UNIVERSITÉ DE GENÈVE
Faculté des sciences de la société
Certificat complémentaire en géomatique

UNEP GRID Geneva

Mémoire du certificat complémentaire en géomatique – Janvier 2024

Land use mapping in the Geneva region using Data Cube and Machine Learning

Rapport de stage
Janvier 2024

Asma Chafter

Sous la direction de Gregory Giuliani
Jury : Denisa Rodila

Contents

Remerciements

Je tiens à exprimer ma gratitude envers le Professeur Gregory Giuliani, qui m'a offert l'opportunité exceptionnelle de m'immerger dans un sujet d'une actualité prégnante. Sa confiance en mes compétences m'a non seulement honoré, mais elle a également été la source d'une motivation inépuisable tout au long de ce parcours de recherche.

Je le remercie pour sa générosité, sa disponibilité et son soutien sans faille qui ont été un catalyseur essentiel dans l'approfondissement de mes connaissances sur la couverture et l'utilisation du sol dans le canton genevois, tout en explorant les intrications complexes des modèles de Machine Learning.

Résumé

This work was carried out as part of an internship at UNEP/Grid-Geneva with a view to obtaining a complementary certificate in geomatics at the University of Geneva.
The aim of this work is to map Land Use using Data Cube and Machine Learning. The project involves exploring the use of Amazon Web Services (AWS) and testing the SITS (Satellite Image Time Series Analysis on Earth Observation Data Cubes) model to generate detailed land-use maps for the canton of Geneva.
Land Use data is one of the crucial datasets in geospatial data. Geospatial data is produced from satellite imagery. In Switzerland, these data are supplied by the Federal Statistical Office and are produced by photointerpretation on a 100mX100m grid.

1. Introduction

In 2016, the Swiss Federal Department of Environment, Transport, Energy, and Communication (DETEC) outlined its strategy, declaring the goal of transforming Switzerland into a sustainable country by 2030 while upholding attractiveness, economic competitiveness, and high quality of life (DETEC, 2016; Giuliani et al., 2022). This strategy aligns with the United Nations' 2030 Agenda for Sustainable Development, which presents a global initiative urging nations to pursue economic, social, and environmental sustainability by 2030 (DESA, 2016; Owers et al., 2021). The Sustainable Development Goals (SDGs) identify and trace harmful practices, intervening where necessary for greater sustainability (UNGA, 2015; Owers et al., 2021). The SDGs comprise 17 goal headlines, 169 specific targets and 231 indicators to assess progress, including indicators on land cover and land use.

Indeed, since the industrial revolution (~1750), human-induced environmental changes have profoundly affected the world, both locally and globally (Thomas & Giuliani, 2022). Human land use appears to be one of the main contributors to greenhouse gas emissions and biodiversity loss (Foley et al., 2005). The increasing interdependence between human well-being and the Earth's ecosystems underlines the urgent need to embrace sustainable practices. Land cover conversion due to human use appears to be one of the main factors behind these environmental challenges. While a consensus exists on the need to curb land degradation, projections indicate an expected increase throughout the 21st century, regardless of the development scenario (UNCCD, 2022; Giuliani, 2023). During the last 50 years, the growing demand for natural resources has progressively altered most of the planet's landscapes, affecting 83% of the terrestrial land surface and have degraded about 60% of services provided by ecosystems (Giuliani et al., 2022). Land degradation has almost reached a tipping point and is forecast to compromise the well- being of 3.2 billion people by 2050 (IPBES,2018; Giuliani et al., 2022). Furthermore, over the period 1985-2009, 15% of the country's land area was transformed (SFSO, 2013. Giuliani et al, 2022). Urbanization has led to the expansion of housing and urban areas, farmland has shrunk, forests have expanded, and glaciers have retreated (SFSO, 2001; FSO, 2018). Hence, the Swiss government has prioritized environmental protection one of its core missions, aiming to encourage and adopt more sustainable approaches to the use of natural resources (Giuliani et al., 2022). Thus, measures such as protecting natural resources, planning urban development, reducing greenhouse gas emissions, preserving water quality, maintaining biodiversity and ecosystem services, protecting soil, and preserving rural areas are crucial (Giuliani et al., 2022; Rounsevell et al., 2006).

To understand the environmental change drivers, it is critical to provide an accurate overview of land surface distribution and land use changes for socio-economic purposes, as well as to precisely map variations in biophysical cover (e.g., Land Use and Cover - LUC) (Comberet al., 2005; Giuliani, 2023). Land cover (LC) relates to the land surface cover, covering vegetation, urban infrastructure, water, bare soil or other (Government of Canada, 2015). It is recognized as an Essential Climate Variable (ECV) because of its relevance in assessing changes in the climate system and producing thematic maps (Thomas & Giuliani, 2023). The land cover recognition, delimitation and mapping play a key role in worldwide monitoring projects, resource management and planning efforts (Government of Canada, 2015). On the other hand, land use (LU) refers to the purpose the land serves, such as for recreation, wildlife habitat or agriculture (Government of Canada, 2015). Land management applications encompass both initial map-

ping and continual monitoring (Government of Canada, 2015). Updated information is essential to understand present land cover and use, and to identify changes in land use over time. This enables the stakeholders and decision-makers to promote appropriate action to balance conservation efforts, resolve land-use conflicts and manage development pressures (Thomas & Giuliani, 2023; Chaves et al., 2023; Government of Canada, 2015; Grêt-Regamey et al., 2017). Therefore, changes in land cover (LC) and land use (LU) are regarded as significant tangible indicator of human impact on the environment (UN, 2012; Szantoi et al., 2020).

LU/LC is of pivotal concern in addressing major global challenges such as climate change, biodiversity conservation, food security, sustainable management, and ecosystem services (Giuliani et al., 2022; Giuliani, 2023). The impacts of land-use-related changes on the environment (pollution, climate change, etc.) and human activities (food security, economic development, etc.) are considerable. Thus, the diversity of Land use changes changes and their consequences require accurate, timely and systematic monitoring and analysis. This concern is embedded in regional and global policies, including the 2030 Agenda for Sustainable Development, and addresses 14 of the 17 Sustainable Development Goals (Guo, 2020; Mushtaq et al. 2022).

Conventional approaches to gather information on Land Use and Land Cover (LUC) include field surveys (D'Andrimont et al., 2021), visual interpretation of high-resolution images (Hadi et al., 2022), crowdsourcing (Fonte et al., 2019), multi-source data integration (Delgado and Valcárcel, 2022), and analysis of remotely sensed data (Friedl et al. 2022). Land use assessment and management have benefited greatly from the wealth of Earth Observation (EO) data collected by satellites since the 1970s (Wulder et al. 2019). Satellites, such as those operated by the European Space Agency's (ESA) Climate Change Initiative (CCI) and the Moderate Resolution Imaging Spectroradiometer (MODIS) program, play a crucial role in monitoring and analyzing land use patterns on a global scale (Plummer, Lecomte and Doherty 2017; Sulla-Menashe et al. 2019). Nevertheless, national and regional LUC outputs with high thematic accuracy are still lacking, and many regional or local studies still rely on global datasets (Phan et al., 2022; Rwanga et al., 2017; Giuliani, 2023). The accuracy of LUC data is crucial, as it directly affects the performance of downstream applications, such as ecosystem, hydrological or climate models. Consequently, it is imperative to evolve accurate and consistent national and regional products to adequately map and monitor LUC change (Giuliani, 2023).

1.1. Land Use Data in Switzerland

In Switzerland, the official LUC data source derives either from the Topographic Landscape Model (TLM) supplied by Swisstopo or from the Land Use Statistics (Arealstatistik) provided by the Federal Statistical Office (FSO). The Topographic Landscape Model comprises high-resolution vector dataset depicting various landscape features (swissTLM3D) and the primary surface (VECTOR25), derived from the aerial image's interpretation. The Land Use Statistic (Arealstatistik) data are generated by visually interpreting aerial images and allocating LC and LU categories to the bottom-left corner of each sample point within a standardized 100m grid cell, corresponding to more than 4 million points over the country following three nomenclatures: standard (72 categories); Land Use (46 categories) and Land Cover (27 categories) over four-time periods (1979 – 85, 1992-97, 2004-09, 2013-18). These databases have limitations for consistent environmental monitoring, while being useful and more accurate than current used classifications (Giuliani et al. 2022). In particular, SwissTLM3D lacks definitions for key classes such as grassland and agriculture (Price et al., 2023). In addition, swissTLM3D and Arealstatistik have limited spatial (1hectare) and temporal (i.e. updated every 6 years) resolutions. Therefore,

relying on these survey-based methods prevents an accurate representation of field realities for research and policymakers. This limitation hinders the identification and quantification of determining factors and rates of change (Thomas & Giuliani, 2023). Thus, the acquisition of a high-resolution annual LUC dataset would be invaluable in better understanding these factors and the dynamics of change.

A proficient operational monitoring of Land Use and Land Cover change (LU/LCC) demands consistent and reproducible mapping at regular intervals, with a spatial resolution adapted to the requirements of political decision-makers and strategic management objectives (Giuliani, 2023). Despite attempts to develop methodologies that seamlessly integrate new information, gaps remain in these approaches (Owers et al., 2021). The current LULC information often fails to meet the operational criteria required for national policy objectives, and lacks comparability between countries (Metternicht et al., 2020). To quantify land use and land cover (LU/LC) accurately, two key components are fundamental: (1) nationally significant high-resolution time series and (2) methodologies to establish dynamics and identify patterns, including trends, breaks and disturbances (Giuliani, 2023; Giuliani et al., 2022). The scarcity of national Land-Use datasets and challenges in integrating regional and global data compromise the effectiveness of environmental monitoring. This hampers governments' decision-making on the impact of human activities, strategic planning, sustainable use of resources, biodiversity conservation, etc.

1.2. EO Data Science

Over the last decade, advances in Earth Observation (EO) data science have led to major breakthroughs, paving the way for improved technologies and the production of more accurate Land Use Change (LUC) maps. These improvements permit the generation of accurate annual data, as well as consistent maps at high spatial (10 m) and thematic resolution (Pandey et al. 2019; Venter et al. 2022; Giuliani, 2023). The increasing abundance of EO data has opened the door to the production of more accurate land-use change (LUC) maps. These advances, using Data Cube, Machine Learning, Cloud and High-Performance Computing, offer new opportunities for LUC mapping at finer spatial, temporal and thematic resolutions (Price et al. 2023; Camara 2020; Giuliani, 2023).

The integration of these new techniques significantly reduces the challenges associated with high-resolution LUC mapping (Owers et al. 2021; Planque et al. 2020). This enables dense time-series analysis, going beyond the simple diachronic comparison of a set of images, and significantly improves LUC change monitoring capabilities by identifying spatio-temporal patterns of LUC types and integrating distinct LUC maps (Simoes et al. 2021). The use of these technological advances can support the development of innovative methods for extracting information and improving the accuracy of LUC classification (Chaves et al., 2020). An alternative approach is to adopt a temporal approach first, followed by spatial considerations. This method associates each spatial location with time series data as input, using supervised or unsupervised classification methods to label individual pixels. The results are then combined, and spatial post-processing is applied to capture environmental information, generating more accurate maps better suited to continuous change monitoring (Simoes et al. 2021).

This study aims to present the first results of applying this temporal approach. It first uses 4 "main areas", then 10 "classes" and finally 46 "basic categories", testing machine/deep learning (ML/DL) methods to produce a land use map based on time series extracted from Sentinel-2 data in a Data Cube and *Arealstatistik* samples in the Lake Leman region (Geneva, Switzerland)

This initiative could help complement official national statistics by generating accurate annual information on land use changes at medium and high resolution on a national scale.

2. Study area

The study area covers the western part of Switzerland. It stretches between latitudes 46.0° and 46.8° North and longitudes 5.7° and 7.1° East. This region corresponds to Sentinel-2 tile 31TGM covering an area of 100 x 100 km² (CRS: EPSG:4326 – WGS 84 / UTM zone 31N). The research focuses on four out of six biogeographical regions defined in Switzerland, namely the Central Plateau, the Jura, the North Alps, and the Western Alps. The classification is based on the system established by Gonseth et al. 2001 (Giuliani, 2023)

The different regions offer a variety of landscape and ecological features, characterized by different climatic, geological conditions, as well as diverse flora and fauna.

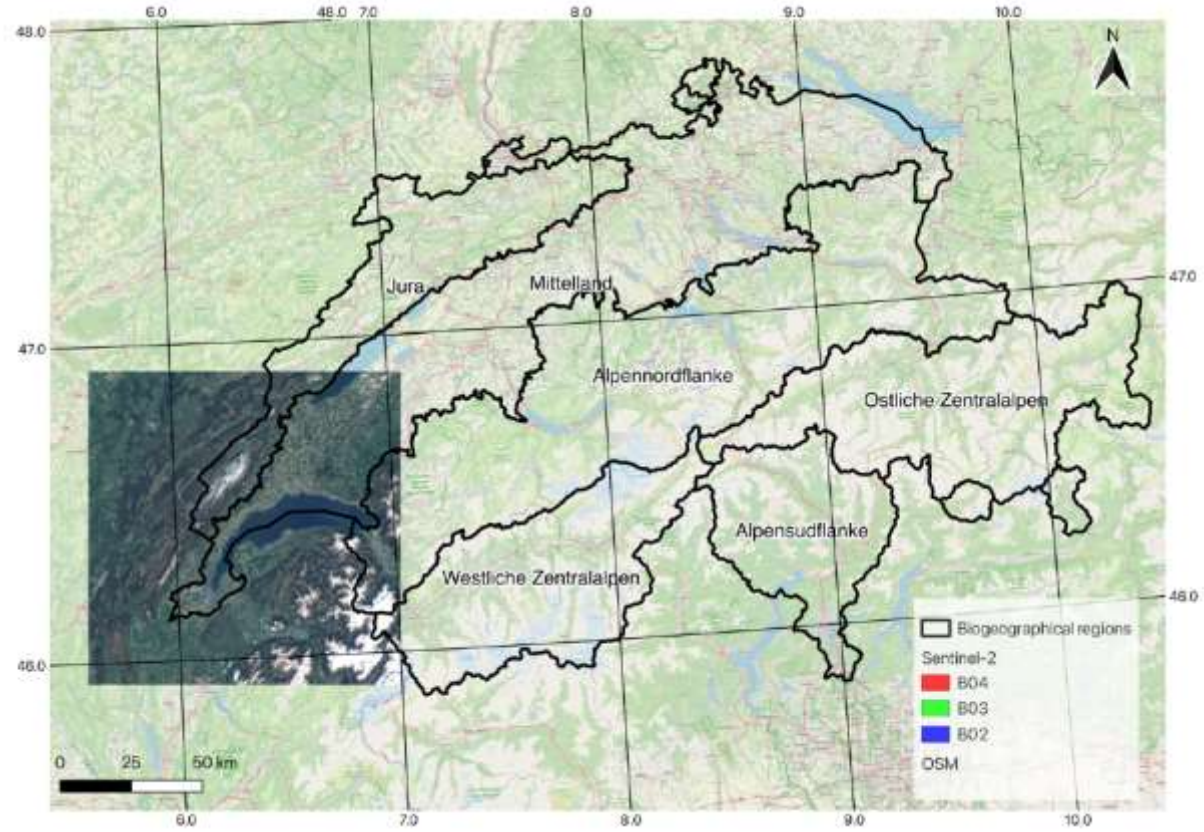

Figure 1: The study area's location (31 TGM) shown using a Sentinel-2 RGB composite image (B04, B03, B02) and biogeographic zones. Source: Giuliani, 2023.

The regions can be clustered into three main zones depending on altitude. The Alps (North, Central, South are concentrated at high altitudes and account for 60% of the country's total surface area. The Plateau and the Jura are situated at lower altitudes and occupy respectively 30% and 10% of the country's total surface area respectively (OFEV, 2022; Thomas & Giuliani, 2023).

Switzerland's climate is significantly shaped by the Alps and its proximity to the Atlantic Ocean. The plateau is typified by a moderately continental climate, the mountainous regions by an alpine climate, and the Southern Alps by a more temperate climate. Switzerland enjoys four

distinct seasons, with temperatures and precipitation fluctuating in accordance with the prevailing climatic conditions (NCCS, 2018; Giuliani, 2023).

3. Materials and methodology

The "Time-first, space-later" is a satellite image classification concept that considers time series analysis as the first step in remote sensing data analysis, with spatial information being considered after time series classification (SITS). The time-first part component enhances understanding of landscape evolution. It becomes possible to detect seasonal and long-term trends, as well as to identify abnormal events in the dataset such as forest fires, floods, or droughts. By processing each pixel of a data cube as a time series, pixel-based time series classification generates labelled pixels. The result is used as input for the spatial phase, where a smoothing algorithm improves the results of the temporal classification by considering the spatial neighbourhood of each pixel. The resulting map therefore incorporates both spatial and temporal information (SITS).

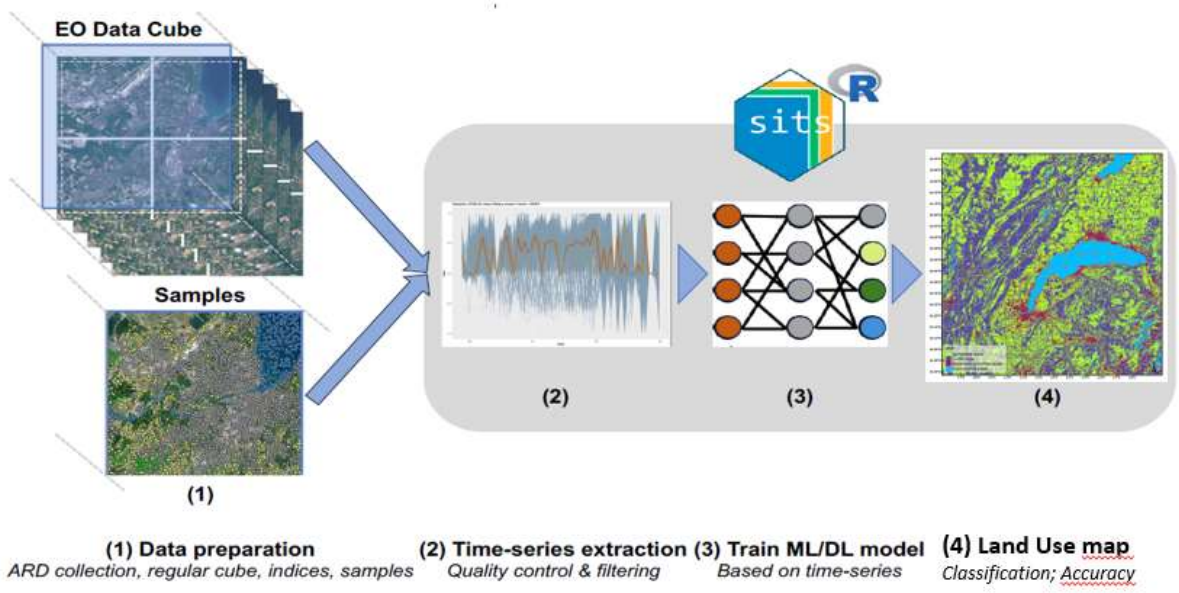

Figure 2: General workflow for the land use map production using with the SITS package in R. (source: Simoes et al., 2021)

Thus, the time-first approach employed for land use mapping follows the workflow outlined below (Figure 2). Large-scale Earth observation data are structured into data cubes, associating spatial location with time series. A machine/ learning (ML/DL) algorithm is trained using samples with known labels and it classifies the unlabelled time-series in the data cube (Camara et al. n.d.; Simoes et al. 2021). In this study, a time series of Sentinel-2 data covering the entire year 2018, as well as samples of official Land Use statistics (*Arealstatistik*) in the Greater Geneva region (*Switzerland*) were used. The entire process was implemented using R (version 4.3.1) in Rstudio (version 2023-06-16 ucrt) and the SITS package (version 1.4.2-1) (Giuliani, 2023).

3.1. SITS - Satellite Image Time Series Analysis on EO Data Cubes

SITS, an open-source R package(https://cran.r-project.org/web/packages/sits/index.html), is designed for land use and land cover (LUC) classification using Big Earth Observation data accessible via data cubes (e.g. Brazil, Switzerland, Digital Earth Africa) (Ferreira et al. 2020; Giuliani et al. 2017; Giuliani, 2023) or large image collections (e.g. Amazon Web Services, Microsoft Planetary Computer) (Giuliani, 2023). The SITS package uses remote sensing images time series for land classification, based on a time-first, space-later approach (SITS). For data preparation, collections of big Earth Observation images are structured into data cubes, associating each spatial location with a time-series. Each spatial location of a data cube is associated with a time series. A self-learning algorithm is trained on labeled locations, facilitating the classification of time series with a data cube. SITS provides a complete solution with an application programming interface (API) supporting the various stages of the LUC classification workflow, including sampling selection, time series clustering, machine learning model training and validation, classification and map post-processing (https://e-sensing.github.io/sitsbook/). Users typically follow a four-step workflow aligned with the dedicated functions of the SITS API (Table 1) (Giuliani, 2023; SITS).

1. Step 1: Data preparation
   - Select an analysis-ready data image collection `[sits_cube ()]`.
   - Build a regular data cube using the selected image collection `[sits_regularize ()]`.
   - Compute new bands and indices `[sits_apply ()]`.
2. Step 2: Time-series extraction
   - Extract time-series from samples and data cube that will be used as training data `[sits_get_data ()]`.
   - Perform quality control, and filtering of noisy samples in the time series. `[sits_reduce_imbalance (); sits_som_map (); sits_som_map ()]`.
3. Step 3: ML/DL model training
   - Train a ML/DL model using the time series samples `[sits_train ()]`.
4. Step 4: Land cover map production
   - Classify the data cube using the model to get class probabilities for each pixel `[sits_classify ()]`.
   - Post-process the probability cube to remove outliers `[sits_smooth ()]`.
   - Produce a labeled map from the post-processed probability cube `[sits_label_classify ()]`.
   - Evaluate the accuracy of the classification using best practices `[sits_accuracy ()]`.

Table 1: SITS API main functions, their respective inputs, and outputs for Land Use classification (Source: SITS Book).

Step	API_function	Inputs	Output
1	`sits_cube()`	ARD image collection	Irregular data cube
	`sits_regularize()`	Irregular data cube	Regular data cube
	`sits_apply()`	Regular data cube	Regular data cube with new bands and indices
2	`get_ts_by_chunk()`	Data cube and sample locations	Time series samples
	`sits_reduce_imabalance()`	Time series samples	Balanced time-series
	`sits_som_map()`	Balanced time-series	Clean balanced time-series
	`sits_kfold_validate()`	Balanced time-series	Confusion matrix + F1 scores
3	`sits_train()`	Time series and ML method	ML classification model
4	`sits_classify()`	ML classification model and regular data cube	Probability cube
	`sits_smooth()`	Probability cube	Post-processed probability cube
	`sits_uncertainty()`	Post-processed probability cube	Uncertainty cube
	`sits_label_classification()`	Post-processed probability cube	Classified map
	`sits_accuracy()`	Classified map and validation samples	Accuracy assessment

Figure 3 illustrates the SITS API's major functions and their respective interactions and connections with the various phases, from data preparation to the production of the final LU map (Giuliani, 2023).

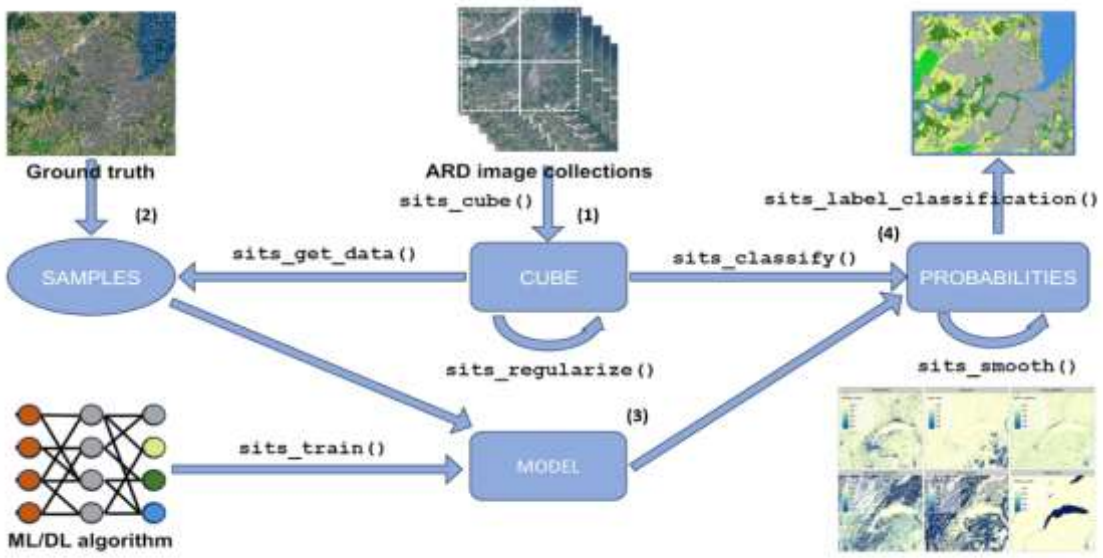

Figure 3: Main Function of the SITS API and their respective interactions in the processing chain. Source: SITS book.

Following the general approach using SITS, the workflow implemented uses Amazon Web Services to access analysis-ready satellite data (ARD) and *Arealstatistik* for reference data (Figure 4). The classification system NOLU04 was collected from OFS (https://www.bfs.admin.ch/bfs/fr/home/statistiques/espace-environnement/nomenclatures/arealstatistik/nolu2004.html) for the 4 main domains. Finally, a Machine Learning ML algorithm (Randon Forest) and two Deep Learning DL algorithms (Light TAE, TempsCNN) have been tested. The main steps in the workflow are detailed in the following subsections.

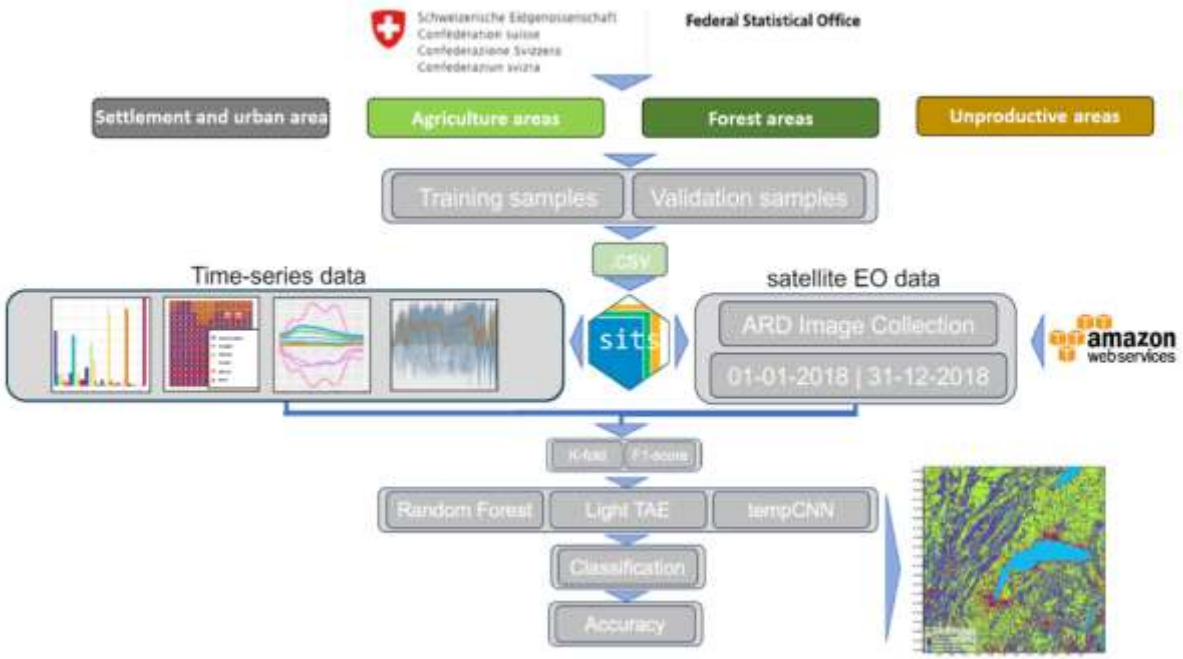

Figure 4: General implementation of the SITS workflow. Satellite Analysis Ready Data are provided by AWS, and samples are gathered from the *Arealstatistik* dataset supplied by the OFS.

3.2. STEP 1: Data preparation

The first step is to gather (1) reference data and (2) time series of multispectral data provided by a data cube. These will be used to extract the time series and classes for which the label (i.e. the classes) is known.

3.2.1. Satellite data

Satellite data is sourced from a wide range of cloud-based services that deliver analysis-ready data. The SITS package works with ARD image collections available in different cloud services such as AWS, Swiss Data Cube and Digital Earth Africa. According to the Committee on Earth Observation Satellites (CEOS), Analysis Ready Data (ARD) has been treated in accordance with a minimum set of requirements and organised in a form suitable for immediate analysis (Dwyer et al. 2018; Lewis et al. 2018; Giuliani, 2023).

First, the function `sits_list_collections ()` is used to identify available collections. In this work, satellite imagery is provided by Amazon Web Services (AWS) cloud. Data was downloaded using the `sits_cube ()` function and the Spatio Temporal Asset Catalog (STAC) interface. STAC is a geospatial information specification adopted by providers of large image collections (Hanson, M., 2021). The use of STAC brings significant advantages to sits, as the software can access up-to-date information via STAC endpoints (Giuliani, 2023). With SITS, the user defines a data cube by selecting a collection of ARD images and determining a spatio-temporal

extent (Simoes et al., 2021). Thus, data acquired via AWS are matched with *Arealstatistik* data. Accurate ML/DL classification requires consistent input data in terms of space, time, and bands. It is imperative to have no gaps or missing values, ensuring that the dimensionality of the training data matches that of the data to be classified (Appel and Pebesma 2019). The `sits_regularize ()` function ensures that all the cells in the data cube have the same spatio-temporal extent, resolution, and time interval, and that each cell contains a valid set of measurements. Finally, to complete the spectral bands of the regular cube and improve the separability of the LUC classes, it is recommended to add different spectral indices, which is a well-established practice in remote sensing (Chaves et al. 2023). This will enable discrimination between vegetation, water, built-up and bare structures. At the end of this process, a regular data cube with the 12 spectral bands.

3.2.2. Reference data

The reference data used in this study were collected via the Federal Statistical Office (FSO). The FSO provides and maintains freely the official LUC data (https://www.bfs.admin.ch/bfs/en/home/services/geostat/swiss-federal-statistics-geodata/land-use-cover-suitability/swiss-land-use-statistics.assetdetail.20104753.html). The *Arealstatistik* dataset is generated via visual interpretation of aerial photographs, with land cover (LC) and land use (LU) categories assigned to 4 million sample points uniformly distributed across the country on a 100m grid.

In this study, we used samples from the most recent survey period (2013-2018) as reference data (FSO, 2024). A subset of the reference data covering the study area consists of approximately 410,000 features that are separated in the training and validation data using the commonly accepted 70/30% separation threshold (Shetty et al. 2021) corresponding approximately to 287,000 samples for training and 123,000 for validation.

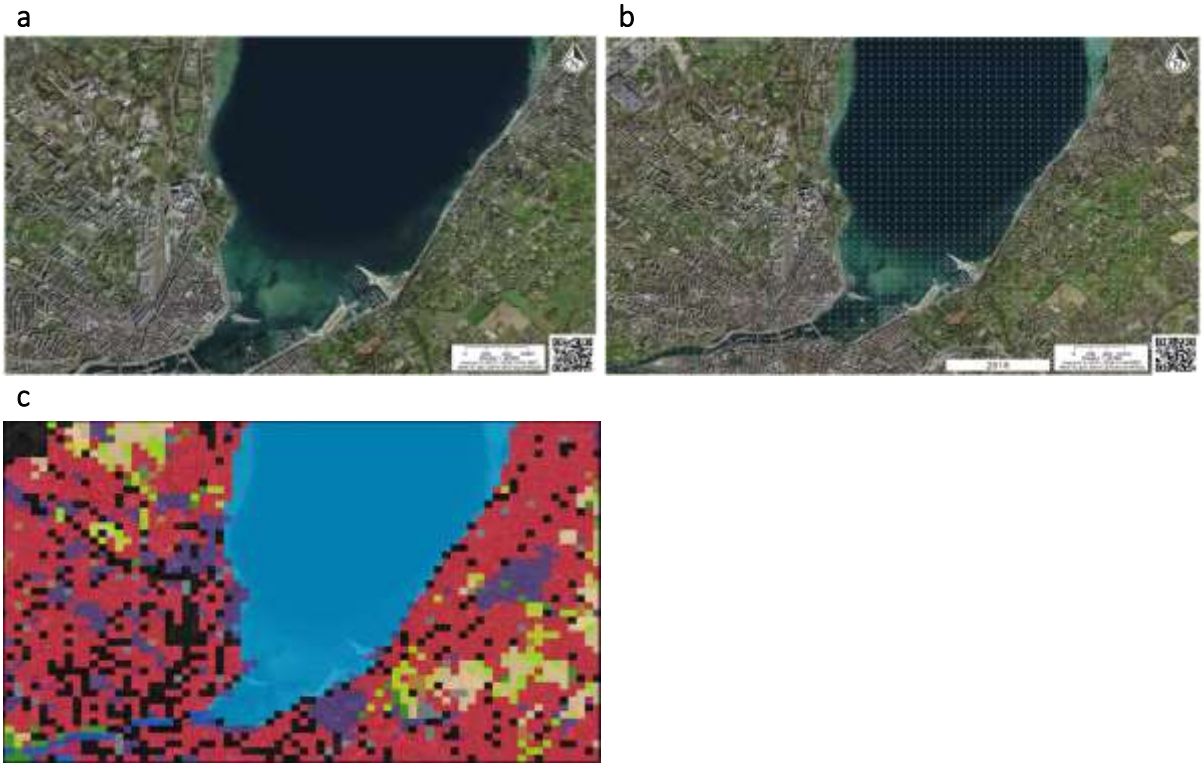

Figure 5: (a) Aerial view over an area in Geneva. (b) Aerial view of the Geneva area with *Arealstatistik* sampling points, showing the 4 main domains of NOLU04. (c) Translation in gridded land use map.

For the LU classification, the NOLU04 classification scheme was applied. It considers 46 basic categories", but for better statistical reliability, particularly for small-scale use, these categories are grouped into 10 “classes” and 4 "main areas" (table 2). As a result, it was decided to use the main areas to test the "time-first" approach.

Table 2: Categories according to the NOLU04 land use classification, separated into main areas and basic categories. (Source: OFS)

Principal Domains	Classes	Basic Categories
1-Settlement and urban area	100 – Building areas	101 – Industrial and commercial areas > 1 ha 102 – Industrial and commercial areas < 1 ha 103 – Single and two-family house site 104 – Aligned and terraced house area 105 – Apartment blocks 106 – Public building area 107 – Agricultural building area 108 – Unspecified building area
	120 – Transport surfaces	121 – Highway areas 122 – Road areas 123 – Parking areas 124 – Railway areas 125 – Airfields
	140 – Special urban areas	141 – Energy supply facilities 142 – Wastewater treatment plants 143 – Other supply and disposal facilities 144 – Landfill sites 145 – Material extraction 146 – Construction sites 147 – Brownfield sites and disused buildings
	160 – Recreational areas and cemeteries	161 – Public parks 162 – Sports facilities 163 – Golf courses 164 – Camping sites 165 – Family gardens 166 – Cemeteries
2-Agriculture area	200 – Orchards, vineyards, horti-culture	201 – Fruit growing 202 – Viticulture 203 – Horticulture
	220 – Arable and Grassland	221 – Arable land 222 – Natural grassland 223 – Local pasture
	240 – Alpine grazing areas	241 – Alpine meadows 242 – Alpine and Jura pasture 243 – Sheep pastures
3-Wooded area	300 – Forest (excluding farm-land)	301 – Forest stands 302 – Afforested areas 303 – Wood cutting 304 – Forest damage
4-Unproductive area	400 – Lakes and rivers	401 – Lakes 402 – Rivers, streams 403 – Flood barriers
	420 – Unproductive land	421 – No use 422 – Protective structures 423 – Alpine sports infrastructure 424 – Landscape interventions

3.3. STEP 2: Time-series extraction

The second step in the workflow aims to extract time series for the training samples from the regular data cube using the `get_ts_by_chunk ()` function. The result is a time series of the 12 spectral bands for each reference data point. This requires handling large time series and hence adequately capturing the temporal variability of each class. Patterns are extracted with SITS using `sits_patterns ()` the time-weighted dynamic approach for satellite image time series (dtwSat) (Maus et al. 2019). Using Generalized Additive Model (GAM), single time series can be approximated and analysed, providing insight into time-series behaviour and the separability of each class (Giuliani, 2023).

It is crucial to select quality training samples to obtain accurate results based on these time series (Bratic et., 2023; Hermosilla et al. 2022; Giuliani, 2023). The quantity and quality of these samples have a considerable influence on the accuracy of the results (Maxwell et al., 2018). Extensive and accurate training datasets are preferably required, regardless of the algorithm used. Noisy training samples can significantly reduce classification performance (Frenay and Verleysen, 2014). Therefore, before training an ML/DL model, it is essential to apply pre-processing methods to improve sample quality by eliminating incorrect labels and those with low discriminatory power (Giuliani, 2023).

Quality control and filtering of noisy samples is carried out via three distinct processes: (1) cross-validation of training samples, (2) quality control of samples and (3) reduction of sample imbalance.

1. Cross- validation is applied to estimate the prediction error inherent in a model using the `sits_kfold_validate ()` function.
2. Quality control is based on the SOM (Self-Organizing Maps) clustering technique using the `sits_som_map ()` function. This approach has proven effective in filtering noisy samples and evaluating appropriate spectral bands and indices to improve the separability of LUC classes.
3. Imbalance reduction aims to decrease the training sample imbalance common in the distribution of Earth observation-based samples associated with each label. The `sits_reduce_imbalance ()` function applies the Synthetic Minority Over-sampling TEchnique (SMOTE) method to balance the number of samples from the least frequent and most frequent labels.

The end of the process guarantees the quality of the training samples, which improves class discrimination and enables efficient training of the ML/DL models.

3.4. STEP 3: Train ML/DL model

SITS implements various machine learning methods that can be assimilated to supervised classification. Classification is applied to train an algorithm to predict the class to which the input data belongs (Giuliani, 2023). It classifies individual time series by applying the “time-first” approach mentioned in the introduction.

Therefore, two methods are applied:

1. Machine learning techniques that fail to explicitly incorporate the temporal structure of a given time series exemplified by Random Forest (RF) or Support Vector Machine (SVM)
2. Deep learning (DL) methodologies that deal with temporal relationships between observed values in a time series, illustrated by the Temporal Convolution Neural Network (tempCNN) or the Lightweight Temporal Self-Attention Encoder (LTAE) (Pelletier et al., 2019; Cheng et al. 2023; Fawaz et al. 2018; Sainte Fare Garnot et al. 2020; Giuliani, 2023)

This study seeks to compare the prevalent method (i.e. Random Forest) with two emerging methods (i.e. tempCNN and LTAE) which are receiving increasing interest in the community (Papoutsis et al., 2023). These approaches can be deployed via the `sits_train ()` function, which supplies a standardised interface for training different ML/DL algorithms. SITS offers a sensible set of defaults to help novice users train their models, while allowing expert users to adjust model parameters using the function `sits_tuning ()` (Giuliani, 2023).

3.5. STEP 4: Land Use map production

The final step in the workflow consists of applying the trained model to classify the data cube with the time series using the `sits_classify ()` function. The process output is a data cube with distinctive probability layers for each class. These layers indicate the probability that a given pixel belongs to a specific class (Figure 6).

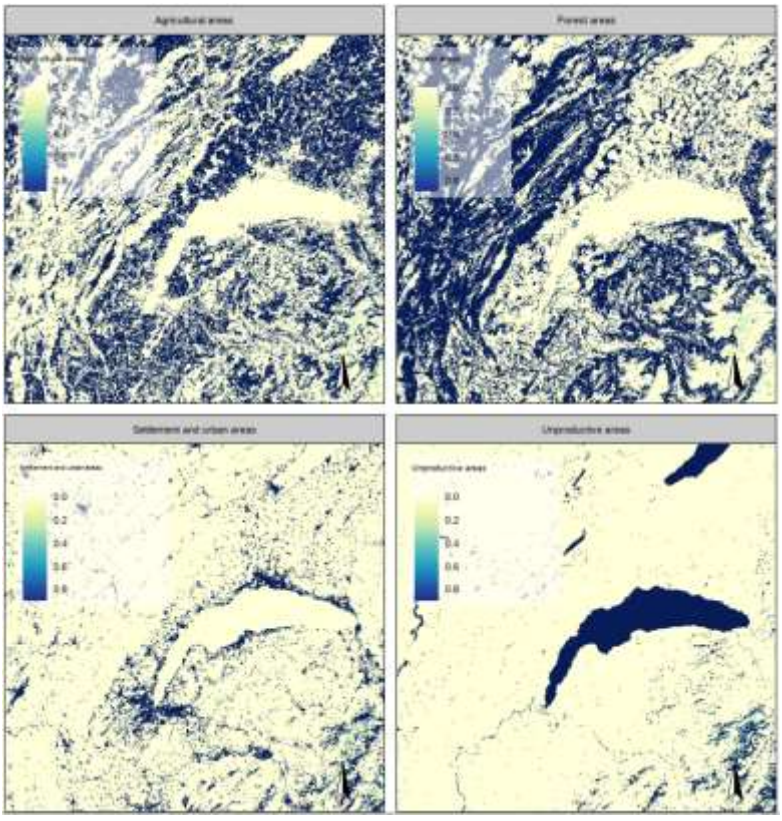

Figure 6: Smoothed probabilities obtained for the tempCNN model, and each land use clasess according to the NOLU04 nomenclature.

Firstly, to reduce outliers, `sits_smooth ()` function is used to apply post-processing smoothing techniques, considering the spatial neighborhood. This improves the quality of the classification and the interpretability of the resulting map by reducing 'salt and pepper' effects and

fuzzy edges. Then, the function `sits_label_classification ()` is performed on the smoothed probabilities to assign a label to each pixel based on the highest probability. Finally, to assess accuracy and uncertainty, the function `sits_accuracy ()` is implemented using an area-weighted approach and the function `sits_uncertainty ()` is run to estimate the entropy of the land use map, accounting various potential sources of uncertainty, such as classification errors, classification scheme ambiguity, landscape variability and data limitations.

4. Results

The main objective of this work is to test a temporal approach with different ML/DL algorithms to classify Sentinel-2 data according to the NOLU04 national nomenclature. First, a classification is made according to the 4 principal domains, then the 10 classes and finally the 46 categories.

4.1. STEP 1: Data preparation

Satellite data were collected via the Amazon Web Service cloud in the g_cube using the API function `sits_cube ()`. The data acquired includes Sentinel-2 images from 2018, encompassing 12 bands, totaling 112 scenes on tile 31 TGM, covering the specified area of interest. The API function `sits_timeline ()` displays the original cube timelines before regularization for time series data analysis (Figure 7a). The `sits_regularize ()` function then constructs a data cube with a regular timeline and a best estimate of a valid pixel for each interval (Figure 7b). The g_reg_cube contains Sentinel-2 images from the 2018 period encompassing 12 bands and totaling 57 scenes on tile 31 TGM.

(a)

```
> sits_timeline(g_cube)
  [1] "2018-03-21" "2018-03-26" "2018-03-29" "2018-03-31" "2018-04-03" "2018-04-05" "2018-04-08" "2018-04-10" "2018-04-13" "2018-04-15" "2018-04-18"
 [12] "2018-04-20" "2018-04-23" "2018-04-25" "2018-04-28" "2018-04-30" "2018-05-03" "2018-05-05" "2018-05-08" "2018-05-10" "2018-05-13" "2018-05-15"
 [23] "2018-05-18" "2018-05-20" "2018-05-23" "2018-05-25" "2018-05-28" "2018-05-30" "2018-06-02" "2018-06-04" "2018-06-07" "2018-06-09" "2018-06-12"
 [34] "2018-06-14" "2018-06-17" "2018-06-19" "2018-06-22" "2018-06-24" "2018-06-27" "2018-06-29" "2018-07-02" "2018-07-04" "2018-07-07" "2018-07-09"
 [45] "2018-07-12" "2018-07-14" "2018-07-17" "2018-07-19" "2018-07-22" "2018-07-24" "2018-07-27" "2018-07-29" "2018-08-01" "2018-08-03" "2018-08-06"
 [56] "2018-08-08" "2018-08-11" "2018-08-13" "2018-08-16" "2018-08-18" "2018-08-21" "2018-08-23" "2018-08-26" "2018-08-28" "2018-08-31" "2018-09-02"
 [67] "2018-09-05" "2018-09-07" "2018-09-10" "2018-09-12" "2018-09-17" "2018-09-20" "2018-09-22" "2018-09-25" "2018-09-27" "2018-09-30" "2018-10-02"
 [78] "2018-10-05" "2018-10-07" "2018-10-10" "2018-10-12" "2018-10-15" "2018-10-17" "2018-10-20" "2018-10-22" "2018-10-25" "2018-10-27" "2018-10-30"
 [89] "2018-11-01" "2018-11-04" "2018-11-06" "2018-11-09" "2018-11-11" "2018-11-14" "2018-11-16" "2018-11-19" "2018-11-21" "2018-11-24" "2018-11-26"
[100] "2018-11-29" "2018-12-01" "2018-12-04" "2018-12-06" "2018-12-09" "2018-12-11" "2018-12-14" "2018-12-16" "2018-12-19" "2018-12-21" "2018-12-24"
[111] "2018-12-26" "2018-12-29"
```

(b)

```
> sits_timeline(g_reg_cube)
 [1] "2018-03-21" "2018-03-26" "2018-03-31" "2018-04-05" "2018-04-10" "2018-04-15" "2018-04-20" "2018-04-25" "2018-04-30" "2018-05-05" "2018-05-10"
[12] "2018-05-15" "2018-05-20" "2018-05-25" "2018-05-30" "2018-06-04" "2018-06-09" "2018-06-14" "2018-06-19" "2018-06-24" "2018-06-29" "2018-07-04"
[23] "2018-07-09" "2018-07-14" "2018-07-19" "2018-07-24" "2018-07-29" "2018-08-03" "2018-08-08" "2018-08-13" "2018-08-18" "2018-08-23" "2018-08-28"
[34] "2018-09-02" "2018-09-07" "2018-09-12" "2018-09-17" "2018-09-22" "2018-09-27" "2018-10-02" "2018-10-07" "2018-10-12" "2018-10-17" "2018-10-22"
[45] "2018-10-27" "2018-11-01" "2018-11-06" "2018-11-11" "2018-11-16" "2018-11-21" "2018-11-26" "2018-12-01" "2018-12-06" "2018-12-11" "2018-12-16"
[56] "2018-12-21" "2018-12-26"
```

Figure 7: (a) A timeline of the original cubes before regularization for the period 2018, data collected from AWS. (b) A timeline of the original cubes after regularization for the period 2018, data collected from AWS.

4.2. Results for the 4 Principal Domains

4.2.1. STEP 2: Time-series extraction

After applying the API function `sits_patterns ()`, a graph grouping 4-time series is captured for the 4 Principal Domains (Agricultural areas, Forest areas, Settlement and urban areas, Unproductive areas). A time series of the 12 spectral bands is produced for each domain and for each reference data point. The distribution pattern of the main domain "Unproductive areas" differs significantly from the other main domains. However, the other three main domains, namely, "Agricultural areas", "Forest areas" and "Settlement and urban areas", show broadly similar distribution patterns (Figure 8). This is supported by the "before" histograms displayed in (Figure 9), which show noisy samples in all the three domains mentioned above.

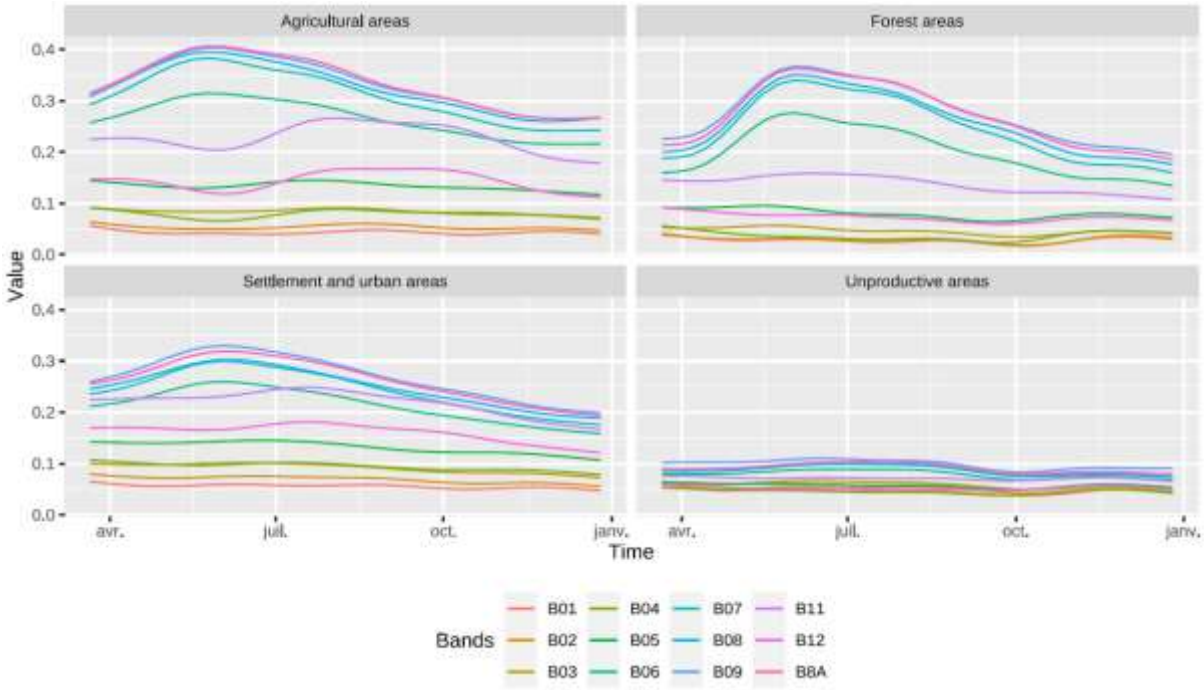

Figure 8: Time series of 12 modelled spectral bands for each reference data point for the 4 Principal Domains (Agricultural areas, Forest areas, Settlement and urban areas and Unproductive areas) obtained using `sits_patterns ()` in SITS.

The application of the SOM (Self-Organizing Maps) technique using the `sits_som_map ()` function therefore proves effective in reducing noisy samples. Comparison of the histograms "before" and "after" the pre-processing indicates a considerable improvement in the separability for the 3 main land use domains, i.e. "Agricultural areas", "Forest areas" and "Settlement and urban areas" (Figure 9). Table 3 summarizes the use of the function `sits_reduce_imbalance ()`, which balances the number of samples from the least frequent and most frequent labels. Figure 9 and Table 3 guarantee the quality of the training samples, enabling efficient training of ML/DL models.

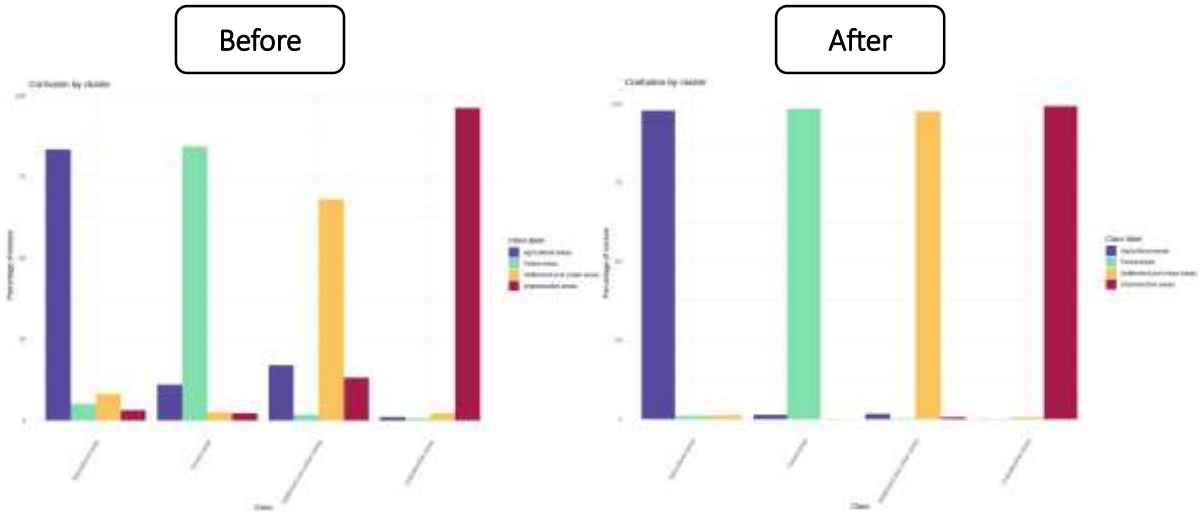

Figure 9: Confusion by cluster between the 4 Principal Domains before and after pre-processing of the training samples.

Table 3: Comparing sample proportion before and after pre-processing of the training samples for 4 main domains.

	Before pre-processing		After pre-processing	
Label	Count	Proportion	Count	Proportion
Agricultural areas	129806	0.455	1600	0.251
Forest areas	78472	0.275	1596	0.251
Settlement and urban areas	33600	0.118	1600	0.251
Unproductive areas	43629	0.153	1568	0.246

Cross-validation results on the balanced training dataset indicate similar performance between the two models (LTAE and tempCNN) and the model (RF) in terms of accuracy and F1 scores (Table 4). None of the algorithms clearly stands out as outperforming the others.

Table 4: Cross-validation for 4 Principal Domains highlighting Accuracy, 95% Confidence Intervals (CI), Kappa coefficients and F1 scores for the three tested methods: Random Forest (RF), Lightweight Temporal Self-Attention Encoder (LTAE) and Temporal Convolutional Neural Network (tempCNN).

	Label	RF	LTAE	tempCNN
Accuracy		0.9859	0.9782	0.9837
95% CI		(0.9826,0.9886)	(0.9743,0.9816)	(0.9802,0.9866)
Kappa		0.9811	0.9709	0.9782
F1 - scores	Agricultural areas	0.9185	-	-
	Forest areas	0.9324	-	-
	Settlement and urban areas	0.7825	-	-
	Unproductive areas	0.9136	-	-

N.B: Torch package issue prevented the calculation of LTAE F1-score and tempCNN F1-score.

4.2.2. STEP 3: Train ML/DL model

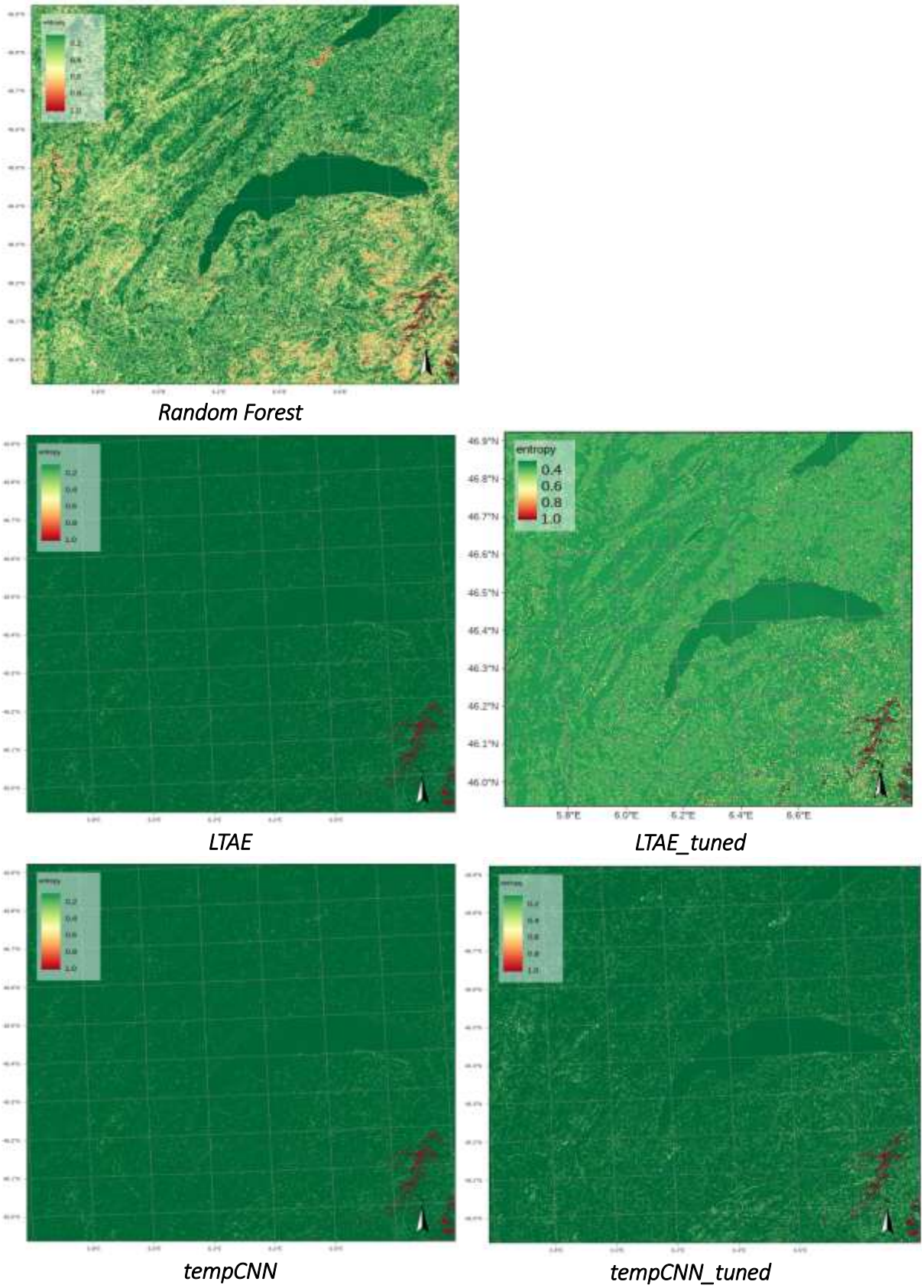

Figure 10: Uncertainty of RF, LTAE, tempCNN, LTAE_tuned and tempCNN_tuned models estimated through entropy estimation for the 4 principal domains.

An analysis of the uncertainty map shows that the models (LTAE and tempCNN) have significantly lower uncertainty than the classification based on the RF model. In addition, the tempCNN_tuned model has a lower uncertainty at classification level than the LTAE_tuned model. However, a degradation is observed for the LTAE_tuned model in comparison with the LTAE model (Figure 10).

Table 5 summarizes overall accuracy, user accuracy and producer accuracy for the three ML/DP models (RF, LTAE, tempCNN). Overall accuracy represents the proportion of samples correctly classified from the reference data. The producer's accuracy measures the performance of the classification in capturing reference pixels of the vegetation cover type, thus assessing errors of omission. The user's accuracy assesses the probability that a pixel assigned to a specific category accurately represent that category in the field, thus quantifying errors of commission.

Table 5: Overall, User (UA), Producer (PA) accuracy for three models (RF, LTAE, tempCNN).

	RF	LTAE	LTAE_tuned	tempCNN	tempCNN_tuned
Overall	0.86	0.86	0.86	0.86	0.87

	UA	PA	UA	PA	UA	PA	UA	PA	UA	PA
Agricultural areas	0.87	0.86	0.88	0.89	0.90	0.85	0.89	0.84	0.88	0.88
Forest areas	0.84	0.97	0.87	0.95	0.83	0.98	0.82	0.97	0.86	0.96
Settlement and urban areas	0.72	0.60	0.70	0.63	0.74	0.64	0.74	0.64	0.75	0.64
Unproductive areas	0.99	0.69	0.96	0.71	0.97	0.71	0.94	0.69	0.95	0.72

The results indicate that the Random Forest model gives acceptable accuracies both overall and by class. The best results are the models with adjusted hyper-parameters (LTAE_tuned, tempCNN_tuned). These models produce slightly higher accuracy values with less uncertainty in the model. Their results are comparable. They both appear to be the best-performing models to use for a time-first approach. Furthermore, when we combine the results (Table 5 and Figure 10), we find that the tempCNN/ tempCNN_tuned models are the most suitable.

4.2.3. STEP 4: Land Use map production

The final out map (Figure 11) was generated using a tuned tempCNN model. This model is recognized as the optimal choice for achieving highly accurate classification results. Visually, the landscape is largely dominated by “Forest areas”, “Agricultural areas” and “Unproductive areas”.

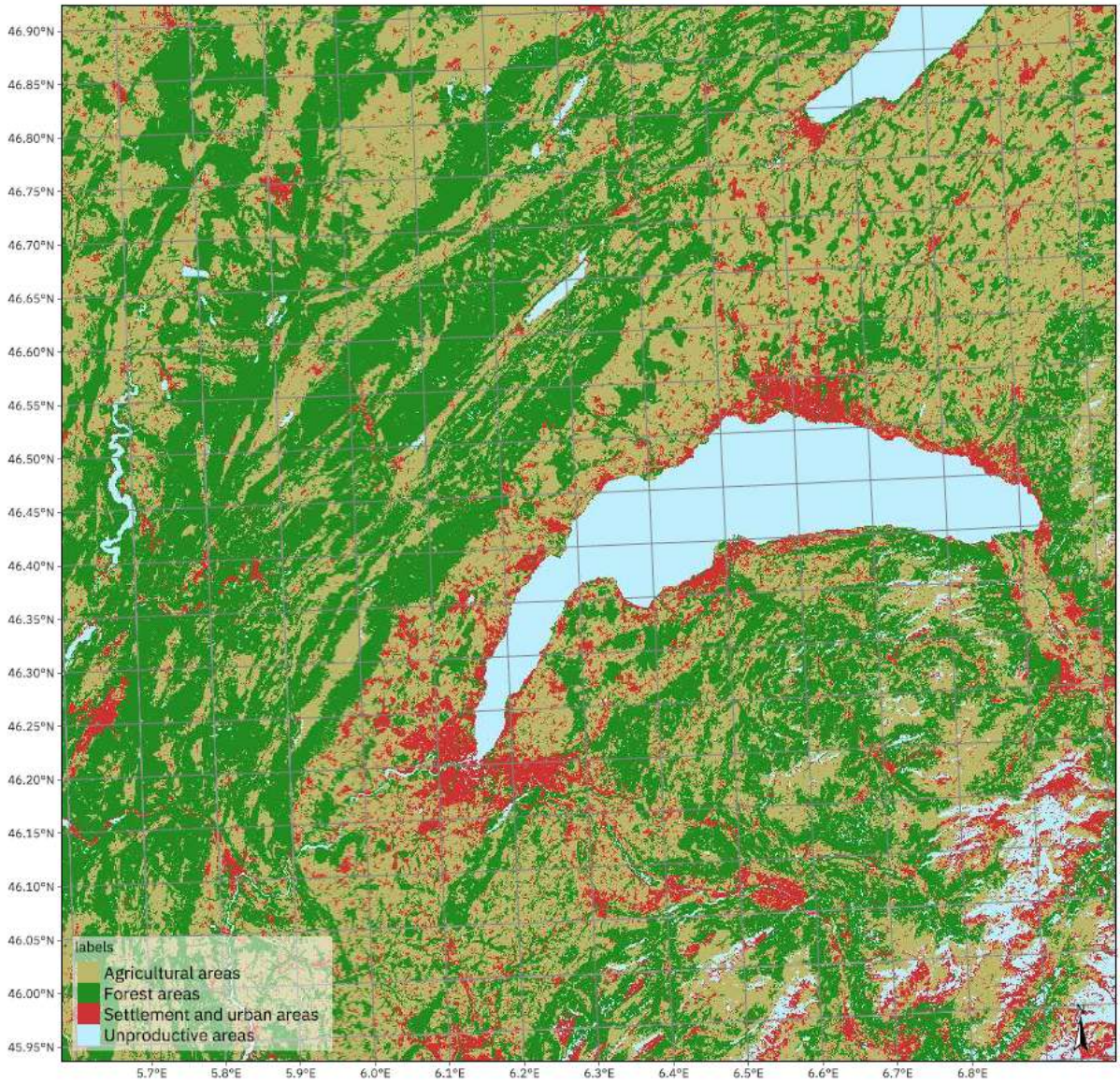

Figure 11: Final output of the classification obtained with the tempCNN_tuned model.

4.3. Results for 10 classes

4.3.1. STEP 2: Time-series extraction

The presented findings depict a graphical representation comprising 10-time series corresponding to distinct land-use classes; “Alpine grazing areas”, “Arable and Grassland”, “Building areas”, “Forest”, “Lakes and rivers”, “Orchards, vineyards, horticulture”, “Recreational areas and cemeteries”, “Special urban areas”, “Transport surfaces” and “Unproductive land” (Figure 13). A time series of the 12 spectral bands is produced for each class and for each reference data point. Particularly the "Lakes and rivers" class exhibits a distinct distribution from the others.

Whereas the “Building areas”, “Special urban areas”, “Recreational areas and cemeteries” and “Transport surfaces” classes show broadly similar distributions. The “Orchards, vineyards, horticulture” class displays similarities with the distributions of the “Arable and grassland” and “Alpine grazing areas” classes. These observations are confirmed by the results presented in figure 13.

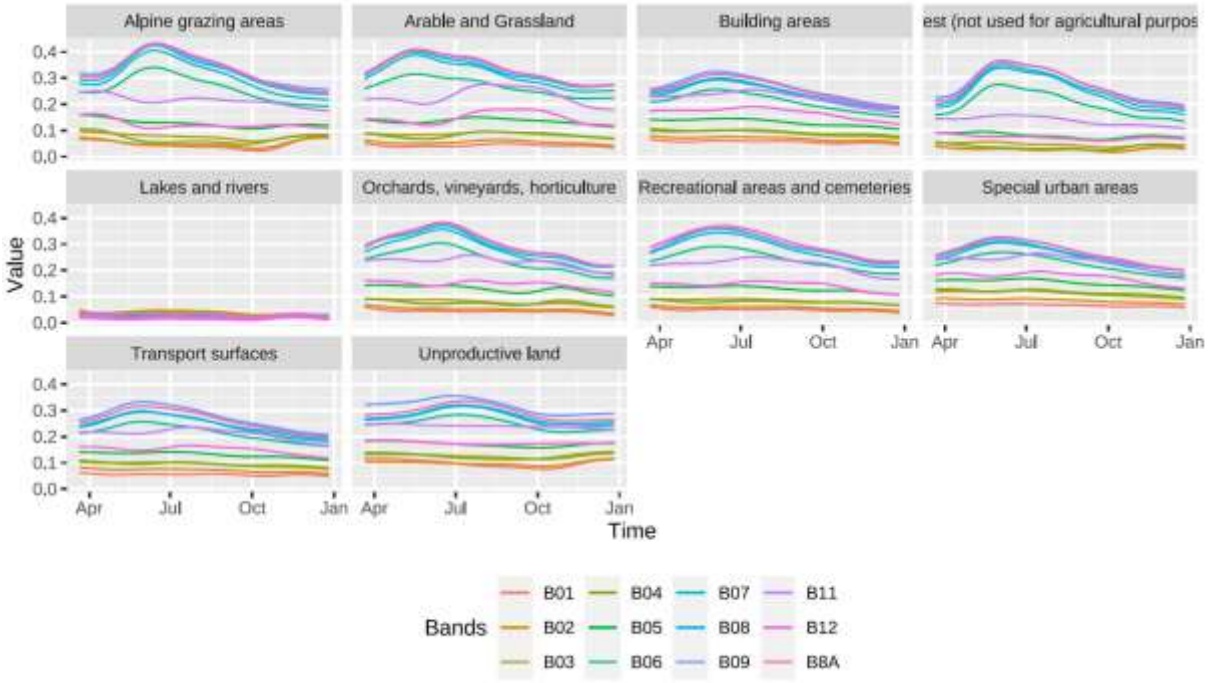

Figure 13: Time series of 12 modelled spectral bands for each reference data point for the 10 classes (Alpine grazing areas, Arable and Grassland, Building areas, Forest, Lakes and rivers, Orchards, vineyards, Horticulture, Recreational areas and cemeteries, Special urban areas, Transport surface and Unproductive land) obtained using `sits_patterns ()` in SITS.

The application of the self-organizing map (SOM) technique, using the `sits_som_map ()` function, effectively mitigated the presence of noisy samples. Examination of the histograms representing the distribution of "before" classes highlights five noisy classes, even though they each present a minimum of 50% of the overall distribution: "Alpine grazing areas", "Arable and Grassland", "Building areas", "Forests" and "Lakes and rivers" (Figure 14). Notably, for the “Arable and Grassland”, “Forest”, “Alpine grazing areas”, “Building areas”, discernible improvements in separability are observed. Thus, the refinement process reduced the number of classes from 10 to 6 main categories, i.e. "Alpine grazing areas", "Arable and Grassland", "Building areas", "Forests", "Lakes and rivers" and "Unproductive land" (Figure 14).

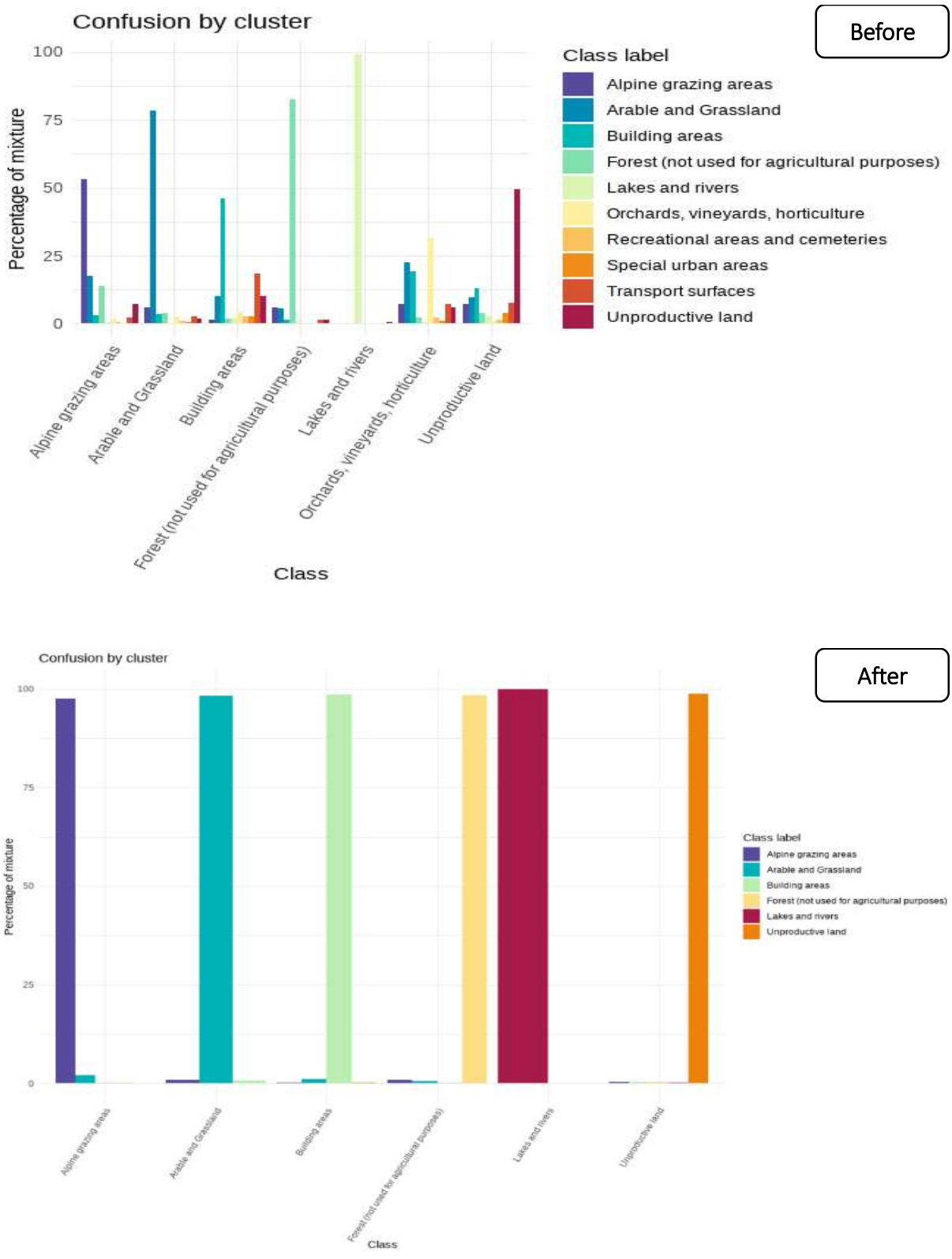

Figure 14: Confusion by cluster between the 10 Classes before and after pre-processing of the training samples.

The Table 7 provides a comprehensive summary of the application of the `sits_reduce_imbalance ()` function, which is used to equalize the sample distribution between the least frequent and most frequent labels. The combined results of (Figure 14) and (Table 7 confirm the effectiveness of the methodology applied to guarantee the high quality of training samples, facilitating the efficient training of machine learning (ML) and deep learning (DL) models.

Table 7: Comparing sample proportion before and after pre-processing of the training samples for 10 classes.

	Before pre-processing		After pre-processing	
Label	Count	Proportion	Count	Proportion
Alpine grazing areas	23108	0.0809	1592	0.169
Arable and Grassland	100602	0.352	1600	0.170
Building areas	19506	0.0683	1600	0.170
Forest (excluding farmland)	78472	0.275	1592	0.169
Lakes and rivers	33505	0.117	1552	0.165
Unproductive land	10124	0.0355	1464	0.156
Orchards, vineyards, horticulture	6096	0.0214	-	-
Recreational areas and cemetries	1606	0.0081	-	-
Special urban areas	10158	0.0056	-	-
Transport surfaces	10124	0.0356	-	-

The outcomes of cross-validation results on the balanced training dataset indicate comparable levels of performance and accuracy between the models (LTAE and tempCNN) and the RF model (Table 8). Notably, no algorithms stand out as outperforming the others.

Table 8: Cross-validation for 10 classes highlighting Accuracy, 95% Confidence Intervals (CI), Kappa coefficients for the three tested methods: Random Forest (RF), Lightweight Temporal Self-Attention Encoder (LTAE) and Temporal Convolutional Neural Network (tempCNN).

	RF	LTAE	TempCNN
Accuracy	0.991	0.9839	0.9937
95% CI	(0.9888,0.9928)	(0.9812,0.9864)	(0.9919,0.9952)
Kappa	0.9891	0.9807	0.9925

N.B: F1- scores were not calculated due to the computation process' slowness s, limited available RAM, and Torch package issues.

4.3.2. STEP 3: Train ML/DL model

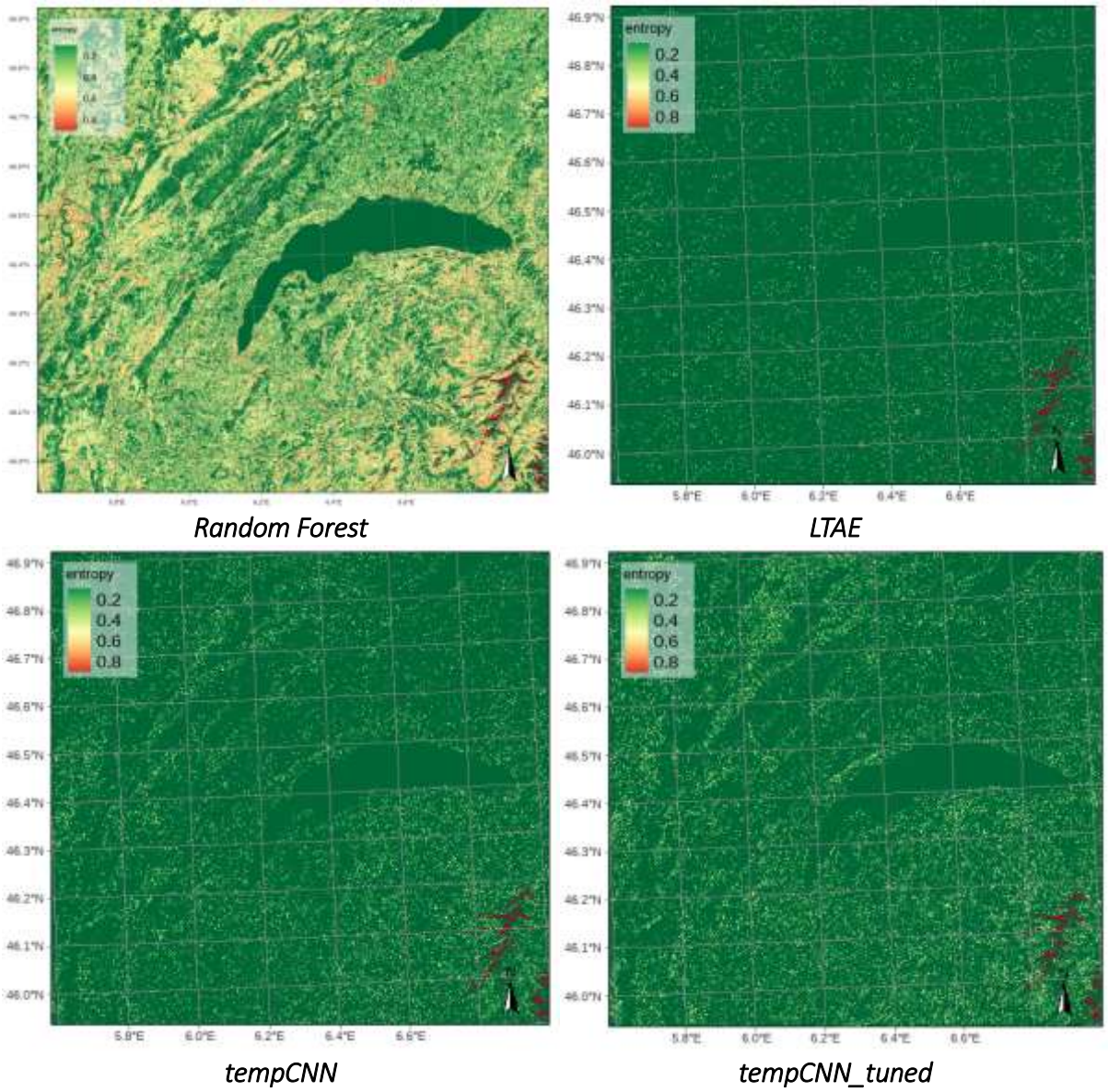

Figure 15: Uncertainty of RF, LTAE, tempCNN and tempCNN_tuned models estimated through entropy estimation for the 10 classes.

An analysis of the uncertainty map indicates a notable reduction in uncertainty between the models (LTAE and tempCNN) compared to the RF model. Furthermore, a noticeable drop in uncertainty quality is observed between the LTAE model and its optimized counterpart, LTAE_tuned model. In addition, a slight deterioration in uncertainty quality is observed between the models tempCNN and tempCNN_tuned (Figure 15).

Table 9 provides a comprehensive summary of overall accuracy, user accuracy and producer accuracy for the three ML/DP models (RF, LTAE, tempCNN). Global accuracy quantifies the proportion of samples correctly classified on the reference data. Producer accuracy evaluates classification performance by capturing reference pixels of the vegetation cover type, focusing on

omission errors. User accuracy assesses the probability that a pixel assigned to a specific category accurately represents that category in the field, by measuring errors of commission.

Table 9: Overall, User (UA), Producer (PA) accuracy for three models (RF, LTAE, tempCNN)

	RF		LTAE		LTAE_tuned		tempCNN		tempCNN_tuned	
Overall	0.84		0.82				0.86		0.85	
	UA	PA	UA	PA	UA	PA	UA	PA	UA	PA
Alpine grazing areas	0.62	0.73	0.52	0.83	N/A	N/A	0.68	0.71	0.68	0.71
Arable and Grassland	0.93	0.82	0.94	0.76	N/A	N/A	0.92	0.87	0.94	0.83
Building areas	0.74	0.73	0.72	0.76	N/A	N/A	0.76	0.77	0.77	0.73
Forest	0.86	0.96	0.90	0.93	N/A	N/A	0.88	0.96	0.86	0.97
Lakes and rivers	1.00	0.88	0.99	0.88	N/A	N/A	0.99	0.89	1.00	0.88
Unproductive land	0.71	0.42	0.69	0.38	N/A	N/A	0.69	0.47	0.65	0.51

The results indicate that the Random Forest model presents acceptable accuracies both in terms of overall performance and per class. However, the tempCNN and tempCNN_tuned models unequivocally achieve superior results. These models show slightly higher accuracy values with reduced model uncertainty. Their results are comparable, and they both emerge as the most effective models for a time-first approach. Furthermore, a synthesis of the results in Table 9 and Figure 15 reveals that the adjusted tempCNN model emerges as the most appropriate choice.

The results for LTAE_tuned could not be collected due the computation process' slowness and insufficient available RAM. The computation time for the 4 principal domains took around 3 days.

4.3.3. STEP 4: Land Use map production

The final out map (Figure 16) was generated using a tuned tempCNN model. This model is recognized as the optimal choice for achieving highly accurate classification results. Visually, the landscape is largely dominated by "Forest", "Arable and Grassland" and "Lakes and rivers".

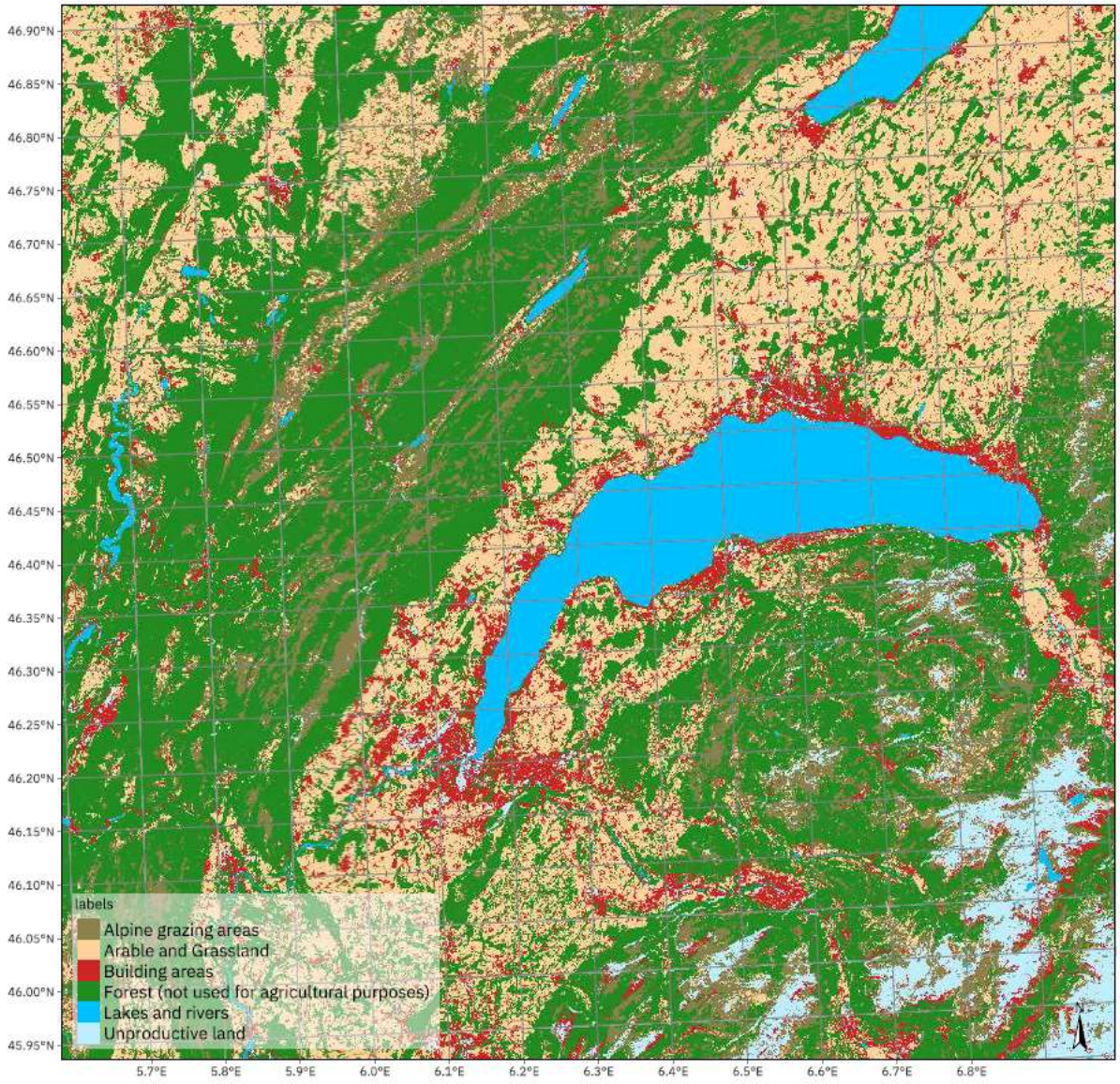

Figure 16: Final output of the classification obtained with the tempCNN_tuned model for 10 classes.

4.4. Results for the 46 basic categories

4.4.1. STEP 2 : Time-series extraction

The results presented show a graphical representation comprising 46-time series corresponding to distinct land use classes: "Afforestation", "Airports and Airfields", "Agricultural building areas", "Alpine meadows", "Alpine pastures", "Alpine sports facilities", "Arable land", "Camping areas", “Cemeteries", "Construction sites", "Damaged Forest", etc. A time series of the 12 spectral bands is produced for each class and for each reference data point. The "Lakes and rivers" class shows a distinct distribution from the others. The "Unused", "Unexploited urban areas", "Roads" and "Public parks", “Camping areas” classes show broadly similar distributions. Similar distribution patterns are also observed in classes such as "Forests", "Golf courses", "Afforestation" and several others. These observations are confirmed by the results presented in figure 17.

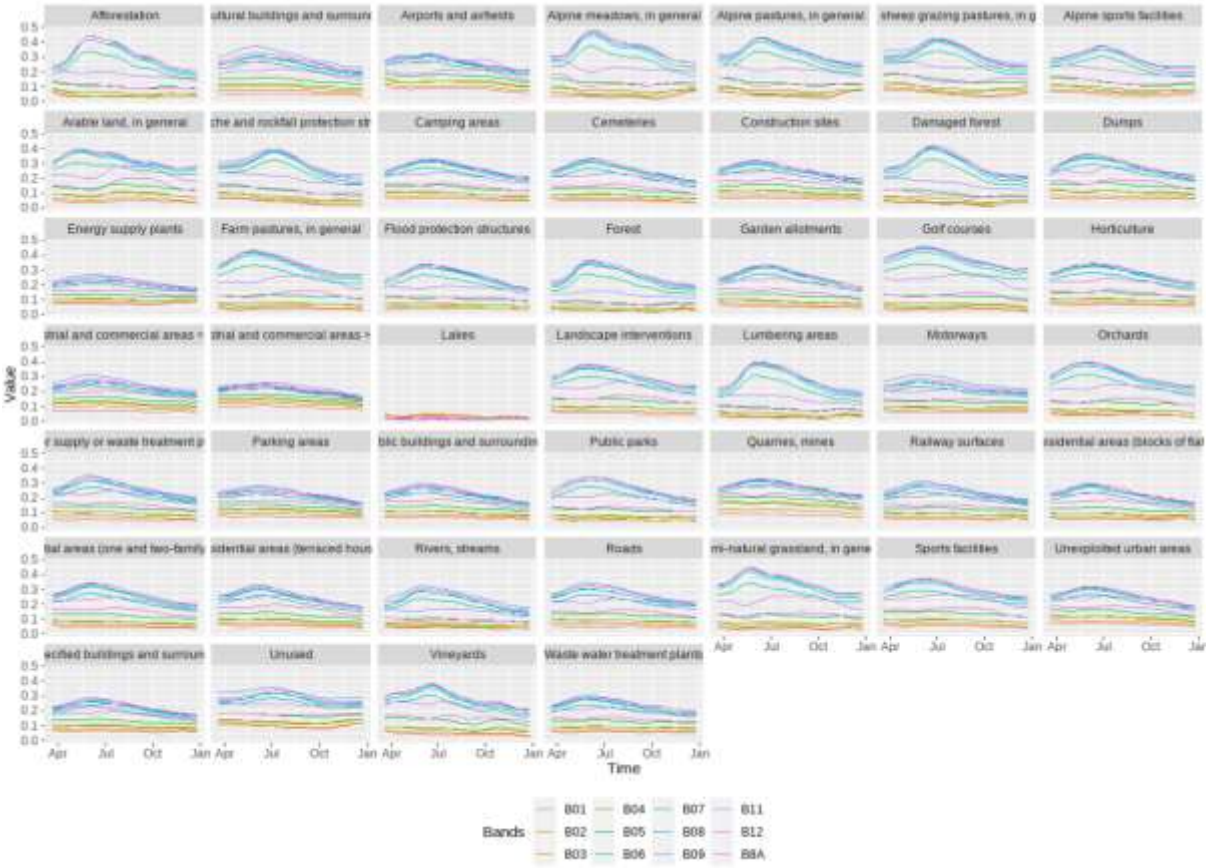

Figure 17: Time series of 12 spectral bands modelled for each reference data point for the 46 categories obtained using `sits_patterns ()` in SITS.

The application of the SOM (Self-Organizing Maps) technique using the `sits_som_map ()` function enables the reduction of noisy samples. A comparison of the histograms "before" and "after" pre-processing shows a significant reduction in terms of classes. This reduces the number of classes from 46 to 5 (Figure 18).

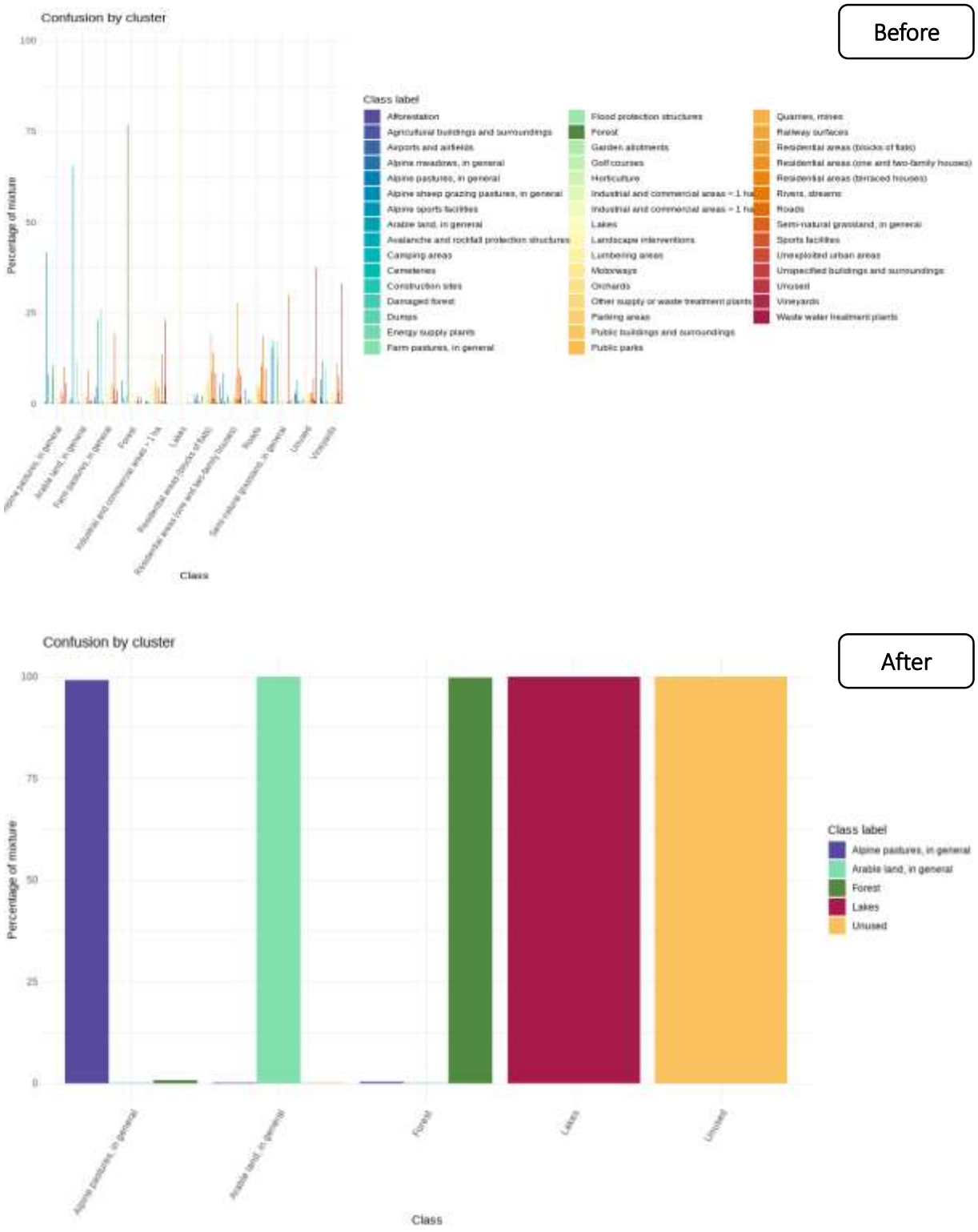

Figure 18: Confusion by cluster between the 10 Classes before and after pre-processing of the training samples.

The table 11 supplies a summary of the `sits_reduce_imbalance ()` function, which aims to equalize the sample distribution between the least frequent and most frequent labels. It outlines the results for the 10 most frequent classes “before pre-processing”, revealing a substantial reduction in the class number from 46 to 5 (Table 11).

Table 11: Comparing sample proportion before and after pre-processing of the training samples for 46 categories.

	Before pre-processing		After pre-processing	
Label	Count	Proportion	Count	Proportion
Alpine pastures, in general	22318	0.0781	1600	0.209
Arable land, in general	65866	0.2306	1600	0.209
Forest	731117	0.2560	1600	0.209
Lakes	31633	0.1107	1564	0.205
Unused	9932	0.0347	1283	0.168
Farm pastures, in general	17538	0.0614	-	-
Semi-natural grassland, in general	17198	0.0602	-	-
Lumbering areas	4639	0.0162	-	-
Residential areas (houses)	9207	0.0032	-	-
Roads	7297	0.0255	-	-
Vineyards	3669	0.0128	-	-
Industrial and commercial areas > 1ha	1761	0.0061	-	-

The outcomes of cross-validation results on the balanced training dataset indicate comparable levels of performance and accuracy between the two models (LTAE and tempCNN) and the RF model (Table 12). Noteworthy, no algorithms stand out as outperforming the others.

Table 12: Cross-validation for 46 categoreis highlighting Accuracy, 95% Confidence Intervals (CI), Kappa coefficients for the three tested methods: Random Forest (RF), Lightweight Temporal Self-Attention Encoder (LTAE) and Temporal Convolutional Neural Network (tempCNN).

	RF	LTAE	TempCNN
Accuracy	0.9957	0.9916	0.9953
95% CI	(0.9939,0.997)	(0.9893,0.9935)	(0.9935,0.9967)
Kappa	0.9946	0.9895	0.9941

N.B: F1- scores were not calculated due to the computation process' slowness s, limited available RAM, and Torch package issues.

4.4.2. STEP 3: Train ML/DL model

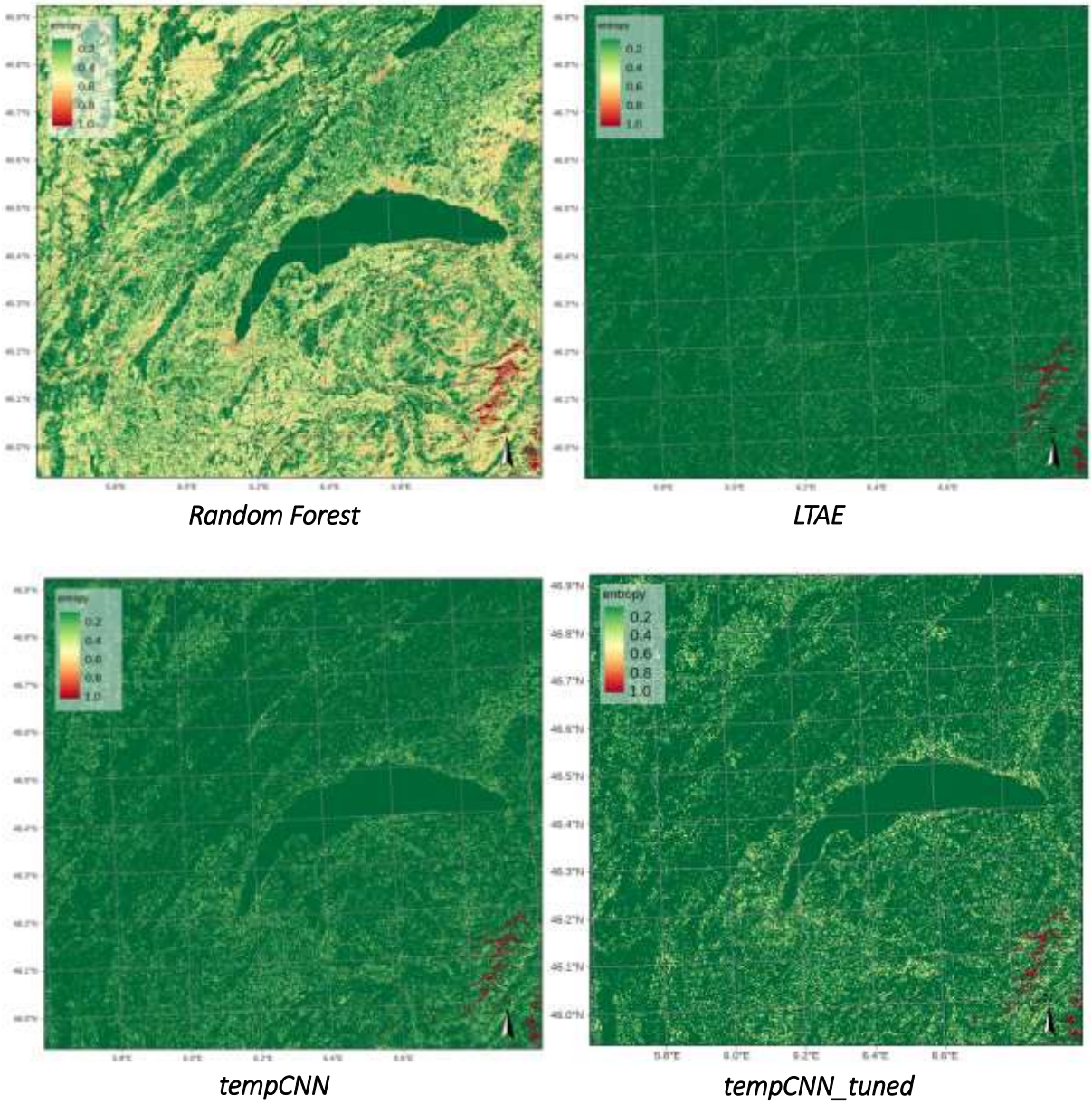

Figure 19: Uncertainty of RF, LTAE, tempCNN, LTAE_tuned and tempCNN_tuned models estimated through entropy estimation for the 46 categories.

An analysis of the uncertainty map indicates a notable reduction in uncertainty between the models (LTAE and tempCNN) and the RF model. In addition, a substantial drop in uncertainty quality is observed between the LTAE model and its optimized counterpart, LTAE_tuned model. Furthermore, a slight deterioration in uncertainty quality is observed between the models tempCNN and tempCNN_tuned Fiigure 19).

Table 9 provides a comprehensive summary of overall accuracy, user accuracy and producer accuracy for the three ML/DP models (RF, LTAE, tempCNN). Global accuracy quantifies the proportion of samples correctly classified on the reference data. Producer accuracy evaluates classification performance by capturing reference pixels of the vegetation cover type, focusing on omission errors. User accuracy assesses the probability that a pixel assigned to a specific category accurately represents that category in the field, by measuring errors of commission.

Table 13: Overall, User (UA), Producer (PA) accuracy for three models (RF, LTAE, tempCNN)

	RF		LTAE		LTAE_tuned		tempCNN		tempCNN_tuned	
Overall	0.88		0.82				0.88		0.9	
	UA	PA	UA	PA	N/A	N/A	UA	PA	UA	PA
Alpine pastures	0.71	0.78	0.76	0.64	N/A	N/A	0.76	0.76	0.78	0.77
Arable land	0.95	0.91	0.98	0.90	N/A	N/A	0.95	0.92	0.98	0.93
Forest	0.89	0.95	0.85	0.98	N/A	N/A	0.89	0.96	0.89	0.97
Lakes	0.99	0.97	0.98	0.98	N/A	N/A	0.96	0.99	0.95	0.99
Unused	0.78	0.37	0.81	0.56	N/A	N/A	0.68	0.41	0.80	0.52

N.B: The results for LTAE_tuned could not be collected due to the computation process' slowness and insufficient available RAM. The computation time for the 4 principal domains took around 3 days.

4.4.3. STEP 4: Land Use map production

The final map (Figure 20) was generated using an adapted tempCNN model. This model is recognized as the optimal choice for obtaining highly accurate classification results. Visually, the landscape is largely dominated by "forests", "arable land and grasslands" and "lakes and rivers".

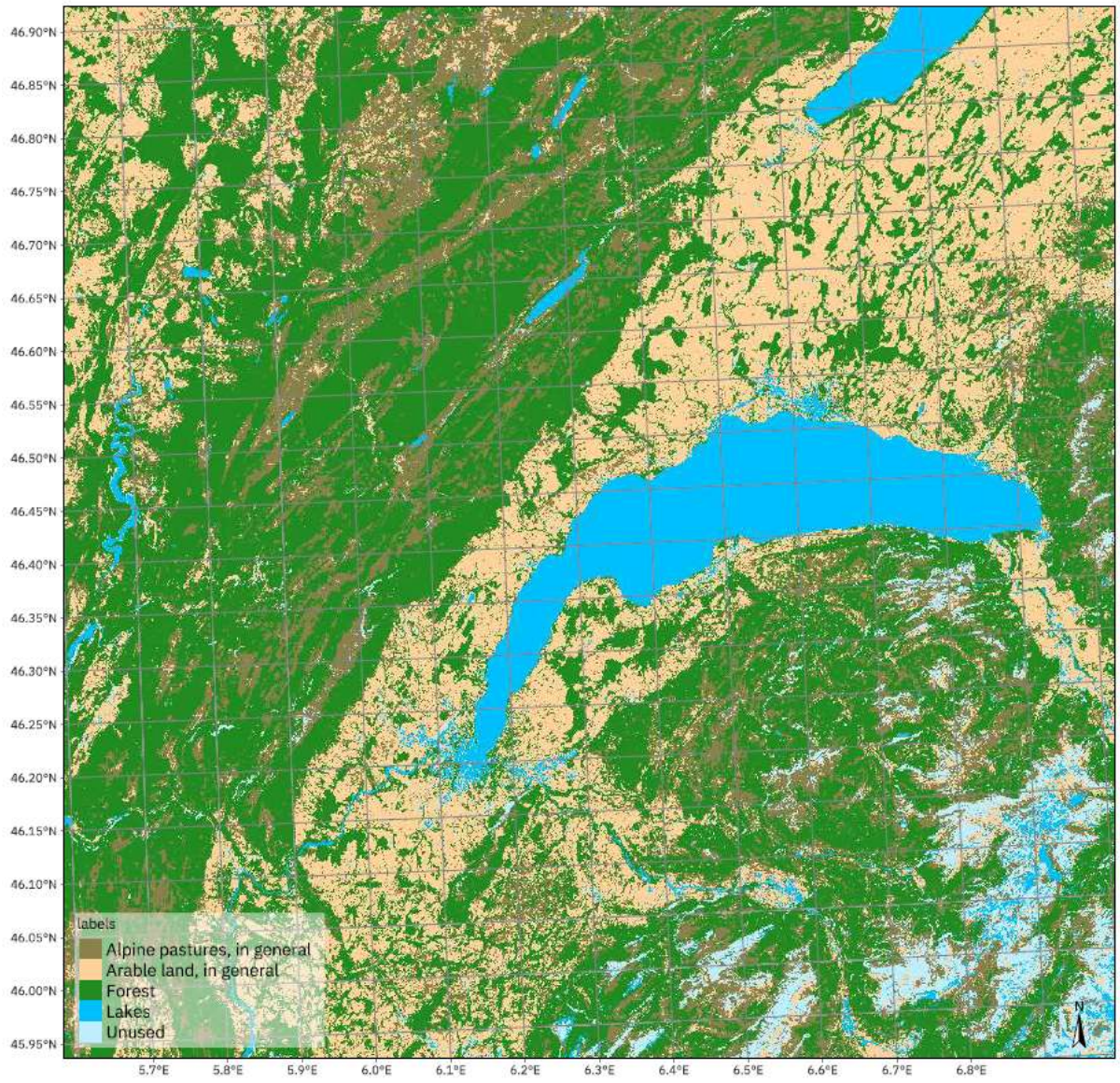

Figure 20: Final output of the classification obtained with the tempCNN model for 46 categories.

5. Discussion

The results demonstrate that the suggested approach meets the need for accurate and consistent and higher spatial resolution Land Use information at the national scale. Incorporating temporal features, considering intra-annual class variability, significantly improves accuracy and separability compared to single-date results (i.e. space-first). Spectral band time series integration over one-year enhanced classification performance. Deep Learning methos (LTAE/tempCNN) surpass traditional Machine Learning (RF), offering lower uncertainty for effective change detection (Alem and Kumar, 2022; Phan et al., 2022). Adequate training samples (i.ie clean and balanced samples) and hyperparameter adjustment are crucial to achieve relevant classification performance. The tempCNN algorithm, without adjusted hyperparameters, is identified as the most practical method for generating precise and reliable predictions, using Arealstatistik as training dataset.

To meet the challenges, it is vital to enhance the temporal resolution of Land use information. The current survey frequency is six years. This means that only gradual changes are considered (Thomas & Giuliani, 2023). Effective environmental monitoring is required to identify changes in land use, whether occurring slowly or rapidly. The uptake of annual land use outputs would provide a better understanding of land use processes and pressures and their impacts on climate, biodiversity, and ecosystems (Teixeira et al. 2014; Kennedy et al. 2014; Bontemps et al. 2013; Giuliani, 2023). The current approach only allows the use of training samples from the same acquisition period as the satellite data. To overcome this limitation, transfer learning methods can alleviate the constraint of using samples from the same acquisition period, allowing the model to be reused from one year to the next (Alem & Kumar 2022). Another limitation concerns the neglect of spatial autocorrelation in the training samples pre-processing, which can lead to over-fitting of the models tested (Phan et al. 2022). To alleviate this problem, point samples could be selected based on minimum distance criteria (e.g. 1 km) to ensure complete coverage of the entire study area.

The proposed approach focuses on the synergistic use of statistical sample surveys and satellite data for LULC mapping. Although the product generated has significant potential, it is not intended to replace the official Arealstatistik, but rather to complement it effectively. For instance, Portugal's land cover monitoring system is proactively addressing concerns by integrating space-based products to overcome the limitations of official land cover information obtained by photointerpretation every 3-5 years (Costa et al., 2023). This approach can be applied to Land Use. Despite the progress made by the Copernicus Land Monitoring Service's land use and cover (LUC) products, their applicability on a national scale remains limited (Verde et al., 2020). Portugal is improving its land cover information by integrating spatial products, in line with the preferences of national end-users. The official land use map is now complemented by an annual map generated by Sentinel-2 classification and machine learning algorithms, enabling continuous monitoring of land changes, as demonstrated by the LCMAP approach (Brown et al., 2019; Giuliani, 2023).

This is an exploratory project, which marks the first step towards an important goal: to actively contribute to the development of a unified, uninterrupted, and accurate land cover and land use monitoring system on a national scale. Based on these initial results, we have drawn up a list of numerous possibilities for improvement. It is therefore necessary to be able to evaluate them subsequently.

1. Integrate contextual data to improve the accuracy of Land use classification. The incorporation of Digital Elevation Models (DEMs) (Sang et al., 2021) could significantly improve the discrimination of topographical features (Giuliani, 2023). In addition, the use of spectral indices can add significant value to the characterization of different land use classes. The Normalized Difference Vegetation Index (NDVI) is essential for more accurate identification of classes and categories (i.e. Orchards, vineyards, horticulture; Arable and Grassland, Alpine grazing areas). It captures the nuances of the vegetation, contributing to a finer classification (Talukdar et al., 2020; Loukika et al., 2021). On the other hand, the Normalized Difference Building Index (NDBI) offers an exceptional opportunity to distinguish between built-up areas, transport zones and recreational areas, providing a better

understanding of urban land use (Prasad et al., 2021). The introduction of the Bare Soil Index (BSI) into the analysis can provide crucial information on areas devoid of vegetation, completing the characterization of different soil surfaces.

2. Improving the quality of training samples is essential for increasing the accuracy of land use classification (LUC). Samples must be representative and varied, faithfully reflecting the different land use classes. Sample quality relies on the relative accuracy of classes, complete spectral coverage, and adherence to an appropriate sampling plan. Satellite image time series (SITS) incorporate filtering and clustering techniques, while advanced sampling strategies can help improve sample quality (Pelletier et al., 2017). By using sophisticated filtering, clustering, and stratification approaches, we can significantly improve the representation of LUC classes in training samples, leading to more accurate classification (Maxwell et al., 2018). These adjustments aim to optimize the quality of the training data, thereby improving the performance of the LUC classification models.

3. Reducing uncertainty in image classification is a major challenge, requiring an accurate description of landscape diversity. The active learning approach, based on iterative sample selection, labeling and model retraining, offers a solution for continuously refining landscape understanding (Safonova et al., 2023; Chen et al., 2023; Giualini, 2023). Ensemble modeling, combining predictions of several models, improves the accuracy and robustness of results (Chai and Li, 2023; Du et al., 2023; Song et al., 2023). At the same time, the use of blending models enables the spectral contributions of different land covers within a pixel to be deciphered, leading to a more accurate assessment of class proportions within a blended pixel (Quintano et al., 2012; Halbgewachs et al., 2022; Shimabukuro et al., 2019; Giuliani, 2023). Integrating these strategies significantly reduces uncertainty and improves the reliability and accuracy of image classifications.

Finally, the use of satellite Earth observation data for LULC land use and cover classification involves the management of various trade-offs resulting from constraints such as the spectral, temporal and spatial resolution of sensors. It is critical to precisely reconcile the objectives of generating Land Use and Cover (LUC) maps with the abstract descriptions of ecosystems defined in typologies such as the FAO Land Use matrix. The key to reaching high accuracy in line with public policy is the ability to effectively describe land use change due to anthropization, and to use satellite data for land use and land cover classification. These constraints, complicate the classification of Earth Observation data to satisfy all users simultaneously. Existing LUC products are deficient in distinguishing natural savannahs from anthropogenic grasslands, which are essential for monitoring biodiversity loss. Rather than starting with what is measurable by satellite observations, many data producers adopt a data-independent taxonomy, sometimes out of step with observed reality. Thus, adopting a case-by-case, data-driven approach is imperative to take full advantage of available technologies to improve predictability.

Conclusion

Land use and land cover data are of crucial importance in managing global challenges such as climate change, biodiversity loss and pollution, in addition to food security, sustainable management and ecosystem services that have a vital impact on human well-being. LUC is emerging as a significant indictor of the human footprint on the environment, playing a critical role in the

development of effective environmental policies. Yet, despite the importance of LULC, there are still gaps in the thematically precise national products required to support downstream uses such as ecosystem hydrology and atmospheric models. Accurate, consistent, and high-resolution land use and land cover (LUC) products are necessary to improve air and water quality in urban areas, and to effectively monitor landscape dynamics. The suggested methodology represents a step towards this objective by merging photo-interpreted land use data from national statistics with time series of satellite observation data. Preliminary results indicate that the temporal approach performs better than the spatial approach, enabling better identification of classes by capturing intra-annual variability. Among the methods tested, tempCNN obtained better results than LTAE and traditional random forest.

The use of deep learning (DL) methods showed reduced uncertainties compared with Random Forest model, underlining the importance of these approaches for monitoring changes in land cover. The resulting map has high overall and class accuracy, as well as improved spatial resolution, while retaining the statistical significance of the official national dataset. Nevertheless, the challenge remains to increase the temporal resolution of the LUC information to produce a high-resolution annual map of land use in Switzerland. Such a product could complement the official land use statistics, optimizing the synergistic use of statistical surveys and satellite data for accurate land use mapping.

References

Alem, A., and Kumar, S. (2022). "Transfer Learning Models for Land Cover and Land Use Classification in Remote Sensing Image." *Applied Artificial Intelligence* 36 (1): 2014192. https://doi.org/10.1080/08839514.2021.2014192.

Appel, M, and Pebesma, E. (2019). "On-Demand Processing of Data Cubes from SatelliteImage Collections with the Gdalcubes Library." *Data* 4 (3): 92. https://doi.org/10.3390/data4030092

Bontemps, S., Defourny, P., Radoux, J., Van Bogaert, E., Lamarche, C., Achard, F., Mayaux, P., Boettcher, M., Brockmann, C., & Kirches, G. (2013). "Consistent Global Land Cover Maps for Climate Modelling Communities: Current Achievements of the ESA's Land Cover CCI." In, 9–13.

Bratic, G., Oxoli, D., & Brovelli, M-A. (2023). "Map of Land Cover Agreement: Ensambling Existing Datasets for Large-Scale Training Data Provision." *Remote Sensing* 15 (15): 3774. https://doi.org/10.3390/rs15153774

Brown, J.F., Tollerud, H.J., Barber, C.P., Zhou, Q., Dwyer, J.L., Vogelmann, J.E., Loveland, T.R., et al. (2019). "Lessons Learned Implementing an Operational Continuous United States National Land Change Monitoring Capability: The Land Change Monitoring, Assessment, and Projection (LCMAP) Approach." *Remote Sensing of Environment*, August, 111356. https://doi.org/10.1016/j.rse.2019.111356

Bronwyn, P., Huber, N., Nussbaumer, A., & Ginzler, C. (2023). The Habitat Map of Switzerland: A Remote Sensing, Composite Approach for a High Spatial and Thematic Resolution Product. *Remote Sensing* 15 (3): 643. https://doi.org/10.3390/rs15030643

Camara, G. (2020). "On the Semantics of Big Earth Observation Data for Land Classification" 20: 21–34. https://doi.org/10.5311/JOSIS.2020.20.645

Camara, G., Simoes, R., Souza, F., Pelletier, C., Sanchez. A, Andrade, R. P., Ferreira, K., & Queiroz, G. n.d. (Accessed 2024, january 19). **SITS:** Satellite Image Time Series Analysis on Earth Observation Data Cubes. https://e-sensing.github.io/sitsbook/introduction.html

Chai, B., and Li, P. (2023). "An Ensemble Method for Monitoring Land Cover Changes in Urban Areas Using Dense Landsat Time Series Data." *ISPRS Journal of Photogrammetry and Remote Sensing* 195 (January): 29–42. https://doi.org/10.1016/j.isprsjprs.2022.11.002

Chaves, M.E.D., Soares, A.R., Mataveli, G.A.V., Sánchez, A.H., Sanches, I.D. A. (2023). Semi-Automated Workflow for LULC Mapping via Sentinel-2 Data Cubes and Spectral Indices. Automation 2023, 4, 94–109. https://www.mdpi.com/2673-4052/4/1/7

Chaves, M.E.D., Picoli, M. C. A., & Sanches. I. D. (2020). Recent Applications of Landsat 8/OLI and Sentinel-2/MSI for Land Use and Land Cover Mapping: A Systematic Review. *Remote Sensing* 12 (18): 3062. https://doi.org/10.3390/rs12183062

Chen, H., Yang, L., and Wu, Q. (2023). "Enhancing Land Cover Mapping and Monitoring: An Interactive and Explainable Machine Learning Approach Using Google Earth Engine." *Remote Sensing* 15 (18): 4585. https://doi.org/10.3390/rs15184585

Cheng, X., Sun, Y., Zhang, W., Wang, Y., Cao, X., & Wang, Y. (2023). Application of Deep Learning in Multitemporal Remote Sensing Image Classification. *Remote Sensing* 15 (15): 3859. https://doi.org/10.3390/rs15153859

Comber, A., Fisher, P., & Wadsworth, R. (2005). "What Is Land Cover?" Environment and Planning B: Planning and Design 32 (2): 199–209. https://doi.org/10.1068/b31135.

Costa, H., Benevides, P., &Caetano, M. (2023). "The Portuguese Land Cover Monitoring System (SMOS): From Research and Development (R&D) to Operations." *REVISTA INTERNACIONAL MAPPING* 32 (210): 44–51. https://doi.org/10.59192/mapping.387

D'Andrimont, R., Verhegghen, A., Meroni, M., Lemoine, G., Strobl, P., Eiselt, B., Yordanov, M., Martinez-Sanchez, L., & Van der Velde, M. (2021). LUCAS Copernicus 2018: Earth-observation-relevant in situ data on land cover and use throughout the European Union; https://essd.copernicus.org/articles/13/1119/2021/

DETEC – Conseil fédéral suisse Stratégie pour le développement durable 2016 – 2019 (27 janvier 2016).

Du, H., Li, M., Xu, Y., & Zhou, C. (2023). "An Ensemble Learning Approach for Land Use/Land Cover Classification of Arid Regions for Climate Simulation: A Case Study of Xinjiang, Northwest China." *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 16: 2413–26. https://doi.org/10.1109/JSTARS.2023.3247624

Fawaz, H. I., Forestier, G., Weber, J., Idoumghar, L., & Muller, P-A. (2018). "Deep Learning for Time Series Classification: A Review." *arXiv:1809.04356 [Cs, Stat]*, September. http://arxiv.org/abs/1809.04356

FAO. (2023). FAOSTAT Analytical Brief 71. Land statistics and indicators 2000 – 2021. Global, regional, and country trends. https://www.fao.org/3/cc6907en/cc6907en.pdf

Federal Office of Topography Swisstopo. (2024). SwissTLM3D. https://www.swisstopo.admin.ch/en/landscape-model-swisstlm3d

Ferreira, K R., Queiroz, G. R., Vinhas, L., Marujo, R.F.B., Simoes, R.O.F., Picoli, M.C.A., Camara, G. et al. (2020). "Earth Observation Data Cubes for Brazil: Requirements, Methodology and Products." *Remote Sensing* 12 (24): 4033. https://doi.org/10.3390/rs12244033

Foley, J. A., DeFries, R., Asner, G. P., Barford, C., Bonan, G., Carpenter, S. R., Chapin, F. S., Coe, M. T., Daily, G. C., Gibbs, H. K., et al. (2005). Global Consequences of Land Use. Science 2005, 309, 570–574. https://www.science.org/doi/10.1126/science.1111772

Frenay, B., and Verleysen, M. (2014). "Classification in the Presence of Label Noise: A Survey." *IEEE Transactions on Neural Networks and Learning Systems* 25 (5): 845–69. https://doi.org/10.1109/TNNLS.2013.2292894

FSO – Federal Statistical Office. Land Use in Switzerland: Results of the Swiss land use statistics 2018. https://www.bfs.admin.ch/bfs/en/home/statistics/territory-environment/land-use-cover.assetdetail.19365054.html

Giuliani, G, Chatenoux, B., De Bono, A., Rodila, D., Richard, J-F., Allenbach, K., Dao, H., & Peduzzi, P. (2017). "Building an Earth Observations Data Cube: 22 Lessons Learned from the Swiss Data Cube (SDC) on Generating Analysis Ready Data (ARD)." *Big Earth Data* 1 (1): 1–18. https://doi.org/10.1080/20964471.2017.1398903

Government of Canada. (2015). Land Cover & Land Use

Grêt-Regamey, A., Altwegg, J., Sirén, E.A., van Strien, M. J., & Weibel, B. (2017). Integrating Ecosystem Services into Spatial Planning—A Spatial Decision Support Tool. Landsc. Urban Plan. 2017, 165, 206–219. https://www.sciencedirect.com/science/article/pii/S0169204616300615

Giuliani, G. (2023). Time-first approach for Land Cover mapping using Big Earth Observation Data time-series in a Data Cube – a case study from the Lake Geneva region (Switzerland)

Giuliani, G., Rodila, D., Külling, N., Maggini, R., & Lehmann, A. (2022). Downscaling Switzerland Land Use/Land Cover Data Using Nearest Neighbors and an Expert System.

Gonseth, Y., W. Wohlgemuth, B. Sansonnens, & Buttler. A. (2001). "Les Régions Biogéographiques de La Suisse – Explications et Division Standard." 137. Cahier de l'environnement. Bern, Switzerland : Office fédéral de l'environnement, des forêts et du paysage.

Guo, H. (2020). Big Earth Data Facilitates Sustainable Development Goals. *Big Earth Data* 0 (0): 1–2 https://doi.org/10.1080/20964471.2020.1730568

Halbgewachs, M., Wegmann, M., & da Ponte, E. (2022). "A Spectral Mixture Analysis and Landscape Metrics Based Framework for Monitoring Spatiotemporal Forest Cover Changes: A Case Study in Mato Grosso, Brazil." *Remote Sensing* 14 (8): 1907. https://doi.org/10.3390/rs14081907

Hanson, M. The Open-Source Software Ecosystem for Leveraging Public Datasets in Spatio-Temporal Asset Catalogs (STAC). https://ui.adsabs.harvard.edu/abs/2019AGUFMIN23B..07H/abstract (accessed on 20 january 2023).

Hermosilla, T., Wulder, M.A., White, J.C., Coops, N.C., & Hobart, G.W. (2018). "Disturbance-Informed Annual Land Cover Classification Maps of Canada's Forested Ecosystems for a 29-Year Landsat Time Series." *Canadian Journal of Remote Sensing* 0 (0): 1–21. https://doi.org/10.1080/07038992.2018.1437719

IPBES. Summary for Policymakers of the Thematic Assessment of Land Degradation and Restoration; IPBES Secretariat: Bonn, Germany, 2018. https://www.ipbes.net/assessment-reports/ldr

Kennedy, R.E., Serge, A., Cohen, W.B., Gomez, C., Griffiths, P., Hais, M., Healey, S.P., et al. (2014). "Bringing an Ecological View of Change to Landsat-based Remote Sensing." *Frontiers in Ecology and the Environment* 12 (6): 339–46. https://doi.org/10.1890/130066

Loukika, K. N., Keesara, V. R., & Sridhar, V. (2021). Analysis of Land Use and Land Cover Using Mahine Learning Algorithms on Google Earth Engine for Munneru River Basin, India. https://doi.org/10.3390/su132413758

Maus, V., Camara, G., Appel, M., & Pebesma, E. (2019). "dtwSat: Time-Weighted Dynamic Time Warping for Satellite Image Time Series Analysis in R." *Journal of Statistical Software* 88 (1): 1–31. https://doi.org/10.18637/jss.v088.i05

Maxwell, A.E., Warner, T.A., and Fang Fang. (2018). "Implementation of Machine- LearningClassification in Remote Sensing: An Applied Review." *International Journal of Remote Sensing* 39 (9): 2784–2817. https://doi.org/10.1080/01431161.2018.1433343

Metternicht, G., Mueller, N., & Lucas, R. (2020). "Digital Earth for Sustainable Development Goals." In *Manual of Digital Earth*, edited by Huadong Guo, Michael F. Goodchild, and Alessandro Annoni, 443–71. Singapore: Springer. https://doi.org/10.1007/978-981-32-9915-3_13

Mushtaq, F., Henry, M., O'Brien, C. D., Di Gregorio, A., Jalal, R., Latham, J., Muchoney, D., et al. (2022). "An International Library for Land Cover Legends: The Land Cover Legend Registry." *Land* 11 (7): 1083. https://doi.org/10.3390/land11071083.

NCCS. 2018. "CH2018 – Climate Scenarios for Switzerland, Technical Report." Zurich, Switzerland.

Office Fédéral de l'environnement (OFEV). (2022). Les régions biogéographiques de la Suisse.

Owers, C. J., Lucas, R. M., Clewley, D., Planque, C., Punalekar, S., Tissott, B., Chua, S. M. T., Bunting, P., Mueller, N., & Metternicht, G. (2021). "Living Earth: Implementing National Standardised Land Cover Classification Systems for Earth Observation in Support of Sustainable Development." *Big Earth Data* 0 (0): 1–23. https://doi.org/10.1080/20964471.2021.1948179

Papoutsis, I., Bountos, N.I., Zavras, A., Michail, D., & Tryfonopoulos, C. (2023). "Benchmarking and Scaling of Deep Learning Models for Land Cover Image Classification." *ISPRS Journal of Photogrammetry and Remote Sensing* 195 (January): 250–68. https://doi.org/10.1016/j.isprsjprs.2022.11.012

Pelletier, C., Valero, S., Inglada, J., Champion, N., Marais Sicre, C., & Dedieu, G. (2017). "Effect of Training Class Label Noise on Classification Performances for Land Cover Mapping with Satellite Image Time Series." *Remote Sensing* 9 (2): 173. https://doi.org/10.3390/rs9020173

Pelletier, C., Webb, G.I., & Petitjean, F. (2019). "Temporal Convolutional Neural Network for the Classification of Satellite Image Time Series." *Remote Sensing* 11 (5): 523. https://doi.org/10.3390/rs11050523

Planque, C., Punalekar, S., Lucas, R., Chognard, S., Owers, J. C., Clewley, D., Bunting, P., Sykes, H., & Horton, C. (2020). Living Wales: Automatic and Routine Environmental Monitoring Using Multi-Source Earth Observation Data.

Prasad, P, Loveson, V. J., Chandra, P., & Kotha, M. (2022). Evaluation and comparison of the earth observing sensors in land cover/ land use studies using machine learning algorithms. https://doi.org/10.1016/j.ecoinf.2021.101522

Prem Chandra, P., Koutsias, N., Petropoulos, P. G., Srivastava, K. P., & Ben Dor. E. (2019). "Land Use/Land Cover in View of Earth Observation: Data Sources, Input Dimensions, and Classifiers—a Review of the State of the Art." *Geocarto International*, June. https://www.tandfonline.com/doi/abs/10.1080/10106049.2019.1629647

Quintano, C., Fernandez-Manso, A., Shimabukuro, Y.E. and Pereira, G. (2012). "Spectral Unmixing." *International Journal of Remote Sensing* 33 (17): 5307–40. https://doi.org/10.1080/01431161.2012.661095

Rwanga, S. S. & Ndambuki, J. M. (2017). Accuracy Assessment of Land Use/ Land Cover classification using remote sensing and GIS. International Journal of Geosciences Vol.08 No.04(2017).

Rolf, S., Camara, G., Queiroz, G., Souza, F., Andrade, R. P., Santos, L., Carvalho, A., & Ferreira, K. (2021). Satellite Image Time Series Analysis for Big Earth Observation Data. *Remote Sensing* 13 (13): 2428. https://doi.org/10.3390/rs13132428

Rounsevell M. D., Reginster, I., Araújo, M.B., Carter, T. R., Dendoncker, N., Ewert, F., House, J.I., Kankaanpää, S., Leemans, R., Metzger, M.J., et al. (2006). A Coherent Set of Future Land Use Change Scenarios for Europe. Agric. Ecosyst. Environ. 2006, 114, 57–68.

Safonova, A., Ghazaryan, G., Stiller, S., Main-Knorn, M., Nendel, C., & Ryo, M. (2023). "Ten Deep Learning Techniques to Address Small Data Problems with Remote Sensing." *International Journal of Applied Earth Observation and Geoinformation* 125 (December): 103569. https://doi.org/10.1016/j.jag.2023.103569

Sainte Fare Garnot, V., Landrieu, L., Giordano, S., & Chehata, N. (2020). "Satellite Image Time Series Classification with Pixel-Set Encoders and Temporal Self-Attention." In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 12322–31. https://doi.org/10.1109/CVPR42600.2020.01234

Sang, X., Guo, Q., Wu, X., Xie, T., He, C., Zang, J., Qiao, Y., Wu, H., & Li, Y. (2021). "The Effect of DEM on the Land Use/Cover Classification Accuracy of Landsat OLI Images." *Journal of the Indian Society of Remote Sensing* 49 (7): 1507–18. https://doi.org/10.1007/s12524-021-01318-5

SFSO - Swiss Federal Statistical Office. Land Use in Switzerland—Results of the Swiss Land Use Statistics; SFO: Neuchâtel, Switzerland, 2013.

SFSO - Swiss Federal Statistical Office. The Changing Face of Land Use: Land Use Statistics of Switzerland; SFO: Neuchâtel, Switzerland, 2001; p. 32.

Shetty, S., Gupta, P.K., Belgiu, M., & Srivastav. S.K. (2021). "Assessing the Effect of Training Sampling Design on the Performance of Machine Learning Classifiers for Land Cover Mapping Using Multi-Temporal Remote Sensing Data and Google Earth Engine." *Remote Sensing* 13 (8): 1433. https://doi.org/10.3390/rs13081433

Shimabukuro, Y.E., and Ponzoni, F.J. (2019). *Spectral Mixture for Remote Sensing: Linear Model and Applications*. Springer.

Simoes, R., Camara, G., Queiroz, G., Souza, F., Andrade, P. R., Santos, L., Carvalho, A., & Ferreira, K. (2021). "Satellite Image Time Series Analysis for Big Earth Observation Data." *Remote Sensing* 13 (13): 2428. https://doi.org/10.3390/rs13132428

Song, L., Estes, A.B., and Estes, L.D. (2023). "A Super-Ensemble Approach to Map Land Cover Types with High Resolution over Data-Sparse African Savanna Landscapes." *International Journal of Applied Earth Observation and Geoinformation* 116 (February): 103152. https://doi.org/10.1016/j.jag.2022.103152

Phan, T-N., Dashpurev, B., Wiemer, F., & Lehnert, W. L. (2022). "A Simple, Fast, and Accurate Method for Land Cover Mapping in Mongolia." *Geocarto International* 37 (26): 14432–50. https://doi.org/10.1080/10106049.2022.2087759

Talukdar, S., Singha, P., Mahato, S., Shahfahad, Pal, S., Liou, Y-A., & Rahman, A. (2020). Land-Use Land-Cover Classification by Machine Learning Classifiers for Satellite Observations – A Review. https://doi.org/10.3390/rs12071135

Teixeira, Z., Teixeira, H., & Marques, J.C. (2014). "Systematic Processes of Land Use/Land Cover Change to Identify Relevant Driving Forces: Implications on Water Quality." *Science of* 26 *The Total Environment* 4.

Thomas, I. N., & Giuliani, G. (2023). Exploring Switzerland's Land Cover Change Dynamics Using a National Statistical Survey. Land 2023, 12, 1386. https://doi.org/10.3390/land12071386

United Nations Department of Economic and Social Affairs. Sustainable Land Use for the 21st Century. United Nations Department of Economic and Social Affairs: New York, NY, USA, 2012. https://sustainabledevelopment.un.org/content/documents/1124landuse.pdf

UNCCD. (2022). "Global Land Outlook: Land Restoration for Recovery and Resilience." Bonn, Germany : UNCCD.

UNGA. (2015). *United Nations General Assembly. Sustainable Development Goals (SDGs)*. New York, NY, USA: United Nations.

Verde, N., Kokkoris, I.P., Georgiadis, C., Kaimaris, D., Dimopoulos, P., Mitsopoulos, I., & Mallinis, G. (2020). "National Scale Land Cover Classification for Ecosystem Services Mapping and Assessment, Using Multitemporal Copernicus EO Data and Google Earth Engine." *Remote Sensing* 12 (20): 3303. https://doi.org/10.3390/rs12203303

Wulder, M. A., Roy, P. D., Radeloff C. V., Loveland, R. T., Anderson, C. M., Johnson, M. D., Healey, S., Zhu, Z. et al. (2022). Fifty years of Landsat science and impacts. https://doi.org/10.1016/j.rse.2022.113195

Yowargana, H. P., Zulkarnain, M. T., Mohamad, F., Goib, B. K., Hultera, P., Sturn, T., et al. (2022). "A National-Scale Land Cover Reference Dataset from Local Crowdsourcing Initiatives in Indonesia." *Scientific Data* 9 (1): 574. https://doi.org/10.1038/s41597-022-01689-5.

Zander S. V., Barton, N. D., Chakraborty, T., Simensen, T., & Singh, G. (2022) "Global 10 m Land Use Land Cover Datasets: A Comparison of Dynamic World, World Cover and Esri Land Cover." *Remote Sensing* 14 (16): 4101. https://doi.org/10.3390/rs14164101

Zoltan, S., Geller, G. N., Tsendbazar, N-E., See, L., Griffiths, P., Fritz, S., Gong, P., Herold, M., Mora, B., & Obregon, A. (2020). Addressing the Need for Improved Land Cover Map Products for Policy Support. *Environmental Science & Policy* 112. 28–35. https://pubmed.ncbi.nlm.nih.gov/33013195/

ANNEXE

- Installing the packages

```
####Packages####
##Installing packages for SITS
install.packages("sf")
install.packages("terra")
install.packages("sits", dependencies = TRUE)
install.packages("devtools")
devtools::install_github("e-sensing/sitsdata")
install.packages("torch")
##Install the bundle provided in the notebook##
options(warn=-1)
system("cp -u -R ../input/sits-bundle/sits-bundle/* /usr/local/lib/R/site-library/")
##load the sits library##
library(sits)
library(sitsdata)
library(torch)
library(dplyr)
library(purrr)
##options for plotting##
options("sp_evolution_status" = 0)
library(sp)

##configure graphs output##
options(repr.plot.width = 14, repr.plot.height = 8.5)
```

- Creating and regularizing the data cube

```
sits_list_collections()
####Creating data cubes####
##Assessing Amazon Web Services##
##create a data cube from Sentinel-2 image covering Geneva area##
g_cube <- sits_cube(
  source = "AWS",
  collection = "SENTINEL-S2-L2A-COGS",
  tiles = "31TGM",
  bands = c("B01", "B02", "B03", "B04", "B05", "B06", "B07", "B08", "B8A", "B09", "B11", "B12", "CLOUD"),
  start_date = "2018-01-01",
  end_date = "2018-12-31"
)

# Show the different timelines of the cube tiles
sits_timeline(g_cube)
####Regularizing data cubes####
##Create a directory to store files##
##creating an irregular data cube from AWS##
g_reg_cube <- sits_regularize(
  cube       = g_cube,
  output_dir = ".",
  res        = 10,
  period     = "P5D",
  multicores = 4
)
```

- Preparing the samples

```
##Compute time-series for samples##
#retrieve a list of samples described by a CSV file
samples_LU04_18_TRAIN <- read.csv("C:/Users/chafter1/Desktop/Stage Geomatique/Projet_Geneve/SITS/NOLU04/samples/samplesOFS_LU04_18_train.csv")
#compute TS from local DC based on samples Done directly in the R console using the get_ts_by_chunk function
source ('get_ts_by_chunk.R')
samples <- get_ts_by_chunk(
    cube = g_reg_cube,
    samples = samples_LU04_18_TRAIN,
    multicores = 24,
    n_chunks = 40,
    output_dir = 'C:/Users/chafter1/Desktop/Stage Geomatique/Projet_Geneve/SITS/NOLU04/TS_LU04_18'
)
#Summary
summary (samples)
```

```
##Patterns##
patterns <- sits_patterns(samples)
plot (patterns)
```

- Training the models

```
####Training and running Machine/deep learning models####
##Random forest##
rf_model <- sits_train(
    samples = balanced_samples,
    ml_method = sits_rfor()
)

##Lightweight Temporal Self-Attention Encoder [LTAE]##
ltae_model <- sits_train(balanced_samples,
    sits_lighttae(
    epochs               = 150,
    batch_size           = 64,
    optimizer            = torchopt::optim_adamw,
    validation_split     = 0.2,
    verbose              = FALSE
))

##Temporal Convolutional Neural Network [TempCNN]##
tempcnn_model <- sits_train(balanced_samples,
    sits_tempcnn(
    optimizer         = torchopt::optim_adamw,
    cnn_layers        = c(128, 128, 128),
    cnn_kernels       = c(7, 7, 7),
    cnn_dropout_rates = c(0.2, 0.2, 0.2),
    epochs            = 40,
    batch_size        = 64,
    validation_split  = 0.2,
    verbose              = FALSE
  )
)
```

- Classification of Raster Data Cubes

```
####Classification of Raster Data Cubes####
##Random forest##
gva_probs_rf <- sits_classify(g_reg_cube,
    ml_model = rf_model,
    multicores = 24,
    memsize = 72,
    output_dir = "C:/Users/chafter1/Desktop/Stage Geomatique/Projet_Geneve/SITS/NOLU04",
    version = 'rf_v5',
    crs = 4326,
    progress = TRUE
)
```

```
##Lightweight Temporal Self-Attention Encoder [LTAE]##
gva_probs_ltae <- sits_classify(g_reg_cube,
                                ml_model = ltae_model,
                                multicores = 24,
                                memsize = 72,
                                output_dir = "C:/Users/chafter1/Desktop/Stage Geomatique/
                                version = 'ltae_v1',
                                crs = 4326,
                                progress = TRUE
)
```

```
## Temporal Convolutional Neural Network [TempCNN]##
gva_probs_tempcnn <- sits_classify(g_reg_cube,
                                   ml_model = tempcnn_model,
                                   multicores = 24,
                                   memsize = 72,
                                   output_dir = "C:/Users/chafter1/Desktop/Stage Geomatiqu
                                   version = 'tempcnn-v1',
                                   crs = 4326,
                                   progress = TRUE
)
```

- Performing spatial smoothing

```
####Perform spatial smoothing####
##Random forest##
gva_bayes_rf <- sits_smooth(  cube = gva_probs_rf,
                              multicores = 24,
                              memsize = 72,
                              output_dir = "C:/Users/chafter1/Desktop/Stage Geomatique/Projet_
                              version = 'rf_v1'
)
```

```
##Lightweight Temporal Self-Attention Encoder [LTAE]##
gva_bayes_ltae <- sits_smooth(  cube = gva_probs_ltae,
                                multicores = 24,
                                memsize = 72,
                                output_dir = "C:/Users/chafter1/Desktop/Stage Geomatique/Proje
                                version = 'ltae_v1'
)
```

```
## Temporal Convolutional Neural Network [TempCNN]##
gva_bayes_tempcnn <- sits_smooth(  cube = gva_probs_tempcnn,
                                   multicores = 24,
                                   memsize = 72,
                                   output_dir = "C:/Users/chafter1/Desktop/Stage Geomatique/Pr
                                   version = 'tempcnn-v1'
)
```

- Final map/ Label probability

```
####Label the probability file & plot the final map####
##Random forest##
gva_map_rf <- sits_label_classification(
  cube = gva_bayes_rf,
  output_dir = "C:/Users/chafter1/Desktop/Stage Geomatique/Projet_Gene
)

##Lightweight Temporal Self-Attention Encoder [LTAE]##
gva_map_ltae <- sits_label_classification(
  cube = gva_bayes_ltae,
  output_dir = "C:/Users/chafter1/Desktop/Stage Geomatique/Projet_Gene
  version = 'ltae_v1'
)

##Temporal Convolutional Neural Network [TempCNN]##
gva_map_tempcnn <- sits_label_classification(
  cube = gva_probs_tempcnn,
  output_dir = "C:/Users/chafter1/Desktop/Stage Geomatique/Projet_Gene
  version = 'tempcnn-v1'
)
```

- Measurement of Validation, Accuracy, Uncertainty

```
####Validation, Accuracy, Uncertainty####
###Uncertainty###
##Random forest##
uncert_rf <- sits_uncertainty(
  cube = gva_probs_rf,
  type = 'entropy',
  multicores = 24,
  memsize = 72,
  output_dir = "C:/Users/chafter1/Desktop/Stage Geomatique/Projet_
)

##Lightweight Temporal Self-Attention Encoder [LTAE]##
uncert_ltae <- sits_uncertainty(
  cube = gva_probs_ltae,
  type = 'entropy',
  multicores = 24,
  memsize = 72,
  output_dir = "C:/Users/chafter1/Desktop/Stage Geomatique/Projet_
  version = 'ltae_v1'
)

##Temporal Convolutional Neural Network [TempCNN]##
uncert_tempcnn <- sits_uncertainty(
  cube = gva_probs_tempcnn,
  type = 'entropy',
  multicores = 24,
  memsize = 72,
  output_dir = "C:/Users/chafter1/Desktop/Stage Geomatique/Projet_
  version = 'tempcnn-v1'
)
```

```r
###Accuracy###
#retrieve a list of samples described by a CSV file
samples_acc_LU04_18<- read.csv("C:/Users/chafter1/Desktop/Stage Geo

##Random forest##
area_acc_rf <- sits_accuracy(gva_map_rf,
  validation = samples_acc_LU04_18
)

##Lightweight Temporal Self-Attention Encoder [LTAE]##
area_acc_ltae <- sits_accuracy(gva_map_ltae,
                               validation = samples_acc_LU04_18
)
##Temporal Convolutional Neural Network [TempCNN]##
area_acc_tempcnn <- sits_accuracy(gva_map_tempcnn,
                                  validation = samples_acc_LU04_18
)

###Validation###
##Random forest##
val_rfor <- sits_kfold_validate(
    samples = balanced_samples,
    ml_method = sits_rfor(),
    folds = 5,
    multicores = 24
)
F1_rfor <- sits_kfold_validate(
  samples = samples,
  ml_method = sits_rfor(),
  folds = 5,
  multicores = 24
)
##Lightweight Temporal Self-Attention Encoder [LTAE]##
val_ltae <- sits_kfold_validate(
  samples = balanced_samples,
  ml_method = sits_lighttae(),
  folds = 5,
  multicores = 24
)

F1_ltae <- sits_kfold_validate(
  samples = samples,
  ml_method = sits_lighttae(),
  folds = 5,
  multicores = 24
)
```

```
##Temporal Convolutional Neural Network [TempCNN]##
val_tempcnn <- sits_kfold_validate (
  samples = balanced_samples,
  ml_method = sits_tempcnn(),
  folds = 5,
  multicores = 24
)
F1_tempcnn <- sits_kfold_validate (
  samples = samples,
  ml_method = sits_tempcnn(),
  folds = 5,
  multicores = 24
)
```

- Tuning TempCNN

```
####Hyper-parameters tunning[TempCNN]####
###Temporal Convolutional Neural Network [TempCNN]###
tuned_tempcnn <- sits_tuning(
    samples = balanced_samples,
    samples_validation = NULL,
    validation_split = 0.2,
    ml_method = sits_tempcnn(),
    params = sits_tuning_hparams(
    optimizer = torchopt::optim_adamw,
    opt_hparams = list(
    lr = beta(0.35, 10),
    weight_decay = beta(0.1, 2))),
    trials = 100,
    multicores = 24,
    progress = TRUE
)
# save
saveRDS(tuned_tempcnn, file = 'C:/Users/chafter1/Desktop/Stage Geomatique/

##Obtain accuracy, Kappa, Lr and weight decay for the 10 best results##
##Hyper-parameters are organized as a list##
hparams_10 <- tuned_tempcnn[1:10, ]$opt_hparams
##Extract learning rate and weight decay from the list##
lr_10 <- purrr::map_dbl(hparams_10, function(h) h$lr)
wd_10 <- purrr::map_dbl(hparams_10, function(h) h$weight_decay)
##Create a tibble to display the results##
best_10 <- tibble::tibble(
    accuracy = tuned_tempcnn[1:10, ]$accuracy,
    kappa = tuned_tempcnn[1:10, ]$kappa,
    lr = lr_10,
    weight_decay = wd_10
)
```

```r
##Train a new model##
tempcnn_model_tuned <- sits_train(
        samples = balanced_samples,
        ml_method = sits_tempcnn(
          epochs                = 100,
          batch_size            = 64,
          optimizer             = torchopt::optim_adamw,
          validation_split      = 0.2,
          verbose               = FALSE,
          cnn_layers            = c(128, 128, 128),
          cnn_kernels           = c(7, 7, 7),
          cnn_dropout_rates     = c(0.2, 0.2, 0.2),
          opt_hparams = list(
            lr = 0.00129,
            weight_decay = 0.0000965
)))

##Classify##
#read it if needed
tempcnn_model_tuned <- readRDS(file='C:/Users/chafter1/Desktop/Stage Geom
gva_probs_tempcnn_tuned <- sits_classify(
    data = g_reg_cube,
    ml_model = tempcnn_model_tuned,
    multicores = 24,
    memsize = 72,
    output_dir = "C:/Users/chafter1/Desktop/Stage Geomatique/Projet_Geneve
    version = 'tempcnn-v1-tuned',
    crs = 4326,
    progress = TRUE
)

##Smooth##
gva_bayes_tempcnn_tuned <- sits_smooth(
    cube = gva_probs_tempcnn_tuned,
    multicores = 24,
    memsize = 72,
    version = 'tempcnn-v1-tuned',
    output_dir = "C:/Users/chafter1/Desktop/Stage Geomatique/Projet_Geneve
)

##Label##
gva_map_tempcnn_tuned <- sits_label_classification(
    cube = gva_bayes_tempcnn_tuned,
    output_dir = "C:/Users/chafter1/Desktop/Stage Geomatique/Projet_Geneve
    version = 'tempcnn-tuned'
)
```

```
##Uncertainty##
uncert_tempcnn_tuned <- sits_uncertainty(
     cube = gva_probs_tempcnn_tuned,
     type = 'entropy',
     multicores = 24,
     memsize = 72,
     output_dir = "C:/Users/chafter1/Desktop/Stage (
     version = 'tempcnn-v1-tuned'
     )
##Accuracy##
#retrieve a list of samples described by a CSV file
samples_acc_LU04_18<- read.csv("C:/Users/chafter1/Desktop/Stage Geomati(
area_acc_tempcnn_tuned <- sits_accuracy(gva_map_tempcnn_tuned,
    validation = samples_acc_LU04_18)
```

- Tuning LTAE

```
####Hyper-parameters tunning[LTAE]####
###Lightweight Temporal Self-Attention Encoder  [LTAE]###
tuned_ltae <- sits_tuning(
  samples = balanced_samples,
  samples_validation = NULL,
  validation_split = 0.2,
  ml_method = sits_lighttae(),
  params = sits_tuning_hparams(
    optimizer = torchopt::optim_adamw,
    opt_hparams = list(
      lr = beta(0.35, 10),
      weight_decay = beta(0.1, 2))),
  trials = 100,
  multicores = 24,
  progress = TRUE
)
# save
saveRDS(tuned_ltae, file = 'C:/Users/chafter1/Desktop/Stage Geomatique/Pro

##Obtain accuracy, Kappa, Lr and weight decay for the 10 best results##
##Hyper-parameters are organized as a list##
hparams_10_ltae <- tuned_ltae[1:10, ]$opt_hparams
##Extract learning rate and weight decay from the list##
lr_10_ltae <- purrr::map_dbl(hparams_10_ltae, function(h) h$lr)
wd_10_ltae <- purrr::map_dbl(hparams_10_ltae, function(h) h$weight_decay)
##Create a tibble to display the results##
best_10_ltae <- tibble::tibble(
  accuracy = tuned_ltae[1:10, ]$accuracy,
  kappa = tuned_ltae[1:10, ]$kappa,
  lr = lr_10_ltae,
  weight_decay = wd_10_ltae
)
```

```r
##Train a new model with optimized parameters##
ltae_model_tuned <- sits_train(
  samples = balanced_samples,
  ml_method = sits_lighttae(
    epochs                = 100,
    batch_size            = 64,
    optimizer             = torchopt::optim_adamw,
    validation_split      = 0.2,
    verbose               = FALSE,
    opt_hparams = list(
      lr = 0.000071,
      weight_decay = 0.013
    )))

##Classify##
#read it if needed
ltae_model_tuned <- readRDS(file='C:/Users/chafter1/Desktop/Sta
gva_probs_ltae_tuned <- sits_classify(
  data = g_reg_cube,
  ml_model = ltae_model_tuned,
  multicores = 24,
  memsize = 72,
  output_dir = "C:/Users/chafter1/Desktop/Stage Geomatique/Proj
  version = 'ltae_v1_tuned',
  crs = 4326,
  progress = TRUE
)
##Smooth##
gva_bayes_ltae_tuned <- sits_smooth(
  cube = gva_probs_ltae_tuned,
  multicores = 24,
  memsize = 72,
  version = 'ltae_v1_tuned',
  output_dir = "C:/Users/chafter1/Desktop/Stage Geomatique/Proj
)
##Label##
gva_map_ltae_tuned <- sits_label_classification(
  cube = gva_bayes_ltae_tuned,
  output_dir = "C:/Users/chafter1/Desktop/Stage Geomatique/Proj
  version = 'ltae-tuned'
)
```

```r
##Uncertainty##
uncert_ltae_tuned <- sits_uncertainty(
  cube = gva_probs_ltae_tuned,
  type = 'entropy',
  multicores = 24,
  memsize = 72,
  output_dir = "C:/Users/chafter1/Desktop/Stage Geomatique/Projet_Geneve/SITS
  version = 'ltae_v1_tuned'
)

##Accuracy##
#retrieve a list of samples described by a CSV file
samples_acc_LU04_18<- read.csv("C:/Users/chafter1/Desktop/Stage Geomatique/Pr
area_acc_ltae_tuned <- sits_accuracy(gva_map_ltae_tuned,
                                     validation = samples_acc_LU04_18)
```