later.

A first example

To get started with the putdocx commands, it is best to see them in action. Here, we demonstrate how to create a .docx file, include text, add a graph, and incorporate an estimation table all from within Stata.

This example shows the basic tools you need to create your own document. However, this is only a starting point. You may want to create more extensive and more customized documents, and putdocx allows you to do that. We save the details of customizing text, tables, and images for the individual entries of the commands listed above.

Create a document

To demonstrate, we create a report on low birthweight using data from the study described in Hosmer, Lemeshow, and Sturdivant (2013, 24).

```
. use https://www.stata-press.com/data/r18/lbw
(Hosmer & Lemeshow data)
```

Before we can add any content to the report, we first need to create an active .docx document in memory. We do this with the putdocx begin command.

```
. putdocx begin
```

Because we did not include any options with putdocx begin, the document created uses the letter page size and the portrait orientation.

Add a paragraph with text

Now that the document is created, we can add other objects such as paragraphs, images, and tables to it. We begin by adding a title to our report. To do this, we add a paragraph using the `Title` style. Then we add the text of our title using `putdocx text`.

```
. putdocx paragraph, style(Title)
. putdocx text ("Report on low birthweights")
```

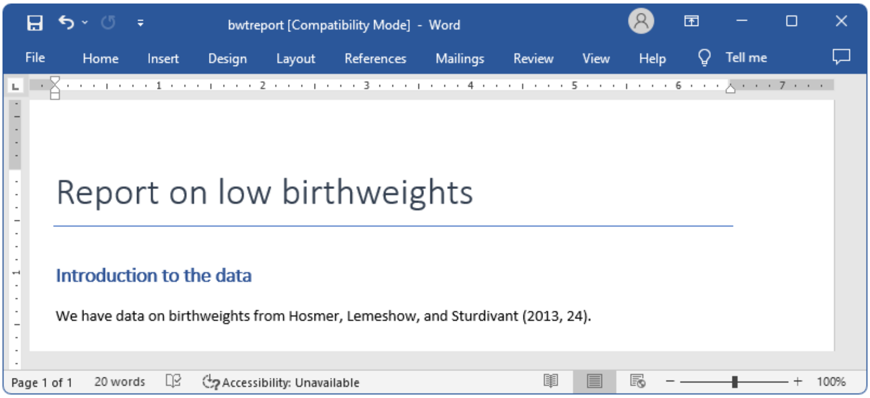
Next we add a heading for the description of our data:

```
. putdocx paragraph, style(Heading1)
. putdocx text ("Introduction to the data")
```

Now we are ready to add a standard paragraph where we cite the source of our dataset.

```
. putdocx paragraph
. putdocx text ("We have data on birthweights from Hosmer, Lemeshow, and ")
. putdocx text ("Sturdivant (2013, 24).")
. putdocx save bwtreport
successfully created "C:/mypath/bwtreport.docx"
```

We save the document we created in memory under the filename `bwtreport.docx`. When we open the document in Word, we see the following:



Report on low birthweights

Introduction to the data

We have data on birthweights from Hosmer, Lemeshow, and Sturdivant (2013, 24).

When we typed `putdocx save` above, our work was saved and the document was closed, so we now type `putdocx begin` to continue our work. So far, we have only added strings to our paragraphs, but text can also include any valid Stata expression. In the next section of our report, we add text with summary statistics for our data by referring directly to the results stored after `summarize`. We type `return list` and see that the mean is stored in the `r(mean)` scalar and with more decimal places than we wish to include in our sentence. Therefore, we use the `%5.2f` format to request that only two digits be displayed after the decimal.

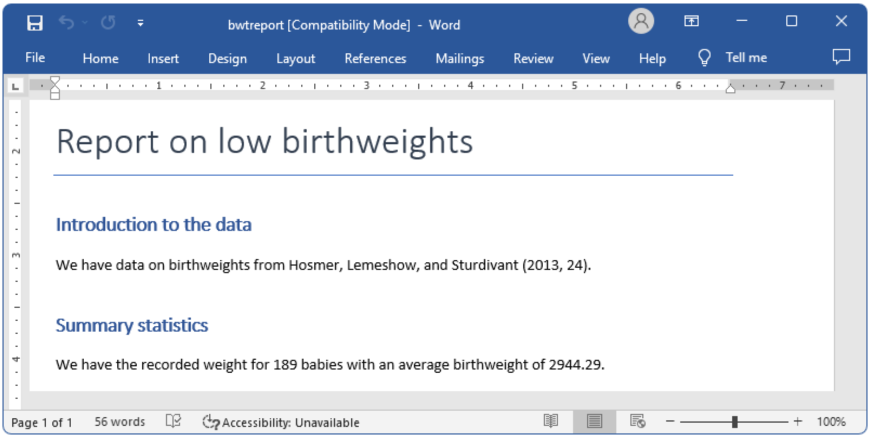
```
. putdocx begin
. putdocx paragraph, style(Heading1)
. putdocx text ("Summary statistics")
. summarize bwt
    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
         bwt |        189    2944.286     729.016        709       4990
```

```
. return list
scalars:
                  r(N) =  189
              r(sum_w) =  189
               r(mean) =  2944.285714285714
                r(Var) =  531464.3541033434
                 r(sd) =  729.0160177275554
                r(min) =  709
                r(max) =  4990
                r(sum) =  556470
. putdocx paragraph
. putdocx text ("We have the recorded weight for ‘r(N)’ babies ")
. putdocx text ("with an average birthweight of ")
. putdocx text (" ‘r(mean)’ "), nformat(%5.2f)
. putdocx text (".")
```

To review our document after adding this section, we save it again. However, because we want to add this new content to our existing `bwtreport.docx` file, we specify that we are appending to the file.

```
. putdocx save bwtreport, append
successfully appended to "C:/mypath/bwtreport.docx"
```

Our updated `bwtreport.docx` now looks like this:



Report on low birthweights

Introduction to the data

We have data on birthweights from Hosmer, Lemeshow, and Sturdivant (2013, 24).

Summary statistics

We have the recorded weight for 189 babies with an average birthweight of 2944.29.

Add an image to a paragraph

Next we graphically compare the average birthweights for babies according to the mother’s characteristics. We begin this section of the report by adding another heading.

```
. putdocx begin
. putdocx paragraph, style(Heading1)
. putdocx text ("Birthweight by mother’s smoking status")
```

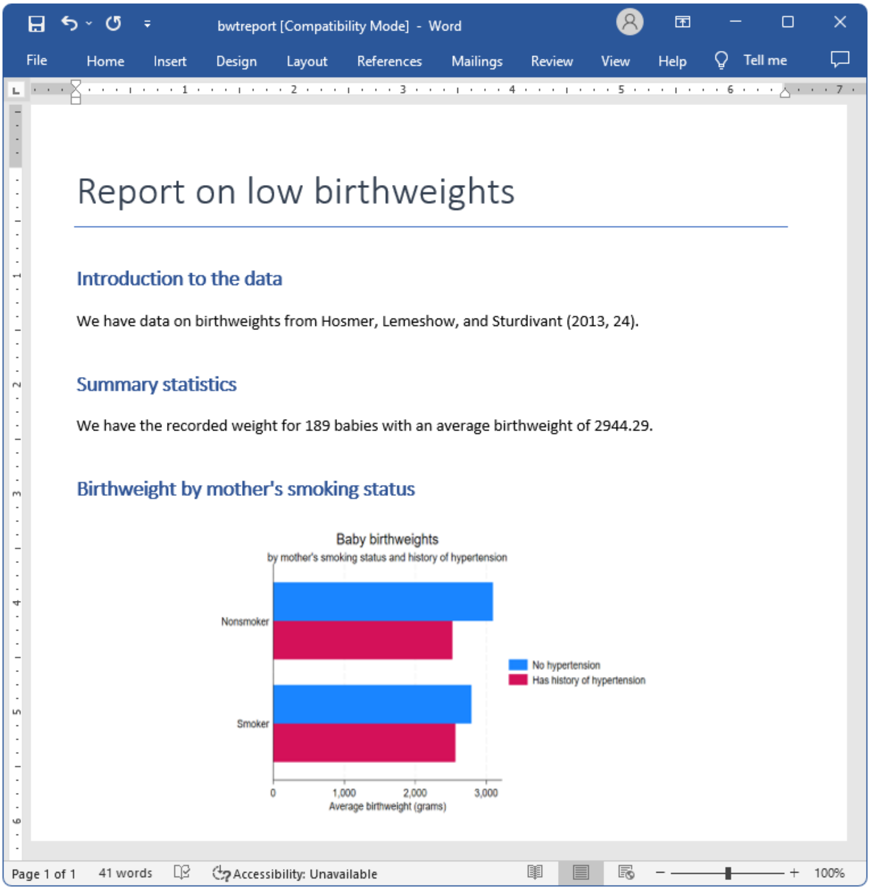
We graph the mean birthweight for babies with mothers who smoke versus those who do not, and separately for mothers with and without a history of hypertension. We use `graph hbar` to create our graph, specifying a title for the overall graph as well as for the y axis. We must convert the graph to one of the supported image formats: `.jpg`, `.emf`, `.tif`, or `.png`. We save it as a `.png` file with `graph export`.

```
. graph hbar bwt,
> over(ht,relabel(1 "No hypertension" 2 "Has history of hypertension"))
> over(smoke) asyvars ytitle(Average birthweight (grams)) title(Baby birthweights)
> subtitle(by mother's smoking status and history of hypertension)
. graph export bweight.png
file bweight.png saved as PNG format
```

Now we use `putdocx image` to append it to the active paragraph. To center the image, we specify the alignment of the paragraph. We also resize the image by setting the width at 4 inches and the height at 2.8 inches.

```
. putdocx paragraph, halign(center)
. putdocx image bweight.png, width(4) height(2.8)
. putdocx save bwtreport, append
successfully appended to "C:/mypath/bwtreport.docx"
```

We are ready to save our work again and take a look at our report. Again, we specify the `append` option with `putdocx save` to add the bar graph to the existing content of `bwtreport.docx`. Now the document contains the sections we previously exported plus the bar graph:


bwtreport [Compatibility Mode] - Word
File
Home
Insert
Design
Layout
References
Mailings
Review
View
Help
Tell me

Report on low birthweights

Introduction to the data

We have data on birthweights from Hosmer, Lemeshow, and Sturdivant (2013, 24).

Summary statistics

We have the recorded weight for 189 babies with an average birthweight of 2944.29.

Birthweight by mother's smoking status

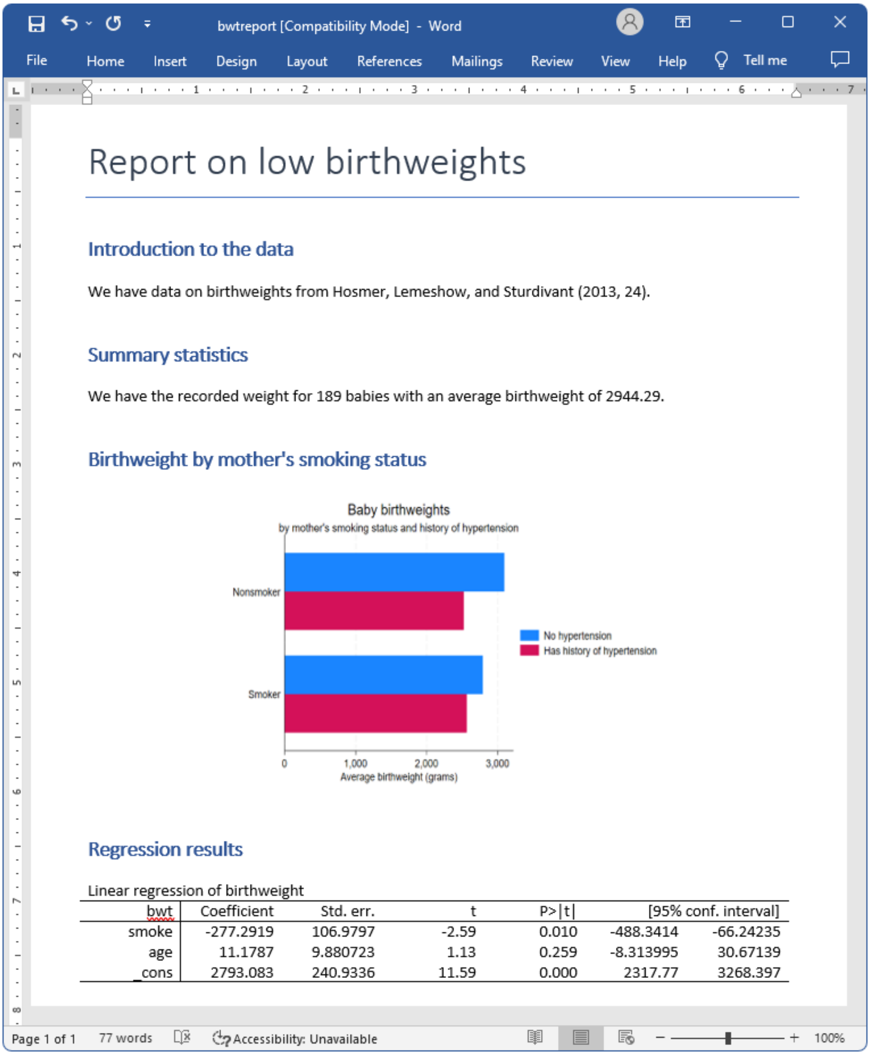
Baby birthweights
by mother's smoking status and history of hypertension
Nonsmoker
Smoker
0
1,000
2,000
3,000
Average birthweight (grams)
No hypertension
Has history of hypertension
Page 1 of 1
41 words
Accessibility: Unavailable
100%

Add a table of estimation results

Next we add a table with regression results after modeling birthweight as a function of the mother's age and whether she smokes. We export a table named `bweight`, including all the statistics shown in the regression output below. This is done effortlessly by specifying the `etable` output type with `putdocx table`. We also use the `title()` option to add a title to our table.

```
. putdocx begin

. putdocx paragraph, style(Heading1)

. putdocx text ("Regression results")

. regress bwt smoke age, noheader
------------------------------------------------------------------------------
         bwt | Coefficient  Std. err.      t    P>|t|     [95% conf. interval]
-------------+----------------------------------------------------------------
       smoke |  -277.2919   106.9797    -2.59   0.010    -488.3414   -66.24235
         age |    11.1787   9.880723     1.13   0.259    -8.313995    30.67139
       _cons |   2793.083   240.9336    11.59   0.000      2317.77    3268.397
------------------------------------------------------------------------------

. putdocx table bweight = etable, title("Linear regression of birthweight")

. putdocx save bwtreport, append
successfully appended to "C:/mypath/bwtreport.docx"
```

Again, we use `append` to add to the existing content in the file. `bwtreport.docx` now looks like this:


bwtreport [Compatibility Mode] - Word
File
Home
Insert
Design
Layout
References
Mailings
Review
View
Help
Tell me

Report on low birthweights

Introduction to the data

We have data on birthweights from Hosmer, Lemeshow, and Sturdivant (2013, 24).

Summary statistics

We have the recorded weight for 189 babies with an average birthweight of 2944.29.

Birthweight by mother's smoking status

Baby birthweights
by mother's smoking status and history of hypertension
Nonsmoker
Smoker
🟦 No hypertension
🟥 Has history of hypertension
0
1,000
2,000
3,000
Average birthweight (grams)

Regression results

Linear regression of birthweight

bwt	Coefficient	Std. err.	t	P>|t|	[95% conf.	interval]
smoke	-277.2919	106.9797	-2.59	0.010	-488.3414	-66.24235
age	11.1787	9.880723	1.13	0.259	-8.313995	30.67139
_cons	2793.083	240.9336	11.59	0.000	2317.77	3268.397

We will treat this as our final report. However, you will likely want to create `.docx` files with more content and perhaps more customization. See [RPT] **putdocx begin** for information on formatting the document as a whole, including specifying page size, page layout, font, and headers and footers. See [RPT] **putdocx paragraph** for information on adding entire blocks of text to a document; modifying the style, font, alignment, and other formatting of a paragraph; customizing the size and location of an image; and adding content to a header or footer. See [RPT] **putdocx table** for information on creating tables from stored results, matrices, data, and even images and for information on customizing these tables. See [RPT] **putdocx collect** for information on adding customized tables created with the `collect` suite of commands to a document. Finally, see [RPT] **putdocx pagebreak** for information on adding page breaks and section breaks to your document.

Automating a report

In the process of creating our report, we saved the document after exporting each section, specifying append to add on to our previous work. Saving the document intermittently allowed us to view the document in each stage of progress to ensure that it looked the way we wanted.

Once we have the layout we want, we do not need to view the Word document at each stage. Perhaps we need to create variations of this report rather frequently. Say that we receive monthly data on birthweights from a local hospital and we want to update the report with the new data. We condense the series of commands we previously ran by omitting all but the final putdocx save command and all but the initial putdocx begin command. We save our series of commands in a do-file. We add version 18.0, or version 18.5 in StataNow, to the top of our do-file to ensure that in future versions of Stata, our commands will continue to run and produce the same results they do today. In addition, because we do not want to save the report with the same name each time we run the do-file, we add the args filename command to the top of our file; see [P] syntax for information on this command. Now we can specify the name of the new Word document to be created when we run this file with the do command.

begin bwtreport.do

```
version 18.0              // (or version 18.5 for StataNow)
args filename
putdocx begin
putdocx paragraph, style(Title)
putdocx text ("Report on low birthweights")
putdocx paragraph, style(Heading1)
putdocx text ("Introduction to the data")
putdocx paragraph
putdocx text ("We have data on birthweights from Hosmer, Lemeshow, and ")
putdocx text ("Sturdivant (2013, 24).")
putdocx paragraph, style(Heading1)
putdocx text ("Summary statistics")
summarize bwt
putdocx paragraph
putdocx text ("We have the recorded weight for ‘r(N)’ babies ")
putdocx text ("with an average birthweight of ")
putdocx text (" ‘r(mean)’ "), nformat(putdocx text ("."))
putdocx paragraph, style(Heading1)
putdocx text ("Birthweight by mother’s smoking status")
graph hbar bwt,                                                   ///
over(ht,relabel(1 "No hypertension" 2 "Has history of hypertension")) ///
over(smoke) asyvars ytitle(Average birthweight (grams))           ///
title(Baby birthweights)                                          ///
subtitle(by mother’s smoking status and history of hypertension)
graph export bweight.png
putdocx paragraph, halign(center)
putdocx image bweight.png, width(4) height(2.8)
putdocx paragraph, style(Heading1)
putdocx text ("Regression results")
regress bwt smoke age, noheader
putdocx table bweight = etable, title("Linear regression of birthweight")
putdocx save ‘"‘filename’"’, replace
```

end bwtreport.do

After saving our do-file with the name bwtreport.do, we can now type

```
. use lbw_june, clear
. do bwtreport lbwreport_june
```

to create a new report in the same format as the previous one. This report will be run on lbw_june.dta (whereas we used lbw.dta earlier), and it will be saved under the filename lbwreport_june.docx.

Workflow options for report building

The putdocx suite is both capable and flexible because it has an abundance of formatting options to create and format your document completely from within Stata. However, by exporting content to a .docx file, it also allows you to interact Stata's capabilities with Word's additional formatting features. Depending on the contents of your document, you may find that one of the following methods of interacting features in Stata and Word is most suitable to creating your document:

1. Create a Word document completely from within Stata.
2. Use Stata to append documents created in both Stata and Word to complete a report.
3. Within a Word document, insert files created with Stata to complete a report.

We discuss the advantages of each approach below.

Create a complete document in Stata

In *A first example* and *Automating a report*, we demonstrated how to create a Word document directly from Stata. We showed how to automate the process of creating a report once you have decided on specific formatting. Automation of reports and easy reproducibility are advantages of creating your document completely from within Stata. In addition, you can incorporate many of Stata's other features in the same do-file that creates the .docx file. For instance, you might include assert commands in your do-file to verify expectations you have of your data before generating your report.

One disadvantage of this approach is that it is not as conducive to reports with large amounts of text. You may prefer to type text in Word to take advantage of spell checking and other features. Also, if you create documents using Word's themes, tables of contents, bibliographies, and the like, you will need to access those from directly within Word. If either of these apply to the document you wish to create, consider method 2 or 3.

Create a document from Stata and Word

Methods 2 and 3 correspond to building your document in fragments, which may be preferable for three reasons. One is that you can write lengthy segments that are not dependent on Stata results or graphs directly in Word. While you can add blocks of text with putdocx, it may be done more easily in Word. You can then combine the file created in Word with another file created by using putdocx. Another reason is that, if you are writing a lengthy report, you can focus on one section at a time. Perhaps you have not quite decided how best to display your data graphically or what statistics you want to include in your estimation tables. You can save each section under its own filename as you complete it and then combine all the components. The third reason is that you do not have to re-create any formatting with putdocx that you already have in an existing Word document.

Whether you piece your document together in Stata or Word will likely depend on the amount of graphs and tables you will be including, and whether you are still deciding on formatting options or already have a customized template. If your report is centered on graphs and estimation results produced in Stata, you may find it easier to append all your files with putdocx—method 2. This method might also be preferred if you already have a template in Word with customized formatting.

Append files in Stata

Using this approach, the final .docx file is created in multiple steps. One or more .docx files are created in Word directly. Likewise, one or more .docx files are created using putdocx. We then use putdocx append or putdocx save, append to combine all of these files into a final .docx file.

The advantage of appending files in Stata is best explained with a hypothetical example. Suppose you are creating a report that will consist of an introduction, a graphics section, and an estimation section. You have already written a long introduction, added a header and footer, and saved your work under the filename report.docx. You create a file in Stata for your work on the graphics section and add the heading "Graphics". After revising and formatting your graphs, you save your work:

```
. putdocx save graphs
```

Next you create a file for your work on estimation results with the heading "Estimation". You test whether your table of results is better displayed under a portrait or landscape layout and whether the table should include certain statistics. Once you decide which results to display, you save your work for the estimation section:

```
. putdocx save estimation
```

Now you can complete your report by appending the graphics and estimation sections to your file containing the introduction.

```
. putdocx append report graphs estimation
```

The document report.docx will now contain your introduction, followed by the graphs, and finally the estimation results. The header and footer you formatted initially for report.docx will be applied throughout the complete document. Also, if you choose to add a table of contents to your report, the graphics and estimation headers will be incorporated into it. Using this method to build your document allowed you to experiment with the formatting of your graphs and tables, while easily applying your customized header and footer.

Append files in Word

Using this approach, the final document is created using Word. Typically, a single Word file is created that includes the majority of the content as well as the desired formatting. One or more segments of the document that contain Stata results are created using putdocx. We use Word's insert features to incorporate the documents created by putdocx into the main Word document.

One advantage of inserting portions of your report into a main Word document is that you can experiment with different layouts. Suppose you are discussing a graph in your document, and you are not sure where it should be placed. Within Word, you can try inserting the graph in different gaps between the text. You can instantly see which structure makes the most sense for your report.

Another advantage is that you can also interact Excel's capabilities by linking in an Excel worksheet in your Word document. Whether you worked directly in Excel or you worked in Stata and exported some results by using putexcel, you can link content from Excel in Word. This way, you can interact Excel's features, Word's features, and Stata results all in one document.

This approach to building your document might also be preferable if you only need to include a limited number of graphs or statistics from Stata and you have already created a Word document customized to your preference.

References

Chatfield, M. D. 2018. Graphing each individual's data over time. *Stata Journal* 18: 503–516.

Hosmer, D. W., Jr., S. A. Lemeshow, and R. X. Sturdivant. 2013. *Applied Logistic Regression*. 3rd ed. Hoboken, NJ: Wiley.

Jann, B. 2016. Creating LaTeX documents from within Stata using texdoc. *Stata Journal* 16: 245–263.

Xue, Y., C. Li, and H. Si. 2023. Reporting empirical results to .docx files. *Stata Journal* 23: 545–577.

Also see

[RPT] **putdocx begin** — Create an Office Open XML (.docx) file

[RPT] **putdocx collect** — Add a table from a collection to an Office Open XML (.docx) file

[RPT] **putdocx pagebreak** — Add breaks to an Office Open XML (.docx) file

[RPT] **putdocx paragraph** — Add text or images to an Office Open XML (.docx) file

[RPT] **putdocx table** — Add tables to an Office Open XML (.docx) file

For suggested citations, see the FAQ on citing Stata documentation.