



Introduction to Pandas

An Overview of Python's Data Analysis Library

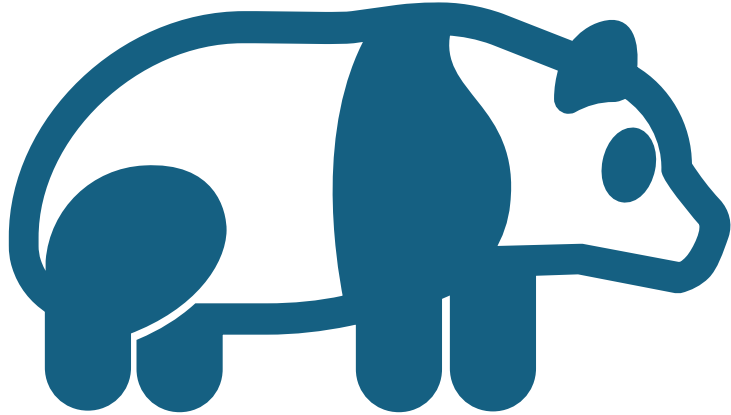
Samuel Gelman,

Weizmann Institute: LSCF, Bioinformatics unit



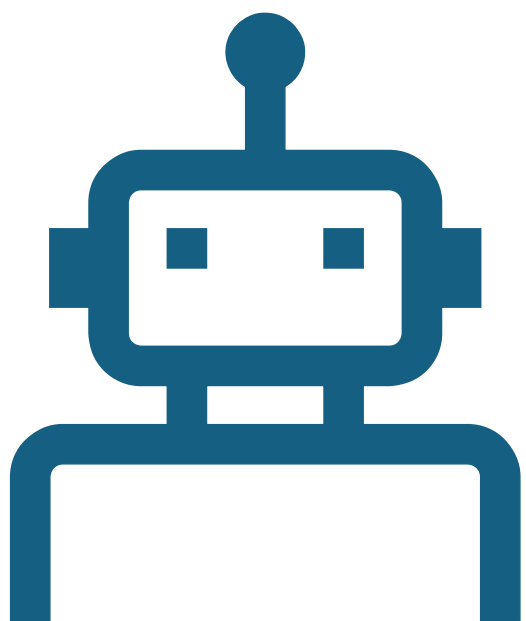
What will I learn?

- The what?
- The how?
- The why?



What

- What is Pandas and where does it fall in relation to Python as a language.
- Capabilities and functionalities.
- Useful tools which help make the most out of Pandas.



How

- How to work with Pandas through a hands-on approach.
- Familiarizing and experimenting with its capabilities.
- How to dream



”

Why

- The why now
- The why after



Workshop outline

- Outline and overview of Pandas and the functions we will be working with.
- Loading the repository containing the scripts we need.
- Building the proper environment to work in.
- Getting our hands dirty and writing some code.



Other tools

- A few other tools to help us along the way



Github

- Github is the graphical user-interface (GUI) to git. A language developed by Linus Torvalds to help him build the Linux Kernel!
- Hosts billions of lines of code across millions of repositories.
- This workshop lives on git.



Anaconda

- Anaconda streamlines the setup of Python environments.
 - Anaconda is the backbone for setting up the Python environment needed to run our pandas workshop in Jupyter notebooks.
- `conda env create -f environment.yml`

PyPI and pip



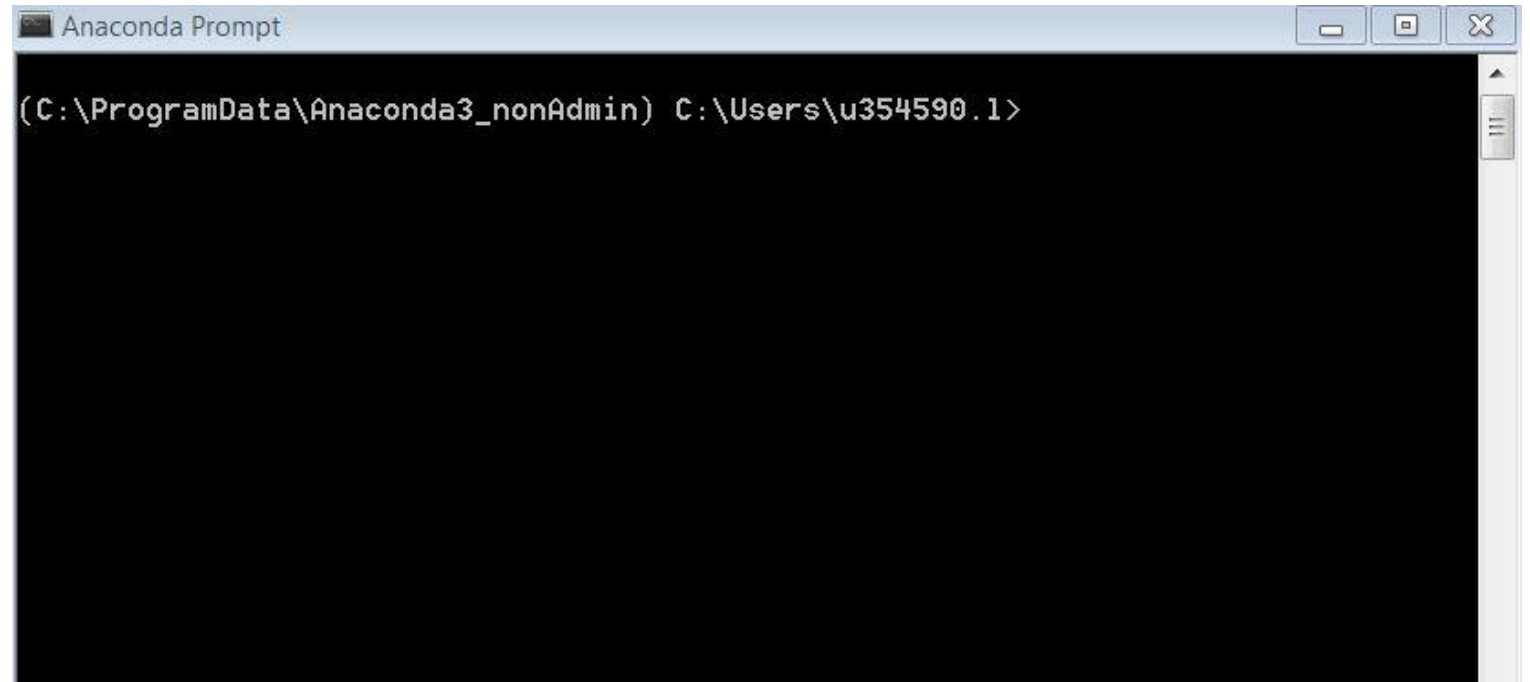
- Just as GitHub hosts code repositories, pip interfaces with the Python Package Index (PyPI).
- PyPI is mainly community driven but is overseen by the Python Software Foundation which sets the guidelines and infrastructure for the Index and its security.
- When you use pip to install a package, it connects to PyPI, locates the package, and downloads it along with any required dependencies.



Jupyter

- Jupyter provides an interactive computing environment where you can write, run, and visualize code.
- Widely adopted in academia and industry.
- This is where we will be doing all of our hands-on work.

Let's do a
little prep
together!

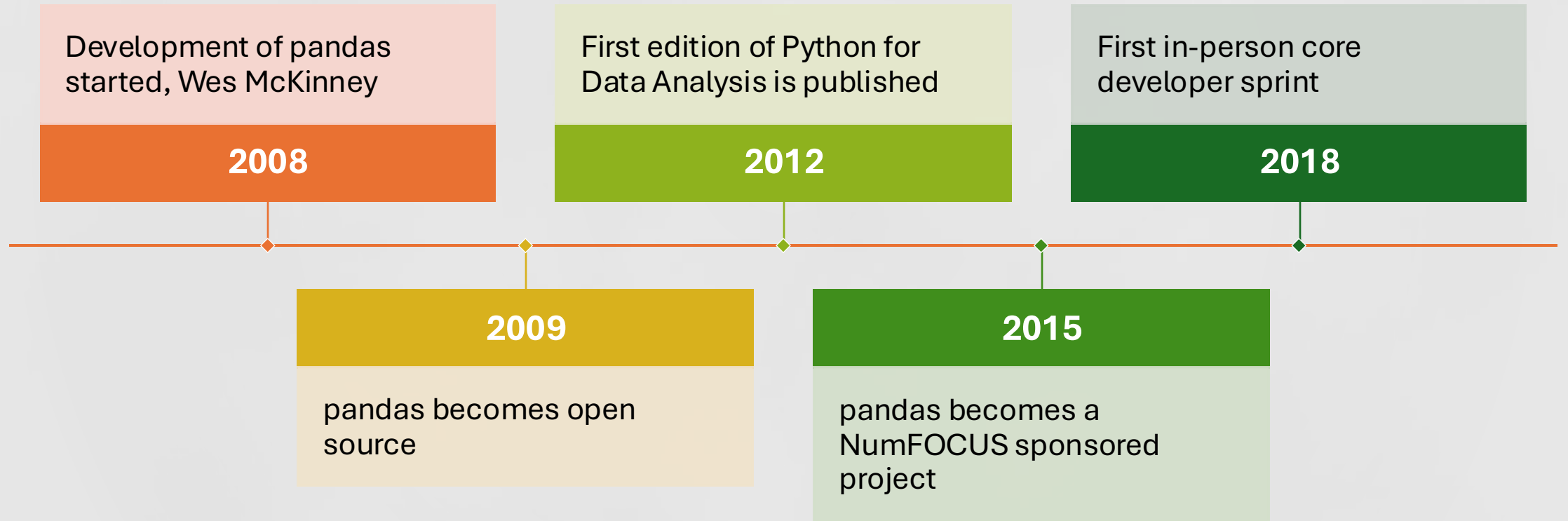


```
conda env create -f environment.yml
```

What is Pandas?



Short History



Pandas namesake?

Panel Data + Python Data Analysis = Pandas

Series

One-dimensional labeled array capable of holding any data type.

```
class pandas.Series(data=None, index=None, dtype=None,  
name=None, copy=None, fastpath=_NoDefault.no_default) #
```


DataFrame

A DataFrame in Pandas is a two-dimensional labeled data structure capable of holding data of various types in a tabular format

DataFrames can be thought of as a collection of Pandas Series where each Series represents a column.

With many built-in methods!!

Functions vs methods



Both run blocks of code but differ in scope.



Function are standalone blocks that are not bound to a particular object.

```
result = my_function(argument)
```



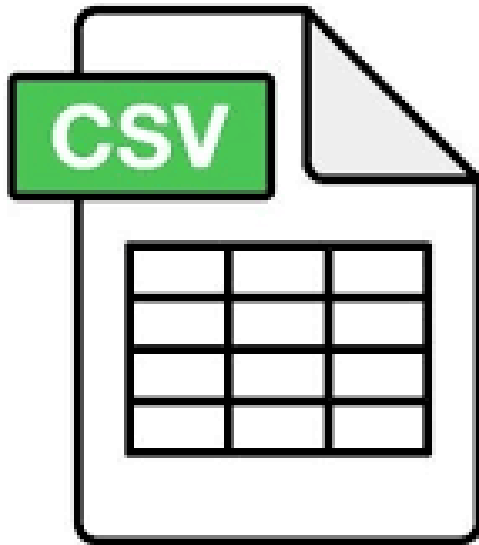
Methods are functions defined within a class and are bound to instances of that class.

```
my_object = MyClass()  
Result = my_object.my_method()
```

Dictionaries vs DataFrames



Loading in data.



Subsets of the data

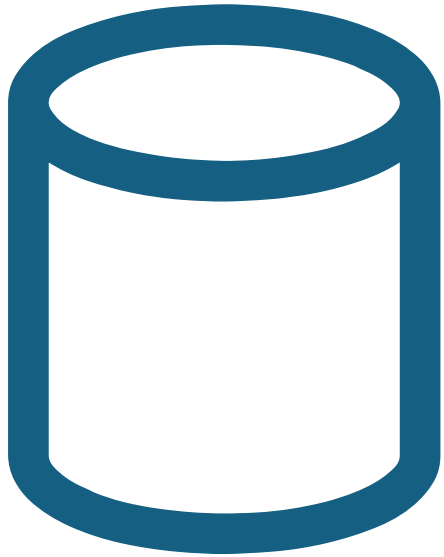
HEAD

LOAD

COLUMN
NAME

INDEXING

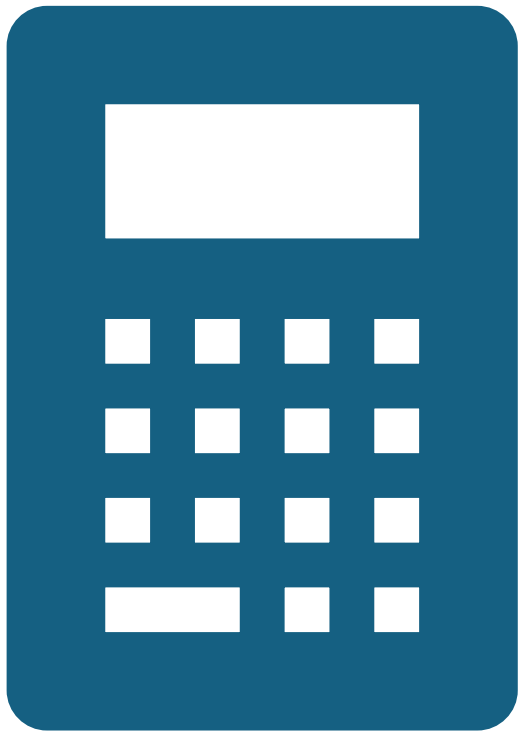
SLICING



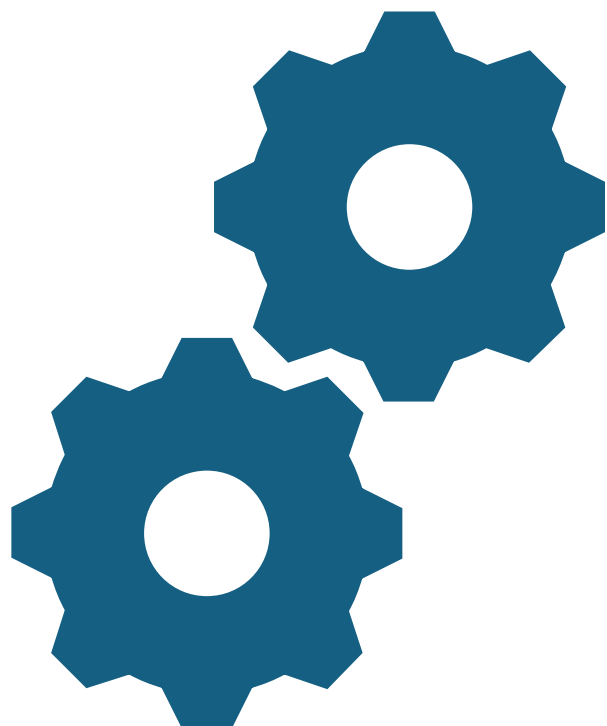
Dtype exercise

- Dtypes
- Subsets
- Managing Data

Mathematical Operations



Feature Engineering





Masks

Time



Groupby





Plotting



Let's Go!

