

Apr 14, 2025

Methods for assessing the temporal dynamics of landscape cover based on procrustean analysis of spectral indices

DOI

<https://dx.doi.org/10.17504/protocols.io.n92ld59k7v5b/v1>

Olha Kunakh¹, Hanna Tutova², Olena Lisovets¹, Olexander Zhukov²

¹Oles Honchar Dnipro National University; ²Bogdan Khmelnytsky Melitopol State Pedagogical University



Olexander Zhukov

Bogdan Khmelnytsky melitopol state pedagogical university

Create & collaborate more with a free account

Edit and publish protocols, collaborate in communities, share insights through comments, and track progress with run records.

Create free account

OPEN  ACCESS



DOI: <https://dx.doi.org/10.17504/protocols.io.n92ld59k7v5b/v1>

Protocol Citation: Olha Kunakh, Hanna Tutova, Olena Lisovets, Olexander Zhukov 2025. Methods for assessing the temporal dynamics of landscape cover based on procrustean analysis of spectral indices. **protocols.io**

<https://dx.doi.org/10.17504/protocols.io.n92ld59k7v5b/v1>

License: This is an open access protocol distributed under the terms of the **Creative Commons Attribution License**, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited

Protocol status: Working

We use this protocol and it's working

Created: April 14, 2025

Last Modified: April 14, 2025

Protocol Integer ID: 126659

Keywords: Sentinel, spectral indices, temporal dynamics, remote sensing, anthropogenic disasters, floodplain ecosystems, various vegetation indice, procrustean analysis of spectral indices comparison, spectral indices comparison, using individual spectral indice, characteristics of vegetation, vegetation cover, dynamics of vegetation cover, individual spectral indice, homogeneous vegetation, other single spectral indice, spectral indice, multidimensionality of the spectral index space, temporal dynamics of landscape cover, vegetation, spectral index space, dimensionality of the spectral index space, principal component analysis, heterogeneous land cover type, results of principal component analysis, landscape cover, complicated landscape complex, comparing different landscape, procrustean analysis, different landscape, state of floodplain ecosystem, physiological state of plant, floodplain ecosystem, such as agricultural land, multidimensionality, assessing dynamic, dimensionality, surface type, using indice, agricultural l

Abstract

Comparisons in different time periods using individual spectral indices are often used to assess the dynamics of vegetation cover. The NDVI is a very popular index for indicating disturbances in the structure of vegetation cover. Various vegetation indices are highly correlated for homogeneous vegetation cover, so assessing dynamics based on one of them is quite appropriate. Similarly, other single spectral indices can be used for other homogeneous surface types, such as agricultural land, rocks, or water. However, in the case of complicated landscape complexes, there is a need to assess the dynamics of heterogeneous land cover types using indices that are sensitive to the characteristics of vegetation, soil and water surface. Anthropogenic pressure also strongly affects the physiological state of plants, so it is also appropriate to use spectral indices that are sensitive to plant health. The multidimensionality of the spectral index space raises the problem of reducing the dimensionality of this space, as well as comparing different landscape states over time. The method uses principal component analysis to reduce the dimensionality of the spectral index space and Procrustes analysis to compare the structures established by the results of principal component analysis. The R code uses as an example an image of the surface of the southern part of Khortytsia Island, where floodplain ecosystems are located, which were negatively affected by the environmental disaster caused by the undermining of the Kakhovka Dam by the Russian occupiers in 2023. Accordingly, the case study compares the state of floodplain ecosystems in 2022 and 2024.

General characteristics of spectral indices

- 1 The Kakhovka Dam was destroyed, resulting in the subsequent devastation of the Kakhovka Reservoir on June 6, 2023. To evaluate the changes in landscape cover in the affected area, we analyzed images of the southern part of Khortytsia Island and the surrounding waters of the Dnipro River, taken one year before the disaster (August 24, 2022) and one year after the disaster (August 18, 2024). We utilized Sentinel-2B Level-2A imagery, which includes 13 spectral channels (B1 – B12, including B8A) and has a resolution of 10 meters: B2 (Blue), B3 (Green), B4 (Red), B8 (NIR); 20 meters: B5–B7, B8A, B11, B12; and 60 meters: B1, B9, B10. For further analysis, the images were reclassified to a resolution of 10 meters. We characterized the landscape cover features using 29 spectral indices (Table 1). The most commonly used channel for calculating spectral indices was B4 (red, 665 nm), which was utilized to compute 17 out of 29 indices, accounting for 58.6%. This channel is essential for classical NDVI-like indices that evaluate green biomass, pigment composition, and vegetation productivity. B8 (NIR, 842 nm) was employed to calculate 15 indices (51.7%). This channel is primarily used for detecting vegetation due to the high reflectivity of plants in the near-infrared range. The B2 (blue, 490 nm) and B3 (green, 560 nm) channels were used to compute 12 (41.4%) and 11 (37.9%) indices, respectively. These channels are important for detecting water stress, pigment imbalance, and spectral contrast. Red-edge channels are also widely utilized: B5 (705 nm) and B7 (783 nm) are included in 11 indices (37.9% each), while B6 (740 nm) is included in 7 indices (24.1%). This underscores the significance of the red-edge range for accurately determining chlorophyll concentration, plant physiological status, and early signs of stress. SWIR channels, which reflect soil moisture, salinity, and surface structural properties, also exhibit a high frequency of use: B11 (1610 nm) was employed in 10 indices (34.5%), and B12 (2190 nm) in 8 indices (27.6%). Less frequently, B1 (coastal aerosol, 443 nm) was used in 3 indices (10.3%), and B8A (narrow NIR, 865 nm) in 2 indices (6.9%). These channels serve highly specialized functions, such as assessing atmospheric impurities (B1) or providing detailed spectral detection of chlorophyll (B8A). The overall structure of channel usage frequency reveals a dominance of indices focused on assessing total phytomass, water stress, moisture, and soil conditions, which are critical for monitoring the state of natural cover and anthropogenically transformed landscapes. The high frequency of use of the B4 (665 nm), B8 (842 nm), B2 (490 nm), and B3 (560 nm) channels highlights the key role of traditional NDVI-like indices, while the inclusion of red-edge (705–783 nm) and SWIR (1610–2190 nm) channels enhances the accuracy of assessments regarding the physiological state of plants and soils.

2

3

Table 1. Spectral Indices for Ecological Analysis

4

A	B	C	D	E
Aerosol Contrast Index	AC_Index	$(B2 - B1) / (B1 + B2)$	This spectral index is applied for detailed examination of coastal and inland waters. It is capable of observing sediments, particles, organic matter and chlorophyll-rich phytoplankton in these waters.	(Komlyk <i>et al.</i> , 2024)
Blue-Green Index	BIG2	$B2 / B3$	BIG2 distinguishes between vegetation and water or soil; sensitive to pigment concentration.	(Elfanah <i>et al.</i> , 2023)
Blue-normalized difference vegetation index	BNDVI	$(B12 - B1) / (B12 + B1)$	The BNDVI is a modification of the standard NDVI, where a blue channel is used instead of the red channel (Red). Its sensitivity is more focused on low chlorophyll levels, which allows it to better detect stressed or degraded plants. The use of the blue channel makes the index more stable in cases of noise pollution or shadows, which is especially important in the early stages of plant growth.	(Yang <i>et al.</i> , 2004)
Green Leaf Index	GLI	$(2 * B3 - B4 - B2) / (2 * B3 + B4 + B2)$	Indicates vegetation density and photosynthetic activity. The index ranges from -1 to +1. A negative value indicates soil or dead cover, while a positive value indicates green leaves and stems. A	(Louhaichi <i>et al.</i> , 2001)



A	B	C	D	E
			threshold with a value close to 0 divides the image into two classes: green leaves and soil or non-living cover.	
Green NDVI	GNDVI	$(B7 - B3) / (B7 + B3)$	Highly sensitive to chlorophyll content; used in vegetation stress monitoring.	(Gitelson <i>et al.</i> , 1996)
Index Name	Abbreviation	Formula	Ecological Meaning	Reference
Land Surface Water Index	LSWI	$(B5 - B6) / (B5 + B6)$	Reflects water content in vegetation and soil; used for drought monitoring.	(Chandrasekar <i>et al.</i> , 2010)
Modified NDWI	MNDW	$(B3 - B11) / (B3 + B11)$	Enhanced water body extraction by replacing NIR with SWIR.	(Xu, 2006)
Normalized Burn Ratio Index	NBRI	$(B8 - B12) / (B8 + B12)$	Detects burned areas and vegetation loss due to fires.	(Seydi <i>et al.</i> , 2021)
Normalized Difference Bareness Index	NDBaI	$(B6 - B11) / (B6 + B11)$	Extracts bare soil areas automatically.	(Zhao and Chen, 2005)
Normalized Difference Chlorophyll-a	NDChl _a	$(B3 - B4) / (B3 + B4)$	Estimates chlorophyll-a concentration in water.	(Ha <i>et al.</i> , 2017)
Normalized Difference Green	NDGCI	$(B8a - B3) / (B8a + B3)$	Estimates leaf chlorophyll content, indicator of plant health.	(Chaves <i>et al.</i> , 2020)

	A	B	C	D	E
	Chlorophyll Index				
	Normalized Difference Index	NDI	$(B_{12} - B_7) / (B_{12} + B_7)$	Used to estimate soil salinity.	(Wang <i>et al.</i> , 2019)
	Normalized Difference Infrared Index	NDII	$(B_8 - B_{11}) / (B_8 + B_{11})$	Indicates water content in vegetation; used for drought assessment.	(Hardisky <i>et al.</i> , 1983)
	Normalized Difference Iron Oxide	NDIO	$(B_4 - B_2) / (B_2 + B_4)$	Detection of ferric iron oxides.	(Yazdi <i>et al.</i> , 2013)
	Normalized Difference Red Edge Index	NDRE	$(B_7 - B_5) / (B_7 + B_5)$	NDRE is a modified version of NDVI that uses red-edge instead of red band. Red-edge is a spectral region between red and NIR (~700-740 nm) that is sensitive to chlorophyll content, even in dense vegetation. The index does not saturate at high biomass, unlike NDVI. It is a better indicator of chlorophyll, especially in the later stages of crop growth, when NDVI is no longer sensitive. Sensitive to nitrogen stress - as chlorophyll content is often directly related to nitrogen content in plants. It is used to assess stress associated with water and nitrogen deficiencies. Reliable for partial soil coverage - less dependent on cover density than other indices. It has a high correlation with the nitrogen content in the	(Barnes <i>et al.</i> , 2000)



A	B	C	D	E
			leaves, surpassing other vegetation indices in this regard.	
Normalized Difference Red-Edge Improved	NDREI	$(B8 - B5) / (B8 + B5)$	NDREI is an improved version of the NDRE index, which uses RE1 (B6 or B7) instead of RE2 (B5), as well as near-infrared (NIR). The choice of RE1 + NIR provides better spectral sensitivity to early changes in chlorophyll and plant condition. The index reduces saturation, which is typical for NDVI, and responds better to light plant stress, even with a high leaf area index (LAI). The index is highly sensitive to chlorophyll concentration. NDREI accurately reflects chlorophyll even in dense cover. It is optimal for assessing nitrogen supply. It is particularly effective in crops where nitrogen is closely related to chlorophyll content. Less dependent on cover density than NDVI or GNDVI. Increased stability when changing the sun angle or atmospheric conditions due to the use of a red-edge channel closer to the NIR. Effective for monitoring late vegetation stages when NDVI becomes less sensitive.	(Barnes <i>et al.</i> , 2000)
Normalized Difference Tillage Index	NDTI	$(B11 - B12) / (B11 + B12)$	The index uses two SWIR channels that are sensitive to soil moisture content and surface structure. The NDTI index detects the difference between treated and untreated soil based on their different reflectivity in the SWIR range. Changes in the surface structure (e.g. after ploughing or harrowing) affect the spectral response that this index captures. High ability to	(Van Deventer <i>et al.</i> , 1997)

	A	B	C	D	E
				distinguish between tillage methods, e.g. no-till, minimum-till, conventional tillage. It is resistant to changes in vegetation cover and is particularly effective in areas with no or minimal vegetation. It can be used to assess erosion vulnerability, as the surface structure directly affects erosion processes.	
	Normalized Difference Total Suspended Matter Index	NDTSM	$(B7 - B2) / (B7 + B2)$	Estimates total suspended matter in water bodies.	(Premkumar <i>et al.</i> , 2021)
	Normalized Difference Vegetation Index	NDVI	$(B8 - B4) / (B8 + B4)$	Indicates vegetation productivity and greenness. One of the most popular vegetation indices.	(Rouse <i>et al.</i> , 1974)
	Normalized Difference Water Index 1	NDWI1	$(B3 - B8) / (B3 + B8)$	NDWI1 exploits the difference in spectral reflectivity between water and land in the green and NIR bands. Positive values (close to +1): indicate open water. Zero or negative values are typical for vegetation and soils. The main application is to identify lakes, rivers, and reservoirs in images. NDWI1 can be an indicator of overall turbidity.	(McFeeters, 1996)
	Normalized Difference Water Index 2	NDWI ₂	$(B5 - B12) / (B5 + B12)$	NDWI ₂ is sensitive to the moisture content of soil and vegetation. It is primarily applied to assess water stress in plants and soil moisture levels. Low NDWI ₂ values indicate drought or reduced moisture. The NDWI ₂ is	(Gašparović and Singh, 2020)



A	B	C	D	E
			not intended to detect water as much as the classic NDWI ₁ , but to indicate the moisture in the environment. NDWI ₂ is often used under the name NDMI (Normalised Difference Moisture Index).	
Red Edge Difference Index	REDI	$(B7 - B6) / (B7 + B6)$	The index is highly sensitive to changes in chlorophyll. The index compares two adjacent red-edge bands, where subtle changes in the slope of the spectral curve are recorded as chlorophyll content increases or decreases. The index also has low sensitivity to soil and water background. A narrow spectral range between channels B6 and B7 minimises the influence of background components, particularly in the case of incomplete vegetation coverage. In dense vegetation, this index is less prone to saturation than classical vegetation indices. It can be used for high-precision analyses of the dynamics of changes in vegetation condition due to its spectral stability. The index is sensitive to changes in chlorophyll content, so it is efficient for detecting nutritional stress, aging, and pest damage.	(Addabbo <i>et al.</i> , 2016)
Red Edge NDVI 1	RedEdge_NDVI1	$(B6 - B4) / (B6 + B4)$	This index is a variant of NDVI, where instead of NIR, red-edge, which is sensitive to chlorophyll, is used, and instead of the traditional green, red, which is highly absorbed by pigments, is used. The red-edge range (680-750 nm) is characterised by a sharp transition between chlorophyll absorption and mesophyll	(Xie <i>et al.</i> , 2018)



A	B	C	D	E
			reflection. B6 is particularly sensitive to early changes in chlorophyll, making it useful for detecting subtle or initial changes in vegetation. Red-edge reflectance is predominantly driven by chlorophyll content, not just leaf area, which allows for an assessment of the physiological state of plants. Red Edge NDVI 1 displays LAI changes much more accurately than classic NDVI. It allows you to identify the stages of plant development, initial signs of stress, leaf aging or nutrient deficiencies.	
Red Edge NDVI 2	RedEdge_NDVI2	$(B7 - B4) / (B7 + B4)$	This index is a modified NDVI based solely on the red-edge spectrum. It can be considered a highly specialised index for assessing the physiological state of plants, in particular chlorophyll content. The index is particularly sensitive to chlorophyll concentration in leaves and less sensitive to soil background. Even small changes in the structure or concentration of pigments can significantly affect the index value. Red-edge NDVI 2 does not reach a plateau as quickly at high LAI values, which allows for better distinction of dense crowns. Relatively less dependent on water content: this index focuses more specifically on photosynthetic pigments.	(Xie <i>et al.</i> , 2018)
Red Edge Normalized Difference Vegetati	RENDVI	$(B7 - B5) / (B7 + B5)$	Unlike the classical NDVI, which is based on the red and NIR bands, RENDVI uses two red-edge channels, which gives high sensitivity to chlorophyll concentration	(Gitelson <i>et al.</i> , 1996)

	A	B	C	D	E
	on Index			without saturation under high LAI conditions. The index is particularly effective in detecting plant stress, nitrogen deficiency, and aging. Since the reflections in the red-edge are formed mainly by leaves, the index is less sensitive to the brightness of the litter layer than the classical indices. It shows changes in chlorophyll content even under early stress, before visual symptoms appear. Suitable for analysing the dynamics of medium and dense vegetation in time series.	
	Red-Blue NDVI	RBND VI	$\frac{(B8 - (B5 + B1))}{(B8 + B5 + B1)}$	A three-component index derived from NDVI can better assess mixed vegetation and soil conditions. The B1 (coastal aerosol) channel is sensitive to aerosols and dust, so its inclusion in the denominator allows for normalisation of the effect of atmospheric influence on reflectance. The B5 red-edge channel provides a better assessment of vegetation stress conditions, including early signs of wilting, dryness, or salinity. The integration of information on soil, vegetation and atmosphere makes the index more stable in arid and semi-arid regions, where other indices have a lower correlation with real indicators of land cover. It is used for monitoring soil salinity, assessing vegetation cover in saline environments, and has improved discriminatory ability to distinguish between vegetation, bare soil and saline areas.	(Wang <i>et al.</i> , 2019)



A	B	C	D	E
Rock Index	RI	$\frac{(B3 - B12)}{(B3 + B12)}$	Index based on the contrast of visible and SWIR reflectance. The green band (B3) is well reflected by most illuminated surfaces (including rocks), while the SWIR band (B12) is sensitive to moisture, clay minerals, carbonates and sulphates. Rocks with high reflectivity in the visible spectrum but low SWIR reflectivity (quartzite, flint) give high positive values. Mixed moistened clay and altered rocks (with hydrothermal influence) exhibit lower or negative values. The index is analogous to NDVI for geological purposes. However, RI characterises rock surface types instead of vegetation. It is used to identify rock outcrops, especially in open, dry regions. High RI values (closer to +1) indicate dry, reflective surfaces (light rocks, quartzites). Low RI values (closer to -1) may indicate wet, hydrothermally altered areas or dense vegetation.	(Imbroane <i>et al.</i> , 2007)



A	B	C	D	E
Sentinel-2 Vegetation Salinity Index	SVSI	$(B4 - B2) / (B5 + B11)$	The index assesses the difference between the visual absorption of pigments (chlorophyll, carotenoids) in the blue/red range (B2, B4) and the combined contribution of leaf structure and water (B5 + B11). The index reveals the secondary effects of soil salinity primarily through a decrease in chlorophyll, a decrease in water content, and damage to leaf structure. Indicates the accumulation of Cl^- and Na^+ ions in plant leaves through their spectral effect on pigment composition and moisture content. It allows to assess the level of plant stress in saline areas.	(Lugassi <i>et al.</i> , 2017)
Structure Insensitive Pigment Index	SIPI	$(B8 - B2) / (B8 - B4)$	SIPI is developed to minimise the influence of structural characteristics of the plant cover (e.g. leaf area, leaf distribution) on the assessment of plant pigment composition. SIPI increases with decreasing relative chlorophyll content and increasing carotenoids, and is therefore a marker of stress, aging, and leaf yellowing. $SIPI \approx 1.0$ indicates a high chlorophyll content and low stress level. $SIPI > 1.2$ indicates an increase in the proportion of carotenoids, possible degradation or stress. SIPI effectively reflects changes in the pigment balance due to water deficiency, pathogens, and nutrient deficiencies. It is used to monitor the degradation of stands in forests and parks.	(Penuelas <i>et al.</i> , 1995)

Vegetation indexes

- 5 The group of indices, including NDVI, GNDVI, GLI, NDGCI, NDRE, RedEdge NDVI1/2, and RENDVI, effectively reflect both quantitative (density, Leaf Area Index [LAI]) and qualitative (pigment content, physiological state of plants) aspects of vegetation cover. Their application enables the monitoring of productivity, stress, and degradation of phytomass over extensive areas, particularly exemplified by the landscape changes following the destruction of the Kakhovka reservoir.

Spectral indices are sensitive to soil cover properties

- 6 Among the spectral indices utilized in this study, several specialized indices enable to characterize the properties of soil cover, including its bareness, surface structure, moisture content, salinity, and iron oxide levels. Notably, the Normalized Difference Bare Soil Index (NDBaI) automatically identifies areas with bare soil, serving as an indicator of vegetation degradation or active erosion processes. The Normalized Difference Tillage Index (NDTI), which employs two shortwave infrared (SWIR) channels, is sensitive to moisture content and changes in micro-relief. This sensitivity allows it to differentiate between treated and untreated soil and to identify structures that increase the risk of erosion. The NDI index reflects the degree of salinity at the soil surface, which is particularly important for monitoring degraded areas or those transformed by irrigation. The Rock Index (RI) differentiates between surface types based on the ratio of visible to shortwave infrared (SWIR) reflectance, which can be useful for identifying mineral differences, especially in arid and open regions. The NDIO index characterizes the presence of ferrous oxides, enabling the assessment of soil chemistry and signs of hydrothermal changes. The NDWI2, NDII, and LSWI indices, although traditionally used to evaluate the moisture content of vegetation, are also sensitive to soil moisture, particularly in bare areas where infrared wavelengths dominate the spectral response. The selected indices facilitate a comprehensive assessment of soil cover, considering its physical structure, moisture content, chemical composition, and degree of bareness. This is critical for detecting degradation processes, spatial heterogeneity, and evaluating anthropogenic impacts on ecosystems.

Spectral indices are sensitive to the properties of the water environment

- 7 Among the spectral indices utilized in this study, several indicators are crucial for characterizing the ecological conditions of the aquatic environment, as they reflect both the presence of open water and the quality of aquatic vegetation cover. These indices include the Normalized

Difference Water Index 1 (NDWI1), which facilitates the identification of water surfaces by contrasting the green and near-infrared spectral ranges. Its modification, the Modified Normalized Difference Water Index (MNDWI), enhances water separation in the presence of aquatic vegetation or suspended solids by utilizing the Short-Wave Infrared (SWIR) channel. Concurrently, the Land Surface Water Index (LSWI), the Normalized Difference Infrared Index (NDII), and NDWI2 (also known as the Normalized Difference Moisture Index, NDMI) are sensitive to the moisture content of vegetation and soil, making them widely used for assessing water stress levels, particularly in agroecosystems and wetlands. Additionally, the NDTSM facilitates the estimation of suspended solids concentration in water, a crucial indicator of turbidity and anthropogenic load. The NDChla index is employed to approximate chlorophyll-a content, which may signify levels of eutrophication or substantial phytoplankton development. A distinct category encompasses indices related to saline environmental conditions, such as SVSI and NDI, which reflect the secondary effects of salinity through reductions in pigment content and alterations in vegetation structure. Another significant index is the RBNDVI, a three-component metric that integrates the characteristics of atmospheric exposure, vegetation, and soil salinity. Finally, the AC_Index enables the estimation of aerosol and organic matter content in coastal waters, which is essential for pollution detection. Thus, spectral indices designed to assess the aquatic environment provide a comprehensive overview of its condition, including humidity, the presence of water bodies, pollution levels, eutrophication, and salinity. This information is vital for monitoring changes in the context of hydrological and anthropogenic transformations.

Statistical analysis features

- 8 Spectral indices for reducing the dimensionality of the trait space were subjected to principal component analysis (Eastman and Fulk, 1993). Primary principal component analysis (PCA) typically identifies dominant patterns that reveal the most significant features of landscape cover structure, such as the ratio of vegetated to non-vegetated areas and the ratio of land to water. These land cover types are markedly distinct from one another and occupy a substantial portion of the imagery. When land cover changes are not widespread—such as forest fires or the complete destruction of water bodies—general patterns of landscape structure dominate the variability in the feature space, obscuring smaller-scale land cover changes. To detect these smaller-scale spatial patterns, the variability associated with principal components 1 and 2 was extracted from the spectral index values. This was accomplished by calculating linear regressions of the spectral indices against PC 1 and PC 2. The residuals from these regression models were then subjected to secondary principal component analysis. The results of the PCA on the residuals were further

analyzed using Procrustes analysis with the use of the vegan library (Oksanen et al., 2022). The procrustes function from this library rotates one configuration to the maximum similarity with another configuration minimizing sum of squared differences. The protest function tests for nonrandomness (significance) between two configurations. Procrustes rotation is typically used in comparison of ordination results. In this case, the solutions obtained from the analysis of the principal components of the residuals of the regression dependencies of spectral indices on the primary principal components are compared.

Code for calculations in Project R

9

Software

R programming language	NAME
The R Foundation	DEVELOPER
Comprehensive R Archive Network	REPOSITORY
https://cran.r-project.org/	SOURCE LINK

```
# Завантаження необхідних пакетів
# Loading necessary packages
library(terra) # робота з растровими даними | for working with raster data
library(sf)    # робота з векторними даними | for working with vector data

# 1. Завантаження полігону (shapefile)
# 1.
Loading the polygon (shapefile)
poly ←
st_read("D:/Science/Хортиця/Form_3.shp")
poly_vect
← vect(poly)

# 2. Задаємо шлях до SAFE-продукту Sentinel-2
# 2. Define
the path to the Sentinel-2 SAFE product
```



```
safe_folder  
← "D:/GIS  
Data/SENTINEL/S2B_MSIL2A_20220824T083559_N0400_R064_T36UXU_20220824T10  
0829.SAFE"
```

```
# 3. Отримання списку всіх файлів з розширенням .jp2  
(рекурсивно)
```

```
# 3.
```

```
Retrieve a list of all files with the .jp2 extension (recursively)
```

```
all_files
```

```
← list.files(safe_folder, pattern = "\\.*jp2$", recursive = TRUE,  
full.names = TRUE)
```

```
cat("Знайдено JP2 файлів (без фільтрації):",
```

```
length(all_files), "\n") #
```

```
Print count of found JP2 files without filtering
```

```
# Відфільтруємо файли, що містять "IMG_DATA" в  
шляху (як правило, саме там знаходяться потрібні канали)
```

```
# Filter
```

```
the files that contain "IMG_DATA" in their path (usually where the  
required bands are located)
```

```
file_paths
```

```
← all_files[grepl("IMG_DATA", all_files)]
```

```
cat("Знайдено файлів у
```

```
IMG_DATA:", length(file_paths), "\n")
```

```
print(file_paths)
```

```
if(length(file_paths)
```

```
== 0) {
```

```
  stop("Не знайдено файлів .jp2 у підкаталозі IMG_DATA. Перевірте шлях та  
pattern.")
```

```
  # Stop execution if no .jp2 files are found
```

```
  in IMG_DATA folder
```

```
}
```

```
# 4. Встановлюємо рік зйомки (для іменування файлів)
```

```
# 4. Set
```

```
the image acquisition year (for file naming)
```

```
year_of_image
```

```
← "2022"
```

```
# 5. Встановлюємо вихідну папку та створюємо її, якщо неіснує
```

```
# 5.
Specify the output folder and create it if it does not exist
output_folder
← "D:/GIS Data/SENTINEL/Output"
if
(!dir.exists(output_folder)) {
  dir.create(output_folder, recursive = TRUE)
}

# 6. Цикл обробки файлів
# 6. Loop
for processing files
for(file_path
in file_paths) {
  cat("Обробляється файл:", file_path, "\n")
# Processing the file

  # Завантаження зображення
  # Load the image
  r_band ← try(rast(file_path), silent =
TRUE)
  if(inherits(r_band, "try-error")){
    cat("Не вдалося завантажити
файл:", file_path, "\n")
    next
  }

  # Обрізання
зображення за межами полігону
  # Crop the image to the polygon boundaries
  r_crop ← try(crop(r_band, poly_vect),
silent = TRUE)
  if(inherits(r_crop, "try-error")){
    cat("Помилка crop для:", file_path, "\n")
    next
  }
  r_mask ← try(mask(r_crop, poly_vect),
silent = TRUE)
  if(inherits(r_mask, "try-error")){
    cat("Помилка mask для:", file_path, "\n")
    next
  }
}
```

```
# Перевірка роздільної здатності
# Check the current resolution
current_res ← res(r_mask)
cat("Поточна роздільна здатність:",
current_res[1], "x", current_res[2], "\n")

# Ресемплінг до 10 метрів, якщо потрібно
# Resample to 10 meters resolution if needed
if(any(current_res != 10)) {
  ext_r ← ext(r_mask)
  ncol_new ← ceiling((ext_r[2] -
ext_r[1]) / 10)
  nrow_new ← ceiling((ext_r[4] -
ext_r[3]) / 10)
  target ← rast(ext = ext_r, ncols =
ncol_new, nrows = nrow_new, crs = crs(r_mask))
  r_final ← try(resample(r_mask, target,
method = "bilinear"), silent = TRUE)
  if(inherits(r_final,
"try-error")){
    cat("Помилка resample для:", file_path, "\n")
    next
  }
} else {
  r_final ← r_mask
}

# Формування імені файлу: використання базового імені + рік
# Form the file name using the base name and
the year
base_name ←
tools::file_path_sans_ext(basename(file_path))
output_path ← file.path(output_folder,
paste0(base_name, "_", year_of_image, ".tif"))

# Запис результату
# Write the result raster to file
tryCatch({
  writeRaster(r_final, output_path, overwrite
= TRUE)
  cat("Збережено:", output_path, "\n\n")
}
```

```
    }, error = function(e){
      cat("Помилка запису для:",
file_path, "\n", e, "\n")
    })
  }

cat("Обробку завершено!\n")
# Processing complete!

# Розрахунок індексів
#
Calculation of indices
library(terra) # Ensure terra is loaded (if not already)

year =
"2024"

# === 1. ШЛЯХИ ===
# === 1.
PATHS ===
input_folder
← "D:/GIS Data/SENTINEL/Output"
output_folder
← paste0("D:/GIS Data/SENTINEL/Indices_", year)
dir.create(output_folder,
showWarnings = FALSE, recursive = TRUE)

# === 2. КАНАЛИ ===
# === 2.
BANDS ===
bands_codes
← c(
  B1 =
"B01",
  B2 =
"B02",
  B3 =
"B03",
  B4 =
"B04",
  B5 =
"B05",
```

```
B6 =  
"B06",  
B7 =  
"B07",  
B8 =  
"B08",  
B8A = "B8A",  
B9  
= "B09",  
B11 =  
"B11",  
B12 =  
"B12"  
)  
  
# === 3. ПОШУК ФАЙЛУ ЗА КАНАЛОМ ===  
# === 3.  
SEARCH FOR FILE BY BAND ===  
get_input_band  
← function(band_code, year = year) {  
  files ← list.files(input_folder, pattern  
= paste0(band_code, ".*", year, "\\tif$"), full.names =  
TRUE)  
  if (length(files) == 0) {  
    warning(paste("Не знайдено файл для", band_code, "за", year))  
    return(NA)  
  }  
  return(files[1])  
}  
  
# === 4. ЗАВАНТАЖЕННЯ КАНАЛІВ ===  
# === 4.  
LOADING BANDS ===  
bands ←  
list()  
for (b in  
names(bands_codes)) {  
  cat("Завантаження", b, "\n") # Loading band  
  path ← get_input_band(bands_codes[b],  
year = year)  
  if (!is.na(path)) {  
    bands[[b]] ← rast(path)
```

```
cat("
✓", basename(path),
"\n")
} else {
cat("
! Пропущено\n")
}
}
# === 5. БАЗОВЕ ІМ'Я ===
# === 5.
SET BASE NAME ===
b2_file
← get_input_band("B02", year = year)
if
(is.na(b2_file)) stop("Не знайдено B02 для іменування індексів")
base_name
← tools::file_path_sans_ext(basename(b2_file))
year_of_image
← year

# === 5.1 ПРИВЕДЕННЯ ДОШАБЛОНУ B2 ===
# === 5.1
ALIGN TO B2 TEMPLATE ===
if
(!"B2" %in% names(bands)) stop("Канал B2 відсутній — шаблон неможливий")

template
← bands$B2
for (b in
names(bands)) {
if (b == "B2") next
if (!inherits(bands[[b]],
"SpatRaster")) next
r ← bands[[b]]
if (!compareGeom(r, template, stopOnError =
FALSE)) {
cat("
Приведення до шаблону:", b, "\n")
bands[[b]] ← resample(r, template, method =
"bilinear")
}
}
```

```
# === ЗБЕРЕЖЕННЯ ІНДЕКСІВ (тільки аббревіатура + рік) ===
# ===
SAVING INDICES (using abbreviation + year) ===
save_index
← function(r, abbr) {
  out_name ← paste0(abbr, "_",
year_of_image, ".tif")
  out_path ← file.path(output_folder,
out_name)
  writeRaster(r, out_path, overwrite = TRUE)
  cat("📁 Збережено:", out_name, "\n")
}

# === 7. РОЗРАХУНОК ІНДЕКСУ З ПЕРЕВІРКОЮ ===
# === 7.
CALCULATE INDEX WITH CHECK ===
try_index
← function(expr, required_bands, abbr) {
  if (all(required_bands %in% names(bands))
&&
  all(sapply(bands[required_bands],
function(x) inherits(x, "SpatRaster")))) {
    result ← tryCatch(eval(expr), error =
function(e) NA)
    if (inherits(result,
"SpatRaster")) {
      save_index(result, abbr)
    } else {
      cat("⚠️ Індекс", abbr, "не обчислено (результат не SpatRaster)\n")
    }
  } else {
    cat("⚠️ Індекс",
abbr, "пропущено (немає каналів:", paste(setdiff(required_bands,
names(bands)), collapse = ", ", "\n")
  )
}
}

# === 8. РОЗРАХУНОК ІНДЕКСІВ ===
# === 8.
CALCULATE INDICES ===
try_index(quote((bands$B1
- bands$B2)/(bands$B1 + bands$B2)), c("B1", "B2"),
"AC_Index")
```

```
try_index(quote((bands$B4
- bands$B2)/(bands$B2 + bands$B4)), c("B2", "B4"),
"NDIO")
try_index(quote((bands$B3
- bands$B12)/(bands$B3 + bands$B12)), c("B3", "B12"),
"RI")
try_index(quote((bands$B8A
- bands$B3)/(bands$B8A + bands$B3)), c("B3", "B8A"),
"NDGCI")
try_index(quote((bands$B3
- bands$B2)/(bands$B2 + bands$B3)), c("B2", "B3"),
"BIG2")
try_index(quote((bands$B8
- bands$B2)/(bands$B8 + bands$B4)), c("B2", "B8"),
"SIPI")
try_index(quote((bands$B4
- bands$B2)/(bands$B5 + bands$B11)), c("B4", "B2",
"B5", "B11"), "SVSI")
try_index(quote((bands$B8
- bands$B5)/(bands$B8 + bands$B5)), c("B8", "B5"),
"NDREI")
try_index(quote((bands$B7
- bands$B5)/(bands$B7 + bands$B5)), c("B7", "B6"),
"NDRE")
try_index(quote((bands$B6
- bands$B11)/(bands$B6 + bands$B11)), c("B6", "B11"),
"NDBal")
try_index(quote((bands$B11
- bands$B12)/(bands$B11 + bands$B12)), c("B11", "B12"),
"NDTI")
try_index(quote((bands$B8
- bands$B11)/(bands$B8 + bands$B11)), c("B8", "B11"),
"NDII")
try_index(quote((bands$B8
- bands$B12)/(bands$B8 + bands$B12)), c("B8", "B12"),
"NBRI")
try_index(quote((bands$B8
- bands$B4)/(bands$B8 + bands$B4)), c("B8", "B4"),
"NDVI")
try_index(quote((bands$B6
- bands$B4)/(bands$B6 + bands$B4)), c("B6", "B4"),
"RedEdge_NDVI1")
```



```
try_index(quote((bands$B7
- bands$B4)/(bands$B7 + bands$B4)), c("B7", "B4"),
"RedEdge_NDVI2")
try_index(quote((bands$B7
- bands$B3)/(bands$B7 + bands$B3)), c("B7", "B3"),
"GNDVI")
try_index(quote((bands$B7
- bands$B2)/(bands$B7 + bands$B2)), c("B7", "B2"),
"NDTSM")
try_index(quote((bands$B6
- bands$B5)/(bands$B6 + bands$B5)), c("B6", "B5"),
"RENDVI")
try_index(quote((bands$B12
- bands$B7)/(bands$B12 + bands$B7)), c("B12", "B7"),
"NDI")
try_index(quote((bands$B7
- bands$B6)/(bands$B6 + bands$B7)), c("B6", "B7"),
"REDI")
try_index(quote((bands$B3
- bands$B11)/(bands$B3 + bands$B11)), c("B3", "B11"),
"MNDW")
try_index(quote((bands$B3
- bands$B8)/(bands$B3 + bands$B8)), c("B3", "B8"),
"NDWI1")
try_index(quote((bands$B5
- bands$B12)/(bands$B5 + bands$B12)), c("B5", "B12"),
"NDWI2")
try_index(quote((bands$B8A
- bands$B12)/(bands$B8A + bands$B12)), c("B8A", "B12"),
"LSWI")
try_index(quote((bands$B8
- (bands$B5 + bands$B1))/(bands$B8 + bands$B5 + bands$B1)), c("B1",
"B5", "B9"), "RBNDVI")
try_index(quote((bands$B3
- bands$B4)/(bands$B3 + bands$B4)), c("B3", "B4"),
"NDChla")
try_index(quote((bands$B12
- bands$B1)/(bands$B12 + bands$B1)), c("B1", "B12"),
"NDNIRBlue")
try_index(quote((2*bands$B3
- bands$B4 - bands$B2)/(2*bands$B3 + bands$B4 + bands$B2)), c("B2",
"B3", "B4"), "GLI")
```

```
cat("✅ Усі доступні індекси обчислено
та збережено!\n")
# All
available indices have been calculated and saved!
library(terra)
library(vegan)

# === 1. Встановлюємо директорії та отримуємо спільні індекси ===
# === 1.
Set directories and obtain common indices ===
dir_2022
← "D:/GIS Data/SENTINEL/Indices_2022"
dir_2024
← "D:/GIS Data/SENTINEL/Indices_2024"

get_index_name
← function(path) {
  gsub("_\\d{4}\\\\.tif$",
"", basename(path))
}

files_2022
← list.files(dir_2022, pattern = "\\\\.tif$", full.names = TRUE)
files_2024
← list.files(dir_2024, pattern = "\\\\.tif$", full.names = TRUE)

indices_2022
← get_index_name(files_2022)
indices_2024
← get_index_name(files_2024)
common_indices
← intersect(indices_2022, indices_2024)
cat("✅ Спільні індекси:",
paste(common_indices, collapse = ", ", "\n")
# Print
common indices available for both years

# === 2. Завантаження шаблонного растру (наприклад,
AC_Index_2022.tif) ===
# === 2.
Loading template raster (e.g., AC_Index_2022.tif) ===
template_path
← "D:/GIS Data/SENTINEL/Indices_2022/AC_Index_2022.tif"
```

```
template <- rast(template_path)

# === 3. Функція для перетворення растрових даних у матрицю ===
# === 3.
Function to convert raster data into a matrix ===
rast_to_matrix
<- function(file_list, index_names, template) {
  mat <- list()
  for (i in seq_along(index_names)) {
    fname <- file_list[grepl(index_names[i],
file_list)]
    if (length(fname) == 0) next
    r <- rast(fname[1])
    if (!compareGeom(r, template, stopOnError =
FALSE)) {
      r <- resample(r, template, method =
"bilinear")
    }
    mat[[index_names[i]]] <- values(r)
  }
  return(do.call(cbind, mat))
}

mat_2022
<- rast_to_matrix(files_2022, common_indices, template)
mat_2024
<- rast_to_matrix(files_2024, common_indices, template)

# === 4. Видалення NA та Inf (формуємо підмножину пікселів) ===
# === 4.
Removing NA and Inf (forming a subset of pixels) ===
bad_rows
<- apply(mat_2022, 1, function(x) any(is.na(x) | is.infinite(x))) |
  apply(mat_2024, 1, function(x)
any(is.na(x) | is.infinite(x)))
mat_2022_clean
<- mat_2022[!bad_rows, ]
mat_2024_clean
<- mat_2024[!bad_rows, ]
colnames(mat_2022_clean)
<- common_indices
colnames(mat_2024_clean)
<- common_indices
```

```
cat("🧹 Залишено пікселів:",  
nrow(mat_2022_clean), "з", nrow(mat_2022), "\n")  
# Report  
number of pixels kept after cleaning  
# Запам'ятовуємо індекси використаних клітин (у порядку  
шаблону)  
# Save  
indices of used cells (according to template order)  
sel ←  
which(!bad_rows) # ці індекси відповідають рядкам у mat_2022_clean  
  
X_2022  
← mat_2022_clean  
X_2024  
← mat_2024_clean  
  
# --- 2.  
PCA для кожного року ---  
# --- 2.  
PCA for each year ---  
pca_2022  
← rda(X_2022, scale = TRUE)  
pca_2024  
← rda(X_2024, scale = TRUE)  
  
# --- 3. Витяг PC1 ---  
# --- 3.  
Extract PC1 (first two principal components) ---  
pc_2022  
← scores(pca_2022, display = "sites", choices = c(1,2))[, 1:2]  
pc_2024  
← scores(pca_2024, display = "sites", choices = c(1,2))[, 1:2]  
  
# === 7.  
Procrustes аналіз (вирівнюємо 2024 до 2022) ===  
# === 7.  
Procrustes analysis (align 2024 to 2022) ===  
proc ←  
procrustes(pc_2022, pc_2024, scale = FALSE, symmetric = TRUE)  
  
# === 5. Перестановковий тест  
# === 5.  
Permutation test ===
```

```
proc_test
← protest(pc_2022, pc_2024, permutations = 999)

Yhat ← proc$Yrot
# вирівняні координати для 2024 (підмножина, для тих клітин, що
використовувались)
# Yhat
contains aligned coordinates for 2024 for the subset of used pixels

# === 8. Обчислюємо залишки (деформації) для підмножини
пикселів ===
# === 8.
Calculate residuals (deformations) for the pixel subset ===
resid_PC1_subset
← Yhat[, 1] - pc_2022[, 1]
resid_PC2_subset
← Yhat[, 2] - pc_2022[, 2]

# === 9. Створюємо повні вектори залишків для всіх клітин
шаблону ===
# === 9.
Create full residual vectors for all template cells ===
n_cells
← ncell(template)
resid_PC1_full
← rep(NA, n_cells)
resid_PC2_full
← rep(NA, n_cells)
resid_PC1_full[sel]
← resid_PC1_subset
resid_PC2_full[sel]
← resid_PC2_subset

# === 10. Створення растрових шарів з залишками за шаблоном
===
# === 10.
Create raster layers of residuals based on the template ===
r_PC1 ←
setValues(template, resid_PC1_full)
r_PC2 ←
setValues(template, resid_PC2_full)

# Переконаємося, що CRS збережено
```

```
# Ensure
that the CRS is maintained
crs(r_PC1)
← crs(template)
crs(r_PC2)
← crs(template)

# === 11. Візуалізація карти прокрустових деформацій ===
# === 11.
Visualize the Procrustes deformation map ===
windowsFonts(Times
= windowsFont("Times New Roman"))
par(family
= "Times", mar = c(3,3,0.5,0.5), mfrow = c(1, 2))
plot(r_PC1,
main = "PC1", col = hcl.colors(100, "Spectral"))
plot(r_PC2,
main = "PC2", col = hcl.colors(100, "Spectral"))
# === 12. (Опціонально) Збереження результатів ===
# === 12.
(Optional) Save the results ===
writeRaster(r_PC1,
"D:/GIS Data/SENTINEL/Output/proc_deform_PC1.tif", overwrite = TRUE)
writeRaster(r_PC2,
"D:/GIS Data/SENTINEL/Output/proc_deform_PC2.tif", overwrite = TRUE)

# Завантажуємо векторні дані з точками з shapefile
# Load
vector data of points from a shapefile
points
← vect("D:/Science/Хортиця/Plants_Total.shp")

# Об'єднуємо растрові шари в один стек.
# Combine
the raster layers into one stack.
rasters_stack
← c(r_PC1, r_PC2, r_PC3, r_PC4)

# Екстрагуємо значення для кожного пункту. Функція повертає
data.frame,
# де перший стовпець – ідентифікатор точки, а інші –
значення відповідно до входжень у кожен шар.
```

```
# Extract
values for each point. The function returns a data.frame where the first column
is the point ID and the others are the values from each layer.
extracted_values
← extract(rasters_stack, points)

# Переглянемо отримані дані
# Display
the extracted data
print(head(extracted_values))

# Припускаємо, що extracted_values містить 5 стовпців, перший — "ID"
# та 4 стовпці з даними
# Assume
that extracted_values contains 5 columns: the first being "ID" and
the next 4 the data values.
names(extracted_values)
← c("ID", "PC1", "PC2", "PC3",
"PC4")

# Перевіряємо, як виглядають перші рядки
# Check the
first few rows
head(extracted_values)

# Зберігаємо data.frame у файл (CSV, наприклад)
# Save the
data.frame to a file (e.g., CSV)
write.csv(extracted_values,
"D:/Science/Хортиця/Plants_Total_extracted.csv",
row.names = FALSE)

#####
#####
# Додатковий код для PCA аналізу залишків та прокрустової
оцінки
#####
#####

# --- 4. Вилучення PC1 з кожної змінної ---
# --- 4.
Extract PC1 from each variable ---
```

```
resid_2022
← apply(X_2022, 2, function(y) lm(y ~ pc_2022)$residuals)
resid_2024
← apply(X_2024, 2, function(y) lm(y ~ pc_2024)$residuals)

# --- 5.
PCA для залишків ---
# --- 5.
PCA on the residuals ---
pca_2022_resid
← rda(resid_2022, scale = TRUE)
pca_2024_resid
← rda(resid_2024, scale = TRUE)

pc1_2022_resid
← scores(pca_2022_resid, display = "sites", choices = c(1:4)),
1:4]
pc1_2024_resid
← scores(pca_2024_resid, display = "sites", choices = c(1:4)),
1:4]

# --- 6. Порівняння eigenvalue до та після очищення ---
# --- 6.
Comparison of eigenvalues before and after cleaning ---
eigen_2022
← eigenvals(pca_2022)
eigen_2022_resid
← eigenvals(pca_2022_resid)

eigen_2024
← eigenvals(pca_2024)
eigen_2024_resid
← eigenvals(pca_2024_resid)

# --- 7. Порівняльна таблиця ---
# --- 7.
Comparative table ---
max_len
← max(length(eigen_2022), length(eigen_2022_resid))
eigen_compare_2022
← data.frame(
  PC = paste0("PC", 1:max_len),
```



```
Before = c(eigen_2022, rep(NA, max_len -
length(eigen_2022))),
After
= c(eigen_2022_resid, rep(NA, max_len - length(eigen_2022_resid)))
)
```

```
max_len
<- max(length(eigen_2024), length(eigen_2024_resid))
eigen_compare_2024
<- data.frame(
  PC = paste0("PC", 1:max_len),
  Before = c(eigen_2024, rep(NA, max_len -
length(eigen_2024))),
  After
= c(eigen_2024_resid, rep(NA, max_len - length(eigen_2024_resid)))
)
```

```
# --- Альтернативний спосіб: використати rownames ---
# ---
```

```
Alternative approach: using rownames ---
```

```
rownames(eigen_compare_2022)
<- eigen_compare_2022$PC
eigen_compare_2022
<- eigen_compare_2022[, -1]
print("===
2022: Порівняння
eigenvalue ===")
print(round(eigen_compare_2022,
2))
```

```
rownames(eigen_compare_2024)
<- eigen_compare_2024$PC
eigen_compare_2024
<- eigen_compare_2024[, -1]
print("===
2024: Порівняння
eigenvalue ===")
print(round(eigen_compare_2024,
2))
```

```
# === 5. Перестановковий тест
# === 5.
Permutation test for residual PCA data ===
```

```
proc_test
← protest(pc1_2022_resid, pc1_2024_resid, permutations = 999)
print(proc_test)

loadings_2022_cor
← pca_2022$CA$v[,1:2]
loadings_2024_cor
← pca_2024$CA$v[,1:2]

round(cbind(loadings_2022_cor,
loadings_2024_cor), 3)
# === 4. Прокрустовий аналіз на основі PCA ===
# === 4.
Procrustes analysis based on PCA of residuals ===
proc ←
procrustes(pc1_2022_resid, pc1_2024_resid, scale = F, symmetric = TRUE)
Yhat ←
proc$Yrot # Aligned coordinates for 2024
(subset for used cells)

summary(proc)

loadings_2022_cor
← pca_2022_resid$CA$v[,1:4]
loadings_2024_cor
← pca_2024_resid$CA$v[,1:4]
round(cbind(loadings_2022_cor,
loadings_2024_cor), 3)
# Прокрустов аналіз навантажень
#
Procrustes analysis of loadings
#
install.packages("plotrix") # uncomment if not already installed
library(plotrix)
# Виконуємо прокрустовий аналіз для loadings
# Perform
Procrustes analysis for the loadings
proc_load
← procrustes(loadings_2022_cor, loadings_2024_cor, scale = FALSE, symmetric
= TRUE)

# Побудова стандартного графіку прокрустового аналізу
```

```
# Plot a
standard graph for the Procrustes analysis
library(stats)
# Встановлюємо шрифт Times (якщо
"Times New Roman" встановлено у Windows)
windowsFonts(Times
= windowsFont("Times New Roman"))
par(family
= "Times", mar = c(4,4,0.5,0.5))
plot(proc_load,
main = " ", choices=c(1,2))
orig_x
<- proc_load$X[, 1]
orig_y
<- proc_load$X[, 2]
adjust_coords_iterative
<- function(x, y, orig_x, orig_y,
            min_dist
= 0.05, top_n = 5, max_iter = 10) {
  # Формуємо data.frame з мітками та
поточними координатами
  # Create a data.frame with labels and current
coordinates
  df <- data.frame(label = names(x), x = x,
y = y, stringsAsFactors = FALSE)

  # Обчислюємо зсув (shift) для кожної
точки як відстань між початковими та поточними координатами
  # Calculate the shift for each point as the
distance between the original and current coordinates
  df$shift <- sqrt((orig_x[names(x)] - x)^2
+ (orig_y[names(x)] - y)^2)

  # Відбираємо top_n точок з найбільшим зсувом (заспаданням shift)
  # Select the top_n points with the largest
shift (in descending order)
  df <- df[order(-df$shift), ]
  df <- head(df, top_n)

  # Якщо після відбору залишилося менше
2 точок, неможливо проводити розсовування
  # If fewer than 2 points remain after
selection, adjustment cannot be performed
```

```
if (nrow(df) < 2) {
  warning("Недостатньо точок для розсовування після відбору
top_n.")
  return(df)
}
# Ітеративне
розсовування вибраних точок, якщо вони надто близькі один до одного
# Iteratively adjust the selected points if
they are too close to each other
n <- nrow(df)
iter <- 1
moved <- TRUE

while (moved && iter <= max_iter)
{
  moved <- FALSE
  for (i in 1:(n - 1)) {
    for (j in (i + 1):n) {
      d <- sqrt((df$x[i] - df$x[j])^2 +
(df$y[i] - df$y[j])^2)
      if (d < min_dist) {
        angle <- atan2(df$y[j] - df$y[i],
df$x[j] - df$x[i])
        shift_val <- (min_dist - d) / 2 +
1e-4
        df$x[i] <- df$x[i] - cos(angle) *
shift_val
        df$y[i] <- df$y[i] - sin(angle) *
shift_val
        df$x[j] <- df$x[j] + cos(angle) *
shift_val
        df$y[j] <- df$y[j] + sin(angle) *
shift_val
        moved <- TRUE
      }
    }
  }
  iter <- iter + 1
}

return(df)
}
# Поточні координати (наприклад, для компонент 1 та 2)
```

```
# Current
coordinates (for components 1 and 2)
x ←
proc_load$Yrot[, 1]
y ←
proc_load$Yrot[, 2]
names(x)
← rownames(proc_load$Yrot)
names(y)
← rownames(proc_load$Yrot)
# Початкові координати
# Original
coordinates
names(orig_x)
← rownames(proc_load$X)
names(orig_y)
← rownames(proc_load$X)

# Викликаємо функцію: вибираємо top 5 точок з найбільшим зсувом,
# які при цьому (якщо занадто близько) будуть розсунуті
# Call the
function: select top 5 points with the largest shift and adjust if they are too
close
adjusted_top5
← adjust_coords_iterative(x, y, orig_x, orig_y,

min_dist = 0.02, top_n = 5, max_iter = 10)
# Виводимо результати
# Print the
results
print(adjusted_top5)

text(adjusted_top5$x,
adjusted_top5$y, labels = adjusted_top5$label, pos = 4, cex = 0.8, col =
"red")

round(cbind(loadings_2022_cor,
loadings_2024_cor), 3)
# === 8. Обчислюємо залишки (деформації) для підмножини
пікселів ===
# === 8.
Calculate residuals (deformations) for the subset of pixels ===
```

```
resid_PC1_subset
← Yhat[, 1] - pc1_2022_resid[, 1]
resid_PC2_subset
← Yhat[, 2] - pc1_2022_resid[, 2]
resid_PC3_subset
← Yhat[, 3] - pc1_2022_resid[, 3]
resid_PC4_subset
← Yhat[, 4] - pc1_2022_resid[, 4]
# === 9. Створюємо повні вектори залишків для всіх клітин
шаблону ===
# === 9.
Create full residual vectors for all template cells ===
n_cells
← ncell(template)
resid_PC1_full
← rep(NA, n_cells)
resid_PC2_full
← rep(NA, n_cells)
resid_PC3_full
← rep(NA, n_cells)
resid_PC4_full
← rep(NA, n_cells)

resid_PC1_full[sel]
← resid_PC1_subset
resid_PC2_full[sel]
← resid_PC2_subset
resid_PC3_full[sel]
← resid_PC3_subset
resid_PC4_full[sel]
← resid_PC4_subset
# === 10. Створення растрових шарів з залишками за шаблоном
===
# === 10.
Create raster layers from the residuals based on the template ===
r_PC1 ←
setValues(template, resid_PC1_full)
r_PC2 ←
setValues(template, resid_PC2_full)
r_PC3 ←
setValues(template, resid_PC3_full)
r_PC4 ←
setValues(template, resid_PC4_full)
```

```
# Переконаємося, що CRS збережено
# Ensure
the CRS is retained
crs(r_PC1)
← crs(template)
crs(r_PC2)
← crs(template)
crs(r_PC3)
← crs(template)
crs(r_PC4)
← crs(template)
# === 11. Візуалізація карти прокрустових деформацій ===
# === 11.
Visualization of the Procrustes deformation map ===
# Встановлюємо параметри графіки: 2 рядки і 2 стовпці, шрифт
Times
# Set up
graphics parameters: 2 rows and 2 columns, using Times font

# Функція для обчислення квантильних розбивань
# Function
to compute quantile breaks
getQuantileBreaks
← function(r, n = 7) {
  vals ← values(r, na.rm = TRUE)
  quantile(vals, probs = seq(0, 1, length.out =
n + 1), na.rm = TRUE)
}
# Функція для створення інтервальних міток у форматі
"a.aa - b.bb"
# Function
to create interval labels in the format "a.aa - b.bb"
createIntervalLabels
← function(breaks) {
  paste(sprintf("%.2f",
breaks[-length(breaks)]),
    "-",
    sprintf("%.2f", breaks[-1]))
}
# Встановлюємо шрифт Times (якщо
"Times New Roman" встановлено у Windows)
windowsFonts(Times
= windowsFont("Times New Roman"))
```

```
par(family
= "Times", mfrow = c(2, 2))

# Розрахунок квантильних розбивань та інтервальних міток для
кожного шару
# Compute
quantile breaks and interval labels for each raster layer (example: 7 classes)
qbreaks1
<- round(getQuantileBreaks(r_PC1, 7), 2)
qbreaks2
<- round(getQuantileBreaks(r_PC2, 7), 2)
qbreaks3
<- round(getQuantileBreaks(r_PC3, 7), 2)
qbreaks4
<- round(getQuantileBreaks(r_PC4, 7), 2)

labels1
<- createIntervalLabels(qbreaks1)
labels2
<- createIntervalLabels(qbreaks2)
labels3
<- createIntervalLabels(qbreaks3)
labels4
<- createIntervalLabels(qbreaks4)

#
----- СТВОРЮЄМО LAYOUT 2×4 -----
# Create a
2×4 layout: 2 rows and 4 columns for maps and legends
layout(matrix(1:8,
nrow = 2, ncol = 4, byrow = TRUE),
widths = c(3, 1, 3, 1),
heights = c(1,
1))
# ----- ПОЛЯ ДЛЯ ГРАФІКІВ ТА ЛЕГЕНД
-----
# Set
margins for maps and legends: for maps (mar = c(1,1,0.5,0.5)) and legends (mar
= c(1,0.5,0.5,0.5))
# --- Комірка 1 (r_PC1) ---
par(mar =
c(1,1,0.5,0.5))
```



```
plot(r_PC1,
main = "Residual PC1",
  breaks = qbreaks1,
  col
= hcl.colors(length(qbreaks1) - 1, "Spectral"),
  legend = FALSE)
# --- Комірка 2 (легенда для r_PC1) ---
par(mar =
c(1,0.5,0.5,0.5))
plot.new()
legend("center",
  legend = labels1,
  fill
= hcl.colors(length(qbreaks1) - 1, "Spectral"),
  title
= paste("Range:", sprintf("%.2f", min(qbreaks1)),
  "_",
sprintf("%.2f", max(qbreaks1))),
  bty
= "n", cex = 0.9)
# --- Комірка 3 (r_PC2) ---
par(mar =
c(1,1,0.5,0.5))
plot(r_PC2,
main = "Residual PC2",
  breaks = qbreaks2,
  col
= hcl.colors(length(qbreaks2) - 1, "Spectral"),
  legend = FALSE)

# --- Комірка 4 (легенда для r_PC2) ---
par(mar =
c(1,0.5,0.5,0.5))
plot.new()
legend("center",
  legend = labels2,
  fill
= hcl.colors(length(qbreaks2) - 1, "Spectral"),
  title
= paste("Range:", sprintf("%.2f", min(qbreaks2)),
  "_",
sprintf("%.2f", max(qbreaks2))),
```

```
      bty
= "n", cex = 0.9)

# --- Комірка 5 (r_PC3) ---
par(mar =
c(1,1,0.5,0.5))
plot(r_PC3,
main = "Residual PC3",
      breaks = qbreaks3,
      col
= hcl.colors(length(qbreaks3) - 1, "Spectral"),
      legend = FALSE)

# --- Комірка 6 (легенда для r_PC3) ---
par(mar =
c(1,0.5,0.5,0.5))
plot.new()
legend("center",
      legend = labels3,
      fill
= hcl.colors(length(qbreaks3) - 1, "Spectral"),
      title
= paste("Range:", sprintf("%.2f", min(qbreaks3)),
        "-",
        sprintf("%.2f", max(qbreaks3))),
      bty
= "n", cex = 0.9)

# --- Комірка 7 (r_PC4) ---
par(mar =
c(1,1,0.5,0.5))
plot(r_PC4,
main = "Residual PC4",
      breaks = qbreaks4,
      col
= hcl.colors(length(qbreaks4) - 1, "Spectral"),
      legend = FALSE)

# --- Комірка 8 (легенда для r_PC4) ---
par(mar =
c(1,0.5,0.5,0.5))
plot.new()
legend("center",
```

```
    legend = labels4,
    fill
= hcl.colors(length(qbreaks4) - 1, "Spectral"),
    title
= paste("Range:", sprintf("%.2f", min(qbreaks4)),
        "-",
    sprintf("%.2f", max(qbreaks4))),
    bty
= "n", cex = 0.9)
#
----- (Опціонально) Збереження результатів -----
#
(Optional) Save the resulting raster outputs
writeRaster(r_PC1,
"D:/GIS Data/SENTINEL/Output/proc_deform_PC1_res.tif", overwrite =
TRUE)
writeRaster(r_PC2,
"D:/GIS Data/SENTINEL/Output/proc_deform_PC2_res.tif", overwrite =
TRUE)
writeRaster(r_PC3,
"D:/GIS Data/SENTINEL/Output/proc_deform_PC3_res.tif", overwrite =
TRUE)
writeRaster(r_PC4,
"D:/GIS Data/SENTINEL/Output/proc_deform_PC4_res.tif", overwrite =
TRUE)

library(terra)

# Завантажуємо векторні дані з точками з shapefile
# Load
point vector data from a shapefile
points
← vect("D:/Science/Хортиця/Plants_Total.shp")

# Об'єднуємо растрові шари в один стек.
# Combine
the raster layers into a single stack
rasters_stack
← c(r_PC1, r_PC2, r_PC3, r_PC4)

# Екстрагуємо значення для кожного пункту.
# Extract
values at each point; the function returns a data.frame with the first column
```

as the point ID and subsequent columns for each raster.

```
extracted_values
```

```
← extract(rasters_stack, points)
```

```
# Переглянемо отримані дані
```

```
# View the
```

```
extracted values
```

```
print(head(extracted_values))
```

```
# Припускаємо, що extracted_values містить 5 стовпців, перший — "ID"  
та 4 стовпці з даними
```

```
# We assume
```

```
that extracted_values contains 5 columns: the first is "ID" and the  
next four are data values
```

```
names(extracted_values)
```

```
← c("ID", "PC1", "PC2", "PC3",  
    "PC4")
```

```
# Перевіряємо, як виглядають перші рядки
```

```
# Check the
```

```
first rows of the data
```

```
head(extracted_values)
```

```
# Зберігаємо data.frame у файл (CSV, наприклад)
```

```
# Save the
```

```
data.frame to a CSV file
```

```
write.csv(extracted_values,
```

```
"D:/Science/Хортиця/Plants_Total_extracted.csv",
```

```
row.names = FALSE)
```

Protocol references

- Addabbo, P., Focareta, M., Marcuccio, S., Votto, C., and Ullo, S. L. (2016). Contribution of Sentinel-2 data for applications in vegetation monitoring. *Acta IMEKO*, 5(2): 44–54. https://doi.org/10.21014/acta_imeko.v5i2.352
- Barnes, E. M., Clarke, T. R., Richards, S. E., Colaizzi, P. D., Haberland, J., Kostrzewski, M., Waller, P., Choi, C., Riley, E., and Thompson, T. (2000). Coincident detection of crop water stress, nitrogen status and canopy density using ground-based multispectral data. *Proceedings of the Fifth International Conference on Precision Agriculture*, 16–19.
- Chandrasekar, K., Sessa Sai, M. V. R., Roy, P. S., and Dwevedi, R. S. (2010). Land surface water index (LSWI) response to rainfall and NDVI using the MODIS vegetation index product. *International Journal of Remote Sensing*, 31(15): 3987–4005. <https://doi.org/10.1080/01431160802575653>
- Chaves, M. E. D., Picoli, M. C. A., and Sanches, I. D. (2020). Recent applications of Landsat 8/OLI and Sentinel-2/MSI for land use and land cover mapping: A systematic review. *Remote Sensing*, 12(18): 3062. <https://doi.org/10.3390/rs12183062>
- Eastman, J. R., and Fulk, M. (1993). Long sequence time series evaluation using standardized principal components. *Photogrammetric Engineering and Remote Sensing*, 59 1307–1312. <https://api.semanticscholar.org/CorpusID:128867383>
- Elfanah, A. M. S., Darwish, M. A., Selim, A. I., Elmoselhy, O. M. A., Ali, A. M., El-Maghraby, M. A., and Abdelhamid, M. T. (2023). Hyperspectral reflectance and agro-physiological traits for field identification of salt-tolerant wheat genotypes using the genotype by yield*trait biplot technique. *Frontiers in Plant Science*, 14. <https://doi.org/10.3389/fpls.2023.1165113>
- Gašparović, M., and Singh, S. K. (2020). Spatio-temporal salinity monitoring of the Ghaghara river using Landsat time-series imagery and multiple regression analysis. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLIII-B3-2 401–405. <https://doi.org/10.5194/isprs-archives-XLIII-B3-2020-401-2020>
- Gitelson, A. A., Kaufman, Y. J., and Merzlyak, M. N. (1996). Use of a green channel in remote sensing of global vegetation from EOS-MODIS. *Remote Sensing of Environment*, 58(3): 289–298. [https://doi.org/10.1016/S0034-4257\(96\)00072-7](https://doi.org/10.1016/S0034-4257(96)00072-7)
- Ha, N. T. T., Thao, N. T. P., Koike, K., and Nhuan, M. T. (2017). Selecting the best band ratio to estimate chlorophyll-a concentration in a tropical freshwater lake using Sentinel 2A images from a case study of

- Lake Ba Be (Northern Vietnam). *ISPRS International Journal of Geo-Information*, 6(9): 290. <https://doi.org/10.3390/ijgi6090290>
- Hardisky, M. A., Klemas, V., and Smart, R. M. (1983). The influence of soil salinity, growth form, and leaf moisture on the spectral radiance of *Spartina alterniflora* canopies. *Photogrammetric Engineering and Remote Sensing*, 49(1): 77–83.
- Imbroane, M. A., Melenti, C., and Gorgan, D. (2007). Mineral explorations by Landsat image ratios. *Ninth International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC 2007)*, 335–340. <https://doi.org/10.1109/SYNASC.2007.52>
- Komlyk, Y., Ponomarenko, O., and Zhukov, O. (2024). A hemeroby gradient reveals the structure of bird communities in urban parks. *Biosystems Diversity*, 32(4): 426–436. <https://doi.org/10.15421/012446>
- Louhaichi, M., Borman, M. M., and Johnson, D. E. (2001). Spatially located platform and aerial photography for documentation of grazing impacts on wheat. *Geocarto International*, 16(1): 65–70. <https://doi.org/10.1080/10106040108542184>
- Lugassi, R., Goldshleger, N., and Chudnovsky, A. (2017). Studying vegetation salinity: From the field view to a satellite-based perspective. *Remote Sensing*, 9(2): 122. <https://doi.org/10.3390/rs9020122>
- McFeeters, S. K. (1996). The use of the Normalized Difference Water Index (NDWI) in the delineation of open water features. *International Journal of Remote Sensing*, 17(7): 1425–1432. <https://doi.org/10.1080/01431169608948714>
- Oksanen, J., Blanchet, F. G., Friendly, M., Kindt, R., Legendre, P., McGlinn, D., Minchin, P. R., O'Hara, R. B., Simpson, G. L., Solymos, P., Stevens, M. H. H., Szoecs, E., and Wagner, H. (2022). *vegan: Community Ecology Package. R package version 2.6-4*. <https://cran.r-project.org/package=vegan>
- Penuelas, J., Baret, F., and Filella, I. (1995). Semi-empirical indices to assess carotenoids/chlorophyll a ratio from leaf spectral reflectance. *Photosynthetica*, 31 221–230.
- Premkumar, R., Venkatachalapathy, R., and Visweswaran, S. (2021). Mapping of Total Suspended Matter based on Sentinel-2 data on the Hooghly River, India. *Indian Journal of Ecology*, 48(1): 159–165.
- Rouse, J. W., Haas, R. H., Schell, J. A., and Deering, D. W. (1974). Monitoring vegetation systems in the Great Plains with ERTS. *NASA Goddard Space Flight Center 3d ERTS-1 Symposium, Vol. 1, Sect. A.*, 309–317.
- Seydi, S. T., Akhoondzadeh, M., Amani, M., and Mahdavi, S. (2021). Wildfire damage assessment over Australia using Sentinel-2 imagery and MODIS Land Cover Product within the Google Earth Engine Cloud Platform. *Remote Sensing*, 13(2): 220. <https://doi.org/10.3390/rs13020220>

- Van Deventer, A. P., Ward, A. D., Gowda, P. M., and Lyon, J. G. (1997). Using thematic mapper data to identify contrasting soil plains and tillage practices. *Photogrammetric Engineering and Remote Sensing*, 63(1): 87–93.
- Wang, J., Ding, J., Yu, D., Ma, X., Zhang, Z., Ge, X., Teng, D., Li, X., Liang, J., Lizaga, I., Chen, X., Yuan, L., and Guo, Y. (2019). Capability of Sentinel-2 MSI data for monitoring and mapping of soil salinity in dry and wet seasons in the Ebinur Lake region, Xinjiang, China. *Geoderma*, 353 172–187. <https://doi.org/10.1016/j.geoderma.2019.06.040>
- Xie, Q., Dash, J., Huang, W., Peng, D., Qin, Q., Mortimer, H., Casa, R., Pignatti, S., Laneve, G., Pascucci, S., Dong, Y., and Ye, H. (2018). Vegetation indices combining the red and red-edge spectral information for leaf area index retrieval. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 11(5): 1482–1493. <https://doi.org/10.1109/JSTARS.2018.2813281>
- Xu, H. (2006). Modification of normalised difference water index (NDWI) to enhance open water features in remotely sensed imagery. *International Journal of Remote Sensing*, 27(14): 3025–3033. <https://doi.org/10.1080/01431160600589179>
- Yang, C., Everitt, J. H., Bradford, J. M., and Murden, D. (2004). Airborne hyperspectral imagery and yield monitor data for mapping cotton yield variability. *Precision Agriculture*, 5(5): 445–461. <https://doi.org/10.1007/s11119-004-5319-8>
- Yazdi, M., Taheri, M., Navi, P., and Sadati, N. (2013). Landsat ETM+ imaging for mineral potential mapping: application to Avaj area, Qazvin, Iran. *International Journal of Remote Sensing*, 34(16): 5778–5795. <https://doi.org/10.1080/01431161.2013.797127>
- Zhao, H., and Chen, X. (2005). Use of normalized difference bareness index in quickly mapping bare areas from TM/ETM+. *Proceedings. 2005 IEEE International Geoscience and Remote Sensing Symposium, 2005. IGARSS '05.*, 3 1666–1668. <https://doi.org/10.1109/IGARSS.2005.1526319>