

# Tools for Reproducible Research

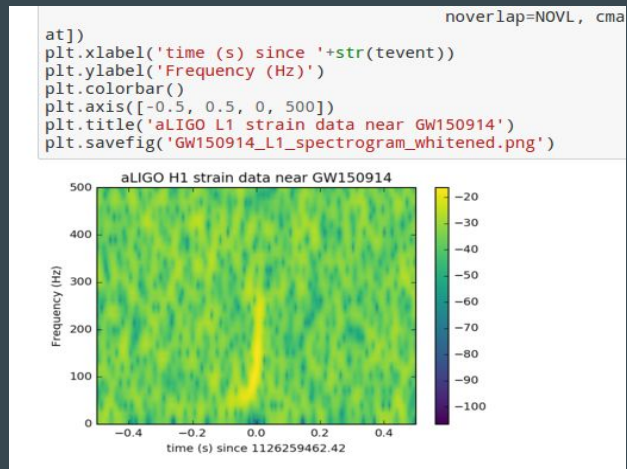
...

Valentina Staneva

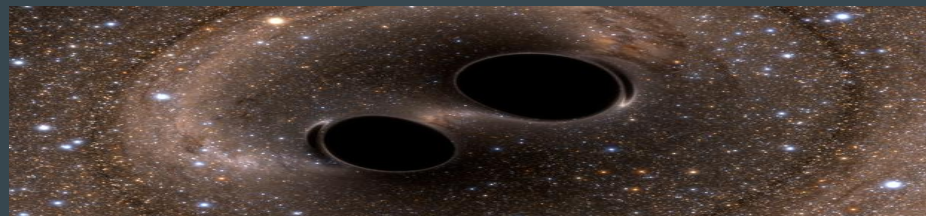
Data Science Summit, University of Washington, 2018

# LIGO experiment

Jupyter Notebooks analyzing the data:



[https://losc.ligo.org/s/events/GW150914/GW150914\\_tutorial.html](https://losc.ligo.org/s/events/GW150914/GW150914_tutorial.html)



Is the experiment reproducible?

Quora

Ask or Search Quora

Ask Question

Detection of Gravitational Waves (February 2016)

+2



**Gravitational waves discovery - is LIGO experiment reproducible or is it just a lucky timing that it caught a signal?**

It took so many years for LIGO to detect the waves - is it because the instrumentation improved a lot recently? Will this enable us to detect these waves every day or will it require signals from massive black hole collisions?

[general relativity - Why didn't LIGO wait for a second observation of a ... physics.stackexchange.com/.../246611](https://physics.stackexchange.com/.../246611) Stack Exchange ▾

Apr 1, 2016 - My whole life I have been taught that the very hallmark of scientific experiment are reproducible results. So why didn't LIGO wait for a second ...

Second Gravitational Wave Detected!

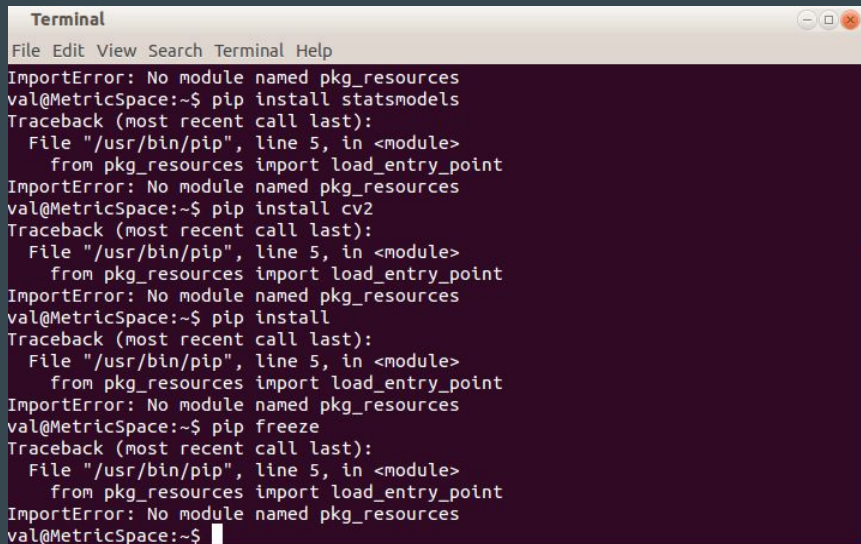
# Reproducibility vs. Replicability



Two main notions:

- Results of an experiment are regenerated using the same data and methods.
- Results of an experiment are regenerated using new data or alternative methods.

# It is hard...

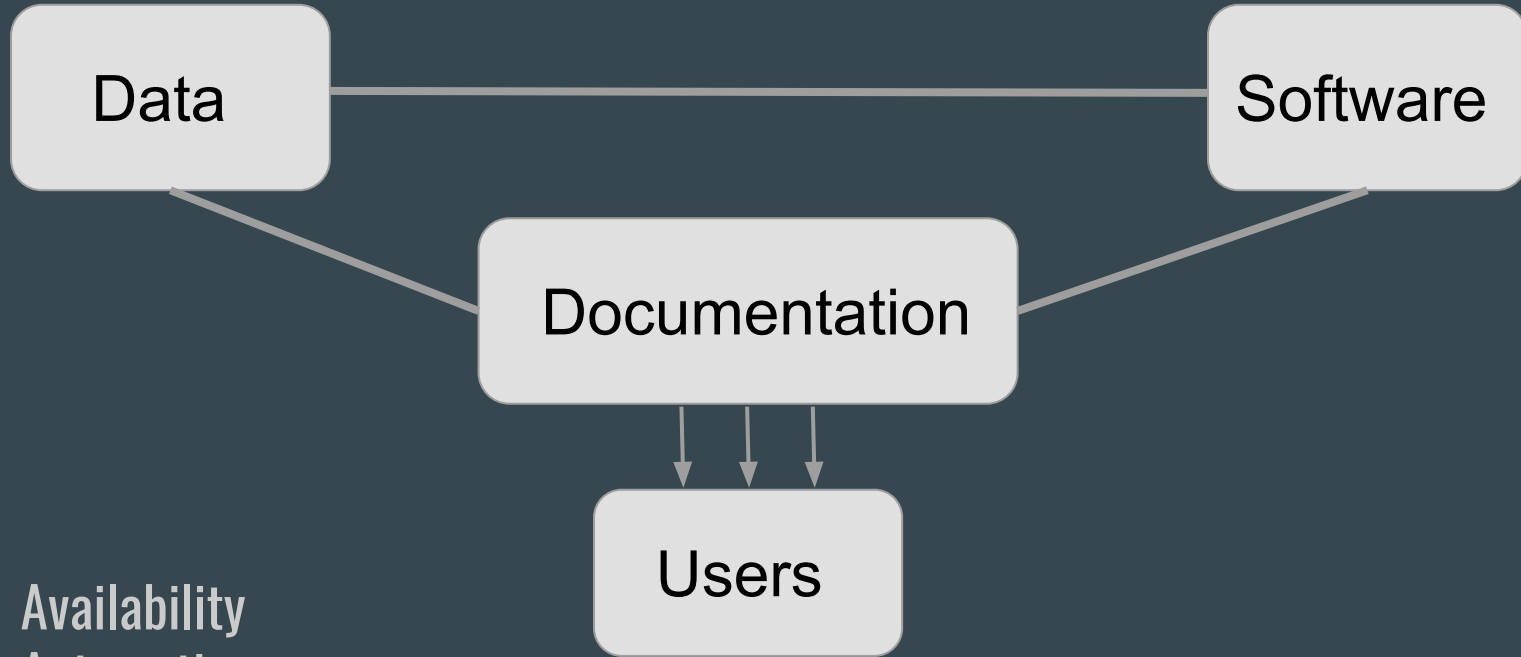
A terminal window titled "Terminal" with a menu bar (File, Edit, View, Search, Terminal, Help) and standard window controls. The terminal output shows a series of failed pip commands. Each time a command is run, it results in an "ImportError: No module named pkg\_resources" followed by a "Traceback (most recent call last):" block showing the error path through the pip script. The commands shown are: "pip install statsmodels", "pip install cv2", and "pip install". The prompt is "val@MetricSpace:~\$".

```
Terminal
File Edit View Search Terminal Help
ImportError: No module named pkg_resources
val@MetricSpace:~$ pip install statsmodels
Traceback (most recent call last):
  File "/usr/bin/pip", line 5, in <module>
    from pkg_resources import load_entry_point
ImportError: No module named pkg_resources
val@MetricSpace:~$ pip install cv2
Traceback (most recent call last):
  File "/usr/bin/pip", line 5, in <module>
    from pkg_resources import load_entry_point
ImportError: No module named pkg_resources
val@MetricSpace:~$ pip install
Traceback (most recent call last):
  File "/usr/bin/pip", line 5, in <module>
    from pkg_resources import load_entry_point
ImportError: No module named pkg_resources
val@MetricSpace:~$ pip freeze
Traceback (most recent call last):
  File "/usr/bin/pip", line 5, in <module>
    from pkg_resources import load_entry_point
ImportError: No module named pkg_resources
val@MetricSpace:~$
```

It is not about reproducible or not reproducible.

It is about **more reproducible**.

# Improving Reproducibility



- Availability
- Automation
- Sustainability

# Project Templates

- Python Module Template: [Shablona](#)
- R Project Structure: <https://nicercode.github.io/blog/2013-04-05-projects/>
- Data Science Project Structure: [Cookiecutter](#)

```
.
├── AUTHORS.md
├── LICENSE
├── README.md
├── bin                <- Your compiled model code can be stored here (not tracked by git)
├── config             <- Configuration files, e.g., for doxygen or for your model if needed
├── data
│   ├── external      <- Data from third party sources.
│   ├── interim       <- Intermediate data that has been transformed.
│   ├── processed     <- The final, canonical data sets for modeling.
│   └── raw           <- The original, immutable data dump.
├── docs               <- Documentation, e.g., doxygen or scientific papers (not tracked by git)
├── notebooks          <- Ipython or R notebooks
├── reports            <- For a manuscript source, e.g., LaTeX, Markdown, etc., or any project reports
│   └── figures        <- Figures for the manuscript or reports
├── src                <- Source code for this project
│   ├── data           <- scripts and programs to process data
│   ├── external       <- Any external source code, e.g., pull other git projects, or external libraries
│   ├── models         <- Source code for your own model
│   ├── tools          <- Any helper scripts go here
│   └── visualization  <- Scripts for visualisation of your results, e.g., matplotlib, ggplot2 related.
```

# Choose a license for your code

*Code without a license is protected by the author's copyright law.*

Choose a license website: <http://choosealicense.com/>

- Permissible licenses: MIT, BDS
- Copyleft licenses: GPL

# Documentation

Python - [Sphinx](#), [Read the Docs](#)



R - [Vignettes](#)

## dplyr: A Grammar of Data Manipulation

A fast, consistent tool for working with data frame like objects, both in memory and out of memory.

Version: 0.7.4  
Depends: R (≥ 3.1.2)  
Imports: [assertthat](#), [bindrcpp](#) (≥ 0.2), [glue](#) (≥ 1.1.1), [magrittr](#), methods, [pkgconfig](#), [rlang](#) (≥ 0.1.2), [R6](#), [Rcpp](#) (≥ 0.12.7), [tibble](#) (≥ 1.3.1), utils  
LinkingTo: [Rcpp](#) (≥ 0.12.0), [BH](#) (≥ 1.58.0-1), [bindrcpp](#), [plogr](#)  
Suggests: [bit64](#), [covr](#), [dbplyr](#), [dplyr](#), [DBI](#), [ggplot2](#), [hms](#), [knitr](#), [Lahman](#) (≥ 3.0-1), [mgcv](#), [microbenchmark](#), [nycflights13](#), [rmarkdown](#), [RMySQL](#), [RPostgreSQL](#), [RSQLite](#), [testthat](#), [withr](#)  
Published: 2017-09-28  
Author: Hadley Wickham [aut, cre], Romain Francois [aut], Lionel Henry [aut], Kirill Müller [aut], RStudio [cph, fnd]  
Maintainer: Hadley Wickham <hadley at rstudio.com>  
BugReports: <https://github.com/tidyverse/dplyr/issues>  
License: MIT + file LICENSE  
URL: <http://dplyr.tidyverse.org>, <https://github.com/tidyverse/dplyr>  
NeedsCompilation: yes  
Materials: [README NEWS](#)  
In views: [ModelDeployment](#)  
CRAN checks: [dplyr results](#)

- [Journal of Open Source Software](#)
- [Journal of Statistical Software](#)



# Literate Programming

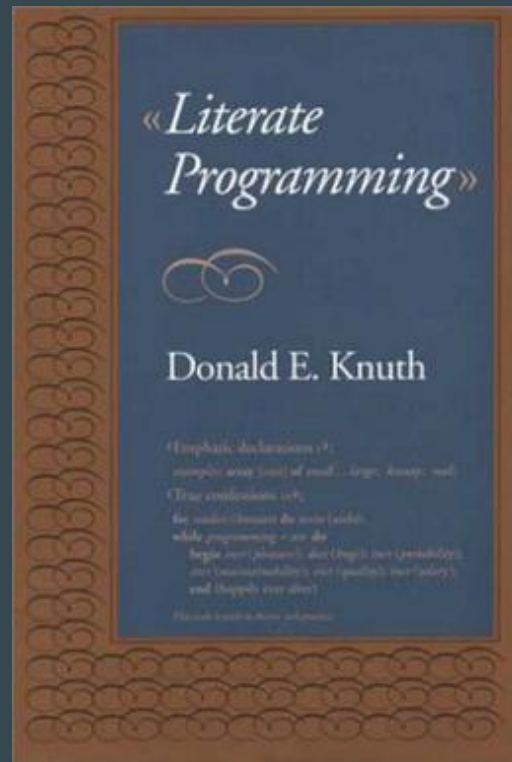
Combining documentation and code in a single program.

*“Instead of imagining that our main task is to instruct a computer what to do, let us concentrate rather on explaining to human beings what we want a computer to do.”*

R Reporting and sharing: [Knitr](#), [RPubs](#)

Notebooks - [Jupyter](#), [R Notebooks](#), [Zeppelin](#), [Sage](#), [Beaker](#)

Notebook Environments: [Binder](#), [CoCalc](#), [Colaboratory](#)

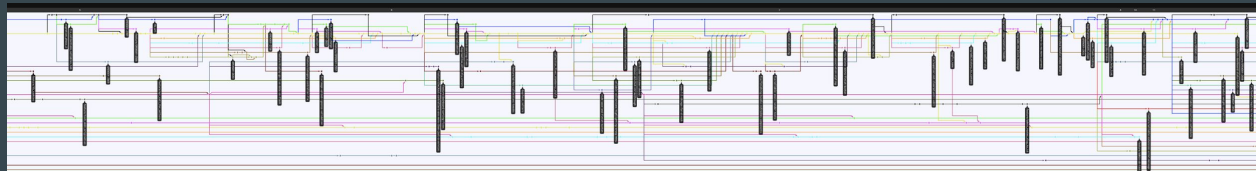
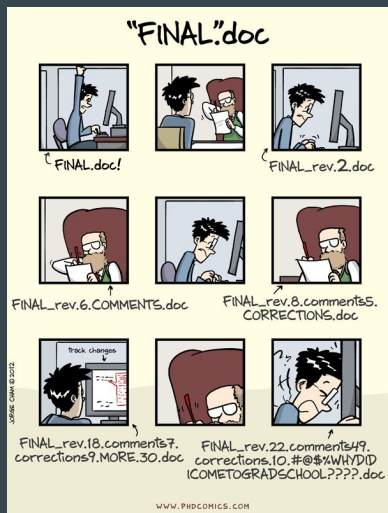


[Image by Wikipedia](#)

# Virtualization

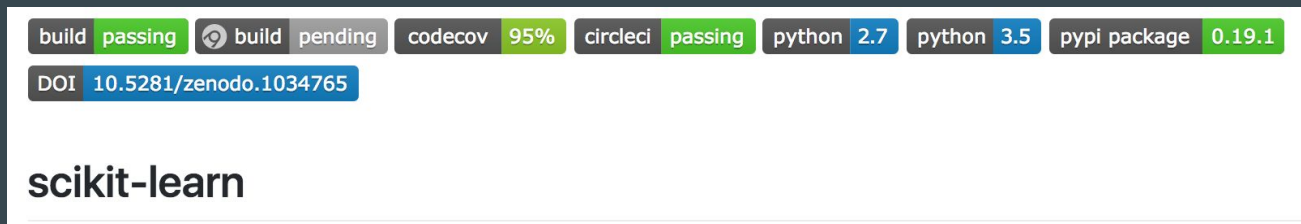
- Virtual Environments (Conda) - package dependencies
- Docker Containers - Linux environment, works on all OS, scriptable
- Vagrant - virtual machine manager which can run Docker containers and full VMs
- Virtual Machines - VirtualBox, VMWare
- Cloud Images

# Versioning



- Version control for code: git & Github
- Version control for data: <https://quiltdata.com/>

# Testing



The screenshot shows the top of the scikit-learn GitHub repository. It features a row of status badges: 'build passing' (green), 'build pending' (grey), 'codecov 95%' (green), 'circleci passing' (green), 'python 2.7' (blue), 'python 3.5' (blue), and 'pypi package 0.19.1' (green). Below these is a DOI badge: 'DOI 10.5281/zenodo.1034765'. The project name 'scikit-learn' is displayed in a large, bold font.

build passing build pending codecov 95% circleci passing python 2.7 python 3.5 pypi package 0.19.1

DOI 10.5281/zenodo.1034765

**scikit-learn**

*We are already writing tests, need to save them.*

- Locally
  - Python - unittest, nose, pytest
  - R - testthat
- Remotely - Continuous Integration
  - Travis, CircleCI, AppVeyor

Start by testing the environment.

# Data Repositories

	 <b>figshare</b>	
Up to 50GB free Not-for-profit - EU funded	100GB free per manuscript Institutional plans For-profit	Publishing Fee - \$120 Excess fees after 20GB Associated with articles Not-for-profit

- Datasets receive DOI.
- Cloud Storage: free to upload, fees to download
- Nature Journal Scientific Data: <https://www.nature.com/sdata/>

# Get Feedback

eScience Office Hours: <http://escience.washington.edu/office-hours/>

Reproducibility Mailing List: [escience\\_reproducibility](#)

# References

Reproducibility vs Replicability: or is it the other way around -

<http://languagelog ldc.upenn.edu/nll/?p=21956>

Chris Drummond's Interpretation - <https://core.ac.uk/download/files/21/107703.pdf>