Notebook

January 9, 2019

Contents

1 Jupyter Notebooks (with STATA?!)

1.1 What are Jupyter Notebooks?

- A way to do literate programming
- Provide code and writing/analysis, on a language agnostic platform
  - Meaning that it is not restricted to just one language
  - Currently there are so-called kernels for many languages
  - Including Stata, Python, R, C, Golang, C++, Fortran and more coming!
- Uses the power of Markdown/Latex Math and Code to tell a story and provide an efficient workflow
- Convert into several different formats including Latex, HTML, Presentations etc...
- The Jupyter engine is also available in other text editors such as Atom and VS Code.
- And now available in STATA!

1.2 Two ways to watch presentation

- (Recommended) Start from an empty jupyter notebook and follow along, doing similar things as me
- (If something doesn't go right) Download this jupyter notebook, and follow along
  - Two ways to do that:
    - go to `https://osf.io/dpwae/`
      - click on
      - Above, click `Download as zip`
    - If you want to see a preview of the git seminar, do the following:

1.2.1 Continued

- Two ways:
  - Method 1: Console
    - Windows (only): Open Git Bash Console by pressing Start -> cmd
    - type `cd <directory where you want to the repository to be>`
      - Recommended: your Documents folder
    - type `git clone https://github.com/lordflaron/Experimetrics-BITSS-Workshop.git`
    - lordflaron is me, I used to be into World of Warcraft, don't judge me.
  - Method 2: GitHub Desktop
    - Open up Github Desktop
      - Go to File -> Clone repository -> Put in `https://github.com/lordflaron/Experimetrics-BITSS-Wor`

1.3 Troubleshooting

Stata Kernel	-	Other Stuff	
Assertion Error `pip install stata_kernel --upgrade`	Stata Conf Go to the User Directory: Windows: `C:/Users/<your username>/.stata_kernel.conf` Mac: ~/.stata_kernel.conf Linux: Anybody?	Morning Installation - Try the other prompt- Come here - for the StataDoExport, try - `pip install --upgrade https://github.com/lordflaron/StataDoExport.git`	Other Other Stuff Come here

1.4 Under the Hood

- Jupyter Notebooks are written in python and are themselves a JSON document
- Which makes them suited for working on in a browser

1.5 Starting Jupyter

- You can start Jupyter from any directory you want
- By default, it starts in your documents folder
- to change the directory, use the `--notebook-dir` flag
- Example `jupyter notebook --notebook-dir=C:/Documents and Settings/Users/WHATEVER`

1.6 Extensions

- Jupyter can be made to be a full featured IDE (Integrated Development Environment)
- Which really means you can get all kinds of nifty things
  - Autocompletion
  - Multi-cursor support
  - Scratchpad
  - Highlighting a selected word
  - Translation
  - Spellcheck

1.7 Installing Extensions

- In order to do this, we need to go to our conda console and type:

```
conda install -c conda-forge jupyter_contrib_nbextensions
```

- And restart Jupyter

1.8 Markdown

- Using the same idea as in markstat that Oscar showed you before.

1.9 Showing Math

- It is possible to show math
  - $y_{it} = \alpha + \beta \cdot X$

1.10 The Stata Kernel

- This is a relatively new kernel that is implemented by Kyle Barron, Mauricio Cáceres, and other contributors
  - It provides the ability to run code and show graphics, which was previously unavailable for Stata in Jupyter.
- Ironically, even though we are using Stata in these presentations, there are other, free, open-source languages that are just as good (if not more powerful) for which dynamic documents have existed for over a decade.
- As a small nudge towards getting you to try something like R or Python, here's an addendum that Kyle Barron wrote on this State_kernel page:

> As an ardent open-source advocate and someone who actively dislikes using Stata, it somewhat pains me that my work creates value for a proprietary, closed-source program. I hope that this program improves research in a utilitarian way, and shows to new users the scope of the open-source tools that have existed for upwards of a decade.

1.11 Running Code

- In this case we will be using the Stata kernel, so we will have Stata running in the background.

```

(1978 Automobile Data)

      Source |       SS           df       MS      Number of obs   =
      74-------------
+----------------------------------   F(2, 71)        =     14.74
       Model |   186321280         2  93160639.9   Prob > F        =
  0.0000
    Residual |   448744116        71  6320339.67   R-squared       =
  0.2934-------------
+----------------------------------   Adj R-squared   =    0.2735
       Total |   635065396        73  8699525.97   Root MSE        =
    2514-------------------------------------------------------------------
---


       price |      Coef.   Std. Err.      t    P>|t|     [95% Conf.
Interval]-------------
+-------------------------------------------------------------
      weight |   1.746559   .6413538     2.72   0.008      .467736
3.025382
         mpg |  -49.51222   86.15604    -0.57   0.567    -221.3025
122.278
       _cons |   1946.069    3597.05     0.54   0.590    -5226.245
9118.382-------------------------------------------------------------------
---

(est7 stored)

      Source |       SS           df       MS      Number of obs   =
      74-------------
+----------------------------------   F(3, 70)        =     23.29
```

```
      Model |   317252881         3   105750960   Prob > F        =
  0.0000
   Residual |   317812515        70  4540178.78   R-squared       =
  0.4996-------------
+-----------------------------   Adj R-squared   =    0.4781
      Total |   635065396        73  8699525.97   Root MSE        =
  2130.8------------------------------------------------------------------
---



      price |      Coef.   Std. Err.      t    P>|t|     [95% Conf.
Interval]-------------
+-------------------------------------------------------
     weight |   3.464706    .630749     5.49   0.000     2.206717
4.722695
        mpg |    21.8536   74.22114     0.29   0.769    -126.1758
 169.883
    foreign |    3673.06   683.9783     5.37   0.000     2308.909
5037.212
      _cons |  -5853.696   3376.987    -1.73   0.087    -12588.88
881.4934------------------------------------------------------------------
---

(est8 stored)
```

1.12 Stata Kernel Magics

- Many Jupyter kernels have something called magics
  - A way to make certain actions easy without having to write too much code
  - Stata has some magics that make things a little easier

1.13 %browse, %head, %tail

- This has the ability to choose varlist, the number of observations and with `if` statements as well

1.14 %html and %latex

- This allows the rendering of table during export into html or latex, as well as rendering in the notebook (with HTML only)

```
This front-end or document format cannot display HTML
```

1.14.1 %latex

Table 1.2: A regression table or something

	(1)
	price
mpg	-49.51
	(-0.57)
weight	1.747**
	(2.72)
_cons	1946.1
	(0.54)
N	74

t statistics in parentheses

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

1.15 %help

- You can use this to get a help file

1.16 Code in Markdown

- You can render code in a markdown cell, so you don't have to keep rewriting it!

```

    Variable |        Obs        Mean    Std. Dev.       Min
Max-------------
+-------------------------------------------------------
       price |         74    6165.257    2949.496       3291
15906
```

The mean of price is 6165.257.

1.17 Figures

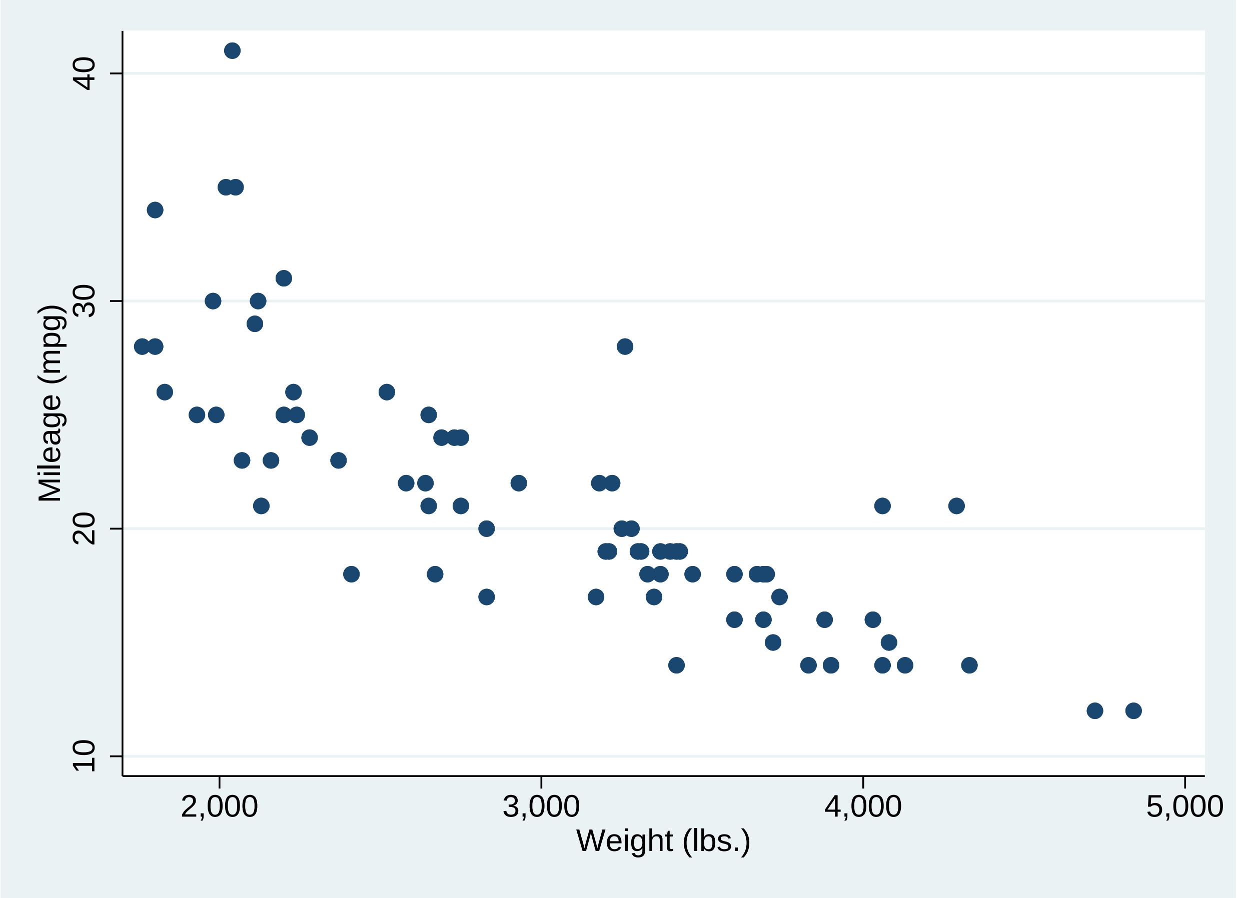

Figure 1.1: A scatter plot

1.18 Exporting

1.18.1 Using `ipypublish` to Get Publication Ready PDFs

- `ipypublish` is a utility developed for Jupyter Notebooks to make nice looking documents
- To get this working, we need to use `pip`
  - In the conda console, type `pip install ipypublish`
  - Hopefully it'll work
- Doing this requires playing with the JSON code of a cell itself (called the metadata).
- This allows a subsequent PDF output to be processed through latex, without any code cells and with figure and table environments.

1.19 Port-forwarding and setting up Jupyter to work on a server

- Many people might have servers in their universities/organizations that are more powerful than a laptop.
- Jupyter allows the ability to run a notebook locally (on your laptop screen), but using the power of the server.
  - This requires jupyter being installed on the server

    - This isn't a difficult thing to do for a sysadmin, so it's worth finding out whether that's possible

1.19.1 Setting up jupyter on a server

- The first thing you need to do is log on to the server and start a jupyter instance:

```
jupyter notebook --no-browser --port=8888
```

- This tells the server to start an instance of jupyter, without a browser (we won't need it, nor can a server open up a browser window), in port 8888 (this will be important later)
- For Mac users, you can use `ssh` to finish the process. Just type: `ssh username@host -L 8888:localhost:8888`
- Which will forward your computer 8888 port, to the server's 8888 port.
- For Windows, ssh also exists, but you will need to enable it.
    - head to Settings > Apps and click "Manage optional features" under Apps & features.
    - Click Add a Feature, and find OpenSSH
- Then use the same command as for Macs: `ssh username@host -L 8888:localhost:8888`
- Then go to your browser:
    - `localhost:8888` and you should be taken to a Jupyter page and prompted for a token.
    - You can find this token in the window where you started Jupyter on the server
        - Copy and paste this token into the prompt, and VOILA!
- Now you have Jupyter running on your computer's browser window, but with the power of the server!

1.20 Advanced Techniques

- Jinja Templates
    - Allows the control of how a notebook is exported using the Jinja templating language
- Downloading new kernals (R, Python)
    - All the above applies (with even more features with Python)
- Containerization (Docker or `virtualenv`)
    - Extreme Dynamic Documents
    - Allows not just to reproduce the code and writing
    - But also packages (`virtualenv`) and even state of operating system (`docker`)
    - Allows freezing of package dependencies so that updated packages don't stop replication
    - Relatively easier way done in `binder`
    - This uses a public GitHub repository with some special files into a docker container
    - See an example here

1.21 The Next Frontier

- Although Jupyter Notebooks are very popular and much science has been done with them (including an economics textbook: more here)
- the next generation Jupyter is Jupyter Lab, while allows extensions to be made better, and for the environment to be even better for data analysis.