

IPyStata

Stata + Python + Jupyter Notebook

“The whole is greater than the sum of its parts.”

- Aristotle

A photograph of a modern university building facade. The building is light-colored with a dark grey horizontal band across the top. On this band, the words 'TILBURG' and 'UNIVERSITY' are written in large, white, sans-serif capital letters. Between the two words is a gold-colored logo consisting of a stylized tree or plant with a crown-like top, surrounded by a circle of blue squares.

TILBURG UNIVERSITY

Introduction



Workflow with Python and Stata

Generate data



- Web scraping
- I/O operations and IOT (Internet-of-things)

Process data



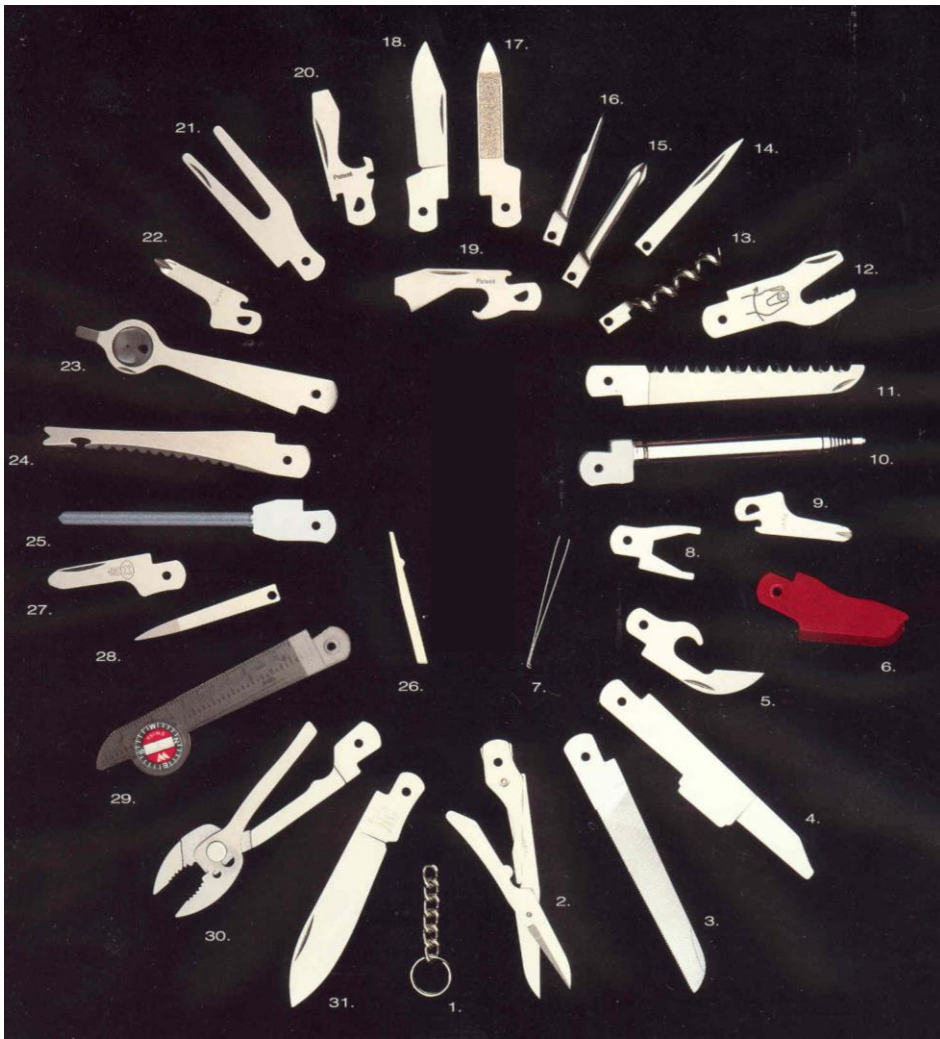
- NLP (Natural language processing)
- Ever have the urge to “just do it manually in Excel”?
→ Use Pandas!

Document all your actions in code → replicability

Insight from data



- Statistical analysis (Stata)
- Visualize results / data (Stata / Python)



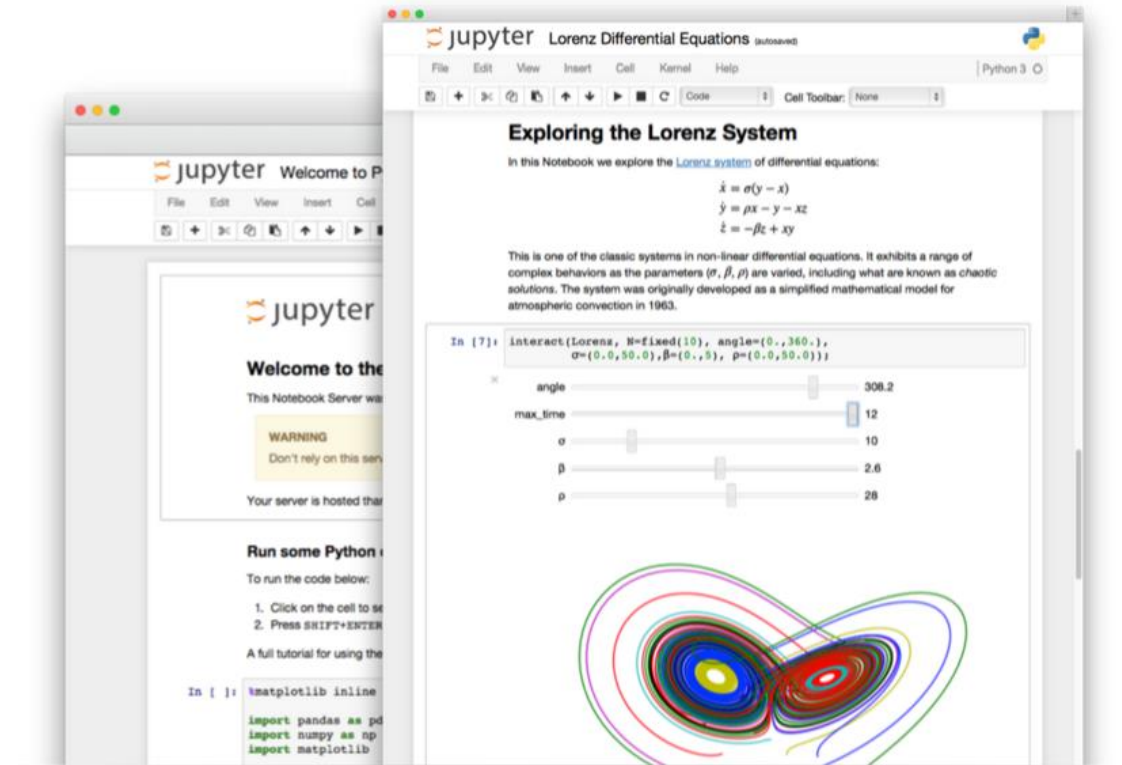
What is the Jupyter Notebook?

Open source, interactive data science and scientific computing across over 40 programming languages.

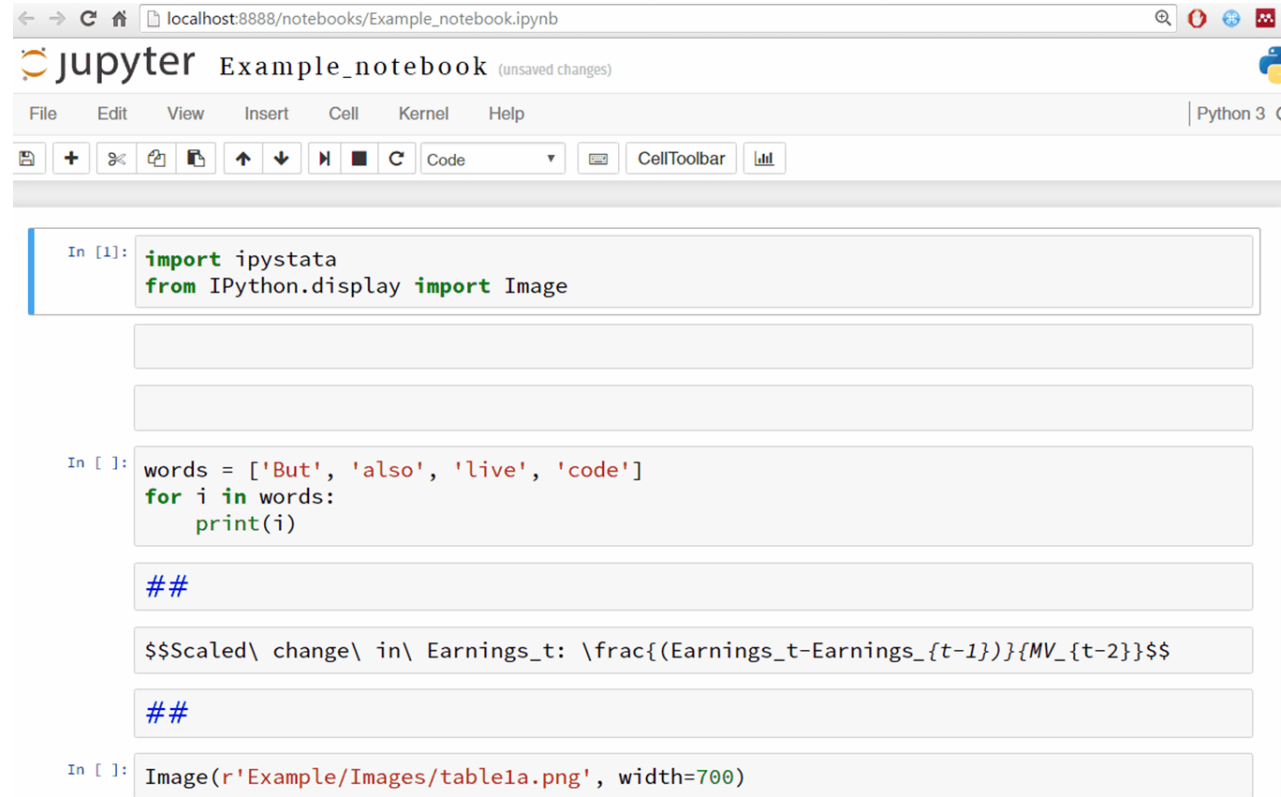


Jupyter Notebook

The Jupyter Notebook is a web application that allows you to create and share documents that contain live code, equations, visualizations and explanatory text. Uses include: data cleaning and transformation, numerical simulation, statistical modeling, machine learning and much more.



Demonstration: Jupyter basics



```
In [1]: import ipystata
        from IPython.display import Image

In [ ]: words = ['But', 'also', 'live', 'code']
        for i in words:
            print(i)

##
$$Scaled\ change\ in\ Earnings_t: \frac{(Earnings_t-Earnings_{t-1})}{MV_{t-2}}$$
##

In [ ]: Image(r'Example/Images/table1a.png', width=700)
```

[Click here to play](#)

What is IPyStata?

A wrapper to execute Stata code in a Jupyter Notebook environment

Seamlessly use both Python and Stata
in one environment

→ Interchangeable data structures
using Pandas DataFrames

Stata do-file / log-file on steroids

- Input + Output in one place
 - (plots are supported)
- Format / Headers / Comments
 - Markdown / Images / Latex equations
- Run multiple Stata sessions at the same time from one notebook
- Easy sharing of code + results
 - export 'static' version of the notebook
- Works well with version control
 - GitHub renders notebooks!



Demonstration Notebook

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

[Click here to play](#)



Move data around between Stata and Python

Moving data around is very easy using the ``-d`` (`--data``) or ``-o`` (`--output``) arguments

```
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
```

[Click here to play](#)



```
##  
  
In [ ]:  
  
In [ ]:  
  
In [ ]:  
  
In [ ]:  
  
In [ ]:  
  
In [ ]:  
  
In [ ]:  
  
In [ ]:  
  
In [ ]:  
  
In [ ]:  
  
In [ ]:  
  
In [ ]:
```

[Click here to play](#)



Interchange lists and macros

```
In [ ]: |
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
In [ ]:
```

[Click here to play](#)



_cons | -5163.323 5512.845 -0.94 0.353 -16183.34 5856.697

It is also possible to retrieve a Stata macro and convert it to a Python list

```
In [17]: %%stata -gm numbers  
        local numbers 1 3 5
```

Several (1x) macros have been added to the dictionary: macro_dict

```
In [18]: macro_dict['numbers']
```

Out[18]: ['1', '3', '5']

Interact with Stata

In []:

In []:

In []:

In []:

In []:

In []:

[Click here to play](#)



Multiple instances of Stata

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

[Click here to play](#)

Use IPyStata to make offline materials

- Jupyter Notebooks can easily be converted into offline HTML documents
- These HTML documents can be easily edited for teaching and demonstration purposes
 - Converting to PDF works well

Estimate HEALY variable

This model assumes that the non-discretionary accrual part depends on the accruals of last year.

$$NDA_t = TA_{t-1}$$

In [10]:

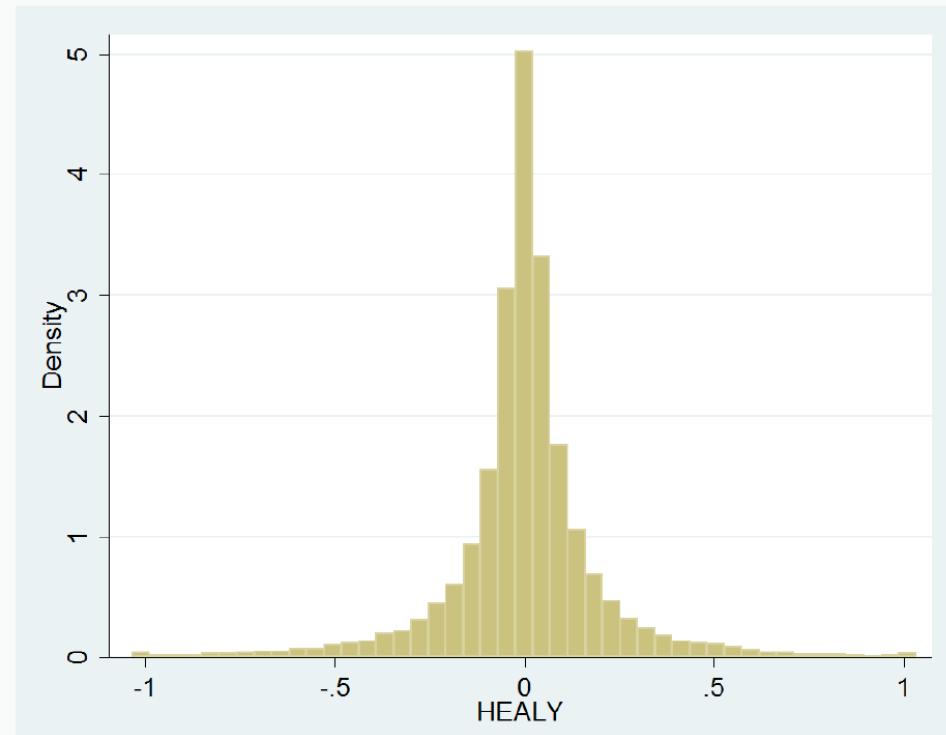
```
gen HEALY=WTA-lag_WTA
```

In [11]:

```
histogram HEALY
```

```
(bin=45, start=-1.0332508, width=.04592226)
```

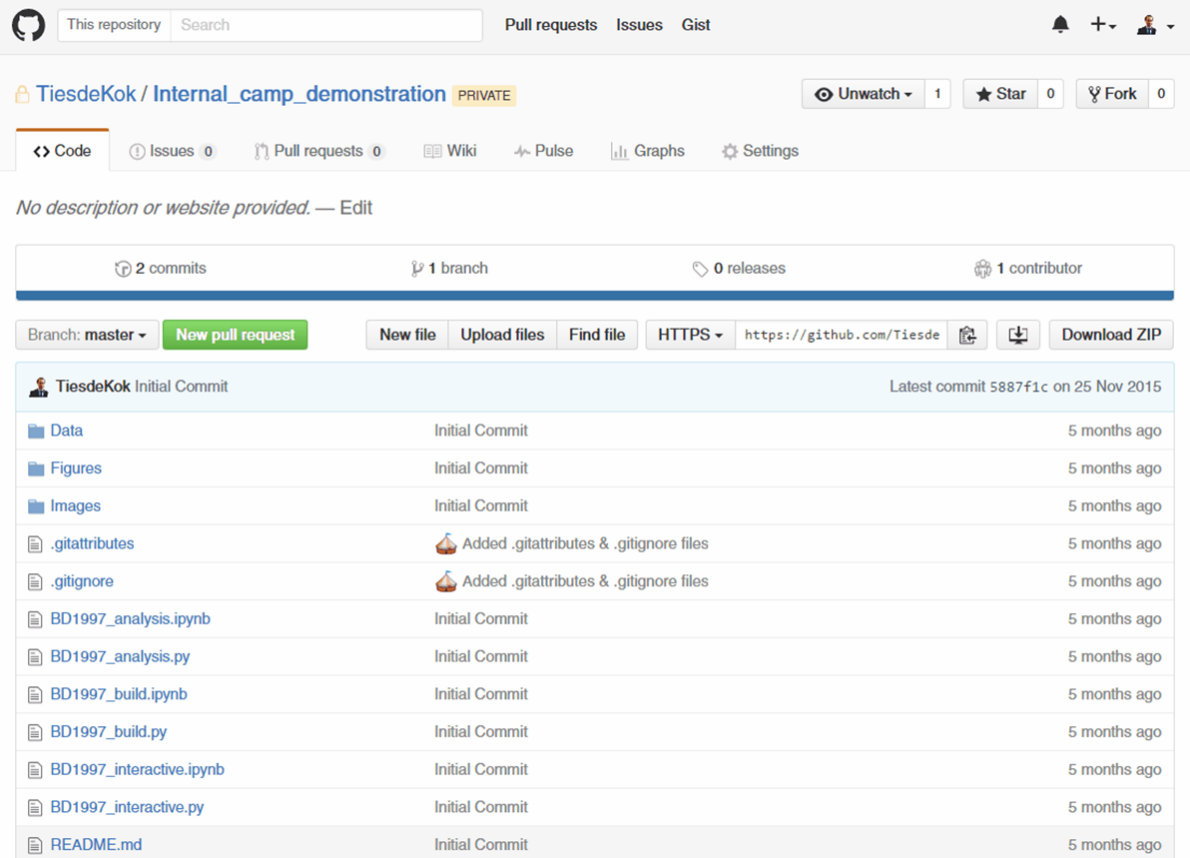
Out[11]:



Jupyter Notebook + Version Control

- Officially supported by GitHub

Code, results, comments, etc.
all under version control!



The screenshot shows a GitHub repository page for 'TiesdeKok / Internal_camp_demonstration'. The repository is private and has 1 contributor, 2 commits, 1 branch, and 0 releases. The commit history table is as follows:

File	Commit Message	Time
Data	Initial Commit	5 months ago
Figures	Initial Commit	5 months ago
Images	Initial Commit	5 months ago
.gitattributes	Added .gitattributes & .gitignore files	5 months ago
.gitignore	Added .gitattributes & .gitignore files	5 months ago
BD1997_analysis.ipynb	Initial Commit	5 months ago
BD1997_analysis.py	Initial Commit	5 months ago
BD1997_build.ipynb	Initial Commit	5 months ago
BD1997_build.py	Initial Commit	5 months ago
BD1997_interactive.ipynb	Initial Commit	5 months ago
BD1997_interactive.py	Initial Commit	5 months ago
README.md	Initial Commit	5 months ago

[Click here to play](#)

How to start using IPyStata?

- Check out my “getting started” guide on the IPyStata GitHub:

<https://git.io/vKrff>

- IPyStata GitHub page:

<https://github.com/TiesdeKok/ipystata>

- Official Jupyter website:

<http://www.jupyter.org/>

- 1) Install the Anaconda distribution
- 2) Install IPyStata
- 3) Configure IPyStata

Questions? Contact me!

- GitHub/TiesdeKok
- t.c.j.dekok@uvt.nl