
dem2basin

Release 0.9.0

Daniel Hardesty Lewis

Jan 04, 2022

CONTENTS:

1	Introduction	1
1.1	Motivation	1
1.2	Limitations	1
1.3	Hydrology data processing functions	2
1.4	Core vector and raster processing functions	2
2	dem2basin	3
2.1	dem2basin package	3
3	Indices and tables	5
	Python Module Index	7
	Index	9

INTRODUCTION

`dem2basin` is Python package to simplify common surface hydrology data preparation steps.

Numerous functions are provided to: - accomplish common vector and raster processing workflows in GeoPandas and GDAL/OGR - accomplish higher-level hydrology data processing goals

1.1 Motivation

It is common for hydrologists to spend some time preparing raster and vector source data for study at the individual watershed level. Once the source data is prepared, they can then generate HAND or other common terrain model. These preparation steps are largely similar or the same each and every time the hydrologist needs generate a terrain model from source data. This package provided functions to simplify these common preparation steps.

1.2 Limitations

This packages has only been thoroughly tested against Texas Lidar raster data, NHD vector flowline and catchment data, and WBD vector HUC watershed data.

There are a number of proposed extensions, including:

- to NED 10m raster data
- to FIM / Fathom 3m raster data
- to non-WBD and non-NHD watershed data, for example internationally
- to Texas Lidar hypsography vector data
- to Texas Lidar point-cloud data

Below I only include the most thoroughly vetted `dem2basin` functions. These are the vector data processing functions.

While there are workable raster data processing functions, they are currently being significantly refactored, with nearly every existing function being deprecated in favor of entirely different approaches.

1.3 Hydrology data processing functions

- `get_hucs_by_shape` finds HUCs that intersect a study area given as a vector image
- `get_flowlines_and_representative_points_by_huc` assigns HUCs to NHD flowlines and their representative points, returning both
- `get_representative_points` retrieve representative points of flowlines and assign HUCs to these points
- `set_roughness_by_streamorder` assign Manning's n roughness value by each flowline's stream order
- `get_catchments_by_huc` assigns HUCs to NHD catchments
- `set_index_by_huc` returns a geodataframe with its index set to its HUC column
- `find_huc_level` finds the name of the HUC column of a geodataframe
- `get_nhd_by_shape` retrieves specific NHD layer masked by another geodataframe
- `get_hucs_from_catchments` dissolves NHD catchments into HUC equivalents
- `write_roughness_table` write Manning's n roughness table to CSV filename or concrete path

1.4 Core vector and raster processing functions

- `reproject_to_utm_and_buffer` finds best UTM for a geodataframe, reprojects, and then buffers it
- `find_utm` finds a single UTM CRS best suited for the geometries of a geodataframe
- `find_common_utm` determines the mode of the UTM's of the representative points of a geodataframe's geometries
- `reproject_and_buffer` reprojects geodataframe to a CRS and then buffers it
- `write_geodataframe` write geodataframe to filename or concrete path
- `to_crs` reprojects multiples geodataframes simultaneously
- `_drop_index_columns` drops columns named 'index', 'index_left', and 'index_right' either to prevent issues with geopandas functions like `geopandas.sjoin` and to clean up after some geopandas functions
- `clip_geodataframe_by_attribute` assign attribute from one geodataframe to another by their mutual index values
- `set_and_sort_index` sets a geodataframe's index to column and sorts by that column
- `read_file_or_gdf` enables functions to take either filenames or geodataframes as inputs
- `get_merged_column` returns the mutual elements of an identically names column in multiple dataframes
- `index_dataframe_by_dataframe` indexes a dataframe by another dataframe
- `skip_function_if_file_exists` wrapper to skip a particular step in a workflow if a file already exists
- `delete_file` deletes a file in all versions of Python

DEM2BASIN

2.1 dem2basin package

2.1.1 Submodules

2.1.2 dem2basin.dem2basin module

2.1.3 dem2basin.dem2basin_1m_dev module

2.1.4 Module contents

INDICES AND TABLES

- `genindex`
- `modindex`
- `search`

PYTHON MODULE INDEX

d

`dem2basin`, 3

INDEX

D

dem2basin
 module, [3](#)

M

module
 dem2basin, [3](#)