Bollettino dell'Unione Matematica Italiana
https://doi.org/10.1007/s40574-021-00300-w

High order singular value decomposition for plant diversity estimation

Alessandra Bernardi[1] · **Martina Iannacito**[2] · **Duccio Rocchini**[3,4]

Received: 9 February 2021 / Accepted: 1 June 2021

Abstract
We propose a new method to estimate plant diversity with Rényi and Rao indexes through the so called High Order Singular Value Decomposition (HOSVD) of tensors. Starting from NASA multi-spectral images we evaluate diversity and we compare original diversity estimates with those realized via the HOSVD compression methods for big data. Our strategy turns out to be extremely powerful in terms of memory storage and precision of the outcome. The obtained results are so promising that we can support the efficiency of our method in the ecological framework.

1 Introduction

In order to face Earth changes in space and time, satellite imagery is being used to provide timely information over the whole globe. From this point of view, starting from the early 1970s, technological improvements in remote sensors led to a growth in the number and in the size of available data. For example to store a band of the entire Earth's surface from the MODIS sensor with a low spectral resolution, 5600 m, we need around 99 MB. Therefore for rasters with higher spectral, spatial or radiometric resolutions the memory request increases significantly. One may point out that 99 MB are not much, if compared with the features of modern machines. However the memory needed to process thousands of images might represent a crucial issue. Moreover, in order to perform any index computation over these

✉ Alessandra Bernardi
alessandra.bernardi@unitn.it

Martina Iannacito
martina.iannacito@inria.fr

Duccio Rocchini
duccio.rocchini@unibon.it

[1] Department of Mathematics, University of Trento, 38123 Povo, TN, Italy

[2] Inria Bordeaux-Sud-Ouest, 200, avenue de la Vieille Tour, 33405 Talence, France

[3] BIOME Lab, Department of Biological, Geological and Environmental Sciences, Alma Mater Studiorum-University of Bologna, via Irnerio 42, 40126 Bologna, Italy

[4] Department of Spatial Sciences, Faculty of Environmental Sciences, Czech University of Life Sciences Prague, Kamýcká 129, 165 00 Praha-Suchdol, Czech Republic

Published online: 10 June 2021

images, they should be loaded into the computer RAM, which usually has lower capacity and is occupied also by the system application and by our computing script.

Since the advent of big data, which include multi-spectral and hyperspectral images, improving storage and analytical techniques became fundamental. Mathematicians are facing this challenge together with computer scientists, physicists and engineers (cf. e.g. [1–7,13,19,26,38,39,42]). The usual structure for storing multi-spectral images are *tensors*. For matrices approximation there exists an optimal technique, called *Singular Values Decomposition*, SVD (cf. [18,27,37,40]). One application of SVD is image compression (cf. e.g. [20,23,44]). Since we need to store data in tensors, i.e. the generalisation of matrices, we will present *High Order Singular Value Decomposition* (HOSVD) firstly introduced in [13,42], a generalisation of SVD to tensors. There are various techniques to approximate tensors by taking advantage of the HOSVD, those that will be implemented in this paper are the so called Truncated HOSVD (T-HOSVD) and Sequentially Truncated HOSVD (ST-HOSVD). Even if only for some special cases HOSVD provides optimal results [9,17,41], it is possible to present an estimate of the tensor approximation errors. Indeed the core of the present paper will be: (i) the application of T-HOSVD and ST-HOSVD techniques, (ii) their modern versions and (iii) the associated errors. Indeed, in the last section we will apply some possible HOSVD implementations to tensors in which we stored RED and NIR bands. Next from the compressed tensors we get NDVI rasters and over them we compute diversity indexes. So we will be able to compare our diversity estimations from compressed data with those realized over original data.

As far as we know, this is the first attempt to estimate diversity with the presented indexes from HOSVD compressed data. Moreover the obtained results are extremely promising, letting us support its efficiency in ecological framework.

2 Ecological background

One of the main components of Earth's biosphere is diversity, which is in strict relationship with the planet status in space and time. Measuring diversity over wide spatial scales is difficult concerning time, costs and logistical issues, e.g.: (i) the number of sampling units to be investigated, (ii) the choice of the sampling design, (iii) the need to clearly define the statistical population, (iv) the need for an operational definition of a species community, etc. (cf. [11]). Hence, ecological proxies of species diversity are important for developing effective management strategies and conservation plans worldwide (cf. [32]). From this point of view, environmental heterogeneity is considered to be one of the main factors associated to a high degree of biological diversity given that areas with higher environmental heterogeneity can host more species due to the greater number of available niches within them. Therefore, measuring heterogeneity from satellite images might represent a powerful approach to gather information of diversity changes in space and time in an effective manner (cf. [45]). Nowadays, the advent of satellites has made possible having real images of a territory, even with a remarkable quality. This approach is behind the remote sensing discipline. Many definitions have been proposed during the years for this subject. We refer to the one presented in [10, p. 6].

Definition 2.1 Remote sensing is the practice of deriving information about Earth's land and water surfaces using images acquired from an overhead perspective, using electromagnetic radiation in one or more regions of the electromagnetic spectrum, reflected or emitted from the Earth's surface.

This field of study is based on a well known physical phenomenon: different materials and different organisms absorb and reflect electromagnetic radiations differently. Most of the modern satellite sensors can acquire multiple images, divided in the so called bands.

Definition 2.2 *Bands* or *channels* are the recorded spectral measurements.

Depending on the sensor features, we can acquire images dived into the different bands.

Definition 2.3 *Multi-spectral sensor* can acquire from 4 up to 10 bands. *Hyper-spectral sensor* can acquire more than 100 bands.

Data used in the present paper come from the *MODerate-Resolution Imaging Spectroradiometer* sensor, or MODIS, built on both the satellites Terra and Aqua, cf. [12]. MODIS measures 36 bands in the visible and infrared spectrum, at different resolutions. Firstly there are RED and NIR bands with pixel size of 250 km and next 5 bands, still with RED and NIR, at 500m of spatial resolution. These are extremely useful for land observation. The remaining bands with 1 km of resolution consist of monitoring images from visible spectrum, MIR, NIR and TIR. The data are registered four times a day: twice daytime and twice night-time. Images usually include the entire Earth's surface.

From the 1960s the scientific community highlighted two important relations between spectral measurements and biomass:

- a direct relation between NIR region and biomass, i.e. greater the biomass greater the NIR reflected radiation measured and vice versa;
- an inverse relation between RED spectral region and biomass, i.e. greater the biomass, lower the RED visible reflected spectrum measured and vice versa.

Therefore the relation between NIR and RED reflectance is central for the vegetal biomass estimation. The aim of vegetation indexes is measuring biomass or vegetative vigour on the base of digital brightness values. The one used in the present article is the *Normalized Difference Vegetation Index*, or NDVI, presented in 1974 by Rouse and others, cf. [36].

Definition 2.4 Given a region R, let $RED, NIR \in \mathbb{M}^{m\times n}(\mathbb{R})$ be respectively the RED and the NIR raster band of R imagery. The *normalized difference vegetation index* of region R is $NDVI \in \mathbb{M}^{m\times n}(\mathbb{R})$ such that

$$\mathrm{NDVI}_{ij} = \frac{NIR_{ij} - RED_{ij}}{NIR_{ij} + RED_{ij}}$$

for every $i \in \{1, \ldots, m\}$ and for every $j \in \{1, \ldots, n\}$, when it is defined.

The innovative idea of [33] was the application of information theory studies and ecological indexes to remote sensed images. The ecological indexes used in the present paper are the Rao and Rényi ones, which we are going to define.

Definition 2.5 Given a spectral image of a sample area, let N be the image radiometric resolution and let p_i be the relative abundances of the i-th value for every $i \in \{1, \ldots, N\}$. Fixed a distance function d, we build up a pairwise spectral difference matrix $D \in \mathbb{M}^N(\mathbb{R})$ such that

$$D_{ij} = d(i, j)$$

for every $i, j \in \{1, \ldots, N\}$. The *Rao's Q index* for the sample area is

$$I_{RQ} = \sum_{j=1}^{N}\sum_{i=1}^{N} p_i\, p_j\, D_{ij}.$$

Definition 2.6 Given a sample area with N species and defined p_i the relative abundances for every $i \in \{1, \ldots, N\}$, in decreasing order, the *Rényi index* is

$$I_R = -\log \sum_{i=1}^{N} p_i^2.$$

The main difference between them is that Rao index takes into account both the frequencies and the numerical values of each pixel. On the other side Rényi index considers only pixels frequencies. Lastly notice that rasters usually are split into small chunks, called *windows*, over which the chosen diversity index is computed. We need to generate Rényi's and Rao's computation codes to measure how much diversity information is lost using approximated tensor data. Since we keep the ecologists moving window approach, our implementation presents also a very basic multi-core modality. The multi-cores Rény's and Rao's implementations crux is presented in the Appendix, Sect. 6.1.

3 Mathematical background

The celebrated Schmidt [37,40], Mirsky [27], Eckhart, Young [18] theorem states that for every rank-r matrix $A \in \mathbb{M}^{m\times n}(\mathbb{C})$ there exists the best rank s approximation for every $s < r$ in the Frobenius norm, actually the result holds for arbitrary $(O_m \times O_n)$-invariant norms [28]. The famous generalization of this result [13] to higher order tensors fails in general to produce an output that is the "best" approximation of a given tensor, cf. [42]. It succeeds for the so called orthogonally decomposable tensors [9]. Another type of generalization of the concept of "best rank-r approximation of a tensor" proposed in [17] works for general tensors. Despite the possible non-existence of the best approximation of a tensor under the construction of [13], this technique turns out to be very convenient from the computational point of view and it is possible to prove that the outcome is a "quasi-optimal solution". In this paper we will mainly use the technique popularized by De Lathauwer and Vandewalle in [13], the so called *High Order Singular Value Decomposition* (HOSVD), we will study two types of errors made by the specific approximation, we will make a comparison among them and we will show that they will be small enough to have very precise results.

Here some basic mathematical preliminaries are provided. The number field will always be the complex field of numbers $\mathbb{C}$.

Definition 3.1 Let V be a linear subspace of the tensor space $\mathbb{C}^{n_1} \otimes \cdots \otimes \mathbb{C}^{n_d}$. If for every $i \in \{1, \ldots, d\}$, there exist a subspace $V_i \subseteq \mathbb{C}^{n_i}$ such that

$$V = V_1 \otimes \cdots \otimes V_d,$$

then V is *a separable tensor subspace* of $\mathbb{C}^{n_1} \otimes \cdots \otimes \mathbb{C}^{n_d}$.

Not every subspace of $\mathbb{C}^{n_1} \otimes \cdots \otimes \mathbb{C}^{n_d}$ is separable. The structure of a separable tensor space has some consequences on its elements.

Definition 3.2 The *multilinear rank* of a tensor $A \in \mathbb{C}^{n_1} \otimes \cdots \otimes \mathbb{C}^{n_d}$ is the d-uple $(r_1, \ldots, r_d)$ with the property that r_i is the minimal dimension of a subspace $V_i \subset \mathbb{C}^{n_i}$ such that $A \in V_1 \otimes \cdots \otimes V_d$ for every $i \in \{1, \ldots, d\}$.

Let $A \in \mathbb{C}^{n_1} \otimes \cdots \otimes \mathbb{C}^{n_d}$ be a tensor and let $(r_1, \ldots, r_d) \in \mathbb{N}^d$. We will discuss if there exists a tensor M of multilinear rank lower or equal component-wise than $(r_1, \ldots, r_d)$ which

minimizes the Frobenius norm of the tensor difference, i.e.

$$M = \arg \inf_{\text{mlrank}(T) \leq (r_1, \ldots, r_d)} \left\| A - T \right\|.$$

This problem is also known as *Low MultiLinear Rank Approximation* (LMLRA). Ishteva and De Lathauwer firstly stated this problem and introduced this acronym, cf. [22]. Looking for a tensor $M \in \mathbb{C}^{n_1} \otimes \cdots \otimes \mathbb{C}^{n_d}$ satisfying the stated rank properties means searching a subspace $V_i \subset \mathbb{C}^{n_i}$ of dimension r_i for every $i \in \{1, \ldots, d\}$ such that, if the approximation tensor M exists, it belongs to $V_1 \otimes \cdots \otimes V_d$.

The techniques of T-HOSVD and ST-HOSVD are described by De Lathauwer, De Moor and Vandewalle, in [13] and by Vannieuwenhoven, Vandebril and Meerbergen, in [42] respectively.

Definition 3.3 The *multilinear multiplication with respect to mode h* of an order-d tensor $A \in \mathbb{C}^{n_1 \times \cdots \times n_d}$ by a matrix $M_h \in \mathbb{C}^{m_h \times n_h}$ is

$$M_h \cdot_h A := \sum_{i=1}^{r} a_1^i \otimes \cdots \otimes (M_h a_h^i) \otimes \cdots \otimes a_d^i,$$

whenever $A = \sum_{i=1}^{r} a_1^i \otimes \cdots \otimes a_d^i$. The *multilinear multiplication* of an order-d tensor $A \in \mathbb{C}^{n_1 \times \cdots \times n_d}$ by a tuple of matrices $(M_1, \ldots, M_d)$, $M_i \in \mathbb{C}^{m_i \times n_i}$ is

$$(M_1, \ldots, M_d) \cdot A := M_1 \cdot_1 \cdots M_d \cdot_d A.$$

The resulting tensor belongs to $\mathbb{C}^{m_1 \times \cdots \times m_d}$.

Definition 3.4 [8,29,31] Given a matrix $M \in \mathbb{M}^{m \times n}(\mathbb{C})$, the *Moore–Penrose inverse* is a matrix $M^\dagger \in \mathbb{M}^{n \times m}$ of M satisfying the following properties:

- $MM^\dagger M = M$;
- $M^\dagger M M^\dagger = M^\dagger$;
- $(MM^\dagger)^H = MM^\dagger$;
- $(M^\dagger M)^H = M^\dagger M$.

If $B = (M_1, \ldots, M_d) \cdot A$ in Definition 3.3 with M_i's square invertible matrices, then

$$(M_1^\dagger, \ldots, M_d^\dagger) \cdot B = A.$$

A tensor also naturally defines the following multilinear maps.

Definition 3.5 Let $A \in \mathbb{C}^{n_1} \otimes \cdots \otimes \mathbb{C}^{n_d}$. For any $k \in \{1, \ldots, d\}$ we can define the k-th *standard flattening* of A as

$$\begin{aligned} A_{(k)} &: (\mathbb{C}^{n_1})^* \times \cdots \times (\mathbb{C}^{n_{k-1}})^* \times (\mathbb{C}^{n_{k+1}})^* \times \cdots \times (\mathbb{C}^{n_d})^* \to \mathbb{C}^{n_k} \\ &(w_1, \ldots, w_{k-1}, w_{k+1}, \ldots, w_d) \mapsto (w_1, \ldots, w_{k-1}, I, w_{k+1}, \ldots, w_d)^T \cdot A. \end{aligned}$$

More generally, let $p \sqcup q = [1, d]$ be a partition of d with $s = |p|$ and $t = |q|$. Then, we can associate to A even more multilinear maps, namely

$$\begin{aligned} A_{(p;q)} &: (\mathbb{C}^{n_{q_1}})^* \times \cdots \times (\mathbb{C}^{n_{q_t}})^* \to \mathbb{C}^{n_{p_1}} \otimes \cdots \otimes \mathbb{C}^{n_{p_s}} \\ &(w_1, \ldots, w_t) \mapsto w_1^T \cdot_{q_1} \cdots w_{t-1}^T \cdot_{q_{t-1}} w_t^T \cdot_{q_t} A \end{aligned}$$

whose $\prod_{j=1}^{s} n_{p_j} \times \prod_{j=1}^{t} n_{q_j}$ matrix is

$$A_{(p;q)} = \sum_{i=1}^{r} (a^i_{p_1} \otimes \cdots \otimes a^i_{p_s})(a^i_{q_1} \otimes \cdots \otimes a^i_{q_t})^T.$$

This interpretation of A is called a *flattening* of A.

The punchline of the HOSVD is to apply the SVD to the flattenings of a given tensor and to use the More-Penrose transform, Definition 3.4, to build an approximated tensor. As already outlined, this technique is optimal for the so called orthogonally decomposable (ODeCo) tensors, cf. [9], while for other cases there are no evidences that this procedure would lead to "the best multilinear-rank approximation" of a given tensor. Unlike matrices, tensors of higher order can fail to have best rank-r approximations, cf. [14]. Anyway, HOSVD algorithms are extremely explicit and it is possible to estimate the measure of the error made by using this approximation.

HOSVD provides a sparse representation of the given tensor, whose costs in terms of storage use can be easily computed. Indeed given $A \in \mathbb{C}^{n_1} \otimes \cdots \otimes \mathbb{C}^{n_d}$ whose multilinear rank is known to be $(r_1, \ldots, r_d)$, from HOSVD we get that $A = (U_1, \ldots, U_d) \cdot C$ with U_i rank-r_i orthogonal $(n_i \times r_i)$-matrices and $C \in \mathbb{C}^{r_1} \otimes \cdots \otimes \mathbb{C}^{r_d}$ the so called *core tensor*. The storage of each matrix U_i costs $n_i r_i$ memory units for every $i \in \{1, \ldots, d\}$ and storing the core tensor C needs $\prod_{i=1}^{d} r_i$ memory units. In conclusion the sparse representation of A costs

$$\sum_{i=1}^{d} n_i r_i + \prod_{i=1}^{d} r_i \text{ memory units}$$

which is extremely better than $\prod_{i=1}^{d} n_i$.

The following proposition reveals the main strategy of the HOSVD.

Proposition 3.6 (HOSVD [42]) *Let $V = V_1 \otimes \cdots \otimes V_d$ be a separable tensor subspace of $\mathbb{C}^{n_1} \otimes \cdots \otimes \mathbb{C}^{n_d}$ with* $\dim V_i = r_i$, $i = 1, \ldots, d$. *Let*

$$\mathcal{B} = \{u^1_{i_1} \otimes \cdots \otimes u^d_{i_d}\}_{\substack{i_j=1,\ldots,r_j \\ j=1,\ldots,d}}$$

be an orthogonal basis of V for the standard product, and let $(U_i) = (u^i_{j_i})_{i=1,\ldots,d;\, j_i=1,\ldots,r_i}$ be the corresponding orthogonal bases for the V_i's, i=1, …, d. The projector

$$\pi_i : \mathbb{C}^{n_1} \otimes \cdots \otimes \mathbb{C}^{n_d} \mapsto \mathbb{C}^{n_1} \otimes \cdots \otimes \mathbb{C}^{n_d}$$

is such that for every $A \in \mathbb{C}^{n_1} \otimes \cdots \otimes \mathbb{C}^{n_d}$

$$\pi_i(A) = (U_i^H U_i) \cdot_i A$$

for every $i \in \{1, \ldots, d\}$. Define $P_V(A) := \pi_1 \ldots \pi_d(A)$. Then for every $A \in \mathbb{C}^{n_1} \otimes \cdots \otimes \mathbb{C}^{n_d}$ and for every ρ permutation of d elements, we get that:

$$\|A - P_V(A)\|^2 = \sum_{i=1}^{d} \left\| \pi_{\rho_{i-1}} \ldots \pi_{\rho_1}(A) - \pi_{\rho_i} \ldots \pi_{\rho_1}(A) \right\|^2. \tag{1}$$

Corollary 3.7 *Under the hypothesis of Proposition 3.6, we have that for every tensor* $A \in \mathbb{C}^{n_1} \otimes \cdots \otimes \mathbb{C}^{n_d}$

$$\|A - P_V(A)\|^2 \leq \sum_{i=1}^{d} \left\| \pi_i^{\perp}(A) \right\|^2 .$$

Usually, the multilinear rank of the given tensor is not known. Consequently, working on the exact error is not convenient. Therefore, on the basis of Proposition 3.6 and Corollary 3.7, Vannieuwenhoven Meerbergen and Vandebril developed two new strategies to approximate tensors, cf. [42], when the multilinear rank is not known. The first is based on the approximation of the upper bound of the error, stated in Corollary 3.7. As a matter of fact reducing the upper bound implies reducing the exact error.

The *Truncated Higher Order Singular Value Decomposition* (T-HOSVD) has the SVD as key concept. Fixed $A \in \mathbb{C}^{n_1} \otimes \cdots \otimes \mathbb{C}^{n_d}$, consider the i-th term of the upper bound summation, i.e.

$$0 \leq \left\| \pi_i^{\perp}(A) \right\|^2 = \left\| A - (U_i U_i^H)_{\cdot_i} A \right\|^2 .$$

By the positivity of each term of the previous expression, minimizing the upper bound means minimizing each term of the upper bound summation, i.e. taking the minimum of the norm of the difference between A and its projection in just one direction each time. Because the tensor is approximated only with respect to one direction each time, the minimization problem is mathematically

$$\arg \min_{\pi_i \text{ projection into } V_i} \left\| \pi_i^{\perp}(A) \right\|^2 = \arg \min_{U_i \in \mathbb{O}(n_i \times r_i)} \left\| A_{(i)} - (U_i U_i^H) A_{(i)} \right\|^2 ,$$

i.e. looking for the best approximation at rank r_i of the i-th flattening of A for every $i \in \{1, \ldots, d\}$. However thanks to Schmidt, Mirsky, Eckhart, Young theorem the problem for matrices has a close solution which is obtained through SVD of the i-th flattening of A, truncated at the r_i singular values for every $i \in \{1, \ldots, d\}$. It is clear that the order of projectors application is not significant for the T-HOSVD. This won't be true anymore for the *Sequentially Truncated High Order Singular Value* (ST-HOSVD). ST-HOSVD minimizes each term of the summation of Proposition 3.6. Let $A \in \mathbb{C}^{n_1} \otimes \cdots \otimes \mathbb{C}^{n_d}$ be a tensor and let $(r_1, \ldots, r_d)$ be the target multilinear rank of the approximation. The first step is looking for the projector which minimises the first error term of Equation (1):

$$\arg \min_{\pi_1} \left\| \pi_1^{\perp}(A) \right\|^2 = \arg \min_{U_1 \in \mathbb{O}(n_1 \times r_1)} \left\| (U_1 U_1^H)_{\cdot_1} A \right\|^2 ,$$

i.e.

$$\arg \min_{U_1 \in \mathbb{O}(n_1 \times r_1)} \left\| (U_1 U_1^H) A_{(1)} \right\|^2 .$$

The last formulation of this first step has a close solution, thanks to Schmidt, Mirsky, Eckhart, Young theorem. We are looking for the best rank r_1 approximation of the 1-st flattening. So we can conclude that the matrix U_1 obtained from the SVD of the first flattening of A truncated at the r_1-th column is such that

$$\hat{U}_1 = \arg \min_{U_1 \in \mathbb{O}(n_1 \times r_1)} \left\| (U_1 U_1^H) A_{(1)} \right\|^2 .$$

The core tensor of the first step,

$$C^{(1)} = (\hat{U}_1^H, I, \ldots, I) \cdot A.$$

Fixed the $\hat{\pi}_1 = \hat{U}_1 \hat{U}_1^H$, the next step is looking for the projector which minimises the second error term of Equation (1):

$$\arg\min_{\pi_2} \left\| \pi_2^{\perp} \hat{\pi}_1(A) \right\|^2 = \arg \min_{U_2 \in \mathbb{O}(n_2 \times r_2)} \left\| (U_2 U_2^H)_{\cdot 2} (\hat{U}_1 \hat{U}_1^H)_{\cdot 1} A \right\|^2.$$

Thanks to multilinearity, the last equation becomes

$$\arg \min_{U_2 \in \mathbb{O}(n_2 \times r_2)} \left\| (\hat{U}_1)_{\cdot 1} (U_2 U_2^H)_{\cdot 2} (\hat{U}_1^H)_{\cdot 1} A \right\|^2$$

i.e.

$$\arg \min_{U_2 \in \mathbb{O}(n_2 \times r_2)} \left\| (\hat{U}_1)_{\cdot 1} (U_2 U_2^H)_{\cdot 2} C^{(1)} \right\|^2.$$

Since $\hat{U}_1$ is fixed, the second step problem becomes

$$\arg \min_{U_2 \in \mathbb{O}(n_2 \times r_2)} \left\| (U_2 U_2^H)_{\cdot 2} C^{(1)} \right\|^2 = \arg \min_{U_2 \in \mathbb{O}(n_2 \times r_2)} \left\| (U_2 U_2^H)_{\cdot 2} C^{(1)}_{(2)} \right\|^2$$

whose solution is the matrix $\hat{U}_2$ obtained through the SVD of the 2-nd flattening of $C^{(1)}$ truncated at the r_2-th column. Once the new core tensor is defined as

$$C^{(2)} = (\hat{U}_1^H, \hat{U}_2^H, I, \ldots, I) \cdot A,$$

we proceed similarly with the next ones. At the $(d-1)$-th step the $(d-1)$-th core tensor is defined as

$$C^{(d-1)} = (\hat{U}_1^H, \hat{U}_2^H, \ldots, \hat{U}_{d-1}^H, I) \cdot A.$$

For the last d-th step, we look for the projectors which minimise the last error term of Equation (1):

$$\arg\min_{\pi_d} \left\| \pi_d^{\perp} \hat{\pi}_{d-1} \ldots \hat{\pi}_1(A) \right\|^2,$$

which, using the multilinearity and the projectors properties, becomes

$$\arg \min_{U_d \in \mathbb{O}(n_d \times r_d)} \left\| (U_d U_d^H)_{\cdot d} (\hat{U}_1 \hat{U}_1^H, .., \hat{U}_{d-1} \hat{U}_{d-1}^H)_{\cdot 1, \ldots, d-1} A \right\|^2$$

i.e.

$$\arg \min_{U_d \in \mathbb{O}(n_d \times r_d)} \left\| (U_d U_d^H) C^{(d-1)}_{(k)} \right\|^2.$$

The solution of the last problem is the matrix $\hat{U}_d$ obtained through the SVD of the d-th flattening of the core tensor $C^{(d-1)}$ truncated at the r_d-th column.

We can now state both the T-HOSVD and the ST-HOSVD algorithms, whose implementations are in the Appendix, Sects. 6.2 and 6.3 respectively.

Algorithm 1: T-HOSVD

Input: a tensor $A \in \mathbb{C}^{n_1} \otimes \cdots \otimes \mathbb{C}^{n_d}$
Input: a target multilinear rank $(r_1, \ldots, r_d)$
Output: the T-HOSVD basis in matrix form $(\overline{U}_1, \ldots, \overline{U}_d)$
Output: the T-HOSVD core tensor $C \in \mathbb{C}^{n_1} \otimes \cdots \otimes \mathbb{C}^{n_d}$
for $i = 1, 2, \ldots, d$ **do**
  Compute SVD of $A_{(i)}$, i.e. $A_{(i)} = U_i \Sigma_i V_i^T$;
  Store in $\overline{U}_i$ the first r_i columns of U_i
end
$C \leftarrow (\overline{U}_1^H, \ldots, \overline{U}_d^H) \cdot A$;

Algorithm 2: ST-HOSVD

Input: a tensor $A \in \mathbb{C}^{n_1} \otimes \cdots \otimes \mathbb{C}^{n_d}$
Input: a target multilinear rank $(r_1, \ldots, r_d)$
Output: the ST-HOSVD basis in matrix form $\hat{U}_1, \ldots, \hat{U}_d$
Output: the ST-HOSVD core tensor $C \in \mathbb{C}^{n_1} \otimes \cdots \otimes \mathbb{C}^{n_d}$
$C \leftarrow A$;
for $i = 1, 2, \ldots, d$ **do**
  Compute SVD of $C_{(i)}$, i.e. $C_{(i)} = U_i \Sigma_i V_i^T$;
  Store in $\hat{U}_i$ the first r_i columns of U_i;
  $C \leftarrow (\hat{U}_i^H)_{\cdot_i} C$;
end

4 Application

In this section we will first describe the data used: RED (red band), NIR (near infrared band) and the Normalized Difference Vegetation Index (NDVI, which is a combination of NIR and RED as (NIR-RED)/(NIR+RED)) of different territories, provided by the NASA, [15,16]. The first step in our analysis will be computing the diversity index over the NASA NDVI images. Then we will generate a new NDVI from RED and NIR using Definition 2.4. Over these "relative" NDVI we will compute the diversity indexes. Next we will generate 3-order tensors with just RED and NIR bands. Then we will approximate these tensors with T-HOSVD and ST-HOSVD. Lastly from the approximated tensors we will extract RED and NIR imageries to get "approximated" NDVI. Over these last NDVI we compute the diversity indexes. In conclusion we will measure the error made in estimating diversity when we use approximated NDVI instead of NASA NDVI or relative NDVI.

4.1 Data

From MOD13C2v006 and MOD13A3v006, both sensed by MODIS, cf. [12], and available at [15,16], we select the RED and NIR rasters and the NASA computed NDVI.

MOD13C2v006 is a product characterized by 13 layers, each of which stores an imagery of the Earth with different properties. Each raster is a matrix of 3600 rows and 7200 columns. The side of each pixel corresponds to 5600 m, which is the spatial resolution. The imageries are monthly: they are obtained from the daily data through NASA's algorithms. The chosen data from January 2010 until December 2018 are in `hdf` format.

MOD13A3v006 is a similar product with a higher spatial resolution. While each image from the previous data-set covers the entire Earth's surface, each one from this second data-set covers just $1200 \times 1200\,\mathrm{m}^2$. We select the 20 components, called *granules*, to compose an Europe's map. We download in `GEOTiff` the granules from RED and NIR bands and from the NDVI computed by the NASA, from June 2011 until December 2018. Also in this case they are obtained by the NASA researchers from daily data. Most of the rasters have 4800 rows and 6000 columns. From the MOD13A3v006 data-set, the elements gathered in December 2016 and in December 2017 did not include all the 20 granules. Moreover rasters of December 2012 and December 2015 have the correct dimension, but they have respectively one and two missing areas. Lastly NASA stores the data in 16-bit signed integers.

The next step is creating 3-order tensors. Taking advantage of the GDAL dependencies for python, we converted rasters into matrices, removing the metadata useless for our aims. Then we stored NDVI, RED and NIR into 3-order arrays.

Definition 4.1 Let $\mathcal{T}_E$ be the set of all the tensor $A \in \mathbb{R}^{n_1} \otimes \mathbb{R}^{n_2} \otimes \mathbb{R}^3$ with $n_1 = 4800$ and $n_2 = 6000$ such that

- $A_{\cdot,\cdot,1}$ is the NDVI raster,
- $A_{\cdot,\cdot,2}$ is the RED raster band,
- $A_{\cdot,\cdot,3}$ is the NIR raster band

of Europe data-set for every month and for every year. The cardinality of $\mathcal{T}_E$ is $n_E = 91$.

Similarly let $\mathcal{T}_W$ be the set of all the tensor $A \in \mathbb{R}^{m_1} \otimes \mathbb{R}^{m_2} \otimes \mathbb{R}^3$ with $m_1 = 3600$ and $m_2 = 7200$ such that

- $A_{\cdot,\cdot,1}$ is the NDVI raster,
- $A_{\cdot,\cdot,2}$ is the RED raster band,
- $A_{\cdot,\cdot,3}$ is the NIR raster band

of Earth data-set for every month and for every year. The cardinality of $\mathcal{T}_W$ is $n_W = 108$.

Remark 4.2 Henceforth we will denote with $\mathbb{R}^{\otimes E}$ the tensor space $\mathbb{R}^{n_1} \otimes \mathbb{R}^{n_2} \otimes \mathbb{R}^3$, where $E = (n_1, n_2, 3)$, $n_1 = 4800$ and $n_2 = 6000$ and with $\mathbb{R}^{\otimes W}$ the tensor space $\mathbb{R}^{m_1} \otimes \mathbb{R}^{m_2} \otimes \mathbb{R}^3$, where $W = (m_1, m_2, 3)$, $m_1 = 3600$ and $m_2 = 7200$.

The diversity indexes measured are Rao's and Rényi's ones, both with a window side equal to 11. We maintain the same window side on Europe's and on Earth's images, since the different spatial resolutions lead to similar rasters dimensions.

4.2 NASA and relative NDVI

The first step is computing both Rao and Rényi indexes over the NDVI raster, extracted from the loaded tensor, i.e. over $(A_{k_h})_{\cdot,\cdot,1}$ for every $k_h \in \{1, \ldots, n_h\}$ for every $h \in \{E, W\}$.

Definition 4.3 Let $\mathcal{R}_h$ be the set of Rao index computed over $(A_{k_h})_{\cdot,\cdot,1}$ for every $k_h \in \{1, \ldots, n_h\}$ for every $h \in \{E, W\}$. Similarly let $\mathcal{I}_h$ be the set of Rényi index computed over $(A_{k_h})_{\cdot,\cdot,1}$ for every $k_h \in \{1, \ldots, n_h\}$ for every $h \in \{E, W\}$.

We call R *original estimates* for every $R \in \mathcal{R}_E \cup \mathcal{R}_W \cup \mathcal{I}_E \cup \mathcal{I}_W$.

The obtained images are used as comparison term. Then we compute also an NDVI from the RED and NIR raster, using Definition 2.4, since the algorithm used by NASA for NDVI creation is not public.

Definition 4.4 Let $g_E : \mathbb{R}^{\otimes E} \mapsto \mathbb{M}^{n_1 \times n_2}(\mathbb{R}) \times \mathbb{M}^{n_1 \times n_2}(\mathbb{R})$ and $g_W : \mathbb{R}^{\otimes W} \mapsto \mathbb{M}^{m_1 \times m_2}(\mathbb{R}) \times \mathbb{M}^{m_1 \times m_2}(\mathbb{R})$ be functions such that

$$g_h(A) = (A_{\cdot,\cdot,2}, A_{\cdot,\cdot,3})$$

for every $A \in \mathcal{T}_h$ for every $h \in \{E, W\}$.

Let $M \in \mathbb{R}$ be a real number used as default missing value and let $l : \mathbb{R} \times \mathbb{R} \mapsto \mathbb{R}$ be a function such that

$$l(a, b) = \begin{cases} \frac{a-b}{a+b} & \text{if } a + b \neq 0 \\ M & \text{if } a + b = 0 \end{cases}$$

for every $a, b \in \mathbb{R}$. Let $\bar{l} : \mathbb{M}^{p \times q}(\mathbb{R}) \times \mathbb{M}^{p \times q}(\mathbb{R}) \mapsto \mathbb{M}^{p \times q}(\mathbb{R})$ be such that

$$\bar{l}(A, B) = C \quad \text{such that} \quad C_{i,j} = l(A_{i,j}, B_{i,j})$$

for every $i \in \{1, \ldots, p\}$ for every $j \in \{1, \ldots, q\}$, for every $A, B \in \mathbb{M}^{p \times q}(\mathbb{R})$.

Then

$$f_h = \bar{l} \circ g_h$$

is the function that associates to each tensor $A \in \mathcal{T}_h$ the NDVI raster obtained from $(A_{k_h})_{\cdot,\cdot,2}$ and $(A_{k_h})_{\cdot,\cdot,3}$ for every $k_h \in \{1, \ldots, n_h\}$ for every $h \in \{E, W\}$. Then it is defined the image set

$$\mathcal{N}_h = f_h(\mathcal{T}_h)$$

for every $h \in \{E, W\}$. We call elements of $\mathcal{N}_h$ *self-made NDVI images* for every $h \in \{E, W\}$.

Remark 4.5 Since rasters have only non negative elements for every NIR and RED band, the second case of function l in the previous definition is verified when both elements of NIR and RED rasters are zero. In that case we assigned to NDVI the default missing value, $M = -3000$.

Consequently we performed again index estimation over $\mathcal{N}_h$ elements for every $h \in \{E, W\}$.

Definition 4.6 Let $\tilde{\mathcal{R}}_h$ be the set of Rao index computed over $A_{k_h} \in \mathcal{N}_h$ for every $k_h \in \{1, \ldots, n_h\}$ for every $h \in \{E, W\}$. Similarly let $\tilde{\mathcal{I}}_h$ be the set of Rényi index computed over $A_{k_h} \in \mathcal{N}_h$ for every $k_h \in \{1, \ldots, n_h\}$ for every $h \in \{E, W\}$.

We call R *relative estimates* for every $R \in \tilde{\mathcal{R}}_E \cup \tilde{\mathcal{R}}_W \cup \tilde{\mathcal{I}}_E \cup \tilde{\mathcal{I}}_W$.

Remark 4.7 Henceforth we assume that elements of the same set pairs $(\mathcal{R}_h, \tilde{\mathcal{R}}_h)$ and $(\mathcal{I}_h, \tilde{\mathcal{I}}_h)$ are ordered equally for every $h \in \{E, W\}$.

To present some comparisons between these two types of estimates we compute the error $\|A_j - B_j\|_F$ for every $A_j \in \mathcal{R}_i$ and $B_j \in \tilde{\mathcal{R}}_i$ for every $j \in \{1, \ldots, n_i\}$ for every $i \in \{E, W\}$. Then, we also estimate the error per pixel dividing the error by the number of pixels. Since we are working with huge matrices, this type of distributed error is useful to understand how big is on average the error made pointwise. In Table 1 we report some statistics, where e is the error vector and ep is the error per pixel vector. Besides for every $v \in \mathbb{C}^n$, we set

$$\min v = \min\{v_1, \ldots, v_n\} \quad \text{and} \quad \max v = \max\{v_1, \ldots, v_n\}.$$

Table 1 Rao index statistics for original and relative data

Dataset	$\mathbb{E}[e]$	$\mathbb{E}[ep]$	$\mathbb{V}\text{ar}[e]$	$\mathbb{V}\text{ar}[ep]$	min ep	max ep
Europe	197877.819	0.0069	409993378560.6464	0.0005	0.0013	0.1777
World	1731817.3949	0.0668	687978783275.7339	0.001	0.0155	0.1678

Table 2 Rényi index statistics for original and relative data

Dataset	$\mathbb{E}[e]$	$\mathbb{E}[ep]$	$\mathbb{V}\text{ar}[e]$	$\mathbb{V}\text{ar}[ep]$	min ep	max ep
Europe	132517.8038	0.0046	420678175017.4135	0.0005	0.0003	0.1777
World	782043.4085	0.0302	148701528527.3207	0.0002	0.0112	0.0688

We find on average a 0.6% error per pixel for Europe Rao's Q estimation using self-made NDVI instead of NASA NDVI, while the error is on average of 6% per pixel when Rao is computed over Earth NDVIs. Both the unbiased variance are very small, which means that errors are not very different from the mean. Also the minimum approximation error is greater for Earth than for Europe data. We suppose that this is linked to the higher amount of water surface in World rasters than in the Europe's ones.

Similarly we compute the error $\left\| A_j - B_j \right\|_F$ for every $A_j \in \mathcal{I}_i$ and $B_j \in \tilde{\mathcal{I}}_i$ for every $j \in \{1, \ldots, n_i\}$ for every $i \in \{E, W\}$. With the same notation previously introduced, we present some statistics for Rényi index in Table 2.

In this case the average error per pixel for both data-sets is smaller than the previous one. One possible explication could be that Rényi index takes into account only frequencies while Rao index considers both pixel values and their frequencies. However in this case variance in World error is smaller than in Europe, while in the Rao case there is an opposite trend. Moreover, the minimum and the maximum approximation errors in the World data-set for both the indexes are due to the same element, i.e. April 2018 and May 2014 respectively. This consideration holds also for Europe data-set: the minimum error comes out from April 2018 and the maximum error from December 2012.

4.3 Approximated NDVI

Before applying the approximation tensors codes, described in the Appendix, Sects. 6.2 and 6.3, we have to highlight one limit of the python function `svds`. It takes as additional parameter `k`, which is the rank of the approximation and which has to be strictly lower than both the dimensions of the given matrix. For instance, if we had passed just a 3-order tensor such that $n_3 = 2$, for the third flattening we would have fixed `k` equal to 1, getting a vector: this is a low order tensor for our aims. Therefore we decided to increase n_3 up to 3, adding another matrix to our tensor: in the first case we took twice RED band raster and once NIR one, in the second case we took twice NIR band and once the RED one.

Definition 4.8 Let $g_{R,h} : \mathcal{T}_h \subset R^{\otimes h} \mapsto \mathbb{R}^{\otimes h}$ be the function that associates to each tensor $A \in \mathcal{T}_h$ a tensor B such that

$$B_{\cdot,\cdot,1} = B_{\cdot,\cdot,2} = A_{\cdot,\cdot,2} \quad \text{and} \quad B_{\cdot,\cdot,3} = A_{\cdot,\cdot,3},$$

Table 3 Rate of compression

Rank	Europe			Earth		
	Rel	Abs. R	Abs. N	Rel	Abs. R	Abs. N
10	0.0019	0.0013	0.0013	0.0021	0.0014	0.0014
50	0.0095	0.0063	0.0063	0.0105	0.007	0.007
100	0.0191	0.0127	0.0127	0.0212	0.0141	0.0141
500	0.1024	0.0683	0.0683	0.1138	0.0759	0.0759
1000	0.2222	0.1481	0.1481	0.2469	0.1646	0.1646

for every $h \in \{E, W\}$. Then

$$\mathcal{T}_{R,h} = g_{R,h}(\mathcal{T}_h)$$

for every $h \in \{E, W\}$.

Similarly let $g_{N,h} : \mathcal{T}_h \subset \mathbb{R}^{\otimes h} \mapsto \mathbb{R}^{\otimes h}$ be the function that associates to each tensor $A \in \mathcal{T}_h$ a tensor B such that

$$B_{\cdot,\cdot,1} = B_{\cdot,\cdot,3} = A_{\cdot,\cdot,3} \quad \text{and} \quad B_{\cdot,\cdot,2} = A_{\cdot,\cdot,2},$$

for every $h \in \{E, W\}$. Then

$$\mathcal{T}_{N,h} = g_{N,h}(\mathcal{T}_h)$$

for every $h \in \{E, W\}$,

To compute T-HOSVD and ST-HOSVD, we fix five multilinear target ranks.

Definition 4.9 Let $\mathcal{R} = \{10, 50, 100, 500, 1000\}$ be a set with the given order fixed, then the *target multilinear rank* we choose are

$$r_j = (i_j, i_j, 2)$$

for every $i_j \in \mathcal{R}$ for every $j \in \{1, \ldots, 5\}$. Let $\mathcal{T}_{k,h,j}$ be the set of T-HOSVD approximation at multilinear rank r_j of tensors from the set $\mathcal{T}_{k,h}$ for every $h \in \{E, W\}$, for every $k \in \{N, R\}$ and for every $j \in \{1, \ldots, 5\}$. Similarly let $\mathcal{S}_{k,h,j}$ be the set of ST-HOSVD approximation.

Before presenting the results related to indexes computation, we show data about storage memory use. Since it strictly depends on the core tensor and on the projectors dimensions, which are the same for T-HOSVD and ST-HOSVD, we report only one table for each dataset. For each tensor $A \in \mathcal{T}_{k,W}$ the ratio between the memory used for storing the core tensor and the projectors over the memory used for storing A is the same, for every $k \in \{N, R\}$. For tensors of $\mathcal{T}_{k,E}$ it holds the same, except for those elements composed by a lower number of granules, for every $k \in \{N, R\}$. Since they were 2 over 91, we neglect them and in Table 3 we present the ratios of memory usage for each rank approximation. We call these ratios *absolute compression ratios*, because they have as denominator the memory used to store two time RED band and once NIR band, or vice-versa. In Table 3 *Abs. R* refers to the ratios for the repeated RED band raster and *Abs. N* refers to the NIR case. Beside we list also a *relative compression ratio*, where the denominator is the amount of memory necessary to store once RED and once NIR band: this ratio goes in the *Rel* column. For the sake of clarity, in Table 3 it is reported for the rank only the significant component, i.e. i_j for every $i_j \in \mathcal{R}$ for every $j \in \{1, \ldots, 5\}$.

Even with the greater component of the multilinear rank, we use just a small percentage of memory compared to the amount needed for storing the entire tensor. Moreover, even the relative ratio at the highest multilinear rank show a significant saving in memory use.

After generating new tensors and approximating them, we compute new NDVIs through function f_h of Definition 4.4 applied on $\mathcal{T}_{k,h,j} \cup \mathcal{S}_{k,h,j}$ for every $h \in \{E, W\}$, for every $k \in \{N, R\}$ and for every $j \in \{1, \ldots, 5\}$. We compute the diversity indices over them. Finally, we measure the difference in estimating diversity from approximated NDVI and NASA or self-made NDVI.

Definition 4.10 Let $\mathcal{N}_{k,h,T,j} = f_h(\mathcal{T}_{k,h,R,j})$ and let $\mathcal{N}_{k,h,S,j} = f_h(\mathcal{S}_{k,h,R,j})$ every $j \in \{1, \ldots, 5\}$ and for every $h \in \{E, W\}$. We call elements of $\mathcal{N}_{k,h,T,j} \cup \mathcal{N}_{k,h,S,j}$ *approximated NDVIs*.

4.3.1 Rényi index

Definition 4.11 Let $\mathcal{I}_{R,h,k,j}$ be the set of Rényi index computed over elements of $\mathcal{N}_{R,h,t,j}$ for every $h \in \{E, W\}$, for every $t \in \{T, S\}$ and for every $j \in \{1, \ldots, 5\}$. They are called i_j*-approximated estimates* for every $i_j \in \mathcal{R}$ and for every $j \in \{1, \ldots, 5\}$.

Remark 4.12 Notice that there are 4 indexes for approximated estimates set:

1st **index** indicates the repeated matrix in the starting tensor R for RED and N for NIR;
2nd **index** the belonging data-set, E for Europe and W for Earth;
3rd **index** the approximation algorithm, T for T-HOSVD and S for ST-HOSVD;
4th **index** is associated with the target multilinear rank.

The final step is the computation of the error with respect to the original estimates, i.e.

$$\left\| A_k - C_{k,j} \right\|$$

for every $A_k \in \mathcal{I}_h$ and for every $C_{k,j} \in \mathcal{I}_{R,h,T,j} \cup \mathcal{I}_{R,h,S,j}$, for every $k \in \{1, \ldots, n_h\}$, for every $j \in \{1, \ldots, 5\}$ and for every $h \in \{E, W\}$. Moreover, we compute the error with respect to relative estimates, i.e.

$$\left\| B_k - C_{k,j} \right\|$$

for every $B_k \in \tilde{\mathcal{I}}_h$ and for every $C_{k,j} \in \mathcal{I}_{R,h,T,j} \cup \mathcal{I}_{R,h,S,j}$, for every $k \in \{1, \ldots, n_h\}$, for every $j \in \{1, \ldots, 5\}$ and for every $h \in \{E, W\}$.

Considering the errors per pixel with respect to original estimates, stored in vector epO and relative estimates, in vector epR for Europe data-set, for both the decomposition techniques (and each target multilinear rank), in Table 4 we report as rank only the significant component of the multilinear rank.

Remark 4.13 To simplify the discussion henceforth the i_j*-original error* will be the error between original estimate and i_j-approximated estimate, while the i_j*-relative error* will be the error between relative estimate and i_j-approximated estimate for every $i_j \in \mathcal{R}$.

ST-HOSVD original and relative error per pixel are lower than T-HOSVD original and relative error per pixel, when the first two components of target multilinear rank are greater than or equal to 500. The variance of both errors is quite low, even if it increases in the last three multilinear ranks. The minimum relative error and the minimum original error are

Table 4 Statistics for Rényi index over $N \in \mathcal{n}_{R,E,t,j}$

Rank	T-HOSVD					ST-HOSVD				
	10	50	100	500	1000	10	50	100	500	1000
$\mathbb{E}[epO]$	0.1576	0.0914	0.0801	0.0807	0.0766	0.1597	0.0917	0.0803	0.0798	0.0755
$\mathbb{V}\text{ar}[epO]$	0.0003	0.0002	0.0002	0.0013	0.0013	0.0003	0.0002	0.0002	0.0013	0.0013
$\mathbb{E}[epR]$	0.1572	0.0913	0.0807	0.0821	0.079	0.1593	0.0918	0.0811	0.0815	0.0779
$\mathbb{V}\text{ar}[epR]$	0.0003	0.0002	0.0003	0.0014	0.0014	0.0003	0.0002	0.0003	0.0014	0.0014
min epO	0.1164	0.0641	0.0537	0.0466	0.0383	0.1164	0.0646	0.0542	0.0459	0.0367
min epR	0.1164	0.0641	0.0537	0.0466	0.0383	0.1164	0.0646	0.0542	0.0459	0.0367
max epO	0.1864	0.1377	0.1136	0.194	0.1999	0.1931	0.1371	0.1104	0.1932	0.199
max epR	0.1864	0.147	0.1618	0.194	0.1999	0.1931	0.1479	0.1679	0.1932	0.199

Table 5 Statistics for Rényi index over $N \in \mathcal{n}_{R,W,t,j}$

Rank	T-HOSVD					ST-HOSVD				
	10	50	100	500	1000	10	50	100	500	1000
$\mathbb{E}[epO]$	0.1376	0.0943	0.0875	0.0626	0.0584	0.1383	0.0946	0.0876	0.0625	0.058
$\mathbb{V}\text{ar}[epO]$	0.0001	0.0001	0.0002	0.0002	0.0001	0.0001	0.0001	0.0002	0.0002	0.0001
$\mathbb{E}[epR]$	0.1358	0.091	0.0846	0.0545	0.0491	0.1365	0.0912	0.0847	0.0545	0.0485
$\mathbb{V}\text{ar}[epR]$	0.0001	0.0001	0.0002	0.0002	0.0	0.0001	0.0001	0.0002	0.0002	0.0
min epO	0.1156	0.0758	0.0668	0.048	0.0471	0.1156	0.076	0.0672	0.0483	0.0468
min epR	0.1155	0.0748	0.0655	0.0421	0.0414	0.1155	0.075	0.0658	0.0421	0.0411
max epO	0.1599	0.1182	0.1255	0.0967	0.0813	0.1601	0.1185	0.1253	0.0973	0.0811
max epR	0.155	0.1134	0.1263	0.0968	0.0744	0.1551	0.1138	0.126	0.0974	0.0731

equal up to the fourth decimal digit. They also decrease when the components of multilinear rank increase. On the other hand, the maximum of relative errors and the maximum of original errors do not coincide. Besides, they increase significantly in the fourth and fifth approximation. Finally, the average relative error per pixel is frequently slightly greater than the original one. For T-HOSVD this inequality between original and relative error average happens from the third approximation, while for the ST-HOSVD from the second one. The difference grows for increasing multilinear rank components. We expected an inverse pattern. However, we have to remark that Rényi index takes into account only raster values frequencies, neglecting the values themselves. In addiction from the complete data, we notice that the relative error of elements with missing granules, tensors of December 2012 and December 2015, is more than 3 times the original error.

Considering the Earth data-set, similarly in vector epO, we show the errors per pixel with respect to original Rényi estimates, while in epR with respect to relative estimates for each target multilinear rank, see Table 5.

Even in this case, on average, ST-HOSVD technique leads to lower original and relative errors than T-HOSVD, for the last two and for the last one target multilinear ranks, respectively. Moreover, we underline that relative error average is slightly lower than the original one, as expected. Minimum and maximum of both errors decrease with increasing multilinear rank components. The most stunning value is the variance of relative error per pixel at the last approximation. For both T-HOSVD and ST-HOSVD it is lower than 10^{-4}.

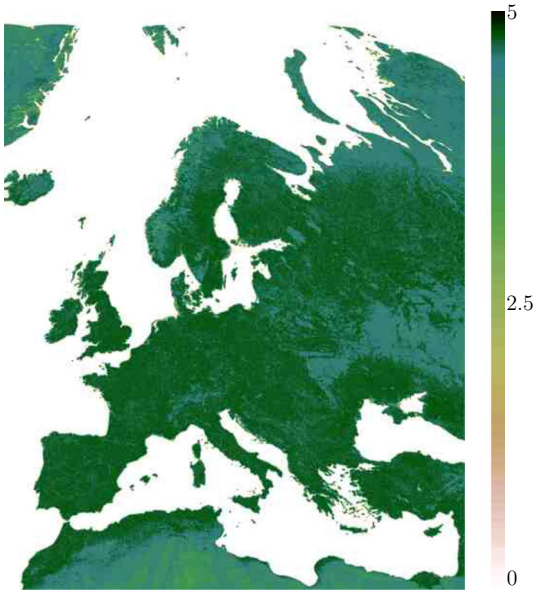


Fig. 1 Rényi index computed over NASA NDVI of February 2013

We include the images associated to Rényi index in the five approximations for the Europe worst case and the Earth best case, both from ST-HOSVD approximation technique.

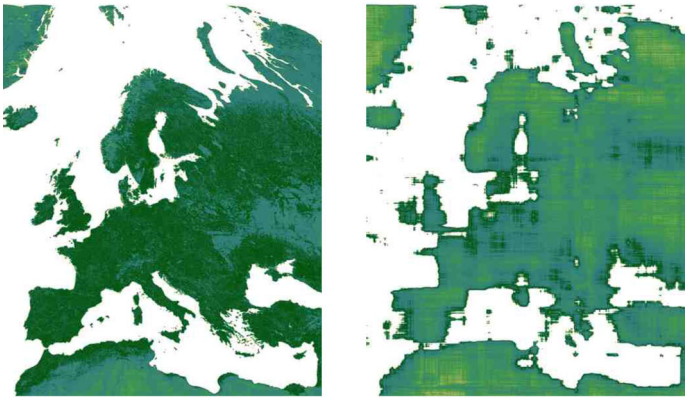
Example 4.14 Looking at the Rényi index computed over Europe NDVI of February 2013 as it is in Fig. 1, diversity is quite high, near 4.5, almost everywhere in Europe. However as remarked in [34], Rényi index tends to overestimate diversity. Furthermore, Rényi index computed over self-made NDVI did not differ from its computation over NASA NDVI.

In Fig. 2A, the same index computed over self-made NDVI and the approximated Rényi estimates at different multilinear ranks is shown. When the first two components of the multilinear rank grows, in the Rényi estimation some new noising elements appeared, leading to high errors. This type of phenomenon deserves further analysis. However in the internal land of Europe, the diversity estimation is close to the relative and original one, for multilinear rank components strictly greater than 100.

The element of the Earth data-set is shown in the next example.

Example 4.15 Firstly we present in Fig. 3 the Rényi estimate over NASA NDVI of October 2017. As we said in the Example 4.14 Rényi index provides high diversity values. Indeed also in this case there are many Earth areas with a diversity value close to 4.5. The same index over self-made and approximated NDVI is shown in Fig. 4. Despite the small printing dimensions, at first glance, no differences between the last three index approximations and the index of Fig. 3 can be perceived. However, with a closer observation, islands of the Pacific ocean disappear in first approximation and they partially reappear in the last two images. We may conclude that the original and relative error is in this case linked with these missing territories, but also with the different evaluation of diversity in Amazon area, for example.

Rényi index over approximated NDVI of $\mathcal{N}_{N,h,t,j}$ for every $h \in \{E, W\}$, for every $t \in \{T, S\}$ and $j \in \{1, \ldots, 5\}$ is further computed.



(A) Relative approximation

(B) Component rank 10 compression

(C) Component rank 50 compression

(D) Component rank 100 compression

(E) Component rank 500 compression

(F) Component rank 1000 compression

Fig. 2 Approximation of Rényi index for February 2013, from NDVI of $\mathcal{n}_{R,E,S,j} \cup \mathcal{n}_E$

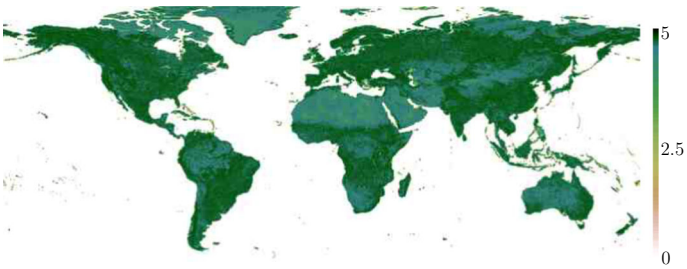
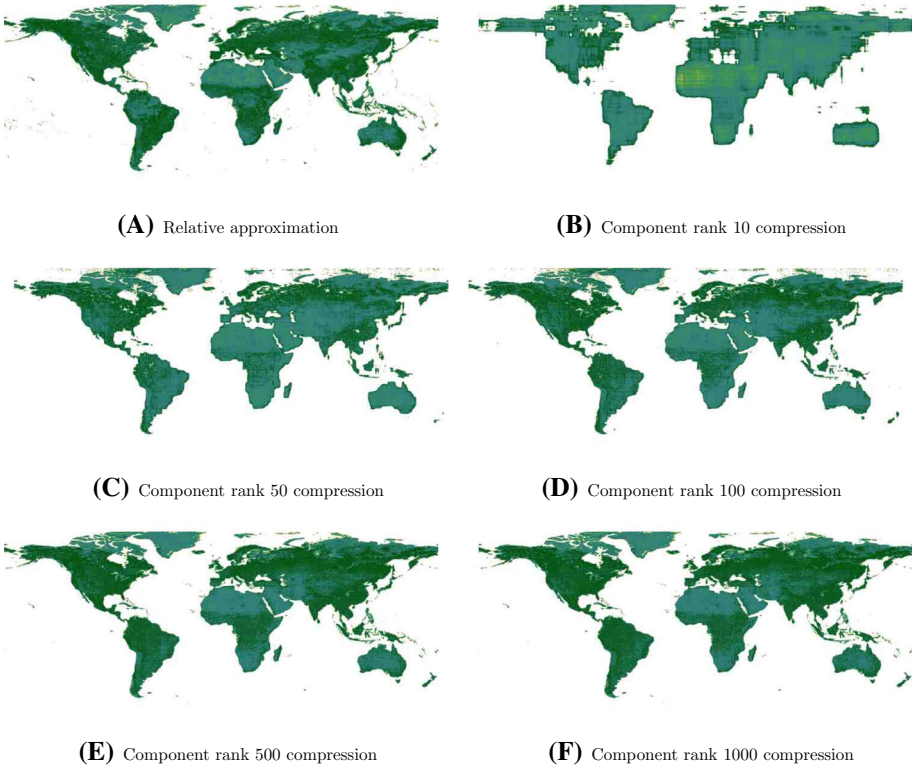


Fig. 3 Rényi index computed over NASA NDVI of October 2017



(A) Relative approximation

(B) Component rank 10 compression

(C) Component rank 50 compression

(D) Component rank 100 compression

(E) Component rank 500 compression

(F) Component rank 1000 compression

Fig. 4 Approximation of Rényi index for October 2017, from NDVI of $\mathcal{n}_{R,W,S,j} \cup \mathcal{n}_W$

Definition 4.16 Let $\mathcal{I}_{N,h,t,j}$ be the set of Rényi index computed over elements of $\mathcal{n}_{N,h,t,j}$ for every $h \in \{E, W\}$, for every $t \in \{T, S\}$ and for every $j \in \{1, \ldots, 5\}$. We decide to call these i_j*-approximated estimates* for every $i_j \in \mathcal{R}$ and for every $j \in \{1, \ldots, 5\}$.

As before, we compute the error with respect to the original estimates, i.e.

$$\left\| A_k - C_{k,j} \right\|$$

for every $A_k \in \mathcal{I}_h$ and for every $C_{k,j} \in \mathcal{I}_{N,h,T,j} \cup \mathcal{I}_{N,h,S,j}$, for every $k \in \{1, \ldots, n_h\}$, for every $j \in \{1, \ldots, 5\}$ and for every $h \in \{E, W\}$. Moreover we compute the error with respect

Table 6 Statistics for Rényi index over $N \in \mathcal{n}_{N,E,t,j}$

Rank	T-HOSVD					ST-HOSVD				
	10	50	100	500	1000	10	50	100	500	1000
$\mathbb{E}[epO]$	0.1569	0.09	0.0782	0.0794	0.076	0.1586	0.0903	0.0783	0.0787	0.0751
$\mathbb{V}\text{ar}[epO]$	0.0003	0.0002	0.0002	0.0013	0.0013	0.0003	0.0002	0.0002	0.0013	0.0013
$\mathbb{E}[epR]$	0.1563	0.0894	0.0782	0.0806	0.0783	0.158	0.0897	0.0785	0.08	0.0775
$\mathbb{V}\text{ar}[epR]$	0.0003	0.0002	0.0002	0.0014	0.0015	0.0004	0.0002	0.0002	0.0015	0.0014
min epO	0.1128	0.0623	0.0518	0.0449	0.0436	0.1143	0.0625	0.0521	0.0438	0.0428
min epR	0.1128	0.0623	0.0518	0.0449	0.0436	0.1143	0.0625	0.0521	0.0438	0.0428
max epO	0.1827	0.1462	0.1275	0.1936	0.1991	0.1855	0.1457	0.1222	0.1932	0.1988
max epR	0.1827	0.1363	0.1484	0.1936	0.1991	0.1855	0.1373	0.1528	0.1932	0.1988

to relative estimates, i.e.

$$\left\| B_k - C_{k,j} \right\|$$

for every $B_k \in \tilde{\mathcal{G}}_h$ and for every $C_{k,j} \in \mathcal{G}_{N,h,T,j} \cup \mathcal{G}_{N,h,S,j}$, for every $k \in \{1, \ldots, n_h\}$, for every $j \in \{1, \ldots, 5\}$ and for every $h \in \{E, W\}$.

Lastly we report some statistics about i_j-original and i_j-relative errors per pixel, stored in vector epO and epR for each $i_j \in \mathcal{R}$. Europe related statistics are shown in Table 6.

Almost every consideration for statistics of approximated estimates of $\mathcal{G}_{R,E,t,j}$ for every $t \in \{T, S\}$ and for every $j \in \{1, \ldots, 5\}$ holds also in this case. However we can notice that on average the 100 relative and original error are lower than the 500 one, but this is not true anymore for 1000 relative and original errors. Moreover, when the first two multilinear rank components are strictly smaller that 500, T-HOSVD relative and original errors are lower than ST-HOSVD ones. For the statistics over $\mathcal{G}_{R,E,t,j}$ elements, this consideration holds only for relative error at the third multilinear rank. The remarks about not full granules elements are true also in this case. At each multilinear rank both the original and the relative errors on average are smaller in this second case, i.e. applying the procedure to tensors where the NIR band raster is repeated.

Concerning the statistics related to the Earth approximated estimates, we provide Table 7 with mean, variance, min and max for i_j-original and i_j-relative errors per pixel, stored in vector epO and epR, respectively, for each $i_j \in \mathcal{R}$.

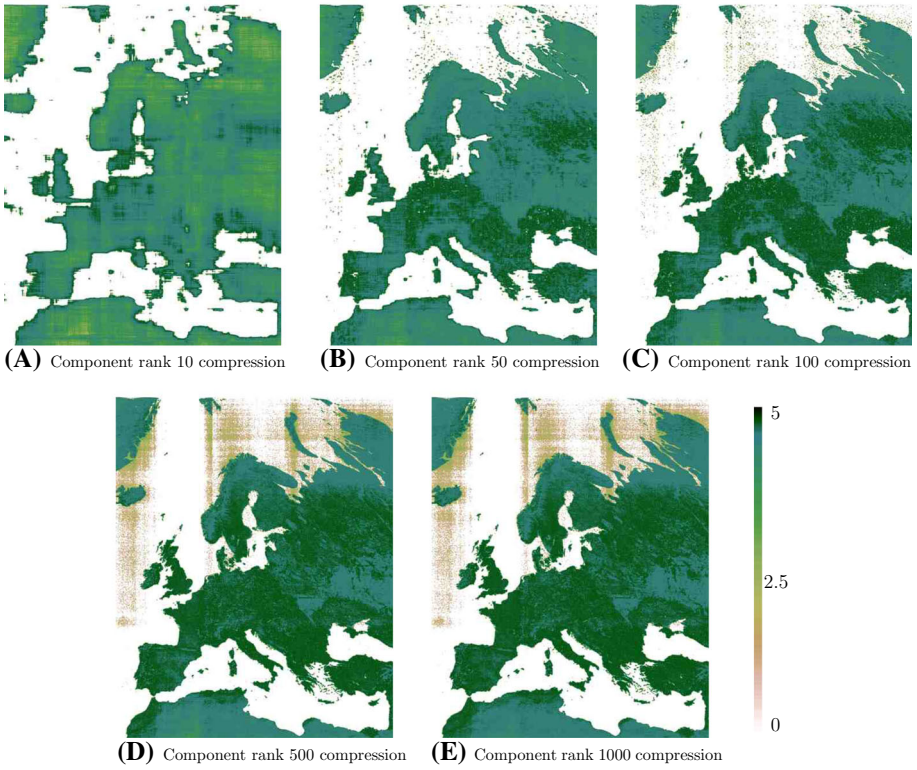
The considerations presented for $\mathcal{G}_{R,W,t,j}$ error statistics hold true also in this case. Variance is lower than 10^{-4} for 1000-relative error of ST-HOSVD. Moreover, also in this case, ST-HOSVD is convenient only when the first two multilinear rank components are greater than 500. Finally, the original and relative errors on average are smaller in this second case, i.e. when we apply our method to tensors where NIR band is repeated.

The Rényi index images calculated on Europe NDVI of February 2013 and over Earth NDVI of October 2017 are obtained starting from the correspondent tensors of $\mathcal{T}_{N,h}$ for $h \in \{E, W\}$.

Example 4.17 In Example 4.14 we present the Rényi index image which realizes the highest original and relative error, starting from RED repeated band. Figure 5 presents Rényi index for the same element, with approximation obtained from NIR band repeated. We remark that also starting from twice NIR and once RED band tensor element of February 2013 leads to the highest original and relative error.

Table 7 Statistics for Rényi index over $N \in \mathcal{n}_{N,W,t,j}$

Rank	T-HOSVD					ST-HOSVD				
	10	50	100	500	1000	10	50	100	500	1000
$\mathbb{E}[epO]$	0.1351	0.0915	0.084	0.0601	0.0556	0.1355	0.0917	0.0842	0.0601	0.0553
$\mathbb{V}\text{ar}[epO]$	0.0001	0.0001	0.0002	0.0002	0.0001	0.0001	0.0001	0.0002	0.0002	0.0001
$\mathbb{E}[epR]$	0.1334	0.0879	0.0807	0.052	0.046	0.1338	0.0882	0.0809	0.052	0.0455
$\mathbb{V}\text{ar}[epR]$	0.0001	0.0001	0.0002	0.0002	0.0001	0.0001	0.0001	0.0002	0.0002	0.0
min epO	0.1119	0.0727	0.0638	0.0443	0.0424	0.112	0.0728	0.0641	0.0445	0.0422
min epR	0.1114	0.0715	0.0623	0.0385	0.0382	0.1115	0.0716	0.0628	0.0387	0.0379
max epO	0.1564	0.1154	0.121	0.0963	0.0792	0.1574	0.116	0.1209	0.0964	0.0791
max epR	0.1536	0.1098	0.1218	0.0965	0.0731	0.1536	0.1099	0.1217	0.0966	0.0719



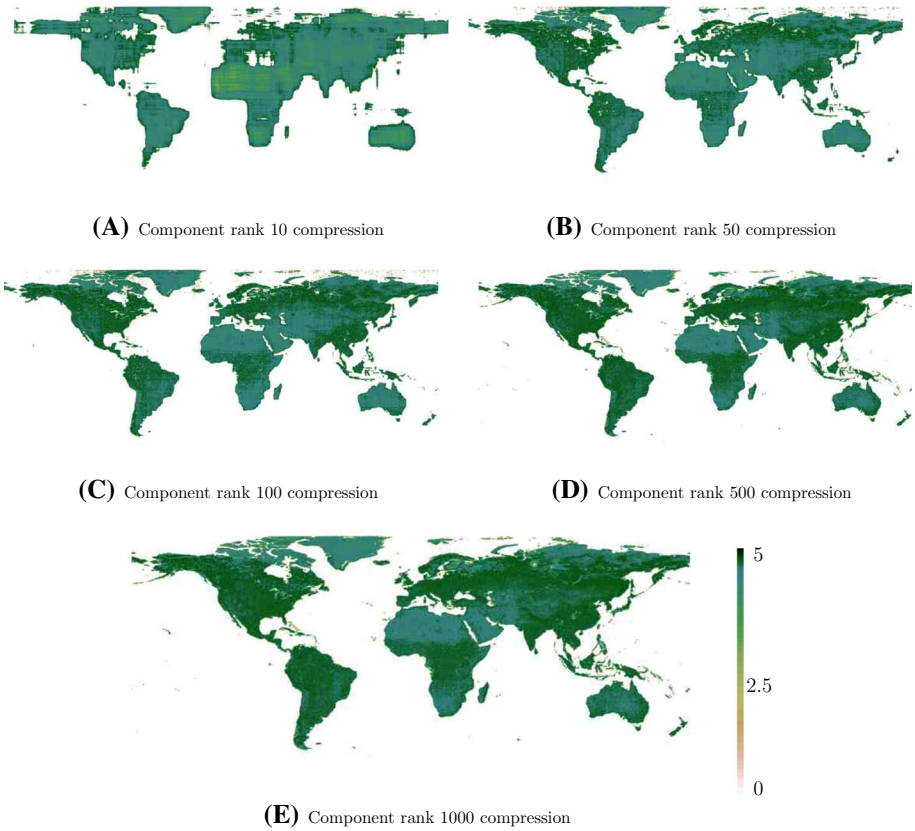
(A) Component rank 10 compression **(B)** Component rank 50 compression **(C)** Component rank 100 compression

(D) Component rank 500 compression **(E)** Component rank 1000 compression

Fig. 5 Approximation of Rényi index for February 2013, from NDVI of $\mathcal{n}_{R,E,S,j}$

Once again, we observe an increasing presence of noise in Northern Europe for growing multilinear rank components.

Example 4.18 Similarly, we display in Fig. 6 the approximation of Rényi index for Earth element of October 2017, obtained from tensors where is repeated twice the NIR band. As in Example 4.14, the number of detected territories in the Pacific ocean grows when the first two multilinear rank components grow. Moreover comparing Fig. 4E and Fig. 6E, the presence



(A) Component rank 10 compression

(B) Component rank 50 compression

(C) Component rank 100 compression

(D) Component rank 500 compression

(E) Component rank 1000 compression

Fig. 6 Approximation of Rényi index for October 2017, from NDVI of $\mathcal{N}_{N,W,S,j}$

of Pacific islands is greater in the second one. Indeed, this is the element that minimizes the original error also in this second proceeding way. Therefore, even in this case, choosing a starting tensor with repeated NIR band is more convenient that starting with repeated RED band.

4.3.2 Rao's Q index

Definition 4.19 Let $\mathcal{R}_{R,h,k,j}$ be the set of Rao index computed over elements of $\mathcal{N}_{R,h,t,j}$ for every $h \in \{E, W\}$, for every $t \in \{T, S\}$ and for every $j \in \{1, \ldots, 5\}$. We call these i_j*-approximated estimates* for every $i_j \in \mathcal{R}$ and for every $j \in \{1, \ldots, 5\}$.

As illustrated previously, we compute the error with respect to the original estimates, i.e.

$$\left\| A_k - C_{k,j} \right\|$$

for every $A_k \in \mathcal{R}_h$ and for every $C_{k,j} \in \mathcal{R}_{R,h,T,j} \cup \mathcal{R}_{R,h,S,j}$, for every $k \in \{1, \ldots, n_h\}$, for every $j \in \{1, \ldots, 5\}$ and for every $h \in \{E, W\}$. Moreover, we compute the error with

Table 8 Statistics for Rao index over $N \in \mathcal{N}_{R,E,t,j}$

Rank	T-HOSVD					ST-HOSVD				
	10	50	100	500	1000	10	50	100	500	1000
$\mathbb{E}[epO]$	0.6419	0.3621	0.2944	0.2059	0.1922	0.6328	0.3604	0.293	0.2081	0.1951
$\mathbb{V}\text{ar}[epO]$	0.0044	0.0012	0.001	0.0031	0.003	0.0038	0.0011	0.001	0.003	0.003
$\mathbb{E}[epR]$	0.6424	0.3633	0.2961	0.2083	0.1953	0.6333	0.3615	0.2946	0.2104	0.1982
$\mathbb{V}\text{ar}[epR]$	0.0043	0.0013	0.0013	0.0038	0.0045	0.0037	0.0013	0.0013	0.0038	0.0045
min epO	0.4819	0.274	0.2194	0.1306	0.1128	0.4825	0.2724	0.2185	0.1326	0.1144
min epR	0.4819	0.274	0.2193	0.1306	0.1127	0.4825	0.2724	0.2185	0.1326	0.1144
max epO	0.7559	0.4081	0.3591	0.3858	0.3894	0.7246	0.4065	0.3722	0.3871	0.3917
max epR	0.7558	0.465	0.4659	0.4486	0.5599	0.7246	0.4746	0.4734	0.4742	0.5618

respect to relative estimates, i.e.

$$\left\| B_k - C_{k,j} \right\|$$

for every $B_k \in \tilde{\mathcal{R}}_h$ and for every $C_{k,j} \in \mathcal{R}_{R,h,T,j} \cup \mathcal{R}_{R,h,S,j}$, for every $k \in \{1, \ldots, n_h\}$, for every $j \in \{1, \ldots, 5\}$ and for every $h \in \{E, W\}$.

The statistics about i_j-original and i_j-relative error per pixel are stored in vector epO and epR for each $i_j \in \mathcal{R}$ and reported Table 8. Firstly we list information for Europe related data. The most evident aspect is the high mean of both original and relative errors, even when the components of multilinear rank are growing. Indeed, this average is close to a 20% of error per pixel. Comparing it with the average error per pixel made for Rényi index might lead us to conclude that, in this case, HOSVD is not well performing. However, it might be important to recall that: i) the Rao's Q index takes into account also the values of NDVI, not only their frequencies, and ii) at multilinear rank r_5 we are keeping nearly 15% of the total information, as highlighted in Table 3. Therefore, even if results are not as good as those of the Rényi index case, we do not exclude the power of this method for Rao index computation. In fact (in the best case) we find both the original and the relative errors per pixel to be near 11%.

In the previous analysis T-HOSVD performed better when multilinear rank components were small with respect to ST-HOSVD, in the present case T-HOSVD provides better results than ST-HOSVD when the first two multilinear rank components are greater than 500. Moreover, in this case the relative error is on average greater than the original one. This phenomenon is linked partially to the two incomplete elements, December 2012 and 2015. Indeed, for these elements, the relative error is slightly less than twice the original error.

The statistics for the original and relative errors of Rao's Q estimates for Earth data-set are presented in Table 9, where epO is the vector of the errors per pixel computed with respect to original estimates, while epR is the error per pixel with respect to relative estimates.

Even with the highest multilinear rank components, the average original and relative errors per pixel are quite high if compared with the ones in the Rényi index case. Notice that 16% error per pixel on average is obtained using only 16.4% of the total information. However, even in the best case, the relative error per pixel is 14.5%, in line with the minimum error per pixel in the Europe case.

Besides, the relative error per pixel is slightly lower than the original ones as in the previous Earth case. As before, when the first two components of the multilinear rank are

Table 9 Statistics for Rao index over $N \in \mathcal{n}_{R,W,t,j}$

Rank	T-HOSVD					ST-HOSVD				
	10	50	100	500	1000	10	50	100	500	1000
$\mathbb{E}[epO]$	0.5442	0.3476	0.2949	0.1845	0.1735	0.5416	0.3467	0.2945	0.1851	0.1752
$\mathbb{V}\text{ar}[epO]$	0.0021	0.0007	0.0007	0.0004	0.0002	0.0018	0.0007	0.0007	0.0004	0.0002
$\mathbb{E}[epR]$	0.5425	0.3445	0.2917	0.175	0.1628	0.54	0.3436	0.2913	0.1756	0.1646
$\mathbb{V}\text{ar}[epR]$	0.0019	0.0006	0.0006	0.0003	0.0001	0.0017	0.0005	0.0006	0.0003	0.0001
min epO	0.4614	0.2943	0.2472	0.1551	0.1523	0.4613	0.294	0.2475	0.1555	0.1539
min epR	0.4625	0.2952	0.2473	0.1509	0.1454	0.4623	0.295	0.2473	0.1513	0.1473
max epO	0.613	0.3992	0.3498	0.2442	0.2295	0.5991	0.3994	0.3497	0.2435	0.2306
max epR	0.6124	0.3859	0.3512	0.2396	0.1895	0.5971	0.3834	0.3511	0.2387	0.1926

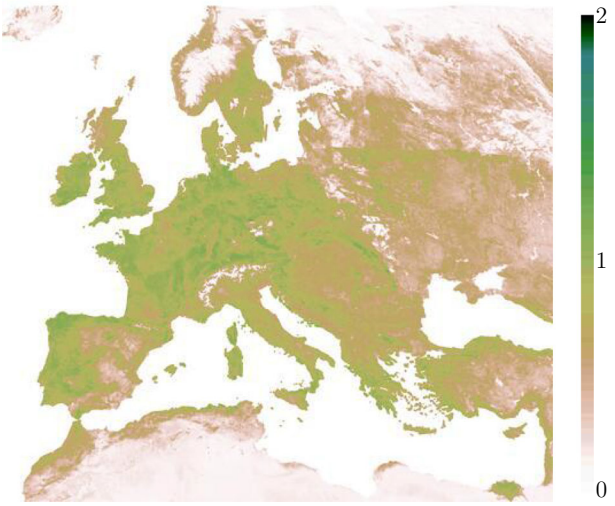


Fig. 7 Rao index computed over NASA NDVI of December 2014

lower than 500, ST-HOSVD techniques leads to better results than T-HOSVD. Moreover, as in the previous case, the variance decreases when the first two multilinear rank components increase.

Before moving to the analysis of the error related to the Rao's Q index over $\mathcal{n}_{N,h,p,j}$ for every $h \in \{E, W\}$, for every $p \in \{T, S\}$ and for every $j \in \{1, \ldots, 5\}$, we show the best case for the Rao's Q index for Europe and the worst case for Earth data-set in the following examples. In both cases we use T-HOSVD as approximation technique.

Example 4.20 Looking at Rao index computed over Europe NDVI of December 2014, in Fig. 7, we notice that diversity is much lower than the one predicted by Rényi index, 4.14. Besides, even in this case the value of the Rao's Q index computed over self-made NDVIs is close to the one computed over NASA NDVI.

The same index computed over self-made NDVI and the approximated Rao's Q estimated at different multilinear ranks is reported in Fig. 8A. It is hard to distinguish the Rao's Q

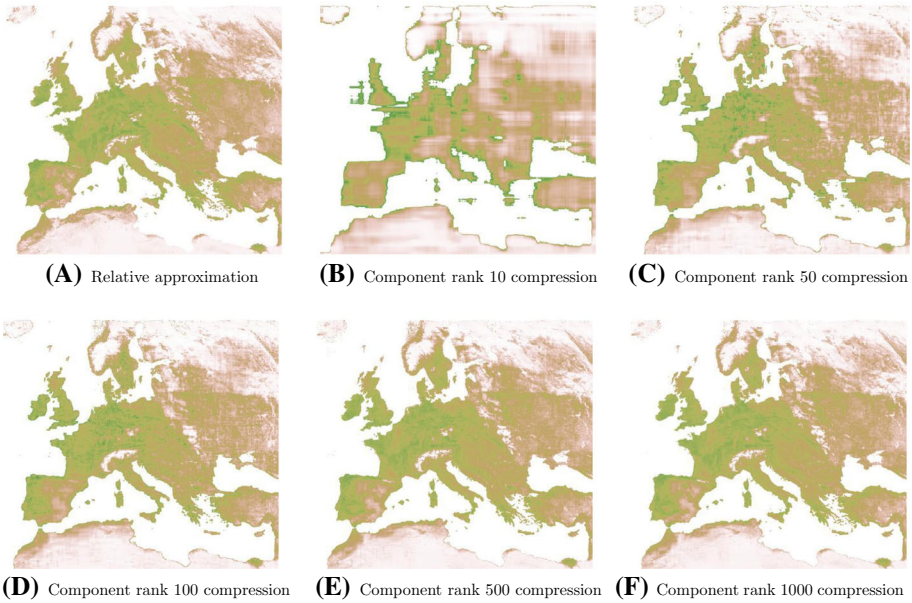


Fig. 8 Approximation of Rao index for December 2014, from NDVI of $\mathcal{n}_{R,E,S,j} \cup \mathcal{n}_E$

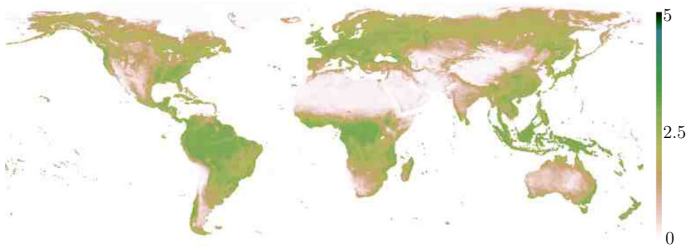


Fig. 9 Rao index computed over NASA NDVI of December 2014

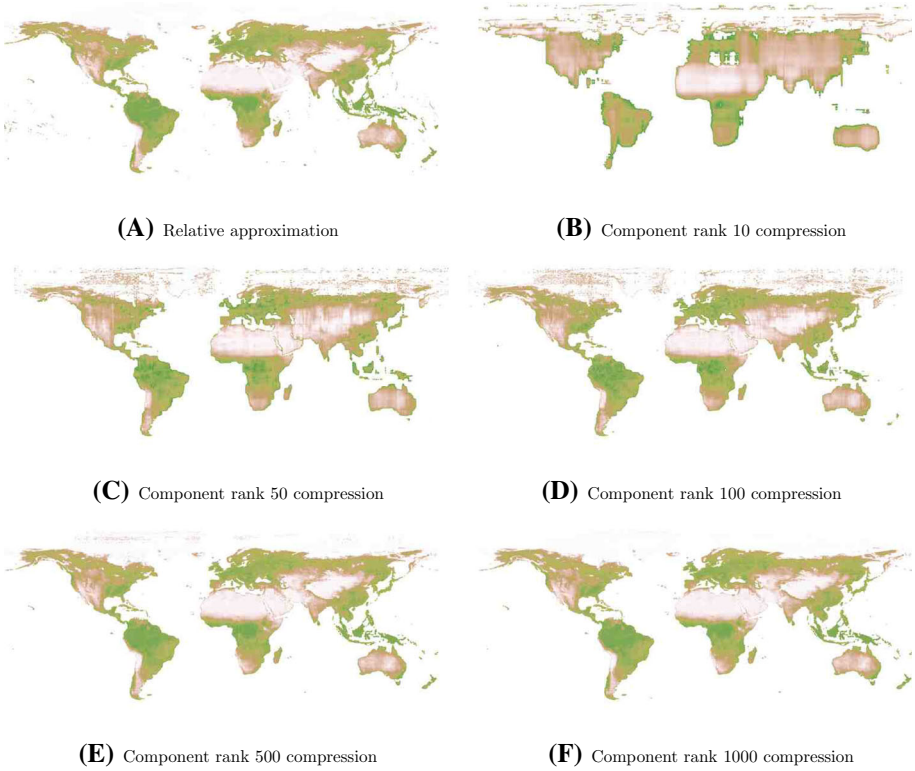
index computed over NASA NDVI from the ones computed over the fourth and the fifth approximated NDVIs.

The worst case of the Rao's Q index for the Earth data-set is reported in Fig. 9.

Example 4.21 First, in Fig. 9 the Rao's Q index over NASA NDVI of May 2014 is considered. As in the Europe case, diversity estimated by the Rao's Q index shows lower values. In the tropical regions, for example, we find the highest values, which are close to 2. Moreover there are some areas as the Sahara desert which presented extremely low diversity values.

In Fig. 10 the Rao's Q index computed over self-made and approximated NDVIs is reported. The highest errors are linked to the overestimation of diversity in the North pole area. In Fig. 10E this phenomenon decreases, but it is still appreciable. Besides, also the majority of the Pacific ocean islands are missing in all the approximations. These regions are probably related to high errors also in Example 4.15.

As before, we further compute the Rao's Q index over approximated NDVI of $\mathcal{n}_{N,h,t,j}$ for every $h \in \{E, W\}$, for every $t \in \{T, S\}$ and $j \in \{1, \ldots, 5\}$.



(A) Relative approximation

(B) Component rank 10 compression

(C) Component rank 50 compression

(D) Component rank 100 compression

(E) Component rank 500 compression

(F) Component rank 1000 compression

Fig. 10 Approximation of Rao index for May 2014, from NDVI of $\mathcal{N}_{R,W,S,j} \cup \mathcal{N}_W$

Definition 4.22 Let $\mathcal{R}_{N,h,t,j}$ be the set of Rao index computed over elements of $\mathcal{N}_{N,h,t,j}$ for every $h \in \{E, W\}$, for every $t \in \{T, S\}$ and for every $j \in \{1, \ldots, 5\}$ and we call them i_j*-approximated estimates* for every $i_j \in \mathcal{R}$ and for every $j \in \{1, \ldots, 5\}$.

Furthermore, we compute the error with respect to the original estimates, i.e.

$$\left\| A_k - C_{k,j} \right\|$$

for every $A_k \in \mathcal{R}_h$ and for every $C_{k,j} \in \mathcal{R}_{N,h,T,j} \cup \mathcal{R}_{N,h,S,j}$, for every $k \in \{1, \ldots, n_h\}$, for every $j \in \{1, \ldots, 5\}$ and for every $h \in \{E, W\}$. Moreover, we compute the error with respect to relative estimates, i.e.

$$\left\| B_k - C_{k,j} \right\|$$

for every $B_k \in \tilde{\mathcal{R}}_h$ and for every $C_{k,j} \in \mathcal{R}_{N,h,T,j} \cup \mathcal{R}_{N,h,S,j}$, for every $k \in \{1, \ldots, n_h\}$, for every $j \in \{1, \ldots, 5\}$ and for every $h \in \{E, W\}$.

Some statistics about the original and the relative errors per pixel are reported in Table 10, stored in vector epO and epR, respectively, for the Europe data-set. Comparing Tables 8 and 10, we notice that the average original and relative error per pixel are greater in the second case. In other words, starting from a tensor with repeated NIR band is not convenient for computing the Rao's Q index over Europe elements. Moreover, the minimum relative

Table 10 Statistics for Rao index over $N \in \mathcal{N}_{N,E,t,j}$

Rank	T-HOSVD					ST-HOSVD				
	10	50	100	500	1000	10	50	100	500	1000
$\mathbb{E}[epO]$	0.6392	0.3641	0.2962	0.2094	0.1969	0.6332	0.3632	0.2953	0.2111	0.1991
$\mathbb{V}\text{ar}[epO]$	0.0049	0.0012	0.001	0.003	0.0029	0.0044	0.0012	0.0009	0.0029	0.0029
$\mathbb{E}[epR]$	0.6396	0.3647	0.2974	0.2115	0.1999	0.6335	0.3638	0.2964	0.2131	0.2021
$\mathbb{V}\text{ar}[epR]$	0.0048	0.0013	0.0011	0.0036	0.0043	0.0044	0.0013	0.0011	0.0035	0.0042
min epO	0.4801	0.2741	0.2204	0.1357	0.1246	0.4777	0.2739	0.2202	0.1378	0.1269
min epR	0.4801	0.2741	0.2204	0.1357	0.1246	0.4778	0.2739	0.2202	0.1377	0.1269
max epO	0.7505	0.4167	0.3497	0.3933	0.3953	0.7403	0.4155	0.3491	0.3946	0.3949
max epR	0.7504	0.4444	0.4259	0.4487	0.5458	0.7402	0.4506	0.4193	0.4472	0.547

Table 11 Statistics for Rao index over $N \in \mathcal{N}_{N,W,t,j}$

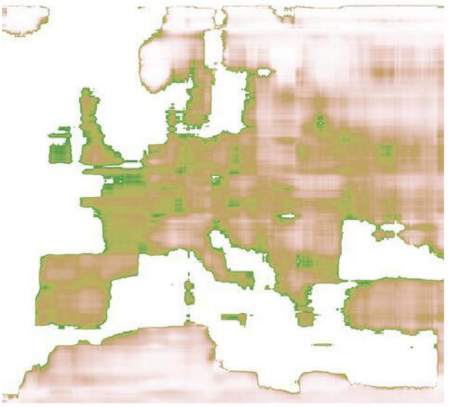
Rank	T-HOSVD					ST-HOSVD				
	10	50	100	500	1000	10	50	100	500	1000
$\mathbb{E}[epO]$	0.5366	0.3419	0.2869	0.1754	0.1636	0.5353	0.3417	0.287	0.1761	0.1651
$\mathbb{V}\text{ar}[epO]$	0.0023	0.0008	0.0007	0.0005	0.0003	0.0022	0.0008	0.0007	0.0005	0.0003
$\mathbb{E}[epR]$	0.535	0.3389	0.2835	0.1657	0.1525	0.5337	0.3387	0.2836	0.1664	0.1541
$\mathbb{V}\text{ar}[epR]$	0.0021	0.0006	0.0006	0.0003	0.0001	0.002	0.0006	0.0006	0.0003	0.0001
min epO	0.4481	0.2873	0.2385	0.1431	0.1357	0.4487	0.2881	0.2394	0.1437	0.1376
min epR	0.4486	0.2885	0.2385	0.1382	0.1331	0.4483	0.2887	0.2394	0.1391	0.135
max epO	0.6063	0.3887	0.3458	0.2352	0.226	0.6006	0.389	0.3466	0.2352	0.2267
max epR	0.6057	0.375	0.3478	0.2307	0.1746	0.5997	0.3743	0.3487	0.2314	0.1768

and original errors are found when the starting tensor has the RED band repeated. Finally, the average original error is lower than the relative one. As in the repeated RED case, ST-HOSVD is convenient when the first two components of multilinear are strictly lower than 500, otherwise T-HOSVD provides better results. The statistics for original and relative errors of the Rao's Q estimates for Earth data-set are reported in Table 11. The original and relative errors per pixel are stored in vectors epO and epR, respectively.

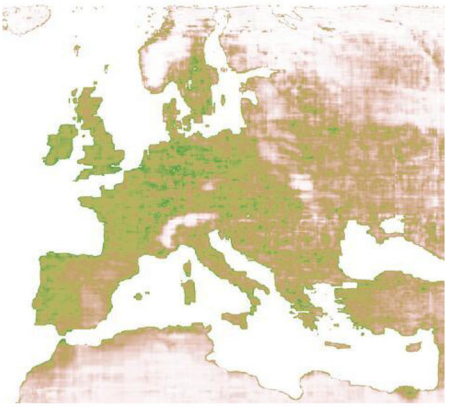
In this case comparing, Table 9 and Table 11, starting tensors with repeated NIR band provide on average better results. Indeed for both the decomposition techniques at the fifth approximation, the average original and relative errors per pixel are around 17.5% and 16.5%. This 1% difference between the RED and NIR case is present also in the minimum and maximum relative and original errors per pixel for both the decomposition techniques. Besides, also in this case ST-HOSVD is convenient when the first two multilinear components are lower than 100, otherwise T-HOSVD is preferable. Further, an average error of 16.5% per pixel is not high, recalling that we made use of only 16.4% of the total information available.

Finally, the Rao's Q index computed for December 2014 and May 2014, starting from tensors with repeated NIR bands, is reported in example 4.23.

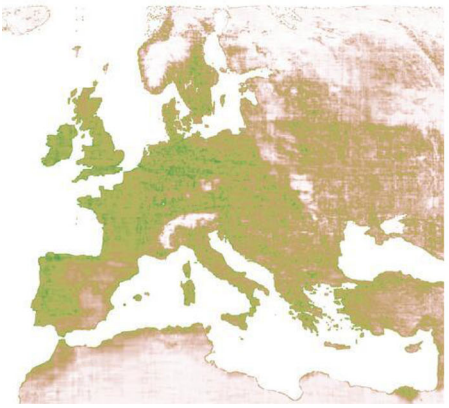
Example 4.23 December 2014 corresponds to the minimum original and relative errors, also starting from tensors with repeated NIR band. In Fig. 11 approximated Rao's Q estimates at different multilinear ranks, for this NIR repeated case, are reported. As before, it is difficult



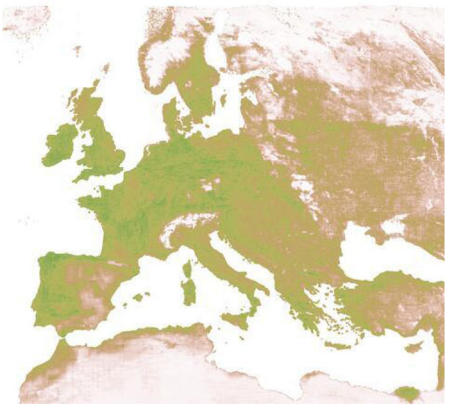
(A) Component rank 10 compression



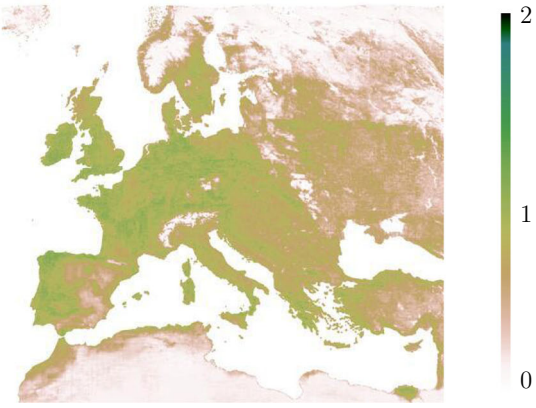
(B) Component rank 50 compression



(C) Component rank 100 compression

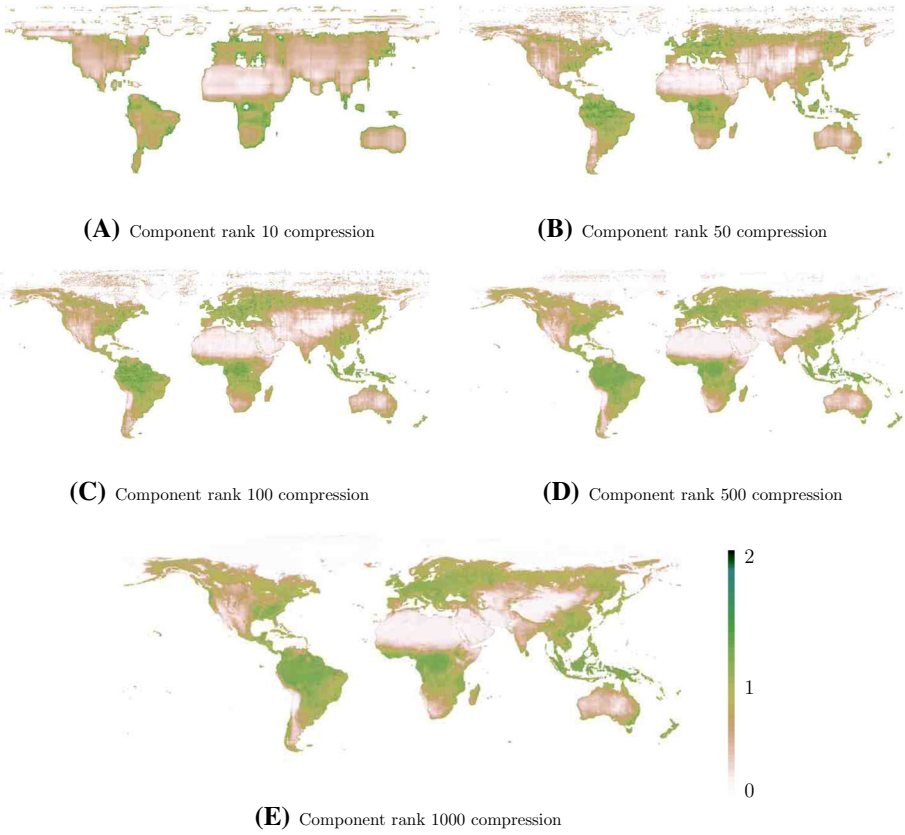


(D) Component rank 500 compression



(E) Component rank 1000 compression

Fig. 11 Approximation of Rao index for December 2014, from NDVI of $n_{N,E,S,j}$



(A) Component rank 10 compression

(B) Component rank 50 compression

(C) Component rank 100 compression

(D) Component rank 500 compression

(E) Component rank 1000 compression

Fig. 12 Approximation of the Rao's Q index for May 2014, from NDVI of $\mathcal{n}_{N,W,S,j}$

to perceive differences between the Rao's Q computed over NASA or self-made NDVI and over the last three approximated NDVIs.

Example 4.24 May 2014 does not show high values for the minimum original and relative errors starting from tensors with repeated NIR band. Therefore this is not the worst approximation in the Earth data-set. In Fig. 12 approximated Rao's Q estimates at different multilinear ranks, for this NIR repeated case, are reported. As in Fig. 10 the problematic areas are the North pole and the Pacific ocean. Indeed, there is a diversity overestimation in the Arctic regions, which decreases with increasing multilinear rank components. Moreover, even in the fifth approximation, there are missing islands in the Pacific ocean. However, no substantial differences can be found with respect to Fig. 10.

5 Conclusion

We have shown that the our approach is extremely convenient for Rényi index estimation to save storage memory. As reported in Tables 4, 6, 5 and 7 the average error per pixel is around 5.5% for Earth data-set and is close to 7.6% for the Europe's one when using respectively

16.4% and 14.8% of the total tensor information. Moreover, starting from tensor with repeated NIR bands is more convenient than with RED repeated, in the Rényi case. For the Rao's Q case starting with twice NIR rasters is more convenient only for the Earth data-set. Finally, in the Rényi case, T-HOSVD made on diversity estimation shows lower errors than ST-HOSVD when the first two multilinear rank components are relatively low.

In the Rao's Q case we find higher original and relative error per pixel, on average. Indeed in Tables 8 and 9 we notice an average error per pixel close to 19.5% and 17% for Europe and for Earth data-set respectively.

In conclusion we believe that the presented work might help ecologists in their remote sensing diversity estimation, when dealing with big data. Indeed, fixed a certain accuracy, they can compress through T-HOSVD and ST-HOSVD the tensors with NIR and RED bands, to save storage memory and at the same time computing with fixed tolerance the diversity estimates.

Lastly, if it was possible to directly download decomposed tensors, ecologists would benefit of a significant computer memory saving.

Funding Open access funding provided by Università degli Studi di Trento within the CRUI-CARE Agreement.

6 Appendix

6.1 Rao's and Rényi's codes

Remind that for every raster of size (m, n), fixed l the side of the moving window, the diversity index is computed $l \times m \times n$ times. To speed up the entire work we decide to implement a parallel version of the diversity index algorithm. When a computer executes a parallel function, its cores perform independently different tasks at the same time. In our case we want each core of the used machine to work on a different position of the moving window. Consequently, to implement this mechanism, we need a parallel computing library compatible with `Python`, the chosen programming language. We prefer `Joblib`, cf. [43], because of its user-friendliness. Other two used libraries are `itertools`, to create iterable elements and `spatial` from `SciPy`, to compute the distance element-wise for two matrices. The main Rao computation code is

```
#### computation
import numpy as np
import scipy
from scipy import spatial
import itertools
#### parallelisation
import joblib
```

```
from joblib import Parallel, delayed
import multiprocessing

def raop(rw):
def raout (cl, rw = rw, rasterm = rasterm, missing = missing
    ↪ , w = w, distance_m = distance_m):
tw_labels, tw_values = np.unique(rasterm[(rw-w):(rw+w+1),(cl
    ↪ -w):(cl+w+1)], return_counts=True)
if len(np.where(tw_labels == missing)) != 0:
tw_values = np.delete( tw_values, np.where(tw_labels ==
    ↪ missing))
tw_labels = np.delete( tw_labels, np.where(tw_labels ==
    ↪ missing))
if len(tw_values) > 1:
d1 = spatial.distance.cdist(np.diag(tw_labels), np.diag(
    ↪ tw_labels), distance_m)
p = tw_values/np.sum(tw_values)
p1 = np.zeros((len(tw_values),len(tw_values)))
comb = np.array([x[0]*x[1] for x in list(itertools.
    ↪ combinations(p, 2))])
p1[np.triu_indices(len(tw_values), k=1)] = comb
p1[np.tril_indices(len(tw_values), k=-1)] = comb
return ((np.sum(np.multiply(p1,d1))))
elif len(tw_values) == 1:
return (((0)))
else:
return ((missing))
Raout = Parallel(n_jobs = NcCores)(delayed(raout)(cl) for cl
    ↪  in range(w,c-w))
return (Raout)
out[w:(r-w), w:(c-w)] = (np.asarray(Parallel(n_jobs =
    ↪ NcCores)(delayed(raop)(rw) for rw in range(w,(r-w)))).
    ↪ reshape(r-2*w,c-2*w))
```

Since the moving window scrolls over rows and over columns, we define two parallelised functions one inside the other, `raop` whose variable is just the row index and `raout`. This second function takes as variable the column index and has other set parameters: `rasterm` the raster, `rw` the row index, `missing` a value used in the raster when data are not present, `w` the window side and a function `distance_m` to compute the distance between raster values. The first step is storing the values and their frequencies of the raster area covered by the moving window in the arrays `tw_labels` and `tw_values` respectively. Then checking if there is the `missing` values in the considered area: if it is present, we remove it and its frequency from the storing arrays. Next if there are at least two different elements in the `tw_values` array, we compute their distance with the function `spatial.distance.cdist` and save the result in the `d1` matrix. In `p` we put the relative frequencies, obtained from the absolute ones. We define a 0 matrix `p1`, with the same size of `d1`. We compute and store in `comb` all the possible combinations between different elements of vector `p`. We assign these values to the upper and lower triangular parts of the matrix `p1`. Finally, we return the sum of the elements of the product matrix between `d1` and `p1`. Notice that the product matrix will

coherently have diagonal null elements, since the distance between a pixel value and itself is 0.

If there is just one element in vector `tw_values`, we return 0 since the difference between a pixel value and itself is 0. Otherwise, if `tw_values` is an empty vector, we return `missing`, to highlight this particular situation. Inside the function `raop` we define function `raout` and we do the first `Parallel` call. Outside it, we do the second `Parallel call`. Notice that this function is thought to compute Rao's index when the moving window is completely contained into the raster. When this condition is not satisfied, to speed up computation, there are special implementations of Rao's Q, available here [21]. With a similar approach, we implement Rényi's index, whose core is

```
#### computation
import numpy as np
import scipy
#### parallelisation
import joblib
from joblib import Parallel, delayed
import multiprocessing

def IndexOP (rw):
def IndexOut (cl, rw = rw, rasterm = rasterm, missing =
    ↪ missing, w = w, alpha=alpha, base = base):
tw_labels, tw_values = np.unique(rasterm[(rw-w):(rw+w+1),(cl
    ↪ -w):(cl+w+1)], return_counts=True)
if len(np.where(tw_labels == missing)[0]) != 0:
tw_values = np.delete( tw_values, np.where(tw_labels ==
    ↪ missing))
if len(tw_values) != 0:
p = tw_values/np.sum(tw_values)
if np.log(np.sum(p**alpha)) == 0:
return(0)
else:
return((1/(1-alpha)) * np.log(np.sum(p**alpha)) / np.log(
    ↪ base))
else:
return (missing)
Index_Out = Parallel(n_jobs = NcCores)(delayed(IndexOut)(cl)
    ↪  for cl in range(w,c-w))
return (Index_Out)
out[w:(r-w), w:(c-w)] = np.asarray(Parallel(n_jobs = NcCores
    ↪ )(delayed(IndexOP)(rw) for rw in range(w,r-w))).
    ↪ reshape(r-2*w,c-2*w)
```

Similarly we declar the function `IndexOP`, with just row index as variable and, inside it, the function `IndexOut`. The variable proper of `IndexOut` are `alpha`, the Rényi's parameter, and `base`, the logarithm base. The code structure is similar to that of `raout`. We compute and store the values, presented inside the raster area covered by the moving window, and their frequencies. We check and in case delete every `missing` value and its frequency. If there is still an element into `tw_values`, we compute the relative frequencies and returned the Rényi index. We tested if the index final value is 0 and return 0 to avoid sign problems.

If `tw_values` is an empty vector, we return a `missing` value. We do a first `Parallel` call of `IndexOut` inside `IndexOP` and a second one of `IndexOP` outside it. Even in this case, the presented code will work only when the moving window is entirely contained in the raster. To speed up the computations, we have developed a special version of this code which works when the moving window is not fully contained. It is available here [21].These codes have been used as prototype for the some function availabe in the `rasterdiv R` library fully described in [35]

6.2 Approximation codes for T-HOSVD

In order to implement the approximation Algorithm 1 of T-HOSVD we need a python library expressly developed to manage tensors. Moreover we also want it to interact properly with `NumPy`, i.e. the mostly used python library for numerical computation, cf. [30]. Therefore, we choose `TensorLy`, cf. [25]. This recently developed library, firstly presented in 2016, was thought to make tensor study and manipulation easy and accessible. Besides, their creators projected `TensorLy` to perfectly match with other famous python libraries, as `NumPy`, `SciPy`. They developed most of the main tensor operations and related functions. To compute the singular value decomposition of tensors flattening, the central step in T-HOSVD and ST-HOSVD algorithm, we need a python function able to manage huge arrays. This is not a trivial task, since the development of python functions is left to singular initiative. We use `svds` function from `SciPy` sparse linear algebra function, cf. [24]. This implementation of SVD takes advantage of matrix sparsity in performing the matrix-vector multiplication. The final implementation of T-HOSVD is

```
import numpy as np
import tensorly as tl
from tensorly import decomposition as decompose
from tensorly import tenalg as Tla
import scipy
from scipy import spatial
from scipy.sparse.linalg import svds

def T_hosvd(T, rank, projector = Ture):
L = []
dim = T.shape
for i in range(len(dim)):
flat = tl.unfold(T,i)
res = svds(flat, k = rank[i])
L.append(np.transpose(res[0][0:dim[i], :]))
core = Tla.multi_mode_dot(T,L)
if projector:
P = [np.transpose(u) for u in L]
return [core,P]
return core
```

This function takes as input variable a tensor `T` and a list or a tuple, whose values are the target multilinear rank components. Inside the `T_hosvd` we declare an empty list, `L`, in which at the i-th step we store matrix `U_i` from the thin SVD of the i-th flattening for every $i \in \{1, \ldots, d\}$. The `dim` variable is a tuple containing the size of tensor `T`. Then for each

direction we compute the flattening and its SVD. After the for loop, we get the core tensor with the multilinear product between a list of matrices, our projectors, and the original tensor `T`. Notice that there is the option `projector`, with `True` as default value, to return a list `L` containing the projectors matrices together with the standard result, i.e. the core tensor.

6.3 Approximation codes for ST-HOSVD

The ST-HOSVD implementation of Algorithm 2 is

```
import numpy as np
import tensorly as tl
from tensorly import decomposition as decompose
from tensorly import tenalg as Tla
import scipy
from scipy import spatial
from scipy.sparse.linalg import svds

def ST_hosvd(T, rank, projector = True):
dim = T.shape
core = T
if projector:
P=[]
for i in range(len(dim)):
flat = tl.unfold(core,i)
res = svds(flat, k=rank[i])
core = Tla.mode_dot(core,np.transpose(res[0][0:dim[i],:]),
    ↪ mode=i)
if projector:
P.append(res[0][0:dim[i],:])
if projector:
return [core,P]
return core
```

The input arguments of ST-HOSVD and T-HOSVD implementations are the same. The first difference with `T_hosvd` is the initial declaration of `core` tensor, set equal to `T`. Only if the `projector` variable is `True`, we initialize an empty list to store the projectors matrices. Another difference is the computation of the core tensor inside the for loop with the component wise product between matrix and tensor, given a certain direction. The basic output is still the final core tensor.

References

1. Acar, E., Dunlavy, D.M., Kolda, T.G., Mørup, M.: Scalable tensor factorizations for incomplete data. Chemom. Intell. Lab. Syst. **106**(1), 41–56 (2011). Multiway and Multiset Data Analysis
2. Ballico, E., Bernardi, A., Carusotto, I., Mazzucchi, S., Moretti, V.: Introduction. Lecture Notes of the Unione Matematica Italiana, pp. 1–4 (2019)
3. Benzi, M., Bini, D., Kressner, D., Munthe-Kaas, H., Van Loan, C.: Exploiting hidden structure in matrix computations: algorithms and applications. Lecture Notes in Mathematics (2016)

4. Bernardi, A., Brachat, J., Comon, P., Mourrain, B.: Multihomogeneous polynomial decomposition using moment matrices. In: ISSAC 2011—Proceedings of the 36th International Symposium on Symbolic and Algebraic Computation, pp. 35–42. ACM, New York (2011)
5. Bernardi, A., Brachat, J., Comon, P., Mourrain, B.: General tensor decomposition, moment matrices and applications. J. Symb. Comput. **52**, 51–71 (2013). cited By 28
6. Bernardi, A., Daleo, N.S., Hauenstein, J.D., Mourrain, B.: Tensor decomposition and homotopy continuation. Differ. Geom. Appl. **55**, 78–105 (2017). Geometry and complexity theory
7. Bini, D., Pan, V.Y.: Fundamental computations with general and dense structured matrices. Polynomial and Matrix Computations, pp. 81–227 (1994)
8. Bjerhammar, A.: Application of Calculus of Matrices to Method of Least Squares; with Special References to Geodetic Calculations. Trans. Roy. Inst. Tech, Stockholm (1951)
9. Boralevi, A., Draisma, J., Horobeţ, E., Robeva, E.: Orthogonal and unitary tensor decomposition from an algebraic perspective. Isr. J. Math. **222**(1), 223–260 (2017)
10. Campbell, J.B., Wynne, R.H.: Introduction to Remote Sensing, vol. 1, 5th edn. Guilford Publications, New York (2011)
11. Chiarucci, A.: To sample or not to sample? That is the question... for the vegetation scientist. Folia Geobot. **42**(2), 209 (2007)
12. Chuvieco, E., Justice, C.: Nasa earth observation satellite missions for global change research. Earth Observation of Global Change, pp. 23–47
13. De Lathauwer, L., De Moor, B., Vandewalle, J.: A multilinear singular value decomposition. SIAM J. Matrix Anal. Appl. **21**(4), 1253–1278 (2000)
14. de Silva, V., Lim, L.-H.: Tensor rank and the ill-posedness of the best low-rank approximation problem. SIAM J. Matrix Anal. Appl. **30**, 1084–1127 (2008)
15. Didan, K.: Mod13c2 modis/terra vegetation indices monthly l3 global 0.05deg cmg v006. 2015. https://doi.org/10.5067/MODIS/MOD13C2.006 (2010–2018)
16. Didan, K.: MOD13A3 MODIS/Terra vegetation Indices Monthly L3 Global 1km SIN Grid V006. 2015. https://doi.org/10.5067/MODIS/MOD13A3.006 (2011–2018)
17. Draisma, J., Ottaviani, G., Tocino, A.: Best rank-k approximations for tensors: generalizing Eckart–Young. Res. Math. Sci. **5**, 27 (2018)
18. Eckart, C., Young, G.: The approximation of one matrix by another of lower rank. Psychometrika **1**, 211–218 (1936)
19. Gerster, M., Silvi, P., Rizzi, M., Fazio, R., Calarco, T., Montangero, S.: Unconstrained tree tensor network: an adaptive gauge picture for enhanced performance. Phys. Rev. B **90**(12), 125154 (2014)
20. Golub, G.H., Van Loan, C.F.: Matrix Computations, 3rd edn. The Johns Hopkins University Press, Baltimore (1996)
21. Iannacito, M.: Biodiversity index computation (2019). https://zenodo.org/badge/latestdoi/196228972, https://github.com/MartinaIannacito/Biodiversity-index-computation
22. Ishteva, M., Absil, P.-A., Van Huffel, S., De Lathauwer, L.: Best low multilinear rank approximation of higher-order tensors, based on the Riemannian trust-region scheme. Soc. Ind. Appl. Math. **32**(1), 115–135 (2011)
23. Jia, Y.-J., Xu, P.-F., Pei, X.-M.: An investigation of image compression using block singular value decomposition. Communications and Information Processing, pp. 723–731 (2012)
24. Jones, E., Oliphant, T., Peterson, P.: Scipy: open source scientific tools for python (2001). http://www.scipy.org/
25. Kossaifi, J., Panagakis, Y., Anandkumar, A., Pantic, M.: Tensorly: tensor learning in python. J. Mach. Learn. Res. **20**(26), 1–6 (2019)
26. Lim, L.-H., Comon, P.: Multiarray signal processing: tensor decomposition meets compressed sensing. Comptes Rendus Mécanique **338**(6), 311–320 (2010)
27. Mirsky, L.: On the trace of matrix products. Math. Nachr. **20**, 171–174 (1959)
28. Mirsky, L.: Symmetric gauge functions and unitarily invariant norms. Q. J. Math. Oxf. II. Ser. **11**, 50–59 (1960)
29. Moore, E.H.: On the reciprocal of the general algebraic matrix. Bull. Am. Math. Soc. **26**, 394–395 (1920)
30. Oliphant, T.E.: A guide to NumPy, vol. 1. Trelgol Publishing USA (2006)
31. Penrose, R.: A generalized inverse for matrices. Proc. Camb. Philos. Soc. **51**, 406–413 (1955)
32. Rocchini, D., Balkenhol, N., Carter, G.A., Foody, G.M., Gillespie, T.W., He, K.S., Kark, S., Levin, N., Lucas, K., Luoto, M., Nagendra, H., Oldeland, J., Ricotta, C., Southworth, J., Neteler, M.: Remotely sensed spectral heterogeneity as a proxy of species diversity: recent advances and open challenges. Ecol. Inform. **5**(5), 318–329 (2010)

33. Rocchini, D., Delucchi, L., Bacaro, G., Cavallini, P., Feilhauer, H., Foody, G.M., He, K.S., Nagendra, H., Porta, C., Ricotta, C., Schmidtlein, S., Spano, L.D., Wegmann, M., Neteler, M.: Calculating landscape diversity with information-theory based indices: a GRASS GIS solution. Ecol. Inform. **17**, 82–93 (2013)
34. Rocchini, D., Marcantonio, M., Ricotta, C.: Measuring Rao's q diversity index from remote sensing: an open source solution. Ecol. Indic. **72**, 234–238 (2017)
35. Rocchini, D., Thouverai, E., Marcantonio, M., Iannacito, M., Da. Re, D., Torresani, M., Bacaro, G., Bazzichetto, M., Bernardi, A., Foody, G.M., Furrer, R., Kleijn, D., Larsen, S., Lenoir, J., Malavasi, M., Marchetto, E., Messori, F., Montaghi, A., Moudrý, V., Naimi, B., Ricotta, C., Rossini, M., Santi, F., Santos, M.J., Schaepman, M.E., Schneider, F.D., Schuh, L., Silvestri, S., Símová, P., Skidmore, A.K., Tattoni, C., Tordoni, E., Vicario, S., Zannini, P., Wegmann, M.: Rasterdiv—an information theory tailored r package for measuring ecosystem heterogeneity from space: to the origin and back. Methods in Ecology and Evolution, pp. 1–10 (2021)
36. Rouse Jr., J.W., Haas, R.H., Schell, J.A., Deering, D.W.: Monitoring vegetation systems in the great plains with ERTS. NASA Spec. Publ. **351**, 309 (1974)
37. Schmidt, E.: Zur theorie der linearen und nichtlinearen integralgleichungen. Math. Ann. **65**, 433–476 (1907)
38. Schön, C., Solano, E., Verstraete, F., Cirac, J.I., Wolf, M.M.: Sequential generation of entangled multiqubit states. Phys. Rev. Lett. **95**, 110503 (2005)
39. Silvi, P., Tschirsich, F., Gerster, M., Jünemann, J., Jaschke, D., Rizzi, M., Montangero, S.: The tensor networks anthology: simulation techniques for many-body quantum lattice systems. SciPost Phys. Lect. Notes, vol. 8 (2019)
40. Stewart, G.W.: On the early history of the singular value decomposition. SIAM Rev. **35**, 551–566 (1993)
41. Vannieuwenhoven, N., Nicaise, J., Vandebril, R., Meerbergen, K.: On generic nonexistence of the Schmidt–Eckart–Young decomposition for complex tensors. SIAM J. Matrix Anal. Appl. **35**, 886–903 (2014)
42. Vannieuwenhoven, N., Vandebril, R., Meerbergen, K.: A new truncation strategy for the higher-order singular value decomposition. SIAM J. Sci. Comput. **34**, 1027–1052 (2012)
43. Varoquaux, G.: Joblib: running python functions as pipeline jobs (2009). https://github.com/joblib/joblib
44. Wang, L., Bai, J., Wu, J., Jeon, G.: Hyperspectral image compression based on lapped transform and Tucker decomposition. Signal Process. Image Commun. **36**, 63–69 (2015)
45. Zellweger, F., Frenne, P.D., Lenoir, J., Rocchini, D., Coomes, D.: Advances in microclimate ecology arising from remote sensing. Trends Ecol. Evol. **34**(4), 327–341 (2019)