cottoncandy: scientific python package for easy cloud storage

Anwar O. Nunez-Elizalde[1]**, James S. Gao**[1]**, Tianjiao Zhang**[2]**, and Jack L. Gallant**[1,2,3]

1 Helen Wills Neuroscience Institute, University of California, Berkeley, CA, USA **2** Program in Bioengineering, UCSF and UC Berkeley, CA, USA **3** Department of Psychology, University of California, Berkeley, CA, USA

Short description

cottoncandy is a python scientific library for storing and accessing numpy array data from cloud-based object stores. cottoncandy obviates the need to write (or read) numpy array data to disk as an intermediate step to storing (or accessing) data in a cloud-based object store (e.g. AWS S3). cottoncandy is a bucket full of syntactic sugar that facilitates the use of cloud storage in typical data science workflows.

Long description

Modern big data storage solutions typically revolve around cloud-based object stores. cottoncandy is a powerful python-based tool for accessing and storing data in cloud-based object stores (e.g AWS S3, S3 API-enabled CEPH, Google Drive). The cottoncandy API is designed to simplify the use of cloud-based storage solutions in typical data science workflows (e.g. Jupyter (Pérez and Granger 2007)).

cottoncandy works by directly streaming arrays to and from memory during download and upload while minimizing memory requirements. This feature makes cottoncandy an ideal solution for data science workflows that rely on cloud-based storage. cottoncandy is optimized for accessing and storing numpy (T. E. Oliphant 2006) array data and provides support for other data formats widely used in data science (e.g. json, pickle, sparse arrays (Jones, Oliphant, and Peterson 2014)). cottoncandy also allows users to seamlessly encrypt and compress data according to their needs. Finally, cottoncandy provides a single API that supports different cloud-storage solutions as back-ends (S3 and Google Drive currently). cottoncandy can thus be used as an abstraction layer to avoid vendor lock-in though other packages also exist for this (e.g. see cloudstorage).

Acknowledgments

References

boto3

Jones, Eric, Travis Oliphant, and Pearu Peterson. 2014. "SciPy: Open Source Scientific Tools for Python."

Oliphant, Travis E. 2006. *A Guide to NumPy*. Vol. 1. Trelgol Publishing USA.

Pérez, Fernando, and Brian E Granger. 2007. "IPython: A System for Interactive Scientific Computing." *Computing in Science & Engineering* 9 (3). IEEE. https://doi.org/10.1109/MCSE.2007.53.