xDEM: A python library for reproducible geodetic mass balance

Erik Mannerfelt[1,2,*], Romain Hugonnet[1,2,3], **Amaury Dehecq[4]**, Andrew Tedstone[5]

1. Laboratory of Hydraulics, Hydrology and Glaciology (VAW), ETH Zurich, Zurich, CH
2. Swiss Federal Institute for Forest, Snow and Landscape Research (WSL), Birmensdorf, CH
3. LEGOS, Université de Toulouse, CNES, CNRS, IRD, UPS, F-31400 Toulouse, FR
4. Institut des Geosciences de l'Environnement (IGE), Université Grenoble Alpes, IRD, Grenoble, FR
5. University of Fribourg, Fribourg, CH

**** ranked in order of contributing lines of code/documentation !***

Fridriech Knuth
Diego Cusicanqui
Johannes Landmann
Bob McNabb
Eli Schwat
Adrien Wehrle
Shashank Bushan
David Shean
and all users

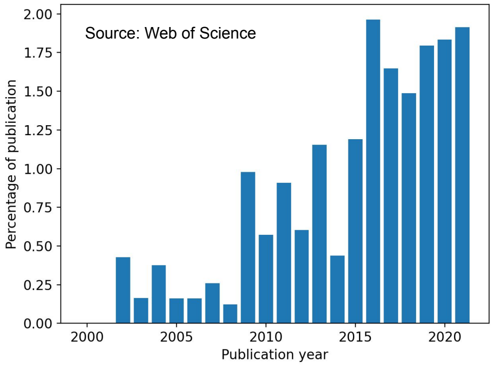
Motivations

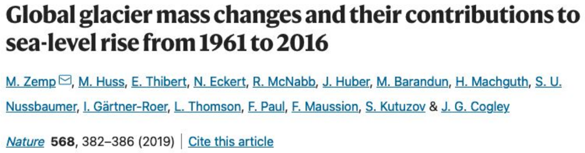
- DEMs and geodetic mass balance have become topics of increasing importance in glaciology
- But, these numbers hide several issues:
  - obsolete: method or data outdated (e.g. glacier outlines)
  - erroneous: due to the processing complexity
  - inconsistent: arbitrary criteria in the methods

=> Need for reproducibility of the results.
Yet, most results are not reproducible, because **data/code is not easily accessible.**

> “Data and code available upon request to the authors” 🤦

Percentage of “glacier” papers with words “geodetic mass balance” (~1 per week)

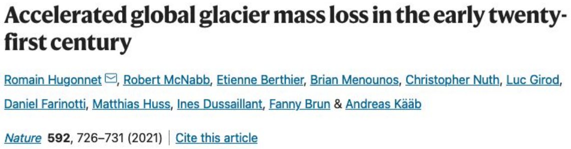




Global glacier mass changes and their contributions to sea-level rise from 1961 to 2016

M. Zemp, M. Huss, E. Thibert, N. Eckert, R. McNabb, J. Huber, M. Barandun, H. Machguth, S. U. Nussbaumer, I. Gärtner-Roer, L. Thomson, F. Paul, F. Maussion, S. Kutuzov & J. G. Cogley

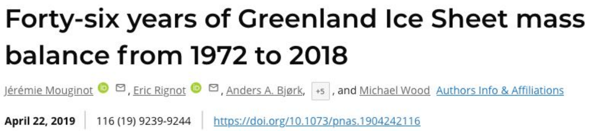
Nature **568**, 382–386 (2019) | Cite this article



Accelerated global glacier mass loss in the early twenty-first century

Romain Hugonnet, Robert McNabb, Etienne Berthier, Brian Menounos, Christopher Nuth, Luc Girod, Daniel Farinotti, Matthias Huss, Ines Dussaillant, Fanny Brun & Andreas Kääb

Nature **592**, 726–731 (2021) | Cite this article



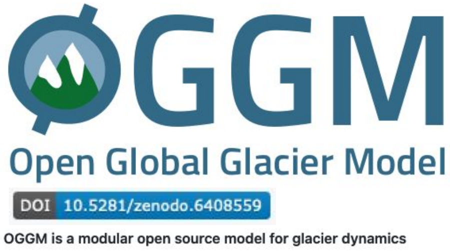
Forty-six years of Greenland Ice Sheet mass balance from 1972 to 2018

Jérémie Mouginot, Eric Rignot, Anders A. Bjørk, +5 , and Michael Wood Authors Info & Affiliations

April 22, 2019 | 116 (19) 9239-9244 | https://doi.org/10.1073/pnas.1904242116

Motivations

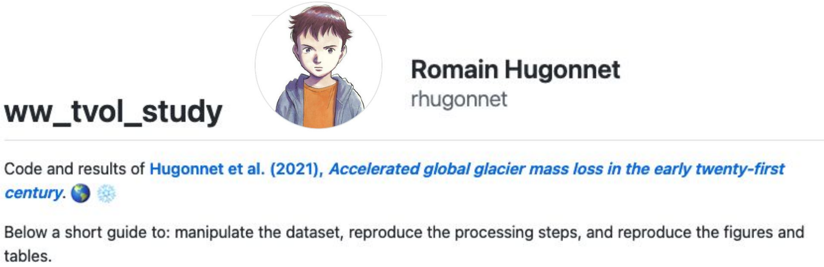
Some good examples of open-source code:



Open Global Glacier Model

DOI 10.5281/zenodo.6408559

OGGM is a modular open source model for glacier dynamics

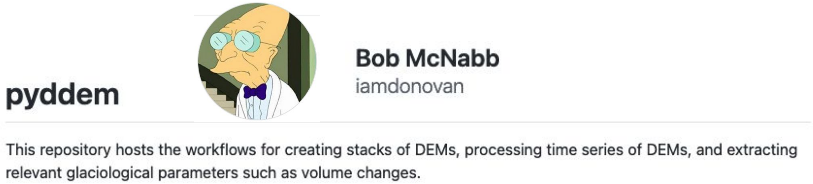


ww_tvol_study

Romain Hugonnet
rhugonnet

Code and results of Hugonnet et al. (2021), *Accelerated global glacier mass loss in the early twenty-first century.* 🌍 ❄️

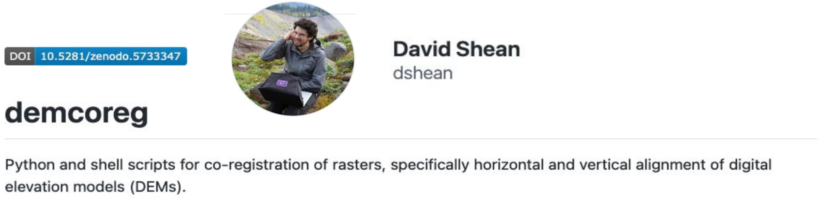
Below a short guide to: manipulate the dataset, reproduce the processing steps, and reproduce the figures and tables.



pyddem

Bob McNabb
iamdonovan

This repository hosts the workflows for creating stacks of DEMs, processing time series of DEMs, and extracting relevant glaciological parameters such as volume changes.



DOI 10.5281/zenodo.5733347

demcoreg

David Shean
dshean

Python and shell scripts for co-registration of rasters, specifically horizontal and vertical alignment of digital elevation models (DEMs).

More at https://github.com/awesome-cryosphere/cryosphere-links

A lof of re-inventing the wheel…
⇒ Why not team up and create a community tool for DEM analysis and mass balance calculation?

xDEM

- a Python module for working with DEMs.
- include typical post-processing steps such as coregistration, elevation/volume change measurements and error statistics.
- Based on well-known maintained libraries: rasterio, geopandas etc.
- Core concept: **ease of use,** flexibility and modularity, reproducibility, documentation
- Volunteer development and maintenance, by glaciologists (so far).

Code: https://github.com/GlacioHack/xdem

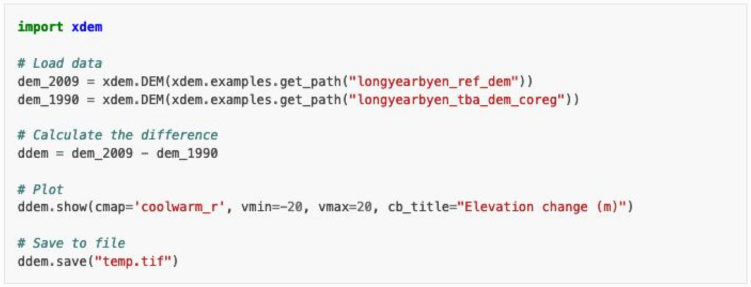
Documentation: https://xdem.readthedocs.io

DOI: 10.5281/zenodo.4809698

One-line install

```
conda install -c conda-forge xdem
```

DEM difference calculation



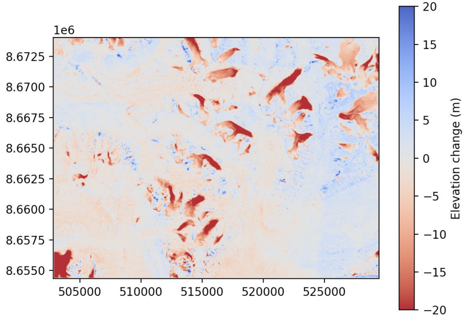
```python
import xdem

# Load data
dem_2009 = xdem.DEM(xdem.examples.get_path("longyearbyen_ref_dem"))
dem_1990 = xdem.DEM(xdem.examples.get_path("longyearbyen_tba_dem_coreg"))

# Calculate the difference
ddem = dem_2009 - dem_1990

# Plot
ddem.show(cmap='coolwarm_r', vmin=-20, vmax=20, cb_title="Elevation change (m)")

# Save to file
ddem.save("temp.tif")
```



xDEM

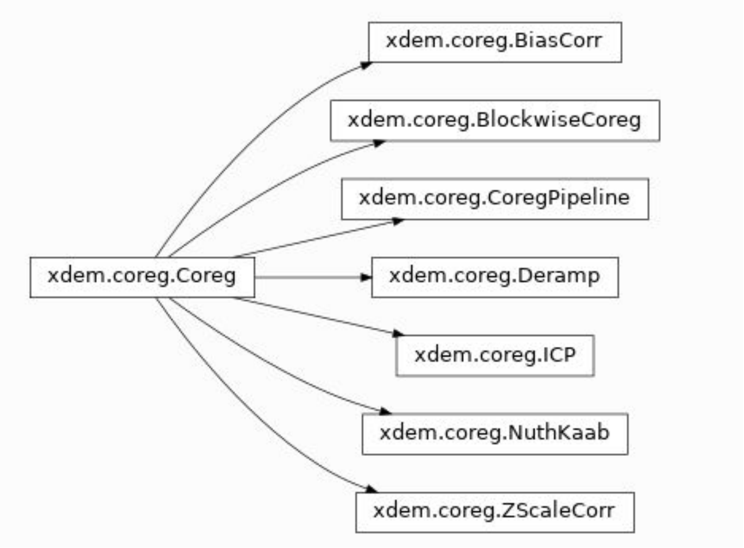
- a Python module for working with DEMs.
- include typical post-processing steps such as coregistration, elevation/volume change measurements and error statistics.
- Based on well-known maintained libraries: rasterio, geopandas etc.
- Core concept: ease of use, **flexibility and modularity,** reproducibility, documentation
- Volunteer development and maintenance, by glaciologists (so far).

Code: https://github.com/GlacioHack/xdem

Documentation: https://xdem.readthedocs.io

DOI: 10.5281/zenodo.4809698

Example with coregistration



xDEM

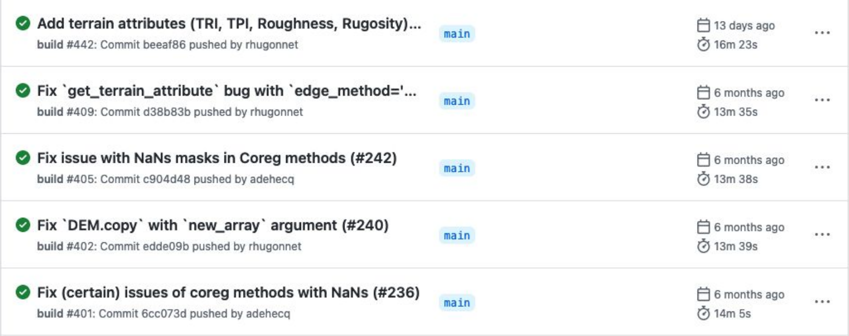
- a Python module for working with DEMs.
- include typical post-processing steps such as coregistration, elevation/volume change measurements and error statistics.
- Based on well-known maintained libraries: rasterio, geopandas etc.
- Core concept: ease of use, flexibility and modularity, **reproducibility,** documentation
- Volunteer development and maintenance, by glaciologists (so far).

Code: https://github.com/GlacioHack/xdem

Documentation: https://xdem.readthedocs.io

DOI: 10.5281/zenodo.4809698

Test suite ensures the code behaves as we “expect”



✓	**Add terrain attributes (TRI, TPI, Roughness, Rugosity)...** build #442: Commit beeaf86 pushed by rhugonnet	main	13 days ago 16m 23s	...
✓	**Fix `get_terrain_attribute` bug with `edge_method='...** build #409: Commit d38b83b pushed by rhugonnet	main	6 months ago 13m 35s	...
✓	**Fix issue with NaNs masks in Coreg methods (#242)** build #405: Commit c904d48 pushed by adehecq	main	6 months ago 13m 38s	...
✓	**Fix `DEM.copy` with `new_array` argument (#240)** build #402: Commit edde09b pushed by rhugonnet	main	6 months ago 13m 39s	...
✓	**Fix (certain) issues of coreg methods with NaNs (#236)** build #401: Commit 6cc073d pushed by adehecq	main	6 months ago 14m 5s	...

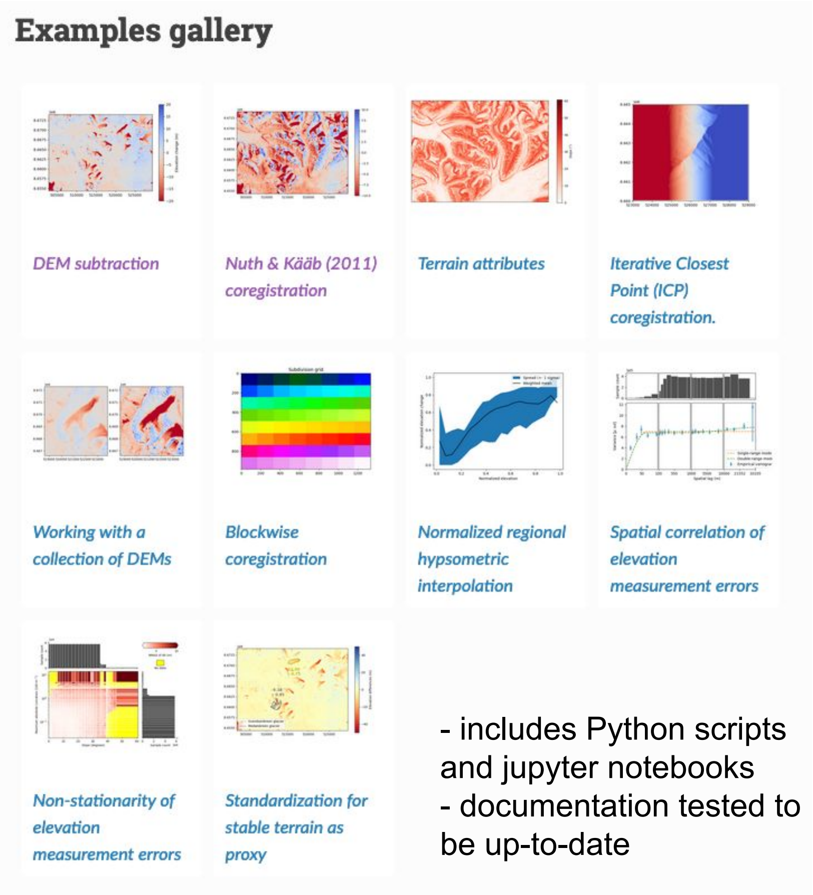
xDEM

- a Python module for working with DEMs.
- include typical post-processing steps such as coregistration, elevation/volume change measurements and error statistics.
- Based on well-known maintained libraries: rasterio, geopandas etc.
- Core concept: ease of use, flexibility and modularity, reproducibility, **documentation**
- Volunteer development and maintenance, by glaciologists (so far).

Code: https://github.com/GlacioHack/xdem

Documentation: https://xdem.readthedocs.io

DOI: 10.5281/zenodo.4809698



Examples gallery

DEM subtraction

Nuth & Kääb (2011) coregistration

Terrain attributes

Iterative Closest Point (ICP) coregistration.

Working with a collection of DEMs

Blockwise coregistration

Normalized regional hypsometric interpolation

Spatial correlation of elevation measurement errors

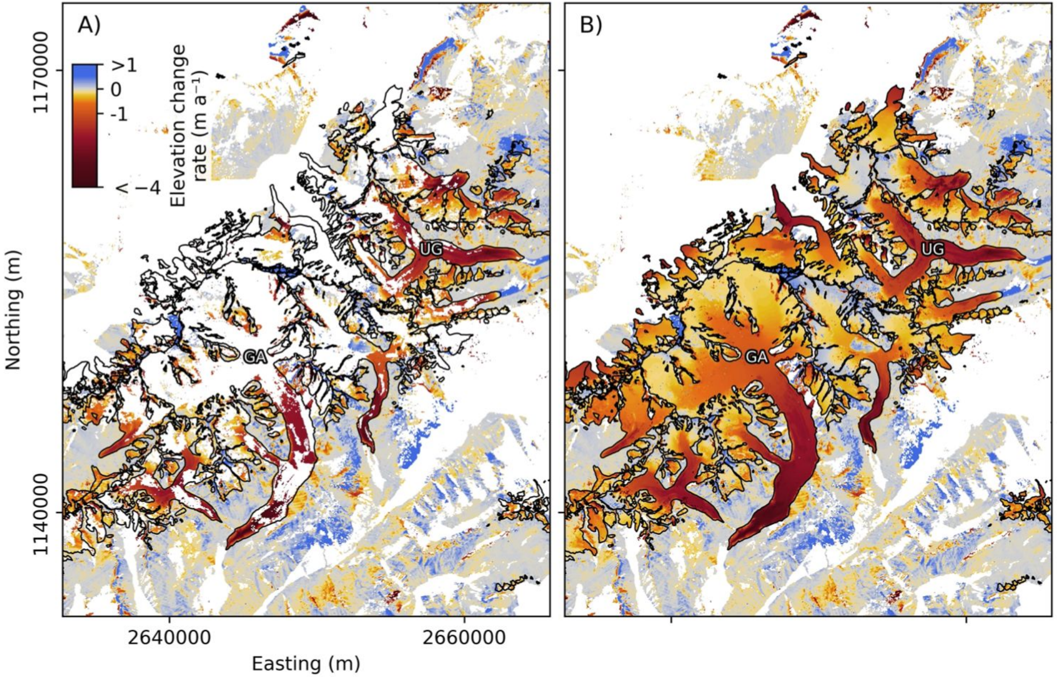
Non-stationarity of elevation measurement errors

Standardization for stable terrain as proxy

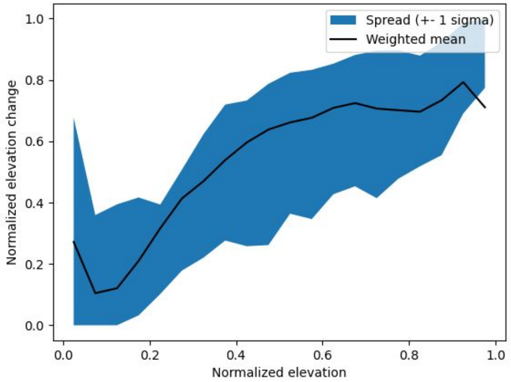
- includes Python scripts and jupyter notebooks
- documentation tested to be up-to-date

Example 1 - Gap-filling

Normalized regional hypsometric interpolation



Mannerfelt et al., TCD, 2022

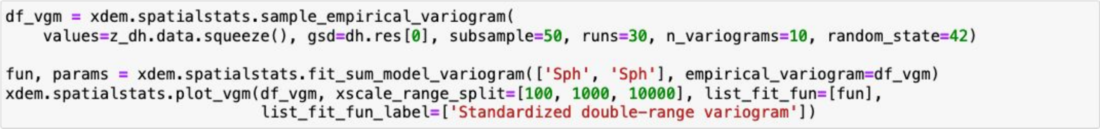


Detailed example at: https://xdem.readthedocs.io/en/doc_struc/auto_examples/plot_norm_regional_hypso.html

Example 2 - Volume change uncertainty

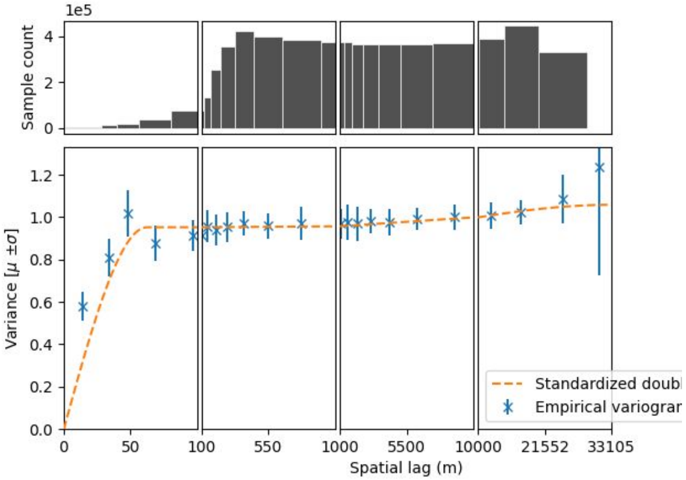
Calculating a semivariogram takes 2 lines.

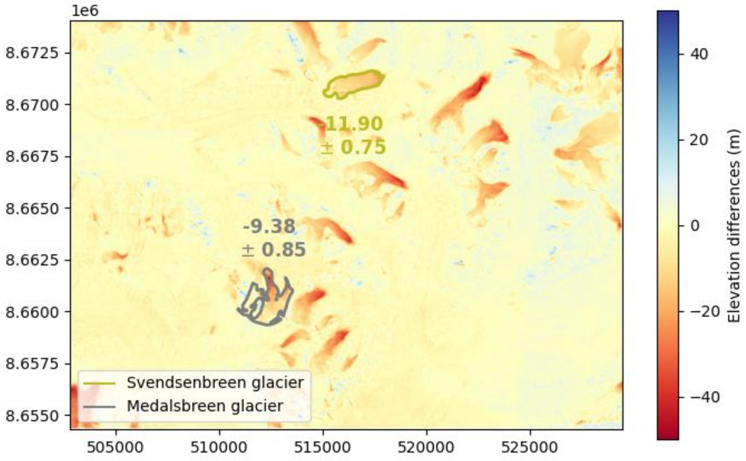
A few more lines to get the error bar for each individual glacier.



```
df_vgm = xdem.spatialstats.sample_empirical_variogram(
    values=z_dh.data.squeeze(), gsd=dh.res[0], subsample=50, runs=30, n_variograms=10, random_state=42)

fun, params = xdem.spatialstats.fit_sum_model_variogram(['Sph', 'Sph'], empirical_variogram=df_vgm)
xdem.spatialstats.plot_vgm(df_vgm, xscale_range_split=[100, 1000, 10000], list_fit_fun=[fun],
                           list_fit_fun_label=['Standardized double-range variogram'])
```





Hugonnet et al., submitted

Conclusions

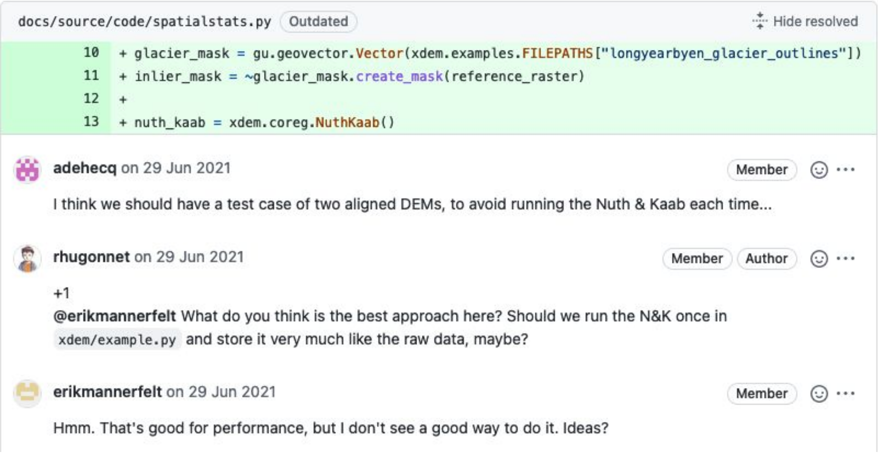
- If you're familiar with Python and work with DEMs, you know what to do !

  ```
  conda install -c conda-forge xdem
  ```

- Work in progress - long issue list (70 ! 😱), many features yet to implement.

- We really welcome any **new contributors**. No need to be a Python expert, with collaborative code, you get a lot of valuable feedback ! 🙌

Terrain attributes - ✓
Vertical datum - ✓
Coregistration - ✓
Uncertainty - ✓
Interpolation - ❌
Bias correction - ❌
Filtering - ❌
Time series analysis - ❌


docs/source/code/spatialstats.py Outdated

```
+ glacier_mask = gu.geovector.Vector(xdem.examples.FILEPATHS["longyearbyen_glacier_outlines"])
+ inlier_mask = ~glacier_mask.create_mask(reference_raster)
+
+ nuth_kaab = xdem.coreg.NuthKaab()
```

adehecq on 29 Jun 2021 Member

I think we should have a test case of two aligned DEMs, to avoid running the Nuth & Kaab each time...

rhugonnet on 29 Jun 2021 Member Author

+1
@erikmannerfelt What do you think is the best approach here? Should we run the N&K once in `xdem/example.py` and store it very much like the raw data, maybe?

erikmannerfelt on 29 Jun 2021 Member

Hmm. That's good for performance, but I don't see a good way to do it. Ideas?

And it goes on and on !

Thank you for your attention !

Code: https://github.com/GlacioHack/xdem

Documentation: https://xdem.readthedocs.io