

Building “Geographic Data Science...”

Universidad de Salta

Dani Arribas-Bel [@darribas]

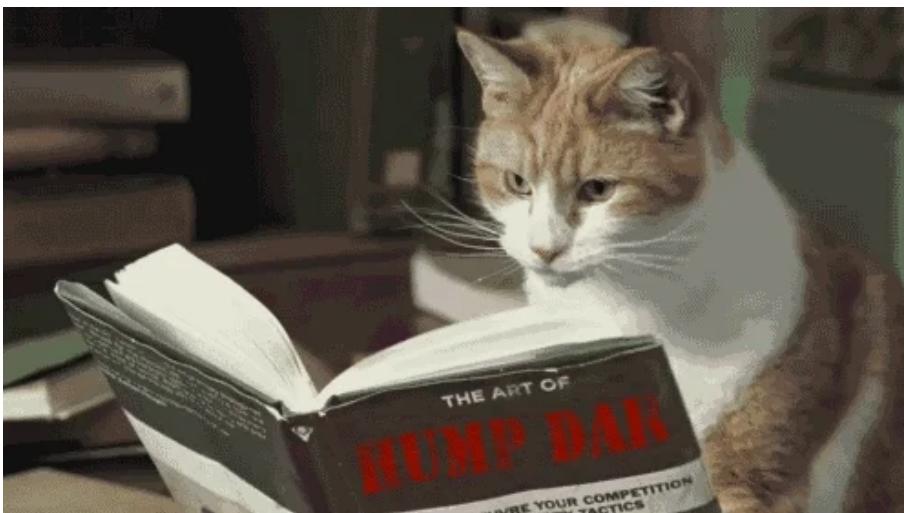


The
Alan Turing
Institute



Geographic
Data Science
Lab

We have a book!



via GIPHY

Almost...



I am this close

via GIPHY

Coming in 2021 but...

Coming in 2021 but...

... you can already:

- <https://geographicdata.science>
- <https://github.com/gdsbook/book>



The Authors



@sreyog

Serge Rey



@darribas

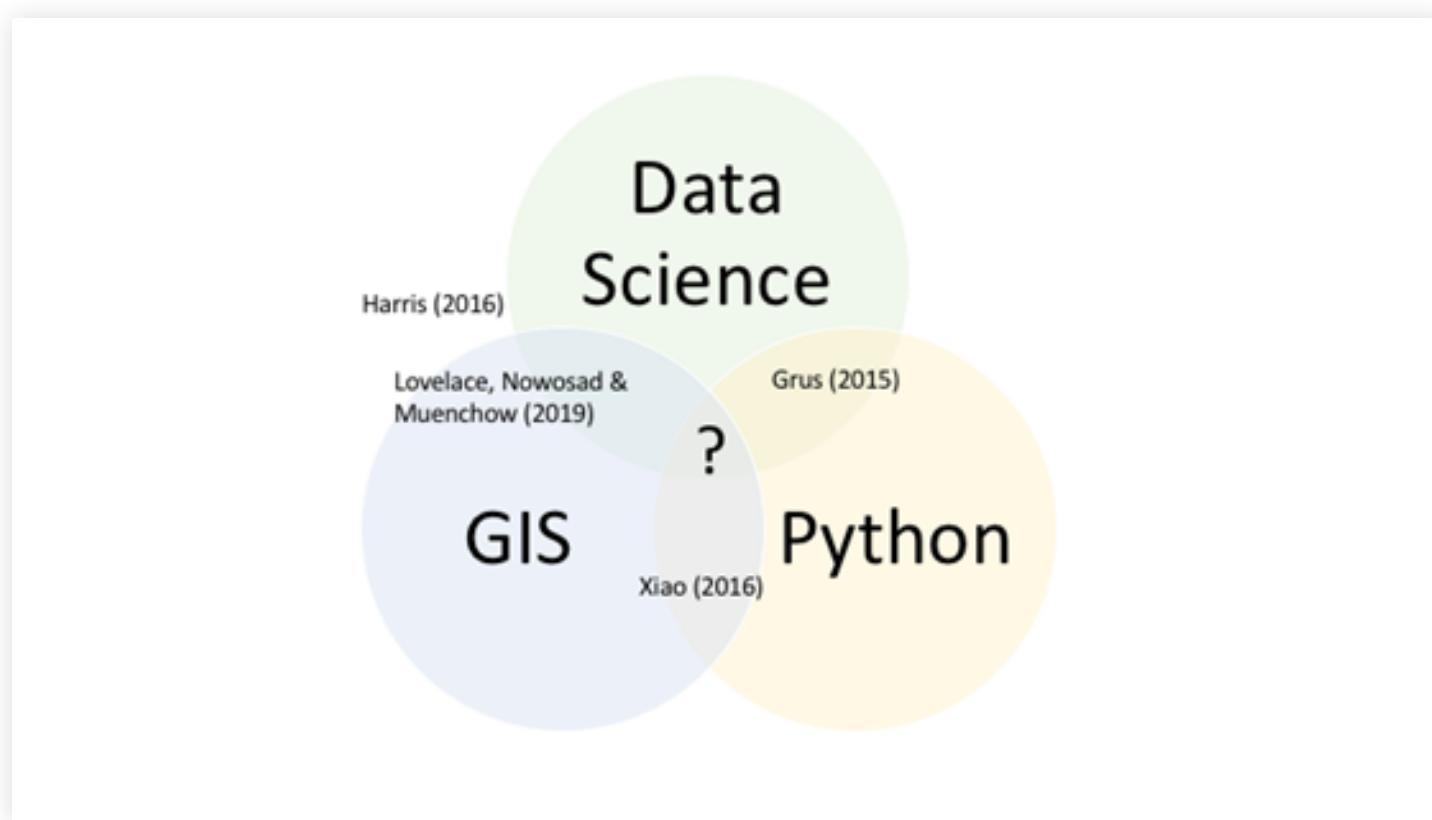
Dani
Arribas-Bel



@levijohnwolf

Levi Wolf

The Book



This Talk

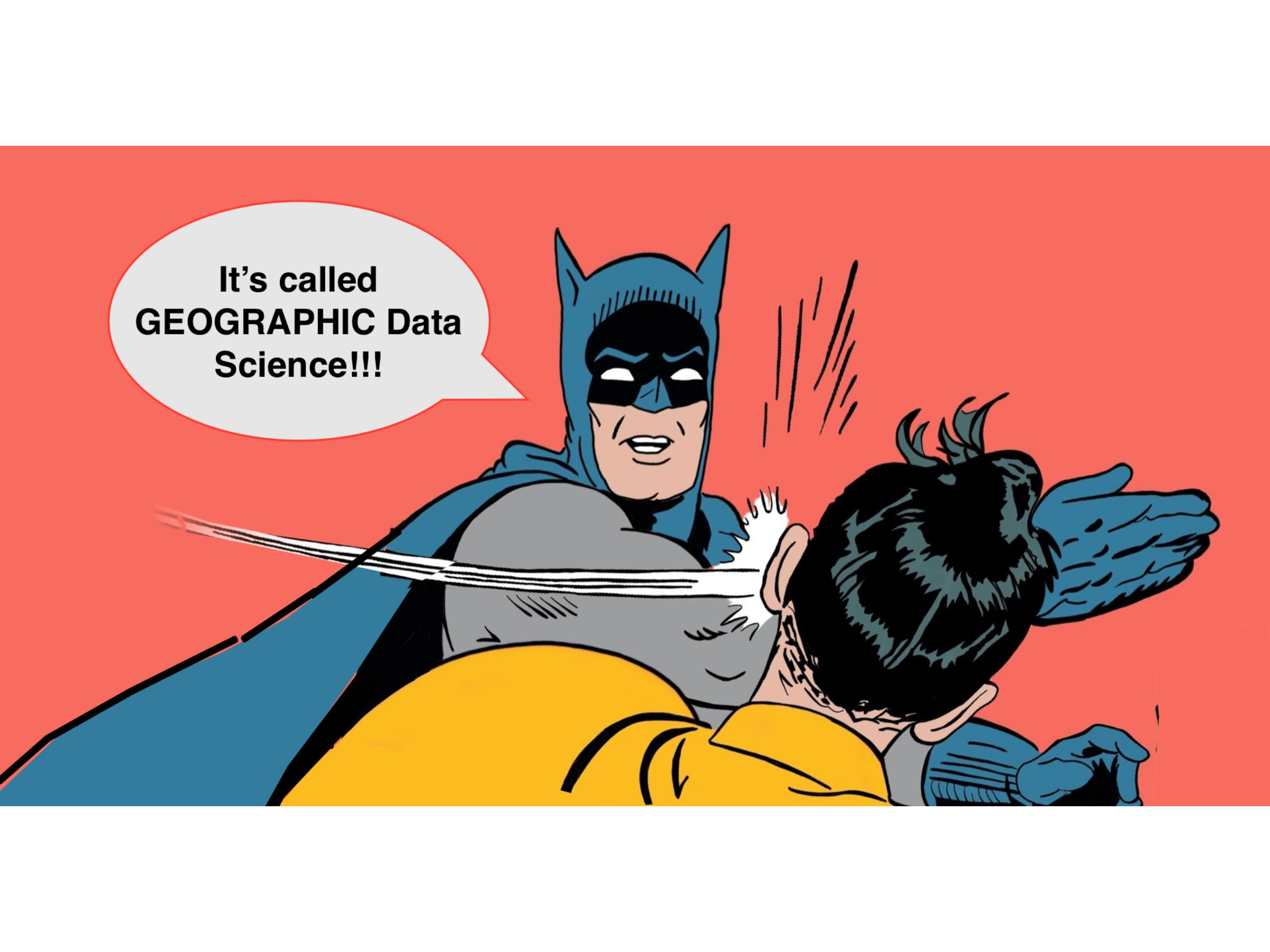
- Why
- What
- How

Why

Data, data, data

Data Science

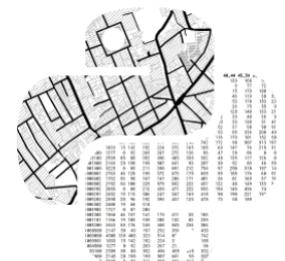
...



**It's called
GEOGRAPHIC Data
Science!!!**

Geographic Data Science

What



Geographic Data Science with Python

Search this book...

Home

PREFACE

[Table of Contents](#)

References

PART I - BUILDING BLOCKS

Overview

Geospatial Computational Environment

Geographic thinking for data scientists

Spatial Data Processing

Spatial Weights

PART II - SPATIAL DATA ANALYSIS

Overview

Choropleth Mapping

Global Spatial Autocorrelation

Local Spatial Autocorrelation

Point Pattern Analysis

PART III - ADVANCED TOPICS

Overview



Table of Contents

Part I: Building Blocks

- [Geospatial Computational Environment](#)
- [Spatial data](#)
- [Spatial data processing](#)
- [Spatial weights](#)

Part II: Spatial Data Analysis

- [Choropleth Mapping](#)
- [Spatial Autocorrelation](#)
- [Local Spatial Autocorrelation](#)
- [Point Pattern Analysis](#)

Part III: Advanced Topics

- [Spatial Inequality](#)
- [Clustering and Regionalization](#)
- [Spatial Regression](#)
- [Spatial Feature Engineering](#)

[<< Home](#)

[References >>](#)

By Sergio J. Rey, Dani Arribas-Bel, Levi J. Wolf

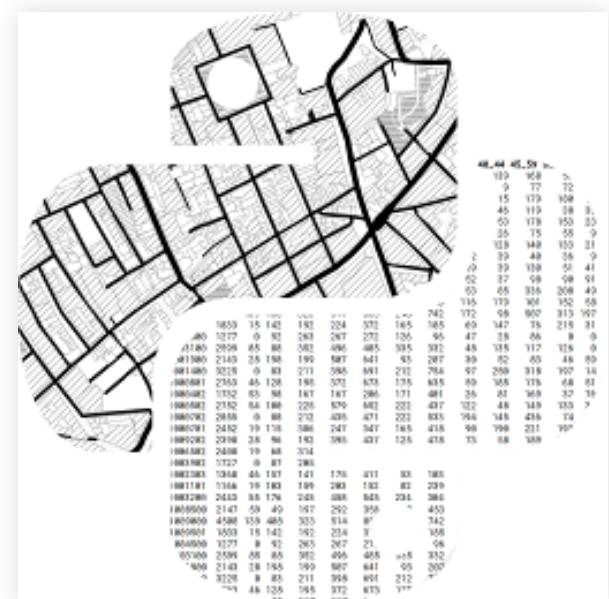
© Copyright 2020.



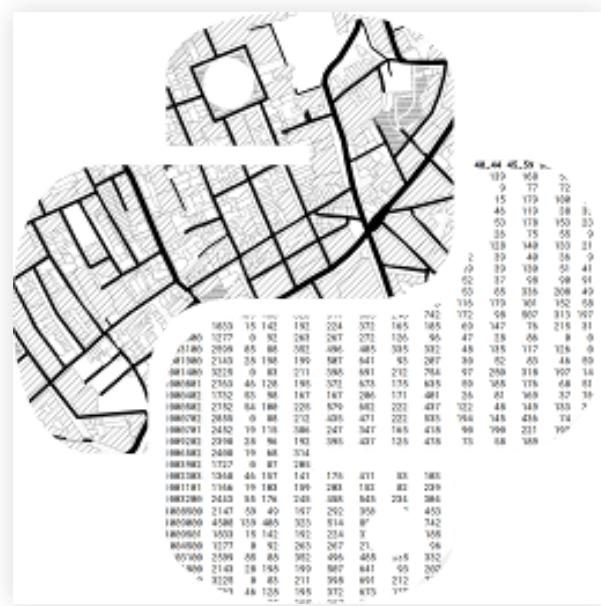
This work is licensed under a [Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License](#).

Building Blocks

- Computational Environment
- Geographical Thinking
- Spatial Data
- Spatial Weights



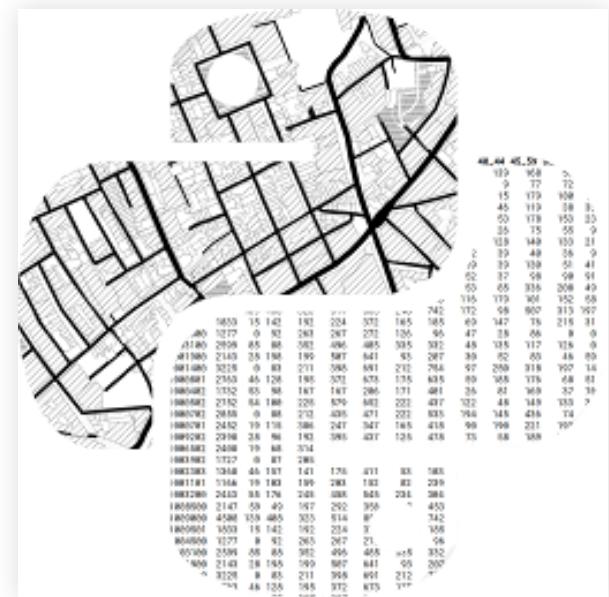
Fundamentals



- Choropleths
 - Spatial Autocorrelation
 - Local Spatial Autocorrelation
 - Point Patterns

Advanced Topics

- Spatial Inequality
 - (Spatial) Unsupervised Learning
 - Spatial Feature Engineering
 - Spatial Modelling



Bonus: Datasets

The screenshot displays three separate browser windows, each representing a different dataset on the DataHub platform:

- AirBnb**: Shows a list of download options for Airbnb data, including "Import requests", "Import parquet as zip", "Import parquet as csv", and "Import parquet as sql".
- Airports**: Shows a preview of the "airports" dataset. The code block shows how to import the data from various URLs. The preview table includes columns: source_id, feature_id, type, name, abbrev, location, gpx_code, iata_code. One row is shown: 0, 9, Airport, small, Seward, LKH, terminal, VLD, LKH.
- Brexit dataset**: Shows a preview of the "brexit_vote" dataset. The code block shows how to import the data from a URL. The preview table includes columns: constituency, party, result, turnout, and total_votes. One row is shown: 0, Conservative, 1000000, 70.0, 1400000.



This dataset includes geometries for Texas counties.

Source: Geographic Data Science with Python - Scipy 1.5

URL:

<https://geodatascience.com/texas/tif>

Previous: no processing was required for this dataset; see original source for additional information

Randomly subsetting

Set the 'seed' as every run produces the same random numbers

```
# Set the seed
np.random.seed(1234)
```

Print the first 10 rows of the table to make sure it's of length equal to the number of rows in the table

```
# Print the first 10 rows of the table
df = pd.read_csv('texas.tif')
df.head(10)
```

A function to generate a shuffled version of the table

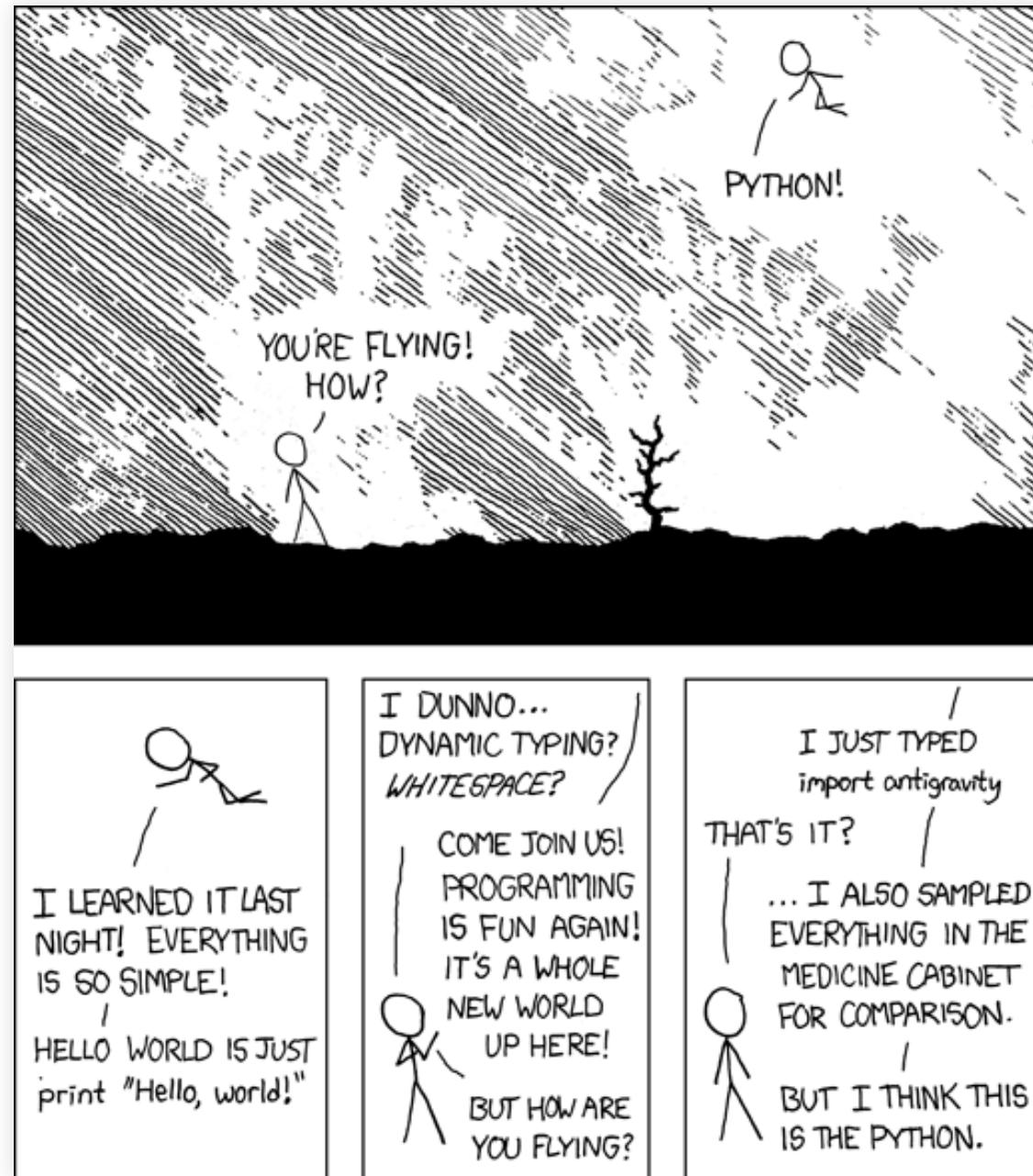
```
# Function to shuffle the table
def shuffle_table(df):
    # Create a copy of the table
    df_shuffled = df.copy()
    # Shuffle the table
    np.random.shuffle(df_shuffled)
    # Return the shuffled table
    return df_shuffled
```

If you run this function, it will produce a randomly arranged sequence

```
# If you run this function, it will produce a randomly arranged sequence
df_shuffled = shuffle_table(df)
```

How

Python



Source

Radically Open

Welcome!

Geographic Data Science with PySAL and the PyData Stack

This is the site for the book "Geographic Data Science with PySAL and the PyData Stack", by Sergio J. Rey, Dani Arribas-Bel and Levi J. Wolf. Here you can find out about the latest news regarding the book, read more about the [authors](#), or jump straight to the [book](#).

Latest news

Oct 30, 2020
Geographic data science on the road!

Apr 21, 2020
New Data Section available online

Aug 29, 2019
Work in progress...

Aug 24, 2019
Hello world!!!

Subscribe

Geographic Data Science with PySAL and the PyData Stack This book serves as an introduction to a whole new way of thinking systematically about geographic data, using geographical analysis and computation to unlock new insights hidden within data.

Sergio J. Rey, Dani Arribas-Bel & Levi J. Wolf
geographicdatascience@gmail.com

[Home](#) [Authors](#) [Book](#)

gdsbook / book

Code Issues Pull requests Actions Projects Security Insights

master

actions-user GA build of book HTML 12 days ago 341

github Change GA commit message to clarify further 2 months ago

data Fix data downloads and update gds_env for ... 2 months ago

docs GA build of book HTML 12 days ago

figures Complete draft of ch01 11 months ago

infrastructure Fix titles so they look OK on side TOC 2 months ago

notebooks GA build of book HTML 12 days ago

.gitignore Ignore NASADEM.hgt files 8 months ago

.nojekyll Create .nojekyll 2 years ago

Dockerfile Removed paired markdowns on Binder setup 2 months ago

LICENSE restructure of infrastructure to move to new JB 2 months ago

Makefile Remove CNAME (taken care of in parent rep...) 2 months ago

README.md Update README.md 15 months ago

appveyor.yml add substance and infrastructure 2 years ago

About

This book serves as an introduction to a whole new way of thinking systematically about geographic data, using geographical analysis and computation to unlock new insights hidden within data.

geographicdata.science

data-science data-analysis-python geographical-information-system geographic-data spatial-analysis spatial-statistics statistics spatial-data-analysis

Readme View license

Commits · gdsbook / book · GitHub

Code Issues Pull requests Actions Projects Security Insights

master

Commits on Nov 6, 2020

- GA build of book HTML actions-user committed 12 days ago ✓ 7375dca
- Merge pull request #102 from ljjwolf/ch2 darribas committed 12 days ago ✓ 4762249
- update spatial data chapter with additional structure ljjwolf committed 12 days ago 63adeb9

Commits on Oct 21, 2020

- GA build of book HTML actions-user committed 29 days ago ✓ d8631b1
- Merge pull request #100 from darribas/master ljjwolf committed 29 days ago ✓ d24db1a
- GA build of book HTML actions-user committed 29 days ago 0400d49
- Ch.12: Remove DEM code to go to blog: fix header hierarchy for cluste... darribas committed 29 days ago 6bbfacb

Commits on Oct 16, 2020

- add sketch of spatial data chapter ljjwolf committed on Oct 16 e67d876

Commits on Sep 21, 2020

- GA build of book HTML actions-user committed on Sep 21 310cad1
- Add refs page darribas committed on Sep 21 39d1da4
- interim build

Code as text; text as codE

The screenshot shows a Jupyter notebook interface. On the left, there's a sidebar with a map icon and the title "Geographic Data Science with Python". Below it are sections for Home, PREFACE, PART I - BUILDING BLOCKS, PART II - SPATIAL DATA ANALYSIS, PART III - ADVANCED TOPICS, and DATASETS. The main area contains a code cell and its output. The code uses the descartes library to plot an alpha shape and a convex hull over a map of Tokyo. The output shows several red circles representing the alpha shape, a blue dashed polygon for the convex hull, and black dots for source points. A legend in the bottom right corner identifies the symbols.

```
from descartes import PolygonPatch # to plot the alpha shape easily
f,ax = plt.subplots(1,1, figsize=(9,9))

# Plot a green alpha shape
ax.add_patch(PolygonPatch(alpha_shape, edgecolor='green',
                           facecolor='green', alpha=.2, label = 'Tighest alpha shape'))

# Include the points for our prolific user in black
ax.scatter(coordinates_t, color='k', marker='.', label='Source Points')
# Add a basemap
ax.imshow(basemap, extent=basemap_extent)

# plot the circles forming the boundary of the alpha shape
for i, circle in enumerate(circles):
    # only label the first circle of its kind
    if i == 0:
        label = 'Bounding Circles'
    else:
        label = None
    ax.add_patch(pit.Circle(circle, radius=alpha, facecolor='none', edgecolor='r', label=label))

# add a blue convex hull
ax.add_patch(plt.Polygon(convex_hull_vertices,
                        closed=True,
                        edgecolor='blue', facecolor='none',
                        linestyle=':', linewidth=2,
                        label='Convex Hull'))

plt.legend()
```

The screenshot shows a GitHub pull request merge page. The URL is https://github.com/gdsbook/book/runs/1285952518?cl. The page displays a list of actions for a merge pull request from darribas/master. One action, "build-html-and-deploy", is highlighted in blue and has a status of "succeeded 29 days ago in 2m 50s". Other actions listed include "Set up job", "Checkout", "Setup Miniconda", "Build HTML", "Commit files", "Push changes", "Post Setup Miniconda", "Post Checkout", and "Complete job". The GitHub interface includes a header with navigation links like Why GitHub?, Team, Enterprise, Explore, Marketplace, Pricing, and a sidebar with contents like Introduction, Location, Location, Location, Visualization, and Centrography.

Try it out!!!

Spatial Feature Engineering — [X](#)

[←](#) [→](#) [⟳](#) <https://geographicdata.science> [90%](#) [...](#) [🔒](#) [★](#)

[Download](#) [Share](#) [Print](#) [Table of Contents](#)



Geographic Data Science with Python

[!\[\]\(e7cf99a695198f48d7c53d4da8381241_img.jpg\) Search this book](#)

[Home](#)

[PREFACE](#)

[Table of Contents](#)

[References](#)

https://mybinder.org/v2/gh/gdsbook/book/master?urlpath=lab/tree/notebooks/12_feature_engineering.ipynb

Spatial Feature Engineering

In machine learning and data science, we are often equipped with *tons* of data. Indeed, given the constellation of packages to query data services, free and open source data sets, and the rapid and persistent collection of geographical data, there is simply too much data to even represent coherently in a single, tidy fashion. However, we often need to be able to construct useful *features* from this rich and deep sea of data.

Where data is available, but not yet directly *usable*, *feature engineering* helps to construct useful data for modelling.

[Launch Binder](#)

[Binder](#)

[Colab](#)

Contents

- What is spatial feature engineering?
- Feature Engineering
- Using Map Matching
- Feature Engineering using Map Synthesis
- Conclusion
- Questions

Building “Geographic Data Science...”

Universidad de Salta

Dani Arribas-Bel [@darribas]



The
Alan Turing
Institute



Geographic
Data Science
Lab