Understanding Dynamics with Advanced Time-Series Processing Techniques

2024 IEEE IGARSS – Time-Series Tutorial




Charlotte Pelletier[1] Marc Rußwurm[2] Dainius Masiliunas[2] Jan Verbesselt[3]

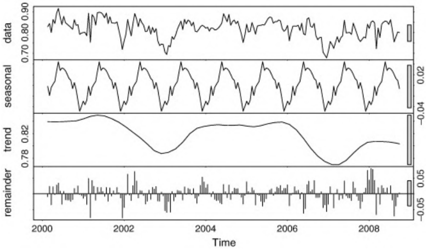
(1) Univ. Bretagne Sud / IRISA, (2) Wageningen University, (3) Belgian Science Policy Office

Contacts: charlotte.pelletier@univ-ubs.fr and dainius.masiliunas@wur.nl

About us

We are working on time series analysis

- ◇ unsupervised and supervised learning
- ◇ using machine learning and deep learning techniques
- ◇ in various contexts: large-scale mapping, low supervision, multimodal, etc.



Seasonal-trend decomposition

[1] Verbesselt, J., Hyndman, R., Newnham, G., & Culvenor, D. (2010). Detecting trend and seasonal changes in satellite image time series. *Remote Sensing of Environment*, 114(1), 106-115.

The BreizhCrops benchmark datasets
`https://breizhcrops.org/`

About you

We would like to know you better and learn about your expertise.

Please go to `menti.com`
Enter the following code to participate in the survey: 5808 4593

Link to the poll.

Link to the results.

Tutorial Outline

July 7, 2024, from 09:00 to 12:30

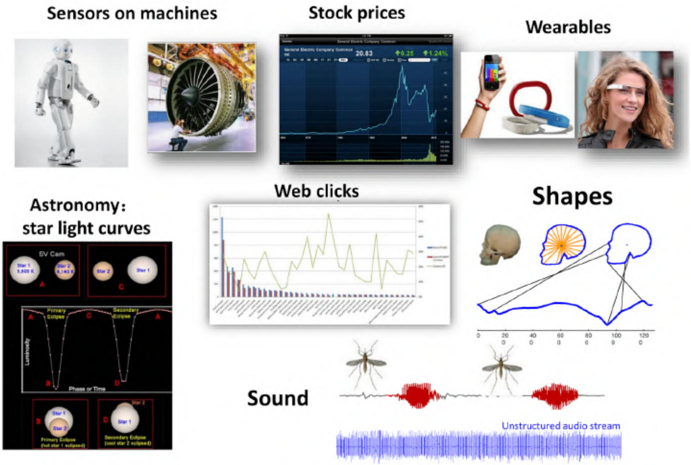
Timeline	Topic
09:00 - 09:15	Part I. Introduction to Time-Series Analysis
09:15 - 09:45	Part II. Time-series segmentation and break detection
09:45 - 10:15	Part III. Deep learning techniques for satellite image time series
10:15 - 10:45	Break
10:45 - 12:00	Practical session: (i) break detection, and (ii) deep learning

Links to all materials available: `https://dl4sits.github.io/igarss2024`

Content

Time series

- ◇ describe the evolution of a process over time
- ◇ are ubiquitous: daily life, medical, food security, financial, environmental...
- ◇ increase in quantity and velocity



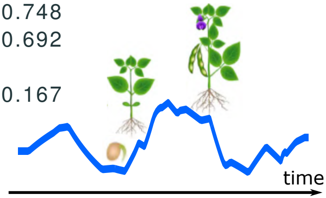
Time Series

Formally, a time series

- is a sequence of values **ordered** in time
- either univariate or multivariate
- not necessarily regularly-sampled

An example univariate time series:

time	value
t1	0.236
t2	0.563
t3	0.748
t4	0.692
...	
tL	0.167



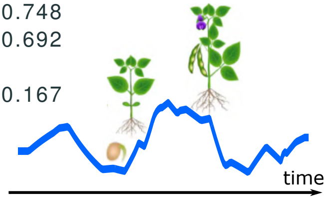
Time Series

Formally, a time series

- is a sequence of values **ordered** in time
- either univariate or multivariate
- not necessarily regularly-sampled

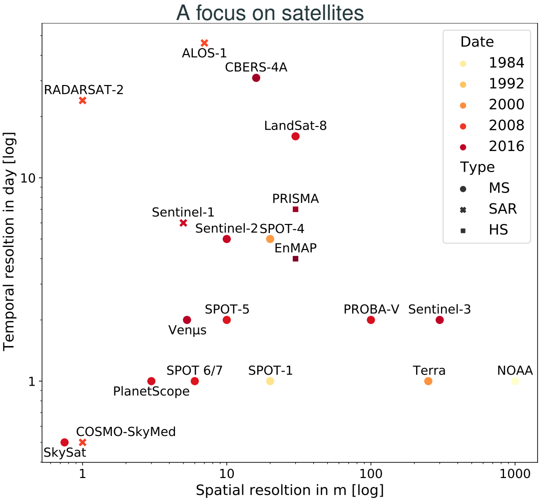
An example univariate time series:

time	value
t1	0.236
t2	0.563
t3	0.748
t4	0.692
...	
tL	0.167



Time series analysis includes

- **unsupervised** techniques – no prior knowledge on the data
  - clustering: grouping similar time series together
  - retrieval: finding similar time series
  - **segmentation**: dividing a time series into "homogeneous" subseries
- **supervised** techniques – requires labelled data (examples)
  - forecasting: predicting future values
  - (extrinsic) regression: predicting a continuous scalar variable
  - **classification**: predicting a category that describes the time series



- ◇ a data increase
- ◇ acquisition of data in various modalities
- ◇ open access to satellite imagery and archives

Satellite image Time Series (SITS) are

- A stack of images of the same region acquired over time
- that forms a complex datacube
- (possibly) irregularly sampled.

Sentinel-2 images over Brittany, France

Time Series in Remote Sensing

Satellite image Time Series (SITS) are

- ◇ A stack of images of the same region acquired over time
- ◇ that forms a complex datacube
- ◇ (possibly) irregularly sampled.

Sentinel-2 images over Brittany, France

The example of crop-type identification

Can you guess where rapeseed grew in this image from May?

http://www.cesbio.ups-tlse.fr/multitemp/?p=1192

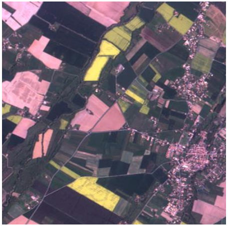
Time Series in Remote Sensing

Satellite image Time Series (SITS) are

- ◇ A stack of images of the same region acquired over time
- ◇ that forms a complex datacube
- ◇ (possibly) irregularly sampled.

Sentinel-2 images over Brittany, France

The example of crop-type identification



Rapeseed in April

Can you guess where rapeseed grew in this image from May?

http://www.cesbio.ups-tlse.fr/multitemp/?p=1192

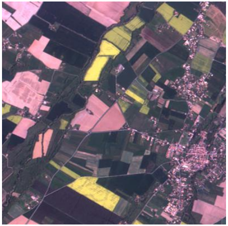
Time Series in Remote Sensing

Satellite image Time Series (SITS) are

- A stack of images of the same region acquired over time
- that forms a complex datacube
- (possibly) irregularly sampled.

Sentinel-2 images over Brittany, France

The example of crop-type identification



Rapeseed in April

Can you guess where rapeseed grew in this image from May?

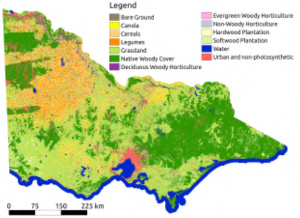
http://www.cesbio.ups-tlse.fr/multitemp/?p=1192

SITS are crucial to monitoring the Earth's dynamics over large areas

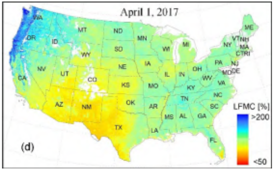
- landscape changes
- vegetation monitoring
- landslide analysis

Satellite Image Time Series

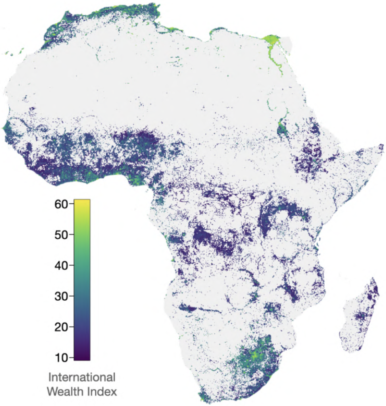
SITS have **various applications** in agriculture, for land cover land use mapping, and soil moisture, vegetation condition or socio-economic indicator estimation.



An example land-cover map of Victoria State, Australia
`https://tinyurl.com/yc6juv6d`



Life fuel moisture content (LFMC) estimation [1]



International wealth index (IWI) predictions [2]

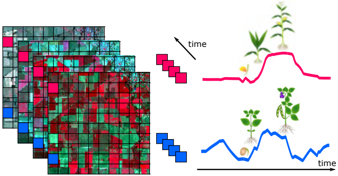
[1] Zhu, L., Webb, G. I., Yebra, M., Scortechini, G., Miller, L., & Petitjean, F. (2021). Live fuel moisture content estimation from MODIS: A deep learning approach. *ISPRS Journal of Photogrammetry and Remote Sensing*, 179, 81-91.

[2] Pettersson, M. B., Kakooei, M., Ortheden, J., Johansson, F. D., & Daoud, A. (2023). Time series of satellite imagery improve deep learning estimates of neighborhood-level poverty in Africa. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence* (pp. 6165-6173).

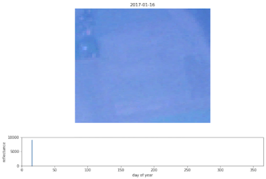
How to process SITS?

From satellite images to time series

Pixel-based analysis



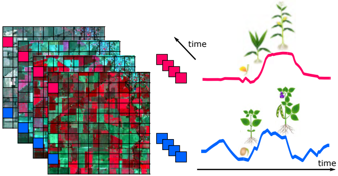
Object-based analysis, *e.g.*, averaging the reflectance values within an agricultural parcel



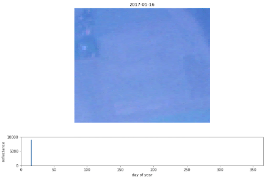
How to process SITS?

From satellite images to time series

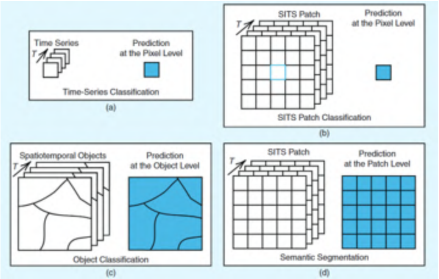
Pixel-based analysis



Object-based analysis, *e.g.*, averaging the reflectance values within an agricultural parcel



A taxonomy based on the types of input-output



Time-first, space-later [1]

- ◇ modelling temporal correlations
- ◇ learning dynamics
- ◇ ensuring temporal consistency

[1] Camara, G., Assis, L. F., Ribeiro, G., Ferreira, K. R., Llapa, E., & Vinhas, L. (2016, October). Big earth observation data analytics: Matching requirements to system architectures. In *Proc. of the 5th ACM SIGSPATIAL International Workshop on Analytics for Big Geospatial Data* (pp. 1-6).

Working with SITS

The tutorial focuses on approaches to automatically extract knowledge from SITS.

The tutorial focuses on approaches to automatically extract knowledge from SITS.

A (very) brief supplement

1. Gather satellite images
   - THEIA, USGSS, *etc.*
   - Sentinels Scientific Data Hub, Copernicus DIAS
   - cloud platforms: GEE, Amazon, Microsoft Planetary Computer
2. Prepare the data
   - coregistration, atmospheric correction
   - gapfilling
   - normalisation
3. Run your analysis and evaluate it

The tutorial focuses on approaches to automatically extract knowledge from SITS.

A (very) brief supplement

1. Gather satellite images
   - THEIA, USGSS, *etc.*
   - Sentinels Scientific Data Hub, Copernicus DIAS
   - cloud platforms: GEE, Amazon, Microsoft Planetary Computer
2. Prepare the data
   - coregistration, atmospheric correction
   - gapfilling
   - normalisation
3. Run your analysis and evaluate it

What about a pipeline?

- develop your own Python/R code
- use dedicated libraries, *e.g.*, snap, OTB, TorchGeo
- use existing frameworks, *e.g.*, `iota2` (Python/C++) or R-SITS

Working with SITS

The tutorial focuses on approaches to automatically extract knowledge from SITS.

A (very) brief supplement

1. Gather satellite images
   - THEIA, USGSS, *etc.*
   - Sentinels Scientific Data Hub, Copernicus DIAS
   - cloud platforms: GEE, Amazon, Microsoft Planetary Computer
2. Prepare the data
   - coregistration, atmospheric correction
   - gapfilling
   - normalisation
3. Run your analysis and evaluate it

What about a pipeline?

- develop your own Python/R code
- use dedicated libraries, *e.g.*, snap, OTB, TorchGeo
- use existing frameworks, *e.g.*, `iota2` (Python/C++) or R-SITS

Let us now move to the first focus of the tutorial: **break detection**!

Link for all the tutorial's material: `https://dl4sits.github.io/igarss2024`