

data science for (physical) scientists I

I: epistemological concepts and working environment

1 what is data science

2 the scientific method

falsifiability

probabilistic induction

reproducibility

epistemology

3 data science tools

github

python

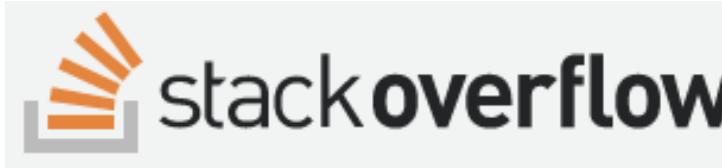
jupyter notebooks

google colab

stackoverflow

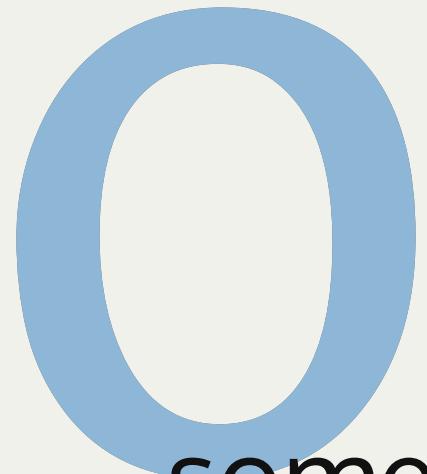


python™



this slide deck

http://bit.ly/dsps2019_1



some administrative stuff

Syllabus

<http://bit.ly/dspssyllabus>

Learning Outcomes

By the end of this class you should be able to formulate an appropriate analysis plan for a research question, select, gather, and prepare data for analysis, and choose and apply machine learning methods to the data.

Syllabus

<http://bit.ly/dspssyllabus>

The instructors is: Dr. **Federica Bianco** fbianco@udel.edu

office hours: Tentatively: Tuesday 3:30-5PM (or by appointment).

The Class assistants are:

Grader: **Yuqi Kong** kongyq@udel.edu

Office Hours: Monday 5PM Smith Hall 220

Technical and coding questions

Physics instructor: Dr. **Alexandre David-Uraz** adu@udel.edu

Office Hours: TBD Sharp 101B physics help center

Physics and Method-Application questions

Syllabus

<http://bit.ly/dspssyllabus>

- 10% pre-class questions
- 10% class performance and participation
- 20% homeworks
- 25% midterm
- 35% final

quizz

<http://bit.ly/dspssyllabus>

- 10% pre-class questions
- 10% class performance and participation
- 20% homeworks
- 25% midterm
- 35% final

*from beginning of class to 5 minutes past (be on time!)
questions on previous class material and reading assignments*

participation

<http://bit.ly/dspssyllabus>

- 10% pre-class questions ask questions
- 10% class performance and participation answer questions
- 20% homeworks get up and code
- 25% midterm extra credit assignments
- 35% final

homework

<http://bit.ly/dspssyllabus>

- **10% pre-class questions**
- **10% class performance and participation**
- **20% homeworks**
- **25% midterm**
- **35% final**

Homework projects must be turned in as *jupyter notebooks* by checking them into your [github](#) account in a DSPS_<firstinitialLastname> repo and the project directories HW<hw number> (unless otherwise stated).
<finitialLastname> is e.g. fBianco

homework

Please work in groups of up to 5 people on homework as a collaborative projects.

Individual notebooks must be returned for each homework. Different group members should lead different aspects of the work. A statement **must be included in the README** explaining each team member's contribution (similar to an acknowledge of contribution you would find in a *Nature* letter see, for example [these contributions](#)).

- 10% pre-class questions
- 10% class performance and participation
- 20% homeworks
- 25% midterm
- 35% final

homework

Please work in groups of up to 5 people on homework as a collaborative projects.

Individual notebooks must be returned for each homework. Different group members should lead different aspects of the work. A statement **must be included in the README** explaining each team member's contribution (similar to an acknowledge of contribution you would find in a *Nature* letter see, for example [these contributions](#)).

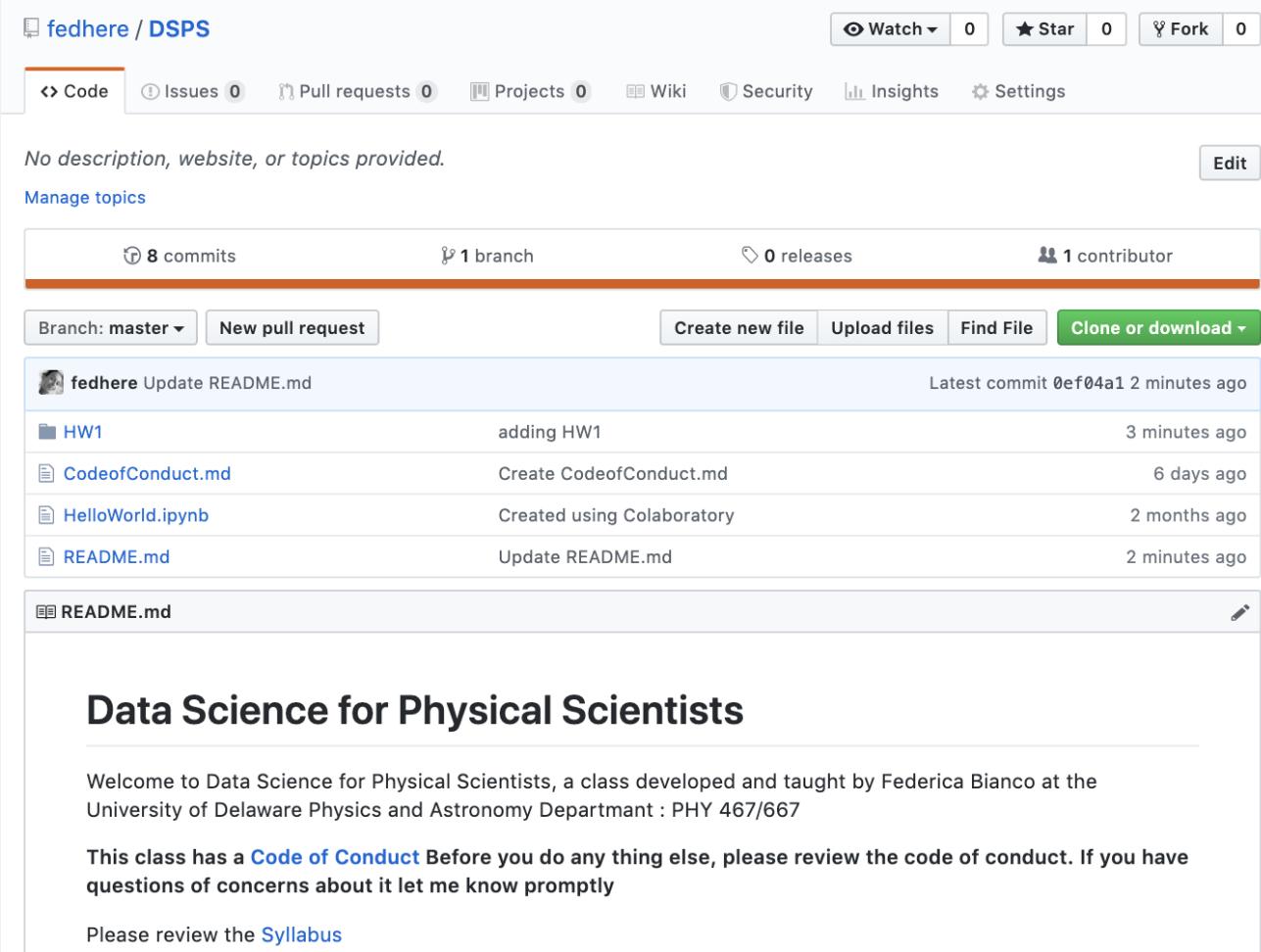
- **10% pre-class questions**
- **10% class performance and participation**
- **20% homeworks**
- **25% midterm**
- **35% final**

Contributions

K.T. led the project and reduced the ALMA data. K.T. and D.I. wrote the manuscript. M.S.Y. reduced the Large Millimeter Telescope data and edited the final manuscript. Other authors contributed to the interpretation and commented on the ALMA proposal and the paper.

homework

- 10% pre-class questions
- 10% class performance and participation
- 20% homeworks
- 25% midterm
- 35% final

A screenshot of a GitHub repository page for "fedhere / DSPS". The repository has 8 commits, 1 branch, 0 releases, and 1 contributor. The README.md file contains the following content:
Data Science for Physical Scientists
Welcome to Data Science for Physical Scientists, a class developed and taught by Federica Bianco at the University of Delaware Physics and Astronomy Department : PHY 467/667
This class has a [Code of Conduct](#) Before you do any thing else, please review the code of conduct. If you have questions of concerns about it let me know promptly
Please review the [Syllabus](#)

instructions will be here

<https://github.com/fedhere/DSPS>

homework

- 10% pre-class questions
- 10% class performance and participation
- 20% homeworks
- 25% midterm
- 35% final

fedhere / DSPS

Code Issues 0 Pull requests 0 Projects 0 Wiki Security Insights Settings

No description, website, or topics provided.

Edit Manage topics

8 commits 1 branch 0 releases 1 contributor

Branch: master New pull request Create new file Upload files Find File Clone or download

fedhere Update README.md Latest commit 0ef04a1 2 minutes ago

HW1 adding HW1 3 minutes ago

CodeofConduct.md Create CodeofConduct.md 6 days ago

HelloWorld.ipynb Created using Colaboratory 2 months ago

README.md Update README.md 2 minutes ago

README.md

Data Science for Physical Scientists

Welcome to Data Science for Physical Scientists, a class developed and taught by Federica Bianco at the University of Delaware Physics and Astronomy Department : PHY 467/667

This class has a [Code of Conduct](#) Before you do any thing else, please review the code of conduct. If you have questions of concerns about it let me know promptly

Please review the [Syllabus](#)

instructions will be here
<https://github.com/fedhere/DSPS>

online help: <http://bit.ly/2HtkQoc>
please sign up asap!!



homework

The screenshot shows a course page from UD Canvas. At the top left is the UD logo. To its right is the course identifier "19F-PHYS467/PHYS667-011". On the far right is a three-line menu icon. Below the course identifier is the text "2019 Fall". A vertical sidebar on the left contains links: Account (with a person icon), Dashboard (with a clock icon), Courses (with a book icon), Calendar (with a calendar icon), and Inbox (with an envelope icon). The "Courses" link is highlighted with a blue box. To the right of the sidebar, the main content area has a header "Recent Activity in 19F-PHYS467/PHYS667-011". Below it is a message box containing the text "No Recent Messages You don't have any messages to show in your stream yet. Once you begin participating in your courses you'll see this stream fill up with messages from discussions, grading updates, private messages between you and other users, etc." with an information icon. Further down the page, there are sections for "To Do" (which says "Nothing for now") and a "View Course Calendar" button.

- **10% pre-class questions**
- **10% class performance and participation**
- **20% homeworks**
- **25% midterm**
- **35% final**

of course there is also UD Canvas, which will be used to give you grades and occasionally post messages (I am still learning how to use it tho!)

HW 1 is posted already

midterm

- 10% pre-class questions
- 10% class performance and participation
- 20% homeworks
- 25% midterm
- 35% final

For the *Midterm* and the *Final* you are responsible for material in the labs, the reading, and the homework. **In preparing for the exams, use the homework as a guide to which material is essential.** In the Midterm and Final YOU WILL BE EXPECTED TO WORK INDIVIDUALLY.

Midterm... probably in class

issues: stereotype thread - working under derass is not necessarily a required skill
advantages: interviews for jobs are often timed

final

- 10% pre-class questions
- 10% class performance and participation
- 20% homeworks
- 25% midterm
- 35% final

For the *Midterm* and the *Final* you are responsible for material in the labs, the reading, and the homework. **In preparing for the exams, use the homework as a guide to which material is essential.** In the Midterm and Final **YOU WILL BE EXPECTED TO WORK INDIVIDUALLY.**

Final: take home, multiple days.

Resources

The screenshot shows a GitHub repository page for the user 'fedhere' named 'DSPS'. The repository has 8 commits, 1 branch, 0 releases, and 1 contributor. The latest commit was made 2 minutes ago. The repository description is: 'Data Science for Physical Scientists'. It includes a welcome message from Federica Bianco, a code of conduct link, and a syllabus link.

No description, website, or topics provided. [Edit](#)

Manage topics

8 commits 1 branch 0 releases 1 contributor

Branch: master ▾ New pull request Create new file Upload files Find File Clone or download ▾

fedhere Update README.md Latest commit 0ef04a1 2 minutes ago

HW1 adding HW1 3 minutes ago

CodeofConduct.md Create CodeofConduct.md 6 days ago

HelloWorld.ipynb Created using Colaboratory 2 months ago

README.md Update README.md 2 minutes ago

README.md [Edit](#)

Data Science for Physical Scientists

Welcome to Data Science for Physical Scientists, a class developed and taught by Federica Bianco at the University of Delaware Physics and Astronomy Department : PHY 467/667

This class has a [Code of Conduct](#) Before you do any thing else, please review the code of conduct. If you have questions of concerns about it let me know promptly

Please review the [Syllabus](#)

<https://github.com/fedhere/DSPS>

Resources

<https://github.com/fedhere/DSPS>

- SLIDES here
- HOMEWORK INSTRUCTIONS here
- RESOURCES here

If notebooks do not display

use

<https://nbviewer.jupyter.org>

The screenshot shows the GitHub repository page for 'fedhere / DSPS'. The repository has 8 commits, 1 branch, 0 releases, and 1 contributor. The latest commit was made 2 minutes ago. The README.md file contains the following content:

```
Data Science for Physical Scientists

Welcome to Data Science for Physical Scientists, a class developed and taught by Federica Bianco at the University of Delaware Physics and Astronomy Department : PHY 467/667

This class has a Code of Conduct Before you do any thing else, please review the code of conduct. If you have questions of concerns about it let me know promptly

Please review the Syllabus
```

Resources

Resources

The primary textbooks are:

- **Elements of Statistical Learning**, Hastie,Tibshirani,Friedman, Springer 2001
- **Python Data Science Handbook**, Jake VanderPlas, O'Reilly Media
[<https://www.oreilly.com/library/view/python-data-science/9781491912126/>]
- **Statistics, Data Mining, and Machine Learning in Astronomy**, Ivezic, Connolly, VanderPlas, Gray, Princeton Press

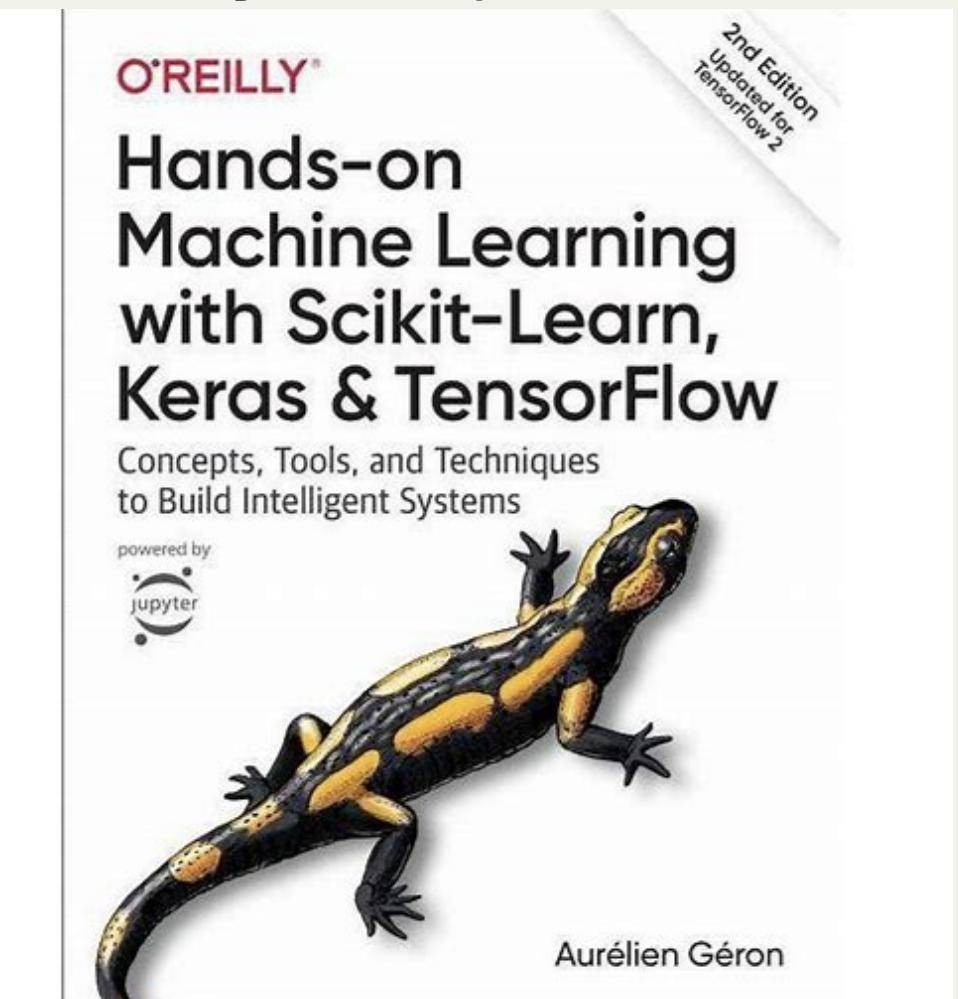
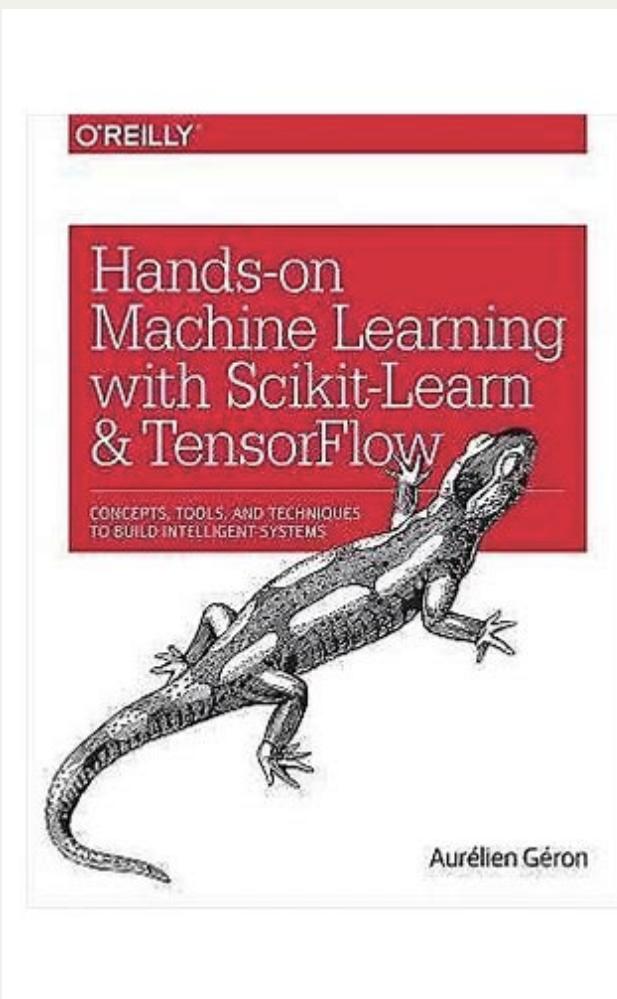
In addition, depending on your familiarity with coding, statistics, and visualization

- **ML in python: Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow** probably the book that is closer to the syllabus in terms of techniques, but don't buy it, because the second edition is due to come out imminently and the deep learning chapters of the previous edition are out of date now
- computing and coding: **Beginning Python Visualization**, 2009
- data analysis: **Statistics in a nutshell**, S. Boslaugh, O'Reilly Media
- **Interactive Data Visualization**, S. Murray, O'Reilly Media
- Visualizations: **Visualizations Analysis and Design**, T. Munzer, 2014

Resources

<https://github.com/ageron/handson-ml>

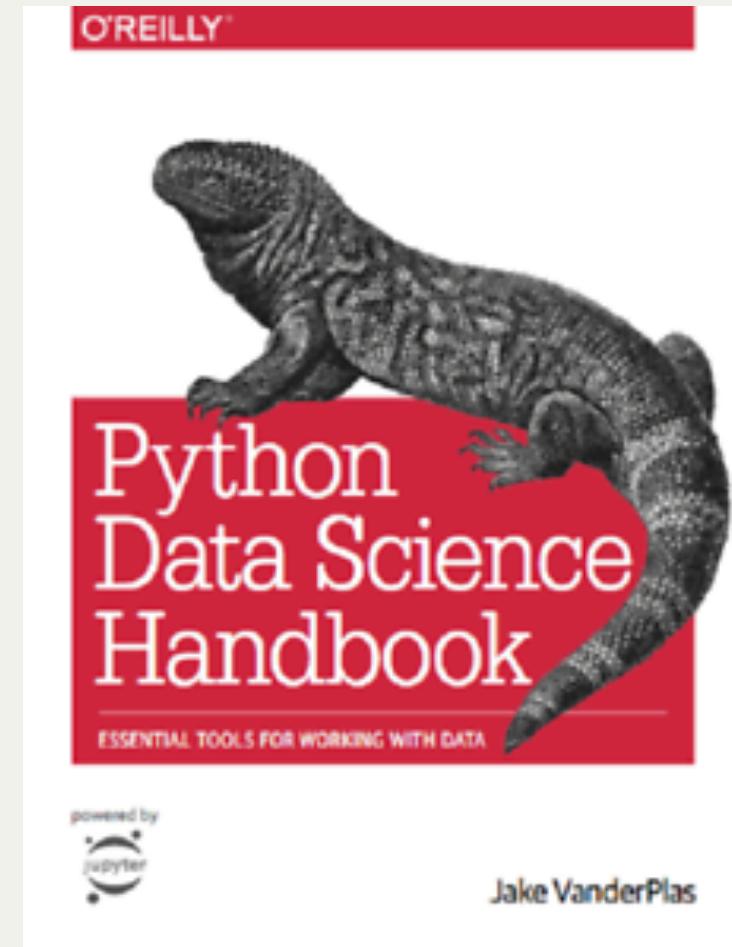
don't buy this! the second edition will come out soon and TensorFlow has changed significantly!



Resources

<http://vanderplas.com/>

Jake Vanderplas is a physicist-data scientists



<http://vanderplas.com/>

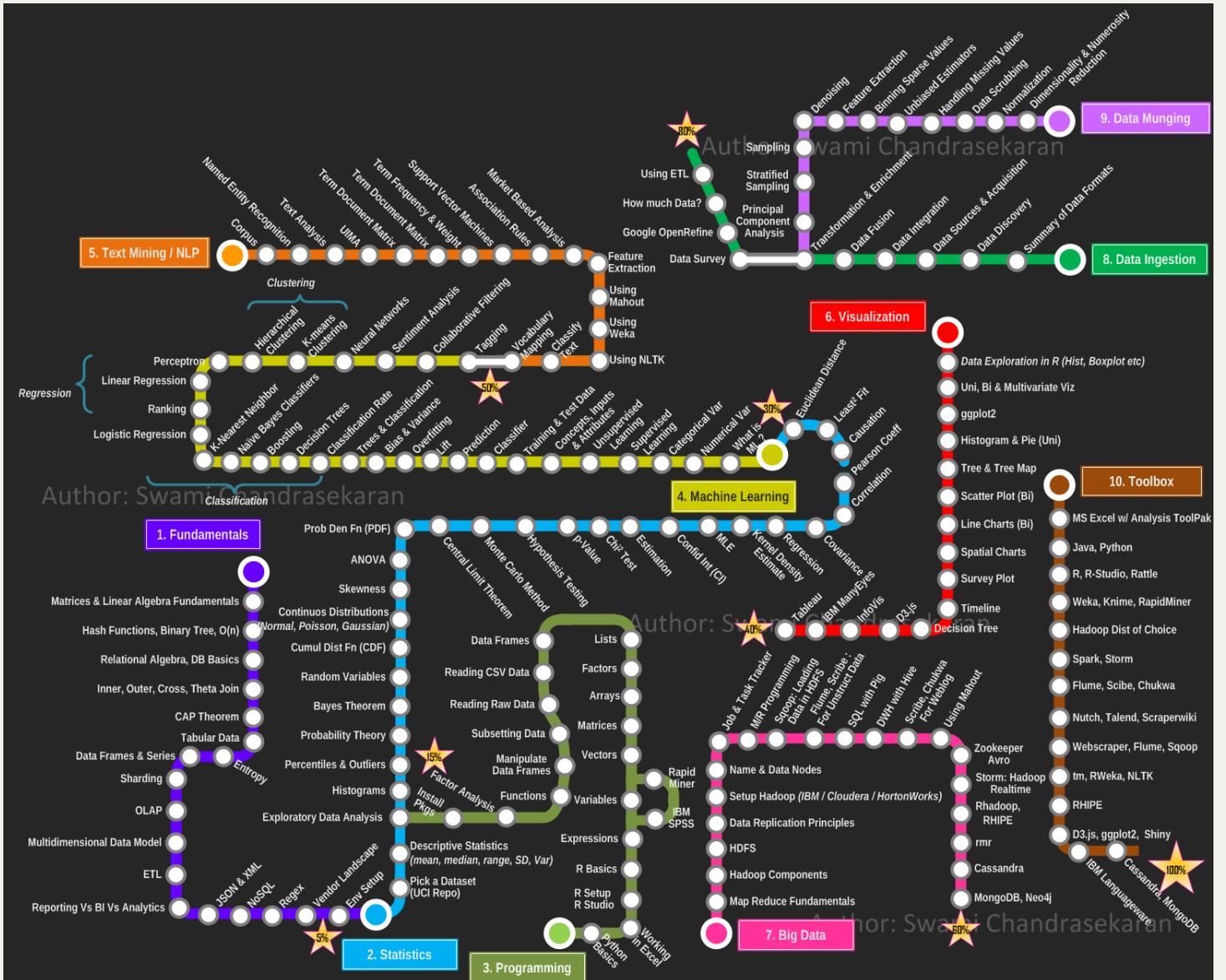
Title Text

Subtitle Text

<https://www.youtube.com/embed/ZyjCqQEUA8o?start=310&enablejsapi=1>

1

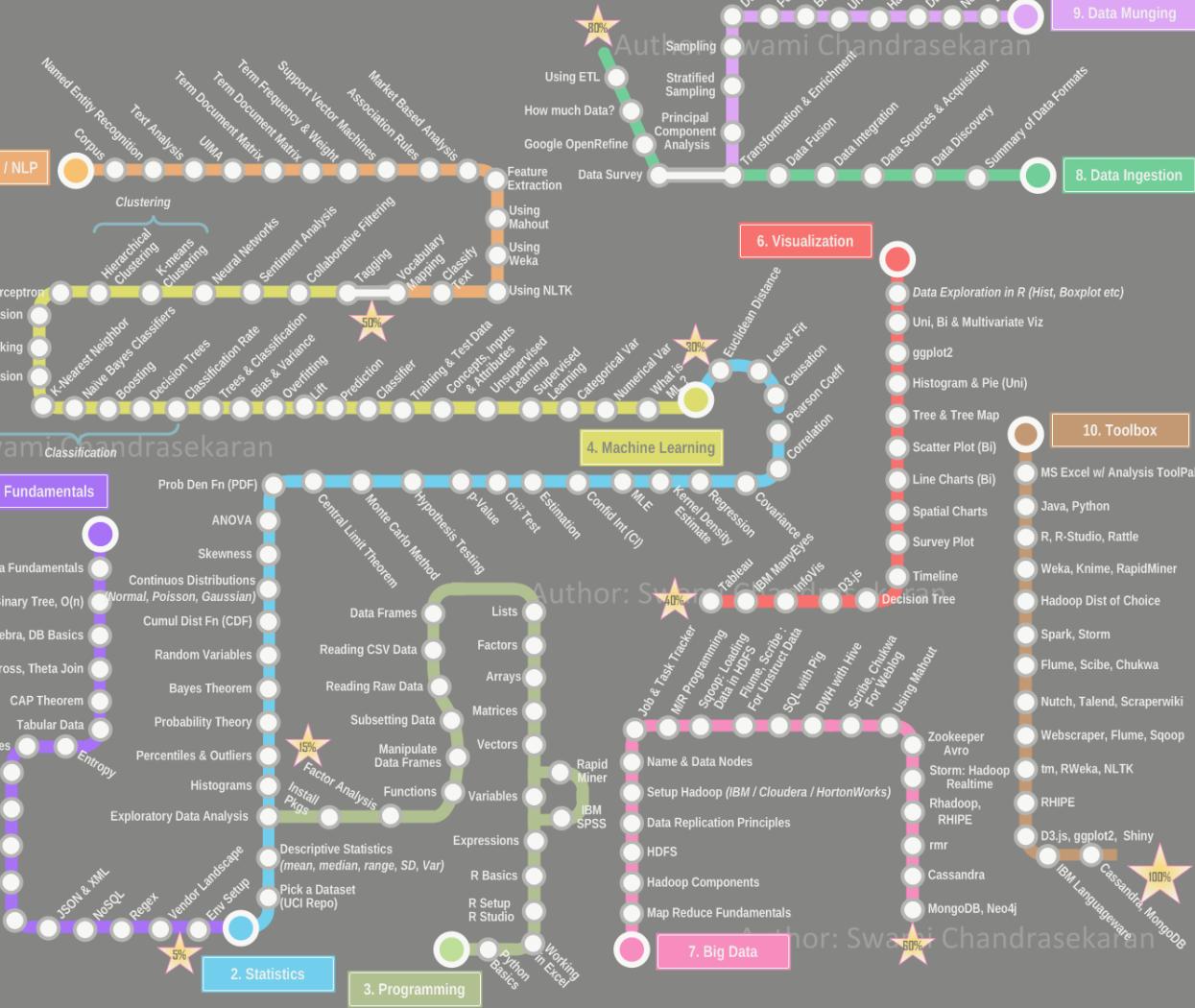
what is data science?



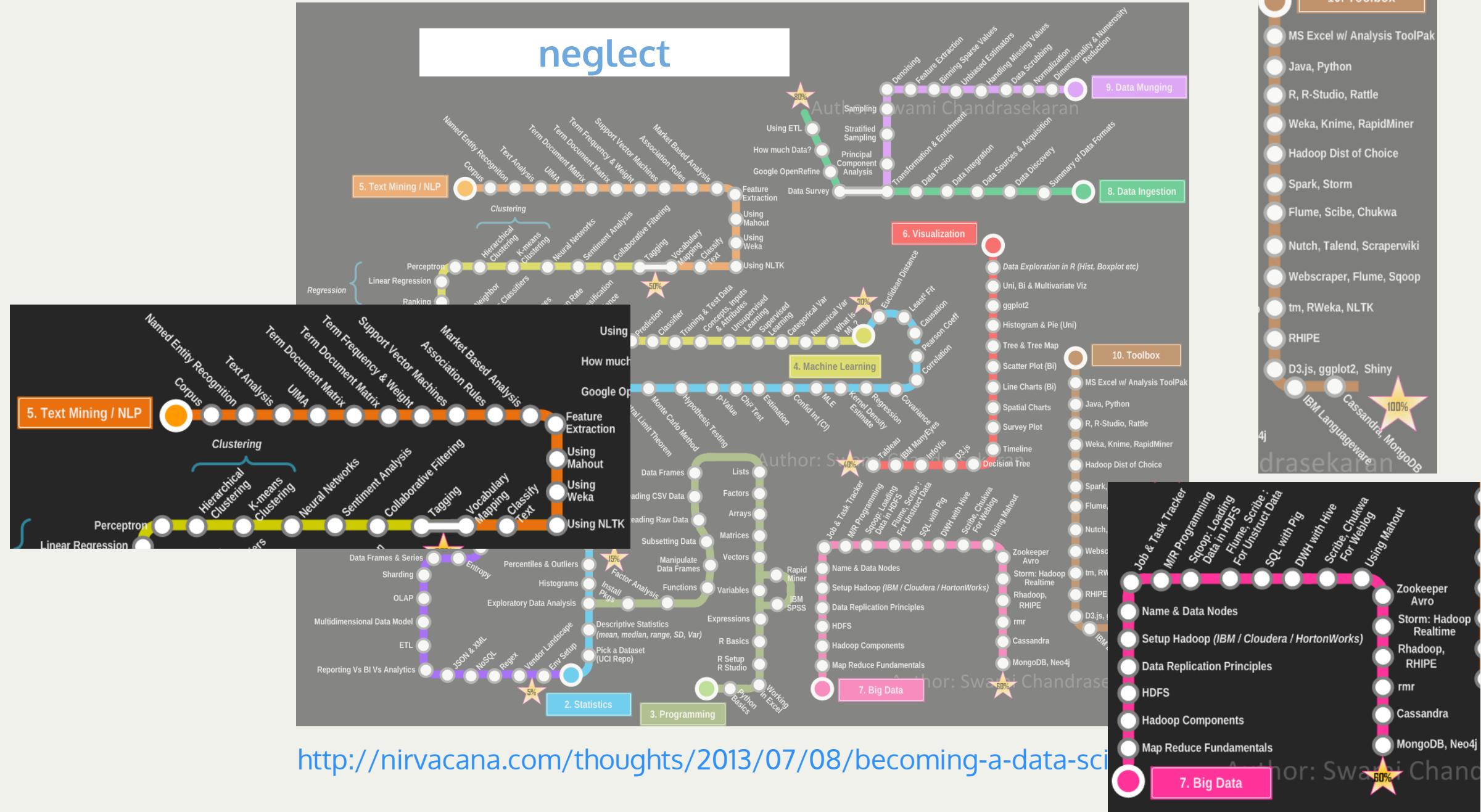
1. Fundamentals

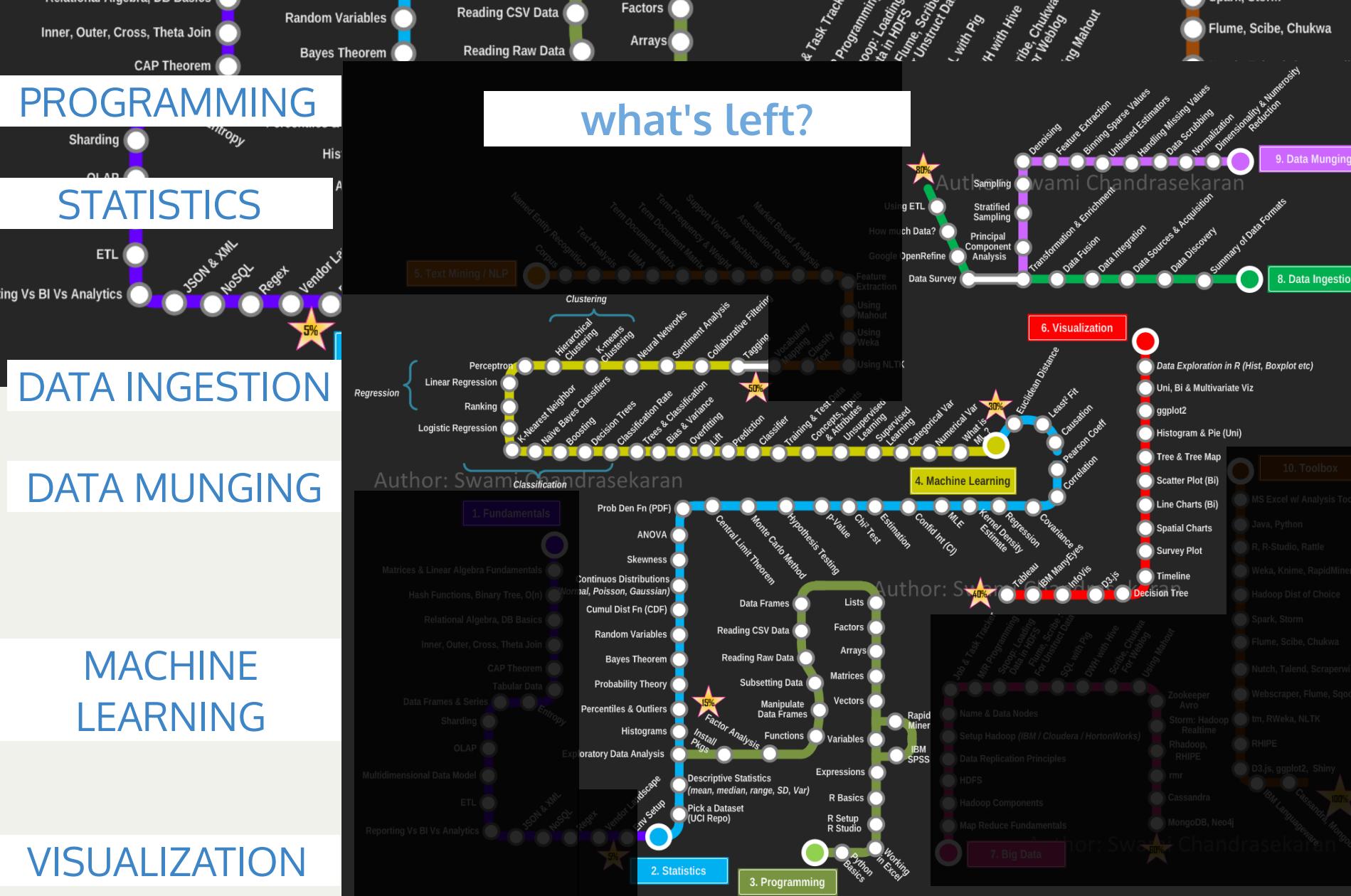


assume you know



<http://nirvacana.com/thoughts/2013/07/08/becoming-a-data-scientist>





<http://nirvacana.com/thoughts/2013/07/08/becoming-a-data-scientist>

PROGRAMMING

STATISTICS

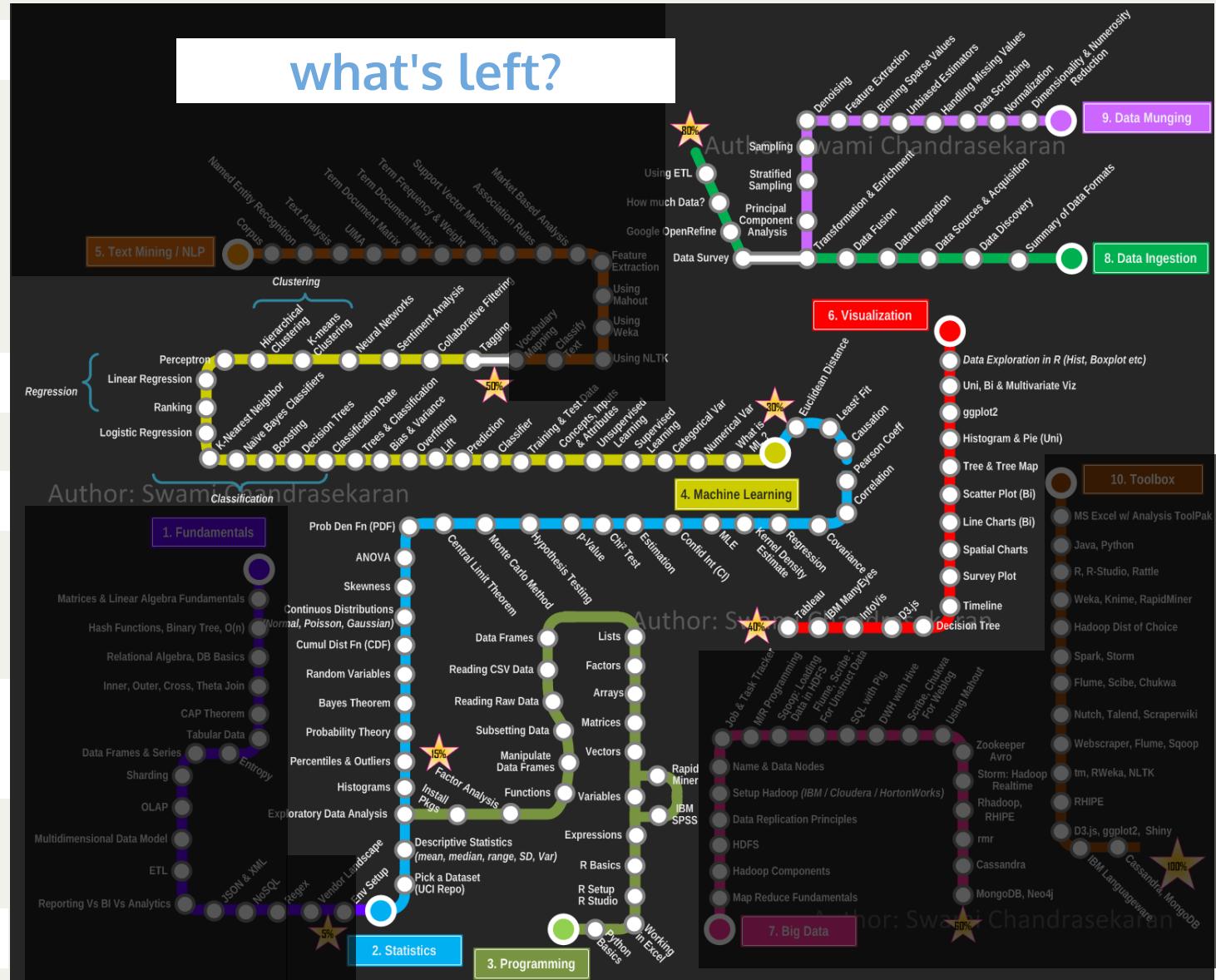
DATA INGESTION

DATA MUNGING

MACHINE LEARNING

VISUALIZATION

python



<http://nirvacana.com/thoughts/2013/07/08/becoming-a-data-scientist>

PROGRAMMING

STATISTICS

DATA INGESTION

DATA MUNGING

MACHINE LEARNING

VISUALIZATION

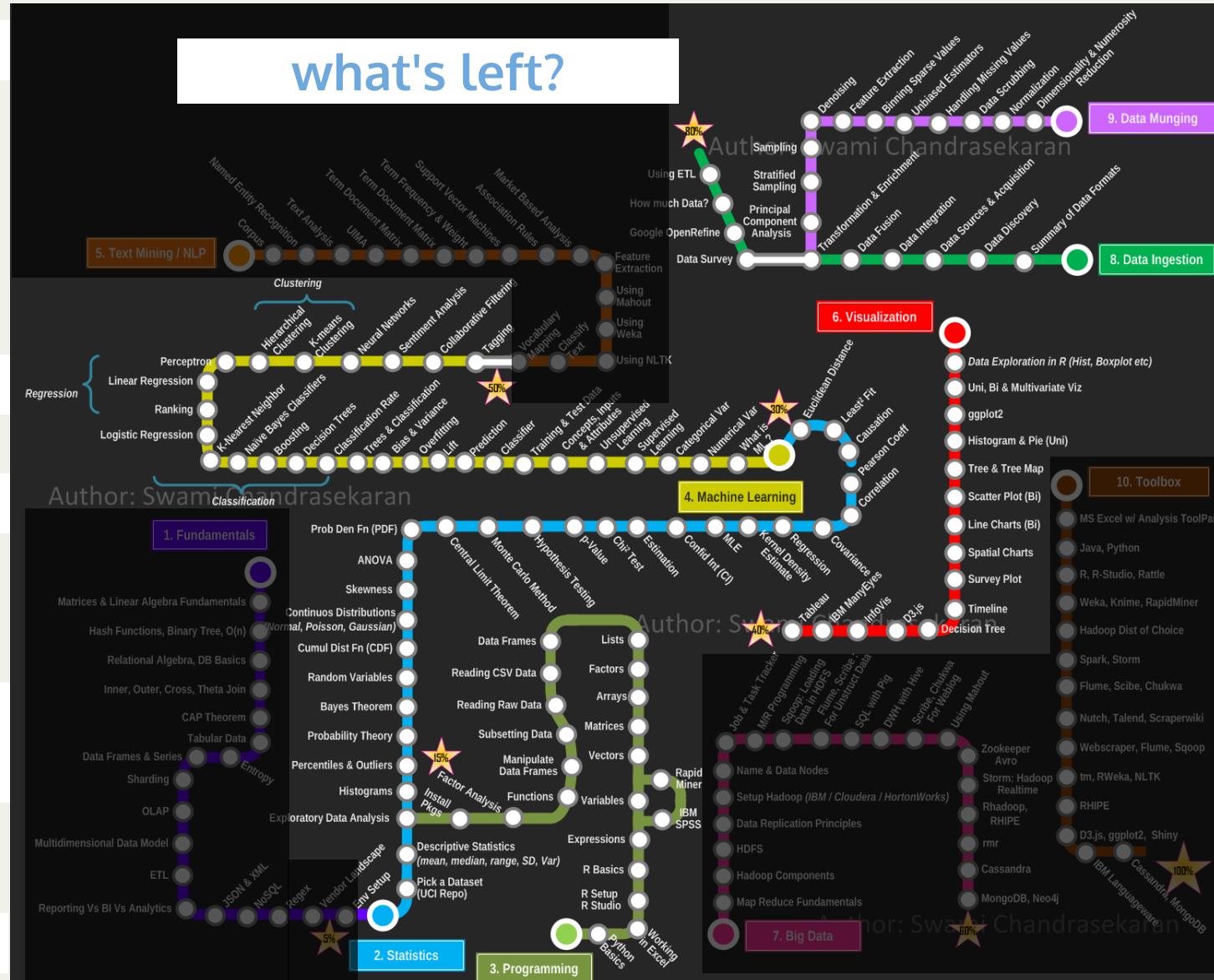
python

probability distributions

p-values

uncertainties

MCMC



PROGRAMMING

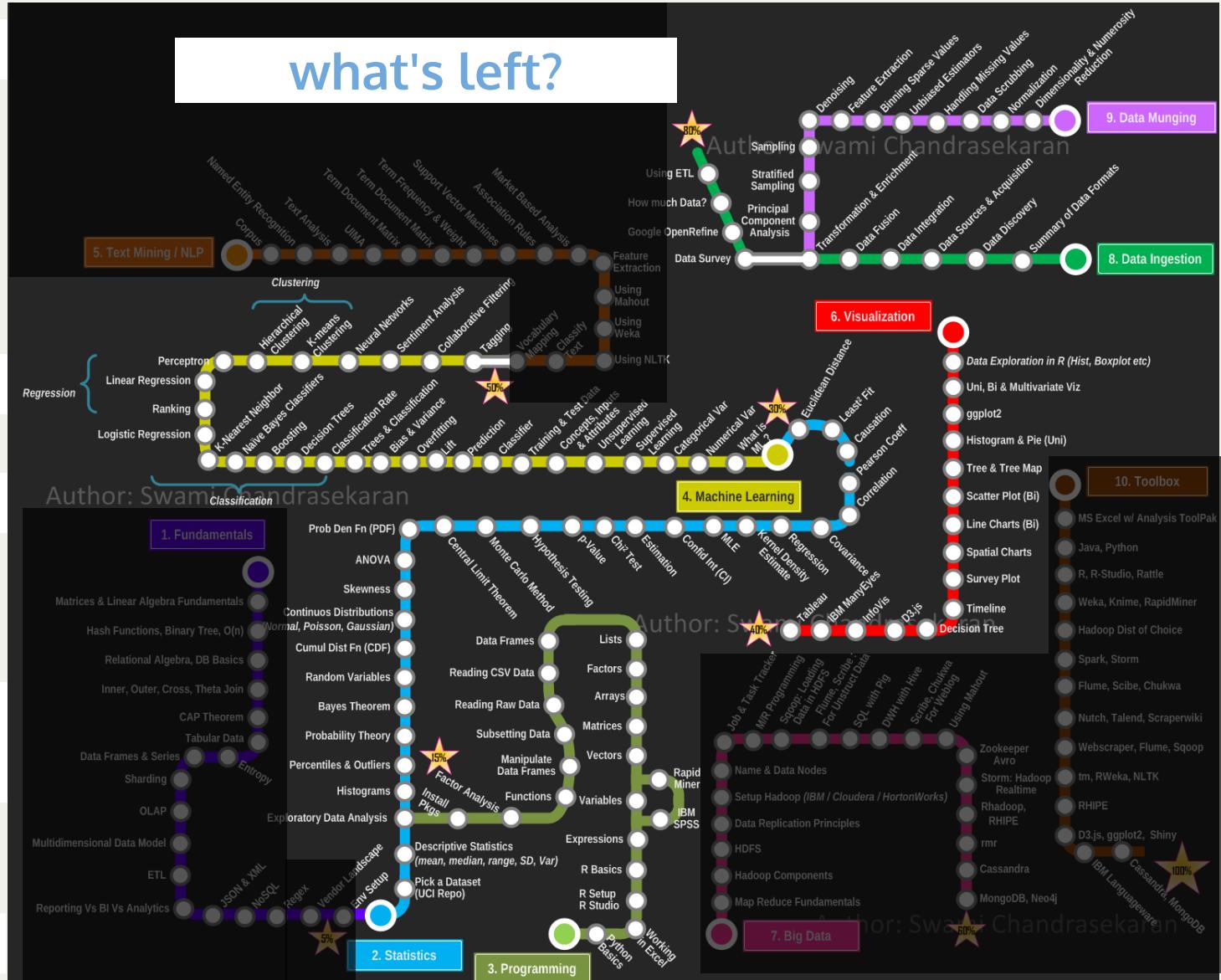
STATISTICS

DATA INGESTION

DATA MUNGING

MACHINE LEARNING

VISUALIZATION



python

probability

p-values

uncertainties

MCMC

regression

(linear, template)

classification (trees, neural networks)

clustering

2

the scientific method
(what is science?)

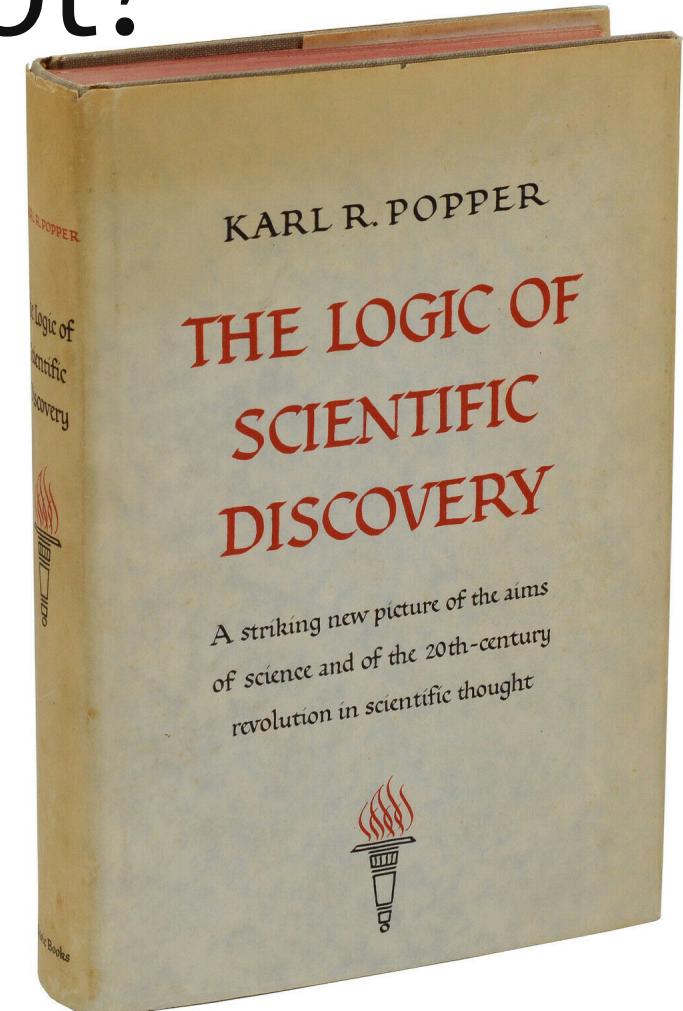
epistemology:

the philosophy of science and
of the scientific method

the *demarcation* problem: what is science? what is not?

My proposal is based upon an *asymmetry* between **verifiability** and **falsifiability**; an asymmetry which results from the logical form of universal statements. For these are never derivable from singular statements, but can be contradicted by singular statements.

—Karl Popper, *The Logic of Scientific Discovery*

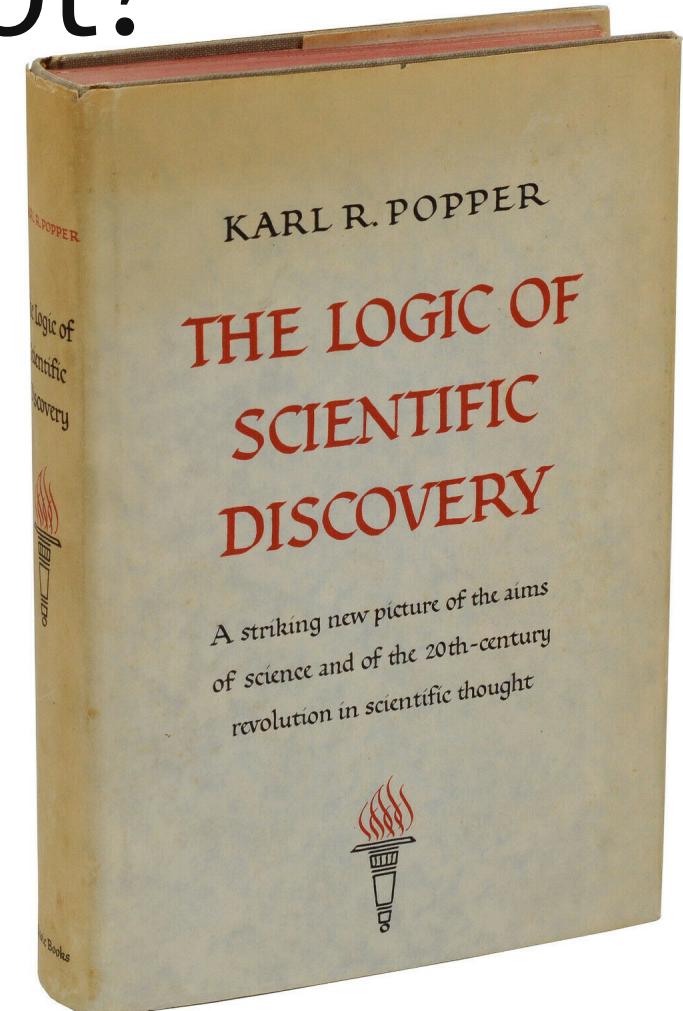


the *demarcation* problem: what is science? what is not?

My proposal is based upon an *asymmetry* between **verifiability** and **falsifiability**; an asymmetry which results from the logical form of universal statements. For these are never derivable from singular statements, but can be contradicted by singular statements.

—Karl Popper, *The Logic of Scientific Discovery*

a scientific theory must be
falsifiable



the *demarcation* problem

model —————→ prediction

the *demarcation* problem

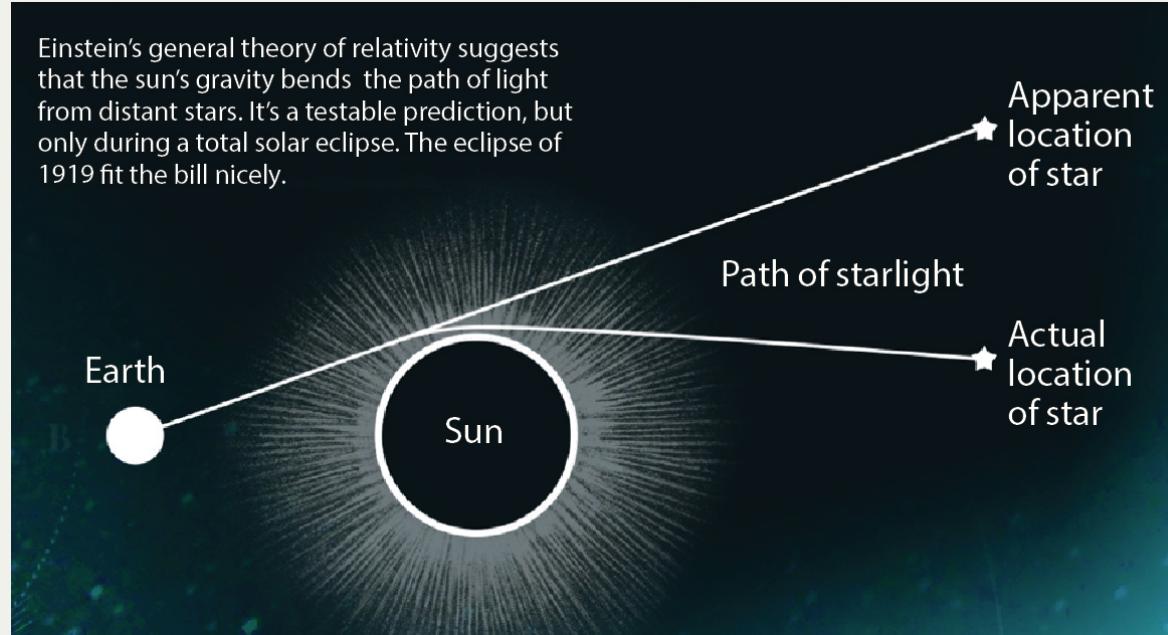
model

Einstein GR

$$R_{\mu\nu} - \frac{1}{2}Rg_{\mu\nu} = 8\pi GT_{\mu\nu}$$



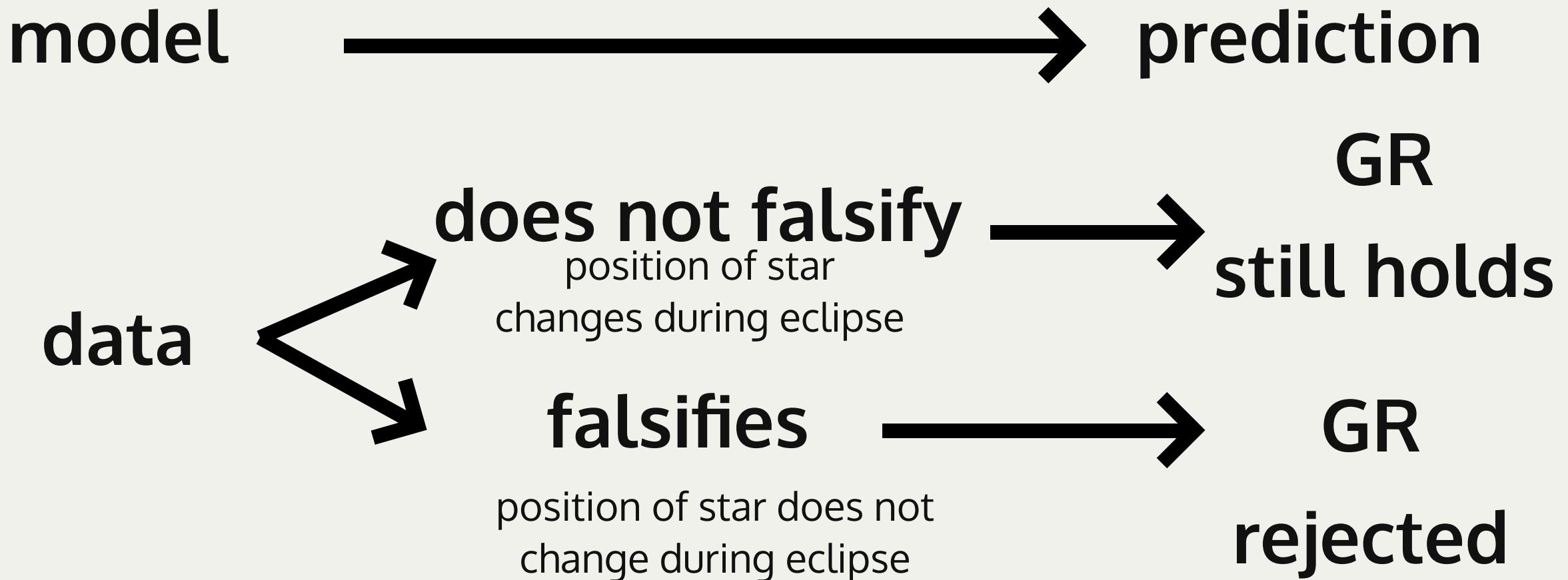
prediction



Light rays are deflected by mass

<http://discovermagazine.com/2019/may/why-it-took-the-1919-solar-eclipse-for-physicists-to-believe-einstein>

the *demarcation* problem



the *demarcation* problem

is psychology a science?

DISCUSS!

the *demarcation* problem

things can get more complicated though:

most scientific theories are actually based largely on *probabilistic induction* and modern *inductive inference* (Solomonoff, frequentist vs Bayesian methods...)

Reproducibility

Reproducible research means:

the ability of a researcher to duplicate the results of a prior study using the same materials as were used by the original investigator. That is, a second researcher might use the same raw data to build the same analysis files and implement the same statistical analysis in an attempt to yield the same results.

<https://acmedsci.ac.uk/viewFile/56314e40aac61.pdf>

Reproducibility

Reproducible research means:

the ability of a researcher to duplicate the results of a prior study using the same materials as were used by the original investigator. That is, a second researcher might use the same raw data to build the same analysis files and implement the same statistical analysis in an attempt to yield the same results.

<https://acmedsci.ac.uk/viewFile/56314e40aac61.pdf>

why?



assures a result is
grounded in evidence

#openscience
#opendata

Reproducibility

Reproducible research means:

the ability of a researcher to duplicate the results of a prior study using the same materials as were used by the original investigator. That is, a second researcher might use the same raw data to build the same analysis files and implement the same statistical analysis in an attempt to yield the same results.

why?



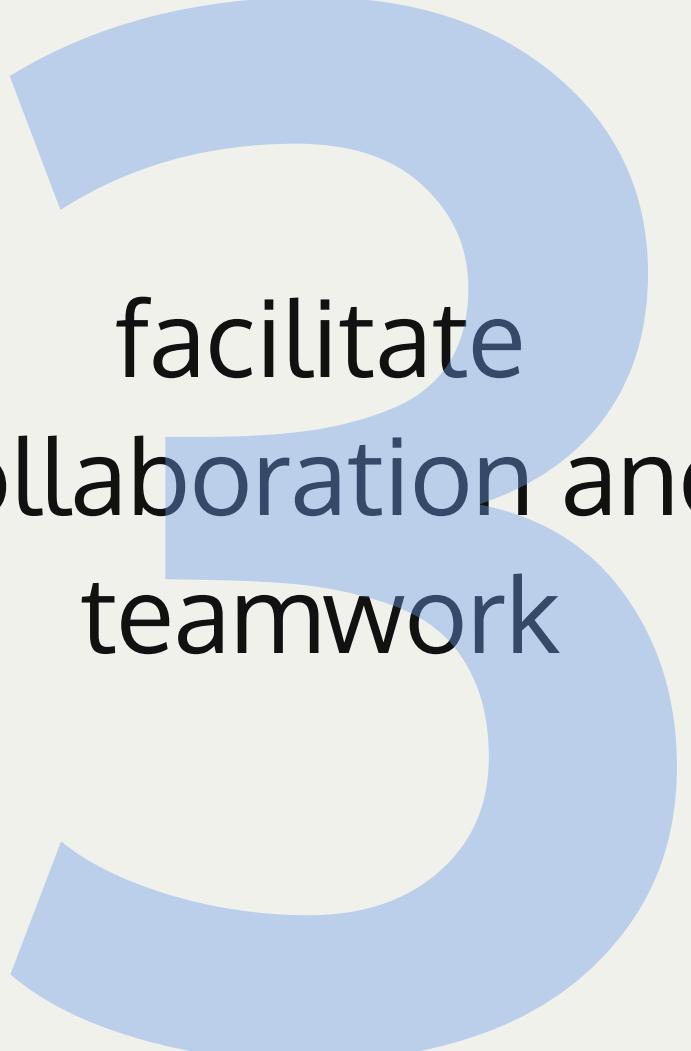
facilitates scientific progress by avoiding the need to duplicate unoriginal research

Reproducibility

Reproducible research means:

the ability of a researcher to duplicate the results of a prior study using the same materials as were used by the original investigator. That is, a second researcher might use the same raw data to build the same analysis files and implement the same statistical analysis in an attempt to yield the same results.

why?



facilitate
collaboration and
teamwork

Reproducibility

Reproducible research means:

the ability of a researcher to duplicate the results of a prior study using the same materials as were used by the original investigator. That is, a second researcher might use the same raw data to build the same analysis files and implement the same statistical analysis in an attempt to yield the same results.

<https://acmedsci.ac.uk/viewFile/56314e40aac61.pdf>

Reproducible research in practice:
all numbers in a data analysis can be recalculated exactly (down to stochastic variables!) using the **code** and **raw data** provided by the analyst.

*Claerbout, J. 1990,
Active Documents and Reproducible
Results, Stanford Exploration Project
Report, 67, 139*

Repdorucibility

Reproducible research means:

the ability of a researcher to duplicate the results of a prior study using the same materials as were used by the original investigator. That is, a second researcher might use the same raw data to build the same analysis files and implement the same statistical analysis in an attempt to yield the same results.

<https://acmedsci.ac.uk/viewFile/56314e40aac61.pdf>

Reproducible research in practice:
all numbers in a data analysis can be recalculated exactly (down to stochastic variables!) using the **code** and **raw data** provided by the analyst.

- provide raw data and code to reduce it to all stages needed to get outputs
- provide code to reproduce all figures
- provide code to reproduce all number outcomes

3

the tools

github *reproducibility*



allows reproducibility through code distribution

<https://github.com>

Reproducible research means:

all numbers in a data analysis can be recalculated exactly (down to stochastic variables!) using the **code** and **raw data** provided by the analyst.

Claerbout, J. 1990,

Active Documents and Reproducible Results, Stanford Exploration Project Report, 67, 139

github *version control*



allows version control

<https://github.com>

the Git software

is a distributed *version control system*:
a version of the files on your local computer
is made also available at a central server.
The history of the files is saved remotely so
that any version (that was checked in) is
retrievable.

github **collaborative platform**



allows effective collaboration

<https://github.com>

collaboration tool

by fork, fork and pull request, or by working directly as a collaborator

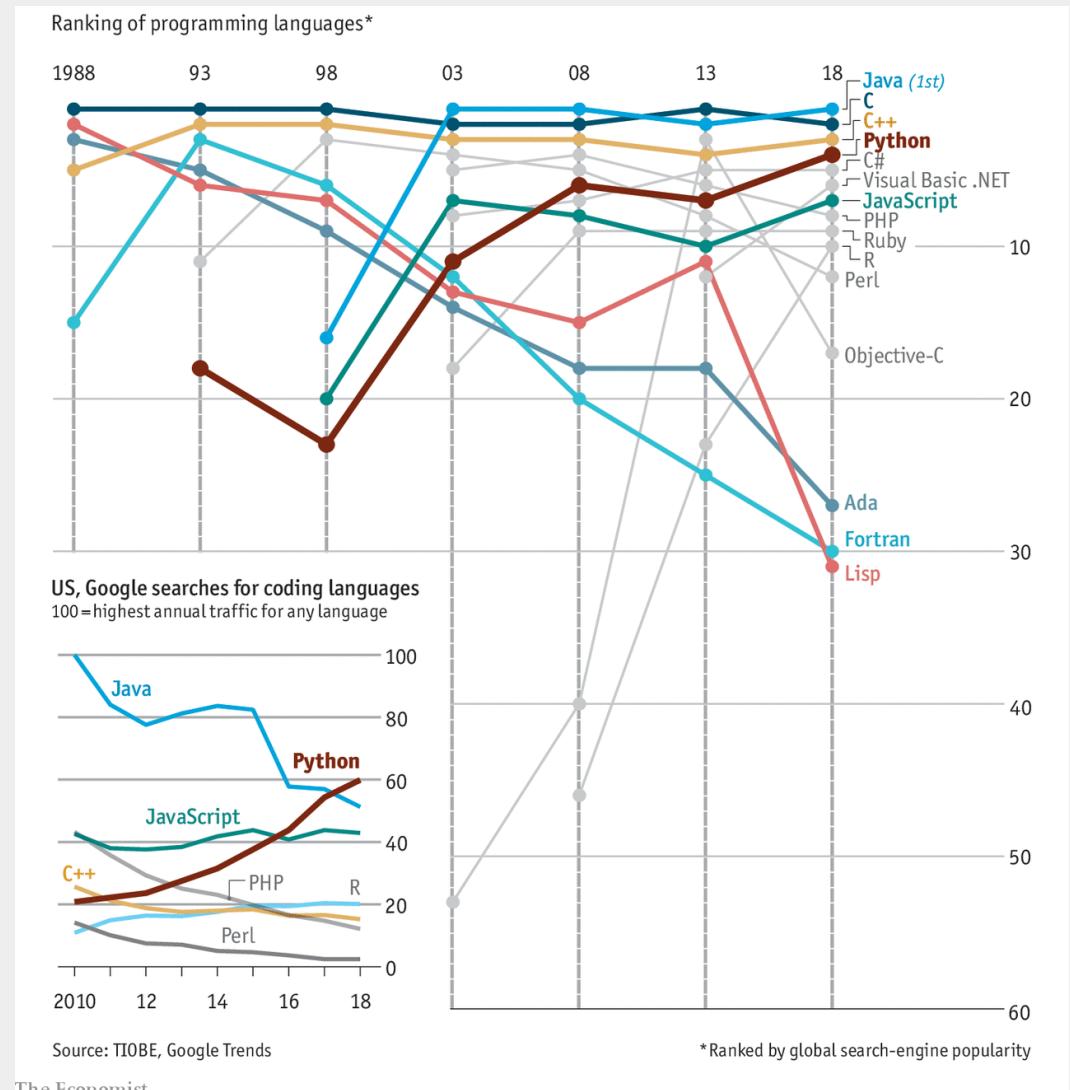


python

- intuitive and readable
- open source
- support C integration for performance
- packages designed for science:
 - scipy
 - statsmodels
 - numpy (computation)
 - sklearn (machine learning)



<https://www.economist.com/graphic-detail/2018/07/26/python-is-becoming-the-worlds-most-popular-coding-language>

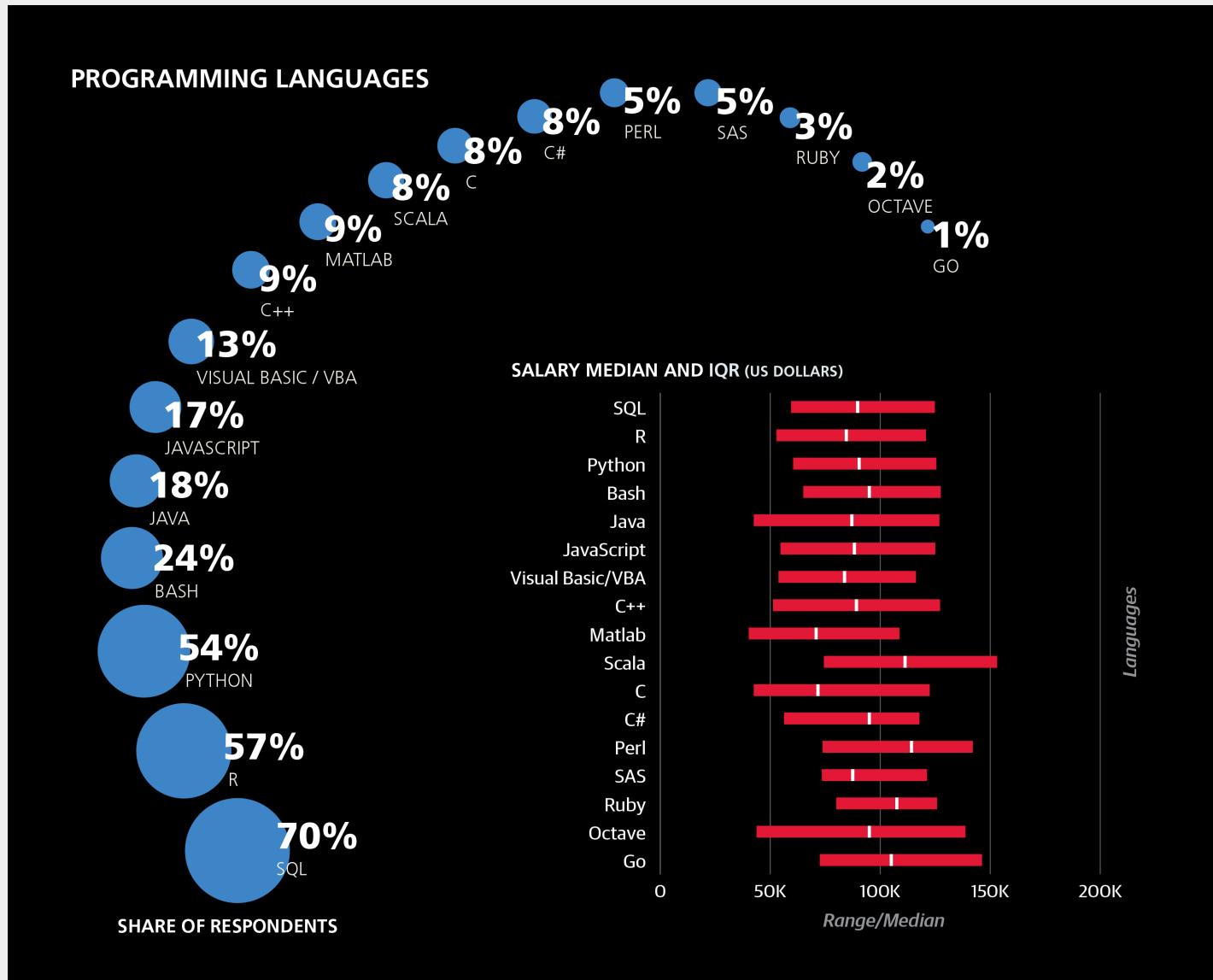


python

- intuitive and readable
- open source
- support C integration for performance
- packages designed for science:
 - scipy
 - statsmodels
 - numpy (computation)
 - sklearn (machine learning)



<https://www.oreilly.com/ideas/2016-data-science-salary-survey-results>



python

<https://sharmamohit.com/work/tutorials/ucsl/>

series of notebooks designed
for Urban Science students
by Dr. Mohit Sharma (in
consultation with me)

recommended if you are
brand new to python and
coding or are serious about
cleaning up your
fundamentals

<https://sharmamohit.com/work/tutorials/ucsl/>

python

quick bootcamp

recommended if you know some
python or if you know some other
coding language reasonably
proficiently

<https://github.com/fedhere/PyBOOT>

Table of Contents

- [1 Native variable types](#)
 - [1.1 strings, int, floats](#)
 - [1.1.1 print formatting](#)
 - [1.2 bool](#)
 - [1.2.1 if/else statements with bools](#)
 - [1.2.2 concatenating bool statements](#)
 - [1.2.3 math with bools](#)
 - [1.3 lists](#)
 - [1.4 dictionaries](#)
- [2 IDE other than jupyter notebooks](#)
 - [2.1 python](#)
 - [2.2 ipython](#)
 - [2.3 execute python from the shell](#)
- [3 Numpy types](#)
- [4 numpy arrays](#)
- [5 PART 2: Slicing, Broadcasting, and math operators on arrays and lists](#)
 - [5.1 operations with arrays](#)
 - [5.2 slicing](#)
- [6 PART 3: Functions](#)
- [7 file IO](#)
- [8 PART 4: multi dimensional arrays](#)
- [9 Part 5: iterators - for loops, enumerate, and list comprehensions](#)
 - [9.1 for loops](#)
 - [9.2 enumerate](#)
 - [9.3 list comprehension](#)
- [10 PART 6: matplotlib](#)
 - [10.1 setting up pylab plotting](#)
 - [10.2 figures and axis objects and simple plots](#)
 - [10.3 plotting errorbars](#)
 - [10.4 plotting 2D arrays](#)

```
from __future__ import print_function, division
# importing this to make code python2&3 compatible
# overwrites the default print to require parenthesis
# overwrites the default / (division operator) behavior
# so division of 2 integers returns a float
```

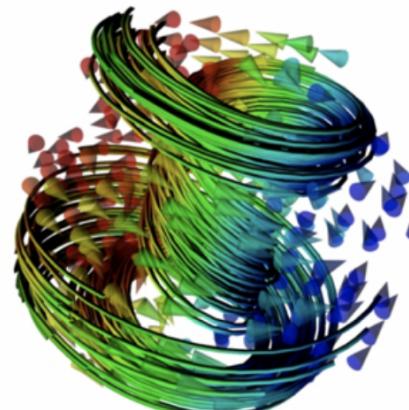
python

online book

<https://www.southampton.ac.uk/~fangohr/training/python/pdfs/Python-for-Computational-Science-and-Engineering.pdf>

Introduction to

Python for Computational Science and Engineering
(A beginner's guide)



Hans Fangohr
Faculty of Engineering and the Environment
University of Southampton

September 7, 2015

python

[PEP8](#): Python Enhancement Proposals 8

“This document gives coding conventions for the Python code comprising the standard library in the main Python distribution.”

*Indentation, Tabs vs Spaces, Maximum Line Length,
Blank Lines, Source File Encoding, Imports,
Whitespace in Expressions and Statements , Imports,
Comments Bookeeping, Naming*

Jupyter Notebook

Google Colaboratory

<https://colab.research.google.com/notebooks/welcome.ipynb#>

The screenshot shows the Google Colaboratory interface. At the top, there's a dark header bar with the 'co' logo, the file name 'HelloWorld.ipynb', a star icon, and standard menu options: File, Edit, View, Insert, Runtime, Tools, Help. To the right of the menu are 'COMMENT' and 'SHARE' buttons, and a user profile picture. Below the header is a toolbar with buttons for '+ CODE', '+ TEXT', and cell navigation arrows. The main workspace contains a text cell with the following content:

```
> This is a notebook that prints Hello World. Notebooks are mixes of code and text. We can write code, describe the code purpose, and display the results as outputs or plots within the notebook itself. Thus notebooks are excellent for prototyping, writing tutorials and reproducible code, and ... delivering homework.
```

Below this is a code cell with the following content:

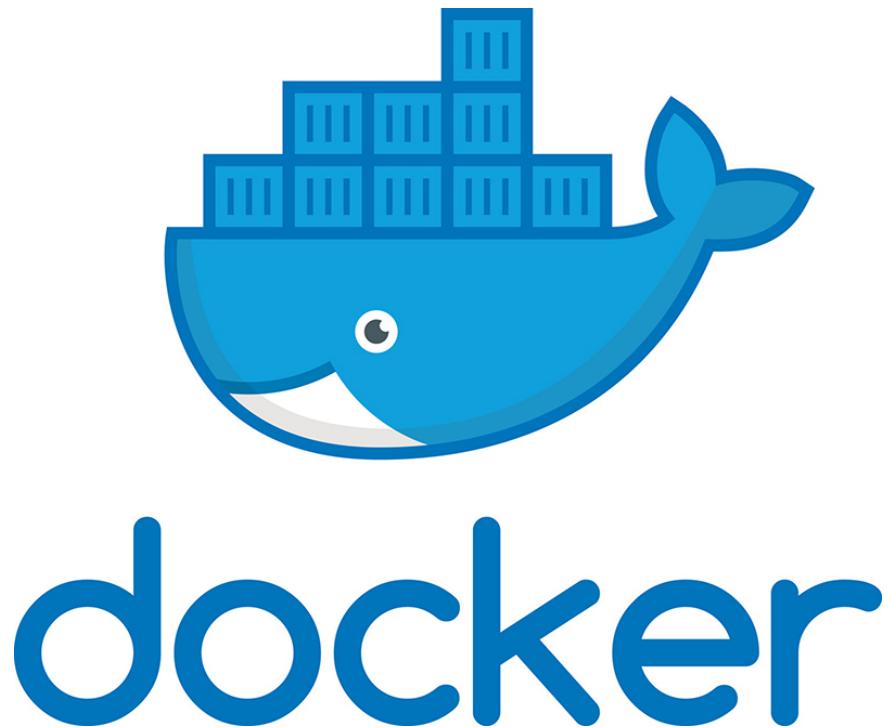
```
▶ print("Hello World")
```

The output of the code cell is:

```
>Hello World
```

Jupyter Notebook

local setup



install docker image
from my account here

Jupyter Notebook local setup



handle your own
installation with
python or anaconda
(or whatever else on
linux and windows)
but make sure results
are reproducible on
google colab

stackoverflow *for when you need help*

<https://stackoverflow.com/>

you can ask coding questions,
installation questions, colab
questions...

How to type list comprehensions

[Ask Question](#)

I have the following list comprehensions in Python:

0

```
from typing import cast
# everything is fine
print([value for value in [1, 2, 3, 4]])
# on the first "value": Expression type contains "Any" (has type "List[Any]")
print("{}".format([value for value in [1, 2, 3, 4]]))
# on the "cast": Expression type contains "Any" (has type "List[Any]")
print("{}".format([cast(int, value) for value in [1, 2, 3, 4]]))
```

▼

★

Why does using `format` cause Mypy to give me back errors? As you can see, I tried to use casting and it still failed.

[This question](#) looks similar, but my particular case is weird because Mypy seems to be fine as long as I'm not using the `format` function (yet it's always okay with the `print` function).

Is there anything I can do to not have the lines with formatting give me errors? (Or should I just `# type: ignore them?`)

[python](#) [python-3.x](#) [list-comprehension](#) [typing](#) [mypy](#)

stackoverflow *for when you need help*

<https://stackoverflow.com/>

you can ask coding questions,
installation questions, colab
questions...

Multiple output regression or classifier with one (or more) parameters with Python

[Ask Question](#)

▲ I wrote a simple linear regression and decision tree classifier code with Python's Scikit-learn library for predicting the outcome. It works good.

5 ▼ My question is, Is there a way to do this backwards, to predict the best combination of parameter values based on imputed outcome (parameters where accuracy will be the best).

★ Or I can ask like this, is there a classification, regression or some other type of algorithm (decision tree, SVM, KNN, logistic regression, linear regression, polynomial regression...) that can predict multiple outcomes based on one (or more) parameter/s?

I have tried to do this with putting multivariate outcome, but it shows the error:

```
ValueError: Expected 2D array, got 1D array instead:  
array=[101 905 182 268 646 624 465].  
Reshape your data either using array.reshape(-1, 1) if your data has a single feature
```

This is the code that I wrote for regression:

```
import pandas as pd  
from sklearn import linear_model  
from sklearn import tree  
  
dic = {'par_1': [10, 30, 13, 19, 25, 33, 23],  
       'par_2': [1, 3, 1, 2, 3, 3, 2],  
       'outcome': [101, 905, 182, 268, 646, 624, 465]}
```

stackoverflow *for when you need help*

it can be a toxic environment...

<https://stackoverflow.com/>

you can ask coding questions,
installation questions, colab
questions...

Multiple output regression or classifier with one (or more) parameters with Python

[Ask Question](#)

▲ I wrote a simple linear regression and decision tree classifier code with Python's Scikit-learn library for predicting the outcome. It works good.

▼ 5 My question is, Is there a way to do this backwards, to predict the best combination of parameter values based on imputed outcome (parameters where accuracy will be the best).

★ Or I can ask like this, is there a classification, regression or some other type of algorithm (decision tree, SVM, KNN, logistic regression, linear regression, polynomial regression...) that can predict multiple outcomes based on one (or more) parameter/s?

I have tried to do this with putting multivariate outcome, but it shows the error:

```
ValueError: Expected 2D array, got 1D array instead:  
array=[101 905 182 268 646 624 465].  
Reshape your data either using array.reshape(-1, 1) if your data has a single feature
```

This is the code that I wrote for regression:

```
import pandas as pd  
from sklearn import linear_model  
from sklearn import tree  
  
dic = {'par_1': [10, 30, 13, 19, 25, 33, 23],  
       'par_2': [1, 3, 1, 2, 3, 3, 2],  
       'outcome': [101, 905, 182, 268, 646, 624, 465]}
```

Science and Data Science
Falsifiability
Reproducibility

key concepts

homework

1

- make an account on GitHub if you do not have one yet
- Create a repository called DSPS_<firstinitialLastname>
- upload your repo URL on canvas as HW 1

2 homework

- make an account on GitHub if you do not have one yet
- Create a repository called DSPS_<firstinitialLastname>

<https://github.com/fedhere/DSPS/tree/master/HW1>

Jeff Leek & Rodger Peng.
2015,
What is the Question?

<http://fbb.space/dsps/The%20Research%20Question-2015-Leek-1314-5.pdf>

reads

the original link:

<https://science.sciencemag.org/content/347/6228/1314.summary>
is link nees access to science magazine, but ou can use the link above
which is the same file)

STATISTICS

What is the question?

Mistaking the type of question being considered is the most common error in data analysis

2

By Jeffery T. Leek and Roger D. Peng

Karl Popper, J. 1934,

The Logic of Scientific Discovery

<http://strangebeautiful.com/other-texts/popper-logic-scientific-discovery.pdf>

Claerbout, J. 1990,

**Active Documents and Reproducible Results,
Stanford Exploration Project Report, 67, 139**

http://sepwww.stanford.edu/data/media/public/docs/sep67/jon2/paper_html/

additional reading