

CGC: an open-source Python module for geospatial data clustering

Francesco Nattino

ITC – University of Twente

17 November 2022

netherlands
eScience center

Contributors



Francesco Nattino



Ou Ku



Meiert Grootes



Raul Zurita-Milla



Serkan Girgin

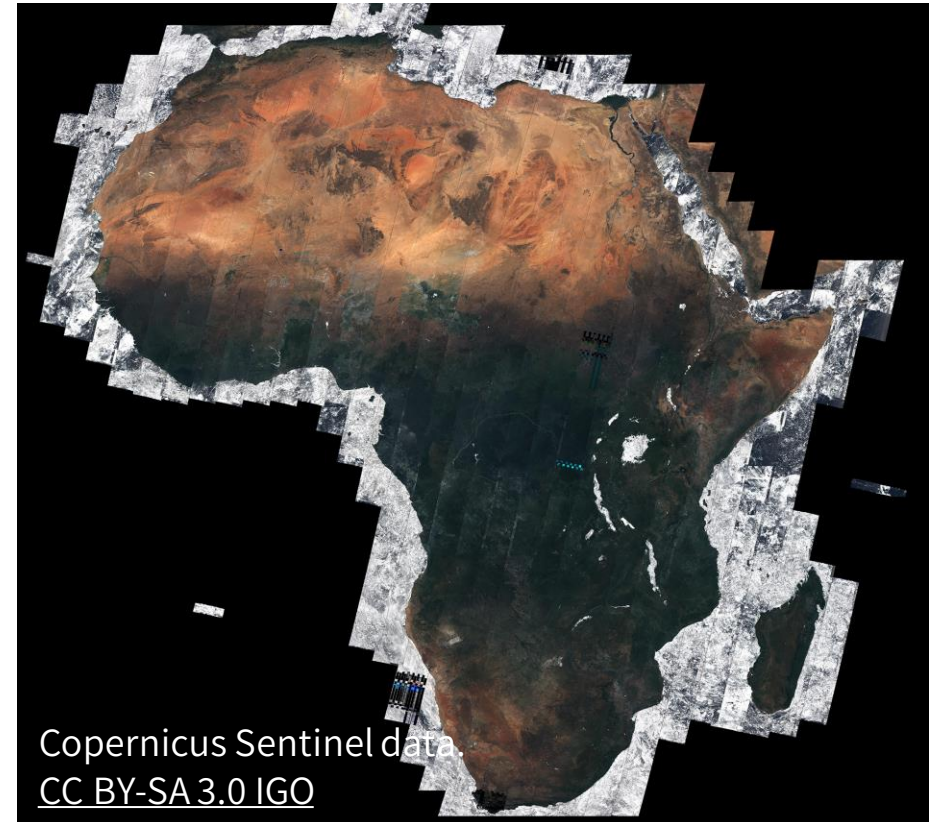


Emma Izquierdo-
Verdiguier



Motivation and challenges

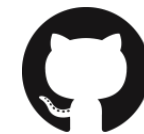
Space \ Time	T1	T2	T3	...	TN
S1	1	15	3	...	1
S2	2	65	4	...	21
S3	5	70	3	...	34
S4	7	115	4	...	2
...
SM	3	67	1	...	40



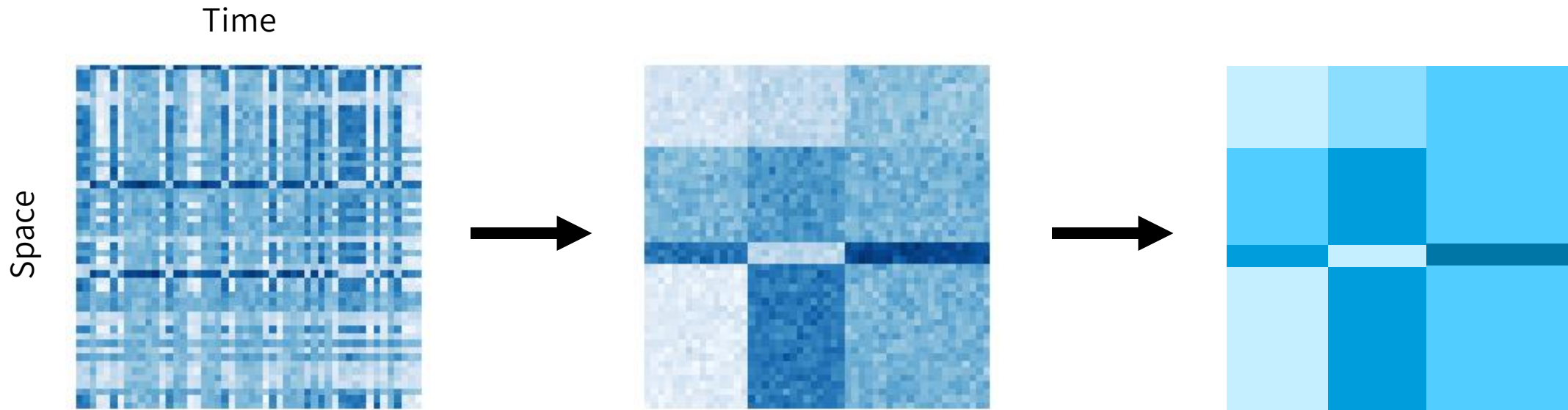
Clustering spatio-temporal data

Clustering Geo-data Cubes (CGC)

- Iterative **co-clustering** (2D) and **tri-clustering** (3D) algorithms.
- Optional cluster refinement step.
- Implementations for **single machine** and **computer clusters** (Dask).
- Generic partitional clustering algorithms, targeting geospatial data.

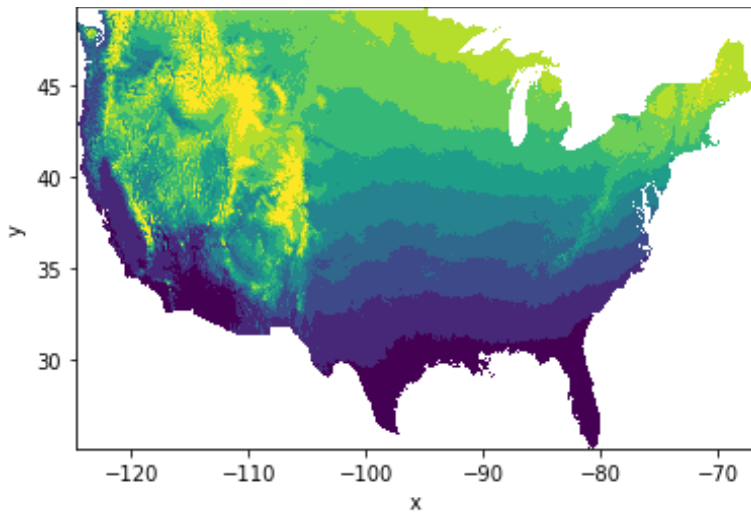


Clustering spatio-temporal data

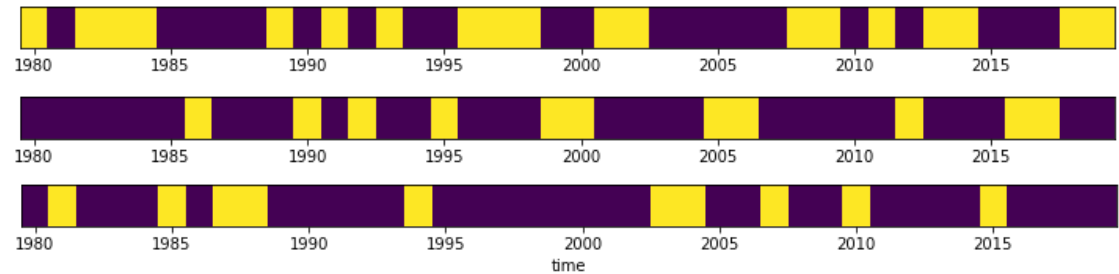


Use case: Identifying phenoregions

- Gridded spring indices (day of first leaf and bloom) at **continental scale** and **high resolution** (1 km) for three plant species.
- Extract phenological patterns (**co-clustering**) and correlation between indices (**tri-clustering**).



Space clusters



Time clusters

Tutorials

CGC Tutorial
latest

Search docs

CONTENTS

- Co-clustering
 - Introduction
 - The data
 - Imports and general configuration
 - Reading the data
- The co-clustering analysis
- K-means refinement
- Tri-clustering

Read the Docs

» Co-clustering [Edit on GitHub](#)

Run this notebook [launch](#) [binder](#) or view it on [Github](#).

Co-clustering

Introduction

This notebook illustrates how to use [Clustering Geo-data Cubes \(CGC\)](#) to perform a co-clustering analysis of geo-spatial data.

main ▾ tutorial-cgc / notebooks / coclustering.ipynb [Go to file](#) [...](#)

[fnattino](#) add triclustering notebook Latest commit 3adfb27 5 days ago [History](#)

1 contributor

1321 lines (1321 sloc) | 313 KB [<>](#) [Files](#) [Raw](#) [Blame](#) [Icons](#) [Edit](#) [Trash](#)

CGC Tutorial
latest

Search docs

CONTENTS

- Co-clustering
 - Introduction
 - The data
 - Imports and general configuration
 - Reading the data
 - The co-clustering analysis
 - K-means refinement
 - Tri-clustering
- The co-clustering analysis
- K-means refinement
- Tri-clustering

Read the Docs

v: latest ▾

Spatial clusters can also be visualized after 'unstacking' the location index that we have initially created, thus reverting to the original (x, y) coordinates:

```
[27]: space_clusters_xy = space_clusters.unstack('space')
space_clusters_xy.plot.imshow(
    x='x', y='y', levels=range(num_space_clusters+1)
)
```

[27]: <matplotlib.image.AxesImage at 0x7f9366b8de10>

space cluster

File Edit View Run Kernel Tabs Settings Help

Launcher [x](#) [coclustering.ipynb](#)

Filter files by name

Name	Last Modified
/ notebooks /	
coclusterin...	5 minutes ago
triclusterin...	5 minutes ago

Co-clustering

Introduction

This notebook illustrates how to use [Clustering Geo-data Cubes \(CGC\)](#) to perform a co-clustering analysis of geo-spatial data.

The use of co-clustering overcomes some of the limitations of traditional one-dimensional clustering techniques (such as k-means). For spatio-temporal data, traditional clustering allows the identification of spatial patterns across the full time span of the data, or temporal patterns for the whole spatial domain under consideration. With co-clustering, the space and time dimensions are simultaneously clustered. This allows the creation of groups of elements that behave similarly in both dimensions. These groups are called co-clusters and typically represent a region of the study area that has similar temporal dynamics for a subset of the study period.

In this notebook we illustrate how to perform a co-clustering analysis with the CGC package using a phenological dataset representing the day of the year of first leaf appearance in the conterminous United States. For more information about this dataset please check [here](#).


Simple 0 1 Python 3 (ipykernel) | Idle Mem: 154.75 / 8192.00 MB Mode: Command Ln 1, Col 1 coclustering.ipynb

Let's stay in touch



 www.eScienceCenter.nl

 f.nattino@esciencecenter.nl

 +31 (0)20 460 4770