

# GET: Global Envelopes in R

Mari Myllymäki

Natural Resources Institute Finland (Luke)

Tomáš Mrkvička

University of South Bohemia

---

## Abstract

This work describes the R package **GET** that implements global envelopes for a general set of  $d$ -dimensional vectors  $T$  in various applications. A  $100(1 - \alpha)\%$  global envelope is a band bounded by two vectors such that the probability that  $T$  falls outside this envelope in any of the  $d$  points is equal to  $\alpha$ . Global means that the probability is controlled simultaneously for all the  $d$  elements of the vectors. The global envelopes can be employed for central regions of functional or multivariate data, for graphical Monte Carlo and permutation tests where the test statistic is multivariate or functional, and for global confidence and prediction bands. Intrinsic graphical interpretation property is introduced for global envelopes. The global envelopes included in the **GET** package have the property, which particularly helps to interpret the test results, by having a graphical interpretation that shows the reasons of rejection of the tested hypothesis. Examples of different uses of global envelopes and their implementation in the **GET** package are presented, including global envelopes for single and several one- or two-dimensional functions, Monte Carlo goodness-of-fit tests for simple and composite hypotheses, comparison of distributions, functional analysis of variance, and functional linear model, and confidence bands in polynomial regression.

*Keywords:* functional linear model, central region, goodness-of-fit, graphical normality test, Monte Carlo test, multiple testing, permutation test, R, spatial point pattern.

---

## Preface

This vignette corresponds to Myllymäki and Mrkvička (2020, **GET**: Global envelopes in R. arXiv:1911.06583 [stat.ME]) available at <https://arxiv.org/abs/1911.06583>. When citing the vignette and package please cite Myllymäki and Mrkvička (2020) and references given by typing `citation("GET")` in R.

## 1. Introduction

Global envelopes are useful for formal testing of various hypotheses using functional or multivariate statistics when interpretation of the test results is of key interest, for determining central regions of functional or multivariate data, and also for determining confidence or prediction bands (e.g., Myllymäki, Mrkvička, Grabarnik, Seijo, and Hahn 2017; Mrkvička, Myllymäki, and Hahn 2017; Narisetty and Nair 2016). Global envelopes have shown their usefulness already in many areas, e.g., spatial statistics, functional data analysis and image analysis, with applications to agriculture, architecture and art, astronomy and astro-

physics, ecology, evolution, economics, eye movement research, fisheries, forestry, geography, material science, and medicine, health and neurosciences (e.g., Stoyan 2016; Murrell 2018; Häbel, Rajala, Marucci, Boissier, Schladitz, Redenbach, and Särkkä 2017; Chaiban, Biscio, Thanapongtharm, Tildesley, Xiao, Robinson, Vanwambeke, and Gilbert 2019; Pollington, Tildesley, Hollingsworth, and Chapman 2020). To make these methods easily accessible, the R (R Core Team 2020) package **GET** has been developed that is available from the Comprehensive R Archive Network (CRAN) at <https://cran.r-project.org/package=GET>. A development version of the package is available via the repository <https://github.com/myllym/GET>. The package provides an implementation of global envelopes in various settings.

Because there are many other methods that can be used for the same purposes as global envelopes, we first give a motivating example of functional analysis of covariance (ANCOVA) to demonstrate the main features and advantages of global envelope tests (Section 1.1). Other possible usages are also discussed in more detail. The second part of this introductory section describes the competing and complementary methods and software for these same usages. Thereafter, in Section 2, we give an overview of global envelopes including the formal definition of their graphical interpretation, summary of the types of global envelopes and their implementation in **GET**. In Section 3, the usage of global envelopes is shown for several examples of applications illustrating main features of the **GET** package, namely 1) the computation of central regions and functional boxplots for a set of functions or jointly for several sets of functions (Section 3.1); 2) the Monte Carlo goodness-of-fit test for simple hypotheses with application to spatial statistics (Section 3.2); 3) the Monte Carlo goodness-of-fit test for composite hypotheses with application to graphical normality testing (Section 3.3); 4) the graphical  $n$ -sample test of correspondence of distribution functions,  $n \geq 2$  (Section 3.4); 5) the graphical functional one-way ANOVA (Section 3.5); 6) the functional GLM for images (Section 3.6); and 7) the computation of the confidence band in polynomial regression (Section 3.7). The final section, Section 4, is left for discussion.

### 1.1. A motivating example and possible usages

Let us study the differences between population growth accumulated in different continents over the period of 55 years. We assume that GDP (gross domestic product) can be the main driver of population growth in different countries, therefore we add GDP into the model as a nuisance factor. Figure 1 (left) shows population growth, defined here as the population at the end of the year divided by the population at the beginning of the year, in years from 1960 to 2014 in Africa, Asia, Europe and North America, and Latin America, total in 112 countries with more than one million inhabitants in 1950 (Nagy, Gijbels, and Hlubinka 2017; Dai, Mrkvička, Sun, and Genton 2020). Figure 1 (right) shows the GDP of every country in the study discounted to the 1960 USD. The discounting was performed according to USD inflation. The data was obtained from the World Bank. The missing values of the GDP of a country were extrapolated using the closest known ratio of the GDP of the country and the median GDP in that year. The missing values of GDP were interpolated using linear interpolation of the two closest ratios.

Since the global envelopes are a nonparametric tool, we can use as a test statistic any multivariate or functional test statistic without the worry of breaking assumptions of normality or homogeneity of the distribution of the test statistic in different years. Also, since we can

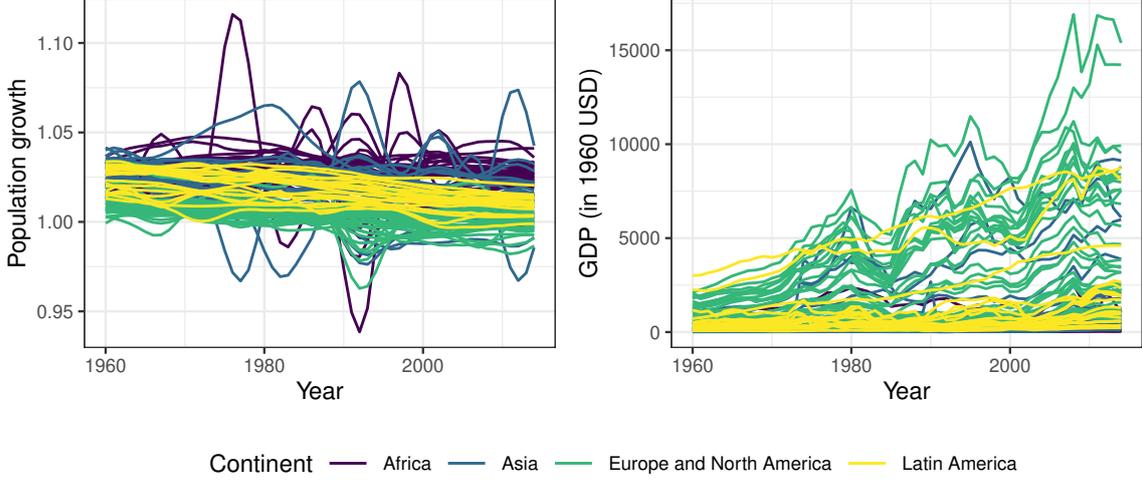


Figure 1: Population growth and GDP (in 1960 USD) curves for 1960–2014 for 112 countries from the four continents.

use any test statistic, we can combine several functional test statistic into one (for more details about combining test statistics see Appendix B). Here, we combine four functional test statistics, each representing a coefficient in our ANCOVA model attached to a continent, i.e., our test statistic is

$$\boldsymbol{\beta}^{\text{Cont}} = \left( \underbrace{\beta_{1,1960}^{\text{Cont}}, \dots, \beta_{1,2014}^{\text{Cont}}}_{\text{1: Africa}}, \underbrace{\beta_{2,1960}^{\text{Cont}}, \dots, \beta_{2,2014}^{\text{Cont}}}_{\text{2: Asia}}, \underbrace{\beta_{3,1960}^{\text{Cont}}, \dots, \beta_{3,2014}^{\text{Cont}}}_{\text{3: Europe and North America}}, \underbrace{\beta_{4,1960}^{\text{Cont}}, \dots, \beta_{4,2014}^{\text{Cont}}}_{\text{4: Latin America}} \right), \quad (1)$$

where  $\beta_{1,i}^{\text{Cont}}, \dots, \beta_{4,i}^{\text{Cont}}$  are the parameters of the univariate ANCOVA model for year  $i$ , computed with condition  $\sum_{j=1}^4 \beta_{j,i}^{\text{Cont}} = 0$ . This model parametrization enables that the parameter  $\beta_{j,i}^{\text{Cont}}$  is the difference between the  $j$ -th continent's growth rate and the overall mean.

This test statistic allows for a direct comparison of each continent's growth rate with the world's mean growth rate. Figure 2 shows the test statistic (Equation 1) computed for the data together with 95% global envelope computed under the null model of no continent effect but with nuisance effect of GDP. This output shows, first, the formal global  $p$  value, which indicates that the effect of the continent is significant after applying an adjustment for multiple comparisons for the 55 years, and four continents. Second, it shows the reasons of rejections, i.e., which years had led to the significant test result. Due to the intrinsic graphical interpretation (IGI, see Definition 2.1) of the global envelope, we have a one-to-one correspondence between the formal global test and its graphical interpretation, i.e., the data function lies outside the envelope for some point (here years and continents) if and only if the global test is significant. This allows for direct identification of the significant years. Third, the output shows the direction and amount of deviation from the null hypothesis. For example, the growth rate in Europe and North America is significantly lower than the world's

average in all years, whereas Asia's growth rate is slightly significantly higher than the world's average only up to the year 1966. Fourth, due to the free choice of the test statistic, the test output shows here the comparison individually for every continent (level of the categorical predictor). The **GET** package allows also for the choice of comparison between all pairs of continents in the style of the post hoc test. Note that usually, the two tests are applied - global AN(C)OVA and multiple comparison post hoc test. Note also that we can calculate results also for continuous GDP effect and interaction effect between the GDP and all continents (all levels of the categorical effect) in the same way.

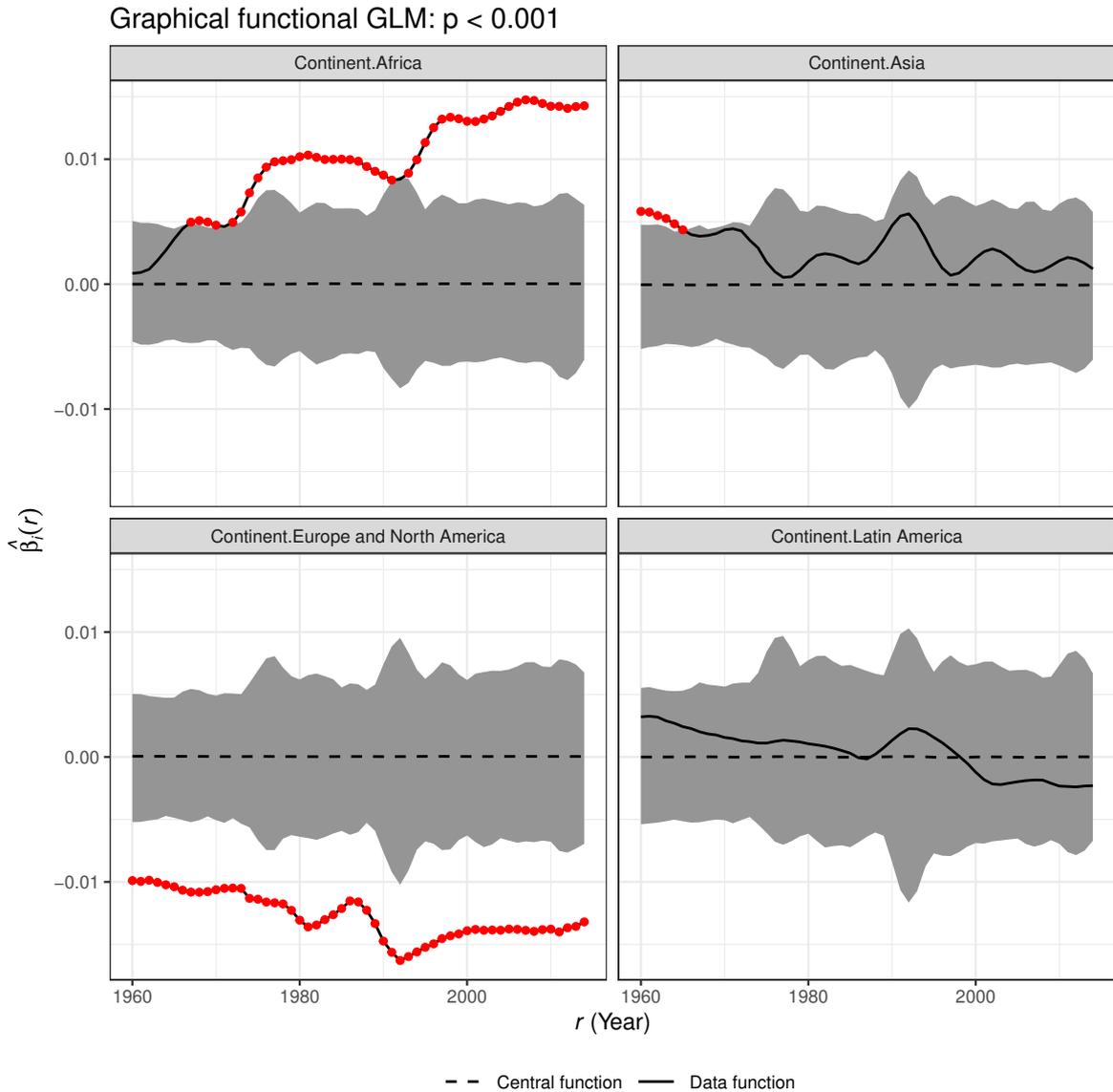


Figure 2: Test for the main effect of the continent, given the GDP (in 1960 USD) as a nuisance factor. The red color is used to highlight the (significant) years where the data coefficient  $\hat{\beta}_{j,i}^{\text{Cont}}$ ,  $i = 1960, \dots, 2014$ ,  $j = 1, 2, 3, 4$ , goes outside the 95% global envelope computed under the null model of no continent effect. Here  $j$  refers to the continent given in the titles.

Technically, the global envelope of Figure 2 was computed from 5000 permutations under the null model using the standard [Freedman and Lane \(1983\)](#) permutation procedure. The IGI measure used was the area measure (see [Appendix A](#) for details).

The above example illustrates the global envelope test, a possible usage of global envelopes. The usage of global envelopes is however more versatile. They can be used for producing

- (i) a central region: a central region is constructed for a set of vectors or functions in order to find central or outlying vectors or functions (e.g., outlier detection, functional boxplot);
- (ii) a global envelope test: a Monte Carlo goodness-of-fit test or a permutation test where the test statistic is multivariate or a function of any dimension (e.g., goodness-of-fit test for point patterns or random sets, for a family of distributions, functional analysis of variance (ANOVA), functional general linear model (GLM),  $n$ -sample test of correspondence of distribution functions);
- (iii) global confidence or prediction bands: a confidence or prediction band is produced from a set of vectors or functions obtained by bootstrap or sampling from Bayesian posterior distribution (e.g., confidence band in polynomial regression, Bayesian posterior prediction).

In fact, the global envelope test uses the 95% central region (case (i) usage) computed under the null hypothesis for determining the test results. The global confidence bands (case (iii) usage) also uses the 95% central region computed under the full model. Thus in core of all applications lies the central region. Central regions can be constructed based on any functional ordering, but we concentrate only in those satisfying the IGI, the one-to-one correspondence of a functional ordering and its graphical interpretation. In fact, when central region is not satisfying IGI and we see a function which is not contained completely inside the central region, we still do not know if such a function can be considered as extremal. On the contrary, when we use IGI ordering, we know that such a function is among the most, say 5%, extremal functions.

## 1.2. Competing and complementary methods and software

Below are listed the other R packages (or code) that we know to provide functions for some global envelopes or central regions. Further, as already mentioned, the problems (i)-(iii) can be solved by other methods as well, not just by global envelopes. The relation of these methods to the global envelope methods are also discussed below.

### Global envelopes and central regions:

- The R package **fda** ([Ramsay, Graves, and Hooker 2022](#)) provides the function `fbplot()` for the computation of the central region and functional boxplot according to two different orderings than those described here, namely the band depth and modified band depth (MBD) ([López-Pintado and Romo 2009](#); [Sun, Genton, and Nychka 2012](#)), but these depths do not allow for IGI.
- The R package **depthTools** ([López-Pintado and Torrente 2013](#); [Torrente, López-Pintado, and Romo 2013](#)) similarly allows for central regions based on MBD (no IGI).

- The R package **spatstat** (Baddeley, Rubak, and Turner 2015) provides the function `envelope()` for the simulation of envelopes based on a given summary function of a spatial point pattern. By default, `envelope()` provides a pointwise envelope, but the option `global = TRUE` allows one to compute the global envelope of Ripley (1981), which corresponds to the 'unscaled' envelope in GET (see Table 1). It has been shown that this *unscaled* global envelope test has generally lower power than the other methods of Table 1 (Myllymäki, Grabarnik, Seijo, and Stoyan 2015; Myllymäki *et al.* 2017). The corresponding adjusted unscaled global envelope (Dao and Genton 2014; Baddeley, Hardegen, Lawrence, Milne, Nair, and Rakshit 2017) for composite hypotheses is also provided in **spatstat** (the function `dg.envelope()`).
- Aldor-Noiman, Brown, Buja, Rolke, and Stine (2013) presented a global envelope for a Q-Q plot (and provided a link to an R script). The shape of the envelope is derived theoretically, but the size of the envelope has to be computed from simulations. The methods of the GET package can be used for this purpose as well, both for simple and composite hypotheses, but the theoretical achievement of Aldor-Noiman *et al.* (2013) for simple hypotheses has apparent advantages.
- The R package **boot** (Canty and Ripley 2017; Davison and Hinkley 1997) provides the function `envelope()` for the computation of a global envelope from bootstrapped functions. This envelope has the same shape as the global rank envelope ('rank' in Table 1), but the appropriate envelope ( $l$  of Equation 8) is chosen in **boot** experimentally (Davison and Hinkley 1997). Since the differences in the nominal levels of the subsequent ( $l$ -)envelopes from which the choice is made can be large, the predetermined level is reached only approximately.
- The package **dbmss** (Marcon, Traissac, Puech, and Lang 2015) provides similar global envelopes as the **boot** package (Duranton and Overman 2005) but for the global confidence envelopes of spatial summaries.
- There are other R packages with the ability to compute simultaneous confidence bands for various models, e.g., **excursions** (Bolin and Lindgren 2015, 2017, 2018) for Gaussian processes, **AdaptFitOS** (Wiesenfarth, Krivobokova, Klasen, and Sperlich 2012) for semiparametric regression models and **SCBmeanfd** (Degras 2016) for nonparametric regression models with functional data using a functional asymptotic normality result. Instead, the global envelopes of the GET package (see Table 1) are constructed non-parametrically from a set of vectors. Apparently, the package **excursions** allows also for a non-parametric version with the same shape as the global rank envelope ('rank' in Table 1) similarly as **boot**.

**Multiple testing:** The global envelope tests can be seen as a general solution to the multiple testing problem in Monte Carlo tests (Mrkvička *et al.* 2017). There are several other methods and R packages to the multiple testing problem controlling the family-wise error rate (FWER). The few packages mentioned below have a link to the methods of GET:

- The R packages **coin** (Hothorn, Hornik, van de Wiel, and Zeileis 2008, 2006) and **multtest** (Pollard, Dudoit, and van der Laan 2005) enable one to compute the  $p$  value adjusted for multiple testing in a multiple permutation test based on the minimum  $p$  value computed from all individual tests. The null distribution of the minimum  $p$  values or

the maximum of a test statistic is obtained from permutations. The minimum  $p$  value method corresponds to the conservative rank test based on the  $p_+$  value (see global rank envelope in Appendix A).

- General multiple test procedures are also provided by the package **sgof** (Conde and de Una Alvarez 2016) for goodness-of-fit testing and by the package **stats** (R Core Team 2020) for adjusting the  $p$  values for multiple comparisons by Bonferroni type methods (the function `p.adjust()`).

**Functional GLM:** The global envelope tests can also be used for functional GLM using a permutation strategy to generate samples under the null hypothesis. There are several other methods and software to the functional GLM problem:

- The PALM software (Winkler, Ridgway, Webster, Smith, and Nichols 2014) allows for the computation of various functional GLM designs using permutation tests. The multiple testing problem is solved by an unscaled envelope constructed for the test statistic (e.g.,  $F$  statistic).
- The R packages **fda.usc** (Febrero-Bande and Oviedo de la Fuente 2012) and **fdANOVA** (Gorecki and Smaga 2017) allow for the computation of functional ANOVA designs by several methods together with the computation of factor's significances. Similarly the package **fda** (Ramsay *et al.* 2022) allows for computations in functional regression designs. However, to the best of our knowledge, these do not provide the IGI of tests of a factor significance.

## 2. Overview of global envelopes in GET

### 2.1. Global envelope and intrinsic graphical interpretation (IGI)

Global envelopes are constructed for general multivariate statistics, so in the case when the data are purely functional, they first have to be discretized. The discretization of the functions can be arbitrary, as long as it is the same for each function. Therefore, let  $\mathbf{T}_1, \mathbf{T}_2, \dots, \mathbf{T}_s$  be  $d$ -dimensional vectors,  $\mathbf{T}_i = (T_{i1}, T_{i2}, \dots, T_{id})$  for  $i = 1, \dots, s$ . In the case (i), the vectors  $\mathbf{T}_i$  are observed functions assumed to follow the same distribution and the aim is to find the least extreme  $100(1 - \alpha)\%$  of the  $s$  vectors to construct the central region, with  $\alpha \in (0, 1)$ . In the case (ii), the vector  $\mathbf{T}_1$  is an observed function and  $\mathbf{T}_2, \dots, \mathbf{T}_s$  are vectors simulated under the tested null hypothesis. The key idea of a global envelope test is the same as a general Monte Carlo or permutation test: the rank of the data vector  $\mathbf{T}_1$  among all the vectors  $\mathbf{T}_1, \dots, \mathbf{T}_s$  determines the result and the  $p$  value of the test (Barnard 1963). And, in the case (iii),  $\mathbf{T}_i$  are generated under the given bootstrap or Bayesian scheme. As mentioned already, in the core of all cases (i)-(iii) (see Section 1.1) lies the central region and, for the purpose of ordering the  $d$ -dimensional vectors  $\mathbf{T}_i$  (or functions) from the most extreme to the least extreme and finding the central region, many different measures exist. The **GET** package focuses on such measures for which it is possible to construct the global envelope with a practically interesting graphical interpretation, which we call IGI.

In general, an *envelope* is considered to be a band bounded by two vectors. A  $100(1 - \alpha)\%$  *global envelope* is a set  $(\mathbf{T}_{\text{low}}^{(\alpha)}, \mathbf{T}_{\text{upp}}^{(\alpha)})$  of envelope vectors with  $\mathbf{T}_{\text{low}}^{(\alpha)} = (T_{\text{low } 1}^{(\alpha)}, \dots, T_{\text{low } d}^{(\alpha)})$  and  $\mathbf{T}_{\text{upp}}^{(\alpha)} = (T_{\text{upp } 1}^{(\alpha)}, \dots, T_{\text{upp } d}^{(\alpha)})$  such that the probability that  $\mathbf{T}_i$  falls outside this envelope in any of the  $d$  points is equal to  $\alpha$ , for  $\alpha \in (0, 1)$ , i.e.,

$$P(T_{ik} \notin [T_{\text{low } k}^{(\alpha)}, T_{\text{upp } k}^{(\alpha)}] \text{ for any } k \in \{1, \dots, d\}) = \alpha. \quad (2)$$

In all cases (i)-(iii), *global* means that the envelope is given with the prescribed coverage  $100(1 - \alpha)\%$  simultaneously for all the elements of the multivariate or functional statistic, but the probability depends on the situation (i)-(iii). In case (i), the probability is taken under the distribution of the vectors  $\mathbf{T}_i$ . In case (ii), the probability is taken under the null hypothesis  $H_0$ , and, in case (iii), the probability is taken under the distribution of the random vector  $\mathbf{T}_i$  generated under the given bootstrap or Bayesian scheme. It should be noted that in a *pointwise* (or local) envelope the probability to fall out of the envelope is controlled instead individually for every element of the vector  $\mathbf{T}_i$ .

We define the IGI property for global envelopes as follows.

**Definition 2.1** *Assume that a general ordering  $\prec$  of the vectors  $\mathbf{T}_i, i = 1, \dots, s$ , is induced by a univariate measure  $M_i$ . That is,  $M_i \geq M_j$  iff  $\mathbf{T}_i \prec \mathbf{T}_j$ , which means that  $\mathbf{T}_i$  is less extreme or as extreme as  $\mathbf{T}_j$ . (The smaller the measure  $M_i$ , the more extreme the  $\mathbf{T}_i$  is.) The  $100(1 - \alpha)\%$  global envelope  $[\mathbf{T}_{\text{low } k}^{(\alpha)}, \mathbf{T}_{\text{upp } k}^{(\alpha)}]$  has intrinsic graphical interpretation (IGI) with respect to the ordering  $\prec$  if*

1.  $M_{(\alpha)} \in \mathbb{R}$  is the largest of the  $M_i$  such that the number of those  $i$  for which  $M_i < M_{(\alpha)}$  is less or equal to  $\alpha s$ ;
2.  $T_{ik} < T_{\text{low } k}^{(\alpha)}$  or  $T_{ik} > T_{\text{upp } k}^{(\alpha)}$  for some  $k = 1, \dots, d$  iff  $M_i < M_{(\alpha)}$  for every  $i = 1, \dots, s$ ;
3.  $T_{\text{low } k}^{(\alpha)} \leq T_{ij} \leq T_{\text{upp } k}^{(\alpha)}$  for all  $k = 1, \dots, d$  iff  $M_i \geq M_{(\alpha)}$  for every  $i = 1, \dots, s$ .

The points 2. and 3. are equivalent, but we specify them both in order to stress both graphical properties of IGI. The IGI property means that the vector  $\mathbf{T}_i$  is outside the  $100(1 - \alpha)\%$  global envelope in any of its components if and only if the vector is considered to be extreme by the measure  $M$  at the level  $\alpha$ , and the vector  $\mathbf{T}_i$  is completely inside the global envelope if and only if the vector is not extreme at the level  $\alpha$ . Thus, the global envelope with IGI provides a solution to the tasks (i)-(iii) in a graphical manner.

In particular in case (ii), the data vector  $\mathbf{T}_1$  is compared with a global envelope in order to decide if the data vector is extreme ( $H_0$  is rejected) or not extreme ( $H_0$  is not rejected), and to find reasons for the possible rejection of  $H_0$  through inspecting for which  $k = 1, \dots, d$  the data vector  $\mathbf{T}_1$  is outside the global envelope. For this testing task, in addition to a global envelope, a Monte Carlo  $p$  value is computed according to the measure  $M_i$ :  $p = \sum_{i=1}^s \mathbf{1}(M_i \leq M_1) / s$  (see, e.g., Myllymäki *et al.* 2017; Mrkvička *et al.* 2017; Mrkvička, Myllymäki, Jílek, and Hahn 2020; Mrkvička, Myllymäki, Kuronen, and Narisetty 2022; Mrkvička, Roskovec, and Rost 2021). In order to obtain an exact Monte Carlo test, i.e., a test that achieves the prescribed FWER, the exchangeability of the test vectors  $\mathbf{T}_i$  is required. All the examples in Section 3 satisfy exchangeability, except the functional GLM where the permutation of the residuals from the null model (Freedman and Lane 1983) is used. This permutation scheme

is commonly used in univariate permutation GLMs and widely accepted as the best available solution (Anderson and Robinson 2001; Anderson and Ter Braak 2003; Winkler *et al.* 2014).

## 2.2. Types of global envelopes

The **GET** package implements seven global envelopes defined in earlier works as specified in Table 1, together with their short descriptions and specifications in the **GET** functions. Definitions of these envelopes are given in Appendix A. The first four envelopes in Table 1 ('rank', 'er1', 'cont', 'area') are completely non-parametric envelopes and are called global rank envelopes, because they are all based on pointwise ranks of the vectors  $\mathbf{T}_i$ . The extreme rank length, continuous and area envelopes are refinements to the rank envelope. Note here that if a global FWER control is provided within other packages, it is only provided for not refined 'rank' envelope or 'unscaled' envelope. (see details in Appendix A). The 'st' and 'qdir' envelopes parameterize the marginal distributions of  $T_{ik}$ ,  $i = 1, \dots, s$  by one or two parameters, respectively, for all  $k = 1, \dots, d$ . Thus they can be regarded as approximations of the first four envelopes.

In a typical application one needs to choose one of the measures with IGI (Table 1). In general, the first five types of Table 1 instead of the last two, 'st' and 'unscaled', can be recommended based on previous studies (Myllymäki *et al.* 2015, 2017). Regarding the choice between the first five types, when one can afford a large number of simulations in cases (ii)-(iii) of Section 1.1, one can very well use the type 'er1' that is based only on the ranks, thus also suiting particularly well for combined tests where the test statistic is a combination of several functional test statistics (see Appendix B and examples in Sections 1.1 and 3.1). On the other hand, any other choice is also fine, because the 'rank', 'er1', 'cont' and 'area' measures lead to an equivalent outcome for a large number of simulations or permutations. However, the definition of *large* depends on the situation. A simulation study presented in Myllymäki and Mrkvička (2020) supplements this article, giving guidance on the required number of simulations under different scenarios (see also the supplement S2 of Myllymäki *et al.* 2017).

Another situation arises in case (i) with a low number of vectors or functions, or in cases (ii)-(iii) where the simulations or permutations are too time consuming to have a large number of them. Then the choice of the measure plays a role. Based on our experience supported by the simulation study presented in Myllymäki and Mrkvička (2020), the 'er1' and 'area' measures are typically good choices for detecting integral type of extremeness where the vector  $\mathbf{T}_i$  is extreme in the set of vectors for a large range of its components. On the other hand, the 'cont' and 'qdir' measures are most sensitive to the maximum type of extremeness, i.e., the case where  $\mathbf{T}_i$  is extreme only for a few of its components, but also the 'area' measure performs well. Thus, if no particular type of extremeness is expected a priori, the 'area' measure is often a good compromise, since it is sensitive to the amount of outlyingness (similarly as 'er1') and to the value of outlyingness (similarly as 'cont' and 'qdir'). Illustration of the different measures can be found in Mrkvička *et al.* (2022, Section 2.4).

To conclude, for testing, if just possible, we recommend to use a large number of simulations and one of the first five measures of Table 1. If large number of simulations is not possible, then still Barnard's Monte Carlo test is valid. For testing and ordering functional data, use then the measure which is sensitive to the type of extremeness that you regard as important

Type	Introduced in	Description
'rank'	Myllymäki <i>et al.</i> (2017)	Global rank envelope corresponding to the extreme rank measure (with ties); unique ordering (or $p$ value) provided additionally as specified in the argument <b>ties</b> , e.g., 'er1' for extreme rank lengths
'er1'	Myllymäki <i>et al.</i> (2017); Narisetty and Nair (2016); Mrkvička <i>et al.</i> (2020)	Global rank envelope corresponding to extreme rank length (ERL) measure
'cont'	Hahn (2015); Mrkvička <i>et al.</i> (2022)	Global rank envelope corresponding to the continuous rank measure
'area'	Mrkvička <i>et al.</i> (2022)	Global rank envelope corresponding to the area measure
'qdir'	Myllymäki <i>et al.</i> (2017, 2015)	Directional quantile envelope test corresponding to the directional quantile maximum absolute deviation (MAD) measure
'st'	Myllymäki <i>et al.</i> (2017, 2015)	Studentized envelope test corresponding to the studentized MAD measure
'unscaled'	Ripley (1981)	Unscaled envelope test corresponding to the classical, unscaled MAD measure. The envelope has a constant width.

Table 1: Overview of different types of global envelopes in the **GET** package. The types 'er1', 'cont' and 'area' refine the type 'rank' by breaking the ties in the extreme ranks, for details see Appendix A.

(see illustration of the different measures in Mrkvička *et al.* 2022, Section 2.4).

As it was mentioned earlier, other measures or depths without IGI can be used for the ordering the data. We believe that there exists no specific depth/measure which would perform universally the best and that for every depth/measure it is possible to construct a situation for which the chosen depth/measure will be the most powerful. However, for an application at hand, it is typically not possible to know the most powerful depth/measure; one has to make the choice of the depth/measure in advance. Therefore, we believe that it is often practical to choose the IGI measure, which brings an easier interpretation. In a particular study of the power of goodness of fit tests in spatial statistics our IGI measures were much more powerful than the modified band depth and modified half region depth (Myllymäki *et al.* 2017), but no universal conclusions can be drawn from a single study.

### 2.3. Implementation and key functions in GET

As described above, the construction of a global envelope is based on a measure  $M$ . The calculation of different measures in the **GET** package is provided by the function `forder()` (functional ordering). Most often, the user however calls either `central_region()` for constructing central regions with IGI or `global_envelope_test()` for performing global envelope tests (equipped with  $p$  values as well). Both functions utilize `forder()` for the calculation of the measures  $M$ . The most important arguments of these functions are

```
central_region(curve_sets, type = 'erl', coverage = 0.50, ...)
global_envelope_test(curve_sets, type = 'erl', alpha = 0.05, ...)
```

where the multivariate or functional data are provided in `curve_sets`, `type` specifies type of the global envelope (see Table 1 and descriptions in Appendix A), and the coverage or level of the global envelope is specified by `coverage` or `alpha` ( $= 1 - \text{coverage}$ ), respectively. Additionally, one can, for example, specify the one or two-sided `alternative`, i.e., whether only small or large values of  $\mathbf{T}_i$  or both should be considered extreme. These two functions are the core functions for global envelopes in the package **GET**: given an appropriate set of curves, or, in fact vectors, they can be used for producing global envelopes of Table 1 in all tasks (i)-(iii) listed in Section 1.1.

Different objects are supported for the data in `curve_sets` (see help files of the functions and examples below), but the basic form provided by the **GET** package is a ‘`curve_set`’ object that can be constructed by the function `create_curve_set()` simply providing a list containing the observed and/or simulated curves, and optionally the (one- or two-dimensional) argument values where the curves have been observed (see Section 3.2 for an example). The function `create_curve_set()` takes care of checking the appropriateness of the data, and saving the data in the form that contains the relevant information of the curves for global envelope methods, in particular for plotting the results with graphical interpretation (see examples in Section 3).

In addition to constructing global envelopes from a set of curves, the `central_region()` and `global_envelope_test()` functions provide combined central regions or combined global envelope tests if the user provides a list consisting of (appropriate) sets of curves in the argument `curve_sets`. Details of the combining methodology is given in Appendix B and examples are given in Sections 1.1 and 3. The **GET** package also provides functions for specific tasks (see Table 2 and the examples in Section 3). These functions utilize `central_region()` and `global_envelope_test()` for the global envelope construction. In addition, many of these functions take care of preparing the simulations or permutations for the specific testing task. The `print()` and `plot()` methods are available for the objects obtained as the output of the global envelope methods of **GET**. The plots present the results with IGI. They are produced using the **ggplot2** package (Wickham 2016).

## 3. Using GET

### 3.1. Sets of functions, ordering, combining, and central regions

The R package **fda** contains Berkeley Growth Study data (Ramsay and Silverman 2005) of the heights of 39 boys and 54 girls from ages 1 to 18 and the ages at which the data were collected. To illustrate different features of **GET**, we investigated whether there are any outliers in the girls regarding both their annual heights and changes within years. First two ‘`curve_set`’ objects were created containing the raw heights and the differences within the years (see Figure 3):

```
R> library("fda")
R> years <- paste(1:18)
R> curves <- growth[['hgtf']][years,]
```

Function name	Description
<code>create_curve_set()</code>	Create a 'curve_set' out of data given in the right form
<code>crop_curves()</code>	Crop curves
<code>forder()</code>	Different measures for ordering the multivariate statistics from the most extreme to least extreme one
<code>central_region()</code>	Central regions or global envelopes or confidence bands with IGI (see types in Table 1)
<code>global_envelope_test()</code>	Global envelope tests (see types in Table 1)
<code>GET.composite()</code>	Adjusted global envelope tests for composite null hypotheses
<code>fBoxplot()</code>	Functional boxplot based on a central region with IGI
<code>graph.fanova()</code>	One-way ANOVA tests for functional data with graphical interpretation (Mrkvička <i>et al.</i> 2020)
<code>frank.fanova()</code>	One-way functional ANOVA tests based on the global envelopes applied to $F$ values (Mrkvička <i>et al.</i> 2020)
<code>graph.flm()</code>	Non-parametric graphical tests of significance in functional general linear model (GLM) (Mrkvička <i>et al.</i> 2021)
<code>frank.flm()</code>	Global envelope tests applied to $F$ values in permutation inference for the GLM (Mrkvička <i>et al.</i> 2022)
<code>fclustering()</code>	Functional clustering based on a specified measure (Dai, Athanasiadis, and Mrkvička 2022)
<code>GET.necdf()</code>	Graphical $n$ sample test of correspondence of distribution functions
<code>GET.distrindep()</code>	Permutation-based tests of independence to samples from any bivariate distribution (Dvořák and Mrkvička 2022)
<code>GET.spatialF()</code>	Testing global and local covariate effects in point process models (Myllymäki, Kuronen, and Mrkvička 2020)

Table 2: Key and special purpose functions in the **GET** package. The complete list of functions can be found from the main help page of the package available by typing `help('GET-package')` in R.

```
R> cset1 <- create_curve_set(list(r = as.numeric(years), obs = curves))
R> cset2 <- create_curve_set(list(r = as.numeric(years[-1]),
+   obs = curves[-1,] - curves[-nrow(curves),]))
```

Ordering the functions from the most extreme to the least extreme by the 'area' measure (see Table 1 and Appendix A), the 8th girl was observed to have the most extreme heights and the 15th girl the most extreme changes (below the first ten most extreme girl indices are printed):

```
R> A1 <- forder(cset1, measure = 'area'); order(A1)[1:10]
```

```
[1] 8 13 29 48 42 25 7 38 18 40
```

```
R> A2 <- forder(cset2, measure = 'area'); order(A2)[1:10]
```

```
[1] 15 7 3 8 25 52 19 16 24 5
```

Generally, ordering with respect to heights or height differences leads to two different orderings of the girls. Combined ordering can be done by combining these two by the ERL measure as described in Appendix B (two-step combining procedure). In R, the two sets of curves need to be provided in a list to the function `forder()`:

```
R> csets <- list(Height = cset1, Change = cset2)
R> A <- forder(csets, measure = 'area'); order(A)[1:10]

[1] 8 15 7 13 3 29 48 25 42 52
```

Figure 3 highlights the curves of the three most extreme girls. The plots of the two sets of curves were produced using the **GET** and **ggplot2** packages and combined by the **patchwork** package (Pedersen 2020):

```
R> library("ggplot2")
R> library("patchwork")
R> cols <- c("#21908CFF", "#440154FF", "#5DC863FF")
R> p1 <- plot(cset1, idx = order(A)[1:3], col_idx = cols) +
+   labs(x = "Age (years)", y = "Height")
R> p2 <- plot(cset2, idx = order(A)[1:3], col_idx = cols) +
+   labs(x = "Age (years)", y = "Change")
R> p1 + p2 + plot_layout(guides = "collect")
```

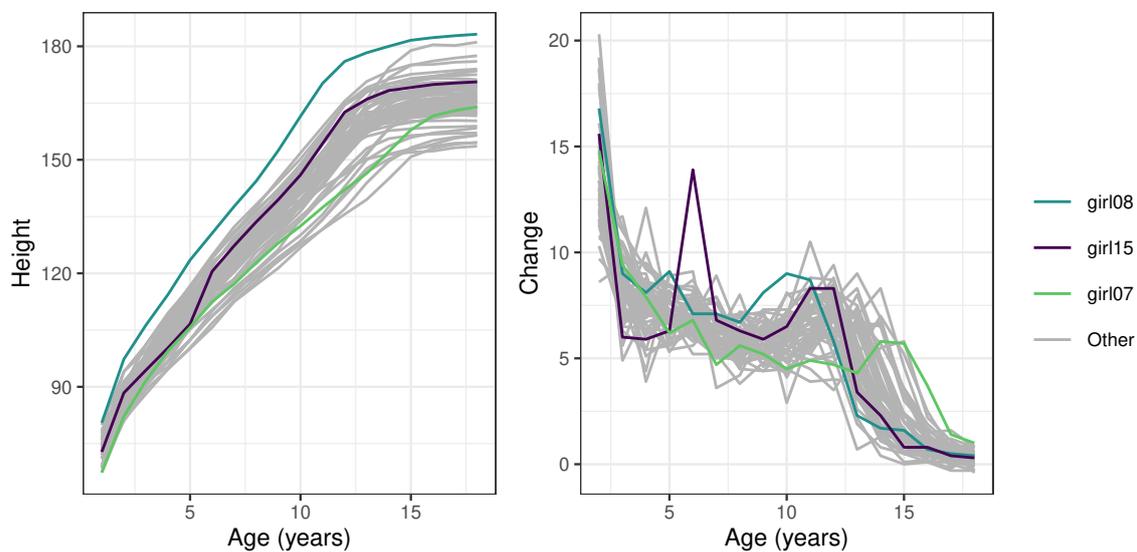


Figure 3: The heights (left) and height differences (right) of the 54 girls of the `growth` data of the R package `fda` at ages from 1 to 18. Three girls having the most extreme curves (combined ordering by the area measure) are highlighted with the colors specified in the legend.

The labels were above redefined for the default plots by the function `labs()` of the **ggplot2** package. In general, parts of the default plots of **GET** can be edited in this manner using **ggplot2** functions such as `labs()` and `ggtitle()`.

The combined central region can be constructed directly by passing the sets of curves to the function `central_region()`, in a similar manner as for `forder()` above. By using the functional boxplot (Sun and Genton 2011) with the same measure and central region, an investigation can be made into whether the most extreme girls are outliers with respect to height or its change. Figure 4 shows the 50% central region and the functional boxplot with the inflation factor 1.5 jointly for the heights and their changes obtained by:

```
R> res <- fBoxplot(csets, type = 'area', factor = 1.5)
R> plot(res) + labs(x = "Age (years)", y = "Value")
```

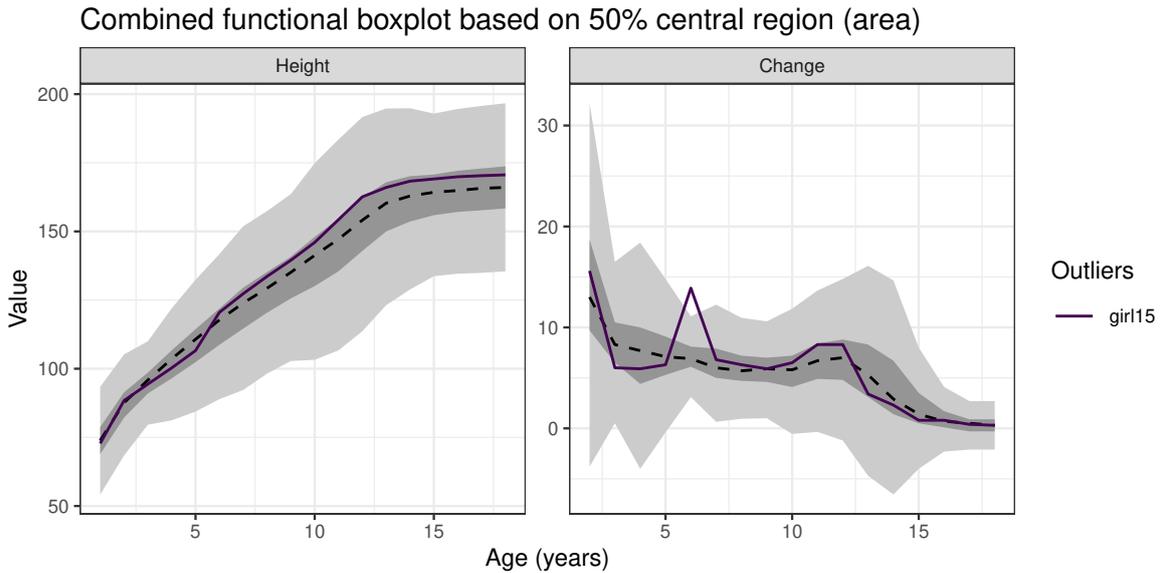


Figure 4: The functional boxplot (entire gray band) using the 50% central region (inner dark gray band) and the expansion factor 1.5 jointly for the heights and changes of heights of the 54 girls (see Figure 3). The solid line is an observed data function (vector) that goes outside the functional boxplot (the 15th girl in the data).

One can see that one of the girls (the 15th girl in fact) is an outlier, because she has grown extraordinarily much in her sixth year. However, the highest height curve of Figure 3 (left) is not regarded as an outlier with the given specifications.

Note that the combined central region computed using any measure of Table 1 has IGI. On the contrary, central regions computed with the use of band depths implemented in the **fd** package do not satisfy IGI. Narisetty and Nair (2016) proposed central regions and functional boxplots based on the ERL measure (see Table 1) and compared them to those based on band depths, claiming more reasonable behavior. Note also that IGI is not a property of the functional boxplot.

### 3.2. Monte Carlo goodness-of-fit testing for simple hypotheses: complete spatial randomness

Figure 5 shows the locations of 67 large trees (with height  $> 25$  m) in an area of size 75 m  $\times$  75 m from an uneven aged multi-species broadleaf nonmanaged forest in Kaluzhskie Zaseki,

Russia (Grabarnik and Chiu 2002; van Lieshout 2010). The  $x$ - and  $y$ -coordinates of the locations are available in the data `adult_trees` in the **GET** package.

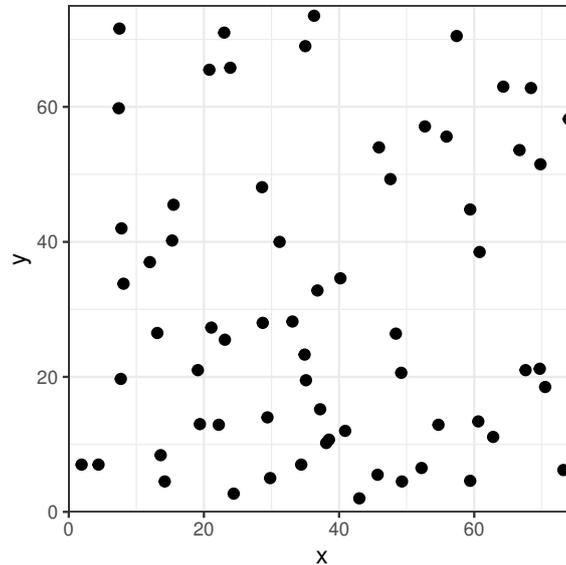


Figure 5: Locations of 67 trees with height  $> 25$  m observed in an area of  $75 \text{ m} \times 75 \text{ m}$ .

The test of complete spatial randomness (CSR) is a typical first step in analyzing a spatial point pattern such as the tree pattern of Figure 5. CSR along with other hypotheses for spatial point patterns are commonly tested using an estimator of a summary function that is a function of distance  $r$ , e.g., Ripley's  $K$  function or its transformation  $L(r) = \sqrt{K(r)/\pi} - r$  for  $r \geq 0$  (Ripley 1977; Besag 1977). In this context, one typically resorts to the Monte Carlo simulation (see, e.g., Illian, Penttinen, Stoyan, and Stoyan 2008; Diggle 2013; Myllymäki *et al.* 2017). First, this example is used to show the general steps to prepare a global envelope test for testing a simple hypothesis. Second, it is shown how the same example of testing a simple hypothesis for a spatial point pattern can be performed by utilizing the R package **spatstat**.

The testing of a simple hypothesis does not require the estimation of any model parameters, and the one-stage test illustrated below can be used. In the case of a composite null hypothesis, the level of the test needs to be adjusted, see Section 3.3 and Appendix C.

#### *General setup for simple hypotheses*

The first step of a Monte Carlo test is to generate `nsim` simulations under the null hypothesis and to calculate the chosen test function (vector) for the data and simulations. Here the functions `runifpoint()` and `Lest()` of **spatstat** are used to generate a simulation from the binomial process (CSR with the number of points fixed to the observed number of points in the pattern `X`) and to estimate the centred  $L$ -function for a pattern, respectively:

```
R> library("spatstat.explore")
R> data("adult_trees")
R> X <- as.ppp(adult_trees, W = square(75))
R> nsim <- 999
```

```
R> obs.L <- Lest(X, correction = "translate")
R> r <- obs.L[['r']]
R> obs <- obs.L[['trans']] - r
R> sim <- matrix(nrow = length(r), ncol = nsim)
R> for(i in 1:nsim) {
+   sim.X <- runifpoint(ex = X)
+   sim[, i] <- Lest(sim.X, correction = "translate", r = r)[['trans']] - r
+ }
```

Thereafter, a ‘`curve_set`’ object can be constructed by the function `create_curve_set()` providing a list of the argument values where the test vectors were evaluated (`r`), the observed vector (`obs`) and the simulated vectors (`sim_m`):

```
R> cset <- create_curve_set(list(r = r, obs = obs, sim_m = sim))
```

In some cases, missing or infinite values can occur for the computed statistic at some chosen  $r$  (e.g., for too large  $r$  with another spatial summary function  $J$ ). These problematic, uninteresting  $r$ -values can be easily omitted from `cset` using the function `crop_curves()`. The final step is to make the global envelope test on the given set of vectors:

```
R> res <- global_envelope_test(cset, type = "erl")
R> plot(res) + ylab(expression(hat(L)(r)-r))
```

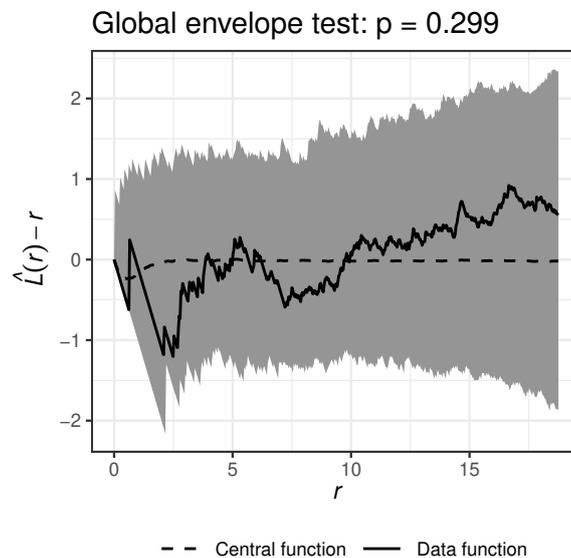


Figure 6: The global envelope test for the CSR of the tree pattern of Figure 5 using the centred  $L$ -function. The gray band represents the 95% global envelope (ERL).

In this manner, the global envelope test can be constructed for any simple hypothesis and any test vector, as long as one can generate the simulations and calculate the test vectors.

The test output is shown in Figure 6 (left), which shows no evidence against CSR (see more detailed description in Myllymäki *et al.* 2017, Section S4).

### *Simple hypothesis for a point pattern utilizing the R package **spatstat***

For point process testing, the **GET** package and `global_envelope_test()` support the use of the R package **spatstat** (Baddeley *et al.* 2015) for the simulations and calculations of the summary functions by the function `envelope()`. Namely, the object returned by `envelope()` can simply be given to the function `global_envelope_test()` in the argument `curve_sets`. Importantly, the functions must be saved setting `savefuns = TRUE` in the `envelope()` call:

```
R> env <- envelope(X, nsim = 999, fun = "Lest", correction = "translate",
+   transform = expression(.-r), simulate = expression(runifpoint(ex = X)),
+   savefuns = TRUE, verbose = FALSE)
R> res <- global_envelope_test(env, type = "erl")
```

Above the arguments `fun`, `correction` and `transform` define the summary function to be calculated (the latter two parameters are passed to the function `Lest()`) and `simulate` specifies how the patterns are simulated under the null hypothesis (here CSR). The result can be plotted similarly as above.

Further examples of use of the **GET** package for point pattern analysis are given in an accompanying vignette available in R by typing `library("GET")` and `vignette("pointpatterns")`.

### **3.3. Monte Carlo goodness-of-fit testing for composite hypotheses: graphical normality test**

Aldor-Noiman *et al.* (2013) provided a graphical test for normality for simple hypotheses (i.e., known parameters of sample distribution) based on a qq-plot envelope, whose shape was derived from theoretical properties of quantiles of the uniform distribution. They also provided a version of this algorithm for composite hypotheses (i.e., unknown parameters of sample distribution). However, according to our unpublished study, this test does not achieve the required significance level. Therefore, the example of the exact adjustment for the composite hypothesis is provided here, based on the two-stage procedure of Baddeley *et al.* (2017) (see Appendix C). For simplicity, in this example, this adjustment is applied directly to the empirical distribution functions. Apparently, the adjustment could also be applied to the qq-plot envelopes of Aldor-Noiman *et al.* (2013).

The normality test is illustrated for nitrogen oxides ( $\text{NO}_x$ ) emission levels available in the data `poblenou` from the R package `fda.usc` (Febrero-Bande and Oviedo de la Fuente 2012). The data contains  $\text{NO}_x$  emission levels ( $\mu\text{g}/\text{m}^3$ ) measured every hour by a control station close to an industrial area in Poblenou in Barcelona (Spain) for 115 days from 23 February to 26 June, 2005.  $\text{NO}_x$  is a pollutant which is caused by combustion processes in sources that burn fuels, e.g., motor vehicles, electric utilities, and industries (Febrero, Galeano, and González-Manteiga 2008). In Section 3.5, the whole functional trajectories of 24 h observations are studied, but for illustration purposes, here the attention is restricted to the  $\text{NO}_x$  levels at 10 am.

A general solution to make the adjusted test is to prepare all the required simulations and provide them to the function `GET.composite()` in arguments `X` and `X.ls`. Let `dat` be a vector containing the data values and `n` is the number of observations specified as follows:

```
R> library("fda.usc")
R> data("poblenou")
```

```
R> dat <- poblenou[['nox']] [['data']] [, 'H10']
R> n <- length(dat)
```

Then, first, the parameters of the normal distribution are estimated (1. step of the algorithm of Appendix C)

```
R> mu <- mean(dat)
R> sigma <- sd(dat)
```

and, using the function `ecdf()` of the R package **stats** (R Core Team 2020), the empirical cumulative distribution functions are calculated for the data and for `nsimsub` replicates of  $n$  simulations from the fitted normal distribution (2. step):

```
R> nsim <- nsimsub <- 199 # The number of simulations
R> r <- seq(min(dat), max(dat), length = 100)
R> obs <- stats::ecdf(dat)(r)
R> sim <- replicate(nsimsub, {
+   x <- rnorm(n, mean = mu, sd = sigma)
+   stats::ecdf(x)(r)
+ })
R> cset <- create_curve_set(list(r = r, obs = obs, sim_m = sim))
```

Here the last command creates a ‘`curve_set`’ object of the observed and simulated empirical cumulative distribution functions. Thereafter, another `nsim` replicates of the  $n$  simulations from the fitted model are simulated, and the same calculations as above for the data are done for each of these simulations (steps 3.-4. of the algorithm of Appendix C):

```
R> cset.ls <- list()
R> for(rep in 1:nsim) {
+   x <- rnorm(n, mean = mu, sd = sigma)
+   mu2 <- mean(x)
+   sigma2 <- sd(x)
+   obs2 <- stats::ecdf(x)(r)
+   sim2 <- replicate(nsimsub, {
+     x2 <- rnorm(n, mean = mu2, sd = sigma2)
+     stats::ecdf(x2)(r)
+   })
+   cset.ls[[rep]] <- create_curve_set(list(r = r, obs = obs2,
+     sim_m = sim2))
+ }
```

Thus, the list `cset.ls` contains all the simulations from the second stage of the algorithm. As a final step, `GET.composite()` can be used to prepare the adjusted test:

```
R> res <- GET.composite(X = cset, X.ls = cset.ls, type = 'erl')
R> plot(res) + labs(x = "NOx", y = "Ecdf")
```

Figure 7 (left) shows the test result for the  $\text{NO}_x$  levels at 10 am. One can see that the normality does not hold according to the test: the estimated distribution function is skewed

to the right with respect to the normal envelope. Therefore, we further applied the same normality test to the logarithm of the  $\text{NO}_x$  values as well, and then the normality hypothesis was not rejected (Figure 7, right).

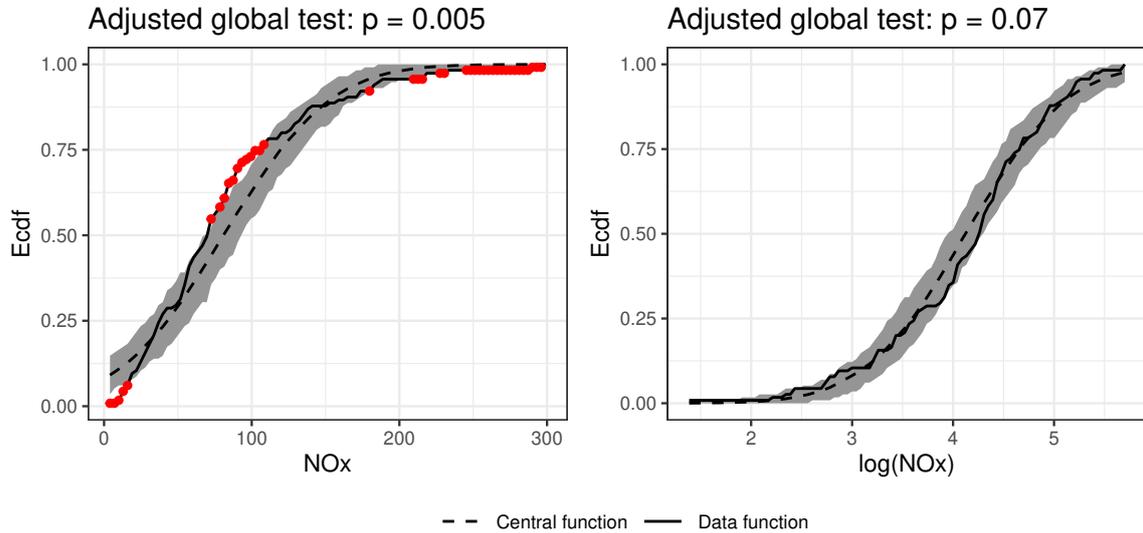


Figure 7: Graphical normality test for the  $\text{NO}_x$  (left) and logarithm of the  $\text{NO}_x$  (right) levels at 10 am. The gray band represents the 95% global envelope (ERL). Red dots are attached to the data function outside the envelope.

### 3.4. Graphical $n$ -sample test of correspondence of distribution functions

The graphical  $n$ -sample test of correspondence of distribution functions serves as a simple example of permutation tests. Figure 8 shows the empirical cumulative distribution functions (ECDFs) obtained by the `ecdf()` of the R package `stats` (R Core Team 2020) for the heights of the 54 girls and 39 boys of the `growth` data (see above, and Ramsay and Silverman 2005) at ages 10 (left) and 14 (right). A global envelope test can be performed to investigate whether the two (or more generally  $n$ ) distribution functions differ from each other significantly and how they differ. This test is a generalization of the two-sample Kolmogorov-Smirnov test with a graphical interpretation. The generalization is provided in two ways, in the number of samples and in the variable width of the envelopes. Namely, the Kolmogorov-Smirnov test provides an envelope of constant width which suffers in determining differences in extreme quantiles. Here it is assumed that the heights in the sample  $i$  are an i.i.d. sample from the distribution  $F_i(r)$ ,  $i = 1, \dots, n$ , and the hypothesis  $F_1(r) = \dots = F_n(r)$  is to be tested. The simulations under the null hypothesis that the distributions are the same can be obtained by permuting the individuals of the groups. The `GET` package provides the wrapper function `GET.necdf()` that can be used to compare  $n$  distribution functions graphically,  $n = 2, 3, \dots$ . The (default) test vector is

$$\mathbf{T} = (\hat{F}_1(r), \dots, \hat{F}_n(r)),$$

where  $\hat{F}_i(r) = (\hat{F}_i(r_1), \dots, \hat{F}_i(r_d))$  is the ECDF of the  $i$ th sample evaluated at argument values  $r = (r_1, \dots, r_d)$ . To test the equality of distributions, one simply needs to provide the

samples as a list (code for age 10 shown here) for `GET.necdf()` and plot the object returned by `GET.necdf()` (Figure 9, left):

```
R> fm10.1 <- list('Girls, 10 yr.' = growth$hgtf["10",],
+               'Boys, 10 yr.' = growth$hgtm["10",])
R> res10 <- GET.necdf(fm10.1, nsim = 1999)
R> myxlab <- substitute(paste(italic(i), "(", j, ")", sep = ""),
+                       list(i = "x", j = "Height in cm"))
R> plot(res10) + xlab(myxlab)
```

The height distributions at age 10 do not differ from each other significantly, but at age 14 the boys are taller, particularly with a difference that the proportion of girls reaching a height of around 175 cm is much lower (Figure 9, right).

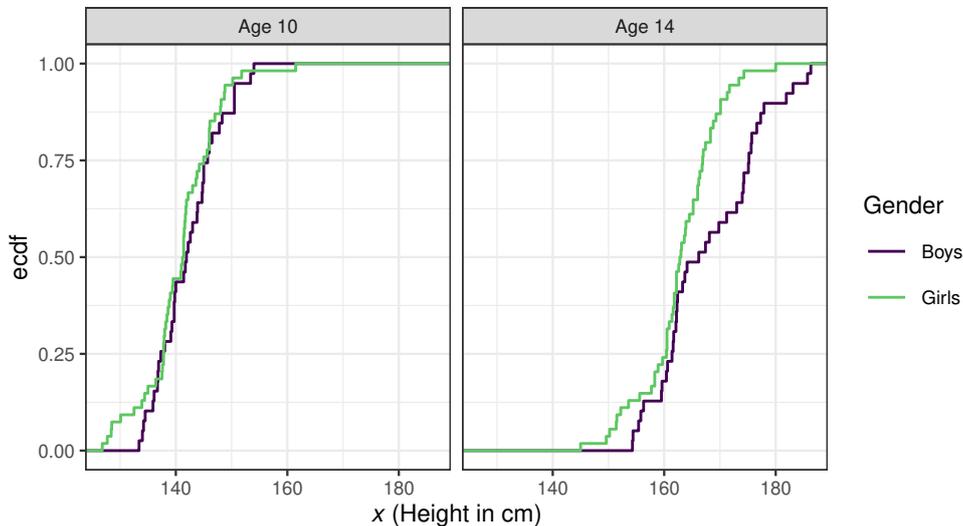


Figure 8: The empirical cumulative distribution functions of the heights of the 54 girls and 39 boys of the `growth` data of the R package `fda` at ages 10 (left) and 14 (right).

### 3.5. Graphical functional one-way ANOVA

The use of the function `graph.fanova()` of the `GET` package for the graphical functional one-way ANOVA is illustrated using the data set `poblenuou` of the R package `fda.usc` (Febrero-Bande and Oviedo de la Fuente 2012, see also Section 3.3 above). The trajectories of the 24 h observations of the  $\text{NO}_x$  levels for Monday-Thursday (MonThu), Friday (Fri) and non-working days (Free) including weekend and festive days (Figure 10) are compared. For the purposes of this example, a factor vector `Type` was prepared containing the type of the day for each of the 115 days having levels "MonThu", "Fri" and "Free":

```
R> library("fda.usc")
R> data("poblenuou")
R> fest <- poblenuou$df$day.festive; week <- as.integer(poblenuou$df$day.week)
R> Type <- vector(length = length(fest))
```

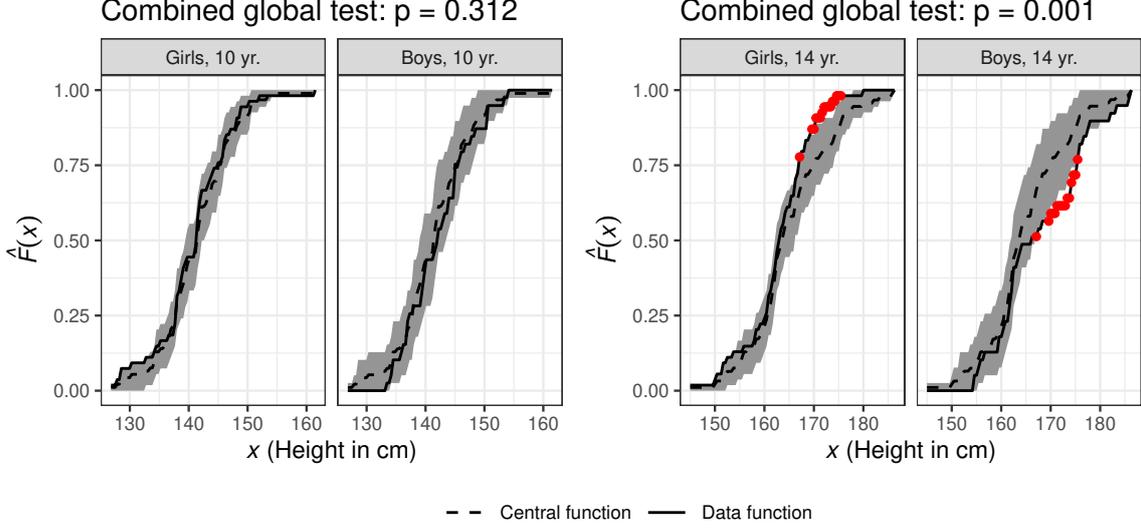


Figure 9: Global envelope tests for comparison of the empirical cumulative distribution functions of the heights of the girls and boys (see Figure 8). Red color indicates the heights where the observed distribution functions go outside the 95% global envelope (ERL; gray bands). Left: Age 10; Right: Age 14.

```
R> Type[fest == 1 | week >= 6] <- "Free"
R> Type[fest == 0 & week %in% 1:4] <- "MonThu"
R> Type[fest == 0 & week == 5] <- "Fri"
R> Type <- factor(Type, levels = c("MonThu", "Fri", "Free"))
```

Assuming that the  $\text{NO}_x$  levels  $T_{ij}(r)$  at times  $r \in R = [0, 24]$  are i.i.d. samples from stochastic processes  $SP(\mu_j, \gamma_j)$  with mean functions  $\mu_j(r)$ ,  $r \in R$ , and covariance functions  $\gamma_j(s, t)$ ,  $s, t \in R$ , for  $j = 1, \dots, J$  (here  $J = 3$ ), the groups of  $\text{NO}_x$  levels can be compared by means of the graphical functional ANOVA (Mrkvička *et al.* 2020). The hypothesis

$$H_0 : \mu_1(r) = \dots = \mu_J(r), r \in R,$$

can be tested by the test statistic

$$\mathbf{T} = (\bar{T}_1, \bar{T}_2, \dots, \bar{T}_J), \quad (3)$$

where  $\bar{T}_j = (\bar{T}_j(r_1), \dots, \bar{T}_j(r_d))$  is the mean of functions in the  $j$ th group at the discrete number of arguments  $r_1, \dots, r_d$  (here each hour of the day). The hypothesis can be equivalently expressed as

$$H'_0 : \mu_{j'}(r) - \mu_j(r) = 0, r \in R, j' = 1, \dots, J-1, j = j', \dots, J$$

and an alternative test statistic is

$$\mathbf{T}' = (\bar{T}_1 - \bar{T}_2, \bar{T}_1 - \bar{T}_3, \dots, \bar{T}_{J-1} - \bar{T}_J), \quad (4)$$

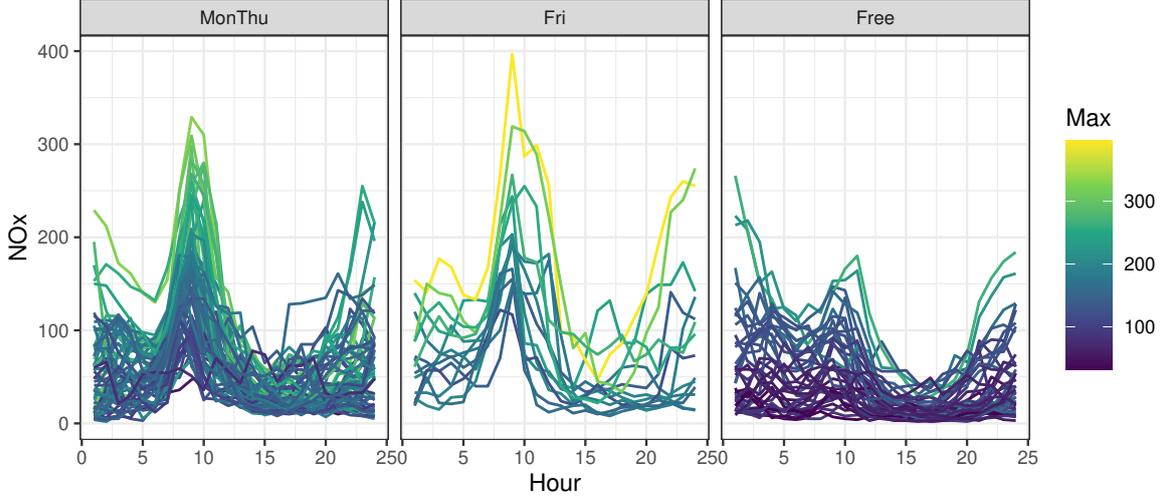


Figure 10: The  $\text{NO}_x$  levels for Monday-Thursday (Mon-Thu), Friday (Fri) and non-working days (Free) including weekend and festive days in Poblenou for 115 days from 23 February to 26 June, 2005. The color of the daily curves is according to the maximum  $\text{NO}_x$  level ( $\mu\text{g}/\text{m}^3$ ) of the day.

where we used the notation  $\bar{T}_j - \bar{T}_{j'} = (\bar{T}_j(r_1) - \bar{T}_{j'}(r_1), \dots, \bar{T}_j(r_d) - \bar{T}_{j'}(r_d))$ . The latter test statistic (Equation 4) can be obtained by setting `contrasts = TRUE` in the call of `graph.fanova()`.

Because the ANOVA design assumes equal covariance functions, we first tested for the homoscedasticity using the test proposed by Mrkvička *et al.* (2020, Section 2.3) and implemented in **GET**. We created the ‘`curve_set`’ object

```
R> cset <- create_curve_set(list(r = 0:23,
+   obs = t(poblenou[['nox']][['data']])))
```

and we used the function `graph.fanova()` with the option `test.equality = "var"` to test the equality of variances (Figure 11):

```
R> res.v <- graph.fanova(nsim = 2999, curve_set = cset, groups = Type,
+   test.equality = "var")
R> myxlab <- substitute(paste(italic(i), " (", j, ")", sep = ""),
+   list(i = "r", j = "Hour of day"))
R> plot(res.v) + xlab(myxlab)
```

Here the test suggests heteroscedasticity and also Febrero *et al.* (2008) assumed heteroscedasticity of working and non-working days. Therefore, we applied correction for unequal variances to the three groups by rescaling the functions  $T_{ij}(r)$  of  $J$  groups containing  $n_1, \dots, n_J$  functions observed on the finite interval  $R = [0, 24]$  by the transformation

$$Y_{ij}(r) = \frac{T_{ij}(r) - \bar{T}_j(r)}{\sqrt{\text{Var}(T_j(r))}} \cdot \sqrt{\text{Var}(T(r)) + \bar{T}_j(r)}, \quad j = 1, \dots, J, i = 1, \dots, n_j, \quad (5)$$

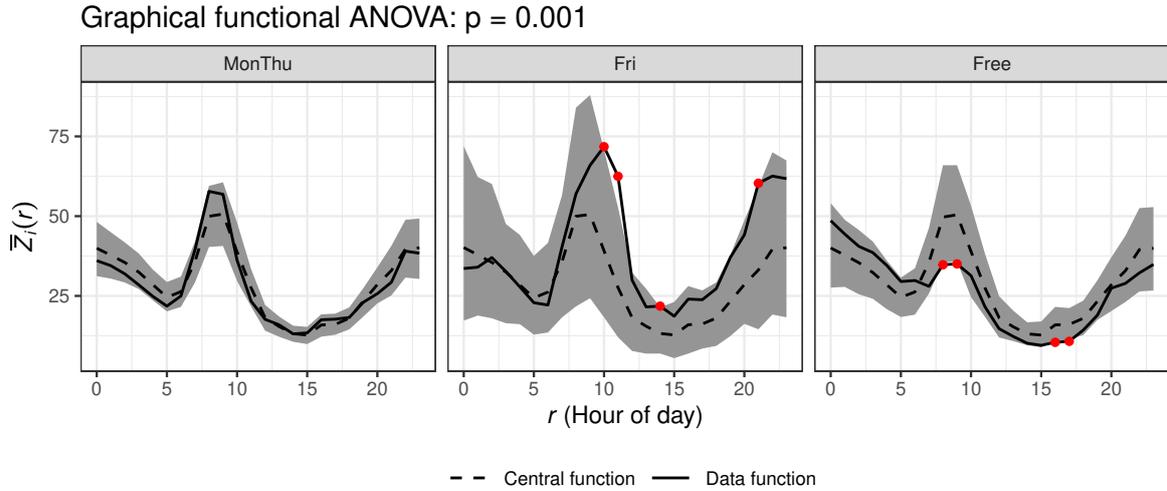


Figure 11: The test for the equality of variances of the  $\text{NO}_x$  levels observed each hour ( $r$ ) of the day. The 95% global envelope (ERL; gray band) representing the null hypothesis of equal variances and the observed test statistic, the mean of functions  $Z_{ij}(r) = |T_{ij}(r) - \bar{T}_j(r)|$  in each group (solid line with red dots when outside the envelope).

where the group sample mean  $\bar{T}_j(r)$  and overall sample variance  $\text{Var}(T(r))$  are involved to keep the mean and variability of the functions at the original scale. The group sample variance  $\text{Var}(T_j(r))$  corrects the unequal variances. This scaling is applied to the set of curves given to the function `graph.fanova()` if the user specifies `variances = "unequal"` (the default is no correction, `variances = "equal"`). When using the rescaled functions, the test vectors (Equations 3 and 4) are asymptotically exchangeable under the null hypothesis of equal means only under the assumption of normality of stochastic processes  $SP(\mu_j, \gamma_j)$  (Mrkvička *et al.* 2020). Therefore, the log transformation was applied to the  $\text{NO}_x$  values prior to the transformation (Equation 5):

```
R> lcset <- create_curve_set(list(r = 0:23,
+   obs = t(log(poblenou[['nox']] [['data']])))
```

To sample from the null hypotheses, the simple permutation of raw functions among the groups is performed. The permutations and the global envelope test can be done by the `graph.fanova()` function (Figure 12):

```
R> res.c <- graph.fanova(nsim = 2999, curve_set = lcset, groups = Type,
+   variances = "unequal", contrasts = TRUE)
R> plot(res.c) + xlab(myxlab)
```

Thus, the test rejects the null hypothesis  $H'_0$  that the differences between the groups would be zero and shows that on Monday-Thursday and Friday the (log)  $\text{NO}_x$  levels are significantly larger than on free days basically during the whole day with peaks around 8 am and 4 pm. The difference between Monday-Thursday and Friday was not significant.

The graphical functional ANOVA allows one to detect either a) which groups deviate from the mean (default) or b) which specific groups are different (option `contrasts = TRUE`). The

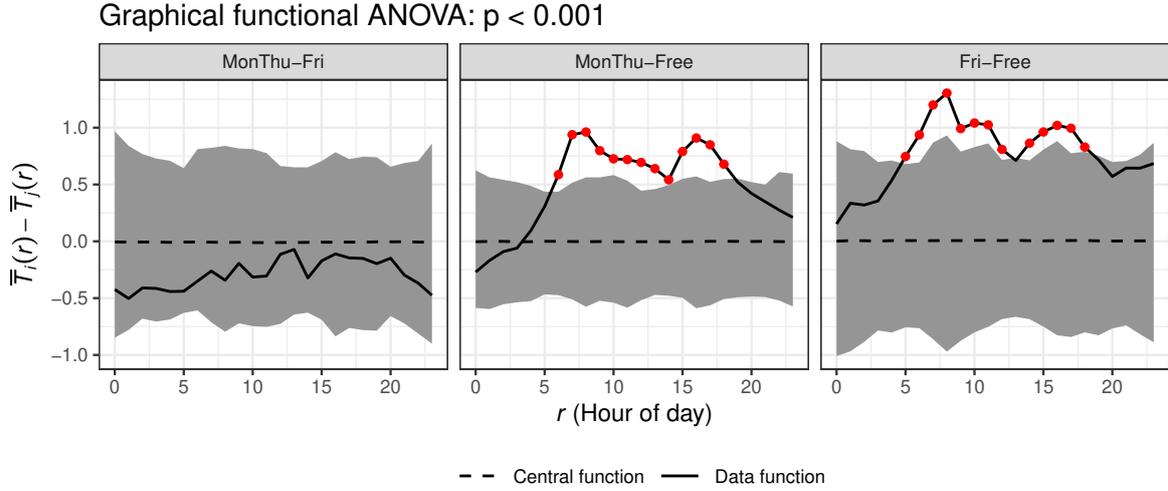


Figure 12: The output of the graphical functional ANOVA to test the difference between the type of the day on the log  $\text{NO}_x$  levels observed each hour ( $r$ ) of the day. The 95% global envelope (ERL; gray band) accompanied with the observed differences between the group means (solid line with red dots when outside the envelope).

example above was for the latter. Note that this test directly has the nature of a post hoc test. Furthermore, both versions of the test allow one to identify which  $r$  values lead to the potential rejection of the null hypothesis.

When a graphical interpretation for group specific differences is not of interest but the area of rejection is, instead of `graph.fanova()` it is possible to apply the one-way functional ANOVA based on the  $r$ -wise  $F$  statistics,  $r \in R$ . This test is implemented in the function `frank.fanova()`. For the log  $\text{NO}_x$  data, the test result was that there are differences between the groups for the hours from 5 am to 6 pm (figure omitted).

### 3.6. Functional GLM and two-dimensional plots

Similar type of methods as in the functional one-way ANOVA above can be used in a more general setup of functional general linear models (GLMs). The global envelopes for functional GLMs are illustrated here by an example of a small subset of the autism brain imaging data collected by resting state functional magnetic resonance imaging (R-fMRI) (Di Martino, Yan, Li, Denio, Castellanos, Alaerts, Anderson, Assaf, Bookheimer, Dapretto, and et al. 2014). The preprocessed fMRI data contains measurements from 514 individuals with the autism spectrum disorder (ASD) and 557 typical controls (TC), where subjects with low quality on imaging data or having a large proportion of the missing values were removed. The imaging measurement for local brain activity at resting state was fractional amplitude of low frequency fluctuations (Zou, Zhu, Yang, Zuo, Long, Cao, Wang, and Zang 2008). The data considered here and available as the data object `abide_9002_23` in the **GET** package contains data from one of the 116 different anatomical regions in the brain partitioning being based on the anatomical automatic labeling system of Tzourio-Mazoyer, Landeau, Papathanassiou, Crivello, Etard, Delcroix, Mazoyer, and Joliot (2002). The studied region is the right Crus Cerebellum 1 region of the brain at one slice (23) accompanied with three subject-specific

factors, i.e., group (autism and control), sex and age. Figure 13 obtained by

```
R> data("abide_9002_23")
R> plot(abide_9002_23[['curve_set']], idx = c(1, 27))
```

shows the data for two subjects, illustrating the small region used as the example. In the examples below, the effect of the group on the images is studied in the presence of nuisance factors sex and age.

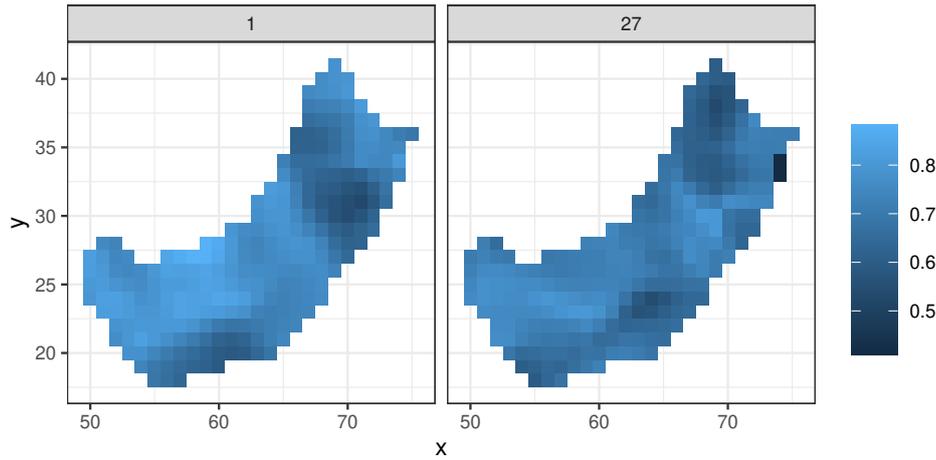


Figure 13: The example brain image data in the right Crus Cerebellum 1 region at a slice for one subject from the autism group (subject number 1) and one from the control group (subject number 27).

### Graphical functional GLM

The functional GLM is the general linear model

$$\mathbf{Y}(r) = \mathbf{X}(r)\boldsymbol{\beta}(r) + \mathbf{Z}(r)\boldsymbol{\gamma}(r) + \boldsymbol{\epsilon}(r) \quad (6)$$

where the argument  $r \in \{1, \dots, d\}$  determines the component of the vector or the spatial point or pixel of an image. For every argument  $r$ , a one-dimensional GLM is considered with  $\mathbf{X}(r)$  being a  $n \times k$  matrix of regressors of interest (here group),  $\mathbf{Z}(r)$  being a  $n \times l$  matrix of nuisance regressors (here sex and age),  $\mathbf{Y}(r)$  being a  $n \times 1$  vector of observed data, and  $\boldsymbol{\epsilon}(r)$  being a  $n \times 1$  vector of random errors with a mean of zero and a finite variance  $\sigma^2(r)$  for every  $r \in I$ . Further,  $\boldsymbol{\beta}(r)$  and  $\boldsymbol{\gamma}(r)$  are the regression coefficient vectors of dimensions  $k \times 1$  and  $l \times 1$ , respectively, and the null hypothesis to be tested is

$$H_0 : \beta_i(r) = 0, \quad \forall r = 1, \dots, d, \quad \forall i = 1, \dots, k,$$

where  $\beta_i(r)$  are the elements of the  $\boldsymbol{\beta}(r)$ . For a continuous factor of interest  $k = 1$  and  $\boldsymbol{\beta}(r)$  serves as the test statistic. For a discrete factor of interest, in the default setup,  $k$  is equal to the number of groups of the discrete factor, adding the additional condition that  $\sum_i \beta_i(r) = 0$  for all  $r \in \{1, \dots, d\}$ . Similarly, for interaction of a continuous and a discrete factor,  $k$  is also equal to the number of groups of the categorical factor, adding the same additional condition.

For the interaction of two discrete factors,  $k$  is equal to the product of the numbers of groups of the discrete factors, adding the same additional condition. The default test statistic for a discrete factor consist of  $\beta_i(r)$  for all  $r = 1, \dots, d$  and  $i = 1, \dots, k$ . This test allows to detect which groups deviate from the zero (mean). An alternative test statistic is obtained by the setting the argument `contrasts` of the function `graph.flm()` to `TRUE`: then the test statistic is formed by all the pairwise differences between the group effects,  $\beta_i(r) - \beta_j(r)$  for all  $r$  and  $i \neq j$ ,  $1 \leq i < j \leq k$ . This test specifies which specific groups are different in the post hoc nature. Note that this also holds for interaction terms. Furthermore, all the options allow one to identify which of the components of the vector,  $r \in \{1, \dots, d\}$ , lead to the potential rejection of the null hypothesis. Permutations under the null hypothesis are obtained using the Freedman-Lane procedure (Freedman and Lane 1983; Mrkvička *et al.* 2022; Mrkvička *et al.* 2021).

Often factors are given for the whole function, i.e., they do not depend on argument  $r$ , and so the matrices  $\mathbf{X}(r)$  and  $\mathbf{Z}(r)$  are identical for every  $r$ . These kind of constant factors (such as sex and age in the considered example) can be provided in the argument `factors` of the `graph.flm()` function. However, this simplification is not necessary and factors varying in space can be provided in the argument `curve_sets`, along with the data curves in a named list.

The functional GLM is performed by the function `graph.flm()`:

```
R> res <- graph.flm(nsim = 999, formula.full = Y ~ Group + Sex + Age,
+   formula.reduced = Y ~ Sex + Age,
+   curve_sets = list(Y = abide_9002_23[['curve_set']]),
+   factors = abide_9002_23[['factors']], contrasts = TRUE,
+   GET.args = list(type = "area"))
```

Here the arguments `formula.full` and `formula.reduced` specify the full GLM and the GLM where the interesting factor has been dropped out, and the number of simulations is given in `nsim`. Further arguments to `global_envelope_test()` can be passed in `GET.args`, e.g., the type of the global envelope.

The `r` component of the `abide_9002_23[['curve_set']]` object is a data frame with columns `x`, `y`, `width` and `height`, where the `width` and `height` give the width and height of the pixels placed at `x` and `y`. When such two-dimensional argument values are provided in a `curve_set` object, the resulting default envelope plots produced by

```
R> plot(res) + scale_radius(range = 1.5 * c(1, 6))
```

are two-dimensional as well (Figure 14). Above `scale_radius()` function of `ggplot2` is used to enlarge the circles which show the size of deviation of the observed function from the bound (here lower) of the envelope divided by the width of the envelope. Here only two groups were compared, and the plot shows that the brain measurements were lower in the autism group than in the control group in a part of the small example region (blue circles in Figure 14).

When the basic assumption of the homoscedasticity in the linear model (6) for every argument  $r$  is violated, it is important to handle it. One possibility is to apply transformations to the functions a priori as suggested by Mrkvička *et al.* (2020) and Mrkvička *et al.* (2021) (see Equation 5). Alternatively weighted least squares might be used for estimation of regression coefficients.

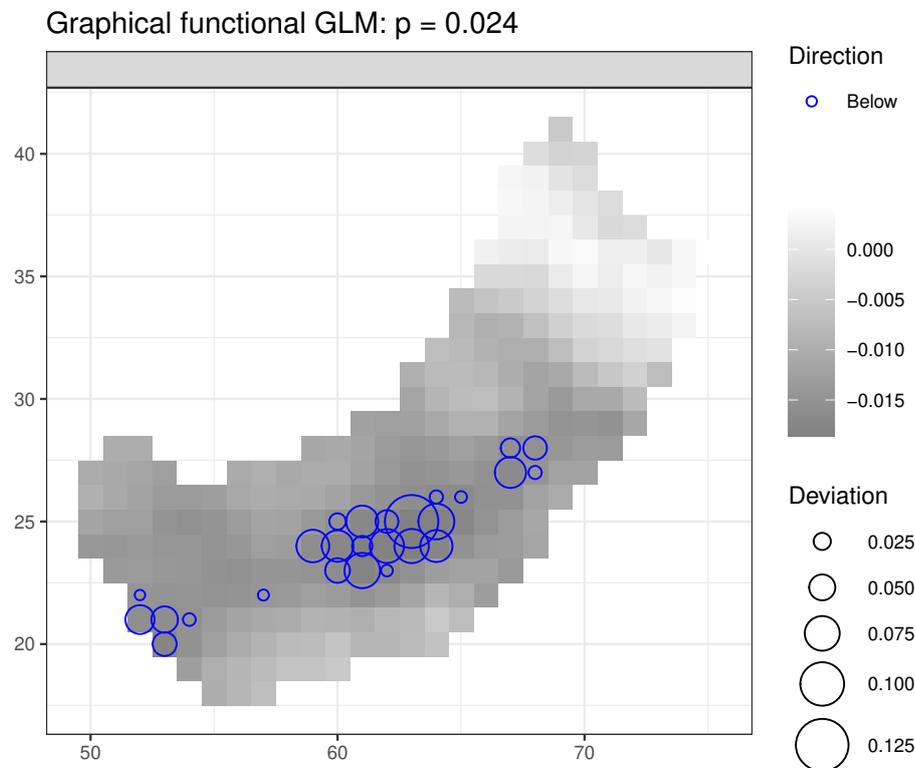


Figure 14: Graphical functional GLM for testing the effect of the group (autism, control) in the brain image example: the observed difference (autism-control) and the locations (blue circles) where the observed coefficient goes below the 95% global envelope (area). The size of the circles are proportional to the size of the deviation of the observed function from the lower bound of the envelope divided by the width of the envelope.

### *F*-rank GLM

In the *F*-rank GLM, the same linear model (Equation 6) is fitted at each  $r \in \{1, \dots, d\}$  and permutations under the null hypothesis are obtained similarly by the Freedman-Lane procedure as in the graphical functional GLM (Freedman and Lane 1983; Mrkvička *et al.* 2022). However, the test statistic is the classical *F* statistic (see, e.g., Winkler *et al.* 2014) which is calculated for the hypothesis that the data follows the simpler reduced model of the two proposed linear models that are nested within each other (given in `formula.full` and `formula.reduced`). The use of the function `frank.flm()` is similar to that of `graph.flm()`:

```
R> res.F <- frank.flm(nsim = 999, formula.full = Y ~ Group + Age + Sex,
+   formula.reduced = Y ~ Age + Sex,
+   curve_sets = list(Y = abide_9002_23[['curve_set']]),
+   factors = abide_9002_23[['factors']], GET.args = list(type = "area"))
```

```
R> plot(res.F, what = c("obs", "hi", "hi.sign"), sign.type = "contour")
```

For illustration of different options, we plot here the observed function (`what = "obs"`), the upper envelope (`what = "hi"`) and the observed function with the significant region (`what = "hi.sign"`). The significant region is here shown by contour lines, as chosen in the argument `sign.type`. Contours do not indicate the size of deviation from the bounds of the global envelope, but contours or colored regions (`sign.type = "col"`) can be preferable if the observed function exceeds the envelope on large parts of the domain.

Figure 15 shows that the  $F$ -rank GLM found significant differences between the groups approximately at the same pixels  $r \in \{1, \dots, d\}$  of the brain image as the graphical functional GLM above. In general, for a factor with more than two groups, the  $F$ -rank GLM is however not able to tell between which specific groups of a discrete factor the differences occur (or which of the groups deviate from the mean). In the case of heteroscedasticity, the weighted least squares test statistics can be used instead (Christensen 2002).

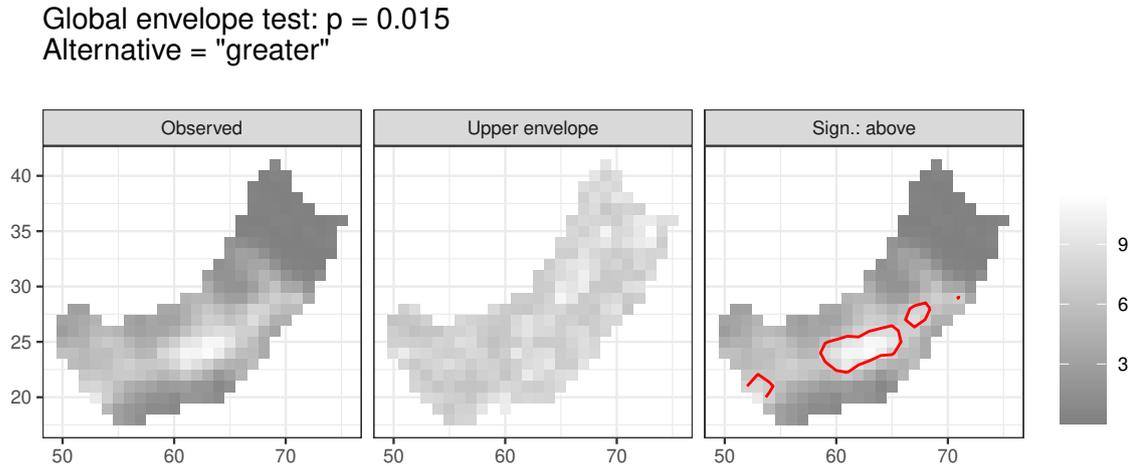


Figure 15:  $F$ -rank functional GLM for testing the effect of the group (autism, control) in the brain image example: the observed  $F$  statistic, the upper bound of the one-sided 95% global envelope (area), and the significant region (red contour lines) where the observed  $F$  statistic exceeds the envelope.

### 3.7. Confidence band in polynomial regression

The bootstrap procedure described in Narisetty and Nair (2016) can be used to compute global confidence bands for the fitted curve in the linear or polynomial regression. In this example, regression data was simulated according to the cubic model  $f(x) = 0.8x - 1.8x^2 + 1.05x^3$  for  $x \in [0, 1]$  with i.i.d. random noise (dots in Figure 16). Then the data was fitted with cubic regression (black solid line in Figure 16) and by permuting the residuals 2000 bootstrap samples were obtained and functions fitted (see more details about the bootstrap procedure in Narisetty and Nair 2016). Finally a 'curve\_set' object was constructed of these bootstrapped

functions by the `create_curve_set()` function and the `central_region()` function was applied to this set to obtain the 95%, 80% and 50% global confidence bands. The multiple global bands were obtained by setting the argument `coverage = c(0.95, 0.80, 0.50)`.

The result of the procedure is shown in Figure 16. The code can be found in the help file of the function `central_region()` in R.

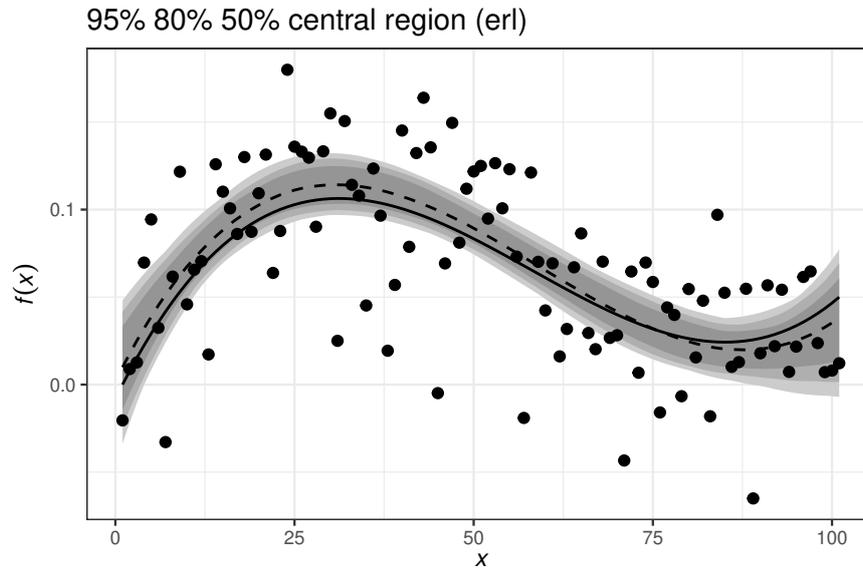


Figure 16: The global 95%, 80% and 50% confidence bands (ERL) for the cubic regression (nested gray bands; widest is 95%), the true function (solid line) from which the data points (dots) were simulated and the median computed from the simulations (dashed line).

A theoretical 95% confidence band can be considered under the given bootstrap scheme. Based on a simulation experiment where the theoretical confidence band was computed from 200000 bootstrapped functions, we observed that the 95% confidence region computed as a convex hull from  $s$  functions converged to the theoretical one from inside for increasing  $s$ . The 95% confidence band computed as the extreme rank envelope from  $s$  functions (see Equation 8) converged to the theoretical one from outside instead. Both these envelopes are finite approximations of the theoretical envelope. On the other hand, in the sense of Barnard's Monte Carlo test (Barnard 1963), the global envelope test (convex hull) is exact for the given set of simulated functions. In the same sense, the confidence band reaches the given global level exactly under the given set of functions.

## 4. Summary and discussion

We presented the **GET** package which was designed for global envelopes that are constructed for a general vector and have IGI (Definition 2.1). The great advantage of these methods is their graphical output, which helps one to interpret the results in the various applications. The package implements different types of global envelopes (see Table 1) and their usage in general and for several specific problems. Because the global envelopes can be used for so many different purposes specified in cases (i)-(iii) in Section 1, there are several other

software, particularly other R packages, that deal with methods that can be used for the same purposes as the methods in **GET**. However, to the best of our knowledge, **GET** is the first package specializing to the global envelopes with IGI.

Besides the graphical interpretation, another advantage of the proposed global rank envelopes is their non-parametric (rank-based) nature, which ensures that the functional or multivariate data coming into the analysis can be inhomogeneous across the domain of their arguments and this phenomenon does not influence the result of the analysis. For example, before the methods discussed in this paper appeared, formal goodness-of-fit testing in spatial statistic was commonly based on the unscaled MAD test (Ripley 1981) or its non-graphical integrated counterpart (Diggle 1979). However, the result of these tests is influenced by unequal variability of the test function across its domain leading in general to loss of power (Myllymäki *et al.* 2015, 2017). A similar situation appears in the permutation GLM tests which, in the functional data analysis or neuroimage analysis (see, e.g., Winkler *et al.* 2014), are commonly based on the  $F$  statistic that standardizes the first and second moments of the data but not the high quantiles. Thus, when the data are inhomogeneous across the domain and non-normal, the commonly used  $F$  max test (which is similar to the unscaled MAD test) is influenced by the inhomogeneous quantiles. The rank-based tests discussed here can then lead to higher power (for details see Mrkvička *et al.* 2022). Similarly, the rank based methods can adjust the shape of the central region to inhomogeneous distribution of the studied functions. Therefore, the global rank envelopes are a valuable tool in all these situations.

A further advantage of the rank tests is that it allows one to give equal weights to the components fed in. Thus, the method is particularly well suited also for multiple testing with several univariate or functional test statistics (Mrkvička *et al.* 2017), for constructing central regions jointly for various transformations of the functions (for details see Dai *et al.* 2020), and for combining various dimensions of multidimensional functions or various functional elements of multivariate functions. Further, it can be used for functional cluster analysis so that various transformations of the functions have the same impact on the clustering algorithm (Dai *et al.* 2022).

Finally, the good properties of the methods presented here are retained also in the case of testing a composite hypothesis: the two-stage Monte Carlo test is applicable to these graphical methods.

We are committed to developing the **GET** package further. For example, new types of global envelopes can be added, if such are invented, and support for specific applications or different type of data will be extended.

## Acknowledgments

MM was financially supported by the Academy of Finland (project numbers 295100, 306875, 327211 and 348154) and TM by the Grant Agency of Czech Republic (Project No. 19-04412S and 24-11096S). The authors thank Jiří Dvořák, Pavel Grabarnik, Ute Hahn, Mikko Kuronen, Michael Rost and Henri Seijo for their contributions and suggestions of code, as well as Stavros Athanasiadis, Wenlin Dai, Marc G. Genton, Milan Jílek, Naveen Naidu Narisetty, Tomáš Roskovec, Samuel Soubeyrand and Ying Sun for their collaboration on developing global envelope methods.

## References

- Aldor-Noiman S, Brown LD, Buja A, Rolke W, Stine RA (2013). “The Power to See: A New Graphical Test of Normality.” *The American Statistician*, **67**(4), 249–260. doi:[10.1080/00031305.2013.847865](https://doi.org/10.1080/00031305.2013.847865).
- Anderson M, Ter Braak C (2003). “Permutation Tests for Multi-Factorial Analysis of Variance.” *Journal of Statistical Computation and Simulation*, **73**(2), 85–113. doi:[10.1080/00949650215733](https://doi.org/10.1080/00949650215733).
- Anderson MJ, Robinson J (2001). “Permutation Tests for Linear Models.” *Australian & New Zealand Journal of Statistics*, **43**(1), 75–88. doi:[10.1111/1467-842X.00156](https://doi.org/10.1111/1467-842X.00156).
- Baddeley A, Hardegen A, Lawrence T, Milne RK, Nair G, Rakshit S (2017). “On Two-Stage Monte Carlo Tests of Composite Hypotheses.” *Computational Statistics & Data Analysis*, **114**, 75–87. doi:<https://doi.org/10.1016/j.csda.2017.04.003>.
- Baddeley A, Rubak E, Turner R (2015). *Spatial Point Patterns: Methodology and Applications with R*. Chapman and Hall/CRC Press, London.
- Barnard GA (1963). “Discussion on ‘The Spectral Analysis of Point Processes’ (by M. S. Bartlett).” *Journal of the Royal Statistical Society B*, **25**, 294.
- Besag JE (1977). “Comment on ‘Modelling Spatial Patterns’ by B. D. Ripley.” *Journal of the Royal Statistical Society B*, **39**(2), 193–195.
- Bolin D, Lindgren F (2015). “Excursion and Contour Uncertainty Regions for Latent Gaussian Models.” *Journal of the Royal Statistical Society B*, **77**(1), 85–106. doi:[10.1111/rssb.12055](https://doi.org/10.1111/rssb.12055).
- Bolin D, Lindgren F (2017). “Quantifying the Uncertainty of Contour Maps.” *Journal of Computational and Graphical Statistics*, **26**(3), 513–524. doi:[10.1080/10618600.2016.1228537](https://doi.org/10.1080/10618600.2016.1228537).
- Bolin D, Lindgren F (2018). “Calculating Probabilistic Excursion Sets and Related Quantities Using excursions.” *Journal of Statistical Software, Articles*, **86**(5), 1–20. doi:[10.18637/jss.v086.i05](https://doi.org/10.18637/jss.v086.i05).
- Canty A, Ripley B (2017). **boot: Bootstrap R (S-PLUS) Functions**. R package version 1.3-20, URL <https://cran.r-project.org/package=boot>.
- Chaiban C, Biscio C, Thanapongtharm W, Tildesley M, Xiao X, Robinson TP, Vanwambeke SO, Gilbert M (2019). “Point Pattern Simulation Modelling of Extensive and Intensive Chicken Farming in Thailand: Accounting for Clustering and Landscape Characteristics.” *Agricultural Systems*, **173**, 335–344. doi:[10.1016/j.agsy.2019.03.004](https://doi.org/10.1016/j.agsy.2019.03.004).
- Christensen R (2002). *Plane Answers to Complex Questions: The Theory of Linear Models*. Springer-Verlag, New York.
- Conde IC, de Una Alvarez J (2016). **sgof: Multiple Hypothesis Testing**. R package version 2.3, URL <https://CRAN.R-project.org/package=sgof>.

- Dai W, Athanasiadis S, Mrkvička T (2022). “A New Functional Clustering Method with Combined Dissimilarity Sources and Graphical Interpretation.” In R López-Ruiz (ed.), *Computational Statistics and Applications*. IntechOpen. ISBN 978-1-83969-782-1 978-1-83969-783-8. doi:10.5772/intechopen.100124.
- Dai W, Mrkvička Ts, Sun Y, Genton MG (2020). “Functional Outlier Detection and Taxonomy by Sequential Transformations.” *Computational Statistics & Data Analysis*, **149**, 106960. doi:10.1016/j.csda.2020.106960.
- Dao NA, Genton MG (2014). “A Monte Carlo Adjusted Goodness-Of-Fit Test for Parametric Models Describing Spatial Point Patterns.” *Journal of Computational and Graphical Statistics*, **23**, 497–517. doi:10.1080/10618600.2012.760459.
- Davison AC, Hinkley DV (1997). *Bootstrap Methods and Their Application*. Cambridge University Press, Cambridge.
- Degras D (2016). **SCBmeanfd**: *Simultaneous Confidence Bands for the Mean of Functional Data*. R package version 1.2.2, URL <https://CRAN.R-project.org/package=SCBmeanfd>.
- Di Martino A, Yan C, Li Q, Denio E, Castellanos F, Alaerts K, Anderson J, Assaf M, Bookheimer S, Dapretto M, et al (2014). “The Autism Brain Imaging Data Exchange: Towards a Large-Scale Evaluation of the Intrinsic Brain Architecture in Autism.” *Molecular Psychiatry*, **19**, 659–667.
- Diggle PJ (1979). “On Parameter Estimation and Goodness-of-Fit Testing for Spatial Point Patterns.” *Biometrics*, **35**(1), 87–101. doi:10.2307/2529938.
- Diggle PJ (2013). *Statistical Analysis of Spatial and Spatio-Temporal Point Patterns*. 3rd edition. CRC Press, Boca Raton. ISBN 978-0-340-74070-5.
- Duranton G, Overman HG (2005). “Testing for Localization Using Micro-Geographic Data.” *Review of Economic Studies*, **72**(4), 1077–1106.
- Dvořák J, Mrkvička Ts (2022). “Graphical Tests of Independence for General Distributions.” *Computational Statistics*, **37**, 671–699. doi:10.1007/s00180-021-01134-y.
- Febrero M, Galeano P, González-Manteiga W (2008). “Outlier Detection in Functional Data by Depth Measures, with Application to Identify Abnormal NOx Levels.” *Environmetrics*, **19**(4), 331–345. doi:10.1002/env.878.
- Febrero-Bande M, Oviedo de la Fuente M (2012). “Statistical Computing in Functional Data Analysis: The R Package **fdA.usc**.” *Journal of Statistical Software, Articles*, **51**(4), 1–28. doi:10.18637/jss.v051.i04.
- Freedman D, Lane D (1983). “A Nonstochastic Interpretation of Reported Significance Levels.” *Journal of Business and Economic Statistics*, **1**(4), 292–298. doi:10.2307/1391660.
- Gorecki T, Smaga L (2017). **fdANOVA**: *Analysis of Variance for Univariate and Multivariate Functional Data*. R package version 0.1.0, URL <https://CRAN.R-project.org/package=fdANOVA>.

- Grabarnik P, Chiu SN (2002). “Goodness-of-Fit Test for Complete Spatial Randomness Against Mixtures of Regular and Clustered Spatial Point Processes.” *Biometrika*, **89**(2), 411–421.
- Häbel H, Rajala T, Marucci M, Boissier C, Schladitz K, Redenbach C, Särkkä A (2017). “A Three-Dimensional Anisotropic Point Process Characterization for Pharmaceutical Coatings.” *Spatial Statistics*, **22**, 306–320. doi:10.1016/j.spasta.2017.05.003.
- Hahn U (2015). “A Note on Simultaneous Monte Carlo Tests.” *Technical report*, Centre for Stochastic Geometry and advanced Bioimaging, Aarhus University.
- Hothorn T, Hornik K, van de Wiel M, Zeileis A (2008). “Implementing a Class of Permutation Tests: The **coin** Package.” *Journal of Statistical Software, Articles*, **28**(8), 1–23. doi:10.18637/jss.v028.i08.
- Hothorn T, Hornik K, van de Wiel MA, Zeileis A (2006). “A Lego System for Conditional Inference.” *The American Statistician*, **60**(3), 257–263. doi:10.1198/000313006X118430.
- Illian J, Penttinen A, Stoyan H, Stoyan D (2008). *Statistical Analysis and Modelling of Spatial Point Patterns*. Statistics in Practice, 1st edition. John Wiley & Sons, Chichester. ISBN 978-0-470-01491-2.
- López-Pintado S, Romo J (2009). “On the Concept of Depth for Functional Data.” *Journal of the American Statistical Association*, **104**(486), 718–734. doi:10.1198/jasa.2009.0108.
- López-Pintado S, Torrente A (2013). **depthTools**: *Depth Tools Package*. R package version 0.4, URL <https://CRAN.R-project.org/package=depthTools>.
- Marcon E, Traissac S, Puech F, Lang G (2015). “Tools to Characterize Point Patterns: **dbmss** for R.” *Journal of Statistical Software, Code Snippets*, **67**(3), 1–15. doi:10.18637/jss.v067.c03.
- Mrkvička T, Myllymäki M, Hahn U (2017). “Multiple Monte Carlo Testing, with Applications in Spatial Point Processes.” *Statistics and Computing*, **27**(5), 1239–1255. doi:10.1007/s11222-016-9683-9.
- Mrkvička T, Roskovec T, Rost M (2021). “A Nonparametric Graphical Tests of Significance in Functional GLM.” *Methodology and Computing in Applied Probability*, **23**, 593–612. doi:10.1007/s11009-019-09756-y.
- Mrkvička T, Myllymäki M, Jílek M, Hahn U (2020). “A One-Way ANOVA Test for Functional Data with Graphical Interpretation.” *Kybernetika*, **56**(3), 432–458. doi:10.14736/kyb-2020-3-0432.
- Mrkvička T, Myllymäki M, Kuronen M, Narisetty NN (2022). “New Methods for Multiple Testing in Permutation Inference for the General Linear Model.” *Statistics in Medicine*, **41**(2), 276–297. doi:10.1002/sim.9236.
- Murrell DJ (2018). “A Global Envelope Test to Detect Non-Random Bursts of Trait Evolution.” *Methods in Ecology and Evolution*, **9**(7), 1739–1748. doi:10.1111/2041-210X.13006.

- Myllymäki M, Grabarnik P, Seijo H, Stoyan D (2015). “Deviation Test Construction and Power Comparison for Marked Spatial Point Patterns.” *Spatial Statistics*, **11**, 19–34. doi: [10.1016/j.spasta.2014.11.004](https://doi.org/10.1016/j.spasta.2014.11.004).
- Myllymäki M, Kuronen M, Mrkvička T (2020). “Testing Global and Local Dependence of Point Patterns on Covariates in Parametric Models.” *Spatial Statistics*, **42**, 100436. ISSN 2211-6753. doi: [10.1016/j.spasta.2020.100436](https://doi.org/10.1016/j.spasta.2020.100436).
- Myllymäki M, Mrkvička T (2020). “Comparison of Non-parametric Global Envelopes.” arXiv:2008.09650 [stat.ME]. URL <https://arxiv.org/abs/2008.09650>.
- Myllymäki M, Mrkvička T, Grabarnik P, Seijo H, Hahn U (2017). “Global Envelope Tests for Spatial Processes.” *Journal of the Royal Statistical Society B*, **79**, 381–404. doi: [10.1111/rssb.12172](https://doi.org/10.1111/rssb.12172).
- Nagy S, Gijbels I, Hlubinka D (2017). “Depth-Based Recognition of Shape Outlying Functions.” *Journal of Computational and Graphical Statistics*, **26**(4), 883–893. doi: [10.1080/10618600.2017.1336445](https://doi.org/10.1080/10618600.2017.1336445).
- Narisetty NN, Nair VJ (2016). “Extremal Depth for Functional Data and Applications.” *Journal of the American Statistical Association*, **111**(516), 1705–1714. doi: [10.1080/01621459.2015.1110033](https://doi.org/10.1080/01621459.2015.1110033).
- Pedersen TL (2020). **patchwork**: *The Composer of Plots*. R package version 1.1.0, URL <https://CRAN.R-project.org/package=patchwork>.
- Pollard K, Dudoit S, van der Laan M (2005). “Multiple Testing Procedures: The **multtest** Package and Applications to Genomics.” In R Gentleman, V Carey, W Huber, R Irizarry, S Dudoit (eds.), *Bioinformatics and Computational Biology Solutions Using R and Bioconductor. Statistics for Biology and Health*, pp. 249–271. Springer-Verlag, New York.
- Pollington TM, Tildesley MJ, Hollingsworth TD, Chapman LA (2020). “Developments in Statistical Inference When Assessing Spatiotemporal Disease Clustering with the Tau Statistic.” *Spatial Statistics*, p. 100438. ISSN 2211-6753. doi: [10.1016/j.spasta.2020.100438](https://doi.org/10.1016/j.spasta.2020.100438).
- R Core Team (2020). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL <https://www.R-project.org/>.
- Ramsay JO, Graves S, Hooker G (2022). **fda**: *Functional Data Analysis*. R package version 6.0.5, URL <https://CRAN.R-project.org/package=fda>.
- Ramsay JO, Silverman BW (2005). *Functional Data Analysis*. Springer Series in Statistics, 2nd edition. Springer-Verlag, New York. ISBN 978-0387-40080-8.
- Ripley BD (1977). “Modelling Spatial Patterns.” *Journal of the Royal Statistical Society B*, **39**(2), 172–212.
- Ripley B D (1981). *Spatial Statistics*. Wiley Series in Probability and Mathematical Statistics. John Wiley & Sons, New Jersey. ISBN 978-0-471-08367-2.
- Stoyan D (2016). “Point Process Statistics: Application to Modern and Contemporary Art and Design.” *Journal of Mathematics and the Arts*, **10**(1-4), 20–34. doi: [10.1080/17513472.2016.1208980](https://doi.org/10.1080/17513472.2016.1208980).

- Sun Y, Genton MG (2011). “Functional Boxplots.” *Journal of Computational and Graphical Statistics*, **20**(2), 316–334. doi:10.1198/jcgs.2011.09224.
- Sun Y, Genton MG, Nychka DW (2012). “Exact Fast Computation of Band Depth for Large Functional Datasets: How Quickly Can One Million Curves Be Ranked?” *Stat*, **1**(1), 68–74. doi:10.1002/sta4.8.
- Torrente A, López-Pintado S, Romo J (2013). “**DepthTools**: an R Package for a Robust Analysis of Gene Expression Data.” *BMC Bioinformatics*, **14**(237). doi:10.1186/1471-2105-14-237.
- Tzourio-Mazoyer N, Landeau B, Papathanassiou D, Crivello F, Etard O, Delcroix N, Mazoyer B, Joliot M (2002). “Automated Anatomical Labeling of Activations in SPM Using a Macroscopic Anatomical Parcellation of the MNI MRI Single-Subject Brain.” *NeuroImage*, **15**, 273–289.
- van Lieshout MC (2010). “Spatial Point Process Theory.” In AE Gelfand, PJ Diggle, M Fuentes, P Guttorp (eds.), *Handbook of Spatial Statistics*, Handbooks of Modern Statistical Methods, 1st edition. CRC Press. ISBN 978-1-4200-7287-7.
- Wickham H (2016). *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag, New York. ISBN 978-3-319-24277-4.
- Wiesenfarth M, Krivobokova T, Klasen S, Sperlich S (2012). “Direct Simultaneous Inference in Additive Models and Its Application to Model Undernutrition.” *Journal of the American Statistical Association*, **107**(500), 1286–1296. doi:10.1080/01621459.2012.682809.
- Winkler AM, Ridgway GR, Webster MA, Smith SM, Nichols TE (2014). “Permutation Inference for the General Linear Model.” *NeuroImage*, **92**, 381–397. doi:10.1016/j.neuroimage.2014.01.060.
- Zou QH, Zhu CZ, Yang Y, Zuo XN, Long XY, Cao QJ, Wang YF, Zang YF (2008). “An Improved Approach to Detection of Amplitude of Low-Frequency Fluctuation (ALFF) for Resting-State fMRI: Fractional ALFF.” *Journal of Neuroscience Methods*, **172**, 137–141.

## A. Definitions of global envelopes

Here, we provide definitions of the different global envelopes and the corresponding measures, as outlined in Table 1. When working with functions, it is assumed that they have already been discretized as required in practice. Thus, we consider general vectors  $\mathbf{T}_i = (T_{i1}, \dots, T_{id})$ ,  $i = 1, \dots, s$ , for which a global envelope is to be constructed.

The measures satisfy the IGI property with probability 1 if there are no pointwise ties, meaning that there are no ties in  $T_{1k}, \dots, T_{sk}$  for every  $k = 1, \dots, d$ . This holds if  $\mathbf{T}_i$  are realizations of an absolutely continuous random vector. If pointwise ties do occur, the IGI property may be violated in elements of the vector where ties are present. For the measures based on pointwise ranks (Sections A.1–A.4), we use the pointwise mid-rank for tied values to weigh down the influence of elements ( $k$ ) where many  $T_{ik}$ ,  $i = 1, \dots, s$ , coincide.

### A.1. Global rank envelope

The extreme rank  $R_i$  of the vector  $\mathbf{T}_i$  is defined as the minimum of its pointwise ranks, namely

$$R_i = \min_{k=1,\dots,d} R_{ik}, \quad (7)$$

where the pointwise rank  $R_{ik}$  is the rank of the element  $T_{ik}$  among the corresponding elements  $T_{1k}, T_{2k}, \dots, T_{sk}$  of the  $s$  vectors such that the lowest ranks correspond to the most extreme values of the statistics. How the pointwise ranks are determined depends on whether a one-sided or a two-sided global envelope (test) is to be constructed: Let  $r_{1k}, r_{2k}, \dots, r_{sk}$  be the raw ranks of  $T_{1k}, T_{2k}, \dots, T_{sk}$ , such that the smallest  $T_{ik}$  has rank 1. In the case of ties among  $T_{1k}, T_{2k}, \dots, T_{sk}$ , the raw ranks are averaged. The pointwise ranks are then calculated as

$$R_{ik} = \begin{cases} r_{ik}, & \text{for the one-sided case 'less',} \\ s + 1 - r_{ik}, & \text{for the one-sided case 'greater',} \\ \min(r_{ik}, s + 1 - r_{ik}), & \text{for the two-sided case.} \end{cases}$$

The cases 'less' and 'greater' are most often relevant in a hypothesis testing case (ii) (see Section 1). The case 'less' corresponds to the alternative that the values of interesting vector ( $\mathbf{T}_1$ ) are smaller than under the null hypothesis. The case 'greater' corresponds to the alternative that the values are larger than under the null hypothesis. The extreme rank measure  $R_i$  induces an ordering of  $\mathbf{T}_i$ ,  $i = 1, \dots, s$ , which can be used to detect the extremeness of the vectors among each other. Given that  $\mathbf{T}_1$  is the observed vector in the Monte Carlo or permutation test, the (conservative)  $p$  value of the test is equal to  $p_+ = \sum_{i=1}^s \mathbf{1}(R_i \leq R_1)/s$ . Since the extreme ranks can have many ties, the test is also equipped with the liberal  $p$  value,  $p_- = \sum_{i=1}^s \mathbf{1}(R_i < R_1)/s$ . Then, when  $\alpha$  falls inside the  $p$  interval  $(p_-, p_+]$ , the decision of the test is not defined.

The  $100(1 - \alpha)\%$  global rank envelope induced by this measure is defined through

$$\mathbf{T}_{\text{low } k}^{\text{local}, l} = \min_{i=1,\dots,s}^l T_{ik} \quad \text{and} \quad \mathbf{T}_{\text{upp } k}^{\text{local}, l} = \max_{i=1,\dots,s}^l T_{ik} \quad \text{for } k = 1, \dots, d, \quad (8)$$

by taking  $l = R_{(\alpha)}$ , according to the point 1. of IGI (see Definition 2.1), i.e., setting  $\mathbf{T}_{\text{low } k}^{(\alpha)} = \mathbf{T}_{\text{low } k}^{\text{local}, R_{(\alpha)}}$  and  $\mathbf{T}_{\text{upp } k}^{(\alpha)} = \mathbf{T}_{\text{upp } k}^{\text{local}, R_{(\alpha)}}$ . Here  $\min^l$  and  $\max^l$  denote the  $l$ -th smallest and largest values, respectively, and  $l = 1, 2, \dots, \lfloor s/2 \rfloor$ . If  $\mathbf{T}_i$  is strictly outside the envelope for some  $k = 1, \dots, d$ , then also  $R_i < R_{(\alpha)}$ , and if  $\mathbf{T}_{\text{low } k}^{(\alpha)} < \mathbf{T}_{ik} < \mathbf{T}_{\text{upp } k}^{(\alpha)}$  for all  $k = 1, \dots, d$ , then  $R_i > R_{(\alpha)}$ . However, if  $\mathbf{T}_i$  coincides either with  $\mathbf{T}_{\text{low } k}^{(\alpha)}$  or  $\mathbf{T}_{\text{upp } k}^{(\alpha)}$  for some  $k = 1, \dots, d$ , then  $R_i = R_{(\alpha)}$ , and  $\alpha \in (p_-, p_+]$  in the testing case.

Because the extreme rank can achieve many ties (see, e.g., Mrkvička *et al.* 2022), it is necessary to use a relatively large  $s$  for the global rank envelope. The following three refinements of the extreme rank solve the ties problem and enable the use of a smaller  $s$ .

## A.2. Global extreme rank length (ERL) envelope

The extreme rank length (ERL) measure (Myllymäki *et al.* 2017; Narisetty and Nair 2016) refines the extreme rank measure by breaking the ties in the extreme ranks  $R_i$  by taking into account also the number of  $R_{ik}$  which are equal to  $R_i$ . Further, the numbers of ranks equal to  $R_i + 1$ ,  $R_i + 2$ , ... are used to break any remaining ties.

Formally, the ERL measure of  $\mathbf{T}_i$  is defined based on the vector of the pointwise ordered ranks  $\mathbf{R}_i = (R_{i[1]}, R_{i[2]}, \dots, R_{i[d]})$ , where the ranks are arranged from smallest to largest, i.e.,  $R_{i[k]} \leq R_{i[k']}$  whenever  $k \leq k'$ . While the extreme rank (Equation 7) corresponds to  $R_i = R_{i[1]}$ , the ERL measure takes all these ranks into account by the reverse lexical ordering. The ERL measure of  $\mathbf{T}_i$  is

$$E_i = \frac{1}{s} \sum_{i'=1}^s \mathbf{1}(\mathbf{R}_{i'} \prec \mathbf{R}_i) \quad (9)$$

where

$$\mathbf{R}_{i'} \prec \mathbf{R}_i \iff \exists n \leq d : R_{i'[k]} = R_{i[k]} \forall k < n, R_{i'[n]} < R_{i[n]}.$$

The division by  $s$  leads to normalized ranks that obtain values between 0 and 1. Consequently, the ERL measure corresponds to the extremal depth of Narisetty and Nair (2016).

The probability of having a tie in the ERL measure is rather small, thus practically the ERL solves the ties problem. The final  $p$  value of a Monte Carlo test is  $p_{\text{erl}} = \sum_{i=1}^s \mathbf{1}(E_i \leq E_1) / s$ . Let  $E_{(\alpha)}$  be defined according to the point 1. of IGI and  $I_\alpha = \{i \in 1, \dots, s : E_i \geq E_{(\alpha)}\}$  be the index set of vectors less or as extreme as  $E_{(\alpha)}$ . Then the  $100(1 - \alpha)\%$  global ERL envelope induced by  $E_i$  is

$$\mathbf{T}_{\text{low } k}^{(\alpha)} = \min_{i \in I_\alpha} T_{ik} \quad \text{and} \quad \mathbf{T}_{\text{upp } k}^{(\alpha)} = \max_{i \in I_\alpha} T_{ik} \quad \text{for } k = 1, \dots, d, \quad (10)$$

see Narisetty and Nair (2016) and Mrkvička *et al.* (2020).

### A.3. Global continuous rank envelope

The ties can alternatively be broken by the continuous rank measure (Hahn 2015; Mrkvička *et al.* 2022) which refines the extreme rank measure by considering instead of the (discrete) pointwise ranks  $R_{ik}$  continuous pointwise ranks  $C_{ik}$  defined by the ratios of  $T_{ik}$  to the closest other  $T_{jk}$ ,  $j = 1, \dots, s$ ,  $j \neq i$ .

Formally, the continuous rank measure is

$$C_i = \frac{1}{s} \min_{k=1, \dots, d} C_{ik},$$

where  $s$  scales the values to interval from 0 to 1. The definition of pointwise continuous ranks  $C_{ik}$  depends again on whether a one-sided or two-sided global envelope (test) is to be constructed:

$$C_{ik} = \begin{cases} c_{ik}, & \text{for the one-sided case 'less'} \\ s - c_{ik}, & \text{for the one-sided case 'greater'} \\ \min(c_{ik}, s - c_{ik}), & \text{for the two-sided case.} \end{cases}$$

where  $c_{ik}$  is the raw continuous rank of  $T_{ik}$  among  $T_{1k}, \dots, T_{sk}$  according to Definition A.1 and the three cases are similar to those of  $R_{ik}$  above.

**Definition A.1** Let  $y_{[1]} \leq y_{[2]} \leq \dots \leq y_{[s]}$  denote the ordered set of values  $y_i$ ,  $i = 1, 2, \dots, s$ . Define the raw continuous rank such that the smallest  $y_i$  has smallest rank following Mrkvička *et al.* (2022):

$$c_{[j]} = j - 1 + \frac{y_{[j]} - y_{[j-1]}}{y_{[j+1]} - y_{[j-1]}}$$

for  $j = 2, 3, \dots, s - 1$  and

$$c_{[1]} = \exp\left(-\frac{y_{[2]} - y_{[1]}}{y_{[s]} - y_{[2]}}\right), \quad c_{[s]} = s - \exp\left(-\frac{y_{[s]} - y_{[s-1]}}{y_{[s-1]} - y_{[1]}}\right).$$

If there are ties,  $y_{[i-1]} < y_{[i]} = \dots = y_{[j]} < y_{[j+1]}$ , then the raw continuous rank is defined as  $c_{[k]} = \frac{i+j}{2} - \frac{1}{2}$  for  $k = i, i + 1, \dots, j$ .

The  $p$  value of the univariate Monte Carlo test is  $p_{\text{cont}} = \sum_{i=1}^s \mathbf{1}(C_i \leq C_1)/s$ . The  $100(1-\alpha)\%$  global continuous rank envelope induced by  $C_i$  is constructed in the same way as global ERL envelope, i.e., as a hull of  $\mathbf{T}_i$  for which  $C_i \geq C_{(\alpha)}$ , where  $C_{(\alpha)}$  is defined according to the point 1. of IGI. This is achieved by having  $I_\alpha = \{i \in 1, \dots, s : C_i \geq C_{(\alpha)}\}$  in Equation 10.

#### A.4. Global area rank envelope

Another refinement of rank envelope is the area rank measure (Mrkvička *et al.* 2022),

$$A_i = \frac{1}{s} \left( R_i - \frac{1}{d} \sum_{k=1}^d (R_i - C_{ik}) \mathbf{1}(C_{ik} < R_i) \right).$$

The area measure breaks the ties in the extreme ranks by the sum (area) of the differences between the extreme rank  $R_i$  and the pointwise continuous rank  $C_{ik}$  from those  $k = 1, \dots, d$  where the continuous rank is smaller than the extreme rank. The univariate Monte Carlo test is performed based on  $A_i$  with  $p_{\text{area}} = \sum_{i=1}^s \mathbf{1}(A_i \leq A_1)/s$ . The  $100(1-\alpha)\%$  global area rank envelope induced by  $A_i$  is constructed similarly as the global ERL and continuous rank envelopes, with  $I_\alpha = \{i \in 1, \dots, s : A_i \geq A_{(\alpha)}\}$  in Equation 10.

#### A.5. Global directional quantile, studentized and unscaled envelope

The above four global envelopes are based on the whole distributions of  $T_{1k}, \dots, T_{sk}$ ,  $k = 1, \dots, d$ . It is also possible to approximate the distribution by a few sample characteristics. The sample characteristics are in the package **GET** estimated from  $T_{ik}$ ,  $i = 1, \dots, s$ , for each  $k$ .

The *global directional quantile envelope* uses the expectation  $T_{0k}$ ,  $\beta\%$  upper quantile  $\bar{T}_{.k}$  and  $\beta\%$  lower quantile  $\underline{T}_{.k}$  to approximate the distributions. Setting  $\beta = 2.5$  was used in Myllymäki *et al.* (2017); setting  $\beta = 25$  can also be useful especially for defining the 50% central region from a low number of functions. Note that  $\beta$  has to be greater than  $100/s$  in order to be able to estimate the  $\beta$  and  $1 - \beta$  quantiles. The directional quantile measure (Myllymäki *et al.* 2015, 2017)  $D_i$  is defined as

$$D_i = \max_k \left( \mathbf{1}(T_{ik} \geq T_{0k}) \frac{T_{ik} - T_{0k}}{|\bar{T}_{.k} - T_{0k}|} + \mathbf{1}(T_{ik} < T_{0k}) \frac{T_{0k} - T_{ik}}{|\underline{T}_{.k} - T_{0k}|} \right), \quad (11)$$

From historical reasons,  $D_i$  is defined to be bigger for more extreme vectors. The same holds for the following two measures. The univariate Monte Carlo test is performed based on  $D_i$  with  $p_{\text{qdir}} = \sum_{i=1}^s \mathbf{1}(D_i \geq D_1)/s$ , and the  $100(1-\alpha)\%$  global directional quantile envelope induced by  $D_i$  is defined by

$$\mathbf{T}_{\text{low } k}^{(\alpha)} = T_{0k} - D_{(\alpha)} |\underline{T}_{.k} - T_{0k}| \quad \text{and} \quad \mathbf{T}_{\text{upp } k}^{(\alpha)} = T_{0k} + D_{(\alpha)} |\bar{T}_{.k} - T_{0k}| \quad \text{for } k = 1, \dots, d, \quad (12)$$

where  $D_{(\alpha)}$  is taken according to the point 1. of IGI.

The *global studentized envelope* approximates the distribution instead by the expectation  $T_{0k}$  and the standard deviation  $\text{sd}(T_{\cdot k})$ . The studentized measure (Myllymäki *et al.* 2015, 2017) is

$$S_i = \max_k \left| \frac{T_{ik} - T_{0k}}{\text{sd}(T_{\cdot k})} \right|, \quad (13)$$

and the univariate Monte Carlo test is performed based on  $s_i$  with  $p_{\text{st}} = \sum_{i=1}^s \mathbf{1}(S_i \geq S_1)/s$ . The  $100(1 - \alpha)\%$  global studentized envelope induced by  $S_i$  is defined by

$$\mathbf{T}_{\text{low } k}^{(\alpha)} = T_{0k} - S_{(\alpha)} \text{sd}(T_{\cdot k}) \quad \text{and} \quad \mathbf{T}_{\text{upp } k}^{(\alpha)} = T_{0k} + S_{(\alpha)} \text{sd}(T_{\cdot k}) \quad \text{for } k = 1, \dots, d, \quad (14)$$

where  $S_{(\alpha)}$  is taken according to the point 1. of IGI.

The *global unscaled envelope* considered for the sake of completeness has its origin in the classical Kolmogorov-Smirnov statistic. The unscaled measure  $U_i$  can be defined as  $U_i = \max_k |T_{ik} - T_{0k}|$ , the univariate Monte Carlo test performed based on  $U_i$  has the  $p$  value  $p_{\text{unsc}} = \sum_{i=1}^s \mathbf{1}(U_i \geq U_1)/s$ , and the  $100(1 - \alpha)\%$  global unscaled envelope induced by  $u_i$  is given by

$$\mathbf{T}_{\text{low } k}^{(\alpha)} = T_{0k} - U_{(\alpha)} \quad \text{and} \quad \mathbf{T}_{\text{upp } k}^{(\alpha)} = T_{0k} + U_{(\alpha)} \quad \text{for } k = 1, \dots, d, \quad (15)$$

where  $U_{(\alpha)}$  is taken according to the point 1. of IGI. A problem of this envelope is that its width is the same along the whole domain, thus it cannot account for the changes in the variability of the distributions  $\mathbf{T}_i$  across different values of  $k$  (Myllymäki *et al.* 2015, 2017).

## B. Combined global envelopes

Assume that there are  $G$  vectors  $\mathbf{T}_i^j = (T_{i1}^j, \dots, T_{id_j}^j)$ ,  $j = 1, \dots, G$ ,  $i = 1, \dots, s$ ,  $d_j \geq 1$ , and the construction of a global envelope is wanted jointly for all of them. A combined global envelope test can be made in two different ways.

In the *two-step combining procedure*, first, a measure is chosen for each  $j = 1, \dots, G$  and computed for the vectors  $\mathbf{T}_i^j$ ,  $i = 1, \dots, s$  and  $j = 1, \dots, G$ . Let the resulting measures be  $m_i^j$ . As the second step, the one-sided extreme rank length is applied to the new vector  $\mathbf{T}'_i = (m_i^1, m_i^2, \dots, m_i^G)$  of the measures. As a result, a joint sorting of vectors  $\mathbf{T}'_1, \dots, \mathbf{T}'_s$ ,  $i = 1, \dots, s$ , is obtained and a joint extreme rank length measure  $E_i$  is attached to every  $i = 1, \dots, s$ . The  $p$  value of the combined Monte Carlo test is  $p_{\text{erl}} = \sum_{i=1}^s \mathbf{1}(E_i \leq E_1)/s$ , and the common  $100(1 - \alpha)\%$  global envelope is constructed similarly as the  $100(1 - \alpha)\%$  global extreme rank length envelope (Equation 10): Let  $E_{(\alpha)}$  be defined according to the point 1. of IGI and  $I_\alpha = \{i \in 1, \dots, s : E_i \geq E_{(\alpha)}\}$  be the index set of vectors less or as extreme as  $E_{(\alpha)}$ . Then the common  $100(1 - \alpha)\%$  global envelope is

$$\mathbf{T}_{\text{low } k}^{(\alpha),j} = \min_{i \in I_\alpha} T_{ik}^j \quad \text{and} \quad \mathbf{T}_{\text{upp } k}^{(\alpha),j} = \max_{i \in I_\alpha} T_{ik}^j \quad \text{for } k = 1, \dots, d_j, j = 1, \dots, G. \quad (16)$$

The extreme rank length measure is chosen in the second step because it gives the same weight to every component (even when  $d_j, j = 1, \dots, G$ , are different or even if different measures are used in the first step), it is based on ranks only and it achieves almost no ties.

In cases where  $d_1 = \dots = d_G (= d)$ , it is also possible to use a simple *one-step combining procedure*. Then the global envelope (any of those in Table 1) is constructed for the long

vectors

$$\mathbf{T}_i = (T_{i1}^1, \dots, T_{id}^1, T_{i1}^2, \dots, T_{id}^2, \dots, T_{i1}^G, \dots, T_{id}^G), \quad i = 1, \dots, s.$$

The one-step combining can be used for example when multivariate functional data  $(\mathcal{T}_{i1}, \dots, \mathcal{T}_{id})$ ,  $i = 1, \dots, s$ , where  $\mathcal{T}_{ik} = (T_{ik}^1, \dots, T_{ik}^J)$  are multivariate vectors of  $J$  elements, are investigated. Then it is possible to separate the dimensions into a set of  $J$  marginal vectors applying the one-step combining procedure, i.e., to take  $\mathbf{T}_i = (T_{i1}^1, \dots, T_{id}^1, \dots, T_{i1}^J, \dots, T_{id}^J)$ . Further, it is also possible to add other vectors expressing the correlation between the elements of the vectors, e.g., if we have a two-dimensional functional data, the vector  $(T_{i1}^1 T_{i1}^2 - T_{01}^1 T_{01}^2, \dots, T_{id}^1 T_{id}^2 - T_{0d}^1 T_{0d}^2)$  can be added into  $\mathbf{T}_i$  behind the marginal vectors. Here  $T_{0k}^j$  denotes the expectation of  $T_{ik}^j$ ,  $i = 1, \dots, s$ .

The graphical functional ANOVA and GLM (see the functions in Table 2) use the one-step combining procedure to merge the mean or contrast vectors under inspection, because in this case all the vectors have the same structure (see Sections 3.5 and 3.6 and Mrkvička *et al.* 2020; Mrkvička *et al.* 2021). On the other hand, for generality, the default combining procedure of global envelope construction functions in **GET** is the two-step procedure, which is presented for the first time here as an improvement of the combined tests of Mrkvička *et al.* (2017) (see an example in Section 3.1). The combined envelopes are implemented in the `central_region()` and `global_envelope_test()` functions as mentioned above, and the one- or two-step procedure can be specified in the argument `nstep` (either 1 or 2).

## C. Adjusted global envelopes for composite null hypotheses

The Monte Carlo tests for which the global envelopes are constructed are exact only in the case when the null hypothesis is simple, i.e., when no parameters have to be estimated. This is the case in permutation tests of task (iii), but in task (ii) the null hypothesis can often be composite, i.e., some parameters of the null model have to be estimated. In such a composite case, the classical Monte Carlo test can be liberal or conservative. This problem can be solved by a two-stage procedure, where in the first stage the level of the test is estimated. Such a procedure was first introduced by Dao and Genton (2014) for Monte Carlo tests. Myllymäki *et al.* (2017) extended this adjusted method for global envelopes. Baddeley *et al.* (2017) improved the procedure further in order to obtain an exact significance level. Here the procedure of Baddeley *et al.* (2017) is summarized and extended for global envelopes as implemented in **GET**.

Let  $M$  denote the chosen measure and  $\alpha$  the chosen significance level. Let  $\mathbf{T}_1$  be the test vector computed from the data.

1. Estimate the parameters  $\theta_1$  of the null model.
2. Simulate  $s_2 - 1$  replicates of the data from the null model with the estimated parameters  $\hat{\theta}_1$ , and compute the test vectors  $\mathbf{T}_1^1 = \mathbf{T}_1, \mathbf{T}_2^1, \dots, \mathbf{T}_{s_2}^1$  (create a ‘`curve_set`’ object of vectors,  $C_1$ ).
3. Simulate another  $s - 1$  replicates of the data from the null model with the parameters  $\hat{\theta}_1$  and estimate the parameters of the null model from each of them  $(\hat{\theta}_i, i = 2, \dots, s)$ ,
4. For every  $i = 2, \dots, s$ , simulate  $s_2 - 1$  replicates from the null model with parameters  $\hat{\theta}_i$ ,

and compute the test vectors  $\mathbf{T}_1^i, \mathbf{T}_2^i, \dots, \mathbf{T}_{s_2}^i$  (create a ‘`curve_set`’ object of vectors,  $C_i$ ).

5. For each set of curves  $C_i$ ,  $i = 1, 2, \dots, s$ , compute the Monte Carlo  $p$  value  $p_i = \sum_{j=1}^{s_2} \mathbf{1}(M_j^i \leq M_1^i) / s_2$ , where  $M_1^i, \dots, M_{s_2}^i$  are the chosen measure computed for  $\mathbf{T}_1^i, \dots, \mathbf{T}_{s_2}^i$ .
6. The adjusted MC  $p$  value is  $p_{\text{adj}} = \sum_{j=1}^s \mathbf{1}(p_i \leq p_1) / s$ .
7. Let  $p^\alpha$  denote the lower  $\alpha$  quantile of the sample  $p_1, \dots, p_s$ . Construct the chosen  $100(1 - p^\alpha)\%$  global envelope from  $\mathbf{T}_1^1, \dots, \mathbf{T}_{s_2}^1$ .

This adjusted test is implemented in the `GET.composite()` function of the **GET** package. Examples can be found in the help page of the function in R and in Section 3.3.

### Affiliation:

Mari Myllymäki

Natural Resources Institute Finland (Luke)

Latokartanonkaari 9

FI-00790 Helsinki, Finland

E-mail: [mari.myllymaki@luke.fi](mailto:mari.myllymaki@luke.fi)

URL: <https://www.luke.fi/en/experts/mari-myllymaki/>

and

Tomáš Mrkvička

Dpt. of Applied Mathematics and Informatics

Faculty of Economics

University of South Bohemia,

Studentská 13

37005 České Budějovice, Czech Republic

E-mail: [mrkvicka.toma@gmail.com](mailto:mrkvicka.toma@gmail.com)

URL: <http://home.ef.jcu.cz/~mrkvicka/>