

# foundations of data science for everyone

I: what is data science 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



this slide deck

[https://slides.com/federicabianco/fds\\_01](https://slides.com/federicabianco/fds_01)

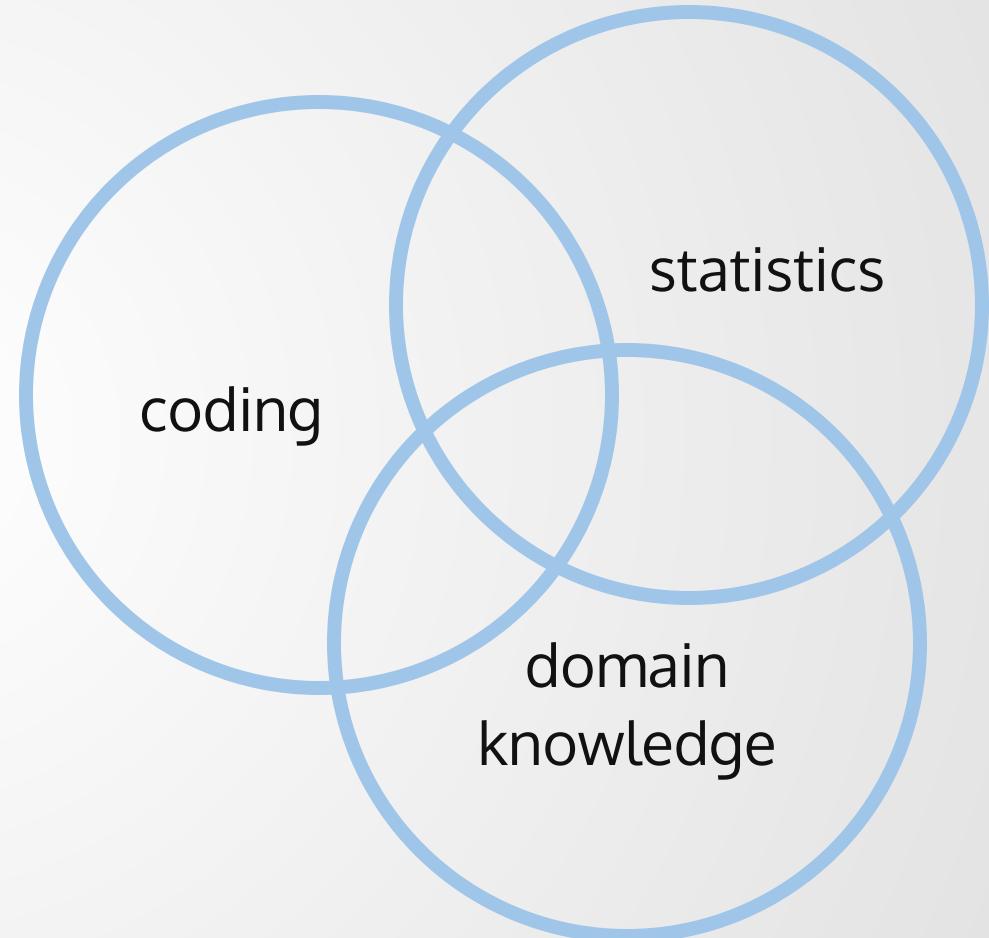
# 1

what is data science?

# foundations of data science for everyone

Data Science: the field of studies that deals with the extraction of information from data within a domain context to enable interpretation and prediction of phenomena.

This includes development and application of statistical tools and machine learning and AI methods



# foundations of data science for everyone

Data Science: the field of studies that deals with the extraction of information from data within a domain context to enable interpretation and prediction of phenomena.

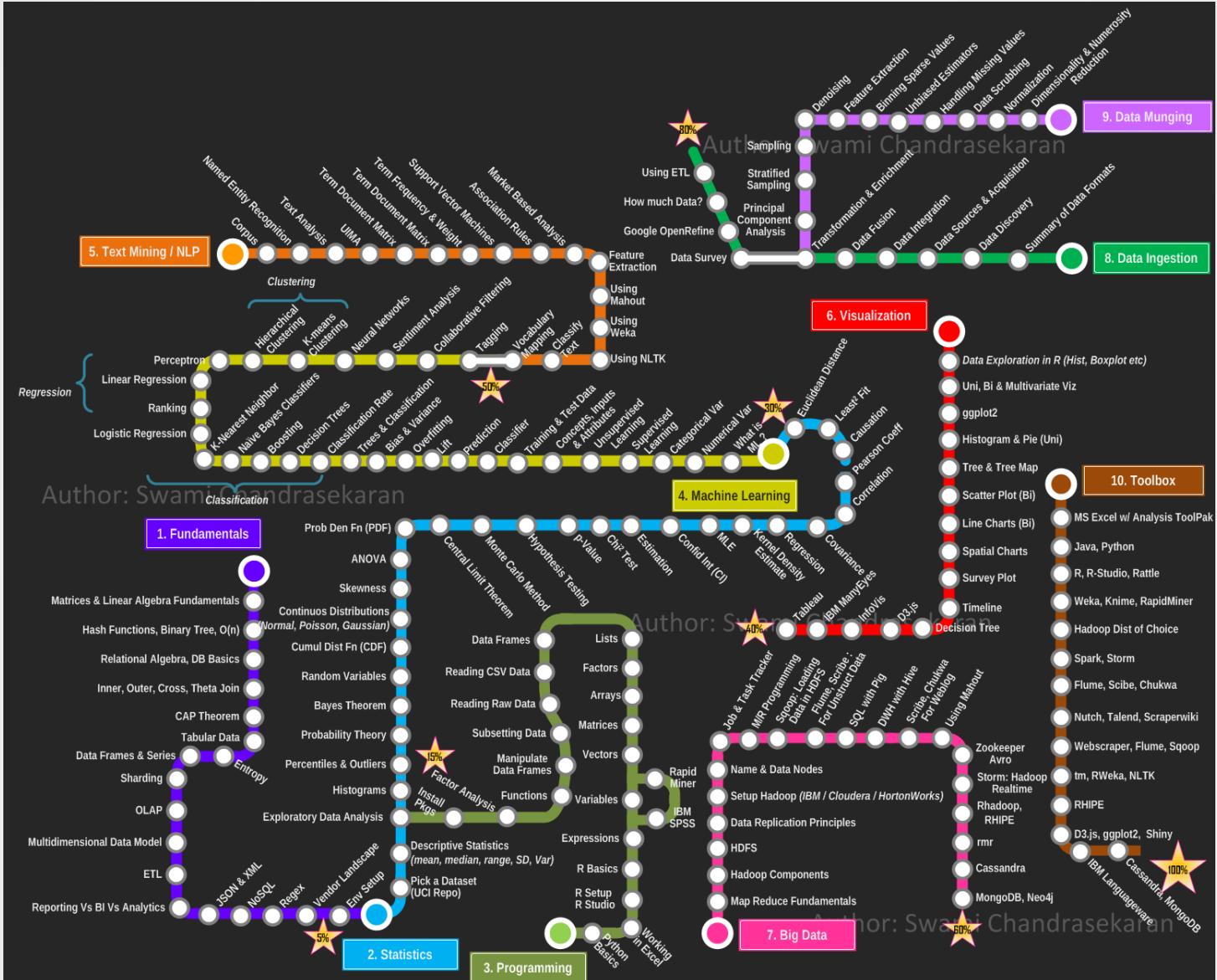
This includes development and application of statistical tools and machine learning and AI methods

Artificial Intelligence:  
enable machines to make decisions without being explicitly programmed

Machine Learning:  
machines learn directly from data and examples

Deep Learning  
(Neural Networks)

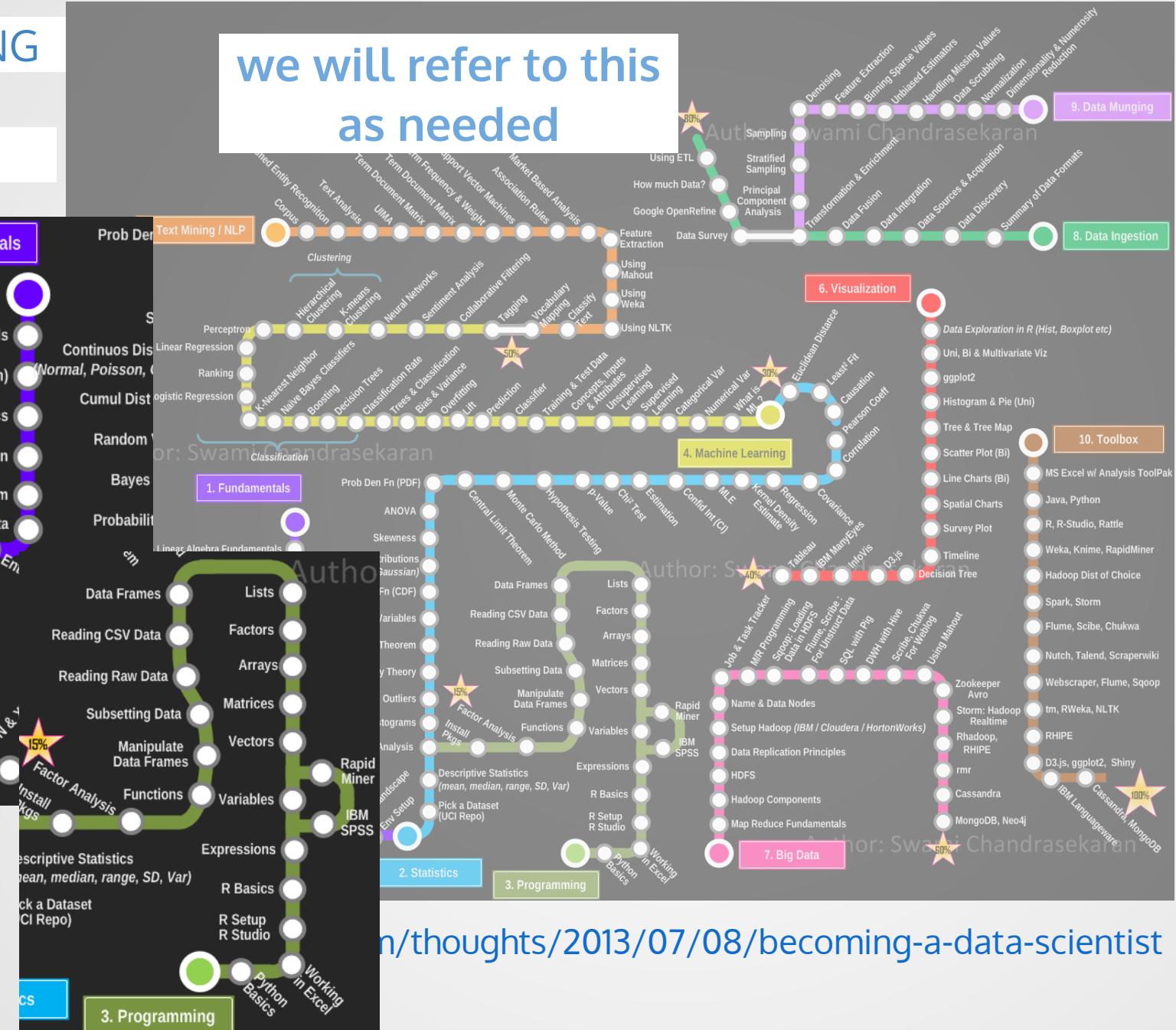


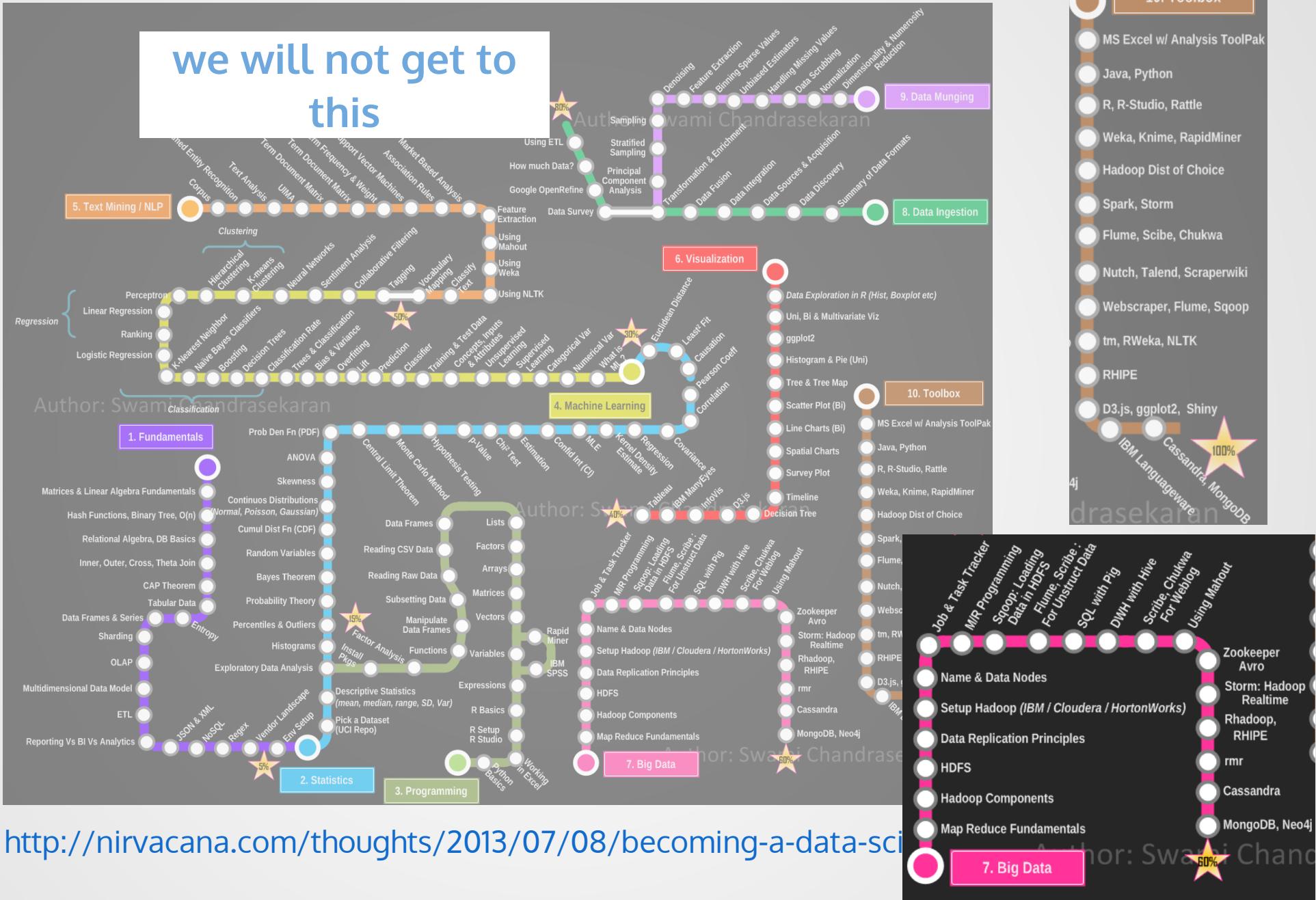


<http://nirvacana.com/thoughts/2013/07/08/becoming-a-data-scientist>

# PROGRAMMING

# STATISTICS





# PROGRAMMING

# STATISTICS

# DATA INGESTION

# DATA MUNGING

# NLP

# MACHINE LEARNING

# VISUALIZATION



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

# PROGRAMMING

# STATISTICS

# DATA INGESTION

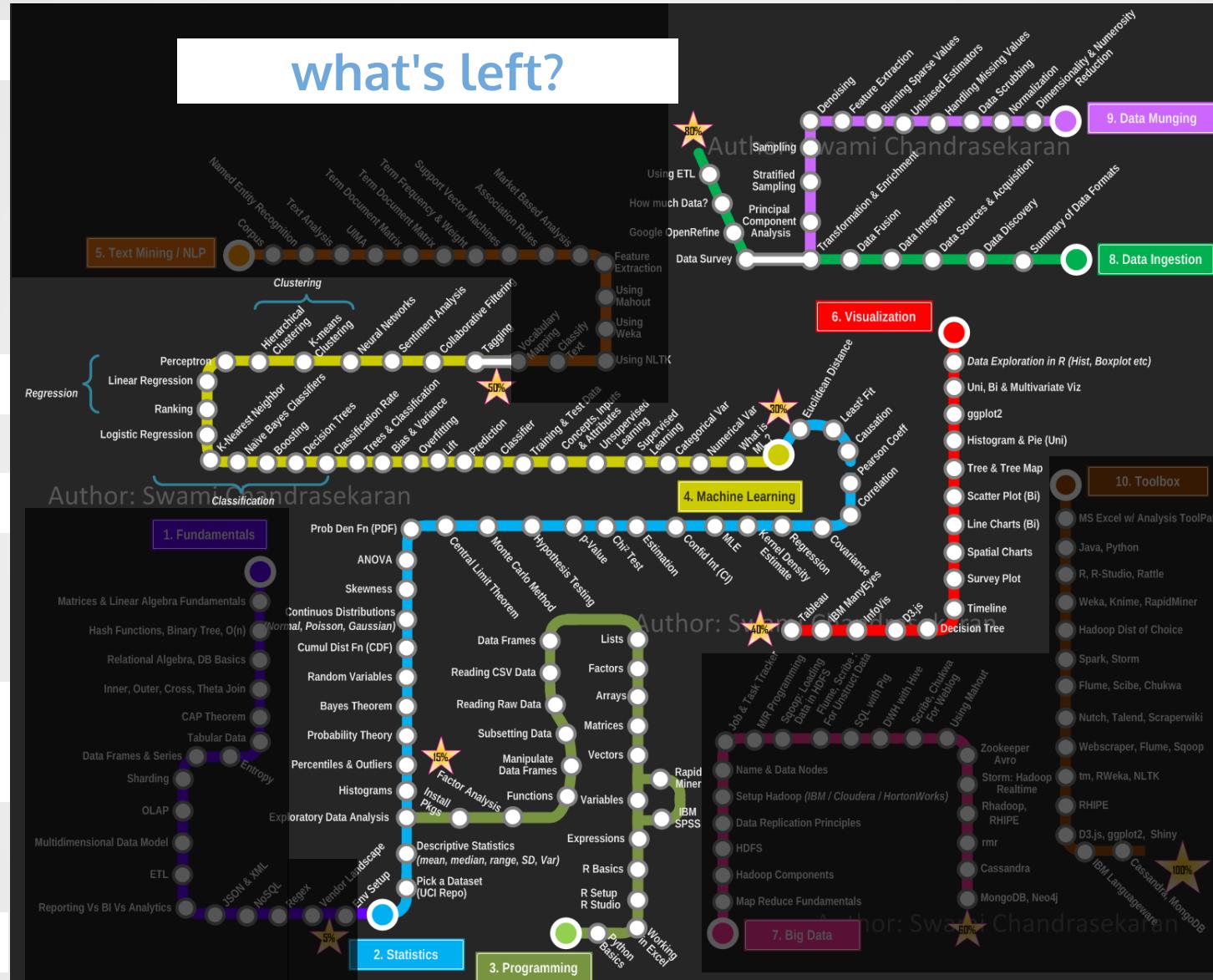
# DATA MUNGING

# MACHINE LEARNING

# VISUALIZATION

python

probability  
distributions  
p-values  
uncertainties



# PROGRAMMING

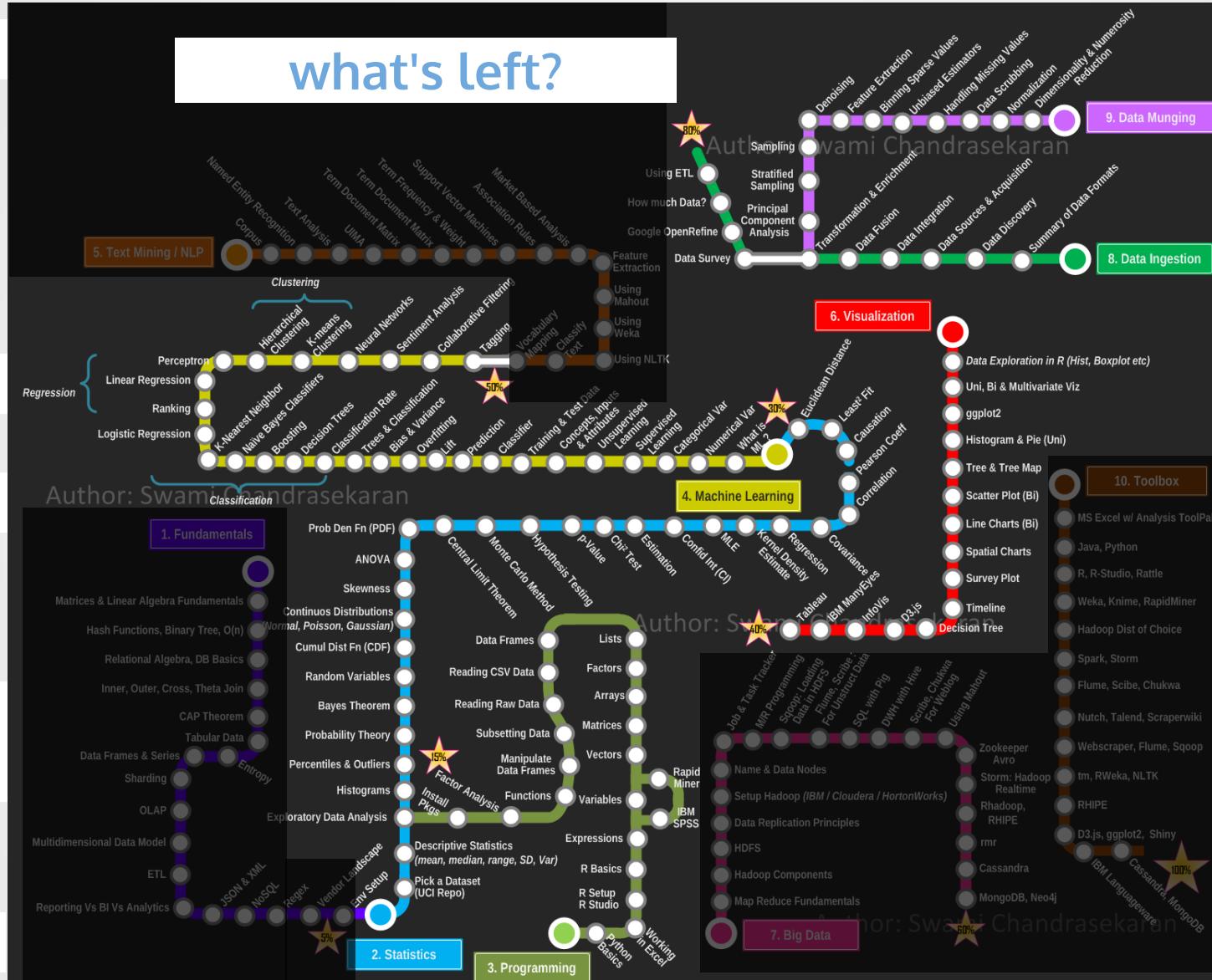
# STATISTICS

# DATA INGESTION

# DATA MUNGING

# MACHINE LEARNING

# VISUALIZATION



# python

probability  
distributions  
p-values  
uncertainties

regression  
(linear, template)

classification

(trees, neural  
networks)

clustering

# 2

some administrative stuff

# Syllabus

## syllabus

### 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

## syllabus

- 5% pre-class questions
- 15% class participation (ask questions, contribute in breakouts!!)
- 25% homework
- 15% midterm
- 30% final

# Syllabus

## syllabus

- 5% pre-class questions
- 15% class participation (ask questions, contribute in breakouts!!)
- 25% homework
- 15% midterm
- 30% final

*from beginning of class to 5 minutes past  
(be on time!)*

*questions on previous class material and  
reading assignments*

# Syllabus

## syllabus

- 5% pre-class questions
- 15% class participation (ask questions, contribute in breakouts!!)
  - ask questions
- 25% homework
  - answer questions
- 15% midterm
  - get up and code
- 30% final
  - extra credit assignments

# homework

- 5% pre-class questions
- 15% class participation
- 25% homework
- 15% midterm
- 30% final

Please work in groups of up to 3-5 people on homework as a collaborative projects.

**All members of the group are responsible for the assignment.**

The assignment must be uploaded in every student's repository. Does not have to be identical for all group members, but it has to be just as complete as the one turned in

# Code of Conduct

to ensure a healthy and safe collaborative environment

**Code of Conduct:** Diversity is considered a resource that enriches us culturally and intellectually in this class. No instances of harassment or attempts to marginalize students will be tolerated in my class. Be respectful and collaborate instead of competing. If you have concerns please come talk to me.

1) Read this: [bit.ly/fdsfe\\_coc](https://bit.ly/fdsfe_coc)



2) Answer questions here:  
[https://bit.ly/fdsfe\\_cocform](https://bit.ly/fdsfe_cocform)



3) join the slack with the link you receive after finishing the form



# 3

the tools

# Jupyter Notebook Google Colaboratory

A collaborative platform for python coding

<https://colab.research.google.com>

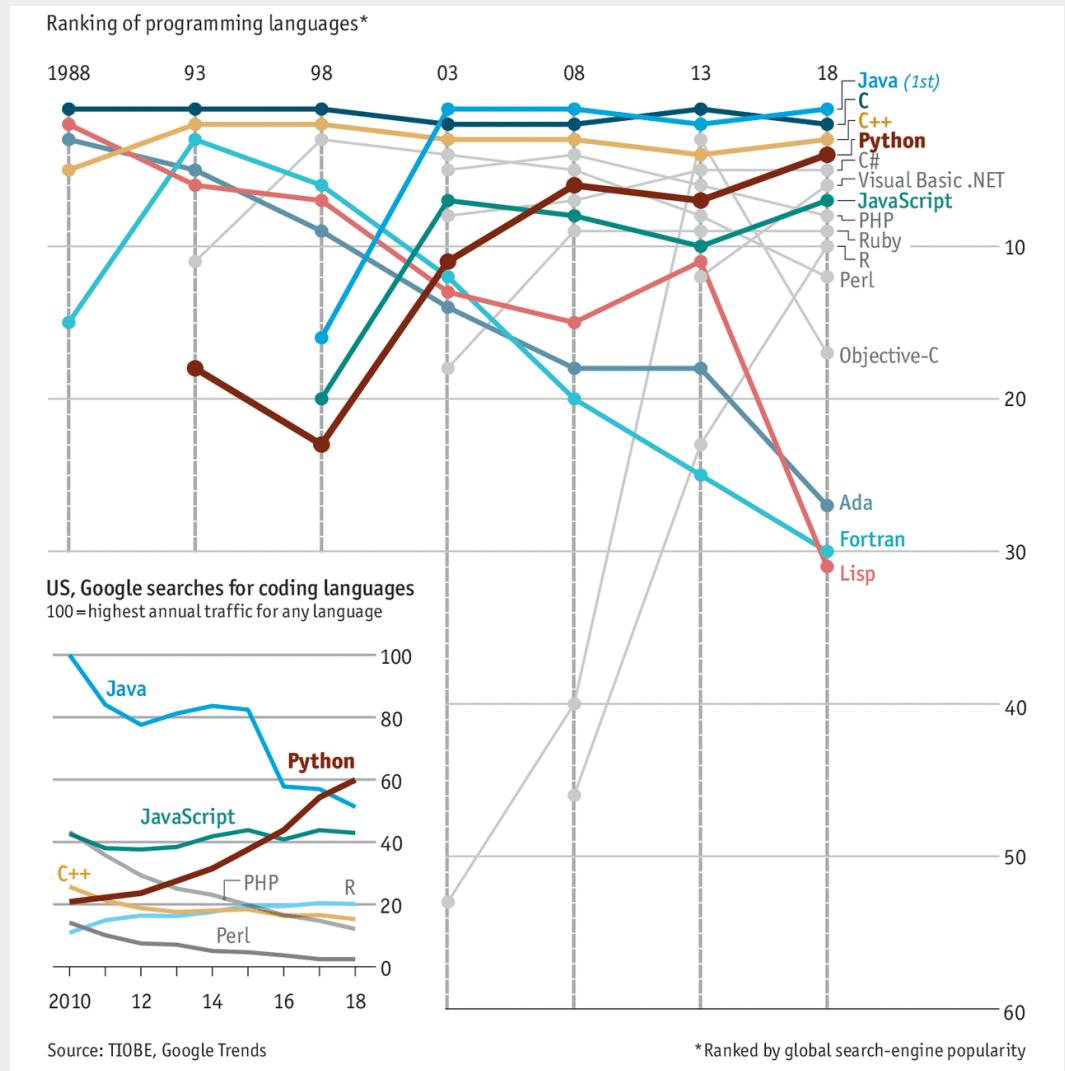


The screenshot shows the Google Colaboratory interface. At the top, there's a header bar with the 'co' logo, the notebook title 'HelloWorld.ipynb' with a star icon, and menu options: File, Edit, View, Insert, Runtime, Tools, Help. To the right of the menu are 'COMMENT' and 'SHARE' buttons, and a user profile picture. Below the header is a toolbar with buttons for 'CODE' and 'TEXT', and arrows for 'CELL'. On the far right of the toolbar are buttons for 'RAM' and 'Disk' status, and an 'EDITING' button. The main area displays a text cell containing the explanatory text: '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 a play button icon and the Python code 'print("Hello World")'. The output of the cell is 'Hello World', preceded by a small arrow icon. The bottom right corner of the interface has a three-dot menu icon.

# *python*

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

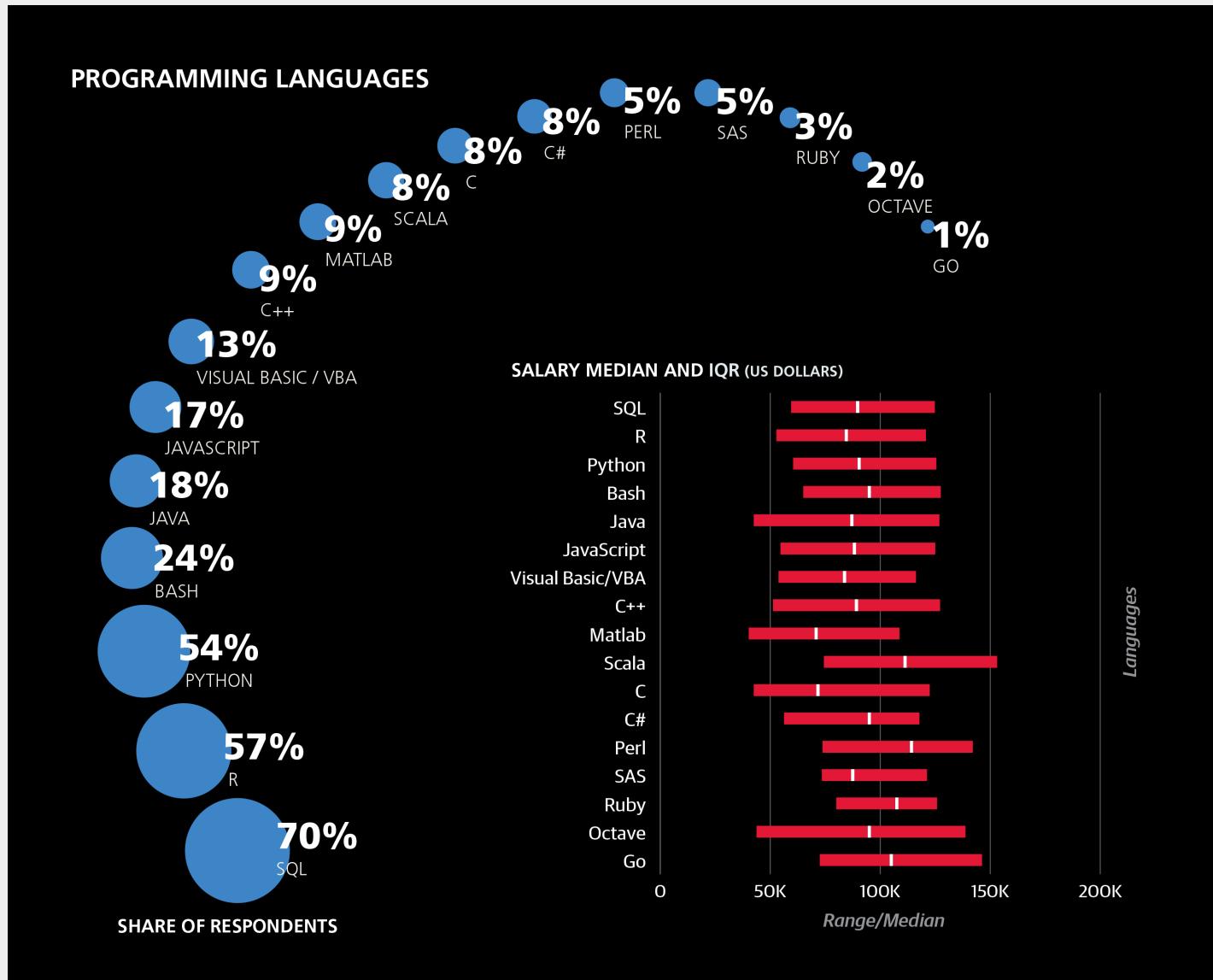
<https://www.economist.com/graphic-detail/2018/07/26/python-is-becoming-the-worlds-most-popular-coding-language>



# *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*

*Resources: Notebook based*

Most compact and rapid:

Xiaolong Li crash course

[https://github.com/fedhere/FDSFE\\_FBianco  
/blob/main/pythoncrashcourse.ipynb](https://github.com/fedhere/FDSFE_FBianco/blob/main/pythoncrashcourse.ipynb)

## **python crash course by Xiaolong Li <https://github.com/xiaolng>**

a very short introduction to python

### basics

- [variables and data types](#) number, boolean, string, list, dictionary
- [conditionals and loops](#) if, for, while
- [functions](#)
- [classes](#)
- [standard library](#)

### packages

- [numpy/scipy](#) numerical/scientific computing
- [pandas](#) data analysis and manipulation
- [matplotlib](#) visualization
- [statsmodels](#) statistical modeling
- [sklearn](#) machine learning
- [keras/tensorflow pytorch](#) deep learning

### resources

# *python*

*Resources: Notebook based*

Slightly more comprehensive -  
python bootcamp

*If there is demand in a couple of weeks  
I will run a live session going over this  
bootcamp*

<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](#)

# *python*

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

*Resources: Notebook based*

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/courses/ucsl/>

# python

*Resources: other*

Free series of videos on the  
Giraffe Academy

[https://www.giraffeacademy.com/program  
ming-languages/python](https://www.giraffeacademy.com/programming-languages/python)

The screenshot shows the Giraffe Academy website with a blue header. The header features the Giraffe Academy logo and a 'Courses' link. The main content area has a white background. On the left, there's a sidebar with a list of Python topics. The main area displays the first video in the series, titled 'Python'. The video thumbnail is blue and yellow, featuring the word 'PYTHON' in large letters. Below the thumbnail, the word 'introduction' is written in large, bold, white letters. At the bottom of the video card, there's a 'Watch on YouTube' button.

Giraffe Academy

Courses

Python

- Installation
- Hello World
- Drawing A Shape
- Variables
- Strings
- Math
- Getting User Input
- Creating A Calculator
- Building A Mad Libs Game
- Lists
- List Functions
- Tuples
- Functions
- Return Statements
- If Statements
- If Statements (Cont)
- Better Calculator
- Dictionaries
- While Loops
- Guessing Game
- For Loops
- Exponent Function
- 2d Arrays & Nested Loops
- Building A Translator

# Python

Lesson 1

Author :

Last Updated : October, 2017

Introduction | Python | Tutorial 1

**PYTHON**  
PROGRAMMING LANGUAGE

Watch later Share

introduction

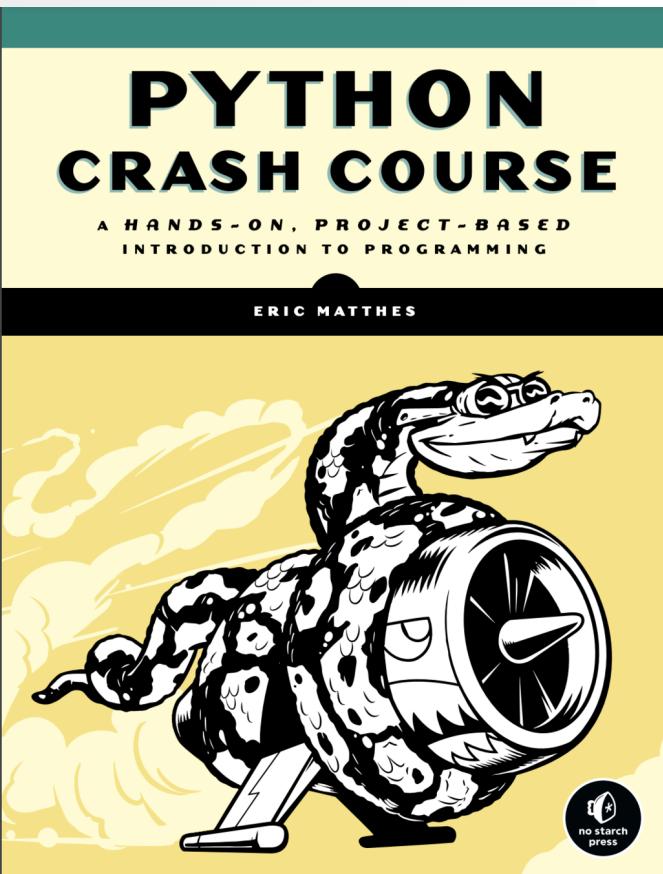
Watch on YouTube

## Python Overview

# Resources

many books, on [github](#) you can find links to the PDFs... but we will do things a bit differently

O'REILLY®



Joel Grus

Think Python

How to Think Like a Computer Scientist

Version 2.0.17

# Resources

Jake Vanderplas is a physicist-data scientists  
[https://www.academia.edu/40917232/Python\\_Data\\_Science\\_Handbook](https://www.academia.edu/40917232/Python_Data_Science_Handbook)

Just as before we stretched or broadcasted one value to match the shape of the other, here we've stretched *both* *a* and *b* to match a common shape, and the result is a two-dimensional array! The geometry of these examples is visualized in Figure 2-4.<sup>1</sup>

$$\text{np.arange}(3)+5 \quad \begin{array}{|c|c|c|} \hline 0 & 1 & 2 \\ \hline \end{array} + \begin{array}{|c|c|c|} \hline 5 & 5 & 5 \\ \hline \end{array} = \begin{array}{|c|c|c|} \hline 5 & 6 & 7 \\ \hline \end{array}$$

$$\text{np.ones}((3, 3))+\text{np.arange}(3) \quad \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array} + \begin{array}{|c|c|c|} \hline 0 & 1 & 2 \\ \hline 0 & 1 & 2 \\ \hline 0 & 1 & 2 \\ \hline \end{array} = \begin{array}{|c|c|c|} \hline 1 & 2 & 3 \\ \hline 1 & 2 & 3 \\ \hline 1 & 2 & 3 \\ \hline \end{array}$$

$$\text{np.ones}((3, 1))+\text{np.arange}(3) \quad \begin{array}{|c|c|c|} \hline 0 & 0 & 0 \\ \hline 1 & 1 & 1 \\ \hline 2 & 2 & 2 \\ \hline \end{array} + \begin{array}{|c|c|c|} \hline 0 & 1 & 2 \\ \hline 0 & 1 & 2 \\ \hline 0 & 1 & 2 \\ \hline \end{array} = \begin{array}{|c|c|c|} \hline 0 & 1 & 2 \\ \hline 1 & 2 & 3 \\ \hline 2 & 3 & 4 \\ \hline \end{array}$$

Figure 2-4. Visualization of NumPy broadcasting



Jake VanderPlas

# *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*

# *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*

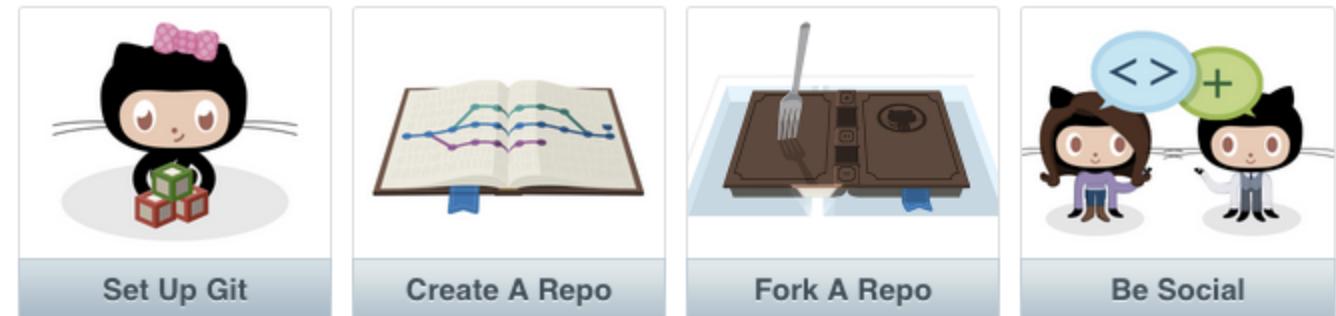


allows effective collaboration

<https://github.com>

**collaboration tool**

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



# *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]}
```

# 1

some administrative stuff  
(cont'd)

# homework

- 5% pre-class questions
- 15% class participation
- 25% homework
- 15% midterm
- 30% 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

homework are assigned as skeleton notebooks with missing code

You will have to insert the code to get the correct output

You should then discuss the results you got (e.g. comment on a plot

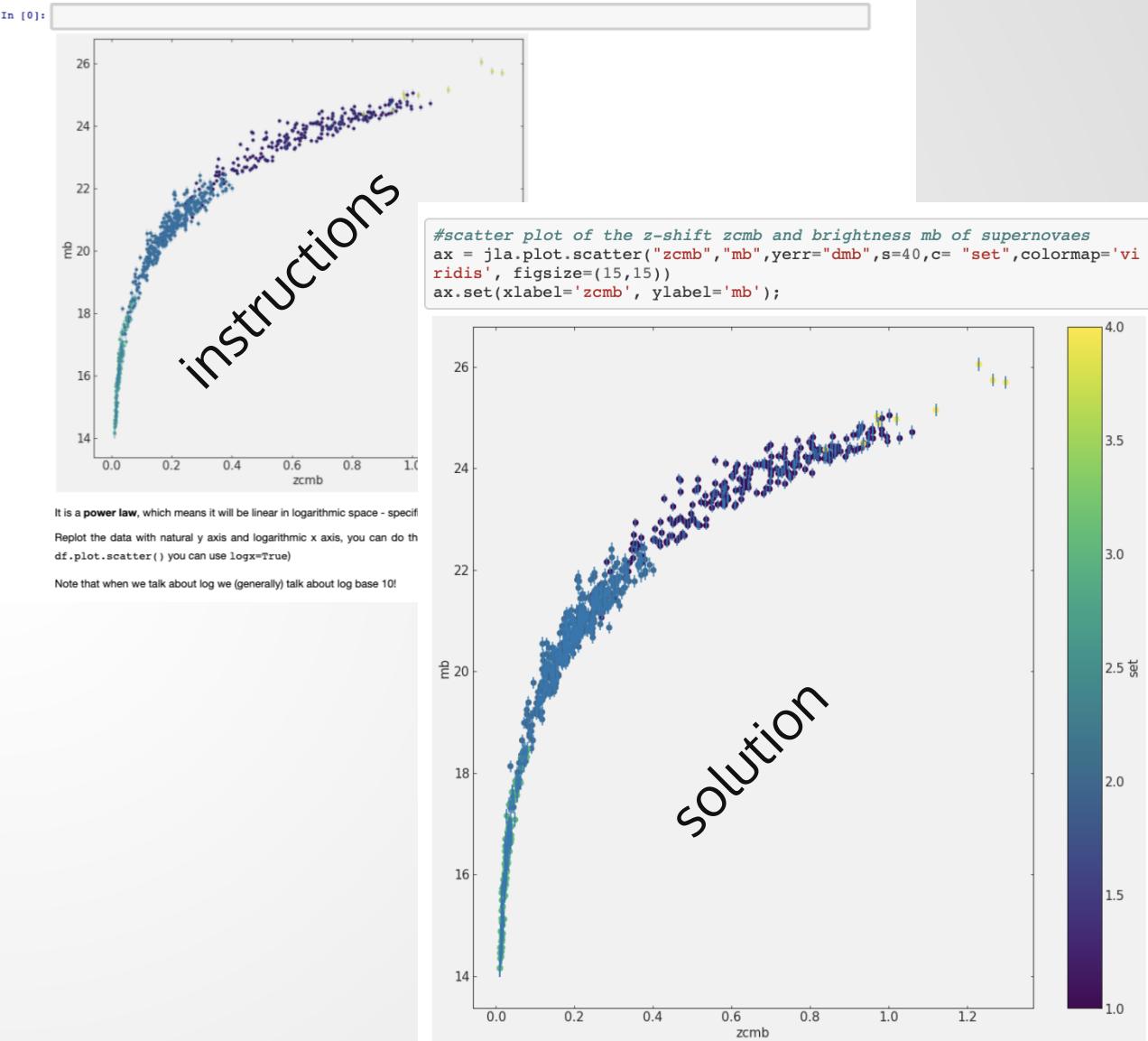
you will be graded on

**(80% of the grade)**

- 1) rendered plots (does it show what it should)
- 2) plot captions (can you interpret what it shows)
- 2) obtaining "correct" numbers where needed
- 3) interpreting each result you get

- **5% pre-class questions**
- **15% class participation**
- **25% homework**
- **15% midterm**
- **30% final**

The target variables for our analysis are redshift and brightness: "zcmb" and "mb". This is an exercise about fitting lines to data. Why does it look like there is a linear relationship between them? How can the relationship look? Plot the "zcmb" vs "mb". Include the y uncertainty which is reported in "dmb". Plot each data point with a different color based on the survey it comes from. The survey data is indicated by the variable "set". To do it you can use the dataframe plotting methods (df.plot.scatter()), or pylab: pylab.scatter() or pylab.plot() --asking to plot the data as point ('.')-- but for each the way you relate the color to a column value is a bit different. Stackoverflow is your friend here!



# homework

- 5% pre-class questions
- 15% class participation
- 25% homework
- 15% midterm
- 30% final

A statement **must** be included in the README explaining each team member's contribution (similar to an acknowledgement of contribution you would find in a *Nature* letter see, for example [these contributions](#)).

## nature

### Contributions

All authors contributed to the drafting of the paper. A.R., N.S. and R.C.S. imaged the area around η Car. A.R. and M.E.H. reduced the imaging data. H.E.B. provided images of the echoes that guided our spectroscopic pointings. J.L.P., R.C., R.J.F. and W.F. obtained the spectra and reduced them. A.R. and J.L.P. performed spectral analysis and interpretation. A.R., N.R.W. and F.B.B. performed spectral classification. F.B.B. and K.M. correlated the spectra. A.R., D.L.W. and B.S. modelled the light echo. I.T. and D.M. provided imaging of η Car. F.B.B. and D.A.H. provided the FTS images, and F.B.B. and A.R. reduced them.

# homework

- **5% pre-class questions**
- **15% class participation**
- **25% homework**
- **15% midterm**
- **30% final**

README.md

Partners were Theodore Fessaras, Melvin Tejada, Samuel Matylewicz, and Desi Pilla. We all met up on Saturday to do the lab. I wrote the derivation and data ingestion part on my own the day prior, and compared with the rest of the group when we met up. Desi helped the group a lot with writing the functions. Each of us proofread and error tested his functions and gave feedback.

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](#)).

## ☞ Homework 4:

Data importing and formatting by Victor Ramirez and Liam Kelley.

Astronomy knowledge by Victor Ramirez. Administrative direction by Victor Ramirez

Debugging by Liam Kelley

Mathematical derivation by Shea Fitzgerald

General structure of the code by Shea Fitzgerald

# homework

- 5% pre-class questions
- 15% class participation
- 25% homework
- 15% midterm
- 30% final

**Each student must write a README file for  
their repository  
(20% of the points)**

in the readme you must state in your own words

1. what was this homework about? relate it to what we discussed in class
2. what was the hardest part of the homework for you?
3. what was the easiest part of the homework for you?
4. one new thing that you have learned

# homework

- **5% pre-class questions**
- **15% class participation**
- **25% homework**
- **15% midterm**
- **30% final**

The screenshot shows a GitHub repository page for 'fedhere/FDSfE'. The repository is public and has 1 branch and 0 tags. It was created using Colaboratory and has 8 commits. The README.md file is visible, containing the following text:

```
FDSfE

Repo for Foundations of Data Science for Everyone - class taught at Lincoln University + University of Delaware

This course will teach the basics of data-driven research. Students will acquire basic computational skills, basic knowledge of statistical analysis, error analysis, familiarize with good practices for handling small- and big-data, and the basics of Machine Learning. After this class students should be able to formulate a question, find appropriate data to answer the question, prepare and analyze the data, get an answer, and understand the answer's confidence level. The course will be organized in a modular fashion, with labs and projects assigned to students for group work.
```

instructions will be here

[https://github.com/fedhere/FDSfe\\_FBianco](https://github.com/fedhere/FDSfe_FBianco)

# homework

- **5% pre-class questions**
- **15% class participation**
- **25% homework**
- **15% midterm**
- **30% final**

The screenshot shows a GitHub repository page for 'fedhere/FDSfE'. At the top, there's a search bar with placeholder text 'Search or jump to...', a pull request button, and navigation links for 'Pull requests', 'Issues', 'Marketplace', and 'Explore'. Below the header, the repository name 'fedhere/FDSfE' is displayed with a 'Public' badge. A dropdown menu for the 'main' branch is open. To the right, there are buttons for 'Go to file', 'Add file', and 'Code'. The repository was created by 'fedhere' using Colaboratory, as indicated in the commit history. The commit history shows three files: 'CodeExamples', 'README.md', and 'Resources.md', all created using Colaboratory. The 'README.md' file is expanded, showing its content.

**FDSfE**

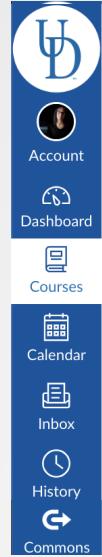
Repo for Foundations of Data Science for Everyone - class taught at Lincoln University + University of Delaware

This course will teach the basics of data-driven research. Students will acquire basic computational skills, basic knowledge of statistical analysis, error analysis, familiarize with good practices for handling small- and big-data, and the basics of Machine Learning. After this class students should be able to formulate a question, find appropriate data to answer the question, prepare and analyze the data, get an answer, and understand the answer's confidence level. The course will be organized in a modular fashion, with labs and projects assigned to students for group work.

help: always available on slack!  
please sign up asap by filling in the form!!  
<https://forms.gle/j4vejz7R2qwwQrdr9>



# homework



22F-GEOG167-010 > Announcements > Welcome to Founda...

6d Student View

2022 Fall

Home

Syllabus

Announcements

Files

Assignments

Grades

My Media

Media Gallery

New Analytics

Quizzes

Welcome to Fou  
Federica Bianco  
All Sections

Dear students,  
Welcome to Lincoln MAT115!  
This is a joint UDel - Lincoln class taught to students from the two Universities simultaneously. This class will teach you the basics of Data Science so that you can apply them across fields and domains of interest. It is part of a larger project to set up certificates in Data Science at both our institutions.

MAT-115... > Announce... > Welcome t...

6d Student View

2022 Fall

Home

Announcements

Assignments

Discussions

Grades

People

Pages

Files

Welcome to Foundations of Data Science for Everyone!  
Federica Bianco  
All Sections

Aug 29 at 5:15pm

Dear students,  
Welcome to Lincoln MAT115 and UDel PHYS167/SPPA167/GEOG167!  
This is a joint UDel - Lincoln class taught to students from the two Universities simultaneously. This class will teach you the basics of Data Science so that you can apply them across fields and domains of interest. It is part of a larger project to set up certificates in Data Science at both our institutions.

- **5% pre-class questions**
- **15% class participation**
- **25% homework**
- **15% midterm**
- **30% final**

of course there is also Canvas, which will be used to give you grades and occasionally post messages

# midterm

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.

- 5% pre-class questions
- 15% class participation
- 25% homework
- 15% midterm
- 30% final

**Midterm... probably in class**

*advantages:* interviews for jobs are often timed

*issues:* working under pressure is not necessarily a required skill but that is why the midterm counts only 15%!

# 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.**

- 5% pre-class questions
- 15% class participation
- 25% homework
- 15% midterm
- 30% final

Final: take home, multiple days.

# Resources

- SLIDES --> <https://slides.com/federicabianco/decks/fdsfe>
- HOMEWORK INSTRUCTIONS --> [github](#)
- RESOURCES --> slack, [github](#)

If notebooks do not display

use

<https://nbviewer.jupyter.org>

[https://github.com/fedhere/FDSfE\\_FBianco](https://github.com/fedhere/FDSfE_FBianco)

# Foundations of Data Science for Everyone - Resources

Instructors: Dr. Federica Bianco

contact: [fbianco@udel.edu](mailto:fbianco@udel.edu)

Synchronous online class

Mon-Wed 3:35-4:50pm

Attendance is required.

[Code of Conduct](#) [Syllabus](#)

Academic Calendar - LU and UD have different calendars. Stay tuned for modifications of the schedule

## Resources

### Class material

- lecture slides will be posted here, hopefully well in advance
- Github homework repository
- computing will be done in the google cloud
- Python Bootcamp material.
- data we are using will be [here](#) or on github

### Python resources

- Mohit's excellent python guide for the CUSP bootcamp
- the zen of python
- PEP8 python guide style
- @Mark\_Graph python cheat-sheet
- @Mark\_Graph Pandas cheat-sheet
- Matplotlib cheat-sheet 1
- Matplotlib cheat-sheet 1

[https://github.com/fedhere/FDSfE\\_FBianco  
/blob/main/Resources.md](https://github.com/fedhere/FDSfE_FBianco/blob/main/Resources.md)

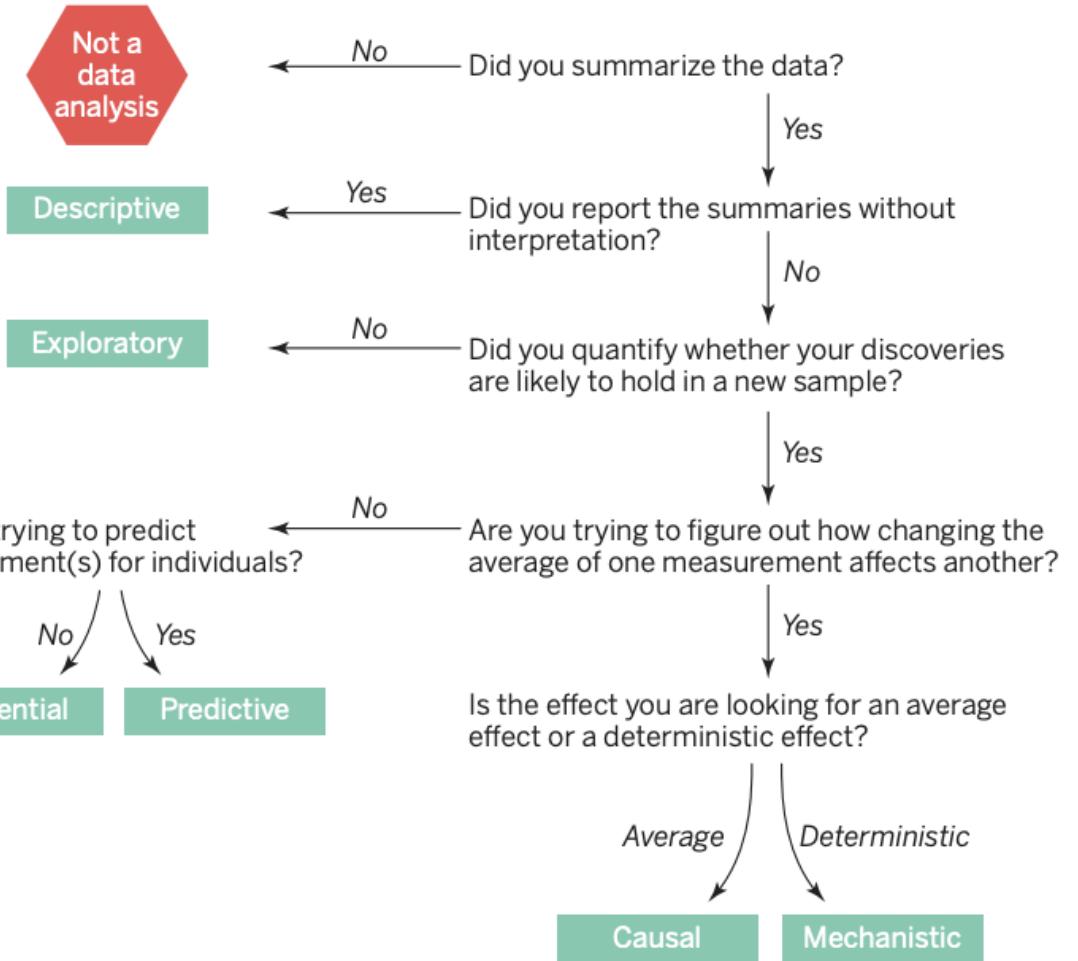
The screenshot shows a Slack interface. On the left is a sidebar with a dark purple header containing the text "Foundations of Data ...". Below the header are sections for "Mentions & reactions", "Channels", "Direct messages", and "Teammates". The "Channels" section is expanded, showing a list of channels including "general", "# hello-my-name-is", "# hw1", "# quiz", "# random", and "# resources", which is highlighted with a blue bar at the bottom. To the right of the sidebar is the main workspace. At the top of the workspace is a header with the text "# resources" and a count of "2 Pinned". Below the header is a message from user "fbianco" at 2:04 PM: "joined #resources along with Farid Qamar." Below this is another message from "fbianco" at 2:05 PM: "Instructors: Federica Bianco @fbianco email: fbianco@udel.edu (but slack is better) Farid Qamar @Farid Qamar email: qamar@udel.edu". Further down, a message from "TA: Willow Fox Fortino" at 2:05 PM is shown: "email: fortino@udel.edu (edited)". At the bottom of the workspace, there are two pinned messages: one from "fbianco" at 2:04 PM about being pinned, and another from "fbianco" at 2:05 PM about pinned class material, which lists the same resources as the GitHub link above.

# Resources

# 4

the steps of data-analysis and inference:  
descriptive and exploratory analysis

Data analysis flowchart





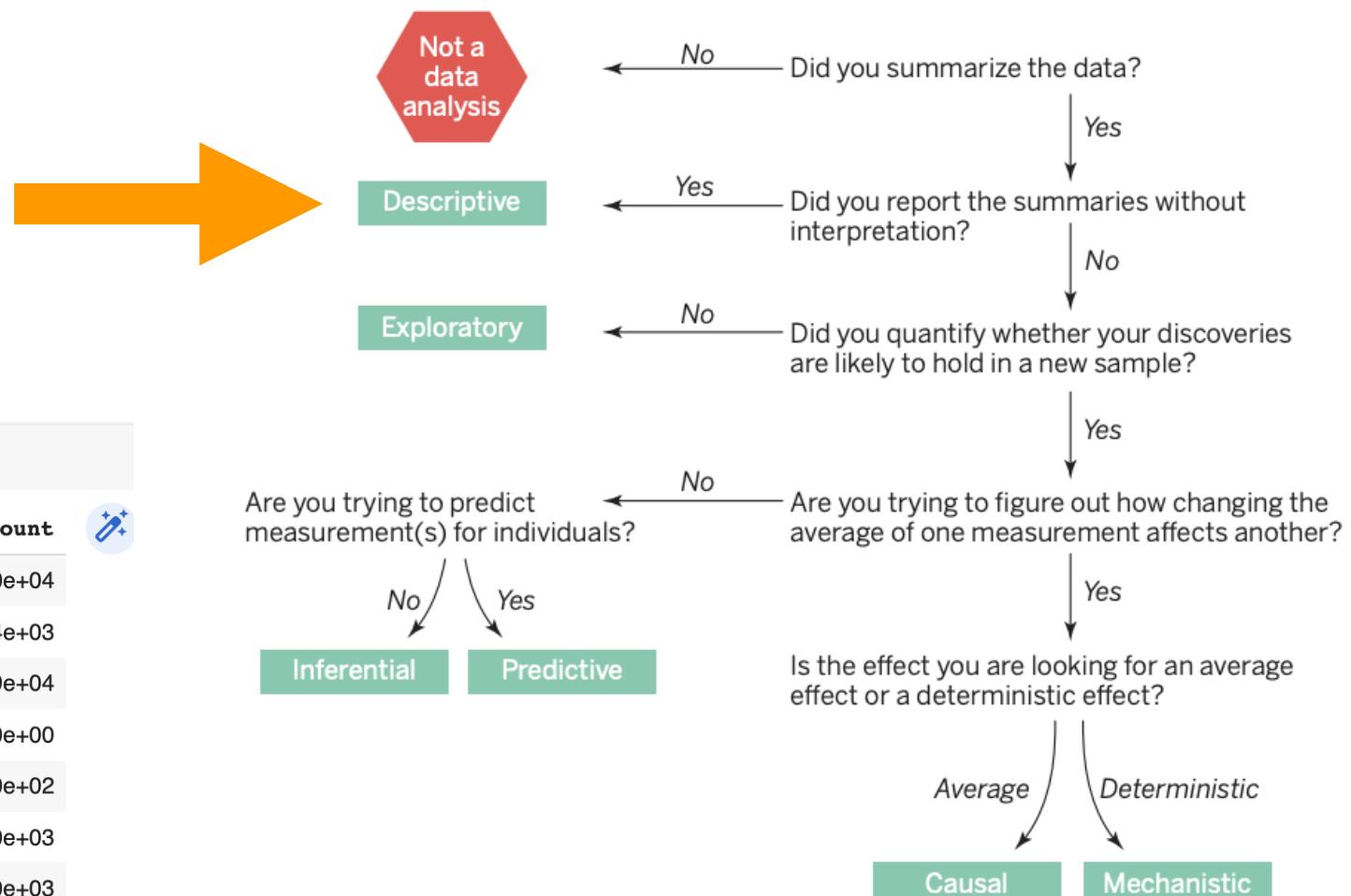
- how is data organized
- is data complete?
- what are the statistical properties of the data

```
1 import pandas as pd  
2 df = pd.read_csv(file_name)  
3 df.describe()
```

videos.describe()

	category_id	views	likes	dislikes	comment_count
count	40949.000000	4.094900e+04	4.094900e+04	4.094900e+04	4.094900e+04
mean	19.972429	2.360785e+06	7.426670e+04	3.711401e+03	8.446804e+03
std	7.568327	7.394114e+06	2.288853e+05	2.902971e+04	3.743049e+04
min	1.000000	5.490000e+02	0.000000e+00	0.000000e+00	0.000000e+00
25%	17.000000	2.423290e+05	5.424000e+03	2.020000e+02	6.140000e+02
50%	24.000000	6.818610e+05	1.809100e+04	6.310000e+02	1.856000e+03
75%	25.000000	1.823157e+06	5.541700e+04	1.938000e+03	5.755000e+03
max	43.000000	2.252119e+08	5.613827e+06	1.674420e+06	1.361580e+06

## Data analysis flowchart



we will look at the statistical properties next week: mean, standard deviation, median, quantiles...



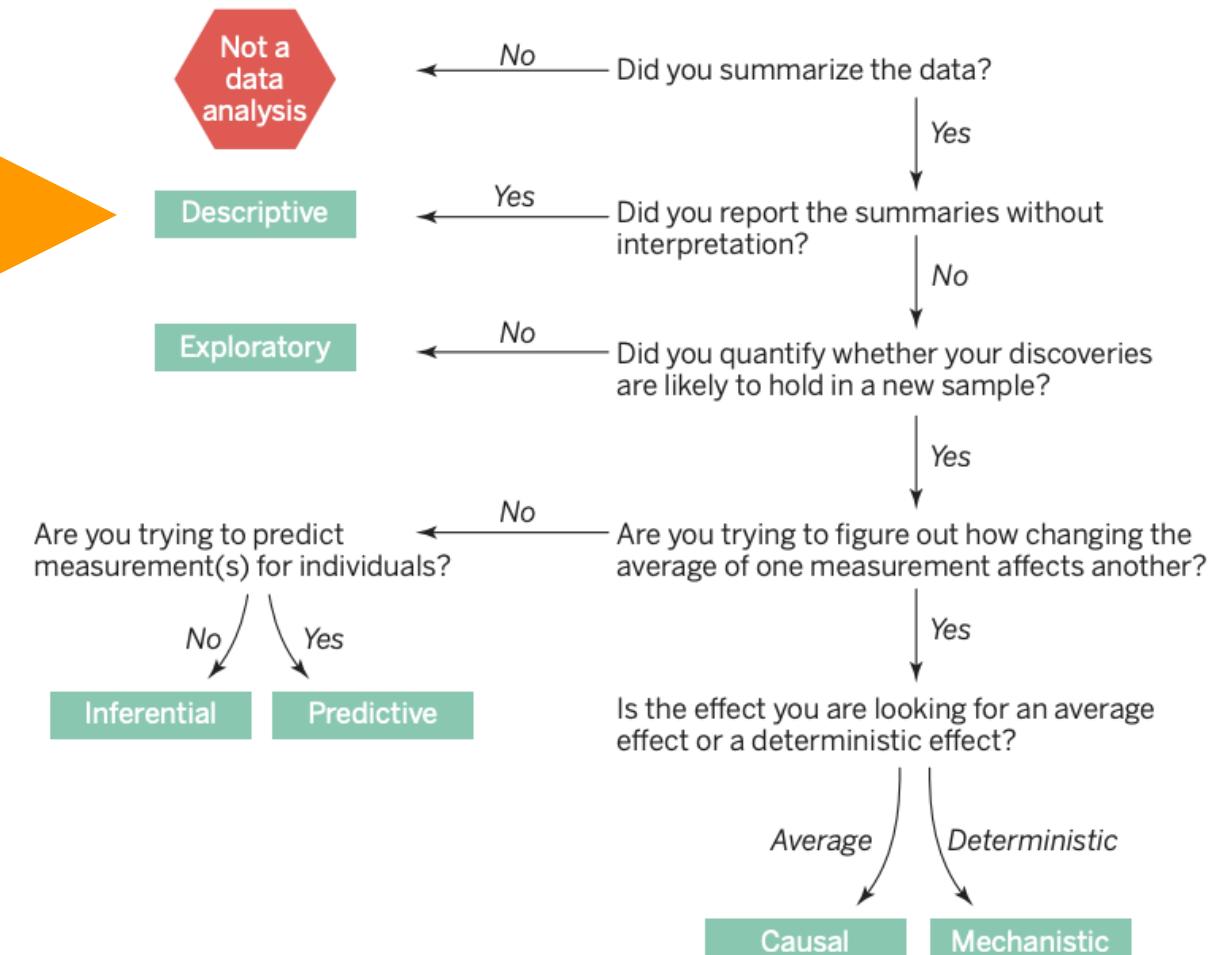
- how is data organized
  - is data complete?
  - what are the statistical properties of the data

```
1 import pandas as pd  
2 df = pd.read_csv(file_name)  
3 df.describe()
```

videos.describe()

	category_id	views	likes	dislikes	comment_count
count	40949.000000	4.094900e+04	4.094900e+04	4.094900e+04	4.094900e+04
mean	19.972429	2.360785e+06	7.426670e+04	3.711401e+03	8.446804e+03
std	7.568327	7.394114e+06	2.288853e+05	2.902971e+04	3.743049e+04
min	1.000000	5.490000e+02	0.000000e+00	0.000000e+00	0.000000e+00
25%	17.000000	2.423290e+05	5.424000e+03	2.020000e+02	6.140000e+02
50%	24.000000	6.818610e+05	1.809100e+04	6.310000e+02	1.856000e+03
75%	25.000000	1.823157e+06	5.541700e+04	1.938000e+03	5.755000e+03
max	43.000000	2.252119e+08	5.613827e+06	1.674420e+06	1.361580e+06

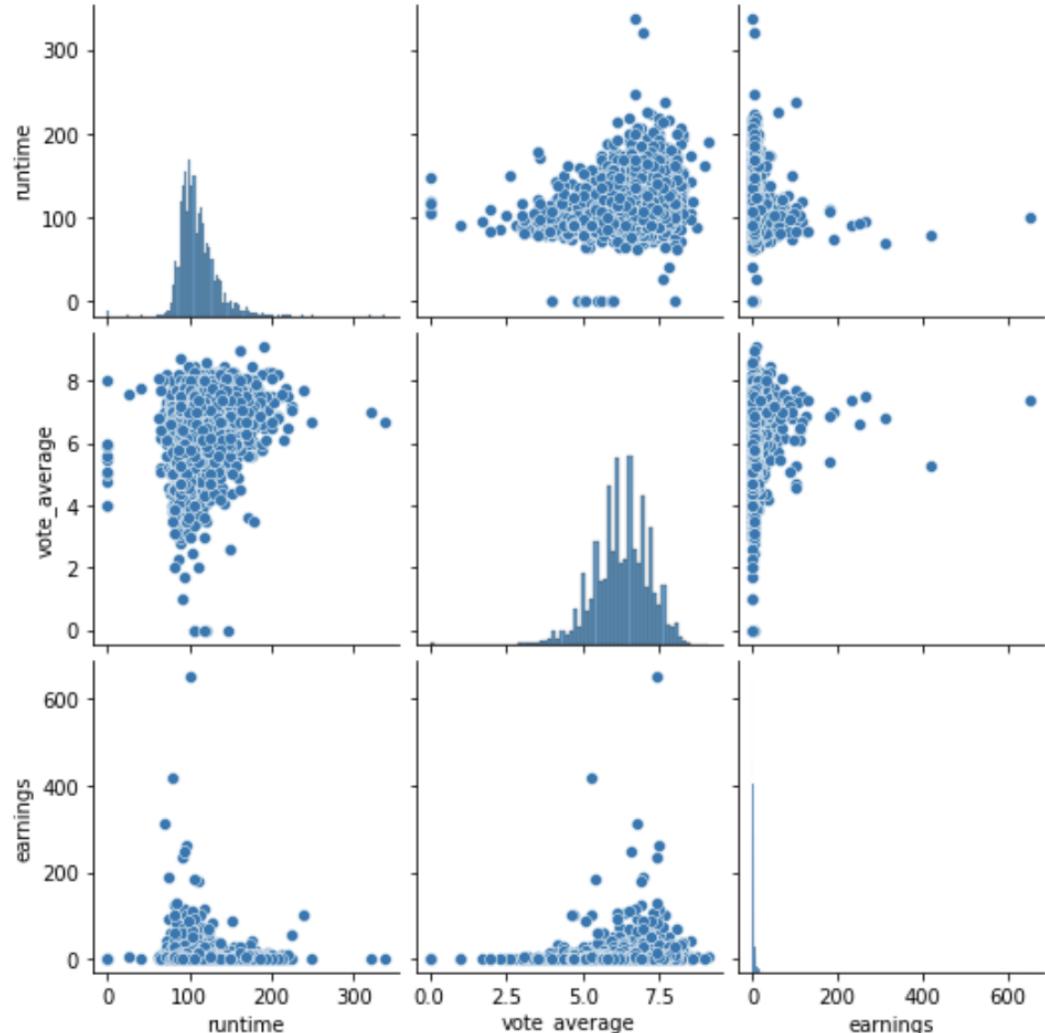
## Data analysis flowchart



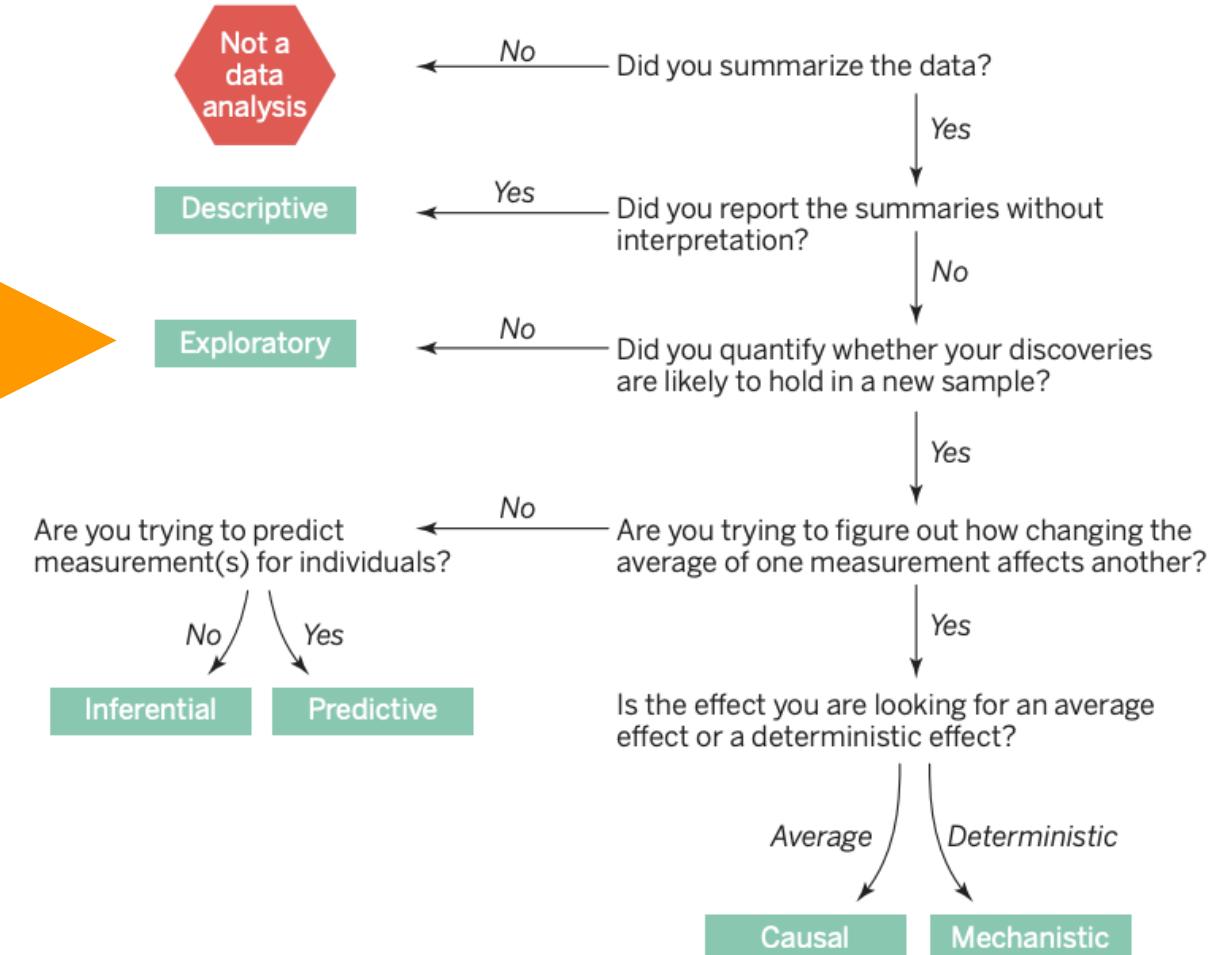
we will look at the statistical properties next week: mean, standard deviation, median, quantiles...



- searching for anomalies, trends, correlations, or relationships between the measurements

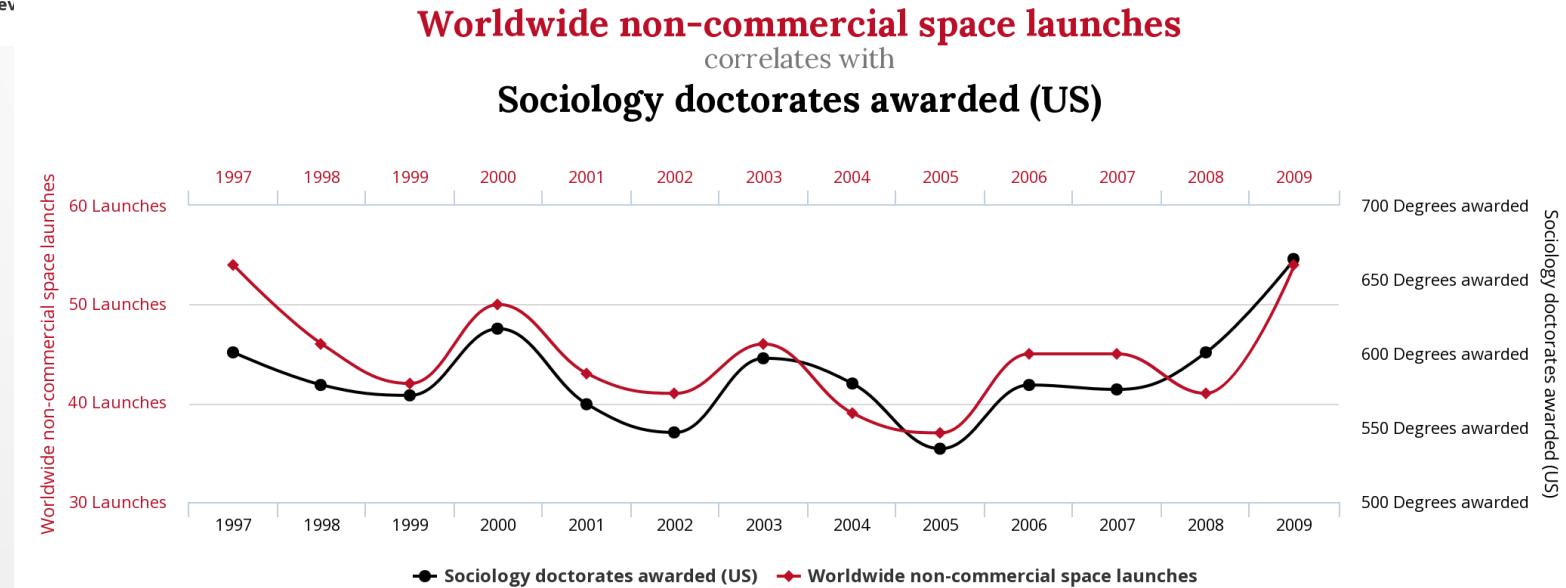
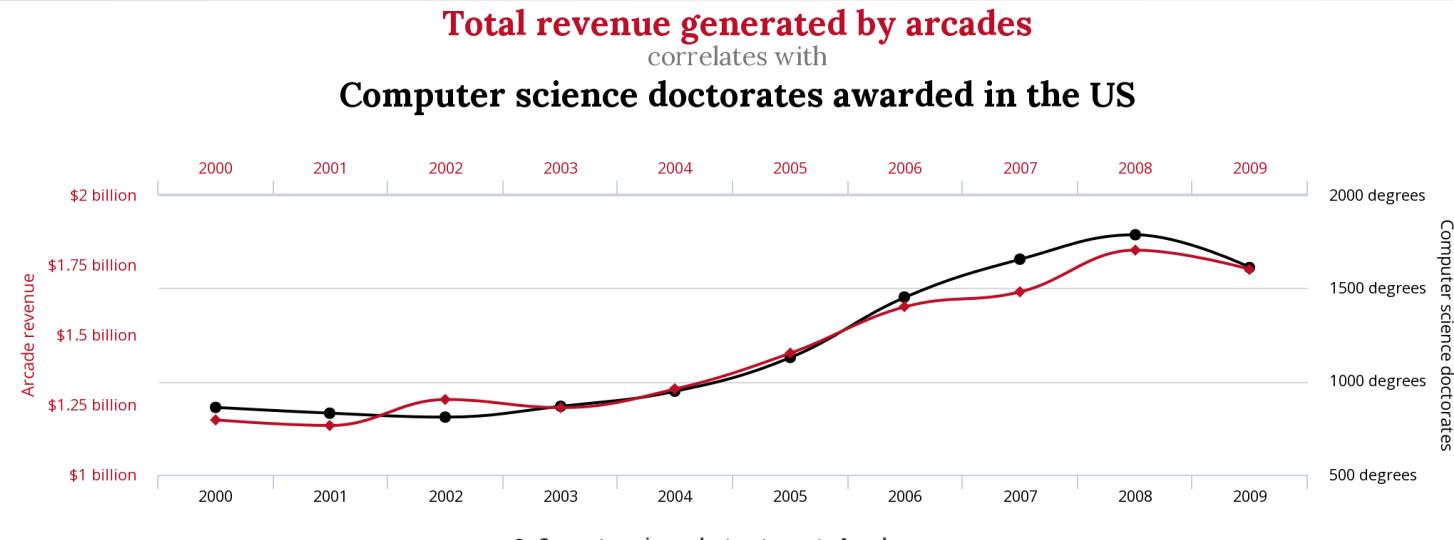


## Data analysis flowchart



An exploratory data analysis builds on a descriptive analysis by searching for discoveries, trends, correlations, or relationships between the measurements to generate ideas or hypotheses.

# correlation



<http://www.tylervigen.com/spurious-correlations>

Pearson's correlation

$$r_{xy} = \frac{1}{n-1} \sum_{i=1}^N \left( \frac{x_i - \bar{x}}{s_x} \right) \left( \frac{y_i - \bar{y}}{s_y} \right)$$

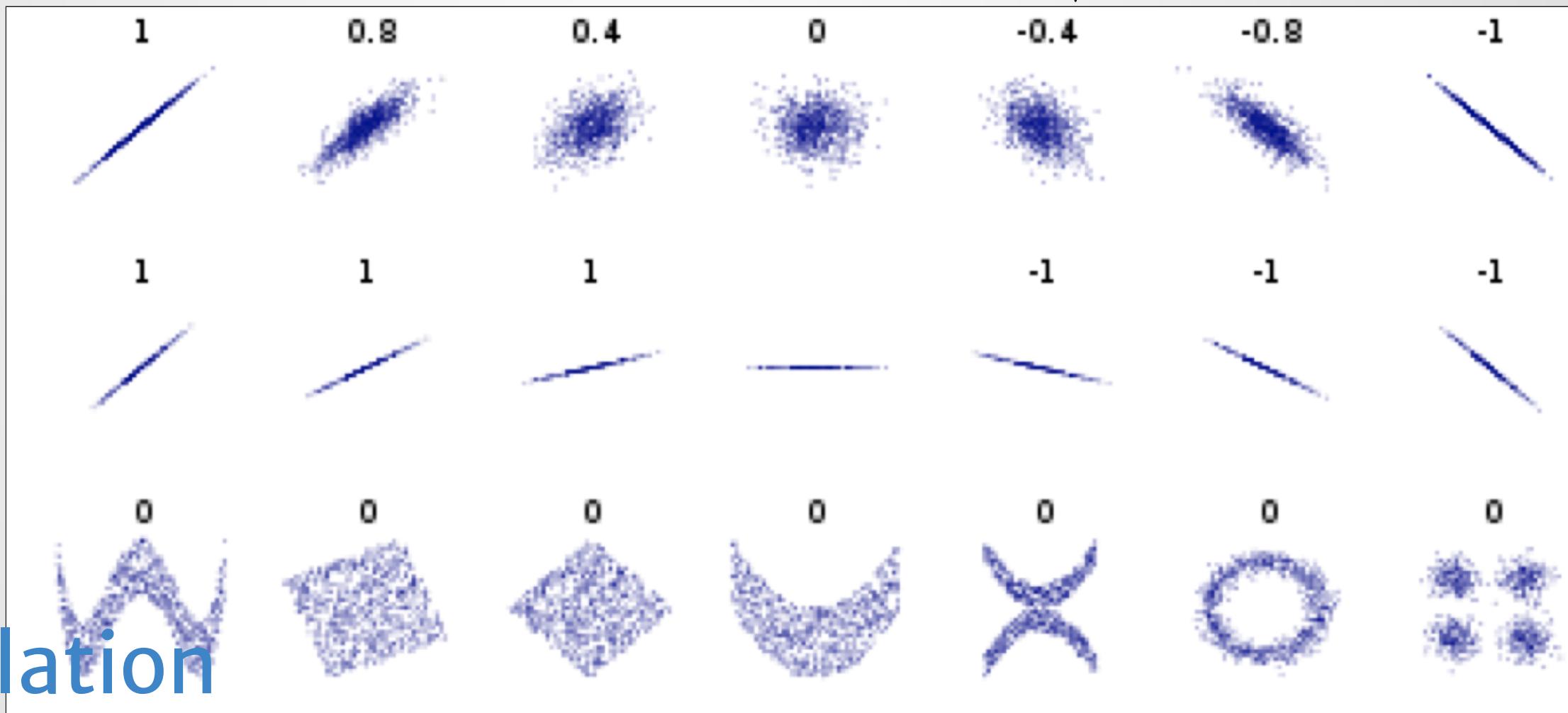
$\bar{x}$  : mean value of  $x$

$\bar{y}$  : mean value of  $y$

$n$  : number of datapoints

$$s_x = \sqrt{\frac{1}{n-1} \sum_{i=1}^N (x_i - \bar{x})^2}$$

Pearson's correlation measures *linear* correlation



Pearson's correlation

$$r_{xy} = \frac{1}{n-1} \sum_{i=1}^N \left( \frac{x_i - \bar{x}}{s_x} \right) \left( \frac{y_i - \bar{y}}{s_y} \right)$$

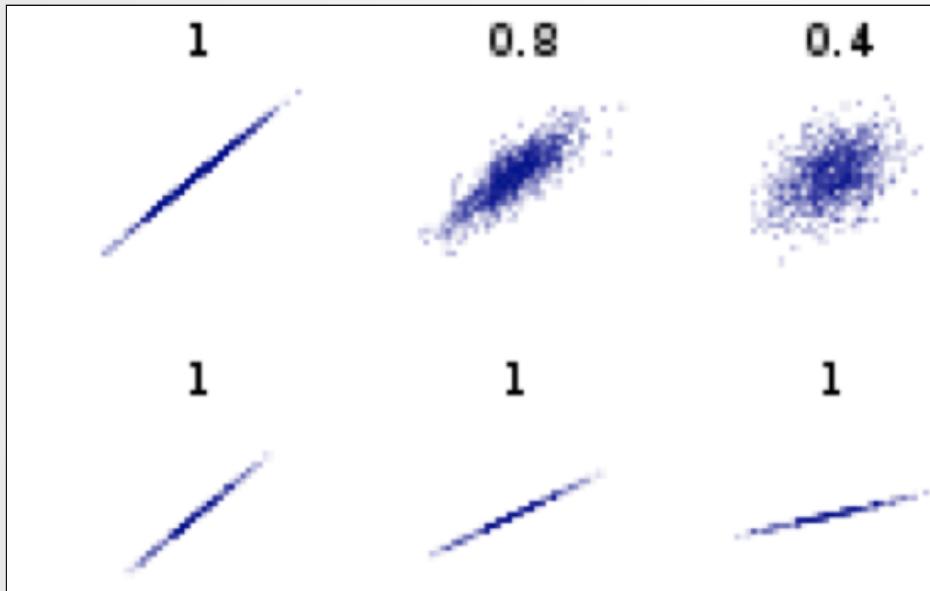
$\bar{x}$  : mean value of  $x$

$\bar{y}$  : mean value of  $y$

$n$  : number of datapoints

$$s_x = \sqrt{\frac{1}{n-1} \sum_{i=1}^N (x_i - \bar{x})^2}$$

Pearson's correlation measures *linear* correlation



correlated

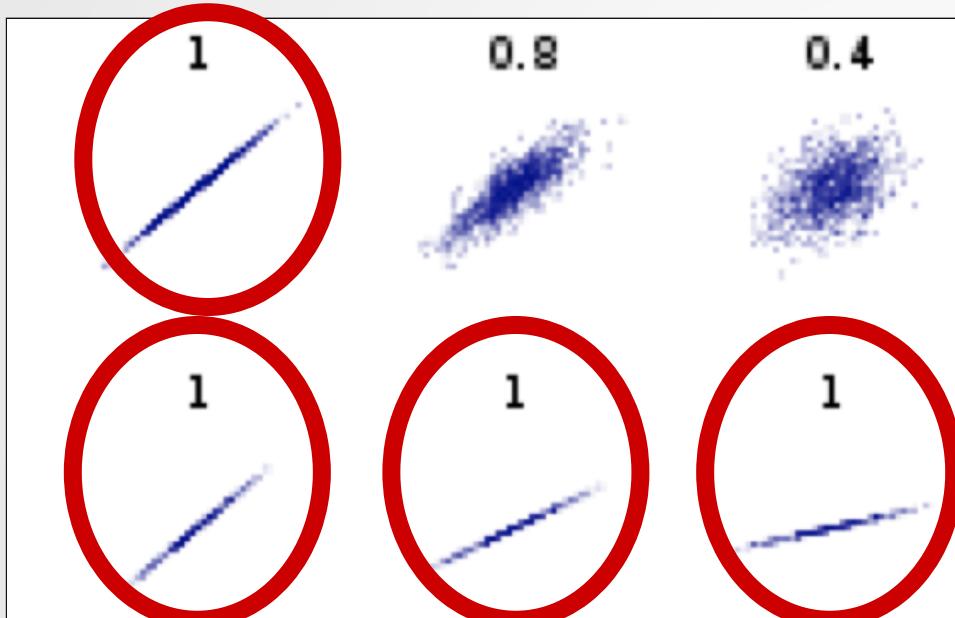
"positively" correlated

correlation

Pearson's correlation

$$r_{xy} = \frac{1}{n-1} \sum_{i=1}^N \left( \frac{x_i - \bar{x}}{s_x} \right) \left( \frac{y_i - \bar{y}}{s_y} \right)$$

Pearson's correlation measures *linear* correlation



correlation

$\bar{x}$  : mean value of  $x$

$\bar{y}$  : mean value of  $y$

$n$  : number of datapoints

$$s_x = \sqrt{\frac{1}{n-1} \sum_{i=1}^N (x_i - \bar{x})^2}$$

correlated

"positively" correlated

$$r_{xy} = 1 \text{ iff } y = ax$$

maximally correlated

Pearson's correlation

$$r_{xy} = \frac{1}{n-1} \sum_{i=1}^N \left( \frac{x_i - \bar{x}}{s_x} \right) \left( \frac{y_i - \bar{y}}{s_y} \right)$$

Pearson's correlation measures *linear* correlation

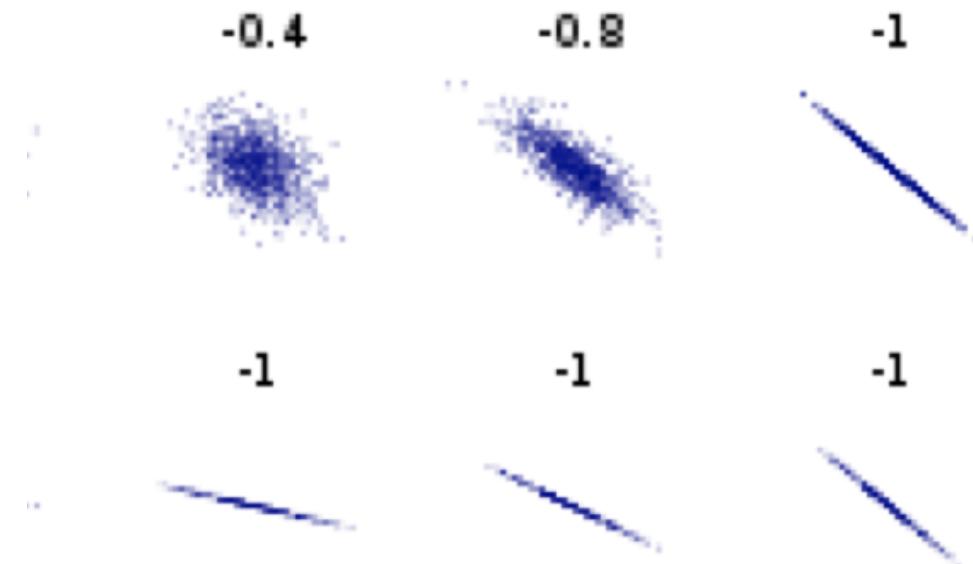
$\bar{x}$  : mean value of  $x$

$\bar{y}$  : mean value of  $y$

$n$  : number of datapoints

$$s_x = \sqrt{\frac{1}{n-1} \sum_{i=1}^N (x_i - \bar{x})^2}$$

anticorrelated  
"negatively" correlated



correl

Pearson's correlation

$$r_{xy} = \frac{1}{n-1} \sum_{i=1}^N \left( \frac{x_i - \bar{x}}{s_x} \right) \left( \frac{y_i - \bar{y}}{s_y} \right)$$

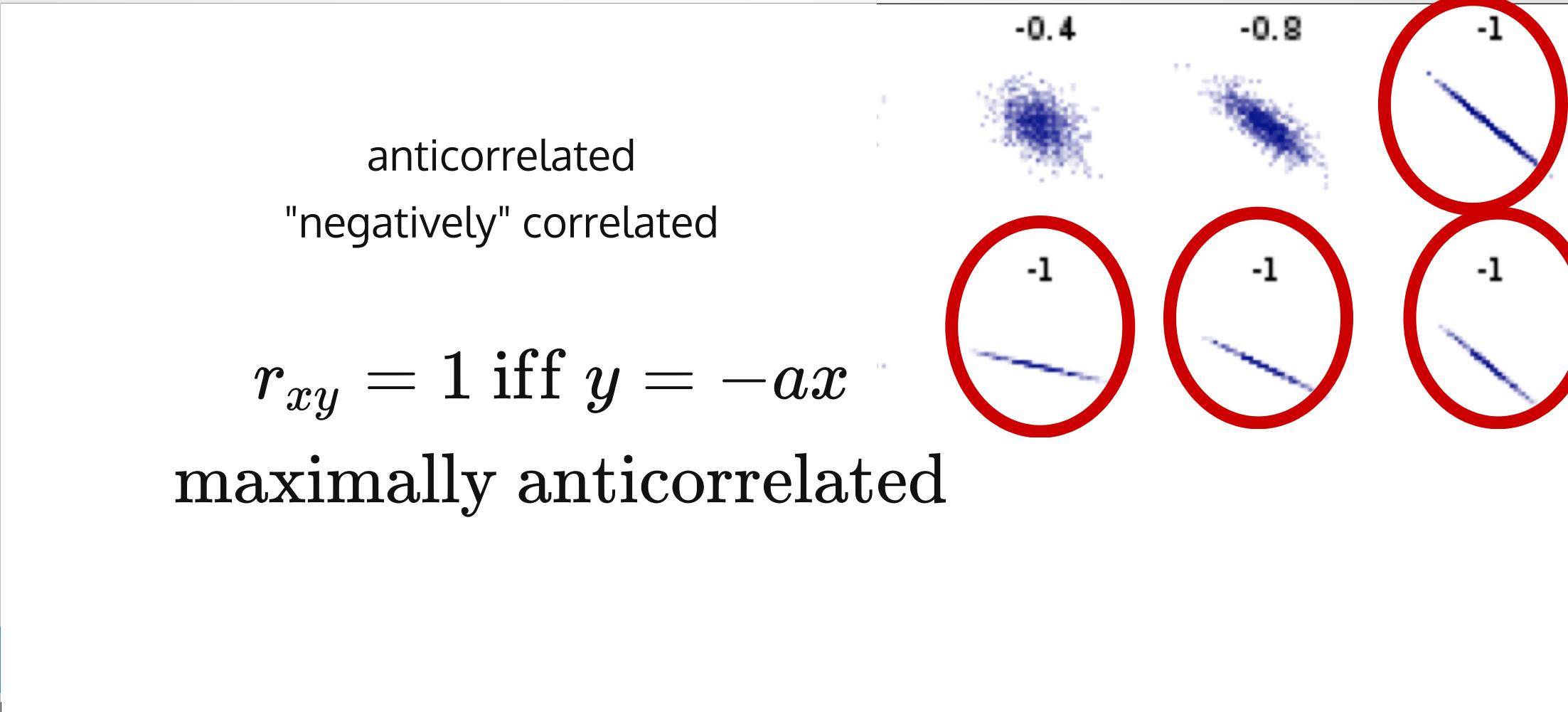
Pearson's correlation measures *linear* correlation

$\bar{x}$  : mean value of  $x$

$\bar{y}$  : mean value of  $y$

$n$  : number of datapoints

$$s_x = \sqrt{\frac{1}{n-1} \sum_{i=1}^N (x_i - \bar{x})^2}$$



Pearson's correlation

$$r_{xy} = \frac{1}{n-1} \sum_{i=1}^N \left( \frac{x_i - \bar{x}}{s_x} \right) \left( \frac{y_i - \bar{y}}{s_y} \right)$$

$\bar{x}$  : mean value of  $x$

$\bar{y}$  : mean value of  $y$

$n$  : number of datapoints

$$s_x = \sqrt{\frac{1}{n-1} \sum_{i=1}^N (x_i - \bar{x})^2}$$

Pearson's correlation measures *linear* correlation

not *linearly* correlated

Pearson's coefficient = 0

does not mean that  $x$  and  $y$  are independent!

correlation



Pearson's correlation

$$r_{xy} = \frac{1}{n-1} \sum_{i=1}^N \left( \frac{x_i - \bar{x}}{s_x} \right) \left( \frac{y_i - \bar{y}}{s_y} \right)$$

$\bar{x}$  : mean value of  $x$

$\bar{y}$  : mean value of  $y$

$n$  : number of datapoints

$$s_x = \sqrt{\frac{1}{n-1} \sum_{i=1}^N (x_i - \bar{x})^2}$$

Spearman's test

(Pearson's for ranked values)

$$\rho_{xy} = 1 - \frac{6 \sum_{i=1}^N (x_i - y_i)^2}{n(n^2 - 1)}$$

How many dichotomous <sup>+</sup> (binary) variables?				
	Y	Both variables <a href="#">interval or ratio</a> ?		
	Y	Measures are linear? (No = monotonic <sup>*</sup> )		
	N	<a href="#">Pearson correlation</a>		
0	N	<a href="#">Spearman correlation</a>		
Both variables are <a href="#">ordinal</a> ?				
	Y	<a href="#">Kendall correlation</a>		
	N	Both variables can be ranked?		
	Y	<a href="#">Kendall correlation</a>		
	N	Convert to frequency data and use <a href="#">Chi-square test</a> for independence		
1 <a href="#">Biserial Correlation Coefficient</a>				
2	2 x 2 table?		<a href="#">Save the figure</a>	
	Y	<a href="#">Phi</a>		
	N	<a href="#">Cramer's V</a>		

correlation

## Pearson's correlation

$$r_{xy} = \frac{1}{n-1} \sum_{i=1}^N \left( \frac{x_i - \bar{x}}{s_x} \right) \left( \frac{y_i - \bar{y}}{s_y} \right)$$

$\bar{x}$  : mean value of  $x$

$\bar{y}$  : mean value of  $y$

$n$  : number of datapoints

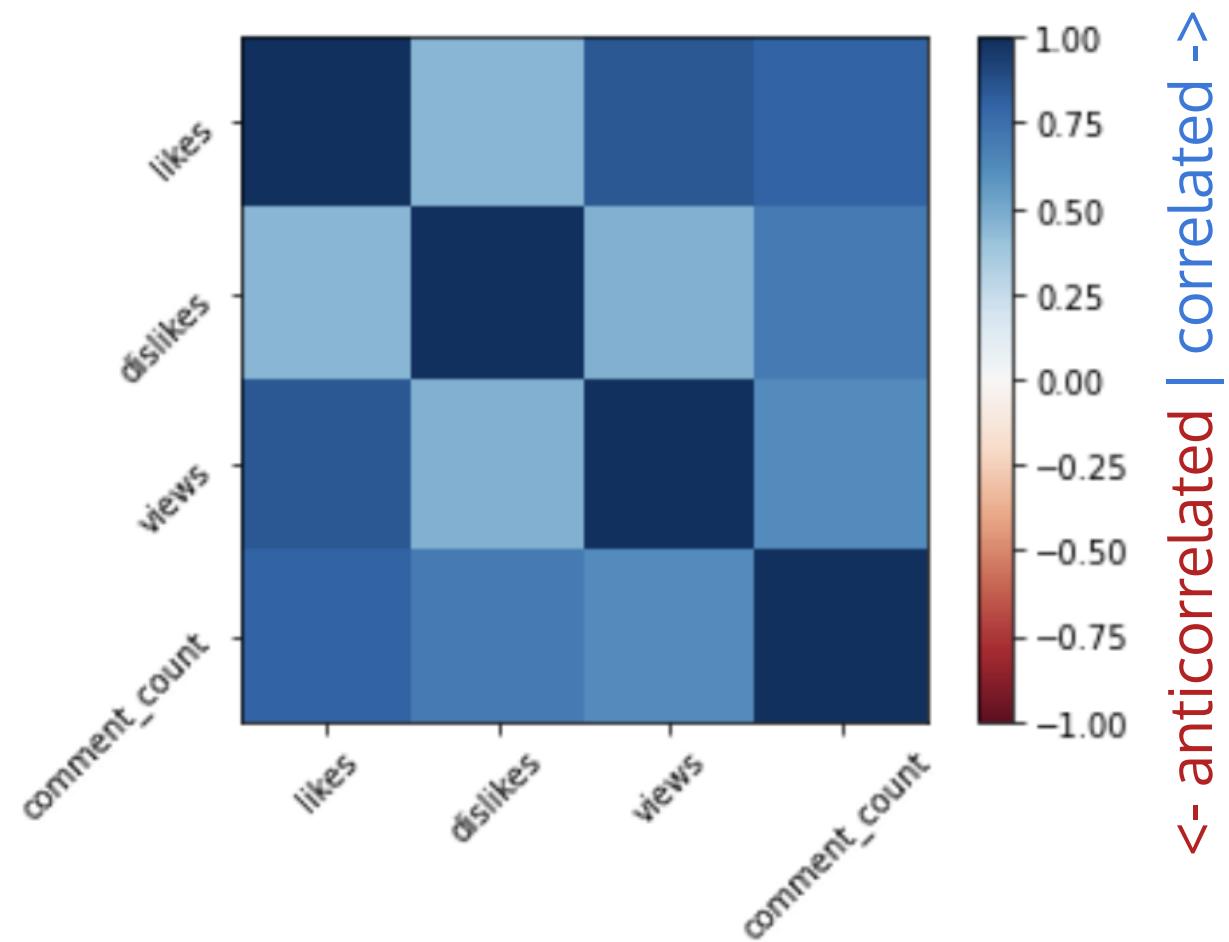
$$s_x = \sqrt{\frac{1}{n-1} \sum_{i=1}^N (x_i - \bar{x})^2}$$

```
1 import pandas as pd
2 df = pd.read_csv(file_name)
3 df.corr()
```

	likes	dislikes	views	comment_count
likes	1.000000	0.447186	0.849177	0.803057
dislikes	0.447186	1.000000	0.472213	0.700184
views	0.849177	0.472213	1.000000	0.617621
comment_count	0.803057	0.700184	0.617621	1.000000

```
1 pl.imshow(vdf.corr(), clim=(-1,1), cmap='RdBu')
2 pl.xticks(list(range(len(df.corr()))),
3           df.columns, rotation=45)
4 pl.yticks(list(range(len(df.corr()))),
5           df.columns, rotation=45)
6 pl.colorbar();
```

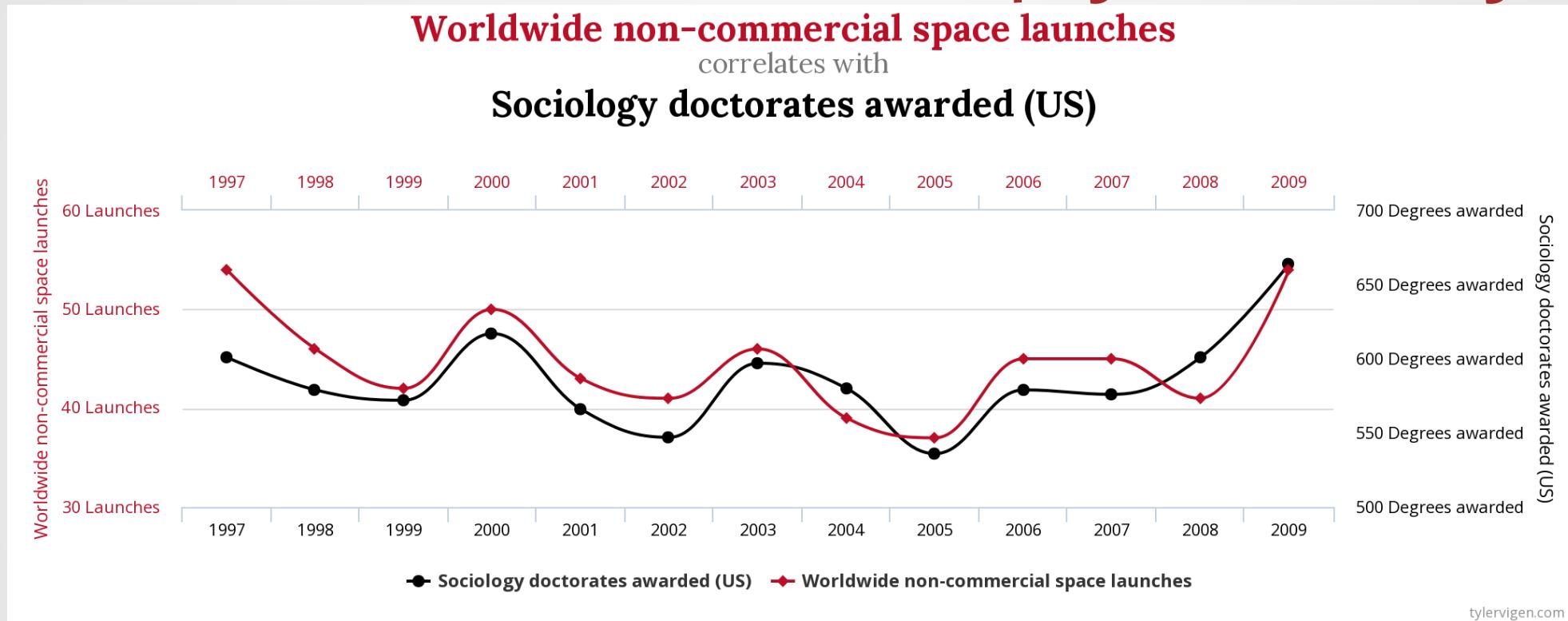
# correlation



<- anticorrelated | correlated ->

# correlation

# Correlation does not imply causality!!



2 things may be related because they share a cause but not cause each other:

*icecream sales with temperature / death by drowning with temperature*

In the era of big data you may encounter truly spurious correlations

*divorce rate in Maine / consumption of Margarine*

What is Data Science

key concepts

# 1

# homework

- make an account on GitHub if you do not have one yet
- Create a repository called FDSfE\_<firstinitialLastname>
- Add a README file to your repository that indicates that this is the repo for Foundations of Data Science

# 2

# homework

main · 1 branch · 0 tags

Go to file Add file · Code

 fedhere Created using Colaboratory · 16760c5 · 10 seconds ago · 13 commits

 CodeExamples · Created using Colaboratory · 1 hour ago

 HW1 · Created using Colaboratory · 10 seconds ago

 README.md · Update README.md · 7 hours ago

 Resources.md · Update Resources.md · 17 minutes ago

 README.md

**FDSfE**

Reno for Foundations of Data Science for Everyone - class taught at Lincoln University + University of Delaware

About

Repo for Foundations of Data Science for Everyone

 Readme ·  0 stars ·  2 watching ·  0 forks

Releases

No releases published · Create a new release

Packages

- upload to your github repo the colab notebook we have worked on in class. the colab notebooks should appear in a folder HW1 and have a google colab link (more instructions will appear on github)

main · FDSfE\_FBianco / HW1 / first\_python\_nb.ipynb

fedhere · Created using Colaboratory · Latest commit 16760c5 now · History

1 contributor

703 lines (703 sloc) | 15.2 KB

 Open in Colab

Intro: this is a text cell

In [1]: `# this is a code cell  
x = 1`

simple math:  $2 + 2 \dots$

In [2]: `2 + 2`

Out[2]: 4

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

[https://www.aaas.org/sites/default/files/Stats\\_What\\_Question\\_2015.pdf?  
g\\_zGQR5m3rDJqwXqj3DxLI5pXZ3hNdHk](https://www.aaas.org/sites/default/files/Stats_What_Question_2015.pdf?g_zGQR5m3rDJqwXqj3DxLI5pXZ3hNdHk)

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*

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

**Reproducibility and Replicability in  
Science**

[Consensus Study Report \(2019\)](#)

<https://www.nap.edu/catalog/25303/reproducibility-and-replicability-in-science>

additional reading

# Text

STATISTICS

## *What is the question?*

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

By Jeffery T. Leek and Roger D. Peng

