



**UNIVERSITY OF PAVIA
FACULTY OF ENGINEERING**

DEPARTMENT OF ELECTRICAL, COMPUTER
AND BIOMEDICAL ENGINEERING

MASTER'S DEGREE IN COMPUTER ENGINEERING

MASTER THESIS

**Detection of manure application on crop fields
leveraging satellite data and Machine Learning**

**Rilevazione delle operazioni di concimazione tramite
dati satellitari e tecniche di *Machine Learning***

Candidate: Francesco Amato

Supervisor: Prof. Fabio Dell'Acqua

Co-supervisor: David Marzi, Ph.D.

Academic Year 2022/2023

Abstract

Detecting application of manure on crop fields is crucial for remotely assessing the correct management of crops; this is important to various goals such as maintaining soil fertility, productivity, environmental compliance, and - in the EU - also for verifying farmers' observance with the nitrates directive.

In the framework outlined above, this thesis research aims at developing an automated, Machine Learning (ML) based, method for detecting manure application leveraging Earth Observation (EO) satellite data.

Time series of spectral indexes (radar, optical and thermal) have been extracted from specific regions of interest (ROIs), located in Spain and Italy, using different EO satellites; a Python library, made available for public use, has been developed to efficiently accomplish this purpose.

After that, the spectral indexes most impacted by manure application have been identified and selected as input to different ML models, which have been compared - especially for what regards their generalization capabilities.

Tests have been conducted over both ROIs, including mixed cases where training was carried out on fields from one country and classification on fields located in another (or same).

In short, results suggest that the spectral signature of manure application is homogeneous within fields located in the same country, and that combining optical and thermal data allows achieving the best classification performances. Radar data, instead, provides no significant contribution to system performances.

The proposed method provides a valuable foundation toward development of a tool to monitor manure application in crop fields and ensure compliance with environmental regulations.

Sommario

L'individuazione dell'applicazione di concime sui campi coltivati è fondamentale per valutare a distanza la corretta gestione delle colture; ciò è importante per vari obiettivi quali il mantenimento della fertilità del suolo, la produttività, la conformità ambientale, e - nell'UE - anche per verificare il rispetto della direttiva sui nitrati da parte degli agricoltori.

Nel quadro di cui sopra, questa tesi di ricerca mira a sviluppare un metodo automatizzato, basato sul *Machine Learning* (ML), per rilevare l'applicazione di concime sfruttando dati satellitari per l'osservazione della Terra.

Serie temporali di indici spettrali (radar, ottici e termici) sono state estratte da specifiche regioni di interesse, situate in Spagna e Italia, utilizzando diversi satelliti; una libreria Python, resa disponibile per uso pubblico, è stata sviluppata per realizzare efficacemente questo scopo.

In seguito, gli indici spettrali maggiormente influenzati dall'applicazione del concime sono stati identificati e selezionati come input per diversi modelli di ML, che sono stati successivamente confrontati - specialmente per quanto riguarda le loro capacità di generalizzazione.

I test sono stati condotti su entrambe le regioni di interesse, compresi casi misti in cui il *training* è stato effettuato su campi di un paese e la classificazione su campi situati in un altro (o lo stesso).

In breve, i risultati suggeriscono che la firma spettrale dell'applicazione del concime è omogenea all'interno di campi situati nello stesso paese e che la combinazione di dati ottici e termici consente di ottenere le migliori prestazioni di classificazione. I dati radar, invece, non forniscono alcun contributo significativo alle prestazioni del sistema.

Il metodo proposto fornisce una base preziosa per lo sviluppo di uno strumento per monitorare l'applicazione di concime nei campi coltivati e garantire il rispetto delle normative ambientali.

Contents

Abstract	1
Sommario	2
1 Introduction	5
2 Context and research objectives	7
3 Tools and data	10
3.1 Datasets	10
3.2 Tools	10
4 Method and experiments	14
4.1 Introduction	14
4.2 Obtaining JSON files	15
4.2.1 Spanish crop fields	15
4.2.2 Italian crop fields	16
4.3 Features	16
4.3.1 Features extraction	16
4.3.2 Features importance	23
4.3.3 Features correlation	24
4.4 Data processing	25
4.4.1 Original datasets modification	25
4.4.2 Balancing classes	26
4.4.3 Normalization methods	28
4.5 Machine Learning models	30
4.5.1 Features selection	31
4.5.2 Classifiers	32
4.5.3 Hyperparameters tuning	34
4.6 Performance evaluation	36
4.6.1 KFold Cross-Validation	36
4.6.2 Confusion matrix	37

4.6.3 Accuracy, Precision, Recall and F1-score	39
5 Results	41
6 Conclusions	51
List of Figures	53
List of Tables	54
Bibliography	55
Acknowledgements	60

1 Introduction

The project carried out for this thesis aims to be a significant and innovative contribution in the field of Earth Observation applied to agriculture. Specifically, the objective is to develop a model that can detect the application of manure on crop fields, using multiple time series relative to different satellite imagery indexes. It is the very first attempt to develop such a model, and it can be a help for identifying potential breaches concerning the European nitrates directive, but still improvements and suggestions are welcome.

The initial step has been, once identified a dataset containing information about crop fields and related manure application dates, to extract spectral indexes from satellites. To achieve this, Google Earth Engine (GEE) APIs have been exploited, allowing for the necessary data to be accessed and processed fast, at no cost and exploiting machine parallelism. Following this, a comprehensive analysis has been conducted, focusing on the spectral indexes that have been most influenced by manure application. This analytical phase has been characterized by a keen eye for detail and a relentless pursuit of accuracy, as the various indicators of manure application and their associated effects have been identified and later on used as input to different ML models. With this foundation in place, the pipeline progressed to the next phase, which involved comparing multiple ML models capable of detecting the timeframe between which manure has been applied. The model with the most favorable performance metrics has been identified as the best one, following a rigorous evaluation of its accuracy, reliability, and other key factors - like generalization capability.

In order to allow readers to easily navigate through this research, a brief description of the different chapters is now introduced.

Chapter 2 discusses the context and research objectives in more details, focusing on why it is crucial to build such a system, and summarizes other state-of-the-art research projects correlated with this thesis. The chapter also outlines the research gap and how this study aims to fill that gap.

Chapter 3 presents tools and data, including the geographical distribution of the considered crop fields, and discusses potential generalization issues. The chapter

also describes the specific datasets used in the research, highlighting their importance and relevance.

Chapter 4 introduces methods and experiments used to conduct the research, including data collection, cleaning, analysis, and interpretation. The chapter also provides a detailed overview on the research procedures used. It allows other researchers interested in this topic to replicate the study. Furthermore, it outlines different ML models considered in the study and the specific metrics used to measure the performances' goodness.

Chapter 5 shows and discusses the experimental obtained results, with the crop fields of interest. It also provides a comprehensive evaluation of the models, highlighting their strengths and weaknesses.

Finally, in **Chapter 6**, conclusions are pointed out, summarizing the entire work and the final obtained model, especially how to interpret it in a simple and explainable manner. The chapter also focuses on open research aspects, highlighting areas that require further research and improvement.

Overall, this research project is a significant contribution to the field of ML learning applied to agriculture, providing insights and knowledge that can help improve agricultural systems' sustainability.

2 Context and research objectives

Agriculture, the practice of cultivating land, producing food, fiber and other products necessary for human survival and development, is a crucial sector that sustains human life and contributes significantly to the global economy. In 2020, it was worth almost 3.8 billion Dollars [1] and it is estimated that the world's population will reach over 10 billion by 2060 [2]. As a consequence, the demand for food is expected to increase from the current levels. Thus, farmers are under tremendous pressure to increase crop yields while minimizing the impact on the environment.

The use of manure as fertilizer is a common practice in agriculture, but also the creation of new hectares of crop fields, which is one of the main root causes of deforestation [3, 4].

The benefits of using fertilizers in agriculture have been discovered empirically over centuries, including in ancient and medieval times [5]. Now, instead, there are a lot of studies proving that the application of exogenous organic matter (EOM), such as minerals compost, wood ash and/or manure on crop fields allow enhancing the productivity of farms. It improves the levels of available plant nutrients and/or the chemical and physical properties of soil [6].

However, the overuse of fertilizers can lead to soil degradation, water pollution, and greenhouse gas emissions causing serious repercussions on the environment [7]. Although nitrogen is essential for the growth of plants and crops, excessive amounts can have negative consequences on both humans and the environment. In Europe, one of the leading causes of water pollution is the high concentration of nitrogen from agricultural sources. This is because nitrates and organic nitrogen compounds found in fertilizers and manure can easily infiltrate groundwater through leaching and reach surface water through outflow from agricultural fields [8].

As a result of this pollution, a high level of nitrates can make water unsuitable for drinking. In rivers, lakes, and marine waters, excessive nitrogen and other nutrients, particularly phosphorus, stimulate algae growth. While algae at moderate levels serve as food for aquatic organisms, including fish, excessive nutrient concentrations can cause algae to grow excessively, leading to eutrophication [9].

This phenomenon affects the natural ecosystem and can cause oxygen depletion

in the water, with negative consequences for biodiversity, fisheries, and recreation. Pure and clean water is vital for human health and natural ecosystems, and it is important to take steps to mitigate the negative impacts of excess nitrogen in the environment.

For that particular reason, two directives in the European Union [10, 11], termed the *nitrates directives*, have been emended in order to control the use of fertilizers. To address the issue of excess nitrogen from agricultural sources, a potential solution that has been proposed is the implementation of closed periods for organic fertilizer application in nitrate vulnerable zones [12]. These closed periods would prohibit farmers from spreading organic fertilizers during specific times. The exact duration of closed periods varies depending on factors such as the terrain and seasonal characteristics of each Member State, as well as the type of fertilizer used.

Although EOMs are commonly applied in agriculture, there is a lack of documentation regarding the frequency of amendment applications and, even worse, their spatial allocation. The current practice of monitoring EOM amendments involves surveying farmers, which is a time-consuming and costly process. However, the advent of Earth observation through satellites data and Machine Learning has opened up new opportunities [13–17], including identifying manure application on crop fields more efficiently and accurately.

In this framework, a key research paper [18] reported an experiment that has been conducted in order to obtain some spectral indexes, extracted from specific optical bands through satellites, that can be used to detect manure application for some crop fields, located in France. Our research aims at - among other things - validating whether the same features are correlated to manure application in another context, but also to measure the correlation on other optical, radar and thermal indexes (leveraging Sentinel-1, Sentinel-2 and Landsat-8 satellites). Furthermore, the objective is to provide a robust solution, and gain a deeper understanding of the importance of different indexes for our classification task.

Whereas, another research [19] created a method in order to recognize manure spreading (pixelwise detection) between two consequent Sentinel-2 acquisitions, knowing a priori that manure was applied within the two considered acquisitions, obtaining an *F-1* score in test around 89%. The potential problem of their approach

is that it is not always possible to know a priori the dates on which fertilization was carried out, and - as far as the authors' knowledge - there are no methods - other than the one proposed with this research - for solving it, analyzing multiple time series. In addition, they used too many optical indexes, which is the root cause of some issues (e.g., difficulty in interpretation, collinearity and multicollinearity, increased complexity and computation). Our objective is also to possibly decrease the number of features involved in the models.

Although our proposed research method results in loss of information about the spatial spreading of manure on crop fields, it should not necessarily be seen as a disadvantage. In fact, it enables to create light-weighted datasets that occupy less memory, speeding up data retrieval, analysis, and processing, among other benefits. Our proposed solution, combined with the one deployed in the previously mentioned paper [19], can create a potential framework to **(i)** detect, analyzing multiple time series, when a crop field has been manured and **(ii)** to obtain the spatial spreading of manure for that crop, within the period of interest.

Different ML models have been trained, tested and compared in terms of multiple performance metrics [20], including their generalization capabilities. The final model can be a useful instrument to automatically monitor agricultural activities, more specifically the application of manure, leveraging multiple satellites. Thus, it can detect potential violations of the nitrates directives in a fast and accurate manner, although further improvements are needed.

3 Tools and data

This section contains a description of all datasets and tools that have been used to conduct this research.

3.1 Datasets

The datasets that have been used are different; one comes from "*Satellite imagery dataset of manure application on pasture fields*" (license CC BY 4.0) [21]. From that dataset, which contains details about crop fields located in Spain (Figure 1), a JSON file has been obtained, in order to facilitate the conduct of research and to reduce the space allocated in memory (instead of storing plain spectral images). The mean extension of those crop fields is of something more than 10 thousand square meters.

Whereas, for what concerns fields located in Italy (Figure 1), three JSON files have been purposely created - to measure generalization capabilities, and perform mixed cases where training was carried out on fields from one country and testing on fields located in the other (or in the same). Manure dates have been obtained by on-site investigations - except the fields appertaining to the DUSAf dataset (whose translation stands for "*Destination of the Use of Agricultural and Forestry Soils*") [22], where manure dates are not available - while the set of coordinates that form the fields polygons have been obtained with Google Earth Engine. The mean extension of these fields, instead, is of almost 32 thousand square meters.

In order to understand more details about the fields of interest, other images have been added (Figures 2, 3).

However, all the datasets used in this research can be accessed on a suited GitHub repository [23], more precisely inside the *Datasets* folder (license CC BY-NC 4.0).

3.2 Tools

The tools that have been used to conduct this experiment are mainly Google Earth Engine and Jupyter Notebook.

Google Earth Engine is a cloud-based platform for analyzing and visualizing geospatial data. It provides access to a vast collection of satellite imagery and other geospatial data sets, along with the tools and infrastructure for processing and

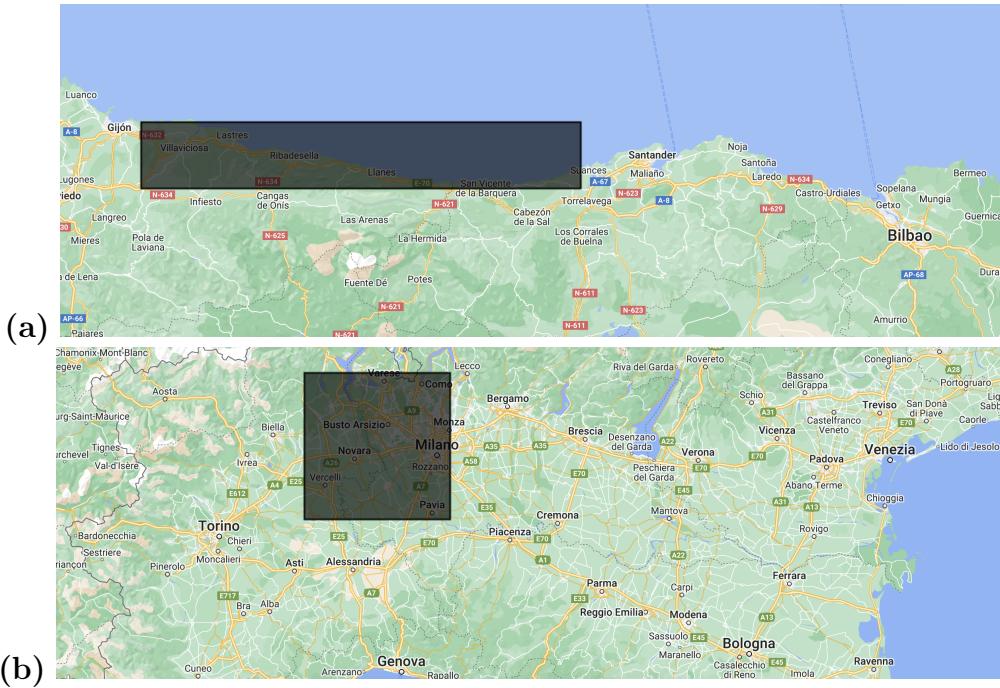


Figure 1: Smallest rectangles that comprehend all analyzed fields. (a) Plots located in Spain. (b) Plots located in Italy.

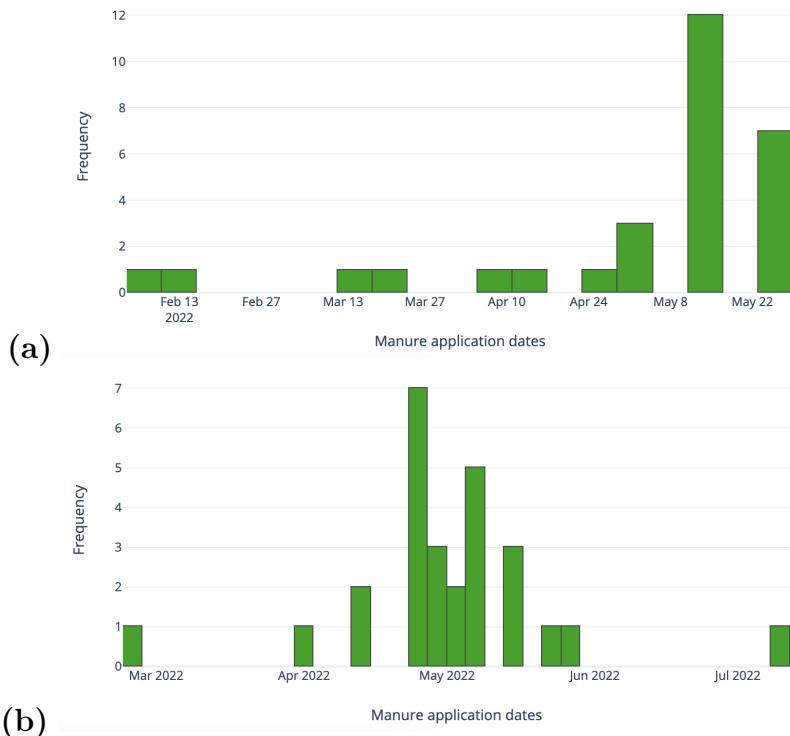


Figure 2: Frequency distribution of manure application. (a) Plots located in Spain. (b) Plots located in Italy.

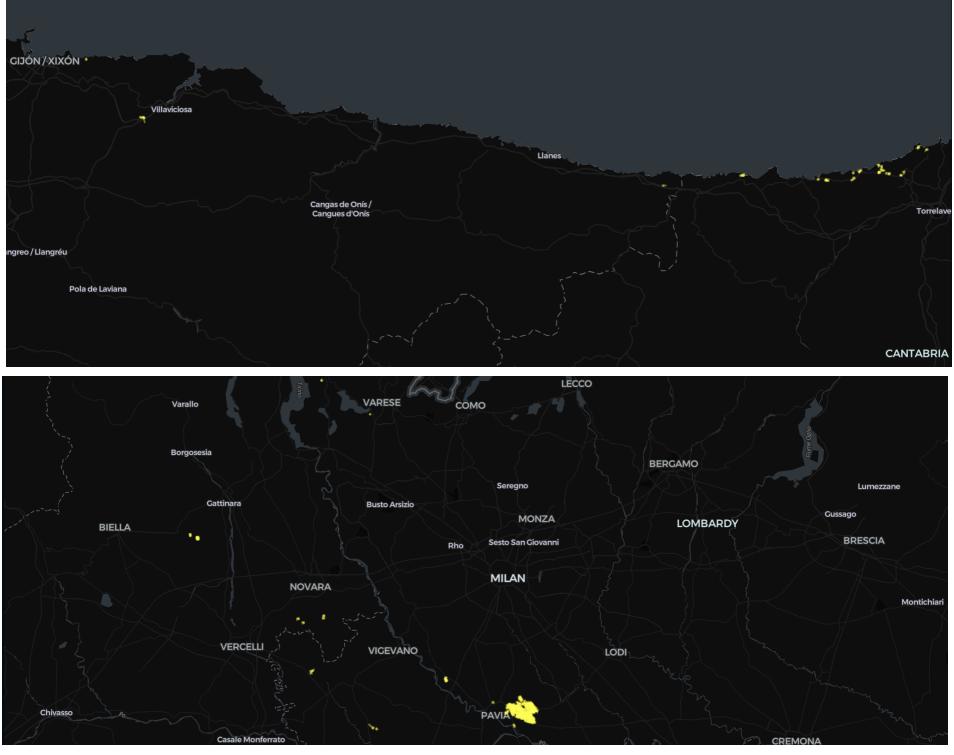


Figure 3: Polygons (in yellow) representing all fields. (a) Plots located in Spain. (b) Plots located in Italy.

analyzing that data on a large scale. Google Earth Engine can be used for a variety of applications, including mapping and monitoring changes in land use, tracking the movement of wildlife, measuring the health of ecosystems, and assessing the impacts of climate change.

On the other hand, the use of Jupyter Notebook provides an interactive and powerful environment, where users can write and execute Python code (which can be executed independently or in sequence) and integrate Markdown/Latex to improve the file readability. It makes it possible - among the other things - to explore data, test different hypotheses, build different ML models, visualize data and share results, in an interactive way.

In this specific project, Google Earth Engine and its API have been used to retrieve spectral data about the crop fields of interest and also to obtain a few data visualization images. Whereas, Jupyter Notebook has been used to perform data analysis, build models and test generalization. Furthermore, it has been used to describe all the steps that were followed during our research, so that all people interested in this study can understand, reproduce or improve the obtained results

(e.g. adding new spectral features, testing generalization on other fields of interest, training the model with more data, etc.).

Using the Google Earth Engine Python APIs comes with its set of challenges, particularly when it comes to retrieving data efficiently and exploiting machine parallelism for multiple I/O requests. GEE is a vast geospatial platform with an extensive collection of satellite images, and fetching data from such a large repository can be time-consuming, requiring careful optimization to reduce processing times. Preprocessing satellite image collections is also a daunting task, involving the application of appropriate filters to ensure data accuracy and quality. This includes handling cloud cover, atmospheric interference, and other factors that may impact the image's usability. Additionally, selecting the right satellites for a specific project is crucial, as each satellite has its own sensor characteristics, resolution, and temporal coverage. Thus, a thorough understanding of the available satellites and their capabilities is necessary to make informed choices that align with the research objectives.

4 Method and experiments

4.1 Introduction

The comprehensive methodology employed in this research project can be easily visualized by referring to Figure 4, which illustrates the step-by-step followed process from the upper left corner, akin to a bouncing ball, where the investigation starts.

The first crucial research step involved to gather details about the specific dates in which each crop field was manured (except the fields appertaining to the already cited DUSAf dataset). This information, together with others, has then been used to create distinct JSON files, each containing a set of fields characterized by their names, manure dates, and a set of coordinates that formed the corresponding polygon. While one of these files pertained exclusively to crop fields located in the northern part of Spain and was the subject of a previously published study [19], the others have been utilized to validate the models' goodness and to check for any potential generalization issues in the methodology (recall the initially explained "mixed cases"), using crops located in the northern part of Italy.

After the initial data collection phase, the next crucial step in the methodology involved obtaining time series data for each crop field using Google Earth Engine APIs, which facilitated the tracking of the evolution of different optical, radar and thermal spectral indexes.

These data have then been analyzed through the application of features selection techniques, followed by the training and testing of multiple ML models to determine the most accurate, reliable and generalizable approach.

Each step of this comprehensive flowchart will be described in more detail in the following sections, providing a clearer overview of the research methodology employed. It is also worth noting that the code, notebooks, datasets, and materials used or produced during the course of this research activity are publicly available in the same GitHub repository [23], enabling others to replicate and build on the published results.

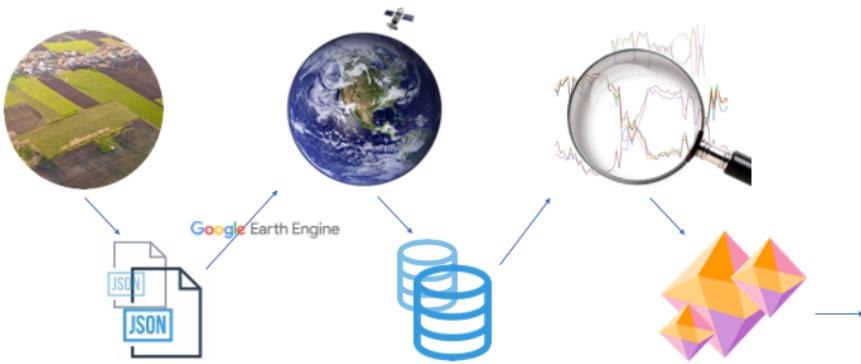


Figure 4: Flowchart of the overall approach.

4.2 Obtaining JSON files

This particular section provides a detailed description of the specific procedure that has been followed to obtain the essential files containing crucial information about the crop fields. It is important to note that this initial step in the research process was absolutely crucial, and without it, it couldn't have been possible to proceed with investigation (all subsequent analysis and conclusions drawn).

4.2.1 Spanish crop fields

To analyze the crop fields located in Spain, the dataset "*Satellite imagery dataset of manure application on pasture fields*" has been used [21]. It includes valuable information about each crop field, such as its shape, size in square meters, and center coordinates. Those data allowed to generate a structured and memory-efficient JSON file, which has been essential for the subsequent analysis phases.

Using the Google Earth Engine online IDE, it has been possible to process and collapse the data from the original dataset to a unique JSON file. This file contains all the necessary information about each crop field in a more concise format.

To visually represent the geographical location of those fields, Figure 3 has been used.

4.2.2 Italian crop fields

For the Italian plots, on-site investigations - except the ones regarding the DUSAf dataset, where this has not been possible - have been carried out to check (or ask) whether and when they have been manured.

The dates of manure application have been recorded and different JSON files containing the crop fields details have been obtained using Google Earth Engine (the reason why different files will be clearer during the results' explanation). Their structure is the same as of the one described in the previous subsection (Spanish).

Figure 3 illustrates the Italian crop fields' geographical location.

4.3 Features

The objective of this section is to describe the extraction of indexes for each instance that a satellite passed over a specific crop of interest within a given time frame. Additionally, an analytical approach was used to identify the spectral indexes that are most correlated with manure application, which was validated using a statistical test to prove the significance of their importance across different repeated manure applications and crop fields.

4.3.1 Features extraction

This section discusses the extraction of spectral indexes, both optical (Table 2), radar (Table 3) and thermal. In order to do this, a Python library has been developed and made publicly available in a PyPI repository named **ee-satellites** (license MIT) [24].

The very first question that you might ask is why Sentinel and Landsat satellites have been chosen, even if there are plenty of others available (e.g. RapidEye [25], SkySat [26], DOVE [27] and CloudSat [28] - just to cite a few of them)? There are several reasons why those satellites have been chosen, the main ones are:

- *Open Data Policy*: One of the key features of the Copernicus and Landsat programs is its commitment to providing free and open access to data. The data collected by those satellites are made available to the public, researchers, and

other stakeholders at no cost. This open data policy promotes collaboration and enables a wide range of users to benefit from the data.

- *Customized Design*: They are equipped with a variety of sensors and instruments that enable comprehensive EO, including radar, optical and thermal imaging capabilities. Their sensors provide a lot of spectral bands, such that they can provide valuable data on various aspects of the Earth's environment.
- *Global Coverage*: Those constellations ensure global coverage, meaning that it can observe the entire Earth on a regular basis. This is essential for monitoring changes and phenomena that occur on a global scale, such as climate patterns, deforestation, pollution, and natural disasters.
- *Continuity and Long-Term Monitoring*: They are designed to operate over the long term, providing consistent and continuous monitoring of the Earth's environment. This continuity is crucial for detecting and analyzing long-term trends, tracking changes over time, and assessing the impact of human activities on the planet. By having a constellation of satellites, there is redundancy and the ability to ensure continuous data collection even if one satellite fails or needs maintenance.
- *Synergy with Existing Systems*: Those satellites can be easily accessed by the usage of other well known systems like Google Earth Engine, Planet and so on. They also collaborate and share data with other satellite constellations and ground-based monitoring networks, so they aim at creating a comprehensive and integrated approach to EO. This synergy increases the accuracy and reliability of the data collected and provides a broader perspective on global environmental challenges.

Now let's discuss, instead, about the implementation and details of the created library. It provides an easy-to-use, comprehensive, and flexible way to work with satellites data from the Sentinel-1, Sentinel-2 and Landsat-8 satellites (Figure 5). Its key advantages include: a well-documented API, support for some of the most commonly used satellites, an open-source codebase and regular updates. In addition to the already mentioned advantages, the implemented code relies on Google Earth

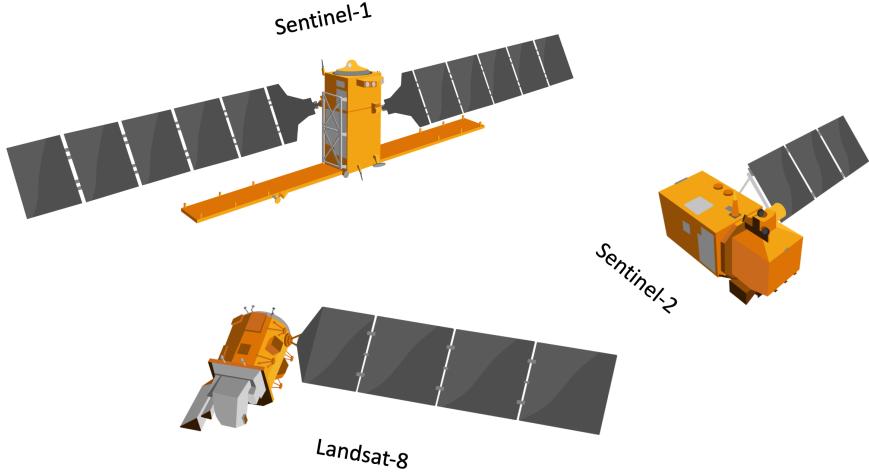


Figure 5: Satellites representation.

Engine (GEE) APIs (used to access satellites data and perform some tasks such as cloud masking, image compositing, and time series selection) and exploits multi-threading (designed to work efficiently, by making a lot of parallel I/O requests to GEE).

These advantages make it an excellent tool for anyone working with satellites data, since it allows generating datasets that can be both easily used for data analysis and efficiently integrated with well-known ML libraries, to deploy models.

The most important function is the `get_features` one, which allows getting from an input (Pandas) DataFrame composed of fields information, an output DataFrame that contains for each time selected satellites (Sentinel-1, Sentinel-2 or Landsat-8) passed over the specified fields, within a given time period, all the mean values of some of the most used indexes (optical, radar or thermal).

The input DataFrame, named `fields_df` (for example), should be structured as shown in Table 1.

Then, once authenticated with the Google Earth Engine APIs, you can call the function `get_features` from the `ee_satellites` library to retrieve the desired data.

The output DataFrame contains one row for each time the selected satellite passed over the specified crop fields during the desired time period. Each column represents the mean band/index value for the corresponding polygon (for the single acquisition). In order to improve the values' reliability, not only on tile cloud filtering have been done, but also in-field pixelwise cloud and shadow filtering (for Sentinel-2

Table 1: Input DataFrame structure: example.

crop_field_name	polygon_coordinates	manure_dates
P-BLD	[(-4.202723286616649, 43.39683579015289), (-4....	[2022-05-26]
P-BLLT1	[(-4.085622203603083, 43.429605845026266), (-4....	[2022-05-16]
P-BLLT2	[(-4.084840437376829, 43.430826294936246), (-4....	[2022-05-26]
...
P-VG2	[(-5.480254884662825, 43.4687861587232), (-5.4...	[2022-04-13]
P-VLDMR	[(-4.157038444925938, 43.40720711284701), (-4....	[2022-02-07]
P-VNS	[(-4.151167740565273, 43.40535762666503), (-4....	[2022-04-23]

acquired images). Unfortunately the spatial information is lost, so it is not possible to obtain pixel-level information, but this is not the purpose of the research.

Few notes are needed in order to describe what are the function parameters:

- *fields_df*: A DataFrame containing crop field names and closed polygons coordinates.
- *start_date*: The start date of the date range to filter the collection by.
- *end_date*: The end date of the date range to filter the collection by.
- *satellite*: The type of satellite to use ('sentinel-1', 'sentinel-2' or 'landsat-8').
- *filters_params*: The list of parameters values to be used for filters to extracting images collections.
 - For *Sentinel-1*: first parameter in the list represents the value of the 'orbitProperties_pass' filter ('ASCENDING' or 'DESCENDING')
 - For *Sentinel-2*: first parameter in the list represents the value of the 'CLOUDY_PIXEL_PERCENTAGE' filter ('LTE' - values in range [0, 100])
- *fields_threads*: The number of threads to dedicate to parallelization of GEE API requests over the fields level, the remaining ones are used to apply parallelization over dates level. The value of this parameter should be high (with respect to the overall number of threads exploitable) if you have a lot of crop fields but a little time-span to consider, whereas if you have fewer fields but a

bigger time-span, you should decrease this parameter. Finally, if you have a lot of fields with a lot of dates to process, it is optimal to consider half of the overall number of threads available on the local machine used to perform the I/O requests.

Multithreading has been chosen instead of multiprocessing because the problem is of making a lot of parallel I/O requests to Google Earth Engine, and not a lot of local computations. The bottleneck in this case is the network, so the extraction depends on many factors, but one of the most relevant ones is the traffic that Google Earth Engine has when carrying out requests for extracting spectral indexes.

Table 2: Optical indexes.

Abbr.	Name	Equation
NDVI	Normalized Difference Vegetation Index [29]	$\frac{B8 - B4}{B8 + B4}$
NSNDVI	NIR-SWIR Normalized Difference Vegetation Index [30]	$\frac{B11 - B7}{B11 + B7}$
GNDVI	Green Normalized Difference Vegetation Index [29]	$\frac{B8 - B3}{B8 + B3}$
RENDVI1	Red Edge Normalized Difference Vegetation Index 1 [29]	$\frac{B5 - B4}{B5 + B4}$
RENDVI2	Red Edge Normalized Difference Vegetation Index 2 [29]	$\frac{B6 - B4}{B6 + B4}$
RENDVI3	Red Edge Normalized Difference Vegetation Index 3 [29]	$\frac{B7 - B4}{B7 + B4}$
GRNDVI	Green-Red Normalized Difference Vegetation Index [30]	$\frac{B8 - (B3 + B4)}{B8 + (B3 + B4)}$
GBNDVI	Green-Red Normalized Difference Vegetation Index [30]	$\frac{B8 - (B3 + B2)}{B8 + (B3 + B2)}$
SAVI	Soil Adjusted Vegetation Index [29]	$\frac{B8 - B4}{B8 + B4 + 0.428} * 1.428$
OSAVI	Optimized Soil Adjusted Vegetation Index [29]	$(1 + 0.16) * \frac{B8 - B4}{B8 + B4 + 0.16}$
MSAVI	Modified Soil Adjusted Vegetation Index [29]	$\frac{2.0 * B8 + 1.0 - \sqrt{(2.0 * B8 + 1.0)^2 - 8.0(B8 - B4)}}{2.0}$
TSAVI	Transformed Soil Adjusted Vegetation Index [29]	$\frac{0.421 * (B8 - 0.421 * B4 - 0.824)}{B4 + 0.421 * (B8 - 0.824) + 0.114 * (1.0 + 0.421^2)}$
ATSAVI	Adjusted Transformed Soil Adjusted Vegetation Index [30]	$\frac{1.22 * (B8 - 1.22 * B4 - 0.03)}{B4 + 1.22 * B8 - 1.22 * 0.03 + 0.08 * (1.0 + 1.22^2)}$
RVI	Ratio Vegetation Index [29]	$\frac{B8}{B4}$
DVI	Difference Vegetation Index [29]	$B8 - B4$
CVI	Chlorophyll Vegetation Index [30]	$\frac{B8 * B4}{B3^2}$
CTVI	Corrected Transformed Vegetation Index [30]	$\frac{((B4 - B3)/(B4 + B3)) + 0.5}{ B4 - B3 + 0.5 * \sqrt{\frac{B4 - B3}{B4 + B3} + 0.5}}$
AVI	Advanced Vegetation Index [30]	$(B8 * (1 - B4) * (B8 - B4))^{1/3}$
ARVI1	Atmospherically Resistant Vegetation Index 1 [29]	$\frac{B8A - B4 - 0.069 * (B4 - B2)}{B8A + B4 - 0.069 * (B4 - B2)}$
ARVI2	Atmospherically Resistant Vegetation Index 2 [29]	$-0.18 + 1.17 * \frac{B8 - B4}{B8 + B4}$
EVI1	Enhanced Vegetation Index 1 [30]	$\frac{2.5 * (B8 - B4)}{B8 + 6 * B4 - 7.5 * B2 + 1}$

Continued on next page

Table 2 – *Continued from previous page*

Abbr.	Name	Equation
EVI2	Enhanced Vegetation Index 2 [30]	$2.4 * \frac{B8 - B4}{B8 + B4 + 1}$
EVI3	Enhanced Vegetation Index 3 [30]	$2.5 * \frac{B8 - B4}{B8 + 2.4 * B4 + 1}$
WDRVI	Wide Dynamic Range Vegetation Index [29]	$\frac{0.1 * B8 - B4}{0.1 * B8 + B4}$
MTVI1	Modified Triangular Vegetation Index 1 [31]	$1.2 * (1.2 * (B8 - B3) - 2.5 * (B4 - B3))$
MTVI2	Modified Triangular Vegetation Index 2 [31]	$1.5 * \frac{1.2 * (B8 - B3) - 2.5 * (B8 - B3)}{\sqrt{(2 * B8 + 1)^2 - (6 * B8 - 5 * \sqrt{B4})} - 0.5}$
EOMI1	Exogenous Organic Matter Index 1 [18]	$\frac{B11 - B8A}{B11 + B8A}$
EOMI2	Exogenous Organic Matter Index 2 [18]	$\frac{B12 - B4}{B12 + B4}$
EOMI3	Exogenous Organic Matter Index 3 [18]	$\frac{(B11 - B8A) + (B12 - B4)}{B11 + B8A + B12 + B4}$
EOMI4	Exogenous Organic Matter Index 4 [18]	$\frac{B11 - B4}{B11 + B4}$
NBR	Normalized Burned Ratio Index [30]	$\frac{B8 - B12}{B8 + B12}$
NBR2	Normalized Burned Ratio Index 2 [18]	$\frac{B11 - B12}{B11 + B12}$
CI1	Chlorophyll Index 1 [30]	$\frac{B8}{B5} - 1$
CI2	Chlorophyll Index 2 [30]	$\frac{B8}{B6} - 1$
CI3	Chlorophyll Index 3 [30]	$\frac{B8}{B7} - 1$
GCI	Green Coverage Index [30]	$\frac{B9}{B3} - 1$
SCI	Soil Composition Index [30]	$\frac{B11 - B8}{B11 + B8}$
NDRE1	Normalized Difference Red Edge Index 1 [29]	$\frac{B8 - B5}{B8 + B5}$
NDRE2	Normalized Difference Red Edge Index 2 [29]	$\frac{B8 - B6}{B8 + B6}$
NDRE3	Normalized Difference Red Edge Index 3 [29]	$\frac{B8 - B7}{B8 + B7}$
CARI1	Chlorophyll Absorption Ratio Index 1 [30]	$\frac{B5 - B3}{B4} * \frac{ \frac{B5 - B3}{150} * 670 + B4 + B3 - \frac{B5 - B3}{150} * 550 }{\sqrt{\frac{B5 - B3}{150^2} + 1}}$
CARI2	Chlorophyll Absorption Ratio Index 2 [30]	$\frac{ \frac{B5 - B3}{150 * B4} + B4 + B3 - 0.496 * B3 }{\sqrt{(0.496^2 + 1) * \frac{B5}{B4}}}$
MCARI	Modified Chlorophyll Absorption in Reflectance Index [32]	$((B5 - B4) - 0.2 * (B5 - B3)) * \frac{B5}{B4}$
MCARI1	Modified Chlorophyll Absorption in Reflectance Index 1 [30]	$1.2 * (2.5 * (B8 - B4) - 1.3 * (B8 - B3))$
MCARI2	Modified Chlorophyll Absorption in Reflectance Index 2 [31]	$1.5 * \frac{2.5 * (B8 - B4) - 1.3 * (B8 - B3)}{\sqrt{(2 * B8 + 1)^2 - (6 * B8 - 5 * \sqrt{B4})} - 0.5}$
BSI	Bare Soil Index [30]	$\frac{B11 + B4 + \frac{B8 + B2}{B11 + B4} + B8 + B2}{2 * B3 - B4 - B2}$
GLI	Green Leaf Index [30]	$\frac{2 * B3 + B4 + B2}{2 * B3 + B4 + B2}$
ALTERATION	Alteration Index [30]	$\frac{B11}{B12}$
SDI	SWIR Difference Index [33]	$B8 - B12$

As previously mentioned, this library is public, so anyone interested in integrating it with his/her own research is welcome to do so. As you can imagine, it can be used for a variety of different purposes, allowing also to avoid storing satellite imagery in

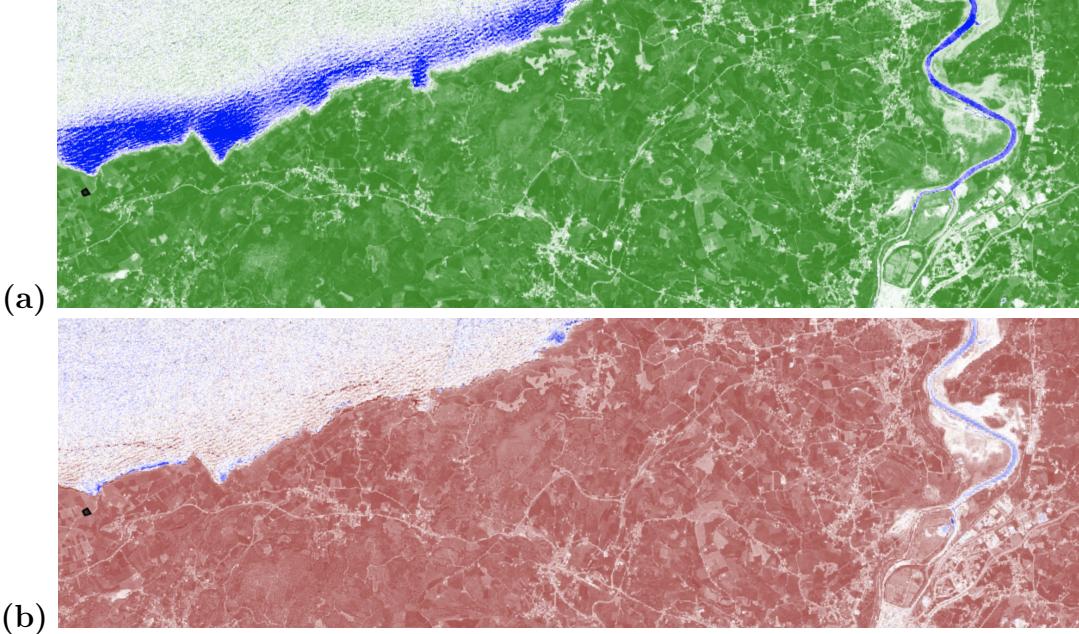


Figure 6: Visualization of (a) NDVI (the greener, the higher the value, the more the blue, the lower). (b) EOMI2 (the more the brown, the higher the value, the more the blue, the lower). Those indexes have been calculated near the plot represented in black, on the left side (namely P-BLD).

memory, which permits to perform investigations also on a local machine. Therefore, there is not the effort of performing the analysis on the cloud, which means lower costs, better security, higher speed and the possibility to work offline.

In order to understand how the feature extraction procedure works, an example is now explained (Figure 6), considering just optical indexes (Sentinel-2).

For each time - typically a day, it depends on the revisit frequency - a selected satellite, appertaining to the specified constellation, passed over the area of interest (in this case is the plot P-BLD, shown in black on the left), its main spectral bands mean values (13, for Sentinel-2) are calculated only for that region (through the GEE *reduceRegion*).

Next, the other spectral indexes (e.g. Table 2) are calculated using the values of the already extracted mean bands. This causes the fact that the obtained spectral indexes mean values are an acceptable approximation of their true values (it is not a problem of precision agriculture or the like), at the cost of reducing the number of requests to GEE and so to gain a significant speedup in the feature extraction time.

Table 3: Radar indexes.

Abbr.	Name	Equation
AVE	Average [34]	$\frac{VV * VH}{2}$
DIF	Difference [34]	$VV - VH$
RAT1	Ratio 1 [34]	$\frac{VV}{VH}$
RAT2	Ratio 2 [34]	$\frac{VH}{VV}$
NDI	Normalized Difference Index [34]	$\frac{VV - VH}{VV + VH}$
RVI	Radar Vegetation Index [35]	$\frac{VH * 4}{VV + VH}$

4.3.2 Features importance

An a priori study - relative to the discovery of the indexes that are changing most significantly when manure have been applied on crops - is needed, that is to consider the trend of individual features throughout a year and then see if during the manure date that feature has changed significantly (the objective is to understand what is the spectral response of the manure application itself). This step is of fundamental importance since it allows having a general understanding of the features that should be used in the models that will be later on developed.

Feature selection should be done on a portion of the entire dataset. This is because selecting features on the entire dataset can lead to over-fitting, which occurs when the model becomes too complex and is fitted too closely to the training data, causing it to perform poorly on new, unseen data.

For each crop field, the relevant formula is:

$$feat_imp = \frac{|feat_val_{imm_after_manure} - feat_val_{imm_before_manure}|}{\max |daily_feat_diff_manure|} \quad (1)$$

Where *feat* is a shorter way to say feature, *imp* importance, *val* value, *imm* immediately, *diff* difference.

The feature importance (called *feat_imp*) has been therefore defined as: (i) the absolute difference of the feature's value during the acquisition immediately after manure application, and the mean value of the feature *N* (2) acquisitions before

manure has been applied, (ii) divided by the maximum feature's value difference between two consequent acquisitions, when manure has not been applied (in absolute value). An important paper [18] shows that the effect of manure application on certain features decreases over time, that's why the acquisition immediately after manure application has been considered.

Leaving the feature importance as it has been formulated gives no assurance of statistical significance, which is yet another important aspect to consider. In order to cover this aspect, a one-sample t -test [36] has been conducted in order to determine how significant is the impact of manure application on a single feature. In a nutshell, it has been defined as how many times a significant change (high value of the feature importance) has been observed across multiple fields (null hypothesis being the expected group mean of 0 and the alternative hypothesis being that it is greater than 0). If the obtained p-value is below a prior established threshold (0.05 is a typically accepted value), it suggests that the observed change is unlikely to occur by chance alone, providing evidence to reject the null hypothesis.

4.3.3 Features correlation

Once identified the features most impacted by manure application, it is important to consider the correlation between the considered spectral features. This, in order to have a deeper understanding on which indexes should be considered in the final model.

Correlation, in the context of data analysis, refers to a statistical measure that quantifies the strength and direction of the relationship between two or more variables. It serves as a valuable tool for understanding how changes in one variable are associated with changes in another.

In the specific case of studying the impact of manure application on spectral features (such as optical, radar or thermal measurements), identifying the features that are most affected is just the initial step. To gain a more comprehensive understanding, it becomes essential to explore the correlation between these selected features.

This analysis enables us to identify which indexes or measurements are highly related to each other and may provide redundant or overlapping information.

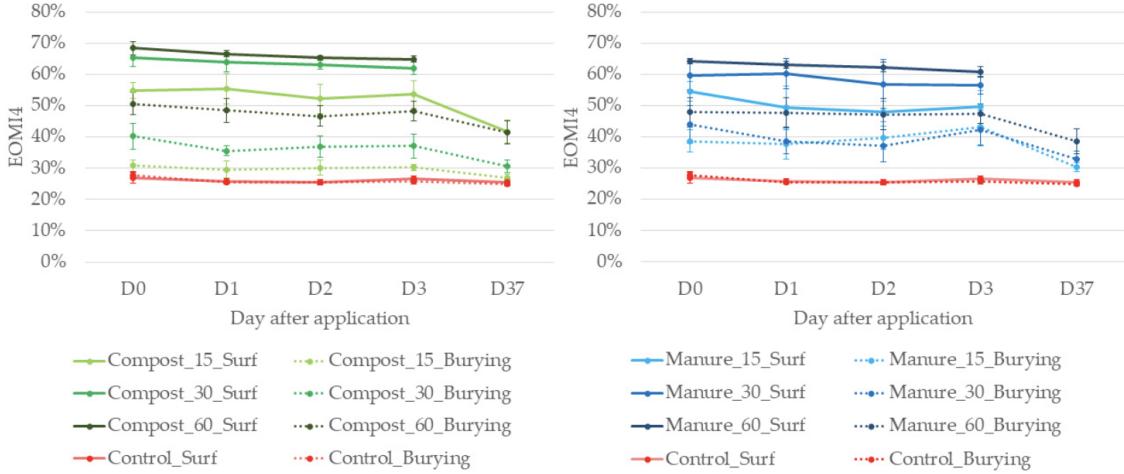


Figure 7: Temporal change of EOMI4 values for all treatments. (**Left**) Green waste compost; (**Right**) sheep manure treatments [18].

It is worth noting that although the data used as input for the final model may differ from the initially extracted spectral features, considering correlation remains crucial. The insights gained from the correlation analysis help guide the decision-making process regarding which features should be incorporated into the model, ensuring its accuracy and effectiveness in capturing the underlying relationships between variables.

4.4 Data processing

Before discussing the different models that have been considered in order to solve the classification problem at hand, it is important to focus on the data passed as an input to those models. The very first step is to modify purposely the original datasets (the one having mean spectral bands for each time a satellite crossed that field) to have better data, which can help to achieve final better results. For the very same reason, two other methods have also been considered, which are: class balancing and feature normalization.

4.4.1 Original datasets modification

As already mentioned in a previous section, the effect of manure application on the different spectral indexes decreases over time (Figure 7), that's why it is important to consider the satellite acquisition right after manure application.

	crop_field_name	consequent_s2_acquisitions	B1	B2	B3	B4	B5	B6	B7	B8	CARI2	MCARI	MCARI1	MCARI2	BSI	GLI	ALTERATION	SDI	manure_dates	y
0	P-BLD	[2022-01-06, 2022-01-16]	11.525510	-8.440476	-21.562925	-9.930272	6.205782	124.039116	220.292517	193.943878	... -14.821130	214.161369	275.91837	-0.001536	262.245588	0.004881	-0.003841	151.874150	[2022-05-26]	0.0
1	P-BLD	[2022-01-16, 2022-01-26]	1078.435374	1062.938776	1057.346939	1060.000000	1126.557823	1147.452381	1167.767007	1123.741497	... 3564.075800	-2129.186864	87.648980	-0.175677	4166.625746	-0.391301	-0.629555	163.807823	[2022-05-26]	0.0
2	P-BLD	[2022-01-26, 2022-01-31]	2484.477891	2260.610544	1760.287415	1861.488395	1280.738095	-483.047619	-894.066327	-886.345238	... 2216.904547	-1006.267915	-4114.747959	-0.406936	3935.251609	-0.118245	-0.277985	-1866.045918	[2022-05-26]	0.0
3	P-BLD	[2022-01-31, 2022-02-05]	-2304.959184	-2102.022109	-1548.894558	-1711.838435	-1013.522109	1331.586735	1878.299320	1974.547619	... -1508.152782	1195.170656	5562.588367	0.420944	-1925.189004	0.121698	0.329274	2617.057570	[2022-05-26]	0.0
4	P-BLD	[2022-02-05, 2022-02-10]	1566.938776	1633.841837	1222.882653	1338.595238	744.566327	-1291.680272	-1764.062925	-1971.335034	... 1402.703367	-1072.556874	-4946.811224	-0.339269	371.829666	-0.104554	-0.321017	-2013.053271	[2022-05-26]	0.0
...	
821	P-VNS	[2022-09-08, 2022-10-08]	5606.143498	5374.345291	4764.394619	4737.530942	4444.085202	2348.694170	1702.821525	1609.651121	... 5029.973432	-4542.13280	-4462.239605	-0.604684	14214.068737	-0.097813	-0.469385	-1611.464574	[2022-04-23]	0.0
822	P-VNS	[2022-10-08, 2022-10-23]	-2735.709417	-2652.242152	-2654.146188	-2698.858296	-2656.408072	-2181.641256	-2108.233184	-1965.759641	... -3804.884046	63.383846	1125.412951	0.200635	-9480.670199	0.002424	0.019581	111.947085	[2022-04-23]	0.0
823	P-VNS	[2022-10-23, 2022-11-12]	-2516.098655	-2423.871749	-1739.669955	-1933.901345	-1126.258296	-380.855605	-1779.75785	-130.049327	... 3243.365785	262.395659	1280.547874	0.190679	-3448.417006	0.046757	0.294226	185.921973	[2022-04-23]	0.0
824	P-VNS	[2022-11-12, 2022-11-17]	-310.51121	-237.513004	-243.252018	-341.180269	-277.982960	26.653812	128.503139	97.081614	... -1977.622835	150.359500	783.865184	0.098095	-763.198817	0.022308	0.055699	403.480717	[2022-04-23]	0.0
825	P-VNS	[2022-11-17, 2022-12-17]	-178.017040	-156.201794	-109.430493	-235.243049	-166.998206	267.694170	347.0398565	416.705830	... -1449.713318	189.153790	1135.073973	0.093841	-647.427495	0.036957	0.064493	973.523767	[2022-04-23]	0.0

Figure 8: An example of the structure of the modified dataset.

After this phase, the resulting datasets consist of multiple rows, each corresponding to a single specific crop field and contains multiple pieces of information.

Each column (once fixed the row, hence considering a specific crop field) represents the difference between two consecutive satellite constellation acquisitions, for the considered spectral index.

The y column, which is a binary indicator variable, instead indicates whether any manure application date falls within the two consecutive acquisition dates being considered (Figure 8).

In a nutshell, it contains the spectral index variation between two consecutive sentinel acquisitions, together with another piece of information that expresses whether that crop has been manured between the two consecutive considered acquisitions.

4.4.2 Balancing classes

Dataset balancing techniques, such as over-sampling and under-sampling, are crucial for classification tasks to address the problem of class imbalance. Class imbalance occurs when the number of instances in one class significantly outweighs the number of instances in the others (which is exactly the case, as you can imagine there are fewer observations when crops are manured with respect to the other case), leading to biased model performance and inaccurate predictions.

Over-sampling involves increasing the number of instances in the minority class to match the number of instances in the majority class. This can be achieved by duplicating existing minority class instances or generating synthetic data points



Figure 9: Under-sampling and over-sampling illustration [38].

using techniques like SMOTE [37]. By increasing the representation of the minority class, over-sampling aims to provide the model with more balanced training data, improving its ability to learn patterns and make accurate predictions for the minority class.

Under-sampling, on the other hand, involves reducing the number of instances in the majority class to align with the ones in the minority class (Figure 9). This can be done, for example, by randomly picking a subset of instances from the majority class. By reducing the dominance of the majority class, under-sampling ensures that the model is not overwhelmed by the abundant instances from the majority class, allowing it to focus on learning the patterns and characteristics of the minority class.

While under-sampling can be effective in reducing the dominance of the majority class (Figure 9), it also runs the risk of discarding valuable data. Over-sampling techniques generate synthetic or duplicated instances to address this issue, ensuring that no information is lost in the process (but the information is synthetic or duplicated, so other issues can turn out).

In this research, both balancing methods have been implemented in order to determine the most suited technique. For what concerns over-sampling, it has been chosen to duplicate existing minority class instances, whereas for under-sampling it has been chosen to follow the procedure of randomly taking samples belonging to the majority class.

Another method that should be considered (but which is actually not a balancing technique) is to use cost-sensitive learning techniques, in order to associate more weight to the error made by the classifier on the minority class (this allows also to avoid losing information or to generate synthetic data).

These balancing techniques are important for classification tasks for several reasons, for example:

- *Mitigating bias*: Class imbalance can introduce bias in the model's predictions, where it tends to favor the majority class. Balancing the dataset helps to reduce this bias, enabling the model to learn from both classes more effectively.
- *Capturing Rare Events*: In real-world scenarios, there are often instances of rare events or classes that may have significant importance. Without balancing techniques, these rare events may be underrepresented in the training data, leading to poor performance in detecting and predicting them. Those techniques ensure that the model pays adequate attention to these rare events, improving its ability to describe the reality accurately.
- *Learning Discriminative Patterns*: Balancing techniques help to overcome the issue of ignoring or downplaying the patterns and characteristics of the minority class, by providing a more balanced representation of the classes, allowing the model to learn discriminative patterns from both classes. As a result, the model becomes more capable of capturing the nuances and complexities of the real-world data distribution.
- *Improved generalization*: Unbalanced datasets can lead to poor generalization, as the model may struggle to learn the minority class patterns and fail to make accurate predictions on unseen data. Balancing the dataset helps the model capture important features from both classes, leading to improved generalization and better performance on unseen data.

4.4.3 Normalization methods

Normalization is typically needed before building Machine Learning models because many of them are sensitive to the scale of the input features. Normalization can help to improve the model performances by ensuring that the input features have a similar scale. It is very important to keep the statistics computed on the training data and reuse them also to normalize test and validation data.

Here are some reasons why normalization is important in this context:

- *Better performances*: Normalization can help to improve the performance of Machine Learning algorithms. Some of them are based on distance metrics that are affected by the scale of the input features. If the input features have different scales, the algorithm may be biased towards features with larger scales, leading to suboptimal performance.
- *Faster convergence*: Normalization can help ML algorithms to converge more quickly. Some optimization algorithms, such as gradient descent, converge faster when the input features have a similar scale.
- *Improved interpretability*: Normalization can improve the models interpretability. When the input features have vastly different scales, it can be difficult to interpret the coefficients of the model or the importance of the features.

There are several normalization techniques that can be used, each one with its own advantages and drawbacks.

Min-Max scaling One of the simplest normalization technique consists in scaling all the data in such a way that all features have values in the same range, typically between 0 and 1 (using a different range is trivial). This technique is called min-max scaling, and it is based on the computation of the minimum m_j and the maximum M_j values for each feature ($j = 0, 1, \dots, n - 1$). Then each feature of an input feature vector x is normalized by applying the following linear scaling:

$$\bar{x}_j = \frac{x_j - m_j}{M_j - m_j}$$

This ensures that all the training features assume values between 0 and 1. Note that for data outside the training set, normalized values may still fall outside the $[0, 1]$ range. Sometimes the transformation performed by min-max scaling degenerates due to the presence of a few unrepresentative outliers in the training data. A single very large (or very small) value causes the compression of all the others. This scaling algorithm works very well in cases where the standard deviation is very small, or in cases when features are not normally distributed.

Mean-Var scaling Mean-var scaling is another normalization technique that does not present the same drawback. In mean-var scaling, each feature is linearly scaled

to have zero mean and unit variance. Training data is used to compute the mean and the standard deviation of each feature. Then the components of a given feature vector x are normalized accordingly:

$$\bar{x}_j = \frac{x_j - \mu_j}{\sigma_j}$$

It assumes a normal distribution of data within each feature.

Max-Abs scaling In some applications' sparseness of features is a very important property. A feature is sparse when its value is most of the time exactly zero. Both min-max scaling and mean-var scaling do not preserve sparsity. A normalization scheme that is suitable for sparse data is max-abs scaling. It consists in dividing each feature by the largest absolute value found in the training set:

$$\bar{x}_j = \frac{x_j}{V_j}$$

Where V_j is the maximum absolute value element of the j -th feature considered.

Robust scaling Robust scalar algorithms scale features that are robust to outliers. The method it follows is almost similar to the min-max scaler, but it uses the interquartile range. The median and scales of the data are removed by this scaling algorithm according to the quantile range. Thus, follows the following formula:

$$\bar{x}_j = \frac{x_j - Q_1(x_j)}{Q_3(x_j) - Q_1(x_j)}$$

Where Q_1 is the first quartile, while Q_3 the third.

4.5 Machine Learning models

To address the classification problem at hand, different Machine Learning models have been considered. This section explores the various employed models, highlighting their strengths and limitations. Additionally, the significance of feature selection methods in enhancing the performance of these models is here described, as selecting relevant and informative features from the satellite data can significantly improve the classification accuracy.

4.5.1 Features selection

One of the main problem is to find which are the most relevant and informative features to solve the particular classification problem at hand (feature selection). It has already been discussed all the spectral indexes (features) extracted from satellite data, and the assessment of which of them are most affected by manure application, in our specific context of interest.

There are a lot of many different feature selection techniques, in this particular case wrapper methods have been used. Wrapper techniques approach feature selection as a search problem. Different combinations of features are considered, evaluated using specific performance metrics (such as detection accuracy), and compared to determine the most promising subsets. Two of the most popular wrapper methods to address this challenge are backward and forward feature selection.

Forward feature selection begins with an empty set of features. It explores the feature space by adding one feature at a time, and then evaluates the performance improvement. The feature that provides the highest improvement is selected, and the process iterates until a stopping criterion is met, such as reaching a desired level of accuracy or including a predefined number of features.

On the other hand, backward feature selection starts with the complete set of features and progressively eliminates features that are considered less relevant or contribute less to the detection accuracy. The process iteratively removes one feature at a time, evaluates the performance of the reduced feature set, and discards the feature that results in the least improvement. This process continues until a satisfactory subset of features is obtained.

Both backward and forward feature selection methods have their advantages and limitations. The backward procedure can be computationally efficient when the number of features is large, as it starts with the complete set and progressively reduces it. However, it may overlook interactions between features and potentially discard useful features too early in the process. Instead, the forward method explicitly explores the feature space, which can be time-consuming for large feature sets. Nevertheless, it may capture interactions between features more effectively and generally leads to higher detection accuracy.

4.5.2 Classifiers

In this specific section, a few hints will be given regarding the Machine Learning models chosen for performing the classification task.

One may wonder why Deep Learning (DL) was not used. For the purposes of this research project, it was chosen to use classical Machine Learning methods for several reasons, primarily:

- *Limited availability of labeled data*: Deep Learning models typically require large, labeled datasets for training. If labeled data is scarce, classical Machine Learning techniques can still provide effective results, unlike DL methods.
- *Interpretability*: Classical ML algorithms such as decision trees or logistic regression offer better interpretability compared to DL models. If understanding the reasoning behind predictions is crucial, classical ML methods are preferred.
- *Feature engineering*: Deep Learning models excel at learning features from raw data, but they may not perform well when clear patterns are lacking or when extensive feature engineering is required. Classical ML methods provide more control over feature selection and extraction.
- *Computational resources*: Deep Learning models are computationally demanding and often require specialized hardware. If there are limitations in computational resources or time constraints, classical ML methods may be more viable.

Now that the main reasons have been briefly discussed, let's explore the Machine Learning models (classifiers) that have been considered and utilized in this thesis activity.

Logistic Regression Logistic regression is one of the methods particularly effective for binary classification problems, even though it can suit the multi-class case. It focuses on predicting the probability of an event (or outcome) occurring based on a set of input variables. The fundamental idea behind logistic regression is to transform the linear regression equation into a sigmoid function, which maps the continuous range of predicted values to a probability range between 0 and 1. This

allows logistic regression to classify data points into two distinct categories. Logistic regression assumes that the relationship between the input variables and the log-odds of the outcome follows a linear pattern, enabling it to estimate the probability of a particular outcome given the input variables. Furthermore, there should be no, or very little, multicollinearity between the predictor variables.

Linear Discriminant Analysis Linear Discriminant Analysis (LDA) is yet another method used also to perform classification tasks. It aims to find a linear combination of features that maximally separates different classes in a dataset. LDA assumes that data is normally distributed within each class and that the classes have equal covariance matrices. By projecting the data onto a lower-dimensional subspace, it maximizes the inter-class scatter while minimizing the intra-class scatter. This results in a discriminant function that can be used to classify new observations and allows also to prevent over-fitting. In addition to its classification capabilities, it offers several other useful features, such as providing insights into the feature reduction problem.

K-Nearest Neighbors Classifier K-Nearest Neighbor (KNN) is another algorithm used for classification tasks. It operates on the principle of proximity, where the class label of a new instance is determined by the class labels of its K nearest neighbors in the training dataset. KNN assumes that instances with similar features are likely to belong to the same class. To classify a new observation, KNN calculates the distances between that observation and all the training instances, and then selects the K nearest neighbors based on these distances. The class label of the new observation is determined by majority voting among the K neighbors. KNN is a non-parametric algorithm, meaning it does not make any assumptions about the underlying distribution of the data. The choice of the value of K is crucial, as it affects the algorithm's sensitivity, and so it can lead to over-fitting or under-fitting issues. KNN can be slower during the prediction phase, especially for large datasets.

Support Vector Classifier Support Vector Classifiers (SVC) operate by finding an optimal hyperplane that separates different classes in the feature space. The main idea is to maximize the margin (hard/soft), which is the distance between

the hyperplane and the nearest data points from each class. By maximizing it, SVC aims to achieve a robust decision boundary that generalizes well to unseen data. SVC can handle either linearly separable or non-linearly separable data, by using various kernel functions (linear, polynomial, radial basis function, sigmoid). These kernels map the original feature space into a higher-dimensional space, where the data becomes linearly separable. SVC is particularly effective in scenarios with high-dimensional data. However, their performance can be sensitive to the choice of hyperparameters.

Random Forest Classifier The Random Forest Classifier is a commonly used ML algorithm for classification tasks. It is an ensemble learning method that combines multiple decision trees to make predictions. A single decision tree, shortly, is a predictive modeling technique that maps out a series of decisions and their possible consequences in a tree-like structure. The Random Forest Classifier works by constructing a multitude of decision trees during the training phase. Each tree is trained on a randomly sampled subset of the original data, and the final prediction is determined by aggregating the predictions of all the individual trees. This aggregation can be done through majority voting (for classification tasks) or averaging (for regression tasks). The Random Forest Classifier offers several advantages, including robustness against over-fitting, the ability to handle high-dimensional data, and resistance to noise and outliers. Additionally, it can capture complex relationships between features and target variables, making it suitable for a wide range of classification problems. The algorithm also provides insights into feature importance, allowing for feature selection and interpretation. However, in Random Forest classifiers the number of trees and other hyperparameters should be carefully tuned for optimal performance.

4.5.3 Hyperparameters tuning

For the machine learning models under consideration, the hyperparameters have been fine-tuned (Figure 10). However, what does the process of tuning entail, and how does it work?

Hyperparameter tuning refers to the procedure of identifying the optimal values

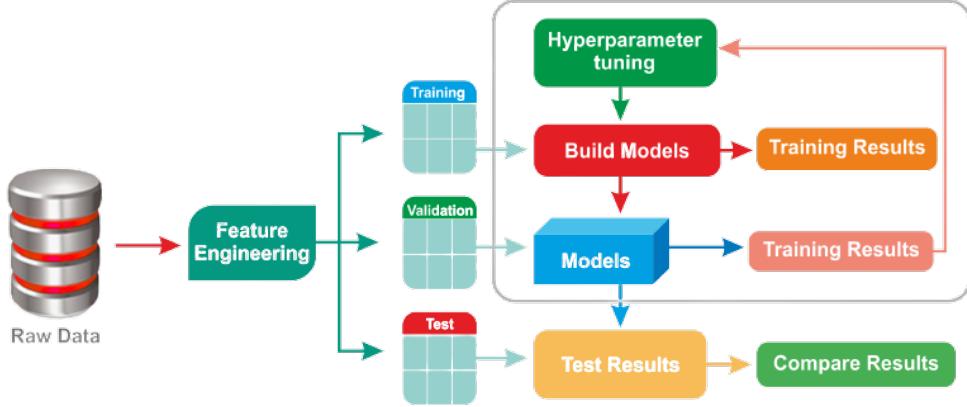


Figure 10: A typical data pipeline [39].

for the hyperparameters of a machine learning algorithm. Hyperparameters are predetermined parameters that govern the behavior of the algorithm and are set prior to the learning process.

There exist numerous methods for performing hyperparameter tuning, but two widely employed approaches are grid search and random search.

Grid search is a straightforward and effective method, although it can be time-consuming. Typically, it is feasible to use grid search for selecting one or two hyperparameters (sometimes three). If the best parameter combination lies on the boundary of the grid, the grid must be expanded to ensure that the range of parameters encompasses the optimum value. Determining a reasonable set of potential values for certain hyperparameters can be challenging. Conducting exploratory tests may aid in identifying the suitable range and number of values to try. In certain cases, hyperparameters may assume values with varying orders of magnitude, and the set of possible values should follow a geometric progression. The results obtained from grid search should provide insights into the model’s sensitivity and suggest which hyperparameters require careful tuning. If computational resources permit, it can be beneficial to repeat the search with a finer grid centered around the combination found in the initial round.

An alternative to grid search is random search, which involves trying multiple random combinations of the hyperparameters in a sequence. Random search offers several advantages over grid search: it optimally utilizes the available time by trying out the exact number of combinations within a given time budget, it enables the optimization of any number of hyperparameters, and it avoids wasting time on

redundant combinations when there is a hyperparameter with minimal impact on the final performance. However, it is important to note that random search is inherently random and thus less predictable compared to grid search.

4.6 Performance evaluation

The evaluation of the proposed models is crucial to objectively assess their effectiveness and efficiency in addressing the research problem, bridging existing gaps in the field, and contributing to the advancement of knowledge in the domain. Different aspects are covered in this section, regarding different metrics and techniques that have been used in order to assess and compare classifiers performances.

4.6.1 KFold Cross-Validation

KFold Cross Validation is a technique commonly used to assess the performance of a model and mitigate potential bias in the evaluation process. It involves splitting a dataset into K equally sized folds or subsets, where each fold is used as a validation set while the remaining $K - 1$ folds are used for training. This process is repeated K times, with each fold serving as the validation set exactly once (Figure 11).

The primary purpose of KFold Cross Validation is to provide a more robust estimate of the model's performance by reducing the impact of the data partitioning. By iteratively using different subsets of data for training and validation, it helps to reduce the variance in the evaluation metrics. This method is particularly useful when the available dataset is limited (which is the case), as it allows maximizing the use of the available data for both training and evaluation.

One of the main advantages of KFold Cross Validation is that it enables the evaluation of the model on multiple, diverse subsets of data, ensuring that the performance metrics are not skewed by the specific characteristics of a single training-validation split. This helps to capture the model's generalization ability and assess its performance across different subsets of the data.

If K is equal to the number of samples in the dataset, it means that each fold will contain exactly one sample. This scenario is known as leave-one-out cross-validation (LOOCV). Even though it provides an unbiased estimate of the model's performance, it can be computationally expensive, especially for large datasets, as



Figure 11: KFold Cross-Validation example with $K = 10$ [40].

it requires fitting the model K times. Furthermore, it's worth noting that LOOCV tends to have higher variance in its performance estimate (compared to other K-fold cross-validation approaches with a smaller K). This is because each training set has a high degree of overlap with the other training sets, making them highly correlated.

Another case is the Stratified KFold, a modification of the KFold Cross Validation technique, that addresses the potential issue of imbalanced class distributions in the target variable. It ensures that each fold contains approximately the same proportion of samples from each class as the whole dataset. This is particularly important when dealing with classification tasks where the classes are not evenly distributed (beneficial when the minority class is of particular interest or when the classification problem requires accurate predictions for all classes).

4.6.2 Confusion matrix

A confusion table, also known as a confusion matrix, is a tabular representation that provides a summary of the performance of a classification algorithm. It is commonly used in Machine Learning and statistics to evaluate the accuracy of a model's predictions.

The confusion table organizes the predicted class labels against the actual class labels in a square matrix format. In the 2×2 case (Table 4), which is the case of this research since just two class are present (manured and not manured), it has four cells. Each row represents the instances in a predicted class, and each column

Table 4: Confusion matrix.

		Predicted	
		False	True
Actual	False	TN	FP
	True	FN	TP

represents the instances in an actual class.

The four cells of the confusion table represent different outcomes:

- *True Positive (TP)*: This cell represents the instances that were correctly predicted as positive by the classifier. In other words, it shows the number of instances that were actually positive and were correctly identified as positive.
- *False Positive (FP)*: This cell represents the instances that were incorrectly predicted as positive by the classifier. These are instances that were actually negative but were mistakenly identified as positive.
- *True Negative (TN)*: This cell represents the instances that were correctly predicted as negative by the classifier. It shows the number of instances that were actually negative and were correctly identified as negative.
- *False Negative (FN)*: This cell represents the instances that were incorrectly predicted as negative by the classifier. These are instances that were actually positive but were mistakenly identified as negative.

By examining the values in the confusion table, we can gain insights into the model's performance. It allows us to understand how well the classifier is able to distinguish between different classes and identify any patterns of misclassification.

It's important to note that the confusion table can be extended beyond the 2×2 case to accommodate classification problems with more than two classes. In such cases, the table would have multiple rows and columns, with each cell representing the intersection between a predicted class and an actual class. This generalization allows for a comprehensive analysis of the classifier's performance across multiple classes.

4.6.3 Accuracy, Precision, Recall and F1-score

In addition to the confusion matrix, several other performance metrics can be calculated to further evaluate the effectiveness of a model.

One of the most commonly used indicators is the accuracy, which is defined as the number of samples correctly classified out of all the samples present in the test set.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

While accuracy provides a general measure of correct classification, precision, recall, and F1 score offer deeper insights into specific aspects of the classification process. When dealing with balanced class distributions, accuracy serves as a reliable indicator. However, in scenarios where class imbalances exist, precision and recall become crucial.

Precision (for the positive class) is the number of samples actually belonging to the positive class out of all the samples that were predicted to be of the positive class by the model.

$$Precision = \frac{TP}{TP + FP}$$

Recall (for the positive class) is the number of samples predicted correctly to be belonging to the positive class out of all the samples that actually belong to the positive class.

$$Recall = \frac{TP}{TP + FN}$$

F1_score is the harmonic mean of the precision and recall scores obtained for the positive, or negative, class.

$$F1_Score = \frac{2 * precision * recall}{precision + recall}$$

These metrics provide additional insights into different aspects of the classification process, allowing for a more comprehensive assessment, allowing practitioners to make informed decisions based on specific requirements and domain considerations. By examining a combination of these metrics, analysts can uncover the strengths and weaknesses of a model, optimize it for desired outcomes, and make meaningful comparisons between different models.

Ultimately, the use of these performance metrics goes beyond a single numerical value, allowing for a comprehensive assessment and ensuring that models are well-suited for real-world applications.

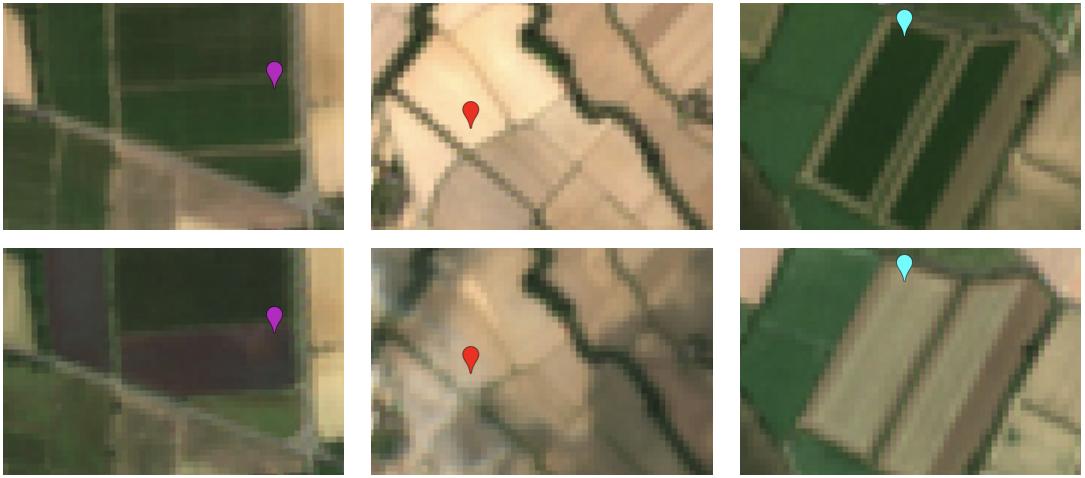


Figure 12: Difference in the visible spectrum considering the acquisitions right before (top) and after (bottom) manure application for some fields (marker), located in Spain.

5 Results

Now that all the premises - necessary to better understand the used techniques - have been made, it is time to discuss one of the most important parts, namely the results obtained from the experiments carried out.

Once all the features have been extracted from all the crop fields of interest, the very first step has been to perform an analysis on the dataset. Except from the standard preliminary phases - like visualization and description of data, in order to get a clearer idea on the data at hand (outlier detection and removal) - what has been initially done was to visualize whether manure application had similar effects on the visible spectrum for some, randomly selected, fields (Figure 12).

It is known that fertilizers are providing important nutrients to plants, such as nitrogen, phosphorus and potassium, which are essential for healthy and vigorous plant growth. These nutrients promote the development of green leaves and the improvement of overall plant production. Unfortunately, the response times of plants depend a lot on the type of fertilizer. Chemical fertilizers give rather short response times, even of a few days, while organic fertilizers cause changes in the vitality of plants in much longer times, so the effect may not even be visible at all because it mixes with other factors.

As it can be visually noticed, the effect of manure application on the visible

Table 5: Top 5 indexes more correlated by manure application, for crop fields located in Spain (left Sentinel-2, right Sentinel-1).

feature	importance	feature	importance
EOMI3	0.720542	VH	0.366449
NSNDVI	0.672114	DIF	0.362870
SCI	0.661566	AVE	0.340196
EOMI1	0.628979	RAT1	0.312078
SDI	0.584395	RVI	0.305876

spectrum is not very clear. This suggests that focusing on the visible spectrum is not enough, and therefore other indexes are needed.

Thus, the following step was to identify the spectral features more impacted by the manure application on crop fields. It has already been described, in the method and experiments section, the formula used to calculate numerically the importance of each index (or feature), as well as the p-value.

What turns out from this analysis is what has been illustrated in Table 5, where it can be seen what are the features (top 5) more impacted by manure application for all the indexes extracted from the considered satellites constellations (Sentinel-1 and Sentinel-2), for the crop fields located in Spain (within the top 5, all the spectral indexes have a p-value significantly lower than the classical 0.05 threshold). Initially, the Italian crop field behavior has not been studied, since the objective was to see whether a model, trained on the Spain context can also generalize the detection of manure application in a completely different context.

Then, once the features more impacted by manure application have been identified, the trend of few indexes has been studied for few - randomly selected - crop fields of reference, still located in Spain (Figures 13, 14). Please notice that the charts are not illustrating the true feature value, but the one scaled between -1 and 1 , just for visualization purposes.

As it can be seen on the illustrated figures, the effect of manure application on crop fields varies. For some of them, the same index is changing more after manure application with respect to other fields. Furthermore, the indexes are not only affected by the target phenomenon, they also vary significantly during other

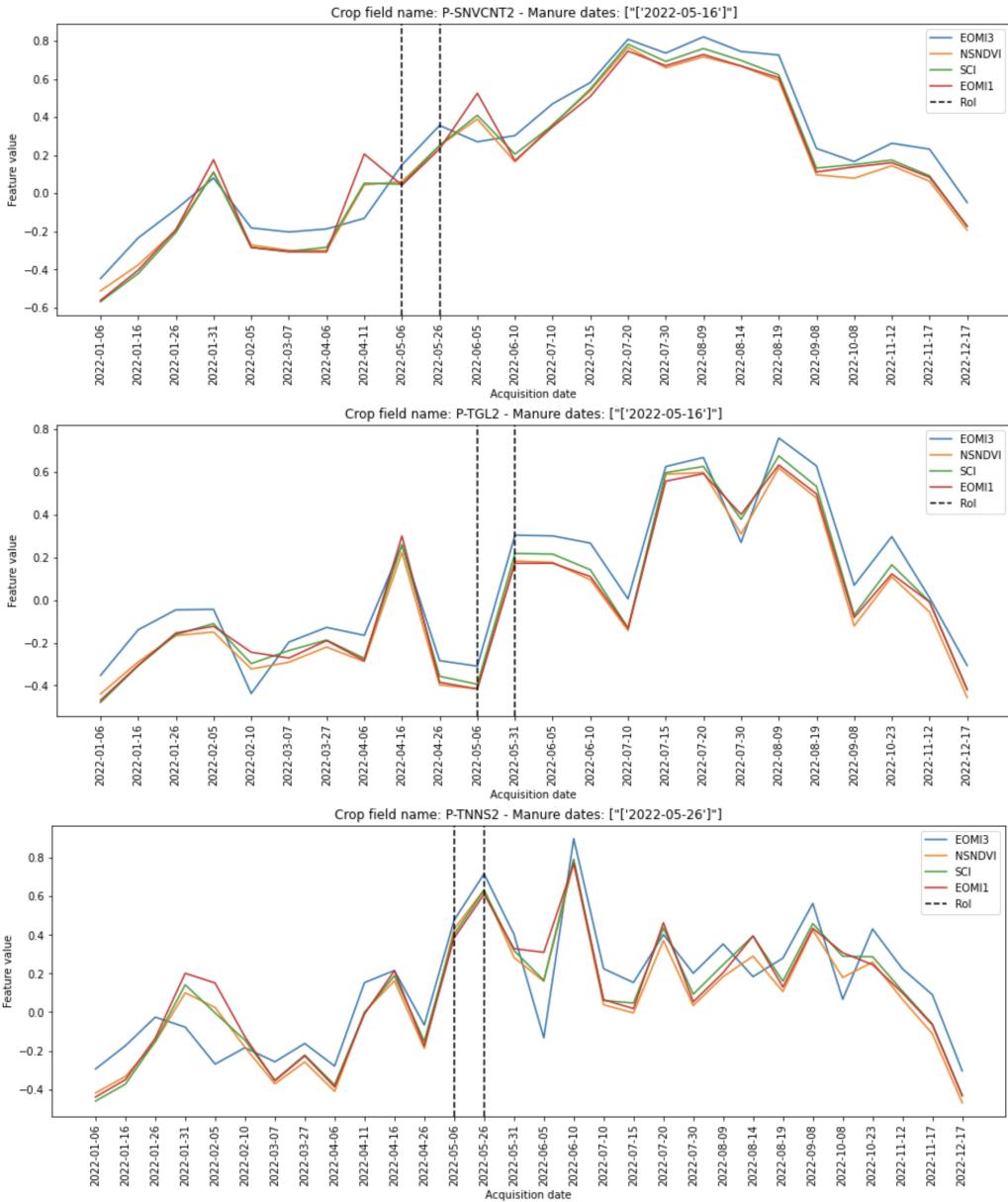


Figure 13: Trend of top Sentinel-2 spectral indexes more impacted by manure application (Spanish context).

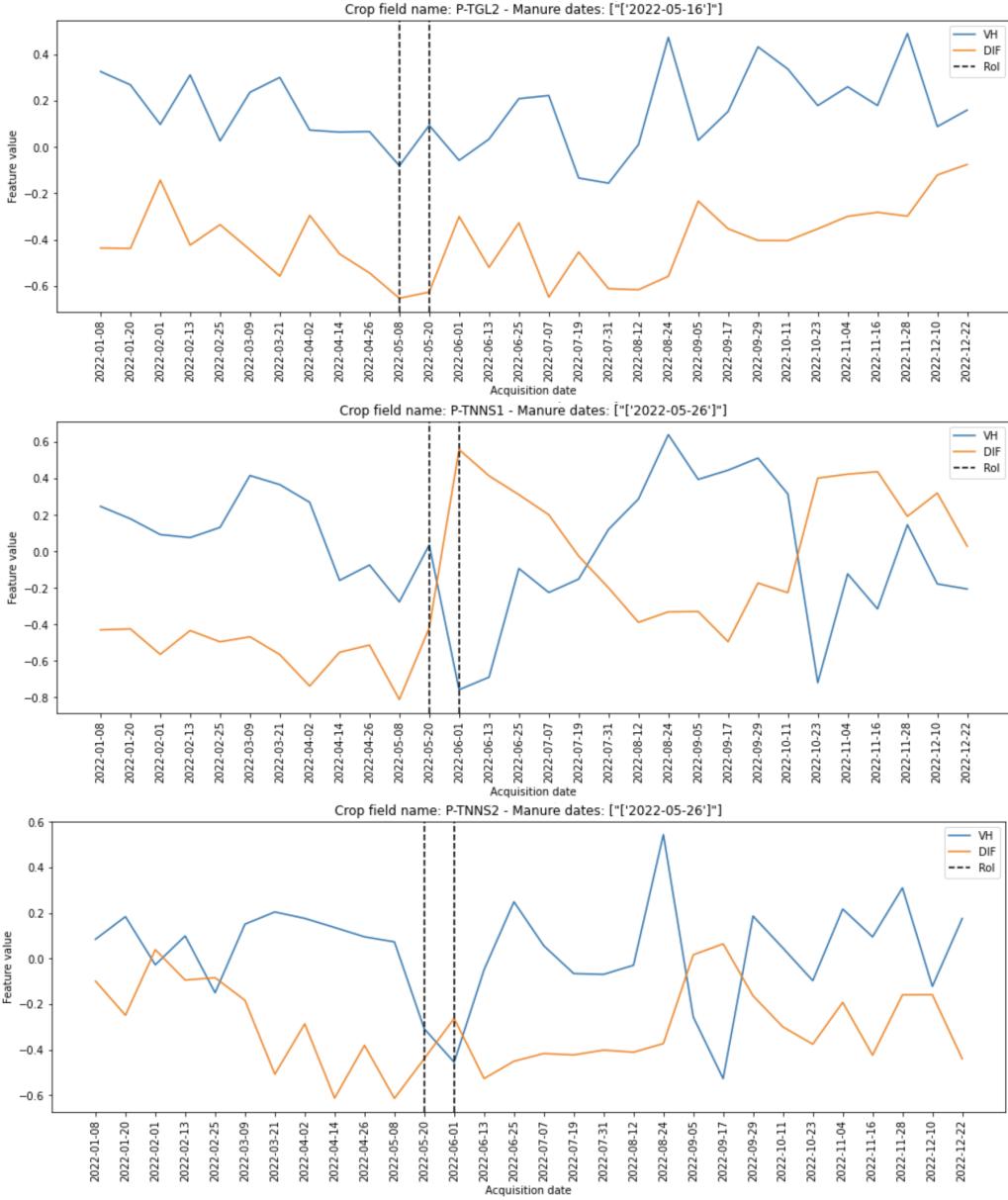


Figure 14: Trend of top Sentinel-1 spectral indexes more impacted by manure application (Spanish context).

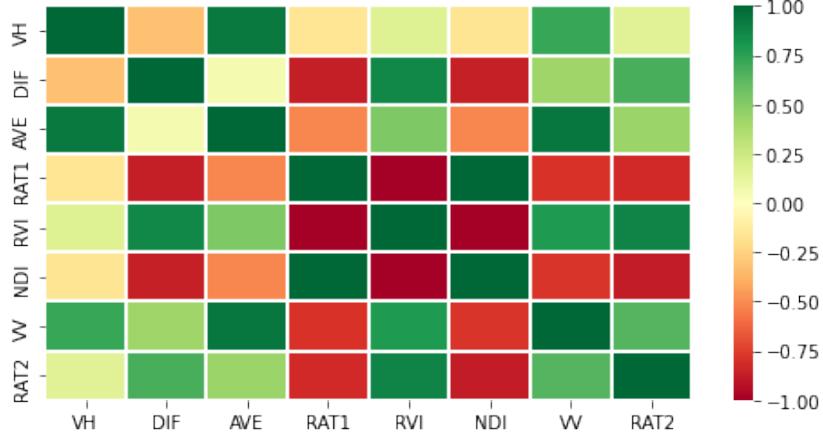


Figure 15: Correlation between different Sentinel-1 indexes (Spanish context).

periods of the year.

It seems that most of these indexes are correlated (Figures 16, 15), but recall that our model uses the difference between two consequent satellite acquisitions (for all the indexes), in order to predict whether manure has been applied within the period of interest.

The next step, immediately after the one regarding analysis, is to build a model - using the previously mentioned features - and see how well it behaves in generalization both for the same context (Spanish crop fields) and for a completely different context.

Once the modified DataFrames are obtained (see the previous section) and balanced (using under-sampling, since it provides better results), multiple models have been trained. Stratified k-fold cross-validation ($K=5$) has been used to assess the goodness of the performances, averaging across multiple seeds.

Initially, a model built upon Sentinel-2 extracted indexes more impacted by manure (EOMI3, SCI, EOMI1, SDI) has been compared to one that used just Sentinel-1 features (DIF). In addition, those indexes have been normalized using different techniques and the one that provided better results is the Max-Abs scaler. The indexes have been selected through forward feature selection method.

Since the dataset has been balanced, for the sake of simplicity, only accuracy results are illustrated (Table 6).

It is clear that opting for Sentinel-2 indexes provides better results than using Sentinel-1 ones, so it can be said that radar indexes are not so relevant to the

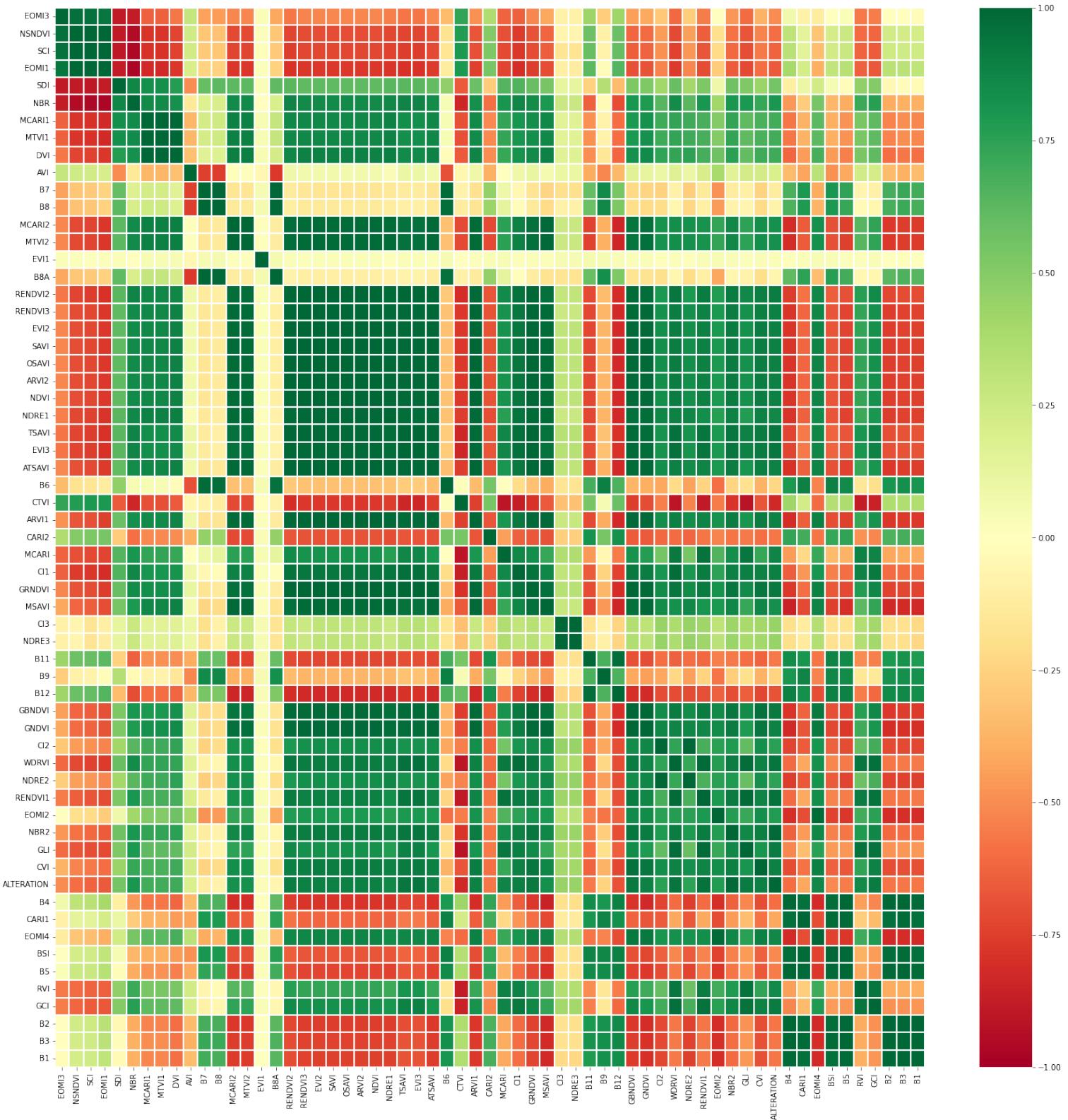


Figure 16: Correlation between different Sentinel-2 indexes (Spanish context).

Table 6: Accuracy of models using "best" indexes, trained on the Spanish context (left Sentinel-2, right Sentinel-1).

Model	Train acc	Test acc	Model	Train acc	Test acc
LR	0.80	0.77	LR	0.52	0.52
LDA	0.86	0.84	LDA	0.51	0.50
SVC	0.90	0.88	SVC	0.67	0.64
KNN	0.82	0.80	KNN	0.67	0.62
RFC	0.85	0.83	RFC	0.70	0.52



Figure 17: Confusion matrix of a model trained on Spanish context and tested on the Italian one.

classification problem at hand. The SVC model is best for both cases, providing good performances, without over-fitting nor under-fitting.

The next step has been to validate whether the same model can work well at detecting manure application for crop fields located in a different country (Italy). It results that its performances - especially for what regards the class of interest (manured observations) - are not at all remarkable (Figure 17).

A possible reason for this behavior is that the spectral signature of manure application is homogeneous within fields located in the same country. Thus, different models have been trained on the Italian context, without considering this time Sentinel-1 indexes, since it is plausible that these features are not relevant (as happens for the Spanish context).

For sake of simplicity, the different models use the same set of features, normal-

Table 7: Accuracy of models using "best" Sentinel-2 extracted indexes, trained on the Italian context.

Model	Train acc	Test acc
LR	0.58	0.54
LDA	0.65	0.60
SVC	0.70	0.69
KNN	0.68	0.57
RFC	0.78	0.63

ization techniques and K for Stratified Cross Validation (5), as of the models trained on the Spanish context. Even in this case the dataset has been balanced so just the accuracy results are illustrated (Table 7).

It seems that the performance are worse than the one of the model that was trained on detecting manure application for crop fields located in Spain. By the way, despite this phenomenon, its generalization capabilities are better. If the model performances are measured on other crop fields still located in Italy (north) the recall (for the manured class) passes from 30% to almost 75%.

This suggests the fact that the spectral signature of manure application is homogeneous within fields located in the same country, as stated previously.

Then, the objective has been to see how the model trained on detecting manure application for Italian fields behaves - in classification - for other crop fields, where the information about manure dates is not available (DUSAf). The purpose is to check (using a model trained on all available manured crops located in Italy) the distribution of detections during a whole reference year (2018) for crops (still located in Italy) where there are no information about the actual manure application dates (DUSAf). The expected behavior is that the model detects more manure applications in periods of the year when it is more likely (prior knowledge) that the fields are actually manured (spring period mainly).

It can be said that there are many false positives (Figure 18), since the number of manure application events identified seems to be excessive (especially during summer months).

Since this phenomenon is quite strange, other ways have been tried in order to

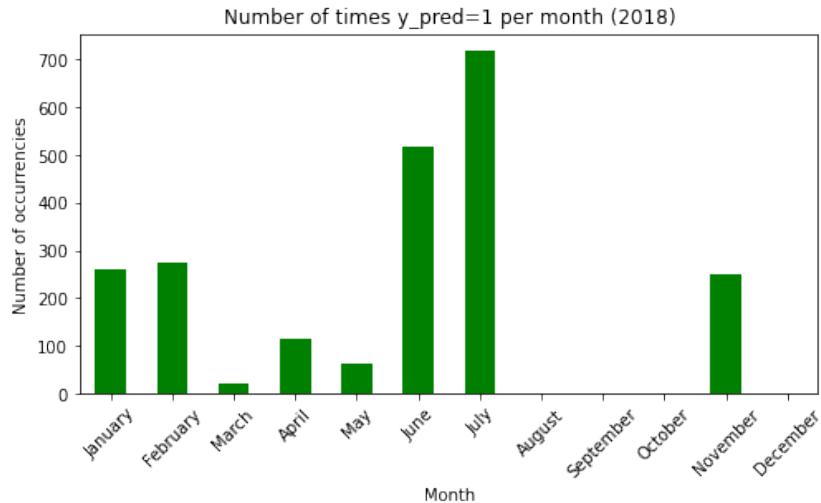


Figure 18: Number of occurrences detected, per month, on the DUSAf (model that uses just Sentinel-2 indexes)

improve model performances, especially for what regards the generalization capabilities.

An interesting experiment showed that combining thermal indexes (B10, B11) from Landsat-8 with the optical ones already extracted from Sentinel-2 (EOMI3, SCI, EOMI1, SDI) allows improving the performances of models trained - and tested - on crops located within the same country, handling properly the difference in temporal resolution for the two satellites constellations.

This idea comes from the fact that manure could - as a conjecture - cause physical and chemical reactions that can vary the thermal properties, over the area of interest.

Whereas, from an implementation view point, the concatenation was made by taking the closest observation of Landsat-8 - in terms of acquisition date - to the considered observation for Sentinel-2.

What happens is that performances improve both by training the model on the Spanish context (accuracy increases from 90% to 92%, without overfitting), and for the one trained considering the Italian reality (accuracy increases from 70% to 82%, for both train and test sets).

Therefore, if the model trained on the Italian context (combining Landsat-8 and Sentinel-2 indexes) is "used" on the DUSAf dataset, detections during the expected period - spring, May - increase, which makes sense. The same cannot be said for the ones during July and August, which are strange in a sense (Figure 19). Thus,

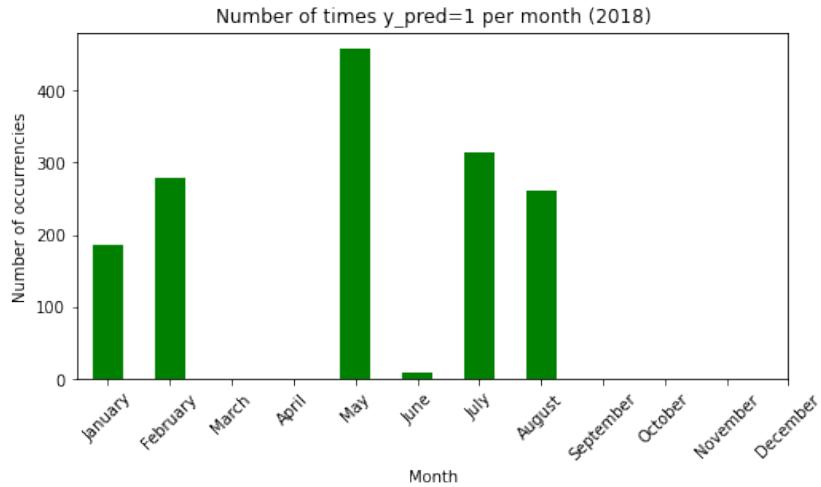


Figure 19: Number of occurrences detected, per month, on the DUSAf (model that combines Sentinel-2 and Landsat-8 indexes)

thermal data provides additional information to this specific context of application.

In conclusion, using the same indexes (EOMI3, SDI, EOMI1, SDI, thermal) exploiting only Landsat-8 satellite produces worse performance (overfitting, with almost 5% decrease for test accuracy), compared to combining Landsat-8 (thermal) and Sentinel-2 (optical) indexes, albeit with two different temporal resolutions. This is probably given by some aspects, including the lower spatial (and temporal) resolution of Landsat-8 and that the bands are not really coincident between the two satellites constellations (something is lost in the translation of the indexes).

6 Conclusions

The developed model and its results present a significant milestone in addressing the complex classification problem of manure application in crop fields using time series of satellite indexes. The model's ability to detect specific manure application dates opens new avenues for monitoring agricultural practices, ensuring compliance with environmental regulations, such as the EU Nitrates Directives.

This research represents the very first attempt to tackle this challenging issue, and the achieved results should not be disregarded. The model's successful identification of manure application dates serves as a valuable foundation for developing fast and cost-effective tools for monitoring agricultural practices on a larger scale.

Moving forward, there exist several potential lines of expansion that can enhance the impact of this research. Investigating the homogeneity of spectral responses in manured crops across different countries or regions beyond those considered in this study could provide valuable insights for international agricultural management. Additionally, exploring the influence of other spectral indexes derived from satellite data on manure application detection could further refine the model's accuracy.

Furthermore, incorporating cost-sensitive learning techniques, instead of dataset balancing, as observed in this study, could lead to even better results and improved model generalization. Though challenging, efforts must be made to increase the dataset by incorporating new manured fields to enhance the model's robustness and adaptability.

In conclusion, this research represents a crucial step towards sustainable agriculture and environmental conservation. Utilizing satellite data to detect manure application dates can lead to more effective compliance with environmental regulations, promotion of responsible farming practices, and safeguarding ecosystems. However, there is still ample space for further improvements and exploration, enabling the unlocking of the model's full potential to revolutionize crop management and environmental protection on a global scale.

List of Figures

1	Smallest rectangles that comprehend all analyzed fields. (a) Plots located in Spain. (b) Plots located in Italy.	11
2	Frequency distribution of manure application. (a) Plots located in Spain. (b) Plots located in Italy.	11
3	Polygons (in yellow) representing all fields. (a) Plots located in Spain. (b) Plots located in Italy.	12
4	Flowchart of the overall approach.	15
5	Satellites representation.	18
6	Visualization of (a) NDVI (the greener, the higher the value, the more the blue, the lower). (b) EOMI2 (the more the brown, the higher the value, the more the blue, the lower). Those indexes have been calculated near the plot represented in black, on the left side (namely P-BLD).	22
7	Temporal change of EOMI4 values for all treatments. (Left) Green waste compost; (Right) sheep manure treatments [18].	25
8	An example of the structure of the modified dataset.	26
9	Under-sampling and over-sampling illustration [38].	27
10	A typical data pipeline [39].	35
11	KFold Cross-Validation example with $K = 10$ [40].	37
12	Difference in the visible spectrum considering the acquisitions right before (top) and after (bottom) manure application for some fields (marker), located in Spain.	41
13	Trend of top Sentinel-2 spectral indexes more impacted by manure application (Spanish context).	43
14	Trend of top Sentinel-1 spectral indexes more impacted by manure application (Spanish context).	44
15	Correlation between different Sentinel-1 indexes (Spanish context). .	45
16	Correlation between different Sentinel-2 indexes (Spanish context). .	46
17	Confusion matrix of a model trained on Spanish context and tested on the Italian one.	47

18	Number of occurrences detected, per month, on the DUSAf (model that uses just Sentinel-2 indexes)	49
19	Number of occurrences detected, per month, on the DUSAf (model that combines Sentinel-2 and Landsat-8 indexes)	50

List of Tables

1	Input DataFrame structure: example.	19
2	Optical indexes.	20
3	Radar indexes.	23
4	Confusion matrix.	38
5	Top 5 indexes more correlated by manure application, for crop fields located in Spain (left Sentinel-2, right Sentinel-1).	42
6	Accuracy of models using "best" indexes, trained on the Spanish context (left Sentinel-2, right Sentinel-1).	47
7	Accuracy of models using "best" Sentinel-2 extracted indexes, trained on the Italian context.	48

Bibliography

- [1] Food and Agriculture Organization of the United Nations. In: Statistical Yearbook - 2022. FAO statistics, 2022. Chap. 1. Economic dimensions of agriculture. ISBN: 9789251369302. DOI: 10 . 4060 / cc2211en. URL: <https://www.fao.org/3/cc2211en/cc2211en.pdf>.
- [2] John Cleland. "World Population Growth; Past, Present and Future". In: Environmental and Resource Economics 55 (Aug. 2013). DOI: 10 . 1007 / s10640-013-9675-6.
- [3] Sumit Chakravarty et al. "Deforestation: Causes, Effects and Control Strategies". In: Apr. 2012. ISBN: 978-953-51-0569-5. DOI: 10.5772/33342.
- [4] Fabien H. Wagner et al. "Mapping Tropical Forest Cover and Deforestation with Planet NICFI Satellite Images and Deep Learning in Mato Grosso State (Brazil) from 2015 to 2021". In: Remote Sensing 15.2 (2023). ISSN: 2072-4292. DOI: 10 . 3390/rs15020521. URL: <https://www.mdpi.com/2072-4292/15/2/521>.
- [5] Travis P. Hignett. "History of Chemical Fertilizers". In: Fertilizer Manual. Ed. by Travis P. Hignett. Dordrecht: Springer Netherlands, 1985, pp. 3–10. ISBN: 978-94-017-1538-6. DOI: 10 . 1007 / 978 - 94 - 017 - 1538 - 6 _ 1. URL: https://doi.org/10.1007/978-94-017-1538-6_1.
- [6] Heinrich W. Scherer et al. "Fertilizers". In: Ullmann's Encyclopedia of Industrial Chemistry. John Wiley and Sons, Ltd, 2009. Chap. 1. General. ISBN: 9783527306732. DOI: 10 . 1002/14356007. a10_323 . pub3. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/14356007.a10_323.pub3.
- [7] Peter J.A. Kleinman et al. "Managing Animal Manure to Minimize Phosphorus Losses from Land to Water". In: Animal Manure. John Wiley and Sons, Ltd, 2020, pp. 201–228. ISBN: 9780891183716. DOI: 10 . 2134/asaspecpub67 . c12. URL: <https://acsess.onlinelibrary.wiley.com/doi/abs/10.2134/asaspecpub67.c12>.

- [8] Dong-Kyun Kim et al. “Evaluating the relationships between watershed physiography, land use patterns, and phosphorus loading in the bay of Quinte basin, Ontario, Canada”. In: Journal of Great Lakes Research 42.5 (2016), pp. 972–984. ISSN: 0380-1330. DOI: 10.1016/j.jglr.2016.07.008. URL: <https://www.sciencedirect.com/science/article/pii/S0380133016301046>.
- [9] F. Fahmy. “Pollution erodes fish stocks and livelihoods in egyptian lake”. In: (2022). URL: <https://www.reuters.com/world/africa/pollution-erodes-fish-stocks-livelihoods-egyptian-lake-2022-09-01>.
- [10] Council Directive. “91/676/EEC of 12 December 1991 concerning the protection of waters against pollution caused by nitrates from agricultural sources”. In: Official Journal L 375.31 (1991), p. 12.
- [11] K Todo and K Sato. “Directive 2000/60/EC of the European Parliament and of the Council of 23 October 2000 establishing a framework for Community action in the field of water policy”. In: Environmental Research Quarterly (2002), pp. 66–106.
- [12] J. Tzilivakis et al. “A broad-scale spatial analysis of the environmental benefits of fertiliser closed periods implemented under the Nitrates Directive in Europe”. In: Journal of Environmental Management 299 (2021), p. 113674. ISSN: 0301-4797. DOI: 10.1016/j.jenvman.2021.113674. URL: <https://www.sciencedirect.com/science/article/pii/S0301479721017369>.
- [13] Gonzalo Mateo-Garcia et al. “Towards global flood mapping onboard low cost satellites with machine learning”. In: Scientific Reports 11 (Mar. 2021). DOI: 10.1038/s41598-021-86650-z.
- [14] Neal Jean et al. “Combining satellite imagery and machine learning to predict poverty”. In: Science 353.6301 (2016), pp. 790–794. DOI: 10.1126/science.aaf7894. URL: <https://www.science.org/doi/abs/10.1126/science.aaf7894>.
- [15] Preeti Verma and Sunil Patil. “A Machine Learning Approach and Methodology for Solar Radiation Assessment Using Multispectral Satellite Images”. In: Annals of Data Science (Aug. 2021). DOI: 10.1007/s40745-021-00352-x.

- [16] Társilo Girona, Vincent Realmuto, and Paul Lundgren. “Large-scale thermal unrest of volcanoes for years prior to eruption”. In: Nature Geoscience 14 (Apr. 2021). DOI: 10.1038/s41561-021-00705-4.
- [17] Mario Cunha, André Marçal, and Lisa Silva. “Very early prediction of wine yield based on satellite data from vegetation”. In: International Journal of Remote Sensing 31 (Aug. 2013), pp. 3125–3142. DOI: 10.1080/01431160903154382.
- [18] Maxence Dodin et al. “Potential of Sentinel-2 Satellite Images for Monitoring Green Waste Compost and Manure Amendments in Temperate Cropland”. In: Remote Sensing 13.9 (2021). ISSN: 2072-4292. DOI: 10.3390/rs13091616. URL: <https://www.mdpi.com/2072-4292/13/9/1616>.
- [19] Oscar D. Pedrayes et al. “Remote sensing for detecting freshly manured fields”. In: Ecological Informatics 75 (2023), p. 102006. ISSN: 1574-9541. DOI: 10.1016/j.ecoinf.2023.102006. URL: <https://www.sciencedirect.com/science/article/pii/S1574954123000353>.
- [20] Margherita Grandini, Enrico Bagli, and Giorgio Visani. “Metrics for Multi-Class Classification: an Overview”. In: (2020). DOI: 10.48550/arXiv.2008.05756. URL: <https://arxiv.org/abs/2008.05756>.
- [21] Oscar Diaz Pedrayes and Ruben Usamentiaga. “Satellite imagery dataset of manure application on pasture fields”. Version V1. In: (2022). DOI: 10.17632/fbvvvf55kp.1.
- [22] Territory and Civil Protection - Region of Lombardy. Destinazione d'Uso dei Suoli Agricoli e Forestali. URL: <https://www.dati.lombardia.it/Territorio/Dusaf-6-0-Uso-del-suolo-2018/7rae-fng6>.
- [23] F. Amato, F. Dell'Acqua, and D. Marzi. Detection of manure application on crop fields leveraging satellite data and ML. Version 1.0.0. May 2023. URL: <https://github.com/Amatofrancesco99/master-thesis>.
- [24] F. Amato. ee-satellites. URL: <https://pypi.org/project/ee-satellites/>.
- [25] European Space Agency. RapidEye Satellite description. URL: <https://earth.esa.int/eogateway/missions/rapideye>.

- [26] European Space Agency. SkySat Satellite description. URL: <https://earth.esa.int/eogateway/missions/skysat>.
- [27] European Space Agency. Dove Satellite description. URL: <https://earth.esa.int/eogateway/missions/planetscope>.
- [28] National Aeronautics and Space Administration. CloudSat Satellite description. URL: <https://www.jpl.nasa.gov/missions/cloudsat>.
- [29] Rajendra P. Sishodia, Ram L. Ray, and Sudhir K. Singh. “Applications of Remote Sensing in Precision Agriculture: A Review”. In: Remote Sensing 12.19 (2020). ISSN: 2072-4292. DOI: 10.3390/rs12193136. URL: <https://www.mdpi.com/2072-4292/12/19/3136>.
- [30] Sentinel-Hub. “Sentinel-2 RS Indices – Sentinel-Hub Custom Scripts”. In: (2022). Accessed December 6, 2022.
- [31] Nikrooz Bagheri et al. “Multispectral remote sensing for site-specific nitrogen fertilizer management”. In: Pesquisa Agropecuária Brasileira 48 (Oct. 2013), pp. 1394–1401. DOI: 10.1590/S0100-204X2013001000011.
- [32] Yuanyuan Fu et al. “An overview of crop nitrogen status assessment using hyperspectral remote sensing: Current status and perspectives”. In: European Journal of Agronomy 124 (Mar. 2021), p. 126241. DOI: 10.1016/j.eja.2021.126241.
- [33] Qiang Ma, Wantai Yu, and Hua Zhou. “The relationship between soil nutrient properties and remote sensing indices in the Phaeozem region of Northeast China”. In: International Conference on Computational Intelligence and Natural Computing 2 (Sept. 2010). DOI: 10.1109/CINC.2010.5643777.
- [34] Thuan Sarzynski et al. “Combining Radar and Optical Imagery to Map Oil Palm Plantations in Sumatra, Indonesia, Using the Google Earth Engine”. In: Remote Sensing 12 (Apr. 2020). DOI: 10.3390/rs12071220.

- [35] Rouhollah Nasirzadehdizaji et al. “Sensitivity Analysis of Multi-Temporal Sentinel-1 SAR Parameters to Crop Height and Canopy Coverage”. In: Applied Sciences 9.4 (2019). ISSN: 2076-3417. DOI: 10.3390/app9040655. URL: <https://www.mdpi.com/2076-3417/9/4/655>.
- [36] F. Pedregosa et al. ttest_1samp. URL: https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.ttest_1samp.html.
- [37] Nitesh Chawla et al. “SMOTE: Synthetic Minority Over-sampling Technique”. In: J. Artif. Intell. Res. (JAIR) 16 (June 2002), pp. 321–357. DOI: 10.1613/jair.953.
- [38] Roweida Mohammed, Jumanah Rawashdeh, and Malak Abdullah. “Machine Learning with Oversampling and Undersampling Techniques: Overview Study and Experimental Results”. In: (2020), pp. 243–248. DOI: 10.1109/ICICS49469.2020.239556.
- [39] Johar Ashfaque Maat and Amer Iqbal. Hyperparameter tuning and cross-val. URL: <https://subscription.packtpub.com/book/big-data-and-business-intelligence/9781788479042/1/ch01lvl1sec11/hyperparameter-tuning-and-cross-validation>.
- [40] Johar Ashfaque Maat and Amer Iqbal. “Introduction to Support Vector Machines and Kernel Methods”. In: (Apr. 2019).

Acknowledgements

I would like to express my deepest thanks to my esteemed supervisors. Their treasured guidance, mentorship, instructional knowledge and constructive responses have been instrumental in forming my personal development and academic fulfillment.

I would also like to express my gratitude to all teachers of the University of Pavia, for their contribution to my activities. Their determination for excellence, tireless efforts and the availability of resources have created a perfect environment for my intellectual growth.

Moreover, I would love to express my appreciation to all who have performed a position in my academic achievements and helped me to achieve this milestone. Your assist have been fundamental, and so I am really grateful for your contributions.

In conclusion, the biggest thanks go to my family, friends, soulmate and her family, for their constant love, unwavering support and encouragement throughout my instructional journey. Their presence, inspiring phrases and guidance have been a source of energy and motivation to me. I'll never be able to repay you.

Thank you all, from the bottom of my heart.