DATAFRAME EVOLUTION: VAEX

Jovan Veljanoski

https://vaex.io/

https://github.com/vaexio/

VAEX.IO: WHO ARE WE

Maarten Breddels

Freelancer / consultant / data scientist

Core Jupyter-Widgets developer

Former astrophysicist

QuantStack partner

Founder of vaex.io

Principal author of Vaex

maartenbreddels@gmail.com

G www.maartenbreddels.com

@maartenbreddels

github.com/maartenbreddels

Jovan Veljanoski

Data scientist @ cloudsolutions.co.uk

Previously @ XebiaLabs

Former astrophysicist

Co-Founder of vaex.io

jovan.veljanoski@gmail.com

in https://www.linkedin.com/in/jovanvel/

@jovanvaex

github.com/jovanveljanoski

Yonatan Alexander

Head of AI @ CYBEAR.co

Previously head of data science at BuiltOn

jonathan@xdss.io

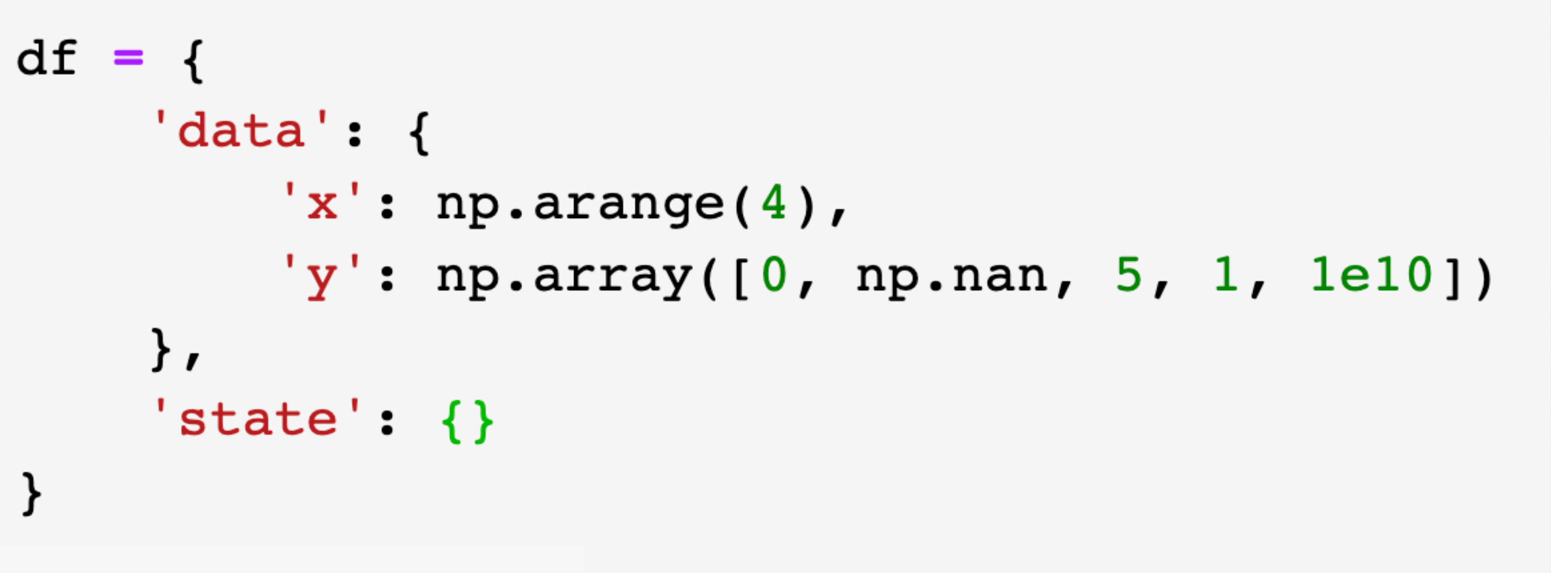
in https://www.linkedin.com/in/xdssio/

@xdssio

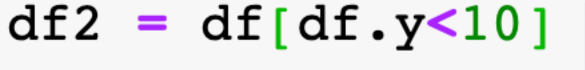
https://github.com/xdssio/

WHAT IS VAEX

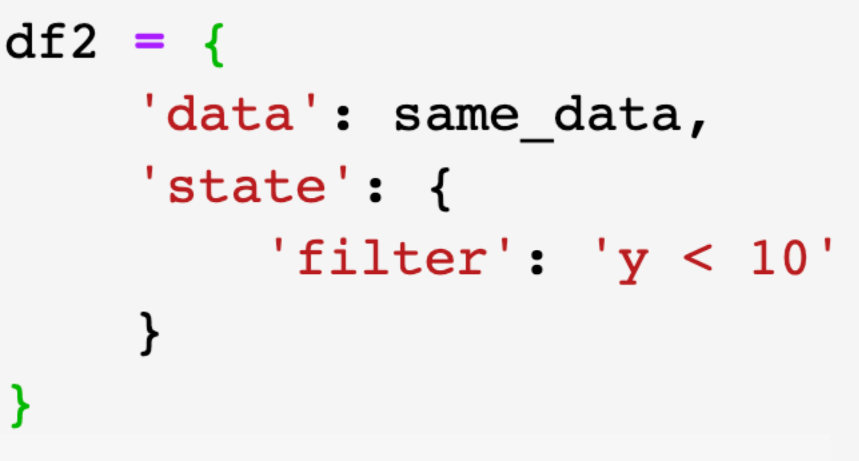
- High-performance, out-of-core DataFrame library
- Goal is to work with billions (10^9) of samples on a single machine / laptop interactively
- Like Pandas (similar API) but not built on Pandas
- Key concepts:
  - Memory mapping - work with datasets the size of your hard drive (Arrow, HDF5)
  - Expression system - memory efficiency, computational graph
  - Lazy evaluations - control flow, performance increase
  - High performance - efficient C++ algorithms, Just-In-Time compilation via Numba / Pythran / Cuda
- Legal: Free & Open Source, MIT Licence



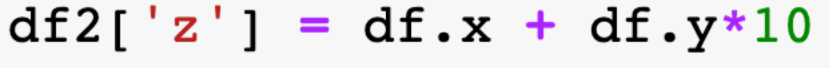
```
df = {
    'data': {
        'x': np.arange(4),
        'y': np.array([0, np.nan, 5, 1, 1e10])
    },
    'state': {}
}
```



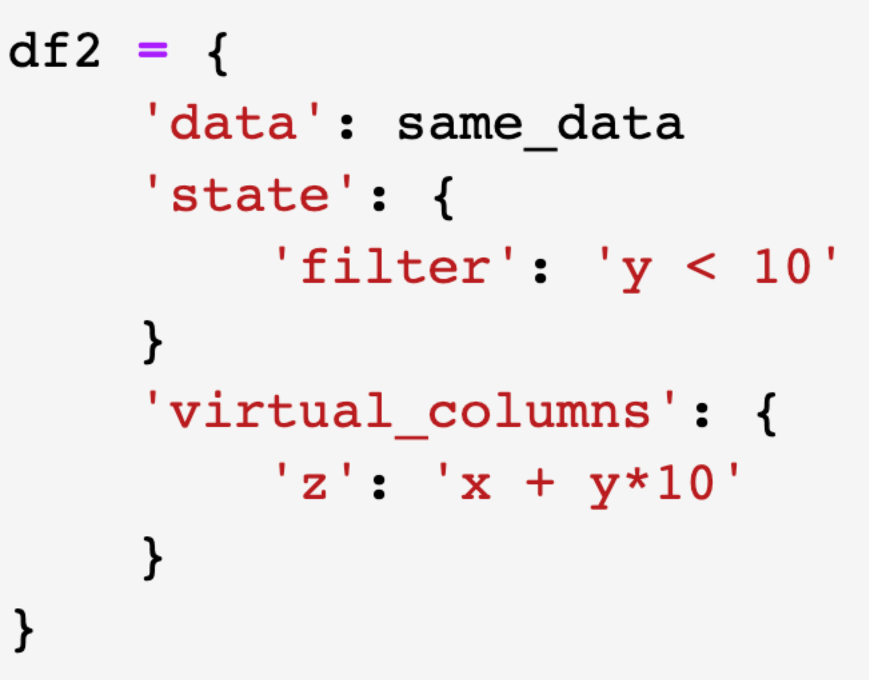
```
df2 = df[df.y<10]
```



```
df2 = {
    'data': same_data,
    'state': {
        'filter': 'y < 10'
    }
}
```



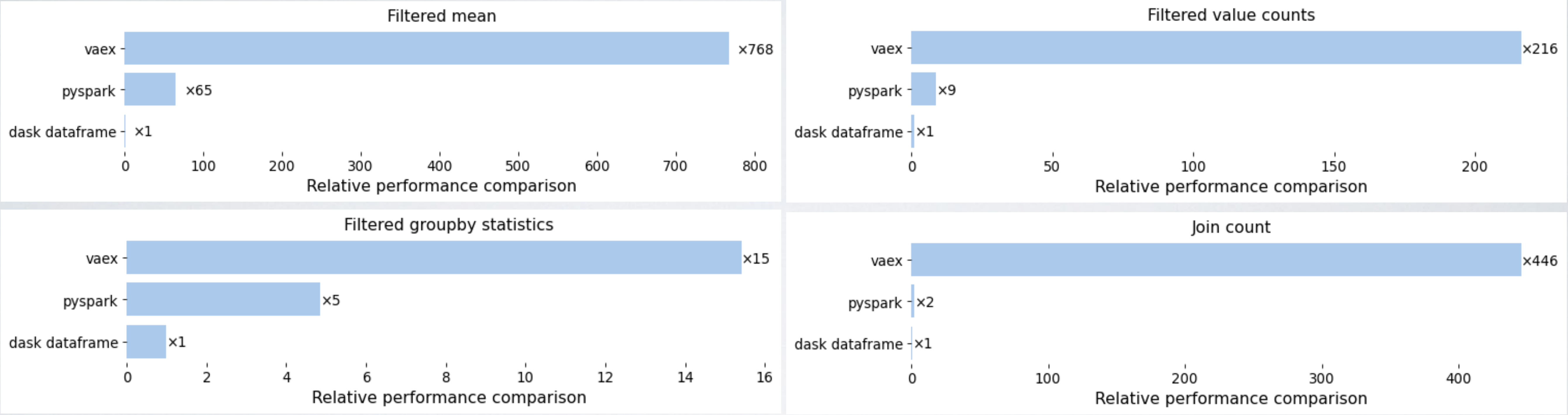
```
df2['z'] = df.x + df.y*10
```



```
df2 = {
    'data': same_data
    'state': {
        'filter': 'y < 10'
    }
    'virtual_columns': {
        'z': 'x + y*10'
    }
}
```

PERFORMANCE COMPARISONS

AWS ml.c5d.4xlarge instance: 16 vCPU, 32GB RAM, 500GB SSD

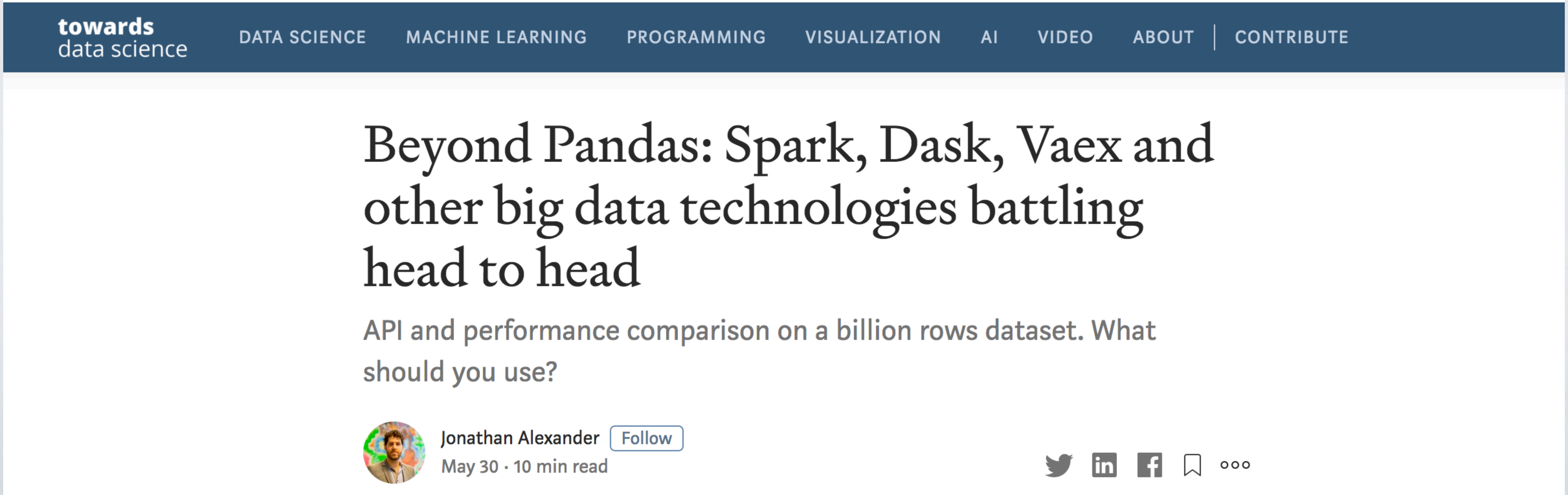


Filtered mean

Filtered value counts

Filtered groupby statistics

Join count



Beyond Pandas: Spark, Dask, Vaex and other big data technologies battling head to head

API and performance comparison on a billion rows dataset. What should you use?

Jonathan Alexander Follow

May 30 · 10 min read

“Never do a live demo”

–Many People

BENEFITS OF USING VAEX

- Makes working with large datasets simple
  - 1TB of data / 1 billion samples on a laptop
  - multiple users share the same physical memory
- Easy set-up:
  - `pip install vaex / conda install -c conda-forge vaex`
  - No need to configure and maintain a cluster
- Rapid development for ML applications, Easy deployment
- S3 support, Remote DataFrames.

ROADMAP & VISION

- Better Arrow integration
- Scikit-Learn integration via NEP13/NEP18 (scikit-learn PR #14963)
- Distributed DataFrames - Dask, Ray
- Better Cuda integration (?)

RESOURCES

- contact@vaex.io - support / consultancy / training
- https://github.com/vaexio/vaex
- @vaex_io
- Documentation: https://vaex.readthedocs.io/en/latest/
- Examples: https://github.com/vaexio/vaex-examples/