Institute for Research and Innovation in Software for High Energy Physics

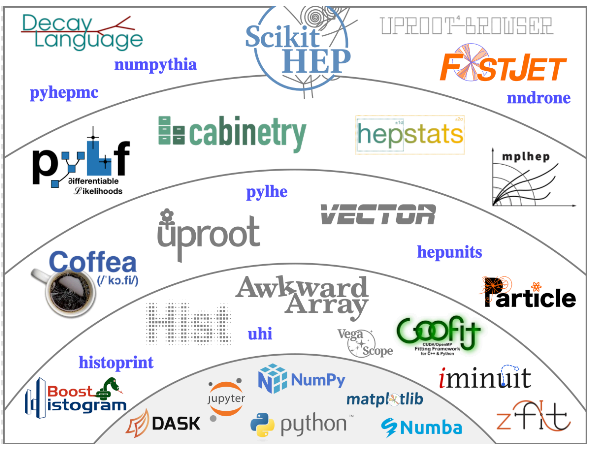
Vector: Constructors, documentation, and benchmarks

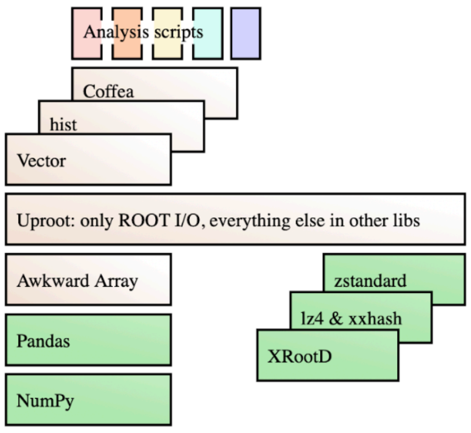
Mentors:
Dr. Henry Schreiner (Princeton University)
Dr. Jim Pivarski (Princeton University)

Saransh Chopra
Cluster Innovation Centre
20312915047

Introduction

- Vector allows working with 2D, 3D, and Lorentz vectors, to solve common physics problems in a NumPy-like way.
- Can create vectors in a variety of coordinate systems, including Cartesian, cylindrical, spherical, and any combination of these with time or proper time for Lorentz vectors.
- Comes loaded with 3 + 2 backends; a pure Python object backend, a NumPy backend, an Awkward Array backend, an Object-Numba, and an Awkward-Numba backend to leverage JIT (Just In Time) compiled calculations on vectors.

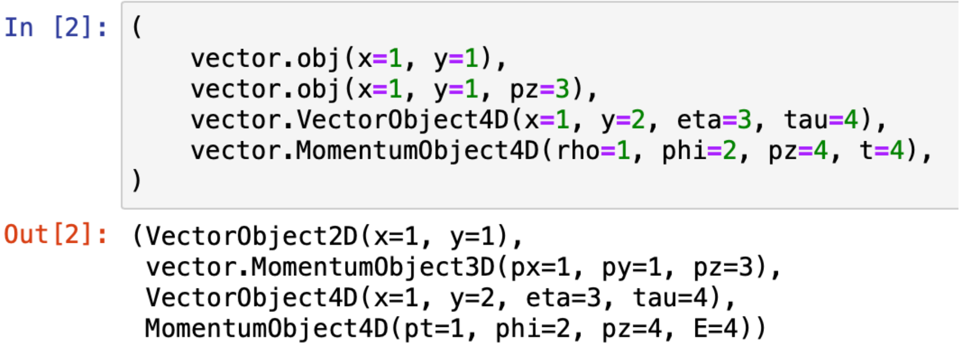




Goals and motivations

1. **Constructors**
   - The user-facing API of Vector did not expose its true API, which were the Python classes. The API provided wrapper functions (named as obj, arr, and awk) to users.
2. **Documentation**
   - Vector lacked proper user as well as API documentation and tutorials.
   - There were no detailed explanations in the existing tutorials, and most of the functions and classes did not contain docstrings.
3. **Benchmarks**
   - Scikit-HEP packages, including Vector, lacked benchmarks.
   - Benchmarking would have been a valuable addition to the Vector project, which could then be propagated to all Scikit-HEP packages in the future.

Work done

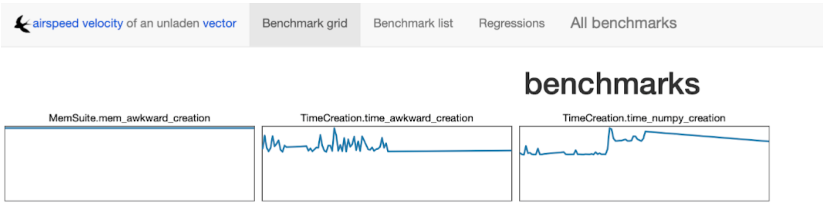


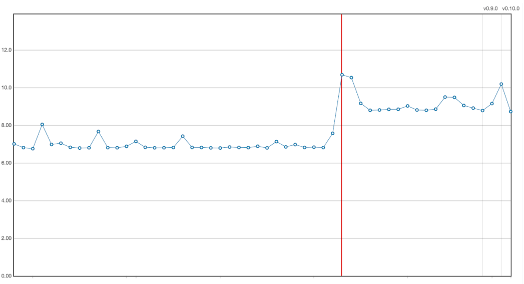
```
In [2]: (
            vector.obj(x=1, y=1),
            vector.obj(x=1, y=1, pz=3),
            vector.VectorObject4D(x=1, y=2, eta=3, tau=4),
            vector.MomentumObject4D(rho=1, phi=2, pz=4, t=4),
        )

Out[2]: (VectorObject2D(x=1, y=1),
         vector.MomentumObject3D(px=1, py=1, pz=3),
         VectorObject4D(x=1, y=2, eta=3, tau=4),
         MomentumObject4D(pt=1, phi=2, pz=4, E=4))
```

Changelog

Vector v0.9.0

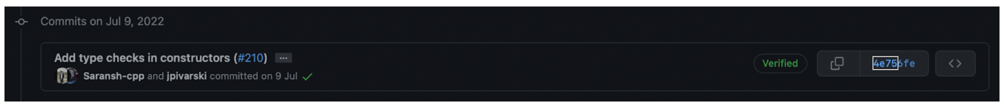
Vector v0.10.0


airspeed velocity of an unladen vector
Benchmark grid
Benchmark list
Regressions
All benchmarks
benchmarks
MemSuite.mem_awkward_creation
TimeCreation.time_awkward_creation
TimeCreation.time_numpy_creation



Vector's new documentation

Scikit-HEP's new developer page



Commits on Jul 9, 2022

Add type checks in constructors (#210)

Saransh-cpp and jpivarski committed on 9 Jul ✓

Talks

- **S. Chopra**, H. Schreiner, and J. Pivarski. Compiling Awkward Lorentz vectors with Numba. 21st International Workshop on Advanced Computing and Analysis Techniques in Physics Research (ACAT). Poster. Oct. 2022.
- **S. Chopra**, H. Schreiner, and J. Pivarski. Constructing HEP vectors and analyzing HEP data using Vector. 5th International Workshop on Python in High Energy Physics (PyHEP). Presentation. Oct. 2022.
- **S. Chopra**. Python packaging: from stone age to the future. PyDelhi workshop. Presentation. Sept. 2022.
- **S. Chopra**. Code coverage through unit tests running in sub-processes/threads: Locally and automated on GitHub. 10th Annual Conference on Python - Asia-Pacific (PyCon APAC). Presentation. Sept. 2022.
- **S. Chopra**. Code coverage through unit tests running in sub-processes/threads: Locally and automated on GitHub. 21st Annual Conference On Python - Europe (EuroPython). Presentation. July 2022.

Other achievements

- Worked as a Technical Writer at FluxML, funded by the Julia Language
- Invited to join FluxML's GitHub organisation
- Invited to be on PyBaMM's (Python Battery Mathematical Modelling) steering committee
- Received Shubhra Kar Linux Foundation Training Scholarship in the "Developer Do-Gooders" category for my work on Open Source Research Software! (500 people across the globe awarded every year)
- Was a Google Summer of Code mentor this year
- Invited to join Vector's conda feedstock (joined conda-forge's GitHub organisation)

References

Cite us!

[1] *Chopra, S.*, Schreiner, H., & Pivarski, J. (2022). Vector: vector classes and utilities. doi: 10.5281/zenodo.7054478.

[2] Rodrigues, E., & others (2020). The Scikit HEP Project – overview and prospects. EPJ Web Conf., 245, 06028.

[3] Pivarski, J., Osborne, I., Ifrim, I., Schreiner, H., Hollands, A., Biswas, A., Das, P., Roy Choudhury, S., Smith, N., & Goyal, M. (2018). Awkward Array [Computer software].