
Agenda

Boost your reproducibility with Binder

- 13:30 Registration and introductions
- 13:40 Introduction to the workshop and The Turing Way
- 13:50 Presentation: Why you need a reproducible computing environment and how Binder can help
- 15:00 Coffee break
- 15:30 Code along demo: Zero to Binder, build a Binder resource
- 16:30 Build your own Binder
- 16:50 Feedback, group picture and close

#TuringWay @kirstie_j @mybinderteam

<https://doi.org/10.5281/zenodo.3632913>

The
Alan Turing
Institute

Reproducible Computational Environments

Kirstie Whitaker

Pronouns: she/her

#TuringWay @kirstie_j @mybinderteam
<https://doi.org/10.5281/zenodo.3632913>



The science is the code

An article about computational science in a scientific publication is not the scholarship itself, it is merely advertising of the scholarship. The actual scholarship is the complete software development environment and the complete set of instructions which generated the figures.

Buckheit and Donoho

(paraphrasing John Claerbout)

WaveLab and Reproducible Research, 1995

Upsetting take home message

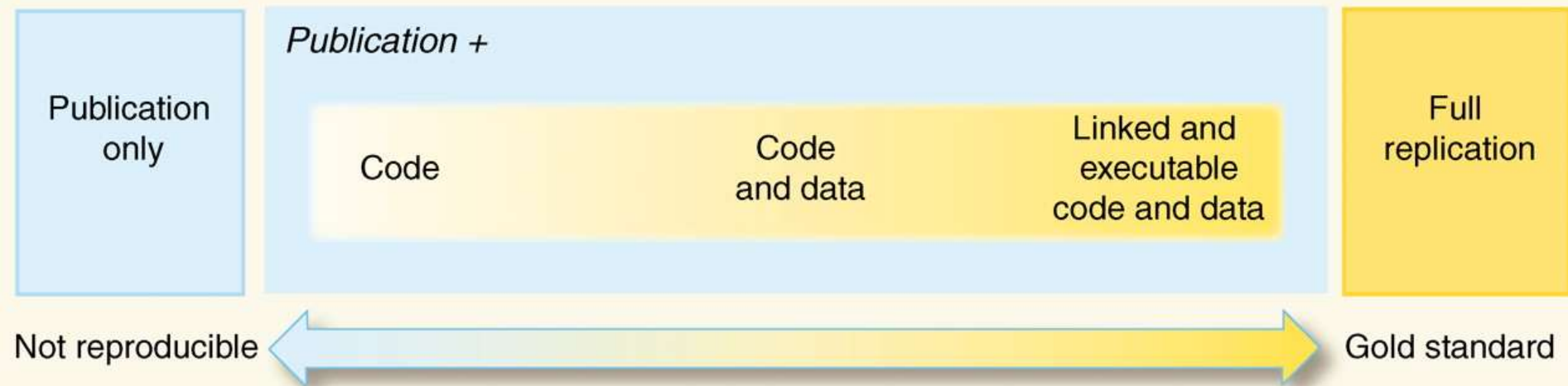
Sharing your code and
data isn't enough



#TuringWay @kirstie_j @mybinderteam
<https://doi.org/10.5281/zenodo.3632913>

You need the computational environment too

Reproducibility Spectrum



#TuringWay @kirstie_j @mybinderteam
<https://doi.org/10.5281/zenodo.3632913>

You need the computational environment too

Reproducibility Spectrum



#TuringWay @kirstie_j @mybinderteam
<https://doi.org/10.5281/zenodo.3632913>

The computational environment includes:

- Hardware (GPU, CPU)
- Operating system (mac, windows, linux)
- Software
 - Language version
 - Package version(s)

**And all the interactions
between the layers**



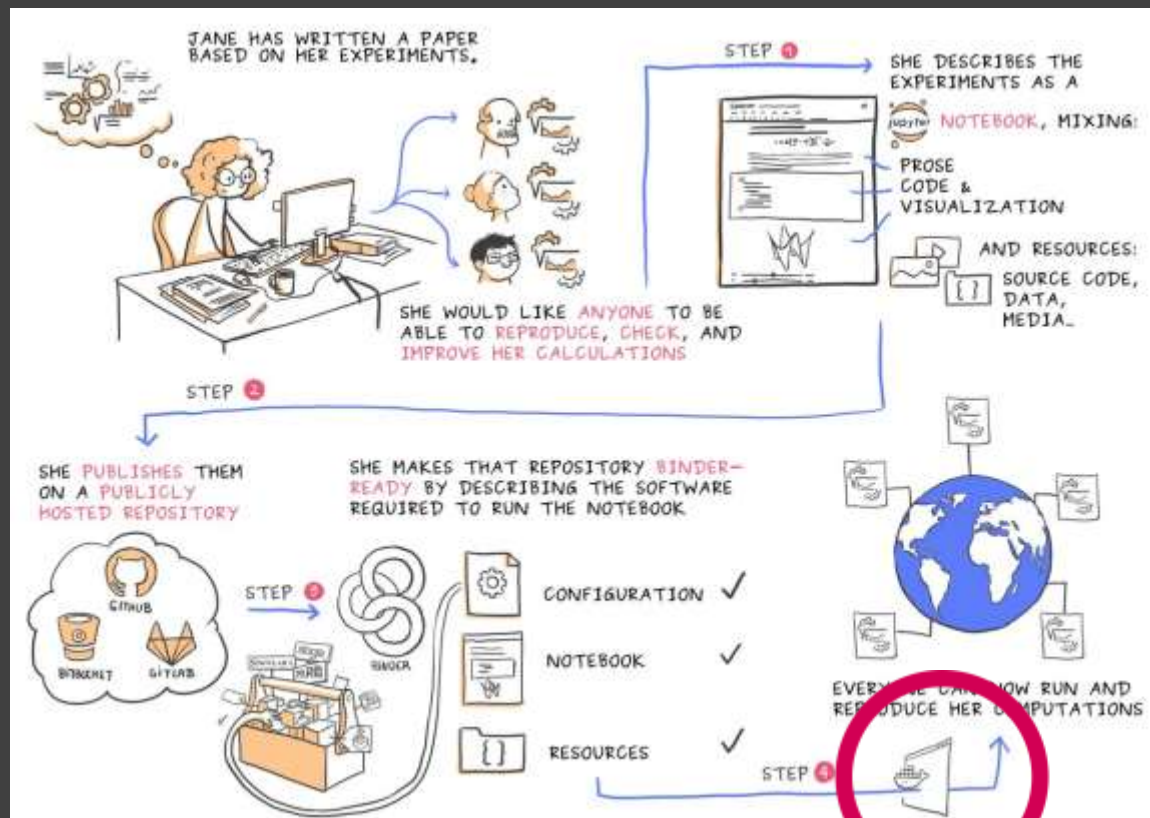
What is Binder?



#TuringWay @kirstie_j @mybinderteam
<https://doi.org/10.5281/zenodo.3632913>

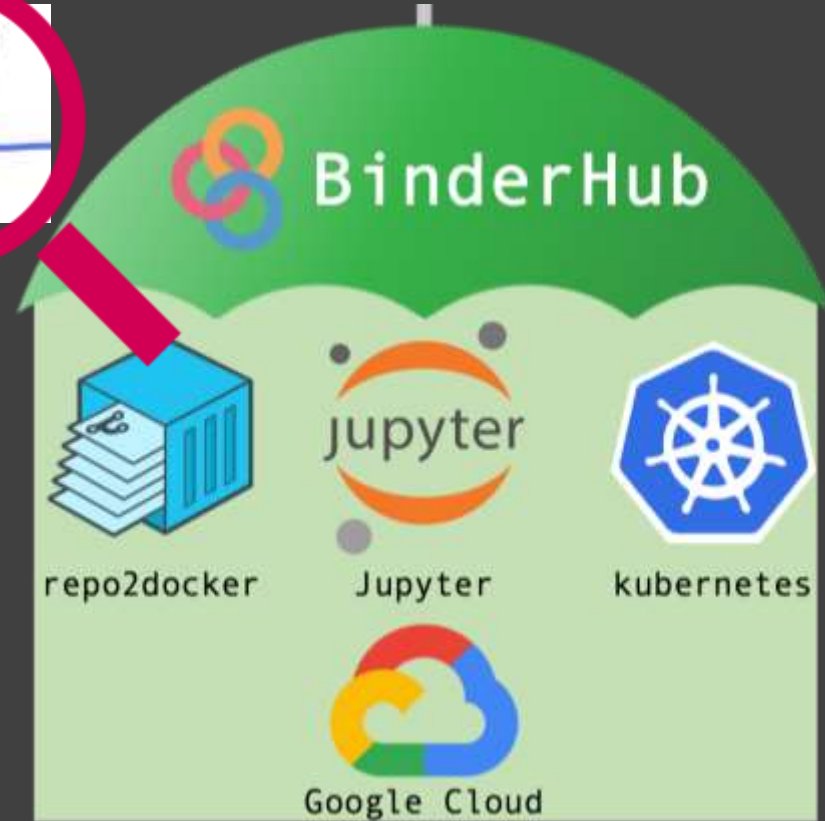


Courtesy of Juliette Taka: <https://twitter.com/mybinderteam/status/1082556317842264064>
#TuringWay @kirstie_j @mybinderteam
<https://doi.org/10.5281/zenodo.3632913>



Courtesy of Juliette Taka: <https://twitter.com/mybinder-techn/status/1082556317842264064>
#TuringWay @kirstie_j @mybinderteam
<https://doi.org/10.5281/zenodo.3632913>

- Coordinate cloud computing resources with Kubernetes (k8s)
- Make it easy for users to access with a JupyterHub
- Set up the environment from your GitHub repository



<https://binderhub.readthedocs.io>
#TuringWay @kirstie_j @mybinderteam
<https://doi.org/10.5281/zenodo.3632913>

Sarah Gibson

“It took me a while to feel like I knew enough to contribute to Binder. But the team are always so excited to have my input. Its really motivating to be part of such a welcoming community.”



<https://www.turing.ac.uk/people/researchers/sarah-gibson>
#TuringWay @kirstie_j @mybinderteam
<https://doi.org/10.5281/zenodo.3632913>

- Check analysis on my phone
- Share the responsibility with busy PIs
- Requires version control, capturing environment and new build for each change



#TuringWay @kirstie_j @mybinderteam
<https://doi.org/10.5281/zenodo.3632913>



Table of Contents

Getting started with Binder

Getting started with Binder
Common usage patterns in Binder

How to...

Choose languages for your environment
Configure the user interface
Generate custom launch badges for your Binder repository
Track repository data on mybinder.org
Change launch link

What is mybinder.org?

mybinder.org is a single deployment of a BinderHub instance, managed by the Binder community. It serves as both a public service and a demonstration of the BinderHub technology, though it is by no means the only BinderHub in existence. If you're interested in deploying your own BinderHub for your own uses, please see the [BinderHub documentation](#) and don't hesitate to reach out to the [Binder community](#).

For more information, check out [About mybinder.org](#).

Is mybinder.org free to use?

Yes! Though note that it has relatively [limited computational resources](#).

How much does running mybinder.org cost?

Great question! If you're interested in the technical costs of running **mybinder.org**, we publish a semi-up-to-date dataset of our costs at the [binder-data](#) repository. In addition, you can explore these costs with the binder link below!



How can mybinder.org be free to use?

On this page

What is a Binder?
What is the Binder community?
What is BinderHub?
What is **mybinder.org**?
Is **mybinder.org** free to use?
How much does running **mybinder.org** cost?
How can **mybinder.org** be free to use?
How much memory am I given when using Binder?
How long will my Binder session last?
Can I use mybinder.org for a live demo or workshop?
How does mybinder.org ensure user privacy?
How secure is mybinder.org?
Where can I report a security issue?
Can I push data from my Binder session back to my repository?
Can I put my configuration files outside the root of  **v: latest**?
What factors influence how long it takes a Binder session to start?
Will repos with fewer notebooks launch faster? Should I split my

<https://mybinder.readthedocs.io/en/latest/faq.html#how-much-does-running-mybinder-org-cost>
#TuringWay @kirstie_j @mybinderteam
<https://doi.org/10.5281/zenodo.3632913>

Thanks to Google Cloud and OVH for sponsoring our computers 🐛



Starting repository: jupyterhub/binder-billing/master

New to Binder? Check out the [Binder Documentation](#) for more information.

Build logs

show

Here's a non-interactive preview on [nbviewer](#) while we start a server for you. Your binder will open automatically when it is ready.



JUPYTER

FAQ



binder-billing

<https://mybinder.readthedocs.io/en/latest/faq.html#how-much-does-running-mybinder-org-cost>
#TuringWay @kirstie_j @mybinderteam
<https://doi.org/10.5281/zenodo.3632913>

Thanks to Google Cloud and OVH for sponsoring our computers 🐛



Starting repository: jupyterhub/binder-billing/master

New to Binder? Check out the [Binder Documentation](#) for more information.

Build logs

show

Here's a non-interactive preview on [nbviewer](#) while we start a server for you. Your binder will open automatically when it is ready.



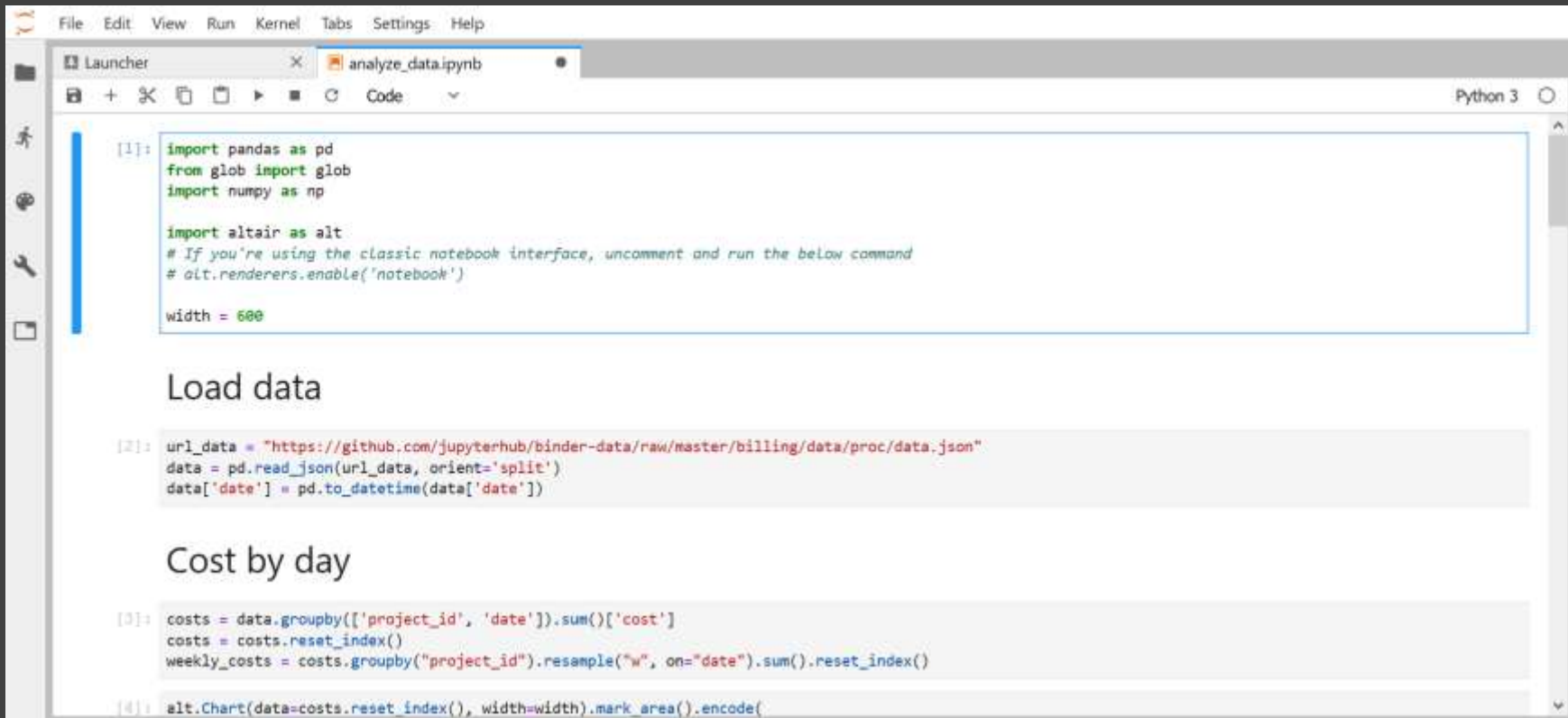
JUPYTER

FAQ



binder-billing

<https://mybinder.readthedocs.io/en/latest/faq.html#how-much-does-running-mybinder-org-cost>
#TuringWay @kirstie_j @mybinderteam
<https://doi.org/10.5281/zenodo.3632913>



The image shows a Jupyter Notebook window titled 'analyze_data.ipynb'. The interface includes a top menu bar with 'File', 'Edit', 'View', 'Run', 'Kernel', 'Tabs', 'Settings', and 'Help'. Below the menu is a toolbar with icons for file operations and execution. The notebook content is divided into cells. The first cell contains import statements for pandas, glob, numpy, and altair, along with a comment about enabling altair's notebook renderer and a variable 'width' set to 600. The second cell is titled 'Load data' and contains code to read a JSON file from a GitHub URL. The third cell is titled 'Cost by day' and contains code to group data by project ID and date, calculate costs, and create a weekly cost series. The fourth cell contains code to create an altair chart from the weekly costs data.

```
[1]: import pandas as pd
from glob import glob
import numpy as np

import altair as alt
# If you're using the classic notebook interface, uncomment and run the below command
# alt.renderers.enable('notebook')

width = 600
```

Load data

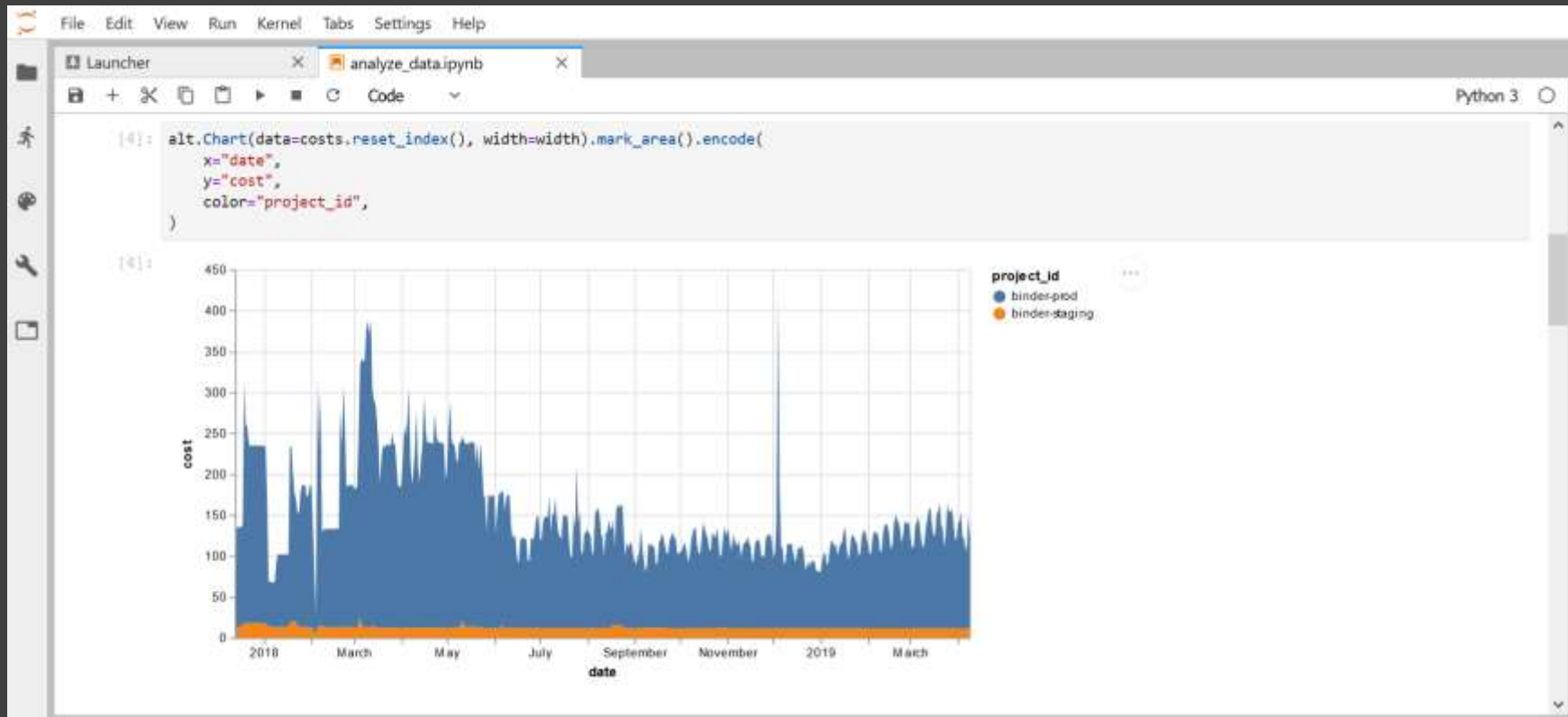
```
[2]: url_data = "https://github.com/jupyterhub/binder-data/raw/master/billing/data/proc/data.json"
data = pd.read_json(url_data, orient='split')
data['date'] = pd.to_datetime(data['date'])
```

Cost by day

```
[3]: costs = data.groupby(['project_id', 'date']).sum()['cost']
costs = costs.reset_index()
weekly_costs = costs.groupby("project_id").resample("w", on="date").sum().reset_index()

[4]: alt.Chart(data=costs.reset_index(), width=width).mark_area().encode(
```

<https://mybinder.readthedocs.io/en/latest/faq.html#how-much-does-running-mybinder-org-cost>
#TuringWay @kirstie_j @mybinderteam
<https://doi.org/10.5281/zenodo.3632913>



<https://mybinder.readthedocs.io/en/latest/faq.html#how-much-does-running-mybinder-org-cost>
#TuringWay @kirstie_j @mybinderteam
<https://doi.org/10.5281/zenodo.3632913>

Gertjan van den Burg

“The fun part of data science is the modelling. Being able to read in information from a csv file should not be the hardest part.”



<https://gertjanvandenburger.com>
#TuringWay @kirstie_j @mybinderteam
<https://doi.org/10.5281/zenodo.3632913>

alan-turing-institute / CleverCSVDemo

Unwatch 0 Star 0 Fork 1

Code Issues 0 Pull requests 0 Actions Projects 0 Wiki Insights

No description, website, or topics provided.

23 commits 1 branch 0 releases 1 contributor MIT

branches: master • New pull request

Create new file Upload files Find file Clone or download +

GjrdBurg add more examples and clarify · Latest commit 1304aaf 4 days ago

| | | |
|--------------------------------------------|----------------------------------------------------|-------------|
| data | add more examples and clarify | 4 days ago |
| images | add qr code with link to repo | 12 days ago |
| CSV_dialect_detection_with_CleverCSV.ipynb | add more examples and clarify | 4 days ago |
| CSV_dialect_detection_with_CleverCSV.md | add more examples and clarify | 4 days ago |
| LICENSE | Add makefile and create the notebook from Markdown | 7 days ago |
| Makefile | Add makefile and create the notebook from Markdown | 7 days ago |
| README.md | Add binder thingy to Readme | 13 days ago |
| requirements.txt | add termcolor dependency | 6 days ago |

README.md

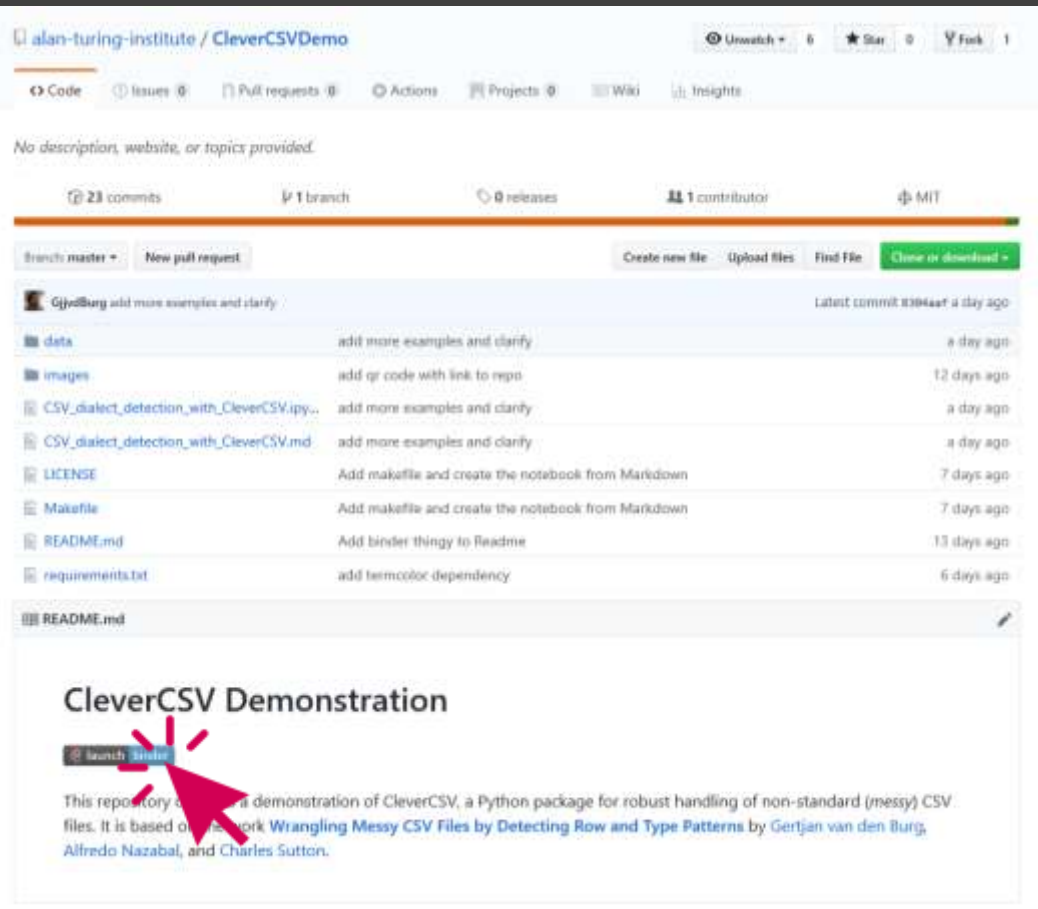
CleverCSV Demonstration

[launch](#) [binder](#)

This repository contains a demonstration of CleverCSV, a Python package for robust handling of non-standard (messy) CSV files. It is based on the work [Wrangling Messy CSV Files by Detecting Row and Type Patterns](#) by Gertjan van den Burg, Alfredo Nazabal, and Charles Sutton.

– <https://github.com/alan-turing-institute/CleverCSVDemo>

#TuringWay @kirstie_j @mybinderteam
<https://doi.org/10.5281/zenodo.3632913>



– <https://github.com/alan-turing-institute/CleverCSVDemo>

– “Wrangling Messy CSV Files by Detecting Row and Type Patterns”
arXiv:1811.11242

#TuringWay @kirstie_j @mybinderteam
<https://doi.org/10.5281/zenodo.3632913>

CSV dialect detection with CleverCSV

Author: [Gertjan van den Burg](#)

In this note we'll show some examples of using CleverCSV, a package for handling messy CSV files. We'll start with a motivating example and then show some other files where CleverCSV shines. CleverCSV was developed as part of a research project on automating data wrangling. It achieves an accuracy of 97% on over 9300 real-world CSV files and improves the accuracy on messy files by 21% over standard tools.

Handy links:

- [Paper on arXiv](#)
- [CleverCSV on GitHub](#)
- [CleverCSV on PyPI](#)
- [Reproducible Research Repo](#)

IMDB Movie data

Alice is a data scientist who would like to analyse the movie ratings on IMDB for movies of different genres. She found [a dataset shared by a user on Kaggle](#) that contains information of over 14,000 movies. Great!

The data is stored in a CSV file, which is a very common data format for sharing tabular data. The first few lines of the file look like this:

<https://github.com/alan-turing-institute/CleverCSVDemo>
#TuringWay @kirstie_j @mybinderteam
<https://doi.org/10.5281/zenodo.3632913>

IMDB Movie data

Alice is a data scientist who would like to analyse the movie ratings on IMDB for movies of different genres. She found [a dataset shared by a user on Kaggle](#) that contains information of over 14,000 movies. Great!

The data is stored in a CSV file, which is a very common data format for sharing tabular data. The first few lines of the file look like this:

```
fn,tid,title,wordsInTitle,url,imdbRating,ratingCount,duration,year,type,nrOfWins,nrOfNominations,nrOfPhotos,nrOf
NewsArticles,nrOfUserReviews,nrOfGenre>Action,Adult,Adventure,Animation,Biography,Comedy,Crime,Documentary,Drama
,Family,Fantasy,FilmNoir,GameShow,History,Horror,Music,Musical,Mystery,News,RealityTV,Romance,SciFi,Short,Sport,
TalkShow,Thriller,War,Western
titles01/tt0012349,tt0012349,Der Vagabund und das Kind (1921),der vagabund und das kind,http://www.imdb.com/titl
e/tt0012349/,8.4,40550,3240,1921,video.movie,1,0,19,96,85,3,0,0,0,0,0,1,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0
titles01/tt0015864,tt0015864,Goldrausch (1925),goldrausch,http://www.imdb.com/title/tt0015864/,8.3,45319,5700,19
25,video.movie,2,1,35,110,122,3,0,0,1,0,0,1,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
titles01/tt0017136,tt0017136,Metropolis (1927),metropolis,http://www.imdb.com/title/tt0017136/,8.4,81007,9180,19
27,video.movie,3,4,67,428,376,2,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0
titles01/tt0017925,tt0017925,Der General (1926),der general,http://www.imdb.com/title/tt0017925/,8.3,37521,6420,
1926,video.movie,1,1,53,123,219,3,1,0,1,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
titles01/tt0021749,tt0021749,Lichter der Großstadt (1931),lichter der gro stadt,http://www.imdb.com/title/tt0021
749/,8.7,70057,5220,1931,video.movie,2,0,38,187,186,3,0,0,0,0,0,0,1,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0
```

Seems pretty standard, let's load it with Pandas!

In [1]: %xmode Minimal

```
In [1]: %xmode Minimal
import pandas as pd
df = pd.read_csv('./data/imdb.csv')
```

Exception reporting mode: Minimal

ParserError: Error tokenizing data. C error: Expected 44 fields in line 66, saw 46

Oh, that doesn't work. Maybe there's something wrong with the file? Let's try opening it with the Python CSV reader:

```
In [2]: import csv
with open('./data/imdb.csv', 'r', newline='') as fid:
    dialect = csv.Sniffer().sniff(fid.read())
    print("Detected delimiter = %r, quotechar = %r" % (dialect.delimiter, dialect.quotechar))
    fid.seek(0)
    reader = csv.reader(fid, dialect=dialect)
    rows = list(reader)

print("Loaded %i rows." % len(rows))
```

Detected delimiter = ' ', quotechar = '"'
Loaded 13928 rows.

Huh, that's strange, Python thinks the space is the delimiter and loads 13928 rows, but the file should contain 14,762 rows according to the documentation. What's going on here?

Huh, that's strange, Python thinks the space is the delimiter and loads 13928 rows, but the file should contain 14,762 rows according to the documentation. What's going on here?

It turns out that on the 65th line of the file, there's a movie with the title `Dr. Seltsam\, oder wie ich lernte\, die Bombe zu lieben (1964)` (the German version of Dr. Strangelove). The title has commas in it, that are escaped using the `\` character! Why are CSV files so hard? 😞

CleverCSV to the rescue!

CleverCSV detects the dialect of CSV files much more accurately than existing approaches, and it is therefore robust against these kinds of format variations. It even has a wrapper that works with DataFrames!

```
In [3]: from csv.wrappers import csv2df
df = csv2df('./data/imdb.csv')
df
```

Out [3]:

| | fn | tid | title | wordsInTitle | url | imdbRating | ratingCount | duration | year | type | ... | News |
|---|--------------------|-----------|----------------------------------|---------------------------|--------------------------------------|------------|-------------|----------|--------|-------------|-----|------|
| 0 | titles01/tt0012349 | tt0012349 | Der Vagabund und das Kind (1921) | der vagabund und das kind | http://www.imdb.com/title/tt0012349/ | 8.4 | 40550.0 | 3240.0 | 1921.0 | video.movie | ... | 0 |
| 1 | titles01/tt0015864 | tt0015864 | Goldrausch (1925) | goldrausch | http://www.imdb.com/title/tt0015864/ | 8.3 | 45319.0 | 5700.0 | 1925.0 | video.movie | ... | 0 |
| 2 | titles01/tt0017136 | tt0017136 | Metropolis (1927) | metropolis | http://www.imdb.com/title/tt0017136/ | 8.4 | 81007.0 | 9180.0 | 1927.0 | video.movie | ... | 0 |
| 3 | titles01/tt0017925 | tt0017925 | Der General (1926) | der general | http://www.imdb.com/title/tt0017925/ | 8.3 | 37521.0 | 6420.0 | 1926.0 | video.movie | ... | 0 |
| | | | Lichter der | lichter der gro | http://www.imdb.com | | | | | | | |

Huh, that's strange, Python thinks the space is the delimiter and loads 13928 rows, but the file should contain 14,762 rows according to the documentation. What's going on here?

It turns out that on the 65th line of the file, there's a movie with the title `Dr. Seltsam\, oder wie ich lernte\, die Bombe zu lieben` (German version of Dr. Strangelove). The title has commas in it, that are escaped using the `\` character! Why are CSV files so hard? 😞

CleverCSV to the rescue!

CleverCSV detects the dialect of CSV files much more accurately than existing approaches, and it is therefore robust against these kinds of f... even has a wrapper that works with DataFrames!

```
In [3]: from csv.wrappers import csv2df

df = csv2df('./data/imdb.csv')
df
```

Out [3]:

| | fn | tid | title | wordsInTitle | url | imdbRating | ratingCount | duration | year | type | ... | News |
|---|--------------------|-----------|----------------------------------|---------------------------|--------------------------------------|------------|-------------|----------|--------|-------------|-----|------|
| 0 | titles01/tt0012349 | tt0012349 | Der Vagabund und das Kind (1921) | der vagabund und das kind | http://www.imdb.com/title/tt0012349/ | 8.4 | 40550.0 | 3240.0 | 1921.0 | video.movie | ... | 0 |
| 1 | titles01/tt0015864 | tt0015864 | Goldrausch (1925) | goldrausch | http://www.imdb.com/title/tt0015864/ | 8.3 | 45319.0 | 5700.0 | 1925.0 | video.movie | ... | 0 |
| 2 | titles01/tt0017136 | tt0017136 | Metropolis (1927) | metropolis | http://www.imdb.com/title/tt0017136/ | 8.4 | 81007.0 | 9180.0 | 1927.0 | video.movie | ... | 0 |
| 3 | titles01/tt0017925 | tt0017925 | Der General (1926) | der general | http://www.imdb.com/title/tt0017925/ | 8.3 | 37521.0 | 6420.0 | 1926.0 | video.movie | ... | 0 |
| | | | Lichter der | lichter der gro | http://www.imdb.com | | | | | | | |



| File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 | | | | | | | | | | | | | |
|-----------------------------------------------------------------|--------------------------|-----------|---------------------------------------------------|---------------------------------------------------|--------------------------------------|-----|------|--------|--------|---------------|-----|---|--|
| + - Copy Paste Run Markdown | | | | | | | | | | | | | |
| df | | | | | | | | | | | | | |
| | | | Episode 2005) | episode | /title/tt0675644/ | | | | | | | | |
| 14757 | titles04/index.html.9992 | tt0675644 | "Playhouse 90" The Miracle Worker (TV Episode ... | playhouse the miracle worker tv episode | http://www.imdb.com/title/tt0675644/ | 7.3 | 8.0 | 5400.0 | 1957.0 | video.episode | ... | 0 | |
| 14758 | titles04/index.html.9994 | tt0679222 | "Private Screenings" Robert Mitchum and Jane R... | private screenings robert mitchum and jane rus... | http://www.imdb.com/title/tt0679222/ | 7.0 | 20.0 | 3600.0 | 1996.0 | video.episode | ... | 0 | |
| 14759 | titles04/index.html.9995 | tt0680064 | "Providence" All the King's Men (TV Episode 2002) | providence all the king s men tv episode | http://www.imdb.com/title/tt0680064/ | NaN | NaN | 3600.0 | 2002.0 | video.episode | ... | 0 | |
| 14760 | titles04/index.html.9997 | tt0681024 | "QI" Adam (TV Episode 2003) | qi adam tv episode | http://www.imdb.com/title/tt0681024/ | 7.6 | 89.0 | 1800.0 | 2003.0 | video.episode | ... | 0 | |

14761 rows x 44 columns

Hooray! 🎉

How does it work? CleverCSV searches the space of all possible dialects of a file, and computes a *data consistency measure* that quantifies how much the resulting table "looks like real data". The consistency measure combines patterns of row lengths in the parsing result and the data type of the resulting cells. This mimicks how a human would identify the dialect. If you're wondering why this problem is hard, it's because every dialect will give you *some* table, but not necessarily the correct one. More details can be found [in the paper](#).

<https://github.com/alan-turing-institute/CleverCSVDemo>
 #TuringWay @kirstie_j @mybinderteam
<https://doi.org/10.5281/zenodo.3632913>

Defining your computational environment



#TuringWay @kirstie_j @mybinderteam
<https://doi.org/10.5281/zenodo.3632913>

pip freeze

- Python Package Index (PyPI)
- Pip is a recursive acronym that can stand for either "Pip Installs Packages" or "Pip Installs Python"

pip freeze

Contents

- pip freeze
 - Usage
 - Description
 - Options
 - Examples

Usage

```
pip freeze [options]
```

#TuringWay @kirstie_j @mybinderteam

<https://doi.org/10.5281/zenodo.3632913>

pip freeze

- `pip freeze` captures the versions of all packages that you're currently using
- Can print to screen or save in a file called `requirements.txt`

Examples

1. Generate output suitable for a requirements file.

```
$ pip freeze
docutils==0.11
Jinja2==2.7.2
MarkupSafe==0.19
Pygments==1.6
Sphinx==1.2.2
```

2. Generate a requirements file and then install from it in another

```
$ env1/bin/pip freeze > requirements.txt
$ env2/bin/pip install -r requirements.txt
```

Binder example: requirements.txt

Branch: master ▾

[requirements](#) / requirements.txt

Find file

Copy path



yuvipanda Bump numpy pin

a73ba12 16 days ago

[2 contributors](#)



4 lines (3 sloc) | 45 Bytes

Raw

Blame

History



```
1  numpy==1.16.*
2  matplotlib==3.*
3  seaborn==0.8.1
```

#TuringWay @kirstie_j @mybinderteam

<https://doi.org/10.5281/zenodo.3632913>

<https://github.com/binder-examples/requirements>

Binder example: requirements.txt

Branch: master ▾

[requirements](#) / requirements.txt

Find file

Copy path



yuvipanda Bump numpy pin

a73ba12 16 days ago

[2 contributors](#)



4 lines (3 sloc) | 45 Bytes

Raw

Blame

History



```
1  numpy==1.16.*
2  matplotlib==3.*
3  seaborn==0.8.1
```

#TuringWay @kirstie_j @mybinderteam

<https://doi.org/10.5281/zenodo.3632913>

<https://github.com/binder-examples/requirements>

Conda env create

- Package manager for multiple languages
- Information about installed software saved in file called `environment.yml`

<https://docs.conda.io/projects/conda/en/latest/user-guide/tasks/manage-environments.html>
https://the-turing-way.netlify.com/reproducible_environments/02/package-management.html

Creating an environment from an `environment.yml` file

Use the terminal or an Anaconda Prompt for the following steps:

1. Create the environment from the `environment.yml` file:

```
conda env create -f environment.yml
```

The first line of the `yml` file sets the new environment's name. For details see [Creating](#)

2. Activate the new environment: `conda activate myenv`
3. Verify that the new environment was installed correctly:

```
conda env list
```

You can also use `conda info --envs`.

#TuringWay @kirstie_j @mybinderteam
<https://doi.org/10.5281/zenodo.3632913>

Binder example: environment.yml

Branch: master ▾

python-conda_pip / environment.yml

Find file

Copy path



choldgraf adding a notebook

10ba338 on 23 Nov 2017

1 contributor

12 lines (11 sloc) | 165 Bytes

Raw

Blame

History



```
1 name: example-environment
2 channels:
3   - conda-forge
4 dependencies:
5   - python
6   - numpy
7   - pip:
8     - nbgitpuller
9     - sphinx-gallery
10    - pandas
11    - matplotlib
```

#TuringWay @kirstie_j @mybinderteam


<https://doi.org/10.5281/zenodo.3632913>

https://github.com/binder-examples/python-conda_pip

install.R

- Binder interface includes Rstudio
- `install.R` contains required packages
- `runtime.txt` sets the version on MRAN

Specifying an R environment with a `runtime.txt` file

Jupyter+R: 

RStudio: 

RShiny: 

Binder supports using R and RStudio, with libraries pinned to a specific snapshot on [MRAN](#).

You need to have a `runtime.txt` file that is formatted like:

```
R-<YYYY>-<MM>-<DD>
```

where YYYY-MM-DD is a snapshot at MRAN that will be used for installing libraries.

You can also have an `install.R` file that will be executed during build, and can be used to install additional packages.


Both [RStudio](#) and [IRKernel](#) are installed by default, so you can use either the Jupyter notebook or the RStudio interface.






This repository also contains an example of a Shiny app.

Binder example: install.R

Branch: master ▾ New pull request

Create new file Upload files Find file Clone or download ▾

 **gedankenstuecke** add readme & test notebook Latest commit d55e70d on 25 Sep 2018

| | | |
|------------------------------------------------------------------------------------------------------------|----------------------------|--------------|
|  R | first commit | 5 months ago |
|  DESCRIPTION | first commit | 5 months ago |
|  NAMESPACE | first commit | 5 months ago |
|  README.md | add readme & test notebook | 5 months ago |
|  test-library.ipynb | add readme & test notebook | 5 months ago |


#TuringWay @kirstie_j @mybinderteam


<https://doi.org/10.5281/zenodo.3632913>


<https://github.com/binder-examples/binder-r-description>


Binder example: install.R


Branch: master


 gedar

 R


 DESCRIPTION


 NAME

 README

 test-lit

Specifying an R environment by having a DESCRIPTION file

Jupyter+R: 

RStudio: 

Binder supports using R and RStudio, with libraries pinned to a specific snapshot on [MRAN](#).

If you specify a `runtime.txt` file that is formatted like:

```
r-<YYYY>-<MM>-<DD>
```

where YYYY-MM-DD it will use the MRAN snapshot of that day for setting up the R runtime.

Without specifying a `runtime.txt` it will use a 2-day old snapshot of MRAN.

Both [RStudio](#) and [IRKernel](#) are installed by default, so you can use either the Jupyter notebook interface or the RStudio interface.

Download

Sep 2018

months ago

months ago

months ago

months ago

months ago

#TuringWay @kirstie_j @mybinderteam

<https://doi.org/10.5281/zenodo.3632913>

<https://github.com/binder-examples/binder-r-description>

Becky Arnold

“There are a lot of things you need to know before you can jump into continuous integration.

Version control is a prerequisite for pretty much everything.”



<https://software.ac.uk/about/fellows/becky-arnold>

#TuringWay @kirstie_j @mybinderteam

<https://doi.org/10.5281/zenodo.3632913>

Distributed and remote version control



#TuringWay @kirstie_j @mybinderteam
<https://doi.org/10.5281/zenodo.3632913>

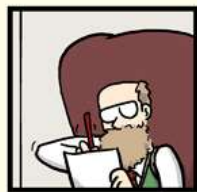
"FINAL".doc



FINAL.doc!



FINAL_rev.2.doc



FINAL_rev.6.COMMENTS.doc



FINAL_rev.8.comments5.
CORRECTIONS.doc



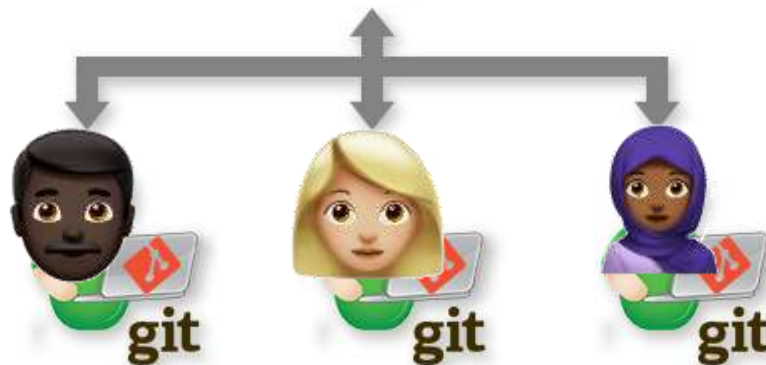
FINAL_rev.18.comments7.
corrections9.MORE.30.doc



FINAL_rev.22.comments49.
corrections.10.#@\$%WHYDID
ICOMETOGRADSCHOOL?????.doc



JORGE CHAN © 2012



<http://phdcomics.com/comics/archive.php?comid=1531>
#TuringWay @kirstie_j @mybinderteam
<https://doi.org/10.5281/zenodo.3632913>

The cloud is just someone else's computer



#TuringWay @kirstie_j @mybinderteam
<https://doi.org/10.5281/zenodo.3632913>

Our Data Centers

AWS pioneered cloud computing in 2006, creating cloud infrastructure that allows you to securely build and innovate faster. We are continuously innovating the design and systems of our data centers to protect them from man-made and natural risks. Then we implement controls, build automated systems, and undergo third-party audits to confirm security and compliance. As a result, the most highly-regulated organizations in the world trust AWS every day. Take a virtual tour of one of our data centers to learn about our security approach to protect the data of millions of active monthly customers.



#TuringWay @kirstie_j @mybinderteam

<https://doi.org/10.5281/zenodo.3632913>

<https://aws.amazon.com/compliance/data-center/data-centers>

Our Data Centers

AWS pioneered cloud computing in 2006, creating cloud infrastructure that allows you to securely build and innovate faster. We are continuously innovating the design and systems of our data centers to protect them from man-



PERIMETER LAYER

AWS data center physical security begins at the Perimeter Layer. This layer includes a number of security features depending on the location, such as security guards, fencing, security feeds, intrusion detection technology, and other security measures.

EXPLORE »



DATA LAYER

The Data Layer is the most critical point of protection because it is the only area that holds customer data. Protection begins by restricting access and maintaining a separation of privilege for each layer. In addition, we deploy threat detection devices and system protocols, further safeguarding this layer.

EXPLORE »



INFRASTRUCTURE LAYER

The Infrastructure Layer is the data center building and the equipment and systems that keep it running. Components like back-up power equipment, the HVAC system, and fire suppression equipment are all part of the Infrastructure Layer.

EXPLORE »



ENVIRONMENTAL LAYER

The Environmental Layer is dedicated to environmental considerations from site selection and construction to operations and sustainability. AWS carefully chooses our data center locations to mitigate environmental risk, such as flooding, extreme weather, and seismic activity.

EXPLORE »

#TuringWay @kirstie_j @mybinderteam

<https://doi.org/10.5281/zenodo.3632913>

<https://aws.amazon.com/compliance/data-center/data-centers>

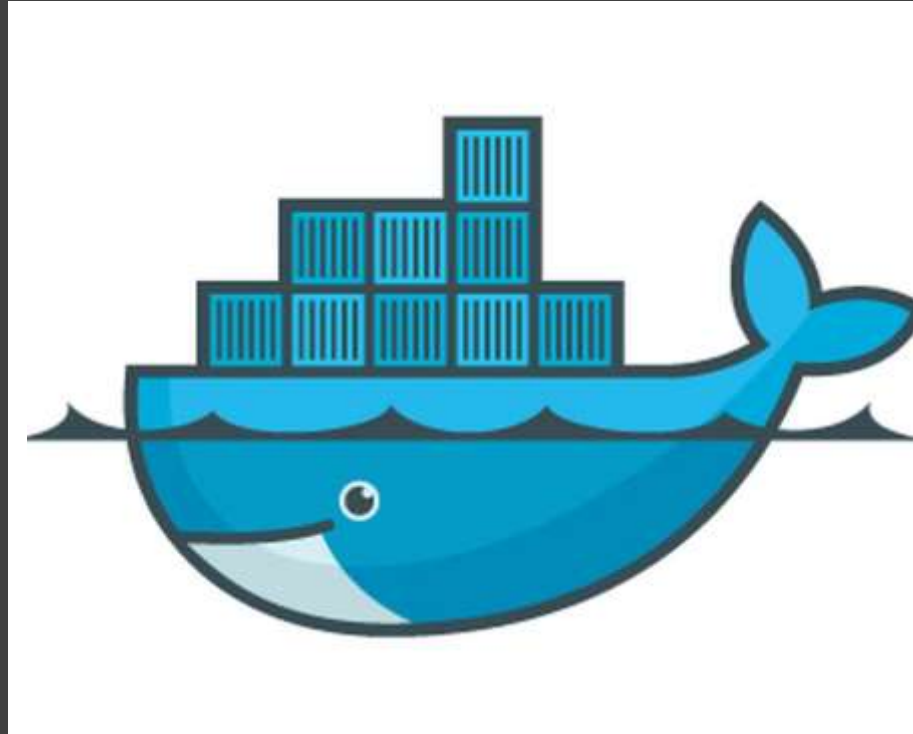
Containers



#TuringWay @kirstie_j @mybinderteam
<https://doi.org/10.5281/zenodo.3632913>

Docker

- A container that bundles all the infrastructure and software together.

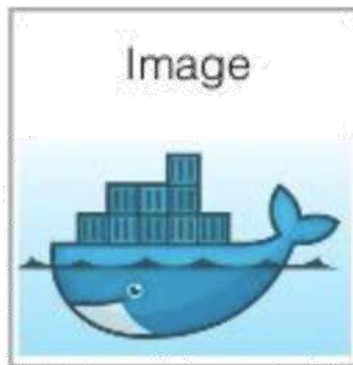


Human and machine readable files

```
FROM ubuntu:14.04
MAINTAINER kirstie.j@mybinderteam.com
RUN apt-get update && apt-get install -y python-pip python-dev python-setuptools
RUN pip install --user docker
RUN pip install --user docker-compose
FROM ubuntu:14.04
MAINTAINER kirstie.j@mybinderteam.com
RUN apt-get update && apt-get install -y python-pip python-dev python-setuptools
RUN pip install --user docker
RUN pip install --user docker-compose
```

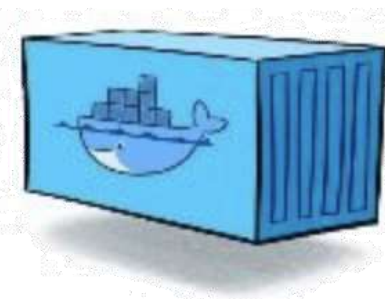
Dockerfile

build



Docker Image


run






Docker Container

Binder example: docker

Branch: master ▾ [minimal-dockerfile](#) / Dockerfile Find file Copy path

 minrk use 3.7-slim 9402b58 on 20 Sep 2018

[1 contributor](#)

17 lines (14 sloc) | 357 Bytes Raw Blame History   

```
1 FROM python:3.7-slim
2 # install the notebook package
3 RUN pip install --no-cache --upgrade pip && \
4     pip install --no-cache notebook
5
6 # create user with a home directory
7 ARG NB_USER
8 ARG NB_UID
9 ENV USER ${NB_USER}
10 ENV HOME /home/${NB_USER}
11
12 RUN adduser --disabled-password \
13     --gecos "Default user" \
14     --uid ${NB_UID} \
15     ${NB_USER}
16 WORKDIR ${HOME}
```

#TuringWay @kirstie_j @mybinderteam

<https://doi.org/10.5281/zenodo.3632913>

<https://github.com/binder-examples/binder-r-description>

Small group exercise



#TuringWay @kirstie_j @mybinderteam
<https://doi.org/10.5281/zenodo.3632913>

Small group exercise

Get into groups of 2-3 and explore these examples

- Are there differences between different branches?
- Does that give different results?
- Did you get what you'd expect?

[https://github.com/alan-turing-institute/
the-turing-way/blob/master/
workshops/
boost-research-reproducibility-binder/
paired_examples.md](https://github.com/alan-turing-institute/the-turing-way/blob/master/workshops/boost-research-reproducibility-binder/paired_examples.md)

#TuringWay @kirstie_j @mybinderteam
<https://doi.org/10.5281/zenodo.3632913>

Zero to Binder

Please open these instructions in your browser

[https://github.com/alan-turing-institute/
the-turing-way/blob/master/
workshops/
boost-research-reproducibility-binder/
workshop-presentations/
zero-to-binder.md](https://github.com/alan-turing-institute/the-turing-way/blob/master/workshops/boost-research-reproducibility-binder/workshop-presentations/zero-to-binder.md)

#TuringWay @kirstie_j @mybinderteam
<https://doi.org/10.5281/zenodo.3632913>

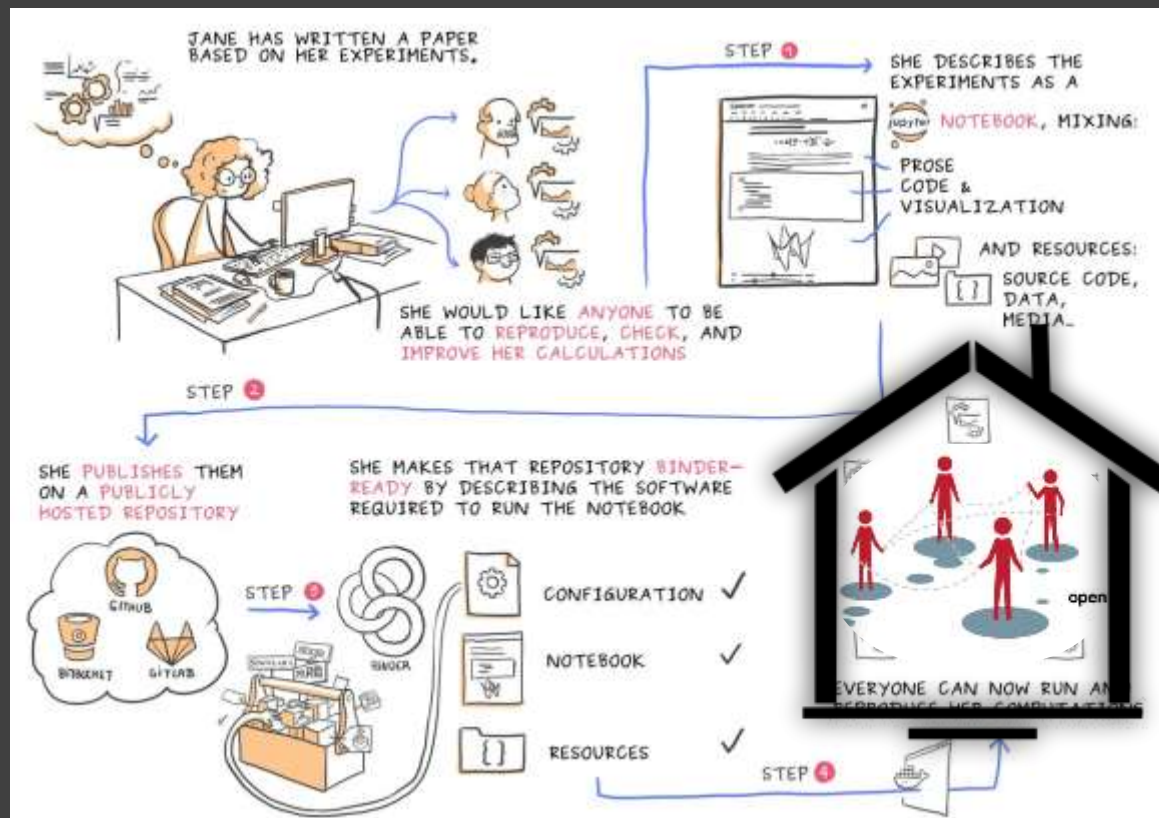
Agenda

Boost your reproducibility with Binder

- 13:30 Registration and introductions
- 13:40 Introduction to the workshop and The Turing Way
- 13:50 Presentation: Why you need a reproducible computing environment and how Binder can help
- 15:00 Coffee break
- 15:30 Code along demo: Zero to Binder, build a Binder resource
- 16:30 Build your own Binder
- 16:50 Feedback, group picture and close

#TuringWay @kirstie_j @mybinderteam

<https://doi.org/10.5281/zenodo.3632913>



Courtesy of Juliette Taka: <https://twitter.com/mybinderteam/status/1082556317842264064>

#PyDataLDN #TuringWay @kirstie_j

<https://doi.org/10.5281/zenodo.3632913>

The Alan Turing Institute



Loading repository (can take 30s or more to load): sgibson91/branchLSTM/sgibson91python-runtime-patch

New to Binder? Check out the [Binder Documentation](#) for more information.

Build logs

[show](#)

Here's a non-interactive preview on nbviewer while we start a server for you. Your binder will open automatically when it is ready.



JUPYTER

FAQ



branchLSTM

sgibson91python-runtime-patch

<https://github.com/kochkinaelena/branchLSTM> (on Turing Way Hub)

#PyDataLDN #TuringWay @kirstie_j

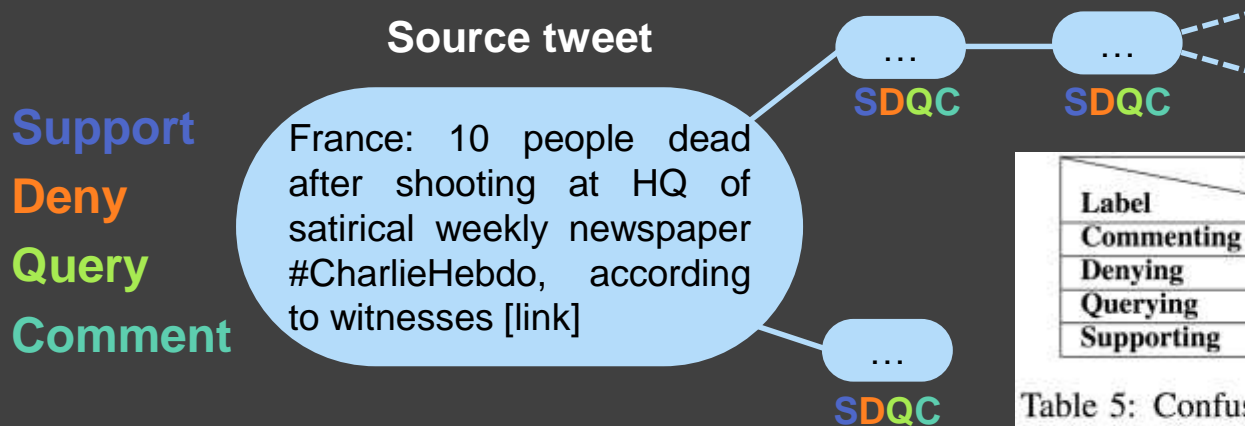
<https://doi.org/10.5281/zenodo.3632913>

Champion: Elena Kochkina



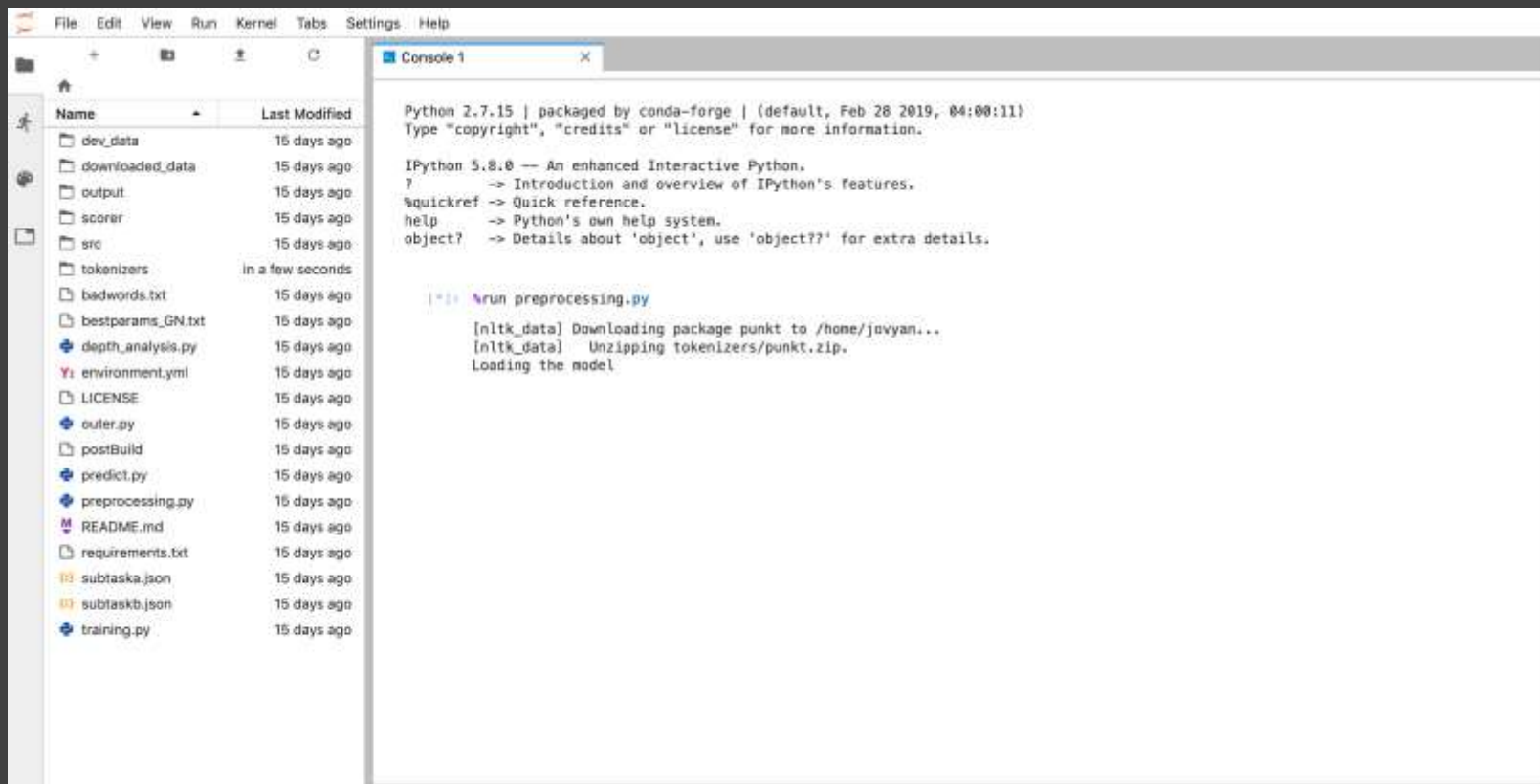
Turing at SemEval-2017 Task 8: Sequential Approach to Rumour Stance Classification with Branch-LSTM

Elena Kochkina, Maria Liakata, Isabelle Augenstein



| Label \ Prediction | Prediction | | | |
|--------------------|------------|---|----|----|
| | C | D | Q | S |
| Commenting | 760 | 0 | 12 | 6 |
| Denying | 68 | 0 | 1 | 2 |
| Querying | 69 | 0 | 36 | 1 |
| Supporting | 67 | 0 | 1 | 26 |

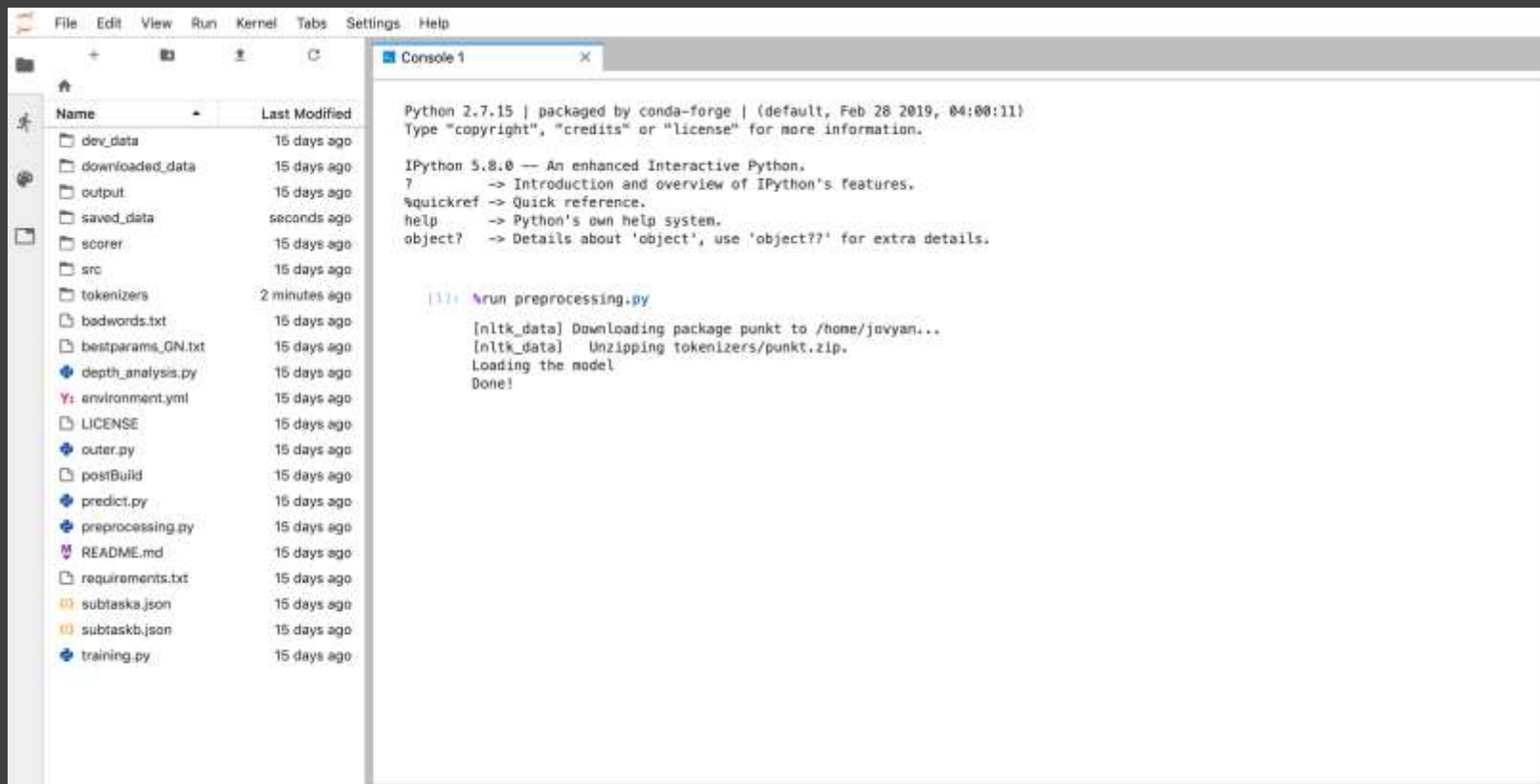
Table 5: Confusion matrix for testing set predictions



<https://github.com/kochkinaelena/branchLSTM> (on Turing Way Hub)

#PyDataLDN #TuringWay @kirstie_j

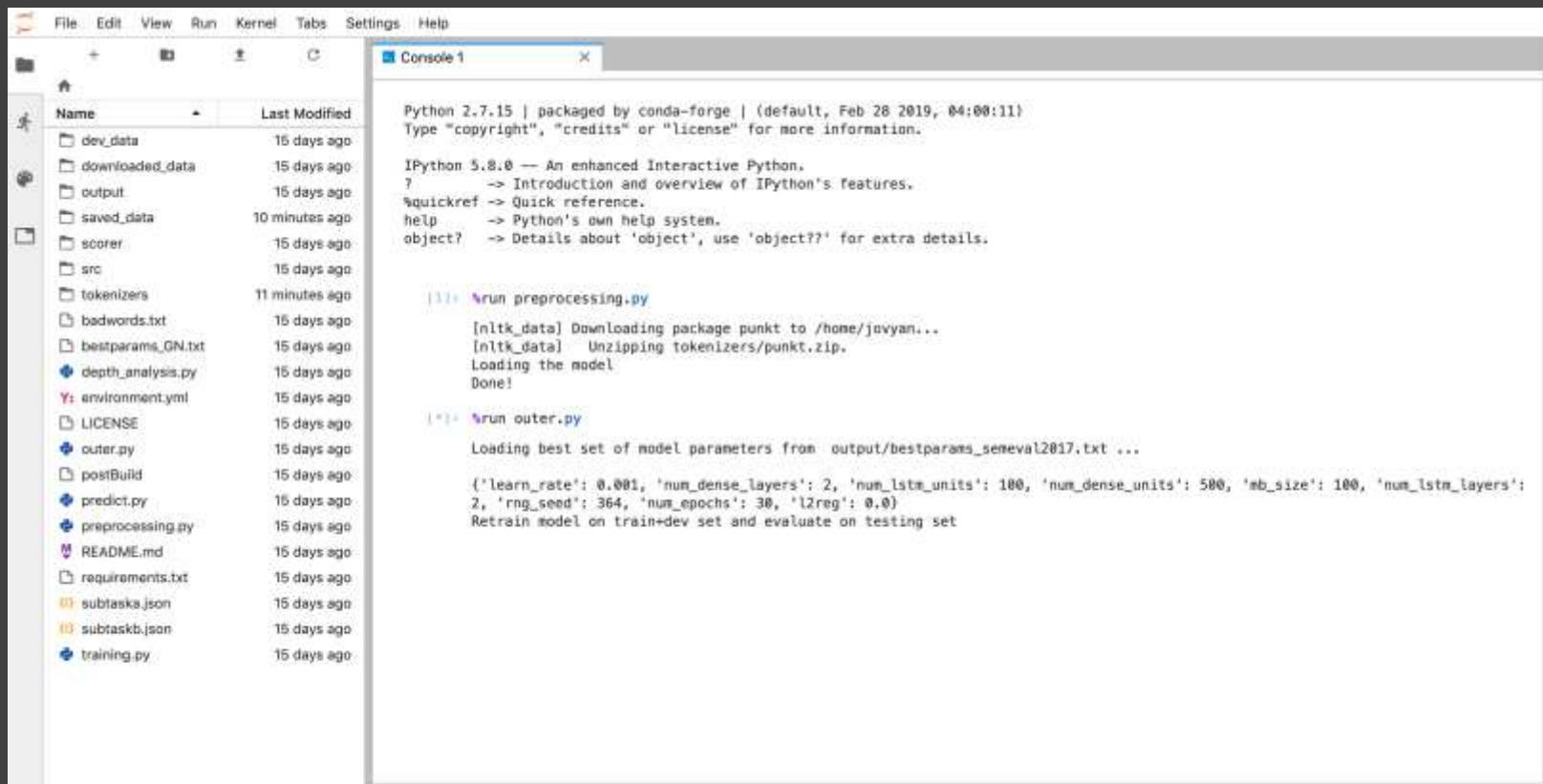
<https://doi.org/10.5281/zenodo.3632913>



<https://github.com/kochkinaelena/branchLSTM> (on Turing Way Hub)

#PyDataLDN #TuringWay @kirstie_j

<https://doi.org/10.5281/zenodo.3632913>



<https://github.com/kochkinaelena/branchLSTM> (on Turing Way Hub)

#PyDataLDN #TuringWay @kirstie_j

<https://doi.org/10.5281/zenodo.3632913>

Elena Kochkina

“How would I have known that it would be different on a different machine?! I only have access to the university HPC to run deep learning analyses.”



<https://warwick.ac.uk/fac/sci/dcs/people/research/mapmbc>
#PyDataLDN #TuringWay @kirstie_j
<https://doi.org/10.5281/zenodo.3632913>