The Journal of Open Source Software

HyRiver: Hydroclimate Data Retriever

Taher Chegini[1], **Hong-Yi Li**[1], **and L. Ruby Leung**[2]

1 University of Houston **2** Pacific Northwest National Laboratory

DOI: DOIunavailable

Software

- Review
- Repository
- Archive

Editor: Pending Editor

Reviewers:

- @Pending Reviewers

Submitted: N/A
Published: N/A

Summary

Over the last decade, increasing availability of web services for hydrology and climatology data has facilitated publication of reproducible scientific researches in hydrological and climate studies. Such web services allow researchers to subset big databases and perform some common data processing operations on the server-side. However, implementing such services increases the technical complexity of code development as it requires sufficient understanding of their underlying protocols to generate valid queries and filters. `HyRiver` bridges this gap by providing a unified and simple Application Programming Interface (API) to web services that are based on three of the most commonly used protocols for geo-spatial/temporal data publication: REpresentational State Transfer (RESTful), Web Feature Services (WFS), and Web Map Services (WMS). `HyRiver` is a software stack consisting of the following Python packages:

- PyGeoHydro: Provides access to NWIS (National Water Information System), NID (National Inventory of Dams), HCDN-2009 (Hydro-Climatic Data Network), NLCD (National Land Cover Database), and SSEBop (operational Simplified Surface Energy Balance) databases. Moreover, it can generate an interactive map for exploring NWIS stations within a bounding box, compute categorical statistics of land use/land cover data, and plot five hydrologic signature graphs. There is also a helper function which returns a roughness coefficients lookup table for NLCD's land cover types. These coefficients can be useful for overland flow routing among other applications.
- PyGeoOGC: Generates valid queries for retrieving data from supported RESTful-, WMS-, and WFS-based services. Although these web services limit the number of features in a single query, under-the-hood, `PyGeoOGC` takes care of breaking down a large query into smaller queries according to specifications of the services. Additionally, this package offers several notable utilities, such as data re-projection and asynchronous data retrieval for speeding up sending/receiving queries.
- PyGeoUtils: Converts responses from PyGeoOGC's supported web services to geo-dataframes (vector data type) or datasets (raster data type). Moreover, for gridded data, it can mask the output dataset based on any given geometry.
- PyNHD: Provides the ability to navigate and subset National Hydrography Database (Buto & Anderson, 2020), at medium- and high-resolution, using NLDI (Hydro Network-Linked Data Index), WaterData, and TNM (The National Map) web services. Additionally, it can retrieve over 30 catchment-scale attributes from ScienceBase that are linked to the NHDPlus database via Common Identifiers (ComIDs). `PyNHD` has some additional river network tools that use NHDPlus data for routing through a river network. This flow routing module is general and accepts any user-defined transport equation for computing flow accumulation through a given river network. It sorts the river network topologically from upstream to downstream, then accumulates a given attribute based on the user-defined transport equation.
- Py3DEP: Gives access to topographic data through the 3D Elevation Program (3DEP) service (Thatcher et al., 2020). This package can pull 12 types of topographic data from the 3DEP service, such as Digital Elevation Model, slope, aspect, and hillshade.
- PyDaymet: Retrieves daily climate data as well as their monthly and annual summaries from the Daymet dataset (Thornton et al., 2020). It is possible to request data for a

single location as well as a grid (any valid geometrical shape) at 1-km spatial resolution.

Furthermore, `PyGeoOGC` and `PyGeoUtils` are low-level engines of this software stack that the other four packages utilize for providing access to some of the most popular databases in the hydrology community. These two low-level packages are generic and developers can use them for connecting and sending queries to any other web services that are based on the protocols that `HyRiver` supports.

Statement of need

Preparing input data for conducting studies, is often one of the most time-consuming steps. The difficulties of processing such input data stem from the diverse data sources and types as well as sizes. For example, hydrological modeling of watersheds might require climate data such as precipitation and temperature, topology data such as Digital Elevation Model, and a river network. Climate and topology data are available in raster format, and river network could be from a vector data type. Additionally, these datasets often have large sizes and subsetting operations can be computationally demanding. Geospatial web services can carry out subsetting and some common geographic information system (GIS) operations on server-side. However, these services usually have different specifications, thus implementing them can be technically challanging. Moreover, since the underlying protocols of these web services are under active development by organizations such as Open Geospatial Consortium, keeping track of the latest developments adds another level of complexity. `HyRiver` removes these barriers by providing consistent and simple, yet configurable, interfaces to these web services. Since these interfaces are web protocol-specific, not web service-specific, researchers can utilize `HyRiver` to access a plethora of databases that are offered through RESTFul-, WFS-, and WMS-based services.

There are several open-source packages that offer similar functionalities. For example, hydrofunctions is a Python package that retrieves streamflow data from NWIS and ulmo is another Python package that provides access to several public hydrology and climatology data. Sentinelhub-py can download and process satellite images from six data sources through Python. `Dataretrieval` gives access to some of the USGS (United States Geological Survey) databases and has two versions in R and Python. Another R Package called HydroData, provides access to 15 earth system datasets. Although there are overlaps between `HyRiver` and these packages, to the best of our knowledge, none of them offer access to the diverse data sources and types that this software stack provides.

Acknowledgements

T. Chegini was supported by the University of Houston's internal funds and the US National Science Foundation (EAR #1804560). H.-Y. Li and L. R. Leung were supported by the U.S. Department of Energy Office of Science Biological and Environmental Research as part of the Earth System Model Development and Regional and Global Model Analysis program areas through the collaborative, multi-program Integrated Coastal Modeling (ICoM) project. PNNL is operated for the Department of Energy by Battelle Memorial Institute under contract DE-AC05-76RL01830.

References

Buto, S. G., & Anderson, R. D. (2020). *NHDPlus high resolution (NHDPlus HR)—a hydrography framework for the nation*. US Geological Survey. https://doi.org/10.3133/fs20203033

Thatcher, C. A., Lukas, V., & Stoker, J. M. (2020). *The 3D elevation program and energy for the nation*. US Geological Survey. https://doi.org/10.3133/fs20193051

Thornton, M. M., Shrestha, R., Wei, Y., Thornton, P. E., Kao, S., & Wilson, B. E. (2020). *Daymet: Daily surface weather data on a 1-km grid for north america, version 4*. ORNL Distributed Active Archive Center. https://doi.org/10.3334/ORNLDAAC/1840