21 Creating reports

Contents

21.1   Overview

Stata's commands for report generation allow you to create complete Word, Excel, PDF, and HTML documents that include formatted text, summary statistics, regression results, and graphs.

There are two varieties of commands for creating reports. The first includes the full output from Stata commands in the document and allows you to format the text using Markdown. The second uses stored results from Stata commands and inserts these results into text and tables in the document.

With either variety, you can create reports that are reproducible. Save the do-file or text file that runs the Stata commands and generates the report. Then rerun your commands at any time in the future to reproduce the Stata results and re-create the report. Make sure you include the version command so that your results are reproducible; see [U] **16.1.1 Version**.

These documents can also be dynamic. If your data change, simply rerun the do-file using the updated dataset. All Stata results in the report will be automatically updated.

You can also create highly customized tables to include in your report; see [R] **table intro** and [TABLES] **Intro**.

21.2   The dynamic document commands

Stata's dynamic document commands allow you to embed Stata output in text files and to create HTML files and Word documents from Markdown text and Stata output. Dynamic tags are used to process Stata commands in a text file; they run the code and export the output to the destination file.

To create text files with Stata output, you simply enclose Stata commands within these dynamic tags throughout your source file and then use dyntext to create the output file. For instance, suppose we fit a regression model by typing

```
. sysuse auto
. regress mpg weight length i.foreign
```

and we want to create a simple report that includes the output from the regression in a plain text file. In addition, we want a heading that says "Regression results" and a sentence explaining the model. We can create this text file as follows:

```
─────────────────────────────────────────── begin dynex1.txt ───────────

    Regression results
    ------------------

    Linear regression of mpg on weight, length, and foreign.

    <<dd_do>>
    sysuse auto, clear
    regress mpg weight length i.foreign
    <</dd_do>>
─────────────────────────────────────────────── end dynex1.txt ─────────
```

The «dd_do» and «/dd_do» dynamic tags tell Stata to execute the commands between them and to put the output in the output.txt file that is created when we type

```
. dyntext dynex1.txt, saving(output.txt)
```

We might instead want to create an HTML document with the regression results. We can use Markdown to format the heading and to bold the variable names in our text file as follows:

```
─────────────────────────────────────────── begin dynex2.txt ───────────

    Regression results
    ==================

    Linear regression of **mpg** on **weight**, **length**, and **foreign**.

    ~~~~
    <<dd_do>>
    sysuse auto, clear
    regress mpg weight length i.foreign
    <</dd_do>>
    ~~~~

─────────────────────────────────────────────── end dynex2.txt ─────────
```

Then we create an HTML file, dynex2.html, with the Markdown-formatted text and the regression results by typing

```
. dyndoc dynex2.txt
```

Alternatively, we could type

```
. dyndoc dynex2.txt, docx
```

to create a Word document named dynex2.docx with the same results.

If you prefer a PDF document, you can first create a Word document and then use docx2pdf to convert the Word document to a PDF file.

For further introduction to the dynamic document commands, including examples of the text files, HTML documents, and Word documents created by these commands, see [RPT] **Dynamic documents intro**. See [RPT] **Dynamic tags** for information on including graphs, results of expressions, and more in dynamic documents. Also see [RPT] **dyndoc** for examples that demonstrate how to write a single, flexible text file that dyndoc can use to create similar reports but with different variables and even different datasets.

21.3   The putdocx, putpdf, and putexcel commands

The putdocx, putpdf, and putexcel commands create customized Word, PDF, and Excel files, respectively, that include Stata results. Unlike the dynamic document commands discussed in the previous section, these commands do not include Stata output directly in the document. Instead, they place the results of Stata commands into tables and text. With a series of commands when creating a document, you can specify formatting for the entire document or specific elements of the document, what text and graphs to include, and how to incorporate the statistical results from Stata commands.

Let's say we want to create a Word document with the results from the regression

```
. sysuse auto
. regress mpg weight length i.foreign
```

We also want a header and a sentence explaining the results. We could type

```
. sysuse auto
. putdocx begin
. putdocx paragraph, style(Heading1)
. putdocx text ("Regression results")
. putdocx paragraph
. putdocx text ("Linear regression of mpg on weight, length, and foreign.")
. regress mpg weight length i.foreign
. putdocx table regtable = e(table)
. putdocx save myreg
```

This creates a Word document named myreg.docx that includes a header with the text "Regression results" and a standard paragraph with the sentence about the regression. The putdocx table regtable = e(table) command creates a table in Word using the results returned from the regress command. The table includes coefficients, standard errors, tests, and confidence intervals for each of the covariates in the model.

Creating a PDF document works in much the same way. We could type

```
. sysuse auto
. putpdf begin
. putpdf paragraph, font("",20)
. putpdf text ("Regression results")
. putpdf paragraph
. putpdf text ("Linear regression of mpg on weight, length, and foreign.")
. regress mpg weight length i.foreign
. putpdf table regtable = e(table)
. putpdf save myreg
```

to create myreg.pdf. We replaced each putdocx command with putpdf, and we specified a font size of 20 points for the heading instead of using one of Word's heading styles.

We can, similarly, put results in an Excel file.

```
. sysuse auto
. putexcel set myreg
. regress mpg weight length i.foreign
. putexcel A3 = etable
```

This creates myreg.xlsx with the header and table of regression results.

For more information on putdocx, including more extensive examples and suggested workflows, see [RPT] **putdocx intro**. For more information on putpdf, see [RPT] **putpdf intro**. For more information on putexcel, see [RPT] **putexcel**.