Hello everyone !!

NJIT
New Jersey Institute
of Technology

Interactive Large-Scale Data and Graph Analytics through Arkouda and Arachne

Data Analytics

Naren Khatwani
Oliver Alvarado Rodrigues
Dr Zhihui Du
Dr David Bader

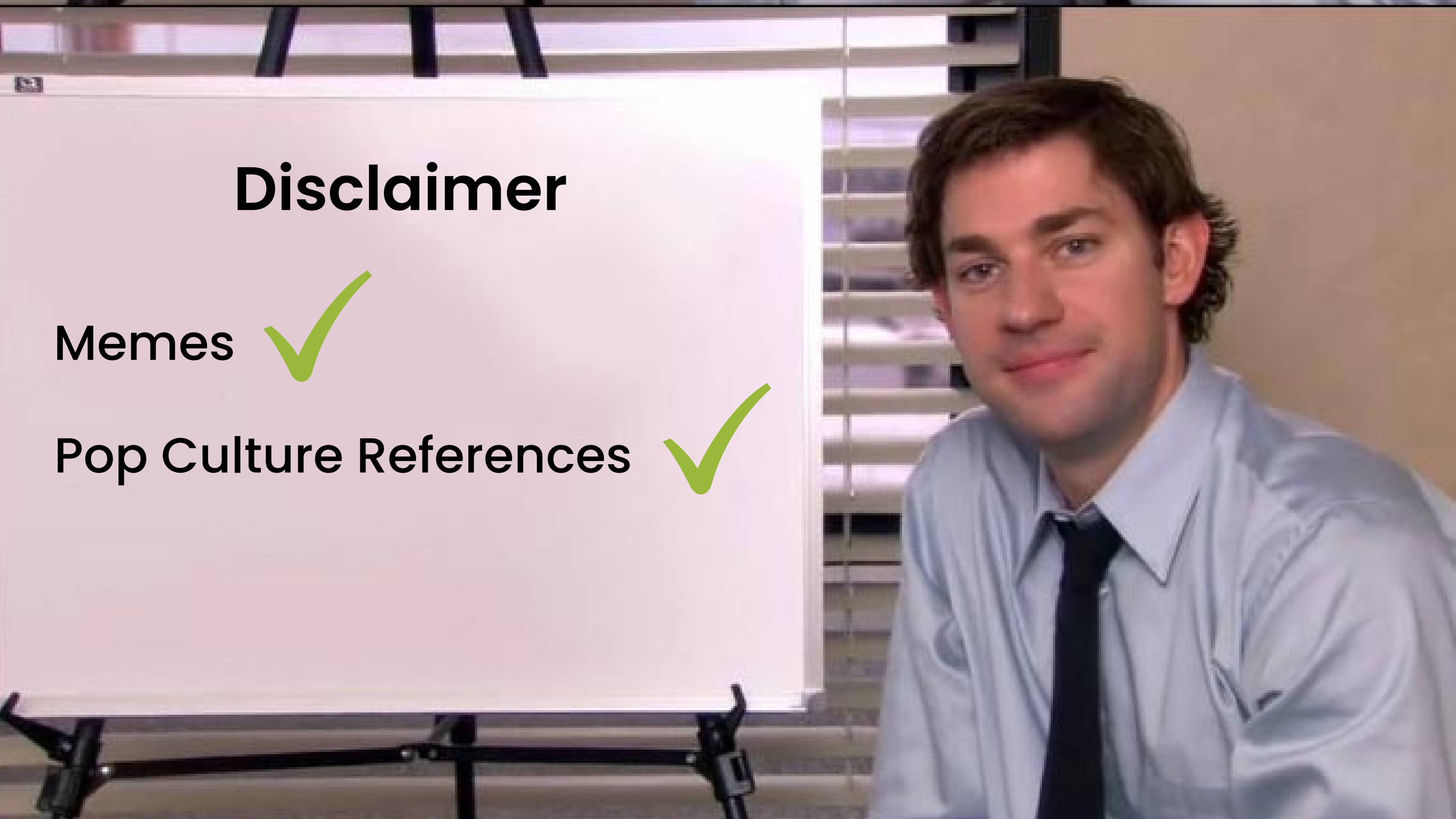
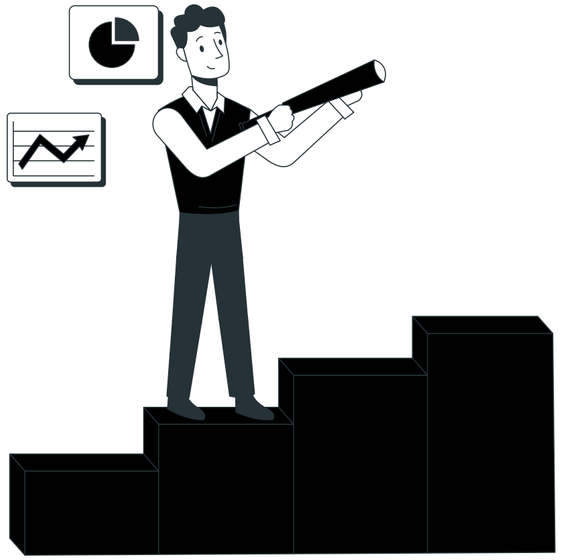

Disclaimer
Memes
Pop Culture References

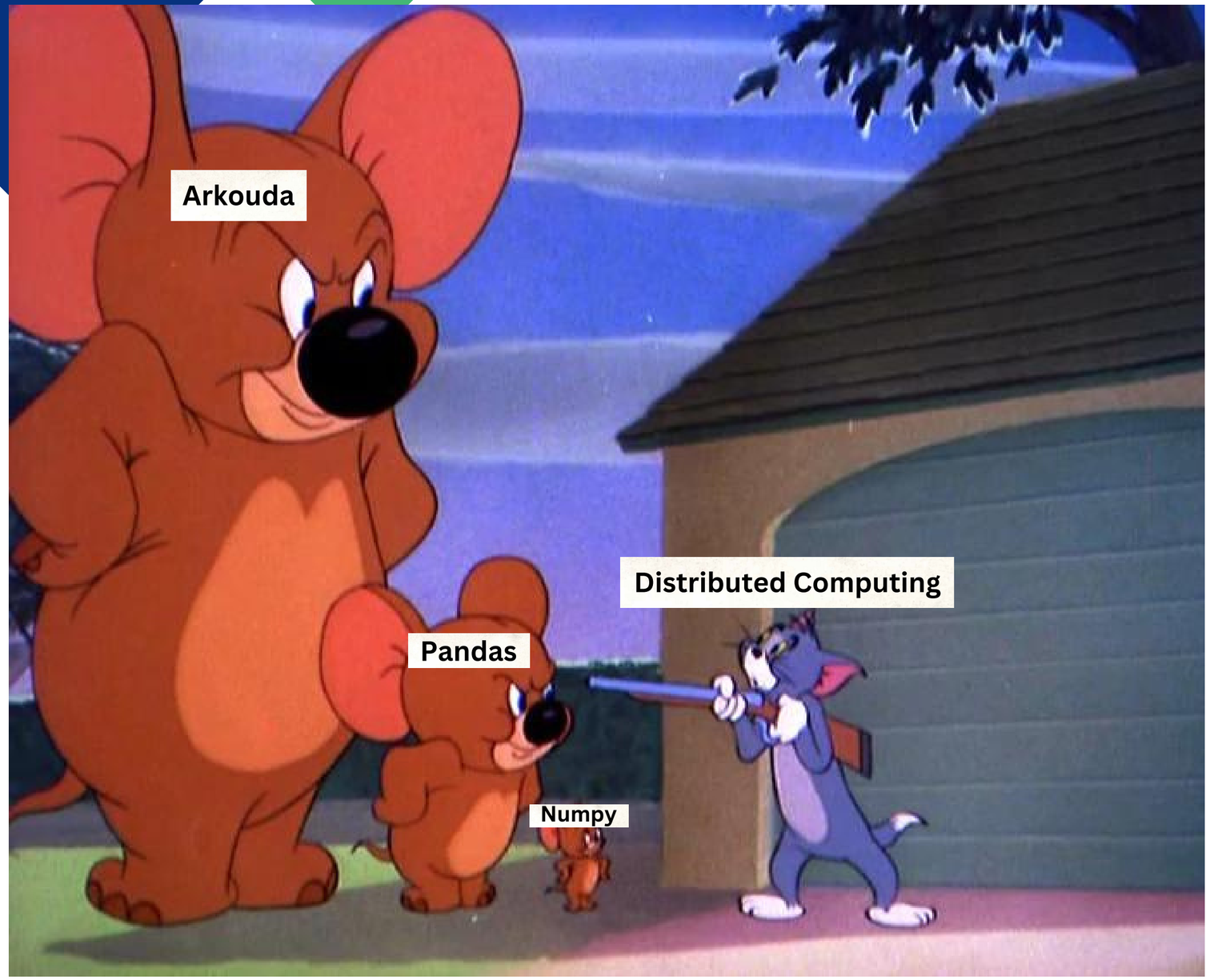
Table of contents

A gist of the topics that will be covered today

Data Analytics using Arkouda

- Systematic computational analysis
- Used for the discovery, interpretation, and communication of meaningful patterns in data
- Eventually, helping toward effective decision-making.
- Expanding the limits of existing libraries like numpy and pandas





- We are just trying to extend the threshold of numpy and pandas, not replace it.

- Pandas can handle data frames up to about 500 million rows before performance becomes a real issue; provided that you run on a sufficiently capable compute server.

- Arkouda breaks the shared memory paradigm and scales its operations to data frames with over 200 billion rows, maybe even a trillion

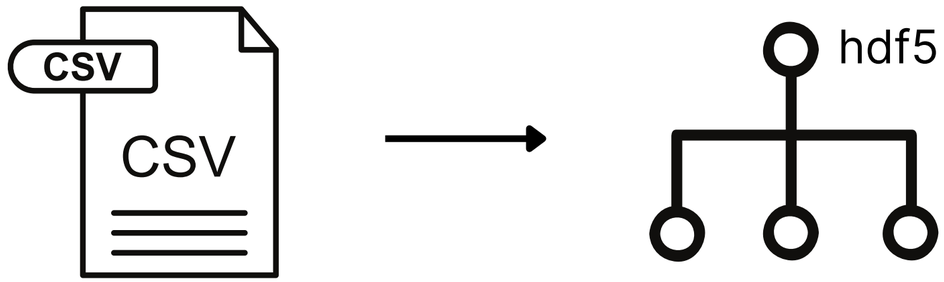
nd.array	pd.array	ak.pdarray
• Operations run sequentially	• Operations run sequentially	• Operations run with parallelism
• Operations cannot be distributed amongst locales	• Operations cannot be distributed amongst locales	• Operations can be distributed amongst locales
• Example: import numpy as np arr = np.array((1, 2, 3, 4, 5)) print(arr)	• Example: import pandas as pd arr = pd.array(data=[1, 2, 3, 4, 5] dtype=str) print(arr)	• Example: import arkouda as ak arr = ak.arange(0,5,1) print(arr)

Use of CSV in Pandas and HDF5 in Arkouda

- For Pandas, data sets need to be either in the form of CSV, XLSX, ZIP, Plain Text (txt), JSON, XML, or HTML.

- However, when using Arkouda to analyze large datasets, they need to be in the form of HDF5 (Hierarchical Data Format Files).

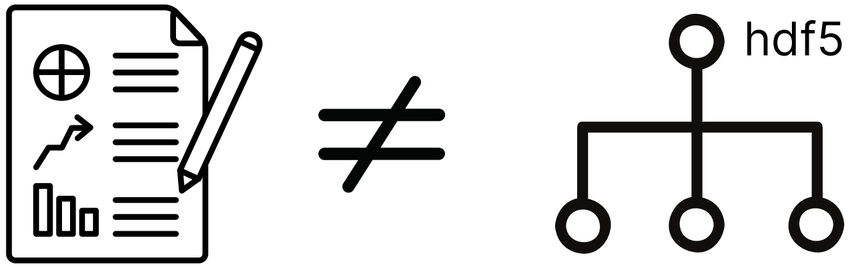
- We will provide examples of how to convert CSV files into HDF5 to be used with Arkouda.



Differences between HDF5 and CSV for Pandas

- An HDF5 file consists only of data types integer, floats, and strings. Additionally, it also can handle some advanced data types like date and time.

- On the contrary, a CSV file can contain data of varied types date time, strings, integer, boolean values and so on.



Differences in terms of structure

.csv

A CSV file is a delimited text file that uses a comma to separate values. Each line of the file is a data record. Each record consists of one or more fields, separated by commas.

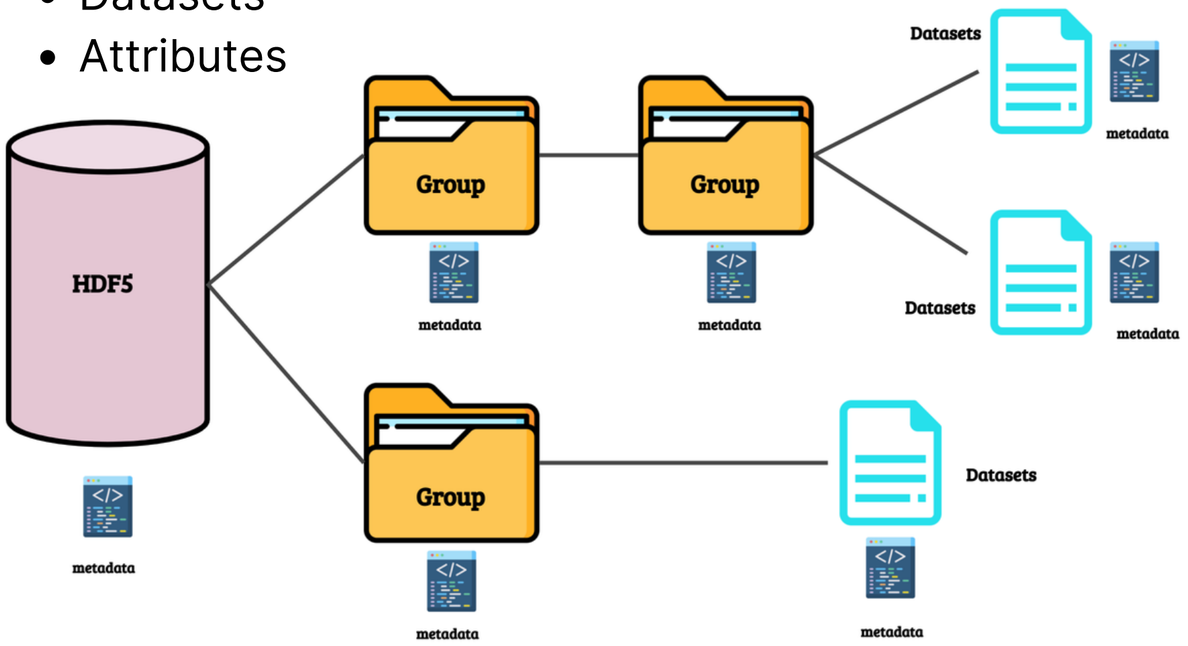
data.csv

name, school, age
Naren, NJIT, 23
Oliver, NJIT, 25
Dave, NJIT, 22
Vanshika, NJIT, 19

.hdf5

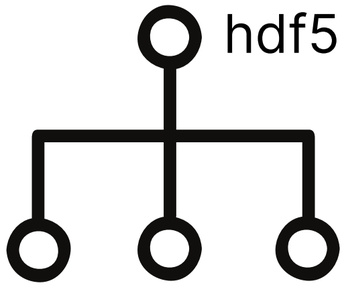
HDF5 uses a "file directory" like structure that allows you to organize data within the file in many different structured ways, as you might do with files on your computer.

- Groups
- Datasets
- Attributes



Advantages of using .hdf5 for Arkouda

- Supported by arkouda and pandas out of the box.
- Therefore, supported by the jupyter-notebook out of the box.
- Multiple data sets
- Includes a good amount of metadata.
- Data is self-describing.



Supported Datatypes in hdf5

- int (any width) - int64
- float (any width) - float64
- custom string formats - ak.strings

Additional Datatypes in Arkouda

- Boolean
- DateTime (from int64)
- Timedelta (from int64) - 2009/2/10,14::00

How to convert an CSV file to a .hdf5 file?

A	B	C
1	Naren	NJIT
2	Oliver	MIT
3	Dave	NJIT

```
{
'A':array([1,2,3]),
'B': array([Naren, Oliver,Dave]),
'C': array([NJIT,MIT,NJIT])
}
```

Formatter File for converting a CSV to hdf5

- A converter that comes bundled with Arkouda
- A python module that takes the exact same options as the pandas read_csv method

Considerations in a formatter file:

- Separator
- Header
- Date Time parser
- Converters to modify the existing column's data types

NYCTaxi_format.py

- Considering the ideal NYCTaxi Dataset

```
%%file NYCTaxi_format.py

import numpy as np

OPTIONS = {}

def YNint(yn):
    return (0, 1)[yn.upper() in 'YES']

def nullint(x):
    try:
        return np.int64(x)
    except:
        return np.int64(-1)
```

Convert Boolean Values

- True - 1
- False - 0

Handling erroneous values

- They are replaced by -1

NYCTaxi_format.py

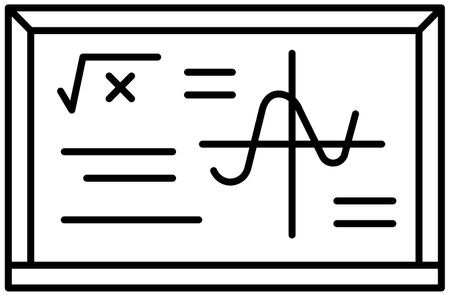
```
yellow_format = {'sep': ',',
                 'header': 0,
                 'parse_dates':['tpep_dropoff_datetime', 'tpep_pickup_datetime'],
                 'infer_datetime_format': True,
                 'converters': {'store_and_fwd_flag': YNint,
                                'VendorID': nullint,
                                'RatecodeID': nullint,
                                'PULocationID': nullint,
                                'DOLocationID': nullint,
                                'passenger_count': nullint,
                                'payment_type': nullint,
                                'trip_type': nullint}}

OPTIONS['yellow'] = yellow_format

green_format = yellow_format.copy()
green_format['parse_dates'] = ['lpep_dropoff_datetime', 'lpep_pickup_datetime']
OPTIONS['green'] = green_format
```

Mention separator

Row number(s) to use as the column names

Pandas will attempt to infer the format of the DateTime strings in the columns

Dictionary of functions for converting values in specific columns. Keys can either be integers or column labels.



Arithmetic Functions in Arkouda

- Arithmetic functions are present in both pandas and Arkouda, but in Arkouda, we deal with columns as arkouda PD arrays, and it becomes easy if we want to perform some operations similar to set operations on column-store type data.

```
>> A = ak.arange(10)
>> A
array([0 1 2 3 4 5 6 7 8 9])

>> A += 2
>> A
>> array([2, 3, 4, 5, 6, 7, 8, 9, 10, 11])
```

Example of an Arithmetic Function

- Here, we are dealing with arkouda PD arrays and we can just use the intersection function similar to finding an intersection between two sets. We will show some of the different Arkouda functions and how they can be utilized for data set exploration.

- Example of an arkouda pdarray :

```
>> a = ak.array([4, 2, 5, 6, 4, 7, 2])
>> b = ak.array([1, 5, 4, 11, 9, 6])

>> ak_int = ak.intersect1d(a, b)
>> ak_intarray([4 5 6])
```

Not Restricting the use of Pandas or NumPy

- Another incentive of Arkouda is that we are not restricting the use of pandas or NumPy, we are providing the group-by function which will sort the data faster using Arkouda and then export it as a data frame and the user can do an analysis using NumPy or pandas.

```
>> df
   F_Name    L_Name  Age    Salary
0  John      Doe     37     75000
1  Jane      Doe     35     77000
2  John      Smith   50     100000
3  Jake      Brown   32     35000      (4 rows x 4 columns)
```

because.....

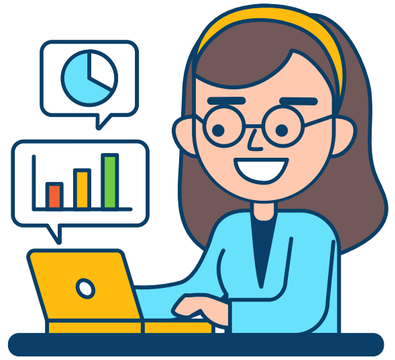

```
>> pd_df = df.to_pandas()

>> pd_df
```

	F_Name	L_Name	Age	Salary
0	John	Doe	37	75000
1	Jane	Doe	35	77000
2	John	Smith	50	100000
3	Jake	Brown	32	35000

- Now you can use pd_df for further data analytics with any pandas methods
- NOTE: Converting a HUGE dataset from arkouda pdarray to pd.array will take a really long time, so it should only be used on a small subset of data

Arkouda using Docker

- Ease of access
- Pre-packed dependencies
- Steps to use Arkouda via docker
  - Build a developer environment in conda
  - Run the docker image
  - Open up a jupyter notebook and connect the server



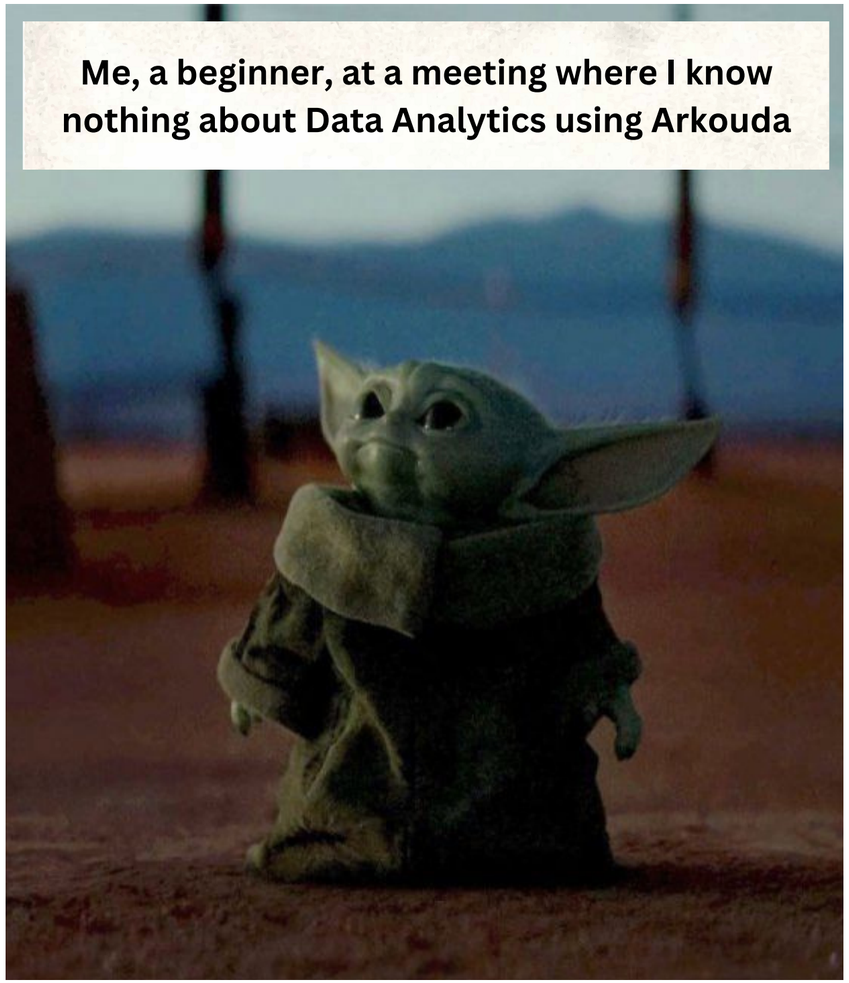
Example Dataset for analysis using Arkouda

- Dataset Name - Global Food Prices

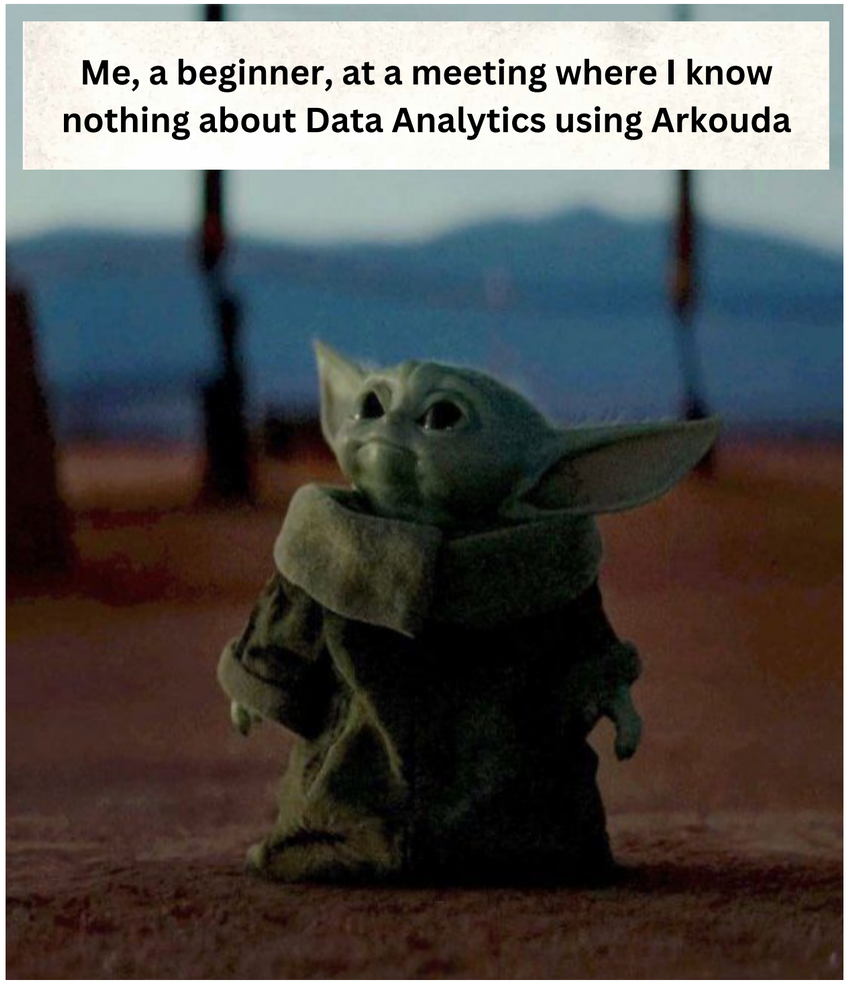
- Global food price fluctuations can cause famine and large population shifts. Price changes are increasingly critical to policymakers as global warming threatens to destabilize the food supply.

- Over 740k rows of prices obtained in developing world markets for various goods. Data includes information on country, market, price of good in local currency, quantity of good, and month recorded.

Day 1


Me, a beginner, at a meeting where I know nothing about Data Analytics using Arkouda

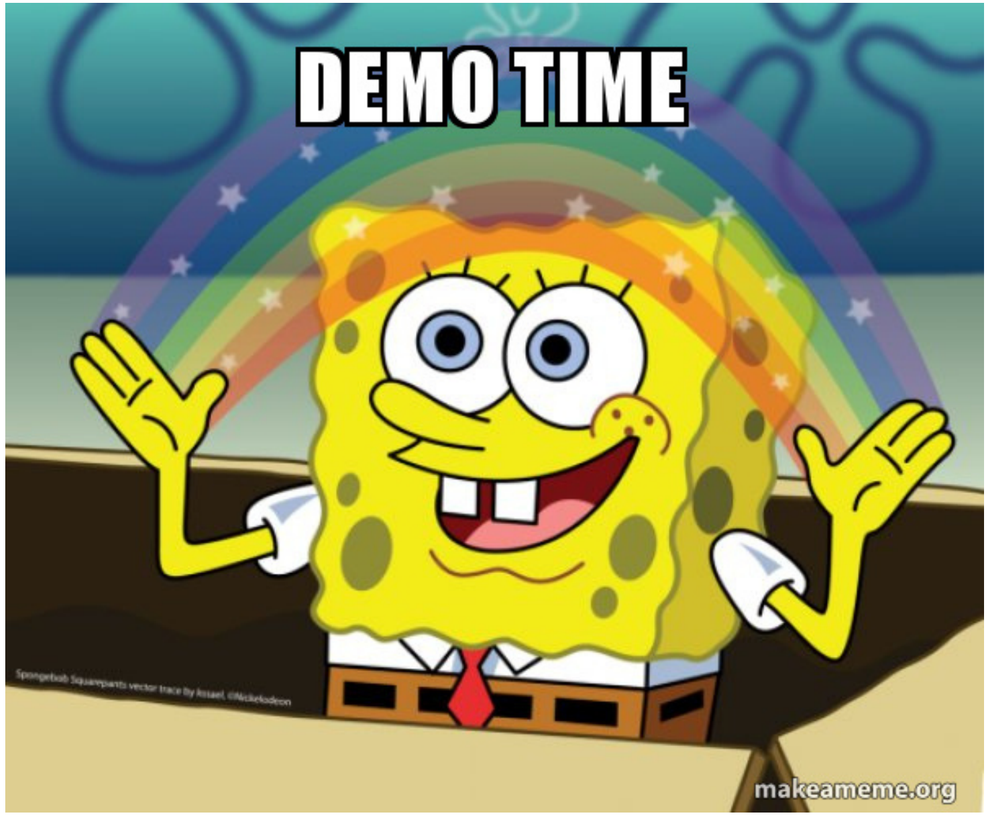
Day 1


Me, a beginner, at a meeting where I know nothing about Data Analytics using Arkouda

Day 20


Me, after analysing a Kaggle dataset

Moving to the Jupyter Notebook for further analysis



Conclusion

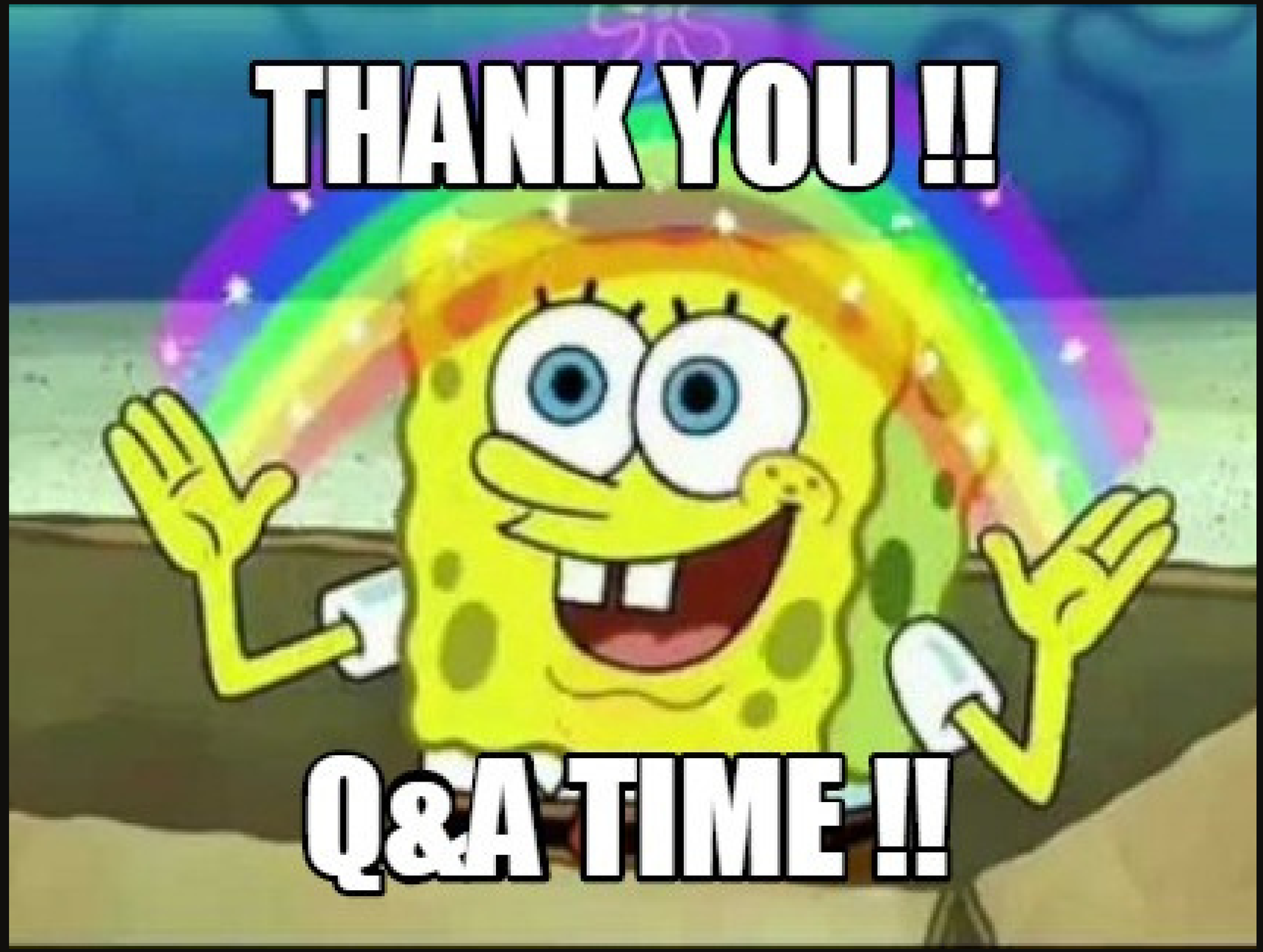
Thank you to all of my mentors

- Dr David Bader
- Dr Zhihui Du

- Oliver Alvarado Rodrigues
- Davor Petrovikj
- Vanshika Agrawal

&

stackoverflow


THANK YOU !!
Q&A TIME !!


NO HARD QUESTIONS, PLEASE!!!